

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用途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、海底中継機器、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。

ASM72 V.1.20

ユーザーズマニュアル

720シリーズ用アブソリュートアセンブラ

安全設計に関するお願い

- 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

- 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株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズ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。
- 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、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは責任を負いません。
- 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、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、予告なしに、本資料に記載した製品又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、事前に株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス 販売又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。
- 本資料に記載した情報は、正確を期すため、慎重に制作したのですが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、株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズはその責任を負いません。
- 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。株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズは、適用可否に対する責任を負いません。
- 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、株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス 販売又は特約店へご照会ください。
- 本資料の転載、複製については、文書による株式会社ルネサス テクノロジおよび株式会社ルネサス ソリューションズの事前の承諾が必要です。
- 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたら株式会社ルネサス テクノロジ、株式会社ルネサス ソリューションズ、株式会社ルネサス 販売又は特約店までご照会ください。

製品内容及び本書についてのお問い合わせ先

インストーラが生成する以下のテキストファイルに必要な事項を記入の上、ツール技術サポート窓口 support_tool@renesas.com まで送信ください。

¥SUPPORT¥製品名¥SUPPORT.TXT

株式会社ルネサス ソリューションズ

ツール技術サポート窓口	support_tool@renesas.com
ユーザ登録窓口	regist_tool@renesas.com
ホームページ	http://www.renesas.com/jp/tools

はじめに

PD72 は 720 シリーズ専用のエミュレータデバッガです。本書は、PD72 に含まれている以下の 3 つのソフトウェアの機能と操作方法について記述しています。

本製品に同封されていますリリースノートには、マニュアルの補足等が記載されていますので必ずお読みください。

1. アブソリュートアセンブラ **ASM72**
2. クロスリファレンサ **CRF72**
3. ブランチ範囲外エラー修正ツール **BRANCH**

本書の構成

本書は以下に示す 3 部から構成されています。マニュアルは、各ソフトウェアごとに 3 部に分かれています。

- 第 1 部 **ASM72** 操作マニュアル
アブソリュートアセンブラ **ASM72** の操作方法とソースプログラム記述方法について説明します。
- 第 2 部 **CRF72** 操作マニュアル
クロスリファレンサ **CRF72** の操作方法について説明します。
- 第 3 部 **BRANCH** 操作マニュアル
ブランチ範囲外エラー修正ツール **BRANCH** の操作方法について説明します。

MEMO

第1部

720シリーズ用
アブソリュートアセンブラ

ASM72操作マニュアル

目次

1 マニュアルの構成.....	1-4
2 概要.....	1-5
2.1 機能.....	1-5
2.2 生成ファイル.....	1-5
2.2.1 HEX ファイル.....	1-5
2.2.2 PRN ファイル.....	1-7
2.2.3 TAG ファイル.....	1-8
2.2.4 SYM ファイル.....	1-8
2.2.5 MAP ファイル.....	1-9
3 ソースプログラムの記述方法.....	1-12
3.1 ソースプログラムの構成.....	1-12
3.2 行の構成.....	1-12
3.2.1 MCU 名指定行.....	1-12
3.2.2 アセンブリ言語命令.....	1-13
3.2.3 ビットシンボル加減算行.....	1-13
3.2.4 擬似命令行.....	1-14
3.2.5 マクロ命令行.....	1-14
3.2.6 コメント行.....	1-15
3.3 欄の記述方法.....	1-15
3.3.1 シンボルタイプ 1~4/ラベル欄.....	1-15
3.3.5 コメント欄.....	1-15
3.4 オペランドデータ形式.....	1-15
3.4.1 数値定数.....	1-16
3.4.2 文字定数.....	1-16
3.4.3 記号定数.....	1-16
3.4.4 式.....	1-16
3.5 特殊文字.....	1-17
3.6 演算子.....	1-17
4 擬似命令.....	1-18
4.1 擬似命令の機能.....	1-18
4.2 アセンブル制御.....	1-19
4.2.1 アセンブル終了宣言.....	1-19
4.2.2 条件付きアセンブル.....	1-19
4.2.3 ファイル読み込み.....	1-19
4.3 アドレス制御.....	1-19
4.3.1 同義定義.....	1-19
4.3.2 アドレス宣言.....	1-19
4.3.3 データ設定.....	1-19
4.4 リスト制御.....	1-19
4.4.1 改ページおよびタイトル指定.....	1-19
4.4.2 リスト形式指定.....	1-19
4.4.3 リスト出力/抑止指定.....	1-20

5 マクロ命令	1-21
5.1 マクロ命令の機能	1-21
5.2 ビットマクロ命令	1-21
5.2.1 指定ビットのクリア	1-21
5.2.2 指定ビットの設定	1-21
5.2.3 指定ビットの状態によりプログラムの流れを変更	1-21
5.3 レジスタマクロ命令	1-21
5.3.1 Z, X, Y の値設定	1-21
6 操作方法	1-22
6.1 起動方法	1-22
6.2 入力パラメータ	1-22
6.2.1 ソースファイル名	1-22
6.2.2 コマンドパラメータ	1-22
6.3 入力方法	1-24
6.4 エラー	1-25
6.4.1 エラーの種類	1-25
6.4.2 DOS への戻り値	1-26
A エラーメッセージ一覧表	1-27
A.1 システムエラー一覧表	1-27
A.2 アセンブルエラー一覧表	1-28
A.3 ワーニング一覧表	1-30
B 擬似命令一覧	1-31
B.1 擬似命令一覧の見方	1-31
B.2 擬似命令一覧	1-31
C マクロ命令一覧	1-36
C.1 マクロ命令一覧の見方	1-36
C.2 マクロ命令一覧	1-36

第 1 章

マニュアルの構成

ASM72 操作マニュアルは、以下の章から構成されています。

- 第 2 章 概要
ASM72 の基本的な機能について説明します。
- 第 3 章 ソースプログラムの記述方法
ASM72 で処理するソースプログラムの記述方法について説明します。
- 第 4 章 疑似命令
ASM72 で使用可能な疑似命令について説明します。
- 第 5 章 マクロ命令
ASM72 で使用可能なマクロ命令について説明します。
- 第 6 章 操作方法
ASM72 の操作方法について説明します。
- 付録 A エラーメッセージ一覧表
ASM72 が表示するエラーメッセージについて、その内容と対策を一覧表で示します。
- 付録 B 疑似命令一覧
ASM72 で使用可能な全疑似命令について、疑似命令ごとの内容を示します。
- 付録 C マクロ命令一覧
ASM72 で使用可能な全マクロ命令について、マクロ命令ごとの内容を示します。

本マニュアルは、ASM72 V.1.20.xx に対応しています。

第2章

概要

ASM72は720シリーズ用のアブソリュートアセンブラです。アセンブリ言語で記述されたソースプログラム(以下ソースファイルと呼びます。)を機械語データに変換します。以下、この作業をアセンブルと呼びます。

2.1 機能

ASM72には、以下のような機能があります。

1. エラー内容を格納したタグファイル¹を生成できます(アセンブルエラーの修正を効率的に行うことができます)。
2. アセンブルパラメータの指定により、エディタ、クロスリファレンサの起動が行えます。
3. アセンブルパラメータの指定により、出力するオブジェクトプログラムの指定が行えます。

2.2 生成ファイル

ASM72では、以下の5種類のファイルを生成します。

1. オブジェクトファイル(以下HEXファイルと呼びます。)
 2. プリントファイル(以下PRNファイルと呼びます。)
 3. タグファイル(以下TAGファイルと呼びます。)
 4. シンボルファイル(以下SYMファイルと呼びます。)
 5. マップファイル(以下MAPファイルと呼びます。)
- 個々のファイルについて説明します。

2.2.1 HEX ファイル

- 機械語データを格納したファイルです。
- ASM72では用途により以下の形式のフォーマットのファイルを生成します。
 1. HEXファイル(図2.1参照)
 - コマンドパラメータ"-V"、"-W"および"-WM"ともに指定がない場合は、ピギーバック用としてオブジェクトコードを上位1ビットと下位8ビットに分割しHEXファイルに下位、上位の順に出力します。
 2. HEU、HELファイル(図2.1参照)
 - コマンドパラメータ"-V"が指定されている場合には、エバーボード用としてオブジェクトコードを上位1ビットと下位8ビットに分割し、それぞれHEU(上位)、HEL(下位)という2つのファイルに出力します。
 3. OTP²ファイル(図2.2参照)
 - コマンドパラメータ"-W"が指定されている場合には、お客様がワンタイム EPROM を使用してデバッグできるようにオブジェクトコードを上位5ビットと下位4ビットに分割しOTPファイルに下位、上位の順に出力します。
 - データの空き領域にはすべて'1'を格納します。
 - ROM容量2KのプログラムをワンタイムEPROMを使用して実行させる場合、0000₁₆番地から07FF₁₆番地までのデータを0800₁₆番地以降に、1000₁₆番地から17FF₁₆番地までのデータを1800₁₆番地以降にブロックコピーします。

-
1. タグの名称は、エラーやワーニングの場所を示す荷札(タグ)に由来しています。
 2. ワンタイム版プログラマブルROMの略

4. CAD ファイル(図 2.3 参照)

- コマンドパラメータ“-WM”が指定されている場合には、ルネサスワнтаイム EPROM 書き込み用としてオブジェクトコードを上位 1 ビットと下位 8 ビットに分割し CAD ファイルに下位、上位の順に出力します。
 - データの空き領域にはすべて‘1’を格納します。
 - ROM容量 2KのプログラムをワнтаイムEPROMを使用して実行させる場合、0000₁₆番地から 07FF₁₆番地までのデータを 0800₁₆番地以降に、1000₁₆番地から 17FF₁₆番地までのデータを 1800₁₆番地以降にブロックコピーします。
 - ROM容量 1KのプログラムをワнтаイムEPROMを使用して実行させる場合、0000₁₆番地から 03FF₁₆番地までのデータを 0800₁₆番地以降に、1000₁₆番地から 13FF₁₆番地までのデータを 1800₁₆番地以降にブロックコピーします。0400₁₆番地から 07FF₁₆番地まで、0C00₁₆番地から 0FFF₁₆番地までの下位 8 ビット領域にはすべてFF₁₆を格納します。1400₁₆番地から 17FF₁₆番地まで 1C00₁₆番地から 1FFF₁₆番地までの上位 1 ビット領域にはすべて 01₁₆を格納します。
- これらのファイルは、コマンドパラメータ“-O”が指定されている場合には、その指定されたディレクトリに、指定されない場合はカレントディレクトリに出力します。
 - ファイル属性は、それぞれのパラメータの指定により .HEU、.HEL、.OTP、.CAD および.HEX のいずれかになります。

