

Synergy Software Package (SSP)

データシート
(参考資料)

Renesas Synergy™ プラットフォーム
Synergy ソフトウェア
SSP v1.4.0

本資料は英語版を翻訳した参考資料です。内容に相違がある場合には英語版を優先します。資料によっては英語版のバージョンが更新され、内容が変わっている場合があります。日本語版は参考用としてご使用のうえ、最新および正式な内容については英語版のドキュメントをご参照ください。

資料番号 R01DS0329EU0102、リビジョン Rev.1.02、発行日 2018 年 5 月 18 日の翻訳版です。

ご注意書き

1. 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・システムの設計において、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て生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 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。
4. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等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。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
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、産業用ロボット等
高品質水準: 輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、
金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、**Harsh environment** 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、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、データシート等において高信頼性、**Harsh environment** 向け製品と定義しているものを除き、耐放射線設計を行っていません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等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、お客様の機器・システムとしての出荷保証を行ってください。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、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の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。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
9. 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・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
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禁じます。
12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
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が直接的、間接的に支配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。

目次

1. 概要	6
1.1 特長	7
1.2 はじめに	9
2. ThreadX® RTOS と X-Ware.....	11
2.1 ThreadX と X-Ware の概要.....	11
2.2 ThreadX®コンポーネントの概要.....	13
3. SNMP エージェント.....	16
4. NetX 組み込み TCP/IP および UDP スタック	17
5. NetX Duo デュアル IPv4/IPv6 スタック	18
6. NetX アプリケーションバンドル.....	19
7. NetX Duo アプリケーションバンドル	20
8. NetX Secure および MQTT for NetX Duo.....	21
8.1 NetX Secure	21
8.2 MQTT for NetX Duo	22
9. FileX 組み込みファイルシステム	23
10. GUIX GUI 開発ツールキット.....	24
11. USBX.....	25
12. アプリケーションフレームワーク.....	27
12.1 概要	27
12.2 SSP v1.4.0 で利用可能なソフトウェアフレームワークモジュールの一覧.....	28
12.3 周期サンプリング ADC フレームワーク	30
12.4 オーディオ再生フレームワーク	31
12.5 オーディオ再生 DAC フレームワーク.....	32
12.6 オーディオ再生 I ² S フレームワーク.....	33
12.7 オーディオ録音 ADC フレームワーク.....	34
12.8 オーディオ録音 I ² S フレームワーク.....	35
12.9 Bluetooth Low Energy (BLE) フレームワーク.....	36
12.10 ブロックメディア QSPI フレームワーク	38
12.11 ブロックメディア RAM フレームワーク.....	39

12.12	ブロックメディア SDMMC フレームワーク	39
12.13	静電容量式タッチセンシングユニット (CTSU) ボタンフレームワーク	39
12.14	静電容量式タッチセンシングユニット (CTSU) フレームワーク	41
12.15	静電容量式タッチセンシングユニット (CTSU) スライダーフレームワーク	42
12.16	セルラーフレームワーク	43
12.17	NetX Telnet 通信フレームワーク	47
12.18	USBX 通信フレームワーク	48
12.19	コンソールフレームワーク	49
12.20	暗号化フレームワーク	50
12.21	外部割り込みフレームワーク	52
12.22	I ² C フレームワーク	53
12.23	JPEG デコードフレームワーク	54
12.24	メッセージフレームワーク	55
12.25	パワープロファイルフレームワークバージョン 1	56
12.26	パワープロファイルフレームワークバージョン 2	57
12.27	シリアルペリフェラルインタフェース (SPI) フレームワーク	58
12.28	Synergy FileX ポートブロックメディアインタフェースフレームワーク	59
12.29	Synergy GUIX インタフェースフレームワーク	60
12.30	Synergy NetX ポートイーサネットモジュール	61
12.31	Synergy USBX ポートフレームワーク	62
12.32	スレッドモニタフレームワーク	63
12.33	タッチパネル I ² C フレームワーク	64
12.34	UART 通信フレームワーク	65
12.35	Wi-Fi フレームワーク	66
13.	CMSIS DSP ライブラリ	68
14.	ハードウェア抽象化レイヤー (HAL) モジュール	69
14.1	概要	69
14.2	SSP v1.4.0 で利用可能なモジュールの一覧	69
14.3	高速アナログコンパレータ (ACMPHS)	72
14.4	低消費アナログコンパレータ (ACMPLP)	72
14.5	A/D コンバータ (ADC)	73
14.6	非同期汎用タイマ (AGT)	75
14.7	クロック周波数精度測定回路 (CAC)	76
14.8	コントローラエリアネットワーク (CAN)	79
14.9	クロック発生回路 (CGC)	81
14.10	CRC 演算器 (CRC)	82
14.11	静電容量式タッチセンシングユニット (CTSU)	83
14.12	12 ビット D/A コンバータ (DAC12)	84
14.13	8 ビット D/A コンバータ (DAC8)	85
14.14	ダイレクトメモリアクセスコントローラ (DMAC)	86

14.15	データ演算回路 (DOC)	87
14.16	データトランスファコントローラ (DTC)	88
14.17	イベントリンクコントローラ (ELC)	89
14.18	フラッシュメモリ	90
14.19	ファクトリーMCU 情報 (FMI)	92
14.20	グラフィックス LCD コントローラ (GLCDC)	93
14.21	汎用 PWM タイマ (GPT)	95
14.22	独立ウォッチドッグタイマ (IWDT)	96
14.23	インプットキャプチャ付き汎用 PWM タイマ (GPT_INPUT_CAPTURE)	98
14.24	割り込みコントローラユニット (外部割り込み) (ICU)	99
14.25	汎用 I/O ポート (GPIO/IOPORT)	100
14.26	JPEG コーデック	102
14.27	JPEG エンコード	103
14.28	キー割り込み (KINT)	105
14.29	ローパワーモード V1 (LPMV1)	106
14.30	ローパワーモード V2 (LPMV2)	106
14.31	低電圧検出 (LVD)	108
14.32	オペアンプ (OPAMP)	109
14.33	パラレルデータキャプチャユニット (PDC)	111
14.34	クワッド SPI (QSPI)	113
14.35	リアルタイムクロック (RTC)	114
14.36	I ² C バスインタフェース (RIIC)	116
14.37	SD マルチメディアカード (SDMMC)	120
14.38	セキュア暗号エンジン (SCE)	123
14.39	シリアルコミュニケーションインタフェース I ² C (SCI_I2C)	125
14.40	シリアルコミュニケーションインタフェース SPI (SCI_SPI)	126
14.41	シリアルコミュニケーションインタフェース UART (SCI_UART)	128
14.42	シリアルペリフェラルインタフェース (RSPI)	131
14.43	セグメント LCD コントローラ (SLCDC)	134
14.44	シリアルサウンドインタフェース (SSI)	135
14.45	シグマデルタ ADC (SDADC)	137
14.46	ウォッチドッグタイマ (WDT)	138
15.	ボードサポートパッケージ (BSP)	140
16.	推定メモリサイズ	142
16.1	GCC ROM の推定メモリサイズ	143
16.2	IAR ROM の推定メモリサイズ	156
16.3	暗号化の推定メモリサイズ	169
16.4	DSP ライブラリの推定メモリサイズ	171
16.5	X-Ware の推定メモリサイズ	173

1. 概要

Renesas Synergy プラットフォームの中心となる Synergy Software Package (SSP) は、リアルタイム OS (RTOS)、ミドルウェア、コミュニケーションスタック、関数ライブラリ、高度なアプリケーションフレームワーク、ハードウェア抽象化レイヤー (HAL) ドライバー、ボードサポートパッケージ (BSP) から構成される統合ソフトウェアです。

Express Logic 社の X-Ware™は SSP の主要なコンポーネントのひとつです。X-Ware には、高機能な RTOS およびミドルウェアである ThreadX®、NetX™ IPv4 および NetX Duo™ IPv4/IPv6 に準拠する TCP/IP スタック、USB ホスト/デバイスプロトコルスタックである USBX™、MS-DOS 互換ファイルシステムの FileX® および GUIX™グラフィックスランタイムライブラリなどが含まれています。

これらのコンポーネントは、Renesas Synergy マイクロコントローラの使用に最適化された SSP に組み込まれており、IEC/ISO/IEEE-12207 ソフトウェアライフサイクルプロセス標準に沿って、MISRA C:2012 (MISRA C は C プログラミング言語を使う重要システム向けの一連のプログラミングガイドライン) ガイドラインを使用して開発されています。SSP の継続的な保守およびサポートなどの各種サービスはルネサスが直接行います。データシートに記載されている条件でソフトウェアが動作することを保証 (Warranty) します。

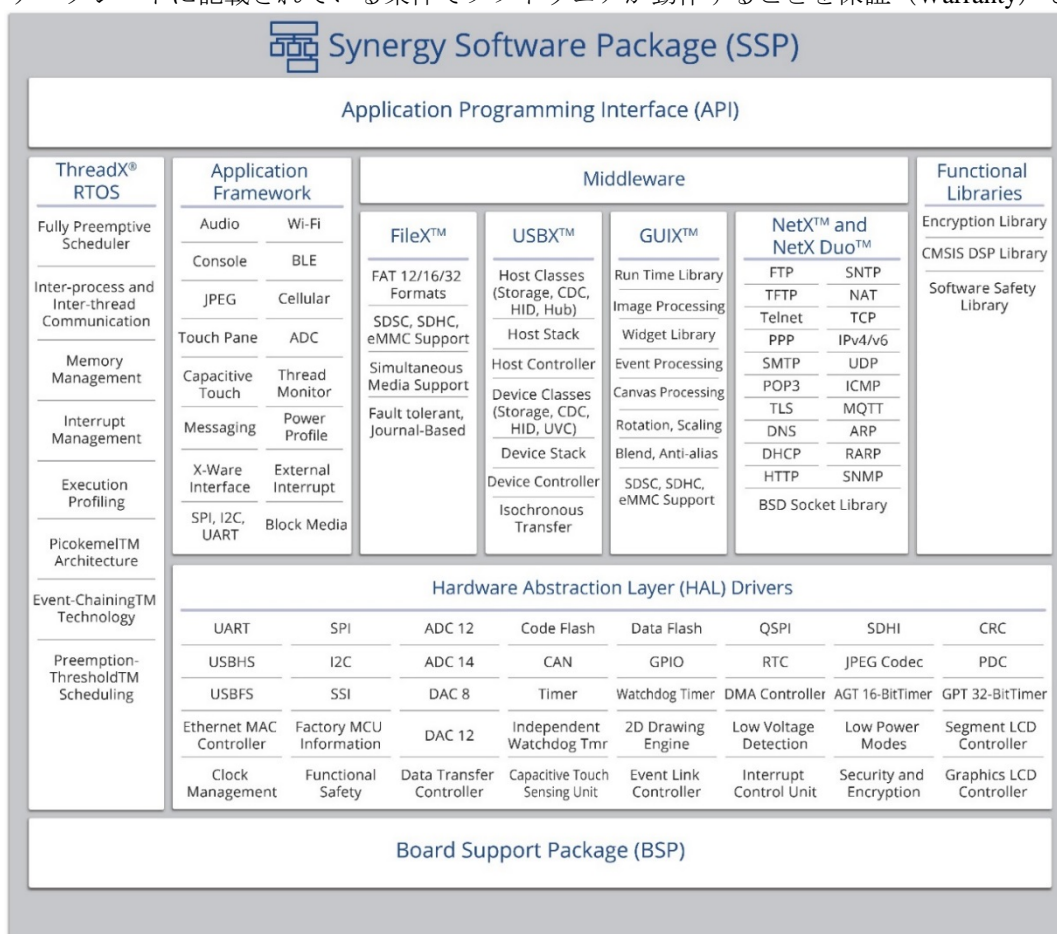


図 1.1 SSP の概念図

SSP は次の 2 つの開発環境で動作します。

1. Renesas e² studio 統合ソリューション開発環境 (ISDE) は、独自の ISDE ソフトウェアコンフィギュレーターによりパワフルになった GCC C/C++コンパイラツールチェーンを備えています。ユーザーは、e² studio 統合ソリューション開発環境で IAR コンパイラをインストールして使用することもできます。
2. IAR Embedded Workbench® for Renesas Synergy™ (IAR EW for Synergy)

どちらの開発環境も Synergy ギャラリーからダウンロードして利用できます。各主要コンポーネントの個別インストーラに加え、SSP v1.3.3 以降 (SSP v1.4.0 リリースを含む) には新しいプラットフォームインストーラが搭載されているため、それを利用することで簡単かつ迅速に開発を開始できます。

1.1 特長

ThreadX RTOS

- マルチスレッドが高度に組み込まれたリアルタイムシステム
- 小サイズで高速な picokernel™アーキテクチャ
- マルチスレッド機能
- プリエンプティブおよび協調スケジューリング
- 柔軟なスレッド優先度サポート (32~1024 優先度レベル)
- 小さいメモリフットプリントと高速な割り込み応答時間
- 最適化された割り込み処理
- スタックポインタオーバーフローモニター

GUIX

- 2D グラフィックスアクセラレータをサポート
- 無制限のオブジェクト (画面、ウィンドウ、ウィジェット)
- 動的なオブジェクトの作成・削除
- 高色深度でのアルファブレンドとアンチエイリアスのサポート
- 完全なウィンドウ化のサポート、ビューポートと Z オーダーの保持
- 複数キャンバスと複数の物理ディスプレイ
- ウィンドウのブレンドとフェード
- 画面遷移、スプライト、動的アニメーション
- タッチスクリーンと仮想キーボード
- UTF8 文字列エンコーディングを使用した多言語対応
- オブジェクトサイズの自動拡大・縮小
- 8 ビットカラールックアップテーブル (CLUT) のサポート
- タッチローテーション
- 円形の進行状況バー

USBX

- USB 2.0 Full Speed および High Speed をサポート
- デバイスクラス : MSC、HID、CDC
- ホストクラス : MSC、HID、CDC ACM、UVC
- 高速 DMA 転送のサポート

FileX

- ThreadX と統合された MS-DOS 互換ファイルシステム
- FAT12/16/32 ビットサポート
- フォールトトレラントなファイルシステム (ジャーナリングを使用)
- 無制限のメディアインスタンス (メディア、ディレクトリ、ファイル)

NetX

- イーサネットドライバ
- IPv4 準拠 TCP/IP プロトコルスタック
- ThreadX との統合
- ゼロコピーAPI

- UDP 高速パステクノロジ
- BSD ソケット互換 API
- RFC791 インターネットプロトコル (IP)
- RFC826 アドレス解決プロトコル (ARP)
- RFC903 逆アドレス解決プロトコル (RARP)
- RFC792 インターネット制御通知プロトコル (ICMP)
- RFC3376 インターネットグループ管理プロトコル (IGMP)
- RFC768 ユーザーデータグラムプロトコル (UDP)
- RFC793 伝送制御プロトコル (TCP)
- RFC1112 IP マルチキャスト用ホスト拡張機能

NetX Duo

- IPv4/IPv6 準拠 TCP/IP プロトコルスタック
- ThreadX との統合
- ゼロコピーAPI
- UDP 高速パステクノロジ
- BSD ソケット互換 API
- RFC2460 IPv6 仕様
- RFC4861 IPv6 近隣探索
- RFC4862 IPv6 ステートレスアドレス
- RFC1981 IPv6 経路 MTU 探索
- RFC4443 ICMPv6
- RFC791 インターネットプロトコル (IP)
- RFC826 アドレス解決プロトコル (ARP)
- RFC903 逆アドレス解決プロトコル (RARP)
- RFC792 インターネット制御通知プロトコル (ICMP)
- RFC3376 インターネットグループ管理プロトコル (IGMP)
- RFC768 ユーザーデータグラムプロトコル (UDP)
- RFC793 伝送制御プロトコル (TCP)
- RFC1112 IP マルチキャスト用ホスト拡張機能

NetX (IPv4 Networking Services)

- DHCP クライアント・サーバー
- DNS クライアント
- HTTP クライアント・Web サーバー
- FTP クライアント・サーバー
- TFTP クライアント・サーバー
- Telnet クライアント・サーバー
- Auto IP
- NAT
- SMTP クライアント
- POP3 クライアント・サーバー
- SNMP エージェント
- SNMP クライアント
- PPP (Synergy コンフィギュレーションツールではサポート対象外)

NetX Duo (IPv4/IPv6 Networking Services)

- DHCP クライアント・サーバー
- DNS クライアント
- HTTP クライアント・Web サーバー
- FTP クライアント・サーバー
- TFTP クライアント・サーバー
- Telnet クライアント・サーバー
- Auto IP
- NAT
- SMTP クライアント
- POP3 クライアント・サーバー
- SNMP エージェント
- SNTP クライアント
- PPP (Synergy コンフィギュレーションツールではサポート対象外)

NetX Secure

- TLS v1.0、TLS v1.2

アプリケーションフレームワーク

- オーディオ再生フレームワーク
- オーディオ再生 HW DAC フレームワーク
- オーディオ再生 HW FS フレームワーク
- オーディオ録音フレームワーク
- オーディオ録音 HW ADC フレームワーク
- SD マルチメディアカード用ブロックメディアインタフェースフレームワーク
- BLE フレームワーク
- 静電容量式タッチセンシングユニットフレームワーク
- 静電容量式タッチセンシングユニットボタンフレームワーク
- 静電容量式タッチセンシングユニットスライダーフレームワーク
- セルラーフレームワーク
- コンソールフレームワーク
- 外部割り込みフレームワーク
- I²C フレームワーク
- スレッド間メッセージフレームワーク
- JPEG デコードフレームワーク
- 周期サンプリング ADC フレームワーク
- 電力モードプロファイルフレームワーク V1
- 電力モードプロファイルフレームワーク V2
- シリアルペリフェラルインタフェース (SPI) フレームワーク
- Synergy FileX インタフェースフレームワーク
- Synergy GUIX インタフェースフレームワーク
- Synergy NetX 通信インタフェースフレームワーク
- Synergy USBX 通信インタフェースフレームワーク
- スレッド監視フレームワーク
- タッチパネルフレームワーク
- UART フレームワーク
- Wi-Fi フレームワーク

セキュリティ暗号化ライブラリ

- True RNG (TRNG)
- SHA1/SHA224/SHA256
- ECC P-192、ECC P-256、ECDSA
- AES128/192/256 ビット
- 3DES、192 ビットキー、ECB、CBC、CTR
- ARC4
- RSA (最大 2048 ビットキー)
- DLP、DSA (最大 2048 ビットキー)
- 鍵生成 (平文およびラップ鍵) とインストール
- 署名生成および認証
- MD5

CMSIS DSP ライブラリ

- 基本的な演算機能
- 高速演算機能
- 複雑な演算機能
- フィルタ
- コンボリユーション
- 行列関数
- 変換
- モーター制御機能
- 統計関数
- サポート機能
- 補完関数

MQTT for NetX Duo

- MQTT

メモリサポート

- JTAG によるフラッシュプログラミングサポート
- コードフラッシュおよびデータフラッシュドライバ
- 外部メモリバスのサポート

ヒューマンマシンインタフェース (HMI)

- グラフィックス LCD コントローラ (GLCDC)
- セグメント LCD コントローラ (SLCDC)
- 静電容量式タッチセンシングユニット (CTSUS)

ハードウェア抽象化レイヤー (HAL) ドライバー

- A/D コンバータ (12 ビット ADC、14 ビット ADC)
- 非同期汎用タイマ (AGT)
- 静電容量式タッチセンシングユニット (CTSUS)
- クロック周波数精度測定回路 (CAC)
- クロック発生回路 (CGC)
- コントローラエリアネットワークインタフェース (CAN)
- CRC 演算器 (CRC)
- データ演算回路 (DOC)
- データトランスファコントローラ (DTC)
- D/A コンバータ (DAC)
- 8 ビット D/A コンバータ (DAC)

- ダイレクトメモリアクセスコントローラ (DMAC)
- イベントリンクコントローラ (ELC)
- ファクトリーMCU情報 (FMI)
- フラッシュメモリ、ハイパフォーマンス (FLASH_HP)
- フラッシュメモリ、ローパワー (FLASH_LP)
- 汎用 I/O ポート (GPIO/IOPORT)
- 汎用タイマ (GPT)
- 汎用インプットキャプチャ (GPT_INPUT_CAPTURE)
- グラフィックス LCD コントローラ (GLCDC)
- I²C バスインタフェース (RIIC)
- 独立ウォッチドッグタイマ (IWDT)
- 割り込みコントローラユニット (ICU)
- JPEG コーデック (JPEG_COMMON、JPEG_ENCODE、JPEG_DECODE)
- キー割り込み (KINT)
- (廃止予定) ローパワーモード (LPM)
- ローパワーモード V2 (LPMv2)
- 低電圧検出 (LVD)
- パラレルデータキャプチャユニット (PDC)
- クワッド SPI (QSPI)
- リアルタイムクロック (RTC)
- SDIO および SD/MMC メモリデバイス用の SDHI ドライバー (SDMMC)
- セグメント LCD コントローラ (SLCDC)
- シリアルコミュニケーションインタフェース I²C (SCI_I²C)
- シリアルコミュニケーションインタフェース SPI (SCI_SPI)
- シリアルコミュニケーションインタフェース UART (SCI_UART)
- シリアルペリフェラルインタフェース (SPI)
- シリアルサウンドインタフェース (SSI)
- ウォッチドッグタイマ (WDT)

ボードサポートパッケージ (BSP)

- S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA の各シリーズの MCU をサポート
- PE-HMI1、DK-S7G2、DK-S3A7、DK-S124、SK-S7G2、PK-S5D9、TB-S1JA、TB-S3A1、TB-S3A3、TB-S3A6、TB-S5D5 の各キットをサポート
- e² studio ISDE を使用したカスタム BSP の作成
- スタートアップ時のシステムの初期化と構成
- ソフトウェアとハードウェアのアクセス制御
- レジスタライト保護

GPIO とキー割り込み

- GPIO モジュール
- キー割り込みモジュール

1.2 はじめに

本データシートには、SSP に含まれる主要なソフトウェアモジュールの機能を記載しています。

今回のリリースでは、次の MCU グループの製品がサポートされています。

S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S1JA、S124、S128

本データシートでは、エンジニア向けに以下の情報を記載しています。

- SSP を構成するコンポーネントとレイヤーの一覧
- SSP の多様なレイヤーとコンポーネントの概要
- 各コンポーネントのブロック図
- 各コンポーネントで推定されるメモリの要件

本データシートの他に SSP に関する資料として、下記が提供されています。

- SSP v1.4.0 ソフトウェアユーザーズマニュアル
- SSP v1.4.0 リリースノート
- SSP Customer Care document
- e² studio ISDE リリースノート

- IAR Embedded Workbench® for Renesas Synergy™ (IAR EW for Synergy) リリースノート
- Synergy スタンドアロンコンフィギュレーターリリースノート
- アプリケーションノート付きアプリケーションプロジェクト

1.2.1 SSP の保証

ユーザーが SSP 開発/量産ライセンスを所持している場合、弊社は SSP がこのリリースバージョンのデータシートに記載されている仕様に準拠することを保証します。その他のドキュメントに記載されている他のすべての仕様は保証の対象外です。詳細については、SSP Customer Care document を参照してください。

1.2.2 ソフトウェアの品質保証とテストデータ

弊社では、SSP および各コンポーネント、モジュール、ライブラリに関する重要かつ詳細なソフトウェア品質保証とテストデータを提供しています。概要についてはウェブサイト (<https://www.renesas.com/ja-jp/products/synergy/software/ssp.html>) を参照してください。このウェブサイトには、ソフトウェア品質ハンドブックとソフトウェアの品質概要の最新情報も提供しています。

1.2.3 互換性のある開発ツール

ThreadX、X-Ware、およびすべてのバージョンの SSP は、以下の互換性があるテスト済みソフトウェア開発ツールと連携します。

ツール名	バージョン
e ² studio	6.2.0
IAR Embedded Workbench® for Renesas Synergy™	8.21.1
Synergy Standalone Configurator (SSC)	6.2.0
GNU Arm® Compiler (GCC)	GCC_4.9.3.20150529
IAR Compiler	8.21.1

すべてのバージョンの SSP はこれらのツールと互換性があり、かつこれらのツールでテストされています。詳細については SSP リリースノートを参照してください。

1.2.4 サポート対象のキット

SSP v1.4.0 では、以下のキットがサポートされています。

名称	バージョン	説明
PE-HMI1	2.0	S7G2 MCU グループを評価するためのヒューマンマシンインタフェース用製品事例です。
DK-S124	3.1	S124 MCU グループ用開発キットです。
DK-S7G2	3.1	S7G2 MCU グループ用開発キットです。
SK-S7G2	3.3	S7G2 MCU グループ用スターターキットです。
DK-S3A7	2.0	S3A7 MCU グループ用開発キットです。
PK-S5D9	1.0	S5D9 用プロモーションキットです。
DK-S128	1.1	S128 MCU グループ用開発キットです。
TB-S5D5	1.0	S5D5 MCU グループ用ターゲットボードです。
TB-S3A6	1.05b	S3A6 MCU グループ用ターゲットボードです。
TB-S3A3	0.6	S3A3 MCU グループ用ターゲットボードです。
TB-S3A1	0.5A 注	S3A1 MCU グループ用ターゲットボードです。
TB-S1JA	0.5A 注	S1JA MCU グループ用ターゲットボードです。
J-Link Software	6.30g	SEGGER J-Link®デバッグプローブ

注：量産前のバージョンのキットを使用して SSP をテストしています。そのため、量産版のキットの結果と異なる可能性があります。

2. ThreadX® RTOS と X-Ware

2.1 ThreadX と X-Ware の概要

SSP は、ThreadX と X-Ware (NetX/NetX Duo、USBX、FileX、GUIX、および Windows PC 側のツールである TraceX® と GUIX Studio™) を組み込んでいます。以下に、各コンポーネントのバージョンを示します。

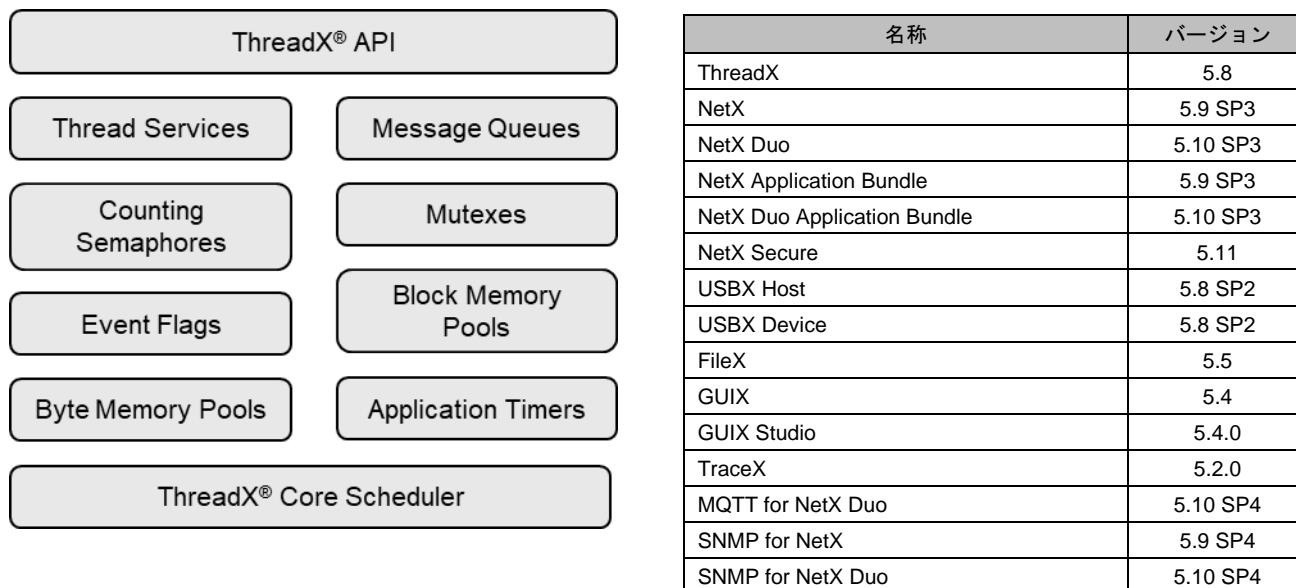


図 2.1 ThreadX の特長と各コンポーネントのバージョン

2.1.1 SSP v1.4.0 で利用可能なモジュールの一覧

サポートするモジュールは、次の基準に従ってそれぞれ対応する MCU で利用可能です。

1. モジュールの主要機能がテスト済みで、MCU 上で動作する場合、既知のバグが存在しても、そのモジュールは MCU でサポートされます。
2. 主要機能が MCU 上で正しく動作しないまたはテストされていない場合、そのモジュールは該当の MCU でサポートされません。
3. モジュールがいずれかの MCU でテスト済みで、基盤となるハードウェアまたは HAL ドライバーと独立している場合、そのモジュールはすべての MCU について、そのモジュールが使用する基礎となるドライバー、フレームワーク、スタックがその MCU 上で完全にテスト済みなら、サポートされます。

Synergy ソフトウェア X-Ware スタック	機能	サポート対象の MCU
fx	FileX	S7G2、S5D9、S3A7、S3A3、S3A6、S124
gx	GUIX	S7G2、S5D9
nx	NetX	S7G2、S5D9
nx_auto_ip	NetX Auto IP	S7G2、S5D9 注
nx_bsd	NetX BSD	S7G2、S5D9 注
nx_dhcp_client	NetX DHCP Client	S7G2、S5D9 注
nx_dhcp_server	NetX DHCP Server	S7G2、S5D9 注
nx_dns_client	NetX DNS Client	S7G2、S5D9 注
nx_ftp_client	NetX FTP Client	S7G2、S5D9 注
nx_ftp_server	NetX FTP Server	S7G2、S5D9 注
nx_http_client	NetX HTTP Client	S7G2、S5D9 注
nx_http_server	NetX HTTP Server	S7G2、S5D9 注
nx_pop3	NetX POP3	S7G2、S5D9 注

Synergy ソフトウェア X-Ware スタック	機能	サポート対象の MCU
nx_ppp	NetX PPP	S5D9、S7G2 注
nx_smtp_client	NetX SMTP Client	S7G2、S5D9 注
nx_snmp_agent	NetX SNMP Agent	S7G2、S5D9 注
nx_snmp_client	NetX SNMP Client	S7G2、S5D9 注
nx_telnet_client	NetX Telnet Client	S7G2、S5D9 注
nx_telnet_server	NetX Telnet Server	S7G2、S5D9 注
nx_tftp_client	NetX TFTP Client	S7G2、S5D9 注
nx_tftp_server	NetX TFTP Server	S7G2、S5D9 注
nxd	NetX Duo Stack	S7G2、S5D9 注
nxd_auto_ip	NetX Duo Auto IP	S7G2、S5D9 注
nxd_bsd	NetX Duo BSD	S7G2、S5D9 注
nxd_dhcp	NetX Duo DHCP IPv4 Client	S7G2、S5D9 注
nxd_dhcp	NetX Duo DHCP IPv6 Client	S7G2、S5D9 注
nxd_dhcp_server	NetX Duo DHCP IPv4 Server	S7G2、S5D9 注
nxd_dhcp_server	NetX Duo DHCP IPv6 Server	S7G2、S5D9 注
nxd_dns	NetX Duo DNS Client	S7G2、S5D9 注
nxd_ftp_client	NetX Duo FTP Client	S7G2、S5D9 注
nxd_ftp_server	NetX Duo FTP Server	S7G2、S5D9 注
nxd_http_client	NetX Duo HTTP Client	S7G2、S5D9 注
nxd_http_server	NetX Duo HTTP Server	S7G2、S5D9 注
nxd_nat	NetX Duo NAT	S7G2、S5D9 注
nxd_pop3	NetX Duo POP3	S7G2、S5D9 注
nxd_ppp	NetX Duo PPP	S7G2 注、S5D9 注
nxd_smtp_client	NetX Duo SMTP Client	S7G2、S5D9 注
nxd_snmp_agent	NetX Duo SNMP Agent	S7G2、S5D9 注
nxd_snmp_client	NetX Duo SNMP Client	S7G2、S5D9 注
nxd_telnet_client	NetX Duo Telnet Client	S7G2、S5D9 注
nxd_telnet_server	NetX Duo Telnet Server	S7G2、S5D9 注
nxd_tftp_client	NetX Duo TFTP Client	S7G2、S5D9 注
nxd_tftp_server	NetX Duo TFTP Server	S7G2、S5D9 注
nxd_mqtt_client	NetX Duo MQTT Client	S7G2、S5D9
nxd_tls_secure	NetX Duo TLS Secure	S7G2、S5D9
tx	ThreadX	S7G2、S5D9、S5D5、S3A7、S3A3、S3A6、S3A1、S124、S128、S1JA
ux_device_class_storage	USBX Device Class Mass Storage	S7G2、S5D9、S3A7、S3A3、S3A6、S3A1、S124、S128、S1JA
ux_device_class_hid	USBX Device Class HID	S7G2、S5D9、S3A7、S3A3、S3A6、S124、S128
ux_device_class_cdc_acm	USBX Device Class CDC-ACM	S124、S128、S3A3、S3A6、S3A7、S5D9、S7G2
ux_host_class_cdc_acm	USBX Host Class CDC-ACM	S7G2、S5D9、S3A7、S3A3、S3A6
ux_host_class_hid	USBX Host Class HID	S7G2、S5D9、S3A7、S3A3
ux_host_class_hub	USBX Host HUB	S7G2、S5D9
ux_host_class_storage	USBX Host Class Mass Storage	S7G2、S5D9、S3A7、S3A3

注：NetX および NetX Duo アプリケーションは MCU と独立のアプリケーションレイヤプロトコルです。NetX およびイーサネットドライバに依存しており、これらのプロトコルは NetX がテストおよび検証済みのすべての MCU でサポートされます。

実験・試作モジュールは、該当するモジュールまたは機能が MCU 上でテストされていないモジュールです。次に示す USBX クラスモジュールは、SSP の実験的なソフトウェアモジュールで、完全なテストがまだ行われておらず、現在のところ Synergy コンフィギュレーションツールのサポート対象外です。また、ユーザ側のプロジェクトでこれらのモジュールの使用は弊社のサポート対象外です。

実験・試作モジュール		
ux_device_class_cdc_ecm	USBX デバイスクラス CDC-ECM	S7G2、S5D9、S3A7、S3A3、S3A1、S124
ux_device_class_rndis	USBX デバイスクラス RNDIS	S7G2、S5D9、S3A7、S3A3、S124
ux_host_class_gser	USBX ホストクラスジェネリックシリアル	S7G2、S5D9、S3A7、S3A3
ux_host_class_printer	USBX ホストクラスプリンタ	S7G2、S5D9、S3A7、S3A3
ux_host_class_prolific	USBX ホストクラスプロリフィック	S7G2、S5D9、S3A7、S3A3
ux_host_class_swar	USBX ホストクラススウェア	S7G2、S5D9、S3A7、S3A3
ux_network_driver	USBX ネットワークドライバー	S7G2、S5D9、S3A7、S3A3、S124

2.2 ThreadX®コンポーネントの概要

SSP のコアとなるのは、全世界で採用実績が豊富な Express Logic 社の ThreadX RTOS です。ThreadX RTOS は、Synergy マイクロコントローラ向けに最適化されており、SSP と密接に統合されています。最適化された ThreadX RTOS はハイパフォーマンスのリアルタイムカーネル機能を搭載しています。とくに、リアルタイム処理を要求する組み込みアプリケーション向けには最適な設計が施されており、高速なコンテキストスイッチを実現、2 KB と小さなメモリフットプリントを提供します。

ThreadX の主な機能は以下のとおりです。

- picokernel™設計：カーネルの機能を階層的に重ねるのではなく、直接コアに接続してパフォーマンスの低下を避けるアーキテクチャを採用
- プリエンプティブスケジューリングとプリエンプティブスレッショルドスケジューリング
- イベントチューニング
- スレッド間の同期
- 最適化された割り込み処理：プリエンプションが不要の場合、割り込みサービスルーチン (ISR) の入出力時にスクラッチレジスタだけが保存/復元される
- 高速な割り込み応答時間
- 高速なコンテキストスイッチ
- 負荷のかからない RTOS サービス
- スタックポインタオーバーフローモニター

ThreadX が提供するメモリ保護機能により、アプリケーションスレッドと ThreadX カーネルは、他のスレッドによる予期しないリード・ライトアクセスから保護されます。これによって、アプリケーションの潜在的なバグによるコードやデータの破損を防ぐことができ、アプリケーションのクラッシュの最もよくある原因の 1 つを解消できます。

2.2.1 ThreadX の認定取得

ThreadX は、TUV と UL から、IEC 61508 SIL 4、IEC 62304 Class C、ISO 26262 ASIL D、UL/IEC 60730、UL/IEC 60335、UL 1998、および SW-SIL EN 50128 の事前認定を取得しています。

2.2.2 ThreadX API

直感的で一貫した API 命名規則 (すべての API には、ThreadX コールとしてのコールを簡単に識別するため "tx_" を付けています) によってグループ化された ThreadX API により、マルチスレッドのリアルタイム処理が必要な IoT 向けのアプリケーションを構築することができます。

ブロッキング API にはデッドハングを無効にするオプションのスレッドタイムアウト機能があり、多くの API はアプリケーション割り込みサービスルーチン (ISR) から直接利用できます。

2.2.3 スレッドサービス

スレッドサービスを動的に作成することで、スレッドサービスはスレッド数を無制限にすることができます（利用可能なハードウェアリソースとリアルタイム要求に基づく）。

2.2.4 メッセージキュー

スレッドサービスと同様に、メッセージキューは動的キューの作成を可能にします。メッセージキューの数に制限はありません（利用可能なハードウェアリソースとリアルタイム要求に基づく）。メッセージは、値によって、またはポインタを介して参照によってコピーできます。メッセージサイズは、32ビットワードで1~16個（両方を含む）です。オプションのスレッドサスペンド（空のキューによる中断とキューが満杯のときの中断）、オプションのタイムアウト（すべてのサスペンド時）は、ロックアップを回避するのに役立ちます。

2.2.5 カウンティングセマフォ

動的セマフォの作成とセマフォの数に制限がないため、これらの32ビットセマフォはスレッド間の連携サービスを提供します。カウンティングセマフォを生産者-消費者方式でイベント通知に使用することもできます。セマフォが利用できない場合の任意のスレッドの中断および一時停止時のオプションのタイムアウトは堅牢性を向上させます。

2.2.6 ミューテックス

スレッド間通信の同期のもう1つの形式では、（使用可能なハードウェアリソースに基づいて）持つことができるミューテックスの数に制限がなく、ダイナミックミューテックスの作成が可能なため、ネストされたリソース保護をサポートします。オプションの優先継承がサポートされています。ミューテックスが利用できない場合のオプションのスレッドサスペンドもサポートされています。

2.2.7 イベントフラグ

他のThreadXリソース、ダイナミックイベントフラググループの作成、イベントフラググループの制限なし（常に利用可能なハードウェアリソースに基づく）により、イベントフラグは1つのスレッドまたは複数のスレッドの同期を可能にします。AND/OR イベントセットのマルチスレッドサスペンションオプションとすべてのサスペンションのタイムアウトオプションと同様に、アトミックな「get」と「clear」がサポートされています。

2.2.8 ブロックメモリプール

動的ブロックプールの作成やブロックプールの数に制限はありません（物理メモリの制限を除く）。固定サイズのブロックのサイズやプールのサイズに制限はありません。可能なかぎり速いメモリ割り当て/解放がサポートされています。空のプールにオプションのスレッドサスペンション、すべてのサスペンドでオプションのタイムアウトなどの機能が利用できます。

2.2.9 バイトメモリプール

動的なバイトプールの作成および（物理メモリの制限を除く）バイトプールの数には制限がないため、管理されるバイトプールの数に制限はありません。これは最も柔軟な可変長メモリ割り当て/解放であり、割り当てサイズローカリティがサポートされています。空きプールにオプションのスレッドサスペンションが含まれ、すべてのサスペンド時にオプションのタイムアウトが使用可能です。

2.2.10 アプリケーションタイマ

ThreadXは、動的タイマの作成、制限のないタイマ数、周期タイマまたはワンショットタイマのサポートを提供します。周期タイマは、異なる初期有効期限値を有することができ、タイマの起動または非アクティ

ブ化に関するサーチはありません。

2.2.11 ThreadX コアスケジューラ

ThreadX は最小 2KB のフラッシュフットプリント、1KB の RAM フットプリント機能を提供しますが、スケジューラの最も重要な機能は、非常に高速でサブマイクロ秒のコンテキストスイッチ機能です。スレッドの数に関係なく完全に決定論的なこの優先順位ベースの完全優先スケジューラは、32 のデフォルト優先レベル（オプションで最大 1024）を持ちます。プリエンプティブスケジューリングに加えて、プライオリティレベル（FIFO）内で協調スケジューリングを実行することもできます。プリエンプシヨンスレッシュホールドの技術はスレッドの逆転を防止します。オプションの時間サービスには以下が含まれます。

- スレッド単位のオプションのタイムスライス
- ハードウェア時間割り込みにおけるすべてのブロッキング API 要求でのタイムアウトオプション

アプリケーション全体のトレースだけでなく、アプリケーションのチューニングに役立つ実行プロファイリングも提供されます。

3. SNMP エージェント

NetX および NetX Duo SNMP エージェントモジュールには、SSP 用の SNMP エージェントを実装するハイレベルの API が用意されています。このモジュールは、SSP+X-Ware に含まれている NetX および NetX Duo アプリケーションバンドルの一部です。特定の MCU に依存するものではなく独立しているモジュールのため、NetX および NetX Duo をサポートしているすべての MCU に SNMP エージェントを実装できます。

なお、とくに明記されていない限り、NetX プロジェクトと NetX Duo プロジェクトにおけるこのモジュールの動作に違いはありません。サポート対象の機能は以下のとおりです。

- NetX および NetX Duo SNMP エージェントモジュールは、RFC1155、RFC1157、RFC1215、RFC1901、RFC1905、RFC1906、RFC1907、RFC1908、RFC2571、RFC2572、RFC2574、RFC2575、RFC 3414、および関連する RFC に準拠しています。
- SNMP エージェントは UDP でのみ動作します。TCP はサポートされていません。
- SNMP エージェントモジュールでは、Transport Layer Security (TLS) と Datagram Transport Layer Security (DTLS) はサポートされていません。
- NetX および NetX Duo SNMP プロトコルは、SNMP バージョン 1、2、3 を実装します。SNMPv3 の実装では、MD5、SHA-1 認証、DES 暗号化がサポートされます。このバージョンの NetX および NetX Duo SNMP エージェントには、以下の制約があります。
 - 1 つの NetX IP インスタンスに対して 1 つの SNMP エージェント
 - RMON はサポート対象外
 - SNMP v3 のインフォームメッセージはサポート対象外
- SNMP エージェントの作成時に、username、get、set、getnext を処理するためのコールバックを登録するメカニズムが提供されます。

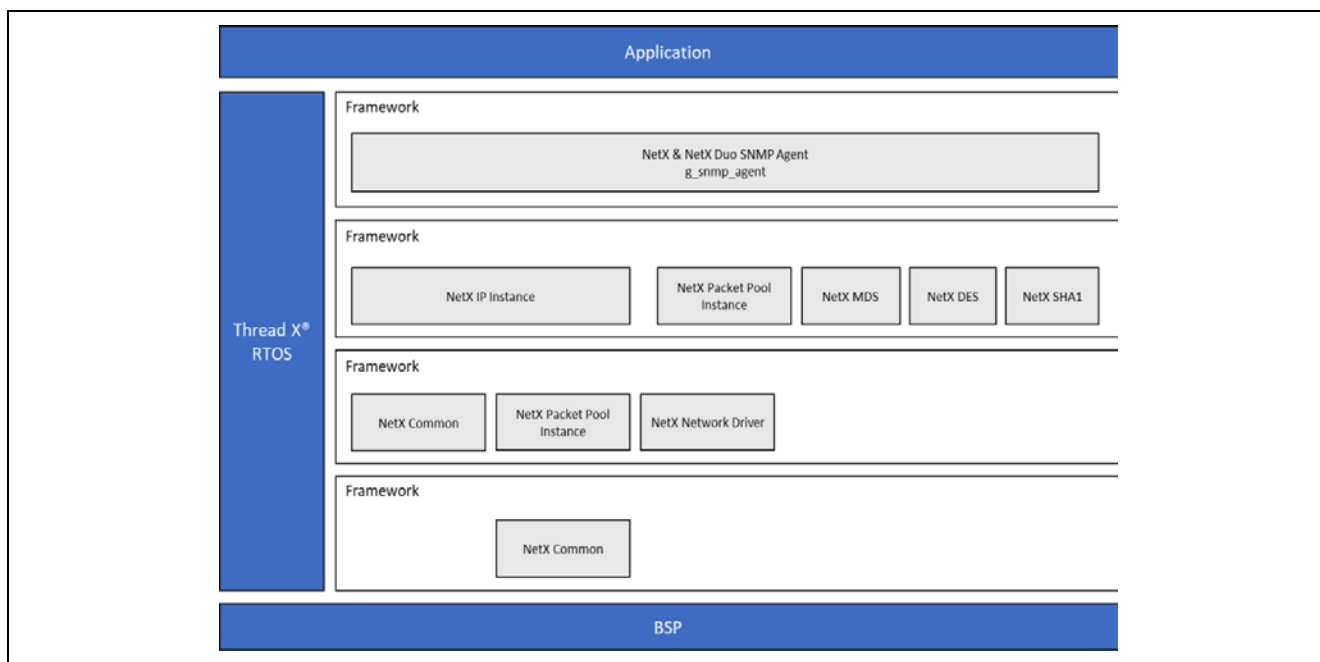


図 3.1 SNMP エージェント

4. NetX 組み込み TCP/IP および UDP スタック

SSP には、高度に最適化された組み込み TCP/IP IPv4 準拠プロトコルスタックである NetX が含まれています。このコンポーネントによって、IoT/M2M 通信プロトコルやネットワーク接続を必要とする組み込みアプリケーションに対応できます。NetX は ThreadX と完全に統合され、Express Logic 社独自の Piconet™アーキテクチャに基づいて設計されており、アプリケーションにゼロコピーAPI インタフェースを提供します。

