

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日  
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】<http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事情報の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

改訂一覧は表紙をクリックして直接ご覧になれます。  
改訂一覧は改訂箇所をまとめたものであり、詳細については、  
必ず本文の内容をご確認ください。

# SH7615

ハードウェアマニュアル

ルネサス32ビットRISCマイクロコンピュータ

SuperH™ RISC engineファミリ / SH7600シリーズ

SH7615

HD6417615

## 安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご注意ください。

## 本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (<http://www.renesas.com>) などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。

## 製品に関する一般的注意事項

### 1. NC 端子の処理

【注意】NC 端子には、何も接続しないようにしてください。

NC(Non-Connection) 端子は、内部回路に接続しない場合の他、テスト用端子やノイズ軽減などの目的で使用します。このため、NC 端子には、何も接続しないようにしてください。

### 2. 未使用入力端子の処理

【注意】未使用の入力端子は、ハイまたはローレベルに固定してください。

CMOS 製品の入力端子は、一般にハイインピーダンス入力となっています。未使用端子を開放状態で動作させると、周辺ノイズの誘導により中間レベルが発生し、内部で貫通電流が流れて誤動作を起こす恐れがあります。

未使用の入力端子は、入力をプルアップかプルダウンによって、ハイまたはローレベルに固定してください。

### 3. 初期化前の処置

【注意】電源投入時は、製品の状態は不定です。

すべての電源に電圧が印加され、リセット端子にローレベルが入力されるまでの間、内部回路は不確定であり、レジスタの設定や各端子の出力状態は不定となります。この不定状態によってシステムが誤動作を起こさないようにシステム設計を行ってください。

リセット機能を持つ製品は、電源投入後は、まずリセット動作を実行してください。

### 4. 未定義・リザーブアドレスのアクセス禁止

【注意】未定義・リザーブアドレスのアクセスを禁止します。

未定義・リザーブアドレスは、将来の機能拡張用の他、テスト用レジスタなどが割り付けられています。

これらのレジスタをアクセスしたときの動作および継続する動作については、保証できませんので、アクセスしないようにしてください。



---

# はじめに

---

SH7615 は、ルネサス テクノロジオリジナルの RISC 方式の CPU をコアにして、システム構成に必要な周辺機能を集積した RISC マイクロプロセッサです。

**対象者** このマニュアルは、SH7615 を用いた応用システムを設計するユーザーを対象としています。  
このマニュアルを使用される読者には、電気回路、論理回路、およびマイクロコンピュータに関する基本的な知識を必要とします。

**目的** このマニュアルは、SH7615 のハードウェア機能と電気的特性をユーザーに理解していただくことを目的としています。  
なお、実行命令の詳細については、「SH-1/SH-2/SH-DSP ソフトウェアマニュアル」に記載しておりますので、あわせてご覧ください。

## 読み方

- 対象製品と製品略称の表記について。

製品分類と製品略称	基本製品型名
基本分類	
SH7615	HD6417615

- 機能全体を理解しようとするとき。  
目次にしたがって読んでください。  
本書は、大きく分類すると、CPU、システム制御機能、周辺機能、電気的特性の順に構成されています。
- CPU 機能の詳細を理解したいとき。  
別冊の「SH-1/SH-2/SH-DSP ソフトウェアマニュアル」を参照してください。

**凡例** レジスタ表記 : シリアルコミュニケーションインタフェースモジュールなど、同一または類似した機能が複数チャンネルに存在する場合に、次の表記を使用します。

XXX\_N (XXX は基本レジスタ名称、N はチャンネル番号)

ビット表記 : 左側が上位ビット、右側が下位ビットの順に表記します。

数字の表記 : 2 進数は B'xxxx、16 進数は H'xxxx、10 進数は xxxx。

信号の表記 : ローアクティブの信号にはオーバーバーを付けます。xxxx

関連資料一覧 ウェブ・サイトに最新資料を掲載しています。ご入手の資料が最新版であるかを確認してください。

( <http://www.renesas.com/jpn/> )

- SH7615 に関するユーザズマニュアル

資料名	資料番号
SH7615 ハードウェアマニュアル	本マニュアル
SH-1/SH-2/SH-DSP ソフトウェアマニュアル	RJJ09B0228

- 開発ツール関連ユーザズマニュアル

資料名	資料番号
SuperH RISC engine C/C++ コンパイラ、アセンブラ、最適化リンカージェネレーターユーザズマニュアル	RJJ10B0156
SuperH RISC engine High-performance Embedded Workshop ユーザズマニュアル	RJJ10B0028
SuperH RISC engine High-performance Embedded Workshop チュートリアル	RJJ10B0026

- アプリケーションノート

資料名	資料番号
SuperH RISC engine C/C++ コンパイラパッケージアプリケーションノート	RJJ05B0557

---

## 本版で修正または追加された箇所

---

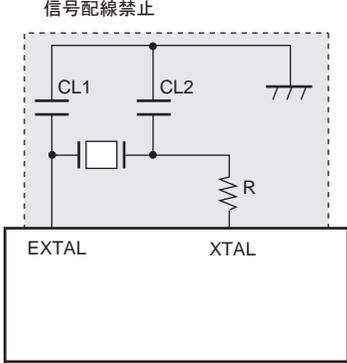
修正項目	ページ	修正内容（詳細はマニュアル参照）
全体		パッケージ（BP-240A, BP-240AV）の追加
1.1 イーサネットコントローラ搭載 SuperH マイコンの特長 表 1.1 特長（5）	5	ダイレクトメモリアクセスコントローラ（DMAC）の説明を追加 ・SDRAM 接続時、31.25MHz まで 1 クロックシングルアドレス転送可能
1.1 イーサネットコントローラ搭載 SuperH マイコンの特長 表 1.1 特長（8）	8	クロック発振器（CPG）の説明追加 ・クロック発振器内蔵
1.3.1 ピン配置 図 1.2 HD6417615ARF/ ARFV ピン配置図（FP- 208C, FP-208CV）	10	説明の修正 HD6417615ARF/ARFV のピン配置を図 1.2 に、HD6417615ARBP/ARBPV のピン配置を図 1.3 に示します。 図 1.2 のタイトルと、パッケージ型名を修正 図 1.2 HD6417615ARF/ARFV ピン配置図（FP-208C, FP-208CV） FP-208C、FP-208CV
図 1.3 HD6417615ARBP/ ARBPV ピン配置図（BP- 240A, BP-240AV）	11	図 1.3 追加

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																																																
1.3.3 端子マルチプレクスの説明 表 1.3 端子マルチプレクスの説明	18 ~ 23	表を修正 <table border="1" data-bbox="747 357 935 815" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>F4</td><td>12</td></tr> <tr><td>E1</td><td>9</td></tr> <tr><td>F2</td><td>11</td></tr> <tr><td>F1</td><td>10</td></tr> <tr><td>H3</td><td>19</td></tr> <tr><td>J1</td><td>21</td></tr> <tr><td>K2</td><td>23</td></tr> <tr><td>K3</td><td>24</td></tr> <tr><td>K4</td><td>25</td></tr> <tr><td>D3</td><td>8</td></tr> <tr><td>G1</td><td>13</td></tr> <tr><td>G2</td><td>14</td></tr> <tr><td>G3</td><td>15</td></tr> <tr><td>G4</td><td>16</td></tr> <tr><td>H1</td><td>17</td></tr> <tr><td>C1</td><td>5</td></tr> <tr><td>B2</td><td>1</td></tr> <tr><td>D2</td><td>2</td></tr> <tr><td>B1</td><td>3</td></tr> <tr><td>C2</td><td>4</td></tr> <tr><td>L3</td><td>27</td></tr> </tbody> </table> <table border="1" data-bbox="747 853 935 1391" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>K18</td><td>131</td></tr> <tr><td>H16</td><td>138</td></tr> <tr><td>J19</td><td>137</td></tr> <tr><td>J18</td><td>136</td></tr> <tr><td>J16</td><td>135</td></tr> <tr><td>J17</td><td>134</td></tr> <tr><td>F19</td><td>148</td></tr> <tr><td>G19</td><td>145</td></tr> <tr><td>R17</td><td>115</td></tr> <tr><td>L18</td><td>128</td></tr> <tr><td>P19</td><td>117</td></tr> <tr><td>P18</td><td>118</td></tr> <tr><td>N19</td><td>119</td></tr> <tr><td>N18</td><td>120</td></tr> <tr><td>N17</td><td>121</td></tr> <tr><td>N16</td><td>122</td></tr> <tr><td>M19</td><td>123</td></tr> <tr><td>M18</td><td>124</td></tr> <tr><td>M17</td><td>125</td></tr> <tr><td>M16</td><td>126</td></tr> <tr><td>L19</td><td>127</td></tr> <tr><td>L16</td><td>129</td></tr> <tr><td>K19</td><td>133</td></tr> <tr><td>H17</td><td>139</td></tr> <tr><td>H18</td><td>140</td></tr> </tbody> </table>	BP-240AAV 番号	FP-208CV 番号	F4	12	E1	9	F2	11	F1	10	H3	19	J1	21	K2	23	K3	24	K4	25	D3	8	G1	13	G2	14	G3	15	G4	16	H1	17	C1	5	B2	1	D2	2	B1	3	C2	4	L3	27	BP-240AAV 番号	FP-208CV 番号	K18	131	H16	138	J19	137	J18	136	J16	135	J17	134	F19	148	G19	145	R17	115	L18	128	P19	117	P18	118	N19	119	N18	120	N17	121	N16	122	M19	123	M18	124	M17	125	M16	126	L19	127	L16	129	K19	133	H17	139	H18	140
BP-240AAV 番号	FP-208CV 番号																																																																																																	
F4	12																																																																																																	
E1	9																																																																																																	
F2	11																																																																																																	
F1	10																																																																																																	
H3	19																																																																																																	
J1	21																																																																																																	
K2	23																																																																																																	
K3	24																																																																																																	
K4	25																																																																																																	
D3	8																																																																																																	
G1	13																																																																																																	
G2	14																																																																																																	
G3	15																																																																																																	
G4	16																																																																																																	
H1	17																																																																																																	
C1	5																																																																																																	
B2	1																																																																																																	
D2	2																																																																																																	
B1	3																																																																																																	
C2	4																																																																																																	
L3	27																																																																																																	
BP-240AAV 番号	FP-208CV 番号																																																																																																	
K18	131																																																																																																	
H16	138																																																																																																	
J19	137																																																																																																	
J18	136																																																																																																	
J16	135																																																																																																	
J17	134																																																																																																	
F19	148																																																																																																	
G19	145																																																																																																	
R17	115																																																																																																	
L18	128																																																																																																	
P19	117																																																																																																	
P18	118																																																																																																	
N19	119																																																																																																	
N18	120																																																																																																	
N17	121																																																																																																	
N16	122																																																																																																	
M19	123																																																																																																	
M18	124																																																																																																	
M17	125																																																																																																	
M16	126																																																																																																	
L19	127																																																																																																	
L16	129																																																																																																	
K19	133																																																																																																	
H17	139																																																																																																	
H18	140																																																																																																	

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																																																																						
1.3.3 端子マルチプレクスの説明 表 1.3 端子マルチプレクスの説明	18 ~ 23	表を修正 <table border="1" data-bbox="747 357 935 869"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>T19</td><td>111</td></tr> <tr><td>U18</td><td>108</td></tr> <tr><td>V19</td><td>107</td></tr> <tr><td>T18</td><td>106</td></tr> <tr><td>V18</td><td>105</td></tr> <tr><td>V16</td><td>104</td></tr> <tr><td>W18</td><td>103</td></tr> <tr><td>V17</td><td>102</td></tr> <tr><td>V15</td><td>100</td></tr> <tr><td>U16</td><td>98</td></tr> <tr><td>W15</td><td>97</td></tr> <tr><td>U15</td><td>96</td></tr> <tr><td>W14</td><td>95</td></tr> <tr><td>V14</td><td>94</td></tr> <tr><td>U14</td><td>93</td></tr> <tr><td>T14</td><td>92</td></tr> <tr><td>V13</td><td>90</td></tr> <tr><td>W12</td><td>88</td></tr> <tr><td>V12</td><td>87</td></tr> <tr><td>U12</td><td>86</td></tr> <tr><td>T12</td><td>85</td></tr> <tr><td>W11</td><td>84</td></tr> <tr><td>V11</td><td>83</td></tr> <tr><td>U11</td><td>82</td></tr> <tr><td>T10</td><td>80</td></tr> </tbody> </table> <table border="1" data-bbox="747 904 935 1555"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>U9</td><td>77</td></tr> <tr><td>W9</td><td>75</td></tr> <tr><td>T8</td><td>74</td></tr> <tr><td>U8</td><td>73</td></tr> <tr><td>V8</td><td>72</td></tr> <tr><td>W8</td><td>71</td></tr> <tr><td>U7</td><td>70</td></tr> <tr><td>W7</td><td>68</td></tr> <tr><td>V6</td><td>65</td></tr> <tr><td>W6</td><td>64</td></tr> <tr><td>U5</td><td>63</td></tr> <tr><td>U3</td><td>62</td></tr> <tr><td>W4</td><td>59</td></tr> <tr><td>W3</td><td>57</td></tr> <tr><td>V3</td><td>56</td></tr> <tr><td>W2</td><td>55</td></tr> <tr><td>V4</td><td>54</td></tr> <tr><td>V2</td><td>53</td></tr> <tr><td>T2</td><td>51</td></tr> <tr><td>U2</td><td>49</td></tr> <tr><td>U1</td><td>48</td></tr> <tr><td>R2</td><td>47</td></tr> <tr><td>T1</td><td>46</td></tr> <tr><td>R1</td><td>44</td></tr> <tr><td>R3</td><td>43</td></tr> <tr><td>P2</td><td>41</td></tr> <tr><td>P3</td><td>40</td></tr> <tr><td>P4</td><td>39</td></tr> <tr><td>N1</td><td>38</td></tr> <tr><td>N2</td><td>37</td></tr> <tr><td>N3</td><td>36</td></tr> <tr><td>M1</td><td>34</td></tr> </tbody> </table>	BP-240AAV 番号	FP-208CV 番号	T19	111	U18	108	V19	107	T18	106	V18	105	V16	104	W18	103	V17	102	V15	100	U16	98	W15	97	U15	96	W14	95	V14	94	U14	93	T14	92	V13	90	W12	88	V12	87	U12	86	T12	85	W11	84	V11	83	U11	82	T10	80	BP-240AAV 番号	FP-208CV 番号	U9	77	W9	75	T8	74	U8	73	V8	72	W8	71	U7	70	W7	68	V6	65	W6	64	U5	63	U3	62	W4	59	W3	57	V3	56	W2	55	V4	54	V2	53	T2	51	U2	49	U1	48	R2	47	T1	46	R1	44	R3	43	P2	41	P3	40	P4	39	N1	38	N2	37	N3	36	M1	34
BP-240AAV 番号	FP-208CV 番号																																																																																																																							
T19	111																																																																																																																							
U18	108																																																																																																																							
V19	107																																																																																																																							
T18	106																																																																																																																							
V18	105																																																																																																																							
V16	104																																																																																																																							
W18	103																																																																																																																							
V17	102																																																																																																																							
V15	100																																																																																																																							
U16	98																																																																																																																							
W15	97																																																																																																																							
U15	96																																																																																																																							
W14	95																																																																																																																							
V14	94																																																																																																																							
U14	93																																																																																																																							
T14	92																																																																																																																							
V13	90																																																																																																																							
W12	88																																																																																																																							
V12	87																																																																																																																							
U12	86																																																																																																																							
T12	85																																																																																																																							
W11	84																																																																																																																							
V11	83																																																																																																																							
U11	82																																																																																																																							
T10	80																																																																																																																							
BP-240AAV 番号	FP-208CV 番号																																																																																																																							
U9	77																																																																																																																							
W9	75																																																																																																																							
T8	74																																																																																																																							
U8	73																																																																																																																							
V8	72																																																																																																																							
W8	71																																																																																																																							
U7	70																																																																																																																							
W7	68																																																																																																																							
V6	65																																																																																																																							
W6	64																																																																																																																							
U5	63																																																																																																																							
U3	62																																																																																																																							
W4	59																																																																																																																							
W3	57																																																																																																																							
V3	56																																																																																																																							
W2	55																																																																																																																							
V4	54																																																																																																																							
V2	53																																																																																																																							
T2	51																																																																																																																							
U2	49																																																																																																																							
U1	48																																																																																																																							
R2	47																																																																																																																							
T1	46																																																																																																																							
R1	44																																																																																																																							
R3	43																																																																																																																							
P2	41																																																																																																																							
P3	40																																																																																																																							
P4	39																																																																																																																							
N1	38																																																																																																																							
N2	37																																																																																																																							
N3	36																																																																																																																							
M1	34																																																																																																																							

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																																																																								
1.3.3 端子マルチプレクスの説明 表 1.3 端子マルチプレクスの説明	18 ~ 23	表を修正 <table border="1" data-bbox="747 353 935 927"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>L1</td><td>30</td></tr> <tr><td>M4</td><td>31</td></tr> <tr><td>L2</td><td>29</td></tr> <tr><td>L4</td><td>28</td></tr> <tr><td>M3</td><td>32</td></tr> <tr><td>E2</td><td>6</td></tr> <tr><td>A5</td><td>201</td></tr> <tr><td>A8</td><td>192</td></tr> <tr><td>A4</td><td>203</td></tr> <tr><td>A2</td><td>207</td></tr> <tr><td>B3</td><td>206</td></tr> <tr><td>A3</td><td>205</td></tr> <tr><td>B5</td><td>204</td></tr> <tr><td>B4</td><td>208</td></tr> <tr><td>A9</td><td>188</td></tr> <tr><td>C6</td><td>197</td></tr> <tr><td>D6</td><td>196</td></tr> <tr><td>A7</td><td>195</td></tr> <tr><td>B7</td><td>194</td></tr> <tr><td>B9</td><td>187</td></tr> <tr><td>C8</td><td>190</td></tr> <tr><td>D8</td><td>189</td></tr> <tr><td>A6</td><td>199</td></tr> <tr><td>B6</td><td>198</td></tr> <tr><td>G17</td><td>143</td></tr> <tr><td>G18</td><td>144</td></tr> <tr><td>H19</td><td>141</td></tr> <tr><td>G16</td><td>142</td></tr> </tbody> </table> <table border="1" data-bbox="747 967 935 1657"> <thead> <tr> <th>BP-240AAV 番号</th> <th>FP-208CV 番号</th> </tr> </thead> <tbody> <tr><td>D19</td><td>151</td></tr> <tr><td>E18</td><td>152</td></tr> <tr><td>C19</td><td>153</td></tr> <tr><td>C18</td><td>154</td></tr> <tr><td>D18</td><td>156</td></tr> <tr><td>B16</td><td>158</td></tr> <tr><td>A18</td><td>159</td></tr> <tr><td>B17</td><td>160</td></tr> <tr><td>A17</td><td>161</td></tr> <tr><td>B15</td><td>162</td></tr> <tr><td>A16</td><td>163</td></tr> <tr><td>C16</td><td>164</td></tr> <tr><td>A15</td><td>165</td></tr> <tr><td>C17</td><td>166</td></tr> <tr><td>A14</td><td>168</td></tr> <tr><td>C14</td><td>170</td></tr> <tr><td>D14</td><td>171</td></tr> <tr><td>A13</td><td>172</td></tr> <tr><td>B13</td><td>173</td></tr> <tr><td>C13</td><td>174</td></tr> <tr><td>A12</td><td>175</td></tr> <tr><td>B12</td><td>176</td></tr> <tr><td>C12</td><td>177</td></tr> <tr><td>D12</td><td>178</td></tr> <tr><td>B11</td><td>180</td></tr> <tr><td>C11</td><td>182</td></tr> <tr><td>B10</td><td>183</td></tr> <tr><td>C10</td><td>184</td></tr> <tr><td>A10</td><td>185</td></tr> <tr><td>C9</td><td>186</td></tr> </tbody> </table>	BP-240AAV 番号	FP-208CV 番号	L1	30	M4	31	L2	29	L4	28	M3	32	E2	6	A5	201	A8	192	A4	203	A2	207	B3	206	A3	205	B5	204	B4	208	A9	188	C6	197	D6	196	A7	195	B7	194	B9	187	C8	190	D8	189	A6	199	B6	198	G17	143	G18	144	H19	141	G16	142	BP-240AAV 番号	FP-208CV 番号	D19	151	E18	152	C19	153	C18	154	D18	156	B16	158	A18	159	B17	160	A17	161	B15	162	A16	163	C16	164	A15	165	C17	166	A14	168	C14	170	D14	171	A13	172	B13	173	C13	174	A12	175	B12	176	C12	177	D12	178	B11	180	C11	182	B10	183	C10	184	A10	185	C9	186
BP-240AAV 番号	FP-208CV 番号																																																																																																																									
L1	30																																																																																																																									
M4	31																																																																																																																									
L2	29																																																																																																																									
L4	28																																																																																																																									
M3	32																																																																																																																									
E2	6																																																																																																																									
A5	201																																																																																																																									
A8	192																																																																																																																									
A4	203																																																																																																																									
A2	207																																																																																																																									
B3	206																																																																																																																									
A3	205																																																																																																																									
B5	204																																																																																																																									
B4	208																																																																																																																									
A9	188																																																																																																																									
C6	197																																																																																																																									
D6	196																																																																																																																									
A7	195																																																																																																																									
B7	194																																																																																																																									
B9	187																																																																																																																									
C8	190																																																																																																																									
D8	189																																																																																																																									
A6	199																																																																																																																									
B6	198																																																																																																																									
G17	143																																																																																																																									
G18	144																																																																																																																									
H19	141																																																																																																																									
G16	142																																																																																																																									
BP-240AAV 番号	FP-208CV 番号																																																																																																																									
D19	151																																																																																																																									
E18	152																																																																																																																									
C19	153																																																																																																																									
C18	154																																																																																																																									
D18	156																																																																																																																									
B16	158																																																																																																																									
A18	159																																																																																																																									
B17	160																																																																																																																									
A17	161																																																																																																																									
B15	162																																																																																																																									
A16	163																																																																																																																									
C16	164																																																																																																																									
A15	165																																																																																																																									
C17	166																																																																																																																									
A14	168																																																																																																																									
C14	170																																																																																																																									
D14	171																																																																																																																									
A13	172																																																																																																																									
B13	173																																																																																																																									
C13	174																																																																																																																									
A12	175																																																																																																																									
B12	176																																																																																																																									
C12	177																																																																																																																									
D12	178																																																																																																																									
B11	180																																																																																																																									
C11	182																																																																																																																									
B10	183																																																																																																																									
C10	184																																																																																																																									
A10	185																																																																																																																									
C9	186																																																																																																																									

修正項目	ページ	修正内容（詳細はマニュアル参照）
1.4 処理状態 図 1.4 処理状態遷移図	24	図番号の修正
(5) バス権解放状態	25	<p>説明の修正</p> <p>(i) CPU がバス権を要求したデバイスにバスを解放する状態です。</p> <p>(ii) マニュアルリセット信号アサート中のバス権開放  マニュアルリセット信号アサート中（RES=Low, NMI=Low）は、アービトレーション要求（BRLS 入力）を受け付けません。BRLS 信号が引き続きアサートされている場合には、そのままバス解放状態を保ちます（BGR 信号をアサート）。  マニュアルリセット信号アサート中、バス解放状態から BRLS 信号がネゲートされた場合には、本 LSI はバスの使用を再開します（BGR 信号をネゲート）。</p>
2.6 使用上の注意事項	85, 86	<p>説明を修正</p> <p>1. PCLR A0 : A0 レジスタを 0 クリアする。</p> <p>3. を追加</p>
3.2.3 水晶発振子の接続方法 図 3.2 水晶発振子の接続例	93	<p>図の注を修正</p> <p>・CKIO 端子はクロックモード 0、1、2 で出力、または、ハイインピーダンスとなります。また、クロックモード 3 ではハイインピーダンスとなります。</p>
3.2.4 外部クロックの入力方法 (1) EXTAL 端子からのクロック入力 図 3.3 外部クロック入力方法	94	<p>図の注を修正</p> <p>・CKIO 端子はクロックモード 0、1、2 で出力、またはハイインピーダンスとなります。また、クロックモード 3 ではハイインピーダンスとなります。</p>

修正項目	ページ	修正内容 (詳細はマニュアル参照)															
3.2.7 ボード設計上の注意事項 (1) 水晶発振子使用時の注意 図 3.5 水晶発振子使用時の注意	103	説明の修正  図の説明を追加   <p>CL1、CL2=20~22pF R=0Ω 発振周波数15.625MHz 水晶発振子: 九州電通(株) 型名: HC-49/u</p> <p>【注】 CL1、CL2およびダンピング抵抗は使用する水晶発振子のメーカーと評価検討の上、値を決めてください。</p>															
(2) バイパスコンデンサについての注意	103, 104	説明の修正 <FP-280C/CV 使用時の Vss-Vcc ペア > <BP-240A/AV 使用時の Vss-Vcc ペア > < BP-240A/AV 使用時の Vss-Vcc ペア > PLL系: E1、F1、F2、F4 3V デジタル系: H4-H2、K1-J2、N4-M2、T3-P1、W1-V1、V5-U4、W5-U6、V7-T6、V10-V9、U10-W10、W13-U13、W17-W16、T17-U19、R19-R18、U17-R16、L17-K17、E17-F17、F18-E19 5V デジタル系: B18-B19、B14-C15、D11-A11、B8-C7、C4-C5															
7.2.1 バスコントロールレジスタ1 (BCR1) ビット 12: CS2 空間のエンディアン指定 (A2ENDIAN)	222	注の追加 【注】 リトルエンディアン機能を使うことで、データの並び替えに時間がかかることはありません。															
ビット 3: CS4 空間のエンディアン指定 (A4ENDIAN)	223	【注】 リトルエンディアン機能を使うことで、データの並び替えに余分な時間がかかることはありません。															
7.2.7 個別メモリコントロールレジスタ (MCR) ビット 1、15: RAS プリチャージ期間 (TRP1、TRP0)	232	表の追加  [シンクロナスDRAMインタフェース] <table border="1" data-bbox="504 1445 1190 1576"> <thead> <tr> <th>ビット1 TRP1</th> <th>ビット15 TRP0</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1サイクル (初期値)</td> </tr> <tr> <td>0</td> <td>1</td> <td>2サイクル</td> </tr> <tr> <td>1</td> <td>0</td> <td>3サイクル</td> </tr> <tr> <td>1</td> <td>1</td> <td>4サイクル</td> </tr> </tbody> </table>	ビット1 TRP1	ビット15 TRP0	説明	0	0	1サイクル (初期値)	0	1	2サイクル	1	0	3サイクル	1	1	4サイクル
ビット1 TRP1	ビット15 TRP0	説明															
0	0	1サイクル (初期値)															
0	1	2サイクル															
1	0	3サイクル															
1	1	4サイクル															

修正項目	ページ	修正内容（詳細はマニュアル参照）
7.2.7 個別メモリコントロールレジスタ（MCR） ビット7、5、4 アドレスマルチプレクス（AMX2～0）	235	シンクロナス DRAM インタフェースの表を差し替え
7.5.11 64M ビットシンクロナス DRAM（2Mワード×32ビット品）との接続方法 （3）シンクロナス DRAM に対するモード設定	278	説明の修正 シンクロナス DRAM に対してモードを設定する場合には、CPU から X+H'FFFF0000、または X+ H'FFFF8000 番地に書き込みを行ってください（X は、設定した値）。X+ H'FFFF0000 と X+ H'FFFF8000 のどちらを使うかは、シンクロナス DRAM の仕様に依存します。
図 7.35 128M ビットシンクロナス DRAM（4Mワード×32ビット品）との接続例 図 7.36 128M ビットシンクロナス DRAM（8Mワード×16ビット品）との接続例 図 7.37 256M ビットシンクロナス DRAM（8Mワード×32ビット品）との接続例	279 ～ 281	図を追加
7.6.1 DRAM 直結方式 図 7.38 DRAM 接続例（32ビットデータ幅） 図 7.39 DRAM 接続例（16ビットデータ幅）	282、 283	図番号変更
7.6.3 基本タイミング 図 7.40 基本アクセスタイミング	284	図番号変更
7.6.4 ウェイトステート制御 図 7.41 ウェイトステートタイミング 図 7.42 外部ウェイトステートタイミング	285、 286	図番号変更
7.6.5 バーストアクセス 図 7.43 バーストアクセスタイミング 図 7.44 $\overline{\text{RAS}}$ ダウンモード、同一ロウアクセスタイミング 図 7.45 $\overline{\text{RAS}}$ ダウンモード、異なるロウアクセスタイミング	287 ～ 289	図番号変更

修正項目	ページ	修正内容（詳細はマニュアル参照）
7.6.6 EDO モード 図 7.46 EDO DRAM 接続例（32 ビットデータ幅） 図 7.47 EDO DRAM 接続例（16 ビットデータ幅） 図 7.48 DRAM EDO モードのノーマルアクセスタイミング 図 7.49 DRAM EDO モードのバーストアクセスタイミング	290 ～ 293	図番号変更
7.6.7 DMA シングル転送 図 7.50 DMA シングル転送モードでのライトサイクルタイミング（RAS ダウンモード、同一ロウアドレス）	294	図番号変更
7.6.8 リフレッシュ 図 7.51 DRAM CAS ピフォ RAS リフレッシュサイクルタイミング 図 7.52 DRAM セルフリフレッシュサイクルタイミング	295, 296	図番号変更
7.7 バースト ROM インタフェース 図 7.53 データ幅とバースト ROM アクセス（ウェイト数 1） 図 7.54 バースト ROM ニブルアクセス（2 ウェイトステート） 図 7.55 バースト ROM ニブルアクセス（ウェイトなし）	297, 298	図番号変更
7.8 アクセスサイクル間ウェイト 図 7.56 アクセスサイクル間ウェイト	300	図番号変更
7.9 バスアービトレーション 図 7.57(a) バスアービトレーションタイミング（E-DMAC リード DMAC16 バイト転送 CPU リード） 図 7.57(b) バスアービトレーションタイミング（E-DMAC ライト DMAC16 バイト転送 CPU リード）	303, 304	図番号変更
7.9.1 マスタモード 図 7.58 バスアービトレーション	306	図番号変更

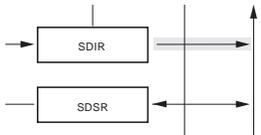
修正項目	ページ	修正内容（詳細はマニュアル参照）								
7.10.3 STATS1、0 端子について 図 7.59 STATS 出力タイミング	309	図番号変更								
7.10.4 BUSHIZ の仕様について 図 7.60 BUSHIZ バスタイミング	310	図番号変更								
7.11.1 DMAC 使用時のシンクロナス DRAM ライト直後の通常空間アクセス時の注意事項 図 7.61 シンクロナス DRAM ライト直後の通常空間アクセス	312	図番号変更								
7.11.3 シンクロナス DRAM へのデータ誤出力防止について	313	項を追加								
8.2.1 キャッシュコントロールレジスタ (CCR) ビット 4: キャッシュパージビット (CP)	317	説明を追加。 ...CP ビットは 0 に復帰します。CP ビットは読み出し時には常に 0 です。 初期化終了はキャッシュをリードしてご確認ください。								
9.2.1 EtherC モードレジスタ (ECMR) ビット 1: デュプレックスモード (DM)	341	説明を追加。 DM ビットには、物理層 LSI が検出した全二重 / 半二重のモードの情報を反映させる必要があります。物理層 LSI の通信モードと本 LSI の設定が合致していない場合、転送効率の低下、データの衝突が起こる要因となります。								
9.2.8 PHY 部ステータスレジスタ (PSR)	350	注を追加 LMON ビットは LNKSTA 端子が High レベル時 0、Low レベル時 1 となります。								
9.3.1 送信動作	363	説明を修正 (4) フレーム間隔時間 (10Base では 9.6 $\mu$ s、100Base では 0.96 $\mu$ s) を経た後、アイドル状態に遷移し、以後送信データがあれば送信を継続します。								
9.5 使用上の注意	373	節を追加								
10.2.2 E-DMAC 送信要求レジスタ (EDTRR) ビット 0: 送信要求 (TR)	380	【注】を追加								
10.2.3 E-DMAC 受信要求レジスタ (EDRRR) ビット 0: 受信要求 (RR)	381	【注】を追加								
10.2.6 EtherC/E-DMAC スタータスレジスタ (EESR)	384	ビットの図修正 <table border="1" style="margin-left: auto; margin-right: auto;"><tr><td style="text-align: center;">26</td><td style="text-align: center;">25</td></tr><tr><td style="text-align: center;">TABT</td><td style="text-align: center;">RABT</td></tr><tr><td style="text-align: center;">0</td><td style="text-align: center;">0</td></tr><tr><td style="text-align: center;">R/W</td><td style="text-align: center;">R/W</td></tr></table>	26	25	TABT	RABT	0	0	R/W	R/W
26	25									
TABT	RABT									
0	0									
R/W	R/W									

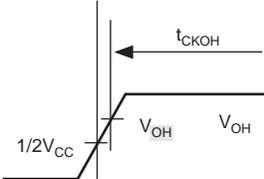
修正項目	ページ	修正内容（詳細はマニュアル参照）												
10.2.6 EtherC/E-DMAC スタータスレジスタ (EESR)	385	<p><b>ビットの説明修正と追加</b></p> <ul style="list-style-type: none"> <li>・ <b>ビット 31～27：予約ビット</b> 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。</li> </ul> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <ul style="list-style-type: none"> <li>・ <b>ビット 26：送信アボート検出 (TABT)</b> 送信アボートを検出したことを示します。</li> </ul> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">ビット 26 TABT</th> <th style="width: 70%;">説明</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>送信アボート未検出 (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>送信アボート検出</td> </tr> </tbody> </table> <p>下記のいずれかのビットが1にセットされると本ビットがセットされます。</p> <ul style="list-style-type: none"> <li>EESR ビット12：送信フレーム長異常 (ITF)</li> <li>EESR ビット11：キャリア未検出 (CND)</li> <li>EESR ビット10：キャリア消失検出 (DLC)</li> <li>EESR ビット9：遅延衝突検出 (CD)</li> </ul> </div> <ul style="list-style-type: none"> <li>・ <b>ビット 25：受信アボート検出 (RABT)</b> 受信アボートを検出したことを示します。</li> </ul> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">ビット 25 RABT</th> <th style="width: 70%;">説明</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>受信アボート未検出 (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>受信アボート検出</td> </tr> </tbody> </table> <p>下記のいずれかのビットが1にセットされると本ビットがセットされます。</p> <ul style="list-style-type: none"> <li>EESR ビット4：端数ビットフレーム受信 (RRF)</li> <li>EESR ビット3：ロングフレーム受信エラー (RTLFE)</li> <li>EESR ビット2：ショートフレーム受信エラー (RTSFE)</li> <li>EESR ビット1：PHY-LSI受信エラー (PRE)</li> <li>EESR ビット0：受信フレームCRCエラー (CERF)</li> </ul> </div>	ビット 26 TABT	説明	0	送信アボート未検出 (初期値)	1	送信アボート検出	ビット 25 RABT	説明	0	受信アボート未検出 (初期値)	1	受信アボート検出
ビット 26 TABT	説明													
0	送信アボート未検出 (初期値)													
1	送信アボート検出													
ビット 25 RABT	説明													
0	受信アボート未検出 (初期値)													
1	受信アボート検出													
	388	<p><b>ビット 9 の説明修正</b></p> <ul style="list-style-type: none"> <li>・ <b>ビット 9：遅延衝突検出 (CD)</b> フレーム送信中に衝突を検出したことを示します。</li> </ul> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">ビット 9 CD</th> <th style="width: 70%;">説明</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>遅延衝突未検出 (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>遅延衝突検出 (割り込み要因)</td> </tr> </tbody> </table>	ビット 9 CD	説明	0	遅延衝突未検出 (初期値)	1	遅延衝突検出 (割り込み要因)						
ビット 9 CD	説明													
0	遅延衝突未検出 (初期値)													
1	遅延衝突検出 (割り込み要因)													

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																																																																																																								
10.2.8 送受信ステータス コピー指示レジスタ ( TRSCER )	396	<p>頂の修正</p> <p>EtherC/E-DMACステータスレジスタ (EESR) のビット7で報告される、マルチキャストアドレスフレーム受信ステータス情報を当該ディスクリプタに反映するか否かを指示します。本レジスタのビットはEtherC/E-DMACステータスレジスタEESR[7]に対応し、ビット7に0を設定すると、受信ステータス (EESR[7]) は受信ディスクリプタのRFEビットに反映されず、1を設定すると、該当する要因が発生してもディスクリプタに反映されません。LSIのリセット後は、ビット7は0に設定されています。</p> <table border="1" data-bbox="587 473 1094 502"> <tr> <td>ビット</td> <td>31</td> <td>30</td> <td>29</td> <td>28</td> <td>27</td> <td>26</td> <td>25</td> <td>24</td> </tr> <tr> <td></td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>初期値</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W:</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> </tr> </table> <table border="1" data-bbox="587 556 1094 585"> <tr> <td>ビット</td> <td>23</td> <td>22</td> <td>21</td> <td>20</td> <td>19</td> <td>18</td> <td>17</td> <td>16</td> </tr> <tr> <td></td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>初期値</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W:</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> </tr> </table> <table border="1" data-bbox="587 639 1094 668"> <tr> <td>ビット</td> <td>15</td> <td>14</td> <td>13</td> <td>12</td> <td>11</td> <td>10</td> <td>9</td> <td>8</td> </tr> <tr> <td></td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>初期値</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W:</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> </tr> </table> <table border="1" data-bbox="587 722 1094 751"> <tr> <td>ビット</td> <td>7</td> <td>6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td></td> <td>RMAFCE</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>初期値</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W:</td> <td>R/W</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> </tr> </table> <ul data-bbox="577 813 1094 871" style="list-style-type: none"> <li>・ ビット31～8: 予約ビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。</li> <li>・ ビット7: マルチキャストアドレスフレーム受信 (RMAF) ビットコピー指示 (RMAFCE)</li> </ul> <table border="1" data-bbox="570 884 1094 952"> <thead> <tr> <th>ビット7</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>RMAFCE</td> <td></td> </tr> <tr> <td>0</td> <td>RMAFビットのステータスを受信ディスクリプタのRFS7ビットに反映する。</td> </tr> <tr> <td>1</td> <td>該当する要因が発生しても受信ディスクリプタのRFS7ビットに反映しない。</td> </tr> </tbody> </table> <ul data-bbox="577 981 1094 1010" style="list-style-type: none"> <li>・ ビット6～0: 予約ビット 読み出すと常に0が読み出されず。書き込む値も常に0にしてください。</li> </ul>	ビット	31	30	29	28	27	26	25	24		—	—	—	—	—	—	—	—	初期値	0	0	0	0	0	0	0	0	R/W:	R	R	R	R	R	R	R	R	ビット	23	22	21	20	19	18	17	16		—	—	—	—	—	—	—	—	初期値	0	0	0	0	0	0	0	0	R/W:	R	R	R	R	R	R	R	R	ビット	15	14	13	12	11	10	9	8		—	—	—	—	—	—	—	—	初期値	0	0	0	0	0	0	0	0	R/W:	R	R	R	R	R	R	R	R	ビット	7	6	5	4	3	2	1	0		RMAFCE	—	—	—	—	—	—	—	初期値	0	0	0	0	0	0	0	0	R/W:	R/W	R	R	R	R	R	R	R	ビット7	説明	RMAFCE		0	RMAFビットのステータスを受信ディスクリプタのRFS7ビットに反映する。	1	該当する要因が発生しても受信ディスクリプタのRFS7ビットに反映しない。
ビット	31	30	29	28	27	26	25	24																																																																																																																																																		
	—	—	—	—	—	—	—	—																																																																																																																																																		
初期値	0	0	0	0	0	0	0	0																																																																																																																																																		
R/W:	R	R	R	R	R	R	R	R																																																																																																																																																		
ビット	23	22	21	20	19	18	17	16																																																																																																																																																		
	—	—	—	—	—	—	—	—																																																																																																																																																		
初期値	0	0	0	0	0	0	0	0																																																																																																																																																		
R/W:	R	R	R	R	R	R	R	R																																																																																																																																																		
ビット	15	14	13	12	11	10	9	8																																																																																																																																																		
	—	—	—	—	—	—	—	—																																																																																																																																																		
初期値	0	0	0	0	0	0	0	0																																																																																																																																																		
R/W:	R	R	R	R	R	R	R	R																																																																																																																																																		
ビット	7	6	5	4	3	2	1	0																																																																																																																																																		
	RMAFCE	—	—	—	—	—	—	—																																																																																																																																																		
初期値	0	0	0	0	0	0	0	0																																																																																																																																																		
R/W:	R/W	R	R	R	R	R	R	R																																																																																																																																																		
ビット7	説明																																																																																																																																																									
RMAFCE																																																																																																																																																										
0	RMAFビットのステータスを受信ディスクリプタのRFS7ビットに反映する。																																																																																																																																																									
1	該当する要因が発生しても受信ディスクリプタのRFS7ビットに反映しない。																																																																																																																																																									
10.2.10 送信 FIFO しきい値指定レジスタ (TFTR)	399	制限事項を追加																																																																																																																																																								

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																												
10.2.13 E-DMAC 動作制御レジスタ (EDOCR)	403, 404	<p>ビット 3-0 の表と説明を修正</p> <table border="1" style="margin-left: 20px;"> <tr> <td>ビット</td> <td>7</td> <td>6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td></td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>FEC</td> <td>AEC</td> <td>EDH</td> <td>—</td> </tr> <tr> <td>初期値</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W:</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R</td> </tr> </table> <ul style="list-style-type: none"> <li>・ ビット31~4: 予約ビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。</li> <li>・ ビット3: FIFOエラー制御 (FEC) 送信FIFOにおけるアンダフローあるいは受信FIFOにおけるオーバフロー発生時のE-DMACの動作を指定します。</li> </ul> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ビット3</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>FEC</td> <td></td> </tr> <tr> <td>0</td> <td>アンダフローあるいはオーバフローが発生してもE-DMACの動作を継続します。 (初期値)</td> </tr> <tr> <td>1</td> <td>アンダフローあるいはオーバフローが発生するとE-DMACの動作を停止します。</td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>・ ビット2: アドレスエラー制御 (AEC) E-DMACが転送しようとしたメモリアドレスが不正であったことを示します。</li> </ul> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ビット2</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>AEC</td> <td></td> </tr> <tr> <td>0</td> <td>不正なメモリアドレスを検出していない (正常動作)。 (初期値)</td> </tr> <tr> <td>1</td> <td>不正なメモリアドレスを検出した。0を書き込むことによってクリアすることができません。</td> </tr> </tbody> </table> <p>【参考】本エラーは、E-DMACが使用するディスクリプタ内のメモリアドレス設定が不正である場合に発生します。</p> <ul style="list-style-type: none"> <li>・ ビット1: E-DMAC停止 (EDH) 本LSIのNMI入力端子がアサートされるとE-DMACは動作を停止します。</li> </ul> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ビット1</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>EDH</td> <td></td> </tr> <tr> <td>0</td> <td>E-DMACは正常に動作中です。 (初期値)</td> </tr> <tr> <td>1</td> <td>NMI端子がアサートされたためE-DMACの動作を停止します。0を書き込むことによってE-DMACは動作を再開します。</td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>・ ビット0: 予約ビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。</li> </ul>	ビット	7	6	5	4	3	2	1	0		—	—	—	—	FEC	AEC	EDH	—	初期値	0	0	0	0	0	0	0	0	R/W:	R	R	R	R	R/W	R/W	R/W	R	ビット3	説明	FEC		0	アンダフローあるいはオーバフローが発生してもE-DMACの動作を継続します。 (初期値)	1	アンダフローあるいはオーバフローが発生するとE-DMACの動作を停止します。	ビット2	説明	AEC		0	不正なメモリアドレスを検出していない (正常動作)。 (初期値)	1	不正なメモリアドレスを検出した。0を書き込むことによってクリアすることができません。	ビット1	説明	EDH		0	E-DMACは正常に動作中です。 (初期値)	1	NMI端子がアサートされたためE-DMACの動作を停止します。0を書き込むことによってE-DMACは動作を再開します。
ビット	7	6	5	4	3	2	1	0																																																						
	—	—	—	—	FEC	AEC	EDH	—																																																						
初期値	0	0	0	0	0	0	0	0																																																						
R/W:	R	R	R	R	R/W	R/W	R/W	R																																																						
ビット3	説明																																																													
FEC																																																														
0	アンダフローあるいはオーバフローが発生してもE-DMACの動作を継続します。 (初期値)																																																													
1	アンダフローあるいはオーバフローが発生するとE-DMACの動作を停止します。																																																													
ビット2	説明																																																													
AEC																																																														
0	不正なメモリアドレスを検出していない (正常動作)。 (初期値)																																																													
1	不正なメモリアドレスを検出した。0を書き込むことによってクリアすることができません。																																																													
ビット1	説明																																																													
EDH																																																														
0	E-DMACは正常に動作中です。 (初期値)																																																													
1	NMI端子がアサートされたためE-DMACの動作を停止します。0を書き込むことによってE-DMACは動作を再開します。																																																													
10.3.1 ディスクリプタリストとデータバッファ (1) 送信ディスクリプタ	409	注を差し替え																																																												
ビット 27	410	説明の一部を削除 ビット 26 ~ 0 に示す送信フレームステータスのいずれかのビットがセットされていることを示します。																																																												
(a) 送信ディスクリプタ 0 (TD0) ビット 26 ~ 0	411	TFS26 ~ 5 の説明を差し替え ・ TFS8: 送信アバート検出 【注】送信フレームステータス 4 ~ 0 (TFS4 ~ 0) のいずれかのビットがセットされると本ビットがセットされます。本ビットがセットされた場合は、ビット 27 に示す送信フレームエラー (TFE) がセットされます。 TFS1 の修正 送信中の遅延衝突検出 (EESR の CD ビットに相当)																																																												
(c) 送信ディスクリプタ 2 (TD2)	411	注の差し替え																																																												
(2) 受信ディスクリプタ	411	注の差し替え																																																												
ビット 27	413	説明を修正 ...とを示します。また受信フレームステータスのうちマルチキャストアドレスフレーム受信情報を本ビットに反映するか否かは、送受信ステータスコピー指示レジスタによって設定されます。																																																												

修正項目	ページ	修正内容（詳細はマニュアル参照）				
ビット 26 ~ 0 : 受信フレームステータス 26 ~ 0 (RFS26 ~ RFS0)	413	説明を修正 ・ RFS8 : 受信アポートエラー 【注】受信フレームステータス 9 (RFS9)、7 (RFS7)、4 ~ 0 (RFS4 ~ 0) のいずれかのビットがセットされると本ビットがセットされます。 本ビットがセットされた場合は、ビット 27 に示す受信フレームエラー (RFE) がセットされます。				
(c) 受信ディスクリプタ 2 (RD2)	414	注の差し替え				
10.3.4 マルチバッファフレームの送受信処理について (1) マルチバッファフレームの送信処理	416	タイトルを修正				
10.4 使用上の注意事項	418、 419	新規追加				
11.2.4 DMA チャンネルコントロールレジスタ 0、1 (CHCR0, CHCR1) ビット 4 : トランスファバスマードビット (TB)	429	説明を修正 ...必ず DREQ セレクトビット (DS) を 1 (エッジ検出) に設定してください。				
11.5 使用上の注意	469 ~ 473	(12)、(13)、(14)、(15)、(16) の説明を追加				
13.1.4 レジスタ構成 表 13.2 レジスタ構成	501	表を修正 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>リセットコントロール/ステータスレジスタ</td> <td>RSTCSR</td> <td>R(W)*3</td> <td>H'1E</td> </tr> </table>	リセットコントロール/ステータスレジスタ	RSTCSR	R(W)*3	H'1E
リセットコントロール/ステータスレジスタ	RSTCSR	R(W)*3	H'1E			
13.2.3 リセットコントロール/ステータスレジスタ (RSTCSR)	504	ビット 0 の表を修正  <div style="text-align: center;">           ビット :                    0  <div style="border: 1px solid black; width: 20px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;">—</div>            初期値 :                    0             R/W :                        R         </div> 説明を修正 RSTCSR は、RES 端子からのリセット信号で H'1E に初期化されますが、WDT のオーバーフローによる内部リセット信号では初期化されません。スタンバイモード時、クロックポーズモード時には、H'1E に初期化されます。				
	505	ビット 0 を追加 ・ ビット 4 ~ 1 : 予約ビット 読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。 ・ ビット 0 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。				
13.4.6 スリープモード中のウォッチドッグタイマ (WDT) による内部リセット動作について	512	項を追加				
14.2.8 シリアルステータス 2 レジスタ (SC2SSR)	531	ビット 5、4 の説明追加 ・ ビット 5、4 : クロック・ビットレート比 基本クロックをビットレート比によって選択します。				

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																					
14.2.9 ビットレートレジスタ (SCBRR) 表 14.3 ビットレートに対する SCBRR の設定例 [ 調歩同期式モード ] ( 2 ) ( 3 )	535, 536	表を修正  <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td rowspan="3">ビットレート (bit/s)</td> <td colspan="3">Pφ (MHz)</td> </tr> <tr> <td colspan="3">12</td> </tr> <tr> <td>n</td> <td>N</td> <td>誤差 (%)</td> </tr> <tr> <td>19200</td> <td>0</td> <td>19</td> <td>- 2.34</td> </tr> </table> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td rowspan="3">ビットレート (bit/s)</td> <td colspan="3">Pφ (MHz)</td> </tr> <tr> <td colspan="3">30</td> </tr> <tr> <td>n</td> <td>N</td> <td>誤差 (%)</td> </tr> <tr> <td>4800</td> <td>0</td> <td>194</td> <td>0.16</td> </tr> </table>	ビットレート (bit/s)	Pφ (MHz)			12			n	N	誤差 (%)	19200	0	19	- 2.34	ビットレート (bit/s)	Pφ (MHz)			30			n	N	誤差 (%)	4800	0	194	0.16																																									
ビットレート (bit/s)	Pφ (MHz)																																																																						
	12																																																																						
	n	N	誤差 (%)																																																																				
19200	0	19	- 2.34																																																																				
ビットレート (bit/s)	Pφ (MHz)																																																																						
	30																																																																						
	n	N	誤差 (%)																																																																				
4800	0	194	0.16																																																																				
14.3.2 調歩同期式モード時の動作 ( 1 ) 送信 / 受信フォーマット 表 14.10 シリアル送信 / 受信フォーマット ( 調歩同期式モード )	546	表を修正  <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <th colspan="4">SCSMRの設定</th> <th colspan="12">シリアル送信 / 受信フォーマットとフレーム長</th> </tr> <tr> <th>CHR</th> <th>PE</th> <th>MP</th> <th>STOP</th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> <th>8</th> <th>9</th> <th>10</th> <th>11</th> <th>12</th> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>S</td> <td colspan="7">7ビットデータ</td> <td>STOP</td> <td colspan="5"></td> </tr> <tr> <td></td> <td></td> <td></td> <td>1</td> <td>S</td> <td colspan="7">7ビットデータ</td> <td>STOP</td> <td>STOP</td> <td colspan="5"></td> </tr> </table>	SCSMRの設定				シリアル送信 / 受信フォーマットとフレーム長												CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12	1	0	0	0	S	7ビットデータ							STOP									1	S	7ビットデータ							STOP	STOP					
SCSMRの設定				シリアル送信 / 受信フォーマットとフレーム長																																																																			
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12																																																								
1	0	0	0	S	7ビットデータ							STOP																																																											
			1	S	7ビットデータ							STOP	STOP																																																										
14.3.3 マルチプロセッサ通信機能 ( c ) マルチプロセッサシリアルデータ受信 図 14.14 マルチプロセッサシリアル受信のフローチャートの例 ( 1 )	561	説明を追加 [5]SCIF の状態を確認して受信データを読み出し : シリアルステータス 1 レジスタ ( SC1SSR ) を読み出して、RDF= 1 であることを確認した後、レシーブ FIFO データレジスタ ( SCFRDR ) の受信データを読み出します。																																																																					
14.3.4 クロック同期式モード時の動作	564	説明を修正 クロック同期式モードでは、SCIF は同期クロック立ち上がり同期してデータを受信します。																																																																					
14.5 使用上の注意	582	説明を追加 ( 12 ) SCIF 初期化フローチャートと受信 FIFO データフル割り込み ( RXI ) 要求について																																																																					
17.1.2 H-UDI ブロック図	660	図を修正  																																																																					
17.1.4 レジスタ構成 表 17.2 レジスタ構成	661	表を修正  <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <th>レジスタ</th> <th>略称</th> <th>R/W*1</th> <th>初期値*2</th> </tr> <tr> <td>IDコードレジスタ</td> <td>SDIDR</td> <td>-</td> <td>H'0101000F</td> </tr> </table>	レジスタ	略称	R/W*1	初期値*2	IDコードレジスタ	SDIDR	-	H'0101000F																																																													
レジスタ	略称	R/W*1	初期値*2																																																																				
IDコードレジスタ	SDIDR	-	H'0101000F																																																																				

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																												
17.3.6 ID コードレジスタ (SDIDR)	673	図を修正 <table border="1" style="margin-top: 10px;"> <tr> <td style="text-align: center;">31 28</td> <td style="text-align: center;">27</td> <td style="text-align: center;">12</td> <td style="text-align: center;">11</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">0000</td> <td style="text-align: center;">0001 0000 0001 0000</td> <td></td> <td style="text-align: center;">0000 0000 111</td> <td></td> <td style="text-align: center;">1</td> </tr> <tr> <td>Version (4 ビット)</td> <td>Part Number (16 ビット)</td> <td></td> <td>Manufacture Identify (11 ビット)</td> <td></td> <td>固定コード (1 ビット)</td> </tr> </table>	31 28	27	12	11	1	0	0000	0001 0000 0001 0000		0000 0000 111		1	Version (4 ビット)	Part Number (16 ビット)		Manufacture Identify (11 ビット)		固定コード (1 ビット)																																																										
31 28	27	12	11	1	0																																																																									
0000	0001 0000 0001 0000		0000 0000 111		1																																																																									
Version (4 ビット)	Part Number (16 ビット)		Manufacture Identify (11 ビット)		固定コード (1 ビット)																																																																									
17.6 使用上の注意事項 図 17.7 シリアルデータ 入出力 (2)	681	図を修正 Capture-DR 時																																																																												
21.2 DC 特性 表 21.2 DC 特性	716	表を修正 <table border="1" style="margin-top: 10px;"> <thead> <tr> <th>項目</th> <th>記号</th> <th>min</th> <th>typ</th> <th>max</th> <th>単位</th> <th>測定条件</th> </tr> </thead> <tbody> <tr> <td rowspan="3">出力ハイレベル電圧</td> <td rowspan="3">3.3V/5V兼用端子 その他の出力端子</td> <td rowspan="3"><math>V_{OH}</math></td> <td><math>PV_{CC} - 0.7</math></td> <td>—</td> <td>—</td> <td>V</td> <td><math>I_{OH} = -200</math> A</td> </tr> <tr> <td><math>V_{CC} - 0.5</math></td> <td>—</td> <td>—</td> <td>V</td> <td><math>I_{OH} = -200</math> A</td> </tr> <tr> <td><math>V_{CC} + 1.0</math></td> <td>—</td> <td>—</td> <td>V</td> <td><math>I_{OH} = -1</math>mA</td> </tr> <tr> <td rowspan="2">出力</td> <td rowspan="2">3.3V/5V兼用端子 その他の出力端子</td> <td rowspan="2"><math>V_{OL}</math></td> <td>—</td> <td>—</td> <td>0.6</td> <td>V</td> <td><math>I_{OL} = 1.6</math>mA</td> </tr> <tr> <td>—</td> <td>—</td> <td>0.4</td> <td>V</td> <td><math>I_{OL} = 1.6</math>mA</td> </tr> <tr> <td rowspan="2">端子</td> <td rowspan="2">CAP1、CAP2 その他の全入力端子</td> <td rowspan="2"><math>C_{in}</math></td> <td>—</td> <td>—</td> <td>40</td> <td>pF</td> <td><math>T_a = 25</math> °C</td> </tr> <tr> <td>—</td> <td>—</td> <td>15</td> <td>pF</td> <td></td> </tr> <tr> <td rowspan="4">消費</td> <td rowspan="2">通常動作時</td> <td rowspan="2"><math>I_{CC}</math></td> <td>—</td> <td>—</td> <td>350</td> <td>mA</td> <td>3.6V、CPU 動作クロック = 62.5MHz、DMAC 使用時</td> </tr> <tr> <td>—</td> <td>—</td> <td>300</td> <td>mA</td> <td>3.6V、CPU 動作クロック = 62.5MHz、DMAC 未使用時</td> </tr> <tr> <td>スリープ時</td> <td>—</td> <td>—</td> <td>250</td> <td>mA</td> <td>3.6V、CPU 動作クロック = 62.5MHz、周辺モジュール未使用時</td> </tr> <tr> <td>スタンバイ時</td> <td>—</td> <td>—</td> <td>990</td> <td>A</td> <td></td> </tr> </tbody> </table>	項目	記号	min	typ	max	単位	測定条件	出力ハイレベル電圧	3.3V/5V兼用端子 その他の出力端子	$V_{OH}$	$PV_{CC} - 0.7$	—	—	V	$I_{OH} = -200$ A	$V_{CC} - 0.5$	—	—	V	$I_{OH} = -200$ A	$V_{CC} + 1.0$	—	—	V	$I_{OH} = -1$ mA	出力	3.3V/5V兼用端子 その他の出力端子	$V_{OL}$	—	—	0.6	V	$I_{OL} = 1.6$ mA	—	—	0.4	V	$I_{OL} = 1.6$ mA	端子	CAP1、CAP2 その他の全入力端子	$C_{in}$	—	—	40	pF	$T_a = 25$ °C	—	—	15	pF		消費	通常動作時	$I_{CC}$	—	—	350	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 使用時	—	—	300	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 未使用時	スリープ時	—	—	250	mA	3.6V、CPU 動作クロック = 62.5MHz、周辺モジュール未使用時	スタンバイ時	—	—	990	A	
項目	記号	min	typ	max	単位	測定条件																																																																								
出力ハイレベル電圧	3.3V/5V兼用端子 その他の出力端子	$V_{OH}$	$PV_{CC} - 0.7$	—	—	V	$I_{OH} = -200$ A																																																																							
			$V_{CC} - 0.5$	—	—	V	$I_{OH} = -200$ A																																																																							
			$V_{CC} + 1.0$	—	—	V	$I_{OH} = -1$ mA																																																																							
出力	3.3V/5V兼用端子 その他の出力端子	$V_{OL}$	—	—	0.6	V	$I_{OL} = 1.6$ mA																																																																							
			—	—	0.4	V	$I_{OL} = 1.6$ mA																																																																							
端子	CAP1、CAP2 その他の全入力端子	$C_{in}$	—	—	40	pF	$T_a = 25$ °C																																																																							
			—	—	15	pF																																																																								
消費	通常動作時	$I_{CC}$	—	—	350	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 使用時																																																																							
			—	—	300	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 未使用時																																																																							
	スリープ時	—	—	250	mA	3.6V、CPU 動作クロック = 62.5MHz、周辺モジュール未使用時																																																																								
	スタンバイ時	—	—	990	A																																																																									
21.3.1 クロックタイミング 表 21.5 クロックタイミ ング	719	表を修正 <table border="1" style="margin-top: 10px;"> <thead> <tr> <th>項目</th> <th>記号</th> <th>min</th> <th>max</th> <th>単位</th> <th>参照図</th> </tr> </thead> <tbody> <tr> <td>CKPO クロック出力サイクル時間</td> <td><math>t_{CKPCYC}</math></td> <td>32</td> <td>1000</td> <td>ns</td> <td rowspan="5">21.4</td> </tr> <tr> <td>CKPO クロック出力Lowレベルパルス幅</td> <td><math>t_{CKPOL}</math></td> <td>11</td> <td>—</td> <td>ns</td> </tr> <tr> <td>CKPO クロック出力Highレベルパルス幅</td> <td><math>t_{CKPOH}</math></td> <td>11</td> <td>—</td> <td>ns</td> </tr> <tr> <td>CKPO クロック立ち上がり時間</td> <td><math>t_{CKPOr}</math></td> <td>—</td> <td>5</td> <td>ns</td> </tr> <tr> <td>CKPO クロック立ち下がり時間</td> <td><math>t_{CKPOF}</math></td> <td>—</td> <td>5</td> <td>ns</td> </tr> <tr> <td>パワーオン発振安定時間</td> <td><math>t_{OSC1}</math></td> <td>10</td> <td>—</td> <td>ms</td> <td>21.5</td> </tr> <tr> <td>スタンバイ復帰発振安定時間1</td> <td><math>t_{OSC2}</math></td> <td>10</td> <td>—</td> <td>ms</td> <td>21.6</td> </tr> <tr> <td>スタンバイ復帰発振安定時間2</td> <td><math>t_{OSC3}</math></td> <td>10</td> <td>—</td> <td>ms</td> <td>21.7</td> </tr> <tr> <td>PLL同期安定化時間</td> <td><math>t_{PLL}</math></td> <td>1</td> <td>—</td> <td>ms</td> <td>21.8</td> </tr> </tbody> </table>	項目	記号	min	max	単位	参照図	CKPO クロック出力サイクル時間	$t_{CKPCYC}$	32	1000	ns	21.4	CKPO クロック出力Lowレベルパルス幅	$t_{CKPOL}$	11	—	ns	CKPO クロック出力Highレベルパルス幅	$t_{CKPOH}$	11	—	ns	CKPO クロック立ち上がり時間	$t_{CKPOr}$	—	5	ns	CKPO クロック立ち下がり時間	$t_{CKPOF}$	—	5	ns	パワーオン発振安定時間	$t_{OSC1}$	10	—	ms	21.5	スタンバイ復帰発振安定時間1	$t_{OSC2}$	10	—	ms	21.6	スタンバイ復帰発振安定時間2	$t_{OSC3}$	10	—	ms	21.7	PLL同期安定化時間	$t_{PLL}$	1	—	ms	21.8																				
項目	記号	min	max	単位	参照図																																																																									
CKPO クロック出力サイクル時間	$t_{CKPCYC}$	32	1000	ns	21.4																																																																									
CKPO クロック出力Lowレベルパルス幅	$t_{CKPOL}$	11	—	ns																																																																										
CKPO クロック出力Highレベルパルス幅	$t_{CKPOH}$	11	—	ns																																																																										
CKPO クロック立ち上がり時間	$t_{CKPOr}$	—	5	ns																																																																										
CKPO クロック立ち下がり時間	$t_{CKPOF}$	—	5	ns																																																																										
パワーオン発振安定時間	$t_{OSC1}$	10	—	ms	21.5																																																																									
スタンバイ復帰発振安定時間1	$t_{OSC2}$	10	—	ms	21.6																																																																									
スタンバイ復帰発振安定時間2	$t_{OSC3}$	10	—	ms	21.7																																																																									
PLL同期安定化時間	$t_{PLL}$	1	—	ms	21.8																																																																									
21.3.1 クロックタイミング 図 21.3 CKIO クロック出 力タイミング	720	図を修正 																																																																												

修正項目	ページ	修正内容 (詳細はマニュアル参照)																		
図 21.4 CKPO クロック出力タイミング	721	<p>図の追加</p>																		
21.3.3 バスタイミング 表 21.7 PLL オンバスタイミング [モード 0、4] (3)	727	<p>条件を修正 条件: <math>V_{CC} = PLLV_{CC} = 3.3V \pm 5\%</math>、<math>PV_{CC} = 5.0V \pm 5\%/3.3V \pm 5\%</math>、<math>V_{CC}</math>、<math>PV_{CC}</math> <math>V_{SS} = PV_{SS} = PLLV_{SS} = 0V</math>、<math>T_a = -5 \sim +70</math>、SDRAM バスサイクル時</p>																		
図 21.33 DRAM 読み出しサイクル (TRP=1 サイクル、RCD = 1 サイクル、ノーウェイト)	749	<p>注を修正 1. <math>t_{RDH5}</math> は <b>RD</b>、<b>CASxx</b> の立ち上がりのいずれか早いものから規定</p>																		
図 21.39 EDO リードサイクル (TRP=1 サイクル、RCD = 1 サイクル、ノーウェイト)	755	<p>図を修正</p>																		
21.3.6 シリアルコミュニケーションインタフェース タイミング 表 21.10 シリアルコミュニケーションインタフェース タイミング	765	<p>単位を修正 入力クロックパルス幅 <math>t_{cscyc}</math> <math>t_{scyc}</math></p>																		
21.3.11 イーサネットコントローラ タイミング 表 21.16 イーサネットコントローラ タイミング	774	<p>表を修正</p> <table border="1"> <thead> <tr> <th>項目</th> <th>記号</th> <th>min</th> <th>typ</th> <th>max</th> <th>単位</th> </tr> </thead> <tbody> <tr> <td>TX-CLK サイクル時間</td> <td><math>t_{Toyc}</math></td> <td>40</td> <td>—</td> <td>—</td> <td>ns</td> </tr> <tr> <td>RX-CLK サイクル時間</td> <td><math>t_{Ryc}</math></td> <td>40</td> <td>—</td> <td>—</td> <td>ns</td> </tr> </tbody> </table>	項目	記号	min	typ	max	単位	TX-CLK サイクル時間	$t_{Toyc}$	40	—	—	ns	RX-CLK サイクル時間	$t_{Ryc}$	40	—	—	ns
項目	記号	min	typ	max	単位															
TX-CLK サイクル時間	$t_{Toyc}$	40	—	—	ns															
RX-CLK サイクル時間	$t_{Ryc}$	40	—	—	ns															

修正項目	ページ	修正内容 (詳細はマニュアル参照)																																																																																																																																	
21.3.12 STATS、BH、BUSHIZ 信号タイミング 図 21.78 STATS 出力タイミング	777	<p>図を修正</p>																																																																																																																																	
図 21.79 BH 出力タイミング		<p>図を修正</p>																																																																																																																																	
A.1 アドレス一覧	783	<p>表を修正</p> <table border="1"> <thead> <tr> <th rowspan="2">アドレス</th> <th rowspan="2">レジスタ 略称</th> <th colspan="8">ビット名</th> </tr> <tr> <th>ビット7</th> <th>ビット6</th> <th>ビット5</th> <th>ビット4</th> <th>ビット3</th> <th>ビット2</th> <th>ビット1</th> <th>ビット0</th> </tr> </thead> <tbody> <tr> <td>H'FFFF FD14</td> <td rowspan="3">EESR</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>TABT</td> <td>RABT</td> <td>RFCOF</td> </tr> <tr> <td>H'FFFF FD15</td> <td>—</td> <td>ECI</td> <td>TC</td> <td>TDE</td> <td>TFUF</td> <td>FR</td> <td>RDE</td> <td>RFOF</td> </tr> <tr> <td>H'FFFF FD16</td> <td>—</td> <td>—</td> <td>—</td> <td>ITF</td> <td>CND</td> <td>DLC</td> <td>CD</td> <td>TRO</td> </tr> <tr> <td>H'FFFF FD17</td> <td rowspan="4">TRSCER</td> <td>RMAF</td> <td>—</td> <td>—</td> <td>RRF</td> <td>RTLf</td> <td>RTSF</td> <td>PRE</td> <td>CERF</td> </tr> <tr> <td>H'FFFF FD1C</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD1D</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD1E</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD1F</td> <td rowspan="2">EDOCR</td> <td>RMAFCE</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD30</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD31</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD32</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>H'FFFF FD33</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>FEC</td> <td>AEC</td> <td>EDH</td> <td>—</td> </tr> </tbody> </table>	アドレス	レジスタ 略称	ビット名								ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	H'FFFF FD14	EESR	—	—	—	—	—	TABT	RABT	RFCOF	H'FFFF FD15	—	ECI	TC	TDE	TFUF	FR	RDE	RFOF	H'FFFF FD16	—	—	—	ITF	CND	DLC	CD	TRO	H'FFFF FD17	TRSCER	RMAF	—	—	RRF	RTLf	RTSF	PRE	CERF	H'FFFF FD1C	—	—	—	—	—	—	—	—	H'FFFF FD1D	—	—	—	—	—	—	—	—	H'FFFF FD1E	—	—	—	—	—	—	—	—	H'FFFF FD1F	EDOCR	RMAFCE	—	—	—	—	—	—	—	H'FFFF FD30	—	—	—	—	—	—	—	—	H'FFFF FD31	—	—	—	—	—	—	—	—	H'FFFF FD32	—	—	—	—	—	—	—	—	H'FFFF FD33	—	—	—	—	FEC	AEC	EDH	—
アドレス	レジスタ 略称	ビット名																																																																																																																																	
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0																																																																																																																										
H'FFFF FD14	EESR	—	—	—	—	—	TABT	RABT	RFCOF																																																																																																																										
H'FFFF FD15		—	ECI	TC	TDE	TFUF	FR	RDE	RFOF																																																																																																																										
H'FFFF FD16		—	—	—	ITF	CND	DLC	CD	TRO																																																																																																																										
H'FFFF FD17	TRSCER	RMAF	—	—	RRF	RTLf	RTSF	PRE	CERF																																																																																																																										
H'FFFF FD1C		—	—	—	—	—	—	—	—																																																																																																																										
H'FFFF FD1D		—	—	—	—	—	—	—	—																																																																																																																										
H'FFFF FD1E		—	—	—	—	—	—	—	—																																																																																																																										
H'FFFF FD1F	EDOCR	RMAFCE	—	—	—	—	—	—	—																																																																																																																										
H'FFFF FD30		—	—	—	—	—	—	—	—																																																																																																																										
H'FFFF FD31	—	—	—	—	—	—	—	—																																																																																																																											
H'FFFF FD32	—	—	—	—	—	—	—	—																																																																																																																											
H'FFFF FD33	—	—	—	—	FEC	AEC	EDH	—																																																																																																																											
	787	<p>表を修正</p> <table border="1"> <thead> <tr> <th>アドレス</th> <th>レジスタ 略称</th> </tr> </thead> <tbody> <tr> <td>H'FFFF FE71</td> <td>DRCR0</td> </tr> <tr> <td>H'FFFF FE72</td> <td>DRCR1</td> </tr> </tbody> </table>	アドレス	レジスタ 略称	H'FFFF FE71	DRCR0	H'FFFF FE72	DRCR1																																																																																																																											
アドレス	レジスタ 略称																																																																																																																																		
H'FFFF FE71	DRCR0																																																																																																																																		
H'FFFF FE72	DRCR1																																																																																																																																		
C . 型名一覧変更 表 C.1 SH7615 型名一覧	796	<p>表を修正</p> <table border="1"> <thead> <tr> <th>略称</th> <th>電圧</th> <th>動作周波数</th> <th>製品型名</th> <th>パッケージ</th> </tr> </thead> <tbody> <tr> <td rowspan="4">SH7615</td> <td rowspan="4">V<sub>CC</sub> = PLLV<sub>CC</sub> = 3.3V PV<sub>CC</sub> = 5.0V/3.3V</td> <td rowspan="4">62.5MHz</td> <td>HD6417615ARF</td> <td>FP-208C</td> </tr> <tr> <td>HD6417615ARFV</td> <td>FP-208CV</td> </tr> <tr> <td>HD6417615ARBP</td> <td>BP-240A</td> </tr> <tr> <td>HD6417615ARBPV</td> <td>BP-240AV</td> </tr> </tbody> </table>	略称	電圧	動作周波数	製品型名	パッケージ	SH7615	V <sub>CC</sub> = PLLV <sub>CC</sub> = 3.3V PV <sub>CC</sub> = 5.0V/3.3V	62.5MHz	HD6417615ARF	FP-208C	HD6417615ARFV	FP-208CV	HD6417615ARBP	BP-240A	HD6417615ARBPV	BP-240AV																																																																																																																	
略称	電圧	動作周波数	製品型名	パッケージ																																																																																																																															
SH7615	V <sub>CC</sub> = PLLV <sub>CC</sub> = 3.3V PV <sub>CC</sub> = 5.0V/3.3V	62.5MHz	HD6417615ARF	FP-208C																																																																																																																															
			HD6417615ARFV	FP-208CV																																																																																																																															
			HD6417615ARBP	BP-240A																																																																																																																															
			HD6417615ARBPV	BP-240AV																																																																																																																															
D . 外形寸法 図 D.1 外形寸法図 (FP-208C、FP-208CV)	797	<p>説明を修正 SH7615 の外形寸法を図 D.1、<b>図 D.2</b> に示します。 図 D.1 変更</p>																																																																																																																																	
図 D.2 外形寸法図 (BP-240A、BP-240AV)	798	<p>図 D.2 を追加</p>																																																																																																																																	



---

# 目次

---

1. 概要	1
1.1 イーサネットコントローラ搭載 SuperH マイコンの特長	1
1.2 ブロック図	9
1.3 端子の説明	10
1.3.1 ピン配置	10
1.3.2 端子の機能	12
1.3.3 端子マルチプレクスの説明	18
1.4 処理状態	24
2. CPU	29
2.1 レジスタ構成	29
2.1.1 汎用レジスタ	29
2.1.2 コントロールレジスタ	31
2.1.3 システムレジスタ	34
2.1.4 DSP レジスタ	35
2.1.5 ガードビットとオーバフローの扱いに関する注意事項	37
2.1.6 レジスタの初期値	37
2.2 データ形式	38
2.2.1 レジスタのデータ形式	38
2.2.2 メモリ上でのデータ形式	38
2.2.3 イミディエイトデータのデータ形式	39
2.2.4 DSP タイプデータ形式	39
2.2.5 DSP タイプ命令とデータ形式	41
2.3 CPU コア命令の特長	45
2.4 命令形式	48
2.4.1 CPU 命令のアドレッシングモード	48
2.4.2 DSP データアドレッシング	50
2.4.3 CPU 命令の命令形式	56
2.4.4 DSP 命令の命令形式	59
2.5 命令セット	63
2.5.1 CPU 命令の命令セット	63
2.5.2 DSP データ転送命令の命令セット	73
2.5.3 DSP 演算命令の命令セット	76
2.6 使用上の注意事項	85

3.	発振回路と動作モード .....	89
3.1	概要 .....	89
3.2	内蔵発振回路と動作モード .....	90
3.2.1	発振回路の構成 .....	90
3.2.2	クロック動作モードの設定 .....	92
3.2.3	水晶発振子の接続方法 .....	93
3.2.4	外部クロックの入力方法 .....	94
3.2.5	レジスタによる動作周波数の選択 .....	95
3.2.6	クロックモードと周波数範囲 .....	102
3.2.7	ボード設計上の注意事項 .....	103
3.3	CS0 空間のバス幅 .....	105
4.	例外処理 .....	107
4.1	概要 .....	107
4.1.1	例外処理の種類と優先順位 .....	107
4.1.2	例外処理の動作 .....	108
4.1.3	例外処理ベクタテーブル .....	109
4.2	リセット .....	112
4.2.1	リセットの種類 .....	112
4.2.2	パワーオンリセット .....	112
4.2.3	マニュアルリセット .....	112
4.3	アドレスエラー .....	114
4.3.1	アドレスエラー発生要因 .....	114
4.3.2	アドレスエラー例外処理 .....	114
4.4	割り込み .....	116
4.4.1	割り込み要因 .....	116
4.4.2	割り込み優先順位 .....	117
4.4.3	割り込み例外処理 .....	117
4.5	命令による例外 .....	118
4.5.1	命令による例外の種類 .....	118
4.5.2	トラップ命令 .....	118
4.5.3	スロット不当命令 .....	118
4.5.4	一般不当命令 .....	119
4.6	例外要因が受け付けられない場合 .....	120
4.6.1	遅延分岐命令の直後 .....	120
4.6.2	割り込み禁止命令の直後 .....	120
4.6.3	繰り返ループにおける命令 .....	121
4.7	例外処理終了後のスタックの状態 .....	122
4.8	使用上の注意 .....	123
4.8.1	スタックポインタ (SP) の値 .....	123
4.8.2	ベクタベースレジスタ (VBR) の値 .....	123

4.8.3	アドレスエラー例外処理のスタッキングで発生するアドレスエラー .....	123
4.8.4	マニュアルリセット時のレジスタアクセス .....	123
<b>5.</b>	<b>割り込みコントローラ (INTC).....</b>	<b>125</b>
5.1	概要 .....	125
5.1.1	特長 .....	125
5.1.2	ブロック図 .....	126
5.1.3	端子構成 .....	127
5.1.4	レジスタ構成 .....	127
5.2	割り込み要因 .....	129
5.2.1	NMI 割り込み .....	129
5.2.2	ユーザブレーク割り込み .....	129
5.2.3	H-UDI 割り込み .....	129
5.2.4	IRL 割り込み .....	129
5.2.5	IRQ 割り込み .....	130
5.2.6	内蔵周辺モジュール割り込み .....	134
5.2.7	割り込み例外処理ベクタと優先順位 .....	135
5.3	レジスタの説明 .....	141
5.3.1	割り込み優先レベル設定レジスタ A (IPRA) .....	141
5.3.2	割り込み優先レベル設定レジスタ B (IPRB) .....	142
5.3.3	割り込み優先レベル設定レジスタ C (IPRC) .....	142
5.3.4	割り込み優先レベル設定レジスタ D (IPRD) .....	143
5.3.5	割り込み優先レベル設定レジスタ E (IPRE) .....	143
5.3.6	ベクタ番号設定レジスタ WDT (VCRWDT) .....	144
5.3.7	ベクタ番号設定レジスタ A (VCRA) .....	145
5.3.8	ベクタ番号設定レジスタ B (VCRB) (予約レジスタ) .....	145
5.3.9	ベクタ番号設定レジスタ C (VCRC) .....	146
5.3.10	ベクタ番号設定レジスタ D (VCRD) .....	146
5.3.11	ベクタ番号設定レジスタ E (VCRE) .....	147
5.3.12	ベクタ番号設定レジスタ F (VCRF) .....	147
5.3.13	ベクタ番号設定レジスタ G (VCRG) .....	148
5.3.14	ベクタ番号設定レジスタ H (VCRH) .....	149
5.3.15	ベクタ番号設定レジスタ I (VCRI) .....	149
5.3.16	ベクタ番号設定レジスタ J (VCRJ) .....	150
5.3.17	ベクタ番号設定レジスタ K (VCRK) .....	151
5.3.18	ベクタ番号設定レジスタ L (VCRL) .....	151
5.3.19	ベクタ番号設定レジスタ M (VCRM) .....	152
5.3.20	ベクタ番号設定レジスタ N (VCRN) .....	153
5.3.21	ベクタ番号設定レジスタ O (VCRO) .....	153
5.3.22	ベクタ番号設定レジスタ P (VCRP) .....	154
5.3.23	ベクタ番号設定レジスタ Q (VCRQ) .....	155

5.3.24	ベクタ番号設定レジスタ R (VCRR).....	155
5.3.25	ベクタ番号設定レジスタ S (VCRS).....	156
5.3.26	ベクタ番号設定レジスタ T (VCRT).....	156
5.3.27	ベクタ番号設定レジスタ U (VCRU).....	158
5.3.28	割り込みコントロールレジスタ (ICR).....	160
5.3.29	IRQ コントロール/ステータスレジスタ (IRQCSR).....	161
5.4	動作説明.....	163
5.4.1	割り込み動作の流れ.....	163
5.4.2	割り込み例外処理終了後のスタック状態.....	165
5.5	割り込み応答時間.....	166
5.6	IRL3 ~ IRL0 端子のサンプリング.....	167
5.7	使用上の注意.....	168
5.7.1	モジュールスタンバイ実行に関する注意事項.....	168
5.7.2	割り込み要因のクリアに関する注意事項.....	168
<b>6.</b>	<b>ユーザブ레이크コントローラ (UBC).....</b>	<b>173</b>
6.1	概要.....	173
6.1.1	特長.....	173
6.1.2	ブロック図.....	175
6.1.3	レジスタ構成.....	176
6.2	各レジスタの説明.....	177
6.2.1	ブ레이크アドレスレジスタ A (BARA).....	177
6.2.2	ブ레이크アドレスマスクレジスタ A (BAMRA).....	177
6.2.3	ブ레이크バスサイクルレジスタ A (BBRA).....	178
6.2.4	ブ레이크アドレスレジスタ B (BARB).....	180
6.2.5	ブ레이크アドレスマスクレジスタ B (BAMRB).....	180
6.2.6	ブ레이크バスサイクルレジスタ B (BBRB).....	181
6.2.7	ブ레이크アドレスレジスタ C (BARC).....	183
6.2.8	ブ레이크アドレスマスクレジスタ C (BAMRC).....	184
6.2.9	ブ레이크データレジスタ C (BDRC).....	185
6.2.10	ブ레이크データマスクレジスタ C (BDMRC).....	185
6.2.11	ブ레이크バスサイクルレジスタ C (BBRC).....	186
6.2.12	実行回数ブ레이크レジスタ C (BETRC).....	187
6.2.13	ブ레이크アドレスレジスタ D (BARD).....	188
6.2.14	ブ레이크アドレスマスクレジスタ D (BAMRD).....	189
6.2.15	ブ레이크データレジスタ D (BDRD).....	190
6.2.16	ブ레이크データマスクレジスタ D (BDMRD).....	190
6.2.17	ブ레이크バスサイクルレジスタ D (BBRD).....	191
6.2.18	実行回数ブ레이크レジスタ D (BETRD).....	192
6.2.19	ブ레이크コントロールレジスタ (BRCR).....	193
6.2.20	ブランチフラグレジスタ (BRFR).....	198

6.2.21	ブランチソースレジスタ (BRSR).....	199
6.2.22	ブランチディスティネーションレジスタ (BRDR).....	200
6.3	動作説明.....	201
6.3.1	ユーザブ레이크動作の流れ.....	201
6.3.2	命令フェッチサイクルのブ레이크.....	202
6.3.3	データアクセスサイクルによるブ레이크.....	202
6.3.4	退避するプログラムカウンタ (PC) の値.....	203
6.3.5	X メモリまたは Y メモリバスサイクルでのブ레이크.....	204
6.3.6	シーケンシャルブ레이크.....	204
6.3.7	PC トレース.....	206
6.3.8	使用例.....	207
6.3.9	注意事項.....	212
<b>7.</b>	<b>バスステートコントローラ (BSC).....</b>	<b>215</b>
7.1	概要.....	215
7.1.1	特長.....	215
7.1.2	ブロック図.....	217
7.1.3	端子構成.....	218
7.1.4	レジスタ構成.....	219
7.1.5	アドレスマップ.....	220
7.2	レジスタの説明.....	222
7.2.1	バスコントロールレジスタ 1 (BCR1).....	222
7.2.2	バスコントロールレジスタ 2 (BCR2).....	225
7.2.3	バスコントロールレジスタ 3 (BCR3).....	226
7.2.4	ウェイトコントロールレジスタ 1 (WCR1).....	227
7.2.5	ウェイトコントロールレジスタ 2 (WCR2).....	229
7.2.6	ウェイトコントロールレジスタ 3 (WCR3).....	231
7.2.7	個別メモリコントロールレジスタ (MCR).....	232
7.2.8	リフレッシュタイムコントロール/ステータスレジスタ (RTCSR).....	236
7.2.9	リフレッシュタイムカウンタ (RTCNT).....	238
7.2.10	リフレッシュタイムコンスタントレジスタ (RTCOR).....	239
7.3	アクセスサイズとデータアライメント.....	240
7.3.1	通常デバイスとの接続.....	240
7.3.2	リトルエンディアンのデバイスとの接続.....	242
7.4	通常空間アクセス.....	244
7.4.1	基本タイミング.....	244
7.4.2	ウェイトステート制御.....	249
7.4.3	$\overline{CS}$ アサート期間拡張.....	253
7.5	シンクロナス DRAM インタフェース.....	254
7.5.1	シンクロナス DRAM 直結方式.....	254
7.5.2	アドレスマルチプレクス.....	256

7.5.3	バーストリード	257
7.5.4	シングルリード	261
7.5.5	シングルライトモード	263
7.5.6	バーストライトモード	263
7.5.7	バンクアクティブ	265
7.5.8	リフレッシュ	272
7.5.9	オートプリチャージサイクル (Tap) と次アクセスのオーバラップについて	274
7.5.10	パワーオンシーケンス	275
7.5.11	64M ビットシンクロナス DRAM (2M ワード × 32 ビット品) との接続方法	278
7.6	DRAM インタフェース	282
7.6.1	DRAM 直結方式	282
7.6.2	アドレスマルチプレクス	283
7.6.3	基本タイミング	283
7.6.4	ウェイトステート制御	284
7.6.5	バーストアクセス	286
7.6.6	EDO モード	289
7.6.7	DMA シングル転送	293
7.6.8	リフレッシュ	294
7.6.9	パワーオンシーケンス	296
7.7	バースト ROM インタフェース	297
7.8	アクセスサイクル間ウェイト	299
7.9	バスアービトレーション	301
7.9.1	マスタモード	305
7.10	その他	307
7.10.1	リセット	307
7.10.2	CPU、DMAC および E-DMAC からみたアクセス	307
7.10.3	STATS1、0 端子について	309
7.10.4	$\overline{\text{BUSHIZ}}$ の仕様について	309
7.11	使用上の注意	311
7.11.1	DMAC 使用時のシンクロナス DRAM ライト直後の通常空間アクセス時の注意事項	311
7.11.2	クロック比 I : E = 1 : 1、バス幅 8 ビット、外部ウェイト入力時の注意事項	312
7.11.3	シンクロナス DRAM へのデータ誤出力防止について	313

## 8. キャッシュ .....315

8.1	概要	315
8.1.1	特長	315
8.1.2	レジスタ構成	316
8.2	レジスタの説明	317
8.2.1	キャッシュコントロールレジスタ (CCR)	317
8.3	アドレス空間とキャッシュ	319
8.4	キャッシュ動作	319

8.4.1	キャッシュ読み出し .....	319
8.4.2	書き込み .....	321
8.4.3	キャッシュスルーアクセス .....	324
8.4.4	TAS 命令 .....	325
8.4.5	擬似 LRU とキャッシュ置き換え .....	325
8.4.6	キャッシュの初期化 .....	326
8.4.7	連想ページ .....	327
8.4.8	キャッシュフラッシュ .....	327
8.4.9	データレイアクセス .....	327
8.4.10	アドレスレイアクセス .....	328
8.5	使用方法 .....	329
8.5.1	初期化 .....	329
8.5.2	特定アドレスのページ .....	330
8.5.3	キャッシュデータのコピーレンシ .....	330
8.5.4	2ウェイキャッシュモード .....	331
8.6	使用上の注意事項 .....	333
8.6.1	スタンバイ .....	333
8.6.2	キャッシュコントロールレジスタ .....	333
<b>9.</b>	<b>イーサネットコントローラ ( EtherC ) .....</b>	<b>335</b>
9.1	概要 .....	335
9.1.1	特長 .....	335
9.1.2	構成 .....	336
9.1.3	端子構成 .....	337
9.1.4	イーサネットコントローラレジスタ構成 .....	338
9.2	レジスタの説明 .....	339
9.2.1	EtherC モードレジスタ ( ECMR ) .....	339
9.2.2	EtherC ステータスレジスタ ( ECSR ) .....	342
9.2.3	EtherC 割り込み許可レジスタ ( ECSIPR ) .....	344
9.2.4	PHY 部インタフェースレジスタ ( PIR ) .....	345
9.2.5	MAC アドレス上位設定レジスタ ( MAHR ) .....	347
9.2.6	MAC アドレス下位設定レジスタ ( MALR ) .....	348
9.2.7	受信フレーム長上限レジスタ ( RFLR ) .....	349
9.2.8	PHY 部ステータスレジスタ ( PSR ) .....	350
9.2.9	送信リトライオーバーカウンタレジスタ ( TROCR ) .....	351
9.2.10	衝突検出カウンタレジスタ ( CDCR ) .....	352
9.2.11	キャリア消失カウンタレジスタ ( LCCR ) .....	353
9.2.12	キャリア未検出カウンタレジスタ ( CNDCR ) .....	354
9.2.13	フレーム長異常カウンタレジスタ ( IFLCR ) .....	355
9.2.14	CRC エラーフレーム受信カウンタレジスタ ( CEFCR ) .....	356
9.2.15	フレーム受信エラーカウンタレジスタ ( FRECR ) .....	357

9.2.16	64 バイト未満フレーム受信カウンタレジスタ (TSFRCCR).....	358
9.2.17	指定バイト超フレーム受信カウンタレジスタ (TLFRCCR).....	359
9.2.18	端数ビットフレーム受信カウンタレジスタ (RFCR).....	360
9.2.19	マルチキャストアドレスフレーム受信カウンタレジスタ (MAFCR).....	361
9.3	動作説明.....	362
9.3.1	送信動作.....	362
9.3.2	受信動作.....	364
9.3.3	MII フレームタイミング.....	365
9.3.4	MII レジスタのアクセス方法.....	367
9.3.5	Magic Packet の検出.....	370
9.3.6	CPU 動作モードとイーサネットコントローラの動作.....	370
9.4	PHY-LSI との接続.....	372
9.5	使用上の注意.....	373
<b>10.</b>	<b>イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC).....</b>	<b>375</b>
10.1	概要.....	375
10.1.1	特長.....	375
10.1.2	構成.....	375
10.1.3	ディスクリプタ管理方式.....	376
10.1.4	レジスタ構成.....	377
10.2	レジスタの説明.....	378
10.2.1	E-DMAC モードレジスタ (EDMR).....	378
10.2.2	E-DMAC 送信要求レジスタ (EDTRR).....	379
10.2.3	E-DMAC 受信要求レジスタ (EDRRR).....	381
10.2.4	送信ディスクリプタリスト先頭アドレスレジスタ (TDLAR).....	382
10.2.5	受信ディスクリプタリスト先頭アドレスレジスタ (RDLAR).....	383
10.2.6	EtherC/E-DMAC ステータスレジスタ (EESR).....	384
10.2.7	EtherC/E-DMAC ステータス割込み許可レジスタ (EESIPR).....	391
10.2.8	送受信ステータスコピー指示レジスタ (TRSCER).....	396
10.2.9	ミスドフレームカウンタレジスタ (RMFCR).....	397
10.2.10	送信 FIFO しきい値指定レジスタ (TFTR).....	398
10.2.11	FIFO 容量指定レジスタ (FDR).....	401
10.2.12	受信方式制御レジスタ (RCR).....	402
10.2.13	E-DMAC 動作制御レジスタ (EDOCR).....	403
10.2.14	受信バッファライトアドレスレジスタ (RBWAR).....	405
10.2.15	受信ディスクリプタフェッチアドレスレジスタ (RDFAR).....	406
10.2.16	送信バッファリードアドレスレジスタ (TBRAR).....	407
10.2.17	送信ディスクリプタフェッチアドレスレジスタ (TDFAR).....	408
10.3	動作説明.....	409
10.3.1	ディスクリプタリストとデータバッファ.....	409

10.3.2	送信機能 .....	414
10.3.3	受信機能 .....	415
10.3.4	マルチバッファフレームの送受信処理について .....	416
10.4	使用上の注意事項 .....	418
10.4.1	E-DMAC 送信要求レジスタ (EDTRR) および E-DMAC 受信要求レジスタ (EDRRR) 使用上の注意事項 .....	418
<b>11.</b>	<b>ダイレクトメモリアクセスコントローラ (DMAC).....</b>	<b>421</b>
11.1	概要 .....	421
11.1.1	特長 .....	421
11.1.2	ブロック図 .....	423
11.1.3	端子構成 .....	424
11.1.4	レジスタ構成 .....	424
11.2	レジスタの説明 .....	425
11.2.1	DMA ソースアドレスレジスタ 0、1 (SAR0、SAR1).....	425
11.2.2	DMA デスティネーションアドレスレジスタ 0、1 (DAR0、DAR1).....	425
11.2.3	DMA トランスファカウンタレジスタ 0、1 (TCR0、TCR1).....	426
11.2.4	DMA チャンネルコントロールレジスタ 0、1 (CHCR0、CHCR1).....	426
11.2.5	ベクタ番号設定レジスタ DMA0、1 (VCRDMA0、VCRDMA1).....	431
11.2.6	DMA 要求 / 応答選択制御レジスタ 0、1 (DRCR0、DRCR1).....	432
11.2.7	DMA オペレーションレジスタ (DMAOR).....	434
11.3	動作説明 .....	436
11.3.1	DMA 転送フロー .....	436
11.3.2	DMA 転送要求 .....	438
11.3.3	チャンネルの優先順位 .....	442
11.3.4	DMA 転送の種類 .....	444
11.3.5	バスサイクル数 .....	452
11.3.6	DMA 転送要求受け付け信号出力タイミング .....	452
11.3.7	DREQn 端子の入力検出タイミング .....	461
11.3.8	DMA 転送終了 .....	465
11.3.9	$\overline{\text{BH}}$ 端子の出力タイミング .....	466
11.4	使用例 .....	467
11.4.1	内蔵 SCIF と外部メモリとの DMA 転送例 .....	467
11.5	使用上の注意 .....	468
<b>12.</b>	<b>16 ビットフリーランニングタイマ (FRT).....</b>	<b>475</b>
12.1	概要 .....	475
12.1.1	特長 .....	475
12.1.2	ブロック図 .....	476
12.1.3	端子構成 .....	477
12.1.4	レジスタ構成 .....	477

12.2	各レジスタの説明	478
12.2.1	フリーランニングカウンタ (FRC)	478
12.2.2	アウトプットコンペアレジスタ A、B (OCRA、B)	478
12.2.3	インプットキャプチャレジスタ (FICR)	479
12.2.4	タイムインタラプトイネーブルレジスタ (TIER)	479
12.2.5	フリーランニングタイムコントロール / ステータレジスタ (FTCSR)	481
12.2.6	タイマコントロールレジスタ (TCR)	482
12.2.7	タイマアウトプットコンペアコントロールレジスタ (TOCR)	483
12.3	CPU とのインタフェース	485
12.4	動作説明	488
12.4.1	FRC のカウントタイミング	488
12.4.2	アウトプットコンペア出力タイミング	489
12.4.3	FRC クリアタイミング	489
12.4.4	インプットキャプチャ入力タイミング	489
12.4.5	インプットキャプチャフラグ (ICF) のセットタイミング	491
12.4.6	アウトプットコンペアフラグ (OCFA、B) のセットタイミング	491
12.4.7	タイマオーバフローフラグ (OVF) のセットタイミング	492
12.5	割り込み要因	492
12.6	FRT の使用例	493
12.7	使用上の注意	494
12.7.1	FRC のライトとクリアの競合	494
12.7.2	FRC のライトとカウントアップの競合	495
12.7.3	OCR のライトとコンペアマッチの競合	496
12.7.4	内部クロックの切り替えとカウンタの動作	497
12.7.5	タイマ出力 (FTOA、FTOB)	498
<b>13.</b>	<b>ウォッチドッグタイマ (WDT)</b>	<b>499</b>
13.1	概要	499
13.1.1	特長	499
13.1.2	ブロック図	500
13.1.3	端子構成	500
13.1.4	レジスタ構成	501
13.2	レジスタの説明	502
13.2.1	ウォッチドッグタイマカウンタ (WTCNT)	502
13.2.2	ウォッチドッグタイマコントロール / ステータスレジスタ (WTCSR)	502
13.2.3	リセットコントロール / ステータスレジスタ (RSTCSR)	504
13.2.4	レジスタアクセス時の注意	506
13.3	動作説明	508
13.3.1	ウォッチドッグタイマモード時の動作	508
13.3.2	インターバルタイマモード時の動作	509
13.3.3	スタンバイモード解除時の動作	509

13.3.4	オーバフローフラグ (OVF) のセットタイミング .....	510
13.3.5	ウォッチドッグタイマオーバフローフラグ (WOVF) のセットタイミング .....	510
13.4	使用上の注意 .....	511
13.4.1	ウォッチドッグタイマカウンタ (WTCNT) の書き込みとカウントアップの競合 .....	511
13.4.2	CKS2 ~ CKS0 ビットの書き換え .....	511
13.4.3	ウォッチドッグタイマモードとインターバルタイマモードの切り替え .....	511
13.4.4	WDTOVF 信号によるシステムのリセット .....	511
13.4.5	ウォッチドッグタイマモードでの内部リセット .....	512
13.4.6	スリープモード中のウォッチドッグタイマ (WDT) による内部リセット動作について .....	512

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF).....513

14.1	概要 .....	513
14.1.1	特長 .....	513
14.1.2	ブロック図 .....	515
14.1.3	端子構成 .....	516
14.1.4	レジスタ構成 .....	517
14.2	レジスタの説明 .....	518
14.2.1	レシーブシフトレジスタ (SCRSR) .....	518
14.2.2	レシーブ FIFO データレジスタ (SCFRDR) .....	518
14.2.3	トランスミットシフトレジスタ (SCTSR) .....	519
14.2.4	トランスミット FIFO データレジスタ (SCFTDR) .....	519
14.2.5	シリアルモードレジスタ (SCSMR) .....	520
14.2.6	シリアルコントロールレジスタ (SCSCR) .....	523
14.2.7	シリアルステータス 1 レジスタ (SC1SSR) .....	526
14.2.8	シリアルステータス 2 レジスタ (SC2SSR) .....	530
14.2.9	ビットレートレジスタ (SCBRR) .....	532
14.2.10	FIFO コントロールレジスタ (SCFCR) .....	538
14.2.11	FIFO データ数レジスタ (SCFDR) .....	540
14.2.12	FIFO エラーレジスタ (SCFER) .....	541
14.2.13	IrDA モードレジスタ (SCIMR) .....	541
14.3	動作説明 .....	543
14.3.1	概要 .....	543
14.3.2	調歩同期式モード時の動作 .....	545
14.3.3	マルチプロセッサ通信機能 .....	556
14.3.4	クロック同期式モード時の動作 .....	564
14.3.5	送信 / 受信 FIFO バッファの使用法 .....	573
14.3.6	IrDA モード時の動作 .....	575
14.4	SCIF の割り込み要因と DMAC .....	578
14.5	使用上の注意 .....	578

<b>15. シリアル I/O ( SIO )</b> .....	<b>585</b>
15.1 概要.....	585
15.1.1 特長.....	585
15.2 レジスタ構成.....	588
15.2.1 レシーブシフトレジスタ ( SIRSR ).....	588
15.2.2 レシーブデータレジスタ ( SIRDR ).....	589
15.2.3 トランスミットシフトレジスタ ( SITSR ).....	589
15.2.4 トランスミットデータレジスタ ( SITDR ).....	590
15.2.5 シリアルコントロールレジスタ ( SICTR ).....	590
15.2.6 シリアルステータスレジスタ ( SISTR ).....	592
15.3 動作.....	594
15.3.1 入力.....	594
15.3.2 出力.....	595
15.4 SIO の割り込み要因と DMAC.....	597
<b>16. 16 ビットタイマパルスユニット ( TPU )</b> .....	<b>599</b>
16.1 概要.....	599
16.1.1 特長.....	599
16.1.2 ブロック図.....	602
16.1.3 端子構成.....	603
16.1.4 レジスタ構成.....	604
16.2 各レジスタの説明.....	605
16.2.1 タイマコントロールレジスタ ( TCR ).....	605
16.2.2 タイマモードレジスタ ( TMDR ).....	608
16.2.3 タイマ I/O コントロールレジスタ ( TIOR ).....	610
16.2.4 タイマインタラプトイネーブルレジスタ ( TIER ).....	615
16.2.5 タイマステータスレジスタ ( TSR ).....	617
16.2.6 タイマカウンタ ( TCNT ).....	620
16.2.7 タイマジェネラルレジスタ ( TGR ).....	620
16.2.8 タイマスタートレジスタ ( TSTR ).....	620
16.2.9 タイマシンクロレジスタ ( TSYR ).....	621
16.3 バスマスタとのインタフェース.....	622
16.3.1 16 ビットレジスタ.....	622
16.3.2 8 ビットレジスタ.....	622
16.4 動作説明.....	624
16.4.1 概要.....	624
16.4.2 基本機能.....	624
16.4.3 同期動作.....	629
16.4.4 バッファ動作.....	630
16.4.5 PWM モード.....	633
16.4.6 位相計数モード.....	637

16.5	割り込み	642
16.5.1	割り込み要因と優先順位	642
16.5.2	DMAC の起動	643
16.6	動作タイミング	644
16.6.1	入出力タイミング	644
16.6.2	割り込み信号タイミング	647
16.7	使用上の注意	651
16.7.1	入力クロックの制限事項	651
16.7.2	周期設定上の注意事項	651
16.7.3	TCNT のライトとクリアの競合	652
16.7.4	TCNT のライトとカウントアップの競合	652
16.7.5	TGR のライトとコンペアマッチの競合	653
16.7.6	パツファレジスタのライトとコンペアマッチの競合	653
16.7.7	TGR のリードとインプットキャプチャの競合	654
16.7.8	TGR のライトとインプットキャプチャの競合	654
16.7.9	パツファレジスタのライトとインプットキャプチャの競合	655
16.7.10	オーバフロー / アンダフローとカウンタクリアの競合	655
16.7.11	TCNT のライトとオーバフロー / アンダフローの競合	656
16.7.12	入出力端子の兼用	656
16.7.13	モジュールストップ時の割り込み	656
16.8	使用上の注意事項	657
16.8.1	TSR0 ~ 2 のフラグクリア	657
16.8.2	TPU0 による DMA 転送	657
<b>17.</b>	<b>ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)</b>	<b>659</b>
17.1	概要	659
17.1.1	特長	659
17.1.2	H-UDI ブロック図	660
17.1.3	端子構成	661
17.1.4	レジスタ構成	661
17.2	外部信号	662
17.2.1	テストクロック (TCK)	662
17.2.2	テストモードセレクト (TMS)	662
17.2.3	テストデータ入力 (TDI)	662
17.2.4	テストデータ出力 (TDO)	662
17.2.5	テストリセット ( $\overline{\text{TRST}}$ )	662
17.3	レジスタ	663
17.3.1	インストラクションレジスタ (SDIR)	663
17.3.2	ステータスレジスタ (SDSR)	664
17.3.3	データレジスタ (SDDR)	664

17.3.4	バイパスレジスタ (SDBPR).....	665
17.3.5	バウンダリスキャンレジスタ (SDBSR).....	665
17.3.6	ID コードレジスタ (SDIDR).....	673
17.4	動作.....	674
17.4.1	TAP コントローラ.....	674
17.4.2	H-UDI 割り込みとシリアル転送.....	674
17.4.3	H-UDI のリセット.....	677
17.5	バウンダリスキャン.....	678
17.5.1	サポートする命令.....	678
17.5.2	注意事項.....	679
17.6	使用上の注意事項.....	680
<b>18.</b>	<b>ピンファンクションコントローラ (PFC).....</b>	<b>683</b>
18.1	概要.....	683
18.2	レジスタ構成.....	684
18.3	レジスタの説明.....	685
18.3.1	ポート A コントロールレジスタ (PACR).....	685
18.3.2	ポート A・IO レジスタ (PAIOR).....	689
18.3.3	ポート B コントロールレジスタ (PBCR、PBCR2).....	689
18.3.4	ポート B・IO レジスタ (PBIOR).....	694
<b>19.</b>	<b>I/O ポート.....</b>	<b>695</b>
19.1	概要.....	695
19.2	ポート A.....	696
19.2.1	レジスタ構成.....	696
19.2.2	ポート A データレジスタ (PADR).....	696
19.3	ポート B.....	698
19.3.1	レジスタ構成.....	698
19.3.2	ポート B データレジスタ (PBDR).....	698
<b>20.</b>	<b>低消費電力状態.....</b>	<b>701</b>
20.1	概要.....	701
20.1.1	低消費電力状態の種類.....	701
20.1.2	レジスタ構成.....	702
20.2	レジスタの説明.....	703
20.2.1	スタンバイコントロールレジスタ 1 (SBYCR1).....	703
20.2.2	スタンバイコントロールレジスタ 2 (SBYCR2).....	705
20.3	スリープモード.....	707
20.3.1	スリープモードへの遷移.....	707
20.3.2	スリープモードの解除.....	707
20.4	スタンバイモード.....	708

20.4.1	スタンバイモードへの遷移 .....	708
20.4.2	スタンバイモードの解除 .....	710
20.4.3	NMI 割り込みによるスタンバイモード解除方法 .....	711
20.4.4	クロックポーズ機能 .....	711
20.4.5	スタンバイモードの注意事項 .....	713
20.5	モジュールスタンバイ機能 .....	714
20.5.1	モジュールスタンバイ状態への遷移 .....	714
20.5.2	モジュールスタンバイ状態の解除 .....	714
<b>21.</b>	<b>電気的特性 .....</b>	<b>715</b>
21.1	絶対最大定格 .....	715
21.2	DC 特性 .....	716
21.3	AC 特性 .....	718
21.3.1	クロックタイミング .....	719
21.3.2	制御信号タイミング .....	723
21.3.3	バスタイミング .....	725
21.3.4	ダイレクトメモリアクセスコントローラタイミング .....	763
21.3.5	フリーランニングタイマタイミング .....	763
21.3.6	シリアルコミュニケーションインタフェースタイミング .....	765
21.3.7	ウォッチドッグタイマタイミング .....	768
21.3.8	シリアル I/O タイミング .....	769
21.3.9	ハイパフォーマンスユーザデバッグインタフェースタイミング .....	771
21.3.10	I/O ポートタイミング .....	772
21.3.11	イーサネットコントローラタイミング .....	774
21.3.12	STATS、 $\overline{\text{BH}}$ 、 $\overline{\text{BUSHIZ}}$ 信号タイミング .....	777
21.4	AC 特性測定条件 .....	778
<b>付録</b>	<b>.....</b>	<b>779</b>
A.	内蔵周辺モジュールレジスタ .....	779
A.1	アドレス一覧 .....	779
B.	端子状態 .....	793
B.1	リセット、低消費電力状態、バス権解放状態での端子状態 .....	793
C.	型名一覧 .....	796
D.	外形寸法図 .....	797



---

# 図目次

---

## 1. 概要

図 1.1	SH7615 ブロック図.....	9
図 1.2	HD6417615ARF/ARFV ピン配置図 (FP-208C, FP-208CV).....	10
図 1.3	HD6417615ARBP/ARBPV ピン配置図 (BP-240A, BP-240AV).....	11
図 1.4	処理状態の状態遷移図.....	24

## 2. CPU

図 2.1	汎用レジスタの構成.....	30
図 2.2	コントロールレジスタの構成.....	32
図 2.3	システムレジスタの構成.....	34
図 2.4	DSP レジスタの構成.....	36
図 2.5	レジスタのデータ形式.....	38
図 2.6	メモリ上のデータ形式.....	38
図 2.7	DSP タイプデータ形式.....	40
図 2.8	データ転送時の DSP レジスタとバスとの関係.....	44
図 2.9	X、Y データ転送のアドレッシング.....	52
図 2.10	シングルデータ転送のアドレッシング.....	53
図 2.11	モジュロアドレッシング.....	54
図 2.12	DSP 命令の命令形式.....	59
図 2.13	並行処理プログラム例.....	77

## 3. 発振回路と動作モード

図 3.1	内蔵発振回路のブロック図.....	90
図 3.2	水晶発振子の接続例.....	93
図 3.3	外部クロック入力方法.....	94
図 3.4	外部クロック入力方法.....	95
図 3.5	水晶発振子使用時の注意.....	103
図 3.6	PLL 発振回路使用時の注意.....	104

## 4. 例外処理

図 4.1	リピートモードにおける割り込み受け付けの制限.....	121
-------	-----------------------------	-----

## 5. 割り込みコントローラ (INTC)

図 5.1	INTC のブロック図.....	126
図 5.2	外部ベクタモード割り込みの接続例.....	131
図 5.3	オートベクタモード割り込みの接続例.....	132
図 5.4	外部ベクタフェッチ (I : E =1 : 1).....	132
図 5.5	外部ベクタフェッチ (I : E =1 : 1 以外).....	133
図 5.6	外部ベクタフェッチ (I : E =1 : 1 (WAIT 入力)).....	133

図 5.7	外部ベクタフェッチ (I : E = 1 : 1 以外 (WAIT 入力))	134
図 5.8	割り込み動作フロー	164
図 5.9	割り込み例外処理終了後のスタック状態	165
図 5.10	IRL3 ~ IRL0 端子のサンプリング	167
図 5.11	RTE 命令による復帰時のパイプライン動作	169
図 5.12	SR 変更による割り込み許可時のパイプライン動作	169
図 5.13	RTE 命令による復帰時のパイプライン動作	170
図 5.14	SR 変更による割り込み許可時のパイプライン動作	171

## 6. ユーザブレイクコントローラ (UBC)

図 6.1	ユーザブレイクコントローラのブロック図	175
図 6.2	分岐命令を実行する前に割り込みが生じる場合	206

## 7. バスステートコントローラ (BSC)

図 7.1	BSC のブロック図	217
図 7.2	32 ビット外部デバイスとアクセス単位の関係	240
図 7.3	16 ビット外部デバイスとアクセス単位の関係	240
図 7.4	8 ビット外部デバイスとアクセス単位の関係	241
図 7.5	32 ビット外部デバイスとアクセス単位の関係	242
図 7.6	16 ビット外部デバイスとアクセス単位の関係	242
図 7.7	8 ビット外部デバイスとアクセス単位の関係	243
図 7.8	通常空間アクセスの基本タイミング	245
図 7.9	クロック比 I : E = 1 : 1 で、通常空間、16 ビットバス幅に対し、 ロングワードアクセスした場合のタイミング	246
図 7.10	32 ビットデータ幅 SRAM 接続例	247
図 7.11	16 ビットデータ幅 SRAM 接続例	248
図 7.12	8 ビットデータ幅 SRAM 接続例	248
図 7.13	通常空間アクセスのウェイトタイミング (ソフトウェアウェイトのみ)	249
図 7.14	通常空間アクセスのウェイトスタートタイミング (WAIT 信号によるウェイトスタート)	251
図 7.15	CS4 空間における通常空間アクセスのウェイトスタートタイミング	252
図 7.16	CS アサート期間拡張機能	253
図 7.17	シンクロナス DRAM32 ビットデバイス幅接続例	255
図 7.18	シンクロナス DRAM16 ビットデバイス幅接続例	255
図 7.19	(a) パーストリード基本タイミング (オートプリチャージ) I : E = 1 : 1 以外	259
図 7.19	(b) パーストリード基本タイミング (オートプリチャージ) I : E = 1 : 1	259
図 7.20	(a) パーストリードウェイト指定タイミング (オートプリチャージ) I : E = 1 : 1 以外	260
図 7.20	(b) パーストリードウェイト指定タイミング (オートプリチャージ) I : E = 1 : 1	260
図 7.21	(a) シングルリードタイミング (オートプリチャージ) I : E = 1 : 1 以外	261
図 7.21	(b) シングルリードタイミング (オートプリチャージ) I : E = 1 : 1	262
図 7.22	シングルライトモード基本タイミング (オートプリチャージ)	263
図 7.23	(a) パーストライトモード基本タイミング (オートプリチャージ) I : E = 1 : 1 以外	264
図 7.23	(b) パーストライトモード基本タイミング (オートプリチャージ) I : E = 1 : 1	265
図 7.24	(a) パーストリードタイミング (プリチャージなし) I : E = 1 : 1 以外	267
図 7.24	(b) パーストリードタイミング (プリチャージなし) I : E = 1 : 1	267
図 7.25	(a) パーストリードタイミング (バンクアクティブ、同一ロウアドレス) I : E = 1 : 1 以外	268
図 7.25	(b) パーストリードタイミング (バンクアクティブ、同一ロウアドレス) I : E = 1 : 1	268

図 7.26	(a) パーストリードタイミング (バンクアクティブ、異なるロウアドレス) I : E = 1 : 1 以外	269
図 7.26	(b) パーストリードタイミング (バンクアクティブ、異なるロウアドレス) I : E = 1 : 1	269
図 7.27	シングルライトモードタイミング (プリチャージなし)	270
図 7.28	シングルライトモードタイミング (バンクアクティブ、同一ロウアドレス)	271
図 7.29	シングルライトモードタイミング (バンクアクティブ、異なるロウアドレス)	272
図 7.30	オートリフレッシュタイミング	273
図 7.31	セルフリフレッシュタイミング	274
図 7.32	オーバラップイメージ (条件: CS2 空間に SDRAM を接続 (RAS プリチャージ期間を 2 サイクルに設定)、CS3 空間に SDRAM を接続)	275
図 7.33	SDRAM モード書き込みタイミング	277
図 7.34	64M ビットシンクロナス DRAM (2M ワード × 32 ビット品) との接続例	278
図 7.35	128M ビットシンクロナス DRAM (4M ワード × 32 ビット品) との接続例	279
図 7.36	128M ビットシンクロナス DRAM (8M ワード × 16 ビット品) との接続例	280
図 7.37	256M ビットシンクロナス DRAM (8M ワード × 32 ビット品) との接続例	281
図 7.38	DRAM 接続例 (32 ビットデータ幅)	282
図 7.39	DRAM 接続例 (16 ビットデータ幅)	283
図 7.40	基本アクセスタイミング	284
図 7.41	ウェイトステートタイミング	285
図 7.42	外部ウェイトステートタイミング	286
図 7.43	パーストアクセスタイミング	287
図 7.44	$\overline{\text{RAS}}$ ダウンモード、同一ロウアクセスタイミング	288
図 7.45	$\overline{\text{RAS}}$ ダウンモード、異なるロウアクセスタイミング	289
図 7.46	EDO DRAM 接続例 (32 ビットデータ幅)	290
図 7.47	EDO DRAM 接続例 (16 ビットデータ幅)	291
図 7.48	DRAM EDO モードのノーマルアクセスタイミング	292
図 7.49	DRAM EDO モードのパーストアクセスタイミング	293
図 7.50	DMA シングル転送モードでのライトサイクルタイミング (RAS ダウンモード、同一ロウアドレス)	294
図 7.51	DRAM $\overline{\text{CAS}}$ ピフォ $\overline{\text{RAS}}$ リフレッシュサイクルタイミング	295
図 7.52	DRAM セルフリフレッシュサイクルタイミング	296
図 7.53	データ幅とパースト ROM アクセス (ウェイト数 1)	297
図 7.54	パースト ROM ニブルアクセス (2 ウェイトステート)	298
図 7.55	パースト ROM ニブルアクセス (ウェイトなし)	298
図 7.56	アクセスサイクル間ウェイト	300
図 7.57	(a) バスアービトレーションタイミング (E-DMAC リード DMAC16 バイト転送 CPU リード)	303
図 7.57	(b) バスアービトレーションタイミング (E-DMAC ライト DMAC16 バイト転送 CPU リード)	304
図 7.58	バスアービトレーション	306
図 7.59	STATS 出力タイミング	309
図 7.60	BUSHiZ バスタイミング	310
図 7.61	シンクロナス DRAM ライト直後の通常空間アクセス	312

## 8. キャッシュ

図 8.1	キャッシュ構成	316
図 8.2	アドレス構成	316
図 8.3	キャッシュヒット時の読み出し	320

図 8.4	キャッシュミス時の読み出し	321
図 8.5	書き込み (ライトスルー方式)	322
図 8.6	キャッシュヒット時の書き込み (ライトバック方式)	323
図 8.7	キャッシュミス時の書き込み (ライトバック方式)	323
図 8.8	ライトバックバッファの構成	324
図 8.9	キャッシュスルー領域の読み出し	324
図 8.10	LRU 情報とアクセス順序	326
図 8.11	連想パージアクセス	327
図 8.12	データアレイアクセス	328
図 8.13	アドレスアレイアクセス	329
図 8.14	キャッシュの初期化	330
図 8.15	特定アドレスのパージ	330
図 8.16	2 ウェイモード時の 2K バイト RAM のアドレスマッピング	332

## 9. イーサネットコントローラ (EtherC)

図 9.1	EtherC の構成	336
図 9.2	EtherC 送信部状態遷移	363
図 9.3	EtherC 受信部状態遷移	364
図 9.4	(a) MII フレーム送信タイミング (正常時)	365
図 9.4	(b) MII フレーム送信タイミング (衝突発生)	365
図 9.4	(c) MII フレーム送信タイミング (送信エラー発生)	366
図 9.4	(d) MII フレーム受信タイミング (正常受信)	366
図 9.4	(e) MII フレーム受信タイミング (受信エラー (1))	366
図 9.4	(f) MII フレーム受信タイミング (受信エラー (2))	366
図 9.5	MII 管理フレームフォーマット	367
図 9.6	(a) 1 ビットデータの書き込みフロー	368
図 9.6	(b) バス解放フロー (図 9.5 中のリード時の TA)	368
図 9.6	(c) 1 ビットデータ読み出しフロー	369
図 9.6	(d) 単独バス解放フロー (図 9.5 中のライト時の IDLE)	369
図 9.7	AM79C873 との接続例	372
図 9.8	DP83843 との接続例	372

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

図 10.1	E-DMAC とディスクリプタおよびバッファの構成	376
図 10.2	送信ディスクリプタと送信バッファの関係	409
図 10.3	受信ディスクリプタと受信バッファの関係	412
図 10.4	送信フローの例	415
図 10.5	受信フローの例	416
図 10.6	送信エラー発生後の E-DMAC 動作	417
図 10.7	受信エラー発生後の E-DMAC 動作	418

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

図 11.1	DMAC のブロック図	423
図 11.2	DMA 転送フローチャート	437
図 11.3	バーストモード時の固定モード DMA 転送 (デュアルアドレス、DREQn 立ち下がりエッジ検出)	442

図 11.4	サイクルスチールモード時の固定モード DMA 転送 (デュアルアドレス、DREQn ローレベル検出).....	442
図 11.5	ラウンドロビンモードでのチャンネル優先順位変更例.....	443
図 11.6	シングルアドレスモードでのデータの流れ.....	445
図 11.7	シングルアドレスモードでの DMA 転送タイミング例.....	446
図 11.8	デュアルアドレスモードでのデータの流れ.....	447
図 11.9	デュアルアドレスモードでの DMA 転送タイミング例 (外部メモリ空間 外部メモリ空間、読み出しサイクルに DACKn 出力の場合).....	448
図 11.10	サイクルスチールモードでの DMA 転送例 (デュアルアドレス、DREQn ローレベル検出).....	449
図 11.11	バーストモードでの DMA 転送例 (シングルアドレス、DREQn 立ち下がり検出).....	449
図 11.12	複数チャンネルが動作する場合のバス状態 (優先順位: チャンネル 0 > チャンネル 1、チャンネル 1 はバーストモード、 チャンネル 0 はサイクルスチールモードが設定されている場合).....	451
図 11.13	DACKn 出力タイミングの例.....	452
図 11.14	通常空間アクセスにおける DACKn 出力 (AM = 0).....	452
図 11.15	通常空間アクセスにおける DACKn 出力 (AM = 1).....	453
図 11.16	通常空間アクセスにおける DACKn 出力 (AM = 0、16 ビット外部デバイスにロングワードアクセスをした場合).....	453
図 11.17	通常空間アクセスにおける DACKn 出力 (AM = 0、8 ビット外部デバイスにロングワードアクセスをした場合).....	453
図 11.18	通常空間アクセスにおける DACKn 出力 (AM = 0、8 ビット外部デバイスにワードアクセスをした場合).....	454
図 11.19	シンクロナス DRAM バーストリード (オートプリチャージ) における DACKn 出力 (AM = 0).....	454
図 11.20	シンクロナス DRAM シングルリード (オートプリチャージ) における DACKn 出力 (AM = 0).....	455
図 11.21	シンクロナス DRAM ライト (オートプリチャージ) における DACKn 出力 (AM = 1).....	455
図 11.22	シンクロナス DRAM バーストリード (バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 0).....	455
図 11.23	シンクロナス DRAM バーストリード (バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 0).....	456
図 11.24	シンクロナス DRAM シングルリード (バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 0).....	456
図 11.25	シンクロナス DRAM シングルリード (バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 0).....	457
図 11.26	シンクロナス DRAM ライト (バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 1).....	457
図 11.27	シンクロナス DRAM ライト (バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 1).....	457
図 11.28	(a) シンクロナス DRAM1 サイクルライトタイミング.....	458
図 11.28	(b) シンクロナス DRAM1 サイクルライトタイミング.....	459
図 11.29	DRAM ノーマルアクセス時の DACKn 出力 (AM = 0、1 同様).....	459
図 11.30	DRAM バーストアクセス時 (同一ロウアドレス) の DACKn 出力 (AM = 0、1 同様).....	460
図 11.31	DRAM バーストアクセス時 (異なるロウアドレス) の DACKn 出力 (AM = 0、1 同様).....	460
図 11.32	バースト ROM ニブルアクセス時の DACKn 出力.....	460
図 11.33	DREQn、DACKn のハンドシェイク.....	461
図 11.34	サイクルスチールモード・エッジ検出設定時の DREQn 端子入力検出タイミング.....	462
図 11.35	16 ビット外部デバイス接続時 (エッジ検出).....	462
図 11.36	8 ビット外部デバイス接続時 (エッジ検出).....	462
図 11.37	サイクルスチール・エッジ検出 16 バイト転送設定時の DREQn 端子入力検出タイミング.....	463
図 11.38	サイクルスチールモード・レベル検出 バイト・ワード・ロングワード設定時の DREQn 端子入力検出タイミング.....	464

図 11.39	16 ビット外部デバイス接続時 (レベル検出).....	464
図 11.40	8 ビット外部デバイス接続時 (レベル検出).....	464
図 11.41	バーストモード・エッジ検出設定時の DREQn 端子入力検出タイミング.....	465
図 11.42	BH 使用時のレジスタ設定.....	467
図 11.43	BH 出力の概略タイミング.....	467

## 12. 16 ビットフリーランニングタイマ (FRT)

図 12.1	FRT のブロック図.....	476
図 12.2	(a) FRC のアクセス動作 (CPU FRC [H'AA55] 書き込み時).....	486
図 12.2	(b) FRC のアクセス動作 (FRC CPU [H'AA55] 読み出し時).....	487
図 12.3	内部クロック動作時のカウントタイミング.....	488
図 12.4	外部クロック動作時のカウントタイミング.....	488
図 12.5	アウトプットコンペア A 出力タイミング.....	489
図 12.6	コンペアマッチ A によるクリアタイミング.....	489
図 12.7	インプットキャプチャ信号タイミング (通常時).....	490
図 12.8	インプットキャプチャ信号タイミング (FICR の読み出し時に、インプットキャプチャ入力を入力した場合).....	490
図 12.9	ICF のセットタイミング.....	491
図 12.10	OCF セットタイミング.....	491
図 12.11	OVF のセットタイミング.....	492
図 12.12	パルス出力例.....	493
図 12.13	FRC のライトとクリアの競合.....	494
図 12.14	FRC ライトとカウントアップの競合.....	495
図 12.15	OCR とコンペアマッチの競合.....	496

## 13. ウォッチドッグタイマ (WDT)

図 13.1	WDT のブロック図.....	500
図 13.2	WTCNT、WTCSTR への書き込み.....	506
図 13.3	RSTCSR への書き込み.....	507
図 13.4	ウォッチドッグタイマモード時の動作.....	508
図 13.5	インターバルタイマモード時の動作.....	509
図 13.6	オーバフローフラグ (OVF) のセットタイミング.....	510
図 13.7	ウォッチドッグタイマオーバフローフラグ (WOVF) のセットタイミング.....	510
図 13.8	WTCNT の書き込みとカウントアップの競合.....	511
図 13.9	$\overline{\text{WDTOVF}}$ 信号によるシステムリセット回路例.....	512

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

図 14.1	SCIF のブロック図.....	515
図 14.2	IrDA のブロック図.....	516
図 14.3	調歩同期式通信のデータフォーマット (8 ビットデータ/パリティあり/2 ストップビット/LSB ファーストの例).....	545
図 14.4	SCIF の初期化フローチャートの例.....	548
図 14.5	シリアル送信のフローチャートの例.....	549
図 14.6	調歩同期式モードでの送信時の動作例 (8 ビットデータ/パリティあり/1 ストップビット/LSB ファーストの例).....	551
図 14.7	モデムコントロールを使用した動作例 ( $\overline{\text{CTS}}$ ).....	551
図 14.8	シリアル受信のフローチャートの例 (1).....	552

図 14.8	シリアル受信のフローチャートの例 (2).....	553
図 14.9	SCIF の受信時の動作例 (8ビットデータ/パリティあり / 1ストップビット / LSB ファーストの例).....	555
図 14.10	モデムコントロール使用時の動作例.....	555
図 14.11	マルチプロセッサフォーマットを使用したプロセッサ間通信の例 (受信局 A へのデータ H'AA の送信の例).....	556
図 14.12	マルチプロセッサシリアル送信のフローチャートの例.....	558
図 14.13	SCIF の送信時の動作例 (8ビットデータ/マルチプロセッサビットあり / 1ストップビット / LSB ファーストの例).....	560
図 14.14	マルチプロセッサシリアル受信のフローチャートの例 (1).....	561
図 14.14	マルチプロセッサシリアル受信のフローチャートの例 (2).....	562
図 14.15	SCIF の受信時の動作例 (8ビットデータ/マルチプロセッサビットあり / 1ストップビット / LSB ファーストの例).....	563
図 14.16	クロック同期式通信のデータフォーマット (LSB ファーストの例).....	564
図 14.17	SCIF の初期化フローチャートの例.....	566
図 14.18	シリアル送信のフローチャートの例.....	567
図 14.19	SCIF の送信時の動作例 (LSB ファーストの例).....	568
図 14.20	シリアルデータ受信フローチャートの例 (1).....	569
図 14.20	シリアルデータ受信フローチャートの例 (2).....	570
図 14.21	SCIF の受信時の動作例 (LSB ファーストの例).....	571
図 14.22	シリアルデータ送受信フローチャートの例.....	572
図 14.23	送信 / 受信 FIFO の構成.....	573
図 14.24	IrDA モード送受信動作.....	576
図 14.25	調歩同期式モードの受信データサンプリングタイミング (基本クロックは転送レートの 16 倍の周波数を使用、8 クロック目でサンプリング時).....	580
図 14.26	DMAC によるクロック同期式送信時の例.....	581
図 14.27	SCIF 初期化フローチャート.....	583

## 15. シリアル I/O (SIO)

図 15.1	SIO ブロック図.....	586
図 15.2	受信: インターバル転送モード.....	594
図 15.3	受信: 連続転送モード.....	594
図 15.4	送信: TM = 0 モードでのインターバル転送モード.....	595
図 15.5	送信: TM = 0 モードでの連続転送モード.....	595
図 15.6	送信: TM = 1 モードでのインターバル転送モード.....	596
図 15.7	送信: TM = 1 での連続転送モード.....	596

## 16. 16 ビットタイマパルスユニット (TPU)

図 16.1	TPU のブロック図.....	602
図 16.2	16 ビットレジスタのアクセス動作 [バスマスタ TCNT (16 ビット)].....	622
図 16.3	8 ビットレジスタのアクセス動作 [バスマスタ TCR (上位 8 ビット)].....	622
図 16.4	8 ビットレジスタのアクセス動作 [バスマスタ TMDR (下位 8 ビット)].....	623
図 16.5	8 ビットレジスタのアクセス動作 [バスマスタ TCR、TMDR (16 ビット)].....	623
図 16.6	カウンタ動作設定手順例.....	625
図 16.7	フリーランニングカウンタの動作.....	625
図 16.8	周期カウンタの動作.....	626
図 16.9	コンペアマッチによる波形出力動作例.....	626
図 16.10	0 出力 / 1 出力の動作例.....	627
図 16.11	トグル出力の動作例.....	627

図 16.12	インプットキャプチャ動作の設定例.....	628
図 16.13	インプットキャプチャ動作例.....	628
図 16.14	同期動作の設定手順例.....	629
図 16.15	同期動作の動作例.....	630
図 16.16	コンペアマッチバッファ動作.....	631
図 16.17	インプットキャプチャバッファ動作.....	631
図 16.18	バッファ動作の設定手順例.....	631
図 16.19	バッファ動作例 (1).....	632
図 16.20	バッファ動作例 (2).....	633
図 16.21	PWM モードの設定手順例.....	634
図 16.22	PWM モードの動作例 (1).....	635
図 16.23	PWM モードの動作例 (2).....	635
図 16.24	PWM モード動作例 (3).....	636
図 16.25	位相計数モードの設定手順例.....	637
図 16.26	位相計数モード 1 の動作例.....	638
図 16.27	位相計数モード 2 の動作例.....	639
図 16.28	位相計数モード 3 の動作例.....	640
図 16.29	位相計数モード 4 の動作例.....	641
図 16.30	内部クロック動作時のカウントタイミング.....	644
図 16.31	外部クロック動作時のカウントタイミング.....	644
図 16.32	アウトプットコンペア出力タイミング.....	645
図 16.33	インプットキャプチャ入力信号タイミング.....	645
図 16.34	カウンタクリアタイミング (コンペアマッチ).....	646
図 16.35	カウンタクリアタイミング (インプットキャプチャ).....	646
図 16.36	バッファ動作タイミング (コンペアマッチ).....	647
図 16.37	バッファ動作タイミング (インプットキャプチャ).....	647
図 16.38	TGI 割り込みタイミング (コンペアマッチ).....	648
図 16.39	TGI 割り込みタイミング (インプットキャプチャ).....	648
図 16.40	TCIV 割り込みのセットタイミング.....	649
図 16.41	TCIU 割り込みのセットタイミング.....	649
図 16.42	CPU によるステータスフラグのクリアタイミング.....	650
図 16.43	DMAC の起動によるステータスフラグのクリアタイミング.....	650
図 16.44	位相計数モード時の位相差、オーバーラップ、およびパルス幅.....	651
図 16.45	TCNT のライトとクリアの競合.....	652
図 16.46	TCNT のライトとカウントアップの競合.....	652
図 16.47	TGR のライトとコンペアマッチの競合.....	653
図 16.48	バッファレジスタのライトとコンペアマッチの競合.....	653
図 16.49	TGR のリードとインプットキャプチャの競合.....	654
図 16.50	TGR のライトとインプットキャプチャの競合.....	654
図 16.51	バッファレジスタのライトとインプットキャプチャの競合.....	655
図 16.52	オーバーフローとカウンタクリアの競合.....	655
図 16.53	TCNT のライトとオーバーフローの競合.....	656

## 17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

図 17.1	H-UDI ブロック図.....	660
図 17.2	TAP コントローラ状態遷移図.....	674

図 17.3	データ入出力タイミングチャート (1).....	675
図 17.4	データ入出力タイミングチャート (2).....	676
図 17.5	データ入出力タイミングチャート (3).....	676
図 17.6	シリアルデータ入出力 (1).....	680
図 17.7	シリアルデータ入出力 (2).....	681

## 18. ピンファンクションコントローラ (PFC)

## 19. I/O ポート

図 19.1	ポート A.....	696
図 19.2	ポート B.....	698

## 20. 低消費電力状態

図 20.1	NMI 割り込みによるスタンバイモード解除方法.....	711
図 20.2	クロックポーズ機能のタイミングチャート (PLL 回路 1 動作時).....	712
図 20.3	クロックポーズ機能のタイミングチャート (PLL 回路 1 停止時).....	713
図 20.4	クロックポーズ機能のタイミングチャート (NMI 入力による解除).....	713

## 21. 電気的特性

図 21.1	EXTAL クロック入力タイミング.....	720
図 21.2	CKIO クロック入力タイミング.....	720
図 21.3	CKIO クロック出力タイミング.....	720
図 21.4	CKPO クロック出力タイミング.....	721
図 21.5	パワーオン時発振安定時間.....	721
図 21.6	スタンバイ復帰時発振安定時間 (RES による復帰).....	721
図 21.7	スタンバイ復帰時発振安定時間 (NMI による復帰).....	722
図 21.8	PLL 同期安定化時間.....	722
図 21.9	リセット入力タイミング.....	723
図 21.10	割り込み信号入力タイミング.....	724
図 21.11	バス権解放タイミング.....	724
図 21.12	基本読み出しサイクル (ノーウェイト).....	728
図 21.13	基本書き込みサイクル (ノーウェイト).....	729
図 21.14	基本バスサイクル (1 ウェイトサイクル).....	730
図 21.15	基本バスサイクル (外部ウェイト入力).....	731
図 21.16	シンクロナス DRAM 読み出しバスサイクル (RCD=1 サイクル、CAS レイテンシ=1 サイクル、バースト=4).....	732
図 21.17	シンクロナス DRAM シングル読み出しバスサイクル (RCD= 1 サイクル、CAS レイテンシ =1 サイクル、バースト=4).....	733
図 21.18	シンクロナス DRAM 読み出しバスサイクル (RCD=2 サイクル、CAS レイテンシ=2 サイクル、バースト= 4 ).....	734
図 21.19	シンクロナス DRAM 読み出しバスサイクル (バンクアクティブ、同一ロウアクセス、CAS レイテンシ =1 サイクル).....	735
図 21.20	シンクロナス DRAM 読み出しバスサイクル (バンクアクティブ、同一ロウアクセス、CAS レイテンシ =2 サイクル).....	736
図 21.21	シンクロナス DRAM 読み出しバスサイクル (バンクアクティブ、別ロウアクセス、TRP=1 サイクル、 RCD=1 サイクル、CAS レイテンシ =1 サイクル).....	737
図 21.22	シンクロナス DRAM 読み出しバスサイクル (バンクアクティブ、別ロウアクセス、TRP=2 サイクル、 RCD=1 サイクル、CAS レイテンシ =1 サイクル).....	738

☒ 21.23	シンクロナス DRAM 書き込みバスサイクル (RASD=0、RCD=1 サイクル、TRWL=1 サイクル).....	739
☒ 21.24	シンクロナス DRAM 書き込みバスサイクル (RASD=0、RCD=2 サイクル、TRWL=2 サイクル).....	740
☒ 21.25	シンクロナス DRAM 書き込みバスサイクル (バンクアクティブ、同一ロウアクセス、I : E = 1 : 1 以外).....	741
☒ 21.26	シンクロナス DRAM 書き込みサイクル (バンクアクティブ、同一ロウアクセス、I : E = 1 : 1).....	742
☒ 21.27	シンクロナス DRAM 連続書き込みサイクル (バンクアクティブ、同一ロウアクセス、I : E = 1 : 1 以外).....	743
☒ 21.28	シンクロナス DRAM 連続書き込みサイクル (バンクアクティブ、同一ロウアクセス、I : E = 1 : 1).....	744
☒ 21.29	シンクロナス DRAM 書き込みバスサイクル (バンクアクティブ、別ロウアクセス、TRP=1 サイクル、RCD=1 サイクル).....	745
☒ 21.30	シンクロナス DRAM 書き込みバスサイクル (バンクアクティブ、別ロウアクセス、TRP=2 サイクル、RCD=2 サイクル).....	746
☒ 21.31	シンクロナス DRAM オートリフレッシュサイクル (TRAS=4 サイクル).....	747
☒ 21.32	シンクロナス DRAM オートリフレッシュサイクル (プリチャージサイクルから図示、TRP=1 サイクル、TRAS=4 サイクル).....	748
☒ 21.33	シンクロナス DRAM セルフリフレッシュサイクル (TRAS=3).....	749
☒ 21.34	DRAM 読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト).....	750
☒ 21.35	DRAM 書き込みサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト).....	751
☒ 21.36	DRAM バスサイクル (TRP=2 サイクル、RCD=2 サイクル、1 ウェイト).....	752
☒ 21.37	DRAM バスサイクル (TRP=1 サイクル、RCD=1 サイクル、外部ウェイト入力).....	753
☒ 21.38	DRAM パースト読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト).....	754
☒ 21.39	DRAM パースト書き込みサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト).....	755
☒ 21.40	EDO リードサイクル (TRP = 1 サイクル、RCD = 1 サイクル、ノーウェイト).....	756
☒ 21.41	EDO パースト読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト).....	757
☒ 21.42	DRAM $\overline{\text{CAS}}$ ビフォ $\overline{\text{RAS}}$ リフレッシュサイクル (TRP=1 サイクル、TRAS=2 サイクル).....	758
☒ 21.43	パースト ROM 読み出しサイクル (ウェイト=1).....	759
☒ 21.44	割り込みベクタフェッチサイクル (ノーウェイト、I : E = 1 : 1).....	760
☒ 21.45	割り込みベクタフェッチサイクル (ノーウェイト、I : E = 1 : 1 以外).....	761
☒ 21.46	割り込みベクタフェッチサイクル (外部ウェイト入力 I : E = 1 : 1 以外).....	762
☒ 21.47	REFOUT 遅延時間.....	762
☒ 21.48	DREQ0、DREQ1 入力タイミング.....	763
☒ 21.49	FRT 入出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ ).....	764
☒ 21.50	FRT 入出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ 以外).....	764
☒ 21.51	FRT クロック入力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ ).....	764
☒ 21.52	FRT クロック入力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ 以外).....	764
☒ 21.53	入力クロック入出力タイミング.....	765
☒ 21.54	SCI 入出力タイミング (クロック同期式モード).....	765
☒ 21.55	$\overline{\text{RTS}}$ 、 $\overline{\text{CTS}}$ 入出力タイミング.....	766
☒ 21.56	TPU 入出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ ).....	767
☒ 21.57	TPU 入出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ 以外).....	767
☒ 21.58	TPU クロック入力タイミング.....	767
☒ 21.59	ウォッチドッグタイマ出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ ).....	768
☒ 21.60	ウォッチドッグタイマ出力タイミング ( $t_{\text{Eycyc}} : t_{\text{Pycyc}} = 1 : 1$ 以外).....	768
☒ 21.61	SIO 入力クロックタイミング.....	769
☒ 21.62	SIO 受信タイミング.....	769
☒ 21.63	SIO 送信タイミング (TMn = 0 モード).....	770
☒ 21.64	SIO 送信タイミング (TMn = 1 モード).....	770

図 21.65	H-UDI クロックタイミング.....	771
図 21.66	H-UDI TRST タイミング.....	771
図 21.67	H-UDI 入出力タイミング.....	772
図 21.68	I/O ポート入出力タイミング ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ ).....	773
図 21.69	I/O ポート入出力タイミング ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ 以外).....	773
図 21.70	MII 送信タイミング (正常動作時).....	774
図 21.71	MII 送信タイミング (衝突発生ケース).....	775
図 21.72	MII 受信タイミング (正常動作時).....	775
図 21.73	MII 受信タイミング (エラー発生ケース).....	775
図 21.74	MDIO 入力タイミング.....	775
図 21.75	MDIO 出力タイミング.....	776
図 21.76	WOL 出力タイミング.....	776
図 21.77	EXOUT 出力タイミング.....	776
図 21.78	STATS 出力タイミング.....	777
図 21.79	BH 出力タイミング.....	777
図 21.80	BUSHiZ バスタイミング.....	777
図 21.81	出力付加回路.....	778

## 付録

図 D.1	外形寸法図 (FP-208C、FP-208CV).....	797
図 D.2	外形寸法図 (BP-240A、BP-240AV).....	798



---

# 表目次

---

## 1. 概要

表 1.1	特長 (1).....	1
表 1.1	特長 (2).....	2
表 1.1	特長 (3).....	3
表 1.1	特長 (4).....	4
表 1.1	特長 (5).....	5
表 1.1	特長 (6).....	6
表 1.1	特長 (7).....	7
表 1.1	特長 (8).....	8
表 1.2	端子の機能 (1).....	12
表 1.2	端子の機能 (2).....	13
表 1.2	端子の機能 (3).....	15
表 1.2	端子の機能 (4).....	17
表 1.3	端子マルチプレクスの説明 (1).....	18
表 1.3	端子マルチプレクスの説明 (2).....	20
表 1.3	端子マルチプレクスの説明 (3).....	21
表 1.3	端子マルチプレクスの説明 (4).....	22
表 1.3	端子マルチプレクスの説明 (5).....	23
表 1.4	低消費電力状態.....	27

## 2. CPU

表 2.1	SR レジスタのビット.....	33
表 2.2	DSR レジスタのビット.....	36
表 2.3	レジスタの初期値.....	37
表 2.4	DSP 命令のソースレジスタのデータ形式.....	43
表 2.5	DSP 命令のデスティネーションレジスタのデータ形式.....	43
表 2.6	ワードデータの符号拡張.....	45
表 2.7	遅延分岐命令.....	45
表 2.8	T ビット.....	46
表 2.9	イミディエイトデータによる参照.....	46
表 2.10	絶対アドレスによる参照.....	46
表 2.11	ディスプレイメントによる参照.....	47
表 2.12	CPU 命令のアドレッシングモードと実効アドレス.....	48
表 2.13	データ転送命令の概要.....	50
表 2.14	CPU 命令の命令形式.....	56
表 2.15	ダブルデータ転送の命令形式.....	60
表 2.16	シングルデータ転送命令の命令形式.....	60
表 2.17	A フィールドの並行データ転送命令.....	61
表 2.18	B フィールドの ALU 演算命令、乗算命令.....	62
表 2.19	CPU 命令の分類.....	63

表 2.20	追加された CPU 命令	72
表 2.21	DSP データ転送命令の分類	73
表 2.22	DSP データ転送のオペランドとレジスタとの対応	75
表 2.23	DSP 演算命令の命令形式	76
表 2.24	DSP 命令のオペランドとレジスタの対応	77
表 2.25	DSP 演算命令の分類	78
表 2.26	NOPX と NOPY の命令コードの例	85

### 3. 発振回路と動作モード

表 3.1	端子の機能	91
表 3.2	動作モード	92
表 3.3	クロックモードの端子設定と状態	93
表 3.4	クロックモードの端子設定と周波数変更レジスタの初期値	95
表 3.5	レジスタ構成	96
表 3.6	CS0 空間のバス幅指定	105

### 4. 例外処理

表 4.1	例外処理の種類と優先順位	107
表 4.2	例外要因検出と例外処理タイミング	108
表 4.3	(a) 例外処理ベクタテーブル	109
表 4.3	(b) 例外処理ベクタテーブル (IRQ モード)	110
表 4.3	(c) 例外処理ベクタテーブル (IRL モード)	110
表 4.4	例外処理ベクタテーブルアドレスの算出法	111
表 4.5	リセットの種類	112
表 4.6	バスサイクルとアドレスエラー	114
表 4.7	割り込み要因	116
表 4.8	割り込み優先順位	117
表 4.9	命令による例外の種類	118
表 4.10	遅延分岐命令、割り込み禁止命令の直後の例外要因発生	120
表 4.11	例外処理終了後のスタックの状態	122

### 5. 割り込みコントローラ (INTC)

表 5.1	端子構成	127
表 5.2	レジスタ構成	128
表 5.3	IRL 割り込みの優先レベルとオートベクタ番号	131
表 5.4	(a) 割り込み例外処理ベクタと優先順位 (IRL モード)	136
表 5.4	(b) 割り込み例外処理ベクタと優先順位 (IRQ モード)	139
表 5.5	割り込み要求元と IPR.A ~ IPRE	144
表 5.6	割り込み要求元とベクタ番号設定レジスタ (1)	159
表 5.7	割り込み要求元とベクタ番号設定レジスタ (2)	160
表 5.8	割り込み応答時間	166

### 6. ユーザブレイクコントローラ (UBC)

表 6.1	レジスタ構成	176
表 6.2	データアクセスサイクルのアドレスとオペランドサイズの比較条件	202
表 6.3	分岐先命令実行前の例外処理で格納される BSA 値	206

## 7. バスステートコントローラ (BSC)

表 7.1	端子構成 (1).....	218
表 7.2	レジスタ構成.....	220
表 7.3	アドレスマップ.....	221
表 7.4	BCR1、3 のレジスタ設定におけるウェイト値 (全空間共通).....	224
表 7.5	$\overline{CSn}$ 空間の $T_w$ 指定ビット.....	250
表 7.6	SZ、AMX とアドレスマルチプレクス出力の関係.....	257
表 7.7	Tap サイクルと次アクセスとのオーバーラップが発生する状態.....	275
表 7.8	AMX 1 ~ 0 とアドレスマルチプレクスの関係.....	283
表 7.9	エンコード結果.....	309
表 7.10	アクセスシーケンス.....	311

## 8. キャッシュ

表 8.1	レジスタ構成.....	316
表 8.2	アドレス空間とキャッシュ動作.....	319
表 8.3	LRU 情報の更新.....	326
表 8.4	置き換えウェイの選択条件.....	326

## 9. イーサネットコントローラ (EtherC)

表 9.1	MII 端子機能.....	337
表 9.2	レジスタ構成.....	338

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

表 10.1	E-DMAC レジスタ.....	377
--------	------------------	-----

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

表 11.1	端子構成.....	424
表 11.2	レジスタ構成.....	424
表 11.3	AR、RS ビットによる DMA 転送要求の選択.....	438
表 11.4	TA、AM ビットによる外部リクエストモードの選択.....	439
表 11.5	DS、DL ビットによる外部リクエスト信号の選択.....	439
表 11.6	AR、RS ビットによる内蔵周辺モジュールリクエストモードの選択.....	441
表 11.7	サポートできる DMA 転送.....	444
表 11.8	表 DMA 転送区間とバスモード、リクエストモードとの関係一覧.....	450
表 11.9	外部リクエストモード時に設定可能なリクエストモード、バスモード、アドレスモードの組み合わせ一覧.....	451
表 11.10	内蔵 SCIF と外部メモリ間転送の転送条件とレジスタ設定値.....	467

## 12. 16 ビットフリーランニングタイム (FRT)

表 12.1	端子構成.....	477
表 12.2	レジスタ構成.....	477
表 12.3	FRT 割り込み要因.....	492
表 12.4	内部クロックの切り替えと FRC 動作 (1).....	497
表 12.4	内部クロックの切り替えと FRC 動作 (2).....	498

### 13. ウォッチドッグタイマ (WDT)

表 13.1	端子構成.....	500
表 13.2	レジスタ構成.....	501

### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

表 14.1	端子構成.....	516
表 14.2	レジスタ構成.....	517
表 14.3	ビットレートに対する SCBRR の設定例 [ 調歩同期式モード ] ( 1 ).....	534
表 14.3	ビットレートに対する SCBRR の設定例 [ 調歩同期式モード ] ( 2 ).....	535
表 14.3	ビットレートに対する SCBRR の設定例 [ 調歩同期式モード ] ( 3 ).....	536
表 14.4	ビットレートに対する SCBBR の設定例 [ クロック同期式モード ].....	536
表 14.5	ポーレートジェネレータを使用する場合の 各周波数における最大ビットレート ( 調歩同期式モード ).....	537
表 14.6	外部クロック入力時の最大ビットレート ( 調歩同期式モード ).....	537
表 14.7	外部クロック入力時の最大ビットレート ( クロック同期式モード ).....	538
表 14.8	SCSMR、SCIMR の設定値とシリアル送信 / 受信フォーマット.....	544
表 14.9	SCSMR、SCSCR の設定と SCIF のクロックソースの選択.....	544
表 14.10	シリアル送信 / 受信フォーマット ( 調歩同期式モード ).....	546
表 14.11	受信エラーと発生条件.....	554
表 14.12	IrDA モード時の ICK3 ~ 0 ビットと動作周波数の関係 ( PSEL = 1 のとき ).....	577
表 14.13	SCIF 割り込み要因.....	578
表 14.14	SC1SSR、SC2SSR のステータスフラグの状態と受信データの転送.....	579

### 15. シリアル I/O (SIO)

表 15.1	SIO の外部端子.....	587
表 15.2	レジスタ構成.....	588
表 15.3	SIO 割り込み要因.....	597

### 16. 16 ビットタイマパルスユニット (TPU)

表 16.1	TPU 機能一覧 ( 1 ).....	600
表 16.1	TPU 機能一覧 ( 2 ).....	601
表 16.2	端子構成.....	603
表 16.3	レジスタ構成.....	604
表 16.4	TPU のクロックソース一覧.....	607
表 16.5	レジスタの組み合わせ.....	630
表 16.6	各 PWM 出力のレジスタと出力端子.....	634
表 16.7	位相計数モードクロック入力端子.....	637
表 16.8	位相計数モード 1 のアップ / ダウンカウント条件.....	638
表 16.9	位相計数モード 2 のアップ / ダウンカウント条件.....	639
表 16.10	位相計数モード 3 のアップ / ダウンカウント条件.....	640
表 16.11	位相計数モード 4 のアップ / ダウンカウント条件.....	641
表 16.12	TPU 割り込み一覧.....	642

### 17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

表 17.1	端子構成.....	661
--------	-----------	-----

表 17.2	レジスタ構成.....	661
表 17.3	H-UDI レジスタのシリアル転送.....	662
表 17.4	インストラクション構成.....	663
表 17.5	端子とバウンダリスキャンレジスタの対応.....	665

## 18. ピンファンクションコントローラ (PFC)

表 18.1	マルチプレクス端子一覧.....	683
表 18.2	レジスタ構成.....	684

## 19. I/O ポート

表 19.1	レジスタ構成.....	696
表 19.2	ポート A データレジスタ (PADR) の読み出し / 書き込み時の動作.....	697
表 19.3	レジスタ構成.....	698
表 19.4	ポート B データレジスタ (PBDR) の読み出し / 書き込み時の動作.....	699

## 20. 低消費電力状態

表 20.1	低消費電力状態.....	702
表 20.2	レジスタ構成.....	702
表 20.3	スタンバイモード時のレジスタ状態.....	709

## 21. 電気的特性

表 21.1	絶対最大定格.....	715
表 21.2	DC 特性.....	716
表 21.3	出力許容電流値.....	717
表 21.4	最大動作周波数.....	718
表 21.5	クロックタイミング.....	719
表 21.6	制御信号タイミング.....	723
表 21.7	PLL オンバスタイミング [モード 0, 4] (1).....	725
表 21.7	PLL オンバスタイミング [モード 0, 4] (2).....	726
表 21.7	PLL オンバスタイミング [モード 0, 4] (3).....	727
表 21.8	ダイレクトメモリアクセスコントローラタイミング.....	763
表 21.9	フリーランニングタイマタイミング.....	763
表 21.10	シリアルコミュニケーションインタフェースタイミング.....	765
表 21.11	16 ビットタイマパルスユニット.....	766
表 21.12	ウォッチドッグタイマタイミング.....	768
表 21.13	シリアル I/O タイミング.....	769
表 21.14	ハイパフォーマンスユーザデバックインタフェースタイミング.....	771
表 21.15	I/O ポートタイミング.....	772
表 21.16	イーサネットコントローラタイミング.....	774
表 21.17	STATS、 $\overline{\text{BH}}$ 、 $\overline{\text{BUSHIZ}}$ 信号タイミング.....	777

## 付録

表 C.1	SH7615 型名一覧.....	796
-------	------------------	-----



# 1. 概要

## 1.1 イーサネットコントローラ搭載 SuperH マイコンの特長

本 LSI は、ルネサステクノロジオリジナル・アーキテクチャを採用した高速 CPU を核にして、イーサネットシステムに必要な周辺機能を集積した CMOS シングルチップマイコンです。

本 LSI の CPU は、RISC ( Reduced Instruction Set Computer ) タイプの命令セットを持っており、基本的に 1 命令 1 サイクルで動作するので、命令実行速度が飛躍的に向上しています。また内部 32 ビット構成となっており、データ処理能力を強化しています。さらに、従来の DSP 機能を強化しており、拡張ハーバード型アーキテクチャの DSP データバス機能を実現しています。本 LSI の CPU によって、従来のマイコンでは実現が不可能だった、高速性が要求されるリアルタイム制御等のアプリケーションでも、より低コストでかつ高性能 / 高機能なシステムを組むことができるようになります。その上、本 LSI には最大 4kB のキャッシュを搭載しており、外部メモリアクセス時の CPU 処理性能向上が可能です。

本 LSI は、IEEE802.3u 規格に準拠したメディアアクセスコントローラ ( MAC ) とメディアインディペンデントインタフェース ( MII ) 標準ユニットを含むイーサネットコントローラを搭載しており、10/100Mbps での LAN 接続を実現することができます。また、システムに必要な周辺機能として、RAM、タイマ、FIFO 付シリアルコミュニケーションインタフェース ( SCIF )、割り込みコントローラ ( INTC )、I/O ポート等をサポートします。

表 1.1 特長 ( 1 )

項目	仕様
CPU	ルネサステクノロジオリジナルアーキテクチャを採用 内部 32 ビット構成 汎用レジスタマシン ・ 汎用レジスタ 32 ビット × 16 本 ・ コントロールレジスタ 32 ビット × 6 本 ( 内 3 本は DSP 用に追加 ) ・ システムレジスタ 32 ビット × 10 本 RISC ( Reduced Instruction Set Computer ) タイプの命令セット ・ 命令長 : 16 ビット固定長による、コード効率の向上 ・ ロードストアアーキテクチャ ( 基本演算はレジスタ間で実行 ) ・ 遅延分岐命令の採用で、分岐時のパイプラインの乱れを軽減 ・ C 言語指向の命令セット 命令実行時間 1 命令 / 1 サイクル ( 62.5MHz 動作時 : 16.0ns / 命令 ) アドレス空間 アーキテクチャ上は 4GB 乗算器内蔵 乗算器内蔵により、32 × 32 64 乗算を 2 ~ 4 サイクル実行 32 × 32 + 64 64 積和演算を 2 ~ 4 サイクル実行 パイプライン 5 段パイプライン方式

1. 概要

表 1.1 特長 (2)

項目	仕様
DSP	<p>DSP エンジン</p> <ul style="list-style-type: none"> <li>• 乗算器</li> <li>• 算術演算器 (ALU : Arithmetic Logic Unit)</li> <li>• シフト器</li> <li>• DSP レジスタ</li> </ul> <p>乗算器</p> <ul style="list-style-type: none"> <li>• 16 ビット × 16 ビット 32 ビット</li> <li>• 1 サイクル乗算器</li> </ul> <p>DSP レジスタ</p> <ul style="list-style-type: none"> <li>• 40 ビットデータレジスタ × 2 本</li> <li>• 32 ビットデータレジスタ × 6 本</li> <li>• モジュロレジスタ (MOD、32 ビット) をコントロールレジスタに追加</li> <li>• リピートカウンタ (RC) をステータスレジスタ (SR) に追加</li> <li>• 繰り返し開始レジスタ (RS、32 ビット)、繰り返し終了レジスタ (RE、32 ビット) をコントロールレジスタに追加</li> </ul> <p>DSP データバス</p> <ul style="list-style-type: none"> <li>• 拡張ハーバード型アーキテクチャ</li> <li>• 2 つのデータバスおよび 1 つの命令バスを同時にアクセス</li> </ul> <p>並行処理</p> <ul style="list-style-type: none"> <li>• 最大 4 つの並行処理</li> <li>• ALU 演算、乗算、および 2 つのロードまたはストア</li> </ul> <p>アドレス演算器</p> <ul style="list-style-type: none"> <li>• 2 つのアドレス演算器</li> <li>• 2 つのメモリをアクセスするためのアドレス演算</li> </ul> <p>DSP データアドレッシングモード</p> <ul style="list-style-type: none"> <li>• インクリメントおよびインデクス</li> <li>• それぞれモジュロアドレッシング付きまたは無し</li> </ul> <p>繰り返し制御：ゼロオーバーヘッド繰り返し (ループ) 制御</p> <p>命令セット</p> <ul style="list-style-type: none"> <li>• 16 ビット長 (ロードまたはストアだけの場合)</li> <li>• 32 ビット長 (ALU 演算、乗算を含む場合)</li> <li>• DSP レジスタをアクセスする SuperH マイコン命令を追加</li> </ul> <p>パイプライン 最後の第 5 ステージが DSP ステージ</p>

表 1.1 特長 (3)

項目	仕様
キャッシュ	命令/データ混在型キャッシュ 容量は、最大 4k バイト 4 ウェイセットアソシアティブ方式 ライン長は 16 バイト キャッシュタグは 64 エントリ ライトバックバッファは 16 バイト データ書き込みはライトバック方式/ライトスルー方式の選択可能 LRU リプレースアルゴリズム 2k バイトのキャッシュと 2k バイトの RAM としても使用可能 (2 ウェイキャッシュモード) 命令/データ混在キャッシュ、命令キャッシュ、データキャッシュモード指定可能 リード 1cyc、ライト 2cyc (ライトバック方式時)
割り込み コントローラ (INTC)	16 レベルの優先順位設定が可能 内蔵周辺モジュール割り込みベクタ番号設定可能 内部割り込み要因 41 要因 <ul style="list-style-type: none"> <li>• E-DMAC 割り込み (EINT) は、EtherC と E-DMAC の 22 割り込み要因 (最大) の OR となって、INTC に入力されます。つまり、INTC から見れば、EtherC、E-DMAC の割り込み要因は 1 つとなります。</li> </ul> 外部割り込み端子 × 5 本 (NMI、 $\overline{\text{IRL0}}$ ~ $\overline{\text{IRL3}}$ )、うち $\overline{\text{IRL0}}$ ~ $\overline{\text{IRL3}}$ 端子は 15 外部割り込み要因をエンコード入力も選択可能 (IRL 割り込み) IRL 割り込みのベクタ番号設定選択可能 (オートベクタ、外部ベクタの選択可能) IRQ 割り込みを設定可能 (ローレベル、立ち上がり、立ち下がり、両エッジ検出)
ユーザ ブレークコントローラ (UBC) × 4 チャンネル (A、B、C、D)	チャンネル A、B、C、D について単独条件または、シーケンシャル条件で割り込み発生可能 <ul style="list-style-type: none"> <li>• シーケンシャル設定: A B C D、B C D、C D の 3 パターン</li> </ul> ブレーク条件として、アドレス、データ (チャンネル C、D のみ)、バスマスタ (CPU/DMAC)、バスサイクル (命令フェッチ/データアクセス)、リード/ライト、オペランドサイクル (バイト/ワード/ロングワード) を設定可能 ブレーク条件成立により、ユーザブレーク割り込みを発生 命令フェッチサイクルで命令の実行前に停止するか、実行後に停止するか選択可能 実行回数指定ブレーク (チャンネル C、D のみ) 設定可能実行回数: 最大 $2^{12}-1$ 回 (4095 回) PC トレース機能 分岐命令フェッチ時の分岐元/分岐先アドレスをトレース可能 (最大 4 組 8 アドレス)

1. 概要

表 1.1 特長 (4)

項目	仕様
<p>バスステート コントローラ (BSC)</p>	<p>アドレスエリアを 5 エリアに分岐 (CS0 ~ 4 空間、それぞれ最大リニア 32M バイト)</p> <ul style="list-style-type: none"> <li>• エリアごとに、DRAM、シンクロナス DRAM、バースト ROM 等のメモリ種類を指定</li> <li>• SDRAM 2 エリア空間 (CS2、3)、CS3 は、DRAM もサポート</li> <li>• エリアごとに、バス幅 (8 ビット、16 ビット、32 ビット) を選択可能</li> <li>• エリアごとに、ウェイトステートの挿入制御</li> <li>• エリアごとに、対応した制御信号を出力</li> <li>• CS2、4 は、エンディアン設定可能</li> </ul> <p>キャッシュ</p> <ul style="list-style-type: none"> <li>• キャッシュ領域 / キャッシュスルー領域をアクセスアドレスで選択</li> <li>• 書き込みは、ライトスルー / ライトバック方式選択可能</li> </ul> <p>リフレッシュ機能</p> <ul style="list-style-type: none"> <li>• CAS ビフォ RAS リフレッシュ (オートリフレッシュ) / セルフリフレッシュ</li> <li>• リフレッシュ用カウンタ、クロックセレクトにより、リフレッシュ間隔を設定</li> <li>• リフレッシュ回数設定 (1、2、4、6、8) による集中リフレッシュ</li> <li>• リフレッシュ要求出力可能 (REFOUT)</li> </ul> <p>DRAM 直結インタフェース</p> <ul style="list-style-type: none"> <li>• ロウアドレス / カラムアドレスのマルチプレクス出力</li> <li>• 読み出し時のバースト転送、連続アクセスに対する高速ページモード</li> <li>• EDO モード</li> <li>• RAS プリチャージタイム確保用 TP サイクル発生</li> </ul> <p>シンクロナス DRAM 直結インタフェース</p> <ul style="list-style-type: none"> <li>• ロウアドレス / カラムアドレスのマルチプレクス出力</li> <li>• バンクアクティブモード (CS3 のみ有効)</li> <li>• バースト読み出し / シングル書き込みモード、バースト読み出し / バースト書き込みモード 選択可能</li> </ul> <p>バスアービトレーション (BRLS、BGR)</p> <p>リフレッシュ用カウンタをインターバルタイマとして使用可能</p> <ul style="list-style-type: none"> <li>• コンペアマッチで割り込み要求発生 (CMI 割り込み要求信号)</li> </ul>

表 1.1 特長 (5)

項目	仕様
ダイレクトメモリ アクセスコントローラ (DMAC) × 2 チャンネル	アドレス空間 4G バイト 最大転送回数 16M (16,777,216) 回 転送データ長 : 8 ビット / 16 ビット / 32 ビット / 16 バイトから選択可能 キャッシュヒット時は、CPU の命令処理と DMA 動作が並列実行可能 デュアルアドレス / シングルアドレスモード転送選択可 <ul style="list-style-type: none"> <li>• シングルアドレス (1 バスサイクルで 1 転送単位のデータ転送)</li> <li>• デュアルアドレス (2 バスサイクルで 1 転送単位のデータ転送)</li> <li>• SDRAM 接続時、16 バイト連続リード 連続ライトの転送可能 (デュアル)</li> </ul> SDRAM 接続時、31.25MHz まで 1 クロックシングルアドレス転送可能 サイクルスチールまたはバースト転送可能 チャンネル間優先順位設定可能 (固定モード / ラウンドロビンモード) 下記デバイス間の DMA 転送が可能 <ul style="list-style-type: none"> <li>• 外部メモリ、内蔵メモリ、内蔵周辺モジュール (除く DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC)</li> </ul> 外部リクエスト、内蔵周辺モジュールからの DMA 転送要求、およびオートリクエスト可能 データ転送終了時に CPU へ割り込み要求可能 (DEIn) DREQ サンプリングは、DACK をトリガとする (ただし、最初の DACK までに一回受け付け となるので、必ず一回オーバーラン)
内蔵 RAM	X-RAM : 4k バイト Y-RAM : 4k バイト
イーサネット コントローラ用 ダイレクトメモリ アクセス コントローラ (E-DMAC) × 2 チャンネル	EtherC- 外部メモリ / 内部メモリ間転送可能 16 バイトバースト転送可能 シングルアドレス転送 チェーンブロック転送 転送データ幅 : 32 ビット アドレス空間 4G バイト

1. 概要

表 1.1 特長 (6)

項目	仕様
イーサネット コントローラ (EtherC)	<p>MAC (Media Access Control) 機能</p> <ul style="list-style-type: none"> <li>データフレームの組み立て / 分解 (IEEE802.3 準拠フレーム形式)</li> <li>CSMA/CD 方式のリンク管理 (衝突回避、衝突発生時の処理)</li> <li>CRC 処理</li> <li>FIFO 搭載 (送信、受信それぞれ 512 バイト)</li> <li>全二重受信サポート</li> <li>ショートパケット</li> <li>ロングパケット送受信</li> </ul> <p>MII (Media Independent Interface) 標準規格に対応</p> <ul style="list-style-type: none"> <li>MAC 層からの 8 ビットストリームデータを MII ニブルストリーム (4 ビット) に変換</li> <li>ステーション管理 (STA) 機能</li> <li>18 本の TTL レベル信号</li> <li>転送レート可変: 10/100Mbps</li> </ul> <p>Magic Packet™* (WOL (Wake On LAN) 出力あり)</p>
FIFO 付 シリアル コミュニケーション インタフェース (SCIF) × 2 チャンネル	<p>調歩同期式モード</p> <ul style="list-style-type: none"> <li>データ長: 7 ビット、または 8 ビット</li> <li>ストップビット長: 1 ビット、または 2 ビット</li> <li>パリティ: 偶数パリティ、奇数パリティ、パリティ無し</li> <li>受信エラー検出: パリティエラー、フレーミングエラー、オーバランエラー</li> <li>ブレークの検出</li> </ul> <p>クロック同期式モード</p> <ul style="list-style-type: none"> <li>シリアル通信フォーマットは 1 種類 (データ長: 8 ビット)</li> <li>受信エラー検出: オーバランエラー</li> </ul> <p>IrDA モード (IrDA 1.0 に準拠)</p> <p>送受信を同時に行うことが可能 (全二重)</p> <ul style="list-style-type: none"> <li>ただし、IrDA 通信時には、半二重通信を行います。</li> </ul> <p>内蔵ボーレート・ジェネレータにより任意のビットレートを選択可能</p> <p>送受信に FIFO レジスタを各々 16 段内蔵しており、高速連続通信が可能</p> <p>内部または外部 (SCK) 送受信クロックソース</p> <p>4 種類の割り込み要因</p> <ul style="list-style-type: none"> <li>送信 FIFO データエンプティ</li> <li>ブレーク</li> <li>受信 FIFO データフル</li> <li>受信エラー</li> </ul> <p>内蔵モデムコントロール機能 (RTS、CTS)</p> <p>送信、および受信 FIFO レジスタのデータ数、および受信 FIFO レジスタの受信データのエラー数を検出可能</p> <p>受信中、タイムアウトエラー (DR) を検出可能</p>

【注】 \* Magic Packet™ は、Advanced Micro Devices, Inc. の登録商標です。

表 1.1 特長 (7)

項目	仕様
シリアル I/O (SIO) × 3 チャンネル	全二重動作 (独立した送受信レジスタと独立した送受信クロック) ダブルバッファ構造の送受信ポート (データの連続送受信が可能) インターバル転送モードと連続転送モード 8 ビット / 16 ビットのデータ長を選択可能 ボーリングまたは割り込みによるデータ転送通信 データ I/O に対して MSB 先頭の転送
ハイパフォーマンス ユーザデバッグ インタフェース (H-UDI)	IEEE1149.1 対応 <ul style="list-style-type: none"> <li>• 5 本のテスト信号 (TCK、TDI、TDO、TMS、TRST)</li> <li>• TAP コントローラ</li> <li>• インストラクションレジスタ</li> <li>• データレジスタ</li> <li>• バイパスレジスタ</li> </ul> IEEE1149.1 規格対応のテストモード <ul style="list-style-type: none"> <li>• 標準命令: BYPASS、SAMPLE/PRELOAD、EXTEST</li> <li>• オプション命令: CLAMP、HIGHZ、IDCODE</li> </ul> H-UDI 割り込み <ul style="list-style-type: none"> <li>• INTC への H-UDI 割り込み要求</li> </ul> リセットホールド
タイマパルス ユニット (TPU) × 3 チャンネル	最大 8 本のパルス入出力が可能 チャンネル 0 は 4 本、チャンネル 1、2 は各 2 本、合計 8 本のタイマジェネラルレジスタ (TGR) を内蔵 <ul style="list-style-type: none"> <li>• コンペアマッチによる波形出力: 0 出力、1 出力、トグル出力が選択可能</li> <li>• インพุットキャプチャ機能: 立ち上がりエッジ、立ち下がりエッジ、両エッジ選択可能</li> <li>• カウンタクリア動作: コンペアマッチ、インพุットキャプチャによるカウンタクリア許可</li> <li>• 同期動作: 複数のタイマカウンタ (TCNT) への同時書き込みが可能、コンペアマッチ / インพุットキャプチャによる同時クリア許可、カウンタの同期動作による各レジスタの同期入出力が可能</li> <li>• PWM モード: 任意デューティの PWM 出力が可能、同期動作と組み合わせることにより、最大 7 相の PWM 出力可能</li> </ul> チャンネル 0 はバッファ動作を設定可能 <ul style="list-style-type: none"> <li>• インพุットキャプチャレジスタのダブルバッファ構成が可能</li> <li>• アウトプットコンペアレジスタの自動書き換えが可能</li> </ul> チャンネル 1、2 は各々独立に位相係数モードを設定可能 <ul style="list-style-type: none"> <li>• 2 相エンコーダパルスのアップダウンカウンタが可能</li> </ul> 13 種類の割り込み要因 <ul style="list-style-type: none"> <li>• チャンネル 0 はコンペアマッチ / インพุットキャプチャ兼用割り込み × 4 本、オーバフロー割り込み × 1 本が独立に要求可能</li> <li>• チャンネル 1、2 はコンペアマッチ / インพุットキャプチャ兼用割り込み × 2 本、オーバフロー割り込み × 1 本、アンダフロー割り込み × 1 本が独立に要求可能</li> </ul>

1. 概要

表 1.1 特長 (8)

項目	仕様
16 ビット フリーランニング タイマ (FRT) × 1 チャンネル	4 種類のカウンタ入力クロックを選択可能 <ul style="list-style-type: none"> <li>• 3 種類の内部クロック (P /8、P /32、P /128)</li> <li>• 外部クロック (外部イベントのカウントが可能)</li> </ul> 2 本の独立したコンパレータ (2 種類の波形出力可能) インพุットキャプチャ (立ち上がりエッジ / 立ち下がりエッジ選択可能) カウンタのクリア指定が可能 <ul style="list-style-type: none"> <li>• コンペアマッチ A により、カウンタの値をクリアすることが可能</li> </ul> 4 種類の割り込み要因 <ul style="list-style-type: none"> <li>• コンペアマッチ × 2 要因 (OCIA、OCIB)</li> <li>• インพุットキャプチャ × 1 要因 (ICI)</li> <li>• オーバフロー × 1 要因 (OVI)</li> </ul>
ウォッチ ドッグタイマ (WDT) × 1 チャンネル	ウォッチドッグタイマモード / インターバルタイマモードの切り替えが可能 カウントオーバーフロー時、内部リセット、外部信号 ( $\overline{\text{WDTOVF}}$ ) または割り込みを発生 スタンバイモードの解除時、クロック周波数変更時、クロックポーズモード時に使用 8 種類のカウンタ入力クロックを選択可能
クロック発振器 (CPG)	クロック発振器内蔵 クロックソースは水晶または外部クロックから選択可能 クロック選信用 PLL 回路内蔵 外部クロック / 内部クロック間の位相同期化用 PLL 回路内蔵 CPU/DSP コア用クロック (I)、周辺モジュール用クロック (P)、外部インタフェース用 クロック (E) の周波数を各々独立に分周可能
システム コントローラ (SYSC)	7 つの動作モード、3 つの低消費電力モードを設定 動作モード <ul style="list-style-type: none"> <li>• クロック生成方法 (PLL ON/OFF)、クロック分周比を制御</li> </ul> 低消費電力モード <ul style="list-style-type: none"> <li>• スリープモード : CPU 機能を停止</li> <li>• スタンバイモード : 全ての機能を停止</li> <li>• モジュールスタンバイ機能 : FRT、SCIF、DMAC、UBC、DSP、TPU、SIO 内蔵周辺 モジュールの動作を選択的に停止する</li> </ul>
I/O ポート	入出力 : 29 本

1.2 ブロック図

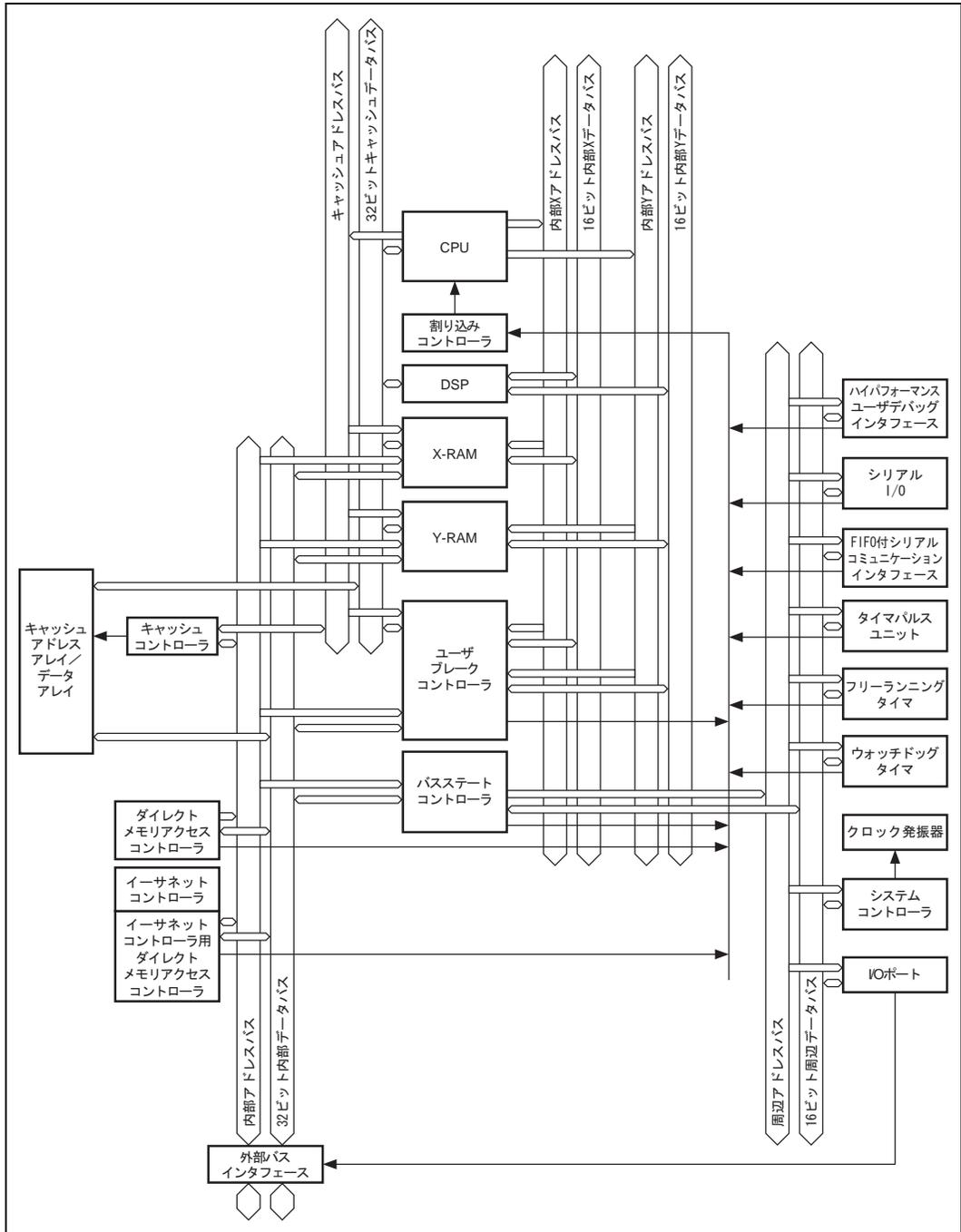


図 1.1 SH7615 ブロック図

1. 概要

1.3 端子の説明

1.3.1 ピン配置

HD6417615ARF/ARFV のピン配置図を図 1.2 に HD6417615ARB/ARBV のピン配置図を図 1.3 に示します。

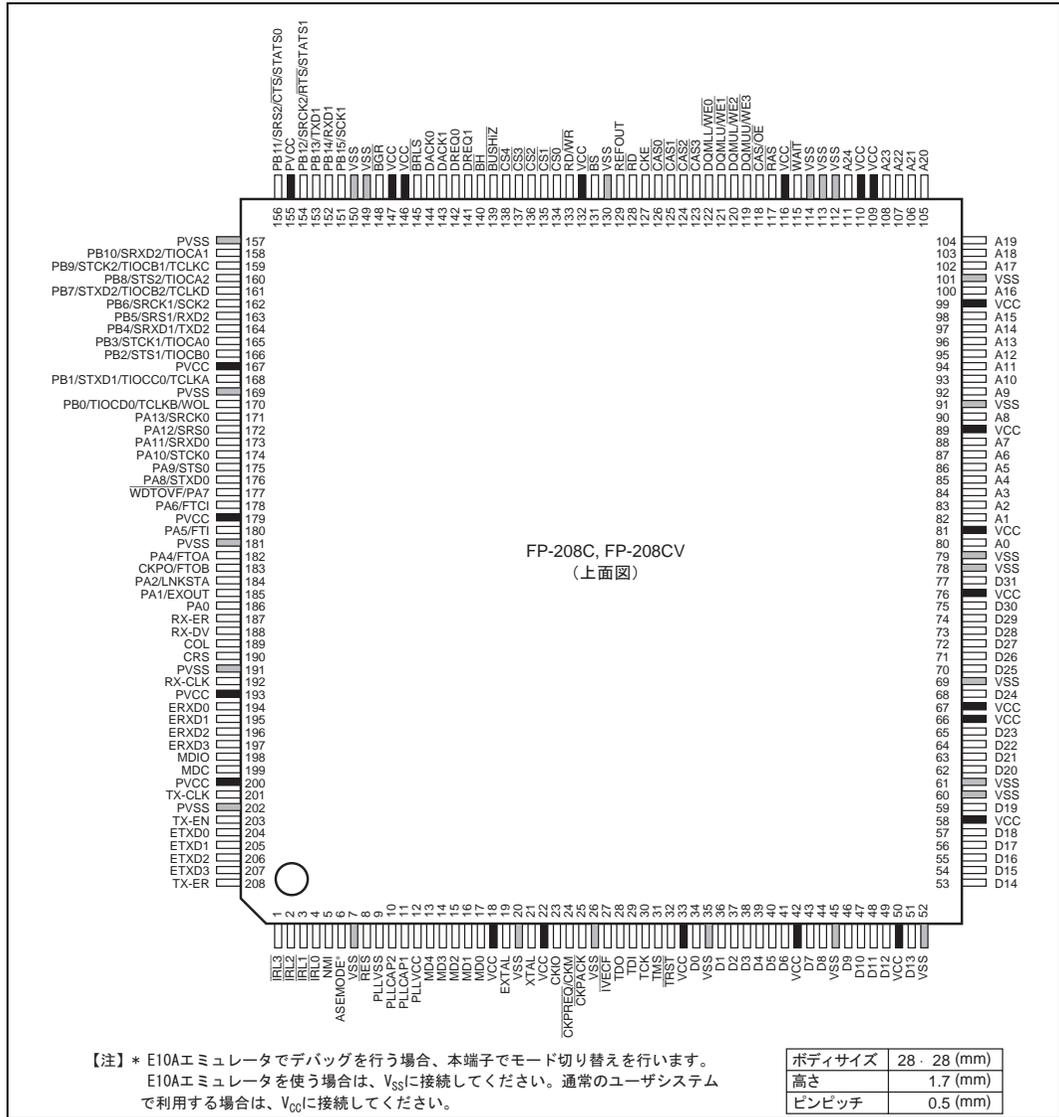


図 1.2 HD6417615ARF/ARFV ピン配置図 (FP-208C, FP-208CV)

	X'																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19		
Y	A	NC	ETXD3	EXTD1	TX-EN	TX-CLK	MDC	ERXD1	RX-CLK	RX-DV	PA1/ EXOUT	PVCC	PA8/ ST50	PA12/ SR50	PB1/STXD1/ TIOCC0/ TCLKA	PB3/ STCK1/ TIOCA0	PB5/ SR51/ RXD2	PB7/STXD2/ TIOCB2/ TCLKD	PB8/STCK2/ TIOCB1/ TCLKG	NC	A
	B	IRL1	IRL3	ETXD2	TX-ER	ETXD0	MDIO	ERXD0	PVSS	RX-ER	CKPO/ FTOB	PA5/FTI	PA8/ STXD0	PA11/ SRXD0	PVSS	PB6/ SRCK1/ SCK2	PB10/ SRXD2/ TIOCA1	PB8/ST52/ TIOCA2	PVSS	PVCC	B
	C	NMI	IRL0	NC	PVSS	PVCC	ERXD3	PVCC	CRS	PA0	PA2/ LNKSTA	PA4/FTDA	WDTOVF/ PA7	PA10/ STCK0	PB0/ TIOCD0/ TCLKB/WCL	PVCC	PB4/ SRXD1/ TXD2	PB2/ST51/ TIOCB0	PB12/ SRCK2/ RTS/STAT3	PB13/ TXD1	C
	D	VSS	IRL2	RES	NC	NC	ERXD2	NC	COL	NC	NC	PVSS	PA6/FTCI	NC	PA13/ SRCK0	NC	NC	NC	PB11/ SR52/ CTS/STAT30	PB15/ SCK1	D
	E	PLLSS	ASEMODE	NC	NC	BP-240A, BP-240AV Bump No. (Top View)										NC	VSS	PB14/ RXD1	VSS	E	
	F	PLLAP2	PLLAP1	NC	PLLCC											NC	VCC	BGR	F		
	G	MD4	MD3	MD2	MD1											DREQ0	DACK1	DACK0	BRL3	G	
	H	MD0	VCC	EXTAL	VSS											CS4	BUSHZ	BH	DREQ1	H	
	J	XTAL	VCC	NC	NC											CS1	CS0	CS2	CS3	J	
	K	VSS	CKIO	CKPREQ/ CKM	CKPACK											NC	VCC	B5	RDWR	K	
	L	TCK	TDI	IVECF	TDO											REFOUT	VSS	RD	CKE	L	
	M	D0	VCC	TRST	TMS											CAS0	CAS1	CAS2	CAS3	M	
	N	D3	D2	D1	VSS											DMUL/ WE0	DMUL/ WE1	DMUL/ WE2	DMUL/ WE3	N	
	P	VCC	D6	D5	D4											NC	NC	CAS/OE	RAS	P	
	R	D8	D10	D7	NC	VCC	WAF	VCC	VSS	R											
	T	D9	D13	VSS	NC	NC	VCC	NC	D29	NC	A0	NC	A4	NC	A9	NC	NC	VSS	A21	A24	T
	U	D11	D12	D20	VSS	D21	VCC	D25	D28	D31	VSS	A1	A5	VCC	A10	A13	A15	VSS	A23	VCC	U
	V	VCC	D14	D17	D15	VCC	D23	VSS	D27	VCC	VSS	A2	A6	A8	A11	A16	A19	A17	A20	A22	V
	W	VSS	D16	D18	D19	VSS	D22	D24	D26	D30	VCC	A3	A7	VSS	A12	A14	VCC	VSS	A18	NC	W
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	

図 1.3 HD6417615ARBP/ARBPV ピン配置図 (BP-240A, BP-240AV)

## 1. 概要

### 1.3.2 端子の機能

表 1.2 端子の機能 (1)

分類	記号	入出力	名称	機能
電源	V <sub>CC</sub>	入力	電源	電源に接続します。全ての V <sub>CC</sub> 端子はシステムの電源に接続してください。開放端子があると動作しません。
	V <sub>SS</sub>	入力	グランド	グランドに接続します。全ての V <sub>SS</sub> 端子はシステムのグランドに接続してください。開放端子があると動作しません。
	PV <sub>CC</sub>	入力	入出力回路用電源	入出力回路用の電源です。
	PV <sub>SS</sub>	入力	入出力回路用グランド	入出力回路用のグランドです。
クロック	XTAL	出力	クリスタル入出力端子	水晶発振子を接続します。
	EXTAL	入力		水晶発振子を接続します。また外部クロック入力端子として使用します。
	CKIO	入出力	システムクロック入出力	外部クロック入力、または内部クロック出力端子として使用します。
	CKPREQ/ CKM	入力	クロックポーズリクエスト入力	CKIO 端子から入力されるクロックの周波数変更、またはクロック停止を行うため、クロックポーズ要求端子として使用します。
	CKPACK	出力	クロックポーズアクノリッジ出力	LSI 内部がクロックポーズ状態 (スタンバイ状態) であることを示します。
	CKPO	出力	内蔵周辺クロック (P ) 出力	内蔵周辺クロック (P ) を出力します。
	PLL <sub>CAP1</sub>	入力	PLL 用容量接続端子	PLL 回路 1 動作用の容量を接続します。
	PLL <sub>CAP2</sub>	入力		PLL 回路 2 動作用の容量を接続します。
	PLL <sub>V<sub>CC</sub></sub>	入力	PLL 用電源	PLL 発振器用の電源
PLL <sub>V<sub>SS</sub></sub>	入力	PLL 用グランド	PLL 発振器用のグランド	
システム制御	RES	入力	リセット	RES = 0、NMI = 1 のとき、パワーオンリセット状態になります。RES = 0、NMI = 0 のとき、マニュアルリセット状態になります。
	WDTOVF	出力	ウォッチドッグタイマオーバーフロー	ウォッチドッグタイマモード時のカウンタオーバーフロー信号出力
	BGR	出力	バス権許可	バス権を外部デバイスに開放したことを示します。BRLS 信号を出力したデバイスは BGR 信号を受けてバス権を獲得したことを知らずことができます。
	BRLS	入力	バス権要求	外部デバイスがバス権の開放を要求するときにローレベルにします。
動作モード	MD0 ~ MD4	入力	モード設定	この端子に印加するレベルで動作モードを指定
割り込み	NMI	入力	ノンマスクابل割り込み	マスク不可能な割り込み要求信号を入力
	IRL3 ~ IRL0	入力	外部割り込み要因入力 0 ~ 3	マスク可能な割り込み要求信号を入力
	IVECF	出力	割り込みベクタフェッチサイクル	外部ベクタリードサイクルであることを示します。

表 1.2 端子の機能 (2)

分類	記号	入出力	名称	機能
バス制御	$\overline{BS}$	出力	バスサイクル開始	バスサイクルの開始を示す信号 バースト転送時は毎データサイクルごとにアサート
	$\overline{CS4} \sim \overline{CS0}$	出力	チップセレクト 0 ~ 4	アクセス中のエリアを示すチップセレクト信号
	$\overline{WAIT}$	入力	ウェイト	ウェイトステート要求信号
	$\overline{RD}$	出力	リード	リードサイクルを示すストロープ信号
	$\overline{RAS}$	出力	ロウアドレス ストロープ	DRAM、SDRAM 用 RAS 信号
	$\overline{CAS}$	出力	カラムアドレス ストロープ	SDRAM 用 CAS 信号
	$\overline{OE}$	出力	アウトプット イネーブル	EDO DRAM 用アウトプットイネーブル信号 RAS ダウンモードでアクセス時に使用
	$\overline{DQMUU/WE3}$	出力	最上位バイト アクセス	SRAM、SDRAM 用最上位バイト選択信号
	$\overline{DQMUL/WE2}$	出力	第 2 バイト アクセス	SRAM、SDRAM 用 2 バイト目選択信号
	$\overline{DQMLU/WE1}$	出力	第 3 バイト アクセス	SRAM、SDRAM 用 3 バイト目選択信号
	$\overline{DQMLL/WE0}$	出力	最下位バイト アクセス	SRAM、SDRAM 用最下位バイト選択信号
	$\overline{CAS3}$	出力	カラムアドレス ストロープ 3	DRAM 用最上位バイト選択信号
	$\overline{CAS2}$	出力	カラムアドレス ストロープ 2	DRAM 用 2 バイト目選択信号
	$\overline{CAS1}$	出力	カラムアドレス ストロープ 1	DRAM 用 3 バイト目選択信号
	$\overline{CAS0}$	出力	カラムアドレス ストロープ 0	DRAM 用最下位バイト選択信号
	$\overline{CKE}$	出力	クロック イネーブル	SDRAM 用クロックイネーブル制御
	$\overline{REFOUT}$	出力	リフレッシュ アウト	バス権解放時リフレッシュ実行要求出力
	$\overline{RD/WR}$	出力	リード/ライト	DRAM / シンクロナス DRAM の書き込み信号
	$\overline{BUSHIZ}$	入力	バスハイインピーダ ンス	WAIT 信号と組み合わせて、バスサイクルを終了せ ずにバスおよびストロープ信号を HiZ にする信号
	$\overline{BH}$	出力	バーストヒント	DMA バーストスタートでアサート、バースト終了 の 1 バスサイクル手前でネゲート
$\overline{STATS0},$ $\overline{STATS1}$	出力	ステータス	CPU、DMAC、E-DMAC のステータス情報	
A24 ~ A0	出力	アドレスバス	アドレス出力	
D31 ~ D0	入出力	データバス	データ入出力	

## 1. 概要

---

表 1.2 端子の機能 (2)

分類	記号	入出力	名称	機能
H-UDI	TCK	入力	テストクロック	テストクロック入力
	TMS	入力	テストモード セレクト	テストモードセレクト入力信号
	TDI	入力	テストデータ入力	シリアルデータ入力
	TDO	出力	テストデータ出力	シリアルデータ出力
	$\overline{\text{TRST}}$	入力	テストリセット	テストリセット入力信号
	ASEMODE *	入力	ASE モード入力	ASE モード / ユーザモードセレクト信号

【注】 \* E10A エミュレータでデバッグを行う場合、本端子でモード切り替えを行います。E10A エミュレータを使う場合 (ASE モード) は、 $V_{SS}$  に接続してください。E10A エミュレータを使わずに通常のユーザシステムで本チップを使用する場合 (ユーザモード) は、 $V_{CC}$  に接続してください。また、H-UDI にてバウンダリスキャンテストを行う場合は、必ずユーザモードを使用してください。ASE モードでバウンダリスキャンテストを行うことはできません。

表 1.2 端子の機能 (3)

分類	記号	入出力	名称	機能
イーサネット コントローラ (EtherC)	TX-CLK	入力	送信クロック	TX-EN、ETXD0 ~ 3、TX-ER のタイミング参照信号
	RX-CLK	入力	受信クロック	RX-DV、ERXD0 ~ 3、RX-ER のタイミング参照信号
	TX-EN	出力	送信イネーブル	ETXD0 ~ 3 上に送信データが準備できたことを示す信号
	ETXD0 ~ ETXD3	出力	送信データ 0 ~ 3	4 ビットの送信データ
	TX-ER	出力	送信エラー	エラー状態を他のポートに伝搬させる信号
	RX-DV	入力	受信データ有効	ERXD0 ~ 3 上に有効な受信データがあることを示す信号
	ERXD0 ~ ERXD3	入力	受信データ 0 ~ 3	4 ビットの受信データ
	RX-ER	入力	受信エラー	フレームデータを転送中に発生したエラー状態を通知
	CRS	入力	キャリア検出	キャリア検出通知信号
	COL	入力	衝突検出	衝突検出信号
	MDC	出力	管理用データ クロック	MDIO による情報転送用の参照クロック信号
	MDIO	入出力	管理用データ 入出力	STA と PHY との間で管理情報を交換するための双方向 信号
	LNKSTA	入力	リンクステータス	PHY からのリンク状態入力
	EXOUT	出力	汎用外部出力	汎用外部出力端子
WOL	出力	ウェイクオンラン	Magic Packet を検出したことを示す信号	
ダイレクト メモリアクセス コントローラ (DMAC)	DACK0, DACK 1	出力	DMAC チャンネル 0、1 アクノリッジ	外部デバイスへ DMA 転送要求の受け付けを出力します。
	DREQ0, DREQ 1	入力	DMAC チャンネル 0、1 リクエスト	外部デバイスからの DMA 転送要求の入力端子です。
FIFO 付 シリアルコミュ ニケーションイ ンタフェース (SCIF)	TXD1, TXD2	出力	送信データ 出力チャンネル 1、2	SCIF チャンネル 1、2 の送信データ出力端子です。
	RXD1, RXD2	入力	受信データ 入力チャンネル 1、2	SCIF チャンネル 1、2 の受信データ入力端子です。
	SCK1, SCK2	入出力	シリアルクロック入 出力チャンネル 1、2	SCIF のクロック入出力端子です。
	RTS	出力	送信要求	SCIF チャンネル 1 の送信要求出力端子
	CTS	入力	送信許可	SCIF チャンネル 1 の送信許可入力端子

1. 概要

表 1.2 端子の機能 (3)

分類	記号	入出力	名称	機能
タイマパルス ユニット (TPU)	TCLKA TCLKB TCLKC TCLKD	入力	TPU タイマ クロック入力 A、B、C、D	TPU カウンタへの外部クロック入力端子です。
	TIOCA0 TIOCB0 TIOCC0 TIOCD0	入出力	TPU インพุット キャブチャ / アウ ツットコンペア (チャンネル 0)	チャンネル 0 の インพุットキャブチャ入力 / アウツットコンペア出力 / PWM 出力端子です。
	TIOCA1 TIOCB1	入出力	TPU インพุット キャブチャ / アウ ツットコンペア (チャンネル 1)	チャンネル 1 の インพุットキャブチャ入力 / アウツットコンペア出力 / PWM 出力端子です。
	TIOCA2 TIOCB2	入出力	TPU インพุット キャブチャ / アウ ツットコンペア (チャンネル 2)	チャンネル 2 の インพุットキャブチャ入力 / アウツットコンペア出力 / PWM 出力端子です。

表 1.2 端子の機能 (4)

分類	記号	入出力	名称	機能
16ビットフリーランニングタイマ (FRT)	FTCI	入力	カウンタクロック入力	FRCのカウンタクロックの入力端子
	FTOA	出力	アウトプットコンペア A 出力	アウトプットコンペア A 出力端子
	FTOB	出力	アウトプットコンペア B 出力	アウトプットコンペア B 出力端子
	FTI	入力	インプットキャプチャ入力	インプットキャプチャ入力端子
シリアル I/O (SIO)	SRXD0 ~ SRXD2	入力	シリアル受信データ入力 0 ~ 2	シリアルデータ入力ポート
	SRCK0 ~ SRCK2	入力	シリアル受信クロック入力 0 ~ 2	シリアル受信クロックポート
	SRS0 ~ SRS2	入力	シリアル受信同期入力 0 ~ 2	シリアル受信同期入力ポート
	STXD0 ~ STXD2	出力	シリアル送信データ出力 0 ~ 2	シリアルデータ出力ポート
	STCK0 ~ STCK2	入力	シリアル送信クロック入力 0 ~ 2	シリアル送信クロックポート
	STS0 ~ STS 2	入出力	シリアル送信同期入出力 0 ~ 2	シリアル送信同期入出力ポート
I/O ポート	PA0 ~ PA13*	入出力	汎用ポート	汎用入出力ポート端子です。 1ビットごとに入出力を指定できます。
	PB0 ~ PB15	入出力	汎用ポート	汎用入出力ポート端子です。 1ビットごとに入出力を指定できます。

【注】 \* PA3 は、使用不可。代わりに CKPO が有効。

1. 概要

---

1.3.3 端子マルチプレクスの説明

表 1.3 端子マルチプレクスの説明 (1)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1	機能 2	機能 3	機能 4	分類
F4	12	PLL <sub>VCC</sub>				クロック系
E1	9	PLL <sub>VSS</sub>				
F2	11	PLLCAP1				
F1	10	PLLCAP2				
H3	19	EXTAL				
J1	21	XTAL				
K2	23	CKIO				
K3	24	CKPREQ/CKM				
K4	25	CKPACK				
D3	8	RES				9 端子
G1	13	MD4				システム コントロール
G2	14	MD3				
G3	15	MD2				
G4	16	MD1				
H1	17	MD0				
C1	5	NMI				
B2	1	IRL3				割り込み
D2	2	IRL2				
B1	3	IRL1				
C2	4	IRL0				
L3	27	IVECF				
						6 端子

表 1.3 端子マルチプレクスの説明(1)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1	機能 2	機能 3	機能 4	分類
K18	131	BS				バスコントロール
H16	138	CS4				
J19	137	CS3				
J18	136	CS2				
J16	135	CS1				
J17	134	CS0				
F19	148	BGR				
G19	145	BRLS				
R17	115	WAIT				
L18	128	RD				
P19	117	RAS				
P18	118	CAS/OE				
N19	119	DQMUU/WE3				
N18	120	DQMUL/WE2				
N17	121	DQMLU/WE1				
N16	122	DQMLL/WE0				25 端子
M19	123	CAS3				
M18	124	CAS2				
M17	125	CAS1				
M16	126	CAS0				
L19	127	CKE				
L16	129	REFOUT				
K19	133	RD/WR				
H17	139	BUSHIZ				
H18	140	BH				

1. 概要

表 1.3 端子マルチプレクスの説明 (2)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1	機能 2	機能 3	機能 4	分類
T19	111	A24				アドレスバス
U18	108	A23				
V19	107	A22				
T18	106	A21				
V18	105	A20				
V16	104	A19				
W18	103	A18				
V17	102	A17				
V15	100	A16				
U16	98	A15				
W15	97	A14				
U15	96	A13				
W14	95	A12				
V14	94	A11				
U14	93	A10				
T14	92	A9				
V13	90	A8				
W12	88	A7				
V12	87	A6				
U12	86	A5				
T12	85	A4				
W11	84	A3				
V11	83	A2				
U11	82	A1				
T10	80	A0				25 端子

表 1.3 端子マルチプレクスの説明 (3)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1	機能 2	機能 3	機能 4	分類
U9	77	D31				データバス
W9	75	D30				
T8	74	D29				
U8	73	D28				
V8	72	D27				
W8	71	D26				
U7	70	D25				
W7	68	D24				
V6	65	D23				
W6	64	D22				
U5	63	D21				
U3	62	D20				
W4	59	D19				
W3	57	D18				
V3	56	D17				
W2	55	D16				
V4	54	D15				
V2	53	D14				
T2	51	D13				
U2	49	D12				
U1	48	D11				
R2	47	D10				
T1	46	D9				
R1	44	D8				
R3	43	D7				
P2	41	D6				
P3	40	D5				
P4	39	D4				
N1	38	D3				
N2	37	D2				
N3	36	D1				
M1	34	D0				

32 端子

1. 概要

表 1.3 端子マルチプレクスの説明 (4)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1	機能 2	機能 3	機能 4	分類
L1	30	TCK				H-UDI
M4	31	TMS				
L2	29	TDI				
L4	28	TDO				
M3	32	TRST				
E2	6	ASEMODE *				6 端子
A5	201	TX-CLK				EtherC
A8	192	RX-CLK				
A4	203	TX-EN				5V I/O 対応
A2	207	ETXD3				
B3	206	ETXD2				
A3	205	ETXD1				
B5	204	ETXD0				
B4	208	TX-ER				
A9	188	RX-DV				
C6	197	ERXD3				
D6	196	ERXD2				
A7	195	ERXD1				
B7	194	ERXD0				
B9	187	RX-ER				
C8	190	CRS				
D8	189	COL				
A6	199	MDC				
B6	198	MDIO				18 端子
G17	143	DACK1				DMAC
G18	144	DACK0				
H19	141	DREQ1				
G16	142	DREQ0				

【注】 \* E10A エミュレータでデバッグを行う場合、本端子でモード切り替えを行います。E10A エミュレータを使う場合 (ASE モード) は、 $V_{SS}$  に接続してください。E10A エミュレータを使わずに通常のユーザシステムで本チップを使用する場合 (ユーザモード) は、 $V_{CC}$  に接続してください。また、H-UDI にてバウンダリスキャンテストを行う場合は、必ずユーザモードを使用してください。ASE モードでバウンダリスキャンテストを行うことはできません。

表 1.3 端子マルチプレクスの説明 (5)

BP-240A/AV 番号	FP-208C/CV 番号	機能 1 [ 00 ] *	機能 2 [ 01 ] *	機能 3 [ 10 ] *	機能 4 [ 11 ] *	分類
D19	151	PB15		SCK1		ポート B SCIF, SIO, TPU 5V I/O 対応
E18	152	PB14		RXD1		
C19	153	PB13		TXD1		
C18	154	PB12	SRCK2	$\overline{RTS}$	STATS1	
D18	156	PB11	SRS2	$\overline{CTS}$	STATS0	
B16	158	PB10	SRXD2	TIOCA1		
A18	159	PB9	STCK2	TIOCB1/TCLKC		
B17	160	PB8	STS2	TIOCA2		
A17	161	PB7	STXD2	TIOCB2/TCLKD		
B15	162	PB6	SRCK1	SCK2		
A16	163	PB5	SRS1	RXD2		
C16	164	PB4	SRXD1	TXD2		
A15	165	PB3	STCK1	TIOCA0		
C17	166	PB2	STS1	TIOCB0		
A14	168	PB1	STXD1	TIOCC0/TCLKA		
C14	170	PB0		TIOCD0/TCLKB	WOL	
D14	171	PA13	SRCK0			ポート A, SIO, FRT 5V I/O 対応
A13	172	PA12	SRS0			
B13	173	PA11	SRXD0			
C13	174	PA10	STCK0			
A12	175	PA9	STS0			
B12	176	PA8	STXD0			
C12	177	WDTOVF	PA7			
D12	178	PA6	FTCI			
B11	180	PA5	FTI			
C11	182	PA4	FTOA			
B10	183	CKPO	FTOB			
C10	184	PA2	LNKSTA			
A10	185	PA1	EXOUT			
C9	186	PA0				

【注】 \* [ ] 内の数字は、ポート A [ 0:13 ] およびポート B [ 0:15 ] において、マルチプレクス機能を選択するための PFC における各モードビット (MD1, MD0) の設定を表しています。

WDTOVF : 本端子はリセット時は、出力端子となります。

汎用入出力として使用する場合は、端子の極性に注意してください。

# 1. 概要

## 1.4 処理状態

### 状態遷移

CPUの処理状態には、リセット状態、例外処理状態、バス権解放状態、プログラム実行状態、低消費電力状態の5種類あります。状態間の遷移を図1.3に示します。

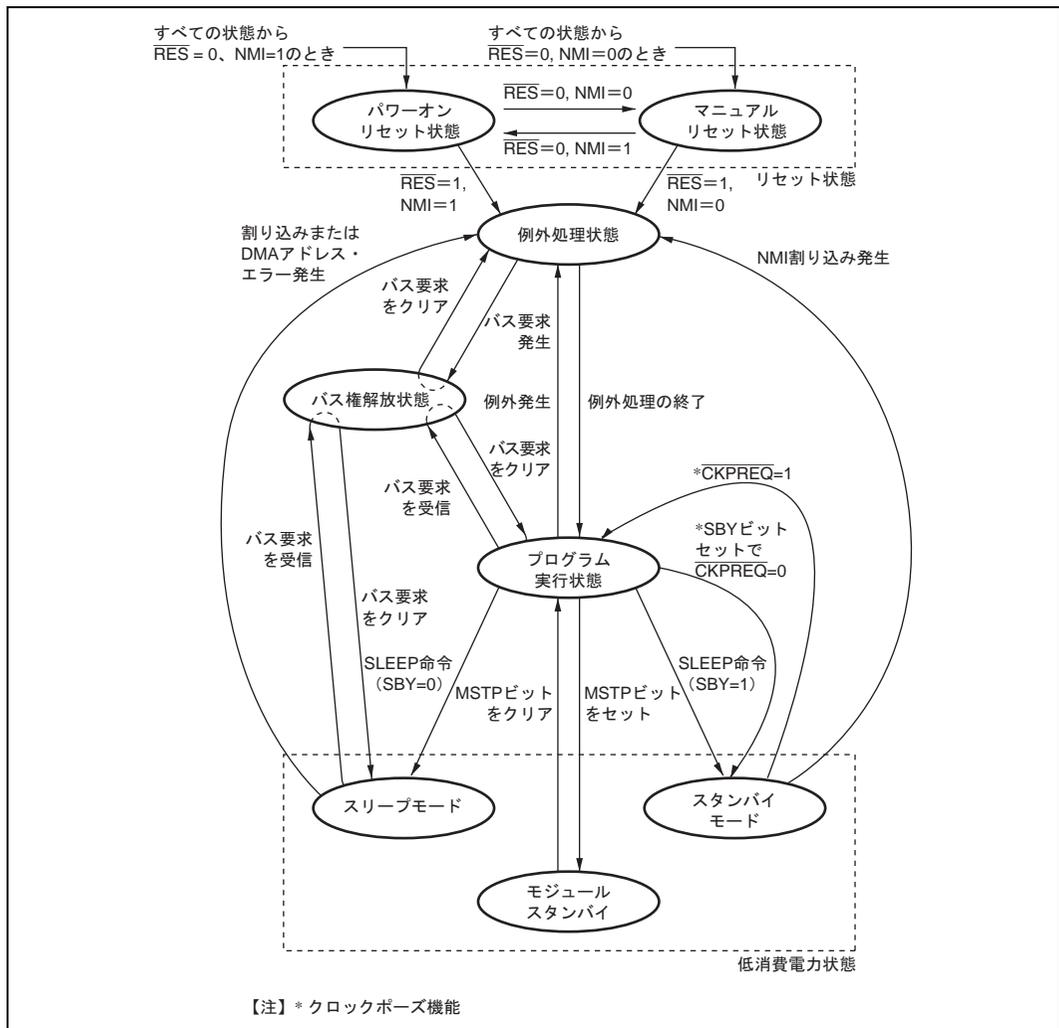


図 1.4 処理状態の状態遷移図

### (1) リセット状態

CPU がリセットされている状態です。 $\overline{\text{RES}}$  端子がローレベルになるとリセット状態になります。NMI 端子がハイレベルのときパワーオンリセット状態になり、NMI 端子がローレベルのとき、マニュアルリセット状態になります。

### (2) 例外処理状態

リセットや割り込みなどの例外処理要因によって、CPU が処理状態の流れを変えるとときの過渡的な状態です。

リセットの場合は、例外処理ベクタテーブルからプログラムカウンタ (PC) の初期値としての実行開始アドレスとスタックポインタ (SP) の初期値を取り出しそれぞれ格納し、スタートアドレスに分岐してプログラムの実行を開始します。

割り込みなどの場合は、SP を参照して、PC とステータスレジスタ (SR) をスタック領域に退避します。例外処理ベクタテーブルから例外サービスルーチンの開始アドレスを取り出し、そのアドレスに分岐してプログラムの実行を開始します。

その後処理状態はプログラム実行状態となります。

### (3) プログラム実行状態

CPU が順次プログラムを実行している状態です。

### (4) 低消費電力

CPU の動作が停止し電力が低い状態です。SLEEP 命令で低消費電力状態になります。スリープモード、スタンバイモードの2つのモードおよびモジュールスタンバイ機能があります。

### (5) バス権解放状態

(i) CPU がバス権を要求したデバイスにバスを解放する状態です。

(ii) マニュアルリセット信号アサート中のバス権解放

マニュアルリセット信号アサート中 ( $\overline{\text{RES}}=\text{Low}$ 、 $\text{NMI}=\text{Low}$ ) は、アービトレーション要求 ( $\overline{\text{BRLS}}$  入力) を受け付けません。 $\overline{\text{BRLS}}$  信号が引き続きアサートされている場合には、そのままバス解放状態を保ちます ( $\overline{\text{BGR}}$  信号をアサート)。

マニュアルリセット信号アサート中、バス解放状態から  $\overline{\text{BRLS}}$  信号がネゲートされた場合には、本 LSI はバスの使用を再開します ( $\overline{\text{BGR}}$  信号をネゲート)。

#### 低消費電力状態

CPU の処理状態の一つとして、通常のプログラム実行状態のほかに、CPU の動作を停止し、消費電力を低くする低消費電力状態には、スリープモード、スタンバイモードの2つのモードおよびモジュールスタンバイ機能があります。

#### (1) スリープモード

スタンバイコントロールレジスタ 1 (SBYCR1) のスタンバイビット (SBY) を 0 にクリアして、SLEEP 命令を実行すると、CPU はスリープモードになります。スリープモードでは、CPU の動作は停止しますが、CPU の内部レジスタのデータと、内蔵キャッシュメモリおよび内蔵 RAM のデータは保持されます。CPU 以外の内蔵周辺モジュールの機能は停止しません。

## 1. 概要

---

スリープモードからの復帰は、リセット、すべての割り込み、または DMA アドレスエラーによって行われ、例外処理状態を経て通常のプログラム実行状態へ遷移します。

### (2) スタンバイモード

SBYCR1 の SBY を 1 にセットして、SLEEP 命令を実行すると、スタンバイモードになります。スタンバイモードでは、CPU、内蔵モジュール、および発振器のすべての機能が停止します。

ただしスタンバイモードに入るときには、DMAC の DMA マスタイネブールビットを 0 にしてください。またスタンバイモードに入る前にキャッシュはオフにしてください。スタンバイモード時には、キャッシュおよび内蔵 RAM のデータは保持されません。

スタンバイモードからの復帰は、リセット、外部の NMI 割り込みにより行われます。スタンバイモードからの復帰は、発振安定時間経過後、例外処理状態を経て通常プログラム実行状態へ遷移します。

またクロックポーズ機能を使用してスタンバイモードへ遷移すると、CKIO 端子入力クロックの周波数を変更したりクロックそのものを停止させたりすることができます。SBYCR1 の SBY を 1 にセットして  $\overline{\text{CKPREQ/CKM}}$  端子にローレベルを印加すると、スタンバイモードに遷移し、 $\overline{\text{CKPACK}}$  端子からローレベルが出力されます。このときクロックの停止、または周波数の変更を行うことができます。

各内蔵周辺モジュールの状態および端子状態は SLEEP 命令による通常のスタンバイモードと同じ状態です。プログラム実行状態へ遷移するためには  $\overline{\text{CKPREQ/CKM}}$  端子にハイレベルを印加します。

本モードでは、発振器が停止しますので、消費電力は著しく低減します。

### (3) モジュールスタンバイ機能

内蔵周辺モジュールのダイレクトメモリアクセスコントローラ (DMAC)、DSP、16 ビットフリーランニングタイマ (FRT)、FIFO 付シリアルコミュニケーションインタフェース (SCIF)、シリアル I/O (SIO)、ユーザブ레이크コントローラ (UBC)、タイマパルスユニット (TPU) には、モジュールスタンバイ機能があります。イーサネットコントローラ (EtherC)、イーサネット用ダイレクトメモリアクセスコントローラ (E-DMAC) では、モジュールスタンバイ機能をサポートしていません。

モジュールスタンバイ機能を使用する場合、スタンバイコントロールレジスタ (SBYCR1/2) のモジュールストップビット 11 ~ 3、1 (MSTP11 ~ MSTP3、MSTP1) に 1 をセットすることで、それぞれ対応した内蔵周辺モジュールへのクロック供給を停止させることができます。この機能を使用することで、消費電力を減少させることができます。

モジュールスタンバイ機能の解除は、該当する MSTP ビットを 0 にクリアすることで行います。

DSP をモジュールスタンバイ状態にした場合、DSP 命令を実行しないでください。

DMAC のモジュールスタンバイ機能を使用する時は、ダイレクトメモリアクセスコントローラの DMA マスタイネブールビットを 0 にしてください。

表 1.4 低消費電力状態

モード	遷移の条件	状態					解除方法
		クロック	CPU	内蔵周辺 モジュール	CPU レジスタ	内蔵キャッ シュまたは 内蔵 RAM	
スリープ モード	SBYCR1 の SBY をクリアした状態 で、SLEEP 命令を実行	動作	停止	動作	保持	保持	(1) 割り込み (2) DMA アドレスエラー (3) パワーオンリセット (4) マニュアルリセット
スタンバイ モード	SBYCR1 の SBY をセットした状態 で、SLEEP 命令を実行	停止	停止	停止および 初期化 <sup>*1</sup>	保持	不定	(1) NMI 割り込み (2) パワーオンリセット (3) マニュアルリセット
モジュール スタンバイ 機能	各モジュールに該当する MSTP ビットを セット	動作	動作 (DSP は 停止)	指定したモ ジュールへの クロック 供給を停止 および初期 化 <sup>*2</sup>	保持	保持	(1) MSTP ビットをクリア (2) パワーオンリセット (3) マニュアルリセット

【注】 \*1 それぞれの周辺モジュール、端子によって異なります。

\*2 DMAC、DSP の各レジスタと指定のモジュールの割り込みベクタは設定値が保持されます。



---

## 2. CPU

---

### 2.1 レジスタ構成

本 LSI には汎用レジスタ (32 ビット× 16 本)、コントロールレジスタ (32 ビット× 6 本)、システムレジスタ (32 ビット× 10 本) があります。

本 LSI は SH-1、SH-2 とオブジェクトコードレベルで上位互換性があります。そのため、従来の SuperH マイコンのレジスタのほかに、いくつかのレジスタが追加されています。追加されたのは、コントロールレジスタの繰り返し開始レジスタ (RS)、繰り返し終了レジスタ (RE)、モジュロレジスタ (MOD) の 3 本と、システムレジスタの DSP ステータスレジスタ (DSR)、DSP データレジスタの内の A0、A1、X0、X1、Y0、Y1 の 6 本です。

汎用レジスタは、SuperH マイコンタイプの命令では、SH-1、SH-2 と同じように使われます。これに対して DSP タイプの命令では、メモリをアクセスするためのアドレスレジスタ、インデックスレジスタとして使われま

#### 2.1.1 汎用レジスタ

汎用レジスタ ( $R_n$ ) は、32 ビットの長さで、R0 から R15 までの 16 本あります。汎用レジスタは、データ処理、アドレス計算に使われます。

SuperH マイコンタイプの命令では、R0 は、インデックスレジスタとしても使用します。いくつかの命令では使用できるレジスタが R0 に固定されています。R15 は、スタックポインタ (SP) として使われます。例外処理でのステータスレジスタ (SR) とプログラムカウンタ (PC) の退避、復帰は R15 を用いてスタックを参照し行います。

DSP タイプの命令では、汎用レジスタ 16 本のうち、8 つのレジスタが X、Y データメモリおよび I バスを使うデータメモリ (シングルデータ) のアドレッシングに使われます。

X メモリをアクセスするためには、X アドレスレジスタ [ $A_x$ ] として R4、R5 を使い、X インデックスレジスタ [ $I_x$ ] として R8 を使います。Y メモリをアクセスするためには、Y アドレスレジスタ [ $A_y$ ] として R6、R7 を使い、Y インデックスレジスタ [ $I_y$ ] として R9 を使います。I バスを使ってシングルデータをアクセスするためには、シングルデータアドレスレジスタ [ $A_s$ ] として R2、R3、R4、R5 を使い、シングルデータインデックスレジスタ [ $I_s$ ] として R8 を使います。

DSP タイプの命令は X と Y データメモリを同時にアクセスできます。X と Y データメモリのアドレスを指定するために、2 組のアドレスポインタがあります。

汎用レジスタを図 2.1 に示します。

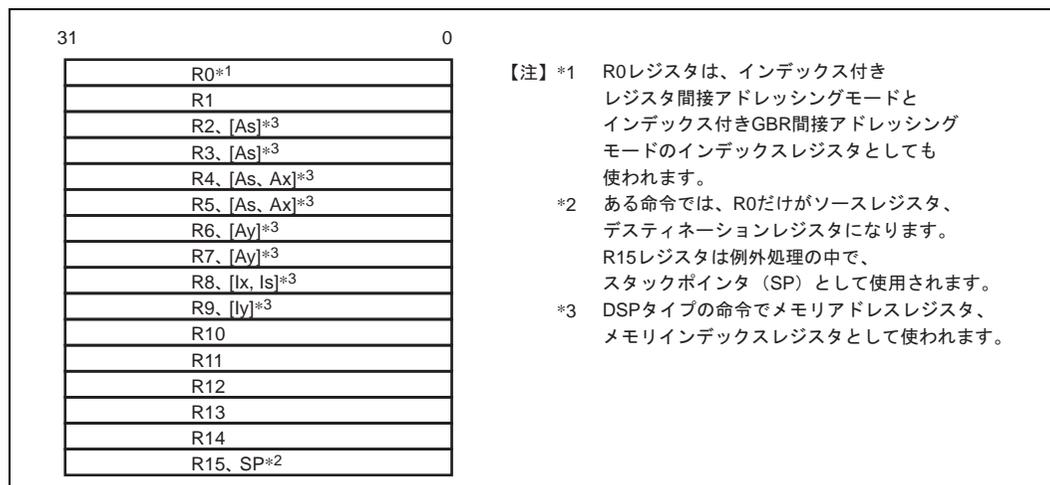


図 2.1 汎用レジスタの構成

アセンブラでは R2、R3、...、R9 の記号名 (シンボル) を使います。もし DSP タイプ命令のためのレジスタの役割を明示した名前にしたいときは、レジスタの別名 (エイリアス、alias) を使います。アセンブラで次のように書きます。

```
Ix: .REG (R8)
```

名前 Ix が R8 の別名になります。そのほか次のように別名を付けます。

```
Ax0: .REG (R4)
```

```
Ax1: .REG (R5)
```

```
Ix: .REG (R8)
```

```
Ay0: .REG (R6)
```

```
Ay1: .REG (R7)
```

```
Iy: .REG (R9)
```

```
As0: .REG (R4); これはシングルデータ転送のために別名が必要なときの定義です。
```

```
As1: .REG (R5); これはシングルデータ転送のために別名が必要なときの定義です。
```

```
As2: .REG (R2); これはシングルデータ転送のために別名が必要なときの定義です。
```

```
As3: .REG (R3); これはシングルデータ転送のために別名が必要なときの定義です。
```

```
Is: .REG (R8); これはシングルデータ転送のために別名が必要なときの定義です。
```

### 2.1.2 コントロールレジスタ

コントロールレジスタは 32 ビットの長さで、ステータスレジスタ (SR: Status register)、繰り返し開始レジスタ (RS: Repeat start register)、繰り返し終了レジスタ (RE: Repeat end register)、グローバルベースレジスタ (GBR: Global base register)、ベクタベースレジスタ (VBR: Vector base register)、モジュロレジスタ (MOD: Modulo register) の 6 本があります。

SR レジスタは処理の状態を表します。

GBR レジスタは GBR 間接アドレッシングモードのベースアドレスとして使用し、内蔵周辺モジュールのレジスタのデータ転送などに使用します。

VBR レジスタは割り込みを含む例外処理ベクタ領域のベースアドレスとして使用します。

RS レジスタと RE レジスタはプログラムの繰り返し (ループ) を制御するために使います。SR レジスタの繰り返しカウンタ (RC: Repeat counter) に繰り返し回数を指定し、RS レジスタには繰り返し開始アドレスを指定し、RE レジスタに繰り返し終了アドレスを指定します。ただし、RS レジスタと RE レジスタに格納されるアドレスの値は、物理的な繰り返し開始アドレス、終了アドレスとは値が必ずしも同じとは限りません。

MOD レジスタは繰り返しデータのバッファリングのためのモジュロアドレッシングに使います。SR レジスタの DMX ビットまたは DMY ビットでモジュロアドレッシングの指定をし、MOD レジスタの上位 16 ビットにモジュロ終了アドレス (ME) を指定し、下位 16 ビットにモジュロ開始アドレス (MS) を指定します。なお、DMX と DMY ビットは同時にモジュロアドレッシングを指定することはできません。モジュロアドレッシングは X、Y データ転送命令 (MOVX、MOVY) のとき可能です。シングルデータ転送命令 (MOVS) ではできません。

図 2.2 にコントロールレジスタを示します。表 2.1 に SR レジスタのビットを示します。

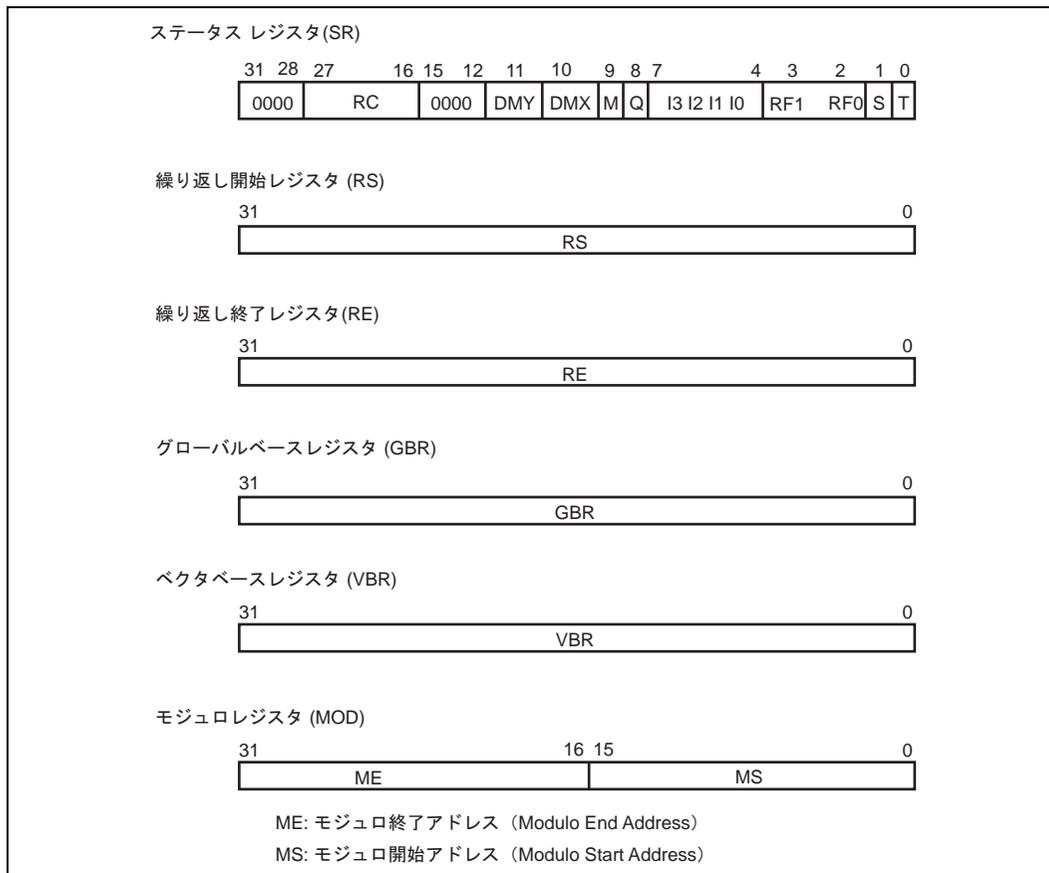


図 2.2 コントロールレジスタの構成

表 2.1 SR レジスタのビット

ビット	名称 (略称)	機能																
27 ~ 16	リピートカウンタ (RC)	繰り返し (ループ) 制御の繰り返し回数を指定します (2 ~ 4095)。																
11	Y ポインタ用モジュロアドレッシング指定 (DMY)	1: Y メモリアドレスポインタ、Ay (R6、R7) に対し、モジュロアドレッシングモードが有効になります。																
10	X ポインタ用モジュロアドレッシング指定 (DMX)	1: メモリアドレスポインタ、Ax (R4、R5) に対し、モジュロアドレッシングモードが有効になります。																
9	M ビット	DIV0S/U、DIV1 命令で使用します。																
8	Q ビット																	
7 ~ 4	割り込み要求マスク (I3 ~ I0)	割り込み要求を受け付けるレベルを表します (0 ~ 15)。																
3、2	リピートフラグ (RF1、0)	ゼロオーバーヘッド繰り返し (ループ) 制御に使用します。 SETRC 命令の時、以下のようにセットされます。 <table style="margin-left: 20px;"> <tr> <td>1Step リピートの時</td> <td>00</td> <td>RE</td> <td>RS = - 4</td> </tr> <tr> <td>2Step リピートの時</td> <td>01</td> <td>RE</td> <td>RS = - 2</td> </tr> <tr> <td>3Step リピートの時</td> <td>11</td> <td>RE</td> <td>RS = 0</td> </tr> <tr> <td>4Step 以上の時</td> <td>10</td> <td>RE</td> <td>RS &gt; 0</td> </tr> </table>	1Step リピートの時	00	RE	RS = - 4	2Step リピートの時	01	RE	RS = - 2	3Step リピートの時	11	RE	RS = 0	4Step 以上の時	10	RE	RS > 0
1Step リピートの時	00	RE	RS = - 4															
2Step リピートの時	01	RE	RS = - 2															
3Step リピートの時	11	RE	RS = 0															
4Step 以上の時	10	RE	RS > 0															
1	飽和演算ビット (S)	MAC 命令および DSP 命令で使用します。 1: 飽和演算を指定します (オーバーフローを防止します)。																
0	T ビット	MOVt、CMP/cond、TAS、TST、BT、BT/S、BF、BF/S、SETT、CLRT、および DT 命令のとき 0: 偽を表します 1: 真を表します  ADDV/C、SUBV/C、DIV0U/S、DIV1、NEGC、SHAR/L、SHLR/L、ROTR/L、および ROTCR/L 命令のとき 1: キャリ、ポロー、オーバーフローまたはアンダフローの発生を表します。																
31 ~ 28 15 ~ 12	0 ビット	0: 常に 0 が読み出されます。 0 を書き込んでください。																

RS、RE、MOD レジスタをアクセスするため専用のロード / ストア命令があります。例えば RS レジスタをアクセスするときは次のようになります。

```

LDC      Rm, RS;          Rm          RS
LDC.L   @Rm+, RS;       (Rm)        RS, Rm+4  Rm
STC     RS, Rn;         RS          Rn
STC.L   RS, @-Rn;      Rn-4        Rn, RS  (Rn)

```

ゼロオーバーヘッド繰り返し制御のために RS、RE レジスタにアドレスを設定する命令は次のとおりです。

```

LDRS    @(disp, PC); disp × 2 + PC  RS
LDRE    @(disp, PC); disp × 2 + PC  RE

```

GBR レジスタと VBR レジスタは従来の SuperH マイコンのレジスタと同じです。本 LSI には SR レジスタに 4 つの制御ビット (DMX ビット、DMY ビット、RF1 ビット、RF0 ビット) と、RC カウンタが追加されました。また RS レジスタ、RE レジスタ、MOD レジスタも新しいレジスタです。

### 2.1.3 システムレジスタ

システムレジスタは 32 ビットの長さで、積和レジスタ (MACH: Multiply and accumulate register high、MACL: Multiply and accumulate register low)、プロシージャレジスタ (PR: Procedure register)、プログラムカウンタ (PC: Program counter) の 4 本があります。

MACH, MACL は乗算または積和演算の結果を格納します\*。PR はサブルーチンプロシージャからの戻り先アドレスを格納します。PC は実行中のプログラムのアドレスを示し、処理の流れを制御します。PC は現在実行中の命令の 4 バイト先を示しています。これらのレジスタは SuperH マイコンのレジスタと同じです。

【注】 \* SH-1、SH-2 でサポートされていた命令実行時にのみ使用されます。SH-DSP で新たに追加された乗算命令 (PMULS) では使用しません。

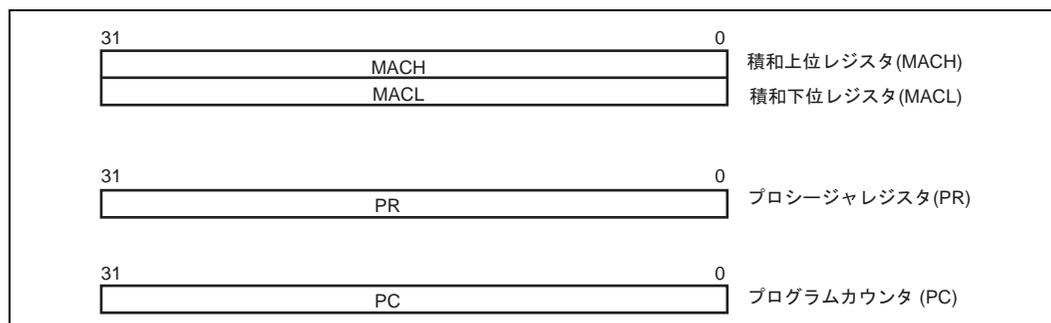


図 2.3 システムレジスタの構成

本 LSI ではさらに、後述の DSP ユニット用のレジスタ (DSP レジスタ) の内、DSP ステータスレジスタ (DSR) および 8 本のデータレジスタのうちの 5 本 (A0、X0、X1、Y0、Y1) がシステムレジスタとしても扱われます。このうち A0 レジスタは 40 ビットレジスタですが、A0 レジスタからデータを出力する場合はガードビット部分 (A0G) は無視され、A0 レジスタにデータを入力する場合はデータの MSB がガードビット部分 (A0G) にコピーされます。

### 2.1.4 DSP レジスタ

DSP ユニットには DSP レジスタとして 8 つのデータレジスタと 1 つのコントロールレジスタがあります。

DSP データレジスタは 2 本の 40 ビット長の A0、A1 レジスタと、6 本の 32 ビット長の M0、M1、X0、X1、Y0、Y1 レジスタがあります。A0、A1 レジスタには、それぞれ 8 ビットのガードビット、A0G、A1G がありません。

DSP データレジスタは、DSP 命令のオペランドとして DSP データのデータ転送、データ処理に使われます。DSP データレジスタをアクセスする命令には、DSP データ処理、X、Y データ転送処理、の 3 つのタイプがあります。

コントロールレジスタは 32 ビット長の DSP ステータスレジスタ (DSR: DSP status register) で、演算結果を表します。DSR レジスタには演算結果を表すビット、符号付き大ビット (GT: Signed greater than)、ゼロビット (Z: Zero value)、負値ビット (N: Negative value)、オーバーフロービット (V: overflow)、DSP 状態ビット (DC: DSP condition) と、DC ビットの設定を制御する状態選択ビット (CS: Condition select) があります。

DC ビットは状態フラグの一つを表し、SuperH マイコン CPU コアの T ビットとよく似ています。条件付き DSP タイプ命令の場合、DSP データ処理は、DC ビットに従って実行が制御されます。この制御は DSP ユニットでの実行にだけ関係し、DSP レジスタだけが更新されます。アドレス計算や、ロード/ストア命令などの SuperH マイコンの CPU コアの実行命令には関係しません。コントロールビット CS (ビット 2 から 0) は DC ビットを設定する状態を指定します。

DSP タイプ命令には、無条件 DSP タイプ命令と条件付き DSP タイプ命令があります。無条件 DSP タイプのデータ処理は、PMULS、MOVX、MOVY、MOVS 命令を除いて、状態ビットと DC ビットを更新します。条件付き DSP タイプ命令は DC ビットの状態によって実行されますが、実行された場合も実行されない場合も DSR レジスタは更新されません。

DSP レジスタを図 2.4 に示します。DSR レジスタのビットの機能を表 2.2 に示します。

A0、X0、X1、Y0、Y1、DSR レジスタは CPU コア命令ではシステムレジスタとして取り扱われます。

## 2. CPU

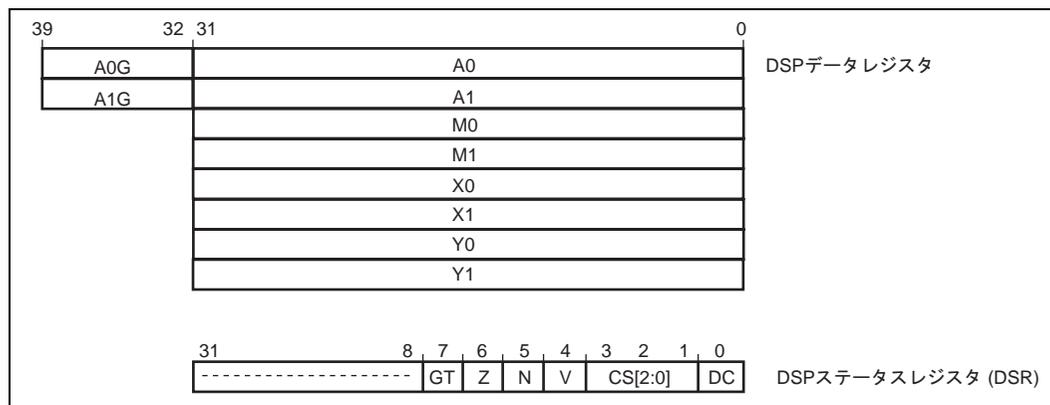


図 2.4 DSP レジスタの構成

表 2.2 DSR レジスタのビット

ビット	名称 (略称)	機能
31 ~ 8	予約ビット	0 : 常に 0 が読み出されます 書き込む値も 0 にしてください。
7	符号付き大ビット (GT)	演算結果が正 (ゼロをのぞく)、またはオペランド 1 がオペランド 2 より大きいことを示します 1 : 演算結果が正、またはオペランド 1 が大きい
6	ゼロビット (Z)	演算結果がゼロ (0)、またはオペランド 1 がオペランド 2 と等しいことを示します 1 : 演算結果がゼロ (0)、または等しい
5	負値ビット (N)	演算結果が負、またはオペランド 1 がオペランド 2 より小さいことを示します 1 : 演算結果が負、またはオペランド 1 が小さい
4	オーバーフロービット (V)	演算結果がオーバーフローしたことを示します 1 : 演算結果がオーバーフロー
3 ~ 1	状態選択ビット (CS)	DC ビットに設定する演算結果状態を選択するためのモードを指定します 110、111 は指定しないでください 000 : キャリ/ポローモード 001 : 負値モード 010 : ゼロモード 011 : オーバフローモード 100 : 符号付き大モード 101 : 符号付き以上モード
0	DSP 状態ビット (DC)	CS ビットで指定されたモードで演算結果の状態を設定します 0 : 指定されたモードの状態が成立しない (不成立) 1 : 指定されたモードの状態が成立

### 2.1.5 ガードビットとオーバーフローの扱いに関する注意事項

DSP ユニットでのデータ演算は基本的には 32 ビット演算ですが、演算時には、常時 8 ビットのガードビット部分も含めて 40 ビット長で実行されます。ガードビット部分が 32 ビット部分の MSB の値と一致しない場合、演算結果はオーバーフローとして扱われます。この場合、N ビットは、オーバーフローの有無にかかわらず、演算結果の正しい状態を示します。これはデスティネーションオペランドが 32 ビット長のレジスタであっても同じです。常に 8 ビット分のガードビットが仮定され、各状態フラグがアップデートされます。

ガードビットを使っても正しく結果を表示できないような桁あふれが生じた場合は、N フラグは正しい状態を示すことはできません

### 2.1.6 レジスタの初期値

リセット後のレジスタの値を表 2.3 に示します。

表 2.3 レジスタの初期値

区分	レジスタ	初期値
汎用レジスタ	R0 ~ R14	不定
	R15 (SP)	ベクタアドレステーブル中の SP の値
コントロールレジスタ	SR	I3 ~ I0 は 1111 (HF)、予約ビット、RC、DMY、DMX は 0、その他は不定
	RS	不定
	RE	
	GBR	不定
	VBR	H'0000 0000
	MOD	不定
システムレジスタ	MACH、MACL、PR	不定
	PC	ベクタアドレステーブル中の PC の値
DSP レジスタ	A0、A0G、A1、A1G、M0、M1、X0、X1、Y0、Y1	不定
	DSR	H'0000 0000

## 2.2 データ形式

### 2.2.1 レジスタのデータ形式

レジスタオペランドのデータサイズは常にロングワード (32 ビット) です。メモリ上のデータをレジスタへロードするとき、メモリオペランドのデータサイズがバイト (8 ビット)、またはワード (16 ビット) の場合は、ロングワードに符号拡張し、レジスタに格納します。

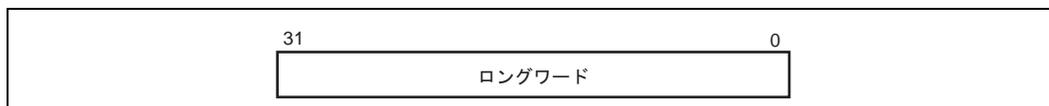


図 2.5 レジスタのデータ形式

### 2.2.2 メモリ上でのデータ形式

バイト、ワード、ロングワードのデータ形式があります。

バイトデータは任意番地に、ワードデータは  $2n$  番地から、ロングワードデータは  $4n$  番地から配置してください。その境界以外からアクセスすると、アドレスエラーが発生します。このとき、アクセスした結果は保証しません。特に、スタックポインタ (SP、R15) が指し示すスタックエリアには、プログラムカウンタ (PC) とステータスレジスタ (SR) をロングワードで格納しますので、スタックポインタの値が必ず  $4n$  になるように設定してください。

リトルエンディアンでメモリをアクセスするプロセッサとメモリを共用するために、CS2、4 空間 (エリア 2、4) をリトルエンディアンでアクセスできる機能を持っています。リトルエンディアンと通常のビッグエンディアンとは、バイトデータの並び順が異なります。

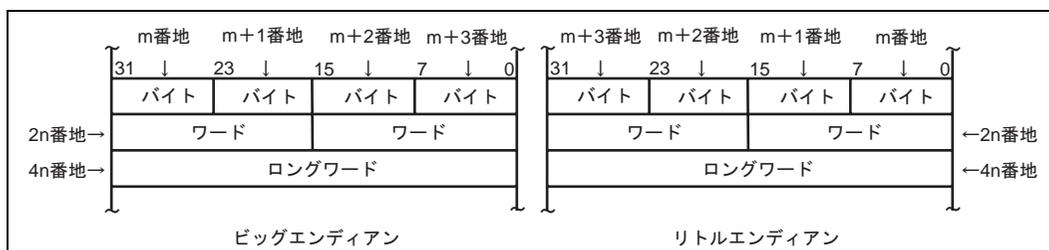


図 2.6 メモリ上のデータ形式

### 2.2.3 イミディエイトデータのデータ形式

バイトのイミディエイトデータは命令コードの中に配置します。

MOV、ADD、CMP/EQ 命令ではイミディエイトデータを符号拡張後、レジスタとロングワードで演算します。一方、TST、AND、OR、XOR 命令ではイミディエイトデータをゼロ拡張後、ロングワードで演算します。したがって、AND 命令でイミディエイトデータを用いると、デスティネーションレジスタの上位 24 ビットは常にクリアされます。

ワードとロングワードのイミディエイトデータは命令コードの中に配置せず、メモリ上のテーブルに配置してください。メモリ上のテーブルは、ディスプレースメント付き PC 相対アドレッシングモードを使ったイミディエイトデータのデータ転送命令 (MOV) で、参照してください。

### 2.2.4 DSP タイプデータ形式

本 LSI には命令に対応して 3 つの異なるデータ形式があります。固定小数点データ形式、整数データ形式、論理データ形式です。

DSP タイプの固定小数点データ形式はビット 31 とビット 30 の間に 2 進小数点があります。ガードビット付き、ガードビットなし、乗算入力の 3 種類があり、それぞれ有効ビット長と表せる値の範囲が異なります。

DSP タイプの整数データ形式はビット 16 とビット 15 の間に 2 進小数点があります。ガードビット付き、ガードビットなし、シフト量の 3 種類があり、それぞれ有効ビット長と表せる値の範囲が異なります。算術シフト (PSHA) のシフト量は 7 ビットの領域で - 64 ~ + 63 までを表せますが、実際に有効なのは - 32 ~ + 32 までの値です。同様に論理シフトのシフト量は 6 ビットの領域ですが、実際に有効なのは - 16 ~ + 16 までの値です。

DSP タイプの論理データ形式は小数点がありません。

データ形式とデータの有効な長さは命令と DSP レジスタによって決まります。

3 つの DSP タイプのデータ形式とその 2 進小数点の位置、および参考として SuperH タイプのデータ形式を図 2.7 に示します。

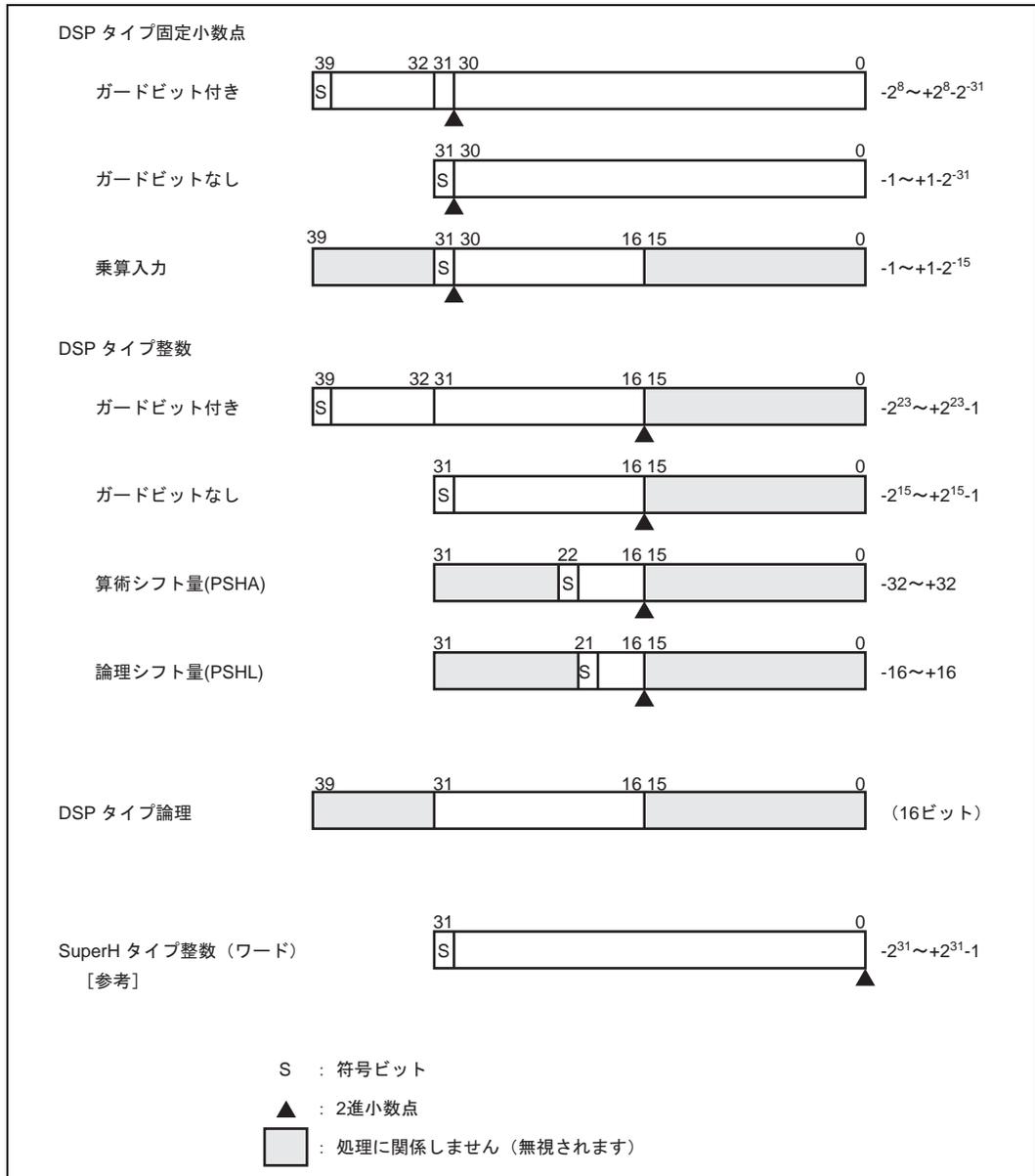


図 2.7 DSP タイプデータ形式

### 2.2.5 DSP タイプ命令とデータ形式

DSP データ形式とデータの有効な長さは DSP タイプ命令と DSP レジスタによって決まります。DSP データレジスタをアクセスする命令には、DSP データ処理、X、Y データ転送処理、シングルデータ転送処理の 3 つのタイプがあります。

#### (1) DSP データ処理

DSP 固定小数点データ処理で、A0、A1 レジスタをソースレジスタとして使うときは、ガードビット（ビット 39 ~ 32）は有効です。A0、A1 以外のレジスタ（M0、M1、X0、X1、Y0、Y1 レジスタ）をソースレジスタとして使うときは、そのレジスタデータの符号拡張されたものがビット 39 ~ 32 のデータとなります。A0、A1 レジスタをデスティネーションレジスタとして使うときは、ガードビット（ビット 39 ~ 32）は有効です。A0、A1 以外のレジスタをデスティネーションレジスタとして使うときは、結果のデータのビット 39 ~ 32 は無視されません。

DSP 整数データ処理の場合は DSP 固定小数点データ処理と同じです。ただし、ソースレジスタの下位ワード（下位 16 ビット、ビット 15 ~ 0）は無視されます。デスティネーションレジスタの下位ワードは 0 でクリアされます。

DSP 論理データ処理のソースレジスタは上位ワード（上位 16 ビット、ビット 31 ~ 16）が有効です。下位ワードと A0、A1 レジスタのガードビットは無視されます。デスティネーションレジスタは上位ワードが有効です。下位ワードと A0、A1 レジスタのガードビットは 0 でクリアされます。

#### (2) X、Y データ転送

MOVX.W、MOVY.W 命令は、16 ビットの X、Y データバスを介して X、Y メモリをアクセスします。レジスタにロードされるデータ、レジスタからストアされるデータは、常に上位ワード（上位 16 ビット、ビット 31 ~ 16）です。

ロードの場合 MOVX.W は X0、X1 レジスタをデスティネーションレジスタとして、X メモリをロードします。MOVY.W は Y0、Y1 レジスタをデスティネーションレジスタとして、Y メモリをロードします。データはレジスタの上位ワードに格納され、レジスタの下位ワードは 0 でクリアされます。

A0、A1 レジスタの上位ワードは、これらのデータ転送命令で X または Y メモリにデータをストアすることができますが、ほかのレジスタからはストアできません。A0、A1 レジスタのガードビットおよび下位ワードは無視されます。

#### (3) シングルデータ転送

MOVS.W、MOVS.L 命令は、データバス（CDB）を介して、どのメモリでもアクセスできます。すべての DSP レジスタは CDB バスとつながっており、データ転送のときソースレジスタ、デスティネーションレジスタいずれにもなります。データ転送にはワードとロングワードの 2 つのモードがあります。ワードモードでは、A0G、A1G レジスタを除いた DSP レジスタの上位ワードにロードまたは上位ワードからストアされます。ロングワードモードでは、A0G、A1G レジスタを除いた DSP レジスタの 32 ビットにロードまたは 32 ビットからストアされます。シングルデータ転送では A0G、A1G レジスタを独立したレジスタとして取り扱うことができます。A0G、A1G レジスタにロード、ストアするデータ長は 8 ビットです。

ワードモードで DSP レジスタをソースレジスタとして使う場合は、A0G、A1G 以外のレジスタからデータをストアすると、レジスタの上位ワードが転送されます。A0、A1 レジスタの場合は、ガードビットは無視されません。ワードモードで、A0G、A1G レジスタがソースレジスタの場合は、データは 8 ビットだけがレジスタからストアされ、上位ビットは符号拡張されます。

ワードモードでデスティネーションレジスタとして使う場合は、A0G、A1G レジスタを除いた DSP レジスタの上位ワードにロードされます。A0G、A1G 以外のレジスタにデータがロードされると、レジスタの下位ワードは 0 でクリアされます。A0、A1 レジスタの場合は、データの符号が拡張されてガードビットに格納され、下位ワードは 0 でクリアされます。ワードモードで、A0G、A1G レジスタがデスティネーションレジスタの場合は、データは最下位 8 ビットがレジスタにロードされ、A0、A1 レジスタは 0 でクリアされずに、それまでの値を保持します。

ロングワードモードでソースレジスタとして使う場合は、A0G、A1G 以外のレジスタからデータをストアすると、レジスタの 32 ビットが転送されます。A0、A1 レジスタをソースレジスタとして使う場合は、ガードビットは無視されます。ロングワードモードで、A0G、A1G レジスタがソースレジスタの場合は、データは 8 ビットだけがレジスタからストアされ、上位ビットは符号拡張されます。

ロングワードモードでデスティネーションレジスタとして使う場合は、A0G、A1G レジスタを除いた DSP レジスタの 32 ビットにロードされます。A0、A1 レジスタの場合は、データの符号が拡張されてガードビットに格納されます。ロングワードモードで、A0G、A1G レジスタがデスティネーションレジスタの場合は、データは最下位 8 ビットがレジスタにロードされ、A0、A1 レジスタは 0 でクリアされずに、それまでの値を保持します。

DSP 命令でのレジスタ上のデータ形式を表 2.4、表 2.5 に示します。命令によってはアクセスできないレジスタがあります。たとえば、PMULS 命令は、A1 レジスタをソースレジスタに指定できますが、A0 レジスタはできません。詳細は、命令の説明を参照してください。

データ転送時の DSP レジスタとバスとの関係を図 2.8 に示します。

表 2.4 DSP 命令のソースレジスタのデータ形式

レジスタ	命令		ガードビット		レジスタビット			
			39	32	31	16	15	0
A0, A1	DSP 演算	固定小数点、PDMSB、PSHA	40bit データ					
		整数	24bit データ					
		論理、PSHL、PMULS			16bit データ			
	データ転送	MOVX.Y.W、MOV.S.W			16bit データ			
		MOV.S.L			32bit データ			
A0G、 A1G	データ転送	MOV.S.W	データ					
		MOV.S.L	データ					
X0, X1 Y0, Y1 M0, M1	DSP 演算	固定小数点、PDMSB、PSHA	符号*		32bit データ			
		整数	符号*		16bit データ			
		論理、PSHL、PMULS			16bit データ			
	データ転送	MOV.S.W			16bit データ			
		MOV.S.L			32bit データ			

【注】 \* 符号が拡張され ALU のガードビットに格納されます

表 2.5 DSP 命令のデスティネーションレジスタのデータ形式

レジスタ	命令		ガードビット		レジスタビット			
			39	32	31	16	15	0
A0, A1	DSP 演算	固定小数点、PSHA、PMULS	(符号拡張)		40bit 結果			
		整数、PDMSB	(符号拡張)		24bit 結果		0 クリア	
		論理、PSHL	0 クリア		16bit 結果		0 クリア	
	データ転送	MOV.S.W	符号拡張		16bit データ		0 クリア	
		MOV.S.L	符号拡張		32bit データ			
A0G、 A1G	データ転送	MOV.S.W	データ		更新せず			
		MOV.S.L	データ		更新せず			
X0, X1 Y0, Y1 M0, M1	DSP 演算	固定小数点、PSHA、PMULS			32bit 結果			
		整数、論理、PDMSB、PSHL			16bit 結果		0 クリア	
	データ転送	MOVX.W、MOVY.W、MOV.S.W			16bit データ		0 クリア	
		MOV.S.L			32bit データ			

## 2. CPU

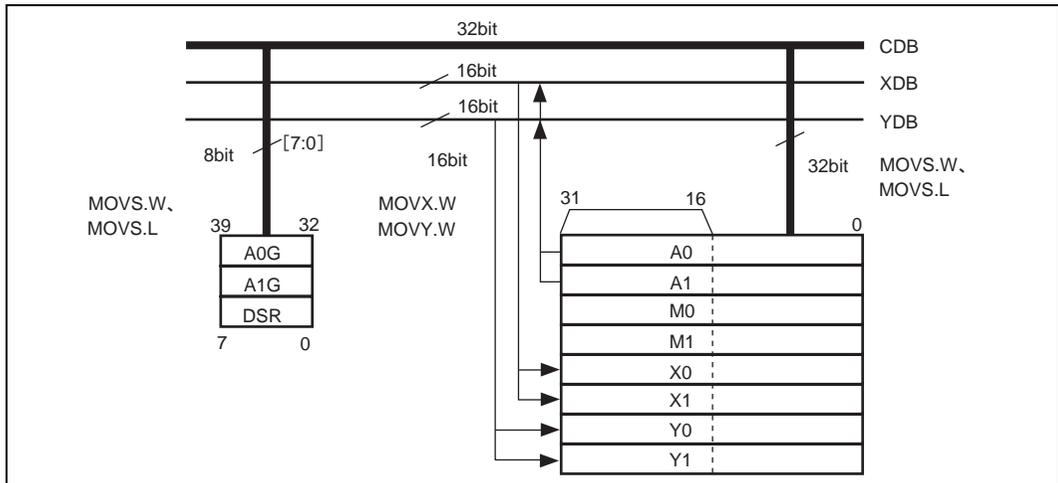


図 2.8 データ転送時の DSP レジスタとバスとの関係

## 2.3 CPU コア命令の特長

CPU コア命令は RISC 形式の命令です。特長は次のとおりです。

### (1) 16 ビット固定長命令

命令長はすべて 16 ビット固定長です。これによりプログラムのコード効率が向上します。

### (2) 1 命令 / 1 ステート

パイプライン方式を採用し、基本命令は、1 命令を 1 ステートで実行できます。62.5MHz 動作時、1 ステートは 16.0ns になります。

### (3) データサイズ

演算の基本的なデータサイズはロングワードです。メモリのアクセスサイズは、バイト / ワード / ロングワードを選択できます。メモリのバイトとワードのデータは符号拡張後、ロングワードで演算されます。イミディエイトデータは算術演算では符号拡張後、論理演算ではゼロ拡張後、ロングワードで演算されます。

表 2.6 ワードデータの符号拡張

本 LSI の CPU	説明	他の CPU の例
MOV.W @(disp,PC),R1 ADD R1,R0 ..... .DATA.W H'1234	32 ビットに符号拡張され、R1 は H'00001234 になります。次に ADD 命令で演算されます。	ADD.W #H'1234,R0

【注】 @(disp,PC) でイミディエイトデータを参照します。

### (4) ロードストアアーキテクチャ

基本演算はレジスタ間で実行します。メモリとの演算は、レジスタにデータをロードし実行します（ロードストアアーキテクチャ）。ただし、AND などのビットを操作する命令は直接メモリに対して実行します。

### (5) 遅延分岐

無条件分岐命令などは、遅延分岐として実行されます。遅延分岐命令の場合、遅延分岐命令の直後の命令（スロット命令）を実行してから、分岐します。これにより、分岐時のパイプラインの乱れを軽減しています。

遅延分岐の分岐動作そのものは、スロット命令実行後に発生します。しかし、分岐動作を除くレジスタの更新などの命令の実行は、遅延分岐命令、遅延スロット命令の順に行われます。例えば、遅延スロットで分岐先アドレスが格納されているレジスタの内容を変更しても、分岐先アドレスは変更前のレジスタ内容のままです。

表 2.7 遅延分岐命令

本 LSI の CPU	説明	他の CPU の例
BRA TRGET ADD R1,R0	TRGET に分岐する前に ADD を実行します。	ADD.W R1,R0 BRA TRGET

## 2. CPU

### (6) 乗算 / 積和演算

16 × 16 32 の乗算を 1 ~ 3 ステート、16 × 16+64 64 の積和演算を 2 ~ 3 ステートで実行します。32 × 32 64 の乗算や、32 × 32+64 64 の積和演算を 2 ~ 4 ステートで実行します。

### (7) T ビット

比較結果はステータスレジスタ (SR) の T ビットに反映し、その真、偽によって条件分岐します。必要最小限の命令によってのみ T ビットを変化させ、処理速度を向上させています。

表 2.8 T ビット

本 LSI の CPU	説明	他の CPU の例
CMP/GE R1,R0	R0 R1 のとき T ビットがセットされます。	CMP.W R1,R0
BT TRGET0	R0 R1 のとき TRGET0 へ	BGE TRGET0
BF TRGET1	R0<R1 のとき TRGET1 へ分岐します。	BLT TRGET1
ADD #-1,R0	ADD では T ビットが変化しません。	SUB.W #1,R0
CMP/EQ #0,R0	R0=0 のとき T ビットがセットされます。	BEQ TRGET
BT TRGET	R0=0 のとき分岐します。	

### (8) イミディエイトデータ

バイトのイミディエイトデータは命令コードの中に配置します。ワードとロングワードのイミディエイトデータは命令コードの中に配置せず、メモリ上のテーブルに配置します。メモリ上のテーブルはディスプレイースメント付き PC 相対アドレッシングモードを使ったイミディエイトデータのデータ転送命令 (MOV) で参照します。

表 2.9 イミディエイトデータによる参照

区分	本 LSI の CPU	他の CPU の例
8 ビットイミディエイト	MOV #H'12,R0	MOV.B #H'12,R0
16 ビットイミディエイト	MOV.W @(disp,PC),R0 ..... .DATA.W H'1234	MOV.W #H'1234,R0
32 ビットイミディエイト	MOV.L @(disp,PC),R0 ..... .DATA.L H'12345678	MOV.L #H'12345678,R0

【注】 @(disp,PC) でイミディエイトデータを参照します。

### (9) 絶対アドレス

絶対アドレスでデータを参照するときは、あらかじめ絶対アドレスの値を、メモリ上のテーブルに配置しておきます。命令実行時にイミディエイトデータをロードする方法で、この値をレジスタに転送し、レジスタ間接アドレッシングモードでデータを参照します。

表 2.10 絶対アドレスによる参照

区分	本 LSI の CPU	他の CPU の例
絶対アドレス	MOV.L @(disp,PC),R1 MOV.B @R1,R0 ..... .DATA.L H'12345678	MOV.B @H'12345678,R0

## (10) 16 ビット / 32 ビットディスプレースメント

16 ビットまたは 32 ビットディスプレースメントでデータを参照するときは、あらかじめディスプレースメントの値をメモリ上のテーブルに配置しておきます。命令実行時にイミディエイトデータをロードする方法で、この値をレジスタに転送し、インデックス付きレジスタ間接アドレッシングモードでデータを参照します。

表 2.11 ディスプレースメントによる参照

区分	本 LSI の CPU	他の CPU の例
16 ビットディスプレースメント	MOV.W @(disp,PC),R0 MOV.W @(R0,R1),R2 ..... .DATA.W H'1234	MOV.W @(H'1234,R1),R2

## 2.4 命令形式

### 2.4.1 CPU 命令のアドレッシングモード

CPU コアで実行される命令のアドレッシングモードと実効アドレスの計算方法は次のとおりです。

表 2.12 CPU 命令のアドレッシングモードと実効アドレス

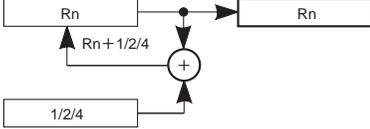
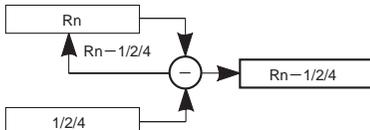
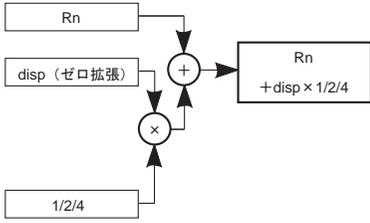
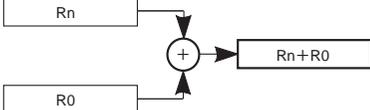
アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
レジスタ直接	Rn	実効アドレスはレジスタ Rn です。(オペランドはレジスタ Rn の内容です。)	
レジスタ間接	@Rn	実効アドレスはレジスタ Rn の内容です。 	Rn
ポストインクリメントレジスタ間接	@Rn+	実効アドレスはレジスタ Rn の内容です。命令実行後 Rn に定数を加算します。定数はオペランドサイズがバイトのとき 1、ワードのとき 2、ロングワードのとき 4 です。 	Rn 命令実行後バイト : Rn + 1 Rn ワード : Rn + 2 Rn ロングワード : Rn + 4 Rn
プリデクリメントレジスタ間接	@(disp:4,Rn)	実効アドレスは、あらかじめ定数を減算したレジスタ Rn の内容です。定数はバイトのとき 1、ワードのとき 2、ロングワードのとき 4 です。 	バイト : Rn - 1 Rn ワード : Rn - 2 Rn ロングワード : Rn - 4 Rn (計算後の Rn で命令実行)
ディスプレイースメント付きレジスタ間接	@(disp:4,Rn)	実効アドレスはレジスタ Rn に 4 ビットディスプレイースメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってバイトで 1 倍、ワードで 2 倍、ロングワードで 4 倍します。 	バイト : Rn + disp ワード : Rn + disp × 2 ロングワード : Rn + disp × 4
インデックス付きレジスタ間接	@(R0,Rn)	実効アドレスはレジスタ Rn に R0 を加算した内容です。 	Rn + R0

表 2.12 CPU 命令のアドレッシングモードと実効アドレス

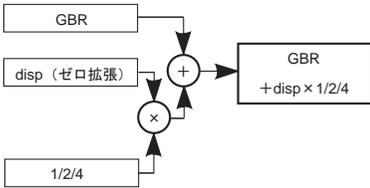
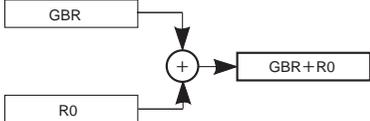
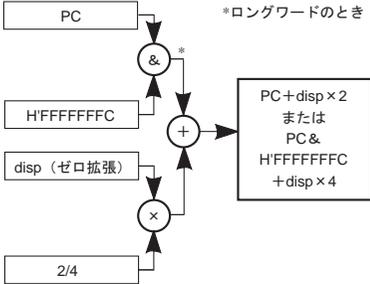
アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
ディスプレイメント付き GBR 間接	@ (disp:8,GBR)	<p>実効アドレスはレジスタ GBR に 8 ビットディスプレイメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってバイトで 1 倍、ワードで 2 倍、ロングワードで 4 倍します。</p> 	バイト : $GBR + disp$ ワード : $GBR + disp \times 2$ ロングワード : $GBR + disp \times 4$
インデックス付き GBR 間接	@ (R0,GBR)	<p>実効アドレスはレジスタ GBR に R0 を加算した内容です。</p> 	$GBR + R0$
ディスプレイメント付き PC 相対	@ (disp:8,PC)	<p>実効アドレスはレジスタ PC に 8 ビットディスプレイメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってワードで 2 倍、ロングワードで 4 倍します。さらにロングワードのときは PC の下位 2 ビットをマスクします。</p> 	ワード : $PC + disp \times 2$ ロングワード : $PC \& H'FFFFFFC + disp \times 4$

表 2.12 CPU 命令のアドレッシングモードと実効アドレス

アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
PC 相対	disp:8	<p>実効アドレスはレジスタ PC に 8 ビットディスプレースメント disp を符号拡張後 2 倍し、加算した内容です。</p>	$PC + disp \times 2$
	disp:12	<p>実効アドレスはレジスタ PC に 12 ビットディスプレースメント disp を符号拡張後 2 倍し、加算した内容です。</p>	$PC + disp \times 2$
	Rn	<p>実効アドレスはレジスタ PC に Rn を加算した内容です。</p>	$PC + Rn$
イミディエイト	#imm:8	TST、AND、OR、XOR 命令の 8 ビットイミディエイト imm はゼロ拡張します。	
	#imm:8	MOV、ADD、CMP/EQ 命令の 8 ビットイミディエイト imm は符号拡張します。	
	#imm:8	TRAPA 命令の 8 ビットイミディエイト imm はゼロ拡張後、4 倍します。	