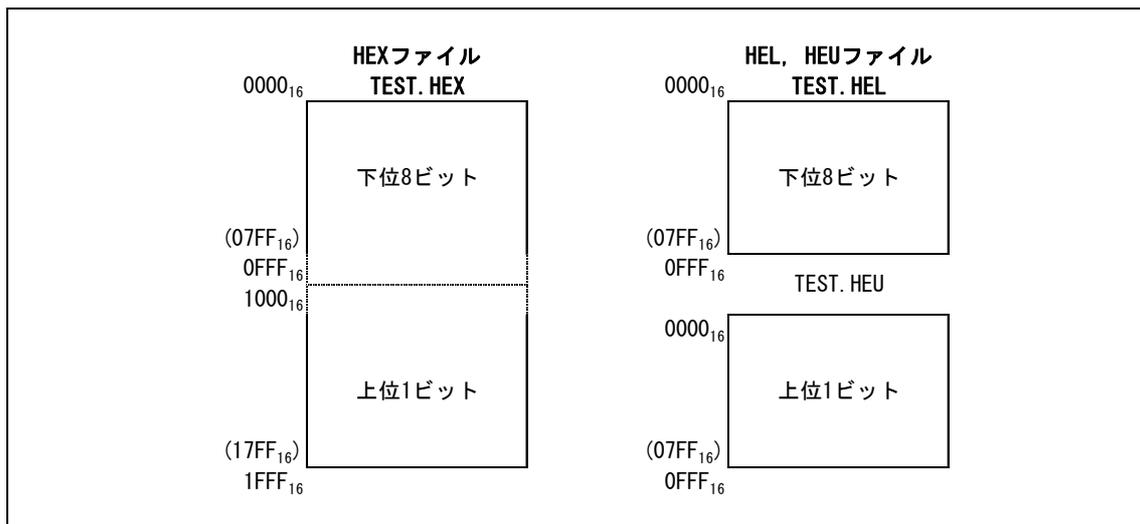


図 2.1: HEX ファイルと HEL, HEU ファイル

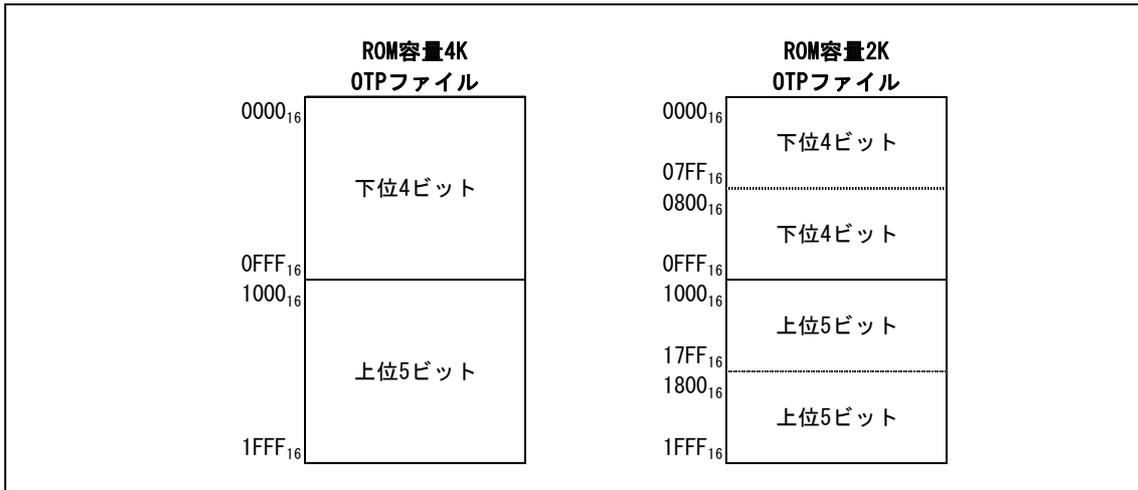


図 2.2: OTP ファイル

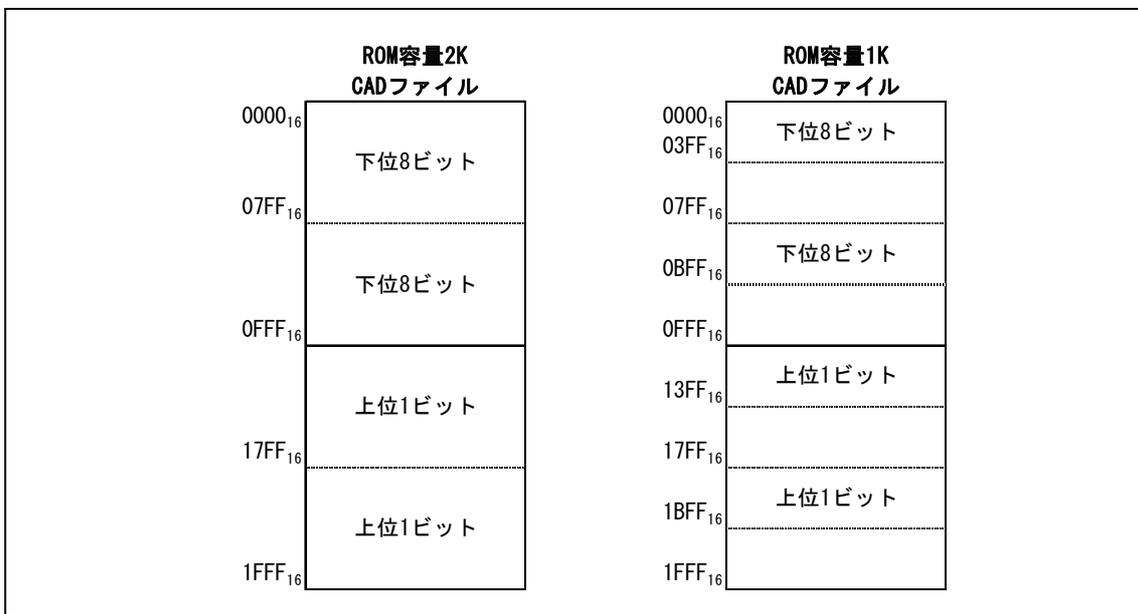


図 2.3: CAD ファイル

2.2.2 PRN ファイル

- 処理対象のソースファイルと、その配置アドレスおよび生成データを示したファイルです。
- PRN ファイルは、プリント出力してデバッグ等にお使いください。
- PRN ファイルは、コマンドパラメータ”-L”を指定した時に出力します。
- コマンドパラメータ”-O”が同時に指定されている場合には、その指定されたディレクトリに、指定されない場合はカレントディレクトリに出力します。
- ファイル属性は、.PRN です。

【 PRNファイルの構成 】

図2.5から図2.7に、PRNファイルの出力例を示します。PRNファイルは、以下の情報を示しています。

- シンボルリスト情報(図 2.5)
プログラム中のシンボルおよびラベルをファイルの先頭に表示します。
- ソースファイル情報(図 2.6)

SEQ.	LOC.	OBJ.....	DEST.	N	M*1....*2....*3....*4....*
1	2	3	4	5	6						7

1. ソースファイル行情報(SEQUENCE)
 2. ソースファイルの内容に対応するアドレス(LOCATION)
 3. ソースファイルの内容に対応するオブジェクトコード(OBJECT)
 4. 飛び先のページと番地(DESTINATION)
飛び先のページと番地が16進で表示されます。
 5. ネスティング情報(NEST)
ネスト(入れ子)箇所には'1'が表示されます。
 6. マクロ命令情報(MACRO)
マクロ命令の展開部には'+'プラスが表示されます。
 7. ソースファイルのカラム情報
5カラム毎に'*'あるいは数値を表示します。
- アセンブル結果情報(図 2.7)
エラー数、ワーニング数、全行数、コメント行数、メモリ容量を示します。
 - 疑似命令.COLによるカラム数指定が 132 文字の場合、リストのヘッダ部にアセンブルを実行した時間を次の形式で表示します。

DATE(Mon Jun 3 15:06:42 1989)

2.2.3 TAG ファイル

- アセンブル中に発生したアセンブルエラーメッセージおよびワーニングメッセージを格納したファイルです。
- TAG ファイルは、エディタによるエラー修正時に参照用としてお使いください。
- TAG ファイルは、コマンドパラメータ"-E"を指定した時に出力します。
- ファイル属性は、.TAG です。

【 TAGファイルの構成 】

図 2.8 に、TAG ファイルの出力例を示します。TAG ファイルは、以下の情報を示しています。

- エラー、またはワーニングが発生した場所のファイル名、そのファイル中の行番号、通し行番号、エラー番号、エラーメッセージを示します。

2.2.4 SYM ファイル

- PD72 でシンボリックデバッグを行なう時に必要な情報を格納したファイルです。
- SYM ファイルは、コマンドパラメータ"-S"を指定した時に出力します。
- ファイル属性は、.SYM です。
- コマンドパラメータ"-O"が同時に指定されている場合には、その指定されたドライブのディレクトリに、指定されない場合はカレントディレクトリに出力します。
- SYM ファイル内の情報は次の構成になっています。

1. ラベル情報
ラベル名とアドレス値(9 ビット)を対応づけています。
2. シンボルタイプ 1 情報
 0_{16} から $FFFF_{16}$ までの数値を表現するためのシンボルです。
3. シンボルタイプ 2 情報
X、Y レジスタの値を表現するためのシンボルです。
4. シンボルタイプ 3 情報
Z、X、Y レジスタの値を表現するためのシンボルです。
5. シンボルタイプ 4 情報
ビット位置 BIT、Z、X、Y レジスタの値を表現するためのシンボルです。データページ(DP)内のあるビットを指定する際に使用します。

図2.9に、SYMファイルの出力例を示します。

2.2.5 MAP ファイル

- ページ毎の使用メモリ情報を格納したファイルです。
- MAP ファイルは、コマンドパラメータ"-A"を指定した時に出力します。
- ファイル属性は、.MAP です。

```

-M50727
;*****
;*
;*      M50727  TEST  PROGRAM
;*
;*****
                .COL  80
                .ORG  0,0
                .TTL  727TEST PROG.          ;SET  TITLE
PORTD1 .EQU  0,0                               ;SET  LABEL
PORTD2 .EQU  0,1
PORTD3 .EQU  0,2
PORTK  .EQU  0,3
;*****
;*      MAIN  ROUTIN
;*****
MAIN:          DI                               ;STOP  INT
                LZ   0                          ;RAM  CLEAR
                LXY  0,0
WAIT:          LA   0
                XAMI 0
                B    WAIT
RAM_X0:        LA   0
                XAMI 0
                B    RAM_X0
                LXY  10
RAM_X1:        LA   0
                XAMI 0
                B    RAM_X1
                LXY  2,0
RAM_X2:        LA   0
                XAMI 0
                B    RAM_X2
                LXY  3,0
                :
                :