NetX の主な機能は以下のとおりです。

- 高速処理
- TraceX システム分析サポート
- BSD ソケット互換 API
- UDP 高速パスにより基本的な UDP パケットは、コピーやコンテキストスイッチなしで NetX をパス
- 柔軟なパケットプール管理

NetX は TCP/IP 標準を構成するプロトコルコンポーネントの完全なセットを提供します。

- RFC791 インターネットプロトコル (IP)
- RFC826 アドレス解決プロトコル (ARP)
- RFC903 逆アドレス解決プロトコル (RARP)
- RFC792 インターネット制御通知プロトコル (ICMP)
- RFC3376 インターネットグループ管理プロトコル (IGMP)
- RFC768 ユーザーデータグラムプロトコル (UDP)
- RFC793 伝送制御プロトコル (TCP)
- RFC1112 IP マルチキャスト用ホスト拡張機能

SSP では、エンジニアに対して NetX のコード作成をサポートするコンフィギュレータが、e² studio ISDE と IAR EW for Synergy の両方に搭載されています。

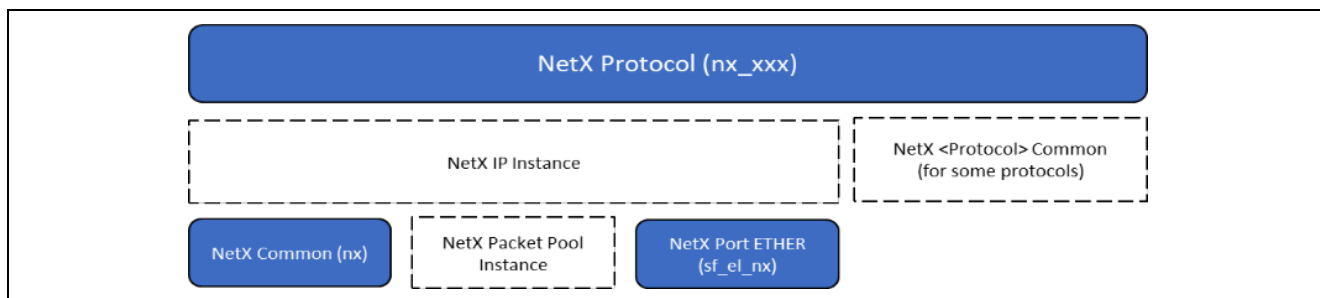


図 4.1 NetX スタックコンフィギュレーション

5. NetX Duo デュアル IPv4/IPv6 スタック

IPv6 サポートを必要とするアプリケーション用に、SSP には NetX Duo (デュアル IPv4/IPv6 準拠 TCP/IP プロトコルスタック) が含まれています。NetX Duo は ThreadX RTOS と完全に統合されており、NetX で利用可能な機能をすべて完備しています。これにより、SSP ベースデバイスの機能がさらに拡張され、ステートレスアドレス自動構成プロトコルを使用してインタフェースアドレスを自動構成できるようになります。各デバイスは階層構造を使用して、IPv6 ヘッダーをより効率的に処理できるようになります。NetX Duo アプリケーションはプロジェクトごとに個別に選択可能なため、システム設計者がターゲットアプリケーションに必要なアプリケーションだけを組み込めるという柔軟性がもたらされます。

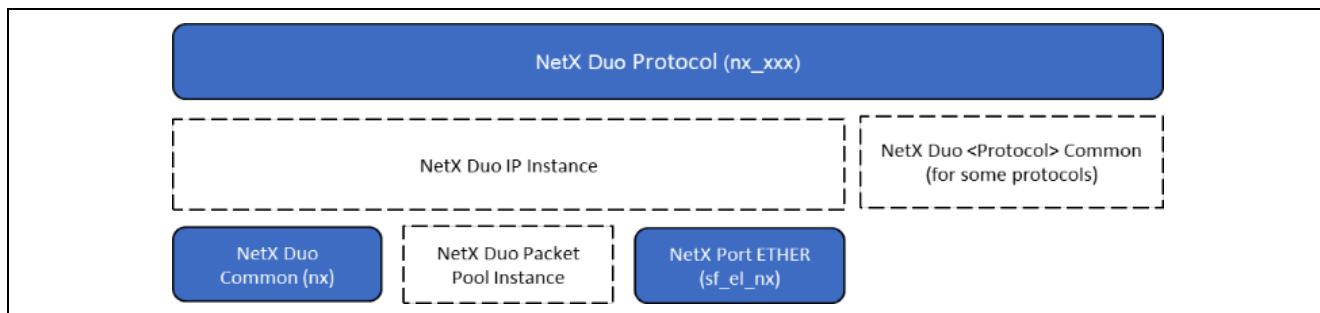


図 5.1 NetX Duo スタックコンフィギュレーション

NetX Duo は次のプロトコルを実装します。

- NetX で使用可能な全 IPv4 プロトコル
- ゼロコピー API
- UDP 高速パステクノロジ
- BSD ソケット互換 API
- RFC2460 IPv6 仕様
- RFC4861 IPv6 近隣探索
- RFC4862 IPv6 ステートレスアドレス自動構成
- RFC1981 IPv6 経路 MTU 探索
- RFC4443 ICMPv6
- RFC791 インターネットプロトコル (IP)
- RFC826 アドレス解決プロトコル (ARP)
- RFC903 逆アドレス解決プロトコル (RARP)
- RFC792 インターネット制御通知プロトコル (ICMP)
- RFC3376 インターネットグループ管理プロトコル (IGMP)
- RFC768 ユーザーデータグラムプロトコル (UDP)
- RFC793 伝送制御プロトコル (TCP)
- RFC1112 IP マルチキャスト用ホスト拡張機能

NetX Duo は IPv6 フォーラムによって認定され、Phase-II IPv6-Ready Logo 認証を取得しています。NetX Duo は、TUV と UL から、IEC 61508 SIL 4、IEC 62304 Class C、ISO 26262 ASIL D、UL/IEC 60730、UL/IEC 60335、UL 1998、および EN 50128 SW-SIL 4 の事前認定を取得しています。

6. NetX アプリケーションバンドル

NetX には、ネットワークデバイスでよく使用される追加のアプリケーションレイヤープロトコルが含まれています。

- Auto IP
 - RFC 3927 IPv4 リンクローカルアドレスの動的構成
- サーバー (DHCP サーバー) およびクライアント (DHCP クライアント) 向け動的ホスト構成プロトコル
 - RFC 2131 動的ホスト構成プロトコル
 - RFC 2132 DHCP オプションおよび BOOTP ベンダー拡張機能
- ドメイン名システム (DNS クライアント)
 - RFC1034 ドメイン名：概念と機能
 - RFC1035 ドメイン名：実装と仕様
 - RFC1480 US ドメイン
 - RFC2782 サービスの場所を指定するための DNS RR (DNS SRV)
- HTTP クライアントおよび Web サーバー
 - RFC1945 ハイパーテキストトランスファープロトコル/1.0
 - RFC2581 TCP 輻輳制御
 - RFC1122 インターネットホストに関する要件：通信レイヤー
- RFC959 ファイルトランスファープロトコル (FTP) クライアントおよびサーバー
- TFTP クライアントおよびサーバー
 - RFC1350 TFTP プロトコル (リビジョン 2)
- Telnet クライアントおよびサーバー
 - RFC 854 Telnet プロトコル仕様
- RFC1939 ポストオフィスプロトコルバージョン 3 (POP3)
- RFC1661 ポイントツーポイントプロトコル (PPP)
 - RFC1332 PPP インターネットプロトコル制御プロトコル (IPCP)
 - RFC1334 PPP 認証プロトコル
 - RFC1994 PPP チャレンジハンドシェイク認証プロトコル (CHAP)
- シンプルメールトランスファープロトコル (SMTP)
 - RFC2821 シンプルメールトランスファープロトコル (SMTP)
 - RFC2554 SMTP 認証用サービス拡張機能
- RFC4330 シンプルネットワークタイムプロトコル (SNTP) バージョン 4、IPv4/IPv6/OSI 向け

これらのコアネットワークプロトコルの実装はスレッドセーフであり、各 RFC 標準に準拠し、メモリフットプリントと CPU 使用率に対し最適化されています。ネットワークアプリケーションは各プロジェクトでそれぞれ選択可能です。エンジニアはターゲットアプリケーションのために必要なアプリケーションのみを柔軟に組み込むことができます。

7. NetX Duo アプリケーションバンドル

NetX Duo には、ネットワークデバイスでよく使用される追加のアプリケーションレイヤープロトコルが含まれています。

- Auto IP
 - RFC 3927 IPv4 リンクローカルアドレスの動的構成
- サーバー (DHCP サーバー) およびクライアント (DHCP クライアント) 向け動的ホスト構成プロトコル
 - RFC 2131 動的ホスト構成プロトコル
 - RFC 2132 DHCP オプションおよび BOOTP ベンダー拡張機能
- ドメイン名システム (DNS クライアント)
 - RFC1034 ドメイン名：概念と機能
 - RFC1035 ドメイン名：実装と仕様
 - RFC1480 US ドメイン
 - RFC2782 サービスの場所を指定するための DNS RR (DNS SRV)
- HTTP クライアントおよび Web サーバー
 - RFC1945 ハイパーテキストトランスファープロトコル/1.0
 - RFC2581 TCP 輻輳制御
 - RFC1122 インターネットホストに関する要件：通信レイヤー
- RFC959 ファイルトランスファープロトコル (FTP) クライアントおよびサーバー
- TFTP クライアントおよびサーバー
 - RFC1350 TFTP プロトコル (リビジョン 2)
- Telnet クライアントおよびサーバー
 - RFC 854 Telnet プロトコル仕様
- RFC1939 ポストオフィスプロトコルバージョン 3 (POP3)
- RFC1661 ポイントツーポイントプロトコル (PPP)
 - RFC1332 PPP インターネットプロトコル制御プロトコル (IPCP)
 - RFC1334 PPP 認証プロトコル
 - RFC1994 PPP チャレンジハンドシェイク認証プロトコル (CHAP)
- シンプルメールトランスファープロトコル (SMTP)
 - RFC2821 シンプルメールトランスファープロトコル (SMTP)
 - RFC2554 SMTP 認証用サービス拡張機能
- RFC4330 シンプルネットワークタイムプロトコル (SNTP) バージョン 4、IPv4/IPv6/OSI 向け

これらのコアネットワークプロトコルの実装はスレッドセーフであり、各 RFC 標準に準拠し、メモリフットプリントと CPU 使用率に対し最適化されています。ネットワークアプリケーションは各プロジェクトでそれぞれ選択可能です。エンジニアはターゲットアプリケーションのために必要なアプリケーションのみを柔軟に組み込むことができます。

8. NetX Secure および MQTT for NetX Duo

8.1 NetX Secure

NetX Secure は、TLS (Transport Layer Security) などの暗号化ネットワークセキュリティ標準をリアルタイムに実装できる高性能ツールであり、ThreadX ベースの組み込み Renesas Synergy アプリケーション専用で設計されています。NetX Secure は、Amazon Web Services™ (AWS) IoT インタフェースなどのプライベートクラウドやパブリッククラウドと連携し、他の Renesas Synergy NetX/NetX Duo API に似たフォーマットと構造を持つ API を実装します。

NetX Secure では、TLS に関連する以下のプロトコルがサポートされています。TLS と暗号方式に関する RFC は数多く存在するため、このリストにすべての RFC が含まれているとは限りません。NetX Secure は、メモリフットプリントが小さく効率的に実行できるリアルタイム OS という制約の範囲内で、すべての一般的な推奨事項と基本要件に従っています。

- Transport Layer Security (TLS) プロトコル
 - RFC 2246、TLS プロトコルバージョン 1.0
 - RFC 5246、Transport Layer Security (TLS) プロトコルバージョン 1.2
- X.509 PKI 証明書
 - RFC 5280、X.509 PKI 証明書バージョン 3
- TLS 暗号方式
 - RFC 3268、Transport Layer Security (TLS) 向け Advanced Encryption Standard (AES) 暗号スイート
 - RFC 3447、公開鍵暗号化標準 (PKCS) #1: RSA 暗号仕様バージョン 2.1
 - RFC 2104、HMAC: メッセージ認証用キー付きハッシュ
 - RFC 623、US セキュアハッシュアルゴリズム (SHA および SHA ベースの HMAC と HKDF)

8.1.1 NetX Secure の要件

NetX Secure 実行時ライブラリを適切に機能させるためには、NetX IP インスタンスがすでに作成されている必要があります。また、アプリケーションによっては、TLS インスタンスの識別用またはリモートホストから取得する証明書の検証用に、1つ以上の DER エンコード済み X.509 デジタル証明書が必要になります。

NetX Secure は、ThreadX と NetX/NetX Duo の存在を前提としています。ThreadX に対して要求されるのは、スレッド実行、サスペンション、定期タイマ、相互排除の各機能です。NetX/NetX Duo に対して要求されるのは、TCP/IP ネットワーク機能およびドライバーです。

8.1.2 NetX Secure の制約とサポート

NetX Secure プロトコルは、RFC 5246 の要件と TLS 1.2 向けの標準、および RFC 2246 (TLS 1.0) との後方互換性を実装します。NetX Secure には、以下の制約とサポートレベルが適用されます。

1. 内蔵デバイスの特性のため、アプリケーションによっては、最大 TLS レコードサイズである 16 KB をサポートするためのリソースを確保できない場合があります。NetX Secure は、十分なリソースのあるデバイスでは 16 KB のレコードを処理できます。
2. NetX Secure は、証明書に対して基本的な検証を実行して、証明書が有効であり、信頼できる証明機関によって署名されていることを確認します。また、リモートホストの最上位のドメイン名と比較して証明書共通名を検証することができます。ただし、証明書の拡張機能および他のデータの検証は、アプリケーションの実装者が行う必要があります。
3. 正式な SSL プロトコルはどのバージョンも古く、安全性が低いとみなされており、現在の NetX Secure では SSL の実装を提供していません。
4. NetX Secure TLS を使用する前に、NetX または NetX Duo TCP/IP スタックを初期化する必要があります。TCP/IP スタックを適切に初期化する方法については、NetX または NetX Duo ユーザーガイドを参照してください。
5. サーバーで TLS が正しく実装されていない場合、NetX Secure TLS はデフォルトで最新バージョンの TLS (1.2) にだけ接続されます。

6. パディングエラーを通知するために `bad_record_mac` を使用するサーバー実装がサポートされています。ただし、パディングエラーの有無に関係なく、NetX Secure TLS では MAC の計算がサポートされています。
7. サーバーの実装では、ライブラリベースのソフトウェア暗号システムによる定数時間での復号化をサポートしている必要があります（つまり、ハードウェア復号化のサポートは不要です）。NetX Secure では、現在のところ、定数時間での復号化はサポートされていません。ハードウェアによる暗号化方式では、定数時間での復号化がサポートされています。
8. 現在、サーバー証明書とそのチェーンは無条件に送信されています。
9. サーバーの実装では、クライアントが証明書または許容可能な証明書を所有していない場合に致命的なハンドシェイク障害アラートとともに接続を終了できることが必要であり、その機能がサポートされています。
10. 証明書失効リスト (CRL) はサポートされていますが、オンライン証明書ステータス確認プロトコル (OCSP) は現時点ではサポートされていません。
11. 基本的な X.509 チェーン検証がサポートされており、証明書を受信するたびに、信頼できる証明書ストアとの照合による検証を行います。`user-initial-policy-set` には、X.509 の特殊値である "any-policy" が使用されます（この値は、アプリケーションで証明書ポリシーが考慮されないことを示しています）。
12. RFC 5280 の 6 節 (NIST 800-52 の 3.5.1 節を参照) には、証明書パス検証の定義が記載されています。6.1 節には、NetX Secure が一部を実装する基本的なパス認証の定義が記載されています。NIST 800-52 を満たすため、RFC 5280 の 6.1.3 および 6.1.4 節に指定されているとおり、証明書処理が拡張されてポリシーの適用が追加されます。基本ポリシーの適用を追加する必要があります。X.509 検証に失敗した場合は、TLS 接続が終了されます。
13. 不正な証明書が適切に拒否されるように、クライアントの実装で名前の制約チェックをサポートする必要がありますが、現時点でこの機能はサポートされていません。
14. クライアントの実装では、クライアントの TLS 要求に含まれる DNS 名や IP アドレスが、サーバー証明書のサブジェクト識別名フィールドまたはサブジェクト別名拡張に含まれる名前や IP アドレスと一致していることをチェックする必要があります。この機能は、現時点ではアプリケーションに依存していません。NetX Secure TLS アプリケーションで DNS が有効になっている場合は、名前のチェック時に、証明書内の DNS 名がリモートサーバーの IP アドレスと一致しているかどうかの検証が行われます。アプリケーションによっては DNS サポートが含まれていない場合もあるため、証明書検証コールバックで DNS と IP アドレスのチェックが行われます。

8.1.3 実装対象の暗号

NetX Secure では、以下の暗号がサポートされています。

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_PSK_WITH_AES_128_CBC_SHA
- TLS_PSK_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_NULL_MD5
- TLS_RSA_WITH_NULL_SHA

暗号化のサポート対象は、RSA、DES、3DES、AES、MD5、SHA1、SHA2 です。ハードウェアによって高速化した暗号化のサポート対象は、RSA、AES、SHA1、SHA2 です。

8.2 MQTT for NetX Duo

MQTT (Message Queue Telemetry Transport) は、ローパワーセンサーやその他 M2M または IoT 分野のアプリケーション向けに役立つ軽量のパブリッシュ/サブスクライブモデルです。MQTT for NetX Duo の実装は、2014 年 10 月 29 日の OASIS MQTT バージョン 3.1.1 に準拠します。MQTT は公開標準仕様で、ウェブサイト上に詳細な情報が掲載されています (<http://mqtt.org/>)。MQTT for NetX は、HTTP よりも制約のある環境に適しており、非同期通信向けの手法を提供するほか、インターネット (IP) を介して利用できます。

8.2.1 MQTT for NetX Duo の要件

MQTT for NetX Duo クライアントパッケージを適切に機能させるためには、NetX Duo (バージョン 5.10 以降) をインストールする必要があります。MQTT を使用するには、まずアプリケーションで IP インスタンスを適切に構成して開始する必要があります。TCP も有効化されていなければなりません。さらに、TLS セキュリティが必要な場合は、TLS を構成する必要があります。

9. FileX 組み込みファイルシステム

SSP は、ファイル操作を必要とする組み込みアプリケーションに、高いパフォーマンスと小さなメモリフットプリントの MS-DOS 互換ファイルシステムである FileX を提供します。FileX は C ライブラリとして実装されています。アプリケーションで使用する機能のみを最終イメージに組み込むことができます。FileX のメモリフットプリントは最小 6 KB 程度です。さらに、FileX は最小限の関数コールレイヤー、内部論理セクターキャッシュ、連続クラスタ割り当て、連続クラスタリード・ライト機能を備えているため、非常に高速で効率的なファイルシステムです。

FileX の主な機能は以下のとおりです。

- 無制限のメディアインスタンス (メディア、ディレクトリ、ファイル)
- FAT12/16/32 ビットサポート
- ロングファイルネームに対応
- 連続ファイルサポート
- 連続セクター・クラスタのリード・ライト
- 内部論理セクターキャッシュ
- 高速なシークロジック
- 複数パーティションサポート
- ジャーナリング機能を持つフォールトトレラントなファイルシステム

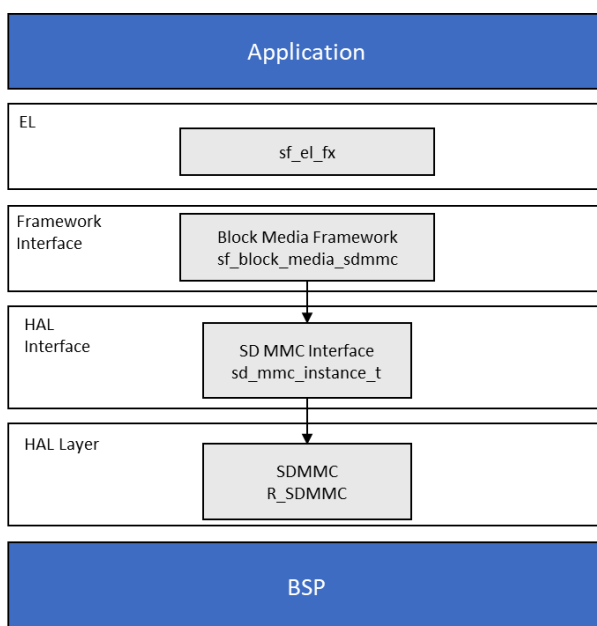


図 9.1 FileX 組み込みファイルシステム

10. GUIX GUI 開発ツールキット

GUIX は SSP のグラフィカルユーザーインターフェースフレームワークであり、高いパフォーマンスを発揮します。GUIX には、完全な機能を備えたランタイム UI ライブラリと、連携するデスクトップ PC 用の GUI 設計アプリケーションである GUIX Studio が含まれています。

GUIX の主な機能は以下のとおりです。

- 高い信頼性、重要アプリケーションのためのフェールセーフで安全な設計
- 無制限の GUIX オブジェクト（画面、ウィンドウ、ウィジェット）、使用可能メモリによる制限のみ
- 動的な GUIX オブジェクトの作成/削除
- 高色深度でのアルファブレンドとアンチエイリアスのサポート
- 完全なウィンドウ化のサポート、ビューポートと Z オーダーの保持
- 複数キャンバスと複数の物理ディスプレイ、ウィンドウのブレンドとフェード、画面遷移、スプライト、動的アニメーションのサポート
- ハードウェアによって高速化した JPEG および MJPEG デコーディング
- タッチスクリーンと仮想キーボードのサポート
- UTF8 文字列エンコーディングを使用した多言語対応
- ハードウェアの 2D グラフィックスアクセラレータをサポート
- 柔軟なメモリ使用
- オブジェクトサイズの自動拡大・縮小
- 小さなメモリフットプリント
- エンディアン中立
- ローテーション
- ハードウェアまたはソフトウェアレンダリングによる 8 ビットカラールックアップテーブル (CLUT)

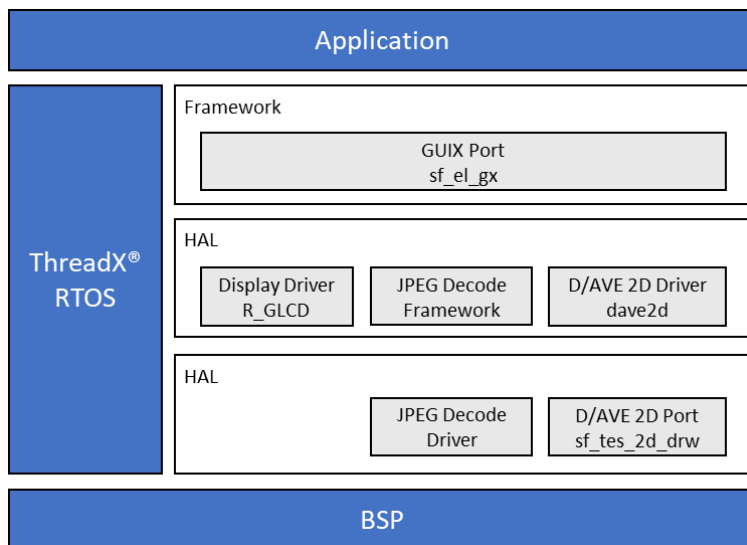


図 10.1 GUIX ランタイムライブラリ

11. USBX

SSPには、ThreadXに完全に統合された組み込みUSBスタックが含まれており、組み込みアプリケーションのためのUSBホスト、デバイスをサポートし、高いパフォーマンスを発揮します。USBXは小さなメモリフットプリントでモジュール化されているため、アプリケーションで使用する機能のみを最終イメージに含めることができます。これにより、ターゲットデバイスのためにビルドするUSBスタックのフットプリントを最小化できます。Low speed、Full speed、High speedの各モードをサポートします。

USBXの主な機能は以下のとおりです。

- マスストレージ、HID、CDC-ACMなど、標準USBクラスドライバーのほとんどをサポート。UVCとHUBは、USBホスト向けにサポートされる追加クラス
- USBアイソクロナス転送のサポート
- Express Logic社のコンポーネント（FileXおよびNetX）との統合
- MCU専用モードのビルドオプションによりコードサイズを縮小可能
- S124向けUSBXのプリビルドライブラリは初期設定ではデバイスモードに設定

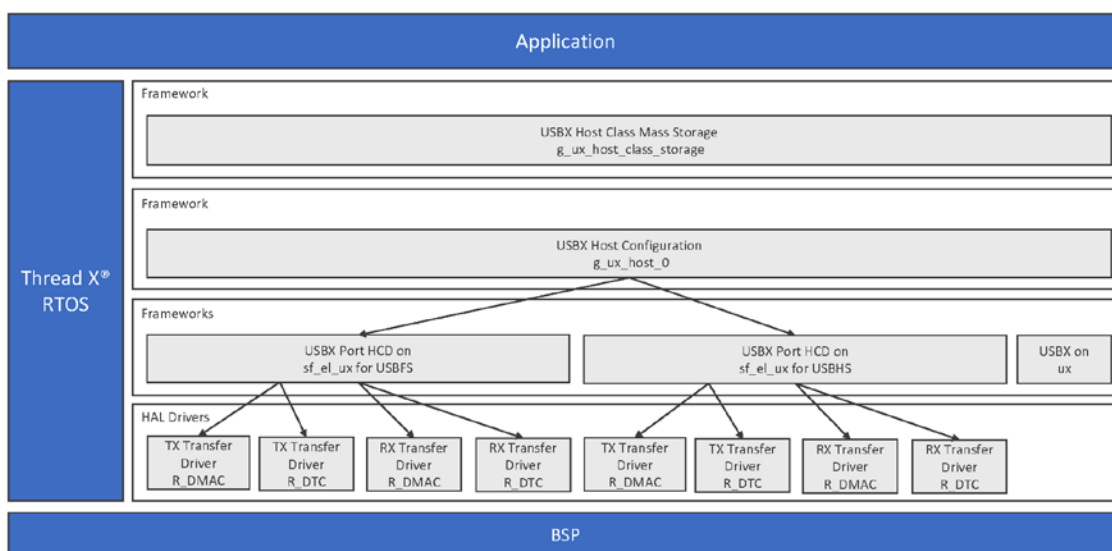


図 11.1 USBX マスストレージクラス (MSC) ホストスタック

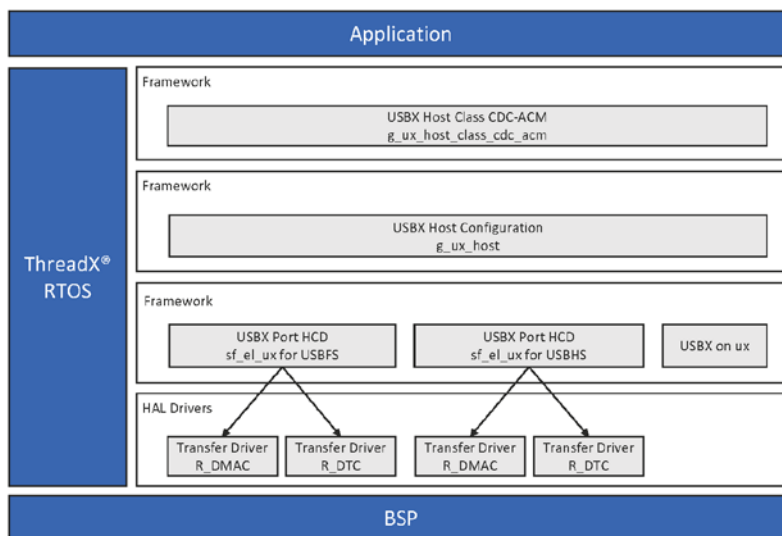


図 11.2 USBX CDC/ACM ホストスタック

グループ名	USBX マスストレージクラス				USBX HID クラス				USBX CDC/ACM			
	ホスト		デバイス		ホスト		デバイス		ホスト		デバイス	
	High speed	Full speed	High speed	Full speed	High speed	Full speed	High speed	Full speed	High speed	Full speed	High speed	Full speed
S1JA	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓
S124	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓
S128	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓	N/A	N/A	N/A	✓
S3A1	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓
S3A3	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓
S3A6	N/A	☒	N/A	✓	N/A	☒	N/A	✓	N/A	✓	N/A	✓
S3A7	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓
S5D5	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓	N/A	✓
S5D9	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

グループ名	USBX Video クラス		USBX HUB クラス	
	ホスト		ホスト	
	High speed	Full speed	High speed	Full speed
S1JA	N/A	N/A	N/A	N/A
S124	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A
S3A1	N/A	☒	N/A	N/A
S3A3	N/A	☒	N/A	N/A
S3A6	N/A	☒	N/A	N/A
S3A7	N/A	☒	N/A	N/A
S5D5	N/A	☒	N/A	✓注
S5D9	✓	☒	✓	✓
S7G2	✓	☒	✓	✓

注：HUB クラスがセルフパワーハブだけでテストされていることを示しています。

記号の説明

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

SSP では、エンジニアに対して USBX のコード作成をサポートするコンフィギュレータが、e² studio ISDE と IAR EW for Synergy の両方に搭載されています。

12. アプリケーションフレームワーク

12.1 概要

アプリケーションフレームワークは SSP の主要なサブシステムです。ハードウェア抽象化レイヤー (HAL) とともにハードウェア周辺機能を抽象化し、タスクを 1 か所で完了できるように複数のソフトウェア機能を 1 つにまとめます。アプリケーションフレームワークは、標準 API による一貫したプログラミングインタフェースを提供します。これによって、エンジニアはプラットフォームの複雑なデバイスインタフェースに煩わされることなく、SSP の主要な機能にアクセスして活用することができます。

これらのアプリケーションフレームワークによって、エンジニアは高度に抽象化された環境でプログラム開発を行うことができ、プログラミング作業を大幅に軽減できるほか、HAL レイヤーや BSP に直接アクセスできます。

アプリケーションフレームワークは ThreadX と密接に統合されており、スレッドセーフな API を提供します。共用リソースへのアクセスやアプリケーションスレッド間のアクセス競合を管理し、相互排他と同期のサービスを提供します。このことは、特に共用リソースを使用する場合に役立ちます。アプリケーションフレームワークによってタスクが論理 API と結合されるため、エンジニアは正確なコードを素早く作成できます。SSP のアプリケーションフレームワークは、RTOS と HAL を連携し、頻繁に使用されるプラットフォームシステムサービスのためのハイレベルな C コール可能なインタフェースを提供します。

フレームワーク API は、ルネサスがコントロールおよびエンジニアリングし、保守を行って一貫性を維持します。エンジニアは、必要に応じて HAL ドライバーに直接アクセスできます。HAL ドライバーは RTOS オブジェクトを使用せず、RTOS API コールを実行しません。しかし、HAL ドライバーを RTOS とともに (または RTOS なしで) 使用することはできます。

フレームワークレイヤーモジュールはセマフォ、ミューテックス、イベントフラグをはじめとするスレッド間通信ツールなどの RTOS オブジェクトを使用します。これらのモジュールは、必要に応じて独自のオブジェクトを作成し、HAL レイヤー経由で最下位層のハードウェアにアクセスします。

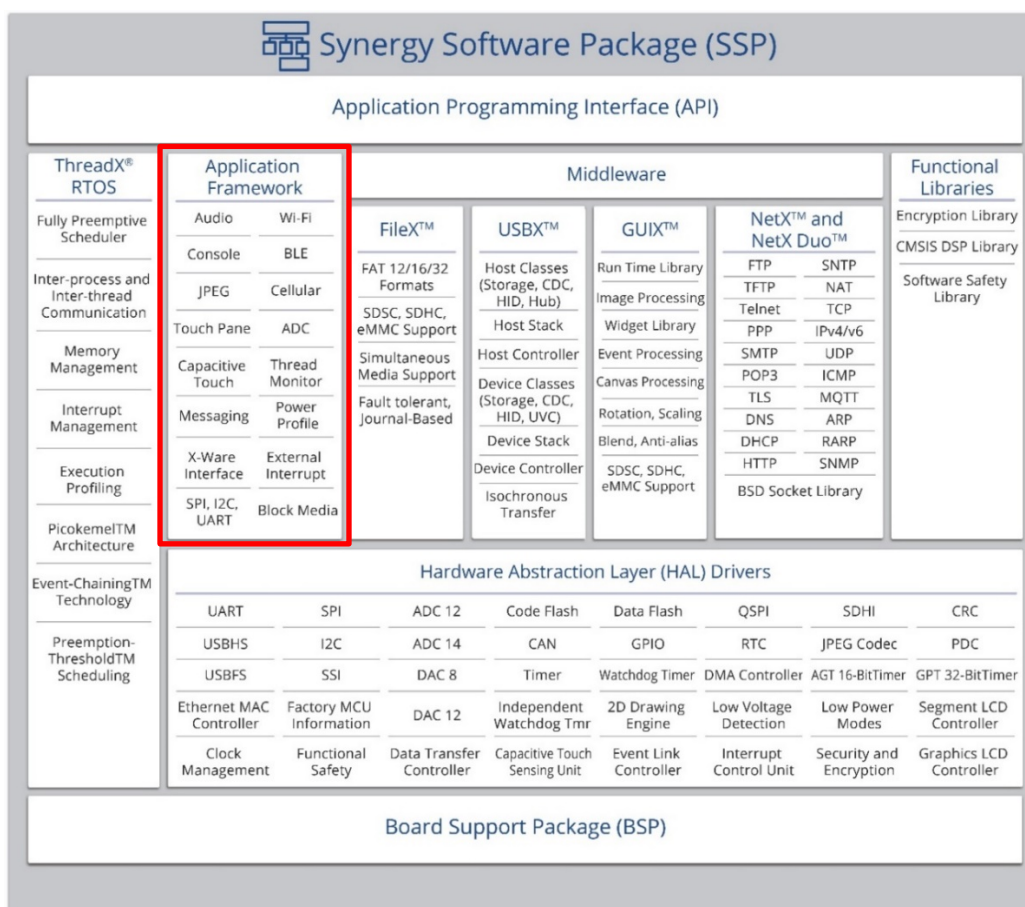


図 12.1 アプリケーションフレームワーク

12.2 SSP v1.4.0 で利用可能なソフトウェアフレームワークモジュールの一覧

サポートするモジュールは、次の基準に従ってそれぞれ対応する MCU で利用可能です。

1. モジュールの主要機能がテスト済みで、MCU 上で動作する場合、既知のバグが存在しても、そのモジュールは MCU でサポートされます。
2. 主要機能が MCU 上で正しく動作しないまたはテストされていない場合、そのモジュールは該当の MCU でサポートされません。
3. モジュールがいずれかの MCU でテスト済みで、基盤となるハードウェアまたは HAL ドライバーと独立している場合、そのモジュールはすべての MCU について、そのモジュールが使用する基礎となるドライバー、フレームワーク、スタックがその MCU 上で完全にテスト済みなら、サポートされます。

Synergy ソフトウェア フレームワーク	機能	サポート対象の MCU
sf_adc_periodic	Periodic Sampling ADC	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_audio_playback	Audio Playback	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_audio_playback_hw_dac	Audio Playback HW DAC	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_audio_playback_hw_i2s	Audio Playback HW I2S	S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_audio_record_adc	Audio Record HW ADC	S124, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_audio_record_i2s	Audio Record I2S	S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_ble_rl78g1d	BLE Framework	S124, S128, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_ble_rl78g1d_onboard_profile	BLE Framework Onboard Profiles	S124, S128, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_block_media_qspi	File System Support on QSPI	S3A3, S3A7, S5D5, S5D9, S7G2
sf_block_media_ram	File System Support on RAM	S3A3, S3A7, S5D9, S7G2
sf_block_media_sdmmc	Block Media Interface for SD Multi Media Card	S3A3, S3A7, S5D9, S7G2
sf_cellular_cat1	Cellular Framework CAT1	S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_cellular_cat1_socket	Cellular Framework CAT1 Socket	S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_cellular_cat3	Cellular Framework CAT3	S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_cellular_cat3_socket	Cellular Framework CAT3 Socket	S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_cellular_catm1	Cellular Framework CATM1	S5D9, S7G2
sf_cellular_catm1_socket	Cellular Framework CATM1 Socket	S5D9, S7G2
sf_comms_telnet	Synergy Telnet Communication Interface	S5D5, S5D9, S7G2
sf_console	Console	S124, S128, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_comms_telnet ^{注1}	Synergy Telnet Communication Interface	S5D5, S5D9, S7G2
sf_crypto ^{注1}	Crypto Framework Common	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_crypto_cipher ^{注1}	Crypto Framework Cipher	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_crypto_hash ^{注1}	Crypto Framework Hash	S5D5, S5D9, S7G2
sf_crypto_key ^{注1}	Crypto Framework Key	S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_crypto_key_installation ^{注1}	Crypto Framework Key Installation	S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_crypto_key_signature ^{注1}	Crypto Framework Key Signature	S5D5, S5D9, S7G2
sf_crypto_trng ^{注1}	Crypto True Random Number Generator	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_el_fx	Synergy FileX interface	S3A1, S3A3, S3A7, S5D9, S7G2
sf_el_gx	Synergy GUIX Interface	S7G2, S5D9
sf_el_nx	Synergy NetX Interface	S5D5, S5D9, S7G2

Synergy ソフトウェア フレームワーク	機能	サポート対象の MCU
sf_el_nx_comms	Synergy NetX Communication Interface	S5D5, S5D9, S7G2
sf_el_ux	Synergy USBX Interface	S1JA, S124, S128, S3A1, S3A3, S3A7, S5D5, S5D9, S7G2
sf_el_ux_comms ^{注2}	Synergy USBX Communication Interface	S1JA, S124, S128, S3A1, S3A3, S3A7, S5D5, S5D9, S7G2
sf_el_ux_comms_v2	Synergy USBX Communication Interface V2	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_external_irq	External Interrupt	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_i2c	I2C Framework	S124, S128, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_jpeg_decode	JPEG Decode	S5D9, S7G2
sf_message	Inter-Thread Messaging	S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_power_profiles ^{注2}	Power Mode Profile	S124, S3A7, S7G2
sf_power_profiles_v2	Power Mode Profile V2	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_spi	SPI Framework	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_tes_2d_drw	2D Drawing Engine Framework	S5D9, S7G2
sf_thread_monitor	Thread Monitor (Watchdog)	S1JA, S124, S128, S3A1, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_touch_ctsu	Capacitive Touch Sensing Unit	S124, S128, S3A7, S7G2
sf_touch_ctsu_button	Capacitive Touch Sensing Unit Button	S124, S128, S3A7, S7G2
sf_touch_ctsu_slider	Capacitive Touch Sensing Unit Slider	S124, S128, S3A7, S7G2
sf_touch_panel_i2c	Touch Panel I2C	S5D9, S7G2
sf_uart_comms	UART Framework	S124, S128, S3A3, S3A6, S3A7, S5D5, S5D9, S7G2
sf_wifi_gt202	WiFi Framework	S3A3, S3A7, S5D5, S5D9, S7G2
sf_wifi_gt202_onchip	WiFi framework on Chip Stack	S3A3, S3A7, S5D5, S5D9, S7G2

注1：暗号化機能の詳細については表 14.1 を参照してください。なお、表に示す暗号化機能は r_sce 暗号化ライブラリの一部として利用可能です。

注2：このモジュールは、SSP v1.3.0 以降のバージョンでは廃用となったもので、そのモジュールを使用している可能性がある既存のプロジェクトとの互換性を維持する目的でのみ利用可能です。新しいプロジェクトではこれら廃用となったモジュールを使用せず、代わりに推奨される代替品を使用することを強くお勧めします。詳細は「SSP v1.4.0 ユーザーズマニュアル」を参照してください。

12.3 周期サンプリング ADC フレームワーク

ハイレベル API を提供するフレームワークであり、`sf_adc_periodic` に実装されています。このモジュールは、使用可能な任意のチャンネルを指定したレートで（シングルスキャンモードを使用して）サンプリングするように ADC を構成し、指定したサンプリング繰り返し回数分のデータをバッファに格納してからアプリケーションに通知します。周期サンプリング ADC フレームワークは、MCU の ADC、GPT、AGT、および DTC 周辺機能を使用します。ユーザー定義のコールバックを作成し、新しいサンプルの取得が完了するたびにデータを処理することができます。

このフレームワークの主な機能は以下のとおりです。

- 該当する MCU 上の 14 ビット ADC および 12 ビット ADC をサポート
- 複数の動作モード（シングルスキャン、グループスキャン、連続スキャン）
- 複数のチャンネルをサポート
S7G2・S5D9：13 チャンネル（ユニット 0） および 12 チャンネル（ユニット 1）
S124：18 チャンネル
S3A7：28 チャンネル

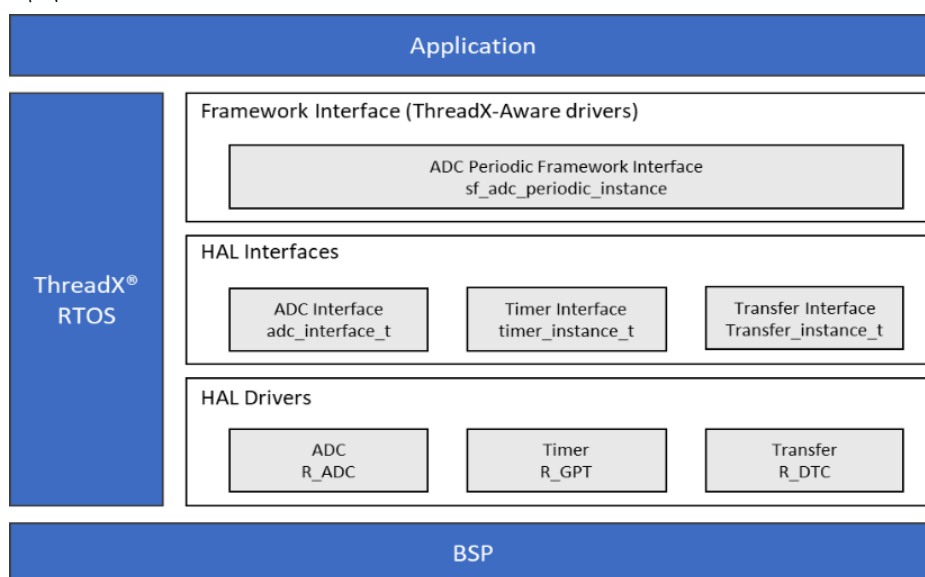


図 12.2 周期サンプリング ADC フレームワーク

12.4 オーディオ再生フレームワーク

オーディオ再生フレームワークは、オーディオ再生アプリケーション向けのハイレベルの API を提供し、sf_audio_playback_hw_dac または sf_audio_playback_hw_i2s に実装されています。このフレームワークは、16 および 8 ビットパルス符号変調 (PCM) サンプルを再生するために必要な同期を処理します。オーディオ再生フレームワークは、MCU 上の DAC または I2S、タイマ (AGT または GPT)、データ転送 (DMA または DTC) 周辺機能を使用します。ユーザー定義のコールバックを作成し、追加データが必要な場合に対応できます。

このフレームワークの主な機能は以下のとおりです。

- データを扱いやすいチャンクに分割することにより長いバッファを再生
- ThreadX タイムアウトが発生するまで再生をリピート (正弦波音やループするバックグラウンドミュージックのように音声が続けられる場合)
- 最後のバッファ再生が開始した後、コールバックを使用して次のデータを要求
- ソフトウェアのボリュームを制御
- 一時停止と再開機能
- 符号付き 16 ビット PCM データを符号なし 12 ビット DAC の範囲にシフトするためのスケールリング
- 複数のストリームについての基本的なミキシング

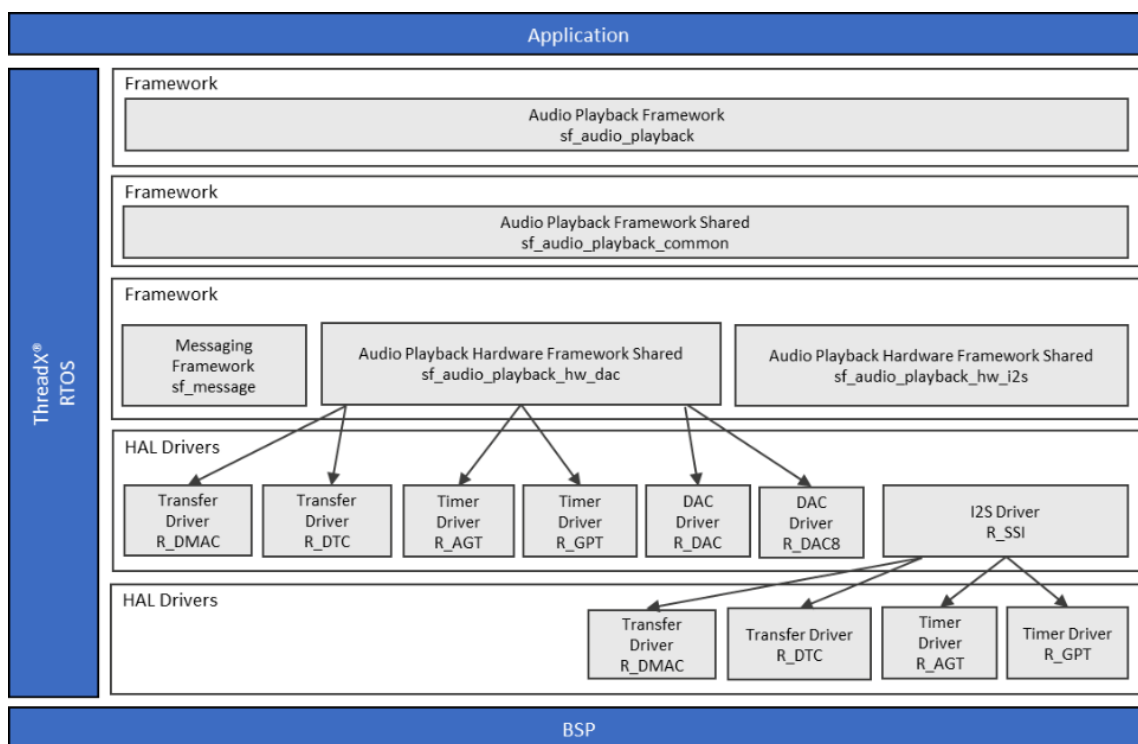


図 12.3 オーディオ再生フレームワーク

12.5 オーディオ再生 DAC フレームワーク

オーディオ再生 DAC フレームワークは、オーディオ再生アプリケーション向けのハイレベルの API を提供し、`sf_audio_playback_hw_dac` に実装されています。このモジュールは、8 および 16 ビットパルス符号変調 (PCM) サンプルを再生するために必要な同期を処理します。オーディオ再生 DAC フレームワークは、MCU 上の DAC、タイマ (AGT または GPT)、データ転送 (DMA または DTC) 周辺機能を使用します。ユーザー定義のコールバックを作成し、追加データが必要な場合に対応できます。