## 2.4.2 DSP データアドレッシング

DSP 命令では 2 つの異なるメモリアクセスをします。1 つは X、Y データ転送命令 (MOVX.W、MOVY.W) で、もう 1 つはシングルデータ転送命令 (MOVS.W、MOVS.L) です。これらの 2 種類の命令のデータアドレッシングは異なります。データ転送命令の概要を表 2.13 に示します。

表 2.13 データ転送命令の概要

	X、Y データ転送処理 (MOVX.W、MOVY.W)	シングルデータ転送処理 (MOVS.W、MOVS.L)
アドレスレジスタ	Ax : R4、R5、Ay : R6、R7	As : R2、R3、R4、R5
インデックスレジスタ	Ix : R8、Iy : R9	Is : R8

表 2.13 データ転送命令の概要

	X、Y データ転送処理 (MOVX.W, MOVY.W)	シングルデータ転送処理 (MOVS.W, MOVS.L)
アドレッシング	Nop/Inc(+2)/ インデクス加算 : ポスト更新	Nop/Inc(+2,+4)/ インデクス加算 : ポスト更新 Dec(-2,-4) : プレ更新
モジュロアドレッシング	可能	不可
データバス	XDB、YDB	CDB
データ長	16bit (ワード)	16bit/32bit (ワード/ロングワード)
バス競合	なし	あり
メモリ	X、Y データメモリ	すべてのメモリ空間
ソースレジスタ	Dx, Dy : A0, A1	Ds : A0/A1, M0/M1, X0/X1, Y0/Y1, A0G, A1G
デスティネーションレジスタ	Dx : X0/X1, Dy : Y0/Y1	Ds : A0/A1, M0/M1, X0/X1, Y0/Y1, A0G, A1G

## (1) X、Y データアドレッシング

DSP 命令では MOVX.W、MOVY.W 命令を使って、X、Y データメモリを同時にアクセスすることができます。DSP 命令には同時に X、Y データメモリをアクセスするために 2 つのアドレスポイントがあります。DSP 命令にはポイントアドレッシングだけが可能で、イミディエイトアドレッシングはありません。アドレスレジスタは 2 つに分かれ、R4、R5 レジスタが X メモリのアドレスレジスタ (Ax) となり、R6、R7 レジスタが Y メモリのアドレスレジスタ (Ay) となります。X、Y データ転送命令には次の 3 つのアドレッシングがあります。

## (1) 更新なしアドレスレジスタ :

Ax、Ay レジスタがアドレスポイントです。更新されません。

## (2) 加算インデクスレジスタ :

Ax、Ay レジスタがアドレスポイントです。データ転送後それぞれ Ix、Iy レジスタの値が加算されます (ポスト更新)。

## (3) インクリメントアドレスレジスタ :

Ax、Ay レジスタがアドレスポイントです。データ転送後それぞれ + 2 が加算されます (ポスト更新)。それぞれのアドレスポイントにはインデクスレジスタがあります。R8 レジスタは X メモリアドレスレジスタ (Ax) のインデクスレジスタ (Ix) となり、R9 レジスタは Y メモリアドレスレジスタ (Ay) のインデクスレジスタ (Iy) となります。

X、Y データ転送命令はワードで処理します。X、Y データメモリを 16 ビットでアクセスします。そのためインクリメント処理は、アドレスレジスタに 2 を加えます。デクリメントさせるためには、- 2 をインデクスレジスタに設定し加算インデクスレジスタアドレッシングを指定します。X、Y データアドレッシング時は、アドレスポイントのビット 1 ~ 15 のみ有効となります。X、Y データアドレッシング時は、アドレスポイント、インデクスレジスタのビット 0 には必ず 0 を書き込んでください。

X、Y データ転送のアドレッシングを図 2.9 に示します。X、Y バスを使用して X メモリ、Y メモリへアクセスする場合、Ax (R4 または R5)、Ay (R6 または R7) の上位ワードは無視されます。また、@Ay+、@Ay+Iy の結果は、Ay の下位ワードに格納され、上位ワードは元の値が保持されます。

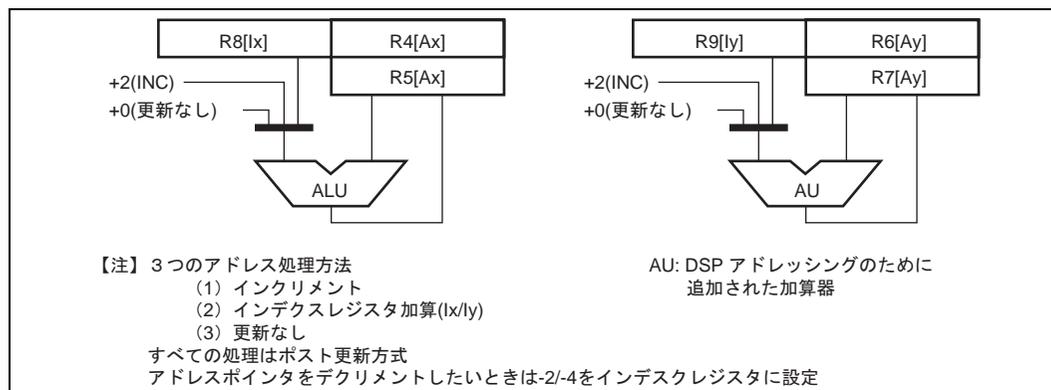


図 2.9 X、Y データ転送のアドレッシング

## (2) シングルデータアドレッシング

DSP 命令にはシングルデータ転送命令 (MOV.S.W、MOV.S.L) があり、DSP レジスタにデータをロードし、DSP レジスタからデータをストアします。この命令で R2 ~ R5 レジスタはシングルデータ転送のアドレスレジスタ ( $A_s$ ) として使われます。

シングルデータ転送命令には次の 4 つのデータアドレッシング命令があります。

### (1) 更新なしアドレスレジスタ :

$A_s$  レジスタがアドレスポインタです。更新されません。

### (2) 加算インデクスレジスタ :

$A_s$  レジスタがアドレスポインタです。データ転送後  $I_s$  レジスタの値が加算されます (ポスト更新)。

### (3) インクリメントアドレスレジスタ :

$A_s$  レジスタがアドレスポインタです。データ転送後 + 2 または + 4 が加算されます (ポスト更新)。

### (4) デクリメントアドレスレジスタ :

$A_s$  レジスタがアドレスポインタです。データ転送前に - 2、- 4 が加算 (+ 2 または + 4 が減算) されます (プレ更新)。

アドレスポインタ ( $A_s$ ) は R8 レジスタをインデクスレジスタ ( $I_s$ ) として使います。シングルデータ転送のアドレッシングを図 2.10 に示します。

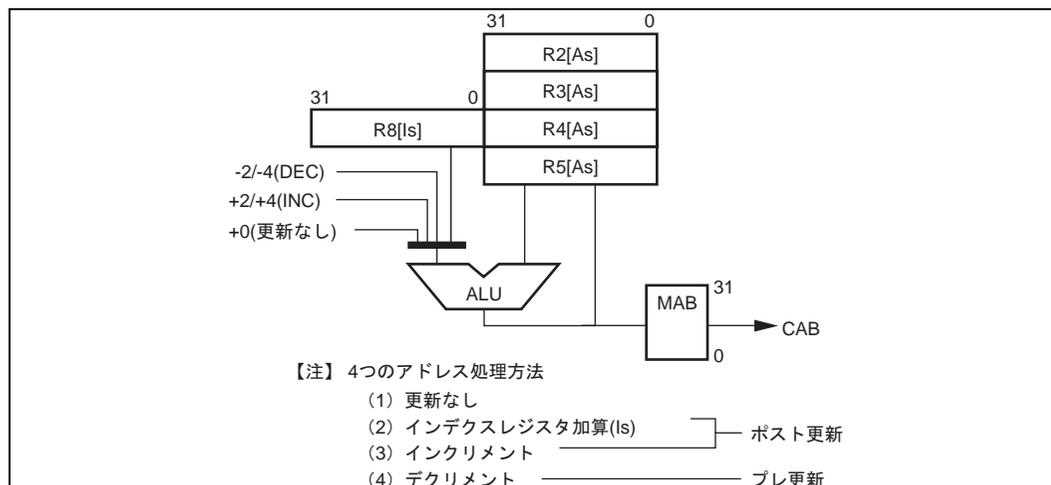


図 2.10 シングルデータ転送のアドレッシング

### (3) モジュロアドレッシング

本 LSI には、他の DSP と同じに、モジュロアドレッシングモードがあります。このモードでもアドレスレジスタは同じように更新されます。アドレスポインタの値がすでに設定されたモジュロ終了アドレスになると、アドレスポインタはモジュロ開始アドレスになります。

モジュロアドレッシングは X、Y データ転送命令 (MOVX.W、MOVY.W) にだけ有効です。SR レジスタの DMX ビットをセットすると X アドレスレジスタが、DMY ビットをセットすると Y アドレスレジスタがそれぞれモジュロアドレッシングモードになります。モジュロアドレッシングは X、Y アドレスレジスタどちらかに対してだけ有効です。両方を同時にモジュロアドレッシングモードにすることはできません。したがって、DMX と DMY を同時にセットしないでください。万一同時にセットされた場合には、DMY 側のみ有効となります。

モジュロアドレス領域の開始と終了アドレスを指定するための MOD レジスタがあり、MOD レジスタは MS (Modulo Start : モジュロ開始) と、ME (Modulo End : モジュロ終了) を格納します。MOD レジスタ (MS、ME) の使用例を次に示します。

```

MOV.L ModAddr, Rn;    Rn=ModEnd,   ModStart
LDC Rn, MOD;          ME=ModEnd,   MS=ModStart

ModAddr:              .DATA.W      mEnd;           ModEnd
                     .DATA.W      mStart;          ModStart

ModStart:             .DATA
                     :
ModEnd:               .DATA

```

## 2. CPU

MS、ME には開始、終了アドレスを指定して、その後で DMX または DMY ビットを 1 にセットします。アドレスレジスタの内容が ME と比較されます。もし ME と一致したら、開始アドレス MS をアドレスレジスタに格納します。アドレスレジスタの下位 16 ビットが ME と比較されます。

最大のモジュールサイズは 64K バイトです。これは X、Y データメモリをアクセスするには十分です。モジュールアドレッシングのブロック図を図 2.11 に示します。

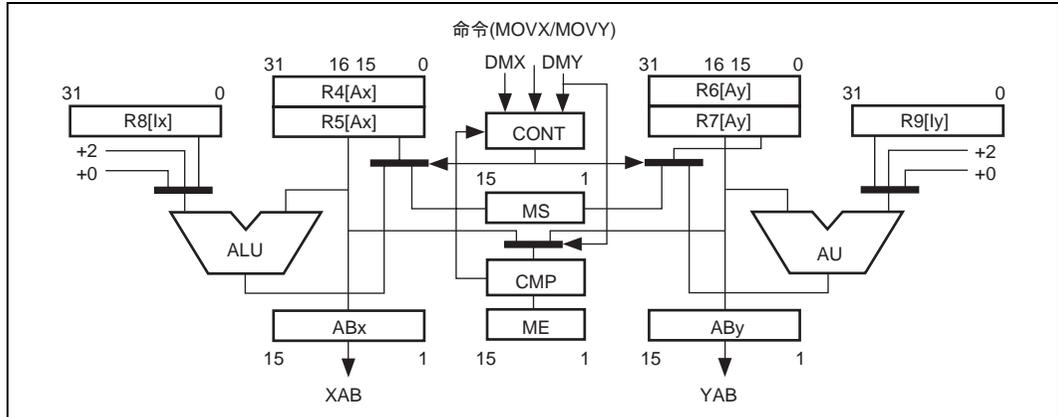


図 2.11 モジュールアドレッシング

モジュールアドレッシングの例を次に示します。

```
MS = H'E008; ME=H'E00C; R4=H'1000E008;
```

```
DMX=1; DMY=0; ( アドレスレジスタ Ax(R4,R5) に対するモジュールアドレッシングの設定です )
```

以上の設定により R4 レジスタは次のように変化します。

```
R4: H'1000E008
Inc.  R4: H'1000E00A
Inc.  R4: H'1000E00C
Inc.  R4: H'1000E008 (モジュール終了アドレスになったので、モジュール開始アドレスになります)
```

モジュール開始、終了アドレスの上位 16 ビットは同じになるようデータを配置してください。これはモジュール開始アドレスがアドレスレジスタの下位のビット 0 を除く 15 ビットだけを置き換えるからです。

**【注】** DSP データアドレッシングに加算インデックスを使う場合は、アドレスポインタは ME と一致せずその値を超えてしまうことがあります。この場合は、アドレスポインタはモジュール開始アドレスには戻りません。モジュールアドレッシングに限らず、X、Y データアドレッシング時は、ビット 0 が無視されます。アドレスポインタ、インデックスレジスタ、MS、ME のビット 0 には必ず 0 を書き込んでください。

#### (4) DSP アドレッシング動作

モジュールアドレッシングを含めて、パイプラインの実行ステージ (EX) での DSP アドレッシングの動作を次に示します。

```
if ( Operation is MOVX.W MOVY.W ) {
    ABx=Ax; ABy=Ay;
```

---

```

/* memory access cycle uses ABx and ABy. The addresses to be used have not been
updated */

/* Ax is one of R4,5 */
if ( DMX==0 || DMX==1 && DMY == 1 )} Ax=Ax+(+2 or R8[Ix] or +0);
/* Inc,Index,Not-Update */
else if (! not-update) Ax=modulo( Ax, (+2 or R8[Ix]) );

/* Ay is one of R6,7 */
if ( DMY==0 ) Ay=Ay+(+2 or R9[Iy] or +0); /* Inc,Index,Not-Update */
else if (! not-update) Ay=modulo( Ay, (+2 or R9[Iy]) );
}
else if ( Operation is MOVS.W or MOVS.L ) {
  if ( Addressing is Nop, Inc, Add-index-reg ) {
    MAB=As;
    /* memory access cycle uses MAB. The address to be used has not been updated */
    /* As is one of R2~5 */
    As=As+(+2 or +4 or R8[Is] or +0); /* Inc,Index,Not-Update */
  else { /* Decrement, Pre-update */
    /* As is one of R2~5 */
    As=As+(-2 or -4);
    MAB=As;
    /* memory access cycle uses MAB. The address to be used has been updated */
  }
}

/* The value to be added to the address register depends on addressing operations.
For example, (+2 or R8[Ix] or +0) means that
    +2                : if operation is increment
    R8[Ix]            : if operation is add-index-reg
    +0                : if operation is not-update
*/

function modulo ( AddrReg, Index ) {
  if ( AddrReg[15:0]==ME ) AddrReg[15:0]=MS;
  else AddrReg=AddrReg+Index;
  return AddrReg;
}

```

## 2. CPU

### 2.4.3 CPU 命令の命令形式

CPU コアで実行される命令の命令形式とソースオペランドとデスティネーションオペランドの意味を示します。命令コードによりオペランドの意味が異なります。記号は次のとおりです。

xxxx : 命令コード

mmmm : ソースレジスタ

nnnn : デスティネーションレジスタ

iiii : イミディエイトデータ

dddd : ディスプレースメント

表 2.14 CPU 命令の命令形式

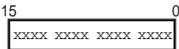
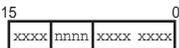
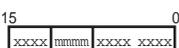
命令形式	ソースオペランド	デスティネーション オペランド	命令の例
0 形式 			NOP
n 形式 		nnnn : レジスタ直接	MOV T Rn
	コントロールレジスタ またはシステムレジスタ	nnnn : レジスタ直接	STS MACH,Rn
	コントロールレジスタ またはシステムレジスタ	nnnn : プリデクリメント レジスタ間接	STC.L SR,@-Rn
m 形式 	mmmm : レジスタ直接	コントロールレジスタ またはシステムレジスタ	LDC Rm,SR
	mmmm : ポストインクリ メントレジスタ間接	コントロールレジスタ またはシステムレジスタ	LDC.L @Rm+,SR
	mmmm : レジスタ間接		JMP @Rm
	mmmm : Rm を用いた PC 相対		BRAF Rm

表 2.14 CPU 命令の命令形式

命令形式	ソースオペランド	デスティネーション オペランド	命令の例	
nm 形式		mmmm : レジスタ直接	nnnn : レジスタ直接	ADD Rm,Rn
		mmmm : レジスタ直接	nnnn : レジスタ間接	MOV.L Rm,@Rn
		mmmm : ポストインクリメントレジスタ間接 (積和演算) nnnn : * ポストインクリメントレジスタ間接 (積和演算)	MACH,MACL	MAC.W @Rm+,@Rn+
		mmmm : ポストインクリメントレジスタ間接	nnnn : レジスタ直接	MOV.L @Rm+,Rn
		mmmm : レジスタ直接	nnnn : プリデクリメントレジスタ間接	MOV.L Rm,@-Rn
		mmmm : レジスタ直接	nnnn : インデックス付きレジスタ間接	MOV.L Rm,@(R0,Rn)
md 形式		mmmmdddd : ディスプレースメント付きレジスタ間接	R0 (レジスタ直接)	MOV.B @(disp,Rm),R0
nd4 形式		R0 (レジスタ直接)	nnnndddd : ディスプレースメント付きレジスタ間接	MOV.B R0,@(disp,Rn)
nmd 形式		mmmm : レジスタ直接	nnnndddd : ディスプレースメント付きレジスタ間接	MOV.L Rm,@(disp,Rn)
		mmmmdddd : ディスプレースメント付きレジスタ間接	nnnn : レジスタ直接	MOV.L @(disp,Rm),Rn
d 形式		dddddddd : ディスプレースメント付き GBR 間接	R0 (レジスタ直接)	MOV.L @(disp,GBR),R0
		R0 (レジスタ直接)	dddddddd : ディスプレースメント付き GBR 間接	MOV.L R0,@(disp,GBR)
		dddddddd : ディスプレースメント付き PC 相対	R0 (レジスタ直接)	MOVA @(disp,PC),R0
		dddddddd : PC 相対		BF label
d12 形式		dddddddddddd : PC 相対		BRA label (label=disp+PC)
nd8 形式		dddddddd : ディスプレースメント付き PC 相対	nnnn : レジスタ直接	MOV.L@(disp,PC),Rn

## 2. CPU

表 2.14 CPU 命令の命令形式

命令形式		ソースオペランド	デスティネーション オペランド	命令の例
i 形式		iiiiiii: イミディエイト	インデックス付き GBR 間接	AND.B#imm,@(R0,GBR)
		iiiiiii: イミディエイト	R0 (レジスタ直接)	AND #imm,R0
		iiiiiii: イミディエイト		TRAPA#imm
ni 形式		iiiiiii: イミディエイト	nnnn: レジスタ直接	ADD #imm,Rn

【注】 \* 積和命令では nnnn は、ソースレジスタです。

#### 2.4.4 DSP 命令の命令形式

本 LSI にはデジタル信号処理のための新しい命令が追加されています。新しい命令は次の 2 つに分けられます。

- (1) メモリと DSP レジスタのダブル、シングルデータ転送命令 (16 ビット長)
- (2) DSP ユニットで処理される並行処理命令 (32 ビット長)

それぞれの命令形式を図 2.12 に示します。

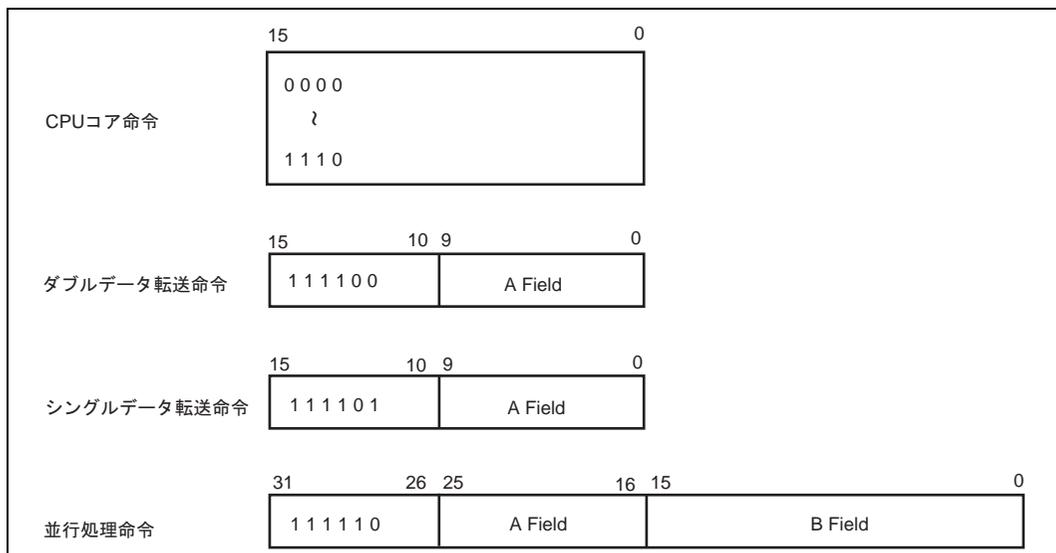


図 2.12 DSP 命令の命令形式

##### (1) ダブル、シングルデータ転送命令

ダブルデータ転送命令の命令形式を表 2.15 に、シングルデータ転送命令の命令形式を表 2.16 に示します。

2. CPU

表 2.15 ダブルデータ転送の命令形式

分類	ニーモニック	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
X メモリ データ 転送	NOPX	1	1	1	1	0	0	0		0		0		0	0				
	MOVX.W @Ax,Dx							Ax		Dx		0			0	1			
	MOVX.W @Ax+,Dx														1	0			
	MOVX.W @Ax+Ix,Dx														1	1			
	MOVX.W Da,@Ax										Da		1			0	1		
	MOVX.W Da,@Ax+															1	0		
	MOVX.W Da,@Ax+Ix															1	1		
Y メモリ データ 転送	NOPY	1	1	1	1	0	0		0		0		0				0	0	
	MOVY.W @Ay,Dy								Ay		Dy			0			0	1	
	MOVY.W @Ay+,Dy																1	0	
	MOVY.W @Ay+Iy,Dy																1	1	
	MOVY.W Da,@Ay											Da		1				0	1
	MOVY.W Da,@Ay+																	1	0
	MOVY.W Da,@Ay+Iy																	1	1

【注】 Ax : 0 = R4, 1 = R5 Ay : 0 = R6, 1 = R7 Dx : 0 = X0, 1 = X1 Dy : 0 = Y0, 1 = Y1  
Da : 0=A0, 1=A1

表 2.16 シングルデータ転送命令の命令形式

分類	ニーモニック	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
シングル データ 転送	MOVS.W @-As,Ds	1	1	1	1	0	1	As		Ds		0:(*1)		0	0	0	0	
	MOVS.W @As,Ds								0:R4			1:(*1)		0	1			
	MOVS.W @As+,Ds								1:R5			2:(*1)		1	0			
	MOVS.W @As+Is,Ds								2:R2			3:(*1)		1	1			
	MOVS.W Ds,@-As								3:R3			4:(*1)		0	0		1	
	MOVS.W Ds,@As											5:A1		0	1			
	MOVS.W Ds,@As+											6:(*1)		1	0			
	MOVS.W Ds,@As+Is											7:A0		1	1			
	MOVS.L @-As,Ds											8:X0		0	0	1	0	
	MOVS.L @As,Ds											9:X1		0	1			
	MOVS.L @As+,Ds											A:Y0		1	0			
	MOVS.L @As+Is,Ds											B:Y1		1	1			
	MOVS.L Ds,@-As												C:M0		0	0		1
	MOVS.L Ds,@As												D:A1G		0	1		
	MOVS.L Ds,@As+												E:M1		1	0		
	MOVS.L Ds,@As+Is												F:A0G		1	1		

【注】 \*1 システム予約コード

## (2) 並行処理命令

並行処理命令は DSP ユニットを使ったデジタル信号処理を効率よく実行するための命令です。32 ビット長で、同時に並行して 4 つの処理、ALU 演算、乗算、2 つのデータ転送ができます。

並行処理命令は A フィールドと B フィールドに分かれています。A フィールドはデータ転送命令を定義し、B フィールドは ALU 演算命令、乗算命令を定義します。これらの命令は独立に定義することができ、処理は独立に、しかも同時に並行して実行されます。A フィールドの並行データ転送命令を表 2.17 に、B フィールドの ALU 演算命令、乗算命令を表 2.18 に示します。A フィールドの命令は、表 2.15 のダブルデータ転送と同じです。

表 2.17 A フィールドの並行データ転送命令

分類	ニーモニック	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
X メモリ データ 転送	NOPX	1	1	1	1	1	0	0	0	0	0	0	0	0	0																							
	MOVX.W @Ax, Dx							Ax	Dx		0																											
	MOVX.W @Ax+, Dx																																					
	MOVX.W @Ax+lx, Dx																																					
	MOVX.W Da, @Ax									Da		1																										
	MOVX.W Da, @Ax+																																					
	MOVX.W Da, @Ax+lx																																					
Y メモリ データ 転送	NOPY							0	0	0	0	0	0	0	0																							
	MOVY.W @Ay, Dy							Ay	Dy		0																											
	MOVY.W @Ay+, Dy																																					
	MOVY.W @Ay+ly, Dy																																					
	MOVY.W Da, @Ay									Da		1																										
	MOVY.W Da, @Ay+																																					
	MOVY.W Da, @Ay+ly																																					

【注】 Ax: 0=R4, 1=R5 Ay: 0=R6, 1=R7 Dx: 0=X0, 1=X1 Dy: 0=Y0, 1=Y1 Da: 0=A0, 1=A1

2. CPU

表 2.18 B フィールドの ALU 演算命令、乗算命令

分類	ニーモニック	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Imm. シフト	PSHL #Imm, Dz	1	1	1	1	1	0	Aフィールド										0	0	0	0	-16<=Imm<=+16				Dz							
	PSHA #Imm, Dz											0	0	0	1	0	-32<=Imm<=+32																
	予約											0	0	0	0	1																	
6オペランド パラレル 命令	PMULS Sx, Sf, Dg											0	1	0	0	Se	Sf	Sx	Sy	Dg	Du												
	予約											0	1	0	1	0:X0	0:Y0	0:X0	0:Y0	0:M0	0:X0												
	PSUB Sx, Sy, Du											0	1	1	0	1:X1	1:Y1	1:X1	1:Y1	1:M1	1:Y0												
	PMULS Sx, Sf, Dg											0	1	1	0	2:Y0	2:X0	2:A0	2:M0	2:A0	2:A0												
	PADD Sx, Sy, Du											0	1	1	1	3:A1	3:A1	3:A1	3:M1	3:A1	3:A1												
3オペランド 命令	PMULS Sx, Sf, Dg											1	0	0	0	0	0	0	0	0	0	Dz											
	予約											0	1																				
	PSUBC Sx, Sy, Dz											1	0					0:(*1)															
	PADDC Sx, Sy, Dz											1	1					1:(*1)															
	PCMP Sx, Sy											0	0	0	1					2:(*1)													
	予約											0	1					3:(*1)															
	予約											1	0					4:(*1)															
	予約											1	1					5:A1															
	PABS Sx, Dz											0	0	1	0					6:(*1)													
	PRND Sx, Dz											0	1					7:A0															
	PABS Sy, Dz											1	0					8:X0															
	PRND Sy, Dz											1	1					9:X1															
	予約											0	0	1	1					A:Y0													
	予約											0	1					B:Y1															
	予約											1	0					C:M0															
予約											1	1					D:(*1)																
条件付き 3オペランド 命令	[if cc] PSHL Sx, Sy, Dz											0	0	0	0	if cc					E:M1												
	[if cc] PSHA Sx, Sy, Dz											0	1					F:(*1)															
	[if cc] PSUB Sx, Sy, Dz											1	0																				
	[if cc] PADD Sx, Sy, Dz											1	1																				
	予約											0	0	0	1	01: 無条件																	
	[if cc] PAND Sx, Sy, Dz											0	1																				
	[if cc] PXOR Sx, Sy, Dz											1	0																				
	[if cc] POR Sx, Sy, Dz											1	1																				
	[if cc] PDEC Sx, Dz											0	0	1	0	10: DCT																	
	[if cc] PINC Sx, Dz											0	1																				
	[if cc] PDEC Sy, Dz											1	0																				
	[if cc] PINC Sy, Dz											1	1																				
	[if cc] PCLR Dz											0	0	1	1	11: DCF																	
	[if cc] PDMSB Sx, Dz											0	1																				
	予約											1	0																				
	[if cc] PDMSB Sy, Dz											1	1																				
	[if cc] PNEG Sx, Dz											1	1	0	0	1	0																
	[if cc] PCOPY Sx, Dz											0	1																				
	[if cc] PNEG Sy, Dz											1	0																				
	[if cc] PCOPY Sy, Dz											1	1																				
	予約											0	0																				
	[if cc] PSTS MACH, Dz											0	0	1	1	if cc																	
	[if cc] PSTS MACL, Dz											0	1																				
[if cc] PLDS Dz, MACH											1	0																					
[if cc] PLDS Dz, MACL											1	1																					
(*2) 予約											0	0																					
予約											0	*																					
		1	1	1	1	1	1																										

【注】 \*1 システム予約コード

\*2 [if cc] : DCT (DCビット真)、DCF (DCビット偽) またはなし (無条件命令)。

## 2.5 命令セット

本 LSI の命令は 3 つに分けることができます。CPU コアで実行される CPU 命令、DSP ユニットで実行される DSP データ転送命令、DSP 演算命令があります。CPU 命令には DSP の機能をサポートするための命令がいくつかあります。命令セットの説明をそれぞれ 3 つに分けて説明します。

### 2.5.1 CPU 命令の命令セット

CPU 命令を分類別に表 2.19 に示します。

表 2.19 CPU 命令の分類

分類	命令の種類	オペコード	機能	命令数
データ転送命令	5	MOV	データ転送 イミディエイトデータの転送 周辺モジュールデータの転送 構造体データの転送	39
		MOVA	実効アドレスの転送	
		MOV T	T ビットの転送	
		SWAP	上位と下位の交換	
		XTRCT	連結レジスタの中央切り出し	
算術演算命令	21	ADD	2 進加算	33
		ADDC	キャリ付き 2 進加算	
		ADDV	オーバフロー付き 2 進加算	
		CMP/cond	比較	
		DIV1	除算	
		DIV0S	符号付き除算の初期化	
		DIV0U	符号なし除算の初期化	
		DMULS	符号付き倍精度乗算	
		DMULU	符号なし倍精度乗算	
		DT	デクリメントとテスト	
		EXTS	符号拡張	
		EXTU	ゼロ拡張	
		MAC	積和演算、倍精度積和演算	
		MUL	倍精度乗算	
		MULS	符号付き乗算	
		MULU	符号なし乗算	
		NEG	符号反転	
		NEGC	ポロー付き符号反転	
		SUB	2 進減算	
		SUBC	ポロー付き 2 進減算	
		SUBV	アンダフロー付き 2 進減算	
論理演算命令	6	AND	論理積演算	14
		NOT	ビット反転	
		OR	論理和演算	
		TAS	メモリテストとビットセット	
		TST	論理積演算の T ビットセット	
		XOR	排他的論理和演算	

2. CPU

表 2.19 CPU 命令の分類

分類	命令の種類	オペコード	機能	命令数
シフト命令	10	ROTCL	T ビット付き 1 ビット左回転	14
		ROTCR	T ビット付き 1 ビット右回転	
		ROTL	1 ビット左回転	
		ROTR	1 ビット右回転	
		SHAL	算術的 1 ビット左シフト	
		SHAR	算術的 1 ビット右シフト	
		SHLL	論理的 1 ビット左シフト	
		SHLLn	論理的 n ビット左シフト	
		SHLR	論理的 1 ビット右シフト	
		SHLRn	論理的 n ビット右シフト	
		分岐命令	9	
BT	条件分岐、遅延付き条件分岐 (T = 1 で分岐)			
BRA	無条件分岐			
BRAF	無条件分岐			
BSR	サブルーチンプロシージャへの分岐			
BSRF	サブルーチンプロシージャへの分岐			
JMP	無条件分岐			
JSR	サブルーチンプロシージャへの分岐			
RTS	サブルーチンプロシージャからの復帰			
システム制御命令	14	CLRMAC	MAC レジスタのクリア	71
		CLRT	T ビットのクリア	
		LDC	コントロールレジスタへのロード	
		LDRE	繰り返し終了レジスタへのロード	
		LDRS	繰り返し開始レジスタへのロード	
		LDS	システムレジスタへのロード	
		NOP	無操作	
		RTE	例外処理からの復帰	
		SETRC	繰り返し回数の設定	
		SETT	T ビットのセット	
		SLEEP	低消費電力モードへの遷移	
		STC	コントロールレジスタからのストア	
		STS	システムレジスタからのストア	
		TRAPA	トラップ例外処理	
	計 65			計 182

CPU 命令の命令コード、動作、実行ステートを、以下の形式で分類別に説明します。

命令	命令コード	動作	実行 ステート	Tビット
<p>ニーモニックで表示しています。</p> <p>記号の説明</p> <p>OP: Sz SRC、DEST OP: オペコード Sz: サイズ SRC: ソース DEST: デスティネーション Rm: ソースレジスタ Rn: デスティネーションレジスタ imm: イミディエイトデータ disp: ディスプレースメント *2</p>	<p>MSB LSB の順で表示しています。</p> <p>記号の説明</p> <p>mmmm: ソースレジスタ nnnn: デスティネーションレジスタ 0000: R0 0001: R1 ..... 1111: R15 iiii: イミディエイトデータ dddd: ディスプレースメント</p>	<p>動作の概略を表示しています。</p> <p>記号の説明</p> <p>、 : 転送方向 (xx): メモリオペランド M/Q/T: SR 内のフラグビット &amp; : ビット毎の論理積   : ビット毎の論理和 ^ : ビット毎の排他的論理和 ~ : ビット毎の論理否定 &lt;&lt; n: 左 n ビットシフト &gt;&gt; n: 右 n ビットシフト</p>	<p>ノーウェイトのときの値です。*1</p>	<p>命令実行後の、Tビットの値を表示していません。</p> <p>記号の説明 : 変化 しない</p>

- 【注】 \*1 命令の実行ステートについて  
表に示した実行ステートは最少値です。実際は、
- (1) 命令フェッチとデータアクセスの競合が起こる場合
  - (2) ロード命令(メモリ レジスタ)のデスティネーションレジスタと、その直後の命令が使うレジスタが同一な場合
- などの条件により、命令実行ステート数は増加します。
- \*2 命令のオペランドサイズなどに応じてスケールリング(× 1、× 2、× 4)されます。  
詳細は「SH-1/SH-2/SH-DSP プログラミングマニュアル」を参照してください。

## 2. CPU

### (1) データ転送命令

命令	命令コード	動作	実行 ステート	Tビット
MOV #imm,Rn	1110nnnniiiiiii	imm 符号拡張 Rn	1	
MOV.W @(disp,PC),Rn	1001nnnnddddddd	(disp × 2 + PC) 符号拡張 Rn	1	
MOV.L @(disp,PC),Rn	1101nnnnddddddd	(disp × 4 + PC) Rn	1	
MOV Rm,Rn	0110nnnnmmmm0011	Rm Rn	1	
MOV.B Rm,@Rn	0010nnnnmmmm0000	Rm (Rn)	1	
MOV.W Rm,@Rn	0010nnnnmmmm0001	Rm (Rn)	1	
MOV.L Rm,@Rn	0010nnnnmmmm0010	Rm (Rn)	1	
MOV.B @Rm,Rn	0110nnnnmmmm0000	(Rm) 符号拡張 Rn	1	
MOV.W @Rm,Rn	0110nnnnmmmm0001	(Rm) 符号拡張 Rn	1	
MOV.L @Rm,Rn	0110nnnnmmmm0010	(Rm) Rn	1	
MOV.B Rm,@-Rn	0010nnnnmmmm0100	Rn - 1 Rn, Rm (Rn)	1	
MOV.W Rm,@-Rn	0010nnnnmmmm0101	Rn - 2 Rn, Rm (Rn)	1	
MOV.L Rm,@-Rn	0010nnnnmmmm0110	Rn - 4 Rn, Rm (Rn)	1	
MOV.B @Rm+,Rn	0110nnnnmmmm0100	(Rm) 符号拡張 Rn, Rm + 1 Rm	1	
MOV.W @Rm+,Rn	0110nnnnmmmm0101	(Rm) 符号拡張 Rn, Rm + 2 Rm	1	
MOV.L @Rm+,Rn	0110nnnnmmmm0110	(Rm) Rn, Rm + 4 Rm	1	
MOV.B R0,@(disp,Rn)	10000000nnnnddd	R0 (disp + Rn)	1	
MOV.W R0,@(disp,Rn)	10000001nnnnddd	R0 (disp × 2 + Rn)	1	
MOV.L Rm,@(disp,Rn)	0001nnnnmmmmddd	Rm (disp × 4 + Rn)	1	
MOV.B @(disp,Rm),R0	10000100mmmmddd	(disp + Rm) 符号拡張 R0	1	
MOV.W @(disp,Rm),R0	10000101mmmmddd	(disp × 2 + Rm) 符号拡張 R0	1	
MOV.L @(disp,Rm),Rn	0101nnnnmmmmddd	(disp × 4 + Rm) Rn	1	
MOV.B Rm,@(R0,Rn)	0000nnnnmmmm0100	Rm (R0 + Rn)	1	
MOV.W Rm,@(R0,Rn)	0000nnnnmmmm0101	Rm (R0 + Rn)	1	
MOV.L Rm,@(R0,Rn)	0000nnnnmmmm0110	Rm (R0 + Rn)	1	
MOV.B @(R0,Rm),Rn	0000nnnnmmmm1100	(R0 + Rm) 符号拡張 Rn	1	
MOV.W @(R0,Rm),Rn	0000nnnnmmmm1101	(R0 + Rm) 符号拡張 Rn	1	
MOV.L @(R0,Rm),Rn	0000nnnnmmmm1110	(R0 + Rm) Rn	1	
MOV.B R0,@(disp,GBR)	11000000ddddddd	R0 (disp + GBR)	1	
MOV.W R0,@(disp,GBR)	11000001ddddddd	R0 (disp × 2 + GBR)	1	
MOV.L R0,@(disp,GBR)	11000010ddddddd	R0 (disp × 4 + GBR)	1	
MOV.B @(disp,GBR),R0	11000100ddddddd	(disp + GBR) 符号拡張 R0	1	
MOV.W @(disp,GBR),R0	11000101ddddddd	(disp × 2 + GBR) 符号拡張 R0	1	
MOV.L @(disp,GBR),R0	11000110ddddddd	(disp × 4 + GBR) R0	1	
MOVA @(disp,PC),R0	11000111ddddddd	disp × 4 + PC R0	1	
MOVT Rn	0000nnnn00101001	T Rn	1	
SWAP.B Rm,Rn	0110nnnnmmmm1000	Rm 下位2バイトの上下バイト交換 Rn	1	
SWAP.W Rm,Rn	0110nnnnmmmm1001	Rm 上下ワード交換 Rn	1	
XTRCT Rm,Rn	0010nnnnmmmm1101	Rm と Rn の中央 32 ビット Rn	1	

## (2) 算術演算命令

命令	命令コード	動作	実行 ステート	Tビット
ADD Rm,Rn	0011nnnnmmmm1100	Rn + Rm Rn	1	
ADD #imm,Rn	0111nnnniiiiiii	Rn + imm Rn	1	
ADDC Rm,Rn	0011nnnnmmmm1110	Rn + Rm + T Rn, キャリ T	1	キャリ
ADDV Rm,Rn	0011nnnnmmmm1111	Rn + Rm Rn, オーバフロー T	1	オーバフロー
CMP/EQ #imm,R0	10001000iiiiiii	R0 = imm のとき 1 T	1	比較結果
CMP/EQ Rm,Rn	0011nnnnmmmm0000	Rn = Rm のとき 1 T	1	比較結果
CMP/HS Rm,Rn	0011nnnnmmmm0010	無符号で Rn > Rm のとき 1 T	1	比較結果
CMP/GE Rm,Rn	0011nnnnmmmm0011	有符号で Rn > Rm のとき 1 T	1	比較結果
CMP/HI Rm,Rn	0011nnnnmmmm0110	無符号で Rn > Rm のとき 1 T	1	比較結果
CMP/GT Rm,Rn	0011nnnnmmmm0111	有符号で Rn > Rm のとき 1 T	1	比較結果
CMP/PL Rn	0100nnnn00010101	Rn > 0 のとき 1 T	1	比較結果
CMP/PZ Rn	0100nnnn00010001	Rn = 0 のとき 1 T	1	比較結果
CMP/STR Rm,Rn	0010nnnnmmmm1100	いずれかのバイトが等しいとき 1 T	1	比較結果
DIV1 Rm,Rn	0011nnnnmmmm0100	1ステップ除算 (Rn ÷ Rm)	1	計算結果
DIV0S Rm,Rn	0010nnnnmmmm0111	Rn の MSB Q, Rm の MSB M, M^Q T	1	計算結果
DIV0U	000000000011001	0 M/Q/T	1	0
DMULS.L Rm,Rn	0011nnnnmmmm1101	符号付きで Rn × Rm MACH,MACL 32 × 32 64 ビット	2 ~ 4* <sup>1</sup>	
DMULU.L Rm,Rn	0011nnnnmmmm0101	符号なしで Rn × Rm MACH,MACL 32 × 32 64 ビット	2 ~ 4* <sup>1</sup>	
DT Rn	0100nnnn00010000	Rn - 1 Rn, Rn が 0 のとき 1 T Rn が 0 以外のとき 0 T	*1	比較結果
EXTS.B Rm,Rn	0110nnnnmmmm1110	Rm をバイトから符号拡張 Rn	1	
EXTS.W Rm,Rn	0110nnnnmmmm1111	Rm をワードから符号拡張 Rn	1	
EXTU.B Rm,Rn	0110nnnnmmmm1100	Rm をバイトからゼロ拡張 Rn	1	
EXTU.W Rm,Rn	0110nnnnmmmm1101	Rm をワードからゼロ拡張 Rn	1	
MAC.L @Rm+,@ Rn+	0000nnnnmmmm1111	符号付きで (Rn) × (Rm) + MAC MAC 32 × 32 + 64 64 ビット	3/(2 ~ 4)* <sup>1</sup>	
MAC.W @Rm+,@ Rn+	0100nnnnmmmm1111	符号付きで (Rn) × (Rm) + MAC MAC 16 × 16 + 64 64 ビット	3/(2) * <sup>1</sup>	
MUL.L Rm,Rn	0000nnnnmmmm0111	Rn × Rm MACL 32 × 32 32 ビット	2 ~ 4* <sup>1</sup>	
MULS.W Rm,Rn	0010nnnnmmmm1111	符号付きで Rn × Rm MAC 16 × 16 32 ビット	1 ~ 3* <sup>1</sup>	
MULU.W Rm,Rn	0010nnnnmmmm1110	符号なしで Rn × Rm MAC 16 × 16 32 ビット	1 ~ 3* <sup>1</sup>	
NEG Rm,Rn	0110nnnnmmmm1011	0 - Rm Rn	1	
NEGC Rm,Rn	0110nnnnmmmm1010	0 - Rm - T Rn, ボロー T	1	ボロー
SUB Rm,Rn	0011nnnnmmmm1000	Rn - Rm Rn	1	
SUBC Rm,Rn	0011nnnnmmmm1010	Rn - Rm - T Rn, ボロー T	1	ボロー
SUBV Rm,Rn	0011nnnnmmmm1011	Rn - Rm Rn, アンダフロー T	1	アンダ フロー

【注】 \*1 通常実行ステートを示します。( ) 内の値は前後の命令との競合関係による実行ステートです。

## 2. CPU

### (3) 論理演算命令

命令	命令コード	動作	実行 ステート	Tビット
AND Rm,Rn	0010nnnnmmmm1001	Rn & Rm Rn	1	
AND #imm,R0	11001001iiiiiii	R0 & imm R0	1	
AND.B #imm,@(R0,GBR)	11001101iiiiiii	(R0 + GBR) & imm (R0 + GBR)	3	
NOT Rm,Rn	0110nnnnmmmm0111	~Rm Rn	1	
OR Rm,Rn	0010nnnnmmmm1011	Rn   Rm Rn	1	
OR #imm,R0	11001011iiiiiii	R0   imm R0	1	
OR.B #imm,@(R0,GBR)	11001111iiiiiii	(R0 + GBR)   imm (R0 + GBR)	3	
TAS.B @Rn	0100nnnn00011011	(Rn) が 0 のとき 1 T, 1 MSBof(Rn)	4	テスト結果
TST Rm,Rn	0010nnnnmmmm1000	Rn & Rm, 結果が 0 のとき 1 T	1	テスト結果
TST #imm,R0	11001000iiiiiii	R0 & imm, 結果が 0 のとき 1 T	1	テスト結果
TST.B #imm,@(R0,GBR)	11001100iiiiiii	(R0 + GBR) & imm, 結果が 0 のとき 1 T	3	テスト結果
XOR Rm,Rn	0010nnnnmmmm1010	Rn ^ Rm Rn	1	
XOR #imm,R0	11001010iiiiiii	R0 ^ imm R0	1	
XOR.B #imm,@(R0,GBR)	11001110iiiiiii	(R0 + GBR) ^ imm (R0 + GBR)	3	

### (4) シフト命令

命令	命令コード	動作	実行 ステート	Tビット
ROTL Rn	0100nnnn00000100	T Rn MSB	1	MSB
ROTR Rn	0100nnnn00000101	LSB Rn T	1	LSB
ROTCL Rn	0100nnnn00100100	T Rn T	1	MSB
ROTCR Rn	0100nnnn00100101	T Rn T	1	LSB
SHAL Rn	0100nnnn00100000	T Rn 0	1	MSB
SHAR Rn	0100nnnn00100001	MSB Rn T	1	LSB
SHLL Rn	0100nnnn00000000	T Rn 0	1	MSB
SHLR Rn	0100nnnn00000001	0 Rn T	1	LSB
SHLL2 Rn	0100nnnn00001000	Rn << 2 Rn	1	
SHLR2 Rn	0100nnnn00001001	Rn >> 2 Rn	1	
SHLL8 Rn	0100nnnn00011000	Rn << 8 Rn	1	
SHLR8 Rn	0100nnnn00011001	Rn >> 8 Rn	1	
SHLL16 Rn	0100nnnn00101000	Rn << 16 Rn	1	
SHLR16 Rn	0100nnnn00101001	Rn >> 16 Rn	1	

### (5) 分岐命令

命令	命令コード	動作	実行 ステート	Tビット
BF label	10001011dddddddd	T = 0 のとき disp × 2 + PC PC, T = 1 のとき nop	3/1* <sup>2</sup>	
BF/S label	10001111dddddddd	遅延分岐、T = 0 のとき disp × 2 + PC PC, T = 1 のとき nop	2/1* <sup>2</sup>	

命令	命令コード	動作	実行 ステート	Tビット
BT label	10001001ddddddddd	T = 1 のとき disp × 2 + PC PC, T = 0 のとき nop	3/1*2	
BT/S label	10001101ddddddddd	遅延分岐、T = 1 のとき disp × 2 + PC PC, T = 0 のとき nop	2/1*2	
BRA label	1010ddddddddd	遅延分岐、disp × 2 + PC PC	2	
BRAF Rm	0000mmmm00100011	遅延分岐、Rm + PC PC	2	
BSR label	1011ddddddddd	遅延分岐、PC PR, disp × 2 + PC PC	2	
BSRF Rm	0000mmmm00000011	遅延分岐、PC PR, Rm + PC PC	2	
JMP @Rm	0100mmmm00101011	遅延分岐、Rm PC	2	
JSR @Rm	0100mmmm00001011	遅延分岐、PC PR, Rm PC	2	
RTS	0000000000001011	遅延分岐、PR PC	2	

【注】 \*2 分岐しないときは1ステートになります。

#### (6) システム制御命令

命令	命令コード	動作	実行 ステート	Tビット
CLRMACH	0000000000101000	0 MACH、MACL	1	
CLRT	0000000000001000	0 T	1	0
LDC Rm,SR	0100mmmm00001110	Rm SR	1	LSB
LDC Rm,GBR	0100mmmm00011110	Rm GBR	1	
LDC Rm,VBR	0100mmmm00101110	Rm VBR	1	
LDC Rm,MOD	0100mmmm01011110	Rm MOD	1	
LDC Rm,RE	0100mmmm01111110	Rm RE	1	
LDC Rm,RS	0100mmmm01101110	Rm RS	1	
LDC.L @Rm+,SR	0100mmmm00000111	(Rm) SR、Rm + 4 Rm	3	LSB
LDC.L @Rm+,GBR	0100mmmm00010111	(Rm) GBR、Rm + 4 Rm	3	
LDC.L @Rm+,VBR	0100mmmm00100111	(Rm) VBR、Rm + 4 Rm	3	
LDC.L @Rm+,MOD	0100mmmm01010111	(Rm) MOD、Rm + 4 Rm	3	
LDC.L @Rm+,RE	0100mmmm01110111	(Rm) RE、Rm + 4 Rm	3	
LDC.L @Rm+,RS	0100mmmm01100111	(Rm) RS、Rm + 4 Rm	3	
LDRE @(disp,PC)	10001110ddddddd	disp × 2 + PC RE	1	
LDRS @(disp,PC)	10001100ddddddd	disp × 2 + PC RS	1	
LDS Rm,MACH	0100mmmm00001010	Rm MACH	1	
LDS Rm,MACL	0100mmmm00011010	Rm MACL	1	
LDS Rm,PR	0100mmmm00101010	Rm PR	1	
LDS Rm,DSR	0100mmmm01101010	Rm DSR	1	
LDS Rm,A0	0100mmmm01111010	Rm A0	1	
LDS Rm,X0	0100mmmm10001010	Rm X0	1	
LDS Rm,X1	0100mmmm10011010	Rm X1	1	
LDS Rm,Y0	0100mmmm10101010	Rm Y0	1	
LDS Rm,Y1	0100mmmm10111010	Rm Y1	1	

2. CPU

命令	命令コード	動作	実行 ステート	Tビット
LDS.L @Rm+,MAC H	0100mmmm00000110	(Rm) MACH、Rm + 4 Rm	1	
LDS.L @Rm+,MAC L	0100mmmm00010110	(Rm) MACL、Rm + 4 Rm	1	
LDS.L @Rm+,PR	0100mmmm00100110	(Rm) PR、Rm + 4 Rm	1	
LDS.L @Rm+,DSR	0100mmmm01100110	(Rm) DSR、Rm + 4 Rm	1	
LDS.L @Rm+,A0	0100mmmm01110110	(Rm) A0、Rm + 4 Rm	1	
LDS.L @Rm+,X0	0100mmmm10000110	(Rm) X0、Rm + 4 Rm	1	
LDS.L @Rm+,X1	0100mmmm10010110	(Rm) X1、Rm + 4 Rm	1	
LDS.L @Rm+,Y0	0100mmmm10100110	(Rm) Y0、Rm + 4 Rm	1	
LDS.L @Rm+,Y1	0100mmmm10110110	(Rm) Y1、Rm + 4 Rm	1	
NOP	0000000000001001	無操作	1	
RTE	000000000101011	遅延分岐、スタック領域 PC/SR	4	LSB
SETRC Rm	0100mmmm00010100	RE - RSの演算結果(リピート状態) RF1、RF0 Rm[11:0] RC(SR[27:16])	1	
SETRC #imm	10000010iiiiiiii	RE - RSの演算結果(リピート状態) RF1、RF0 imm RC(SR[23:16])、0 SR[27:24]	1	1
SETT	0000000000011000	1 T	1	1
SLEEP	0000000000011011	スリープ	3*3	
STC SR,Rn	0000nnnn00000010	SR Rn	1	
STC GBR,Rn	0000nnnn00010010	GBR Rn	1	
STC VBR,Rn	0000nnnn00100010	VBR Rn	1	
STC MOD,Rn	0000nnnn01010010	MOD Rn	1	
STC RE,Rn	0000nnnn01110010	RE Rn	1	
STC RS,Rn	0000nnnn01100010	RS Rn	1	
STC.L SR,@-Rn	0100nnnn00000011	Rn-4 Rn、SR (Rn)	2	
STC.L GBR,@-Rn	0100nnnn00010011	Rn-4 Rn、GBR (Rn)	2	
STC.L VBR,@-Rn	0100nnnn00100011	Rn-4 Rn、VBR (Rn)	2	
STC.L MOD,@-Rn	0100nnnn01010011	Rn - 4 Rn、MOD (Rn)	2	
STC.L RE,@-Rn	0100nnnn01110011	Rn - 4 Rn、RE (Rn)	2	
STC.L RS,@-Rn	0100nnnn01100011	Rn - 4 Rn、RS (Rn)	2	
STS MACH,Rn	0000nnnn00001010	MACH Rn	1	
STS MACL,Rn	0000nnnn00011010	MACL Rn	1	
STS PR,Rn	0000nnnn00101010	PR Rn	1	
STS DSR,Rn	0000nnnn01101010	DSR Rn	1	
STS A0,Rn	0000nnnn01111010	A0 Rn	1	
STS X0,Rn	0000nnnn10001010	X0 Rn	1	
STS X1,Rn	0000nnnn10011010	X1 Rn	1	
STS Y0,Rn	0000nnnn10101010	Y0 Rn	1	
STS Y1,Rn	0000nnnn10111010	Y1 Rn	1	
STS.L MACH,@- Rn	0100nnnn00000010	Rn-4 Rn、MACH (Rn)	1	

命令	命令コード	動作	実行 ステート	Tビット
STS.L MACL,@-Rn	0100nnnn00010010	Rn-4 Rn, MACL (Rn)	1	
STS.L PR,@-Rn	0100nnnn00100010	Rn-4 Rn, PR (Rn)	1	
STS.L DSR,@-Rn	0100nnnn01100010	Rn - 4 Rn, DSR (Rn)	1	
STS.L A0,@-Rn	0100nnnn01110010	Rn - 4 Rn, A0 (Rn)	1	
STS.L X0,@-Rn	0100nnnn10000010	Rn - 4 Rn, X0 (Rn)	1	
STS.L X1,@-Rn	0100nnnn10010010	Rn - 4 Rn, X1 (Rn)	1	
STS.L Y0,@-Rn	0100nnnn10100010	Rn - 4 Rn, Y0 (Rn)	1	
STS.L Y1,@-Rn	0100nnnn10110010	Rn - 4 Rn, Y1 (Rn)	1	
TRAPA #imm	11000011iiiiiiii	PC/SR スタック領域、(imm ´ 4+VBR) PC	8	

【注】 \*3 スリープ状態に遷移するまでのステート数です。

#### [ 注意事項 ]

命令の実行ステートについて

表に示した実行ステートは最少値です。実際は、

- (1) 命令フェッチとデータアクセスの競合が起こる場合
- (2) ロード命令（メモリ レジスタ）のデスティネーションレジスタと、その直後の命令が使うレジスタが同一な場合
- (3) 分岐命令の分岐先アドレスが  $4n+2$  番地などの条件により、命令実行ステート数は増加します。

#### (7) DSP 機能をサポートする CPU 命令

DSP 機能をサポートするために CPU コア命令にいくつかのシステム制御命令が追加されました。繰り返し制御、モジュロアドレッシングをサポートする RS、RE、MOD レジスタが追加され、ステータスレジスタ (SR) にはリピートカウンタ (RC) が追加されました。これらにアクセスするため、LDC、STC 命令が追加されました。さらに DSP レジスタの DSR、A0、X0、X1、Y0、Y1 レジスタにアクセスするために、LDS、STS 命令が追加されました。

SR レジスタのリピートカウンタ (RC、ビット 27 ~ 16) およびリピートフラグ (RF1、RF0、ビット 3、2) に値を設定する SETRC 命令が追加されました。SETRC 命令のオペランドがイミディエイトのときは、8 ビットのイミディエイトデータが SR のビット 23 ~ 16 に格納され、ビット 27 ~ 24 は 0 にクリアされます。オペランドがレジスタのときは、レジスタのビット 11 ~ 0 の 12 ビットが SR のビット 27 ~ 16 に格納されます。さらに、RS、RE の設定値から、1 命令リピート (00)、2 命令リピート (01)、3 命令リピート (11)、4 命令以上 (10) のいずれかの状態がセットされます。

繰り返し開始アドレス、繰り返し終了アドレスを RS、RE レジスタに設定する命令には、LDC 命令の他に、LDRS、LDRE 命令を追加しました。

追加された命令を表 2.20 に示します。

## 2. CPU

表 2.20 追加された CPU 命令

命令	命令コード	動作	実行 ステート	T ビット
LDC Rm,MOD	0100mmmm01011110	Rm MOD	1	
LDC Rm,RE	0100mmmm01111110	Rm RE	1	
LDC Rm,RS	0100mmmm01101110	Rm RS	1	
LDC.L @Rm+,MOD	0100mmmm01010111	(Rm) MOD, Rm + 4 Rm	3	
LDC.L @Rm+,RE	0100mmmm01110111	(Rm) RE, Rm + 4 Rm	3	
LDC.L @Rm+,RS	0100mmmm01100111	(Rm) RS, Rm + 4 Rm	3	
STC MOD,Rn	0000nnnn01010010	MOD Rn	1	
STC RE,Rn	0000nnnn01110010	RE Rn	1	
STC RS,Rn	0000nnnn01100010	RS Rn	1	
STC.L MOD,@-Rn	0100nnnn01010011	Rn - 4 Rn, MOD (Rn)	2	
STC.L RE,@-Rn	0100nnnn01110011	Rn - 4 Rn, RE (Rn)	2	
STC.L RS,@-Rn	0100nnnn01100011	Rn - 4 Rn, RS (Rn)	2	
LDS Rm,DSR	0100mmmm01101010	Rm DSR	1	
LDS.L @Rm+,DSR	0100mmmm01100110	(Rm) DSR, Rm + 4 Rm	1	
LDS Rm,A0	0100mmmm01111010	Rm A0	1	
LDS.L @Rm+,A0	0100mmmm01110110	(Rm) A0, Rm + 4 Rm	1	
LDS Rm,X0	0100mmmm10001010	Rm X0	1	
LDS.L @Rm+,X0	0100mmmm10000110	(Rm) X0, Rm + 4 Rm	1	
LDS Rm,X1	0100mmmm10011010	Rm X1	1	
LDS.L @Rm+,X1	0100mmmm10010110	(Rm) X1, Rm + 4 Rm	1	
LDS Rm,Y0	0100mmmm10101010	Rm Y0	1	
LDS.L @Rm+,Y0	0100mmmm10100110	(Rm) Y0, Rm + 4 Rm	1	
LDS Rm,Y1	0100mmmm10111010	Rm Y1	1	
LDS.L @Rm+,Y1	0100mmmm10110110	(Rm) Y1, Rm + 4 Rm	1	
STS DSR,Rn	0000nnnn01101010	DSR Rn	1	
STS.L DSR,@-Rn	0100nnnn01100010	Rn - 4 Rn, DSR (Rn)	1	
STS A0,Rn	0000nnnn01111010	A0 Rn	1	
STS.L A0,@-Rn	0100nnnn01110010	Rn - 4 Rn, A0 (Rn)	1	
STS X0,Rn	0000nnnn10001010	X0 Rn	1	
STS.L X0,@-Rn	0100nnnn10000010	Rn - 4 Rn, X0 (Rn)	1	
STS X1,Rn	0000nnnn10011010	X1 Rn	1	
STS.L X1,@-Rn	0100nnnn10010010	Rn - 4 Rn, X1 (Rn)	1	
STS Y0,Rn	0000nnnn10101010	Y0 Rn	1	
STS.L Y0,@-Rn	0100nnnn10100010	Rn - 4 Rn, Y0 (Rn)	1	
STS Y1,Rn	0000nnnn10111010	Y1 Rn	1	
STS.L Y1,@-Rn	0100nnnn10110010	Rn - 4 Rn, Y1 (Rn)	1	
SETRC Rm	0100mmmm00010100	Rm[11:0] RC (SR[27:16])	1	
SETRC #imm	10000010iiiiiii	imm RC(SR[23:16]),0 SR[27:24]	1	
LDRS @(disp,PC)	10001100ddddddd	disp × 2+PC RS	1	
LDRE @(disp,PC)	10001110ddddddd	disp × 2+PC RE	1	

## 2.5.2 DSP データ転送命令の命令セット

DSP データ転送命令を分類別に表 2.21 に示します。

表 2.21 DSP データ転送命令の分類

分類	命令の種類	オペコード	機能	命令数
ダブルデータ転送命令	4	NOPX	X メモリ無操作	14
		MOVX	X メモリデータ転送	
		NOPY	Y メモリ無操作	
		MOVY	Y メモリデータ転送	
シングルデータ転送命令	1	MOVS	シングルデータ転送	16
	計 5			計 30

データ転送命令は 2 つのグループに分けられます。ダブルデータ転送とシングルデータ転送です。ダブルデータ転送は DSP 演算命令と組み合わせて、DSP 並行処理命令とすることができます。並行処理命令は 32 ビット長で、A フィールドにダブルデータ転送命令が組み込まれます。並行処理命令でないダブルデータ転送とシングルデータ転送命令は 16 ビット長です。

ダブルデータ転送では X メモリと Y メモリを同時に並行してアクセスできます。それぞれ X、Y メモリデータアクセスから一つずつ命令を指定します。Ax ポインタは X メモリをアクセスするために使い、Ay ポインタは Y メモリをアクセスするために使います。ダブルデータ転送は X、Y メモリだけをアクセスできます。

シングルデータ転送はどこのエリアからでもアクセスできます。シングルデータ転送では Ax ポインタとその他の 2 つのポインタを As ポインタとして使います。

### (1) ダブルデータ転送命令 (X メモリデータ)

命令	動作	命令コード	実行 ステート	DC ビット
NOPX	No Operation	1111000*0*0*00**	1	
MOVX.W @Ax,Dx	(Ax) MSW of Dx, 0 LSW of Dx	111100A*D*0*01**	1	
MOVX.W @Ax+,Dx	(Ax) MSW of Dx, 0 LSW of Dx, Ax + 2 Ax	111100A*D*0*10**	1	
MOVX.W @Ax+lx,Dx	(Ax) MSW of Dx, 0 LSW of Dx, Ax + lx Ax	111100A*D*0*11**	1	
MOVX.W Da,@Ax	MSW of Da (Ax)	111100A*D*1*01**	1	
MOVX.W Da,@Ax+	MSW of Da (Ax), Ax + 2 Ax	111100A*D*1*10**	1	
MOVX.W Da,@Ax+lx	MSW of Da (Ax), Ax + lx Ax	111100A*D*1*11**	1	

### (2) ダブルデータ転送命令 (Y メモリデータ)

命令	動作	命令コード	実行 ステート	DC ビット
NOPY	No Operation	111100*0*0*0**00	1	
MOVY.W @Ay,Dy	(Ay) MSW of Dy, 0 LSW of Dy	111100*A*D*0**01	1	
MOVY.W @Ay+,Dy	(Ay) MSW of Dy, 0 LSW of Dy, Ay + 2 Ay	111100*A*D*0**10	1	

## 2. CPU

命令	動作	命令コード	実行 ステート	DC ビット
MOVY.W @Ay+ly,Dy	(Ay) MSW of Dy、0 LSW of Dy、 Ay + ly Ay	111100*A*D*0**11	1	
MOVY.W Da,@Ay	MSW of Da (Ay)	111100*A*D*1**01	1	
MOVY.W Da,@Ay+	MSW of Da (Ay)、Ay + 2 Ay	111100*A*D*1**10	1	
MOVY.W Da,@Ay+ly	MSW of Da (Ay)、Ay + ly Ay	111100*A*D*1**11	1	

### (3) シングルデータ転送命令

命令	動作	命令コード	実行 ステート	DC ビット
MOVS.W @-As,Ds	As - 2 As、(As) MSW of Ds、0 LSW of Ds	111101AADDDD0000	1	
MOVS.W @As,Ds	(As) MSW of Ds、0 LSW of Ds	111101AADDDD0100	1	
MOVS.W @As+,Ds	(As) MSW of Ds、0 LSW of Ds、 As + 2 As	111101AADDDD1000	1	
MOVS.W @As+lx,Ds	(As) MSW of Ds、0 LSW of Ds、 As + lx As	111101AADDDD1100	1	
MOVS.W Ds,@-As	As - 2 As、MSW of Ds (As)*	111101AADDDD0001	1	
MOVS.W Ds,@As	MSW of Ds (As)*	111101AADDDD0101	1	
MOVS.W Ds,@As+	MSW of Ds (As)*、As + 2 As	111101AADDDD1001	1	
MOVS.W Ds,@As+ls	MSW of Ds (As)*、As + ls As	111101AADDDD1101	1	
MOVS.L @-As,Ds	As - 4 As、(As) Ds	111101AADDDD0010	1	
MOVS.L @As,Ds	(As) Ds	111101AADDDD0110	1	
MOVS.L @As+,Ds	(As) Ds、As + 4 As	111101AADDDD1010	1	
MOVS.L @As+ls,Ds	(As) Ds、As + ls As	111101AADDDD1110	1	
MOVS.L Ds,@-As	As - 4 As、Ds (As)*	111101AADDDD0011	1	
MOVS.L Ds,@As	Ds (As)*	111101AADDDD0111	1	
MOVS.L Ds,@As+	Ds (As)*、As + 4 As	111101AADDDD1011	1	
MOVS.L Ds,@As+ls	Ds (As)*、As + ls As	111101AADDDD1111	1	

【注】 \* ソースオペランド Ds にガードビットレジスタ A0G、A1G を指定した場合は、データは符号拡張され、転送され  
ません。

DSP データ転送のオペランドとレジスタとの対応を表 2.22 に示します。CPU コアのレジスタはメモリアドレスを示すポインタアドレスとして使われます。

表 2.22 DSP データ転送のオペランドとレジスタとの対応

オペランド	SH (CPU コア) レジスタ									
	R0	R1	R2(As2)	R3(As3)	R4(Ax0, As0)	R5(Ax1, As1)	R6(Ay0)	R7(Ay1)	R8(lx,ls)	R9(ly)
Ax					Yes	Yes				
lx, (ls)									Yes	
Dx										
Ay							Yes	Yes		
ly										Yes
Dy										
Da										
As			Yes	Yes	Yes	Yes				
Ds										

オペランド	DSP レジスタ									
	X0	X1	Y0	Y1	M0	M1	A0	A1	A0G	A1G
Ax										
lx, (ls)										
Dx	Yes	Yes								
Ay										
ly										
Dy			Yes	Yes						
Da							Yes	Yes		
As										
Ds	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

【注】 Yes : 設定可能なレジスタ

### 2.5.3 DSP 演算命令の命令セット

DSP 演算命令は DSP ユニットで処理されるデジタル信号処理の命令です。これらの命令は 32 ビット長の命令コードで、複数の命令を並列に実行します。命令コードは A フィールド、B フィールドの 2 つに分かれており、A フィールドにはパラレルデータ転送命令を指定し、B フィールドにはシングルまたはダブルデータ演算命令を指定します。命令は独立して指定することができ、実行も独立に実行されます。A フィールドに指定するパラレルデータ転送命令はダブルデータ転送命令と全く同じです。

B フィールドのデータ演算命令は 3 つに分かれています。ダブルデータ演算命令、条件付きシングルデータ演算命令、無条件シングルデータ演算命令の 3 つです。DSP 演算命令の命令形式を表 2.23 に示します。それぞれオペランドは独立に DSP レジスタから選べます。DSP 演算命令のオペランドとレジスタの対応を表 2.24 に示します。

表 2.23 DSP 演算命令の命令形式

分類	命令形式	命令	
ダブルデータ演算命令 (6 オペランド)		ALUop. Sx, Sy, Du MLTop. Se, Sf, Dg	PADD PMULS, PSUB PMUL
条件付き シングルデータ 演算命令	3 オペランド	ALUop. Sx, Sy, Dz DCT ALUop. Sx, Sy, Dz DCF ALUop. Sx, Sy, Dz	PADD, PAND, POR, PSHA, PSHL, PSUB, PXOR
	2 オペランド	ALUop. Sx, Dz DCT ALUop. Sx, Dz DCF ALUop. Sx, Dz ALUop. Sy, Dz DCT ALUop. Sy, Dz DCF ALUop. Sy, Dz	PCOPY, PDEC, PDMSB, PINC, PLDS, PSTS, PNEG
	1 オペランド	ALUop. Dz DCT ALUop. Dz DCF ALUop. Dz	PCLR, PSHA #imm, PSHL #imm
無条件 シングルデータ 演算命令	3 オペランド	ALUop. Sx, Sy, Du MLTop. Se, Sf, Dg	PADDC, PSUBC, PMULS
	2 オペランド	ALUop. Sx, Dz ALUop. Sy, Dz ALUop. Sx, Sy	PCMP, PABS, PRND
	1 オペランド	ALUop. Dz	PSHA #imm, PSHL #imm

表 2.24 DSP 命令のオペランドとレジスタの対応

レジスタ	ALU、BPU 命令				乗算命令		
	Sx	Sy	Dz	Du	Se	Sf	Dg
A0	Yes		Yes	Yes			Yes
A1	Yes		Yes	Yes	Yes	Yes	Yes
M0		Yes	Yes				Yes
M1		Yes	Yes				Yes
X0	Yes		Yes	Yes	Yes	Yes	
X1	Yes		Yes		Yes		
Y0		Yes	Yes	Yes	Yes	Yes	
Y1		Yes	Yes			Yes	

並行命令を書くときは最初に B フィールドの命令を書いて、次に A フィールドの命令を書きます。並行処理プログラム例を図 2.13 に示します。

PADD A0,M0,A0	PMULS X0,Y0,M0	MOVX.W @R4+,X0	MOVY.W @R6+,Y0[:]
DCF PINC X1,A1		MOVX.W A0,@R5+R8	MOVY.W @R7+,Y0[:]
PCMP X1,M0		MOVX.W @R4+R8	[NOPY][:]

図 2.13 並行処理プログラム例

ここで [ ] は省略可能を意味します。無操作命令 NOPX、NOPY は省略できます。';' は命令行の区切りですが、省略できます。もし区切り ';' を使うときはその後ろをコメント欄として使うことができます。

DSR レジスタの各状態コード (DC、N、Z、V、GT) は無条件の ALU 演算命令、シフト演算命令で常に更新されます。条件付き命令は条件が成立した場合でも状態コードを更新しません。乗算命令も状態コードを更新しません。DC ビットの定義は、DSR レジスタの CS ビットの指定によって決まります。

## 2. CPU

DSP 演算命令を分類別に表 2.25 に示します。

表 2.25 DSP 演算命令の分類

分類		命令の種類	オペコード	機能	命令数
A L U 算 術 演 算 命 令	ALU 固定小数点演算 命令	11	PABS	絶対値演算	28
			PADD	加算	
			PADD PMULS	加算と符号付き乗算	
			PADDC	キャリー付き加算	
			PCLR	クリア	
			PCMP	比較	
			PCOPY	コピー	
			PNEG	符号反転	
			PSUB	減算	
			PSUB PMULS	減算と符号付き乗算	
	PSUBC	ボロ付き減算			
	ALU 整数演算命令	2	PDEC	デクリメント	12
			PINC	インクリメント	
MSB 検出命令	1	PDMSB	MSB 検出	6	
丸め演算命令	1	PRND	丸め演算	2	
ALU 論理演算命令	3	PAND	論理積演算	9	
		POR	論理和演算		
		PXOR	排他的論理和演算		
固定小数点乗算命令	1	PMULS	符号付き乗算	1	
シ フ ト	算術シフト演算命令	1	PSHA	算術シフト	4
	論理シフト演算命令	1	PSHL	論理シフト	4
システム制御命令	2	PLDS	システムレジスタのロード	12	
		PSTS	システムレジスタからのストア		
		計 23			計 78

## (1) ALU 算術演算命令

## (a) ALU 固定小数点演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PABS Sx,Dz	もし Sx 0 ならば Sx Dz もし Sx < 0 ならば 0 - Sx Dz	111110***** 10001000xx00zzzz	1	更新
PABS Sy,Dz	もし Sy 0 ならば Sy Dz もし Sy < 0 ならば 0 - Sy Dz	111110***** 1010100000yyzzzz	1	更新
PADD Sx,Sy,Dz	Sx + Sy Dz	111110***** 10110001xxyyzzzz	1	更新
DCT PADD Sx,Sy,Dz	もし DC = 1 ならば Sx + Sy Dz もし 0 ならば nop	111110***** 10110010xxyyzzzz	1	
DCF PADD Sx,Sy,Dz	もし DC = 0 ならば Sx + Sy Dz もし 1 ならば nop	111110***** 10110011xxyyzzzz	1	
PADD Sx,Sy,Du PMULS Se,Sf,Dg	Sx + Sy Du Se の上位ワード × Sf の上位ワード Dg	111110***** 0111eefxxyygguu	1	更新
PADDC Sx,Sy,Dz	Sx + Sy + DC Dz	111110***** 10110000xxyyzzzz	1	更新
PCLR Dz	H'00000000 Dz	111110***** 100011010000zzzz	1	更新
DCT PCLR Dz	もし DC = 1 ならば H'00000000 Dz もし 0 ならば nop.	111110***** 100011100000zzzz	1	
DCF PCLR Dz	もし DC = 0 ならば H'00000000 Dz もし 1 ならば nop.	111110***** 100011110000zzzz	1	
PCMP Sx,Sy	Sx - Sy	111110***** 10000100xxyy0000	1	更新
PCOPY Sx,Dz	Sx Dz	111110***** 11011001xx00zzzz	1	更新
PCOPY Sy,Dz	Sy Dz	111110***** 1111100100yyzzzz	1	更新
DCT PCOPY Sx,Dz	もし DC = 1 ならば Sx Dz もし 0 ならば nop.	111110***** 11011010xx00zzzz	1	
DCT PCOPY Sy,Dz	もし DC = 1 ならば Sy Dz もし 0 ならば nop.	111110***** 1111101000yyzzzz	1	
DCF PCOPY Sx,Dz	もし DC = 0 ならば Sx Dz もし 1 ならば nop	111110***** 11011011xx00zzzz	1	
DCF PCOPY Sy,Dz	もし DC = 0 ならば Sy Dz もし 1 ならば nop	111110***** 11011011xx00zzzz	1	
PNEG Sx,Dz	0 - Sx Dz	111110***** 11001001xx00zzzz	1	更新
PNEG Sy,Dz	0 - Sy Dz	111110***** 1110100100yyzzzz	1	更新
DCT PNEG Sx,Dz	もし DC = 1 ならば 0 - Sx Dz もし 0 ならば nop	111110***** 11001010xx00zzzz	1	

2. CPU

命令	動作	命令コード	実行 ステート	DC ビット
DCT PNEG Sy,Dz	もし DC = 1 ならば 0 - Sy Dz もし 0 ならば, nop	111110***** 1110101000yyzzzz	1	
DCF PNEG Sx,Dz	もし DC = 0 ならば 0 - Sx Dz もし 1 ならば nop	111110***** 11001011xx00zzzz	1	
DCF PNEG Sy,Dz	もし DC = 0 ならば 0 - Sy Dz もし 1 ならば nop	111110***** 1110101100yyzzzz	1	
PSUB Sx,Sy,Dz	Sx - Sy Dz	111110***** 10100001xxyyzzzz	1	更新
DCT PSUB Sx,Sy,Dz	もし DC = 1 ならば Sx - Sy Dz もし 0 ならば nop	111110***** 10100010xxyyzzzz	1	
DCF PSUB Sx,Sy,Dz	もし DC = 0 ならば Sx - Sy Dz もし 1 ならば nop	111110***** 10100011xxyyzzzz	1	
PSUB Sx,Sy,Du PMULS Se,Sf,Dg	Sx - Sy Du Se の上位ワード x Sf の上位ワード Dg	111110***** 0110eefxxyygguu	1	更新
PSUBC Sx,Sy,Dz	Sx - Sy - DC Dz	111110***** 10100000xxyyzzzz	1	更新

(b) ALU 整数演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PDEC Sx,Dz	Sx の上位ワード - 1 Dz の上位ワード Dz の下位ワードをクリア	111110***** 10001001xx00zzzz	1	更新
PDEC Sy,Dz	Sy の上位ワード - 1 Dz の上位ワード Dz の下位ワードをクリア	111110***** 1010100100yyzzzz	1	更新
DCT PDEC Sx,Dz	もし DC = 1 ならば Sx の上位ワード - 1 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10001010xx00zzzz	1	
DCT PDEC Sy,Dz	もし DC = 1 ならば Sy の上位ワード - 1 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 1010101000yyzzzz	1	
DCF PDEC Sx,Dz	もし DC = 0 ならば Sx の上位ワード - 1 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10001011xx00zzzz	1	
DCF PDEC Sy,Dz	もし DC = 0 ならば Sy の上位ワード - 1 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 1010101100yyzzzz	1	

命令	動作	命令コード	実行 ステート	DC ビット
PINC Sx,Dz	Sx の上位ワード+ 1 Dz の上位ワード Dz の下位ワードクリア	111110***** 10011001xx00zzzz	1	更新
PINC Sy,Dz	Sy の上位ワード+ 1 Dz の上位ワード Dz の下位ワードクリア	111110***** 1011100100yyzzzz	1	更新
DCT PINC Sx,Dz	もし DC = 1 ならば Sx の上位ワード+ 1 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10011010xx00zzzz	1	
DCT PINC Sy,Dz	もし DC = 1 ならば Sy の上位ワード+ 1 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 1011101000yyzzzz	1	
DCF PINC Sx,Dz	もし DC = 0 ならば Sx の上位ワード+ 1 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10011011xx00zzzz	1	
DCF PINC Sy,Dz	もし DC = 0 ならば Sy の上位ワード+ 1 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 1011101100yyzzzz	1	

## (c) MSB 検出命令

命令	動作	命令コード	実行 ステート	DC ビット
PDMSB Sx,Dz	Sx データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア	111110***** 10011101xx00zzzz	1	更新
PDMSB Sy,Dz	Sy データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア	111110***** 1011110100yyzzzz	1	更新
DCT PDMSB Sx,Dz	もし DC = 1 ならば Sx データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10011110xx00zzzz	1	
DCT PDMSB Sy,Dz	もし DC = 1 ならば Sy データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア もし 0 ならば nop.	111110***** 1011111000yyzzzz	1	
DCF PDMSB Sx,Dz	もし DC = 0 ならば Sx データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10011111xx00zzzz	1	

## 2. CPU

命令	動作	命令コード	実行 ステート	DC ビット
DCF PDMSB Sx,Dz	もし DC = 0 ならば Sy データの MSB 位置 Dz の上位ワード、 Dz の下位ワードクリア もし 1 ならば nop.	111110***** 1011111100yyzzzz	1	

### (d) 丸め演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PRND Sx,Dz	Sx + H'00008000 Dz Dz の下位ワードクリア	111110***** 10011000xx00zzzz	1	更新
PRND Sy,Dz	Sy + H'00008000 Dz Dz の下位ワードクリア	111110***** 1011100000yyzzzz	1	更新

### (2) ALU 論理演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PAND Sx,Sy,Dz	Sx & Sy Dz Dz の下位ワードクリア	111110***** 10010101xxyyzzzz	1	更新
DCT PAND Sx,Sy,Dz	もし DC = 1 ならば Sx&Sy Dz Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10010110xxyyzzzz	1	
DCF PAND Sx,Sy,Dz	もし DC = 0 ならば Sx&Sy Dz Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10010111xxyyzzzz	1	
POR Sx,Sy,Dz	Sx   Sy Dz Dz の下位ワードクリア	111110***** 10110101xxyyzzzz	1	更新
DCT POR Sx,Sy,Dz	もし DC = 1 ならば Sx Sy Dz Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10110110xxyyzzzz	1	
DCF POR Sx,Sy,Dz	もし DC = 0 ならば Sx Sy Dz Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10110111xxyyzzzz	1	
PXOR Sx,Sy,Dz	Sx ^ Sy Dz Dz の下位ワードクリア	111110***** 10100101xxyyzzzz	1	更新
DCT PXOR Sx,Sy,Dz	もし DC = 1 ならば Sx^Sy Dz Dz の下位ワードクリア もし 0 ならば nop.	111110***** 10100110xxyyzzzz	1	
DCF PXOR Sx,Sy,Dz	もし DC = 0 ならば Sx^Sy Dz Dz の下位ワードクリア もし 1 ならば nop.	111110***** 10100111xxyyzzzz	1	

## (3) 固定小数点乗算命令

命令	動作	命令コード	実行 ステート	DC ビット
PMULS Se,Sf,Dg	Se の上位ワード x Sf の上位ワード Dg	111110***** 0100eeff0000gg00	1	

## (4) シフト演算命令

## (a) 算術シフト演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PSHA Sx,Sy,Dz	もし Sy = 0 ならば Sx << Sy Dz もし Sy < 0 ならば Sx >> Sy Dz	111110***** 10010001xxyyzzzz	1	更新
DCT PSHA Sx,Sy,Dz	もし DC = 1 & Sy = 0 ならば Sx << Sy Dz もし DC = 1 & Sy < 0 ならば Sx >> Sy Dz もし DC = 0 ならば nop	111110***** 10010010xxyyzzzz	1	
DCF PSHA Sx,Sy,Dz	もし DC = 0 & Sy = 0 ならば Sx << Sy Dz もし DC = 0 & Sy < 0 ならば Sx >> Sy Dz もし DC = 1 ならば nop	111110***** 10010011xxyyzzzz	1	
PSHA #imm,Dz	もし imm = 0 ならば Dz << imm Dz もし imm < 0 ならば Dz >> imm Dz	111110***** 00010iiiiiiizzzz	1	更新

## (b) 論理シフト演算命令

命令	動作	命令コード	実行 ステート	DC ビット
PSHL Sx,Sy,Dz	もし Sy = 0 ならば Sx << Sy Dz, Dz の下位ワードクリア もし Sy < 0 ならば Sx >> Sy Dz, Dz の下位ワードクリア	111110***** 10000001xxyyzzzz	1	更新
DCT PSHL Sx,Sy,Dz	もし DC = 1 & Sy = 0 ならば Sx << Sy Dz, Dz の下位ワードクリア もし DC = 1 & Sy < 0 ならば Sx >> Sy Dz, Dz の下位ワードクリア もし DC = 0 ならば nop	111110***** 10000010xxyyzzzz	1	

2. CPU

命令	動作	命令コード	実行 ステート	DC ビット
DCF PSHL Sx,Sy,Dz	もし DC = 0 & Sy = 0 ならば Sx << Sy Dz, Dz の下位ワードクリア もし DC = 0 & Sy < 0 ならば Sx >> Sy Dz, Dz の下位ワードクリア もし DC = 1 ならば nop	111110*****  10000011xxyyzzzz	1	
PSHL #imm,Dz	もし imm = 0 ならば Dz << imm Dz, Dz の下位ワードクリア もし imm < 0 ならば Dz >> imm Dz, Dz の下位ワードクリア	111110*****  00000iiiiiiizzzz	1	更新

(5) システム制御命令

命令	動作	命令コード	実行 ステート	DC ビット
PLDS Dz,MACH	Dz MACH	111110*****  111011010000zzzz	1	
PLDS Dz,MACL	Dz MACL	111110*****  111111010000zzzz	1	
DCT PLDS Dz,MACH	もし DC = 1 ならば Dz MACH もし 0 ならば nop.	111110*****  111011100000zzzz	1	
DCT PLDS Dz,MACL	もし DC = 1 ならば Dz MACL もし 0 ならば nop.	111110*****  111111100000zzzz	1	
DCF PLDS Dz,MACH	もし DC = 0 ならば Dz MACH もし 1 ならば nop.	111110*****  111011110000zzzz	1	
DCF PLDS Dz,MACL	もし DC = 0 ならば Dz MACL もし 1 ならば nop.	111110*****  111111110000zzzz	1	
PSTS MACH,Dz	MACH Dz	111110*****  110011010000zzzz	1	
PSTS MACL,Dz	MACL Dz	111110*****  110111010000zzzz	1	
DCT PSTS MACH,Dz	もし DC = 1 ならば MACH Dz もし 0 ならば nop.	111110*****  110011100000zzzz	1	
DCT PSTS MACL,Dz	もし DC = 1 ならば MACL Dz もし 0 ならば nop.	111110*****  110111100000zzzz	1	
DCF PSTS MACH,Dz	もし DC = 0 ならば MACH Dz もし 1 ならば nop.	111110*****  110011110000zzzz	1	
DCF PSTS MACL,Dz	もし DC = 0 ならば MACL Dz もし 1 ならば nop.	111110*****  110111110000zzzz	1	

## (6) NOPX と NOPY の命令コード

DSP 演算命令と同時に並行処理されるデータ転送命令がないときは、データ転送命令に NOPX、NOPY 命令を書くかあるいは命令を省略することもできます。NOPX、NOPY 命令を書いても省略しても命令コードは同じです。NOPX と NOPY の命令コードの例を表 2.26 に示します。

表 2.26 NOPX と NOPY の命令コードの例

命令	コード
PADD X0, Y0, A0 MOVX. W @R4+, X0 MOVY.W @R6+R9, Y0	1111100000001011 1011000100000111
PADD X0, Y0, A0 NOPX      MOVY.W @R6+R9, Y0	1111100000000011 1011000100000111
PADD X0, Y0, A0 NOPX      NOPY	1111100000000000 1011000100000111
PADD X0, Y0, A0 NOPX	1111100000000000 1011000100000111
PADD X0, Y0, A0	1111100000000000 1011000100000111
MOVX. W @R4+, X0 MOVY.W @R6+R9, Y0	1111000000001011
MOVX. W @R4+, X0 NOPY	1111000000001000
MOVS. W @R4+, X0	1111010010001000
NOPX      MOVY.W @R6+R9, Y0	1111000000000011
MOVY.W @R6+R9, Y0	1111000000000011
NOPX      NOPY	1111000000000000
NOP	0000000000001001

## 2.6 使用上の注意事項

1. DSP 命令を未使用時、消費電流低減のためにアプリケーションソフトの初期化部分で以下ダミー命令を実行するようにしてください。

PCLR      A0 ;      A0 レジスタを 0 クリアする。

PSHA #5, A0 ;      5 ビットを左シフトする。

2. DSP 命令を実行後に SR レジスタの S ビットを変更する場合、パイプラインは正しく動作しません。以下の (a) (b) に従って実行するようにしてください。

(a) DSP 命令が実行された後に SR レジスタの S ビットを変更しない。

## 2. CPU

---

(b)LDC Rn, SR 命令の前に NOP 命令を挿入する

<例>

```
PSHA    #1, A0
PINC    X0, A0 MOVX.W A1,@R5
NOP
LDC     R0, SR
```

3. CPU 命令の倍精度乗算 (MUL.L/DMULU.L/DMULS.L) または倍精度積和演算 (MAC.L) と DSP 演算命令を組み合わせて実行する場合、誤動作することがあります。

下記、発生条件および回避方法をご参考の上、ご使用ください。

### 【発生条件】

下記 (a)、(b) の条件が同時に成立した場合に、(b) の (2) に記述されている命令が誤動作することがあります。

(a) 内蔵メモリや内蔵キャッシュから命令を実行

(b) 下記命令列を (1)(2)(3) の順に実行

(1) 倍精度乗算命令 (MUL.L/DMULU.L/DMULS.L) または倍精度積和演算命令 (MAC.L)

(2) (PMULS、PSTS、PLDS) 以外の DSP 演算命令 (注 1)

(3) PMULS、PSTS、PLDS のいずれかの命令

上記 (b) の (1) は、実行に複数サイクルかかります。このため、実行中に同じリソースを使用する (b) の (3) のような命令が発行された場合、実行中の演算動作が終了するまで (3) の実行開始が抑止されます。(2) の命令は (1) の命令と相関関係がありませんので実行されますが、その後 (3) の命令の実行抑止動作によって (2) の命令を正しく実行終了出来ない場合が生じます。なお、(2) の命令が存在せず、(1) の命令の直後に (3) の命令が連続して記述されている場合は、問題なく正常に実行されます。

### 【プログラム上の回避方法】

本制限事項を回避するためには、以下の (1) ~ (3) のいずれかを実施してください。

(1) 上記条件 (b) 命令列を実行しない。

(2) 上記条件 (b) 命令列が命令コード上に存在していて、(2) と (3) の命令を入れ替えても問題ない場合は、(2) と (3) の命令を入れ替えてください。

(3) 上記条件 (b) 命令列が命令コード上に存在していて、(2) と (3) の命令を入れ替えると問題がある場合は、命令列 (1) と (2) の間に NOP 命令もしくは乗算器に関係しない CPU 命令を 1 つ以上挿入して下さい。

注 1 : PMULS、PSTS、PLDS 以外の DSP 演算命令とは、以下の命令を指します。

PABS、PADD、PADDC、PAND、PCLR、PCMP、PCOPY、PDEC、PDMSB、PINC、PNEG、POR、PRND、PSHA、PSHL、PSUB、PSUBC、PXOR

【補足】 この注意事項は、上記 (b) の (1) の直前に遅延分岐命令があり、(b) の (1) の命令が遅延スロットに入っていて、(b) の (2)(3) が分岐先で連続して記述されている場合も適用されます。

4. 乗算 / 積和命令による乗算器競合による命令の実行ストール現象回避の事例及び方法を以下に示します。

【不具合内容】

乗算 / 積和命令による乗算器競合が発生し、命令の実行がストールされる状況において、乗算 / 積和命令直後に SR (ステータスレジスタ) の S ビット (飽和演算ビット) を変化させると、命令実行順番が逆転してしまい、そのため、S ビット変更前に実行されるべき命令が S ビット変更後に実行され、誤った演算結果が得られてしまう場合があります。

S ビット変更の影響を受ける命令

積和命令 : MAC.W、MAC.L

【発生条件】

以下に不具合の発生する例を示します。

(1) 乗算 / 積和命令の場合

DMULU.L R4, R10 MUL.L、DMULS.L、DMULU.L、MAC.L が該当

MAC.L @R5+, @R5+ MAC.W、MAC.L が該当、乗算器競合が発生し、命令実行をストールする条件発生

LDC R0, SR 飽和演算モード変更

の DMULU.L 命令と の MAC.L 命令で乗算器競合が発生し、 の MAC.L 命令実行はストールされます。ところで、 の S ビットの変更はパイプライン動作の為、 の MAC.L 命令実行前に CPU 内で実行されてしまい、結果的に と の実行順番が逆転し、MAC.L 命令の演算結果が不正な結果となってしまいます。

【プログラム上の回避方法】

本制限事項を回避するためには、以下の 3 つの方法のうちいずれかを行ってください。

- (1) 積和命令直後には SR レジスタアクセスを行わないでください。
- (2) LDC Rn,SR 命令の直前に NOP 命令を挿入してください。
- (3) 乗算器競合が発生しない (ストールが発生しない) ようにしてください。



---

## 3. 発振回路と動作モード

---

### 3.1 概要

動作モード端子によって、内蔵発振回路の動作、CS0 空間のバス幅指定を制御します。  
クロックソースとして、水晶発振子と外部クロックを選ぶことができます。

### 3. 発振回路と動作モード

## 3.2 内蔵発振回路と動作モード

### 3.2.1 発振回路の構成

内蔵発振回路のブロックを図 3.1 に示します。

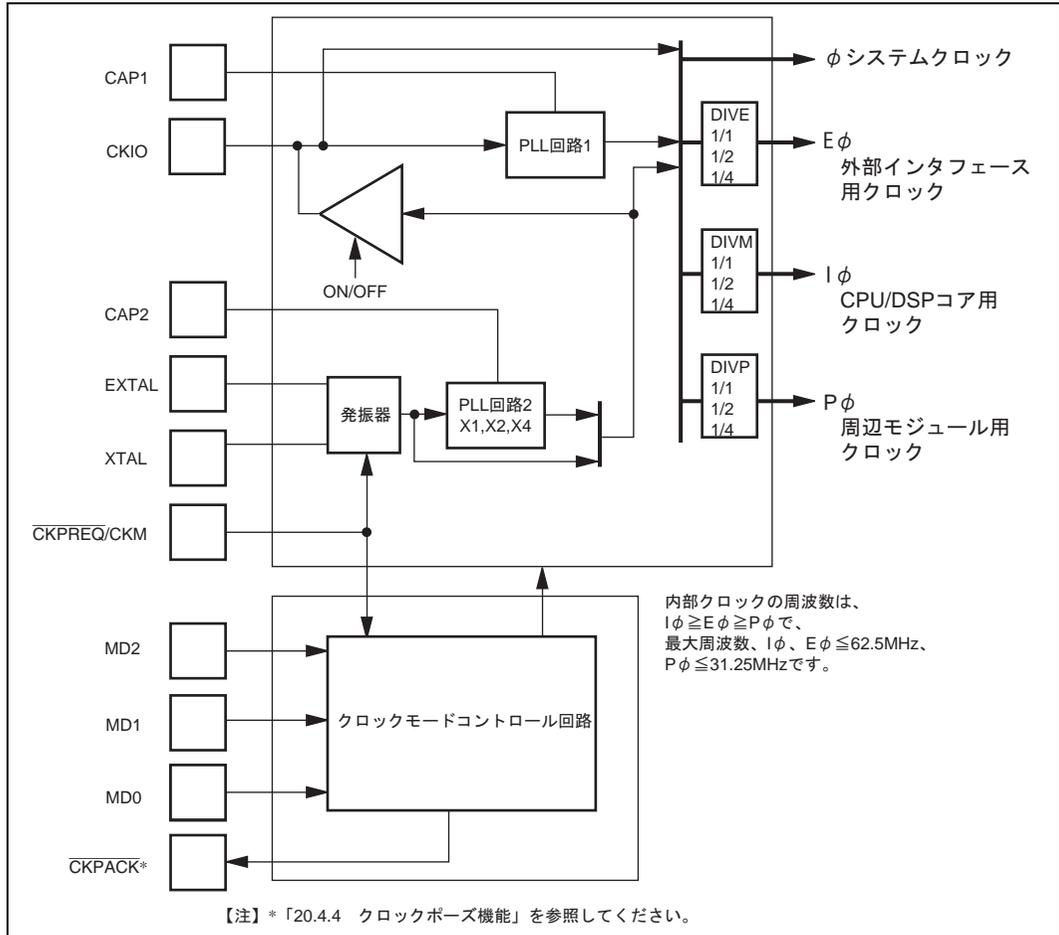


図 3.1 内蔵発振回路のブロック図

## (1) 端子構成

発振回路に関係した各端子の機能を表 3.1 に示します。

表 3.1 端子の機能

機能名	入出力	機能
CKIO	入出力	外部クロック入力端子または内部クロック出力端子
XTAL	出力	水晶発振子を接続
EXTAL	入力	水晶発振子を接続、または PLL 回路 2 を使用する場合の外部クロック入力
CAP1	入力	PLL 回路 1 動作用の容量を接続
CAP2	入力	PLL 回路 2 動作用の容量を接続
MD0	入力	この端子に印加するレベルでクロックモードを指定
MD1	入力	この端子に印加するレベルでクロックモードを指定
MD2	入力	この端子に印加するレベルでクロックモードを指定
CKPREQ/CKM	入力	クロックポーズリクエスト、または水晶発振子の動作指定
CKPACK	出力	クロックポーズ機能

## (2) PLL 回路 1

PLL 回路 1 は、外部クロックと LSI 内部に提供されるクロックとの位相ずれをなくす回路です。高速動作する場合、基準となるクロックと LSI 内部の動作クロックの位相差が、そのまま、周辺デバイスとのインタフェースマージンに影響します。このようなことがないように PLL 回路 1 を内蔵しています。

## (3) PLL 回路 2

PLL 回路 2 は、水晶発振子、または EXTAL 端子外部クロック入力から与えられたクロック周波数を 1 倍、2 倍、4 倍して LSI の動作周波数として使用する場合に用います。クロックを何倍して使用するかは、周波数変更レジスタにて設定します。

### 3. 発振回路と動作モード

#### 3.2.2 クロック動作モードの設定

クロックモード 0 ~ 6 の機能、動作を表 3.2 に示します。

表 3.2 動作モード

クロックモード	機能・動作	クロックソース
モード 0	PLL 回路 1 と PLL 回路 2 が動作します。このとき、CKIO 端子からは LSI の内部クロック (I、E、P) と同位相のクロック (E と同一周波数) を出力します。 周波数変更レジスタ (FMR) の制御ビットにより、PLL 回路 1 と PLL 回路 2 の動作 / 停止が切り替え可能です。また、CKIO 端子を Hi-Z 状態にすることも可能です。	水晶発振子 / 外部クロック入力
モード 1	PLL 回路 1 と PLL 回路 2 が動作します。このとき、CKIO 端子からは LSI 内部のシステムクロック に対して 1/4 サイクル早いクロック (E と同一周波数) を出力します。 FMR の制御ビットにより、PLL 回路 1 と PLL 回路 2 の動作 / 停止が切り替え可能です。また、CKIO 端子を Hi-Z 状態にすることも可能です。 通常はモード 0 をお使いください。	
モード 2	PLL 回路 2 が動作します。このとき、CKIO 端子からは PLL 回路 2 からのクロックを出力します (E と同一周波数)。PLL 回路 1 は常時停止するため、位相合わせは行いません。 FMR の制御ビットにより、PLL 回路 2 の動作 / 停止が切り替え可能です。また、CKIO 端子を Hi-Z 状態にすることも可能です。	
モード 3	PLL 回路 2 が動作します。このとき、CKIO 端子は常時 Hi-Z 状態です。 FMR の制御ビットにより、PLL 回路 2 の動作 / 停止が切り替え可能です。	
モード 4	PLL 回路 1 が動作します。CKIO 端子から入力したクロックと LSI の内部クロック (I、E、P) との位相を合わせて動作させるときに PLL 回路 1 を動作させてください。PLL 回路 2 は常時停止します。 FMR の制御ビットにより、PLL 回路 1 の動作 / 停止が切り替え可能です。	外部クロック入力
モード 5	PLL 回路 1 が動作します。CKIO 端子から入力したクロックと LSI の内部クロック (I、E、P) が、システムクロック に対して 1/4 サイクルずらせて動作させるときに PLL 回路 1 を動作させてください。PLL 回路 2 は常時停止します。 FMR の制御ビットにより、PLL 回路 1 の動作 / 停止が切り替え可能です。 通常はモード 4 をお使いください。	
モード 6	PLL 回路 1、PLL 回路 2 は共に常時停止します。CKIO 端子から入力したクロックと同じ周波数のクロックで動作させるときに設定してください。	

各クロックモードで、内部クロック周波数を切り替えることができます (「3.2.5 レジスタによる動作周波数の選択」参照)。

クロックモードで 4 ~ 6 では CKIO 端子から入力されるクロックの周波数変更、クロックの停止を行うことができます (「20.4.4 クロックポーズ機能」参照)。

MD2 ~ 0 端子の組み合わせとクロックモードとの関係を表 3.3 に示します。動作途中に MD2 ~ 0 端子を切り替えないでください。切り替えた場合の動作は保証できません。

表 3.3 クロックモードの端子設定と状態

クロックモード	端子名						
	MD2	MD1	MD0	CKPREQ/CKM	EXTAL	XTAL	CKIO
モード0	0	0	0	0	クロック入力	OPEN	出力 / Hi-Z
				1	水晶発振		
モード1	0	0	1	0	クロック入力	OPEN	
				1	水晶発振		
モード2	0	1	0	0	クロック入力	OPEN	
				1	水晶発振		
モード3	0	1	1	0	クロック入力	OPEN	Hi-Z
				1	水晶発振		
モード4	1	0	0	*	OPEN		クロック入力
モード5	1	0	1				
モード6	1	1	0				

【注】 上記以外の組み合わせでは使用しないでください。

\* クロックモード 4、5、6 のときには、CKPREQ/CKM 端子はクロックポーズリクエストとして機能します。

### 3.2.3 水晶発振子の接続方法

図 3.2 に水晶発振子との接続例を示します。ダンピング抵抗 ( $R_d$ ) と負荷容量 ( $CL1$ 、 $CL2$ ) の値は使用する水晶発振子のメーカーと部品検討の上、決めてください。水晶発振子は、AT カット並列共振タイプを使用してください。水晶発振子と負荷容量は、できるだけ EXTAL 端子と XTAL 端子の近くに配置してください。誘導による誤発振を避けるため、端子の信号線は他の信号線と交差させないでください。

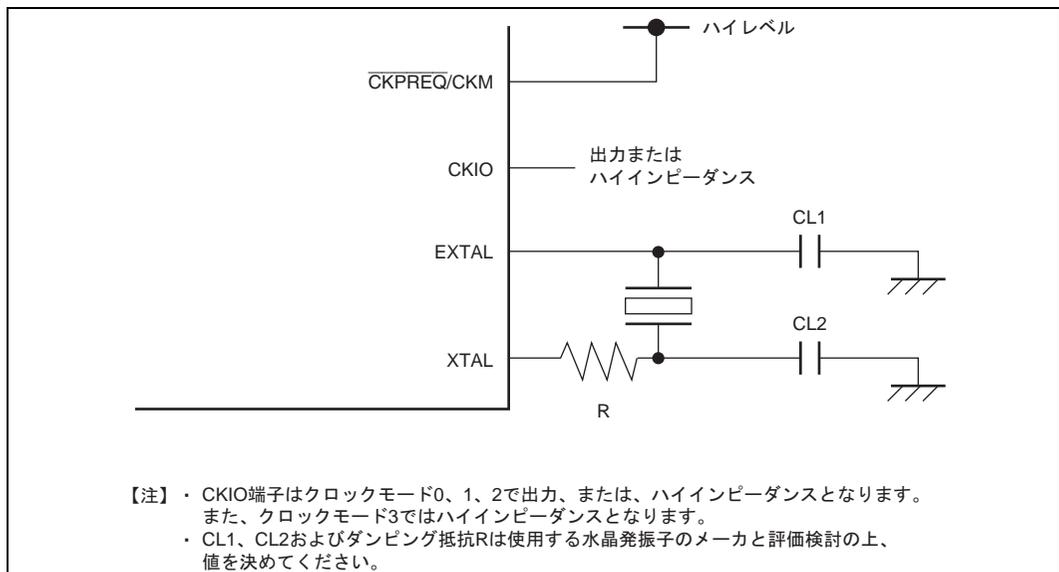


図 3.2 水晶発振子の接続例

### 3. 発振回路と動作モード

#### 3.2.4 外部クロックの入力方法

外部クロック入力、クロックモードによって EXTAL 端子または CKIO 端子から入力します。

- (1) EXTAL 端子からのクロック入力  
クロックモード 0、1、2、3 で使用できます。

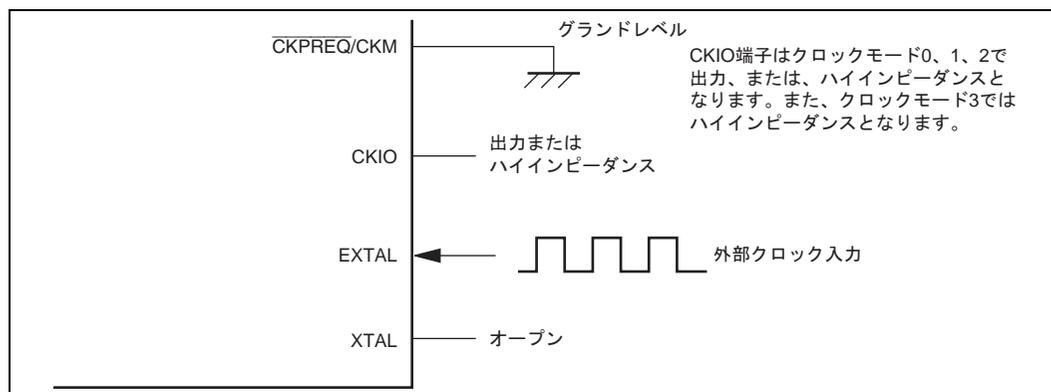


図 3.3 外部クロック入力方法

- (2) CKIO 端子からのクロック入力  
クロックモード 4、5、6 で使用します

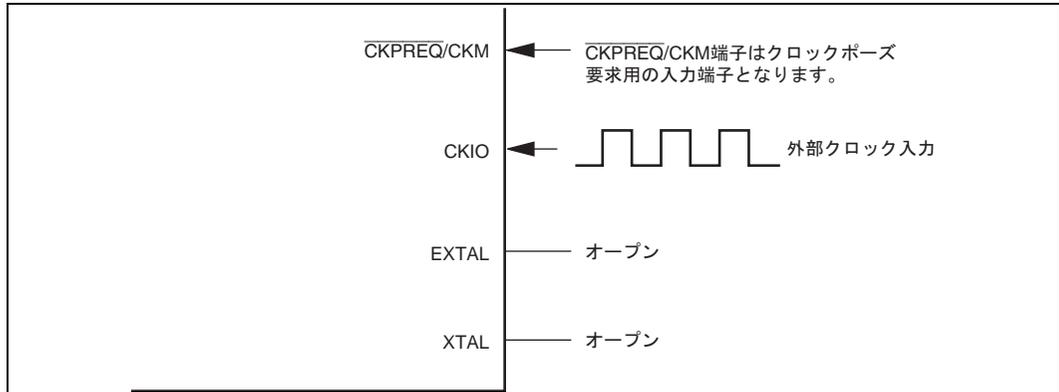


図 3.4 外部クロック入力方法

### 3.2.5 レジスタによる動作周波数の選択

周波数変更レジスタ (FMR) を用いて、内部クロック (I、E、P) の動作周波数分周率の指定ができます。内部クロックの周波数は、PLL 回路 1、PLL 回路 2、および分周器 DIVM、DIVE、DIVP の制御によって決まります。

#### (1) 周波数変更レジスタ (FMR)

周波数変更レジスタは、RES 端子からのパワーオンリセット時のみ初期化されますが、WDT のオーバーフローによる内部リセットでは初期化されません。また周波数変更レジスタの初期値は、MD2 ~ 0 端子の設定により異なります。MD2 ~ 0 端子の組み合わせと周波数変更レジスタ初期値の関係を表 3.4 に示します。

表 3.4 クロックモードの端子設定と周波数変更レジスタの初期値

クロックモード	MD2	MD1	MD0	初期値
モード 0	0	0	0	H'00
モード 1	0	0	1	
モード 2	0	1	0	H'40
モード 3	0	1	1	H'60
モード 4	1	0	0	H'A6
モード 5	1	0	1	
モード 6	1	1	0	H'E0

### 3. 発振回路と動作モード

レジスタ構成を表 3.5 に示します。

表 3.5 レジスタ構成

名称	略称	R/W	初期値	アドレス
周波数変更レジスタ	FMR	R/W	*	H'FFFFFFE90

【注】 \* クロックモードにより初期値が異なります。表 3.4 を参照してください。

ビット:	7	6	5	4	3	2	1	0
	PLL2ST	PLL1ST	CKIOST	-	FR3	FR2	FR1	FR0
初期値:	-	-	-	0	0	-	-	0
R/W:	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

・ ビット 7: PLL2ST

モード 0 ~ 3 において切り替え可能です。モード 4 ~ 6 では、PLL 回路 2 は使用できません。このとき、本ビットは常に 1 が読み出されます。

ビット 7	PLL2ST	説明
0		PLL 回路 2 を使用する
1		PLL 回路 2 を使用しない

・ ビット 6: PLL1ST

モード 0、1、4、5 において切り替え可能です。モード 2、3、6 では、PLL 回路 1 は使用できません。このとき、本ビットは常に 1 が読み出されます。

ビット 6	PLL1ST	説明
0		PLL 回路 1 を使用する
1		PLL 回路 1 を使用しない

・ ビット 5: CKIOST

モード 0 ~ 3 において設定可能です。モード 4 ~ 6 では、CKIO 端子が入力端子となります。このとき、本ビットは常に 1 が読み出されます。

ビット 5	CKIOST	説明
0		CKIO 端子は、E を出力
1		CKIO 端子は、Hi-Z 状態 (PLL 回路 1 が動作状態のときは、CKIO を Hi-Z 状態にしないでください)

・ ビット 4: 予約ビット

読み出すと、常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 3 ~ 0 : FR3 ~ 0

周波数設定ビット FR3 ~ 0 によって、内部クロック周波数、および、CKIO 出力周波数（モード 0 ~ 2）が設定できます。モード、および、PLL 回路 1、PLL 回路 2 の動作 / 停止状態によって、FR3 ~ 0 に設定できる値が異なります。以下に、FR3 ~ 0 に設定できる値と、そのときの内部クロック、および CKIO 出力の周波数比を、外部入力クロック周波数を 1 として示します。

モード 0、1

- PLL 回路 1、PLL 回路 2 が動作している場合
- EXTAL 入力、または、水晶発振子使用

FR3	FR2	FR1	FR0		I	E	P	CKIO
0	0	0	0	× 4	× 1	× 1	× 1	E
0	1	0	0	× 4	× 2	× 1	× 1	E
0	1	0	1	× 4	× 2	× 2	× 1	E
0	1	1	0	× 4	× 2	× 2	× 2	E
1	0	0	0	× 4	× 4	× 1	× 1	E
1	0	0	1	× 4	× 4	× 2	× 1	E
1	0	1	0	× 4	× 4	× 2	× 2	E
1	1	0	0	× 4	× 4	× 4	× 1	E
1	1	1	0	× 4	× 4	× 4	× 2	E

【注】 上記以外の組み合わせでは、使用しないでください。

モード 0 ~ 3

- PLL 回路 1 が停止、PLL 回路 2 が動作している場合
- EXTAL 入力、または、水晶発振子使用

FR3	FR2	FR1	FR0		I	E	P	CKIO
0	0	0	0	× 1	× 1	× 1	× 1	E
0	1	0	1	× 2	× 2	× 2	× 1	E
0	1	1	0	× 2	× 2	× 2	× 2	E
1	1	0	0	× 4	× 4	× 4	× 1	E
1	1	1	0	× 4	× 4	× 4	× 2	E

【注】 上記以外の組み合わせでは、使用しないでください。

モード 0、1

- PLL 回路 1 が動作、PLL 回路 2 が停止している場合
- EXTAL 入力、または、水晶発振子使用

FR3	FR2	FR1	FR0		I	E	P	CKIO
0	0	0	0	× 4	× 1	× 1	× 1	E
0	1	0	0	× 4	× 2	× 1	× 1	E
1	0	0	0	× 4	× 4	× 1	× 1	E

【注】 上記以外の組み合わせでは、使用しないでください。

### 3. 発振回路と動作モード

モード 4、5PLL 回路 1 が動作、PLL 回路 2 が停止している場合

- PLL 回路 1 が動作、PLL 回路 2 が停止している場合
- CKIO 入力

FR3	FR2	FR1	FR0		I	E	P	CKIO
0	1	0	1	× 2	× 1	× 1	× 1/2	E
0	1	1	0	× 2	× 1	× 1	× 1	E
1	0	0	1	× 2	× 2	× 1	× 1/2	E
1	0	1	0	× 2	× 2	× 1	× 1	E

【注】 上記以外の組み合わせでは、使用しないでください。

モード 0 ~ 6

- PLL 回路 1、PLL 回路 2 が停止している場合
- EXTAL 入力、または、水晶発振子使用 (モード 0 ~ 3)
- CKIO 入力 (モード 4 ~ 6)

FR3	FR2	FR1	FR0		I	E	P	CKIO
0	0	0	0	× 1	× 1/4	× 1/4	× 1/4	× 1
0	1	0	0	× 1	× 1/2	× 1/4	× 1/4	× 1
0	1	0	1	× 1	× 1/2	× 1/2	× 1/4	× 1
0	1	1	0	× 1	× 1/2	× 1/2	× 1/2	× 1
1	0	0	0	× 1	× 1	× 1/4	× 1/4	× 1
1	0	0	1	× 1	× 1	× 1/2	× 1/4	× 1
1	0	1	0	× 1	× 1	× 1/2	× 1/2	× 1
1	1	0	0	× 1	× 1	× 1	× 1/4	× 1
1	1	1	0	× 1	× 1	× 1	× 1/2	× 1
1	1	1	1	× 1	× 1	× 1	× 1	× 1

【注】 上記以外の組み合わせでは、使用しないでください。

#### (2) 周波数変更方法

周波数変更レジスタを書き換えた後、PLL 回路 1 または、PLL 回路 2 が動作状態 (動作状態での周波数変更レジスタの書き換えを含む) となる場合、下記の手順、注意事項にしたがって周波数変更レジスタにアクセスしてください。

- 周波数変更フロー
1. PLL 回路の発振安定時間を確保するために、内蔵ウォッチドッグタイマ (WDT) のオーバフロー時間を設定します (WTCSR の CKS2 ~ CKS0 ビット)
  2. WTCSR の  $\overline{WT}/\overline{IT}$ 、TME ビットを 0 に設定します。
  3. 外部メモリ空間 0 ~ 4 のキャッシュスルー領域のどこかをリードしてください。

4. 周波数変更レジスタを目的とする周波数、または、PLL 回路 1、PLL 回路 2 の動作 / 停止状態を変更してください (このとき、LSI 内部では一時的にクロックが止まります)。
5. 発振回路の各回路は動作し、WDT にクロックを供給します。このクロックで WDT がカウントアップします。
6. WDT のオーバフローで周波数設定ビット FR3 ~ FR0 で設定した周波数のクロックが供給され始めます。このとき、WTCSR の OVF ビット、RSTCSR の WOVF ビットはセットされません。また、インターバルタイマ割り込み (ITI) は発生せず、 $\overline{\text{WDTOVF}}$  信号もアサートされません。

以下に周波数変更プログラム例を示します。

```

;          SH7615 周波数変更
          1999 06.01 presented by Renesas

FMR      .equ      h'fffffe90
WTCSR    .equ      h'fffffe80
RSTCSR   .equ      h'fffffe83

PACR     .equ      h'fffffc80

XRAM     .equ      h'1000e000

          .export  _init_FMR

_init_FMR:
          mov.l     #XRAM,r1
          mov.l     r1,r5
          mov.l     #FREQUENCY,r2
          mov.l     #FREQUENCY_END,r3

program_move:
          mov.w     @r2,r0
          mov.w     r0,@r1
          add       #2,r1
          add       #2,r2
          cmp/eq    r2,r3
          bf        program_move
          nop

```

### 3. 発振回路と動作モード

---

```
mov.l      #PACR,r1
mov.w      #h'0008,r0
mov.w      r0,@r1
MOV.L      #WTCSR,R1
MOV.W      #H'A51F,R2
MOV.L      #H'26200000,R3
MOV.L      #FMR,R4

jmp        @r5
nop
nop
nop
nop
nop
nop
clock4_err:
bra        clock4_err
nop
nop
nop
nop
nop
;
;      周波数変更プログラムの本体
;      これを XRAM へコピーしてから
;      XRAM 上で実行する
FREQUENCY:
; < ウォッチドッグタイムコントロール / ステータスレジスタ設定 >
; TME ビットクリア
; WTCNT に入力するクロック = /16384
; ( オーバーフロー周期 =262.144ms )
MOV.W      R2,@R1

; < 外部キャッシュスルー領域リード >
; 外部メモリ空間 3 のキャッシュスルー領域 : H'26200000
MOV.L      @R3,R0

; < 周波数変更レジスタ設定 >
; PLL 回路 1   動作しない
; PLL 回路 2   動作する
; I ( x 4)=62.5MHz、E ( x 4)=62.5MHz、
; P ( x 2)=31.25MHz、CKIO(E )=62.5MHz、
```

```

;          MOV          #H'4E,R0

;PLL 回路 1&2   動作する
;I ( × 4)=62.5MHz、 E ( × 2)=31.25MHz、
;P ( × 2)=31.25MHz、 CKIO(E )=31.25MHz、

          MOV          #H'0A,R0

;PLL 回路 1&2   動作する
;I ( × 4)=62.5MHz、 E ( × 1)=15.625MHz、
;P ( × 1)=15.625MHz、 CKIO(E )=15.625MHz、
;          MOV          #H'08,R0

MOV.B     R0,@R4
rts
nop
FREQUENCY_END:
NOP

.END

```

- ・ 注意事項
- ・ 外部メモリ空間 0 ~ 4 のキャッシュスルー領域からのリード、および、周波数変更レジスタへの書き込みは、内蔵 X/Y メモリ上で行ってください。外部メモリ空間 0 ~ 4 からのリード後、周波数変更レジスタへの書き込みまでの間に、外部メモリ空間 0 ~ 4 への書き込み動作を行わないでください。
- ・ 周波数変更レジスタにライトアクセスした場合、WDT は自動的に起動します。
- ・ PLL 回路 1 が動作状態のとき、CKIO 出力を OFF にしないでください。
- ・ PLL 回路が安定していない期間は、CKIO 出力も同様に安定していません。
- ・ 周波数の変更を行う際は、必ず内蔵 DMAC ( E-DMAC、DMAC ) を停止してから行ってください。

周波数変更レジスタ書き換え後 PLL 回路 1 または PLL 回路 2 が動作状態 ( 動作状態での周波数変更レジスタの書き換えを含む ) とならない場合、上記フローと注意事項は当てはまりません。つまり、周波数変更レジスタを変更しても、WDT は起動しません。

### 3. 発振回路と動作モード

#### 3.2.6 クロックモードと周波数範囲

動作モードとこれに伴う入力クロックの周波数範囲を以下に示します。

モード	クロック入力		PLL 回路		内部クロック <sup>*2,*3</sup>			CKIO 出力 (MHz)
	印加端子	入力周波数範囲 (MHz)	PLL1	PLL2	I (MHz)	E (MHz)	P (MHz)	
0、1	EXTAL または 水晶発振子 <sup>*1</sup>	8 ~ 15.625	ON	ON	8 ~ 62.5	8 ~ 62.5	8 ~ 31.25	8 ~ 62.5
			OFF	ON	8 ~ 62.5	8 ~ 62.5	8 ~ 31.25	8 ~ 62.5
			ON	OFF	8 ~ 62.5	8 ~ 15.625	8 ~ 15.625	8 ~ 15.625
		1 ~ 31.25	OFF	OFF	1 ~ 31.25	1 ~ 31.25	1 ~ 31.25	1 ~ 31.25
2		8 ~ 15.625	OFF	ON	8 ~ 62.5	8 ~ 62.5	8 ~ 31.25	8 ~ 62.5
				OFF	1 ~ 31.25	1 ~ 31.25	1 ~ 31.25	1 ~ 31.25
3		8 ~ 15.625	OFF	ON	8 ~ 62.5	8 ~ 62.5	8 ~ 31.25	-
				OFF	1 ~ 31.25	1 ~ 31.25	1 ~ 31.25	-
4、5	CKIO	16 ~ 31.25	ON	OFF	16 ~ 62.5	16 ~ 31.25	16 ~ 31.25	-
		1 ~ 31.25	OFF		1 ~ 31.25	1 ~ 31.25	1 ~ 31.25	
6		1 ~ 31.25	OFF		1 ~ 31.25	1 ~ 31.25	1 ~ 31.25	

- 【注】 \*1 水晶発振子を使用する場合は、その周波数範囲は、8 ~ 15.625MHz に設定してください。  
\*2 すべての内部クロックの周波数が 1MHz を下回らないように、周波数設定レジスタを設定してください。  
\*3 内部クロックの周波数は I E P で使用してください。

### 3.2.7 ボード設計上の注意事項

#### (1) 水晶発振子使用時の注意

水晶発振子と容量 CL1、CL2 およびダンピング抵抗 R はできるだけ XTAL 端子と EXTAL 端子の近くに置いてください。また、誘導を避け、正しい発振を行うために、発振子に付加するコンデンサの接地点は共通にし、これらの部品の近くには配線パターンを置かないでください。

発振回路例を図 3.5 に示します。本構成は発振回路の製作例であり、実システムにおいては、ノイズ、電源特性、配線パターン等の周辺環境により、値が異なります。したがって、弊社は、この回路定数を保証するものではなく、必ず参考値としてご利用ください。お客様のシステムにおける最適発振回路定数が必要な場合は、発振子メーカーと評価検討の上、値を決めてください。

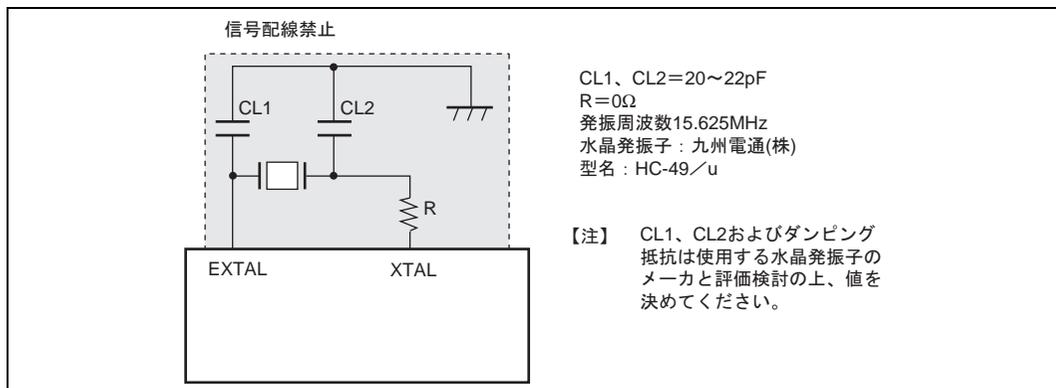


図 3.5 水晶発振子使用時の注意

#### (2) バイパスコンデンサについての注意

できるだけ Vss と Vcc のペアごとに 0.01 ~ 0.1 (μF) の積層セラミックコンデンサをバイパスコンデンサとして入れてください。

バイパスコンデンサはできるだけ LSI の電源端子の近くに実装してください。

容量値だけでなく、周波数特性は LSI の動作周波数に合った特性を持つものを使用してください。

< FP-208C/CV 使用時の Vss-Vcc ペア >

- PLL 系：9-12
- 3V デジタル系：
  - 20-18、26-22、35-33、45-42、52-50、60-58、61-67、69-66、78-76、79-81、91-89、101-99、112-109、113-110、114-116、130-132、149-146、150-147
- 5V デジタル系：157-155、169-167、181-179、191-193、202-200

< BP-240A/AV 使用時の Vss-Vcc ペア >

- PLL 系：E1、F1、F2、F4
- 3V デジタル系：
  - H4-H2、K1-J2、N4-M2、T3-P1、W1-V1、V5-U4、W5-U6、V7-T6、V10-V9、U10-W10、W13-U13、W17-W16、T17-U19、R19-R18、U17-R16、L17-K17、E17-F17、F18-E19

### 3. 発振回路と動作モード

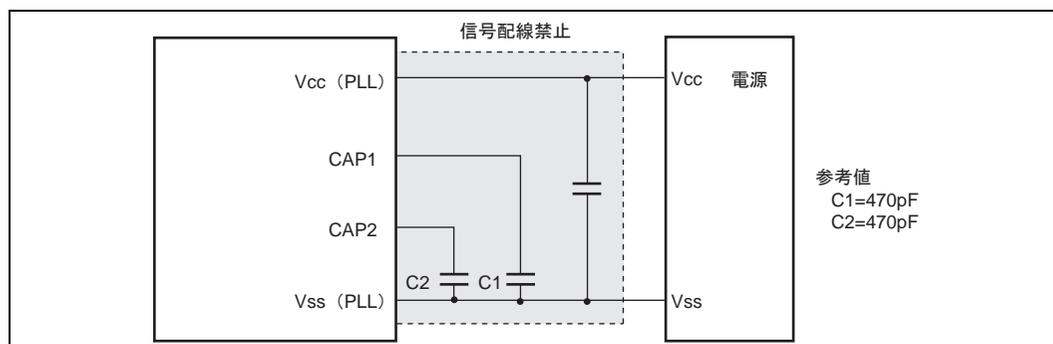
- 5V デジタル系 : B18-B19、B14-C15、D11-A11、B8-C7、C4-C5

#### (3) PLL 発振回路使用時の注意

PLL 用 Vcc と Vss の接続パターンはボード電源供給端子からの配線長を短くし、パターン幅は広くしてインダクタンス成分をできるだけ減らしてください。

発振安定用の容量 C1、C2 の接地はそれぞれ Vss (PLL1) と Vss (PLL2) を接地してください。

また、できるだけ CAP1、CAP2 端子の近くに置き、その近辺に配線パターンを配置しないでください。



### 3.3 CS0 空間のバス幅

CS0 空間のバス幅指定は MD4、MD3 端子を使用します。各端子の組み合わせ、機能を表 3.6 に示します。MD4、MD3 端子は動作途中で切り替えないでください。切り替えた場合の動作は保障できません。

表 3.6 CS0 空間のバス幅指定

端子		機能
MD4	MD3	
0	0	8 ビットバス幅を選択
0	1	16 ビットバス幅を選択
1	0	32 ビットバス幅を選択
1	1	設定禁止



## 4. 例外処理

### 4.1 概要

#### 4.1.1 例外処理の種類と優先順位

例外処理は、表 4.1 に示すようにリセット、アドレスエラー、割り込み、および命令の各要因によって起動されます。例外要因には、表 4.1 に示すように優先順位が設けられており、複数の例外要因が同時に発生した場合は、この優先順位に従って受け付けられ、処理されます。

表 4.1 例外処理の種類と優先順位

例外処理		優先順位	
リセット	パワーオンリセット		
	マニュアルリセット		
アドレスエラー	CPUアドレスエラー		
	DMAアドレスエラー (DMACおよびE-DMACによる)		
割り込み	NMI		
	ユーザブレイク		
	ハイパフォーマンスユーザデバッグインタフェース (H-UDI)		
	外部割り込み (IRL1~IRL15、IRQ0~IRQ3 (IRL3、IRL2、IRL1、IRL0端子で設定))		
	内蔵周辺モジュール		ダイレクトメモリアクセスコントローラ (DMAC)
			ウォッチドッグタイマ (WDT)
			コンペアマッチ割り込み (バスステートコントローラの一部)
			イーサネットコントローラ (EtherC) およびイーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)
			16ビットフリーランニングタイマ (FRT)
			16ビットタイマパルスユニット (TPU)
FIFO内蔵シリアルコミュニケーションインタフェース (SCIF)			
シリアルI/O (SIO)			
命令	トラップ命令 (TRAPA命令)		
	一般不当命令 (未定義コード)		
	スロット不当命令 (遅延分岐命令*1 直後に配置された未定義コードまたはPCを書き換える命令*2)		

【注】 \*1 遅延分岐命令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA

\*2 PC を書き換える命令：JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRA

## 4. 例外処理

### 4.1.2 例外処理の動作

各例外要因は表 4.2 に示すタイミングで検出され、処理が開始されます。

表 4.2 例外要因検出と例外処理タイミング

例外処理		要因検出および処理開始タイミング
リセット	パワーオンリセット	NMI 端子がハイレベルのときの RES 端子のローレベルからハイレベルへの変化で開始される
	マニュアルリセット	NMI 端子がローレベルのときの RES 端子のローレベルからハイレベルへの変化で開始される
アドレスエラー		命令のデコード時に検出され、この前までに実行中の命令が完了後開始される
割り込み		命令のデコード時に検出され、この前までに実行中の命令が完了後開始される
命令	トラップ命令	TRAPA 命令の実行により開始される
	一般不当命令	遅延分岐命令直後（遅延スロット）以外にある未定義コードがデコードされると開始される
	スロット不当命令	遅延分岐命令直後（遅延スロット）に配置された未定義コードまたは PC を書き換える命令がデコードされると開始される

例外処理が起動されると、CPU は次のように作動します。

#### (1) リセットによる例外処理

プログラムカウンタ (PC) とスタックポインタ (SP) の初期値を例外処理ベクタテーブル (PC、SP をそれぞれ、パワーオンリセット時は H'00000000 番地、H'00000004 番地、またマニュアルリセット時は H'00000008 番地、H'0000000C 番地) から取り出します。例外処理ベクタテーブルについては「4.1.3 例外処理ベクタテーブル」を参照してください。次にベクタベースレジスタ (VBR) を 0 に、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) を 1111 にセットします。例外処理ベクタテーブルから取り出した PC のアドレスからプログラムの実行を開始します。

#### (2) アドレスエラー、割り込み、命令による例外処理

SR と PC を R15 で示すスタック上に退避します。割り込み例外処理の場合、割り込み優先レベルを SR の割り込みマスクビット (I3 ~ I0) に書き込みます。アドレスエラー、命令による例外処理の場合、I3 ~ I0 ビットは影響を受けません。次に例外処理ベクタテーブルからスタートアドレスを取り出し、そのアドレスからプログラムの実行を開始します。

### 4.1.3 例外処理ベクタテーブル

例外処理実行前には、あらかじめ例外処理ベクタテーブルが、メモリ上に設定されている必要があります。例外処理ベクタテーブルには、例外サービスルーチンの開始アドレスを格納しておきます（リセット例外処理のテーブルには、PC と SP の初期値を格納しておきます）。

各例外要因には、それぞれ異なるベクタ番号とベクタテーブルアドレスオフセットが割り当てられています。ベクタテーブルアドレスは、対応するベクタ番号やベクタテーブルアドレスオフセットから算出されます。例外処理では、このベクタテーブルアドレスが示す例外処理ベクタテーブルから、例外サービスルーチンのスタートアドレスが取り出されます。

ベクタ番号とベクタテーブルアドレスオフセットを表 4.3 に、ベクタテーブルアドレスの算出法を表 4.4 に示します。

表 4.3 (a) 例外処理ベクタテーブル

例外要因		ベクタ番号	ベクタテーブルアドレスオフセット	ベクタアドレス
パワーオンリセット	PC	0	H'00000000 ~ H'00000003	ベクタ番号 × 4
	SP	1	H'00000004 ~ H'00000007	
マニュアルリセット	PC	2	H'00000008 ~ H'0000000B	
	SP	3	H'0000000C ~ H'0000000F	
一般不当命令		4	H'00000010 ~ H'00000013	VBR + (ベクタ番号 × 4)
(システム予約)		5	H'00000014 ~ H'00000017	
スロット不当命令		6	H'00000018 ~ H'0000001B	
(システム予約)		7	H'0000001C ~ H'0000001F	
		8	H'00000020 ~ H'00000023	
CPU アドレスエラー		9	H'00000024 ~ H'00000027	
DMA アドレスエラー (DMAC および E-DMAC)		10* <sup>5</sup>	H'00000028 ~ H'0000002B	
割り込み	NMI	11	H'0000002C ~ H'0000002F	
	ユーザブレイク	12	H'00000030 ~ H'00000033	
	H-UDI	13	H'00000034 ~ H'00000037	
(システム予約)		14	H'00000038 ~ H'0000003B	
		: : :		
トラップ命令 (ユーザベクタ)		31	H'0000007C ~ H'0000007F	
		: : :		
		32	H'00000080 ~ H'00000083	
		: : :		
		63	H'000000FC ~ H'000000FF	

#### 4. 例外処理

表 4.3 (b) 例外処理ベクタテーブル (IRQ モード)

例外要因		ベクタ番号	ベクタテーブルアドレスオフセット	ベクタアドレス
割り込み	IRQ0	64* <sup>2</sup>	H'00000100 ~ H'00000103	
	IRQ1	65* <sup>2</sup>	H'00000104 ~ H'00000107	
	IRQ2	66* <sup>2</sup>	H'00000108 ~ H'0000010B	
	IRQ3	67* <sup>2</sup>	H'0000010C ~ H'0000010F	
	内蔵周辺 モジュール* <sup>3</sup>	0* <sup>4</sup> : 127* <sup>4</sup>	H'00000000 ~ H'00000003 : : : H'000001FC ~ H'000001FF	

表 4.3 (c) 例外処理ベクタテーブル (IRL モード)

例外要因		ベクタ番号	ベクタテーブルアドレスオフセット	ベクタアドレス
割り込み	IRL1* <sup>1</sup>	64* <sup>2</sup>	H'00000100 ~ H'00000103	VBR + (ベクタ番号 × 4)
	IRL2* <sup>1</sup>			
	IRL3* <sup>1</sup>	65* <sup>2</sup>	H'00000104 ~ H'00000107	
	IRL4* <sup>1</sup>			
	IRL5* <sup>1</sup>			
	IRL6* <sup>1</sup>	66* <sup>2</sup>	H'00000108 ~ H'0000010B	
	IRL7* <sup>1</sup>			
	IRL8* <sup>1</sup>	67* <sup>2</sup>	H'0000010C ~ H'0000010F	
	IRL9* <sup>1</sup>			
	IRL10* <sup>1</sup>	68* <sup>2</sup>	H'00000110 ~ H'00000113	
	IRL11* <sup>1</sup>			
	IRL12* <sup>1</sup>			
	IRL13* <sup>1</sup>	69* <sup>2</sup>	H'00000114 ~ H'00000117	
	IRL14* <sup>1</sup>			
	IRL15* <sup>1</sup>	70* <sup>2</sup>	H'00000118 ~ H'0000011B	
IRL16* <sup>1</sup>				
IRL17* <sup>1</sup>	71* <sup>2</sup>	H'0000011C ~ H'0000011F		
IRL18* <sup>1</sup>				
内蔵周辺 モジュール* <sup>3</sup>	0* <sup>4</sup> : 127* <sup>4</sup>	H'00000000 ~ H'00000003 : : : H'000001FC ~ H'000001FF		

- 【注】 \*1 IRL3、IRL2、IRL1、IRL0 端子に "1110" を入力したとき、IRL1 割り込みになります。"0000" のとき、IRL15 割り込みになります。
- \*2 この表のオートベクタ番号を使わずに、外部ベクタ番号フェッチも可能です。
- \*3 各内蔵周辺モジュール割り込みのベクタ番号とベクタテーブルアドレスオフセットは「5. 割り込みコントローラ (INTC)」の「表 5.4 割り込み例外処理ベクタと優先順位」を参照してください。
- \*4 ベクタ番号は内蔵ベクタ番号を設定するレジスタで設定します。詳細は、「5. 割り込みコントローラ (INTC)」の「5.3 レジスタの説明」および「11. ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。
- \*5 DMAC による DMA アドレスエラーが発生した場合および、E-DMAC による DMA アドレスエラーが発生した場合、同一のベクタ番号 "10" が生成されます。したがって例外サービスルーチンで「DMAC 内 DMA オペレーションレジスタ (DMAOR) のアドレスエラーフラグビット (AE)」と「E-DMAC 内 E-DMAC 動作制御レジスタ (EDOCR) のアドレスエラー制御ビット (AEC)」を読み出し、どちらの DMA アドレスエラーが発生したかを判断してください。

表 4.4 例外処理ベクタテーブルアドレスの算出法

例外要因	ベクタテーブルアドレス算出法
パワーオンリセット	(ベクタテーブルアドレス) = (ベクタテーブルアドレスオフセット)
マニュアルリセット	= (ベクタ番号) × 4
その他の例外処理	(ベクタテーブルアドレス) = VBR + (ベクタテーブルアドレスオフセット) = VBR + (ベクタ番号) × 4

【注】 VBR : ベクタベースレジスタ  
ベクタテーブルアドレスオフセット : 表 4.3 を参照  
ベクタ番号 : 表 4.3 を参照

## 4.2 リセット

### 4.2.1 リセットの種類

リセットは最も優先順位の高い例外要因です。リセットには、パワーオンリセットとマニュアルリセットの2種類があります。表 4.5 に示すように、パワーオンリセット、マニュアルリセットのどちらでも CPU の内部状態は初期化されます。また、パワーオンリセットでは内蔵周辺モジュールのレジスタがすべて初期化されるのに対し、マニュアルリセットでは、バスステートコントローラ (BSC)、ユーザブレイクコントローラ (UBC)、ピンファンクションコントローラ (PFC)、周波数変更レジスタを除いた内蔵周辺モジュールのレジスタが初期化されます (電源立ち上げ時には、パワーオンリセットを使用してください)。

表 4.5 リセットの種類

種類	リセット状態への遷移条件		内部状態	
	NMI	RES	CPU	内蔵周辺モジュール
パワーオンリセット	ハイレベル	ローレベル	初期化	初期化
マニュアルリセット	ローレベル	ローレベル	初期化	BSC、UBC、PFC、周波数変更レジスタ以外初期化

### 4.2.2 パワーオンリセット

NMI 端子がハイレベルのとき  $\overline{\text{RES}}$  端子をローレベルにすると、本 LSI はパワーオンリセット状態になります。本 LSI を確実にリセットするために最低、PLL 回路が停止している場合は、発振安定時間の間、PLL 回路が動作している場合は 20t<sub>pcyc</sub> の間  $\overline{\text{RES}}$  端子をローレベルに保持してください。パワーオンリセット状態では、CPU の内部状態と内蔵周辺モジュールのレジスタがすべて初期化されます。パワーオンリセット状態での各端子の状態は「付録 B. 端子状態」を参照してください。

パワーオンリセット状態で、NMI 端子をハイレベルに保持したまま  $\overline{\text{RES}}$  端子を一定期間ローレベルに保持した後ハイレベルにすると、パワーオンリセット例外処理が開始されます。このとき、CPU は次のように動作します。

1. プログラムカウンタ (PC) の初期値 (実行開始アドレス) を、例外処理ベクタテーブルから取り出します。
2. スタックポインタ (SP) の初期値を、例外処理ベクタテーブルから取り出します。
3. ベクタベースレジスタ (VBR) を H'00000000 にクリアし、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) を HF (1111) にセットします。
4. 例外処理ベクタテーブルから取り出した値をそれぞれプログラムカウンタ (PC) とスタックポインタ (SP) に設定し、プログラムの実行を開始します。

### 4.2.3 マニュアルリセット

NMI 端子がローレベルのとき、 $\overline{\text{RES}}$  端子をローレベルにすると本 LSI はマニュアルリセット状態になります。本 LSI を確実にリセットするために最低 20 クロックサイクルの間  $\overline{\text{RES}}$  端子をローレベルに保持してください。

---

マニュアルリセット状態では、CPU の内部状態が初期化されます。また、バステートコントローラ (BSC)、ユーザブ레이크コントローラ (UBC)、ピンファンクションコントローラ (PFC)、周波数変更レジスタを除いた内蔵周辺モジュールの各レジスタが初期化されます。バスサイクルの途中でマニュアルリセット状態にすると、バスサイクルの終了を待ってからマニュアルリセット例外処理を開始します。したがって、マニュアルリセットによってバスサイクルが途中で止まることはありません。マニュアルリセット状態での各端子の状態は「付録 B. 端子状態」を参照してください。

マニュアルリセット状態で、NMI 端子をローレベルに保持したまま  $\overline{\text{RES}}$  端子を一定期間ローレベルに保持した後ハイレベルにすると、マニュアルリセット例外処理が開始されます。このとき、CPU はパワーオンリセット例外処理と同じ手順で動作します。

### 4.3 アドレスエラー

#### 4.3.1 アドレスエラー発生要因

アドレスエラーは、表 4.6 に示すように命令フェッチ、データ読み出し / 書き込み時に発生します。

表 4.6 バスサイクルとアドレスエラー

バスサイクル		バスサイクルの内容	アドレスエラー発生
種類	バスマスタ		
命令 フェッチ	CPU	偶数アドレスから命令フェッチ	なし (正常)
		奇数アドレスから命令フェッチ	アドレスエラー発生
		内蔵周辺モジュール空間以外から命令をフェッチ	なし (正常)
		内蔵周辺モジュール空間から命令をフェッチ	アドレスエラー発生
データ 読み出し / 書き込み	CPU、DMAC、 または E-DMAC	ワードデータを偶数アドレスからアクセス	なし (正常)
		ワードデータを奇数アドレスからアクセス	アドレスエラー発生
		ロングワードデータをロングワード境界からアクセス	なし (正常)
		ロングワードデータをロングワード境界以外からアクセス	アドレスエラー発生
		キャッシュバージ空間、アドレスアレイリードライト空間、内蔵 I/O 空間から PC 相対アドレッシングでアクセス	アドレスエラー発生
		キャッシュバージ空間、アドレスアレイリードライト空間、データアレイリードライト空間、内蔵 I/O 空間に対して TAS.B 命令でアクセス	アドレスエラー発生
		アドレスが H'FFFFFFC00 ~ H'FFFFFFCFF の内蔵周辺モジュール空間のバイト、ワード、ロングワードデータをアクセス	なし (正常)
		アドレスが H'FFFFFFE00 ~ H'FFFFFFE0F の内蔵周辺モジュール空間のロングワードデータをアクセス	アドレスエラー発生
		アドレスが H'FFFFFFE00 ~ H'FFFFFFE0F の内蔵周辺モジュール空間のワード、バイトデータをアクセス	なし (正常)
		アドレスが H'FFFFFFF00 ~ H'FFFFFFF0F の内蔵周辺モジュール空間のバイトデータをアクセス	アドレスエラー発生
		アドレスが H'FFFFFFF00 ~ H'FFFFFFF0F の内蔵周辺モジュール空間のワード、ロングワードデータをアクセス	なし (正常)

- 【注】
1. シンクロナス DRAM のモードレジスタ書き込みサイクルはアドレスエラーは発生しません。
  2. ダイレクトメモリアクセスコントローラ (DMAC) の 16 バイト転送は、ロングワードデータのアクセスで行っています。

#### 4.3.2 アドレスエラー例外処理

アドレスエラーが発生すると、アドレスエラーを起こしたバスサイクルが終了し、実行中の命令が完了してからアドレスエラー例外処理が開始されます。このとき、CPU は次のように動作します。

1. ステータスレジスタ (SR) をスタックに退避します。
2. プログラムカウンタ (PC) をスタックに退避します。退避する PC の値は、最後に実行した命令の次命令の先頭アドレスです。

3. 発生したアドレスエラーに対応する例外処理ベクタテーブルから例外サービスルーチン開始アドレスを取り出し、そのアドレスから、プログラムを実行します。このときのジャンプは遅延分岐ではありません。

【注】 DMAC による DMA アドレスエラーが発生した場合および E-DMAC による DMA アドレスエラーが発生した場合、同一のベクタ番号 "10" が生成されます。(表 4.3 (a) を参照)

したがって、例外サービスルーチンで「DMAC 内 DMA オペレーションレジスタ (DMAOR) のアドレスエラーフラグ (AE)」と「E-DMAC 内 E-DMAC 動作制御レジスタ (EDOCR) のアドレスエラー制御ビット (AEC)」を読み出し、どちらの DMA アドレスエラーが発生したかを判断してください。

## 4. 例外処理

### 4.4 割り込み

#### 4.4.1 割り込み要因

割り込み例外処理を起動させる要因には、表 4.7 に示すように NMI、ユーザブレイク、H-UDI、IRL、IRQ、内蔵周辺モジュールがあります。

表 4.7 割り込み要因

種類	要求元	要因数
NMI	NMI 端子 (外部からの入力)	1
ユーザブレイク	ユーザブレイクコントローラ (UBC)	1
H-UDI	ハイパフォーマンスユーザデバッグインタフェース (H-UDI)	1
IRL	IRL1 ~ IRL15 (外部からの入力)	15
IRQ	IRQ0 ~ IRQ3 (外部からの入力)	4
内蔵周辺モジュール	ダイレクトメモリアクセスコントローラ (DMAC)	2
	イーサネットコントローラ (EtherC) およびイーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)	1
	16 ビットフリーランニングタイマ (FRT)	3
	ウォッチドッグタイマ (WDT)	1
	バスステートコントローラ (BSC)	1
	シリアル I/O (SIO)	4
	FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)	4
	16 ビットタイマパルスユニット (TPU)	13

各割り込み要因には、それぞれ異なるベクタ番号とベクタテーブルアドレスオフセットが割り当てられます。ベクタ番号とベクタテーブルアドレスオフセットについては「5. 割り込みコントローラ (INTC)」の「表 5.4 割り込み例外処理ベクタと優先順位」を参照してください。

#### 4.4.2 割り込み優先順位

割り込み要因には優先順位が設けられており、複数の割り込みが同時に発生した場合（多重割り込み）、割り込みコントローラ（INTC）によって優先順位が判定され、その判定結果に従って例外処理が起動されます。

割り込み要因の優先順位は、優先レベル0～16の値で表され、優先レベル0が最低で、優先レベル16が最高です。NMI割り込みは、優先レベル16のマスクできない最優先の割り込みで、常に受け付けられます。ユーザブ레이크割り込みの優先レベルは15です。IRL割り込みの優先レベルは1～15です。IRQ割り込みと内蔵周辺モジュール割り込みの優先レベルは、INTCの割り込み優先レベル設定レジスタA～E（IPRA～IPRE）に自由に設定することができます（表4.8）。設定できる優先レベルは0～15で、優先レベル16は設定できません。IPRA～Eの詳細については「5.3.1 割り込み優先レベル設定レジスタA（IPRA）」～「5.3.5 割り込み優先レベル設定レジスタE（IPRE）」を参照してください。

表 4.8 割り込み優先順位

種類	優先レベル	備考
NMI	16	優先レベル固定、マスク不可能
ユーザブ레이크	15	優先レベル固定
H-UDI	15	優先レベル固定
IRL	1～15	IRL3～IRL0端子で設定
IRQ	0～15	割り込み優先レベル設定レジスタC（IPRC）
内蔵周辺モジュール	0～15	割り込み優先レベル設定レジスタA、B、D、E（IPRA、B、D、E）に設定

#### 4.4.3 割り込み例外処理

割り込みが発生すると、割り込みコントローラ（INTC）によって優先順位が判定されます。NMIは常に受け付けられますが、それ以外の割り込みは、その優先レベルがステータスレジスタ（SR）の割り込みマスクビット（I3～I0）に設定されている優先レベルより高い場合だけ受け付けられます。

割り込みが受け付けられると割り込み例外処理が開始されます。割り込み例外処理では、CPUはSRとプログラムカウンタ（PC）をスタックに退避し、受け付けた割り込みの優先レベル値をSRのI3～I0ビットに書き込みます。ただし、NMIの場合、優先レベルは16ですが、I3～I0ビットに設定される値はHF（レベル15）です。次に、受け付けた割り込みに対応する例外処理ベクタテーブルから、例外サービスルーチン開始アドレスを取り出し、そのアドレスにジャンプして、実行を開始します。割り込み例外処理の詳細については、「5.4 動作説明」を参照してください。

## 4. 例外処理

---

### 4.5 命令による例外

#### 4.5.1 命令による例外の種類

例外処理を起動する命令には、表 4.9 に示すように、トラップ命令、スロット不当命令、一般不当命令があります。

表 4.9 命令による例外の種類

種類	要因となる命令	備考
トラップ命令	TRAPA	
スロット不当命令	遅延分岐命令直後（遅延スロット）に配置された未定義コードまたは PC を書き換える命令	遅延分岐命令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA PC を書き換える命令：JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRA
一般不当命令	遅延スロット以外にある未定義コード	

#### 4.5.2 トラップ命令

TRAPA 命令を実行すると、トラップ命令例外処理が開始されます。このとき、CPU は次のように動作します。

1. ステータスレジスタ（SR）をスタックに退避します。
2. プログラムカウンタ（PC）をスタックに退避します。退避する PC の値は、TRAPA 命令の次命令の先頭アドレスです。
3. 例外サービスルーチン開始アドレスを、TRAPA 命令で指定したベクタ番号に対応する例外処理ベクタテーブルから取り出し、そのアドレスにジャンプして、プログラムの実行を開始します。このときのジャンプは遅延分岐ではありません。

#### 4.5.3 スロット不当命令

遅延分岐命令の直後に配置された命令のことを「遅延スロットに配置された命令」と呼びます。遅延スロットに配置された命令が未定義コードのとき、この未定義コードがデコードされるとスロット不当命令例外処理が開始されます。また、遅延スロットに配置された命令がプログラムカウンタ（PC）を書き換える命令のときも、この PC を書き換える命令がデコードされるとスロット不当命令例外処理が開始されます。スロット不当命令例外処理のとき、CPU は次のように動作します。

1. ステータスレジスタ（SR）をスタックに退避します。
2. PC をスタックに退避します。退避する PC の値は、未定義コードまたは PC を書き換える命令の直前にある遅延分岐命令の飛び先アドレスです。

3. 例外サービスルーチンスタートアドレスを、発生した例外に対応する例外処理ベクタテーブルから取り出し、そのアドレスにジャンプして、プログラムの実行を開始します。このときのジャンプは遅延分岐ではありません。

#### 4.5.4 一般不当命令

遅延分岐命令の直後（遅延スロット）以外に配置された未定義コードをデコードすると、一般不当命令例外処理が開始されます。このとき、CPU はスロット不当命令例外処理と同じ手順で動作します。ただし、退避するプログラムカウンタ（PC）の値は、スロット不当命令例外処理と異なり、この未定義コードの先頭アドレスになります。

## 4.6 例外要因が受け付けられない場合

アドレスエラーと割り込みは、表 4.10 に示すように、遅延分岐命令や割り込み禁止命令の直後に発生すると、すぐに受け付けられず保留される場合があります。この場合、例外を受け付けられる命令がデコードされたときに受け付けられます。

表 4.10 遅延分岐命令、割り込み禁止命令の直後の例外要因発生

発生した時点	例外要因	
	アドレスエラー	割り込み
遅延分岐命令 *1 の直後	×	×
割り込み禁止命令 *2 の直後		×
3 つ以下の命令で構成される繰り返しループ（命令フェッチサイクルは発生しない） 4 つ以上の命令を持つ繰り返しループにある最初の命令または最後の 3 つの命令	×	×
4 つ以上の命令を持つ繰り返しループにある最後から 4 つ目の命令		×

【注】 : 受け付けられる

× : 受け付けられない

\*1 遅延分岐命令 : JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA<sub>F</sub>

\*2 割り込み禁止命令 : LDC、LDC.L、STC、STC.L、LDS、LDS.L、STS、STS.L

### 4.6.1 遅延分岐命令の直後

遅延分岐命令直後（遅延スロット）に配置されている命令のデコード時は、アドレスエラーと割り込みの両方とも受け付けられません。遅延分岐命令とその直後（遅延スロット）にある命令は、必ず連続して実行され、この間に例外処理が行われることはありません。

### 4.6.2 割り込み禁止命令の直後

割り込み禁止命令直後の命令のデコード時は、割り込みは受け付けられません。アドレスエラーは受け付けられます。

### 4.6.3 繰り返しループにおける命令

繰り返しループが3つ以下の命令で構成される場合は、例外も割り込みも受け付けられません。繰り返しループに4つ以上の命令がある場合は、繰り返しループの最初の命令または最後の3つの命令の実行サイクル中、例外も割り込みも受け付けられません。繰り返しループに4つ以上の命令がある場合は、繰り返しループの最後から4つ目の命令の実行サイクル中、アドレスエラーのみ受け付けられます。詳細については、「SH-1、SH-2、SH-DSP プログラミングマニュアル」を参照してください。

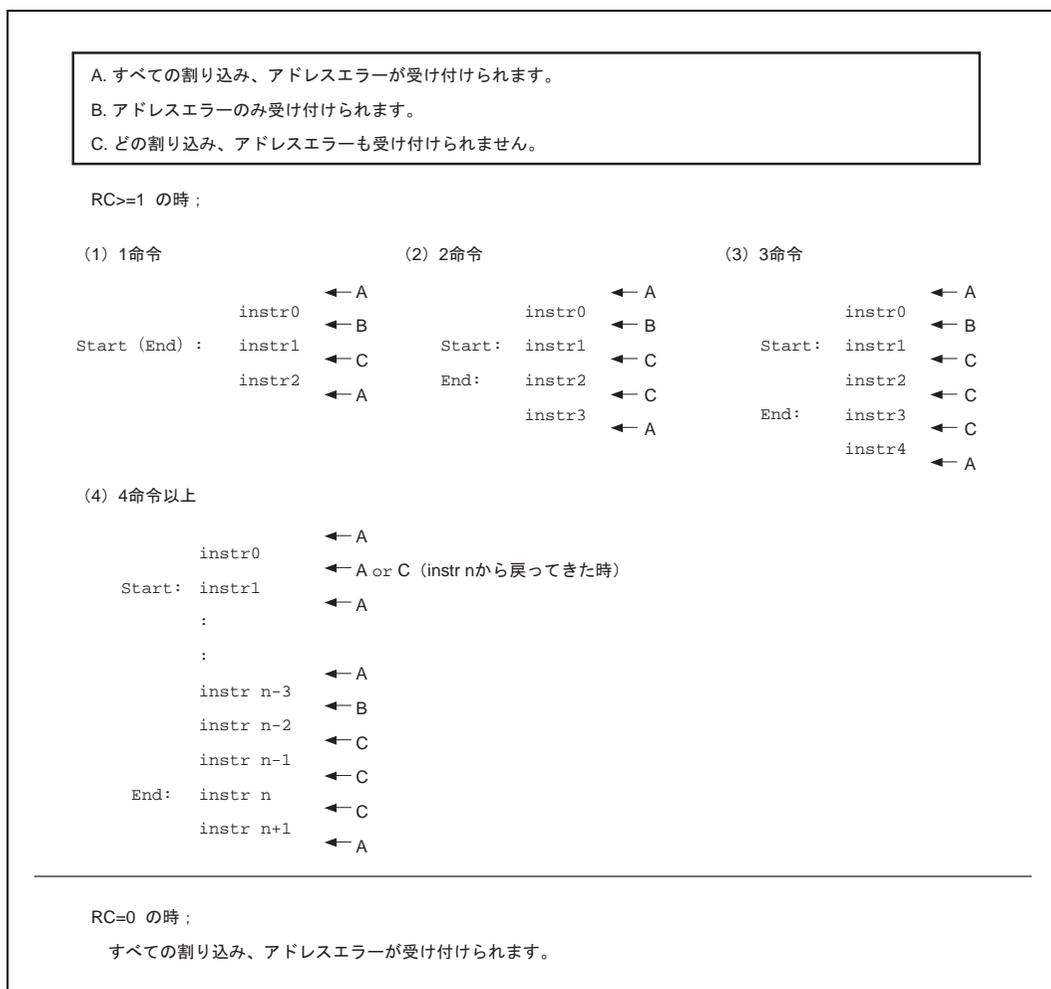


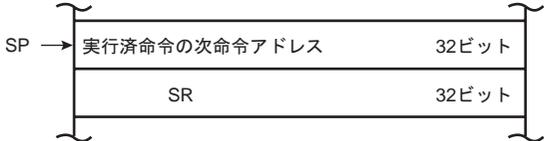
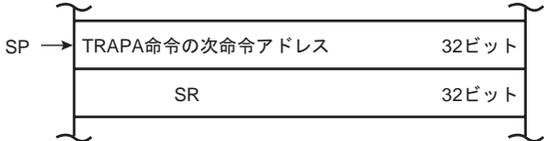
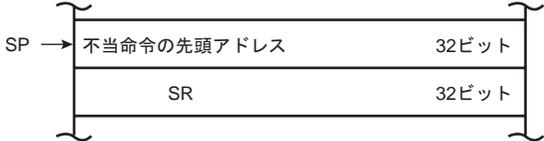
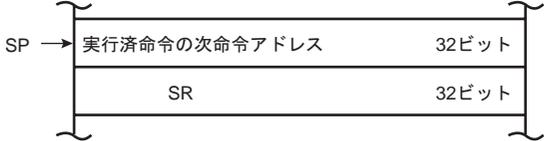
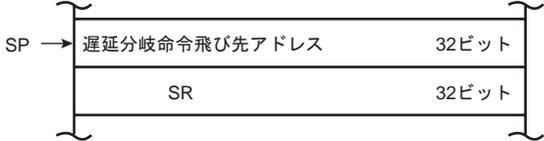
図 4.1 リピートモードにおける割り込み受け付けの制限

#### 4. 例外処理

### 4.7 例外処理終了後のスタックの状態

例外処理終了後のスタックの状態は、表 4.11 に示すようになります。

表 4.11 例外処理終了後のスタックの状態

種類	スタックの状態
アドレスエラー	 <p>SP → 実行済命令の次命令アドレス 32ビット</p> <p>SR 32ビット</p>
トラップ命令	 <p>SP → TRAPA命令の次命令アドレス 32ビット</p> <p>SR 32ビット</p>
一般不当命令	 <p>SP → 不当命令の先頭アドレス 32ビット</p> <p>SR 32ビット</p>
割り込み	 <p>SP → 実行済命令の次命令アドレス 32ビット</p> <p>SR 32ビット</p>
スロット不当命令	 <p>SP → 遅延分岐命令飛び先アドレス 32ビット</p> <p>SR 32ビット</p>

## 4.8 使用上の注意

### 4.8.1 スタックポインタ (SP) の値

スタックポインタ (SP) の値は必ず 4 の倍数になるようにしてください。SP が 4 の倍数以外るとき、例外処理でスタックがアクセスされると、アドレスエラーが発生します。

### 4.8.2 ベクタベースレジスタ (VBR) の値

ベクタベースレジスタ (VBR) の値は必ず 4 の倍数になるようにしてください。VBR が 4 の倍数以外るとき、例外処理でベクタテーブルがアクセスされると、アドレスエラーが発生します。

### 4.8.3 アドレスエラー例外処理のスタッキングで発生するアドレスエラー

スタックポインタ (SP) が 4 の倍数になっていないと、例外処理 (割り込みなど) のスタッキングでアドレスエラーが発生し、その例外処理終了後、アドレスエラー例外処理に移ります。アドレスエラー例外処理でのスタッキングでも、アドレスエラーが発生しますが、無限にアドレスエラー例外処理によるスタッキングが続かないように、そのときのアドレスエラーは受け付けなくなっています。これにより、プログラムの制御をアドレスエラー例外サービスルーチンに移すことができ、エラー処理を行うことができます。

なお、例外処理のスタッキングでアドレスエラーが発生した場合、スタッキングのバスサイクル (ライト) は実行されます。ステータスレジスタ (SR) とプログラムカウンタ (PC) のスタッキングでは、SP がそれぞれ -4 されますので、スタッキング終了後も SP の値は 4 の倍数になっていません。また、スタッキング時に出力されるアドレスの値は SP の値で、エラーの発生したアドレスそのものが出力されます。このとき、スタッキングされたライトデータは不定です。

### 4.8.4 マニュアルリセット時のレジスタアクセス

バスステートコントローラ (BSC)、ユーザブレイクコントローラ (UBC)、ピンファクションコントローラ (PFC) のレジスタ、周波数変更レジスタをアクセス中にマニュアルリセットを入れないでください。誤った書き込みがされることがあります。

#### 4. 例外処理

---

---

## 5. 割り込みコントローラ (INTC)

---

### 5.1 概要

割り込みコントローラ (INTC) は、割り込み要因の優先順位を判定し、CPU への割り込み要求を制御します。INTC には、各割り込みの優先順位を設定するためのレジスタがあり、これにより、ユーザが設定した優先順位に従って、割り込み要求を処理させることができます。

#### 5.1.1 特長

INTC には、次の特長があります。

- 割り込み優先順位を 16 レベル設定可能  
5 本の割り込み優先レベル設定レジスタにより、内蔵周辺モジュール割り込みの優先順位を要求元別に 16 レベルまで設定することができます。
- 内蔵周辺モジュール割り込みベクタ番号を設定可能  
24 本のベクタ番号設定レジスタにより、内蔵周辺モジュール割り込みのベクタ番号を割り込み要因別に 0 ~ 127 の値で設定することができます。
- IRL 割り込みのベクタ番号設定の方法を選択可能  
内部で決めてあるベクタ番号を使うオートベクタモードと、外部端子からベクタ番号を設定する外部ベクタモードがあり、レジスタを設定することで選択できます。
- IRQ 割り込みを設定可能 (ローレベル、立ち上がり、立ち下がり、両エッジ検出)

## 5. 割り込みコントローラ (INTC)

### 5.1.2 ブロック図

INTCのブロック図を図5.1に示します。

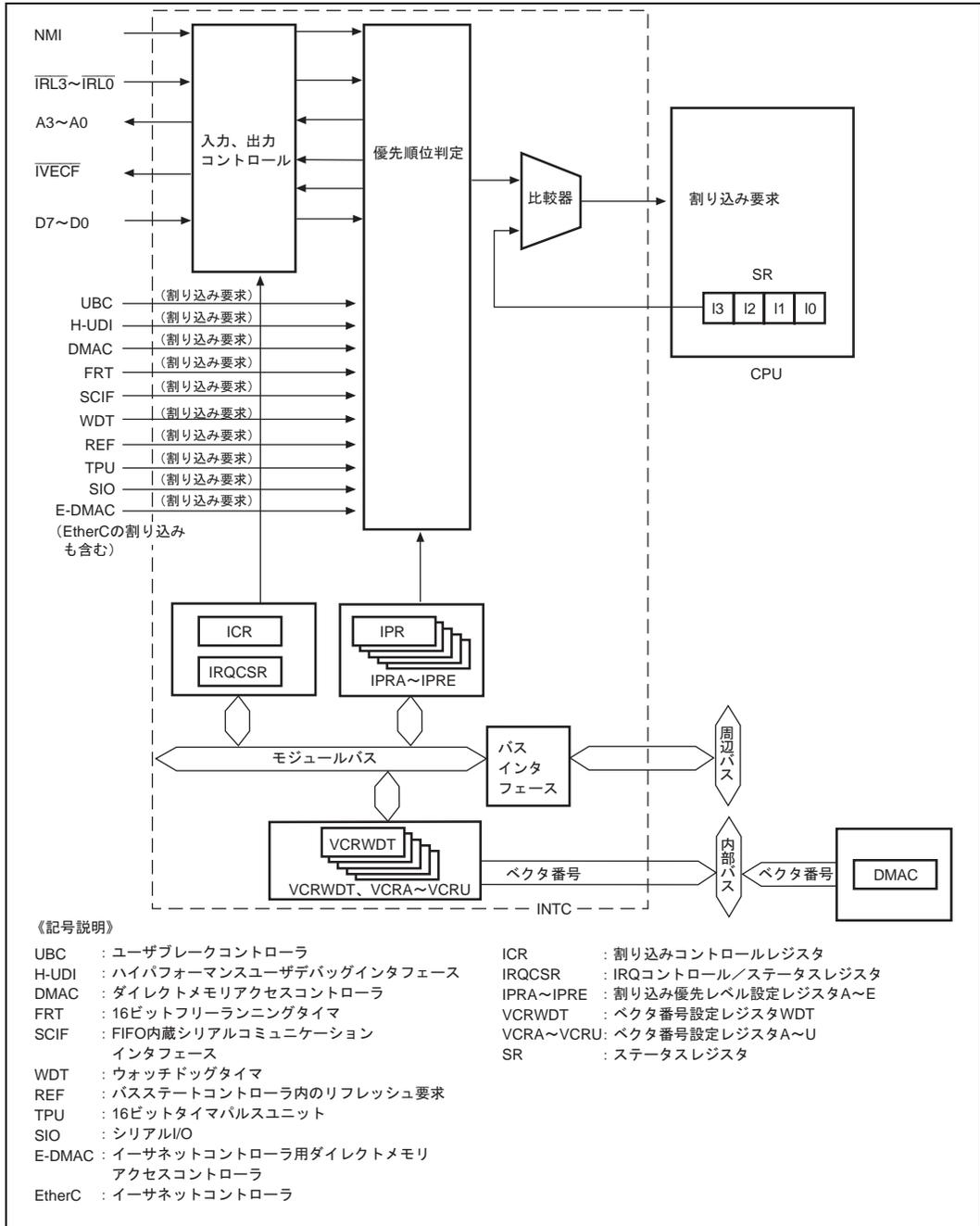


図 5.1 INTC のブロック図

### 5.1.3 端子構成

INTC の端子を表 5.1 に示します。

表 5.1 端子構成

名称	略称	入出力	機能
ノンマスクابل割り込み入力端子	NMI	入力	マスク不可能な割り込み要求信号を入力
レベル要求割り込み入力端子	IRL3 ~ IRL0	入力	マスク可能な割り込み要求信号を入力
受け付け割り込みレベル出力端子	A3 ~ A0	出力	外部ベクタモード時、IRL/IRQ 割り込みを受け付けたとき、その割り込みレベル信号を出力
外部ベクタフェッチ端子	IVECF	出力	外部ベクタのリードサイクルであることを表示
外部ベクタ番号入力端子	D7 ~ D0	入力	外部ベクタ番号を入力

### 5.1.4 レジスタ構成

INTC には、表 5.2 に示すように 31 本のレジスタがあります。これらのレジスタにより、割り込み優先順位の設定や、外部割り込み入力信号の検出制御などを行います。

## 5. 割り込みコントローラ (INTC)

表 5.2 レジスタ構成

名称	略称	R/W	初期値	アドレス	アクセスサイズ
割り込み優先レベル設定レジスタ A	IPRA	R/W	H'0000	H'FFFFFFE2	8、16
割り込み優先レベル設定レジスタ B	IPRB	R/W	H'0000	H'FFFFFFE6	8、16
割り込み優先レベル設定レジスタ C	IPRC	R/W	H'0000	H'FFFFFFEA	8、16
割り込み優先レベル設定レジスタ D	IPRD	R/W	H'0000	H'FFFFFFEE	8、16
割り込み優先レベル設定レジスタ E	IPRE	R/W	H'0000	H'FFFFFFF2	8、16
ベクタ番号設定レジスタ A	VCRA	R/W	H'0000	H'FFFFFFE2	8、16
ベクタ番号設定レジスタ B* <sup>3</sup>	VCRB	R/W	H'0000	H'FFFFFFE4	8、16
ベクタ番号設定レジスタ C	VCRC	R/W	H'0000	H'FFFFFFE6	8、16
ベクタ番号設定レジスタ D	VCRD	R/W	H'0000	H'FFFFFFE8	8、16
ベクタ番号設定レジスタ E	VCRE	R/W	H'0000	H'FFFFFFE4	8、16
ベクタ番号設定レジスタ F	VCRF	R/W	H'0000	H'FFFFFFE8	8、16
ベクタ番号設定レジスタ G	VCRG	R/W	H'0000	H'FFFFFFEA	8、16
ベクタ番号設定レジスタ H	VCRH	R/W	H'0000	H'FFFFFFEC	8、16
ベクタ番号設定レジスタ I	VCRI	R/W	H'0000	H'FFFFFFE0	8、16
ベクタ番号設定レジスタ J	VCRJ	R/W	H'0000	H'FFFFFFE4	8、16
ベクタ番号設定レジスタ K	VCRK	R/W	H'0000	H'FFFFFFE8	8、16
ベクタ番号設定レジスタ L	VCRL	R/W	H'0000	H'FFFFFFE50	8、16
ベクタ番号設定レジスタ M	VCRM	R/W	H'0000	H'FFFFFFE52	8、16
ベクタ番号設定レジスタ N	VCRN	R/W	H'0000	H'FFFFFFE54	8、16
ベクタ番号設定レジスタ O	VCRO	R/W	H'0000	H'FFFFFFE56	8、16
ベクタ番号設定レジスタ P	VCRP	R/W	H'0000	H'FFFFFFE2	8、16
ベクタ番号設定レジスタ Q	VCRQ	R/W	H'0000	H'FFFFFFE4	8、16
ベクタ番号設定レジスタ R	VCRR	R/W	H'0000	H'FFFFFFE6	8、16
ベクタ番号設定レジスタ S	VCRS	R/W	H'0000	H'FFFFFFE8	8、16
ベクタ番号設定レジスタ T	VCRT	R/W	H'0000	H'FFFFFFEA	8、16
ベクタ番号設定レジスタ U	VCRU	R/W	H'0000	H'FFFFFFEC	8、16
ベクタ番号設定レジスタ WDT	VCRWDT	R/W	H'0000	H'FFFFFFE4	8、16
ベクタ番号設定レジスタ DMA0* <sup>4</sup>	VCRDMA0	R/W	不定	H'FFFFFFA0	32
ベクタ番号設定レジスタ DMA1* <sup>4</sup>	VCRDMA1	R/W	不定	H'FFFFFFA8	32
割り込みコントロールレジスタ	ICR	R/W	H'8000/ H'0000* <sup>1</sup>	H'FFFFFFE0	8、16
IRQ コントロール / ステータスレジスタ	IRQCSR	R/W	* <sup>2</sup>	H'FFFFFFE8	8、16

- 【注】
- \*1 NMI 端子がハイレベルのとき H'8000、NMI 端子がローレベルのとき H'0000
  - \*2 IRL3 ~ IRL0 端子がハイレベルのとき IRQCSR のビット 7 ~ 4 は 1 になります。IRL3 ~ IRL0 端子がローレベルのとき IRQCSR のビット 7 ~ 4 は 0 になります。ビット 7 ~ 4 以外は初期値 0 になります。
  - \*3 SH7615 では、VCRB を予約レジスタとし、本レジスタへのアクセスを禁止しています。
  - \*4 VCRDMA0、VCRDMA1 については、「11. ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。

## 5.2 割り込み要因

割り込み要因は、NMI、ユーザブ레이크、H-UDI、IRL/IRQ、内蔵周辺モジュールの5つに分類されます。各割り込みの優先順位は優先レベル値(0 ~ 16)で表され、レベル0が最低で、レベル16が最高です。レベル0に設定すると、その割り込みはマスクされます。

### 5.2.1 NMI 割り込み

NMI 割り込みは、レベル16の最優先の割り込みで、常に受け付けられます。NMI 端子からの入力はエッジで検出され、検出エッジは、割り込みコントロールレジスタ(ICR)のNMIエッジセレクトビット(NMIE)の設定によって、立ち上がりエッジまたは立ち下がりエッジを選択できます。

NMI 割り込み例外処理によって、ステータスレジスタ(SR)の割り込みマスクビット(I3 ~ I0)はレベル15に設定されます。

### 5.2.2 ユーザブ레이크割り込み

ユーザブ레이크割り込みは、ユーザブ레이크コントローラ(UBC)で設定したブ레이크条件が成立したときに発生する割り込みで、優先レベルは15です。ユーザブ레이크割り込み例外処理によって、ステータスレジスタ(SR)の割り込みマスクビット(I3 ~ I0)はレベル15に設定されます。

ユーザブ레이크の詳細は、「6. ユーザブ레이크コントローラ(UBC)」を参照してください。

### 5.2.3 H-UDI 割り込み

H-UDI 割り込みは、優先順位レベル15を持ち、H-UDI 割り込みのインストラクションをシリアル入力すると発生します。H-UDI 割り込み例外処理によって、SRの割り込みマスクビット(I3 ~ I0)はレベル15に設定されます。

H-UDI 割り込みの詳細は、「17. ハイパフォーマンスユーザデバッグインタフェース(H-UDI)」を参照してください。

### 5.2.4 IRL 割り込み

IRL 割り込みは、 $\overline{IRL3}$  ~  $\overline{IRL0}$  端子からの入力による割り込みです。 $\overline{IRL3}$  ~  $\overline{IRL0}$  端子から IRL15 ~ IRL1 の15個の割り込みを外部から入力することができます。IRL15 ~ IRL1 の各割り込みの優先レベル値は、それぞれ15 ~ 1で、ベクタ番号は71 ~ 64になります。ベクタ番号は、割り込みコントロールレジスタ(ICR)の割り込みベクタモードセレクト(VECMD)の設定によって、外部から入力することもできます。ベクタ番号の外部入力は、外部ベクタ入力端子(D7 ~ D0)から0 ~ 127のベクタ番号を入力します。外部ベクタを使用したとき、D7には0を入力します。内部のベクタをオートベクタと呼び、外部からのベクタを外部ベクタと呼びます。IRL 割り込みの優先レベルとオートベクタ番号を表5.3に示します。

外部ベクタモード時、IRL 割り込みを受け付けると、受け付け割り込みレベル出力端子(A3 ~ A0)からIRL 割り込みレベルを出力します。また、外部ベクタフェッチ端子( $\overline{IVECF}$ )をアサートします。このときD7 ~ D0から外部ベクタ番号を読み出します。

## 5. 割り込みコントローラ (INTC)

---

IRL 割り込み例外処理では、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) は、受け付けた IRL 割り込みの優先レベル値に設定されます。

### 5.2.5 IRQ 割り込み

ICR の外部割り込みベクタモードセレクトビット (EXIMD) を 1 に設定することで IRQ 割り込みになります。IRQ 割り込みは、 $\overline{\text{IRL3}} \sim \overline{\text{IRL0}}$  端子のどれか 1 本に対する入力に対応しています。IRQ コントロール / ステータスレジスタ (IRQCSR) の IRQ センスセレクトビット (IRQ3IS ~ IRQ00S) は、各端子に対して別々に Low レベル検出か、立ち下がり / 立ち上がり / 両エッジ検出かを選択することができます。割り込み優先レベル設定レジスタ C (IPRC) は、各端子に対して別々に 0 ~ 15 の優先順位レベルを選択することができます。ベクタ番号の外部入力を許可するためには、割り込みコントロールレジスタ (ICR) の割り込みベクタモードセレクトビット (VECMD) を設定してください。外部ベクタ番号は 0 ~ 127 の値で、割り込みベクタフェッチバスサイクル中に外部ベクタ入力端子 (D7 ~ D0) に入力します。外部ベクタを使用したとき、D7 には 0 を入力します。

IRQ 割り込みが外部ベクタモードで受け付けられると、IRQ 割り込み優先順位レベルが、割り込み受付レベル出力信号 (A3 ~ A0) から出力されます。外部ベクタフェッチ信号 ( $\overline{\text{IVECF}}$ ) もアサートされます。外部ベクタ番号は、このとき信号 D7 ~ D0 から読み出されます。

IRQ 割り込み例外処理では、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) は、受け付けた IRQ 割り込みの優先レベル値に設定されます。

表 5.3 IRL 割り込みの優先レベルとオートベクタ番号

端子				優先レベル	ベクタ番号
IRL3	IRL2	IRL1	IRL0		
0	0	0	0	15	71
0	0	0	1	14	
0	0	1	0	13	
0	0	1	1	12	
0	1	0	0	11	69
0	1	0	1	10	
0	1	1	0	9	68
0	1	1	1	8	
1	0	0	0	7	
1	0	0	1	6	67
1	0	1	0	5	
1	0	1	1	4	66
1	1	0	0	3	
1	1	0	1	2	
1	1	1	0	1	64

外部ベクタモード割り込みの接続例を図 5.2 に、オートベクタモード割り込みの接続例を図 5.3 に示します。

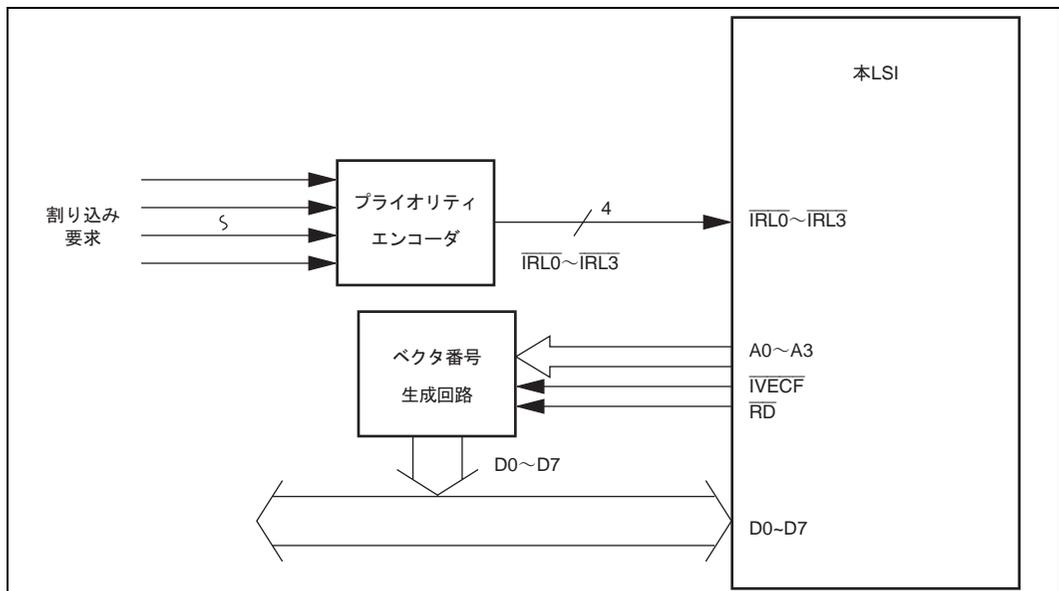


図 5.2 外部ベクタモード割り込みの接続例

## 5. 割り込みコントローラ (INTC)

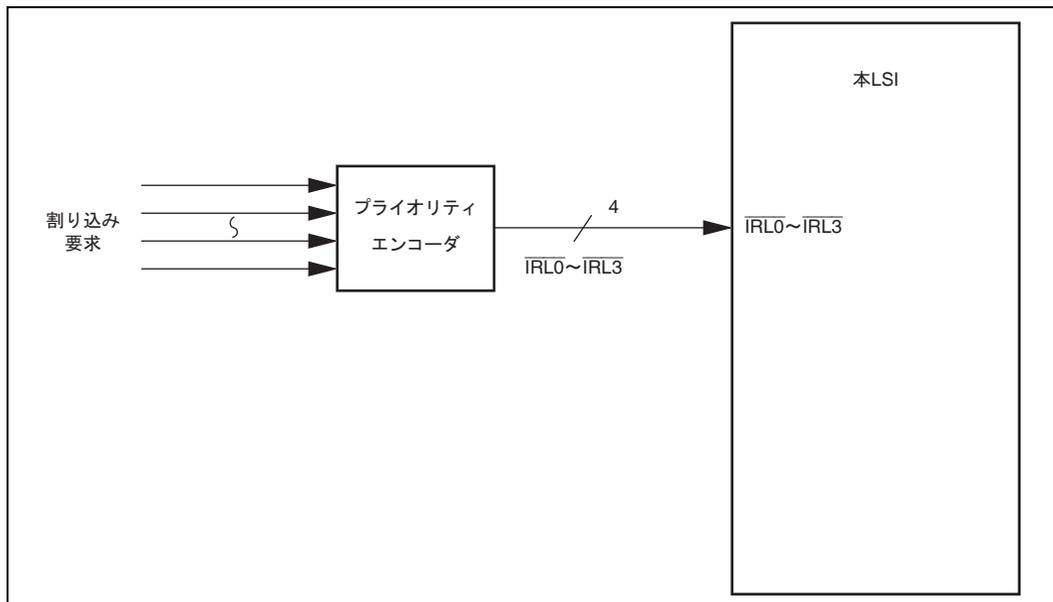


図 5.3 オートベクタモード割り込みの接続例

図 5.4 ~ 図 5.7 に、外部ベクタモードの割り込みベクタフェッチサイクルを示します。このサイクルでは  $\overline{CS0} \sim \overline{CS4}$  は 'H' 状態のままです。また A24 ~ A4 は不定値が出力されます。 $\overline{WAIT}$  端子はサンプリングされますが、プログラマブルウェイトは無効です。

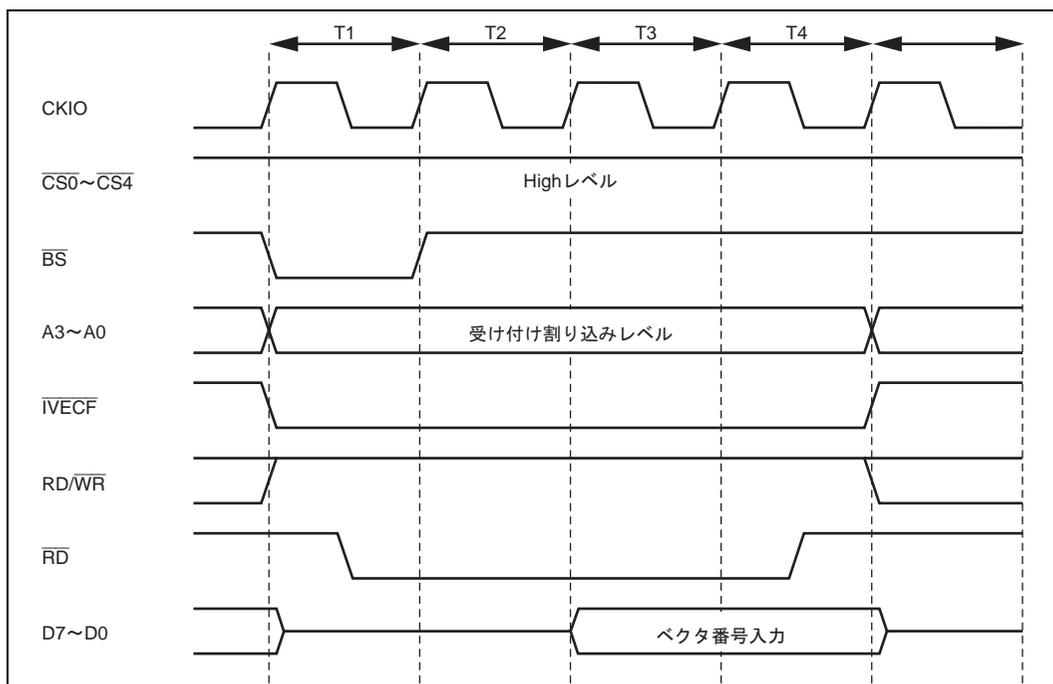


図 5.4 外部ベクタフェッチ (I : E = 1 : 1)

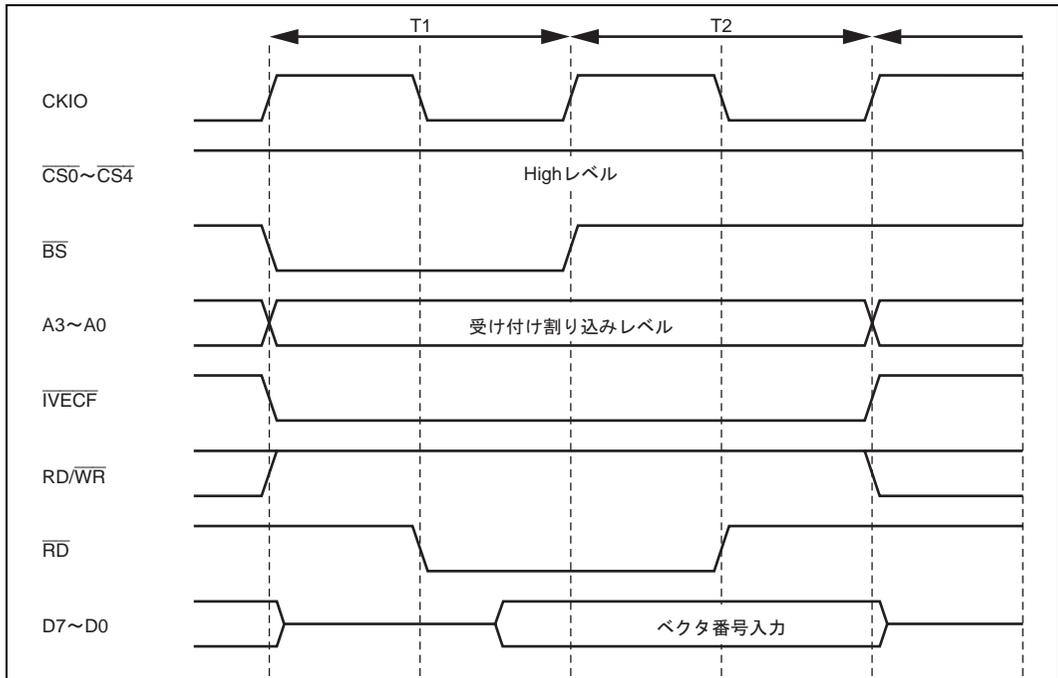


図 5.5 外部ベクタフェッチ (I : E = 1 : 1 以外)

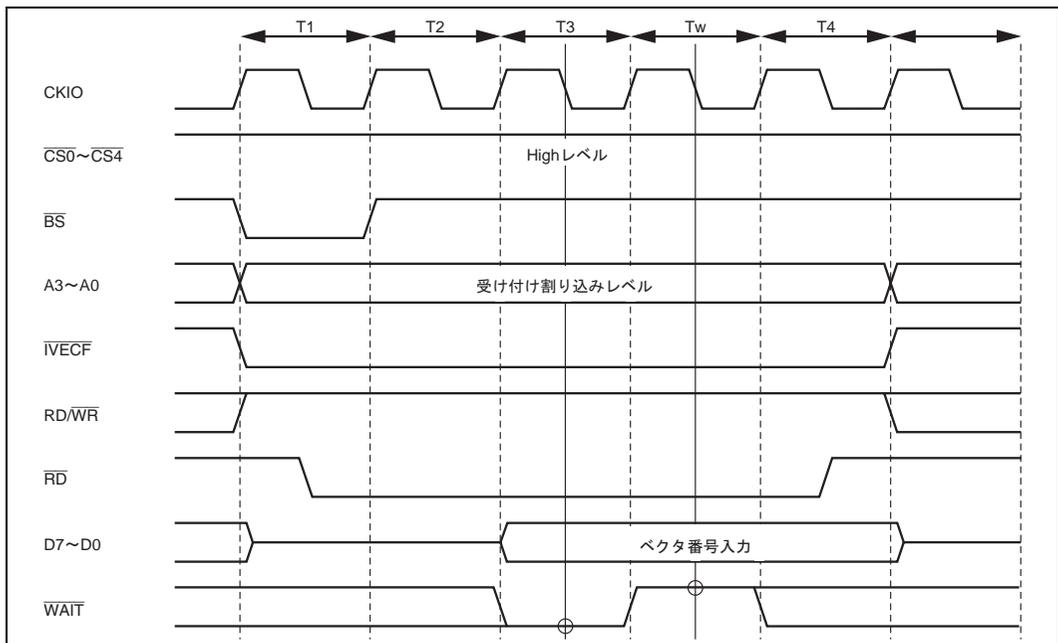


図 5.6 外部ベクタフェッチ (I : E = 1 : 1 (WAIT 入力))

## 5. 割り込みコントローラ (INTC)

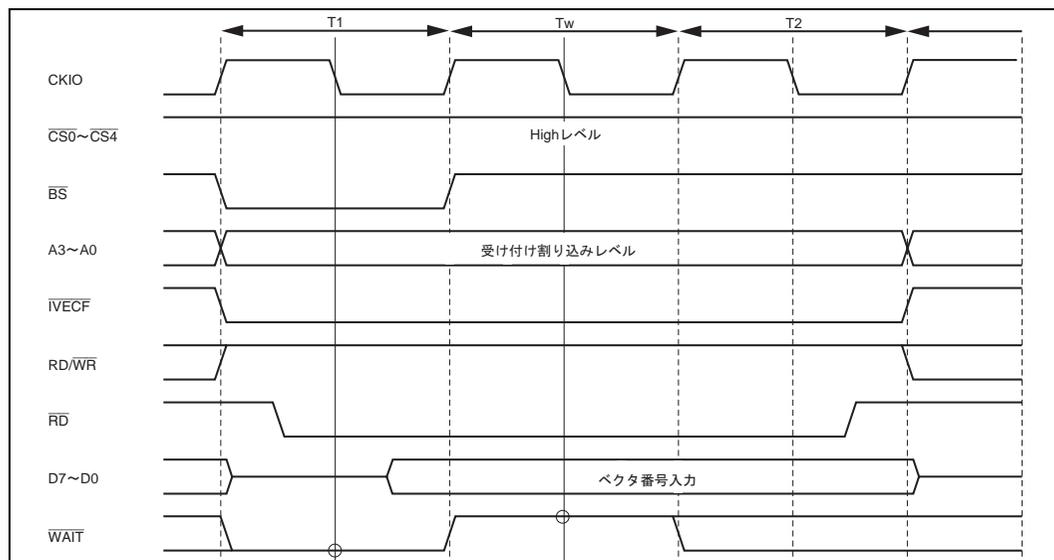


図 5.7 外部ベクタフェッチ (I : E =1 : 1 以外 (WAIT 入力))

### 5.2.6 内蔵周辺モジュール割り込み

内蔵周辺モジュール割り込みは、以下に示す 8 種類の内蔵周辺モジュールで発生する割り込みです。

- ダイレクトメモリアクセスコントローラ (DMAC)
- バスステートコントローラ (BSC)
- ウォッチドッグタイマ (WDT)
- 16 ビットフリーランニングタイマ (FRT)
- イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) (EtherC の割り込みも含む)
- 16 ビットタイマパルスユニット (TPU)
- FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)
- シリアル I/O (SIO)

要因ごとに異なる割り込みベクタが割り当てられるため、例外サービスルーチンで要因を判定する必要はありません。優先順位は、割り込み優先レベル設定レジスタ A、B、D、E (IPRA、B、D、E) によって、モジュールごとに優先レベル 0 ~ 15 の範囲で設定できます。

内蔵周辺モジュール割り込み例外処理では、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) は、受け付けた内蔵周辺モジュール割り込みの優先レベル値に設定されます。

### 5.2.7 割り込み例外処理ベクタと優先順位

表 5.4 に、割り込み要因とベクタ番号、ベクタテーブルアドレスオフセット、割り込み優先順位を示します。

各割り込み要因には、それぞれ異なるベクタ番号とベクタテーブルアドレスオフセットが割り当てられています。ベクタテーブルアドレスは、このベクタ番号やベクタテーブルアドレスオフセットから算出されます。割り込み例外処理では、このベクタテーブルアドレスが示すベクタテーブルから例外サービスルーチンの開始アドレスが取り出されます。ベクタテーブルアドレスの算出法は、「4. 例外処理」の「表 4.4 例外処理ベクタテーブルアドレスの算出法」を参照してください。

IRL 割り込みの優先順位は、IRL15 ~ IRL1 がそれぞれ優先レベル 15 ~ 1 になります。IRQ 割り込みと内蔵周辺モジュール割り込みの優先順位は、割り込み優先レベル設定レジスタ A ~ E (IPRA ~ IPRE) によって、モジュールごとに、優先レベル 0 ~ 15 の範囲で任意に設定できます。ただし、A ~ E (IPRA ~ IPRE) に対応する割り込み要因の優先順位は、表 5.4 の「IPR 設定単位内の優先順位」に示すように定められており、変更できません。内蔵周辺モジュール割り込みの優先順位は、リセットによって、優先レベル 0 に設定されます。複数の割り込み要因の優先順位を同じレベルに設定した場合、それらの割り込みが同時に発生したときは、表 5.4 に示す「デフォルト優先順位」に従って処理されます。



5. 割り込みコントローラ (INTC)

割り込み要因		ベクタ		割り込み優先順位 (初期値)	対応する IPR	IPR設定 単位内の 優先順位	対応する VCR	デフォルト 優先順位
		ベクタ番号	ベクタテーブル アドレス					
FRT	ICI	0~127*2	VBR + (ベクタ番号×4)	15~0 (0)	IPRB (11~8)	高 ↑ ↓ 低	VCRC (14~8)	高 ↑ ↓ 低
	OCI	0~127*2					VCRC (6~0)	
	OVI	0~127*2					VCRD (14~8)	
TPU0	TGI0A	0~127*2		15~0 (0)	IPRD (15~12)	高 ↑ ↓ 低	VCRE (14~8)	
	TGI0B	0~127*2					VCRE (6~0)	
	TGI0C	0~127*2					VCRF (14~8)	
	TGI0D	0~127*2					VCRF (6~0)	
	TCI0V	0~127*2					VCRG (14~8)	
	TCI0V	0~127*2					VCRG (14~8)	
TPU1	TGI1A	0~127*2		15~0 (0)	IPRD (11~8)	高 ↑ ↓ 低	VCRH (14~8)	
	TGI1B	0~127*2					VCRH (6~0)	
	TCI1V	0~127*2					VCRI (14~8)	
	TCI1U	0~127*2					VCRI (6~0)	
TPU2	TGI2A	0~127*2		15~0 (0)	IPRD (7~4)	高 ↑ ↓ 低	VCRJ (14~8)	
	TGI2B	0~127*2					VCRJ (6~0)	
	TCI2V	0~127*2					VCRK (14~8)	
	TCI2U	0~127*2					VCRK (6~0)	
SCIF1	ERI1	0~127*2		15~0 (0)	IPRD (3~0)	高 ↑ ↓ 低	VCRL (14~8)	
	RXI1	0~127*2					VCRL (6~0)	
	BRI1	0~127*2					VCRM (14~8)	
	TXI1	0~127*2					VCRM (6~0)	
SCIF2	ERI2	0~127*2		15~0 (0)	IPRE (15~12)	高 ↑ ↓ 低	VCRN (14~8)	
	RXI2	0~127*2					VCRN (6~0)	
	BRI2	0~127*2					VCRO (14~8)	
	TXI2	0~127*2					VCRO (6~0)	
SIO0	RER10	0~127*2	15~0 (0)	IPRE (11~8)	高 ↑ ↓ 低	VCRP (14~8)		
	TER10	0~127*2				VCRP (6~0)		
	RDF10	0~127*2				VCRQ (14~8)		
	TDE10	0~127*2				VCRQ (6~0)		
SIO1	RER11	0~127*2	15~0 (0)	IPRE (7~4)	高 ↑ ↓ 低	VCRR (14~8)		
	TER11	0~127*2				VCRR (6~0)		
	RDF11	0~127*2				VCRS (14~8)		
	TDE11	0~127*2				VCRS (6~0)		
SIO2	RER12	0~127*2	15~0 (0)	IPRE (3~0)	高 ↑ ↓ 低	VCRT (14~8)		
	TER12	0~127*2				VCRT (6~0)		
	RDF12	0~127*2				VCRU (14~8)		
	TDE12	0~127*2				VCRU (6~0)		
予約		128~255	—	—	—	—	—	低

- 【注】 \*1 この表のオートベクタ番号を使わずに、外部ベクタ番号フェッチも可能です。  
外部ベクタ番号は0 ~ 127 になります。  
\*2 ベクタ番号は内蔵ベクタ番号設定レジスタで設定します。  
\*3 REF はバスステートコントローラ内のリフレッシュ制御部です。  
\*4 ICR の EXIMD ビットで IRL1 ~ 15 か IRQ0 ~ 3 に設定します。

## 5. 割り込みコントローラ (INTC)

---

- \*5 SH7615 では、VCRB を予約レジスタとし、本レジスタへのアクセスを禁止します。
- \*6 E-DMAC 割り込み (EINT) は、EtherC/E-DMAC ステータスレジスタ (EESR) の 19 割り込み要因のうち、EtherC/E-DMAC ステータス割り込み許可レジスタ (EESIPR) で許可をうけた割り込みの OR となります。また、EESR の ECI ビットには、EtherC ステータスレジスタ (ECSR) の中の 3 つのステータスを割り込み要因として反映させることができますので、EINT は、最大で 22 割り込み要因の OR として、INTC に入力されます。



## 5. 割り込みコントローラ (INTC)

割り込み要因		ベクタ		割り込み優先順位 (初期値)	対応する IPR	IPR設定 単位内の 優先順位	対応する VCR	デフォルト 優先順位
		ベクタ番号	ベクタテーブル アドレス					
TPU1	TGI1A	0~127*2	VBR + (ベクタ番号×4)	15~0 (0)	IPRD (11~8)	高 ↑ ↓ 低	VCRH (14~8)	高 ↑ ↓ 低
	TGI1B	0~127*2					VCRH (6~0)	
	TCI1V	0~127*2					VCRI (14~8)	
	TCI1U	0~127*2					VCRI (6~0)	
TPU2	TGI2A	0~127*2		15~0 (0)	IPRD (7~4)	高 ↑ ↓ 低	VCRJ (14~8)	
	TGI2B	0~127*2					VCRJ (6~0)	
	TCI2V	0~127*2					VCRK (14~8)	
	TCI2U	0~127*2					VCRK (6~0)	
SCIF1	ERI1	0~127*2		15~0 (0)	IPRD (3~0)	高 ↑ ↓ 低	VCRL (14~8)	
	RX11	0~127*2					VCRL (6~0)	
	BRI1	0~127*2					VCRM (14~8)	
	TX11	0~127*2					VCRM (6~0)	
SCIF2	ERI2	0~127*2		15~0 (0)	IPRE (15~12)	高 ↑ ↓ 低	VCRN (14~8)	
	RX12	0~127*2					VCRN (6~0)	
	BRI2	0~127*2					VCRO (14~8)	
	TX12	0~127*2					VCRO (6~0)	
SIO0	RERI0	0~127*2	15~0 (0)	IPRE (11~8)	高 ↑ ↓ 低	VCRP (14~8)		
	TERI0	0~127*2				VCRP (6~0)		
	RDFI0	0~127*2				VCRQ (14~8)		
	TDEI0	0~127*2				VCRQ (6~0)		
SIO1	RERI1	0~127*2	15~0 (0)	IPRE (7~4)	高 ↑ ↓ 低	VCRR (14~8)		
	TERI1	0~127*2				VCRR (6~0)		
	RDFI1	0~127*2				VCRS (14~8)		
	TDEI1	0~127*2				VCRS (6~0)		
SIO2	RERI2	0~127*2	15~0 (0)	IPRE (3~0)	高 ↑ ↓ 低	VCRT (14~8)		
	TERI2	0~127*2				VCRT (6~0)		
	RDFI2	0~127*2				VCRU (14~8)		
	TDEI2	0~127*2				VCRU (6~0)		
予約		128~255		—	—	—	—	低

- 【注】
- \*1 この表のオートベクタ番号を使わずに、外部ベクタ番号フェッチも可能です。  
外部ベクタ番号は0 ~ 127 になります。
  - \*2 ベクタ番号は内蔵ベクタ番号設定レジスタで設定します。
  - \*3 REF はバーステートコントローラ内のリフレッシュ制御部です。
  - \*4 ICR の EXIMD ビットで IRL1 ~ 15 か IRQ0 ~ 3 に設定します。
  - \*5 SH7615 では、VCRB を予約レジスタとし、本レジスタへのアクセスを禁止します。
  - \*6 E-DMAC 割り込み (EINT) は、EtherC/E-DMAC ステータスレジスタ (EESR) の 19 割り込み要因のうち、EtherC/E-DMAC ステータス割り込み許可レジスタ (EESIPR) で許可をうけた割り込みの OR となります。また、EESR の ECI ビットには、EtherC ステータスレジスタ (ECSR) 中の 3 つのステータスを割り込み要因として反映させることができますので、EINT は、最大で 22 割り込み要因の OR として、INTC に入力されます。

## 5.3 レジスタの説明

### 5.3.1 割り込み優先レベル設定レジスタ A (IPRA)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DMAC IP3	DMAC IP2	DMAC IP1	DMAC IP0	WDT IP3	WDT IP2	WDT IP1	WDT IP0	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

割り込み優先レベル設定レジスタ A (IPRA) は、読み出し / 書き込み可能な 16 ビットのレジスタで、内蔵周辺モジュール割り込みの優先順位 (レベル 0 ~ 15) を設定します。

IPRA は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

(リセットはパワーオン、マニュアルリセット両方を意味します。以下も同様です。)

- ビット 15 ~ 12 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 11 ~ 8 : ダイレクトメモリアクセスコントローラ (DMAC) 割り込み優先レベル 3 ~ 0 (DMACIP3 ~ DMACIP0)  
ダイレクトメモリアクセスコントローラ (DMAC) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。DMAC は、2 チャンネルありますが、同じレベルを設定します。同時に割り込みが発生したときは、チャンネル 0 が優先されます。
- ビット 7 ~ 4 : ウォッチドッグタイマ (WDT) 割り込み優先レベル 3 ~ 0 (WDTIP3 ~ WDTIP0)  
ウォッチドッグタイマ (WDT) 割り込み優先レベルとバスステートコントローラ (BSC) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。同時に WDT 割り込みと BSC 割り込みが発生したときは、WDT 割り込みが優先されます。
- ビット 3 ~ 0 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 5. 割り込みコントローラ (INTC)

### 5.3.2 割り込み優先レベル設定レジスタ B (IPRB)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	E-DMAC IP3	E-DMAC IP2	E-DMAC IP1	E-DMAC IP0	FRT IP3	FRT IP2	FRT IP1	FRT IP0	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

割り込み優先レベル設定レジスタ B (IPRB) は、読み出し / 書き込み可能な 16 ビットのレジスタで、内蔵周辺モジュール割り込みの優先順位 (レベル 0 ~ 15) を設定します。

IPRB は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

- ・ ビット 15 ~ 12: イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) 割り込み優先レベル 3 ~ 0 (E-DMACIP3 ~ E-DMACIP0)  
イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。
- ・ ビット 11 ~ 8: 16 ビットフリーランニングタイマ (FRT) 割り込み優先レベル 3 ~ 0 (FRTIP3 ~ FRTIP0)  
16 ビットフリーランニングタイマ (FRT) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。
- ・ ビット 7 ~ 0: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### 5.3.3 割り込み優先レベル設定レジスタ C (IPRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ0 IP3	IRQ0 IP2	IRQ0 IP1	IRQ0 IP0	IRQ1 IP3	IRQ1 IP2	IRQ1 IP1	IRQ1 IP0	IRQ2 IP3	IRQ2 IP2	IRQ2 IP1	IRQ2 IP0	IRQ3 IP3	IRQ3 IP2	IRQ3 IP1	IRQ3 IP0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

割り込み優先レベル設定レジスタ C (IPRC) は、読み出し / 書き込み可能な 16 ビットのレジスタで、IRQ0 ~ 3 割り込みの優先順位 (レベル 0 ~ 15) を設定します。IPRC は、リセットで H'0000 に初期化されますが、スタンバイモード時には初期化されません。

- ・ ビット 15 ~ 0: IRQ0 ~ 3 優先レベル 3 ~ 0 (IRQnIP3 ~ IRQnIP0, n=0 ~ 3)  
IRQ0 ~ 3 優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。

## 5.3.4 割り込み優先レベル設定レジスタ D (IPRD)

ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

TPU0 IP3	TPU0 IP2	TPU0 IP1	TPU0 IP0	TPU1 IP3	TPU1 IP2	TPU1 IP1	TPU1 IP0	TPU2 IP3	TPU2 IP2	TPU2 IP1	TPU2 IP0	SCF1 IP3	SCF1 IP2	SCF1 IP1	SCF1 IP0
初期値: 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W: R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

割り込み優先レベル設定レジスタ D (IPRD) は、読み出し / 書き込み可能な 16 ビットのレジスタで、内蔵周辺モジュール割り込みの優先順位 (レベル 0 ~ 15) を設定します。

IPRD は、リセットで H'0000 に初期化されますが、スタンバイモード時には初期化されません。

- ・ ビット 15 ~ 4 : 16 ビットタイムパルスユニット 0 ~ 2 (TPU0 ~ 2) 割り込み優先レベル 3 ~ 0 (TPU1IP3 ~ TPU1IP0, n=0 ~ 2)  
16 ビットタイムパルスユニット 0 ~ 2 (TPU0 ~ 2) 割り込み優先レベルを設定します。  
4 ビットですので 0 から 15 までの値を設定することができます。
- ・ ビット 3 ~ 0 : FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) 割り込み優先レベル 3 ~ 0 (SCF1IP3 ~ SCF1IP0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。

## 5.3.5 割り込み優先レベル設定レジスタ E (IPRE)

ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

SCF2 IP3	SCF2 IP2	SCF2 IP1	SCF2 IP0	SIO0 IP3	SIO0 IP2	SIO0 IP1	SIO0 IP0	SIO1 IP3	SIO1 IP2	SIO1 IP1	SIO1 IP0	SIO2 IP3	SIO2 IP2	SIO2 IP1	SIO2 IP0
初期値: 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W: R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

割り込み優先レベル設定レジスタ E (IPRE) は、読み出し / 書き込み可能な 16 ビットのレジスタで、内蔵周辺モジュール割り込みの優先順位 (レベル 0 ~ 15) を設定します。

IPRE は、リセットで H'0000 に初期化されますが、スタンバイモード時には初期化されません。

- ・ ビット 15 ~ 12 : FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) 割り込み優先レベル 3 ~ 0 (SCF2IP3 ~ SCF2IP0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。
- ・ ビット 11 ~ 0 : シリアル I/O0 ~ 2 (SIO0 ~ 2) 割り込み優先レベル 3 ~ 0 (SIO1IP3 ~ SIO1IP0, n=0 ~ 2)

## 5. 割り込みコントローラ (INTC)

シリアル I/O0 ~ 2 (SIO0 ~ 2) 割り込み優先レベルを設定します。4 ビットですので 0 から 15 までの値を設定することができます。

内蔵周辺モジュール割り込みと割り込み優先レベル設定レジスタの関係を表 5.5 に示します。

表 5.5 割り込み要求元と IPR.A ~ IPRE

レジスタ	ビット			
	15 ~ 12	11 ~ 8	7 ~ 4	3 ~ 0
割り込み優先レベル設定レジスタ A	予約	DMAC0、DMAC1	WDT、REF	予約
割り込み優先レベル設定レジスタ B	E-DMAC	FRT	予約	予約
割り込み優先レベル設定レジスタ C	IRQ0	IRQ1	IRQ2	IRQ3
割り込み優先レベル設定レジスタ D	TPU0	TPU1	TPU2	SCIF1
割り込み優先レベル設定レジスタ E	SCIF2	SIO0	SIO1	SIO2

表 5.5 に示すように、1 本のレジスタに 2 ~ 4 つの内蔵周辺モジュールが割り当てられています。各 4 ビットに H'0 (0000) から HF (1111) の範囲の値をセットすることによって、それぞれに対応する割り込みの優先順位が設定されます。割り込み優先順位は、H'0 をセットすると優先レベル 0 (最低) に、HF をセットすると優先レベル 15 (最高) になります。2 つの内蔵周辺モジュールが同じビットに割り当てられている場合 (DMAC0 と DMAC1、WDT と BSC リフレッシュ制御部) その 2 つのモジュールは同じ優先順位に設定されます。

IPRA ~ IPRE は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

### 5.3.6 ベクタ番号設定レジスタ WDT (VCRWDT)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	WIT V6	WIT V5	WIT V4	WIT V3	WIT V2	WIT V1	WIT V0	—	BCM V6	BCM V5	BCM V4	BCM V3	BCM V2	BCM V1	BCM V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ WDT (VCRWDT) は、読み出し / 書き込み可能な 16 ビットのレジスタで、WDT のインターバル割り込みと BSC のコンペアマッチ割り込みベクタ番号 (0 ~ 127) を設定します。

VCRWDT は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: ウォッチドッグタイマ (WDT) インターバル割り込みベクタ番号 6 ~ 0 (WITV6 ~ WITV0)  
ウォッチドッグタイマ (WDT) インターバル割り込み (ITI) のベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

- ビット 6 ~ 0: バスステートコントローラ (BSC) コンペアマッチ割り込みベクタ番号 6 ~ 0 (BCM<sub>V6</sub> ~ BCM<sub>V0</sub>)  
バスステートコントローラ (BSC) コンペアマッチ割り込み (CMI) のベクタ番号を設定します。7 ビットです  
ので 0 から 127 までの値を設定することができます。

### 5.3.7 ベクタ番号設定レジスタ A (VCRA)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	EIN V6	EIN V5	EIN V4	EIN V3	EIN V2	EIN V1	EIN V0	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R						

ベクタ番号設定レジスタ A (VCRA) は、読み出し / 書き込み可能な 16 ビットのレジスタで、E-DMAC の割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRA は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

- ビット 15、7 ~ 0: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 14 ~ 8: イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) 割り込みベクタ番号 6 ~ 0 (EIN<sub>V6</sub> ~ EIN<sub>V0</sub>)  
イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) 割り込み (EINT) のベクタ番号を設定します。7 ビットです  
ので 0 から 127 までの値を設定することができます。

### 5.3.8 ベクタ番号設定レジスタ B (VCRB) (予約レジスタ)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

ベクタ番号設定レジスタ B (VCRB) は、16 ビットの予約レジスタです。本レジスタへのアクセスを禁止します。

VCRB は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

## 5. 割り込みコントローラ (INTC)

### 5.3.9 ベクタ番号設定レジスタ C (VCRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	FIC V6	FIC V5	FIC V4	FIC V3	FIC V2	FIC V1	FIC V0	—	FOC V6	FOC V5	FOC V4	FOC V3	FOC V2	FOC V1	FOC V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ C (VCRC) は、読み出し/書き込み可能な 16 ビットのレジスタで、16 ビットフリーランニングタイム (FRT) のインプットキャプチャ割り込みとアウトプットコンペア割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRC は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: 16 ビットフリーランニングタイム (FRT) インプットキャプチャ割り込みベクタ番号 6 ~ 0 (FICV6 ~ FICV0)  
16 ビットフリーランニングタイム (FRT) インプットキャプチャ割り込み (ICI) のベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: 16 ビットフリーランニングタイム (FRT) アウトプットコンペア割り込みベクタ番号 6 ~ 0 (FOCV6 ~ FOCV0)  
16 ビットフリーランニングタイム (FRT) アウトプットコンペア割り込み (OCI) のベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.10 ベクタ番号設定レジスタ D (VCRD)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	FOV V6	FOV V5	FOV V4	FOV V3	FOV V2	FOV V1	FOV V0	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R						

ベクタ番号設定レジスタ D (VCRD) は、読み出し/書き込み可能な 16 ビットのレジスタで、16 ビットフリーランニングタイム (FRT) のオーバフロー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRD は、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

- ・ ビット 15、7 ~ 0: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 14 ~ 8 : 16 ビットフリーランニングタイム (FRT) オーバフロー割り込みベクタ番号 6 ~ 0 (FOVV6 ~ FOVV0)

16 ビットフリーランニングタイム (FRT) オーバフロー割り込み (OVI) のベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.11 ベクタ番号設定レジスタ E (VCRE)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TG0A V6	TG0A V5	TG0A V4	TG0A V3	TG0A V2	TG0A V1	TG0A V0	—	TG0B V6	TG0B V5	TG0B V4	TG0B V3	TG0B V2	TG0B V1	TG0B V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

ベクタ番号設定レジスタ E (VCRE) は、読み出し / 書き込み可能な 16 ビットのレジスタで、16 ビットタイムパルスユニット 0 (TPU0) の TGR0A と TGR0B のインプットキャプチャ / コンペアマッチ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRE は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 14 ~ 8 : 16 ビットタイムパルスユニット 0 (TPU0) TGR0A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG0AV6 ~ TG0AV0)  
16 ビットタイムパルスユニット 0 (TPU0) TGR0A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ビット 6 ~ 0 : 16 ビットタイムパルスユニット 0 (TPU0) TGR0B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG0BV6 ~ TG0BV0)  
16 ビットタイムパルスユニット 0 (TPU0) TGR0B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.12 ベクタ番号設定レジスタ F (VCRF)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TG0C V6	TG0C V5	TG0C V4	TG0C V3	TG0C V2	TG0C V1	TG0C V0	—	TG0D V6	TG0D V5	TG0D V4	TG0D V3	TG0D V2	TG0D V1	TG0D V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

## 5. 割り込みコントローラ (INTC)

ベクタ番号設定レジスタ F (VCRF) は、読み出し/書き込み可能な 16 ビットのレジスタで、16 ビットタイムパルスユニット 0 (TPU0) の TGR0C と TGR0D のインプットキャプチャ/コンペアマッチ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRF は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: 16 ビットタイムパルスユニット 0 (TPU0) TGR0C のインプットキャプチャ/コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG0CV6 ~ TG0CV0)  
16 ビットタイムパルスユニット 0 (TPU0) TGR0C のインプットキャプチャ/コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: 16 ビットタイムパルスユニット 0 (TPU0) TGR0D のインプットキャプチャ/コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG0DV6 ~ TG0DV0)  
16 ビットタイムパルスユニット 0 (TPU0) TGR0D のインプットキャプチャ/コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.13 ベクタ番号設定レジスタ G (VCRG)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TC0V V6	TC0V V5	TC0V V4	TC0V V3	TC0V V2	TC0V V1	TC0V V0	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R						

ベクタ番号設定レジスタ G (VCRG) は、読み出し/書き込み可能な 16 ビットのレジスタで、16 ビットタイムパルスユニット 0 (TPU0) の TCNT0 のオーバーフロー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRG は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 ~ 0: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: 16 ビットタイムパルスユニット 0 (TPU0) TCNT0 のオーバーフロー割り込みベクタ番号 6 ~ 0 (TC0VV6 ~ TC0VV0)  
16 ビットタイムパルスユニット 0 (TPU0) TCNT0 のオーバーフロー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.14 ベクタ番号設定レジスタ H (VCRH)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TG1A V6	TG1A V5	TG1A V4	TG1A V3	TG1A V2	TG1A V1	TG1A V0	—	TG1B V6	TG1B V5	TG1B V4	TG1B V3	TG1B V2	TG1B V1	TG1B V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ H (VCRH) は、読み出し / 書き込み可能な 16 ビットのレジスタで、16 ビットタイマパルスユニット 1 (TPU1) の TGR1A と TGR1B のインプットキャプチャ / コンペアマッチ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRH は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: 16 ビットタイマパルスユニット 1 (TPU1) TGR1A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG1AV6 ~ TG1AV0)  
16 ビットタイマパルスユニット 1 (TPU1) TGR1A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: 16 ビットタイマパルスユニット 1 (TPU1) TGR1B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG1BV6 ~ TG1BV0)  
16 ビットタイマパルスユニット 1 (TPU1) TGR1B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.15 ベクタ番号設定レジスタ I (VCRI)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TC1V V6	TC1V V5	TC1V V4	TC1V V3	TC1V V2	TC1V V1	TC1V V0	—	TC1U V6	TC1U V5	TC1U V4	TC1U V3	TC1U V2	TC1U V1	TC1U V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ I (VCRI) は、読み出し / 書き込み可能な 16 ビットのレジスタで、16 ビットタイマパルスユニット 1 (TPU1) の TCNT1 のオーバフロー / アンダフロー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRI は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 5. 割り込みコントローラ (INTC)

- ・ ビット 14 ~ 8 : 16 ビットタイマパルスユニット 1 (TPU1) TCNT1 のオーバーフロー割り込みベクタ番号 6 ~ 0 (TC1VV6 ~ TC1VV0)  
16 ビットタイマパルスユニット 1 (TPU1) TCNT1 のオーバーフロー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : 16 ビットタイマパルスユニット 1 (TPU1) TCNT1 のアンダフロー割り込みベクタ番号 6 ~ 0 (TC1UV6 ~ TC1UV0)  
16 ビットタイマパルスユニット 1 (TPU1) TCNT1 のアンダフロー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.16 ベクタ番号設定レジスタ J (VCRJ)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TG2A V6	TG2A V5	TG2A V4	TG2A V3	TG2A V2	TG2A V1	TG2A V0	—	TG2B V6	TG2B V5	TG2B V4	TG2B V3	TG2B V2	TG2B V1	TG2B V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

ベクタ番号設定レジスタ J (VCRJ) は、読み出し / 書き込み可能な 16 ビットのレジスタで、16 ビットタイマパルスユニット 2 (TPU2) の TGR2A と TGR2B のインプットキャプチャ / コンペアマッチ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRJ は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8 : 16 ビットタイマパルスユニット 2 (TPU2) TGR2A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG2AV6 ~ TG2AV0)  
16 ビットタイマパルスユニット 2 (TPU2) TGR2A のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : 16 ビットタイマパルスユニット 2 (TPU2) TGR2B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号 6 ~ 0 (TG2BV6 ~ TG2BV0)  
16 ビットタイマパルスユニット 2 (TPU2) TGR2B のインプットキャプチャ / コンペアマッチ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.17 ベクタ番号設定レジスタ K (VCRK)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TC2V V6	TC2V V5	TC2V V4	TC2V V3	TC2V V2	TC2V V1	TC2V V0	—	TC2U V6	TC2U V5	TC2U V4	TC2U V3	TC2U V2	TC2U V1	TC2U V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ K (VCRK) は、読み出し / 書き込み可能な 16 ビットのレジスタで、16 ビットタイムパルスユニット 2 (TPU2) の TCNT2 のオーバフロー / アンダフロー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRK は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: 16 ビットタイムパルスユニット 2 (TPU2) TCNT2 のオーバフロー割り込みベクタ番号 6 ~ 0 (TC2VV6 ~ TC2VV0)  
16 ビットタイムパルスユニット 2 (TPU2) TCNT2 のオーバフロー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: 16 ビットタイムパルスユニット 2 (TPU2) TCNT2 のアンダフロー割り込みベクタ番号 6 ~ 0 (TC2UV6 ~ TC2UV0)  
16 ビットタイムパルスユニット 2 (TPU2) TCNT2 のアンダフロー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.18 ベクタ番号設定レジスタ L (VCRL)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SER1 V6	SER1 V5	SER1 V4	SER1 V3	SER1 V2	SER1 V1	SER1 V0	—	SRX1 V6	SRX1 V5	SRX1 V4	SRX1 V3	SRX1 V2	SRX1 V1	SRX1 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ L (VCRL) は、読み出し / 書き込み可能な 16 ビットのレジスタで、FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の受信エラー割り込みと受信データフル / データレディ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRL は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 5. 割り込みコントローラ (INTC)

---

- ・ ビット 14 ~ 8 : FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の受信エラー割り込みベクタ番号 6 ~ 0 (SER1V6 ~ SER1V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の受信エラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の受信データフル / データレディ割り込みベクタ番号 6 ~ 0 (SRX1V6 ~ SRX1V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の受信データフル / データレディ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.19 ベクタ番号設定レジスタ M (VCRM)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SBR1 V6	SBR1 V5	SBR1 V4	SBR1 V3	SBR1 V2	SBR1 V1	SBR1 V0	—	STX1 V6	STX1 V5	STX1 V4	STX1 V3	STX1 V2	STX1 V1	STX1 V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

ベクタ番号設定レジスタ M (VCRM) は、読み出し / 書き込み可能な 16 ビットのレジスタで、FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) のブ레이크割り込みと送信データエンプティ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRM は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8 : FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) のブ레이크割り込みベクタ番号 6 ~ 0 (SBR1V6 ~ SBR1V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) のブ레이크割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の送信データエンプティ割り込みベクタ番号 6 ~ 0 (STX1V6 ~ STX1V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 1 (SCIF1) の送信データエンプティ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.20 ベクタ番号設定レジスタ N (VCRN)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SER2 V6	SER2 V5	SER2 V4	SER2 V3	SER2 V2	SER2 V1	SER2 V0	—	SRX2 V6	SRX2 V5	SRX2 V4	SRX2 V3	SRX2 V2	SRX2 V1	SRX2 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ N (VCRN) は、読み出し / 書き込み可能な 16 ビットのレジスタで、FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の受信エラー割り込みと受信データフル / データレディ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRN は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の受信エラー割り込みベクタ番号 6 ~ 0 (SER2V6 ~ SER2V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の受信エラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の受信データフル / データレディ割り込みベクタ番号 6 ~ 0 (SRX2V6 ~ SRX2V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の受信データフル / データレディ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.21 ベクタ番号設定レジスタ O (VCRO)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SBR2 V6	SBR2 V5	SBR2 V4	SBR2 V3	SBR2 V2	SBR2 V1	SBR2 V0	—	STX2 V6	STX2 V5	STX2 V4	STX2 V3	STX2 V2	STX2 V1	STX2 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ O (VCRO) は、読み出し / 書き込み可能な 16 ビットのレジスタで、FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) のブ레이크割り込みと送信データエンプティ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRO は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 5. 割り込みコントローラ (INTC)

---

- ・ ビット 14 ~ 8 : FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) のブレイク割り込みベクタ番号 6 ~ 0 (SBR2V6 ~ SBR2V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) のブレイク割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の送信データエンプティ割り込みベクタ番号 6 ~ 0 (STX2V6 ~ STX2V0)  
FIFO 内蔵シリアルコミュニケーションインタフェース 2 (SCIF2) の送信データエンプティ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.22 ベクタ番号設定レジスタ P (VCRP)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RER0 V6	RER0 V5	RER0 V4	RER0 V3	RER0 V2	RER0 V1	RER0 V0	—	TER0 V6	TER0 V5	TER0 V4	TER0 V3	TER0 V2	TER0 V1	TER0 V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

ベクタ番号設定レジスタ P (VCRP) は、読み出し / 書き込み可能な 16 ビットのレジスタで、シリアル I/O0 (SIO0) の受信オーバーランエラー割り込みと送信アンダランエラー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRP は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8 : シリアル I/O 0 (SIO0) の受信オーバーランエラー割り込みベクタ番号 6 ~ 0 (RER0V6 ~ RER0V0)  
シリアル I/O0 (SIO0) の受信オーバーランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : シリアル I/O 0 (SIO0) の送信アンダランエラー割り込みベクタ番号 6 ~ 0 (TER0V6 ~ TER0V0)  
シリアル I/O 0 (SIO0) の送信アンダランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.23 ベクタ番号設定レジスタ Q (VCRQ)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RDF0 V6	RDF0 V5	RDF0 V4	RDF0 V3	RDF0 V2	RDF0 V1	RDF0 V0	—	TDE0 V6	TDE0 V5	TDE0 V4	TDE0 V3	TDE0 V2	TDE0 V1	TDE0 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ Q (VCRQ) は、読み出し / 書き込み可能な 16 ビットのレジスタで、シリアル I/O0 (SIO0) の受信データフル割り込みと送信データエンプティ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRQ は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: シリアル I/O0 (SIO0) の受信データフル割り込みベクタ番号 6 ~ 0 (RDF0V6 ~ RDF0V0)  
シリアル I/O0 (SIO0) の受信データフル割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: シリアル I/O0 (SIO0) の送信データエンプティ割り込みベクタ番号 6 ~ 0 (TDE0V6 ~ TDE0V0)  
シリアル I/O0 (SIO0) の送信データエンプティ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5.3.24 ベクタ番号設定レジスタ R (VCRR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RER1 V6	RER1 V5	RER1 V4	RER1 V3	RER1 V2	RER1 V1	RER1 V0	—	TER1 V6	TER1 V5	TER1 V4	TER1 V3	TER1 V2	TER1 V1	TER1 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ R (VCRR) は、読み出し / 書き込み可能な 16 ビットのレジスタで、シリアル I/O1 (SIO1) の受信オーバーランエラー割り込みと送信アンダランエラー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRR は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 5. 割り込みコントローラ (INTC)

- ・ ビット 14 ~ 8 : シリアル I/O1 (SIO1) の受信オーバーランエラー割り込みベクタ番号 6 ~ 0 (RER1V6 ~ RER1V0)  
シリアル I/O1 (SIO1) の受信オーバーランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : シリアル I/O1 (SIO1) の送信アンダランエラー割り込みベクタ番号 6 ~ 0 (TER1V6 ~ TER1V0)  
シリアル I/O1 (SIO1) の送信アンダランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.25 ベクタ番号設定レジスタ S (VCRS)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RDF1 V6	RDF1 V5	RDF1 V4	RDF1 V3	RDF1 V2	RDF1 V1	RDF1 V0	—	TDE1 V6	TDE1 V5	TDE1 V4	TDE1 V3	TDE1 V2	TDE1 V1	TDE1 V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

ベクタ番号設定レジスタ S (VCRS) は、読み出し / 書き込み可能な 16 ビットのレジスタで、シリアル I/O1 (SIO1) の受信データフル割り込みと送信データエンプティ割り込みのベクタ番号 (0 ~ 127) を設定します。VCRS は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8 : シリアル I/O1 (SIO1) の受信データフル割り込みベクタ番号 6 ~ 0 (RDF1V6 ~ RDF1V0)  
シリアル I/O1 (SIO1) の受信データフル割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : シリアル I/O1 (SIO1) の送信データエンプティ割り込みベクタ番号 6 ~ 0 (TDE1V6 ~ TDE1V0)  
シリアル I/O1 (SIO1) の送信データエンプティ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

### 5.3.26 ベクタ番号設定レジスタ T (VCRT)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RER2 V6	RER2 V5	RER2 V4	RER2 V3	RER2 V2	RER2 V1	RER2 V0	—	TER2 V6	TER2 V5	TER2 V4	TER2 V3	TER2 V2	TER2 V1	TER2 V0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W												

---

ベクタ番号設定レジスタ T (VCRT) は、読み出し/書き込み可能な 16 ビットのレジスタで、シリアル I/O2 (SIO2) の受信オーバーランエラー割り込みと送信アンダランエラー割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRT は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8 : シリアル I/O2 (SIO2) の受信オーバーランエラー割り込みベクタ番号 6 ~ 0 (RER2V6 ~ RER2V0)  
シリアル I/O2 (SIO2) の受信オーバーランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0 : シリアル I/O2 (SIO2) の送信アンダランエラー割り込みベクタ番号 6 ~ 0 (TER2V6 ~ TER2V0)  
シリアル I/O2 (SIO2) の送信アンダランエラー割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

## 5. 割り込みコントローラ (INTC)

---

### 5.3.27 ベクタ番号設定レジスタ U (VCRU)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RDF2 V6	RDF2 V5	RDF2 V4	RDF2 V3	RDF2 V2	RDF2 V1	RDF2 V0	—	TDE2 V6	TDE2 V5	TDE2 V4	TDE2 V3	TDE2 V2	TDE2 V1	TDE2 V0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W												

ベクタ番号設定レジスタ U (VCRU) は、読み出し / 書き込み可能な 16 ビットのレジスタで、シリアル I/O2 (SIO2) の受信データフル割り込みと送信データエンプティ割り込みのベクタ番号 (0 ~ 127) を設定します。

VCRU は、リセットで H'0000 に初期化されます。スタンバイモード時には初期化されません。

- ・ ビット 15、7: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 14 ~ 8: シリアル I/O2 (SIO2) の受信データフル割り込みベクタ番号 6 ~ 0 (RDF2V6 ~ RDF2V0)  
シリアル I/O2 (SIO2) の受信データフル割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。
- ・ ビット 6 ~ 0: シリアル I/O2 (SIO2) の送信データエンプティ割り込みベクタ番号 6 ~ 0 (TDE2V6 ~ TDE2V0)  
シリアル I/O2 (SIO2) の送信データエンプティ割り込みベクタ番号を設定します。7 ビットですので 0 から 127 までの値を設定することができます。

内蔵周辺モジュール割り込みと割り込みベクタ番号設定レジスタの関係を表 5.6、表 5.7 に示します。

表 5.6 割り込み要求元とベクタ番号設定レジスタ (1)

レジスタ	ビット	
	14 ~ 8	6 ~ 0
ベクタ番号設定レジスタ WDT	インターバル割り込み (WDT)	コンペアマッチ割り込み (BSC)
ベクタ番号設定レジスタ A	E-DMAC 割り込み (E-DMAC)	予約
ベクタ番号設定レジスタ B	予約	予約
ベクタ番号設定レジスタ C	インプットキャプチャ割り込み (FRT)	アウトプットコンペアマッチ割り込み (FRT)
ベクタ番号設定レジスタ D	オーバフロー割り込み (FRT)	予約
ベクタ番号設定レジスタ E	インプットキャプチャ / コンペアマッチ割り込み (TPU0 / TGR0A)	インプットキャプチャ / コンペアマッチ割り込み (TPU0 / TGR0B)
ベクタ番号設定レジスタ F	インプットキャプチャ / コンペアマッチ割り込み (TPU0 / TGR0C)	インプットキャプチャ / コンペアマッチ割り込み (TPU0 / TGR0D)
ベクタ番号設定レジスタ G	オーバフロー割り込み (TPU0 / TCNT0)	予約
ベクタ番号設定レジスタ H	インプットキャプチャ / コンペアマッチ割り込み (TPU1 / TGR1A)	インプットキャプチャ / コンペアマッチ割り込み (TPU1 / TGR1B)
ベクタ番号設定レジスタ I	オーバフロー割り込み (TPU1 / TCNT1)	アンダーフロー割り込み (TPU1 / TCNT1)
ベクタ番号設定レジスタ J	インプットキャプチャ / コンペアマッチ割り込み (TPU2 / TGR2A)	インプットキャプチャ / コンペアマッチ割り込み (TPU2 / TGR2B)
ベクタ番号設定レジスタ K	オーバフロー割り込み (TPU2 / TCNT2)	アンダーフロー割り込み (TPU2 / TCNT2)
ベクタ番号設定レジスタ L	受信エラー割り込み (SCIF1)	受信データフル / データレディ割り込み (SCIF1)
ベクタ番号設定レジスタ M	ブレイク割り込み (SCIF1)	送信データエンプティ割り込み (SCIF1)
ベクタ番号設定レジスタ N	受信エラー割り込み (SCIF2)	受信データフル / データレディ割り込み (SCIF2)
ベクタ番号設定レジスタ O	ブレイク割り込み (SCIF2)	送信データエンプティ割り込み (SCIF2)
ベクタ番号設定レジスタ P	受信オーバランエラー割り込み (SIO0)	送信アンダランエラー割り込み (SIO0)
ベクタ番号設定レジスタ Q	受信データフル割り込み (SIO0)	送信データエンプティ割り込み (SIO0)
ベクタ番号設定レジスタ R	受信オーバランエラー割り込み (SIO1)	送信アンダランエラー割り込み (SIO1)
ベクタ番号設定レジスタ S	受信データフル割り込み (SIO1)	送信データエンプティ割り込み (SIO1)
ベクタ番号設定レジスタ T	受信オーバランエラー割り込み (SIO2)	送信アンダランエラー割り込み (SIO2)
ベクタ番号設定レジスタ U	受信データフル割り込み (SIO2)	送信データエンプティ割り込み (SIO2)

表 5.6 に示すように、1 本のレジスタに 2 つの内蔵周辺モジュールの割り込みが割り当てられています。ビット 14 ~ 8、ビット 6 ~ 0 の各 7 ビットに H'00 (0000000) から H'7F (1111111) の範囲の値をセットすることによって、それぞれに対応する割り込みのベクタ番号が設定されます。ベクタ番号は、H'00 をセットするとベクタ番号 0 (最小) に、H'7F をセットするとベクタ番号 127 (最大) になります。

ベクタテーブルアドレスは次の式で求められます。

$$\text{ベクタテーブルアドレス} = \text{VBR} + (\text{ベクタ番号} \times 4)$$

ベクタ番号設定レジスタは、リセットで H'0000 に初期化されます。スタンバイモード時には、初期化されません。

## 5. 割り込みコントローラ (INTC)

表 5.7 割り込み要求元とベクタ番号設定レジスタ (2)

レジスタ	割り込み要求先
ベクタ番号設定レジスタ DMA0 (VCRDMA0)	チャンネル 0 転送終了割り込み (DMAC0)
ベクタ番号設定レジスタ DMA1 (VCRDMA1)	チャンネル 1 転送終了割り込み (DMAC1)

表 5.7 に示すように、ダイレクトメモリアクセスコントローラ (DMAC) の転送終了割り込みのベクタ番号は VCRDMA0、VCRDMA1 に設定します。詳しくは、「11. ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。

### 5.3.28 割り込みコントロールレジスタ (ICR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	—	—	—	—	—	—	NMIE	—	—	—	—	—	—	EXIMD	VECM D
初期値:	0/1*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

【注】 \* NMI 端子入力が高レベルのとき 1、NMI 端子入力がローレベルのとき 0

割り込みコントロールレジスタ (ICR) は、16 ビットのレジスタで、外部割り込み入力端子 NMI の入力信号検出モードを設定し、NMI 端子への入力信号レベルを示します。また、外部割り込みベクタモードセレクトにより、IRQ または IRL モードを指定できます。IRQ/IRL 割り込みベクタ番号は、割り込みベクタモードセレクトによってオートベクタモードと外部ベクタモードのどちらに従って設定するかを選択できます。ICR は、リセットで H'8000 または H'0000 に初期化されます。スタンバイモード時には初期化されません。

#### ・ ビット 15 : NMI 入力レベル (NMIL)

NMI 端子に入力されている信号のレベルが設定されます。このビットを読むことによって、NMI 端子のレベルを知ることができます。書き込みは無効です。

ビット 15	説明
NMIL	
0	NMI 端子にローレベルが入力されている
1	NMI 端子にハイレベルが入力されている

#### ・ ビット 14 ~ 9 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

#### ・ ビット 8 : NMI エッジセレクト (NMIE)

NMI 端子への割り込み要求信号を、立ち下がりエッジで検出するか、立ち上がりエッジで検出するかを選択します。

ビット 8	説明
NMIL	
0	NMI 入力の立ち下がりエッジで割り込み要求を検出 (初期値)
1	NMI 入力の立ち上がりエッジで割り込み要求を検出

・ ビット 7 ~ 2 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 1 : 外部割り込みベクタモードセレクト (EXIMD)

このビットは IRQ モードまたは IRL モードを選択します。IRQ モードでは、 $\overline{IRL3}$  ~  $\overline{IRL0}$  信号は、各々 1 つの割り込み要因となります。IRL モードでは、これらの信号は割り込み優先順位 1 ~ 15 を指定します。

ビット 8	説明
NMIL	
0	IRL モード (初期値)
1	IRQ モード

・ ビット 0 : 割り込みベクタモードセレクト (VECMD)

IRL/IRQ 割り込みのベクタ番号設定をオートベクタモードにするか、外部ベクタモードにするかを選択します。オートベクタモードにすると、内部で決められたベクタ番号に設定されます。IRL15 と IRL14 割り込みのベクタ番号は 71 に、以下、IRL1 は 64 に設定されます。外部ベクタモードにすると、ベクタ番号は外部ベクタ番号入力端子 (D7 ~ D0) から 0 ~ 127 の値を入力することができます。

ビット 0	説明
VECMD	
0	オートベクタモード、ベクタ番号を内部で自動的に設定 (初期値)
1	外部ベクタモード、ベクタ番号を外部から入力

### 5.3.29 IRQ コントロール / ステータスレジスタ (IRQCSR)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ 31S	IRQ 30S	IRQ 21S	IRQ 20S	IRQ 11S	IRQ 10S	IRQ 01S	IRQ 00S	IRL 3PS	IRL 2PS	IRL 1PS	IRL 0PS	IRQ 3F	IRQ 2F	IRQ 1F	IRQ 0F
初期値 :	0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0	0	0	0
R/W :	R/W	R	R	R	R	R/(W)*R/(W)*R/(W)*R/(W)*										

【注】\* フラグクリアのための 0 書き込みのみ可能です (エッジ検出時)。

IRQ コントロール / ステータスレジスタ (IRQCSR) は 16 ビットのレジスタで、 $\overline{IRL3}$  ~  $\overline{IRL0}$  の入力信号検出モードを設定し、 $\overline{IRL3}$  ~  $\overline{IRL0}$  端子への入力信号レベルを示します。また、IRQ 割り込みのステータスを示します。IRQCSR は、リセットで初期化されます。スタンバイモード時には初期化されません。

## 5. 割り込みコントローラ (INTC)

- ビット 15 ~ 8 : IRQ センスセレクトビット (IRQ31S ~ IRQ00S)  
 $\overline{IRLn}$  ~  $\overline{IRL0}$  の IRQ 検出モードを設定します。

ビット 15 ~ 8		説 明
IRQn1S	IRQn0S	
0	0	ローレベル検出 (初期値)
0	1	立ち下がりエッジ検出
1	0	立ち上がりエッジ検出
1	1	両エッジ検出

【注】 n = 0 ~ 3

- ビット 7 ~ 4 : IRL 端子ステータスビット (IRL3PS ~ IRL0PS)  
 $\overline{IRLn}$  ~  $\overline{IRL0}$  の端子状態を示します。このビットを読むことで、IRLn ~ IRL0 端子のレベルを知ることができます。書き込みは無効です。

ビット 7 ~ 4		説 明
IRLnPS		
0		$\overline{IRLn}$ 端子にローレベルが入力されている
1		$\overline{IRLn}$ 端子にハイレベルが入力されている

【注】 n = 0 ~ 3

- ビット 3 ~ 0 : IRQ3 ~ IRQ0 フラグ (IRQ3F ~ IRQ0F)  
IRQ3 ~ IRQ0 の割り込み要求のステータスを表示します。

ビット 3 ~ 0		検出設定	説 明
IRQ3F ~ IRQ0F			
0	レベル検出時	IRQn 割り込み要求が存在しません。 [ クリア条件 ] $\overline{IRLn}$ 入力がハイレベルのとき	(初期値)
	エッジ検出時	IRQn 割り込み要求が検出されていません。 [ クリア条件 ] (1) IRQnF=1 の状態を読み出した後に 0 を書き込んだとき (2) IRQn 割り込みを受け付けたとき	(初期値)
1	レベル検出時	IRQn 割り込み要求が存在します。 [ セット条件 ] $\overline{IRLn}$ 入力がローレベルのとき	
	エッジ検出時	IRQn 割り込み要求が検出されています。 [ セット条件 ] $\overline{IRLn}$ 入力のエッジが検出されたとき	

【注】 n = 0 ~ 3

## 5.4 動作説明

### 5.4.1 割り込み動作の流れ

割り込み発生時の動作の流れを以下に説明します。また、図 5.8 に動作フローを示します。

1. 割り込みコントローラに対して、各割り込み要求元から割り込み要求信号が送られます。
2. 割り込みコントローラでは、送られた割り込み要求の中から、割り込み優先レベル設定レジスタ A ~ E (IPRA ~ IPRE) に従って、最も優先順位の高い割り込みが選択され、それより優先順位の低い割り込みは保留されます。このとき、同一優先順位に設定された割り込み、または同一モジュール内の割り込みが複数発生した場合は、表 5.4 に示す「デフォルト優先順位」と「IPR 設定単位内の優先順位」に従って、最も優先順位の高い割り込みが選択されます。
3. 割り込みコントローラで選択された割り込みの優先レベルと CPU のステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) とが比較されます。I3 ~ I0 ビットに設定されているレベルと同じか低い優先レベルの割り込みは保留されます。I3 ~ I0 ビットのレベルより高い優先レベルの割り込みだけが受け付けられ、CPU へ割り込み要求信号が送られます。
4. 割り込みコントローラから送られた割り込み要求は、CPU が実行しようとしている命令のデコード時に検出され、その命令の実行が割り込み例外処理に置き換えられます。
5. ステータスレジスタ (SR) とプログラムカウンタ (PC) がスタックに退避されます。
6. ステータスレジスタ (SR) の I3 ~ I0 ビットに、受け付けられた割り込みの優先レベルが書き込まれます。
7. IRL/IRQ 割り込みを外部ベクタモードに指定している場合は、外部ベクタ番号入力端子 (D7 ~ D0) からベクタ番号を読み込みます。
8. 受け付けられた割り込みに対応する例外処理ベクタテーブルから、例外サービスルーチン開始アドレスが取り出され、そのアドレスにジャンプして、プログラムの実行が開始されます。このときのジャンプは遅延分岐ではありません。

5. 割り込みコントローラ (INTC)

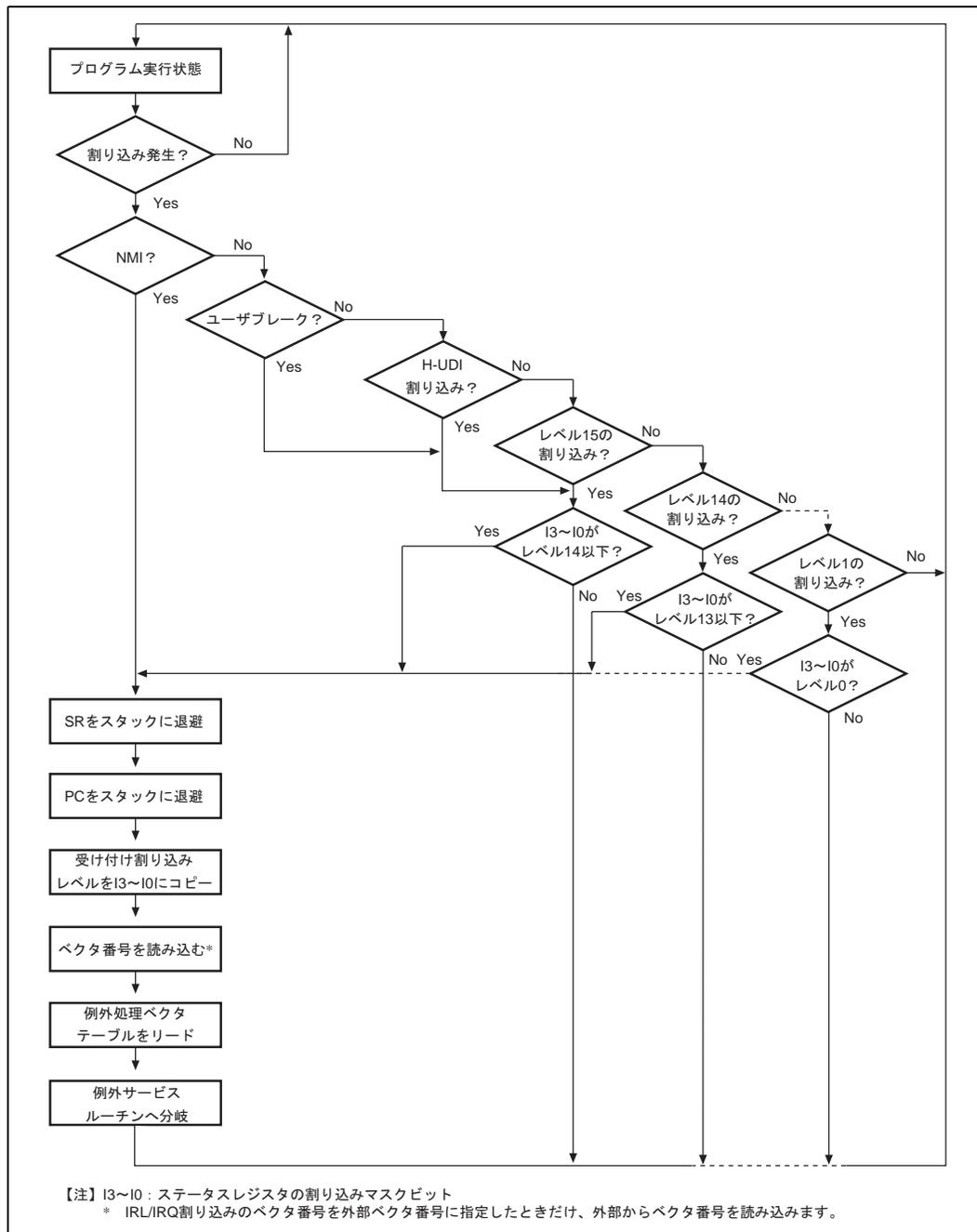


図 5.8 割り込み動作フロー

## 5.4.2 割り込み例外処理終了後のスタック状態

割り込み例外処理終了後のスタックの状態は、図 5.9 に示すようになります。

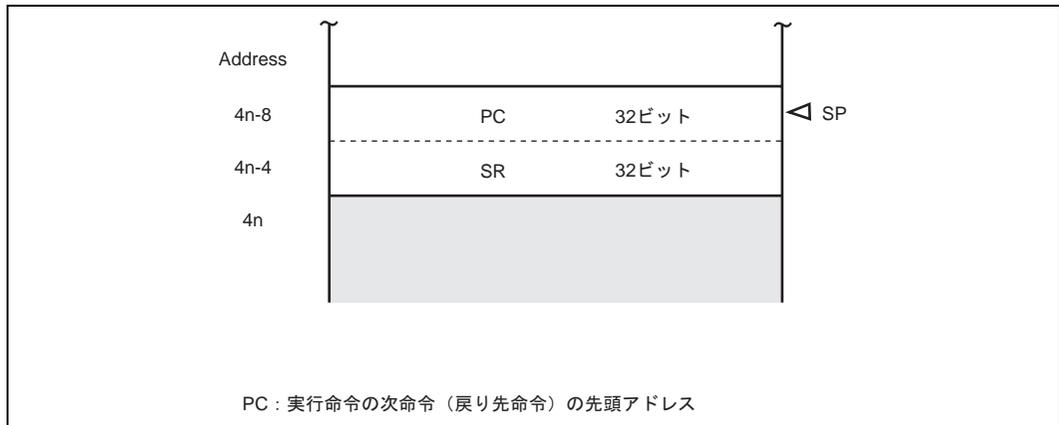


図 5.9 割り込み例外処理終了後のスタック状態

## 5. 割り込みコントローラ (INTC)

### 5.5 割り込み応答時間

割り込み要求が発生してから、割り込み例外処理が行われ、例外サービスルーチンの先頭命令のフェッチが開始されるまでの時間 (割り込み応答時間) を表 5.8 に示します。

表 5.8 割り込み応答時間

項目	ステート数				備考	
	NMI	IRL/IRQ	周辺モジュール			
			A	B		
優先順位判定および SR のマスクビットとの比較時間	$2.0 \times \text{Icyc}$	$0.5 \times \text{Icyc} + 1.0 \times \text{Ecyc} + 1.5 \times \text{Pcyc}$	$0.5 \times \text{Icyc} + 1.0 \times \text{Pcyc}$	$1.0 \times \text{Pcyc}$		
CPU が実行中のシーケンス終了までの待ち時間	$X ( 0 )$				最も長いシーケンスは割り込み例外処理またはアドレスエラー例外処理で、 $X = 4.0 \times \text{Icyc} + m1 + m2 + m3 + m4$ 。ただし、割り込みマスクする命令が続く場合やリビート命令実行中はさらに長くなることもあります。	
割り込み例外処理 (SR、PC の退避とベクタアドレスのフェッチ) から、例外サービスルーチンの先頭命令のフェッチを開始するまでの時間	$5.0 \times \text{Icyc} + m1 + m2 + m3$					
応答時間	合計	$X + 7.0 \times \text{Icyc} + m1 + m2 + m3$	$X + 5.5 \times \text{Icyc} + 1.0 \times \text{Ecyc} + 1.5 \times \text{Pcyc} + m1 + m2 + m3$	$X + 5.5 \times \text{Icyc} + 1.0 \times \text{Pcyc} + m1 + m2 + m3$	$X + 5.0 \times \text{Icyc} + 1.0 \times \text{Pcyc} + m1 + m2 + m3$	
	最小時	10	11	9.5	9	I : E : P = 1 : 1 : 1
	最大時	$11 + 2(m1 + m2 + m3) + m4$	$19.5 + 2(m1 + m2 + m3) + m4$	$13.5 + 2(m1 + m2 + m3) + m4$	$13.0 + 2(m1 + m2 + m3) + m4$	I : E : P = 1 : 1/4 : 1/4

【注】 m1 ~ m4 は下記のメモリアクセスに要するステート数です。

m1 : SR の退避 (ロングワードライト)

m2 : PC の退避 (ロングワードライト)

m3 : ベクタアドレスリード (ロングワードリード)

m4 : 割り込みサービスルーチン先頭命令のフェッチ

Icyc : I の 1cycle 時間

Ecyc : E の 1cycle 時間

Pcyc : P の 1cycle 時間

周辺モジュール A : DMAC、REF (BSC)

周辺モジュール B : WDT、FRT、TPU、SCIF、SIO、E-DMAC

## 5.6 $\overline{\text{IRL3}} \sim \overline{\text{IRL0}}$ 端子のサンプリング

割り込み端子  $\overline{\text{IRL3}} \sim \overline{\text{IRL0}}$  は図 5.10 のようにノイズキャンセラを經由して、その後、割り込みコントローラによって CPU へ割り込み要求として伝えられます。ノイズキャンセラは短い周期で変化するノイズをキャンセルする機能を持っています。CPU は各命令の実行間で割り込み要求をサンプリングします。

この間ノイズキャンセラの出力はノイズを除去した端子のレベルにしたがって変化するので、CPU がサンプリングするまで端子のレベルを保持する必要があります。したがって通常は割り込みルーチン内で割り込み要因をクリアしてください。

外部ベクタフェッチを行う場合には、外部ベクタのフェッチサイクルを検出して割り込み要因をクリアすることもできます。

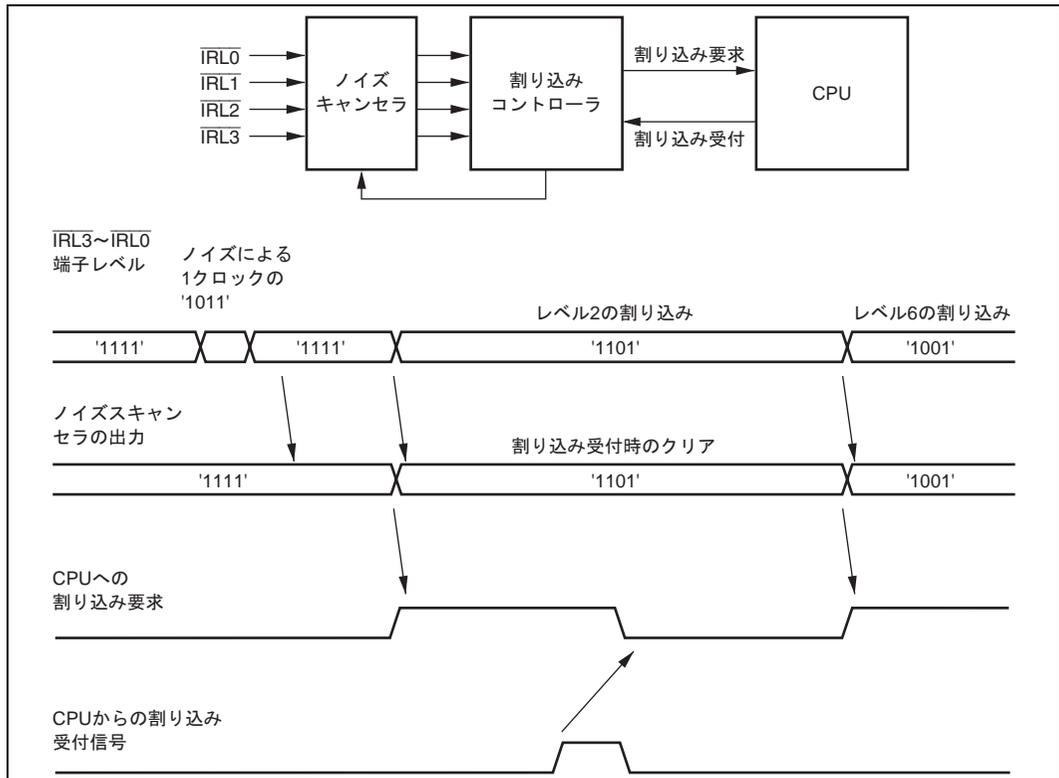


図 5.10  $\overline{\text{IRL3}} \sim \overline{\text{IRL0}}$  端子のサンプリング

## 5.7 使用上の注意

### 5.7.1 モジュールスタンバイ実行に関する注意事項

モジュールスタンバイ機能があるモジュールからの割り込みについては、割り込み要求が出力される可能性が残ったままモジュールスタンバイを実行しないでください。

### 5.7.2 割り込み要因のクリアに関する注意事項

#### (1) 外部割り込み要因をクリアする場合

割り込み要因のクリアを IO アドレス (外部) に対する書き込みで行う場合、ライトバッファの影響で書き込み動作が完了する前に次の命令が実行されてしまいます。書き込み動作が完了してから次の命令が実行されるようにするため、書き込みに続いて同一アドレスからの読み出しを行うと同期がとれます。

#### (a) RTE 命令によって割り込み処理から復帰する場合

図 5.11 に示すように RTE 命令によって割り込み処理から復帰する場合、設定したクロック比 (I : E : P) と外部バスサイクルによって、同期のための読み出し命令と RTE 命令の間に挿入するサイクルを検討してください。

IRL3 ~ IRL0 は、次の割り込みが受け付け可能になる  $0.5I_{cyc} + 1.0E_{cyc} + 1.5P_{cyc}$  以上前にネゲートしてください。

クロック比が I : E : P = 4 : 2 : 2 の場合、 $5.5I_{cyc}$  以上挿入されるようにしてください。

#### (b) 割り込み処理の中でレベルを変える場合

LDC 命令によって SR の値を変更し、別の割り込みが多重にかかることを許可する場合にも、設定したクロック比 (I : E : P) と外部バスサイクルから、図 5.12 に示すように同期命令と LDC 命令の間に挿入するサイクルを検討してください。

IRL3 ~ IRL0 は、次の割り込みが受け付け可能になる  $0.5I_{cyc} + 1.0E_{cyc} + 1.5P_{cyc}$  以上前にネゲートしてください。

クロック比が I : E : P = 4 : 2 : 2 の場合、 $5.5I_{cyc}$  以上挿入されるようにしてください。

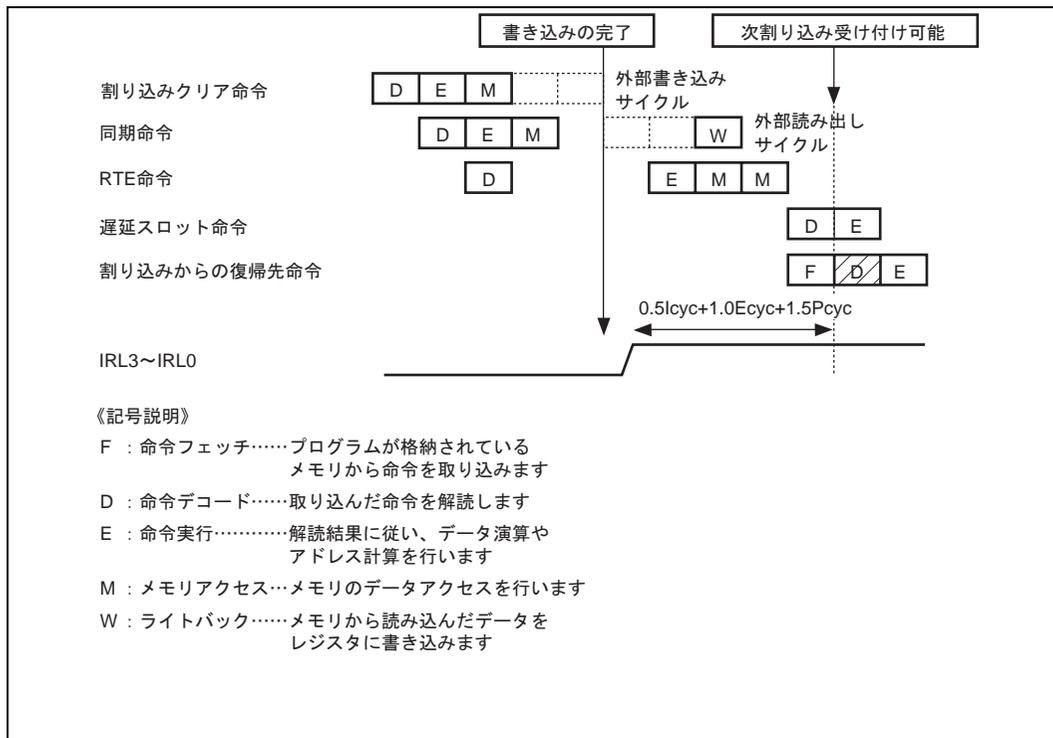


図 5.11 RTE 命令による復帰時のパイプライン動作

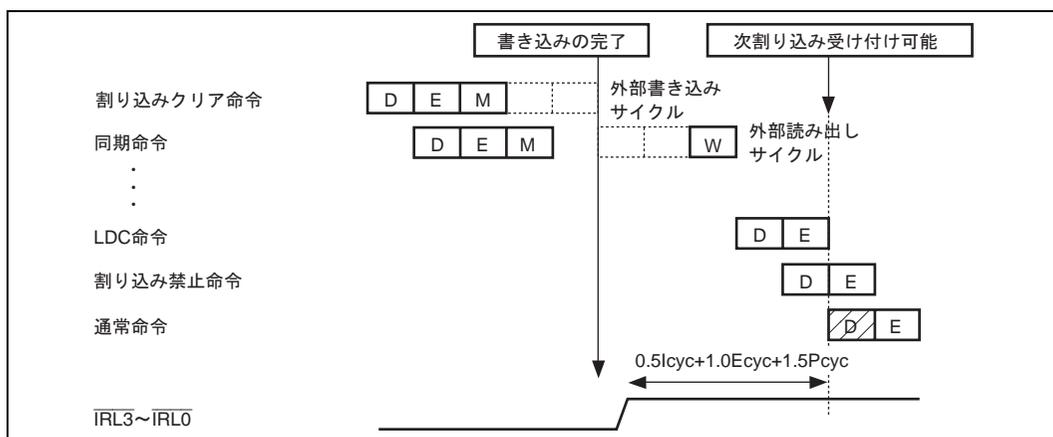


図 5.12 SR 変更による割り込み許可時のパイプライン動作

プログラムによって割り込み要因をクリアする場合、再度同じ割り込みがかからないようにするためには、パイプライン動作を考慮する必要があります。

## 5. 割り込みコントローラ (INTC)

### (2) 内蔵割り込み要因をクリアする場合

割り込み要因が内蔵周辺モジュールからの場合も、再度同じ割り込みがかからないようにするためには、パイプライン動作を考慮する必要があります。内蔵周辺モジュールからの割り込みが CPU に認識されるまでには  $0.5I_{cyc}+1.0P_{cyc}$  が必要です。同様に割り込み要求が無くなったことが伝わるためにも  $0.5I_{cyc}+1.0P_{cyc}$  が必要です。

#### (a) RTE 命令によって割り込み処理から復帰する場合

図 5.13 に示すように RTE によって割り込み処理から復帰する場合、設定したクロック比 (I : E : P ) によって、同期のための読み出し命令と RTE 命令の間に挿入するサイクルを検討してください。

内蔵周辺割り込み信号は、次の割り込みが受け付け可能になる  $0.5I_{cyc}+1.0P_{cyc}$  以上前にネゲートしてください。

クロック比が I : E : P =4 : 2 : 2 の場合、 $2.5I_{cyc}$  以上挿入されるようにしてください。

#### (b) 割り込み処理の中でレベルを変える場合

LDC 命令によって SR の値を変更し、別の割り込みが多重にかかることを許可する場合には、設定したクロック比 (I : E : P ) によって、図 5.14 に示すように同期命令と LDC 命令の間に挿入するサイクルを検討してください。

内蔵周辺割り込み信号は、次の割り込みが受け付け可能になる  $0.5I_{cyc}+1.0P_{cyc}$  以上前にネゲートしてください。

クロック比が I : E : P =4 : 2 : 2 の場合、 $2.5I_{cyc}$  以上挿入されるようにしてください。

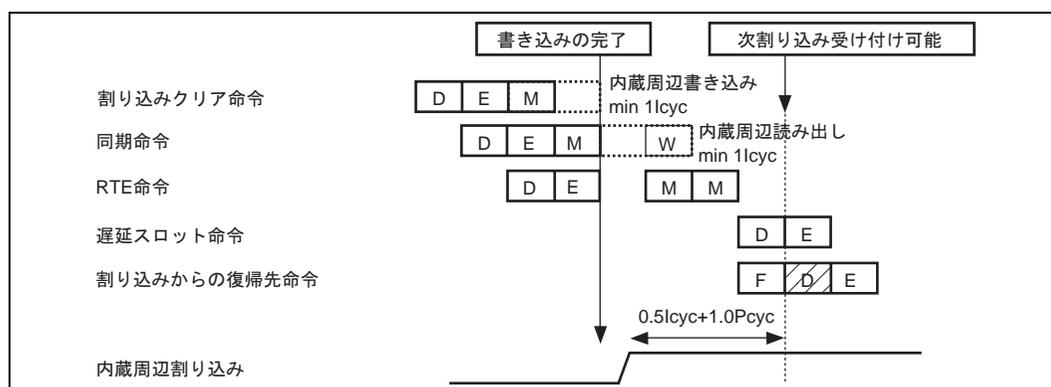


図 5.13 RTE 命令による復帰時のパイプライン動作

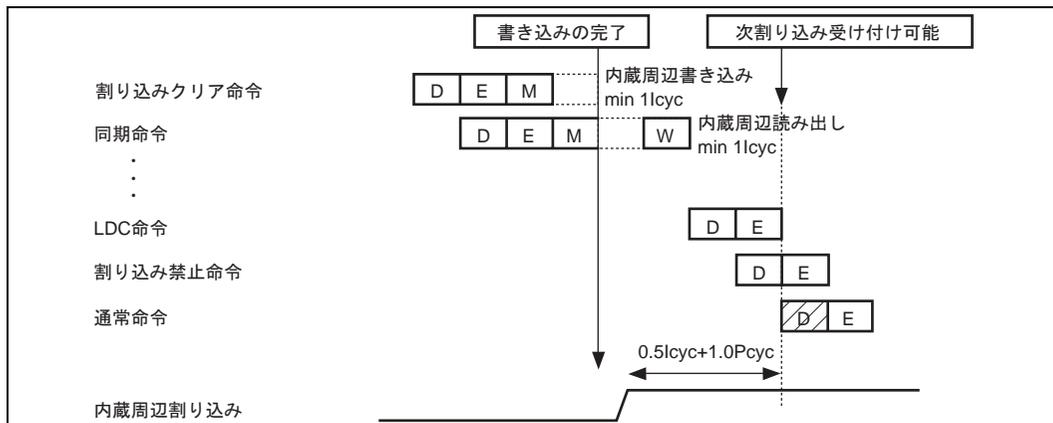


図 5.14 SR 変更による割り込み許可時のパイプライン動作

上記図において、本 LSI では DSP 命令混在のため命令フェッチが発生するステージが特定できないため、パイプライン動作中に命令フェッチの F をほとんどの場合省略しています。



---

## 6. ユーザブレイクコントローラ (UBC)

---

### 6.1 概要

ユーザブレイクコントローラ (UBC) は、ユーザのプログラムデバッグを容易にする機能を提供します。UBC にブレイク条件を設定すると、CPU、内蔵 DMAC (DMAC、E-DMAC) が発生するバスサイクルの内容に応じて、ユーザブレイク割り込みを発生します。

この機能を使用することにより、高機能のセルフモニタデバッグを容易に作成でき、インサーキットエミュレータを使用しなくても、本 LSI 単体で手軽にプログラムをデバッグできます。

#### 6.1.1 特長

ユーザブレイクコントローラは、以下の特長があります。

- ブレイク条件として以下を設定可能
    - ブレイクチャンネル数：4 チャンネル (チャンネル A、B、C および D)
    - チャンネル A、B、C および D について単独条件または、シーケンシャル条件でユーザブレイク割り込みを要求。
    - シーケンシャルブレイク設定：
      - チャンネル A チャンネル B チャンネル C チャンネル D
      - チャンネル B チャンネル C チャンネル D
      - チャンネル C チャンネル D
  