```

図2.4：ソースファイル例

M50727 ASSEMBLER SYMBOL TABLES										P.001
ATOC	029C	A_D	01B6	COUNT	0100	D0_SET	0262	D10	0253	
D10_SET	0280	D1_SET	0265	D2	0206	D2_SET	0268	D3	020F	
D3_SET	026B	D4	0218	D4_SET	026E	D5	0221	D5_SET	0271	
D6	022A	D6_SET	0274	D7	0233	D7_SET	0277	D8_SET	027A	
D9	024A	D9_SET	027D	DH_OUT	025C	DL_OUT	023C	D_OFF1	0083	
D_OFF1.1	01A7	D_OFF1.2	01B3	D_OFF2	0093	D_SET1	01C9	D_SET2	01D7	
ERR	0191	HENKAN	02B0	IN1	01EE	IN2	01E8	IN3	01E0	
LOOP	0027	LOOP1	01BC	LOT	001E	LOTS	0199	MAIN	0000	
OUT	01C3	PORTD1	0000	PORTD2	0001	PORTD3	0002	PORTK	0003	
RAM_X0	0006	RAM_X1	000A	RAM_X2	000E	RAM_X3	0012	SERIAL	0283	
SOUSIN1	028B	SOUSIN2	0295	SUB0	0180	T_1	0109	T_1CNT	010B	
WAIT	0003									

図2.5 : PRNファイル例(シンボルおよびラベルリスト)

```

* 720 SERIES ASSEMBLER V.1.00.00C *                               (00 page) P.002
SEQ. LOC. OBJ..... DEST. N M....*....1....*....2....*....3....*....4....*...

  1          0 ;M50727
  2          0 ;*****
  3          0 ;*
  4          0 ;*      M50727  TEST  PROGRAM
  5          0 ;*
  6          0 ;*****
  7          0          .COL      80
  8          0          .ORG      0,0
  9          0          .TTL      727TEST PROG.
test.ASM 9 ( TOTAL LINE 9 ) Error 12: No " at the top of comment "727TEST"

 10 0000          0 PORTD1 .EQU  0,0
 11 0001          0 PORTD2 .EQU  0,1
 12 0002          0 PORTD3 .EQU  0,2
 13 0003          0 PORTK  .EQU  0,3
 14          0 ;*****
 15          0 ;*      MAIN  ROUTIN
 16          0 ;*****
 17 0000 004          0 MAIN:      DI
 18 0001 048          0          LZ      0
 19 0002 0C0          0          LXY     0,0
 20 0003 0B0          0 WAIT:     LA      0
 21 0004 068          0          XAM     0
 22 0005 183          00/03 0      B      WAIT
 23 0006 0B0          0 RAM_X0:   LA      0
 24 0007 068          0          XAMI    0
 25 0008 186          00/06 0      B      RAM_X0
 26 0009 000          0          LXY     10
test.ASM 26 ( TOTAL LINE 26 ) Error 5: Improper operand type "10"
 27 000A 0B0          0 RAM_X1:   LA      0
 28 000B 068          0          XAMI    0
 29 000C 18A          00/0A 0      B      RAM_X1
 30 000D 0E0          0          LXY     2,0
 31 000E 0B0          0 RAM_X2:   LA      0
 32 000F 068          0          XAMI    0
 33 0010 18E          00/0E 0      B      RAM_X2
 34 0011 0F0          0          LXY     3,0
 35 0012 0B0          0 RAM_X3:   LA      0
 36 0013 068          0          XAMI    0
 37 0014 192          00/12 0      B      RAM_X3
 38 0015 0C0          0          LXY     0,0

```

図2.6 : PRNファイル例(前半部分)

```

* 720 SERIES ASSEMBLER V.1.00.00C *                               (05 page) P.013
SEQ.  LOC.  OBJ..... DEST. N M....*....1....*....2....*....3....*....4....*....
413  02BF  0B0          0          LA      0
414  02C0  082          0          TCA
415  02C1  044          0          RT
416          0          .END

ERROR   COUNT 0003
WARNING COUNT 0000
SOURCE  LINE  0416 LINES
TOTAL   LINE  0416 LINES
COMMENT LINE  0043 LINES
OBJECT  SIZE  0395 BYTES

```

図2.7 : PRNファイル例(後半部分)

```

test.ASM 9 ( TOTAL LINE 9 ) Error 12: No " at the top of comment "727TEST"
test.ASM 26 ( TOTAL LINE 26 ) Error 5: Improper operand type "10"
test.ASM 53 ( TOTAL LINE 53 ) Error 19: Reference to undefined label "F"

```

図2.8 : TAGファイル例

```

#MELPS720
#EQU          ← シンボルタイプ1情報
FALSE 0000   TRUE 0001   ZERO 0000
#SYMBOL      ← シンボルタイプ2,3,4情報
BANK0 *000   BANK1 *100   CNT1 **00   CNT2 **01   EVF 0002
OVF 2002    TSF 3002
#LABEL      ← ラベル情報
RAMCL 0125   RAMCL0 0121   RAMCL1 0122   RAMCL2 0123   RAMCL3 0124

```

図2.9 : SYMファイル例

第3章

ソースプログラムの記述方法

3.1 ソースプログラムの構成

ソースプログラムは、行を単位として構成しています。以下に行記述の規則を示します。

- 各行は1行ごとに完結しています。従って、1命令を2行以上にわたって記述することはできません。
- 1行に記述可能な文字数は最大100文字です。ASM72は101文字以上は無視します。
- 行は内容別に以下の6種類の行に分類できます。

1. MCU名指定行
使用する720シリーズのMCU名を指定した行です。
2. アセンブリ言語命令
720シリーズのアセンブリ言語命令を記述した行です。この行は対応する機械語データを生成します。
3. ビットシンボル加減算行
ビットシンボルに対する加減算を記述した行です。
4. 疑似命令行
720シリーズの疑似命令を記述した行です。
5. マクロ命令行
720シリーズのマクロ命令を記述した行です。この行は対応するアセンブリ言語命令へ展開されます。
6. コメント行
この行はASM72で処理しませんのでユーザが自由に使用できます。

3.2 行の構成

本節では、それぞれの行の構成について説明します。なお、以下に表記上用いている記号の説明および規則を示します。

1. △、▲はスペースまたはタブコードを示しています。△は必須、▲は省略可能です。
2. ラベルを記述する場合‘:’(コロン)は必須ではありません。
3. ラベルと各命令との間にスペース、またはタブコードが必要になります。

3.2.1 MCU名指定行

MCU名指定行の構成を以下に示します。

```
-MCU名 △ ;コメント <RET>
```

1. MCU名欄
 - MCU名を記述します。
 - MCU名の前に必ず‘-’(ハイフン)を記述して下さい。
 - MCU名は、アセンブル時の“-M”オプションでも指定可能です。

2. コメント欄

- この欄は、ASM72 で処理しませんのでユーザが自由に使用できます。
- コメント欄の1文字目は、';'(セミコロン)で始めます。
- ステートメント全てをコメントとする場合は、';'(セミコロン)を行の先頭に置くことにより、これに続くこの行の内容をコメントとみなします。

3.2.2 アセンブリ言語命令行

アセンブリ言語命令行の構成を以下に示します。

ラベル △ オペコード △ オペランド △ ;コメント <RET>

1. ラベル欄

- 行を他の場所から参照するためのラベルを記述します。
- ラベル欄には15文字までの英数字および'_'(アンダーライン)、'.'(ピリオド)、'?'(クエスションマーク)の特種文字から構成される文字列を記述します。
- ラベルの1文字目は、英文字または'_'(アンダーライン)、'.'(ピリオド)、'?'(クエスションマーク)以外の文字、数字は使用できません。
- ラベル内の英大文字/小文字は区別されます。したがって、BIG と big は異なったラベルとして判断されます。

2. オペコード欄

- 720 シリーズのアセンブリ言語ニーモニック(以下オペコードと呼びます。)を記述します。
- オペコードは、英大文字/小文字を区別しません。従って NOP、nop いずれも有効です。

3. オペランド欄

- オペコードの処理対象を記述します。
- オペランドにデータが2つ以上ある場合、データ間を','(カンマ)で区切ってください。
- カンマの両側には、スペースまたはタブコードが記述できます。

4. コメント欄

- この欄は、ASM72 で処理しませんのでユーザが自由に使用できます。
- コメント欄の1文字目は、';'(セミコロン)で始めます。
- ステートメント全てをコメントとする場合は、';'(セミコロン)を行の先頭に置くことにより、これに続くこの行の内容をコメントとみなします。

3.2.3 ビットシンボル加減算行

ビットシンボル加減算行の構成を以下に示します。

命令 △ ビットシンボル △ 演算子 △ ビットシンボル
命令 △ ビットシンボル △ 演算子 △ シンボル/数値 .シンボル/数値 ...

1. 命令欄

- ビットシンボルに対する演算が可能なマクロ命令またはニーモニックを記述します。
マクロ命令: .clb .lxy .seb .szxyb
ニーモニック: lz lxy rb sb szb

2. 演算子前のビットシンボル欄

- ビットシンボルを記述します。

3. 演算子欄

- 演算子は加算(+)と減算(-)が記述できます。
- 1行に記述可能な演算子は1個です。

4. 演算子後のビットシンボル欄
 - XY シンボル、ZXY シンボル、ビットシンボルのいずれかを記述します。
 - 指定可能なビットシンボルは1個です。
5. 演算子後のシンボル/数値欄
 - シンボル/数値を記述します。
 - 指定可能なシンボル/数値は4個までです。
 - 指定が1個の場合、Yレジスタとして扱います。
 - 指定が2個の場合、1つ目をXレジスタ、2つ目をYレジスタとして扱います。
 - 指定が3個の場合、1つ目をZレジスタ、2つ目をXレジスタ、3つ目をYレジスタとして扱います。
 - 指定が4個の場合、1つ目をビット位置、2つ目をZレジスタ、3つ目をXレジスタ、4つ目をYレジスタとして扱います。
 - 数値と数値の間には','(カンマ)を記述して下さい。
 - カンマとカンマの間の数値は省略可能です。
 - 以下の場合にはエラーとなります。
 - 加算の結果が各レジスタ値、ビット位置を超える場合
 - 減算の結果がマイナス値の場合
 - 演算子の後にオペランドがない場合
 - 演算子後のビットシンボル欄でビットシンボルが1個より多く指定されている場合
 - 演算子後のシンボル/数値欄でシンボル/数値が4個より多く指定されている場合

3.2.4 疑似命令行

疑似命令行の構成を以下に示します。この行の詳細については、第4章、および付録Bを参照してください。

```
シンボルタイプ1-4 △ EQU △ オペランド △ ;コメント <RET>
ラベル △ 疑似命令 △ オペランド △ ;コメント <RET>
```

1. シンボルタイプ1~4欄
 - 疑似命令.EQUにより値を割り当てるシンボル、ビットシンボルを記述します。
2. ラベル欄
 - アセンブリ言語命令行の節を参照ください。
3. 疑似命令欄
 - 720シリーズの疑似命令を記述します。
 - 疑似命令は、英大文字/小文字を区別しません。従って.END、.endいずれも有効です。
4. オペランド欄
 - 疑似命令の処理対象を記述します。
 - アセンブリ言語命令行の節を参照ください。
5. コメント欄
 - アセンブリ言語命令行の節を参照ください。