このフレームワークの主な機能は以下のとおりです。

- データを扱いやすいチャンクに分割することにより長いバッファを再生
- ThreadX タイムアウトが発生するまで再生をリピート (正弦波音やループするバックグラウンドミュージックのように音声が続けられる場合)
- 最後のバッファ再生が開始した後、コールバックを使用して次のデータを要求
- ソフトウェアのボリュームを制御
- 一時停止と再開機能
- 符号付き 16 ビット PCM データを符号なし 12 ビット DAC の範囲にシフトするためのスケーリング
- 複数のストリームについての基本的なミキシング

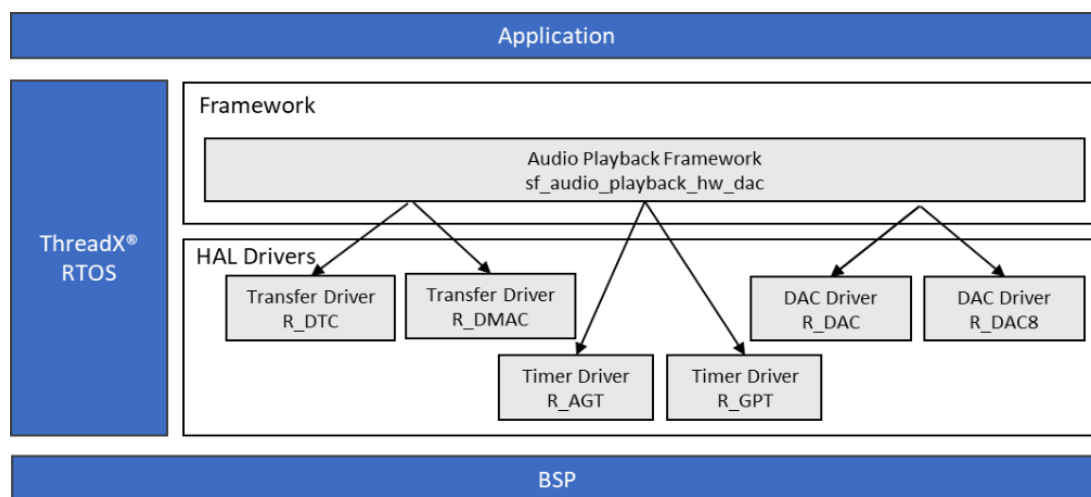


図 12.4 オーディオ再生 DAC フレームワーク

12.6 オーディオ再生 I²S フレームワーク

オーディオ再生 I²S フレームワークは、オーディオ再生アプリケーション向けのハイレベルの API を提供し、sf_audio_playback_hw_i2s に実装されています。このモジュールは、8 および 16 ビットパルス符号変調 (PCM) サンプルを再生するために必要な同期を処理します。オーディオ再生フレームワークは、MCU 上の I²S、タイマ (AGT または GPT)、データ転送 (DMA または DTC) 周辺機能を使用します。ユーザー定義のコールバックを作成し、追加データが必要な場合に対応できます。

このフレームワークの主な機能は以下のとおりです。

- データを扱いやすいチャンクに分割することにより長いバッファを再生
- ThreadX タイムアウトが発生するまで再生をリピート (正弦波音やループするバックグラウンドミュージックのように音声が繰り返される場合)
- 最後のバッファ再生が開始した後、コールバックを使用して次のデータを要求
- ソフトウェアのボリュームを制御
- 一時停止と再開機能
- 複数のストリームについての基本的なミキシング

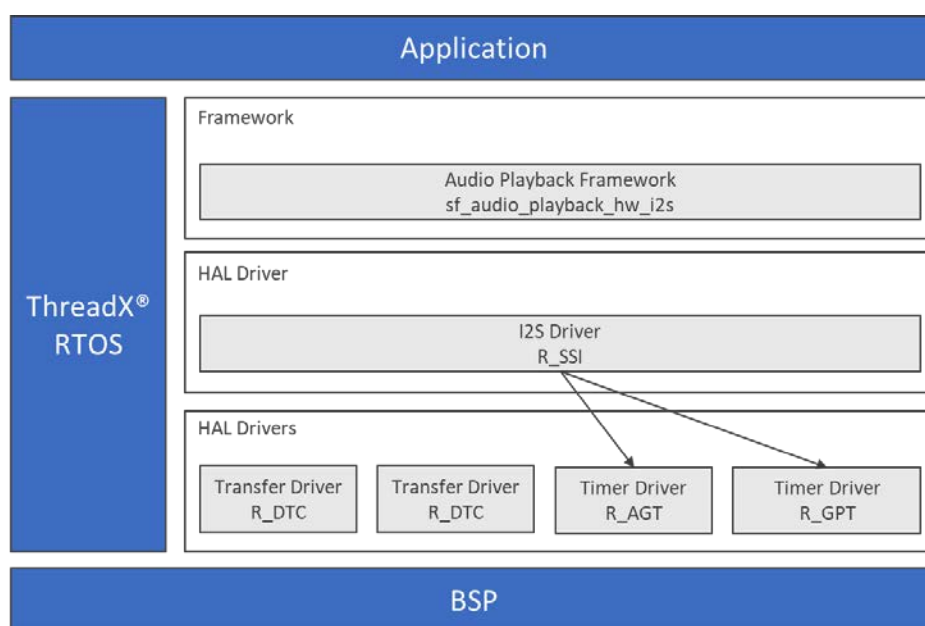


図 12.5 オーディオ再生 I²S フレームワーク

12.7 オーディオ録音 ADC フレームワーク

オーディオ録音 ADC フレームワークは、オーディオ記録アプリケーション向けのハイレベルの API を提供し、sf_adc_periodic に実装されています。オーディオ録音 ADC フレームワークは、MCU 上の sf_adc_periodic とその下位レイヤーの ADC、GPT、DTC 周辺機能を使用します。ユーザー定義のコールバックを作成し、サンプルカウントが完了したことを通知できます。

このフレームワークの主な機能は以下のとおりです。

- 8ビットまたは12ビット PCM でデータを録音
- ADC 周期フレームワークを使用して構成と統合を単純化
- ThreadX オブジェクト（ミューテックスなど）を使用して不適切なアクセスからハードウェアを保護
- ハイレベル関数の API によってコーディングを簡略化

open、start
stop、infoGet
close

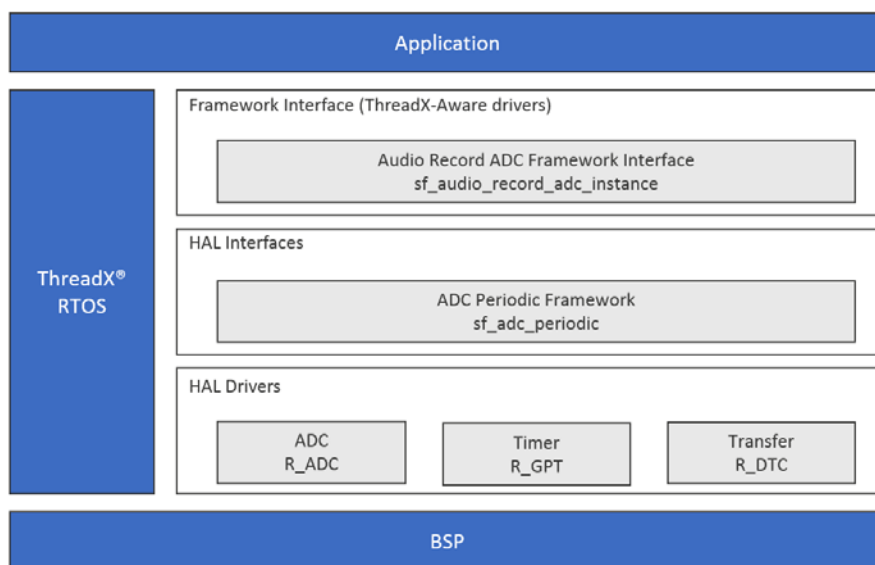


図 12.6 オーディオ録音 ADC フレームワーク

12.8 オーディオ録音 I²S フレームワーク

オーディオ録音 I²S フレームワークは、オーディオ録音アプリケーション向けのハイレベルの API を提供し、I²S インタフェースを使用します。オーディオ録音 I²S フレームワークモジュールは、MCU 上の SSI、GPT、DTC 周辺機能を使用します。ユーザー定義のコールバックを作成し、新しいサンプルがユーザーバッファに格納されていることを通知できます。

このフレームワークの主な機能は以下のとおりです。

- スレッドセーフ
- 8ビットまたは16ビット PCM でデータを録音
- 新しいサンプルが使用可能になると呼び出される周期的コールバック関数
- 構成可能なコールバックあたりのサンプル数 (サンプルカウント)

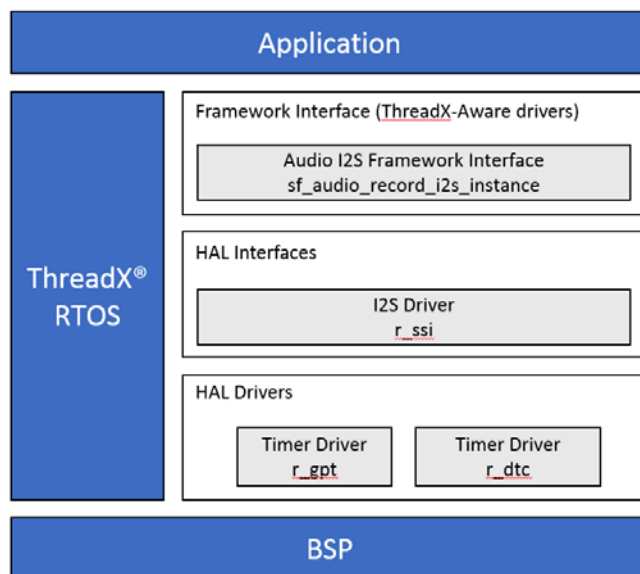


図 12.7 オーディオ録音 I²S フレームワーク

12.9 Bluetooth Low Energy (BLE) フレームワーク

Bluetooth Smart とも呼ばれる Bluetooth Low Energy (BLE) は従来の Bluetooth の軽量サブセットであり、Bluetooth 4.0 の仕様で追加されました。BLE は、従来の Bluetooth とは対照的に、消費電力を非常に低く抑えるように設計されています。この低消費電力により、電力容量が限られている IoT 分野向けのデバイスは、近接するデバイス間で小容量のデータを送信できるようになります。

アプリケーションエンジニアは、BLE スタックによって提供される機能を利用するために API を使用します。さまざまなベンダーから提供される BLE スタック API は標準化されていません。したがって、アプリケーションエンジニアは、異なる BLE スタックにコードを移植する際に、コードを更新する必要があります。

BLE フレームワークは、汎用インタフェースによってこの問題に対処します。汎用インタフェースは、さまざまなベンダーから提供される基礎となる BLE スタックに接続し、アプリケーションとベンダー固有 BLE スタックコードが対になることを防ぎます。汎用 API の使用により、アプリケーション開発が単純かつ移植可能になります。

BLE フレームワークは、BLE アプリケーション用のハイレベルの API を提供するフレームワークで、sf_ble_rl78g1d として実装されます。BLE フレームワークは SSP 通信フレームワークを使用し、基礎となる BLE モジュールと通信するための UART ドライバーを有効にします。SSP 通信フレームワークによって汎用 BLE プロファイルフレームワーク (g_sf_ble_onboard_profile) も統合され、BLE プロファイルに対して統一されたインタフェースが提供されます。RL78G1D BLE ハードウェアモジュールの場合、汎用 BLE プロファイルは BLE モジュールファームウェアによって実装されます。

このフレームワークの主な機能は以下のとおりです。

- ThreadX RTOS 対応およびスレッドセーフ
- Bluetooth v4.2 準拠フレームワーク
- Generic Access Profile (GAP) 機能
 - ユーザー定義アドバタイジングデータ
 - セキュリティモード 1 および 2
 - ペリフェラルロールおよびセントラルロール
 - 最大 6 デバイスに対応するホワイトリスト
 - ボンディングサポート
- Generic Attribute Profile (GATT) 機能
 - GATT クライアントおよびサーバー
- Generic Attribute Profile (GATT) API
- Generic Access Profile (GAP) API
- 汎用オンボードプロファイル API

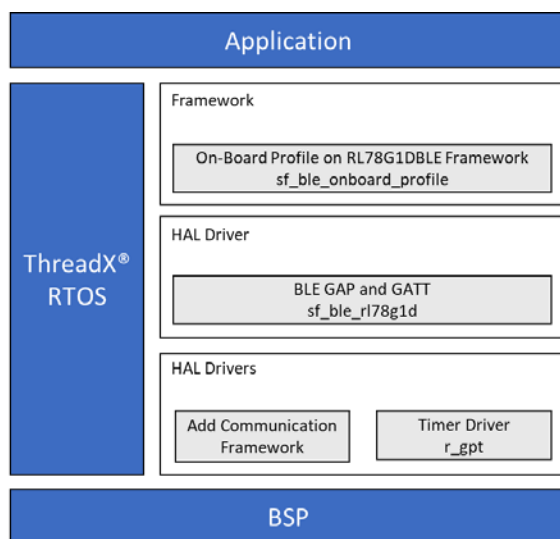


図 12.8 BLE フレームワーク内のブロック

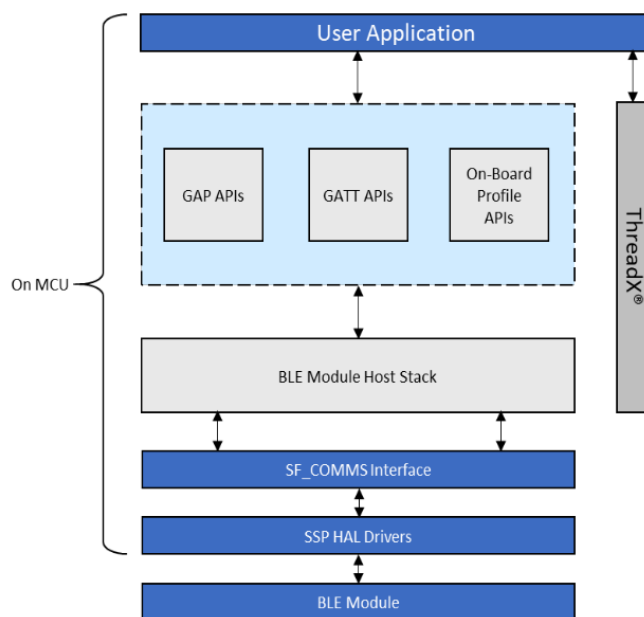


図 12.9 BLE フレームワークのレイヤーアーキテクチャ

BLE フレームワークは以下のブロックで構成されています。

1. GAP API および GATT API

BLE フレームワークは、BLE 通信モジュールの構成とプロビジョニングを行うアプリケーションのために汎用インタフェースを提供します。BLE モジュールには、Bluetooth Smart 規格によって指定されたさまざまな構成パラメータがあります。個別のデバイスドライバまたは BLE 通信モジュール（あるいは両方）ですべてのパラメータの構成がサポートされるとは限りません。プロビジョニング API で最小限提供されるのは、BLE インタフェースの動作モード、セキュリティモード、セキュリティキー、ボンディングモードを設定するメカニズムです。GAP および GATT レイヤーの API も提供されます。

2. オンボードプロファイル API

オンボードプロファイル API では、BLE 通信モジュールファームウェアによって実装される BLE プロファイルに統一インタフェースが提供されます。

3. BLE モジュールのホストスタック

通常、BLE 通信モジュールのホストスタックはモジュールベンダーによって提供されます。

4. BLE アーキテクチャオプション

BLE 通信ハードウェアモジュールは、一般に、ホスト MCU と BLE 通信モジュールの間のハードウェアおよびソフトウェアパーティションに応じて 3 種類の構成で提供されます。

- A. BLE 無線専用モード：リンクレイヤーと他のすべての BLE プロトコルスタックレイヤーに加え、プロファイルとアプリケーションをホスト MCU で実行します。物理レイヤーは BLE チップセットで実行します。
- B. BLE コントローラ実装：リンクレイヤーは BLE チップセットで実行し、L2CAP と上位の BLE プロトコルレイヤー、さらにプロファイルとアプリケーションはホスト MCU で実行します。
- C. ネットワークコントローラ実装：リンクレイヤー、L2CAP、GATT、GAP レイヤー、汎用プロファイルを BLE チップセットで実行します。オプションのプロファイルとアプリケーションはホストプロセッサで実行します。

注：(A) BLE 無線専用モードと (B) BLE コントローラ実装については、例示のみを目的に記載しています。現在のリリースでサポートされているのは、RL78G1D ネットワークコントローラ実装だけです。

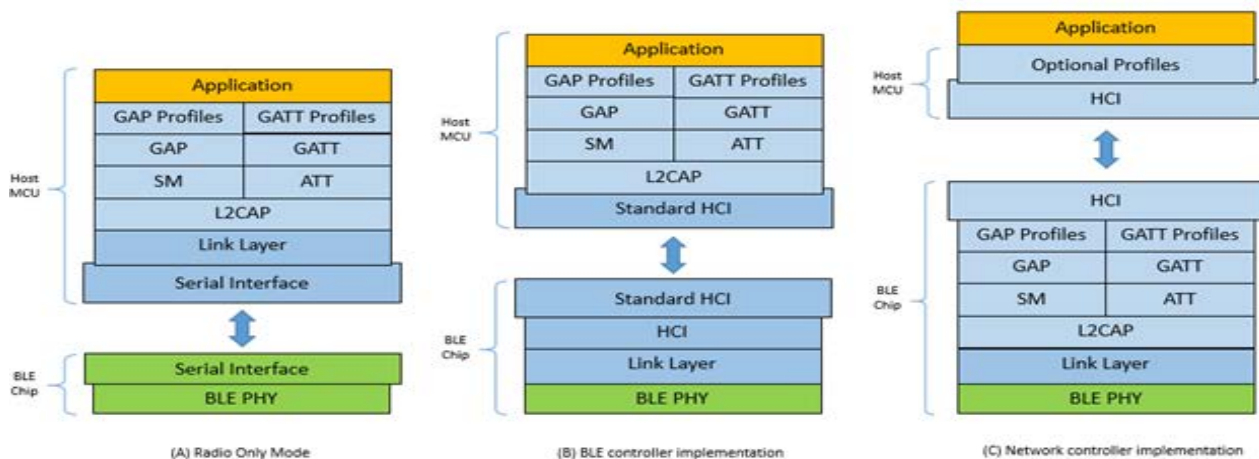


図 12.10 サポートされる BLE 通信ハードウェアモジュールアーキテクチャモード

12.10 ブロックメディア QSPI フレームワーク

ブロックメディア QSPI フレームワークは、QSPI チャンネル用の API を実装し、r_qspi ドライバーによる QSPI フラッシュメモリ周辺機能の読み取り、書き込み、制御をサポートします。このドライバーは、ブロックメディアインタフェースを介してファイルシステムとやり取りするために必要なすべての機能を備えています。

このフレームワークの主な機能は以下のとおりです。

- QSPI フラッシュメモリデバイス用の QSPI チャンネルインタフェースをサポート
- QSPI フラッシュメモリ上のファイルシステムをサポート

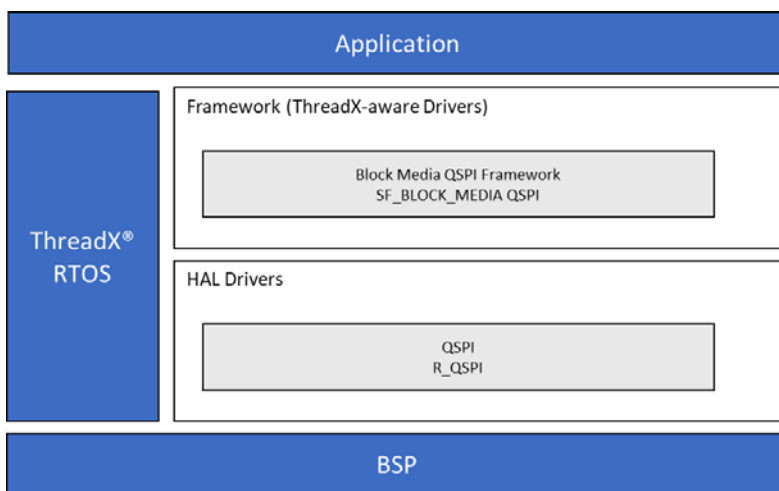


図 12.11 ブロックメディア QSPI フレームワーク

12.11 ブロックメディア RAM フレームワーク

ブロックメディア RAM フレームワークモジュールは、RAM メモリの読み取り、書き込み、および読み取り/書き込み領域の制御用に、RAM 上にファイルシステムを実装します。このフレームワークは、ブロックメディアインタフェースを介してファイルシステムとやり取りするために必要なすべての機能を備えています。

このフレームワークの主な機能は以下のとおりです。

- 線形メモリマップデバイスでの FileX の実行を有効化します。
- RAM に対するデータの高速一時ストレージ

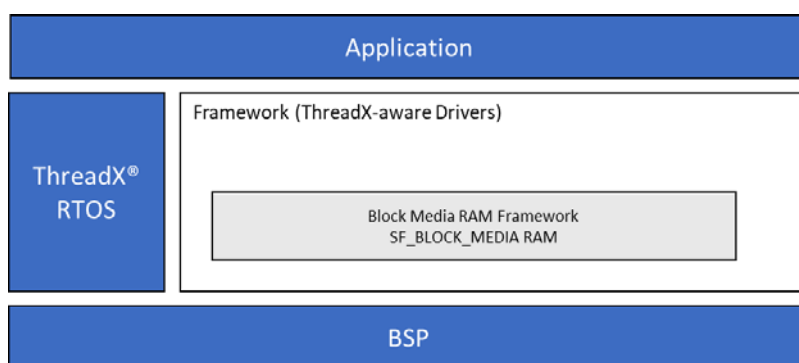


図 12.12 ブロックメディア RAM フレームワーク

12.12 ブロックメディア SDMMC フレームワーク

ブロックメディア SDMMC フレームワークの情報は、FileX ポートブロックメディアに関するセクションに記載されています。詳細については、「12.28 Synergy FileX ポートブロックメディアインタフェースフレームワーク」を参照してください。

12.13 静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワーク

静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワークは、静電容量式タッチボタンアプリケーション用のハイレベルの ThreadX 対応 API を提供し、ThreadX RTOS を使用して `sf_touch_ctsu` に実装されています。静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワークは、MCU 上の CTSUS 周辺機能を使用します。各ボタンに対して列挙された順に応答するユーザー定義のコールバックを作成できます。

静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワークは、システムに存在するすべてのボタンのために、静電容量式タッチセンシングユニットフレームワークから受信するデータを解釈します。

このフレームワークの主な機能は以下のとおりです。

- 構成データを生成する Capacitive Touch Workbench for Renesas Synergy (CTW for Synergy) ツールと連携します。
- S124、S128、S3A3、S3A7、S5D9、S7G2 をサポートします
- コールバック関数をイベントに提供します。
 - デバウンスを実行します
 - Press、Release、LongTouch などの複数のタイプのイベントをサポートします
 - 各ボタンのコールバックをボタン構成テーブルに列挙された順に呼び出します。

静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワークには静電容量式タッチフレームワークが必要です。ほとんどの場合、必要な構成情報はモジュールに自動的に追加されます。



図 12.13 静電容量式タッチセンシングユニットのチューニングツール

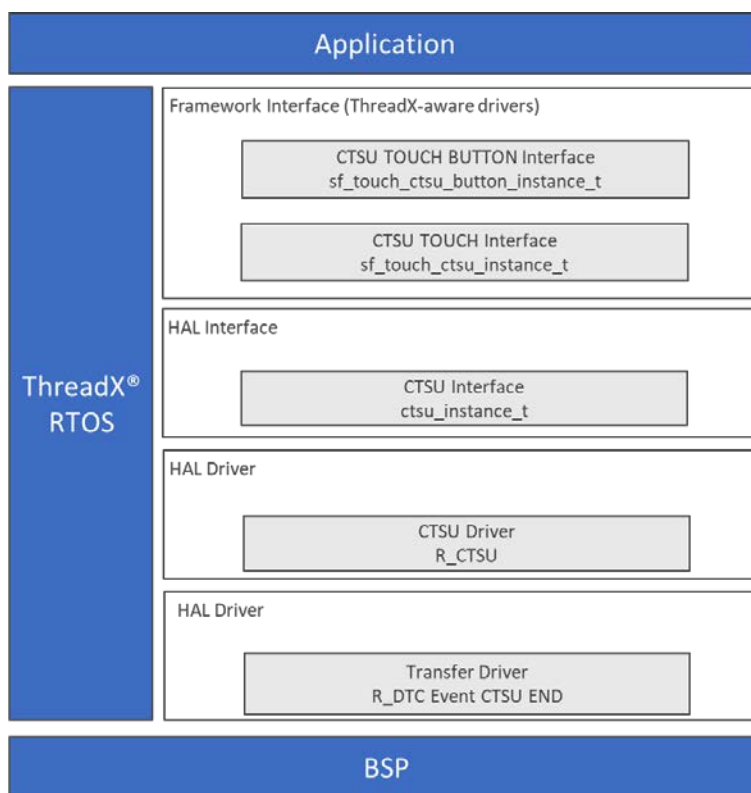


図 12.14 静電容量式タッチセンシングユニット (CTSUS) ボタンフレームワーク

12.14 静電容量式タッチセンシングユニット (CTSUS) フレームワーク

静電容量式タッチセンシングユニット (CTSUS) フレームワークは、ThreadX RTOS を使用して静電容量式タッチアプリケーション用のハイレベルの API を提供し、sf_touch_ctsu に実装されています。静電容量式タッチセンシングユニット (CTSUS) フレームワークは、基本的には r_ctsu HAL ドライバーのフレームワークバージョンとして動作し、追加機能としてスキャンを自律的に実行します。このフレームワークは、ボタンやスライダーのような他のフレームワークと併用できます。また、CTSUS 周辺機能の各チャンネルのステータスに関してバイナリデータのみを要求するアプリケーションでも使用できます。

静電容量式タッチセンシングユニット (CTSUS) フレームワークは、静電容量タッチパネルのハードウェアスキャンを実行してパネルを周期的に更新するプライベートスレッドを作成します。

このフレームワークの主な機能は以下のとおりです。

- 複数のアプリケーションレイヤーがコールバックを登録可能。このフレームワークにより、スライダーやボタンなどの複数のウィジェットがこのレイヤーを使用可能
- 周期的な CTSUS スキャンを自律的に実行し、ユーザーが指定したレートでプロセスを更新

このフレームワークは、CTW for Synergy ツールが生成する設定データと合わせて使用されるように設計されています。

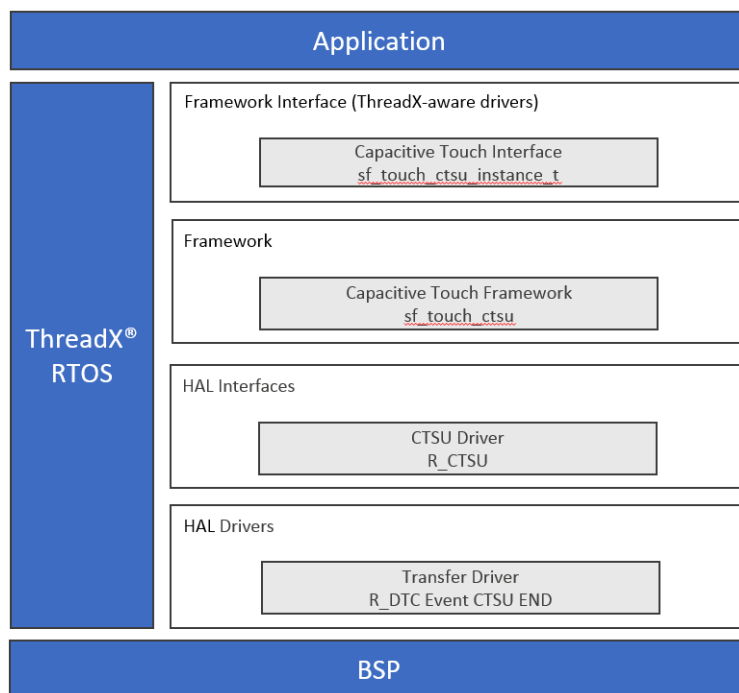


図 12.15 静電容量式タッチセンシングユニット (CTSUS) フレームワーク

12.15 静電容量式タッチセンシングユニット (CTSUS) スライダーフレームワーク

静電容量式タッチセンシングユニット (CTSUS) スライダーフレームワークは、静電容量式タッチスライダーおよびホイールアプリケーション用のハイレベルの ThreadX 対応 API を提供し、sf_touch_ctsu_slider に実装されています。静電容量式タッチセンシングユニット (CTSUS) スライダーフレームワークは、MCU 上の CTSUS を使用します。このフレームワークは、CTW for Synergy ツールによって生成された構成データとともに使用するように設計されています。使用するスライダー、ホイール、およびチャネルは CTW ツールで構成します。ユーザー定義のコールバックを作成しておく、そのコールバックがスキャン後にハードウェアから使用可能になったときにデータを処理できます。

静電容量式タッチセンシングユニットスライダーフレームワークは、システムによって初期化されるすべてのスライダー構成について、CTSUS フレームワークからの受信データを解釈します。

このフレームワークの主な機能は以下のとおりです。

- スライダーとホイールをサポート
- スライダーとホイールの複数インスタンスをサポート
- タッチ処理の簡略化に使用するコールバック
 - CTW for Synergy から生成された構成データを使用
 - 状態が変化したときにコールバックを生成
 - コールバックは各スライダーに関連付けられ、イベントと位置を含む
 - コールバックは構成テーブルに列挙された順にコール
- マルチタッチ検出をサポート (ビルド時にオプションで無効化可能)

なお、このフレームワークは、CTW for Synergy ツールが生成する設定データと合わせて使用されるように設計されています。

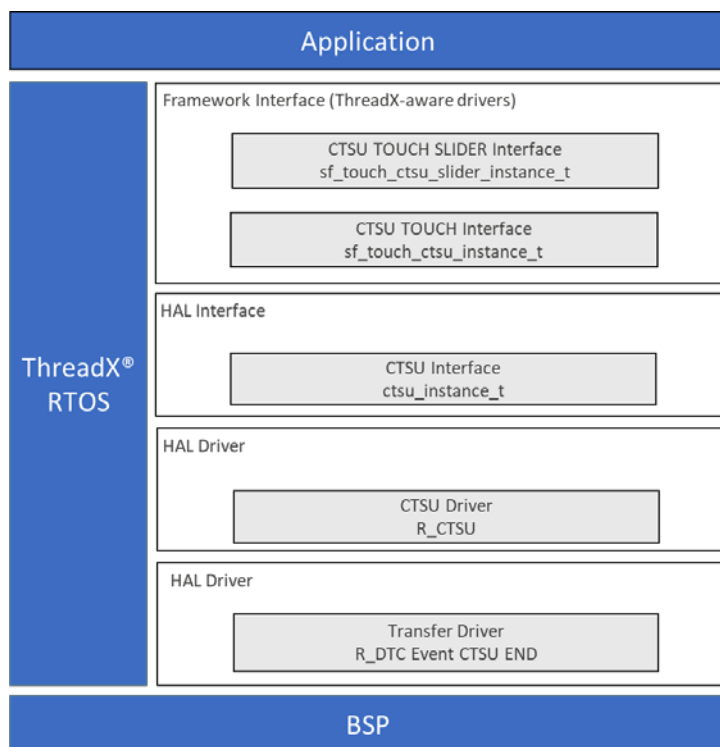


図 12.16 静電容量式タッチセンシングユニット (CTSUS) スライダーフレームワーク

12.16 セルラーフレームワーク

セルラーフレームワークによって、SSP でのセルラーモデム統合のためにハイレベルアプリケーションレイヤーインタフェースと、さまざまなベンダのセルラーモデムをアプリケーションで使用可能にする共通インタフェースが提供されます。

このフレームワークでは、データ通信の目的でセルラーネットワークと通信するために、セルラーモデムのプロビジョニングと構成を行います。コンソールフレームワークにより、AT コマンドを使用してシリアルインタフェース経由でセルラーモデムと通信します。このフレームワークは、NetX によって提供される PPP WAN プロトコルを利用して、データ通信のシリアルインタフェース上でシリアルデータパイプを作成します。TCP/IP を使用するデータ通信は、NetX アプリケーションプロトコル、ソケット、あるいは IoT プロトコル (MQTT など) を使用して、このワイドエリアネットワーク (WAN) リンク上で確立できます。

また、セルラーフレームワークではフレームワークレベルのソケット API も提供されます。これは、特定のセルラーハードウェアモジュール内のオンチップ (セルラーハードウェアモジュール内) の TCP/IP スタックや、さらにはソケット API を使用するネットワークの TCP/IP リンクと通信するためのものです。

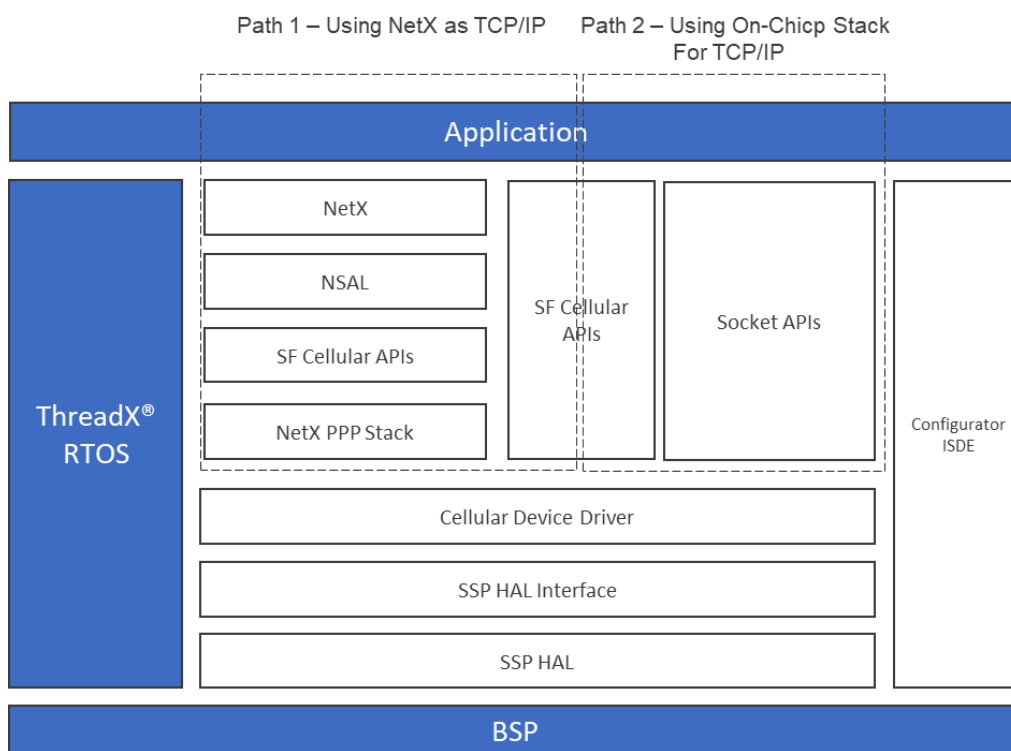


図 12.17 セルラーフレームワークの構成およびインタフェースレイヤー

セルラーフレームワークは以下の論理ブロックで構成されています。

- セルラーフレームワークアプリケーションインタフェース
- NetX TCP/IP スタック用のネットワークスタック抽象化レイヤー (NSAL)
- セルラーデバイスドライバー
- オンチップネットワークスタックをサポートするセルラーハードウェアモジュールのインタフェースになる BSD ソケット互換 API
- SSP HAL インタフェース

1. SF セルラーフレームワーク API

アプリケーションでは、SF セルラーフレームワーク API インタフェースを使用して、アプリケーションを変更することなくさまざまなベンダのセルラーモデムと通信します。モジュール固有の実装により、AT コマンドセットが提供され、必要な機能を実装することで汎用ドライバーが上書きされます。たとえば、汎用ドライバーによって、オープン、クローズ、プロビジョニングなどの API が実装されます。これらの API に必要な AT コマンドは、モジュール固有のドライバーで提供されます。モジュール固有のドライバーに異なる方法でオープン API を実装する必要がある場合は、独自の実装によってオープンな実装を上書きして、実装することができます。

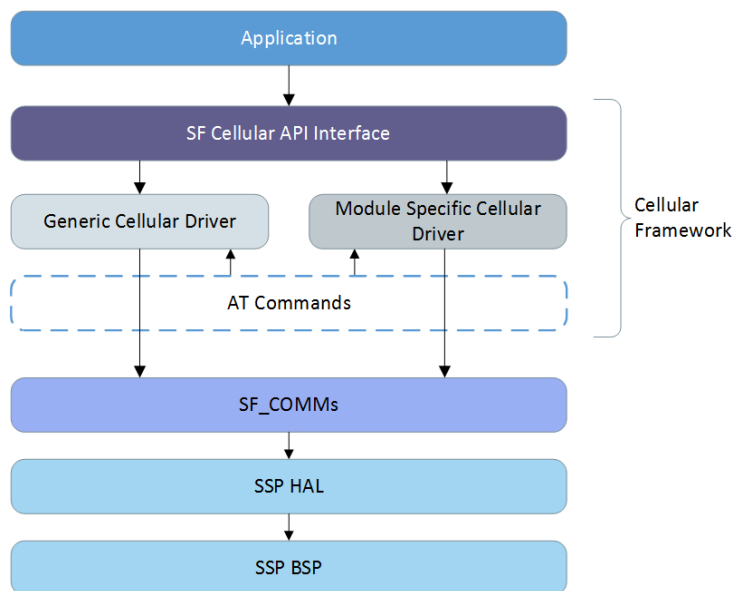


図 12.18 セルラーフレームワーク API

2. ネットワークスタック抽象化レイヤー

セルラーフレームワークは、ネットワークスタック抽象化レイヤー (NSAL) を提供します。NSAL によって、IP 通信で使用する PPP チャンネルが作成されます。また、NSAL では PPP が使用する認証方法 (PAP/CHAP など) も処理されます。これらの認証メカニズムはオプションです。NSAL は、フレームワーク API を使用して、セルラーモジュールとの間でデータの送受信を行います。NSAL を使用すると、ネットワークスタックに関して変更せずにセルラーデバイスドライバーを再利用できます。新しいネットワークスタックのサポートを追加するには、適切な NSAL の実装が必要です。

3. セルラーデバイスドライバー

セルラーフレームワークは AT コマンドセットを使用し、シリアルドライバーを使用してセルラーモデムとやりとりします。モデムとのやりとりに使用されるシリアルインタフェースは UART です。

4. ソケット API

ソケット API によってアプリケーションにインタフェースが提供され、セルラーモジュールに存在するオンチップネットワーキングスタックを BSD ソケット API で使用できるようになります。使用にあたっては、セルラーモジュールまたはドライバーが、オンチップネットワーキングスタックとソケットインタフェースをサポートする必要があります。アプリケーションは、これらの API を使用する場合に、セルラーモジュールに存在するオンチップネットワーキングスタックを使用し、NSAL は使用しません。アプリケーションは、MCU 上で実行中のネットワーキングスタックを使用しません。

5. SSP HAL インタフェース

次の図は、MCU とのローレベル通信のためにセルラーデバイスドライバで使用される SSP HAL コンポーネントのインタフェースを示しています。この実装はセルラーデバイスドライバ固有です。セルラーモジュールは、UART、ICU、IOPORT などのさまざまな HAL コンポーネントを利用します。

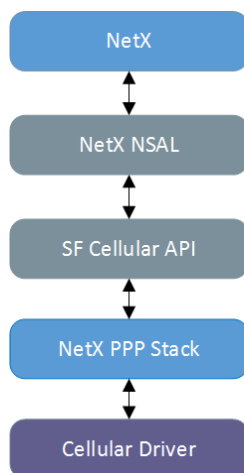


図 12.19 SSP HAL インタフェース

このフレームワークの主な機能は以下のとおりです。

- 以下を使用する接続がサポートされます。
 - CAT1、CAT3、および CAT M1 セルラーモデム
 - セルラーモジュールに存在するオンチップスタック対応の BSD ソケットインタフェース
 - NSAL インタフェースを使用する MCU (ホスト) 上の NetX スタック
- ネットワーキングスタックに対応する API の共通セットと、さまざまなセルラーハードウェアモジュール対応の汎用インタフェースがサポートされます。
- 汎用 API と抽象化を利用して、セルラーハードウェアモジュール用に開発されたアプリケーションを、別のセルラーハードウェアモジュールと連携するために簡単に移行できます。
- サポートされるセルラーモデム:
 - NimbeLink CAT3 (NL-SW-LTE-TSVG) Verizon-US
 - NimbeLink CAT3 (NL-SW-LTE-TEUG) India and Europe
 - NimbeLink CAT1 (NL-SW-LTE-GELS3-B and NL-SW-LTE-GELS3-C) Verizon-US
 - Quectel CAT M1-BG96

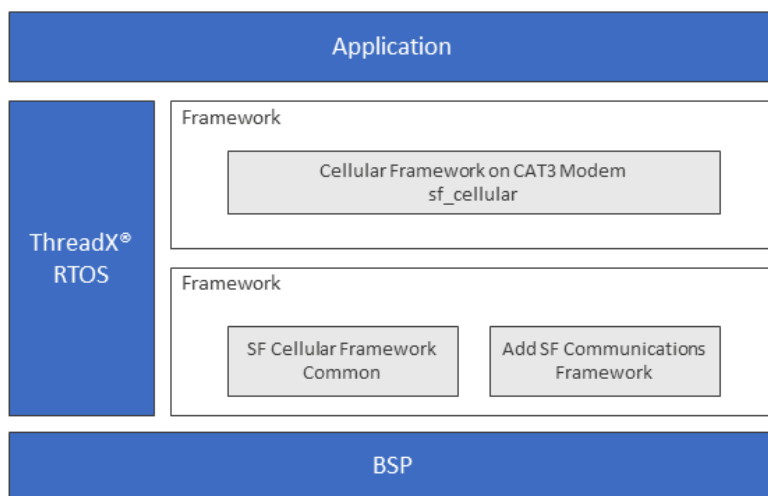


図 12.20 CAT3 モデム上のセルラーフレームワークモジュールの編成、オプション、スタックの実装

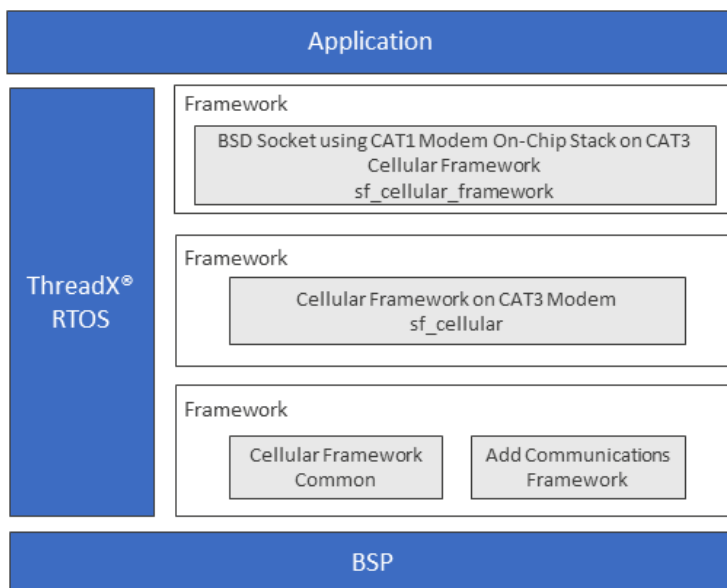


図 12.21 CAT3 セルラーフレームワークモジュール上の CAT3 オンチップスタックを使用した BSD ソケット

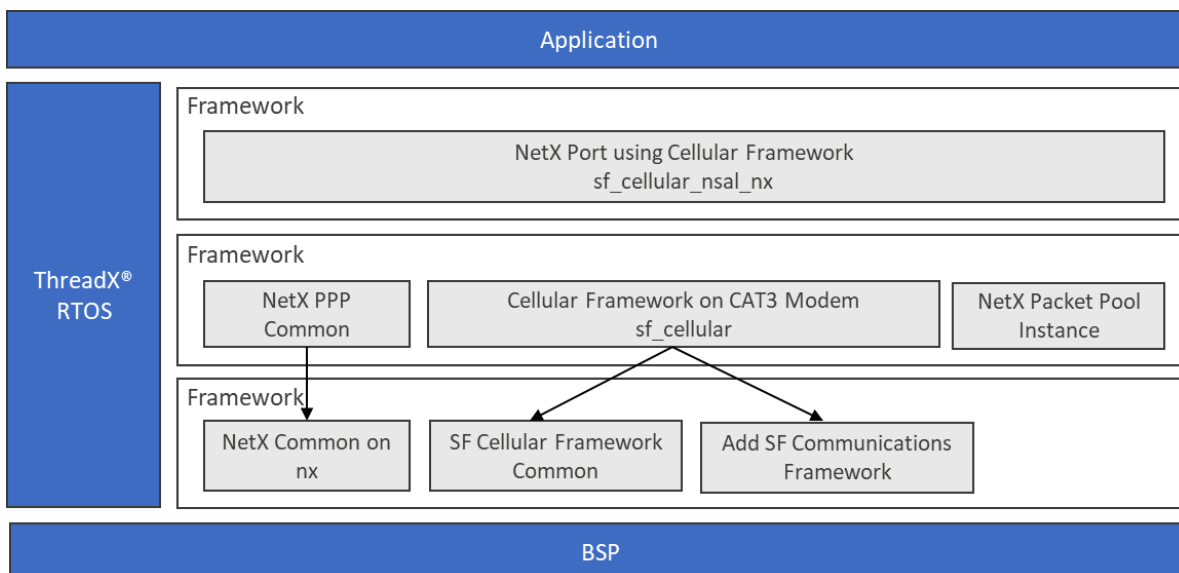


図 12.22 セルラーフレームワークモジュールを使用した NetX ポートの編成、オプション、スタックの実装

12.17 NetX Telnet 通信フレームワーク

NetX Telnet 通信フレームワークは、telnet 通信アプリケーション用のハイレベルの API を提供し、sf_comms_telnet に実装されています。このモジュールは、MCU 上のイーサネット周辺機能を使用します。