  - (1) アドレス：32 ビットマスク可能、各アドレスの設定が可能 (キャッシュバス (CPU)、内部バス (DMAC、E-DMAC)、X/Y バス)
  - (2) データ (チャンネル C、D のみ)：32 ビットマスク可能、各アドレスの設定が可能 (キャッシュバス (CPU)、内部バス (DMAC、E-DMAC)、X/Y バス)
  - (3) バスマスタ：CPU サイクル / 内蔵 DMAC (DMAC、E-DMAC) サイクル
  - (4) バスサイクル：命令フェッチ / データアクセス
  - (5) リード / ライト
  - (6) オペランドサイクル：バイト / ワード / ロングワード
- 
- ブレイク条件成立により、ユーザブレイク割り込みを発生
    - ユーザが作成したユーザブレイク割り込み例外ルーチンを実行させることができます。
  - 命令フェッチサイクルで命令の実行前に停止するか、実行後に停止するか選択可能

## 6. ユーザブレイクコントローラ (UBC)

---

- 実行回数指定ブレイク (チャンネル C、D のみ)
- 設定可能実行回数: 最大  $2^{12} - 1$  回 (4095 回)
- PC トレース機能  
分岐命令フェッチ時の分岐元 / 分岐先アドレスをトレース可能 (最大 4 組 8 アドレス)

## 6.1.2 ブロック図

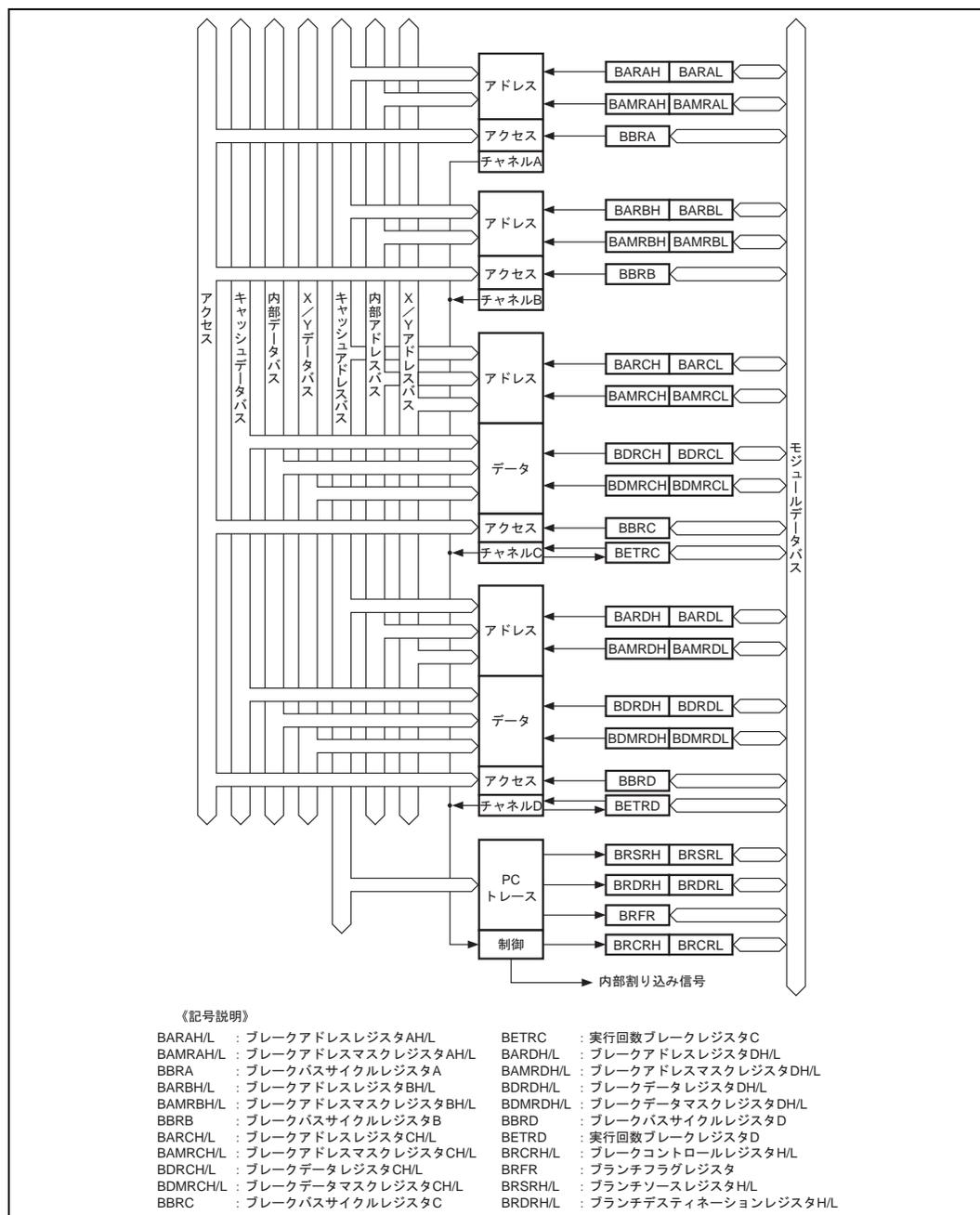


図 6.1 ユーザブレイクコントローラのブロック図

## 6. ユーザブ레이크コントローラ ( UBC )

### 6.1.3 レジスタ構成

表 6.1 レジスタ構成

名 称	略称	R/W	初期値 *1	アドレス	アクセスサイズ *2	
ブ레이크アドレスレジスタ AH	BARAH	R/W	H'0000	H'FFFFFF00	16	32
ブ레이크アドレスレジスタ AL	BARAL	R/W	H'0000	H'FFFFFF02	16	
ブ레이크アドレスマスクレジスタ AH	BAMRAH	R/W	H'0000	H'FFFFFF04	16	32
ブ레이크アドレスマスクレジスタ AL	BAMRAL	R/W	H'0000	H'FFFFFF06	16	
ブ레이크バスサイクルレジスタ A	BBRA	R/W	H'0000	H'FFFFFF08	16, 32	
ブ레이크アドレスレジスタ BH	BARBH	R/W	H'0000	H'FFFFFF20	16	32
ブ레이크アドレスレジスタ BL	BARBL	R/W	H'0000	H'FFFFFF22	16	
ブ레이크アドレスマスクレジスタ BH	BAMRBH	R/W	H'0000	H'FFFFFF24	16	32
ブ레이크アドレスマスクレジスタ BL	BAMRBL	R/W	H'0000	H'FFFFFF26	16	
ブ레이크バスサイクルレジスタ B	BBRB	R/W	H'0000	H'FFFFFF28	16, 32	
ブ레이크アドレスレジスタ CH	BARCH	R/W	H'0000	H'FFFFFF40	16	32
ブ레이크アドレスレジスタ CL	BARCL	R/W	H'0000	H'FFFFFF42	16	
ブ레이크アドレスマスクレジスタ CH	BAMRCH	R/W	H'0000	H'FFFFFF44	16	32
ブ레이크アドレスマスクレジスタ CL	BAMRCL	R/W	H'0000	H'FFFFFF46	16	
ブ레이크データレジスタ CH	BDRCH	R/W	H'0000	H'FFFFFF50	16	32
ブ레이크データレジスタ CL	BDRCL	R/W	H'0000	H'FFFFFF52	16	
ブ레이크データマスクレジスタ CH	BDMRCH	R/W	H'0000	H'FFFFFF54	16	32
ブ레이크データマスクレジスタ CL	BDMRCL	R/W	H'0000	H'FFFFFF56	16	
ブ레이크バスサイクルレジスタ C	BBRC	R/W	H'0000	H'FFFFFF48	16, 32	
実行回数ブ레이크レジスタ C	BETRC	R/W	H'0000	H'FFFFFF58	16, 32	
ブ레이크アドレスレジスタ DH	BARDH	R/W	H'0000	H'FFFFFF60	16	32
ブ레이크アドレスレジスタ DL	BARDL	R/W	H'0000	H'FFFFFF62	16	
ブ레이크アドレスマスクレジスタ DH	BAMRDH	R/W	H'0000	H'FFFFFF64	16	32
ブ레이크アドレスマスクレジスタ DL	BAMRDL	R/W	H'0000	H'FFFFFF66	16	
ブ레이크データレジスタ DH	BDRDH	R/W	H'0000	H'FFFFFF70	16	32
ブ레이크データレジスタ DL	BDRDL	R/W	H'0000	H'FFFFFF72	16	
ブ레이크データマスクレジスタ DH	BDMRDH	R/W	H'0000	H'FFFFFF74	16	32
ブ레이크データマスクレジスタ DL	BDMRDL	R/W	H'0000	H'FFFFFF76	16	
ブ레이크バスサイクルレジスタ D	BBRD	R/W	H'0000	H'FFFFFF68	16, 32	
実行回数ブ레이크レジスタ D	BETRD	R/W	H'0000	H'FFFFFF78	16, 32	
ブ레이크コントロールレジスタ H	BRCRH	R/W	H'0000	H'FFFFFF30	16	32
ブ레이크コントロールレジスタ L	BRCRL	R/W	H'0000	H'FFFFFF32	16	
ブランチフラグレジスタ	BRFR	R	*3	H'FFFFFF10	16, 32	
ブランチソースレジスタ H	BRSRH	R	不定	H'FFFFFF14	16	32
ブランチソースレジスタ L	BRSRL	R	不定	H'FFFFFF16	16	
ブランチデスティネーションレジスタ H	BRDRH	R	不定	H'FFFFFF18	16	32
ブランチデスティネーションレジスタ L	BRDRL	R	不定	H'FFFFFF1A	16	

【注】 \*1 パワーオンリセットで初期化。スタンバイでは値を保持、マニュアルリセットでは不定値になります。

\*2 バイトアクセスはできません。

\*3 BRFR の SVF ビットと DVF ビットは、パワーオンリセットで初期化されます。BRFR の他のビットは初期化されません。

## 6.2 各レジスタの説明

### 6.2.1 ブ레이크アドレスレジスタ A (BARA)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARAH	BAA 31	BAA 30	BAA 29	BAA 28	BAA 27	BAA 26	BAA 25	BAA 24	BAA 23	BAA 22	BAA 21	BAA 20	BAA 19	BAA 18	BAA 17	BAA 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARAL	BAA 15	BAA 14	BAA 13	BAA 12	BAA 11	BAA 10	BAA 9	BAA 8	BAA 7	BAA 6	BAA 5	BAA 4	BAA 3	BAA 2	BAA 1	BAA 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブ레이크アドレスレジスタ A (BARA) は、ブ레이크アドレスレジスタ AH (BARAH) とブ레이크アドレスレジスタ AL (BARAL) の 2 本で 1 組となっています。BARAH と BARAL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタで、BARAH はチャンネル A のブ레이크条件とするアドレスの上位側 (ビット 31 ~ 16) を指定し、BARAL はアドレスの下位側 (ビット 15 ~ 0) を指定します。BARAH と BARAL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ BARAH ビット 15 ~ 0: ブ레이크アドレス A31 ~ 16 (BAA31 ~ 16)  
チャンネル A のブ레이크条件とするアドレスの上位側 (ビット 31 ~ 16) を格納します。
- ・ BARAL ビット 15 ~ 0: ブ레이크アドレス A15 ~ 0 (BAA15 ~ 0)  
チャンネル A のブ레이크条件とするアドレスの下位側 (ビット 15 ~ 0) を格納します。

### 6.2.2 ブ레이크アドレスマスクレジスタ A (BAMRA)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAMRAH	BAMA 31	BAMA 30	BAMA 29	BAMA 28	BAMA 27	BAMA 26	BAMA 25	BAMA 24	BAMA 23	BAMA 22	BAMA 21	BAMA 20	BAMA 19	BAMA 18	BAMA 17	BAMA 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAMRAL	BAMA 15	BAMA 14	BAMA 13	BAMA 12	BAMA 11	BAMA 10	BAMA 9	BAMA 8	BAMA 7	BAMA 6	BAMA 5	BAMA 4	BAMA 3	BAMA 2	BAMA 1	BAMA 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブ레이크アドレスマスクレジスタ A (BAMRA) は、ブ레이크アドレスマスクレジスタ AH (BAMRAH) とブ레이크アドレスマスクレジスタ AL (BAMRAL) の 2 本で 1 組になっています。BAMRAH と BAMRAL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタです。BAMRAH は BARAH に設定されているブ레이크アド

## 6. ユーザブレイクコントローラ (UBC)

レスのどのビットをマスクするかを指定し、BAMRAL は、BARAL に設定されているブレイクアドレスのどのビットをマスクするかを指定します。BAMRAH と BAMRAL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ BAMRAH ビット 15 ~ 0 : ブレイクアドレスマスク A31 ~ 16 (BAMA31 ~ 16)  
BARAH に設定されているチャンネル A のブレイクアドレス 31 ~ 16 (BAA31 ~ BAA16) の各ビットをマスクするかどうかを指定します。
- ・ BAMRAL ビット 15 ~ 0 : ブレイクアドレスマスク A15 ~ 0 (BAMA15 ~ 0)  
BARAL に設定されているチャンネル A のブレイクアドレス 15 ~ 0 (BAA15 ~ BAA0) の各ビットをマスクするかどうかを指定します。

ビット 31 ~ 0	説明
BAMAn	
0	チャンネル A のブレイクアドレス BAA <sub>n</sub> をブレイク条件に含める (初期値)
1	チャンネル A のブレイクアドレス BAA <sub>n</sub> をマスクし、条件に含めない

【注】 【注】 n=31 ~ 0

### 6.2.3 ブレイクバスサイクルレジスタ A (BBRA)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BBRA	—	—	—	—	—	—	—	—	CPA1	CPA0	IDA1	IDA0	RWA1	RWA0	SZA1	SZA0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W							

ブレイクバスサイクルレジスタ A は、読み出し / 書き込み可能な 16 ビットレジスタで、チャンネル A のブレイク条件のうち、(1) CPU サイクル / 内蔵 DMAC (DMAC、E-DMAC) サイクル (2) 命令フェッチ / データアクセス (3) 読み出し / 書き込み (4) オペランドサイズの 4 条件を設定するレジスタです。BBRA は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ ビット 15 ~ 8 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 7、6 : CPU / DMAC、E-DMAC サイクルセレクト A (CPA1、CPA0)  
チャンネル A のブレイク条件にするバスサイクルを CPU サイクルにするか DMAC もしくは E-DMAC サイクルにするかを選択します。

ビット7	ビット6	説明
CPA1	CPA0	
0	0	チャンネルAのユーザブレイク割り込みは発生しない (初期値)
	1	CPUサイクルをユーザブレイク条件とする
1	0	DMAC、E-DMACサイクルのいずれもユーザブレイク条件とする
	1	CPU、DMAC、E-DMACサイクルのいずれもユーザブレイク条件とする

- ・ ビット5、4：命令のフェッチ/データアクセスセレクトA (IDA1、IDA0)  
チャンネルAのブレイク条件にするバスサイクルを命令フェッチサイクルにするかデータアクセスサイクルにするかを選択します。

ビット5	ビット4	説明
IDA1	IDA0	
0	0	チャンネルAのユーザブレイク割り込みは発生しない (初期値)
	1	命令フェッチサイクルをブレイク条件とする
1	0	データアクセスサイクルをブレイク条件とする
	1	命令フェッチサイクル、データアクセスサイクルのいずれもブレイク条件とする

- ・ ビット3、2：リード/ライトセレクトA (RWA1、RWA0)  
チャンネルAのブレイク条件にするバスサイクルをリードサイクルにするかライトサイクルにするかを選択します。

ビット3	ビット2	説明
RWA1	RWA0	
0	0	チャンネルAのユーザブレイク割り込みは発生しない (初期値)
	1	リードサイクルをブレイク条件とする
1	0	ライトサイクルをブレイク条件とする
	1	リードサイクル、ライトサイクルのいずれもブレイク条件とする

- ・ ビット1、0：オペランドサイズセレクトA (SZA1、SZA0)  
チャンネルAのブレイク条件にするバスサイクルのオペランドサイズを選択します。

ビット1	ビット0	説明
SZA1	SZA0	
0	0	ブレイク条件にオペランドサイズを含まない (初期値)
	1	バイトアクセスをブレイク条件とする
1	0	ワードアクセスをブレイク条件とする
	1	ロングワードアクセスをブレイク条件とする

【注】 命令フェッチでブレイクをかける場合は、SZA0ビットを0にしてください。すべての命令は、ワードサイズでアクセスされるものとみなされます (命令フェッチはつねにロングワードで行われます)。オペランドサイズは命令の場合はワード、CPU/DMAC、E-DMACのデータアクセスの場合はその指定したオペランドサイズで決まります。アクセスする空間のバス幅によって決まるものではありません。

## 6.2.4 ブレークアドレスレジスタ B (BARB)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARBH	BAB 31	BAB 30	BAB 29	BAB 28	BAB 27	BAB 26	BAB 25	BAB 24	BAB 23	BAB 22	BAB 21	BAB 20	BAB 19	BAB 18	BAB 17	BAB 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARBL	BAB 15	BAB 14	BAB 13	BAB 12	BAB 11	BAB 10	BAB 9	BAB 8	BAB 7	BAB 6	BAB 5	BAB 4	BAB 3	BAB 2	BAB 1	BAB 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブレークアドレスレジスタ B (BARB) は、ブレークアドレスレジスタ BH (BARBH) とブレークアドレスレジスタ BL (BARBL) の 2 本で 1 組となっています。BARBH と BARBL は、それぞれ読み出し/書き込み可能な 16 ビットのレジスタで、BARBH はチャンネル B のブレーク条件とするアドレスの上位側 (ビット 31 ~ 16) を指定し、BARBL はアドレスの下位側 (ビット 15 ~ 0) を指定します。BARBH と BARBL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ BARBH ビット 15 ~ 0: ブレークアドレス B31 ~ 16 (BAB31 ~ 16)  
チャンネル B のブレーク条件とするアドレスの上位側 (ビット 31 ~ 16) を格納します。
- ・ BARBL ビット 15 ~ 0: ブレークアドレス B15 ~ 0 (BAB15 ~ 0)  
チャンネル B のブレーク条件とするアドレスの下位側 (ビット 15 ~ 0) を格納します。

## 6.2.5 ブレークアドレスマスクレジスタ B (BAMRB)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAMRBH	BAMB 31	BAMB 30	BAMB 29	BAMB 28	BAMB 27	BAMB 26	BAMB 25	BAMB 24	BAMB 23	BAMB 22	BAMB 21	BAMB 20	BAMB 19	BAMB 18	BAMB 17	BAMB 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BAMRBL	BAMB 15	BAMB 14	BAMB 13	BAMB 12	BAMB 11	BAMB 10	BAMB 9	BAMB 8	BAMB 7	BAMB 6	BAMB 5	BAMB 4	BAMB 3	BAMB 2	BAMB 1	BAMB 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブレークアドレスマスクレジスタ B (BAMRB) は、ブレークアドレスマスクレジスタ BH (BAMRBH) とブレークアドレスマスクレジスタ BL (BAMRBL) の 2 本で 1 組になっています。BAMRBH と BAMRBL は、それぞれ読み出し/書き込み可能な 16 ビットのレジスタです。BAMRBH は BARBH に設定されているブレークアド

レスのどのビットをマスクするかを指定し、BAMRBL は、BARBL に設定されているブ레이크アドレスのどのビットをマスクするかを指定します。BAMRBH と BAMRBL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ BAMRBH ビット 15 ~ 0 : ブ레이크アドレスマスク B31 ~ 16 (BAMB31 ~ 16)  
BARBH に設定されているチャンネル B のブ레이크アドレス 31 ~ 16 (BAB31 ~ BAB16) の各ビットをマスクするかどうかを指定します。
- ・ BAMRBL ビット 15 ~ 0 : ブ레이크アドレスマスク B15 ~ 0 (BAMB15 ~ 0)  
BARBL に設定されているチャンネル B のブ레이크アドレス 15 ~ 0 (BAB15 ~ BAB0) の各ビットをマスクするかどうかを指定します。

ビット 31 ~ 0	説明
BAMBn	
0	チャンネル B のブ레이크アドレス BABn をブ레이크条件に含める (初期値)
1	チャンネル B のブ레이크アドレス BABn をマスクし、条件に含めない

【注】 n=31 ~ 0

### 6.2.6 ブ레이크バスサイクルレジスタ B (BBRB)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BBRB	—	—	—	—	—	—	—	—	CPB1	CPB0	IDB1	IDB0	RWB1	RWB0	SZB1	SZB0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W							

ブ레이크バスサイクルレジスタ B は、読み出し/書き込み可能な 16 ビットレジスタで、チャンネル B のブ레이크条件のうち、(1) CPU サイクル/内蔵 DMAC (DMAC、E-DMAC) サイクル (2) 命令フェッチ/データアクセス (3) 読み出し/書き込み (4) オペランドサイズの 4 条件を設定するレジスタです。BBRB は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ ビット 15 ~ 8 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 7、6 : CPU / DMAC、E-DMAC サイクルセレクト B (CPB1、CPB0)  
チャンネル B のブ레이크条件にするバスサイクルを CPU サイクルにするか DMAC もしくは E-DMAC サイクルにするかを選択します。

## 6. ユーザブレイクコントローラ ( UBC )

ビット7	ビット6	説 明
CPB1	CPB0	
0	0	チャンネルBのユーザブレイク割り込みは発生しない (初期値)
	1	CPUサイクルをユーザブレイク条件とする
1	0	DMAC、E-DMACサイクルのいずれもユーザブレイク条件とする
	1	CPU、DMAC、E-DMACサイクルのいずれもユーザブレイク条件とする

- ・ ビット5、4：命令のフェッチ/データアクセスセレクトB ( IDB1、IDB0 )  
チャンネルBのブレイク条件にするバスサイクルを命令フェッチサイクルにするかデータアクセスサイクルにするかを選択します。

ビット5	ビット4	説 明
IDB1	IDB0	
0	0	チャンネルBのユーザブレイク割り込みは発生しない (初期値)
	1	命令フェッチサイクルをブレイク条件とする
1	0	データアクセスサイクルをブレイク条件とする
	1	命令フェッチサイクル、データアクセスサイクルのいずれもブレイク条件とする

- ・ ビット3、2：リード/ライトセレクトB ( RWB1、RWB0 )  
チャンネルBのブレイク条件にするバスサイクルをリードサイクルにするかライトサイクルにするかを選択します。

ビット3	ビット2	説 明
RWB1	RWB0	
0	0	チャンネルBのユーザブレイク割り込みは発生しない (初期値)
	1	リードサイクルをブレイク条件とする
1	0	ライトサイクルをブレイク条件とする
	1	リードサイクル、ライトサイクルのいずれもブレイク条件とする

- ・ ビット1、0：オペランドサイズセレクトB ( SZB1、SZB0 )  
チャンネルBのブレイク条件にするバスサイクルのオペランドサイズを選択します。

ビット1	ビット0	説 明
SZB1	SZB0	
0	0	ブレイク条件にオペランドサイズを含まない (初期値)
	1	バイトアクセスをブレイク条件とする
1	0	ワードアクセスをブレイク条件とする
	1	ロングワードアクセスをブレイク条件とする

【注】 命令フェッチでブレイクをかける場合は、SZB0ビットを0にしてください。すべての命令は、ワードサイズでアクセスされるものとみなされます (命令フェッチはつねにロングワードで行われます)。オペランドサイズは命令の場合はワード、CPU/DMAC、E-DMACのデータアクセスの場合はその指定したオペランドサイズで決まります。アクセスする空間のバス幅によって決まるものではありません。

## 6.2.7 ブレークアドレスレジスタ C (BARC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARC	BAC 31	BAC 30	BAC 29	BAC 28	BAC 27	BAC 26	BAC 25	BAC 24	BAC 23	BAC 22	BAC 21	BAC 20	BAC 19	BAC 18	BAC 17	BAC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BARCL	BAC 15	BAC 14	BAC 13	BAC 12	BAC 11	BAC 10	BAC 9	BAC 8	BAC 7	BAC 6	BAC 5	BAC 4	BAC 3	BAC 2	BAC 1	BAC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ブレークアドレスレジスタ C (BARC) は、ブレークアドレスレジスタ CH (BARC) とブレークアドレスレジスタ CL (BARCL) の 2 本で 1 組となっています。BARC と BARCL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタで、BARC はチャンネル C のブレーク条件とするアドレスの上位側 (ビット 31 ~ 16) を指定し、BARCL はアドレスの下位側 (ビット 15 ~ 0) を指定します。また、ブレークバスサイクルレジスタ C (BBRC) 内の XYEC ビット / XYSC ビットの設定によって、X/Y メモリに接続されているアドレスバスを、ブレーク条件として指定することができます。XYEC=0 の場合、BAC31 ~ 0 はアドレスを指定します。XYEC=1 の場合、BARC の上位側 16 ビット (BAC31 ~ 16) は X アドレスバスを、同様に下位側 16 ビット (BAC15 ~ 0) は Y アドレスバスを指定します。BARC と BARCL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

## BARC の構成

		上位 16 ビット (BAC31 ~ 16)	下位 16 ビット (BAC15 ~ 0)
XYEC=0	アドレス	アドレスバス上位 16 ビット	アドレスバス下位 16 ビット
XYEC=1	X アドレス (XYSC=0 時)	X アドレス (XAB15 ~ 1) *	
	Y アドレス (XYSC=1 時)		Y アドレス (YAB15 ~ 1) *

【注】 \* X/Y バスアクセスは常にワードアクセスのため、XAB0 と YAB0 の値はブレーク条件に含まれていません。

## 6. ユーザブ레이크コントローラ (UBC)

### 6.2.8 ブ레이크アドレスマスクレジスタ C (BAMRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCH	BDMC 31	BDMC 30	BDMC 29	BDMC 28	BDMC 27	BDMC 26	BDMC 25	BDMC 24	BDMC 23	BDMC 22	BDMC 21	BDMC 20	BDMC 19	BDMC 18	BDMC 17	BDMC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCL	BDMC 15	BDMC 14	BDMC 13	BDMC 12	BDMC 11	BDMC 10	BDMC 9	BDMC 8	BDMC 7	BDMC 6	BDMC 5	BDMC 4	BDMC 3	BDMC 2	BDMC 1	BDMC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブ레이크アドレスマスクレジスタ C (BAMRC) は、ブ레이크アドレスマスクレジスタ CH (BAMRCH) とブ레이크アドレスマスクレジスタ CL (BAMRCL) の 2 本で 1 組になっています。BAMRCH と BAMRCL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタです。BAMRCH は BARCH に設定されているブ레이크アドレスのどのビットをマスクするかを指定し、BAMRCL は、BARCL に設定されているブ레이크アドレスのどのビットをマスクするかを指定します。また、BBRC 内の XYEC ビットおよび、XYSC ビットによっては下記のようになります。

#### BAMRC の構成

		上位 16 ビット (BAMC31 ~ 16)	下位 16 ビット (BAMC15 ~ 0)
XYEC=0	アドレス	上位 16 ビットマスク可	下位 16 ビットマスク可
XYEC=1	X アドレス (XYSC=0 時)	マスク可	
	Y アドレス (XYSC=1 時)		マスク可

ビット 31 ~ 0	説 明
BAMCn	
0	チャンネル C のブ레이크アドレス BACn をブ레이크条件に含める (初期値)
1	チャンネル C のブ레이크アドレス BACn をマスクし、条件に含めない

【注】 n=31 ~ 0

## 6.2.9 ブレークデータレジスタ C (BDRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDRCH	BDC 31	BDC 30	BDC 29	BDC 28	BDC 27	BDC 26	BDC 25	BDC 24	BDC 23	BDC 22	BDC 21	BDC 20	BDC 19	BDC 18	BDC 17	BDC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDRCL	BDC 15	BDC 14	BDC 13	BDC 12	BDC 11	BDC 10	BDC 9	BDC 8	BDC 7	BDC 6	BDC 5	BDC 4	BDC 3	BDC 2	BDC 1	BDC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブレークデータレジスタ C (BDRC) は、ブレークデータレジスタ CH (BDRCH) とブレークデータレジスタ CL (BDRCL) の 2 本で 1 組となっています。BDRCH と BDRCL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタで、BDRCH はチャンネル C のブレーク条件とするデータの上位側 (ビット 31 ~ 16) を指定し、BDRCL はデータの下位側 (ビット 15 ~ 0) を指定します。また、ブレークバスサイクルレジスタ (BBRC) 内の XYEC ビット / XYSC ビットの設定によって、X/Y メモリに接続されているデータベースを指定することができます。XYEC=1 の場合、BDRC の上位側 16 ビット (BDC31 ~ 16) は X データバスを、同様に、下位側 16 ビット (BDC15 ~ 0) は Y データバスを指定します。

BDRC の構成

		上位 16 ビット (BDC31 ~ 16)	下位 16 ビット (BDC15 ~ 0)
XYEC=0	データ	データベース上位 16 ビット	データベース下位 16 ビット
XYEC=1	X データ (XYSC=0 時)	X データ (XDB15 ~ 0)	
	Y データ (XYSC=1 時)		Y データ (YDB15 ~ 0)

## 6.2.10 ブレークデータマスクレジスタ C (BDMRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCH	BDMC 31	BDMC 30	BDMC 29	BDMC 28	BDMC 27	BDMC 26	BDMC 25	BDMC 24	BDMC 23	BDMC 22	BDMC 21	BDMC 20	BDMC 19	BDMC 18	BDMC 17	BDMC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCL	BDMC 15	BDMC 14	BDMC 13	BDMC 12	BDMC 11	BDMC 10	BDMC 9	BDMC 8	BDMC 7	BDMC 6	BDMC 5	BDMC 4	BDMC 3	BDMC 2	BDMC 1	BDMC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

## 6. ユーザブレイクコントローラ (UBC)

ブレイクデータマスクレジスタ C (BDMRC) は、ブレイクマスクレジスタ CH (BDMRCH) とブレイクマスクレジスタ CL (BDMRCL) の 2 本で 1 組となっています。BDMRCH と BDMRCL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタです。BDMRCH は BDRCH に設定されているブレイクデータのどのビットをマスクするかを指定し、BDMRCL は、BDRCL に設定されているブレイクデータのどのビットをマスクするかを指定します。また、BBRC 内の X YEC ビット / X YSC ビットによって、下記のようになります。BDMRCH と BDMRCL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

BDMRC の構成

		上位 16 ビット (BDMC31 ~ 16)	下位 16 ビット (BDMC15 ~ 0)
X YEC=0	データ	上位 16 ビットマスク可	下位 16 ビットマスク可
X YEC=1	X データ (X YSC=0 時)	マスク可	
	Y データ (X YSC=1 時)		マスク可

ビット 31 ~ 0	説 明
BDMCn	
0	チャンネル C のブレイクデータ BDCn をブレイク条件に含める (初期値)
1	チャンネル C のブレイクデータ BDCn をマスクし、条件に含めない

- 【注】
- n = 31 ~ 0
  - データバスの値をブレイク条件に含めるときはオペランドサイズを指定してください。
  - バイトサイズを指定するときで、奇数アドレスのデータをブレイク条件とするときは、BDRC、BDMRC のビット 7 ~ 0 に値を設定してください。偶数アドレスのデータをブレイク条件とするときは、ビット 15 ~ 8 に値を設定してください。使わない 8 ビットレジスタはブレイク条件に影響を与えません。

### 6.2.11 ブレイクバスサイクルレジスタ C (BBRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BBRC	—	—	—	—	—	—	X YEC	X YSC	CPC1	CPC0	IDC1	IDC0	RWC1	RWC0	SZC1	SZC0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ブレイクバスサイクルレジスタ C (BBRC) は、読み出し / 書き込み可能な 16 ビットレジスタで、チャンネル C のブレイク条件のうち、(1) 内部バス (C バス、I バス) / X メモリバス / Y メモリバス (2) CPU サイクル / 内蔵 DMAC (DMAC、E-DMAC) サイクル (3) 命令フェッチ / データアクセス (4) 読み出し / 書き込み (5) オペランドサイズの 5 条件を設定するレジスタです。BBRC は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ビット 15 ~ 10: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット9: X/Y メモリバスイネーブル C (XYEC)  
チャンネル C のブレイク条件を X/Y バスにするかを選択します。

ビット9	説明
XYEC	
0	チャンネル C のアドレス / データは、キャッシュバスか内部バスを条件とする (初期値)
1	チャンネル C のアドレス / データは X/Y バスを条件とする

- ビット8: Xバス/Yバスセレクト C (XYSC)  
チャンネル C のブレイク条件を X バスにするか、Y バスにするかを選択します。本ビットは、XYEC ビット = 1 のときのみ有効です。

ビット8	説明
XYSC	
0	チャンネル C のブレイク条件を X バスとする (初期値)
1	チャンネル C のブレイク条件を Y バスとする

ビット7 ~ 0 のビット構成は BBRA と同様です。

## 6.2.12 実行回数ブレイクレジスタ C (BETRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BETRC	—	—	—	—	ETRC											
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W											

チャンネル C の実行回数ブレイク条件を (BRCR の ETBEC ビットに設定することによって) 有効にして、ユーザブレイク割り込みを要求する前にチャンネル C のブレイク条件が発生する回数を、この 16 ビットレジスタに指定します。最大値は  $2^{12} - 1$  回です。チャンネル C のブレイク条件を満たすたびに BETRC は 1 ずつデクリメントされます。BETRC が H'0001 になった後、ブレイク条件を満たすと割り込みが要求されます。

3 つ以下の命令で構成される繰り返しループ内の命令は、例外や割り込みを受け付けません (「4.6 例外要因が受け付けられない場合」参照)。したがって、BETRC は 3 つ以下の命令で構成される繰り返しループ内の命令に対して生じるブレイク条件一致に対してはデクリメントされません。

ビット 15 ~ 12 を読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

パワーオンリセットによって BETRC は H'0000 に初期化されます。

## 6. ユーザブ레이크コントローラ (UBC)

### 6.2.13 ブレークアドレスレジスタ D (BARD)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCH	BDMC 31	BDMC 30	BDMC 29	BDMC 28	BDMC 27	BDMC 26	BDMC 25	BDMC 24	BDMC 23	BDMC 22	BDMC 21	BDMC 20	BDMC 19	BDMC 18	BDMC 17	BDMC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCL	BDMC 15	BDMC 14	BDMC 13	BDMC 12	BDMC 11	BDMC 10	BDMC 9	BDMC 8	BDMC 7	BDMC 6	BDMC 5	BDMC 4	BDMC 3	BDMC 2	BDMC 1	BDMC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ブレークアドレスレジスタ D (BARD) は、ブレークアドレスレジスタ DH (BARDH) とブレークアドレスレジスタ DL (BARDL) の 2 本で 1 組となっています。BARDH と BARDL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタで、BARDH はチャンネル D のブレーク条件とするアドレスの上位側 (ビット 31 ~ 16) を指定し、BARDL はアドレスの下位側 (ビット 15 ~ 0) を指定します。また、ブレークバスサイクルレジスタ D (BBRD) 内の XYED ビット / XYSD ビットの設定によって、X/Y メモリに接続されているアドレスバスを、ブレーク条件として指定することができます。XYED=0 の場合、BAD31 ~ 0 はアドレスを指定します。XYED=1 の場合、BARD の上位側 16 ビット (BAD31 ~ 16) は X アドレスバスを、同様に下位側 16 ビット (BAD15 ~ 0) は Y アドレスバスを指定します。BARDH と BARDL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

#### BARD の構成

		上位 16 ビット (BAD31 ~ 16)	下位 16 ビット (BAD15 ~ 0)
XYED=0	アドレス	アドレスバス上位 16 ビット	アドレスバス下位 16 ビット
XYED=1	X アドレス (XYSD=0 時)	X アドレス (XAB15 ~ 1) *	
	Y アドレス (XYSD=1 時)		Y アドレス (YAB15 ~ 1) *

【注】 \* X/Y バスアクセスは常にワードアクセスのため、XAB0 と YAB0 の値はブレーク条件に含まれていません。

## 6.2.14 ブ레이크アドレスマスクレジスタ D (BAMRD)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCH	BDMC 31	BDMC 30	BDMC 29	BDMC 28	BDMC 27	BDMC 26	BDMC 25	BDMC 24	BDMC 23	BDMC 22	BDMC 21	BDMC 20	BDMC 19	BDMC 18	BDMC 17	BDMC 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRCL	BDMC 15	BDMC 14	BDMC 13	BDMC 12	BDMC 11	BDMC 10	BDMC 9	BDMC 8	BDMC 7	BDMC 6	BDMC 5	BDMC 4	BDMC 3	BDMC 2	BDMC 1	BDMC 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ブ레이크アドレスマスクレジスタ D (BAMRD) は、ブ레이크アドレスマスクレジスタ DH (BAMRDH) とブ레이크アドレスマスクレジスタ DL (BAMRDL) の 2 本で 1 組になっています。BAMRDH と BAMRDL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタです。BAMRDH は BARDH に設定されているブ레이크アドレスのどのビットをマスクするかを指定し、BAMRDL は、BARDL に設定されているブ레이크アドレスのどのビットをマスクするかを指定します。また、BBRD 内の XYED ビットおよび、XYSD ビットによっては下記のようになります。

## BAMRD の構成

		上位 16 ビット (BAMD31 ~ 16)	下位 16 ビット (BAMD15 ~ 0)
XYED=0	アドレス	上位 16 ビットマスク可	下位 16 ビットマスク可
XYED=1	X アドレス (XYSD=0 時)	マスク可	
	Y アドレス (XYSD=1 時)		マスク可

ビット 31 ~ 0	説明
BAMDn	
0	チャンネル D のブ레이크アドレス BADn をブ레이크条件に含める (初期値)
1	チャンネル D のブ레이크アドレス BADn をマスクし、条件に含めない

【注】 n=31 ~ 0

## 6. ユーザブ레이크コントローラ (UBC)

### 6.2.15 ブ레이크データレジスタ D (BDRD)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDRDH	BDD 31	BDD 30	BDD 29	BDD 28	BDD 27	BDD 26	BDD 25	BDD 24	BDD 23	BDD 22	BDD 21	BDD 20	BDD 19	BDD 18	BDD 17	BDD 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDRDL	BDD 15	BDD 14	BDD 13	BDD 12	BDD 11	BDD 10	BDD 9	BDD 8	BDD 7	BDD 6	BDD 5	BDD 4	BDD 3	BDD 2	BDD 1	BDD 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ブ레이크データレジスタ D (BDRD) は、ブ레이크データレジスタ DH (BDRDH) とブ레이크データレジスタ DL (BDRDL) の 2 本で 1 組となっています。BDRDH と BDRDL は、それぞれ読み出し / 書き込み可能な 16 ビットのレジスタで、BDRDH はチャンネル D のブ레이크条件とするデータの上位側 (ビット 31 ~ 16) を指定し、BDRDL はデータの下位側 (ビット 15 ~ 0) を指定します。また、ブ레이크バスサイクルレジスタ (BBRD) 内の XYED ビット / XYSD ビットの設定によって、X/Y メモリに接続されているデータバスを指定することができます。XYED=1 の場合、BDRD の上位側 16 ビット (BDD31 ~ 16) は X データバスを、同様に、下位側 16 ビット (BDD15 ~ 0) は Y データバスを指定します。

BDRD の構成

		上位 16 ビット (BDD31 ~ 16)	下位 16 ビット (BDD15 ~ 0)
XYED=0	データ	データバス上位 16 ビット	データバス下位 16 ビット
XYED=1	X データ (XYSD=0 時)	X データ (XDB15 ~ 0)	
	Y データ (XYSD=1 時)		Y データ (YDB15 ~ 0)

### 6.2.16 ブ레이크データマスクレジスタ D (BDMRD)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRDH	BDMD 31	BDMD 30	BDMD 29	BDMD 28	BDMD 27	BDMD 26	BDMD 25	BDMD 24	BDMD 23	BDMD 22	BDMD 21	BDMD 20	BDMD 19	BDMD 18	BDMD 17	BDMD 16
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMRDL	BDMD 15	BDMD 14	BDMD 13	BDMD 12	BDMD 11	BDMD 10	BDMD 9	BDMD 8	BDMD 7	BDMD 6	BDMD 5	BDMD 4	BDMD 3	BDMD 2	BDMD 1	BDMD 0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ブ레이크データマスクレジスタ D (BDMRD) は、ブ레이크マスクレジスタ DH (BDMRDH) とブ레이크マスクレジスタ DL (BDMRDL) の 2 本で 1 組となっています。BDMRDH と BDMRDL は、それぞれ読み出し / 書き

込み可能な 16 ビットのレジスタです。BDMRDH は BDRDH に設定されているブレイクデータのどのビットをマスクするかを指定し、BDMRDL は、BDRDL に設定されているブレイクデータのどのビットをマスクするかを指定します。また、BBRD 内の XYED ビット / XYSD ビットによって、下記のようになります。BDMRDH と BDMRDL は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

BDMRD の構成

		上位 16 ビット (BDMD31 ~ 16)	下位 16 ビット (BDMD15 ~ 0)
XYED=0	データ	上位 16 ビットマスク可	下位 16 ビットマスク可
XYED=1	X データ (XYSD=0 時)	マスク可	
	Y データ (XYSD=1 時)		マスク可

ビット 31 ~ 0	説明
BDMDn	
0	チャンネル D のブレイクデータ BDDn をブレイク条件に含める (初期値)
1	チャンネル D のブレイクデータ BDDn をマスクし、条件に含めない

- 【注】
1.  $n = 31 \sim 0$
  2. データバスの値をブレイク条件に含めるときはオペランドサイズを指定してください。
  3. バイトサイズを指定するときで、奇数アドレスのデータをブレイク条件とするときは、BDRD、BDMRD のビット 7 ~ 0 に値を設定してください。偶数アドレスのデータをブレイク条件とするときは、ビット 15 ~ 8 に値を設定してください。使わない 8 ビットレジスタはブレイク条件に影響を与えません。

### 6.2.17 ブレイクバスサイクルレジスタ D (BBRD)

ビット	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BBRD	—	—	—	—	—	—	XYED	XYSD	CPD1	CPD0	IDD1	IDD0	RWD1	RWD0	SZD1	SZD0
初期値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W									

ブレイクバスサイクルレジスタ D (BBRD) は、読み出し / 書き込み可能な 16 ビットレジスタで、チャンネル D のブレイク条件のうち、(1) 内部バス (C バス、I バス) / X メモリバス / Y メモリバス (2) CPU サイクル / 内蔵 DMAC (DMAC、E-DMAC) サイクル (3) 命令フェッチ / データアクセス (4) 読み出し / 書き込み (5) オペランドサイズの 5 条件を設定するレジスタです。BBRD は、パワーオンリセットで H'0000 に初期化され、マニュアルリセットでは不定値になります。

- ・ ビット 15 ~ 10 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 9 : X/Y メモリバスイネーブル D (XYED)  
チャンネル D のブレイク条件を X/Y バスにするかを選択します。

## 6. ユーザブレイクコントローラ (UBC)

ビット 9	説 明
XYED	
0	チャンネル D のアドレス / データは、キャッシュバスか内部バスを条件とする (初期値)
1	チャンネル D のアドレス / データは X/Y バスを条件とする

- ・ ビット 8 : X バス / Y バスセレクト D (XYSD)  
チャンネル D のブレイク条件を X バスにするか、Y バスにするかを選択します。本ビットは、XYED ビット = 1 のときのみ有効です。

ビット 8	説 明
XYSD	
0	チャンネル D のブレイク条件を X バスとする (初期値)
1	チャンネル D のブレイク条件を Y バスとする

ビット 7 ~ 0 のビット構成は BBRA と同様です。

### 6.2.18 実行回数ブレイクレジスタ D (BETRD)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BETRD	—	—	—	—	ETRD 11	ETRD 10	ETRD 9	ETRD 8	ETRD 7	ETRD 6	ETRD 5	ETRD 4	ETRD 3	ETRD 2	ETRD 1	ETRD 0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

チャンネル D の実行回数ブレイク条件を (BRCR の ETBED ビットに設定することによって) 有効にして、ユーザブレイク割り込みを要求する前にチャンネル D のブレイク条件が発生する回数を、この 16 ビットレジスタに指定します。最大値は 212 - 1 回です。チャンネル D のブレイク条件を満たすたびに BETRD は 1 ずつデクリメントされます。BETRD が H'0001 になった後、ブレイク条件を満たすと割り込みが要求されます。

3 つ以下の命令で構成される繰り返しループ内の命令は、例外や割り込みを受け付けません (「4.6 例外要因が受け付けられない場合」参照)。したがって、BETRD は 3 つ以下の命令で構成される繰り返しループ内の命令に対して生じるブレイク条件一致に対してはデクリメントされません。

ビット 15 ~ 12 を読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

パワーオンリセットによって BETRD は H'0000 に初期化されます。

## 6.2.19 ブレークコントロールレジスタ (BRCR)

ビット:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRCRH	CMFCA	CMFPA	—	—	PCTE	PCBA	—	—	CMFCB	CMFPB	—	SEQ1	SEQ0	PCBB	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRCLR	CMFCC	CMFPC	ETBEC	—	DBEC	PCBC	—	—	CMFCD	CMFPD	ETBED	—	DBED	PCBD	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W	R	R/W	R/W	R	R

BRCR は、以下の設定を行います。

- (1) チャンネル A、B、C および D を独立 4 チャンネルまたはシーケンシャル条件の設定
- (2) 命令フェッチサイクル時の命令実行前 / 実行後の選択
- (3) C、D チャンネルでデータバスを比較条件に含めるかの選択
- (4) C、D チャンネルで実行回数ブレークを設定するかの選択
- (5) PC トレースを実行するかの選択

また、条件が一致したときセットされるフラグを持っています。BRCR は、パワーオンリセットで H'00000000 に初期化され、マニュアルリセットでは不定値になります。

・ ビット 31 : CPU コンディションマッチフラグ (CMFCA)

チャンネル A に設定したブレーク条件のうち CPU のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

ビット 31	説明
CMFCA	
0	チャンネル A の CPU サイクル条件で、ユーザブレーク割り込みは発生していない (初期値)
1	チャンネル A の CPU サイクル条件で、ユーザブレーク割り込みが発生した

・ ビット 30 : DMAC コンディションマッチフラグ (CMFPA)

チャンネル A に設定したブレーク条件のうち、内蔵 DMAC のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

## 6. ユーザブレイクコントローラ (UBC)

ビット 30 CMFPA	説 明
0	チャンネル A の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル A の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みが発生した

- ・ ビット 29、28 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 27 : PC トレースイネーブル (PCTE)  
PC トレースを実行するかを選択します。

ビット 27 PCTE	説 明
0	PC トレースを実行しない (初期値)
1	PC トレースを実行する

- ・ ビット 26 : PC ブレイクセレクト A (PCBA)  
チャンネル A の命令フェッチサイクルでのブレイクタイミングを命令実行前か命令実行後かを選択します。

ビット 26 PCBA	説 明
0	チャンネル A の命令フェッチサイクルのブレイク条件を命令実行前にする (初期値)
1	チャンネル A の命令フェッチサイクルのブレイク条件を命令実行後にする

- ・ ビット 25、24 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 23 : CPU コンディションマッチフラグ B (CMFCB)  
チャンネル B に設定したブレイク条件のうち CPU のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

ビット 23 CMFCB	説 明
0	チャンネル B の CPU サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル B の CPU サイクル条件で、ユーザブレイク割り込みが発生した

- ビット 22 : DMAC コンディションマッチフラグ B (CMFPB)

チャンネル B に設定したブレイク条件のうち、内蔵 DMAC のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

ビット 22 CMFPB	説 明
0	チャンネル B の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル B の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みが発生した

- ビット 21 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 20、19 : シーケンス条件セレクト (SEQ1、SEQ0)

チャンネル A、B、C および D の 4 チャンネルの条件を独立かシーケンシャルかを選択します。

ビット 20 SEQ1	ビット 19 SEQ0	説 明
0	0	チャンネル A、B、C および D を独立条件でコンペアする (初期値)
	1	チャンネル C D でシーケンシャル。チャンネル A および B は独立
1	0	チャンネル B C D でシーケンシャル。チャンネル A は独立
	1	チャンネル A B C D でシーケンシャル

- ビット 18 : PC ブレイクセレクト B (PCBB)

チャンネル B の命令フェッチサイクルでのブレイクタイミングを命令実行前か命令実行後かを選択します。

ビット 18 PCBB	説 明
0	チャンネル B の命令フェッチサイクルのブレイク条件を命令実行前にする (初期値)
1	チャンネル B の命令フェッチサイクルのブレイク条件を命令実行後にする

- ビット 17、16 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 15 : CPU コンディションマッチフラグ C (CMFCC)

チャンネル C に設定したブレイク条件のうち CPU のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

## 6. ユーザブレイクコントローラ ( UBC )

ビット 15	説 明
CMFCC	
0	チャンネル C の CPU サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル C の CPU サイクル条件で、ユーザブレイク割り込みが発生した

- ・ ビット 14 : DMAC コンディションマッチフラグ C ( CMFPC )  
チャンネル C に設定したブレイク条件のうち、内蔵 DMAC バスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません ( 1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください )

ビット 14	説 明
CMFPC	
0	チャンネル C の内蔵 DMAC サイクル条件で、ユーザブレイクの割り込みは発生していない (初期値)
1	チャンネル C の内蔵 DMAC サイクル条件で、ユーザブレイクの割り込みが発生した

- ・ ビット 13 : 実行回数ブレイクイネーブル C ( ETBEC )  
チャンネル C の実行回数ブレイク条件を有効にします。このビットが 1 の場合、生じたブレイク条件の数が実行回数ブレイクレジスタ ( BETRC ) で指定した実行回数と等しくなったとき、ユーザブレイク割り込みが発生します。

ビット 13	説 明
ETBEC	
0	チャンネル C の実行回数ブレイク条件を無効にする (初期値)
1	チャンネル C の実行回数ブレイク条件を有効にする

- ・ ビット 12 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 11 : データブレイクイネーブル C ( DBEC )  
チャンネル C のブレイク条件にデータベースの条件を含めるかどうかを選択します。

ビット 11	説 明
DBEC	
0	チャンネル C の条件にデータベースの条件を含めない (初期値)
1	チャンネル C の条件にデータベースの条件を含める

- ・ ビット 10 : PC ブレイクセレクト C ( PCBC )  
チャンネル C の命令フェッチサイクルでのブレイクタイミングを命令実行前か命令実行後かを選択します。

ビット 10	説 明
PCBC	
0	チャンネル C の命令フェッチサイクルのブレイク条件を命令実行前にする (初期値)
1	チャンネル C の命令フェッチサイクルのブレイク条件を命令実行後にする

・ ビット 9、8 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 7 : CPU コンディションマッチフラグ D (CMFCD)

チャンネル D に設定したブレイク条件のうち CPU のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

ビット 7	説 明
CMFCD	
0	チャンネル D の CPU サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル D の CPU サイクル条件で、ユーザブレイク割り込みが発生した

・ ビット 6 : DMAC コンディションマッチフラグ D (CMFPD)

チャンネル D に設定したブレイク条件のうち内蔵 DMAC のバスサイクルの条件について成立したとき 1 にセットされます。0 クリアは行いません (1 度セットされた後再度フラグのセットを確認する場合は、書き込みでクリアしてください)。

ビット 6	説 明
CMFPD	
0	チャンネル D の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みは発生していない (初期値)
1	チャンネル D の内蔵 DMAC サイクル条件で、ユーザブレイク割り込みが発生した

・ ビット 5 : 実行回数ブレイクイネーブル D (ETBED)

チャンネル D の実行回数ブレイク条件を有効にします。このビットが 1 の場合、生じたブレイク条件の数が実行回数ブレイクレジスタ (BETRD) で指定した実行回数と等しくなったとき、ユーザブレイク割り込みが発生します。

ビット 5	説 明
ETBED	
0	チャンネル D の実行回数ブレイク条件を無効にする (初期値)
1	チャンネル D の実行回数ブレイク条件を有効にする

## 6. ユーザブ레이크コントローラ (UBC)

- ・ ビット 4 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 3 : データブ레이크イネーブル D (DBED)  
チャンネル D のブ레이크条件にデータベースの条件を含めるかどうかを選択します。

ビット 3	説 明
DBED	
0	チャンネル D の条件にデータベースの条件を含めない (初期値)
1	チャンネル D の条件にデータベースの条件を含める

- ・ ビット 2 : PC ブ레이크セレクト D (PCBD)  
チャンネル D の命令フェッチサイクルでのブ레이크タイミングを命令実行前か命令実行後かを選択します。

ビット 2	説 明
PCBD	
0	チャンネル D の命令フェッチサイクルのブ레이크条件を命令実行前にする (初期値)
1	チャンネル D の命令フェッチサイクルのブ레이크条件を命令実行後にする

- ・ ビット 1、0 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### 6.2.20 ブランチフラグレジスタ (BRFR)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRFR	SVF	PID2	PID1	PID0	—	—	—	—	DVF	—	—	—	—	—	—	—
初期値 :	0	*	*	*	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

\* : 不定

ブランチフラグレジスタ (BRFR) は、4 本の 16 ビットの読み出し専用レジスタのセットです。BRFR には、分岐 (分岐命令、リピート、割り込みなど) した際のアドレスが BRSR、BRDR に格納されたかを示すフラグと最後に実行した命令に対するフェッチから実行までのサイクル数を示す 3 ビットポイントが格納されます。

BRFR は、PC トレースのための FIFO (先入れ先出し) キューを形成します。キューは各分岐でシフトします。

SVF、DVF はパワーオンリセットで初期化されますが、PID2 ~ PID0 は初期化されません。

- ・ ビット 15 : BRSR 確定フラグ (SVF)  
BRSR に分岐元アドレスを計算できるアドレスとポイントが格納されているかどうかを示すフラグです。分岐先アドレスの命令をフェッチしたときにセットされ、BRSR が読み出されたときにリセットされます。

ビット 15	説 明
SVF	
0	BRSR の値は無効 (初期値)
1	BRSR の値は有効

- ビット 14 ~ 12 : PID2 ~ PID0  
分岐が発生する直前に実行された命令の命令バッファの番号を示すポインタです。

ビット 14 ~ 12	説 明
PID2 ~ PID0	
偶数	PID は命令バッファの番号を示します
奇数	PID + 2 は命令バッファの番号を示します

- ビット 7 : BRDR 確定フラグ (DVF)  
BRDR に分岐先アドレスが格納されているかどうかを示すフラグです。分岐先アドレスの命令をフェッチしたときにセットされ、BRDR が読み出されたときにリセットされます。

ビット 7	説 明
DVF	
0	BRDR の値は無効 (初期値)
1	BRDR の値は有効

ブランチソースレジスタ (BRSR)、ブランチディスティネーションレジスタ (BRDR)、ブランチフラグレジスタ (BRFR) を使った PC トレースの方法は、PC トレースを参照してください。

- ビット 11 ~ 8、6 ~ 0 : 予約ビット  
読み出すと常に 0 が読み出されます。

### 6.2.21 ブランチソースレジスタ (BRSR)

ビット :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRSRH	BSA 31	BSA 30	BSA 29	BSA 28	BSA 27	BSA 26	BSA 25	BSA 24	BSA 23	BSA 22	BSA 21	BSA 20	BSA 19	BSA 18	BSA 17	BSA 16
初期値 :	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRSRL	BSA 15	BSA 14	BSA 13	BSA 12	BSA 11	BSA 10	BSA 9	BSA 8	BSA 7	BSA 6	BSA 5	BSA 4	BSA 3	BSA 2	BSA 1	BSA 0
初期値 :	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

\* : 不定

## 6. ユーザブレイクコントローラ (UBC)

---

ブランチソースレジスタ (BRSR) は、分岐前に最後にフェッチしたアドレスを格納する 4 本の 32 ビット読み出し専用レジスタのセットです。これらの値は、PC トレース時に分岐前、最後に実行した命令のアドレスを計算するために使用します。BRSR は、PC トレースのための FIFO (先入れ先出し) キューを形成します。キューは各分岐でシフトします。

BRSR はリセットで初期化されません。

### 6.2.22 ブランチディスティネーションレジスタ (BRDR)

ビット :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRDRH	BDA 31	BDA 30	BDA 29	BDA 28	BDA 27	BDA 26	BDA 25	BDA 24	BDA 23	BDA 22	BDA 21	BDA 20	BDA 19	BDA 18	BDA 17	BDA 16
初期値 :	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRDRL	BDA 15	BDA 14	BDA 13	BDA 12	BDA 11	BDA 10	BDA 9	BDA 8	BDA 7	BDA 6	BDA 5	BDA 4	BDA 3	BDA 2	BDA 1	BDA 0
初期値 :	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

\* : 不定

ブランチディスティネーションレジスタ (BRDR) は、4 本の 32 ビット読み出し専用レジスタのセットです。BRDR は、PC トレース時に使用する分岐先フェッチアドレスを格納します。BRDR は、PC トレースのための FIFO (先入れ先出し) キューを形成します。キューは各分岐でシフトします。

BRDR はリセットで初期化されません。

## 6.3 動作説明

### 6.3.1 ユーザブレイク動作の流れ

ブレイク条件の設定からユーザブレイク割り込み例外処理までの流れを以下に示します。

1. ブレイクするアドレスをブレイクアドレスレジスタ (BARA、BARB、BARC、BARD) に、マスクするビットをブレイクアドレスマスクレジスタ (BAMRA、BAMRB、BAMRC、BAMRD) に、ブレイクするデータをブレイクデータレジスタ (BDRC、BDRD) に、マスクするビットをブレイクデータマスクレジスタ (BDMRC、BDMRD) に設定します。

ブレイクするバス条件をブレイクバスサイクルレジスタ (BBRA、BBRB、BBRC、BBRD) に設定します。BBRA、BBRB、BBRC、BBRD の CPU サイクル / 内蔵 DMAC サイクルセレクト、命令フェッチ / データアクセスセレクト、リード / ライトセレクトの 3 組すべてにそれぞれ設定します。このうち 1 組でも 00 をセットしたチャンネルではユーザブレイク割り込みは発生しません。

BRCR のレジスタの各ビットにそれぞれ条件を設定します。

2. UBC は設定された条件を満たすと割り込みコントローラ (INTC) にユーザブレイク割り込みを要求します。また、一致の発生した条件について、それぞれのチャンネルの CPU コンディションマッチフラグ (CMFCA、CMFCB、CMFCC、CMFCD)、DMAC コンディションマッチフラグ (CMFPA、CMFPB、CMFPC、CMFPD) がセットされます。
3. INTC は、ユーザブレイク割り込みの優先順位の判定をします。ユーザブレイク割り込みは優先レベル 15 なので、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) がレベル 14 以下のとき、受け付けられます。I3 ~ I0 ビットがレベル 15 のとき、ユーザブレイク割り込みは受け付けられませんが、受け付けられるようになるまで保留されます。優先順位判定の詳細については「5. 割り込みコントローラ (INTC)」を参照してください。
4. 優先順位の判定結果、ユーザブレイク割り込みが受け付けられると CPU はユーザブレイク割り込み例外処理を開始します。
5. 設定条件の一致 / 不一致はそれぞれのコンディションマッチフラグ (CMFCA、CMFPA、CMFCB、CMFPB、CMFCC、CMFPC、CMFCD、CMFPD) により知ることができます。このフラグは、設定条件の一致によりセットされますが、リセットはされません。したがって、再度同じフラグのセットを確認する場合は、0 を書き込んでください。  
チャンネル C、チャンネル D に対して実行回数ブレイクを指定した場合、実行回数が BETRC、BETRD で指定した実行回数に一致すると CMFCC、CMFPC、CMFCD、CMFPD がセットされます。

### 6.3.2 命令フェッチサイクルのブレーク

1. ブレークバスサイクルレジスタ (BBRA、BBRB、BBRC、BBRD) に CPU / 命令フェッチ / リード / ワードの設定をすると CPU の命令フェッチサイクルをブレーク条件にできます。このとき、対象命令の実行前または実行後のどちらでブレークするか、それぞれのチャンネルについてブレークコントロールレジスタ (BRCR) の PCBA/PCBB/PCBC/PCBD ビットで選択できます。
2. 実行前条件でブレークの対象となる命令は、命令がフェッチされて実行することが確定したときにブレークします。したがって、オーバランフェッチされた命令 (分岐時や、割り込み遷移時にフェッチされたが実行されない命令) はブレークの対象となりません。また、遅延分岐命令の遅延スロットや、LCD などの割り込み禁止命令の次命令に設定した場合は、次に初めて割り込みを受け付ける命令の実行前に割り込みを発生します。
3. 実行後条件ではブレーク条件に設定した命令が実行され、次の命令の実行前に割り込みを発生します。実行前条件のときと同様にオーバランフェッチ命令は対象となりません。また、遅延分岐命令や、LCD 命令などの割り込み禁止命令に設定した場合は次に初めて割り込みを受け付ける命令の実行前に割り込みを発生します。
4. 命令フェッチサイクルをチャンネル C、チャンネル D に設定した場合、ブレークデータレジスタ C (BDRC)、ブレークデータレジスタ D (BDRD) は無視されます。したがって命令フェッチサイクルのブレークにはブレークデータを設定する必要はありません。
5. 命令フェッチサイクルを設定した場合、ブレークはその命令の置かれている先頭番地のアドレスを設定してください。それ以外のアドレスを設定してもブレークは発生しません。32 ビット命令の下位ワードのアドレスを設定した場合、ブレークは発生しません。

### 6.3.3 データアクセスサイクルによるブレーク

1. CPU データアクセスのブレーク対象となるメモリサイクルは、命令によるメモリサイクルと例外処理時のスタッキング、ベクタリードがあります。外部ベクタ割り込みのベクタフェッチサイクルは対象外です。また、シンクロナス DRAM のパーストライト時、シングルリード時のダミーアクセスサイクルは、ブレークの対象となりません。
2. データアクセスサイクルのアドレスとオペランドサイズの比較条件は表 6.2 に示す関係があります。

表 6.2 データアクセスサイクルのアドレスとオペランドサイズの比較条件

アクセスサイズ	比較するアドレス
ロングワード	ブレークアドレスレジスタの 31 ~ 2 ビットとアドレスバス 31 ~ 2 ビットを比較
ワード	ブレークアドレスレジスタの 31 ~ 1 ビットとアドレスバス 31 ~ 1 ビットを比較
バイト	ブレークアドレスレジスタの 31 ~ 0 ビットとアドレスバス 31 ~ 0 ビットを比較

したがって、例えばサイズ条件を指定しないでアドレス H'00001003 番地に設定した場合、ブレイク条件の成立するバスサイクルは次のようになります (ただし、他の条件は一致しているものとします)。

H'00001000 番地にロングワードでアクセス

H'00001002 番地にワードでアクセス

H'00001003 番地にバイトでアクセス

### 3. チャンネル C でデータ値をブレイク条件に含める場合

データ値をブレイク条件に含める場合は、ブレイクバスサイクルレジスタ (BBRC) のオペランドサイズをロングワード、ワード、バイトのいずれかに指定してください。データ値をブレイク条件に含めた場合は、アドレス条件の一致かつデータ条件の一致でブレイク割り込みが発生します。

このとき、バイトデータを指定するときはブレイクデータレジスタ C (BDRC) およびブレイクデータマスクレジスタ C (BDMRC) のビット 15 ~ 8、ビット 7 ~ 0 の 2 バイトは同じデータを設定してください。

なお、ワードおよびバイトに設定したときは BDRC と BDMRC のビット 31 ~ 16 は無視されます。

また、チャンネル D でデータ値をブレイク条件に含める場合も同様の条件です。

## 6.3.4 回避するプログラムカウンタ (PC) の値

### 1. 命令フェッチ (命令実行前) をブレイク条件に設定した場合

ユーザブレイク割り込み例外処理でスタックに回避されるプログラムカウンタ (PC) の値は、ブレイク条件が一致した命令のアドレスです。このとき、フェッチされた命令は実行されず、その手前でユーザブレイク割り込みが発生します。ただし、割り込み禁止命令の次命令に設定したときは、次の割り込みを受け付ける命令の実行前でブレイクするので、回避される PC はブレイクしたアドレスとなります。

### 2. 命令フェッチ (命令実行後) をブレイク条件に設定した場合

ユーザブレイク割り込み例外処理でスタックに回避される PC の値は、ブレイク条件が一致した命令の次に実行される命令のアドレスです。このとき、フェッチされた命令は実行され、その次の命令の実行前にユーザブレイク割り込みが発生します。ただし、割り込み禁止命令に設定したときは、次の割り込みを受け付ける命令の実行前でブレイクするので、回避される PC はブレイクしたアドレスとなります。

### 3. データアクセス (CPU / 内蔵 DMAC) をブレイク条件に設定した場合

ユーザブレイク例外処理が起動した時点における実行済み命令の次命令の先頭アドレスが回避されます。データアクセス (CPU / 内蔵 DMAC) をブレイク条件に設定した場合、ブレイクがかかる場所は特定することはできません。ブレイクするデータアクセスが発生した付近でフェッチしようとしていた命令がブレイクされます。

### 6.3.5 XメモリまたはYメモリバスサイクルでのブレイク

XバスサイクルまたはYバスサイクルに対するブレイク条件は、チャンネルC、チャンネルDでのみ指定可能です。BBRCのXYECおよびBBRDのXYEDを1にセットにした場合、XメモリまたはYメモリバス上のブレイクアドレスおよびブレイクデータが選択されます。BBRCのXYSCおよびBBRDのXYSDを指定することによってXメモリバスまたはYメモリバスのどちらかを選択する必要があります。ブレイク条件にはXメモリまたはYメモリを同時に含めることはできません。ブレイク条件はBBRC、BBRDでCPUバスマスタ、データアクセスサイクル、リードまたはライトアクセス、オペランドサイズをワード、またはオペランドサイズを指定しない設定にすることによって、XメモリバスサイクルまたはYメモリバスサイクルに適用されます。

ブレイク条件としてXメモリアドレスを選択するときは、BARCとBAMRCもしくはBARDとBAMRDの上位16ビットにXメモリアドレスを指定し、Yメモリアドレスを選択するときは、BARCとBAMRCもしくはBARDとBAMRDの下部16ビットにXメモリアドレスを指定してください。BDRCとBDMRCもしくはBDRDとBDMRDに対するXメモリデータまたはYメモリデータの指定は、同様の方法で行います。

### 6.3.6 シーケンシャルブレイク

#### 1. チャンネルC チャンネルD

BBRCのSEQ1を0、SEQ0を1にセットすると、チャンネルC チャンネルDの順に条件が一致したときシーケンシャルブレイクが発行され、各チャンネルのBBRCの条件一致フラグは1にセットされます。

チャンネルCとチャンネルDのブレイク条件が同時に一致し、以前にチャンネルCで条件一致がなかった場合、チャンネルCのみ条件が一致したものとみなされます。また、すでにチャンネルCで条件一致があった場合は、チャンネルDが条件一致したものとしてブレイクを発行します。

#### 2. チャンネルB チャンネルC チャンネルD

BBRCのSEQ1を1、SEQ0を0にセットすると、チャンネルB チャンネルC チャンネルDの順に条件が一致したときシーケンシャルブレイクが発行され、各チャンネルのBBRCの条件一致フラグは1にセットされます。

チャンネルBとチャンネルCのブレイク条件が同時に一致し、以前にチャンネルBで条件一致がなかった場合、チャンネルBのみ条件が一致したものとみなされます。また、すでにチャンネルBで条件一致があった場合は、チャンネルCが条件一致したものとみなされます。

チャンネルCとチャンネルDのブレイク条件が同時に一致し、以前にチャンネルCで条件一致がなかった場合、チャンネルCのみ条件が一致したものとみなされます。また、すでにチャンネルCで条件一致があった場合は、チャンネルDが条件一致したものとしてブレイクを発行します。

#### 3. チャンネルA チャンネルB チャンネルC チャンネルD

BBRCのSEQ1を1、SEQ0を1にセットすると、チャンネルA チャンネルB チャンネルC チャンネルDの順に条件が一致したときシーケンシャルブレイクが発行され、各チャンネルのBBRCの条件一致フラグは1にセットされます。

チャンネル A とチャンネル B のブレーク条件が同時に一致し、以前にチャンネル A で条件一致がなかった場合、チャンネル A のみ条件が一致したものとみなされます。また、すでにチャンネル A で条件一致があった場合は、チャンネル B が条件一致したものとみなされます。

チャンネル B とチャンネル C のブレーク条件が同時に一致し、以前にチャンネル B で条件一致がなかった場合、チャンネル B のみ条件が一致したものとみなされます。また、すでにチャンネル B で条件一致があった場合は、チャンネル C が条件一致したものと見なされます。

チャンネル C とチャンネル D のブレーク条件が同時に一致し、以前にチャンネル C で条件一致がなかった場合、チャンネル C のみ条件が一致したものとみなされます。また、すでにチャンネル C で条件一致があった場合は、チャンネル D が条件一致したものと見なされます。

ただし、シーケンシャル条件に含まれるチャンネルのうち、隣り合うチャンネルの最初のブレーク条件となるチャンネルのバスサイクルの条件が、実行前ブレーク (BRCR の PCB ビットが 0) および (ブレークバスサイクルレジスタにより) 命令フェッチとして指定されるとき、2 つのチャンネルに対応するバスサイクル条件が一致するとき、ブレークが発行され BRCR の条件一致フラグが 1 にセットされます。

シーケンシャルブレーク指定では、X バス、または Y バスを選択でき、実行回数ブレーク条件も指定することができます。たとえば、チャンネル C およびチャンネル D の実行回数ブレークを指定すると、チャンネル C の BETRC に設定された回数の条件が発生したのち、チャンネル D の BETRD に設定された回数の条件一致が発生したときユーザブレーク割り込みが発生します。

## 6.3.7 PC トレース

1. PC トレースは、BRDR 内の PC トレースイネーブルビット (PCTE) を 1 に設定することでトレースを開始します。分岐 (分岐命令、リピート、割り込み) が発生すると分岐元アドレスを計算できるアドレスと分岐先アドレスをブランチソースレジスタ (BRSR) とブランチディスティネーションレジスタ (BRDR) に格納します。BRDR には分岐先命令のフェッチアドレスが格納されます。BRSR には、分岐が生じる前の最後の命令フェッチアドレスが格納されます。ブランチフラグレジスタ (BRFR) は、分岐の直前に実行された命令の関係を示すポインタを格納します。
2. BRSR に格納されているアドレスと BRFR に格納されているポインタから、分岐が発生する直前に実行されていた命令のアドレスを計算できます。BRSR に格納されているアドレスを BSA、BRFR に格納されているポインタを PID、分岐発生前のアドレスを IA とすると、計算式は、 $IA=BSA-2 \times PID$  となります。分岐先の命令が実行される前に割り込み (分岐) が生じた場合には注意が必要です。図 6.2 の場合、分岐の直前に実行された命令 "Exec" のアドレスは、 $IA=BSA - 2 \times PID$  によって計算されます。しかしながら、分岐 "branch" が遅延スロットを持つ遅延分岐命令で、分岐先が  $4n+2$  のアドレスである場合、分岐命令によって指定される分岐先アドレス "Dest" はそのまま BRSR に格納されます。したがって、計算式  $IA=BSA - 2 \times PID$  はこの場合には適用されないため、この PID は無効です。BSA が  $4n+2$  境界であるのはこの場合だけで、表 6.3 のように分類されます。

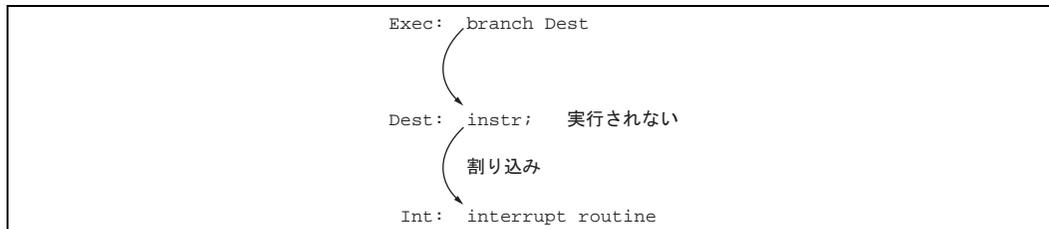


図 6.2 分岐命令を実行する前に割り込みが生じる場合

表 6.3 分岐先命令実行前の例外処理で格納される BSA 値

branch	分岐先 (Dest)	BSA	BRSR と BRFR によって計算できる分岐元アドレス
遅延	4n	4n	$Exec=IA=BSA - 2 \times PID$
	4n+2	4n+2	Dest=BSA
遅延なし	4n or 4n+2	4n	$Exec=IA=BSA - 2 \times PID$

PID 値が奇数の場合、+2 された値が命令バッファを指しますが、表中の式はそれを考慮してあります。したがって、BRSR に格納されている BSA、BRFR に格納されている PID の値によって計算できます。

3. 分岐発生前のアドレス、IA がどこを指すのかは、分岐の種類によって異なります。
  - (a) 分岐命令：分岐命令のアドレス
  - (b) 繰り返しループ：繰り返しループの最後から 2 番目の命令

```

Repeat_Start: inst ( 1 ) ;      BRDR
                                     inst ( 2 ) ;
                                     :
                                     inst ( n-1 ) ;    BRSR、BRFR からアドレスを算出
Repeat_End:   inst ( n ) ;

```

(c) 割り込み：割り込みの直前に実行した命令

BRDR には割り込みルーチンの先頭命令アドレスが格納されます。

3 命令以下の繰り返しループでは、命令フェッチサイクルが生じません。分岐先アドレスがわからないので PC トレースは無効です。

4. BRSR、BRDR、BRFR は 4 組のキュー構造になっています。PC トレースで格納されたアドレスを読み出す場合、読み出しはキューの先頭から行われます。読み出しは、BRFR、BRSR、BRDR の順にしてください。BRDR 読み出し後、キューは一つシフトします。BRSR、BRDR の読み出し時には、ロングワードアクセスを行ってください。

### 6.3.8 使用例

#### (1) CPU 命令フェッチサイクルへのブレイク条件設定

(A) レジスタ設定： BARA=H'00000404 / BAMRA=H'00000000 / BBRA=H'0054  
 BARB=H'00003080 / BAMRB=H'0000007F / BBRB=H'0054  
 BARC=H'00008010 / BAMRC=H'00000006 / BBRC=H'0054  
 BDRC=H'00000000 / BDMRC=H'00000000  
 BARD=H'0000FF04 / BAMRD=H'00000000 / BBRD=H'0054  
 BDRD=H'00000000 / BDMRD=H'00000000  
 BRRCR=H'04000400

設定された条件： 全チャンネル独立モード

Ach： アドレス H'00000404 / アドレスマスク H'00000000  
 バスサイクル CPU、命令フェッチ (命令実行後)  
 リード (オペランドサイズは条件に含まれない)

Bch： アドレス H'00003080 / アドレスマスク H'0000007F  
 バスサイクル CPU、命令フェッチ (命令実行前)  
 リード (オペランドサイズは条件に含まれない)

Cch： アドレス H'00008010 / アドレスマスク H'00000006  
 データ H'00000000 / アドレスマスク H'00000000

## 6. ユーザブレイクコントローラ (UBC)

---

バスサイクル CPU、命令フェッチ (命令実行後)  
リード (オペランドサイズは条件に含まれない)

Dch : アドレス H'0000FF04 / アドレスマスク H'00000000

データ H'00000000 / アドレスマスク H'00000000

バスサイクル CPU、命令フェッチ (命令実行前)  
リード (オペランドサイズは条件に含まれない)

アドレス H'00000404 番地の命令の実行後、アドレス H'00003080 ~ H'000030FF 番地の命令実行前、アドレス H'00008010 ~ H'00008016 番地の命令の実行後、もしくは、アドレス H'0000FF04 番地の命令実行前にユーザブレイクが発生します。

(B) レジスタ設定 : BARA=H'00027128 / BAMRA=H'00000000 / BBRA=H'005A  
BARB=H'00031415 / BAMRB=H'00000000 / BBRB=H'0054  
BARC=H'00037226 / BAMRC=H'00000000 / BBRC=H'0056  
BDRC=H'00000000 / BDMRC=H'00000000  
BARD=H'0003722E / BAMRD=H'00000000 / BBRD=H'0056  
BDRD=H'00000000 / BDMRD=H'00000000  
BRCCR=H'00080000

設定された条件 : チャンネル A、B 独立、チャンネル C チャンネル D シーケンシャル  
モード

Ach : アドレス H'00027128 / アドレスマスク H'00000000

バスサイクル CPU、命令フェッチ (命令実行前)

ライト、ワード

Bch : アドレス H'00031415 / アドレスマスク H'00000000

バスサイクル CPU、命令フェッチ (命令実行前)

リード (オペランドサイズは条件に含まれない)

Cch : アドレス H'00037226 / アドレスマスク H'00000000

データ H'00000000 / データマスク H'00000000

バスサイクル CPU、命令フェッチ (命令実行前)

リード、ワード

Dch : アドレス H'0003722E / アドレスマスク H'00000000

データ H'00000000 / データマスク H'00000000

バスサイクル CPU、命令フェッチ (命令実行前)

リード、ワード

チャンネル A は、命令フェッチはライトサイクルではないので、ユーザブレイク割り込みは発生しません。

チャンネル B は、命令フェッチは偶数アドレスに対して行われるので、ユーザブレイク割り込みは発生しません。

アドレス H'00037226 にある命令を実行し、その後アドレス H'0003722E にある命令の実行前にチャンネル C、D のシーケンシャル条件一致により、ユーザブレイクが発生します。

(C) レジスタ設定： BBRA=H'0000  
 BBRB=H'0000  
 BARC=H'00037226 / BAMRC=H'00000000 / BBRC=H'005A  
 BDRC=H'00000000 / BDMRC=H'00000000  
 BARD=H'0003722E / BAMRD=H'00000000 / BBRD=H'0056  
 BDRD=H'00000000 / BDMRD=H'00000000  
 BRRCR=H'00080000

設定された条件： チャンネル A、B 独立、チャンネル C チャンネル D シーケンシャル  
 モード

Ach： 未使用

Bch： 未使用

Cch： アドレス H'00037226 / アドレスマスク H'00000000  
 データ H'00000000 / データマスク H'00000000  
 バスサイクル CPU、命令フェッチ (命令実行前)  
 ライト、ワード

Dch： アドレス H'0003722E / アドレスマスク H'00000000  
 データ H'00000000 / データマスク H'00000000  
 バスサイクル CPU、命令フェッチ (命令実行前)  
 リード、ワード

チャンネル C のブレイクがライトサイクルなので条件が一致せず、シーケンシャル条件が成立しないので、ユーザブレイク割り込みは発生しません。

(D) レジスタ設定： BBRA=H'0000  
 BARB=H'00000500 / BAMRB=H'00000000 / BBRB=H'0057  
 BARC=H'00000A00 / BAMRC=H'00000000 / BBRC=H'0057  
 BDRC=H'00000000 / BDMRC=H'00000000  
 BARD=H'00001000 / BAMRD=H'00000000 / BBRD=H'0057  
 BDRD=H'00000000 / BDMRD=H'00000000  
 BRRCR=H'00102020 / BETRC=H'0005 / BETRD=H'000A

Ach： 未使用

Bch： アドレス H'00000500 / アドレスマスク H'00000000

## 6. ユーザブレイクコントローラ (UBC)

---

データ H'00000000 / データマスク H'00000000  
バスサイクル CPU、命令フェッチ (命令実行前)  
リード、ワード

Cch : アドレス H'00000A00 / アドレスマスク H'00000000  
データ H'00000000 / データマスク H'00000000  
バスサイクル CPU、命令フェッチ (命令実行前)  
リード、ワード  
実行回数ブレイクイネーブル (5 回)

Dch : アドレス H'00001000 / アドレスマスク H'00000000  
データ H'00000000 / データマスク H'00000000  
バスサイクル CPU、命令フェッチ (命令実行前)  
リード、ワード  
実行回数ブレイクイネーブル (10 回)

アドレス H'00000500 にある命令を実行し、アドレス H'00000A00 にある命令を 5 回実行した後、アドレス H'00001000 にある命令を 9 回実行し、10 回目の命令実行前にユーザブレイク割り込みが発生します。

### (2) CPU データアクセスサイクルへのブレイク条件設定

(A) レジスタ設定 : BARA=H'00123456 / BAMRA=H'00000000 / BBRA=H'0064  
BARB=H'01000000 / BAMRB=H'00000000 / BBRB=H'0066  
BARC=H'000ABCDE / BAMRC=H'000000FF / BBRC=H'006A  
BDRC=H'0000A512 / BDMRC=H'00000000  
BARD=H'1001E000 / BAMRD=H'FFFF0000 / BBRD=H'036A  
BDRD=H'00004567 / BDMRD=H'00000000  
BRCR=H'00000808

設定された条件 : 全チャンネル独立モード

Ach : アドレス H'00123456 / アドレスマスク H'00000000  
バスサイクル CPU、データアクセス  
リード (オペランドサイズは条件に含まれない)

Bch : アドレス H'01000000 / アドレスマスク H'00000000  
バスサイクル CPU、データアクセス  
リード、ワード

Cch : アドレス H'000ABCDE / アドレスマスク H'00000000  
データ H'0000A512 / データマスク H'00000000  
バスサイクル CPU、データアクセス  
ライト、ワード

Dch : Y アドレス H'1001E000 / アドレスマスク H'FFFF0000  
 データ H'00004567 / データマスク H'00000000  
 バスサイクル CPU、データアクセス  
 ライト、ワード

チャンネル A は、H'00123456 番地にロングワードで読み出し / H'00123456 番地にワードで読み出し / H'00123456 番地にバイトで読み出したときにユーザブレイク割り込みが発生します。

チャンネル B は、H'01000000 番地にワードで読み出ししたときにユーザブレイク割り込みが発生します。

チャンネル C は、H'000ABC00 ~ H'000ABCFE 番地にワードで H'A512 を書き込んだときにユーザブレイク割り込みが発生します。

チャンネル D は、Y メモリ空間のアドレス H'1001E000 番地にワードで H'4567 を書き込んだときにユーザブレイク割り込みが発生します。

### (3) DMAC データアクセスサイクルへのブレイク条件設定

レジスタ設定 :     BARA=H'00314156 / BAMRA=H'00000000 / BBRA=H'0094  
                       BBRB=H'0000  
                       BBRC=H'0000  
                       BARD=H'00055555 / BAMRD=H'00000000 / BBRD=H'00A9  
                       BDRD=H'00007878 / BDMRD=H'00000F0F  
                       BRCR=H'00000008

設定された条件 :   全チャンネル独立モード

Ach : アドレス H'00314156 / アドレスマスク H'00000000  
       バスサイクル DMAC、命令フェッチ  
                       リード (オペランドサイズは条件に含まれない)  
 Bch : 未使用  
 Cch : 未使用  
 Dch : アドレス H'00055555 / アドレスマスク H'00000000  
       データ H'00007878 / データマスク H'00000F0F  
       バスサイクル DMAC、データアクセス  
                       ライト、バイト

チャンネル A は、DMAC サイクルでは命令フェッチは行われないので、ユーザブレイク割り込みは発生しません。

チャンネル D は、DMAC が H'00055555 番地にバイトで H'7\* (\* は don't care) を書き込んだときにユーザブレイク割り込みが発生します。

### 6.3.9 注意事項

1. UBC のレジスタは CPU でのみ読み出し / 書き込みが可能です。
2. シーケンシャルブレイクの指定においての注意事項は次のとおりです。
  - ・CPU はパイプライン構成なので、命令フェッチサイクルとメモリサイクルの順序はパイプラインによって決定されます。したがって、バスサイクルの順序においてチャンネル条件の一致がシーケンシャル条件を満たすとブレイクが発生します。
  - ・シーケンシャル条件に含まれるチャンネルのうち、隣り合うチャンネルの始めのブレイク条件となるチャンネルのバスサイクルの条件が、実行前ブレイク (BRCR の PCB ビットが 0) および (ブレイクバスサイクルレジスタにより) 命令フェッチとして指定されるとき以下の注意が必要です。2 つのチャンネルに対応するバスサイクル条件が同時に一致するとき、ブレイクが発行され BRCR の条件一致フラグが 1 にセットされます。
3. レジスタ設定の変更は、通常 3 サイクルで書き込んだ値が有効となります。内蔵メモリの命令フェッチの際、2 命令が同時に取り込まれます。このとき、2 命令のうち、後半命令のフェッチをブレイク条件に設定していた場合、前半命令フェッチ直後に UBC の各レジスタを書き換えてブレイク条件を変更しても、後半命令の手前でユーザブレイク割り込みが発生します。確実に設定が変更されるタイミングをとるためには最後に書き込んだレジスタの値をダミーで読み出してください。それ以降は変更した設定は有効です。
4. 命令フェッチ条件が一致してユーザブレイク割り込みが発生し、例外処理サービスルーチンの実行中に、再度 UBC で条件が一致した場合、そのブレイクは SR の割り込み要求マスクの値が 14 以下になったときに例外処理が発生します。したがってアドレスをマスクして、命令フェッチ / 実行後条件と設定してステップ実行をさせる場合は、UBC の例外処理サービスルーチンの中では、アドレスの一致が起こらないようにしてください。
5. ブレイク条件として、リピート命令を含む繰り返し実行中の命令を指定する際には、次のことに注意してください。

繰り返しループ中の命令がブレイク条件として指定される場合、

  - ・3 命令以下からなる繰り返しループの実行中には、ブレイクは発生しません。
  - ・実行回数ブレイクがセットされるとき、メモリからの命令フェッチは 3 命令以下からなる繰り返しループの実行中には発生しません。したがって、実行回数レジスタ (BETRC、BETRD) の値は減少しません。
6. PC トレースレジスタ (BRFR、BRSR および BRDR) を読み出した直後に分岐命令を実行しないでください。
7. 実行回数ブレイクがセットされているとき CPU と DMAC のバスサイクルがブレイク条件に設定されている場合、CPU と DMAC の条件一致が同時に起こっても BETR は 1 回しかデクリメントされません。

8. UBC と H-UDI はエミュレータによって使用されています。そのため、UBC と H-UDI に関しては、エミュレータと実チップで動作が異なる場合があります。UBC と H-UDI をユーザシステムで使用しない場合は、レジスタ設定を行わないでください。



---

## 7. バスステートコントローラ (BSC)

---

### 7.1 概要

バスステートコントローラ (BSC) は、アドレス空間を管理し、5 つの空間で最適なメモリアクセスができるよう、制御信号を出力します。これにより、DRAM、SDRAM などの各種メモリと周辺 LSI を直接接続することができます。

#### 7.1.1 特長

BSC には、次の特長があります。

- アドレス空間を 5 つに分割して管理
  - CS0 空間から CS4 空間までの空間はそれぞれ最大リニア 32M バイト
  - 空間ごとに、DRAM、シンクロナス DRAM、バースト ROM などメモリの種類を指定
  - 空間ごとに、バス幅 (8 ビット、16 ビット、または 32 ビット) を選択可能
  - 空間ごとにウェイトステートの挿入を制御
  - 空間に対応した制御信号を出力
- キャッシュ
  - キャッシュ領域とキャッシュスルー領域とをアクセスアドレスで選択
  - キャッシュアクセスの場合、キャッシュミス時の読み出しはキャッシュフィルのため 4 バイト単位で 16 バイトを連続して行う。書き込みはライトスルー方式 / ライトバック方式を選択可能
  - キャッシュスルーアクセスは、アクセスサイズに従ってアクセス
- リフレッシュ機能
  - $\overline{\text{CAS}}$  ピフオ  $\overline{\text{RAS}}$  リフレッシュ (オートリフレッシュ) とセルフリフレッシュ
  - リフレッシュ用カウンタ、クロックセレクトにより、リフレッシュ間隔を設定
  - リフレッシュ回数設定 (1、2、4、6、8) による集中リフレッシュ
- DRAM 直結インタフェース
  - ロウアドレス / カラムアドレスのマルチプレクス出力
  - 読み出し時のバースト転送、連続アクセスに対する高速ページモード
  - $\overline{\text{RAS}}$  プリチャージタイム確保用 TP サイクル発生
  - EDO モード
- シンクロナス DRAM 直結インタフェース
  - ロウアドレス / カラムアドレスのマルチプレクス出力
  - バーストリード / シングルライトモードまたはバーストリード / バーストライトモードを選択可能
  - バンクアクティブモード

## 7. バスステートコントローラ (BSC)

---

- バスアービトレーション
  - すべての資源を他の CPU と共有し、外部からのバス解放要求を受け、バス使用許可を出力
- リフレッシュ用カウンタをインターバルタイマとして使用可能
  - コンペアマッチで割り込み要求発生 (CMI 割り込み要求信号)

7.1.2 ブロック図

BSCのブロック図を図 7.1 に示します。

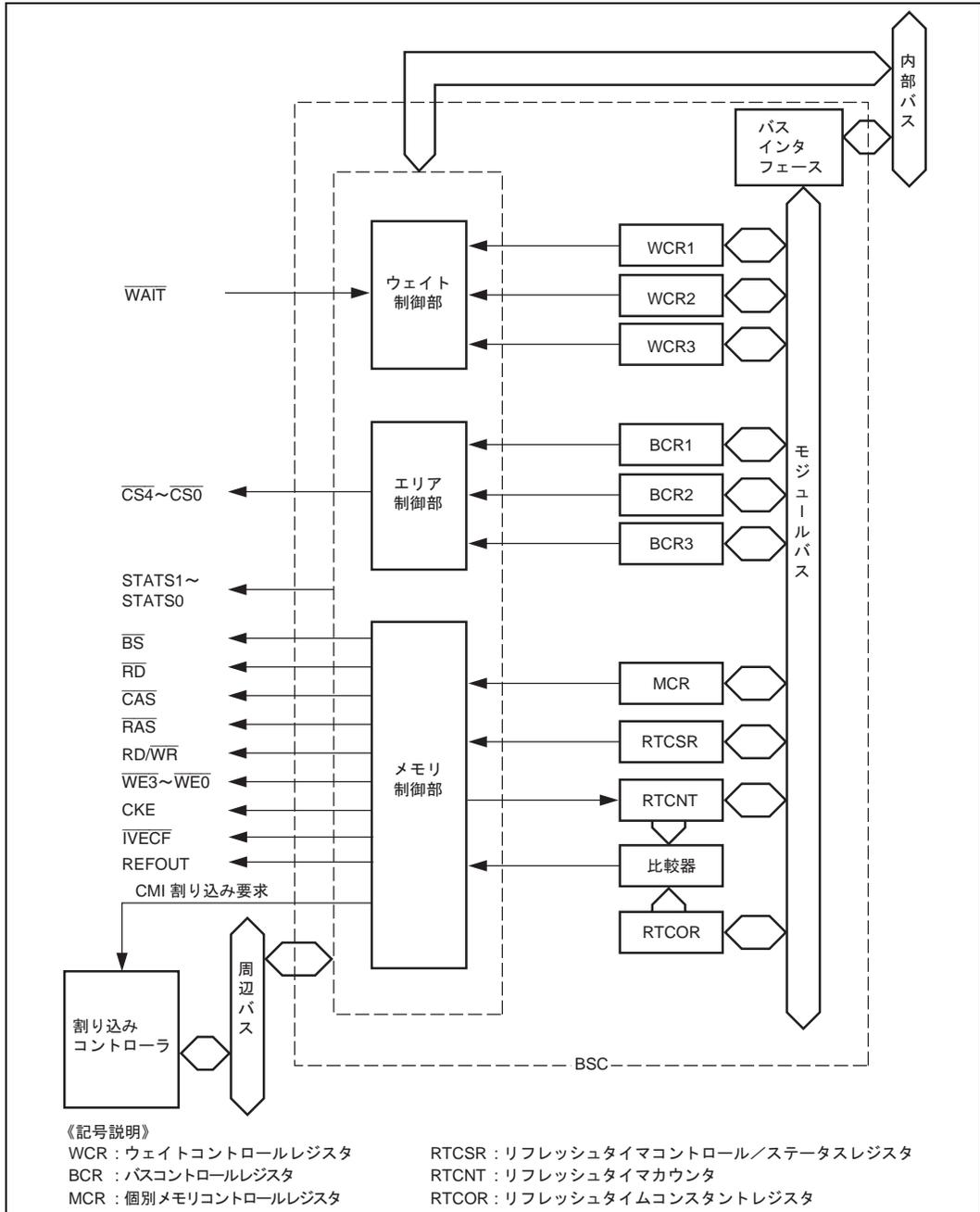


図 7.1 BSC のブロック図

## 7. バスステートコントローラ (BSC)

### 7.1.3 端子構成

BSCの端子構成を表7.1に示します。

表 7.1 端子構成 (1)

端子名	入出力	バス開放時	機能
A24 ~ A0	出力	Hi-Z	アドレスバス 25 ビットで 32M バイトのメモリ空間の指定が可能。
D31 ~ D0	入出力	Hi-Z	32 ビットのデータバス 16 ビット幅のエリアにリード/ライトする場合には D15 ~ D0、8 ビット幅のエリアにリード/ライトする場合には D7 ~ D0 をそれぞれ使用します。8 ビットアクセスで、32 ビット幅のエリアにリード/ライトする場合には 32 ビットバスのアドレス下位ビットによって定まるバイト位置を介してデータの入出力を行います。
BS	出力	Hi-Z	バスサイクルの開始またはモニタを表示する信号 通常空間 (DRAM、シンクロナス DRAM を除いたデバイスインタフェース) のときに、アドレス出力と同時に 1 クロックサイクルだけアサートする信号。本信号によってバスサイクルの開始を判定できます。
CS0 ~ CS4	出力	Hi-Z	チップセレクト CS3 空間が DRAM 空間のときは $\overline{CS3}$ はアサートしません。
RD/ $\overline{WR}$	出力	Hi-Z	リード/ライト信号 アクセスサイクルの方向 (リード/ライト) を示す信号。 DRAM, シンクロナス DRAM 接続時は、 $\overline{WE}$ 端子に接続
RAS	出力	Hi-Z	DRAM・シンクロナス DRAM 接続時: RAS 端子に接続
CAS/ $\overline{OE}$	出力	Hi-Z	DRAM 使用時: 未使用 EDO 使用時: $\overline{OE}$ 端子に接続 シンクロナス DRAM 使用時: $\overline{CAS}$ 端子に接続
$\overline{RD}$	出力	Hi-Z	リードパルス信号 (リードデータ出力許可信号) 通常、デバイスの $\overline{OE}$ 端子に接続、外部にデータバッファを設ける場合、リードサイクルのデータ出力は本信号が L レベルの時のみ可能。
WAIT	入力	don't care	ハードウェアウェイト入力
BRLS	入力	入力	バス解放要求入力
BGR	出力	出力	バス使用許可出力
CKE	出力	出力	シンクロナス DRAM クロックイネーブル制御。 シンクロナス DRAM のセルフリフレッシュをサポートするための信号
$\overline{IVECF}$	出力	出力	割り込みベクタフェッチ
DREQ0	入力	入力	DMA 要求 0
DACK0	出力	出力	DMA アクノレッジ 0
DREQ1	入力	入力	DMA 要求 1
DACK1	出力	出力	DMA アクノレッジ 1
REFOUT	出力	出力	バス解放時リフレッシュ実行要求出力
DQMUU/ $\overline{WE3}$	出力	Hi-Z	シンクロナス DRAM 使用時: 最上位バイト (D31 ~ D24) の DQM 端子に接続 通常空間: 最上位バイト書き込み指示
DQMUL/ $\overline{WE2}$	出力	Hi-Z	シンクロナス DRAM 使用時: 2 バイト目 (D23 ~ D16) の DQM 端子に接続 通常空間: 2 バイト目書き込み指示
DQMLU/ $\overline{WE1}$	出力	Hi-Z	シンクロナス DRAM 使用時: 3 バイト目 (D15 ~ D8) の DQM 端子に接続 通常空間: 3 バイト目書き込み指示

表 7.1 端子構成 (1)

端子名	入出力	バス開放時	機能
DQMLL/WE $\bar{0}$	出力	Hi-Z	シンクロナス DRAM 使用時: 最下位バイト (D7 ~ D0) の DQM 端子に接続 通常空間: 最下位バイト書き込み指示
CAS $\bar{3}$	出力	Hi-Z	DRAM 使用時: 最上位バイト (D31-D24) の CAS 端子に接続
CAS $\bar{2}$	出力	Hi-Z	DRAM 使用時: 2 バイト目 (D23-D16) の CAS 端子に接続
CAS $\bar{1}$	出力	Hi-Z	DRAM 使用時: 3 バイト目 (D15-D8) の CAS 端子に接続
CAS $\bar{0}$	出力	Hi-Z	DRAM 使用時: 最下位バイト (D7-D0) の CAS 端子に接続
STATS0 ~ 1	出力	出力	バスマスタの識別 00: CPU 01: DMAC 10: E-DMAC 11: その他
BUSHIZ	入力	入力	WAIT 信号と組み合わせて、バスサイクルを終了せずにバスおよびストロープ信号を HiZ にする信号

【注】 Hi-Z: ハイインピーダンス

#### 7.1.4 レジスタ構成

バスステートコントローラには 10 本のレジスタがあります。これらのレジスタにより、ウェイトステート、バス幅、DRAM、シンクロナス DRAM、バースト ROM などのメモリとのインタフェース、DRAM、シンクロナス DRAM のリフレッシュの制御などを行います。レジスタ構成を表 7.2 に示します。

レジスタ本体のサイズは 16 ビットです。32 ビットでリードを行うと上位 16 ビットは 0 が読み出されます。レジスタに対する書き込みは誤書き込み防止のため書き込みデータの上位 16 ビットを H'A55A とした 32 ビットライトの場合のみ受け付け、それ以外は書き込みが行われません。レジスタのうち予約となっているビットには初期値を書き込んでください。

- 初期化の手順

メモリとのインタフェースの設定が終了するまでは CS0 空間以外はアクセスしないでください。

## 7. バスステートコントローラ (BSC)

表 7.2 レジスタ構成

名称	略称	R/W	初期値	アドレス *1	アクセスサイズ
バスコントロールレジスタ 1	BCR1	R/W	H'03F0	H'FFFFFFE0	16*2、32
バスコントロールレジスタ 2	BCR2	R/W	H'00FC	H'FFFFFFE4	16*2、32
バスコントロールレジスタ 3	BCR3	R/W	H'0F00	H'FFFFFFFC	16*2、32
ウェイトコントロールレジスタ 1	WCR1	R/W	H'AAFF	H'FFFFFFE8	16*2、32
ウェイトコントロールレジスタ 2	WCR2	R/W	H'000B	H'FFFFFFC0	16*2、32
ウェイトコントロールレジスタ 3	WCR3	R/W	H'0000	H'FFFFFFC4	16*2、32
個別メモリコントロールレジスタ	MCR	R/W	H'0000	H'FFFFFFEC	16*2、32
リフレッシュタイムコントロール/ステータスレジスタ	RTCSR	R/W	H'0000	H'FFFFFFF0	16*2、32
リフレッシュタイムカウンタ	RTCNT	R/W	H'0000	H'FFFFFFF4	16*2、32
リフレッシュタイムコンスタントレジスタ	RTCOR	R/W	H'0000	H'FFFFFFF8	16*2、32

【注】 \*1 このアドレスは 32 ビットアクセスの場合であり、16 ビットアクセス時のアクセスアドレスは + 2 したアドレスです。

\*2 16 ビットアクセスは読み出しのみ可能

### 7.1.5 アドレスマップ

本 LSI のアドレスマップはメモリ空間として 320M バイトあり、これを 5 個の部分空間に分割して使用します。接続されるデバイスの種類、データ幅はアドレスマップ上の各部分空間ごとに指定します。全体の空間のアドレスマップを表 7.3 に示します。キャッシュ領域とキャッシュスルー領域は実際には同一の空間であるため接続可能な最大メモリ空間は 160M バイトです。すなわち、プログラム上で H'20000000 番地のデータをアクセスすると、実際には H'00000000 番地のデータがアクセスされます。

本 LSI は内蔵メモリとして、8k バイトの RAM を内蔵しています。内蔵の RAM は各々 X 領域と Y 領域に分割され、DSP 命令で X、Y 領域が並列にアクセスできます。詳しくは、「SH-1/SH-2/SH-DSP プログラミングマニュアル」をご参照ください。

キャッシュ制御のための空間として、キャッシュパージのための連想パージ空間、アドレス (アドレスタグ) を読み書きするためのアドレスアレイリード/ライト空間、データアレイを強制的に読み書きするためのデータアレイリード/ライト空間があります。

表 7.3 アドレスマップ

アドレス	空間	メモリ種類	サイズ
H'00000000 ~ H'01FFFFFF	CS0 空間 キャッシュ領域	通常空間またはバーストROM	32M バイト
H'02000000 ~ H'03FFFFFF	CS1 空間 キャッシュ領域	通常空間	32M バイト
H'04000000 ~ H'05FFFFFF	CS2 空間 キャッシュ領域	通常空間またはシンクロナス DRAM*2	32M バイト
H'06000000 ~ H'07FFFFFF	CS3 空間 キャッシュ領域	通常空間、シンクロナス DRAM*2、DRAM	32M バイト
H'08000000 ~ H'09FFFFFF	CS4 空間 キャッシュ領域	通常空間 (IO デバイス)	32M バイト
H'0A000000 ~ H'0FFFFFFF	予約 *1		
H'10000000 ~ H'1000DFFF	予約 *1		
H'1000E000 ~ H'1000EFFF	内蔵 XRAM 領域		4k バイト
H'1000F000 ~ H'1001DFFF	予約 *1		
H'1001E000 ~ H'1001EFFF	内蔵 YRAM 領域		4k バイト
H'1001F000 ~ H'1FFFFFFF	予約 *1		
H'20000000 ~ H'21FFFFFF	CS0 空間 キャッシュスルー領域	通常空間またはバーストROM	32M バイト
H'22000000 ~ H'23FFFFFF	CS1 空間 キャッシュスルー領域	通常空間	32M バイト
H'24000000 ~ H'25FFFFFF	CS2 空間 キャッシュスルー領域	通常空間またはシンクロナス DRAM*2	32M バイト
H'26000000 ~ H'27FFFFFF	CS3 空間 キャッシュスルー領域	通常空間、シンクロナス DRAM*2、DRAM	32M バイト
H'28000000 ~ H'29FFFFFF	CS4 空間 キャッシュスルー領域	通常空間 (IO デバイス)	32M バイト
H'2A000000 ~ H'3FFFFFFF	予約 *1		
H'40000000 ~ H'49FFFFFF	連想バージ空間		160M バイト
H'4A000000 ~ H'5FFFFFFF	予約 *1		
H'60000000 ~ H'7FFFFFFF	アドレスアレイ リード/ライト空間		512M バイト
H'80000000 ~ H'BFFFFFFF	予約 *1		
H'C0000000 ~ H'C0000FFF	データアレイ リード/ライト空間		4k バイト
H'C0001000 ~ H'DFFFFFFF	予約 *1		
H'E0000000 ~ H'FFFFFFF	予約 *1		
H'FFFF0000 ~ H'FFFF0FFF	SDRAM モード設定用		4k バイト
H'FFFF1000 ~ H'FFFF7FFF	予約 *1		
H'FFFF8000 ~ H'FFFF8FFF	SDRAM モード設定用		4k バイト
H'FFFF9000 ~ H'FFFFBFFF	予約 *1		
H'FFFFC000 ~ H'FFFFFFF	内蔵周辺モジュール		

【注】 \*1 予約空間はアクセスしないでください。アクセスした場合、動作の保証はできません。

\*2 CS3 空間の SDRAM アクセスはバンクアクティブモードをサポートしています。  
CS2 空間の SDRAM アクセスはバンクアクティブモードをサポートしていません。  
常にオートプリチャージモードとなります。

## 7.2 レジスタの説明

### 7.2.1 バスコントロールレジスタ 1 (BCR1)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	A4LW 1	A4LW 0	A2EN DIAN	BST ROM	—	AHLW 1	AHLW 0	A1LW 1	A1LW 0	A0LW 1	A0LW 0	A4EN DIAN	DRAM 2	DRAM 1	DRAM 0
初期値:	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W						

ENDIAN、BSTROM、DRAM2-0の各ビットはパワーオンリセット後の初期設定時に書き込みを行い、以後は値を変更しないでください。他のビットを変更するための書き込みを行う場合には初期設定と同じ値を書いてください。レジスタの初期設定が終了するまではCS0空間以外はアクセスしないでください。

- ・ ビット 15: 予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

- ・ ビット 14、13: CS4空間のロングウェイト指定 (A4LW1、0)

ウェイトコントロールレジスタ2 (WCR2)のウェイトコントロールビット (W41、W40)がロングウェイト指定すなわち、11のときにCS4空間のアクセスに3 ~ 14サイクルのウェイトが挿入されます (表 7.4)。

- ・ ビット 12: CS2空間のエンディアン指定 (A2ENDIAN)

ビッグエンディアンの場合最上位バイト (MSB)が最も小さいバイトアドレスとなり以下最下位バイト (LSB)に向かって順にバイトデータが並びます。リトルエンディアンの場合にはLSBが最も小さいバイトアドレスとなり以下MSBに向かって順にデータが並びます。本ビットが1の場合CS2空間に対する読み出し/書き込み時にデータの並びをリトルエンディアンに並べなおして転送を行います。リトルエンディアンのプロセッサとのデータのやりとりおよびリトルエンディアンを意識して書かれたプログラムの実行に用います。

【注】 リトルエンディアン機能を使うことで、データの並び替えに時間がかかることはありません。

ビット 12	説明
A2ENDIAN	
0	ビッグエンディアン (初期値)
1	リトルエンディアン

- ・ ビット 11: エリア0バーストROMイネーブル (BSTROM)

ビット 11	説明
BSTROM	
0	エリア0を通常アクセスします。 (初期値)
1	エリア0をバーストROMとしてアクセスします。

- ・ ビット 10 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 9、8 : CS2 空間および CS3 空間のロングウェイト指定 (AHLW1、0)

CS2 空間または CS3 空間が通常空間の設定で、かつウェイトコントロールレジスタ 1 (WCR1) のそれぞれのエリアのウェイトコントロールビット (W21、W20 または W31、W30) がロングウェイト指定すなわち、11 のときに CS2 空間または CS3 空間のアクセスに 3 ~ 14 サイクルのウェイトが挿入されます (表 7.4)。

- ・ ビット 7、6 : CS1 空間のロングウェイト指定 (A1LW1、0)

ウェイトコントロールレジスタ 1 (WCR1) のウェイトコントロールビット (W11、W10) がロングウェイト指定すなわち、11 のときにエリア 1 のアクセスに 3 ~ 14 サイクルのウェイトが挿入されます (表 7.4)。

- ・ ビット 5、4 : CS0 空間のロングウェイト指定 (A0LW1、0)

CS0 空間が通常空間の設定で、かつウェイトコントロールレジスタ 1 (WCR1) のウェイトコントロールビット (W01、W00) がロングウェイト指定すなわち、11 のときにエリア 0 のアクセスに 3 ~ 14 サイクルのウェイトが挿入されます (表 7.4)。

- ・ ビット 3 : CS4 空間のエンディアン指定 (A4ENDIAN)

ビッグエンディアンの場合最上位バイト (MSB) が最も小さいバイトアドレスとなり以下最下位バイト (LSB) に向かって順にバイトデータが並びます。リトルエンディアンの場合には LSB が最も小さいバイトアドレスとなり以下 MSB に向かって順にデータが並びます。本ビットが 1 の場合 CS4 空間に対する読み出し / 書き込み時にデータの並びをリトルエンディアンに並べなおして転送を行います。リトルエンディアンのプロセッサとのデータのやりとりおよびリトルエンディアンを意識して書かれたプログラムの実行に用います。

【注】 リトルエンディアン機能を使うことで、データの並び替えに余分な時間がかかることはありません。

ビット 3	説 明
A4ENDIAN	
0	ビッグエンディアン (初期値)
1	リトルエンディアン

## 7. バスステートコントローラ (BSC)

### ・ ビット 2 ~ 0 : DRAM 他メモリエnable (DRAM2 ~ 0)

ビット 2	ビット 1	ビット 0	説 明
DRAM2	DRAM1	DRAM0	
0	0	0	CS2 空間 ,CS3 空間を通常空間とします。 (初期値)
0	0	1	CS2 空間を通常空間、CS3 空間をシンクロナス DRAM 空間とします。
0	1	0	CS2 空間を通常空間、CS3 空間を DRAM 空間とします。
0	1	1	予約 (設定しないでください)
1	0	0	CS2 空間をシンクロナス DRAM 空間、CS3 空間を通常空間とします。
1	0	1	CS2 空間 ,CS3 空間をともにシンクロナス DRAM 空間とします。
1	1	0	予約 (設定しないでください)
1	1	1	予約 (設定しないでください)

表 7.4 BCR1、3 のレジスタ設定におけるウェイト値 (全空間共通)

BCR3 AnLW2	BCR1		ウェイト値
	AnLW1	AnLW0	
0	0	0	3 サイクル挿入
0	0	1	4 サイクル挿入
0	1	0	5 サイクル挿入
0	1	1	6 サイクル挿入
1	0	0	8 サイクル挿入
1	0	1	10 サイクル挿入
1	1	0	12 サイクル挿入
1	1	1	14 サイクル挿入 (初期値)

【注】 n : 各 CS 空間番号 (0 ~ 4)  
ただし、AHLW2、1、0 は CS2 空間 ,CS3 空間で共通

## 7.2.2 バスコントロールレジスタ 2 (BCR2)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	A4 SZ1	A4 SZ0	A3 SZ1	A3 SZ0	A2 SZ1	A2 SZ0	A1 SZ1	A1 SZ0	—	—
初期値:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0
R/W:	R	R	R	R	R	R	R/W	R	R							

BCR2 はパワーオンリセット後の初期設定時に書き込みを行い、以後は値を変更しないでください。書き込みを行う場合には初期設定と同じ値を書いてください。レジスタの初期設定が終了するまでは CS0 空間以外はアクセスしないでください。

CS0 空間のバスサイズ指定は、端子 MD4、MD3 で設定します。説明は「3.3 CS0 空間のバス幅」を参照してください。

- ・ ビット 15 ~ 10 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 9、8 : CS4 空間のバスサイズ指定 (A4SZ1、0)

ビット 9	ビット 8	説明
A4SZ1	A4SZ0	
0	0	ロングワード (32 ビット) サイズ (初期値)
0	1	バイト (8 ビット) サイズ
1	0	ワード (16 ビット) サイズ
1	1	ロングワード (32 ビット) サイズ

- ・ ビット 7、6 : CS3 空間のバスサイズ指定 (A3SZ1、0) (通常空間設定でのみ有効)

ビット 7	ビット 6	説明
A3SZ1	A3SZ0	
0	0	予約 (設定しないでください)
0	1	バイト (8 ビット) サイズ
1	0	ワード (16 ビット) サイズ
1	1	ロングワード (32 ビット) サイズ (初期値)

- ・ ビット 5、4 : CS2 空間のバスサイズ指定 (A2SZ1、0) (通常空間設定でのみ有効)

ビット 5	ビット 4	説明
A2SZ1	A2SZ0	
0	0	予約 (設定しないでください)
0	1	バイト (8 ビット) サイズ
1	0	ワード (16 ビット) サイズ
1	1	ロングワード (32 ビット) サイズ (初期値)

## 7. バスステートコントローラ (BSC)

- ビット 3、2 : CS1 空間のバスサイズ指定 (A1SZ1、0)

ビット 3	ビット 2	説 明
A1SZ1	A1SZ0	
0	0	予約 (設定しないでください)
0	1	バイト (8 ビット) サイズ
1	0	ワード (16 ビット) サイズ
1	1	ロングワード (32 ビット) サイズ (初期値)

- ビット 1、0 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### 7.2.3 バスコントロールレジスタ 3 (BCR3)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	A4 LW2	AH LW2	A1 LW2	A0 LW2	DS WW1	DS WW0	—	—	—	BA SEL	EDO	BWE
初期値 :	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W

BASEL、EDO、BWE の各ビットはパワーオンリセット後の初期設定時に書き込みを行い、以後は値を変更しないでください。他のビットを変更するための書き込みを行う場合には初期設定と同じ値を書き込んでください。レジスタの初期設定が終了するまでは CS0 空間以外はアクセスしないでください。

- ビット 15 ~ 12 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も 0 にしてください。
- ビット 11 ~ 8 : CS4 ~ 0 空間のロングウェイト指定 (AnLW2)  
CSn 空間が通常空間の設定で、かつウェイトコントロールレジスタの指定ビットがロングウェイト指定すなわち、11 のとき、BCR1 のロングウェイト指定ビット AnLW1、AnLW0 と組み合わせて、CSn 空間のアクセスに 3 ~ 14 サイクルのウェイトを挿入できます。ロングウェイトの具体的な説明は「7.2.1 バスコントロールレジスタ (BCR1)」を参照してください。
- ビット 7、6 : DMA シングルライトウェイト (DSWW1、DSWW0)  
DRAM、EDO に DMA シングルアドレスモードで書き込むときに DACK アサートから  $\overline{\text{CASn}}$  アサートの間に挿入するウェイトステート数を決めるビットです。

ビット7	ビット6	説明
DSWW1	DSWW0	
0	0	0ウェイト (初期値)
	1	1ウェイト
1	0	2ウェイト
	1	予約 (設定しないでください)

- ・ ビット5～3: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット2: 64M シンクロナス DRAM 使用時のバンク数指定 (BASEL)  
MCR の AMX2～0 で、64M シンクロナス DRAM を指定した場合、バンク数を指定できます。

ビット2	説明
BASEL	
0	4バンク (初期値)
1	2バンク

- ・ ビット1: EDO モード指定 (EDO)  
CS3 空間に DRAM を指定した場合、EDO モードを指定できます。

ビット1	説明
EDO	
0	高速ページモード (初期値)
1	EDO モード

- ・ ビット0: SDRAM パーストライト指定 (BWE)  
CS2 または CS3 空間にシンクロナス DRAM を指定した場合、パーストライトモードを指定できます。

ビット0	説明
BWE	
0	シングルライトモード (初期値)
1	パーストライトモード

## 7.2.4 ウェイトコントロールレジスタ 1 (WCR1)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IW31	IW30	IW21	IW20	IW11	IW10	IW01	IW00	W31	W30	W21	W20	W11	W10	W01	W00
初期値:	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W							

## 7. バスステートコントローラ (BSC)

レジスタの初期設定が終了するまでは CS0 空間以外はアクセスしないでください。

### ・ビット 15 ~ 8 : CS3 ~ 0 空間サイクル間アイドル指定 (IW31 ~ IW00)

サイクル間アイドル指定は異なる CS 空間を続けてアクセスする場合にアクセス間に挿入するアイドルサイクルの指定を行います。これはリードデータのバッファオフの遅い ROM などと、高速なメモリ、IO インタフェースなどのデータの衝突を防止するために使用します。同一 CS 空間に対するアクセスであっても、直前のアクセスがリードで、次のアクセスがライトの場合にはアイドルサイクルの挿入を行います。アイドルサイクルの指定は直前にアクセスされた CS 空間のものが有効となります。

なお、下記設定値は、最小アイドルサイクル数を示しており、実際にはサイクル間アイドル数が示されている数以上挿入される場合があります。

IW31、IW21、IW11、IW01	IW30、IW20、IW10、IW00	説 明
0	0	アイドルサイクルなし
0	1	1 アイドルサイクル挿入
1	0	2 アイドルサイクル挿入 (初期値)
1	1	4 アイドルサイクル挿入

### ・ビット 7 ~ 0 : CS3 ~ 0 空間ウェイトコントロール (W31 ~ W00)

CSn 空間が通常空間の設定の場合、Wn1、Wn0 により CSn 空間のウェイト数を設定できます。

#### [ 基本インタフェース ]

W31、W21、W11、W01	W30、W20、W10、W00	説 明
0	0	ノーウェイト外部ウェイト入力無視
0	1	1 ウェイト、外部ウェイト入力イネーブル
1	0	2 ウェイト、外部ウェイト入力イネーブル
1	1	バスコントロールレジスタ 1、3 (BCR1、BCR3) のロングウェイト指定に従います。外部ウェイト入力はイネーブル。 (初期値)

CS3 空間が DRAM の場合 W31、W30 のウェイトコントロールによって  $\overline{\text{CAS}}$  アサートサイクル数の指定を行います。

#### [ DRAM インタフェース ]

ビット 7 W31	ビット 6 W30	説 明
0	0	1 サイクル
0	1	2 サイクル
1	0	3 サイクル
1	1	予約 (設定しないでください) (初期値)

WCR2 の外部ウェイトマスクビット A3WM が 0 で、 $\overline{\text{CAS}}$  アサートサイクル数が 2 サイクル以上設定の場合、外部ウェイト入力がイネーブルとなります。

CS2 空間または CS3 空間がシンクロナス DRAM の場合それぞれ W31、W30 および W21、W20 のウェイトコントロールによって CAS レイテンシの指定を行います。

[ シンクロナス DRAM インタフェース ]

W31、W21	W30、W20	説 明
0	0	1 サイクル
0	1	2 サイクル
1	0	3 サイクル
1	1	4 サイクル (初期値)

シンクロナス DRAM では、いかなる設定でも外部ウェイト入力は無視されます。

### 7.2.5 ウェイトコントロールレジスタ 2 (WCR2)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A4 WD1	A4 WD0	—	A4 WM	A3 WM	A2 WM	A1 WM	A0 WM	—	—	—	—	IW41	IW40	W41	W40
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
R/W :	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

- ・ ビット 15、14 : CS4 空間外部ウェイト数指定 (A4WD1、A4WD0)  
CS4 空間の外部ウェイトネゲート受け付けから、 $\overline{RD}$ 、 $\overline{WE_n}$  ネゲートまでのサイクル数を指定できます。

ビット 15	ビット 14	説 明
A4WD1	A4WD0	
0	0	1 サイクル (初期値)
0	1	2 サイクル
1	0	4 サイクル
1	1	予約 (設定しないでください)

- ・ ビット 13 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 12 ~ 8 : CS4 ~ 0 空間外部ウェイトマスク指定 (A4WM ~ A0WM)  
CS4 ~ 0 空間のウェイトをマスクできます。CS4 ~ 0 空間のウェイトコントロール (W41 ~ W00) を 00 以外に設定すると外部ウェイト入力をイネーブルにできますが、本ビットを 1 にすることによりマスクできます。ただし、シンクロナス DRAM では、いかなる設定でも外部ウェイト入力は無視されます。

## 7. バスステートコントローラ (BSC)

A4WM	W41	W40	説 明
A3WM	W31	W30	
A2WM	W21	W20	
A1WM	W11	W10	
A0WM	W01	W00	
0	0	0	
0	0	1	外部ウェイト入力無視
0	1	0	外部ウェイト入力イネーブル
0	1	1	外部ウェイト入力イネーブル (初期値)
1	*	*	外部ウェイト入力無視

\* : don't care

- ビット7 ~ 4 : 予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

- ビット3、2 : CS4 空間サイクル間アイドル指定 (IW41、IW40)

CS3 ~ 0 空間に対するサイクル間アイドル指定と同様に、CS4 空間にも指定できます。

下記設定値は、最小アイドルサイクル数を示しており、実際にはサイクル間アイドル数が示されている数以上挿入される場合があります。

ビット3	ビット2	説 明
IW41	IW40	
0	0	アイドルサイクルなし
0	1	1 アイドルサイクル挿入
1	0	2 アイドルサイクル挿入 (初期値)
1	1	4 アイドルサイクル挿入

- ビット1、0 : CS4 空間ウェイトコントロール (W41、W40)

CS3 ~ 0 空間に対するウェイトと同様に、CS4 空間にも指定できます。

ビット1	ビット0	説 明
W41	W40	
0	0	ノーウェイト、外部ウェイト入力無視
0	1	1 ウェイト、外部ウェイト入力イネーブル
1	0	2 ウェイト、外部ウェイト入力イネーブル
1	1	バスコントロールレジスタ1、3 (BCR1、BCR3) のロングウェイト指定にしたがいません。外部ウェイト入力はイネーブル。 (初期値)

## 7.2.6 ウェイトコントロールレジスタ 3 (WCR3)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	A4 SW2	A4 SW1	A4 SW0	—	A4 HW1	A4 HW0	A3 SHW1	A3 SHW0	A2 SHW1	A2 SHW0	A1 SHW1	A1 SHW0	A0 SHW1	A0 SHW0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ビット 15、14: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 13 ~ 11: CS4 空間アドレス、 $\overline{CS4}$  -  $\overline{RD}$ 、 $\overline{WE_n}$  アサート (A4SW2 ~ A4SW0)  
CS4 空間における、アドレス、 $\overline{CS4}$  出力から、 $\overline{RD}$ 、 $\overline{WE_n}$  アサートまでのサイクル数を設定できます。

ビット 13 A4SW2	ビット 12 A4SW1	ビット 11 A4SW0	説 明
0	0	0	0.5 サイクル (初期値)
0	0	1	1.5 サイクル
0	1	0	3.5 サイクル
0	1	1	5.5 サイクル
1	0	0	7.5 サイクル
1	0	1	予約 (設定しないでください)
1	1	0	予約 (設定しないでください)
1	1	1	予約 (設定しないでください)

- ビット 10: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 9、8: CS4 空間  $\overline{RD}$ 、 $\overline{WE_n}$  ネゲート - アドレス、 $\overline{CS4}$  ホールド (A4HW1、A4HW0)  
CS4 空間における、 $\overline{RD}$ 、 $\overline{WE_n}$  ネゲートからアドレス、 $\overline{CS4}$  をホールドするサイクル数を設定できます。

ビット 9 A4HW1	ビット 8 A4HW0	説 明
0	0	0.5 サイクル、CS4 のホールドサイクルは 0 サイクル (初期値)
0	1	1.5 サイクル、CS4 のホールドサイクルは 1 サイクル
1	0	3.5 サイクル、CS4 のホールドサイクルは 3 サイクル
1	1	5.5 サイクル、CS4 のホールドサイクルは 5 サイクル

- ビット 7 ~ 0: CS3 ~ 0 空間  $\overline{CS_n}$  アサート期間拡張 (A3SHW1 ~ A0SHW0)  
CS3 ~ 0 空間における、アドレス、 $\overline{CS_n}$  出力から、 $\overline{RD}$ 、 $\overline{WE_n}$  アサートまでのサイクル数、および、 $\overline{RD}$ 、 $\overline{WE_n}$  ネゲートからアドレス、 $\overline{CS_n}$  のホールドサイクル数を指定できます。

## 7. バスステートコントローラ (BSC)

A3SHW1	A3SHW0	説 明
A2SHW1	A2SHW0	
A1SHW1	A1SHW0	
A0SHW1	A0SHW0	
0	0	
0	1	1.5 サイクル、CSn* のホールドサイクルは 1 サイクル
1	0	2.5 サイクル、CSn* のホールドサイクルは 2 サイクル
1	1	予約 (設定しないでください)

【注】 \*n = 0 ~ 3

### 7.2.7 個別メモリコントロールレジスタ (MCR)

ビット	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRP	RCD	TRWL	TRAS	TRAS	BE	RASD	TRWL	AMX2	SZ	AMX1	AMX0	RFSH	RMODE	TRP	RCD
	0	0	0	1	0			1							1	1
初期値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TRP1 ~ 0、RCD1 ~ 0、TRWL1 ~ 0、TRAS1 ~ 0、BE、RASD、AMX2 ~ 0、SZ の各ビットはパワーオンリセット後の初期設定時に書き込みを行い、以後は値を変更しないでください。書き込みを行う場合には初期設定と同じ値を書き込んでください。レジスタの初期設定が終了するまでは CS2、CS3 空間はアクセスしないでください。

#### ・ビット 1、15 : RAS プリチャージ期間 (TRP1、TRP0)

DRAM 接続時、RAS ネゲート後次にアサートするまでの最小サイクル数を指定します。

シンクロナス DRAM 接続時、プリチャージ後、バンクアクティブコマンド出力までの最小サイクル数を指定します。詳細は「7.5 シンクロナス DRAM インタフェース」を参照してください。

#### [ DRAM インタフェース ]

ビット 1	ビット 15	説 明	
TRP1	TRP0		
0	0		1 サイクル (初期値)
0	1		2 サイクル
1	0		予約
1	1	予約	

#### [ シンクロナス DRAM インタフェース ]

ビット 1	ビット 15	説 明	
TRP1	TRP0		
0	0		1 サイクル (初期値)
0	1		2 サイクル
1	0		3 サイクル
1	1	4 サイクル	

- ・ ビット 0、14 :  $\overline{\text{RAS}}$  -  $\overline{\text{CAS}}$  遅延 (RCD1、RCD0)

DRAM 接続時、 $\overline{\text{RAS}}$  アサート後、 $\overline{\text{CAS}}$  アサートまでのサイクル数を指定します。

シンクロナス DRAM 接続時、バンクアクティブ (ACTV) コマンド発行後、リードまたはライト (READ、READA、WRIT、WRITA) コマンド発行までのサイクル数を指定します。

ビット 0	ビット 14	説 明
RCD1	RCD0	
0	0	1 サイクル (初期値)
0	1	2 サイクル
1	0	3 サイクル
1	1	予約 (設定しないでください)

- ・ ビット 8、13 : ライト - プリチャージ遅延 (TRWL1、TRWL0)

シンクロナス DRAM でバンクアクティブモードでないとき、ライトサイクル後オートプリチャージが起動されるまでのサイクル数を指定します。バスステートコントローラ内部ではこれにより次のアクティブコマンド発行可能タイミングを計算します。バンクアクティブモードのときはライト (WRIT) コマンド発行後のプリチャージコマンド発行までのサイクル数を指定します。

シンクロナス DRAM 以外のメモリを接続する場合には本ビットは無視されます。

ビット 8	ビット 13	説 明
TRWL1	TRWL0	
0	0	1 サイクル (初期値)
0	1	2 サイクル
1	0	3 サイクル
1	1	予約 (設定しないでください)

- ・ ビット 12、11 :  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュ  $\overline{\text{RAS}}$  アサート期間 (TRAS1、0)

DRAM 接続時は  $\overline{\text{RAS}}$  のアサート幅を指定します。

#### [ DRAM インタフェース ]

ビット 12	ビット 11	説 明
TRAS1	TRAS0	
0	0	2 サイクル (初期値)
0	1	3 サイクル
1	0	4 サイクル
1	1	5 サイクル

シンクロナス DRAM はオートリフレッシュコマンド発行後、TRP ビットの値にかかわらず、TRAS サイクルの間バンクアクティブコマンドを発行しません。シンクロナス DRAM の場合には  $\overline{\text{RAS}}$  アサート期間はありませんがリフレッシュコマンドを発行してから次にアクセスを行うまでの制限時間があるのでそれを守るようにこの値を設定します。セルフリフレッシュ解除時 TRAS サイクルの間コマンドの発行を行いません。

## 7. バスステートコントローラ (BSC)

### [ シンクロナス DRAM インタフェース ]

ビット 12 TRAS1	ビット 11 TRAS0	説 明
0	0	3 サイクル (初期値)
0	1	4 サイクル
1	0	6 サイクル
1	1	9 サイクル

#### ・ ビット 10 : パーストイネーブル (BE)

ビット 10 BE	説 明
0	パースト禁止 (初期値)
1	DRAM、EDO インタフェース時高速ページモードをイネーブルします。 パーストアクセスする条件は、 ・ 16 ビットバス幅に対する、ロングワードアクセス、キャッシュフィルアクセス、DMAC による 16 バイト転送 ・ 32 ビットバス幅に対する、キャッシュフィルアクセス、DMAC による 16 バイト転送です。 シンクロナス DRAM のアクセスは本ビットの指定によらず常にパーストイネーブル状態です。

#### ・ ビット 9 : バンクアクティブモード (RASD)

ビット 9 RASD	説 明
0	DRAM の場合、アクセス終了後 RAS をネゲートします (通常オペレーション)。シンクロナス DRAM の場合、オートプリチャージモードを用いてリード/ライトを行います。したがって、次のアクセスは必ずバンクアクティブコマンドから開始されます。 (初期値)
1	DRAM の場合、アクセス終了後 RAS をアサートしたまま放置する RAS ダウンモードになります。DRAM 以外に書き込みを行う外部デバイスを接続して本モードを使用する場合は「7.6.5 パーストアクセス」の説明を参照してください。 シンクロナス DRAM の場合、バンクアクティブ状態のままアクセスを終了します。 <u>ただし、CS3 空間のみ有効です。CS2 空間をシンクロナス DRAM とした場合は常にオートプリチャージモードとなります。</u>

- ・ ビット7、5、4：アドレスマルチプレクス (AMX2 ~ 0)

## [ DRAM インタフェース ]

ビット7 AMX2	ビット5 AMX1	ビット4 AMX0	説 明
0	0	0	カラムアドレス 8 ビット品 (初期値)
0	0	1	カラムアドレス 9 ビット品
0	1	0	カラムアドレス 10 ビット品
0	1	1	カラムアドレス 11 ビット品
1	0	0	予約 (設定しないでください)
1	0	1	予約 (設定しないでください)
1	1	0	予約 (設定しないでください)
1	1	1	予約 (設定しないでください)

## [ シンクロナス DRAM インタフェース ]

ビット7 AMX2	ビット5 AMX1	ビット4 AMX0	説 明
0	0	0	16M ビット品 (1M × 16 ビット) 64M ビット品 (2M × 32 ビット) * <sup>2</sup> (初期値)
0	0	1	16M ビット品 (2M × 8 ビット) * <sup>1</sup>
0	1	0	16M ビット品 (4M × 4 ビット) * <sup>1</sup>
0	1	1	4M ビット品 (256K × 16 ビット)
1	0	0	64M ビット品 (4M × 16 ビット) 128M ビット品 (4M × 32 ビット) * <sup>3</sup>
1	0	1	64M ビット品 (8M × 8 ビット) * <sup>1</sup> 、128M ビット品 (8M × 16 ビット) * <sup>1</sup> * <sup>4</sup> 、256M ビット品 (8M × 32 ビット) * <sup>1</sup> * <sup>5</sup>
1	1	0	予約 (設定しないでください)
1	1	1	2M ビット品 (128K × 16 ビット)

- [注] \*<sup>1</sup> MCR の SZ ビットが 0 (16 ビットバス幅) のときは予約となるので設定しないでください。  
\*<sup>2</sup> 64M ビット品 (2M × 32 ビット) との接続方法については、図 7.34 を参照してください。  
\*<sup>3</sup> 128M ビット品 (4M × 32 ビット) との接続方法については、図 7.35 を参照してください。  
\*<sup>4</sup> 128M ビット品 (8M × 16 ビット) と接続する場合には、図 7.36 のように、32 ビット幅で接続してください。  
\*<sup>5</sup> 256M ビット品 (8M × 32 ビット) との接続方法については、図 7.37 を参照してください。

- ・ ビット6：メモリデータサイズ (SZ)

シンクロナス DRAM、DRAM 空間の場合には BCR2 のデータバス幅は無視され本ビットの指定に従います。

ビット6 SZ	説 明
0	ワード (16 ビット) サイズ (初期値)
1	ロングワード (32 ビット) サイズ

- ・ ビット3：リフレッシュ制御 (RFSH)

DRAM、シンクロナス DRAM のリフレッシュ動作を行うか否かを決めます。

本ビットはスレーブモードでは無効となり、常に 0 として扱われます。

## 7. バスステートコントローラ (BSC)

ビット3	説明
RFSH	
0	しない (初期値)
1	する

### ・ ビット2: リフレッシュモード (RMODE)

本ビットは RFSH ビットが 1 のとき、通常のリフレッシュを行うか、セルフリフレッシュを行うかを選択するビットです。RFSH ビットが 0 の場合には、本ビットを 1 にしないでください。RFSH ビットを 1、本ビットを 1 にセットするとその直後にセルフリフレッシュモードに入ります。RFSH ビットを 1、本ビットを 0 にセットすると  $\overline{\text{CAS}}$  ピフォ  $\overline{\text{RAS}}$  リフレッシュまたはオートリフレッシュを 8 ビットインターバルタイムに設定した間隔で行います。外部アクセスを行っている最中にリフレッシュ要求が生じた場合アクセスサイクルが終了するまで待ってリフレッシュを行います。

セルフリフレッシュにセットした場合は、シンクロナス DRAM エリアアクセス中でなければ直ちにセルフリフレッシュモードに入ります。アクセス中の場合はアクセスが終了してからセルフリフレッシュモードに入ります。なお、セルフリフレッシュ中のインターバルタイムによるリフレッシュ要求はすべて無視されます。

ビット2	説明
RMODE	
0	通常のリフレッシュを行う (初期値)
1	セルフリフレッシュを実行する

## 7.2.8 リフレッシュタイムコントロール/ステータスレジスタ (RTCSR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CMF	CMIE	CKS2	CKS1	CKS0	RRC2	RRC1	RRC0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

### ・ ビット 15 ~ 8: 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### ・ ビット 7: コンペアマッチフラグ (CMF)

RTCNT の値と RTCOR の値が一致したことを示すステータスフラグであり、次の条件でセット/クリアされません。

ビット7 CMF	説 明
0	[クリア条件] (初期値) CMF = 1 の状態で RTCSR を読み出した後、CMF に 0 を書き込んだとき
1	[セット条件] RTCNT = RTCOR になったとき

- ・ ビット6: コンペアマッチインタラプトイネーブル (CMIE)  
RTCSR の CMF が 1 にセットされたとき、CMF による割り込み要求を許可または禁止します。

ビット6 CMIE	説 明
0	CMF による割り込み要求を禁止 (初期値)
1	CMF による割り込み要求を許可

- ・ ビット5 ~ 3: クロックセレクトビット (CKS2 ~ 0)

ビット5 CKS2	ビット4 CKS1	ビット3 CKS0	説 明
0	0	0	カウントアップ停止 (初期値)
0	0	1	P / 4
0	1	0	P / 16
0	1	1	P / 64
1	0	0	P / 256
1	0	1	P / 1024
1	1	0	P / 2048
1	1	1	P / 4096

- ・ ビット2 ~ 0: リフレッシュ回数 (RRC2 ~ 0)  
リフレッシュタイマカウンタ (RTCNT) とリフレッシュタイムコンスタントレジスタ (RTCOR) が一致しリフレッシュ要求が出たとき、連続してリフレッシュを行う回数を指定します。

## 7. バスステートコントローラ (BSC)

ビット2	ビット1	ビット0	説 明
RRC2	RRC1	RRC0	
0	0	0	1回 (初期値)
0	0	1	2回
0	1	0	4回
0	1	1	6回
1	0	0	8回
1	0	1	予約 (設定しないでください)
1	1	0	予約 (設定しないでください)
1	1	1	予約 (設定しないでください)

### 7.2.9 リフレッシュタイマカウンタ (RTCNT)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—								
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R/W							

RTCNTは8ビットのカウンタで、入力したクロックによりカウントアップされます。入力するクロックの選択はRTCSCRのクロックセレクトビットで行います。RTCNTの値はCPUから常に読み出し/書き込み可能です。RTCNTがRTCORと一致すると、RTCNTはクリアされます。255までカウントアップすると次は0に戻ります。

- ・ ビット15～8: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

## 7.2.10 リフレッシュタイムコンスタントレジスタ (RTCOR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—									
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W								

RTCOR は 8 ビットの読み出し / 書き込み可能なレジスタです。RTCOR と RTCNT の値は常に比較され、両方の値が一致すると RTCSR のコンペアマッチフラグ CMF がセットされ、RTCNT は 0 にクリアされます。

MCR のリフレッシュビット (RFSH) が 1 にセットされている場合、この一致信号によってリフレッシュ要求が発生します。リフレッシュ要求は実際にリフレッシュ動作が行われるまで保持されます。次の一致までにリフレッシュ要求が処理されない場合は前の要求は無効となります。

RTCSR の CMIE が 1 にセットされているとこの一致信号によって割り込みコントローラに対し割り込み要求が発生させます。割り込み要求は RTCSR の CMF がクリアされるまで続けて出力されます。CMF のクリアは割り込みのみに影響を及ぼし、リフレッシュ要求がこれによってクリアされることはありません。したがってリフレッシュを行いながら、リフレッシュ要求の数を割り込みを用いてカウントするなど、リフレッシュとインターバルタイム割り込みの同時設定を行うことも可能です。

- ・ ビット 15 ~ 8 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 7.3 アクセスサイズとデータアライメント

### 7.3.1 通常デバイスとの接続

アクセスの単位としてはバイト、ワード、ロングワードの単位で可能です。

データのアライメントは各デバイスのデータ幅にあわせて行われます。バイト幅のデバイスからロングワードデータを読み出すためには、4回の読み出し動作が必要です。バスステートコントローラはそれぞれのインタフェースの間でデータのアライメント、およびデータ長の変換を自動的に行います。外部デバイスのデータ幅は、CS0空間はモードピンによって、CS1～CS4空間はBCR2の設定を行うことによって8ビット、16ビット、32ビットのいずれでも接続することができます。ただし、本機能はそれぞれの空間に接続するデバイスのデータ幅を静的に指定するためのものであり、アクセスサイクルごとにデータの幅を変更することはできません。

デバイスのデータ幅とアクセスの単位との関係を図 7.2～図 7.4 に示します。

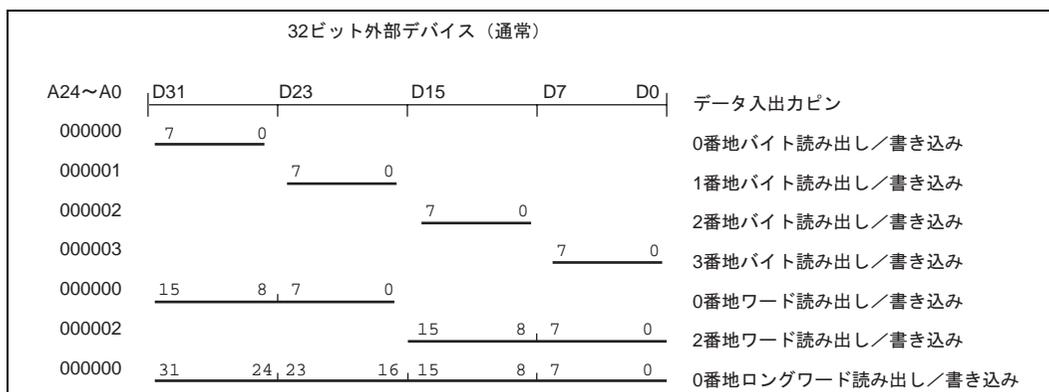


図 7.2 32ビット外部デバイスとアクセス単位の関係

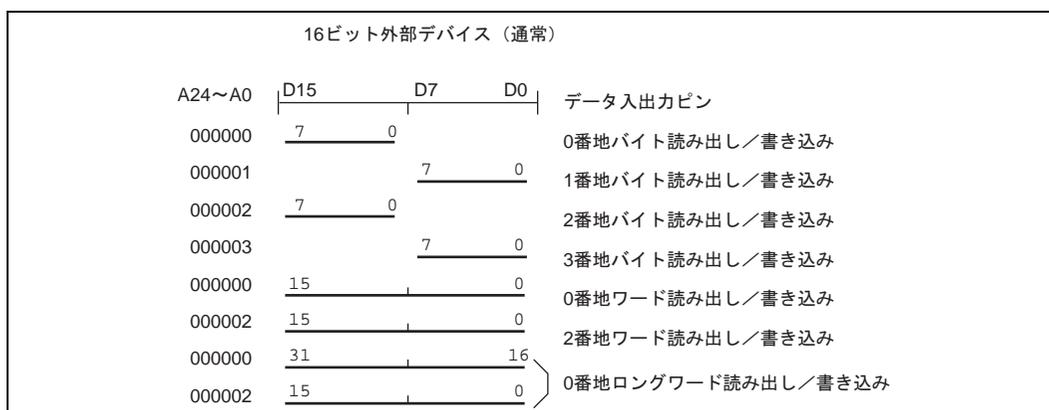


図 7.3 16ビット外部デバイスとアクセス単位の関係

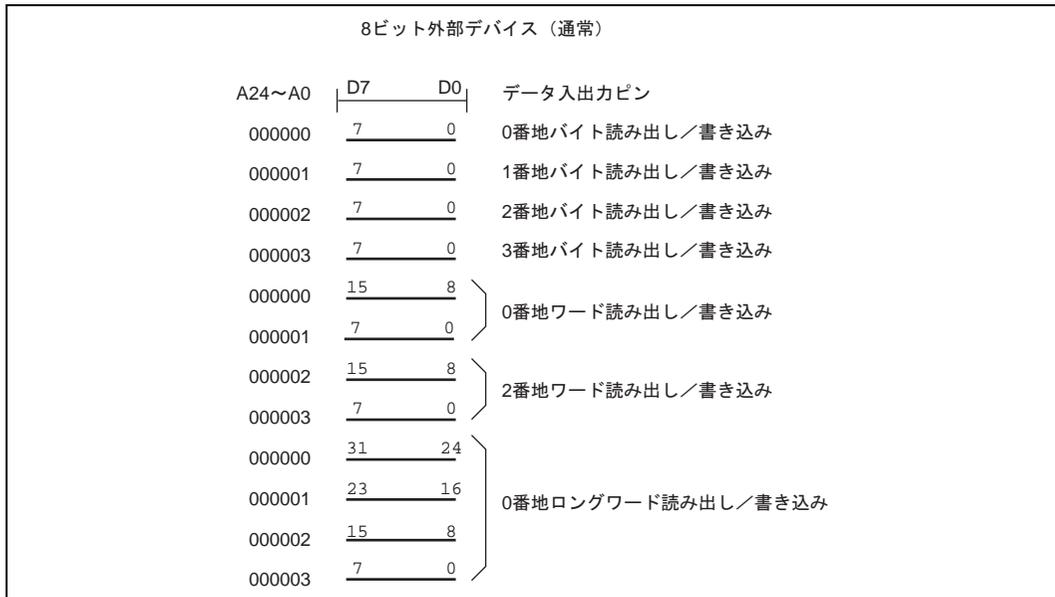


図 7.4 8ビット外部デバイスとアクセス単位の関係

## 7. バスステートコントローラ (BSC)

### 7.3.2 リトルエンディアンのデバイスとの接続

本 LSI では、バイトデータのならばが LSB が 0 番地側になるリトルエンディアンのデバイスとの接続、およびデータの互換性を維持するために、CS2、CS4 空間でエンディアンの変換機能を提供しています。BCR1 のエンディアン指定ビットを 1 に設定すると、CS2、CS4 空間はリトルエンディアンとなります。リトルエンディアン時のデバイスのデータ幅とアクセス単位の間隔を図 7.5 ~ 図 7.7 に示します。リトルエンディアンのバスマスタとメモリなどを共有する場合、32 ビット幅では本 LSI の D31 ~ D24 を相手のバスマスタの最下位バイト (LSB) に、D7 ~ D0 を最上位バイト (MSB) に接続します。16 ビット幅では、D15 ~ D8 を相手の LSB に、D7 ~ D0 を MSB に接続します。

本機能ではデータ変換のみサポートしています。したがって、CS2、CS4 空間にプログラムおよび定数を置かないように注意が必要です。また、本機能を用いるときは、書き込み側と読み出し側のアクセス単位を統一してください。

例えば、ロングワードアクセスで書き込んだデータは、ロングワードで読み出してください。書き込み時と異なるアクセス単位で読み出すと誤った値が読み出されます。

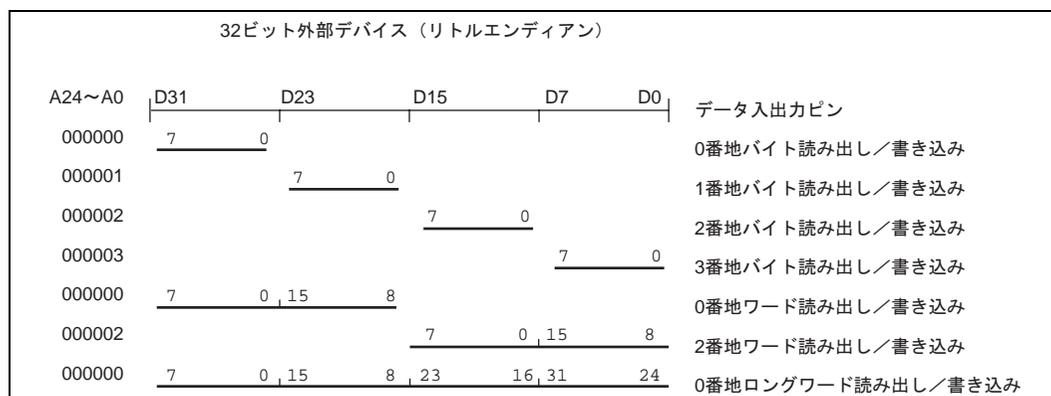


図 7.5 32 ビット外部デバイスとアクセス単位の間隔

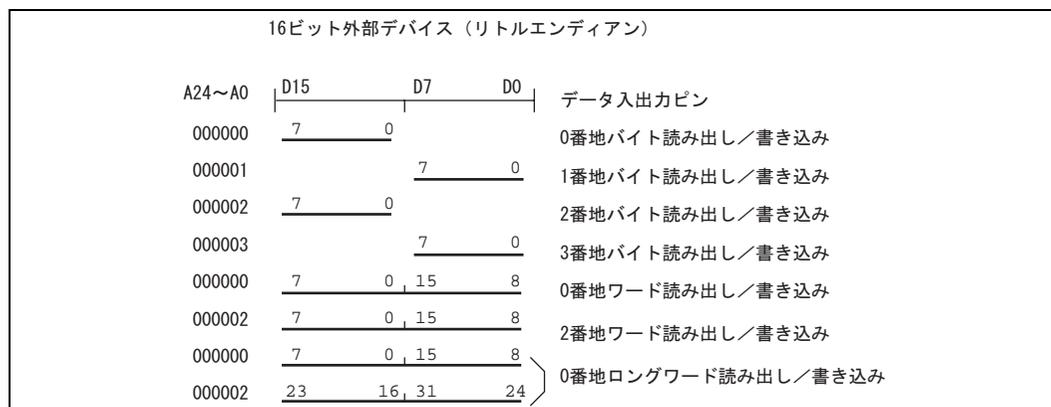


図 7.6 16 ビット外部デバイスとアクセス単位の間隔

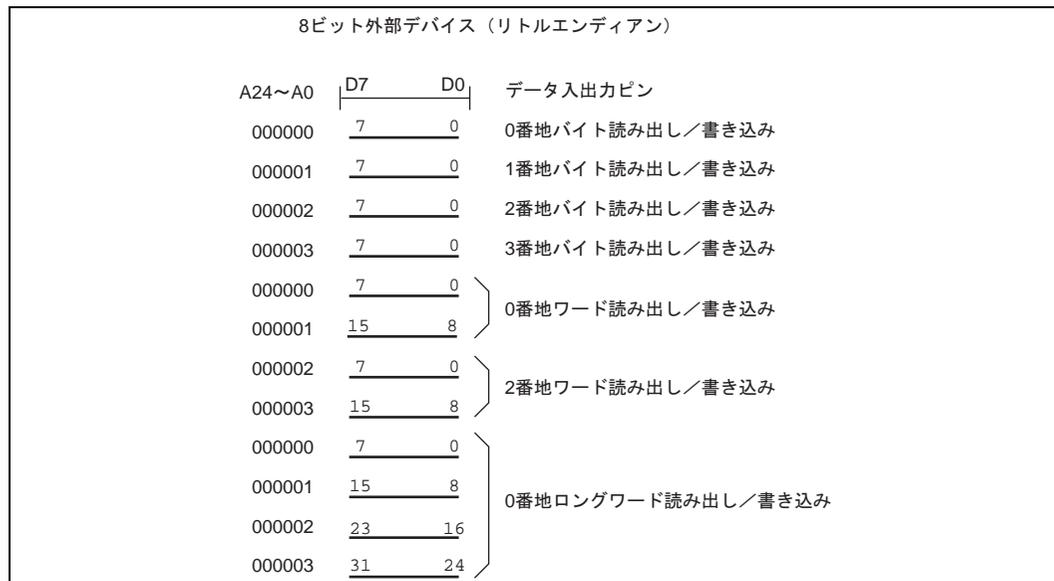


図 7.7 8ビット外部デバイスとアクセス単位の関係

### 7.4 通常空間アクセス

#### 7.4.1 基本タイミング

CS0 ~ CS4 の空間の通常空間アクセスは、主に SRAM の直結を考慮してストロープ信号を出力します。図 7.8 に通常空間アクセスの基本タイミングを示します。ウェイトのない通常アクセスは 2 サイクルで終了します。BS 信号はバスサイクルの開始を表すため、1 サイクルアサートされます。CS<sub>n</sub> 信号はネゲート期間を確保するために、T2 のクロック立ち下がりでネゲートされます。したがって最小ピッチでアクセスする場合にも、半サイクルのネゲート期間が生まれます。

アクセスサイズはリード時は指定がありません。アドレスの最下位ビットに正しいアクセス開始アドレスが出力されていますが、アクセスサイズの指定がないので、32 ビットデバイスでは 32 ビット、16 ビットデバイスでは 16 ビットを常に読み出すことになります。ライト時には書き込みを行うバイトの WE 信号のみがアサートされます。32 ビットデバイスでは WE3 が 4n 番地の書き込み指定であり、WE0 が 4n + 3 番地の書き込み指定です。16 ビットデバイスの場合には WE1 が 2n 番地の指定で、WE0 が 2n + 1 番地の指定です。8 ビットデバイスの場合には WE0 のみを使用します。

データバスにバッファを設ける場合にはリード方向のデータ出力を RD 信号を用いて行う必要があります。RD / WR 信号はアクセスを行っていないとき常時リード状態となっているため、これを用いて外付けデータバッファの制御を行うと出力が衝突する危険性があるので注意が必要です。

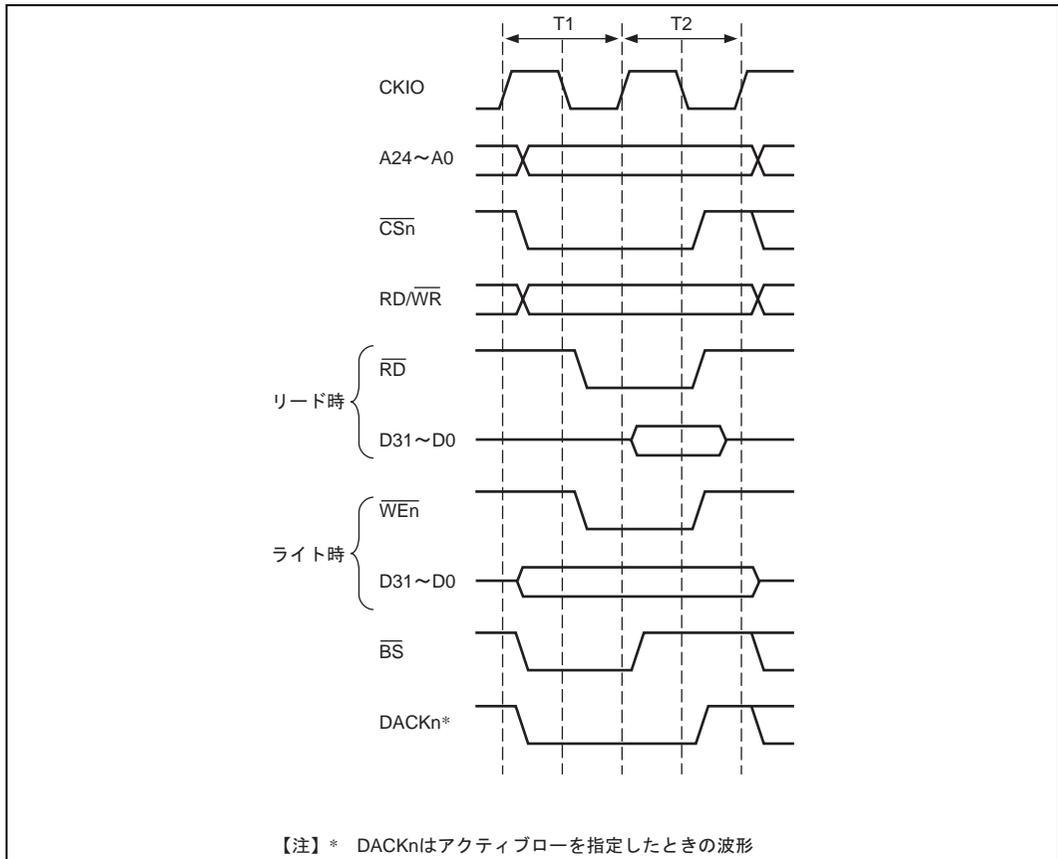


図 7.8 通常空間アクセスの基本タイミング

バスステートコントローラは、8ビットバス幅に対してワード、ロングワードでアクセスした場合、および16ビットバス幅に対してロングワードでアクセスした場合、複数回のアクセスを繰り返します。

クロック比が  $I : E = 1 : 1$  以外のときは、図 7.8 に示す基本タイミングを繰り返しますが、クロック比が  $I : E = 1 : 1$  の場合には、図 7.9 で示すように、 $\overline{CSn}$  のネゲート期間の無いバーストアクセスを行います。

7. バスステートコントローラ (BSC)

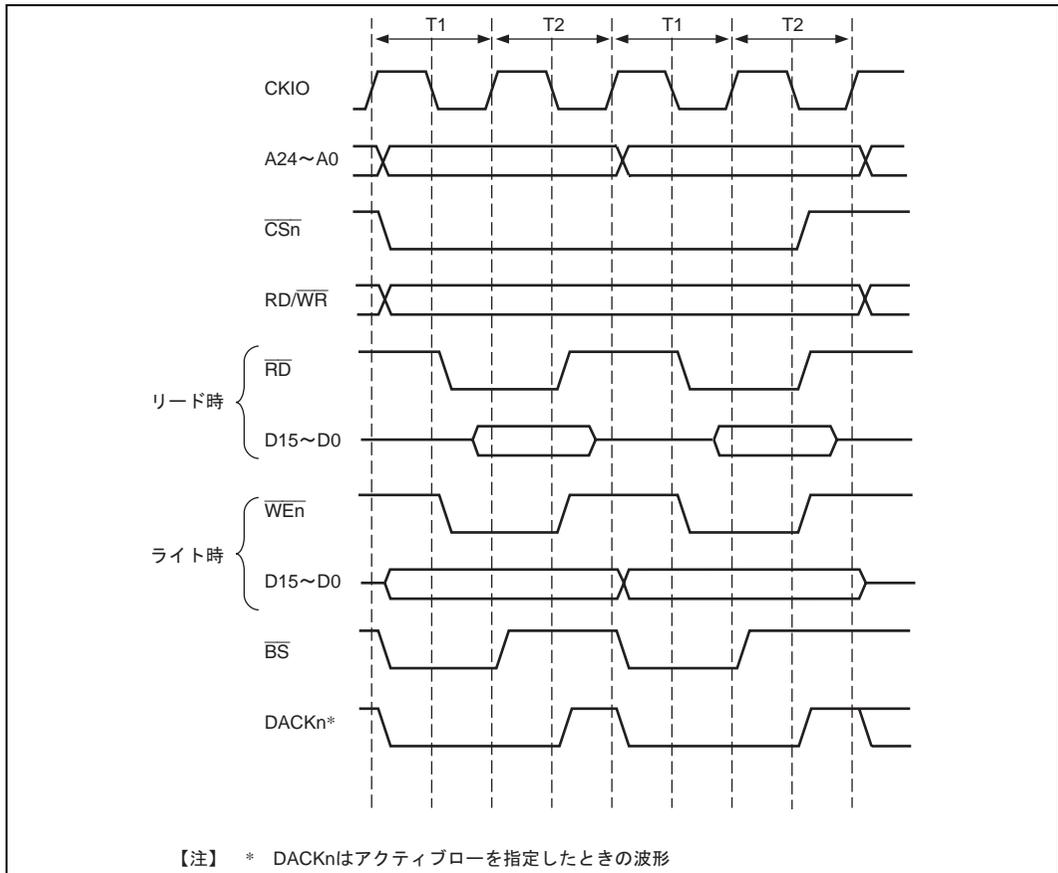


図 7.9 クロック比 I : E = 1 : 1 で、通常空間、16 ビットバス幅に対し、ロングワードアクセスした場合のタイミング

図 7.10 に 32 ビットデータ幅の SRAM との接続例を、図 7.11 に 16 ビットデータ幅の SRAM との接続例を、図 7.12 に 8 ビットデータ幅の SRAM との接続例を示します。

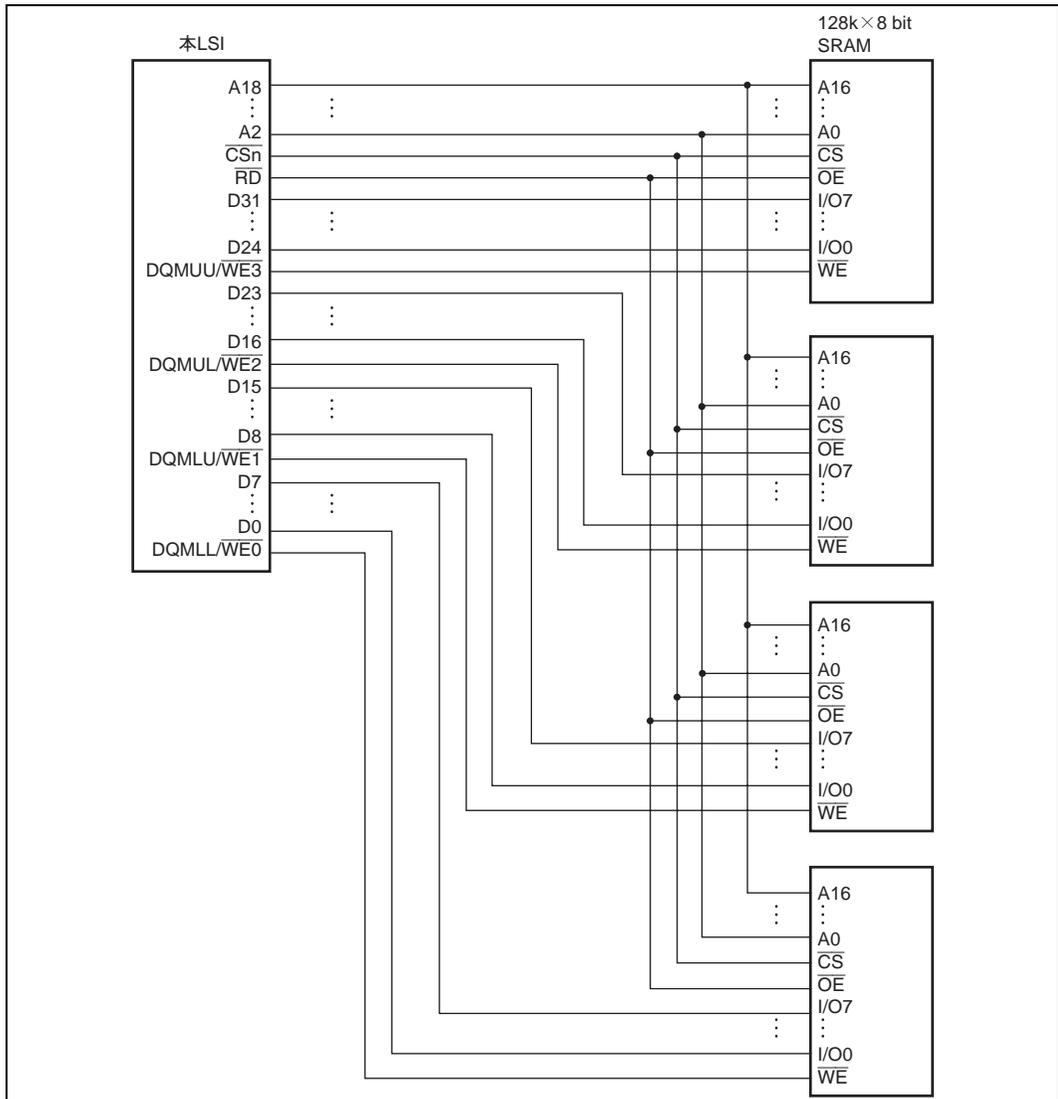


図 7.10 32 ビットデータ幅 SRAM 接続例

7. バスステートコントローラ (BSC)

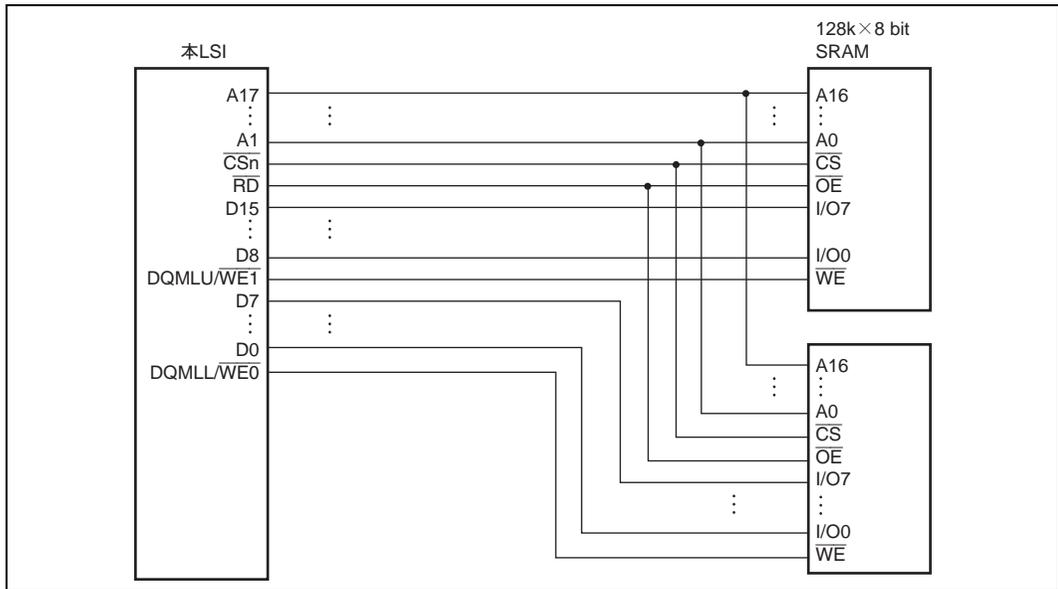


図 7.11 16 ビットデータ幅 SRAM 接続例

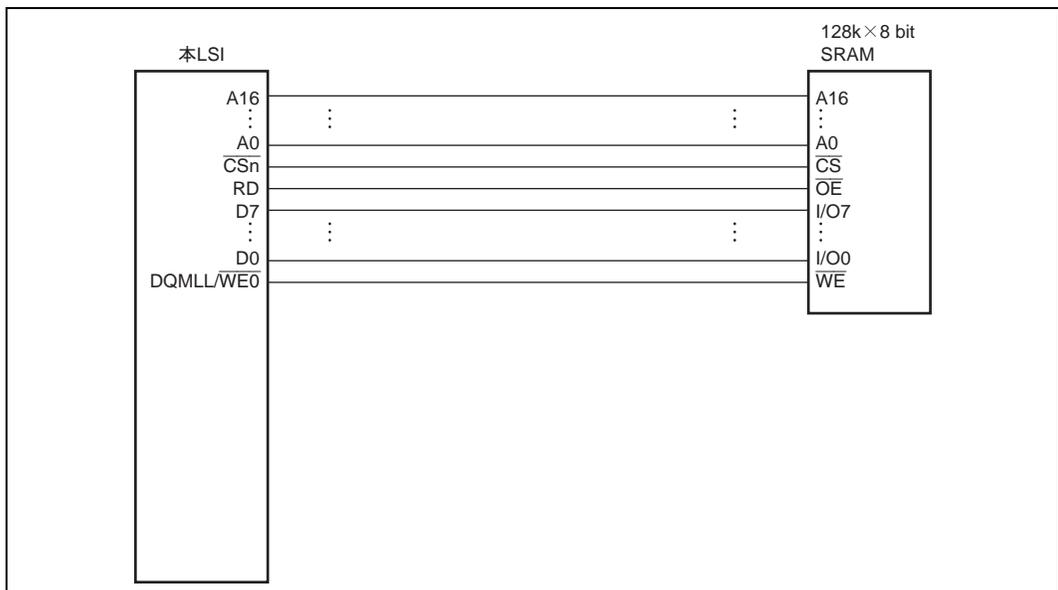


図 7.12 8 ビットデータ幅 SRAM 接続例

## 7.4.2 ウェイトステート制御

WCR1、WCR2、BCR1、BCR3 の設定により、通常空間アクセスのウェイトステートの挿入を制御できます。WCR1、WCR2 の各 CS 空間に対応するウェイト指定ビット 2 ビット ( $W_{n1}$  と  $W_{n0}$ ) の組み合わせが 01 および 10 のときは、このウェイト指定に従ったソフトウェアウェイトが挿入されます。 $W_{n1}$  と  $W_{n0}$  が 11 の時には、BCR1、BCR3 のロングウェイト指定ビット  $AnLW$  の指定に従ってウェイトサイクルの挿入を行います。BCR1、BCR3 のロングウェイト指定は、CS0、CS1 および CS4 空間ではそれぞれ独立に指定ができますが、CS2 と CS3 空間では同じ値が指定されます。ただし、WCR1 の指定はそれぞれ独立しています。WCR1、WCR2、BCR1、BCR3 によって、図 7.13 に示す通常アクセス空間のウェイトタイミングで、 $T_w$  のサイクルがウェイトサイクルとして指定サイクル数だけ挿入されます。

各 CS 空間での  $T_w$  を指定する制御ビット名を表 7.5 に示します。

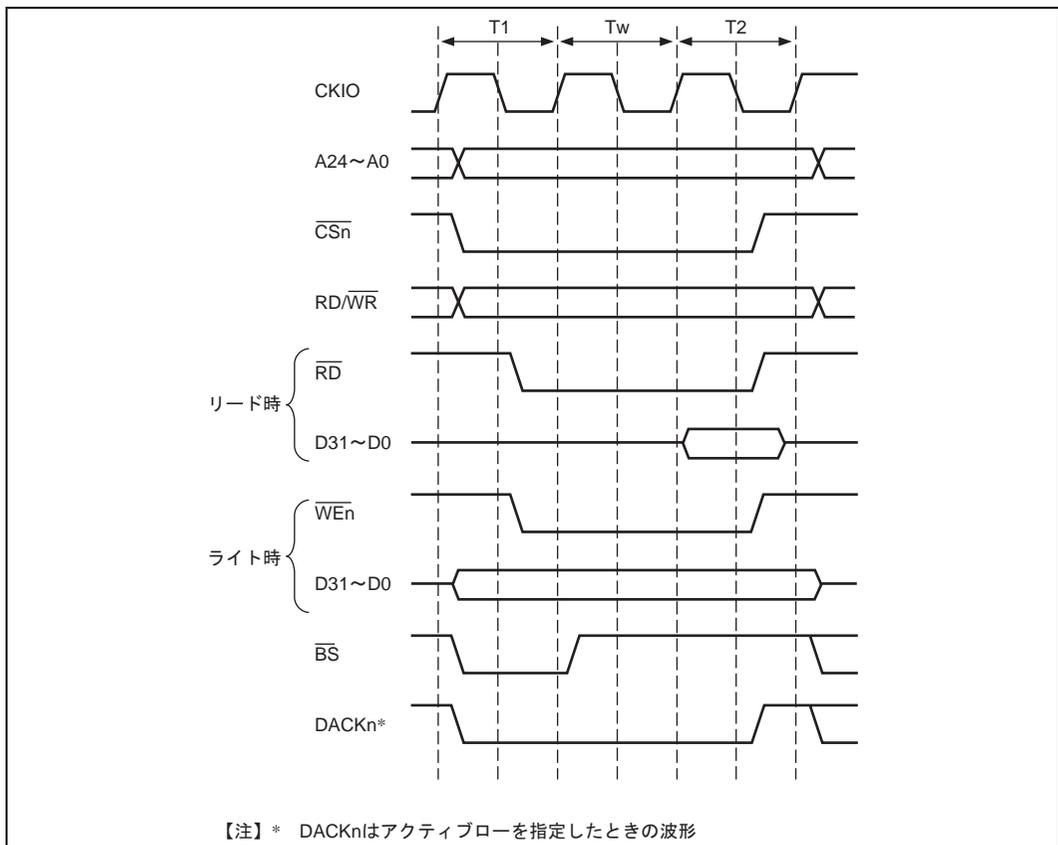


図 7.13 通常空間アクセスのウェイトタイミング (ソフトウェアウェイトのみ)

## 7. バスステートコントローラ (BSC)

---

表 7.5  $\overline{CSn}$  空間の Tw 指定ビット

	BCR3	BCR1		WCR1		WCR2		Tw
CS0	A0LW2	A0LW1	A0LW0	W01	W00	-	-	0 ~ 14
CS1	A1LW2	A1LW1	A2LW0	W11	W10	-	-	0 ~ 14
CS2	AHLW2	AHLW1	AHLW0	W21	W20	-	-	0 ~ 14
CS3	AHLW2	AHLW1	AHLW0	W31	W30	-	-	0 ~ 14
CS4	A4LW2	A4LW1	A4LW0	-	-	W41	W40	0 ~ 14

WCR1、WCR2 (Wn1、Wn0) によってソフトウェアによるウェイトを指定し、かつ、WCR2 の外部ウェイトマスクビット (AnWM) を 0 としたときには、外部からのウェイト入力  $\overline{WAIT}$  信号もサンプリングされます。 $\overline{WAIT}$  信号のサンプリングを図 7.14 に示します。ソフトウェアウェイトとして 2 サイクルのウェイトを指定しています。サンプリングは Tw ステートから T2 ステートに移行する際に行われるので、T1 のサイクルおよび 1 回目の Tw サイクルで  $\overline{WAIT}$  信号をアサートしても、なんら影響を与えません。 $\overline{WAIT}$  信号はクロックの立ち下がり でサンプリングされます。

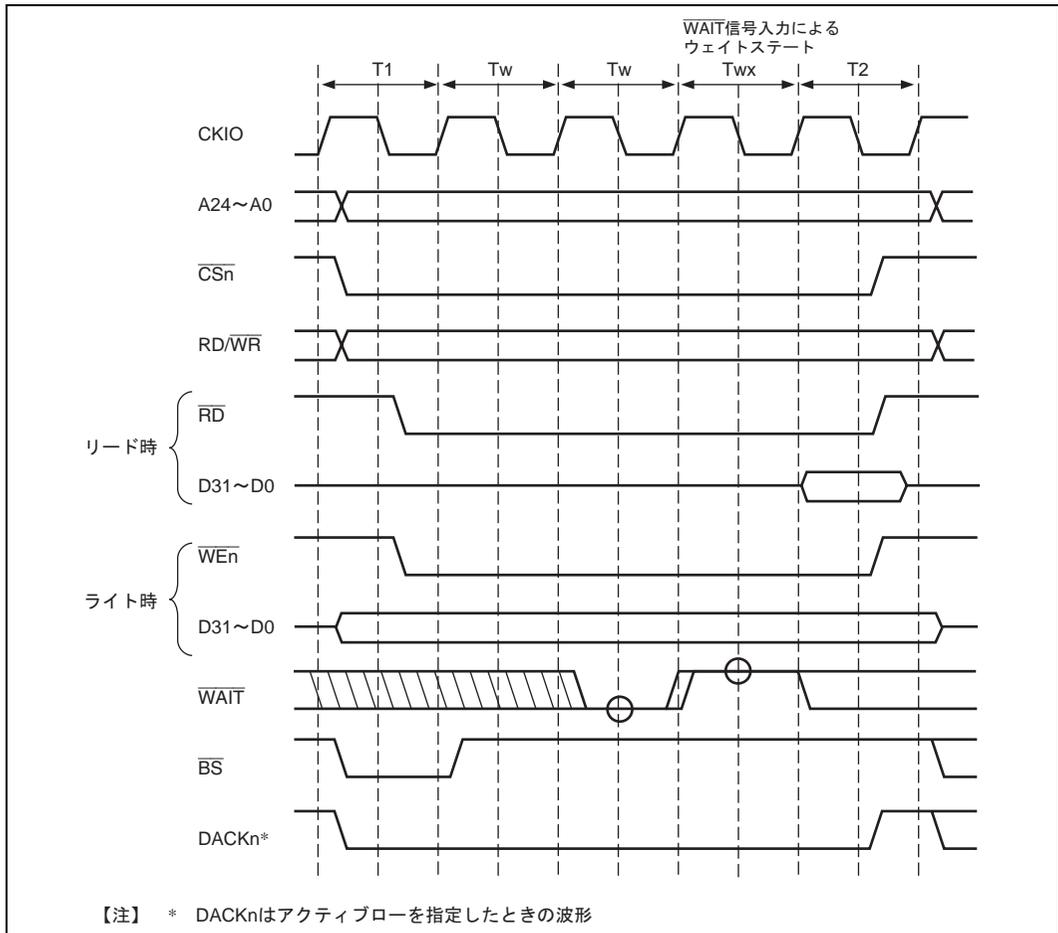


図 7.14 通常空間アクセスのウェイトステートタイミング  
(WAIT 信号によるウェイトステート)

CS0 ~ CS3 空間では、図 7.14 で示すように、外部ウェイト信号のネゲートを受け付け後、1 サイクルで  $\overline{CS}$ 、 $\overline{RD}$ 、 $\overline{WE}_n$  をネゲートします。CS4 空間では、外部ウェイトネゲート受け付け後、 $\overline{CS}$ 、 $\overline{RD}$ 、 $\overline{WE}_n$  をネゲートするまでのサイクルを WCR2 の A4WD1、A4WD0 により 1、2、4 に設定できます。例を図 7.15 に示します。

7. バスステートコントローラ (BSC)

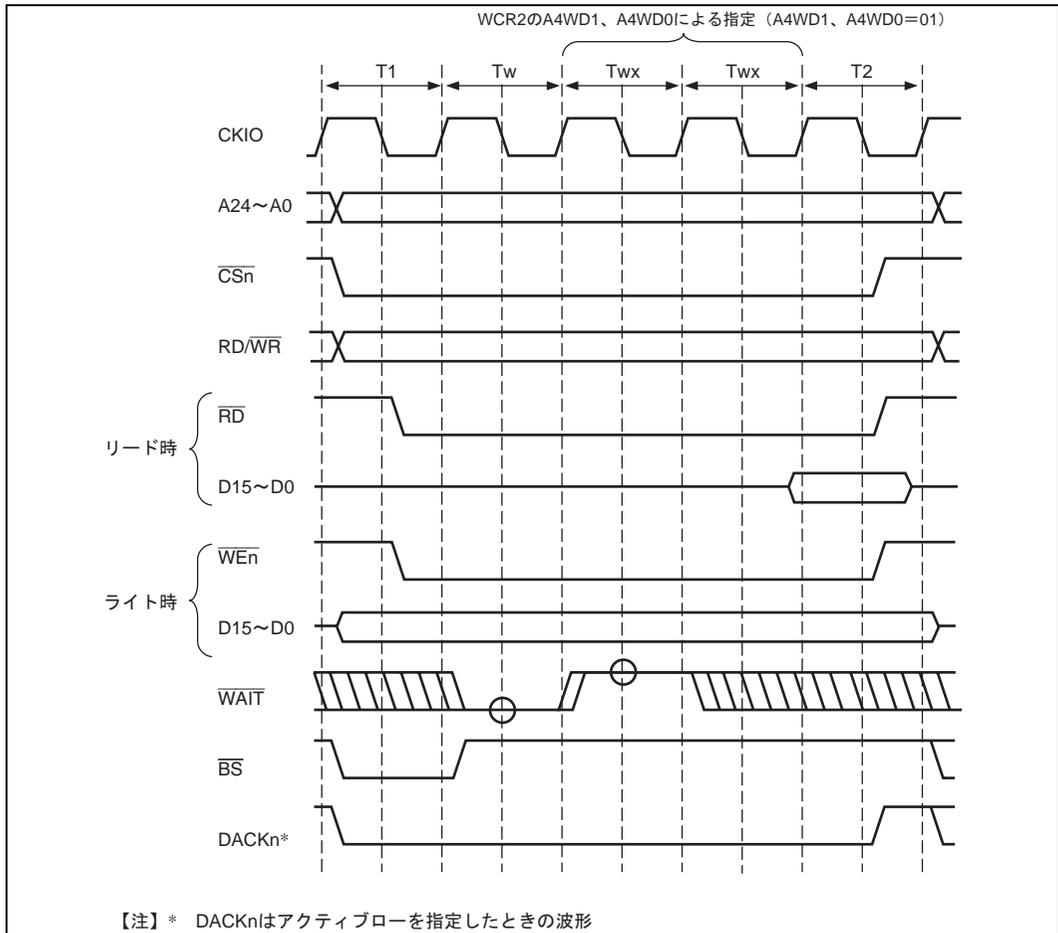
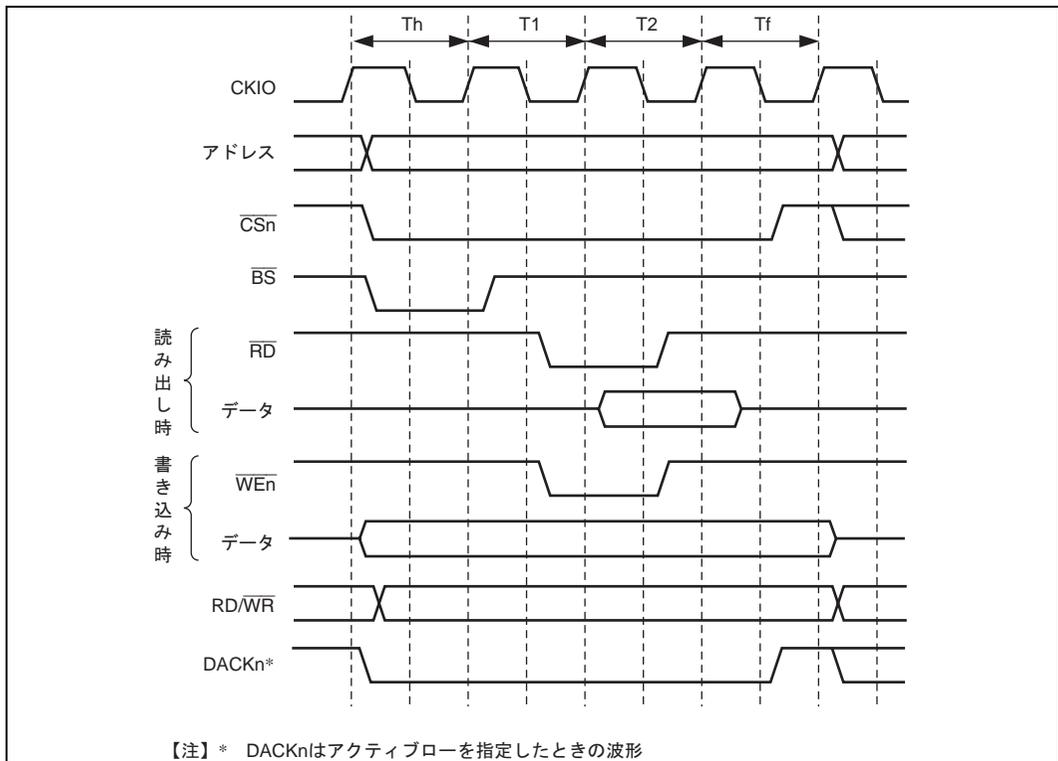


図 7.15 CS4 空間における通常空間アクセスのウェイトスタートタイミング

7.4.3  $\overline{CS}$  アサート期間拡張

WCR3 の制御ビットの設定により、 $\overline{RD}$ 、 $\overline{WE_n}$  のアサート期間が  $\overline{CS_n}$  のアサート期間からはみ出さないようにアイドルサイクルを挿入することができます。これにより、外付け回路とフレキシブルなインタフェースがとれます。タイミングを図 7.16 に示します。Th および Tf サイクルが通常サイクルの前と後ろにそれぞれ付加されています。このサイクルでは  $\overline{RD}$ 、 $\overline{WE_n}$  以外はアサートされますが、 $\overline{RD}$ 、 $\overline{WE_n}$  はアサートされません。また、データは Tf サイクルまで延ばされるので、書き込み動作の遅いデバイスなどに有効です。

図 7.16  $\overline{CS}$  アサート期間拡張機能

Th、Tf は下記のように設定できます。

	Th	Tf	WCR3
CS0 ~ 3	0 ~ 2		AnSW1、AnSW0 (Th=Tf) n=0 ~ 3
CS4	0 ~ 7	0 ~ 5	Th : A4SW2 ~ 0 Tf : A4HW1 ~ 0

## 7.5 シンクロナス DRAM インタフェース

### 7.5.1 シンクロナス DRAM 直結方式

シンクロナス DRAM として本 LSI に直接接続可能なものは、2M ビットの 128k × 16 品、4M ビットの 256k × 16 品、16M ビットの 1M × 16 品、2M × 8 品、4M × 4 品、64M ビットの 4M × 16 品、および 8M × 8 ビット品の 7 種類です。64M ビット品に対しては、内部が 2 つあるいは 4 つのバンクに分割されているものに対応しています。64M ビット品以外ではそれぞれ内部が 2 つのバンクに分割されているものに対応しています。シンクロナス DRAM は  $\overline{CS}$  信号によって選択ができるため、 $\overline{RAS}$  等の制御信号を共通に使用して CS2 空間と CS3 空間に接続が可能です。BCR1 の DRAM 他メモリーネーブルビット (DRAM 2 ~ 0) を 001 に設定すると、CS2 空間が通常空間、CS3 空間がシンクロナス DRAM 空間になり、100 に設定すると CS2 空間がシンクロナス DRAM 空間、CS3 空間が通常空間になり、101 に設定すると CS2 空間、CS3 空間がともにシンクロナス DRAM 空間となります。

シンクロナス DRAM の動作モードとしては、バーストリード/シングルライト (初期値) のモード、およびバーストリード/バーストライトのモードをサポートしています。バースト長はデータバス幅に依存し、16 ビット幅の時には 8 バースト、32 ビット幅の時には 4 バーストを使用します。データのバス幅は MCR のサイズビット SZ によって指定されます。必ずバースト動作を行うので、MCR のバーストイネーブルビット BE は無視されます。バーストライトモードへの切り替えは、BCR3 の BWE ビットによって行います。

シンクロナス DRAM を直結するための制御信号は  $\overline{RAS}$ 、 $\overline{CAS}$  /  $\overline{OE}$ 、 $\overline{RD}$  /  $\overline{WR}$ 、 $\overline{CS2}$  または  $\overline{CS3}$ 、DQM<sub>UU</sub>、DQM<sub>UL</sub>、DQM<sub>LU</sub>、DQM<sub>LL</sub> および CKE 信号です。 $\overline{CS2}$  または  $\overline{CS3}$  を除く信号は各エリア共通であり、CKE を除く信号は  $\overline{CS2}$  または  $\overline{CS3}$  が真のときのみ有効となり取り込まれます。したがって、複数のエリアにシンクロナス DRAM を並列に接続することができます。CKE はセルフリフレッシュを行うときのみアサート (L レベルに) され、それ以外は常にネゲート (H レベルに) されています。

$\overline{RAS}$ 、 $\overline{CAS}$  /  $\overline{OE}$ 、 $\overline{RD}$  /  $\overline{WR}$ 、および特定のアドレス信号によって、シンクロナス DRAM に対するコマンドが指定されます。コマンドには、NOP、オートリフレッシュ (REF)、セルフリフレッシュ (SELF)、全バンクプリチャージ (PALL)、指定バンクプリチャージ (PRE)、ロウアドレスストロブ・バンクアクティブ (ACTV)、リード (READ)、プリチャージ付きリード (READA)、ライト (WRIT)、プリチャージ付きライト (WRITA)、モードレジスタ書き込み (MRS) があります。

バイトの指定は DQM<sub>UU</sub>、DQM<sub>UL</sub>、DQM<sub>LU</sub>、DQM<sub>LL</sub> によって行われます。該当する DQM が L のバイトに対して読み出し書き込みが行われます。32 ビットデータ幅の場合、DQM<sub>UU</sub> は 4n 番地のアクセスを、DQM<sub>LL</sub> は 4n + 3 番地のアクセスを指定します。16 ビット幅の場合は DQM<sub>LU</sub>、DQM<sub>LL</sub> のみが使われます。

図 7.17 に 256k × 16 ビットのシンクロナス DRAM を用い 32 ビット幅で接続する場合の例を示します。図 7.18 には 16 ビット幅で接続する場合の例を示します。

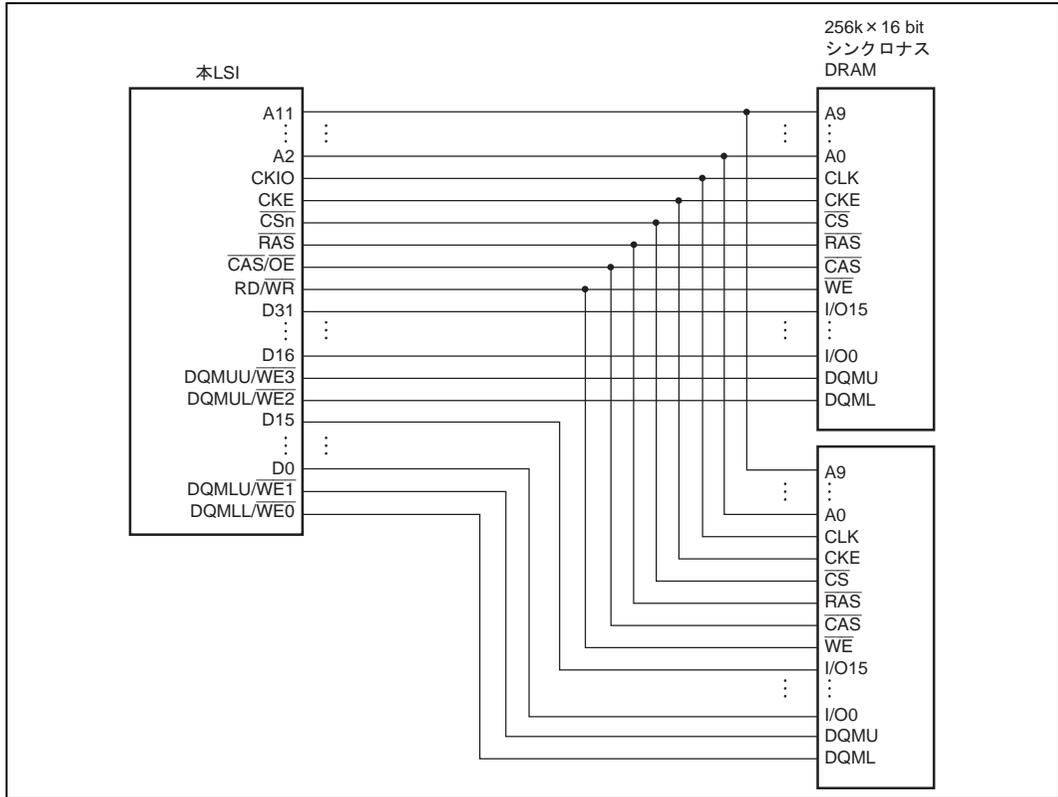


図 7.17 シンクロナス DRAM32 ビットデバイス幅接続例

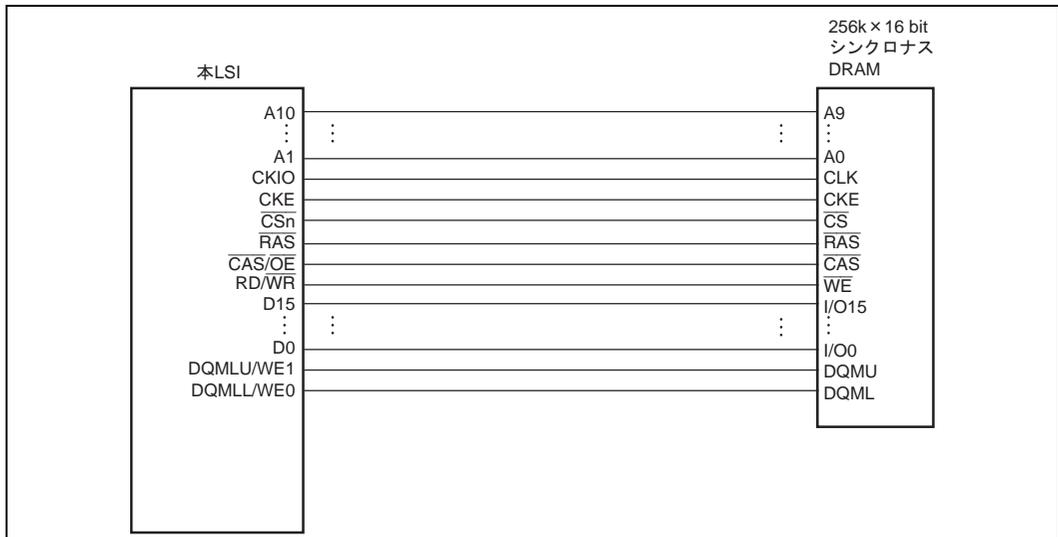


図 7.18 シンクロナス DRAM16 ビットデバイス幅接続例

### 7.5.2 アドレスマルチプレクス

MCR のアドレスマルチプレクス指定 AMX2 ~ AMX0 とサイズ指定 SZ にしたがって、外付けのマルチプレクス回路なしにシンクロナス DRAM に接続できるように、アドレスの

マルチプレクスを行います。表 7.6 にマルチプレクスの指定ビットとアドレス端子に出力されるビットの関係を示します。

A24 ~ A16 はマルチプレクスを行わず常に本来の値が出力されています。

SZ=0 のときは、シンクロナス DRAM 側のデータ幅は 16 ビットであり、デバイスのアドレス端子の LSB である A0 はワードアドレスの指定を行います。したがって、シンクロナス DRAM の A0 端子を本 LSI の A1 端子に接続し、以下 A1 端子を A2 端子にという順で接続してください。

SZ=1 のときは、シンクロナス DRAM 側のデータ幅は 32 ビットであり、デバイスのアドレス端子の LSB である A0 はロングワードアドレスの指定を行います。したがって、シンクロナス DRAM の A0 を本 LSI の A2 端子に接続し、以下 A1 端子を A3 端子にという順で接続してください。

表 7.6 SZ、AMX とアドレスマルチプレクス出力の関係

設定				出カタイミング	外部アドレス端子							
SZ	AMX2	AMX1	AMX0		A1 ~ A8	A9	A10	A11	A12	A13	A14	A15
1	0	0	0	カラムアドレス	A1 ~ A8	A9	A10	A11	L/H*1	A21*2	A14	A15
				ロウアドレス	A9 ~ A16	A17	A18	A19	A20	A21*2	A22	A23
1	0	0	1	カラムアドレス	A1 ~ A8	A9	A10	A11	L/H*1	A22*2	A14	A15
				ロウアドレス	A10 ~ A17	A18	A19	A20	A21	A22*2	A23	A24
1	0	1	0	カラムアドレス	A1 ~ A8	A9	A10	A11	L/H*1	A23*2	A14	A15
				ロウアドレス	A11 ~ A18	A19	A20	A21	A22	A23*2	A24	A25
1	0	1	1	カラムアドレス	A1 ~ A8	A9	L/H*1	A19*2	A12	A13	A14	A15
				ロウアドレス	A9 ~ A16	A17	A18	A19*2	A20	A21	A22	A23
1	1	0	0	カラムアドレス	A1 ~ A8	A9	A10	A11	L/H*1	A13	A22*3	A23*2
				ロウアドレス	A9 ~ A16	A17	A18	A19	A20	A21	A22*3	A23*2
1	1	0	1	カラムアドレス	A1 ~ A8	A9	A10	A11	L/H*1	A13	A23*3	A24*2
				ロウアドレス	A10 ~ A17	A18	A19	A20	A21	A22	A23*3	A24*2
1	1	1	1	カラムアドレス	A1 ~ A8	A9	L/H*1	A18*2	A12	A13	A14	A15
				ロウアドレス	A9 ~ A16	A17	A17	A18*2	A20	A21	A22	A23
0	0	0	0	カラムアドレス	A1 ~ A8	A9	A10	L/H*1	A20*2	A13	A14	A15
				ロウアドレス	A9 ~ A16	A17	A18	A19	A20*2	A21	A22	A23
0	1	0	0	カラムアドレス	A1 ~ A8	A9	A10	L/H*1	A12	A21*3	A22*2	A15
				ロウアドレス	A9 ~ A16	A17	A18	A19	A20	A21*3	A22*2	A23
0	0	1	1	カラムアドレス	A1 ~ A8	L/H*1	A18*2	A11	A12	A13	A14	A15
				ロウアドレス	A9 ~ A16	A17	A18*2	A19	A20	A21	A22	A23
0	1	1	1	カラムアドレス	A1 ~ A8	L/H*1	A17*2	A11	A12	A13	A14	A15
				ロウアドレス	A9 ~ A16	A16	A17*2	A19	A20	A21	A22	A23

- 【注】
- ・ AMX 2 ~ AMX0 = 110 の設定は予約となっていますので設定しないでください。
  - ・ SZ=0 のとき AMX 2 ~ AMX0 = 001, 010, 101 の設定も予約となっていますので設定しないでください。
  - \*1 L/H はコマンド指定に使われるビットであり、アクセスのモードによって L または H に固定されます。
  - \*2 バンクアドレス指定
  - \*3 4 バンク設定時、バンクアドレス指定

### 7.5.3 バーストリード

バーストリード時のタイミングチャートを図 7.19 (a) (b) に示します。以下の例では 256k × 16 ビットのシンクロナス DRAM を 2 個接続し、データ幅 32 ビットで使用した場合を想定しており、バースト長は 4 となっています。ACTV コマンド出力を行う Tr サイクルに続いて、READA コマンドを Tc サイクルに発行し、Td1 から Td4 のサイクルでリードデータを受け取り、Tde サイクルでリードシーケンス終了を待ちます。Tde サイクルは、I :E =1:1 以外のときに 1 サイクル、I :E =1:1 のときには 2 サイクル発生します。Tap はシンクロナス DRAM 内部で READA コマンドに基づくオートプリチャージが完了するのを待つサイクルであり、この間は同一バンクに対して新たなアクセスコマンドの発行は行えません。ただし、別の CS 空間や同じシンクロナス DRAM で、もう一方のバンクに対するアクセスは可能です。本 LSI では MCR の TRP1、TRP0 の指定によって Tap のサイクル数を決定し、この間同一バンクに対するコマンド発行を行いません。

## 7. バスステートコントローラ (BSC)

---

図 7.19 の例は基本サイクルを表したものです。より低速なシンクロナス DRAM を接続するため、WCR1 および MCR のビットを設定することによって、サイクルを延ばすことができます。ACTV コマンド出力サイクル  $T_r$  から READA コマンド出力サイクル  $T_c$  までのサイクル数は、MCR の RCD1、RCD0 ビットによって指定することができます。00 のとき 1 サイクル、01 のとき 2 サイクル、10 のとき 3 サイクルとなります。2 サイクル以上の場合、 $T_r$  サイクルと  $T_c$  サイクルの間にシンクロナス DRAM に対する NOP コマンド発行サイクル  $T_{rw}$  が挿入されます。READA コマンド出力サイクル  $T_c$  から最初のリードデータ取り込みサイクル  $T_{d1}$  までのサイクル数は、WCR1 の W21、W20 および W31、W30 によって、1 サイクルから 4 サイクルまで指定することができます。このサイクル数はシンクロナス DRAM の CAS レイテンシサイクル数に相当します。2 サイクル以上の場合には  $T_c$  サイクルと  $T_{d1}$  サイクルの間に NOP コマンド発行サイクル  $T_w$  が挿入されます。プリチャージ完了待ちサイクル  $T_{ap}$  のサイクル数は、MCR の TRP1、TRP0 ビットによって指定され、CAS レイテンシが 1 のときには TRP1、TRP0 で指定された  $T_{ap}$  サイクルが発生します。CAS レイテンシが 2 以上のときには、TRP の指定 - 1 サイクルの  $T_{ap}$  サイクルが発生します。 $T_{ap}$  サイクルの間は同一バンクに対するコマンドは、NOP を除いて発行されません。RCD1、RCD0 を 01 に、W31、W30 を 01 に、TRP1、TRP0 を 01 に設定したときのバーストリードのタイミングを図 7.20 (a)、(b) に示します。

データ幅が 16 ビットの場合には、16 バイトのデータを転送するため、8 バーストサイクル必要となります。データ取り込みのサイクルは  $T_{d1}$  から  $T_{d8}$  となります。

シンクロナス DRAM の CAS レイテンシは 3 サイクルまでですが、バスステートコントローラの CAS レイテンシは 4 まで指定できます。これは本 LSI とシンクロナス DRAM の間にラッチを含む回路を設けて互いを接続するためのものです。

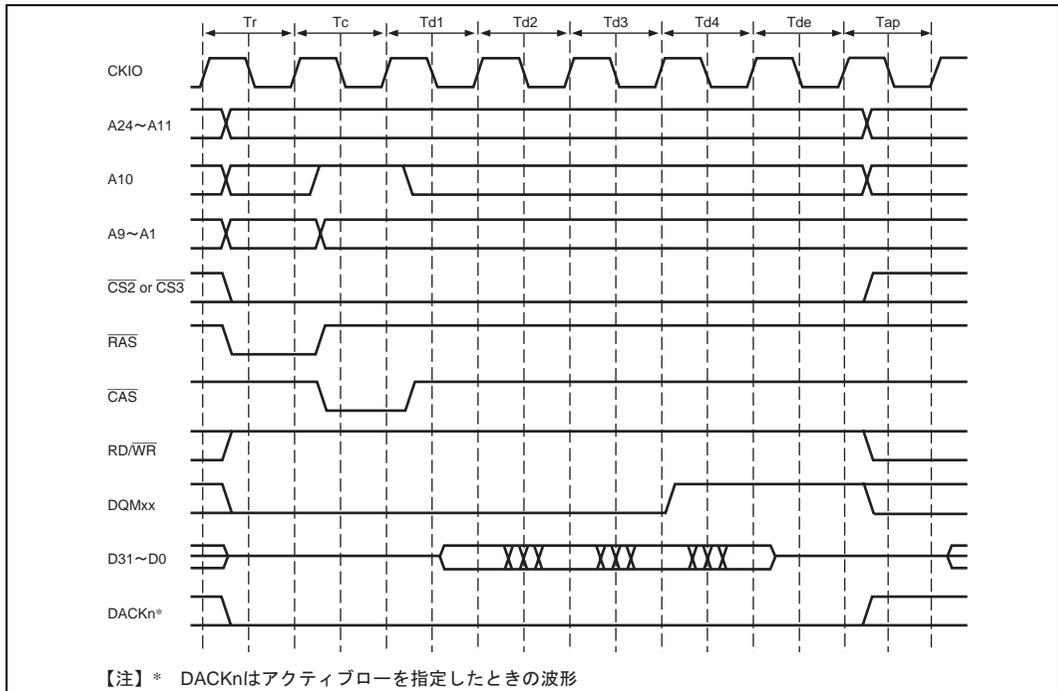


図 7.19 (a) バーストリード基本タイミング (オートプリチャージ) I : E = 1 : 1 以外

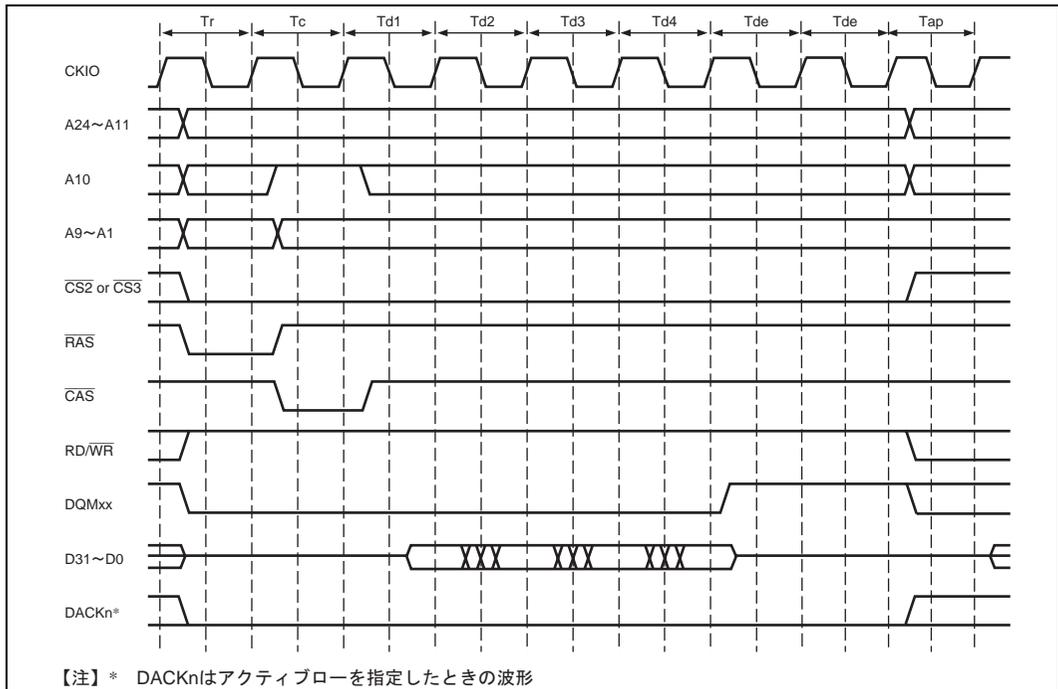


図 7.19 (b) バーストリード基本タイミング (オートプリチャージ) I : E = 1 : 1

## 7. バスステートコントローラ (BSC)

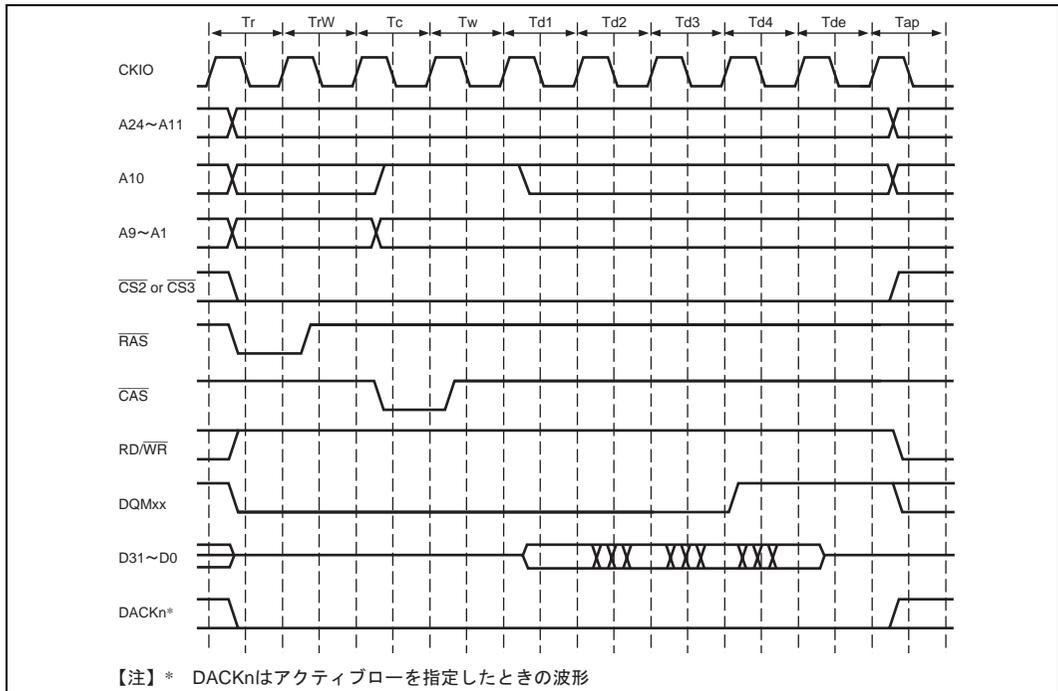


図 7.20 (a) バーストリードウェイト指定タイミング (オートプリチャージ) | : E = 1 : 1 以外

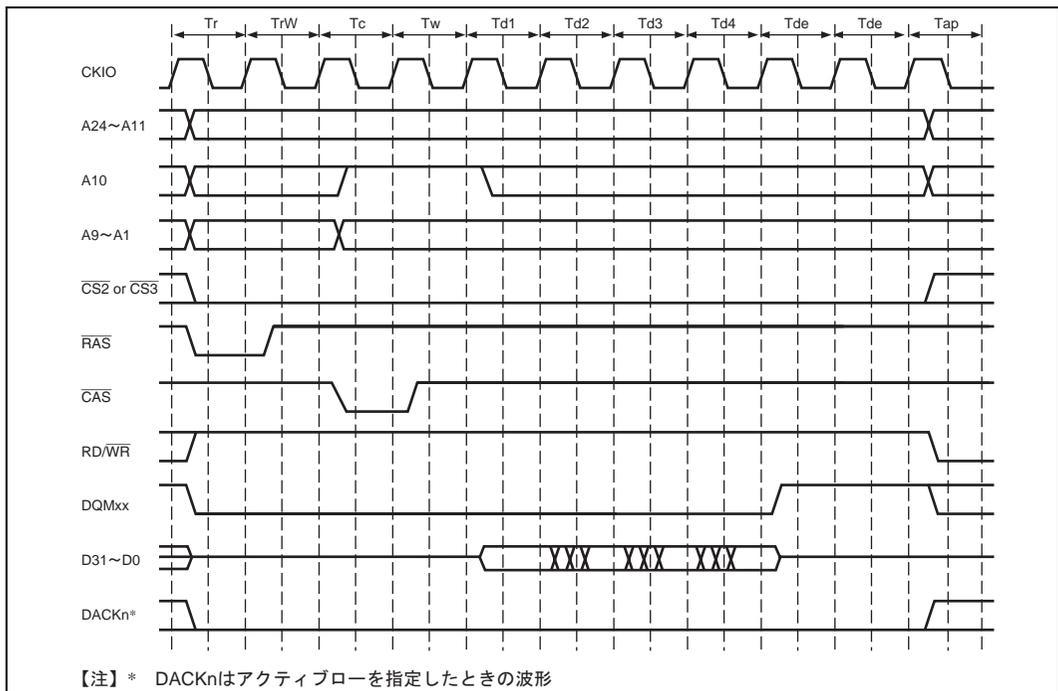


図 7.20 (b) バーストリードウェイト指定タイミング (オートプリチャージ) | : E = 1 : 1

### 7.5.4 シングルリード

キャッシュ領域をアクセスし、キャッシュがミスしたときのキャッシュフィルサイクルは 16 バイト単位で行われるため、バーストリードで読み出されたデータはすべて有効に使用されます。これに対し、キャッシュスルー領域をアクセスする場合には、必要なデータは最長でも 32 ビットであり、残りの 12 バイトは無駄となります。DMAC による DMA 転送でもシンクロナス DRAM をソースに指定し、転送単位に 16 バイト以外を指定すると同様に無駄なデータアクセスが生じます。図 7.21 (a) (b) に単一アドレスのリードを行う場合のタイミングを示します。シンクロナス DRAM は、バーストリードのモードに設定しているため、必要なデータを受け取った後も読み出しデータの出力が続けられます。データの衝突を避けるため、Td1 で必要なデータの読み込みを行ったあと、Td2 から Td4 の空読みサイクルを行い、シンクロナス DRAM の動作終了を待ちます。

データ幅が 16 ビットの場合、読み出し時のバースト転送数は 8 となります。キャッシュスルーおよびその他の DMA リードサイクルでは Td1 から Td8 の 8 サイクルのうち、ロングワードアクセスでは Td1 と Td2 サイクルのみ、ワードまたはバイトアクセスでは Td1 サイクルでのみデータが取り込まれます。

このような空サイクルは、メモリアクセスタイムの増加をもたらし、プログラム実行速度の低下、DMA 転送速度の低下を招くので、不要なキャッシュスルー領域のアクセスを避けるとともに、シンクロナス DRAM をソースに指定した DMA 転送を行う場合、データを 16 バイト境界に配置して 16 バイト単位の転送ができるようなデータ構造を採用することが重要です。

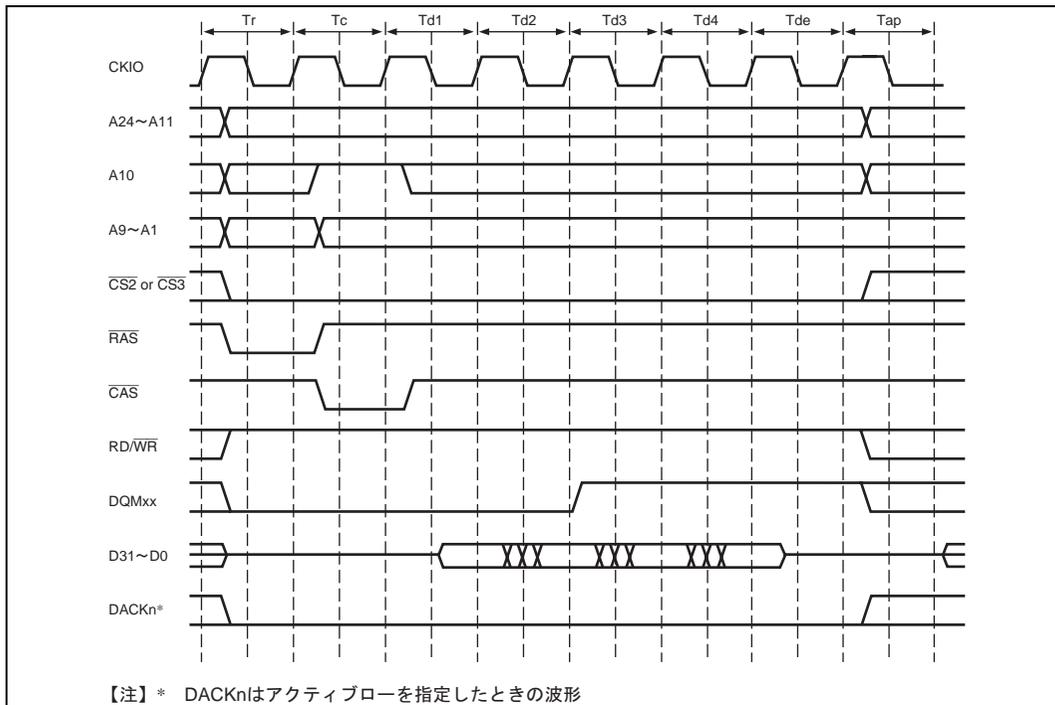


図 7.21 (a) シングルリードタイミング (オートプリチャージ) | : E = 1 : 1 以外

7. バスステートコントローラ (BSC)

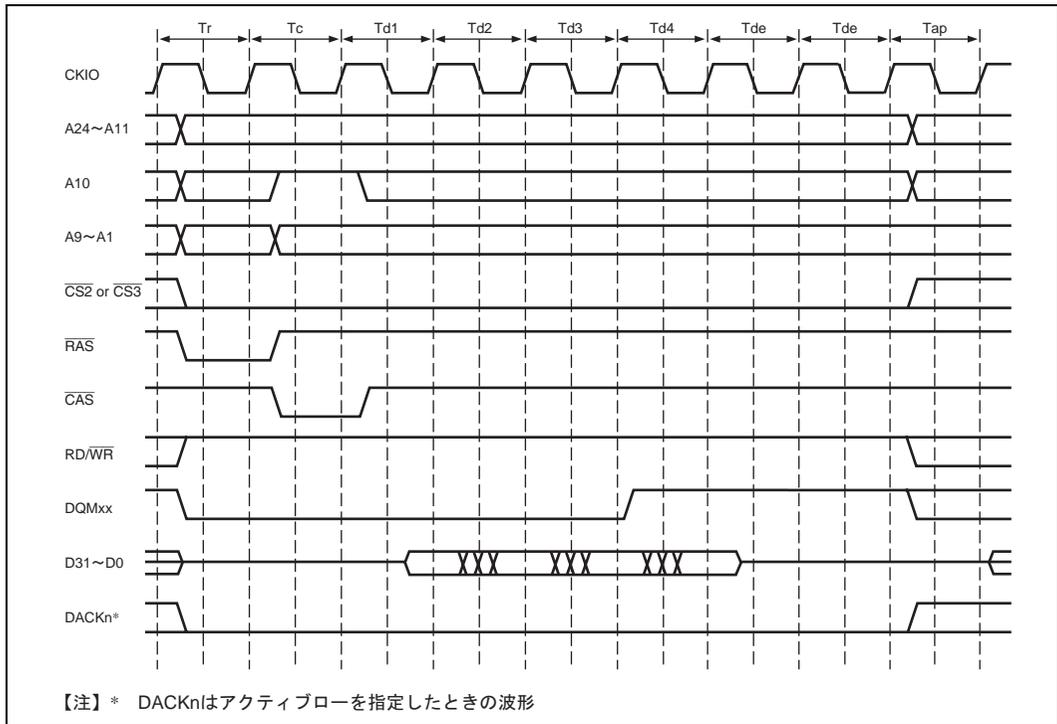


図 7.21 (b) シングルリードタイミング (オートプリチャージ) I : E = 1 : 1

### 7.5.5 シングルライトモード

シンクロナス DRAM の書き込みは、BCR3 の BWE ビットの指定によって、シングルライトモードあるいは、バーストライトモードで書き込みを行います。シングルライトモードアクセスの基本タイミングチャートを図 7.22 に示します。ACTV コマンド  $T_r$  について、オートプリチャージを行う WRITA コマンドを  $T_c$  で発行します。ライトサイクルでは、ライトデータはライトコマンドと同時に出力します。オートプリチャージ付きライトの場合、シンクロナス DRAM の内部では、ライトコマンドの完了後当該バンクのプリチャージを行うので、プリチャージ完了まで同一バンクに対するコマンド発行は行えません。このため、リードアクセス時のプリチャージ待ちサイクル  $T_{ap}$  に加えライトコマンド後、プリチャージが起動されるまでの時間を待つ  $T_{rwl}$  サイクルが加わり、この間同一のバンクに対する新たなコマンドの発行を遅らせます。 $T_{rwl}$  サイクルのサイクル数も指定可能で、MCR の  $TRWL1$ 、 $TRWL0$  ビットによって指定を行います。

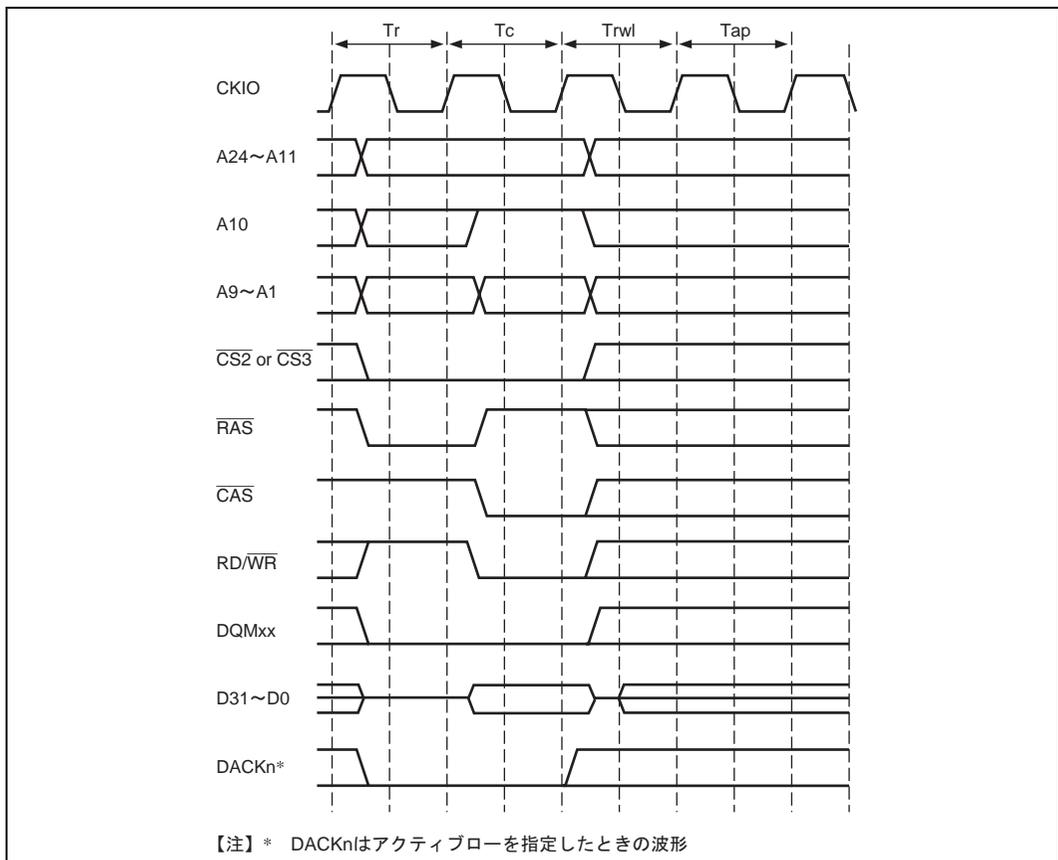


図 7.22 シングルライトモード基本タイミング (オートプリチャージ)

### 7.5.6 バーストライトモード

BCR3 の BWE ビットを 1 に設定することにより、バーストライトモードに設定することができます。バーストライトアクセスの基本タイミングチャートを図 7.23 (a)、(b) に示します。例では、バス幅 32 ビットを想定

## 7. バスステートコントローラ (BSC)

しており、バースト長は4となっています。バーストライトサイクルでは、ACTV コマンド Tr についてオートプリチャージを行う WRITA コマンドを Tc1 で発行します。最初の4バイトのライトデータは、Tc1 で WRITA コマンドと同時に出力され、残りの12バイトのデータは、Tc2、Tc3、Tc4 で続いて出力されます。オートプリチャージ付きライトの場合、シングルライトと同様、ライトデータ出力後に、プリチャージが起動されるまでの時間を待つ Trwl サイクル、続いて、ライトアクセス時のプリチャージ待ちサイクル Tap が挿入されます。Trwl サイクル、Tap サイクルは、それぞれ MCR の TRWL1、TRWL0 あるいは TRP1、TRP0 によって設定可能です。

バーストライトモード時、単発のライトを行うと、SDRAM の設定はバースト長4に設定されていますので Tc1 でデータを書き込んだ後、Tc2、Tc3、Tc4 サイクルでは、DQMxx 信号を H レベルにすることにより、空ライトを行います。

このような空サイクルはメモリアクセスタイムの増加をもたらし、プログラム実行速度の低下、DMA 転送速度の低下を招きます。そこで、 unnecessary キャッシュスルー領域のアクセスを避けるとともに、キャッシュの設定をコピーバックにしてください。また、DMA 転送は16バイト単位の転送ができるようなデータ構造を採用することが重要です。

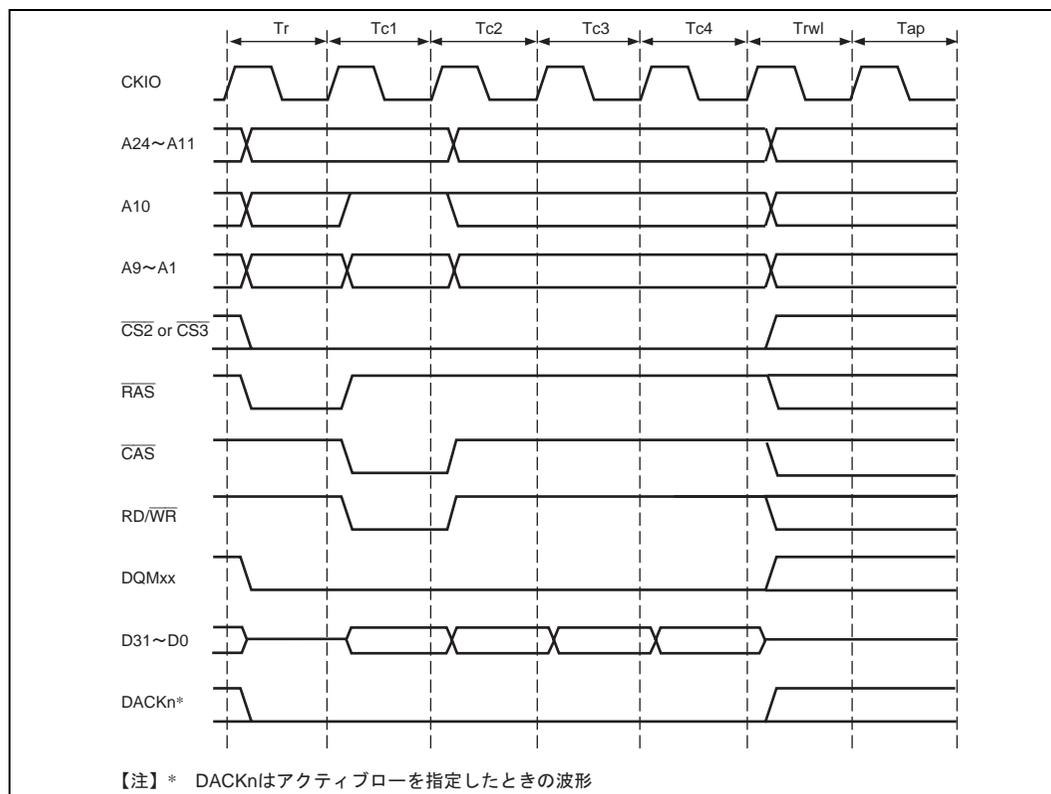


図 7.23 (a) バーストライトモード基本タイミング (オートプリチャージ) | : E = 1 : 1 以外

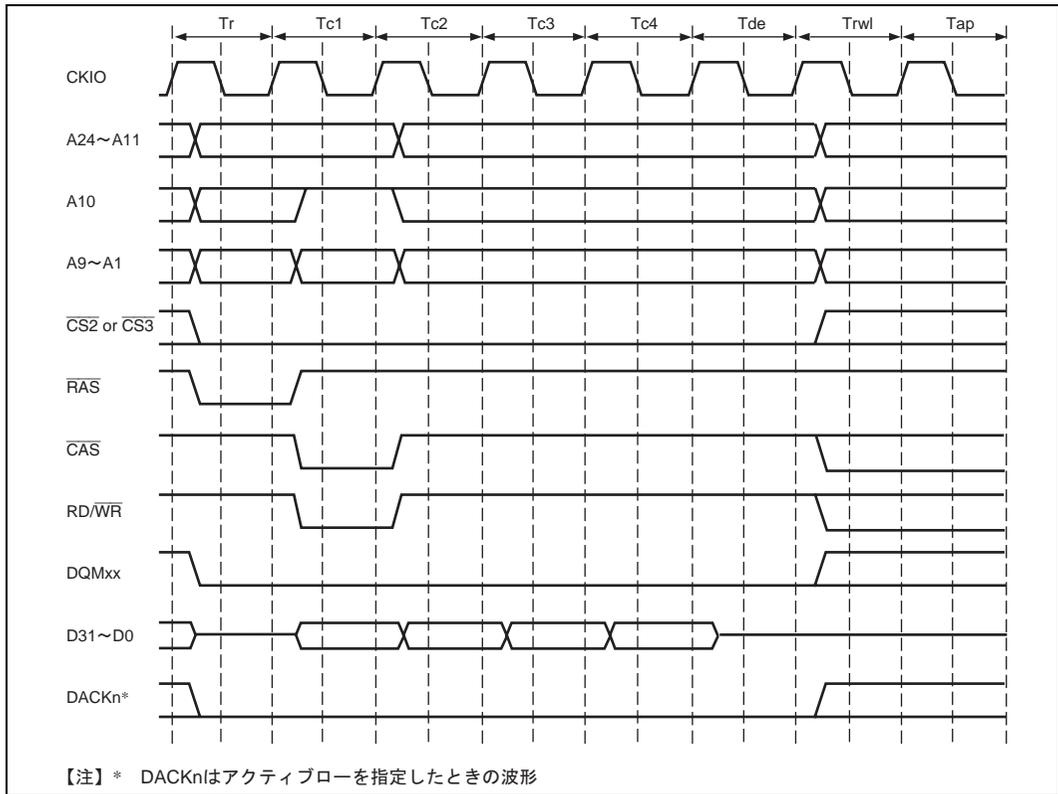


図 7.23 (b) バーストライトモード基本タイミング (オートプリチャージ) I : E = 1 : 1

### 7.5.7 バンクアクティブ

同一のロウアドレスに対するアクセスを高速にサポートするため、シンクロナス DRAM のバンク機能を用います。MCR の RASD ビットが 1 の場合、CS3 空間ではリード/ライトコマンドはオートプリチャージなしのコマンド (READ、WRIT) を使用してアクセスを行います。CS2 空間では本機能はサポートされていません。バンクアクティブ機能を用いた場合、アクセスが終了してもプリチャージは行われません。同じバンクの同じロウアドレスにアクセスする場合、DRAM の高速ページモードにおける  $\overline{\text{RAS}}$  ダウン状態と同様に、ACTV コマンドを発行せずに、直ちに READ または WRIT コマンドを発行することができます。シンクロナス DRAM の内部は 2 つのバンクに分かれているので、それぞれのバンクで 1 つずつロウアドレスをアクティブ状態としておくことができます。次のアクセスが異なるロウアドレスに対するものであった場合には、最初に PRE コマンドを発行して当該バンクのプリチャージを行い、プリチャージ完了後 ACTV コマンド、READ または WRIT コマンドの順にアクセスを行います。異なるロウアドレスに対するアクセスが続く場合には、アクセス要求があってからプリチャージを行うため、かえってアクセス時間が延びてしまいます。

書き込みの場合、オートプリチャージを行うと、WRITA コマンド発行後  $t_{\text{RWL}} + t_{\text{AP}}$  サイクルのあいだコマンド発行を行えません。バンクアクティブモードを用いると、同一ロウアドレスの場合には続けて READ または WRIT コマンドを発行することができます。したがって、1 つの書き込みごとに  $t_{\text{RWL}} + t_{\text{AP}}$  サイクルだけサイ

## 7. バスステートコントローラ (BSC)

---

クル数を短縮することができます。プリチャージコマンド発行からロウアドレスストローブコマンドまでのサイクル数は MCR の TRP1、TRP0 で決まります。

バンクアクティブモードを使用する場合と基本アクセスを用いる場合のどちらが実行速度が速いかは、同一のロウアドレスをアクセスする確率 (P1) と、アクセスが完了してから次にアクセスするまでの平均サイクル数 (tA) によって決まります。tA が tAP よりも大きい場合、リード時のプリチャージ待ちによる遅れが見えなくなります。tA が tRWL + tAP よりも大きければ、ライト時のプリチャージ待ちによる遅れも見えなくなります。この場合、バンクアクティブモードと基本アクセスのアクセス速度はアクセス開始からリード・ライトコマンド発行までのサイクル数となり、それぞれ (tRP + tRCD) × (1 - P1) と tRCD となります。

各バンクをアクティブ状態にしておける時間 tRAS には制限があります。プログラムの実行によってこの値を守る周期で、キャッシュにヒットせず別のロウアドレスにアクセスする保証がない場合、オートリフレッシュを行う設定にし、リフレッシュ周期を tRAS の最大値以下に設定する必要があります。これにより、各バンクの最大アクティブ状態時間の制約を守ることができます。オートリフレッシュを使用しない場合には、所定時間以上バンクがアクティブ状態にとどまらない工夫をプログラムする必要があります。

図 7.24 (a) (b) にオートプリチャージでないバーストリードサイクルを、図 7.25 (a) (b) には同一のロウアドレスに対するバーストリードサイクルを、図 7.26 (a) (b) には異なるロウアドレスに対するバーストリードサイクルを示します。同様に図 7.27 にオートプリチャージでないライトサイクルを、図 7.28 に同一のロウアドレスに対するライトサイクルを、図 7.29 には異なるロウアドレスに対するライトサイクルを示します。

図 7.25 において READ コマンドを発行する Tc サイクルに先立って、何も行わない Tnop サイクルが挿入されていますが、シンクロナス DRAM は読み出し時にバイト指定を行う DQMxx 信号について、2 サイクルのレイテンシがあります。Tnop を挿入しないで Tc サイクルを直ちに行うと、TdI サイクルのデータ出力に対する DQMxx 信号の指定が行えません。このため Tnop サイクルを挿入します。CAS レイテンシが 2 以上の場合には、Tc サイクル以降に DQMxx 信号を設定しても間に合うため、Tnop サイクルの挿入は行われません。

バンクアクティブモードに設定すると、CS3 空間のそれぞれのバンクに対するアクセスのみを見た場合、同一のロウアドレスに対するアクセスが続く限り図 7.24 または図 7.27 で始まり、図 7.25 または図 7.28 を繰り返します。間に別の CS 空間に対するアクセスがあっても影響はしません。バンクアクティブ中に別のロウアドレスに対するアクセスが発生した場合、これを検出した後図 7.25 または図 7.28 の代わりに図 7.26 または図 7.29 のバスサイクルを行います。バンクアクティブモードでも、リフレッシュサイクルの後またはバスアービトレーションによるバス解放の後には、双方のバンクが非アクティブな状態になります。

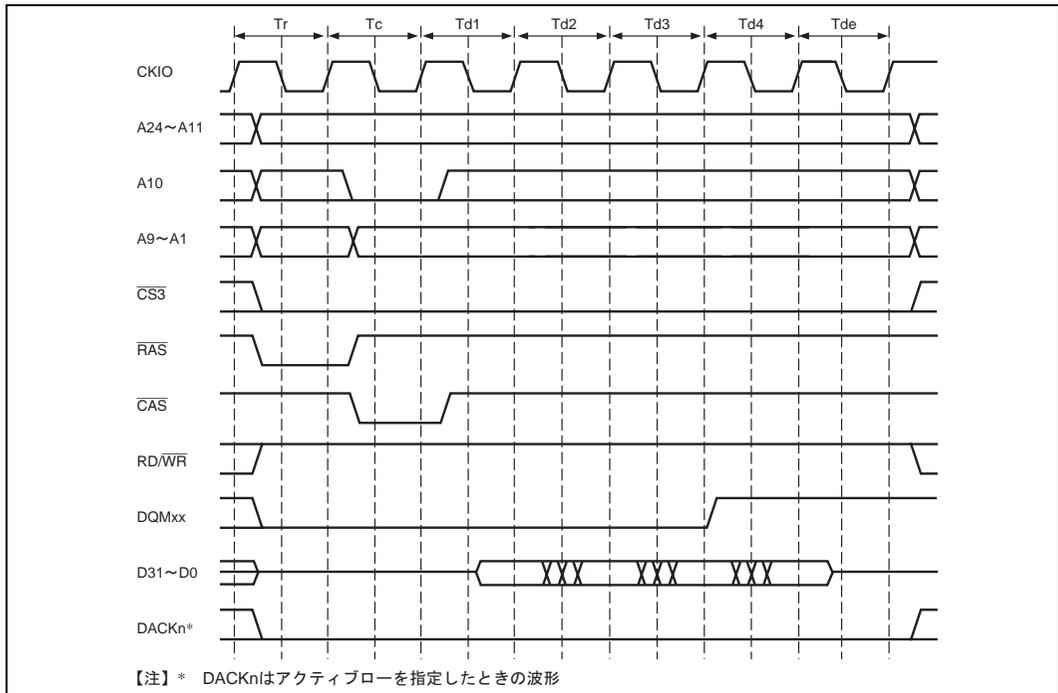


図 7.24 (a) バーストリードタイミング (プリチャージなし) | : E = 1 : 1 以外

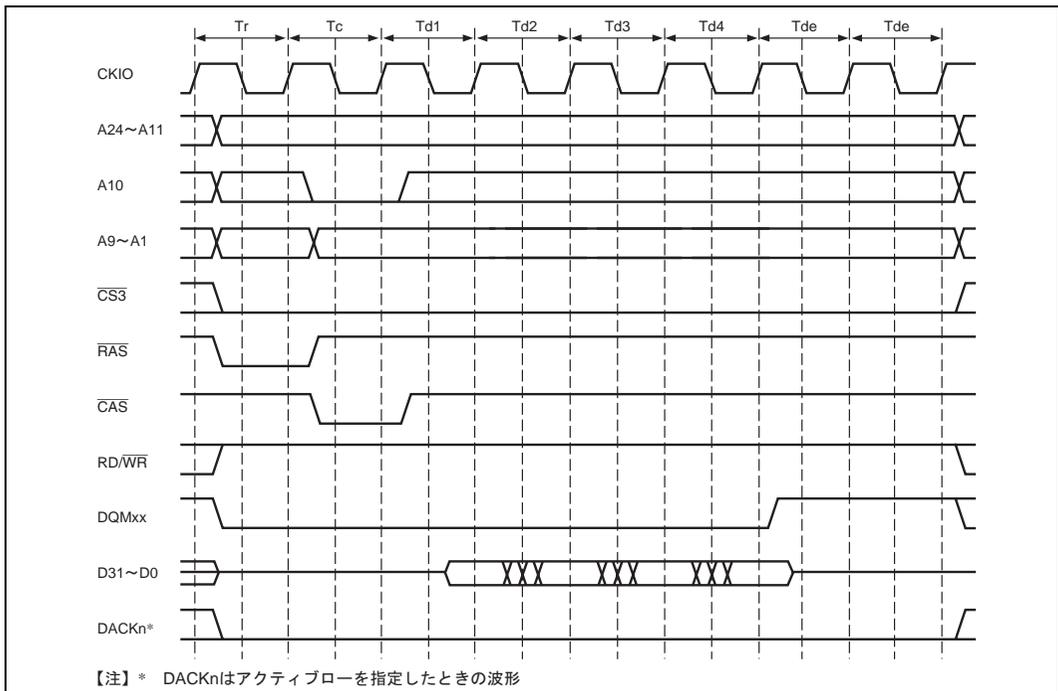


図 7.24 (b) バーストリードタイミング (プリチャージなし) | : E = 1 : 1

7. バスステートコントローラ (BSC)

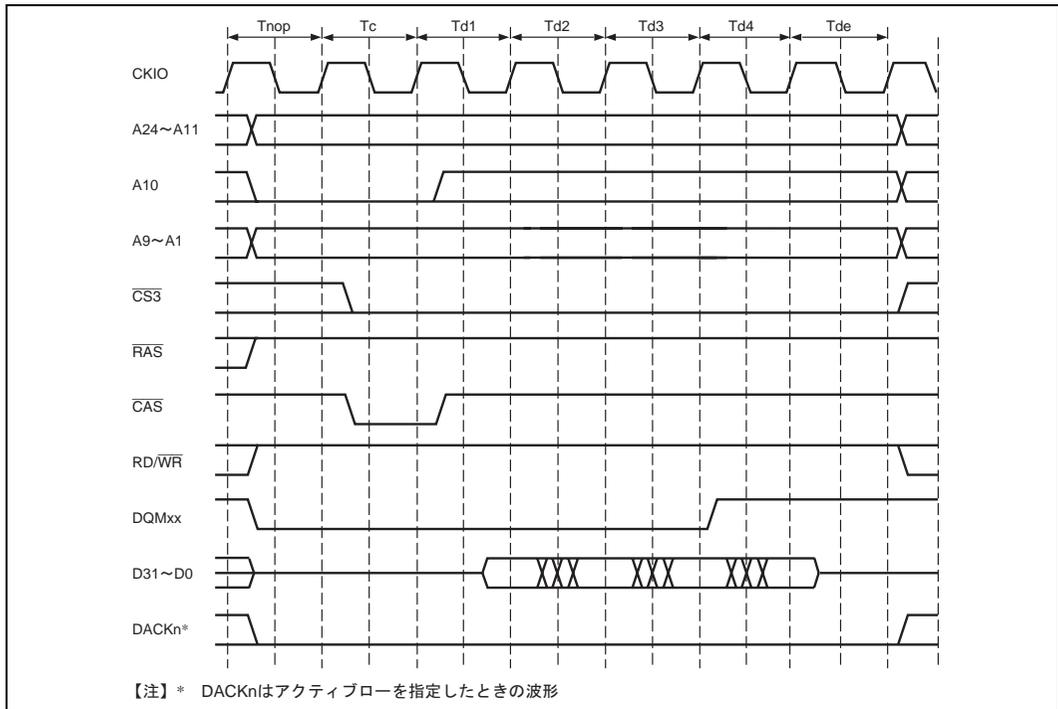


図 7.25 (a) バーストリードタイミング (バンクアクティブ、同一ロウアドレス) : E = 1 : 1 以外

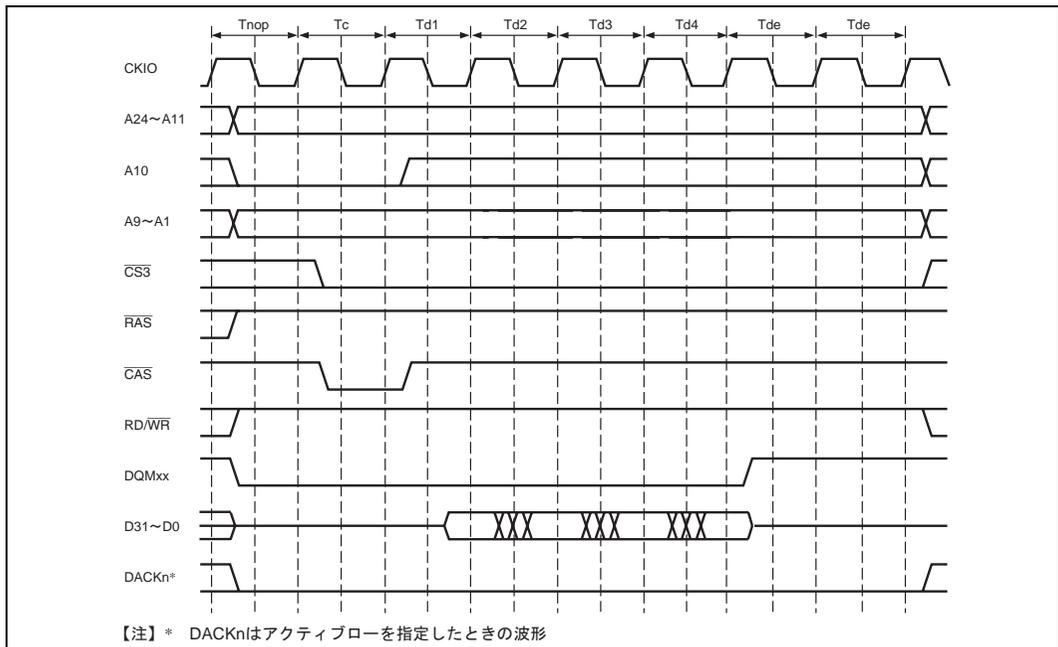


図 7.25 (b) バーストリードタイミング (バンクアクティブ、同一ロウアドレス) | : E = 1 : 1

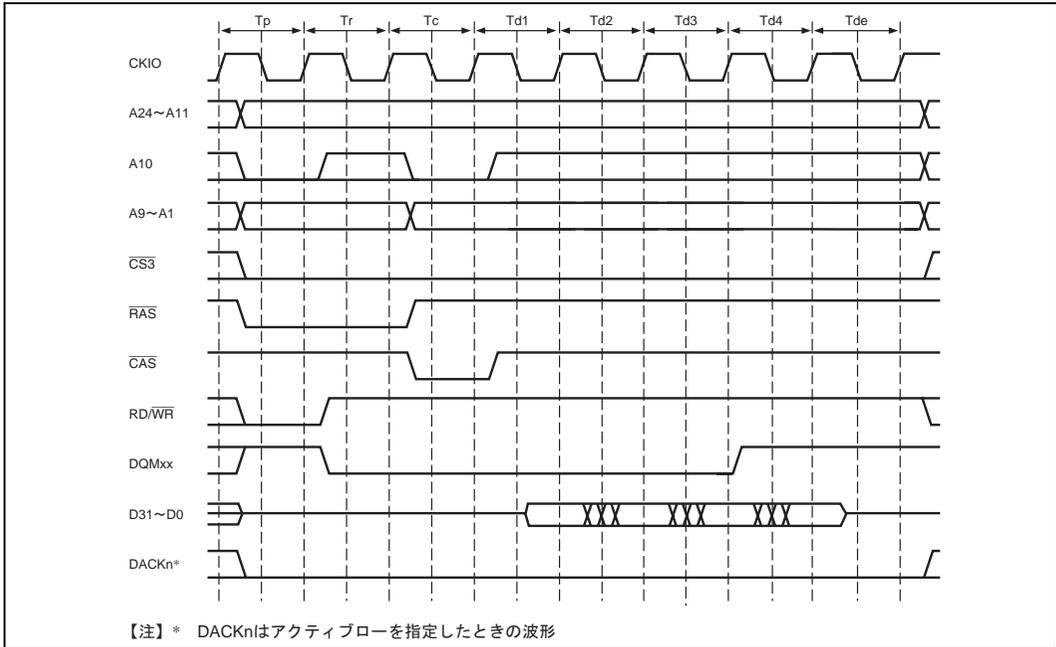


図 7.26 (a) バーストリードタイミング (バンクアクティブ、異なるロウアドレス) | : E = 1 : 1 以外

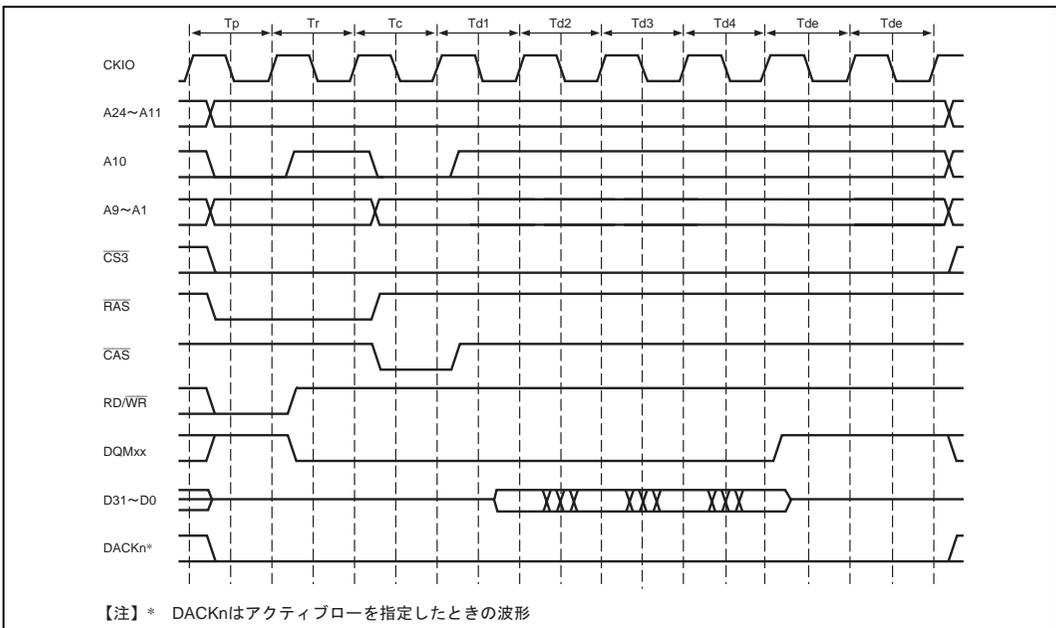


図 7.26 (b) バーストリードタイミング  
(バンクアクティブ、異なるロウアドレス) | : E = 1 : 1

## 7. バスステートコントローラ (BSC)

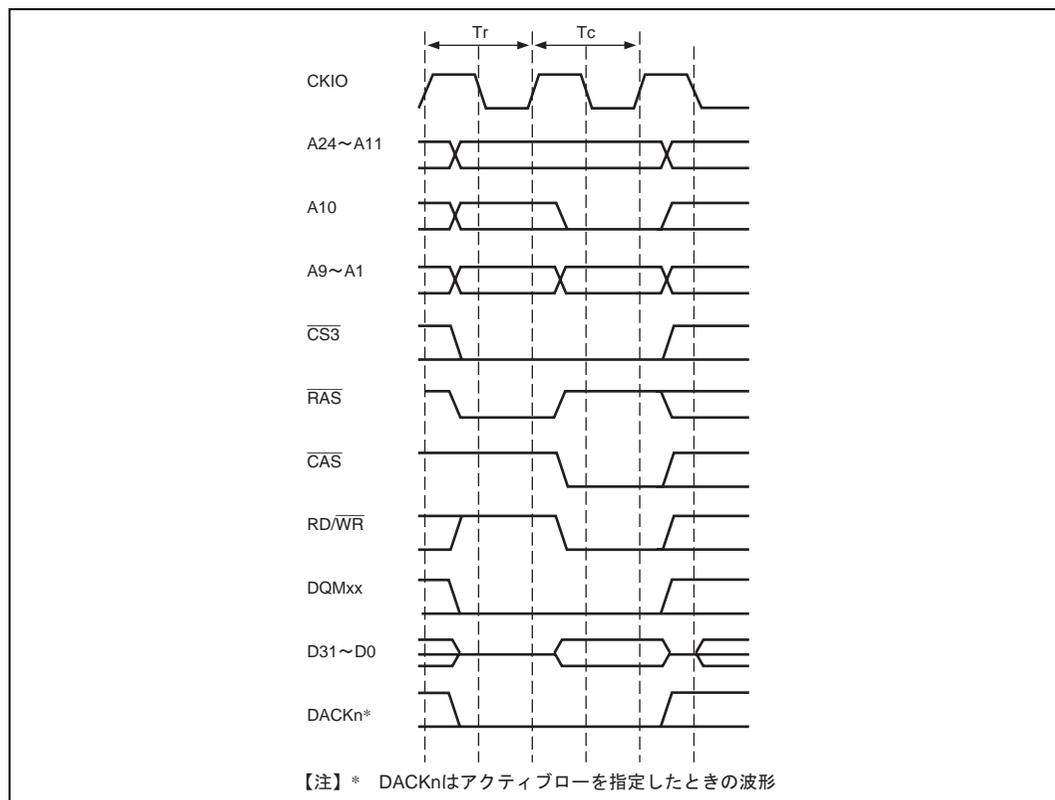


図 7.27 シングルライトモードタイミング (プリチャージなし)

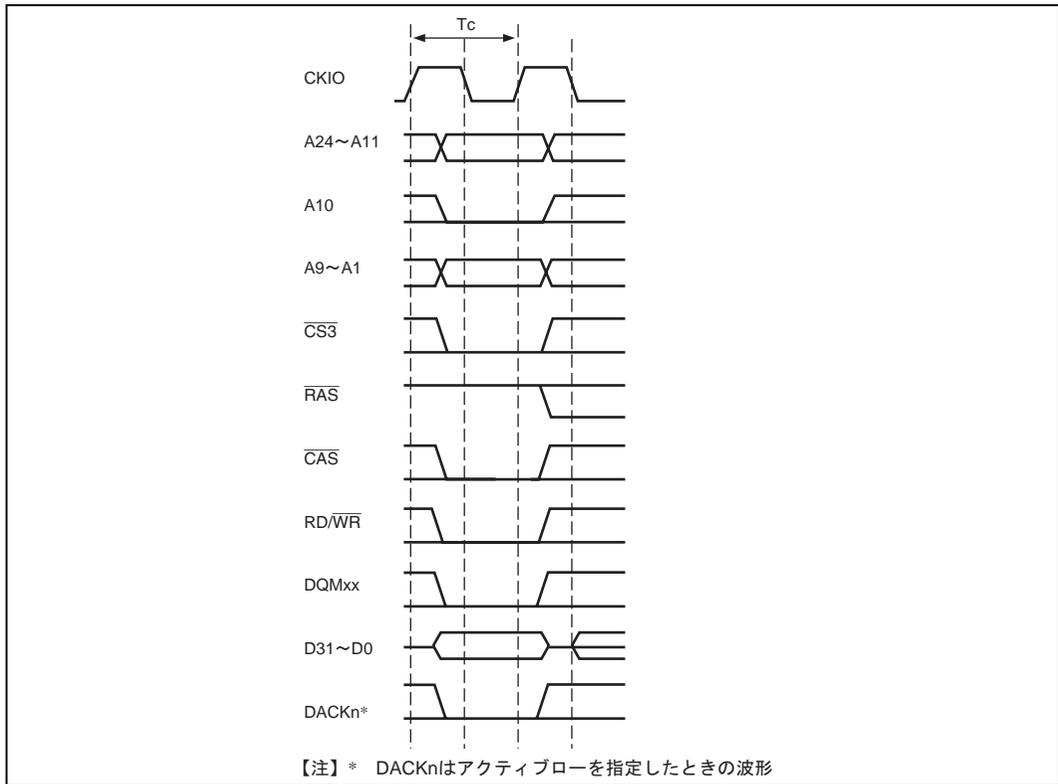


図 7.28 シングルライトモードタイミング (バンクアクティブ、同一ロウアドレス)

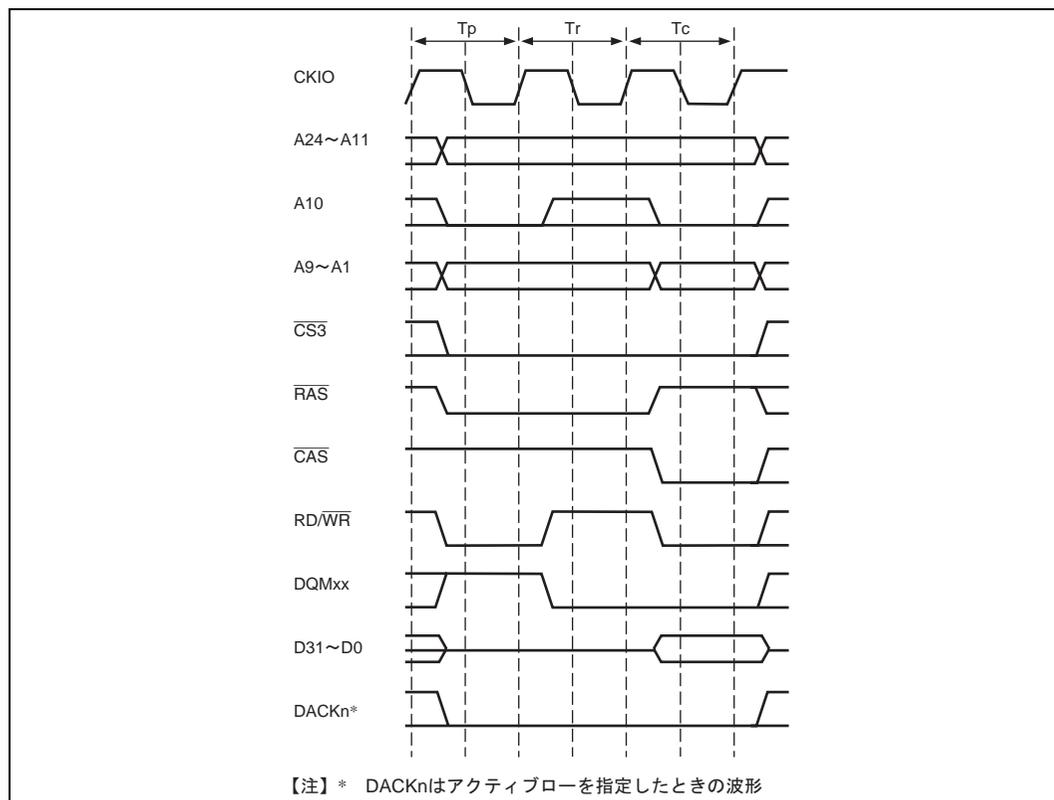


図 7.29 シングルライトモードタイミング (バンクアクティブ、異なるロウアドレス)

### 7.5.8 リフレッシュ

バスステートコントローラはシンクロナス DRAM のリフレッシュを制御する機能を備えています。MCR の RMODE ビットを 0 に、RFSH ビットを 1 にセットすることによって、オートリフレッシュを行わせることができます。また、RTCSR の RRC2 ~ 0 を設定することにより、連続してリフレッシュを発生させることができます。さらに、長時間シンクロナス DRAM にアクセスしないときには、RMODE ビットと RFSH ビットをともに 1 にすることによって、データ保持のための消費電力が少ないセルフリフレッシュモードを起動させることができます。

#### (1) オートリフレッシュ

RTCSR の CKS2 ~ 0 ビットで選択した入力クロックと、RTCOR に設定した値とで決まる間隔で RTCSR の RRC2 ~ 0 に設定した回数のリフレッシュが行われます。使用するシンクロナス DRAM のリフレッシュ間隔規定を満たすように、RTCOR と CKS2 ~ 0 ビットの値を設定してください。最初に RTCOR、RTCNT と MCR の RMODE ビットおよび RFSH ビットの設定を行い、最後に RTCSR の CKS2 ~ CKS0 および RRC2 ~ 0 の設定を行ってください。CKS2 ~ CKS0 によってクロックを選択すると、RTCNT はそのときの値からカウントアップを開始します。RTCNT の値は常に RTCOR の値と比較されており、両者の値が一致するとリフレッシュ要求が立

ち、RRC2 ~ 0 設定回数のオートリフレッシュが行われます。同時に RTCNT はゼロクリアされ、カウントアップが再開されます。図 7.30 にオートリフレッシュサイクルのタイミングを示します。

まず、すべてのバンクをアクティブ状態からプリチャージ状態にするため、 $T_p$  のサイクルで PALL コマンドを発行します。ついで、TRP1、TRP0 の設定値 -1 のアイドルサイクル挿入後、REF コマンドを  $T_{tr}$  サイクルに発行します。 $T_{tr}$  サイクル後 MCR の TRAS1、TRAS0 で指定されるサイクル数の間、新たなコマンドの出力は行いません。シンクロナス DRAM のリフレッシュサイクル時間の規定 (アクティブ・アクティブコマンド遅延時間) を満たすように TRAS を設定する必要があります。MCR の TRP1、TRP0 の設定値が 2 以上の場合、 $T_p$  サイクルと  $T_{tr}$  サイクルの間に NOP サイクルが挿入されます。

マニュアルリセット中は RTCNT のカウントアップが行われないため、リフレッシュ要求が発生しません。正しくリフレッシュを行うためには、マニュアルリセットの期間をリフレッシュサイクル間隔よりも短くし、マニュアルリセット解除後には直ちにリフレッシュが行われるように (RTCOR の値 - 1) を RTCNT に設定してください。

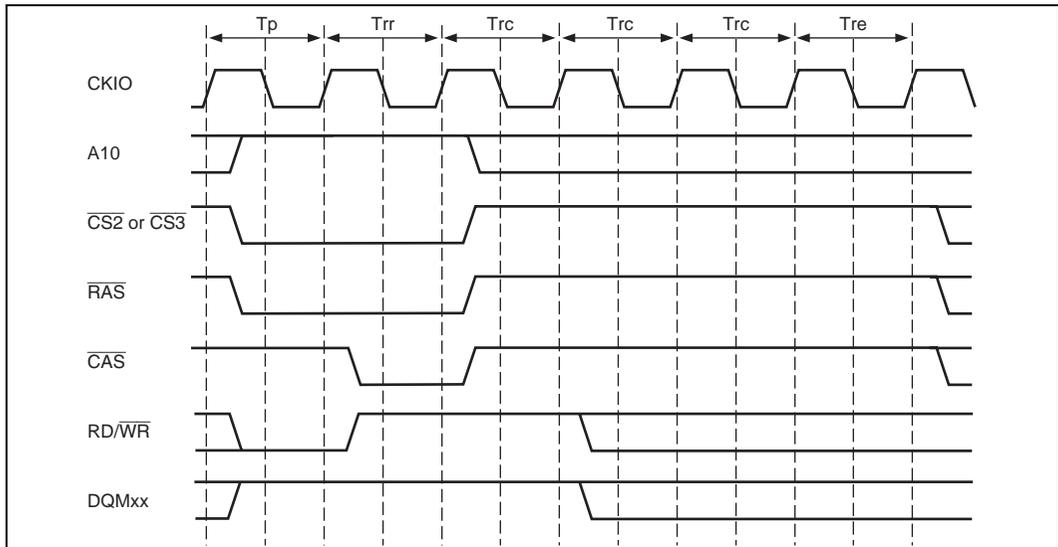


図 7.30 オートリフレッシュタイミング

## (2) セルリフレッシュ

セルリフレッシュのモードはシンクロナス DRAM の内部でリフレッシュタイミングとリフレッシュアドレスを生成する一種のスタンバイモードです。RMODE ビットと RFSH ビットをともに 1 にすることによって起動します。CKE 信号が L レベルの間セルリフレッシュ状態となっています。セルリフレッシュの状態の間は、シンクロナス DRAM にアクセスすることができません。セルリフレッシュの解除は RMODE ビットを 0 にすることによって行われます。セルリフレッシュ解除後、MCR の TRAS1、TRAS0 で指定されるサイクル数の間はコマンドの発行が禁止されています。セルリフレッシュのタイミングを図 7.31 に示します。セルリフレッシュ解除、データ保持が正しく行われるように、直ちにオートリフレッシュが正しい間隔で行われるように設定を行ってください。オートリフレッシュの設定をしている状態からセルリフレッシュにした場合、マニュアルリセットまたは NMI でスタンバイモードを脱出する場合、セルリフレッシュ解除時に RFSH = 1、RMODE = 0 の設定にすればオートリフレッシュが再開されます。セルリフレッシュ解除からオートリフレッ

## 7. バスステートコントローラ (BSC)

シユ開始までに時間がかかる場合には、この時間を考慮して RTCNT の初期値の設定を行ってください。RTCNT の値を RTCOR の値 - 1 に設定すると直ちにリフレッシュを開始することができます。

セルフリフレッシュに設定した後、本 LSI のスタンバイ機能を使ってチップをスタンバイ状態にした場合にもセルフリフレッシュ状態は継続され、スタンバイからの復帰が NMI による場合には復帰後もセルフリフレッシュ状態が保持されます。

マニュアルリセットによってもセルフリフレッシュ状態から抜け出すことはありません。

パワーオンリセットの場合にはバスステートコントローラのレジスタが初期化されるため、セルフリフレッシュ状態から抜け出します。

### (3) リフレッシュ要求とバスサイクル要求の関係

バスサイクル実行中にリフレッシュ要求が生じた場合、リフレッシュの実行はバスサイクルの完了まで待たされます。バスアービトレーション機能でバスを解放しているときにリフレッシュ要求が起きた場合、バスを獲得するまでリフレッシュの実行は待たされます。本 LSI は、リフレッシュの実行が待たされている間にバス権を要求する信号として REFOUT 端子を用意しています。そして、バス権を獲得するまで、REFOUT をアサートします。リフレッシュの実行を待たされている状態で RTCNT と RTCOR の一致が起こり新たなリフレッシュ要求が生じた場合には、前のリフレッシュ要求は消滅してしまいます。リフレッシュを正常に行うためには、リフレッシュ間隔よりも長いバスサイクルやバス権の占有が起こらないよう注意が必要です。

一方、セルフリフレッシュの最中にバスアービトレーション要求が起きても、バス権の解放はセルフリフレッシュが解除されるまで行われません。

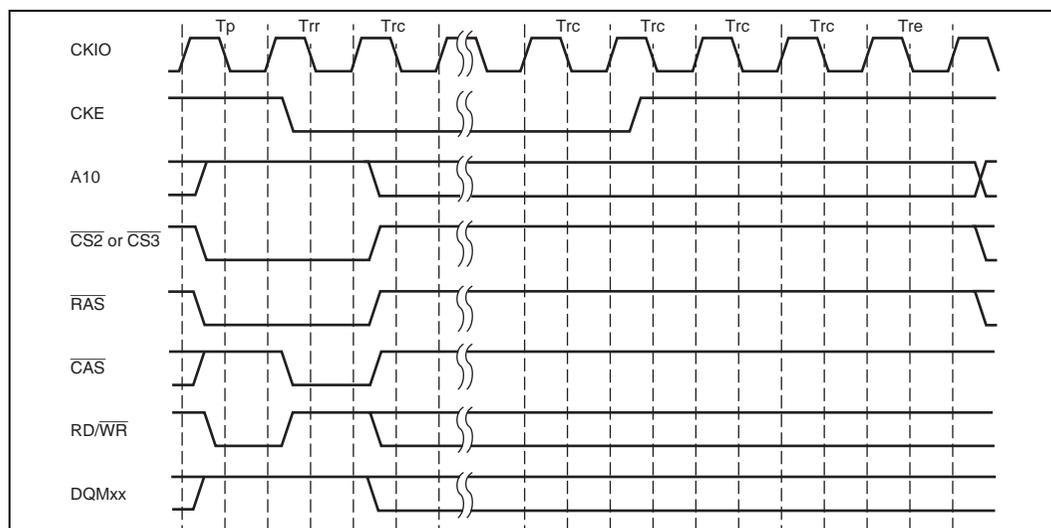


図 7.31 セルフリフレッシュタイミング

### 7.5.9 オートプリチャージサイクル (Tap) と次アクセスのオーバーラップについて

CPU、DMAC または E-DMAC が連続アクセスしている場合、先行するアクセスが SDRAM でかつオートプリチャージモードである場合、次アクセスにおいて異なる空間にアクセスするかまたは同じ SDRAM 内の異なるバ

バンクに対してアクセスを行うと、先行するアクセスのオートプリチャージサイクル (Tap) と次アクセスがオーバーラップします (次アクセスが通常空間の場合でも Tap サイクル分のオーバーラップが発生します)。そのため、次アクセスのサイクル数においては、見かけ上 Tap サイクル分速くアクセスすることになります。オーバーラップが発生する具体的なケースを表 7.7 に示します。また図 7.32 にオーバーラップイメージとして CS2 と CS3 空間に SDRAM が接続される場合 (表 7.7 の No.3) を示します。