3.2.5 マクロ命令行

マクロ命令行の構成を以下に示します。この行の詳細については、第5章、および付録Cを参照してください。

```
ラベル: △ マクロ命令 △ オペランド △ ;コメント <RET>
```

1. ラベル欄
 - マクロ命令行を他の場所から参照するためのラベル(名前)を記述します。
2. マクロ命令欄
 - 720 シリーズのマクロ命令を記述します。マクロ命令は、英大文字/小文字を区別しません。従って、.CLB、.clb いずれも有効です。
3. オペランド欄
 - マクロ命令の処理対象を記述します。
 - アセンブリ言語命令行の節を参照ください。
4. コメント欄
 - アセンブリ言語命令行の節を参照ください。

3.2.6 コメント行

コメント行は、行の最初の文字を';'(セミコロン)で始めてください。コメント行の構成を以下に示します。

▲ ;コメント <RET>

3.3 欄の記述方法

本節では、各命令行において記述形式が共通な欄についてのみ説明します。記述形式が異なる欄については、それぞれ第4章、第5章を参照してください。

3.3.1 シンボルタイプ 1~4/ラベル欄

この欄は各命令行とも記述形式は同じです。ただしASM72は、シンボルタイプ 1~4、ラベル¹を区別して管理します。以下に記述上の規則を示します。

1. 名前には、英数字、特殊文字'_'(アンダーライン)、'.'(ピリオド)、および'?'(クエッションマーク)が使用できます。ただし、1文字目に使用できる文字は英字と特殊文字だけです。また、文字数は最大15文字まで使用できます。
2. 大文字/小文字を区別します。従ってBIGとbigは異なった名前として判断します。

ラベルを記述する場合、名前の直後に':'(コロン)をつけることができます。シンボルとの区別を容易にするため、またエディタでのラベル検索を効率的に行うため、':'を付加した記述を推奨します。

3.3.2 コメント欄

ユーザーの任意の情報を記述することができます。以下に記述形式を示します。

1. コメントの先頭には必ず';'(セミコロン)を付けて記述してください。
2. コメント欄にはどのような文字も記述できます。

3.4 オペランドデータ形式

オペランドには、以下の4種類のデータ形式が記述できます。

1. 数値定数
2. 文字定数
3. 記号定数
4. 式

1. 疑似命令.EQU、または"-D"パラメータにより定義したものをシンボルタイプ1~4、またはこれ以外で定義したものをラベルとして扱います。

3.4.1 数値定数

- 数値の種類を表わす記号と数値との間には、スペースまたはタブをいれることはできません。
例) `.DW $ 64` ⇒エラーとなります。
- 数値定数には、2進、8進、10進、16進数が使用できます。
- 2進数⇒2進数で構成される場合は、先頭に'%'を付けるか、最後に'B'、または'b'を付けて記述してください。
例) `.DW %100110`
例) `.DW 100110B`
- 8進数⇒8進数で構成される場合は、先頭に'@'を付けるか、最後に'O' 'o'、または'Q' 'q'を付けて記述してください。
例) `.DW @70`
例) `.DW 70O`
例) `.DW 70Q`
- 10進数⇒10進数で構成される場合は、特に何も指定しません。23、256のように整数のみで記述してください。
例) `.DW 100`
- 16進数⇒16進数で構成される場合は、先頭に'\$'を付けるか、最後に'H'、または'h'を付けて記述してください。先頭が英文字(A~F)で始まる場合は先頭に0を付加してください。
例) `.DW $64`
例) `.DW 64H`
例) `.DW 0ABH`

3.4.2 文字定数

- 文字には、ASCIIコードで定義されている1~16までの文字が記述できます。
- 文字定数は、'(シングルクォート)で囲んで記述してください。各文字は、7ビットASCIIコード(最上位ビットは0)に対応しています。
例) `.DW 'A'` ⇒ 41Hを設定します。

3.4.3 記号定数

- 記号はラベル、シンボルタイプ 1~4 の 5 種類があります。
 - ラベルは ROM アドレスをもちます。
 - シンボルタイプ 1 は絶対値をもちます。
 - シンボルタイプ 2 はシンボル名と X、Y レジスタの値をもちます。
 - シンボルタイプ 3 はシンボル名と Z、X、Y レジスタの値をもちます。
 - シンボルタイプ 4 はビットシンボルで、ビット位置 BIT、Z、X、Y レジスタの値をもちます。ビットシンボルは、割り当てられたビットの値と、そのビットの属するアドレスをもちます。
- 例) `B MAIN` ⇒ MAIN ヘブランチします。

3.4.4 式

- 式は、数値定数、文字定数、記号定数、演算子の組み合わせで構成します。演算子と各項の間には、必要に応じてスペースまたはタブを入れることができます。
例) `TBL + 1`
- 式は左から右に計算します。また演算の優先順位を示す()が使用できます。
例) `2+6/2` ⇒結果は 4 になります。
例) `2+(6/2)` ⇒結果は 5 になります。

3.5 特殊文字

表3.1に、オペランドデータ記述で使用可能な特殊文字の一覧表を示します。

表3.1特殊文字一覧表

文字	文字の名称	文字	文字の名称
	スペース	?	クエスチョンマーク
+	プラス	#	シャープ
-	マイナス	(左カッコ
*	アスタリスク)	右カッコ
/	スラッシュ	\	逆スラッシュ
\$	ドル	!	エクスペレッション
@	アットマーク	<	不等号(より小)
%	パーセント	>	不等号(より大)
,	コンマ	&	アンパサンド
'	シングルクォーテーション	:	コロン
^	ハイハット	;	セミコロン
.	ピリオド		水平タブ

注意：(逆スラッシュ)をもたないシステムでは、¥(マーク)が代用できます。

3.6 演算子

表3.2に、オペランドデータ記述で使用可能な演算子の一覧表を示します。

表3.2演算子一覧表

分類	演算子 ¹	内容
単項	!	1の補数をとる
	<	ラベル、またはシンボルの上位8ビットを切り出す
	>	ラベル、またはシンボルの下位8ビットを切り出す
	#	ラベル、またはシンボルのページ(上位1ビット)を切り出す
2項演算子	+	加算
	-	減算
	*	乗算
	/	除算
	&	ビットごとのAND
		ビットごとのOR
	^	ビットごとの排他的OR
	()	演算の優先順位の指定

注意事項

1. ビットシンボルに対する演算はできません。

第4章

疑似命令

4.1 疑似命令の機能

疑似命令は、命令が目的とする機械語データを生成するようにASM72に対する指示¹を行います。ASM72は、12個の疑似命令を用意していますが、これらは機能的に以下の3つのグループに分類できます。

1. アセンブル制御

- 疑似命令自身はデータを生成しませんが、アセンブル処理の流れを制御します。
- アドレスの更新には影響しません。
- このグループは以下の3個の疑似命令を含みます。

.END	プログラム終了宣言
.IF (.ELSE) .ENDIF	条件付きアセンブル
.INCLUDE	ファイル読み込み

2. アドレス制御

- データ設定疑似命令は定数データの生成を行います。
- アドレスの更新を行います。
- このグループは以下の3個の疑似命令を含みます。

.EQU	同義定義
.ORG	アドレス指定
.DW	データ設定

3. リスト制御

- PRNファイル出力に関する制御を行います。
- このグループは以下の6個の疑似命令を含みます。

.COL	リスト形式カラム数指定
.LINE	リスト形式行数指定
.LIST	リスト出力指定
.NLIST	リスト出力抑止指定
.PAGE	改ページ指定
.TTL	タイトル指定

以下に各グループ別に疑似命令の機能を説明します。

1. 疑似命令の名称は、ASM72に対する指示を行うものを”宣言”、出力ファイルに影響するものを”指定”と呼んでいます。

4.2 アセンブル制御

4.2.1 アセンブル終了宣言

.END

- ソースプログラムの終りを宣言します。
- ASM72は、この行以降の内容を処理しません。

4.2.2 条件付きアセンブル

.IF ~ (.ELSE) ~ .ENDIF

- シンボル値の内容により、アセンブルを行う場所を指定します。
- 複数仕様に対応するプログラムを1つのソースプログラムで管理する場合、テストルーチンのアセンブルを制御する場合等に使用できます。

4.2.3 ファイル読み込み

.INCLUDE

- この命令を記述した場所に、他のファイル内容を読み込みます。
- 大きなソースプログラムを分割して編集する場合に使用できます。

4.3 アドレス制御

4.3.1 同義定義

.EQU

- シンボルに絶対値を定義します。
- シンボルにファイル、桁の値を定義します。
- シンボルにファイル群、ファイル、桁の値を定義します。
- シンボルにビット値、ファイル群、ファイル、桁の値を定義します。

4.3.2 アドレス宣言

.ORG

- 以降の行のアドレスを宣言します。

4.3.3 データ設定

.DW

- オペランドで指定したデータをROM領域に生成します。

4.4 リスト制御

4.4.1 改ページおよびタイトル指定

.PAGE、.TTL

- リストの改ページおよびタイトルを指定します。

4.4.2 リスト形式指定

.COL、.LINE

- リストのカラム数、行数を指定します。
- これらの疑似命令は、ソースファイル中に1回だけ記述できます。

4.4.3 リスト出力/抑止指定

.LIST、.NLIST

- PRNファイルへのリスト出力制御を行います。
- プログラムの一部をデバッグする場合等で、リストの一部のみが必要な時に使用してください。

第5章

マクロ命令

5.1 マクロ命令の機能

720 シリーズのアセンブリ言語の複数の命令を、マクロ命令を使用してオペコードと同様に記述することができます。ASM72 は、4 個の疑似命令を用意していますが、これらは機能的に以下の2つのグループに分類できます。