このフレームワークの主な機能は以下のとおりです。

- イーサネット上でハイレベルの接続性がサポートされますが、API を変更せずに UART や USB 接続に簡単に変更可能
- 専用アクセスのためのチャンネルロックをサポート
- Thread 対応実装はミューテックスとイベントフラグを内部で使用

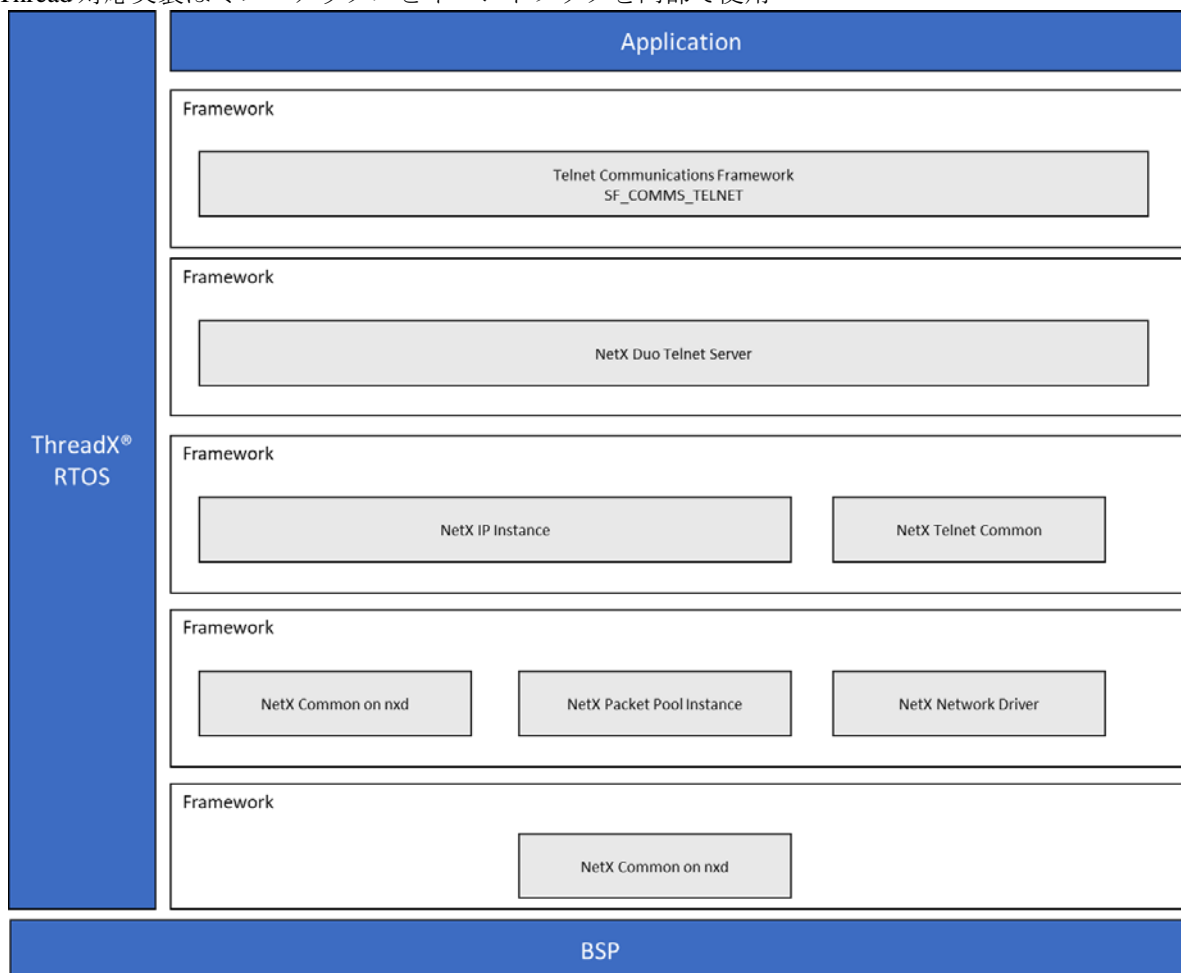


図 12.23 NetX Telnet 通信フレームワーク

12.18 USBX 通信フレームワーク

USBX 上の通信フレームワークは、通信フレームワークアプリケーション用のハイレベルの API を提供し、USBX デバイスクラス CDC-ACM (通信デバイスクラス-抽象制御モデル) である sf_el_ux_comms に実装されています。この通信フレームワークは、MCU デバイス上の USB 周辺機能を使用します。このモジュールは SSP v1.2.0 以降のリリースでは非推奨ですが、下位互換性を維持するために SSP リリースに含まれています。

このフレームワークの主な機能は以下のとおりです。

- Express Logic USBX を SSP サポート USBX API に実装
- USBHS および USBFS 周辺機能用のポートデバイスコントローラドライバー (DCD) をサポート
- USBHS および USBFS 周辺機能用のポートホストコントローラドライバー (HCD) をサポート
- 転送モジュールの動作をサポート (オプション)

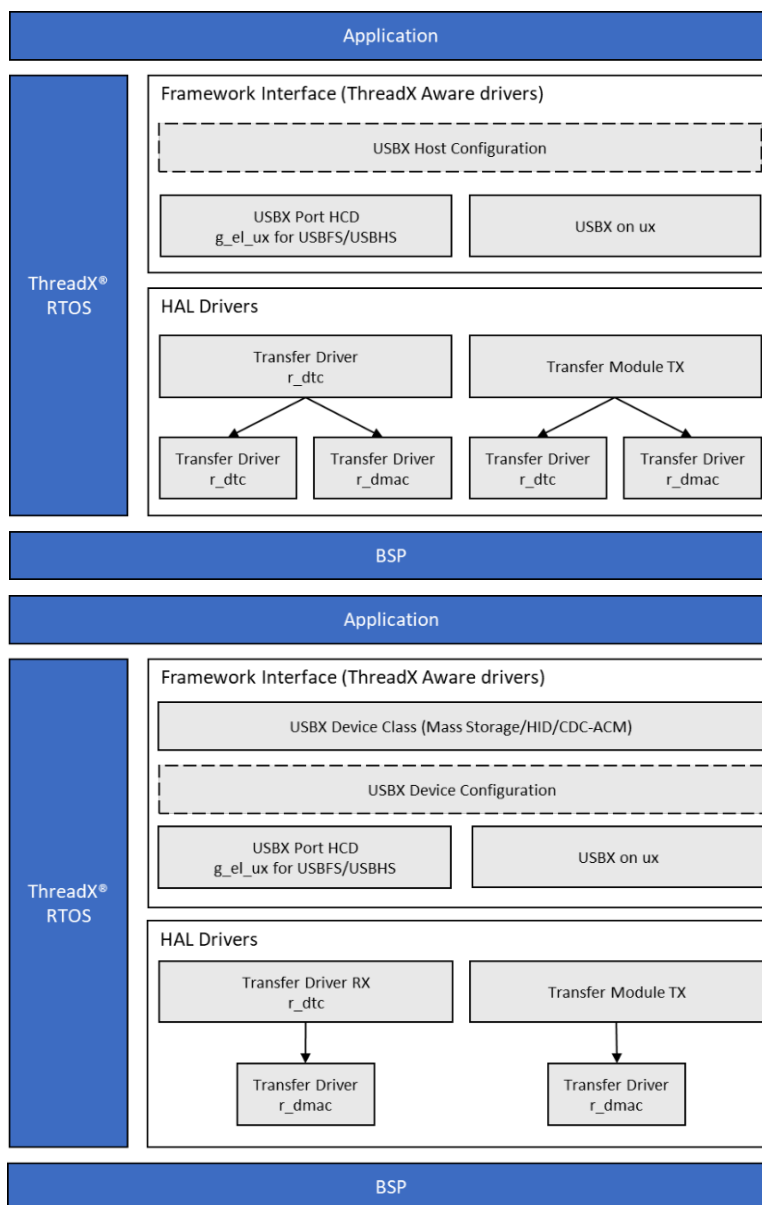


図 12.24 USBX 通信フレームワーク

12.19 コンソールフレームワーク

コンソールフレームワークは、ThreadX RTOS を使用するメニュー駆動型コンソールコマンドラインインタフェース (CLI) 用のハイレベルの API を提供し、sf_console に実装されています。このコンソールフレームワークモジュールでは通信インタフェースを使用して、UART、USB、イーサネット Telnet 接続のいずれかに対応するハードウェアオプションに接続します。コンソールフレームワークモジュールには、ユーザー定義のコマンドのメニューとさまざまな API があり、プロンプトの表示、メニューコマンドのコールバックの識別および発行、入力文字列の読み取り、書き込み、解析を行うことができます。

このフレームワークの主な機能は以下のとおりです。

- メニューに基づくコマンドラインインタフェースの作成
- サブメニューと、単一呼び出しによる複数メニュー内の移動
- 親メニューへの移動またはメニューのルートに戻る
- メニューごとのヘルプメニュー
- NULL 終端文字列の書き込みと改行文字までの読み取り
- 引数をコマンドラインに解析する API
- 大文字と小文字を区別しない入力

コンソールフレームワークモジュールの編成 (SSP コンフィギュレータの[Thread Stack]ウィンドウに表示される) を次の図に示します。各実装の選択肢 (イーサネット、UART、USB) にはそれぞれローレベルモジュールがあり、エンジニアが選択した実装に基づいて自動的に追加されます。ほとんどのケースで、必要なすべての構成情報がモジュールに自動的に追加されるため、エンジニアが選択する必要があるのはごくわずかの重要な構成設定のみです。

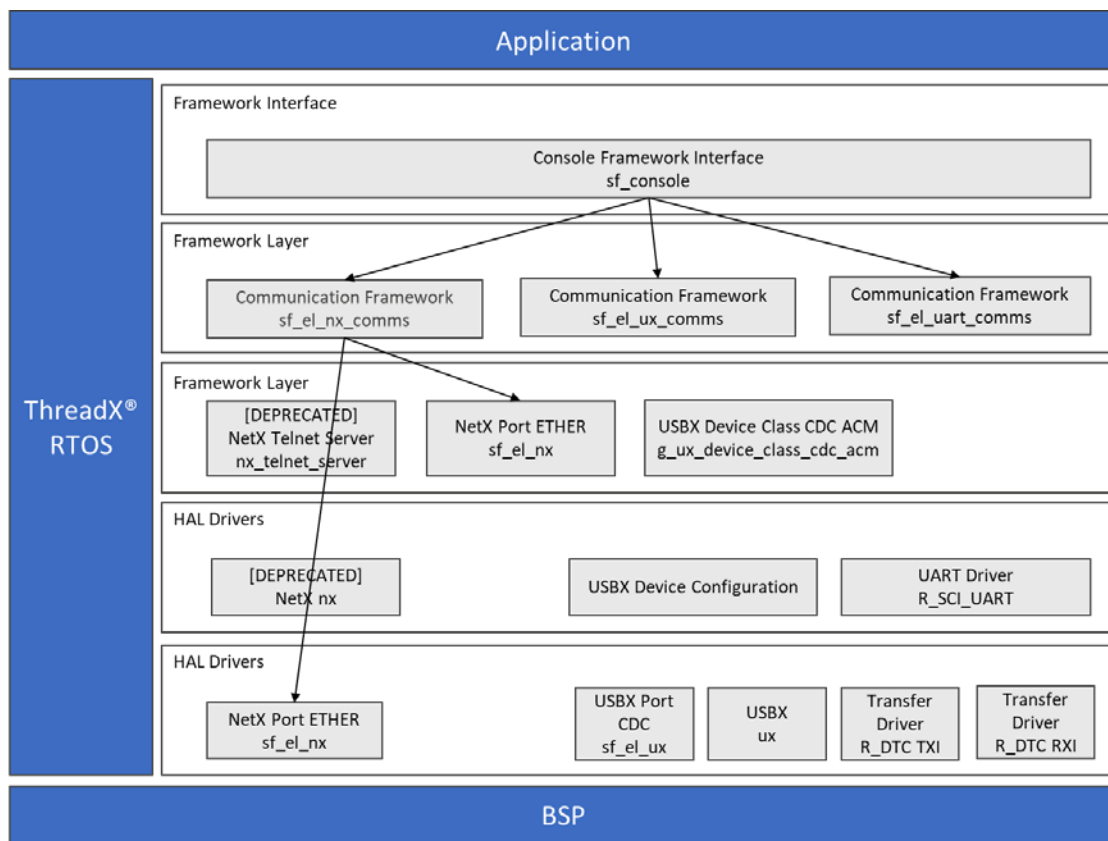


図 12.25 コンソールフレームワーク

12.20 暗号化フレームワーク

暗号化フレームワークは、さまざまな暗号化サービスを提供する複数の暗号化モジュールで構成されています。暗号化フレームワークモジュールには、真性乱数生成用の `SF_CRYPTO_TRNG`、メッセージダイジェスト生成用の `sf_crypto_hash`、RSA および AES キー生成サービス用の `sf_crypto_key` が含まれます。ターゲット MCU でサポートされる暗号化機能については、「14.38 Secure Cryptographic Engine (SCE)」および表 14.1 を参照してください。

このフレームワークの主な機能は以下のとおりです。

- **SF CRYPTO フレームワーク**
 - SF CRYPTO フレームワークは、基礎となるセキュア暗号化エンジンを開きます。
 - 他の暗号化フレームワークモジュールである `SF_CRYPTO_TRNG`、`SF_CRYPTO_HASH`、`SF_CRYPTO_KEY`、`SF_CRYPTO_CIPHER`、`SF_CRYPTO_SIGNATURE`、および `SF_CRYPTO_KEY_INSTALLATION` 用の共有リソースにアクセスするサービスを提供します。
- **SF CRYPTO TRNG フレームワーク**
 - SF CRYPTO TRNG フレームワークモジュールは、基礎となるセキュア暗号化エンジンへの TRNG HAL インタフェースを使用します。
 - SF CRYPTO フレームワークモジュールを通じて共有リソースにアクセスします。
- **SF CRYPTO HASH フレームワーク**
 - SF CRYPTO HASH フレームワークは、基礎となるセキュア暗号化エンジンを利用して HASH サービスを提供します。
 - このモジュールは、以下のような HAL ドライバーに対する機能強化を提供します。
 - HASH 値の初期化
 - ハッシュ操作の終了処理の前に `hashUpdate` API を通じてデータのチャンクを処理します。
 - 最終 HASH 処理の最終ブロックのフォーマットを設定します。
- **SF CRYPTO KEY フレームワークモジュールの特長**
 - RSA 2048 ビット、1024 ビットプレーンテキストまたは未加工キー。
 - RSA 2048 ビット、1024 ビット標準フォーマットのラップ秘密キー（公開キーはプレーンテキスト）。
 - ECB、CBC、CTR および GCM チェーンモード用の AES 128 ビット、192 ビットおよび 256 ビットラップキー。
 - XTS チェーンモード用の AES 128 ビットおよび 256 ビットラップキー。
 - ECC 192 ビット、256 ビットラップキー。
- **SF CRYPTO SIGNATURE フレームワークモジュールの特長**
 - このモジュールは、現在 RSA アルゴリズム用の署名生成と署名検証をサポートしています。標準フォーマットのプレーンテキスト、標準フォーマットのラップ秘密キー、および CRT プレーンテキストキーに対して 1024 ビットおよび 2048 ビットのキー長をサポートしています。
 - 署名方式として、RSASSA-PKCS1 v1.5 をサポートしています。入力メッセージを、署名または検証のために未加工メッセージとして渡すか、または署名あるいは検証前に PKCS1 v1.5 によってエンコードおよびパディングすることができます。
 - SHA1、SHA224、SHA256 が、PKCS1 v1.5 エンコードおよびパディング方式のハッシュアルゴリズムとしてサポートされています。
 - このモジュールでは、ブロックサイズよりも小さなデータに対して署名生成および検証を行うことができます。
 - 一度にすべてのデータが使用できない場合は、更新 API を使用して、入力データのチャンクを蓄積し、メッセージ全体が収集された時点で、最終的にそのメッセージに対して署名や検証を行うことができます。
 - メモリの割り当てや割り当て解除に関与しない、コストの少ない API コールを使用して、署名操作と検証操作を切り替えることができます。
- **SF CRYPTO CIPHER フレームワークモジュールの特長**
 - このモジュールは、現在 AES および RSA アルゴリズム用の暗号化および復号化をサポートしています。
 - このモジュールでは、データがチャンクで使用できる場合に、最後のチャンクまたはデータ全体が収集されたときに `cipherUpdate()` API および `final()` を介してデータの暗号化および復号化が可能です。
 - 特定のキータイプおよびキーサイズに対してモジュールがオープンされると、`cipherInit()` API を呼び出すことで暗号モードおよび復号モードを切り替えることができます。

- SF CRYPTO KEY INSTALLATION フレームワークモジュールの特長
 - SF CRYPTO KEY INSTALLATION フレームワークモジュールは、基礎となるセキュア暗号化エンジンへの KEY INSTALLATION HAL インタフェースを使用します。
 - SF CRYPTO フレームワークモジュールを通じて共有リソースにアクセスします。
 - このモジュールでは、次のキーに対してキーのインストールがサポートされています。
 - RSA
 - 1024 ビットおよび 2048 ビットプレーンテキストキー。
 - AES
 - ECB、CBC、CTR、GCM チェーンモード用の 128 ビット、192 ビット、256 ビットプレーンテキストキー。
 - XTS チェーンモード用の 128 ビットおよび 256 ビットプレーンテキストキー。
 - ECC
 - 192 ビットおよび 256 ビットプレーンテキストキー。
 - ユーザーおよびインストールキーは、指定したフォーマットでキーインストール API に提供する必要があります。
 - インストール後、キーインストールサービスによってラップキーが呼び出し側に返され、以降そのデバイスでインストールキーを使用できます。

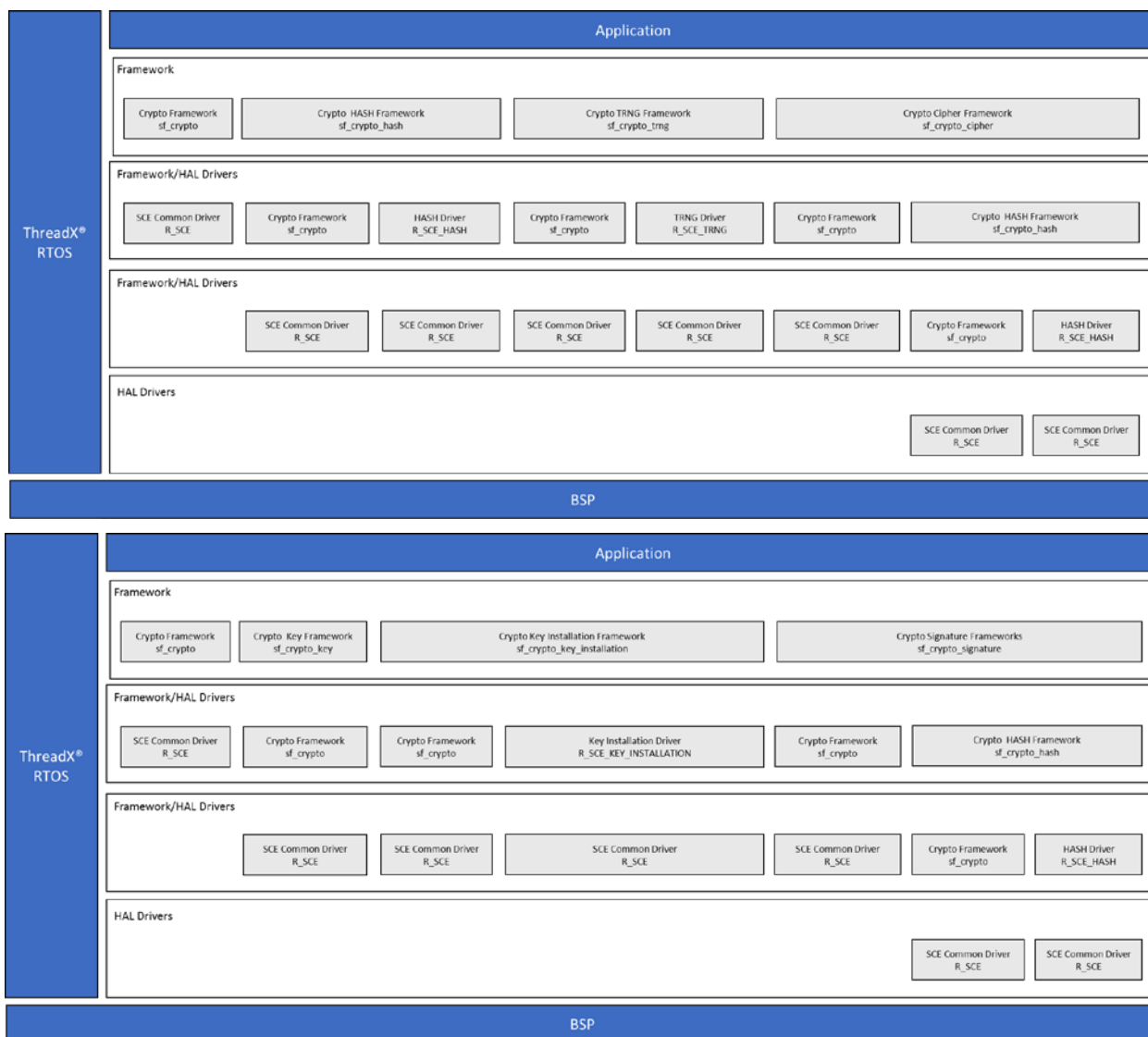


図 12.26 暗号化フレームワークモジュールの構成、オプション、スタックの実装

12.21 外部割り込みフレームワーク

外部割り込みフレームワークは、ThreadX RTOS による外部ピン割り込みを使用するアプリケーション用のハイレベルの API を提供します。このフレームワークは、MCU で利用可能な外部 IRQ ピンをサポートし、sf_external_irq に実装されます。コールバック関数 (sf_external_irq_callback) が使用可能となっており、IRQn がトリガされるたびに割り込みサービスルーチン (ISR) から呼び出されます。

このフレームワークの主な機能は以下のとおりです。

- 外部割り込み入力に応答します
- RTOS はスレッド同期に内部セマフォを使用して実装を認識します
- 内部スレッドに信号を送信できます
- イベントリンクコントローラ (ELC) を介して転送をトリガできます
- MCU で使用可能なポートピンを使用します
- ピンは MCU 間で異なる可能性があるため、詳細については該当する MCU のユーザーズマニュアルを参照してください
- 以下のような複数のハードウェア機能をサポートします
 - チャンネル選択
 - トリガ条件
 - デジタルフィルタ
 - オートスタート

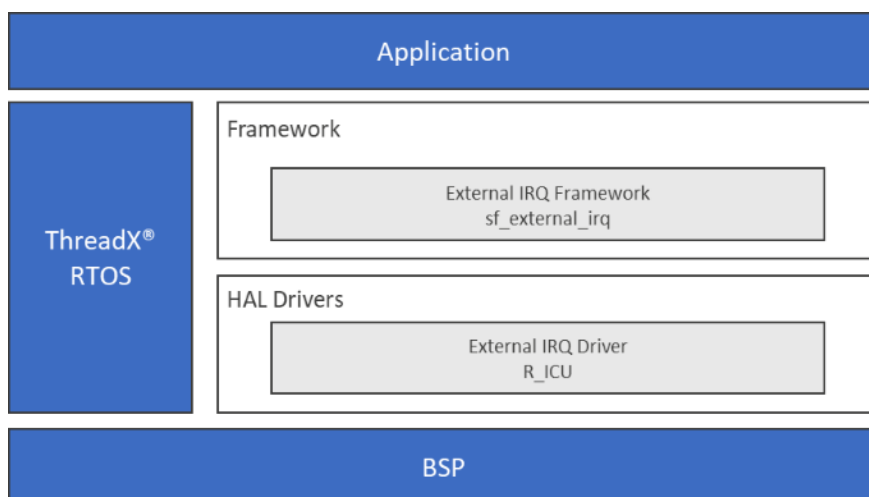


図 12.27 外部割り込みフレームワーク

12.22 I²C フレームワーク

I²C フレームワークは、I²C フレームワークアプリケーション用の ThreadX®対応のハイレベルの API で、sf_i2c に実装されています。I²C HAL モジュールでは I²C 周辺機能を構成し、フレームワークがシリアル通信を使用できるようにします。I²C フレームワークモジュールは、MCU 上の I²C および SCI 周辺機能を使用します。

このフレームワークの主な機能は以下のとおりです。

- ThreadX 対応フレームワーク
- I²C バス上の複数の I²C 周辺機能の統合と同期を処理
- SCI I²C ドライバーと RIIC ドライバーの両方にアクセスするための単一のインタフェースを提供
- I²C フレームワークモジュールはマスタモードで I²C 通信を構成
- I²C フレームワークモジュールは 100 kHz、400 kHz、1 MHz の 3 つのデータレートをサポート
- I²C フレームワークモジュールは 7 ビットアドレッシングと 10 ビットアドレッシングの両方をサポート
- I²C フレームワークモジュールは、内部的なコールバックのサポートも提供します。ユーザー定義のコールバックは使用されません。コールバック関数は、以下のイベント i2c_event_t で呼び出されます。
 - Transfer aborted
 - Transmit complete
 - Receive complete
- コールバック構造体 i2c_callback_args_t には、送信または受信したバイト数も設定されます。
- 以下によって実装されます。
 - Simple I²C on SCI
 - RIIC

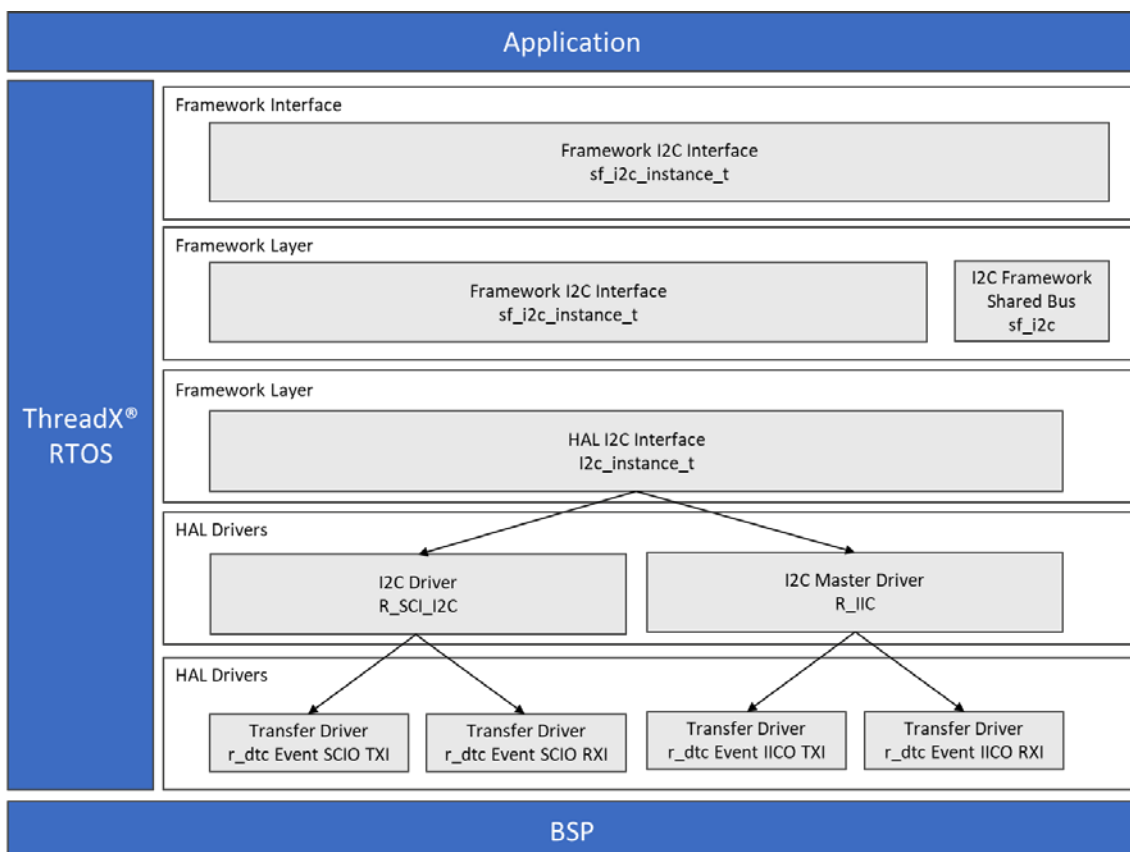


図 12.28 I²C フレームワーク

12.23 JPEG デコードフレームワーク

JPEG デコード HAL モジュールは、JPEG デコード処理のための汎用 API で、`r_jpeg` に実装されています (JPEG デコード HAL モジュールは、JPEG コーデック周辺機能をサポートします)。JPEG デコードフレームワークモジュールは JPEG フレームワークモジュールアプリケーションのための ThreadX 対応のハイレベルの API で、`sf_jpeg_decode` に実装されています。MCU グループ上の JPEG ハードウェアへのスレッドセーフなアクセスを提供します。ユーザー定義のコールバックを作成し、ハードウェアサポートイベントを検出できます。

このフレームワークの主な機能は以下のとおりです。

- Synergy JPEG ハードウェアへのスレッドセーフなアクセスを提供
- JPEG デコード HAL モジュールを使用した JPEG 解凍をサポート
- JPEG デコーダが完了するまでアプリケーションが待機できるようにするポーリングモードをサポート
- ユーザーが指定したコールバック関数を使用した割り込みモードをサポート
- 水平および垂直サブサンプル値、水平ストライド、デコードされたピクセルフォーマット、入力および出力データフォーマット、色空間などのパラメータを設定
- 画像をデコードする前にそのサイズを取得
- デコードされた画像フレームの保管のため、コード化されたデータを入力バッファおよび出力バッファに格納する操作をサポート
- JPEG デコーダモジュールへのコード化されたデータのストリーム転送をサポート。この機能により、アプリケーションは、ファイルやネットワークからのコード化された JPEG 画像を画像全体をバッファリングすることなく読み取ることが可能
- デコードする画像行数を設定。この機能により、アプリケーションは、デコードされた画像の処理を、フレーム全体をバッファリングすることなくオンザフライで実行可能
- 入力値のデコードされたフォーマットとして YCbCr444、YCbCr422、YCbCr420、YCbCr411 をサポート
- 出力値のデコードされたフォーマットとして ARGB8888 と RGB565 をサポート
- モーション JPEG (MJPEG) ビデオデコードに対して構成可能
- JPEG 画像のサイズ、高さ、幅が要件を満たさない場合にエラーを返すことが可能
- JPEG ハードウェアサポートイベントと同期するためのスレッドをサスペンド/再開するための `wait` API 関数をサポート

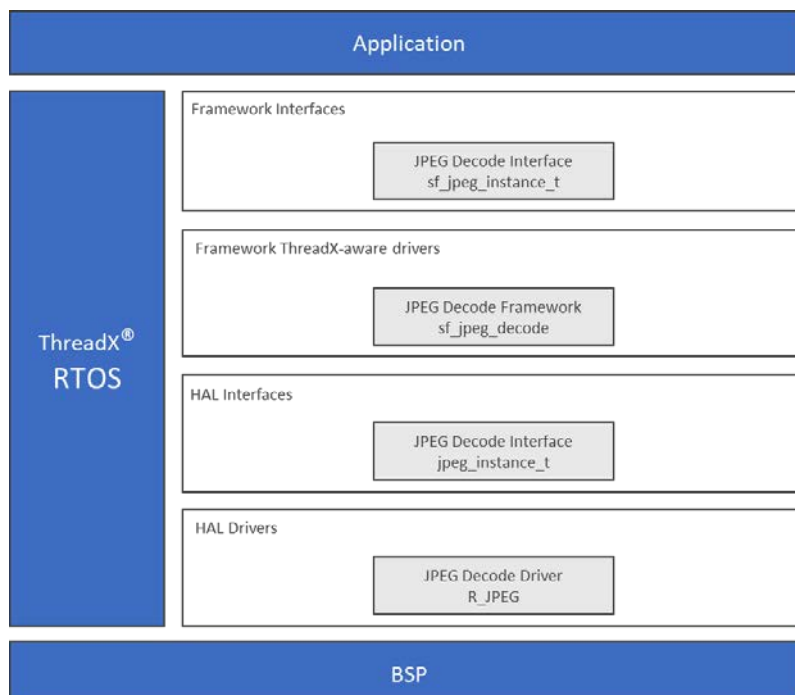


図 12.29 JPEG デコードフレームワーク

12.24 メッセージフレームワーク

メッセージフレームワークは `sf_message` 上で実装され、スレッド間のメッセージを送受信するイベント駆動型の軽量なフレームワーク API を提供します。メッセージフレームワークにより、アプリケーションは 2 つ以上のスレッド間でメッセージを送受信できるようになります。このフレームワークは、ThreadX のメッセージキュープリミティブを使用してメッセージを送受信し、ThreadX RTOS メッセージキューサービス単独の場合よりも多くの利点が得られます。メッセージフレームワーク API は純粋なソフトウェア API であり、ハードウェア周辺機能にはアクセスしません。メッセージフレームワークコールバックを使用すると、イベント作成元スレッドとメッセージサブスクリバースレッドは、メッセージの送受信後にハンドシェイクを行うことができます。

SSP コンフィギュレータでは、[メッセージング]タブを使用して、メッセージフレームワーク用の独自のカスタムイベントクラスやイベント、サブスクリバを作成したり、タッチパネルフレームワークモジュールによって使用されるタッチイベントなどの事前設定されたイベントをカスタマイズしたりできます。

このフレームワークの主な機能は以下のとおりです。

- スレッド間通信
異なるデバイスを制御したりサブシステムを管理したりするアプリケーションスレッドが互いに通信できます。
- パブリッシュ/サブスクライブ方式
フレームワークの設計は、疎結合型のメッセージング方式に基づいています。複数のスレッドが 1 つのイベントクラスを受信待ちできる設計になっています。メッセージ作成元スレッドは、イベントクラスのメッセージにサブスクライブしているスレッドを知る必要がありません。サブスクライブは、メッセージの作成元を知る必要がありません。
- メッセージ管理
フレームワークでは、バッファを制御するフラグやハンドシェイクのためのコールバック関数ポインタなど、各メッセージを管理するためのバッファ制御ブロックがサポートされています。
- メッセージバッファリング
フレームワークは、メッセージング用のバッファの割り当てと解放を管理します。アプリケーションは、割り当てられたバッファを利用してメッセージを書き込んだり、不要になったメッセージを破棄したりすることができます。
- 同期通信
フレームワークでは、ThreadX メッセージキューを使用した非同期メッセージングがサポートされていますが、メッセージ作成元とサブスクライバースレッドの間のハンドシェイクを確立するためのオプションも提供されています。ハンドシェイクは、サブスクライバースレッドから作成元スレッドのユーザーコールバック関数を呼び出すことで実装されています。
- メッセージ構造の指定
定義済みの共通メッセージヘッダーが提供されています。また、いくつかの典型的なペイロード構造体テンプレートも例として提供されています。
- メッセージのプライオリティ
ハイプライオリティのメッセージを送信して、サブスクライバースレッドが、メッセージキューにある他のメッセージの前にそのメッセージを受け取れるようにすることができます。

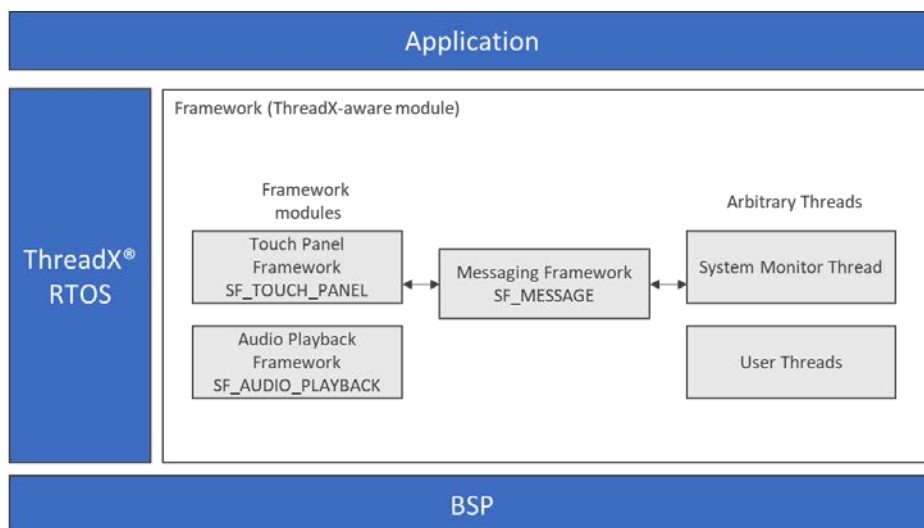


図 12.30 メッセージフレームワーク

12.25 パワープロファイルフレームワークバージョン 1

パワープロファイルフレームワークバージョン 1 は、「12.26 Power Profile Framework Version 2」で説明されているバージョン 2 によって置き換えられました。

このリリースでは、パワープロファイルフレームワークバージョン 1 のモジュールは非推奨ですが、以前の SSP リリースの期間に開発された既存のプロジェクトとの下位互換性を維持するために提供されています。置き換え用モジュールであるパワープロファイルフレームワークバージョン 2 の使用を強く推奨します。

パワープロファイルフレームワークは、より低電力のソフトウェアスタンバイモードに MCU を配置できるようにアプリケーションによって使用可能な汎用 API を提供します。このモジュールは sf_power_profiles に実装されます。

このフレームワークの主な機能は以下のとおりです。

- ソフトウェアスタンバイ操作をサポート
- 複数の動作モードをサポート
実行
RTC
外部割り込み
- ソフトウェアスタンバイ時に無効にするクロックと周辺機能を選択
- 既存のソフトウェアスタンバイの前後に出力ピン状態を選択

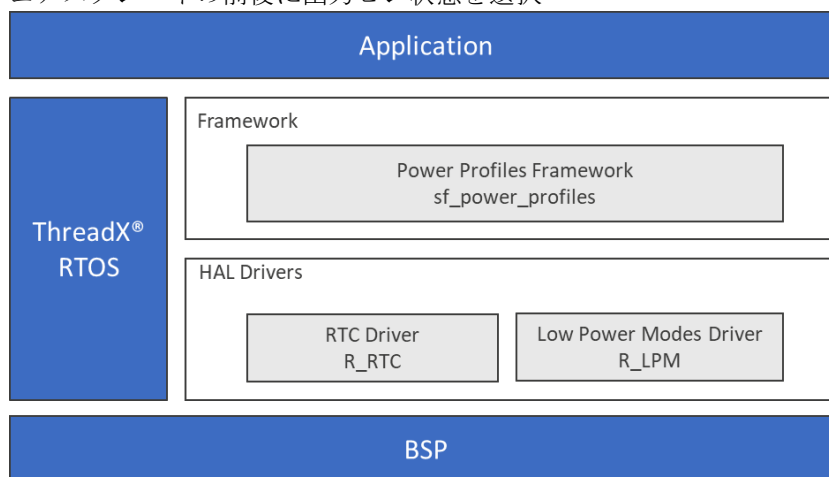


図 12.31 パワープロファイルフレームワークバージョン 1

12.26 パワープロファイルフレームワークバージョン 2

パワープロファイル V2 (PPv2) フレームワークにより、MCU でのユーザーによる電力制御モードと LPM モードの設定をより効果的に制御します。PPv2 フレームワークでは、LPM v1 ドライバーを使用して、現在のリリースの SSP フレームワークで使用可能なパワープロファイル v1 のすべての機能がサポートされます。パワープロファイル v1 と PPv2 フレームワークには互換性はありません。パワープロファイル v1 と PPv2 フレームワークは、同じプロジェクト内では使用できません。

すべての新規プロジェクトでは、アプリケーションで PPv2 フレームワークを使用することをお勧めします。

- カスタマイズ可能なクロックドメインでさまざまな MCU 電力制御モードを設定するための構成可能なオプション。
- 異なる IO ポートまたはピン構成でさまざまな MCU 低消費電力モードを設定するための構成可能なオプション。
- スレッド動作と非スレッド動作の両方をサポート

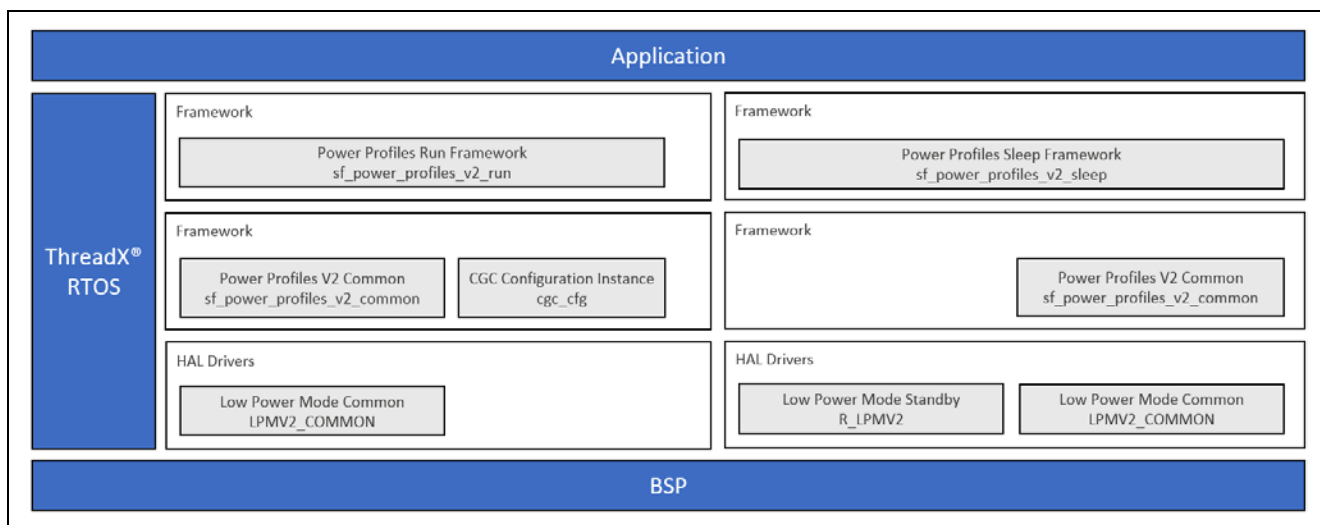


図 12.32 パワープロファイルフレームワークバージョン 2

注：ローレベル LPMV2 HAL モジュールはさまざまな MCU で使用できるようになっており、多くのオプションを用意しています。この図は、使用できるオプションをすべて示しているわけではありません。簡略化した<MCU>インジケータは実際のフレームワーク実装の CPU 名 (S7G2 など) で置換されます。

12.27 シリアルペリフェラルインタフェース (SPI) フレームワーク

SPI フレームワークは、ThreadX 対応の一連のフレームワーク API を提供し、sf_spi に実装されています。SPI フレームワークは、チップ選択処理とそのレベルのアクティブ化を含め、SPI バス上の複数の SPI 周辺機能の統合と同期を処理します。SPI フレームワークを使用すると、1 つ以上の SPI バスを作成し、複数の SPI 周辺機能をそれぞれの SPI バスに接続できます。SPI フレームワークモジュールは、単一のインタフェースを使用して、SCI SPI ドライバーと RSPI ドライバーの両方にアクセスします。SPI フレームワークモジュールは、MCU 上の SCI および RSPI 周辺機能を使用します。

SPI フレームワークモジュールは、SPI モードの SCI を (SCI 共通低レベルモジュールとともに) 使用するか RSPI ローレベルドライバーモジュールを使用して、MCU 上の SPI 周辺機能と通信します。

このフレームワークの主な機能は以下のとおりです。

- バス上の複数のデバイスをサポート
- モジュールの初期化、転送、クローズ用のハイレベルの API を提供
- 同期転送をサポート
- チップ選択操作をサポート
- バスロックをサポート