表 7.7 Tap サイクルと次アクセスとのオーバーラップが発生する状態

No.	先行アクセス	次アクセス
1	CS3 空間 オートプリチャージモード	CS0、1、2、4 空間の異なる空間にアクセス
2		CS3 空間の異なるバンクにアクセス
3	CS2 空間 オートプリチャージモード	CS0、1、3、4 空間の異なる空間にアクセス
4		CS2 空間の異なるバンクにアクセス

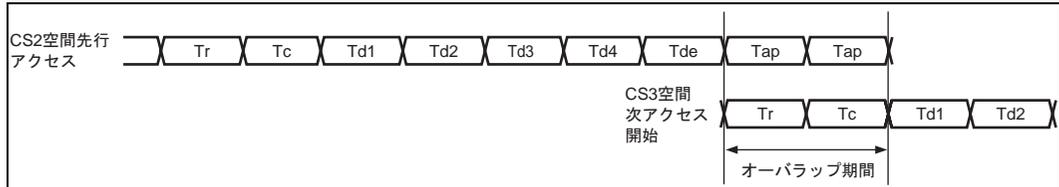


図 7.32 オーラップイメージ (条件: CS2 空間に SDRAM を接続 (RAS プリチャージ期間を 2 サイクルに設定) CS3 空間に SDRAM を接続)

### 7.5.10 パワーオンシーケンス

シンクロナス DRAM を使用するためには、パワーオン後、シンクロナス DRAM に対してモードの設定を行う必要があります。シンクロナス DRAM の初期化を正しく行うためには、まず最初にバスステートコントローラのレジスタを設定した後、シンクロナス DRAM のモードレジスタに対する書き込みを行わなければなりません。シンクロナス DRAM のモードレジスタの設定は  $\overline{CS2}$  または  $\overline{CS3}$  および  $\overline{RAS}$ 、 $\overline{CAS}$  /  $\overline{OE}$ 、 $\overline{RD}$  /  $\overline{WR}$  信号の組み合わせで、その時点のアドレス信号の値が取り込まれます。バスステートコントローラは、設定したい値を X とすると CPU から X + H'FFFF0000、または X + H'FFFF8000 番地に書き込みを行うことによって、値 X がシンクロナス DRAM のモードレジスタに書き込まれるように動作します。ここで X + H'FFFF0000 と X + H'FFFF8000 のどちらを使うかは、シンクロナス DRAM の仕様に依存します。また、X は H'000 ~ H'FFF としてください。この際データは無視されますが、モードの書き込みはワードサイズで行います。本 LSI でサポートしているバーストリードシングルライト、CAS レイテンシ 1 から 3、ラップタイプ=シーケンシャル、バースト長 8 または 4 (16 ビット幅か 32 ビット幅かに依存) を設定するには以下のアドレスにワードサイズで任意のデータを書き込みます。

16 ビット幅の時、	CAS レイテンシ 1	H'FFFF0426 (H'FFFF8426)
	CAS レイテンシ 2	H'FFFF0446 (H'FFFF8446)
	CAS レイテンシ 3	H'FFFF0466 (H'FFFF8466)

## 7. バスステートコントローラ (BSC)

---

32 ビット幅の時、	CAS レイテンシ 1	H'FFFF0848 (H'FFFF8848)
	CAS レイテンシ 2	H'FFFF0888 (H'FFFF8888)
	CAS レイテンシ 3	H'FFFF08C8 (H'FFFF88C8)

また、バーストリードバーストライト、CAS レイテンシ 1 から 3、ラップタイプ=シーケンシャル、バースト長 8 または 4 (16 ビット幅か 32 ビット幅かに依存) を設定するには以下のアドレスにワードサイズで任意のデータを書き込みます。

16 ビット幅の時、	CAS レイテンシ 1	H'FFFF0026 (H'FFFF8026)
	CAS レイテンシ 2	H'FFFF0046 (H'FFFF8046)
	CAS レイテンシ 3	H'FFFF0066 (H'FFFF8066)
32 ビット幅の時、	CAS レイテンシ 1	H'FFFF0048 (H'FFFF8048)
	CAS レイテンシ 2	H'FFFF0088 (H'FFFF8088)
	CAS レイテンシ 3	H'FFFF00C8 (H'FFFF80C8)

モードレジスタ設定タイミングを図 7.33 に示します。

X + H'FFFF0000、または X + H'FFFF8000 番地への書き込みによって、まず全バンクプリチャージコマンド (PALL) を発行し、さらにシンクロナス DRAM のパワーオンシーケンスに必要なダミーのオートリフレッシュコマンド (REF) を 8 回発行します。そして最後に、モードレジスタ書き込みコマンド (MRS) が発行されます。全バンクプリチャージコマンドと 1 回目のオートリフレッシュコマンドの間には 3 アイドルサイクル、オートリフレッシュコマンドとオートリフレッシュコマンドの間、および 8 回目のオートリフレッシュコマンドとモードレジスタ書き込みコマンドの間には、8 アイドルサイクルが挿入されます。

シンクロナス DRAM のモードレジスタに対する書き込み終了後、シンクロナス DRAM の各バンクに対して、ダミーの読み出しを行ってから、通常のアクセスを開始してください。

シンクロナス DRAM は全バンクプリチャージコマンドに先立って、電源投入後に一定のアイドル時間を保証しなければなりません。必要なアイドル時間は、シンクロナス DRAM のマニュアルをご覧ください。リセット信号のパルス幅がこのアイドル時間より長い場合には、直ちにモードレジスタの設定を行っても問題はありませんが、短い場合は注意する必要があります。

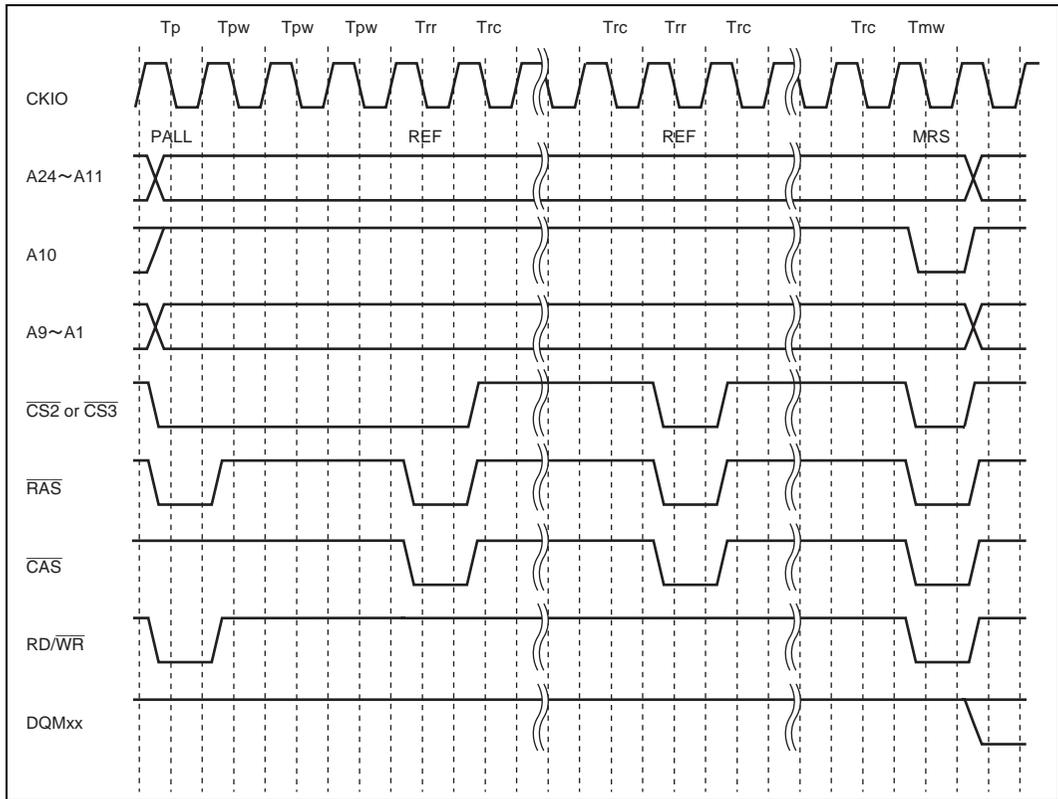


図 7.33 SDRAM モード書き込みタイミング

## 7. バスステートコントローラ (BSC)

### 7.5.11 64M ビットシンクロナス DRAM (2M ワード × 32 ビット品) との接続方法

#### (1) 64M ビットシンクロナス DRAM ( × 32 ビット品) との接続例

本 LSI と 64M ビットシンクロナス DRAM ( × 32 ビット品) との接続例を図 7.34 に示します。

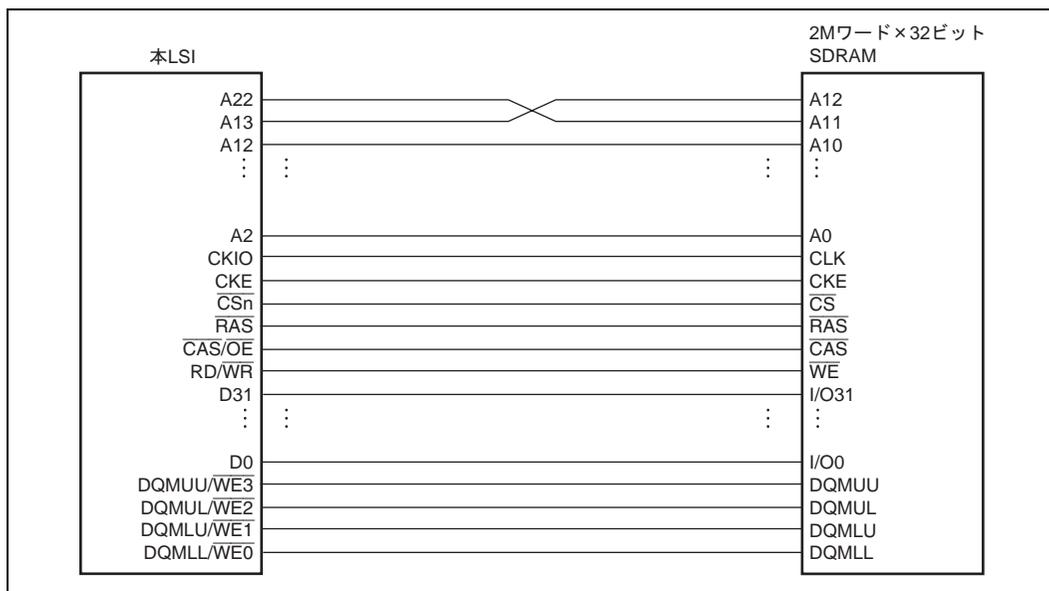


図 7.34 64M ビットシンクロナス DRAM (2M ワード × 32 ビット品) との接続例

#### (2) バスステートコントローラ (BSC) のレジスタ設定

個別メモリコントロールレジスタ (MCR) の内容は、次のように設定してください。

MCR (ビット 6) SZ = 1

MCR (ビット 7) AMX2 = 0

MCR (ビット 5) AMX1 = 0

MCR (ビット 4) AMX0 = 0

#### (3) シンクロナス DRAM に対するモード設定

シンクロナス DRAM に対してモードを設定する場合には、CPU から X+H'FFFF0000、または X+H'FFFF8000 番地へ書き込みを行ってください (X は、設定した値)。X+H'FFFF0000 と X+H'FFFF8000 のどちらを使うかは、シンクロナス DRAM の仕様によって異なります。

## (4) 128M ビットシンクロナス DRAM (4M ワード × 32 ビット品) との接続例

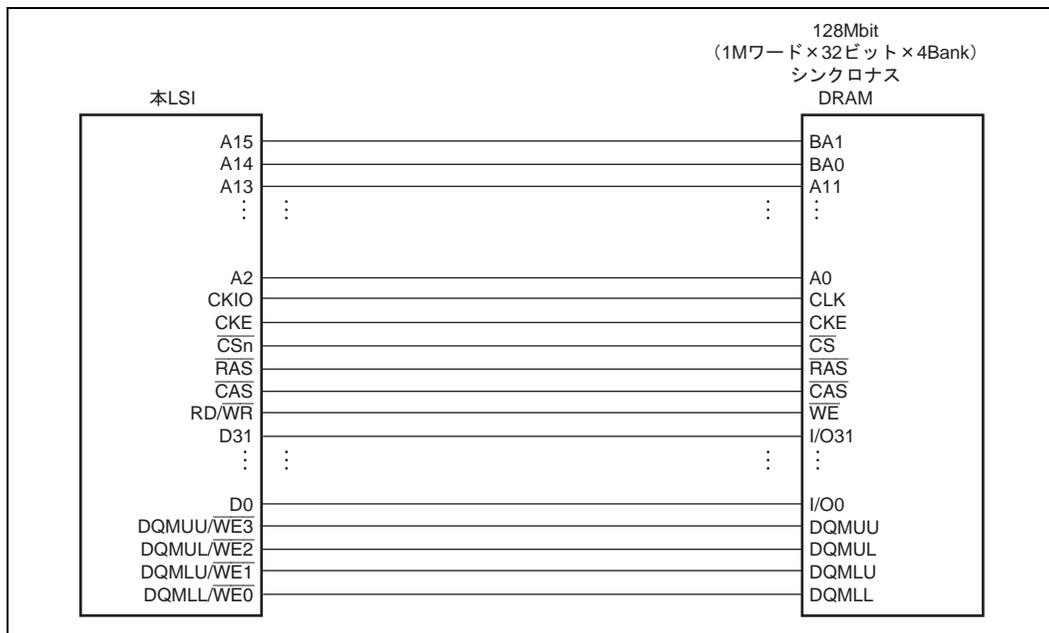


図 7.35 128M ビットシンクロナス DRAM (4M ワード × 32 ビット品) との接続例

7. バスステートコントローラ (BSC)

(5) 128M ビットシンクロナス DRAM (8M ワード × 16 ビット品) との接続例

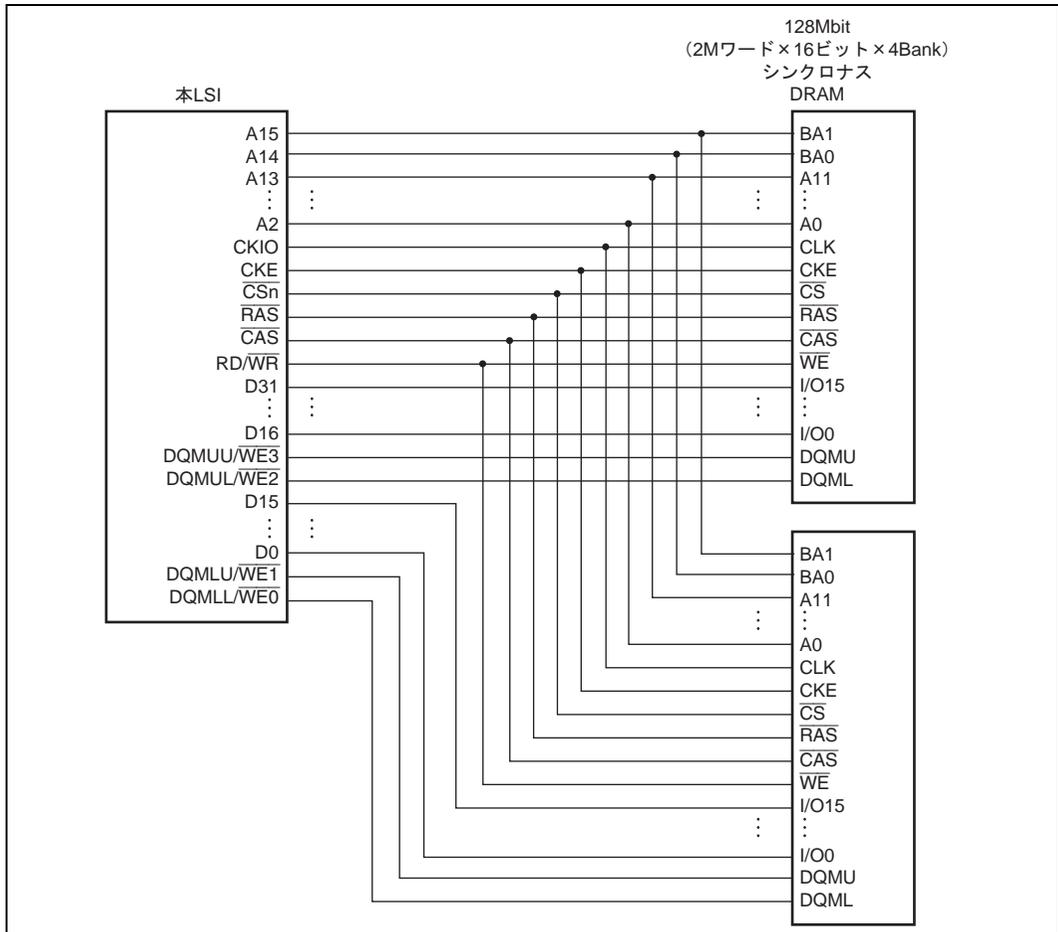


図 7.36 128M ビットシンクロナス DRAM (8M ワード × 16 ビット品) との接続例

(6) 256M ビットシンクロナス DRAM (8M ワード × 32 ビット品) との接続例

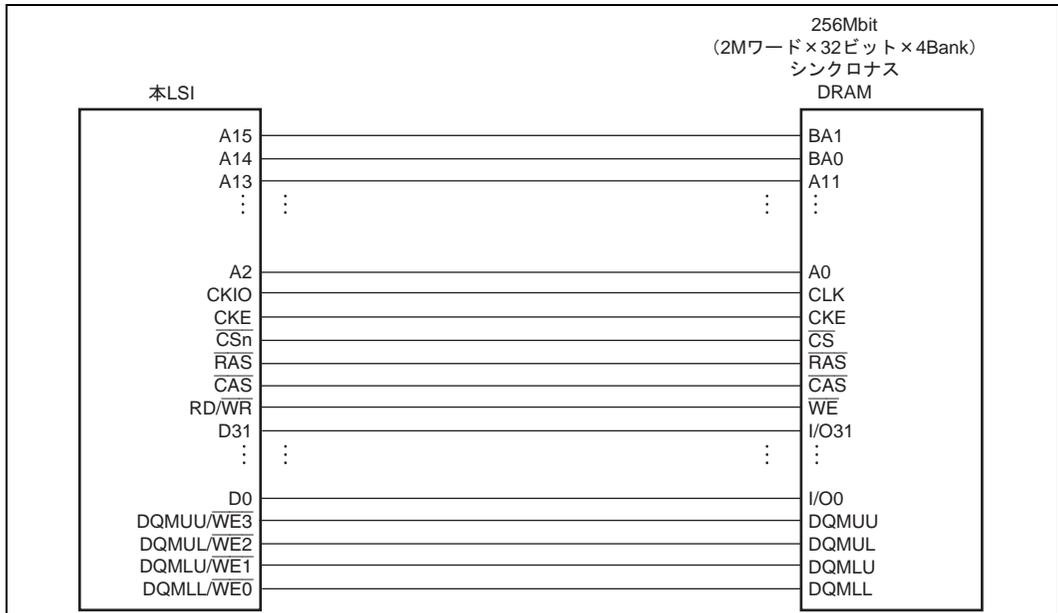


図 7.37 256M ビットシンクロナス DRAM (8M ワード × 32 ビット品) との接続例

## 7.6 DRAM インタフェース

### 7.6.1 DRAM 直結方式

BCR1 の DRAM 他イネーブルビット (DRAM2 ~ 0) を 010 に設定すると、CS3 空間が DRAM 空間となり、本 LSI と DRAM を直結させるための DRAM インタフェース機能が使用できるようになります。

インタフェースのデータ幅は 16 ビットと 32 ビットから選択できます。

バイトアクセスのコントロールは  $\overline{\text{CAS}}$  を用いるので、16 ビット幅 DRAM で接続可能なものは CAS2 本方式のものです。

接続に使用する信号は  $\overline{\text{RAS}}$ 、 $\overline{\text{CAS3}} \sim \overline{\text{CAS0}}$ 、 $\text{RD}/\overline{\text{WR}}$  です。データ幅が 16 ビットのときには  $\overline{\text{CAS3}}$ 、 $\overline{\text{CAS2}}$  は使用しません。

アクセスモードとしては通常のリード、ライトアクセスに加えて、高速ページモードを利用したバーストアクセスをサポートします。

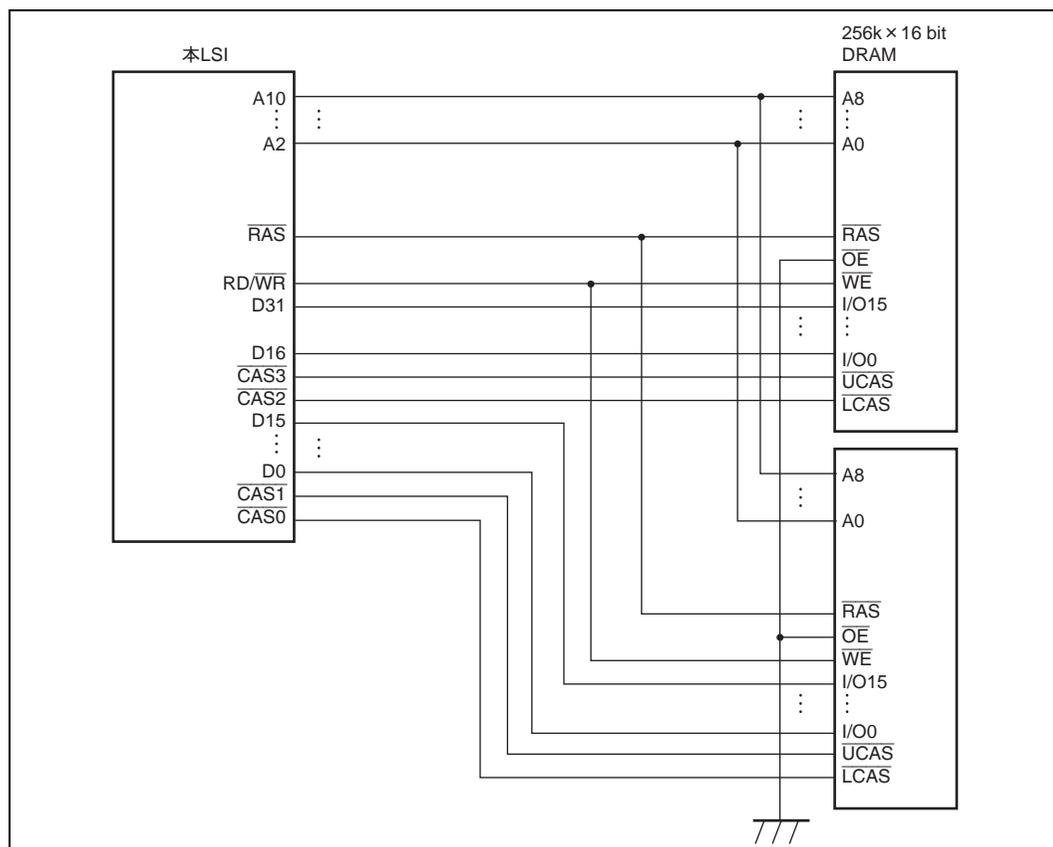


図 7.38 DRAM 接続例 (32 ビットデータ幅)

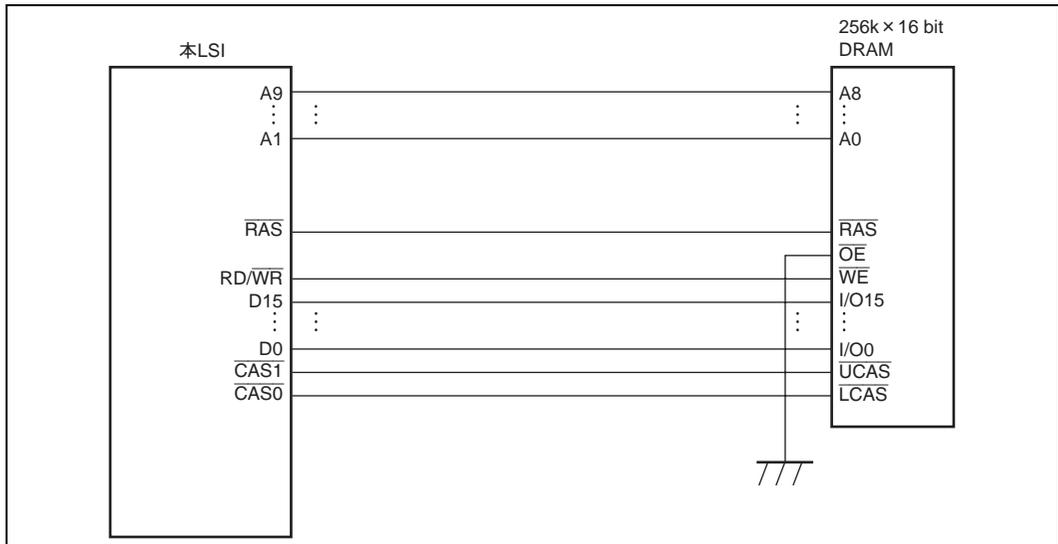


図 7.39 DRAM 接続例 (16 ビットデータ幅)

### 7.6.2 アドレスマルチプレクス

CS3 空間を DRAM 空間に設定すると、常にアドレスのマルチプレクスが行われます。これによって外付けのアドレスマルチプレクス回路なしに、ロウアドレスとカラムアドレスのマルチプレクスが必要な DRAM を本 LSI に直結することができます。マルチプレクスの方法は MCR の AMX1 ~ 0 ビットの設定によって以下の 4 通りの中から選ぶことができます。AMX1 ~ 0 とアドレスマルチプレクスの関係を表 7.8 に示します。アドレスマルチプレクスの対象となるアドレス出力端子は A15 から A1 です。A24 から A16 には元々のアドレスが出力されます。なお、AMX2 は DRAM アクセス時には予約となっていますので、0 を設定するようにしてください。

表 7.8 AMX 1 ~ 0 とアドレスマルチプレクスの関係

AMX1	AMX0	カラムアドレスビット数	出力されるロウアドレス	出力されるカラムアドレス
0	0	8 ビット	A23 ~ A9	A15 ~ A1
0	1	9 ビット	A24 ~ A10	A15 ~ A1
1	0	10 ビット	A24 ~ A11*1	A15 ~ A1
1	1	11 ビット	A24 ~ A12*2	A15 ~ A1

【注】 \*1 アドレス出力端子 A15 は High レベルです。

\*2 アドレス出力端子 A15、A14 は High レベルです。

### 7.6.3 基本タイミング

DRAM アクセスの基本タイミングは 3 サイクルです。DRAM アクセスの基本タイミングを図 7.40 示します。Tp はプリチャージサイクル、Tr は RAS アサートサイクル、Tc1 は CAS アサートサイクル、Tc2 はリードデータ

## 7. バスステートコントローラ (BSC)

取り込みサイクルです。連続してアクセスを行う場合、次のアクセスの  $T_p$  サイクルと前のアクセスの  $T_{c2}$  サイクルは重なり、最小3サイクルごとにアクセスを行うことができます。

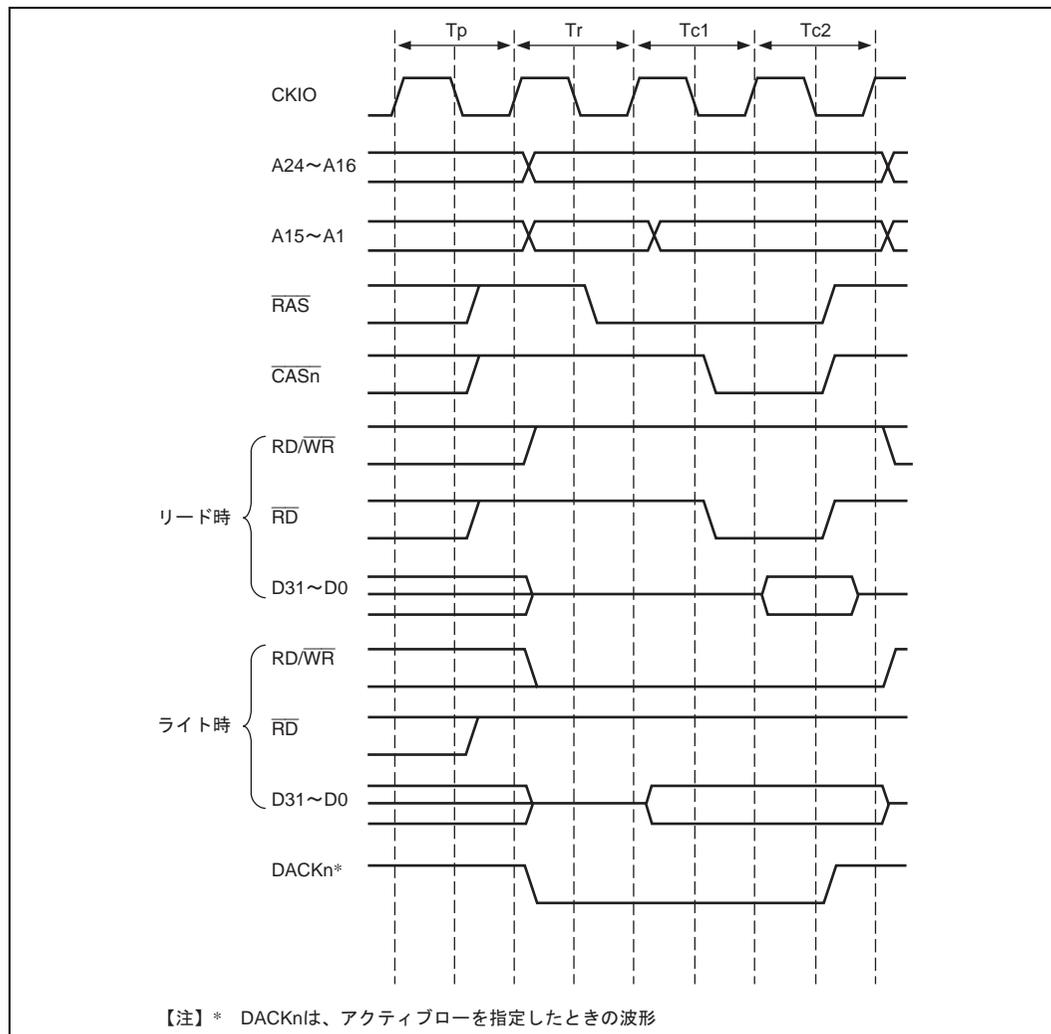


図 7.40 基本アクセスタイミング

### 7.6.4 ウェイトステート制御

クロック周波数を上げていくと、基本アクセスのようにすべてのステートを1サイクルで終わらせることができなくなってきます。そこで、WCR1、WCR2およびMCRにある設定ビットを用いてステートの延長ができるようになっています。設定を使ってステートを延長したタイミングを図7.41に示します。 $\overline{RAS}$ のプリチャージ時間を確保するための  $T_p$  サイクルは、MCRのTRP1、TRP0ビットによって  $T_{pw}$  を挿入し1~2サイクルにすることができます。 $\overline{RAS}$  アサートから  $\overline{CASn}$  アサートまでのサイクル数は、MCRのRCD1、RCD0ビットによって  $T_{rw}$  を挿入し、1~3サイクルに延ばすことができます。 $\overline{CASn}$  アサートからアクセス終了までのサイクル数

は WCR1 の W31、W30 の設定によって 1 サイクルから 3 サイクルまで変えることができます。WCR2 の外部ウェイトマスクビット A3WM を 0、WCR1 の W31、W30 を 00 以外に設定すると、外部ウェイト端子  $\overline{\text{WAIT}}$  のサンプリングも行われ、さらにサイクル数を延ばすことができます。WCR2 の A3WM を 1 に設定すると、WCR1 の W31、W30 の設定にかかわらず、外部ウェイト入力は無視されます。 $\overline{\text{WAIT}}$  端子によるウェイトステート制御のタイミングを図 7.42 に示します。いずれの場合も連続してアクセスがある場合には、次のアクセスの  $T_p$  と前のアクセスの  $T_{c2}$  が重なります。DRAM アクセス時は、 $\overline{\text{BS}}$  がアサートしません。したがって、 $\overline{\text{WAIT}}$  端子の制御には、 $\overline{\text{RAS}}$ 、 $\overline{\text{CASn}}$ 、 $\overline{\text{RD}}$  等を用いて行ってください。

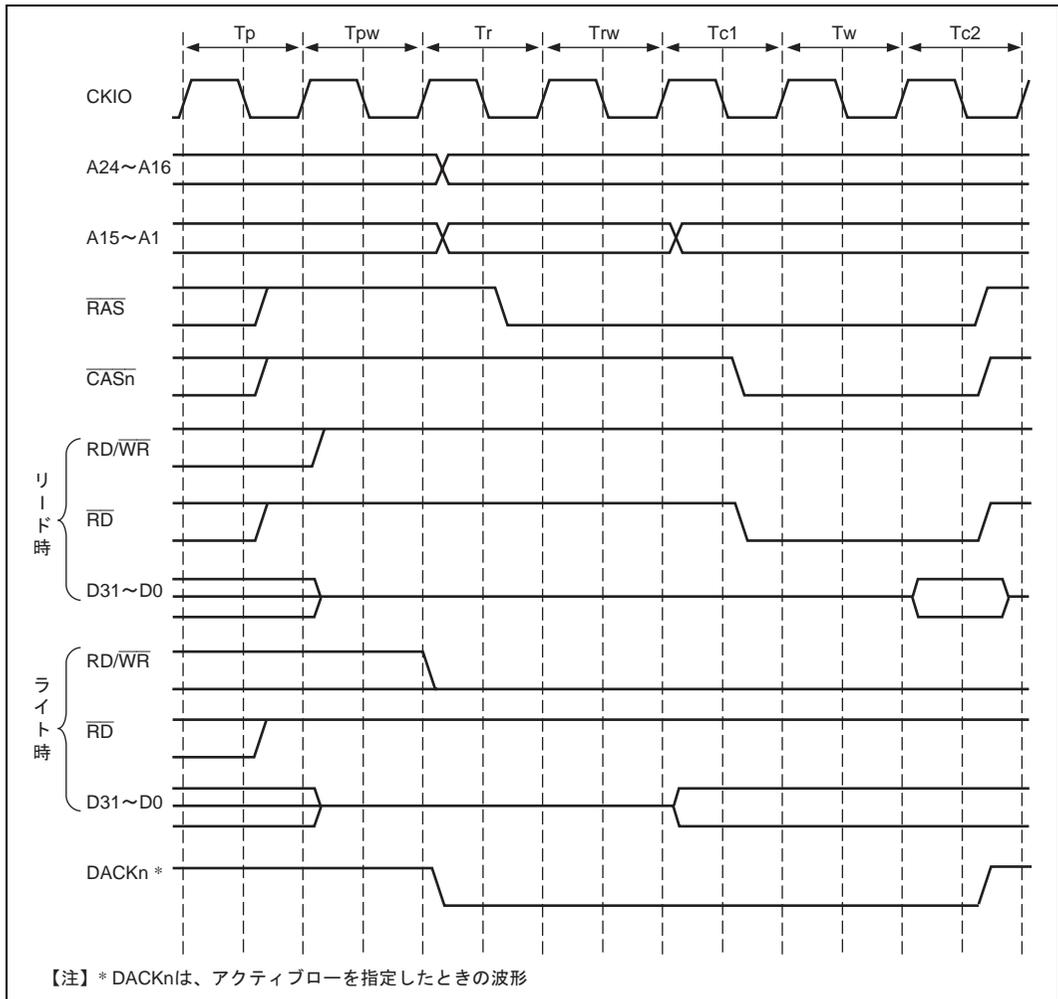


図 7.41 ウェイトステートタイミング

## 7. バスステートコントローラ (BSC)

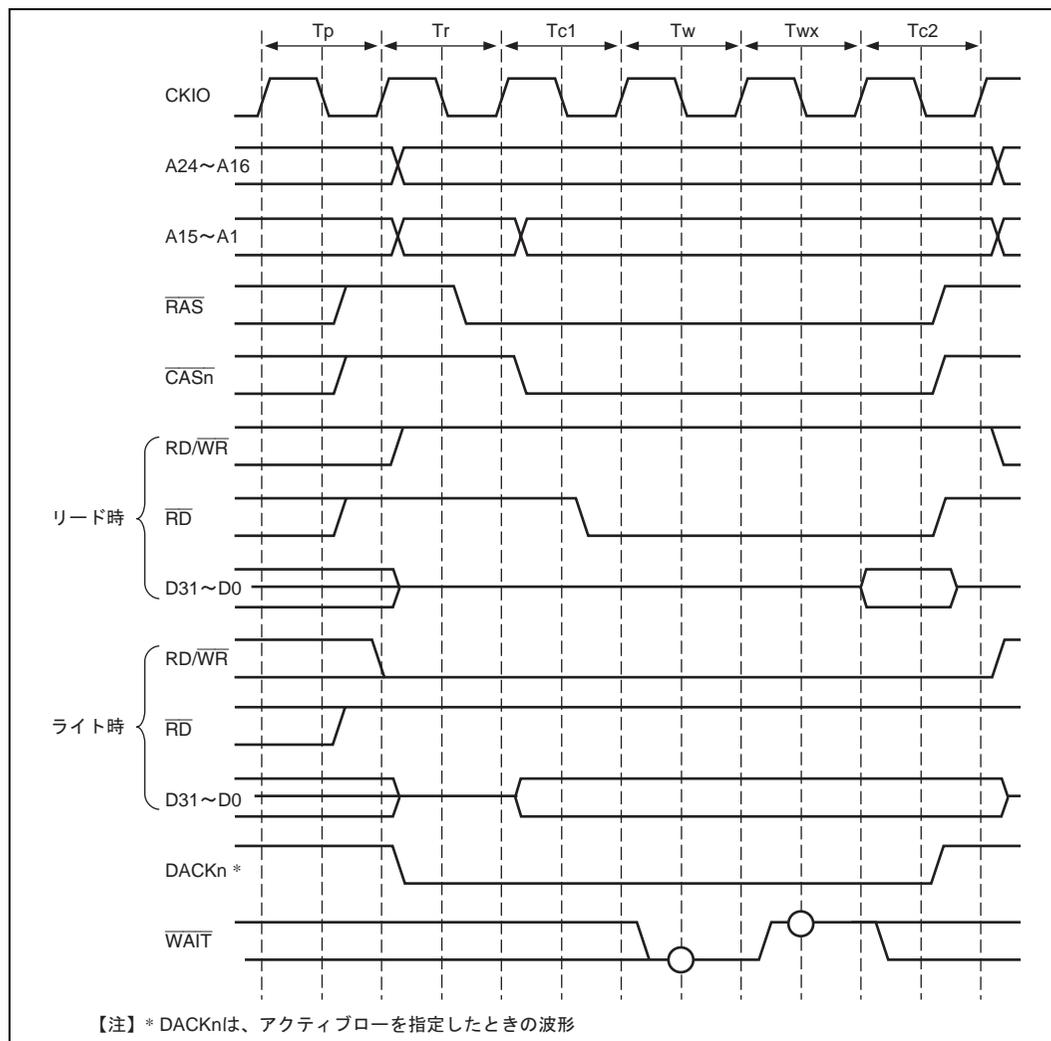


図 7.42 外部ウェイトステートタイミング

### 7.6.5 バーストアクセス

DRAM には、アクセスのたびにロウアドレスを出力してデータをアクセスするノーマルアクセスの他に、同一のロウに対するアクセスが連続する場合、ロウアドレスを 1 度出力した後はカラムアドレスを変更するだけでデータに高速にアクセスできる、高速ページモードを備えているものがあります。MCR のバーストイネーブルビット (BE) の設定によって、ノーマルアクセスと高速ページモードを利用したバーストアクセスを選択することができます。高速ページモードによるバーストアクセスのタイミングを図 7.43 に示します。バーストアクセスを行う場合にも、ウェイトステート制御機能によるサイクルの挿入を行うことができます。

本 LSI はバーストモードにおけるロウアドレスの一致を検出するためのアドレス比較器を持っています。これを利用し MCR の BE を 1 にするとともに、 $\overline{\text{RAS}}$  ダウンモード指定ビット RASD を 1 にすることによって、アク

セス終了後も  $\overline{\text{RAS}}$  をアサートしたまま放置する  $\overline{\text{RAS}}$  ダウンモードにすることができます。 $\overline{\text{RAS}}$  ダウンモードにおけるアクセスタイミングを図 7.44、図 7.45 に示します。 $\overline{\text{RAS}}$  ダウンモードを用いる場合、リフレッシュ周期が DRAM の  $\overline{\text{RAS}}$  アサート時間  $t_{\text{RAS}}$  の最大値よりも長い場合には、リフレッシュ周期を  $t_{\text{RAS}}$  の最大値以下にする必要があります。

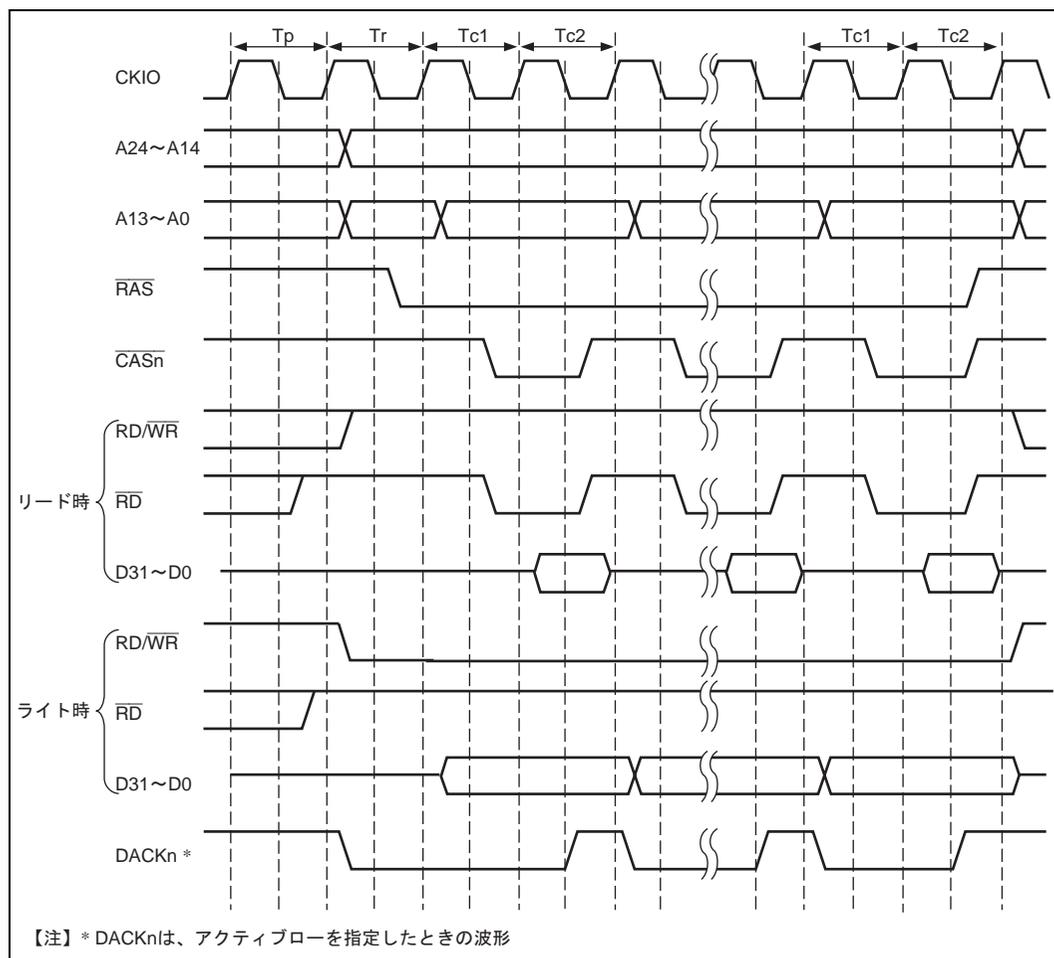


図 7.43 バーストアクセスタイミング

7. バスステートコントローラ (BSC)

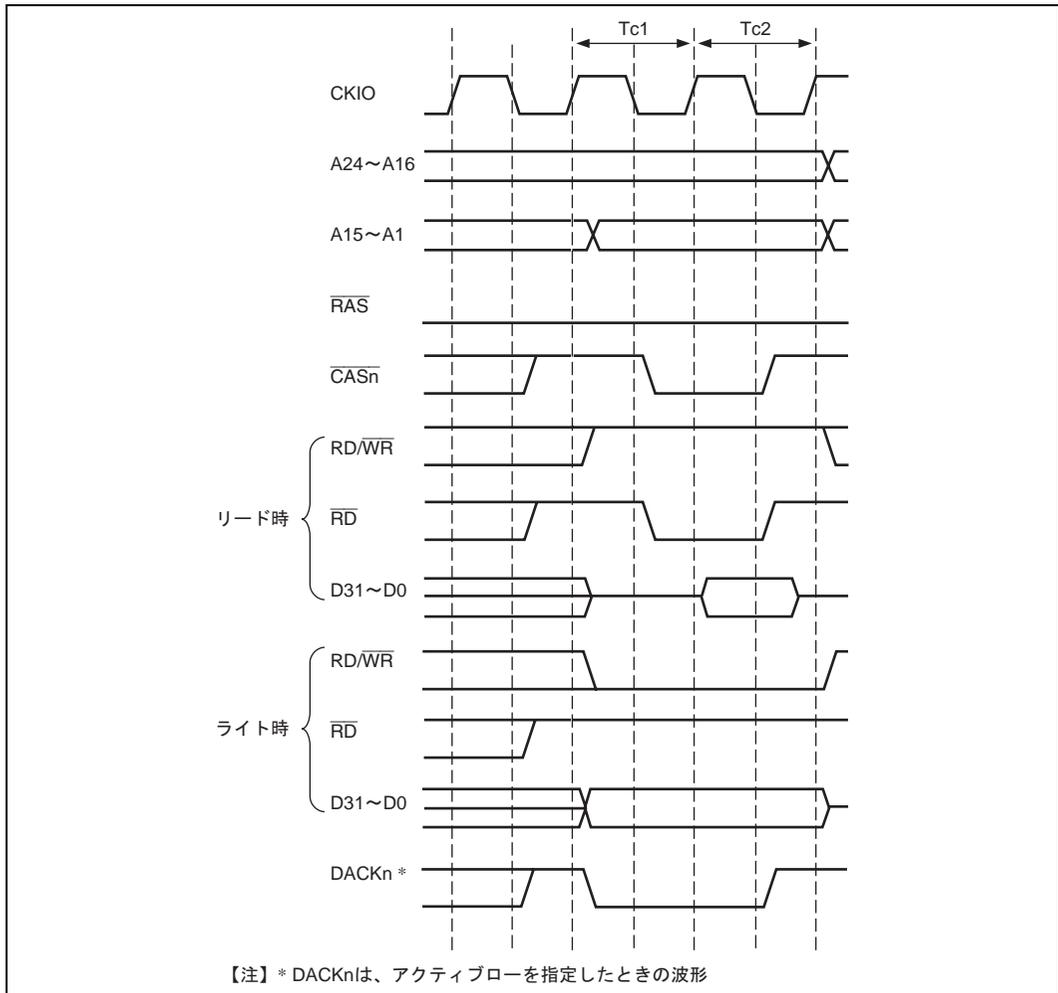


図 7.44 RAS ダウンモード、同一rowアクセスタイミング

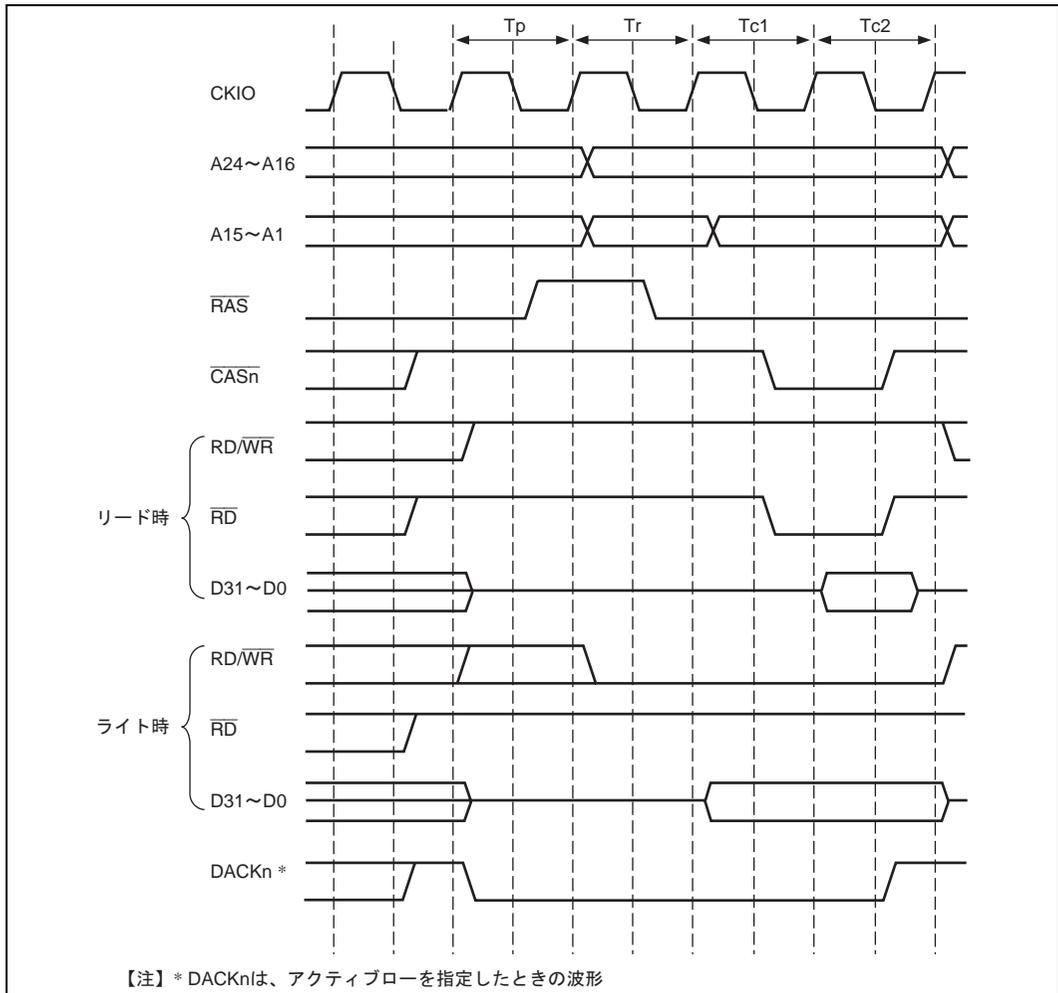


図 7.45 RAS ダウンモード、異なるロウアクセスタイミング

### 7.6.6 EDO モード

DRAM には、データリードサイクル時に  $\overline{\text{CASn}}$  信号のアサート中だけデータバスにデータを出力するもの他に、 $\overline{\text{RAS}}$ 、 $\overline{\text{OE}}$  とともにアサート中は、いったん  $\overline{\text{CASn}}$  信号をアサートすると  $\overline{\text{CASn}}$  信号をネゲートしても次に  $\overline{\text{CASn}}$  をアサートするまで、データバスにデータを出力する EDO モードを備えたものがあります。

本 LSI では、MCR の EDO モードビット (EDO) の設定によって、ノーマルアクセス / 高速ページモードによるバーストアクセスと、EDO モードによるノーマルアクセス / バーストアクセスを選択することができます。また、EDO モードの DRAM へのアクセスは、 $\overline{\text{OE}}$  制御を行っていますので、必ず本 LSI の  $\overline{\text{CAS}}/\overline{\text{OE}}$  を DRAM の  $\overline{\text{OE}}$  端子に接続してください。

EDO モードによるノーマルアクセスを図 7.48 に、バーストアクセスを図 7.49 に示します。

## 7. バスステートコントローラ (BSC)

EDO モードでは、リードサイクル時にデータバスにデータを出力しているタイミングが次の  $\overline{\text{CAS}}_n$  のアサートまで延びるため、データの取り込みタイミングを 1/2 サイクル遅くし、CKIO クロックの立ち上がりとする事により、DRAM に対するアクセスタイムを増やすことができます。

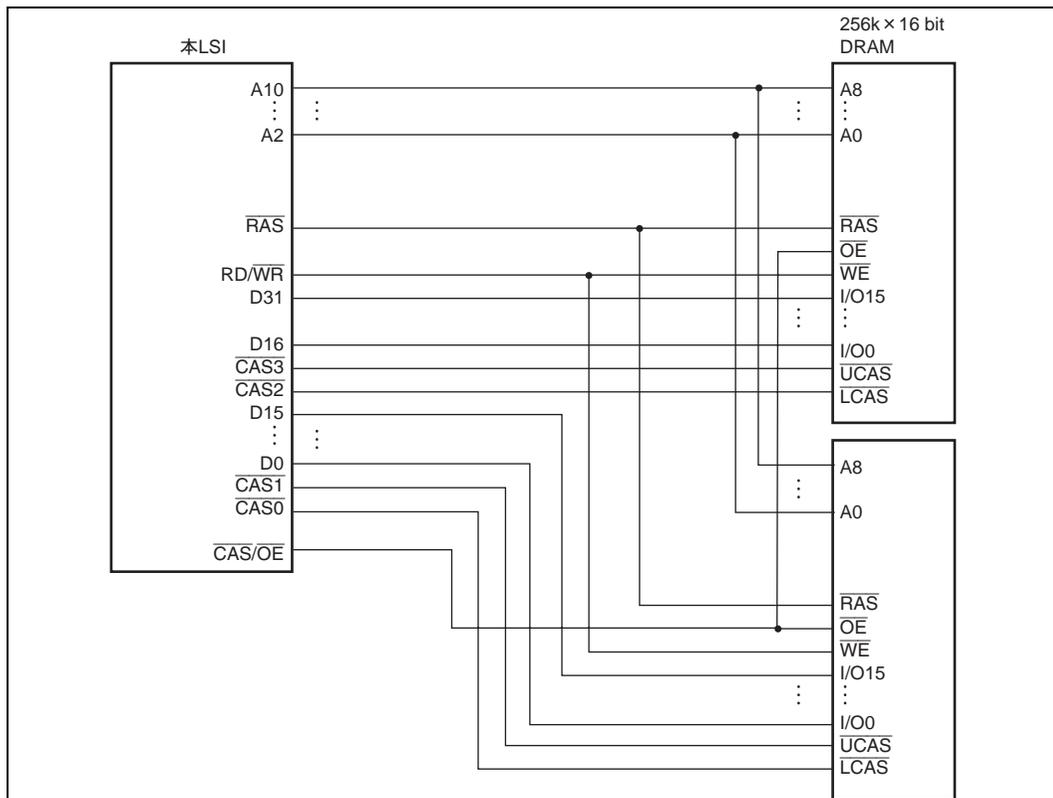


図 7.46 EDO DRAM 接続例 (32 ビットデータ幅)

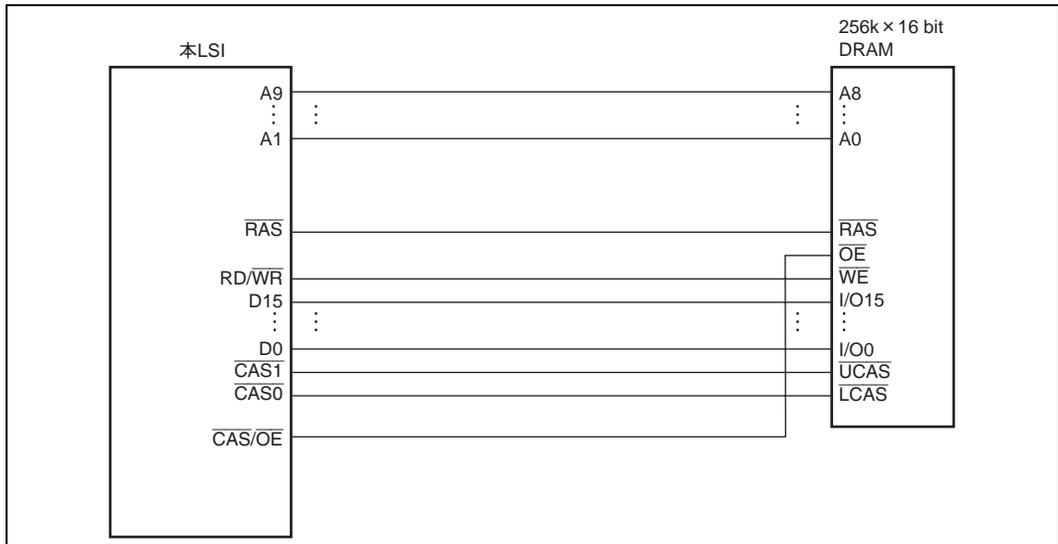


図 7.47 EDO DRAM 接続例 (16 ビットデータ幅)

7. バスステートコントローラ (BSC)

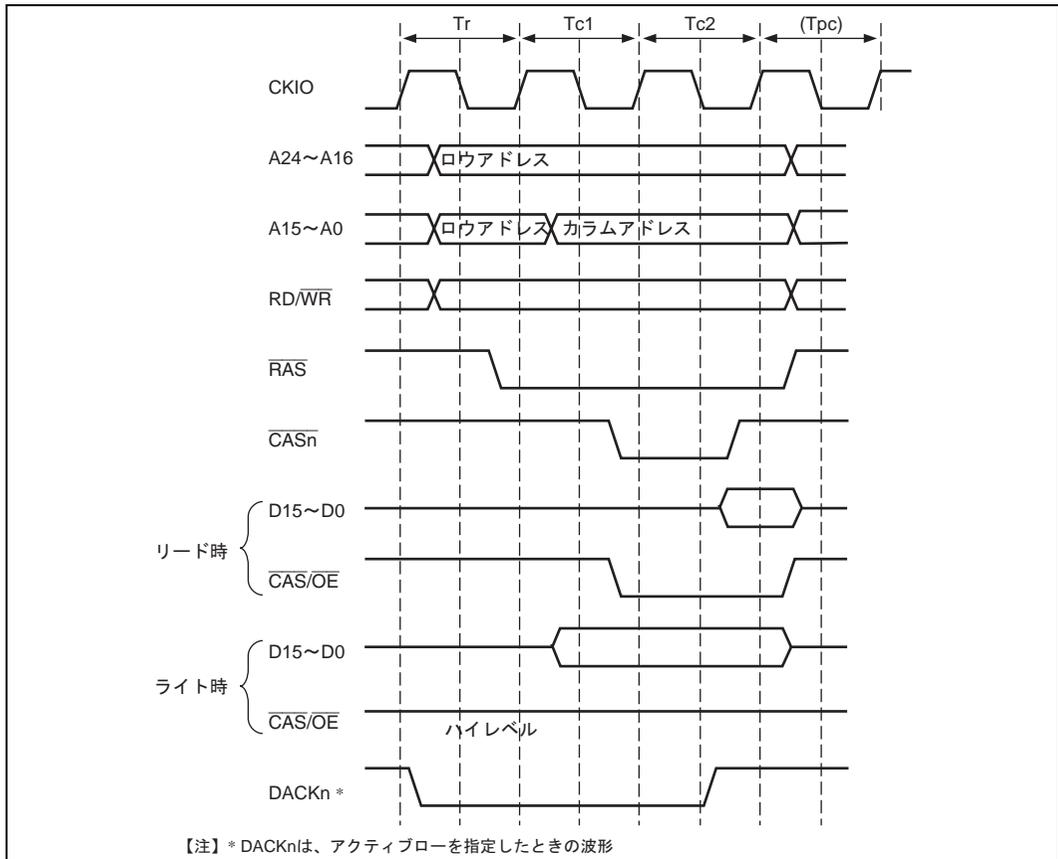


図 7.48 DRAM EDO モードのノーマルアクセスタイミング

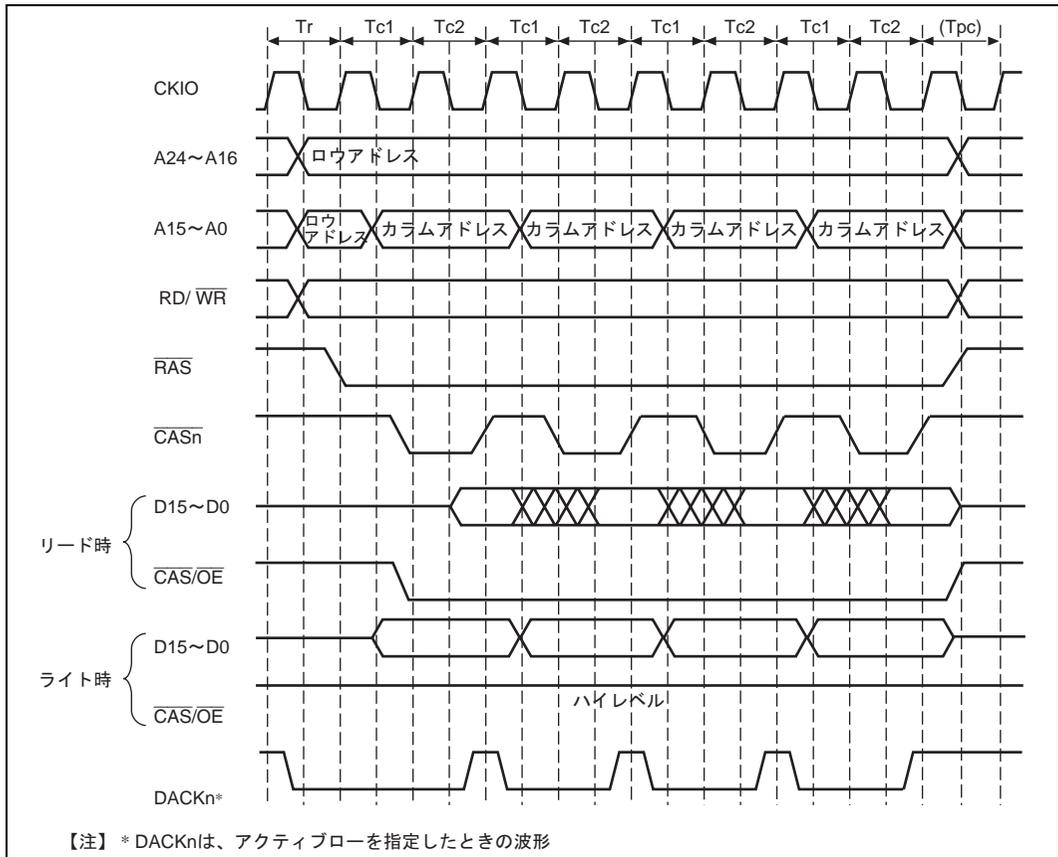


図 7.49 DRAM EDO モードのバーストアクセスタイミング

### 7.6.7 DMA シングル転送

本 LSI では、DMA シングルアドレス転送モードでの書き込み時の DACKn アサートから、 $\overline{\text{CASn}}$  アサートの間に BCR3 の DSWW1、DSWW0 を設定値分のウェイトステートを挿入することができます。ウェイトステートを挿入することにより、外部デバイスのメモリに対するデータのセットアップタイムを設けることができます。図 7.50 に DSWW1、DSWW0=01、RASD=1 に設定時の DMA シングル転送モードでのライトサイクルタイミングを示します。DMA シングル転送モードでのリードサイクルは、CPU あるいは DMA デュアル転送モードによるリードサイクルと同様です。

## 7. バスステートコントローラ (BSC)

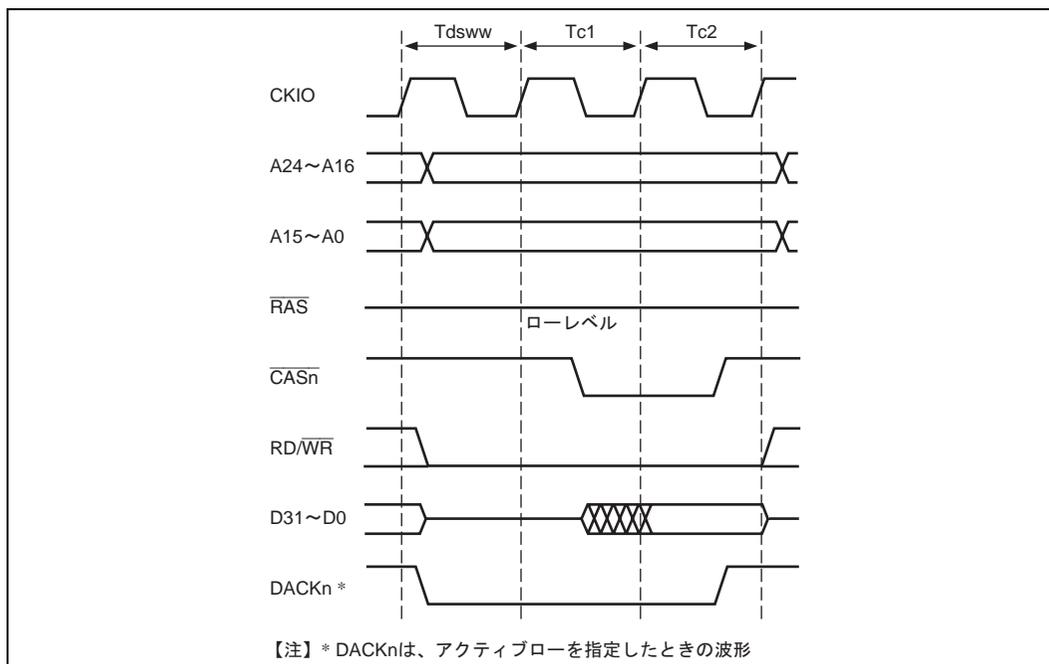


図 7.50 DMA シングル転送モードでのライトサイクルタイミング ( $\overline{\text{RAS}}$  ダウンモード、同一ロウアドレス)

### 7.6.8 リフレッシュ

バスステートコントローラは、DRAM のリフレッシュを制御する機能を備えています。MCR の RMODE ビットを 0 に、RFSH ビットを 1 にセットすることによって、 $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュサイクルによる分散リフレッシュを行わせることができます。また、RTCSR の RRC2 ~ 0 を設定することにより、連続してリフレッシュを発生させることができます。さらに、長時間 DRAM にアクセスしないときには、RMODE ビットと RFSH ビットをともに 1 にすることによって、データ保持のための消費電力が少ないセルフリフレッシュモードを起動させることができます。

#### (1) $\overline{\text{CAS}}$ ビフォ $\overline{\text{RAS}}$ リフレッシュ

RTCSR の CKS2 ~ 0 ビットで選択した入力クロックと、RTCOR に設定した値とで決まる間隔でリフレッシュが行われます。使用する DRAM のリフレッシュ間隔規定を満たすように、RTCOR と CKS2 ~ 0 ビットの値を設定してください。最初に RTCOR、RTCNT と MCR の RMODE ビットおよび RFSH ビットの設定を行い、最後に RTCSR の CKS2 ~ CKS0 と RRC2 ~ 0 の設定を行ってください。CKS2 ~ CKS0 によってクロックを選択すると、RTCNT はそのときの値からカウントアップを開始します。RTCNT の値は常に RTCOR の値と比較されており、両者の値が一致するとリフレッシュ要求が発生し、RRC2 ~ 0 の設定回数の  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュが行われます。同時に RTCNT はゼロクリアされ、カウントアップが再開されます。図 7.51 に  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュサイクルのタイミングを示します。

リフレッシュサイクルでの  $\overline{\text{RAS}}$  アサートサイクル数は、MCR の TRAS1、TRAS0 で指定されます。リフレッシュサイクルにおける  $\overline{\text{RAS}}$  のプリチャージ時間の指定は、通常アクセスと同様 MCR の TRP1、TRP0 ビットの指定にしがいます。

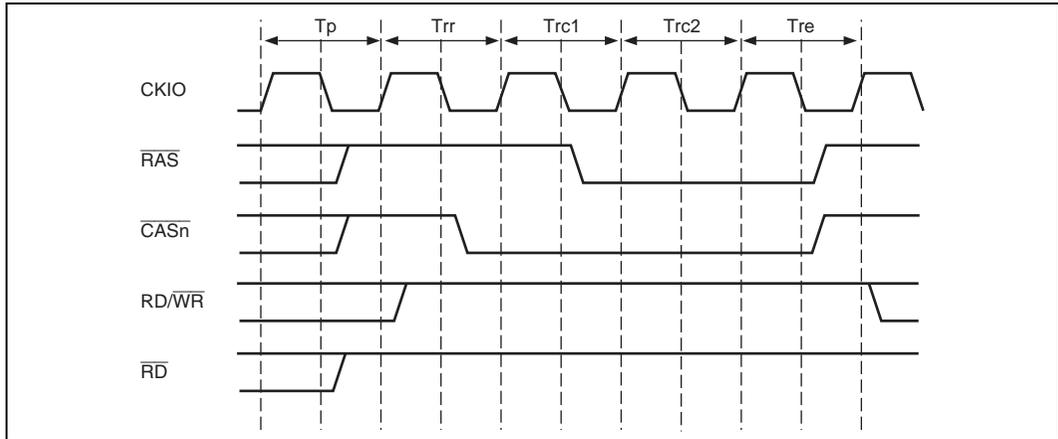


図 7.51 DRAM  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュサイクルタイミング

## (2) セルフリフレッシュ

セルフリフレッシュは RMODE ビットと RFSH ビットをともに 1 にすることによって起動します。セルフリフレッシュの状態の間は、DRAM にアクセスすることができません。セルフリフレッシュの解除は RMODE ビットを 0 にすることによって行われます。セルフリフレッシュのタイミングを図 7.52 に示します。セルフリフレッシュ解除、データ保持が正しく行われるように、直ちに  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュが正しい間隔で行われるように設定を行ってください。  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュの設定をしている状態からセルフリフレッシュにした場合、マニュアルリセットまたは NMI でスタンバイモードを脱出する場合、セルフリフレッシュ解除時に RFSH=1、RMODE=0 の設定にすればオートリフレッシュが再開されます。セルフリフレッシュ解除からオートリフレッシュ開始までに時間がかかる場合には、この時間を考慮して RTCNT の初期化の設定を行ってください。 RTCNT の値を RTCOR の値 -1 に設定すると直ちにリフレッシュを開始することができます。

セルフリフレッシュに設定した後、本 LSI のスタンバイ機能を使ってチップをスタンバイ状態にした場合にもセルフリフレッシュ状態は継続され、スタンバイからの復帰が NMI による場合には復帰後もセルフリフレッシュ状態から抜け出ることはありません。

パワーオンリセットの場合にはバスステートコントローラのレジスタが初期化されるため、セルフリフレッシュ状態から抜け出します。

## 7. バスステートコントローラ (BSC)

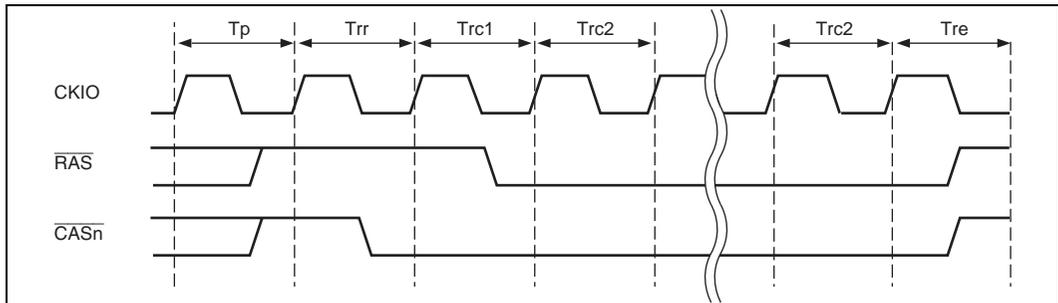


図 7.52 DRAM セルフリフレッシュサイクルタイミング

### 7.6.9 パワーオンシーケンス

電源投入後の DRAM の使用に関しては、アクセスの行えない待機時間 (100  $\mu$ s または 200  $\mu$ s 以上) とそれに続く所定回数 (通常 8 回) 以上のダミーの  $\overline{\text{CAS}}$  ビフォ  $\overline{\text{RAS}}$  リフレッシュサイクルを行うことが要求されています。バスステートコントローラは、パワーオンリセットに対してなんら特別な動作を行わないため、必要なパワーオンシーケンスはパワーオンリセット後に実行する初期化プログラムによって実現する必要があります。

## 7.7 バースト ROM インタフェース

BCR1 の BSTROM ビットを 1 に設定することにより、CS0 空間にバースト ROM の接続を可能となります。バースト ROM インタフェースは、ニブルアクセス機能を有する ROM に高速にアクセスするためのものです。バースト ROM に対するニブルアクセスのタイミングを図 7.54 に示します。ウェイトサイクル 2 サイクルの設定です。基本的には通常空間と同じようなアクセスを行います。最初のサイクルを終了する際  $\overline{CS0}$  信号のネゲートを行わず、アドレスのみを切り替えて、さらに通常空間アクセスの T1 サイクルを省略して次のアクセスを行います。2 回目以降は T1 サイクルが省略されるので、通常のアクセスよりも 1 サイクル高速にアクセスできます。現在 ROM として入手可能なものは、ニブルアクセスの対象として 4 アドレスのみとなっています。そこで、この機能を適用するのは、8 ビット幅の ROM に対するワードまたはロングワードリードと 16 ビット幅の ROM に対するロングワードリードのときに限定します。マスク ROM はアクセス速度も遅くアクセス回数も 8 ビット幅の場合命令フェッチで 4 回、キャッシュフィルでは 16 回と多いので、これを救済する目的でニブルアクセスのサポートを行います。8 ビット幅の ROM を接続する場合には最大 4 回の連続アクセスが行われ、16 ビット幅の ROM を接続する場合には最大 2 回の連続アクセスが行われます。データ幅とアクセスサイズの関係を図 7.53 に示します。キャッシュフィルおよび DMAC の 16 バイト転送はロングワードアクセスを 4 回繰り返します。

バースト ROM アクセスでもウェイトステートを 1 以上に設定すると  $\overline{WAIT}$  端子のサンプリングを行います。

バースト ROM の設定を行いウェイト指定を 0 にした場合も 2 回目以降のアクセスサイクルは 2 サイクルとなります。この場合のタイミングを図 7.55 に示します。

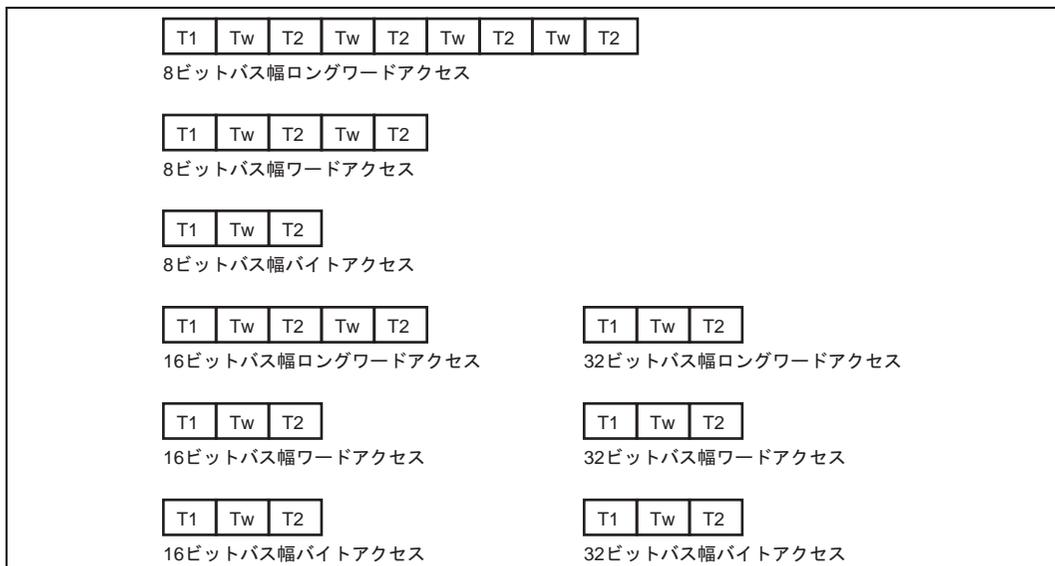


図 7.53 データ幅とバースト ROM アクセス (ウェイト数 1)

7. バスステートコントローラ (BSC)

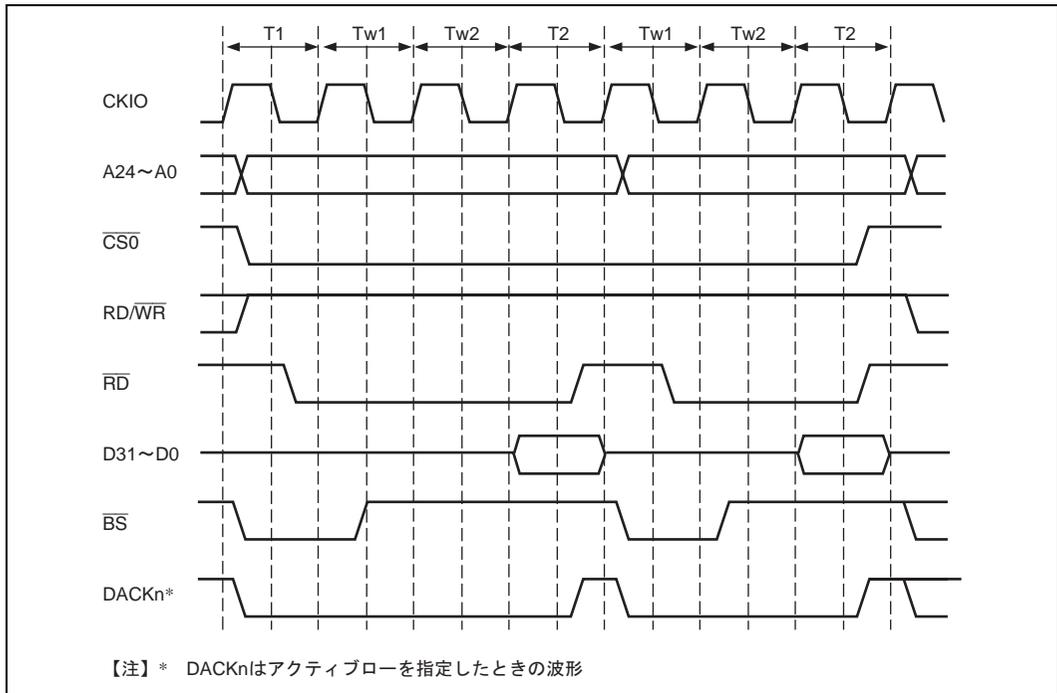


図 7.54 バーストROM ニブルアクセス (2ウェイトスタート)

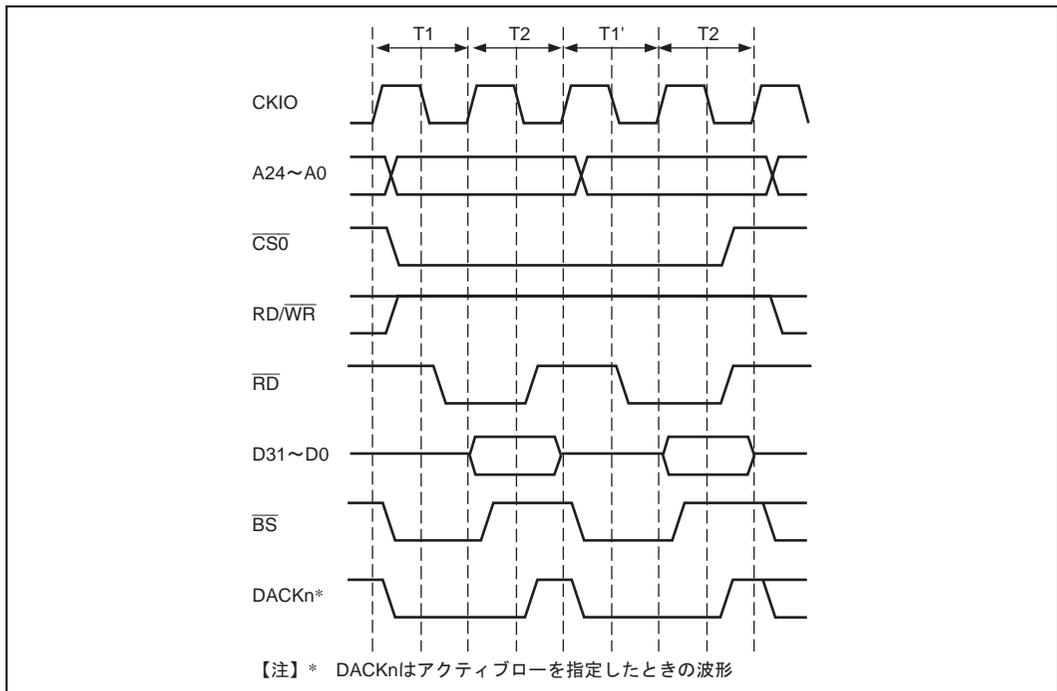


図 7.55 バーストROM ニブルアクセス (ウェイトなし)

## 7.8 アクセスサイクル間ウェイト

動作周波数が高くなってきたため、低速なデバイスからのリードが完了した際のデータバッファのオフが間に合わず、次のアクセスのデータと衝突してデバイスの信頼度を低下させたり、誤動作を引き起こすという事象が起こるようになってきました。これを防止するため、直前のアクセスの空間とリード/ライトの別を記憶しておき、次のアクセスを起動する際に問題があるケースではアクセスサイクル間のウェイトを挿入して、データの衝突を回避する機能を設けました。チェックの対象としては、直前のアクセスがリードサイクルでそれに続いて異なる CS 空間のリードアクセスを行う場合と、直前のアクセスがリードアクセスで、次のアクセスが本 LSI からのライトの 2 つのケースについてです。本 LSI がライトを連続している場合には、データの方向は常に本 LSI から他のメモリという形で統一されており、特に問題とはなりません。次に同一の CS 空間に対するリードアクセスは、原則として同一のデータバッファからデータが出されるものとして、これも対象から外します。WCR1 の IW31、IW30 によって CS3 空間からリードした後、他の CS 空間から読み出しを行う場合と、本 LSI がライトアクセスを行う場合のアクセスサイクルの間に挿入するアイドルサイクル数を指定します。同様に IW21、IW20 は CS2 リード後の、IW11、IW10 は CS1 リード後の、IW01、IW00 は CS0 リード後のアイドルサイクル数を指定します。また、WCR2 の IW41、IW40 は CS4 リード後のアイドルサイクル数を指定します。0、1、2、4 サイクルの指定ができます。アクセス間に、もともと空きがある場合には、指定されたアイドル数からその空きサイクル数を除いたサイクルだけ、アイドルサイクルの挿入を行います。リードアクセスを行った後直ちにライトサイクルを行う場合、サイクル間ウェイトの指定が 0 の場合でも 1 サイクルのウェイトサイクルを挿入します。

本 LSI がライトを行った後直ちにリードサイクルに移る場合、ライトデータはクロックの立ち上がり時にハイインピーダンスにされるのに対し、リードサイクルのデータ出力許可を示す  $\overline{RD}$  信号は、クロックの立ち下がりまでアサートされないため、アクセスサイクル間ウェイトは挿入しません。

バスアービトレーションを行う場合には、アービトレーションのための空きサイクルが入るため、サイクル間ウェイトは入りません。

7. バスステートコントローラ (BSC)

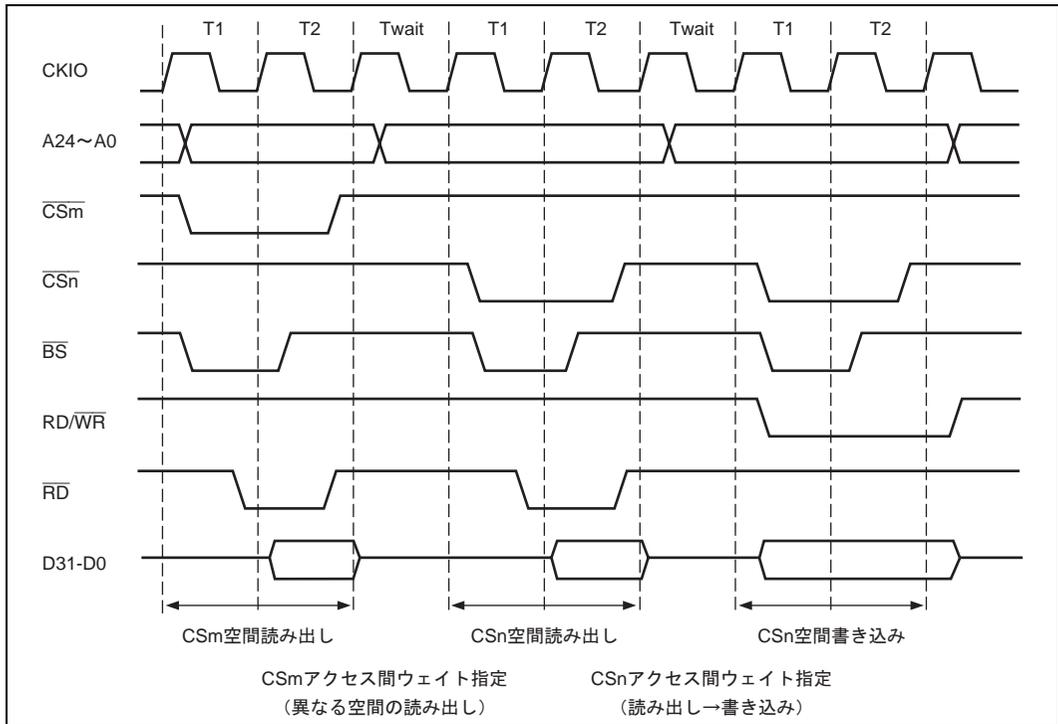


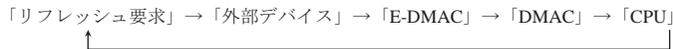
図 7.56 アクセスサイクル間ウェイト

## 7.9 バスアービトレーション

本 LSI には、外部デバイスからバス権の要求があると、そのデバイスにバス権を与えるバスアービトレーション機能が備わっています。

本 LSI は定常状態でバス権を有し、他のデバイスからのバス権使用要求を受けてバスの解放を行い、バスの使用許諾を行います。以下の説明ではバス権要求を行う外部デバイスをスレーブと呼びます。

本 LSI の内部には CPU、DMAC および E-DMAC という 3 つのバスマスタがあります。また、シンクロナス DRAM、DRAM を接続し、リフレッシュ制御を行わせる場合、リフレッシュ要求は第 4 のバスマスタとなります。これらに加え、外部デバイスからのバス権要求が加わります。本 LSI のバスアービトレーションは、ラウンドロビン方式を採用しております。要求が同時に発生した場合のバス権の遷移は、次のようになります。



ただし、「E-DMAC」については、バス権が一周する間に占有できるチャンネル数は 1 チャンネルとします。

E-DMAC は、送信・受信合わせて 2 チャンネルあります。チャンネル間の調停は、E-DMAC 内部で自動的に行っており、送信チャンネルと受信チャンネルとの間で交互にバス権が遷移します。また、DMAC 2 チャンネル内の調停については、DMA オペレーションレジスタ (DMAOR) 内のプライオリティモードビット (PR) の設定により、優先順位固定またはラウンドロビンモードを選択することができます。

マスタとスレーブとの間でバスを受け渡す際、接続されているデバイスの誤動作を防ぐため、バス解放に先立ってすべてのバス制御信号はネゲート状態とします。バス権を受け取る場合にも、バス制御信号はネゲート状態からバスのドライブを開始します。バス権を受け渡すマスタとスレーブで同じ値に信号をドライブするので、出力バッファの衝突は回避できます。バス制御信号のハイインピーダンス状態での外来ノイズによる誤動作を防ぐために、これら制御信号にはプルアップ抵抗が必要となります。

バス権の委譲は内部バスのサイクルの切れ目で行われています。外部デバイスにバス権を要求されたとき、他のマスタによるバス権要求がなく、かつバスサイクルを行っていない場合は、直ちにバス権の解放を行います。しかし、他のマスタによるバス権要求やバスサイクルがあるときは、内部のバス権要求が満たされるまで、バス解放は待たされます。LSI 外部から見るとバスサイクルを行っていない場合でも、アクセスサイクル間ウェイトを挿入するなど、内部的にはバスサイクルが開始されている場合があるため、 $\overline{CS}_n$  信号その他のバス制御信号を見て、直ちにバス権が解放されるかどうかを判定することはできません。さらに、キャッシュフィルのためのバースト転送、DMAC の 16 バイトブロック転送 (デュアルアドレスの場合は、 $16 + 16 = 32$  バイト転送)、E-DMAC の 16 バイトブロック転送の途中のバス解放等はいりません。同様に TAS 命令のリードサイクルとライトサイクルの間でもバス解放は行いません。データバス幅がアクセスサイズよりも小さいことによって生ずる複数バスサイクル、例えば 8 ビットデータ幅のメモリにロングワードアクセスを行う場合のバスサイクル間にもアービトレーションは行いません。一方独立したアクセスである割り込み処理時の外部ベクタフェッチ、PC の退避、SR の退避サイクルの間には、バスアービトレーションが行われます。

本 LSI 内部の CPU は、キャッシュメモリとの間を専用の内部バスで接続しているため、LSI 内部または外部の他のバスマスタがバスを使用している場合でも、キャッシュメモリからの読み出しを行うことができます。CPU からの書き込みの場合は、外部に対する書き込みサイクルが必ず生じます。

## 7. バスステートコントローラ (BSC)

---

また、CPU、DMAC と内蔵周辺モジュールを結ぶ内部バスも外部バスと平行して動作が可能のため、外部のバス権がなくとも CPU から内蔵周辺モジュール、DMAC から内蔵周辺モジュールへのアクセスは読み出し、書き込みともに行うことができます。

E-DMAC、DMAC および CPU から同時にバス権要求が発生した場合の例として図 7.57 (a) および図 7.57 (b) にタイミングチャートを示します。なお、本例は以下に示す設定を行った場合のものです。

- CS2、CS3 空間は、シンクロナス DRAM
- CAS レイテンシは、1 サイクル
- E-DMAC は送信側、受信側ともイネーブル (バッファおよびディスクリプタは CS3 空間を使用)
- DMAC は、オートリクエストモード、サイクルスチールモード、16 バイトデュアルアドレス転送に設定した 1 チャンネルのみイネーブル (CS2 空間)
- シンクロナス DRAM はバーストリード、シングルライトの設定

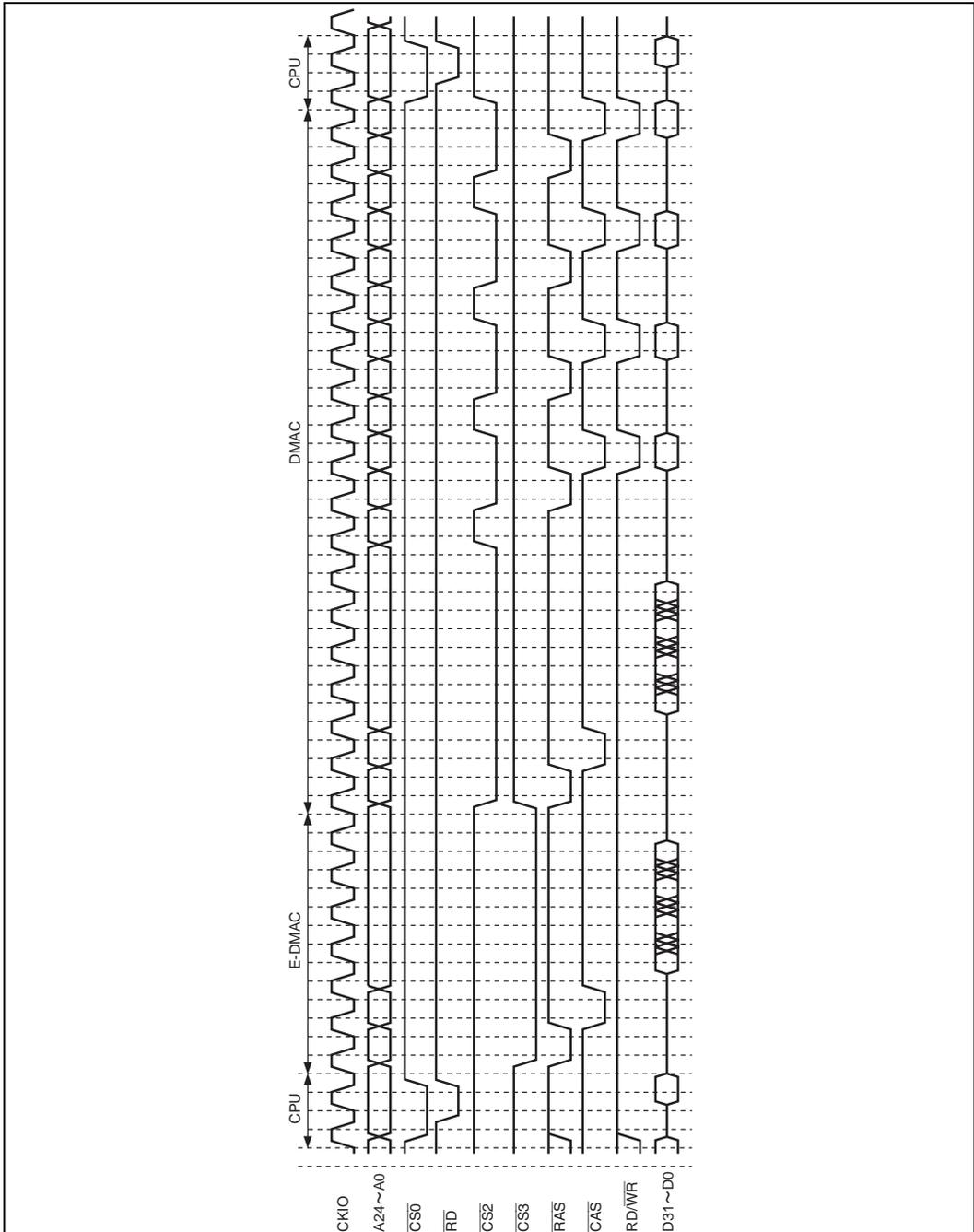


図 7.57 (a) バスアービトラシオンタイミング (E-DMAC リード DMAC16 バイト転送 CPU リード)

7. バスステートコントローラ (BSC)

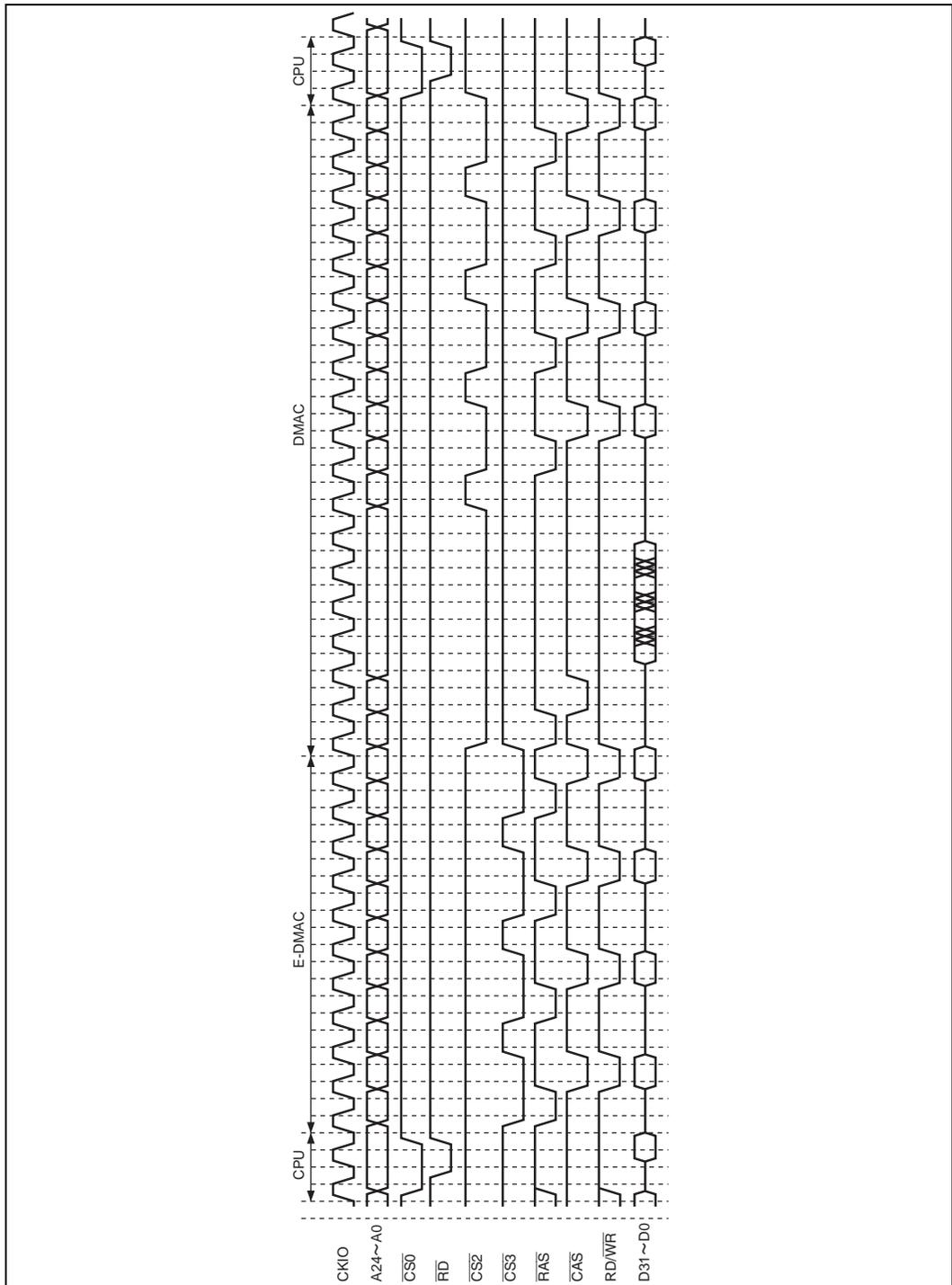


図 7.57 (b) バスアービトラクションタイミング (E-DMAC ライト DMAC16 バイト転送 CPU リード)

### 7.9.1 マスタモード

本 LSI はバス権要求を受けない限り自分でバスを保有しています。

外部からのバス権要求 ( $\overline{\text{BRLS}}$ ) のアサート (L レベル) を受け、実行中のバスサイクルが終わり次第バスの解放を行いバス使用許可 ( $\overline{\text{BGR}}$ ) をアサート (L レベルに) します。スレーブがバスを解放したことを示す  $\overline{\text{BRLS}}$  のネゲート (H レベル) を受けて  $\overline{\text{BGR}}$  をネゲート (H レベルに) し、バスの使用を再開します。バス解放時はシンクロナス DRAM インタフェースの CKE とバスアービトレーションの  $\overline{\text{BGR}}$ 、および DMA 転送を制御する DACK0、DACK1 を除き、バスインタフェースに関連するすべての出力信号および入出力信号をハイインピーダンスとします。

DRAM はプリチャージを完了させてからバスを解放します。シンクロナス DRAM も、アクティブとなっているバンクに対してプリチャージコマンドを発行し、これを完了させた後、バスの解放を行います。

具体的なバス解放シーケンスは次の通りです。まず、クロックの立ち上がり同期してアドレスバスおよびデータバスをハイインピーダンスにします。この 0.5 サイクル後にバス使用許可信号をクロックの立ち下がり同期してアサートします。これに続くクロックの立ち上がりで、バス制御信号 ( $\overline{\text{BS}}$ 、 $\overline{\text{CSn}}$ 、 $\overline{\text{RAS}}$ 、 $\overline{\text{CASn}}$ 、 $\overline{\text{WEn}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{RD}} / \overline{\text{WR}}$ ) をハイインピーダンスにします。これらのバス制御信号は、遅くともハイインピーダンスにする 1 サイクル前には H レベルにされています。バス権要求信号のサンプリングはクロックの立ち下がりで行います。

スレーブからバス権を再獲得するときのシーケンスは次の通りです。

$\overline{\text{BRLS}}$  のネゲートをクロックの立ち下がりで検出すると、0.5 サイクル後にバス制御信号は H レベルでドライブを開始します。これに続くクロックの立ち下がりで、バス使用許可信号をネゲートします。アドレスバスおよびデータバスのドライブを開始するのはこれに続くクロックの立ち上がりです。バス制御信号をアサートしてバスサイクルを実際に開始するのは、最も早い場合にはアドレス、データ信号をドライブするのと同じクロックの立ち上がりからです。バスアービトレーションタイミングは図 7.58 を参照してください。

ユーザが個別に設計したスレーブによっては、アービトレーションによるオーバーヘッドを減少させるため、連続して複数回のバスアクセスを発生しようとする場合があります。このようなケースでは、確実にリフレッシュを行うため、スレーブのバス占有時間がリフレッシュ周期を超えることなくバス権を解放するように設計を行う必要があります。そこで、本 LSI では、リフレッシュの実行が待たされている間にバス権を要求する信号として REFOUT 端子を用意しています。そして、リフレッシュの実行が待たされている間バス権が獲得するまで、REFOUT をアサートします。外部のスレーブデバイスがこれを受けてバス権を解放すれば、バス権が本 LSI に戻り、リフレッシュを実行することができます。

## 7. バスステートコントローラ (BSC)

---

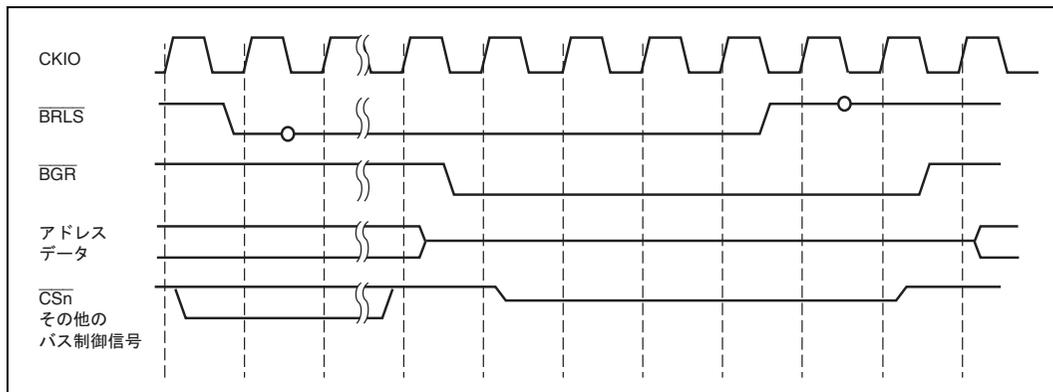


図 7.58 バスアービトレーション

## 7.10 その他

### 7.10.1 リセット

バスステートコントローラは、パワーオンリセットでのみ完全な初期化が行われます。バスサイクルの途中であるなしにかかわらず、直ちにすべての信号をネゲートします。信号のネゲートと出力バッファのオフは同時に行われます。制御レジスタはすべて初期化されます。スタンバイ、スリープおよびマニュアルリセットではバスステートコントローラの制御レジスタの初期化は行われません。マニュアルリセットが行われると、現在実行中のバスサイクルはそのバスサイクルに限り終了まで実行され、あとはアクセス待ちの状態になります。キャッシュフィルや DMAC、E-DMAC の 16 バイト転送を実行中の場合、バスマスタである CPU または DMAC、または E-DMAC がマニュアルリセットによってアクセス要求を取り消すため、ロングワード単位でアクセスが打ち切られます。このためキャッシュフィル時にマニュアルリセットを入れる場合、キャッシュの内容は保障されなくなりますので注意してください。マニュアルリセット中は RTCNT のカウントアップが行われなため、リフレッシュ要求が発生せず、リフレッシュサイクルは起動されません。DRAM、シンクロナス DRAM のデータを保証するためには、マニュアルリセットのパルス幅をリフレッシュ間隔よりも短くする必要があります。本 LSI はマニュアルリセット信号のアサート中でもアービトレーション要求を受け付けます。バス解放中にマスタモードのチップのみにリセットが入った場合、これを表示するため  $\overline{BGR}$  信号をネゲートします。 $\overline{BRLS}$  信号が引き続きアサートされている場合には、そのままバス解放状態を保ちます。

### 7.10.2 CPU、DMAC および E-DMAC からみたアクセス

本 LSI の内部はキャッシュバス、内部バス、周辺バスの 3 つのバスに分割されています。CPU およびキャッシュメモリはキャッシュバスに、DMAC、E-DMAC およびバスステートコントローラは内部バスに、低速な周辺機器とモードレジスタは周辺バスにそれぞれ接続されています。また、キャッシュメモリ以外の内蔵メモリとユーザブ레이크コントローラはキャッシュバスと内部バスの双方に接続されています。キャッシュバスから内部バスのアクセスは行えますが逆は行えません。内部バスから周辺バスのアクセスは行えますが、周辺バスから内部バスのアクセスは行えません。このため以下のようなことが発生します。

DMAC からキャッシュメモリ以外の内蔵メモリへのアクセスは行えますが、キャッシュメモリへのアクセスは行えません。DMAC によって外部メモリへの書き込みが生じ、この結果として外部メモリの内容とキャッシュの内容に食い違いが生ずることがあります。DMA 転送によって外部メモリの内容を書き換えた場合、その番地のデータがキャッシュにある可能性がある場合には、キャッシュメモリのページをソフトウェアで行わなければならない。

CPU が読み出しアクセスを開始し、それがキャッシュ領域の場合にはまず 1 サイクルだけキャッシュの検索が行われます。キャッシュにデータが保持されているとこれを取り込みアクセスは完了します。キャッシュ内にデータがない場合には、内部バスを介してキャッシュデータのフィルを行うため、4 つの連続したロングワードリードが起動されます。バイトまたはワードオペランドアクセス時および奇数ワード境界 ( $4n+2$  番地) へのブランチ時のミスヒットに関しても、チップ外部インタフェース上は必ずロングワードアクセスでフィルを行います。キャッシュスルー領域に関しては、実際のアクセスアドレスにしたがってアクセスを行います。ただし、アクセスが命令フェッチの場合にはアクセスサイズは常にロングワードとなります。

## 7. バスステートコントローラ (BSC)

---

キャッシュスルー領域および内蔵周辺モジュールの読み出しサイクルの場合、その判定に1サイクル費やした後内部バスを介して読み出しサイクルが起動されます。読み出しデータはキャッシュバスを經由してCPUに送られます。

書き込みサイクルで、キャッシュ領域に対するものであった場合、キャッシュの検索を行い該当アドレスのデータがあった場合にはここに書き込みを行います。ライトスルーモード時には、これと平行して内部バスを經由して実際の書き込みが行われます。ライトバックモード時には、該当アドレスのリプレースが発生するまで実際の書き込みが行われません。内部バスの使用権がある場合には、チップ外部への実際の書き込みが終了するまで待つことなくCPUに対して書き込みの完了が通知されます。DMAC等に用いられるなどして内部バスの使用権がない場合にはバスの使用権が獲得できるまで待ちCPUに完了を通知します。

キャッシュスルー領域および内蔵周辺に対するアクセスは、キャッシュの検索および書き込みを除いてキャッシュ領域と同様の動きをします。

バスステートコントローラには1段のライトバッファがあるため、ライトサイクルではチップ外部のバスサイクルが完了しなくても内部バスを別のアクセスに使用することができます。チップ外部の低速メモリに対して書き込みを行った後、内蔵周辺に対する読み出しまたは書き込みを行う場合、低速メモリへの書き込みの完了を待たずに内蔵周辺へのアクセスが可能です。

読み出しでは、常に動作の完了までCPUは待たされるので、データの実際のデバイスに対する書き込みの完了を確認してから処理を続行したい場合、続けて同じアドレスに対するダミーの読み出しアクセスを行うと書き込みの終了が確認できます。

DMACからのアクセスでも同様にバスステートコントローラのライトバッファは働きます。したがってデュアルアドレスのDMA転送を行う場合、書き込みサイクルの完了を待たずに次の読み出しサイクルの起動がかけられます。ただし、DMAのソースアドレスとディスティネーションアドレスがともに外部空間であった場合には、前の書き込みサイクルが完了するまで次の読み出しサイクルの開始が待たされます。

E-DMACは、外部メモリとのアクセスは行えますが、全ての内蔵メモリと周辺モジュールとのアクセスは行えません。

### 7.10.3 STATS1、0 端子について

本 LSI は、バスマスタの状態を識別するために、STATS1 および 0 の 2 本の端子を設けています。本端子から出力される信号は、外部アクセス状態を表しています。CPU (キャッシュミスヒット/キャッシュディスイネーブル時) DMAC (外部アクセスのみ) E-DMAC、その他 (リフレッシュ、内部アクセス他) をエンコード出力しており、全てアドレス信号に同期して出力しています。表 7.9 にエンコード結果と、図 7.59 に出力タイミングを示します。

表 7.9 エンコード結果

識別	STATS1	STATS0
CPU	0	0
DMAC		1
E-DMAC	1	0
その他		1

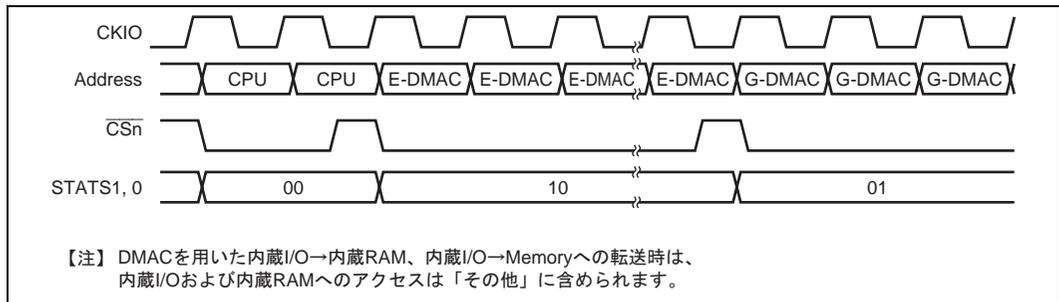


図 7.59 STATS 出力タイミング

### 7.10.4 $\overline{\text{BUSHiZ}}$ の仕様について

$\overline{\text{BUSHiZ}}$  端子は、本 LSI を PCI ブリッジを介して PCI コントローラに接続し、かつ PCI マスタと本 LSI が、本 LSI バス上のローカルメモリを共有する場合に必要とされます。本端子は、 $\overline{\text{WAIT}}$  と組み合わせて使用することで、SH7615 内部ステートを止めたまま、バスおよび特定の制御信号を HiZ にすることができます。以下にバス HiZ の成立条件、対象端子、およびバスタイミング (図 7.60) を示します。なお、PCI ブリッジとの接続例は、アプリケーションノートを参照してください。

- 成立条件 :  $\overline{\text{WAIT}} = \text{L}$  かつ  $\overline{\text{BUSHiZ}} = \text{L}$  で、このときの BCR の設定などに依存しない。
- 対象端子 : A [24 : 0]、D [31 : 0]、 $\overline{\text{CS3}}$ 、RD/WR、 $\overline{\text{RD}}$ 、 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS/OE}}$ 、DQMLL/ $\overline{\text{WE0}}$ 、DQMLU/ $\overline{\text{WE1}}$ 、DQMUL/ $\overline{\text{WE2}}$ 、DQMUU/ $\overline{\text{WE3}}$  (計 66 本)

## 7. バスステートコントローラ (BSC)

---

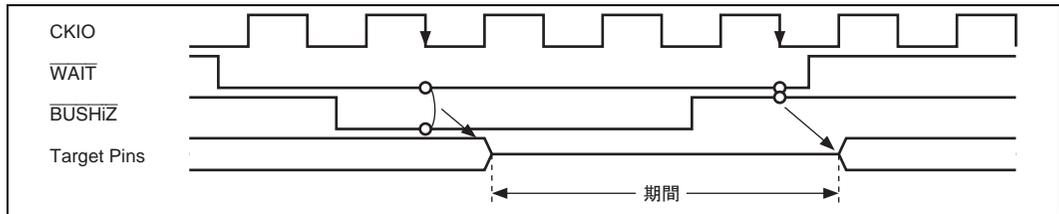


図 7.60 BUSHiZ バスタイミング

1. CPU と外部デバイスに共有するメモリがある場合に使用することができます。
2. このとき  $\overline{\text{WAIT}}$  端子をアサートしておいて  $\overline{\text{BUSHiZ}}$  をアサートすると、CPU が見かけ上バスを離れた状態となります。
3. 共有するメモリのアクセスを行なえる状態になったら  $\overline{\text{BUSHiZ}}$  をネゲートします。
4. データの用意ができたなら  $\overline{\text{WAIT}}$  信号をネゲートします。

上記により CPU と外部デバイスでメモリを共有することができます。

## 7.11 使用上の注意

### 7.11.1 DMAC 使用時のシンクロナス DRAM ライト直後の通常空間アクセス時の注意事項

シンクロナス DRAM ライト時の  $\overline{DQMn}/\overline{WE_n}$  信号のネゲートと、その直後の通常空間アクセス時の  $\overline{CS_n}$  アサートが同一の CKIO 立ち上がりエッジになります (図 7.61)。この場合、通常空間に対して誤書き込みを行う可能性がありますので、通常空間にシンクロナス DRAM や高速デバイスを接続する場合は、システム側で  $\overline{CS_n}$  を遅延させることを推奨します。

シンクロナス DRAM ライトと通常空間アクセスが連続する場合を表 7.10 に示します。

表 7.10 アクセスシーケンス

シンクロナス DRAM へのライト	通常空間アクセス
CPU	DMA
DMA	CPU
DMA	DMA

【注】 CPU でライトした直後に CPU でアクセスした場合は、内部的にアクセスが連続しません。

## 7. バスステートコントローラ (BSC)

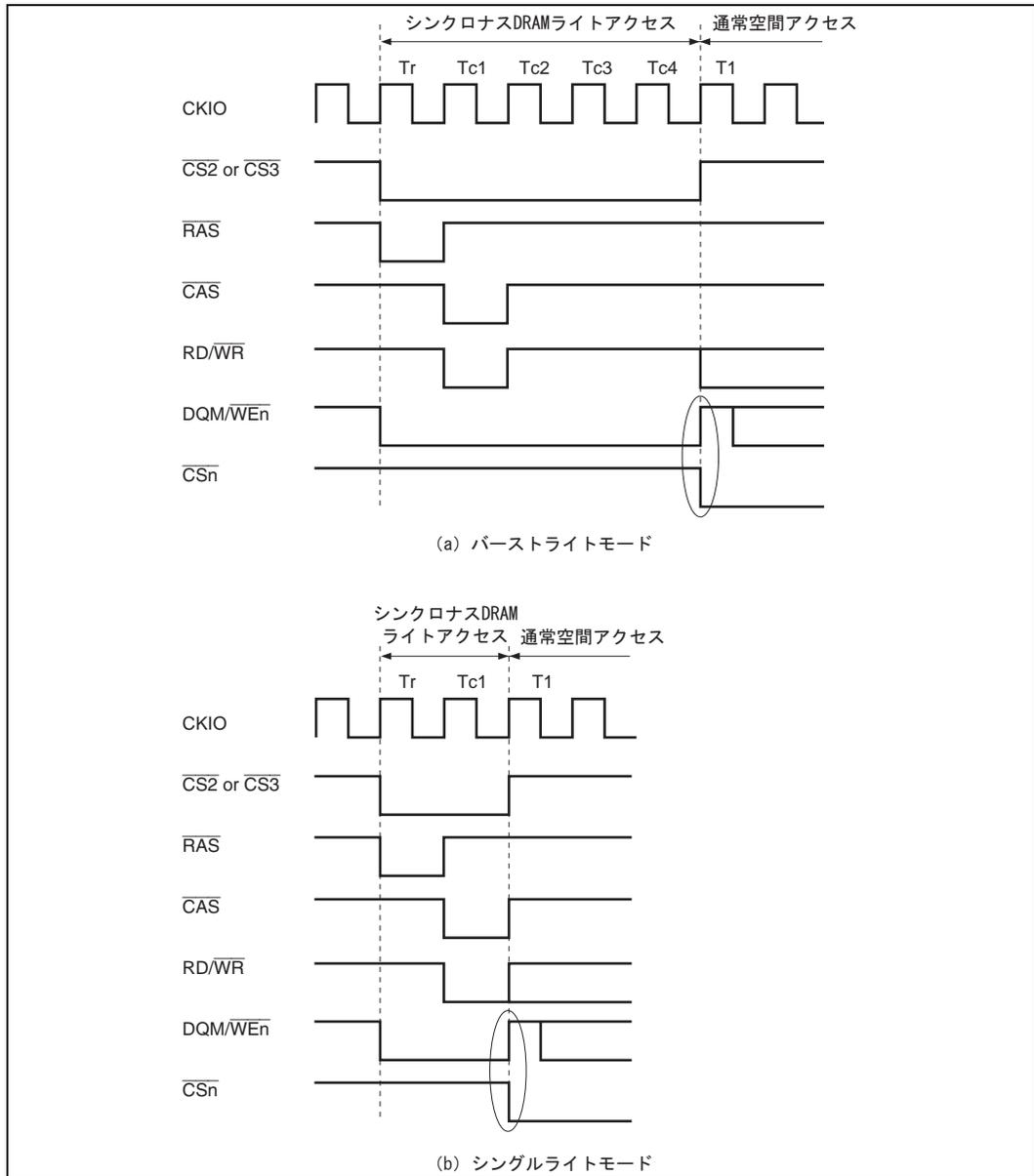


図 7.61 シンクロナス DRAM ライト直後の通常空間アクセス

### 7.11.2 クロック比 I : E = 1 : 1、バス幅 8 ビット、外部ウェイト入力時の注意事項

クロック比 I : E = 1 : 1、バス幅 8 ビットで使用する場合には、アドレスホールドサイクルは 1.5 以上に設定してください。

具体的には、該当する空間で AnSHW1、AnSHW0、A4HW1、A4HW0 ビットにおいて初期値以外を設定してください。

### 7.11.3 シンクロナス DRAM へのデータ誤出力防止について

SDRAM パーストライトモード、バンクアクティブモードに設定し、イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) による SDRAM を受信バッファとした受信 DMA 転送、ダイレクトメモリアクセスコントローラ (DMAC) によるディスティネーションアドレスを SDRAM とした 16 バイト転送、あるいは、キャッシュコントローラによる SDRAM へのライトバックを実行した場合、SDRAM へのデータ誤出力が発生する場合があります。

#### 【発生条件】

以下の (1) から (4) の条件がすべて成立した場合、SDRAM ライトデータ 16 バイトの内、最初の 4 バイトのデータに、直前に SDRAM ヘライトしたデータが誤出力されます。

- (1) 外部クロック (E ): 内部クロック (I ) = 1 : 1 以外で使用している。
- (2) SDRAM パーストライトモードを使用している。
- (3) SDRAM バンクアクティブモードを使用している。
- (4) E-DMAC による SDRAM を受信バッファとした受信 DMA 転送、DMAC による 16 バイト転送 (ソースアドレス = 内蔵メモリあるいは内蔵周辺モジュール空間 / ディスティネーションアドレス = SDRAM)、あるいはキャッシュコントローラによる SDRAM へのライトバックを実行している。

#### 【回避方法】

SDRAM パーストライトモード設定時は、以下の (1) または (2) のいずれかを採用して使用してください。

- (1) 外部クロック (E ): 内部クロック (I ) = 1 : 1 で使用してください。
- (2) SDRAM オートプリチャージモードに設定してください。

## 7. バスステートコントローラ (BSC)

---

---

## 8. キャッシュ

---

### 8.1 概要

#### 8.1.1 特長

本 LSI は 4K バイトの命令・データ混在型 4 ウェイキャッシュメモリを内蔵しています。キャッシュコントロールレジスタ (CCR) の値を特定することで、2K バイトの RAM と 2K バイトの命令・データ混在型キャッシュメモリとして使用することもできます (2 ウェイキャッシュモード)。また、CCR を用いて、命令またはデータのそれぞれに対してキャッシュを使用しないように設定することができます。キャッシュの動作方式は、ライトスルー方式とライトバック方式の両方をサポートしています。

キャッシュメモリの 1 ラインは 16 バイトです。キャッシュメモリの置き換えは常にライン単位に行います。キャッシュの 1 ラインの置き換えには 4 回の 32 ビットアクセスが必要です。エントリ数は 64 であり、アドレス中の 6 ビット (A9 ~ A4) でエントリが定まります。4 ウェイセットアソシアティブ構成をとっており、エントリアドレスが一致する場合にも、最大 4 個の異なる命令・データをキャッシュに格納できます。同一エントリアドレスを持つ 4 つのウェイを効率的に使用するため、擬似 LRU (Least Recently Used) アルゴリズムに基づいた置き換えを提供しています。

## 8. キャッシュ

図 8.1 にキャッシュ構成を、図 8.2 にアドレス構成を示します。

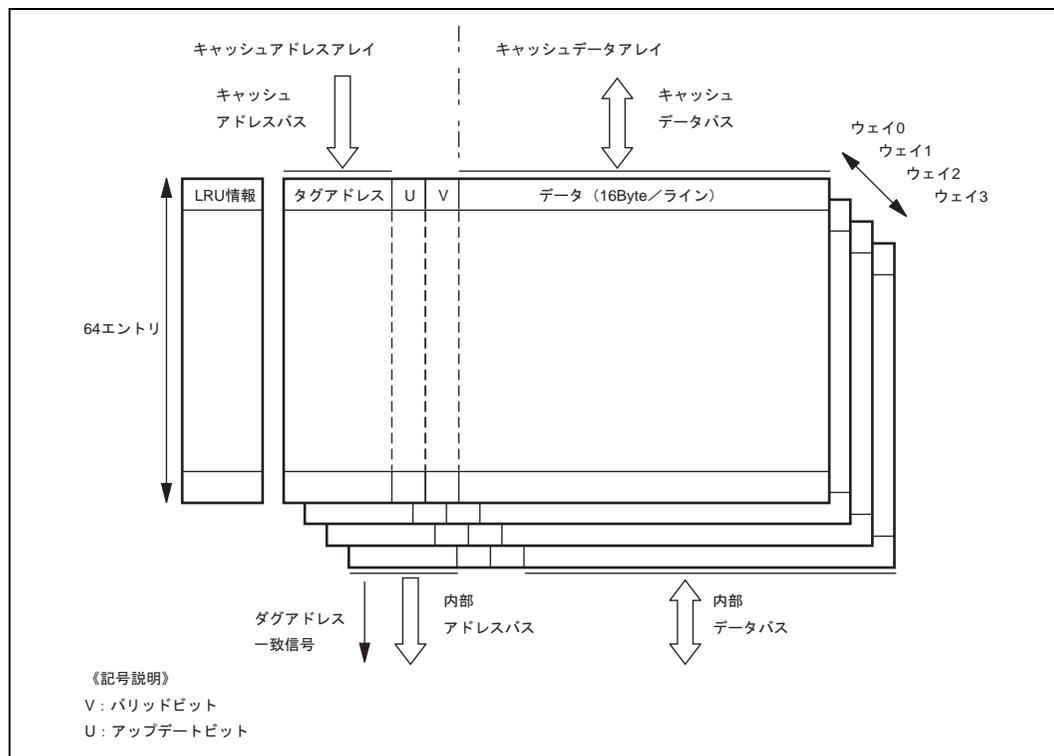


図 8.1 キャッシュ構成

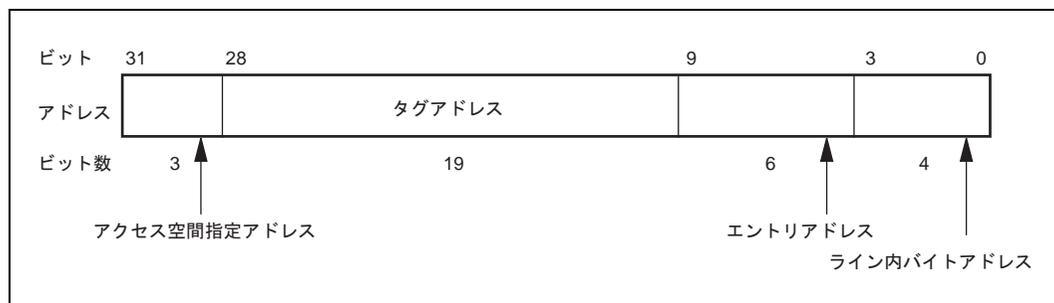


図 8.2 アドレス構成

### 8.1.2 レジスタ構成

キャッシュのレジスタ構成を表 8.1 に示します。

表 8.1 レジスタ構成

名称	略称	R/W	初期値	アドレス
キャッシュコントロールレジスタ	CCR	R/W	H'00	H'FFFFFFE92

## 8.2 レジスタの説明

### 8.2.1 キャッシュコントロールレジスタ (CCR)

キャッシュ制御のためのレジスタに、キャッシュコントロールレジスタ (CCR)があります。キャッシュを制御するためには、CCR の設定とキャッシュの初期化が必要です。

CCR は、パワーオンリセットおよびマニュアルリセットで H'00 に初期化されます。

ビット:	7	6	5	4	3	2	1	0
	W1	W0	WB	CP	TW	OD	ID	CE
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							

- ・ ビット7、6: ウェイ指定ビット1、0 (W1、W0)

W1、W0 はアドレス指定によってアドレスアレイを直接アクセスするときのウェイを指定します。

ビット7	ビット6	説明	
W1	W0		
0	0	ウェイ0	(初期値)
0	1	ウェイ1	
1	0	ウェイ2	
1	1	ウェイ3	

- ・ ビット5: ライトバックビット (WB)

キャッシュ領域へアクセスしたときのキャッシュの動作方式を指定します。

ビット5	説明	
WB		
1	ライトスルー方式	(初期値)
0	ライトバック方式	

- ・ ビット4: キャッシュパーズビット (CP)

CP に1を書き込むことによりキャッシュのすべてのエントリ、すべてのウェイのバリッドビット、アップデートビット、およびLRU情報が初期化されます。初期化完了後、CPビットは0に復帰します。CPビットは読み出し時には常に0です。

初期化終了はキャッシュをリードしてご確認ください。

## 8. キャッシュ

ビット4	説 明
CP	
0	通常動作 (初期値)
1	キャッシュバージ

【注】 読み出し時は常に0です。

### ・ ビット3：2ウェイモード (TW)

TWが0のときは4ウェイセットアソシアティブキャッシュ、TWが1のときには2ウェイセットアソシアティブキャッシュ+2KバイトRAMとして動作します。2ウェイモード時にはウェイ2、ウェイ3がキャッシュ、ウェイ0、ウェイ1がRAMになります。ウェイ0、ウェイ1はアドレス指定によるデータアレイアクセスにより、リード/ライトを行います。

ビット3	説 明
TW	
0	4ウェイモード (初期値)
1	2ウェイモード

### ・ ビット2：データリプレース禁止ビット (OD)

このビットが1の場合、キャッシュミスしても外部メモリから読み出したデータをキャッシュに書き込みません。ただし、キャッシュヒット時のキャッシュ読み出し、更新は行います。CEが1の場合にのみ有効です。

ビット2	説 明
OD	
0	通常動作 (初期値)
1	データアクセス時、キャッシュミスしても置き換えを行わない。

### ・ ビット1：命令リプレース禁止ビット (ID)

このビットが1の場合、キャッシュミスしても外部メモリからフェッチした命令をキャッシュに書き込みません。ただし、キャッシュヒット時のキャッシュ読み出し、更新は行います。CEが1の場合にのみ有効です。

ビット1	説 明
ID	
0	通常動作 (初期値)
1	命令フェッチ時、キャッシュミスしても置き換えを行わない。

### ・ ビット0：キャッシュイネーブルビット (CE)

CEに1を設定することで、キャッシュが使用可能になります。

ビット0	説 明
CE	
0	キャッシュディスエーブル (初期値)
1	キャッシュイネーブル

## 8.3 アドレス空間とキャッシュ

本 LSI ではアドレス空間を 6 個の部分空間に分割しており、キャッシュアクセス方法をアドレスで指定します。表 8.2 は各部分空間とキャッシュ動作の関係を示しています。アドレス空間の詳細は「7. バスステートコントローラ (BSC)」を参照してください。特に、キャッシュ領域とキャッシュスルー領域が実際には同一空間であることに注意してください。

表 8.2 アドレス空間とキャッシュ動作

アドレス A31 ~ A29	部分空間	キャッシュ動作
000	キャッシュ領域	CCR の CE ビットが 1 ならばキャッシュ使用
001	キャッシュスルー領域	キャッシュは使用しない
010	連想バージ領域	指定アドレスのキャッシュのラインをバージ (無効化) する
011	アドレスアレイリード/ライト領域	キャッシュのアドレスアレイに直接アクセス
110	データアレイリード/ライト領域	キャッシュのデータアレイに直接アクセス
111	I/O 領域	キャッシュは使用しない

## 8.4 キャッシュ動作

### 8.4.1 キャッシュ読み出し

キャッシュイネーブル時に CPU から読み出しを行う場合の動作を説明します。

キャッシュアドレスバス上の CPU から出力されたアドレスのエントリアドレス部によって 64 エントリのうちの 1 エントリが選択されます。ウェイ 0 からウェイ 3 の各ウェイのタグアドレスが CPU から出力されたアドレスのタグアドレス部と比較されます。タグアドレスが一致したウェイがある場合をキャッシュヒットといいます。正しい使用状態では各ウェイのタグアドレスは互いに異なっており 1 つのウェイのタグアドレスのみが一致します。すべてのウェイのタグアドレスが一致しなかった場合にはキャッシュミスとなります。パリッドビットが 0 となっているエントリのタグアドレスはいかなる場合も一致しません。

キャッシュヒットの場合、一致したウェイのデータアレイ側からエントリアドレス、ライン内バイトアドレスとアクセスデータサイズに従ってデータが読み出され CPU に送られます。キャッシュアドレスバスに出力されるアドレスは CPU の命令実行ステージで計算され、読み出した結果は CPU のライトバックステージで書き込まれます。キャッシュアドレスバスとキャッシュデータバスも CPU のパイプライン構成にあわせパイプライン動作をしています。アドレスの比較からデータの読み出しまでは 1 サイクルで行われ、アドレスとデータがパイプライン動作するため連続する読み出しをウェイトなしに 1 サイクルごとに処理することができます (図 8.3)。

## 8. キャッシュ

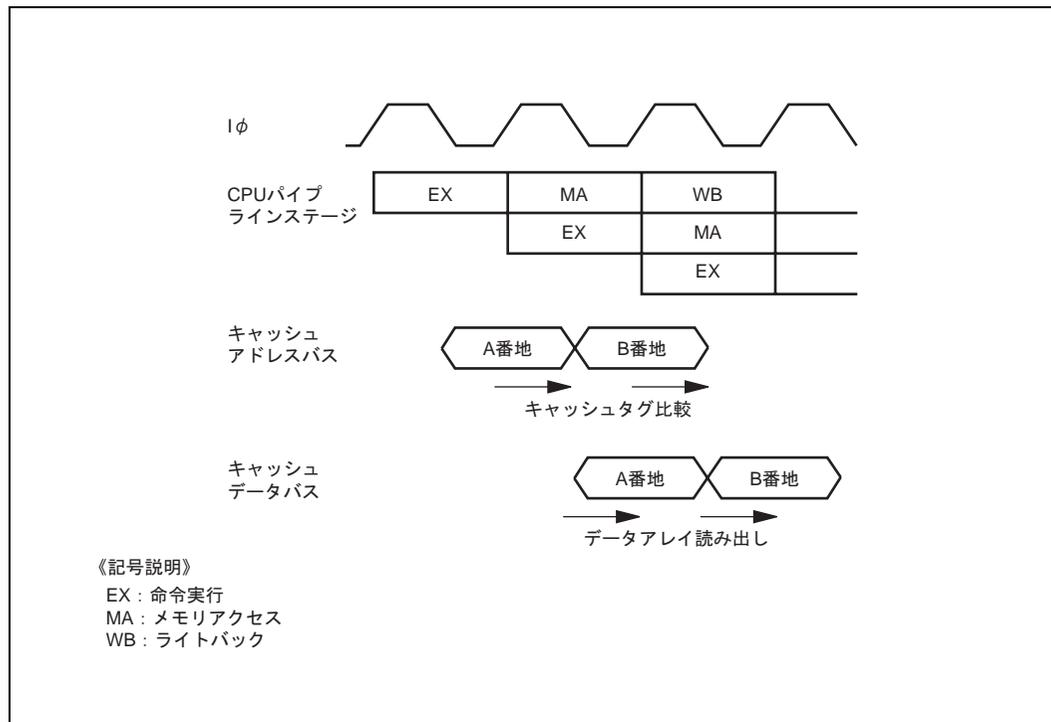


図 8.3 キャッシュヒット時の読み出し

キャッシュミス時には、LRU 情報を用いて置き換えを行うウェイを決定し、そのウェイのアドレスアレイに CPU からの読み出しアドレスを書き込むと同時に、バリッドビットを 1 とします。同時にデータアレイを置き換えるための 16 バイトのデータを読み込むため、キャッシュアドレスバス上のアドレスを内部アドレスバスに出力し、4 ロングワードの連続読み出しを行います。アクセスの順はキャッシュから読み出そうとしているアドレスを含んでいるロングワードが最後になるように、内部アドレスに出力されるアドレスはライン内バイトアドレスを順次 +4 します。内部データバス上の読み出しデータを順次キャッシュデータアレイに書き込み最後のデータはキャッシュデータアレイに書き込むと 1 サイクル遅れてキャッシュデータバスにも出力し、CPU に対して読み出しデータを送ります。

内部アドレスバスと内部データバスもキャッシュバス同様にパイプライン動作を行っています (図 8.4)。

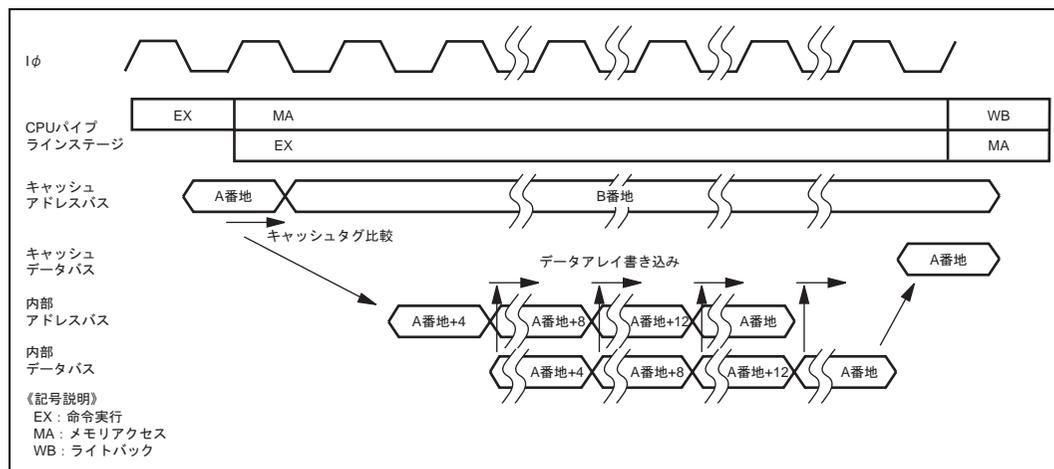


図 8.4 キャッシュミス時の読み出し

## 8.4.2 書き込み

### (1) ライトスルー方式

外部メモリに対する書き込みは、キャッシュヒットの有無にかかわらず行われます。キャッシュアドレスバスに出力された書き込みアドレスを用いて、キャッシュのアドレスアレイのタグアドレスとの比較を行います。一致した場合はこれに続くサイクルにキャッシュデータバスに出力された書き込みデータをデータアレイに書き込みます。一致しない場合、キャッシュデータアレイに対する書き込みは行われません。キャッシュアドレスバスから1サイクル遅れて書き込みアドレスが内部アドレスバスに出力されます。同様にキャッシュデータバスから1サイクル遅れて書き込みデータが内部データバスに出力されます。内部バス上の書き込みが終了するまでCPUは待たされます(図 8.5)。

## 8. キャッシュ

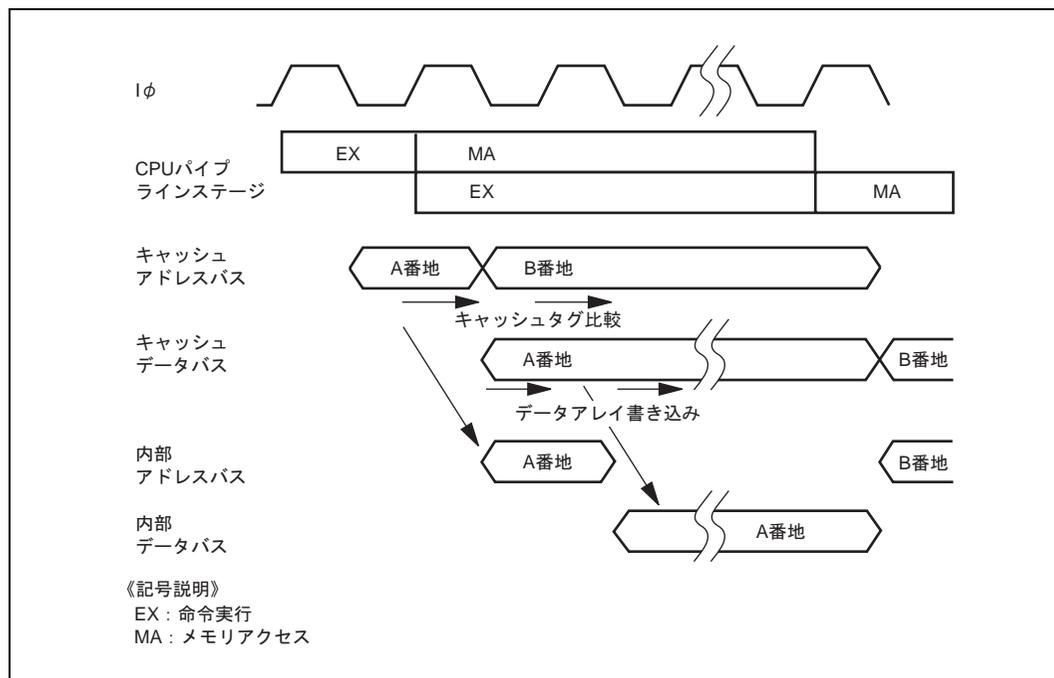


図 8.5 書き込み (ライトスルー方式)

### (2) ライトバック方式

キャッシュヒットの場合、一致したウェイのデータアレイ側へエントリアドレス、ライン内バイトアドレス、およびアクセスデータサイズに従ってデータが書き込まれるとともに、そのエントリのアップデートビットが1にセットされます。データアレイにのみ書き込みが行われ、外部メモリへの書き込みは行われません。ライトヒットは2サイクルで完了します (図 8.6)。

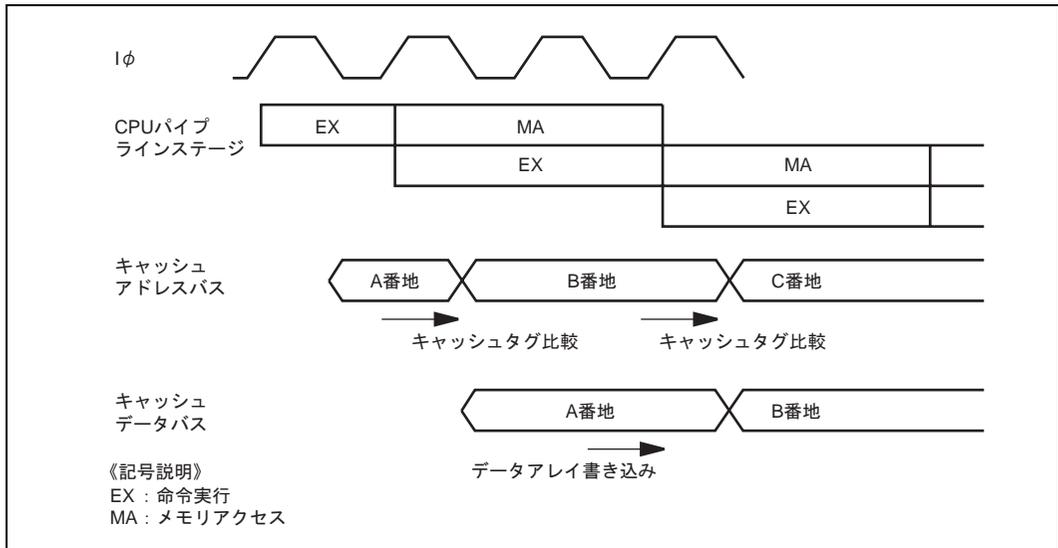


図 8.6 キャッシュヒット時の書き込み (ライトバック方式)

キャッシュミス時には、LRU 情報を用いて置き換えを行うウェイを決定し、そのウェイのアドレスアレイに CPU からの書き込みアドレスを書き込むと同時に、パリティビットとアップデートビットを 1 とします。次にキャッシュバス上のデータをキャッシュに書き込むと同時にデータアレイを置き換えるための 16 バイトのデータを読み込むため、キャッシュアドレスバス上のアドレスを内部アドレスバスに出力し、4 ロングワードの連続読み出しを行います。アクセスの順は、キャッシュへ書き込もうとしているアドレスを含んでいるロングワードが最後になるように、内部アドレスに出力されるアドレスはライン内バイトアドレスを順次 + 4 します。内部データバス上の読み出しデータを順次キャッシュデータアレイに書き込みます。

内部アドレスバスと内部データバスもキャッシュバス同様にパイプライン動作を行っています (図 8.7)。

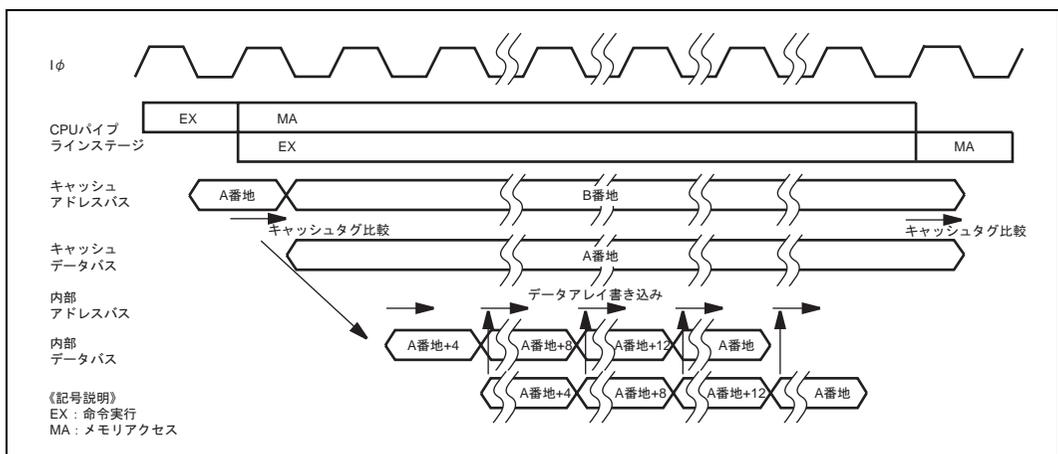


図 8.7 キャッシュミス時の書き込み (ライトバック方式)

## 8. キャッシュ

ライトバック方式で置換されるエントリのアップデートビットが1のとき、外部メモリへの書き戻しが必要になります。性能向上のため、置換されるエントリをまずライトバックバッファに転送し、キャッシュへの新エントリの取り込みを書き戻しより優先させます。キャッシュへの新エントリの取り込み終了後、ライトバックバッファが外部メモリへの書き戻しを行います。この書き戻し中は、キャッシュはアクセス可能です。

ライトバックバッファはキャッシュの1ライン分のデータ（16バイト）とそのアドレスを保持可能です。ライトバックバッファの構成を図8.8に示します。

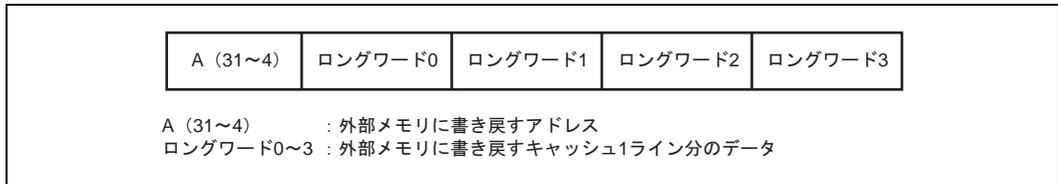


図 8.8 ライトバックバッファの構成

### 8.4.3 キャッシュスルーアクセス

キャッシュスルー領域に対する読み出しまたは書き込み動作を行った場合、キャッシュに対するアクセスは行わず、キャッシュアドレスバスの値を内部アドレスバスに出力します。読み出し動作の場合は、図8.9に示すように内部データバスに出力された読み出しデータを取り込みキャッシュデータバスに出力します。キャッシュスルー領域に対する読み出しは、対象となるアドレスに対してのみ行います。書き込み動作の場合は、キャッシュデータバス上の書き込みデータを内部データバスに出力します。キャッシュスルー領域に対する書き込みはアドレスタグの比較、データアレイへの書き込みを行わない以外は、図8.5の書き込みと同じ動作を行います。

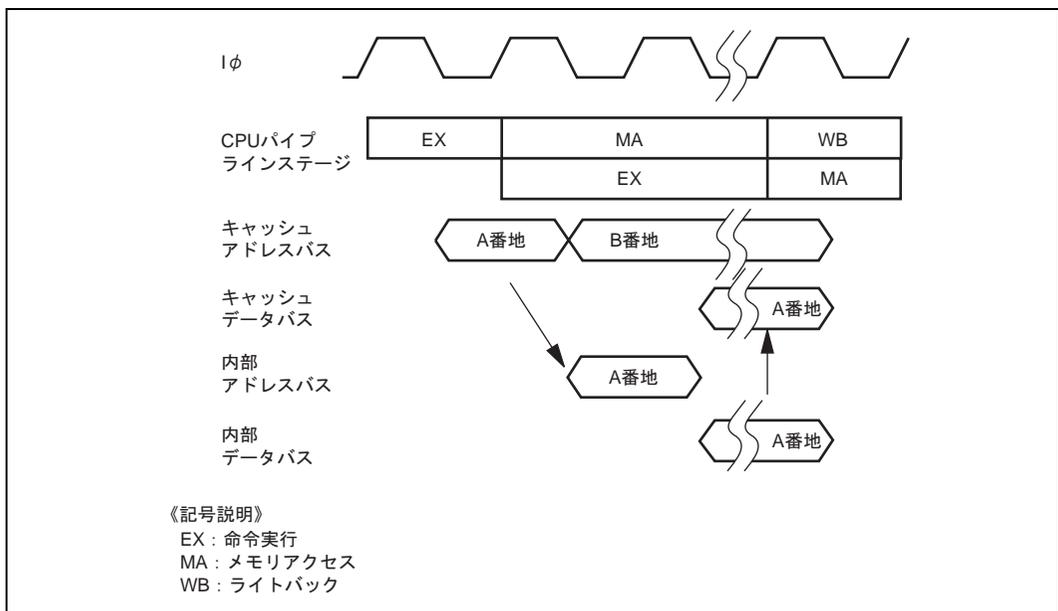


図 8.9 キャッシュスルー領域の読み出し

#### 8.4.4 TAS 命令

TAS 命令はメモリからデータを読み出し、0 と比較した結果をステータスレジスタ (SR) の T ビットに反映させるとともに最上位ビットを 1 にし、同じアドレスに書き込む命令です。

キャッシュ領域に対するアクセスは、通常のデータアクセスと同様に扱われます。

#### 8.4.5 擬似 LRU とキャッシュ置き換え

リード時にキャッシュミスが生じた場合、ミスしたアドレスのデータを 1 ライン (16 バイト) メモリから読み出して置き換えを行います。そのため、4 つのウェイのうちどのウェイを置き換えるかを定める必要があります。一般に最近のアクセス頻度が低いものが次にアクセスされる確率も低いことが予想されます。このウェイを置き換えるアルゴリズムを LRU (Least Recently Used) リプレースアルゴリズムといいますが、実現のためのハードウェアが複雑となります。本キャッシュではウェイのアクセスの順序を記憶しておき、アクセスが古いものから置き換えるという擬似 LRU リプレースアルゴリズムを用いています。アクセスの順序を記憶するため LRU 情報として 6 ビットのデータを用います。それぞれのビットは図 8.10 に示すように、2 つのウェイのアクセスの順序を表し、値が 1 のときに図の矢印の順にアクセスが行われたことを示します。各ビットの値を読み出すことで矢印の向きが確認でき、すべての矢印がそのウェイを起点としているもののアクセスが最も古く、置き換えの対象となります。LRU 情報ビットにアクセスの順を記憶するためリードでキャッシュヒット時、ライトでキャッシュヒット時およびキャッシュミスの後の置き換え時に LRU 情報の書き換えを行います。表 8.3 に書き換える値を、表 8.4 には置き換えるウェイの選択方法を示します。

CCR の CP ビットによるキャッシュパージ後の LRU 情報はすべて 0 とされるため、最初はウェイ 3 ウェイ 2 ウェイ 1 ウェイ 0 の順に使用され、以後はプログラムによるアクセスの順に従ってウェイの選択が行われません。LRU 情報が不当な値を取ると正しい置き換えが行われなため、アドレスアレイ書き込み機能を用いて書き換えを行う場合には LRU 情報として 0 以外の値を書き込まないようにしてください。

CCR の OD または ID ビットが 1 の場合、それぞれ、データ読み出しまたは命令フェッチの際にキャッシュミスが発生してもキャッシュの置き換えを行いません。置き換えを行う代わりにミスしたアドレスのデータを読み込みこれを直接 CPU に送ります。

CCR の TW ビットの設定によるキャッシュの 2 ウェイモードは書き換えの対象となるウェイをウェイ 2、ウェイ 3 のみとすることによって実現されています。アドレスアレイのタグアドレスに対する比較は 2 ウェイモードでも 4 つのウェイすべてに対して行われるため 2 ウェイモードでの動作に先立ってウェイ 0 とウェイ 1 のバリッドビットを 0 にしておく必要があります。

キャッシュの置き換えに伴うタグアドレスおよびバリッドビットの書き込みはメモリからの読み出しの完了を待たずに行われます。置き換えの最中にリセットなどによってメモリアクセスが中断された場合にはキャッシュの内容とメモリの内容に食い違いが生じるので必ずパージを行ってください。

## 8. キャッシュ

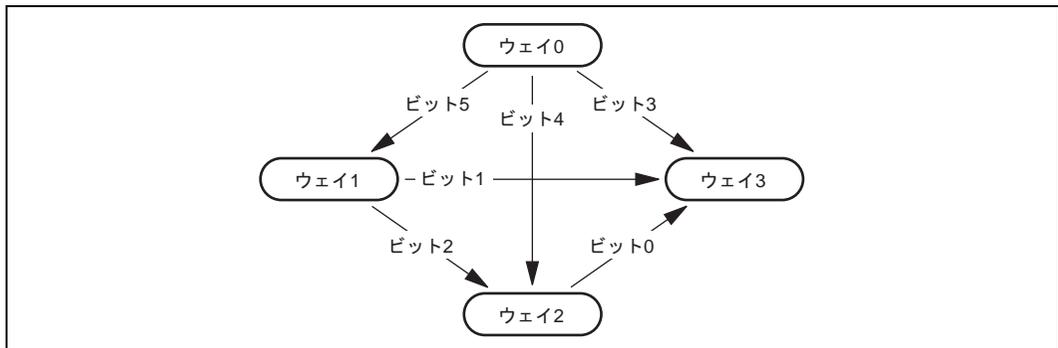


図 8.10 LRU 情報とアクセス順序

表 8.3 LRU 情報の更新

	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
ウェイ0	0	0	0	-	-	-
ウェイ1	1	-	-	0	0	-
ウェイ2	-	1	-	1	-	0
ウェイ3	-	-	1	-	1	1

【注】 - : 更新前の値を保持

表 8.4 置き換えウェイの選択条件

	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
ウェイ0	1	1	1	-	-	-
ウェイ1	0	-	-	1	1	-
ウェイ2	-	0	-	0	-	1
ウェイ3	-	-	0	-	0	0

【注】 - : don't care

### 8.4.6 キャッシュの初期化

キャッシュの全領域のページは CCR の CP ビットに 1 を書き込むことによるのみ行われます。CP への 1 の書き込みによってアドレスレイのパリッドビット、アップデートビット、および LRU 情報の全ビットが 0 に初期化されます。キャッシュのページは 1 サイクルで完了しますが、このほかに CCR に対する書き込みに要する時間が必要となります。キャッシュのイネーブルに先立って必ずパリッドビット、アップデートビットおよび LRU の初期化を行ってください。

キャッシュがイネーブル状態のとき、CCR への書き込みの最中も命令の読み出しはキャッシュから行われるため、プリフェッチされる命令はキャッシュから読み出されてしまいます。正しくページを行うためには CCR の CE ビットに 0 を書き、いったんキャッシュをディスエーブルにしてからページを行ってください。パワーオンリセットおよびマニュアルリセットの際には CCR の CE ビットは 0 に初期化されるためリセット後は直ちにキャッシュのページを行うことができます。

### 8.4.7 連想パージ

特定のアドレスの内容がキャッシュに含まれていた場合に該当する 1 ライン (16 バイト) を無効化するための機能です。マルチプロセッサ構成または本チップ内の E-DMAC (または DMAC) と CPU がメモリを共有するような構成で、共有するアドレスの内容を一方の CPU が書き換えた場合、他方の CPU のキャッシュにそのアドレスが含まれていれば、これを無効化する必要があります。パージしたいアドレスに H'40000000 を加えたアドレスに書き込みまたは読み出しを行うと、加える前のアドレスの格納されているエントリのバリッドビットおよびアップデートビットが 0 に初期化されます。1 回の書き込みで 16 バイトパージされるため 256 バイト連続する領域のパージは 16 回ですみます。

連想パージを行う際のアクセスサイズはロングワードとしてください。1 ラインをパージするために 2 サイクル必要です。

また、キャッシュ内にダーティなラインが存在していた場合、連想パージによってメインメモリへの書き戻し (フラッシュ) は行いません。ご注意ください。

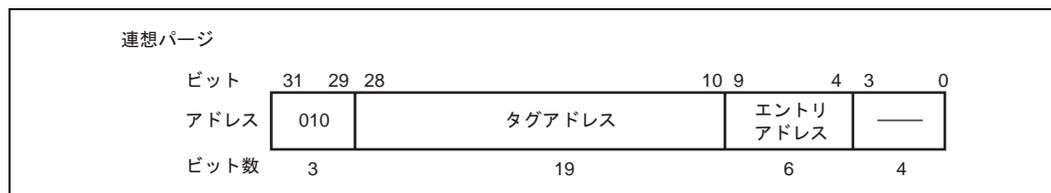


図 8.11 連想パージアクセス

### 8.4.8 キャッシュフラッシュ

マルチプロセッサ構成または本チップ内の E-DMAC (または DMAC) と CPU がメモリを共有するような構成で、CPU がキャッシュ内にある特定のアドレスの内容をライトバック方式で書き換えた場合、バス権を本チップ内の CPU から他マスタ (外部マスタ、E-DMAC、DMAC) に与える前に書き換えたデータをメインメモリに書き戻し、キャッシュの内容を無効化する必要があります。本チップは、特定のアドレスの内容をフラッシュするための命令や手順をサポートしていません。そのため、キャッシュフラッシュを実現するためには、キャッシュから追い出したいアドレス空間以外の空間 4kB (キャッシュ領域) に対してリードを行い、意図的にキャッシュミスが発生させる必要があります。このときキャッシュアクセスは、16 バイトおきにアクセスしてください。これによりライトバックが発生し、本チップ内の CPU によってキャッシュに書き込まれた内容がメインメモリにライトバックされフラッシュを実行することが可能となります。ただし、この方式は、リードミスで発生するキャッシュフィル時間とキャッシュに残しておきたかったデータの再読み込み時間がオーバーヘッドとなってしまいます。そのため、複数のマスタがメモリを共有するようなシステムを構築する上で、ライトバック方式によるオーバーヘッドが気になる場合には、コヒーレンスを保つため共有領域をキャッシュスルー領域としてください。

### 8.4.9 データアレイアクセス

キャッシュのデータアレイはデータアレイリード・ライト領域を介して直接リード/ライトできます。データアレイに対するアクセスサイズは、バイト、ワード、ロングワードのいずれも行うことができます。データアレイ

## 8. キャッシュ

このアクセスはリード1サイクル、ライト2サイクルで完了します。キャッシュバスのみを使用するためDMACほかのマスタがバスを使用している場合でも並行して動作することが可能です。ウェイト0のデータアレイはH'C0000000 ~ H'C00003FF、ウェイト1はH'C0000400 ~ H'C00007FF、ウェイト2はH'C0000800 ~ H'C0000BFF、ウェイト3はH'C0000C00 ~ H'C0000FFFにマッピングされます。2ウェイトモードを使用する場合、2Kバイトの内蔵RAMとしてアクセスするのはH'C0000000 ~ H'C00007FFの領域です。キャッシュをディスエーブルするとH'C0000000 ~ H'C0000FFFの領域が4Kバイトの内蔵RAMとして使えます。

キャッシュとして使用しているウェイトの内容をデータアレイアクセスを用いて書き換えると外部メモリとキャッシュの内容の一致がとれなくなるので行わないでください。

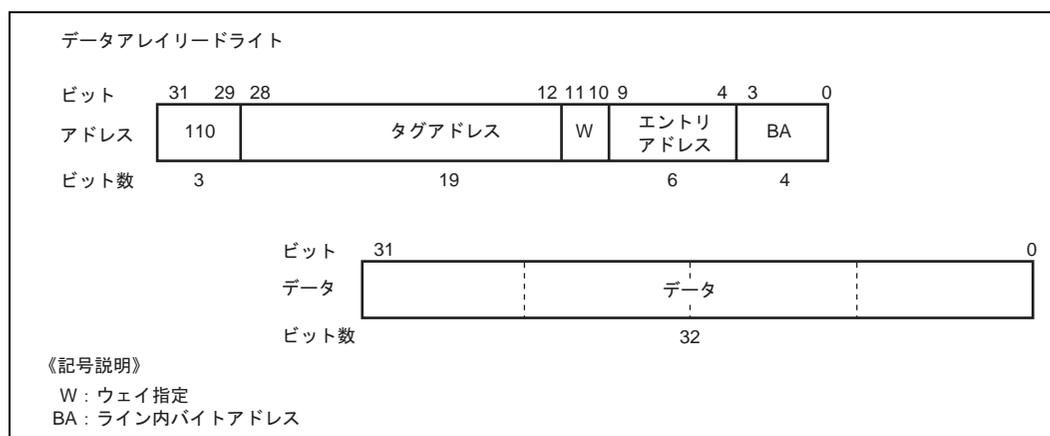


図 8.12 データアレイアクセス

### 8.4.10 アドレスアレイアクセス

プログラムデバッグなどの目的でキャッシュに取り込まれている内容の確認が行えるよう、キャッシュのアドレスアレイにアクセスすることができます。アドレスアレイはH'60000000 ~ H'600003FFにマッピングされます。すべてのウェイトが同じアドレスにマッピングされるため、ウェイトの選択はCCRのW1、W0ビットを書き換えることによって行います。アドレスアレイのアクセスサイズはロングワードのみです。

アドレスアレイを読み出すと、データとしてタグアドレス、LRU情報、アップデートビット、バリッドビットが出力されます。アドレスアレイに書き込みを行う際には、タグアドレス、アップデートビットおよびバリッドビットはキャッシュアドレスバスから書き込まれます。したがって書き込みアドレスを計算して書き込む必要があります。LRU情報はキャッシュデータバスから書き込まれますが、誤動作を防止するため必ず0を書いてください。

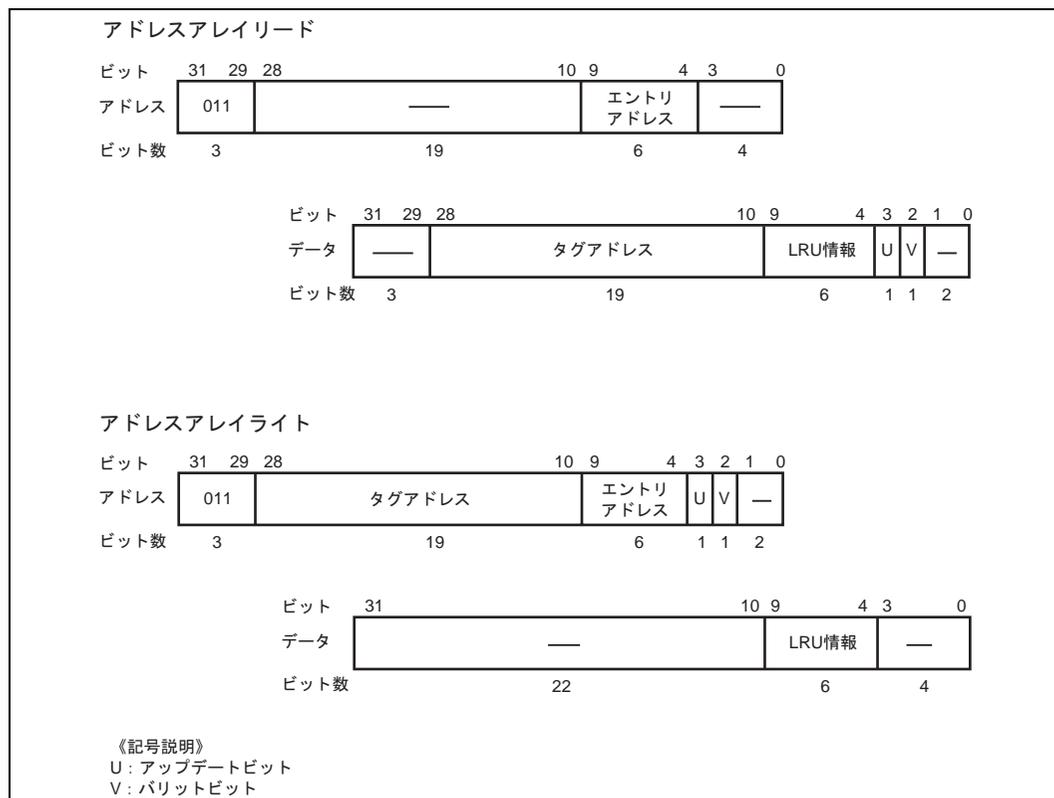


図 8.13 アドレスアレイアクセス

## 8.5 使用方法

### 8.5.1 初期化

キャッシュメモリはリセット時には初期化されません。そのため、キャッシュを使用するためにはソフトウェアによる初期化が必要です。キャッシュの初期化はアドレスアレイのバリッドビットおよびLRU情報をすべて0にします。初期化の方法としてはアドレスアレイライト機能を用いて1ラインずつ初期化することもできますが、CCRのCPビットに1を書き込むことによって1度に初期化する方が簡単です。

図 8.14 にキャッシュの初期化手順を示します。

## 8. キャッシュ

```
MOV.W  #H'FE92,R1
MOV.B  @R1,R0    ;
AND    #H'FE,R0  ;
MOV.B  R0,@R1    ; キャッシュディスエーブル
OR     #H'10,R0  ;
MOV.B  R0,@R1    ; キャッシュパーズ
OR     #H'01,R0  ;
MOV.B  R0,@R1    ; キャッシュイネーブル
```

図 8.14 キャッシュの初期化

### 8.5.2 特定アドレスのパーズ

本 LSI はスヌープ機能（データ書き換えのモニタ機能）を持たないため、キャッシュメモリと外部メモリの内容に違いが生じる操作を行った場合には、キャッシュの特定ラインをパーズする必要が生じます。たとえば、キャッシュ領域に対して DMA 転送を行った場合、書き換えを行ったアドレス領域に対応するキャッシュのラインをパーズする必要があります。CCR の CP ビットを 1 にすることでキャッシュの全エントリをパーズすることもできますが、パーズする必要があるアドレスの範囲が限定されている場合には特定ラインのみパーズする方が効果的です。

特定ラインのパーズには連想パーズを使用します。キャッシュの各ラインは 16 バイトですからパーズも 16 バイト単位に行われます。4 つのウェイが同時にチェックされ指定されたアドレスに対応するデータを保持しているラインのみがパーズされます。アドレスが一致しない場合、指定したアドレスのデータはキャッシュに取り込まれていないのでパーズは行われません。

```
 ;R3のアドレスから32バイトの領域をパーズ
MOV.L  #H'40000000,R0
XOR    R1,R1
MOV.L  R1,@(R0,R3)
ADD    #16,R3
MOV.L  R1,@(R0,R3)
```

図 8.15 特定アドレスのパーズ

DMA 転送のたびにキャッシュのパーズを行うのが煩わしい場合には、あらかじめ CCR の OD ビットを 1 に設定しておくこともできます。その場合、OD ビットが 1 の間はキャッシュは命令専用のキャッシュメモリとして動作します。ただし、すでにキャッシュメモリにデータが取り込まれている場合には、やはり DMA 転送時にキャッシュの特定ラインをパーズする必要があります。

### 8.5.3 キャッシュデータのコヒーレンシ

本 LSI のキャッシュメモリはスヌープ機能を有していません。このため CPU 以外のバスマスタとデータを共有する場合には、データのコヒーレンシ（同一性）をソフトウェアによって保証する必要があります。方法としてはキャッシュスルー領域を用いる方法、ライトスルー方式を採用してプログラムのロジックでキャッシュパーズを行う方法があります。

キャッシュスルー領域を用いる方法は、複数のバスマスタで共有するデータはキャッシュスルー領域に配置するものです。キャッシュスルー領域のアクセスはキャッシュ内にデータを取り込むことがないため、データの coherence は容易に保証できます。共有データに対するアクセスが繰り返行われ、データの書き換えの頻度が低い場合にはアクセス速度の低下が性能に及ぼす影響があります。

プログラムのロジックでキャッシュバージを行う方法は、プログラムの流れによってデータの更新を検出し、キャッシュバージを行う方法です。例えば、ディスクからのデータ入力を行うプログラムの場合、セクタなどの単位の読み出しが完了したときに、読み込みに使用したバッファアドレスまたはすべてのキャッシュをバージすることによって coherence を保つことができます。2つのプロセッサ間でデータのやりとりをする場合には、データの準備の完了や、取り込みの完了を互いに通知するフラグのみをキャッシュスルー領域に設けておき、実際に転送するデータはキャッシュ領域におき、最初のデータの読み出しに先立ってキャッシュのバージを行うことによってデータの coherence を保つことができます。通信の手段としてセマフォを用いる場合には、TAS 命令を利用することによりキャッシュのバージを行わなくてもデータの coherence が保証できます。TAS 命令の読み出しは、必ず直接外部アクセスとしてください。

更新の単位が小さい場合には、特定アドレスのバージの方法を用いて該当するアドレスのバージのみを行います。これに対して更新の単位が大きい場合には、すべてのアドレスを順にバージするよりもキャッシュ全体をバージし、以前キャッシュにあったデータも再度外部メモリから読み込んだほうが早くなります。

ライトバック方式を採用する場合、意図的なキャッシュミスリードによるメモリへの書き戻し（フラッシュ）を実行することで coherence を保つことができますが、フラッシュの実行によるオーバーヘッドが発生するため、複数マスタがメモリ共有するようなシステムでは、ライトスルー方式またはキャッシュスルー領域へのアクセスを推奨いたします。

#### 8.5.4 2ウェイキャッシュモード

CCR の TW ビットを 1 に設定することで、4K バイトのキャッシュを 2K バイトの RAM と 2K バイトの命令・データ混在型キャッシュメモリとして使用することができます。ウェイ 2、ウェイ 3 がキャッシュ、ウェイ 0、ウェイ 1 が RAM になります。

初期化は 4 ウェイと同様に CCR の CP ビットに 1 を書き込むことで行い、パリティビット、アップデートビットおよび LRU ビットを 0 にクリアします。

LRU 情報の初期値を 0 に設定した場合、最初、ウェイ 3、ウェイ 2 の順に使用されます。その後は LRU 情報に従い、ウェイ 3 またはウェイ 2 が置き換えの候補として選択されます。LRU 情報の書き換えが起こる条件は、ウェイ数が 2 であることを除いて 4 ウェイモードと同一です。

2K バイトの RAM となったウェイ 0、ウェイ 1 のアクセスにはデータアレイアクセスを用います。アドレスマッピングは図 8.16 のようになります。

## 8. キャッシュ

---

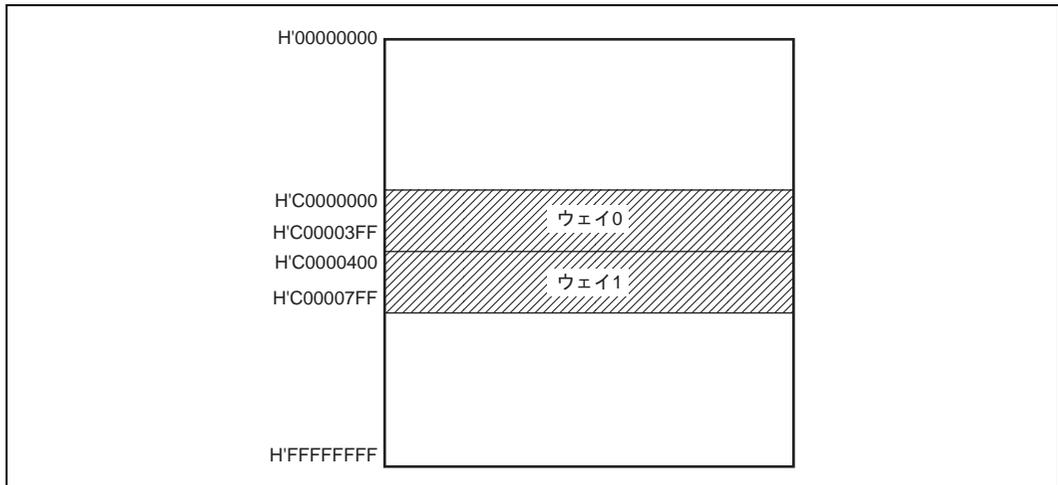


図 8.16 2 ウエイモード時の 2K バイト RAM のアドレスマッピング

## 8.6 使用上の注意事項

### 8.6.1 スタンバイ

低消費電力のためのスタンバイ状態に入る前に、キャッシュはディスエーブルしてください。スタンバイからの復帰後キャッシュを使用する場合、それに先立ってキャッシュの初期化を行ってください。

### 8.6.2 キャッシュコントロールレジスタ

CCR の内容を変更するとキャッシュの動作が変化します。本 LSI はパイプライン動作を多用しているためアクセスの同期がとりにくくなっています。このため、キャッシュコントロールレジスタの内容の変更はキャッシュをディスエーブルにするのと同様か、ディスエーブル後に行ってください。また CCR の内容変更が続いて CCR の読み出しを行ってください。



---

## 9. イーサネットコントローラ (EtherC)

---

### 9.1 概要

本 LSI は、イーサネットあるいは IEEE802.3 の MAC ( Media Access Control ) 層規格に準拠したイーサネットコントローラ ( 以後 EtherC と略します ) を内蔵しています。本イーサネットコントローラは、同規格に合致する物理層 LSI ( 以後 PHY-LSI と略します ) と接続することにより、イーサネット /IEEE802.3 フレームの送受信を行うことができます。イーサネットコントローラは本 LSI 内部で送信・受信各々専用のイーサネットコントローラ用ダイレクトメモリアクセスコントローラ ( 以後 E-DMAC と略します ) に接続されており、メモリとの間のデータ転送を高速に行います。

#### 9.1.1 特長

EtherC には、以下のような特長があります。

- イーサネット /IEEE802.3 フレームの送受信
- 10 / 100Mbps 転送への対応
- 全二重モード・半二重モード対応
- IEEE802.3u 規格の MII ( Media Independent Interface ) 準拠
- Magic Packet の検出および Wake On LAN ( WOL ) 信号の出力

### 9.1.2 構成

図 9.1 に EtherC の構成を示します。

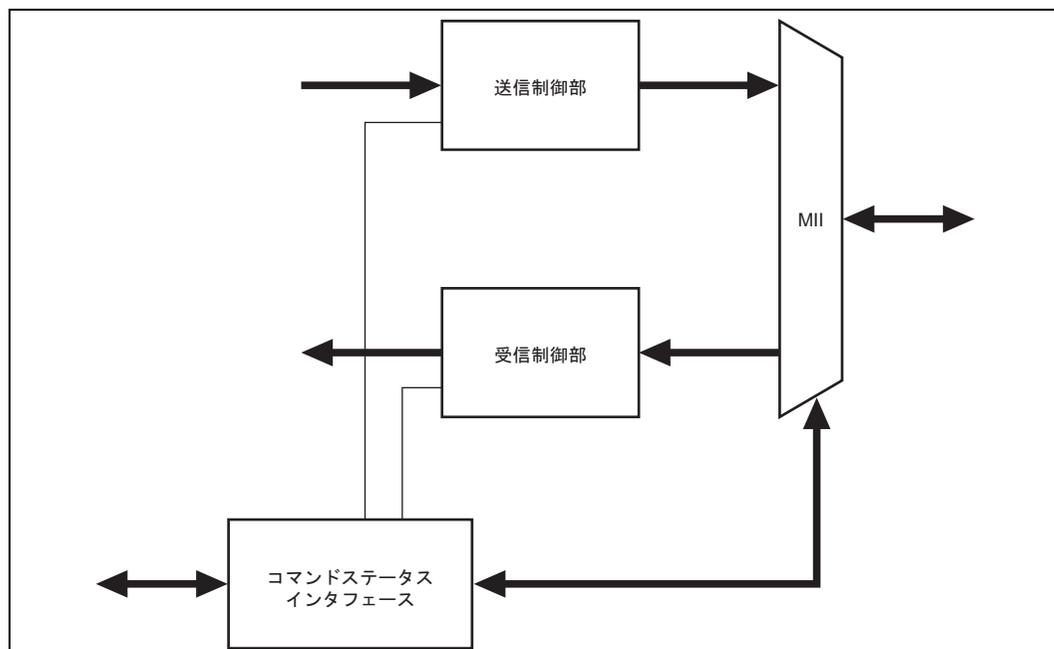


図 9.1 EtherC の構成

#### (1) 送信制御部

送信データは、メモリから送信 E-DMAC を経由して送信 FIFO に蓄えられます。送信制御部では、このデータをイーサネット /IEEE802.3 フレームに組み立て、MII に出力します。MII を経由した送信データは、PHY-LSI によって回線の上に送出されます。主な機能を以下に示します。

- データのフレーム化と送信
- CRC 計算とフレームへの付与
- 衝突発生時の再送 (最大 15 回)
- IEEE802.3u 規格にある MII 準拠
- PHY-LSI の速度に対応したバイト・ニブル変換

#### (2) 受信制御部

MII を経由して受信したフレームは、受信制御部において、アドレス情報、フレーム長、CRC チェックなどを行い、受信データは、受信 E-DMAC によってメモリに転送されます。主な機能を以下に示します。

- 受信したフレームのフォーマットチェック
- 受信フレームの CRC、ならびにフレーム長のチェック
- 自局宛、マルチキャストあるいはブロードキャストの受信フレームをメモリに転送
- IEEE802.3u 規格にある MII 準拠

- PHY-LSI の速度に対応したニブル・バイト変換
- Magic Packet 監視機能

### (3) コマンド・ステータスインタフェース

EtherC を制御する各種コマンド/ステータスレジスタの提供、および MII 経由で PHY-LSI 内部レジスタのアクセスを行います。

#### 9.1.3 端子構成

本 EtherC には IEEE802.3u に規定される 18 本の MII に対応する信号端子および PHY-LSI との接続を容易とする 3 本の関連する信号端子があります。端子構成を表 9.1 に示します。

表 9.1 MII 端子機能

種類	記号	名称	入出力	機能
MII	TX-CLK	送信クロック	入力	TX-EN、ETXD3 ~ 0、TX-ER のタイミング参照信号
	RX-CLK	受信クロック	入力	RX-DV、ERXD3 ~ 0、RX-ER のタイミング参照信号
	TX-EN	送信イネーブル	出力	ETXD3 ~ 0 上に送信データが準備できたことを示す信号
	ETXD3 ~ 0	送信データ (4 ビット)	出力	4 ビットの送信データ
	TX-ER	送信エラー	出力	送信中のエラーを PHY-LSI に通知する
	RX-DV	受信データ有効	入力	有効な受信データが ERXD3 ~ 0 上にあることを示す
	ERXD3 ~ 0	受信データ (4 ビット)	入力	4 ビットの受信データ
	RX-ER	受信エラー	入力	データ受信中に発生したエラー状態を認識する
	CRS	キャリア検出	入力	キャリア検出信号
	COL	衝突検出	入力	衝突検出信号
	MDC	管理用データ クロック	出力	MDIO による情報転送用の参照クロック信号
	MDIO	管理用データ入出力	入出力	STA と PHY との間で管理情報を交換するための双方向信号
その他	LNKSTA	リンクステータス	入力	PHY-LSI からのリンク状態入力
	EXOUT	汎用外部出力	出力	外部出力用端子
	WOL	Wake-On-LAN	出力	Magic Packet 受信

## 9. イーサネットコントローラ (EtherC)

### 9.1.4 イーサネットコントローラレジスタ構成

EtherC には、表 9.2 に示す 19 本の 32 ビットレジスタがあります。

表 9.2 レジスタ構成

名称	略称	R/W	初期値	アドレス
EtherC モードレジスタ	ECMR	R/W	H'00000000	H'FFFFFFD60
EtherC ステータスレジスタ	ECSR	R/W*1	H'00000000	H'FFFFFFD64
EtherC 割り込み許可レジスタ	ECSIPR	R/W	H'00000000	H'FFFFFFD68
PHY 部インタフェースレジスタ	PIR	R/W	H'0000000X	H'FFFFFFD6C
MAC アドレス上位設定レジスタ	MAHR	R/W	H'00000000	H'FFFFFFD70
MAC アドレス下位設定レジスタ	MALR	R/W	H'00000000	H'FFFFFFD74
受信フレーム長上限レジスタ	RFLR	R/W	H'00000000	H'FFFFFFD78
PHY 部ステータスレジスタ	PSR	R	H'00000000	H'FFFFFFD7C
送信リトライオーバーカウンタレジスタ	TROCR	R/W*2	H'00000000	H'FFFFFFD80
衝突検出カウンタレジスタ	CDCR	R/W*2	H'00000000	H'FFFFFFD84
キャリア消失カウンタレジスタ	LCCR	R/W*2	H'00000000	H'FFFFFFD88
キャリア未検出カウンタレジスタ	CNDCR	R/W*2	H'00000000	H'FFFFFFD8C
フレーム長異常カウンタレジスタ	IFLCR	R/W*2	H'00000000	H'FFFFFFD90
CRC エラーフレーム受信カウンタレジスタ	CEFCR	R/W*2	H'00000000	H'FFFFFFD94
フレーム受信エラーカウンタレジスタ	FRECR	R/W*2	H'00000000	H'FFFFFFD98
64 バイト未満フレーム受信カウンタレジスタ	TSFRCR	R/W*2	H'00000000	H'FFFFFFD9C
指定バイト超フレーム受信カウンタレジスタ	TLFRCR	R/W*2	H'00000000	H'FFFFFFDA0
端数ビットフレーム受信カウンタレジスタ	RFCR	R/W*2	H'00000000	H'FFFFFFDA4
マルチキャストアドレスフレーム受信カウンタレジスタ	MAFCR	R/W*2	H'00000000	H'FFFFFFDA8

- 【注】
- 各レジスタは、32 ビット単位でアクセスしてください。
  - レジスタ内にある予約ビットに書き込む場合、必ず 0 としてください。
  - 予約ビットリード時の値は、これを保証するものではありません。  
\*1 各ビットに 1 を書き込むことでクリアされます。  
\*2 該当レジスタへの書き込みによってクリアされます。

## 9.2 レジスタの説明

### 9.2.1 EtherC モードレジスタ (ECMR)

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	PRCEF	-	-	MPDE	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R
ビット:	7	6	5	4	3	2	1	0
	-	RE	TE	-	ILB	ELB	DM	PRM
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R/W	R/W

EtherC モードレジスタは、イーサネットコントローラの動作モードを指定するレジスタです。通常、本レジスタの設定は、リセット後の初期設定時に行います。

【注】 動作モードの設定は、送信・受信機能が有効状態で書き換えることを禁止します。動作モードを切り替える場合には、E-DMAC モードレジスタ (EDMR) 内にあるソフトウェアリセットビット (SWR) により、EtherC および E-DMAC を初期状態に戻してから再設定を行います。

- ・ ビット 31 ~ 13 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 12 : CRC エラーフレーム受信許可 (PRCEF)  
CRC エラーとなった受信フレームの報告指示を行います。

ビット 12	説明
PRCEF	
0	CRC エラーとなった受信フレームを、エラーとする (初期値)
1	CRC エラーとなった受信フレームを、エラーとしない

【参考】 本ビットに 1 を設定することにより、CRC エラーを検出しない場合、CRC エラーフレーム受信カウンタレジスタ (CEFCR、9.2.14 参照) のカウントアップを行いません。

## 9. イーサネットコントローラ (EtherC)

---

- ビット 11 ~ 10 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 9 : Magic Packet 検出許可 (MPDE)

イーサネットからの起動を有効にするため、ハードウェアによる Magic Packet の検出機能を許可するか否かを指定します。Magic Packet を検出すると EtherC ステータスレジスタに反映され、WOL 端子により Magic Packet を受信したことを通知します。

ビット 9	説 明
MPDE	
0	Magic Packet の検出を許可しない (初期値)
1	Magic Packet の検出を許可する

- ビット 8 ~ 7 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 6 : 受信許可 (RE)

受信機能の動作を有効にするか無効にするかを指定します。

ビット 6	説 明
RE	
0	受信機能を無効にする (初期値)
1	受信機能を有効にする

【参考】 フレームを受信中に、受信機能を有効状態 (RE = 1) から無効状態 (RE = 0) に切り替えた場合、当該フレームの受信終了後に受信機能が無効となります。

- ビット 5 : 送信許可 (TE)

送信機能の動作を有効にするか無効にするかを指定します。

ビット 5	説 明
TE	
0	送信機能を無効にする (初期値)
1	送信機能を有効にする

【参考】 フレームを送信中に、送信機能を有効状態 (TE = 1) から無効状態 (TE = 0) に切り替えた場合、当該フレームの送信終了後に送信機能が無効となります。

- ビット 4 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 3 : 内部ループバックモード (ILB)

EtherC 内部でのループバックモードを指定します。

ビット3	説 明
ILB	
0	通常のデータ送受信を行う (初期値)
1	EtherC 内部でのデータの折り返しを行う

【注】 ループバックモードの指定は、全二重転送方式 (DM = 1、本レジスタのビット1)で行ってください。

・ ビット2：外部ループバックモード (ELB)

本ビットの値は、本 LSI の汎用外部出力端子 (EXOUT) にそのまま出力されます。EXOUT 端子を用いて PHY-LSI におけるループバックモードの指示などに利用します。

ビット2	説 明
ELB	
0	EXOUT 端子は Low を出力する (初期値)
1	EXOUT 端子は High を出力する

【参考】 本機能によって PHY-LSI でのループバック機能を実現する場合、PHY-LSI に EXOUT 端子に対応する端子があることが必要です。

・ ビット1：デュプレックスモード (DM)

EtherC の転送方式を指示します。

ビット1	説 明
DM	
0	半二重転送方式を指定する (初期値)
1	全二重転送方式を指定する

【注】 内部ループバックモードを指定 (ILB = 1) する場合は、全二重転送方式 (DM = 1)で行ってください。

DM ビットには、物理層 LSI が検出した全二重 / 半二重のモードの情報を反映させる必要があります。物理層 LSI の通信モードと本 LSI の設定が合致していない場合、転送効率の低下、データの衝突が起こる要因となります。

・ ビット0：プロミスキャスモード (PRM)

本ビットを設定すると、すべてのイーサネットフレームを受信することができます。

ビット0	説 明
PRM	
0	EtherC は通常動作を行う (初期値)
1	EtherC はプロミスキャスモード動作を行う

【参考】 すべてのイーサネットフレームとは、宛先アドレス、ブロードキャストアドレス、マルチキャストビットなどの相違や有無にかかわらず、受信可能なすべてのフレームを示します。

## 9. イーサネットコントローラ (EtherC)

### 9.2.2 EtherC ステータスレジスタ (ECSR)

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	LCHNG	MPD	ICD
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W*	R/W*	R/W*

【注】 \* 1 を書き込むと対応するフラグはクリアされます。0 を書き込んだ場合は、フラグに影響を与えません。

EtherC ステータスレジスタは、EtherC 内のステータス表示用レジスタです。本ステータスは、割り込みによって CPU に通知することが可能です。各ビットは 1 を書くことによってクリアされます。また割り込みを発生するビットは、EtherC 割り込み許可レジスタ (ECSIPR) の対応するビットにより割り込みを許可あるいは禁止することができます。

・ ビット 31 ~ 3 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 2 : リンク信号変化 (LCHNG)

PHY-LSI から入力される LNKSTA 信号が、High Low あるいは Low High に変化したことを示します。本ビットをクリアする場合は、1 を書き込みます。なお 0 を書き込んだ場合は、影響を与えません。

ビット 2 LCHNG	説明
0	LNKSTA 信号の変化を検出していない (初期値)
1	LNKSTA 信号の変化 (High Low あるいは Low High) を検出した

【参考】 現在の Link 状態を確認するには、PHY 部ステータスレジスタ (PSR) の LMON ビットを参照してください。

【注】 ビンファンクションコントローラ (PFC) のポート A コントロールレジスタ (PACR) で LNKSTA の機能を選択されたタイミングで、信号の変化を検出する場合があります。

・ ビット 1 : Magic Packet 検出 (MPD)

回線上から Magic Packet を検出したことを示します。本ビットをクリアする場合は、1 を書き込みます。なお 0 を書き込んだ場合は、影響を与えません。

ビット 1	説 明
MPD	
0	Magic Packet を検出していない (初期値)
1	Magic Packet を検出した

・ ビット 0 : 不正キャリア検出 (ICD)

回線上で PHY-LSI が不正なキャリアを検出したことを示します。本ビットをクリアする場合は、1 を書き込みます。なお 0 を書き込んだ場合は、影響を与えません。

ビット 0	説 明
ICD	
0	PHY-LSI は回線上に不正キャリアを検出していない (初期値)
1	PHY-LSI は回線上で不正キャリアを検出した

【注】 PHY-LSI から入力される信号の変化が、ソフトウェアの認識時間よりも早く変化するような場合、正しい情報が得られない場合があります。採用する PHY-LSI のタイミングを参照してください。

## 9. イーサネットコントローラ (EtherC)

### 9.2.3 EtherC 割り込み許可レジスタ (ECSIPR)

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	LCHNGIP	MPDIP	ICDIP
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

EtherC ステータスレジスタによって報告される割り込み要因の許可を指示します。各ビットは、EtherC ステータスレジスタ内のビットに対応する割り込みを許可することができます。

- ・ ビット 31 ~ 3 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 2 : リンク信号変化割り込み許可ビット (LCHNGIP)  
リンク信号変化による割り込み通知を制御します。

ビット 2	説明
LCHNGIP	
0	ECSR 内 LCHNG ビットの割り込み通知を禁止する (初期値)
1	ECSR 内 LCHNG ビットの割り込み通知を許可する

- ・ ビット 1 : Magic Packet 検出割り込み許可ビット (MPDIP)  
Magic Packet 検出による割り込み通知を制御します。

ビット 1	説明
MPDIP	
0	ECSR 内 MPD ビットの割り込み通知を禁止する (初期値)
1	ECSR 内 MPD ビットの割り込み通知を許可する

- ビット0：不正キャリア検出割り込み許可ビット (ICDIP)  
不正キャリア検出ビットによる割り込み通知を制御します。

ビット0	説明
ICDIP	
0	ECSR 内 ICD ビットの割り込み通知を禁止する (初期値)
1	ECSR 内 ICD ビットの割り込み通知を許可する

### 9.2.4 PHY 部インタフェースレジスタ (PIR)

ビット：	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値：	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R
ビット：	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値：	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R
ビット：	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値：	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R
ビット：	7	6	5	4	3	2	1	0
	-	-	-	-	MDI	MDO	MMD	MDC
初期値：	0	0	0	0	*	0	0	0
R/W：	R	R	R	R	R	R/W	R/W	R/W

\*：不定

PIR は、MII を経由して PHY-LSI 内部のレジスタにアクセスする手段を提供します。

- ビット31～4：予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ビット3：MII マネージメントデータイン (MDI)  
MDIO 端子のレベルを示します。
- ビット2：MII マネージメントデータアウト (MDO)  
MMD ビットが1のとき、本ビットに設定された値を MDIO 端子より出力します。

## 9. イーサネットコントローラ (EtherC)

---

- ・ ビット 1 : MII マネージメントモード (MMD)  
MII とのデータのリード/ライト方向を規定します。  
0 はリード、1 はライト方向を示します。
- ・ ビット 0 : MII マネージメントデータクロック (MDC)  
本ビットに設定された値を MDC 端子より出力し、MII へのマネージメントデータクロックを供給します。  
MII レジスタへのアクセス方法については、「9.3.4 MII レジスタのアクセス方法」を参照してください。

## 9.2.5 MAC アドレス上位設定レジスタ (MAHR)

本レジスタには、48 ビットの MAC アドレスの上位 32 ビットを設定します。通常、本レジスタの設定は、リセット後の初期設定時に行います。

【注】 MAC アドレスの設定は、送信・受信機能が有効状態で書き換えることを禁止します。E-DMAC モードレジスタ (EDMR) の SWR ビットにより EtherC および E-DMAC を初期状態に戻してから再設定してください。

ビット :	31	30	29	28	27	26	25	24
	MA47	MA46	MA45	MA44	MA43	MA42	MA41	MA40
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							
ビット :	23	22	21	20	19	18	17	16
	MA39	MA38	MA37	MA36	MA35	MA34	MA33	MA32
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							
ビット :	15	14	13	12	11	10	9	8
	MA31	MA30	MA29	MA28	MA27	MA26	MA25	MA24
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							
ビット :	7	6	5	4	3	2	1	0
	MA23	MA22	MA21	MA20	MA19	MA18	MA17	MA16
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							

- ・ ビット 31 ~ 0 : MAC アドレスビット 47 ~ 16 (MA47 ~ MA16)  
MAC アドレスの上位 32 ビットを設定します。

【参考】 本 LSI に設定する MAC アドレスが、01-23-45-67-89-AB (16 進数表示) である場合、本レジスタには、H'01234567 を設定します。

## 9. イーサネットコントローラ (EtherC)

### 9.2.6 MAC アドレス下位設定レジスタ (MALR)

本レジスタには、48 ビットの MAC アドレスの下位 16 ビットを設定します。通常、本レジスタの設定は、リセット後の初期設定時に行います。

【注】 MAC アドレスの設定は、送信・受信機能が有効状態で書き換えることを禁止します。E-DMAC モードレジスタ (EDMR) の SWR ビットにより EtherC および E-DMAC を初期状態に戻してから再設定してください。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	MA15	MA14	MA13	MA12	MA11	MA10	MA9	MA8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 15 ~ 0 : MAC アドレスビット 15 ~ 0 (MA15 ~ MA0)  
MAC アドレスの下位 16 ビットを設定します。

【注】 本 LSI に設定する MAC アドレスが、01-23-45-67-89-AB (16 進数表示) である場合、本レジスタには、H'000089AB を設定します。

## 9.2.7 受信フレーム長上限レジスタ (RFLR)

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	RFL11	RFL10	RFL9	RFL8
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	RFL7	RFL6	RFL5	RFL4	RFL3	RFL2	RFL1	RFL0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

本 LSI が受信することのできる最大フレーム長をバイト単位で指定します。

- ・ ビット 31 ~ 12: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 11 ~ 0: 受信フレームデータ長 (RFL)

H'000 ~ H'5EE	1,518 バイト
H'5EF	1,519 バイト
H'5F0	1,520 バイト
:	:
H'7FF	2,047 バイト
H'800 ~ H'FFF	2,048 バイト

- 【注】 1. ここでのフレームデータは、宛先アドレスから CRC データまでを含んだ範囲となります。  
2. ここで指定された値を超えたデータを受信したとき、設定された値を超えた分のデータは廃棄されます。

【参考】 実際には、宛先アドレスからデータまでがメモリ上に転送されます。CRC データは含まれません。

## 9. イーサネットコントローラ (EtherC)

---

### 9.2.8 PHY 部ステータスレジスタ (PSR)

PHY-LSI からのインタフェース信号を読み込むことができます。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	LMON
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

- ・ ビット 31 ~ 1 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 0 : LNKSTA 端子状態 (LMON)

LNKSTA 端子に PHY-LSI から出力される Link 信号を接続することによって、Link 状態を読み込むことができます。極性については、接続する PHY-LSI の仕様を参照してください。

【注】 LMON ビットは、LNKSTA 端子が High レベル時 0、Low レベル時 1 となります。

## 9.2.9 送信リトライオーバーカウンタレジスタ (TROCR)

本レジスタは、送信時に再送を合わせて 16 回の試行で送信できなかったフレーム数を示す 16 ビットのカウンタです。送信を 16 回失敗すると、本レジスタは 1 カウントアップします。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされません。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	TROC15	TROC14	TROC13	TROC12	TROC11	TROC10	TROC9	TROC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	TROC7	TROC6	TROC5	TROC4	TROC3	TROC2	TROC1	TROC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : 送信リトライオーバーカウンタ 15 ~ 0 (TROC15 ~ TROC0)  
送信時に、再送を合わせて 16 回の試行で送信できなかったフレームのカウンタ数を示します。

## 9. イーサネットコントローラ (EtherC)

---

### 9.2.10 衝突検出カウンタレジスタ (CDCR)

本レジスタは、データの送信開始から 512 ビット時間以降に発生した回線上の衝突回数を示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	COLDC15	COLDC14	COLDC13	COLDC12	COLDC11	COLDC10	COLDC9	COLDC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	COLDC7	COLDC6	COLDC5	COLDC4	COLDC3	COLDC2	COLDC1	COLDC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : 衝突検出カウンタ 15 ~ 0 (COLDC15 ~ COLDC0)  
送信開始から 512 ビット時間以降に衝突したカウンタ数を示します。

## 9.2.11 キャリア消失カウンタレジスタ (LCCR)

本レジスタは、データの送信中にキャリアが消失した回数を示す 16 ビットのカウンタです。本レジスタの値が、HFFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	LCC15	LCC14	LCC13	LCC12	LCC11	LCC10	LCC9	LCC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	LCC7	LCC6	LCC5	LCC4	LCC3	LCC2	LCC1	LCC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : 消失キャリアカウント 15 ~ 0 (LCC15 ~ LCC0)  
データ送信中に消失したキャリアのカウント数を示します。

## 9. イーサネットコントローラ (EtherC)

---

### 9.2.12 キャリア未検出カウンタレジスタ (CNDCR)

本レジスタは、プリアンプルを送出中にキャリアを検出できなかった回数を示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	CNDC15	CNDC14	CNDC13	CNDC12	CNDC11	CNDC10	CNDC9	CNDC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	CNDC7	CNDC6	CNDC5	CNDC4	CNDC3	CNDC2	CNDC1	CNDC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : キャリア未検出カウンタ 15 ~ 0 (CNDC15 ~ CNDC0)  
未検出キャリアのカウント数を示します。

### 9.2.13 フレーム長異常カウンタレジスタ (IFLCR)

本レジスタは、データ送信時に4バイト未満のフレーム長のパケットを送信しようとした回数を示す16ビットのカウンタです。本レジスタの値が、H'FFFF (65535)になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は0にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	IFLC15	IFLC14	IFLC13	IFLC12	IFLC11	IFLC10	IFLC9	IFLC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	IFLC7	IFLC6	IFLC5	IFLC4	IFLC3	IFLC2	IFLC1	IFLC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット 15 ~ 0 : フレーム長異常カウンタ 15 ~ 0 (IFLC15 ~ IFLC0)  
異常なフレーム長送信指示のカウンタ数を示します。

## 9. イーサネットコントローラ (EtherC)

### 9.2.14 CRC エラーフレーム受信カウンタレジスタ (CEFCR)

本レジスタは、CRC エラーとなったフレームの受信回数を示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	CEFC15	CEFC14	CEFC13	CEFC12	CEFC11	CEFC10	CEFC9	CEFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	CEFC7	CEFC6	CEFC5	CEFC4	CEFC3	CEFC2	CEFC1	CEFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 15 ~ 0 : CRC エラーフレームカウンタ 15 ~ 0 (CEFC15 ~ CEFC0)

CRC エラーとなったフレームを受信したカウント数を示します。

【参考】 EtherC モードレジスタの CRC エラーフレーム受信許可ビット (ECMR の PRCEF) が 1 の場合、CRC がエラーとなるフレームを受信してもカウントアップしません。

## 9.2.15 フレーム受信エラーカウンタレジスタ (FRECR)

本レジスタは、PHY-LSI から入力される RX-ER 端子により受信エラーとなったフレームの個数を示す 16 ビットのカウンタです。RX-ER 端子がアクティブになる毎に 1 カウントアップします。本レジスタの値が、HFFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	FREC15	FREC14	FREC13	FREC12	FREC11	FREC10	FREC9	FREC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	FREC7	FREC6	FREC5	FREC4	FREC3	FREC2	FREC1	FREC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : フレーム受信エラーカウンタ 15 ~ 0 (FREC15 ~ FREC0)  
フレームを受信中にエラーとなったカウント数を示します。

## 9. イーサネットコントローラ (EtherC)

---

### 9.2.16 64 バイト未満フレーム受信カウンタレジスタ (TSFRCR)

本レジスタは、64 バイト未満のフレームを受信したことを示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	TSFC15	TSFC14	TSFC13	TSFC12	TSFC11	TSFC10	TSFC9	TSFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	TSFC7	TSFC6	TSFC5	TSFC4	TSFC3	TSFC2	TSFC1	TSFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : 64 バイト未満フレーム受信カウンタ 15 ~ 0 (TSFC15 ~ TSFC0)  
64 バイト未満のフレームを受信したカウンタ数を示します。

## 9.2.17 指定バイト超フレーム受信カウンタレジスタ (TLFRCR)

本レジスタは、受信フレーム長上限レジスタ (RFLR) で指定した値を超えるフレームを受信したことを示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	TLFC15	TLFC14	TLFC13	TLFC12	TLFC11	TLFC10	TLFC9	TLFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	TLFC7	TLFC6	TLFC5	TLFC4	TLFC3	TLFC2	TLFC1	TLFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ビット 31 ~ 16: 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 15 ~ 0: 指定バイト超フレーム受信カウンタ 15 ~ 0 (TLFC15 ~ TLFC0)

RFLR の値を超えるフレームを受信したカウンタ数を示します。

【参考】 RFLR が 1518 バイトである場合、1519 バイト以上のフレームを受信するとカウントアップします。

【注】 端数ビットを含むフレームを受信した場合は、本レジスタのカウントアップをしません。この場合は、端数ビットフレーム受信カウンタレジスタ (RFCR) に反映されます。

### 9.2.18 端数ビットフレーム受信カウンタレジスタ (RFCR)

本レジスタは、8ビットに満たない端数ビットデータを含むフレームを受信したことを示す16ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は0にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	RFC15	RFC14	RFC13	RFC12	RFC11	RFC10	RFC9	RFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	RFC7	RFC6	RFC5	RFC4	RFC3	RFC2	RFC1	RFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット 15 ~ 0 : 端数ビットフレーム受信カウンタ 15 ~ 0 (RFC15 ~ RFC0)  
端数ビットデータを含むフレームを受信したカウンタ数を示します。

## 9.2.19 マルチキャストアドレスフレーム受信カウンタレジスタ (MAFCR)

本レジスタは、マルチキャストアドレスを指定するフレームを受信したことを示す 16 ビットのカウンタです。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。本レジスタへの書き込み動作によってカウンタの値は 0 にクリアされます。書き込む値は、いずれでも構いません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	MAFC15	MAFC14	MAFC13	MAFC12	MAFC11	MAFC10	MAFC9	MAFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	MAFC7	MAFC6	MAFC5	MAFC4	MAFC3	MAFC2	MAFC1	MAFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 16 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 15 ~ 0 : マルチキャストアドレスフレームカウンタ 15 ~ 0 (MAFC15 ~ MAFC0)  
マルチキャストフレームを受信したカウンタ数を示します。

## 9.3 動作説明

EtherC は、送信 E-DMAC から送信指示があると、決められた送信手順に従って送信を開始します。そして、指定語数を転送すると 1 フレームの送信を終了します。

自局宛 (ブロードキャストなどを含む) のフレームを受信すると、フレームフォーマットをチェックしながら受信 E-DMAC に転送します。フレーム受信の最後で CRC チェックを行い、1 フレームの受信を完了します。

- 【参考】
- (1) 実際の EtherC の動作は、E-DMAC との組み合わせにより連続するフレームの送信ならびに受信を行います。連続する動作については、E-DMAC 動作を参照してください。
  - (2) 受信 E-DMAC によりメモリ上に転送される受信データには、CRC データは含まれません。

### 9.3.1 送信動作

EtherC の主な送信機能を以下に示します。

- フレームの生成と送信：回線の状態を監視した後、送信するデータにプリアンプル、SFD および CRC を付加し、MII に送出
- CRC 生成：データ部の CRC を生成し、送信フレームに付加
- 送信リトライ：コリジョンウィンドウ (送信開始よりプリアンプル、SFD を含む 512bit のデータ送出中) 内に衝突を検出した場合、15 回までのバックオフアルゴリズムに基づく再送

EtherC 送信部の状態遷移を図 9.2 に示します。

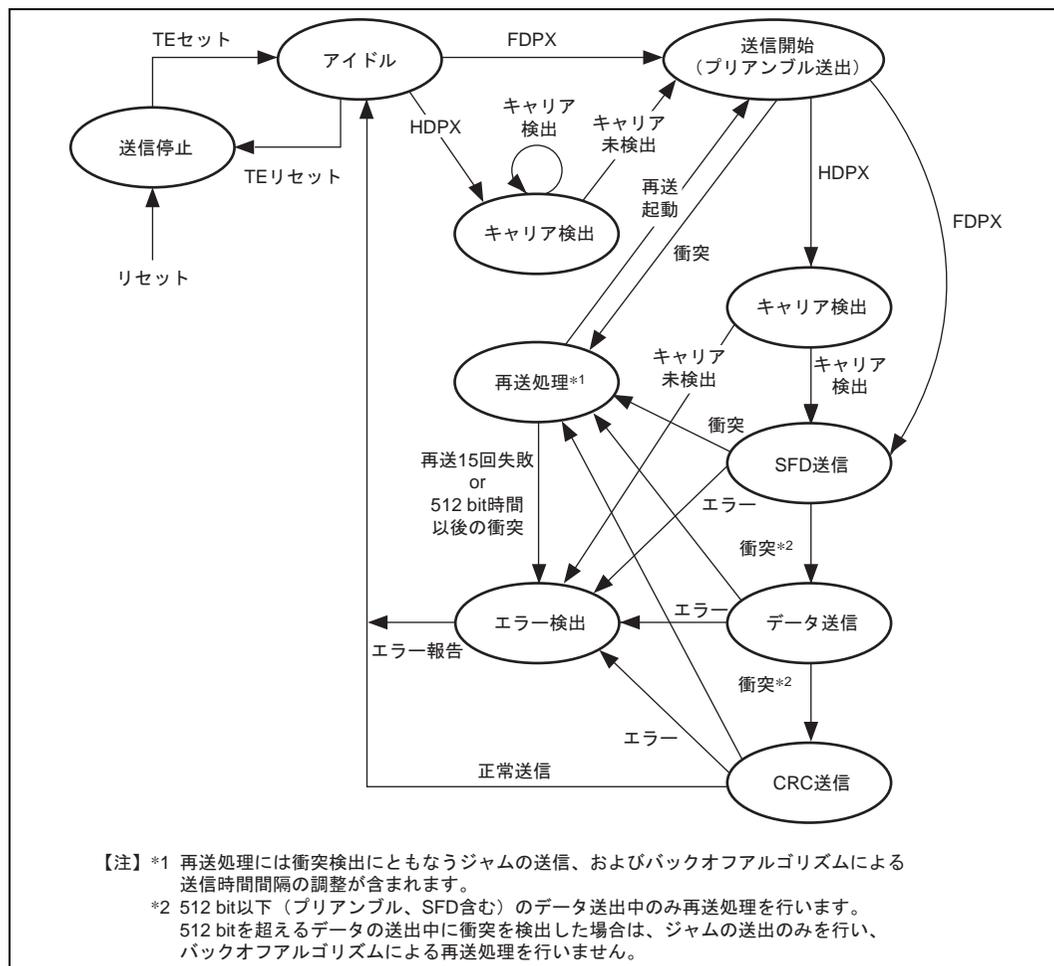


図 9.2 EtherC 送信部状態遷移

- (1) 送信許可 (TE) ビットがセットされると、送信アイドル状態に遷移します。
- (2) 送信 E-DMAC から送信要求があると EtherC はキャリア検出、フレーム間隔時間の送信延期を経てプリアンブルを送出します。

【参考】 キャリア検出を必要としない全二重転送方式を選択している場合、送信 E-DMAC から送信要求があると、即座にプリアンブルを送出します。

- (3) SFD、データ、CRC を順次送信します。送信を終了すると送信 E-DMAC が送信終了割り込み (TC) を発生します。

【参考】 データ送信中に衝突発生あるいはキャリア未検出状態となると、それぞれを割り込み要因として報告します。

- (4) フレーム間隔時間 (10Base では 9.6 $\mu$ s、100Base では 0.96 $\mu$ s) を経た後、アイドル状態に遷移し、以後送信データがあれば送信を継続します。

### 9.3.2 受信動作

EtherC 受信部は、受信したフレームをプリアンブル、SFD、データおよび CRC データに分解し、受信 E-DMAC には DA (宛先アドレス) から CRC データまでが転送されます。主な受信機能を以下に示します。

- 受信フレームヘッダチェック：プリアンブルと SFD のチェックを行い、不当パターンの場合はフレームを廃棄
- 受信フレームデータチェック：ヘッダ内のデータ長をチェックし、64 バイト未満あるいは指定したバイト数を超えるデータ長の場合、エラーステータスを報告
- 受信 CRC チェック：フレームデータ部の CRC チェックを行い、異常時はエラーステータスを報告
- 回線状態の監視：PHY-LSI からの障害検出信号により、不正キャリアを検出した場合、エラーステータスを報告
- Magic Packet の監視：全受信フレームから Magic Packet を検出

EtherC 受信部の状態遷移を図 9.3 に示します。

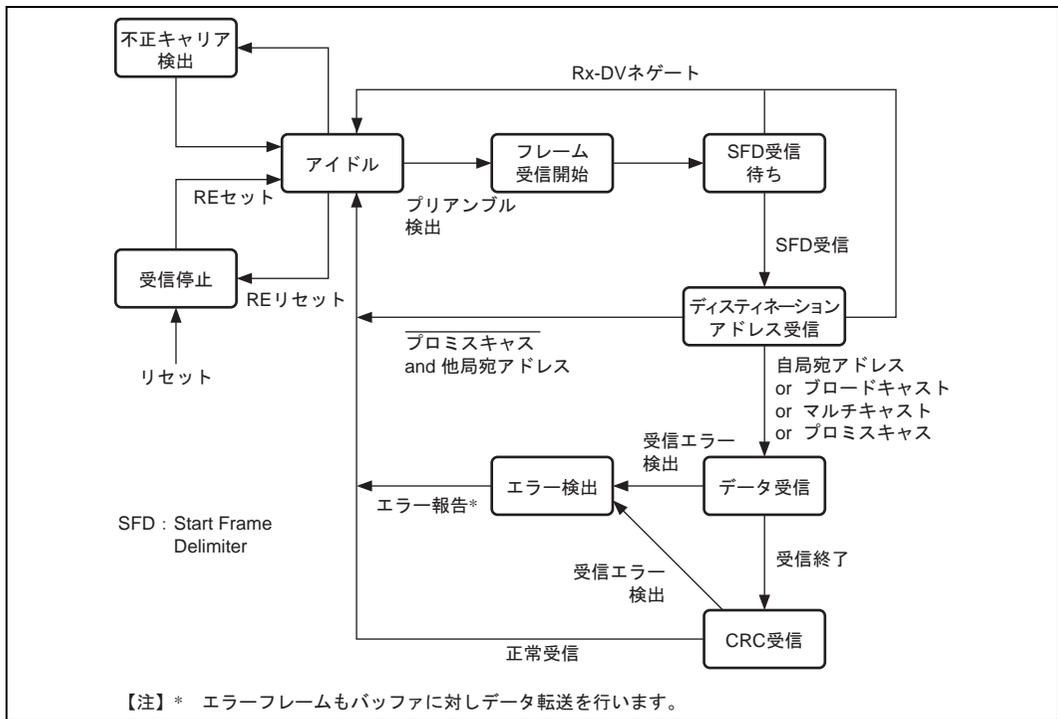


図 9.3 EtherC 受信部状態遷移

- (1) 受信許可 (RE) ビットがセットされると、受信アイドル状態に遷移します。
- (2) 受信パケットのプリアンブルに続く SFD (スタートフレームデリミタ) を検出すると受信処理を開始します。
- (3) ディスティネーションアドレスが、自局当て、あるいはブロードキャスト、マルチキャストを指定、あるいはプロミスキャスモードを指定した場合、データの受信を開始します。

- (4) データ受信後、CRC チェックを行います。結果はメモリ上へのフレームデータを書き込み後、ディスクリプタ内にステータスとして反映されます。
- (5) 1 フレームを受信後、EtherC モードレジスタ内の受信許可ビットが設定 (RE = 1) されていると、次のフレーム受信に備えます。

### 9.3.3 MII フレームタイミング

各種 MII フレームのタイミングを図 9.4 に示します。フレーム送信時における正常タイミングを図 9.4 (a) に、送信時に衝突が発生した場合を図 9.4 (b) に、また送信中にエラーが発生した場合を図 9.4 (c) にそれぞれ示します。またフレーム受信時における正常タイミングを図 9.4 (d) に、また受信中にエラーが発生した場合を図 9.4 (e) と図 9.4 (f) に示します。

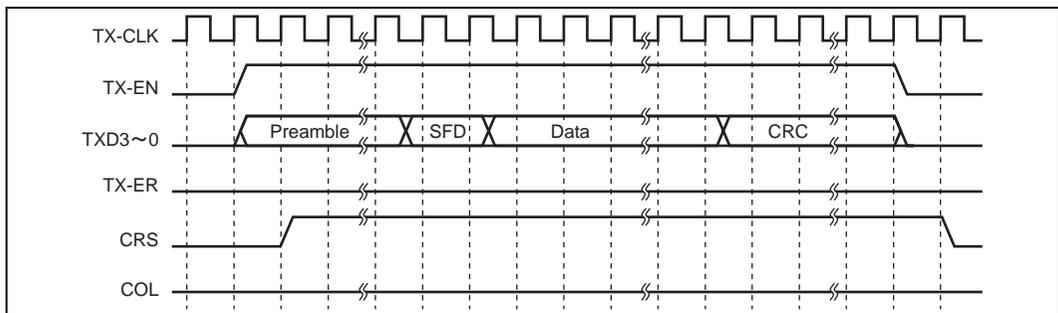


図 9.4 (a) MII フレーム送信タイミング (正常時)

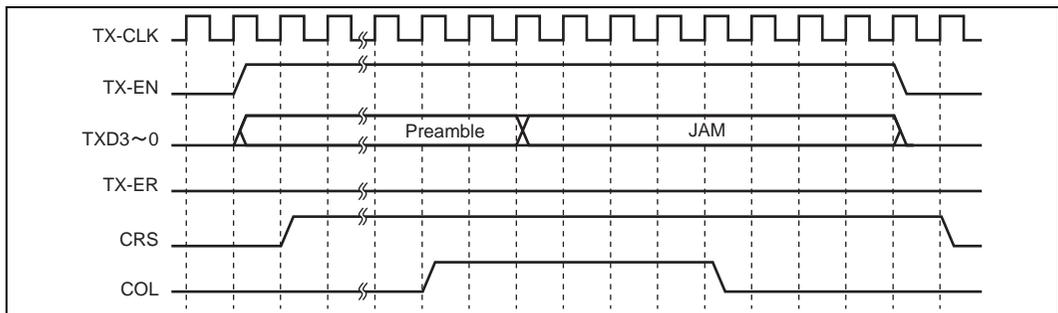


図 9.4 (b) MII フレーム送信タイミング (衝突発生)

9. イーサネットコントローラ (EtherC)

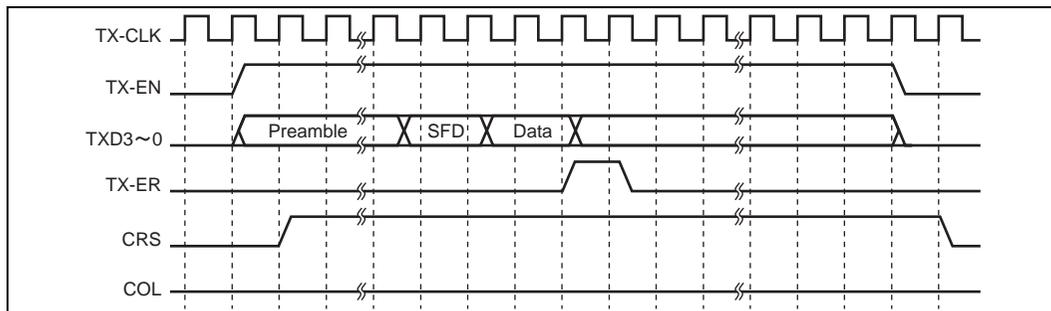


図 9.4 (c) MII フレーム送信タイミング (送信エラー発生)

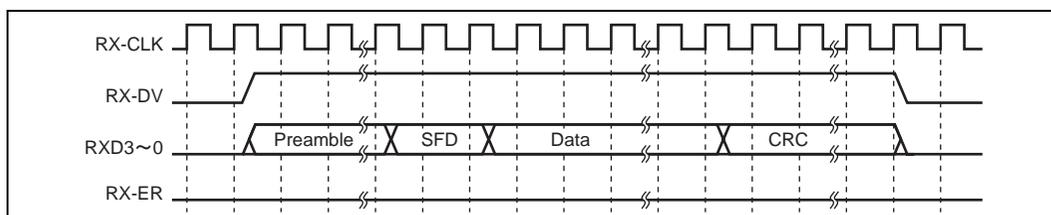


図 9.4 (d) MII フレーム受信タイミング (正常受信)

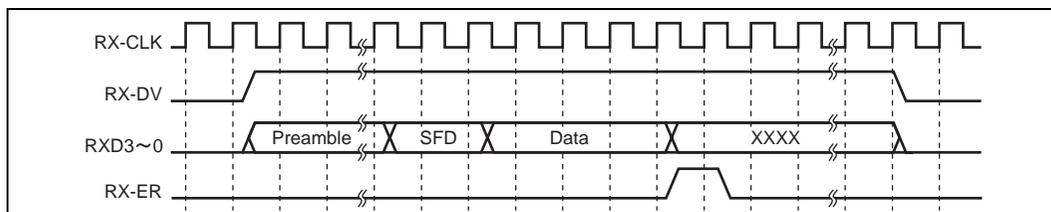


図 9.4 (e) MII フレーム受信タイミング (受信エラー (1))

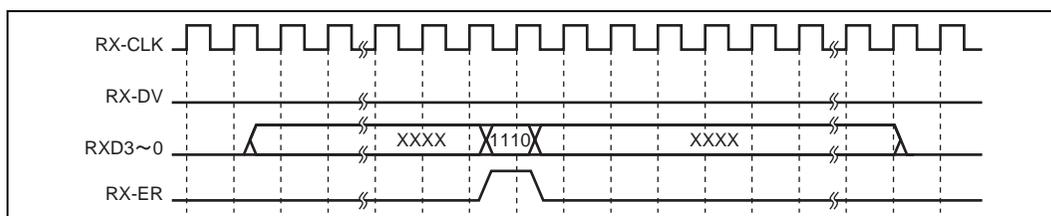


図 9.4 (f) MII フレーム受信タイミング (受信エラー (2))

### 9.3.4 MII レジスタのアクセス方法

PHY-LSI 内にある MII レジスタへは、本 LSI の PHY 部インタフェースレジスタ (PIR) を経由してアクセスします。IEEE802.3u で規定される MII フレームフォーマットに従い、シリアルインタフェースとして接続します。

#### (1) MII 管理フレームのフォーマット

MI I 管理フレームのフォーマットを図 9.5 に示します。MI I レジスタをアクセスするには、(2) で示す手順に従う管理フレームをプログラムによって実現します。

アクセス種別	MI I 管理フレーム							
項目	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
ビット数	32	2	2	5	5	2	16	
リード	1..1	01	10	00001	RRRRR	Z0	D..D	
ライト	1..1	01	01	00001	RRRRR	10	D..D	X

PRE : 32個の連続した1  
 ST : フレームの先頭を示す01の書き込み  
 OP : アクセス種別を示すコードの書き込み  
 PHYAD : PHY-LSIのアドレスが1番の場合、0001を書き込む (MSBから順次書き込む)。このビットは、PHY-LSIアドレスによって可変となる。  
 REGAD : レジスタアドレスが1番の場合、0001を書き込む (MSBから順次書き込む)。このビットは、PHY-LSIのレジスタアドレスによって可変となる。  
 TA : MIIインタフェース上でデータの送信元を切り替える時間  
 (a) ライト時は10を書き込む  
 (b) リード時は、「バス解放」(Z0と表記)を行う  
 DATA : 16ビットのデータ。MSBから順次書き込みあるいは読み出し  
 (a) ライト時は、16ビットデータの書き込み  
 (b) リード時は、16ビットデータの読み出し  
 IDLE : 次のMI I管理フォーマット入力までの待機時間  
 (a) ライト時は、「単独バス解放」(Xと表記)を行う  
 (b) リード時は、すでにTA時にバス解放済みであり制御不要

図 9.5 MII 管理フレームフォーマット

#### (2) MII レジスタアクセス手順

プログラムは、PHY 部インタフェースレジスタ (PIR) を経由して MII レジスタをアクセスします。アクセスは、1 ビット単位のデータ書き込み、1 ビット単位のデータ読み出し、バスの解放および単独バス解放の組み合わせによって実現します。

(1) ~ (4) に MII レジスタアクセスタイミング例を示します。本タイミングは、PHY-LSI の種類によって異なります。

- (1) MII レジスタへの書き込み手順を図 9.6 (a) に示します。
- (2) バス解放手順を図 9.6 (b) に示します。
- (3) MII レジスタからの読み出し手順を図 9.6 (c) に示します。
- (4) 単独バス解放手順を図 9.6 (d) に示します。

9. イーサネットコントローラ (EtherC)

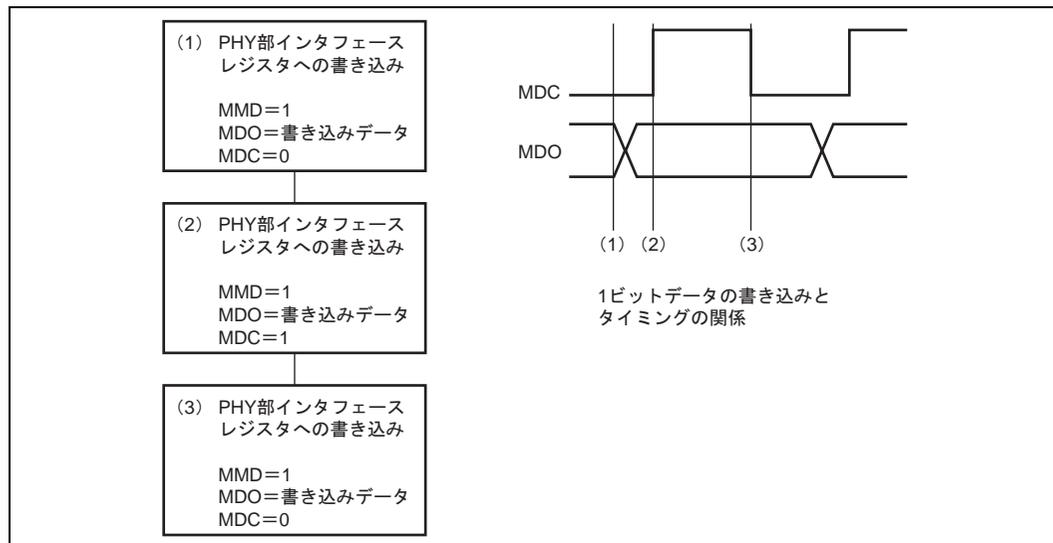


図 9.6 (a) 1 ビットデータの書き込みフロー

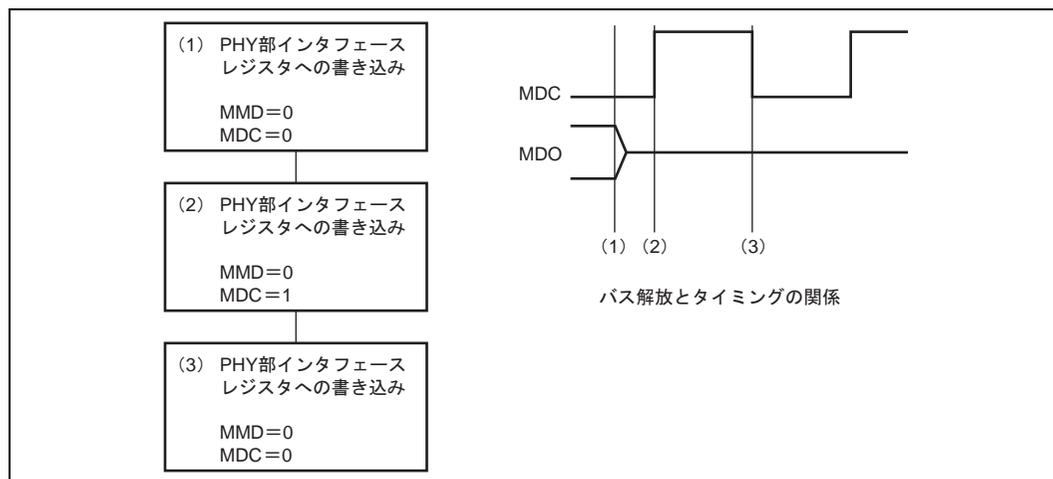


図 9.6 (b) バス解放フロー (図 9.5 中のリード時の TA)

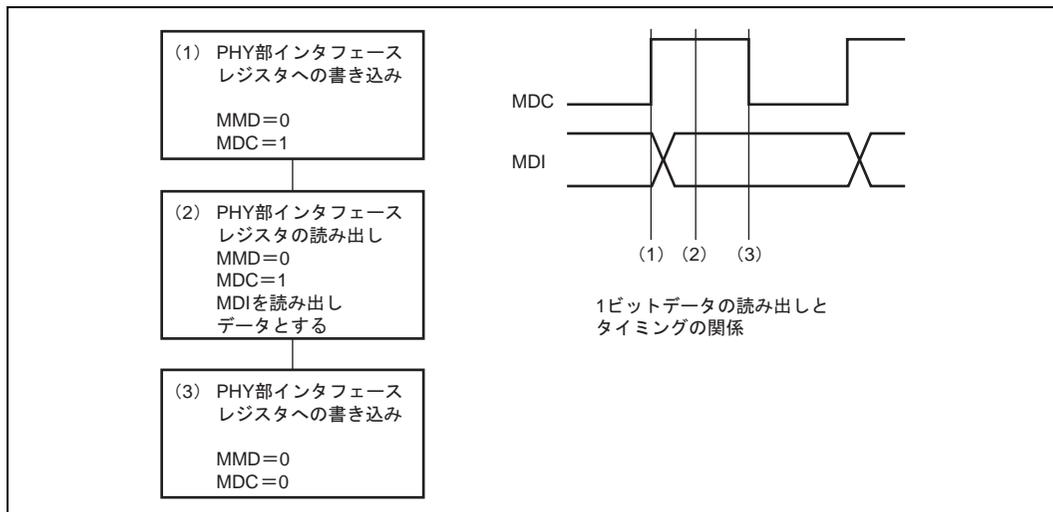


図 9.6 (c) 1 ビットデータ読み出しフロー

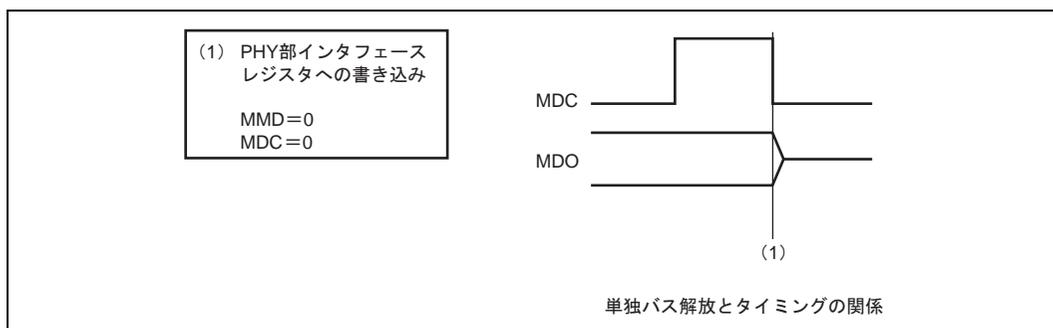


図 9.6 (d) 単独バス解放フロー (図 9.5 中のライト時の IDLE)

### 9.3.5 Magic Packet の検出

EtherC は、Magic Packet の検出機能を有しています。本機能はホスト装置などから LAN に接続される各種周辺装置を起動する機能 (WOL: Wake-On-LAN) を提供します。これによって、ホスト装置などから送出される Magic Packet を周辺装置が受信し、周辺装置が自ら起動するシステムを構築できます。Magic Packet に関する詳細については、AMD 社の技術資料を参照してください。

本 LSI を用いて WOL を利用するには、以下のような設定順序で行います。

- (1) 各種割り込み許可 / マスクレジスタによって割り込み要因の出力を禁止します。
- (2) EtherC モードレジスタ (ECMR) の Magic Packet 検出許可ビット (MPDE) を設定します。
- (3) EtherC 割り込み許可レジスタ (ECSIPR) の Magic Packet 検出割り込み許可ビット (ICDIP) をインエーブルに設定します。
- (4) 必要なら CPU の動作モードをスリープモードあるいは周辺機能をモジュールスタンバイモードに設定します。
- (5) Magic Packet を検出すると、CPU には割り込みが通知されます。また、周辺 LSI に対しては、WOL 端子により Magic Packet を検出したことを通知します。

- 【注】
1. Magic Packet を検出したときには、それ以前に受信していたブロードキャストパケット等によって、受信 FIFO にはデータが蓄積され、EtherC には受信ステータスなどが報告されています。本割り込み処理から通常の動作に復帰するためには、E-DMAC モードレジスタ (EDMR) 内にあるソフトウェアリセットビット (SWR) により、EtherC および E-DMAC の初期化を実行してください。
  2. Magic Packet においては、宛先アドレスにかかわらず受信を行います。結果として、Magic Packet 内のフォーマットで指定される宛先に合致する場合のみ有効となり WOL 端子が有効となります。

### 9.3.6 CPU 動作モードとイーサネットコントローラの動作

本 LSI には、不要な内蔵モジュールの動作を停止させるモジュールスタンバイ、CPU の機能を停止するスリープモード、および全機能を停止するスタンバイ機能の 3 種類を選択あるいは組み合わせ、消費電力の少ないシステムを構築することが可能です。各動作モードの詳細については、「20. 低消費電力状態」を参照してください。ここでは、イーサネットコントローラと組み合わせた場合の特長ならびに注意事項について説明します。

#### ・スリープモード：

CPU と DSP は動作を停止します。EtherC や内蔵周辺機能、外部端子は動作を継続します。EtherC や周辺モジュールからの割り込み、あるいはリセットによって復帰することが可能です。また Magic Packet の受信によって外部端子 WOL 端子を制御するには、当該端子を事前に設定してください。

- 【参考】 Magic Packet による復帰を指定するためには、周辺機能からの割り込み要因をマスクした後に、スリープモードに移行してください。設定方法などは、「9.3.5 Magic Packet の検出」を参照してください。

- スタンバイモード :

本動作モードでは、内蔵発信回路も停止します。このため EtherC へのクロックも供給されないため、EtherC などの周辺モジュールから割り込みを通知することができません。このため通常の動作へ復帰することはできません。したがって WOL 機能を使用することはできません。

【参考】 EtherC を含めた全機能を停止するためには、本モードを選択することが可能です。しかし通常の動作に復帰するためには、NMI 割り込み、パワーオンリセットあるいはマニュアルリセットのいずれかが必要となります。

【注】 本 LSI をスタンバイモードに設定した場合、CPU や DSP をはじめバスステートコントローラなども停止します。DRAM 接続時にはリフレッシュ機能も停止するので、復帰後はリセット時と同様にメモリなどの初期化が必要となります。

- モジュールスタンバイモード :

各周辺モジュールの動作 / 停止を制御することが可能です。CPU と DSP は動作を継続します。ただし EtherC は、機能の性質上、動作を停止することはできません。通常の動作中に、不要な周辺機能モジュールを停止する場合に利用できます。

## 9.4 PHY-LSI との接続

図 9.7 に Advanced Micro Devices, Inc の PHY-LSI AM79C873 との接続例を、図 9.8 に National Semiconductor Corporation の DP83843 との接続例を示します。

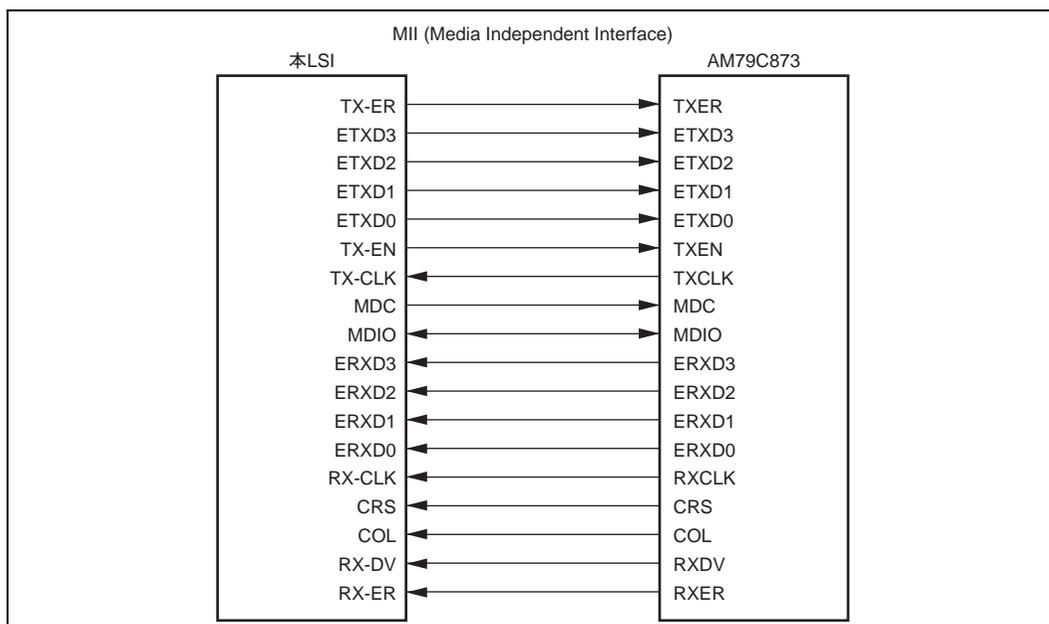


図 9.7 AM79C873 との接続例

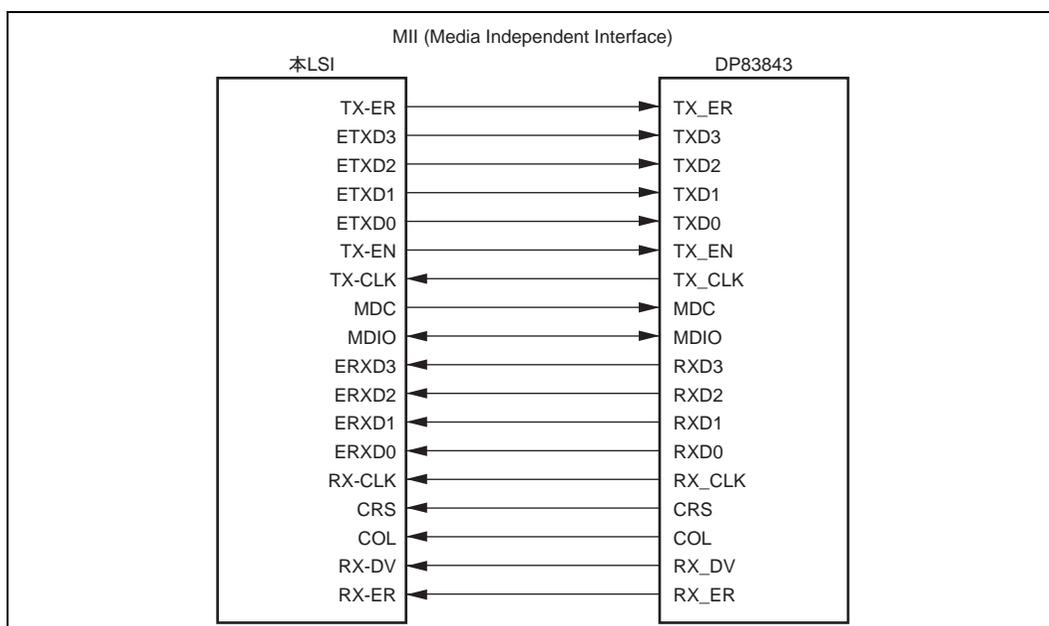


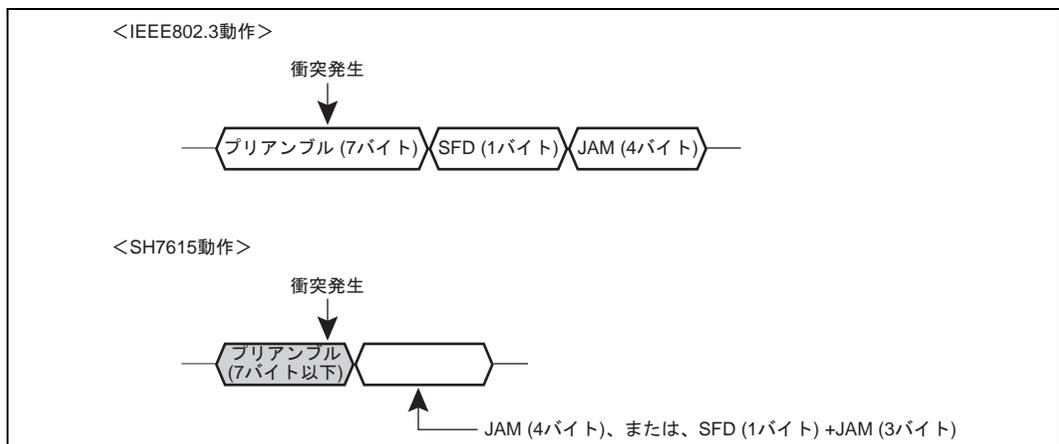
図 9.8 DP83843 との接続例

## 9.5 使用上の注意

SH7615 内蔵イーサネットコントローラ (EtherC) に関しまして、以下のように、一部 IEEE802.3 に準拠しない項目があります。

### 【不具合内容】

プリアンブル送信中に衝突が発生した場合、IEEE802.3 では、送信を中断する前にプリアンブル (7 バイト)、SFD (1 バイト)、JAM (4 バイト) を送信しなければなりません。SH7615 は衝突が発生すると同時にプリアンブルの送信を途中で停止し、JAM (4 バイト) または、SFD (1 バイト)+JAM (3 バイト) を送信した後、送信を停止し再送待ち状態に入ります。



### 【システムへの影響】

衝突の相手局が衝突を認識できずに、MAC レイヤではデータを正常に送信できたと誤認識してしまう可能性があります。この場合、一般には上位レイヤ (例えば、TCP/IP) にて再送によりリカバリを行うため、最終的にはデータは正しく転送されますが、転送効率が下がる可能性があります。



---

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

### 10.1 概要

本 LSI は、2 チャンネルのイーサネットコントローラ (EtherC) に直結したダイレクトメモリアクセスコントローラ (E-DMAC) を内蔵しています。バッファ管理の多くの部分を、E-DMAC 自らディスクリプタを用いて制御します。このため CPU の負荷を軽減し、効率の良いデータ転送制御を行うことができます。

#### 10.1.1 特長

E-DMAC には、以下のような特長があります。

- ディスクリプタ管理方式による CPU 負荷の軽減
- 送受信フレームステータスのディスクリプタへの反映
- ブロック転送 (16 バイト単位) によるシステムバスの効率使用
- シングルフレーム・マルチバッファ方式対応可能

【注】 E-DMAC は、内蔵 RAM および周辺モジュールへのアクセスは行えません。

#### 10.1.2 構成

図 10.1 に E-DMAC とメモリ上のディスクリプタおよび送信 / 受信バッファの構成を示します。

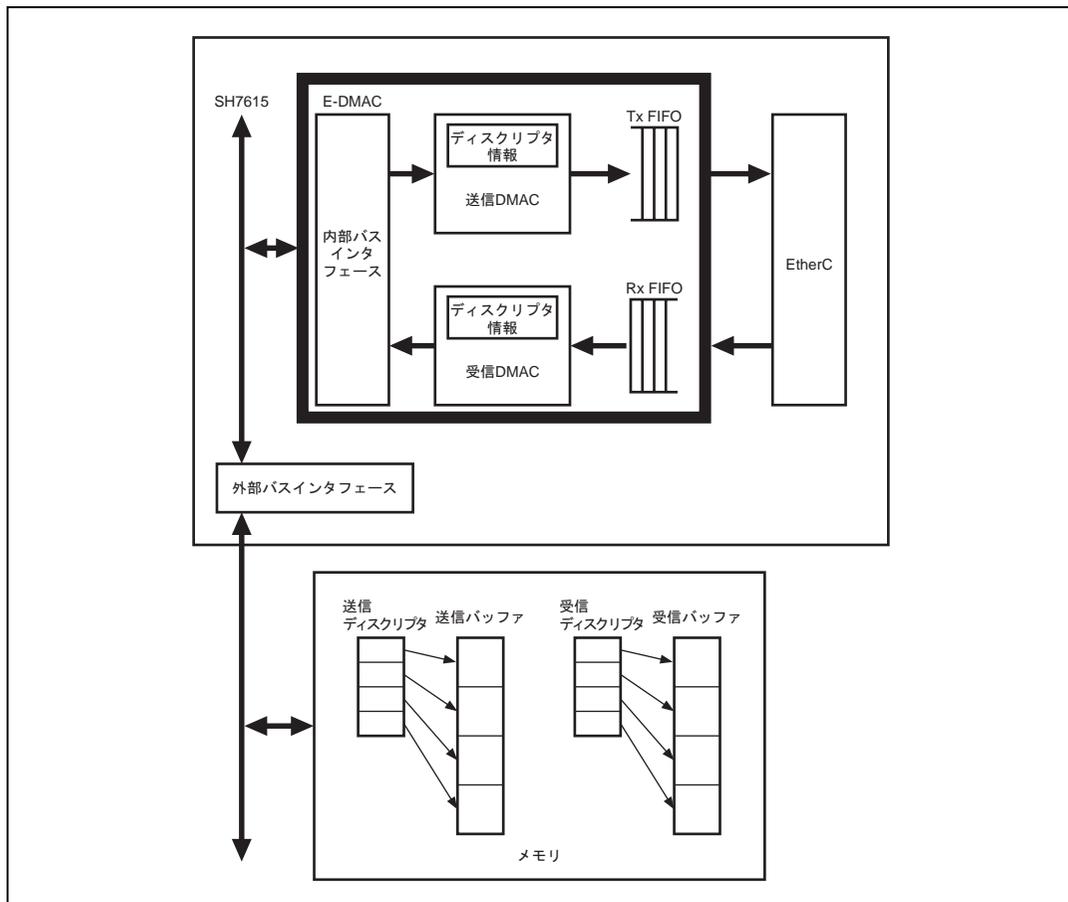


図 10.1 E-DMAC とディスクリプタおよびバッファの構成

### 10.1.3 ディスクリプタ管理方式

E-DMAC は、送受信データバッファについて、対応する送受信ディスクリプタリストを用いて管理します。

#### (1) 送信

送信 E-DMAC は、送信ディスクリプタリストの先頭から送信バッファアドレスを取得し、バッファ内の送信データを送信 FIFO に転送します。ディスクリプタ内の送信指示が継続している場合、次のディスクリプタを読み込み、対応するバッファ内のデータを送信 FIFO に転送します。このようにしてデータ送信を継続することができます。

#### (2) 受信

受信 DMA 転送の開始に当たり、あらかじめ受信 E-DMAC は、受信ディスクリプタリストの先頭から受信バッファアドレスを取得しています。受信 E-DMAC は、受信データが受信 FIFO に貯まると、これを受信バッファに

転送します。1 フレームの受信が終了すると、受信ステータスの書き込みと、次のディスクリプタから受信バッファアドレスを取得します。これを繰り返すことによって連続するフレームを受信することが可能となります。

#### 10.1.4 レジスタ構成

E-DMAC には、表 10.1 に示す 17 個の 32 ビットレジスタがあります。

- 【注】
1. E-DMAC 内のレジスタは、32 ビット単位でアクセスしてください。
  2. レジスタ内にある予約ビットに書き込む場合、必ず 0 としてください。
  3. 予約ビットリード時の値は、これを保証するものではありません。

表 10.1 E-DMAC レジスタ

名称	略称	R/W	初期値	アドレス
E-DMAC モードレジスタ	EDMR	R/W	H'00000000	H'FFFFFFD00
E-DMAC 送信要求レジスタ	EDTRR	R/W	H'00000000	H'FFFFFFD04
E-DMAC 受信要求レジスタ	EDRRR	R/W	H'00000000	H'FFFFFFD08
送信ディスクリプタリスト先頭アドレスレジスタ	TDLAR	R/W	H'00000000	H'FFFFFFD0C
受信ディスクリプタリスト先頭アドレスレジスタ	RDLAR	R/W	H'00000000	H'FFFFFFD10
EtherC/E-DMAC ステータスレジスタ	EESR	R/W*1	H'00000000	H'FFFFFFD14
EtherC/E-DMAC ステータス割り込み許可レジスタ	EESIPR	R/W	H'00000000	H'FFFFFFD18
送受信ステータスコピー指示レジスタ	TRSCER	R*2	H'00000000	H'FFFFFFD1C
ミスフレームカウンタレジスタ	RMFCR	R/W	H'00000000	H'FFFFFFD20
送信 FIFO しきい値指定レジスタ	TFTR	R/W	H'00000000	H'FFFFFFD24
FIFO 容量指定レジスタ	FDR	R/W	H'00000000	H'FFFFFFD28
受信方式制御レジスタ	RCR	R/W	H'00000000	H'FFFFFFD2C
E-DMAC 動作制御レジスタ	EDOC	R/W	H'00000000	H'FFFFFFD30
受信バッファライトアドレスレジスタ	RBWAR	R	H'00000000	H'FFFFFFD40
受信ディスクリプタフェッチアドレスレジスタ	RDFAR	R	H'00000000	H'FFFFFFD44
送信バッファリードアドレスレジスタ	TBRAR	R	H'00000000	H'FFFFFFD4C
送信ディスクリプタフェッチアドレスレジスタ	TDFAR	R	H'00000000	H'FFFFFFD50

- 【注】
- \*1 各ビットに 1 を書き込むことでクリアされます。
  - \*2 本レジスタを読み出すことでクリアされます。

## 10.2 レジスタの説明

### 10.2.1 E-DMAC モードレジスタ (EDMR)

E-DMAC モードレジスタは、E-DMAC の動作モードを指定するレジスタです。本レジスタの設定は、通常リセット後の初期設定時に行います。

【注】 動作モードの設定は、送信・受信機能が有効状態で書き換えることを禁止します。動作モードを切り替えるには、本レジスタ内にあるソフトウェアリセットビット (SWR) により、EtherC および E-DMAC を初期状態に戻してから再設定を行います。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	DL1	DL0	-	-	-	SWR
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W

- ・ ビット 31 ~ 6 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 5 ~ 4 : 送受信ディスクリプタ長 1、0 (DL1、DL0)  
ディスクリプタ長を指定します。

ビット 5	ビット 4	説明
DL1	DL0	
0	0	16 バイト (初期値)
	1	32 バイト
1	0	64 バイト
	1	予約 (設定禁止)

- ・ ビット 3 ~ 1: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 0: ソフトウェアリセット (SWR)  
ソフトウェアによって EtherC および E-DMAC を初期化することができます。  
読み出すと常に 0 が読み出されます。

ビット 0	説 明
SWR	
0	EtherC および E-DMAC のリセットを解除 (初期値)
1	EtherC および E-DMAC をリセットする

- 【注】
1. データ送信中等では、本レジスタによって EtherC および E-DMAC を初期化すると回線上に異常データを送出する可能性があります。
  2. EtherC および E-DMAC は、16 内部クロックで初期化を完了します。このため、EtherC および E-DMAC 内のレジスタアクセスを行う際は、16 内部クロック経過後に行ってください。
  3. E-DMAC の TDLAR、RDLAR、RMFCR レジスタは初期化されません。その他の EtherC、E-DMAC の各レジスタは初期化されます。

### 10.2.2 E-DMAC 送信要求レジスタ (EDTRR)

E-DMAC 送信要求レジスタによって、E-DMAC に送信指示を行います。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TR
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

- ・ ビット 31 ~ 1: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

### ・ ビット 0 : 送信要求 (TR)

本ビットに 1 を書き込むと、E-DMAC はディスクリプタを読み込み、有効なディスクリプタについて、送信バッファ内のデータを EtherC に転送します。

ビット 0	説 明
TR	
0	送信停止状態。0 を書き込んでも送信は停止しません。送信の終了は、送信ディスクリプタ内の有効ビットで制御します。
1	送信開始。該当するディスクリプタを読み込み、送信有効ビットが 1 であるフレームを送信します。

【参考】 1 つのフレームの送信を終了すると、次のディスクリプタを読み込みます。このディスクリプタ内の送信ディスクリプタ有効ビットが有効であれば、送信を続けます。また送信ディスクリプタ有効ビットが無効な場合、この TR ビットをクリアして送信 DMAC の動作を停止します。

【注】 本レジスタの書き込みについては、「10.4 使用上の注意事項」を参照してください。

### 10.2.3 E-DMAC 受信要求レジスタ (EDRRR)

E-DMAC 受信要求レジスタによって、E-DMAC に受信指示を行います。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RR
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

- ・ ビット 31 ~ 1: 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 0: 受信要求 (RR)  
本ビットに 1 を書き込むと、E-DMAC はディスクリプタを読み込みます。以後 EtherC からの受信要求によって、受信データをバッファに転送します。

ビット 0	説明
RR	
0	フレームの受信完了後、受信機能を無効にする (初期値)
1	受信ディスクリプタを読み込み、転送可能状態となる

【参考】 受信要求に伴い、フレームを受信するためには受信ディスクリプタ内の受信ディスクリプタ有効ビットを有効にしておく必要があります。

- \*1 E-DMAC は、受信要求ビットがセットされると、当該受信ディスクリプタを読み込みます。
- \*2 ディスクリプタ内の受信ディスクリプタ有効ビットが有効であれば、EtherC からの受信要求に備えます。
- \*3 受信バッファ分の受信が完了すると、E-DMAC は次のディスクリプタを読み込みフレームの受信に備えます。このとき、ディスクリプタ内の受信ディスクリプタ有効ビットが無効である場合、RR ビットをクリアして受信 DMAC の動作を停止します。

【注】 本レジスタの書き込みについては、「10.4 使用上の注意事項」を参照してください。

## 10.2.4 送信ディスクリプタリスト先頭アドレスレジスタ (TDLAR)

送信ディスクリプタリストの先頭アドレスを設定します。各ディスクリプタは、EDMRのDLビットで示すディスクリプタ長に合致する境界構成とします。

ビット:	31	30	29	28	27	26	25	24
	TDLA31	TDLA30	TDLA29	TDLA28	TDLA27	TDLA26	TDLA25	TDLA24
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	23	22	21	20	19	18	17	16
	TDLA23	TDLA22	TDLA21	TDLA20	TDLA19	TDLA18	TDLA17	TDLA16
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	15	14	13	12	11	10	9	8
	TDLA15	TDLA14	TDLA13	TDLA12	TDLA11	TDLA10	TDLA9	TDLA8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	7	6	5	4	3	2	1	0
	TDLA7	TDLA6	TDLA5	TDLA4	TDLA3	TDLA2	TDLA1	TDLA0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							

ビット 31 ~ 0: 送信ディスクリプタの先頭アドレス 31 ~ 0 (TDLA31 ~ TDLA0)

- 【注】\*1 指定ディスクリプタ長によって下位ビットを以下のように設定します。  
 16 バイトバウンダリ: TDLA[3:0]=0000  
 32 バイトバウンダリ: TDLA[4:0]=00000  
 64 バイトバウンダリ: TDLA[5:0]=000000
- \*2 送信中に本レジスタを書き換えることは禁止します。本レジスタの書き換えは、送信停止状態で行ってください。

## 10.2.5 受信ディスクリプタリスト先頭アドレスレジスタ (RDLAR)

受信ディスクリプタリストの先頭アドレスを設定します。各ディスクリプタは、EDMR の DL ビットで示すディスクリプタ長に合致する境界構成とします。

ビット:	31	30	29	28	27	26	25	24
	RDLA31	RDLA30	RDLA29	RDLA28	RDLA27	RDLA26	RDLA25	RDLA24
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	23	22	21	20	19	18	17	16
	RDLA23	RDLA22	RDLA21	RDLA20	RDLA19	RDLA18	RDLA17	RDLA16
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	15	14	13	12	11	10	9	8
	RDLA15	RDLA14	RDLA13	RDLA12	RDLA11	RDLA10	RDLA9	RDLA8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							
ビット:	7	6	5	4	3	2	1	0
	RDLA7	RDLA6	RDLA5	RDLA4	RDLA3	RDLA2	RDLA1	RDLA0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							

・ ビット 31 ~ 0: 受信ディスクリプタの先頭アドレス 31 ~ 0 (RDLA31 ~ RDLA0)

【注】\*1 指定ディスクリプタ長によって下位ビットを以下のように設定します。

16 バイトバウンダリ: RDLA[3:0]=0000

32 バイトバウンダリ: RDLA[4:0]=00000

64 バイトバウンダリ: RDLA[5:0]=000000

\*2 受信中に本レジスタを書き換えることは禁止します。本レジスタの書き換えは、受信停止状態で行ってください。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

### 10.2.6 EtherC/E-DMAC ステータスレジスタ (EESR)

EtherC と E-DMAC を合わせた通信ステータスを表示します。本レジスタは割り込み要因として報告されます。各ビットは、1 を書き込むことでクリアされます。また EtherC/E-DMAC 割り込み許可レジスタの当該ビットによってマスクすることが可能です。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	TABT	RABT	RFCOF
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W
ビット:	23	22	21	20	19	18	17	16
	-	ECI	TC	TDE	TFUF	FR	RDE	RFOF
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	15	14	13	12	11	10	9	8
	-	-	-	ITF	CND	DLC	CD	TRO
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	RMAF	-	-	RRF	RTLF	RTSF	PRE	CERF
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

- ビット 31 ~ 27 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ビット 26 : 送信アボート検出 (TABT)  
送信アボートを検出したことを示します。

ビット 26	説明
TABT	
0	送信アボート未検出 (初期値)
1	送信アボート検出

下記のいずれかのビットが 1 にセットされると本ビットがセットされます。

- EESR ビット 12 : 送信フレーム長異常 (ITF)
- EESR ビット 11 : キャリア未検出 (CND)
- EESR ビット 10 : キャリア消失検出 (DLC)
- EESR ビット 9 : 遅延衝突検出 (CD)

- ビット 25 : 受信アボート検出 (RABT)  
受信アボートを検出したことを示します。

ビット 25	説明
RABT	
0	受信アボート未検出 (初期値)
1	受信アボート検出

下記のいずれかのビットが 1 にセットされると本ビットがセットされます。

- EESR ビット 4 : 端数ビットフレーム受信 (RRF)
- EESR ビット 3 : ロングフレーム受信エラー (RTLF)
- EESR ビット 2 : ショートフレーム受信エラー (RTSF)
- EESR ビット 1 : PHY-LSI 受信エラー (PRE)
- EESR ビット 0 : 受信フレーム CRC エラー (CERF)

- ビット 24 : 受信フレームカウンタオーバーフロー (RFCOF)  
受信 FIFO 内のフレームカウンタがオーバーフローしたことを示します。

ビット 24	説明
RFCOF	
0	受信フレームカウンタが未オーバーフロー (初期値)
1	受信フレームカウンタオーバーフロー (割込み要因)

【参考】 E-DMAC 内の受信 FIFO には、最大 8 個までのフレームを格納することができます。

受信 FIFO 内に 8 個のフレームが残っている状態で、9 個目のフレームを受信すると、受信フレームカウンタがオーバーフローとなり、このフレームは廃棄されます。なお廃棄されたフレームは、ミスドフレームカウンタレジスタにカウントアップされます。受信 FIFO 内にある 8 個のフレームは保持され、DMA 転送が可能となればメモリ上に転送されます。また、DMA 転送に伴い、フレームカウンタが 8 以下になれば、再びフレームを受信します。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

- ビット 23 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 22 : EtherC ステータスレジスタ要因 (ECI)

EtherC ステータスレジスタ (ECSR) 要因による割り込みを検出したことを示します。

ビット 22	説明
ECI	
0	EtherC ステータス割り込み要因未検出 (初期値)
1	EtherC ステータス割り込み要因検出 (割り込み要因)

【注】 本ビットは、読み出し専用です。EtherC にある ECSR の要因がクリアされると、本ビットもクリアされます。

- ビット 21 : フレーム送信完了 (TC)

送信ディスクリプタによって指定されたデータをすべて EtherC 部より送信したことを示します。1 フレーム / 1 バッファ処理では、1 フレームの送信が完了した場合、またマルチバッファフレーム処理ではフレーム最後のデータを送信し、次のディスクリプタ内の送信ディスクリプタ有効ビット (TACT) がセットされていなかった場合に送信完了とみなし、本ビットが 1 となります。フレーム送信完了後、E-DMAC は転送状態を当該ディスクリプタにライトバックします。

ビット 21	説明
TC	
0	転送未完了あるいは転送未指示 (初期値)
1	転送完了 (割り込み要因)

【参考】 回線上への送出は、EtherC から MII を経由して PHY-LSI によって行われるため、実際の送信完了時間は、さらに遅れる場合があります。

- ビット 20 : 送信ディスクリプタ枯渇 (TDE)

マルチバッファフレーム処理で前ディスクリプタがフレームの最終でない場合、E-DMAC が送信ディスクリプタを読み込んだときに、ディスクリプタ内の送信ディスクリプタ有効ビット (TACT) がセットされていなかったことを示します。結果として不完全なフレームを送出する場合があります。

ビット 20	説明
TDE	
0	送信ディスクリプタ有効ビット TACT = 1 を検出 (初期値)
1	送信ディスクリプタ有効ビット TACT = 0 を検出 (割り込み要因)

【注】 送信ディスクリプタ枯渇 (TDE = 1) が発生した場合、ソフトウェアリセットしてから送信起動をかけてください。このとき、送信ディスクリプタリスト先頭アドレスレジスタ (TDLAR) に格納されているアドレスからの開始となります。

・ ビット 19: 送信 FIFO アンダフロー (TFUF)

フレームを送信中に送信 FIFO にアンダフローが発生したことを示します。回線には、不完全なデータが送出されます。

ビット 19	説明
TFUF	
0	アンダフロー未発生 (初期値)
1	アンダフロー発生 (割り込み要因)

【参考】 アンダフロー発生後の E-DMAC 動作を継続するか停止するかは、E-DMAC 動作制御レジスタ (EDOCR) で制御できます。

・ ビット 18: フレーム受信 (FR)

フレームを受信し、受信ディスクリプタを更新したことを示します。本ビットは 1 フレームを受信する度に 1 にセットされます。

ビット 18	説明
FR	
0	フレーム未受信 (初期値)
1	フレーム受信済み (割り込み要因)

【参考】 実際の受信フレームのステータスはディスクリプタ内にある受信ステータスに反映されています。

・ ビット 17: 受信ディスクリプタ枯渇 (RDE)

受信ディスクリプタを E-DMAC が読み込んだときに、受信ディスクリプタ有効ビット (RACT) が無効な場合 (RACT = 0) 本ビットがセットされます。

ビット 17	説明
RDE	
0	受信ディスクリプタ有効ビット RACT = 1 を検出 (初期値)
1	受信ディスクリプタ有効ビット RACT = 0 を検出 (割り込み要因)

【注】 受信ディスクリプタ枯渇 (RDE = 1) が発生した場合、当該受信ディスクリプタを RACT = 1 に設定し受信起動をかけることで、受信を再開することができます。

・ ビット 16: 受信 FIFO オーバフロー (RFOF)

フレームを受信中に受信 FIFO がオーバフローしたことを示します。

ビット 16	説明
RFOF	
0	オーバフロー未発生 (初期値)
1	オーバフロー発生 (割り込み要因)

【参考】 1. 複数個の受信フレームが受信 FIFO 内にある場合、これらは正しくメモリ上に転送されます。オーバフローの発生したフレームは、受信ディスクリプタに状態がライトバックされます。  
2. オーバフロー発生後、E-DMAC の動作を継続するか停止するかは、E-DMAC 動作制御レジスタ (EDOCR) で制御できます。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

- ・ ビット 15 ~ 13 : 予約ビット  
書き込む場合は、必ず 0 としてください。

- ・ ビット 12 : 送信フレーム長異常 (ITF)  
送信フレーム長指示が 4 バイト未満であったことを示します。

ビット 12	説 明
ITF	
0	送信フレーム長正常 (初期値)
1	送信フレーム長異常 (割り込み要因)

- ・ ビット 11 : キャリア未検出 (CND)  
キャリアの検出状態を示します。

ビット 11	説 明
CND	
0	送信開始時にキャリア検出 (初期値)
1	キャリア未検出 (割り込み要因)

- ・ ビット 10 : キャリア消失検出 (DLC)  
フレーム送信中のキャリア消失を検出したことを示します。

ビット 10	説 明
DLC	
0	キャリア消失未検出 (初期値)
1	キャリア消失検出 (割り込み要因)

- ・ ビット 9 : 遅延衝突検出 (CD)  
フレーム送信中に衝突を検出したことを示します。

ビット 9	説 明
CD	
0	遅延衝突未検出 (初期値)
1	遅延衝突検出 (割り込み要因)

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

・ ビット 8 : 送信リトライオーバ (TRO)

フレーム送信中にリトライオーバが発生したことを示します。これは、EtherC が送信を開始後、バックオフアルゴリズムに基づく 15 回の再送を併せ全部で 16 回の送信試行に失敗したことを示します。

ビット 8	説 明
TRO	
0	送信リトライオーバ未検出 (初期値)
1	送信リトライオーバ検出 (割り込み要因)

・ ビット 7 : マルチキャストアドレスフレーム受信 (RMAF)

マルチキャストアドレスフレームを受信したことを示します。

ビット 7	説 明
RMAF	
0	マルチキャストアドレスフレーム未受信 (初期値)
1	マルチキャストアドレスフレーム受信 (割り込み要因)

・ ビット 6 ~ 5 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 4 : 端数ビットフレーム受信 (RRF)

端数ビットフレームを受信したことを示します。

ビット 4	説 明
RRF	
0	端数ビットフレーム未受信 (初期値)
1	端数ビットフレーム受信 (割り込み要因)

・ ビット 3 : ロングフレーム受信エラー (RTLFL)

1519 バイト以上のフレームを受信したことを示します。

ビット 3	説 明
RTLFL	
0	ロングフレーム未受信 (初期値)
1	ロングフレーム受信 (割り込み要因)

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

- ビット2: ショートフレーム受信エラー (RTSF)  
64 バイト未満のフレームを受信したことを示します。

ビット2	説明
RTSF	
0	ショートフレーム未受信 (初期値)
1	ショートフレーム受信 (割り込み要因)

- ビット1: PHY-LSI 受信エラー (PRE)  
MII (PHY-LSI) からエラー報告のあったことを示します。

ビット1	説明
PRE	
0	PHY-LSI 受信エラー未検出 (初期値)
1	PHY-LSI 受信エラー検出 (割り込み要因)

- ビット0: 受信フレーム CRC エラー (CERF)  
受信したフレームから CRC エラーを検出したことを示します。

ビット0	説明
CERF	
0	CRC エラー未検出 (初期値)
1	CRC エラー検出 (割り込み要因)

## 10.2.7 EtherC/E-DMAC ステータス割り込み許可レジスタ (EESIPR)

EtherC/E-DMAC ステータスレジスタの各ビットに対応する割り込み許可レジスタです。各ビットは、1 を書き込むことで割り込みが許可されます。初期状態では、割り込みは許可されていません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	RFCOFIP
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W
ビット:	23	22	21	20	19	18	17	16
	-	ECIIP	TCIP	TDEIP	TFUFIP	FRIP	RDEIP	RFOFIP
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	15	14	13	12	11	10	9	8
	-	-	-	ITFIP	CNDIP	DLCIP	CDIP	TROIP
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	RMAFIP	-	-	RRFIP	RTLFIP	RTSFIP	PREIP	CERFIP
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

- ・ ビット 31 ~ 25 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 24 : 受信フレームカウンタオーバーフロー割り込み許可 (RFCOFIP)  
受信フレームカウンタオーバーフロー割り込みを許可します。