1. ビットマクロ命令

- オペランドで指定したビットシンボルの示すビットを操作します。
- このグループは以下の3個のマクロ命令を含みます。

.CLB 指定ビットのクリア
.SEB 指定ビットの設定
.SZXYB 指定ビットの状態によりプログラムの流れを変更

2. レジスタマクロ命令

- オペランドで指定したビットシンボルの示すZ, X, Yの値をそれぞれのレジスタへ設定します。

.LZXY 指定Z, X, Yレジスタの設定

以下に各グループ別にマクロ命令の機能を説明します。

5.2 ビットマクロ命令

5.2.1 指定ビットのクリア

.CLB

- オペランドで指定されたビットシンボルの示すビットをクリアします。

5.2.2 指定ビットの設定

.SEB

- オペランドで指定されたビットシンボルの示すビットを設定します。

5.2.3 指定ビットの状態によりプログラムの流れを変更

.SZXYB

- オペランドで指定されたビットシンボルの示すビットをテストしゼロであれば流れをかえます。

5.3 レジスタマクロ命令

5.3.1 Z, X, Y の値設定

.LZXY

- オペランドで指定されたZ, X, Yの値をそれぞれのレジスタへ設定します。

第6章

操作方法

6.1 起動方法

ASM72 を実行するためには、以下の情報(入力パラメータ)を入力する必要があります。

- ソースファイル名(必須項目)
- コマンドパラメータ

ASM72では、これらの情報をDOSのコマンド行から指定します。

6.2節で入力パラメータについて説明し、6.3節で例を示しながらコマンド行入力方法を説明します。

6.2 入力パラメータ

6.2.1 ソースファイル名

- アセンブル対象のソースファイル名を指定します。指定できる個数は1個です。
- ファイル属性(.ASM)を省略した場合、既定値として属性.ASMを選びます。
- ファイル名をフルネームで指定することにより、.ASM以外の属性(例.SRC)のファイルもアセンブル可能です。
- ファイル名にはディレクトリパス指定が可能です。ファイル名のみを指定した場合は、カレントドライブのカレントディレクトリ中のファイル进行处理します。

例) A>ASM72 C:¥WORK¥TEST<RET>

6.2.2 コマンドパラメータ

- コマンドパラメータは大文字/小文字どちらでも有効です。
- 各パラメータは同時に複数指定することができます。この場合は、各パラメータをスペースで区切って入力してください。

表6.1～表6.2に、コマンドパラメータの内容を示します。

表6.1:コマンドパラメータ一覧表(その1)

コマンドパラメータ	内容
-A	マップファイルを作成します。 例) A>ASM72 FILENAME -A<RET>
-C	シンボルファイルにソースラインデバッグ情報を出力します。 例) A>ASM72 FILENAME -C -S<RET>
-D	シンボルに数値を設定します。コマンドの機能は、疑似命令.EQUと同等です。指定の書式は、次のようになります。(複数のシンボルを同時に定義する場合、';'で区切ってください。) -Dシンボル=数値[:シンボル=数値....:シンボル=数値] 例) A>ASM72 FILENAME -DSYMBOL1=10:SYMBOL2=20<RET>

表6.2: コマンドパラメータ一覧表(その2)

コマンドパラメータ	内容
-E	<p>TAGファイルの生成およびエディタの起動を行います。エディタのプログラム名指定の書式は、次のようになります。</p> <p>-E[エディタ名]</p> <p>例) A>ASM72 FILENAME -EM1<RET></p> <p>[]の部分は省略可能です。省略した場合、TAGファイルの生成のみを行います。エディタ名が指定された場合はアセンブル終了後、TAGファイルを引き数としてエディタを起動します。ただし、エディタ名を指定してもエラーが発生しなかった場合は、エディタの起動は行いません。</p>
-L	<p>PRNファイルを生成します。この指定がない場合PRNファイルは生成しません。</p>
-M	<p>使用MCU名が指定できます。この指定がない場合、ソースの1行目に記述してあるMCU指定を参照します。</p> <p>例) A>ASM72 FILENAME -M50720<RET></p>
-O	<p>生成ファイルの出力先パスを指定します。パスにはディレクトリまたはドライブ名が指定できます。この指定がない場合、ソースファイルと同じパスに出力します。指定の書式は次のようになります。</p> <p>-Oパス名</p> <p>例) A>ASM72 FILENAME -OB:¥USR<RET></p>
-P	<p>データファイル(MXXXXX.DAT)が格納されているディレクトリおよびドライブを指定します。</p> <p>-Pドライブ名</p> <p>例) A>ASM72 FILENAME -PB:¥USR<RET></p>
-R	<p>ビットマクロ疑似命令の展開部の記述を、PRNファイル上に出力します。</p> <p>例) A>ASM72 FILENAME -R<RET></p>
-S	<p>PD72用のシンボルフイルを出力します。</p> <p>例) A>ASM72 FILENAME -S<RET></p>
-V	<p>エバボード用にHEU(HEXファイル上位1ビット)、HEL(HEXファイル下位8ビット)を出力します。</p> <p>例) A>ASM72 FILENAME -V<RET></p>
-W	<p>EPROM版用にOTPファイルを出力します。</p> <p>例) A>ASM72 FILENAME -W<RET></p>
-WM	<p>EPROM版用に上位1ビット、下位8ビットのCADファイルを出力します。</p> <p>例) A>ASM72 FILENAME -WM<RET></p>
-X	<p>アセンブル終了後、クロスリファレンサCRF72を起動します。¹</p> <p>例) A>ASM72 FILENAME -X<RET></p>

注意事項

1. カレントディレクトリまたはコマンドパス中にCRF72が存在しない場合は、システムエラーになります。
2. コマンドパラメータ-Vと-W、-WM を、組み合わせて同時に指定することはできません。

6.3 入力方法

ASM72 は、DOS のプロンプト状態でコマンド行を入力することにより起動します。以下に、起動コマンドの入力例を示します。

```
A> ASM72 FILENAME -L -S <Enter>
```

1 2 3 4 5 6

1. DOSのプロンプト
2. ASM72
3. アセンブル対象のソースファイル名
4. コマンドパラメータ-Lを指定することによりPRNファイルを生成します。
5. コマンドパラメータ-Sを指定することによりSYMファイルを生成します。
6. リターンキー

コマンド行入力に誤りを検出すると、図 6.1 のようにヘルプ画面を表示しアセンブルを中止します。コマンド行入力が正常に行われるとアセンブルを開始します。

```
C:¥>ASM72<Enter>
720 SERIES ASSEMBLER V.1.20.02C
COPYRIGHT(C) 1989 (1989-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

Usage: ASM72 <filename> [Options...]

-A : make memory Area information.( output MAP file )
-C : output source line information.( output SYM file )
-D : define symbol ( use -Ds1=1:s2=2 )
-E : make tag file and start editor ( use -E or -Editor name )
-L : make list file
-M : define CPU name ( use -M50720 )
-O : select drive and directory for output ( use -Oa:¥work )
-P : select directory(drive) of mxxxxx.dat file.( use -P¥work )
-R : output bit macro expansion
-S : make symbol file for symbolic debugger
-V : make file for eva.chip
-W : output hex data for EPROM version ( lower 4bit, upper 5bit)
-WM : output hex data for EPROM version ( lower 8bit, upper 1bit)
-X : execute crf72
```

図6.1:コマンドエラー時のヘルプ画面

アセンブルが終了するとエラー数、ワーニング数、全行数、コメント行数、メモリ容量を画面に出力します。アセンブルが正常に終了したときの画面表示例を図 6.2 に示します。

```
D:¥>ASM72 TEST<Enter>
720 SERIES ASSEMBLER V.1.20.02C
COPYRIGHT(C) 1989 (1989-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED
now processing pass 1
----*----*
now processing pass 2
----*----*
ERROR   COUNT  0000
WARNING COUNT  0000
SOURCE  LINE   1028 LINES
TOTAL   LINE   1028 LINES
COMMENT LINE   0000 LINES
OBJECT  SIZE   1024 BYTES
```

図6.2:正常終了時の画面表示

6.4 エラー

6.4.1 エラーの種類

ASM72 実行時に発生するエラーは、以下の原因によるものがあります。

1. DOSに関するエラー

ディスクやメモリ容量の不足等、ASM72を実行するDOS環境に関わるエラーです。
付録Aエラーメッセージ一覧表を参照の上、DOSのコマンドにより対応してください。

2. ASM72のコマンド行入力に関するエラー

ASM72起動時のコマンド行入力に関わるエラーです。本章の内容を確認の上コマンドを再入力してください。

3. アセンブル対象のソースファイルの内容に関するエラー

ラベルの二重定義、未定義シンボルの参照等のソースファイルの内容に関わるエラーです。
該当箇所のソースファイル内容を修正して再度アセンブルを行なってください。アセンブルエラーを検出した場合は、正しいHEXファイルは生成されません。

ASM72 はエラーおよびワーニングを検出すると、図 6.3 の形式でエラー内容(ファイル名、ファイル中の行番号、通し行番号、エラー番号およびエラーメッセージ)を画面と PRN ファイルに出力します。付録 A のエラー番号順のエラー一覧表を参照の上対応してください。

```

D:¥>ASM72 TEST<Enter>
720 SERIES ASSEMBLER V.1.20.02C
COPYRIGHT(C) 1989 (1989-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED
now processing pass 1
----*----*
now processing pass 2
----*----*
1028 0401 102      08/02 0      BM subl
TERST.ASM 1028 ( TOTAL LINE 1028 ) Error 21: Value is out of range "subl"

ERROR   COUNT 0001
WARNING COUNT 0000
SOURCE  LINE  1031 LINES
TOTAL   LINE  1031 LINES
COMMENT LINE  0000 LINES
OBJECT  SIZE  1026 BYTES

```

図6.3エラー表示例

6.4.2 DOS への戻り値

DOS のバッチファイル等に実行コマンドを記述する場合、実行結果に応じて処理の内容を変えたい場合があります。ASM72 では、実行結果を表 6.3 の 4 つのエラーレベルに分けて DOS に返すようにしています。エラーレベルの利用方法については、市販の DOS の説明書を参照してください。

表6.3エラーレベル一覧表

エラーレベル	実行結果の内容
0	正常終了
1	アセンブル対象のソースファイルの内容に関するエラー
2	ASM72のコマンド入力に関するエラー
3	DOSに関するエラー

付録A

エラーメッセージ一覧表

A.1システムエラー一覧表

アセンブル中にシステムエラーを検出すると、エラーメッセージを画面に表示しアセンブルを中止します。表 A.1 にシステムエラー一覧表を示します。

注意事項

1. ASM72 で扱えるシンボル及びラベルの総数は、ASM72 を実行するシステムでの使用可能メモリ容量に依存しています。

表 A.1:システムエラー一覧表

エラーメッセージ	エラー内容と対策
Usage: ASM72 <filename> [Options...]	コマンド入力が誤っています。 ⇒ヘルプ画面を参照して、コマンドを再入力してください。
Can't open xxx	該当ファイルが見つかりません。 ⇒ソースファイル名を確認して、再入力してください。
Can't create xxx	該当ファイルが生成できません。 ⇒-O パラメータの指定を確認して、再入力してください。
Out of disk space	ファイルを出力するためのディスク容量が不足しています。 ⇒ディスク上に空き領域を作ってください。
Out of heap space	アセンブラが動作するために必要なメモリが不足しています。 ¹ ⇒シンボル又はラベルの数を減らしてください。
Can't find crf72.exe	CRF72 が見つかりません。 ⇒CRF72 をカレントディレクトリ又は DOS のコマンドパス内のディレクトリにコピーしてください。
Can't find command.com for execute xxx	-E オプションで指定されたエディタを起動するために必要なファイル COMMAND.COM が見つかりません。 ⇒DOS のコマンドパス指定を確認してください。
Multiple define of cpu name	コマンド入力とソースファイルの先頭で CPU 名の指定を行なっています。 ⇒どちらかの指定をやめてください。
M34xxx.DAT not found	指定のディレクトリに指定の CPU に対応するデータファイルがありません。 ⇒指定のデータファイル有無を確認してください。