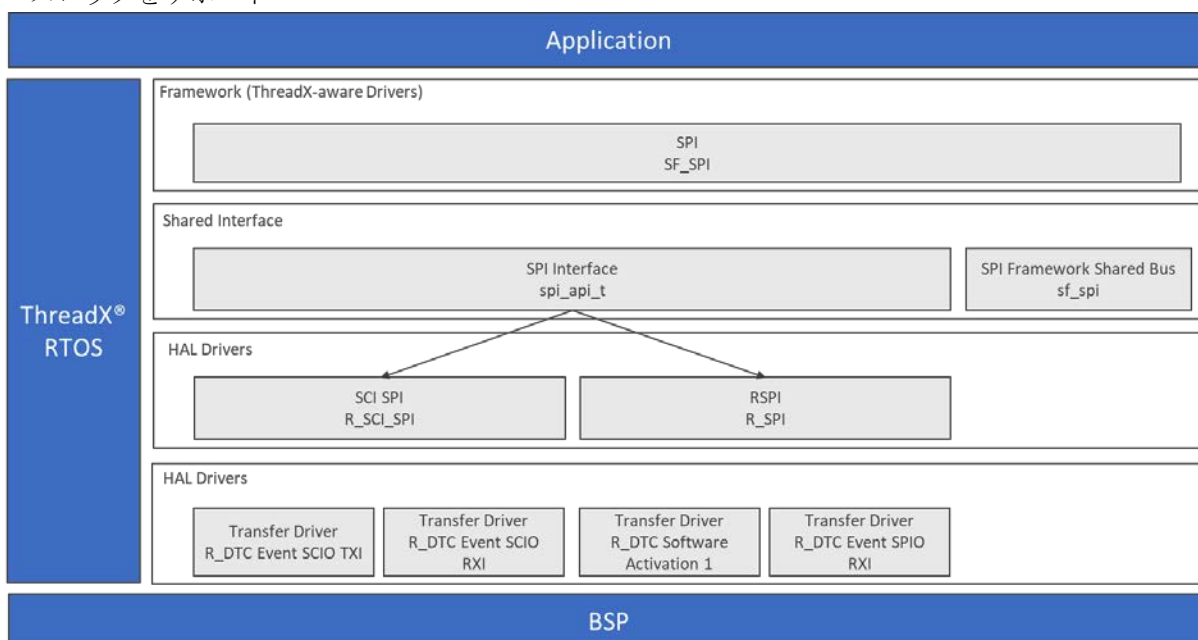


図 12.33 シリアルペリフェラルインタフェース (SPI) フレームワーク

12.28 Synergy FileX ポートブロックメディアインタフェースフレームワーク

sf_el_fx 上に実装されている FileX ポートブロックメディアフレームワークモジュールは、Express Logic FileX システムをサポートします。FileX は、ディープエンベデッドアプリケーション用の完全な FAT フォーマットメディアであり、ファイル管理システムでもあります。FileX ポートブロックメディアモジュールは、FileX がブロックメディアインタフェースと適応レイヤーを介して Synergy メディアドライバーにアクセスするための I/O コールを提供します。SD/MMC HAL モジュールと SDIO HAL モジュールは r_sdmmc 上に実装され、SD/MMC メディアデバイスならびに SDIO カードの読み取り/書き込みと制御に使用されます。sf_el_fx モジュールは、MCU 上の SD/MMC 周辺機能を使用します。

このフレームワークの主な機能は以下のとおりです。

- FileX ポートブロックメディアフレームワークモジュールの特長
 - FAT32、FAT16、FAT12 をサポート
 - 特定のメディアでの最初の FAT パーティションのマウントをサポート
 - 無制限の FileX オブジェクト（メディア、ディレクトリ、ファイル）
 - 動的 FileX オブジェクト作成と削除
 - 柔軟なメモリ使用
 - サイズを自動的に拡張
 - 小さなフットプリント
 - ThreadX との完全な統合
- SD カードインタフェース
 - SD、SDHC、SDXC フォーマットと互換
 - 1 ビットと 4 ビットのバス幅をサポート
 - ハードウェアでサポートされる場合のカード検出機能
 - ライトプロテクトのサポート
- eMMC インタフェース
 - 1 ビット、4、ビット、8 ビットのバス幅をサポート
- SD バスインタフェース
 - SD メモリカードおよび SDIO カードと互換
 - 転送バスモードを 4 ビットのワイドバスモードまたは 1 ビットのデフォルトバスモードから選択可能
 - SD、SDHC、SDXC フォーマットと互換
- SD および MMC 共有
 - SBFAI 割り込み SD バッファによってトリガできる DMAC および DTC は、DMAC を使用してリードおよびライトアクセス可能

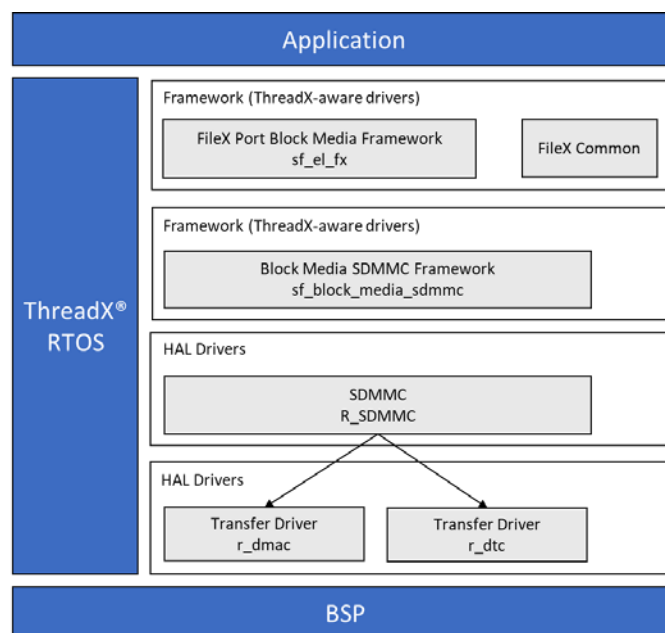


図 12.34 Synergy FileX インタフェースフレームワーク

12.29 Synergy GUIX インタフェースフレームワーク

Express Logic 社の GUIX Synergy ポートモジュール `sf_el_gx` は、GLCDC、DRW (2DG エンジン)、または JPEG デコードエンジンを搭載する MCU 向けの Express Logic GUIX 適応レイヤーです。この API では、GUIX 用のグラフィックハードウェアエンジン設定、グラフィックスのレンダリング、およびハードウェアエンジンによって加速されるグラフィックスの表示をサポートします。このモジュールでは、GUIX ローレベルディスプレイドライバ機能のフルセットを定義します。DRW (2DG エンジン) または JPEG によって加速されるグラフィックスを描画したり、GLCDC を使用してグラフィックスを表示したりします。このモジュールは、グラフィックスのレンダリングのためにハードウェア加速を促進するだけでなく、ハードウェアのサポートなしでソフトウェア処理も可能にします。

このフレームワークの主な機能は以下のとおりです。

- GUIX を SSP フレームワークに適応させる
- SSP ディスプレイインタフェースドライバを GUIX ディスプレイドライバインタフェースに接続する
- 2D Drawing Engine (DRW) エンジンにより、GUIX は高速でウィジェットを描画可能
- Synergy JPEG エンジンにより、GUIX は高速でウィジェットを描画可能
- 切れ目が発生することなく画面の移行を可能とするダブルバッファ切り替え制御をサポート
- 画面のローテーションをサポート (90/180/270 度)
- 各種の色出力形式をサポート
 - 32bpp (ARGB8888、RGB-888)
 - 16bpp (RGB565)
 - 8bpp、8 ビットパレットカラールックアップテーブル (CLUT)
- ユーザーコールバック関数をサポート

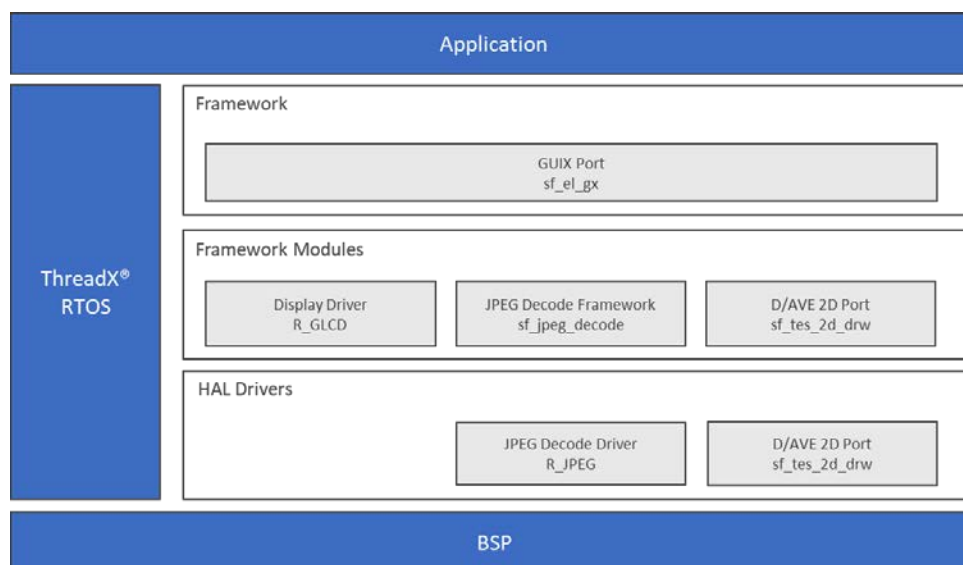


図 12.35 Synergy GUIX インタフェースフレームワーク

12.30 Synergy NetX ポートイーサネットモジュール

NetX および NetX Duo 用 Synergy NetX ポートイーサネットモジュール (sf_el_nx) は SSP に統合されています。このモジュールの機能は、汎用 NetX および NetX Duo ソフトウェアとハードウェアの間のやり取りを行うことです。このモジュールは、MAC ドライバー、PHY ドライバー、その他のグルーロジックおよびユーティリティ機能などを備えています。

注：特に明記されていない限り、NetX プロジェクトと NetX Duo プロジェクトにおけるこのモジュールの動作に違いはありません。

このモジュールの主な機能は以下のとおりです。

- NetX サービスはライブラリとして実装されているため、必要なコードだけがプロジェクトに追加されます。
 - これにより、ほとんどのアプリケーションでは、命令イメージは 5k バイトから 30k バイトであり、IPv6 および ICMPv6 を有効にするとサイズは 30k バイトから 45k バイトです。
- NetX は次のようなさまざまな RFC をサポートしています。
 - RFC 1112 IP マルチキャスト用ホスト拡張機能 (IGMPv1)
 - RFC 1122 インターネットホストに関する要件—通信レイヤー
 - RFC 2236 インターネットグループ管理プロトコルバージョン 2
 - RFC 768 ユーザーデータグラムプロトコル (UDP)
 - RFC 791 インターネットプロトコル (IP)
 - RFC 792 インターネット制御通知プロトコル (ICMP)
 - RFC 793 伝送制御プロトコル (TCP)
 - RFC 826 イーサネットアドレス解決プロトコル (ARP)
 - RFC 903 逆アドレス解決プロトコル (RARP)
 - RFC 2460 インターネットプロトコル v6 (IPv6) 仕様 (NetX Duo のみ)
 - RFC 4443 インターネット制御通知プロトコル (ICMPV6) (NetX Duo のみ)
 - RFC 4861 IPv6 近隣探索 (NetX Duo のみ)
 - RFC 4862 IPv6 ステートレスアドレス自動構成 (NetX Duo のみ)
- TCP/IP のパケットベースのゼロコピー実装。バッファはスタック内またはスタックとユーザーアプリケーションの間を移動するので NetX の内部ではコピーされず、処理サイクルとメモリが解放されて、送信速度が大幅に向上します。
- ファスト UDP 処理

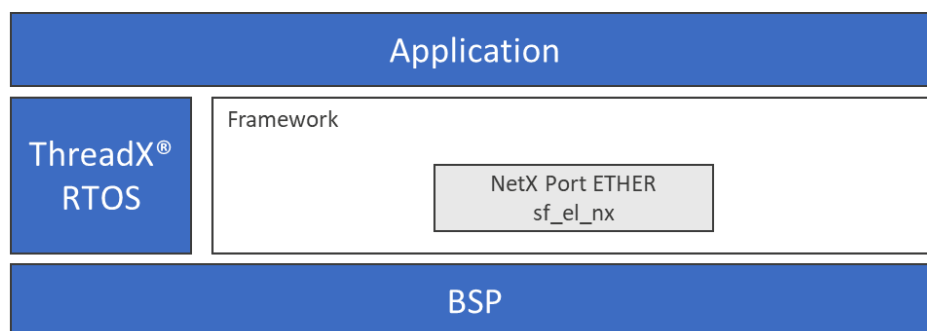


図 12.36 NetX ポートイーサネットモジュールの構成、オプションおよびスタックの実装

12.31 Synergy USBX ポートフレームワーク

Express Logic 社の USBX Synergy ポートフレームワークモジュール (sf_el_ux) は SSP に統合されています。このドライバーは、Express Logic 社の USBX で使用するためのものです。

このフレームワークの主な機能は以下のとおりです。

- Express Logic USBX を SSP サポート USBX API に実装
- USBHS 周辺機能用のポートデバイスコントローラドライバー (DCD) をサポート
- USBFS 周辺機能用のポートデバイスコントローラドライバー (DCD) をサポート
- USBHS 周辺機能用のポートホストコントローラドライバー (HCD) をサポート
- USBFS 周辺機能用のポートホストコントローラドライバー (HCD) をサポート
- 転送モジュールの動作をサポート (オプション)

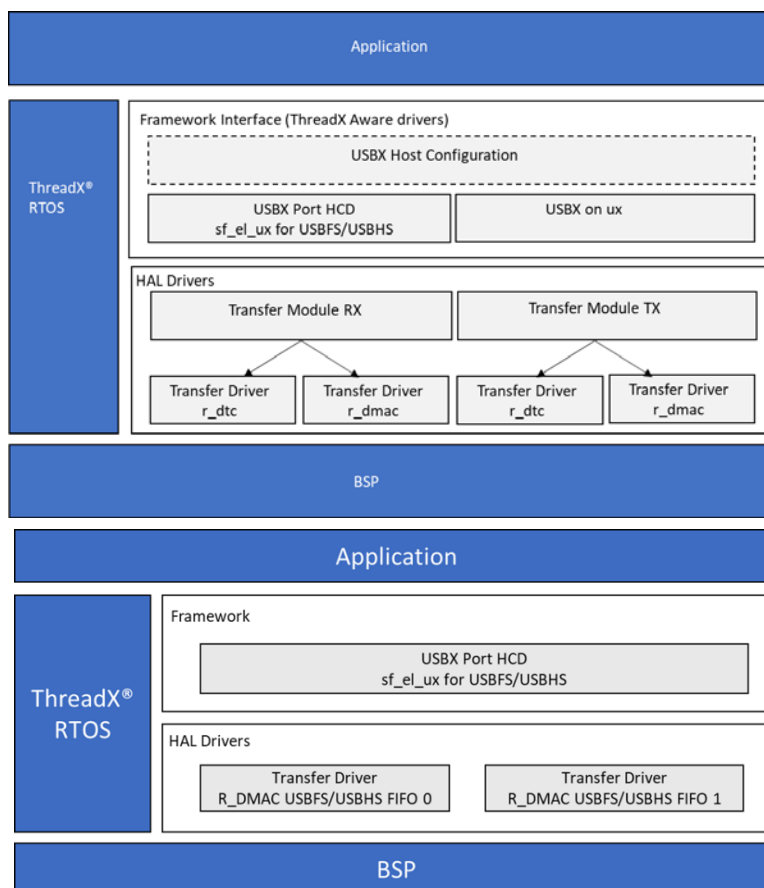


図 12.37 DCD および HCD 用 Express Logic USBX Synergy ポートフレームワークの構成、オプション、スタックの実装

12.32 スレッドモニタフレームワーク

スレッドモニタフレームワークは、スレッドモニタフレームワークアプリケーション用のハイレベルのAPIを提供し、`sf_thread_monitor` に実装されています。このフレームワークは、ウォッチドッグタイマ (WDT) または独立ウォッチドッグタイマ (IWDT) を構成します。スレッドモニタフレームワークは、MCU上のWDTまたはIWDT周辺機能を使用します。

このフレームワークの主な機能は以下のとおりです。

- スレッドモニタフレームワークインタフェースは、ウォッチドッグタイマを使用して RTOS スレッドを監視します。スレッドモニタは、監視対象のスレッドが想定外の動作を行った場合、ウォッチドッグによるMCUのリセットを強制的に実行します。
- スレッドモニタは、API への変更なしに、WDT または IWDT 周辺機能および HAL モジュールを使用して、すべてのMCUをサポートするように設計されています。
- プロファイリングモードでは、登録されたスレッドのカウンタの最小値および最大値を決定できます。プロファイリングモード中は、ウォッチドッグタイマが常にリフレッシュされるため、デバイスはリセットされません。
- このフレームワークモジュールは、WDT と IWDT HAL モジュールの両方をサポートしています。

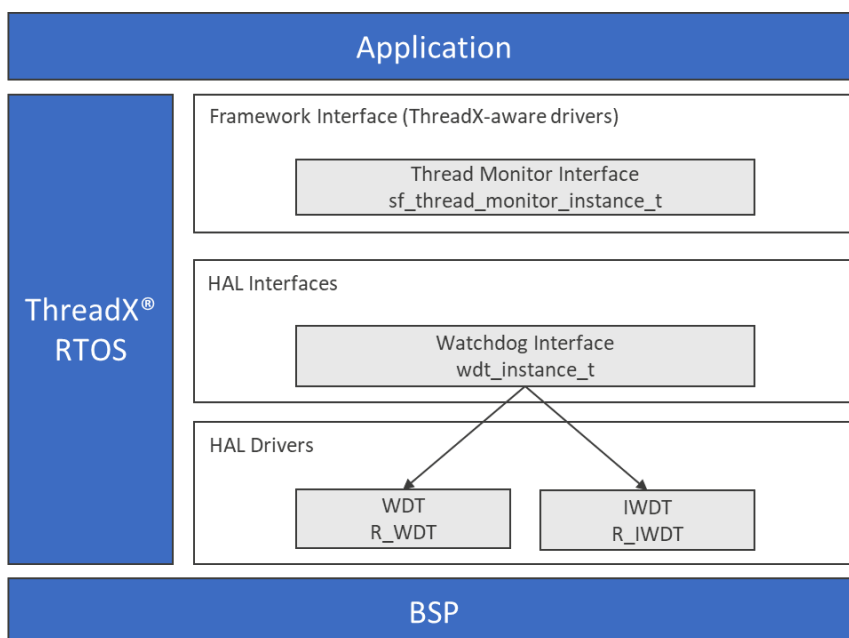


図 12.38 スレッドモニタフレームワーク

12.33 タッチパネル I2C フレームワーク

タッチパネル I2C フレームワークは、タッチコントローラからメッセージを読み取るためのハイレベルの API を提供し、I2C ポートを使用して実装されています。タッチパネルフレームワークは、MCU 上の IIC または SCI 周辺機能を使用します。

このフレームワークの主な機能は以下のとおりです。

- タッチコントローラからデータを読み取り、メッセージフレームワークのタッチイベントにサブスクライブされたキューに対してタッチメッセージをパブリッシュする
- 位置データ (X および Y 座標) を提供する
- タッチイベントタイプ (下、上、移動、保持、無効) を提供する
- 外部割り込み要求による I2C ベースのタッチパネルの実装をサポートする

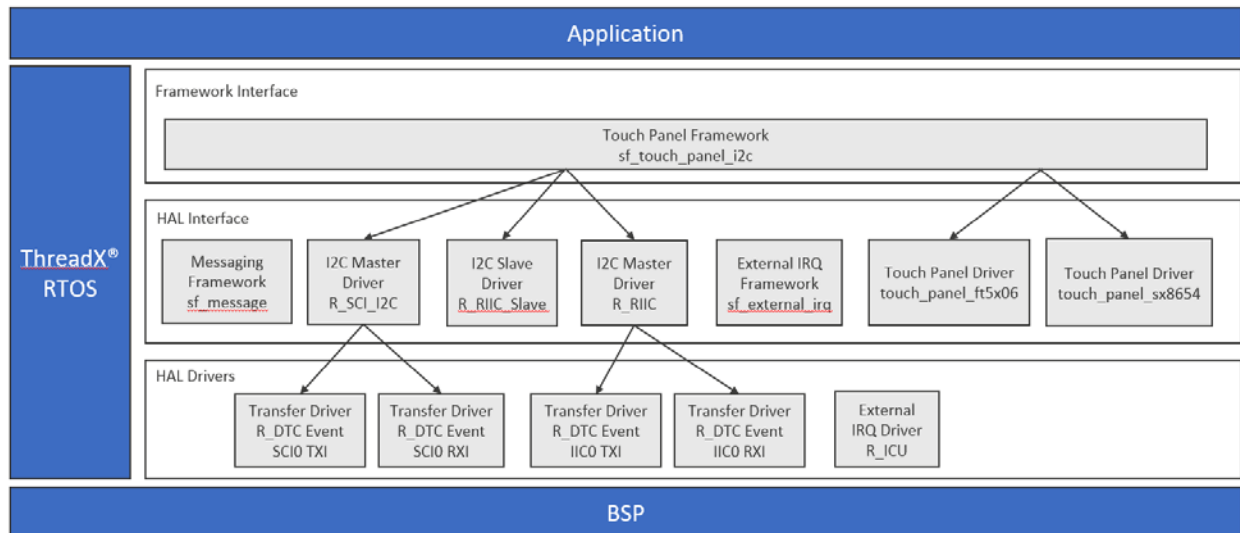


図 12.39 タッチパネル I2C フレームワーク

12.34 UART 通信フレームワーク

UART 通信フレームワークは、UART 準拠の MCU 周辺機能上で Synergy 通信フレームワーク API のインスタンスを提供します。このフレームワークでは、r_sci_uart HAL ドライバーを使用します。このドライバーは MCU の SCI 周辺機能を構成して操作することで UART プロトコルによる通信を行います。

このフレームワークは ThreadX 対応であり、ThreadX オブジェクトを使用して動作がスレッドセーフになるようにします。

UART 通信フレームワークモジュールは、以下の機能に対応します。

- UART 通信プロトコル
- 専用アクセスを確保するためのチャンネルロック
- ThreadX 対応実装

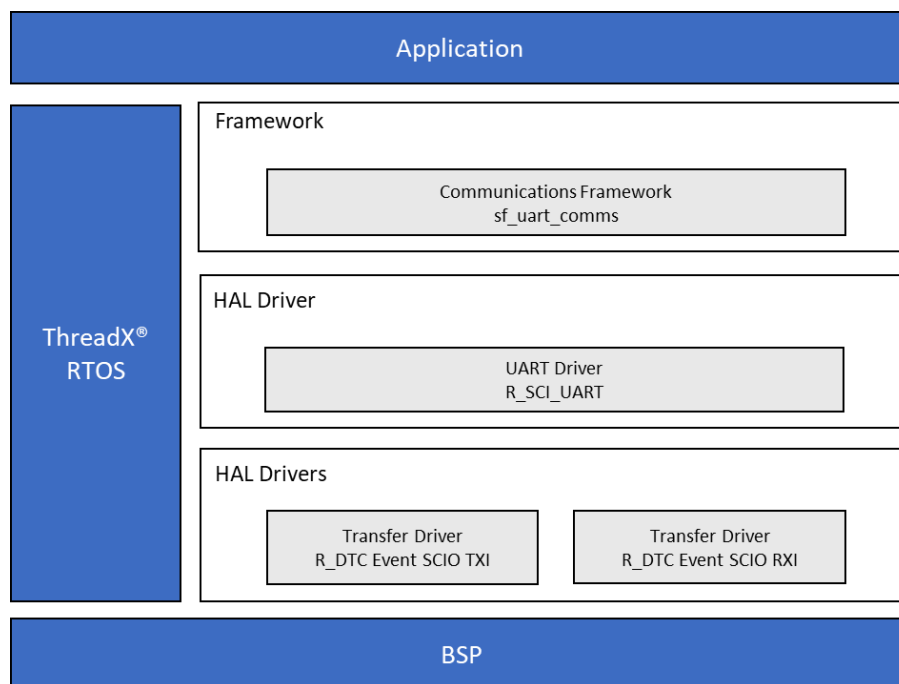


図 12.40 UART 通信フレームワーク

12.35 Wi-Fi フレームワーク

Wi-Fi フレームワークが提供するハイレベルの API モジュールによって、Wi-Fi モジュールの構成とプロビジョニングを行い、オンチップネットワーク機能の有無にかかわらずデータ転送を実行できます。現在、Qualcomm GT202 モジュールのみがサポートされています。Wi-Fi フレームワークは、SPI を介して基礎となる GT202 モジュールと通信します。

このフレームワークの主な機能は以下のとおりです。

- Wi-Fi モジュールを構成し、プロビジョニングするためのハイレベルの API を提供する
- 以下に対して4つの異なる実装を提供する
 - sf_wifi_gt202 フレームワークを使用した Wi-Fi デバイスドライバースタック
 - sf_wifi_onchip_stack フレームワークを使用したオンチップスタック
 - sf_wifi_onchip_stack フレームワークを使用した BSD ソケットスタック
 - sf_wifi_nsal_nx フレームワークを使用した NetX および NetX Duo ポート
- NetX および NASL の使用
 - 同じアプリケーションコードを異なる Wi-Fi モジュールに使用可能
 - イーサネットベースのアプリケーションを Wi-Fi ベースのアプリケーションに容易に移行可能
 - アプリケーションの要件に従ってアプリケーションと TCP/IP スタックのデバッグと微調整が可能
 - 現在の NSAL 実装では、NetX NSAL のみを提供。新しいネットワークスタックのサポートを追加するには、適切な NSAL の実装が必要
- オンチップネットワークスタックの使用
 - MCU を小さなメモリフットプリントで使用する場合に有益
 - オンチップ TCP/UDP を使用してソケットベースのアプリケーションを作成するための BSD ソケットインタフェースを提供
 - NetX スタックを使用せずに TCP/IP 上の MQTT および COAP などのサードパーティアプリケーションプロトコルを統合するオプションを提供

注：このオンチップネットワークスタックは、Wi-Fi モジュールまたは Wi-Fi モジュールドライバーでサポートされている場合にのみサポートされます。

以下の図に、Wi-Fi フレームワークモジュールの編成、オプション、スタックの実装を示します。

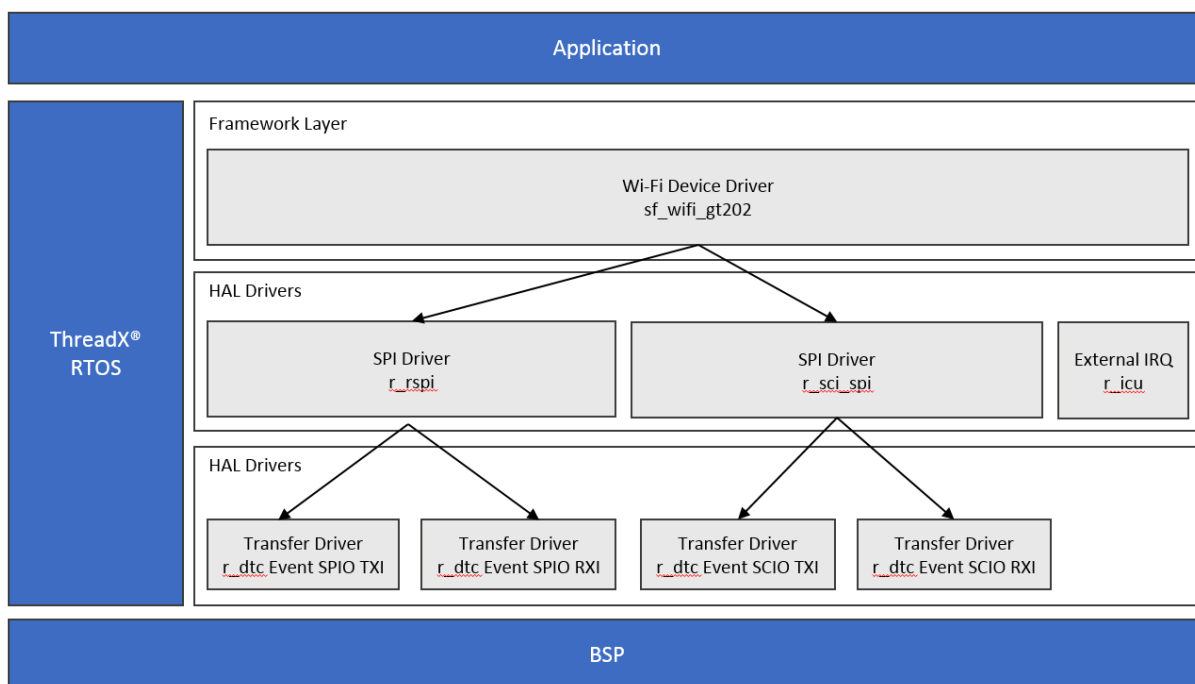


図 12.41 Wi-Fi デバイスドライバの実装

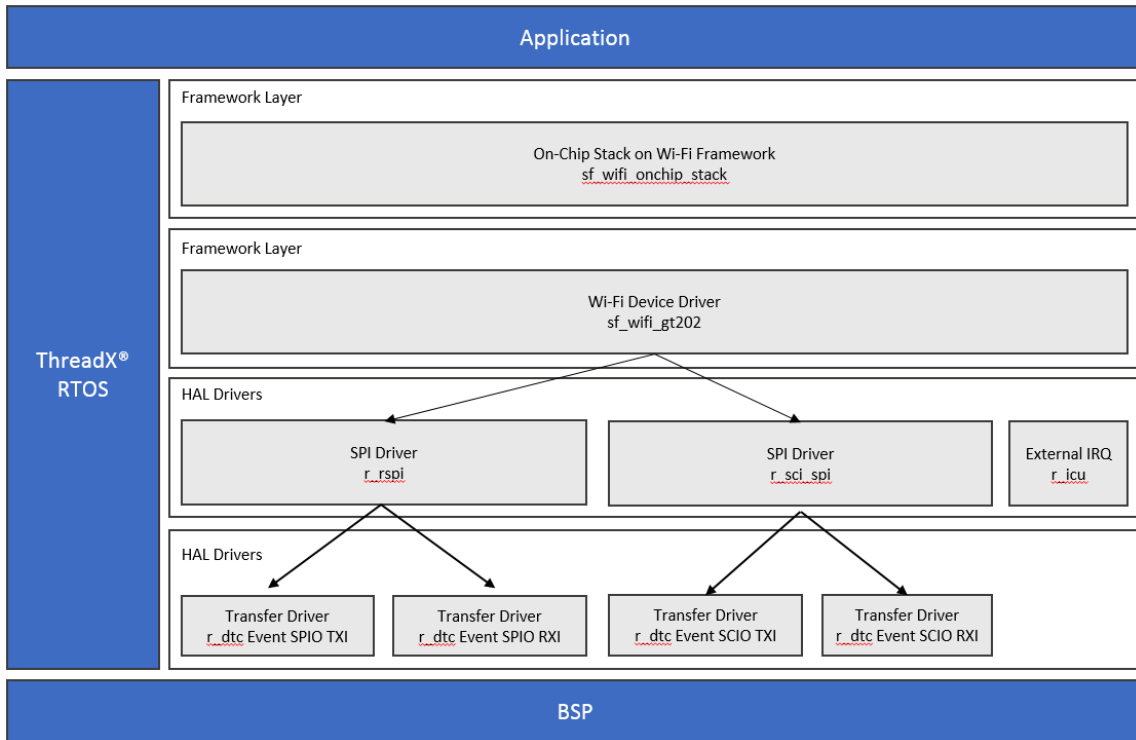


図 12.42 Wi-Fi オンチップスタックの実装

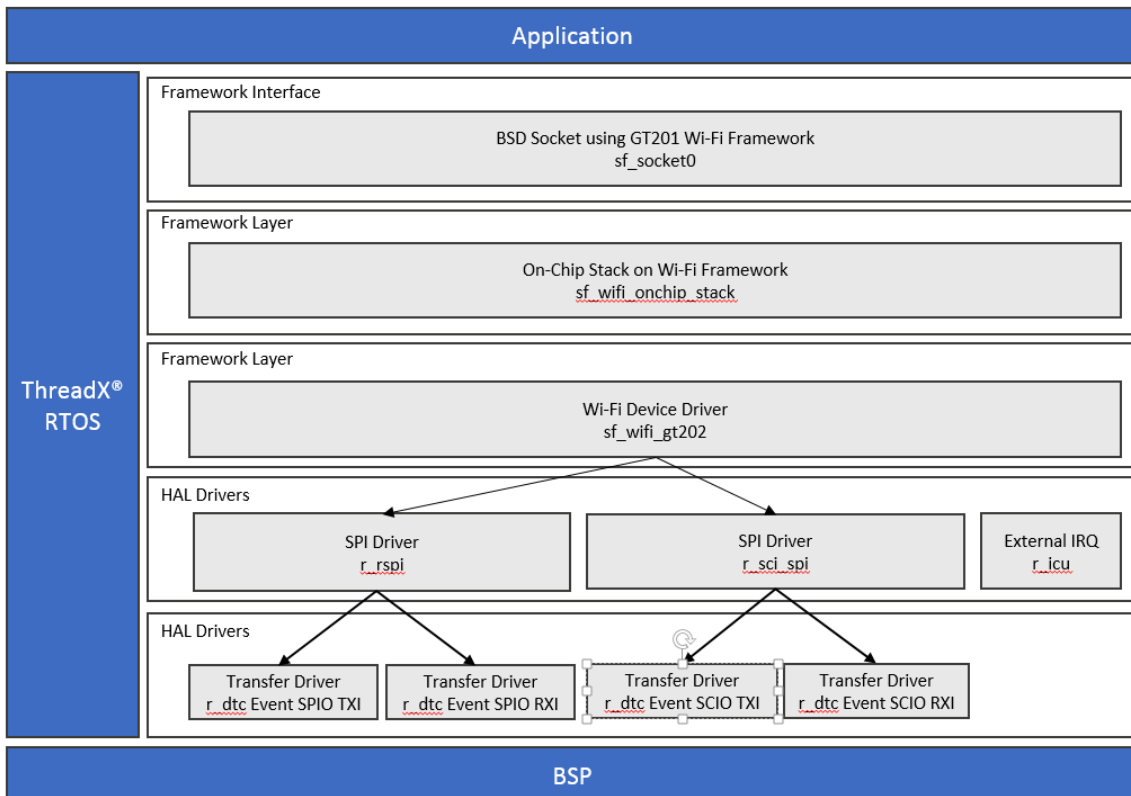


図 12.43 Wi-Fi BSD ソケットの実装

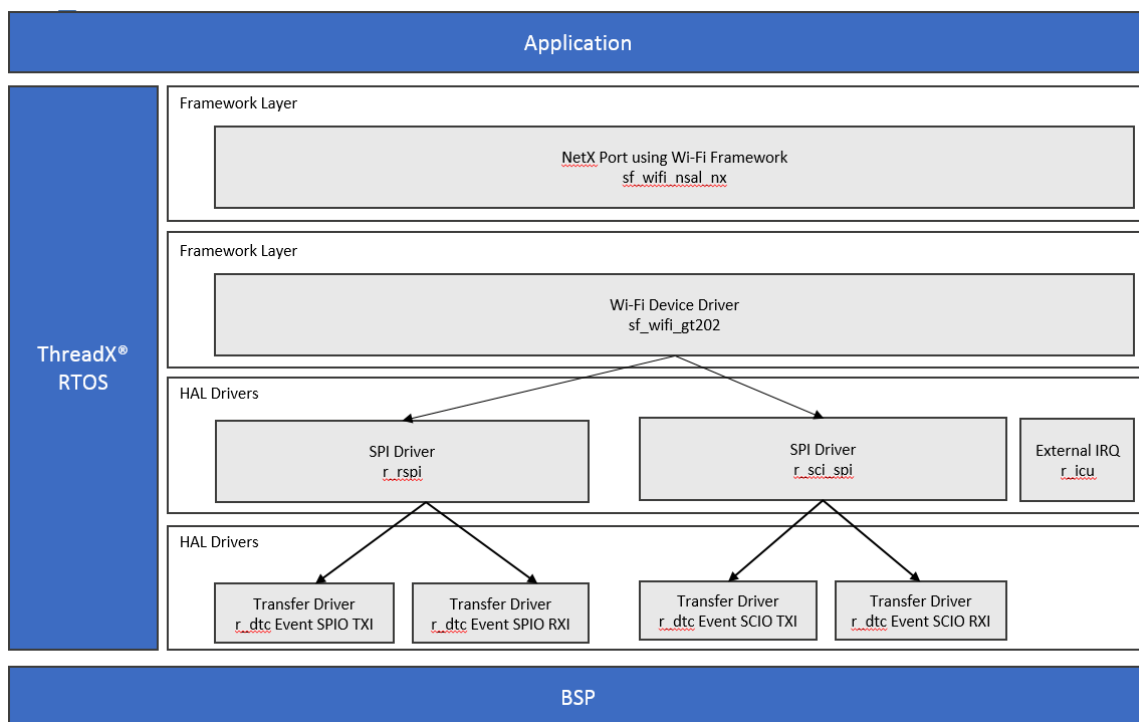


図 12.44 NSAL での Wi-Fi NetX ポートの実装

13. CMSIS DSP ライブラリ

Synergy MCU に搭載される Arm Cortex-M4 コアの Arm Cortex Microcontroller Software Interface Standard DSP ハードウェアブロック (CMSIS-DSP) は、一連の共通信号処理機能を提供します。

CMSIS-DSP ライブラリは、Synergy MCU のための SSP に含まれる HAL レイヤーであり、デジタル信号制御アプリケーションで使用される 60 以上の完全に最適化された信号処理機能が含まれています。このライブラリは、DSP 処理のための固定小数点/分数 (Q7、Q15、Q31) や単精度浮動小数点 (32 ビット) 演算などの主要な演算フォーマットをサポートしています。SSP の高効率な信号処理機能と MCU および基礎となる SIMD アーキテクチャと FPU によるローパワー、低コスト、ハイパフォーマンスのメリットの組み合わせにより、IoT/M2M 市場向けのさまざまなアプリケーション開発にとって高い効果を発揮することができます。

CMSIS-DSP ライブラリは主に次の種類の処理に対応します。

分類	機能
基本数学	絶対値、加算、ドット積、乗算、否定、オフセット、スケール、シフト、減算
共通テーブル	FFT 回転因子、ビット反転、逆数
複雑数学	共役、ドット積、絶対値、絶対値の 2 乗、複素数と複素数の乗算、複素数と実数の乗算
高速数学	コサイン、サイン、平方根
フィルタ	Q31、IIR、FIR、CRC 演算切り替え (LMS)、コンボリューション、相関関係
行列数学	加算、乗算、初期化、反転、スケール、減算、移項
統計	最大、平均、最小、検出力、RMS、標準偏差、分散
変換	複素 FFT、Radix-8 複素 FFT、DCT タイプ IV、実数 FFT、複素 FFT テーブル、実数 FFT

14. ハードウェア抽象化レイヤー (HAL) モジュール

14.1 概要

SSP の HAL モジュールは、MCU で使用可能な周辺機能のためのデバイスに依存しないドライバーです。HAL モジュールは、抽象化されて十分に定義されたインタフェースを提供します。インタフェースの基盤となる機能は、複数のデバイスドライバーで実装されています。HAL ドライバーはタイマなどのシステムサービスを使用し、汎用的なハイレベルの C コール可能なインタフェースを提供します。これらはデバイス非依存で機能します。SSP のアプリケーションフレームワークは、HAL ドライバーを使用してデバイス固有のドライバーとのインタフェースを取ります。HAL ドライバーをアプリケーションプログラムから使用して、SSP フレームワークをバイパスして各周辺機能と直接インタフェースを取ることも可能です。ただし、これらのモジュールは RTOS 非依存 (ThreadX 対応ではない) のため、スレッドセーフではありません。

本データシートでは、Synergy MCU でサポートされる主な機能および SSP HAL ソフトウェアでサポートされる機能を説明します。

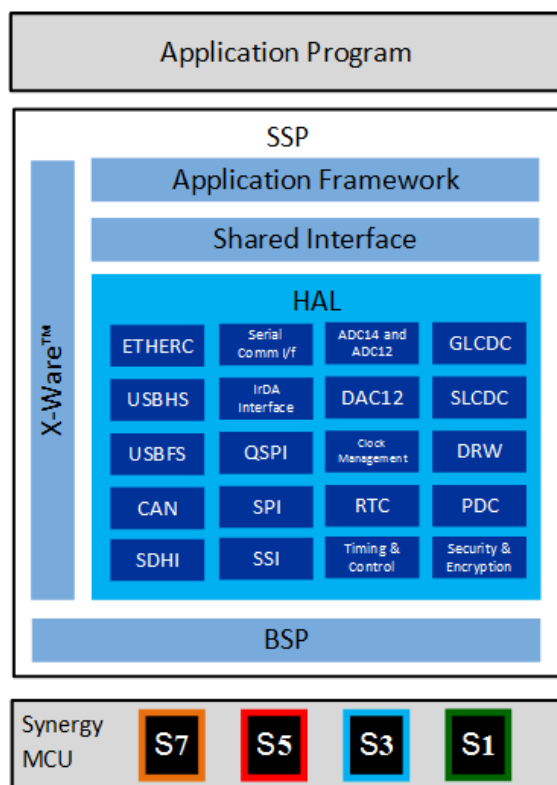


図 14.1 ハードウェア抽象化レイヤー (HAL)

14.2 SSP v1.4.0 で利用可能なモジュールの一覧

サポートするモジュールは、次の基準に従ってそれぞれ対応する MCU で利用可能です。

1. モジュールの主要機能がテスト済みで、MCU 上で動作する場合、既知のバグが存在しても、そのモジュールは MCU でサポートされます。
2. 主要機能が MCU 上で正しく動作しないまたはテストされていない場合、そのモジュールは該当の MCU でサポートされません。
3. モジュールがいずれかの MCU でテスト済みで、基盤となるハードウェアまたは HAL ドライバーと独立している場合、そのモジュールはすべての MCU について、そのモジュールが使用する基礎となるドライバー、フレームワーク、スタックがその MCU 上で完全にテスト済みなら、サポートされます。

HAL ドライバー名	機能	サポート対象の MCU
r_acmphs	高速アナログコンパレータ	S1JA

HAL ドライバー名	機能	サポート対象の MCU
r_acmplp	低消費電力アナログコンパレータ	S1JA
r_adc	A/D コンバータ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_agt	非同期汎用タイマ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_cac	クロック周波数精度測定回路	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_can	コントローラエリアネットワーク	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_cgc	クロック発生回路	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_crc	CRC 演算器	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_ctsu	静電容量式タッチセンシングユニット	S7G2、S5D9、S5D5、S3A7、S3A3、S124、S128
r_dac	D/A コンバータ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124
r_dac8	8 ビット D/A コンバータ	S3A3、S128、S1JA
r_dmac	ダイレクトメモリアクセスコントローラ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1
r_doc	データ演算回路	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_dtc	データトランスファコントローラ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_elc	イベントリンクコントローラ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_flash_hp	フラッシュメモリ (ハイパフォーマンス)	S7G2、S5D9、S5D5
r_flash_lp	フラッシュメモリ (ローパワー)	S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_fmi	ファクトリーMCU 情報	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_glcd	グラフィックス LCD コントローラ	S7G2、S5D9
r_gpt	汎用 PWM タイマ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_gpt_input_capture	汎用インプットキャプチャ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_icu	割り込みコントローラユニット	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_ioport	汎用 I/O ポート	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_iwdt	独立ウォッチドッグタイマ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_jpeg_decode	JPEG デコード	S7G2、S5D9
r_jpeg_encode	JPEG エンコード	S7G2、S5D9
r_kint	キー割り込み	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_lpm ^{注1}	ローパワーモード	S7G2、S3A7、S124
r_lpmv2_s1ja	S1JA 用ローパワーモード V2	S1JA
r_lpmv2_s124	S124 用ローパワーモード V2	S124
r_lpmv2_s128	S128 用ローパワーモード V2	S128
r_lpmv2_s3a1	S3A1 用ローパワーモード V2	S3A1
r_lpmv2_s3a3	S3A3 用ローパワーモード V2	S3A3
r_lpmv2_s3a6	S3A6 用ローパワーモード V2	S3A6

HAL ドライバー名	機能	サポート対象の MCU
r_lpmv2_s3a7	S3A7 用ローパワーモード V2	S3A7
r_lpmv2_s5d5	S5D5 用ローパワーモード V2	S5D5
r_lpmv2_s5d9	S5D9 用ローパワーモード V2	S5D9
r_lpmv2_s7g2	S7G2 用ローパワーモード V2	S7G2
r_lvd	低電圧検出	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_opamp	オペアンプ	S128、S1JA
r_pdc	パラレルデータキャプチャユニット	S7G2、S5D9、S5D5
r_qspi	クアッドシリアルペリフェラルインタフェース	S7G2、S5D9、S5D5、S3A7、S3A3、S3A1
r_riic_master	I ² C バスインタフェース (マスタ)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_riic_slave	I ² C バスインタフェース (スレーブ)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_rspi	シリアルペリフェラルインタフェース	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_rtc	リアルタイムクロック	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_sce ^{注2} この表の後に示す暗号化モジュールの機能の表を参照してください。	暗号化ライブラリ (Crypto HAL)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_sci_i2c	シリアルコミュニケーションインタフェース (I2C)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_sci_spi	シリアルコミュニケーションインタフェース (SPI)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_sci_uart	シリアルコミュニケーションインタフェース (UART)	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA
r_sdadc	シグマデルタ A/D コンバータ	S1JA
r_sdmmc	SDIO および SD/MMC メモリデバイス用の SDHI ドライバー	S7G2、S5D9、S5D5、S3A7、S3A3、S3A1
r_slcdc	セグメント LCD コントローラ	S3A7、S3A6、S3A3、S3A1
r_ssi	(Inter IC Sound) インタフェース [旧シリアルサウンドインタフェース] または r_i2s	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1
r_wdt	ウォッチドッグタイマ	S7G2、S5D9、S5D5、S3A7、S3A6、S3A3、S3A1、S124、S128、S1JA