ビット 24	説明
RFCOFIP	
0	受信フレームカウンタオーバーフロー割り込み禁止 (初期値)
1	受信フレームカウンタオーバーフロー割り込み許可

- ・ ビット 23 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 22 : EtherC ステータスレジスタ要因割り込み許可 (ECIP)  
EtherC ステータスレジスタ要因による割り込みを許可します。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

ビット 22	説 明	
ECIP		
0	EtherC ステータス割り込み禁止	(初期値)
1	EtherC ステータス割り込み許可	

- ・ ビット 21 : フレーム送信完了割り込み許可 (TCIP)  
フレーム送信完了割り込みを許可します。

ビット 21	説 明	
TCIP		
0	フレーム送信完了割り込み禁止	(初期値)
1	フレーム送信完了割り込み許可	

- ・ ビット 20 : 送信ディスクリプタ枯渇割り込み許可 (TDEIP)  
送信ディスクリプタ枯渇割り込みを許可します。

ビット 20	説 明	
TDEIP		
0	送信ディスクリプタ枯渇割り込み禁止	(初期値)
1	送信ディスクリプタ枯渇割り込み許可	

- ・ ビット 19 : 送信 FIFO アンダフロー割り込み許可 (TFUFIP)  
送信 FIFO アンダフロー割り込みを許可します。

ビット 19	説 明	
TFUFIP		
0	アンダフロー割り込み禁止	(初期値)
1	アンダフロー割り込み許可	

- ・ ビット 18 : フレーム受信割り込み許可 (FRIP)  
フレーム受信割り込みを許可します。

ビット 18	説 明	
FRIP		
0	フレーム受信割り込み禁止	(初期値)
1	フレーム受信割り込み許可	

- ・ ビット 17 : 受信ディスクリプタ枯渇割り込み許可 (RDEIP)  
受信ディスクリプタ枯渇割り込みを許可します。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

ビット 17	説 明
RDEIP	
0	受信ディスクリプタ枯渇割り込み禁止 (初期値)
1	受信ディスクリプタ枯渇割り込み許可

- ・ ビット 16 : 受信 FIFO オーバフロー割り込み許可 (RFOFIP)  
受信 FIFO オーバフロー割り込みを許可します。

ビット 16	説 明
RFOFIP	
0	オーバフロー割り込み禁止 (初期値)
1	オーバフロー割り込み許可

- ・ ビット 15 ~ 13 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 12 : 送信フレーム長異常割り込み許可 (ITFIP)  
送信フレーム長異常割り込みを許可します。

ビット 12	説 明
ITFIP	
0	送信フレーム長異常割り込み禁止 (初期値)
1	送信フレーム長異常割り込み許可

- ・ ビット 11 : キャリア未検出割り込み許可 (CNDIP)  
キャリア未検出割り込みを許可します。

ビット 11	説 明
CNDIP	
0	キャリア未検出割り込み禁止 (初期値)
1	キャリア未検出割り込み許可

- ・ ビット 10 : キャリア消失検出割り込み許可 (DLCIP)  
キャリア消失検出割り込みを許可します。

ビット 10	説 明
DLCIP	
0	キャリア消失検出割り込み禁止 (初期値)
1	キャリア消失検出割り込み許可

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

- ・ ビット 9 : 衝突検出割り込み許可 (CDIP)  
衝突検出割り込みを許可します。

ビット 9	説 明
CDIP	
0	衝突割り込み禁止 (初期値)
1	衝突割り込み許可

- ・ ビット 8 : 送信リトライオーバー割り込み許可 (TROIP)  
送信リトライオーバー割り込みを許可します。

ビット 8	説 明
TROIP	
0	送信リトライオーバー割り込み禁止 (初期値)
1	送信リトライオーバー割り込み許可

- ・ ビット 7 : マルチキャストアドレスフレーム受信割り込み許可 (RMAFIP)  
マルチキャストアドレスフレーム受信割り込みを許可します。

ビット 7	説 明
RMAFIP	
0	マルチキャストアドレスフレーム受信割り込み禁止 (初期値)
1	マルチキャストアドレスフレーム受信割り込み許可

- ・ ビット 6 ~ 5 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 4 : 端数ビットフレーム受信割り込み許可 (RRFIP)  
端数ビットフレーム受信割り込みを許可します。

ビット 4	説 明
RRFIP	
0	端数ビットフレーム受信割り込み禁止 (初期値)
1	端数ビットフレーム受信割り込み許可

- ・ ビット 3 : ロングフレーム受信エラー割り込み許可 (RTLFIP)  
ロングフレーム受信エラー割り込みを許可します。

ビット 3	説 明
RTLFIP	
0	ロングフレーム受信エラー割り込み禁止 (初期値)
1	ロングフレーム受信エラー割り込み許可

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

- ・ ビット2：ショートフレーム受信エラー割り込み許可 (RTSFIP)  
ショートフレーム受信エラー割り込みを許可します。

ビット2	説明
RTSFIP	
0	ショートフレーム受信エラー割り込み禁止 (初期値)
1	ショートフレーム受信エラー割り込み許可

- ・ ビット1：PHY-LSI 受信エラー割り込み許可 (PREIP)  
フレーム受信エラー割り込みを許可します。

ビット1	説明
PREIP	
0	PHY-LSI 受信エラー割り込み禁止 (初期値)
1	PHY-LSI 受信エラー割り込み許可

- ・ ビット0：受信フレーム CRC エラー割り込み許可 (CERFIP)  
受信フレーム CRC エラー割り込みを許可します。

ビット0	説明
CERFIP	
0	CRC エラー割り込み禁止 (初期値)
1	CRC エラー割り込み許可

### 10.2.8 送受信ステータスコピー指示レジスタ (TRSCER)

EtherC/E-DMAC ステータスレジスタ (EESR) のビット7で報告される、マルチキャストアドレスフレーム受信ステータス情報を当該ディスクリプタに反映するか否かを指示します。本レジスタのビットは EtherC/E-DMAC ステータスレジスタ EESR[7] に対応し、ビット7に0を設定すると、受信ステータス (EESR[7]) は受信ディスクリプタの RFE ビットに反映されます。1を設定すると、該当する要因が発生してもディスクリプタに反映されません。LSIのリセット後は、ビット7は0に設定されています。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	7	6	5	4	3	2	1	0
	RMAFCE	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

- ・ ビット31～8: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット7: マルチキャストアドレスフレーム受信 (RMAF) ビットコピー指示 (RMAFCE)

ビット7	説明
RMAFCE	
0	RMAF ビットのステータスを受信ディスクリプタの RFS7 ビットに反映する。
1	該当する要因が発生しても受信ディスクリプタの RFS7 ビットに反映しない。

- ・ ビット6～0: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

## 10.2.9 ミスドフレームカウンタレジスタ (RMFCR)

本レジスタは、受信時に受信バッファに転送しきれずに廃棄したフレーム数を示す 16 ビットのカウンタです。受信 FIFO がオーバーフローすると、この FIFO 内にある受信フレームは廃棄されます。このときに廃棄するフレームの数をカウントアップします。本レジスタの値が、H'FFFF (65535) になるとカウントアップを停止します。カウンタの値は、本レジスタを読むと 0 にクリアされます。本レジスタへの書き込みは、他に影響を与えません。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	15	14	13	12	11	10	9	8
	MFC15	MFC14	MFC13	MFC12	MFC11	MFC10	MFC9	MFC8
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット:	7	6	5	4	3	2	1	0
	MFC7	MFC6	MFC5	MFC4	MFC3	MFC2	MFC1	MFC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

## 10.2.10 送信 FIFO しきい値指定レジスタ (TFTR)

本レジスタは、最初の送信を開始するまでの送信 FIFO のしきい値を指定します。実際のしきい値は、設定した 4 倍の値に相当します。EtherC は送信 FIFO 内のデータ数が本レジスタで指定されたバイト数を超えるか、送信 FIFO が満杯、もしくは 1 フレーム分の書き込みが行われると送信を開始します。

【注】 本レジスタの設定は、送信停止状態で行ってください。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	TFT10	TFT9	TFT8
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0
	TFT7	TFT6	TFT5	TFT4	TFT3	TFT2	TFT1	TFT0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

・ ビット 31 ~ 11 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 10 ~ 0 : 送信 FIFO しきい値 10 ~ 0 (TFT10 ~ TFT0)

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

TFT10 ~ 0	しきい値
H'00	ストア&フォワードモード (1 フレーム分の書き込み、または送信 FIFO がフルになると送信開始) (初期値)
H'01	4 バイト
H'02	8 バイト
:	:
H'1F	124 バイト
H'20	128 バイト
:	:
H'3F	252 バイト
H'40	256 バイト
:	:
H'7F	508 バイト
H'80	512 バイト

【注】 送信 FIFO のしきい値を 256 バイト以上に設定する場合は、必ず FIFO 容量を 512 バイトに設定してください。

【制限事項】

伝送速度 10Mbps (TX-CLK クロック入力周波数 = 2.5MHz) 時、TFTR の送信 FIFO しきい値 10 ~ 0 ビット (TFT10 ~ TFT0) に H'001 (4 バイト) から H'00C (48 バイト) のいずれかを設定した場合、イーサネットコントローラ (EtherC) の送信動作が行われない場合があります。

従って、伝送速度 10Mbps 時は、TFTR : TFT10 ~ TFT0 = H'000 (ストア&フォワードモード) または、TFTR : TFT10 ~ TFT0 = H'00D (52 バイト) 以上の値でご使用ください。(次頁参照)

なお、伝送速度 100Mbps (TX-CLK クロック入力周波数 = 25MHz) の場合は、本制限はありません。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

TFT10~0	しきい値
H'000	ストア&フォワードモード (初期値)
H'001	4バイト
H'002	8バイト
H'003	12バイト
H'004	16バイト
H'005	20バイト
H'006	24バイト
H'007	28バイト
H'008	32バイト
H'009	36バイト
H'00A	40バイト
H'00B	44バイト
H'00C	48バイト
H'00D	52バイト
H'00E	56バイト
H'00F	60バイト
.	.
H'01F	124バイト
H'020	128バイト
.	.
.	.
H'03F	252バイト
H'040	256バイト
.	.
.	.
H'07F	508バイト
H'080	512バイト

} 伝送速度10Mbps時  
不具合が発生する  
設定範囲

## 10.2.11 FIFO 容量指定レジスタ (FDR)

本レジスタは、送信および受信 FIFO の容量を指定します。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	TFD
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RFD
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

- ・ ビット 31 ~ 9 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 8 : 送信 FIFO 容量 (TFD)  
最大 512 バイトある送信 FIFO の容量を 256 バイトあるいは 512 バイトのいずれかで指定します。送受信開始後は設定値を変更することを禁止します。

ビット 8	説 明
TFD	
0	256 バイト (初期値)
1	512 バイト

- ・ ビット 7 ~ 1 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 0 : 受信 FIFO 容量 (RFD)  
最大 512 バイトある受信 FIFO の容量を 256 バイトあるいは 512 バイトのいずれかを指定します。送受信開始後は設定値を変更することを禁止します。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

ビット 0	説 明	
RFD		
0	256 バイト	(初期値)
1	512 バイト	

### 10.2.12 受信方式制御レジスタ (RCR)

本レジスタは、フレームを受信時の ECMR 内にある RE ビットの制御の方法を指定します。

【注】 本レジスタの設定は、受信停止状態で行ってください。

ビット :	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RNC
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R/W

- ・ ビット 31 ~ 1 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 0 : 受信許可制御 (RNC)

ビット 0	説 明	
RNC		
0	1つのフレームを受信完了すると、E-DMAC は受信ステータスをディスクリプタに書き込み EDRRR の RR ビットをクリアします。 (初期値)	
1	1つのフレームを受信完了すると、E-DMAC は受信ステータスをディスクリプタに書き込みます。さらに E-DMAC は次のディスクリプタを読み込み、次のフレームの受信に備えます。*	

【注】 \* 通常は、本設定により連続したフレーム受信を継続します。

## 10.2.13 E-DMAC 動作制御レジスタ (EDOCR)

本レジスタは、E-DMAC の動作状態における制御方法を指定します。

ビット:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	-	-	-	-	-	-	-	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	-	-	-	-	FEC	AEC	EDH	-
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R

- ビット 31 ~ 4 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 3 : FIFO エラー制御 (FEC)

送信 FIFO におけるアンダフローあるいは受信 FIFO におけるオーバフロー発生時の E-DMAC の動作を指定します。

ビット 3	説明
FEC	
0	アンダフローあるいはオーバフローが発生しても E-DMAC の動作を継続します。 (初期値)
1	アンダフローあるいはオーバフローが発生すると E-DMAC の動作を停止します。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

- ビット2: アドレスエラー制御 (AEC)

E-DMAC が転送しようとしたメモリアドレスが不正であったことを示します。

ビット2	説明
AEC	
0	不正なメモリアドレスを検出していない(正常動作)。 (初期値)
1	不正なメモリアドレスを検出した。0を書き込むことによってクリアすることができません。

【参考】 本エラーは、E-DMAC が使用するディスクリプタ内のメモリアドレス設定が不正である場合に発生します。

- ビット1: E-DMAC 停止 (EDH)

本 LSI の NMI 入力端子がアサートされると E-DMAC は動作を停止します。

ビット1	説明
EDH	
0	E-DMAC は正常に動作中です。 (初期値)
1	NMI 端子がアサートされたため E-DMAC の動作を停止します。0を書き込むことによって E-DMAC は動作を再開します。

- ビット0: 予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

## 10.2.14 受信バッファライトアドレスレジスタ (RBWAR)

E-DMAC が受信バッファにデータを書き込む際、受信バッファ内で書き込みの対象となるバッファアドレスを格納するためのレジスタです。本レジスタに表示されるアドレスをモニタすることにより、E-DMAC が受信バッファ内のどの辺のアドレスに対し処理を実行しているかを認識することができます。

ビット:	31	30	29	28	27	26	25	24
	RBWA31	RBWA30	RBWA29	RBWA28	RBWA27	RBWA26	RBWA25	RBWA24
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	RBWA23	RBWA22	RBWA21	RBWA20	RBWA19	RBWA18	RBWA17	RBWA16
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	RBWA15	RBWA14	RBWA13	RBWA12	RBWA11	RBWA10	RBWA9	RBWA8
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	RBWA7	RBWA6	RBWA5	RBWA4	RBWA3	RBWA2	RBWA1	RBWA0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

- ・ ビット 31 ~ 0: 受信バッファライトアドレス 31 ~ 0 (RBWA31 ~ RBWA0)  
本ビットは、読み出し専用です。書き込みを禁止します。

【注】 E-DMAC が実行しているバッファライト処理とレジスタの読み出し値が一致してない場合があります。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

### 10.2.15 受信ディスクリプタフェッチアドレスレジスタ (RDFAR)

E-DMAC が受信ディスクリプタからディスクリプタ情報をフェッチする際に必要となるディスクリプタ先頭アドレスを格納するためのレジスタです。本レジスタに表示されるアドレスをモニタすることにより、E-DMAC がどの辺の受信ディスクリプタ情報をもとに処理を実行しているか認識することができます。

ビット :	31	30	29	28	27	26	25	24
	RDFA31	RDFA30	RDFA29	RDFA28	RDFA27	RDFA26	RDFA25	RDFA24
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	23	22	21	20	19	18	17	16
	RDFA23	RDFA22	RDFA21	RDFA20	RDFA19	RDFA18	RDFA17	RDFA16
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	15	14	13	12	11	10	9	8
	RDFA15	RDFA14	RDFA13	RDFA12	RDFA11	RDFA10	RDFA9	RDFA8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	7	6	5	4	3	2	1	0
	RDFA7	RDFA6	RDFA5	RDFA4	RDFA3	RDFA2	RDFA1	RDFA0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R

- ・ ビット 31 ~ 0 : 受信ディスクリプタフェッチアドレス 31 ~ 0 (RDFA31 ~ RDFA0)  
本ビットは、読み出し専用です。書き込みを禁止します。

【注】 E-DMAC が実行しているディスクリプタフェッチ処理とレジスタの読み出し値が一致してない場合があります。

## 10.2.16 送信バッファリードアドレスレジスタ (TBRAR)

E-DMAC が送信バッファからデータを読み出す際、送信バッファ内で読み出しの対象となるバッファアドレスを格納するためのレジスタです。本レジスタに表示されるアドレスをモニタすることにより、E-DMAC が送信バッファ内のどの辺のアドレスに対し処理を実行しているかを認識することができます。

ビット:	31	30	29	28	27	26	25	24
	TBRA31	TBRA30	TBRA29	TBRA28	TBRA27	TBRA26	TBRA25	TBRA24
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	23	22	21	20	19	18	17	16
	TBRA23	TBRA22	TBRA21	TBRA20	TBRA19	TBRA18	TBRA17	TBRA16
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8
	TBRA15	TBRA14	TBRA13	TBRA12	TBRA11	TBRA10	TBRA9	TBRA8
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
ビット:	7	6	5	4	3	2	1	0
	TBRA7	TBRA6	TBRA5	TBRA4	TBRA3	TBRA2	TBRA1	TBRA0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

- ・ ビット 31 ~ 0: 送信バッファリードアドレス 31 ~ 0 (TBRA31 ~ TBRA0)  
本ビットは、読み出し専用です。書き込みを禁止します。

【注】 E-DMAC が実行しているバッファリード処理とレジスタの読み出し値が一致してない場合があります。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

### 10.2.17 送信ディスクリプタフェッチアドレスレジスタ (TDFAR)

E-DMAC が送信ディスクリプタからディスクリプタ情報をフェッチする際に必要となるディスクリプタ先頭アドレスを格納するためのレジスタです。本レジスタに表示されるアドレスをモニタすることにより、E-DMAC がどの辺の送信ディスクリプタ情報をもとに処理を実行しているか認識することができます。

ビット :	31	30	29	28	27	26	25	24
	TDFA31	TDFA30	TDFA29	TDFA28	TDFA27	TDFA26	TDFA25	TDFA24
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	23	22	21	20	19	18	17	16
	TDFA23	TDFA22	TDFA21	TDFA20	TDFA19	TDFA18	TDFA17	TDFA16
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	15	14	13	12	11	10	9	8
	TDFA15	TDFA14	TDFA13	TDFA12	TDFA11	TDFA10	TDFA9	TDFA8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	7	6	5	4	3	2	1	0
	TDFA7	TDFA6	TDFA5	TDFA4	TDFA3	TDFA2	TDFA1	TDFA0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R

- ・ ビット 31 ~ 0 : 送信ディスクリプタフェッチアドレス 31 ~ 0 (TDFA31 ~ TDFA0)  
本ビットは、読み出し専用です。書き込みを禁止します。

【注】 E-DMAC が実行しているディスクリプタフェッチ処理とレジスタの読み出し値が一致してない場合があります。

### 10.3 動作説明

E-DMAC は EtherC と接続され、送受信データを CPU の介在なく、効率的な転送をメモリ (バッファ) との間で行います。E-DMAC は、各バッファと対応したディスクリプタと呼ぶバッファポインタなどを格納した制御情報を自ら読み込みます。この制御情報に従って送信データを送信バッファから読み込み、受信データは受信バッファに書き込みます。このディスクリプタを複数個連続して配置 (ディスクリプタリスト) することで、送信ならびに受信を連続して実行することができます。

#### 10.3.1 ディスクリプタリストとデータバッファ

通信プログラムは送受信の開始に先立って、メモリ上に送信および受信の各ディスクリプタリストを作成します。そしてこのリストの先頭アドレスを、送信 / 受信ディスクリプタリスト先頭アドレスレジスタに設定します。

##### (1) 送信ディスクリプタ

図 10.2 に送信ディスクリプタと送信バッファの関係を示します。本ディスクリプタの指示により、送信フレームと送信バッファの構成を 1 フレーム / 1 バッファあるいは 1 フレーム / マルチバッファのように関連づけることが可能です。

- 【注】
1. ディスクリプタの開始アドレスの設定は E-DMAC モードレジスタ (EDMR) で設定したディスクリプタ長に従ったアドレス境界に設定してください。
  2. 送信バッファの開始アドレスの設定は、ロングワードを境界として設定してください。ただし、SDRAM 接続時は、16 バイトを境界として設定してください。

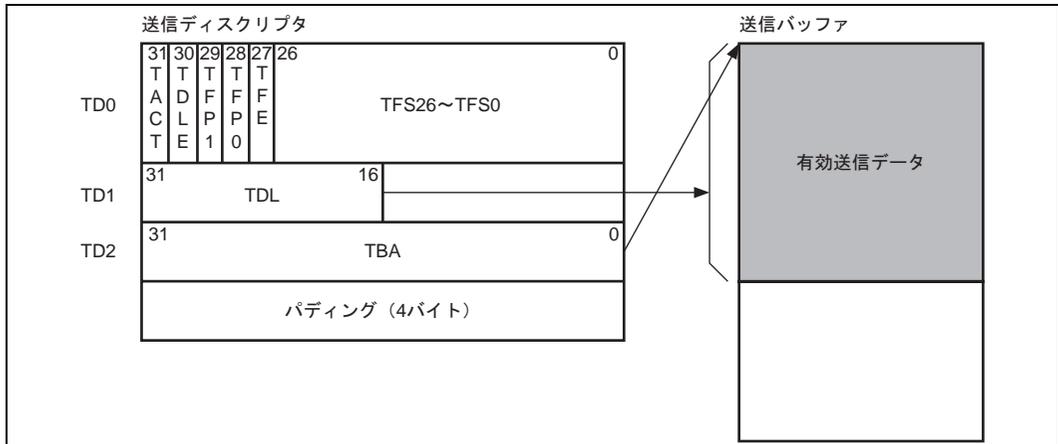


図 10.2 送信ディスクリプタと送信バッファの関係

##### (a) 送信ディスクリプタ 0 (TD0)

TD0 は、送信フレームのステータスを示します。CPU と E-DMAC はこの TD0 によってフレーム送信状態を連絡します。

- ・ ビット 31 : 送信ディスクリプタ有効 (TACT)

当該ディスクリプタが有効であることを示します。CPU は送信データを送信バッファに転送後、本ビットをセットします。また E-DMAC は、フレームの転送を完了、あるいは送信を中断した場合にリセットします。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

ビット 31	説 明
TACT	
0	送信ディスクリプタが無効であることを示します。CPU により送信バッファに有効データを書き込んでいない、または E-DMAC のフレーム転送処理終了によるライトバック動作で、本ビットがリセットされたことを示します (送信完了あるいは中断)。本状態が、E-DMAC のディスクリプタリードにより認識された場合、E-DMAC は送信処理を終了します。送信動作の継続はできません。再起動が必要となります。
1	送信ディスクリプタが有効であることを示します。CPU により送信バッファに有効データを書き込まれ、まだフレーム転送処理を行っていない、あるいはフレーム転送中であることを示します。本状態が、E-DMAC のディスクリプタリードにより認識された場合、E-DMAC は送信動作を継続します。

・ ビット 30 : 送信ディスクリプタリスト最終 (TDLE)

当該ディスクリプタが、送信ディスクリプタリストの最終であることを示します。E-DMAC は当該バッファの転送を終了後、先頭のディスクリプタを参照します。本指示によって送信ディスクリプタはリング構成となります。

ビット 30	説 明
TDLE	
0	送信ディスクリプタリストは最後でない
1	送信ディスクリプタリストは最後

・ ビット 29 ~ 28 : 送信フレーム内位置 1、0 (TFP1、TFP0)

本 2 ビットによって送信バッファと送信フレームの関連づけを行います。

ビット 29	ビット 28	説明
TFP1	TFP0	
0	0	本ディスクリプタで指示する送信バッファのフレーム送信を継続する (フレームを完結しない)
	1	本ディスクリプタで指示する送信バッファはフレームの最後を含む (フレームを完結する)
1	0	本ディスクリプタで指示する送信バッファはフレームの先頭である (フレームを完結しない)
	1	本ディスクリプタで指示する送信バッファの内容が 1 フレームに相当する (1 フレーム / 1 バッファ)

【注】 前後のディスクリプタにおいて、本ビットおよび TDL ビットの設定は論理的に正しい関係を維持してください。

・ ビット 27 : 送信フレームエラー (TFE)

ビット 26 ~ 0 に示す送信フレームステータスのいずれかのビットがセットされていることを示します。

ビット 27	説 明
TFE	
0	送信時にエラーがなかった
1	送信中に何らかのエラーがあった (ビット 26 ~ 0 を参照)

- ・ ビット 26 ~ 0 : 送信フレームステータス 26 ~ 0 (TFS26 ~ TFS0)  
フレーム送信中のエラーステータスを表示します。

- ・ TFS26 ~ 9 : 予約ビット
- ・ TFS8 : 送信アボート検出

【注】 送信フレームステータス 4 ~ 0 (TFS4 ~ 0) のいずれかのビットがセットされると本ビットがセットされます。本ビットがセットされた場合は、ビット 27 に示す送信フレームエラー (TFE) がセットされます。

- ・ TFS7 ~ 5 : 予約ビット
- ・ TFS4 : フレーム長異常検出 (EESR の ITF ビットに相当)
- ・ TFS3 : ノーキャリア検出 (EESR の CND ビットに相当)
- ・ TFS2 : キャリア消失検出 (EESR の DLC ビットに相当)
- ・ TFS1 : 送信中の遅延衝突検出 (EESR の CD ビットに相当)
- ・ TFS0 : 送信リトライオーバ (EESR の TRO ビットに相当)

#### (b) 送信ディスクリプタ 1 (TD1)

最大 64k バイトの送信バッファ長を指定します。

- ・ ビット 31 ~ 16 : 送信バッファデータ長 (TDL)  
当該送信バッファ内の有効転送バイト長を示します。

【注】 1 フレーム / マルチバッファ方式 (TD0.TFP=10 あるいは 00) を指定する場合、先頭と途中のディスクリプタ内で指定する転送バイト長はロングワード境界 (ビット 17 と 16 は 00) としてください。

- ・ ビット 15 ~ 0 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

#### (c) 送信ディスクリプタ 2 (TD2)

32 ビット幅の当該送信バッファの先頭アドレスを示します。

【注】 送信バッファの開始アドレスの設定は、ロングワードを境界として設定してください。ただし、SDRAM 接続時は、16 バイトを境界として設定してください。

- ・ ビット 31 ~ 0 : 送信バッファ先頭アドレス (TBA)

#### (2) 受信ディスクリプタ

図 10.3 に受信ディスクリプタと受信バッファの関係を示します。フレームの受信時、E-DMAC は受信フレーム長に関係なく、受信バッファの 16 バイト境界までデータの書き換えを行います。最終的に実際の受信フレーム長は、ディスクリプタ内にある RD1 の下位 16 ビットに報告されます。受信バッファへのデータ転送は、受信した 1 フレームの大きさにより、1 フレーム / 1 バッファあるいは 1 フレーム / マルチバッファ構成となるように、E-DMAC が自動的に行います。

【注】 1. ディスクリプタの開始アドレスの設定は、E-DMAC モードレジスタ (EDMR) で設定したディスクリプタ長に従ったアドレス境界として設定してください。  
2. 受信バッファの開始アドレスの設定は、ロングワードを境界として設定してください。ただし、SDRAM 接続時は、16 バイトを境界として設定してください。

受信バッファの大きさは 16 バイト境界となるように設定してください。

例 : H'0500 (=1536 バイト)

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

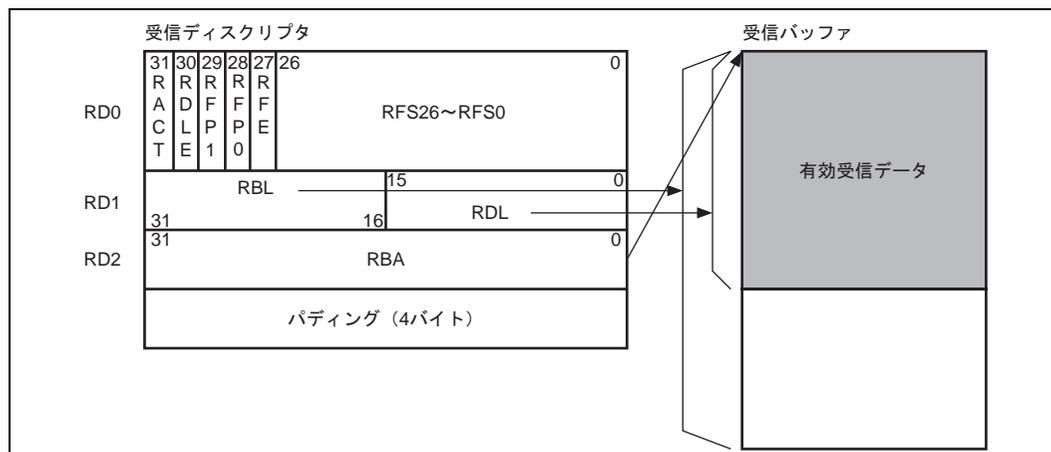


図 10.3 受信ディスクリプタと受信バッファの関係

### (a) 受信ディスクリプタ 0 (RD0)

RD0 は、受信フレームのステータスを示します。CPU と E-DMAC はこの RD0 によってフレーム送信状態を連絡します。

- ・ ビット 31：受信ディスクリプタ有効 (RACT)

当該ディスクリプタが有効であることを示します。E-DMAC は受信データを受信バッファに転送後、本ビットをリセットします。また CPU は、受信フレームの処理を完了した場合に受信準備のためセットします。

ビット 31 RACT	説明
0	受信ディスクリプタが無効であることを示します。受信バッファの準備ができていない (E-DMAC によるアクセス禁止) または E-DMAC のフレーム転送終了によるライトバック動作で、本ビットがリセットされたことを示します (受信完了あるいは中断)。本状態が、E-DMAC のディスクリプタリードにより認識された場合、E-DMAC は受信処理を終了します。受信動作の継続はできません。RACT = 1 に設定し受信起動をかけることで、受信を再開することができます。
1	受信ディスクリプタが有効であることを示します。受信バッファの準備完了 (アクセス許可) がかつ FIFO からのフレーム転送処理を行っていない、あるいはフレーム転送中であることを示します。本状態が、E-DMAC のディスクリプタリードにより認識された場合、E-DMAC は受信動作を継続します。

- ・ ビット 30：受信ディスクリプタリスト最終 (RDLE)

当該ディスクリプタが、受信ディスクリプタリストの最終であることを示します。E-DMAC は当該バッファの転送を終了後、先頭の受信ディスクリプタを参照します。本指示によって受信ディスクリプタはリング構成となります。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

ビット 30	説明
RDLE	
0	受信ディスクリプタリストの最後ではない
1	受信ディスクリプタリストの最後

- ビット 29、28 : 受信フレーム内位置 1、0 (RFP1、RFP0)  
本 2 ビットによって受信バッファと受信フレームの関連づけを行う。

ビット 29	ビット 28	説明
RFP1	RFP0	
0	0	本ディスクリプタで指示する受信バッファのフレーム受信を継続する (フレームを完結しない)
	1	本ディスクリプタで指示する受信バッファはフレームの最後を含む (フレームを完結する)
1	0	本ディスクリプタで指示する受信バッファはフレームの先頭である (フレームを完結しない)
	1	本ディスクリプタで指示する受信バッファの内容が 1 フレームに相当する (1 フレーム / 1 バッファ)

- ビット 27 : 受信フレームエラー (RFE)  
ビット 26 ~ 0 に示す受信フレームステータスのいずれかのビットがセットされていることを示します。また受信フレームステータスのうちマルチキャストアドレスフレーム受信情報を本ビットに反映するか否かは、送受信ステータスコピー指示レジスタ (TRSCER) によって設定されます。

ビット 27	説明
RFE	
0	受信時にエラーがなかった
1	受信中に何らかのエラーがあった (ビット 26 ~ 0 を参照)

- ビット 26 ~ 0 : 受信フレームステータス 26 ~ 0 (RFS26 ~ RFS0)  
フレーム受信中のエラーステータスを表示します。
  - RFS26 ~ 10 : 予約ビット
  - RFS9 : 受信 FIFO オーバフロー (EESR の RFOF ビットに相当)
  - RFS8 : 受信アポートエラー

**【注】** 受信フレームステータス 9 (RFS9)、7 (RFS7)、4 ~ 0 (RFS4 ~ 0) のいずれかのビットがセットされると本ビットがセットされます。

本ビットがセットされた場合は、ビット 27 に示す受信フレームエラー (RFE) がセットされます。

- RFS7 : マルチキャストアドレスフレームを受信 (EESR の RMAF ビットに相当)
- RFS6 ~ 5 : 予約ビット
- RFS4 : 端数ビットフレーム受信エラー (EESR の RRF ビットに相当)
- RFS3 : ロングフレーム受信エラー (EESR の RTLF ビットに相当)

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

- RFS2 : ショートフレーム受信エラー (EESR の RTSF ビットに相当)
- RFS1 : PHY-LSI 受信エラー (EESR の PRE ビットに相当)
- RFS0 : 受信フレーム CRC エラー検出 (EESR の CERF ビットに相当)

### (b) 受信ディスクリプタ 1 (RD1)

最大 64k バイト以内の受信バッファ長を指定します。

- ビット 31 ~ 16 : 受信バッファ長 (RBL)

当該受信バッファ内の最大転送バイト長を示します。

【注】 転送バイト長は 16 バイト境界 (ビット 19 ~ 16 は 0) としてください。

1 フレーム / バッファのとき、受信フレーム長は、CRC データを除き 1,514 バイトが最大です。よって受信バッファ長の指定は、最大受信フレーム長に 16 バイト境界を考慮した値である 1,520 バイト (H'05F0) を設定します。

- ビット 15 ~ 0 : 受信データ長 (RDL)

受信バッファに格納された受信フレームのデータ長を示します。

【注】 受信バッファに転送される受信データには、フレームの最後にある CRC データ (4 バイト) が含まれません。また受信フレーム長は、この CRC データを含めない (有効データバイト) 語数が報告されます。

### (c) 受信ディスクリプタ 2 (RD2)

32 ビット幅の当該受信バッファの先頭アドレスを示します。

【注】 受信バッファの開始アドレスの設定は、ロングワードを境界として設定してください。ただし、SDRAM 接続時は、16 バイトを境界として設定してください。

- ビット 31 ~ 0 : 受信バッファ先頭アドレス (RBA)

## 10.3.2 送信機能

送信機能が有効で、E-DMAC 送信要求レジスタ (EDTRR) の送信要求ビット (TR) をセットすると、E-DMAC は送信ディスクリプタリストから前回使用したディスクリプタの次のディスクリプタ (初期状態では送信ディスクリプタ先頭アドレスレジスタ (TDLAR) で示すディスクリプタ) を読み込みます。読み込んだディスクリプタの TACT ビットが有効な場合、E-DMAC は TD2 で指定される送信バッファ先頭アドレスから順次送信フレームデータを読み出し EtherC に転送します。EtherC は送信フレームを作成し MII に向けて送信を開始します。ディスクリプタ内で指示されるバッファ長分の DMA 転送後、TFP の値によって以下のような処理を行います。

(a) TFP = 00 or 10 (フレーム継続) :

DMA 転送後、ディスクリプタのライトバック (TACT ビットのみ) を行います。

(b) TFP = 01 or 11 (フレーム終了) :

フレームの送信完了後、ディスクリプタのライトバック (TACT ビットおよびステータス) を行います。

読み込んだディスクリプタの TACT ビットが有効な間、E-DMAC ディスクリプタの読み込みとフレームの送信を継続します。TACT ビットが無効なディスクリプタを読み込むと、E-DMAC は送信要求レジスタ (EDTRR) の送信要求ビット (TR) をリセットして送信処理を完了します。

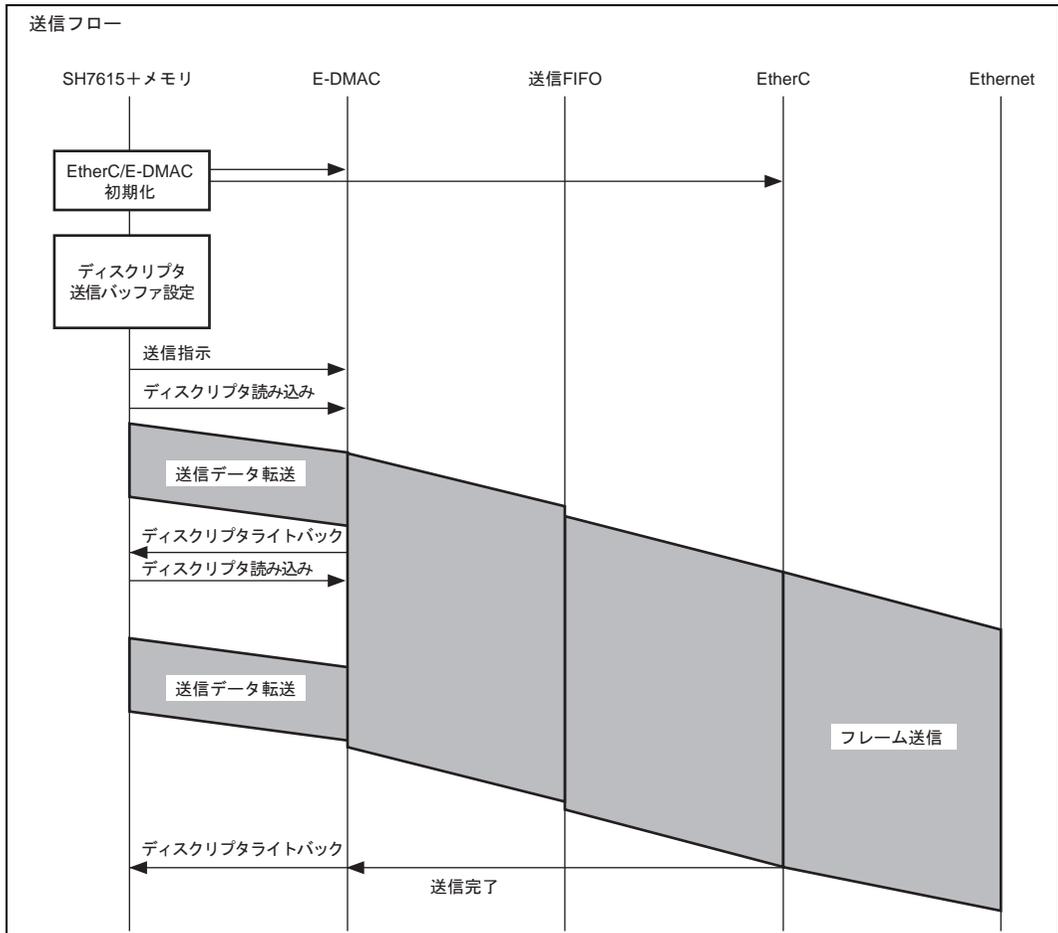


図 10.4 送信フローの例

### 10.3.3 受信機能

受信機能が有効で、CPU が E-DMAC 受信要求レジスタ (EDRRR) の受信要求ビット (RR) をセットすると、E-DMAC は受信ディスクリプタリストから前回使用したディスクリプタの次のディスクリプタ (初期状態では受信ディスクリプタ先頭アドレスレジスタ (RDLAR) で示すディスクリプタ) を読み込んだ後、受信待機状態となります。RACT ビットが有効で、かつ自局宛フレームを受信すると、RD2 で指定される受信バッファに転送します。受信したフレームのデータ長が、RD1 で与えられるバッファ長よりも大きい場合、E-DMAC はバッファが満了となった時点でディスクリプタにライトバック (RFP = 10 or 00) を行った後、次のディスクリプタを読み込みます。そして新たな RD2 によって指定される受信バッファに引き続きデータを転送します。フレームの受信が完了した場合、あるいは何らかのエラーでフレーム受信を中断すると、当該ディスクリプタにライトバック (RFP = 11 or 01) を行った後、受信処理を終了します。そして次のディスクリプタを読み込み受信待機状態となります。

## 10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

【注】 連続してフレームを受信するには、受信方式制御レジスタ (RCR) 内の受信コントロールビット (RNC) を 1 に設定してください。初期化後は、0 になっています。

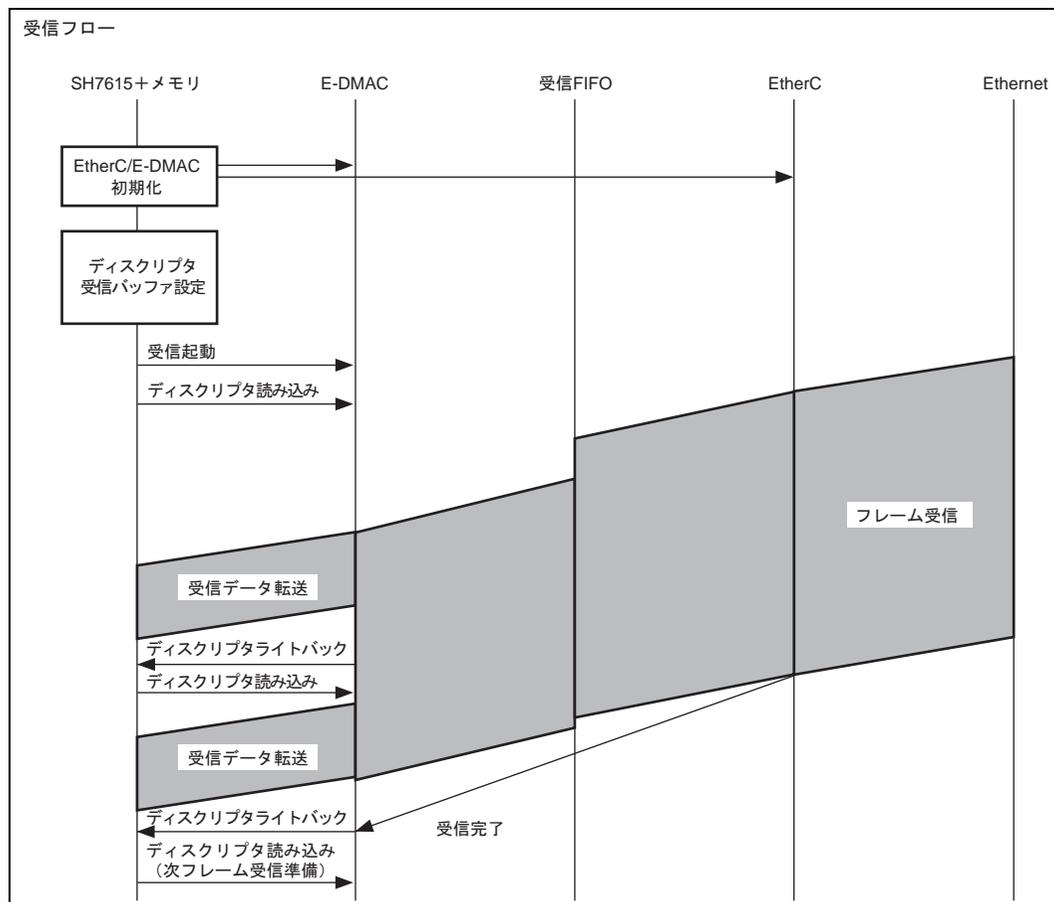


図 10.5 受信フローの例

### 10.3.4 マルチバッファフレームの送受信処理について

#### (1) マルチバッファフレームの送信処理

マルチバッファフレームの送信中にエラーが発生した場合には図 10.6 に示す処理を行います。

図中で送信ディスクリプタが無効 (TACT ビットが 0) である部分は、すでにバッファデータを正常に送信した部分を、送信ディスクリプタが有効 (TACT ビットが 1) である部分は、バッファデータが未送信であることを示します。送信ディスクリプタが有効 (TACT ビットが 1) である最初のディスクリプタ部分でフレーム送信エラーが発生した場合、即座に送信を停止して TACT ビットを 0 クリアします。その後、次のディスクリプタをリードし、送信フレーム内の位置を TFP1、TFP0 ビットをもとに判断していきます (継続 [00] または終了 [01])。継続ディスクリプタである場合は、TACT ビットを 0 クリアするのみで、すぐに次ディスクリプタのリードを行います。最終ディスクリプタである場合は、TACT ビットを 0 クリアするのみでなく、TFE および TFS ビットへのライトバックも同時に行います。エラー発生後から最終ディスクリプタへのライトバックまでの間、バッファ

上のデータは送信しません。EtherC/E-DMAC ステータス割込み許可レジスタ (EESIPR) でエラー割込みが許可されている場合は、最終ディスクリプタのライトバック直後に割込みが発生します。

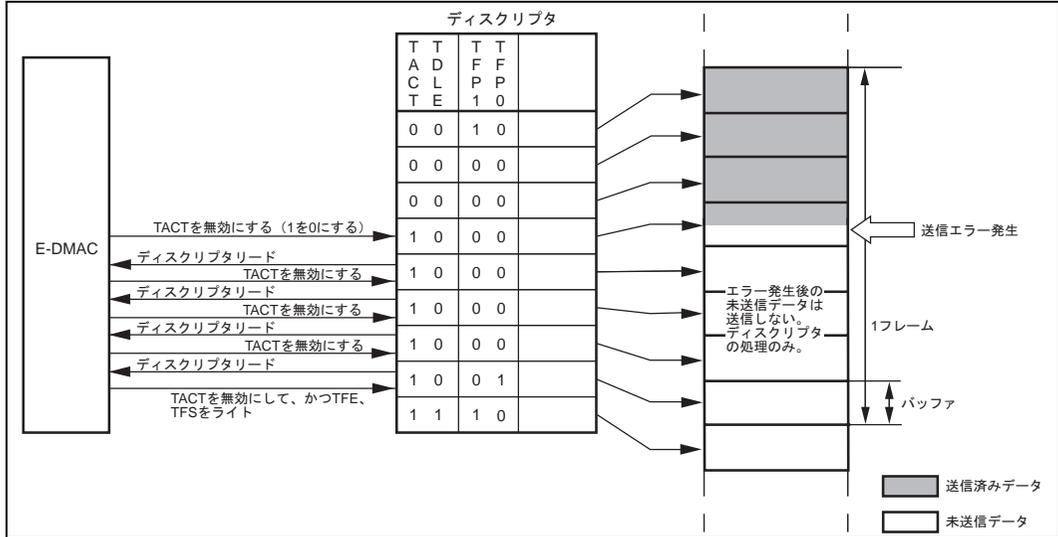


図 10.6 送信エラー発生後の E-DMAC 動作

(2) マルチバッファフレームの受信処理

マルチバッファフレームの受信中にエラーが発生した場合には図 10.7 に示す処理を行います。

図中で受信ディスクリプタが無効 (RACT ビットが 0) である部分は、すでにバッファデータを正常に受信した部分を、受信ディスクリプタが有効 (RACT ビットが 1) である部分は、未受信バッファであることを示します。図中で RACT ビットが 1 である最初のディスクリプタ部分でフレーム受信エラーが発生した場合、ディスクリプタにステータスのライトバックを行います。

EtherC/E-DMAC ステータス割込み許可レジスタ (EESIPR) でエラー割込みを許可している場合は、ライトバック直後に割込みが発生します。新しいフレームの受信要求がある場合には、エラーが発生したバッファの次のバッファから引き続き受信を行います。

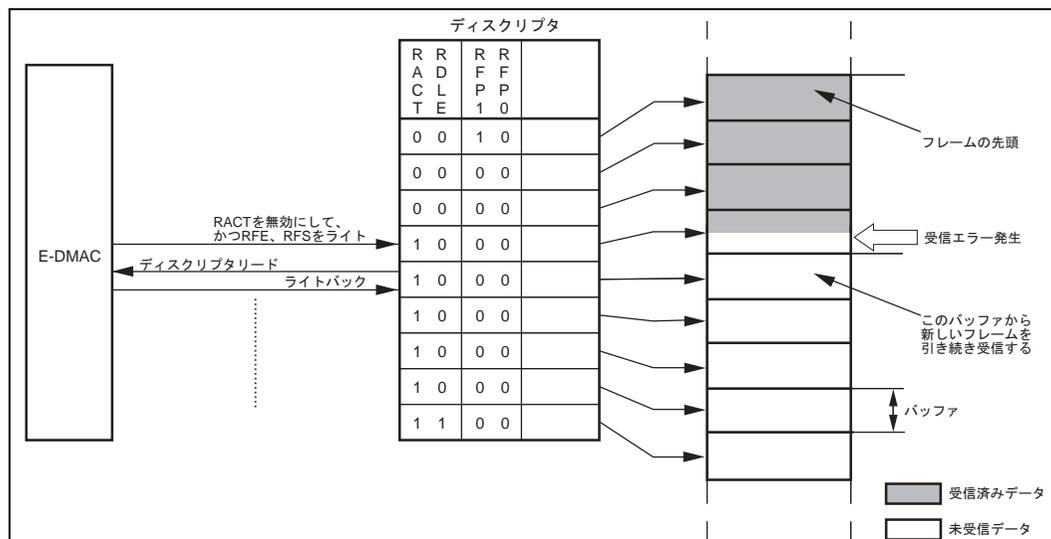


図 10.7 受信エラー発生後の E-DMAC 動作

## 10.4 使用上の注意事項

### 10.4.1 E-DMAC 送信要求レジスタ (EDTRR) および E-DMAC 受信要求レジスタ (EDRRR) 使用上の注意事項

#### 【不具合内容】

Ethernet 機能使用時において、E-DMAC 送信要求レジスタ (EDTRR) 送信要求ビット (TR) および E-DMAC 受信要求レジスタ (EDRRR) 受信要求ビット (RR) は、ディスクリプタ有効ビットが無効な場合、各々の要求ビットをクリアして E-DMAC の動作を停止しますが、E-DMAC による要求ビットのクリア動作とユーザファームウェアによる要求ビットのセット動作が同時に発生した場合、要求ビットが“1”にセットされているにもかかわらず送信および受信動作が開始されない場合があります。

#### 【発生条件】

ユーザファームウェアにおいて、要求ビットが“1”にもかかわらず、再度、要求ビットを“1”にセットしようとする場合がある場合。

#### 【回避方法】

E-DMAC による要求ビットのクリア動作とユーザファームウェアによる要求ビットのセット動作の同時発生を避けるために、E-DMAC による要求ビットのクリア動作を確認してから、ユーザファームウェアにより要求ビットをセットするようにしてください。

E-DMAC による要求ビットのクリア動作を確認する方法として、

#### (1) 送信要求ビット (TR) 確認方法

- a) 直接的には、E-DMAC 送信要求レジスタ (EDTRR) 送信要求ビット (TR) = 0 リードにより確認できません。

b) または、間接的には、割り込み発生後の EtherC/E-DMAC ステータスレジスタ (EESR) 送信ディスクリプタ枯渇ビット (TDE) =1 リードにより確認できます。

(2) 受信要求ビット (RR) 確認方法

a) 直接的には、E-DMAC 受信要求レジスタ (EDRRR) 受信要求ビット (RR) =0 リードにより確認できます。

b) または、間接的には、割り込み発生後の EtherC/E-DMAC ステータスレジスタ (EESR) 受信ディスクリプタ枯渇ビット (RDE) =1 リードにより確認できます。

10. イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)

---

---

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

### 11.1 概要

本 LSI は、2 チャンネルのダイレクトメモリアクセスコントローラ (DMAC) を内蔵しています。DMAC は、DACK (転送要求受け付け信号) 付き外部デバイス、外部メモリ、内蔵メモリ、メモリマップト外部デバイス、内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く) 間のデータ転送を、CPU に代わって高速に行うことができます。DMAC を使うと CPU の負担を減らすとともに本 LSI 全体の動作効率を上げることができます。

#### 11.1.1 特長

DMAC には、次の特長があります。

- チャンネル数 : 2 チャンネル
- アドレス空間 : アーキテクチャ上は 4GB
- データ転送単位が選択可能 :  
バイト、ワード (2 バイト)、ロングワード (4 バイト)、または 16 バイト単位 (16 バイト転送は、ロングワードの読み出しを 4 回実行した後に、ロングワードの書き込みを 4 回実行します)
- 最大転送回数 : 16,777,216 (16M) 回
- キャッシュヒット時は、CPU の命令処理と DMA 動作が並列実行可能
- シングルアドレスモード転送 :  
転送元か転送先の周辺デバイスを DACK 信号でアクセス (選択) し、もう一方をアドレスでアクセスします。1 バスサイクルで 1 転送単位のデータを転送します。
  - 転送可デバイス :  
DACK 付き外部デバイスとメモリマップト外部デバイス (外部メモリ含む)
- デュアルアドレスモード転送 :  
転送元と転送先の両方をアドレスでアクセスします。2 バスサイクルで 1 転送単位のデータを転送します。
  - 転送可デバイス : 外部メモリ同士  
外部メモリとメモリマップト外部デバイス  
メモリマップト外部デバイス同士  
外部メモリと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く)  
メモリマップト外部デバイスと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く)

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

内蔵周辺モジュール同士 (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く) (転送元または転送先である周辺モジュールのレジスタで許されるアクセスサイズ)

内蔵メモリとメモリマップト外部デバイス

内蔵メモリ同士

内蔵メモリと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く)

内蔵メモリと外部メモリ

- 転送要求
  - 外部リクエスト (DREQn 端子による。DREQn はエッジで検出するかレベルで検出するかまたローアクティブかハイアクティブかを選択可能)
  - 内蔵周辺モジュールリクエスト (FIFO 付きシリアルコミュニケーションインタフェース (SCIF)、16 ビットタイマパルスユニット (TPU)、シリアル I/O (SIO))
  - オートリクエスト (転送要求を DMAC 内部で自動的に発生)
- バスモードが選択可能
  - サイクルスチールモード
  - バーストモード
- チャネル優先順位が選択可能
  - 固定モード
  - ラウンドロビンモード
- データ転送終了時に CPU へ割り込み要求可能

11. 1. 2 ブロック図

DMACのブロック図を図 11.1 に示します。

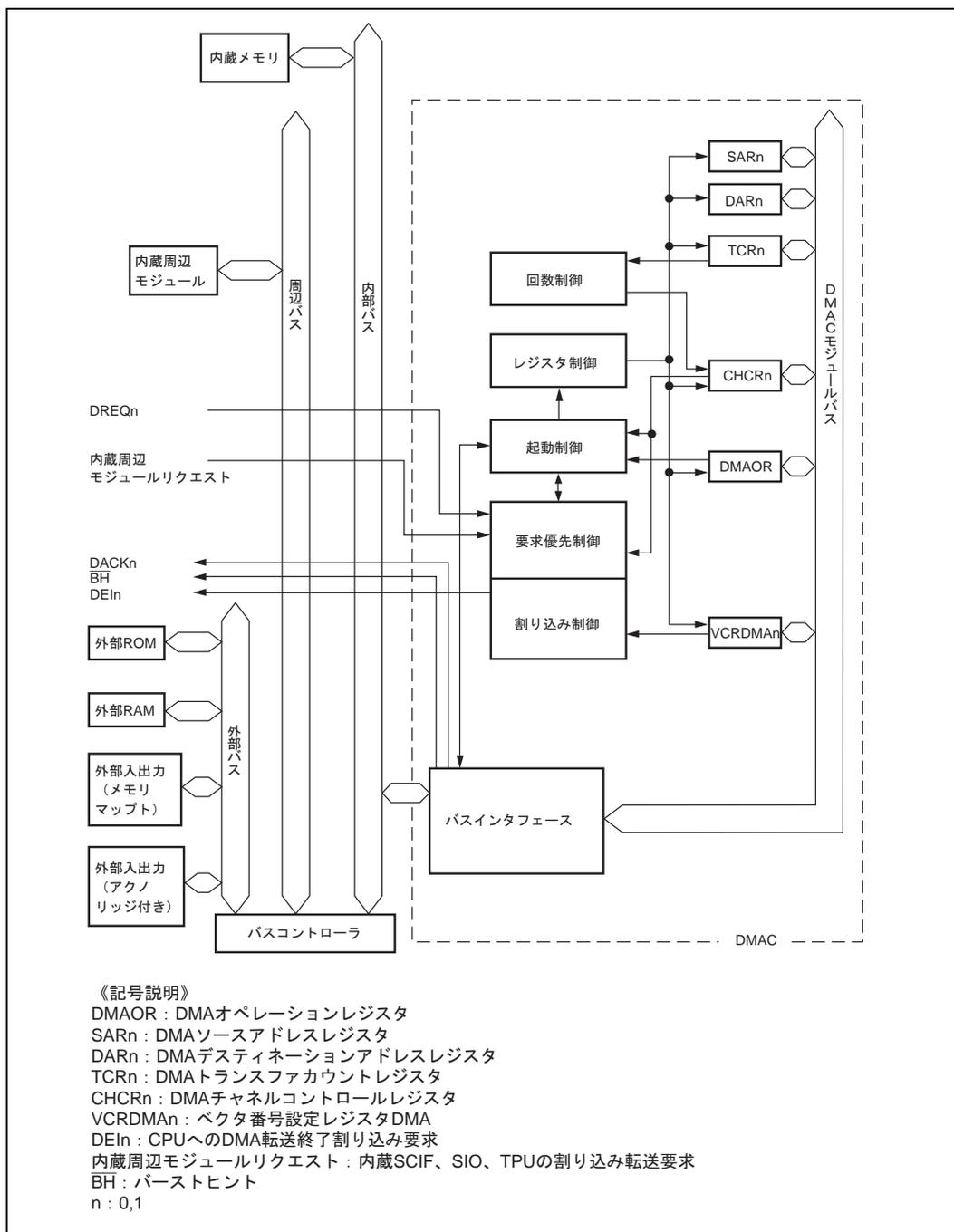


図 11.1 DMACのブロック図

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### 11.1.3 端子構成

DMAC の端子を表 11.1 に示します。

表 11.1 端子構成

チャンネル	名称	略称	入出力	機能
0	DMA 転送要求	DREQ0	入力	外部デバイスからチャンネル 0 への DMA 転送要求入力
	DMA 転送要求受け付け	DACK0	出力	チャンネル 0 から外部デバイスへの DMA 転送要求受け付け出力
1	DMA 転送要求	DREQ1	入力	外部デバイスからチャンネル 1 への DMA 転送要求入力
	DMA 転送要求受け付け	DACK1	出力	チャンネル 1 から外部デバイスへの DMA 転送要求受け付け出力
共通	バーストヒント	BH	出力	16 バイト転送モード時のバースト転送

### 11.1.4 レジスタ構成

DMAC のレジスタを表 11.2 に示します。DMAC には全部で 13 本のレジスタがあります。各チャンネルに 6 本の制御レジスタがあり、すべてのチャンネルに共通な制御レジスタが 1 本あります。

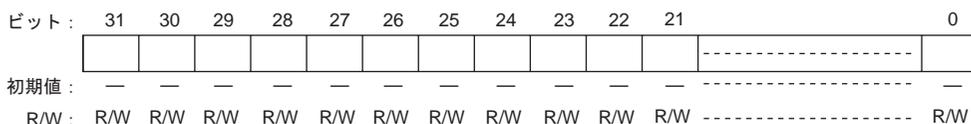
表 11.2 レジスタ構成

チャンネル	名称	略称	R/W	初期値	アドレス	アクセスサイズ <sup>*3</sup>
0	DMA ソースアドレスレジスタ 0	SAR0	R/W	不定	H'FFFFFF80	32
	DMA デスティネーションアドレスレジスタ 0	DAR0	R/W	不定	H'FFFFFF84	32
	DMA トランスファカウントレジスタ 0	TCR0	R/W	不定	H'FFFFFF88	32
	DMA チャンネルコントロールレジスタ 0	CHCR0	R/(W) <sup>*1</sup>	H'00000000	H'FFFFFF8C	32
	ベクタ番号設定レジスタ DMA0	VCRDMA0	R/W	不定	H'FFFFFFA0	32
	DMA 要求 / 応答選択制御レジスタ 0	DRCR0	R/W	H'00	H'FFFFFFE71	8 <sup>*3</sup>
1	DMA ソースアドレスレジスタ 1	SAR1	R/W	不定	H'FFFFFF90	32
	DMA デスティネーションアドレスレジスタ 1	DAR1	R/W	不定	H'FFFFFF94	32
	DMA トランスファカウントレジスタ 1	TCR1	R/W	不定	H'FFFFFF98	32
	DMA チャンネルコントロールレジスタ 1	CHCR1	R/(W) <sup>*1</sup>	H'00000000	H'FFFFFF9C	32
	ベクタ番号設定レジスタ DMA1	VCRDMA1	R/W	不定	H'FFFFFFA8	32
	DMA 要求 / 応答選択制御レジスタ 1	DRCR1	R/W	H'00	H'FFFFFFE72	8 <sup>*3</sup>
共通	DMA オペレーションレジスタ	DMAOR	R/(W) <sup>*2</sup>	H'00000000	H'FFFFFFB0	32

- 【注】
- \*1 CHCR0、CHCR1 のビット 1 は、フラグをクリアするために 1 リード後の 0 ライトのみ可能です。
  - \*2 DMAOR のビット 1 と 2 は、フラグをクリアするために 1 リード後の 0 ライトのみ可能です。
  - \*3 DRCR0、1 のアクセスは、バイト単位、それ以外のレジスタのアクセスはすべてロングワード単位で行ってください。

## 11.2 レジスタの説明

### 11.2.1 DMA ソースアドレスレジスタ 0、1 (SAR0、SAR1)



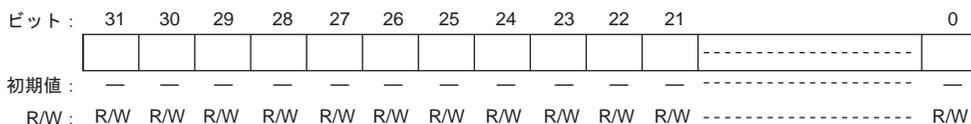
DMA ソースアドレスレジスタ 0、1 (SAR0、SAR1) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送元のアドレスを指定します。DMA 転送中は、次の転送元アドレスを示しています (シングルアドレスモードにおいて、DACK 付きの外部デバイスから、メモリマップト外部デバイスまたは外部メモリへの転送をする場合には SAR は無視されます)。

16 バイト単位で転送するときは、ソースアドレスは必ず 16 バイト境界 (16n 番地) に値を設定してください。間違えた値を設定したときは、動作は保証しません。

なお、16 バイト単位での転送は、オートリクエストモード時、外部リクエストモード・エッジ検出時のみ設定可能です。

リセット、スタンバイモード、およびモジュールスタンバイ機能使用時には値を保持します。

### 11.2.2 DMA デスティネーションアドレスレジスタ 0、1 (DAR0、DAR1)



DMA デスティネーションアドレスレジスタ 0、1 (DAR0、DAR1) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送先のアドレスを指定します。DMA 転送中は、次の転送先アドレスを示しています (シングルアドレスモードにおいて、メモリマップト外部デバイスまたは外部メモリから、DACK 付きの外部デバイスへの転送をする場合には DAR は無視されます)。

16 バイト単位で転送するときは、ソースアドレスは必ず 16 バイト境界 (16n 番地) に値を設定してください。間違えた値を設定したときは、動作は保証しません。

なお、16 バイト単位での転送は、オートリクエストモード時、外部リクエストモード・エッジ検出時のみ設定可能です。

リセット、スタンバイモード、およびモジュールスタンバイ機能使用時には値を保持します。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### 11.2.3 DMA トランスファカウントレジスタ 0、1 (TCR0、TCR1)

ビット:	31	30	29	28	27	26	25	24	23	22	21	-----	2	1	0
	—	—	—	—	—	—	—	—				-----			
初期値:	0	0	0	0	0	0	0	0	—	—	—	-----	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	-----	R/W	R/W	R/W

DMA トランスファカウントレジスタ 0、1 (TCR0、TCR1) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送回数を指定します。ただし、32 ビット中の下位の 24 ビットが有効です。上位 8 ビットを含めて、32 ビットで値を書き込みます。転送回数は、設定値が H'00000001 のときは 1 回、H'00FFFFFF のときは 16,777,215 回で、H'00000000 のときは 16,777,216 回 (最大転送回数) になります。DMA 転送中は、残りの転送回数を示しています。

上位 8 ビットへの書き込みは初期値を設定してください。読み出すと 0 が読み出されます。

リセット、スタンバイモード、およびモジュールスタンバイ機能使用時には値を保持します。

16 バイト転送時には、必ず転送回数 × 4 を設定してください。間違えた値を設定した場合の動作は保証しません。

### 11.2.4 DMA チャネルコントロールレジスタ 0、1 (CHCR0、CHCR1)

ビット:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM1	DM0	SM1	SM0	TS1	TS0	AR	AM	AL	DS	DL	TB	TA	IE	TE	DE
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/(W)*	R/W													

【注】 \* フラグをクリアするために 0 のみ書き込むことができます。

DMA チャネルコントロールレジスタ 0、1 (CHCR0、CHCR1) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送モードを制御します。また、DMA 転送の状態 (ステータス) も示します。ただし、32 ビット中の下位の 16 ビットが有効です。上位 16 ビットを含めて、32 ビットで値を読み書きしてください。

リセットおよびスタンバイモード時に、H'00000000 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

#### ・ ビット 31 ~ 16 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット 15、14 : デスティネーションアドレスモードビット 1、0 (DM1、DM0)

DMA 転送先のアドレスの増減を指定します (シングルアドレスモードにおいて、メモリマップト外部デバイス、内蔵周辺モジュール、または外部メモリから、DACK 付きの外部デバイスへの転送をする場合には、DM1、DM0 ビットは無視されます)。

DM1、DM0 ビットは、リセットおよびスタンバイモード時に、00 に初期化されます。モジュールスタンバイ機能実行時は値を保持します。

ビット 15	ビット 14	説 明
DM1	DM0	
0	0	デスティネーションアドレスは固定 (初期値)
0	1	デスティネーションアドレスは増加 (バイト単位転送時は + 1、ワード単位転送時は + 2、ロングワード単位転送時は + 4、16 バイト単位転送時は + 16)
1	0	デスティネーションアドレスは減少 (バイト単位転送時は - 1、ワード単位転送時は - 2、ロングワード単位転送時は - 4、16 バイト単位転送時は - 16)
1	1	予約 (設定禁止)

・ ビット 13、12 : ソースアドレスモードビット 1、0 (SM1、SM0)

DMA 転送元のアドレスの増減を指定します (シングルアドレスモードにおいて、DACK 付きの外部デバイスから、メモリマップト外部デバイス、内蔵周辺モジュール、または外部メモリへの転送をする場合には SM1、SM0 ビットは無視されます)。

16 バイト転送時は、SM1、SM0 の値にかかわらず + 16 されます。

SM1、SM0 ビットは、リセットおよびスタンバイモード時に、00 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 13	ビット 12	説 明
SM 1	SM 0	
0	0	ソースアドレスは固定 (16 バイト単位転送時は、+ 16) (初期値)
0	1	ソースアドレスは増加 (バイト単位転送時は + 1、ワード単位転送時は + 2、ロングワード単位転送時は + 4、16 バイト単位転送時は + 16)
1	0	ソースアドレスは減少 (バイト単位転送時は - 1、ワード単位転送時は - 2、ロングワード単位転送時は - 4、16 バイト単位転送時は + 16)
1	1	予約 (設定禁止)

・ ビット 11、10 : トランスファサイズビット (TS1、TS0)

DMA 転送の単位を選択します。転送元または転送先が、転送サイズが指定された内蔵周辺モジュールのレジスタの場合には、必ずその転送サイズを選んでください。

TS1、TS0 ビットに 11 (16 バイト単位) を設定した場合のリクエストモードは、オートリクエストモード、もしくは外部リクエストモード・エッジ検出設定時のみ利用可能です。

TS1、TS0 ビットに 11 を設定し、外部リクエストモード・レベル検出、および内部周辺モジュールリクエストモードに設定した場合の動作は保証しません。

TS1、TS0 ビットは、リセットおよびスタンバイモード時に、00 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

ビット 11	ビット 10	説 明
TS1	TS0	
0	0	バイト単位 (初期値)
0	1	ワード (2 バイト) 単位
1	0	ロングワード (4 バイト) 単位
1	1	16 バイト単位 (ロングワード 4 回転送)

### ・ ビット 9 : オートリクエストモードビット (AR)

転送要求を、オートリクエスト (DMAC 内部で転送要求を自動発生) にするか、モジュールリクエスト (外部リクエスト、もしくは内蔵 SCIF モジュール、TPU モジュール、SIO モジュールからのリクエスト) にするかを選択します。

AR ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能実行時は値を保持します。

ビット 9	説 明
AR	
0	モジュールリクエストモード (初期値)
1	オートリクエストモード

### ・ ビット 8 : アクノリッジ/トランスファモードビット (AM)

デュアルアドレスモードで、DACK<sub>n</sub> をデータの読み出しサイクルで出力するか、書き込みサイクルで出力するかを選択します。また、シングルアドレスモードで、メモリからデバイスヘデータを転送するのか、デバイスからメモリヘデータを転送するのかを選択します。

AM ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 8	説 明
AM	
0	読み出しサイクルで DACK <sub>n</sub> を出力 (デュアルアドレスモード) (初期値) メモリからデバイスヘデータを転送 (シングルアドレスモード)
1	書き込みサイクルで DACK <sub>n</sub> を出力 (デュアルアドレスモード) デバイスからメモリヘデータを転送 (シングルアドレスモード)

### ・ ビット 7 : アクノリッジレベルビット (AL)

DACK<sub>n</sub> を、アクティブハイ信号にするか、アクティブロー信号にするかを選択します。

AL ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

AL	説 明
ビット 7	
0	DACKn はアクティブロー信号 (初期値)
1	DACKn はアクティブハイ信号

・ ビット 6 : DREQ セレクトビット (DS)

DREQn 入力の検出方法を選択します。

DS ビットに 0 (レベル検出) を設定した場合は、トランスファバスモードビット (TB) を 0 (サイクルスチールモード) に設定してください。

DS ビットに 0 を設定し、TB ビットを 1 (バーストモード) に設定した場合の動作は保証しません。

DS ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 6	説 明
DS	
0	レベル検出 (サイクルスチールモード時のみ設定可能) (初期値)
1	エッジ検出

・ ビット 5 : DREQn レベルビット (DL)

DREQn 入力の検出レベルを選択します。

DL ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 5	説 明
DL	
0	DS が 0 のとき、DREQ をローレベル検出、DS が 1 のとき、DREQ を立ち下がり検出 (初期値)
1	DS が 0 のとき、DREQ をハイレベル検出、DS が 1 のとき、DREQ を立ち上がり検出

・ ビット 4 : トランスファバスモードビット (TB)

DMA 転送のバスモードを選択します。

TB ビットに 1 (バーストモード) を設定した場合は、必ず DREQ セレクトビット (DS) を 1 (エッジ検出) に設定してください。

TB ビットに 1 を設定し、DS ビットを 0 (レベル検出) に設定した場合の動作は保証しません。

TB ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 4	説 明
TB	
0	サイクルスチールモード (初期値)
1	バーストモード

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### ・ ビット3：トランスファアドレスモードビット (TA)

DMA 転送のアドレスモードを選択します。

TA ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット3	説 明
TA	
0	デュアルアドレスモード (初期値)
1	シングルアドレスモード

### ・ ビット2：インタラプトイネーブルビット (IE)

DMA 転送終了時に CPU に割り込みを要求するかどうかを指定します。IE ビットを 1 にセットしておく、TE ビットがセットされると CPU に対し割り込み (DEI) を要求します。

IE ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット2	説 明
IE	
0	割り込み要求を禁止 (初期値)
1	割り込み要求を許可

### ・ ビット1：トランスファエンドフラグビット (TE)

DMA 転送の終了を示します。DMA トランスファカウンレジスタ (TCR) の値が 0 になり、DMA 転送が終了すると、TE ビットは 1 にセットされます。TCR が 0 にならないときに NMI 割り込み、DMA アドレスエラーによって転送が終了した場合や、DE ビット、DMA オペレーションレジスタ (DMAOR) の DME ビットをクリアして転送を終了させた場合には、TE ビットはセットされません。TE ビットをクリアするには、TE ビットの 1 を読み出してから 0 を書き込みます。

TE ビットがセットされていると、DE ビットを 1 にセットしても転送は許可されません。

TE ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット1	説 明
TE	
0	DMA 転送中または DMA 転送中断 [クリア条件] TE ビットの 1 を読み出してから 0 を書き込む (初期値)
1	(TCR = 0 により) DMA 転送終了

- ビット0 : DMA イネーブルビット (DE)

DMA 転送を許可または禁止します。オートリクエストモードでは、DE ビットおよび DMAOR の DME ビットを 1 にセットすると転送を開始します。ただし、TE ビット、DMAOR の NMIF ビット、AE ビットのすべてが 0 であることが必要です。外部リクエスト、内蔵周辺モジュールリクエストモードでは、DE ビットと DME ビットを 1 にセットした後で、さらに該当デバイスまたは内蔵周辺モジュールから DMA 転送要求があると、転送を開始します。ただし、この場合にもオートリクエストモードと同じく、TE ビット、NMIF ビット、AE ビットのすべてが 0 であることが必要です。DE ビットをクリアすると、転送を中断することができます。

DE ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット0	説明
DE	
0	DMA 転送を禁止 (初期値)
1	DMA 転送を許可

### 11.2.5 ベクタ番号設定レジスタ DMA0、1 (VCRDMA0、VCRDMA1)

ビット :	31	30	29	-----	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	-----	—	—	—	VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0
初期値 :	0	0	0	-----	0	0	0	*	*	*	*	*	*	*	*
R/W :	R	R	R	-----	R	R	R	R/W							

\* : 不定

ベクタ番号設定レジスタ DMA0、1 (VCRDMA0、VCRDMA1) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMAC の転送終了割り込みのベクタ番号を設定します。ただし、32 ビット中の下位の 8 ビットが有効です。上位 24 ビットを含めて、32 ビットで値を読み書きしてください。

リセット、スタンバイモード、およびモジュールスタンバイ機能使用時には値を保持します。

- ビット 31 ~ 8 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット 7 ~ 0 : ベクタ番号ビット 7 ~ 0 (VC7 ~ VC0)

DMAC 転送終了時の割り込みベクタ番号を設定します。割り込みベクタ番号は 0 から 127 の値で設定します。転送終了割り込みが発生すると、ベクタ番号が取り込まれて、指定された割り込み処理ルーチンに制御が渡されます。

VC7 ~ VC0 ビットは、リセットおよびスタンバイモード時には値を保持します。

ベクタ番号は 127 までなので、VC7 は必ず 0 を書き込んでください。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

### 11.2.6 DMA 要求 / 応答選択制御レジスタ 0、1 (DRCR0、DRCR1)

ビット:	7	6	5	4	3	2	1	0
	—	—	—	RS4	RS3	RS2	RS1	RS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

DMA 要求 / 応答選択制御レジスタ 0、1 (DRCR0、DRCR1) は、読み出し / 書き込み可能な 8 ビットのレジスタで、DMAC の転送要求元を設定します。8 ビットで値を書き込みます。

リセット時には H'00 に初期化されます。スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

- ・ ビット 7 ~ 5 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 4 ~ 0 : リソースセレクトビット 4 ~ 0 (RS4 ~ RS0)

どの転送要求を DMAC に入力するかを指定します。転送要求元の変更は、必ず、DMA イネーブルビット (DE) が 0 の状態で行ってください。

利用可能な設定の組み合わせは「11.3.4 DMA 転送の種類」を参照ください。

RS4 ~ RS0 ビットは、リセット時に 00 に初期化されます。

11. ダイレクトメモリアクセスコントローラ (DMAC)

ビット4	ビット3	ビット2	ビット1	ビット0	説明
RS4	RS3	RS2	RS1	RS0	
0	0	0	0	0	DREQ (外部リクエスト)(初期値)
				1	予約 (設定禁止)
			1	0	予約 (設定禁止)
				1	予約 (設定禁止)
		1	0	0	予約 (設定禁止)
				1	SCIF チャンネル 1 RXI (内蔵 FIFO 付 SCI のチャンネル 1 側受信データフル割り込み要求) * <sup>1</sup>
			1	0	SCIF チャンネル 1 TXI (内蔵 FIFO 付 SCI のチャンネル 1 側送信データエンプティ割り込み要求) * <sup>1</sup>
				1	予約 (設定禁止)
0	1	0	0	0	予約 (設定禁止)
				1	SCIF チャンネル 2 RXI (内蔵 FIFO 付 SCI のチャンネル 2 側受信データフル割り込み要求) * <sup>1</sup>
			1	0	SCIF チャンネル 2 TXI (内蔵 FIFO 付 SCI のチャンネル 2 側送信データエンプティ割り込み要求) * <sup>1</sup>
				1	予約 (設定禁止)
		1	0	0	TPU TGI0A (内蔵 TPU のインプットキャプチャ チャンネル 0A 割り込み要求) * <sup>1</sup>
				1	TPU TGI0B (内蔵 TPU のインプットキャプチャ チャンネル 0B 割り込み要求) * <sup>1</sup>
			1	0	TPU TGI0C (内蔵 TPU のインプットキャプチャ チャンネル 0C 割り込み要求) * <sup>1</sup>
				1	TPU TGI0D (内蔵 TPU のインプットキャプチャ チャンネル 0D 割り込み要求) * <sup>1</sup>
1	0	0	0	0	予約 (設定禁止)
				1	SIO チャンネル 0 RDFI (内蔵 SIO チャンネル 0 側受信データフル割り込み要求) * <sup>1</sup>
			1	0	SIO チャンネル 0 TDEI (内蔵 SIO チャンネル 0 側送信データエンプティ割り込み要求) * <sup>1</sup>
				1	予約 (設定禁止)
		1	0	0	予約 (設定禁止)
				1	SIO チャンネル 1 RDFI (内蔵 SIO チャンネル 1 側受信データフル割り込み要求) * <sup>1</sup>
			1	0	SIO チャンネル 1 TDEI (内蔵 SIO チャンネル 1 側送信データエンプティ割り込み要求) * <sup>1</sup>
				1	予約 (設定禁止)
	1	0	0	0	予約 (設定禁止)
				1	SIO チャンネル 2 RDFI (内蔵 SIO チャンネル 2 側受信データフル割り込み要求) * <sup>1</sup>
		1	0	SIO チャンネル 2 TDEI (内蔵 SIO チャンネル 2 側送信データエンプティ割り込み要求) * <sup>1</sup>	
			1	予約 (設定禁止)	
	1	*	*	予約 (設定禁止)	

【注】 \*<sup>1</sup> 内蔵モジュールから転送要求を発生させる場合は、バスモードをサイクルスチール、転送モードをデュアル転送にし、DREQn を立ち下がり検出に設定してください。

\* Don't care

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### 11.2.7 DMA オペレーションレジスタ (DMAOR)

ビット:	31	30	29	-----	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	-----	—	—	—	—	—	—	—	PR	AE	NMIF	DME
初期値:	0	0	0	-----	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	-----	R	R	R	R	R	R	R	R/W	R(W)*	R(W)*	R/W

【注】\*フラグをクリアするために0のみ書き込むことができます。

DMA オペレーションレジスタ (DMAOR) は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送モードを制御します。また、DMA 転送の状態 (ステータス) も示します。ただし、32 ビット中の下位の 4 ビットが有効です。上位 28 ビット含めて、32 ビットで値を読み書きしてください。

リセットおよびスタンバイモード時に、H'00000000 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

- ・ ビット 31 ~ 4 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 3 : プライオリティモードビット (PR)

同時に複数のチャンネルに対して転送要求があった場合のチャンネルの優先順位の決定を固定モードにするかラウンドロビンモードにするかを指定します。

PR ビットは、リセットおよびスタンバイ時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット 3	説明
PR	
0	優先順位固定モード (チャンネル 0 > チャンネル 1) (初期値)
1	ラウンドロビンモード (1 転送毎に優先順位が交代する方式) (リセット後初めての DMA 転送時の優先順位はチャンネル 1 > チャンネル 0)

- ・ ビット 2 : アドレスエラーフラグビット (AE)

DMAC によるアドレスエラーが発生したことを示します。AE ビットがセットされていると、DMA チャンネルコントロールレジスタ (CHCR) の DE ビットを 1 にセットしても、DMA 転送は許可されません。AE ビットをクリアするには、AE ビットの 1 を読み出してから 0 を書き込みます。

アドレスエラーが起きたとき実行中の DMA 転送までは行われます。

AE ビットは、リセットおよびスタンバイモード時に 0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット2	説 明
AE	
0	DMAC によるアドレスエラーなし (初期値) [クリア条件] AE ビットの 1 を読み出してから 0 を書き込む
1	DMAC によるアドレスエラー発生

・ ビット1 : NMI フラグビット (NMIF)

NMI 割り込みが発生したことを示します。NMIF ビットがセットされていると、DMA チャンネルコントロールレジスタ (CHCR) の DE ビットと DME ビットを 1 にセットしても、DMA 転送は許可されません。NMIF ビットをクリアするには、NMIF ビットの 1 を読み出してから 0 を書き込みます。NMI が入力されたとき、実行中の DMA 転送までは行われず、DMAC が動作していないときに、NMI 割り込みが入力されても、NMIF ビットは 1 にセットされます。

NMIF ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット1	説 明
NMIF	
0	NMI 割り込みなし (初期値) [クリア条件] NMIF ビットの 1 を読み出してから 0 を書き込む
1	NMI 割り込み発生

・ ビット0 : DMA マスタイネーブルビット (DME)

すべてのチャンネルの DMA 転送を許可または禁止します。DME ビットおよび各 DMA チャンネルコントロールレジスタ (CHCR) の DE ビットを 1 にセットすると、DMA 転送が許可されます。ただし、転送を行うチャンネルの CHCR にある TE ビット、NMIF ビット、AE ビットのすべてが 0 であることが必要です。DME ビットをクリアすると、すべてのチャンネルの DMA 転送が中断されます。DME ビットは、リセットおよびスタンバイモード時に、0 に初期化されます。モジュールスタンバイ機能使用時は値を保持します。

ビット0	説 明
DME	
0	全チャンネルの DMA 転送を禁止 (初期値)
1	全チャンネルの DMA 転送を許可

### 11.3 動作説明

DMAC は、DMA 転送要求があると、決められたチャネルの優先順位に従って転送を開始し、転送終了条件が満たされると転送を終了します。転送要求には、オートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストの3種類のモードがあります。DMA 転送には、シングルアドレスモード転送とデュアルアドレスモード転送があり、バーストまたはサイクルスチールのどちらかのバスモードで動作します。

#### 11.3.1 DMA 転送フロー

DMA ソースアドレスレジスタ (SAR)、DMA デスティネーションアドレスレジスタ (DAR)、DMA トランスファカウンタレジスタ (TCR)、DMA チャネルコントロールレジスタ (CHCR)、ベクタ番号設定レジスタ DMA (VCRDMA)、DMA 要求 / 応答選択制御レジスタ (DRCR)、DMA オペレーションレジスタ (DMAOR) の初期設定後\*、DMAC は、以下の順序でデータを転送します。

【注】 \* 初期設定は各レジスタを設定し、最後に (DE = 1、DME = 1、TE = 0、NMIF = 0、AE = 0) が成立するように設定します。

- (1) 転送許可状態かどうか (DE = 1、DME = 1、TE = 0、NMIF = 0、AE = 0) をチェックします。
- (2) 転送許可状態で転送要求が発生すると1転送単位のデータを転送します (オートリクエストの場合、レジスタの初期設定後、自動的に転送を開始します)。このとき、TCR の値を1デクリメントします。具体的な転送フローは、アドレスモード、バスモードによって異なります。
- (3) 指定された回数の転送を終える (TCR の値が0になる) と、転送を正常に終了します。このとき、CHCR の IE ビットが1にセットしてあれば、CPU に DEI 割り込みを要求します。
- (4) DMAC によるアドレスエラーか NMI 割り込みが発生した場合には、転送を中断します。また、CHCR の DE ビットか DMAOR の DME ビットが0にされても、転送を中断します。

図 11.2 に、上記の DMA 転送フローチャートを示します。

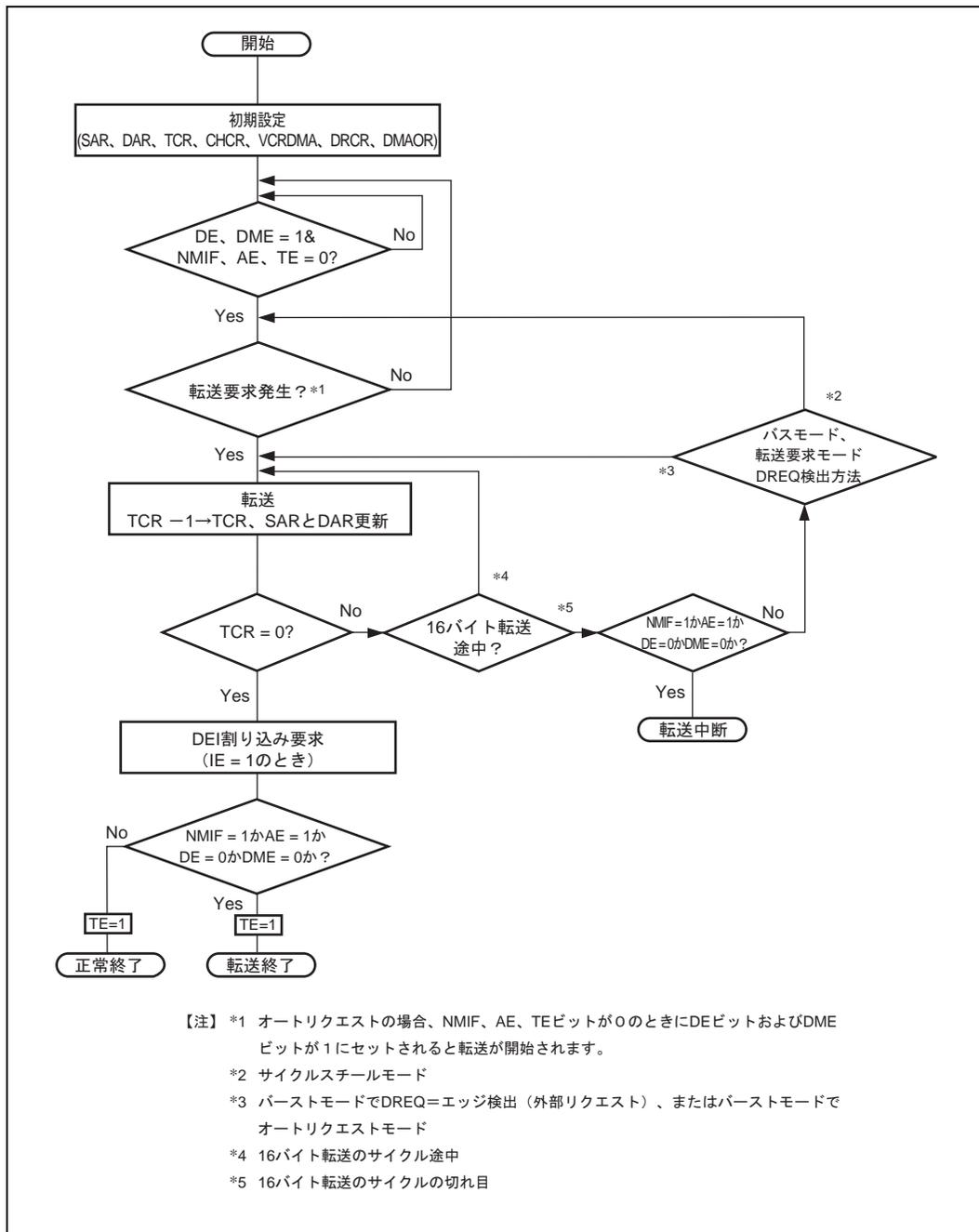


図 11.2 DMA 転送フローチャート

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### 11.3.2 DMA 転送要求

DMA 転送要求は、データの転送元または転送先に発生させるのが基本的な使い方ですが、転送元でも転送先でもないデバイスに発生させる使い方もできます。

転送要求には、オートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストの3種類があります。転送要求の選択は、DMA チャンネルコントロールレジスタ 0、1 (CHCR0、CHCR1) の AR ビットと、DMA 要求 / 応答選択制御レジスタ 0、1 (DRCR0、DRCR1) の RS4 ~ RS0 ビットによって行います。

表 11.3 AR、RS ビットによる DMA 転送要求の選択

CHCR AR	DRCR					リクエストモード	リソース選択		
	RS4	RS3	RS2	RS1	RS0				
0	0	0	0	0	0	モジュール リクエスト モード	DREQ (外部リクエスト)		
			1	0	1		SCIF チャンネル 1 RXI		
			1	0	0		SCIF チャンネル 1 TXI		
		1	0	0	0		1	SCIF チャンネル 2 RXI	
				1	0		0	SCIF チャンネル 2 TXI	
			1	0	0		0	1	TPU TGI0A
					1		0	0	TPU TGI0B
	1	0	0	0	0		0	TPU TGI0C	
				1	0		1	TPU TGI0D	
				1	0		0	SIO チャンネル 0 RDFI	
			1	0	0		0	0	SIO チャンネル 0 TDEI
					1		0	1	SIO チャンネル 1 RDFI
					1		0	0	SIO チャンネル 1 TDEI
					1		0	1	SIO チャンネル 2 RDFI
1	0	0	1	0	SIO チャンネル 2 TDEI				
1	*	*	*	*	*	オートリクエストモード			

【注】 \* Don't care.

#### (1) オートリクエストモード

オートリクエストモードは、メモリ同士の転送や、メモリと転送を要求できない内蔵周辺モジュールとの転送のように、転送要求信号が DMAC の外部から来ない場合に、DMAC の内部で自動的に転送要求信号を発生するモードです。CHCR0 ~ CHCR1 の DE ビットおよび DMA オペレーションレジスタ (DMAOR) の DME ビットを 1 にセットすると転送が開始されます (ただし、CHCR0、CHCR1 の TE ビット、DMAOR の NMIF ビット、AE ビットのすべてが 0 である必要があります)。

#### (2) 外部リクエストモード

外部リクエストモードは、本 LSI の外部のデバイスからの転送要求信号 (DREQ<sub>n</sub>) によって転送を開始させるモードです。応用システムに応じて、表 11.4 に示すモードの中から 1 つを選んで使います。DMA 転送が許可されているとき (DE = 1、DME = 1、TE = 0、NMIF = 0、AE = 0) に DREQ<sub>n</sub> が入力されると DMA 転送が開始されます。

表 11.4 TA、AM ビットによる外部リクエストモードの選択

CHCR		トランスファ アドレスモード	アクノリッジ モード	転送元	転送先
TA	AM				
0	0	デュアルアドレス モード	読み出しサイクルで DACKn を出力	任意 *1	任意 *1
0	1	デュアルアドレス モード	書き込みサイクルで DACKn を出力	任意 *1	任意 *1
1	0	シングルアドレス モード	メモリからデバイス ヘデータを転送	外部メモリまたはメモリ マップト外部デバイス	DACK 付き外部デバイス
1	1	シングルアドレス モード	デバイスからメモリ ヘデータを転送	DACK 付き外部デバイス	外部メモリまたはメモリ マップト外部デバイス

【注】 \*1 外部メモリ、メモリマップト外部デバイス、内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く)

DREQn を立ち下がりエッジ / 立ち上がりエッジで検出するか、ローレベル / ハイレベルで検出するかは、CHCR0、CHCR1 の DS、DL ビットで選びます (DS = 0 はレベル検出、DS = 1 はエッジ検出、DL = 0 はアクティブロー、DL = 1 はアクティブハイ)。

転送要求元は、必ずしもデータの転送元が転送先である必要はありません。

CHCR0、CHCR1 の DS ビットに 0 (レベル検出) を設定した場合は、必ず CHCR0、CHCR1 の TB ビットを 0 (サイクルスチールモード) に設定し、CHCR0、CHCR1 の TS1、TS0 ビットをには必ず 00 (バイト単位)、01 (ワード単位)、10 (ロングワード単位) のいずれかを設定してください。

CHCR0、CHCR1 の DS ビットに 0 を設定し、CHCR0、CHCR1 の TB ビットを 1 (バーストモード) に設定した場合、および CHCR0、CHCR1 の TS1、TS0 ビットに 11 (16 バイト単位) を設定した場合の動作は保証しません。

表 11.5 DS、DL ビットによる外部リクエスト信号の選択

CHCR		外部リクエスト
DS	DL	
0	0	ローレベル検出 (サイクルスチールモード時のみ設定可能)
0	1	ハイレベル検出 (サイクルスチールモード時のみ設定可能)
1	0	立ち下がりエッジで検出
1	1	立ち上がりエッジで検出

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

### (3) 内蔵周辺モジュールリクエストモード

内蔵周辺モジュールリクエストモードは、本 LSI の内蔵周辺モジュールからの転送要求信号 ( 割り込み要求信号 ) によって転送を開始させるモードです。転送要求信号には、SCIF、SIO の受信データフル割り込み ( RXI、RDFI )、SCIF、SIO の送信データエンプティ割り込み ( TXI、TDEI )、TPU のジェネラルレジスタがあります ( 表 11.6 )。DMA 転送が許可されているとき ( DE = 1、DME = 1、TE = 0、NMIF = 0、AE = 0 ) に転送要求信号が入力されると DMA 転送が開始されます。

転送要求を RXI/RDFI ( SCIF、SIO の受信データフルによる転送要求 ) に設定した場合は、転送元はそれぞれ対応するモジュールのレシーブデータレジスタ ( SCFRDR、SIRDR ) でなければなりません。転送要求を TXI/TDEI ( SCIF、SIO の送信データエンプティによる転送要求 ) に設定した場合は、転送先はそれぞれ対応するモジュールのトランスミットデータレジスタ ( SCFTDR、SITDR ) でなければなりません。

TPU の転送要求はこの限りではありません。

内蔵周辺モジュールリクエストモードを使用する場合は、必ず CHCR0、CHCR1 の TS1、TS0 ビットに転送元または転送先である周辺モジュールのレジスタで許されるアクセスサイズを設定してください。

表 11.6 AR、RS ビットによる内蔵周辺モジュールリクエストモードの選択

AR	RS4	RS3	RS2	RS1	RS0	DMA 転送 要求元	DMA 転送 要求信号	転送元	転送先	バス モード	DREQ 設定		
0	0	0	1	0	1	SCIF チャンネル 1 受信部	RXI	SCFRDR1	任意	サイクル スチール	エッジ、ロー アクティブ		
				1	0	SCIF チャンネル 1 送信部	TXI	任意	SCFTDR1	サイクル スチール	エッジ、ロー アクティブ		
			1	0	0	1	SCIF チャンネル 2 受信部	RXI	SCFRDR2	任意	サイクル スチール	エッジ、ロー アクティブ	
					1	0	SCIF チャンネル 2 送信部	TXI	任意	SCFTDR2	サイクル スチール	エッジ、ロー アクティブ	
			1	0	0	0	0	TPU チャンネル 0A	TGI0A	内蔵 RAM 以外で任意	内蔵 RAM 以外で任意	サイクル スチール	エッジ、ロー アクティブ
						1	0	TPU チャンネル 0B	TGI0B	内蔵 RAM 以外で任意	内蔵 RAM 以外で任意	サイクル スチール	エッジ、ロー アクティブ
					1	0	0	TPU チャンネル 0C	TGI0C	内蔵 RAM 以外で任意	内蔵 RAM 以外で任意	サイクル スチール	エッジ、ロー アクティブ
						1	0	TPU チャンネル 0D	TGI0D	内蔵 RAM 以外で任意	内蔵 RAM 以外で任意	サイクル スチール	エッジ、ロー アクティブ
	1	0	0	0	0	1	SIO チャンネル 0 受信部	RDFI	SIRDR0	任意	サイクル スチール	エッジ、ロー アクティブ	
					1	0	SIO チャンネル 0 送信部	TDEI	任意	SITDR0	サイクル スチール	エッジ、ロー アクティブ	
				1	0	1	1	SIO チャンネル 1 受信部	RDFI	SIRDR1	任意	サイクル スチール	エッジ、ロー アクティブ
						1	0	SIO チャンネル 1 送信部	TDEI	任意	SITDR1	サイクル スチール	エッジ、ロー アクティブ
			1	0	0	0	1	SIO チャンネル 2 受信部	RDFI	SIRDR2	任意	サイクル スチール	エッジ、ロー アクティブ
						1	0	SIO チャンネル 2 送信部	TDEI	任意	SITDR2	サイクル スチール	エッジ、ロー アクティブ

SCIF、SIO、TPU から転送要求を出力させるためには、該当する割り込み許可ビットをセット (SCSCR の TIE、RIE 等) して、割り込み信号を出力させてください。

なお、この内蔵周辺モジュールからの転送要求信号 (割り込み要求信号) は、DMAC だけではなく CPU にも送られてしまいます。内蔵周辺モジュールを転送要求元に指定した場合には、CPU が割り込み要求信号を受け付

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

けないように、割り込みコントローラ (INTC) の割り込み優先レベル設定レジスタ (IPRD ~ IPRE) に設定するレベル値を、CPU のステータスレジスタ (SR) の I3 ~ I0 ビットの値以下にしてください。

表 11.6 の各 DMA 転送要求信号は、対応する DMA 転送が行われると、1 回目の転送で各モジュールからの DMA 転送要求 (割り込み要求) がクリアされます。

### 11.3.3 チャンネルの優先順位

DMAC は、同時に複数のチャンネルに対して転送要求があった場合には、決められた優先順位に従って転送を行います。チャンネルの優先順位は、固定、ラウンドロビンの 2 種類のモードから選べます。モードの選択は、DMA オペレーションレジスタ (DMAOR) の PR ビットによって行います。

#### (1) 固定モード

固定モードでは、チャンネルの優先順位は決まっています。PR ビットを 0 にすると、優先順位は、高い順にチャンネル 0、チャンネル 1 になります。

バーストモードにおける転送例を図 11.3 に示します。

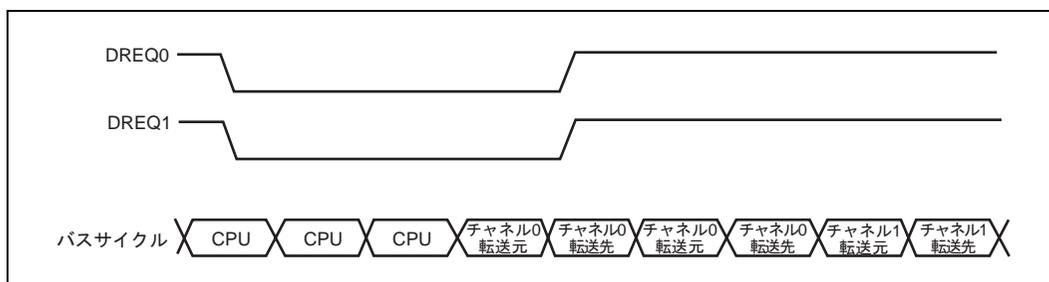


図 11.3 バーストモード時の固定モード DMA 転送  
(デュアルアドレス、DREQn 立ち下がりエッジ検出)

サイクルスチールモードでは、バスサイクル有効活用のため、チャンネル 0 のリクエストが受け付けられると、次のリクエストが受け付け可能となるまでチャンネル 1 のリクエストも受け付け可能となります。したがって DMA 動作開始時にチャンネル 0 とチャンネル 1 のリクエストが同時に来たら、1 回目はチャンネル 0 が転送を行います。その後チャンネル 1、チャンネル 0 と交互に転送を行います。

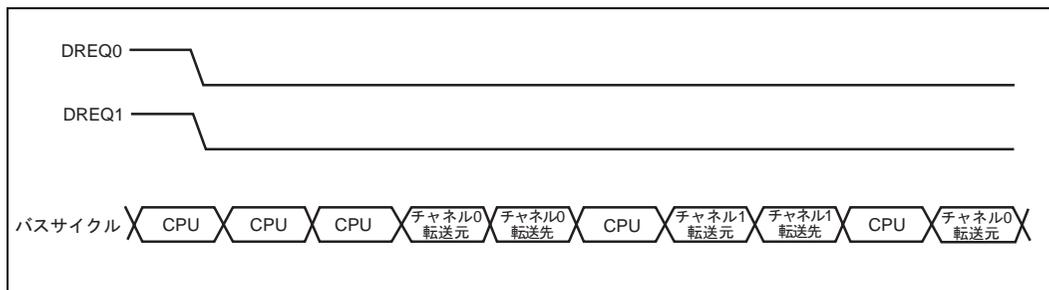


図 11.4 サイクルスチールモード時の固定モード DMA 転送  
(デュアルアドレス、DREQn ロールレベル検出)

(2) ラウンドロビンモード

ラウンドロビンモードでは、転送要求を受け付けることのできるチャンネル0とチャンネル1の間で優先順位を入れ替えます。チャンネル0またはチャンネル1で、転送が1回終了するごとに、そのチャンネルの優先順位がもう一方のチャンネルの優先順位より低くなるように優先順位を入れ替えます。

リセット直後の優先順位は、高い順にチャンネル1、チャンネル0です。

図 11.5 に、チャンネル0とチャンネル1に同時に転送要求が発生し、両方のチャンネルの転送終了後に再びチャンネル0に転送要求が発生した場合のチャンネルの優先順位の変化を示します。この場合のDMACの動作は以下のようになります。

- (1) チャンネル0とチャンネル1に同時に転送要求が発生します。
- (2) チャンネル1のほうがチャンネル0より優先順位が高いので、チャンネル1の転送を開始します(チャンネル0は転送待ち)。
- (3) チャンネル1の転送を終了すると、チャンネル1の優先順位を一番低くします。
- (4) チャンネル0の転送を開始します。
- (5) チャンネル0の転送を終了すると、チャンネル0の優先順位を一番低くします。
- (6) チャンネル0の転送要求が発生します。
- (7) チャンネル0の転送を開始します。
- (8) チャンネル0の転送を終了します。この場合、チャンネル0の優先順位はすでに一番低いので優先順位は変更しません。

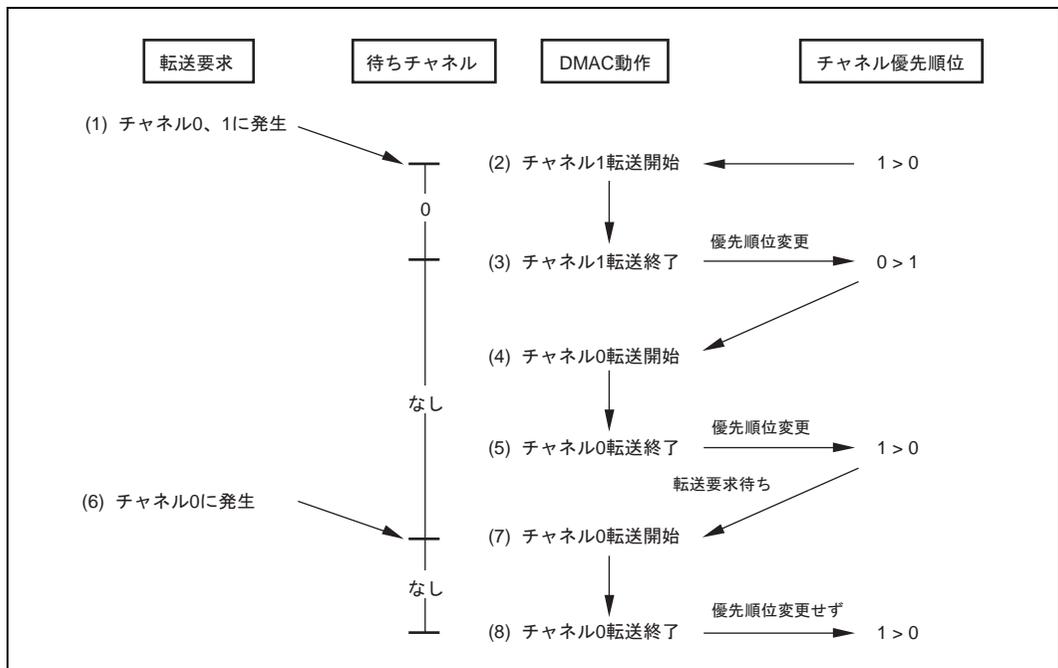


図 11.5 ラウンドロビンモードでのチャンネル優先順位変更例

## 11.3.4 DMA 転送の種類

DMA 転送は、転送元と転送先を何回のバスサイクルでアクセスするかによって、シングルアドレスモード転送とデュアルアドレスモード転送に分けられます。具体的な転送動作タイミングは、バスモードによって異なります。バスモードには、サイクルスチールモードとバーストモードがあります。表 11.7 に、DMAC がサポートできる転送を示します。

表 11.7 サポートできる DMA 転送

転送元 \ 転送先	DACK付き		メモリマップト	内蔵周辺	
	外部デバイス	外部メモリ	外部デバイス	モジュール	内蔵メモリ
DACK付き 外部デバイス	不可	シングル	シングル	不可	不可
外部メモリ	シングル	デュアル	デュアル	デュアル*	デュアル
メモリマップト 外部デバイス	シングル	デュアル	デュアル	デュアル*	デュアル
内蔵周辺 モジュール	不可	デュアル*	デュアル*	デュアル*	デュアル*
内蔵メモリ	不可	デュアル	デュアル	デュアル*	デュアル

シングル：シングルアドレスモード

デュアル：デュアルアドレスモード

【注】 \* 転送元または転送先である内蔵周辺モジュールのレジスタで許されるアクセスサイズ (DMAC、BSC、UBC、キャッシュメモリ、E-DMAC、EtherC を除く)

## (1) アドレスモード

## (a) シングルアドレスモード

シングルアドレスモードは、転送元と転送先がともに外部で、そのうち的一方を DACK<sub>n</sub> 信号によってアクセス (選択) し、もう一方をアドレスによってアクセスする場合に使うモードです。このモードでは、DMAC は、転送要求受け付け信号 DACK<sub>n</sub> を一方の外部デバイスに出力してアクセスすると同時に、転送相手にアドレスを出して、1 つのバスサイクルで DMA 転送を行います。例えば、図 11.6 のような外部メモリと DACK 付き外部デバイスとの転送では、外部デバイスがデータバスにデータを出力するのと同じバスサイクルでそのデータが外部メモリに書き込まれます。

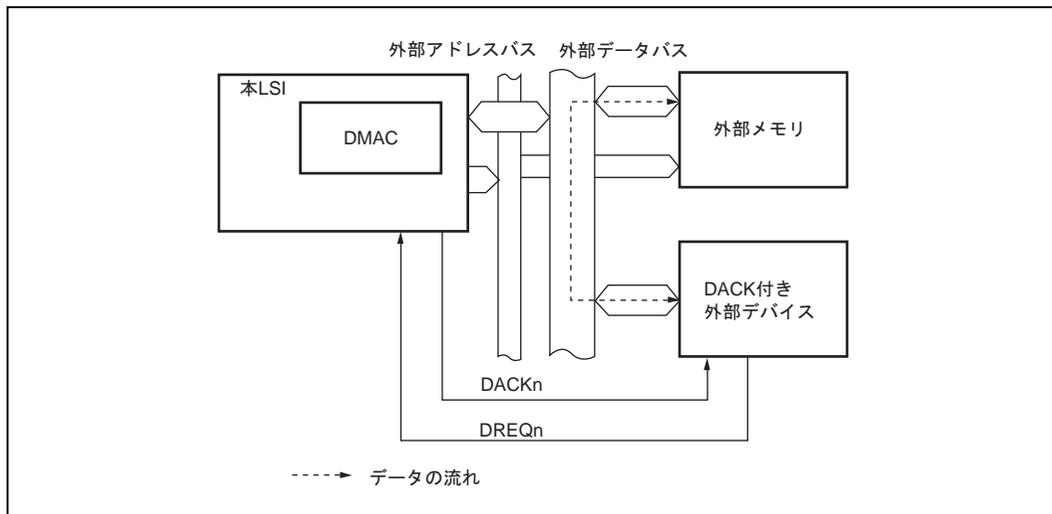


図 11.6 シングルアドレスモードでのデータの流れ

シングルアドレスモードで可能な転送は、1) DACK 付き外部デバイスとメモリマップト外部デバイス間転送、2) DACK 付き外部デバイスと外部メモリ間転送です。いずれの場合も、転送要求は外部リクエスト (DREQ<sub>n</sub>) のみです。

外部リクエスト (DREQ<sub>n</sub>) を使用した転送を行う場合に利用可能な設定の組み合わせは、表 11.9 を参照ください。

図 11.7 に、シングルアドレスモードでの DMA 転送タイミング例を示します。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

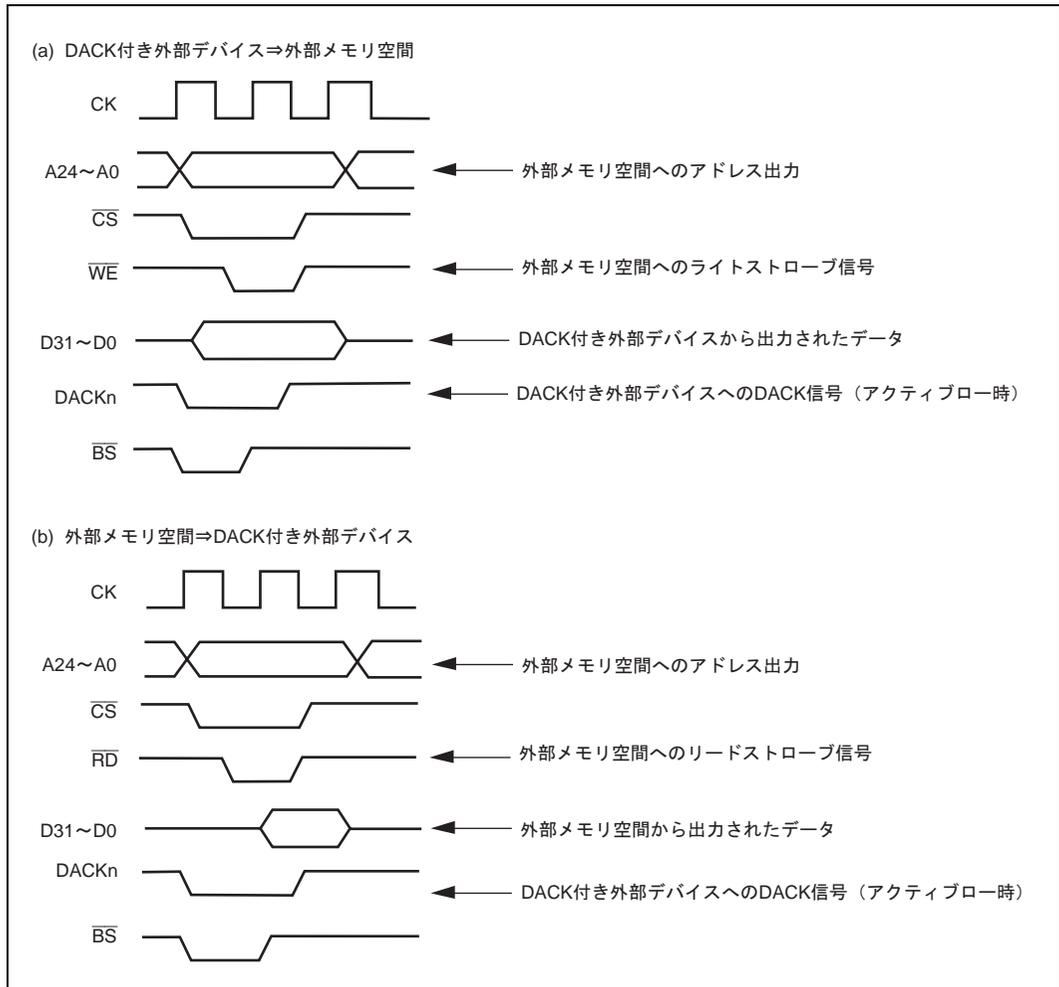


図 11.7 シングルアドレスモードでの DMA 転送タイミング例

### (b) デュアルアドレスモード

デュアルアドレスモードは、転送元と転送先をともにアドレスによってアクセス (選択) する場合に使うモードです。転送元と転送先は外部でも内部でもかまいません。このモードでは、DMAC は、読み出しサイクルで転送元を、書き込みサイクルで転送先をアクセスし、2 つのバスサイクルで転送を行います。このとき、転送データは一時的に DMAC に格納されます。例えば、図 11.8 のような外部メモリ同士の転送では、読み出しサイクルで一方の外部メモリからデータが DMAC に読み出され、続く書き込みサイクルでそのデータがもう一方の外部メモリに書き込まれます。

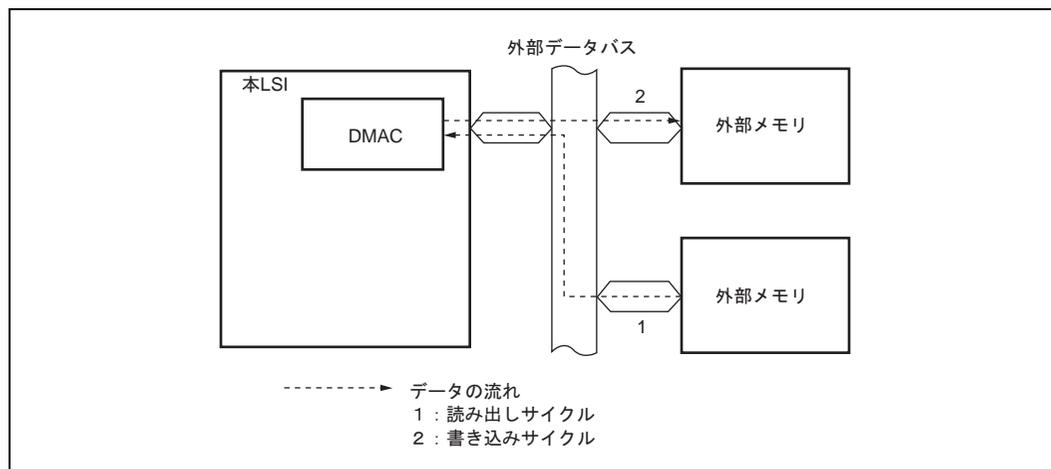


図 11.8 デュアルアドレスモードでのデータの流れ

デュアルアドレスモードでは、外部メモリ、メモリマップト外部デバイス、内蔵周辺モジュールの3種類を自由に組み合わせた転送を行うことができます。具体的には、以下の転送が可能です。

- 1) 外部メモリと外部メモリ間転送
- 2) 外部メモリとメモリマップト外部デバイス間転送
- 3) メモリマップト外部デバイスとメモリマップト外部デバイス間転送
- 4) 外部メモリと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く) 間転送\*
- 5) メモリマップト外部デバイスと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く) 間転送\*
- 6) 内蔵メモリと内蔵メモリ間転送
- 7) 内蔵メモリとメモリマップト外部デバイス間転送
- 8) 内蔵メモリと内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く) 間転送\*
- 9) 内蔵メモリと外部メモリ
- 10) 内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く) と、内蔵周辺モジュール (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く) 間転送\*

転送要求は、オートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストのいずれでも可能です。ただし、転送要求元が SCIF、SIO である場合に限っては、SCIF、SIO のレジスタが転送先または転送元でなければなりません (表 11.6 参照)。

外部リクエスト (DREQn) を使用した転送を行う場合に利用可能な設定の組み合わせは、表 11.9 を参照ください。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

デュアルアドレスモード時は、 $DACK_n$  はリードサイクルあるいはライトサイクルに出力されます。リードサイクルあるいはライトサイクルのどちらに出力するかは DMA チャンネルコントロールレジスタ 0、1 (CHCR0、1) のアクノリッジ/トランスファモードビット (AM) によって設定可能です。

【注】 \* 転送元または転送先である内蔵周辺モジュールのレジスタで許されるアクセスサイズ (DMAC、BSC、UBC、キャッシュ、E-DMAC、EtherC を除く)

図 11.9 に、デュアルアドレスモードでの DMA 転送タイミング例を示します。

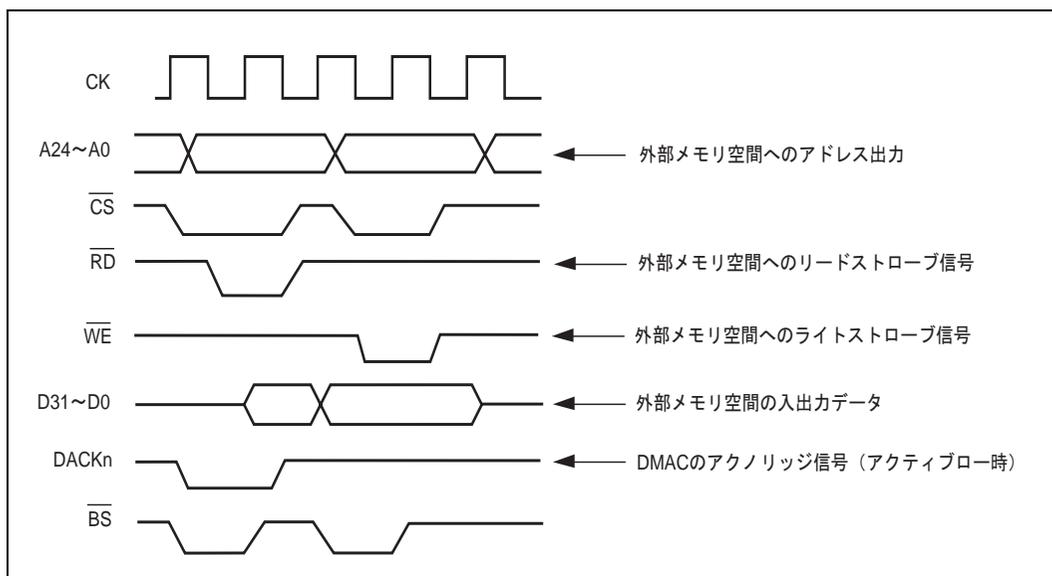


図 11.9 デュアルアドレスモードでの DMA 転送タイミング例  
(外部メモリ空間 外部メモリ空間、読み出しサイクルに  $DACK_n$  出力の場合)

### (2) バスモード

バスモードには、サイクルスチールモードとバーストモードがあります。モードの選択は、CHCR1 ~ CHCR0 の TB ビットによって行います。

#### (a) サイクルスチールモード

サイクルスチールモードでは、DMAC は転送を 1 回終了するたびにバス権を他のバスマスタに渡します。その後転送要求があれば、他のバスマスタからバス権を取り戻し、再び転送を 1 回行い、その転送を終了すると、またバス権を他のバスマスタに渡します。これを転送終了条件が満たされるまで繰り返します。

サイクルスチールモードは、転送要求元、転送先、転送元にかかわらずすべての転送区間で使えます。アクノリッジ信号を書き込みサイクルで出力した場合とシングルモードの場合は、バス権を CPU に返したときに 2 回分 CPU がバス権を取る可能性があります。

図 11.10 に、サイクルスチールモードでの DMA 転送タイミング例を示します。図の例での転送条件は以下のとおりです。

サイクルスチールモードを使用して転送要求元を外部リクエストモード・レベル検出とした場合は、必ず CHCR0、CHCR1 の TS1、TS0 ビットを 00 (バイト単位)、01 (ワード単位)、10 (ロングワード単位) のいずれかに設定してください。

CHCR0、CHCR1 の TS1、TS0 ビットを 11 (16 バイト転送) に設定した場合の動作は保証しません。

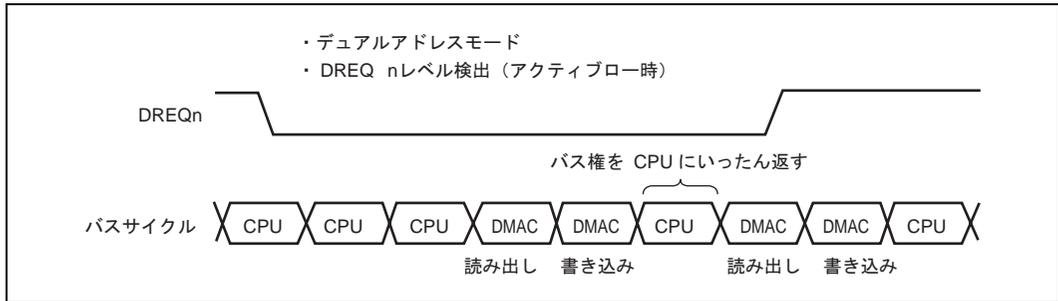


図 11.10 サイクルスチールモードでの DMA 転送例 (デュアルアドレス、DREQ<sub>n</sub> ローレベル検出)

#### (b) バーストモード

バーストモードでは、DMAC は一度バス権を得ると、転送終了条件が満たされるまでバス権を解放せずに転送を続けます。

ただし、内蔵周辺モジュールが転送要求元である場合には常にサイクルスチールになります。

図 11.11 に、バーストモードでの DMA 転送タイミング例を示します。図の例での転送条件は以下のとおりです。

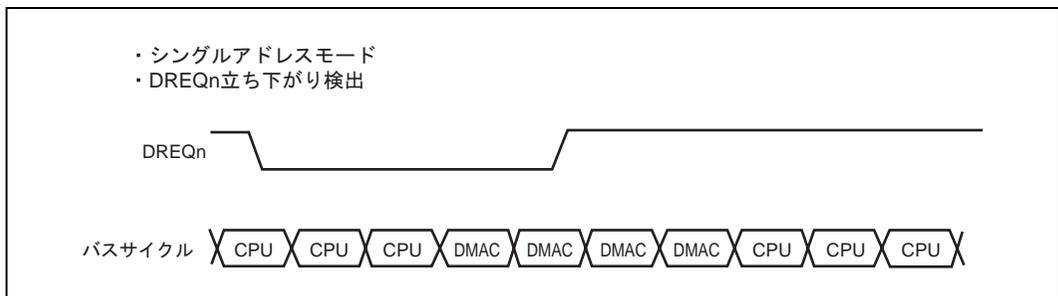


図 11.11 バーストモードでの DMA 転送例 (シングルアドレス、DREQ<sub>n</sub> 立ち下がり検出)

バースト転送中は、リフレッシュが入りませんので、リフレッシュが必要なメモリを使用している場合は、リフレッシュ要求時間を満たすように転送回数を決定してください。

また、バーストモードを使用して、転送要求元を外部リクエスト (DREQ<sub>n</sub>) とした場合は、必ず CHCR0、CHCR1 の DS ビットを 1 (エッジ検出) に設定してください。

CHCR0、CHCR1 の DS ビットを 0 (レベル検出) に設定した場合の動作は保証しません。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### (3) DMA 転送区間とリクエストモード、バスモードとの関係一覧

表 11.8 に、DMA 転送区間とリクエストモード、バスモードなどの関連事項との関係を示します。

表 11.8 表 DMA 転送区間とバスモード、リクエストモードとの関係一覧

アドレスモード	転送区間	リクエストモード *3	バスモード *7	転送サイズ (バイト)
シングル	DACK 付き外部デバイスと外部メモリ	外部	B/C	1/2/4/16*8
	DACK 付き外部デバイスとメモリマップト外部デバイス	外部	B/C	1/2/4/16*8
デュアル	外部メモリと外部メモリ	外部	B/C	1/2/4/16*8
		オート	B/C	1/2/4/16
		内蔵周辺モジュール *1	C	1/2/4
	外部メモリとメモリマップト外部デバイス	外部	B/C	1/2/4/16*8
		オート	B/C	1/2/4/16
		内蔵周辺モジュール *1	C	1/2/4
	メモリマップト外部デバイスとメモリマップト外部デバイス	外部	B/C	1/2/4/16*8
		オート	B/C	1/2/4/16
		内蔵周辺モジュール *1	C	1/2/4
	外部メモリと内蔵周辺モジュール	外部	B/C	1/2/4*4
		オート	B/C	1/2/4*4
		内蔵周辺モジュール *2	C	1/2/4*4
	メモリマップト外部デバイスと内蔵周辺モジュール	外部	B/C	1/2/4*4
		オート	B/C	1/2/4*4
		内蔵周辺モジュール *2	C	1/2/4*4
	内蔵メモリと内蔵メモリ	オート	B/C	1/2/4/16
	内蔵メモリとメモリマップト外部デバイス *5	外部	B/C	1/2/4/16*8
		オート	B/C	1/2/4/16
		内蔵周辺モジュール *1	C	1/2/4
	内蔵メモリと内蔵周辺モジュール	外部	B/C	1/2/4*4
オート		B/C	1/2/4*4	
内蔵周辺モジュール *2		C	1/2/4*4	
内蔵メモリと外部メモリ *6	外部	B/C	1/2/4/16*8	
	オート	B/C	1/2/4/16	
	内蔵周辺モジュール *1	C	1/2/4	
内蔵周辺モジュールと内蔵周辺モジュール	外部	B/C	1/2/4*4	
	オート	B/C	1/2/4*4	
	内蔵周辺モジュール *2	C	1/2/4*4	

【注】 B: パーストモード

C: サイクルスチールモード

\*1 内蔵周辺モジュールリクエストの場合は、SCIF、SIO を転送要求元に指定しないでください。

\*2 転送要求元が SCIF、SIO の場合は、転送元または転送先がそれぞれ SCIF、SIO である必要があります。

\*3 リクエストモードを内蔵周辺モジュールリクエストに設定した場合は、CHCR0、CHCR1 の DS ビットを 1、DL ビットを 0 に設定 (DREQn 立ち下がり検出) してください。また、バスモードは C (サイクルスチールモード) のみ設定可能です。

\*4 転送元または転送先である内蔵周辺モジュールのレジスタで許されるアクセスサイズを指定してください。

\*5 内蔵メモリからメモリマップト外部デバイスへの転送時には DACKn をライト時出力に設定し、メモリマップト外部デバイスから内蔵メモリへの転送時には DACKn をリード時出力に設定してください。

\*6 内蔵メモリから外部メモリへの転送時には DACKn をライト時出力に設定し、外部メモリから内蔵メモリへの転送時には DACKn をリード時出力に設定してください。

- \*7 外部リクエストモードでB(バーストモード)設定時は、必ずCHCR0、CHCR1のDSビットを1(エッジ検出)に設定してください。CHCR0、CHCR1のDSビットを0(レベル検出)に設定した場合の動作は保証しません。
- \*8 16バイト単位の転送は、エッジ検出設定時のみ有効です。レベル検出設定時に16バイト単位の転送をした場合の動作は保証しません。

表 11.9 に、外部リクエストモード時に設定可能なリクエストモード、バスモード、アドレスモードの組み合わせを示します。

表 11.9 外部リクエストモード時に設定可能なリクエストモード、バスモード、アドレスモードの組み合わせ一覧

リクエストモード			デュアルアドレスモード		シングルアドレスモード	
			バーストモード	サイクルスチールモード	バーストモード	サイクルスチールモード
外部リクエスト	レベル検出 *1	バイト	-		-	
		ワード	-		-	
		ロングワード	-		-	
		16バイト単位	-	-	-	-
	エッジ検出 *2	バイト				
		ワード				
		ロングワード				
		16バイト単位				

- 【注】 : 設定可能  
 - : 設定不可  
 \*1 High レベル検出、Low レベル検出とも同じです。  
 \*2 立ち上がりエッジ検出、立ち下がりエッジ検出とも同じです。

(4) バスモードとチャンネルの優先順位

あるチャンネル(たとえばチャンネル1)がバーストモードで転送中、それより高いチャンネル(たとえばチャンネル0)に転送要求が発生すると、ただちにその優先順位の高い方のチャンネル(0)が転送を開始します。チャンネル0もバーストモードの場合、チャンネル0の転送がすべて終了してから、チャンネル1が転送を続けます。また、チャンネル0がサイクルスチールモードの場合、まず、チャンネル0が1転送単位の転送を行った後、チャンネル1が転送を再開しますが、このときチャンネル1は、2転送単位の転送を行います。その後も、チャンネル0 チャンネル1 チャンネル1 チャンネル0 というようにバス権を入れ替えます。この間、チャンネル1がバーストモードなので、CPUにバス権は渡りません。

この例を図 11.12 に示します。

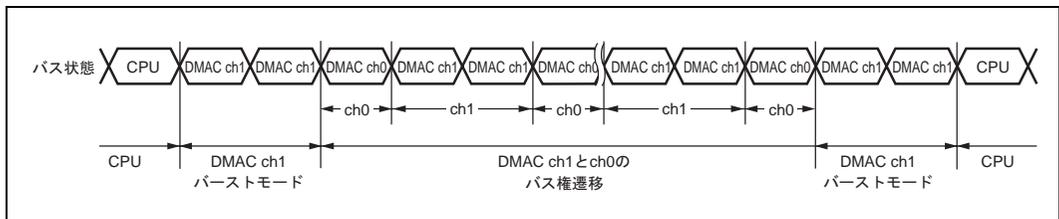


図 11.12 複数チャンネルが動作する場合のバス状態 (優先順位: チャンネル0 > チャンネル1、チャンネル1はバーストモード、チャンネル0はサイクルスチールモードが設定されている場合)

### 11.3.5 バスサイクル数

DMAC がバスマスタのときのバスサイクルのサイクル数は、CPU がバスマスタのときと同様にバーステートコントローラ (BSC) のバスコントロールレジスタ (BCR1) とウェイトコントロールレジスタ (WCR) で制御されます。詳しくは、「7. バーステートコントローラ (BSC)」を参照してください。

### 11.3.6 DMA 転送要求受け付け信号出力タイミング

DMA 転送要求受け付け信号 DACK<sub>n</sub> は、アドレスバスのチャンネルコントロールレジスタ AM ビットで指定した DMA アドレス出力に同期して出力されます。タイミングは通常、DMA アドレス出力が始まると同時にアクノリッジ信号が有効となり、そのアドレス出力が終了する 0.5 サイクル前に無効となります (図 11.13 参照)。

接続されるメモリ空間の設定により、アクノリッジ信号の出力タイミングが異なります。以下に各メモリ空間における、アクノリッジ信号の出力タイミングを示します。

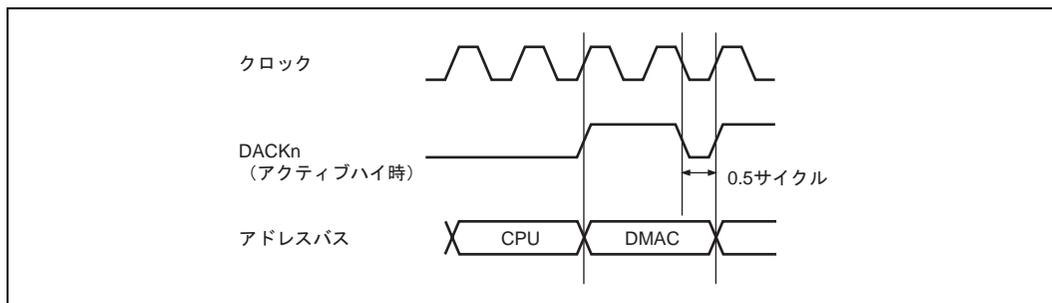


図 11.13 DACK<sub>n</sub> 出力タイミングの例

#### (1) 外部メモリが通常メモリ空間設定時のアクノリッジ信号出力

アクノリッジ信号を出力するタイミングは、AM の指定による DMA 読み出しサイクルと書き込みサイクルで同様です。DMA アドレス出力が始まると同時にアクノリッジ信号は有効となり、そのアドレス出力が終了する 0.5 サイクル前に無効となります。その間ウェイトが入り、アドレス出力が伸びると、アクノリッジ信号も伸びます。

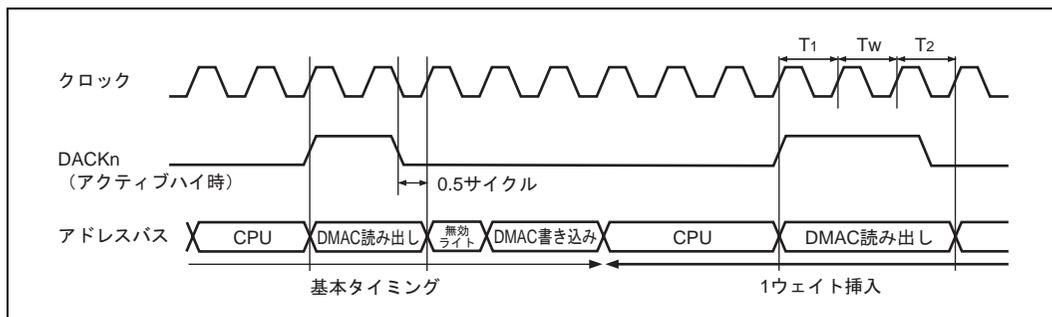


図 11.14 通常空間アクセスにおける DACK<sub>n</sub> 出力 (AM = 0)

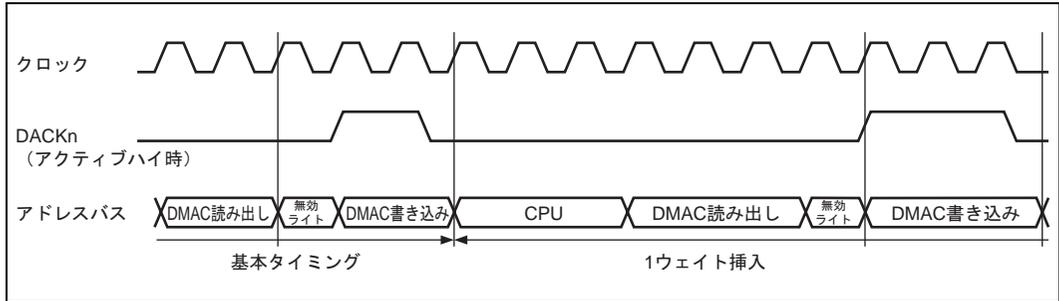


図 11.15 通常空間アクセスにおける DACKn 出力 (AM = 1)

8ビット外部デバイスや16ビット外部デバイスにロングワードアクセスをしたり、8ビット外部デバイスにワードアクセスをする場合、データをアライメントするためDMAC1回のアクセスで下位、上位のアドレスを2、4回出力します。それらのすべてのアドレスに対し、出力が始まると同時にアクノリッジ信号は有効となり、そのアドレス出力が終了する0.5サイクル前に無効となります。SDRAM、DRAM、パーストROMでもデータアライメントのため、1回のアクセスに複数のアドレスが出力される場合、それぞれのアドレスに対してアクノリッジ信号が出力されます。

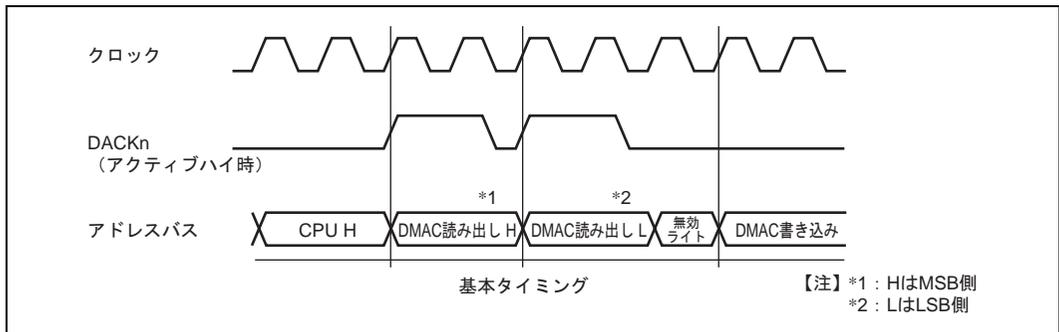


図 11.16 通常空間アクセスにおける DACKn 出力  
(AM = 0、16ビット外部デバイスにロングワードアクセスをした場合)

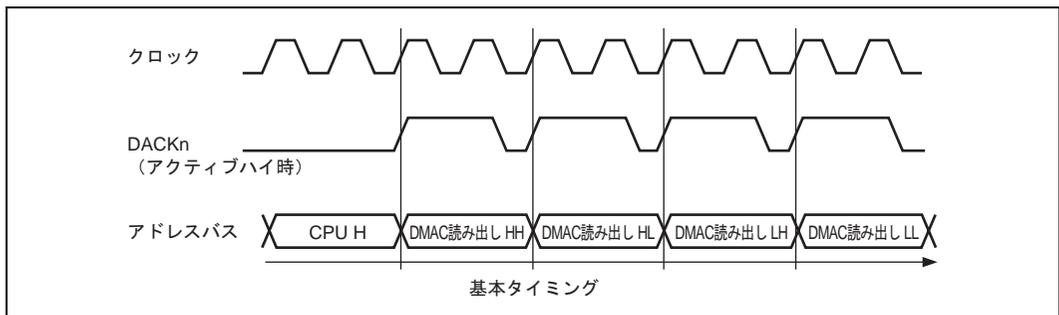


図 11.17 通常空間アクセスにおける DACKn 出力  
(AM = 0、8ビット外部デバイスにロングワードアクセスをした場合)

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

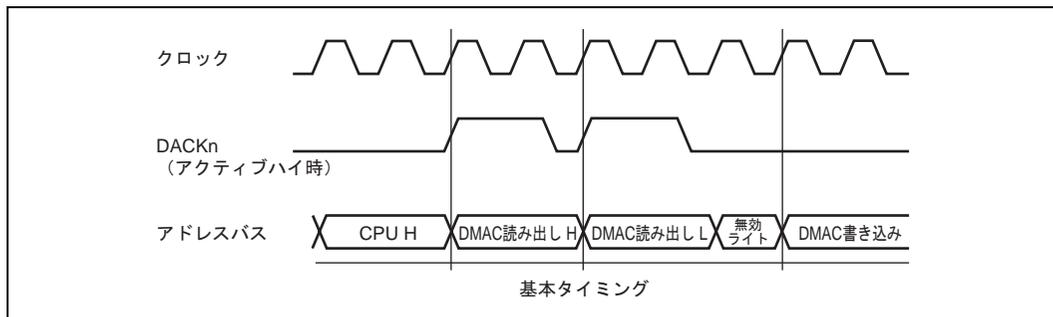


図 11.18 通常空間アクセスにおける DACKn 出力  
(AM = 0、8 ビット外部デバイスにワードアクセスをした場合)

### (2) 外部メモリがシンクロナス DRAM の設定時のアクノリッジ信号出力

外部メモリがシンクロナス DRAM に設定されているとき、DACKn の出力タイミングは、DMA アドレスが始まると同時に有効となり、そのアドレス出力が終了すると同時に無効となります。

外部メモリをシンクロナス DRAM のオートプリチャージと設定した場合、AM = 0 のときには、DMAC 読み出しのロウアドレス、リードコマンド、ウェイト、リードアドレスすべてにまたがりアクノリッジ信号を出力します。またシンクロナス DRAM のリードはバーストしかないので、シングルリード時は無効アドレスが出力されますが、それと同タイミングでもアクノリッジ信号は出力されます。このとき、無効リードに続くライトアドレス出力のタイミングまでアクノリッジ信号は延長されます。なお、シンクロナス DRAM のバーストリードは 16 バイト単位の転送時に行われます。16 バイト単位の転送はオートリクエストモード、および外部リクエストモード・エッジ検出時のみ有効ですので、内蔵周辺モジュールリクエスト、および外部リクエストモード・レベル検出を使用する場合は、バイト単位、ワード単位、ロングワード単位のいずれかを設定してください。内蔵周辺モジュールリクエスト、および外部リクエストモード・レベル検出を使用する場合に 16 バイト単位を設定した場合の動作は保証しません。

AM = 1 のときには、DMAC 書き込みのロウアドレスとコラムアドレスにまたがってアクノリッジ信号が出力されます。

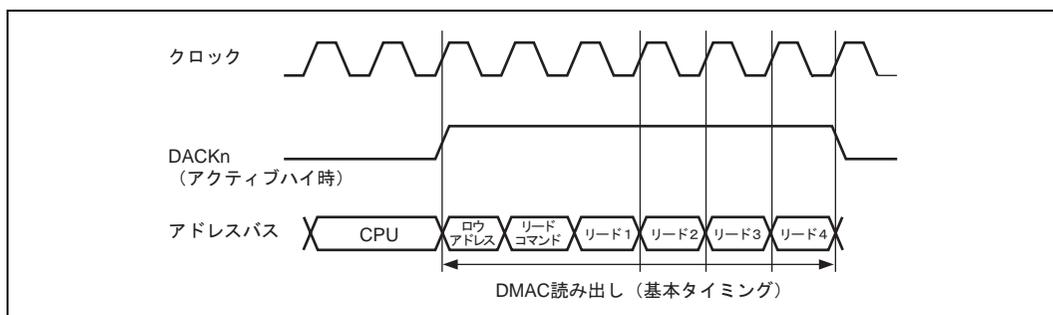


図 11.19 シンクロナス DRAM バーストリード (オートプリチャージ) における DACKn 出力 (AM = 0)

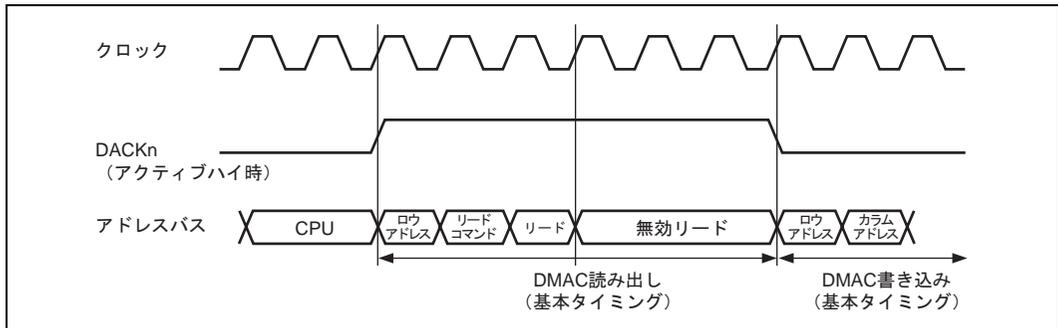


図 11.20 シンクロナス DRAM シングルリード (オートプリチャージ) における DACKn 出力 (AM = 0)

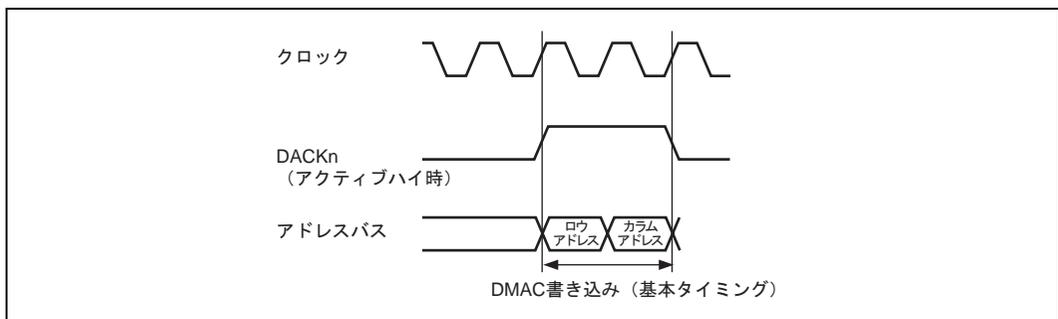


図 11.21 シンクロナス DRAM ライト (オートプリチャージ) における DACKn 出力 (AM = 1)

外部メモリをシンクロナス DRAM のバンクアクティブと設定した場合のバーストリードでは、前回のアドレス出力と同一のロウアドレスのとき、リードコマンド、ウェイト、リードアドレスすべてにまたがりアクノリッジ信号を出力します。また、前回と異なるロウアドレスのとき、プリチャージ、ロウアドレス、リードコマンド、ウェイト、リードアドレスすべてにまたがりアクノリッジ信号を出力します。

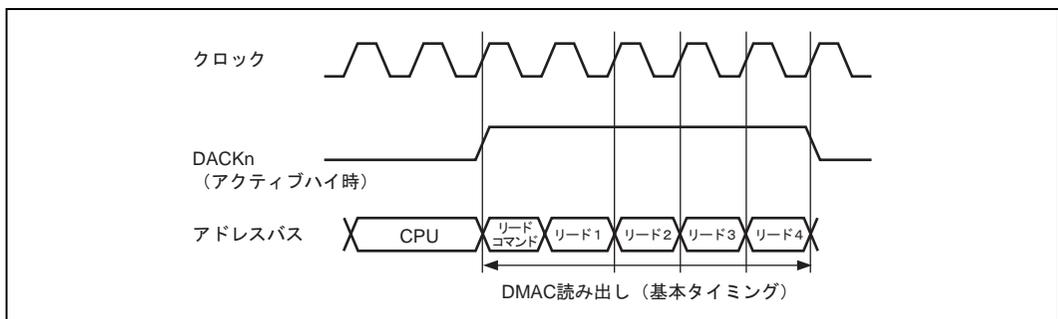


図 11.22 シンクロナス DRAM バーストリード (バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 0)

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

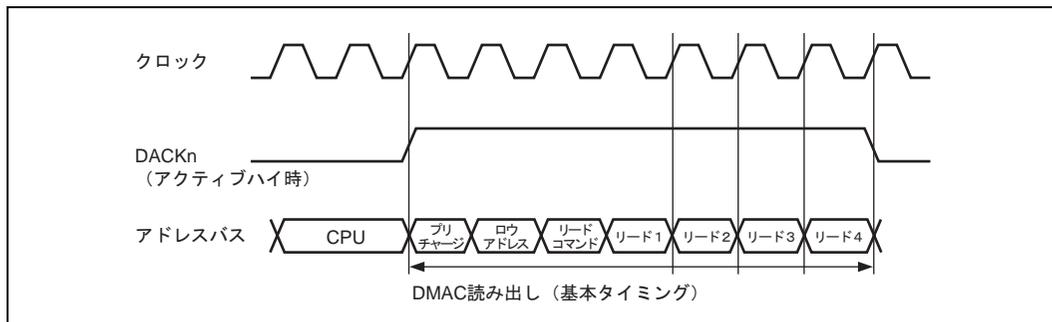


図 11.23 シンクロナス DRAM バーストリード  
(バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 0)

外部メモリをシンクロナス DRAM のバンクアクティブと設定した場合のシングルリードでは、前回のアドレス出力と同一のロウアドレスのとき、リードコマンド、ウェイト、リードアドレスすべてにまたがりアクノリッジ信号を出力します。また前回と異なるロウアドレスのとき、プリチャージ、ロウアドレス、リードコマンド、ウェイト、リードアドレスすべてにまたがりアクノリッジ信号を出力します。また、シンクロナス DRAM のリードはバーストしかないため、シングルリード時は無効リードが出力されますが、それと同タイミングでもアクノリッジ信号は出力されます。このとき、無効リードに続くライトアドレス出力のタイミングまでアクノリッジ信号は延長されます。

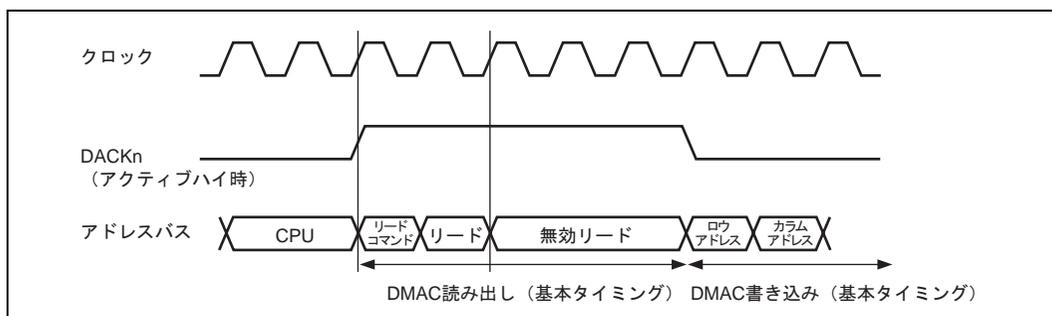


図 11.24 シンクロナス DRAM シングルリード  
(バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 0)

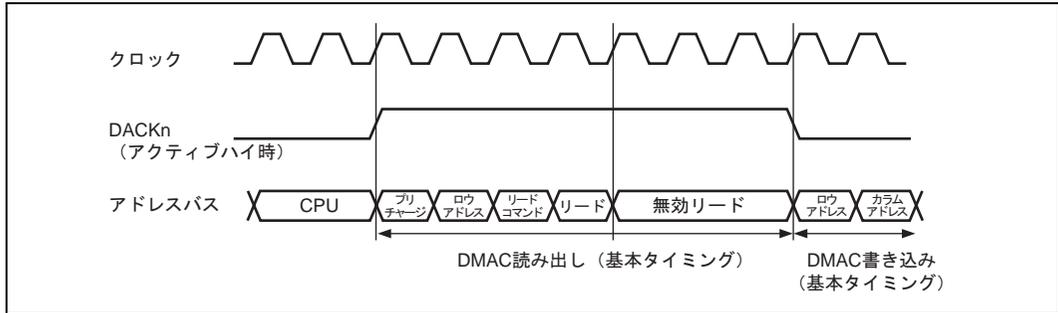


図 11.25 シンクロナス DRAM シングルリード  
(バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 0)

外部メモリをシンクロナス DRAM のバンクアクティブと設定した場合のライト時では、前回のアドレス出力と同一のロウアドレスのとき、ウェイト、カラムアドレスにまたがりアクノリッジ信号を出力します。また前回と異なるロウアドレスのとき、プリチャージ、ロウアドレス、ウェイト、カラムアドレスすべてにまたがりアクノリッジ信号を出力します。

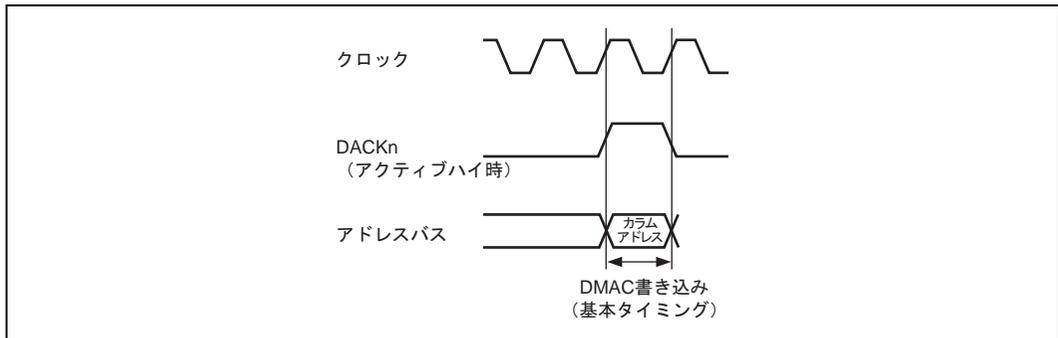


図 11.26 シンクロナス DRAM ライト  
(バンクアクティブ、同一ロウアドレス) における DACKn 出力 (AM = 1)

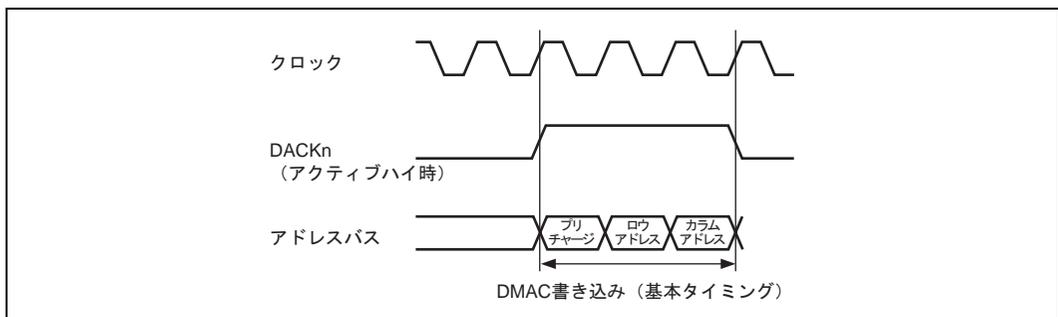


図 11.27 シンクロナス DRAM ライト  
(バンクアクティブ、異なるロウアドレス) における DACKn 出力 (AM = 1)

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### ・シンクロナス DRAM1 サイクルライト

シンクロナス DRAM に対し、1 サイクルライトを行う場合 DACKn 信号はクロックの立ち上がり同期になります。また、リクエスト信号は、DACKn 出力中にクロックの High 期間で要求を受け付けます。

転送幅	バイト・ワード・ロングワード 転送 *1	DREQn 検出方式	レベル検出
転送バスモード	サイクルスチールモード *2	DACKn 出力タイミング	ライト DACK
転送アドレスモード	シングルモード	バスサイクル	基本バスサイクル

【注】 \*1 16 バイト単位は設定しないでください。16 バイト単位を設定した場合の動作は保証しません。

\*2 DREQ をレベル検出した場合は、必ずサイクルスチールモードに設定してください。

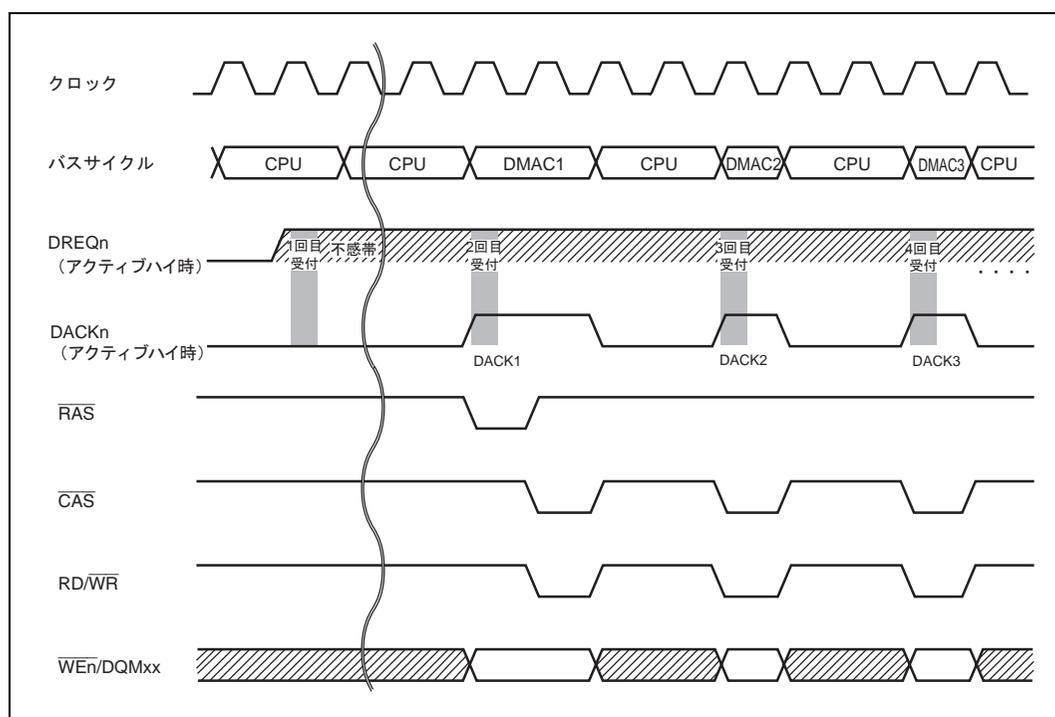


図 11.28 (a) シンクロナス DRAM1 サイクルライトタイミング

転送幅	バイト・ワード・ロングワード 転送	DREQn 検出方式	エッジ検出 *
転送バスモード	バーストモード	DACKn 出力タイミング	ライト DACK
転送アドレスモード	シングルモード	バスサイクル	基本バスサイクル

【注】 \* 転送バスモードをバーストモードに設定した場合は、必ずエッジ検出に設定してください。

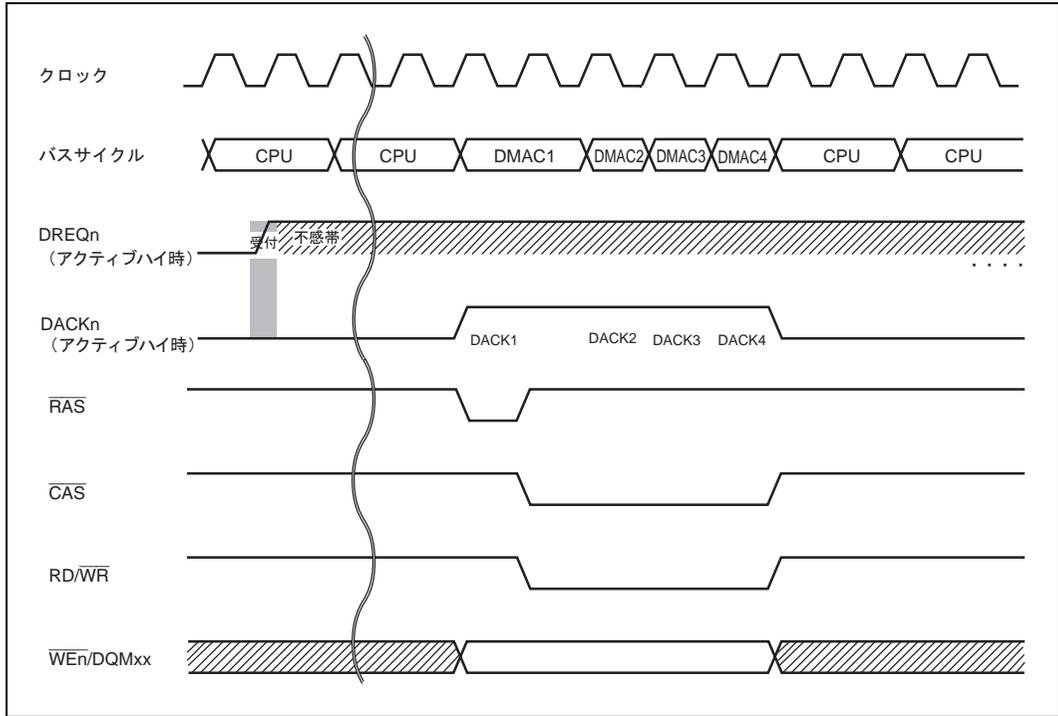


図 11.28 (b) シンクロナス DRAM1 サイクルライトタイミング

(3) 外部メモリが DRAM の設定時のアクノリッジ信号出力

外部メモリを DRAM に設定した場合、リード、ライト共にロウアドレスが出力されるとき、ロウアドレスとカラムアドレスにまたがってアクノリッジ信号が出力されます。

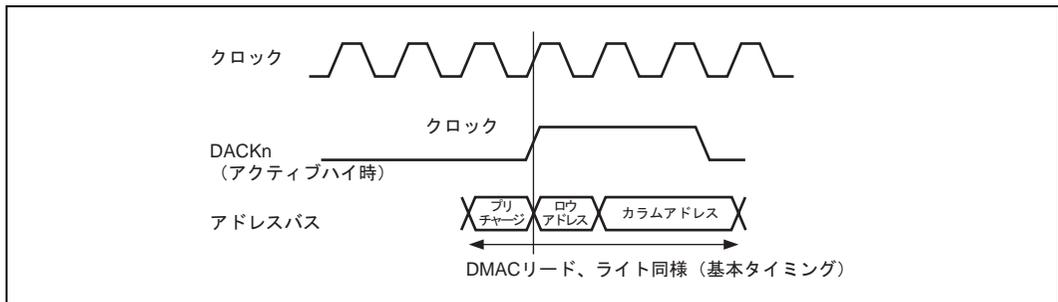


図 11.29 DRAM ノーマルアクセス時の DACKn 出力 (AM = 0、1 同様)

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

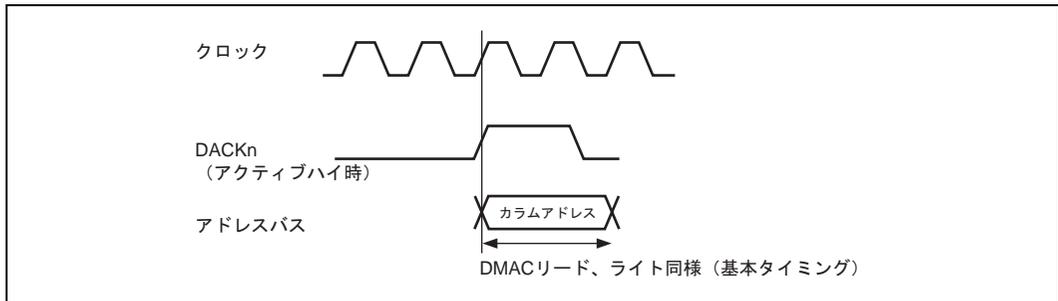


図 11.30 DRAM バーストアクセス時 (同一ロウアドレス) の DACKn 出力 (AM = 0、1 同様)

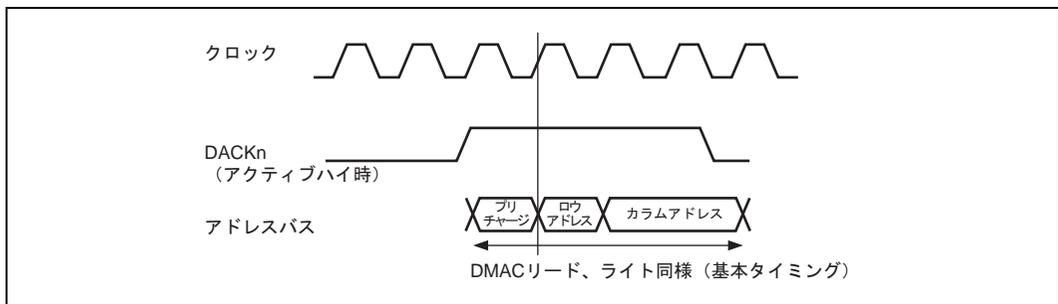


図 11.31 DRAM バーストアクセス時 (異なるロウアドレス) の DACKn 出力 (AM = 0、1 同様)

### (4) 外部メモリがバーストROMの設定時のアクノリッジ信号出力

外部メモリをバーストROMに設定した場合、DMAアドレス(デュアルライトは不可)と同期してアクノリッジ信号が出力されます。

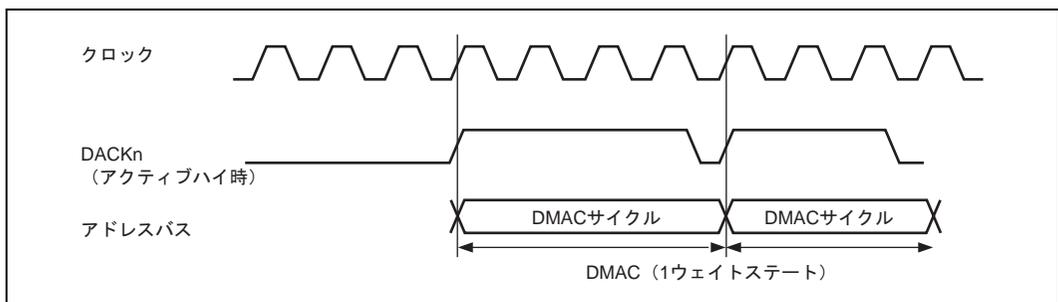


図 11.32 バーストROMニブルアクセス時のDACKn出力

### 11. 3. 7 DREQn 端子の入力検出タイミング

外部リクエストモードにおいて、DREQn 端子は、通常クロックパルス (CKIO) の立ち上がりエッジで信号検出が行われます。リクエストが検出されると、最も早い場合で 4 サイクル後に DMAC のバスサイクルが発生し、DMA 転送が行われます。リクエストが検出された後、次の入力検出タイミングは、バスモードやアドレスモード、DREQn 入力検出方法や接続するメモリによって異なります。

(1) サイクルスチールモードでの DREQn 端子の入力検出タイミング

サイクルスチールモードでは、DREQn 端子から 1 度リクエストが検出されると、次の外部バスサイクルでの DACKn 信号出力までリクエスト信号の検出は行われません。また、サイクルスチールモードでは、DACKn 信号出力からリクエストが検出されるまでリクエスト検出を行います。

1 度受け付けたリクエストは、途中でキャンセルできません。

リクエストが検出されてから次に検出可能となるまでのタイミングを以下に示します。

(a) サイクルスチールモード・エッジ検出

エッジ検出を用いて転送制御を行う場合、図 11.33 のように DREQn、DACKn のハンドシェークを行い、DREQn と DACKn が 1 対 1 となるように DREQn の入力制御を行ってください。対応する DACKn が出力される前に DREQn を入力した場合の動作は保証しません。

DACKn 信号が複数回出力される場合は、入力した DREQn に対する最初の DACKn 信号が次のリクエスト受け付け開始タイミングとなり、それ以降毎クロックエッジをサンプリングします。

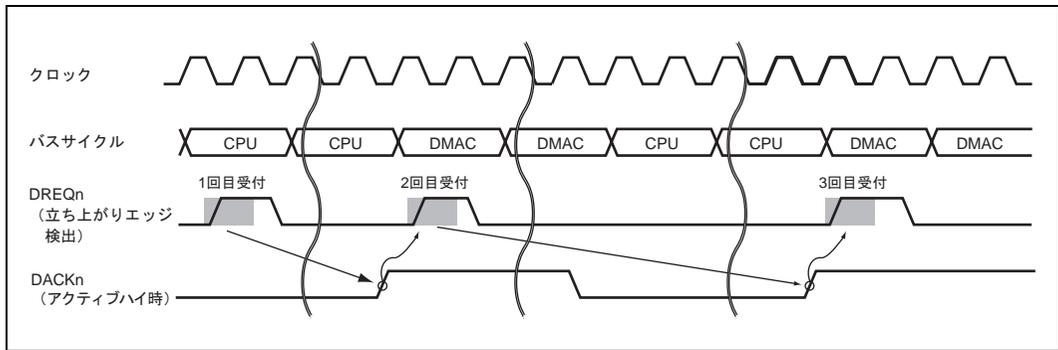


図 11.33 DREQn、DACKn のハンドシェーク

・ エッジ検出 1/2/4 バイト転送の場合

転送幅	バイト・ワード・ロングワード	DREQn 検出方式	エッジ検出
転送バスモード	サイクルスチールモード	DACKn 出力タイミング	リード DACK・ライト DACK
転送アドレスモード	デュアル・シングルモード	バスサイクル	基本バスサイクル

11. ダイレクトメモリアクセスコントローラ (DMAC)

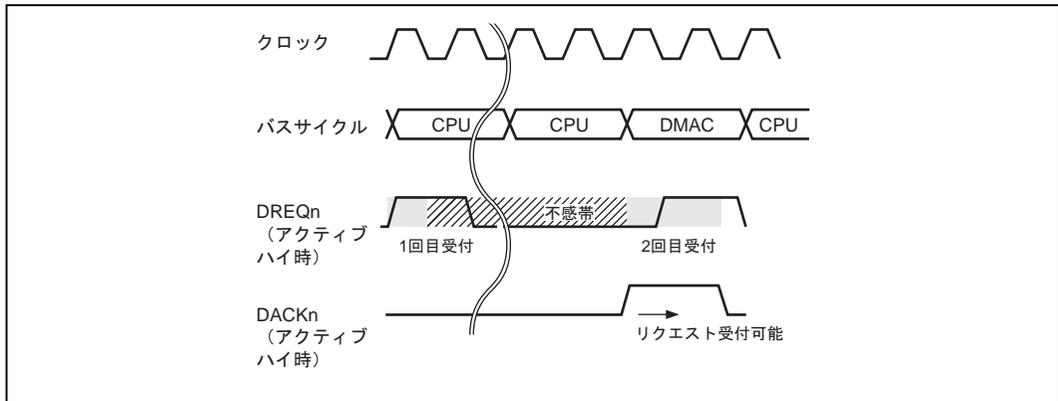


図 11.34 サイクルスチールモード・エッジ検出設定時の DREQn 端子入力検出タイミング

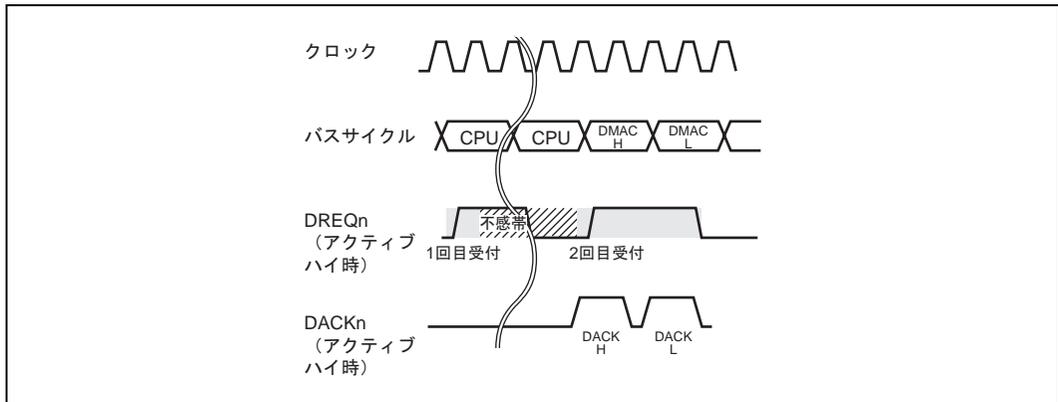


図 11.35 16 ビット外部デバイス接続時 (エッジ検出)

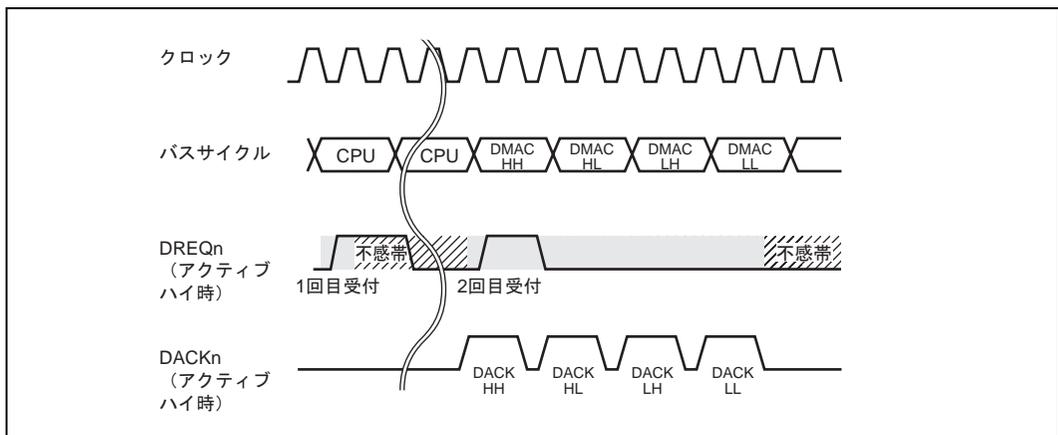


図 11.36 8 ビット外部デバイス接続時 (エッジ検出)

## ・ サイクルスチールモード・エッジ検出 16 バイト転送の場合

16 バイト転送の場合、最初のリクエストは 1 回目の転送要求になり、次の要求受け付けで 2 回目を受け付けます。以降同様に 3 回目 4 回目の要求を受け付けます。

転送幅	16 バイト転送	DREQn 検出方式	エッジ検出
転送バスモード	サイクルスチールモード	DACKn 出力タイミング	リード DACK・ライト DACK
転送アドレスモード	デュアル・シングルモード	バスサイクル	基本バスサイクル

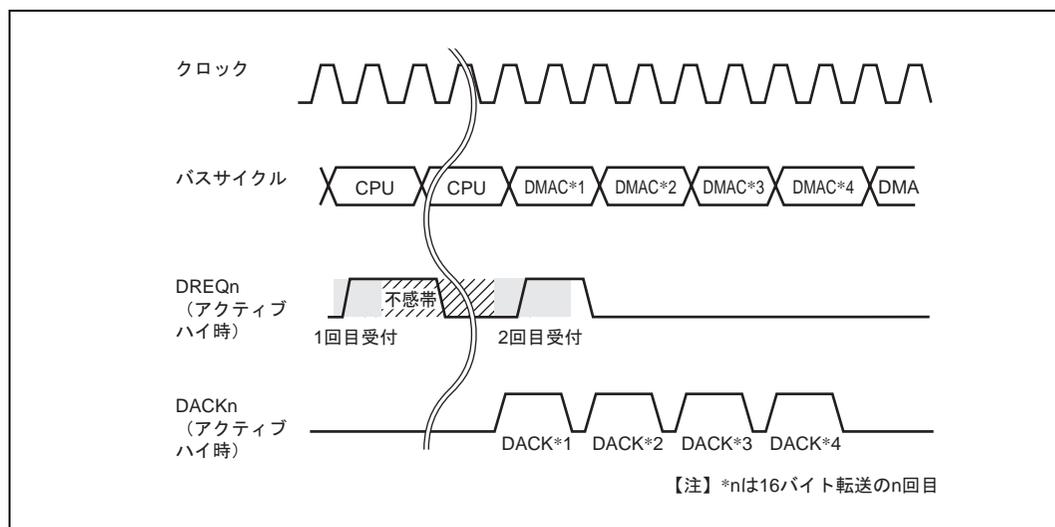


図 11.37 サイクルスチール・エッジ検出 16 バイト転送設定時の DREQn 端子入力検出タイミング

## (b) サイクルスチールモード・レベル検出

レベル検出モードの場合も、リクエストを受け付けた場合これをキャンセルできません。

転送幅	バイト・ワード・ロングワード*	DREQn 検出方式	レベル検出
転送バスモード	サイクルスチールモード	DACKn 出力タイミング	リード DACK・ライト DACK
転送アドレスモード	デュアル・シングルモード	バスサイクル	基本バスサイクル

【注】 \* 16 バイト単位は設定しないでください。16 バイト単位に設定した場合の動作は保証しません。

11. ダイレクトメモリアクセスコントローラ (DMAC)

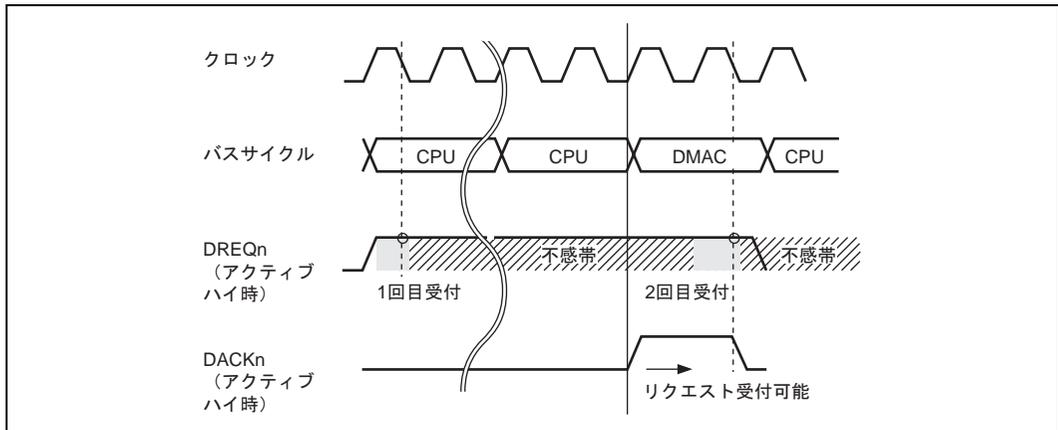


図 11.38 サイクルスチールモード・レベル検出  
バイト・ワード・ロングワード設定時の DREQn 端子入力検出タイミング

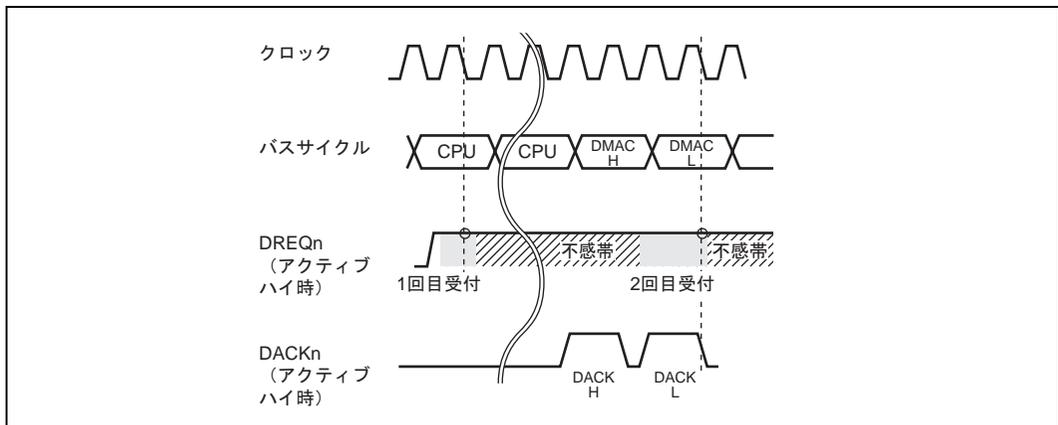


図 11.39 16ビット外部デバイス接続時 (レベル検出)

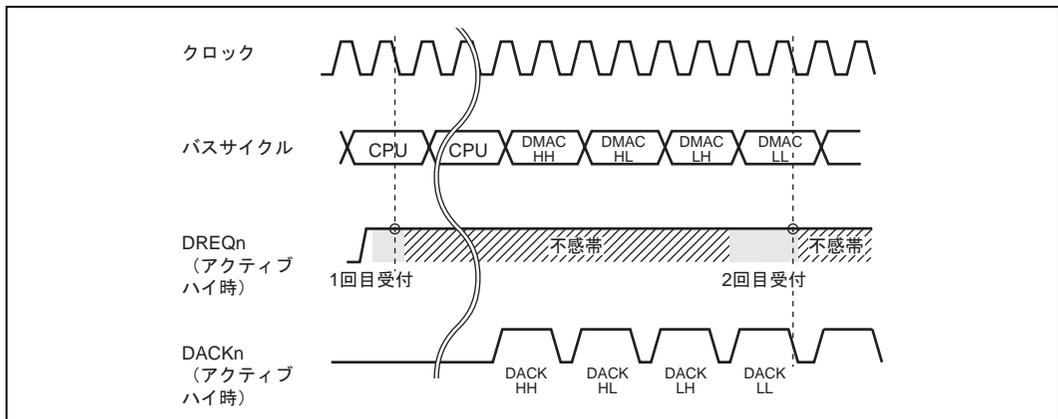


図 11.40 8ビット外部デバイス接続時 (レベル検出)

## (2) バーストモードでの DREQn 端子の入力検出タイミング

バーストモードでは、DREQn 入力はエッジ検出のみ有効です。DREQn 入力をレベル検出に設定した場合の動作は保証しません。

DREQn 入力をエッジで検出する場合、一度リクエストが検出されると、その後は DREQn 端子の状態にかかわらず転送終了条件が満たされるまで DMA 転送が継続されます。この間リクエスト検出は行われません。転送終了後に転送開始条件がそろると再び 1 サイクルごとにリクエスト検出が行われます。

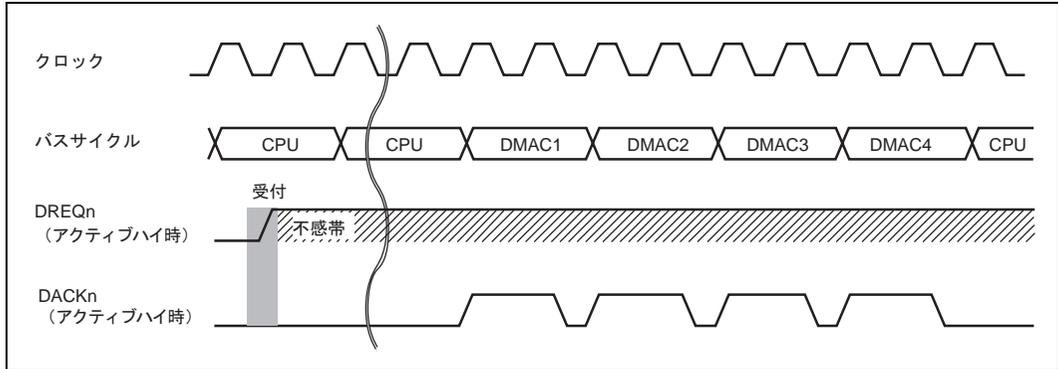


図 11.41 バーストモード・エッジ検出設定時の DREQn 端子入力検出タイミング

## 11.3.8 DMA 転送終了

DMA 転送の終了条件は、1 チャンネルずつの終了と 2 チャンネル同時の終了とで異なります。

## (1) チャンネルごとの転送終了条件

以下のいずれかの条件が満たされると、対応するチャンネルだけが転送を終了します。

- DMA トランスファカウントレジスタ (TCR) の値が 0 になる
- DMA チャンネルコントロールレジスタ (CHCR) の DMA イネーブルビット (DE) が 0 になる

## (a) TCR = 0 による転送終了

TCR の値が 0 になると、対応するチャンネルの DMA 転送が終了し、CHCR のトランスファエンドフラグビット (TE) がセットされます。このときインタラプトイネーブルビット (IE) がセットされていると、CPU に DMAC 割り込み (DEI) が要求されます。

16 バイト転送時には、TCR に転送回数 × 4 を設定してください。間違えた値を設定したときの動作は保証しません。

なお、16 バイト転送はオートリクエスト時、および外部リクエストモード・エッジ検出時のみ有効です。外部リクエスト・レベル検出、内蔵周辺モジュールリクエストを使用する場合は 16 バイト転送は設定しないでください。

## (b) CHCR の DE = 0 による転送終了

CHCR の DMA イネーブルビット (DE) をクリアすると、対応するチャンネルの DMA 転送が中断されます。この場合には TE ビットはセットされません。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

### (2) 全チャネル同時の転送終了

以下のいずれかの条件が満たされると、すべてのチャネルが同時に転送を終了します。

- DMA オペレーションレジスタ (DMAOR) の NMI フラグビット (NMIF) またはアドレスエラーフラグビット (AE) が 1 になる
- DMAOR の DMA マスタイネーブルビット (DME) が 0 になる

#### (a) DMAOR の NMIF = 1 または AE = 1 による転送終了

NMI 割り込みまたは DMAC によるアドレスエラーが発生して、DMAOR の NMIF ビットまたは AE ビットが 1 になると、すべてのチャネルの DMA 転送が中断されます。中断による直前の転送による DMA ソースアドレスレジスタ (SAR)、DMA デスティネーションレジスタ (DAR)、DMA トランスファカウントレジスタ (TCR) の更新は行われます。この転送が最終転送の場合は、TE = 1 となり転送終了となります。NMI 割り込み例外処理、アドレスエラー例外処理後に転送を再開するには、対応するフラグをクリアします。このとき DE ビットが 1 にセットされていると、そのチャネルの転送が再開されてしまうので、再開させたくないチャネルがあれば、DE ビットは 0 にしておく必要があります。

デュアルアドレスモードの場合、最初のリードサイクルでアドレスエラーが発生しても、それに引き続くライトサイクルは実行されてから DMA 転送が中断されます。この最後の転送による SAR、DAR、TCR の更新は行われます。

#### (b) DMAOR の DME = 0 による転送終了

DMAOR の DME ビットを 0 にクリアすると、全チャネルの DMA 転送が、現在実行中の転送終了後に強制的に中断されます。この転送が最終転送の場合は、TE = 1 となり転送終了となります。

### 11.3.9 $\overline{\text{BH}}$ 端子の出力タイミング

#### (1) $\overline{\text{BH}}$ 新設目的

外部バスとして、PCI バスを接続した場合、本 LSI は PCI バスインタフェースを持たないため、必ず外部にグルーロジックが必要となります。

PCI バスは、バースト転送がメインで、単発の転送ではパフォーマンスが上がりません。

そこで、外部のグルーロジックは、バースト転送可能か否かを現在のアドレスと次のアドレスを比較することにより判断する必要があります。しかし、アドレス比較を行うには外部のグルーロジックが大きくなり、またタイミングも間に合わなくなる可能性があります。

これらの問題を解決するために  $\overline{\text{BH}}$  を新設し、現在のアクセスがバースト転送可能であることを外部のグルーロジックに通知し PCI バス接続対応を行います。

#### (2) $\overline{\text{BH}}$ 使用時のレジスタ設定

$\overline{\text{BH}}$  は、16 バイト転送モードを選択した場合にのみ出力します。ただし、アクセスがシンクロナス DRAM あるいは DRAM に対して行われる場合は出力されません。なお、16 バイト転送モードを使用する場合は、必ずオートリクエストモード、もしくは外部リクエストモード・エッジ検出に設定してください。外部リクエストモード・レベル検出、内蔵周辺モジュールリクエストモードに設定した場合の動作は保証しません。

$\overline{\text{BH}}$  を使用する場合は、必ず CHCR0 あるいは CHCR1 の設定を図 11.42 に示す通りにしてください。図 11.42 に示す通りに CHCR0 あるいは CHCR1 を設定しない場合は、 $\overline{\text{BH}}$  は出力されません。

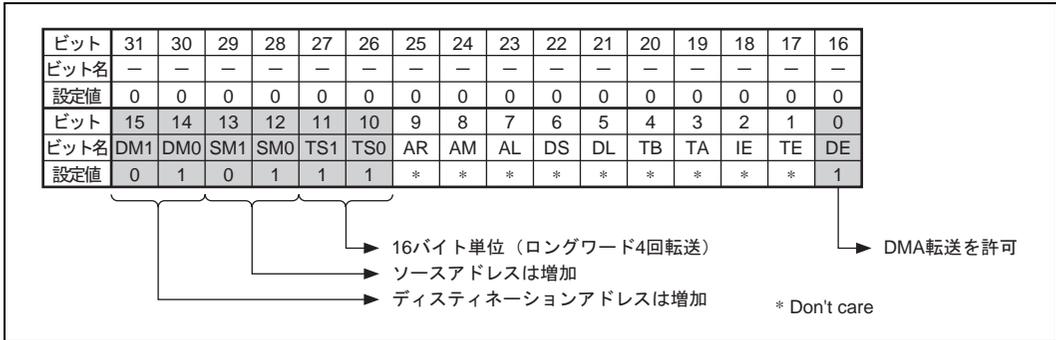


図 11.42 BH 使用時のレジスタ設定

(3) BH 概略タイミング

図 11.43 に、BH 出力の概略タイミングを示します。

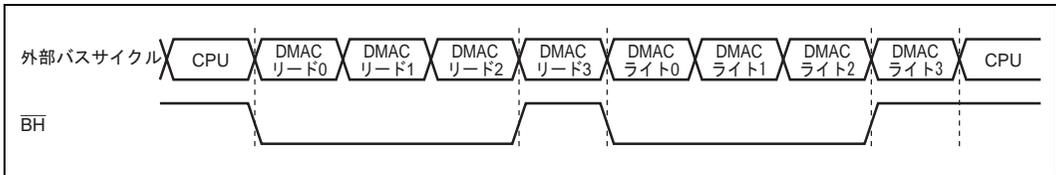


図 11.43 BH 出力の概略タイミング

11.4 使用例

11.4.1 内蔵 SCIF と外部メモリとの DMA 転送例

内蔵 FIFO 付シリアルコミュニケーションインタフェース (SCIF) 受信データを、DMAC のチャネル 1 を使って外部メモリに転送する例を考えます。表 11.10 に転送条件と各レジスタの設定値を示します。

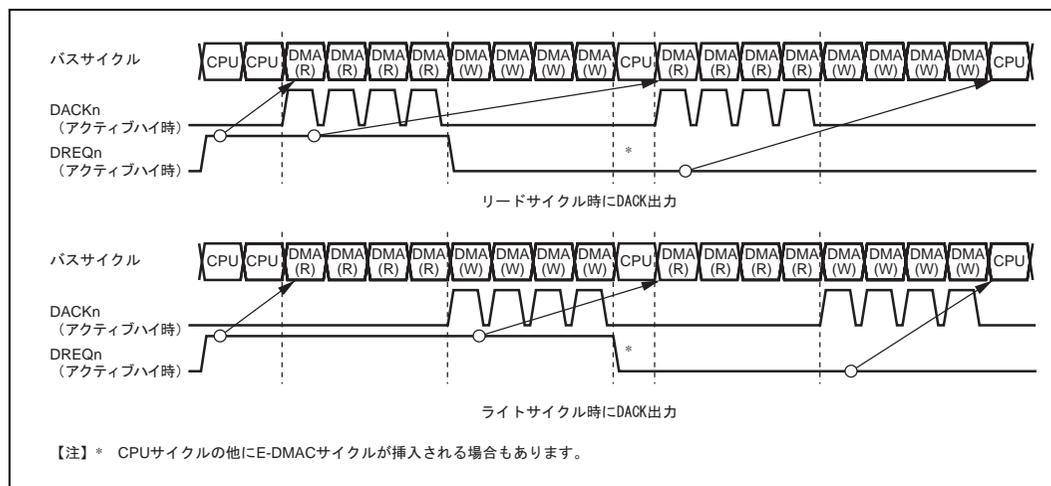
表 11.10 内蔵 SCIF と外部メモリ間転送の転送条件とレジスタ設定値

転送条件	レジスタ	設定値
転送元: 内蔵 SCIF の SCFRDR1	SAR1	H'FFFFFFCC
転送先: 外部メモリ (ワード空間)	DAR1	転送先アドレス
転送回数: 64 回	TCR1	H'0040
転送先アドレス: 増加	CHCR1	H'4045
転送元アドレス: 固定		
バスモード: サイクルスチール		
転送単位: バイト		
転送終了時に DEI 割り込み要求発生 DE = 1		
チャネル優先順位: 固定 (0>1) DME = 1	DMAOR	H'0001
転送要求元 (転送要求信号): SCIF (RXI)	DRCR1	H'05

【注】 \* SCIF 側の設定で割り込みイネーブルと、CPU 割り込みレベルに注意してください。

## 11.5 使用上の注意

- (1) DRCR0 と DRCR1 は、バイト単位のアクセス、それ以外のレジスタはロングワード単位でアクセスしてください。
- (2) CHCR0、CHCR1 および DRCR0、DRCR1 を書き換える場合は、指定チャンネル CHCR の DE ビットを 0 にするか、DMAOR の DME ビットを 0 にしてから書き換えてください。
- (3) DMAC が動作していないときに、NMI 割り込みが入力されても、DMAOR の NMIF ビットはセットされません。
- (4) キャッシュメモリを DMAC がアクセスすることはできません。
- (5) 周波数変更時、もしくはスタンバイモードにするときは、DMAOR の DME ビットを 0 にし、DMAC を停止させてから行ってください。
- (6) 内蔵周辺モジュールのうち DMAC、BSC、UBC、E-DMAC、EtherC をアクセスしないでください。
- (7) キャッシュ (アドレスアレイ、データアレイ、連想バージ領域) はアクセスしないでください。
- (8) シングルアドレスモードでリクエスト信号をレベル検出する場合、DACKn 出力前にリクエスト信号を検出する場合がありますので御注意ください。
- (9) E が 31.25MHz を超えるときは、8 ビットバス幅時のワード、ロングワードアクセス、および 16 ビットバス幅時のロングワードアクセスで、通常空間に対し DACKn 出力を伴う転送は行わないでください。
- (10) 内蔵周辺モジュールからの DMA 転送要求信号によって DMA 転送を行った場合、DMA 転送による内蔵周辺モジュールからの DMA 転送要求信号のクリアが、その内蔵周辺モジュールから出される次の転送要求信号に間に合わないときは、以後の DMA 転送ができなくなる場合があります。
- (11) サイクルスチール、16 バイト転送、デュアルアドレスモードを使用するときには以下の制限があります。
  - (a) 外部リクエスト、レベル検出を設定した場合、DMA 転送開始後は DACKn がアクティブにならないサイクルでは、DREQn を入力しないでください。



(b) 外部リクエスト、DREQ エッジ検出の場合、連続的に DREQ<sub>n</sub> を入力すると CPU サイクルが挿入されず、DMAC が連続して動作します。

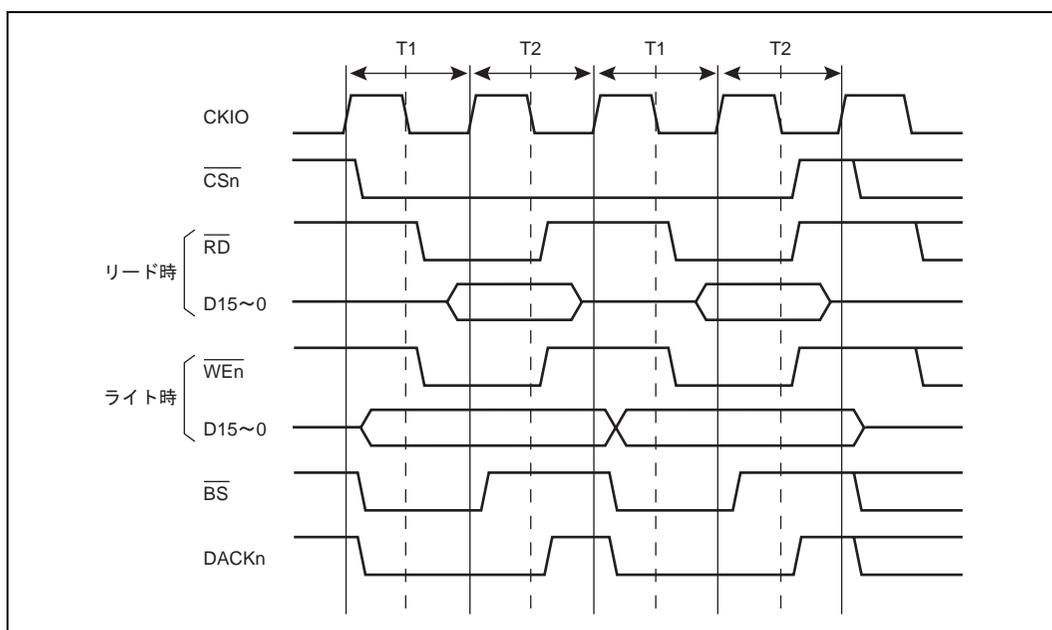
ただし、DREQ<sub>n</sub> からのリクエストがない場合には CPU サイクルが始まります。

(12) DACK<sub>n</sub> 出力に関する注意事項

【不具合内容】

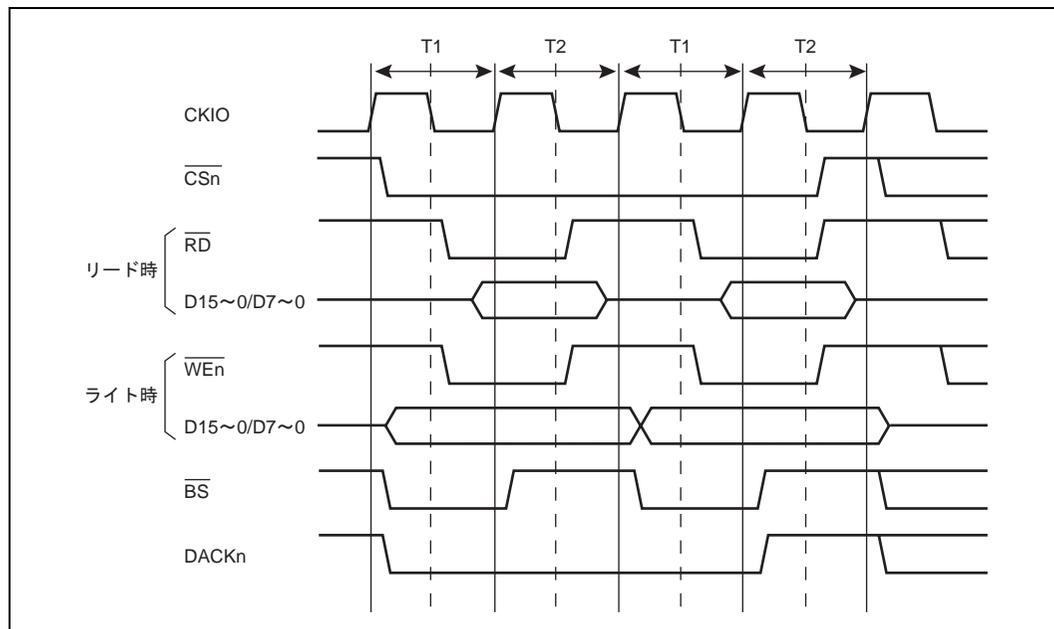
16 ビットバス幅の通常空間に対する 16 バイト単位 / ロングワード単位の DMA 転送、あるいは、8 ビットバス幅の通常空間に対する 16 バイト単位 / ロングワード単位 / ワード単位の DMA 転送を実行した場合、DACK<sub>n</sub>(n=0,1) が本来の仕様で出力されない場合があります。

・ 本来の波形

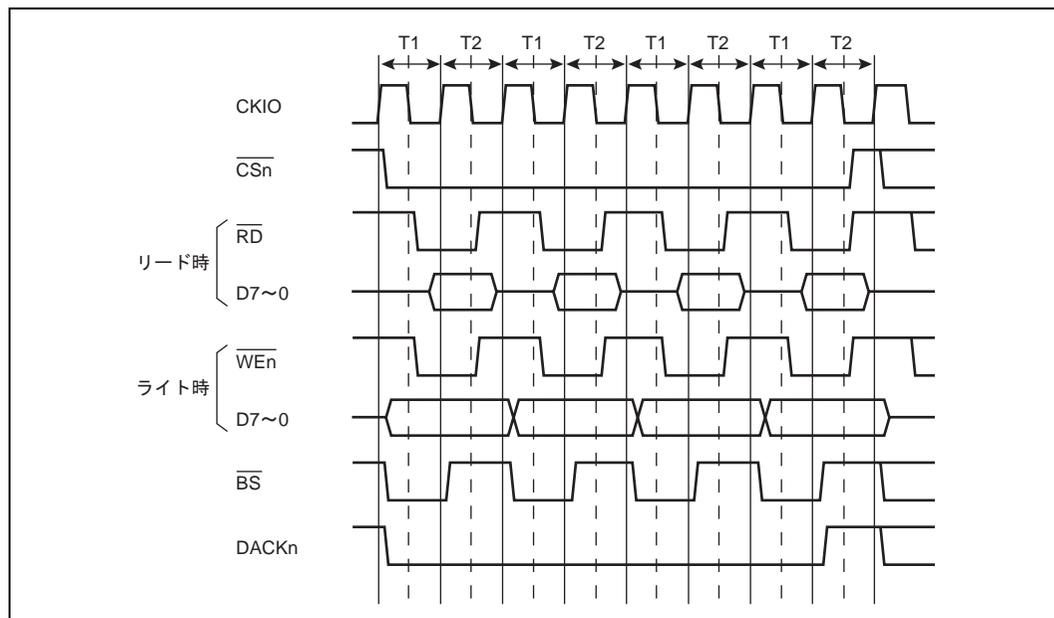


## 11. ダイレクトメモリアクセスコントローラ (DMAC)

### ・ 不具合波形 (1)



### ・ 不具合波形 (2)



・ DACKn(n=0,1) のアサートタイミングおよび回数

	16バイト単位	ロングワード単位	ワード単位
16ビットバス幅	不具合波形 (1) × 4回	不具合波形 (1) × 1回	——
8ビットバス幅	不具合波形 (2) × 4回	不具合波形 (2) × 1回	不具合波形 (1) × 1回

【発生条件】

以下の (1) から (3) のすべての条件が成立した場合、DACKn(n=0,1) が本来の仕様ではないタイミングで出力されます。

- (1)  $I : E = 1 : 1$  で使用している。
- (2) 通常空間、あるいは、バースト ROM 空間に対する DMA 転送を実行している。
- (3) 16 ビットバス幅への 16 バイト / ロングワード転送、あるいは、8 ビットバス幅への 16 バイト / ロングワード / ワード転送のようなマルチバスサイクルが発生する DMA 転送を実行している。

【回避方法】

通常空間、あるいは、バースト ROM 空間に対する DMA 転送を実行する場合は、以下の (1) から (5) のいずれかで使用してください。

- (1) クロック比  $I : E = 1 : 1$  以外で使用してください。
- (2) 32 ビットバス幅で使用してください。
- (3) 16 ビットバス幅で使用する場合は、DMA 転送の単位をワード単位、あるいは、バイト単位に設定してください。
- (4) 8 ビットバス幅で使用する場合は、DMA 転送の単位をバイト単位に設定してください。

(13) DMAC チャンネル 1 の内蔵周辺モジュールリクエストによる DMA 転送不実行

【不具合内容】

DMAC チャンネル 1 の内蔵周辺モジュールリクエストによる DMA 転送不実行

SH7615 内蔵 DMAC のチャンネル 0 をサイクルスチールモード、チャンネル 1 を内蔵周辺モジュールリクエストに設定して、DMA 転送を実行した場合、DMAC チャンネル 1 の転送が実行されない場合があります。

【発生条件】

DMAC チャンネル 1 の内蔵周辺モジュールリクエストによる DMA 転送不実行の発生条件

以下の (1) から (4) のすべての条件が成立した場合、DMAC チャンネル 1 の内蔵周辺モジュールリクエストによる DMA 転送不実行が発生します。

- (1) DMAC チャンネル 0、チャンネル 1 とともにイネーブル状態にしている。
- (2) DMAC チャンネル 0 をサイクルスチールモードに設定している。
- (3) DMAC チャンネル 1 をサイクルスチールモード、デュアルアドレスモード、内蔵周辺モジュールリクエストに設定している。
- (4) DMAC のプライオリティモードをラウンドロビンモードに設定している。

【回避方法】

DMAC チャンネル 1 の内蔵周辺モジュールリクエストによる DMA 転送不実行に関する回避方法

SH7615 内蔵 DMAC のチャンネル 0、チャンネル 1 とともに使用して、チャンネル 0 をサイクルスチールモード、チャンネル 1 をサイクルスチールモード、デュアルアドレスモード、内蔵周辺モジュールリクエストに設定して DMA 転送を実行する場合は、以下の条件で使用してください。

- (1) DMAC のプライオリティモードを優先順位固定モードに設定してください。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

### (14) DMAC の外部リクエストによる内蔵メモリ、内蔵周辺モジュール間 DMA 転送不実行

#### 【不具合内容】

DMAC の外部リクエストによる内蔵メモリ、内蔵周辺モジュール間 DMA 転送不実行  
SH7615 内蔵 DMAC を外部リクエスト (DREQ)、サイクルスチールモードに設定して、内蔵メモリ 内蔵周辺モジュール間、あるいは内蔵周辺モジュール 内蔵周辺モジュール間の DMA 転送を実行した場合、2 回目以降の DREQ 入力で DMA 転送が実行されない場合があります。

#### 【発生条件】

DMAC の外部リクエストによる内蔵メモリ、内蔵周辺モジュール間 DMA 転送不実行の発生条件  
以下の (1) から (3) のすべての条件が成立した場合、DMAC の外部リクエストによる内蔵メモリ 内蔵周辺モジュール間、あるいは内蔵周辺モジュール 内蔵周辺モジュール間の DMA 転送不実行が発生します。

- (1) 外部リクエスト (DREQ) を転送要求に設定している。
- (2) DMA 転送区間を内蔵メモリ 内蔵周辺モジュール間、あるいは内蔵周辺モジュール 内蔵周辺モジュール間に設定している。
- (3) サイクルスチールモードに設定している。

#### 【回避方法】

DMAC の外部リクエストによる内蔵メモリ、内蔵周辺モジュール間 DMA 転送不実行に関する回避方法  
SH7615 内蔵 DMAC の DMA 転送区間を内蔵メモリ 内蔵周辺モジュール間、あるいは内蔵周辺モジュール 内蔵周辺モジュール間に設定して、DMA 転送を実行する場合は、以下の条件で使用してください。

- (1) 転送要求を外部リクエスト (DREQ) に設定しないでください。

### (15) DMAC シングルアドレス転送中のデータバス衝突

#### 【不具合内容】

DMAC シングルアドレス転送中のデータバス衝突  
SH7615 と DACK 付き外部デバイスおよびシンクロナス DRAM (SDRAM) を搭載するシステムにおいて、SDRAM ライトアクセス直後に DACK 付き外部デバイスから SDRAM に対する DMAC のシングルアドレス転送を実行した場合、DACK 付き外部デバイスから SDRAM に対するシングルアドレス転送中に SH7615 がデータバスをドライブし、データバス上で衝突が発生する場合があります。

#### 【発生条件】

DMAC シングルアドレス転送中のデータバス衝突の発生条件  
以下の (1) から (3) のすべての条件が成立した場合、DMAC による DACK 付き外部デバイスから SDRAM に対するシングルアドレス転送中にデータバス上で衝突が発生します。

- (1) 外部クロック (E ): 内部クロック (I ) = 1 : 1 で使用している。
- (2) SDRAM をシングルライトモードで使用している。
- (3) SDRAM ライトアクセス直後に DMAC による DACK 付き外部デバイスから SDRAM に対するシングルアドレス転送を実行する。

#### 【回避方法】

DMAC シングルアドレス転送中のデータバス衝突に関する回避方法  
SH7615 と DACK 付き外部デバイスおよび SDRAM を搭載するシステムにおいて、SH7615 内蔵 DMAC による DACK 付き外部デバイスから SDRAM に対するシングルアドレス転送を実行する場合は、以下の (1) から (2) のいずれかを採用して使用してください。

- (1) 外部クロック (E ): 内部クロック (I ) = 1 : 1 以外で使用してください。
- (2) DMAC の DACK 付き外部デバイスから SDRAM に対するシングルアドレス転送実行中は、CPU、イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC) および DMAC のもう一方のチャネルによる SDRAM へのライトアクセスを行わないでください。

(16) DMAC の DACK 誤出力

【不具合内容】

DMAC の DACK 誤出力

SH7615 内蔵 DMAC のチャンネル 0、チャンネル 1 ともに外部リクエスト (DREQ0、1) を転送要求に設定して、DMA 転送を実行した場合、DREQ1 が入力されていないにもかかわらず、DMAC チャンネル 1 が DACK1 を出力して DMA 転送を実行する場合があります。

【発生条件】

DMAC の DACK 誤出力の発生条件

以下の (1) から (4) のすべての条件が成立した場合、DMAC の DMA 転送中に DREQ1 が入力されていないにもかかわらず、DMAC チャンネル 1 が DACK1 を出力して DMA 転送を実行します。

- (1) DMAC チャンネル 0、チャンネル 1 ともにイネーブル状態にしている。
- (2) DMAC チャンネル 0、チャンネル 1 ともに外部リクエスト (DREQ0、1) を転送要求に設定している。
- (3) DMAC チャンネル 0、チャンネル 1 ともにサイクルスチールモードに設定している。
- (4) DMAC のプライオリティモードをラウンドロビンモードに設定している。

【回避方法】

DMAC の DACK 誤出力に関する回避方法

SH7615 内蔵 DMAC のチャンネル 0、チャンネル 1 ともに外部リクエスト (DREQ0、1) を転送要求に設定して、DMA 転送を実行する場合は、以下の (1) から (2) のいずれかを採用して使用してください。

- (1) DMAC チャンネル 0、あるいはチャンネル 1 のいずれかをバーストモードに設定してください。
- (2) DMAC のプライオリティモードを優先順位固定モードに設定してください。

## 11. ダイレクトメモリアクセスコントローラ (DMAC)

---

---

## 12. 16 ビットフリーランニングタイム (FRT)

---

### 12.1 概要

本 LSI は、16 ビットフリーランニングタイム (FRT) を 1 チャンネル内蔵しています。

FRT は、16 ビットのフリーランニングカウンタ (FRC) をベースにして、2 種類の独立した波形出力が可能であり、また入力パルスの幅や外部クロックの周期を測定することができます。

#### 12.1.1 特長

FRT には、次の特長があります。

- 4 種類のカウンタ入力クロックを選択可能  
3 種類の内部クロック (P /8、P /32、P /128) と外部クロックのうちから選択できます (外部イベントのカウンタが可能)。
- 2 本の独立したコンパレータ  
2 種類の波形出力が可能です。
- インพุットキャプチャ  
立ち上がりエッジ / 立ち下がりエッジの選択可能です。
- カウンタのクリア指定が可能  
コンペアマッチ A により、カウンタの値をクリアすることができます。
- 4 種類の割り込み要因  
コンペアマッチ × 2 要因、インพุットキャプチャ × 1 要因、オーバフロー × 1 要因があり、それぞれ独立に要求することができます。

## 12. 16ビットフリーランニングタイマ (FRT)

### 12.1.2 ブロック図

FRTのブロック図を図12.1に示します。

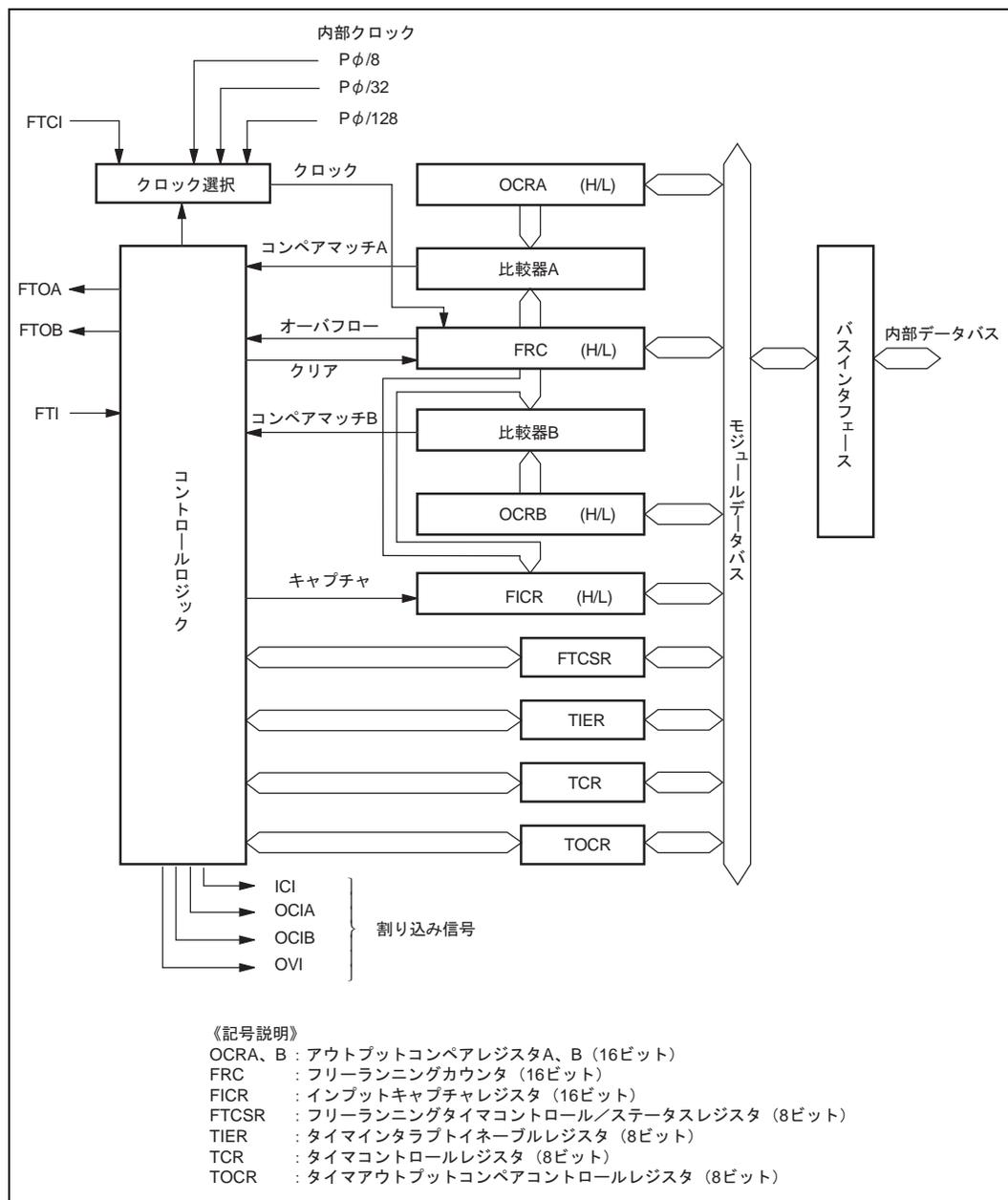


図 12.1 FRT のブロック図

### 12.1.3 端子構成

FRTの入出力端子を表 12.1 に示します。

表 12.1 端子構成

名 称	略 称	入出力	機 能
カウンタクロック入力端子	FTCI	入力	FRCのカウンタクロックの入力
アウトプットコンペア A 出力端子	FTOA	出力	アウトプットコンペア A の出力
アウトプットコンペア B 出力端子	FTOB	出力	アウトプットコンペア B の出力
インプットキャプチャの入力端子	FTI	入力	インプットキャプチャの入力

### 12.1.4 レジスタ構成

FRTのレジスタ構成を表 12.2 に示します。

表 12.2 レジスタ構成

名 称	略 称	R/W	初期値	アドレス
タイムインタラプティネーブルレジスタ	TIER	R/W	H'01	H'FFFFFFE10
フリーランニングタイムコントロール/ ステータスレジスタ	FTCSR	R/(W)*1	H'00	H'FFFFFFE11
フリーランニングカウンタ H	FRC H	R/W	H'00	H'FFFFFFE12
フリーランニングカウンタ L	FRC L	R/W	H'00	H'FFFFFFE13
アウトプットコンペアレジスタ A H	OCRA H	R/W	H'FF	H'FFFFFFE14*2
アウトプットコンペアレジスタ A L	OCRA L	R/W	H'FF	H'FFFFFFE15*2
アウトプットコンペアレジスタ B H	OCRB H	R/W	H'FF	H'FFFFFFE14*2
アウトプットコンペアレジスタ B L	OCRB L	R/W	H'FF	H'FFFFFFE15*2
タイマコントロールレジスタ	TCR	R/W	H'00	H'FFFFFFE16
タイマアウトプットコンペアコントロールレジスタ	TOCR	R/W	H'E0	H'FFFFFFE17
インプットキャプチャレジスタ H	FICR H	R	H'00	H'FFFFFFE18
インプットキャプチャレジスタ L	FICR L	R	H'00	H'FFFFFFE19

【注】 1. レジスタはすべてバイトサイズでアクセスしてください。

\*1 ビット7～1はリード専用で、フラグをクリアするための"0"書き込みのみ可能です。ビット0は読み出し/書き込み可能です。

\*2 OCRAとOCRBのアドレスは同一です。これらの切り替えはTOCRのOCRSビットで行います。

## 12.2 各レジスタの説明

### 12.2.1 フリーランニングカウンタ (FRC)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W															

FRC は、16 ビットの読み出し / 書き込み可能なアップカウンタで、入力したクロックによりカウントアップされます。入力するクロックは、TCR のクロックセレクト 1、0 ビット (CKS1、0) で選択します。

また、FRC はコンペアマッチ A によりクリアすることができます。

FRC がオーバフロー (H'FFFF H'0000) すると、FTCSR のオーバフローフラグ (OVF) は 1 にセットされます。

FRC は、CPU から読み出し / 書き込み可能ですが、16 ビットになっているため、CPU とのデータ転送はテンポラリレジスタ (TEMP) を介して行われます。詳細は、「12.3 CPU とのインタフェース」を参照してください。

FRC は、リセット、スタンバイモード、およびモジュールスタンバイ機能使用時に、H'0000 に初期化されません。

### 12.2.2 アウトプットコンペアレジスタ A、B (OCRA、B)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初期値:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W															

OCR は、16 ビットの読み出し / 書き込み可能な 2 本のレジスタ (OCRA、B) から構成されます。OCR の内容は、FRC の値と常に比較されています。両者の値が一致すると、FTCSR のアウトプットコンペアフラグ (OCFA、B) が 1 にセットされます。

さらに、OCR の値と FRC の値が一致した (コンペアマッチ) とき、TOCR のアウトプットレベルビット (OLVLA、B) で設定された出力レベルの値が、アウトプットコンペア出力端子 (FTOA、FTOB) に出力されます。リセット後、最初のコンペアマッチが起こるまで FTOA、B 出力は 0 出力です。

また、OCR は 16 ビットになっているため、CPU とのデータ転送は TEMP を介して行われます。詳細は、「12.3 CPU とのインタフェース」を参照してください

OCR は、リセット、スタンバイモード、およびモジュールスタンバイ機能使用時に、H'FFFF に初期化されません。

### 12.2.3 インプットキャプチャレジスタ (FICR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

FICR は、16 ビットのリード専用のレジスタから構成されます。

インプットキャプチャ信号 (FTI 端子) の立ち上がりまたは立ち下がりエッジが検出されると、そのときの FRC の値が FICR に転送されます。このとき同時に、FTCSR のインプットキャプチャフラグ (ICF) が 1 にセットされます。入力信号のエッジは、TCR のインプットエッジセレクトビット (IEDG) により選択できます。

FICR は 16 ビットのため、CPU とのデータ転送は TEMP を介して行われます。詳細は「12.3 CPU とのインタフェース」を参照してください。

インプットキャプチャ動作を確実に行うために、インプットキャプチャ入力信号のパルス幅は、6 システムクロック (P ) 以上にしてください。

FICR は、リセット、スタンバイモード、およびモジュールスタンバイ機能使用時に、H'0000 に初期化されません。

### 12.2.4 タイマインタラプトイネーブルレジスタ (TIER)

ビット:	7	6	5	4	3	2	1	0
	ICIE	—	—	—	OCIAE	OCIBE	OVIE	—
初期値:	0	0	0	0	0	0	0	1
R/W:	R/W	R	R	R	R/W	R/W	R/W	R

TIER は、8 ビットの読み出し / 書き込み可能なレジスタで、各割り込み要求の許可を制御します。

TIER は、リセットスタンバイモード、またはモジュールスタンバイ機能使用時に、H'01 に初期化されます。

- ・ ビット 7: インプットキャプチャインタラプトイネーブル (ICIE)

FTCSR のインプットキャプチャフラグ (ICF) が 1 にセットされたとき、ICF による割り込み要求 (ICI) の許可または禁止を選択します。

ビット 7	説明
ICIE	
0	ICF による割り込み要求 (ICI) を禁止 (初期値)
1	ICF による割り込み要求 (ICI) を許可

- ・ ビット 6 ~ 4: 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 12. 16 ビットフリーランニングタイム (FRT)

---

- ・ ビット3: アウトプットコンペアインタラプト A イネーブル (OCIAE)

FTCSR のアウトプットコンペアフラグ A (OCFA) が 1 にセットされたとき、OCFA による割り込み要求 (OCIA) の許可または禁止を選択します。

ビット3	説 明
OCIAE	
0	OCFA による割り込み要求 (OCIA) を禁止 (初期値)
1	OCFA による割り込み要求 (OCIA) を許可

- ・ ビット2: アウトプットコンペアインタラプト B イネーブル (OCIBE)

FTCSR のアウトプットコンペアフラグ B (OCFB) が 1 にセットされたとき、OCFB による割り込み要求 (OCIB) の許可または禁止を選択します。

ビット2	説 明
OCIBE	
0	OCFB による割り込み要求 (OCIB) を禁止 (初期値)
1	OCFB による割り込み要求 (OCIB) を許可

- ・ ビット1: タイマオーパフローインタラプトイネーブル (OVIE)

FTCSR のオーパフローフラグ (OVF) が 1 にセットされたとき、OVF による割り込み要求 (OVI) の許可または禁止を選択します。

ビット1	説 明
OVIE	
0	OVF による割り込み要求 (OVI) を禁止 (初期値)
1	OVF による割り込み要求 (OVI) を許可

- ・ ビット0: 予約ビット

読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。

## 12.2.5 フリーランニングタイムコントロール / ステータレジスタ (FTCSR)

ビット:	7	6	5	4	3	2	1	0
	ICF	—	—	—	OCFA	OCFB	OVF	CCLRA
初期値:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/W

【注】 \* ビット7、3～1はフラグをクリアするための0ライトのみ可能です。

FTCSR は、8ビットのレジスタで、カウンタクリアの選択、各割り込み要求信号の制御を行います。

FTCSR は、リセット、スタンバイモード、およびモジュールスタンバイ機能使用時に、H'00 に初期化されません。

なお、タイミングについては「12.4 動作説明」を参照してください。

・ ビット7: インพุットキャプチャフラグ (ICF)

インพุットキャプチャ信号によって、FRC の値が FICR に転送されたことを示すステータスフラグです。

本フラグのクリアは、ソフトウェアによって行われます。セットは、ハードウェアで行われます。ソフトウェアでセットすることはできません。

ビット7	説明
ICF	
0	〔クリア条件〕 (初期値) ICF=1 の状態で、ICF を読み出した後、ICF に 0 を書き込んだとき
1	〔セット条件〕 インพุットキャプチャ信号により、FRC の値が FICR に転送されたとき

・ ビット6～4: 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット3: アウトプットコンペアフラグ A (OCFA)

FRC と OCRA の値が一致したことを示すステータスフラグです。

本フラグのクリアは、ソフトウェアによって行われます。セットは、ハードウェアで行われます。ソフトウェアでセットすることはできません。

ビット3	説明
OCFA	
0	〔クリア条件〕 (初期値) OCFA=1 の状態で、OCFA を読み出した後、OCFA に 0 を書き込んだとき
1	〔セット条件〕 FRC=OCRA になったとき

## 12. 16 ビットフリーランニングタイマ (FRT)

### ・ ビット 2 : アウトプットコンペアフラグ B (OCFB)

FRC と OCFB の値が一致したことを示すステータスフラグです。

本フラグのクリアは、ソフトウェアによって行われます。セットは、ハードウェアで行われます。ソフトウェアでセットすることはできません。

ビット 2	説 明
OCFB	
0	[クリア条件] (初期値) OCFB=1 の状態で、OCFB を読み出した後、OCFB に 0 を書き込んだとき
1	[セット条件] FRC=OCRB になったとき

### ・ ビット 1 : タイマオーパフロー (OVF)

FRC がオーパフロー (H'FFFF H'0000) したことを示すステータスフラグです。

本フラグのクリアは、ソフトウェアによって行われます。セットは、ハードウェアで行われます。ソフトウェアでセットすることはできません。

ビット 1	説 明
OVF	
0	[クリア条件] (初期値) OVF=1 の状態で、OVF を読み出した後、OVF に 0 を書き込んだとき
1	[セット条件] FRC の値が、H'FFFF H'0000 になったとき

### ・ ビット 0 : カウンタクリア A (CCLRA)

コンペアマッチ A (FRC と OCRA の一致信号) により、FRC をクリアするか、しないかを選択します。

ビット 0	説 明
CCLRA	
0	FRC のクリアを禁止 (初期値)
1	コンペアマッチ A により FRC をクリア

## 12.2.6 タイマコントロールレジスタ (TCR)

ビット :	7	6	5	4	3	2	1	0
	IEDG	—	—	—	—	—	CKS1	CKS0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R	R	R	R	R	R/W	R/W

TCR は、8 ビットの読み出し / 書き込み可能なレジスタです。インプットキャプチャ入力エッジの選択、FRC の入力クロックの選択を行います。

TCR は、リセット、スタンバイモード、またはモジュールスタンバイ機能使用時に、H'00 に初期化されます。

- ビット7: インพุットエッジセレクト (IEDG)

インพุットキャプチャ入力 (FTI) の立ち上がりエッジまたは立ち下がりエッジを選択します。

ビット7	説明
IEDG	
0	インพุットキャプチャ入力の立ち下がりエッジ ( $\downarrow$ ) でキャプチャ (初期値)
1	インพุットキャプチャ入力の立ち上がりエッジ ( $\uparrow$ ) でキャプチャ

- ビット6 ~ 2: 予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

- ビット1, 0: クロックセレクト (CKS1, 0)

FRCに入力するクロックを、内部クロック3種類または外部クロックから選択します。外部クロックは、立ち上がりエッジでカウントします。

ビット1	ビット0	説明
CKS1	CKS0	
0	0	内部クロック: P /8 でカウント (初期値)
0	1	内部クロック: P /32 でカウント
1	0	内部クロック: P /128 でカウント
1	1	外部クロック: 立ち上がりエッジ ( $\uparrow$ ) でカウント

## 12.2.7 タイマアウトプットコンペアコントロールレジスタ (TOCR)

ビット:	7	6	5	4	3	2	1	0
	—	—	—	OCRS	—	—	OLVLA	OLVLB
初期値:	1	1	1	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W

TOCR は、8 ビットの読み出し/書き込み可能なレジスタです。アウトプットコンペア出力レベルの選択、およびアウトプットコンペアレジスタ A、B のアクセスの切り替え制御を行います。

TOCR は、リセット、スタンバイモード、およびモジュールスタンバイ機能使用時に、HE0 に初期化されず。

- ビット7 ~ 5: 予約ビット

読み出すと常に1が読み出されます。書き込む値も常に1にしてください。

## 12. 16 ビットフリーランニングタイム (FRT)

---

- ・ ビット 4 : アウトプットコンペアレジスタセレクト (OCRS)

OCRA と OCRB のアドレスは同一です。OCRS ビットは、このアドレスを読み出し / 書き込みするときにどちらのレジスタを選択するかを制御します。OCRA と OCRB の動作には、影響を与えません。

ビット 4	説 明
OCRS	
0	OCRA レジスタを選択 (初期値)
1	OCRB レジスタを選択

- ・ ビット 3、2 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 1 : アウトプットレベル A (OLVLA)

コンペアマッチ A (FRC と OCRA の一致による信号) により、アウトプットコンペア A 出力端子に出力する出力レベルを選択します。

ビット 1	説 明
OLVLA	
0	コンペアマッチ A により 0 出力 (初期値)
1	コンペアマッチ A により 1 出力

- ・ ビット 0 : アウトプットレベル B (OLVLB)

コンペアマッチ B (FRC と OCRB の一致による信号) により、アウトプットコンペア B 出力端子に出力する出力レベルを選択します。

ビット 0	説 明
OLVLB	
0	コンペアマッチ B により 0 出力 (初期値)
1	コンペアマッチ B により 1 出力

## 12.3 CPU とのインタフェース

FRC、OCRA、OCRB、FICR は、16ビットのレジスタです。一方、CPU と FRT の間のデータバスは 8ビット幅です。したがって、CPU がこれら 3 種類のレジスタをアクセスするには、8ビットのテンポラリレジスタ (TEMP) を介して行います。

各レジスタの読み出し / 書き込みは次のような動作で行われます。

- レジスタへの書き込み時の動作

上位バイトの書き込みにより、上位バイトのデータが TEMP にストアされます。次に下位バイトの書き込みで、TEMP にある上位バイトの値とあわせて、16ビットデータとしてレジスタに書き込みされます。

- レジスタからの読み出し時の動作

上位バイトの読み出しで、上位バイトの値は CPU に転送され、下位バイトの値は TEMP に転送されます。次に下位バイトの読み出しで、TEMP にある下位バイトの値が CPU に転送されます。

これら 3 種類のレジスタをアクセスするときは、常にバイトアクセスを 2 回で行い、かつ上位バイト、下位バイトの順序で行ってください。上位バイトのみや、下位バイトのみのアクセスでは、データは正しく転送されませんので注意してください。

図 12.2 に FRC をアクセスする場合のデータの流れを示します。他のレジスタの場合も同様な動作になります。ただし、OCRA、B の読み出し時には、上位バイト、下位バイトとも TEMP を介さずに直接 CPU にデータを転送します。

12. 16 ビットフリーランニングタイム (FRT)

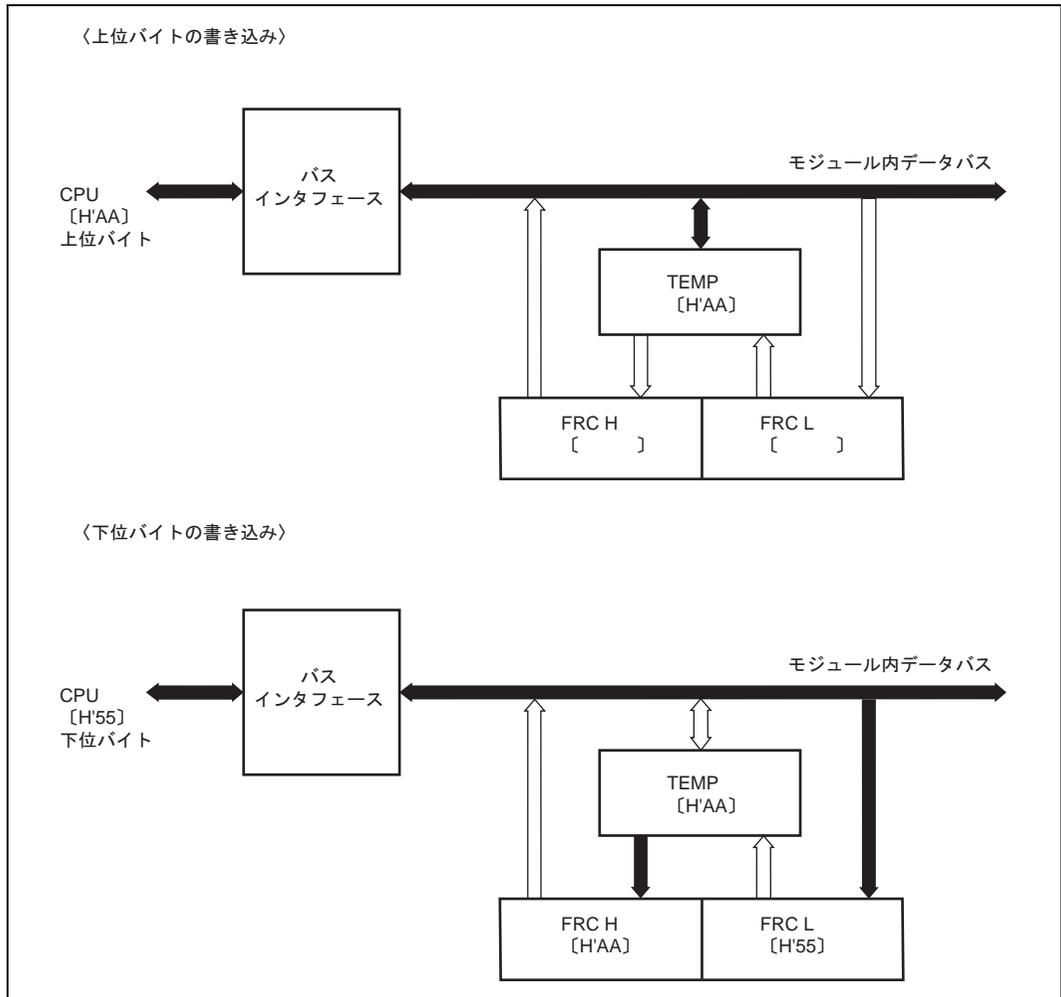


図 12.2 (a) FRC のアクセス動作 (CPU FRC [H'AA55] 書き込み時)

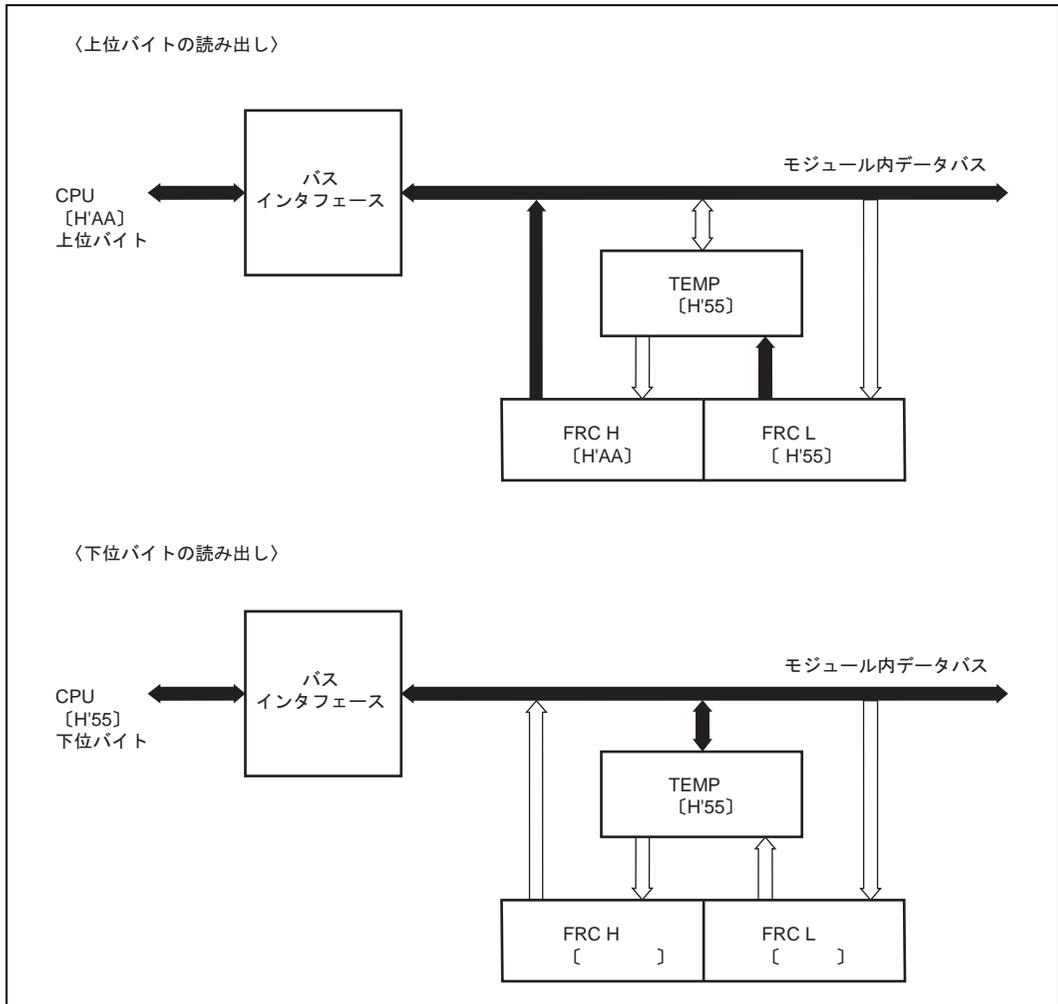


図 12.2 (b) FRC のアクセス動作 (FRC CPU [H'AA55] 読み出し時)

## 12.4 動作説明

### 12.4.1 FRC のカウントタイミング

FRC は、入力されたクロック (内部クロックまたは外部クロック) によりカウントアップされます。

#### (1) 内部クロック動作の場合

TCR の CKS1、0 ビットの設定により、システムクロック ( $P$ ) を分周して作られる 3 種類の内部クロック ( $P/8$ 、 $P/32$ 、 $P/128$ ) が選択されます。このときのタイミングを図 12.3 に示します。

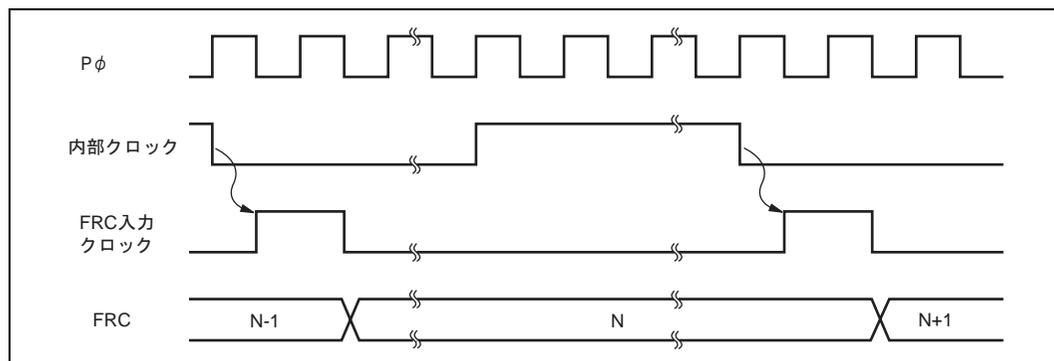


図 12.3 内部クロック動作時のカウントタイミング

#### (2) 外部クロック動作の場合

TCR の CKS1、0 ビットの設定により、外部クロック入力を選択されます。外部クロックは立ち上がりエッジでカウントします。なお、外部クロックのパルス幅は、6 システムクロック ( $P$ ) 以上必要です。これ以下のパルス幅では正しく動作しませんので注意してください。

このときのタイミングを図 12.4 に示します。

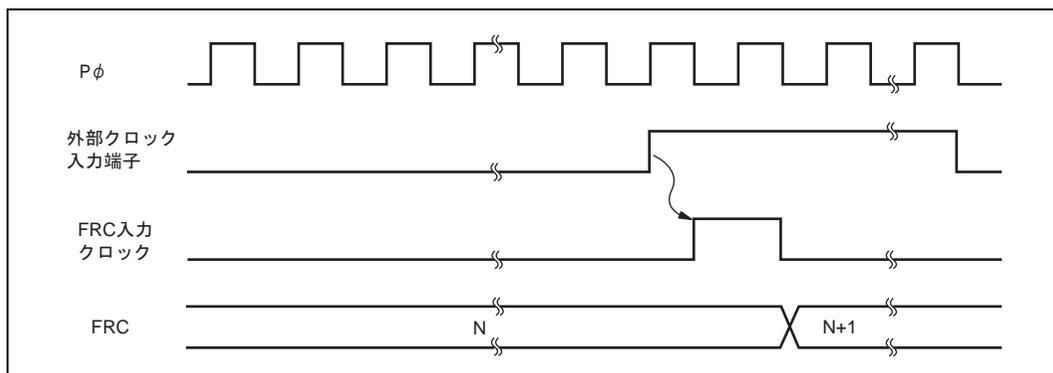


図 12.4 外部クロック動作時のカウントタイミング

### 12.4.2 アウトプットコンペア出力タイミング

コンペアマッチが発生したとき、TOCR の OLVL ビットで設定された出力レベルがアウトプットコンペア出力端子 (FTOA、FTOB) に出力されます。図 12.5 にアウトプットコンペア A の場合の出力タイミングを示します。

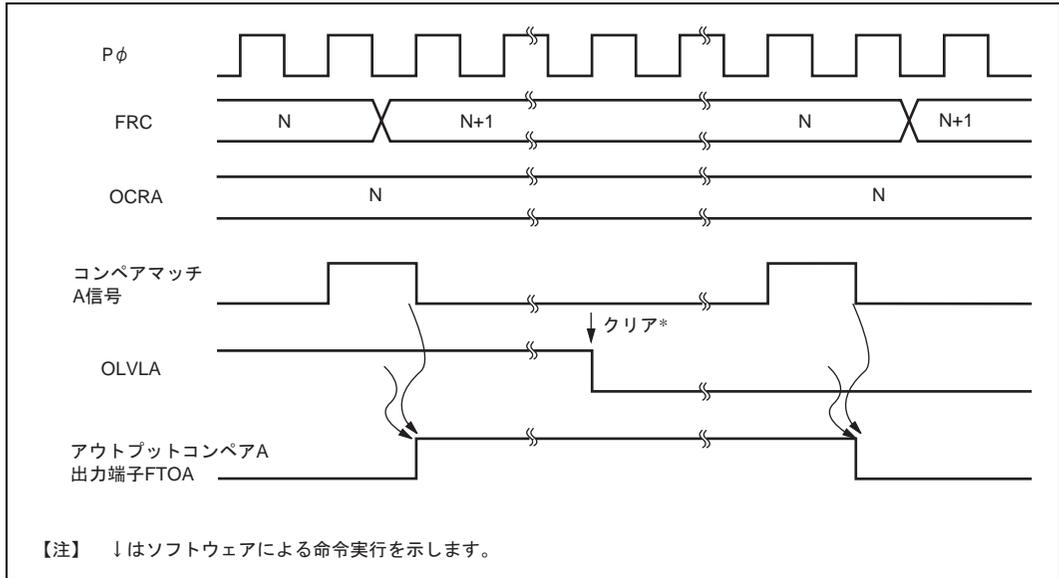


図 12.5 アウトプットコンペア A 出力タイミング

### 12.4.3 FRC クリアタイミング

FRC は、コンペアマッチ A でクリアすることができます。このタイミングを図 12.6 に示します。

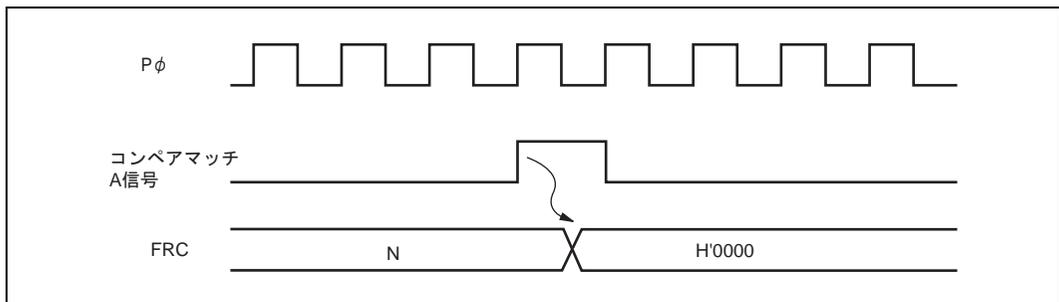


図 12.6 コンペアマッチ A によるクリアタイミング

### 12.4.4 インプットキャプチャ入力タイミング

#### (1) インプットキャプチャ入力タイミング

インプットキャプチャ入力は、TCR の IEDG ビットで立ち上がりエッジ / 立ち下がりエッジを選択します。立ち上がりエッジを選択した (IEDG = 1) 場合のタイミングを図 12.7 に示します。

## 12. 16 ビットフリーランニングタイム (FRT)

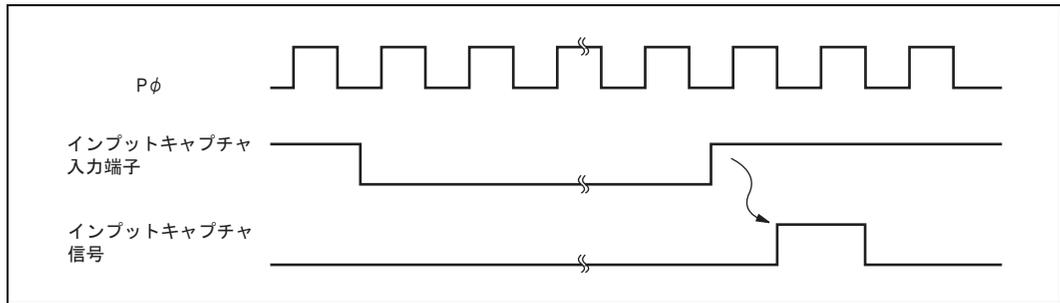


図 12.7 インットキャプチャ信号タイミング (通常時)

また、FICR の読み出し (上位バイトの読み出し) 時に、インットキャプチャ信号を入力すると、インットキャプチャ信号は P の 1 クロック分遅延されます。このタイミングを図 12.8 に示します。

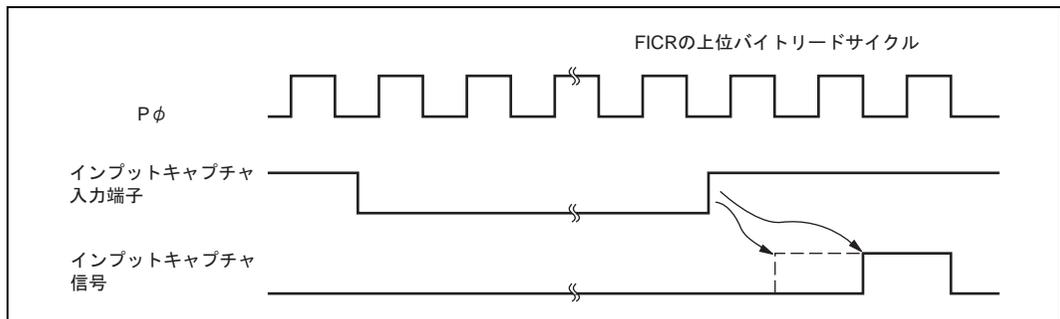


図 12.8 インットキャプチャ信号タイミング  
(FICR の読み出し時に、インットキャプチャ入力を入力した場合)

### 12.4.5 インพุットキャプチャフラグ (ICF) のセットタイミング

インพุットキャプチャ入力により ICF は 1 にセットされ、同時に FRC の値が FICR に転送されます。このタイミングを図 12.9 に示します。

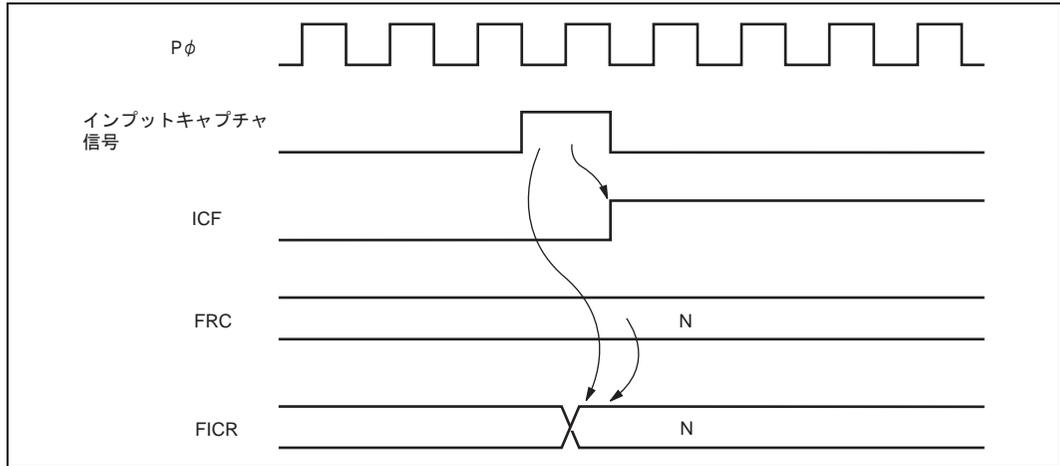


図 12.9 ICF のセットタイミング

### 12.4.6 アウトプットコンペアフラグ (OCFA、B) のセットタイミング

OCFA、B は、OCRA、B と FRC の値が一致したとき出力されるコンペアマッチ信号により、1 にセットされます。コンペアマッチ信号は、値が一致した最後のステート (FRC が一致したカウント値を更新するタイミング) で発生します。

FRC と OCRA、B が一致した後、次のカウンタアップが発生するまでコンペアマッチは発生しません。OCFA、B のセットタイミングを図 12.10 に示します。

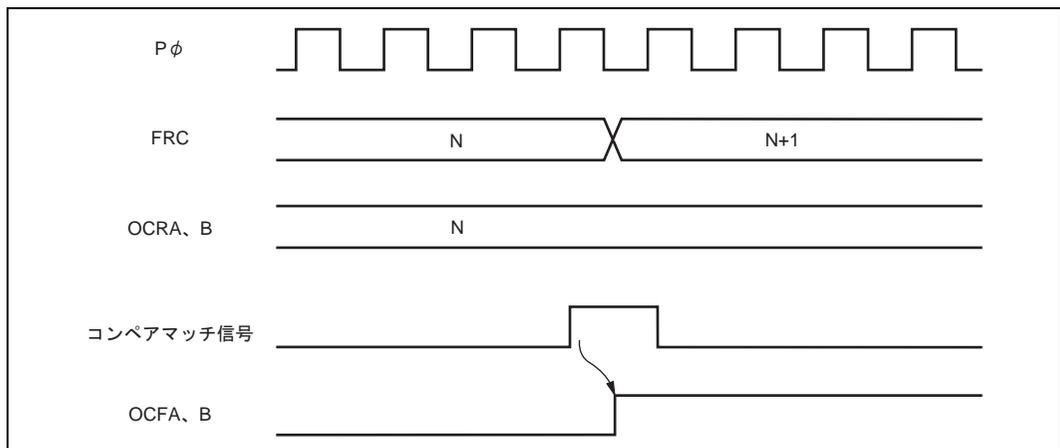


図 12.10 OCF セットタイミング

### 12.4.7 タイマオーバーフローフラグ (OVF) のセットタイミング

OVF は、FRC がオーバーフロー (H'FFFF → H'0000) したとき 1 にセットされます。このときのタイミングを図 12.11 に示します。

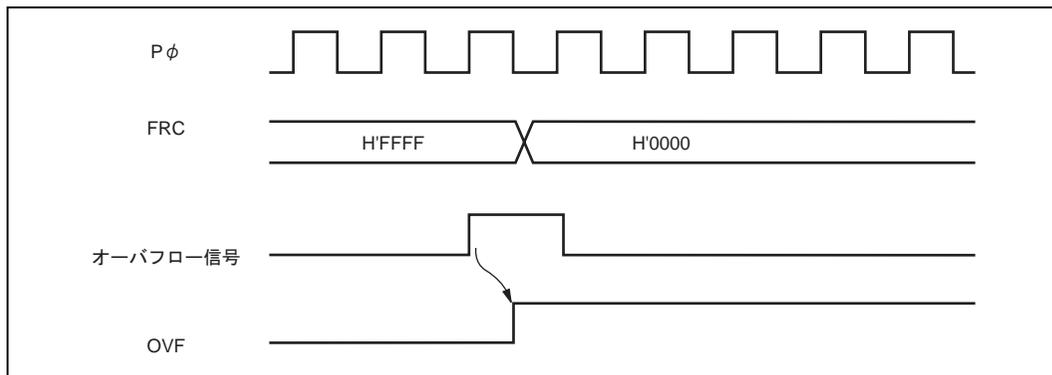


図 12.11 OVF のセットタイミング

## 12.5 割り込み要因

FRT 割り込み要因は、ICI、OCIA、OCIB および OVI の 3 種類合計 4 つあります。表 12.3 に各割り込み要因とリセット解除時の優先順位を示します。各割り込み要因は、TIER の各割り込みイネーブルビットで許可または禁止され、それぞれ独立に割り込みコントローラに送られます。優先順位、FRT 以外の割り込みとの関係は、「5. 割り込みコントローラ (INTC)」を参照してください。

表 12.3 FRT 割り込み要因

割り込み要因	内 容	割り込み優先順位
ICI	ICFによる割り込み	高 ↑ ↓ 低
OCIA、B	OCFAまたはOCFBによる割り込み	
OVI	OVFによる割り込み	

## 12.6 FRT の使用例

デューティ 50% のパルスを任意の位相差で出力させた例を図 12.12 に示します。これは次に示すように設定します。

- (1) FTCSR の CCLRA ビットを 1 にセットします。
- (2) 各コンペアマッチが発生するたびに OLVLA、B ビットをソフトウェアにより反転させます。

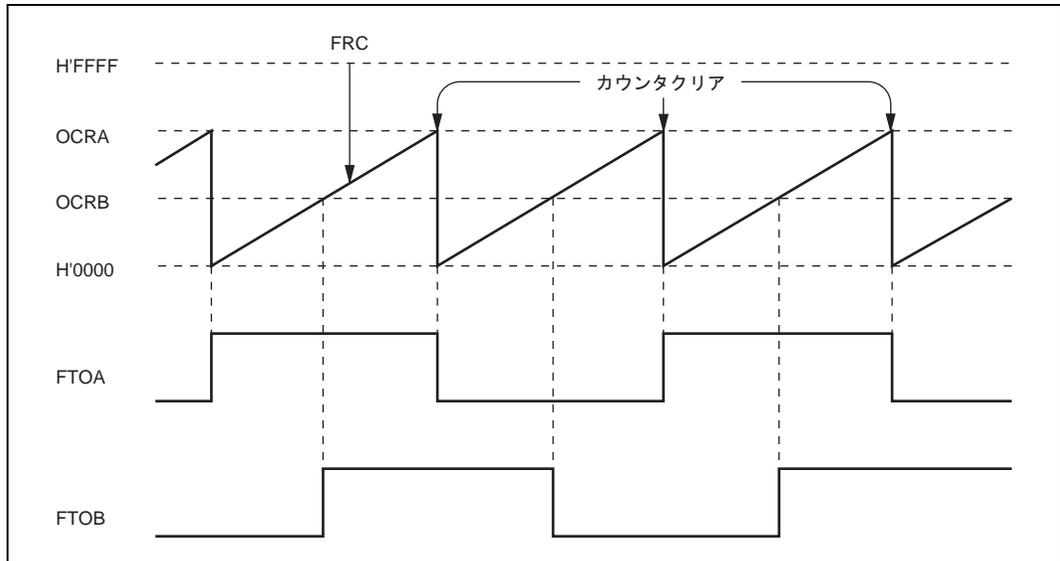


図 12.12 パルス出力例

## 12.7 使用上の注意

FRT の動作中、次のような競合や動作が起こりますので、注意してください。

### 12.7.1 FRC のライトとクリアの競合

FRC の下位バイトライトサイクル中の図 12.13 に示すタイミングで、カウンタクリア信号が発生すると、FRC への書き込みは行われず FRC のクリアが優先されます

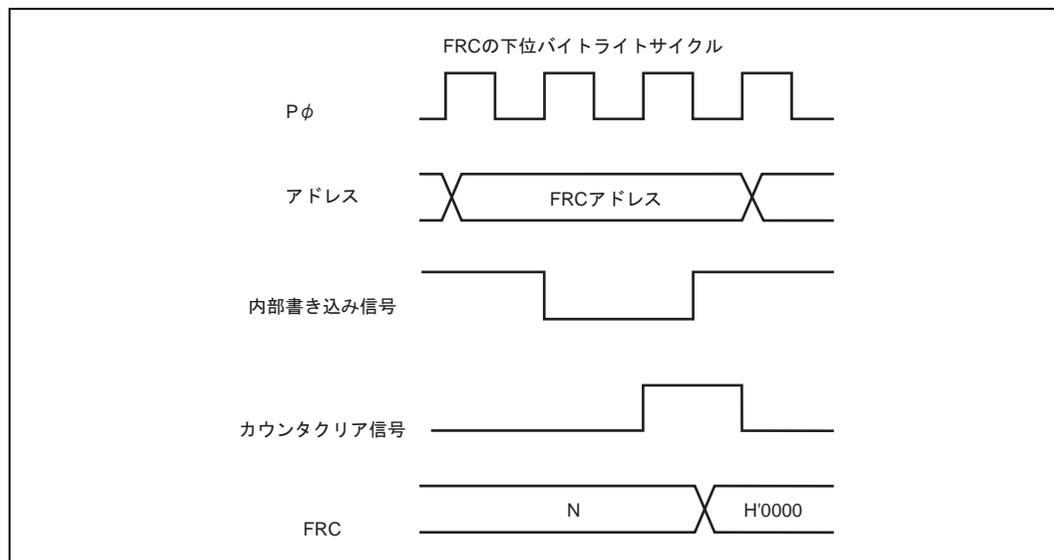


図 12.13 FRC のライトとクリアの競合

### 12.7.2 FRC のライトとカウントアップの競合

FRC の下位バイトライトサイクル中の図 12.14 に示すタイミングでカウントアップが発生しても、カウントアップされずカウンタライトが優先されます。

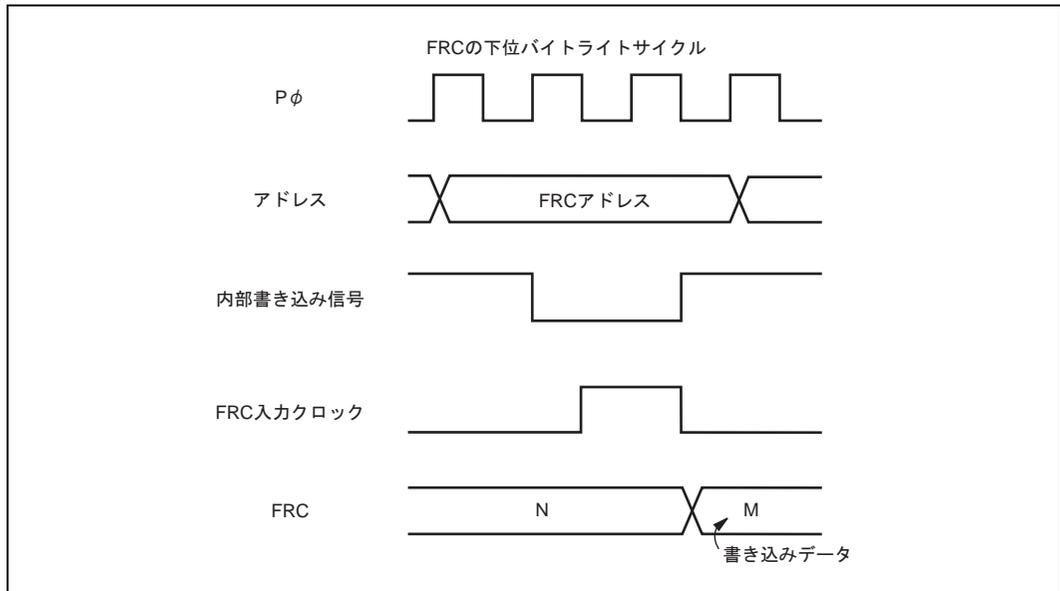


図 12.14 FRC ライトとカウントアップの競合

### 12.7.3 OCR のライトとコンペアマッチの競合

OCRA、B の下位バイトライトサイクル中の図 12.15 に示すタイミングでコンペアマッチが発生した場合、OCR ライトが優先され、コンペアマッチ信号は禁止されます。

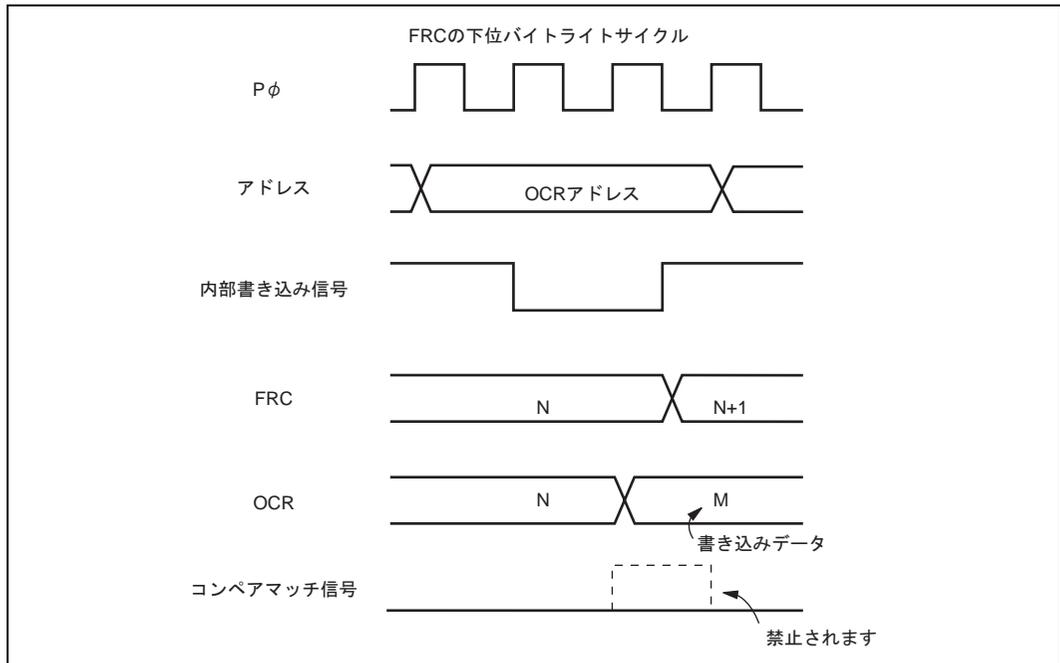


図 12.15 OCR とコンペアマッチの競合

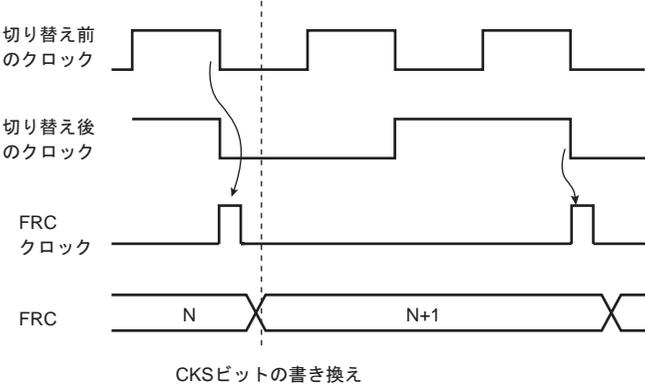
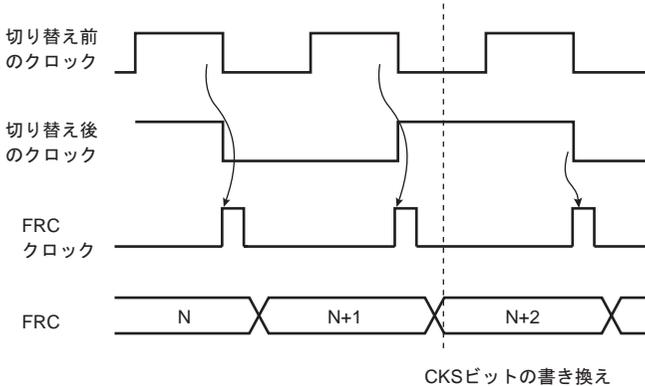
### 12.7.4 内部クロックの切り替えとカウンタの動作

内部クロックを切り替えるタイミングによっては、FRC がカウントアップされてしまう場合があります。内部クロックの切り替えタイミング (CKS1、0 ビットの書き換え) と FRC 動作の関係を表 12.4 に示します。

内部クロックを使用する場合、システムクロック (P) を分周した内部クロック立ち下がりエッジで検出して FRC クロックを発生しています。そのため表 12.4 の No.3 のように、切り替え前のクロックがハイレベル、切り替え後のクロックがローレベルになるようなクロックの切り替えを行うと、切り替えタイミングを立ち下がりエッジとみなして FRC クロックが発生し、FRC がカウントアップされてしまいます。

また、内部クロックと外部クロックを切り替えるときも、FRC がカウントアップされることがあります。

表 12.4 内部クロックの切り替えと FRC 動作 (1)

No.	CKS1、0 ビット書き換えタイミング	FRC 動作
1	L L レベルの切り替え	 <p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>FRC クロック</p> <p>FRC</p> <p>CKSビットの書き換え</p>
2	L H レベルの切り替え	 <p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>FRC クロック</p> <p>FRC</p> <p>CKSビットの書き換え</p>

12. 16 ビットフリーランニングタイマ (FRT)

表 12.4 内部クロックの切り替えと FRC 動作 (2)