A.2アセンブルエラー一覧表

アセンブルエラーを検出すると、エラーメッセージを画面と PRN ファイルに出力します。表 A.2 にアセンブルエラーとその内容を示します。

表 A.2:アセンブルエラー一覧表

エラー番号	エラーメッセージ	エラー内容と対策
1	Already had same statement	ソースファイル中に 1 回しか使用できない疑似命令を、2 回以上使用しています。 例) <code>.LINE 60</code> : <code>.LINE 80</code> ⇒宣言を 1 回にしてください。
2	Reference to backward label or symbol	疑似命令が後方のラベル又はシンボルを参照しています。 例) <code>.ORG TOP</code> <code>TOP:</code> ⇒ラベル又はシンボルの定義を参照以前に行ってください。
3	Division by 0	数式中に 0 による割り算を含んでいます。 ⇒数式の記述を確認してください。
4	Illegal operand	オペランドに使用できない文字を含んでいます。 例) <code>LDA #&10</code> ⇒オペランドの記述を確認してください。
5	Improper operand type	ニーモニックとオペランドの組み合わせが誤っています。 例) <code>ASL work,Y</code> ⇒命令記述形式を確認してください。
6	Invalid label definition	ラベル定義ができない場所でラベル定義を行っています。 例 1) <code>LABEL1: .LINE 60</code> ⇒ラベルを削除してください。 例 2) <code>LABEL2: .EQU 100</code> ⇒ラベルをシンボルに変更してください。
7	Out of maximum program size	アドレスが ROM 容量を超えています。 例) <code>.ORG 0FFFFH</code> <code>.WORD 1,2,3,4,5,6,7,8,9</code> ⇒アドレスが範囲内になるようにプログラムを変更してください。
8	Label is multiple defined	同一のラベル又はシンボルが 2 回以上定義されています。 例) <code>MAIN: NOP</code> <code>MAIN: NOP</code> ⇒ラベル名又はシンボル名を確認してください。

エラー番号	エラーメッセージ	エラー内容と対策
9	Nesting error	疑似命令 .IF 又は .INCLUDE が入れ子になっています。 例) <pre>.IF DATA1 .IF DATA2 : .ENDIF .ELSE : .ENDIF</pre> ⇒入れ子にならないようにプログラムを変更してください。
10	No .END statement	ソースファイル内に .END 文がありません。 ⇒プログラムの最後に .END 文を記述してください。
11	No label definition	シンボルが記述されていません。 例) <pre>.EQU 60</pre> ⇒シンボルを記述してください。
12	No ';' at the top of comment	コメント欄の先頭に ';' (セミコロン)がありません。 例) <pre>LDA #CNT counter set</pre> ⇒コメント欄の先頭に ';' を付加してください。
13	Not in conditional block	対応する .IF 文がないのに、.ELSE 又は .ENDIF が記述されています。(対応する .IF 文がエラーになっている場合もこのエラーが発生します。) 例) <pre>.IF DATA1 : .ENDIF : .ELSE : .ENDIF</pre> ⇒.IF 文を確認してください。
14	Operand is expected	命令に必要なオペランドが不足しています。 例) <pre>.BYTE</pre> ⇒オペランドの記述を確認してください。
15	Questionable syntax	ニーモニックのスペルに誤りがあります。 例) <pre>ADD #DATA</pre> ⇒ニーモニックのスペルを確認してください。
16	Reference to multi defined label	重複定義されているラベル又はシンボルを参照しています。 例) <pre>MAIN: NOP MAIN: NOP BRA MAIN</pre> ⇒ラベル名又はシンボル名を確認してください。
17	Relative jump is out of range	相対ジャンプ命令のジャンプ先アドレスが範囲内にありません。 ⇒プログラムの再配置を行うか、命令を変更してください。
18	Label is reserved word	ラベル又はシンボルに予約語と同一名称が使われています。 例) <pre>A .EQU 1FFH</pre> ⇒ラベル名又はシンボル名を変更してください。

エラー番号	エラーメッセージ	エラー内容と対策
19	Reference to undefined label	未定義のラベル又はシンボルを参照しています。 ⇒ラベル又はシンボルを確認してください。
20	Value error	データ記述形式に誤りがあります。 例) ADC #'A ⇒データ記述形式を確認してください。
21	Value is out of range	データの範囲が許容値を越えています。 例) ADC #100H ⇒オペランド記述形式を確認してください。
22	"()" format error	左カッコ '(' と右カッコ ')' の数がありません。 例) ADC (WORK ⇒オペランド記述形式を確認してください。
23	Label error	ラベルの文字数が 16 文字以上か、使用できない文字を含んでいます。 例) L123456789012345: ⇒ラベルを確認してください。

A.3ワーニング一覧表

ワーニングを検出すると、ワーニングメッセージを画面と PRN ファイルに出力します。表 A.3 にワーニングとその内容を示します。

表 A.3:ワーニング一覧表

ワーニング番号	ワーニングメッセージ	ワーニング内容と対策
1	Phase warning	1)疑似命令.ORG で指定したアドレスが、それ以前のアドレスよりも前になっています。 例) .ORG 0E000H MAIN: LDA WORK : .ORG 0C000H
		2)命令がこの行より後方のラベル又はシンボルを参照しています。 例) LDA WK,X : WK .EQU 80H ⇒ラベル又はシンボルを参照する行より前で定義してください。
2	.END statement in include file	疑似命令.END をインクルードファイル中に記述しています。 ⇒.END をソースファイル中に記述してください。

付録B

疑似命令一覧

B.1疑似命令一覧の見方

ASM72 で使用可能な疑似命令をアルファベット順に掲載します。表記上の規則を以下に示します。

1. 説明中の[]内の部分は省略可能部分を示しています。
2. △はスペースまたはタブコードを示しています。
3. 以下の説明ではラベルと疑似命令間は△で示しています。ラベルを記述する場合';'(コロン)は必須ではありません。
4. ラベルと疑似命令の間にスペース、またはタブコードが必要になります。

B.2疑似命令一覧

.COL

コラム数指定(既定値は 132)

書式

△.COL△数値

内容

- リストの 1 行の文字数(80 または 132)を指定します。
- 79 以下を指定した場合は 80 に、81 以上を指定した場合は 132 文字になります。
- 本疑似命令は、プログラム中に 1 回だけ記述できます。

記述例

```
.COL      80      ; コラム数を 80 に設定します。  
:
```

.DW

データ設定

書式

△ラベル△.DW△数式

内容

- 数式のもつ値を設定します(単位:ワード)。
- 2つ以上のデータを設定する場合は、各データを','で区切って指定してください。
- 1行で指定できる個数は16個までです。

記述例

```
label:  .DW      1FF      ; 1FFH を設定します。  
      .DW      symbol   ; symbol のもつ値を設定します。  
      :  
      :
```

書式

△.END

内容

- ソースプログラムの終了を指定します。
- 本疑似命令以降の行はアセンブルしません。

記述例

```

:
.END                ; プログラムの終了を宣言します。

```

書式 1

シンボルタイプ 1△.EQU△数値

書式 2

シンボルタイプ 1△.EQU△数値 X,数値 Y

書式 3

シンボルタイプ 1△.EQU△数値 Z,数値 X,数値 Y

書式 4

シンボルタイプ 1△.EQU△ビット位置,数値 Z,数値 X,数値 Y

内容

- 左辺のシンボルに数値を割り当てます。
- 書式 1 はシンボルに 16 ビットの整数値を割り当てます。
- 書式 2 はシンボルにレジスタ X の値とレジスタ Y の値を割り当て、LXY 命令で使えます。
- 書式 3 はシンボルにレジスタ Z の値、レジスタ X の値とレジスタ Y の値を割り当てます。また、LXY 命令、LZ 命令、マクロ疑似命令(.LZXY)で使えます。
- 書式 4 はシンボルに 0~3 のビット値、レジスタ Z の値、レジスタ X の値とレジスタ Y を割り当てます。また、LXY 命令、LZ 命令、SB 命令、RB 命令、SZB 命令及びマクロ命令(.LZXY、.CLB、.SZXYB、.SEB)で使えます。
- 数値に使用するシンボルは、この行より前に定義してください。

記述例

```

COUNT   .EQU    4           ; 書式 1
TYPE     .EQU    3,0        ; 書式 2
DATE5    .EQU    0,3,4      ; 書式 3(ビット位置指定なし)
FLAG0    .EQU    1,0,3,4    ; 書式 3(ビット位置指定)
LXY      TYPE     ; X=3,Y=0
LXY      DATE5    ; X=3,Y=4
LXY      FLAG0    ; X=3,Y=4
LZ       DATE5    ; Z=0
LZ       FLAG0    ; Z=0
SB       FLAG0    ; DP の内容の第 1 ビットを 1 にセットします。
RB       FLAG0    ; DP の内容の第 1 ビットを 0 にセットします。
SZB      FLAG0    ; DP の内容の第 1 ビットが 0 のとき次の命令
                ; をスキップします。

```

書式

△.IF△数式

<文 1>

△.ELSE

<文 2>

△.ENDIF

内容

- 数式中には、同義命令の書式 1 のタイプ及び-D パラメータで定義されたラベルが記述できません。
- .IF に続く数式が真(0 でない)なら文 1 をアセンブルし、偽(0)なら文 2 をアセンブルします。
- 本疑似命令は入れ子にできません。
- 文 1、文 2 には複数行が記述できます。
- 数式中に使用するラベルまたはシンボルは、この行より前に定義してください。

記述例

```
.IF      FLAG      ; FLAG の値が真なら.ELSE までをアセンブルします。
:
:
.ELSE          ; 偽なら.ENDIF までをアセンブルします。
:
:
.ENDIF
```

書式

△.INCLUDE△ファイル名

内容

- 本疑似命令を記述した場所に、オペランドで指定したファイルの内容を読み込みます。
- ファイル名はフルネームで指定してください。
- ファイル名にパス指定が記述可能です。

記述例

```
.INCLUDE      TEST.INC      ; TEST.INC の内容を読み込む
:
:
```

書式

△.LINE△数値

内容

- リストの 1 ページ当たりの行数(5~255)を指定します。
- なお、行数にはヘッダー 4 行が含まれます。
- 本疑似命令は、プログラム中に 1 回だけ記述できます。
- オペランドに使用するシンボルは、この行より前に定義してください。

記述例

```
.LINE      60      ; 行数を 60 に設定します。
:
:
```

.LIST

リスト出力開始(既定値)