注 1 このモジュールは、SSP v1.3.0 以降のバージョンでは廃用となったもので、そのモジュールを使用している可能性がある既存のプロジェクトとの互換性を維持する目的でのみ利用可能です。新しいプロジェクトではこれら廃用となったモジュールを使用せず、代わりに推奨される代替品を使用することを強くお勧めします。詳細は「SSP v1.4.0 ユーザーズマニュアル」を参照してください。

注 2 暗号化機能の詳細については表 14.1 を参照してください。なお、表に示す暗号化機能は r_sce 暗号化ライブラリの一部として利用可能です。

14.3 高速アナログコンパレータ (ACMPHS)

高速アナログコンパレータ (ACMPHS) HAL モジュールでは、`r_acmphs` 上の信号処理アプリケーション用のコンパレータ API を実装します。このモジュールは、MCU 上で利用可能な ACMPHS 周辺機能をサポートします。コールバックを使用して状態遷移イベントでユーザーアプリケーションに信号を送信できます。

- 立ち上がりエッジ、立ち下がりエッジ、または両方でのコールバック
- 構成可能なデバウンスフィルタ
- VCOUNT ピンでコンパレータ出力を含めるオプション

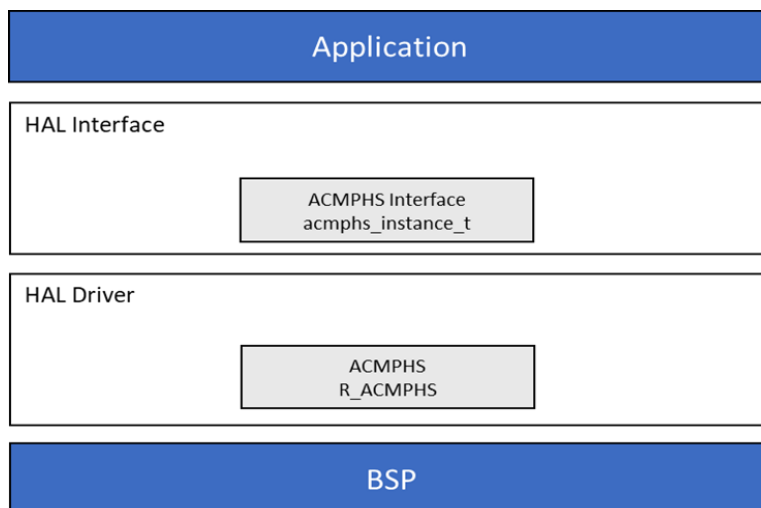


図 14.2 ACMPHS HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の ACMPHS の機能は以下のとおりです。

【記号の説明】

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	HS コンパレータ結果のクエリ	立ち上がりエッジ、立ち下がりエッジ、または両方でのコールバック	構成可能なデバウンス	VCOUNT でコンパレータ出力を含めるオプション
S124	N/A	N/A	N/A	N/A
S128	☒	☒	☒	☒
S1JA	✓	✓	✓	✓
S3A1	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A
S3A7	☒	☒	☒	☒
S5D5	☒	☒	☒	☒
S5D9	☒	☒	☒	☒
S7G2	☒	☒	☒	☒

14.4 低消費アナログコンパレータ (ACMPLP)

ACMPLP HAL モジュールでは、`r_acmplp` 上の信号処理アプリケーション用のコンパレータ API を実装します。このモジュールは、MCU 上で利用可能な ACMPLP 周辺機能をサポートします。コールバックを使用して、状態遷移イベントでユーザーアプリケーションに信号を送信できます。

- ノーマルモードまたはウィンドウモード

- 立ち上がりエッジ、立ち下がりエッジ、または両方でのコールバック
- 構成可能なデバウンスフィルタ
- VCOUNT ピンでコンパレータ出力を含めるオプション

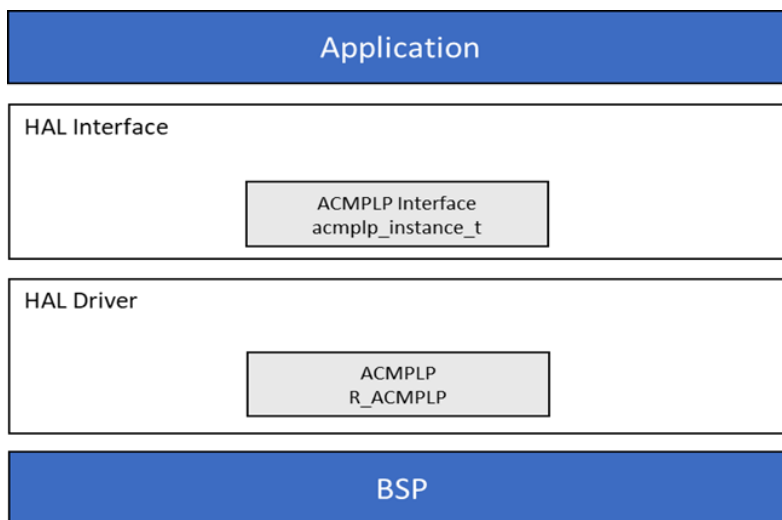


図 14.3 ACMPLP HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の ACMPLP の機能は以下のとおりです。

【記号の説明】

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	ノーマルまたはウィンドウモード	立ち上がりエッジ、立ち下がりエッジ、または両方でのコールバック	構成可能なデバウンス	VCOUT でコンパレータ出力を含めるオプション
S124	☒	☒	☒	☒
S128	☒	☒	☒	☒
S1JA	✓	✓	✓	✓
S3A1	☒	☒	☒	☒
S3A3	☒	☒	☒	☒
S3A6	☒	☒	☒	☒
S3A7	☒	☒	☒	☒
S5D5	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A

14.5 A/D コンバータ (ADC)

ADC HAL モジュールでは、r_adc 上でアナログ/デジタル変換用 API を実装します。このモジュールは、MCU 上で利用可能な関連周辺機能用に ADC12、ADC14、および ADC16 をサポートします。ユーザー定義のコールバックを使用して、新しいサンプルが完了するたびにデータを処理することができます。

- 16 ビット ADC (S1JA) 、14 ビット ADC (S3A7、S3A3、S3A6、S3A1、S128、S124) 、12 ビット ADC (S7G2、S5D9、S5D5)
- 動作モード
 - シングルスキャンモード

- グループスキャンモード
- 連続スキャンモード
- 複数のチャンネルを搭載
 - アナログチャンネル数
 - S7G2 : 13 チャンネル (ユニット 0) および 12 チャンネル (ユニット 1)
 - S1JA : 17 チャンネル
 - S124 : 18 チャンネル
 - S3A7 : 28 チャンネル
 - 温度センサ向けチャンネル
 - 電圧センサ向けチャンネル

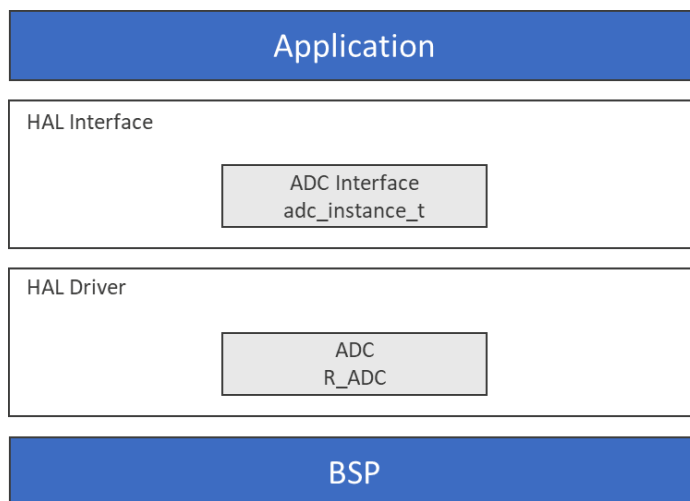


図 14.4 ADC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の ADC の機能は以下のとおりです。

[記号の説明]

- ✓ : 使用可能 (テスト済み)
- ☒ : 使用不可 (未テスト/機能せず、またはその両方)
- N/A : MCU のサポート対象外

グループ名	すべてのアナログチャンネル (ユニット 0、ユニット 1) のサポート	8 ビット	10 ビット	12 ビット	14 ビット	16 ビット	シングルスキャンモード
S124	✓	N/A	N/A	✓	✓	N/A	✓
S128	✓	N/A	N/A	✓	✓	N/A	✓
S1JA	✓	N/A	N/A	N/A	N/A	✓	✓
S3A1	✓	N/A	N/A	✓	✓	N/A	✓
S3A3	✓	N/A	N/A	✓	✓	N/A	✓
S3A6	✓	N/A	N/A	✓	✓	N/A	✓

グループ名	すべてのアナログチャンネル (ユニット 0、ユニット 1) のサポート	8 ビット	10 ビット	12 ビット	14 ビット	16 ビット	シングルスキャンモード
S3A7	✓	N/A	N/A	✓	✓	N/A	✓
S5D5	✓	✓	✓	✓	N/A	N/A	✓
S5D9	✓	✓	✓	✓	N/A	N/A	✓
S7G2	✓	✓	✓	✓	N/A	N/A	✓

グループ名	連続スキャンモード	グループスキャンモード	プログラマブルゲインアンプ	ELC HAL ドライバーによるイベントリンク ^{注1}
S124	✓	✓	☒	☒
S128	✓	✓	☒	☒
S1JA	✓	✓	☒	☒
S3A1	✓	✓	☒	☒
S3A3	✓	✓	☒	☒
S3A6	✓	✓	☒	☒
S3A7	✓	✓	☒	☒
S5D5	✓	✓	☒	☒
S5D9	✓	✓	☒	☒
S7G2	✓	✓	☒	☒

注1 ELCはグループモードでのみサポートされます。ELCで設定してください。

14.6 非同期汎用タイマ (AGT)

AGT HAL モジュールは、タイミングアプリケーションのためのハイレベルの API を提供し、`r_agt` に実装されています。AGT HAL モジュールは、MCU 上の AGT 周辺機能を使用します。ユーザー定義のコールバックを作成し、タイマイベントに応答できます。

AGT HAL モジュールは、ユーザーが指定した時間にタイマを構成します。この期間が経過すると、以下のいずれかのイベントが発生します。

- CPU に割り込み、ユーザーコールバック関数が指定されていればそれを呼び出す
- ポート端子をトグル
- 転送インタフェースで設定されている場合、DMAC/DTC を使用してデータ転送
- ELC インタフェースで設定されている場合、別の周辺機能を起動
- 複数チャンネル：16 ビットのチャンネルが 2 つ
 - チャンネル 1 はチャンネル 0 のアンダーフローによってクロック供給することができ、カスケードされた 32 ビットタイマが作成されます。
- コアクロック: PCLKB、LOCO、または Fsub を使用してクロックを供給できます。LOCO または Fsub でクロックが供給された場合、それを使用して MCU をスリープモードから起動できます。

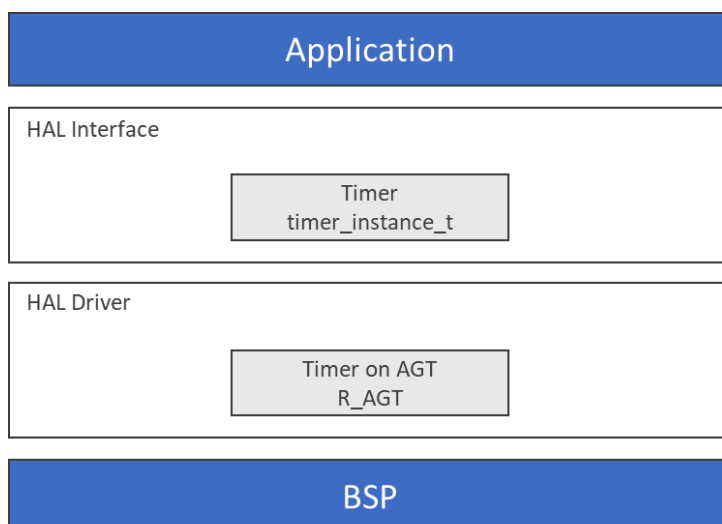


図 14.5 AGT HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の AGT の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	タイマモード	パルス出力モード	イベントカウンタモード	パルス幅測定モード
S124	✓	✓	☒	☒
S128	✓	✓	☒	☒
S1JA	✓	✓	☒	☒
S3A1	✓	✓	☒	☒
S3A3	✓	✓	☒	☒
S3A6	✓	✓	☒	☒
S3A7	✓	✓	☒	☒
S5D5	✓	✓	☒	☒
S5D9	✓	✓	☒	☒
S7G2	✓	✓	☒	☒

グループ名	パルス期間測定モード	ELC HAL ドライバーによるイベントリンク	コンペアマッチ機能 注1
S124	☒	☒	✓
S128	☒	☒	✓
S1JA	☒	☒	✓
S3A1	☒	☒	✓
S3A3	☒	☒	✓
S3A6	☒	☒	✓
S3A7	☒	☒	✓
S5D5	☒	☒	✓
S5D9	☒	☒	✓
S7G2	☒	☒	✓

注1 コンパレータ A/B 出力端子を設定することでコンペアマッチ機能を使用できます。

14.7 クロック周波数精度測定回路 (CAC)

CAC HAL モジュールは、クロック精度制御アプリケーションのためのハイレベルの API を提供し、`r_cac` に実装されています。このモジュールは、MCU 上で信頼性を重視するアプリケーションのフェイルセーフメカニズムを実装する上で役立つクロック周波数精度測定回路 (CAC) 周辺機能を使用します。ユーザー定義のコールバックを作成し、さまざまなエラー表示に応答できます。

CAC HAL モジュールの API には、参照信号入力に基づいてクロック周波数を監視できるクロック周波数測定回路との間にインタフェースがあります。参照信号は、外部から供給されるクロックソース、またはいずれかの内部クロックソースの可能性があり、割り込み要求は、完了した測定、検出された周波数エラー、またはカウンタオーバーフローによってオプションで生成されることがあります。外部から供給される参照クロックではデジタルフィルタが利用でき、除算値は内部提供の測定と参照クロックの両方で利用できます。参照クロックのエッジ検出オプションは、上昇、下降、またはその両方で構成できます。

次のクロックの周波数を測定できます。

- メインクロックオシレータのクロック出力 (メインクロック)
- サブクロックオシレータのクロック出力 (サブクロック)
- 高速オンチップオシレータのクロック出力 (HOCO クロック)
- 中速オンチップオシレータのクロック出力 (MOCO クロック)
- 低速オンチップオシレータのクロック出力 (LOCO クロック)

- IWDT 専用オンチップオシレータのクロック出力 (IWDTCLK クロック)
- 周辺モジュールクロック (PCLKB)

測定クロックは参照クロックを使って監視されます。参照クロックは、CACREF 入力端子で提供される外部クロックか、次のいずれかの内部クロックにすることができます。

- メインクロックオシレータのクロック出力 (メインクロック)
- サブクロックオシレータのクロック出力 (サブクロック)
- 高速オンチップオシレータのクロック出力 (HOCO クロック)
- 中速オンチップオシレータのクロック出力 (MOCO クロック)
- 低速オンチップオシレータのクロック出力 (LOCO クロック)
- IWDT 専用オンチップオシレータのクロック出力 (IWDTCLK クロック)
- 周辺モジュールクロック (PCLKB)

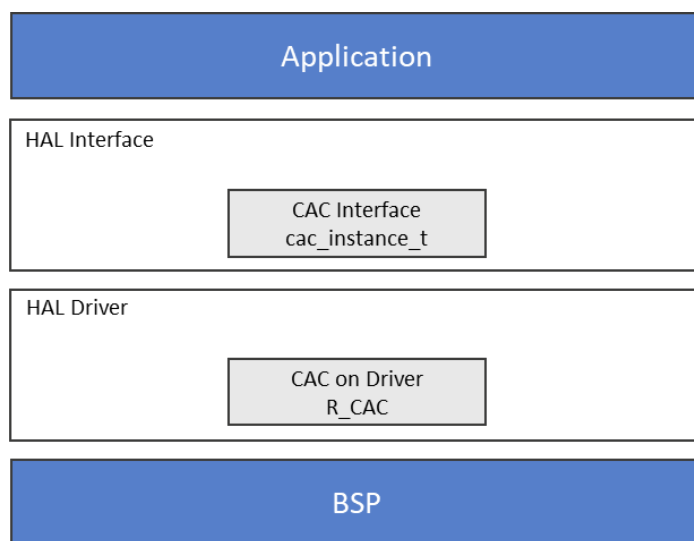


図 14.6 CAC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の CAC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	測定クロック ターゲット： メインクロック	測定クロック ターゲット： サブクロック	測定クロック ターゲット： HOCO	測定クロック ターゲット： MOCO	測定クロック ターゲット： LOCO
S124	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

グループ名	測定クロック ターゲット： IWDTCLK	測定クロック ターゲット： 周辺モジュールクロック B	測定参照クロック ターゲット： CACREF ピンへの外 部クロック入力	測定参照クロック ターゲット： メインクロック	測定参照クロック ターゲット： サブクロック
S124	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

グループ名	測定参照クロック ターゲット： HOCO	測定参照クロック ターゲット： MOCO	測定参照クロック ターゲット： LOCO	測定参照クロック ターゲット： IWDTCLK	測定参照クロック ターゲット： 周辺モジュールク ロック B
S124	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

グループ名	選択可能な機能： デジタルフィルタ	割り込み要因： 測定終了	割り込み要因： 周波数エラー	割り込み要因： オーバーフロー	消費電力削減のため のモジュールストップ 機能
S124	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

14.8 コントローラエリアネットワーク (CAN)

CAN HAL モジュールは、CAN ネットワーク用のハイレベルの API を提供します。CAN HAL モジュールは、MCU で利用可能な CAN 周辺機能をサポートします。ユーザーコールバック関数は、送信、受信、またはエラーの割り込みを受け取ったときにドライバーによって起動されるよう定義する必要があります。このコールバックは、チャンネル、メールボックス、イベントを示すパラメータとともに返されます。

- 標準 (11 ビット) と拡張 (29 ビット) の両方のメッセージ形式
- ビットタイミングの構成 (CAN の仕様での定義に従う)
- 標準または拡張の ID フレームを使用した最大 32 個の送信または受信メールボックス
- データまたはリモートの CAN フレームをキャプチャするように、受信メールボックスを構成可能
- 送信、受信、またはエラーの割り込みを受け取ったときのユーザーコールバック関数

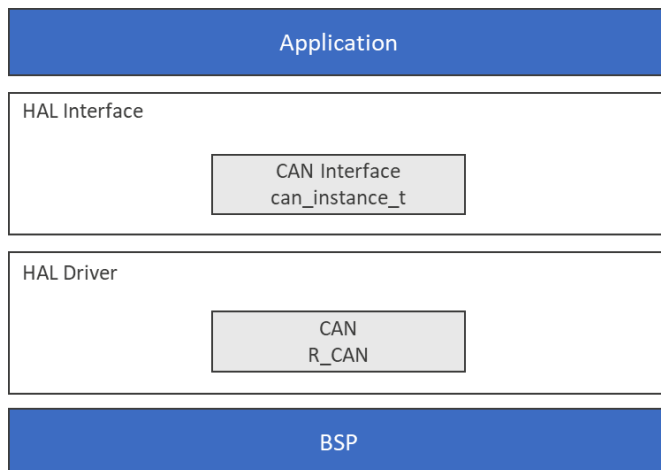


図 14.7 CAN HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の CAC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	プログラム可能なビットレート	最大 32 個のメールボックスのサポート	メールボックスのノーマルモード	メールボックスの FIFO モード	データフレーム受信のサポート	リモートフレーム受信のサポート
S124	✓	✓	✓	☒	✓	✓
S128	✓	✓	✓	☒	✓	✓
S1JA	✓	✓	✓	☒	✓	✓
S3A1	✓	✓	✓	☒	✓	✓
S3A3	✓	✓	✓	☒	✓	✓
S3A6	✓	✓	✓	☒	✓	✓
S3A7	✓	✓	✓	☒	✓	✓
S5D5	✓	✓	✓	☒	✓	✓
S5D9	✓	✓	✓	☒	✓	✓
S7G2	✓	✓	✓	☒	✓	✓

グループ名	プログラム可能なワンショット受信	上書きモードの受信	オーバーランモードの受信	8 つのすべてのアクセプタンスマスクのサポート	メールボックスごとに独立したマスクの有効化または無効化のサポート
S124	☒	✓	✓	✓	✓

グループ名	プログラム可能なワンショット受信	上書きモードの受信	オーバーランモードの受信	8つのすべてのアクセプタンスマスクのサポート	メールボックスごとに独立したマスクの有効化または無効化のサポート
S128	☑	✓	✓	✓	✓
S1JA	☑	✓	✓	✓	✓
S3A1	☑	✓	✓	✓	✓
S3A3	☑	✓	✓	✓	✓
S3A6	☑	✓	✓	✓	✓
S3A7	☑	✓	✓	✓	✓
S5D5	☑	✓	✓	✓	✓
S5D9	☑	✓	✓	✓	✓
S7G2	☑	✓	✓	✓	✓

グループ名	送信要求の中止のサポート	バスオフのためのモード遷移：ISO11898-1仕様に準拠	バスオフのためのモード遷移：バスオフ開始時のCANホルトモードの自動起動	バスオフのためのモード遷移：バスオフ終了時のCANホルトモードの自動起動	バスオフのためのモード遷移：ソフトウェアによるCANホルトモードの起動	バスオフのためのモード遷移：ソフトウェアによるエラーアクティブ状態への遷移
S124	☑	✓	☑	☑	☑	☑
S128	☑	✓	☑	☑	☑	☑
S1JA	☑	✓	☑	☑	☑	☑
S3A1	☑	✓	☑	☑	☑	☑
S3A3	☑	✓	☑	☑	☑	☑
S3A6	☑	✓	☑	☑	☑	☑
S3A7	☑	✓	☑	☑	☑	☑
S5D5	☑	✓	☑	☑	☑	☑
S5D9	☑	✓	☑	☑	☑	☑
S7G2	☑	✓	☑	☑	☑	☑

グループ名	CANバスエラーの監視（スタックエラー、フォームエラー、ACKエラー、15ビットCRCエラー、ビットエラー、ACKデリミタエラー）	エラー状態へのすべての遷移の検出（エラー警告、エラーパッシブ、バスオフ開始、およびバスオフ復帰）	1、2、4、および8ビット期間から選択できる参照クロックのサポート	5つのすべての割り込み要因のサポート（受信完了、送信完了、受信FIFO、送信FIFOエラー割り込み）	CANスリープモード1のサポート（CANクロック停止）	3つのすべてのソフトウェアサポートユニットのサポート（アクセプタンスフィルタサポート、メールボックス検索サポート（受信メールボックス検索、送信メールボックス検索、メッセージロスト検索を含む）、およびチャネル検索サポート
S124	✓	✓	☑	✓	☑	☑
S128	✓	✓	☑	✓	☑	☑
S1JA	✓	✓	☑	✓	☑	☑
S3A1	✓	✓	☑	✓	☑	☑
S3A3	✓	✓	☑	✓	☑	☑
S3A6	✓	✓	☑	✓	☑	☑
S3A7	✓	✓	☑	✓	☑	☑
S5D5	✓	✓	☑	✓	☑	☑
S5D9	✓	✓	☑	✓	☑	☑
S7G2	✓	✓	☑	✓	☑	☑

グループ名	CAN クロック ソース： PCLKB	CAN クロック ソース： CANMCLK	3つのすべてのテ ストモードのサ ポート（リスンオ ンリー、セルフテ スト1（外部ルー プバック）、セル フテスト2（内部 ループバック））	モジュールス トップ機能	標準 CAN（11 ビット）サポート	拡張 CAN （J1939、39 ビット）
S124	N/A	✓	✓	✓	✓	✓
S128	N/A	✓	✓	✓	✓	✓
S1JA	N/A	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

14.9 クロック発生回路 (CGC)

CGC HAL モジュールは、クロック制御アプリケーションのためのハイレベルの API を提供し、`r_cgc` に実装されています。CGC HAL モジュールは、クロック制御周辺機能を使用して MCU のクロック制御機能を構成し、制御します。すべてのプロジェクトでクロック機能が必要となるため、CGC HAL モジュールはデフォルトでプロジェクトに追加されます（モジュールは、統合ソリューション開発環境内で構成されます）。ユーザー定義のコールバックを作成し、メイン発振器が停止したときに信号を送信できます。

- 任意のクロック（HOCO、MOCO、LOCO、メインクロック、PLL、サブオシレータ）をシステムクロックソースとして設定
- 内部クロックを設定（ICLK、PCLK など）
- 出力クロックを構成
- 発振停止検出機能を設定
- 最大6つの各クロックドメインにクロック除数を設定
- 一部の MCU は制御可能な外部クロック出力もサポートし、独立した除数を持つことが可能

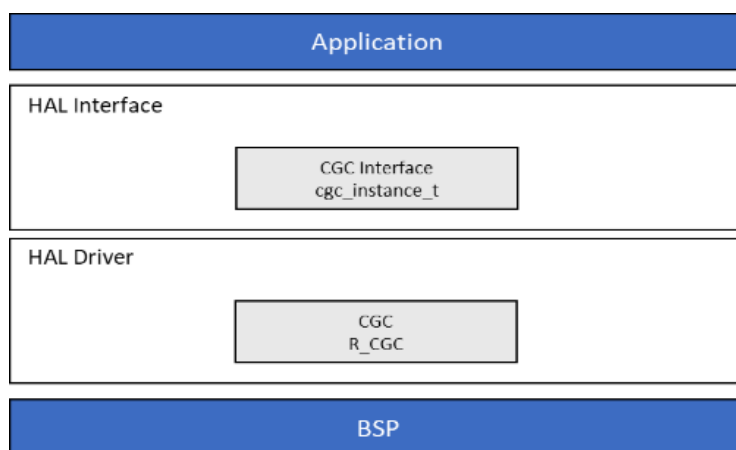


図 14.8 CGC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の CGC の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	MOSC	SOSC	PLL 回路	HOCO	MOCO	LOCO
S124	✓	✓	N/A	✓	✓	✓
S128	✓	✓	N/A	✓	✓	✓
S1JA	✓	✓	N/A	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	IWDTLOCO	JTAG 外部ク ロック入力	SWD 外部クロッ ク入力
S124	✓	N/A	☒
S128	✓	N/A	☒
S1JA	✓	N/A	☒
S3A1	✓	✓	✓
S3A3	✓	✓	✓
S3A6	✓	✓	✓
S3A7	✓	✓	✓
S5D5	✓	✓	✓
S5D9	✓	✓	✓
S7G2	✓	✓	✓

14.10 CRC 演算器 (CRC)

CRC HAL モジュールは、メモリ内のデータブロック上またはシリアル通信インタフェース (SCI) チャンネルを介したデータストリーム上の 8、16、および 32 ビットの CRC 値を、さまざまな種類の業界標準の多項式を利用して計算するためのハイレベルの API を提供します。CRC HAL モジュールは `r_crc` に実装され、MCU 上の CRC 周辺機能を使用します。

- CRC HAL モジュールは、メモリ内のデータブロックに対して CRC 計算を実行可能
- CRC HAL モジュールは、シリアル通信インタフェース (SCI) チャンネル上で送受信されるデータストリーム上の CRC を計算可能 (スヌープモード)
- CRC HAL モジュールは、以下の 8 および 16 ビット CRC 多項式をサポート。これらの多項式は並行して 8 ビットデータに使用可能
 - $X^8 + X^2 + X + 1$ (CRC-8)
 - $X^{16} + X^{15} + X^2 + 1$ (CRC-16)
 - $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT)
- CRC HAL モジュールは、以下の 32 ビット CRC 多項式をサポート。これらの多項式は並行して 32 ビットデータに使用可能
 - $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (CRC-32)
 - $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$ (CRC-32C)
- CRC HAL モジュールは、LSB ファーストまたは MSB ファーストのビット順で CRC を計算可能

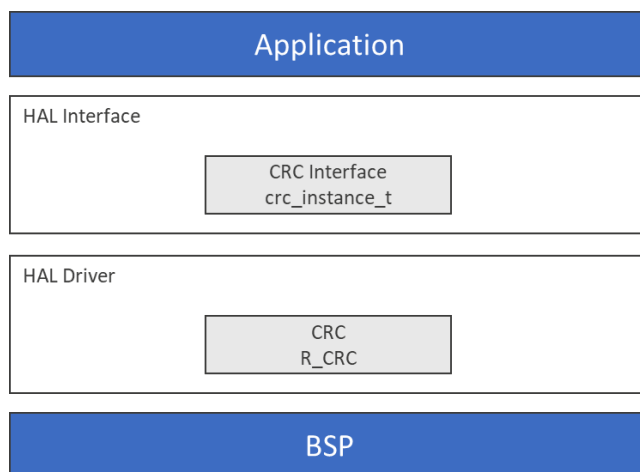


図 14.9 CRC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の CRC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	データサイズ 8ビット	データサイズ 32ビット	CRC 計算切り替え	モジュールストップ	CRC Snoop
S124	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

14.11 静電容量式タッチセンシングユニット (CTSU)

CTSU ドライバーは MCU の CTSU をサポートします。CTSU ドライバーには、引数として渡される設定に応じて CTSU をオープン、クローズ、実行、および制御するための機能が用意されています。CTSU HAL ドライバーは、CTSU 周辺デバイスを初期化して任意の設定済み (かつ有効化済み) チャンルのキャパシタンスの変化を検出し、必要なフィルタリングを実行し、ボタン、ホイール、スライダーなど上位のフレームワークレイヤーで使用できるさまざまなデータを生成するために使用されます。こうしたレイヤーで要求されるさまざまな種類のデータをサポートするため、この実装では上位レベルのレイヤーで多様な種類の処理済みデータを必要性に基づいて読み取れるための関数が提供されています。ドライバーは、設定されたチャンネルのスキャンを行い、DTC を使ってデータの移動を行い、フィルタリング、ドリフト補償、自動チューニングを行い、各反復が完了して新しい処理データが使用可能になると、コールバックによってユーザーに通知します。このコールバックを使用して上位レイヤーでデータを読み取ることができます。

このモジュールは CTW for Synergy ツールと合わせて使用されるように設計されています。CTW for Synergy ツールは、初期化と処理に必要な構造を生成します。このドライバーにより、ユーザーは独自のフィルタリングや自動調整アルゴリズムを設定し、それをプロセスに統合することも可能になります。ドライバーでは一度に 1 つの構成しかサポートされませんが、アプリケーションの必要に応じて複数のチャンネル

設定でドライバーを再開することができます。CTSU ドライバーを使用すると、ユーザーは相互容量モードや自己容量モードなどのサポートされるすべての動作モード用に CTSU チャンネルを設定できます。

CTSU ドライバーは SSP の CTSU インタフェースを実装しています。

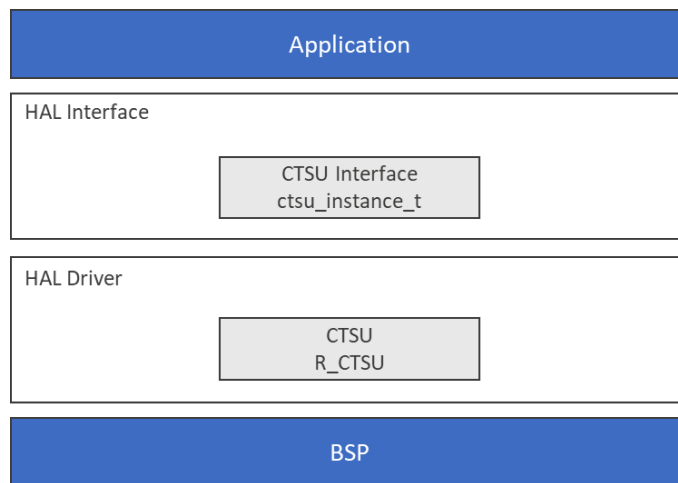


図 14.10 CTSU HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の CRC の機能は以下のとおりです。

【記号の説明】

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	自己容量 シングルスキャン モード	自己容量 マルチスキャン モード	相互容量 フルスキャン モード	センサー安定待ち 時間と測定時間	CTSU 割り込み	ELC リンク
S124	✓	✓	✓	✓	✓	N/A
S128	✓	✓	✓	✓	✓	N/A
S1JA	☒	☒	☒	☒	☒	N/A
S3A1	☒	☒	☒	☒	☒	N/A
S3A3	✓	✓	✓	✓	✓	N/A
S3A6	☒	☒	☒	☒	☒	N/A
S3A7	✓	✓	✓	✓	✓	N/A
S5D5	✓	✓	✓	✓	✓	N/A
S5D9	✓	✓	✓	✓	✓	N/A
S7G2	✓	✓	✓	✓	✓	N/A

14.12 12 ビット D/A コンバータ (DAC12)

DAC HAL モジュールは、r_dac に実装されているデジタル/アナログ変換アプリケーションのためのハイレベルの API を提供します。DAC HAL モジュールは、MCU のデュアルチャンネル 12 ビット D/A コンバータ (DAC12) 周辺機能をサポートします。

このモジュールは、デュアルチャンネル 12 ビット D/A コンバータ (DAC12) を、正および負の基準電圧の間の 4096 段階の電圧レベルのいずれかを出力するように構成します。この DAC HAL モジュールには以下の構成設定が含まれます。

- 16 ビット入力データレジスタ用に左詰めまたは右詰め 12 ビット値形式を設定
- 出力の増幅器を有効化または無効化
- ADC モジュールを使用して同期干渉防止モードで動作

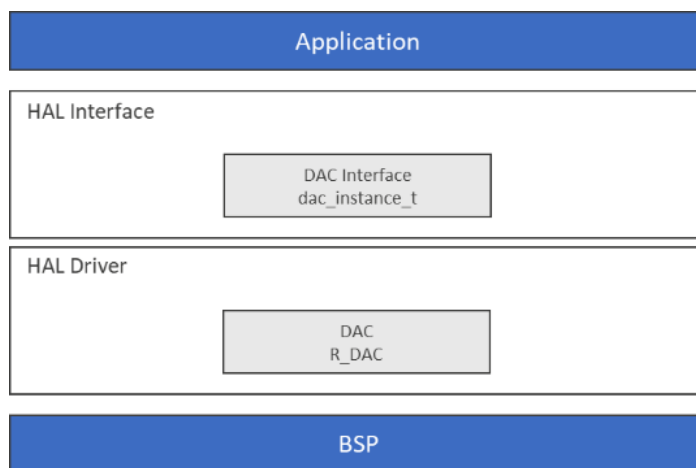


図 14.11 DAC HAL (DAC12) モジュールのブロック図

SSP でサポート対象およびサポート対象外の DAC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	12 ビット	2 チャンネル出力	モジュールストップ	ELC HAL ドライバーによるイベントリンク ^{注1}
S124	✓	N/A	✓	☒
S128	N/A	N/A	N/A	☒
S1JA	N/A	N/A	N/A	N/A
S3A1	✓	N/A	✓	☒
S3A3	✓	N/A	✓	☒
S3A6	✓	N/A	✓	☒
S3A7	✓	✓	✓	☒
S5D5	✓	✓	✓	☒
S5D9	✓	✓	✓	☒
S7G2	✓	✓	✓	☒

注1 ELC イベントは DAC start() インタフェースコールの代わりに使用できます。これは ELC API を使用して設定するのではなく、ユーザーがリンクを設定することで使用できるようになります。

14.13 8 ビット D/A コンバータ (DAC8)

DAC8 HAL モジュールは、r_dac8 に実装されているデジタル/アナログ変換アプリケーションのためのハイレベルの API を提供します。DAC 8 HAL モジュールは、該当する MCU の 8 ビット D/A コンバータ (DAC 8) 周辺機能をサポートします。

- S1JA、S128、S3A3、および S3A6 MCU で利用可能
- S1JA、S3A3、S3A6 用に 2 チャンネル、S128 用に 3 チャンネルを備えた 8 ビット D/A コンバータ
- 左詰めまたは右詰めの入力データフォーマット
- アナログ/デジタルコンバータ (ADC) モジュールとの同期
- 動作モード
 - ノーマル
 - リアルタイム (イベントリンク)
- チャージポンプコントロール

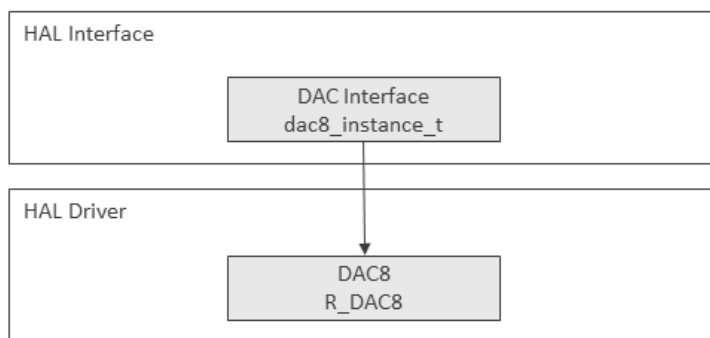


図 14.12 DAC8 HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の DAC8 の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	8 ビット	2 チャンネル出力	3 チャンネル出力	モジュールストップ	ELC HAL ドライバーによるイベントリンク ^{注1}
S124	N/A	N/A	N/A	N/A	N/A
S128	✓	✓	✓	✓	✓
S1JA	✓	✓	N/A	✓	✓
S3A1	N/A	N/A	N/A	☒	☒
S3A3	✓	✓	N/A	✓	✓
S3A6	☒	☒	N/A	☒	☒
S3A7	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A	N/A

注 1 ELC イベントは DAC start() インタフェースコールの代わりに使用できます。これは ELC API を使用して設定するのではなく、ユーザーがリンクを設定することで使用できるようになります。

14.14 ダイレクトメモリアクセスコントローラ (DMAC)

DMAC HAL モジュールは、データ転送アプリケーションのためのハイレベルの API を提供し、r_dmac に実装されています。DMAC HAL モジュールは、割り込みまたはイベントの発生時に、ユーザー指定の移動元からユーザー指定の移動先にデータを移動します。DMAC HAL モジュールは、MCU 上の DMAC 周辺機能を使用します。ユーザー定義のコールバックを作成し、転送イベント発生時に CPU に通知できます。

- MCU に搭載されている DMAC モジュール
- 割り込み (要求に応じた)
- 転送モード
 - ノーマル転送モード
 - リピート転送モード
 - ブロック転送モード
 - アドレスモード
- 拡張リピート領域機能
- 複数チャンネル (数は使用されている MCU に依存)

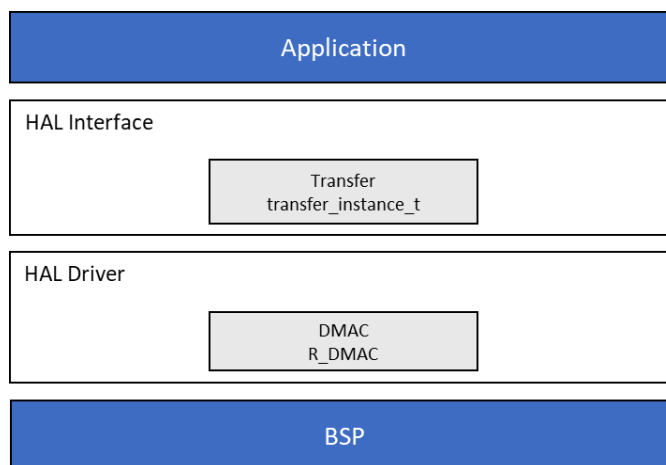


図 14.13 DMAC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の DMAC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	ノーマル転送モード	リピート転送モード	ブロック転送モード	拡張リピート領域機能	ELC HAL ドライバーによるイベントリンク
S124	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	✓	✓	☒
S3A3	✓	✓	✓	✓	☒
S3A6	✓	✓	✓	✓	☒
S3A7	✓	✓	✓	✓	☒
S5D5	✓	✓	✓	✓	☒
S5D9	✓	✓	✓	✓	☒
S7G2	✓	✓	✓	✓	☒

14.15 データ演算回路 (DOC)

データ演算回路 (DOC) HAL モジュールは、DOC アプリケーションのためのハイレベルの API を提供し、`r_doc` に実装されています。DOC HAL モジュールは、MCU 上の DOC 周辺機能を使用します。ユーザー定義のコールバックを作成し、イベント発生時に CPU に通知できます。

DOC HAL モジュールの周辺機能は 16 ビットデータを比較するために使用され、以下のイベントを検出できます。

- データ値の不一致または一致
- 加算演算のオーバーフロー
- 減算演算のアンダーフロー

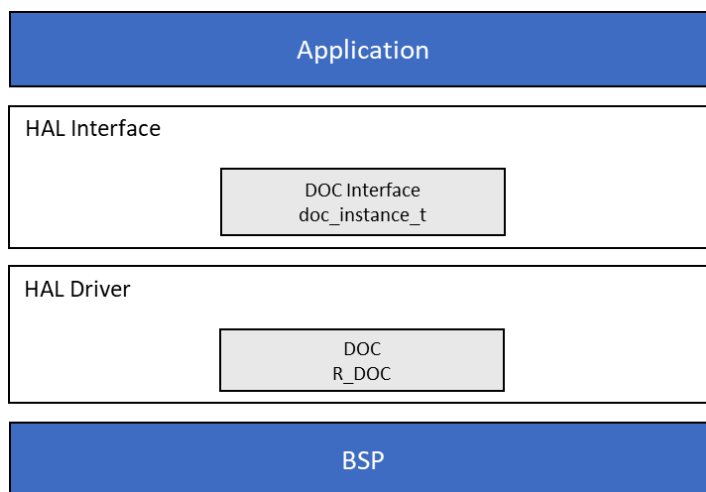


図 14.14 DOC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の DOC の機能は以下のとおりです。

【記号の説明】

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	モジュールストップ機能	ELC HAL ドライバーによるイベントリンク
S124	✓	☒
S128	✓	☒
S1JA	✓	☒
S3A1	✓	☒
S3A3	✓	☒
S3A6	✓	☒
S3A7	✓	☒
S5D5	✓	☒
S5D9	✓	☒
S7G2	✓	☒

14.16 データトランスファコントローラ (DTC)

データトランスファコントローラ (DTC) HAL モジュールは、データ転送アプリケーションのためのハイレベルの API を提供し、r_dtc に実装されています。DTC HAL モジュールは、割り込みまたはイベントの発生時に、ユーザー指定の移動元からユーザー指定の移動先にデータを移動します。このモジュールは、MCU 上の DTC 周辺機能を使用します。ユーザー定義のコールバックを作成し、転送イベント発生時に CPU に通知できます。

- 割り込みまたはイベントの発生時に、ユーザー指定の移動元からユーザー指定の移動先にデータを移動
- MCU に搭載されている DTC モジュール
- 割り込み（要求に応じた）
- 転送モード
 - ノーマル転送モード
 - リピート転送モード
 - ブロック転送モード
 - アドレスモード
 - チェーン転送モード

- 複数チャンネル（数は使用されている MCU に依存）
チャンネル数はRAM ベースのベクタテーブルのサイズによってのみ制限される

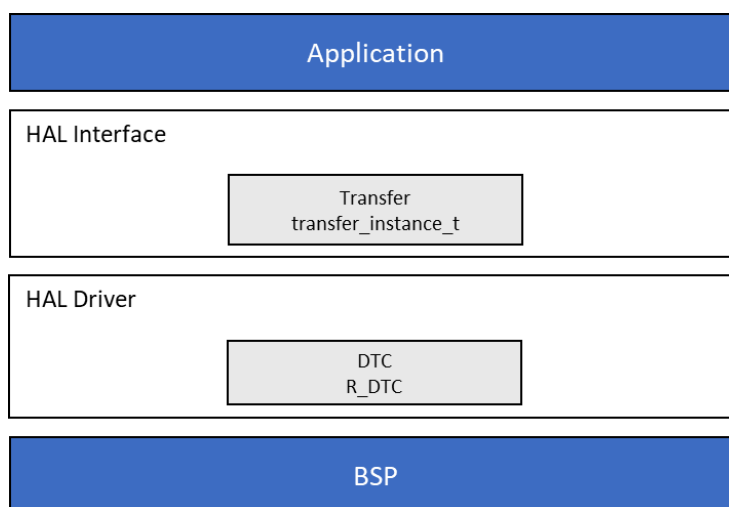


図 14.15 DTC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の DTC の機能は以下のとおりです。

【記号の説明】

- ✓：使用可能（テスト済み）
- ☒：使用不可（未テスト/機能せず、またはその両方）
- N/A：MCU のサポート対象外

グループ名	ノーマル転送モード	リピート転送モード	ブロック転送モード	データ転送単位 (8ビット、16ビット、32ビット)	DTC 転送完了時の イベントリンク機能 用の API 構成
S124	✓	✓	✓	✓	☒
S128	✓	✓	✓	✓	☒
S1JA	✓	✓	✓	✓	☒
S3A1	✓	✓	✓	✓	☒
S3A3	✓	✓	✓	✓	☒
S3A6	✓	✓	✓	✓	☒
S3A7	✓	✓	✓	✓	☒
S5D5	✓	✓	✓	✓	☒
S5D9	✓	✓	✓	✓	☒
S7G2	✓	✓	✓	✓	☒

14.17 イベントリンクコントローラ (ELC)