No.	CKS1、0 ビット書き換えタイミング	FRC 動作
3	H L レベルの切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>FRC クロック</p> <p>FRC</p> <p>N N+1 N+2</p> <p>CKSビットの書き換え</p>
4	H H レベルの切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>FRC クロック</p> <p>FRC</p> <p>N N+1 N+2</p> <p>CKSビットの書き換え</p>

【注】 切り替えのタイミングを立ち下がりエッジとみなすために発生し、FRC はカウントアップされてしまいます。

12.7.5 タイマ出力 (FTOA、FTOB)

電源立ち上げ時のリセット中、発振が安定するまで、タイマ出力 (FTOA、FTOB) は不定です。発振安定時間が経過したとき、初期値を出力します。

---

## 13. ウォッチドッグタイマ (WDT)

---

### 13.1 概要

ウォッチドッグタイマ (WDT) は 1 チャンネルのタイマで、システムの監視を行うことができます。WDT は、システムの暴走などによりカウンタの値を CPU が正しく書き換えられずにオーバーフローすると、外部にオーバーフロー信号 ( $\overline{\text{WDTOVF}}$ ) を出力します。同時に、本 LSI の内部リセット信号を発生することもできます。

ウォッチドッグタイマとして使用しないときは、インターバルタイマとして使用することもできます。インターバルタイマとして使用した場合は、カウンタがオーバーフローするごとにインターバルタイマ割り込みが発生します。また、WDT はスタンバイモードの解除時、クロック周波数変更時、クロックポーズモード時にも使用されます。

#### 13.1.1 特長

WDT には次の特長があります。

- ウォッチドッグタイマモードとインターバルタイマモードを切り替え可能
- ウォッチドッグタイマモード時、 $\overline{\text{WDTOVF}}$  を出力  
カウンタがオーバーフローすると、外部に  $\overline{\text{WDTOVF}}$  信号を出力します。このとき、同時に本 LSI 内部をリセットするかどうか選択できます。この内部リセットは、パワーオンリセットまたはマニュアルリセットを選択できます。
- インターバルタイマモード時、割り込みを発生  
カウンタがオーバーフローすると、インターバルタイマ割り込みが発生します。
- スタンバイモードの解除時、クロック周波数変更時、クロックポーズモード時に使用
- 8 種類のカウンタ入力クロックを選択可能

### 13. ウォッチドッグタイマ (WDT)

#### 13.1.2 ブロック図

WDTのブロック図を図 13.1 に示します。

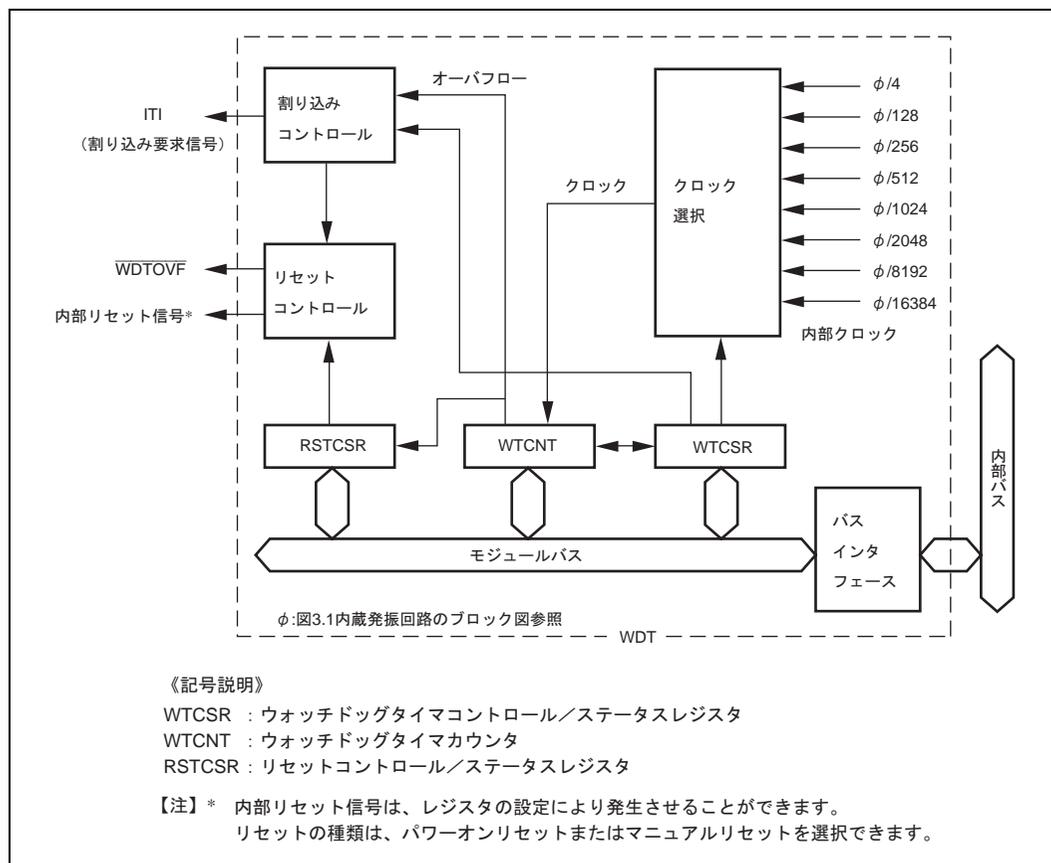


図 13.1 WDT のブロック図

#### 13.1.3 端子構成

WDTの端子を表 13.1 に示します。

表 13.1 端子構成

名称	略称	入出力	機能
ウォッチドッグタイマオーバーフロー	WDTOVF	出力	ウォッチドッグタイマモード時のカウンタオーバーフロー信号出力

### 13.1.4 レジスタ構成

WDT には、表 13.2 に示すように 3 本のレジスタがあります。これらのレジスタにより、クロックの選択、WDT のモードの切り替え、リセット信号の制御などを行います。

表 13.2 レジスタ構成

名称	略称	R/W	初期値	アドレス	
				書き込み *1	読み出し *2
ウォッチドッグタイマコントロール / ステータスレジスタ	WTCSR	R/(W)*3	H'18	H'FFFFFFE80	H'FFFFFFE80
ウォッチドッグタイマカウンタ	WTCNT	R/W	H'00		H'FFFFFFE81
リセットコントロール / ステータスレジスタ	RSTCSR	R/(W)*3	H'1E	H'FFFFFFE82	H'FFFFFFE83

- 【注】
- \*1 書き込みは、ワード単位で行ってください。バイトおよびロングワード単位では書き込むことができません。
  - \*2 読み出しは、バイト単位で行ってください。ワードおよびロングワード単位では正しい値を読み出すことができません。
  - \*3 ビット 7 には、フラグをクリアするために、0 のみ書き込むことができます。

## 13.2 レジスタの説明

### 13.2.1 ウォッチドッグタイマカウンタ (WTCNT)

ビット:	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W							

ウォッチドッグタイマカウンタ (WTCNT) は、読み出し/書き込み可能な \*8 ビットのアップカウンタです。ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR) のタイムアウトビット (TME) を 1 にすると、WTCSR の CKS2 ~ CKS0 ビットで選択した内部クロックにより、WTCNT はカウントアップを開始します。WTCNT の値がオーバーフロー (H'FF H'00) すると、WTCSR の  $\overline{WT/IT}$  ビットで選択したモードによって、ウォッチドッグタイマオーバーフロー信号 ( $\overline{WDTOVF}$ ) またはインターバルタイム割り込み (ITI) が発生します。

WTCNT は、リセットまたは TME ビットが 0 のとき、H'00 に初期化されます。スタンバイモード時、クロック周波数変更時、クロックポーズモード時には初期化されません。

【注】 \* WTCNT は容易に書き換えられないように、書き込み方法が一般のレジスタと異なります。詳細は、「13.2.4 レジスタアクセス時の注意」を参照してください。

### 13.2.2 ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)

ビット:	7	6	5	4	3	2	1	0
	OVF	WT/IT	TME	—	—	CKS2	CKS1	CKS0
初期値:	0	0	0	1	1	0	0	0
R/W:	R/(W)*	R/W	R/W	R	R	R/W	R/W	R/W

ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR) は、読み出し/書き込み可能な \*8 ビットのレジスタで、ウォッチドッグタイマカウンタ (WTCNT) に入力するクロック、モードの選択などを行います。

ビット 7 ~ 5 は、リセットおよびスタンバイモード時、クロック周波数変更時、クロックポーズモード時に 000 に初期化されます。ビット 2 ~ 0 は、リセットで 000 に初期化されますが、スタンバイモード時、クロック周波数変更時、クロックポーズモード時には初期化されません。

【注】 \* WTCSR は容易に書き換えられないように、書き込み方法が一般のレジスタと異なります。詳細は、「13.2.4 レジスタアクセス時の注意」を参照してください。

・ ビット7：オーバーフローフラグ (OVF)

インターバルタイマモードで、WTCNT がオーバーフロー (H'FF H'00) したことを示します。ウォッチドッグタイマモードではセットされません。

ビット7	説 明
OVF	
0	インターバルタイマモードで WTCNT のオーバーフローなし (初期値) [クリア条件] OVF を読み出してから 0 を書き込む
1	インターバルタイマモードで WTCNT のオーバーフロー発生

・ ビット6：タイマモードセレクト (WT/IT)

ウォッチドッグタイマとして使用するか、インターバルタイマとして使用するかを選択します。この選択によって、WTCNT がオーバーフローしたとき、インターバルタイマ割り込み (ITI) が発生するか、 $\overline{\text{WDTOVF}}$  信号が発生するかが決まります。

ビット6	説 明
WT/IT	
0	インターバルタイマモード：WTCNT がオーバーフローしたとき CPU ヘインターバルタイマ割り込み (ITI) を要求 (初期値)
1	ウォッチドッグタイマモード：WTCNT がオーバーフローしたとき $\overline{\text{WDTOVF}}$ 信号を外部へ出力*

【注】 \* ウォッチドッグタイマモードのとき、WTCNT がオーバーフローした場合についての詳細は「13.2.3 リセットコントロール/ステータスレジスタ (RSTCSR)」を参照してください。

・ ビット5：タイマイネーブル (TME)

タイマ動作の開始または停止を設定します。

ビット5	説 明
TME	
0	タイマディスエーブル：WTCNT を H'00 に初期化し、カウントアップを停止 (初期値)
1	タイマイネーブル：WTCNT はカウントアップを開始。WTCNT がオーバーフローすると、 $\overline{\text{WDTOVF}}$ 信号または割り込みが発生。

・ ビット4、3：予約ビット

読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。

・ ビット2～0：クロックセレクト2～0 (CKS2～CKS0)

システムクロック ( ) を分周して得られる 8 種類の内部クロックから、WTCNT に入力するクロックを選択します。

### 13. ウォッチドッグタイマ (WDT)

ビット2	ビット1	ビット0	説 明	
CKS2	CKS1	CKS0	クロック	オーバーフロー周期* ( =60MHz の場合 )
0	0	0	/4 (初期値)	17.0 μs
		1	/128	544 μs
	1	0	/256	1.1 ms
		1	/512	2.2 ms
1	0	0	/1024	4.4 ms
		1	/2048	8.7 ms
	1	0	/8192	34.8 ms
		1	/16384	69.6 ms

【注】 \* オーバフロー周期は、WTCNT が H'00 からカウントアップを開始し、オーバーフローするまでの時間です。

#### 13.2.3 リセットコントロール/ステータスレジスタ (RSTCSR)

ビット:	7	6	5	4	3	2	1	0
	WOVF	RSTE	RSTS	—	—	—	—	—
初期値:	0	0	0	1	1	1	1	0
R/W:	R/(W)*	R/W	R/W	R	R	R	R	R

【注】 \* ビット7には、フラグをクリアするために、0のみ書き込むことができます。

リセットコントロール/ステータスレジスタ (RSTCSR) は、読み出し/書き込み可能な\*8ビットのレジスタで、ウォッチドッグタイマカウンタ (WTCNT) のオーバーフローによる内部リセット信号の発生を制御し、内部リセット信号の種類を選択します。

RSTCSR は、 $\overline{RES}$  端子からのリセット信号で H'1E に初期化されますが、WDT のオーバーフローによる内部リセット信号では初期化されません。スタンバイモード時、クロックポーズモード時には、H'1E に初期化されません。

【注】 \* RSTCSR は容易に書き換えられないように、書き込み方法が一般のレジスタと異なっています。詳細は「13.2.4 レジスタアクセス時の注意」を参照してください。

・ ビット7：ウォッチドッグタイマオーバーフローフラグ (WOVF)

ウォッチドッグタイマモードで、WTCNT がオーバーフロー (H'FF → H'00) したことを示します。インターバルタイマモードではセットされません。

ビット7	説 明
WOVF	
0	ウォッチドッグタイマモードで WTCNT のオーバーフローなし (初期値) [クリア条件] WOVF を読み出してから WOVF に 0 を書き込む
1	ウォッチドッグタイマモードで WTCNT のオーバーフロー発生

- ・ ビット6：リセットイネーブル (RSTE)

ウォッチドッグタイマモードで WTCNT がオーバーフローしたとき、本 LSI 内部をリセットする信号を発生するかどうかを選択します。

ビット6	説明
RSTE	
0	WTCNT がオーバーフローしたとき、内部リセットしない* (初期値)
1	WTCNT がオーバーフローしたとき、内部リセットする

【注】 本 LSI 内部はリセットされませんが、WDT 内の WTCNT、WTCSR はリセットされます。

- ・ ビット5：リセットセレクト (RSTS)

ウォッチドッグタイマモードで WTCNT がオーバーフローして発生する内部リセットの種類を選択します。

ビット5	説明
RSTS	
0	パワーオンリセット (初期値)
1	マニュアルリセット

- ・ ビット4～1：予約ビット

読み出すと常に1が読み出されます。書き込む値も常に1にしてください。

- ・ ビット0

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

### 13. ウォッチドッグタイマ (WDT)

#### 13.2.4 レジスタアクセス時の注意

ウォッチドッグタイマカウンタ (WTCNT)、ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)、リセットコントロール/ステータスレジスタ (RSTCSR) は、容易に書き換えられないように、書き込み方法が一般のレジスタと異なっています。次の方法で、読み出し/書き込みを行ってください。

##### (1) WTCNT、WTCSR への書き込み

WTCNT、WTCSR へ書き込むときは、必ずワード転送命令を使用してください。バイト転送、ロングワード転送命令では、書き込みません。

書き込み時は、WTCNT と WTCSR が同一アドレスに割り当てられています。このため、図 13.2 に示すように、WTCNT へ書き込むときは上位バイトを H'5A にし、下位バイトを書き込みデータにして転送してください。WTCSR へ書き込むときは上位バイトを H'A5 にし、下位バイトを書き込みデータにして転送してください。このように転送すると、下位バイトのデータが WTCNT または WTCSR へ書き込まれます。

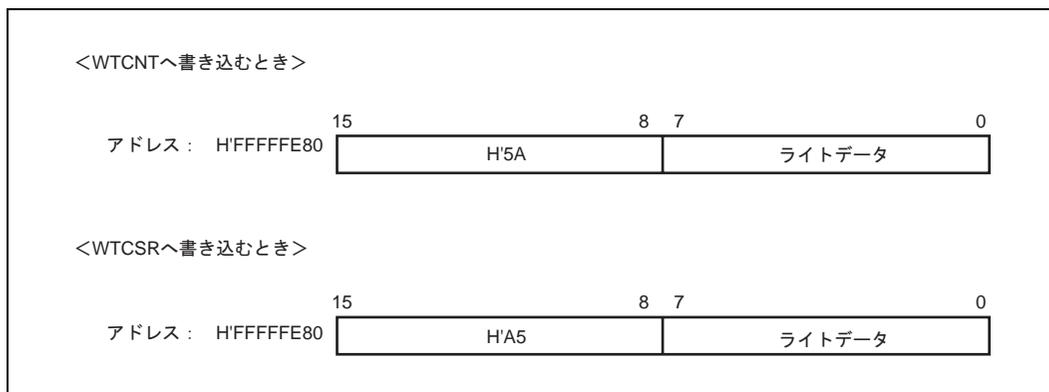


図 13.2 WTCNT、WTCSR への書き込み

## (2) RSTCSR への書き込み

RSTCSR へ書き込むときは、アドレス H'FFFFFFE82 に対してワード転送を行ってください。バイト転送、ロングワード転送命令では、書き込めません。

WOVF ビット (ビット 7) へ 0 を書き込む場合と、RSTE ビット (ビット 6) と RSTS ビット (ビット 5) に書き込む場合では、図 13.3 に示すように、書き込みの方法が異なります。

WOVF ビットへ 0 を書き込むときは、上位バイトを H'A5 にし、下位バイトを H'00 にして転送してください。このようにすると、WOVF ビットが 0 にクリアされます。このとき、RSTE、RSTS ビットは影響を受けません。RSTE、RSTS ビットに書き込むときは、上位バイトを H'5A にし、下位バイトを書き込みデータにして転送してください。このようにすると、下位バイトのビット 6 と 5 の値が RSTE ビットと RSTS ビットにそれぞれ書き込まれます。このとき、WOVF ビットは影響を受けません。

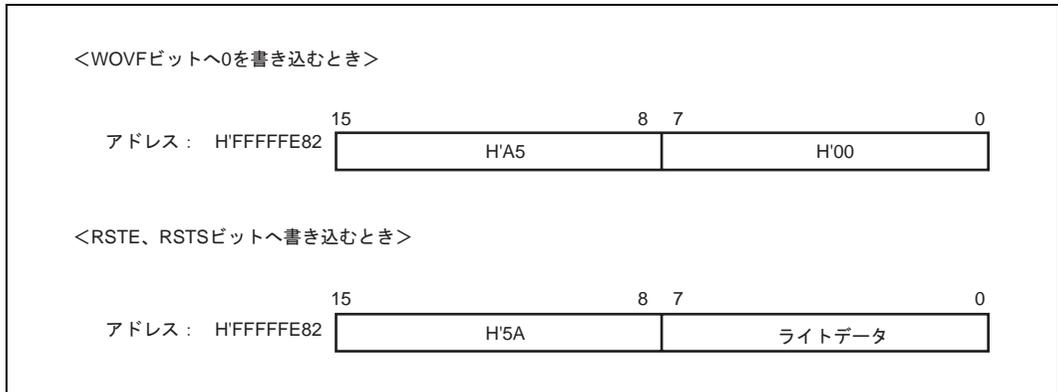


図 13.3 RSTCSR への書き込み

## (3) WTCNT、WTCSR、RSTCSR からの読み出し

読み出しは、一般のレジスタと同様の方法で行うことができます。WTCSR はアドレス H'FFFFFFE80 に、WTCNT はアドレス H'FFFFFFE81 に、RSTCSR はアドレス H'FFFFFFE83 に割り当てられています。読み出すときは、必ずバイト転送命令を使用してください。

## 13.3 動作説明

### 13.3.1 ウォッチドッグタイマモード時の動作

ウォッチドッグタイマとして使用するときには、ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSCR) の  $WT/\overline{IT}$  ビットと TME ビットの両方を 1 に設定してください。また、ウォッチドッグタイマカウンタ (WTCNT) がオーバーフローする前に必ず WTCNT の値を書き換えて (通常は H'00 を書き込む)、オーバーフローを発生させないようにプログラムしてください。このようにすると、システムが正常に動作している間は、WTCNT のオーバーフローが発生しませんが、システムの暴走などにより WTCNT の値が書き換えられずオーバーフローすると、 $\overline{WDTOVF}$  が外部に出力されます。これを図 13.4 に示します。この  $\overline{WDTOVF}$  信号を用いて、システムをリセットすることができます。 $\overline{WDTOVF}$  信号は、512 クロックの間出力されます。

リセットコントロール/ステータスレジスタ (RSTCSR) の RSTE ビットを 1 にセットしておく、WTCNT がオーバーフローしたときに、 $\overline{WDTOVF}$  信号と同時に、本 LSI の内部をリセットする信号が発生します。このリセットは、RSTCSR の RSTS ビットの設定によって、パワーオンリセットまたはマニュアルリセットが選択できます。内部リセット信号は、2048 クロックの間出力されます。

$\overline{RES}$  端子からの入力信号によるリセットと WDT のオーバーフローによるリセットが同時に発生したときは、 $\overline{RES}$  端子によるリセットが優先され、RSTCSR の WOVF ビットは 0 にクリアされます。

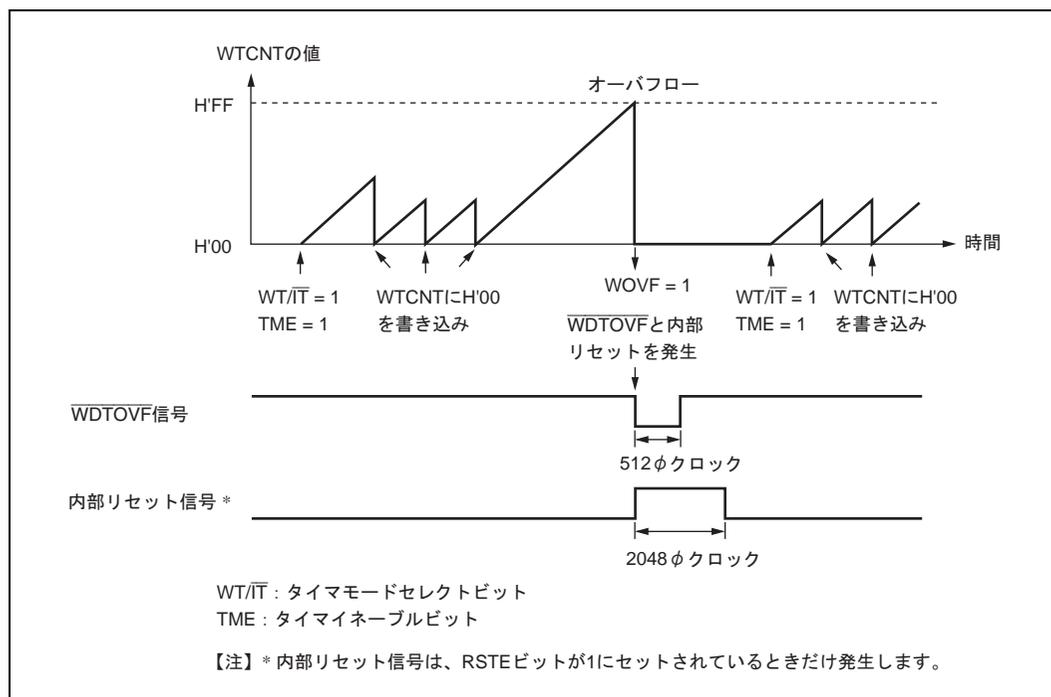


図 13.4 ウォッチドッグタイマモード時の動作

### 13.3.2 インターバルタイマモード時の動作

インターバルタイマとして使用するときは、ウォッチドッグタイマコントロール/ステータスレジスタ (WTCNR) の  $WT/\overline{IT}$  ビットを 0 に、TME ビットを 1 に設定してください。インターバルタイマとして動作しているときは、図 13.5 に示すように、ウォッチドッグタイマカウンタ (WTCNT) がオーバーフローするごとにインターバルタイマ割り込み (ITI) が発生します。したがって、一定時間ごとに、割り込みを発生させることができます。

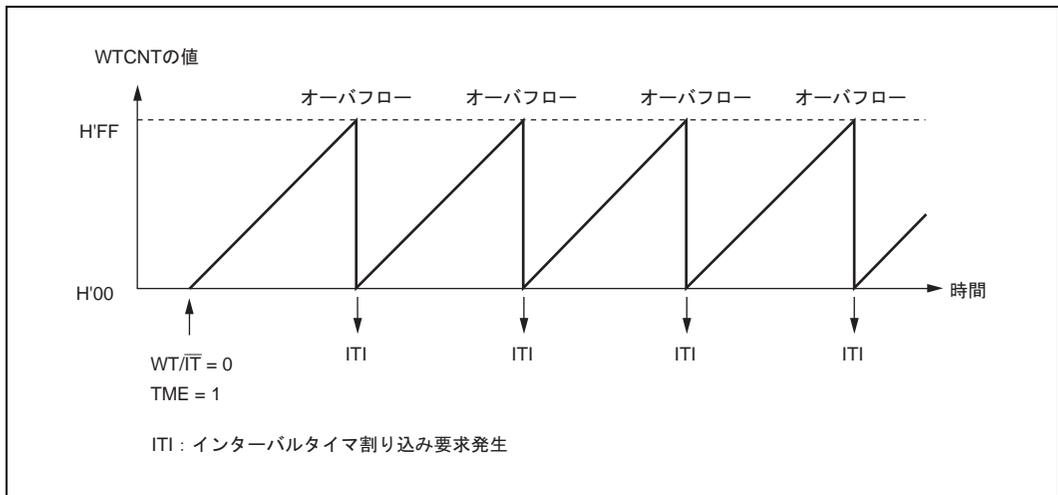


図 13.5 インターバルタイマモード時の動作

### 13.3.3 スタンバイモード解除時の動作

WDT は、スタンバイモードが NMI 割り込みで解除されるときに使用されます。スタンバイモードを使用する場合は、WDT を次の (1) に示すように設定してください。

#### (1) スタンバイモード遷移前の設定

スタンバイモードに遷移する前に、必ずウォッチドッグタイマコントロール/ステータスレジスタ (WTCNR) の TME ビットを 0 にして、WDT を停止させてください。TME ビットが 1 になっていると、スタンバイモードに遷移できません。また、ウォッチドッグタイマカウンタ (WTCNT) のオーバーフロー周期が発振安定時間以上になるように、WTCNR の CKS2 ~ CKS0 ビットを設定してください。発振安定時間については、「21. 電気的特性」を参照してください。

#### (2) スタンバイモード解除時の動作

スタンバイモードで NMI 信号が入力されると、発振器が動作を開始し、WTCNT はスタンバイモード遷移前に CKS2 ~ CKS0 ビットで選択しておいたクロックにより、カウントアップを開始します。WTCNT がオーバーフロー (H'FF ~ H'00) すると、クロックが安定し使用可能であると判断され、本 LSI 全体にクロックが供給されます。これによって、スタンバイモードが解除されます。

スタンバイモードの詳細については、「20. 低消費電力状態」を参照してください。

### 13.3.4 オーバフローフラグ (OVF) のセットタイミング

インターバルタイマモードでウォッチドッグタイマカウンタ (WTCNT) がオーバフローするとウォッチドッグタイマコントロール/ステータスレジスタ (WTCSCR) の OVF ビットが 1 にセットされ、同時にインターバルタイマ割り込み (ITI) が要求されます。このタイミングを図 13.6 に示します。

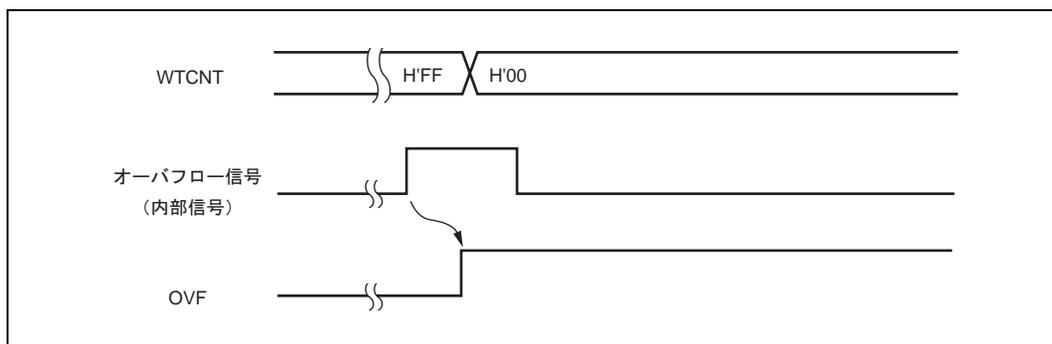


図 13.6 オーバフローフラグ (OVF) のセットタイミング

### 13.3.5 ウォッチドッグタイマオーバフローフラグ (WOVF) のセットタイミング

ウォッチドッグタイマモードでウォッチドッグタイマカウンタ (WTCNT) がオーバフローすると、リセットコントロール/ステータスレジスタ (RSTCSR) の WOVF ビットが 1 にセットされ、 $\overline{\text{WDTOVF}}$  信号が外部に出力されます。また、RSTCSR の RSTE ビットが 1 にセットしてあると、WTCNT がオーバフローしたとき、本 LSI 全体に対して内部リセット信号を発生します。これらのタイミングを図 13.7 に示します。

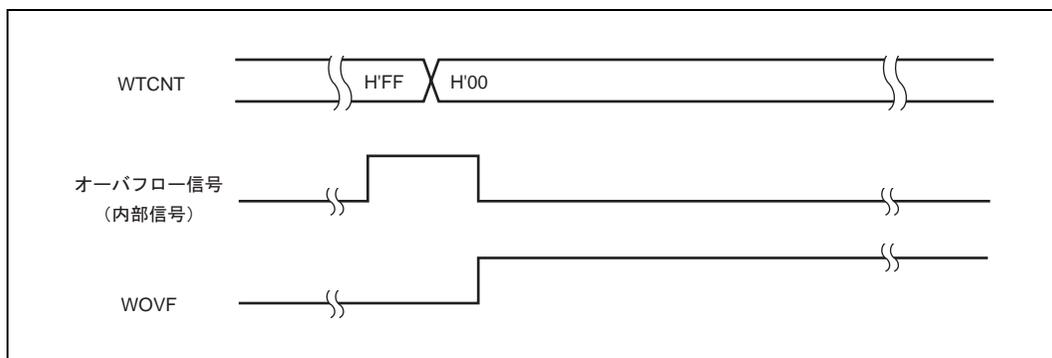


図 13.7 ウォッチドッグタイマオーバフローフラグ (WOVF) のセットタイミング

## 13.4 使用上の注意

### 13.4.1 ウォッチドッグタイマカウンタ (WTCNT) の書き込みとカウントアップの競合

ウォッチドッグタイマカウンタ (WTCNT) のライトサイクル中、図 13.8 に示すタイミングでカウントアップが発生しても、WTCNT へのデータ書き込みが優先され、カウントアップされません。これを図 13.8 に示します。

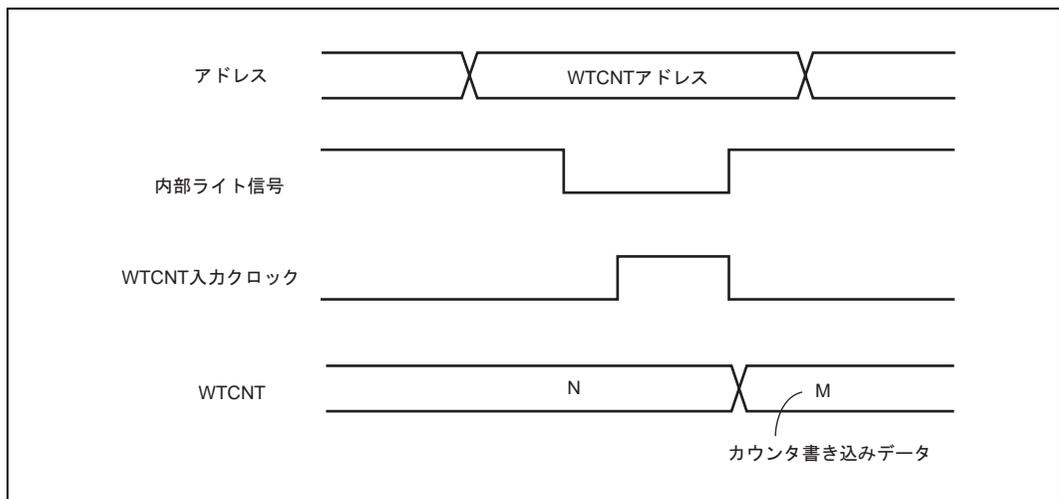


図 13.8 WTCNT の書き込みとカウントアップの競合

### 13.4.2 CKS2 ~ CKS0 ビットを書き換え

WDT の動作中にウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR) の CKS2 ~ CKS0 ビットを書き換えると、カウントアップが正しく行われず場合があります。CKS2 ~ CKS0 ビットを書き換えるときは、必ず WDT を停止させてから (TME ビットを 0 にクリアしてから) 行ってください。

### 13.4.3 ウォッチドッグタイマモードとインターバルタイマモードの切り替え

WDT の動作中にウォッチドッグタイマモードとインターバルタイマモードを切り替えると、正しい動作が行われず場合があります。タイマモードの切り替えは、必ず WDT を停止させてから (TME ビットを 0 にクリアしてから) 行ってください。

### 13.4.4 $\overline{\text{WDTOVF}}$ 信号によるシステムのリセット

$\overline{\text{WDTOVF}}$  出力信号を本 LSI の  $\overline{\text{RES}}$  端子に入力すると、本 LSI を正しく初期化できません。

$\overline{\text{WDTOVF}}$  信号は、本 LSI の  $\overline{\text{RES}}$  端子に論理的に入力しないようにしてください。 $\overline{\text{WDTOVF}}$  信号でシステム全体をリセットするときは、図 13.9 に示すような回路で行ってください。

### 13. ウォッチドッグタイマ (WDT)

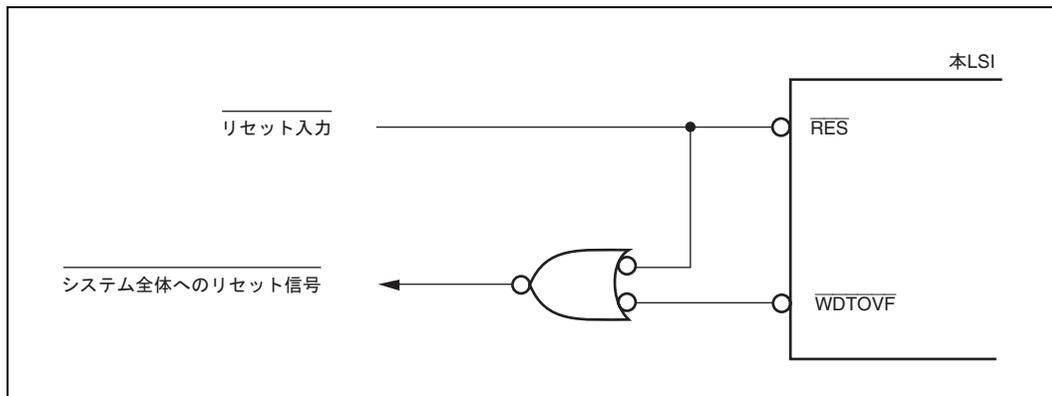


図 13.9 WDTOVF 信号によるシステムリセット回路例

#### 13.4.5 ウォッチドッグタイマモードでの内部リセット

ウォッチドッグタイマモード時に RSTE ビットを 0 にしておくと、WTCNT がオーバーフローしても本 LSI 内部をリセットしませんが、WDT 内の WTCNT、WTCSR はリセットされます。

#### 13.4.6 スリープモード中のウォッチドッグタイマ (WDT) による内部リセット動作について

ウォッチドッグタイマモード時にウォッチドッグタイマカウンタ (WTCNT) がオーバーフローした際、内部リセット (パワーオンリセット、もしくはマニュアルリセット) を行う仕様があります。ところが、この WTCNT オーバーフローがスリープモード中に発生した場合、内部リセットを正しく実行することができず、リセットによる例外処理が開始できない場合があります。

- 条件
  - スリープモード中
  - WDT.WTCSR.WT/IT ビット = 1 (ウォッチドッグタイマモード)
  - WDT.RSTCSR.RSTE ビット = 1 (内部リセットイネーブル)
  - WTCNT のオーバーフロー発生

#### 対策

ウォッチドッグタイマモードで内部リセットを使用する際は、下記のように対応ください。

- (1) スリープモードを使用しない場合、本機能をご使用ください。
- (2) スリープモードを使用する場合、本機能を使用せず、 $\overline{\text{RES}}$  端子によりリセットを行ってください。  
ただし、 $\overline{\text{WDTOVF}}$  出力信号を本 LSI の  $\overline{\text{RES}}$  端子に論理的に入力しないようにしてください。

---

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

---

### 14.1 概要

本 LSI には、FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF: SCI with FIFO) が 2 チャンネル内蔵されています。SCIF は、調歩同期式通信とクロック同期式通信の 2 方式でシリアル通信ができます。また、複数のプロセッサ間のシリアル通信機能 (マルチプロセッサ通信機能) を備えています。

IrDA1.0 システムに基づく赤外線データアソシエーション (IrDA) インタフェースを内蔵しているので赤外線通信を行うことができます。

送信、受信ともに 16 段の FIFO レジスタを内蔵しており、効率の良い高速連続通信を行うことができます。

#### 14.1.1 特長

SCIF には次の特長があります。

- シリアル通信モードを調歩同期式モード、またはクロック同期式モードから選択可能

- 調歩同期式モード

キャラクタ単位で同期をとる調歩同期方式でシリアルデータの通信を行います。Universal Asynchronous Receiver/Transmitter (UART) や Asynchronous Communication Interface Adapter (ACIA) など標準の調歩同期式通信用 LSI とのシリアルデータ通信が可能です。また、複数のプロセッサとシリアルデータ通信ができるマルチプロセッサ間通信機能を備えています。

シリアルデータ通信フォーマットを 12 種類のフォーマットから選択できます。

- データ長 : 7 ビット、または 8 ビット
- ストップビット長 : 1 ビット、または 2 ビット
- パリティ : 偶数パリティ、奇数パリティ、またはパリティなし
- マルチプロセッサビット : 1 または 0
- 受信エラーの検出 : パリティエラー、オーバランエラー、フレーミングエラーを検出
- ブレークの自動検出が可能です。

- クロック同期式モード

クロックに同期してシリアルデータ通信を行います。クロック同期式通信機能を持つ他の LSI とのシリアルデータ通信が可能です。

シリアルデータ通信フォーマットは 1 種類です。

- データ長 : 8 ビット
- 受信エラーの検出 : オーバランエラーを検出

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

---

- IrDA1.0 準拠
- 全二重通信が可能

独立した送信部と受信部を備えているので、送信と受信を同時に行うことができます。

また、送信部、受信部ともに 16 段の FIFO バッファ構造になっていますのでシリアルデータの連続送信、連続受信ができます。

ただし、IrDA 通信時には半二重通信を行います。

- 内蔵ボーレートジェネレータにより任意のビットレートを選択可能
- 送受信クロックソースを、ボーレートジェネレータからの内部クロック、または SCK 端子からの外部クロックから選択可能
- 4 種類の割り込み要因

送信 FIFO データエンプティ、ブ레이크、受信 FIFO データフル、受信エラーの 4 種類の割り込み要因があり、それぞれ独立に要求することができます。また、送信 FIFO データエンプティ、受信 FIFO データフル割り込みにより、内蔵 DMAC を起動させてデータ転送を行うことができます。

- SCIF を使用しないときは、消費電力低減のため SCIF に対してクロックの供給を止めて動作を停止させることができます。
- LSB ファースト、MSB ファーストを選択可能
- 調歩同期式モードにおいて、ビットレートの 4、8、16 倍の基本クロックでの動作を選択可能
- 内蔵モデムコントロール機能 ( $\overline{\text{RTS}}$  および  $\overline{\text{CTS}}$ )

14.1.2 ブロック図

図 14.1 に SCIF のブロック図を、図 14.2 に IrDA のブロック図を示します。

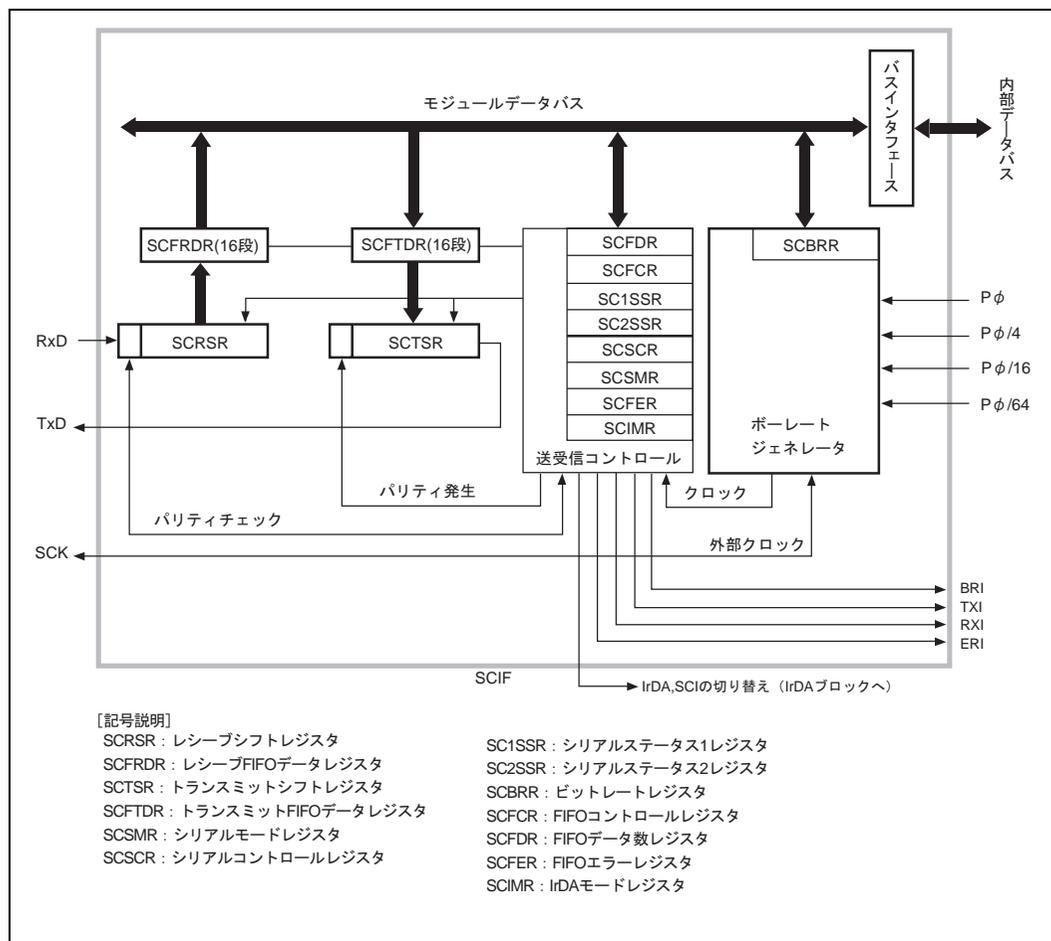


図 14.1 SCIF のブロック図

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

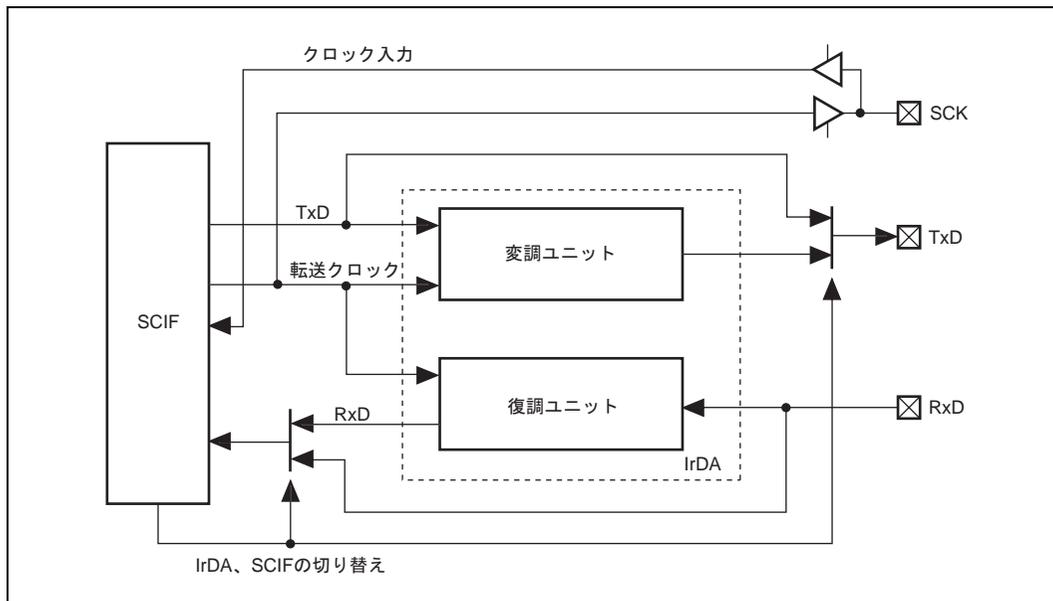


図 14.2 IrDA のブロック図

### 14.1.3 端子構成

SCIF は、チャンネルごとに表 14.1 に示すシリアル端子を持っています。

表 14.1 端子構成

チャンネル	端子名	略 称	入出力	機 能
1	シリアルクロック端子	SCK1	入出力	クロック入出力
	受信データ端子	RxD1	入力	受信データ入力
	送信データ端子	TxD1	出力	送信データ出力
	送信要求端子	RTS	出力	送信要求
	送信許可端子	CTS	入力	送信許可
2	シリアルクロック端子	SCK2	入出力	クロック入出力
	受信データ端子	RxD2	入力	受信データ入力
	送信データ端子	TxD2	出力	送信データ出力

## 14.1.4 レジスタ構成

SCIF には、表 14.2 に示す内部レジスタがあります。これらのレジスタにより調歩同期式モード / クロック同期式モードおよび IrDA 通信モードの指定、データフォーマットの指定、ビットレートの指定、および送信部 / 受信部の制御を行うことができます。

表 14.2 レジスタ構成

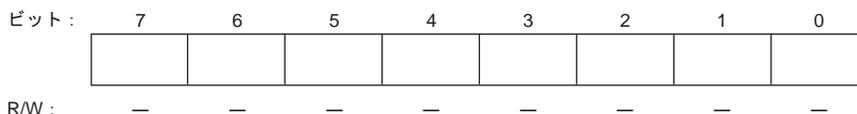
チャンネル	名 称	略 称	R/W	初期値	アドレス	アクセス サイズ
1	シリアルモードレジスタ	SCSMR1	R/W	H'00	H'FFFFFFC0	8
	ビットレートレジスタ	SCBRR1	R/W	H'FF	H'FFFFFFC2	8
	シリアルコントロールレジスタ	SCSCR1	R/W	H'00	H'FFFFFFC4	8
	トランスミット FIFO データレジスタ	SCFTDR1	W	-	H'FFFFFFC6	8
	シリアルステータス 1 レジスタ	SC1SSR1	R/(W)*	H'0060	H'FFFFFFC8	16
	シリアルステータス 2 レジスタ	SC2SSR1	R/(W)*	H'20	H'FFFFFFCA	8
	レシーブ FIFO データレジスタ	SCFRDR1	R	不定	H'FFFFFFCC	8
	FIFO コントロールレジスタ	SCFCR1	R/W	H'00	H'FFFFFFCE	8
	FIFO データ数レジスタ	SCFDR1	R	H'0000	H'FFFFFFCD0	16
	FIFO エラーレジスタ	SCFER1	R	H'0000	H'FFFFFFCD2	16
	IrDA モードレジスタ	SCIMR1	R/W	H'00	H'FFFFFFCD4	8
	2	シリアルモードレジスタ	SCSMR2	R/W	H'00	H'FFFFFFE0
ビットレートレジスタ		SCBRR2	R/W	H'FF	H'FFFFFFE2	8
シリアルコントロールレジスタ		SCSCR2	R/W	H'00	H'FFFFFFE4	8
トランスミット FIFO データレジスタ		SCFTDR2	W	-	H'FFFFFFE6	8
シリアルステータス 1 レジスタ		SC1SSR2	R/(W)*	H'0060	H'FFFFFFE8	16
シリアルステータス 2 レジスタ		SC2SSR2	R/(W)*	H'20	H'FFFFFFEA	8
レシーブ FIFO データレジスタ		SCFRDR2	R	不定	H'FFFFFFEC	8
FIFO コントロールレジスタ		SCFCR2	R/W	H'00	H'FFFFFFEE	8
FIFO データ数レジスタ		SCFDR2	R	H'0000	H'FFFFFFF0	16
FIFO エラーレジスタ		SCFER2	R	H'0000	H'FFFFFFF2	16
IrDA モードレジスタ		SCIMR2	R/W	H'00	H'FFFFFFF4	8

【注】 \* フラグをクリアするために 0 のみ書き込むことができます。また、アクセスサイズ 8 のレジスタはバイトアクセス、アクセスサイズ 16 のレジスタはワードアクセスにしてください。

## 14.2 レジスタの説明

IrDA 通信モードでは、IrDA モードレジスタ (SCIMR) とシリアルモードレジスタ (SCSMR) のビット 6 ~ 3 (ICK3 ~ 0) 以外は、調歩同期モードの設定と同じです。

### 14.2.1 レシーブシフトレジスタ (SCRSR)

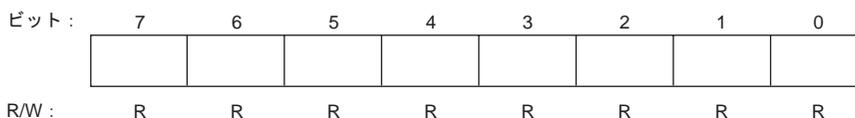


レシーブシフトレジスタ (SCRSR) は、シリアルデータを受信するためのレジスタです。

SCIF は、SCRSR に RxD 端子から入力されたシリアルデータを LSB (ビット 0) または MSB (ビット 7) から受信した順にセットし、パラレルデータに変換します。1 バイトのデータ受信を終了すると、データは自動的にレシーブ FIFO データレジスタ (SCFRDR) へ転送されます。

直接 SCRSR の読み出し / 書き込むことはできません。

### 14.2.2 レシーブ FIFO データレジスタ (SCFRDR)



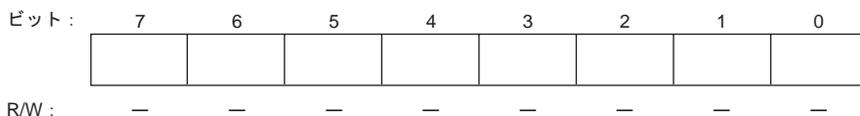
レシーブ FIFO データレジスタ (SCFRDR) は、受信したシリアルデータを格納する 8 ビット長の 16 段 FIFO レジスタです。

SCIF は、1 バイトのシリアルデータの受信が終了すると、レシーブシフトレジスタ (SCRSR) から SCFRDR へ受信したシリアルデータを転送して格納し、受信動作を完了します。この後、SCRSR は受信可能になり、SCFRDR がいっぱいになる 16 データまで連続した受信動作が可能です。

SCFRDR は、読み出し専用レジスタですので書き込むことはできません。

また、SCFRDR に受信データが無い状態で読み出した値は不定値になります。SCFRDR 内の受信データがいっぱいになると、以降のシリアルデータは失われます。

## 14.2.3 トランスミットシフトレジスタ (SCTSR)



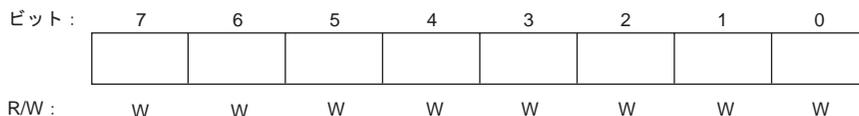
トランスミットシフトレジスタ (SCTSR) は、シリアルデータを送信するためのレジスタです。

SCIF は、トランスミット FIFO データレジスタ (SCFTDR) から送信データをいったん SCTSR に転送し、LSB (ビット 0) または MSB (ビット 7) から順に TxD 端子に送り出すことでシリアルデータ送信を行います。

1 バイトのデータ送信を終了すると自動的に SCFTDR から SCTSR へ次の送信データを転送し、送信を開始します。

直接 SCTSR の読み出し / 書き込みをすることはできません。

## 14.2.4 トランスミット FIFO データレジスタ (SCFTDR)



トランスミット FIFO データレジスタ (SCFTDR) は、シリアル送信するデータを格納する 8 ビット長の 16 段 FIFO レジスタです。

SCIF は、トランスミットシフトレジスタ (SCTSR) の空を検出すると、SCFTDR に書き込まれた送信データを SCTSR に転送してシリアル送信を開始します。SCFTDR の送信データが空になるまで、連続シリアル送信を行います。

SCFTDR は、書き込み専用レジスタですので読み出すことはできません。

SCFTDR 内の送信データが 16 バイトで満たされているときは次のデータを書き込むことはできません。書き込んだデータは無視されます。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

### 14.2.5 シリアルモードレジスタ (SCSMR)

ビット:	7	6	5	4	3	2	1	0
	C/ $\bar{A}$	CHR/ ICK3	PE/ICK2	O/ $\bar{E}$ /ICK1	STOP/ ICK0	MP	CKS1	CKS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

シリアルモードレジスタ (SCSMR) は、SCIF のシリアル通信フォーマットの設定と、ポーレートジェネレータのクロックソースを選択するための 8 ビットのレジスタです。IrDA 通信モードでは、出力パルス幅の選択に使用します。

SCSMR は、常に CPU による読み出し / 書き込みが可能です。

SCSMR は、リセット、モジュールスタンバイ、またはスタンバイモード時に H'00 に初期化されます。

- ・ ビット 7: コミュニケーションモード (C/ $\bar{A}$ )

SCIF の動作モードを調歩同期式モードとクロック同期式モードのいずれかから選択します。IrDA 通信モード時は、必ず 0 を設定してください。

ビット 7	説 明
C/ $\bar{A}$	
0	調歩同期式モード (初期値)
1	クロック同期式モード

- ・ ビット 6: キャラクタレンクス (CHR) / IrDA クロックセレクト 3 (ICK3)

調歩同期式モードのデータ長を 7 ビット / 8 ビットデータのいずれかから選択します。クロック同期式モードでは CHR の設定にかかわらず、データ長は 8 ビットデータ固定です。

ビット 6	説 明
CHR	
0	8 ビットデータ (初期値)
1	7 ビットデータ*

【注】 \* 7 ビットデータを選択した場合、トランスミット FIFO データレジスタ (SCFTDR) の MSB (ビット 7) は送信されません。

IrDA 通信モードでは、IrDA クロックセレクト 3 (ICK3) となり、設定により適切なクロックパルスを発生させることができます。「14.3.6 IrDA モード時の動作 (3) パルス幅の選択」を参照してください。

・ ビット 5 : パリティイネーブル (PE) / IrDA クロックセレクト 2 (ICK2)

調歩同期式モードでは、送信時にパリティビットの付加を、受信時にパリティビットのチェックを行うかどうかを選択します。クロック同期式モードでは、PE ビットの設定にかかわらずパリティビットの付加、およびチェックは行いません。

ビット 5	説 明
PE	
0	パリティビットの付加、およびチェックを禁止 (初期値)
1	パリティビットの付加、およびチェックを許可*

【注】 \* PE ビットに 1 をセットすると送信時には、O/E ビットで指定した偶数、または奇数パリティを送信データに付加して送信します。受信時には、受信したパリティビットが O/E ビットで指定した偶数、または奇数パリティになっているかどうかをチェックします。

IrDA 通信モードでは、IrDA クロックセレクト 2 (ICK2) となり、設定により適切なクロックパルスを発生させることができます。「14.3.6 IrDA モード時の動作 (3) パルス幅の選択」を参照してください。

・ ビット 4 : パリティモード (O/E) / IrDA クロックセレクト 1 (ICK1)

パリティの付加やチェックを偶数パリティ、または奇数パリティのいずれで行うかを選択します。O/E ビットの設定は、調歩同期式モードで PE ビットに 1 を設定しパリティビットの付加やチェックを許可したときのみ有効になります。クロック同期式モードや、調歩同期式モードでパリティの付加やチェックを禁止している場合には、O/E ビットの指定は無効です。

ビット 4	説 明
O/E	
0	偶数パリティ* <sup>1</sup> (初期値)
1	奇数パリティ* <sup>2</sup>

【注】 \*<sup>1</sup> 偶数パリティに設定すると送信時には、パリティビットと送信キャラクタを合わせたうち、1 の数の合計が偶数になるようにパリティビットを付加して送信します。

受信時には、パリティビットと受信キャラクタを合わせたうち、1 の数の合計が偶数であるかどうかをチェックします。

\*<sup>2</sup> 奇数パリティに設定すると送信時には、パリティビットと送信キャラクタを合わせたうち、1 の数の合計が奇数になるようにパリティビットを付加して送信します。

受信時には、パリティビットと受信キャラクタを合わせたうち、1 の数の合計が奇数であるかどうかをチェックします。

IrDA 通信モードでは、IrDA クロックセレクト 1 (ICK1) となり、設定により適切なクロックパルスを発生させることができます。「14.3.6 IrDA モード時の動作 (3) パルス幅の選択」を参照してください。

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

##### ・ ビット 3 : ストップビット長 (STOP) / IrDA クロックセレクト 0 (ICK0)

調歩同期式モードでのストップビットの長さを 1 ビット / 2 ビットのいずれかから選択します。STOP ビットの設定は調歩同期式モードでのみ有効になります。クロック同期式モードに設定した場合にはストップビットは付加されませんので、このビットの設定は無効です。

ビット 3	説 明	
STOP		
0	1 ストップビット *1	(初期値)
1	2 ストップビット *2	

【注】 \*1 送信時には、送信キャラクタの最後尾に 1 ビットの 1 (ストップビット) を付加して送信します。

\*2 送信時には、送信キャラクタの最後尾に 2 ビットの 1 (ストップビット) を付加して送信します。

なお、受信時には STOP ビットの設定にかかわらず、受信したストップビットの 1 ビット目のみをチェックします。ストップビットの 2 ビット目が 1 の場合は、ストップビットとして扱いますが、0 の場合は、次の送信キャラクタのスタートビットとして扱います。

IrDA 通信モードでは、IrDA クロックセレクト 0 (ICK0) となり、設定により適切なクロックパルスを発生させることができます。「14.3.6 IrDA モード時の動作 (3) パルス幅の選択」を参照してください。

##### ・ ビット 2 : マルチプロセッサモード (MP)

マルチプロセッサフォーマットを選択します。マルチプロセッサフォーマットを選択した場合、PE ビット、および O/E ビットにおけるパリティの設定は無効になります。また、MP ビットの設定は、調歩同期式モードの時のみ有効です。クロック同期式モード、IrDA モードでは、MP ビットの設定は無効です。

マルチプロセッサ通信機能については、「14.3.3 マルチプロセッサの通信機能」を参照してください。

ビット 2	説 明	
MP		
0	マルチプロセッサ機能を禁止	(初期値)
1	マルチプロセッサフォーマットを選択	

##### ・ ビット 1、0 : クロックセレクト 1、0 (CKS1、CKS0)

内蔵ポーレートジェネレータのクロックソースを選択します。CKS1、CKS0 ビットの設定で P、P /4、P /16、P /64 の 4 種類からクロックソースを選択できます。

クロックソースと、ビットレートレジスタの設定値、およびポーレートの関係については、「14.2.9 ビットレートレジスタ (SCBRR)」を参照してください。

ビット 1	ビット 0	説 明	
CKS1	CKS0		
0	0	P クロック	(初期値)
	1	P /4 クロック	
1	0	P /16 クロック	
	1	P /64 クロック	

【注】 P = 周辺クロック

## 14.2.6 シリアルコントロールレジスタ (SCSCR)

ビット:	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	—	CKE1	CKE0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

シリアルコントロールレジスタ (SCSCR) は、SCIF の送信 / 受信動作、調歩同期モードでのシリアルクロック出力、割り込み要求の許可 / 禁止、および送信 / 受信クロックソースの選択を行うレジスタです。

SCSCR は、常に CPU による読み出し / 書き込みが可能です。

SCSCR は、リセット、モジュールスタンバイ、またはスタンバイモード時に H'00 に初期化されます。

・ ビット7: トランスミットインタラプトイネーブル (TIE)

トランスミット FIFO データレジスタ (SCFTDR) からトランスミットシフトレジスタ (SCTSR) へシリアル送信データが転送され、SCFTDR 内のデータ数が送信トリガ設定数以下となり、シリアルステータス1レジスタ (SC1SSR) の TDFE フラグが1にセットされたときに、送信 FIFO データエンプティ割り込み (TXI) 要求の発生を許可 / 禁止します。

ビット7	説明
TIE	
0	送信 FIFO データエンプティ割り込み (TXI) 要求を禁止* (初期値)
1	送信 FIFO データエンプティ割り込み (TXI) 要求を許可

【注】 \* TXI の解除は、SCFTDR に送信トリガ設定数より多い送信データを書き込み、TDFE フラグの1を読み出した後、0にクリアするか、または TIE を0にクリアすることで行うことができます。内蔵 DMAC を使用して SCFTDR に送信データを書き込む場合は、TDFE フラグは自動的にクリアされます。

・ ビット6: レシーブインタラプトイネーブル (RIE)

シリアル受信データがレシーブシフトレジスタ (SCRSR) からレシーブ FIFO データレジスタ (SCFRDR) へ転送されて SCFRDR 内のデータ数が受信トリガ設定数以上になり、SC1SSR の RDF フラグが1にセットされたときの、受信 FIFO データフル割り込み (RXI) 要求、および受信エラー割り込み (ERI) 要求、およびブ레이크割り込み (BRI) 要求の発生を許可 / 禁止します。

ビット6	説明
RIE	
0	受信 FIFO データフル割り込み (RXI) 要求、および受信エラー割り込み (ERI) 要求およびブ레이크割り込み (BRI) 要求を禁止* (初期値)
1	受信 FIFO データフル割り込み (RXI) 要求、および受信エラー割り込み (ERI) 要求およびブ레이크割り込み (BRI) 要求を許可*

【注】 \* RXI、ERI、および BRI 割り込み要求の解除は、RDF、DR フラグ、FER、PER、ORER、ER フラグ、または BRK フラグの1を読み出した後、0にクリアするか、RIE ビットを0にクリアすることで行えます。RDF フラグでは、受信データ数が受信トリガの設定数より小さくなるまで SCFRDR から受信データを読み出した後、RDF フラグの1を読み出し0にクリアします。

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

##### ・ビット5：トランスミットイネーブル (TE)

SCIF のシリアル送信動作の開始を許可 / 禁止します。

ビット5	説 明	
TE		
0	送信動作を禁止 *1	(初期値)
1	送信動作を許可 *2	

【注】 \*1 SC1SSR の TDFE フラグは 1 に固定されます。

\*2 この状態で、SCFTDR に送信データを書き込むと、シリアル送信を開始します。

なお、TE ビットを 1 にセットする前に必ずシリアルモードレジスタ (SCSMR)、FIFO コントロールレジスタ (SCFCR) の設定を行い送信フォーマットを決定し、FIFO バッファをリセットしてください。

##### ・ビット4：レシーブイネーブル (RE)

SCIF のシリアル受信動作の開始を許可 / 禁止します。

ビット4	説 明	
RE		
0	受信動作を禁止 *1	(初期値)
1	受信動作を許可 *2	

【注】 \*1 RE ビットを 0 にクリアしても RDF、DR、FER、PER、ORER、ER、BRK の各フラグは影響を受けず、状態を保持しますので注意してください。

\*2 この状態で調歩同期式モードの場合はスタートビットを、クロック同期式モードの場合は同期クロック入力をそれぞれ検出すると、シリアル受信を開始します。

なお、RE ビットを 1 にセットする前に必ず SCSMR の設定を行い、受信フォーマットを決定してください。

##### ・ビット3：マルチプロセッサインタラプトイネーブル (MPIE)

マルチプロセッサ割り込みを許可 / 禁止します。MPIE ビットの設定は、調歩同期式モードで、かつ、SCSMR の MP ビットが 1 に設定されている受信時にのみ有効です。

クロック同期式モード、IrDA モードのとき、あるいは MP ビットが 0 のときには MPIE ビットの設定は無効です。

ビット3	説 明	
MPIE		
0	マルチプロセッサ割り込み禁止状態 (通常の実行動作をします) [クリア条件] (1) MPIE ビットを 0 にクリア (2) MPB = 1 のデータを受信したとき	(初期値)
1	マルチプロセッサ割り込み許可状態 * マルチプロセッサビットが 1 のデータを受け取るまで受信割り込み (RXI) 要求、受信エラー割り込み (ERI) 要求、および SC1SSR の RDF、FER、SC2SSR の ORER の各フラグのセットを禁止します。	

【注】 \* SCRSR から SCFRDR への受信データの転送、および受信エラーの検出と SC1SSR の RDF、FER、SC2SSR の ORER の各フラグのセットは行いません。MPB = 1 を含む受信データを受信すると、SC2SSR の MPB フラグを 1 にセットし、MPIE ビットを自動的に 0 にクリアし、RXI、ERI の発生 (SCSCR の RIE ビットが 1 にセットされている場合) と FER、ORER フラグのセットが許可されます。

##### ・ビット2：予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

・ ビット1、0：クロックイネーブル1、0 (CKE1、CKE0)

SCIFのクロックソースの選択、およびSCK端子からのクロック出力の許可/禁止を設定します。CKE1ビットとCKE0ビットの組み合わせによってSCK端子をシリアルクロック出力端子にするか、またはシリアルクロック入力端子にするかが決まります。このとき、ピンファンクションコントローラ (PFC) で、SCK端子の機能を選択しておいてください。

ただし、CKE0ビットの設定は調歩同期式モードで内部クロック動作 (CKE1 = 0) 時のみ有効です。クロック同期式モードのとき、および外部クロック動作 (CKE1 = 1) の場合はCKE0ビットの設定は無効です。また、SCSMRでSCIFの動作モードを決定する前に、必ずCKE1、CKE0ビットの設定をしてください。

SCIFのクロックソースの選択についての詳細は「14.3 動作説明」の表14.9を参照してください。

ビット1	ビット0	説 明	
CKE1	CKE0		
0	0	調歩同期式モード	内部クロック / SCK端子は入力端子 (入力信号は無視) *1
	1	クロック同期式モード	内部クロック / SCK端子は同期クロック出力 *1
1		調歩同期式モード	内部クロック / SCK端子はクロック出力 *2
	1	クロック同期式モード	内部クロック / SCK端子は同期クロック出力
1		*4	調歩同期式モード
	1	*4	クロック同期式モード

- 【注】
- \*1 初期値
  - \*2 ビットレートの16/8/4倍の周波数のクロックを出力
  - \*3 ビットレートの16/8/4倍の周波数のクロックを入力
  - \*4 Don't Care.

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

### 14.2.7 シリアルステータス 1 レジスタ (SC1SSR)

ビット:	15	14	13	12	11	10	9	8
	PER3	PER2	PER1	PER0	FER3	FER2	FER1	FER0

初期値: 0 0 0 0 0 0 0 0

R/W: R R R R R R R R

ビット:	7	6	5	4	3	2	1	0
	ER	TEND	TDFE	BRK	FER	PER	RDF	DR

初期値: 0 1 1 0 0 0 0 0

R/W: R/(W)\* R R/(W)\* R/(W)\* R R R/(W)\* R/(W)\*

【注】\*フラグをクリアするための0のみ書き込むことができます。

シリアルステータス 1 レジスタ (SC1SSR) は、16 ビット長のレジスタであり、下位 8 ビットは SCIF の動作状態を示すステータスフラグ、上位 8 ビットはレシーブ FIFO レジスタ内のデータのエラー数を示します。

SC1SSR は常に読み出し / 書き込みができます。ただし、ER、TDFE、BRK、RDF、DR の各ステータスフラグへ 1 を書き込むことはできません。また、これらを 0 にクリアするためには、あらかじめ 1 を読み出ししておく必要があります。また、TEND、FER、PER の各フラグは読み出し専用であり、書き込むことはできません。

SC1SSR は、リセット、モジュールスタンバイ、またはスタンバイモード時に H'0084 に初期化されます。

- ビット 15 ~ 12: パリティエラー数 3 ~ 0 (PER3 ~ PER0)

レシーブ FIFO データレジスタ内にある受信データで、パリティエラーであるデータの数を示します。

クリアはレシーブ FIFO データレジスタ内のすべての受信データを読み出すか、SCFCR の RFRST ビットを 1 にセットし、レシーブ FIFO データレジスタを空の状態にリセットします。

- ビット 11 ~ 8: フレーミングエラー数 3 ~ 0 (FER3 ~ FER0)

レシーブ FIFO データレジスタ内にある受信データで、フレーミングエラーであるデータの数を示します。

クリアはレシーブ FIFO データレジスタ内のすべての受信データを読み出すか、SCFCR の RFRST ビットを 1 にセットし、レシーブ FIFO データレジスタを空の状態にリセットします。

## ・ ビット7: レシーブエラー (ER)

ビット7	説明
ER	
0	受信中、または正常に受信を完了したことを表示 * <sup>1</sup> (初期値) [クリア条件] (1) リセットまたはスタンバイモード時 (2) ER=1の状態を読み出した後、0を書き込んだとき
1	受信時にフレーミングエラーまたはパリティエラーまたはオーバランエラーが発生したことを表示 [セット条件] (1) 受信終了時に受信データの最後尾のストップビットが1であるかどうかをチェックし、ストップビットが0であったとき * <sup>2</sup> (2) 受信時の受信データとパリティビットを合わせた1の数がシリアルモードレジスタ (SCSMR) の O/E ビットで指定した偶数/奇数パリティの設定と一致しなかったとき (3) SCFRDR 内の受信データ数が 16の状態での次のシリアル受信を完了したとき

【注】 \*<sup>1</sup> SCSCR の RE ビットを 0 にクリアしたときには、ER フラグは影響を受けず以前の状態を保持します。フレーミングエラーまたはパリティエラーが発生しても受信データは SCFRDR に転送され、EI ビットの設定により、その後受信動作を続けるかどうか選択します。オーバランエラーが発生したときは、受信データは SCFRDR に転送されず、受信動作を続けることはできません。

\*<sup>2</sup> 2ストップモードのときは1ビット目のストップビットが1であるかどうかのみを判定し、2ストップビット目のストップビットはチェックしません。

## ・ ビット6: トランスミットエンド (TEND)

送信キャラクタの最後尾ビットの送信時に SCFTDR に有効なデータがなく、送信を終了したことを示します。

ビット6	説明
TEND	
0	送信中であることを表示 [クリア条件] (1) TE = 1の状態では SCFTDR にデータを書き込んだとき
1	送信を終了したことを表示 (初期値) [セット条件] (1) リセット、またはスタンバイモード時 (2) SCSCR の TE ビットが0のとき (3) 1バイトのシリアル送信キャラクタの最後尾ビットの送信時に SCFTDR に送信データがないとき

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

##### ・ ビット 5 : トランスミット FIFO データエンブティ (TDFE)

トランスミット FIFO データレジスタ (SCFTDR) からトランスミットシフトレジスタ (SCTSR) にデータ転送が行われ、SCFTDR 内のデータ数が FIFO コントロールレジスタ (SCFCR) の TTRG1、TTRG0 ビットで設定した送信トリガ設定数以下になり、SCFTDR に送信データの書き込みが可能になったことを示します。

ビット 5	説 明
TDFE	
0	SCFTDR に送信トリガ設定数より多い送信データが書き込まれていることを表示 [クリア条件] (1) SCFTDR に送信トリガ設定数を越える送信データを書き込み、TDFE=1 の状態を読み出した後、0 を書き込んだとき (2) 内蔵 DMAC で SCFTDR に送信トリガ設定数を越える送信データを書き込んだとき
1	SCFTDR の送信データ数が送信トリガ設定数以下であることを表示 (初期値) [セット条件] (1) リセット、スタンバイモード時 (2) SCFTDR の送信データ数が送信動作によって送信トリガ設定数以下になったとき *

【注】 \* SCFTDR は 16 バイトの FIFO レジスタですので TDFE=0 で書き込むことができる最大データ数は、{ 16 (送信トリガ設定数) } になります。これより多くデータを書き込んだ場合無視されます。また、SCFTDR 内のデータ数は SCFCR の上位 8 ビットに示されます。

##### ・ ビット 4 : ブレーク検出 (BRK)

受信データのブレーク信号を検出したことを示します。

ビット 4	説 明
BRK	
0	ブレーク信号を受信していないことを表示 (初期値) [クリア条件] (1) リセットまたはスタンバイモード時 (2) BRK=1 の状態を読み出した後、0 を書き込んだとき
1	ブレーク信号を受信していることを表示 [セット条件] フレーミングエラーを伴うデータを受信したとき、次の受信データもすべてスペース 0 でフレーミングエラーが発生している場合

【注】 ブレーク検出すると検出後の受信データ (H'00) の SCFRDR 転送は停止します。ブレークが終了し、受信信号がマーク 1 に戻ると受信データの転送が再開します。

##### ・ ビット 3 : フレーミングエラー (FER)

レシーブ FIFO データレジスタ (SCFRDR) から読み出したデータのフレーミングエラーを表示します。

ビット 3	説 明
FER	
0	SCFRDR から読み出した受信データにフレーミングエラーがないことを表示 (初期値) [クリア条件] (1) リセット、またはスタンバイモード時 (2) SCFRDR 読み出しデータにフレーミングエラーなし
1	SCFRDR から読み出した受信データにフレーミングエラーが発生していることを表示 [セット条件] SCFRDR 読み出しデータにフレーミングエラーあり

・ ビット2：パリティエラー (PER)

調歩同期式モードで、レシーブ FIFO データレジスタ (SCFRDR) から読み出したデータのパリティエラーを表示します。

ビット2	説明
PER	
0	SCFRDR から読み出した受信データにパリティエラーがないことを表示 (初期値) [クリア条件] (1) リセット、またはスタンバイモード時 (2) SCFRDR 読み出しデータにパリティエラーなし
1	SCFRDR から読み出した受信データにパリティエラーが発生していることを表示 [セット条件] SCFRDR 読み出しデータにパリティエラーあり

・ ビット1：レシーブデータレジスタフル (RDF)

受信したデータがレシーブ FIFO データレジスタ (SCFRDR) 転送され、SCFRDR 内の受信データ数が、FIFO コントロールレジスタ (SCFCR) の RTRG1、RTRG0 ビットで設定した受信トリガ設定数以上になったことを示します。

ビット1	説明
RDF	
0	SCFRDR 内の受信データ数が受信トリガ設定数より少ないことを表示 (初期値) [クリア条件] (1) リセット、またはスタンバイモード時 (2) SCFRDR 内の受信データ数が受信トリガ設定数より少なくなるまで SCFRDR を読み出し、RDF=1 を読み出した後、0 を書き込んだとき (3) 内蔵 DMAC で SCFRDR の受信データ数が受信トリガ設定数より少なくなるまでデータを読み出したとき
1	SCFRDR 内の受信データ数が受信トリガ設定数以上であることを表示 [セット条件] SCFRDR に受信トリガ設定数以上の受信データが格納されたとき

【注】 SCFRDR は 16 バイトの FIFO レジスタです。RDF=1 で少なくとも受信トリガ設定数のデータを読み出すことができます。SCFRDR 内のデータを全て読み出し、さらに読み出すとデータは不定値になります。なお SCFRDR 内の受信データ数は SCFCR の下位 8 ビットに示されます。

・ ビット0：レシーブデータレディ (DR)

レシーブ FIFO データレジスタ (SCFRDR) に受信トリガ設定数未満のデータがあり、かつ最後に受信したデータのストップビットから 16ETU 時間以上、次のデータが来ないことを示します。

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

ビット0	説明
DR	
0	受信中または正常に受信完了して SCFRDR に受信データが残っていないことを表示 (初期値) [クリア条件] (1) リセットまたはスタンバイモード時 (2) 残っている受信データを全て読み出した後、0を書き込んだとき *1
1	次の受信データが来ず、SCFRDR に受信トリガ設定数未満のデータが残っていることを表示 [セット条件] SCFRDR に受信トリガ設定数未満のデータがあり、かつ最後に受信したデータのストップビットから 16ETU 時間以上次のデータが来ないとき *2

【注】 \*1 DR フラグのクリアの前に、残っている受信データをすべて読み出してください。

\*2 8ビット長 1ストップビットフォーマット時の 1.6 フレーム時間に相当します。

ETU : element time unit = s/bit

#### 14.2.8 シリアルステータス 2 レジスタ (SC2SSR)

ビット:	7	6	5	4	3	2	1	0
	TLM	RLM	N1	N0	MPB	MPBT	EI	ORER
初期値:	0	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/(W)*

【注】 \*フラグをクリアするための0のみ書き込むことができます。

シリアルステータス 2 レジスタ (SC2SSR) は、8 ビット長のレジスタです。

SC2SSR は常に読み出し / 書き込みができます。ただし、ORER フラグへ 1 を書き込むことはできません。また、0 にクリアするためには、あらかじめ 1 を読み出ししておく必要があります。SC2SSR は、リセット、モジュールスタンバイ、またはスタンバイモード時に H'20 に初期化されます。

- ・ ビット 7 : トランスミット LSB / MSB ファースト選択  
データ送信時に、LSB ファースト / MSB ファーストを選択します。

ビット 7	説明
TLM	
0	LSB ファーストで送信 (初期値)
1	MSB ファーストで送信

- ・ ビット 6 : レシーブ LSB / MSB ファースト選択  
データ受信時に、LSB ファースト / MSB ファーストを選択します。

ビット 6	説 明	
RLM		
0	LSB ファーストで受信	(初期値)
1	MSB ファーストで受信	

- ・ ビット 5、4 : クロック・ビットレート比  
基本クロックをビットレート比によって選択します。

ビット 5	ビット 4	説 明
N1	N0	
0	0	ビットレートの 4 倍の基本クロックで動作
	1	ビットレートの 8 倍の基本クロックで動作
1	0	ビットレートの 16 倍の基本クロックで動作 (初期値)
	1	設定禁止

- ・ ビット 3 : マルチプロセッサビット (MPB)  
調歩同期式モードで受信をマルチプロセッサフォーマットで行うときに、受信データ中のマルチプロセッサビットを格納します。

MPB フラグは、読み出し専用ですので、書き込むことはできません。

ビット 3	説 明	
MPB		
0	マルチプロセッサビットが 0 のデータを受信したことを表示 *	(初期値)
1	マルチプロセッサビットが 1 のデータを受信したことを表示	

【注】 \* マルチプロセッサフォーマットで RE ビットを 0 にクリアしたときには、以前の状態を保持します。

- ・ ビット 2 : マルチプロセッサビットトランスファ (MPBT)  
調歩同期式モードで送信をマルチプロセッサフォーマットで行うときに、送信データに付加するマルチプロセッサビットを格納します。

クロック同期式モード、IrDA モードやマルチプロセッサフォーマットでないとき、あるいは送信でないときには MPBT ビットの設定は無効です。

ビット 2	説 明	
MPBT		
0	マルチプロセッサビットが 0 のデータを送信	(初期値)
1	マルチプロセッサビットが 1 のデータを送信	

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

### ・ ビット 1: 受信データエラー無視イネーブル (EI)

受信データにフレーミングエラーまたはパリティエラーが発生したとき (ER=1)、受信動作を続けるかどうかを選択します。

ビット 1	説明
EI	
0	受信中にフレーミングエラーまたはパリティエラーが発生したとき (ER=1) 受信動作を止める (初期値)
1	受信中にフレーミングエラーまたはパリティエラーが発生したとき (ER=1) 受信動作を続ける

【注】 EI=0 のとき、SCFRDR の最後のデータだけはエラーを含んだデータになります。EI=1 のとき、受信データにエラーがあってもデータは SCFRDR に送られます。

### ・ ビット 0: オーバランエラー (ORER)

受信時にオーバランエラーが発生して異常終了したことを示します。

ビット 0	説明
ORER	
0	受信中、または正常に受信を完了したことを表示 *1 (初期値) [クリア条件] (1) リセット、またはスタンバイモード時 (2) ORER = 1 の状態を読み出した後、0 を書き込んだとき
1	受信時にオーバランエラーが発生したことを表示 *2 [セット条件] SCFRDR 内の受信データ数が 16 の状態で次のシリアル受信を完了したとき

【注】 \*1 SCSCR の RE ビットを 0 にクリアしたときには、ORER フラグは影響を受けず以前の状態を保持します。

\*2 SCFRDR ではオーバランエラーが発生する前の受信データを保持し、後から受信したデータが失われます。さらに、ORER = 1 にセットされた状態で、以降のシリアル受信を続けることはできません。なお、クロック同期モードでは、シリアル送信を続けることもできません。

## 14.2.9 ビットレートレジスタ (SCBRR)

ビット:	7	6	5	4	3	2	1	0
	<input type="checkbox"/>							
初期値:	1	1	1	1	1	1	1	1
R/W:	R/W							

ビットレートレジスタ (SCBRR) は、シリアルモードレジスタ (SCSMR) の CKS1、CKS0 ビットで選択されるポーレートジェネレータの動作クロックとあわせて、シリアル送信 / 受信のビットレートを設定する 8 ビットのレジスタです。

SCBRR は、常に CPU による読み出し / 書き込みが可能です。

SCBRR は、リセット、モジュールスタンバイまたはスタンバイモード時に H'FF に初期化されます。

SCBRR の設定値は以下の計算式で求められます。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

[調歩同期式モード]

$$N = \frac{P\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{ビットレートの16倍の基本クロックで動作時})$$

$$N = \frac{P\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{ビットレートの8倍の基本クロックで動作時})$$

$$N = \frac{P\phi}{16 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{ビットレートの4倍の基本クロックで動作時})$$

[クロック同期式モード]

$$N = \frac{P\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B : ビットレート (bit/s)

N : ポーレートジェネレータのSCBRRの設定値 ( $0 \leq N \leq 255$ )

Pφ : 周辺モジュール用動作周波数 (MHz)

n : ポーレートジェネレータ入力クロック (n=0、1、2、3)

(nとクロックの関係は、下表を参照してください)

n	クロック	SCSMR の設定値	
		CKS1	CKS0
0	P	0	0
1	P / 4		1
2	P / 16	1	0
3	P / 64		1

調歩同期式モードのビットレート誤差は、以下の計算式で求められます。

$$\text{誤差 (\%)} = \left\{ \frac{P\phi \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

(ビットレートの16倍の基本クロックで動作時)

調歩同期式モードのビットレート誤差は、以下の計算式で求められます。

$$\text{誤差 (\%)} = \left\{ \frac{P\phi \times 10^6}{(N+1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(ビットレートの8倍の基本クロックで動作時)

調歩同期式モードのビットレート誤差は、以下の計算式で求められます。

$$\text{誤差 (\%)} = \left\{ \frac{P\phi \times 10^6}{(N+1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100$$

(ビットレートの4倍の基本クロックで動作時)

14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

表 14.3 に調歩同期式モードの SCBRR の設定例を、表 14.4 にクロック同期式モードの SCBRR の設定例を示します。以下の表の値は、いずれもビットレートの 16 倍の基本クロックで動作する場合の例です。

表 14.3 ビットレートに対する SCBRR の設定例 [調歩同期式モード](1)

ビットレート (bit/s)	P (MHz)											
	2			2.097152			2.4576			3		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	1	141	0.03	1	148	- 0.04	1	174	- 0.26	1	212	0.03
150	1	103	0.16	1	108	0.21	1	127	0.00	1	155	0.16
300	0	207	0.16	0	217	0.21	0	255	0.00	1	77	0.16
600	0	103	0.16	0	108	0.21	0	127	0.00	0	155	0.16
1200	0	51	0.16	0	54	- 0.70	0	63	0.00	0	77	0.16
2400	0	25	0.16	0	26	1.14	0	31	0.00	0	38	0.16
4800	0	12	0.16	0	13	- 2.48	0	15	0.00	0	19	- 2.34
9600	0	6	- 6.99	0	6	- 2.48	0	7	0.00	0	9	- 2.34
19200	0	2	8.51	0	2	13.78	0	3	0.00	0	4	- 2.34
31250	0	1	0.00	0	1	4.86	0	1	22.88	0	2	0.00
38400	0	1	- 18.62	0	1	- 14.67	0	1	0.00			

ビットレート (bit/s)	P (MHz)											
	3.6864			4			4.9152			5		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	64	0.70	2	70	0.03	2	86	0.31	2	88	- 0.25
150	1	191	0.00	1	207	0.16	1	255	0.00	2	64	0.16
300	1	95	0.00	1	103	0.16	1	127	0.00	1	129	0.16
600	0	191	0.00	0	207	0.16	0	255	0.00	1	64	0.16
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	129	0.16
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	64	0.16
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	32	- 1.36
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	15	1.73
19200	0	5	0.00	0	6	- 6.99	0	7	0.00	0	7	1.73
31250				0	3	0.00	0	4	- 1.70	0	4	0.00
38400	0	2	0.00	0	2	8.51	0	3	0.00	0	3	1.73

表 14.3 ビットレートに対する SCBRR の設定例 (調歩同期式モード) (2)

ビットレート (bit/s)	P (MHz)											
	6			6.144			7.37288			8		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	106	- 0.44	2	108	0.08	2	130	- 0.07	2	141	0.03
150	2	77	0.16	2	79	0.00	2	95	0.00	2	103	0.16
300	1	155	0.16	1	159	0.00	1	191	0.00	1	207	0.16
600	1	77	0.16	1	79	0.00	1	95	0.00	1	103	0.16
1200	0	155	0.16	0	159	0.00	0	191	0.00	0	207	0.16
2400	0	77	0.16	0	79	0.00	0	95	0.00	0	103	0.16
4800	0	38	0.16	0	39	0.00	0	47	0.00	0	51	0.16
9600	0	19	- 2.34	0	19	0.00	0	23	0.00	0	25	0.16
19200	0	9	- 2.34	0	9	0.00	0	11	0.00	0	12	0.16
31250	0	5	0.00	0	5	2.40	0	6	5.33	0	7	0.00
38400	0	4	- 2.34	0	4	0.00	0	5	0.00	0	6	- 6.99

ビットレート (bit/s)	P (MHz)											
	9.8304			10			12			12.288		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	174	- 0.26	2	177	- 0.25	2	212	0.03	2	217	0.08
150	2	127	0.00	2	129	0.16	2	155	0.16	2	159	0.00
300	1	255	0.00	2	64	0.16	2	77	0.16	2	79	0.00
600	1	127	0.00	1	129	0.16	1	155	0.16	1	159	0.00
1200	0	255	0.00	1	64	0.16	1	77	0.16	1	79	0.00
2400	0	127	0.00	0	129	0.16	0	155	0.16	0	159	0.00
4800	0	63	0.00	0	64	0.16	0	77	0.16	0	79	0.00
9600	0	31	0.00	0	32	- 1.36	0	38	0.16	0	39	0.00
19200	0	15	0.00	0	15	1.73	0	19	- 2.34	0	19	0.00
31250	0	9	- 1.70	0	9	0.00	0	11	0.00	0	11	2.40
38400	0	7	0.00	0	7	1.73	0	9	- 2.34	0	9	0.00

14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

表 14.3 ビットレートに対する SCBRR の設定例〔調歩同期式モード〕(3)

ビットレート (bit/s)	P (MHz)								
	14.7456			16			30		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	3	64	0.70	3	70	0.03	3	132	0.13
150	2	191	0.00	2	207	0.16	3	97	- 0.35
300	2	95	0.00	2	103	0.16	2	194	0.16
600	1	191	0.00	1	207	0.16	2	97	- 0.35
1200	1	95	0.00	1	103	0.16	1	194	0.16
2400	0	191	0.00	0	207	0.16	1	97	- 0.35
4800	0	95	0.00	0	103	0.16	0	194	0.16
9600	0	47	0.00	0	51	0.16	0	97	- 0.35
19200	0	23	0.00	0	25	0.16	0	48	- 0.35
31250	0	14	- 1.70	0	15	0.00	0	29	0.00
38400	0	11	0.00	0	12	0.16	0	23	1.73

表 14.4 ビットレートに対する SCBBR の設定例〔クロック同期式モード〕

ビットレート (bit/s)	P (MHz)							
	4	8	16	32				
	n	N	n	N	n	N	n	N
110								
250	2	249	3	124	3	249		
500	2	124	2	249	3	124	3	249
1k	1	249	2	124	2	249	3	124
2.5k	1	99	1	199	2	99	2	199
5k	0	199	1	99	1	199	2	99
10k	0	99	0	199	1	99	1	199
25k	0	39	0	79	0	159	1	79
50k	0	19	0	39	0	79	0	159
100k	0	9	0	19	0	39	0	79
250k	0	3	0	7	0	15	0	31
500k	0	1	0	3	0	7	0	15
1M	0	0*	0	1	0	3	0	7
2M			0	0*	0	1	0	3

【注】 誤差は、なるべく 1%以内になるように設定してください。

《記号説明》

空欄：設定できません。

-：設定可能ですが誤差がです。

\*：連続送信 / 受信はできません。

表 14.5 にポーレートジェネレータを使用する場合の調歩同期式モードの各周波数における最大ビットレートを示します。また、表 14.6 と表 14.7 に外部クロック入力時の最大ビットレートを示します。

表 14.5 ポーレートジェネレータを使用する場合の各周波数における最大ビットレート (調歩同期式モード)

P (MHz)	最大ビットレート (bit/s)	設定値	
		n	N
2	62500	0	0
2.097152	65536	0	0
2.4576	76800	0	0
3	93750	0	0
3.6864	115200	0	0
4	125000	0	0
4.9152	153600	0	0
8	250000	0	0
9.8304	307200	0	0
12	375000	0	0
14.7456	460800	0	0
16	500000	0	0
19.66080	614400	0	0
20	625000	0	0
24	750000	0	0
24.57600	768000	0	0
28	896875	0	0
30	937500	0	0

表 14.6 外部クロック入力時の最大ビットレート (調歩同期式モード)

P (MHz)	外部入力クロック (MHz)	最大ビットレート (bit/s)
2	0.5000	31250
2.097152	0.5243	32768
2.4576	0.6144	38400
3	0.7500	46875
3.6864	0.9216	57600
4	1.0000	62500
4.9152	1.2288	76800
8	2.0000	125000
9.8304	2.4576	153600
12	3.0000	187500
14.7456	3.6864	230400
16	4.0000	250000
30	7.5000	468750

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

表 14.7 外部クロック入力時の最大ビットレート (クロック同期式モード)

P (MHz)	外部入力クロック (MHz)	最大ビットレート (bit/s)
8	1.3333	1333333.3
16	2.6667	2666666.7
30	5.0	5000000.0

### 14.2.10 FIFO コントロールレジスタ (SCFCR)

ビット:	7	6	5	4	3	2	1	0
	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

FIFO コントロールレジスタ (SCFCR) は送信、受信各 FIFO レジスタのデータ数リセットおよびトリガデータ数の設定を行うレジスタです。またループバックテストの許可ビットを含んでいます。

SCFCR は、常に読み出し / 書き込みが可能です。

SCFCR は、リセット、モジュールスタンバイまたはスタンバイモード時に H'00 に初期化されます。

- ・ ビット7、ビット6: レシーブ FIFO データ数トリガ (RTRG1、0)

シリアルステータス1レジスタ (SC1SSR) のレシーブデータフル (RDF) フラグをセットする受信データ数を設定するビットです。

レシーブ FIFO レジスタ (SCFRDR) 内に格納された受信データ数が下表に示したトリガ設定数以上になったとき RDF フラグをセットします。

ビット7	ビット6	受信トリガ数
RTRG1	RTRG0	
0	0	1*
	1	4
1	0	8
	1	14

【注】 \* 初期値

- ・ ビット5、ビット4: トランスミット FIFO データ数トリガ (TTRG1、0)

シリアルステータス1レジスタ (SC1SSR) のトランスミット FIFO データレジスタエンプティ (TDFE) フラグをセットする残りの送信データ数を設定するビットです。

送信動作によりトランスミット FIFO データレジスタ (SCFTDR) 内の送信データ数が、下表に示すトリガ設定数以下になったとき TDFE フラグをセットします。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

ビット5	ビット4	送信トリガ数
TTRG1	TTRG0	
0	0	8 (8) *
	1	4 (12)
1	0	2 (14)
	1	1 (15)

【注】 \* 初期値。( ) 内の値はフラグ発生時の SCFTDR の空き数を示します。

- ・ ビット3：モデムコントロールイネーブル (MCE)  
 モデムコントロール信号 CTS、RTS を許可 / 禁止します。

ビット3	説 明
MCE	
0	モデム信号を禁止 * (初期値)
1	モデム信号を許可

【注】 \* 入力値に関係なく CTS はアクティブ0 に固定され、RTS も0 に固定されます。

- ・ ビット2：トランスミット FIFO データレジスタリセット (TFRST)  
 トランスミット FIFO データレジスタ内の送信データを無効とし、空の状態にリセットします。

ビット2	説 明
TFRST	
0	リセット動作を禁止 (初期値)
1	リセット動作を許可

【注】 リセット、モジュールスタンバイまたはスタンバイモード時にはリセット動作が行われます。

- ・ ビット1：レシーブ FIFO データレジスタリセット (RFRST)  
 レシーブ FIFO データレジスタ内の受信データを無効とし、空の状態にリセットします。

ビット1	説 明
RFRST	
0	リセット動作を禁止 (初期値)
1	リセット動作を許可

【注】 リセット、モジュールスタンバイまたはスタンバイモード時にはリセット動作が行われます。

- ・ ビット0：ループバックテスト (LOOP)  
 送信出力端子 (TxD)、受信入力端子 (RxID) を内部で接続し、ループバックテストを可能にします。

ビット0	説 明
LOOP	
0	ループバックテストを禁止 (初期値)
1	ループバックテストを許可

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

---

### 14.2.11 FIFO データ数レジスタ (SCFDR)

トランスミット FIFO データレジスタ (SCFTDR) およびレシーブ FIFO データレジスタ (SCFRDR) 内に格納されているデータ数を示す 16 ビット長のレジスタです。

上位 8 ビットで SCFTDR 内の送信データ数を、下位 8 ビットで SCFRDR 内の受信データ数を示します。

SCFDR は常に CPU から読み出しができます。

SCFDR は、リセット、モジュールスタンバイ、またはスタンバイモード時に H'0000 に初期化されます。また、SCFCR の TFRST、RFRST ビットを 1 にセットし、SCFTDR、SCFRDR を空の状態にリセットすることで、下各バイトが H'00 に初期化されます。

上位8ビット:	15	14	13	12	11	10	9	8
	—	—	—	T4	T3	T2	T1	T0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

- ・ ビット 15 ~ 13 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 12 ~ 8 : トランスミット FIFO データ数 4 ~ 0 (T4 ~ T0)

上位バイトは SCFTDR 内に格納されている未送信のデータ数を示します。

H'00 は送信データがないことを、H'10 は SCFTDR にいっぱい送信データが格納されていることを示します。

また、上記の初期化条件のほか、全データを送信することで H'00 にクリアされます。

下位8ビット:	7	6	5	4	3	2	1	0
	—	—	—	R4	R3	R2	R1	R0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

- ・ ビット 7 ~ 5 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ・ ビット 4 ~ 0 : レシーブ FIFO データ数 4 ~ 0 (R4 ~ R0)

下位バイトは SCFRDR 内に格納されている受信データ数を示します。

H'00 は受信データがないことを、H'10 は SCFRDR にいっぱい受信データが格納されていることを示します。

また、上記の初期化条件のほか、全受信データを SCFRDR から読み出すことで H'00 にクリアされます。

## 14.2.12 FIFO エラーレジスタ (SCFER)

レシブ FIFO データレジスタ (SCFRDR) 内に格納されている受信データの中で、パリティエラー、またはフレーミングエラーのあったデータの位置を示します。

SCFER は常に読み出しができます。

上位8ビット :	15	14	13	12	11	10	9	8
	ED15	ED14	ED13	ED12	ED11	ED10	ED9	ED8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
下位8ビット :	7	6	5	4	3	2	1	0
	ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R

・ ビット 15 ~ 0 : エラーデータフラグ 15 ~ 0 (ED15 ~ ED0)

レシブ FIFO データレジスタで、エラーのあったデータの位置を示します。n 段目のバッファ内のデータがエラーを含んでいるとき、n ビット目が 1 になります。なお、SCFCR の RFRST ビットを 1 にセットしても、本レジスタはクリアされませんので注意してください。

ビット 15 ~ 0 ED15 ~ 0	説 明
0	レシブ FIFO 内の該当する段数のデータにパリティ、またはフレーミングエラーなし (初期値)
1	レシブ FIFO 内の該当する段数のデータにパリティ、またはフレーミングエラーあり

【注】 リセット、モジュールスタンバイまたはスタンバイモード時にはリセット動作が行われます。また、パリティエラー、フレーミングエラーのあったデータを、SCFRDR から読み出すことでクリアされます。

## 14.2.13 IrDA モードレジスタ (SCIMR)

IrDA モードの選択、IrDA の出力パルス幅の選択、および IrDA 受信データの極性の反転を行うことができます。

SCIMR は、常に読み出し / 書き込みが可能です。

SCIMR は、リセット、モジュールスタンバイまたはスタンバイモード時に H'00 に初期化されます。

ビット :	7	6	5	4	3	2	1	0
	IRMOD	PSEL	RIVS	—	—	—	—	—
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R	R	R	R	R

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

- ビット 7 : IrDA モード (IRMOD)

IrDA シリアルコミュニケーションインタフェースとしての動作を選択します。

ビット 7	説 明
IRMOD	
0	SCIF としての動作を選択 (初期値)
1	IrDA としての動作を選択 *

【注】 \* IrDA としての動作を選択した場合、シリアルモードレジスタ (SCSMR) のビット 7 (C/A) は必ず 0 を設定してください。

- ビット 6 : 出力パルス幅セレクト (PSEL)

シリアルモードレジスタ (SCSMR) の ICK3 ~ 0 で設定したビット長の 3/16、選択したボーレートに対するビット長の 3/16 のどちらかを IrDA の出力パルス幅として選択します。シリアルモードレジスタ (SCSMR) のビット 6 ~ 3 (ICK3 ~ 0) と合わせて、設定を示します。

シリアルモードレジスタ (SCSMR)				SCIMR	説 明
ビット 6	ビット 5	ビット 4	ビット 3	ビット 2	
ICK3	ICK2	ICK1	ICK0	PSEL	
ICK3	ICK2	ICK1	ICK0	1	パルス幅 : ICK3 ~ 0 で設定したビット長の 3/16
don't care				0	パルス幅 : SCBRR で設定したビット長の 3/16 (初期値)

【注】 P クロックを  $1/(2N+2)$  (N は ICK3 ~ 0 の設定値で決まります) することによって一定したクロックパルス IRCLK を生成する必要があります。詳細は「14.3.6 (3) パルス幅の選択」を参照してください。

- ビット 5 : IrDA レシーブデータインバース (RIVS)

IrDA 通信において、受信データの極性反転を選択できます。

ビット 5	説 明
RIVS	
0	受信時、受信データの極性を反転する (初期値)
1	受信時、受信データの極性を反転しない

【注】 IrDA 変復調モジュールの特性により選択してください。

- ビット 4 ~ 0 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## 14.3 動作説明

### 14.3.1 概要

SCIF は、キャラクタ単位で同期をとりながら通信する調歩同期式モードと、クロックパルスにより同期をとりながら通信するクロック同期式モードの 2 方式で、シリアル通信ができます。

また、IrDA ブロックを備えており、赤外線送受信ユニットを接続することによって IrDA1.0 に準拠した赤外線通信を実行することができます。

送受信各々に 16 段の FIFO バッファを内蔵しており、CPU のオーバーヘッドを減らし、高速連続通信が可能です。

調歩同期式モードと、クロック同期式モード、IrDA モードの選択および送信フォーマットの選択は、シリアルモードレジスタ (SCSMR) と IrDA モードレジスタ (SCIMR) で行います。これを表 14.8 に示します。また、SCIF のクロックソースは、SCSMR の  $C/\bar{A}$  ビット、SCIMR の IRMOD ビットおよびシリアルコントロールレジスタ (SCSCR) の CKE1、CKE0 ビットの組み合わせで決まります。これを表 14.9 に示します。

- 調歩同期式モード
  - データ長 : 7 ビット / 8 ビットから選択可能
  - パリティの付加、マルチプロセッサビットの付加、および 1 ビット / 2 ビットのストップビットの付加を選択可能 (これらの組み合わせにより送信 / 受信フォーマット、およびキャラクタ長を決定)
  - 受信時にフレーミングエラー、パリティエラー、オーバランエラー、レシーブ FIFO データフル、レシーブデータレディ、およびブレークの検出が可能
  - 送信時にトランスミット FIFO データエンプティの検出が可能
  - SCIF のクロックソース : 内部クロック / 外部クロックから選択可能
    - 内部クロックを選択した場合 : SCIF はポーレートジェネレータでつくられたビットレートの 16 倍 (または 8 倍、4 倍) の周波数のクロックで動作し、この動作のクロックを出力することが可能
    - 外部クロックを選択した場合 : ビットレートの 16 倍 (または 8 倍、4 倍) の周波数のクロックを入力することが必要
    - (内蔵ポーレートジェネレータを使用しない)
- クロック同期式モード
  - 送信 / 受信フォーマット : 8 ビットデータ固定
  - 受信時にオーバランエラーの検出可能
  - SCIF のクロックソース : 内部クロック / 外部クロックから選択可能
    - 内部クロックを選択した場合 : SCIF はポーレートジェネレータのクロックで動作し、同期クロックを外部へ出力
    - 外部クロックを選択した場合 : 内部ポーレートジェネレータを使用せず、入力された同期クロックで動作

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

- IrDA モード
  - IrDA1.0 に準拠
  - データ長 : 8 ビット
  - ストップビット長 : 1 ビット
  - 送信中、受信部に影響を与えないようにする保護機能
  - クロックソース : 内部クロック

表 14.8 SCSMR、SCIMR の設定値とシリアル送信 / 受信フォーマット

SCIMR	SCSMR の設定値					モード	SCIF の送信 / 受信フォーマット						
	ビット7	ビット6	ビット2	ビット5	ビット3		データ長	MP ビット	パリティ ビット	ストップ ビット長			
IRMOD	C/A	CHR	MP	PE	STOP								
0	0	0	0	0	0	調歩同期式 モード	8ビット データ	なし	なし	1ビット			
					1					2ビット			
					0					1ビット			
				1	2ビット								
				1	0					0	1ビット		
										1	2ビット		
		1	0	1	*	0	調歩同期式 モード (マルチ プロセッサ フォーマット)	8ビット データ	あり	なし	1ビット		
						1					2ビット		
						*					0	1ビット	
					1	1					*	1	2ビット
												0	1ビット
												1	2ビット
0	1	*	*	*	*	クロック同期 式モード	8ビット データ	なし	なし	なし			
1	0	ICK3	ICK2	ICK1	ICK0	IrDA モード	8ビット データ	なし	なし	1ビット			
1	1	*	*	*	*	設定禁止							

【注】 表中の \* は Don't care であることを示します。

表 14.9 SCSMR、SCSCR の設定と SCIF のクロックソースの選択

SCSMR	SCSCR の設定		モード	SCIF の送信 / 受信クロック	
	ビット7	ビット0		クロック ソース	SCK 端子の機能
C/A	CKE1	CKE0			
0	0	0	調歩同期式モード	内部	SCIF は、SCK 端子を使用しません ビットレートの 16 倍 (8 倍、4 倍) の周波数のクロックを出力
		1			
	1	0		外部	ビットレートの 16 倍 (8 倍、4 倍) の周波数のクロックを入力
		1			
1	0	0	クロック同期式モード	内部	同期クロックを出力
		1			
	1	0		外部	同期クロックを入力
		1			

### 14.3.2 調歩同期式モード時の動作

調歩同期式モードは、通信開始を意味するスタートビットと通信終了を意味するストップビットとをデータに付加したキャラクタを送信 / 受信し、1 キャラクタ単位で同期をとりながらシリアル通信を行うモードです。

SCIF 内部では、送信部と受信部は独立していますので、全二重通信を行うことができます。また、送信部と受信部がともに 16 段の FIFO バッファ構造になっていて、送信 / 受信中にデータの読み出し / 書き込みができるので、連続送信 / 受信が可能です。

調歩同期式シリアル通信の一般的なフォーマットを図 14.3 に示します。

調歩同期式シリアル通信では、通信回線は通常、マーク状態（ハイレベル）に保たれています。SCIF は通信回線を監視し、スペース（ローレベル）になったところをスタートビットとみなしてシリアル通信を開始します。

シリアル通信の 1 キャラクタは、スタートビット（ローレベル）から始まり、データ（LSB ファースト：最下位ビットから、MSB ファースト：最上位ビットから選択可能）、パリティビットまたはマルチプロセッサビット（ハイ / ローレベル）、最後にストップビット（ハイレベル）の順で構成されています。

調歩同期式モードでは、SCIF は受信時にスタートビットの立ち下がりエッジで同期化を行います。また SCIF は、データを 1 ビット期間の 16 倍（8 倍、4 倍）の周波数のクロックの 8 番目（4 番目、2 番目）でサンプリングしますので、各ビットの中央で通信データが取り込まれます。



図 14.3 調歩同期式通信のデータフォーマット  
(8 ビットデータ / パリティあり / 2 ストップビット / LSB ファーストの例)

#### (1) 送信 / 受信フォーマット

調歩同期式モードで設定できる送信 / 受信フォーマットを、表 14.10 に示します。

送信 / 受信フォーマットは 12 種類あり、シリアルモードレジスタ (SCSMR) の設定により選択できます。

14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

表 14.10 シリアル送信/受信フォーマット (調歩同期式モード)

SCSMRの設定				シリアル送信/受信フォーマットとフレーム長											
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	S [ 8ビットデータ ] STOP											
			1	S [ 8ビットデータ ] STOP										STOP	STOP
	1		0	S [ 8ビットデータ ]										P	STOP
			1	S [ 8ビットデータ ]										P	STOP
1	0		0	S [ 7ビットデータ ]									STOP		
			1	S [ 7ビットデータ ]									STOP	STOP	
	1		0	S [ 7ビットデータ ]									P	STOP	
			1	S [ 7ビットデータ ]									P	STOP	STOP
0	*	1	0	S [ 8ビットデータ ]										MPB	STOP
	*		1	S [ 8ビットデータ ]										MPB	STOP
1	*		0	S [ 7ビットデータ ]									MPB	STOP	
	*		1	S [ 7ビットデータ ]									MPB	STOP	STOP

【注】 表中の \* はDon't careであることを示します。

- S : スタートビット
- STOP : ストップビット
- P : パリティビット
- MPB : マルチプロセスサビット

## (2) クロック

SCIFの送受信クロックは、SCSMRの $C/\bar{A}$ ビットとシリアルコントロールレジスタ(SCSCR)のCKE1、CKE0ビットの設定により、内蔵ポーレートジェネレータの生成した内部クロック、またはSCK端子から入力された外部クロックの2種類から選択できます。SCIFのクロックソースの選択については表14.9を参照してください。

外部クロックをSCK端子に入力する場合には、使用するビットレートの16倍(8倍、4倍)の周波数のクロックを入力してください。

内部クロックで動作させるとき、SCK端子からクロックを出力することができます。このとき出力されるクロックの周波数はビットレートの16倍(8倍、4倍)です。

## (3) データの送信 / 受信動作

## (a) SCIFの初期化(調歩同期式)

データの送信 / 受信前には、まずSCSCRのTEビット、およびREビットを0にクリアした後、以下の順でSCIFを初期化してください。

動作モードの変更、通信フォーマットの変更などの場合には必ず、TEビットおよびREビットを0にクリアしてから次の手順で変更を行ってください。TEビットを0にクリアすると、トランスミットシフトレジスタ(SCTSR)が初期化されます。TE、REビットを0にクリアしても、シリアルステータス1レジスタ(SC1SSR)、トランスミットFIFOデータレジスタ(SCFTDR)および、レシーブFIFOデータレジスタ(SCFRDR)の内容は保持されますので注意してください。TEビットの0クリアは、送信データを全て送信しSC1SSRのTENDフラグがセットされた後に行ってください。送信中でも0クリア可能ですが、送信中のデータは0クリア後ハイインピーダンス状態になります。また再度TEセットして送信開始する前にSCFCRのTFRSTビットをいったん1にしてSCFTDRをリセットしてください。

外部クロックを使用している場合には、動作が不確実になりますので初期化を含めた動作中にクロックを止めないでください。

図14.4にSCIFの初期化フローチャートの例を示します。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

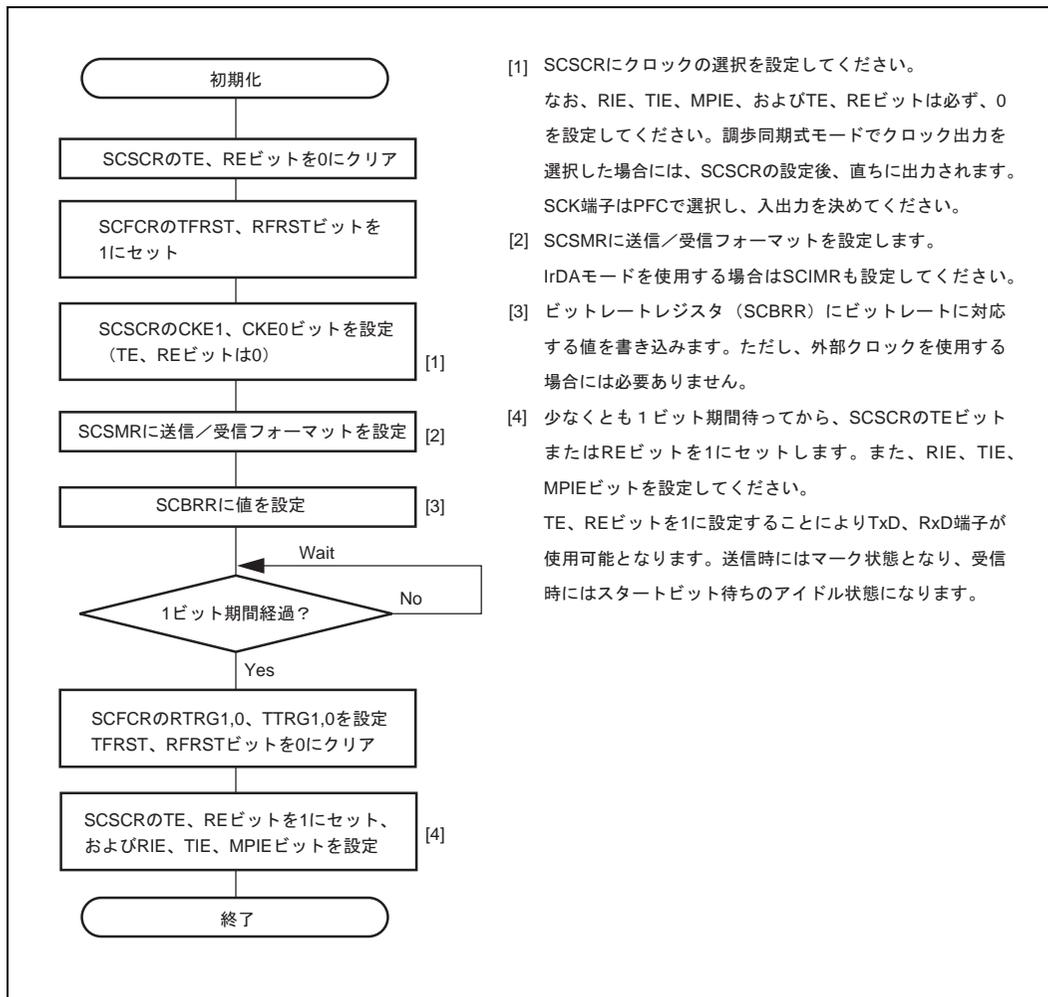
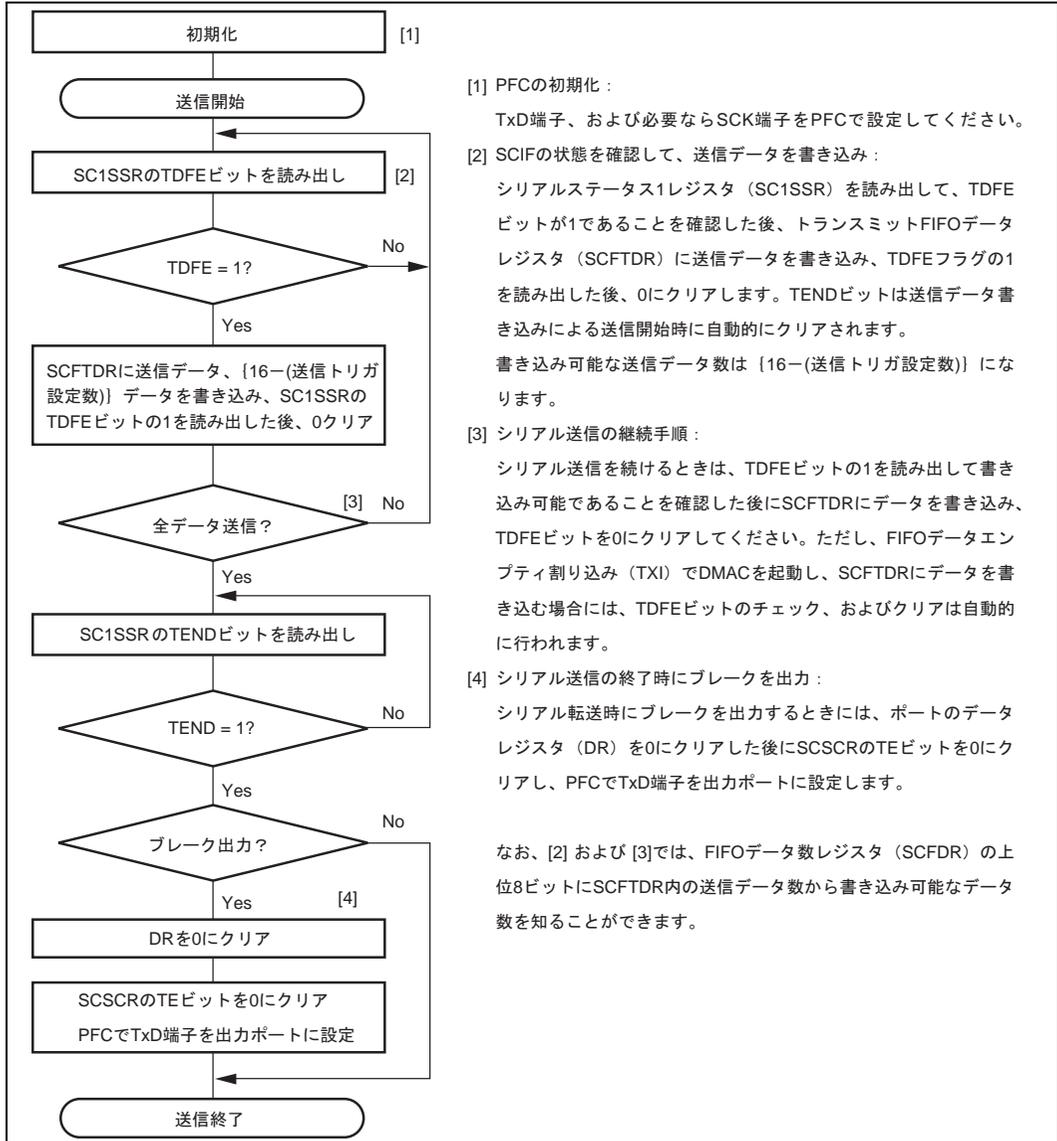


図 14.4 SCIF の初期化フローチャートの例

(b) シリアルデータ送信 (調歩同期式)

図 14.5 にシリアル送信のフローチャートの例を示します。

シリアルデータ送信は、SCIF を送信動作可能状態に設定した後、以下の手順にしたがって行ってください。



[1] PFCの初期化：  
TxD端子、および必要ならSCK端子をPFCで設定してください。

[2] SCIFの状態を確認して、送信データを書き込み：  
シリアルステータス1レジスタ (SC1SSR) を読み出して、TDFEビットが1であることを確認した後、トランスミットFIFOデータレジスタ (SCFTDR) に送信データを書き込み、TDFEフラグの1を読み出した後、0にクリアします。TENDビットは送信データ書き込みによる送信開始時に自動的にクリアされます。  
書き込み可能な送信データ数は {16-(送信トリガ設定数)} になります。

[3] シリアル送信の継続手順：  
シリアル送信を続けるときは、TDFEビットの1を読み出して書き込み可能であることを確認した後SCFTDRにデータを書き込み、TDFEビットを0にクリアしてください。ただし、FIFOデータエンプロティ割り込み (TXI) でDMACを起動し、SCFTDRにデータを書き込む場合には、TDFEビットのチェック、およびクリアは自動的に行われます。

[4] シリアル送信の終了時にブレークを出力：  
シリアル転送時にブレークを出力するときには、ポートのデータレジスタ (DR) を0にクリアした後SCSCRのTEビットを0にクリアし、PFCでTxD端子を出力ポートに設定します。

なお、[2] および [3]では、FIFOデータ数レジスタ (SCFDR) の上位8ビットにSCFTDR内の送信データ数から書き込み可能なデータ数を知ることができます。

図 14.5 シリアル送信のフローチャートの例

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

---

SCIF はシリアル送信時に以下のように動作します。

- [ 1 ] SCIF は、トランスミット FIFO データレジスタ (SCFTDR) にデータが書き込まれると、SCFTDR からトランスミットシフトレジスタ (SCTSR) にデータを転送し、送信を開始します。SCFTDR にはシリアルステータス 1 レジスタ (SC1SSR) の TDFE フラグが 1 であることを確認して送信データを書き込んでください。書き込み可能なデータ数は少なくとも  $\{16 - (\text{送信トリガ設定数})\}$  です。
- [ 2 ] SCFTDR から SCTSR へデータが転送され、送信を開始すると、SCFTDR に送信データが無くなるまで連続して送信動作を続けます。途中、SCFTDR 内のデータ数が FIFO コントロールレジスタ (SCFCR) で設定した送信トリガ数以下になったとき、TDFE フラグをセットします。  
このとき、シリアルコントロールレジスタ (SCSCR) の TE ビットが 1 にセットされていると送信 FIFO データエンプティ割り込み (TXI) 要求を発生します。

シリアル送信データは、以下の順に TxD 端子から送り出されます。

- (a) スタートビット：1 ビットの 0 が出力されます。
  - (b) 送信データ：8 ビット、または 7 ビットのデータが SC2SSR の TLM ビットの設定値により LSB あるいは MSB から順に出力されます。
  - (c) パリティビットまたはマルチプロセッサビット：1 ビットのパリティビット (偶数パリティ、または奇数パリティ) または 1 ビットのマルチプロセッサビットが出力されます。  
なお、パリティビット、またはマルチプロセッサビットを出力しないフォーマットも選択できます。
  - (d) ストップビット：1 ビットまたは 2 ビットの 1 (ストップビット) が出力されます。
  - (e) マーク状態：次の送信を開始するスタートビットを送り出すまで 1 を出力し続けます。
- [ 3 ] SCIF は、ストップビットを送出するタイミングで SCFTDR の送信データをチェックします。データがあると SCFTDR から SCTSR にデータを転送し、ストップビットを送り出した後、次フレームのシリアル送信を開始します。  
送信データが無いとシリアルステータスレジスタ (SC1SSR) の TEND フラグに 1 をセットし、ストップビットを送り出した後、1 を出力するマーク状態になります。

調歩同期式モードでの送信時の動作例を図 14.6 に示します。

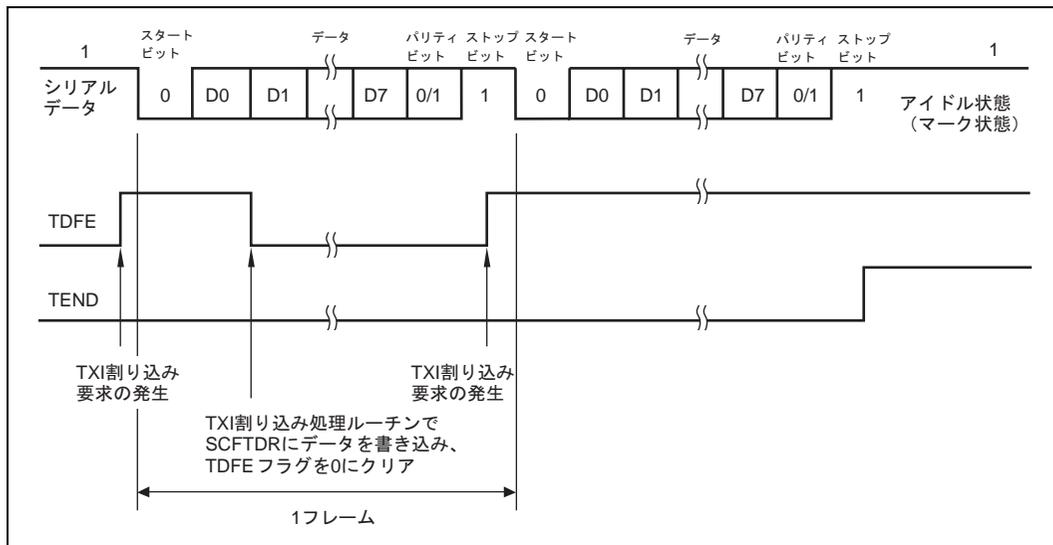


図 14.6 調歩同期式モードでの送信時の動作例  
(8 ビットデータ / パリティあり / 1 ストップビット / LSB ファーストの例)

[4] モデムコントロールを有効にした場合、 $\overline{\text{CTS}}$  入力値に応じて送信を停止、再開することができます。 $\overline{\text{CTS}}$  が 1 にセットされると送信中である場合 1 フレーム送信後マーク状態になります。 $\overline{\text{CTS}}$  が 0 になると、次の送信データがスタートビットを先頭に出力されます。

図 14.7 にモデムコントロールを使用した動作例を示します。

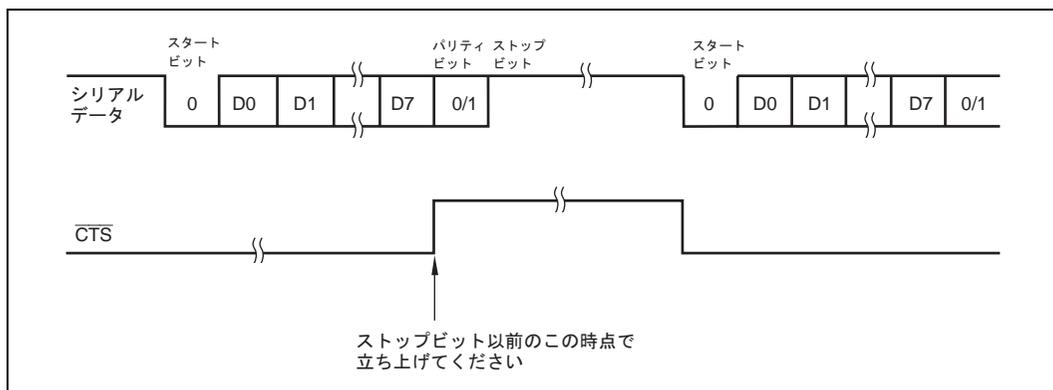


図 14.7 モデムコントロールを使用した動作例 ( $\overline{\text{CTS}}$ )

(c) シリアルデータ受信 (調歩同期式)

図 14.8 にシリアル受信フローチャートの例を示します。

シリアルデータ受信は、SCIF を受信動作可能状態に設定した後、以下の手順にしたがって行ってください。

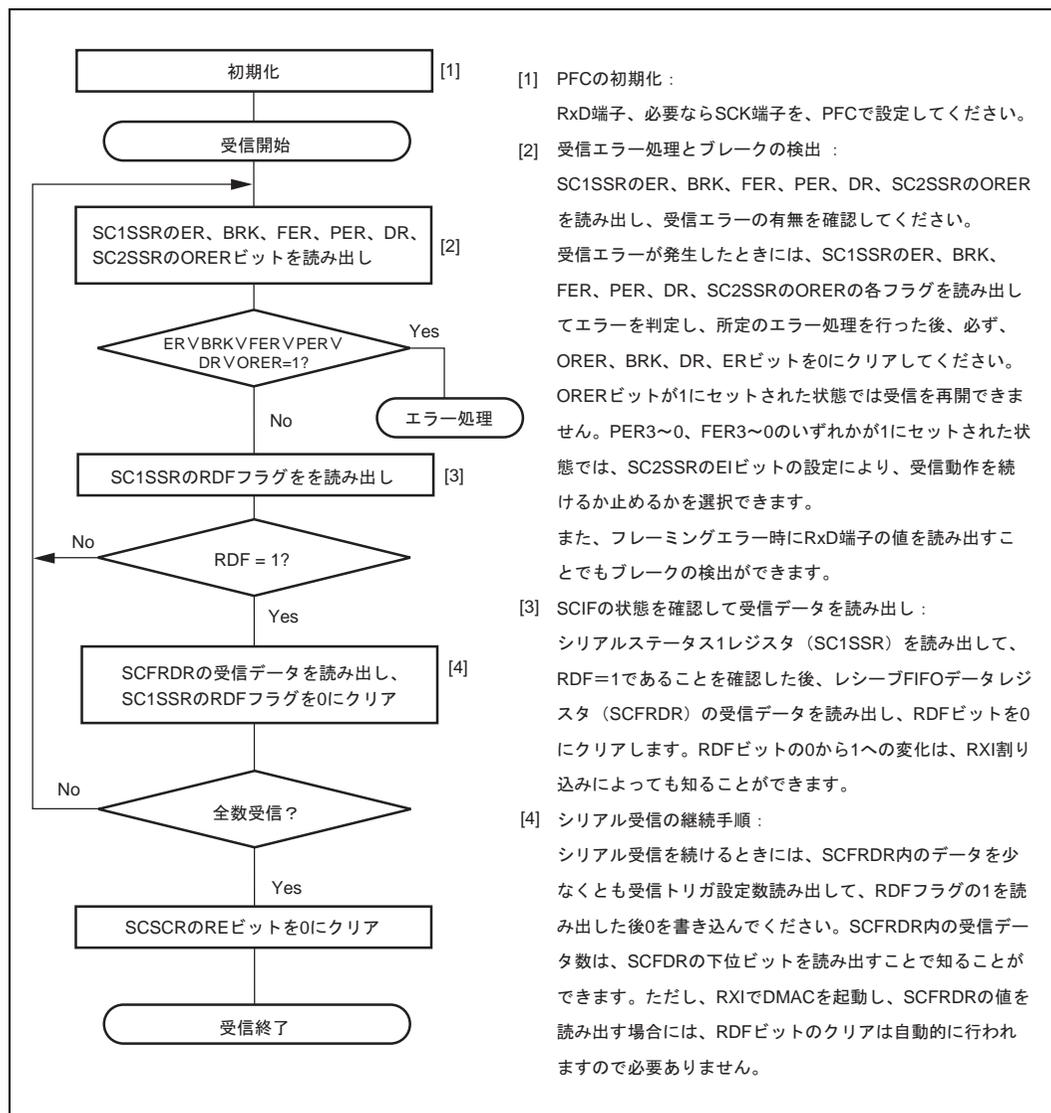


図 14.8 シリアル受信のフローチャートの例 (1)

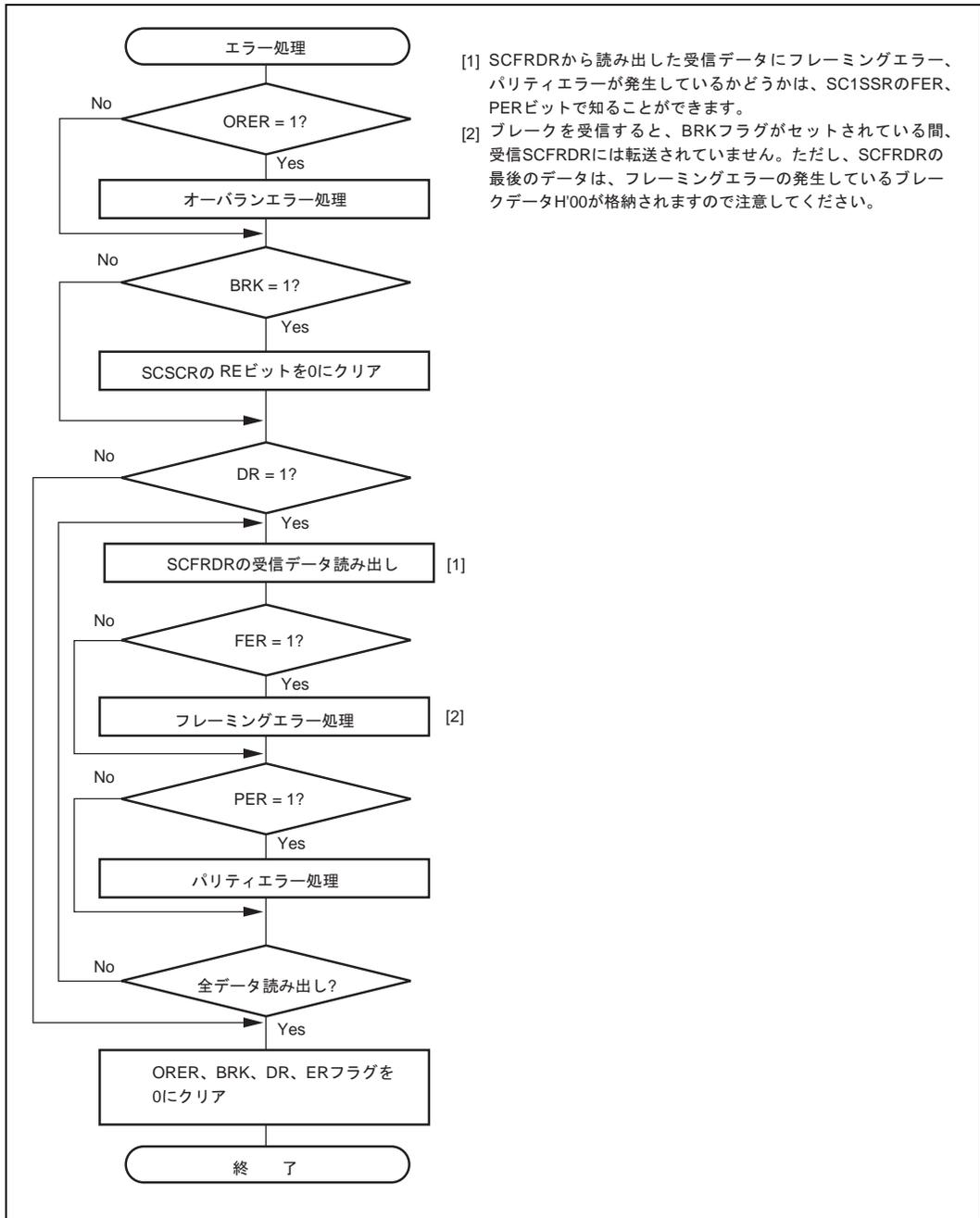


図 14.8 シリアル受信のフローチャートの例 (2)

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

SCIF は受信時に以下のように動作します。

- [1] SCIF は通信回路を監視し、スタートビットの 0 を検出すると内部を同期化し、受信を開始します。
  - [2] SC2SSR の RLM ビットの設定に従い、受信したデータを SCRSR の LSB から MSB または MSB から LSB の順に格納します。
  - [3] パリティビット、およびストップビットを受信します。  
受信後、SCIF は以下のチェックを行います。
    - (a) パリティチェック：受信データの 1 の数をチェックし、これがシリアルモードレジスタ (SCSMR) の 0/E ビットで設定した偶数 / 奇数パリティになっているかをチェックします。
    - (b) ストップビットチェック：ストップビットが 1 であるかをチェックします。  
ただし、2 ストップビットの場合、1 ビット目のストップビットのみをチェックします。
    - (c) 受信データをレシフシフトレジスタ (SCRSR) から SCFRDR に転送できる状態であるかをチェックします。
    - (d) ブレークチェック：BRK フラグが 0 であり、ブレーク状態でないことをチェックします。
- 以上のチェックがすべてパスしたとき、SCFRDR に受信データが格納されます。  
エラーチェックで受信エラーを発生すると表 14.11 のように動作します。
- 【注】 オーバランエラーが発生した状態では、以後の受信動作ができません。フレーミングエラー、パリティエラーが発生した時は、SC2SSR の EI ビットの設定により、受信動作を続けるか止めるかを選択できます。  
また、受信時に RDF フラグが 1 にセットされませんので、必ずエラーフラグを 0 にクリアしてください。
- [4] RDF または DR フラグが 1 になったとき、SCSCR の RIE ビットが 1 にセットされていると受信 FIFO データフル割り込み (RXI) 要求を発生します。  
また、ORER、PER、FER フラグのどれか 1 になったとき、SCSCR の RIE ビットが 1 にセットされていると受信エラー割り込み (ERI) 要求を発生します。  
さらに、BRK フラグが 1 になったとき、SCSCR の RIE ビットが 1 にセットされているとブレーク受信割り込み (BRI) 要求を発生します。

表 14.11 受信エラーと発生条件

受信エラー名	略称	発生条件	データ転送
オーバランエラー	ORER	SCFRDR 内の受信データ数が 16 の状態で次のシリアル受信を完了したとき	SCRSR から SCFRDR に受信データは転送されません。
フレーミングエラー	FER	ストップビットが 0 のとき	SCRSR から SCFRDR に受信データが転送されます。
パリティエラー	PER	SCSMR で設定した偶数 / 奇数パリティの設定と受信したデータが異なるとき	SCRSR から SCFRDR に受信データが転送されます。

調歩同期式モード受信時の動作例を図 14.9 に示します。

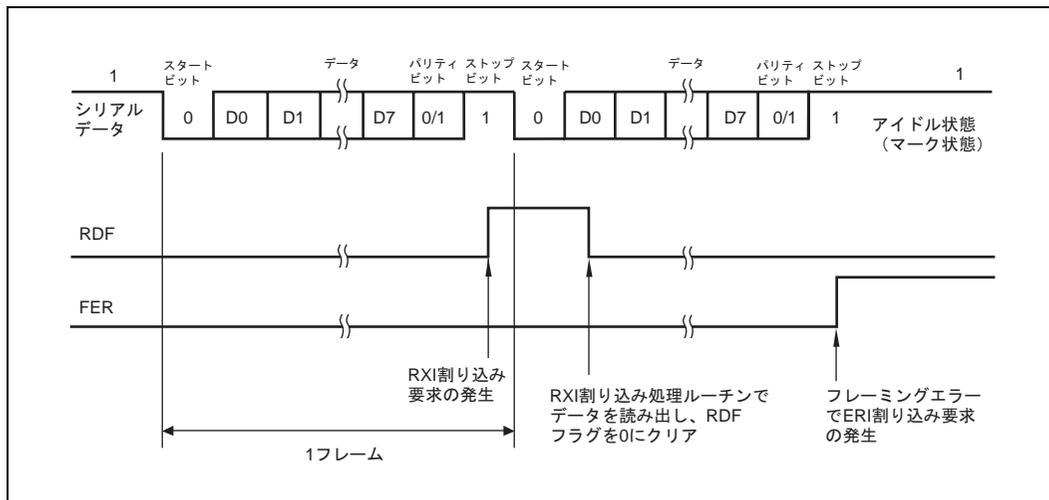


図 14.9 SCIF の受信時の動作例  
(8 ビットデータ / パリティあり / 1 ストップビット / LSB ファーストの例)

[5] モデムコントロールが有効であると、SCFRDR が空のとき  $\overline{\text{RTS}}$  信号が出力されます。 $\overline{\text{RTS}}$  が 0 のときは受信が可能です。 $\overline{\text{RTS}}$  が 1 のときは SCFRDR がいっぱいである受信が不可能であることを示します。

図 14.10 にモデムコントロール使用時の動作例を示します。

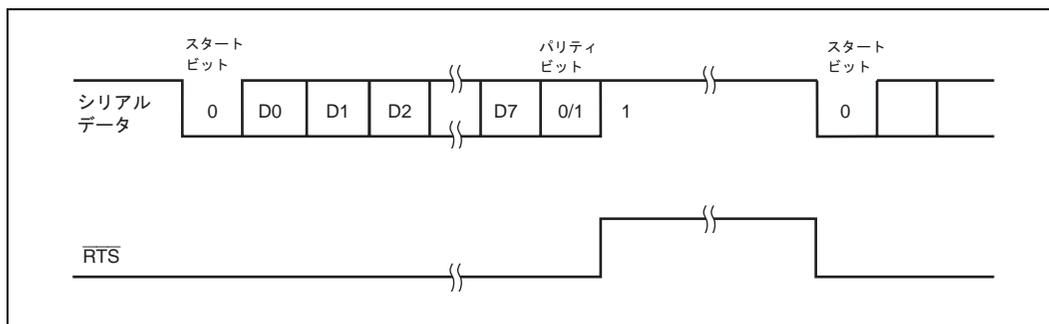


図 14.10 モデムコントロール使用時の動作例

### 14.3.3 マルチプロセッサ通信機能

マルチプロセッサ通信機能とは、調歩同期式モードでマルチプロセッサビットを付加したフォーマット（マルチプロセッサフォーマット）でシリアル通信をする機能です。この機能を使用すると、複数のプロセッサ間でシリアル通信回線を共有したデータの送受信ができます。

マルチプロセッサ通信を行うとき、受信局は各々固有の ID コードでアドレッシングされています。

シリアル通信サイクルは、受信局を指定する ID 送信サイクルとデータ送信サイクルの 2 つから構成されます。この ID 送信サイクルとデータ送信サイクルの区別は、マルチプロセッサビットで行います。

送信局は、まず、シリアル通信を行いたい受信局の ID をマルチプロセッサビット 1 を付加したデータにして送信します。続いて、送信データをマルチプロセッサビット 0 を付加したデータにして送信します。

受信局は、マルチプロセッサビット 1 のデータが送信されるまではデータを読み飛ばします。

マルチプロセッサビット 1 のデータを受信したとき、受信局は自局の ID と比較します。そして、一致した局は続いて送信されるデータを受信します。一方一致しなかった局は、再びマルチプロセッサビット 1 のデータが送信されるまではデータを読み飛ばします。このようにして複数のプロセッサ間のデータ送受信が行われます。

図 14.11 にマルチプロセッサフォーマットを使用したプロセッサ間通信の例を示します。

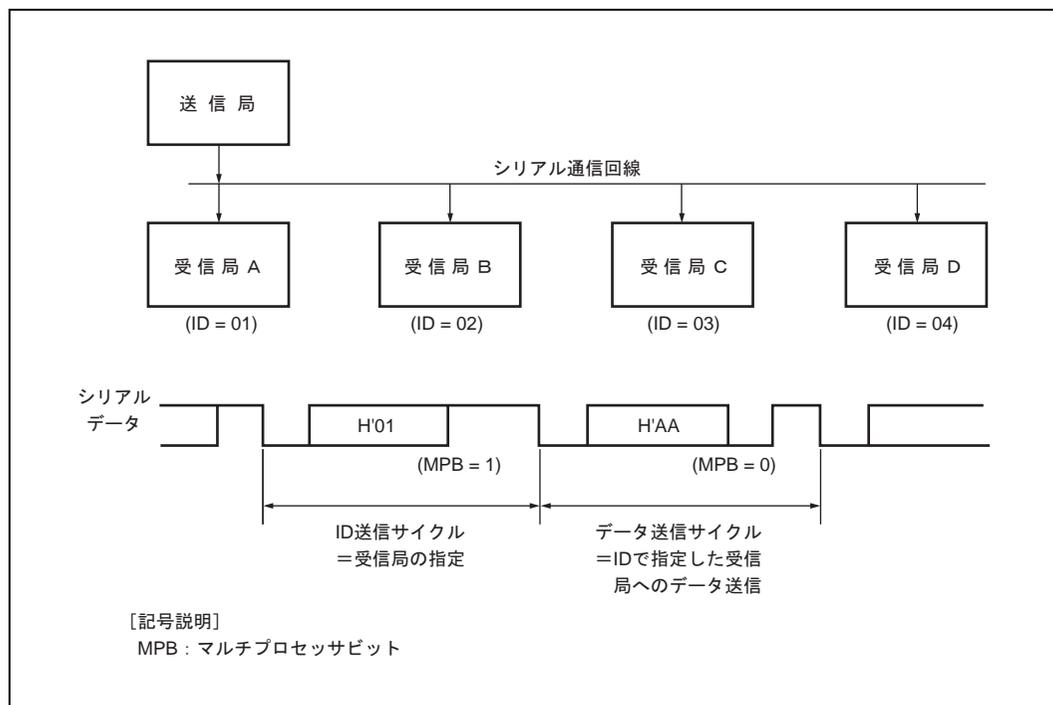


図 14.11 マルチプロセッサフォーマットを使用したプロセッサ間通信の例  
(受信局 A へのデータ H'AA の送信の例)

(1) 送信 / 受信フォーマット

送信 / 受信フォーマットは 4 種類です。

マルチプロセッサフォーマットを指定した場合は、パリティビットの指定は無効です。

詳細は表 14.10 を参照してください。

(2) クロック

調歩同期式モードの項を参照してください。

(3) データの送信 / 受信動作

(a) SCIF 初期化

調歩同期式モードの項を参照してください。

(b) マルチプロセッサシリアルデータ送信

図 14.12 にマルチプロセッサシリアル送信のフローチャートの例を示します。

マルチプロセッサシリアルデータ送信は、SCIF を送信動作可能状態に設定した後、以下の手順にしたがって行ってください。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

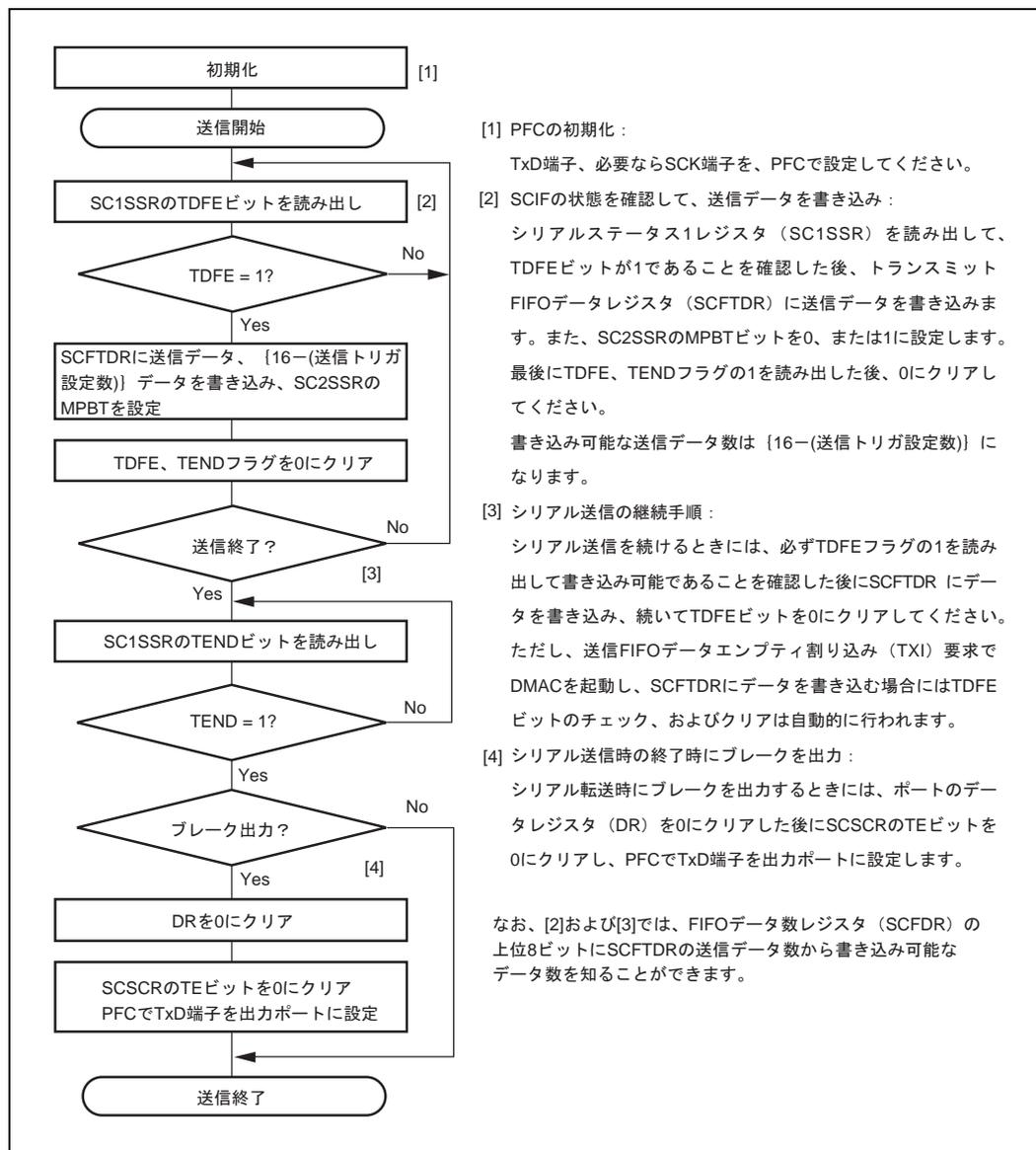


図 14.12 マルチプロセッサシリアル送信のフローチャートの例

SCIF は、シリアル送信時に以下のように動作します。

- (1) SCIF は、SCFTDR にデータが書き込まれると、SCFTDR から SCTSR にデータを転送し、送信を開始します。SCFTDR には SC1SSR の TDFE フラグが 1 であることを確認して送信データを書き込んでください。書き込み可能なデータ数は少なくとも  $\{16 - (\text{送信トリガ設定数})\}$  です。
- (2) SCFTDR から SCTSR へデータが転送され、送信を開始すると、SCFTDR に送信データが無くなるまで連続して送信動作を続けます。途中、SCFTDR 内の送信データ数が SCFCR で設定した送信トリガ設定数以下になったとき、TDFE フラグを 1 にセットします。  
このとき、SCSCR の TIE ビットが 1 にセットされていると送信 FIFO データエンプティ割り込み (TXI) 要求を発生します。

シリアル送信データは、以下の順に TxD 端子から送り出されます。

- (a) スタートビット : 1 ビットの 0 が出力されます。
  - (b) 送信データ : 8 ビット、または 7 ビットのデータが SC2SSR の TLM ビットの設定値により LSB または MSB から順に出力されます。
  - (c) マルチプロセッサビット : 1 ビットのマルチプロセッサビット (MPBT の値) が出力されます。
  - (d) ストップビット : 1 ビット、または 2 ビットの 1 (ストップビット) が出力されます。
  - (e) マーク状態 : 次の送信を開始するスタートビットを送り出すまで 1 を出力し続けます。
- (3) SCIF は、ストップビットを送り出すタイミングで SCFTDR の送信データをチェックします。データがあると SCFTDR から SCTSR にデータを転送し、ストップビットを送り出した後、次のフレームのシリアル送信を開始します。  
データが無いと SC1SSR の TEND フラグを 1 にセットし、ストップビットを送り出した後、1 を出力するマーク状態になります。

図 14.13 にマルチプロセッサフォーマットの SCIF の送信時の動作例を示します。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

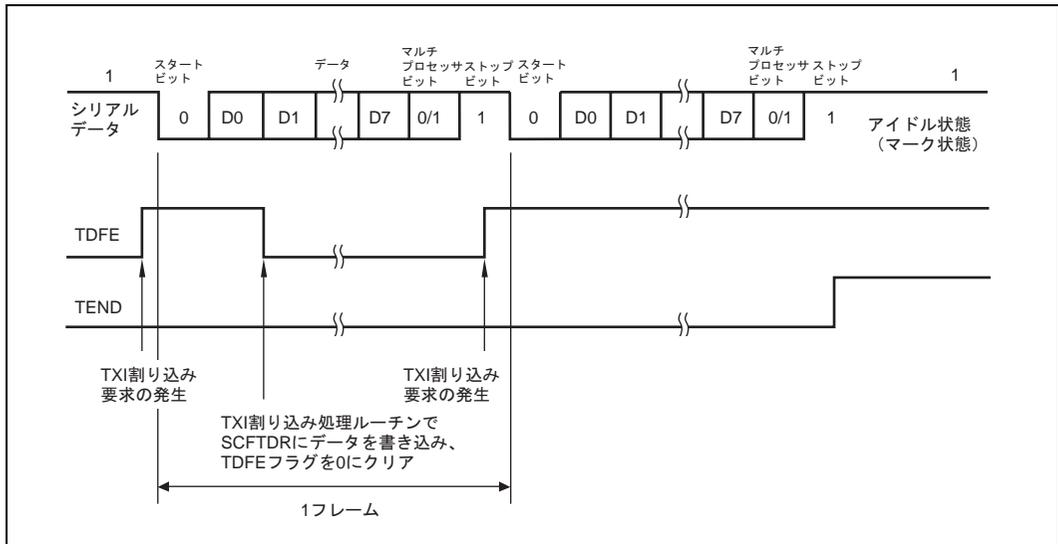


図 14.13 SCIF の送信時の動作例

(8 ビットデータ / マルチプロセッサビットあり / 1 ストップビット / LSB ファーストの例)

### (c) マルチプロセッサシリアルデータ受信

図 14.14 にマルチプロセッサシリアル受信のフローチャートの例を示します。

マルチプロセッサシリアルデータ受信は、SCIF を受信動作可能状態に設定した後、以下の手順にしたがって行ってください。

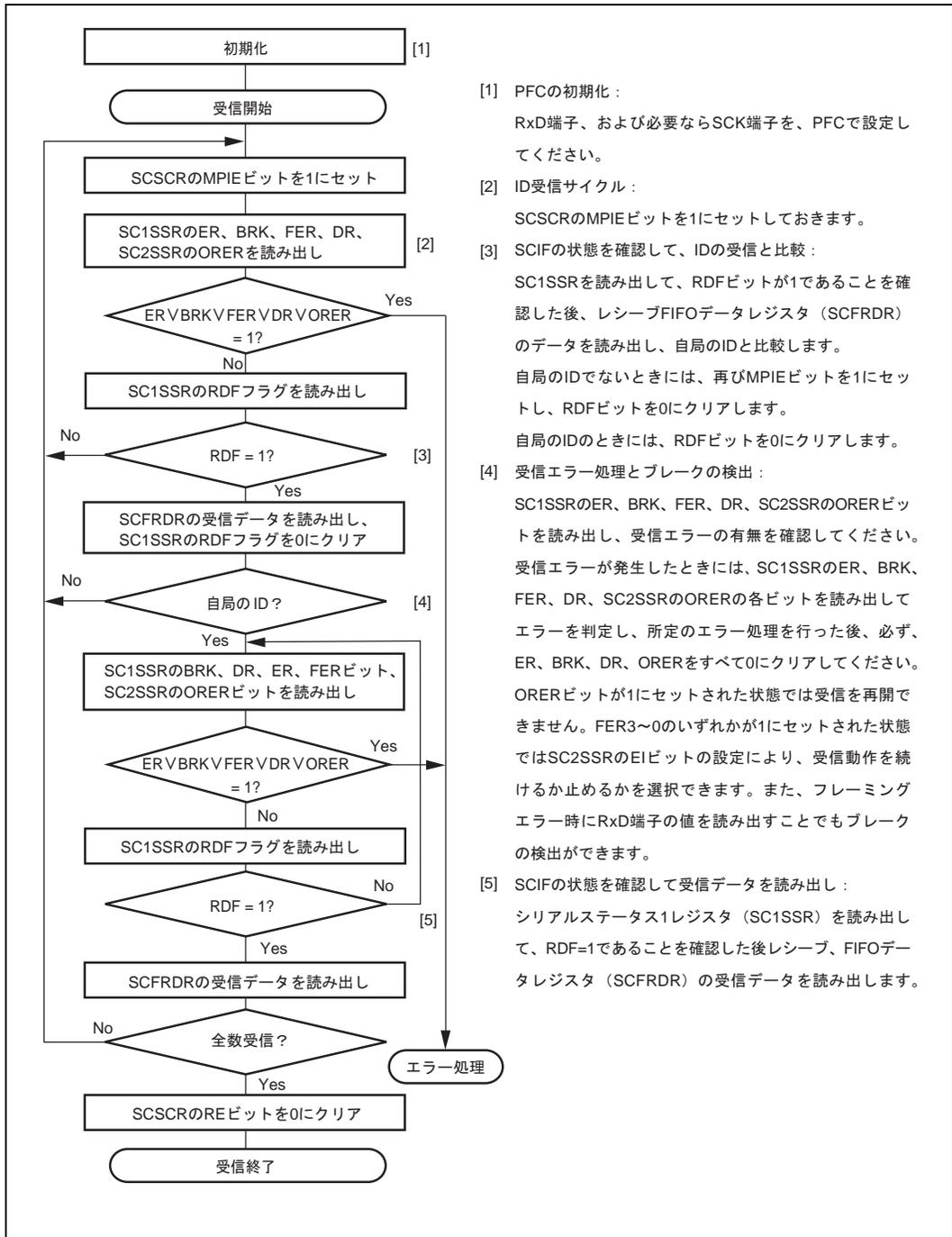


図 14.14 マルチプロセッサシリアル受信のフローチャートの例 (1)

14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

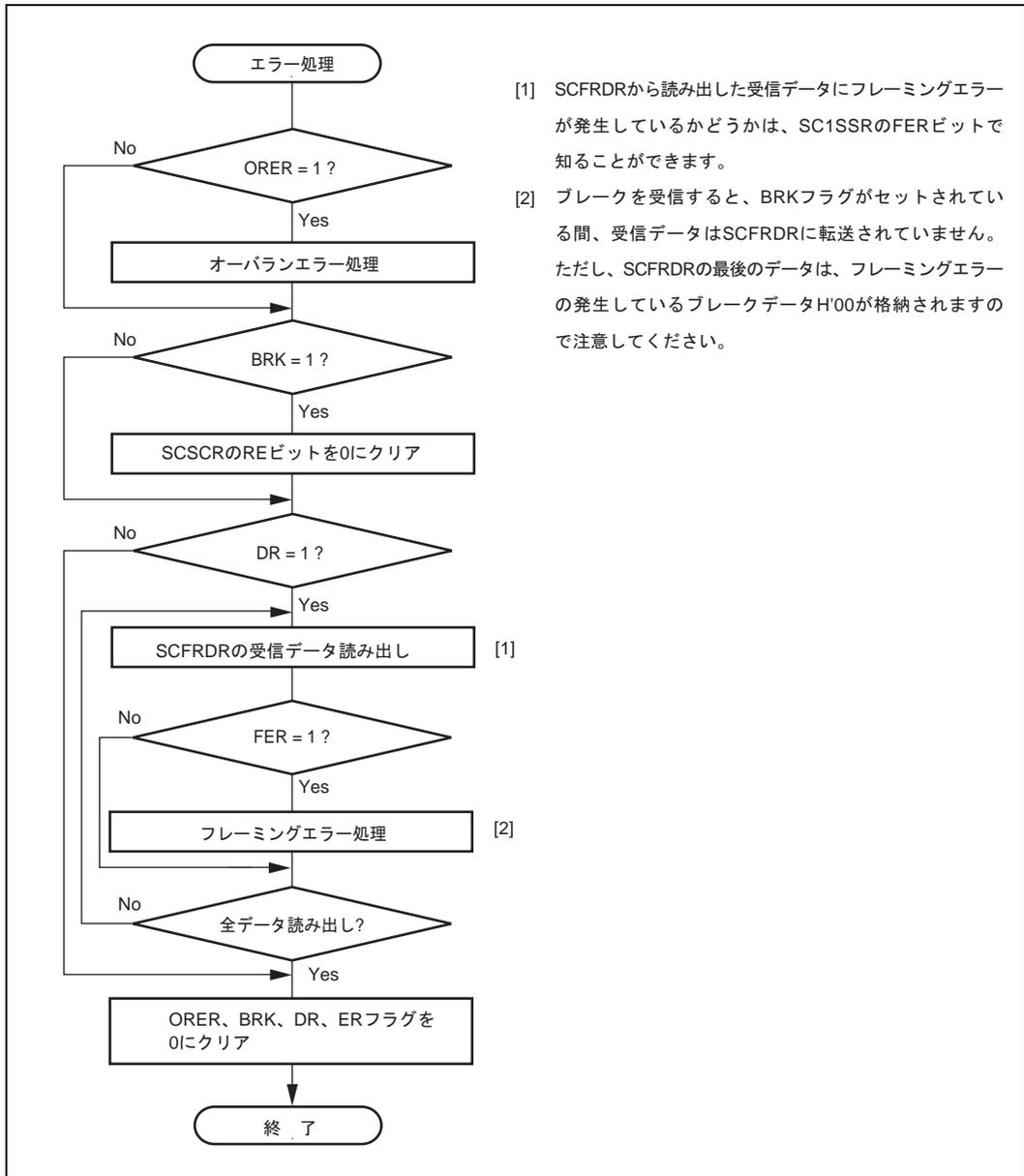


図 14.14 マルチプロセッサシリアル受信のフローチャートの例 (2)

図 14.15 にマルチプロセッサフォーマットの SCIF の受信時の動作例を示します。

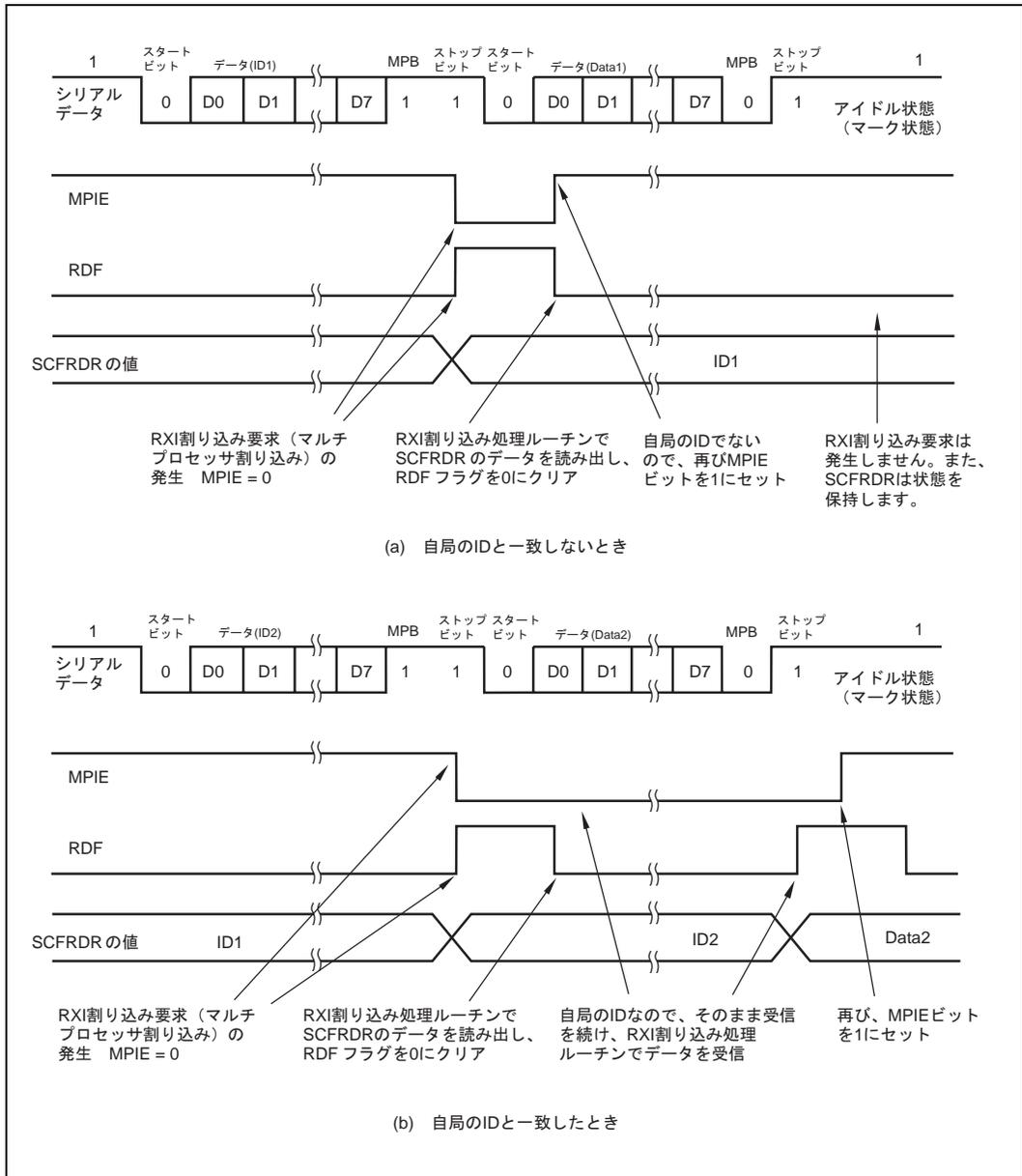


図 14.15 SCIF の受信時の動作例

(8 ビットデータ / マルチプロセッサビットあり / 1 ストップビット / LSB ファーストの例)

### 14.3.4 クロック同期式モード時の動作

クロック同期式モードは、クロックパルスに同期してデータを送信 / 受信するモードで、高速シリアル通信に適しています。

SCIF 内部では、送信部と受信部は独立していますので、クロックを共有することで全二重通信ができます。

また、送信部と受信部が共に 16 段の FIFO バッファ構造になっていますので送信 / 受信中にデータの読み出し / 書き込みができ、連続送信 / 受信が可能です。

クロック同期式シリアル通信の一般的なフォーマットを図 14.16 に示します。

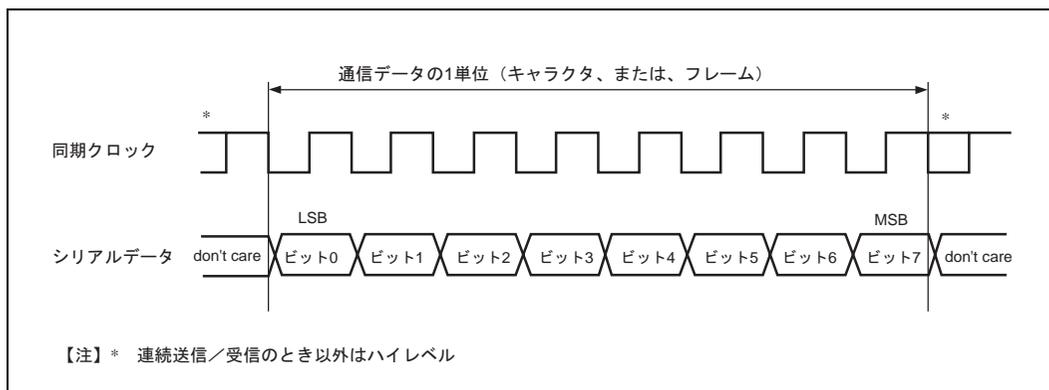


図 14.16 クロック同期式通信のデータフォーマット (LSB ファーストの例)

クロック同期式シリアル通信では、通信回線のデータは同期クロックの立ち下がりから次の立ち上がりまで出力されます。また、同期クロックの立ち上がりでデータの確定が保証されます。

シリアル通信の 1 キャラクタは、シリアルステータス 2 レジスタの TLM ビットにより、データの LSB あるいは MSB から始まり最後に MSB あるいは LSB が出力されます。最終データ出力後の通信回線の状態は最終データの状態を保ちます。

クロック同期式モードでは、SCIF は同期クロックの立ち上がり同期してデータを受信します。

#### (1) 送信 / 受信フォーマット

8 ビットデータ固定です。

パリティビットやマルチプロセッサビットの付加はできません。

#### (2) クロック

SCSMR の  $C/\bar{A}$  ビットと SCSCR の CKE1、CKE0 ビットの設定により内蔵ポーレートジェネレータの生成した内部クロック、または、SCK 端子から入力された外部同期クロックの 2 種類から選択できます。SCIF のクロックソースの選択については表 14.9 を参照してください。

内部クロックで動作させるとき、SCK 端子から同期クロックが出力されます。

同期クロックは 1 キャラクタの送受信で 8 パルス出力され、送信 / 受信を行わないときにはハイレベルに固定されます。ただし、受信のみの動作のときは、SCIF は 2 キャラクタを 1 単位として受信動作を行いますので、

16パルスの同期クロックが出力されます。1キャラクタ端子の受信動作を行いたいときは、クロックソースは外部クロックを選択してください。

(3) データの送信 / 受信動作

(a) SCIF の初期化 (クロック同期式)

データの送信 / 受信前にシリアルコントロールレジスタ (SCSCR) の TE、および RE ビットを 0 にクリアした後、以下の手順で SCIF を初期化してください。

モードの変更、通信フォーマットの変更などの場合には必ず、TE、および RE ビットを 0 にクリアしてから下記手順で変更してください。TE ビットを 0 にクリアすると TDFE フラグは 1 にセットされ、トランスミットシフトレジスタ (SCTSR) が初期化されます。

RE ビットを 0 にクリアしても RDF、PER、FER、ORER の各フラグ、およびレシーブデータレジスタ (SCRDR) の内容は保持されますので注意してください。

図 14.17 に SCIF の初期化フローチャートの例を示します。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

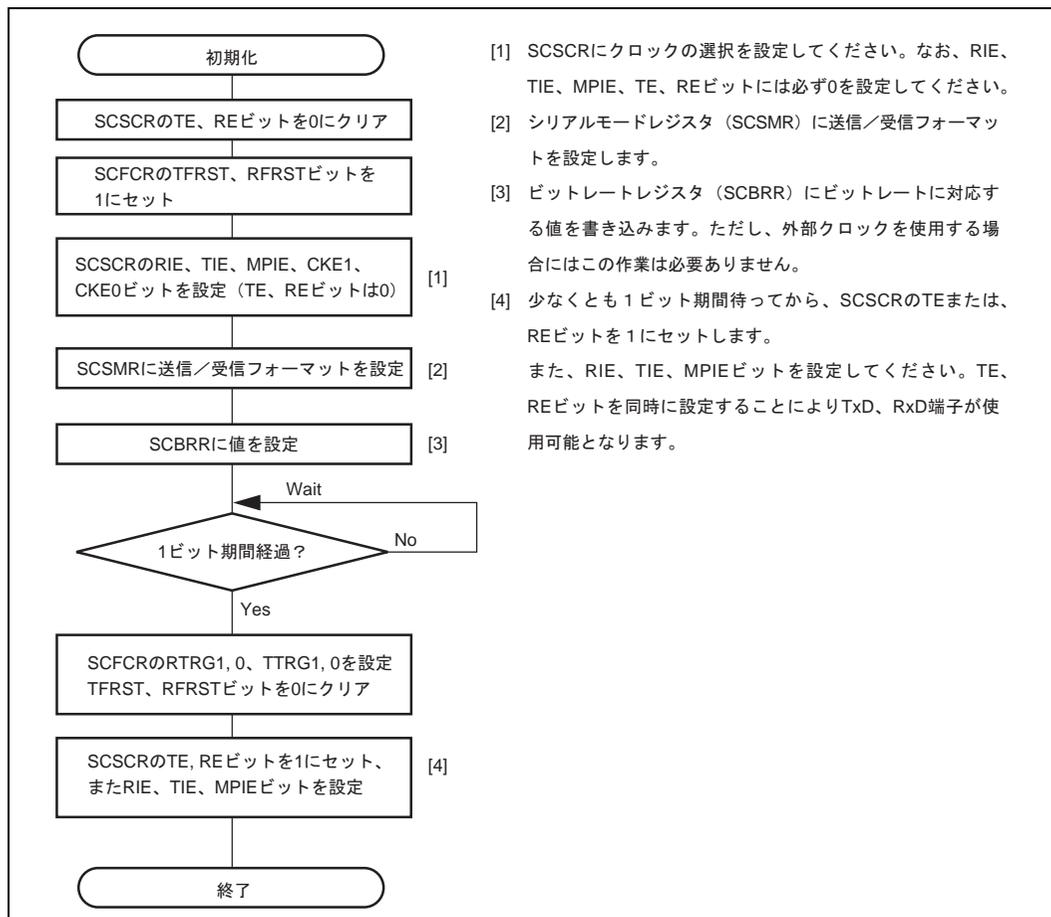


図 14.17 SCIF の初期化フローチャートの例

### (b) シリアルデータ送信 (クロック同期式)

図 14.18 にシリアル送信のフローチャートの例を示します。

シリアルデータ送信は、SCIF を送信動作可能状態に設定した後、以下の手順で行ってください。

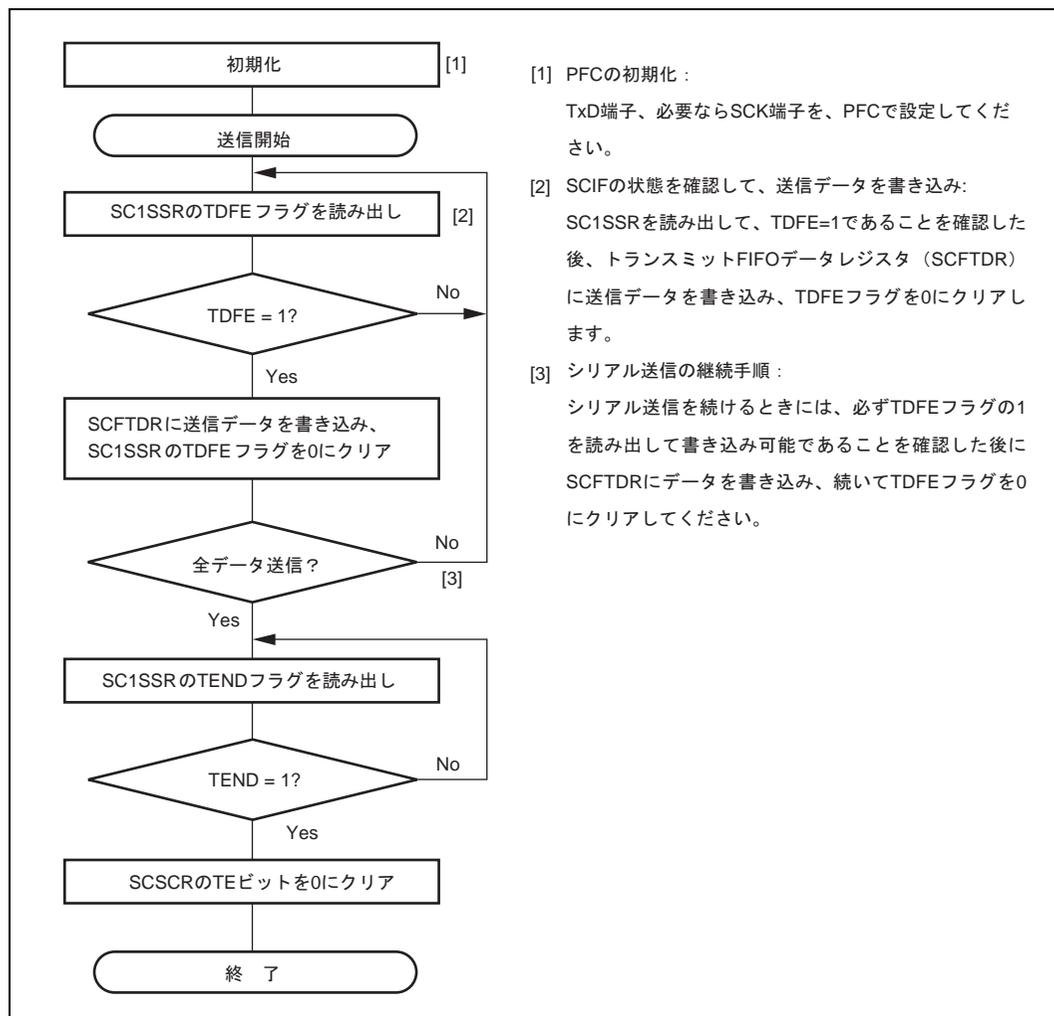


図 14.18 シリアル送信のフローチャートの例

SCIF はシリアル送信時に以下のように動作します。

- (1) SCIF は、トランスミット FIFO データレジスタ (SCFTDR) にデータが書き込まれると、SCFTDR からトランスミットシフトレジスタ (SCTSR) にデータを転送し、送信を開始します。SCFTDR にはシリアルステータス 1 レジスタ (SC1SSR) の TDFE フラグが 1 であることを確認して送信データを書き込んでください。書き込み可能なデータ数は少なくとも { 16 - (送信トリガ設定数) } です。
- (2) SCFTDR から SCTSR へデータが転送され、送信を開始すると、SCFTDR に送信データが無くなるまで連続して送信動作を続けます。途中、SCFTDR 内の送信データ数が FIFO コントロールレジスタ (SCFCR) で設定したトリガ数以下になったとき、TDFE フラグをセットします。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

このとき、シリアルコントロールレジスタ (SCSCR) の TIE ビットが 1 にセットされていると送信 FIFO データエンプティ割り込み (TXI) 要求が発生します。

クロック出力モードに設定したときには、SCIF は 1 データに対し同期クロックを 8 パルス出力します。

外部クロックに設定したときには、入力クロックに同期してデータを出力します。

シリアル送信データは、シリアルステータス 2 レジスタ (SC2SSR) の TLM ビットの設定により LSB (ビット 0) あるいは MSB (ビット 7) から順に TxD 端子から送り出されます。

(3) SCIF は、最終ビットを送り出すタイミングで SCFTDR の送信データをチェックします。

送信データがあると SCFTDR から SCTSR にデータを転送し、次フレームのシリアル送信を開始します。

送信データがないとシリアルステータスレジスタ (SC1SSR) の TEND フラグを 1 にセットし、最終ビットを送り出した後、トランスミットデータ端子 (TxD 端子) は状態を保持します。

(4) シリアル送信終了後は、SCK 端子はハイレベル固定になります。

図 14.19 に SCIF の送信時の動作例を示します。

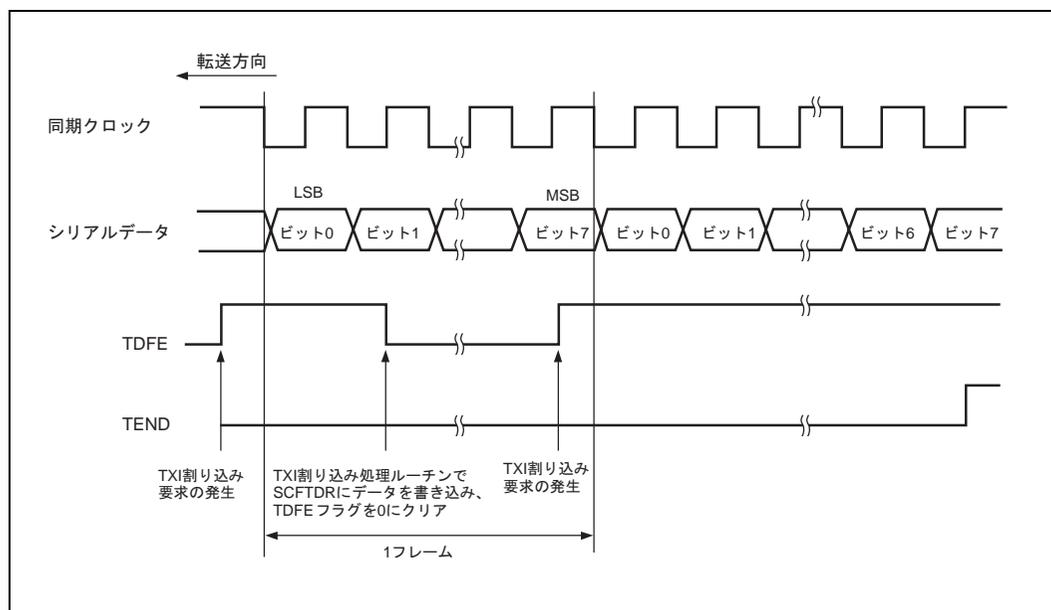


図 14.19 SCIF の送信時の動作例 (LSB ファーストの例)

(c) シリアルデータ受信 (クロック同期式)

図 14.20 にシリアル受信のフローチャートの例を示します。

シリアルデータ受信は、SCIF を受信動作可能状態に設定した後、以下の手順にしたがって行ってください。

SCIF の初期化による SCFRDR、SCFTDR のリセットを行わずに、動作モードを調歩同期モードからクロック同期モードに切り替える際には、必ず、ORER、PER3 ~ 0、FER3 ~ 0 の各フラグが 0 にクリアされていることを確認してください。

FER3 ~ 0、PER3 ~ 0 フラグが 1 にセットされていると RDF フラグがセットされません。また、送信 / 受信動作が行えません。

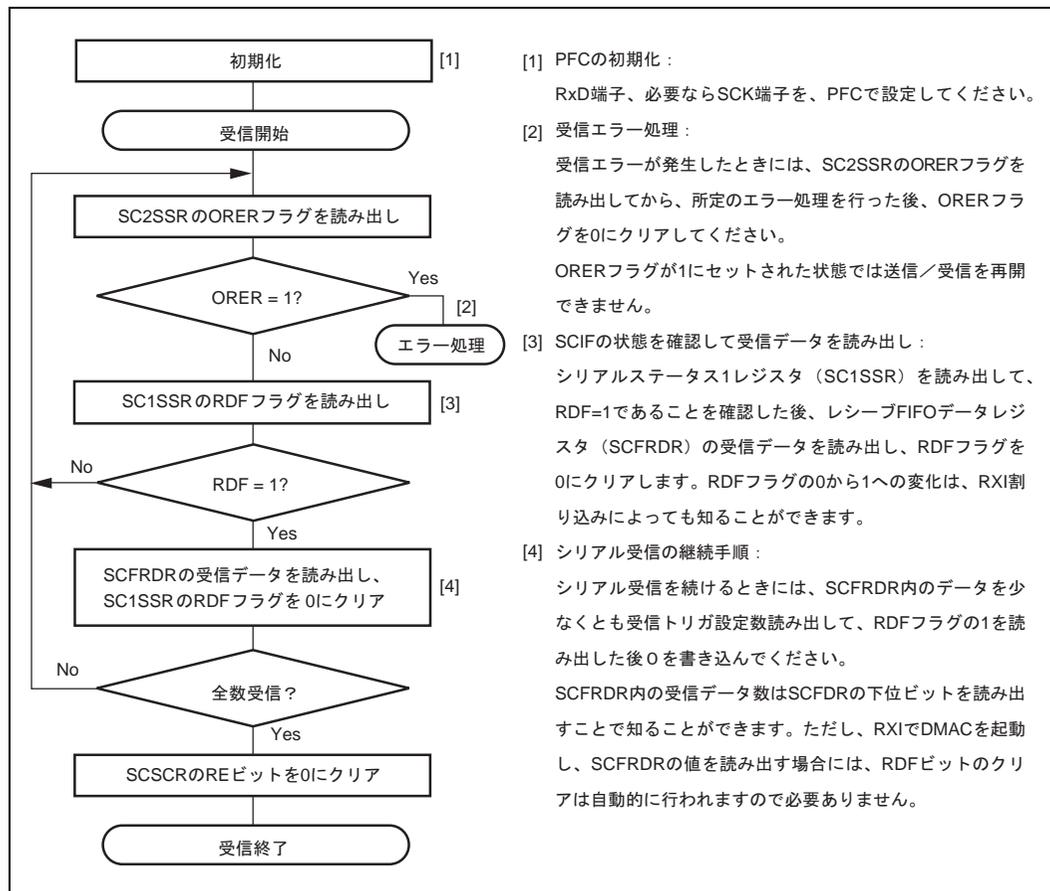


図 14.20 シリアルデータ受信フローチャートの例 (1)

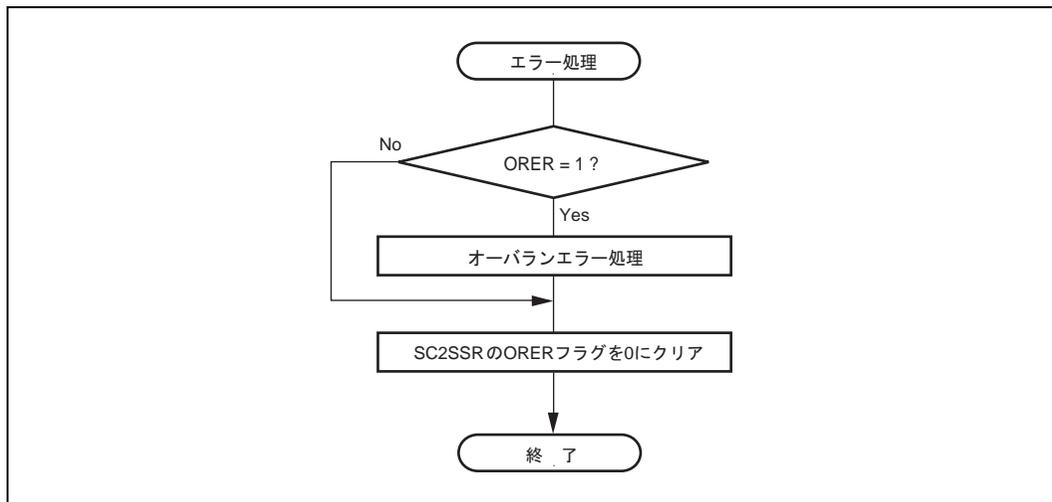


図 14.20 シリアルデータ受信フローチャートの例 (2)

SCIF は受信時に以下のように動作します。

- (1) SCIF は同期クロックの入力または出力に同期して内部を初期化します。
- (2) SC2SSR の RLM ビットの設定に従い、受信したデータをレシーブシフトレジスタ (SCRSR) の LSB から MSB または MSB から LSB の順に格納します。  
受信後、SCIF は受信データを SCRSR からレシーブ FIFO データレジスタ (SCFRDR) に転送できる状態であるかをチェックし、このチェックがパスしたとき SCFRDR に受信データが格納されます。  
エラーチェックで受信エラーが発生すると表 14.11 のように動作し、この状態では以後の送信、受信動作ができません。  
また、受信時に RDF フラグが 1 にセットされませんので、必ずフラグを 0 にクリアしてください。
- (3) RDF フラグが 1 になったとき、シリアルコントロールレジスタ (SCSCR) の RIE ビットが 1 にセットされていると受信 FIFO データフル割り込み (RXI) 要求が発生します。  
また、ORER フラグが 1 になったとき、SCSCR の RIE ビットが 1 にセットされていると受信エラー割り込み (ERI) 要求が発生します。

図 14.21 に SCIF の受信時の動作例を示します。

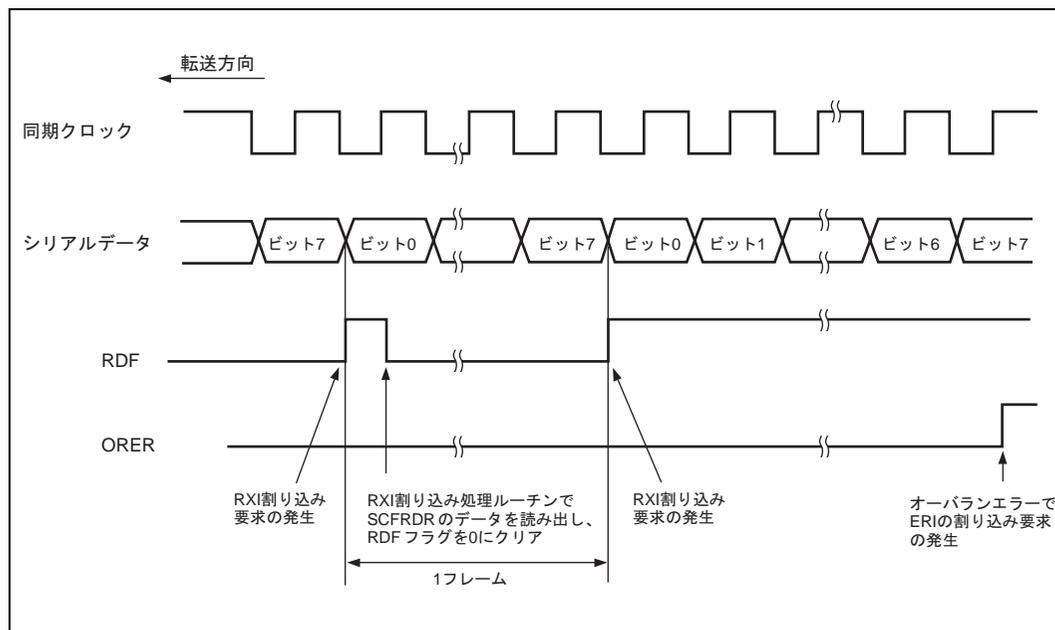


図 14.21 SCIF の受信時の動作例 (LSB ファーストの例)

## (d) シリアルデータ送受信同時動作 (クロック同期式)

図 14.22 にシリアル送受信同時動作のフローチャートの例を示します。

シリアルデータ送受信同時動作は、SCIF を送受信動作可能状態に設定した後、以下の手順に従って行ってください。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

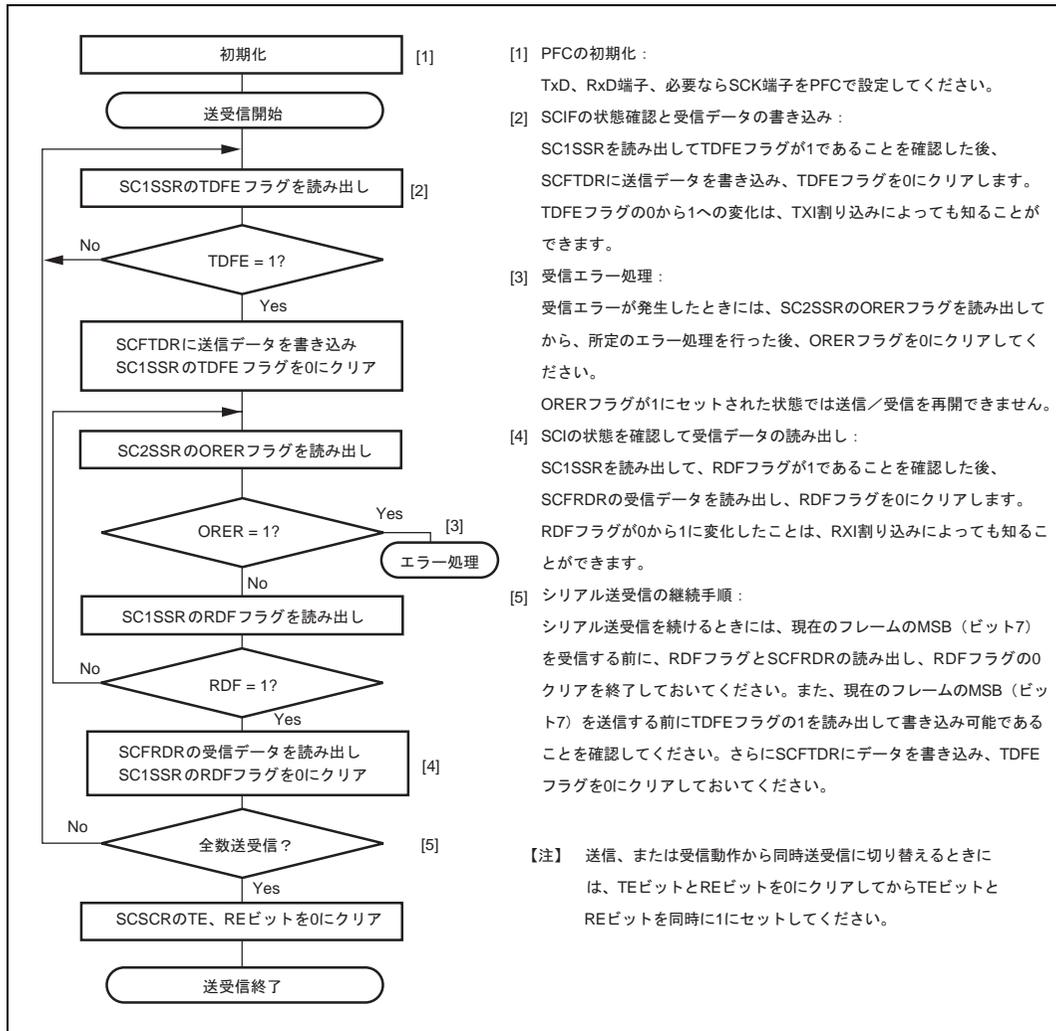


図 14.22 シリアルデータ送受信フローチャートの例

## 14.3.5 送信 / 受信 FIFO バッファの使用方法

SCIF は、送信用と受信用に独立した 16 段の FIFO バッファを内蔵しています。

バッファの構成を図 14.23 に示します。

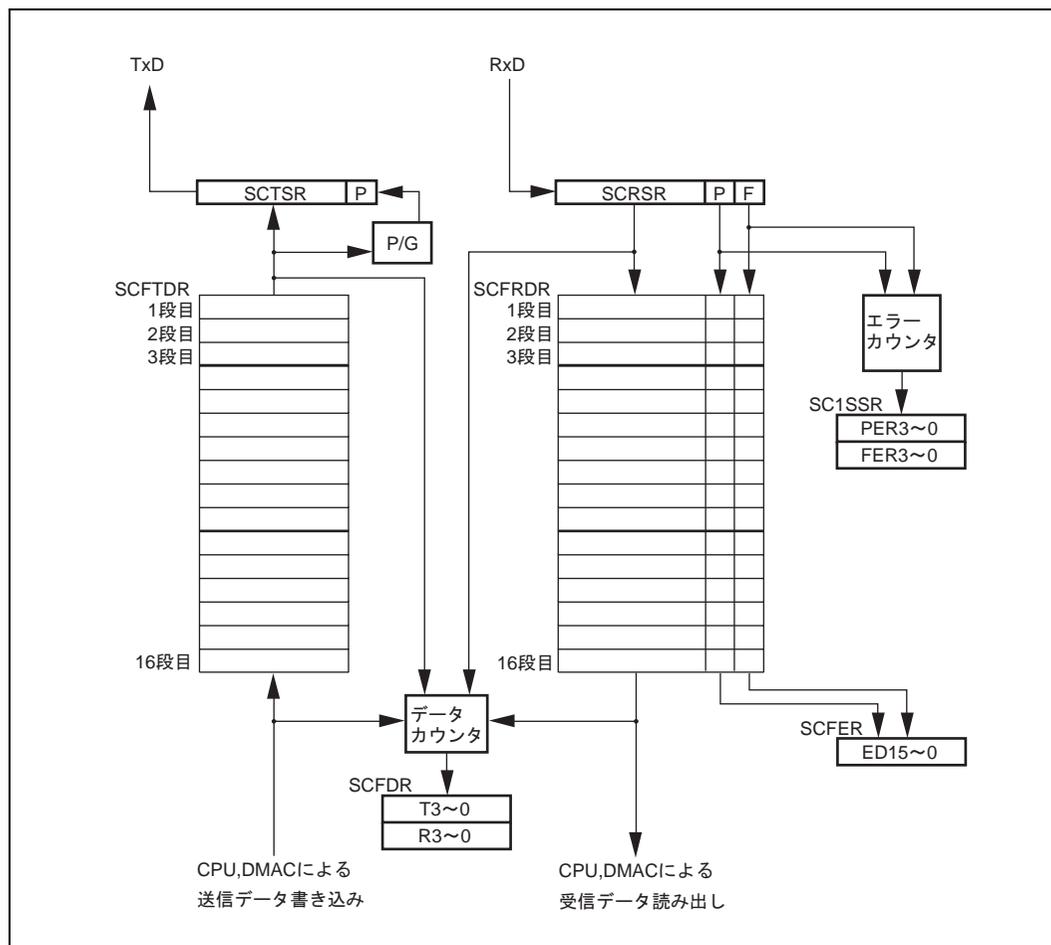


図 14.23 送信 / 受信 FIFO の構成

## (1) シリアルデータ送信動作時

送信時には、CPU あるいは DMAC により送信 FIFO に送信データを書き込み、シリアルコントロールレジスタ (SCSCR) の TE ビットを 1 にすると、送信 FIFO に書き込まれた順にトランスミットシフトレジスタ (SCTSR) にいったんデータを転送し、パリティジェネレータ (P/G) によりパリティビットを付加し、TXD 端子からシリアルデータを送信します。

送信 FIFO にデータを書き込むたびに FIFO データ数レジスタ (SCFDR) の T4 ~ 0 ビットがカウントアップされ、SCTSR にデータが転送されるたびに T4 ~ 0 ビットをカウントダウンするように動作しますので、現在の送信 FIFO 内に存在しているデータの数は SCFDR の T4 ~ 0 ビットを読み出すことで知ることができます。

T4 ~ 0 ビットが H'10 のときは、16 段すべてにデータが書き込まれた状態であり、この状態でさらにデータを書き込むと T4 ~ 0 ビットのカウンタアップを行わず、書き込んだデータも失われます。

送信トリガ数を設定して DMAC で送信データを FIFO に書き込む場合には、「14.2.10 FIFO コントロールレジスタ (SCFCR)」で示した SCFTDR の空き数を超えるデータを書き込まないように注意してください。

### (2) シリアルデータ受信動作時

受信時には、RXD 端子から入力されたシリアルデータをいったんレシブシフトレジスタ (SCRSR) に、シリアルステータス 2 レジスタ (SC2SSR) の RLM ビットの設定に従って取り込みます。この時、パリティビットのチェックを行いパリティエラーであれば、そのデータに対応する P (パリティエラーフラグ) フラグを 1 にし、またストップビットのチェックを行いフレーミングエラーであれば、そのデータに対応する F (フレーミングエラーフラグ) フラグを 1 にします。受信 FIFO バッファは 10 ビット構成になっており、8 ビットデータとともに対応する P、F フラグを一緒に格納します。

#### (a) 通常動作時の受信 FIFO 制御

CPU あるいは DMAC により受信 FIFO バッファに格納された受信データを読み出します。

SCRSR から受信 FIFO にデータが転送されるたびに、SCFDR の R4 ~ 0 ビットがカウンタアップされ、受信 FIFO 内の受信データを CPU あるいは DMAC が読み出すたびに、R4 ~ 0 ビットをカウンタダウンするように動作しますので、現在受信 FIFO 内に受信したデータの数を SCFDR の R4 ~ 0 ビットを読み出すことで知ることができます。

R4 ~ 0 ビットが H'10 のときは、16 段すべてに受信データが転送された状態であり、受信 FIFO 内のデータを CPU あるいは DMAC が読み出す前に次のシリアル受信が完了した場合にはオーバランエラーとなりシリアルデータは失われます。また、R4 ~ 0 ビットが H'00 のときに受信 FIFO のデータを読み出すと不定になります。

#### (b) エラーデータ受信時の受信 FIFO 制御

SCRSR から受信 FIFO にデータが転送されると同時に、P、F フラグも転送され、これらのフラグが 1 であればエラーカウンタがカウンタアップされシリアルステータス 1 レジスタ (SC1SSR) の PER3 ~ 0、FER3 ~ 0 が更新されます。また、CPU あるいは DMAC により受信 FIFO 内のデータを読み出した場合に P、F フラグが 1 であればエラーカウンタはカウンタダウンされます。さらに、読み出した受信データの P、F フラグが SC1SSR の PER、FER フラグに反映されます。PER および FER のセットは、パリティエラーあるいはフレーミングエラーのあるデータを受信 FIFO から読み出したときに行われ、Rx/D 端子からシリアルデータを受信したときにはパリティエラーおよびフレーミングエラーがあってもセットされません。PER および FER のクリアは、パリティエラーあるいはフレーミングエラーのないデータを受信 FIFO から読み出したときに行われます。

パリティエラー、フレーミングエラーがあっても、そのデータは受信 FIFO 内に転送されます。その後受信動作を続けるかどうかは SC2SSR の EI ビットの設定により選択できます。EI ビットを 1 に設定し受信動作を続けるよう選択した場合、エラー後も受信データを順次受信 FIFO 内に転送していきませんが、16 段受信 FIFO バッファの何段目のデータにエラーがあるかは FIFO エラーレジスタ (SCFER) の ED15 ~ 0 ビットを読み出すことにより知ることができます。

受信トリガ数を設定した DMAC で受信データを受信 FIFO から読み出す場合には、「14.2.10 FIFO コントロールレジスタ (SCFCR)」で示した受信トリガ数を超えるデータを読み出さないように注意してください。

(c) DR フラグによる受信 FIFO 制御

受信トリガ数以上のデータを受信した場合には、RXI 割り込み (RDF のみ) により CPU あるいは DMAC に受信データの読み出しの要求を出しますが、受信トリガ数以下のデータを受信した状態ですべての受信が完了した場合には RXI 割り込みは発生しません。この場合には最終データ受信完了後の 16ETU 時間後に DR フラグがセットされ ERI 割り込みが発生しますので、CPU により SCFDR の R4 ~ 0 ビットを読み出して受信 FIFO 内に残っているデータ数を確認し、すべての受信 FIFO 内のデータを読み出してください。

【注】 ETU は 8 ビット長 1 ストップビットフォーマット時の 1.6 フレーム時間に相当します。

ETU : element time unit = sec/bit

### 14.3.6 IrDA モード時の動作

IrDA モードでは、IrDA1.0 赤外線通信仕様を満足するよう、TxD / RxD 送受信データの波形を変更します。これを赤外線送受信トランシーバ / レシーバと接続することで、IrDA1.0 規格に準拠した赤外線送受信を実現することができます。

IrDA1.0 仕様では、通信はまず 9600bps の速度で実行され、その後必要に応じて転送レートを変更させることができます。ただし、通信速度は本モジュールでは自動的に変更されません。したがって、通信を実行する場合には通信速度を確認し、ソフトウェアで本モジュールに適当な速度を設定してください。

【注】 IrDA モードでは、シリアルコントロールレジスタ (SCSCR) の TE ビットが 1 にセット (通信を許可) されていると受信を実行することができません。受信を行う場合は SCSCR の TE ビットを 0 にクリアしてください。

(1) 送信

SCIF からのシリアル出力信号 (UART フレーム) の場合、その波形は修正され、図 14.24 に示すように信号は IrDA モジュールによって IR フレームシリアル出力信号に変換されます。

シリアルデータが 0 の場合、IrDA モードレジスタ (SCIMR) の PSEL ビットが 0 のときは、IR フレームの 3/16 ビット幅のパルスが生成、出力され、PSEL ビットが 1 のときは、シリアルモードレジスタ (SCSMR) の ICK3 ~ 0 ビットで設定したビットレートの 3/16 ビット幅のパルスが生成、出力されます。シリアルデータが 1 の場合、パルスは出力されません。

赤外線 LED は 3/16 幅に復調されたこの信号で駆動します。

(2) 受信

受信した IR フレームの 3/16 ビット幅のパルスは、図 14.24 に示すように復調後 UART フレームに変換されます。

0 への復調はパルス出力に対して実行され、1 への復調はパルス出力のない時に実行されます。

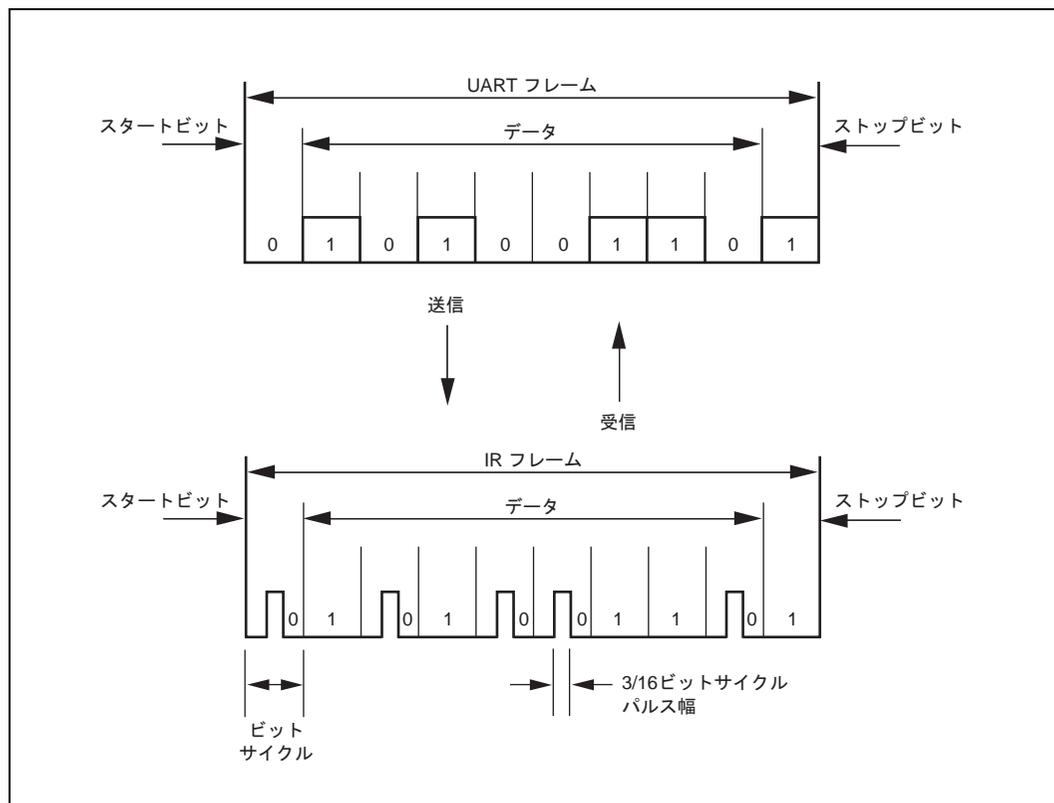


図 14.24 IrDA モード送受信動作

### (3) パルス幅の選択

送信時の IR フレームのパルス幅は、IrDA モードレジスタ (SCIMR) の PSEL ビットの設定により、送信ビットレートの 3/16 の幅で出力するか、それより小さいパルス幅で出力するか選択できます。

SCIF は、送信フレームのビットレートを生成するボーレートジェネレータと、パルス幅を可変とするために IRCLK を生成するボーレートジェネレータを内蔵しています。

SCIMR の PSEL ビットを 0 に設定した場合、ビットレートレジスタ (SCBBR) で設定したビットレートの 3/16 の幅を IR フレームのパルス幅として出力します。パルス幅が直接赤外線発光の時間となりますので、省電力化のためにパルス幅を極力小さくしたい場合は、SCIMR の PSEL ビットを 1 に設定し、さらにシリアルモードレジスタ (SCSMR) の ICK3 ~ 0 ビットの設定により IRCLK を生成し、設定可能な最小のパルス幅で出力します。

ここで、IR フレームのパルス幅は最小で 115.2kbps のビットレートの 3/16 ( $=1.63 \mu\text{s}$ ) とする必要があります。この最小パルス幅の場合  $\text{IRCLK} = 921.6\text{kHz}$  となりますので、設定可能な最小パルス幅を得るための ICK3 ~ 0 ビットの設定は次の式で求められます。

## 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

Pφ : 動作クロック周波数  
 IRCLK : 921.6kHz (固定)  
 N : ICK3~0の設定値 (0<=N<=15)

$$N \geq \frac{P\phi}{2 \times \text{IRCLK}} - 1$$

例えば P = 20MHz のとき、N = 10 となります。

最小パルス幅を得るための適用可能な ICK3 ~ 0 ビットの設定と動作周波数の関係を表 14.12 に示します。

表 14.12 IrDA モード時の ICK3 ~ 0 ビットと動作周波数の関係 (PSEL = 1 のとき)

動作周波数 P (MHz)	SCSMR の ICK3 ~ 0 ビットの設定			
	ICK3	ICK2	ICK1	ICK0
2	0	0	0	0
3	0	0	0	1
5	0	0	1	0
6	0	0	1	1
8	0	1	0	0
10	0	1	0	1
12	0	1	1	0
14	0	1	1	1
16	1	0	0	0
18	1	0	0	1
20	1	0	1	0
21	1	0	1	1
22	1	0	1	1
23	1	1	0	0
24	1	1	0	1
25	1	1	0	1
26	1	1	1	0
27	1	1	1	0
28	1	1	1	1

## 14.4 SCIF の割り込み要因と DMAC

SCIF は、ブレーク割り込み (BRI) 要求、受信エラー割り込み (ERI) 要求、受信 FIFO データフル割り込み (RXI) 要求、送信 FIFO データエンpty 割り込み (TXI) 要求の 4 種類の割り込み要因を持っています。

表 14.13 に各割り込み要因と優先順位を示します。各割り込み要因は、SCSCR の TIE、RIE ビットで、許可または禁止ができます。また、各割り込み要求はそれぞれ独立に割り込みコントローラに送られます。

シリアルステータスレジスタ (SC1SSR) の TDFE フラグが 1 にセットされると、TXI 割り込み要求が発生します。TXI 割り込み要求で内蔵 DMAC を起動してデータ転送を行うことができます。TDFE ビットは DMAC によるトランスミット FIFO データレジスタ (SCFTDR) への書き込みがすべて終了すると自動的に 0 にクリアされます。

SC1SSR の RDF フラグが 1 にセットされると、RXI 割り込み要求が発生します。RXI 割り込み要求で内蔵 DMAC を起動してデータ転送を行うことができます。RDF ビットは DMAC によるレシーブ FIFO データレジスタ (SCFRDR) の読み出しがすべて行われると自動的に 0 にクリアされます。

また、ER フラグが 1 にセットされると、ERI 割り込み要求が発生します。この ERI 割り込み要求で内蔵 DMAC の起動はできません。

さらに、BRK フラグが 1 にセットされると、BRI 割り込み要求が発生します。この BRI 割り込み要求で内蔵 DMAC の起動はできません。

なお、TXI 割り込みは送信データを書き込み可能なことを示し、RXI 割り込みは受信データが SCFRDR にあることを示しています。

表 14.13 SCIF 割り込み要因

割り込み要因	内容	DMAC の起動	リセット解除時の優先順位
ERI	受信エラー (ER) による割り込み	不可	高 ↑ ↓ 低
RXI	受信データフル (RDF) またはデータレディ (DR) による割り込み	可 (RDF のみ)	
BRI	ブレーク (BRK) による割り込み	不可	
TXI	送信データエンpty (TDFE) による割り込み	可	

## 14.5 使用上の注意

SCIF を使用する際は、以下のことに注意してください。

### (1) SCFTDR への書き込みと TDFE フラグの関係について

シリアルステータスレジスタ (SC1SSR) の TDFE フラグはトランスミット FIFO データレジスタ (SCFTDR) 内に書き込んだ送信データ数が、FIFO コントロールレジスタ (SCFCR) の TTRG1、0 ビットで設定した送信トリガ数以下になったときセットします。TDFE がセットされた後、SCFTDR の空きデータ数まで送信データを書き込むことができ効率よい連続送信が可能となります。

しかし TDFE フラグは SCFTDR に書き込まれているデータ数が送信トリガ数以下の場合には 1 読み出し後 0 クリアしても再び 1 にセットされます。TDFE のクリアは送信トリガ数より多い送信データが SCFTDR に格納されたときに行ってください。

SCFTDR 内の送信データ数は FIFO データ数レジスタ (SCFDR) の上位 8 ビットで知ることができます。

## (2) 複数の受信エラーが同時に発生した場合の動作について

複数の受信エラーが同時に発生した場合、SC1SSR、SC2SSR の各ステータスフラグの状態は、表 14.14 のようになります。また、オーバランエラーが発生した場合にはレシープシフトレジスタ (SCRSR) からレシープ FIFO データレジスタ (SCFRDR) へのデータ転送は行われず、受信データは失われます。

表 14.14 SC1SSR、SC2SSR のステータスフラグの状態と受信データの転送

受信エラーの状態	SC1SSR、SC2SSR のステータスフラグ				受信データ転送	
	RDF	ORER	FER	PER	SCRSR	SCFRDR
オーバランエラー	1	1	0	0		x
フレーミングエラー	0	0	1	0		
パリティエラー	0	0	0	1		
オーバランエラー+フレーミングエラー	1	1	1	0		x
オーバランエラー+パリティエラー	1	1	0	1		x
フレーミングエラー+パリティエラー	0	0	1	1		
オーバランエラー+フレーミングエラー+パリティエラー	1	1	1	1		x

【注】 : SCRSR SCFRDR に受信データを転送します。  
x : SCRSR SCFRDR に受信データを転送しません。

## (3) ブレークの検出と処理について

フレーミングエラー (FER) 検出時に RxD 端子の値を直接読み出すことでも、ブレークを検出できます。ブレークでは、RxD 端子からの入力がすべて 0 になりますので FER フラグがセットされ、またパリティエラー (PER) もセットされる場合があります。

SCIF は、ブレークを受信した後は、SCFRDR への受信データの転送は停止しますが、受信動作は続けています。そのため、FER、BRK フラグを 0 にクリアしても再び 1 にセットされますので注意してください。

## (4) ブレークの送り出し

TxD 端子は、I/O ポートのデータレジスタ (DR) とピンファンクションコントローラ (PFC) のコントロールレジスタ (CR) により入出力方向とレベルが決まる汎用入出力端子になります。これを利用してブレークの送り出しができます。

PFC の設定を行うまではマーク状態を DR の値で代替します。このため、最初は 1 を出力する出力ポートに設定しておきます。

シリアル送信時にブレークを送り出したいときは DR を 0 にクリアした後、PFC で TxD 端子を出力ポートに設定します。

TE ビットを 0 にクリアすると現在の送信状態とは無関係に送信部は初期化されます。

## (5) 受信エラーフラグと送信動作について (クロック同期モードのみ)

受信エラーフラグ (ORER、PER3 ~ 0、FER3 ~ 0) が 1 にセットされた状態では、TDFE フラグを 1 にセットしても送信を開始できません。必ず送信開始時には、受信エラーフラグを 0 にクリアしておいてください。

また、RE ビットを 0 にクリアしても受信エラーフラグは 0 にクリアできませんので注意してください。

## (6) 調歩同期モードの受信データサンプリングタイミングと受信マージン

調歩同期モードでは、SCIF は転送レートの 16 倍 (または 8 倍、4 倍) の周波数の基本クロックで動作しています。

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

受信時に SCIF は、スタートビットの立ち下がり基本クロックでサンプリングして、内部を同期化します。また、受信データを基本クロックの 8 または 4 または 2 クロック目の立ち上がりエッジで内部に取り込みます。これを図 14.25 に示します。

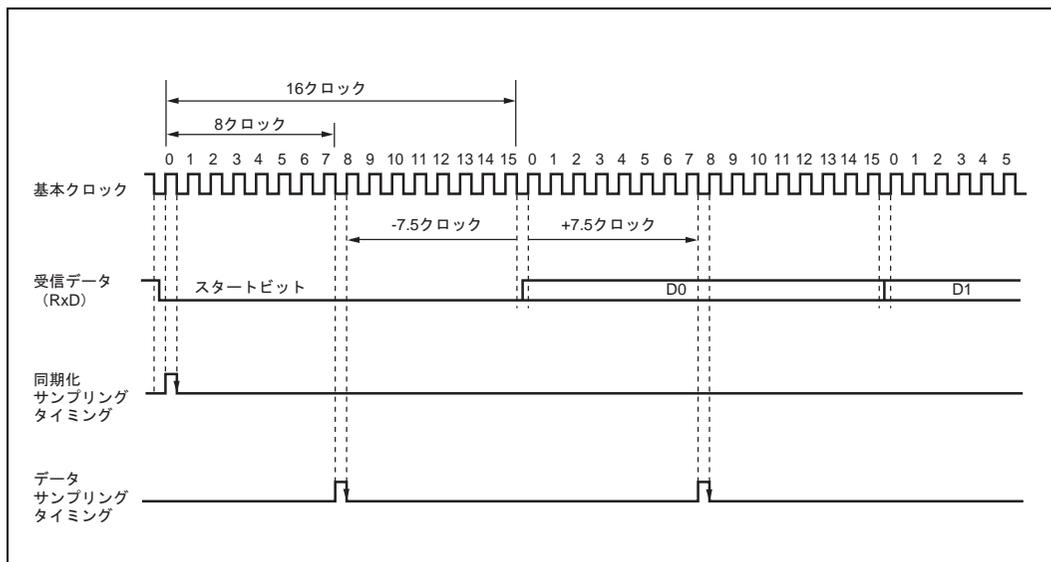


図 14.25 調歩同期式モードの受信データサンプリングタイミング  
(基本クロックは転送レートの 16 倍の周波数を使用、8 クロック目でサンプリング時)

したがって、調歩同期式モードでの受信マージンは式 (1) のように表すことができます。

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100 (\%) \dots\dots (1) \text{ 式}$$

M : 受信マージン (%)

N : クロックに対するビットレートの比 (N = 16、8、4)

D : クロックデューティ (D = 0 ~ 1.0)

L : フレーム長 (L = 9 ~ 12)

F : クロック周波数の偏差の絶対値

式 (1) で、F = 0、D = 0.5 とすると、受信マージンは式 (2) より 46.875% となります。

D = 0.5、F = 0、N = 16 のとき

$$M = \left( 0.5 - \frac{1}{2 \times 16} \right) \times 100\% \\ = 46.875\% \dots\dots \text{式 (2)}$$

ただし、この値はあくまでも計算上の値ですので、システム設計の際には 20 ~ 30% の余裕を持たせてください。

## (7) クロック同期外部クロックモード時の注意事項

- TE = 1、RE = 1 に設定するのは、外部クロック SCK を 0 → 1 にしてから周辺動作クロック 4 クロック以上経過してからにしてください。
- TE = RE = 1 に設定するのは、必ず外部クロック SCK が 1 のときにしてください。
- 受信時において、RxD の D7 ビットの SCK 入力の立ち上がりエッジから周辺動作クロック 2.3 ~ 3.5 クロック後に RE = 0 にすると RDF = 1 になりますが、SCFRDR へのコピーができませんので注意してください。

## (8) クロック同期内部クロックモード時の注意事項

受信時において、RxD の D7 ビットの SCK 出力の立ち上がりエッジから周辺動作クロック 1.5 クロック後に RE = 0 にすると RDF = 1 になりますが、SCFRDR へのコピーができませんので注意してください。

## (9) DMAC 使用上の注意事項

同期クロックに外部クロックソースを使用する場合、DMAC による SCFTDR の更新後、P で 5 クロック以上経過した後に、送信クロックを入力してください。SCFTDR の更新後 P で 4 クロック以内に送信クロックを入力すると、誤動作することがあります。(図 14.26)

DMAC により、SCFRDR の読み出しを行うときは必ず起動要因を当該 SCIF の受信 FIFO データフル割り込み (RXI) に設定してください。

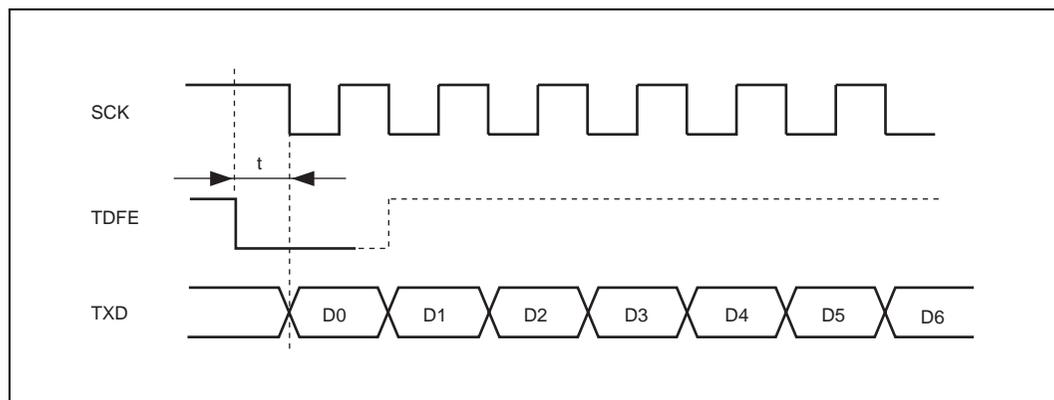


図 14.26 DMAC によるクロック同期式送信時の例

## (10) SCFRDR の読み出しと RDF フラグについて

シリアルステータスレジスタ (SCISSR) の RDF フラグは、レシーブ FIFO データレジスタ (SCFRDR) 内の受信データ数が FIFO コントロールレジスタ (SCFCR) の RTRG1、0 ビットで設定した受信トリガ数以上になったときセットします。RDF がセットされた後、SCFRDR からトリガ数分の受信データを読み出すことで効率のよい連続受信が可能です。

ただし、読み出し後も SCFRDR 内のデータ数がトリガ数以上の場合、RDF フラグを 0 クリアしても再び 1 にセットされますので、SCFRDR 内のデータ数がトリガ数より少なくなるように受信データを読み出した後、RDF フラグの 1 を読み出し 0 クリアしてください。

SCFRDR 内の受信データ数は FIFO データ数レジスタ (SCFDR) の下位 8 ビットで知ることができます。

(11) オーバラン発生時の SCFRDR の読み出しについて

レシーブ FIFO データレジスタ (SCFRDR) に 16 個のデータが格納してあるにもかかわらず、受信動作を続けるとオーバランが発生します。

この状態で SCFRDR にリードを行った場合、17 回目のリードではオーバランを発生させたデータが読み出されます。18 回目以降は、不定となります。

また、FIFO データ数レジスタ (SCFDR) の下位 8 ビットで示される SCFRDR 内の受信データ数は、SCFRDR の 1 回目の読み出し以降、正常な受信データ数 + 1 となりますので注意してください。

(12) SCIF 初期化フローチャートと受信 FIFO データフル割り込み (RXI) 要求について

【不具合内容】

SCIF 機能を使用時、図 14.4 に示す「SCIF の初期化フローチャートの例」を続けて 2 回以上実行すると、受信データが無い状態にもかかわらず、2 回目以降の初期化フローで SCIF 受信 FIFO データフル割り込み (RXI) 要求がセットされる場合があります。

【発生条件】

図 14.27 に、SCIF 初期化フローを 2 回実施した場合のフローチャート例を示します。

1 回目の初期化フローにおいて、FIFO コントロールレジスタ (SCFCR) の受信 FIFO データ数トリガ (RTRG1、RTRG0) を設定し、2 回目の初期化フローにおいて、再度受信 FIFO データ数トリガ (RTRG1、RTRG0) 設定値の変更を行うと、SCIF 受信 FIFO データフル割り込み (RXI) 要求がセットされる場合があります。

【回避方法】

2 回目の初期化フローにおいて、FIFO コントロールレジスタ (SCFCR) にライトアクセスする場合は、以下の (1)、(2) いずれかを採用して使用してください。

- (1) FIFO コントロールレジスタ (SCFCR) を一度読み出し、読み出した受信 FIFO データ数トリガ (RTRG1、RTRG0) 設定値と同じ値を書き込むようにしてください。
- (2) FIFO コントロールレジスタ (SCFCR) の受信 FIFO データ数トリガ (RTRG1、RTRG0) 設定値を変更する場合は、受信 FIFO データ数トリガ (RTRG1、RTRG0) 設定値変更前にシリアルコントロールレジスタ (SCSCR) のレシーブインタラプトイネーブル (RIE) ビットを "0" にして、受信 FIFO データフル割り込み (RXI) 要求をマスクしてください。

また、FIFO コントロールレジスタ (SCFCR) 書き込み変更後は、シリアルステータス 1 レジスタ (SC1SSR) のレシーブデータレジスタフル割り込み (RDF) 要求をクリアし、シリアルコントロールレジスタ (SCSCR) のレシーブインタラプトイネーブル (RIE) ビットを "1" にセットして、マスク解除として下さい。

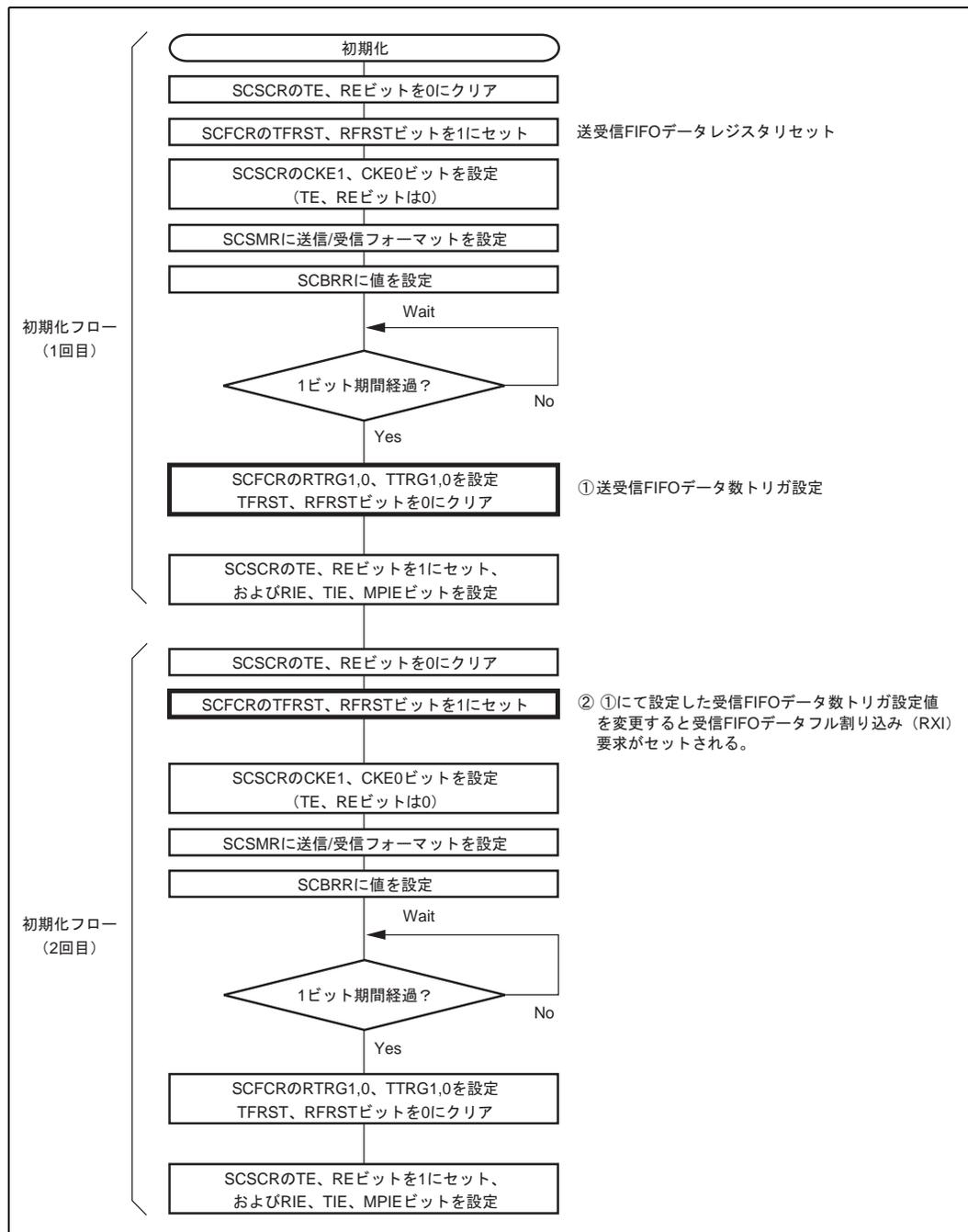


図 14.27 SCIF 初期化フローチャート

#### 14. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

---

---

## 15. シリアル I/O ( SIO )

---

### 15.1 概要

本 LSI には、3 チャンネルの簡易同期方式シリアル I/O があります。シリアル I/O は、主に CODEC、MODEM のアナログフロントエンドに対する本 LSI のインタフェースとして機能します。

#### 15.1.1 特長

SIO には次の特長があります。

- 全二重動作  
独立した送受信レジスタと独立した送受信クロック
- ダブルバッファ構造の送受信ポート  
データの連続送受信が可能
- インターバル転送モードと連続転送モード
- メモリマップされたレシーブレジスタ、トランスミットレジスタ、コントロールレジスタ、ステータスレジスタ  
SIRSR、SITSR を除いてこれらのレジスタはメモリマップされており MOV 命令によってアクセス可能
- 8 ビット / 16 ビットのデータ長を選択可能
- ボーリングまたは割り込みによるデータ転送通信  
シリアルステータスレジスタの受信データレジスタフルフラグ ( RDRF )、送信データレジスタエンプティフラグ ( TDRE ) をボーリングしデータ転送の監視が可能。受信割り込み要求フラグ、送信割り込み要求フラグをセットして、データ転送の際、割り込み要求を発生させることが可能
- データ I/O に対して MSB 先頭の転送

図 15.1 にシリアル I/O のブロック図を示します。

15. シリアルI/O (SIO)

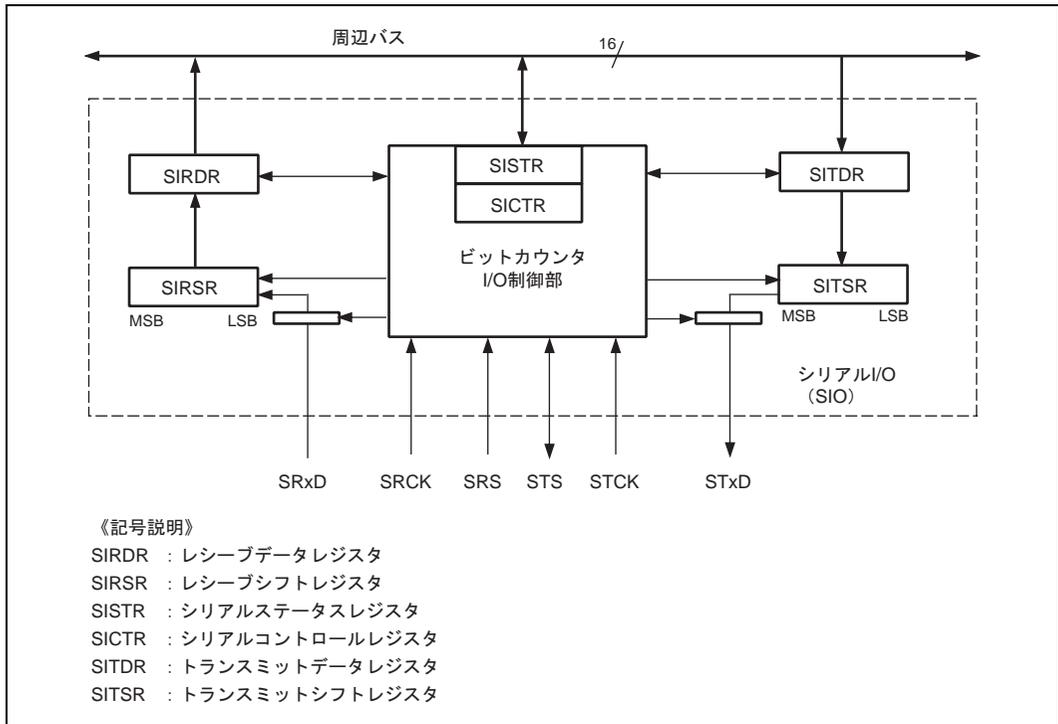


図 15.1 SIO ブロック図

表 15.1 に外部端子の機能を示します。チャンネルは独立しているので、チャンネル番号は以降の章においては信号名から外します。

表 15.1 SIO の外部端子

チャンネル	名称	端子	入出力	機能
0	シリアル受信データ入力端子	SRxD0	入力	シリアルデータ入力ポート 0
	シリアル受信クロック入力端子	SRCK0	入力	シリアル受信クロックポート 0
	シリアル受信同期入力端子	SRS0	入力	シリアル受信同期入力ポート 0
	シリアル送信データ出力端子	STxD0	出力	シリアルデータ出力ポート 0
	シリアル送信クロック入力端子	STCK0	入力	シリアル送信クロックポート 0
	シリアル送信同期入出力端子	STS0	入出力	シリアル送信同期入出力ポート 0
1	シリアル受信データ入力端子	SRxD1	入力	シリアルデータ入力ポート 1
	シリアル受信クロック入力端子	SRCK1	入力	シリアル受信クロックポート 1
	シリアル受信同期入力端子	SRS1	入力	シリアル受信同期入力ポート 1
	シリアル送信データ出力端子	STxD1	出力	シリアルデータ出力ポート 1
	シリアル送信クロック入力端子	STCK1	入力	シリアル送信クロックポート 1
	シリアル送信同期入出力端子	STS1	入出力	シリアル送信同期入出力ポート 1
2	シリアル受信データ入力端子	SRxD2	入力	シリアルデータ入力ポート 2
	シリアル受信クロック入力端子	SRCK2	入力	シリアル受信クロックポート 2
	シリアル受信同期入力端子	SRS2	入力	シリアル受信同期入力ポート 2
	シリアル送信データ出力端子	STxD2	出力	シリアルデータ出力ポート 2
	シリアル送信クロック入力端子	STCK2	入力	シリアル送信クロックポート 2
	シリアル送信同期入出力端子	STS2	入出力	シリアル送信同期入出力ポート 2

【注】 リセット時、全端子はハイインピーダンス状態に初期化されます。

## 15.2 レジスタ構成

表 15.2 に SIO のレジスタを示します。チャンネルは独立しているので、チャンネル番号は以降の章においては信号名から外します。

表 15.2 レジスタ構成

チャンネル	レジスタ	略語	R/W	初期値	アドレス	アクセスサイズ (ビット)
0	レシーブシフトレジスタ 0	SIRSR0	-	-	-	-
	レシーブデータレジスタ 0	SIRDR0	R	H'0000	H'FFFFFFC00	8, 16, 32
	トランスミットシフトレジスタ 0	SITSR0	-	-	-	-
	トランスミットデータレジスタ 0	SITDR0	R/W	H'0000	H'FFFFFFC02	8, 16, 32
	シリアルコントロールレジスタ 0	SICTR0	R/W	H'0000	H'FFFFFFC04	8, 16, 32
	シリアルステータスレジスタ 0	SISTR0	R/(W)*	H'0002	H'FFFFFFC06	8, 16, 32
1	レシーブシフトレジスタ 1	SIRSR1	-	-	-	-
	レシーブデータレジスタ 1	SIRDR1	R	H'0000	H'FFFFFFC10	8, 16, 32
	トランスミットシフトレジスタ 1	SITSR1	-	-	-	-
	トランスミットデータレジスタ 1	SITDR1	R/W	H'0000	H'FFFFFFC12	8, 16, 32
	シリアルコントロールレジスタ 1	SICTR1	R/W	H'0000	H'FFFFFFC14	8, 16, 32
	シリアルステータスレジスタ 1	SISTR1	R/(W)*	H'0002	H'FFFFFFC16	8, 16, 32
2	レシーブシフトレジスタ 2	SIRSR2	-	-	-	-
	レシーブデータレジスタ 2	SIRDR2	R	H'0000	H'FFFFFFC20	8, 16, 32
	トランスミットシフトレジスタ 2	SITSR2	-	-	-	-
	トランスミットデータレジスタ 2	SITDR2	R/W	H'0000	H'FFFFFFC22	8, 16, 32
	シリアルコントロールレジスタ 2	SICTR2	R/W	H'0000	H'FFFFFFC24	8, 16, 32
	シリアルステータスレジスタ 2	SISTR2	R/(W)*	H'0002	H'FFFFFFC26	8, 16, 32

【注】 \* フラグをクリアするための 0 書き込みのみ行ってください (フラグの 1 読み出し後)。

### 15.2.1 レシーブシフトレジスタ (SIRSR)

ビット :	15	14	13	...	3	2	1	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
初期値 :	-	-	-	...	-	-	-	-
R/W :	-	-	-	...	-	-	-	-

SIRSR は、シリアルデータを受信する 16 ビットのレジスタです。データは、MSB を先頭に、シリアル受信クロック (SRCK) 信号の下がりエッジに同期して SRxD 端子より取り込まれ、SIRSR にシフトして入力されます。データ長は、該当するシリアルコントロールレジスタ (SICTR) の送受信データ長セレクトビット (DL) で設定されます。SIRSR へのデータ転送が完了すると、データ内容はレシーブデータレジスタ (SIRDR) に自動的に転送され、シリアルステータスレジスタ (SISTR) の受信データレジスタフルフラグ (RDRF) がセットされます。

RDRF フラグをクリアする前に次のデータワードの入力動作が終了すると、オーバランエラーが発生し、SISTR の受信オーバランエラーフラグ (RERR) がセットされ、オーバランエラー信号が割り込みコントローラ (INTC) に送られます。このとき SIRSR のデータは SIRDR のデータを上書きします。

### 15.2.2 レシーブデータレジスタ (SIRDR)

ビット :	15	14	13	...	3	2	1	0
	□ □ □			...	□ □ □ □			
初期値 :	0	0	0	...	0	0	0	0
R/W :	R	R	R	...	R	R	R	R

SIRDR は、シリアル受信データを格納する 16 ビットのレジスタです。SIRSR から SIRDR へデータが転送されると同時にシリアルステータスレジスタ (SISTR) の受信データレジスタフルフラグ (RDRF) がセットされます。SICTR の受信割り込みイネーブルフラグ (RIE) がセットされていると、受信データフル割り込み (RDFI) 要求が割り込みコントローラ (INTC) と DMA コントローラ (DMAC) に送られます。フラグをクリアすると、この割り込み要求信号は発生しません。SIRDR が DMAC によって読み出されると、RDRF フラグは自動的にクリアされます。SIRDR の値はリセット時に H'0000 に初期化されます。

### 15.2.3 トランスミットシフトレジスタ (SITSR)

ビット :	15	14	13	...	3	2	1	0
	□ □ □			...	□ □ □ □			
初期値 :	-	-	-	...	-	-	-	-
R/W :	-	-	-	...	-	-	-	-

SITSR は、シリアルデータを送信する 16 ビットのレジスタです。データは MSB を先頭に、シリアル送信クロック (STCK) 信号の立上がりエッジに同期して本レジスタの内容をシフトし、STxD 端子より出力されます。シリアルコントロールレジスタ (SICTR) の送受信データ長セレクトビット (DL) は、転送データ長を設定します。DL ビットが 0 に設定されると (データ長は 8 ビット)、SITDR の下位 8 ビットが出力されます。シリアル送信同期信号 (STS) がハイレベルになるとき、もしくは SICTR の同期イネーブル (SE) ビットがセットされておらず、最後のデータ送信が終了したときにトランスミットデータレジスタ (SITDR) の内容は SITSR に転送され、TDRE が 0 ならば、続けて TDRE がセットされます。もし TDRE がクリアされる前に次のデータ出力が始まると、アンダランエラーが発生し、SISTR の送信アンダランエラーフラグ (TERR) をセットし、INTC に送信アンダランエラー割り込み要求を送ります。

## 15.2.4 トランスミットデータレジスタ (SITDR)

ビット :	15	14	13	...	3	2	1	0
	<input type="text"/>	<input type="text"/>	<input type="text"/>	...	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
初期値 :	0	0	0	...	0	0	0	0
R/W :	R/W	R/W	R/W	...	R/W	R/W	R/W	R/W

SITDR は、シリアル送信データを格納する 16 ビットのレジスタです。SISTR の送信データエンプティフラグ (TDRE) が 1 のときに SITDR へのデータ書き込みを行ってください。もし 0 のときに SITDR に書き込むとそれ以前に存在していたデータの上に書き込まれてしまいます。STS がハイレベルになるか、SICTR の SE ビットが 0 でトランスミットシフトレジスタ (SITSR) のデータ出力が終了すると、SITDR のデータは自動的に SITSR に転送され、続けて TDRE が 0 ならば TDRE がセットされます。このとき、送信割り込みイネーブルフラグ (TIE) がセットされていると、送信データエンプティ割り込み (TDEI) 要求が INTC と DMAC に送られます。TIE がクリアされていると、この割り込み要求は発生しません。DMAC が SITDR に書き込んだ場合、TDRE フラグは自動的にクリアされます。TDRE フラグはハードウェアによってのみセットされます。SITDR はリセット時 H'0000 に初期化されます。

## 15.2.5 シリアルコントロールレジスタ (SICTR)

ビット :	15	14	13	12	11	10	9	8
	<input type="text"/>							
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R
ビット :	7	6	5	4	3	2	1	0
	<input type="text"/>							
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R/W						

SICTR は、シリアルポートの制御のためのパラメータを設定する 16 ビットのレジスタです。SICTR はリセット時に H'0000 に初期化されます。

なお、ビット 4、5、6 (TM、SE、DL) を変更するときは TE、RE を 0 にしておいてください。

- ・ ビット 15 ~ 7 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
- ・ ビット 6 : 転送モード制御 (TM)  
送信同期信号を外部ソースから入力するか、本 LSI で内部的に生成するかを指定します。このフラグをクリアすると、送信同期信号は STS 端子入力となります。このフラグをセットすると送信同期信号は本 LSI が生成し、外部デバイスに対して STS 端子から出力されます。このビットは受信には影響しません。

ビット 6	説明
TM	
0	送信開始指示は STS 端子より入力する外部信号を使用 (初期値)
1	送信開始指示は STS 端子から出力する内部信号を使用

・ ビット 5 : 同期信号イネーブル (SE)

同期信号を全シリアルデータ転送に対して使用するのか、最初の転送だけに使用するのかを指定します。

このビットを 0 にすると、同期信号 (SRS、STS) は最初のデータ転送時のみ必要で以降のデータ転送には、不要となります。このビットを 1 にすると、同期信号はすべてのデータ転送時に必要となります。

ビット 5	説 明
SE	
0	連続モード。SRS、STS は最初のデータ転送だけに使用 (初期値)
1	インターバルモード。SRS、STS はすべてのデータに使用

・ ビット 4 : 送受信データ長セレクト (DL)

シリアル I/O の転送データ長を設定します。このビットの初期値は 0 で、8 ビットデータ長を意味します。8 ビットデータ長を指定すると、各 I/O レジスタの下位 8 ビットが使用されます。

ビット 4	説 明
DL	
0	8 ビット転送データ長 (初期値)
1	16 ビット転送データ長

・ ビット 3 : 送信割り込みイネーブル (TIE)

送信データエンpty割り込みを許可します。このビットの初期値は 0 です。

ビット 3	説 明
TIE	
0	送信割り込みを禁止 (初期値)
1	送信割り込みを許可

・ ビット 2 : 受信割り込みイネーブル (RIE)

受信データフル割り込みを許可します。このビットの初期値は 0 です。

ビット 2	説 明
RIE	
0	受信割り込みを禁止 (初期値)
1	受信割り込みを許可

## 15. シリアル I/O (SIO)

### ・ビット 1：送信イネーブル (TE)

データ送信を許可します。このフラグをクリアすると、STxD、STCK、STS 端子はハイインピーダンス状態になります。

ビット 1	説明
TE	
0	送信を禁止し、STxD、STCK、STS 端子をハイインピーダンス状態にする (初期値)
1	送信を許可する

### ・ビット 0：受信イネーブル (RE)

データ受信を許可します。このフラグをクリアすると、SRxD、SRCK、SRS 端子はハイインピーダンス状態になります。

ビット 0	説明
RE	
0	受信を禁止し、SRxD、SRCK、SRS 端子をハイインピーダンス状態にする (初期値)
1	受信を許可する

## 15.2.6 シリアルステータスレジスタ (SISTR)

ビット :	15	14	...	4	3	2	1	0
	—	—	...	—	TERR	RERR	TDRE	RDRF
初期値 :	0	0	...	0	0	0	1	0
R/W :	R	R	...	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】 \* フラグをクリアするための 0 書き込みのみ行ってください。

SISTR は、シリアル I/O の状態を示す 16 ビットのレジスタです。SISTR はリセット時に H'0002 に初期化されます。

### ・ビット 15 ~ 4：予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### ・ビット 3：送信アンダランエラー (TERR)

送信アンダランが発生したことを示すフラグです。

ビット 3	説明
TERR	
0	送信中、または送信が正常終了 (初期値) TERR は以下の条件で 0 にクリアされます • TERR ビットから 1 を読み出した後、0 を書き込んだとき • プロセッサがリセット状態となったとき
1	送信アンダランエラー発生 TDRE = 1 の状態でデータ送信を開始するとき TERR は 1 にセットされます

- ・ ビット 2 : 受信オーバーランエラー (RERR)  
受信オーバーランが発生したことを示すフラグです。

ビット 2	説 明
RERR	
0	受信中、または受信が正常終了 RERR は以下の条件で 0 にクリアされます <ul style="list-style-type: none"> <li>• RERR ビットから 1 を読み出した後、0 を書き込んだとき</li> <li>• プロセッサがリセット状態となったとき</li> </ul> (初期値)
1	受信オーバーランエラー発生 RDRE = 1 の状態でデータ受信が終了したとき RERR は 1 にセットされます

- ・ ビット 1 : 送信データレジスタエンプティ (TDRE)  
SITDR レジスタが空で次のデータを書き込める状態であることを示すフラグです。

ビット 1	説 明
TDRE	
0	SITDR の送信データは有効 TDRE は以下の条件で 0 にクリアされます <ul style="list-style-type: none"> <li>• TDRE ビットから 1 を読み出した後、0 を書き込んだとき</li> <li>• DMAC が SITDR にデータを書き込んだとき</li> </ul> (初期値)
1	SITDR の送信データは無効 TDRE は以下の条件で 1 にセットされます <ul style="list-style-type: none"> <li>• SITDR から SITSR にデータを転送したとき</li> <li>• シリアルコントロールレジスタ (SICTR) の TE ビットを 0 にクリアしたとき</li> <li>• プロセッサがリセット状態となったとき</li> </ul>

- ・ ビット 0 : 受信データレジスタフル (RDRF)  
SIRDR の受信データが待ち状態であることを示すフラグです。

ビット 0	説 明
RDRF	
0	SIRDR の受信データは無効 RDRF は以下の条件で 0 にクリアされます <ul style="list-style-type: none"> <li>• DMAC が SIRDR からデータを読み出すとき</li> <li>• RDRF から 1 を読み出し、0 を書き込むとき</li> <li>• シリアルコントロールレジスタ (SICTR) の RE ビットを 0 にクリアしたとき</li> <li>• プロセッサがリセット状態となったとき</li> </ul> (初期値)
1	SIRDR の受信データは有効 シリアルデータの受信が正常に終了し SIRSR から SIRDR に転送されると RDRF は 1 にセットされます

## 15.3 動作

### 15.3.1 入力

図 15.2 にインターバル転送モード (SICTR の SE は 1) を示します。

図 15.3 に連続転送モード (SICTR の SE は 0) を示します。

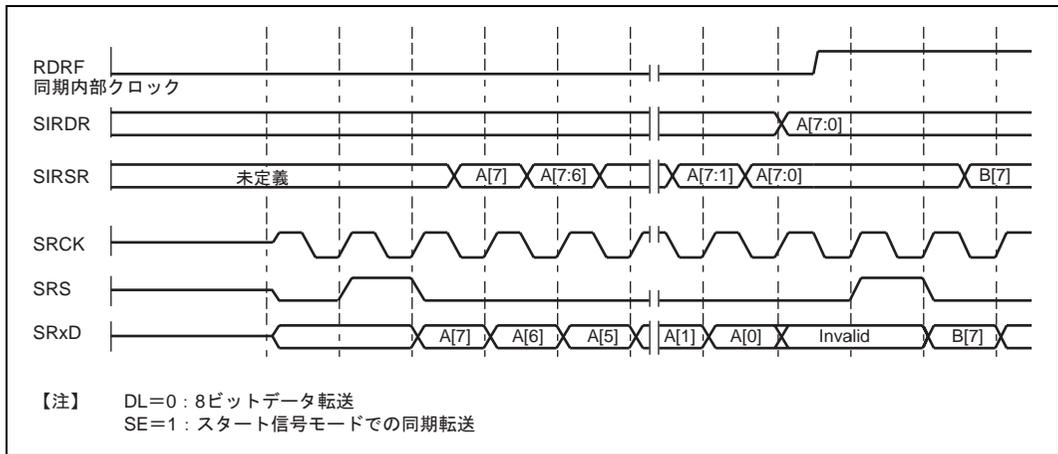


図 15.2 受信：インターバル転送モード

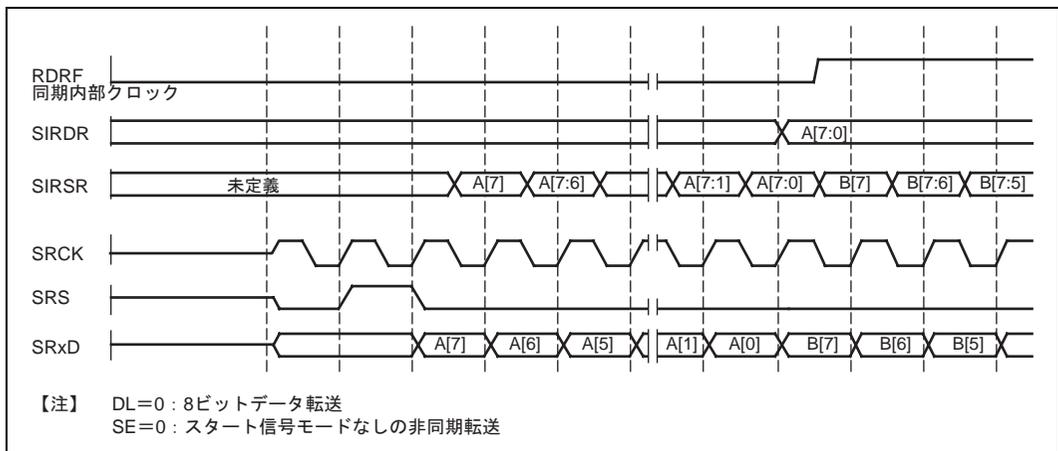


図 15.3 受信：連続転送モード

## 15.3.2 出力

図 15.4 に SICTR の TM を 0 に設定した場合のインターバル転送モード (SICTR の SE は 1) を示します。

図 15.5 に SICTR の TM を 0 に設定した場合の連続転送モード (SICTR の SE は 0) を示します。

図 15.6 に SICTR の TM を 1 に設定した場合のインターバル転送モード (SICTR の SE は 1) を示します。

図 15.7 に SICTR の TM を 1 に設定した場合の連続転送モード (SICTR の SE は 0) を示します。

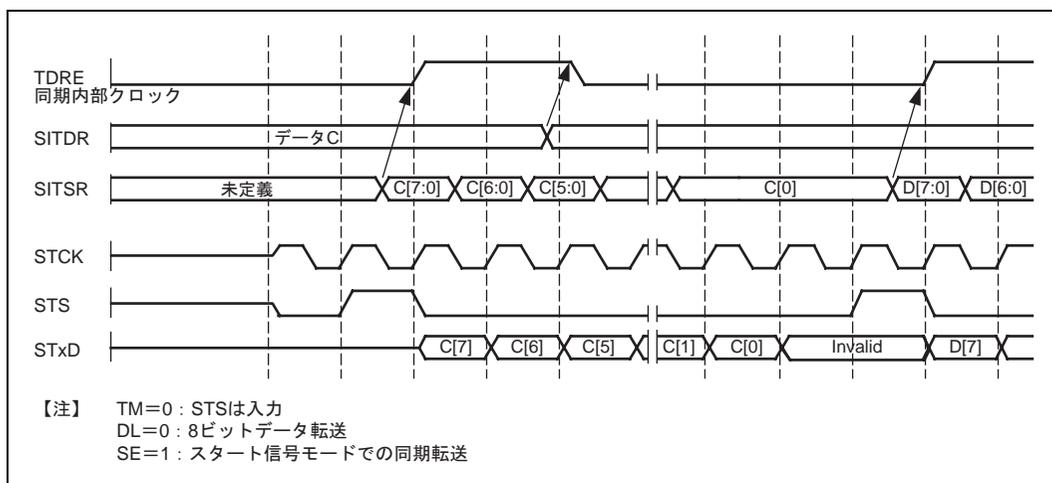


図 15.4 送信 : TM = 0 モードでのインターバル転送モード

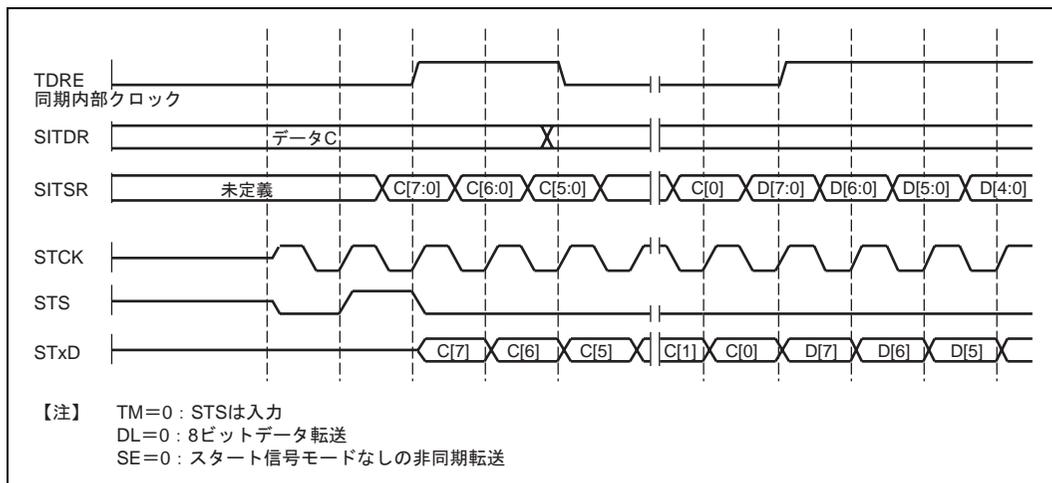


図 15.5 送信 : TM = 0 モードでの連続転送モード

15. シリアルI/O (SIO)

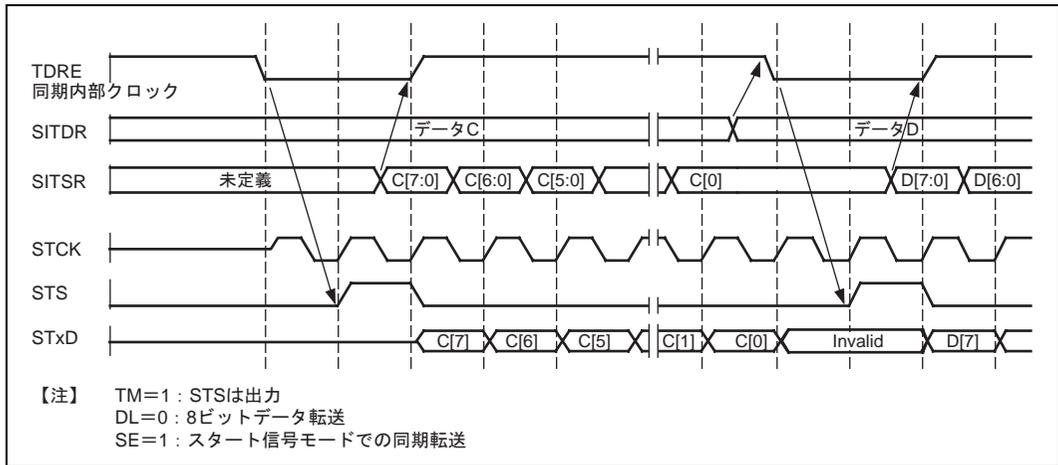


図 15.6 送信 : TM = 1 モードでのインターバル転送モード

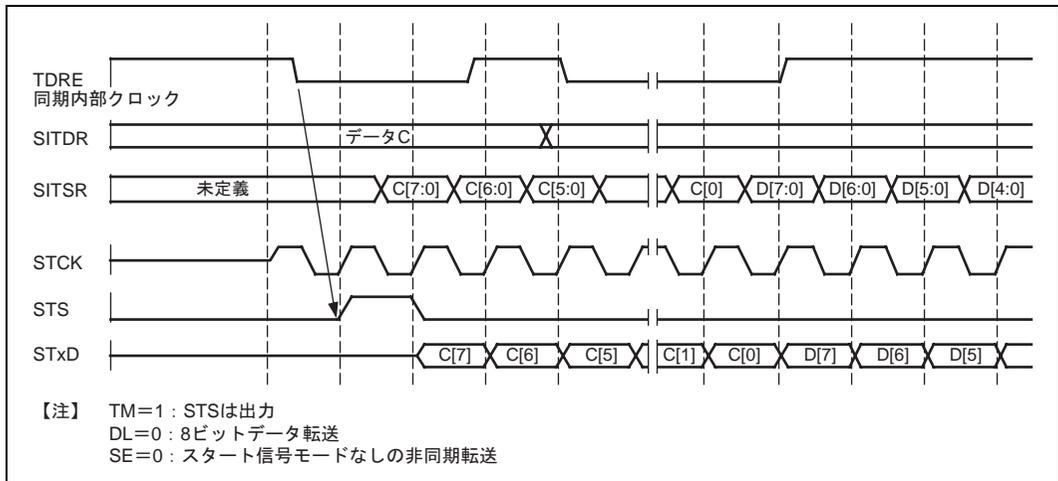


図 15.7 送信 : TM = 1 での連続転送モード

## 15.4 SIO の割り込み要因と DMAC

SIO の各チャンネルには、受信オーバーランエラー (RERI)、送信アンダランエラー (TERI)、受信データフル (RDFI)、および送信データエンプティ (TDEI) の 4 種類の割り込み要因があります。表 15.3 に各割り込み要因とその優先順位を示します。RDFI 割り込みと TDEI 割り込みは、それぞれ SICTR の RIE、TIE ビットで許可します。RERI、TERI 割り込み要求は禁止できません。

RDFI 割り込み要求は SISTR の RDRF ビットを 1 にセットすると発生します。RDFI は汎用 DMA コントローラ (DMAC) を起動して、SIRDR データを読み出すことができます。RDRF は、DMAC が SIRDR のデータを読み出すとき自動的に 0 にクリアされます。

TDEI 割り込み要求は、SISTR の TDRE ビットが 1 にセットされると発生します。TDEI は DMAC を起動して、SITDR に次のデータを書き込むことができます。DMAC が SITDR にデータを書き込むとき TDRE は自動的に 0 にクリアされます。

TDEI 割り込み要求や RDFI 割り込み要求に対する処理を DMAC で行い、割り込みコントローラで処理を行わない場合は、SIO からの割り込み優先順位を低くして、割り込みコントローラが動作しないようにしてください。

SISTR の RERR ビットが 1 にセットされると RERI 割り込み要求が発生します。

SISTR の TERR ビットが 1 にセットされると TERI 割り込み要求が発生します。

チャンネルの割り込み優先順位は、「5. 割り込みコントローラ (INTC)」に示すように IRPE レジスタによって設定されます。

表 15.3 SIO 割り込み要因

割り込み要因	内 容	DMACの起動	優先順位
RERI	受信オーバーランエラー (RERR)	不可	高  低
TERI	送信アンダランエラー (TERR)	不可	
RDFI	受信データフル (RDRF)	可	
TDEI	送信データエンプティ (TDRE)	可	



---

## 16. 16 ビットタイマパルスユニット (TPU)

---

### 16.1 概要

本 LSI は、3 チャンネルの 16 ビットタイマにより構成される 16 ビットタイマパルスユニット (TPU) を内蔵しています。

#### 16.1.1 特長

- 最大 8 本のパルス入出力が可能
- チャンネル 0 は 4 本、チャンネル 1、2 は各 2 本、合計 8 本のタイマジェネラルレジスタ (TGR) を内蔵  
各レジスタ独立にアウトプットコンペア/インプットキャプチャレジスタの設定が可能  
チャンネル 0 の TGRC、TGRD は、バッファレジスタとして使用可能
- 各チャンネルとも 7 ~ 8 種類のカウント入力クロックを選択可能
- 各チャンネルとも次の動作を設定可能
  - コンペアマッチによる波形出力 : 0 出力、1 出力、トグル出力が選択可能
  - インプットキャプチャ機能 : 立ち上がりエッジ、立ち下がりエッジ、両エッジ検出が選択可能
  - カウンタクリア動作 : コンペアマッチ、インプットキャプチャによるカウンタクリアが可能
  - 同期動作 : 複数のタイマカウンタ (TCNT) への同時書き込みが可能  
コンペアマッチ/インプットキャプチャによる同時クリアが可能  
カウンタの同期動作による各レジスタの同期入出力が可能
  - PWM モード : 任意デューティの PWM 出力が可能、同期動作と組み合わせることにより、最大 7 相の PWM 出力が可能
- チャンネル 0 はバッファ動作を設定可能
  - インプットキャプチャレジスタのダブルバッファ構成が可能
  - アウトプットコンペアレジスタの自動書き換えが可能
- チャンネル 1、2 は各々独立に位相計数モードを設定可能  
2 相エンコーダパルスのアップダウンカウントが可能
- 内部 16 ビットバスによる高速アクセス  
16 ビットバスインタフェースによる高速アクセスが可能
- 13 種類の割り込み要因
  - チャンネル 0 はコンペアマッチ/インプットキャプチャ兼用割り込み × 4 本、  
オーバーフロー割り込み × 1 本が独立に要求可能
  - チャンネル 1、2 はコンペアマッチ/インプットキャプチャ兼用割り込み × 2 本、  
オーバーフロー割り込み × 1 本、アンダフロー割り込み × 1 本が独立に要求可能

16. 16ビットタイマパルスユニット (TPU)

- レジスタデータの自動転送が可能

ダイレクトメモリアクセスコントローラ (DMAC) の起動により、ブロック転送、1ワードデータ転送および1バイトデータ転送が可能

表 16.1 に TPU の機能一覧を示します。

表 16.1 TPU 機能一覧 (1)

項 目	チャンネル 0	チャンネル 1	チャンネル 2
カウントクロック	P / 1 P / 4 P / 16 P / 64 TCLKA TCLKB TCLKC TCLKD	P / 1 P / 4 P / 16 P / 64 P / 256 TCLKAT CLKB	P / 1 P / 4 P / 16 P / 64 P / 1024 TCLKA TCLKB TCLKC
ジェネラルレジスタ	TGR0A TGR0B	TGR1A TGR1B	TGR2A TGR2B
ジェネラルレジスタ / バッファレジスタ	TGR0C TGR0D		
入出力端子	TIOCA0 TIOCB0 TIOCC0 TIOCD0	TIOCA1 TIOCB1	TIOCA2 TIOCB2
カウンタクリア機能	TGR のコンペアマッチまたは インプットキャプチャ	TGR のコンペアマッチまたは インプットキャプチャ	TGR のコンペアマッチまたは インプットキャプチャ
コンペア マッチ 出力	0 出力		
	1 出力		
	トグル出力		
インプットキャプチャ機能			
同期動作			
PWM モード			
位相計数モード			
バッファ動作			

【注】 : 可能  
: 不可

表 16.1 TPU 機能一覧 (2)

項 目	チャンネル 0	チャンネル 1	チャンネル 2
DMAC の起動	TGR のコンペアマッチまたは インプットキャプチャ	-	-
割り込み要因	5 要因 <ul style="list-style-type: none"> <li>・コンペアマッチ   / インプットキャプチャ 0A</li> <li>・コンペアマッチ   / インプットキャプチャ 0B</li> <li>・コンペアマッチ   / インプットキャプチャ 0C</li> <li>・コンペアマッチ   / インプットキャプチャ 0D</li> <li>・オーバフロー</li> </ul>	4 要因 <ul style="list-style-type: none"> <li>・コンペアマッチ   / インプットキャプチャ 1A</li> <li>・コンペアマッチ   / インプットキャプチャ 1B</li> <li>・オーバフロー</li> <li>・アンダフロー</li> </ul>	4 要因 <ul style="list-style-type: none"> <li>・コンペアマッチ   / インプットキャプチャ 2A</li> <li>・コンペアマッチ   / インプットキャプチャ 2B</li> <li>・オーバフロー</li> <li>・アンダフロー</li> </ul>

## 16. 16ビットタイマパルスユニット (TPU)

### 16.1.2 ブロック図

TPUのブロック図を図16.1に示します。

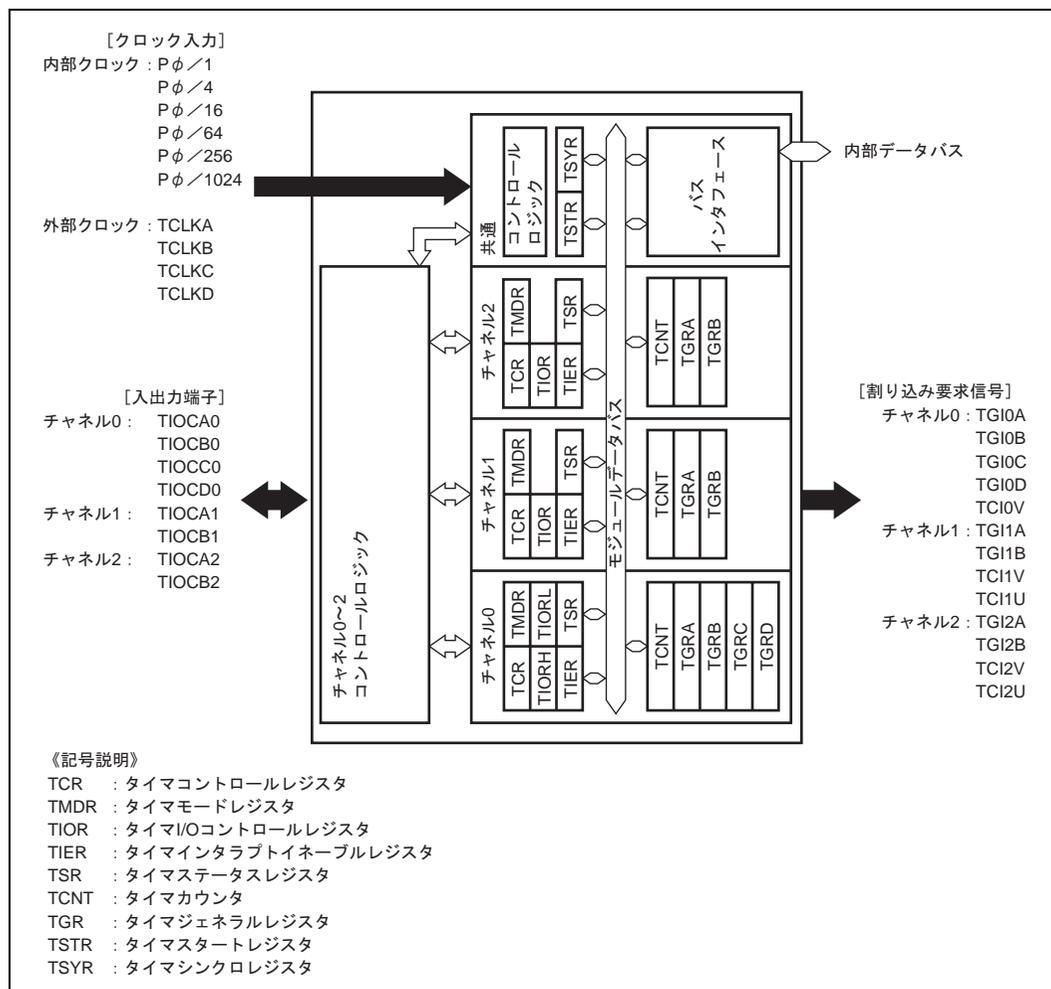


図 16.1 TPUのブロック図

## 16.1.3 端子構成

TPU の端子構成を表 16.2 に示します。

表 16.2 端子構成

チャンネル	名称	記号	入出力	機能
共通	クロック入力 A	TCLKA	入力	外部クロック A 入力端子 (チャンネル 1 の位相計数モード A 相入力)
	クロック入力 B	TCLKB	入力	外部クロック B 入力端子 (チャンネル 1 の位相計数モード B 相入力)
	クロック入力 C	TCLKC	入力	外部クロック C 入力端子 (チャンネル 2 の位相計数モード A 相入力)
	クロック入力 D	TCLKD	入力	外部クロック D 入力端子 (チャンネル 2 の位相計数モード B 相入力)
0	インプットキャプチャ / アウトコンペアマッチ A0	TIOCA0	入出力	TGR0A のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
	インプットキャプチャ / アウトコンペアマッチ B0	TIOCB0	入出力	TGR0B のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
	インプットキャプチャ / アウトコンペアマッチ C0	TIOCC0	入出力	TGR0C のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
	インプットキャプチャ / アウトコンペアマッチ D0	TIOCD0	入出力	TGR0D のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
1	インプットキャプチャ / アウトコンペアマッチ A1	TIOCA1	入出力	TGR1A のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
	インプットキャプチャ / アウトコンペアマッチ B1	TIOCB1	入出力	TGR1B のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
2	インプットキャプチャ / アウトコンペアマッチ A2	TIOCA2	入出力	TGR2A のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子
	インプットキャプチャ / アウトコンペアマッチ B2	TIOCB2	入出力	TGR2B のインプットキャプチャ入力 / アウトプットコンペア出力 / PWM 出力端子