書式

△.LIST

内容

- PRN ファイルへのリスト出力を行います。
- 疑似命令.NLIST により PRN ファイルへの出力を中止後、リスト出力を再開する場合に使用します。

記述例

```
.NLIST          ; リスト出力の抑止を行います。
:              ; ".LIST"までの間を PRN ファイルへ出力しません。
:
.LIST           ; リスト出力の開始を行います。
:              ; この疑似命令以降を PRN ファイルへ出力します。
:
```

.NLIST

リスト出力抑止

書式

△.NLIST

内容

- PRN ファイルへの出力を抑止します。
- この状態は疑似命令.LIST により解除できます。

記述例

```
.NLIST          ; リスト出力の抑止を行います。
:              ; ".LIST"までの間を PRN ファイルへ出力しません。
:
.LIST           ; リスト出力の開始を行います。
:              ; この疑似命令以降を PRN ファイルへ出力します。
:
```

.ORG

アドレス宣言(既定値は 0000H)

書式 1

△.ORG△アドレス

書式 2

△.ORG△ページ, オフセット

内容

- この行以降の開始アドレスを宣言します。
- 指定がない場合、開始アドレスは 0000H として処理します。
- 数式中に使用するラベルまたはシンボルは、この行より前に定義してください。

記述例

```
.ORG      780H      ; 15 ページの番地に設定します。
:
.ORG      5,10      ; 5 ページの 10 番地に設定します。
```

書式 Δ .PAGE Δ ['タイトル']**内容**

- この命令の直前でリストの改ページを行い、オペランドで指定したタイトルをリストのヘッダ部に出力します。
- タイトルは、'(シングルクォート)または"(ダブルクォート)で囲んで記述してください。
- タイトルの文字数は、カラム指定が 80 のとき最大 20 文字、132 のとき最大 30 文字まで使用できます。なおタイトル省略時には改ページのみを行います。

記述例

```
.PAGE      'PROG1'    ; PROG1 を PRN ファイルのヘッダ部に出力します。  
:
```

書式 Δ .TTL Δ ['タイトル']**内容**

- この命令の直前でリストの改ページを行い、オペランドで指定したタイトルをリストのヘッダ部に出力します。
- タイトルは、'(シングルクォート)で囲んで記述してください。
- タイトルの文字数は、最大 16 文字まで使用できます。

記述例

```
.TTL      'PROG1'    ; PROG1 を PRN ファイルのヘッダ部に出力します。  
:
```

付録C

マクロ命令一覧

C.1マクロ命令一覧の見方

ASM72 で使用可能なマクロ命令をアルファベット順に掲載します。表記上の規則を以下に示します。

1. 説明中の[]内の部分は省略可能部分を示しています。
2. △はスペース又はタブコードを示しています。
3. 以下の説明ではラベルとマクロ命令間は△で示しています。
4. ラベルを記述する場合:(コロン)は必須ではありません。
5. ラベルとマクロ命令の間にスペース、又はタブコードが必要になります。

C.2マクロ命令一覧

.CLB

ビットクリア命令

書式

△ラベル△.CLB△シンボルタイプ 4

内容

- 指定したシンボルタイプ 4 の示すビットを"0"に設定します。

記述例

【 記述例 】

```
FLAG0 .EQU 1,0,3,4
      .CLB FLAG0
      :
```

【 展開例 】

```
      LXY 3,4
      RB 1
      :
```

.LZXY

レジスタ設定命令

書式

△[ラベル] △.LZXY△ビットシンボル

内容

- 指定したシンボルの示す Z, X, Y を各レジスタへ設定します。
- Zレジスタが存在しない場合は使用することができません。

記述例

【 記述例 】

```
FLAG0 .EQU 1,0,3,4
      .LZXY FLAG0
      :
```

【 展開例 】

```
      LZ 0
      LXY 3,4
      :
```

書式

△[ラベル:] △.SEB△シンボルタイプ 4

内容

- 指定したシンボルタイプ 4 の示すビットを"1"に設定します。

記述例

【 記述例 】

```
FLAG0      .EQU      1,0,3,4
            .SEB      FLAG0
            :
```

【 展開例 】

```
            LXY       3,4
            SB        1
            :
```

.SZXYB**書式**

△[ラベル].SZXYB△ビットシンボル

内容

- 指定したビットシンボルの示すビットをテストし"0"ならつぎの命令をスキップします。

記述例

【 記述例 】

```
FLAG0      .EQU      1,0,3,4
            .SZXYB   FLAG0
            :
```

【 展開例 】

```
            LXY       3,4
            SZB       1
            :
```

MEMO

第2部

720シリーズ用
クロスリファレンサ

CRF72操作マニュアル

目次

1 CRF72 操作マニュアルの構成.....	2-3
2 概要.....	2-4
2.1 機能.....	2-4
2.2 生成ファイル.....	2-4
2.3 CRF ファイルの構成.....	2-4
3 操作方法.....	2-6
3.1 起動方法.....	2-6
3.2 入力パラメータ.....	2-6
3.2.1 ソースファイル名.....	2-6
3.2.2 コマンドパラメータ.....	2-6
3.3 入力方法.....	2-6
3.3.1 コマンド行入力.....	2-6
3.4 エラー.....	2-7
3.4.1 エラーの種類.....	2-7
3.4.2 DOS への戻り値.....	2-7
3.5 環境変数.....	2-7
A エラーメッセージ一覧表.....	2-8

第 1 章

CRF72 操作マニュアルの構成

CRF72 操作マニュアルは、以下の章から構成されています。

- 第 2 章 概要
CRF72 の基本的な機能と、CRF72 が生成するファイルについて紹介します。
- 第 3 章 操作方法
CRF72 のコマンド入力方法について説明します。
- 付録 A エラー一覧表
CRF72 が表示するエラーメッセージについて、その内容と対策を一覧表で示します。

本マニュアルは、CRF72 V.1.xx.xxC に対応しています。

第2章

概要

CRF72 は、ソースファイル内のラベル及びシンボルの相互参照リスト(以下クロスリファレンスリストと呼ぶ)を生成します。このリストを使用すれば、プログラム修正時にソースファイル各部間の依存関係を容易に把握することができます。

2.1 機能

CRF72 は、ASM72¹と共に利用できます。CRF72 では、以下の機能によりソースファイルの把握を効率的に行うことができます。

1. ラベルの参照命令の種類を参照行番号に表示します。
2. 疑似命令.INCLUDE によるファイル読み込みの実行が可能です。
3. 疑似命令.PAGE によるヘッダの出力が可能です。

2.2 生成ファイル

CRF72 では、次のファイルを生成します。

1. クロスリファレンスファイル(以下 CRF ファイルと呼びます)
 - ラベル及びシンボル名の相互参照リストを示します。
 - 1 行当たりのカラム数は 80、1 ページの行数は 57 行に固定です。
 - このファイルは、プリンタに出力してデバッグやエディットの際にお使いください。
 - ファイル属性は、.CRF です。

2.3 CRF ファイルの構成

図 2.1 に、CRF ファイルの出力例を示します。CRF ファイルは、以下の情報を示しています。

1. ラベル及びシンボル名の横の数字は、そのラベルが使用されているアセンブルリスト上の行番号を示します。
2. ‘#’(シャープ)記号は、その行でラベルが定義されていることを示しています。
3. 参照行のうち BM、BML、BMLA 命令で行っている行は、’&’(アンパサンド)を示します。
4. ラベル及びシンボル名は、8 文字まで表示します。8 文字以上の場合は省略します。
5. リストのヘッダには、疑似命令.PAGE で指定しているタイトルを表示します。(ただし、表示は 25 文字まで行います。)
6. CRF72 は、ソースプログラム中のラベル及びシンボルの値の判断は行いません。従って、条件付きアセンブルの判断はできませんのでご注意ください。

1. 720 シリーズ用アブソリュートアセンブラ名。

M50720 CROSS REFERENCE V1.00C 727TEST PROG. P.001

ATOC	177&	396#					
A_D	118	174#					
COUNT	95#	154&	166&	201&	214&	379&	388&
D0_SET	250&	333#					
D10	318	320#					
D10_SET	322&	363#					
D1_SET	254&	336#					
D2	257	259#					
D2_SET	261&	339#					
D3	264	266#					
D3_SET	268&	342#					
D4	271	273#					
D4_SET	275&	345#					
D5	278	280#					
D5_SET	282&	348#					
D6	285	287#					
:							

図 2.1: CRF ファイル例

第3章

操作方法

3.1 起動方法

CRF72 を実行するためには、以下の情報(入力パラメータ)を入力する必要があります。

1. ソースファイル名(必須項目)
2. コマンドパラメータ

3.2 入力パラメータ

3.2.1 ソースファイル名

1. ソースファイル名は必ず入力してください。
2. ファイル属性(.ASM)を省略した場合、既定値として属性.ASM を選びます。
3. ファイル名をフルネームで指定することにより、.ASM 以外の属性のファイルも処理可能です。
4. ファイル名は、ドライブ名が指定可能です。ファイル名のみを記述した場合、カレントドライブ中のファイルを処理します。ディレクトリパスの指定はできません。
5. ソースファイル名は 16 個まで指定できます。

3.2.2 コマンドパラメータ

コマンドパラメータは、ソースファイル中の疑似命令.INCLUDE の検出の有無や、出力ファイルのドライブ名を指定します。表 3.1 に、コマンドパラメータの内容を示します。

表 3.1:コマンドパラメータ一覧表

コマンドパラメータ	内容
-O	出力ファイルのディレクトリ名を指定します。指定の書式は、次のようになります。 例) A>CRF72 TEST -O¥TEMP<RET> クロスリファレンスファイルを C:¥TEMP ディレクトリに出力します。

3.3 入力方法

3.3.1 コマンド行入力

CRF72 は、DOS のプロンプト状態でコマンド行を入力することにより起動します。図 3.1 に、起動コマンドの入力例を示します。

```
A>CRF72 TEST<RET>
```

図 3.1:コマンド行入力例

コマンド行入力中に誤りを検出すると、図 3.2 のようにヘルプ画面を表示し処理を中止します。

```
C:¥>CRF72<Enter>

720 SERIES CROSS REFERENCE V.1.01.01C
COPYRIGHT(C) 1992 (1992-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

Usage: crf72 <filename> [-.] [-opath]
  .. : all messages suppressed
  -o : select drive and directory for output ( use -otmp )
```

図 3.2: コマンド行エラー時のヘルプ画面

3.4 エラー

3.4.1 エラーの種類

CRF72 実行時に発生するエラーは、以下の原因によるものがあります。

1. DOS に関するエラー
ディスクやメモリ容量の不足等、CRF72 を実行する DOS 環境に関わるエラーです。付録 A エラーメッセージ一覧表を参照の上、DOS のコマンドにより対応してください。
2. CRF72 のコマンド行入力に関するエラー
CRF72 起動時のコマンド行入力に関わるエラーです。本章の内容を確認の上コマンドを再入力して下さい。
3. 処理対象のソースファイル内容に関するエラー
指定されたソースファイルが存在しない場合に発生します。