イベントリンクコントローラ (ELC) HAL モジュールは、ELC HAL アプリケーションのためのハイレベルの API を提供し、`r_elc` に実装されています。ELC HAL モジュールは、MCU 上の ELC 周辺機能を使用します。ELC HAL モジュールに関連付けられたコールバックはありません。e² studio 統合ソリューション開発環境のプロジェクトコンフィギュレータには、ELC HAL モジュールがデフォルトですべてのプロジェクトに含まれています。ELC HAL モジュールを構成するには、[スレッド]タブの[HAL/共通モジュール]で ELC HAL モジュールを選択し、次に[HAL/共通スタック]ウィンドウで選択します。

- 2つのブロック間にイベントリンクを作成
- その2つのブロック間のイベントリンクを切断
- CPU に割り込む2つのソフトウェアイベントのうち1つを生成

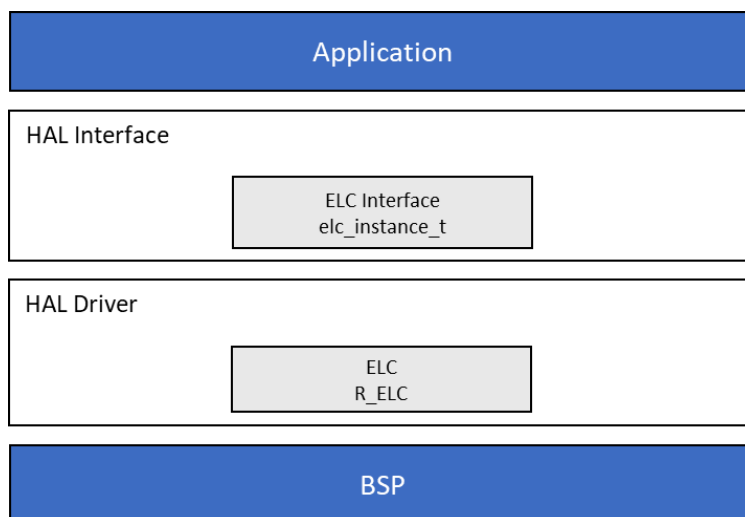


図 14.16 ELC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の ELC の機能は以下のとおりです。

【記号の説明】

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	2つのブロック間にイベントリンクを作成	2つのブロック間のイベントリンクを切断	CPUに割り込む2つのソフトウェアイベントのうち1つを生成
S124	✓	✓	✓
S128	✓	✓	✓
S1JA	✓	✓	✓
S3A1	✓	✓	✓
S3A3	✓	✓	✓
S3A6	✓	✓	✓
S3A7	✓	✓	✓
S5D5	✓	✓	✓
S5D9	✓	✓	✓
S7G2	✓	✓	✓

14.18 フラッシュメモリ

フラッシュモジュールには、`r_flash_lp` と `r_flash_hp` の 2 つがあります。高性能フラッシュモジュール (Flash_HP) は、MCU の S7 ファミリーおよび S5 ファミリーのプログラミングに使用される API です。ローパワーフラッシュモジュール (Flash_LP) は、MCU の S3 ファミリーおよび S1 ファミリーのプログラミングに使用される API です。この 2 つを相互に交換することはできませんが、これらのモジュールの API および他の機能はよく似ています。

フラッシュ HAL モジュールの API により、アプリケーションは MCU 内に存在するデータと ROM 両方のフラッシュエリアのリード、ライト、消去を行うことができます。使用可能なフラッシュメモリの量は MCU パーツごとに異なりますが、API 関数はすべてのデバイスに適用されます。

- コードフラッシュ (ROM) のブロック消去、読み取り、書き込み、およびブランクチェック
- データフラッシュのブロック化および非ブロック化消去、読み取り、書き込み、およびブランクチェック

- コードフラッシュのブロック化消去、読み取り、書き込み、およびブランクチェック
- 非ブロック化データフラッシュ操作の完了時に使用されるコールバック関数
- コードフラッシュの指定されたエリアだけを消去または書き込み可能にする、ROM フラッシュ用のアクセスウィンドウ（書き込み保護）
- スタートアッププログラムを最初に消去することなく安全にリライトできるようにする、ブートブロックスワッピング

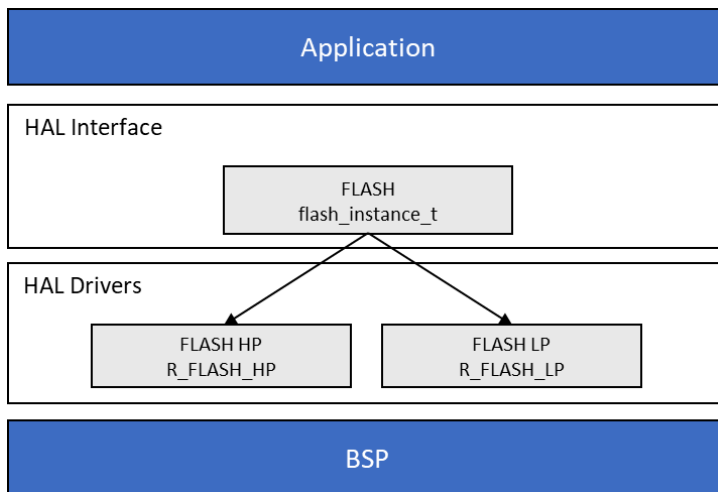


図 14.17 フラッシュメモリ HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の Flash_HP の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	ユーザープログラムによるフラッシュメモリのプログラミング（セルフプログラミング）	バックグラウンドオペレーション（BGO） ^注
S124	N/A	N/A
S128	N/A	N/A
S1JA	N/A	N/A
S3A1	N/A	N/A
S3A3	N/A	N/A
S3A6	N/A	N/A
S3A7	N/A	N/A
S5D5	✓	✓
S5D9	✓	✓
S7G2	✓	✓

注：データフラッシュのみサポート。

SSP でサポート対象およびサポート対象外の Flash_HP の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	ユーザープログラムによるフラッシュメモリのプログラミング (セルフプログラミング)	バックグラウンドオペレーション (BGO) 注
S124	✓	✓
S128	✓	✓
S1JA	✓	✓
S3A1	✓	✓
S3A3	✓	✓
S3A6	✓	✓
S3A7	✓	✓
S5D5	N/A	N/A
S5D9	N/A	N/A
S7G2	N/A	N/A

注 : データフラッシュのみサポート。

14.19 ファクトリーMCU 情報 (FMI)

FMI HAL モジュールは、ファクトリーMCU 情報 (FMI) フラッシュテーブルからレコードを読み込むアプリケーション用のハイレベルの API を提供します。FMI HAL モジュールは r_fmi に実装されます。FMI HAL モジュールは、MCU 上のフラッシュインタフェースを使用します。

FMI HAL モジュールは、MCU の FMIFRT (ファクトリーMCU 情報フラッシュルートテーブル) を読み取って、フラッシュ内のテーブルの先頭アドレスを取得します。このモジュールは、テーブルの製品情報レコードに呼び出し側のポインタを設定します。この情報を使用して、この MCU パッケージ固有の機能を判定できます。FMI HAL モジュールからは次の情報を利用できます。

- 製品情報 : 製品名、パッケージ、ピン数、温度範囲
- 製品の機能: メジャーバージョン、マイナーバージョン、バリエーションデータ
- 割り込みやイベントなどのイベント情報

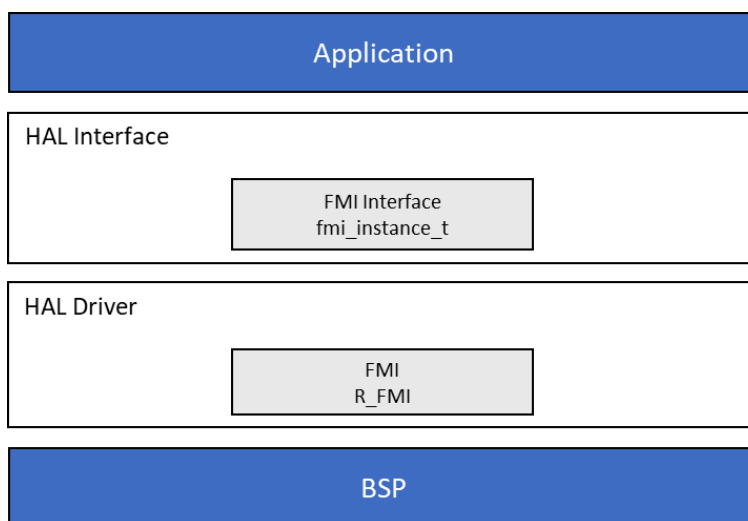


図 14.18 FMI HAL モジュールのブロック図

14.20 グラフィックス LCD コントローラ (GLCDC)

グラフィック LCD コントローラ (GLCDC) HAL モジュールは、GLCDC アプリケーションのためのハイレベルの API を提供し、`r_glcd` に実装されています。GLCDC HAL モジュールは、MCU 上のグラフィックス LCD ドライバ周辺機能を使用します。ユーザー定義のコールバックを作成し、フレームバッファの切り替えまたはアンダーフロー検出を処理することができます。

- RGB インタフェース (最大 24 ビット) と同期信号 (HSYNC、VSYNC、およびデータ有効 (オプション)) を備えた LCD パネル
- 入力グラフィックプレーン用の各種の色フォーマット (RGB888、ARGB8888、RGB565、ARGB1555、ARGB4444、CLUT8、CLUT4、CLUT1)
- 512 ワード (32 ビット/ワード) の入力グラフィックプレーン ARGB8888 のためのカラーlookupアップテーブル (CLUT) の使用
- 出力用の各種の色フォーマット (RGB888、RGB666、RGB565、シリアル RGB888)
- 背景プレーン上に 2 つのグラフィックプレーンを入力し、画面上でブレンド可能
- パネルにドットクロックを生成。クロックソースは内部または外部 (LCD_EXTCLK) から選択可能
- 明るさ調整、コントラスト調整、ガンマ補正
- フレームバッファの切り替えまたはアンダーフロー検出を処理するための GLCDC 割り込み

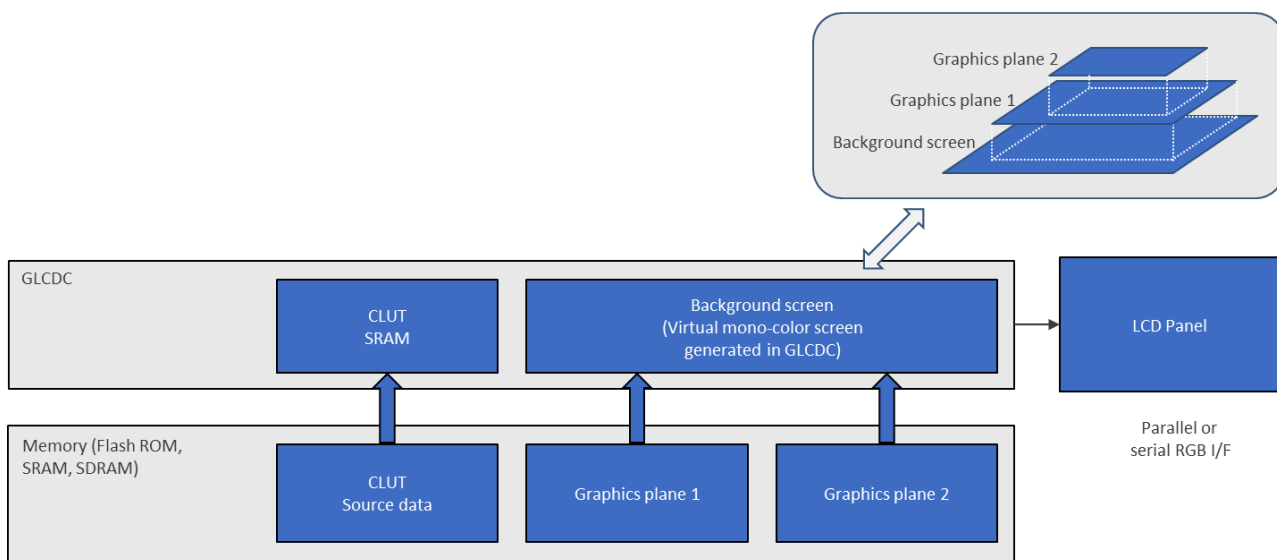


図 14.19 GLCDC のデータフロー

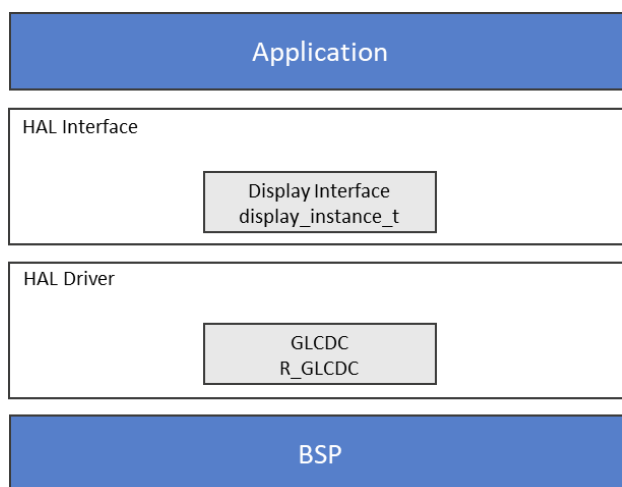


図 14.20 GLCDC モジュールのブロック図

SSP でサポート対象およびサポート対象外の GLCDC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	単色バックグラウンドプレーン	グラフィック 1 プレーン	グラフィック 2 プレーン	ピクセルあたり 16 ビットのグラフィックス	ピクセルあたり 32 ビットのグラフィックス	1 ビット LUT のサポート
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	4 ビット CLUT	8 ビット CLUT	すべてのピクセルフォーマットのサポート	アルファブレンディングのサポート	ビデオシングルタイムミングの調整	すべての出力データフォーマットのサポート
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	すべてのディザリングモードのサポート	同期信号出力	輝度とコントラストのサポート	ガンマ補正
S124	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A
S5D9	✓	✓	✓	✓
S7G2	✓	✓	✓	✓

14.21 汎用 PWM タイマ (GPT)

汎用 PWM タイマ (GPT) HAL モジュールは、タイマアプリケーションのためのハイレベルの API を提供し、r_gpt に実装されています。GPT HAL モジュールは、MCU 上の GPT 周辺機能を使用します。ユーザー定義のコールバックを作成し、タイマイベントに応答できます。

GPT HAL モジュールは、ユーザーが指定した時間にタイマを構成します。この期間が経過すると、以下のいずれかのイベントが発生します。

- ユーザーコールバック関数が指定されている場合はそれを呼び出す CPU 割り込み
- ポート端子をトグル
- 転送インタフェースで設定されている場合、DMAC/DTC を使用してデータ転送
- ELC インタフェースで設定されている場合、別のペリフェラルを起動

GPT の特長は以下のとおりです。

- コアクロックとしての PCLKD
 - チャンネルごとの 2 個の出力端子
 - アップカウントするのこぎり波/三角波
 - 2つのコンペアマッチレジスタと入力キャプチャレジスタ
 - 8つのイベントリンクコントローラ (ELC) へのレスポンスおよび ELC 内のイベントとして構成可能

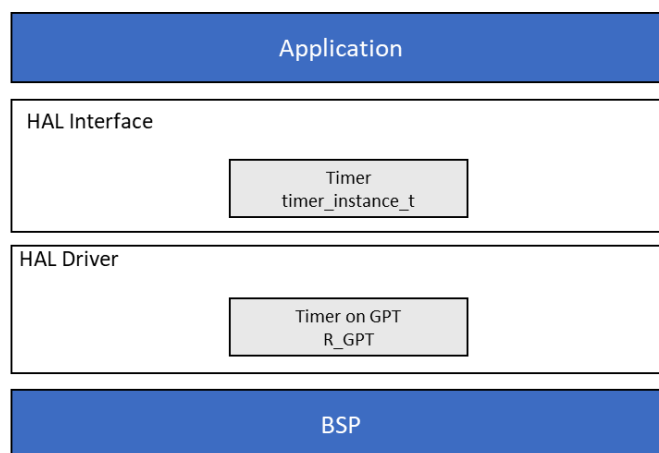


図 14.21 GPT HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の GPT の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	のこぎり波	三角波	ブラシレス直流モーターを制御するための PWM 波形	コンペアマッチ出力 (Low 出力、High 出力、トグル出力)	インプットキャプチャ機能	デッドタイム自動付加機能
S124	☒	✓	☒	✓	☒	☒
S128	☒	✓	☒	✓	☒	☒
S1JA	☒	✓	☒	✓	☒	☒
S3A1	☒	✓	☒	✓	☒	☒
S3A3	☒	✓	☒	✓	☒	☒
S3A6	☒	✓	☒	✓	☒	☒
S3A7	☒	✓	☒	✓	☒	☒

グループ名	のこぎり波	三角波	ブラシレス直流モーターを制御するための PWM 波形	コンペアマッチ出力 (Low 出力、High 出力、トグル出力)	インプットキャプチャ機能	デッドタイム自動付加機能
S5D5	☒	✓	☒	✓	☒	☒
S5D9	☒	✓	☒	✓	☒	☒
S7G2	☒	✓	☒	✓	☒	☒

グループ名	PWM モード	位相計数機能	ワンショット動作	ELC HAL ドライバーによるイベントリンク	ノイズフィルタ機能
S124	✓	☒	✓	☒	☒
S128	✓	☒	✓	☒	☒
S1JA	✓	☒	✓	☒	☒
S3A1	✓	☒	✓	☒	☒
S3A3	✓	☒	✓	☒	☒
S3A6	✓	☒	✓	☒	☒
S3A7	✓	☒	✓	☒	☒
S5D5	✓	☒	✓	☒	☒
S5D9	✓	☒	✓	☒	☒
S7G2	✓	☒	✓	☒	☒

14.22 独立ウォッチドッグタイマ (IWDT)

独立ウォッチドッグタイマ (IWDT) HAL モジュールは、ウォッチドッグタイマアプリケーションのためのハイレベルの API を提供し、`r_iwdt` に実装されています。ウォッチドッグタイマは、MCU の IWDT 周辺機能を使用します。ユーザー定義のコールバックを作成し、イベント通知に応答できます。

- WDT がアンダーフローした場合、または許可されたリフレッシュウィンドウの外でリフレッシュされた場合には、次のいずれかのイベントが生じる可能性があります。
 - デバイスのリセット
 - NMI の生成
- 内部ウォッチドッグタイマ周辺機能 (IWDT) をサポートします。IWDT は独自のクロックソースを持ち、安全性が向上します。
- リセット後の自動ハードウェア構成をサポートします。

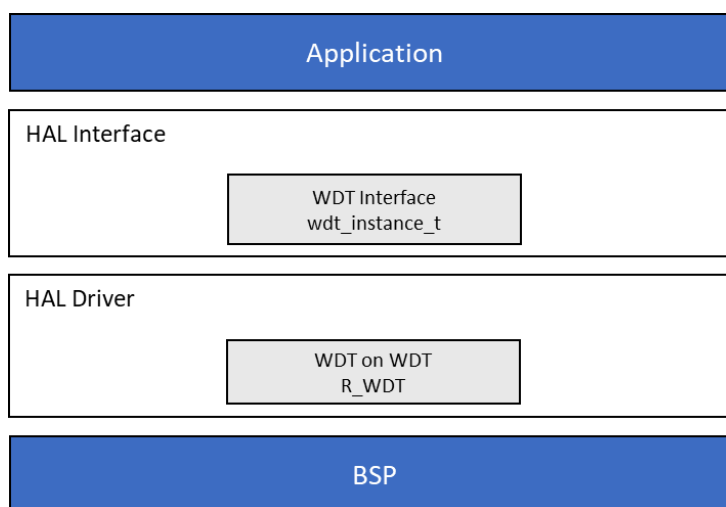


図 14.22 IWDT HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の IWDT の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	カウントダウン	オートスタートモード	リセット出力	割り込み要求出力
S124	✓	✓	✓	✓
S128	✓	✓	✓	✓
S1JA	✓	✓	✓	✓
S3A1	✓	✓	✓	✓
S3A3	✓	✓	✓	✓
S3A6	✓	✓	✓	✓
S3A7	✓	✓	✓	✓
S5D5	✓	✓	✓	✓
S5D9	✓	✓	✓	✓
S7G2	✓	✓	✓	✓

グループ名	スリープモードカウント停止制御出力	ELC HAL ドライバーによるイベントリンク	ウィンドウ機能	カウンタ停止条件 (リセット、カウンタのアンダーフローまたはリフレッシュエラー発生)	リフレッシュ許可期間外でのリフレッシュ	カウンタ値の読み出し
S124	☒	☒	✓	✓	✓	✓
S128	☒	☒	✓	✓	✓	✓
S1JA	☒	☒	✓	✓	✓	✓
S3A1	☒	☒	✓	✓	✓	✓
S3A3	☒	☒	✓	✓	✓	✓
S3A6	☒	☒	✓	✓	✓	✓
S3A7	☒	☒	✓	✓	✓	✓
S5D5	☒	☒	✓	✓	✓	✓
S5D9	☒	☒	✓	✓	✓	✓
S7G2	☒	☒	✓	✓	✓	✓

グループ名	リセット後のクロック周波数分周	IWDT のタイムアウト期間の選択
S124	☒	✓ (BSP の OFS レジスタで設定)
S128	☒	✓ (BSP の OFS レジスタで設定)
S1JA	☒	✓ (BSP の OFS レジスタで設定)
S3A1	☒	✓ (BSP の OFS レジスタで設定)
S3A3	☒	✓ (BSP の OFS レジスタで設定)
S3A6	☒	✓ (BSP の OFS レジスタで設定)
S3A7	☒	✓ (BSP の OFS レジスタで設定)
S5D5	☒	✓ (BSP の OFS レジスタで設定)
S5D9	☒	✓ (BSP の OFS レジスタで設定)
S7G2	☒	✓ (BSP の OFS レジスタで設定)

14.23 インพุットキャプチャ付き汎用 PWM タイマ (GPT_INPUT_CAPTURE)

入力キャプチャ HAL モジュールは、入力パルス幅を測定するためのハイレベルの API を提供し、`r_gpt_input_capture` に実装されています。入力キャプチャ HAL モジュールは、MCU 上の GPT 周辺機能で使用する入力キャプチャパラメータを構成します。ユーザー定義のコールバックを作成し、新しい測定が完了するたびに値を取得することができます。

入力キャプチャ HAL モジュールは、入力キャプチャ機能用の GPT を構成します。

- 入力キャプチャ HAL を使用すると、次のタスクを実行できます。
 - モジュールを初期化する
 - 入力キャプチャの測定を有効にする
 - 入力キャプチャの測定を無効にする
 - 測定カウンタの状態（実行中かどうか）を取得する
 - 最後にキャプチャされたタイマ/オーバーフローカウンタの値を取得する
 - 入力キャプチャ動作を閉じる
- 入力キャプチャ HAL モジュールは以下のものをサポートします。
 - パルス幅の測定のみ
 - 立ち上がりエッジまたは立ち下がりエッジの測定の開始
 - ワンショットモードまたは周期モード
 - キャプチャを有効にするためのハードウェアイネーブル信号（ローイネーブル/ハイイネーブル）
 - 以下のイベントでのコールバック関数：
 - カウンタオーバーフロー
 - 入力キャプチャ発生
- 発生した割り込みおよび関連するカウンタ値など、割り込みイベントに関するデータを提供するコールバック構造体があります。

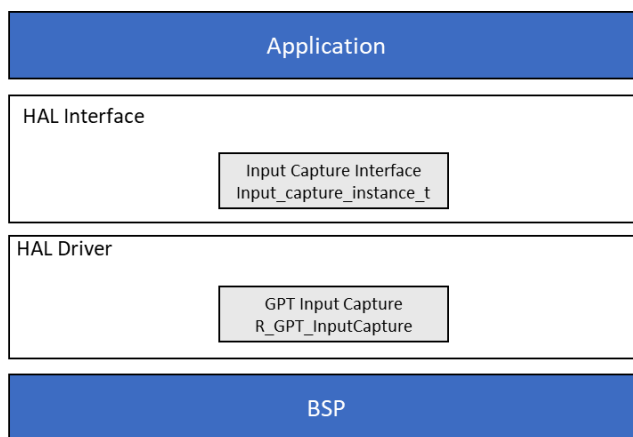


図 14.23 GPT_INPUT_CAPTURE HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の IWDWT の機能は以下のとおりです。

【記号の説明】

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	のこぎり波	三角波	ブラシレス直流モーターを制御するための PWM 波形	コンペアマッチ出力 (Low 出力、High 出力、トグル出力)	インพุットキャプチャ機能	デッドタイム自動付加機能
S124	☒	☒	☒	☒	✓	☒
S128	☒	☒	☒	☒	✓	☒

グループ名	のこぎり波	三角波	ブラシレス直流モーターを制御するための PWM 波形	コンペアマッチ出力 (Low 出力、High 出力、トグル出力)	インプットキャプチャ機能	デッドタイム自動付加機能
S1JA	☒	☒	☒	☒	✓	☒
S3A1	☒	☒	☒	☒	✓	☒
S3A3	☒	☒	☒	☒	✓	☒
S3A6	☒	☒	☒	☒	✓	☒
S3A7	☒	☒	☒	☒	✓	☒
S5D5	☒	☒	☒	☒	✓	☒
S5D9	☒	☒	☒	☒	✓	☒
S7G2	☒	☒	☒	☒	✓	☒

グループ名	PWM モード	位相計数機能	ELC HAL ドライバーによるイベントリンク	ノイズフィルタ機能
S124	☒	☒	☒	✓
S1JA	☒	☒	☒	✓
S3A1	☒	☒	☒	✓
S3A7	☒	☒	☒	✓
S5D9	☒	☒	☒	✓
S7G2	☒	☒	☒	✓

14.24 割り込みコントローラユニット (外部割り込み) (ICU)

外部 IRQ HAL モジュールは、MCU 上の外部 IRQ ピンを構成し、使用するための API です。外部 IRQ HAL モジュールは r_icu に実装され、MCU の割り込みコントローラユニット (ICU) を使用します。

- 対象の MCU で使用可能な外部割り込み要求ピン
- 複数の機能オプション
 - 割り込みの生成の有効化と無効化
 - IRQ ノイズフィルタの有効化と無効化
 - 外部ピン IRQ トリガの設定 (IRQ ピンでの立ち上がりエッジ、立ち下がりエッジ、低レベル)
- ユーザーコールバック関数の構成。この関数は、外部ピン割り込みが生成されると、HAL モジュールによって呼び出されます。

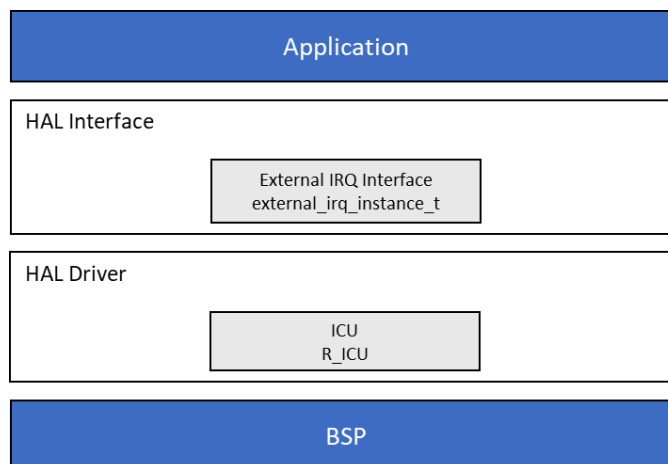


図 14.24 外部割り込みコントローラユニット (ICU)

SSP でサポート対象およびサポート対象外の ICU の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	周辺機能割り込み	外部端子割り込み	DTC および DMAC 制御	NVIC への割り込み要因	ノンマスカブル割り込み (NMI)	低消費電力モードからの復帰
S124	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S128	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S1JA	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S3A1	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S3A3	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S3A6	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S3A7	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S5D5	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S5D9	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)
S7G2	✓	✓	✓	✓	✓ (注を参照)	✓ (注を参照)

注 :

- ・ SSP の ICU モジュール (r_icu) は、外部端子割り込みのみに対応します。上に示す他の機能には対応しません。
- ・ 周辺機能割り込みは、MCU ごとおよび SSP の周辺機能ドライバーモジュールごとにボードサポートパッケージ (BSP) によって制御されます。
- ・ DTC または DMAC の制御は、SSP の DTC モジュールまたは DMAC モジュール (r_dtc または r_dmac) によって処理されます。
- ・ SSP では、ノンマスカブル割り込み (NMI) として、IWDT アンダーフロー、WDT アンダーフロー、および電圧モニタ割り込みがサポートされます。これらの NMI は、それぞれ IWDT、WDT、または LVD モジュール (r_iwdt、r_wdt、または r_lvd) によって制御されます。
- ・ ウェイクアップ割り込み要求イネーブルの設定は、LVD モジュール (r_lvd) によってサポートされます。低消費電力モードの詳細については、「14.30 ローパワーモード V2 (LPMV2)」を参照してください。

グループ名	RPEST	RECCST	BUSST	BUSMST	SPEST
S124	☒	N/A	N/A	N/A	N/A
S128	☒	N/A	N/A	N/A	N/A
S1JA	☒	☒	☒	☒	☒
S3A1	☒	☒	☒	☒	☒
S3A3	☒	☒	☒	☒	☒
S3A6	☒	☒	☒	☒	☒
S3A7	☒	☒	☒	☒	☒
S5D5	☒	☒	☒	☒	☒
S5D9	☒	☒	☒	☒	☒
S7G2	☒	☒	☒	☒	☒

14.25 汎用 I/O ポート (GPIO/IOPORT)

I/O ポート HAL モジュールは I/O ピン制御用のハイレベルの API を提供し、r_ioport に実装されています。I/O ポート HAL モジュールは MCU ピンを構成し、MCU ピンを操作する機能を提供します。I/O ピンの動作状態は、Synergy コンフィギュレータを使用して設定できます。Synergy プロジェクトが構築されると、ピン構成ファイルが作成されます。アプリケーションが実行されると、BSP が該当する API を使用して、適切にポートを構成します。

このモジュールは、1 つ以上の I/O ピンを構成します。ピンの方向は、以下に示す他のいくつかのオプションとともに設定できます。

- プルアップ
- NMOS/PMOS
- ドライブ強度

- イベントエッジトリガ (立ち下がり、立ち上がり、または両方)
- 端子を割り込み端子として使用するかどうか
- 端子をアナログ端子として使用するかどうか
- 端子を周辺機能端子として使用するかどうかとその周辺機能

このドライバーは以下の機能をサポートしています。

- ポート上の 1 つ以上の端子の方向の変更
- ポート上の 1 つ以上の端子への書き込み
- ポート上の 1 つ以上の端子からの読み取り
- イベント出力データの設定
- イベント入力データの読み取り

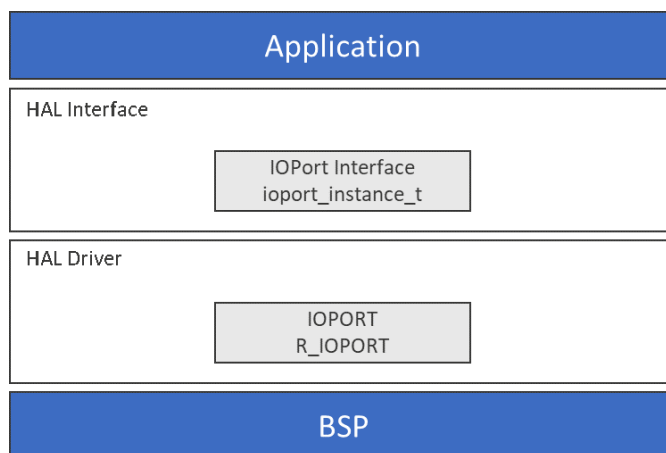


図 14.25 I/O ポート HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の GPIO の機能は以下のとおりです。

【記号の説明】

- ✓ : 使用可能 (テスト済み)
- ☒ : 使用不可 (未テスト/機能せず、またはその両方)
- N/A : MCU のサポート対象外

グループ名	ポートの直接設定	入力データリード機能	出力ポートライト機能	端子モード制御	イーサネットモード	ELC_PORTn イベント入力リード機能 ^注
S124	✓	✓	✓	✓	N/A	✓
S128	✓	✓	✓	✓	N/A	✓
S1JA	✓	✓	✓	✓	N/A	✓
S3A1	✓	✓	✓	✓	N/A	✓
S3A3	✓	✓	✓	✓	N/A	✓
S3A6	✓	✓	✓	✓	N/A	✓
S3A7	✓	✓	✓	✓	N/A	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	ELC_PORTn イベント出力 設定 ^注
S124	✓
S128	✓
S1JA	✓
S3A1	✓
S3A3	✓
S3A6	✓
S3A7	✓
S5D5	✓
S5D9	✓
S7G2	✓

注：ELC のイベントリンクは ELC API を使用して設定するのではなく、ユーザーがリンクを設定することで使用できるようになります。

14.26 JPEG コーデック

JPEG デコード HAL モジュールは、JPEG 画像をデコードするためのハイレベルの API を提供します。JPEG デコード HAL モジュールは、MCU の JPEG コーデック周辺機能を使用します。ユーザーコールバック関数を使用して、アプリケーションプログラムにキー処理イベントを通知できます。

JPEG デコードモジュールは以下の機能に対応します。

- JPEG 解凍をサポート
- JPEG デコーダが完了するまでアプリケーションが待機できるようにするポーリングモードをサポート
- ユーザーが指定したコールバック関数を使用した割り込みモードをサポート
- 水平および垂直サブサンプル値、水平ストライド、デコードされたピクセル フォーマット、入力および出力データフォーマット、色空間などのパラメータを設定
- 画像をデコードする前にそのサイズを取得
- デコードされた画像フレームの保管のため、コード化されたデータを入力バッファおよび出力バッファに格納する操作をサポート
- JPEG デコードモジュールに符号化されたデータ (coded data) をストリーミングすることをサポート。この機能は、全体のイメージをバッファリングすることなしに、ファイルとネットワークから符号化された (coded) JPEG イメージを読み込む応用的な使用を許可します
- デコードする画像のライン数を指定するインタフェース。この機能により、アプリケーションは、すべてのフレームをバッファリングする必要なく、伸張された画像をリアルタイムで処理可能
- 入力デコード形式は YCbCr444、YCbCr422、YCbCr420、YCbCr411
- 出力デコード形式は ARGB8888 および RGB565
- JPEG 画像のサイズ、高さ、および幅が要件を満たさない場合にエラーを返すことが可能

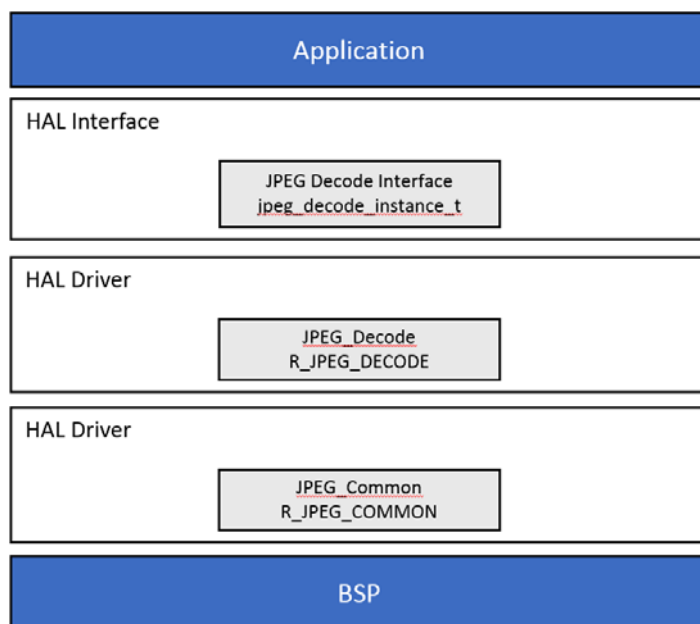


図 14.26 JPEG デコード HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の JPEG の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	YCbCr444 時 8 ピクセル × 8 ラ イン	YCbCr422 時 16 ピクセル × 8 ラ イン	YCbCr411 時、 32 ピクセル × 8 ライン	YCbCr420 時、 16 ピクセル × 16 ラ イン	出力デコード フォーマット ARGB8888	出力デコード フォーマット RGB565
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

14.27 JPEG エンコード

JPEG エンコード HAL モジュールは、画像をエンコードするためのハイレベルの API を提供します。JPEG エンコード HAL モジュールは、MCU 上の JPEG コーデック周辺機能を使用します。ユーザーコールバック関数を使用して、アプリケーションプログラムにキー処理イベントを通知できます。

JPEG コーデックエンコードモジュールは以下の機能に対応します。

- JPEG 圧縮をサポート
- JPEG エンコーダが完了するまでアプリケーションが待機できるようにするポーリングモードをサポート
- ユーザーが指定したコールバック関数を使用した割り込みモードをサポート
- 水平および垂直解像度、水平ストライド、および品質係数などのパラメータを構成
- エンコードおよび圧縮された JPEG 画像の保管のため、未加工の画像データを入力バッファおよび出力

バッファに格納する操作をサポート

- JPEG エンコーダモジュールへの未加工画像データのストリーム転送をサポート。この機能により、アプリケーションは、キャプチャデバイスやカメラまたはネットワークからのコード化された未加工画像の読み取りを、画像全体をバッファリングすることなく実行可能
- 入力用に YCbCr422 色空間のみをサポート
- モーション JPEG (MJPEG) ビデオ圧縮/解凍システムの作成用に構成可能
- RTP ストリーミングアプリケーション用に DRI Maker をサポート
- 品質係数構成のサポート：品質係数値によって出力画像の品質を決定

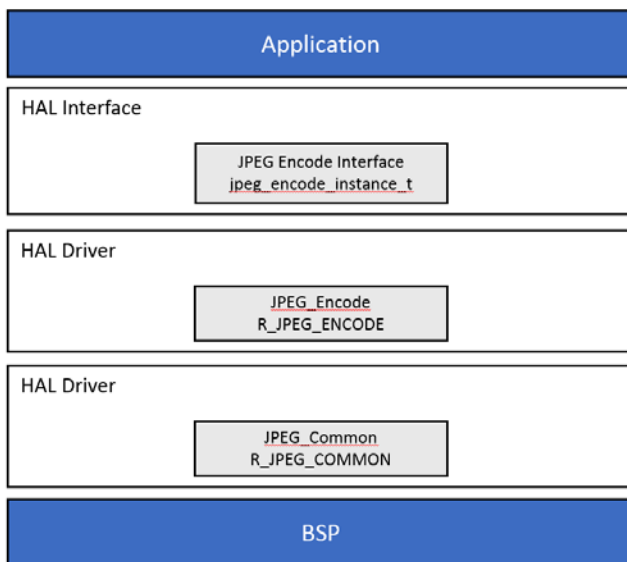


図 14.27 JPEG エンコード HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の JPEG の機能は以下のとおりです。

[記号の説明]

✓：使用可能 (テスト済み)

☒：使用不可 (未テスト/機能せず、またはその両方)

N/A：MCU のサポート対象外

グループ名	YCbCr444 時 8 ピクセル × 8 ラ イン	YCbCr422 時 16 ピクセル × 8 ラ イン	YCbCr411 時、 32 ピクセル × 8 ライン	YCbCr420 時、 16 ピクセル × 16 ラ イン	出力フォーマット JPEG
S124	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	✓	N/A	N/A	✓
S7G2	N/A	✓	N/A	N/A	✓

14.28 キー割り込み (KINT)

Key Matrix HAL モジュールは、Key Matrix HAL アプリケーションのためのハイレベルの API を提供し、`r_kint` に実装されています。Key Matrix HAL モジュールは、MCU 上でキー割り込み関数周辺機能を使用しています。ユーザー定義のコールバックを作成し、CPU にキー押下イベントを通知できます。

この Key Matrix HAL モジュールは、キー割り込み (KINT) 周辺機能を構成および制御します。次のキー機能を実装します。

- KINT チャンネルでの立ち上がりエッジと立ち下がりエッジの両方をサポート
- 割り込みベースのイベント通知をサポート
- 複数のイベントを効率よくキャプチャするためのビットマスク機能をサポート
- 任意の 2 つのチャンネルにエッジのあるマトリックスキーパッドをサポート

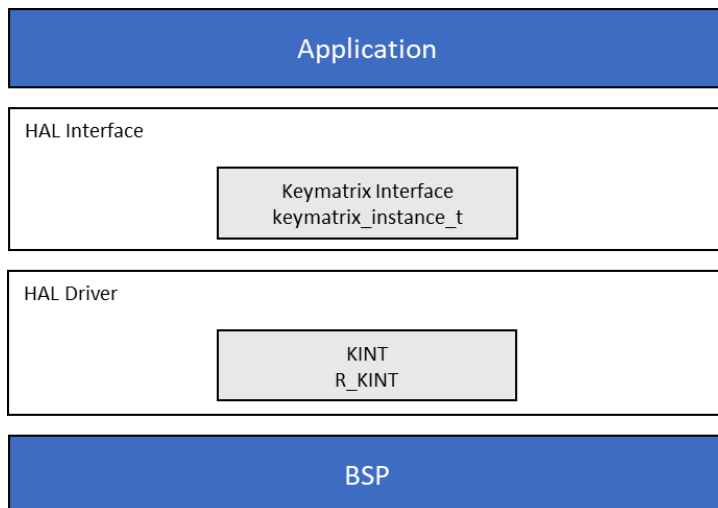


図 14.28 Key Matrix HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の KINT の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	KR00~KR07 の入力切り替え
S124	✓
S128	✓
S1JA	✓
S3A1	✓
S3A3	✓
S3A6	✓
S3A7	✓
S5D5	✓
S5D9	✓
S7G2	✓

14.29 ローパワーモード V1 (LPMV1)

低消費電力モード HAL モジュールは、低消費電力モードアプリケーションのためのハイレベルの API を提供し、`r_lpm` に実装されています。低消費電力モード HAL モジュールは、MCU 上で低消費電力モードハードウェア周辺機能を使用します。

LPM HAL モジュールは、低消費電力モードハードウェア周辺機能を使用して、MCU 動作電力制御モードと MCU 低消費電力モードの構成をサポートしています。

LPM ドライバーは、以下の動作モードをサポートしています。

- 低電圧モード
- 低速モード
- 中速モード
- 高速モード
- サブオシレータモード

LPMV2 ドライバーは、以下のローパワーモードをサポートしています。

- ディープソフトウェアスタンバイモード
- ソフトウェアスタンバイモード
- スリープモード
- スヌーズ

注：LPMV1 は、非推奨のモジュールです。新しいプロジェクトには LPMV1 を使用せず、LPMV2 HAL モジュールを使用してください。すべての MCU で、すべての低消費電力モードが利用可能とは限りません。すべての MCU で、すべての動作モードが利用可能とは限りません。

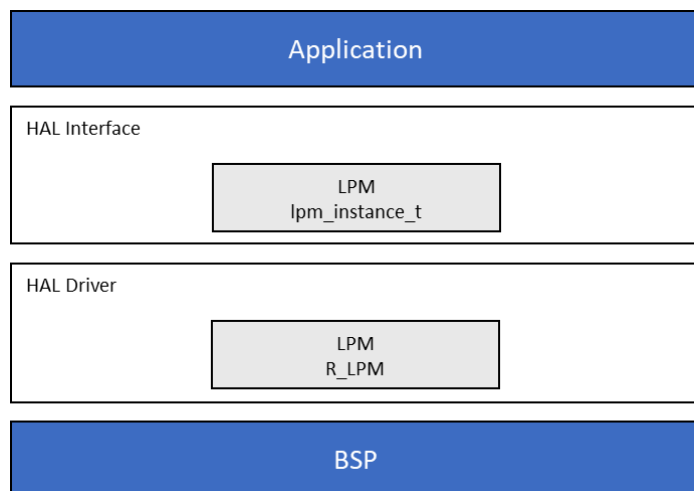


図 14.29 LPMV1 HAL モジュールのブロック図

14.30 ローパワーモード V2 (LPMV2)

低消費電力モード V2 HAL モジュールは、低消費電力モードアプリケーションのためのハイレベルの API を提供し、`r_lpm2` に実装されています。低消費電力モード V2 HAL モジュールは、MCU 上で低消費電力モードハードウェア周辺機能を使用します。

LPM V2 HAL モジュールは、低消費電力モードハードウェア周辺機能を使用して、MCU 動作電力制御モードと MCU 低消費電力モードの構成をサポートしています。

- ディープソフトウェアスタンバイモード
- ソフトウェアスタンバイモード
- スリープモード

● スヌーズ

LPM V2 HAL モジュールは、ディープスタンバイモード時に、内部の電力供給制御と I/O ポートの状態の再設定を通じて消費電力の低減をサポートします。LPM V2 HAL モジュールは、MCU の他のハードウェア周辺機能の無効化と有効化をサポートしています。

注：すべての MCU グループで、すべての低消費電力 V2 モードを利用できるとは限りません。低消費電力モード V2 HAL モジュールは、MCU の動作電力制御モードを処理しなくなります。この機能は、CGC HAL モジュールによって処理されます。

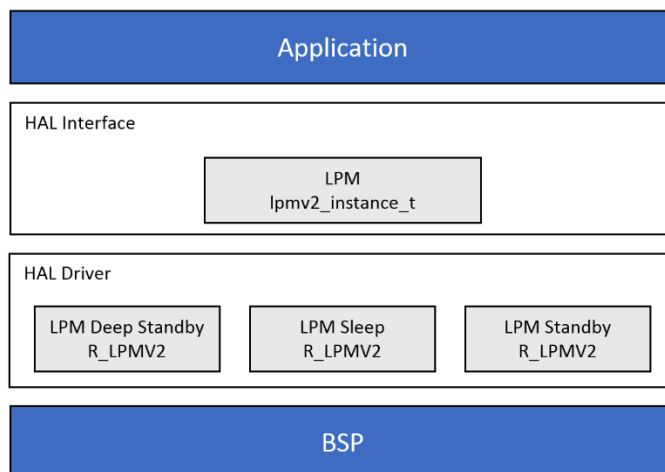


図 14.30 LPMV2 HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の LPMV2 の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	モジュールストップ/スタートビットのアクセス	低電圧動作電力制御モード	サブオシレータ速度動作電力制御モード
S124	✓	☒	✓
S128	✓	☒	✓
S1JA	✓	☒	✓
S3A1	✓	☒	✓
S3A3	✓	☒	✓
S3A6	✓	☒	✓
S3A7	✓	☒	✓
S5D5	✓	N/A	✓
S5D9	✓	N/A	✓
S7G2	✓	N/A	✓