## 16.1.4 レジスタ構成

TPUのレジスタ構成を表16.3に示します。

表 16.3 レジスタ構成

チャンネル	名称	略称	R/W	初期値	アドレス	アクセスサイズ (ビット)
0	タイマコントロールレジスタ 0	TCR0	R/W	H'00	H'FFFFFFC50	8、16
	タイマモードレジスタ 0	TMDR0	R/W	H'C0	H'FFFFFFC51	8、16
	タイマ I/O コントロールレジスタ 0H	TIOR0H	R/W	H'00	H'FFFFFFC52	8、16
	タイマ I/O コントロールレジスタ 0L	TIOR0L	R/W	H'00	H'FFFFFFC53	8、16
	タイマインタラプトイネーブルレジスタ 0	TIER0	R/W	H'40	H'FFFFFFC54	8、16
	タイマステータスレジスタ 0	TSR0	R/(W) *	H'C0	H'FFFFFFC55	8、16
	タイマカウンタ 0	TCNT0	R/W	H'0000	H'FFFFFFC56	16
	タイマジェネラルレジスタ 0A	TGR0A	R/W	H'FFFF	H'FFFFFFC58	16
	タイマジェネラルレジスタ 0B	TGR0B	R/W	H'FFFF	H'FFFFFFC5A	16
	タイマジェネラルレジスタ 0C	TGR0C	R/W	H'FFFF	H'FFFFFFC5C	16
タイマジェネラルレジスタ 0D	TGR0D	R/W	H'FFFF	H'FFFFFFC5E	16	
1	タイマコントロールレジスタ 1	TCR1	R/W	H'00	H'FFFFFFC60	8、16
	タイマモードレジスタ 1	TMDR1	R/W	H'C0	H'FFFFFFC61	8、16
	タイマ I/O コントロールレジスタ 1	TIOR1	R/W	H'00	H'FFFFFFC62	8、16
	タイマインタラプトイネーブルレジスタ 1	TIER1	R/W	H'40	H'FFFFFFC64	8、16
	タイマステータスレジスタ 1	TSR1	R/(W) *	H'C0	H'FFFFFFC65	8、16
	タイマカウンタ 1	TCNT1	R/W	H'0000	H'FFFFFFC66	16
	タイマジェネラルレジスタ 1A	TGR1A	R/W	H'FFFF	H'FFFFFFC68	16
	タイマジェネラルレジスタ 1B	TGR1B	R/W	H'FFFF	H'FFFFFFC6A	16
2	タイマコントロールレジスタ 2	TCR2	R/W	H'00	H'FFFFFFC70	8、16
	タイマモードレジスタ 2	TMDR2	R/W	H'C0	H'FFFFFFC71	8、16
	タイマ I/O コントロールレジスタ 2	TIOR2	R/W	H'00	H'FFFFFFC72	8、16
	タイマインタラプトイネーブルレジスタ 2	TIER2	R/W	H'40	H'FFFFFFC74	8、16
	タイマステータスレジスタ 2	TSR2	R/(W)*	H'C0	H'FFFFFFC75	8、16
	タイマカウンタ 2	TCNT2	R/W	H'0000	H'FFFFFFC76	16
	タイマジェネラルレジスタ 2A	TGR2A	R/W	H'FFFF	H'FFFFFFC78	16
	タイマジェネラルレジスタ 2B	TGR2B	R/W	H'FFFF	H'FFFFFFC7A	16
共通	タイマスタートレジスタ	TSTR	R/W	H'00	H'FFFFFFC40	8、16
	タイマシンクロレジスタ	TSYR	R/W	H'00	H'FFFFFFC41	8、16

【注】 \* フラグをクリアするための0ライトのみ可能です。

## 16.2 各レジスタの説明

### 16.2.1 タイマコントロールレジスタ (TCR)

チャンネル0 : TCR0

ビット :	7	6	5	4	3	2	1	0
	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							

チャンネル1 : TCR1

チャンネル2 : TCR2

ビット :	7	6	5	4	3	2	1	0
	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R/W						

TCR は各チャンネルの TCNT を制御する 8 ビットのレジスタです。TPU には、チャンネル 0 ~ 2 に各 1 本、計 3 本の TCR があります。TCR は、リセット時に H'00 に初期化されます。

TCR の設定は、TCNT の動作が停止した状態で行ってください。

16. 16 ビットタイムパルスユニット (TPU)

- ビット7、6、5：カウンタクリア2、1、0 (CCLR2、CCLR1、CCLR0)  
TCNT のカウンタクリア要因を選択します。

チャンネル	ビット7	ビット6	ビット5	説明
	CCLR2	CCLR1	CCLR0	
0	0	0	0	TCNT のクリア禁止 (初期値)
			1	TGRA のコンペアマッチ / インพุットキャプチャで TCNT クリア
		1	0	TGRB のコンペアマッチ / インพุットキャプチャで TCNT クリア
			1	同期クリア / 同期動作をしている他のチャンネルのカウンタクリアで TCNT をクリア *1
	1	0	0	TCNT のクリア禁止
			1	TGRC のコンペアマッチ / インพุットキャプチャで TCNT クリア *2
		1	0	TGRD のコンペアマッチ / インพุットキャプチャで TCNT クリア *2
			1	同期クリア / 同期動作をしている他のチャンネルのカウンタクリアで TCNT をクリア *1

チャンネル	ビット7	ビット6	ビット5	説明
	リザーブ *3	CCLR1	CCLR0	
1, 2	0	0	0	TCNT のクリア禁止 (初期値)
			1	TGRA のコンペアマッチ / インพุットキャプチャで TCNT クリア
		1	0	TGRB のコンペアマッチ / インพุットキャプチャで TCNT クリア
			1	同期クリア / 同期動作をしている他のチャンネルのカウンタクリアで TCNT をクリア *1

- 【注】 \*1 同期動作の設定は、TSYR の SYNC ビットを 1 にセットすることにより行います。  
 \*2 TGRC または TGRD をバッファレジスタとして使用している場合は、バッファレジスタの設定が優先され、コンペアマッチ / インพุットキャプチャが発生しないため、TCNT はクリアされません。  
 \*3 チャンネル 1、2 ではビット 7 は予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

- ビット4、3：クロックエッジ 1、0 (CKEG1、CKEG0)

入力クロックのエッジを選択します。両エッジカウントを選択すると、入力クロックに対する 1 / 2 周期クロックの選択が可能になります (例:  $P / 4$  の両エッジ =  $P / 2$  の立ち上がりエッジ)。チャンネル 1、2 で位相計数モードを使用する場合は、本設定は無視され、位相計数モードの設定が優先されます。

ビット4	ビット3	説明
CKEG1	CKEG0	
0	0	立ち上がりエッジでカウント (初期値)
	1	立ち下がりエッジでカウント
1		両エッジでカウント

- 【注】 内部クロックのエッジ選択は、入力クロックが  $P / 4$  もしくはそれより遅い場合に有効です。入力クロックに  $P / 1$  を選択した場合は本設定は無視され、立ち上がりエッジカウント選択になります。

- ビット2、1、0：タイマプリスケラ 2、1、0 (TPSC2 ~ TPSC0)

TCNT のカウンタクロックを選択します。各チャンネル独立にクロックソースを選択することができます。表 16.4 に各チャンネルごとに設定可能なクロックソース一覧を示します。

表 16.4 TPU のクロックソース一覧

チャンネル	内部クロック						外部クロック			
	P / 1	P / 4	P / 16	P / 64	P / 256	P / 1024	TCLKA	TCLKB	TCLKC	TCLKD
0										
1										
2										

【注】 : 設定あり  
空欄 : 設定なし

チャンネル	ビット 2	ビット 1	ビット 0	説 明
	TPSC2	TPSC1	TPSC0	
0	0	0	0	内部クロック : P / 1 でカウント (初期値)
			1	内部クロック : P / 4 でカウント
		1	0	内部クロック : P / 16 でカウント
			1	内部クロック : P / 64 でカウント
	1	0	0	外部クロック : TCLKA 端子入力でカウント
			1	外部クロック : TCLKB 端子入力でカウント
		1	0	外部クロック : TCLKC 端子入力でカウント
			1	外部クロック : TCLKD 端子入力でカウント

チャンネル	ビット 2	ビット 1	ビット 0	説 明
	TPSC2	TPSC1	TPSC0	
1	0	0	0	内部クロック : P / 1 でカウント (初期値)
			1	内部クロック : P / 4 でカウント
		1	0	内部クロック : P / 16 でカウント
			1	内部クロック : P / 64 でカウント
	1	0	0	外部クロック : TCLKA 端子入力でカウント
			1	外部クロック : TCLKB 端子入力でカウント
		1	0	内部クロック : P / 256 でカウント
			1	設定禁止

【注】 チャンネル 1 が位相計数モード時、この設定は無効になります。

チャンネル	ビット 2	ビット 1	ビット 0	説 明
	TPSC2	TPSC1	TPSC0	
2	0	0	0	内部クロック : P / 1 でカウント (初期値)
			1	内部クロック : P / 4 でカウント
		1	0	内部クロック : P / 16 でカウント
			1	内部クロック : P / 64 でカウント
	1	0	0	外部クロック : TCLKA 端子入力でカウント
			1	外部クロック : TCLKB 端子入力でカウント
		1	0	外部クロック : TCLKC 端子入力でカウント
			1	内部クロック : P / 1024 でカウント

【注】 チャンネル 2 が位相計数モード時、この設定は無効になります。

## 16.2.2 タイマモードレジスタ (TMDR)

チャンネル0 : TMDR0

ビット :	7	6	5	4	3	2	1	0
	—	—	BFB	BFA	MD3	MD2	MD1	MD0
初期値 :	1	1	0	0	0	0	0	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W

チャンネル1 : TMDR1

チャンネル2 : TMDR2

ビット :	7	6	5	4	3	2	1	0
	—	—	—	—	MD3	MD2	MD1	MD0
初期値 :	1	1	0	0	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W

TMDR は 8 ビットのリード/ライト可能なレジスタで、各チャンネルの動作モードの設定を行います。TPU には、各チャンネル 1 本、計 3 本の TMDR があります。TMDR は、リセット時に H'CO に初期化されます。

TMDR の設定は、TCNT の動作が停止した状態で行ってください。

- ビット 7、6 : 予約ビット

読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。

- ビット 5 : バッファ動作 B (BFB)

TGRB を通常動作させるか、TGRB と TGRD を組み合わせてバッファ動作させるかを設定します。TGRD をバッファレジスタとして使用した場合は、TGRD のインプットキャプチャ/アウトプットコンペアは発生しません。

TGRD を持たないチャンネル 1、2 ではこのビットは予約ビットになります。読み出すと常に 0 が読み出されません。書き込む値も常に 0 にしてください。

ビット 5	説明
BFB	
0	TGRB は通常動作 (初期値)
1	TGRB と TGRD はバッファ動作

- ビット 4 : バッファ動作 A (BFA)

TGRA を通常動作させるか、TGRA と TGRC を組み合わせてバッファ動作させるかを設定します。TGRC をバッファレジスタとして使用した場合は、TGRC のインプットキャプチャ/アウトプットコンペアは発生しません。

TGRC を持たないチャンネル 1、2 ではこのビットは予約ビットになります。読み出すと常に 0 が読み出されません。書き込む値も常に 0 にしてください。

ビット 4	説明
BFA	
0	TGRA は通常動作 (初期値)
1	TGRA と TGRC はバッファ動作

- ・ ビット 3 ~ 0 : モード 3 ~ 0 (MD3 ~ MD0)  
MD3 ~ MD0 はタイマの動作モードを設定します。

ビット 3	ビット 2	ビット 1	ビット 0	説明
MD3* <sup>1</sup>	MD2* <sup>2</sup>	MD1	MD0	
0	0	0	0	通常動作 (初期値)
			1	リザーブ
		1	0	PWM モード 1
			1	PWM モード 2
	1	0	0	位相計数モード 1
			1	位相計数モード 2
		1	0	位相計数モード 3
			1	位相計数モード 4
1	*	*	*	

【注】 \* : Don't care

\*1 MD3 は予約ビットです。書き込み時には常に 0 を書き込んでください。

\*2 チャンネル 0 では、位相計数モードの設定はできません。

MD2 には常に 0 を書き込んでください。

## 16. 16 ビットタイマパルスユニット (TPU)

---

### 16.2.3 タイマ I/O コントロールレジスタ (TIOR)

チャンネル0 : TIOR0H

チャンネル1 : TIOR1

チャンネル2 : TIOR2

ビット :	7	6	5	4	3	2	1	0
	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							

チャンネル0 : TIOR0L

ビット :	7	6	5	4	3	2	1	0
	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W							

【注】 TGRC、あるいはTGRDをバッファ動作に設定した場合は、本設定は無効となり、バッファレジスタとして動作します。

TIOR は TGR を制御する 8 ビットのレジスタです。TPU には、チャンネル 0 に 2 本、チャンネル 1、2 に各 1 本、計 4 本の TIOR があります。TIOR はリセット時に H'00 に初期化されます。

TIOR は TMDR の設定により影響を受けますので注意してください。

TIOR で指定した初期出力は、カウンタが停止した (TSTR の CST ビットを 0 にクリアした) 状態で有効になります。また、PWM モード 2 の場合にはカウンタが 0 にクリアされた時点での出力を指定します。

- ・ ビット 7 ~ 4 : I/O コントロール B3 ~ 0 (IOB3 ~ IOB0)  
I/O コントロール D3 ~ 0 (IOD3 ~ IOD0)  
IOB3 ~ IOB0 ビットは TGRB の機能を設定します。  
IOD3 ~ IOD0 ビットは TGRD の機能を設定します。

## ・ TIOR0H

チャンネル	ビット7	ビット6	ビット5	ビット4	説明			
	IOB3	IOB2	IOB1	IOB0				
0	0	0	0	0	TGR0B は アウトプット コンペア レジスタ	出力禁止 (初期値)		
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
			1	0			コンペアマッチで 1 出力	
				1			コンペアマッチでトグル出力	
			1	0			0	出力禁止
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	1	0	コンペアマッチで 1 出力					
		1	コンペアマッチでトグル出力					
	1	0	0	0	TGR0B は インプット キャプチャ レジスタ	キャプチャ 入力元 は TIOCB0 端子	立ち上がりエッジでインプットキャプ チャ	
				1			立ち下がりエッジでインプットキャプ チャ	
				*			両エッジでインプットキャプチャ	
		1	*	*		設定禁止		

【注】 \* : Don't care

## ・ TIOR0L

チャンネル	ビット7	ビット6	ビット5	ビット4	説明			
	IOD3	IOD2	IOD1	IOD0				
0	0	0	0	0	TGR0D は アウトプット コンペア レジスタ* <sup>1</sup>	出力禁止 (初期値)		
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
			1	0			コンペアマッチで 1 出力	
				1			コンペアマッチでトグル出力	
			1	0			0	出力禁止
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	1	0	コンペアマッチで 1 出力					
		1	コンペアマッチでトグル出力					
	1	0	0	0	TGR0D は インプット キャプチャ レジスタ* <sup>1</sup>	キャプチャ 入力元 は TIOCD0 端子	立ち上がりエッジでインプットキャプ チャ	
				1			立ち下がりエッジでインプットキャプ チャ	
				*			両エッジでインプットキャプチャ	
		1	*	*		設定禁止		

【注】 \* : Don't care

\*<sup>1</sup> TMDR0 の BFB ビットを 1 にセットして TGR0D をバッファレジスタとして使用した場合は、本設定は無効になり、インプットキャプチャ/アウトプットコンペアは発生しません。

16. 16 ビットタイマパルスユニット (TPU)

・ TIOR1

チャンネル	ビット7	ビット6	ビット5	ビット4	説明				
	IOB3	IOB2	IOB1	IOB0					
1	0	0	0	0	TGR1B は アウトプット コンペア レジスタ	出力禁止 (初期値)			
				1		初期出力は 0 出力	コンペアマッチで 0 出力		
			1	0			コンペアマッチで 1 出力		
				1			コンペアマッチでトグル出力		
			1	0			0	出力禁止	
						1	初期出力は 1 出力	コンペアマッチで 0 出力	
	1	0	コンペアマッチで 1 出力						
		1	コンペアマッチでトグル出力						
	1	0	0	0	TGR1B は インプット キャプチャ レジスタ	キャプチャ 入力元 は TIOCB1 端子	立ち上がりエッジでインプットキャプチャ		
							1	*	立ち下がりエッジでインプットキャプチャ
				1				*	両エッジでインプットキャプチャ
							1	*	*

【注】 \* : Don't care

・ TIOR2

チャンネル	ビット7	ビット6	ビット5	ビット4	説明				
	IOB3	IOB2	IOB1	IOB0					
2	0	0	0	0	TGR2B は アウトプット コンペア レジスタ	出力禁止 (初期値)			
				1		初期出力は 0 出力	コンペアマッチで 0 出力		
			1	0			コンペアマッチで 1 出力		
				1			コンペアマッチでトグル出力		
			1	0			0	出力禁止	
						1	初期出力は 1 出力	コンペアマッチで 0 出力	
	1	0	コンペアマッチで 1 出力						
		1	コンペアマッチでトグル出力						
	1	*	0	0	TGR2B は インプット キャプチャレ ジスタ	キャプチャ 入力元 は TIOCB2 端子	立ち上がりエッジでインプットキャプチャ		
							1	*	立ち下がりエッジでインプットキャプチャ
				1				*	両エッジでインプットキャプチャ
							1	*	*

【注】 \* : Don't care

- ビット 3 ~ 0 : I/O コントロール A3 ~ 0 (IOA3 ~ IOA0)  
I/O コントロール C3 ~ 0 (IOC3 ~ IOC0)  
IOA3 ~ IOA0 は TGRA の機能を設定します。  
IOC3 ~ IOC0 は TGRC の機能を設定します。

・ TIOR0H

チャンネル	ビット 3	ビット 2	ビット 1	ビット 0	説 明			
	IOA3	IOA2	IOA1	IOA0				
0	0	0	0	0	TGR0A は アウトプット コンペア レジスタ	出力禁止	(初期値)	
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
				0			コンペアマッチで 1 出力	
			1	コンペアマッチでトグル出力				
			1	0		0	出力禁止	
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	0	コンペアマッチで 1 出力						
	1	コンペアマッチでトグル出力						
	1	0	0	0	TGR0A は インプット キャブチャ レジスタ	キャブチャ 入力元 は TIOCA0 端子	立ち上がりエッジでインプットキャブ チャ	
				1			立ち下がりエッジでインプットキャブ チャ	
				1			両エッジでインプットキャブチャ	
		1	*	*			設定禁止	

【注】 \* : Don't care

・ TIOR0L

チャンネル	ビット 3	ビット 2	ビット 1	ビット 0	説 明			
	IOC3	IOC2	IOC1	IOC0				
0	0	0	0	0	TGR0C は アウトプット コンペア レジスタ *	出力禁止	(初期値)	
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
				0			コンペアマッチで 1 出力	
			1	コンペアマッチでトグル出力				
			1	0		0	出力禁止	
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	0	コンペアマッチで 1 出力						
	1	コンペアマッチでトグル出力						
	1	0	0	0	TGR0C は インプット キャブチャ レジスタ *	キャブチャ 入力元 は TIOCC0 端子	立ち上がりエッジでインプットキャブ チャ	
				1			立ち下がりエッジでインプットキャブ チャ	
				1			両エッジでインプットキャブチャ	
		1	*	*			設定禁止	

【注】 \* : Don't care

\* TMDR0 の BFA ビットを 1 にセットして TGR0C をバッファレジスタとして使用した場合は、本設定は無効になり、インプットキャブチャ / アウトプットコンペアは発生しません。

16. 16 ビットタイマパルスユニット (TPU)

・ TIOR1

チャンネル	ビット3	ビット2	ビット1	ビット0	説明			
	IOA3	IOA2	IOA1	IOA0				
1	0	0	0	0	TGR1A は アウトプット コンペア レジスタ	出力禁止 (初期値)		
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
			1	0			コンペアマッチで 1 出力	
				1			コンペアマッチでトグル出力	
			1	0			0	出力禁止
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	1	0	コンペアマッチで 1 出力					
		1	コンペアマッチでトグル出力					
	1	0	0	0	TGR1A は インプット キャプチャ レジスタ	キャプチャ 入力元 は TIOCA1 端子	立ち上がりエッジでインプットキャプチャ	
				1			立ち下がりエッジでインプットキャプチャ	
			1	*			*	両エッジでインプットキャプチャ
		1	*	*	*	設定禁止		

【注】 \* : Don't care

・ TIOR2

チャンネル	ビット3	ビット2	ビット1	ビット0	説明			
	IOA3	IOA2	IOA1	IOA0				
2	0	0	0	0	TGR2A は アウトプット コンペア レジスタ	出力禁止 (初期値)		
				1		初期出力は 0 出力	コンペアマッチで 0 出力	
			1	0			コンペアマッチで 1 出力	
				1			コンペアマッチでトグル出力	
			1	0			0	出力禁止
						1	初期出力は 1 出力	コンペアマッチで 0 出力
	1	0	コンペアマッチで 1 出力					
		1	コンペアマッチでトグル出力					
	1	*	0	0	TGR2A は インプット キャプチャレ ジスタ	キャプチャ 入力元 は TIOCA2 端子	立ち上がりエッジでインプットキャプチャ	
				1			立ち下がりエッジでインプットキャプチャ	
			1	*			両エッジでインプットキャプチャ	

【注】 \* : Don't care

## 16.2.4 タイマインタラプトイネーブルレジスタ (TIER)

チャンネル0 : TIER0

ビット :	7	6	5	4	3	2	1	0
	—	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
初期値 :	0	1	0	0	0	0	0	0
R/W :	R	R	R	R/W	R/W	R/W	R/W	R/W

チャンネル1 : TIER1

チャンネル2 : TIER2

ビット :	7	6	5	4	3	2	1	0
	—	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
初期値 :	0	1	0	0	0	0	0	0
R/W :	R	R	R/W	R/W	R	R	R/W	R/W

TIER は 8 ビットのレジスタで、各チャンネルの割り込み要求の許可、禁止を制御します。TPU には、各チャンネル 1 本、計 3 本の TIER があります。TIER は、リセット時に H'40 に初期化されます。

## ・ ビット 7 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

## ・ ビット 6 : 予約ビット

読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。

## ・ ビット 5 : アンダフローインタラプトイネーブル (TCIEU)

チャンネル 1、2 で TSR の TCFU フラグが 1 にセットされたとき、TCFU フラグによる割り込み要求 (TCIU) を許可または禁止します。

チャンネル 0 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

ビット 5	
TCIEU	説 明
0	TCFU による割り込み要求 (TCIU) を禁止 (初期値)
1	TCFU による割り込み要求 (TCIU) を許可

## ・ ビット 4 : オーバフローインタラプトイネーブル (TCIEV)

TSR の TCFV フラグが 1 にセットされたとき、TCFV フラグによる割り込み要求 (TCIV) を許可または禁止します。

16. 16 ビットタイマパルスユニット (TPU)

ビット 4	説 明
TCIEV	
0	TCFV による割り込み要求 (TCIV) を禁止 (初期値)
1	TCFV による割り込み要求 (TCIV) を許可

・ ビット 3 : TGR インタラプトイネーブル D (TGIED)

チャンネル 0 で TSR の TGFD ビットが 1 にセットされたとき、TGFD ビットによる割り込み要求 (TGID) を許可または禁止します。

チャンネル 1、2 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

ビット 3	説 明
TGIED	
0	TGFD ビットによる割り込み要求 (TGID) を禁止 (初期値)
1	TGFD ビットによる割り込み要求 (TGID) を許可

・ ビット 2 : TGR インタラプトイネーブル C (TGIEC)

チャンネル 0 で TSR の TGFC ビットが 1 にセットされたとき、TGFC ビットによる割り込み要求 (TGIC) を許可または禁止します。

チャンネル 1、2 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

ビット 2	説 明
TGIEC	
0	TGFC ビットによる割り込み要求 (TGIC) を禁止 (初期値)
1	TGFC ビットによる割り込み要求 (TGIC) を許可

・ ビット 1 : TGR インタラプトイネーブル B (TGIEB)

TSR の TGFB ビットが 1 にセットされたとき、TGFB ビットによる割り込み要求 (TGIB) を許可または禁止します。

ビット 1	説 明
TGIEB	
0	TGFB ビットによる割り込み要求 (TGIB) を禁止 (初期値)
1	TGFB ビットによる割り込み要求 (TGIB) を許可

・ ビット 0 : TGR インタラプトイネーブル A (TGIEA)

TSR の TGFA ビットが 1 にセットされたとき、TGFA ビットによる割り込み要求 (TGIA) を許可または禁止します。

ビット 0	
TGIEA	説 明
0	TGFA ビットによる割り込み要求 (TGIA) を禁止 (初期値)
1	TGFA ビットによる割り込み要求 (TGIA) を許可

### 16.2.5 タイマステータスレジスタ (TSR)

チャンネル0 : TSR0

ビット :	7	6	5	4	3	2	1	0
	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
初期値 :	1	1	0	0	0	0	0	0
R/W :	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】\* フラグをクリアするための0ライトのみ可能です。

チャンネル1 : TSR1

チャンネル2 : TSR2

ビット :	7	6	5	4	3	2	1	0
	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA
初期値 :	1	1	0	0	0	0	0	0
R/W :	R	R	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

【注】\* フラグをクリアするための0ライトのみ可能です。

TSR は 8 ビットのレジスタで、各チャンネルのステータスの表示を行います。TPU には、各チャンネル 1 本、計 3 本の TSR があります。TSR は、リセット時に H'00 に初期化されます。

- ・ ビット 7 : カウント方向フラグ (TCFD)  
チャンネル 1、2 の TCNT のカウント方向を示すステータスフラグです。  
チャンネル 0 では予約ビットです。読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。

ビット 7	
TCFD	説 明
0	TCNT はダウンカウント
1	TCNT はアップカウント (初期値)

- ・ ビット 6 : 予約ビット  
読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。
- ・ ビット 5 : アンダフローフラグ (TCFU)  
チャンネル 1、2 が位相計数モードのとき TCNT のアンダフローの発生を示すステータスフラグです。  
チャンネル 0 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

16. 16 ビットタイマパルスユニット (TPU)

ビット 5	説 明
TCFU	
0	[クリア条件] (初期値) TCFU = 1 の状態で TCFU をリード後、TCFU に 0 を書き込んだとき
1	[セット条件] TCNT の値がアンダフロー (H'0000 H'FFFF) したとき

- ・ ビット 4 : オーバフローフラグ (TCFV)  
TCNT のオーバフローの発生を示すステータスフラグです。

ビット 4	説 明
TCFV	
0	[クリア条件] (初期値) TCFV = 1 の状態で TCFV をリード後、TCFV に 0 を書き込んだとき
1	[セット条件] TCNT の値がオーバフロー (H'FFFF H'0000) したとき

- ・ ビット 3 : インพุットキャプチャ / アウトプットコンペアフラグ D (TGFD)  
チャンネル 0 の TGRD のインพุットキャプチャまたはコンペアマッチの発生を示すステータスフラグです。  
チャンネル 1、2 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

ビット 3	説 明
TGFD	
0	[クリア条件] (初期値) (1) TGID 割り込みにより DMAC が起動され、DMAC の DRCR の設定が TGI0D のとき (2) TGFD = 1 の状態で TGFD をリード後、TGFD に 0 をライトしたとき
1	[セット条件] (1) TGRD がアウトプットコンペアレジスタとして機能している場合、 TCNT = TGRD になったとき (2) TGRD がインพุットキャプチャとして機能している場合、 インพุットキャプチャ信号により TCNT の値が TGRD に転送されたとき

- ・ ビット 2 : インพุットキャプチャ / アウトプットコンペアフラグ C (TGFC)  
チャンネル 0 の TGRC のインพุットキャプチャまたはコンペアマッチの発生を示すステータスフラグです。  
チャンネル 1、2 では予約ビットです。読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

ビット2	説 明
TGFC	
0	[クリア条件] (初期値) (1) TGIC 割り込みにより DMAC が起動され、DMAC の DRCR の設定が TGI0C のとき (2) TGFC = 1 の状態で TGFC をリード後、TGFC に 0 をライトしたとき
1	[セット条件] (1) TGRC がアウトプットコンペアレジスタとして機能している場合、 TCNT = TGRC になったとき (2) TGRC がインプットキャプチャとして機能している場合、 インプットキャプチャ信号により TCNT の値が TGRC に転送されたとき

- ・ ビット1: インプットキャプチャ/アウトプットコンペアフラグ B (TGFB)  
TGRB のインプットキャプチャまたはコンペアマッチの発生を示すステータスフラグです。

ビット1	説 明
TGFB	
0	[クリア条件] (初期値) (1) TGIB 割り込みにより DMAC が起動され、DMAC の DRCR の設定が TGI0B のとき (2) TGFB = 1 の状態で TGFB を読み出した後、TGFB に 0 を書き込んだとき
1	[セット条件] (1) TGRB がアウトプットコンペアレジスタとして機能している場合、 TCNT = TGRB になったとき (2) TGRB がインプットキャプチャとして機能している場合、 インプットキャプチャ信号により TCNT の値が TGRB に転送されたとき

- ・ ビット0: インプットキャプチャ/アウトプットコンペアフラグ A (TGFA)  
TGRA のインプットキャプチャまたはコンペアマッチの発生を示すステータスフラグです。

ビット0	説 明
TGFA	
0	[クリア条件] (初期値) (1) TGIA 割り込みにより DMAC が起動され、DMAC の DRCR の設定が TGI0A のとき (2) TGFA = 1 の状態で TGFA を読み出した後、TGFA に 0 を書き込んだとき
1	[セット条件] (1) TGRA がアウトプットコンペアレジスタとして機能している場合、 TCNT = TGRA になったとき (2) TGRA がインプットキャプチャとして機能している場合、 インプットキャプチャ信号により TCNT の値が TGRA に転送されたとき

## 16.2.6 タイマカウンタ (TCNT)

チャンネル0 : TCNT0 (アップカウンタ)  
 チャンネル1 : TCNT1 (アップ/ダウンカウンタ\*)  
 チャンネル2 : TCNT2 (アップ/ダウンカウンタ\*)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W															

【注】 \* 位相計数モード時のみアップ/ダウンカウンタとして使用できます。それ以外の場合はアップカウンタになります。

TCNTは16ビットのカウンタです。各チャンネルに1本、計3本のTCNTがあります。

TCNTは、リセット時にH'0000に初期化されます。

TCNTの8ビット単位でのアクセスは禁止です。常に16ビット単位でアクセスしてください。

## 16.2.7 タイマジェネラルレジスタ (TGR)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初期値 :	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W :	R/W															

TGRは16ビットのアウトプットコンペア/インプットキャプチャ兼用のレジスタです。チャンネル0に4本、チャンネル1、2に各2本、計8本のジェネラルレジスタがあります。チャンネル0のTGRCとTGRDは、バッファレジスタとして動作設定することができます\*。TGRはリセット時にH'FFFFに初期化されます。

TGRの8ビット単位でのアクセスは禁止です。常に16ビット単位でアクセスしてください。

【注】 \* TGRのバッファレジスタの組み合わせは、TGRA - TGRC、TGRB - TGRDになります。

## 16.2.8 タイマスタートレジスタ (TSTR)

ビット :	7	6	5	4	3	2	1	0
	—	—	—	—	—	CST2	CST1	CST0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R/W	R/W	R/W

TSTRは8ビットの読み出し/書き込み可能なレジスタで、チャンネル0～2のTCNTの動作/停止を選択します。TSTRはリセット時にH'00に初期化されます。

TMDRへ動作モードを設定する場合やTCRへTCNTのカウントクロックを設定する場合は、TCNTのカウント動作を停止してから行ってください。

- ・ ビット7～3: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット2～0: カウンタスタート2～0 (CST2～CST0)  
TCNTの動作または停止を選択します。

ビット n	説明
CSTn	
0	TCNTnのカウンタ動作は停止 (初期値)
1	TCNTnはカウンタ動作

【注】 n = 2～0

1. TIOC端子を出力状態で動作中に、CSTビットに0をライトするとカウンタは停止しますが、TIOC端子のアウトプットコンペア出力レベルは保持されます。CSTビットが0の状態ではTIOCへのライトを行うと、設定した初期出力値に端子の出力レベルが更新されます。

### 16.2.9 タイマシンクロレジスタ (TSYR)

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SYNC2	SYNC1	SYNC0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

TSYRは8ビットの読み出し/書き込み可能なレジスタで、チャンネル0～2のTCNTの独立動作または同期動作を選択します。対応するビットを1にセットしたチャンネルが同期動作を行います。

TSYRはリセット時にH'00に初期化されます。

- ・ ビット7～3: 予約ビット  
読み出すと常に0が読み出されます。書き込む値も常に0にしてください。
- ・ ビット2～0: タイマ同期2～0 (SYNC2～SYNC0)  
他のチャンネルとの独立動作または同期動作を選択します。  
同期動作を選択すると、複数のTCNTの同期プリセット<sup>\*1</sup>や、他チャンネルのカウンタクリアによる同期クリア<sup>\*2</sup>が可能となります。

ビット n	説明
SYNCn	
0	TCNTnは独立動作 TCNTのプリセット/クリアは他チャンネルと無関係 (初期値)
1	TCNTnは同期動作 TCNTの同期プリセット/同期クリアが可能

【注】 n = 2～0

1. 同期動作の設定には、最低2チャンネルのSYNCビットを1にセットする必要があります。
2. 同期クリアの設定には、SYNCビットの他にTCRのCCLR2～CCLR0ビットで、TCNTのクリア要因を設定する必要があります。

## 16.3 バスマスタとのインタフェース

### 16.3.1 16 ビットレジスタ

TCNT、TGR は 16 ビットのレジスタです。バスマスタとの間のデータバスは 16 ビット幅なので、16 ビット単位での読み出し / 書き込みが可能です。

8 ビット単位での読み出し / 書き込みはできません。常に 16 ビット単位でアクセスしてください。

16 ビットレジスタのアクセス動作例を図 16.2 に示します。

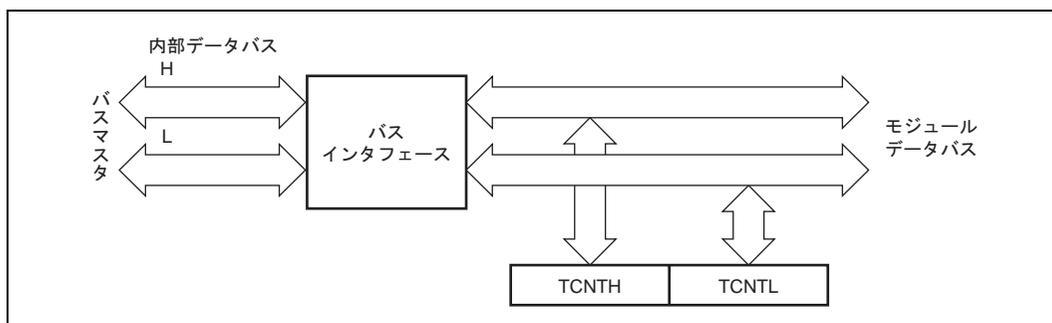


図 16.2 16 ビットレジスタのアクセス動作 [バスマスタ TCNT (16 ビット)]

### 16.3.2 8 ビットレジスタ

TCNT、TGR 以外のレジスタは 8 ビットのレジスタです。CPU との間のデータバスは 16 ビット幅なので、16 ビット単位での読み出し / 書き込みが可能です。また、8 ビット単位での読み出し / 書き込みもできます。

8 ビットレジスタのアクセス動作例を図 16.3、図 16.4、図 16.5 に示します。

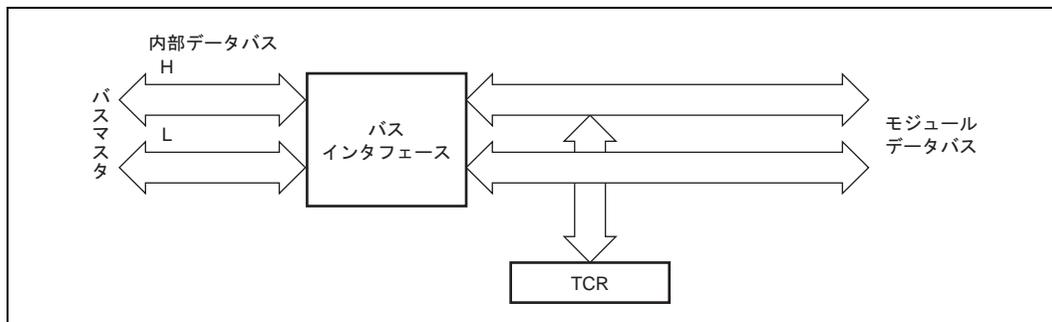


図 16.3 8 ビットレジスタのアクセス動作 [バスマスタ TCR (上位 8 ビット)]

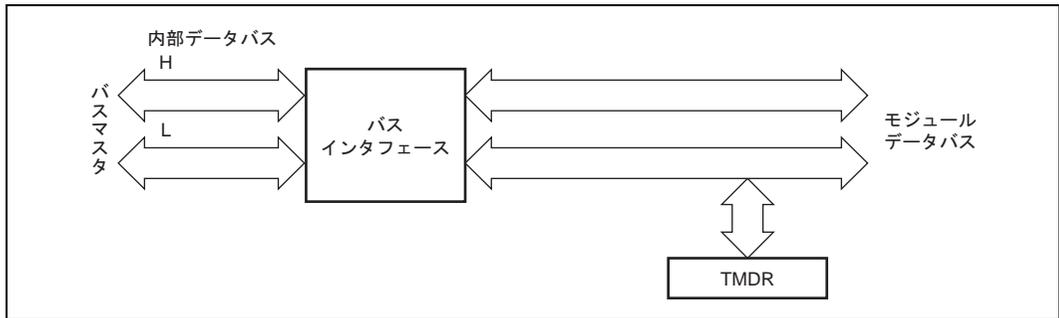


図 16.4 8 ビットレジスタのアクセス動作 [バスマスタ TMDR (下位 8 ビット)]

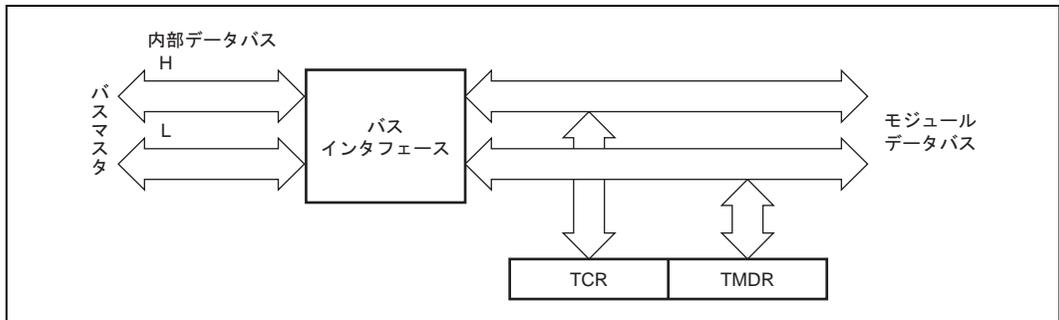


図 16.5 8 ビットレジスタのアクセス動作 [バスマスタ TCR、TMDR (16 ビット)]

## 16.4 動作説明

### 16.4.1 概要

各モードの動作概要を以下に示します。

#### (1) 通常動作

各チャンネルには、TCNT と TGR があります。TCNT は、アップカウント動作を行い、フリーランニング動作、周期カウント動作、または外部イベントカウント動作が可能です。

TGR は、それぞれインプットキャプチャレジスタまたはアウトプットコンペアレジスタとして使用することができます。

#### (2) 同期動作

TSYR により同期動作を設定したチャンネルの TCNT は、同期プリセット動作を行います。すなわち、同期動作に設定されたチャンネルのうち任意の TCNT を書き換えると、他のチャンネルの TCNT も同時に書き換えられます。また、同期動作に設定された複数のチャンネルの TCR のカウンタクリアビットの設定により、TCNT の同期クリアが可能です。

#### (3) バッファ動作

##### (a) TGR がアウトプットコンペアレジスタの場合

コンペアマッチが発生すると対応するチャンネルのバッファレジスタの値が TGR に転送されます。

##### (b) TGR がインプットキャプチャレジスタの場合

インプットキャプチャが発生すると、TCNT の値が TGR に転送されると同時に、それまで格納されていた TGR の値がバッファレジスタに転送されます。

#### (4) PWM モード

PWM 波形を出力するモードです。出力レベルは TIOR により設定できます。

各 TGR の設定により、デューティ 0 ~ 100% の PWM 波形が出力できます。

#### (5) 位相計数モード

チャンネル 1、2 で外部クロック入力端子から入力される 2 つのクロックの位相を検出して、TCNT をアップ/ダウンさせるモードです。位相計数モードに設定すると、対応する TCLK 端子はクロック入力となり、また TCNT はアップ/ダウンカウント動作を行います。

2 相エンコーダパルスの入力として使用できます。

### 16.4.2 基本機能

#### (1) カウンタの動作

TSTR の CST0 ~ CST2 ビットを 1 にセットすると、対応するチャンネルの TCNT はカウント動作を開始します。フリーランニングカウンタ動作、周期カウンタ動作などが可能です。

##### (a) カウント動作の設定手順例

カウント動作の設定手順例を図 16.6 に示します。

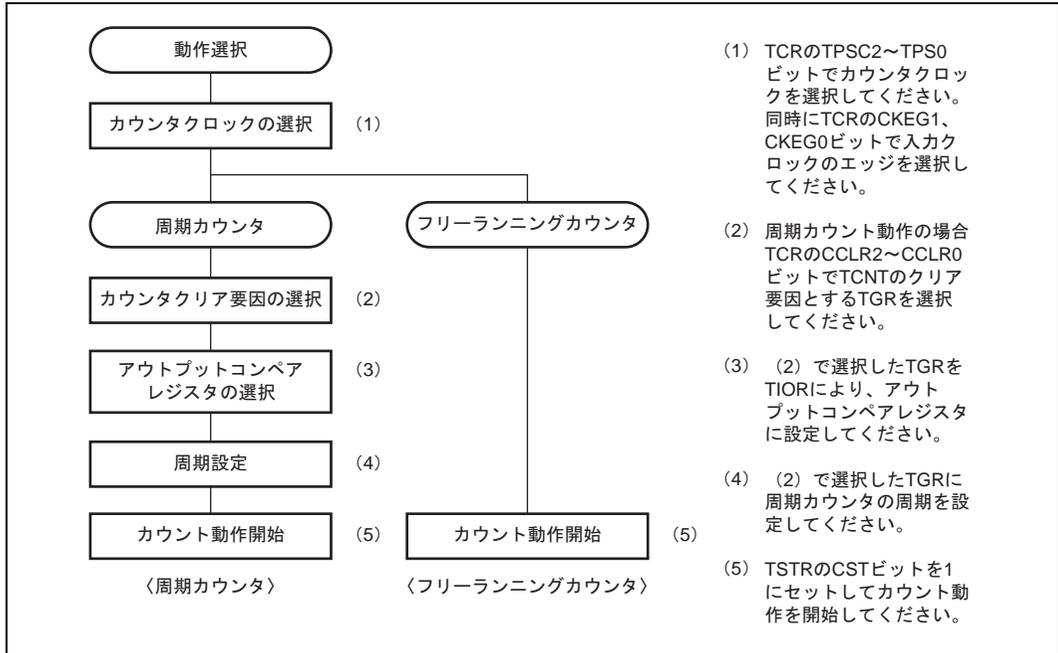


図 16.6 カウンタ動作設定手順例

(b) フリーランニングカウンタ動作と周期カウンタ動作

TPU の TCNT は、リセット直後はすべてフリーランニングカウンタの設定となっており、TSTR の対応するビットを 1 にセットするとフリーランニングカウンタとしてアップカウント動作を開始します。TCNT がオーバーフロー (H'FFFF ~ H'0000) すると、TSR の TCFV ビットが 1 にセットされます。このとき、対応する TIER の TCIEV ビットが 1 ならば、TPU は割り込みを要求します。TCNT はオーバーフロー後、H'0000 からアップカウント動作を継続します。

フリーランニングカウンタの動作を図 16.7 に示します。

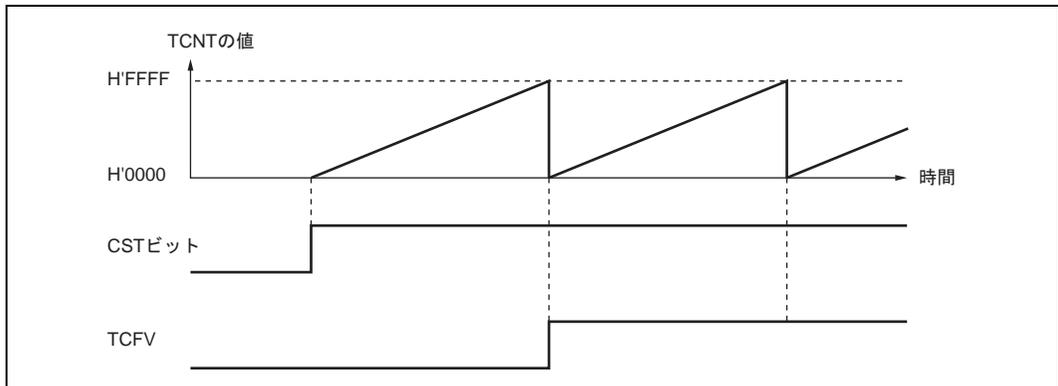


図 16.7 フリーランニングカウンタの動作

## 16. 16ビットタイマパルスユニット (TPU)

TCNTのクリア要因にコンペアマッチを選択したときは、対応するチャンネルのTCNTは周期カウント動作を行います。周期設定用のTGRをアウトプットコンペアレジスタに設定し、TCRのCCLR2～CCLR0ビットによりコンペアマッチによるカウンタクリアを選択します。設定後、TSTRの対応するビットを1にセットすると、周期カウンタとしてアップカウント動作を開始します。カウント値がTGRの値と一致すると、TSRのTGFビットが1にセットされ、TCNTはH'0000にクリアされます。

このとき対応するTIERのTGIEビットが1ならば、TPUは割り込みを要求します。TCNTはコンペアマッチ後、H'0000からアップカウント動作を継続します。

周期カウンタの動作を図16.8に示します。

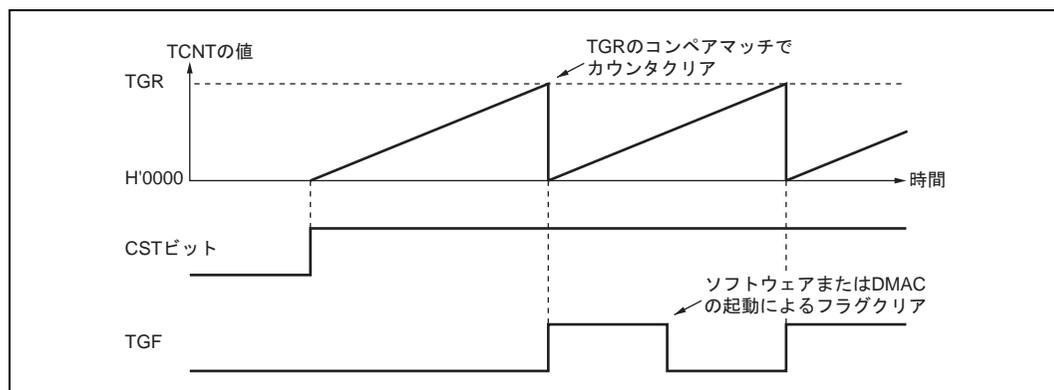


図 16.8 周期カウンタの動作

### (2) コンペアマッチによる波形出力機能

TPUは、コンペアマッチにより対応する出力端子から0出力/1出力/トグル出力を行うことができます。

#### (a) コンペアマッチによる波形出力動作の設定手順例

コンペアマッチによる波形出力動作の設定手順例を図16.9に示します。

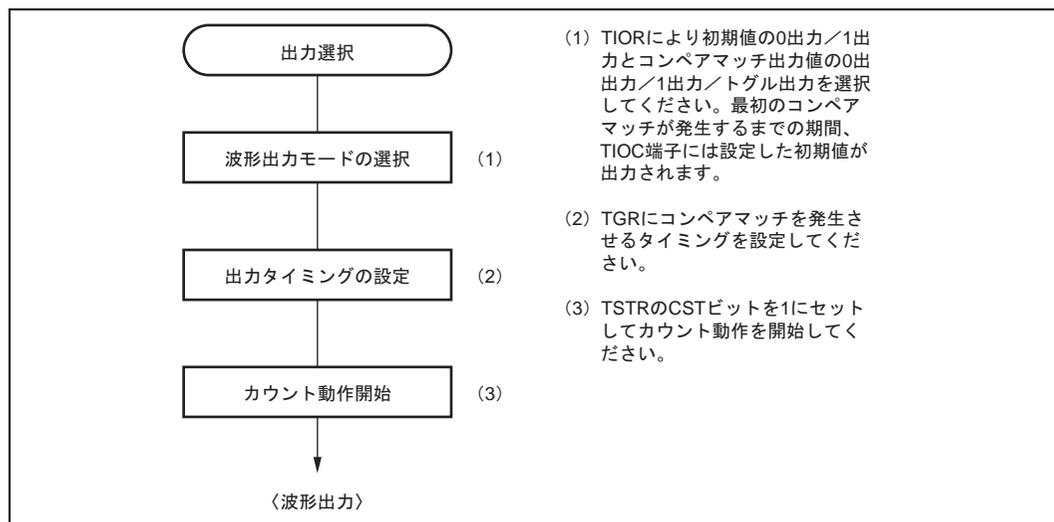


図 16.9 コンペアマッチによる波形出力動作例

## (b) 波形出力動作例

0出力 / 1出力例を図 16.10 に示します。

TCNT を、フリーランニングカウント動作とし、コンペアマッチ A により 1 出力、コンペアマッチ B により 0 出力となるように設定した場合の例です。設定したレベルと端子のレベルが一致した場合には、端子のレベルは変化しません。

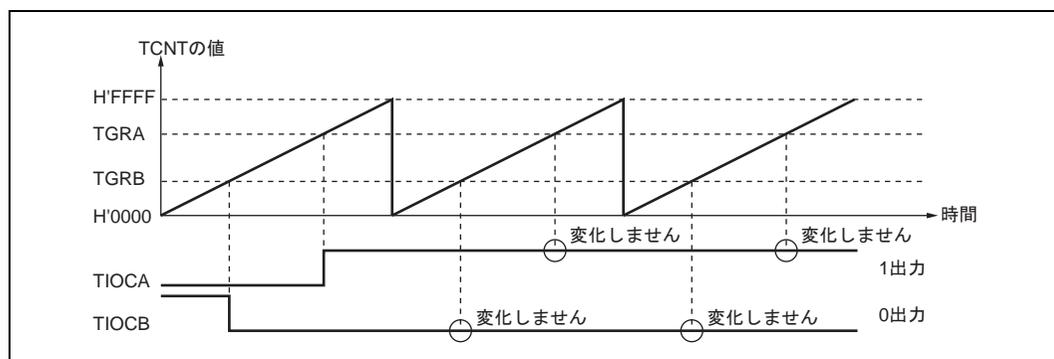


図 16.10 0出力 / 1出力の動作例

トグル出力の例を図 16.11 に示します。

TCNT を周期カウント動作 (コンペアマッチ B によりカウンタクリア) に、コンペアマッチ A、B ともトグル出力となるように設定した場合の例です。

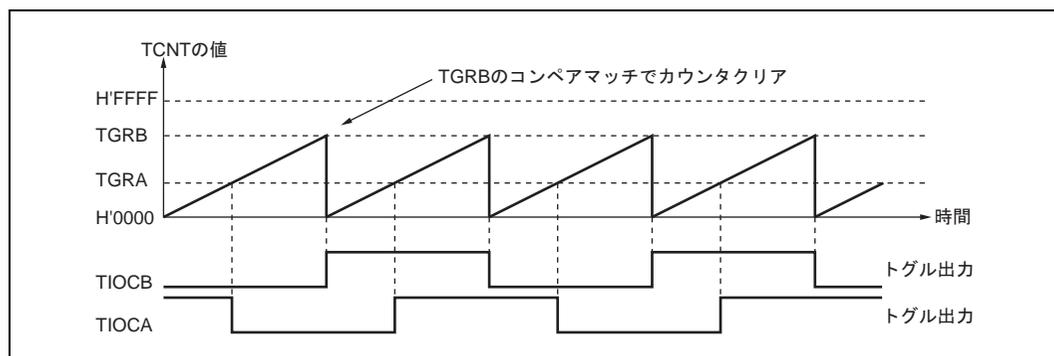


図 16.11 トグル出力の動作例

## (3) インพุットキャプチャ機能

TIOC 端子の入力エッジを検出して TCNT の値を TGR に転送することができます。

検出エッジは立ち上がりエッジ / 立ち下がりエッジ / 両エッジから選択できます。

## (a) インพุットキャプチャ動作の設定手順例

インพุットキャプチャ動作の設定手順例を図 16.12 に示します。

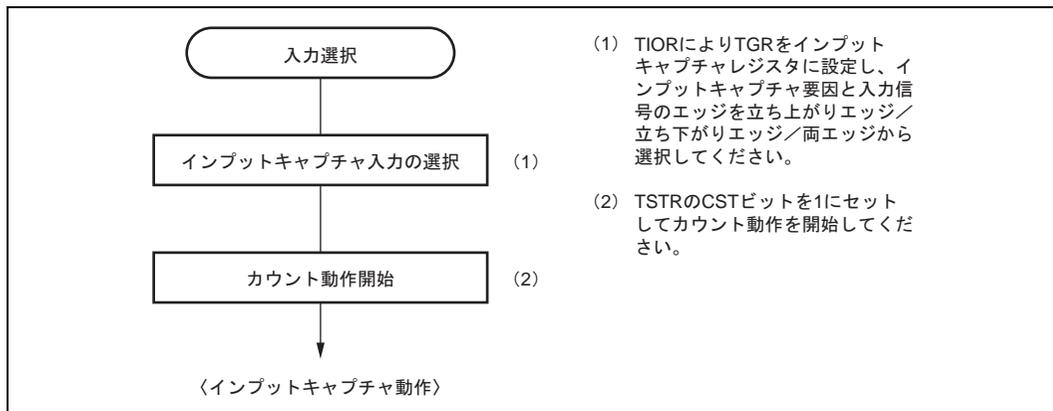


図 16.12 インットキャプチャ動作の設定例

(b) インットキャプチャ動作例

インットキャプチャ動作例を図 16.13 に示します。

TIOCA 端子のインットキャプチャ入力エッジは立ち上がり / 立ち下がり両エッジ、また TIOCB 端子のインットキャプチャ入力エッジは立ち下がりエッジを選択し、TCNT は TGRB のインットキャプチャでカウンタクリアされるように設定した場合の例です。

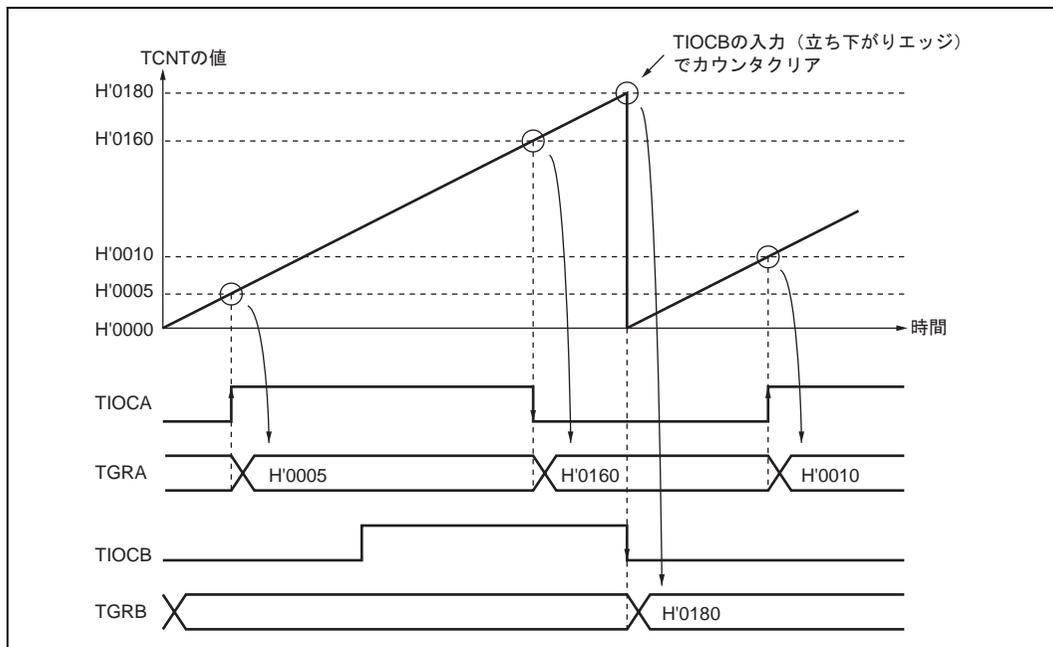


図 16.13 インットキャプチャ動作例

### 16.4.3 同期動作

同期動作は、複数の TCNT の値を同時に書き換えることができます (同期プリセット)。また、TCR の設定により複数の TCNT を同時にクリアすることができます (同期クリア)。

同期動作により、1つのタイムベースに対して TGR を増加することができます。

チャンネル 0 ~ 2 はすべて同期動作の設定が可能です。

#### (1) 同期動作の設定手順例

同期動作の設定手順例を図 16.14 に示します。

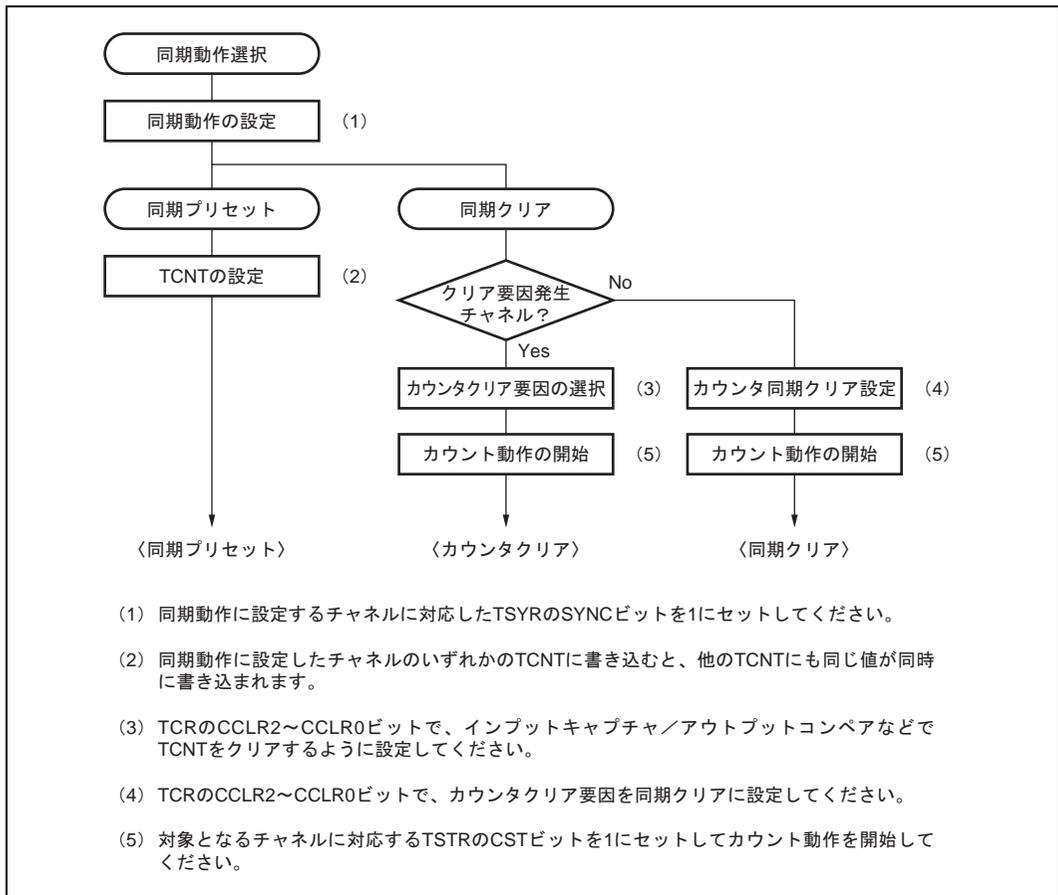


図 16.14 同期動作の設定手順例

#### (2) 同期動作の例

同期動作の例を図 16.15 に示します。

チャンネル 0 ~ 2 を同期動作かつ PWM モード 1 に設定し、チャンネル 0 のカウンタクリア要因を TGR0B のコンペアマッチ、またチャンネル 1、2 のカウンタクリア要因を同期クリアに設定した場合の例です。

## 16. 16ビットタイマパルスユニット (TPU)

3相のPWM波形をTIOC0A、TIOC1A、TIOC2A端子から出力します。このとき、チャンネル0～2のTCNTは同期プリセット、TGR0Bのコンペアマッチによる同期クリアを行い、TGR0Bに設定したデータがPWM周期となります。

PWMモードについては、「16.4.5 PWMモード」を参照してください。

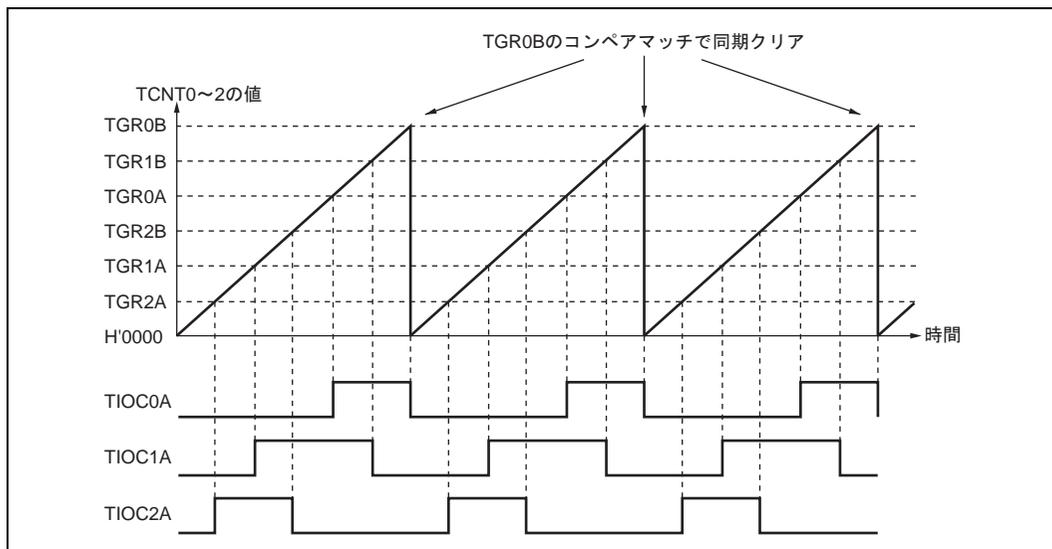


図 16.15 同期動作の動作例

### 16.4.4 バッファ動作

バッファ動作は、チャンネル0が持つ機能です。TGRCとTGRDをバッファレジスタとして使用することができます。

バッファ動作は、TGRをインプットキャプチャレジスタに設定した場合と、コンペアマッチレジスタに設定した場合のそれぞれで動作内容が異なります。

表 16.5 にバッファ動作時のレジスタの組み合わせを示します。

表 16.5 レジスタの組み合わせ

チャンネル	タイマジェネラルレジスタ	バッファレジスタ
0	TGR0A	TGR0C
	TGR0B	TGR0D

#### (a) TGRがアウトプットコンペアレジスタの場合

コンペアマッチが発生すると対応するチャンネルのバッファレジスタの値がタイマジェネラルレジスタに転送されます。

この動作を図 16.16 に示します。

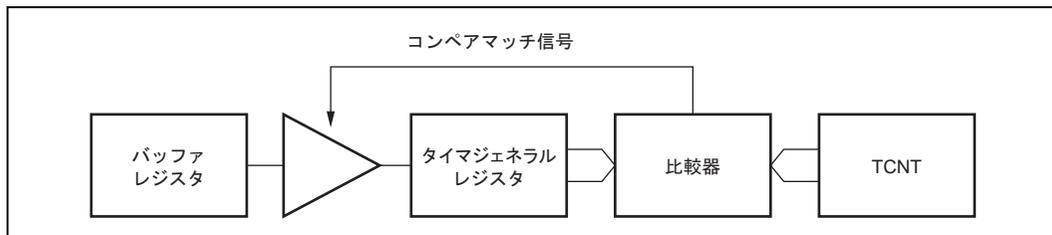


図 16.16 コンペアマッチバッファ動作

## (b) TGR がインプットキャプチャレジスタの場合

インプットキャプチャが発生すると TCNT の値を TGR に転送すると同時に、それまで格納されていたタイムジェネラルレジスタの値をバッファレジスタに転送します。

この動作を図 16.17 に示します。

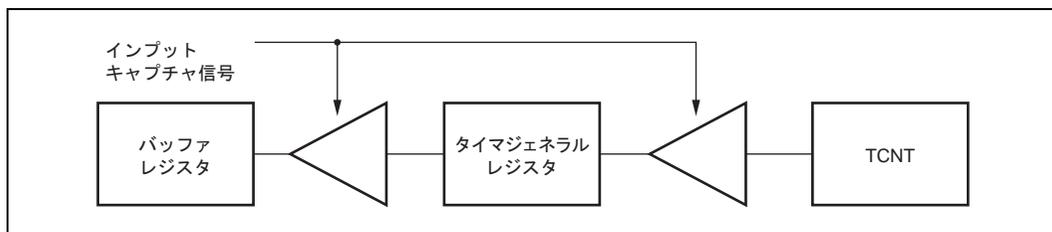


図 16.17 インプットキャプチャバッファ動作

## (1) バッファ動作の設定手順例

バッファ動作の設定手順例を図 16.18 に示します。

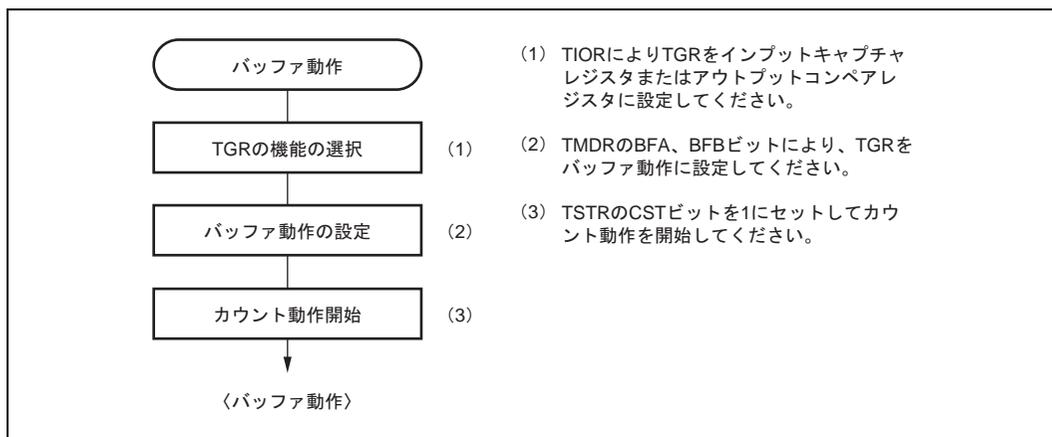


図 16.18 バッファ動作の設定手順例

## (2) バッファ動作例

## (a) TGR がアウトプットコンペアレジスタの場合

チャンネル0をPWMモード1に設定し、TGRAとTGRCをバッファ動作に設定した場合の動作例を図16.19に示します。TCNTはコンペアマッチBによりクリア、出力はコンペアマッチAで1出力、コンペアマッチBで0出力に設定した例です。

バッファ動作が設定されているため、コンペアマッチAが発生すると出力を変化させると同時に、バッファレジスタTGRCの値がタイマジェネラルレジスタTGRAに転送されます。この動作は、コンペアマッチAが発生する度に繰り返されます。

PWMモードについては、「16.4.5 PWMモード」を参照してください。

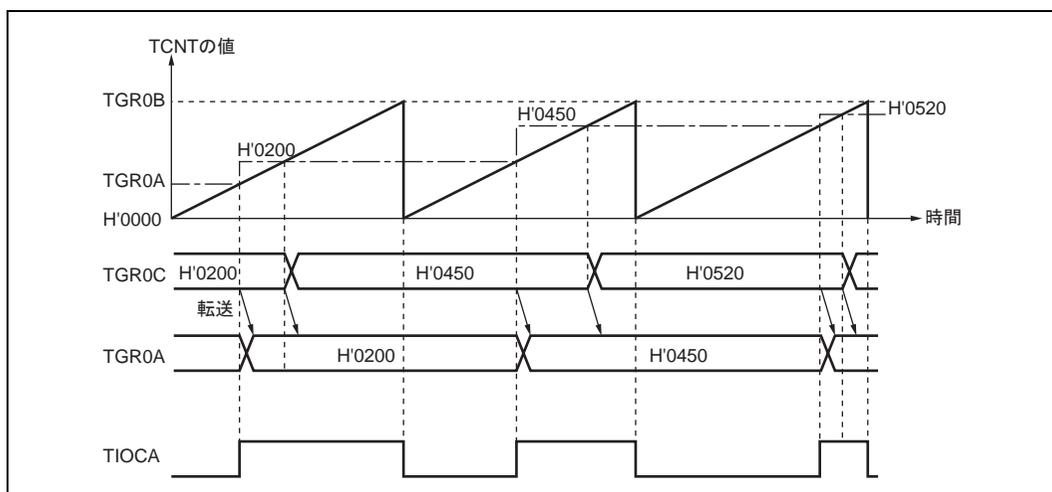


図 16.19 バッファ動作例 (1)

## (b) TGR がインプットキャプチャレジスタの場合

TGRAをインプットキャプチャレジスタに設定し、TGRAとTGRCをバッファ動作に設定したときの動作例を図16.20に示します。

TCNTはTGRAのインプットキャプチャでカウンタクリア、TIOCA端子のインプットキャプチャ入力エッジは立ち上がりエッジ/立ち下がりエッジの両エッジが選択されています。

バッファ動作が設定されているため、インプットキャプチャAによりTCNTの値がTGRAに格納されると同時に、それまでTGRAに格納されていた値がTGRCに転送されます。

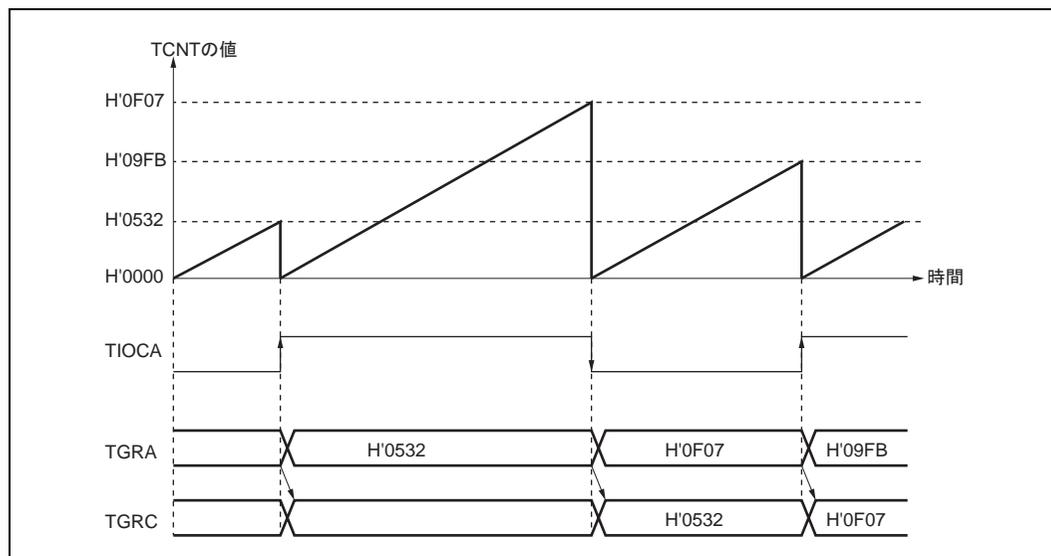


図 16.20 バッファ動作例 (2)

### 16.4.5 PWM モード

PWM モードは出力端子よりそれぞれ PWM 波形を出力します。各 TGR のコンペアマッチによる出力レベルは 0 出力 / 1 出力 / トグル出力の中から選択可能です。

TGR のコンペアマッチをカウンタクリア要因とすることにより、そのレジスタに周期を設定することができます。全チャンネル独立に PWM モードに設定できます。同期動作も可能です。

PWM モードは次に示すように 2 種類あります。

#### (a) PWM モード 1

TGRA と TGRB、TGRC と TGRD をペアで使用して TIOCA、TIOCC 端子から PWM 出力を生成します。

TIOCA、TIOCC 端子からコンペアマッチ A、C によって TIOR の IOA3 ~ IOA0、IOC3 ~ IOC0 ビットで指定した出力を、また、コンペアマッチ B、D によって TIOR の IOB3 ~ IOB0、IOD3 ~ IOD0 ビットで指定した出力を行います。初期出力値は TGRA、TGRC に設定した値になります。ペアで使用する TGR の設定値が同一の場合、コンペアマッチが発生しても出力値は変化しません。

PWM モード 1 では、最大 4 相の PWM 出力が可能です。

#### (b) PWM モード 2

TGR の 1 本を周期レジスタ、他の TGR をデューティレジスタに使用して PWM 出力を生成します。コンペアマッチによって、TIOR で指定した出力を行います。また、同期レジスタのコンペアマッチによるカウンタのクリアで各端子の出力値は TIOR で設定した初期値が出力されます。同期レジスタとデューティレジスタの設定値が同一の場合、コンペアマッチが発生しても出力値は変化しません。

PWM モード 2 では、同期動作と併用することにより最大 7 相の PWM 出力が可能です。

PWM 出力端子とレジスタの対応を表 16.6 に示します。

表 16.6 各 PWM 出力のレジスタと出力端子

チャンネル	レジスタ	出力端子	
		PWM モード 1	PWM モード 2
0	TGR0A	TIOCA0	TIOCA0
	TGR0B		TIOCB0
	TGR0C	TIOCC0	TIOCC0
	TGR0D		TIOCD0
1	TGR1A	TIOCA1	TIOCA1
	TGR1B		TIOCB1
2	TGR2A	TIOCA2	TIOCA2
	TGR2B		TIOCB2

【注】 PWM モード 2 のとき、周期を設定した TGR の PWM 出力はできません。

(1) PWM モードの設定手順例

PWM モードの設定手順例を図 16.21 に示します。

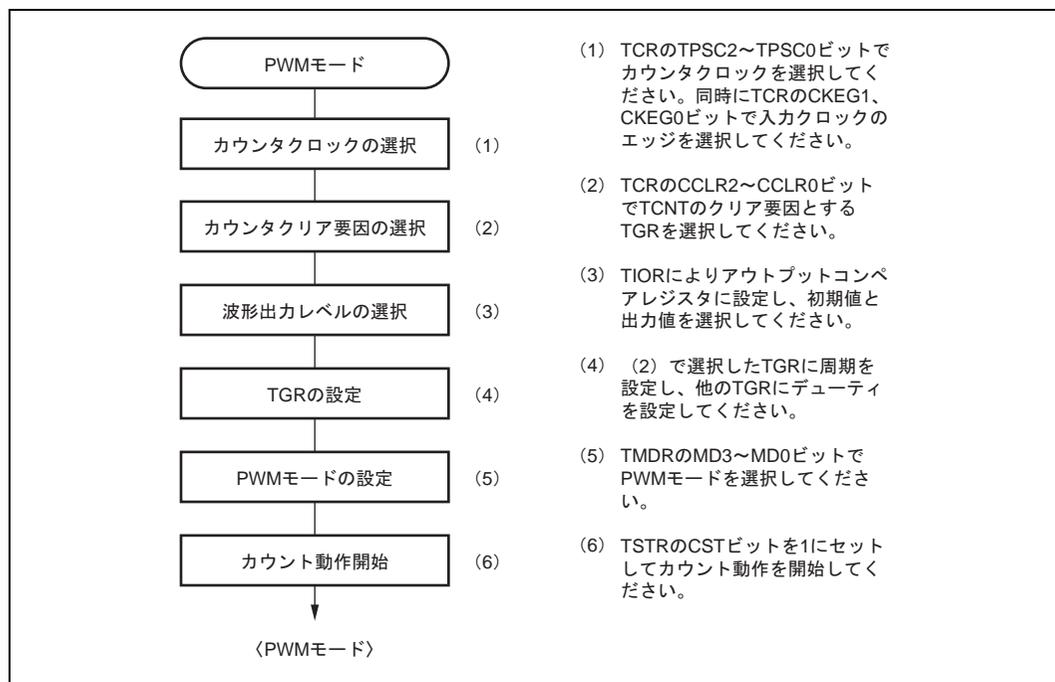


図 16.21 PWM モードの設定手順例

(2) PWM モードの動作例

PWM モード 1 の動作例を図 16.22 に示します。

この図は、TCNT のクリア要因を TGRA のコンペアマッチとし、TGRA の初期出力値と出力値を 0、TGRB の出力値を 1 出力に設定した場合の例です。

この場合、TGRA に設定した値が周期となり、TGRB に設定した値がデューティになります。

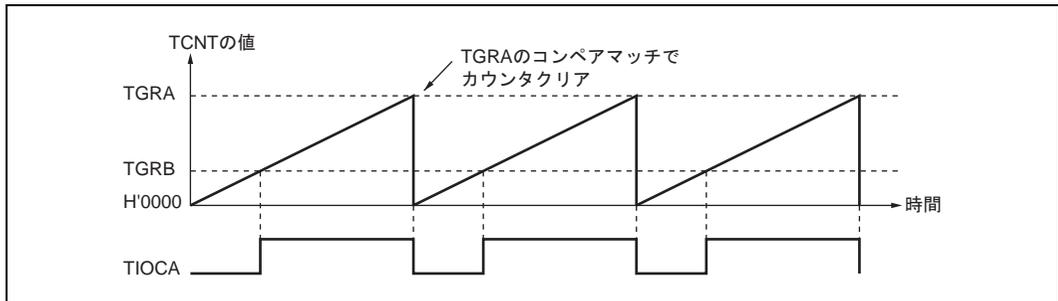


図 16.22 PWM モードの動作例 (1)

PWM モード 2 の動作例を図 16.23 に示します。

この図は、チャンネル 0 と 1 を同期動作させ、TCNT のクリア要因を TGR1B のコンペアマッチとし、他の TGR の初期出力値を 0、出力値を 1 に設定して 5 相の PWM 波形を出力させた場合の例です。

この場合、TGR1B に設定した値が周期となり、他の TGR に設定した値がデューティになります。

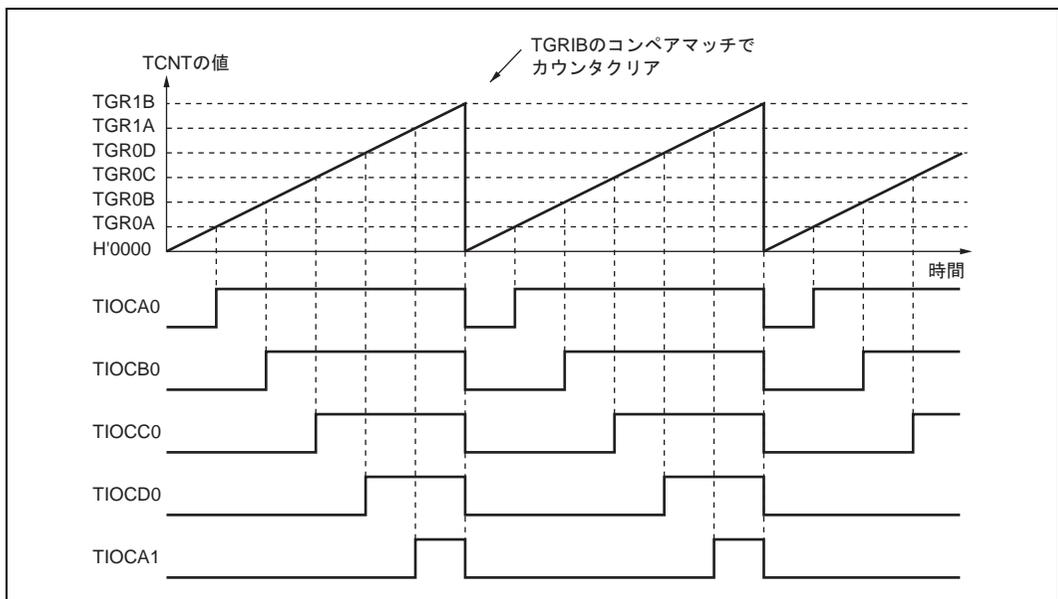


図 16.23 PWM モードの動作例 (2)

PWM モードで、デューティ 0%、デューティ 100% の PWM 波形を出力する例を図 16.24 に示します。

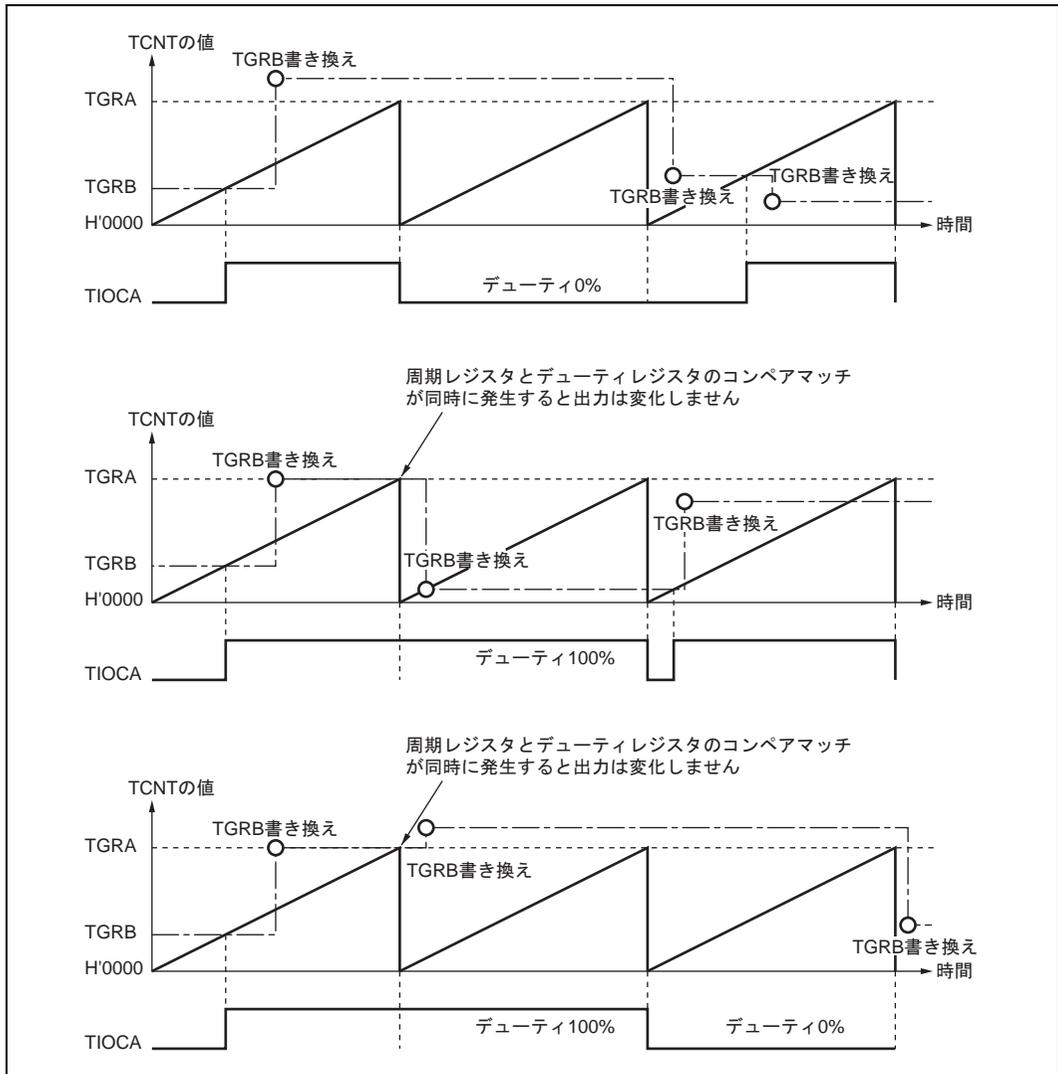


図 16.24 PWM モード動作例 (3)

### 16.4.6 位相計数モード

位相計数モードは、2本の外部クロック入力の位相差を検出し、TCNTをアップ/ダウンカウントします。このモードはチャンネル1、2で設定可能です。

位相計数モードに設定すると、TCRのTPSC2～TPSC0ビット、CKEG1、CKEG0ビットの設定にかかわらずカウンタ入力クロックは外部クロックを選択し、TCNTはアップ/ダウンカウンタとして動作します。ただし、TCRのCCLR1、CCLR0ビット、TIOR、TIER、TGRの機能は有効ですので、インプットキャプチャ/コンペアマッチ機能や割り込み機能は使用することができます。

TCNTがアップカウント時、オーバフローが発生するとするとTSRのTCFVフラグがセットされます。また、ダウンカウント時にアンダフローが発生すると、TCFUフラグがセットされます。

TSRのTCFDビットはカウント方向フラグです。TCFDフラグをリードすることにより、TCNTがアップカウントしているかダウンカウントしているかを確認することができます。

表 16.7 に外部クロック端子とチャンネルの対応を示します。

表 16.7 位相計数モードクロック入力端子

チャンネル	外部クロック端子	
	A相	B相
1を位相計数モードとするとき	TCLKA	TCLKB
2を位相計数モードとするとき	TCLKC	TCLKD

#### (1) 位相計数モードの設定手順例

位相計数モードの設定手順例を図 16.25 に示します。

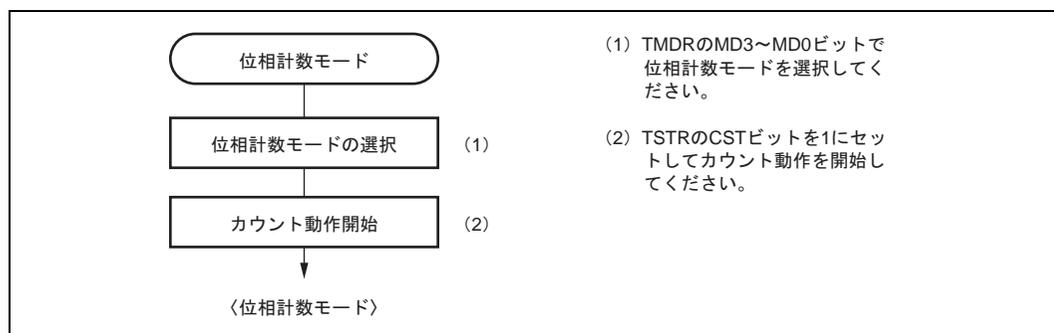


図 16.25 位相計数モードの設定手順例

#### (2) 位相計数モードの動作例

位相計数モードでは、2本の外部クロックの位相差でTCNTがアップ/ダウンカウントします。また、カウント条件により4つのモードがあります。

##### (a) 位相計数モード1

位相計数モード1の動作例を図 16.26 に、TCNTのアップ/ダウンカウント条件を表 16.8 に示します。

16. 16ビットタイマパルスユニット (TPU)

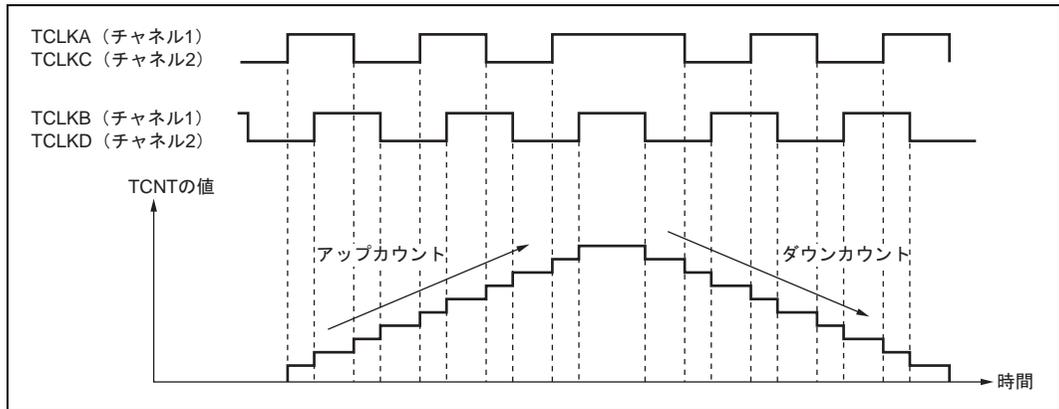


図 16.26 位相計数モード 1 の動作例

表 16.8 位相計数モード 1 のアップ/ダウンカウント条件

TCLKA (チャンネル 1) TCLKC (チャンネル 2)	TCLKB (チャンネル 1) TCLKD (チャンネル 2)	動作内容
High レベル	┌┐	アップカウント
Low レベル	└┘	
┌┐	Low レベル	
└┘	High レベル	
High レベル	└┘	ダウンカウント
Low レベル	┌┐	
┌┐	High レベル	
└┘	Low レベル	

【注】

- ┌┐ : 立ち上がりエッジ
- └┘ : 立ち下がりエッジ

## (b) 位相計数モード 2

位相計数モード 2 の動作例を図 16.27 に、TCNT のアップ/ダウンカウント条件を表 16.9 に示します。

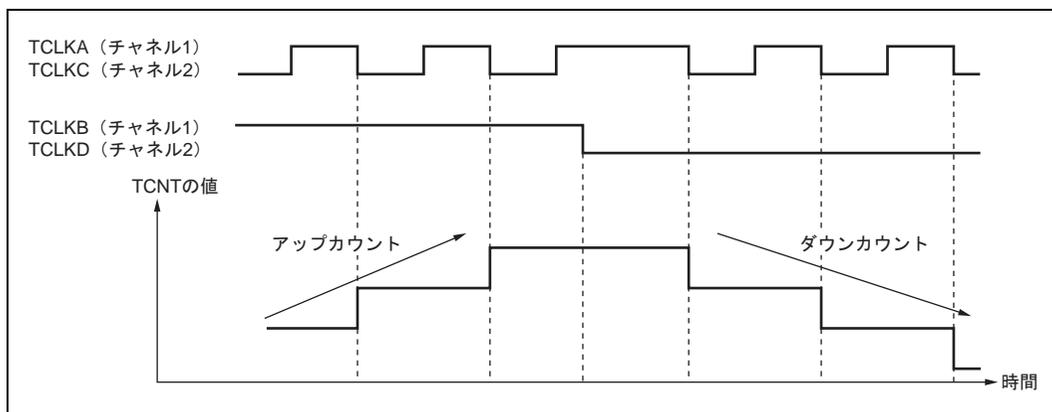


図 16.27 位相計数モード 2 の動作例

表 16.9 位相計数モード 2 のアップ/ダウンカウント条件

TCLKA (チャンネル 1) TCLKC (チャンネル 2)	TCLKB (チャンネル 1) TCLKD (チャンネル 2)	動作内容
High レベル	┌┐	Don't care
Low レベル	└┘	Don't care
┌┐	Low レベル	Don't care
└┘	High レベル	アップカウント
High レベル	└┘	Don't care
Low レベル	┌┐	Don't care
┌┐	High レベル	Don't care
└┘	Low レベル	ダウンカウント

【注】 ┌┐ : 立ち上がりエッジ  
└┘ : 立ち下がりエッジ

(c) 位相計数モード3

位相計数モード3の動作例を図16.28に、TCNTのアップ/ダウンカウント条件を表16.10に示します。

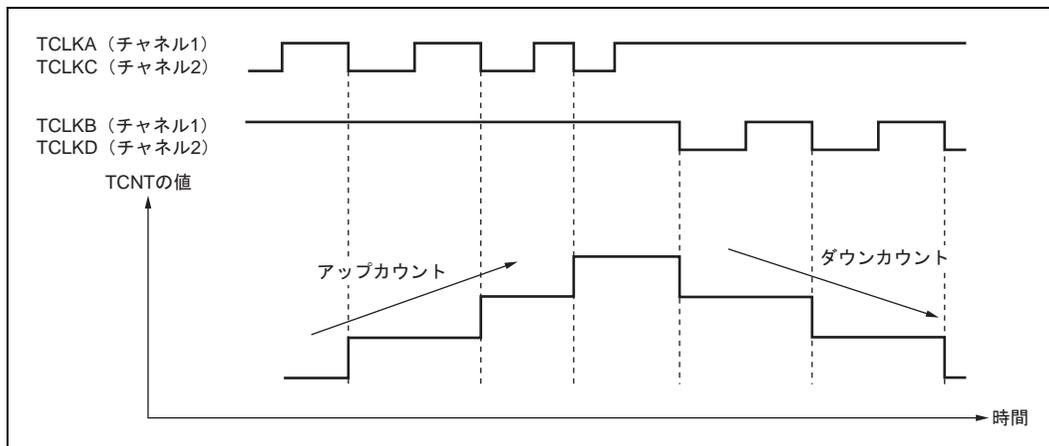


図 16.28 位相計数モード3の動作例

表 16.10 位相計数モード3のアップ/ダウンカウント条件

TCLKA (チャンネル1) TCLKC (チャンネル2)	TCLKB (チャンネル1) TCLKD (チャンネル2)	動作内容
High レベル	$\uparrow$	Don't care
Low レベル	$\downarrow$	Don't care
$\uparrow$	Low レベル	Don't care
$\downarrow$	High レベル	アップカウント
High レベル	$\downarrow$	ダウンカウント
Low レベル	$\uparrow$	Don't care
$\uparrow$	High レベル	Don't care
$\downarrow$	Low レベル	Don't care

【注】  $\uparrow$  : 立ち上がりエッジ  
 $\downarrow$  : 立ち下がりエッジ

## (d) 位相計数モード4

位相計数モード4の動作例を図16.29に、TCNTのアップ/ダウンカウント条件を表16.11に示します。

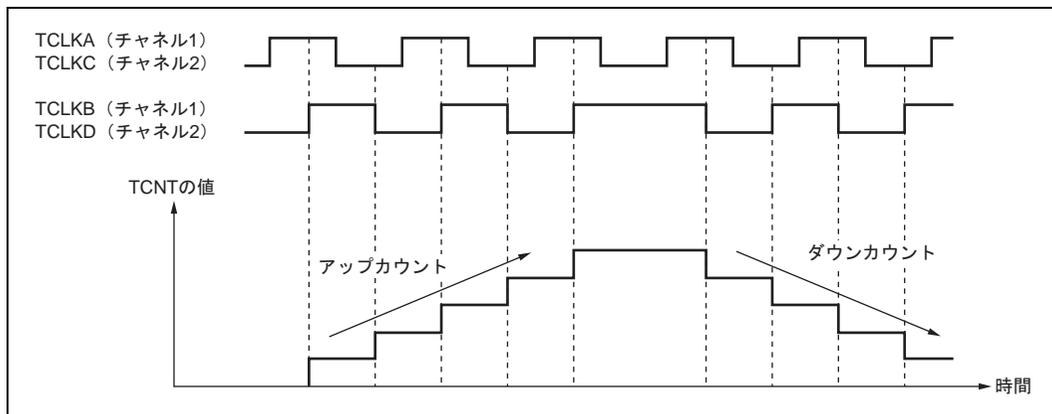


図 16.29 位相計数モード4の動作例

表 16.11 位相計数モード4のアップ/ダウンカウント条件

TCLKA (チャンネル1) TCLKC (チャンネル2)	TCLKB (チャンネル1) TCLKD (チャンネル2)	動作内容
High レベル	$\uparrow$	アップカウント
Low レベル	$\downarrow$	
$\uparrow$	Low レベル	Don't care
$\downarrow$	High レベル	
High レベル	$\downarrow$	ダウンカウント
Low レベル	$\uparrow$	
$\uparrow$	High レベル	Don't care
$\downarrow$	Low レベル	

【注】  $\uparrow$  : 立ち上がりエッジ  
 $\downarrow$  : 立ち下がりエッジ

## 16.5 割り込み

### 16.5.1 割り込み要因と優先順位

TPUの割り込み要因には、TGRのインプットキャプチャ/コンペアマッチ、TCNTのオーバフロー、アンダフローの3種類があります。各割り込み要因は、それぞれ専用のステータスフラグと、許可/禁止ビットを持っているため、割り込み要求信号の発生を独立に許可または禁止することができます。

割り込み要因が発生すると、TSRの対応するステータスフラグが1にセットされます。このときTIERの対応する許可/禁止ビットが1にセットされていれば、割り込みを要求します。ステータスフラグを0にクリアすることで割り込み要求は解除されます。

チャンネル間の優先順位は、割り込みコントローラにより変更可能です。チャンネル内の優先順位は固定です。詳細は「5. 割り込みコントローラ (INTC)」を参照してください。

表 16.12 に TPU の割り込み要因の一覧を示します。

表 16.12 TPU 割り込み一覧

チャンネル	割り込み要因	内容	DMACの起動	優先順位
0	TGI0A	TGR0Aのインプットキャプチャ/コンペアマッチ	可	
	TGI0B	TGR0Bのインプットキャプチャ/コンペアマッチ	可	
	TGI0C	TGR0Cのインプットキャプチャ/コンペアマッチ	可	
	TGI0D	TGR0Dのインプットキャプチャ/コンペアマッチ	可	
	TCI0V	TCNT0のオーバフロー	不可	
1	TGI1A	TGR1Aのインプットキャプチャ/コンペアマッチ	不可	
	TGI1B	TGR1Bのインプットキャプチャ/コンペアマッチ	不可	
	TCI1V	TCNT1のオーバフロー	不可	
	TCI1U	TCNT1のアンダフロー	不可	
2	TGI2A	TGR2Aのインプットキャプチャ/コンペアマッチ	不可	
	TGI2B	TGR2Bのインプットキャプチャ/コンペアマッチ	不可	
	TCI2V	TCNT2のオーバフロー	不可	
	TCI2U	TCNT2のアンダフロー	不可	

【注】 リセット直後の初期状態について示しています。チャンネル間の優先順位は割り込みコントローラにより変更可能です。

#### (1) インプットキャプチャ/コンペアマッチ割り込み

各チャンネルのTGRのインプットキャプチャ/コンペアマッチの発生により、TSRのTGFフラグが1にセットされたとき、TIERのTGIEビットが1にセットされていれば、割り込みを要求します。TGFフラグを0にクリアすることで割り込み要求は解除されます。TPUには、チャンネル0に4本、チャンネル1、2に各2本、計8本のインプットキャプチャ/コンペアマッチ割り込みがあります。

#### (2) オーバフロー割り込み

各チャンネルのTCNTのオーバフローの発生により、TSRのTCFVフラグが1にセットされたとき、TIERのTCIEVビットが1にセットされていれば、割り込みを要求します。TCFVフラグを0にクリアすることで割り込み要求は解除されます。TPUには、各チャンネルに1本、計3本のオーバフロー割り込みがあります。

(3) アンダフロー割り込み

各チャンネルの TCNT のアンダフローの発生により、TSR の TCFU フラグが 1 にセットされたとき、TIER の TCIEU ビットが 1 にセットされていれば、割り込みを要求します。TCFU フラグを 0 にクリアすることで割り込み要求は解除されます。TPU には、チャンネル 1、2 に各 1 本、計 2 本のアンダフロー割り込みがあります。

### 16.5.2 DMAC の起動

各チャンネルの TGR のインプットキャプチャ/コンペアマッチ割り込みによって、DMAC を起動することができます。詳細は「11. ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。

TPU では、チャンネル 0 の 4 本のインプットキャプチャ/コンペアマッチ割り込みを DMAC の起動要因とすることができます。

## 16.6 動作タイミング

### 16.6.1 入出力タイミング

#### (1) TCNTのカウンタタイミング

内部クロック動作の場合のTCNTのカウンタタイミングを図16.30に示します。また、外部クロック動作の場合のTCNTのカウンタタイミングを図16.31に示します。

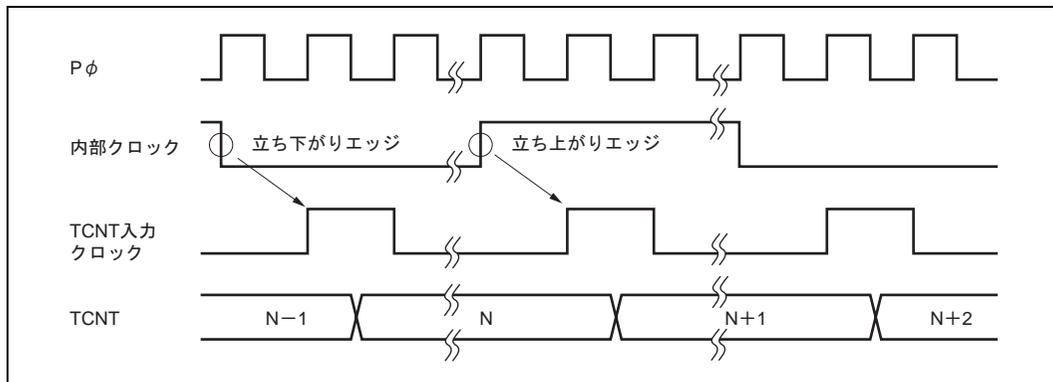


図 16.30 内部クロック動作時のカウンタタイミング

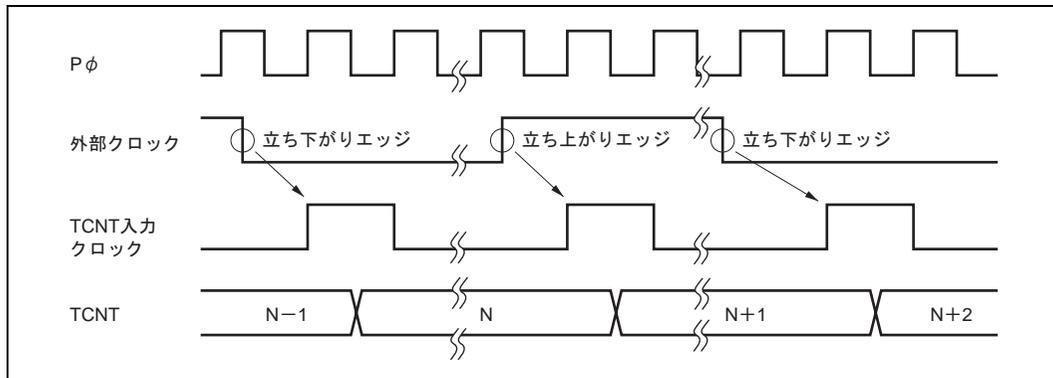


図 16.31 外部クロック動作時のカウンタタイミング

#### (2) アウトプットコンペア出力タイミング

コンペアマッチ信号は、TCNTとTGRが一致した最後のステート（TCNTが一致したカウント値を更新するタイミング）で発生します。コンペアマッチ信号が発生したとき、TIORで設定した出力値がアウトプットコンペア出力端子（TIOC端子）に出力されます。TCNTとTGRが一致した後、TCNT入力クロックが発生するまで、コンペアマッチ信号は発生しません。

アウトプットコンペア出力タイミングを図16.32に示します。

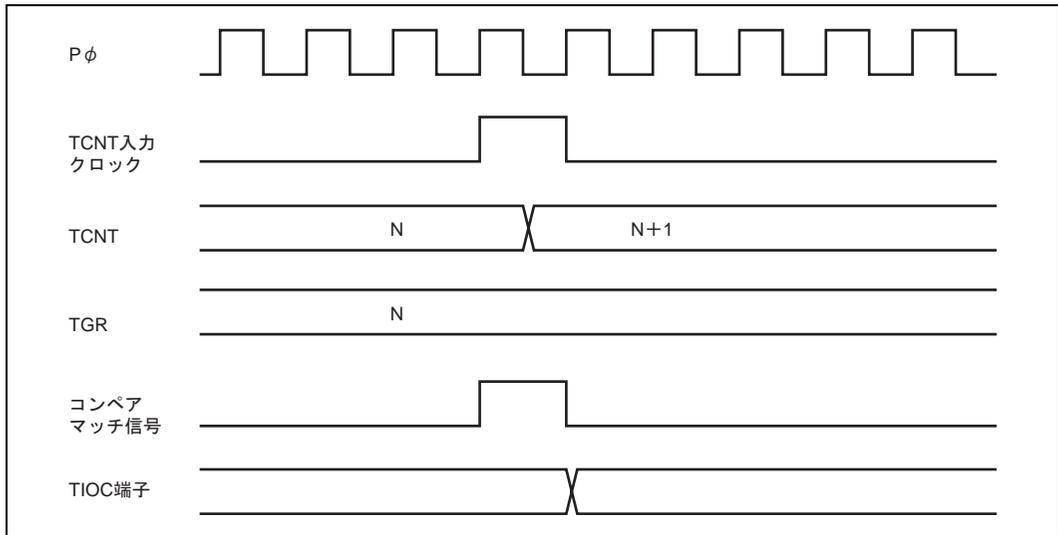


図 16.32 アウトプットコンペア出力タイミング

## (3) インพุットキャプチャ信号タイミング

インพุットキャプチャのタイミングを図 16.33 に示します。

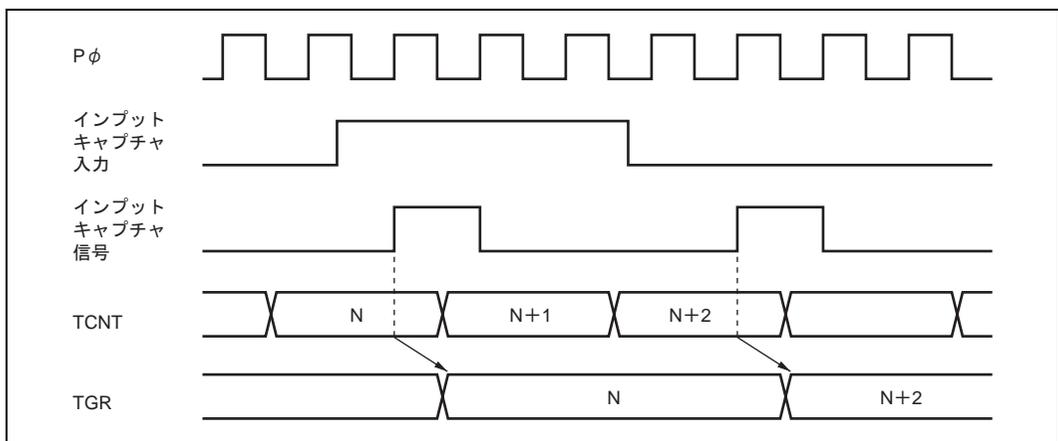


図 16.33 インพุットキャプチャ入力信号タイミング

## (4) コンペアマッチ/インพุットキャプチャによるカウンタクリアタイミング

コンペアマッチの発生によるカウンタクリアを指定した場合のタイミングを図 16.34 に示します。

インพุットキャプチャの発生によるカウンタクリアを指定した場合のタイミングを図 16.35 に示します。

16. 16ビットタイマパルスユニット (TPU)

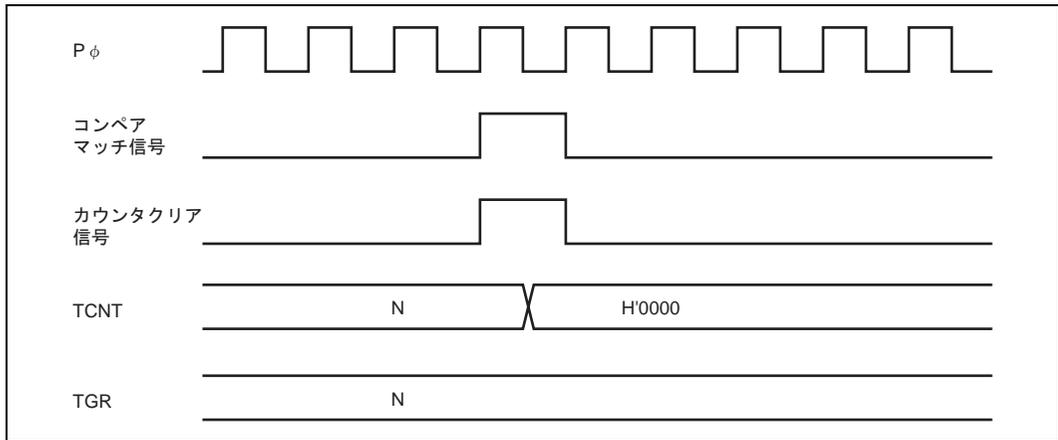


図 16.34 カウンタクリアタイミング (コンペアマッチ)

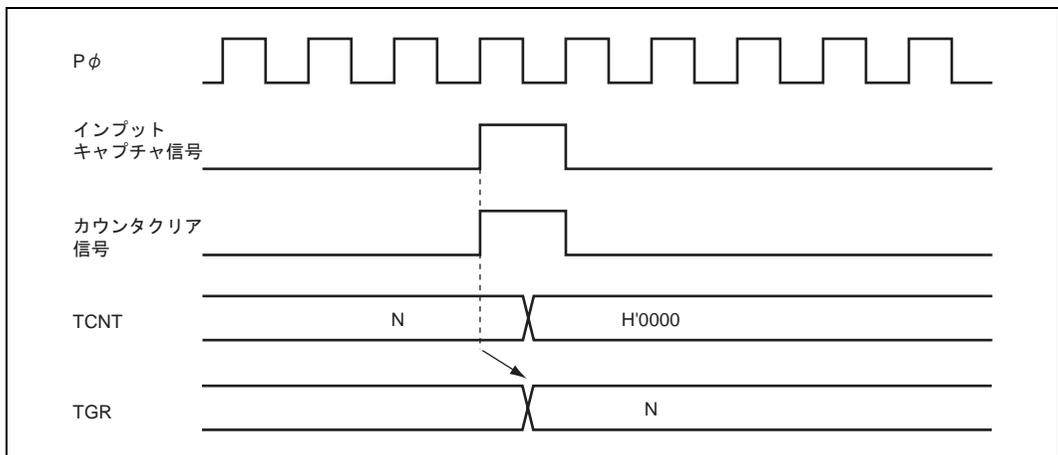


図 16.35 カウンタクリアタイミング (インプットキャプチャ)

(5) バッファ動作タイミング

バッファ動作の場合のタイミングを図 16.36、図 16.37 に示します。

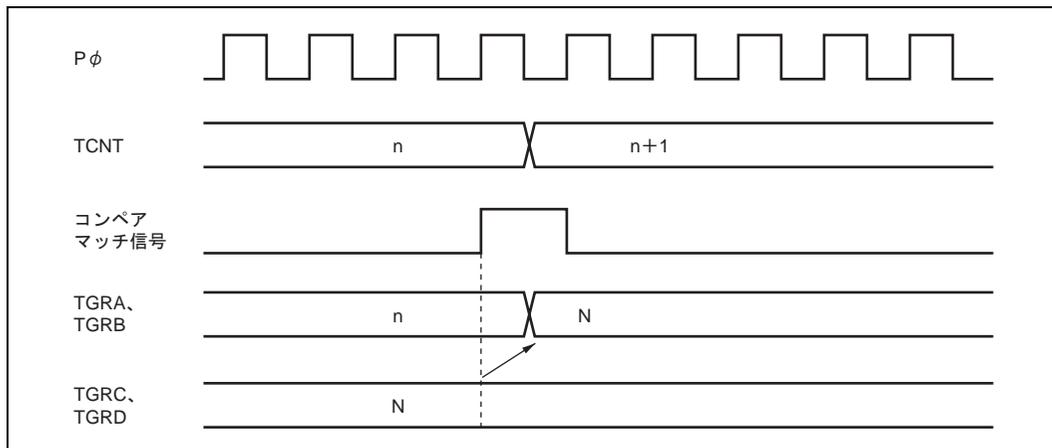


図 16.36 バッファ動作タイミング (コンペアマッチ)

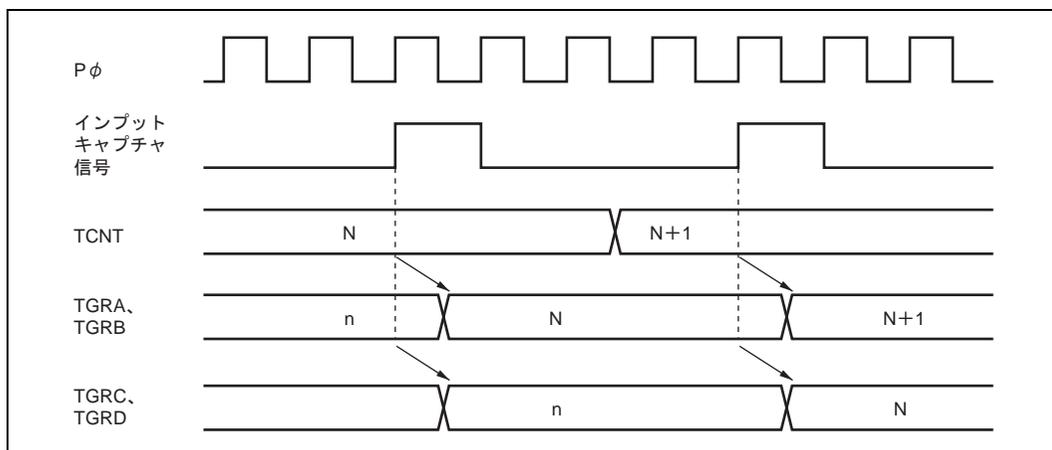


図 16.37 バッファ動作タイミング (インプットキャプチャ)

## 16.6.2 割り込み信号タイミング

### (1) コンペアマッチ時の TGF フラグのセットタイミング

コンペアマッチの発生による TSR の TGF フラグのセットタイミングと、TGI 割り込み要求信号のタイミングを図 16.38 に示します。

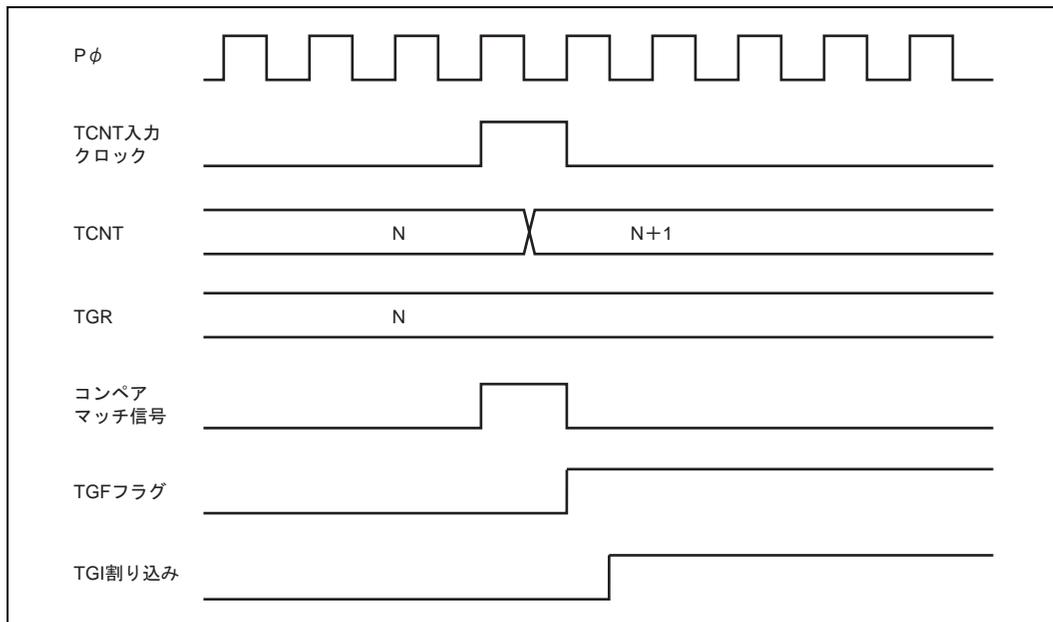


図 16.38 TGI 割り込みタイミング (コンペアマッチ)

(2) インพุットキャプチャ時の TGF フラグのセットタイミング

インพุットキャプチャの発生による TSR の TGF フラグのセットタイミングと、TGI 割り込み要求信号のタイミングを図 16.39 に示します。

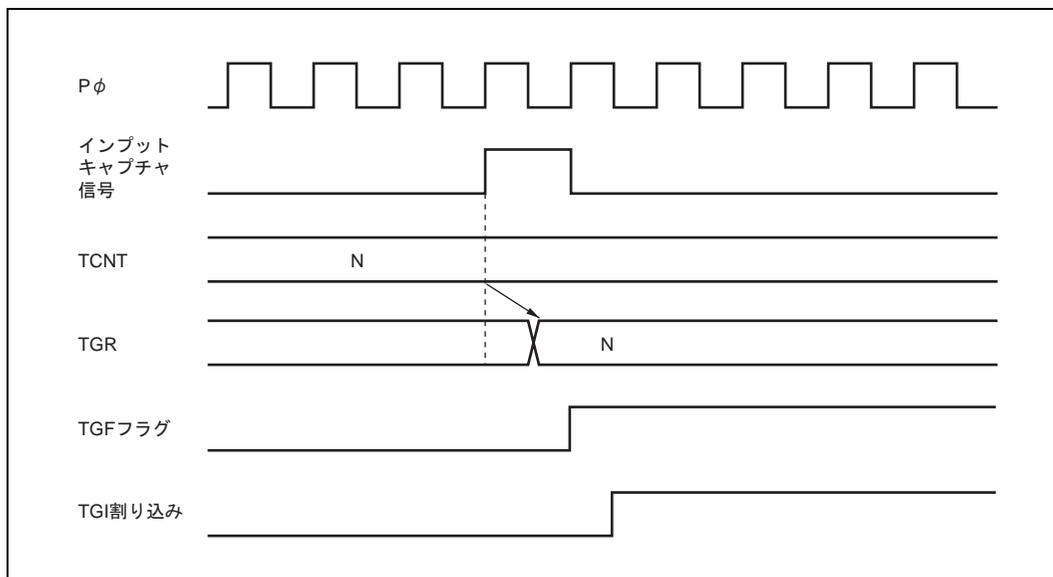


図 16.39 TGI 割り込みタイミング (インพุットキャプチャ)

## (3) TCFV フラグ / TCFU フラグのセットタイミング

オーバフローの発生による TSR の TCFV フラグのセットタイミングと、TCIV 割り込み要求信号のタイミングを図 16.40 に示します。

アンダフローの発生による TSR の TCFU フラグのセットタイミングと、TCIU 割り込み要求信号のタイミングを図 16.41 に示します。

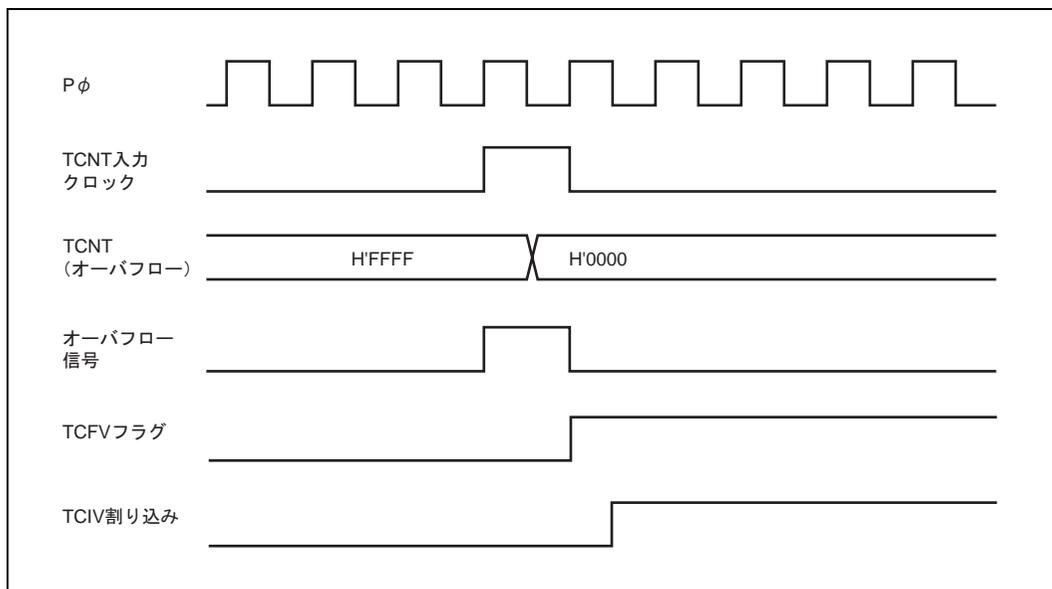


図 16.40 TCIV 割り込みのセットタイミング

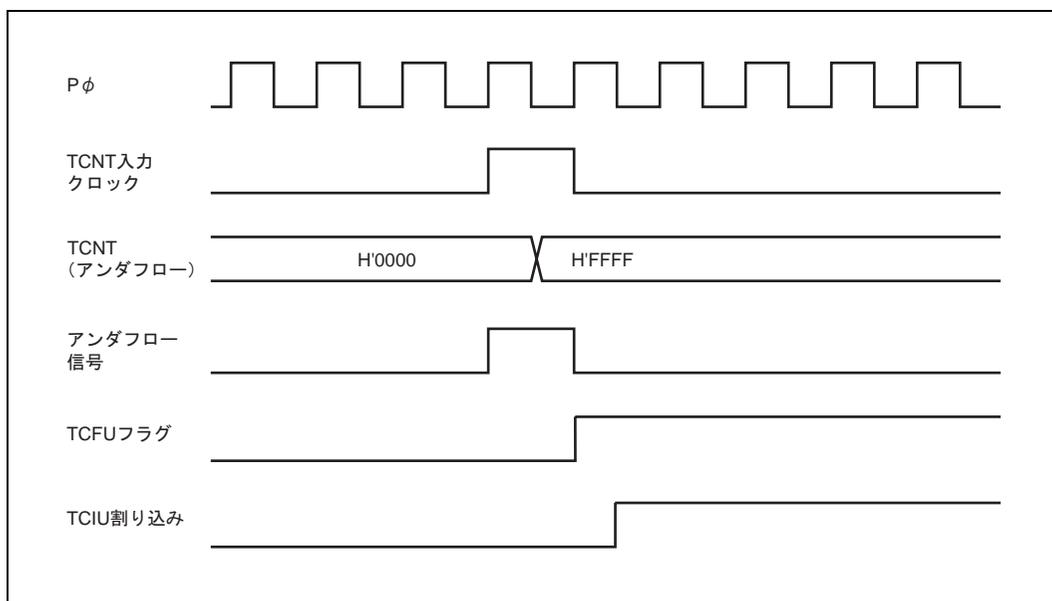


図 16.41 TCIU 割り込みのセットタイミング

(4) ステータスフラグのクリアタイミング

ステータスフラグはCPUが1の状態をリードした後、0を書き込むとクリアされます。DMACを起動する場合は、自動的にクリアすることもできます。CPUによるステータスフラグのクリアタイミングを図16.42に、DMACによるステータスフラグのクリアタイミングを図16.43に示します。

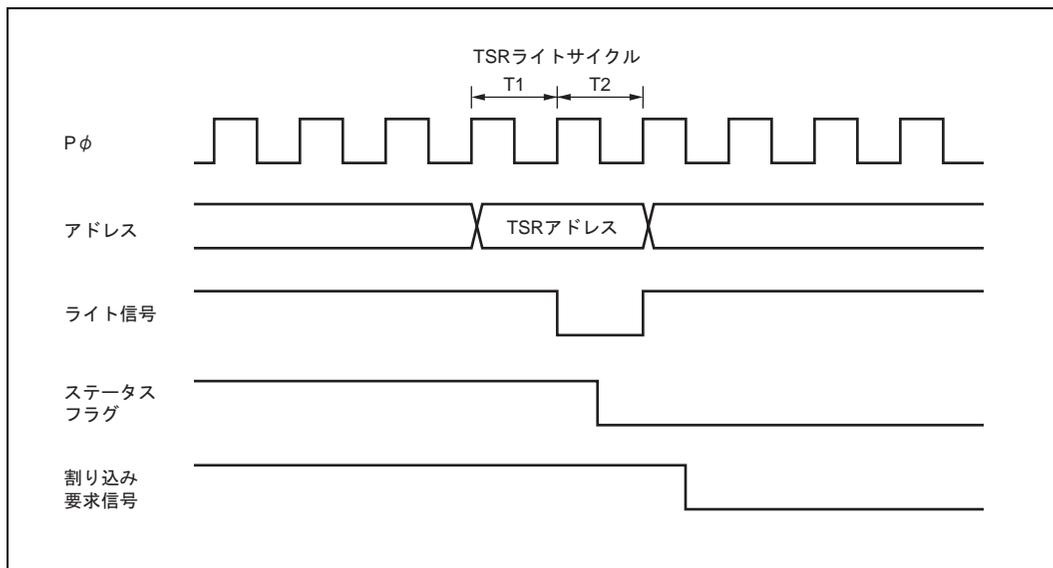


図 16.42 CPU によるステータスフラグのクリアタイミング

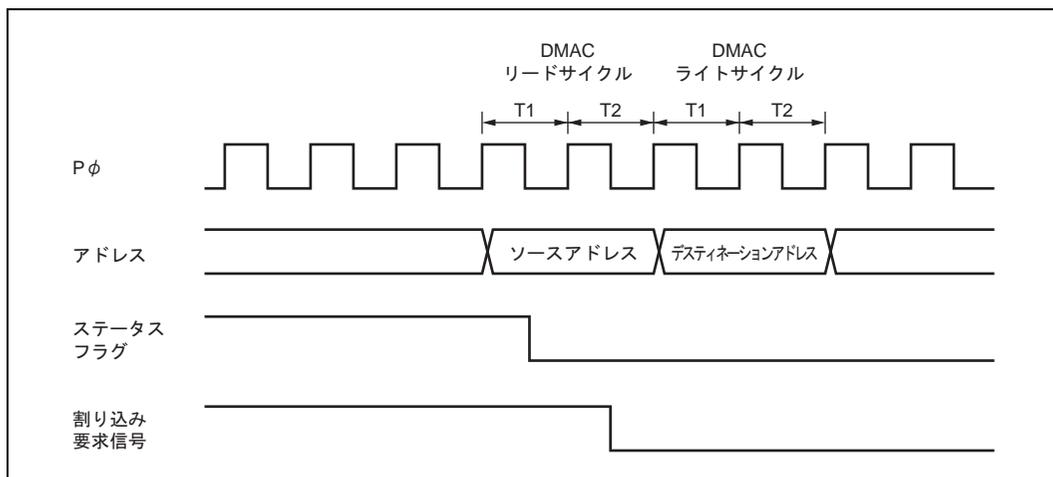


図 16.43 DMAC の起動によるステータスフラグのクリアタイミング

## 16.7 使用上の注意

TPU の動作中、次のような動作や競合が起こりますので注意してください。

### 16.7.1 入力クロックの制限事項

入力クロックのパルス幅は、単エッジの場合は 1.5 ステートクロック以上、両エッジの場合は 2.5 ステート以上が必要です。これ以下のパルス幅では正しく動作しませんのでご注意ください。

位相計数モードの場合は、2 本の入力クロックの位相差およびオーバーラップはそれぞれ 1.5 ステート以上、パルス幅は 2.5 ステート以上必要です。位相計数モードの入力クロックの条件を図 16.44 に示します。

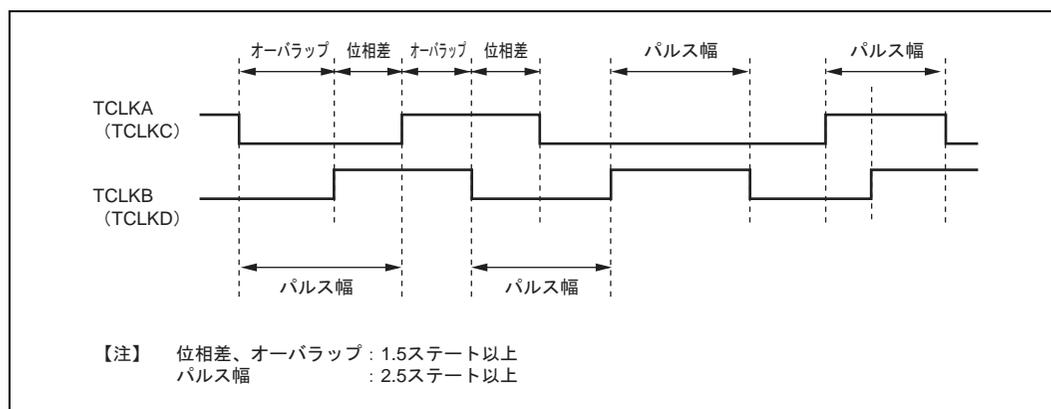


図 16.44 位相計数モード時の位相差、オーバーラップ、およびパルス幅

### 16.7.2 周期設定上の注意事項

コンペアマッチによるカウンタクリアを設定した場合、TCNT は TGR の値と一致した最後のステート (TCNT が一致したカウント値を更新するタイミング) でクリアされます。このため、実際のカウンタの周波数は次の式のようにになります。

$$f = \frac{P\phi}{(N+1)}$$

f : カウンタ周波数

P : 周辺モジュールクロック

N : TGR の設定値

### 16.7.3 TCNT のライトとクリアの競合

TCNT のライトサイクル中の T2 ステートで、カウンタクリア信号が発生すると、TCNT へのライトは行われずに、TCNT のクリアが優先されます。

このタイミングを図 16.45 に示します。

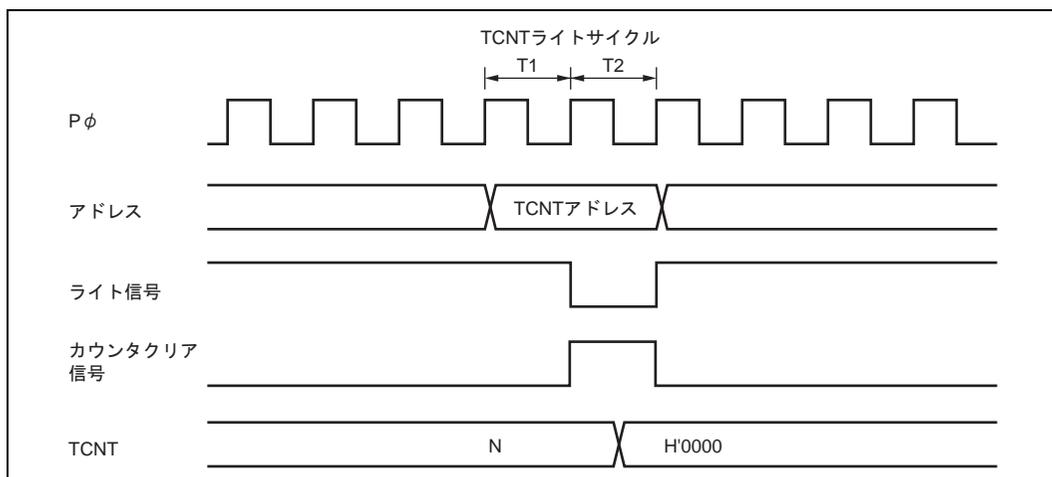


図 16.45 TCNT のライトとクリアの競合

### 16.7.4 TCNT のライトとカウントアップの競合

TCNT のライトサイクル中の T2 ステートで、カウントアップが発生しても、カウントアップされず、TCNT へのライトが優先されます。

このタイミングを図 16.46 に示します。

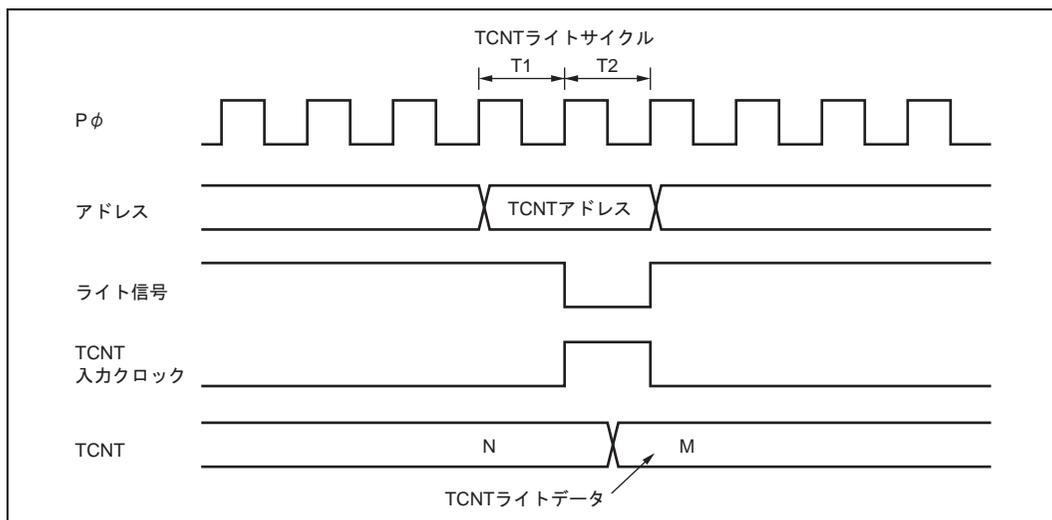


図 16.46 TCNT のライトとカウントアップの競合

### 16.7.5 TGR のライトとコンペアマッチの競合

TGR のライトサイクル中の T2 ステートでコンペアマッチが発生しても、TGR のライトが優先され、コンペアマッチ信号は禁止されます。前回と同じ値を書き込んだ場合でもコンペアマッチは発生しません。

このタイミングを図 16.47 に示します。

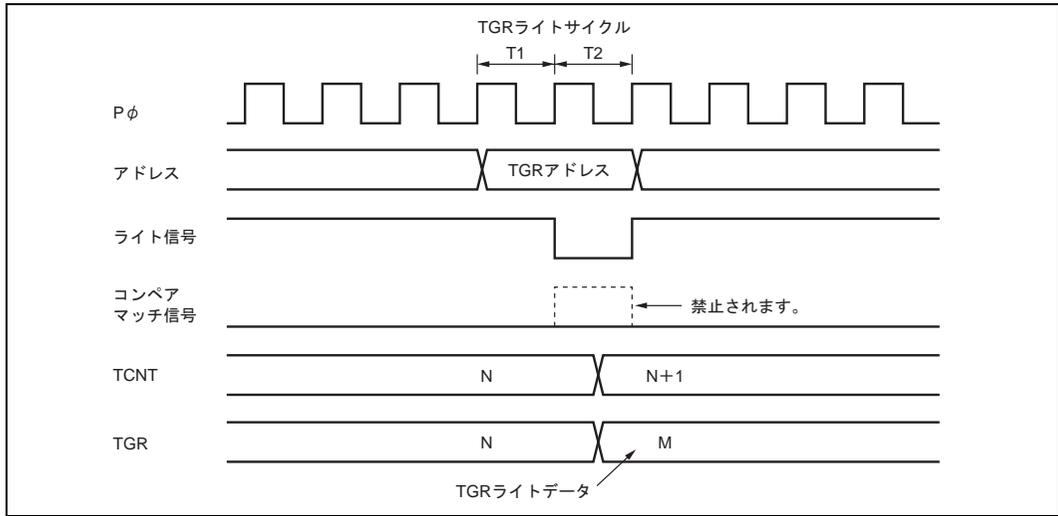


図 16.47 TGR のライトとコンペアマッチの競合

### 16.7.6 バッファレジスタのライトとコンペアマッチの競合

TGR のライトサイクル中の T2 ステートでコンペアマッチが発生すると、バッファ動作によって TGR に転送されるデータはライトデータとなります。

このタイミングを図 16.48 に示します。

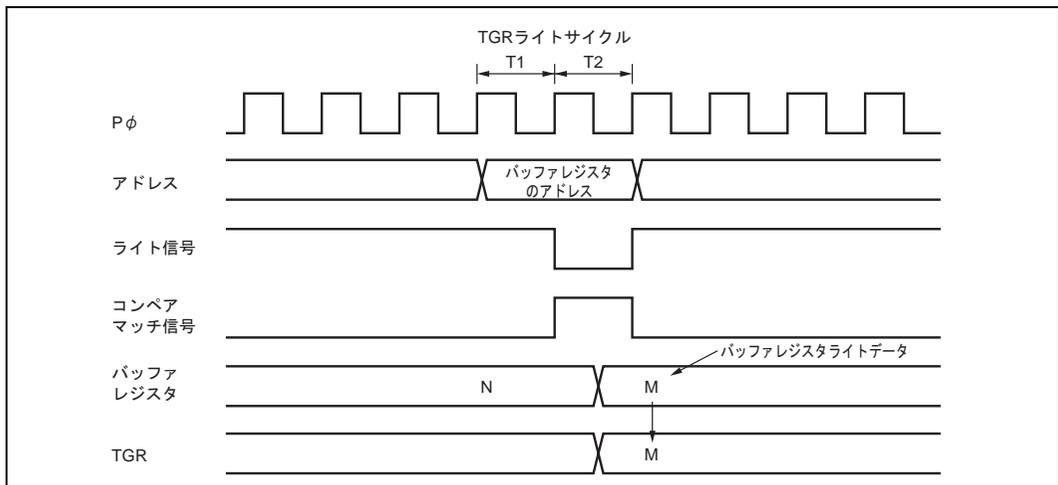


図 16.48 バッファレジスタのライトとコンペアマッチの競合

### 16.7.7 TGR のリードとインプットキャプチャの競合

TGR のリードサイクル中の T1 ステートでインプットキャプチャ信号が発生すると、読み出されるデータはインプットキャプチャ転送前のデータとなります。

このタイミングを図 16.49 に示します。

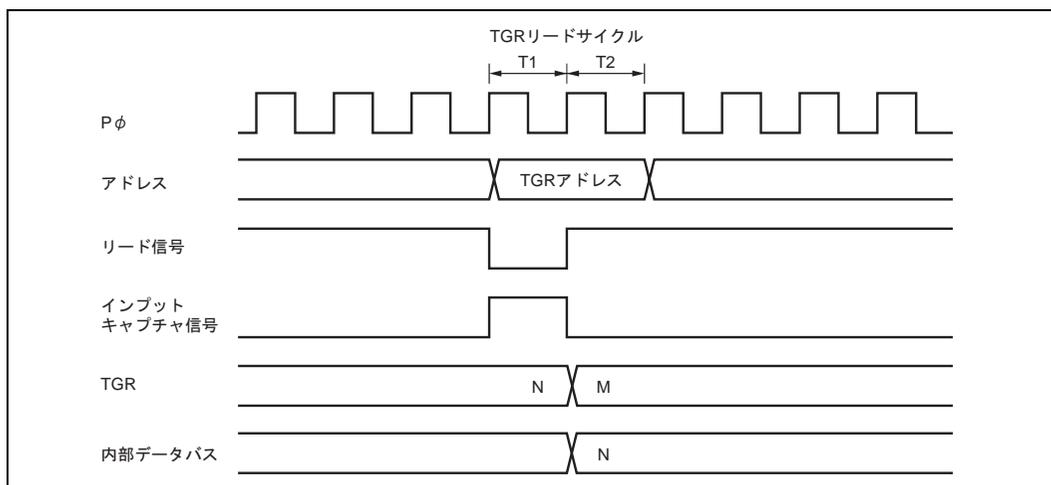


図 16.49 TGR のリードとインプットキャプチャの競合

### 16.7.8 TGR のライトとインプットキャプチャの競合

TGR のライトサイクル中の T2 ステートでインプットキャプチャ信号が発生すると、TGR へのライトは行われず、インプットキャプチャが優先されます。

このタイミングを図 16.50 に示します。

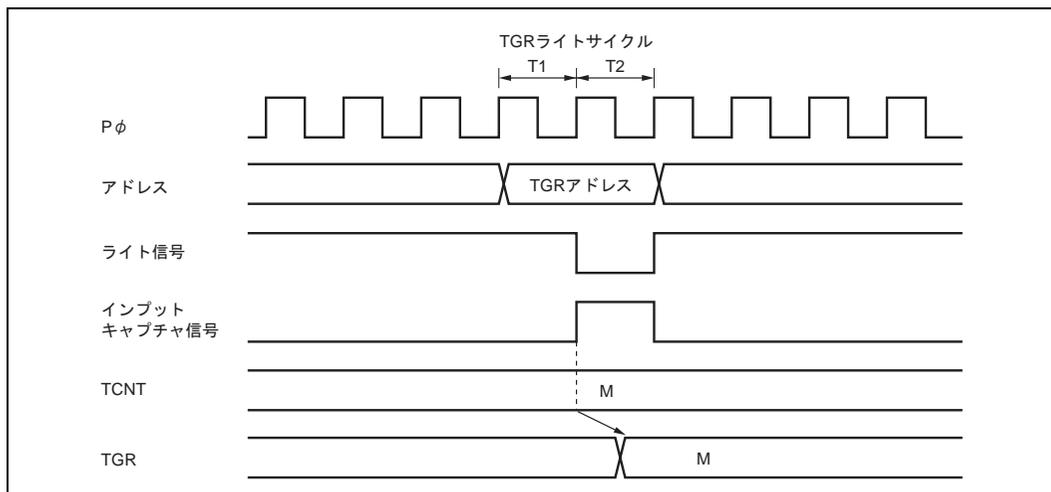


図 16.50 TGR のライトとインプットキャプチャの競合

### 16.7.9 バッファレジスタのライトとインプットキャプチャの競合

バッファレジスタへのライトサイクル中の T2 ステートでインプットキャプチャ信号が発生すると、バッファレジスタへのライトは行われず、バッファ動作が優先されます。

このタイミングを図 16.51 に示します。

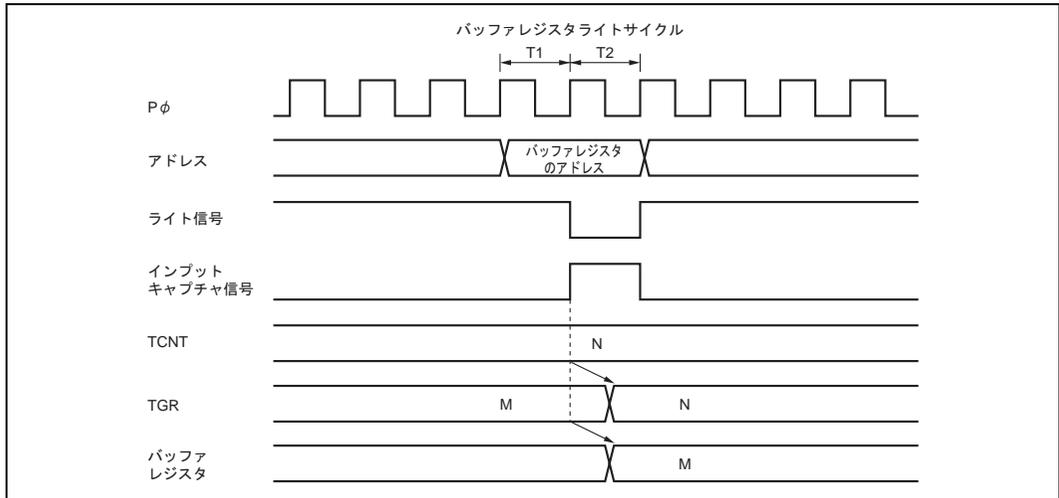


図 16.51 バッファレジスタのライトとインプットキャプチャの競合

### 16.7.10 オーバフロー / アンダフローとカウンタクリアの競合

オーバフロー / アンダフローとカウンタクリアが同時に発生すると、TSR の TCFV / TCFU フラグはセットされず、TCNT のクリアが優先されます。

TGR のコンペアマッチをクリア要因とし、TGR に H'FFFF を設定した場合の動作タイミングを図 16.52 に示します。

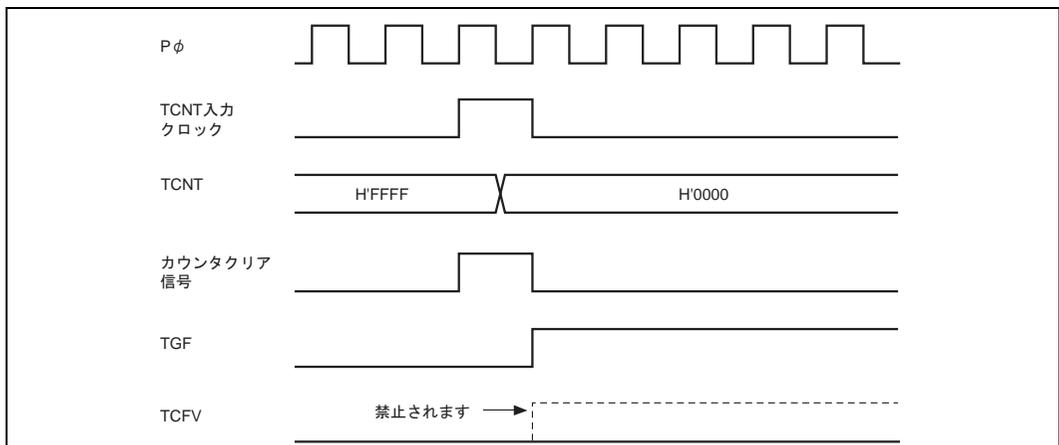


図 16.52 オーバフローとカウンタクリアの競合

## 16.7.11 TCNT のライトとオーバーフロー / アンダフローの競合

TCNT のライトサイクル中の T2 ステートで、カウントアップ / カウントダウンが発生し、オーバーフロー / アンダフローが発生しても、TCNT へのライトが優先され、TSR の TCFV / TCFU フラグはセットされません。

TCNT のライトとオーバーフロー競合時の動作タイミングを図 16.53 に示します。

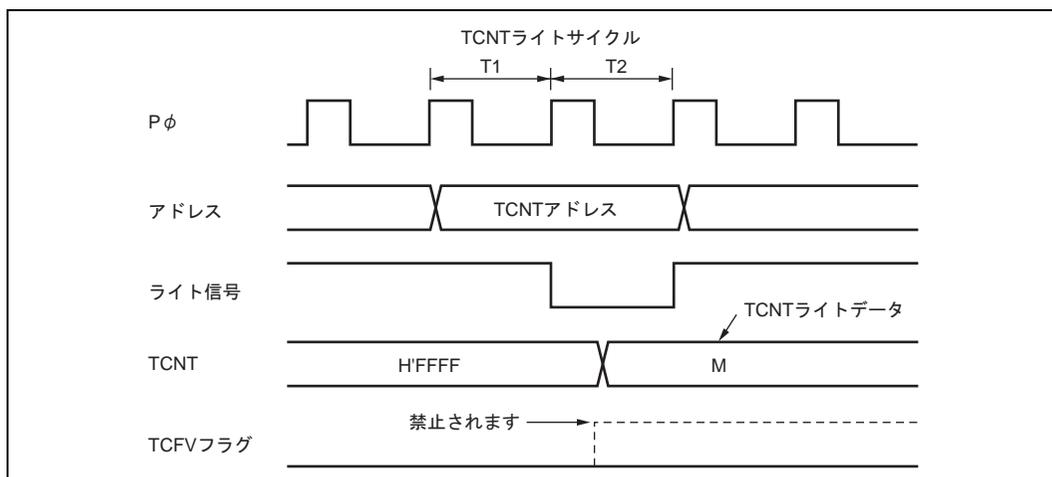


図 16.53 TCNT のライトとオーバーフローの競合

## 16.7.12 入出力端子の兼用

本 LSI では、TCLKA 入力と TIOCC0 入出力、TCLKB 入力と TIOCD0 入出力、TCLKC 入力と TIOCB1 入出力、TCLKD 入力と TIOCB2 入出力の端子がそれぞれ兼用になっています。外部クロックを入力する場合には、兼用されている端子からコンペアマッチ出力を行わないでください。

## 16.7.13 モジュールストップ時の割り込み

割り込みが要求された状態でモジュールストップすると、CPU の割り込み要因または DMAC の起動要因のクリアができません。事前に割り込みをディスエーブルにするなどしてから、モジュールストップモードとしてください。

## 16.8 使用上の注意事項

### 16.8.1 TSR0 ~ 2 のフラグクリア

TSR0 の TCFV、TGFD、TGFC、TGFB、TGFA、および TSR1、2 の TCFU、TCFV、TGFB、TGFA ビットをクリアした際、フラグはクリアされますが、内部論理の割り込み要求がクリアできない場合があります。このとき、割り込み受け付け可能であれば、再度割り込みが発生します。TSR0 ~ 2 のフラグをクリアする場合、以下いずれかの対応をしてください。

- (1) TPU のタイマカウントアップ中にクリアを実施してください。
- (2) TPU のタイマカウントアップ停止時にクリアする際は、クリア実施後、再度フラグへ 0 ライトを実施してください。

### 16.8.2 TPU0 による DMA 転送

TPU チャンネル 0 のコンペアマッチもしくはインプットキャプチャによる DMA 転送を行う場合、内部論理の割り込み要求 (転送要求) が正しくクリアできない場合があります。このため、次に TPU チャンネル 0 のコンペアマッチもしくはインプットキャプチャによる転送要求が発生しても DMA 転送が実行できない場合があります。TPU チャンネル 0 のコンペアマッチもしくはインプットキャプチャによる DMA 転送を行う場合、以下いずれかの対応をしてください。

- (1) DMA 転送元もしくは転送先に内蔵 RAM を設定しないでください。
- (2) DMA 転送元もしくは転送先に内蔵 RAM が設定されていない場合、TPU チャンネル 0 がタイマカウントアップしている状態で実施してください。



---

## 17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

---

### 17.1 概要

ハイパフォーマンスユーザデバッグインタフェース (H-UDI) は、データ転送と割り込み要求、およびバウンダリスキャンの機能を備えています。H-UDI は、外部信号の制御でシリアル転送を行います。

#### 17.1.1 特長

H-UDI は IEEE 1149.1 規格に対応した次の特長を持っています。

- 5本のテスト信号 (TCK、TDI、TDO、TMS、および  $\overline{\text{TRST}}$ )
- TAP コントローラ
- インストラクションレジスタ
- データレジスタ
- バイパスレジスタ
- バウンダリスキャンレジスタ

H-UDI は 7つのインストラクションを備えています。

- BYPASS モード  
IEEE 1149.1 に対応したテストモード
- EXTEST モード  
IEEE 1149.1 に対応したテストモード
- SAMPLE/PRELOAD モード  
IEEE 1149.1 に対応したテストモード
- CLAMP モード  
IEEE 1149.1 に対応したテストモード
- HIGHZ モード  
IEEE 1149.1 に対応したテストモード
- IDCODE モード  
IEEE 1149.1 に対応したテストモード
- H-UDI 割り込み  
INTC への H-UDI 割り込みを要求

17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

17.1.2 H-UDI ブロック図

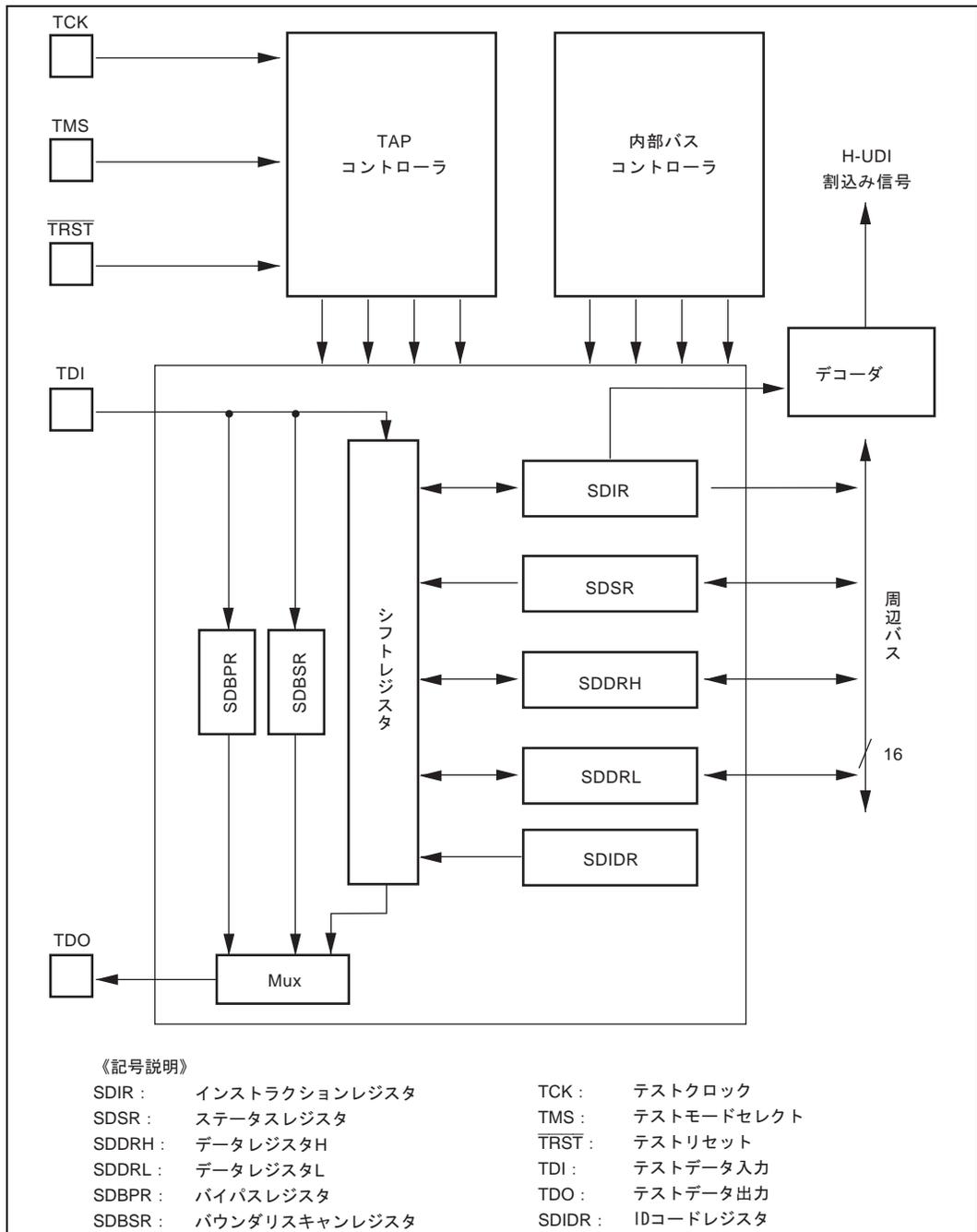


図 17.1 H-UDI ブロック図

### 17.1.3 端子構成

表 17.1 に H-UDI の端子構成を示します。

表 17.1 端子構成

名称	略称	入出力	機能
テストクロック	TCK	入力	テストクロック入力
テストモードセレクト	TMS	入力	テストモードセレクト入力信号
テストデータ入力	TDI	入力	シリアルデータ入力
テストデータ出力	TDO	出力	シリアルデータ出力
テストリセット	TRST	入力	テストリセット入力信号

### 17.1.4 レジスタ構成

表 17.2 に H-UDI のレジスタを示します。

表 17.2 レジスタ構成

レジスタ	略称	R/W <sup>*1</sup>	初期値 <sup>*2</sup>	アドレス	アクセスサイズ (ビット)
インストラクションレジスタ	SDIR	R	H'E000	H'FFFFFFCB0	8 / 16 / 32
ステータスレジスタ	SDSR	R/W	H'0401	H'FFFFFFCB2	8 / 16
データレジスタ H	SDDRH	R/W	不定	H'FFFFFFCB4	8 / 16 / 32
データレジスタ L	SDDRL	R/W	不定	H'FFFFFFCB6	8 / 16
バイパスレジスタ	SDBPR	—	—	—	—
バウンダリスキャンレジスタ	SDBSR	—	—	—	—
ID コードレジスタ	SDIDR	—	H'0101000F	—	—

【注】 \*1 CPU による読み出し / 書き込みが可能かどうかを示します。

\*2 TRST 信号入力時の初期値。リセット (パワーオンリセット / マニュアルリセット) またはスタンバイモードでは初期化されません。

インストラクションとデータは、テストデータ入力端子 (TDI) からシリアル転送によりインストラクションレジスタ (SDIR) とデータレジスタ (SDDR) へ入力できます。SDIR、ステータスレジスタ (SDSR)、SDDR からのデータはテストデータ出力端子 (TDO) を通じて出力できます。バイパスレジスタ (SDBPR) は 1 ビットのレジスタで、BYPASS モード、CLAMP モード、および HIGHZ モード時、TDI と TDO はこのレジスタに接続されます。また、バウンダリスキャンレジスタ (SDBSR) は 330 ビットのレジスタで、SAMPLE/PRELOAD モード、および EXTEST モード時 TDI と TDO はこのレジスタに接続されます。ID コードレジスタ (SDIDR) は 32 ビットのレジスタで IDCODE モード時、TDO を通じて固定コードが出力できます。SDBPR、SDBSR、SDIDR を除くすべてのレジスタは CPU によるアクセスが可能です。

表 17.3 に H-UDI の各レジスタで可能なシリアル転送の種類を示します。

表 17.3 H-UDI レジスタのシリアル転送

レジスタ	シリアル入力	シリアル出力
SDIR	可能	可能
SDSR	不可	可能
SDDRH	可能	可能
SDDRL	可能	可能
SDBPR	可能	可能
SDBSR	可能	可能
SDIDR	不可	可能

## 17.2 外部信号

### 17.2.1 テストクロック (TCK)

テストクロック端子 (TCK) は、H-UDI に独立にクロックを供給します。TCK への入力クロックはそのまま H-UDI へ供給しているため、デューティ比 50% に近いクロック波形を入力してください (詳細については「21. 電気的特性」を参照してください)。何も入力されないと TCK は内部プルアップにより 1 に固定されます。

### 17.2.2 テストモードセレクト (TMS)

テストモードセレクト端子 (TMS) は、TCK の立ち上がりでサンプリングされます。TMS は TAP コントローラの内部状態を制御します。何も入力されないと TMS は内部プルアップにより 1 に固定されます。

### 17.2.3 テストデータ入力 (TDI)

テストデータ入力端子 (TDI) は、H-UDI レジスタに対するインストラクションとデータのシリアル入力を行います。TDI は TCK の立ち上がりでサンプリングされます。何も入力されないと TDI は内部プルアップにより 1 に固定されます。

### 17.2.4 テストデータ出力 (TDO)

テストデータ出力端子 (TDO) は H-UDI レジスタからのインストラクションとデータのシリアル出力を行います。転送は TCK に同期して行われます。TDO は出力していない場合、ハイインピーダンス状態です。

### 17.2.5 テストリセット ( $\overline{\text{TRST}}$ )

テストリセット端子 ( $\overline{\text{TRST}}$ ) は H-UDI を非同期に初期化する信号です。何も入力されないと  $\overline{\text{TRST}}$  は内部プルアップにより 1 に固定されます。

## 17.3 レジスタ

### 17.3.1 インストラクションレジスタ (SDIR)

ビット	15	14	13	12	11	10	9	8
	TS3	TS2	TS1	TS0	—	—	—	—
初期値	1	1	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
ビット	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
初期値	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

インストラクションレジスタ (SDIR) は、CPU による読み出し専用の 16 ビットのレジスタです。H-UDI のインストラクションは、TDI からのシリアル入力によって SDIR に転送することができます。SDIR は  $\overline{\text{TRST}}$  信号によって初期化できますが、リセットまたはスタンバイモードでは初期化されません。

SDIR で定義されるインストラクションの長さは、4 ビットです。4 ビットを越えるインストラクションを入力すると SDIR にはシリアルデータの最後の 4 ビットを格納します。

本レジスタに予約となっているインストラクションをセットした場合の動作は保証しません。

- ・ ビット 15 ~ 12 : テストセットビット (TS3 ~ TS0)  
表 17.4 にインストラクション構成を示します。

表 17.4 インストラクション構成

ビット 15	ビット 14	ビット 13	ビット 12	インストラクション
TS3	TS2	TS1	TS0	
0	0	0	0	EXTEST モード
0	0	0	1	予約
0	0	1	0	CLAMP モード
0	0	1	1	HIGHZ モード
0	1	0	0	SAMPLE/PRELOAD モード
0	1	0	1	予約
0	1	1	0	予約
0	1	1	1	予約
1	0	0	0	予約
1	0	0	1	予約
1	0	1	0	H-UDI 割り込み
1	0	1	1	予約
1	1	0	0	予約
1	1	0	1	予約
1	1	1	0	IDCODE モード (初期値)
1	1	1	1	BYPASS モード

- ・ ビット 11 ~ 0 : 予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### 17.3.2 ステータスレジスタ (SDSR)

ビット	: 15	14	13	12	11	10	9	8
	: —	—	—	—	—	—	—	—
初期値	: 0	0	0	0	0	1	0	0
R/W	: R	R	R	R	R	R	R	R
ビット	: 7	6	5	4	3	2	1	0
	: —	—	—	—	—	—	—	SDTRF
初期値	: 0	0	0	0	0	0	0	1
R/W	: R	R	R	R	R	R	R	R/W

ステータスレジスタ (SDSR) は、CPU による読み出し / 書き込み可能な 16 ビットレジスタです。SDSR は TDO から出力可能ですが、シリアルデータは TDI を通じて SDSR に書き込むことはできません。SDTRF ビットは 1 ビットシフトによって出力されます。2 ビットシフトの場合、SDTRF ビットがまず出力され、続いて予約ビットが出力されます。

SDSR は  $\overline{\text{TRST}}$  信号入力によって初期化されますが、リセットまたはスタンバイモードでは初期化されません。

- ・ ビット 15 ~ 1 : 予約ビット

ビット 15 ~ 11、9 ~ 1 は、読み出すと常に 0 が読み出されます。書き込み値は常に 0 にしてください。

ビット 10 は読み出すと常に 1 が読み出されます。書き込み値は常に 1 にしてください。

- ・ ビット 0 : シリアルデータ転送制御フラグ (SDTRF)

H-UDI レジスタに対して CPU からアクセスできるかどうかを示します。SDTRF ビットは  $\overline{\text{TRST}}$  信号によってリセットされますが、リセットまたはスタンバイモードでは初期化されません。

ビット 0	説明
SDTRF	
0	SDDR へのシリアル転送終了。SDDR へのアクセス可能
1	SDDR へのシリアル転送中。 (初期値)

### 17.3.3 データレジスタ (SDDR)

データレジスタ (SDDR) は、データレジスタ H (SDDRH) とデータレジスタ L (SDDRL) から構成され、それぞれのレジスタは次のような構成を持っています。

ビット	:	15	14	13	12	11	10	9	8
初期値	:	—	—	—	—	—	—	—	—
R/W	:	R/W							
ビット	:	7	6	5	4	3	2	1	0
初期値	:	—	—	—	—	—	—	—	—
R/W	:	R/W							

SDDRH と SDDRL は、CPU による読み出し / 書き込み可能な 16 ビットレジスタです。SDDR は外部とのシリアルデータの転送のため TDO および TDI に接続されます。

32 ビットデータはシリアルデータ転送時に入力および出力されます。32 ビットより大きなデータが入力されると SDDR には最後の 32 ビットのみ格納されます。シリアルデータは SDDR の MSB (SDDRH の 15 ビット) から入力され、LSB (SDDRL の 0 ビット) から出力されます。

このレジスタはリセットまたはスタンバイモード時および  $\overline{\text{TRST}}$  信号によって初期化されません。

#### 17.3.4 バイパスレジスタ (SDBPR)

バイパスレジスタ (SDBPR) は 1 ビットシフトレジスタです。BYPASS モード、CLAMP モード、および HIGHZ モードでは、SDBPR は TDI と TDO の間に接続されます。SDBPR は CPU による読み出し / 書き込みは行えません。

#### 17.3.5 バウンダリスキャンレジスタ (SDBSR)

バウンダリスキャンレジスタ (SDBSR) は、本 LSI の入出力端子の制御を行うために PAD 上に配置されたシフトレジスタです。

EXTEST モード、および SAMPLE/PRELOAD モードを用いて、IEEE1149.1 規格に準拠したバウンダリスキャンテストを行うことができます。

また、SDBSR は CPU による読み出し / 書き込みは行えません。

表 17.5 に本 LSI の端子とバウンダリスキャンレジスタの対応を示します

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
from TDI			
34	D0	入力	329
		出力	328
		出力イネーブル	327
36	D1	入力	326
		出力	325
		出力イネーブル	324

17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
37	D2	入力	323
		出力	322
		出カイネーブル	321
38	D3	入力	320
		出力	319
		出カイネーブル	318
39	D4	入力	317
		出力	316
		出カイネーブル	315
40	D5	入力	314
		出力	313
		出カイネーブル	312
41	D6	入力	311
		出力	310
		出カイネーブル	309
43	D7	入力	308
		出力	307
		出カイネーブル	306
44	D8	入力	305
		出力	304
		出カイネーブル	303
46	D9	入力	302
		出力	301
		出カイネーブル	300
47	D10	入力	299
		出力	298
		出カイネーブル	297
48	D11	入力	296
		出力	295
		出カイネーブル	294
49	D12	入力	293
		出力	292
		出カイネーブル	291
51	D13	入力	290
		出力	289
		出カイネーブル	288
53	D14	入力	287
		出力	286
		出カイネーブル	285
54	D15	入力	284
		出力	283
		出カイネーブル	282

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
55	D16	入力	281
		出力	280
		出力イネーブル	279
56	D17	入力	278
		出力	277
		出力イネーブル	276
57	D18	入力	275
		出力	274
		出力イネーブル	273
59	D19	入力	272
		出力	271
		出力イネーブル	270
62	D20	入力	269
		出力	268
		出力イネーブル	267
63	D21	入力	266
		出力	265
		出力イネーブル	264
64	D22	入力	263
		出力	262
		出力イネーブル	261
65	D23	入力	260
		出力	259
		出力イネーブル	258
68	D24	入力	257
		出力	256
		出力イネーブル	255
70	D25	入力	254
		出力	253
		出力イネーブル	252
71	D26	入力	251
		出力	250
		出力イネーブル	249
72	D27	入力	248
		出力	247
		出力イネーブル	246
73	D28	入力	245
		出力	244
		出力イネーブル	243
74	D29	入力	242
		出力	241
		出力イネーブル	240

17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
75	D30	入力	239
		出力	238
		出力イネーブル	237
77	D31	入力	236
		出力	235
		出力イネーブル	234
80	A0	出力	233
		出力イネーブル	232
82	A1	出力	231
		出力イネーブル	230
83	A2	出力	229
		出力イネーブル	228
84	A3	出力	227
		出力イネーブル	226
85	A4	出力	225
		出力イネーブル	224
86	A5	出力	223
		出力イネーブル	222
87	A6	出力	221
		出力イネーブル	220
88	A7	出力	219
		出力イネーブル	218
90	A8	出力	217
		出力イネーブル	216
92	A9	出力	215
		出力イネーブル	214
93	A10	出力	213
		出力イネーブル	212
94	A11	出力	211
		出力イネーブル	210
95	A12	出力	209
		出力イネーブル	208
96	A13	出力	207
		出力イネーブル	206
97	A14	出力	205
		出力イネーブル	204
98	A15	出力	203
		出力イネーブル	202
100	A16	出力	201
		出力イネーブル	200
102	A17	出力	199
		出力イネーブル	198

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
103	A18	出力	197
		出力イネーブル	196
104	A19	出力	195
		出力イネーブル	194
105	A20	出力	193
		出力イネーブル	192
106	A21	出力	191
		出力イネーブル	190
107	A22	出力	189
		出力イネーブル	188
108	A23	出力	187
		出力イネーブル	186
111	A24	出力	185
		出力イネーブル	184
115	WAIT	入力	183
117	RAS	出力	182
		出力イネーブル	181
118	CAS	出力	180
		出力イネーブル	179
119	DQM $\overline{U}$ 3	出力	178
		出力イネーブル	177
120	DQM $\overline{L}$ 2	出力	176
		出力イネーブル	175
121	DQM $\overline{L}$ 1	出力	174
		出力イネーブル	173
122	DQM $\overline{L}$ 0	出力	172
		出力イネーブル	171
123	CAS3	出力	170
		出力イネーブル	169
124	CAS2	出力	168
		出力イネーブル	167
125	CAS1	出力	166
		出力イネーブル	165
126	CAS0	出力	164
		出力イネーブル	163
127	CKE	出力	162
		出力イネーブル	161
128	RD	出力	160
		出力イネーブル	159
129	REFOUT	出力	158
		出力イネーブル	157
131	$\overline{BS}$	出力	156
		出力イネーブル	155

17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
133	RD/W $\bar{R}$	出力	154
		出力イネーブル	153
134	CS $\bar{0}$	出力	152
		出力イネーブル	151
135	CS $\bar{1}$	出力	150
		出力イネーブル	149
136	CS $\bar{2}$	出力	148
		出力イネーブル	147
137	CS $\bar{3}$	出力	146
		出力イネーブル	145
138	CS $\bar{4}$	出力	144
		出力イネーブル	143
139	BUSHIZ	入力	142
140	BH	出力	141
		出力イネーブル	140
141	DREQ1	入力	139
142	DREQ0	入力	138
143	DACK1	出力	137
		出力イネーブル	136
144	DACK0	出力	135
		出力イネーブル	134
145	BRLS	入力	133
148	BGR	出力	132
		出力イネーブル	131
151	PB15	入力	130
		出力	129
		出力イネーブル	128
152	PB14	入力	127
		出力	126
		出力イネーブル	125
153	PB13	入力	124
		出力	123
		出力イネーブル	122
154	PB12	入力	121
		出力	120
		出力イネーブル	119
156	PB11	入力	118
		出力	117
		出力イネーブル	116
158	PB10	入力	115
		出力	114
		出力イネーブル	113

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
159	PB9	入力	112
		出力	111
		出力イネーブル	110
160	PB8	入力	109
		出力	108
		出力イネーブル	107
161	PB7	入力	106
		出力	105
		出力イネーブル	104
162	PB6	入力	103
		出力	102
		出力イネーブル	101
163	PB5	入力	100
		出力	99
		出力イネーブル	98
164	PB4	入力	97
		出力	96
		出力イネーブル	95
165	PB3	入力	94
		出力	93
		出力イネーブル	92
166	PB2	入力	91
		出力	90
		出力イネーブル	89
168	PB1	入力	88
		出力	87
		出力イネーブル	86
170	PB0	入力	85
		出力	84
		出力イネーブル	83
171	PA13	入力	82
		出力	81
		出力イネーブル	80
172	PA12	入力	79
		出力	78
		出力イネーブル	77
173	PA11	入力	76
		出力	75
		出力イネーブル	74
174	PA10	入力	73
		出力	72
		出力イネーブル	71

17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
175	PA9	入力	70
		出力	69
		出力イネーブル	68
176	PA8	入力	67
		出力	66
		出力イネーブル	65
177	PA7	入力	64
		出力	63
		出力イネーブル	62
178	PA6	入力	61
		出力	60
		出力イネーブル	59
180	PA5	入力	58
		出力	57
		出力イネーブル	56
182	PA4	入力	55
		出力	54
		出力イネーブル	53
183	CKPO	出力	52
		出力イネーブル	51
184	PA2	入力	50
		出力	49
		出力イネーブル	48
185	PA1	入力	47
		出力	46
		出力イネーブル	45
186	PA0	入力	44
		出力	43
		出力イネーブル	42
187	RX-ER	入力	41
188	RX-DV	入力	40
189	COL	入力	39
190	CRS	入力	38
192	RX-CLK	入力	37
194	ERXD0	入力	36
195	ERXD1	入力	35
196	ERXD2	入力	34
197	ERXD3	入力	33
198	MDIO	入力	32
		出力	31
		出力イネーブル	30
199	MDC	出力	29
		出力イネーブル	28

表 17.5 端子とバウンダリスキャンレジスタの対応

端子 NO.	端子名	入出力	ビット NO.
201	TX-CLK	入力	27
203	TX-EN	出力	26
		出力イネーブル	25
204	ETXD0	出力	24
		出力イネーブル	23
205	ETXD1	出力	22
		出力イネーブル	21
206	ETXD2	出力	20
		出力イネーブル	19
207	ETXD3	出力	18
		出力イネーブル	17
208	TX-ER	出力	16
		出力イネーブル	15
1	IRL3	入力	14
2	IRL2	入力	13
3	IRL1	入力	12
4	IRL0	入力	11
5	NMI	入力	10
13	MD4	入力	9
14	MD3	入力	8
15	MD2	入力	7
16	MD1	入力	6
17	MD0	入力	5
24	CKPREQ/CKM	入力	4
25	CKPACK	出力	3
		出力イネーブル	2
27	IVECF	出力	1
		出力イネーブル	0
to TDO			

【注】 出力イネーブルは Low アクティブの信号です。出力イネーブルを Low にすることで該当端子をドライブします。ただし、MDIO 端子のみ出力イネーブルは High アクティブの信号です。

### 17.3.6 ID コードレジスタ (SDIDR)

ID コードレジスタ (SDIDR) は、32 ビットのレジスタです。

IDCODE モード時、SDIDR は TDO から固定コードである H'0101000F を出力可能ですが、シリアルデータは TDI を通じて SDIDR に書き込むことはできません。

また、SDIDR は CPU による読み出し / 書き込みは行えません。

31 28	27	12	11	1	0
0000	0001 0000 0001 0000		0000 0000 111		1
Version (4 ビット)	Part Number (16 ビット)		Manufacture Identify (11 ビット)		固定コード (1 ビット)

## 17.4 動作

### 17.4.1 TAP コントローラ

図 17.2 に TAP コントローラの内部状態を示します。IEEE1149.1 で規定されている状態遷移に準拠しています。

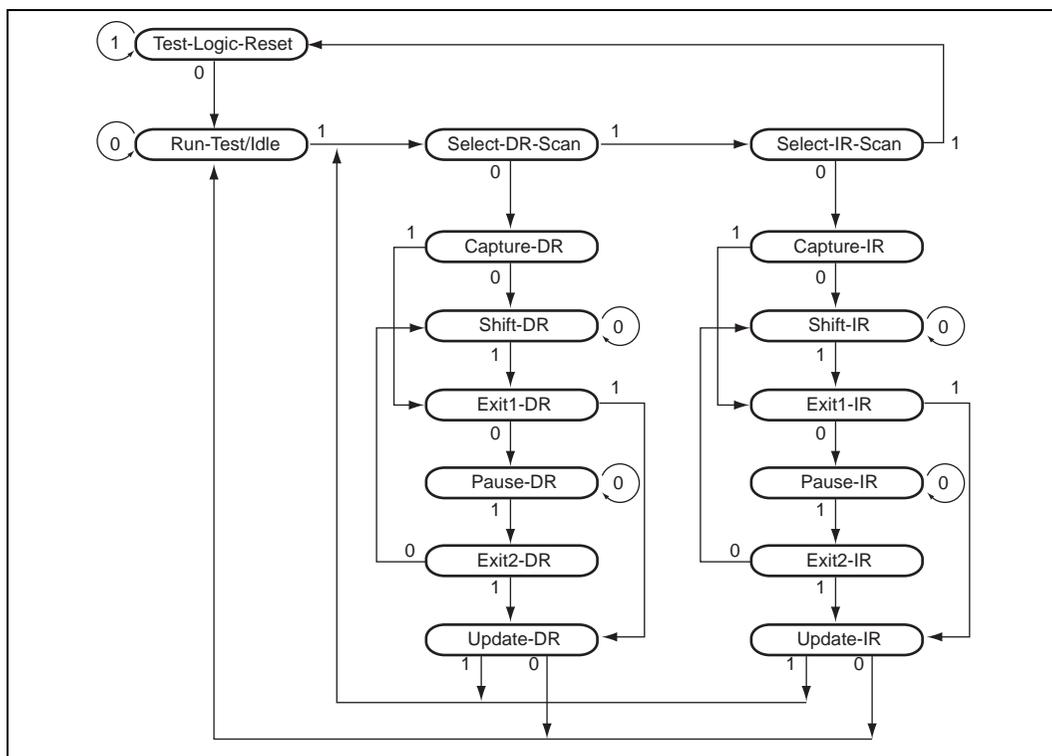


図 17.2 TAP コントローラ状態遷移図

### 17.4.2 H-UDI 割り込みとシリアル転送

SDIR へ H-UDI 割り込みのインストラクションが、TDI を経由して転送されると割り込みが発生します。H-UDI 割り込みのサービスルーチンにより、データ転送を制御することが可能です。転送は SDDR を介して、行うことができます。

外部と H-UDI の間のデータ入出力制御は、外部と内部で SDSR の SDTRF ビットを観測することで行います。内部での SDTRF ビットの観測は、CPU で SDSR を読み出して行うことになります。

H-UDI 割り込みおよびシリアル転送の手順は次のとおりです。

- (1) インストラクションがシリアル転送により SDIR へ入力され、H-UDI 割り込み要求が発生します。
- (2) H-UDI 割り込み要求が発行された後、外部で SDSR の SDTRF ビットを観測します。SDTRF = 1 が TDO から出力されたことを観測した後、SDDR にシリアルデータを転送します。

- (3) SDDR へのシリアル転送が完了すると、SDTRF ビットが 0 にクリアされ、CPU による SDDR へのアクセスが可能となります。SDDR へのアクセス終了後、SDSR の SDTRF ビットを 1 に設定することで、SDDR のシリアル転送が可能となります。
- (4) 外部と内部で、常に SDSA の SDTRF ビットを観測することにより、外部と H-UDI のシリアルデータ転送が可能となります。

図 17.3、図 17.4、図 17.5 に外部と H-UDI 間のデータ転送のタイミングを示します。

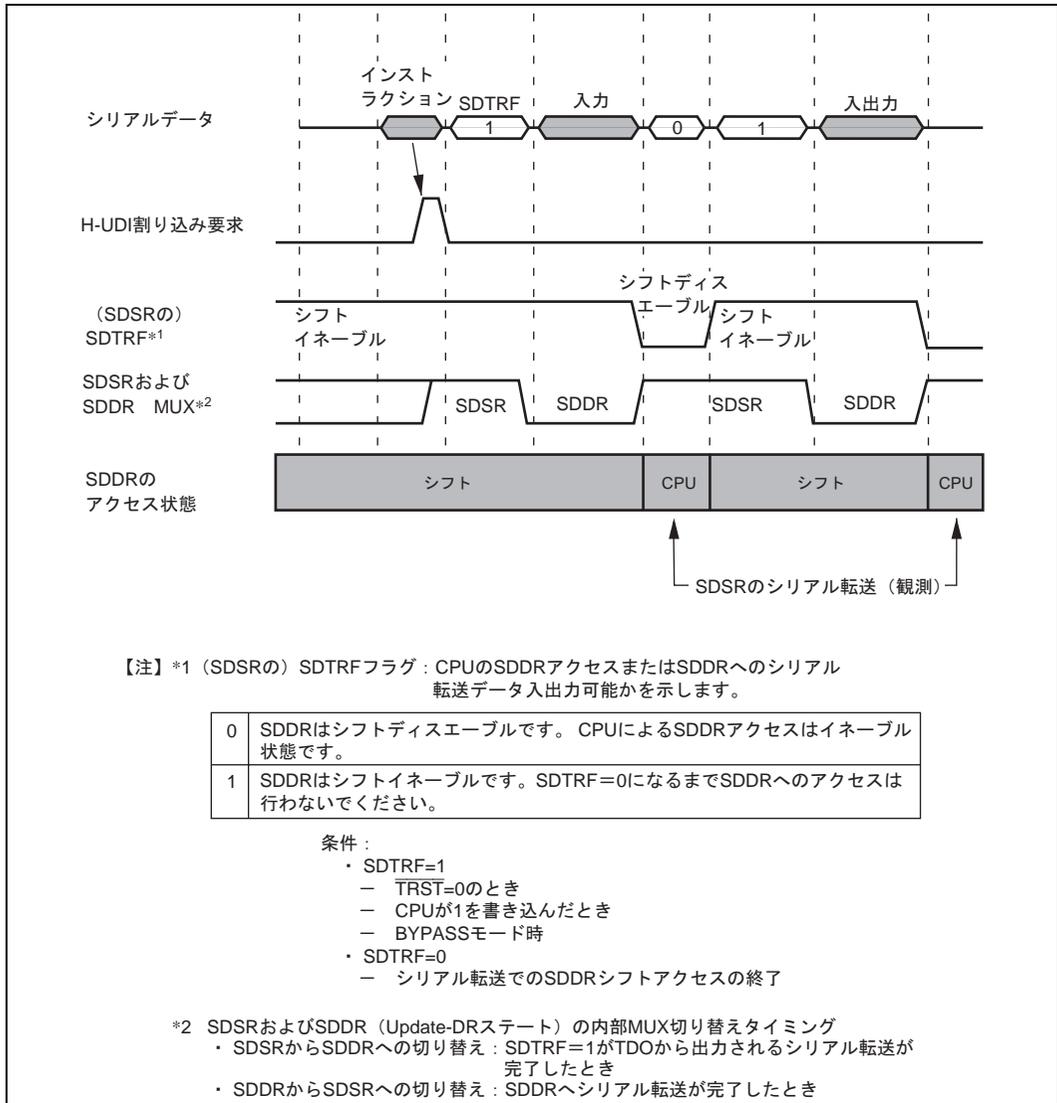


図 17.3 データ入出力タイミングチャート (1)

17. ハイパフォーマンスユーザデバッグインタフェース (H-UDI)

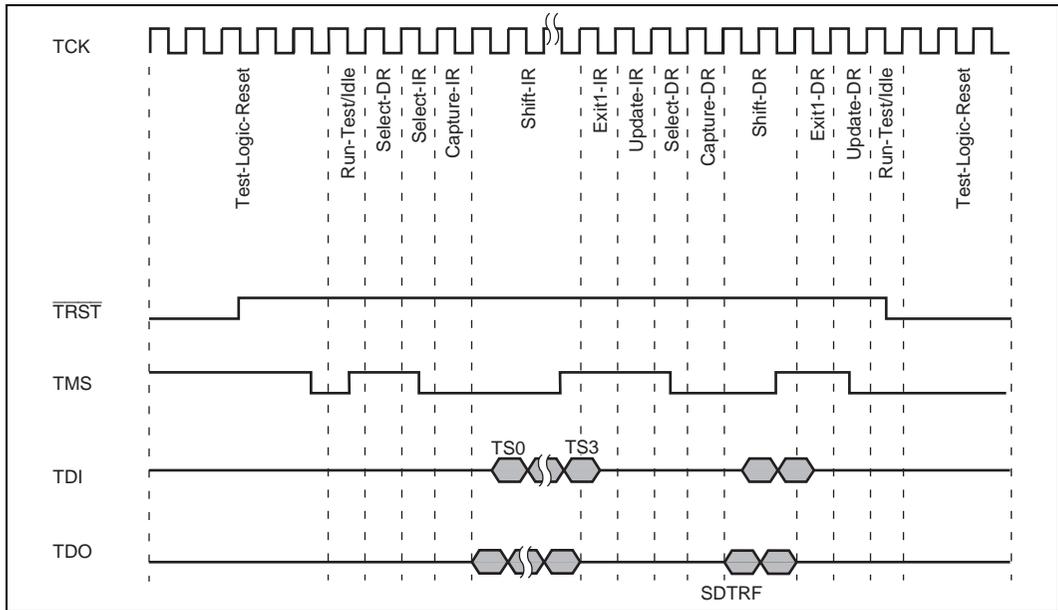


図 17.4 データ入出力タイミングチャート (2)

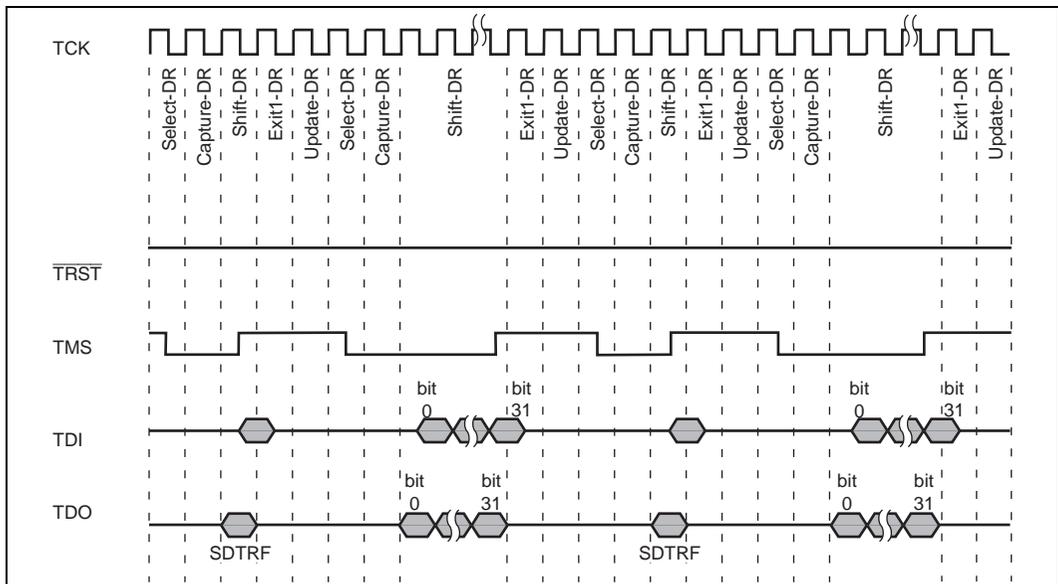


図 17.5 データ入出力タイミングチャート (3)

### 17.4.3 H-UDI のリセット

H-UDI は 2 つの方法でリセットできます。

- ・  $\overline{\text{TRST}}$  信号を 0 に保持すると H-UDI がリセットされます。
- ・ H-UDI は  $\overline{\text{TRST}}=1$  のとき、 $\text{TMS}=1$  の状態で 5 クロック以上の TCK を入力することによってリセットできます。

## 17.5 バウンダリスキャン

SDIR にコマンドを設定することにより、H-UDI 端子を IEEE1149.1 で規定されているバウンダリスキャンモードに設定できます。

### 17.5.1 サポートする命令

IEEE1149.1 で定義される 3 つの必須命令 (BYPASS、SAMPLE/PRELOAD、EXTEST)、およびオプション命令 (CLAMP、HIGHZ、IDCODE) をサポートします。

#### (1) BYPASS

BYPASS 命令は、バイパスレジスタを動作させる必須の標準命令です。この命令はシフトバスを短縮してプリント基板上の他の LSI のシリアルデータを転送高速化するためのものです。この命令の実行中、テスト回路はシステム回路に何も影響を与えません。命令コードは 1111 です。

#### (2) SAMPLE/PRELOAD

SAMPLE/PRELOAD 命令は本 LSI の内部回路からバウンダリスキャンレジスタに値を入力し、スキャンパスから出力したり、スキャンパスにデータをロードする命令です。本命令実行中本 LSI の入力ピンはそのまま内部回路に伝達され、内部回路の値はそのまま出力ピンから外部へ出力されます。本命令の実行により本 LSI のシステム回路は何の影響も受けません。命令コードは 0100 です。

SAMPLE 動作では、入力ピンから内部回路へ転送される値や内部回路から出力ピンへ転送される値のスナップショットをバウンダリスキャンレジスタに取り込み、スキャンパスから読み出します。スナップショットの取り込みは本 LSI の通常動作を妨げずに行われます。

PRELOAD 動作では、EXTEST 命令に先立ちスキャンパスからバウンダリスキャンレジスタの平行出力ラッチに初期値を設定します。PRELOAD 動作がないと、EXTEST 命令を実行するとき、最初のスキャンシーケンスが完了する (出力ラッチへの転送) までの間出力ピンから不定値が出力される (EXTEST 命令では出力ピンに常に平行出力ラッチを出力する) ことになります。

#### (3) EXTEST

本命令では、本 LSI をプリント基板に実装したとき、外部回路をテストするためのものです。本命令の実行時、出力ピンはバウンダリスキャンレジスタからテストデータ (SAMPLE/PRELOAD 命令ですでに設定されています) をプリント基板へ出力するために使用され、入力ピンはプリント基板からバウンダリスキャンレジスタにテスト結果を取り込むために使用されます。EXTEST 命令を N 回用いてテストを行うとき、N 回目のテストデータは (N-1) 回目のスキャンアウトのときにスキャンインされます。

本命令の Capture-DR 状態で出力ピンのバウンダリスキャンレジスタにロードされたデータは外部回路のテストには使用されません。(シフト動作で入れ替えます)。

命令コードは、0000 です。

## (4) CLAMP

CLAMP 命令が選択されると、出力ピンはあらかじめ SAMPLE/PRELOAD 命令によって設定されたバウンダリスキャンレジスタの値を出力します。CLAMP 命令が選択されている間、バウンダリスキャンレジスタの状態は TAP コントローラの状態に関係なく前の状態が保持されます。

TDI、TDO 間にはバイパスレジスタが接続され、BYPASS 命令が選択されたときと同様の動作をします。  
命令コードは 0010 です。

## (5) HIGHZ

HIGHZ 命令が選択されると、すべての出力ピンはハイインピーダンス状態となります。HIGHZ 命令が選択されている間、バウンダリスキャンレジスタの状態は TAP コントローラの状態に関係なく前の状態が保持されず。

TDI、TDO 間にはバイパスレジスタが接続され、BYPASS 命令が選択されたときと同様の動作をします。  
命令コードは 0011 です。

## (6) IDCODE

IDCODE 命令が選択されると、TAP コントローラの Shift-DR 状態時に ID コードレジスタの値を LSB より TDO から出力します。この命令の実行中テスト回路はシステム回路に何も影響を与えません。

TAP コントローラの Test-Logic-Reset 状態時、インストラクションレジスタは IDCODE 命令に初期化されます。  
命令コードは 1110 です。

## 17.5.2 注意事項

- (1) クロック関連信号 (EXTAL、XTAL、CKIO、CAP1、CAP2) はバウンダリスキャンの対象外です。
- (2) リセット関連信号 ( $\overline{\text{RES}}$ 、ASEMODE) はバウンダリスキャンの対象外です。
- (3) H-UDI 関連信号 (TCK、TDI、TDO、TMS、 $\overline{\text{TRST}}$ ) はバウンダリスキャンの対象外です。
- (4) ASEMODE 端子は High 固定としてください。

## 17.6 使用上の注意事項

- (1) H-UDI を起動する / しないにかかわらず、必ず  $\overline{\text{TRST}}$  信号を 0 にしてリセットしてください。この際、 $\overline{\text{TRST}}$  は TCK に対して 20 クロック分、Low レベルに保持してください。詳細は「21. 電気的特性」を参照してください。
- (2) スタンバイモードではレジスタは初期化されません。スタンバイモード時に  $\overline{\text{TRST}}$  を 0 に設定すると IDCODE モードになります。
- (3) TCK の周波数は周辺モジュールクロック (P ) の周波数よりも低くなければなりません。詳細は「21. 電気的特性」を参照してください。
- (4) シリアル転送時のデータ入出力は LSB から開始します。図 17.6 にシリアルデータ入出力を示します。
- (5) TDI、TDO 間に接続されるレジスタのビット数を超過してシリアル転送した場合、レジスタのビット数を超過して TDO から出力されるシリアルデータは、TDI から入力されたデータとなります。
- (6) H-UDI シリアル転送シーケンスがくずれた場合、必ず  $\overline{\text{TRST}}$  のリセットを行ってください。このとき、転送動作にかかわらず、再度転送し直してください。
- (7) TDO の出力タイミングは、IEEE1149.1 で規定する 6 命令を選択したときは TCK の立ち下がりから、その他の命令を選択したときは TCK の立ち上がりからになります。

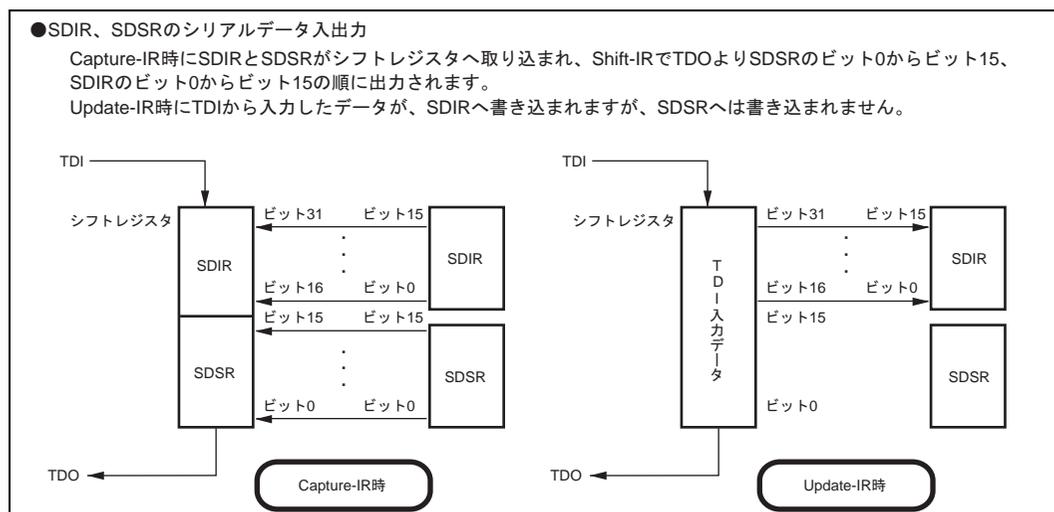
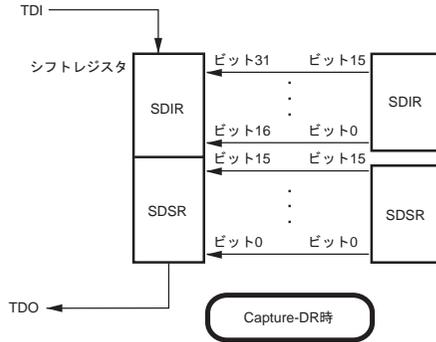


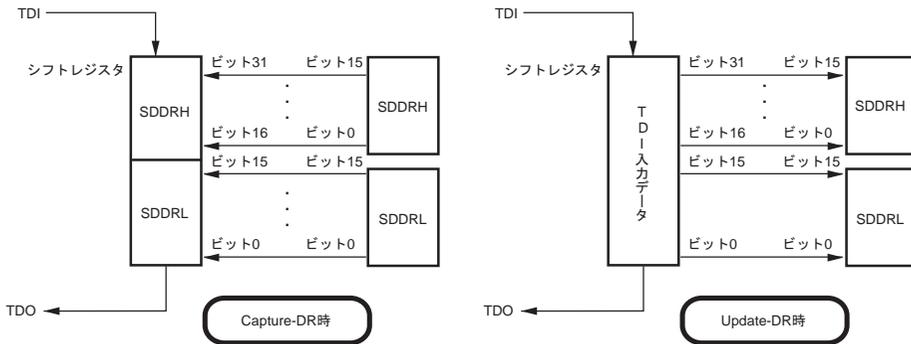
図 17.6 シリアルデータ入出力 (1)

●SDDRH、SDDRLのシリアルデータ入出力

- ① H-UDI割り込みモード時、H-UDI割り込みが発生し、TDOからSDTRF=1が読み出されるまでは、Capture-DRでSDSRとSDIRがシフトレジスタへ取り込まれ、Shift-DRでTDOよりSDSRのビット0からビット15、SDIRのビット0からビット15の順に出力されます。  
Update-DR時、TDI入力データは、どのレジスタへも書き込まれません。



- ② H-UDI割り込みモード時、H-UDI割り込みが発生し、TDOからSDTRF=1が読み出された後、Capture-DRでSDDRHとSDDRLがシフトレジスタへ取り込まれ、Shift-DRでTDOよりSDDRLのビット0からビット15、SDDRHのビット0からビット15の順に出力されます。  
TDIより出力されたデータは、Update-DRでSDDRHとSDDRLへ書き込まれます。



●SDIDRのシリアルデータ入出力

- IDCODEモード時、Capture-DR時にSDIDRがシフトレジスタへ取り込まれ、Shift-DRでTDOよりSDIDRのビット0からビット31の順に出力されます。  
Update-DR時、TDIから入力したデータは、どのレジスタへも書き込まれません。

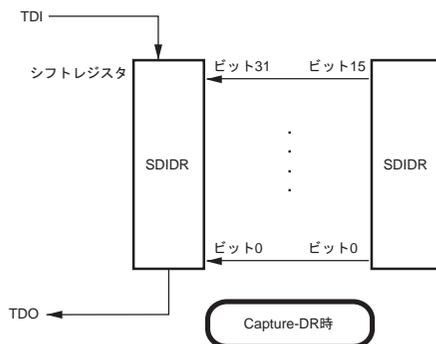


図 17.7 シリアルデータ入出力 (2)

## 17. ハイパフォーマンスユーザデバッグ インタフェース (H-UDI)

---

## 18. ピンファンクションコントローラ (PFC)

### 18.1 概要

ピンファンクションコントローラ (PFC) は、マルチプレクス端子の機能とその入出力の方向を選ぶためのレジスタで構成されています。本 LSI の動作モードとは無関係に、端子の機能とその入出力の方向を 1 本ずつ選ぶことができます。表 18.1 に、本マルチプレクス端子を示します。

表 18.1 マルチプレクス端子一覧

ポート	機能 1 [ 00 ]			機能 2 [ 01 ]			機能 3 [ 10 ]			機能 4 [ 11 ]		
	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール
A	PA13	入出力	ポート	SRCK0	入力	SIO (0)						
A	PA12	入出力	ポート	SRS0	入力	SIO (0)						
A	PA11	入出力	ポート	SRXD0	入力	SIO (0)						
A	PA10	入出力	ポート	STCK0	入力	SIO (0)						
A	PA9	入出力	ポート	STS0	入出力	SIO (0)						
A	PA8	入出力	ポート	STXD0	出力	SIO (0)						
A	WDTO VF	出力	WDT	PA7	入出力	ポート						
A	PA6	入出力	ポート	FTCI	入力	FRT						
A	PA5	入出力	ポート	FTI	入力	FRT						
A	PA4	入出力	ポート	FTOA	出力	FRT						
A	CKPO	出力	ポート	FTOB	出力	FRT						
A	PA2	入出力	ポート	LNKST A	入力	EtherC						
A	PA1	入出力	ポート	EXOUT	出力	EtherC						
A	PA0	入出力	ポート									
B	PB15	入出力	ポート				SCK1	入出力	SCIF (1)			
B	PB14	入出力	ポート				RXD1	入力	SCIF (1)			
B	PB13	入出力	ポート				TXD1	出力	SCIF (1)			
B	PB12	入出力	ポート	SRCK2	入力	SIO (2)	RTS	出力	SCIF (1)	STATS 1	出力	BSC
B	PB11	入出力	ポート	SRS2	入力	SIO (2)	CTS	入力	SCIF (1)	STATS 0	出力	BSC
B	PB10	入出力	ポート	SRXD2	入力	SIO (2)	TIOCA 1	入出力	TPU (1)			

## 18. ピンファンクションコントローラ (PFC)

表 18.1 マルチプレクス端子一覧

ポート	機能 1 [ 00 ]			機能 2 [ 01 ]			機能 3 [ 10 ]			機能 4 [ 11 ]		
	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール	信号名	入出力	関連モジュール
B	PB9	入出力	ポート	STCK2	入力	SIO (2)	TIOCB1	入出力	TPU (1)			
B	PB8	入出力	ポート	STS2	入出力	SIO (2)	TIOCA2	入出力	TPU (2)			
B	PB7	入出力	ポート	STXD2	出力	SIO (2)	TIOCB2	入出力	TPU (2)			
B	PB6	入出力	ポート	SRCK1	入力	SIO (1)	SCK2	入出力	SCIF (2)			
B	PB5	入出力	ポート	SRS1	入力	SIO (1)	RXD2	入力	SCIF (2)			
B	PB4	入出力	ポート	SRXD1	入力	SIO (1)	TXD2	出力	SCIF (2)			
B	PB3	入出力	ポート	STCK1	入力	SIO (1)	TIOCA0	入出力	TPU (0)			
B	PB2	入出力	ポート	STS1	入出力	SIO (1)	TIOCB0	入出力	TPU (0)			
B	PB1	入出力	ポート	STXD1	出力	SIO (1)	TIOCC0	入出力	TPU (0)			
B	PB0	入出力	ポート				TIOCD0	入出力	TPU (0)	WOL	出力	EtherC

【注】 初期状態で機能 1 を選択します。

\* 初期値は入力となります。

[ ] 内の数字は、ポート A [ 0:13 ] およびポート B [ 0:15 ] において、マルチプレクス機能を選択するための PFC における各モードビット (MD1、MD0) の設定を表しています。

## 18.2 レジスタ構成

PFC のレジスタを表 18.2 に示します。

表 18.2 レジスタ構成

名称	略称	R/W	初期値	アドレス	アクセスサイズ
ポート A コントロールレジスタ	PACR	R/W	H'0000	H'FFFFFFC80	8,16
ポート A・IO レジスタ	PAIOR	R/W	H'0000	H'FFFFFFC82	8,16
ポート B コントロールレジスタ	PBCR	R/W	H'0000	H'FFFFFFC88	8,16
ポート B・IO レジスタ	PBIOR	R/W	H'0000	H'FFFFFFC8A	8,16
ポート B コントロールレジスタ 2	PBCR2	R/W	H'0000	H'FFFFFFC8E	8,16

## 18.3 レジスタの説明

### 18.3.1 ポート A コントロールレジスタ (PACR)

ポート A コントロールレジスタ (PACR) は、16 ビットの読み出し / 書き込み可能なレジスタで、ポート A にある 14 本のマルチプレクス端子の機能を選びます。

PACR は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 MD	PA12 MD	PA11 MD	PA10 MD	PA9 MD	PA8 MD	PA7 MD	PA6 MD	PA5 MD	PA4 MD	PA3 MD	PA2 MD	PA1 MD	PA0 MD
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ビット 15、14 : 予約ビット  
読み出すと常に 0 が読み出されます。書き込む値も 0 にしてください。
- ビット 13 : PA13 モードビット (PA13MD)  
PA13 / SRCK0 端子の機能を選びます。

ビット 13	説明
PA13MD	
0	汎用入出力 (PA13) (初期値)
1	SIO0 シリアル受信クロック入力 (SRCK0)

- ビット 12 : PA12 モードビット (PA12MD)  
PA12 / SRS0 端子の機能を選びます。

ビット 12	説明
PA12MD	
0	汎用入出力 (PA12) (初期値)
1	SIO0 シリアル受信同期入力 (SRS0)

- ビット 11 : PA11 モードビット (PA11MD)  
PA11 / SRXD0 端子の機能を選びます。

ビット 11	説明
PA11MD	
0	汎用入出力 (PA11) (初期値)
1	SIO0 シリアル受信データ (SRXD0)

- ビット 10 : PA10 モードビット (PA10MD)

## 18. ピンファンクションコントローラ (PFC)

PA10 / STCK0 端子の機能を選びます。

ビット 10	説 明
PA10MD	
0	汎用入出力 (PA10) (初期値)
1	SIO0 シリアル送信クロック (STCK0)

- ・ ビット 9 : PA9 モードビット (PA9MD)  
PA9 / STS0 端子の機能を選びます。

ビット 9	説 明
PA9MD	
0	汎用入出力 (PA9) (初期値)
1	SIO0 シリアル送信同期入出力 (STS0)

- ・ ビット 8 : PA8 モードビット (PA8MD)  
PA8 / STXD0 端子の機能を選びます。

ビット 8	説 明
PA8MD	
0	汎用入出力 (PA8) (初期値)
1	SIO0 シリアル送信データ出力 (STXD0)

- ・ ビット 7 : PA7 モードビット (PA7MD)  
WDTOVF / PA7 端子の機能を選びます。

ビット 7	説 明
PA7MD	
0	WDT オーバフロー信号出力 (WDTOVF) * (初期値)
1	汎用入出力 (PA7)

【注】 \* WDTOVF は、リセット後出力となりますので、汎用入力ポート (PA7) としてお使いになる場合には、注意してください。

- ・ ビット 6 : PA6 モードビット (PA6MD)  
PA6 / FTCl 端子の機能を選びます。

ビット 6	説 明
PA6MD	
0	汎用入出力 (PA6) (初期値)
1	FRT クロック入力 (FTCl)

- ・ ビット 5 : PA5 モードビット (PA5MD)  
PA5 / FTI 端子の機能を選びます。

ビット 5	説 明
PA5MD	
0	汎用入出力 (PA5) (初期値)
1	FRT インพุットキャプチャ (FTI)

- ・ ビット 4 : PA4 モードビット (PA4MD)  
PA4 / FTOA 端子の機能を選びます。

ビット 4	説 明
PA4MD	
0	汎用入出力 (PA4) (初期値)
1	FRT アウトプットコンペア出力 (FTOA)

- ・ ビット 3 : PA3 モードビット (PA3MD)  
CKPO / FTOB 端子の機能を選びます。

ビット 3	説 明
PA3MD	
0	周辺モジュール用クロック出力 (CKPO) (初期値)
1	FRT アウトプットコンペア出力 (FTOB)

- ・ ビット 2 : PA2 モードビット (PA2MD)  
PA2 / LNKSTA 端子の機能を選びます。

ビット 2	説 明
PA2MD	
0	汎用入出力 (PA2) (初期値)
1	EtherC リンクステータス入力 (LNKSTA)

- ・ ビット 1 : PA1 モードビット (PA1MD)  
PA1 / EXOUT 端子の機能を選びます。

ビット 1	説 明
PA1MD	
0	汎用入出力 (PA1) (初期値)
1	EtherC 汎用外部出力 (EXOUT)

- ・ ビット 0 : PA0 モードビット (PA0MD)  
PA0 端子の機能を選びます。

## 18. ピンファンクションコントローラ (PFC)

---

ビット 0	説 明
PA0MD	
0	汎用入出力 (PA0) (初期値)
1	予約

## 18.3.2 ポート A・IO レジスタ (PAIOR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 IOR	PA12 IOR	PA11 IOR	PA10 IOR	PA9 IOR	PA8 IOR	PA7 IOR	PA6 IOR	PA5 IOR	PA4 IOR	—	PA2 IOR	PA1 IOR	PA0 IOR
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

ポート A・IO レジスタ (PAIOR) は、16 ビットの読み出し / 書き込み可能なレジスタで、ポート A にある 14 本のマルチプレックス端子の入出力方向を選びます。PA13IOR ~ PA4IOR、PA2IOR ~ PA0IOR ビットが、それぞれポート A の各端子に対応しています。PAIOR は、ポート A の端子機能が汎用入出力 (PA13 ~ 4、PA2 ~ 0) の場合に有効で、それ以外の場合では無効です。ポート A の端子機能が PA13 ~ PA0 の場合、PAIOR のビットを 1 にすると、対応する端子は出力となり、0 にすると入力となります。

PAIOR は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

## 18.3.3 ポート B コントロールレジスタ (PBCR、PBCR2)

## (1) ポート B コントロールレジスタ (PBCR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15 MD1	PB15 MD0	PB14 MD1	PB14 MD0	PB13 MD1	PB13 MD0	PB12 MD1	PB12 MD0	PB11 MD1	PB11 MD0	PB10 MD1	PB10 MD0	PB9 MD1	PB9 MD0	PB8 MD1	PB8 MD0
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W											

ポート B コントロールレジスタ (PBCR、PBCR2) は、それぞれ 16 ビットの読み出し / 書き込み可能なレジスタで、ポート B にある 16 本のマルチプレックス端子の機能を選びます。PBCR は、ポート B の上位 8 ビットの端子の機能を、PBCR2 はポート B の下位 8 ビットの機能を選びます。

PBCR、PBCR2 は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

- ・ ビット 15、14 : PB15 モードビット 1、0 (PB15MD1、PB15MD0)  
PB15 / SCK1 端子の機能を選びます。

ビット 15	ビット 14	説 明
PB15MD1	PB15MD0	
0	0	汎用入出力 (PB15) (初期値)
	1	予約
1	0	SCIF1 シリアルクロック入出力 (SCK1)
	1	予約

## 18. ピンファンクションコントローラ (PFC)

- ビット 13、12 : PB14 モードビット 1、0 (PB14MD1、PB14MD0)  
PB14 / RXD1 端子の機能を選びます。

ビット 13 PB14MD1	ビット 12 PB14MD0	説 明
0	0	汎用入出力 (PB14) (初期値)
	1	予約
1	0	SCIF1 シリアルデータ入力 (RXD1)
	1	予約

- ビット 11、10 : PB13 モードビット 1、0 (PB13MD1、PB13MD0)  
PB13 / TXD1 端子の機能を選びます。

ビット 11 PB13MD1	ビット 10 PB13MD0	説 明
0	0	汎用入出力 (PB13) (初期値)
	1	予約
1	0	SCIF1 シリアルデータ出力 (TXD1)
	1	予約

- ビット 9、8 : PB12 モードビット 1、0 (PB12MD1、PB12MD0)  
PB12 / SRCK2 /  $\overline{\text{RTS}}$  / STATS1 端子の機能を選びます。

ビット 9 PB12MD1	ビット 8 PB12MD0	説 明
0	0	汎用入出力 (PB12) (初期値)
	1	SIO2 シリアル受信クロック入力 (SRCK2)
1	0	SCIF1 送信リクエスト ( $\overline{\text{RTS}}$ )
	1	BSC ステータス 1 出力 (STATS1)

- ビット 7、6 : PB11 モードビット 1、0 (PB11MD1、PB11MD0)  
PB11 / SRS2 /  $\overline{\text{CTS}}$  / STATS0 端子の機能を選びます。

ビット 7 PB11MD1	ビット 6 PB11MD0	説 明
0	0	汎用入出力 (PB11) (初期値)
	1	SIO2 シリアル受信同期入力 (SRS2)
1	0	SCIF1 送信許可 ( $\overline{\text{CTS}}$ )
	1	BSC ステータス 0 出力 (STATS0)

- ビット 5、4 : PB10 モードビット 1、0 (PB10MD1、PB10MD0)  
PB10 / SRXD2 / TIOCA1 端子の機能を選びます。

ビット5	ビット4	説 明
PB10MD1	PB10MD0	
0	0	汎用入出力 (PB10) (初期値)
	1	SIO2 シリアル受信データ入力 (SRXD2)
1	0	TPU1 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCA1)
	1	予約

- ・ ビット3、2 : PB9 モードビット 1、0 (PB9MD1、PB9MD0)  
PB9 / STCK2 / TIOCB1、TCLKC 端子の機能を選びます。

ビット3	ビット2	説 明
PB9MD1	PB9MD0	
0	0	汎用入出力 (PB9) (初期値)
	1	SIO2 シリアル送信クロック入力 (STCK2)
1	0	TPU1 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCB1) *
	1	予約

【注】 \* TPU で位相計数モード設定または、TCR の TPSC2 ~ 0 の設定により、タイムクロック入力 C (TCLKC) になります。

- ・ ビット1、0 : PB8 モードビット 1、0 (PB8MD1、PB8MD0)  
PB8 / STS2 / TIOCA2 端子の機能を選びます。

ビット1	ビット0	説 明
PB8MD1	PB8MD0	
0	0	汎用入出力 (PB8) (初期値)
	1	SIO2 シリアル送信同期入出力 (STS2)
1	0	TPU2 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCA2)
	1	予約

## (2) ポート B コントロールレジスタ 2 (PBCR2)

ビット :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB7 MD1	PB7 MD0	PB6 MD1	PB6 MD0	PB5 MD1	PB5 MD0	PB4 MD1	PB4 MD0	PB3 MD1	PB3 MD0	PB2 MD1	PB2 MD0	PB1 MD1	PB1 MD0	PB0 MD1	PB0 MD0
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W															

- ・ ビット15、14 : PB7 モードビット 1、0 (PB7MD1、PB7MD0)  
PB7 / STXD2 / TIOCB2、TCLKD 端子の機能を選びます。

## 18. ピンファンクションコントローラ (PFC)

ビット 15	ビット 14	説 明
PB7MD1	PB7MD0	
0	0	汎用入出力 (PB7) (初期値)
	1	SIO2 シリアル送信データ出力 (STXD2)
1	0	TPU2 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCB2)*
	1	予約

【注】 \*TPU で位相計数モード設定または、TCR の TPSC2 ~ 0 の設定により、タイマクロック入力 D (TCLKD) になります。

- ・ ビット 13、12 : PB6 モードビット 1、0 (PB6MD1、PB6MD0)  
PB6 / SRCK1 / SCK2 端子の機能を選びます。

ビット 13	ビット 12	説 明
PB6MD1	PB6MD0	
0	0	汎用入出力 (PB6) (初期値)
	1	SIO1 シリアル受信クロック入力 (SRCK1)
1	0	SCIF2 シリアルクロック入出力 (SCK2)
	1	予約

- ・ ビット 11、10 : PB5 モードビット 1、0 (PB5MD1、PB5MD0)  
PB5 / SRS1 / RXD2 端子の機能を選びます。

ビット 11	ビット 10	説 明
PB5MD1	PB5MD0	
0	0	汎用入出力 (PB5) (初期値)
	1	SIO1 シリアル受信同期入力 (SRS1)
1	0	SCIF2 シリアルデータ入力 (RXD2)
	1	予約

- ・ ビット 9、8 : PB4 モードビット 1、0 (PB4MD1、PB4MD0)  
PB4 / SRXD1 / TXD2 端子の機能を選びます。

ビット 9	ビット 8	説 明
PB4MD1	PB4MD0	
0	0	汎用入出力 (PB4) (初期値)
	1	SIO1 シリアル受信データ入力 (SRXD1)
1	0	SCIF2 シリアルデータ出力 (TXD2)
	1	予約

- ・ ビット 7、6 : PB3 モードビット 1、0 (PB3MD1、PB3MD0)  
PB3 / STCK1 / TIOCA0 端子の機能を選びます。

ビット7	ビット6	説 明
PB3MD1	PB3MD0	
0	0	汎用入出力 (PB3) (初期値)
	1	SIO1 シリアル送信クロック入力 (STCK1)
1	0	TPU0 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCA0)
	1	予約

- ・ ビット5、4 : PB2 モードビット1、0 (PB2MD1、PB2MD0)  
PB2 / STS1 / TIOCB0 端子の機能を選びます。

ビット5	ビット4	説 明
PB2MD1	PB2MD0	
0	0	汎用入出力 (PB2) (初期値)
	1	SIO1 シリアル送信同期入出力 (STS1)
1	0	TPU0 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCB0)
	1	予約

- ・ ビット3、2 : PB1 モードビット1、0 (PB1MD1、PB1MD0)  
PB1 / STXD1 / TIOCC0、TCLKA 端子の機能を選びます。

ビット3	ビット2	説 明
PB1MD1	PB1MD0	
0	0	汎用入出力 (PB1) (初期値)
	1	SIO1 シリアル送信データ出力 (STXD1)
1	0	TPU0 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCC0) *
	1	予約

【注】 \* TPU での位相計数モード設定または、TCR の TPSC2 ~ 0 の設定により、タイマクロック入力 A (TCLKA) になります。

- ・ ビット1、0 : PB0 モードビット1、0 (PB0MD1、PB0MD0)  
PB0 / TIOCD0、TCLKB / WOL 端子の機能を選びます。

ビット1	ビット0	説 明
PB0MD1	PB0MD0	
0	0	汎用入出力 (PB0) (初期値)
	1	予約
1	0	TPU0 インプットキャプチャ入力 / アウトプットコンペア出力 (TIOCD0) *
	1	EtherC ウェイクオンラン出力 (WOL)

【注】 \* TPU での位相計数モード設定または、TCR の TPSC2 ~ 0 の設定により、タイマクロック入力 B (TCLKB) になります。

## 18.3.4 ポート B・IO レジスタ (PBIOR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15 IOR	PB14 IOR	PB13 IOR	PB12 IOR	PB11 IOR	PB10 IOR	PB9 IOR	PB8 IOR	PB7 IOR	PB6 IOR	PB5 IOR	PB4 IOR	PB3 IOR	PB2 IOR	PB1 IOR	PB0 IOR
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート B・IO レジスタ (PBIOR) は、16 ビットの読み出し / 書き込み可能なレジスタで、ポート B にある 16 本のマルチプレックス端子の入出力方向を選びます。PB15IOR ~ PB0IOR ビットが、それぞれポート B の各端子に対応しています。PBIOR は、ポート B の端子機能が汎用入出力 (PB15 ~ PB0) の場合に有効で、それ以外の場合では無効です。ポート B の端子機能が PB15 ~ PB0 の場合、PBIOR のビットを 1 にすると、対応する端子は出力となり、0 にすると入力となります。

PBIOR は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

---

## 19. I/O ポート

---

### 19.1 概要

ポートは、A、Bの2本から構成されています。ポートAは14ビット、ポートBは16ビットの入出力ポートです。それぞれのポートの端子は、汎用入出力とその他の機能とを兼ねているマルチプレクス端子です。(マルチプレクス端子の機能の選択は、ピンファンクションコントローラ(PFC)で行います。)ポートA、Bは、端子のデータを格納するためのデータレジスタをそれぞれ1本ずつ持っています。

## 19.2 ポート A

ポート A は、図 19.1 に示すような、14 本の端子を持つ入出力ポートです。14 本の端子のうち、CKPO 端子にはポートのデータレジスタは存在しなく、内部のクロックとマルチプレクスしています。

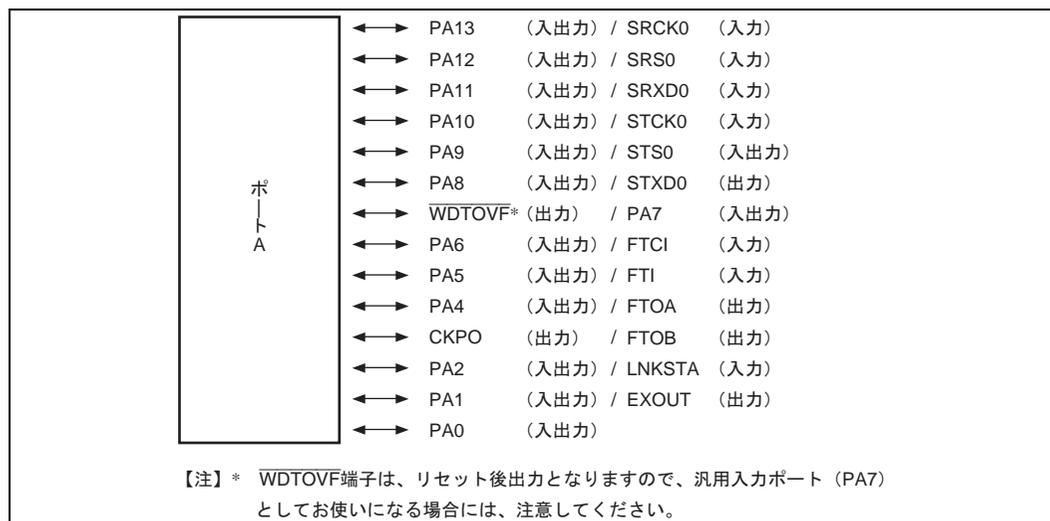


図 19.1 ポート A

### 19.2.1 レジスタ構成

ポート A のレジスタを表 19.1 に示します。

表 19.1 レジスタ構成

名称	略称	R/W	初期値	アドレス	アクセスサイズ
ポート A データレジスタ	PADR	R/W	H'0000	H'FFFFFFC84	8,16

### 19.2.2 ポート A データレジスタ (PADR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 DR	PA12 DR	PA11 DR	PA10 DR	PA9 DR	PA8 DR	PA7 DR	PA6 DR	PA5 DR	PA4 DR	—	PA2 DR	PA1 DR	PA0 DR
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

ポート A データレジスタ (PADR) は、16 ビットの読み出し/書き込み可能なレジスタで、ポート A のデータを格納します。ビット 15、14、3 は予約ビットとなっています。読み出すと常に 0 が読み出されます。書き込む値も 0 にしてください。PA13DR ~ PA0DR ビットが、それぞれ PA13 ~ PA0 端子に対応しています。端子機能が汎用出力の場合には、PADR に値を書き込むと端子からその値が出力され、PADR を読み出すと端子の状態に関係なくレジスタの値が読み出されます。端子機能が汎用入力の場合には、PADR を読み出すとレジスタの値では

なく端子の状態が直接読み出されます。また、PADR に値を書き込むと、PADR にその値を書き込みますが、端子の状態には影響しません。表 19.2 にポート A データレジスタの読み出し / 書き込み時の動作を示します。

PADR は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

表 19.2 ポート A データレジスタ (PADR) の読み出し / 書き込み時の動作

PAIOR	端子機能	読み出し	書き込み
0	汎用入力	端子の状態	PADR に書き込めるが、端子に影響なし
	汎用入力以外	端子の状態	PADR に書き込めるが、端子に影響なし
1	汎用出力	PADR の値	書き込み値が、端子から出力される
	汎用出力以外	PADR の値	PADR に書き込めるが、端子に影響なし

## 19.3 ポート B

ポート B は、図 19.2 に示すような、16 本の端子を持つ入出力ポートです。



図 19.2 ポート B

### 19.3.1 レジスタ構成

ポート B のレジスタを表 19.3 に示します。

表 19.3 レジスタ構成

名称	略称	R/W	初期値	アドレス	アクセスサイズ
ポート B データレジスタ	PBDR	R/W	H'0000	H'FFFFFFC8C	8,16

### 19.3.2 ポート B データレジスタ (PBDR)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15	PB14	PB13	PB12	PB11	PB10	PB9	PB8	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR	DR
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート B データレジスタ (PBDR) は、16 ビットの読み出し / 書き込み可能なレジスタで、ポート B のデータを格納します。PB15DR ~ PB0DR ビットが、それぞれ PB15 ~ PB0 端子に対応しています。端子機能が汎用出力の場合には、PBDR に値を書き込むと端子からその値が出力され、PBDR を読み出すと端子の状態に関係なくレジスタの値が読み出されます。端子機能が汎用入力の場合には、PBDR を読み出すとレジスタの値ではなく端子の状態が直接読み出されます。また、PBDR に値を書き込むと、PBDR にその値を書き込みますが、端子の状態には影響しません。表 19.4 にポート B データレジスタの読み出し / 書き込み時の動作を示します。

PBDR は、パワーオンリセットで H'0000 に初期化されます。しかし、マニュアルリセットおよびスタンバイモード、スリープモード時には初期化されません。

表 19.4 ポート B データレジスタ (PBDR) の読み出し / 書き込み時の動作

PBIOR	端子機能	読み出し	書き込み
0	汎用入力	端子の状態	PBDR に書き込めるが、端子に影響なし
	汎用入力以外	端子の状態	PBDR に書き込めるが、端子に影響なし
1	汎用出力	PBDR の値	書き込み値が、端子から出力される
	汎用出力以外	PBDR の値	PBDR に書き込めるが、端子に影響なし



---

## 20. 低消費電力状態

---

### 20.1 概要

低消費電力状態では、内蔵周辺モジュールや DSP ユニットのうち、不要なモジュールの動作を選択的に停止させ、消費電力を低減させることができるモジュールスタンバイ機能、CPU の機能を停止させるスリープモード、全機能が停止するスタンバイモードがあります。

#### 20.1.1 低消費電力状態の種類

低消費電力モードには、次のようなモードと機能があります。

- (1) スリープモード
- (2) スタンバイモード
- (3) モジュールスタンバイ機能  
(UBC、DMAC、DSP、FRT、SCIF1 ~ 2、TPU、SIO0 ~ 2)

プログラム実行状態から各モードへ遷移する条件、各モードでの CPU や周辺モジュールの状態、各モードの解除方法を表 20.1 に示します。

## 20. 低消費電力状態

表 20.1 低消費電力状態

モード	遷移条件	状態						解除方法
		内蔵発振回路、EtherC、E-DMAC	CPU、キャッシュ	DSP	BSC	UBC、DMAC、FRT、SCIF1 ~ 2、TPU、SIO0 ~ 2	端子	
スリープ	SBYCR1のSBYビットが0の状態 でSLEEP命令を実行	動作	停止	停止	動作	動作	動作	(1) 割り込み (2)DMA アドレスエラー (3) パワーオンリセット (4) マニュアルリセット
スタンバイ	SBYCR1のSBYビットが1の状態 でSLEEP命令を実行	停止	停止	停止	停止かつレジスタの値を保持	UBC：停止かつレジスタの値を保持 UBC以外：停止	保持またはハイインピーダンス	(1)NMI 割り込み (2) パワーオンリセット (3) マニュアルリセット
モジュールスタンバイ機能	各モジュールに該当するMSTPビットを1とする	動作	動作	MSTPが1のときクロックの供給を停止	動作	各モジュールに該当するMSTPビットが1のときクロックの供給を停止	FRT、SCIF1 ~ 2の端子は初期化、その他の端子は動作	(1)MSTP ビットを0 (2) パワーオンリセット (3) マニュアルリセット

### 20.1.2 レジスタ構成

レジスタ構成を表 20.2 に示します。

表 20.2 レジスタ構成

名称	略称	R/W	初期値	アドレス	アクセスサイズ
スタンバイコントロールレジスタ 1	SBYCR1	R/W	H'00	H'FFFFFFE91	8
スタンバイコントロールレジスタ 2	SBYCR2	R/W	H'00	H'FFFFFFE93	8

## 20.2 レジスタの説明

### 20.2.1 スタンバイコントロールレジスタ 1 (SBYCR1)

ビット	7	6	5	4	3	2	1	0
	SBY	HIZ	MSTP5 (UBC)	MSTP4 (DMAC)	MSTP3 (DSP)	—	MSTP1 (FRT)	—
初期値	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R

スタンバイコントロールレジスタ 1 (SBYCR1) は、読み出し / 書き込み可能な 8 ビットのレジスタで、低消費電力モードの状態を設定します。SBYCR1 はリセットで H'00 に初期化されます。

- ・ ビット 7 : スタンバイ (SBY)  
スタンバイモードへの遷移を指定します。  
スタンバイモードへ遷移するときは、必ず WDT を停止 (WDT の WTCSR の TME ビットが '0') させてから、SBY ビットをセットしてください。

ビット 7	説明
SBY	
0	SLEEP 命令の実行でスリープモードへ遷移 (初期値)
1	SLEEP 命令の実行でスタンバイモードへ遷移

- ・ ビット 6 : ポートハイインピーダンス (HIZ)  
スタンバイモード時に、出力端子の状態をハイインピーダンスにするか、出力保持状態とするかを選択します。  
HIZ = 0 (初期状態) で、特定の出力端子は出力を保持します。HIZ = 1 のときは、ハイインピーダンス状態になります。どの端子を制御するかは、「B.1 リセット、低消費電力状態、バス権解放状態での端子状態」を参照してください。

ビット 6	説明
HIZ	
0	スタンバイモード時に、端子状態を前値保持する (初期値)
1	スタンバイモード時に、端子状態をハイインピーダンスする

- ・ ビット 5 : モジュールストップ 5 (MSTP5)  
内蔵周辺モジュールのうち、UBC へのクロック供給の停止を指定します。  
MSTP5 ビットに 1 をセットすると UBC へのクロック供給を停止します。UBC は、クロック供給が停止されても、停止前のレジスタの状態を保持します。ただし、UBC の機能を使用中に本ビットを設定しないでください。

## 20. 低消費電力状態

ビット5	説 明
MSTP5	
0	UBC は動作 (初期値)
1	UBC へのクロック供給を停止

### ・ ビット4：モジュールストップ4 (MSTP4)

内蔵周辺モジュールのうち、DMAC へのクロック供給の停止を指定します。

MSTP4 ビットに 1 をセットすると DMAC へのクロック供給を停止します。DMAC は、クロック供給が停止されても、停止前のレジスタの状態を保持します。したがって、再度 MSTP4 を 0 にクリアして、DMAC の動作を開始すると、停止前の状態から動作を開始します。なお、DMAC の動作中 (転送中) に本ビットをセットしないでください。必ず DMAC 停止中に本ビットをセットしてください。

ビット4	説 明
MSTP4	
0	DMAC は動作 (初期値)
1	DMAC へのクロック供給を停止

### ・ ビット3：モジュールストップ3 (MSTP3)

DSP ユニットへのクロック供給の停止を指定します。

MSTP3 ビットに 1 をセットすると DSP ユニットへのクロック供給を停止します。DSP ユニットは、クロック供給が停止されても、停止前の演算結果を保持します。ただし、本ビットのセットは、DSP ユニット停止時に行ってください。DSP ユニットの停止した場合は、DSP レジスタ、および MACH、MACL をオペランドとするすべての命令は使用できません。

ビット3	説 明
MSTP3	
0	DSP は動作 (初期値)
1	DSP へのクロック供給を停止

### ・ ビット2：予約ビット

読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。

### ・ ビット1：モジュールストップ1 (MSTP1)

内蔵周辺モジュールのうち、FRT へのクロック供給の停止を指定します。

MSTP1 ビットに 1 をセットすると FRT へのクロック供給を停止します。クロック供給が停止されると FRT の各レジスタは初期化されます。ただし、INTC 内にある FRT 用割り込みベクタレジスタは、前の値を保持しています。したがって、再度 MSTP1 を 0 にクリアして、FRT の動作を開始すると、初期状態から動作を開始します。

ビット1	説明
MSTP1	
0	FRT は動作 (初期値)
1	FRT へのクロック供給を停止

- ビット0：予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

## 20.2.2 スタンバイコントロールレジスタ2 (SBYCR2)

ビット	7	6	5	4	3	2	1	0
	—	—	MSTP11 (TPU)	MSTP10 (SIO2)	MSTP9 (SIO1)	MSTP8 (SIO0)	MSTP7 (SCIF2)	MSTP6 (SCIF1)
初期値	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

スタンバイコントロールレジスタ2 (SBYCR2) は、読み出し / 書き込み可能な8ビットのレジスタで、低消費電力モードの状態を設定します。SBYCR2 はリセットで H'00 に初期化されます。

- ビット7、6：予約ビット

読み出すと常に0が読み出されます。書き込む値も常に0にしてください。

- ビット5：モジュールストップ11 (MSTP11)

内蔵周辺モジュールのうち、TPU へのクロック供給の停止を指定します。

MSTP11 ビットに1をセットすると TPU へのクロック供給を停止します。クロック供給が停止されても、停止前の状態を保持します。また、INTC 内にある TPU 用割り込みベクタレジスタは、前の値を保持しています。したがって、再度 MSTP11 を0にクリアして、TPU へのクロック供給を再開させると、再び動作を開始します。

ビット5	説明
MSTP11	
0	TPU は動作 (初期値)
1	TPU へのクロック供給を停止

- ビット4：モジュールストップ10 (MSTP10)

内蔵周辺モジュールのうち、SIO チャンネル2 へのクロック供給の停止を指定します。

MSTP10 ビットに1をセットすると SIO チャンネル2 へのクロック供給を停止します。SIO チャンネル2 は、クロック供給が停止されても、停止前の状態を保持します。また、INTC 内にある SIO チャンネル2 用割り込みベクタレジスタは、前の値を保持しています。したがって、MSTP10 を0にクリアして、SIO チャンネル2 へのクロック供給を再開させると、再び動作を開始します。

## 20. 低消費電力状態

ビット4	説 明
MSTP10	
0	SIO チャンネル 2 は動作 (初期値)
1	SIO チャンネル 2 へのクロック供給を停止

### ・ ビット 3 : モジュールストップ 9 (MSTP9)

内蔵周辺モジュールのうち、SIO チャンネル 1 へのクロック供給の停止を指定します。

MSTP9 ビットに 1 をセットすると SIO チャンネル 1 へのクロック供給を停止します。SIO チャンネル 1 は、クロック供給が停止されても、停止前の状態を保持します。また、INTC 内にある SIO チャンネル 1 用割り込みベクタレジスタは、前の値を保持しています。したがって、MSTP9 を 0 にクリアして、SIO チャンネル 1 へのクロック供給を再開させると、再び動作を開始します。

ビット3	説 明
MSTP9	
0	SIO チャンネル 1 は動作 (初期値)
1	SIO チャンネル 1 へのクロック供給を停止

### ・ ビット 2 : モジュールストップ 8 (MSTP8)

内蔵周辺モジュールのうち、SIO チャンネル 0 へのクロック供給の停止を指定します。

MSTP8 ビットに 1 をセットすると SIO チャンネル 0 へのクロック供給を停止します。SIO チャンネル 0 は、クロック供給が停止されても、停止前の状態を保持します。また、INTC 内にある SIO チャンネル 0 用割り込みベクタレジスタは、前の値を保持しています。したがって、MSTP8 を 0 にクリアして、SIO チャンネル 0 へのクロック供給を再開させると、再び動作を開始します。

ビット2	説 明
MSTP8	
0	SIO チャンネル 0 は動作 (初期値)
1	SIO チャンネル 0 へのクロック供給を停止

### ・ ビット 1 : モジュールストップ 7 (MSTP7)

内蔵周辺モジュールのうち、SCIF2 へのクロック供給の停止を指定します。

MSTP7 ビットに 1 をセットすると SCIF2 へのクロック供給を停止します。クロック供給が停止されると SCIF2 の各レジスタは初期化されます。ただし、INTC 内にある SCIF2 用割り込みベクタレジスタは、前の値を保持しています。したがって、再度 MSTP7 を 0 にクリアして、SCIF2 の動作を開始すると、初期状態から動作を開始します。

ビット1	説 明
MSTP7	
0	SCIF2 は動作 (初期値)
1	SCIF2 へのクロック供給を停止

・ ビット0：モジュールストップ6 (MSTP6)

内蔵周辺モジュールのうち、SCIF1 へのクロック供給の停止を指定します。

MSTP6 ビットに 1 をセットすると SCIF1 へのクロック供給を停止します。クロック供給が停止されると SCIF1 の各レジスタは初期化されます。ただし、INTC 内にある SCIF1 用割り込みベクタレジスタは、前の値を保持しています。したがって、MSTP6 を 0 にクリアして、SCIF1 の動作を開始すると、初期状態から動作を開始します。

ビット0 MSTP6	説明
0	SCIF1 は動作 (初期値)
1	SCIF1 へのクロック供給を停止

## 20.3 スリープモード

### 20.3.1 スリープモードへの遷移

SBYCR1 の SBY ビットが 0 の状態で、SLEEP 命令を実行すると、プログラム実行状態からスリープモード状態に遷移します。CPU は SLEEP 命令実行後に停止しますが、CPU のレジスタ内容は保持されます。内蔵周辺モジュールは、動作を続けます。

### 20.3.2 スリープモードの解除

スリープモードは、割り込み、DMA アドレスエラー、パワーオンリセット、マニュアルリセットにより解除されます。

(1) 割り込みによる解除

割り込みが発生すると、スリープモードが解除され、割り込み例外処理が実行されます。ただし、発生した割り込みの優先レベルが CPU のステータスレジスタ (SR) に設定されている割り込みマスクレベル以下の場合、および内蔵周辺モジュールによる割り込みがモジュール側で禁止されている場合には、割り込み要求は受け付けられず、スリープモードは解除されません。

(2) DMA アドレスエラーによる解除

DMA アドレスエラーが発生すると、スリープモードが解除され、DMA アドレスエラー例外処理を実行します。

(3) パワーオンリセットによる解除

パワーオンリセットにより、スリープモードは解除されます。

(4) マニュアルリセットによる解除

マニュアルリセットにより、スリープモードは解除されます。

## 20.4 スタンバイモード

### 20.4.1 スタンバイモードへの遷移

SBYCR1 の SBY ビットを 1 にセットした後、SLEEP 命令を実行すると、プログラム実行状態からスタンバイ状態に遷移します。このとき、SLEEP 命令実行と SLEEP 命令実行後 5 サイクルは NMI 割り込みを受け付けません。スタンバイモードでは、CPU だけでなくすべてのモジュールへのクロック供給が停止します。

CPU レジスタ内容は保持されますが、内蔵周辺モジュールに関しては、初期化されるものがあります。

表 20.3 スタンバイモード時のレジスタ状態

モジュール	初期化されるレジスタ等	内容が保持されるレジスタ等	内容が不定のレジスタ等
割り込みコントローラ (INTC)		全レジスタ	
ユーザブ레이크コントローラ (UBC)		全レジスタ	
バスステートコントローラ (BSC)		全レジスタ	
汎用ダイレクトメモリアクセスコントローラ (DMAC)	<ul style="list-style-type: none"> <li>• DMA チャンネルコントロールレジスタ 0、1</li> <li>• DMA オペレーションレジスタ</li> </ul>	<ul style="list-style-type: none"> <li>• DMA ソースアドレスレジスタ 0、1</li> <li>• DMA デスティネーションアドレスレジスタ 0、1</li> <li>• DMA トランスファカウンタレジスタ 0、1</li> <li>• DMA 要求 / 応答選択制御レジスタ 0、1</li> <li>• ベクタ番号設定レジスタ DMA0、1</li> </ul>	
ウォッチドッグタイマ (WDT)	<ul style="list-style-type: none"> <li>• タイマコントロール / ステータスレジスタのビット 7 ~ 5</li> <li>• リセットコントロール / ステータスレジスタ</li> </ul>	<ul style="list-style-type: none"> <li>• タイマコントロール / ステータスレジスタのビット 2 ~ 0</li> <li>• タイマカウンタ</li> </ul>	
16 ビットフリーランニングタイマ (FRT)	全レジスタ		
FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF1 ~ 2)	全レジスタ		
シリアル I/O (SIO0 ~ 2)		全レジスタ	
ハイパフォーマンスユーザデバッグインタフェース (H-UDI)		全レジスタ	
16 ビットタイマパルスユニット (TPU)		全レジスタ	
ピンファンクションコントローラ (PFC)		全レジスタ	
イーサネットコントローラ用ダイレクトメモリアクセスコントローラ (E-DMAC)	全レジスタ		
イーサネットコントローラ (EtherC)	全レジスタ		
その他		<ul style="list-style-type: none"> <li>• スタンバイコントロールレジスタ 1、2</li> <li>• 周波数変更レジスタ</li> </ul>	

#### 20.4.2 スタンバイモードの解除

スタンバイモードは、NMI 割り込み、パワーオンリセット、マニュアルリセットにより解除されます。

##### (1) NMI 割り込みによる解除

NMI 信号の立ち上がり、また立ち下がりエッジが検出されると、WDT のタイマコントロール/ステータスレジスタに設定されている時間が経過後、LSI 全体にクロックが供給され、スタンバイモードが解除され、NMI 割り込み例外処理が実行されます。WDT の設定は、発振安定時間以上になるようにしてください。

なお、NMI 信号の立ち下がりエッジでスタンバイモードを解除する場合、スタンバイに入るとき（クロック停止時）の NMI 端子はハイレベルに、かつスタンバイ復帰時（発振安定後のクロック起動時）の NMI 端子はローレベルになるようにしてください。NMI 端子のローレベルは、CKIO 端子からクロック信号出力開始後最低 3 サイクルまではレベルを保持するようにしてください。また、NMI 信号の立ち上がりエッジでスタンバイモードを解除する場合、スタンバイに入るとき（クロック停止時）の NMI 端子はローレベルに、かつスタンバイ復帰時（発振安定後のクロック起動時）の NMI 端子はハイレベルになるようにしてください。NMI 端子のハイレベルは、CKIO 端子からクロック信号出力開始後最低 3 サイクルまではレベルを保持するようにしてください。

##### (2) パワーオンリセットによる解除

パワーオンリセットにより、スタンバイモードは解除されます。

##### (3) マニュアルリセットによる解除

マニュアルリセットにより、スタンバイモードは解除されます。

### 20.4.3 NMI 割り込みによるスタンバイモード解除方法

NMI 信号の立ち下がりによってスタンバイモードに遷移し、NMI 信号の立ち上がりによって解除を行う例を説明します。この例のタイミングを図 20.1 に示します。

割り込みコントロールレジスタ (ICR) の NMI エッジセレクトビット (NMIE) を 0 (立ち下がりエッジ検出) にした状態で NMI 端子をハイレベルからローレベルに変化させると、NMI 割り込みが受け付けられます。NMI 例外サービスルーチンで NMIE ビットを 1 (立ち上がりエッジ検出) にセットし、スタンバイコントロールレジスタ 1 (SBYCR1) のスタンバイビット (SBY) を 1 にセットして SLEEP 命令を実行すると、スタンバイモードに遷移します。その後、NMI 端子をローレベルからハイレベルに変化させると、スタンバイモードが解除されます。NMI 端子のハイレベルは、CKIO 端子からクロック信号出力開始後最低 3 サイクルまではレベルを保持するようにしてください。

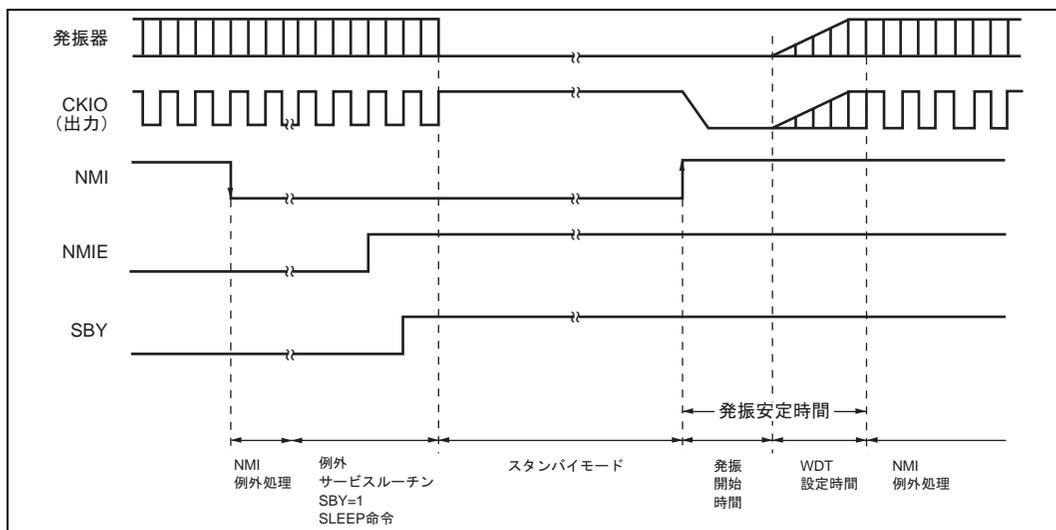


図 20.1 NMI 割り込みによるスタンバイモード解除方法

### 20.4.4 クロックポーズ機能

CKIO 端子からクロックを入力する場合、そのクロックの周波数を変更したり、クロックそのものを停止させたりできます。本 LSI はこのため  $\overline{\text{CKPREQ}}/\text{CKM}$  端子を持っています。ただし、ウォッチドックタイマ (WDT) の動作中 (WDT のタイマコントロール/ステータスレジスタ (WTCSR) のタイムアウトビット (TME) が 1 のとき) には、クロックポーズは受け付けません。また、クロックポーズリクエスト機能を使用する場合には、必ずスタンバイコントロールレジスタ 1 (SBYCR1) のスタンバイビット (SBY) を 1 にセットしてからリクエスト信号を入力してください。

クロックポーズ機能は次のように使用します。

- (1) WDT の WTCSR の TME ビットを 0 に、SBYCR1 の SBY ビットを 1 に設定してください。
- (2)  $\overline{\text{CKPREQ}}/\text{CKM}$  端子にローレベルを印加します。

## 20. 低消費電力状態

- (3) LSI 内部がスタンバイ状態に遷移すると、 $\overline{\text{CKPACK}}$  端子からローレベルが出力されます。
- (4)  $\overline{\text{CKPACK}}$  端子がローレベルになったことを確認した後、クロックの停止、または周波数の変更を行ってください。
- (5) クロックポーズ状態（スタンバイ状態）を解除するには、 $\overline{\text{CKPREQ/CKM}}$  端子にハイレベルを印加してください。（LSI 内部では、 $\overline{\text{CKPREQ/CKM}}$  端子の立ち上がりエッジを検出することにより、スタンバイ状態を解除します）
- (6) PLL 回路 1 が動作状態の場合、LSI 内部では WDT がカウントアップを始めます。PLL 回路 1 が停止状態の場合、WDT は起動しません。
- (7) 内部クロックが安定すると、 $\overline{\text{CKPACK}}$  端子がハイレベルに変化し、LSI が動作可能状態になったことを外部に知らせます。

なお、クロックポーズによるスタンバイ状態、各内蔵周辺モジュールの状態および端子状態は、通常のスタンバイモードと同じ状態となります。

PLL 回路 1 が動作状態における、クロックポーズ機能のタイミングチャートを図 20.2 に示します。

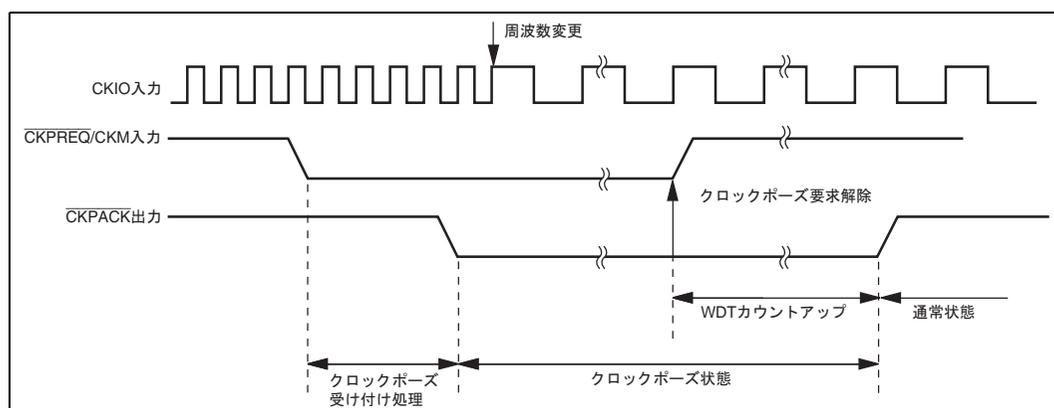


図 20.2 クロックポーズ機能のタイミングチャート（PLL 回路 1 動作時）

PLL 回路 1 が停止状態における、クロックポーズ機能のタイミングチャートを図 20.3 に示します。

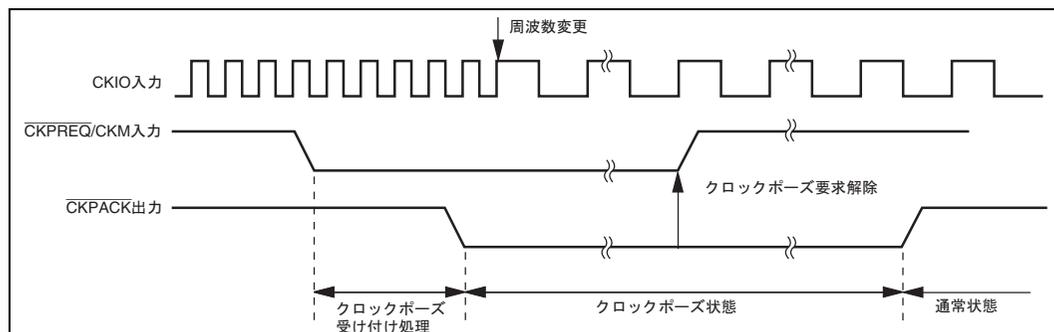


図 20.3 クロックポーズ機能のタイミングチャート (PLL 回路 1 停止時)

クロックポーズ状態は、通常スタンバイ状態と同様、NMI 入力によっても解除することができます。ただし、NMI 入力後 CKIO クロックの 4 サイクル以内にクロックポーズ要求を解除してください。NMI 入力 (立ち上がりエッジ検出時) によるクロックポーズ状態解除のタイミングチャートを図 20.4 に示します。

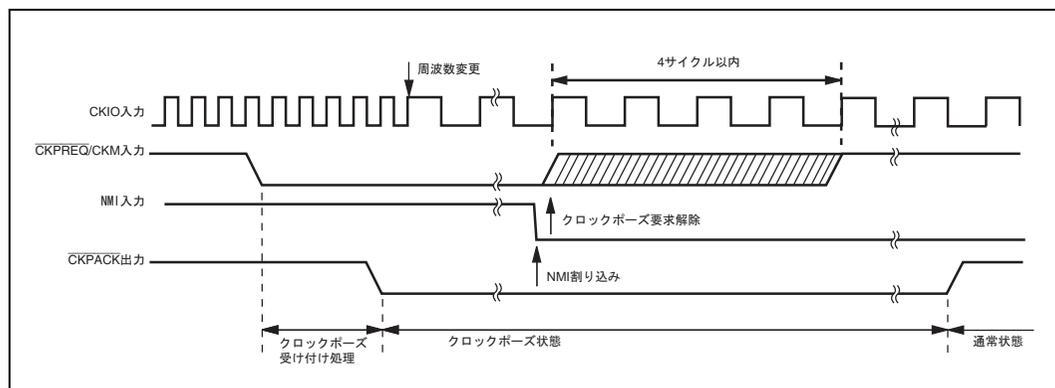


図 20.4 クロックポーズ機能のタイミングチャート (NMI 入力による解除)

#### 20.4.5 スタンバイモードの注意事項

(1) キャッシュを使用しているときに、スタンバイモードにするときは、スタンバイモードに移移する前に、キャッシュをディスエーブルしてください。スタンバイ復帰後、キャッシュを使用する場合、それに先立ってキャッシュの初期化を行ってください。

また、キャッシュを内蔵 RAM として使用しているときも、スタンバイ中は内蔵 RAM の値は保持しません。

(2) スタンバイモードに入る前の 10 クロックサイクル中に内蔵周辺レジスタをライトアクセスする場合は、該当するレジスタをリード後 SLEEP 命令を実行してください。

- (3) クロックモード0、1、2で使用する場合、CKIO 端子はクロック出力端子になります。本クロックモードでスタンバイモードを使用する場合は、以下のことに注意してください。
- スタンバイモードを NMI 割り込みにより解除する場合、NMI 入力後、発振安定時間中に不安定なクロックが CKIO 端子から出力されます。同様にパワーオンリセット、マニュアルリセットによる解除でも出力されます。そのときパワーオンリセット、マニュアルリセットは発振安定時間以上入力してください。
- (4) スタンバイモードに遷移する際は、必ず内蔵 DMAC (E-DMAC、DMAC) の動作を停止させてから行ってください。

## 20.5 モジュールスタンバイ機能

### 20.5.1 モジュールスタンバイ状態への遷移

スタンバイコントロールレジスタ1、2の MSTP11 ~ 3、1ビットに1をセットすることで、それぞれのビットに対応した内蔵周辺モジュールおよび DSP ユニットへのクロック供給を停止させることができます。この機能を使用することで、消費電力を低減させることができます。モジュールスタンバイ中の各モジュールへは、リード/ライトアクセスしないでください。

モジュールスタンバイ機能で、DMAC、SIO0 ~ 2の内蔵周辺モジュールに関わる外部端子は停止前の状態を保持します。また、DMAC、DSP、SIO0 ~ 2のレジスタは停止前の状態を保持します。FRT、SCIF1 ~ 2、TPUに関わる外部端子は、リセット時の状態になります。また、レジスタはすべて初期化されます。

モジュールスタンバイビットに該当するモジュールが動作中には、モジュールスタンバイ状態に遷移させないでください。また、モジュールストップさせるモジュールからの割り込みを禁止させてください。

### 20.5.2 モジュールスタンバイ状態の解除

モジュールスタンバイ状態の解除は、MSTP11 ~ 3、1ビットを0にクリアするか、パワーオンリセット、またはマニュアルリセットにより行います。

---

## 21. 電気的特性

---

### 21.1 絶対最大定格

絶対最大定格を表 21.1 に示します。

表 21.1 絶対最大定格

項目	記号	定格値	単位
電源電圧 (内部)	$V_{CC}$	- 0.3 ~ +4.2	V
電源電圧 (5V I/O)	$PV_{CC}$	- 0.3 ~ +7.0	V
入力電圧 (5V I/O 以外)	$V_{in}$	- 0.3 ~ $V_{CC}+0.3$	V
入力電圧 (5V I/O)	$V_{in}$	- 0.3 ~ $PV_{CC}+0.3$	V
動作温度	$T_{opr}$	- 20 ~ + 75	
保存温度	$T_{stg}$	- 55 ~ +125	

- 【注】
1. 最大定格を超えて LSI を使用した場合、LSI の永久破壊となることがあります。
  2. 電源の投入は、内部用電源 ( $V_{CC}$ ) を投入後または投入と同時に 5V I/O 用電源 ( $PV_{CC}$ ) を投入するシーケンスとしてください。  
電源の切断は、 $PV_{CC}$  を切断後または切断と同時に  $V_{CC}$  を切断するシーケンスとしてください。

## 21. 電気的特性

### 21.2 DC 特性

DC 特性を表 21.2、表 21.3 に示します。

表 21.2 DC 特性

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	typ	max	単位	測定条件	
入力ハイレベル電圧	$\overline{RES}$ , NMI, MD4 ~ MD0, $\overline{TRST}$ , $\overline{CKPREQ}$ /CKM	$V_{CC} \times 0.9$		$V_{CC} + 0.3$	V		
	3.3V/5V 兼用端子	2.6		$PV_{CC} + 0.3$	V		
	EXTAL, CKIO	$V_{CC} \times 0.9$		$V_{CC} + 0.3$	V		
	その他の入力端子	$V_{CC} \times 0.7$		$V_{CC} + 0.3$	V		
入力ローレベル電圧	$\overline{RES}$ , NMI, MD4 ~ MD0, $\overline{TRST}$ , $\overline{CKPREQ}$ /CKM	-0.3		$V_{CC} \times 0.1$	V		
	その他の入力端子	-0.3		0.8	V		
シュミットトリガ入力電圧	PB14/RXD1、	$V_{T-}$		0.8	V	PV <sub>CC</sub> = 5V ± 0.5V の場合	
	PB5/SRS1/RXD2	$V_{T+}$	4.0		V		
		$V_{T+}$	2.6		V	上記以外の場合	
		$V_{T+} - V_{T-}$	0.3		V		
入力リーク電流	全入力端子	$ I_{in} $		1.0	μA	$V_{in} = 0.5 \sim V_{CC}-0.5V$ $V_{in} = 0.5 \sim PV_{CC}-0.5V$	
スリープ状態電流リーク	入出力、出力全端子 (オフ状態)	$ I_{TSI} $		1.0	μA	$V_{in} = 0.5 \sim V_{CC}-0.5V$ $V_{in} = 0.5 \sim PV_{CC}-0.5V$	
出力ハイレベル電圧	3.3V/5V 兼用端子	$V_{OH}$	$PV_{CC} - 0.7$		V	$I_{OH} = -200\mu A$	
	その他の出力端子		$V_{CC} - 0.5$		V	$I_{OH} = -200\mu A$	
			$V_{CC} - 1.0$		V	$I_{OH} = -1mA$	
出力ローレベル電圧	3.3V/5V 兼用端子	$V_{OL}$		0.6	V	$I_{OL} = 1.6mA$	
	その他の出力端子			0.4	V	$I_{OL} = 1.6mA$	
端子容量	CAP1、CAP2	$C_{in}$		40	pF	$T_a = 25$	
	その他の全入力端子			15	pF		
消費電流	通常動作時	$I_{CC}$			350	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 使用時
					300	mA	3.6V、CPU 動作クロック = 62.5MHz、DMAC 未使用時
			スリープ時		250	mA	3.6V、CPU 動作クロック = 62.5MHz、周辺モジュール未使用時
	スタンバイ時			990	μA		

【注】 PLL を使用しないときに、PLL<sub>V<sub>CC</sub></sub>、PLL<sub>V<sub>SS</sub></sub> 端子を解放しないでください。  
 PLL<sub>V<sub>CC</sub></sub> 端子は V<sub>CC</sub> に、PLL<sub>V<sub>SS</sub></sub> 端子は V<sub>SS</sub> にそれぞれ接続してください。

表 21.3 出力許容電流値

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	typ	max	単位
出力ローレベル許容電流 (1 端子当たり)	$I_{OL}$			2.0	mA
出力ローレベル許容電流 (総和)	$I_{OL}$			80	mA
出力ハイレベル許容電流 (1 端子当たり)	$-I_{OH}$			2.0	mA
出力ハイレベル許容電流 (総和)	$(-I_{OH})$			25	mA

【注】 LSI の信頼性を確保するため、出力電流値は表 21.3 の値を超えないようにしてください。

### 21.3 AC 特性

本 LSI の入力は原則としてクロック同期入力です。特にことわりがない限り、各入力信号のセットアップ・ホールド時間は必ず守ってください。

表 21.4 最大動作周波数

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目		記号	min	typ	max	単位	備考
動作周波数	CPU、DSP	f	1		62.5	MHz	$t_{cyc}$
	外部バス (SDRAM 未使用時)		1		31.25		$t_{E_{cyc}}$
	外部バス (SDRAM 使用時)		1		62.5		$t_{E_{cyc}}$
	周辺モジュール		1		31.25		$t_{P_{cyc}}$

## 21.3.1 クロックタイミング

表 21.5 クロックタイミング

条件：  $V_{CC}=PLL_{VCC} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC}$   $V_{CC}$ 、 $V_{SS} =$   
 $PV_{SS} = PLL_{VSS} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
EXTAL クロック入力周波数	$f_{EX}$	1	31.25	MHz	21.1
EXTAL クロック入力サイクル時間	$t_{EXcyc}$	32	1000	ns	
EXTAL クロック入力 Low レベルパルス幅	$t_{EXL}$	$8^{*1}$ 、 $12^{*2}$		ns	
EXTAL クロック入力 High レベルパルス幅	$t_{EXH}$	$8^{*1}$ 、 $12^{*2}$		ns	
EXTAL クロック入力立ち上がり時間	$t_{EXr}$		4	ns	
EXTAL クロック入力立ち下がり時間	$t_{EXf}$		4	ns	
CKIO クロック入力周波数	$f_{CKI}$	1	31.25	MHz	21.2
CKIO クロック入力サイクル時間	$t_{CKIcyc}$	32	1000	ns	
CKIO クロック入力 Low レベルパルス幅	$t_{CKIL}$	$8^{*3}$ 、 $12^{*4}$		ns	
CKIO クロック入力 High レベルパルス幅	$t_{CKIH}$	$8^{*3}$ 、 $12^{*4}$		ns	
CKIO クロック入力立ち上がり時間	$t_{CKIr}$		4	ns	
CKIO クロック入力立ち下がり時間	$t_{CKIf}$		4	ns	
CKIO クロック出力周波数	$f_{OP}$	$1^{*5}$ 、 $8^{*6}$	62.5	MHz	21.3
CKIO クロック出力サイクル時間	$t_{cyc}$	16	$1000^{*5}$ 、 $125^{*6}$	ns	
CKIO クロック出力 Low レベルパルス幅	$t_{CKOL}$	3		ns	
CKIO クロック出力 High レベルパルス幅	$t_{CKOH}$	3		ns	
CKIO クロック立ち上がり時間	$t_{CKOr}$		5	ns	
CKIO クロック立ち下がり時間	$t_{CKOf}$		5	ns	
CKPO クロック出力サイクル時間	$t_{CKPCYC}$	32	1000	ns	21.4
CKPO クロック出力 Low レベルパルス幅	$t_{CKPOL}$	11		ns	
CKPO クロック出力 High レベルパルス幅	$t_{CKPOH}$	11		ns	
CKPO クロック立ち上がり時間	$t_{CKPOr}$		5	ns	
CKPO クロック立ち下がり時間	$t_{CKPOf}$		5	ns	
パワーオン発振安定時間	$t_{OSC1}$	10		ms	
スタンバイ復帰発振安定時間 1	$t_{OSC2}$	10		ms	21.6
スタンバイ復帰発振安定時間 2	$t_{OSC3}$	10		ms	21.7
PLL 同期安定化時間	$t_{PLL}$	1		ms	21.8