CRF72 は、エラーを検出すると図 3.3 の形式でエラー内容を画面に表示します。付録 A のエラー一覧表を参照の上処理して下さい。

```
C:¥>CRF72 TEST<Enter>

720 SERIES CROSS REFERENCE V.1.01.01C
COPYRIGHT(C) 1992 (1992-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

Error 2:Can't open file ( TEST.ASM )
```

図 3.3: エラー表示例

3.4.2 DOS への戻り値

CRF72 では、DOS への戻り値を 0 で返すようにしています。

3.5 環境変数

CRF72 は、DOS の環境変数は使用していません。

付録A

エラーメッセージ一覧表

表 A.1:エラーメッセージ一覧表

エラー番号	エラーメッセージ	エラー内容と対策
1	Can't make cross reference	CRF72 を起動できません。 ⇒メモリ容量を確認してください。
2	Can't open XXX.ASM.	ファイルをオープンできません。 ⇒XXX.ASM ファイルの確認をしてください。 ⇒また、メモリ容量を確認してください。
3	Can't close XXX.ASM.	ファイルをクローズできません。 ⇒メモリ容量を確認してください。
4	Out of disk space.	CRF ファイルを出力できません。 ⇒メモリ容量を確認してください。

第3部

720シリーズ用
ブランチ範囲外エラー修正ツール

BRANCH操作マニュアル

目次

1 BRANCH 操作マニュアルの構成.....	3-3
2 概要.....	3-4
2.1 機能.....	3-4
2.2 生成ファイル.....	3-4
3 操作方法.....	3-5
3.1 起動方法.....	3-5
3.2 入力パラメータ.....	3-5
3.2.1 ソースファイル名.....	3-5
3.2.2 コマンドパラメータ.....	3-5
3.3 入力方法.....	3-5
3.3.1 コマンド行入力.....	3-5
3.4 エラー.....	3-7
3.4.1 エラーの種類.....	3-7
3.4.2 DOS への戻り値.....	3-7
A エラーメッセージ一覧表.....	3-8
A-1 エラー一覧表.....	3-8

第1章

BRANCH操作マニュアルの構成

BRANCH操作マニュアルは、以下の章から構成されています。

- 第2章 概要
BRANCHの基本的な機能と、BRANCHが生成するファイルについて紹介します。
- 第3章 操作方法
BRANCHのコマンド入力方法について説明します。
- 付録A エラーメッセージ一覧表
BRANCHが表示するエラーメッセージについて一覧を示しています。

本マニュアルは、BRANCH V.1.xx.xxに対応しています。

第2章

概要

BRANCHは、アセンブル時に発生するブランチ範囲外エラーを自動修正するソフトウェアです。対象MCUは、4ビットマイコンの4500シリーズ及び720シリーズです。

2.1 機能

BRANCHは、以下のような機能があります。

1. アセンブラASM45/ASM72を自動的に起動します。
2. ブランチ範囲外エラーが発生した全てのページ内ブランチ命令(B)をページ外ブランチ命令(BL)に変換します。
3. 変換前のアセンブラソースファイルを保存します。

2.2 生成ファイル

BRANCHでは、次のファイルを生成します。

1. 修正済みアセンブラソースファイル
ブランチ範囲外エラーを全て修正したアセンブラソースファイルです。ファイル名は、起動時に指定された入力ファイル名がそのまま用いられます。入力ファイルにファイル属性が指定されていない場合、ファイル属性は、.ASMとなります。
2. 修正前のアセンブラソースファイル
修正前のアセンブラソースファイルです。ファイル属性は、.ORGとなります。ブランチ範囲外エラーが発生しなかった場合は、生成されません。

第3章

操作方法

3.1 起動方法

BRANCHを実行するためには、以下の情報(入力パラメータ)を入力する必要があります。

1. ソースファイル名(必須項目)
2. コマンドパラメータ

3.2 入力パラメータ

3.2.1 ソースファイル名

1. 修正対象のアセンブラソースファイル名を指定します。
2. 指定可能なソースファイル数は1個です。
3. ファイル属性を省略した場合、既定値として属性.ASMを選びます。
4. ファイル名をフルネームで指定することにより、.ASM以外の属性のファイルも処理可能です。
5. ファイル名は、ディレクトリパス指定が可能です。ファイル名のみを記述した場合、カレントドライブのカレントディレクトリ中のファイル进行处理します。

3.2.2 コマンドパラメータ

コマンドパラメータでは、対象となるマイコンシリーズや、アセンブラへのオプションを指定します。表3.1に、コマンドパラメータの内容を示します。

表3.1:コマンドパラメータ一覧表

コマンドパラメータ	内容
-.	画面への全てのメッセージ出力を抑制します。
-45/72	プログラムが対象としているMCUのシリーズを指定します。-45を指定した場合は4500シリーズ用、-72を指定した場合は720シリーズ用のMCUであることを意味します。デフォルトは720シリーズです。
-ASM	ASM45/ASM72へのオプションを指定します。オプションの指定は”-ASM”以降に指定して下さい。

3.3 入力方法

3.3.1 コマンド行入力

BRANCHは、DOSのプロンプト状態でコマンド行を入力することにより起動します。以下に、起動コマンドの入力例を示します。

```
A>BRANCH SAMPLE<RET>
```

コマンド行入力中に誤りを検出すると、図3.1のようにヘルプ画面を表示し処理を中止します。

```

C:¥>BRANCH<Enter>
720/4500 Series Branch Instruction Change Tool V.1.01.00
COPYRIGHT(C) 1994 (1994-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

The file name is not specified.

Usage: branch <ASMfilename> [-.][-45/72][-ASM asmOption]

  -.      : all messages suppressed.
  -45/72 : set MCU type(4500 or 720(default)).
  -ASM   : set assembler option after -ASM.

```

図3.1:コマンド行エラー時のヘルプ画面

修正が終了するとアセンブル結果と修正数を表示します。修正が正常に終了したときの画面表示例を図3.2に示します。

```

C:¥>BRANCH TEST<Enter>
720/4500 Series Branch Instruction Change Tool V.1.01.00
COPYRIGHT(C) 1994 (1994-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

[1]720 SERIES ASSEMBLER V.1.20.02C
COPYRIGHT(C) 1989 (1989-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED
now processing pass 1

now processing pass 2
TEST.ASM 6 ( TOTAL LINE 6 ) Error 17: Relative jump is out of range "start"

ERROR    COUNT 0001
WARNING  COUNT 0000
SOURCE   LINE 0008 LINES
TOTAL    LINE 0008 LINES
COMMENT  LINE 0000 LINES
OBJECT   SIZE 0002 BYTES
[2]720 SERIES ASSEMBLER V.1.20.02C
COPYRIGHT(C) 1989 (1989-2004)
RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED
now processing pass 1

now processing pass 2

ERROR    COUNT 0000
WARNING  COUNT 0000
SOURCE   LINE 0008 LINES
TOTAL    LINE 0008 LINES
COMMENT  LINE 0000 LINES
OBJECT   SIZE 0003 BYTES

Conversion was completed.

```

図3.2:正常終了時の画面表示

3.4 エラー

3.4.1 エラーの種類

BRANCH実行時に発生するエラーは、以下の原因によるものがあります。

1. BRANCHのコマンド行入力に関するエラー
BRANCH起動時のコマンド行入力に関わるエラーです。コマンドを再入力して下さい。
2. BRANCHを実行する環境に関するエラー
ディスクやメモリ容量の不足等、BRANCHを実行する環境に関わるエラーです。

3.4.2 DOSへの戻り値

DOSのバッチファイル等に実行ファイルを記述する場合、実行結果に応じて処理の内容を変えた場合があります。BRANCHでは、実行結果を表3.2の3つのエラーレベルに分けてDOSに返すようにしています。エラーレベルの利用方法については、市販のDOSのマニュアルを参照して下さい。

表3.2:エラーレベル一覧表

エラーレベル	実行結果の内容
0	正常終了
1	BRANCHの対象ファイルに関するエラー
2	BRANCHを実行する環境に関わるエラー

付録A

エラーメッセージ一覧表

A.1 エラー一覧表

BRANCH起動中にエラーを検出すると、エラーメッセージを画面に表示し、変換を中止します。以下にエラー一覧を示します。

エラーメッセージ	エラー内容と対策
An error does not exist.	対象ファイルにエラーが存在しません。
Conversion was completed.	ブランチ命令の変換が終了しました。
The instruction for conversion does not exist.	対象ファイルに変換すべき命令がありません。
Conversion was completed, the error exists. Please correct the error of an ORG file and execute BRANCH again.	ブランチ命令の変換は終了しましたが、エラーが存在しています。 ⇒ORGファイルでエラーを修正し、再度BRANCHを実行して下さい。
Option specification of a BRANCH is wrong.	BRANCHのオプション指定が間違っています。 ⇒usageおよびアセンブラのオプションを参照して、再度コマンド入力して下さい。
Option specification of an assembler is wrong.	-ASM以降のオプション指定が間違っています。 ⇒アセンブラのオプションをマニュアルで参照して、再度コマンド入力して下さい。
The file name is not specified.	ファイル名が指定されていません。 ⇒BRANCH対象ファイルを指定して下さい。
An assembler cannot be executed. Please check a PATH, and an environmental setup of DOS.	アセンブラが実行できません。 ⇒PATHの設定等、DOSの環境設定を確認下さい。
Option specification of an assembler is wrong.	アセンブラのオプション指定が間違っています。
The system error occurred in the assembler. Please execute an assembler and check a cause.	アセンブラ内でシステムエラーが発生しました。 ⇒アセンブラを起動して原因を確認下さい。
Out of heap space.	メモリが不足しています。 ⇒DOSのデバイスドライバの排除等を行い、使用可能メモリ領域を増やして下さい。
Can't open source file.	ソースファイルがオープンできません。 ⇒コマンドのファイル名を確認して下さい。 ⇒または環境変数FILESの値を増やして下さい。
Can't open temporary file.	テンポラリファイルがオープンできません。 ⇒環境変数FILESの値を増やして下さい。
The source file has broken.	ソースファイルの内容が壊れています。
Can't write temporary file.	テンポラリファイルに書き込めません。
The specified editor does not execute.	指定エディタが起動しません。
CRFxx cannot be executed.	クロスリファレンサが起動しません。
Execute of cross referencer was completed.	クロスリファレンサの実行が終了しました。
Can't open nul device.	Nulデバイスがオープンできません。

ASM72 V.1.20 ユーザーズマニュアル

Rev.1.00
04.05.01
RJJ10J0651-0100Z

COPYRIGHT ©2004 RENESAS TECHNOLOGY CORPORATION
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

ASM72 V.1.20
ユーザーズマニュアル



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

RJJ10J0651-0100Z