- 【注】 \*1 PLL 回路 2 動作時  
 \*2 PLL 回路 2 不使用时  
 \*3 PLL 回路 1 動作時  
 \*4 PLL 回路 1 不使用时  
 \*5 PLL 回路 1 不使用时、PLL 回路 2 不使用时  
 \*6 PLL 回路 1 または PLL 回路 2 動作時



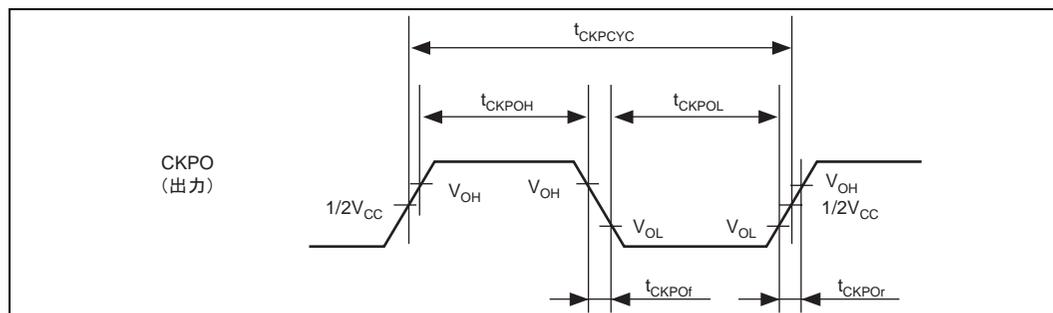


図 21.4 CKPO クロック出力タイミング

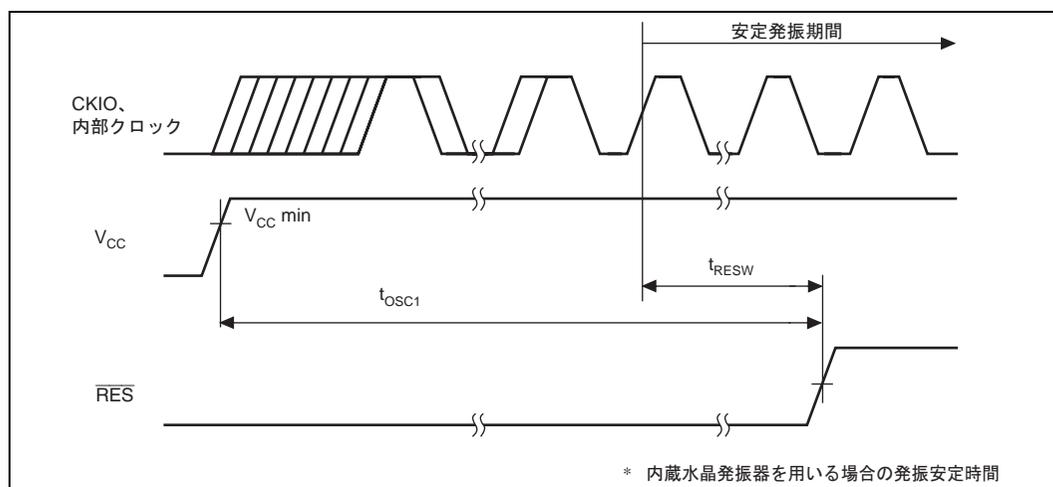


図 21.5 パワーオン時発振安定時間

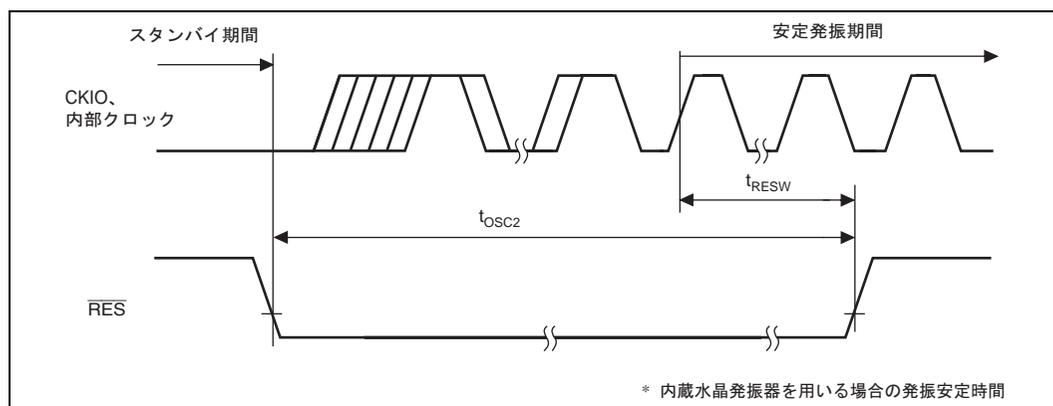


図 21.6 スタンバイ復帰時発振安定時間 (RES による復帰)

21. 電気的特性

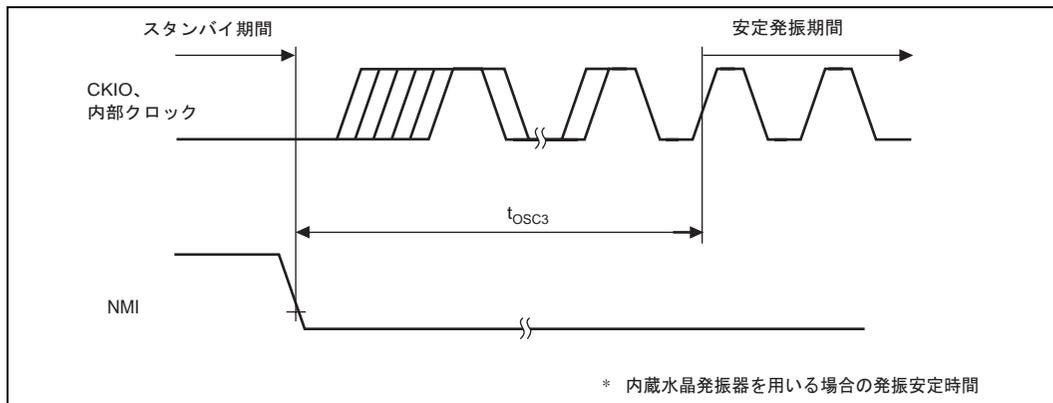


図 21.7 スタンバイ復帰時発振安定時間 (NMI による復帰)

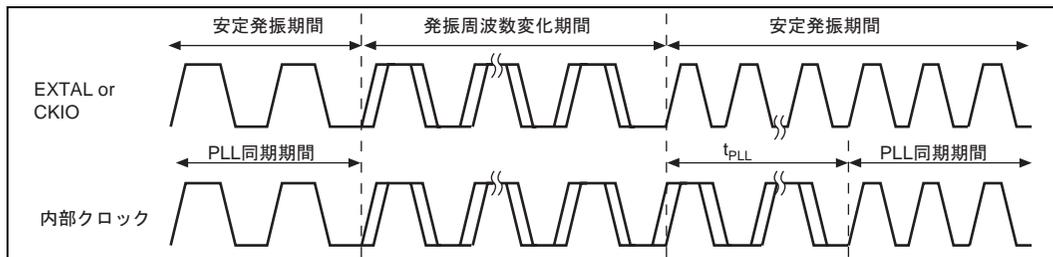


図 21.8 PLL 同期安定化時間

## 21.3.2 制御信号タイミング

表 21.6 制御信号タイミング

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図	
RES 立ち上がり、立ち下がり	$t_{RESr}$		200	ns	21.9	
	$t_{RESf}$					
	RES パルス幅	$t_{RESW}$	20	tPcyc		
	NMI リセットセットアップ時間	$t_{NMIRS}$	$t_{Pcyc}+10$	ns		
	NMI リセットホールド時間	$t_{NMIRH}$	$t_{Pcyc}+10$	ns		
NMI 立ち上がり、立ち下がり	$t_{NMIr}$		200	ns	21.10	
	$t_{NMIf}$					
RES セットアップ時間 *	$t_{RESS}$	$3t_{E_{cyc}}+40$		ns		
NMI セットアップ時間 *	$t_{NMIS}$	40		ns		
IRL3 ~ IRL0 セットアップ時間 *	$t_{IRLS}$	30		ns		
NMI ホールド時間	$t_{NMIH}$	20	r	ns		
IRL3 ~ IRL0 ホールド時間 *	$t_{IRLH}$	20		ns		
BRLS セットアップ時間	$t_{BLSS}$	10		ns	21.11	
	BRLS ホールド時間	$t_{BLSH}$	5			ns
	BGR 遅延時間	$t_{BGRD}$		15		ns
	バスタライステート遅延時間	$t_{BOFF}$	0	35		ns
	バスバッファオン時間	$t_{BON}$	0	35		ns

【注】 \* RES、NMI および IRL3 ~ IRL0 信号は非同期入力ですが、ここに示されたセットアップが守られた場合、クロックの立ち下がりで変化が生じたものとして判定されます。セットアップを守れない場合、次のクロック立ち下がりまで認識が遅れることがあります。

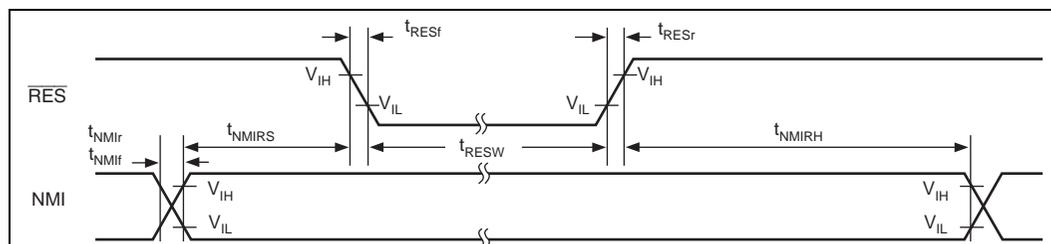


図 21.9 リセット入力タイミング

21. 電気的特性

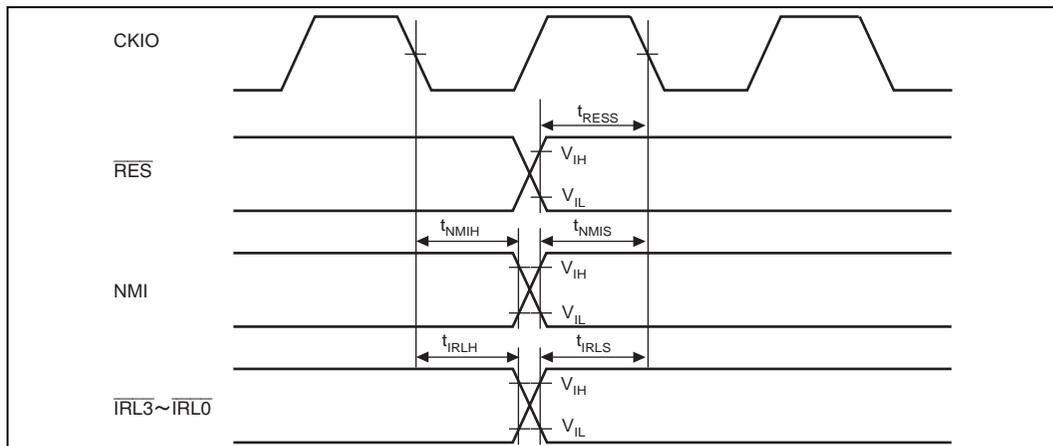


図 21.10 割り込み信号入力タイミング

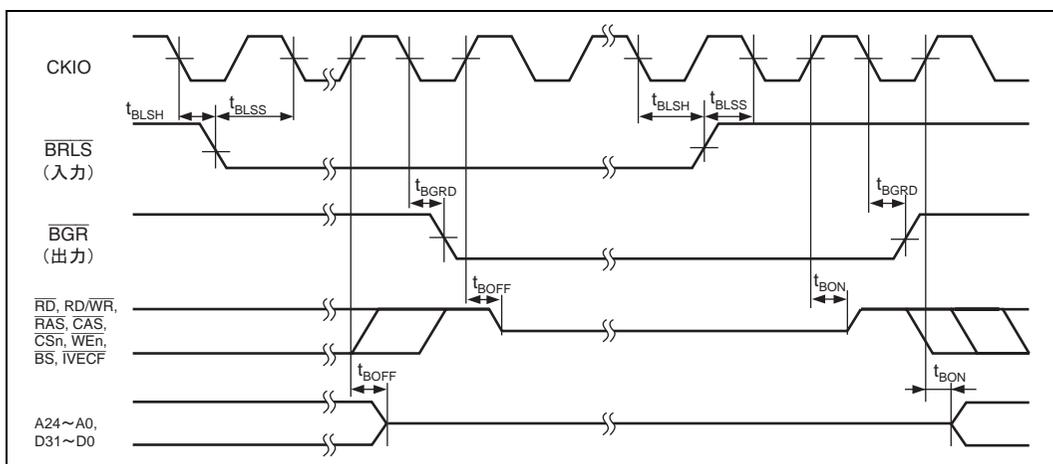


図 21.11 バス権解放タイミング

## 21.3.3 バスタイミング

表 21.7 PLL オンバスタイミング [モード 0、4](1)

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} \quad V_{CC}$ 、 $V_{SS} =$   
 $PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
アドレス遅延時間	$t_{AD}$	1	14	ns	21.12、13、16、17、19、 21、23、25、26、27、28、 29、31、32、33、34、35、 38、39、40、41、43、44、 45
BS 遅延時間	$t_{BSD}$		15	ns	21.12、13、16、17、19、 21、23、25、26、29、31、 32、34、35、40、43、44、 45
CS 遅延時間 1	$t_{CSD1}$	1	14	ns	21.12、13、16、17、19、 21、23、24、25、26、29、 31、32、33、34、35、40、 42、43
CS 遅延時間 2	$t_{CSD2}$		14	ns	21.12、13、34、35、40、43
読み出し / 書き込み遅延時間	$t_{RWD}$	1	14	ns	21.12、13、16、17、19、 21、22、23、25、26、29、 30、31、32、33、34、35、 40、43、44、45
読み出しストロブ遅延時間 1	$t_{RSD1}$		14	ns	21.12、13、16、17、23、 31、34、35、38、40、41、 43、44、45
読み出しデータセットアップ時間 1	$t_{RDS1}$	8		ns	21.12、34、38、43、44、45
読み出しデータセットアップ時間 2 (EDO)	$t_{RDS2}$	8		ns	21.40、41
読み出しデータセットアップ時間 3 (SDRAM)	$t_{RDS3}$	6.5		ns	21.16、17
読み出しデータホールド時間 2	$t_{RDH2}$	0		ns	21.12、43
読み出しデータホールド時間 4 (SDRAM)	$t_{RDH4}$	2		ns	21.16、17
読み出しデータホールド時間 5 (DRAM)	$t_{RDH5}$	0		ns	21.34、38
読み出しデータホールド時間 6 (EDO)	$t_{RDH6}$	3		ns	21.40、41
読み出しデータホールド時間 7 (EDO)	$t_{RDH7}$	1		ns	21.40
読み出しデータホールド時間 8 (割り込みベクタ)	$t_{RDH8}$	2		ns	21.44、45

21. 電気的特性

表 21.7 PLL オンバスタイミング [モード 0、4] (2)

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} \quad V_{CC}$ 、 $V_{SS} =$   
 $PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
書き込みイネーブル遅延時間 1	$t_{WED1}$		14	ns	21.12、13
書き込みデータ遅延時間 1 (E : I = 1:1 以外)	$t_{WDD1}$		22	ns	21.13、23、25、27、35、39
書き込みデータ遅延時間 2 (E : I = 1:1)	$t_{WDD2}$		12	ns	21.26、28
書き込みデータホールド時間 1	$t_{WDH1}$	2		ns	21.13、23、25、26、27、 28、35、39
データバッファオン時間	$t_{DON}$		15	ns	21.13、23、25、26、35
データバッファオフ時間	$t_{DOF}$		15	ns	21.13、23、25、26、35
DACK 遅延時間 1	$t_{DACK1}$		14	ns	21.12、13、16、19、21、 23、25、26、29、34、35、 38、39、40、41、43
DACK 遅延時間 2	$t_{DACK2}$		14	ns	21.12、13、34、35、38、 39、40、41、43
WAIT セットアップ時間	$t_{WTS}$	10		ns	21.14、15、36、37、43、 44、45、46
WAIT ホールド時間	$t_{WTH}$	5		ns	21.14、15、36、37、43、 44、45、46
RAS 遅延時間 1 (SDRAM)	$t_{RASD1}$	1	14	ns	21.16、17、18、19、21、 22、23、24、25、26、29、 30、31、32、33
RAS 遅延時間 2 (DRAM、EDO)	$t_{RASD2}$		14	ns	21.34、35、40、42
RAS 遅延時間 3 (EDO)	$t_{RASD3}$		14	ns	21.40
CAS 遅延時間 1 (SDRAM)	$t_{CASD1}$	1	14	ns	21.16、17、18、19、23、 24、25、26、27、28、29、 31、32、33、43
CAS 遅延時間 2 (DRAM)	$t_{CASD2}$		14	ns	21.34、35、38、39、40、 41、42
DQM 遅延時間	$t_{DQMD}$	1	14	ns	21.16、17、19、20、21、 23、25、26、27、28、29、 30
CKE 遅延時間	$t_{CKED}$	1	14	ns	21.33
OE 遅延時間 1	$t_{OED1}$		14	ns	21.40
OE 遅延時間 2	$t_{OED2}$		14	ns	21.40
IVECF 遅延時間	$t_{IVD}$		15	ns	21.44、45
ロウアドレスセットアップ時間	$t_{ASR}$	0		ns	21.34、35、40
カラムアドレスセットアップ時間	$t_{ASC}$	0		ns	21.34、35、38、39、40
データ入力セットアップ時間	$t_{DS}$	0		ns	21.35、39
読み出し / 書き込みアドレスセット アップ時間	$t_{AS}$	0		ns	21.12、13
REFOUT 遅延時間	$t_{REFOD}$		15	ns	21.47

表 21.7 PLL オンバスタイミング [モード 0、4](3)

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 5\%$ 、 $PV_{CC} = 5.0V \pm 5\%/3.3V \pm 5\%$ 、 $PV_{CC} < V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -5 \sim +70$ 、SDRAM バスサイクル時

項目	記号	min	max	単位	参照図
読み出しデータセットアップ時間 3 (SDRAM)	$t_{RDS3}$	6.5		ns	21.16、17
読み出しデータホールド時間 4 (SDRAM)	$t_{RDH4}$	1.5		ns	21.16、17
書き込みデータ遅延時間 2 (E :I =1:1)	$t_{WDD2}$		9.5	ns	21.26、28
書き込みデータホールド時間 1	$t_{WDH1}$	2		ns	21.26、28
アドレス遅延時間	$t_{AD}$	4	11	ns	21.16、17、19、21、23、 25、26、27、28、29、31、 32、33
CS 遅延時間 1	$t_{CSD1}$	2.5	9.5	ns	21.16、17、19、21、23、 24、25、26、29、31、32、 33
読み出し / 書き込み遅延時間	$t_{RWD}$	2.5	9.5	ns	21.16、17、19、21、22、 23、25、26、29、30、31、 32、33
DQM 遅延時間	$t_{DQMD}$	2.5	9.5	ns	21.16、17、19、20、21、 23、25、26、27、28、29、 30
RAS 遅延時間 1 (SDRAM)	$t_{RASD1}$	2.5	9.5	ns	21.16、17、18、21、22、 23、24、25、26、29、30、 31、32、33
CAS 遅延時間 1 (SDRAM)	$t_{CASD1}$	2.5	9.5	ns	21.16、17、18、19、23、 24、25、26、27、28、29、 31、32、33
CKE 遅延時間	$t_{CKED}$	2.5	9.5	ns	21.33

21. 電気的特性

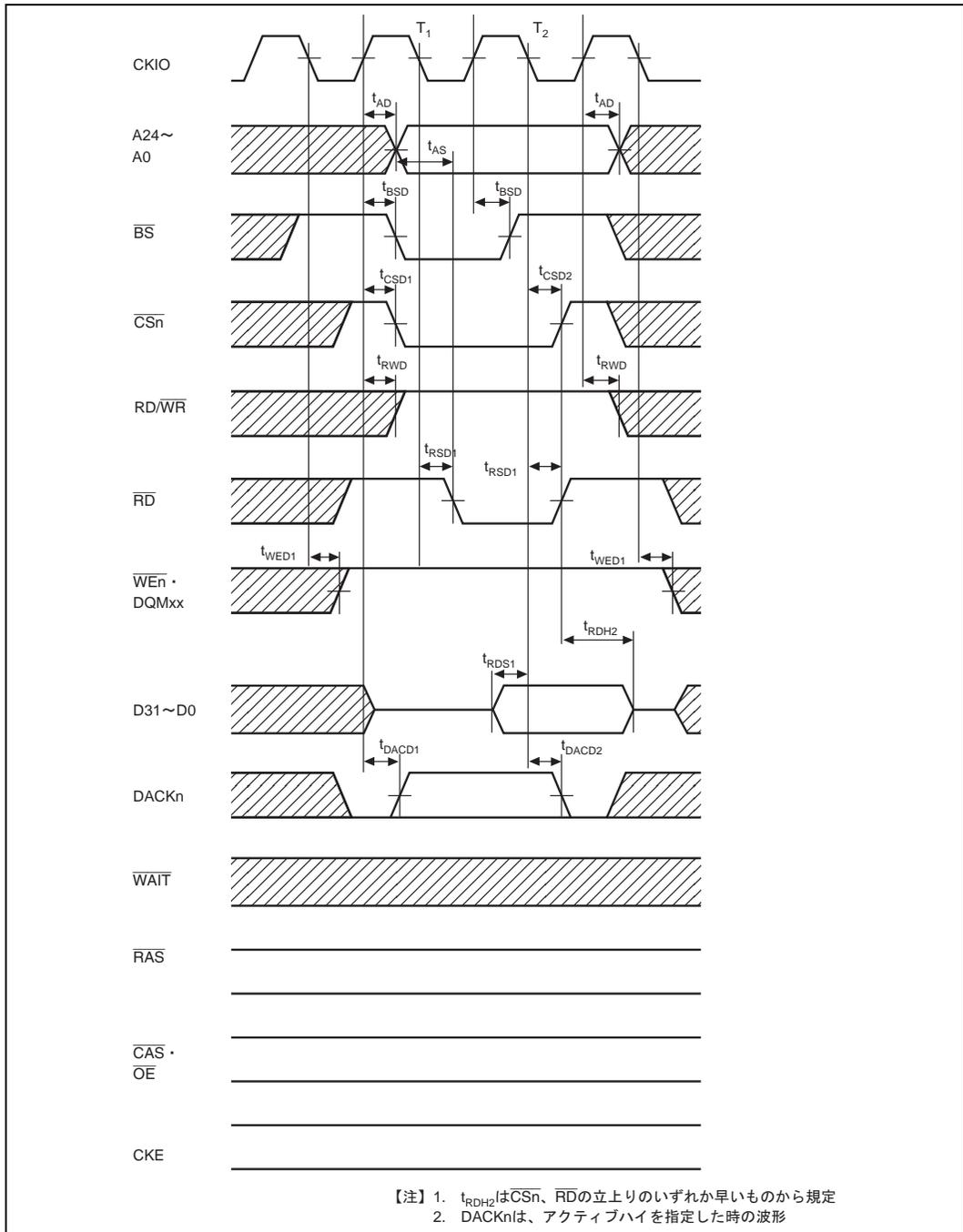


図 21.12 基本読み出しサイクル（ノーウェイト）

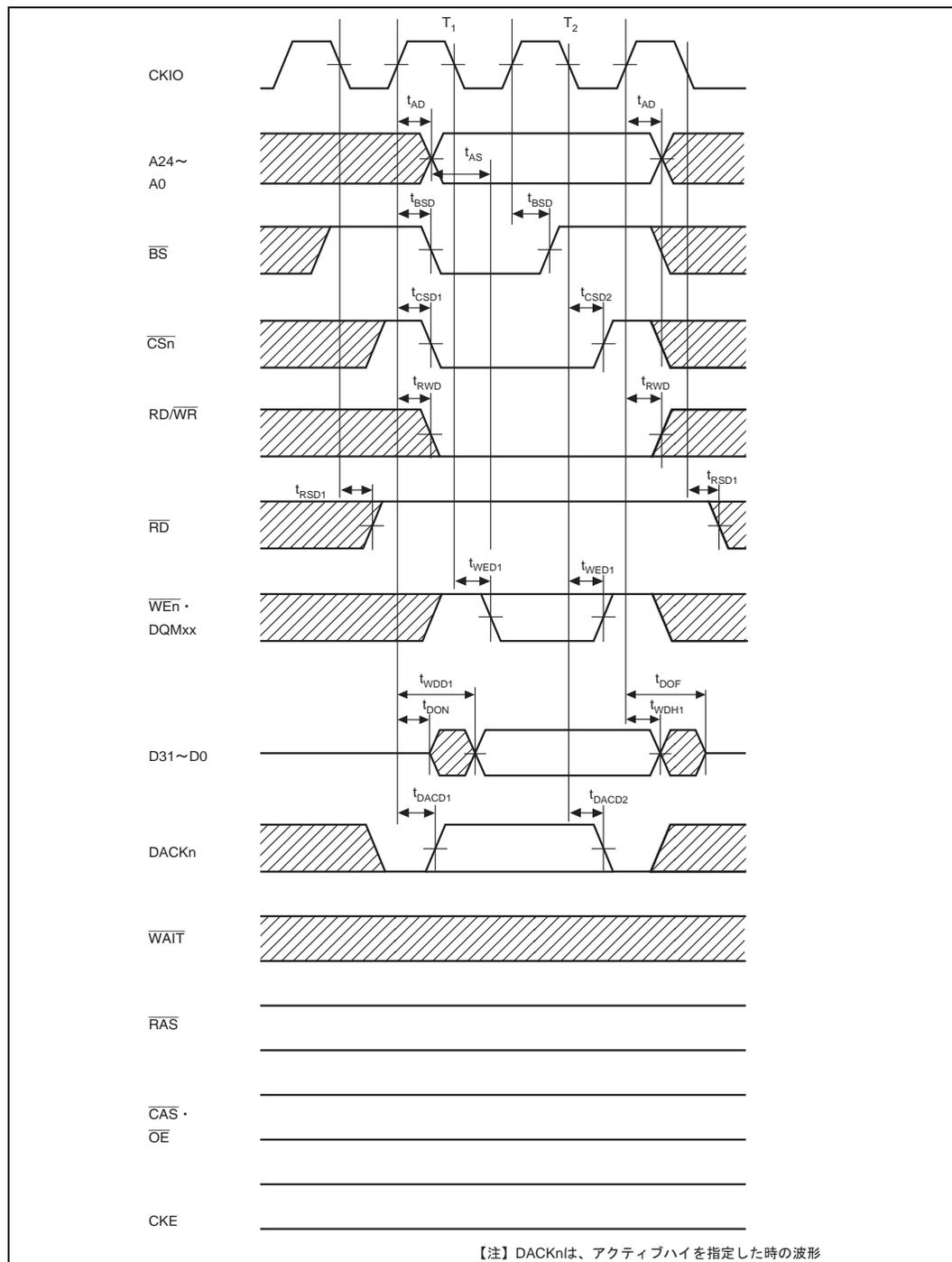


図 21.13 基本書き込みサイクル（ノーウェイト）

21. 電気的特性

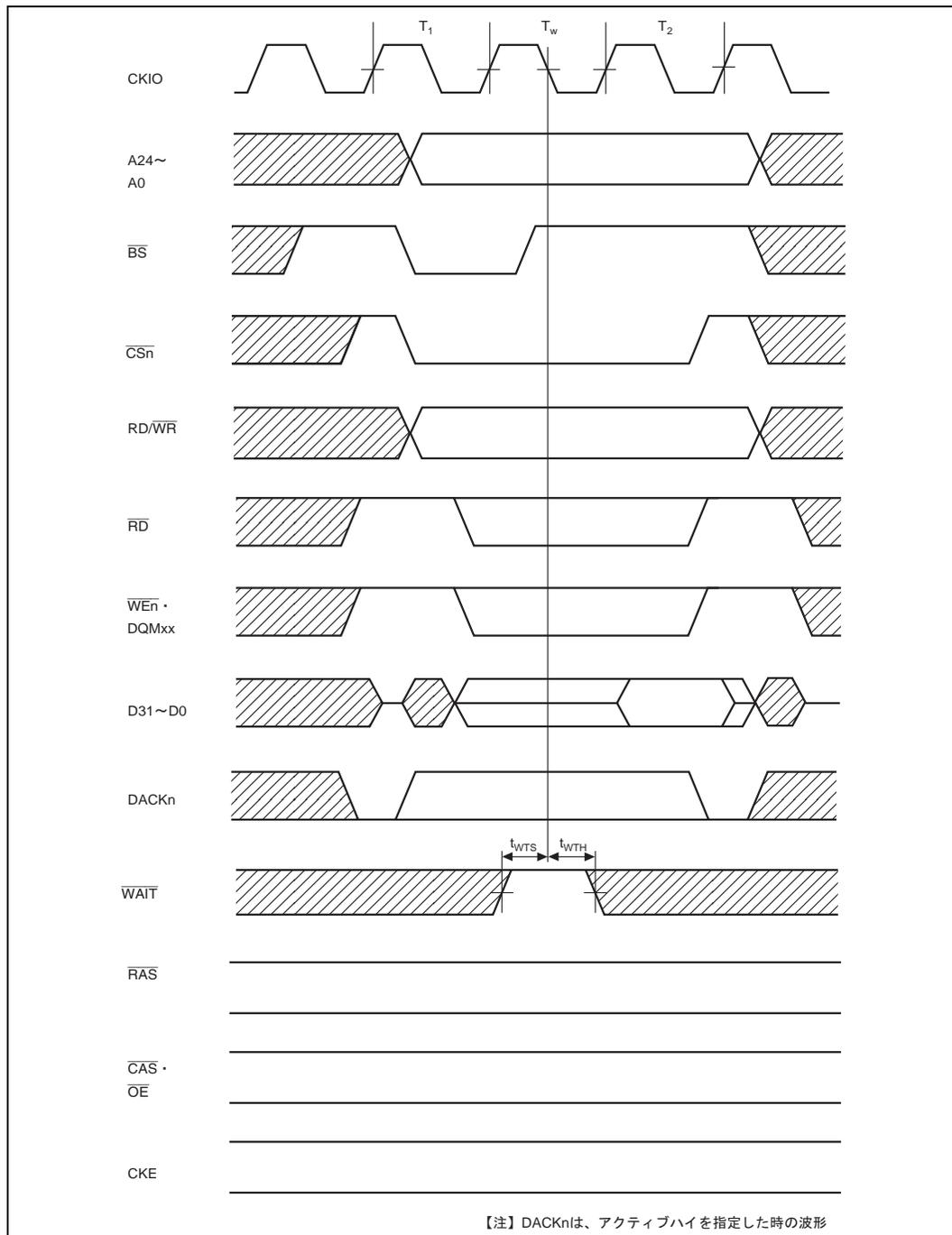


図 21.14 基本バスサイクル (1ウェイトサイクル)

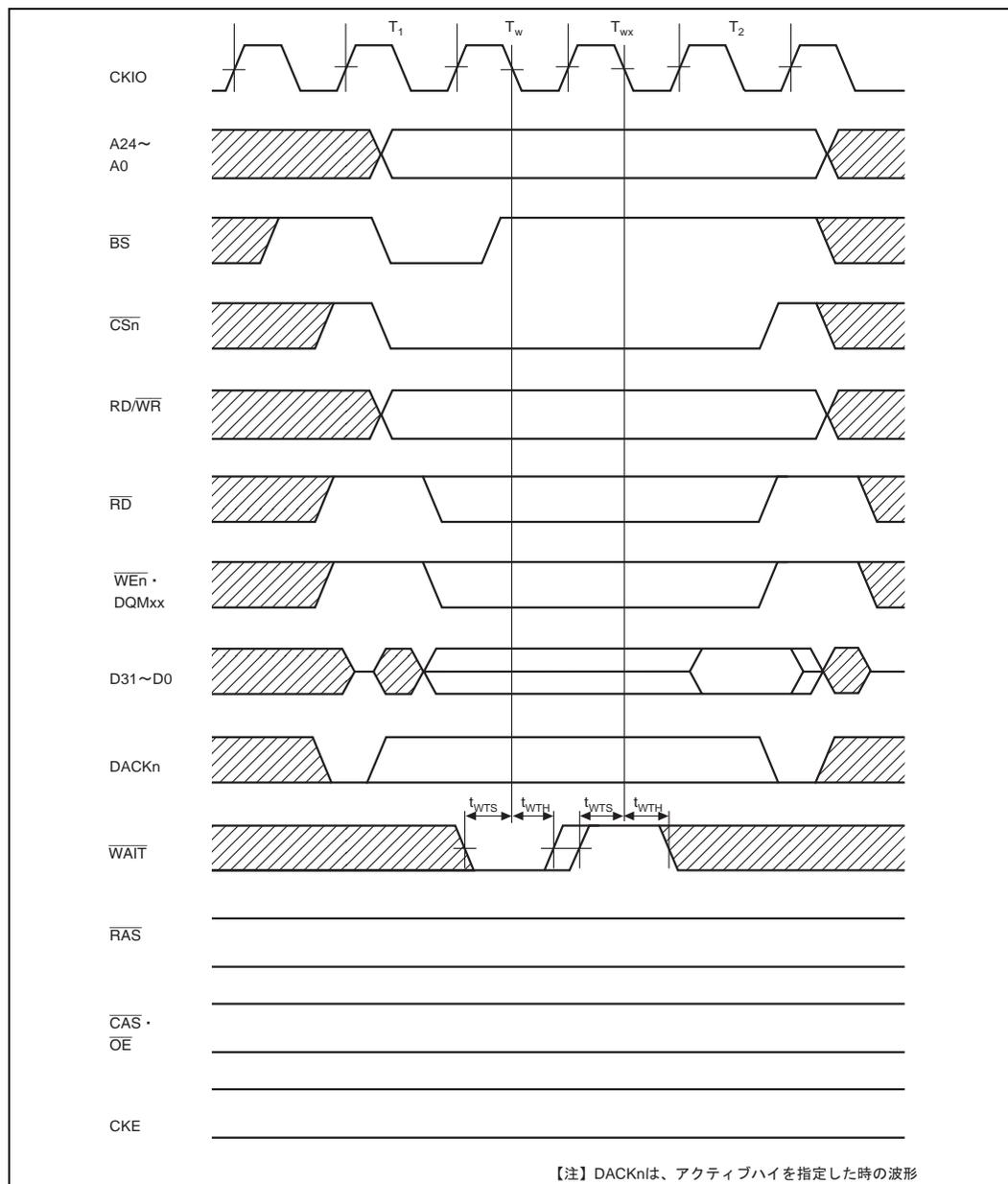


図 21.15 基本バスサイクル (外部ウェイト入力)

21. 電気的特性

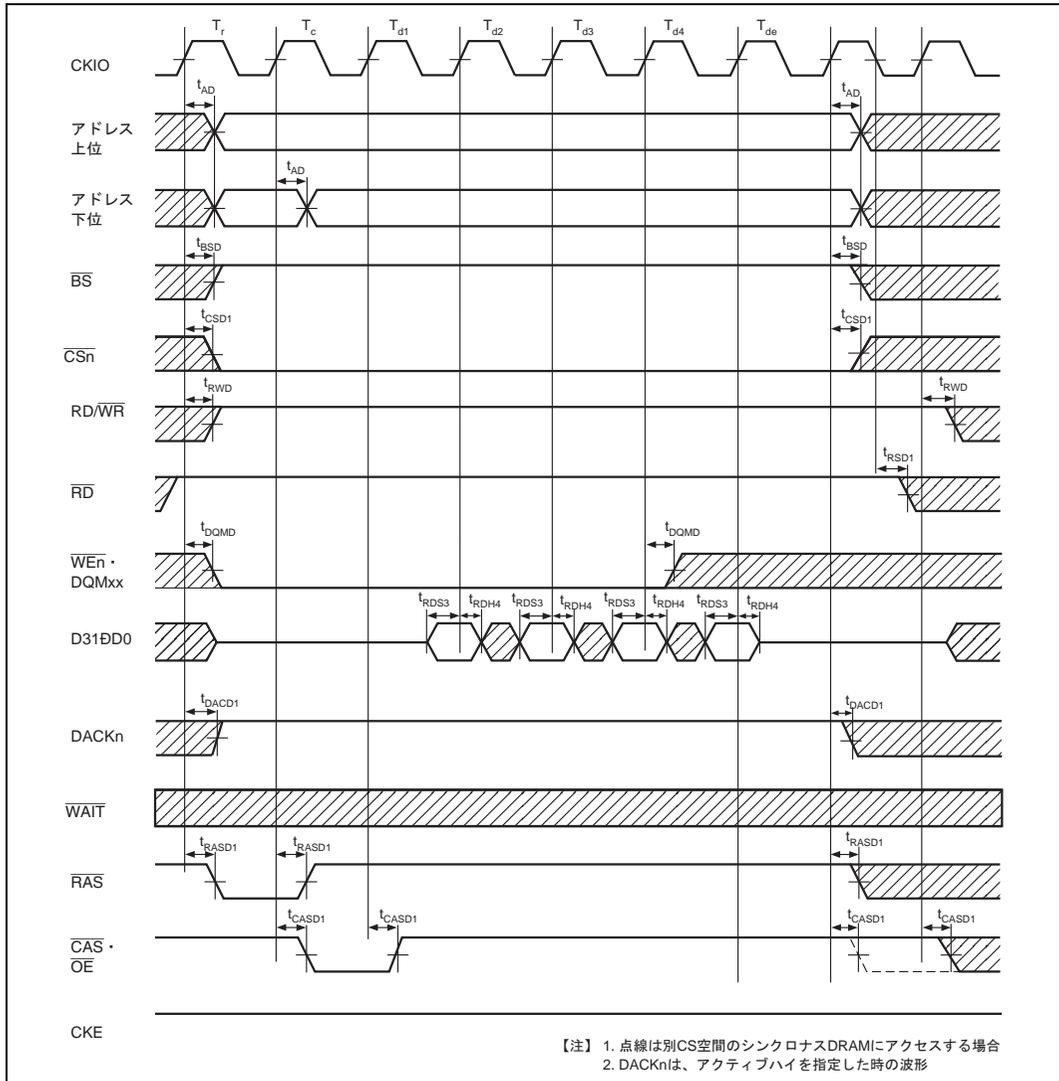


図 21.16 シンクロナス DRAM 読み出しバスサイクル  
(RCD=1 サイクル、CAS レイテンシ=1 サイクル、バースト=4)

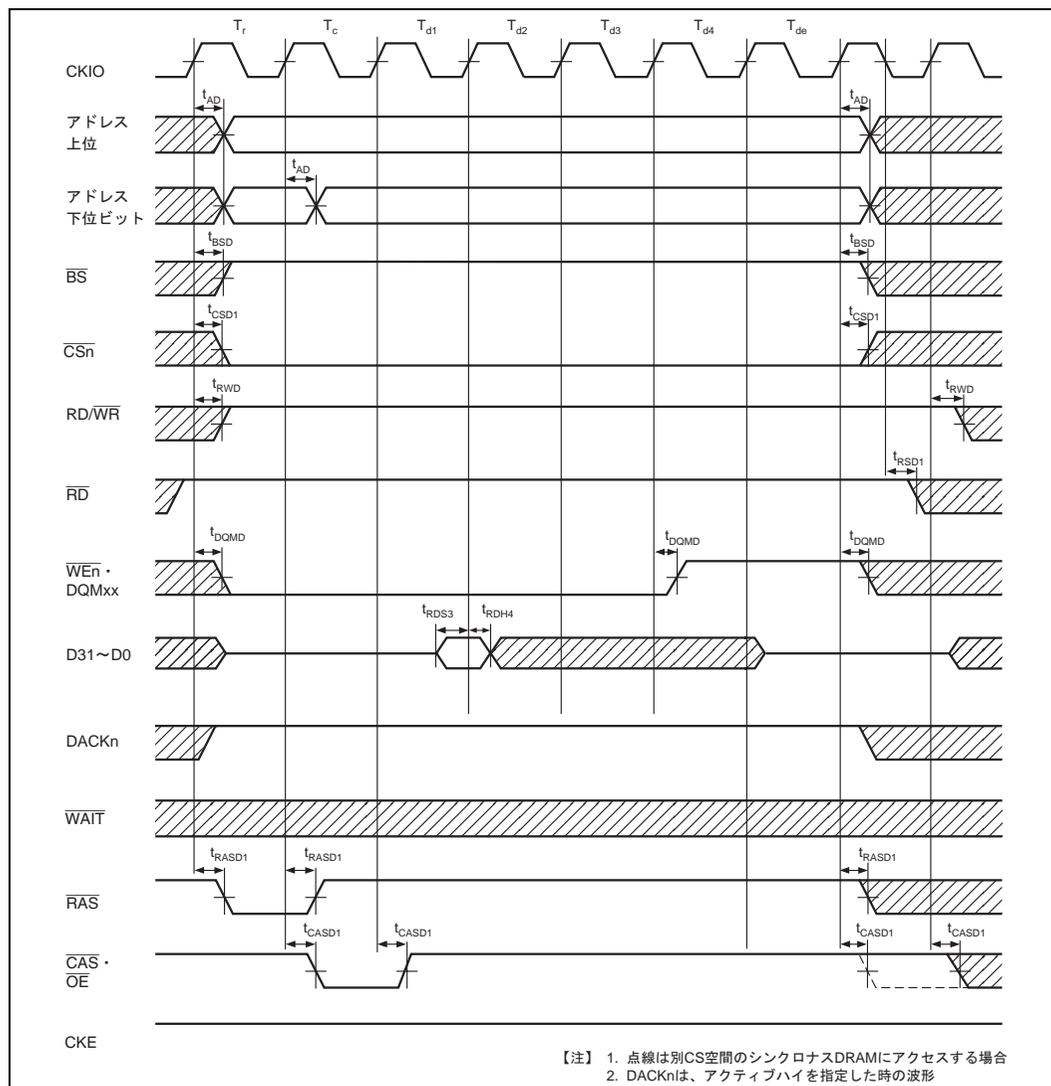


図 21.17 シンクロナス DRAM シングル読み出しバスサイクル  
(RCD= 1 サイクル、CAS レイテンシ =1 サイクル、バースト=4)

21. 電気的特性

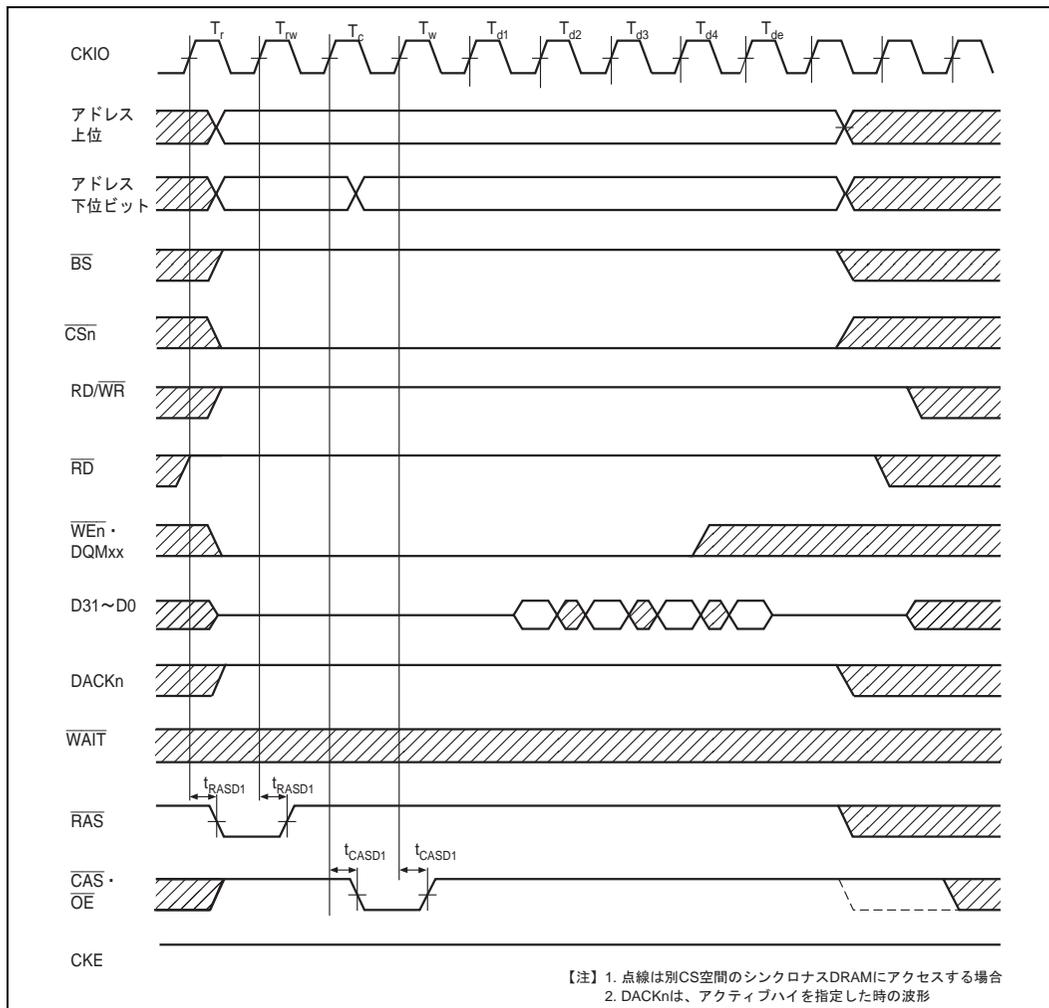


図 21.18 シンクロナス DRAM 読み出しバスサイクル  
(RCD=2 サイクル、CAS レイテンシ=2 サイクル、バースト = 4 )

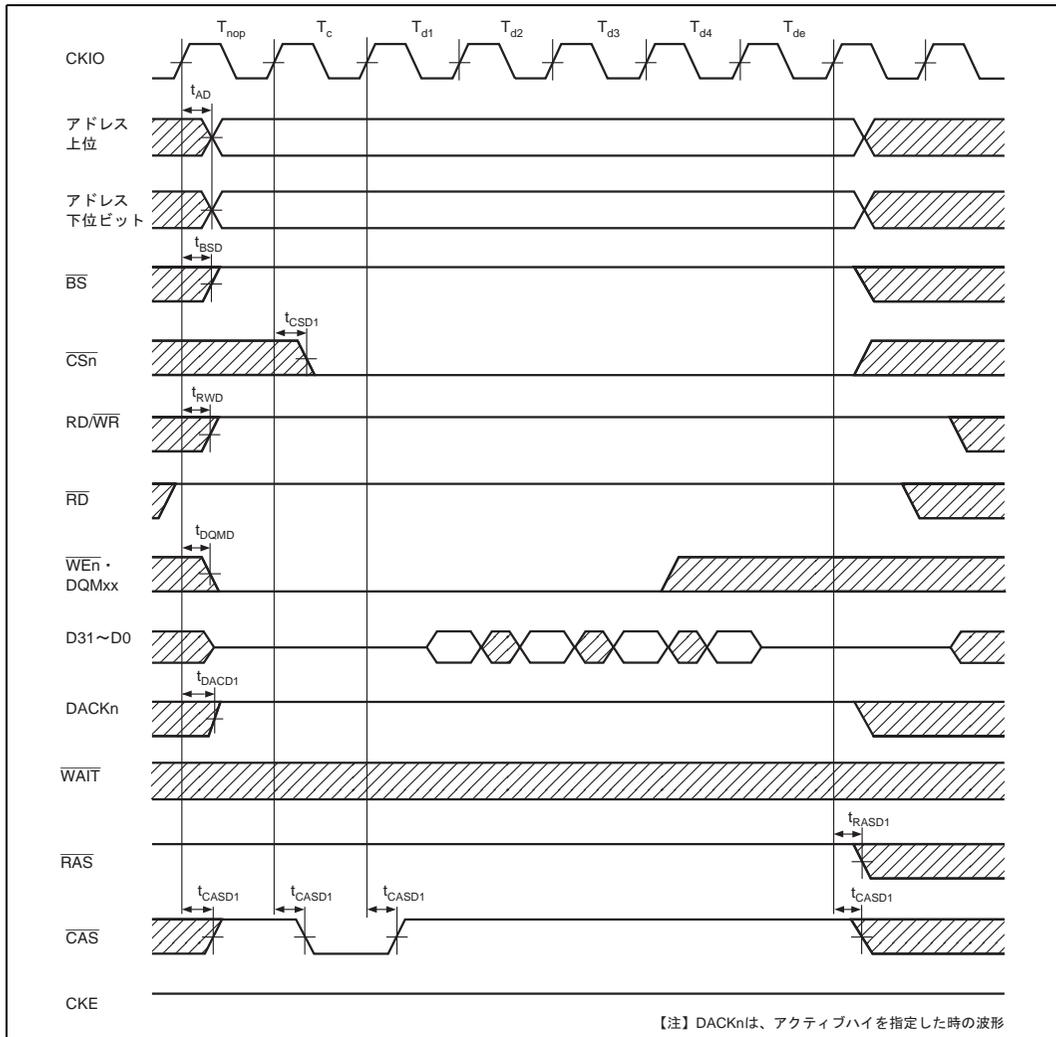


図 21.19 シンクロナス DRAM 読み出しバスサイクル  
(バンクアクティブ、同一ロウアクセス、CAS レイテンシ=1 サイクル)

21. 電気的特性

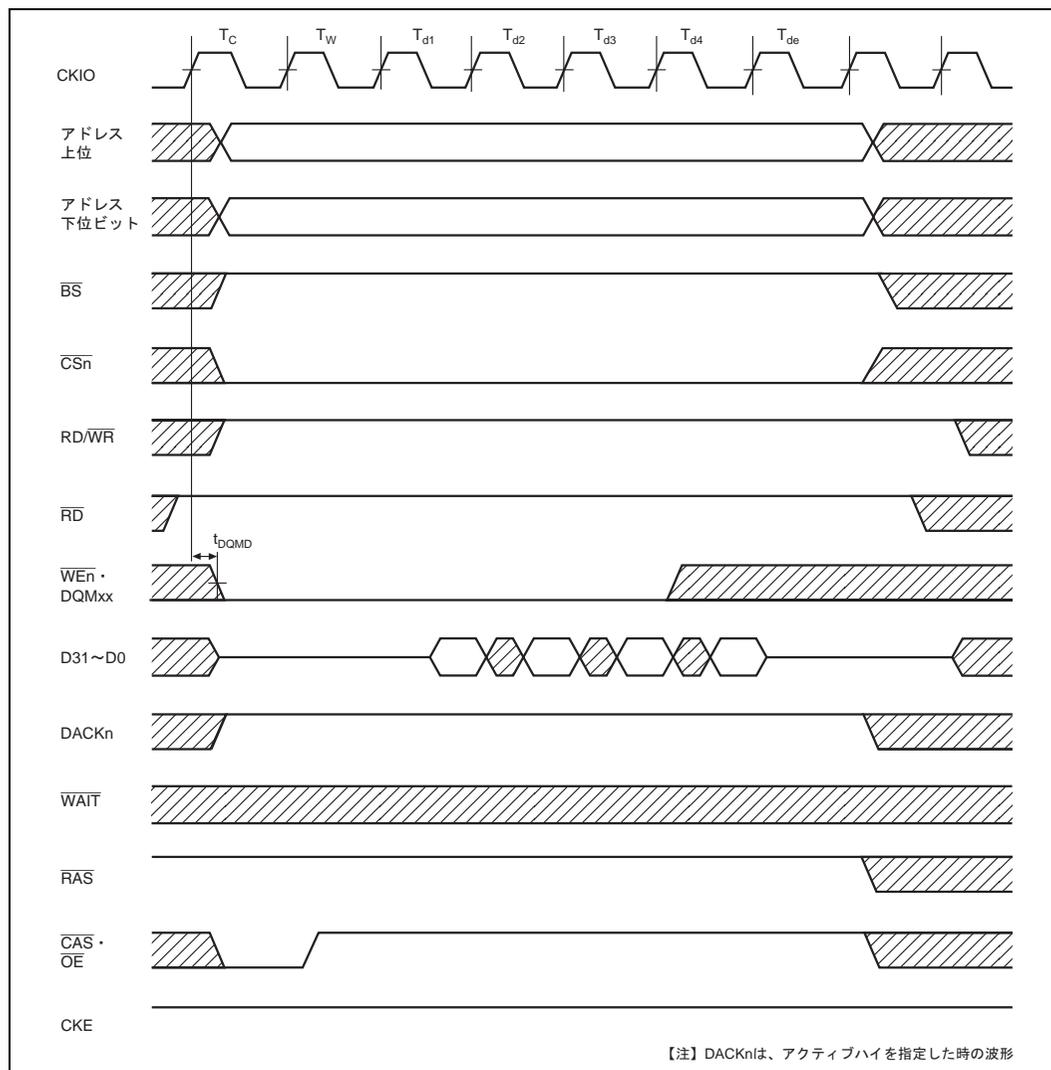


図 21.20 シンクロナス DRAM 読み出しバスサイクル  
(バンクアクティブ、同一ロウアクセス、CAS レイテンシ =2 サイクル)

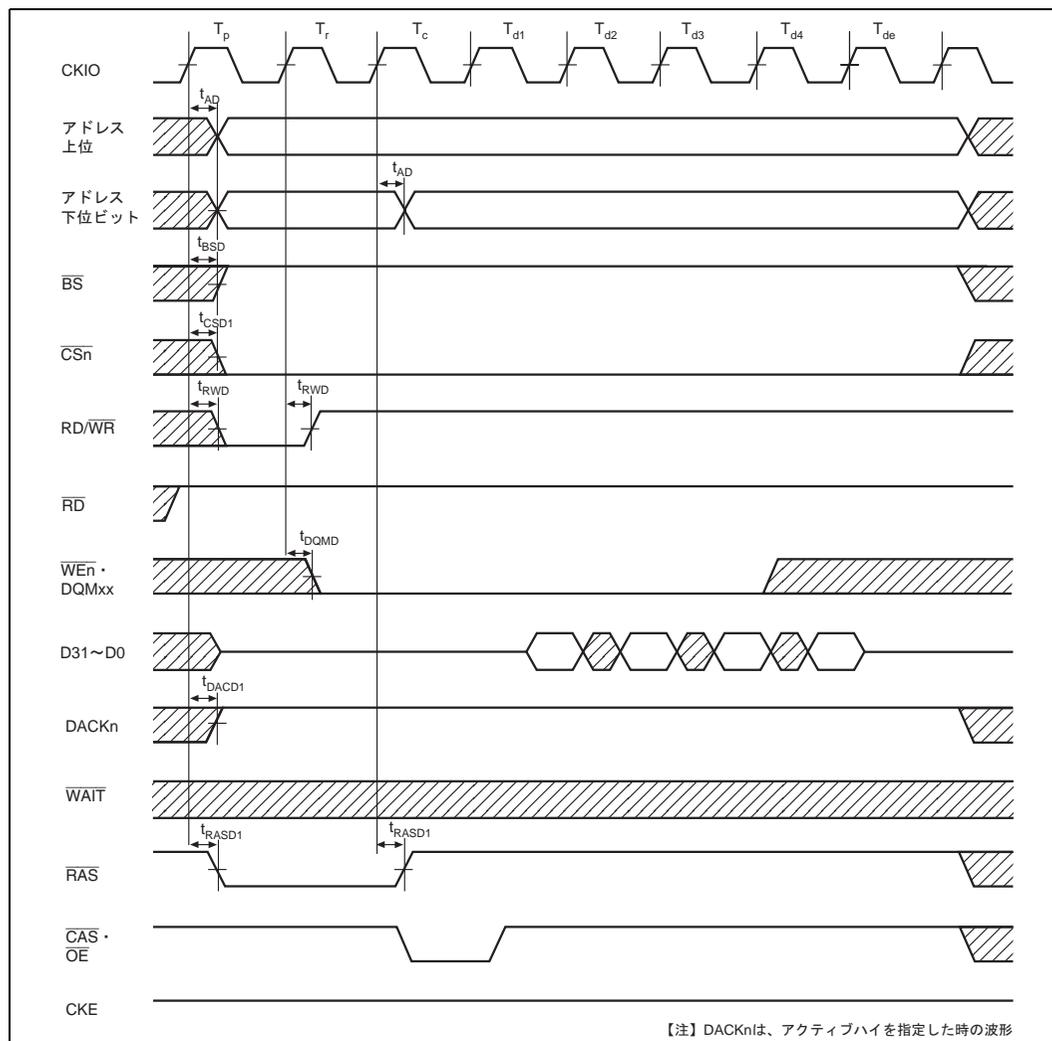


図 21.21 シンクロナス DRAM 読み出しバスサイクル  
 (バンクアクティブ、別口ウアクセス、TRP=1 サイクル、RCD=1 サイクル、CAS レイテンシ =1 サイクル)

21. 電気的特性

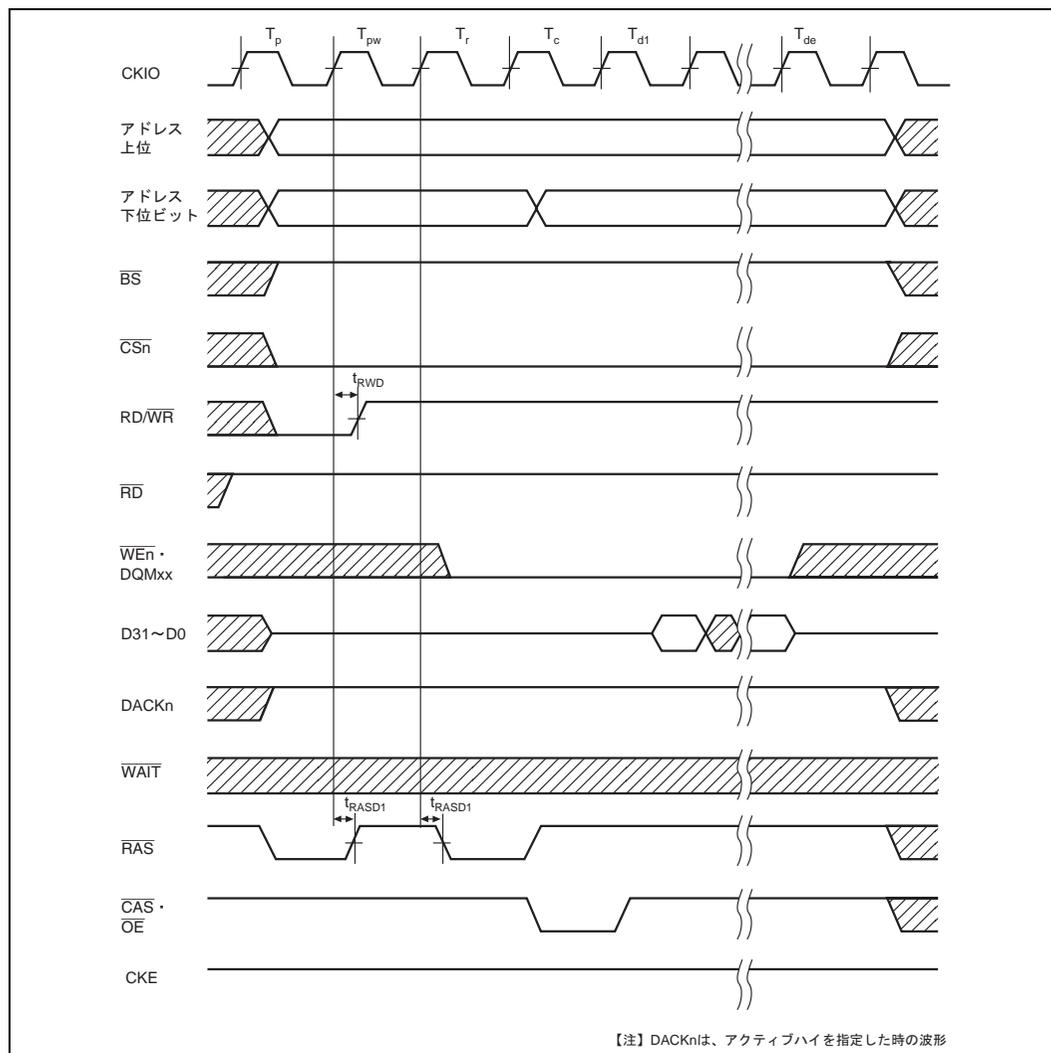


図 21.22 シンクロナス DRAM 読み出しバスサイクル  
 (バンクアクティブ、別口アクセス、TRP=2 サイクル、RCD=1 サイクル、CAS レイテンシ =1 サイクル)

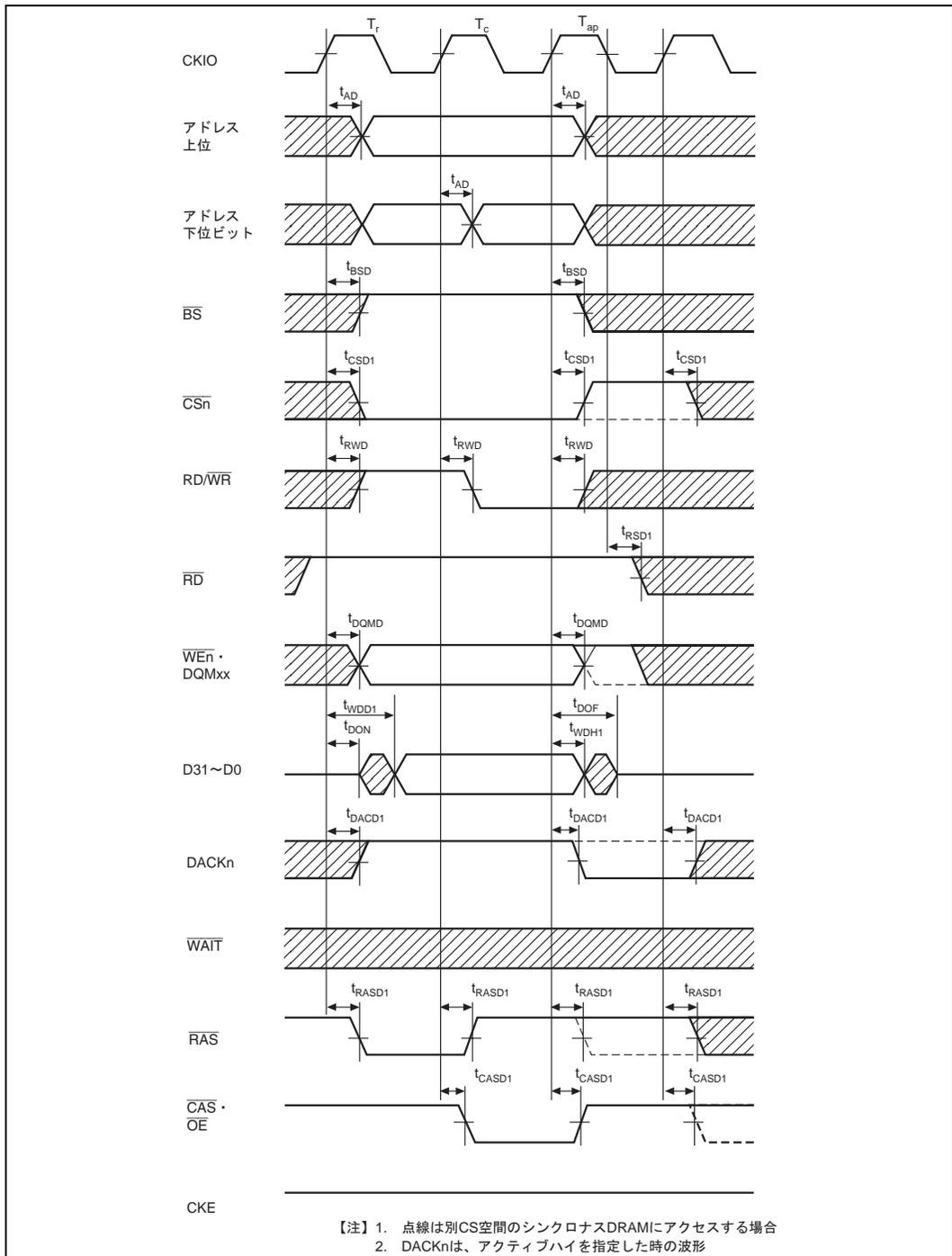


図 21.23 シンクロナス DRAM 書き込みバスサイクル  
 (RASD=0、RCD=1 サイクル、TRWL=1 サイクル)



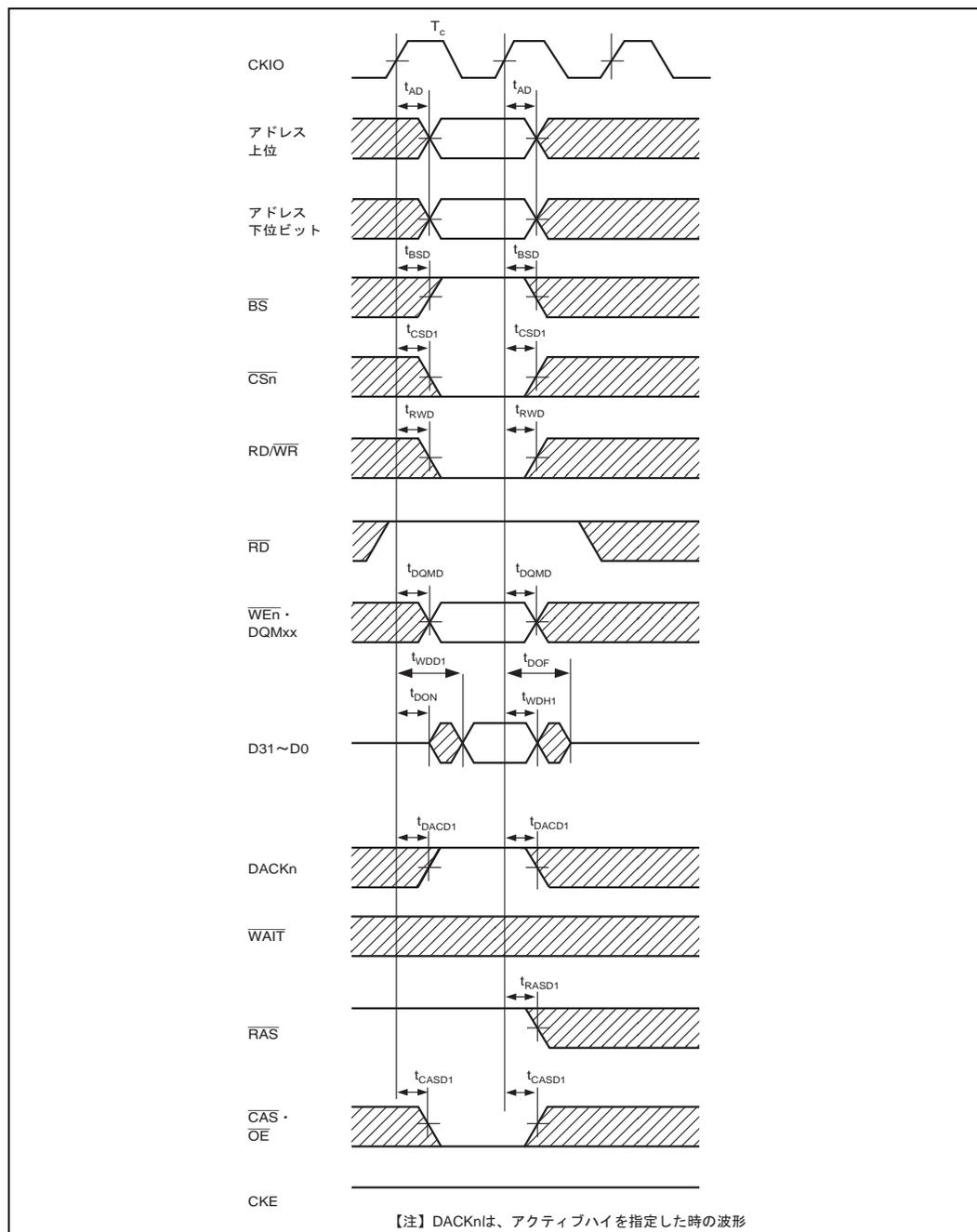


図 21.25 シンクロナス DRAM 書き込みバスサイクル  
(バンクアクティブ、同一口アクセス、I : E = 1 : 1 以外)

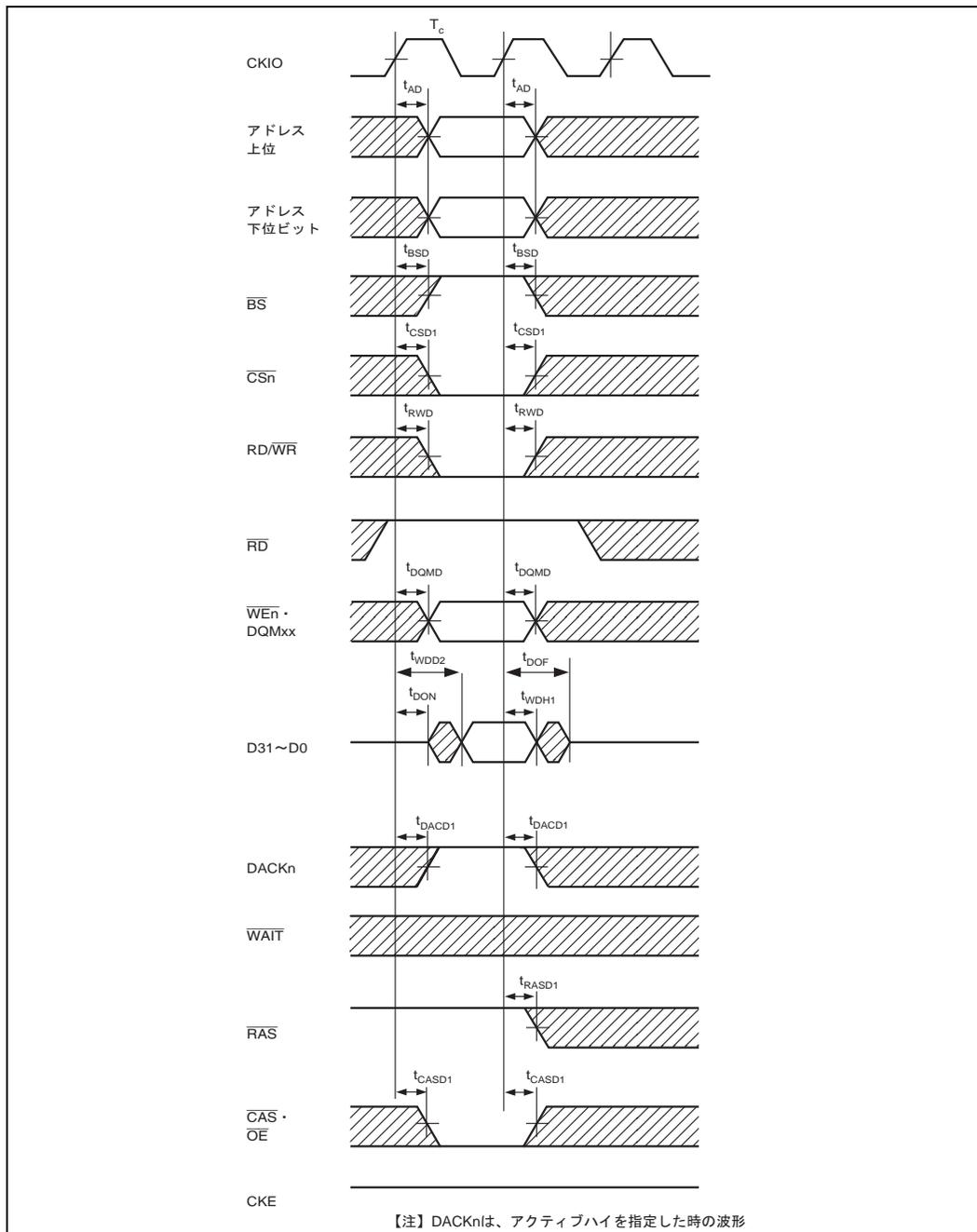


図 21.26 シンクロナス DRAM 書き込みサイクル  
(バンクアクティブ、同一ロウアクセス、I : E = 1 : 1)

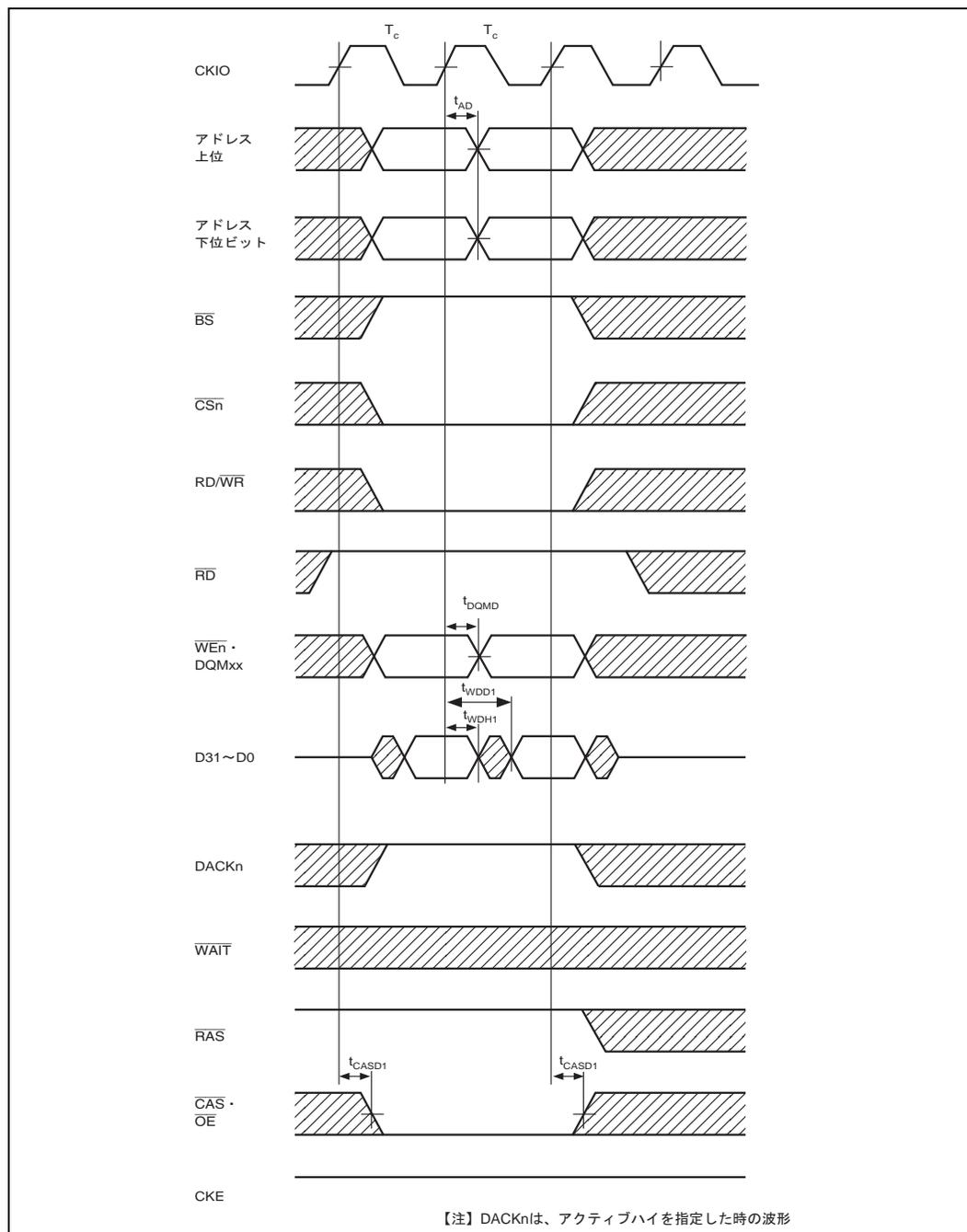


図 21.27 シンクロナス DRAM 連続書き込みサイクル  
(バンクアクティブ、同一ロウアクセス、I : E = 1 : 1 以外)

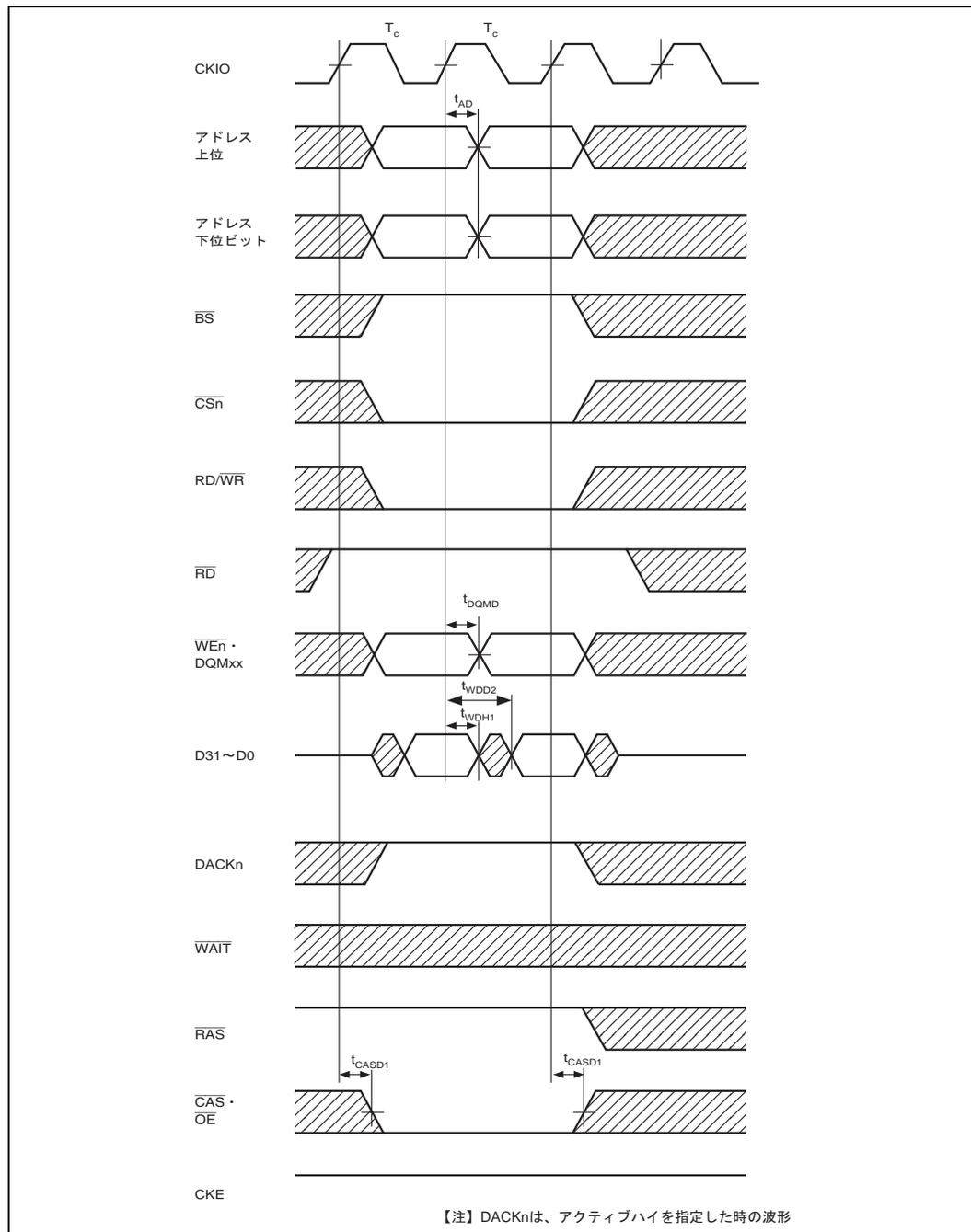


図 21.28 シンクロナス DRAM 連続書き込みサイクル  
(バンクアクティブ、同一ロウアクセス、I : E = 1 : 1)

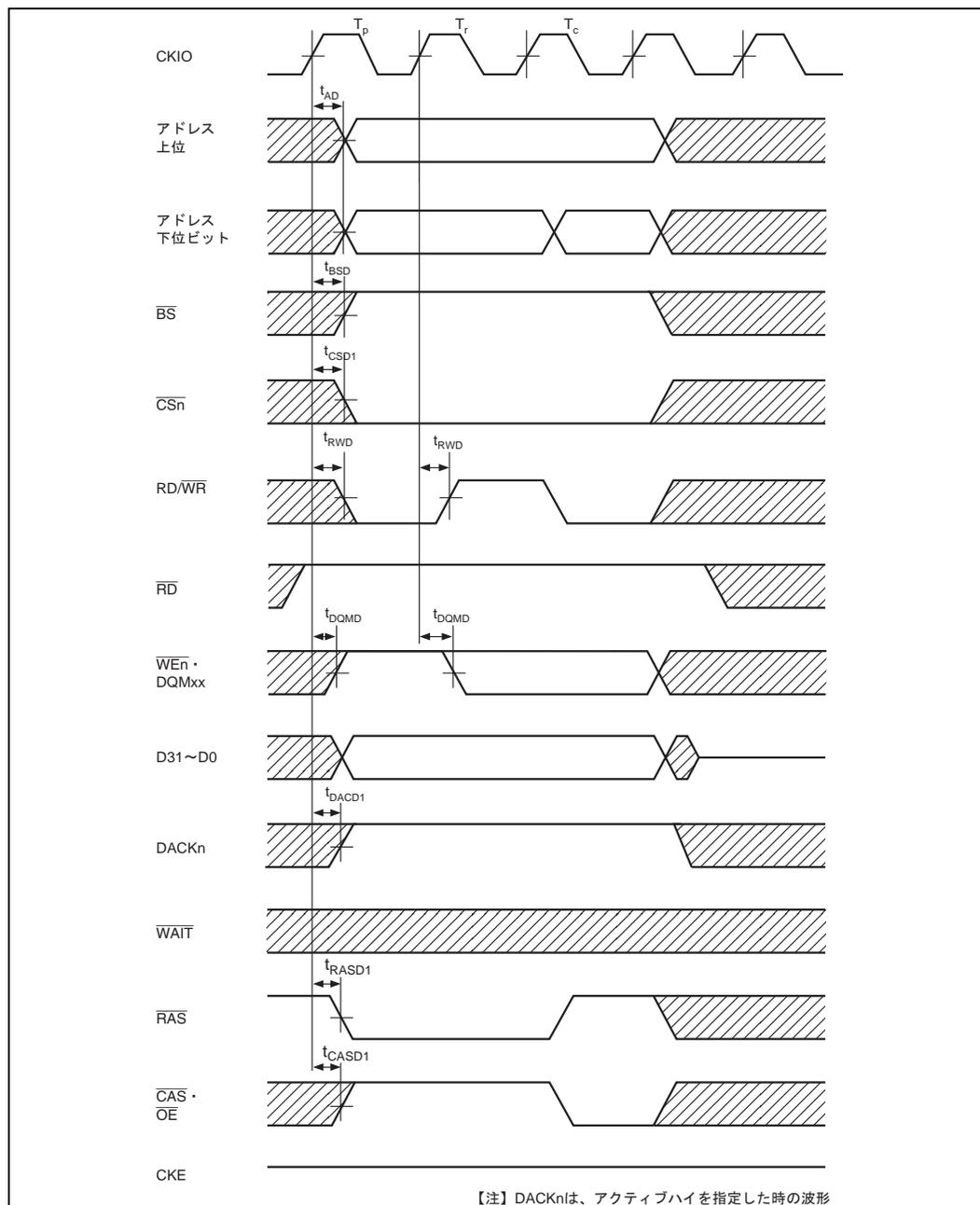


図 21.29 シンクロナス DRAM 書き込みバスサイクル  
(バンクアクティブ、別口ウアクセス、TRP=1 サイクル、RCD=1 サイクル)

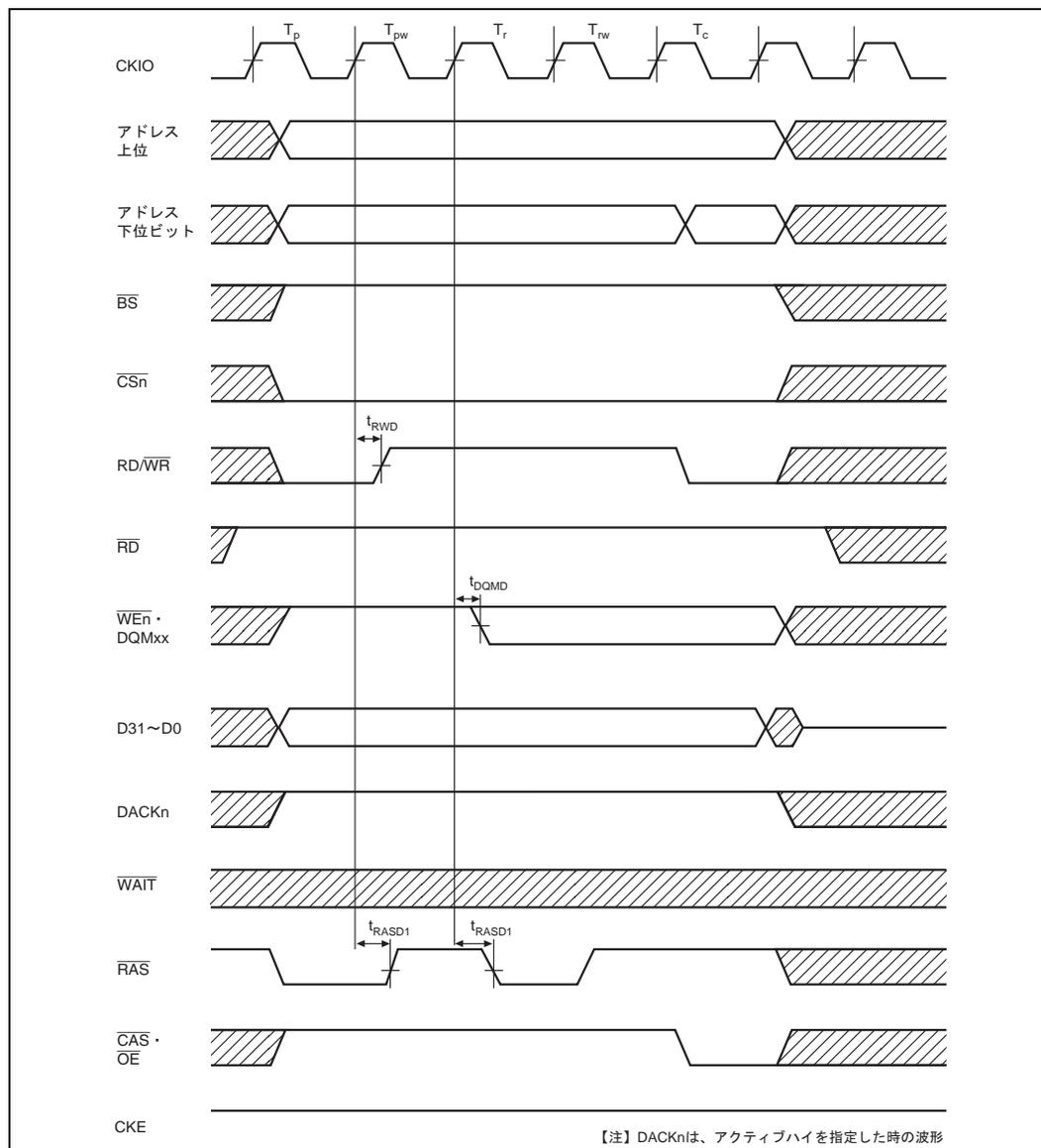


図 21.30 シンクロナス DRAM 書き込みバスサイクル  
 (バンクアクティブ、別口ウアクセス、TRP=2 サイクル、RCD=2 サイクル)

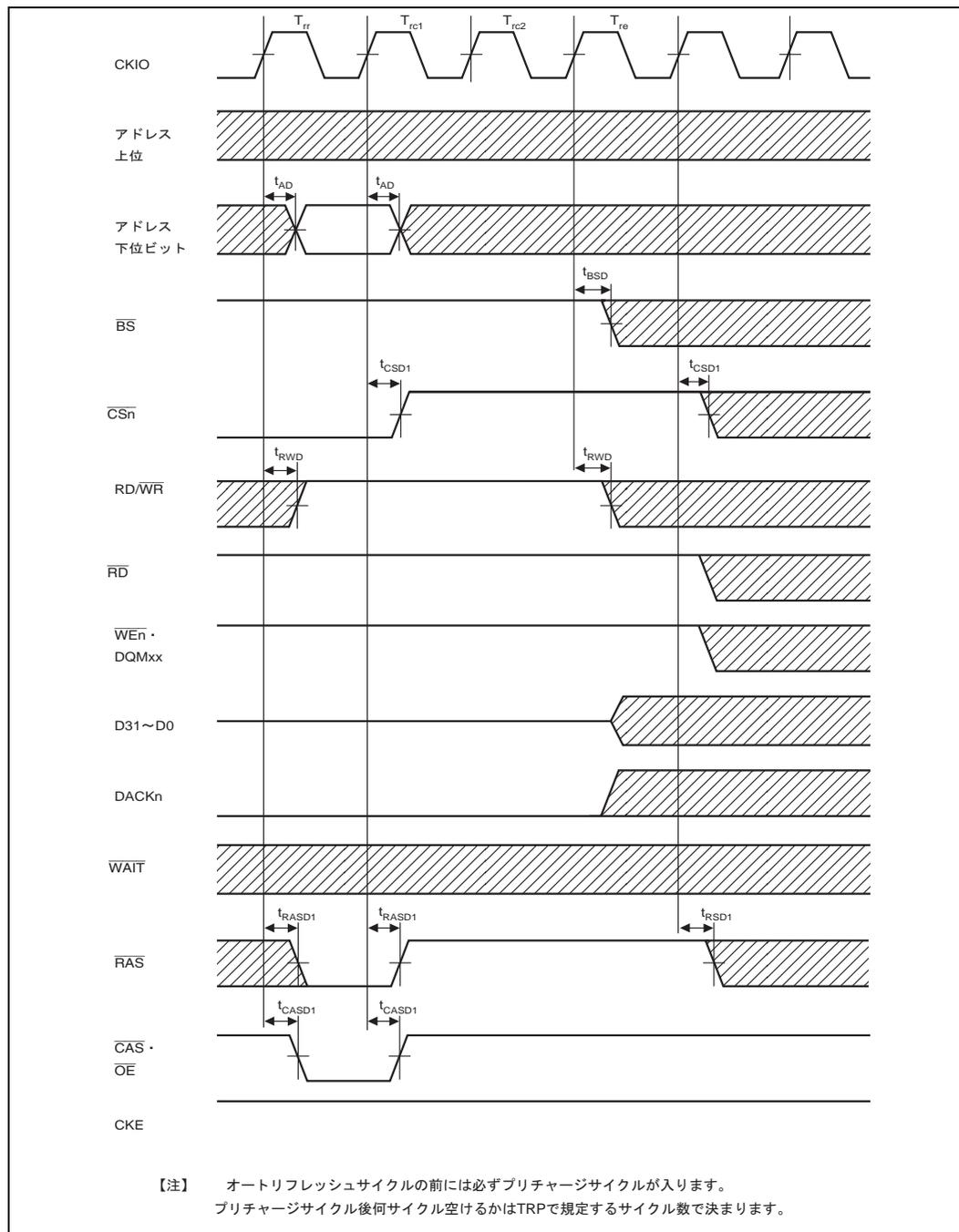


図 21.31 シンクロナス DRAM オートリフレッシュサイクル (TRAS=4 サイクル)

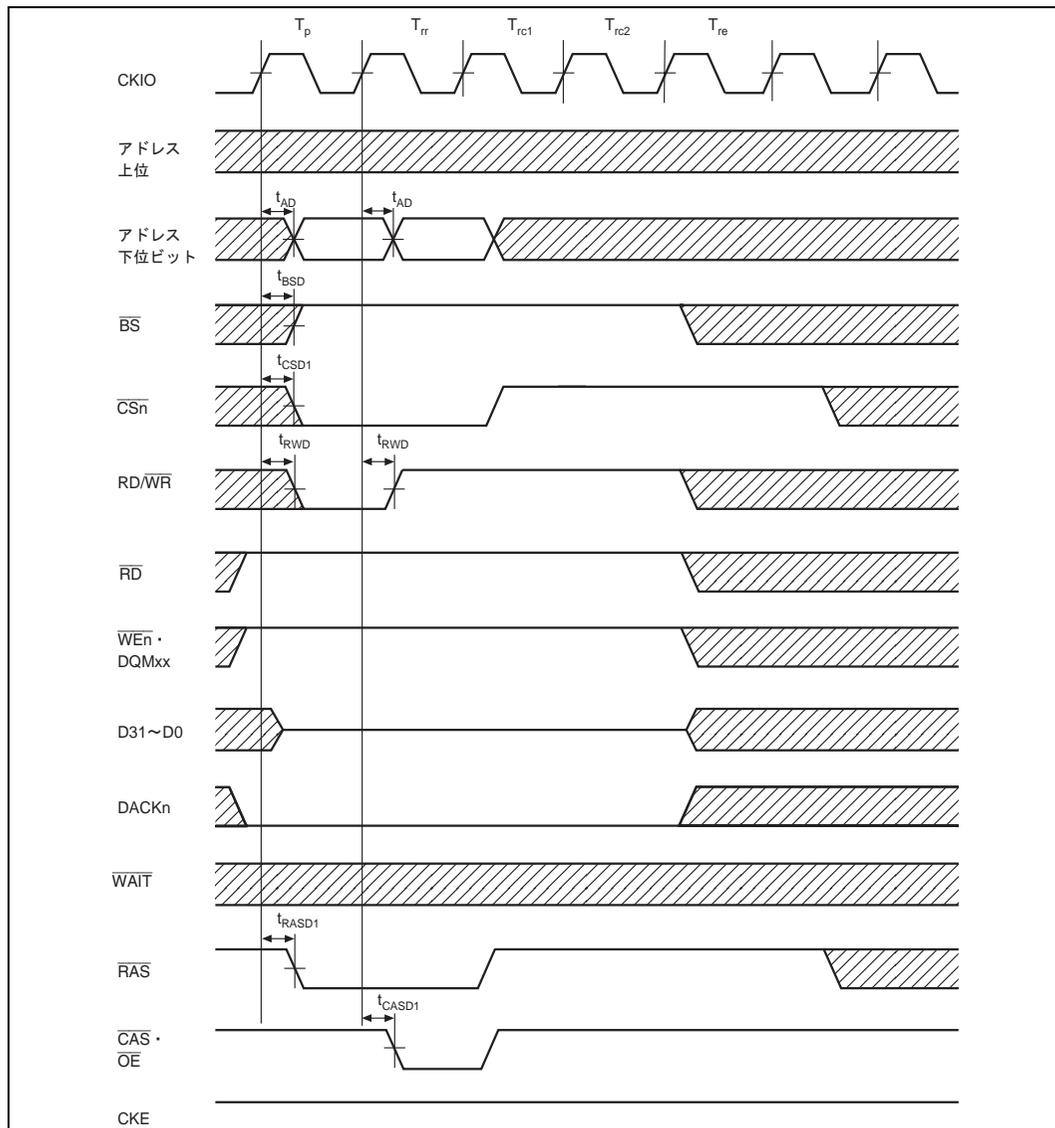


図 21.32 シンクロナス DRAM オートリフレッシュサイクル  
 (プリチャージサイクルから図示、TRP=1 サイクル、TRAS=4 サイクル)

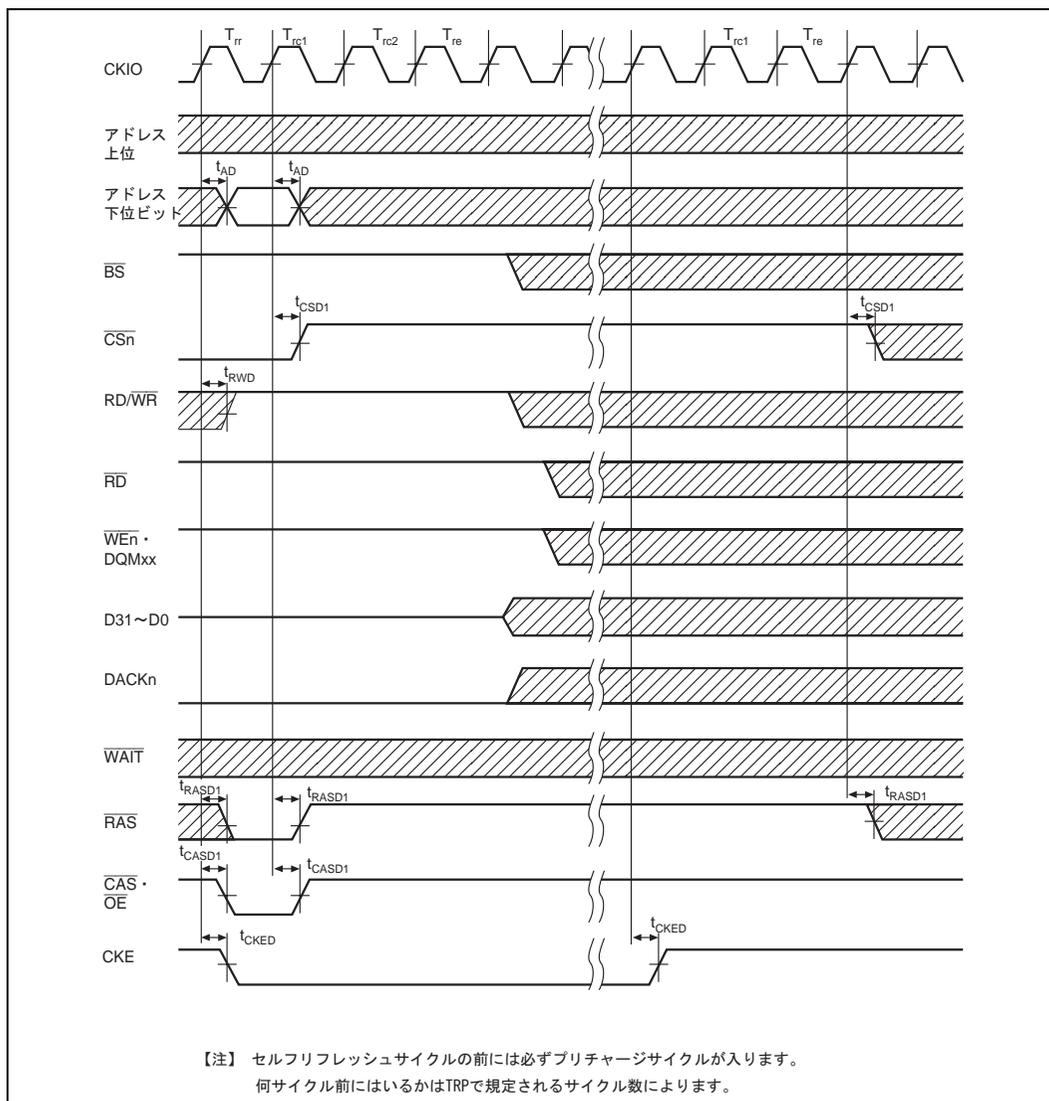


図 21.33 シンクロナス DRAM セルフリフレッシュサイクル (TRAS=3)

21. 電気的特性

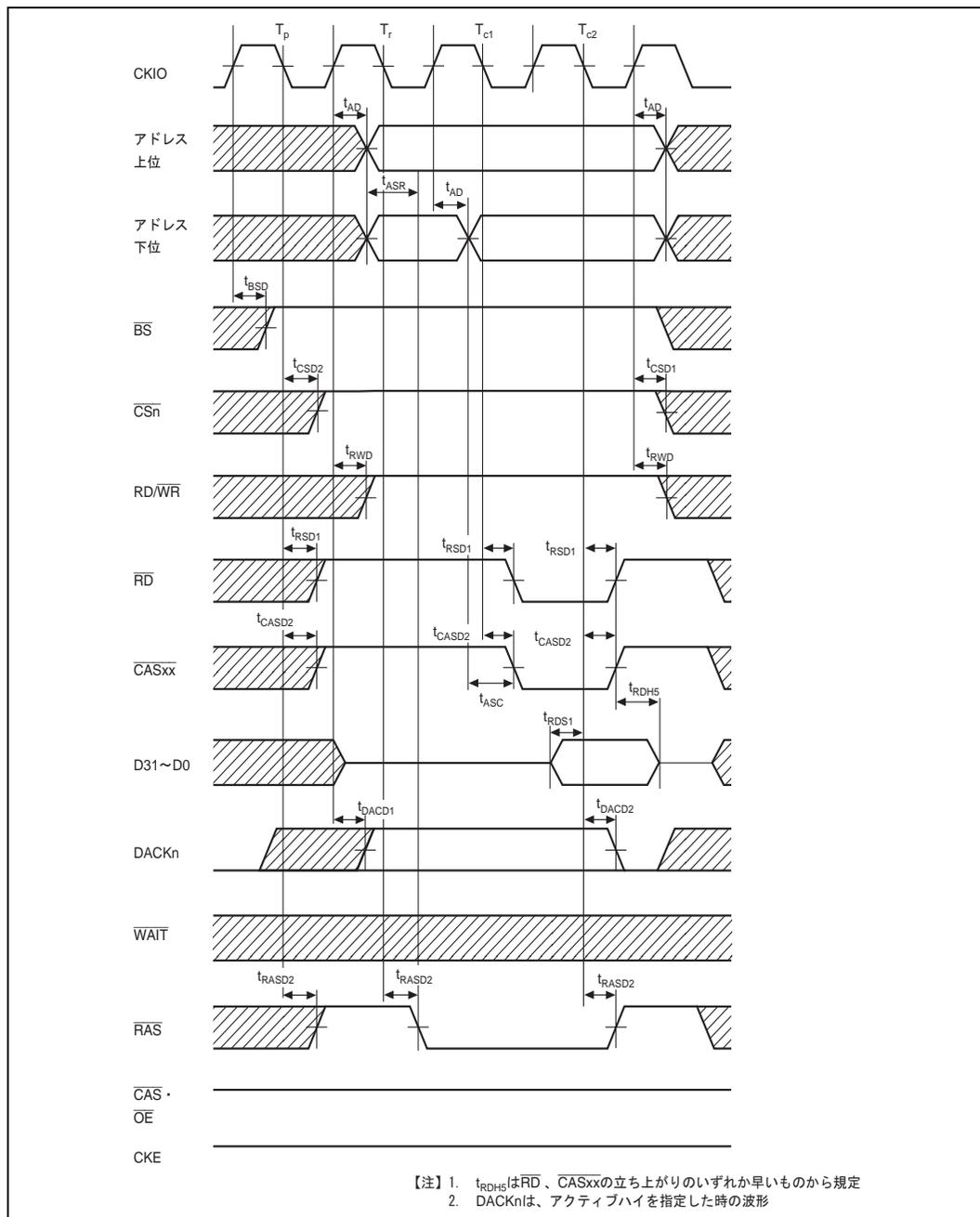


図 21.34 DRAM 読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト)



21. 電気的特性

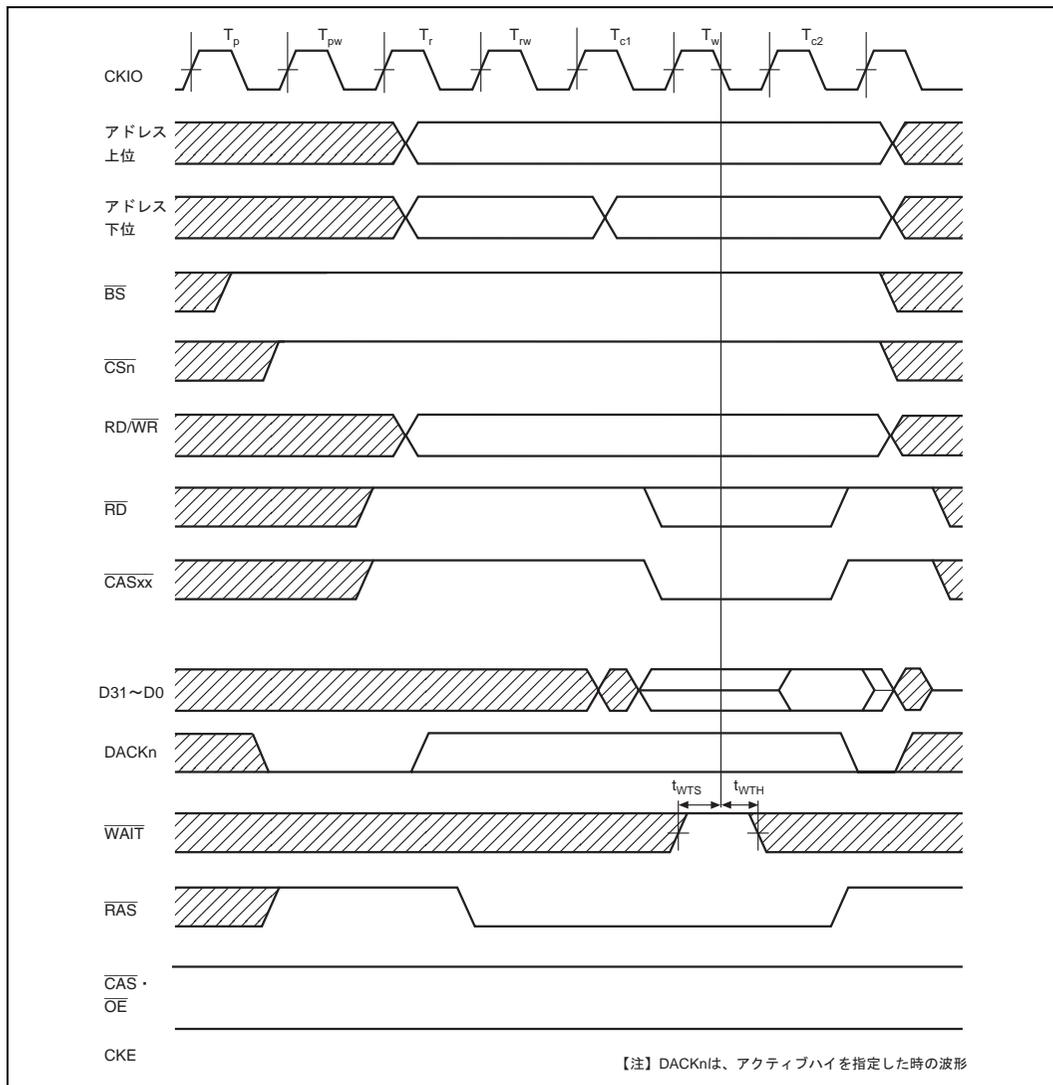


図 21.36 DRAM バスサイクル (TRP=2 サイクル、RCD=2 サイクル、1 ウェイト)

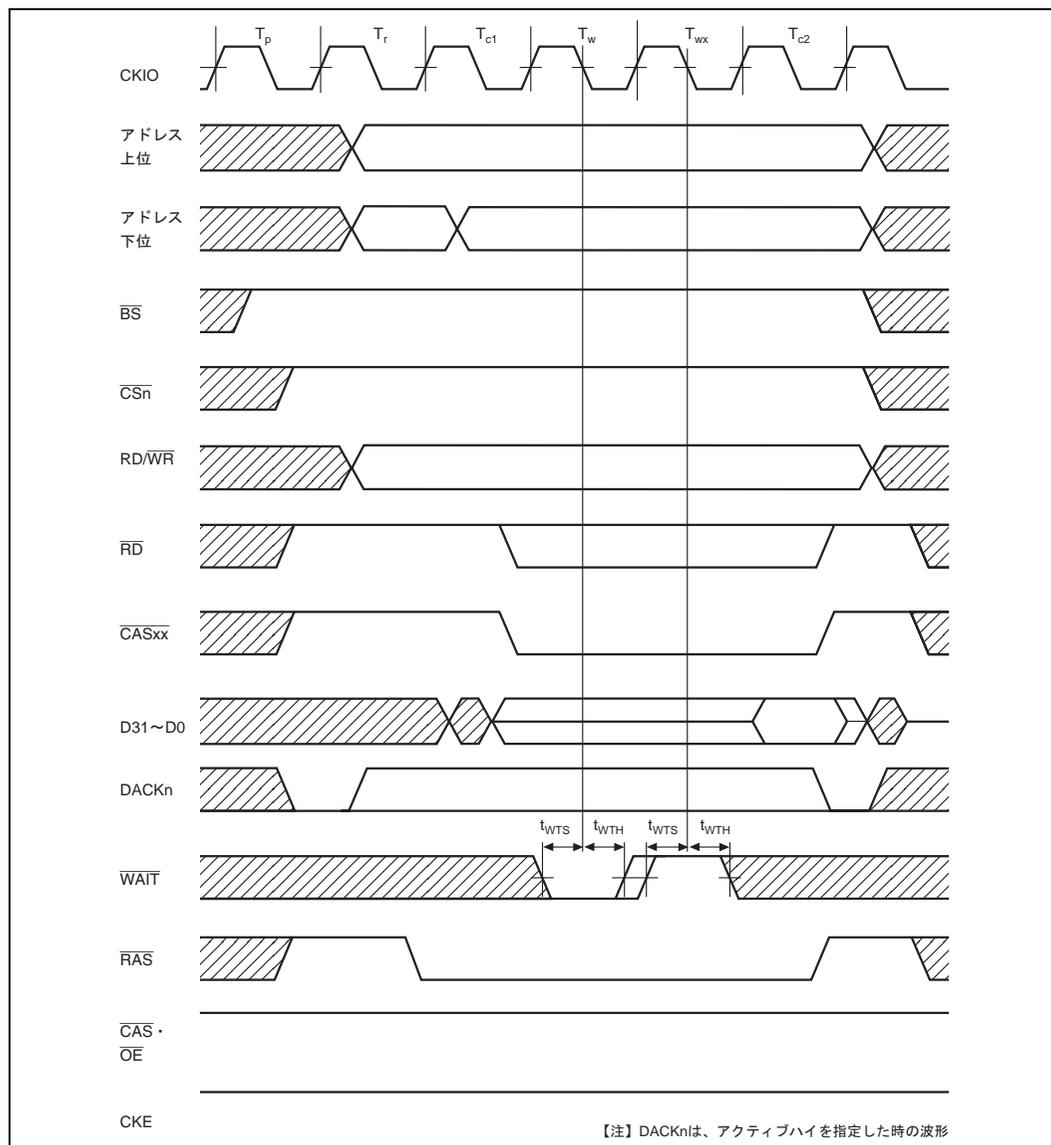


図 21.37 DRAM バスサイクル (TRP=1 サイクル、RCD=1 サイクル、外部ウェイト入力)

## 21. 電気的特性

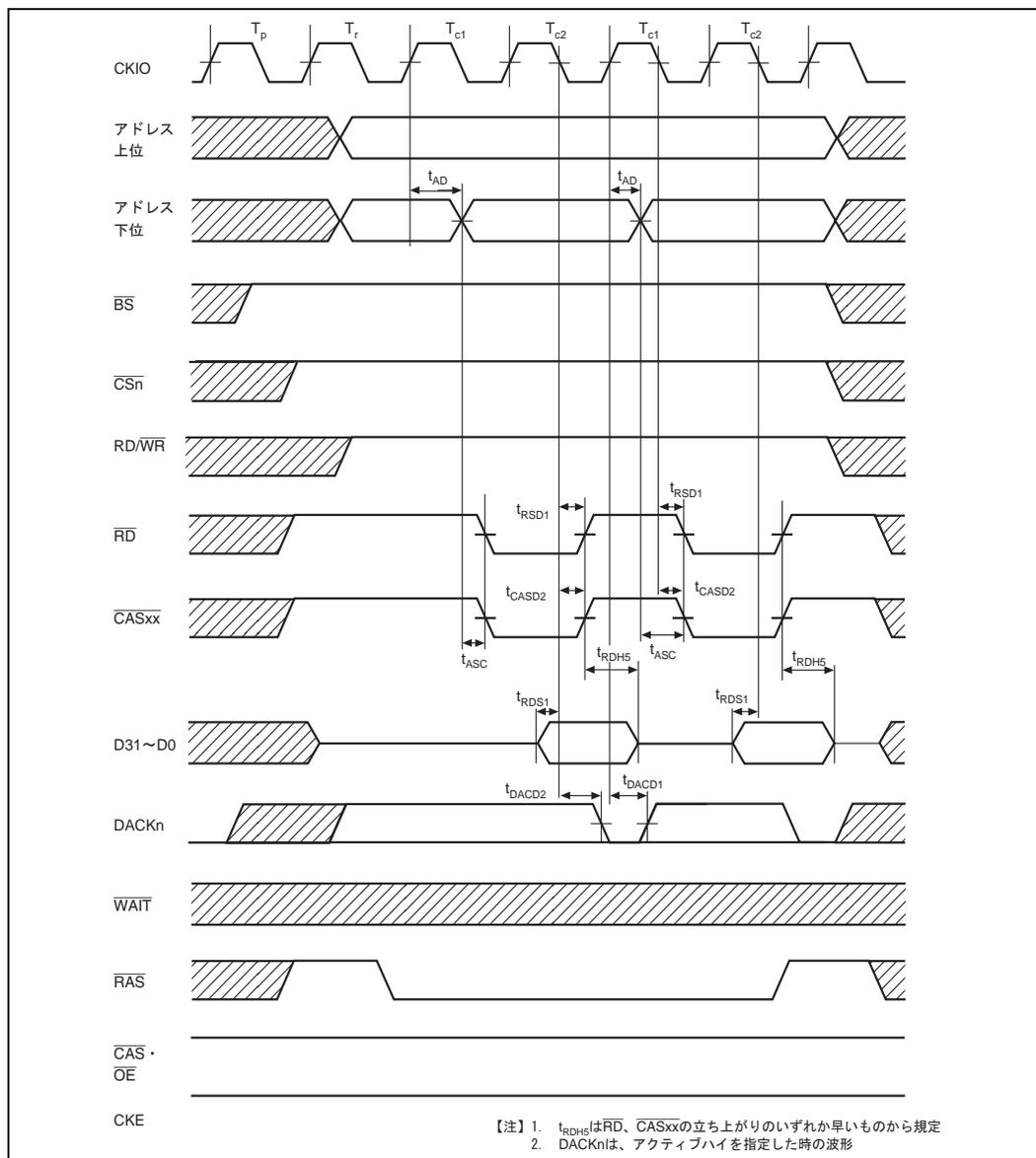


図 21.38 DRAM バースト読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト)

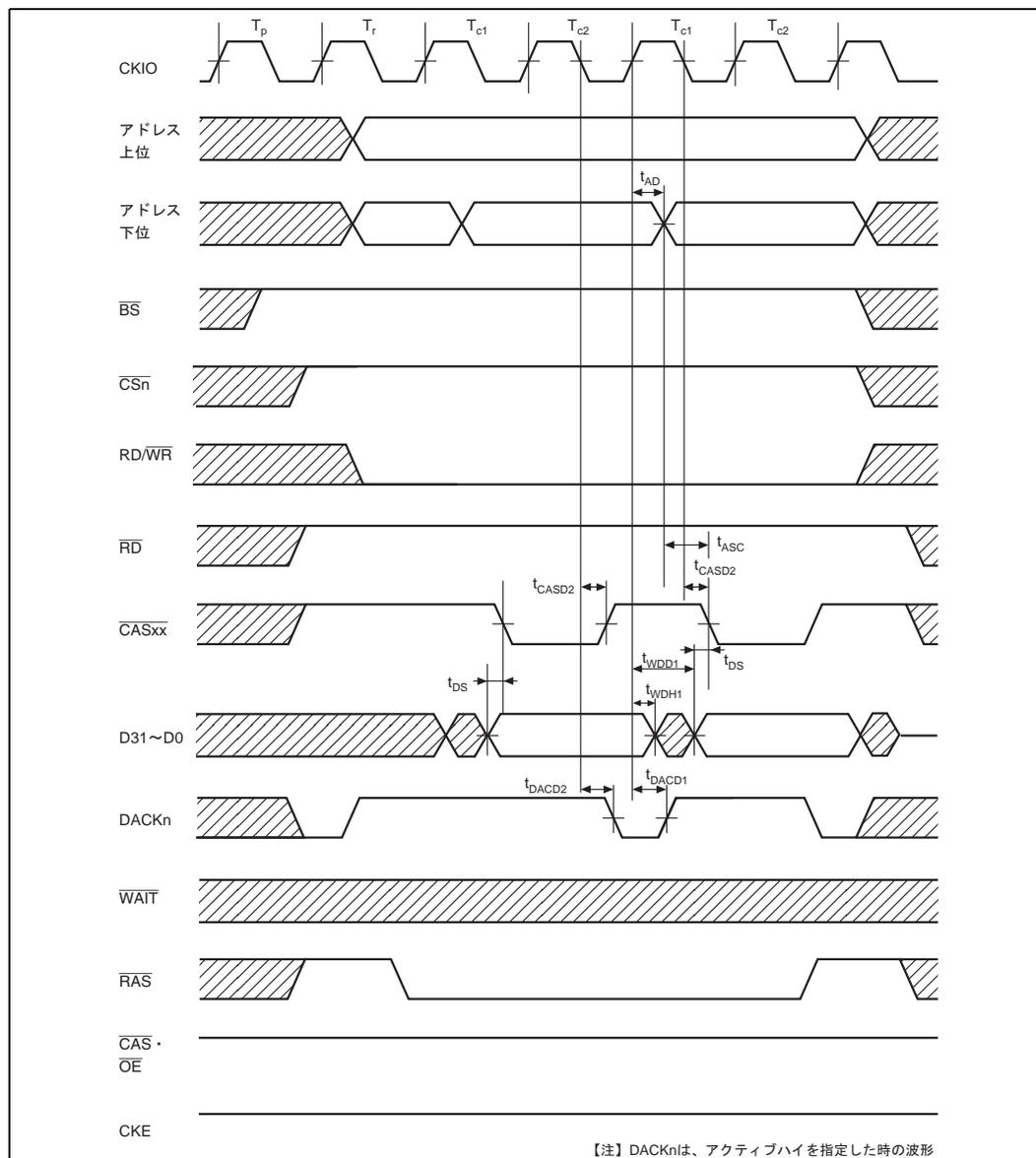


図 21.39 DRAM バースト書き込みサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト)

21. 電気的特性

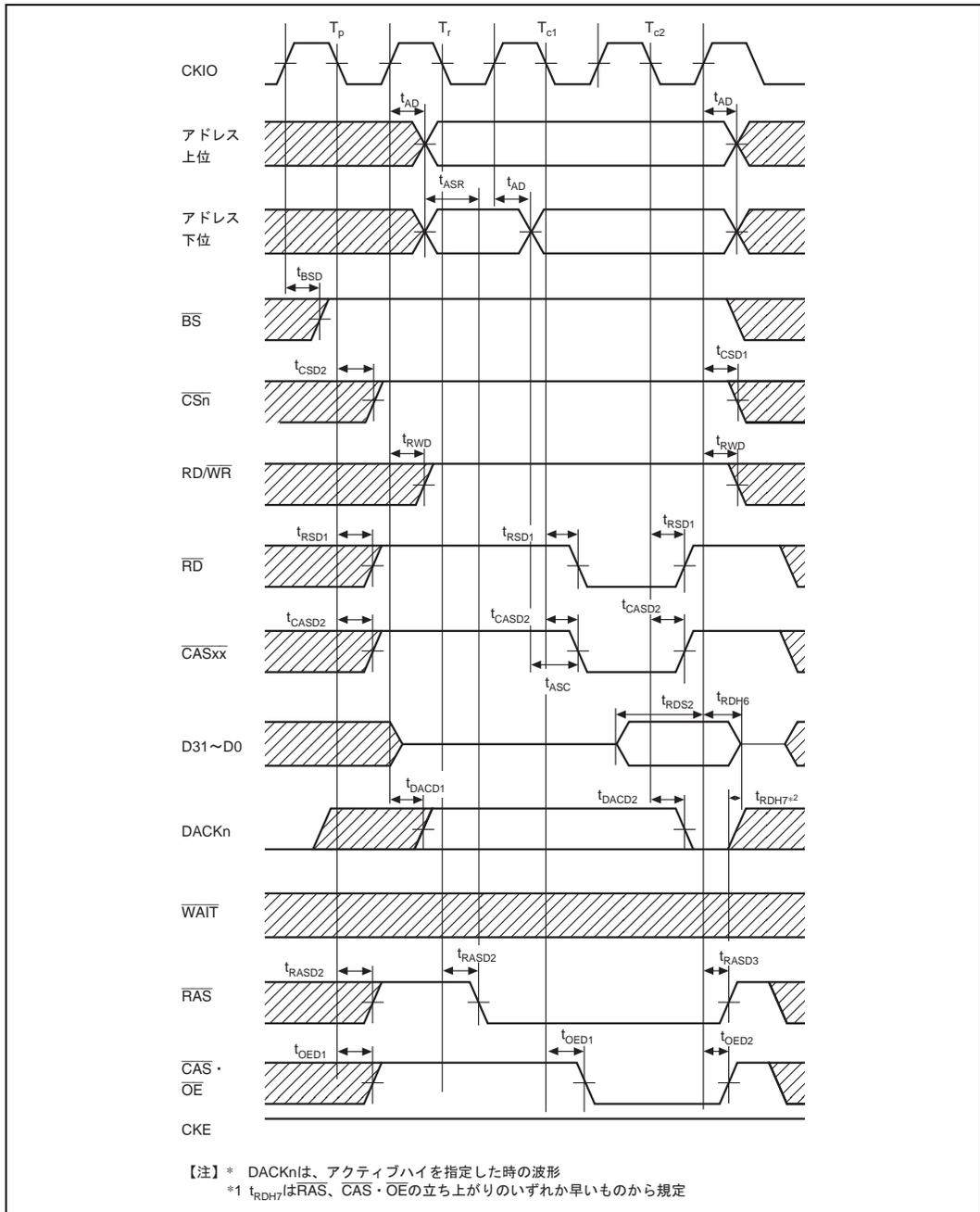


図 21.40 EDO リードサイクル (TRP = 1 サイクル、RCD = 1 サイクル、ノーウェイト)

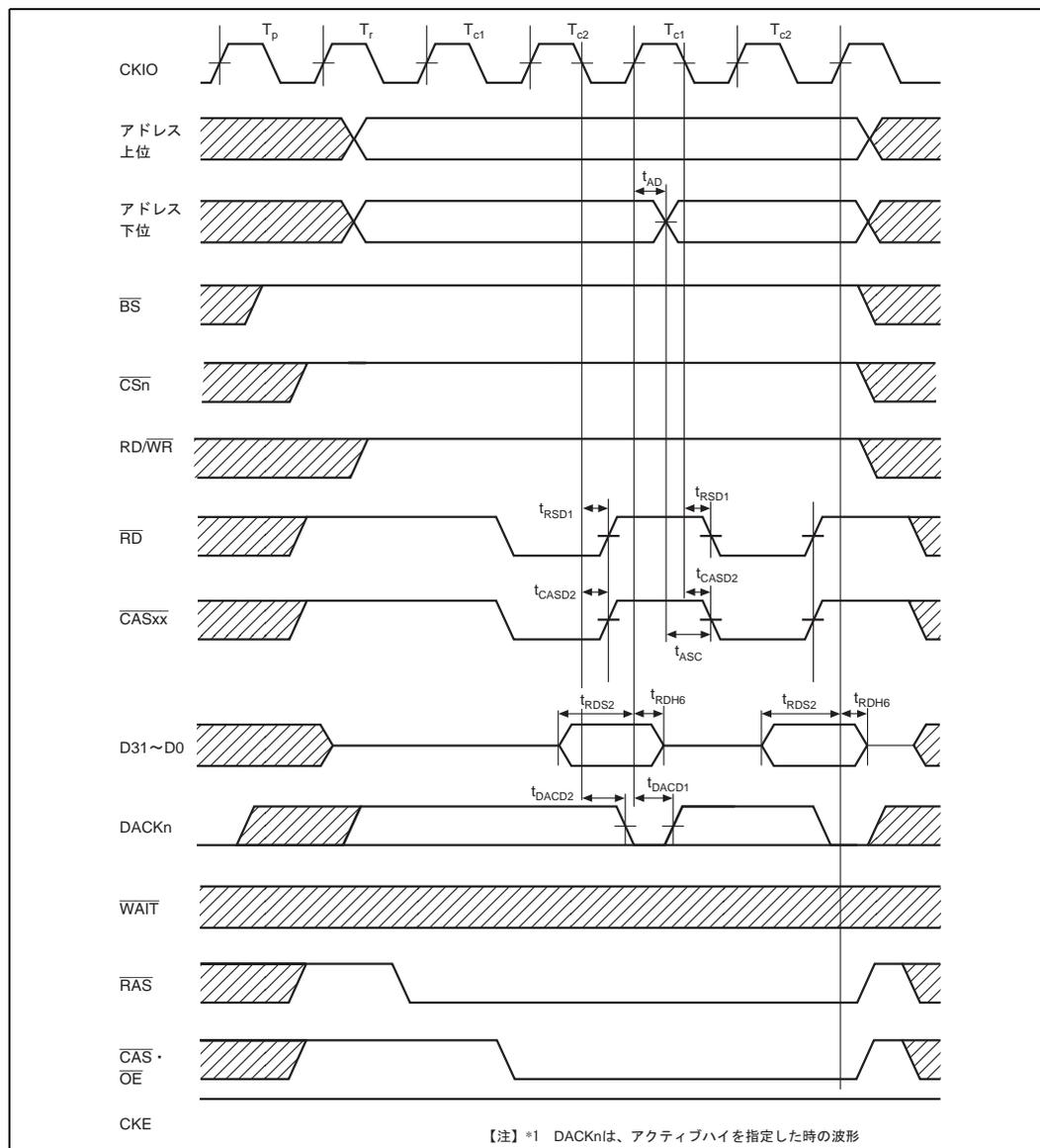


図 21.41 EDO バースト読み出しサイクル (TRP=1 サイクル、RCD=1 サイクル、ノーウェイト)

21. 電気的特性

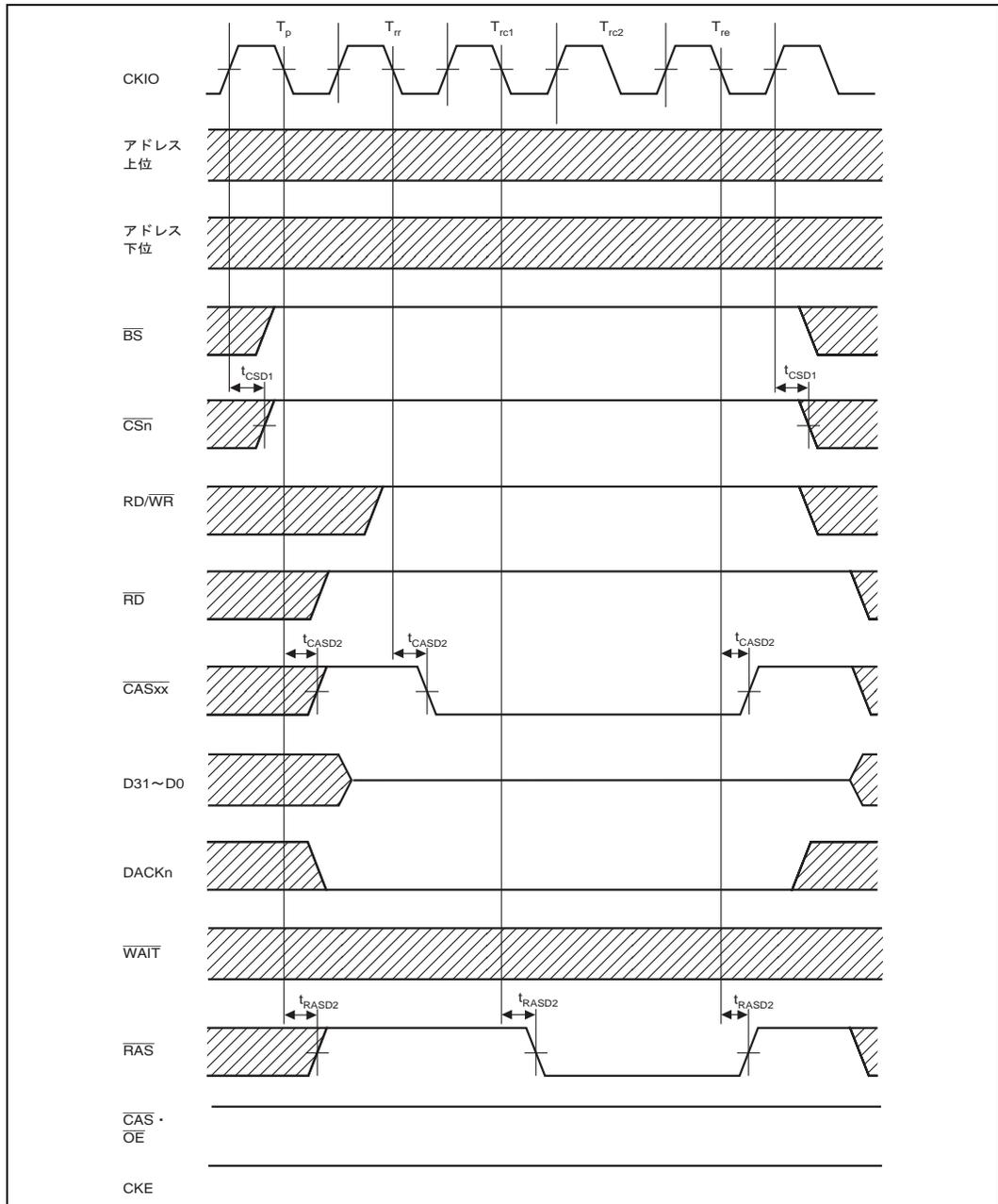


図 21.42 DRAM CAS ピフォ RAS リフレッシュサイクル (TRP=1 サイクル、TRAS=2 サイクル)

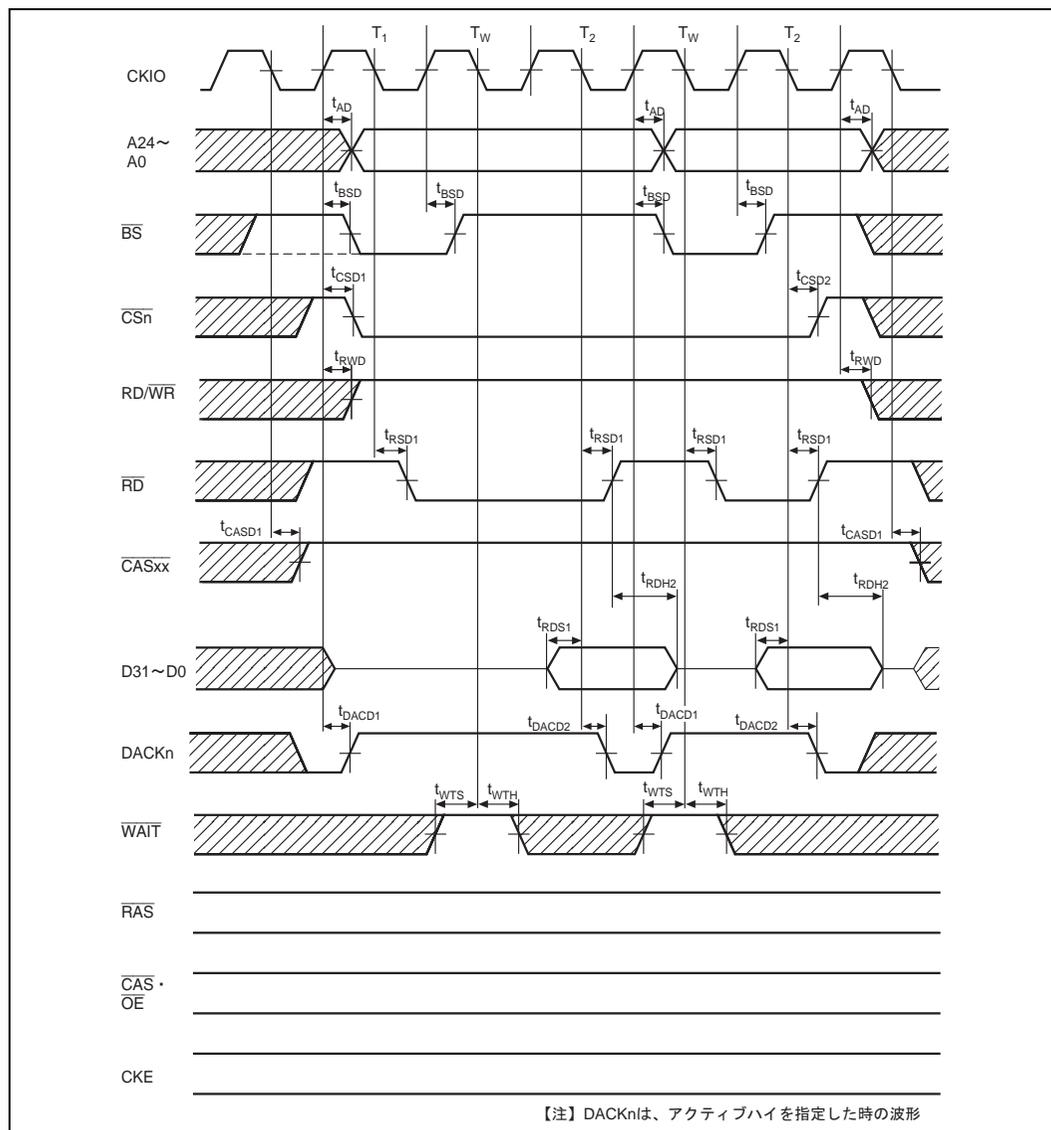


図 21.43 バーストROM 読み出しサイクル (ウェイト=1)

21. 電気的特性

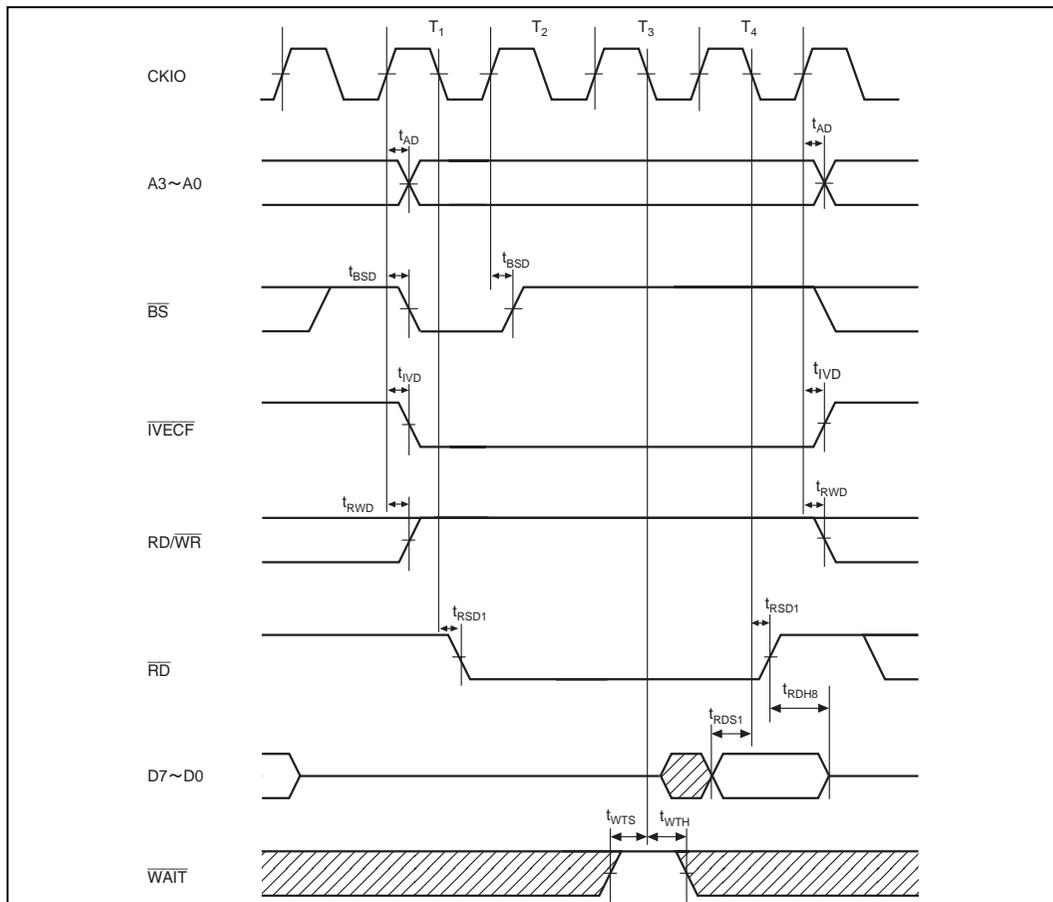


図 21.44 割り込みベクタフェッチサイクル (ノーウェイト、I : E = 1 : 1)

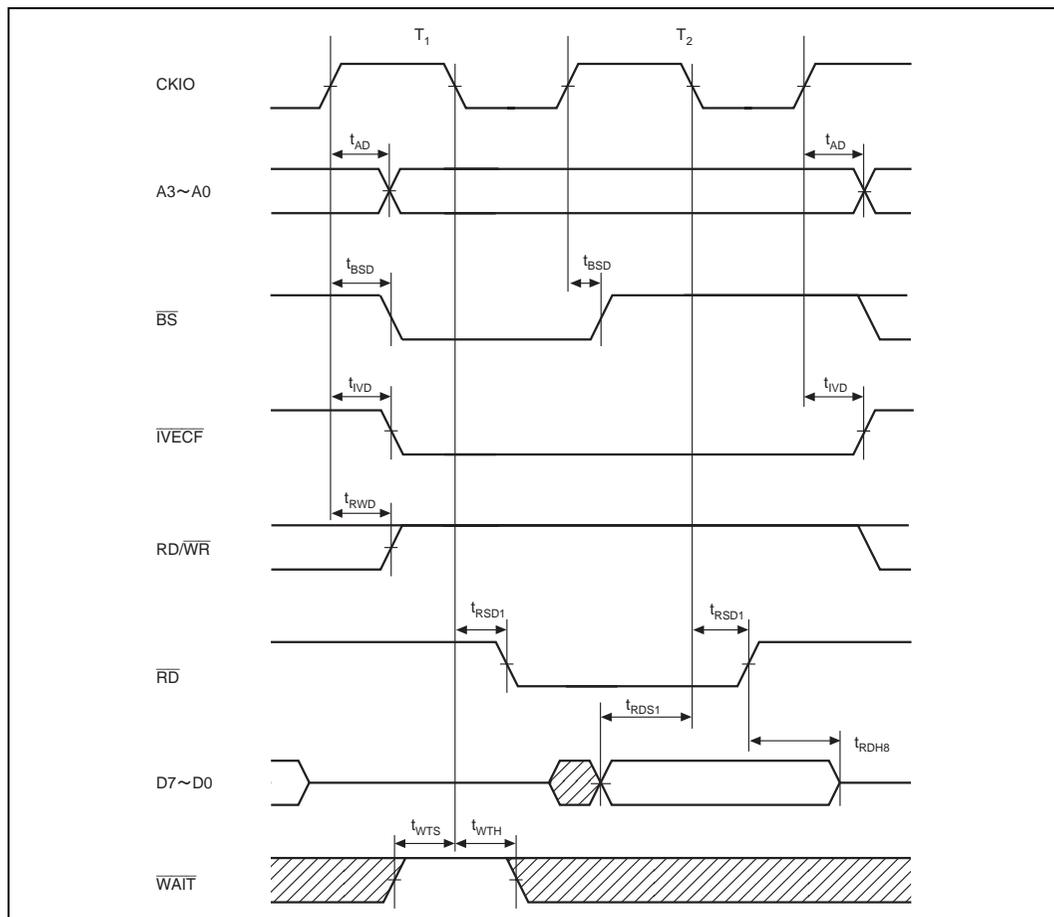


図 21.45 割り込みベクタフェッチサイクル (ノーウェイト、I : E = 1 : 1 以外)

21. 電気的特性

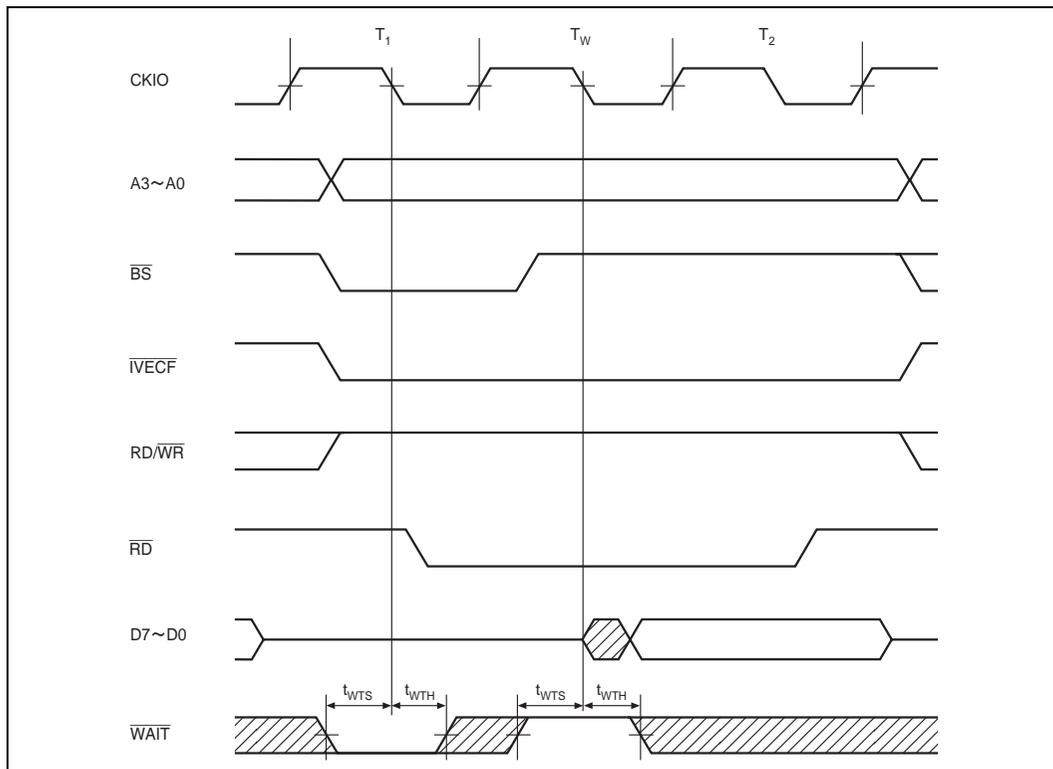


図 21.46 割り込みベクタフェッチサイクル (外部ウェイト入力 I : E = 1 : 1 以外)

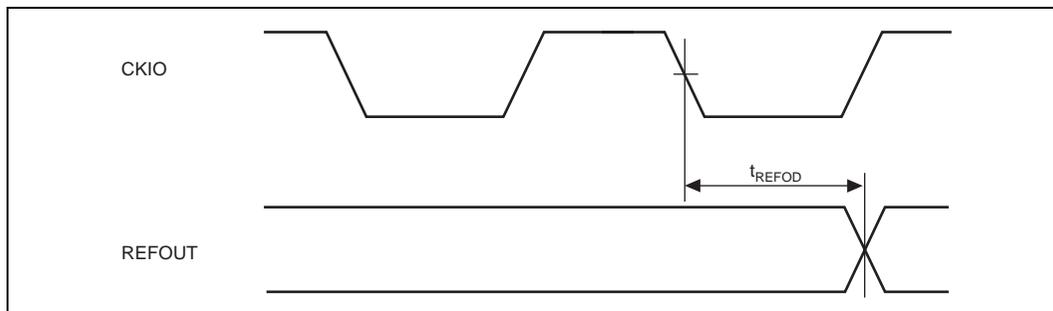


図 21.47 REFOUT 遅延時間

## 21.3.4 ダイレクトメモリアクセスコントローラタイミング

表 21.8 ダイレクトメモリアクセスコントローラタイミング

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} \quad V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
DREQ0、DREQ1 セットアップ時間	$t_{DRQS}$	10		ns	21.48
DREQ0、DREQ1 ホールド時間	$t_{DRQH}$	5		ns	

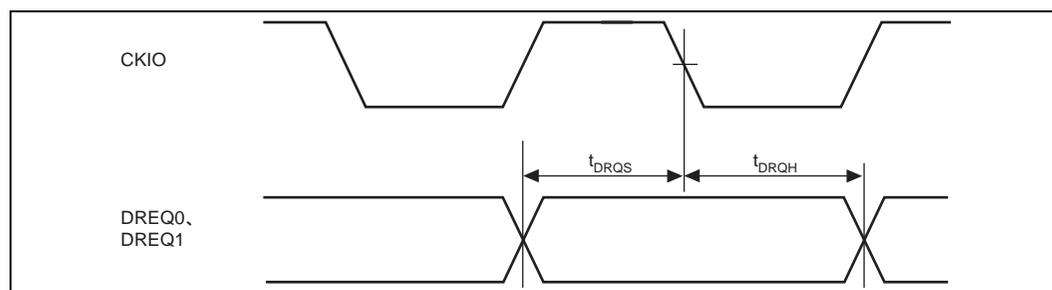


図 21.48 DREQ0、DREQ1 入力タイミング

## 21.3.5 フリーランニングタイマタイミング

表 21.9 フリーランニングタイマタイミング

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} \quad V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
アウトプットコンペア出力遅延時間	$t_{FOCD}$		100	ns	21.49、50
インプットキャプチャ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ )	$t_{FICS}$	50		ns	21.49
インプットキャプチャ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 2$ )	$t_{FICS}$	$t_{cyc}+50$		ns	21.50
インプットキャプチャ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 4$ )	$t_{FICS}$	$3t_{cyc}+50$		ns	21.50
インプットキャプチャ入力ホールド時間	$t_{FICH}$	50		ns	21.49、50
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ )	$t_{FCKS}$	50		ns	21.51
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 2$ )	$t_{FCKS}$	$t_{cyc}+50$		ns	21.52
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 4$ )	$t_{FCKS}$	$3t_{cyc}+50$		ns	21.52
タイマクロックパルス幅 (単エッジ指定)	$t_{FCKWH}$	4.5		$t_{P_{cyc}}$	21.50、52
タイマクロックパルス幅 (両エッジ指定)	$t_{FCKWL}$	8.5		$t_{P_{cyc}}$	

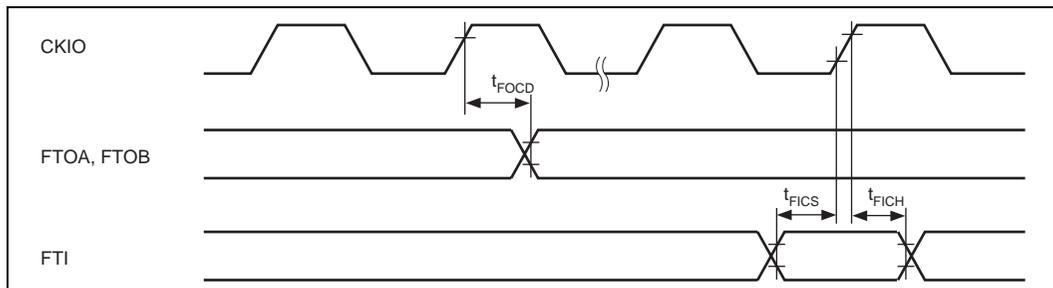


図 21.49 FRT 入出力タイミング ( $t_{Ejyc} : t_{Pjyc} = 1 : 1$ )

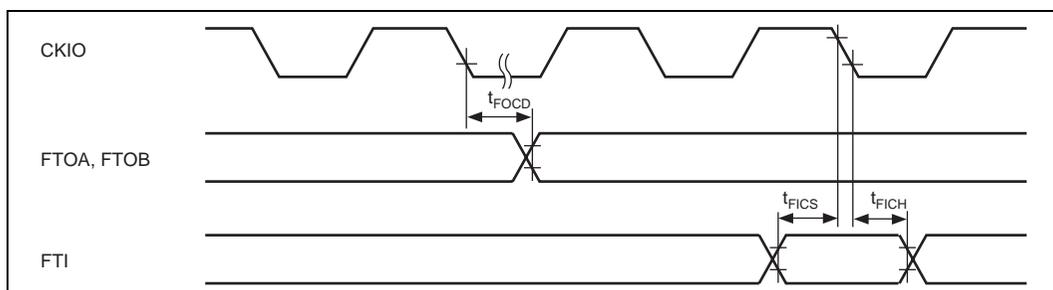


図 21.50 FRT 入出力タイミング ( $t_{Ejyc} : t_{Pjyc} = 1 : 1$  以外)

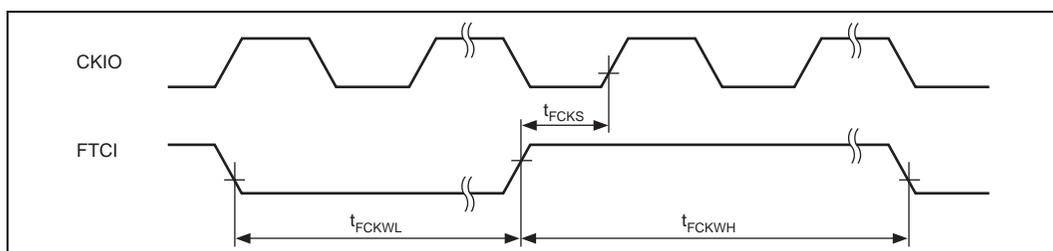


図 21.51 FRT クロック入力タイミング ( $t_{Ejyc} : t_{Pjyc} = 1 : 1$ )

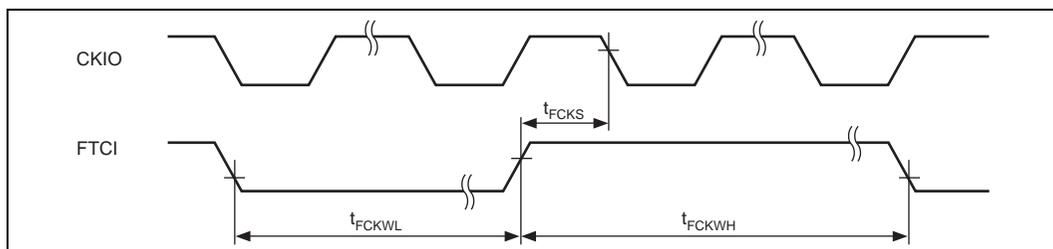


図 21.52 FRT クロック入力タイミング ( $t_{Ejyc} : t_{Pjyc} = 1 : 1$  以外)

## 21.3.6 シリアルコミュニケーションインタフェースタイミング

表 21.10 シリアルコミュニケーションインタフェースタイミング

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
入力クロックサイクル	$t_{scyc}$	4		$t_{pcyc}$	21.53
入力クロックサイクル (クロック同期)	$t_{scyc}$	6		$t_{pcyc}$	21.54
入力クロックパルス幅	$t_{sckw}$	0.4	0.6	$t_{scyc}$	21.53
送信データ遅延時間 (クロック同期)	$t_{txd}$		100	ns	21.54
受信データセットアップ時間 (クロック同期)	$t_{rxs}$	100		ns	
受信データホールド時間 (クロック同期)	$t_{rxh}$	100		ns	
RTS 遅延時間	$t_{rtsd}$		100	ns	21.55
CTS セットアップ時間 (クロック同期)	$t_{ctss}$	100		ns	
CTS ホールド時間 (クロック同期)	$t_{ctsh}$	100		ns	

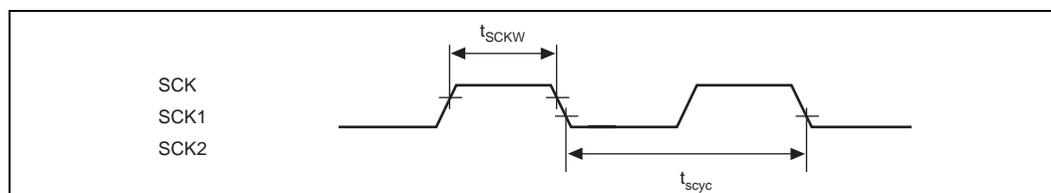


図 21.53 入力クロック入出力タイミング

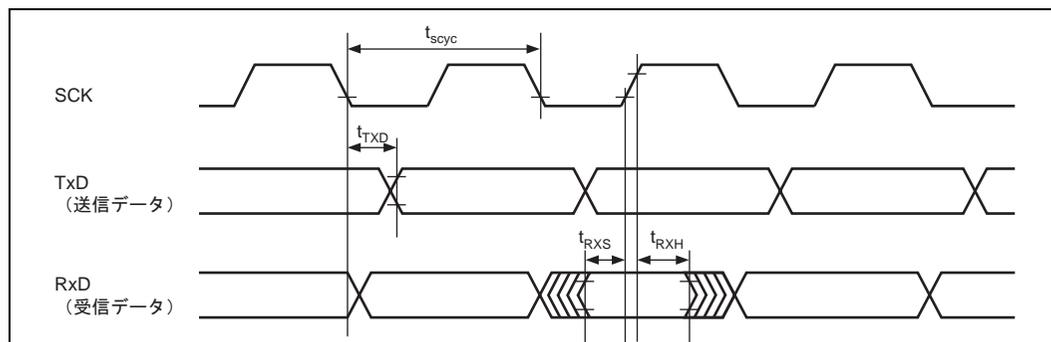


図 21.54 SCI 入出力タイミング (クロック同期式モード)

21. 電気的特性

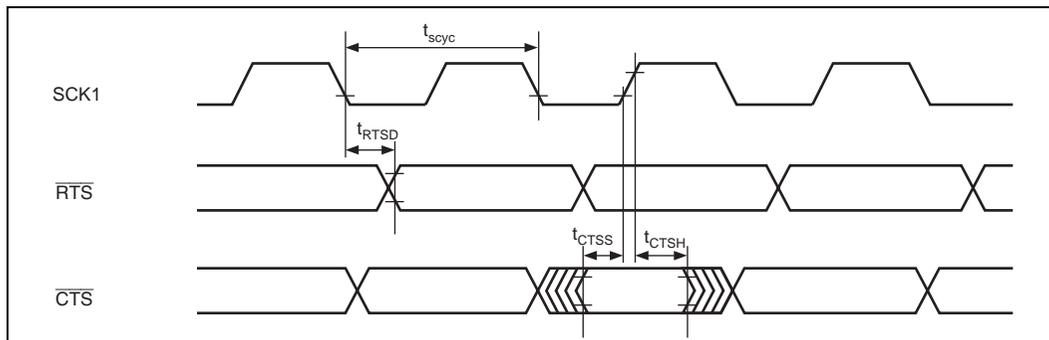


図 21.55 RTS、CTS 入出力タイミング

表 21.11 16 ビットタイマパルスユニット

条件 :  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、

$V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
タイマ出力遅延時間	$t_{TOCD}$		100	ns	21.56、57
タイマ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ )	$t_{TICS}$	50		ns	
タイマ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 2$ )	$t_{TICS}$	$t_{cyc}+50$		ns	21.56、57
タイマ入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 4$ )	$t_{TICS}$	$3t_{cyc}+50$		ns	
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ )	$t_{TCKS}$	50		ns	21.58
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 2$ )	$t_{TCKS}$	$t_{cyc}+50$		ns	
タイマクロック入力セットアップ時間 ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 4$ )	$t_{TCKS}$	$3t_{cyc}+50$		ns	
タイマクロックパルス幅	単エッジ指定	$t_{TCKWH}$	1.5	tcyc	
	両エッジ指定	$t_{TCKWL}$	2.5		

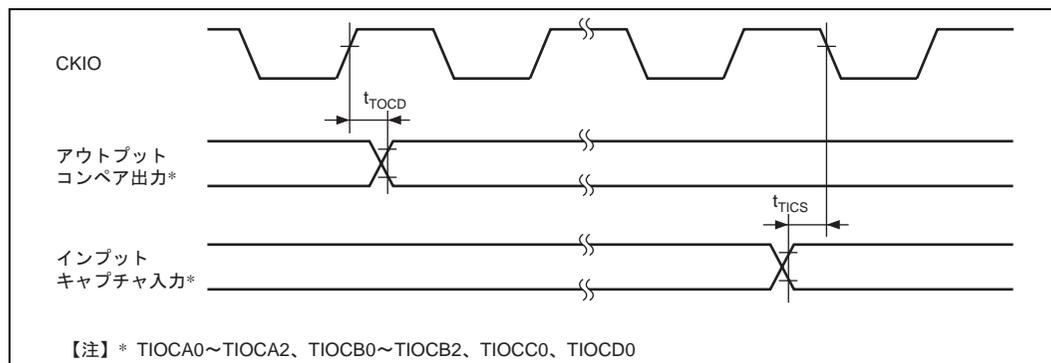
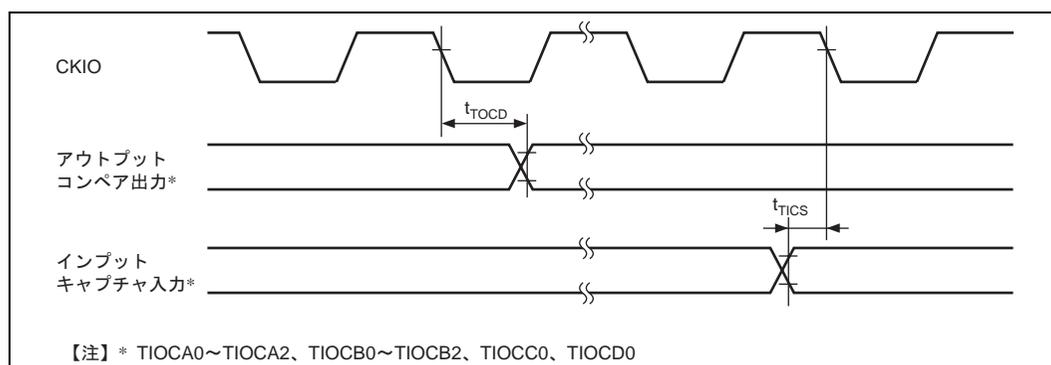
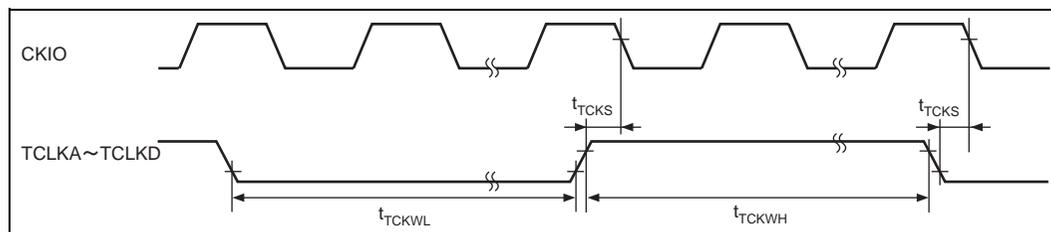
図 21.56 TPU 入出力タイミング ( $t_{Ecyc} : t_{Pcyc} = 1 : 1$ )図 21.57 TPU 入出力タイミング ( $t_{Ecyc} : t_{Pcyc} = 1 : 1$  以外)

図 21.58 TPU クロック入力タイミング

21.3.7 ウォッチドッグタイマタイミング

表 21.12 ウォッチドッグタイマタイミング

条件 :  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
WDTOVF 遅延時間	$t_{WDOVF}$		70	ns	21.59、60

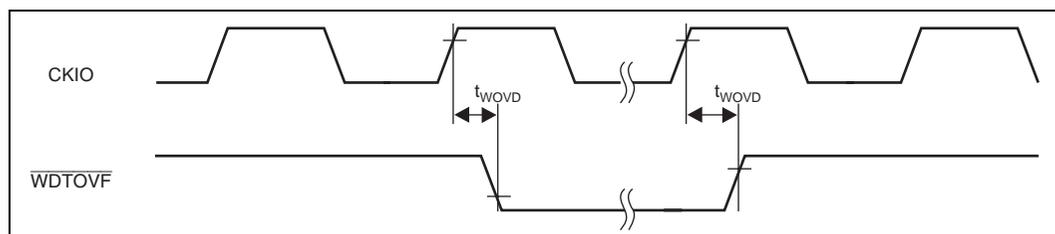


図 21.59 ウォッチドッグタイマ出力タイミング ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$ )

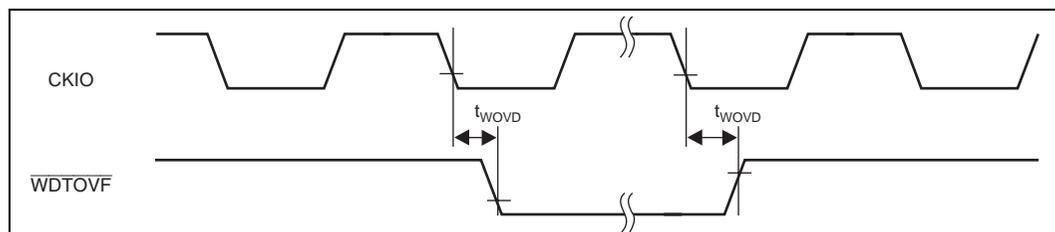


図 21.60 ウォッチドッグタイマ出力タイミング ( $t_{E_{cyc}} : t_{P_{cyc}} = 1 : 1$  以外)

## 21.3.8 シリアル I/O タイミング

表 21.13 シリアル I/O タイミング

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
SRCK、STCK クロック入力サイクル時間	$t_{Slcyc}$	$t_{Pcyc}$ or *66.7		ns	21.61
SRCK、STCK クロック入力ローレベル幅	$t_{WL}$	$0.4 \times t_{Slcyc}$		ns	
SRCK、STCK クロック入力ハイレベル幅	$t_{WH}$	$0.4 \times t_{Slcyc}$		ns	
SRS 入力セットアップ時間	$t_{RSS}$	15		ns	21.62
SRS 入力ホールド時間	$t_{RSH}$	10		ns	
SRXD 入力セットアップ時間	$t_{SRDS}$	15		ns	
SRXD 入力ホールド時間	$t_{SRDH}$	10		ns	21.63
STS 入力セットアップ時間	$t_{TSS}$	15		ns	
STS 入力ホールド時間	$t_{TSH}$	10		ns	
STS 出力遅延時間	$t_{TSD}$	0	20	ns	21.64
STXD 出力遅延時間	$t_{TDD}$	0	20	ns	21.63、64

【注】 \*  $t_{Pcyc}$  と 66.7 の大きな方で規定

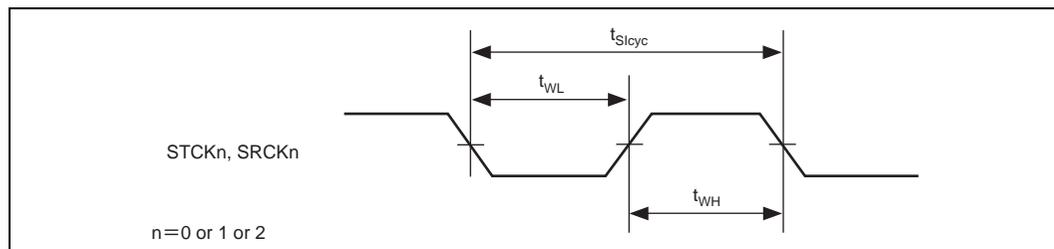


図 21.61 SIO 入力クロックタイミング

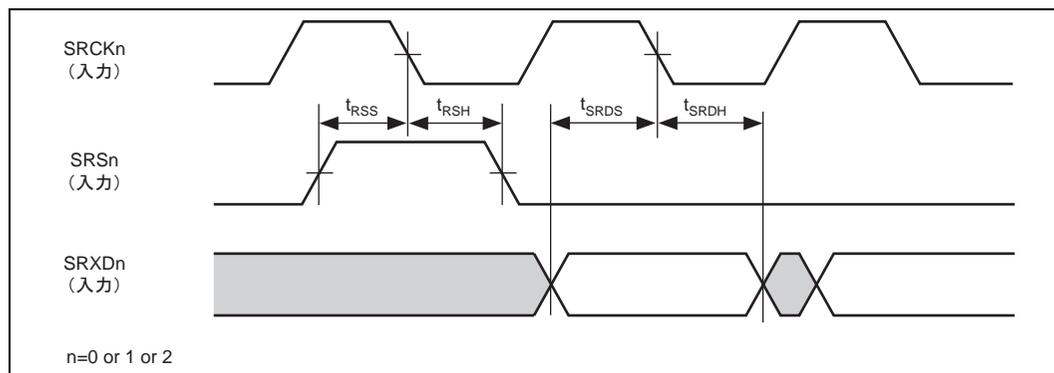


図 21.62 SIO 受信タイミング

21. 電気的特性

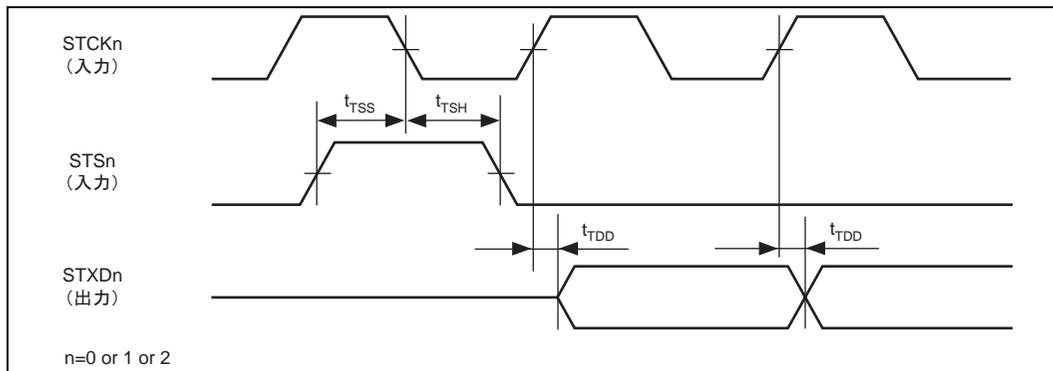


図 21.63 SIO 送信タイミング (TMn = 0 モード)

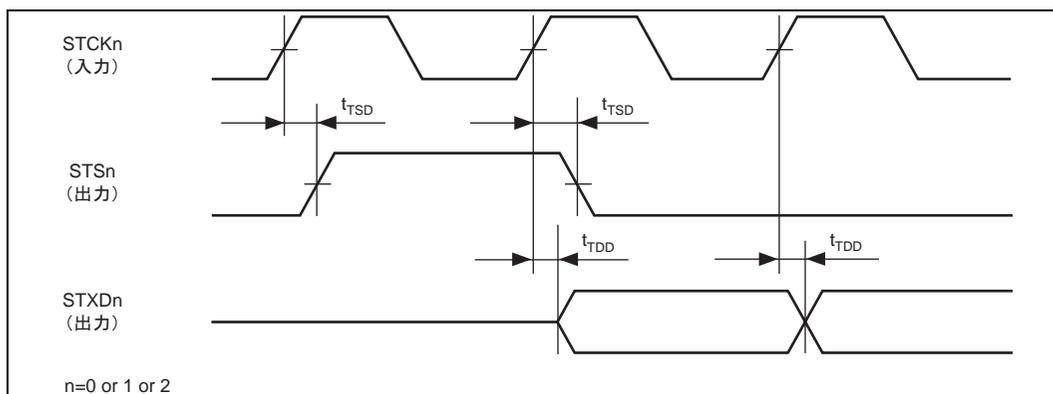


図 21.64 SIO 送信タイミング (TMn = 1 モード)

## 21.3.9 ハイパフォーマンスユーザデバッグインタフェースタイミング

表 21.14 ハイパフォーマンスユーザデバックインタフェースタイミング

条件:  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} > V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
TCK クロック入力サイクル時間	$t_{tcyc}$	$t_{P_{cyc}}$ or*66.7ns		ns	21.65
TCK クロック入力ハイレベル幅	$t_{TCKH}$	0.4	0.6	$t_{tcyc}$	
TCK クロック入力ローレベル幅	$t_{TCKL}$	0.4	0.6	$t_{tcyc}$	
TRST パルス幅	$t_{TRSW}$	20		$t_{tcyc}$	21.66
TRST セットアップ時間	$t_{TRSS}$	40		ns	
TMS セットアップ時間	$t_{TMSS}$	30		ns	21.67
TMS ホールド時間	$t_{TMSh}$	10		ns	
TDI セットアップ時間	$t_{TDis}$	30		ns	
TDI ホールド時間	$t_{TDIH}$	10		ns	
TDO 遅延時間	$t_{TDOD}$	0	30	ns	

【注】 \*  $t_{P_{cyc}}$  と 66.7 の大きな方で規定

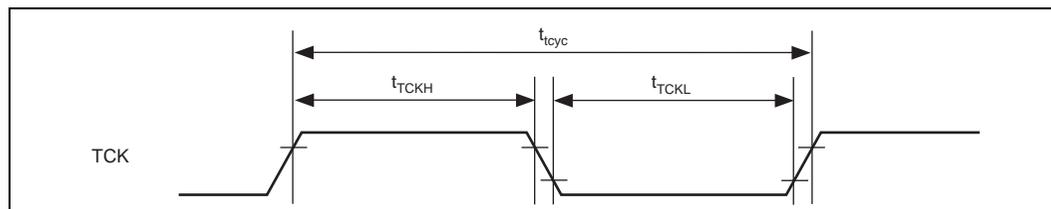


図 21.65 H-UDI クロックタイミング

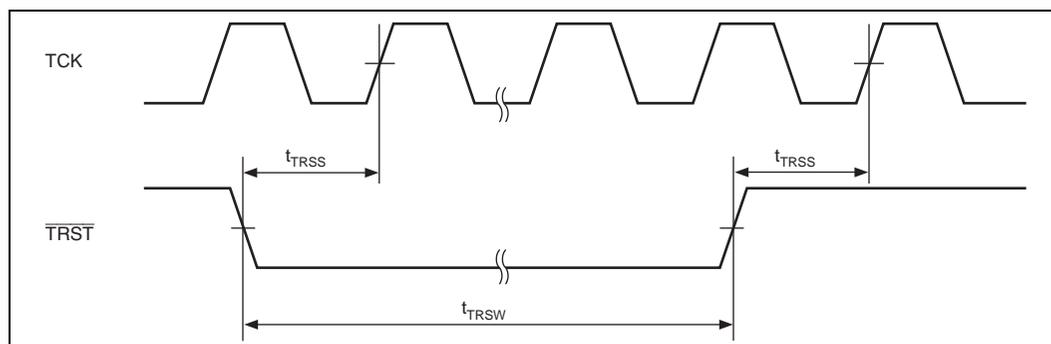


図 21.66 H-UDI TRST タイミング

## 21. 電気的特性

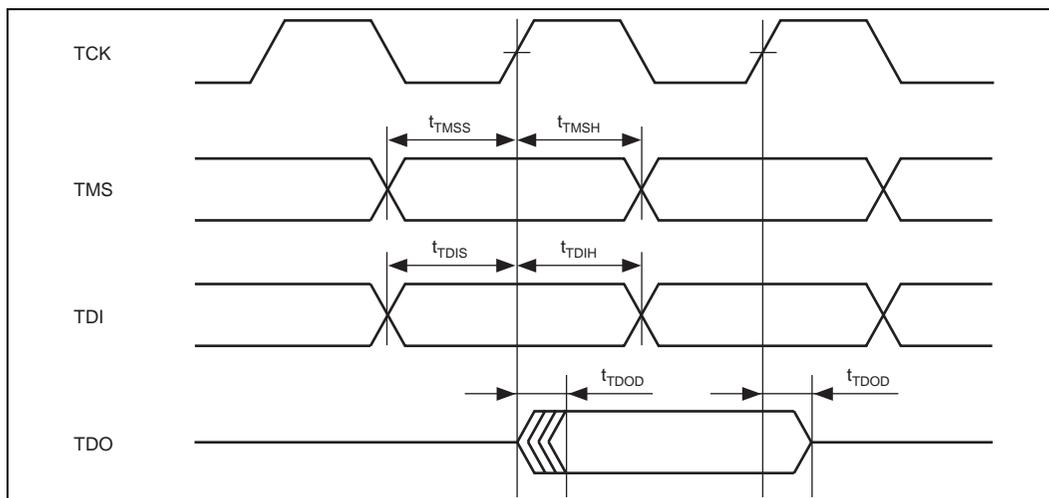


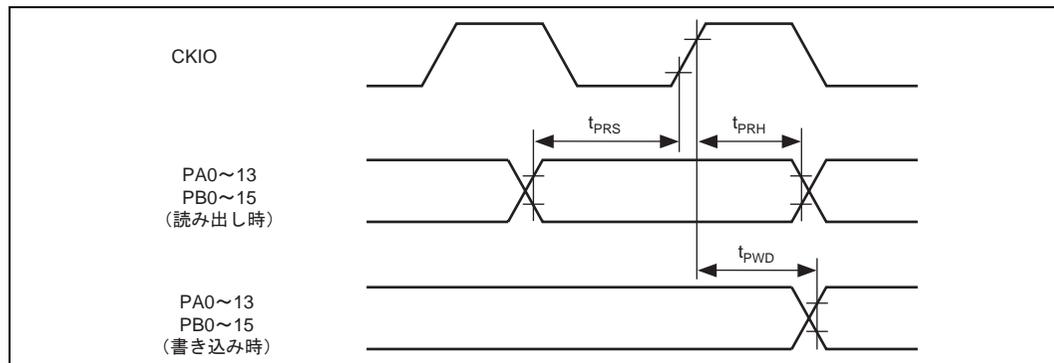
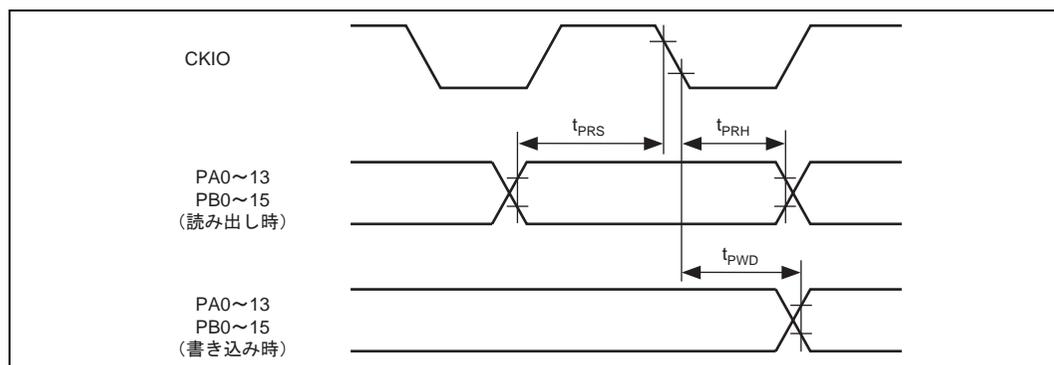
図 21.67 H-UDI 入出力タイミング

### 21.3.10 I/O ポートタイミング

表 21.15 I/O ポートタイミング

条件 :  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	max	単位	参照図
ポート出力データ遅延時間	$t_{PWD}$		50	ns	21.68、69
ポート入力データセットアップ時間 ( $t_{E_{Cyc}} : t_{P_{Cyc}} = 1 : 1$ )	$t_{PRS}$	50		ns	21.68
ポート入力データセットアップ時間 ( $t_{E_{Cyc}} : t_{P_{Cyc}} = 1 : 2$ )	$t_{PRS}$	$t_{cyc}+50$		ns	21.69
ポート入力データセットアップ時間 ( $t_{E_{Cyc}} : t_{P_{Cyc}} = 1 : 4$ )	$t_{PRS}$	$3t_{cyc}+50$		ns	
ポート入力データホールド時間	$t_{PRH}$	50		ns	21.68、69

図 21.68 I/O ポート入出力タイミング ( $t_{Ecyc} : t_{Pcyc} = 1 : 1$ )図 21.69 I/O ポート入出力タイミング ( $t_{Ecyc} : t_{Pcyc} = 1 : 1$  以外)

## 21. 電気的特性

### 21.3.11 イーサネットコントローラタイミング

表 21.16 イーサネットコントローラタイミング

条件：  $V_{CC}=PLL_{V_{CC}} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS} = PLL_{V_{SS}} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	typ	max	単位	参照図
TX-CLK サイクル時間	$t_{Tcyc}$	40			ns	
TX-EN 出力遅延時間	$t_{TENd}$	3		20	ns	21.70
ETXD[3:0] 出力遅延時間	$t_{ETDd}$	3		20	ns	
CRS セットアップ時間	$t_{CRSs}$	10			ns	
CRS ホールド時間	$t_{CRSh}$	10			ns	
COL セットアップ時間	$t_{COLs}$	10			ns	
COL ホールド時間	$t_{COLh}$	10			ns	
RX-CLK サイクル時間	$t_{Rcyc}$	40			ns	21.72
RX-DV セットアップ時間	$t_{RDVs}$	10			ns	
RX-DV ホールド時間	$t_{RDVh}$	3			ns	
ERXD[3:0] セットアップ時間	$t_{ERDs}$	10			ns	
ERXD[3:0] ホールド時間	$t_{ERDh}$	3			ns	
RX-ER セットアップ時間	$t_{RERs}$	10			ns	21.73
RX-ER ホールド時間	$t_{RERh}$	3			ns	
MDIO セットアップ時間	$t_{MDIOs}$	10			ns	21.74
MDIO ホールド時間	$t_{MDIOh}$	10			ns	
MDIO 出力データホールド時間 *	$t_{MDIOdh}$	5		18	ns	21.75
WOL 出力遅延時間	$t_{WOLd}$	1		14	ns	21.76
EXOUT 出力遅延時間	$t_{EXOUTd}$	1		25	ns	21.77

【注】 \* ユーザがプログラムにより本規定を満足するように設定する必要があります。

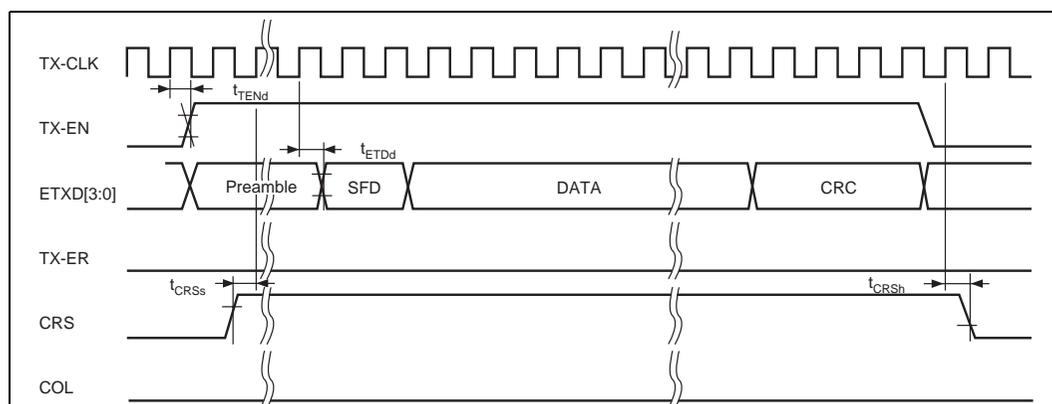


図 21.70 MII 送信タイミング (正常動作時)

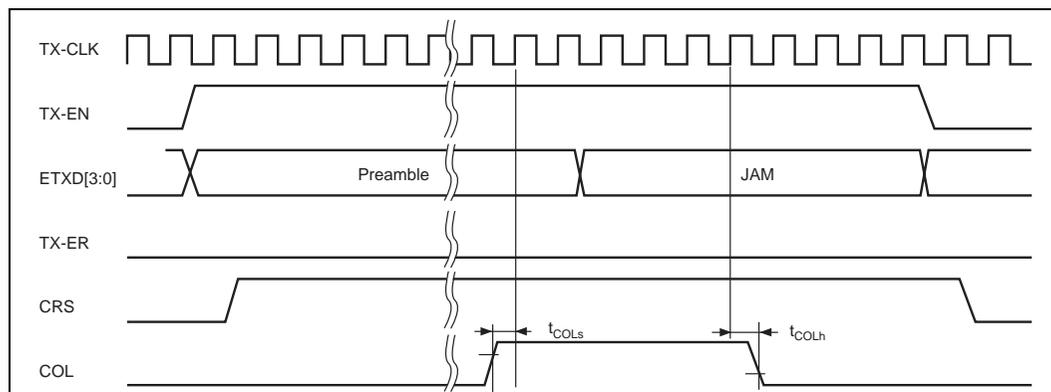


図 21.71 MII 送信タイミング (衝突発生ケース)

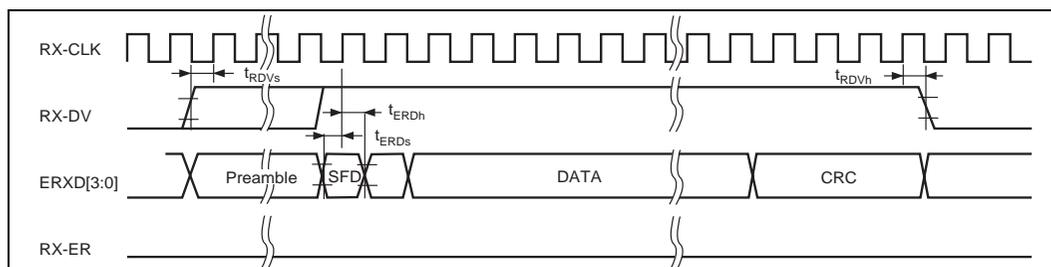


図 21.72 MII 受信タイミング (正常動作時)

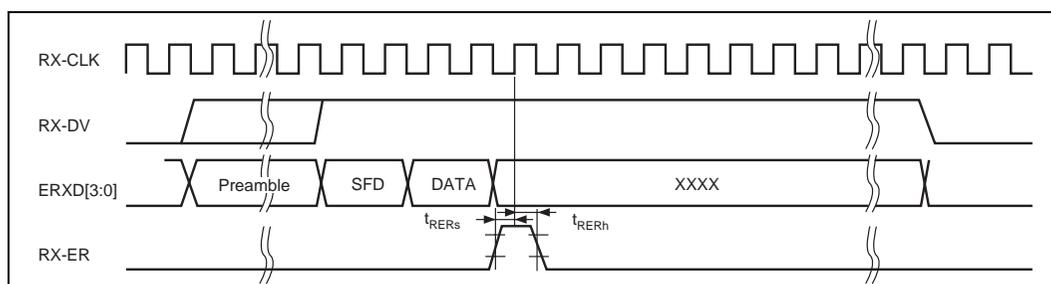


図 21.73 MII 受信タイミング (エラー発生ケース)

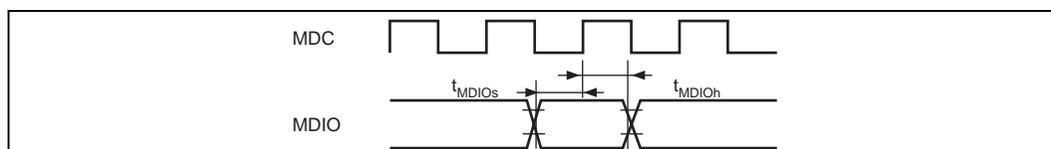


図 21.74 MDIO 入力タイミング

## 21. 電気的特性

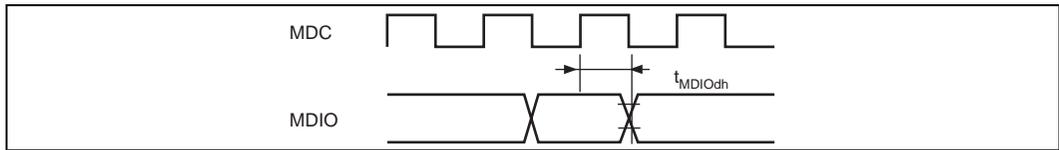


図 21.75 MDIO 出力タイミング

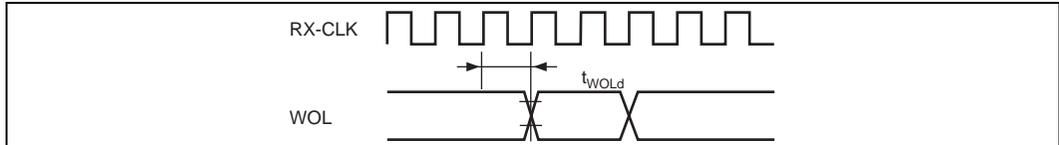


図 21.76 WOL 出力タイミング

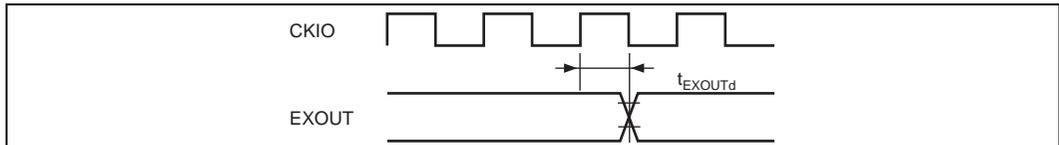


図 21.77 EXOUT 出力タイミング

## 21.3.12 STATS、BH、BUSHIZ 信号タイミング

表 21.17 STATS、BH、BUSHIZ 信号タイミング

条件：  $V_{CC}=PLL V_{CC} = 3.3V \pm 0.3V$ 、 $PV_{CC} = 5.0V \pm 0.5V/3.3V \pm 0.3V$ 、 $PV_{CC} = V_{CC}$ 、  
 $V_{SS} = PV_{SS}=PLL V_{SS} = 0V$ 、 $T_a = -20 \sim +75$

項目	記号	min	typ	max	単位	参照図
STATS1、0 出力遅延時間	$t_{STATd}$			16	ns	21.78
BH 出力立ち上がり遅延時間	$t_{BHNrd}$			16	ns	21.79
BH 出力立ち下がり遅延時間	$t_{BHNfd}$			16	ns	
BUSHIZ セットアップ時間	$t_{BHIZs}$	7			ns	21.80
BUSHIZ ホールド時間	$t_{BHIZh}$	8			ns	
Target Pins 出力遅延時間	$t_{BHIZd}$			16	ns	

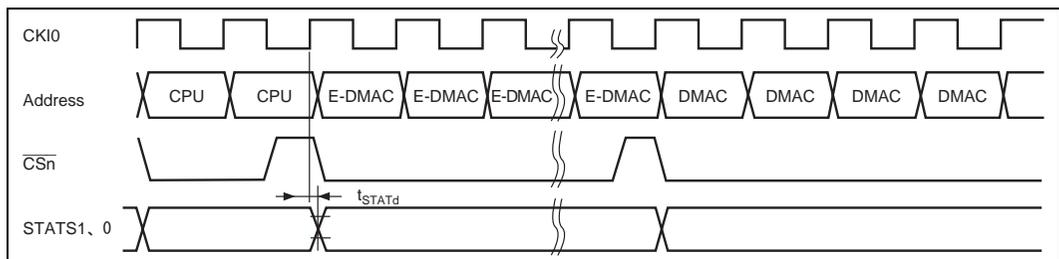


図 21.78 STATS 出力タイミング

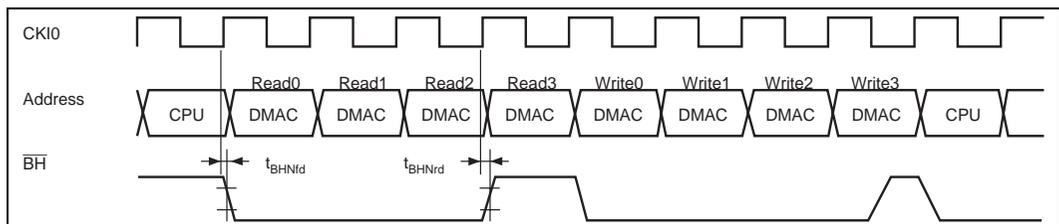


図 21.79 BH 出力タイミング

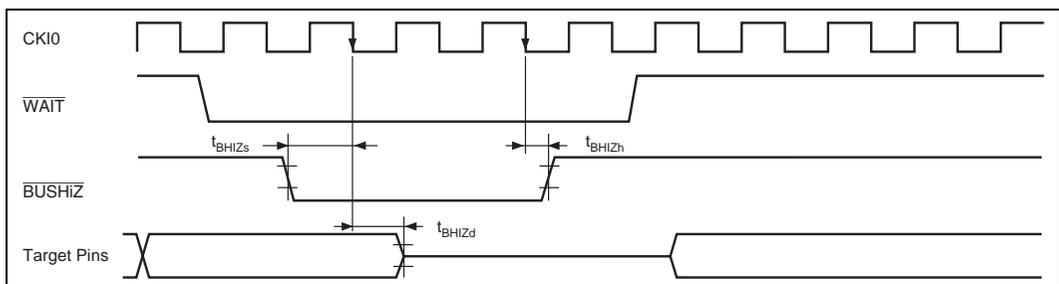


図 21.80 BUSHIZ バスタイミング

## 21.4 AC 特性測定条件

AC 特性測定条件は次のとおりです。

入出力信号参照レベル：1.5V ( $V_{CC} = 3.0 \sim 3.6V$ )

入力パルスレベル： $V_{SS} \sim 3.0V$  (ただし、 $\overline{RES}$ 、 $\overline{TRST}$ 、 $\overline{EXTAL}$ 、 $CKIO$ 、 $MD0 \sim 4$ 、 $NMI$  は  $V_{SS} \sim V_{CC}$ )

入力上昇、下降時間：1ns

出力付加回路を図 21.81 に示します。

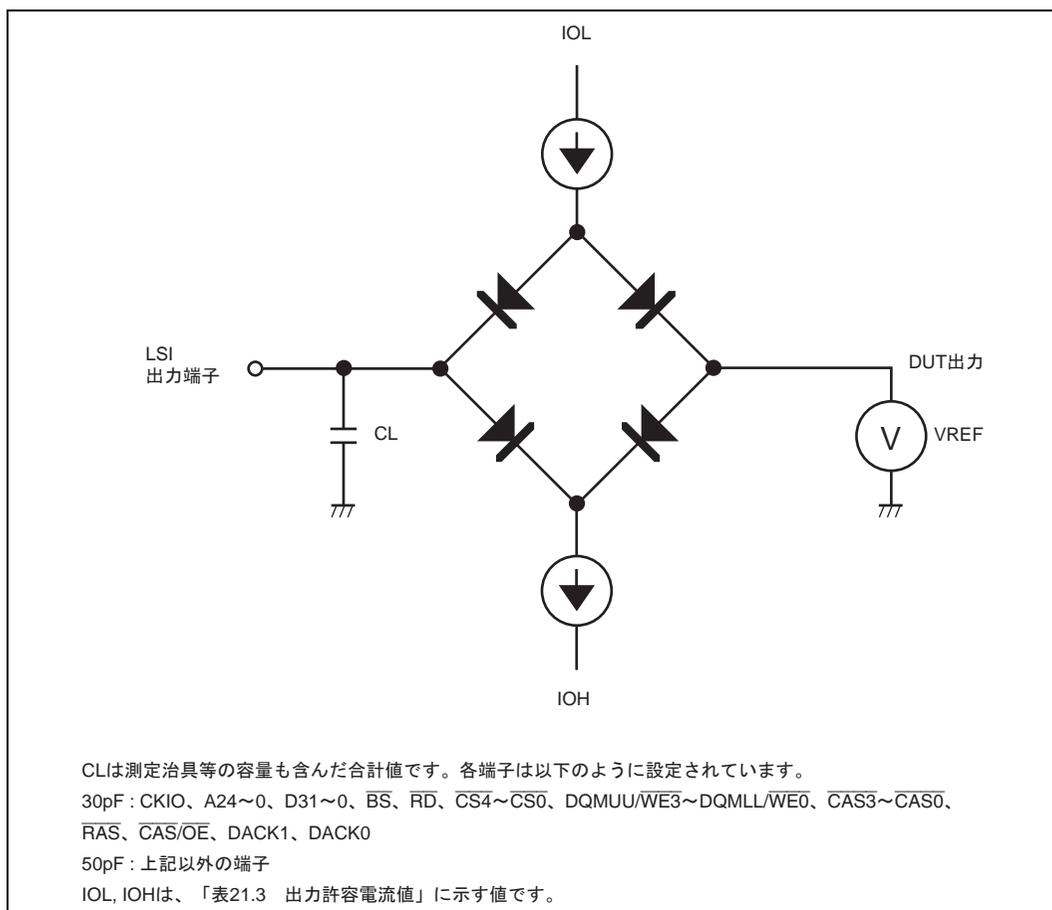


図 21.81 出力付加回路

# 付録

## A. 内蔵周辺モジュールレジスタ

### A.1 アドレス一覧

内蔵周辺モジュールレジスタのアドレスとビット名を示します。16ビットレジスタ、32ビットレジスタはそれぞれ8ビットずつ2段、4段で表しています。

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FC00	SIRDR0									SIO0
H'FFFF FC01										
H'FFFF FC02	SITDR0									
H'FFFF FC03										
H'FFFF FC04	SICTR0	-	-	-	-	-	-	-	-	
H'FFFF FC05		-	TM	SE	DL	TIE	RIE	TE	RE	
H'FFFF FC06	SISTR0	-	-	-	-	-	-	-	-	
H'FFFF FC07		-	-	-	-	TERR	RERR	TDRE	RDRF	
H'FFFF FC08 ~ H'FFFF FC0F	-	-	-	-	-	-	-	-	-	-
H'FFFF FC10	SIRDR1									SIO1
H'FFFF FC11										
H'FFFF FC12	SITDR1									
H'FFFF FC13										
H'FFFF FC14	SICTR1	-	-	-	-	-	-	-	-	
H'FFFF FC15		-	TM	SE	DL	TIE	RIE	TE	RE	
H'FFFF FC16	SISTR1	-	-	-	-	-	-	-	-	
H'FFFF FC17		-	-	-	-	TERR	RERR	TDRE	RDRF	
H'FFFF FC18 ~ H'FFFF FC1F	-	-	-	-	-	-	-	-	-	-
H'FFFF FC20	SIRDR2									SIO2
H'FFFF FC21										
H'FFFF FC22	SITDR2									
H'FFFF FC23										
H'FFFF FC24	SICTR2	-	-	-	-	-	-	-	-	
H'FFFF FC25		-	TM	SE	DL	TIE	RIE	TE	RE	
H'FFFF FC26	SISTR2	-	-	-	-	-	-	-	-	
H'FFFF FC27		-	-	-	-	TERR	RERR	TDRE	RDRF	
H'FFFF FC28 ~ H'FFFF FC3F	-	-	-	-	-	-	-	-	-	-

付 録

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FC40	TSTR	-	-	-	-	-	CST2	CST1	CST0	TPU
H'FFFF FC41	TSYR	-	-	-	-	-	SYNC2	SYNC1	SYNC0	
H'FFFF FC42 ~ H'FFFF FC4F	-	-	-	-	-	-	-	-	-	-
H'FFFF FC50	TCR0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	TPU
H'FFFF FC51	TMDR0	-	-	BFB	BFA	MD3	MD2	MD1	MD0	
H'FFFF FC52	TIOR0H	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
H'FFFF FC53	TIOR0L	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0	
H'FFFF FC54	TIER0	-	-	-	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
H'FFFF FC55	TSR0	-	-	-	TCFV	TGFD	TGFC	TGFB	TGFA	
H'FFFF FC56	TCNT0									
H'FFFF FC57										
H'FFFF FC58	TGR0A									
H'FFFF FC59										
H'FFFF FC5A	TGR0B									
H'FFFF FC5B										
H'FFFF FC5C	TGR0C									
H'FFFF FC5D										
H'FFFF FC5E	TGR0D									
H'FFFF FC5F										
H'FFFF FC60	TCR1	-	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	
H'FFFF FC61	TMDR1	-	-	-	-	MD3	MD2	MD1	MD0	
H'FFFF FC62	TIOR1	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
H'FFFF FC63	-	-	-	-	-	-	-	-	-	
H'FFFF FC64	TIER1	-	-	TCIEU	TCIEV	-	-	TGIEB	TGIEA	
H'FFFF FC65	TSR1	TCFD	-	TCFU	TCFV	-	-	TGFB	TGFA	
H'FFFF FC66	TCNT1									
H'FFFF FC67										
H'FFFF FC68	TGR1A									
H'FFFF FC69										
H'FFFF FC6A	TGR1B									
H'FFFF FC6B										
H'FFFF FC6C ~ H'FFFF FC6F	-	-	-	-	-	-	-	-	-	-
H'FFFF FC70	TCR2	-	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	TPU
H'FFFF FC71	TMDR2	-	-	-	-	MD3	MD2	MD1	MD0	
H'FFFF FC72	TIOR2	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
H'FFFF FC73	-	-	-	-	-	-	-	-	-	
H'FFFF FC74	TIER2	-	-	TCIEU	TCIEV	-	-	TGIEB	TGIEA	
H'FFFF FC75	TSR2	TCFD	-	TCFU	TCFV	-	-	TGFB	TGFA	
H'FFFF FC76	TCNT2									
H'FFFF FC77										
H'FFFF FC78	TGR2A									
H'FFFF FC79										
H'FFFF FC7A	TGR2B									
H'FFFF FC7B										

アドレス	レジスタ 略称	ビット名								モジュール	
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'FFFF FC7C ~ H'FFFF FC7F	-	-	-	-	-	-	-	-	-	-	-
H'FFFF FC80	PACR	-	-	PA13MD	PA12MD	PA11MD	PA10MD	PA9MD	PA8MD	PFC	
H'FFFF FC81		PA7MD	PA6MD	PA5MD	PA4MD	PA3MD	PA2MD	PA1MD	PA0MD		
H'FFFF FC82	PAIOR	-	-	PA13IOR	PA12IOR	PA11IOR	PA10IOR	PA9IOR	PA8IOR		
H'FFFF FC83		PA7IOR	PA6IOR	PA5IOR	PA4IOR	-	PA2IOR	PA1IOR	PA0IOR		
H'FFFF FC84	PADR	-	-	PA13DR	PA12DR	PA11DR	PA10DR	PA9DR	PA8DR	I/O port	
H'FFFF FC85		PA7DR	PA6DR	PA5DR	PA4DR	-	PA2DR	PA1DR	PA0DR		
H'FFFF FC86	-	-	-	-	-	-	-	-	-	-	
H'FFFF FC87	-	-	-	-	-	-	-	-	-	-	
H'FFFF FC88	PBCR	PB15MD1	PB15MD0	PB14MD1	PB14MD0	PB13MD1	PB13MD0	PB12MD1	PB12MD0	PFC	
H'FFFF FC89		PB11MD1	PB11MD0	PB10MD1	PB10MD0	PB9MD1	PB9MD0	PB8MD1	PB8MD0		
H'FFFF FC8A	PBIOR	PB15IOR	PB14IOR	PB13IOR	PB12IOR	PB11IOR	PB10IOR	PB9IOR	PB8IOR		
H'FFFF FC8B		PB7IOR	PB6IOR	PB5IOR	PB4IOR	PB3IOR	PB2IOR	PB1IOR	PB0IOR		
H'FFFF FC8C	PBDR	PB15DR	PB14DR	PB13DR	PB12DR	PB11DR	PB10DR	PB9DR	PB8DR	I/O port	
H'FFFF FC8D		PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	PB0DR		
H'FFFF FC8E	PBCR2	PB7MD1	PB7MD0	PB6MD1	PB6MD0	PB5MD1	PB5MD0	PB4MD1	PB4MD0	PFC	
H'FFFF FC8F		PB3MD1	PB3MD0	PB2MD1	PB2MD0	PB1MD1	PB1MD0	PB0MD1	PB0MD0		
H'FFFF FC90 ~ H'FFFF FCAF	-	-	-	-	-	-	-	-	-	-	
H'FFFF FCB0	SDIR	TS3	TS2	TS1	TS0	-	-	-	-	H-UDI	
H'FFFF FCB1		-	-	-	-	-	-	-	-		
H'FFFF FCB2	SDSR	-	-	-	-	-	-	-	-		
H'FFFF FCB3		-	-	-	-	-	-	-	SDTRF		
H'FFFF FCB4	SDDRH										
H'FFFF FCB5											
H'FFFF FCB6	SDDRL										
H'FFFF FCB7											
H'FFFF FCB8 ~ H'FFFF FCBF	-	-	-	-	-	-	-	-	-	-	
H'FFFF FCC0	SCSMR1	C/Ā	CHR/ICK3	PE/ICK2	OĒ/ICK1	STOP/ ICK0	MP	CKS1	CKS0	SCIF1	
H'FFFF FCC1	-	-	-	-	-	-	-	-	-		
H'FFFF FCC2	SCBRR1										
H'FFFF FCC3	-	-	-	-	-	-	-	-	-		
H'FFFF FCC4	SCSCR1	TIE	RIE	TE	RE	MPIE	-	CKE1	CKE0		
H'FFFF FCC5	-	-	-	-	-	-	-	-	-		
H'FFFF FCC6	SCFTDR1										
H'FFFF FCC7	-	-	-	-	-	-	-	-	-		
H'FFFF FCC8	SC1SSR1	PER3	PER2	PER1	PER0	FER3	FER2	FER1	FER0		
H'FFFF FCC9		ER	TEND	TDFE	BRK	FER	PER	RDF	DR		
H'FFFF FCCA	SC2SSR1	TLM	RLM	N1	N0	MPB	MPBT	EI	ORER		
H'FFFF FCCB	-	-	-	-	-	-	-	-	-		
H'FFFF FCCC	SCFRDR1										
H'FFFF FCCD	-	-	-	-	-	-	-	-	-		

付 録

アドレス	レジスタ 略称	ビット名								モジュール	
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'FFFF FCCE	SCFCR1	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP	SCIF1	
H'FFFF FCCF	-	-	-	-	-	-	-	-	-		
H'FFFF FCD0	SCFDR1	-	-	-	T4	T3	T2	T1	T0		
H'FFFF FCD1		-	-	-	R4	R3	R2	R1	R0		
H'FFFF FCD2	SCFER1	ED15	ED14	ED13	ED12	ED11	ED10	ED9	ED8		
H'FFFF FCD3		ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0		
H'FFFF FCD4	SCIMR1	IRMOD	PSEL	RIVS	-	-	-	-	-		
H'FFFF FCD5 ~ H'FFFF FCDF	-	-	-	-	-	-	-	-	-	-	
H'FFFF FCE0	SCSMR2	C/A	CHR/ICK3	PE/ICK2	O/E/ICK1	STOP/ ICK0	MP	CKS1	CKS0	SCIF2	
H'FFFF FCE1	-	-	-	-	-	-	-	-	-		
H'FFFF FCE2	SCBRR2										
H'FFFF FCE3	-	-	-	-	-	-	-	-	-		
H'FFFF FCE4	SCSCR2	TIE	RIE	TE	RE	MPIE	-	CKE1	CKE0		
H'FFFF FCE5	-	-	-	-	-	-	-	-	-		
H'FFFF FCE6	SCFTDR2										
H'FFFF FCE7	-	-	-	-	-	-	-	-	-		
H'FFFF FCE8	SC1SSR2	PER3	PER2	PER1	PER0	FER3	FER2	FER1	FER0		
H'FFFF FCE9		ER	TEND	TDFE	BRK	FER	PER	RDF	DR		
H'FFFF FCEA	SC2SSR2	TLM	RLM	N1	N0	MPB	MPBT	EI	ORER		
H'FFFF FCEB	-	-	-	-	-	-	-	-	-		
H'FFFF FCEC	SCFRDR2										
H'FFFF FCED	-	-	-	-	-	-	-	-	-		
H'FFFF FCEE	SCFCR2	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP		
H'FFFF FCEF	-	-	-	-	-	-	-	-	-		
H'FFFF FCF0	SCFDR2	-	-	-	T4	T3	T2	T1	T0		
H'FFFF FCF1		-	-	-	R4	R3	R2	R1	R0		
H'FFFF FCF2	SCFER2	ED15	ED14	ED13	ED12	ED11	ED10	ED9	ED8		
H'FFFF FCF3		ED7	ED6	ED5	ED4	ED3	ED2	ED1	ED0		
H'FFFF FCF4	SCIMR2	IRMOD	PSEL	RIVS	-	-	-	-	-		
H'FFFF FCF5 ~ H'FFFF FCFF	-	-	-	-	-	-	-	-	-		-
H'FFFF FD00	EDMR	-	-	-	-	-	-	-	-		E-DMAC
H'FFFF FD01		-	-	-	-	-	-	-	-		
H'FFFF FD02		-	-	-	-	-	-	-	-		
H'FFFF FD03		-	-	DL1	DL0	-	-	-	SWR		
H'FFFF FD04	EDTRR	-	-	-	-	-	-	-	-		
H'FFFF FD05		-	-	-	-	-	-	-	-		
H'FFFF FD06		-	-	-	-	-	-	-	-		
H'FFFF FD07		-	-	-	-	-	-	-	TR		
H'FFFF FD08	EDRRR	-	-	-	-	-	-	-	-		
H'FFFF FD09		-	-	-	-	-	-	-	-		
H'FFFF FD0A		-	-	-	-	-	-	-	-		
H'FFFF FD0B		-	-	-	-	-	-	-	RR		

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FD0C	TDLAR	TDLA31	TDLA30	TDLA29	TDLA28	TDLA27	TDLA26	TDLA25	TDLA24	E-DMAC
H'FFFF FD0D		TDLA23	TDLA22	TDLA21	TDLA20	TDLA19	TDLA18	TDLA17	TDLA16	
H'FFFF FD0E		TDLA15	TDLA14	TDLA13	TDLA12	TDLA11	TDLA10	TDLA9	TDLA8	
H'FFFF FD0F		TDLA7	TDLA6	TDLA5	TDLA4	TDLA3	TDLA2	TDLA1	TDLA0	
H'FFFF FD10	RDLAR	RDLA31	RDLA30	RDLA29	RDLA28	RDLA27	RDLA26	RDLA25	RDLA24	
H'FFFF FD11		RDLA23	RDLA22	RDLA21	RDLA20	RDLA19	RDLA18	RDLA17	RDLA16	
H'FFFF FD12		RDLA15	RDLA14	RDLA13	RDLA12	RDLA11	RDLA10	RDLA9	RDLA8	
H'FFFF FD13		RDLA7	RDLA6	RDLA5	RDLA4	RDLA3	RDLA2	RDLA1	RDLA0	
H'FFFF FD14	EESR	-	-	-	-	-	TABT	RABT	RFCOF	
H'FFFF FD15		-	ECI	TC	TDE	TFUF	FR	RDE	RFOF	
H'FFFF FD16		-	-	-	ITF	CND	DLC	CD	TRO	
H'FFFF FD17		RMAF	-	-	RRF	RTLF	RTSF	PRE	CERF	
H'FFFF FD18	EESIPR	-	-	-	-	-	-	-	RFCOFIP	
H'FFFF FD19		-	ECIIP	TCIP	TDEIP	TFUFIP	FRIP	RDEIP	RFOFIP	
H'FFFF FD1A		-	-	-	ITFIP	CNDIP	DLCIP	CDIP	TROIIP	
H'FFFF FD1B		RMAFIP	-	-	RRFIP	RTLFIIP	RTSFIIP	PREIP	CERFIIP	
H'FFFF FD1C	TRSCER	-	-	-	-	-	-	-	-	
H'FFFF FD1D		-	-	-	-	-	-	-	-	
H'FFFF FD1E		-	-	-	-	-	-	-	-	
H'FFFF FD1F		RMAFCE	-	-	-	-	-	-	-	
H'FFFF FD20	RMFCR	-	-	-	-	-	-	-	-	
H'FFFF FD21		-	-	-	-	-	-	-	-	
H'FFFF FD22		MFC15	MFC14	MFC13	MFC12	MFC11	MFC10	MFC9	MFC8	
H'FFFF FD23		MFC7	MFC6	MFC5	MFC4	MFC3	MFC2	MFC1	MFC0	
H'FFFF FD24	TFTR	-	-	-	-	-	-	-	-	
H'FFFF FD25		-	-	-	-	-	-	-	-	
H'FFFF FD26		-	-	-	-	-	TFT10	TFT9	TFT8	
H'FFFF FD27		TFT7	TFT6	TFT5	TFT4	TFT3	TFT2	TFT1	TFT0	
H'FFFF FD28	FDR	-	-	-	-	-	-	-	-	
H'FFFF FD29		-	-	-	-	-	-	-	-	
H'FFFF FD2A		-	-	-	-	-	-	-	TFD	
H'FFFF FD2B		-	-	-	-	-	-	-	RFD	
H'FFFF FD2C	RCR	-	-	-	-	-	-	-	-	
H'FFFF FD2D		-	-	-	-	-	-	-	-	
H'FFFF FD2E		-	-	-	-	-	-	-	-	
H'FFFF FD2F		-	-	-	-	-	-	-	RNC	
H'FFFF FD30	EDOCR	-	-	-	-	-	-	-	-	
H'FFFF FD31		-	-	-	-	-	-	-	-	
H'FFFF FD32		-	-	-	-	-	-	-	-	
H'FFFF FD33		-	-	-	-	FEC	AEC	EDH	-	
H'FFFF FD34 ~ H'FFFF FD3F	-	-	-	-	-	-	-	-	-	E-DMAC
H'FFFF FD40	RBWAR	RBWA31	RBWA30	RBWA29	RBWA28	RBWA27	RBWA26	RBWA25	RBWA24	
H'FFFF FD41		RBWA23	RBWA22	RBWA21	RBWA20	RBWA19	RBWA18	RBWA17	RBWA16	
H'FFFF FD42		RBWA15	RBWA14	RBWA13	RBWA12	RBWA11	RBWA10	RBWA9	RBWA8	
H'FFFF FD43		RBWA7	RBWA6	RBWA5	RBWA4	RBWA3	RBWA2	RBWA1	RBWA0	

付 録

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FD44	RDFAR	RDFA31	RDFA30	RDFA29	RDFA28	RDFA27	RDFA26	RDFA25	RDFA24	E-DMAC
H'FFFF FD45		RDFA23	RDFA22	RDFA21	RDFA20	RDFA19	RDFA18	RDFA17	RDFA16	
H'FFFF FD46		RDFA15	RDFA14	RDFA13	RDFA12	RDFA11	RDFA10	RDFA9	RDFA8	
H'FFFF FD47		RDFA7	RDFA6	RDFA5	RDFA4	RDFA3	RDFA2	RDFA1	RDFA0	
H'FFFF FD48 ~ H'FFFF FD4B	-	-	-	-	-	-	-	-	-	
H'FFFF FD4C	TBRAR	TBRA31	TBRA30	TBRA29	TBRA28	TBRA27	TBRA26	TBRA25	TBRA24	
H'FFFF FD4D		TBRA23	TBRA22	TBRA21	TBRA20	TBRA19	TBRA18	TBRA17	TBRA16	
H'FFFF FD4E		TBRA15	TBRA14	TBRA13	TBRA12	TBRA11	TBRA10	TBRA9	TBRA8	
H'FFFF FD4F		TBRA7	TBRA6	TBRA5	TBRA4	TBRA3	TBRA2	TBRA1	TBRA0	
H'FFFF FD50	TDFAR	TDFA31	TDFA30	TDFA29	TDFA28	TDFA27	TDFA26	TDFA25	TDFA24	
H'FFFF FD51		TDFA23	TDFA22	TDFA21	TDFA20	TDFA19	TDFA18	TDFA17	TDFA16	
H'FFFF FD52		TDFA15	TDFA14	TDFA13	TDFA12	TDFA11	TDFA10	TDFA9	TDFA8	
H'FFFF FD53		TDFA7	TDFA6	TDFA5	TDFA4	TDFA3	TDFA2	TDFA1	TDFA0	
H'FFFF FD54 ~ H'FFFF FD5F	-	-	-	-	-	-	-	-	-	
H'FFFF FD60	ECMR	-	-	-	-	-	-	-	-	EtherC
H'FFFF FD61		-	-	-	-	-	-	-	-	
H'FFFF FD62		-	-	-	PRCEF	-	-	MPDE	-	
H'FFFF FD63		-	RE	TE	-	ILB	ELB	DM	PRM	
H'FFFF FD64	ECSR	-	-	-	-	-	-	-	-	
H'FFFF FD65		-	-	-	-	-	-	-	-	
H'FFFF FD66		-	-	-	-	-	-	-	-	
H'FFFF FD67		-	-	-	-	-	LCHNG	MPD	ICD	
H'FFFF FD68	ECSIPR	-	-	-	-	-	-	-	-	
H'FFFF FD69		-	-	-	-	-	-	-	-	
H'FFFF FD6A		-	-	-	-	-	-	-	-	
H'FFFF FD6B		-	-	-	-	-	LCHNGIP	MPDIP	ICDIP	
H'FFFF FD6C	PIR	-	-	-	-	-	-	-	-	
H'FFFF FD6D		-	-	-	-	-	-	-	-	
H'FFFF FD6E		-	-	-	-	-	-	-	-	
H'FFFF FD6F		-	-	-	-	MDI	MDO	MMD	MDC	
H'FFFF FD70	MAHR	MA47	MA46	MA45	MA44	MA43	MA42	MA41	MA40	
H'FFFF FD71		MA39	MA38	MA37	MA36	MA35	MA34	MA33	MA32	
H'FFFF FD72		MA31	MA30	MA29	MA28	MA27	MA26	MA25	MA24	
H'FFFF FD73		MA23	MA22	MA21	MA20	MA19	MA18	MA17	MA16	
H'FFFF FD74	MALR	-	-	-	-	-	-	-	-	
H'FFFF FD75		-	-	-	-	-	-	-	-	
H'FFFF FD76		MA15	MA14	MA13	MA12	MA11	MA10	MA9	MA8	
H'FFFF FD77		MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0	
H'FFFF FD78	RFLR	-	-	-	-	-	-	-	-	
H'FFFF FD79		-	-	-	-	-	-	-	-	
H'FFFF FD7A		-	-	-	-	RFL11	RFL10	RFL9	RFL8	
H'FFFF FD7B		RFL7	RFL6	RFL5	RFL4	RFL3	RFL2	RFL1	RFL0	

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FD7C	PSR	-	-	-	-	-	-	-	-	EtherC
H'FFFF FD7D		-	-	-	-	-	-	-	-	
H'FFFF FD7E		-	-	-	-	-	-	-	-	
H'FFFF FD7F		-	-	-	-	-	-	-	LMON	
H'FFFF FD80	TROCR	-	-	-	-	-	-	-		
H'FFFF FD81		-	-	-	-	-	-	-		-
H'FFFF FD82		TROC15	TROC14	TROC13	TROC12	TROC11	TROC10	TROC9		TROC8
H'FFFF FD83		TROC7	TROC6	TROC5	TROC4	TROC3	TROC2	TROC1		TROC0
H'FFFF FD84	CDCR	-	-	-	-	-	-	-		
H'FFFF FD85		-	-	-	-	-	-	-		-
H'FFFF FD86		COLDC15	COLDC14	COLDC13	COLDC12	COLDC11	COLDC10	COLDC9		COLDC8
H'FFFF FD87		COLDC7	COLDC6	COLDC5	COLDC4	COLDC3	COLDC2	COLDC1		COLDC0
H'FFFF FD88	LCCR	-	-	-	-	-	-	-		
H'FFFF FD89		-	-	-	-	-	-	-		-
H'FFFF FD8A		LCC15	LCC14	LCC13	LCC12	LCC11	LCC10	LCC9		LCC8
H'FFFF FD8B		LCC7	LCC6	LCC5	LCC4	LCC3	LCC2	LCC1		LCC0
H'FFFF FD8C	CNDCR	-	-	-	-	-	-	-		
H'FFFF FD8D		-	-	-	-	-	-	-		-
H'FFFF FD8E		CNDC15	CNDC14	CNDC13	CNDC12	CNDC11	CNDC10	CNDC9		CNDC8
H'FFFF FD8F		CNDC7	CNDC6	CNDC5	CNDC4	CNDC3	CNDC2	CNDC1		CNDC0
H'FFFF FD90	IFLCR	-	-	-	-	-	-	-		
H'FFFF FD91		-	-	-	-	-	-	-		-
H'FFFF FD92		IFLC15	IFLC14	IFLC13	IFLC12	IFLC11	IFLC10	IFLC9		IFLC8
H'FFFF FD93		IFLC7	IFLC6	IFLC5	IFLC4	IFLC3	IFLC2	IFLC1		IFLC0
H'FFFF FD94	CEFCR	-	-	-	-	-	-	-		
H'FFFF FD95		-	-	-	-	-	-	-		-
H'FFFF FD96		CEFC15	CEFC14	CEFC13	CEFC12	CEFC11	CEFC10	CEFC9		CEFC8
H'FFFF FD97		CEFC7	CEFC6	CEFC5	CEFC4	CEFC3	CEFC2	CEFC1		CEFC0
H'FFFF FD98	FRECR	-	-	-	-	-	-	-		
H'FFFF FD99		-	-	-	-	-	-	-		-
H'FFFF FD9A		FREC15	FREC14	FREC13	FREC12	FREC11	FREC10	FREC9		FREC8
H'FFFF FD9B		FREC7	FREC6	FREC5	FREC4	FREC3	FREC2	FREC1		FREC0
H'FFFF FD9C	TSFCR	-	-	-	-	-	-	-		
H'FFFF FD9D		-	-	-	-	-	-	-		-
H'FFFF FD9E		TSFC15	TSFC14	TSFC13	TSFC12	TSFC11	TSFC10	TSFC9		TSFC8
H'FFFF FD9F		TSFC7	TSFC6	TSFC5	TSFC4	TSFC3	TSFC2	TSFC1		TSFC0
H'FFFF FDA0	TLFCR	-	-	-	-	-	-	-		
H'FFFF FDA1		-	-	-	-	-	-	-		-
H'FFFF FDA2		TLFC15	TLFC14	TLFC13	TLFC12	TLFC11	TLFC10	TLFC9		TLFC8
H'FFFF FDA3		TLFC7	TLFC6	TLFC5	TLFC4	TLFC3	TLFC2	TLFC1		TLFC0
H'FFFF FDA4	RFCR	-	-	-	-	-	-	-		
H'FFFF FDA5		-	-	-	-	-	-	-		-
H'FFFF FDA6		RFC15	RFC14	RFC13	RFC12	RFC11	RFC10	RFC9		RFC8
H'FFFF FDA7		RFC7	RFC6	RFC5	RFC4	RFC3	RFC2	RFC1		RFC0

付 録

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FDA8	MAFCR	-	-	-	-	-	-	-	-	EtherC
H'FFFF FDA9		-	-	-	-	-	-	-	-	
H'FFFF FDAA		MAFC15	MAFC14	MAFC13	MAFC12	MAFC11	MAFC10	MAFC9	MAFC8	
H'FFFF FDAB		MAFC7	MAFC6	MAFC5	MAFC4	MAFC3	MAFC2	MAFC1	MAFC0	
H'FFFF FDAC ~ H'FFFF FE0F	-	-	-	-	-	-	-	-	-	-
H'FFFF FE10	TIER	ICIE	-	-	-	OCIAE	OCIBE	OVIE	-	FRT
H'FFFF FE11	FTCSR	ICF	-	-	-	OCFA	OCFB	OVF	CCLRA	
H'FFFF FE12	FRCH									
H'FFFF FE13	FRCL									
H'FFFF FE14	OCRAH									
	OCRBH									
H'FFFF FE15	OCRAL									
	OCRBL									
H'FFFF FE16	TCR	IEDG	-	-	-	-	-	CKS1	CKS0	
H'FFFF FE17	TOCR	-	-	-	OCRS	-	-	OLVLA	OLVLB	
H'FFFF FE18	FICRH									
H'FFFF FE19	FICRL									
H'FFFF FE1A ~ H'FFFF FE3F	-	-	-	-	-	-	-	-	-	-
H'FFFF FE40	IPRD	TPU0IP3	TPU0IP2	TPU0IP1	TPU0IP0	TPU1IP3	TPU1IP2	TPU1IP1	TPU1IP0	INTC
H'FFFF FE41		TPU2IP3	TPU2IP2	TPU2IP1	TPU2IP0	SCF1IP3	SCF1IP2	SCF1IP1	SCF1IP0	
H'FFFF FE42	VCRE	-	TG0AV6	TG0AV5	TG0AV4	TG0AV3	TG0AV2	TG0AV1	TG0AV0	
H'FFFF FE43		-	TG0BV6	TG0BV5	TG0BV4	TG0BV3	TG0BV2	TG0BV1	TG0BV0	
H'FFFF FE44	VCRF	-	TG0CV6	TG0CV5	TG0CV4	TG0CV3	TG0CV2	TG0CV1	TG0CV0	
H'FFFF FE45		-	TG0DV6	TG0DV5	TG0DV4	TG0DV3	TG0DV2	TG0DV1	TG0DV0	
H'FFFF FE46	VCRG	-	TC0V6	TC0V5	TC0V4	TC0V3	TC0V2	TC0V1	TC0V0	
H'FFFF FE47		-	-	-	-	-	-	-	-	
H'FFFF FE48	VCRH	-	TG1AV6	TG1AV5	TG1AV4	TG1AV3	TG1AV2	TG1AV1	TG1AV0	
		-	TG1BV6	TG1BV5	TG1BV4	TG1BV3	TG1BV2	TG1BV1	TG1BV0	
H'FFFF FE4A	VCRI	-	TC1V6	TC1V5	TC1V4	TC1V3	TC1V2	TC1V1	TC1V0	
		-	TC1UV6	TC1UV5	TC1UV4	TC1UV3	TC1UV2	TC1UV1	TC1UV0	
H'FFFF FE4C	VCRJ	-	TG2AV6	TG2AV5	TG2AV4	TG2AV3	TG2AV2	TG2AV1	TG2AV0	
		-	TG2BV6	TG2BV5	TG2BV4	TG2BV3	TG2BV2	TG2BV1	TG2BV0	
H'FFFF FE4E	VCRK	-	TC2V6	TC2V5	TC2V4	TC2V3	TC2V2	TC2V1	TC2V0	
		-	TC2UV6	TC2UV5	TC2UV4	TC2UV3	TC2UV2	TC2UV1	TC2UV0	
H'FFFF FE50	VCRL	-	SER1V6	SER1V5	SER1V4	SER1V3	SER1V2	SER1V1	SER1V0	
		-	SRX1V6	SRX1V5	SRX1V4	SRX1V3	SRX1V2	SRX1V1	SRX1V0	
H'FFFF FE52	VCRM	-	SBR1V6	SBR1V5	SBR1V4	SBR1V3	SBR1V2	SBR1V1	SBR1V0	
		-	STX1V6	STX1V5	STX1V4	STX1V3	STX1V2	STX1V1	STX1V0	
H'FFFF FE54	VCRN	-	SER2V6	SER2V5	SER2V4	SER2V3	SER2V2	SER2V1	SER2V0	
		-	SRX2V6	SRX2V5	SRX2V4	SRX2V3	SRX2V2	SRX2V1	SRX2V0	
H'FFFF FE56	VCRO	-	SBR2V6	SBR2V5	SBR2V4	SBR2V3	SBR2V2	SBR2V1	SBR2V0	
		-	STX2V6	STX2V5	STX2V4	STX2V3	STX2V2	STX2V1	STX2V0	
H'FFFF FE58 ~ H'FFFF FE5F	-	-	-	-	-	-	-	-	-	-

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FE60	IPRB	E-DMACIP3	E-DMACIP2	E-DMACIP1	E-DMACIP0	FRTIP3	FRTIP2	FRTIP1	FRTIP0	INTC
H'FFFF FE61		-	-	-	-	-	-	-	-	
H'FFFF FE62	VCRA	-	EINV6	EINV5	EINV4	EINV3	EINV2	EINV1	EINV0	
H'FFFF FE63		-	-	-	-	-	-	-	-	
H'FFFF FE64	VCRB	-	-	-	-	-	-	-	-	
H'FFFF FE65		-	-	-	-	-	-	-	-	
H'FFFF FE66	VCRC	-	FICV6	FICV5	FICV4	FICV3	FICV2	FICV1	FICV0	
H'FFFF FE67		-	FOCV6	FOCV5	FOCV4	FOCV3	FOCV2	FOCV1	FOCV0	
H'FFFF FE68	VCRD	-	FOVV6	FOVV5	FOVV4	FOVV3	FOVV2	FOVV1	FOVV0	
H'FFFF FE69		-	-	-	-	-	-	-	-	
H'FFFF FE6A ~ H'FFFF FE70	-	-	-	-	-	-	-	-	-	-
H'FFFF FE71	DRCR0	-	-	-	RS4	RS3	RS2	RS1	RS0	DMAC
H'FFFF FE72	DRCR1	-	-	-	RS4	RS3	RS2	RS1	RS0	
H'FFFF FE73 ~ H'FFFF FE7F	-	-	-	-	-	-	-	-	-	-
H'FFFF FE80	WTCSR	OVF	WT/IT	TME	-	-	CKS2	CKS1	CKS0	WDT
H'FFFF FE81	WTCNT	-	-	-	-	-	-	-	-	
H'FFFF FE82	-	-	-	-	-	-	-	-	-	
H'FFFF FE83	RSTCSR	WOVF	RSTE	RSTS	-	-	-	-	-	
H'FFFF FE84 ~ H'FFFF FE8F	-	-	-	-	-	-	-	-	-	-
H'FFFF FE90	FMR	PLL2ST	PLL1ST	CKIOST	-	FR3	FR2	FR1	FR0	内蔵発振回路
H'FFFF FE91	SBYCR1	SBY	HIZ	MSTP5	MSTP4	MSTP3	-	MSTP1	-	低消費電力状態
H'FFFF FE92	CCR	W1	W0	WB	CP	TW	OD	ID	CE	キャッシュ
H'FFFF FE93	SBYCR2	-	-	MSTP11	MSTP10	MSTP9	MSTP8	MSTP7	MSTP6	低消費電力状態
H'FFFF FE94 ~ H'FFFF FEBF	-	-	-	-	-	-	-	-	-	-
H'FFFF FEC0	IPRE	SCF2IP3	SCF2IP2	SCF2IP1	SCF2IP0	SIO0IP3	SIO0IP2	SIO0IP1	SIO0IP0	INTC
H'FFFF FEC1		SIO1IP3	SIO1IP2	SIO1IP1	SIO1IP0	SIO2IP3	SIO2IP2	SIO2IP1	SIO2IP0	
H'FFFF FEC2	VCRP	-	RER0V6	RER0V5	RER0V4	RER0V3	RER0V2	RER0V1	RER0V0	
H'FFFF FEC3		-	TER0V6	TER0V5	TER0V4	TER0V3	TER0V2	TER0V1	TER0V0	
H'FFFF FEC4	VCRQ	-	RDF0V6	RDF0V5	RDF0V4	RDF0V3	RDF0V2	RDF0V1	RDF0V0	
H'FFFF FEC5		-	TDE0V6	TDE0V5	TDE0V4	TDE0V3	TDE0V2	TDE0V1	TDE0V0	
H'FFFF FEC6	VCRR	-	RER1V6	RER1V5	RER1V4	RER1V3	RER1V2	RER1V1	RER1V0	
H'FFFF FEC7		-	TER1V6	TER1V5	TER1V4	TER1V3	TER1V2	TER1V1	TER1V0	
H'FFFF FEC8	VCRS	-	RDF1V6	RDF1V5	RDF1V4	RDF1V3	RDF1V2	RDF1V1	RDF1V0	
H'FFFF FEC9		-	TDE1V6	TDE1V5	TDE1V4	TDE1V3	TDE1V2	TDE1V1	TDE1V0	
H'FFFF FECA	VCRT	-	RER2V6	RER2V5	RER2V4	RER2V3	RER2V2	RER2V1	RER2V0	
H'FFFF FECB		-	TER2V6	TER2V5	TER2V4	TER2V3	TER2V2	TER2V1	TER2V0	
H'FFFF FECC	VCRU	-	RDF2V6	RDF2V5	RDF2V4	RDF2V3	RDF2V2	RDF2V1	RDF2V0	
H'FFFF FECD		-	TDE2V6	TDE2V5	TDE2V4	TDE2V3	TDE2V2	TDE2V1	TDE2V0	

付 録

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FECE ~ H'FFFF FEDF	-	-	-	-	-	-	-	-	-	-
H'FFFF FEE0	ICR	NMIL	-	-	-	-	-	-	NMIE	INTC
H'FFFF FEE1		-	-	-	-	-	-	EXIMD	VECMD	
H'FFFF FEE2	IPRA	-	-	-	-	DMACIP3	DMACIP2	DMACIP1	DMACIP0	
H'FFFF FEE3		WDTIP3	WDTIP2	WDTIP1	WDTIP0	-	-	-	-	
H'FFFF FEE4	VCRWDT	-	WITV6	WITV5	WITV4	WITV3	WITV2	WITV1	WITV0	
H'FFFF FEE5		-	BCMV6	BCMV5	BCMV4	BCMV3	BCMV2	BCMV1	BCMV0	
H'FFFF FEE6	IPRC	IRQ0IP3	IRQ0IP2	IRQ0IP1	IRQ0IP0	IRQ1IP3	IRQ1IP2	IRQ1IP1	IRQ1IP0	
H'FFFF FEE7		IRQ2IP3	IRQ2IP2	IRQ2IP1	IRQ2IP0	IRQ3IP3	IRQ3IP2	IRQ3IP1	IRQ3IP0	
H'FFFF FEE8	IRQCSR	IRQ3IS	IRQ3OS	IRQ2IS	IRQ2OS	IRQ1IS	IRQ1OS	IRQ0IS	IRQ0OS	
H'FFFF FEE9		IRL3PS	IRL2PS	IRL1PS	IRL0PS	IRQ3F	IRQ2F	IRQ1F	IRQ0F	
H'FFFF FEEA ~ H'FFFF FEFF	-	-	-	-	-	-	-	-	-	-
H'FFFF FF00	BARAH	BAA31	BAA30	BAA29	BAA28	BAA27	BAA26	BAA25	BAA24	UBC
H'FFFF FF01		BAA23	BAA22	BAA21	BAA20	BAA19	BAA18	BAA17	BAA16	
H'FFFF FF02	BARAL	BAA15	BAA14	BAA13	BAA12	BAA11	BAA10	BAA9	BAA8	
H'FFFF FF03		BAA7	BAA6	BAA5	BAA4	BAA3	BAA2	BAA1	BAA0	
H'FFFF FF04	BAMRAH	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	
H'FFFF FF05		BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16	
H'FFFF FF06	BAMRAL	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	
H'FFFF FF07		BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0	
H'FFFF FF08	BBRA	-	-	-	-	-	-	-	-	
H'FFFF FF09		CPA1	CPA0	IDA1	IDA0	RWA1	RWA0	SZA1	SZA0	
H'FFFF FF0A ~ H'FFFF FF0F	-	-	-	-	-	-	-	-	-	-
H'FFFF FF10	BRFR	SVF	PID2	PID1	PID0	-	-	-	-	UBC
H'FFFF FF11		DVF	-	-	-	-	-	-	-	
H'FFFF FF12 ~ H'FFFF FF13	-	-	-	-	-	-	-	-	-	-
H'FFFF FF14	BRSRH	BSA31	BSA30	BSA29	BSA28	BSA27	BSA26	BSA25	BSA24	UBC
H'FFFF FF15		BSA23	BSA22	BSA21	BSA20	BSA19	BSA18	BSA17	BSA16	
H'FFFF FF16	BR SRL	BSA15	BSA14	BSA13	BSA12	BSA11	BSA10	BSA9	BSA8	
H'FFFF FF17		BSA7	BSA6	BSA5	BSA4	BSA3	BSA2	BSA1	BSA0	
H'FFFF FF18	BRDRH	BDA31	BDA30	BDA29	DA28	BDA27	BDA26	BDA25	BDA24	
H'FFFF FF19		BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16	
H'FFFF FF1A	BRDRL	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	
H'FFFF FF1B		BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0	
H'FFFF FF1C ~ H'FFFF FF1F	-	-	-	-	-	-	-	-	-	-

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FF20	BARBH	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	UBC
H'FFFF FF21		BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16	
H'FFFF FF22	BARBL	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	
H'FFFF FF23		BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0	
H'FFFF FF24	BAMRBH	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	
H'FFFF FF25		BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16	
H'FFFF FF26	BAMRBL	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	
H'FFFF FF27		BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0	
H'FFFF FF28	BBRB	-	-	-	-	-	-	-	-	
H'FFFF FF29		CPB1	CPB0	IDB1	IDB0	RWB1	RWB0	SZB1	SZB0	
H'FFFF FF2A ~ H'FFFF FF2F	-	-	-	-	-	-	-	-	-	-
H'FFFF FF30	BRCRH	CMFCA	CMFPA	-	-	PCTE	PCBA	-	-	UBC
H'FFFF FF31		CMFCB	CMFPB	-	SEQ1	SEQ0	PCBB	-	-	
H'FFFF FF32	BRCRL	CMFCC	CMFPC	ETBEC	-	DBEC	PCBC	-	-	
H'FFFF FF33		CMFCD	CMFPD	ETBED	-	DBED	PCBD	-	-	
H'FFFF FF34 ~ H'FFFF FF3F	-	-	-	-	-	-	-	-	-	-
H'FFFF FF40	BARCH	BAC31	BAC30	BAC29	BAC28	BAC27	BAC26	BAC25	BAC24	UBC
H'FFFF FF41		BAC23	BAC22	BAC21	BAC20	BAC19	BAC18	BAC17	BAC16	
H'FFFF FF42	BARCL	BAC15	BAC14	BAC13	BAC12	BAC11	BAC10	BAC9	BAC8	
H'FFFF FF43		BAC7	BAC6	BAC5	BAC4	BAC3	BAC2	BAC1	BAC0	
H'FFFF FF44	BAMRCH	BAMC31	BAMC30	BAMC29	BAMC28	BAMC27	BAMC26	BAMC25	BAMC24	
H'FFFF FF45		BAMC23	BAMC22	BAMC21	BAMC20	BAMC19	BAMC18	BAMC17	BAMC16	
H'FFFF FF46	BAMRCL	BAMC15	BAMC14	BAMC13	BAMC12	BAMC11	BAMC10	BAMC9	BAMC8	
H'FFFF FF47		BAMC7	BAMC6	BAMC5	BAMC4	BAMC3	BAMC2	BAMC1	BAMC0	
H'FFFF FF48	BBRC	-	-	-	-	-	-	XYEC	XYSC	
H'FFFF FF49		CPC1	CPC0	IDC1	IDC0	RWC1	RWC0	SZC1	SZC0	
H'FFFF FF4A ~ H'FFFF FF4F	-	-	-	-	-	-	-	-	-	-
H'FFFF FF50	BDRCH	BDC31	BDC30	BDC29	BDC28	BDC27	BDC26	BDC25	BDC24	UBC
H'FFFF FF51		BDC23	BDC22	BDC21	BDC20	BDC19	BDC18	BDC17	BDC16	
H'FFFF FF52	BDRCL	BDC15	BDC14	BDC13	BDC12	BDC11	BDC10	BDC9	BDC8	
H'FFFF FF53		BDC7	BDC6	BDC5	BDC4	BDC3	BDC2	BDC1	BDC0	
H'FFFF FF54	BDMRCH	BDMC31	BDMC30	BDMC29	BDMC28	BDMC27	BDMC26	BDMC25	BDMC24	
H'FFFF FF55		BDMC23	BDMC22	BDMC21	BDMC20	BDMC19	BDMC18	BDMC17	BDMC16	
H'FFFF FF56	BDMRCL	BDMC15	BDMC14	BDMC13	BDMC12	BDMC11	BDMC10	BDMC9	BDMC8	
H'FFFF FF57		BDMC7	BDMC6	BDMC5	BDMC4	BDMC3	BDMC2	BDMC1	BDMC0	
H'FFFF FF58	BETRC	-	-	-	-	ETRC11	ETRC10	ETRC9	ETRC8	
H'FFFF FF59		ETRC7	ETRC6	ETRC5	ETRC4	ETRC3	ETRC2	ETRC1	ETRC0	
H'FFFF FF5A ~ H'FFFF FF5F	-	-	-	-	-	-	-	-	-	-

付 録

アドレス	レジスタ 略称	ビット名								モジュール	
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'FFFF FF60	BARDH	BAD31	BAD30	BAD29	BAD28	BAD27	BAD26	BAD25	BAD24	UBC	
H'FFFF FF61		BAD23	BAD22	BAD21	BAD20	BAD19	BAD18	BAD17	BAD16		
H'FFFF FF62	BARDL	BAD15	BAD14	BAD13	BAD12	BAD11	BAD10	BAD9	BAD8		
H'FFFF FF63		BAD7	BAD6	BAD5	BAD4	BAD3	BAD2	BAD1	BAD0		
H'FFFF FF64	BAMRDH	BAMD31	BAMD30	BAMD29	BAMD28	BAMD27	BAMD26	BAMD25	BAMD24		
H'FFFF FF65		BAMD23	BAMD22	BAMD21	BAMD20	BAMD19	BAMD18	BAMD17	BAMD16		
H'FFFF FF66	BAMRDL	BAMD15	BAMD14	BAMD13	BAMD12	BAMD11	BAMD10	BAMD9	BAMD8		
H'FFFF FF67		BAMD7	BAMD6	BAMD5	BAMD4	BAMD3	BAMD2	BAMD1	BAMD0		
H'FFFF FF68	BBRD	-	-	-	-	-	-	XYED	XYSD		
H'FFFF FF69		CPD1	CPD0	IDD1	IDD0	RWD1	RWD0	SZD1	SZD0		
H'FFFF FF6A ~ H'FFFF FF6F	-	-	-	-	-	-	-	-	-	-	
H'FFFF FF70	BDRDH	BDD31	BDD30	BDD29	BDD28	BDD27	BDD26	BDD25	BDD24	UBC	
H'FFFF FF71		BDD23	BDD22	BDD21	BDD20	BDD19	BDD18	BDD17	BDD16		
H'FFFF FF72	BDRDL	BDD15	BDD14	BDD13	BDD12	BDD11	BDD10	BDD9	BDD8		
H'FFFF FF73		BDD7	BDD6	BDD5	BDD4	BDD3	BDD2	BDD1	BDD0		
H'FFFF FF74	BDMRDH	BDMD31	BDMD30	BDMD29	BDMD28	BDMD27	BDMD26	BDMD25	BDMD24		
H'FFFF FF75		BDMD23	BDMD22	BDMD21	BDMD20	BDMD19	BDMD18	BDMD17	BDMD16		
H'FFFF FF76	BDMRDL	BDMD15	BDMD14	BDMD13	BDMD12	BDMD11	BDMD10	BDMD9	BDMD8		USB
H'FFFF FF77		BDMD7	BDMD6	BDMD5	BDMD4	BDMD3	BDMD2	BDMD1	BDMD0		
H'FFFF FF78	BETRD	-	-	-	-	ETRD11	ETRD10	ETRD9	ETRD8		
H'FFFF FF79		ETRD7	ETRD6	ETRD5	ETRD4	ETRD3	ETRD2	ETRD1	ETRD0		
H'FFFF FF7A ~ H'FFFF FF7F	-	-	-	-	-	-	-	-	-	-	
H'FFFF FF80	SAR0									DMAC	
H'FFFF FF81											
H'FFFF FF82											
H'FFFF FF83											
H'FFFF FF84	DAR0										
H'FFFF FF85											
H'FFFF FF86											
H'FFFF FF87											
H'FFFF FF88	TCR0	-	-	-	-	-	-	-	-		
H'FFFF FF89											
H'FFFF FF8A											
H'FFFF FF8B											
H'FFFF FF8C	CHCR0	-	-	-	-	-	-	-	-		
H'FFFF FF8D		-	-	-	-	-	-	-	-		
H'FFFF FF8E		DM1	DM0	SM1	SM0	TS1	TS0	AR	AM		
H'FFFF FF8F		AL	DS	DL	TB	TA	IE	TE	DE		
H'FFFF FF90	SAR1										
H'FFFF FF91											
H'FFFF FF92											
H'FFFF FF93											

アドレス	レジスタ 略称	ビット名								モジュール
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF FF94	DAR1									DMAC
H'FFFF FF95										
H'FFFF FF96										
H'FFFF FF97										
H'FFFF FF98	TCR1	-	-	-	-	-	-	-	-	
H'FFFF FF99										
H'FFFF FF9A										
H'FFFF FF9B										
H'FFFF FF9C	CHCR1	-	-	-	-	-	-	-	-	
H'FFFF FF9D										
H'FFFF FF9E		DM1	DM0	SM1	SM0	TS1	TS0	AR	AM	
H'FFFF FF9F		AL	DS	DL	TB	TA	IE	TE	DE	
H'FFFF FFA0	VCRDMA0	-	-	-	-	-	-	-	-	
H'FFFF FFA1										
H'FFFF FFA2										
H'FFFF FFA3		VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	
H'FFFF FFA4 ~ H'FFFF FFA7	-	-	-	-	-	-	-	-	-	-
H'FFFF FFA8	VCRDMA1	-	-	-	-	-	-	-	-	DMAC
H'FFFF FFA9										
H'FFFF FFAA										
H'FFFF FFAB		VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	
H'FFFF FFAC ~ H'FFFF FFAF	-	-	-	-	-	-	-	-	-	-
H'FFFF FFB0	DMAOR	-	-	-	-	-	-	-	-	DMAC
H'FFFF FFB1										
H'FFFF FFB2										
H'FFFF FFB3						PR	AE	NMIF	DME	
H'FFFF FFB4 ~ H'FFFF FFBF	-	-	-	-	-	-	-	-	-	-
H'FFFF FFC0	WCR2	A4WD1	A4WD0	-	A4WM	A3WM	A2WM	A1WM	A0WM	BSC
H'FFFF FFC1						IW41	IW40	W41	W40	
H'FFFF FFC2 ~ H'FFFF FFC3	-	-	-	-	-	-	-	-	-	-
H'FFFF FFC4	WCR3	-	-	A4SW2	A4SW1	A4SW0	-	A4HW1	A4HW0	BSC
H'FFFF FFC5		A3SHW1	A3SHW0	A2SHW1	A2SHW0	A1SHW1	A1SHW0	A0SHW1	A0SHW0	
H'FFFF FFC6 ~ H'FFFF FFDF	-	-	-	-	-	-	-	-	-	-
H'FFFF FFE0	BCR1	-	A4LW1	A4LW0	A2ENDIA N	BSTROM	-	AHLW1	AHLW0	BSC
H'FFFF FFE1		A1LW1	A1LW0	A0LW1	A0LW0	A4ENDIA N	DRAM2	DRAM1	DRAM0	
H'FFFF FFE2 ~ H'FFFF FFE3	-	-	-	-	-	-	-	-	-	-
H'FFFF FFE4	BCR2	-	-	-	-	-	-	A4SZ1	A4SZ0	BSC
H'FFFF FFE5		A3SZ1	A3SZ0	A2SZ1	A2SZ0	A1SZ1	A1SZ0	-	-	

付 録

アドレス	レジスタ 略称	ビット名								モジュール	
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'FFFF FFE6 ~ H'FFFF FFE7	-	-	-	-	-	-	-	-	-	-	-
H'FFFF FFE8	WCR1	IW31	IW30	IW21	IW20	IW11	IW10	IW01	IW00	BSC	
H'FFFF FFE9		W31	W30	W21	W20	W11	W10	W01	W00		
H'FFFF FFEA ~ H'FFFF FFEB	-	-	-	-	-	-	-	-	-	-	
H'FFFF FFEC	MCR	TRP0	RCD0	TRWLO	TRAS1	TRAS0	BE	RASD	TRWL1	BSC	
H'FFFF FFED		AMX2	SZ	AMX1	AMX0	RFSH	RMODE	TRP1	RCD1		
H'FFFF FFEF ~ H'FFFF FFEF	-	-	-	-	-	-	-	-	-	-	
H'FFFF FFF0	RTCSR	-	-	-	-	-	-	-	-	BSC	
H'FFFF FFF1		CMF	CMIE	CKS2	CKS1	CKS0	RRC2	RRC1	RRC0		
H'FFFF FFF2 ~ H'FFFF FFF3	-	-	-	-	-	-	-	-	-	-	
H'FFFF FFF4	RTCNT	-	-	-	-	-	-	-	-	BSC	
H'FFFF FFF5											
H'FFFF FFF6 ~ H'FFFF FFF7	-	-	-	-	-	-	-	-	-	-	
H'FFFF FFF8	RTCOR	-	-	-	-	-	-	-	-	BSC	
H'FFFF FFF9											
H'FFFF FFFA ~ H'FFFF FFFB	-	-	-	-	-	-	-	-	-	-	
H'FFFF FFFC	BCR3	-	-	-	-	A4LW2	AHLW2	A1LW2	A0LW2	BSC	
H'FFFF FFFD		DSWW1	DSWW0	-	-	-	BASEL	EDO	BWE		
H'FFFF FFFE ~ H'FFFF FFFF	-	-	-	-	-	-	-	-	-	-	

## B. 端子状態

## B.1 リセット、低消費電力状態、バス権解放状態での端子状態

分類	端子名	端子状態						
		パワーオンリセット	マニュアルリセット		低消費電力状態			バス権解放状態
			バス権獲得時	バス権解放時	スタンバイ HIZ = 0	スタンバイ HIZ = 1	スリープ	
バス制御	A24 ~ A0	O	O	Z	Z	Z	O	Z
	D31 ~ D0	Z	IO	Z	Z	Z	IO	Z
	CS4 ~ CS0	H	O	Z	H	H	H	Z
	RD/WR	H	O	Z	H	H	H	Z
	RAS	H	O	Z	H	H	H	Z
	CAS/OE	H	O	Z	H	H	H	Z
	WAIT	Z	I	Z	Z	Z	I	無視
	BS	H	O	Z	H	H	H	Z
	RD	H	O	Z	H	H	H	Z
	BGR	H	O	O	H	H	O	O
	BRLS	Z	I	I	Z	Z	I	I
	CKE	H	O	H	O	O	O	H
	DQMUU/WE3	H	O	Z	H	H	H	Z
	DQMUL/WE2	H	O	Z	H	H	H	Z
	DQMLU/WE1	H	O	Z	H	H	H	Z
	DQMLL/WE0	H	O	Z	H	H	H	Z
	REFOUT	L	O	O	L	Z	O	O
	CAS3 ~ CAS0	H	O	Z	H	H	H	Z
	BH	H	O	Z	H	H	H	Z
	BUSHIZ	Z	I	Z	Z	Z	I	無視
割り込み	NMI	I	I	I	I	I	I	I
	IRL3 ~ IRL0	Z	Z	Z	I	I	I	I
	IVECF	H	H	H	H	Z	H	H
クロック	XTAL	O*	O*	O*	O*	O*	O*	O*
	EXTAL	I*	I*	I*	I*	I*	I*	I*
	CKIO	IO*	IO*	IO*	IO*	IO*	IO*	IO*
	CKPACK	H	H	H	H	H	H	H
	CKPREQ/CKM	I	I	I	I	I	I	I
	PLLCAP2, PLLCAP1	IO	IO	IO	IO	IO	IO	IO
DMAC	DREQ1, DREQ0	Z	Z	Z	Z	Z	I	I
	DACK1, DACK0	H	H	H	K	Z	O	O
システム制御	RES	I	I	I	I	I	I	I
	MD4 ~ MD0	I	I	I	I	I	I	I

分類	端子名	端子状態						バス権解放状態
		パワーオンリセット	マニュアルリセット		低消費電力状態			
			バス権獲得時	バス権解放時	スタンバイ HIZ = 0	スタンバイ HIZ = 1	スリープ	
ポート、 内蔵周辺 モジュール	PB15/SCK1	Z	IO/Z	IO/Z	K	Z	IO	IO
	PB14/RXD1	Z	IO/Z	IO/Z	K	Z	IO/I	IO/I
	PB13/TXD1	Z	IO/Z	IO/Z	K	Z	IO/O	IO/O
	PB12/SRCK2/RTS/ STATS1	Z	IO/Z/Z/O	IO/Z/Z/O	K/K/K/O	Z	IO/I/O/O	IO/I/O/O
	PB11/SRS2/CTS/ STATS0	Z	IO/Z/Z/O	IO/Z/Z/O	K/K/K/O	Z	IO/I/I/O	IO/I/I/O
	PB10/SRXD2/TIOCA1	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/O	IO/I/O
	PB9/STCK2/TIOCB1, TCLKC	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/O	IO/I/O
	PB8/STS2/TIOCA2	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/O/O	IO/O/O
	PB7/STXD2/TIOCB2, TCLKD	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/O/O	IO/O/O
	PB6/SRCK1/SCK2	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/O	IO/I/O
	PB5/SRS1/RXD2	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/I	IO/I/I
	PB4/SRXD1/TXD2	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/O	IO/I/O
	PB3/STCK1/TIOCA0	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/I/O	IO/I/O
	PB2/STS1/TIOCB0	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/O/O	IO/O/O
	PB1/STXD1/TIOCC0, TCLKA	Z	IO/Z/Z	IO/Z/Z	K/K/K	Z	IO/O/O	IO/O/O
	PB0/TIOCD0, TCLKB/ WOL	Z	IO/Z/O	IO/Z/O	K/K/O	Z	IO/O/O	IO/O/O
	PA13/SRCK0	Z	IO/Z	IO/Z	K/K	Z	IO/I	IO/I
	PA12/SRS0	Z	IO/Z	IO/Z	K/K	Z	IO/I	IO/I
	PA11/SRXD0	Z	IO/Z	IO/Z	K/K	Z	IO/I	IO/I
	PA10/STCK0	Z	IO/Z	IO/Z	K/K	Z	IO/I	IO/I
	PA9/STS0	Z	IO/Z	IO/Z	K/K	Z	IO/O	IO/O
	PA8/STXD0	Z	IO/Z	IO/Z	K/K	Z	IO/O	IO/O
	WDTOVF/PA7	H	H/O	H/O	O/K	O/Z	O/O	O/O
	PA6/FTCI	Z	IO/Z	IO/Z	K	Z	IO/I	IO/I
	PA5/FTI	Z	IO/Z	IO/Z	K	Z	IO/I	IO/I
	PA4/FTOA	Z	IO/L	IO/L	K	Z	IO/O	IO/O
	CKPO/FTOB	H	H/L	H/L	K	Z	O/O	O/O
	PA2/LNKSTA	Z	IO/I	IO/I	K	Z	IO/I	IO/I
	PA1/EXOUT	Z	IO/O	IO/O	K	Z	IO/O	IO/O
	PA0	Z	IO	IO	K	Z	IO	IO
H-UDI	TRST	I	I	I	I	I	I	I
	TCK	I	I	I	I	I	I	I
	TMS	I	I	I	I	I	I	I
	TDI	I	I	I	I	I	I	I
	TDO	O	O	O	O	O	O	O
	ASEMODE	I	I	I	I	I	I	I

分類	端子名	端子状態						バス権解放状態
		パワーオンリセット	マニュアルリセット		低消費電力状態			
			バス権獲得時	バス権解放時	スタンバイ HIZ = 0	スタンバイ HIZ = 1	スリープ	
EtherC	TX-CLK	I	I	I	I	I	I	I
	TX-EN	O	O	O	O	O	O	O
	TX-ER	O	O	O	O	O	O	O
	ETXD3 ~ ETXD0	O	O	O	O	O	O	O
	CRS	I	I	I	I	I	I	I
	COL	I	I	I	I	I	I	I
	MDC	O	O	O	O	O	O	O
	MDIO	IO	IO	IO	IO	IO	IO	IO
	RX-CLK	I	I	I	I	I	I	I
	RX-DV	I	I	I	I	I	I	I
	RX-ER	I	I	I	I	I	I	I
	ERXD3 ~ ERXD0	I	I	I	I	I	I	I

## 【記号説明】

- I : 入力
- O : 出力
- H : ハイレベル出力
- L : ローレベル出力
- Z : ハイインピーダンス
- K : 入力端子はハイインピーダンス、出力端子は状態を保持

【注】 スリープのとき、DMAC が動作していればアドレス / データバスおよび、バス制御信号は DMAC の動作に依存して変化します (リフレッシュ時も同様です)。

- \* クロックモードに依存します (CKPREQN、MD2 ~ 0 端子の設定)。

## C. 型名一覧

表 C.1 SH7615 型名一覧

略称	電圧	動作周波数	製品型名	パッケージ
SH7615	$V_{CC} = PLLV_{CC} = 3.3V$ $PV_{CC} = 5.0V/3.3V$	62.5MHz	HD6417615ARF	FP-208C
			HD6417615ARFV	FP-208CV
			HD6417615ARBP	BP-240A
			HD6417615ARBPV	BP-240AV

## D. 外形寸法図

SH7615 の外形寸法図を図 D.1、図 D.2 に示します。

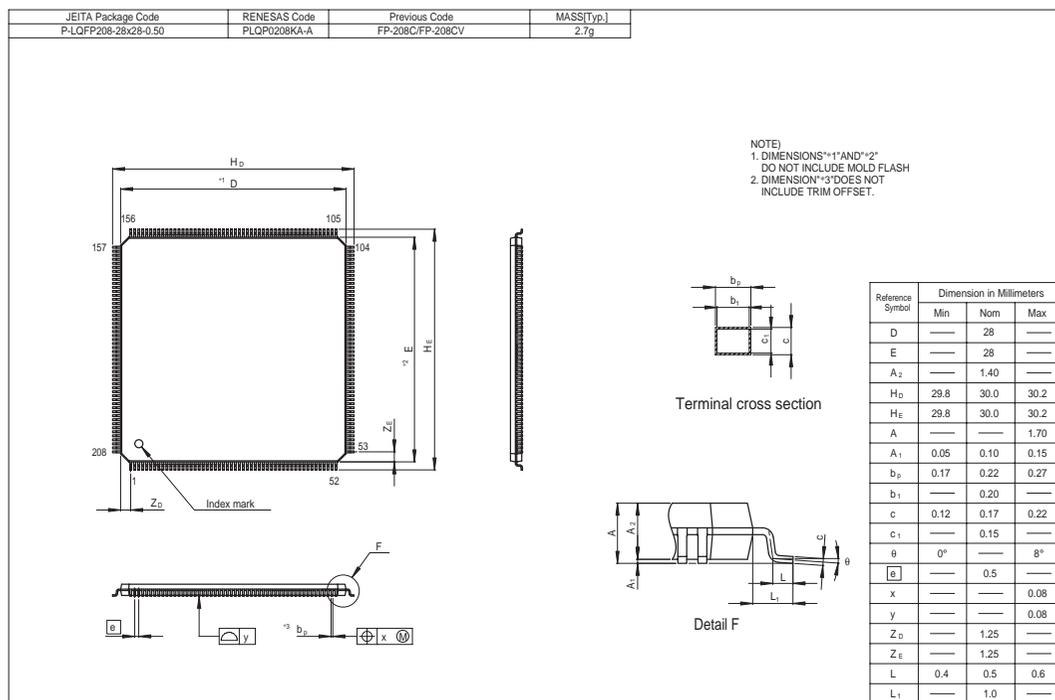


図 D.1 外形寸法図 (FP-208C、FP-208CV)

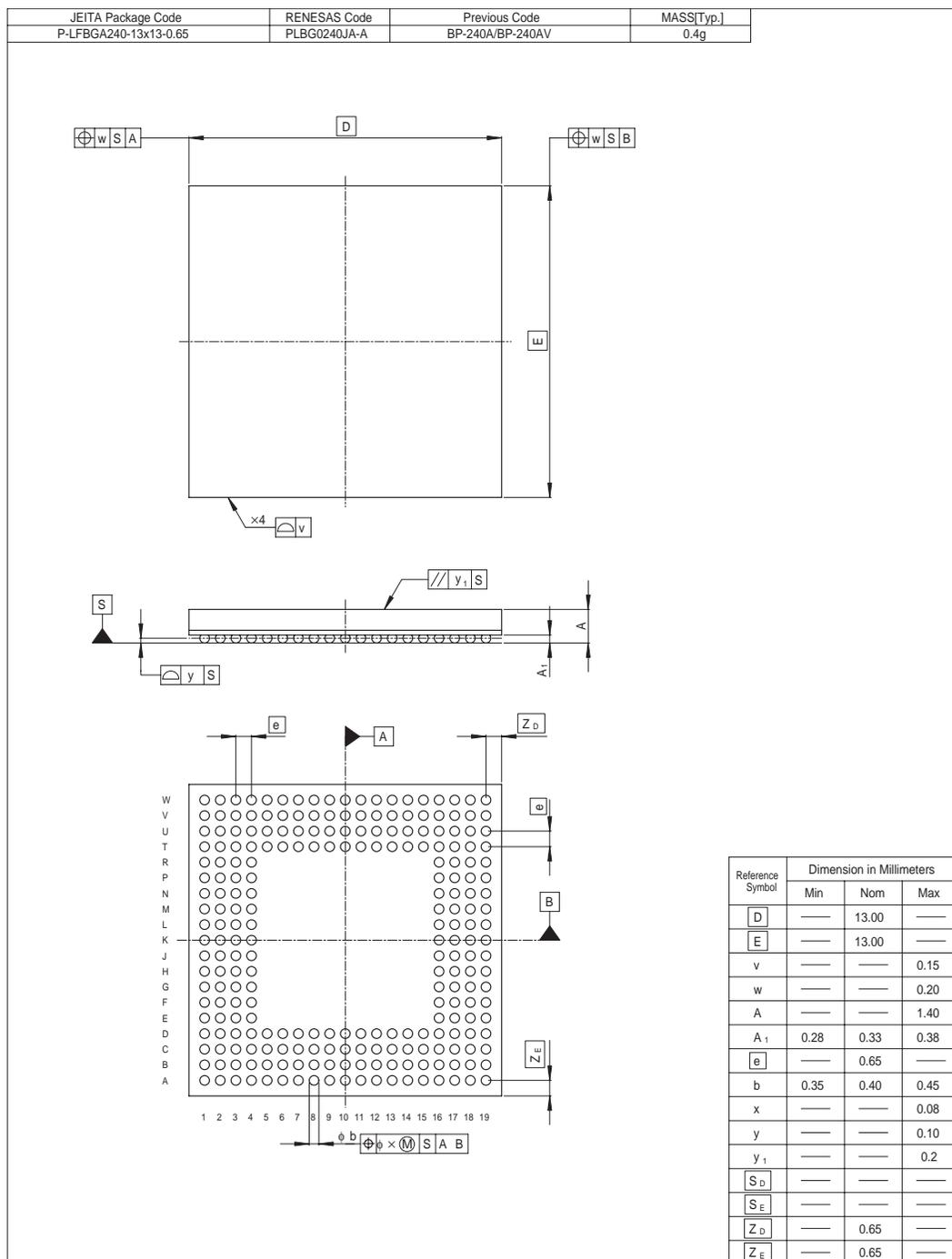


図 D.2 外形寸法図 (BP-240A、BP-240AV)

---

ルネサス32ビットRISCマイクロコンピュータ  
SH7615 ハードウェアマニュアル

発行年月日 2000年8月 第1版  
2005年3月8日 Rev.2.00

発行 株式会社ルネサステクノロジ 営業企画統括部  
〒100-0004 東京都千代田区大手町 2-6-2

編集 株式会社ルネサス小平セミコン 技術ドキュメント部

営業お問合せ窓口  
株式会社ルネサス販売



<http://www.renesas.com>

本		支	社	〒100-0004	千代田区大手町2-6-2 (日本ビル)	(03) 5201-5350
京	浜		社	〒212-0058	川崎市幸区鹿島田890-12 (新川崎三井ビル)	(044) 549-1662
西	東	京	支	〒190-0023	立川市柴崎町2-2-23 (第二高島ビル2F)	(042) 524-8701
札	幌		支	〒060-0002	札幌市中央区北二条西4-1 (札幌三井ビル5F)	(011) 210-8717
東	北		支	〒980-0013	仙台市青葉区花京院1-1-20 (花京院スクエア13F)	(022) 221-1351
い	わ	き	支	〒970-8026	いわき市平小太郎町4-9 (平小太郎ビル)	(0246) 22-3222
茨	城		支	〒312-0034	ひたちなか市堀口832-2 (日立システムプラザ勝田1F)	(029) 271-9411
新	潟		支	〒950-0087	新潟市東大通1-4-2 (新潟三井物産ビル3F)	(025) 241-4361
松	本		支	〒390-0815	松本市深志1-2-11 (昭和ビル7F)	(0263) 33-6622
中	部	営	業	〒460-0008	名古屋市中区栄4-2-29 (名古屋広小路ブレイス)	(052) 249-3330
浜	松		支	〒430-7710	浜松市板屋町111-2 (浜松アクトタワー10F)	(053) 451-2131
西	部	営	業	〒541-0044	大阪市中央区伏見町4-1-1 (明治安田生命大阪御堂筋ビル)	(06) 6233-9500
北	陸		支	〒920-0031	金沢市広岡3-1-1 (金沢パークビル8F)	(076) 233-5980
広	島		支	〒730-0036	広島市中区袋町5-25 (広島袋町ビルディング8F)	(082) 244-2570
鳥	取		支	〒680-0822	鳥取市今町2-251 (日本生命鳥取駅前ビル)	(0857) 21-1915
九	州		支	〒812-0011	福岡市博多区博多駅前2-17-1 (ヒロカネビル本館5F)	(092) 481-7695
鹿	児	島	支	〒890-0053	鹿児島市中央町12-2 (明治安田生命鹿児島中央町ビル)	(099) 284-1748

■技術的なお問合せおよび資料のご請求は下記へどうぞ。  
総合お問合せ窓口：カスタマサポートセンタ E-Mail: [csc@renesas.com](mailto:csc@renesas.com)

SH7615  
ハードウェアマニュアル



ルネサス エレクトロニクス株式会社  
神奈川県川崎市中原区下沼部1753 〒211-8668

RJJ09B0129-0200