グループ名	スリープ	ソフトウェアスタンバイ	ディープソフトウェアスタンバイ	ソフトウェアスタンバイモードでのスヌーズの有効化	ELC を使用したスヌーズリンク	スヌーズモードの DTC 状態
S124	✓	✓	N/A	✓	☒	✓
S128	✓	✓	N/A	✓	☒	✓

S1JA	✓	✓	N/A	✓	☒	✓
S3A1	✓	✓	N/A	✓	☒	✓
S3A3	✓	✓	N/A	✓	☒	✓
S3A6	✓	✓	N/A	✓	☒	✓
S3A7	✓	✓	N/A	✓	☒	✓
S5D5	✓	✓	✓	✓	☒	✓
S5D9	✓	✓	✓	✓	☒	✓
S7G2	✓	✓	✓	✓	☒	✓

グループ名	スタンバイまたはディープソフトウェアスタンバイモードのアドレスバスおよびバス信号の状態	RXD0 (SCI0) を介したスヌーズモードへの移行	ディープソフトウェアスタンバイモードから復帰後の IO ポートの状態制御	ディープソフトウェアスタンバイモードでの内部電力供給制御 (LOCO、スタンバイ SRAM、AGTn、USBHS/FS への電力供給)
S124	☒	✓	✓	☒
S128	☒	✓	✓	☒
S1JA	☒	✓	✓	☒
S3A1	✓	✓	✓	☒
S3A3	✓	✓	✓	☒
S3A6	✓	✓	✓	☒
S3A7	✓	✓	✓	☒
S5D5	✓	✓	✓	✓
S5D9	✓	✓	✓	✓
S7G2	✓	✓	✓	✓

14.31 低電圧検出 (LVD)

低電圧検出 (LVD) HAL モジュールは、電圧検出アプリケーションのためのハイレベルの API を提供し、r_lvd に実装されています。LVD HAL モジュールは、MCU 上の LVD 周辺機能を使用します。ユーザー定義のコールバックを作成し、電圧検出イベントがトリガされたときに CPU に通知できます。VCC は、すべての電圧検出機能のソースです。

- 電圧検出入力としての V_{CC}
- 1 つのビルド時に構成可能な低電圧検出 (OFS1 レジスタを使用)
- 2 つの実行時に構成可能な低電圧検出
- 2 つの結果フラグ。1 つはしきい値チェック用、もう 1 つは現在の状態用
- 割り込みとポーリングイベントチェックの両方をサポート

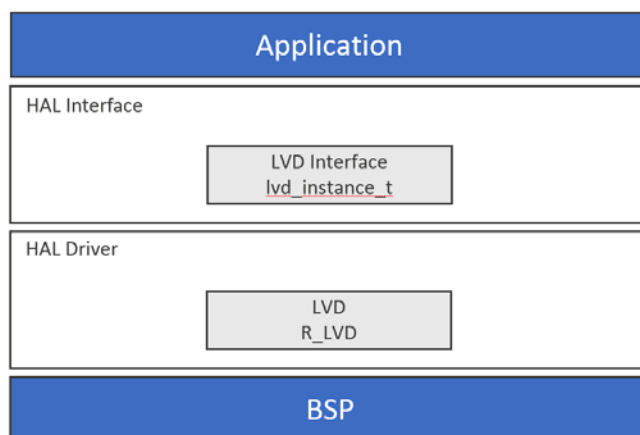


図 14.31 LVD HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の LVD の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	VCC 上昇電圧による割り込みまたはノンマスクابل割り込みの生成	VCC 下降電圧による割り込みまたはノンマスクابل割り込みの生成	VCC 上昇電圧および下降電圧による割り込みまたはノンマスクابل割り込みの生成	マスク可能割り込みおよびノンマスクابل割り込みのコールバック通知	電圧下降時のリセット	ポーリングによる LVD 1 および 2 ステータスフラグの監視
S124	✓	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	調整可能なフィルタ時間によるデジタルフィルタリング	ELC HAL ドライバーによるイベントリンク
S124	N/A	☒
S128	N/A	☒
S1JA	N/A	☒
S3A1	N/A	☒
S3A3	N/A	☒
S3A6	N/A	☒
S3A7	N/A	☒
S5D5	✓	☒
S5D9	✓	☒
S7G2	✓	☒

14.32 オペアンプ (OPAMP)

OPAMP HAL モジュールでは、`r_opamp` 上の信号増幅アプリケーション用の OPAMP API を実装します。このモジュールは、MCU 上の OPAMP 周辺機能をサポートします。

- 低消費電力モードまたは高速モード
- ソフトウェアまたは AGT コンペアマッチによる開始
- ソフトウェアまたは ADC 変換終了による停止（ADC 変換終了による停止は、AGT コンペアマッチによる開始が構成されたオペアンプチャンネルでのみサポート）
- 一部の MCU でトリミングが利用可能（ハードウェアマニュアルを参照）

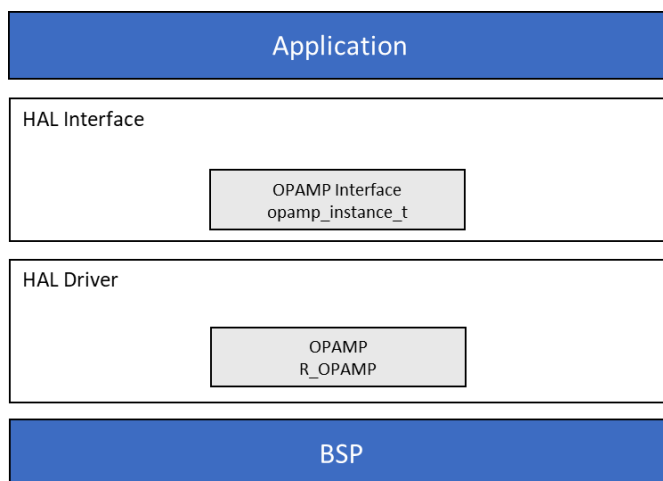


図 14.32 OPAMP HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の OPAMP の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	ローパワーモード	ハイスピードモード	ソフトウェアによる開始	AGT コンペアマッチ	ソフトウェアによるストップ	ADC 変換による終了
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	☒	☒	☒	☒	☒	☒
S1JA	✓	✓	✓	✓	✓	✓
S3A1	☒	☒	☒	☒	☒	☒
S3A3	☒	☒	☒	☒	☒	☒
S3A6	☒	☒	☒	☒	☒	☒
S3A7	☒	☒	☒	☒	☒	☒
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A	N/A	N/A

14.33 パラレルデータキャプチャユニット (PDC)

パラレルデータキャプチャユニット (PDC) HAL モジュールは、カメラアプリケーションから画像をキャプチャするためのハイレベルの API を提供し、r_pdc に実装されています。PDC HAL モジュールは、MCU 上の PDC 周辺機能を使用します。ユーザー定義のコールバックを作成し、キャプチャ完了時に CPU に通知できます。

- 接続された構成済みのカメラからのキャプチャをサポート
- キャプチャの完了を CPU に通知するコールバックをサポート
- キャプチャバッファへのポインタを提供
- コールバックをトリガしたイベントを示す値を提供

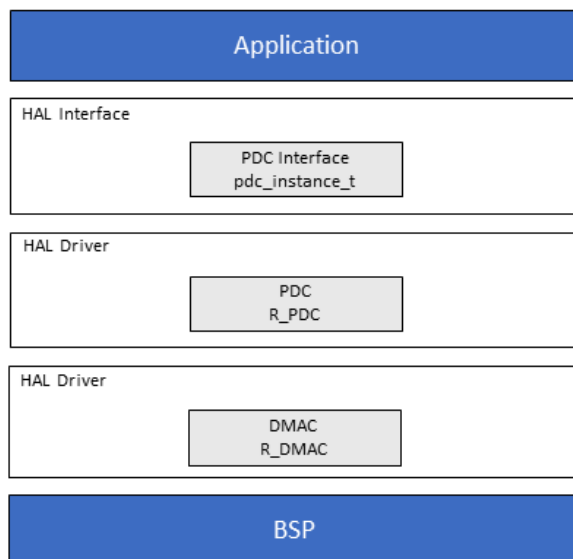


図 14.33 PDC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の OPAMP の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	最大 4095 ラインの垂直方向のサポート	4~4095 バイトの水平方向のサポート	受信データレディ割り込み	フレームエンド割り込み	オーバーラン割り込み	アンダーラン割り込み
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	ラインの本数誤りによるエラーからの割り込みの受け入れ	ライン 1 本当たりのバイト数誤りによるエラーからの割り込みの受け入れ	フレームエンドと受信データレディによる DTC 起動	フレームエンドと受信データレディによる DMAC 起動	周波数分周比 : 2、4、6、8、10、12、14、16 から選択可能	PDC リセット機能のサポート
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	✓	✓	☒	☒	☒	☒
S5D9	✓	✓	☒	☒	☒	☒
S7G2	✓	✓	☒	☒	☒	☒

グループ名	VSYNC と HSYNC の選択可能なアクティブ極性のサポート	VSYNC と HSYNC の監視のサポート	エンディアン順序が選択可能
S124	N/A	N/A	N/A
S128	N/A	N/A	N/A
S1JA	N/A	N/A	N/A
S3A1	N/A	N/A	N/A
S3A3	N/A	N/A	N/A
S3A6	N/A	N/A	N/A
S3A7	N/A	N/A	N/A
S5D5	✓	✓	✓
S5D9	✓	✓	✓
S7G2	✓	✓	✓

14.34 クワッド SPI (QSPI)

クワッド SPI (QSPI) HAL モジュールは、MCU に接続された QSPI フラッシュデバイスの内容を消去およびプログラミングするためのハイレベルの API を提供し、MCU 上の r_qspi 周辺機能に実装されています。他のモジュールの多くと異なり、QSPI 用のコールバック関数はありません。

- QSPI の初期化
- 直接通信モードを使用したクワッド SPI フラッシュデバイスへのアクセス
- QSPI フラッシュデバイスのデータの読み取り
- QSPI フラッシュデバイスのページのプログラミング
- QSPI フラッシュデバイスのセクターの消去
- QSPI フラッシュデバイスへのアクセスを制御するバンクの選択

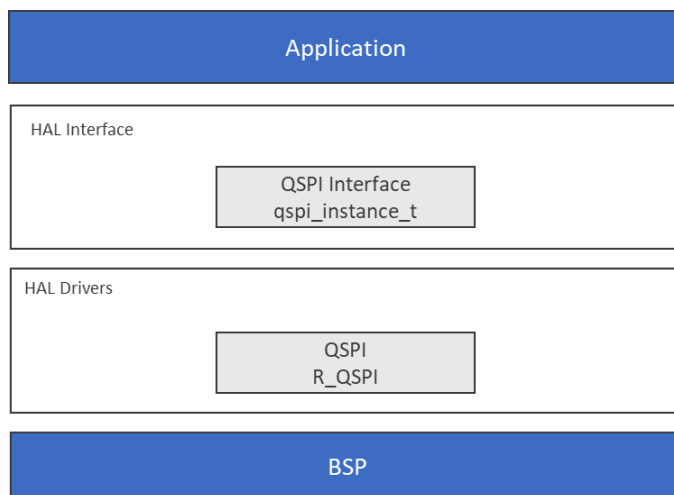


図 14.34 QSPI HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の QSPI の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	拡張 SPI	Dual-SPI	Quad-SPI	SPI モード 1、3	SFMSAC 経由でアドレスが選択可能 (8、16、24、32 ビット)	タイミング調整機能
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	N/A	☒	3 バイトおよび 4 バイトアドレス	☒
S3A3	✓	✓	N/A	☒	3 バイトおよび 4 バイトアドレス	☒
S3A6	✓	✓	N/A	☒	3 バイトおよび 4 バイトアドレス	☒
S3A7	✓	✓	✓	☒	3 バイトおよび 4 バイトアドレス	☒
S5D5	✓	✓	✓	☒	3 バイトおよび 4 バイトアドレス	☒
S5D9	✓	✓	✓	☒	3 バイトおよび 4 バイトアドレス	☒
S7G2	✓	✓	✓	☒	3 バイトおよび 4 バイトアドレス	☒

グループ名	フラッシュリード機能：リード	フラッシュリード機能：ファストリード	フラッシュリード機能：ファストリード Dual 出力	フラッシュリード機能：ファストリード Dual I/O	フラッシュリード機能：ファストリード Quad 出力	フラッシュリード機能：ファストリード Quad I/O
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	☒	☒	☒	☒	☒
S3A3	✓	☒	☒	☒	☒	☒
S3A6	✓	☒	☒	☒	☒	☒
S3A7	✓	☒	☒	☒	☒	☒
S5D5	✓	☒	☒	☒	☒	☒
S5D9	✓	☒	☒	☒	☒	☒
S7G2	✓	☒	☒	☒	☒	☒

グループ名	命令コードの代替	ダミーサイクル数の補正	プリフェッチ機能	ポーリング処理	SPI バスサイクル拡張機能	ダイレクト通信機能
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	☒	☒	✓	N/A	N/A
S3A3	N/A	☒	☒	✓	N/A	N/A
S3A6	N/A	☒	☒	✓	N/A	N/A
S3A7	N/A	☒	☒	✓	N/A	N/A
S5D5	N/A	☒	☒	✓	N/A	N/A
S5D9	N/A	☒	☒	✓	N/A	N/A
S7G2	N/A	☒	☒	✓	N/A	N/A

グループ名	割り込み要因	モジュールストップ機能	XIP
S124	N/A	N/A	N/A
S128	N/A	N/A	N/A
S1JA	N/A	N/A	N/A
S3A1	N/A	N/A	✓
S3A3	N/A	N/A	✓
S3A6	N/A	N/A	✓
S3A7	N/A	N/A	✓
S5D5	N/A	N/A	✓
S5D9	N/A	N/A	✓
S7G2	N/A	N/A	✓

14.35 リアルタイムクロック (RTC)

リアルタイムクロック (RTC) HAL モジュールは、RTC アプリケーションのためのハイレベルの API を提供し、`r_rtc` に実装されています。RTC HAL モジュールは、RTC モジュールを構成し、クロック、カレンダー、およびアラーム機能を制御します。RTC は、MCU 上のリアルタイムクロックモジュールを使用します。ユーザー定義のコールバックを作成し、サポートする 3 つの割り込みタイプ (アラーム、周期、キャリー) のどれにも応答できます。

- RTC 周辺機能の設定
- 時刻と日付の取得と設定
- 時刻と日付のアラームの取得と設定

- 時間カウンタの開始と停止
- アラームイベント、周期イベント、キャリーイベントの通知
- イベントタイプの有効化と無効化
- イベントレートの構成
- クロックソースの取得と設定

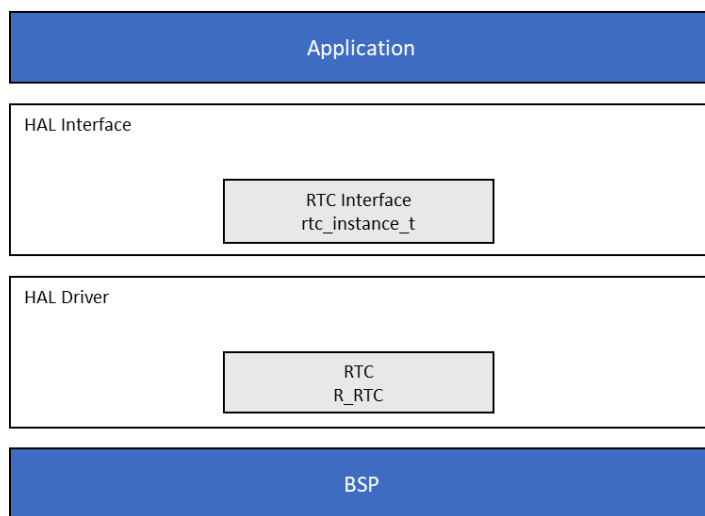


図 14.35 RTC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の RTC の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	カレンダーカウントモード	バイナリカウントモード	サブクロック (XCIN) カウントソース	LOCO カウントソース	12 時間/24 時間のモード切り替え	アラーム割り込み (RTC_ALM)
S124	✓	☒	✓	✓	24 時間	✓
S128	✓	☒	✓	✓	24 時間	✓
S1JA	✓	☒	✓	✓	24 時間	✓
S3A1	✓	☒	✓	✓	24 時間	✓
S3A3	✓	☒	✓	✓	24 時間	✓
S3A6	✓	☒	✓	✓	24 時間	✓
S3A7	✓	☒	✓	✓	24 時間	✓
S5D5	✓	☒	✓	✓	24 時間	✓
S5D9	✓	☒	✓	✓	24 時間	✓
S7G2	✓	☒	✓	✓	24 時間	✓

グループ名	周期イベント出力 (RTC_PRD)	桁上げ割り込み (RTC_CUP)	時間キャプチャ機能	ELC HAL ドライバーによるイベントリンク	スタート/ストップ機能	時間誤差補正機能
S124	✓	✓	☒	☒	✓	✓
S128	✓	✓	☒	☒	✓	✓
S1JA	✓	✓	☒	☒	✓	✓
S3A1	✓	✓	☒	☒	✓	✓
S3A3	✓	✓	☒	☒	✓	✓
S3A6	✓	✓	☒	☒	✓	✓
S3A7	✓	✓	☒	☒	✓	✓
S5D5	✓	✓	☒	☒	✓	✓
S5D9	✓	✓	☒	☒	✓	✓
S7G2	✓	✓	☒	☒	✓	✓

14.36 I²C バスインタフェース (RIIC)

SSPは、2つのRIICドライバーを提供し、I²C周辺機能をサポートする2つの異なるHALインタフェースを実装しています。

- I²C周辺機能をマスタモードで構成して操作するI²Cマスタインタフェース
- I²C周辺機能をスレーブモードで構成して操作するI²Cスレーブインタフェース

この2つのHALの実装は相互に排他的であるため、I²C周辺機能はマスタモードかスレーブモードのいずれかでのみ動作できます（マルチマスタの構成では、I²C周辺機能によってマスタ設定とスレーブ設定を動的に切り替えることはできません）。

RIICマスタドライバーとスレーブドライバーの両方に共通する機能は以下のとおりです。

- 割り込み駆動型の送信/受信処理
- イベントコードを返すことのできるコールバック関数サポート
- 最大ビットレート 100 kbps のI²C標準モードをサポート
- 最大ビットレート 400 kbps のI²Cファストモードをサポート
- ビットレート 1 Mbps のファストモードプラスをサポート（S7G2およびS5D9のみサポート）

コールバック関数は以下のイベントとともに呼び出されます。

- 転送中断
- 送信完了（送信済みバイト数が提供される）
- 受信完了（受信済みバイト数が提供される）

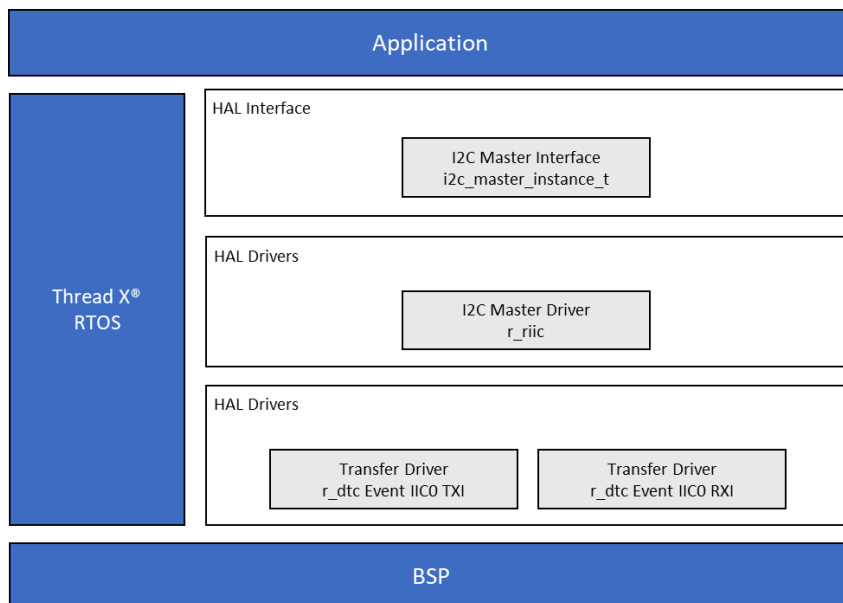


図 14.36 RIIC (マスタ) HAL モジュールのブロック図

RIIC マスタドライバーは、上記の共通機能に加えて以下の機能を備えています。

- 7 または 10 ビットアドレスを使用した I²C スレーブデバイスからのリード
- 7 または 10 ビットアドレスを使用した I²C スレーブデバイスへのライト
- ユーザーコードからの I²C 周辺機能のリセット
- I²C トランザクション前の通信スレーブアドレスの動的な設定

RIIC スレーブドライバーは、上記の共通機能に加えて以下の機能を備えています。

- I²C スレーブ動作のサポート
- I²C マスタデバイスとのトランザクションのサポート
 - リード
 - ライト
- コールバックのサポート
 - 送信完了 (送信済みバイト数が提供される)
 - 受信完了 (受信済みバイト数が提供される)

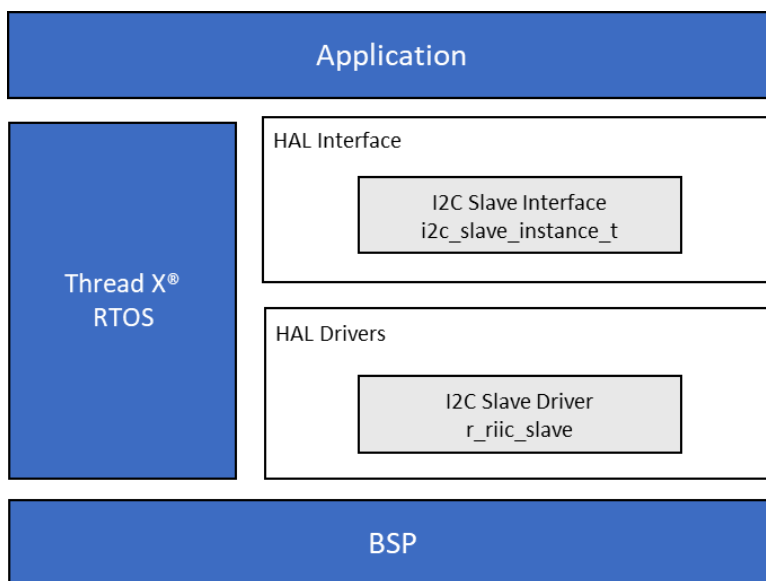


図 14.37 RIIC (スレーブ) HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の RIIC (マスタ) ドライバーの機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	I ² C フォーマット	SMBus フォーマット	マスタモード	スリープモード	ファストモードプラス	デューティサイクルの選択
S124	✓	☒	✓	✓	N/A	☒
S128	✓	☒	✓	✓	N/A	☒
S1JA	✓	☒	✓	✓	N/A	☒
S3A1	✓	☒	✓	✓	N/A	☒
S3A3	✓	☒	✓	✓	N/A	☒
S3A6	✓	☒	✓	✓	N/A	☒
S3A7	✓	☒	✓	✓	N/A	☒
S5D5	✓	☒	✓	✓	✓	☒
S5D9	✓	☒	✓	✓	✓	☒
S7G2	✓	☒	✓	✓	✓	☒

グループ名	最大 3 つの異なるスレーブアドレスの構成	7ビットおよび 10ビットのアドレスフォーマット	一般的なコールアドレス、デバイス ID アドレス、および SMBus ホストアドレスを検出可能	アクノリッジビットの自動読み込み	SDA 出力遅延機能	待機機能が選択可能 (8/9 または 9/1)
S124	☒	7ビット、10ビット	☒	✓	☒	☒
S128	☒	7ビット、10ビット	☒	✓	☒	☒
S1JA	☒	7ビット、10ビット	☒	✓	☒	☒
S3A1	☒	7ビット、10ビット	☒	✓	☒	☒
S3A3	☒	7ビット、10ビット	☒	✓	☒	☒
S3A6	☒	7ビット、10ビット	☒	✓	☒	☒
S3A7	☒	7ビット、10ビット	☒	✓	☒	☒
S5D5	☒	7ビット、10ビット	☒	✓	☒	☒
S5D9	☒	7ビット、10ビット	☒	✓	☒	☒
S7G2	☒	7ビット、10ビット	☒	✓	☒	☒

グループ名	完全なアービトレーションのサポート	内部検出のタイムアウト	プログラム可能なデジタルノイズフィルタ	すべての割り込み要因のサポート
S124	マスタ、NACK アービトレーション	✓	☒	✓
S128	マスタ、NACK アービトレーション	✓	☒	✓
S1JA	マスタ、NACK アービトレーション	✓	☒	✓
S3A1	マスタ、NACK アービトレーション	✓	☒	✓
S3A3	マスタ、NACK アービトレーション	✓	☒	✓
S3A6	マスタ、NACK アービトレーション	✓	☒	✓
S3A7	マスタ、NACK アービトレーション	✓	☒	✓
S5D5	マスタ、NACK アービトレーション	✓	☒	✓
S5D9	マスタ、NACK アービトレーション	✓	☒	✓
S7G2	マスタ、NACK アービトレーション	✓	☒	✓

SSP でサポート対象およびサポート対象外の RIIC (スレーブ) ドライバーの機能は以下のとおりです。

グループ名	I ² C フォーマット	SMBus フォーマット	マスタモード	スレーブモード	ファストモードプラス	デューティサイクルの選択
S124	✓	☒	☒	✓	N/A	☒
S128	✓	☒	☒	✓	N/A	☒
S1JA	✓	☒	☒	✓	N/A	☒
S3A1	✓	☒	☒	✓	N/A	☒
S3A3	✓	☒	☒	✓	N/A	☒
S3A6	✓	☒	☒	✓	N/A	☒
S3A7	✓	☒	☒	✓	N/A	☒
S5D5	✓	☒	☒	✓	✓	☒
S5D9	✓	☒	☒	✓	✓	☒
S7G2	✓	☒	☒	✓	✓	☒

グループ名	最大 3 つの異なるスレーブアドレスの構成	7 ビットおよび 10 ビットのアドレスフォーマット	一般的なコールアドレス、デバイス ID アドレス、および SMBus ホストアドレスを検出可能	アクノリッジビットの自動読み込み	SDA 出力遅延機能	待機機能が選択可能 (8/9 または 9/1)
S124	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S128	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S1JA	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S3A1	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S3A3	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S3A6	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S3A7	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S5D5	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S5D9	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒
S7G2	1 つのスレーブアドレス	7 ビット、10 ビット	☒	✓	☒	☒

グループ名	完全なアービトレーションのサポート	内部検出のタイムアウト	プログラム可能なデジタルノイズフィルタ	すべての割り込み要因のサポート	ELC HAL ドライバーによるイベントリンク
S124	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S128	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S1JA	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S3A1	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S3A3	スレーブおよび NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S3A6	スレーブおよび NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S3A7	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S5D5	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S5D9	スレーブ、NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒
S7G2	スレーブおよび NACK アービトレーション	✓	☒	TEI 割り込みは対象外	☒

14.37 SD マルチメディアカード (SDMMC)

SDMMC (SD/MMC および SDIO) モジュールは `r_sdmmc` 上に実装され、SD/MMC メディアデバイスならびに SDIO カードの読み取り/書き込みと制御に使用されます。SDMMC モジュールはスタンドアロン SD カードとして使用できます。また、eMMC、メディアドライバや SDMMC モジュールは、FileX™やその他の互換性のあるファイルシステムとともに使用することもできます。`r_sdmmc` モジュールは、MCU 上の SD/MMC 周辺機能を使用します。

- SDSC (SD 標準容量)、SDHC (SD 大容量)、SDXC (SD 拡張容量)、eMMC (組み込みマルチメディアカード) の各メモリデバイスをサポート
 - SD および eMMC メモリデバイスのリード、ライト、消去をサポート
 - 1、4、または 8 ビット (eMMC のみ) データバスをサポート
 - ハードウェアライトプロテクトの検出をサポート (SD カードのみ)
 - 下位互換モードと高速シングルデータレート (SDR) モード (eMMC) 間を自動的に選択
- SDIO のサポート
 - SDIO シングルレジスタアクセスをサポート (CMD52)
 - SDIO マルチレジスタアクセスをサポート (CMD53)
 - SDIO 割り込みをサポート
- ホスト (MCU) とデバイスの両方でサポートされる最大クロックレートにクロックを自動的に構成

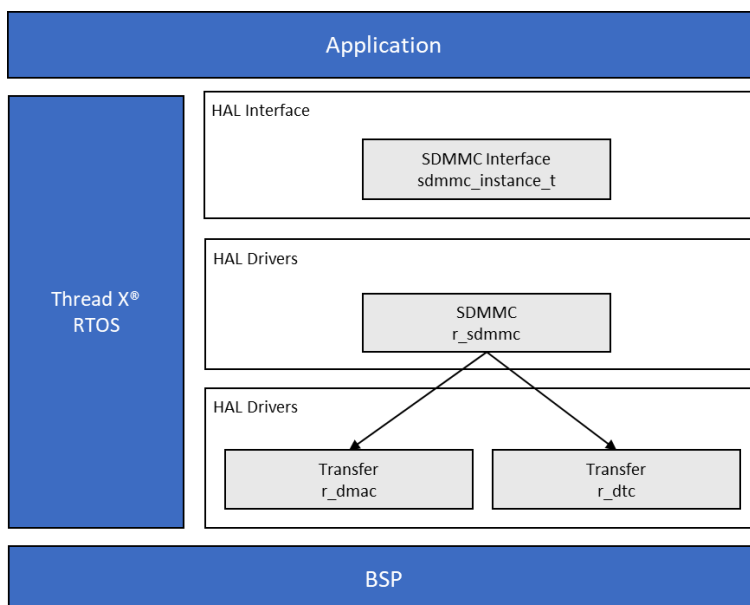


図 14.38 SDMMC HAL ドライバーインターフェース

SSP でサポート対象およびサポート対象外の SDMMC (SDIO) の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	SD 1 ビット	SD 4 ビット	SDHC	SDXC	MMC 1 ビット	MMC4 ビット
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	✓	✓	☒	☒
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	✓	✓	✓	✓	☒	☒
S5D5	✓	✓	✓	✓	☒	☒
S5D9	✓	✓	✓	✓	☒	☒
S7G2	✓	✓	✓	✓	✓	✓

グループ名	MMC 8 ビット	MMC 下位 互換性モード	カード検出 (挿入)	カード検出 (取り出し)	ライト保護	MMC 高速 SDR モード
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	☒
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	☒	☒	N/A	N/A	✓	☒
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	☒	☒	N/A	N/A	✓	☒
S5D5	☒	☒	注	注	✓	☒
S5D9	☒	☒	注	注	✓	☒
S7G2	☒	☒	注	注	✓	✓

注 : 端子の割り当てに応じて一部の MCU で使用可能。

グループ名	DMA リード	DMA ライト	レスポンスとデータのない CMD	データのない CMD	単一ブロック リード	単一ブロック ライト
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	☒	☒	✓	✓
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	✓	✓	☒	☒	✓	✓
S5D5	✓	✓	☒	☒	✓	✓
S5D9	✓	✓	☒	☒	✓	✓
S7G2	✓	✓	☒	☒	✓	✓

グループ名	複数ブロック リード (内部 タイマ)	複数ブロック ライト (外部 タイマ)	複数ブロック リード (内部 タイマ)	複数ブロック ライト (外部 タイマ)	I/O リード/ライト (直接)	I/O リード/ライト (拡張)
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	☒	✓	☒	✓	✓
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	✓	☒	✓	☒	✓	✓
S5D5	✓	☒	✓	☒	✓	✓
S5D9	✓	☒	✓	☒	✓	✓
S7G2	✓	☒	✓	☒	✓	✓

グループ名	サスペンド再開	SPI バス	ストリーム転送 (MMC)	HPI (MMC)	ブート/代替ブート (MMC)	変更可能な複数のブロック転送 (MMC)
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	N/A	N/A	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A	N/A	N/A

14.38 セキュア暗号エンジン (SCE)

セキュア暗号化エンジン (SCE) HAL モジュールは、乱数生成、ダイジェスト計算 (ハッシュ)、データ暗号化および復号化、デジタル署名および検証、キー生成 (RSA、AES、および ECC アルゴリズムを使用)、キーインストール (RSA、AES、および ECC キー用) のためのハイレベルの API を提供します。SCE は専用ハードウェアブロックであり、SCE が提供する機能はサポートされている MCU によって異なります。

SCE HAL モジュールによって構成される暗号モジュールを使用すると、以下の暗号プリミティブを備えたセキュリティ用の暗号プロトコルを作成できます。

- 乱数生成
- AES または Triple DES (3DES)、ARC4 アルゴリズムを使用したデータ暗号化および復号化
- ECC、RSA、または DSA アルゴリズムを使用した署名の生成と検証
- HASH アルゴリズム MD5、SHA1、SHA224、または SHA256 を使用したメッセージダイジェストの計算
- キー生成: AES ラップキー、RSA プレーンテキストおよびラップキー、ECC プレーンテキストおよびラップキー
- 暗号化されたユーザーキーの Synergy プラットフォームへのインストール

SCE インタフェースは、SCE HAL モジュール用の共通 API を提供します。SCE インタフェースは、選択されたモジュール (AES、ARC4、RSA、DSA、HASH、TDDES、TRNG) に応じて、複数の動作をサポートします。

SSP 暗号ライブラリは、SCE7 の以下のセキュリティ機能にハイレベルの C コール可能 API を提供しません。

- 真性乱数生成器 (TRNG)
- セキュアな 128 ビット乱数を暗号で生成
- 決定性乱数生成器 (NIST SP800-90A DRBG など) へのシード生成
- ハッシュ関数
- データのデジタルフィンガープリントを提供するハッシュ値を生成
- SHA1/SHA224/SHA256 をサポート
- 対称鍵暗号化に使用される暗号化メカニズム: 暗号化/復号キーは送信者と受信者で共有

Advanced Encryption Standard (AES)

128 ビット、192 ビット、256 ビットのキーの長さをサポート

さまざまなチェーンモードをサポート: ECB、CBC、CTR、GCM、XTS

Data Encryption Standard (DES/3DES)

192 ビットのキーの長さ、固定 8 バイトブロックデータで動作

ECB と CBC チェーンモードをサポート

レガシーの Secure Socket Layer (SSL) および Transport Layer Security (TLS) プロトコルで使用

3DES は DES を各ブロックに 3 回適用

Alleged RC4 (ARC4)

2,048 ビットキーの長さをサポート

TLS と Wired Equivalent Privacy (WEP) で使用

128 ビットデータのスループロット

非対称鍵暗号化に使用される暗号化メカニズム: 公開鍵は送信者と受信者の間で交換され、公開鍵と秘密鍵を使って送信者と受信者で共通の秘密鍵を計算

Rivest, Shamir, and Adleman (RSA)

公開鍵の暗号化に使用

2 つの鍵を生成: 公開鍵と秘密鍵

送信者は公開鍵を使って暗号化

受信者は秘密鍵を使って復号

最大 2048 ビットのキーの長さをサポート

認証のデジタル検証、署名の生成と検証、鍵交換の暗号化/復号、ラッピング、その他セキュリティ機能に使用

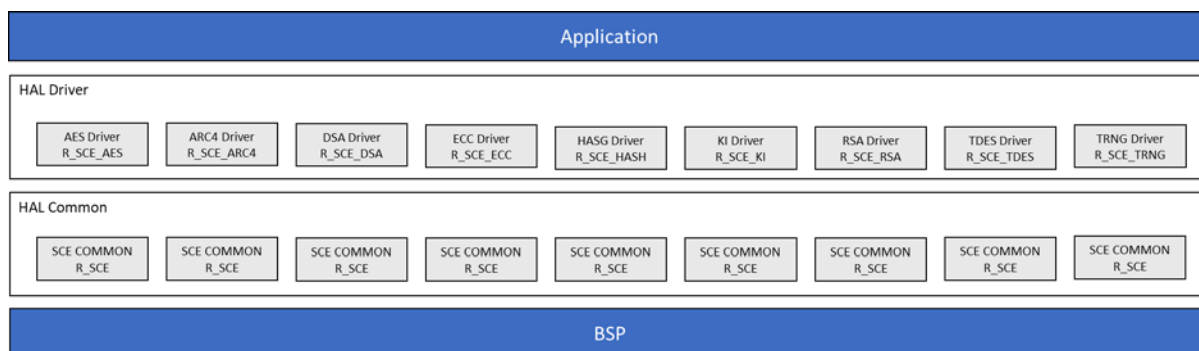


図 14.39 SCE HAL モジュールのブロック図

注：上述した図には、9 つの利用可能な暗号化モジュールがすべて示されています。SCE 共通モジュールは暗号化モジュールごとに繰り返されています。これは、暗号化モジュールがスレッドスタックに追加される時に共通モジュールが含まれるためです。共通モジュールは、複数の Synergy スタックにあるその他の複数のモジュールインスタンスから参照できます。

表 14.1 Synergy MCU でサポートされる SCE ドライバー

	S7G2、S5D9、S5D5	S3A1、S3A3、S3A7、S3A6	S1JA、S124、S128
TRNG がサポートする機能	乱数の生成とリード	乱数の生成とリード	乱数の生成とリード
AES	サポートする機能	暗号化、復号、キー生成、ラップキー	暗号化、復号
	キーのサイズ	128/192/256 ビット	128/256 ビット
	キーの種類	平文/レアキー、ラップキー	平文/レアキー
	チェーンモード	ECB、CBC、CTR、GCM、XTS ^{注1}	ECB、CBC、CTR
ARC4	暗号化、復号	N/A	N/A
DSA	サポートする機能	署名生成、署名検証	N/A
	キーのサイズ	(1024、128) ビット、 (2048、224) ビット、 (2048、256) ビット	N/A
ECC	サポートする機能	キー生成（平文/ラップキー）、スカラ値の乗算、ECDSA：署名生成、ECDSA：署名検証	N/A
	キーのサイズ	P-192、P-256	N/A
	キーの種類	平文/レアキー、ラップキー	N/A
HASH の暗号化方法	SHA1/SHA224/SHA256、MD5	N/A	N/A
キーインストール	AES、ECC、RSA キー	AES キー	N/A
TDES	サポートする機能	暗号化、復号	N/A
	キーのサイズ	192 ビット	N/A
	チェーンモード	ECB、CBC、CTR	N/A
RSA	サポートする機能	署名生成、署名検証、公開鍵および秘密鍵の暗号化、キー生成（平文およびラップキー）	N/A
	キーのサイズ	1024/2048 ビット	N/A
	キーの種類	平文/レアキー、ラップキー	N/A

注 1：このリリースで利用可能な暗号化関数 (sf_crypto) のフレームワークインタフェースには、HASH、TRNG、キー生成 (RSA プレーンテキストおよびラップキー、AES ラップキー、ECC ラップキー)、Cipher (RSA および AES 暗号化/復号化)、署名 (RSA 署名生成および検証)、キーインストール (AES、RSA、および ECC キー) などがあります。

注 2：XTS は、128 ビットおよび 256 ビットキーに対してのみサポートされます。

14.39 シリアルコミュニケーションインタフェース I²C (SCI_I2C)

SCI_I2C ドライバーは、MCU の SCI 周辺機能をサポートします。このドライバーは、マスタモード通信用の I²C プロトコルをサポートし、以下の機能を提供します。

- 7ビットまたは10ビットアドレスを使用したスレーブデバイスからのリード
- 7ビットまたは10ビットアドレスを使用したスレーブデバイスへのライト
- I²Cのリセット
- I²C トランザクション前の通信スレーブアドレスの動的な設定
- 割り込み駆動型の送信/受信処理
- 最大ビットレート 400 kbps の I²C ファストモードをサポート
- イベントコードを返すことのできるコールバック関数サポート
- すべての SCI モジュールへのビットレート変調機能の搭載

コールバック関数は以下のイベントとともに呼び出されます。

- 転送中断
- 転送完了
- 受信完了

コールバック構造体は送信または受信したバイト数が設定されます。SCI の SCI_I²C ドライバーは SSP の I²C インタフェースを実装しています。

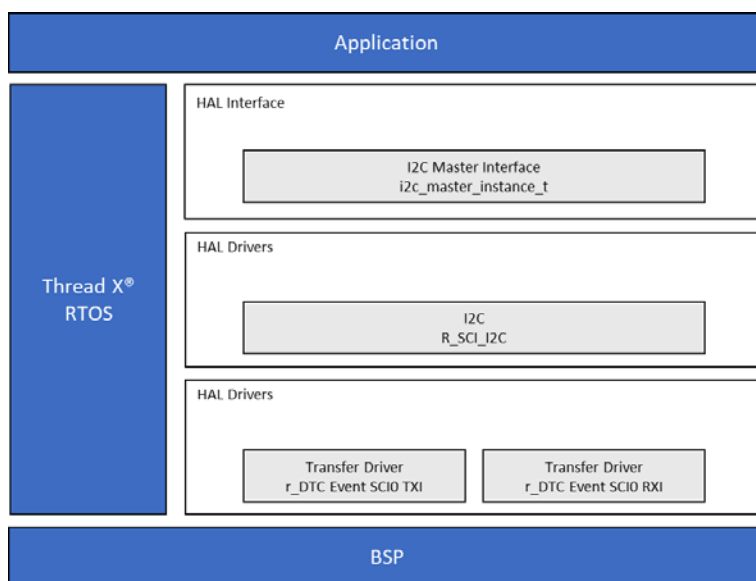


図 14.40 シリアルコミュニケーションインタフェース I²C (SCI_I2C) HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の SCI_I2C の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	マスターモード	スレーブモード	すべての割り込み要因のサポート	デジタルノイズフィルタ	ビットレート変調	SDA 遅延
S124	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S128	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S1JA	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S3A1	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S3A3	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S3A6	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S3A7	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S5D5	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S5D9	✓	N/A	ERI 割り込みは対象外	☒	✓	✓
S7G2	✓	N/A	ERI 割り込みは対象外	☒	✓	✓

グループ名	タイムアウト検出機能
S124	N/A
S128	N/A
S1JA	N/A
S3A1	N/A
S3A3	N/A
S3A6	N/A
S3A7	N/A
S5D5	N/A
S5D9	N/A
S7G2	N/A

14.40 シリアルコミュニケーションインタフェース SPI (SCI_SPI)

SCI SPI HAL モジュールは、マスターベースの SPI シリアル通信用のハイレベルの API を提供し、r_sci_spi (簡易 SPI) モジュールに実装されています。SCI SPI HAL モジュールは、SCI (シリアル通信インタフェース) 周辺機能を使用して MCU の SPI 機能を構成し、制御します。モジュールは、統合ソリューション開発環境内で構成されます。ユーザー定義のコールバックを作成し、SPI がデータの送信、データ転送の中断、またはエラー条件の検出を行ったときに信号を送信できます。

SCI SPI HAL モジュールは、MCU の DTC HAL モジュールを組み込むことにより、データ転送サポートありで有効化され、CPU の介入なく DTC を介して SCI SPI 転送を実行します。

- ドライバーの初期化
- 8ビットデータ転送を使用した SPI 動作によるシリアル通信
- 構成可能な4つのクロック位相とクロック極性の設定
- コールバックのサポート。コールバック関数は、次のイベントで呼び出されます。
 - 転送中断
 - 転送完了
 - オーバーランエラー
- マスタモードでの SPI 通信

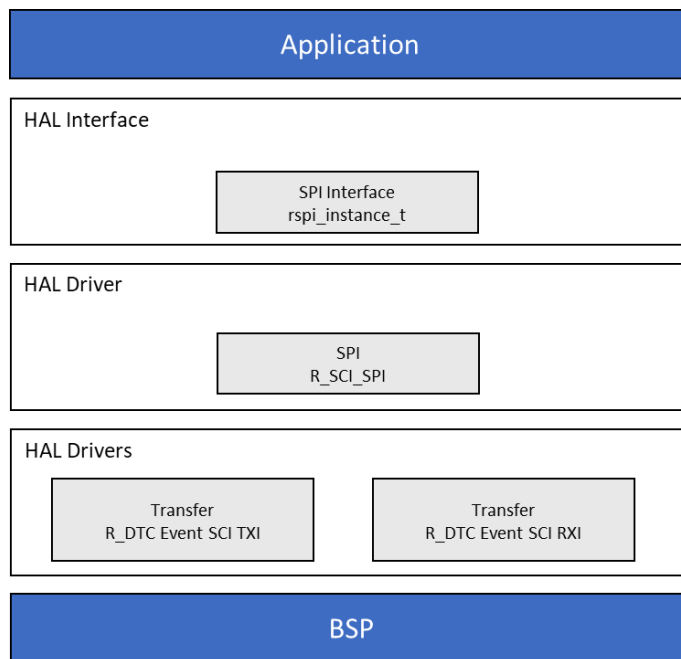


図 14.41 シリアルコミュニケーションインタフェース SPI (SCI_SPI) HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の SCI_SPI の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	Master Out/Slave In (MOSI)	Master In/Slave Out (MISO)	SSL スレーブ	SPI 動作 (4 線式)	クロック同期式動作 (3 線式)	全二重通信または送信のみを選択
S124	✓	✓	GPIO	N/A	✓	N/A
S128	✓	✓	GPIO	N/A	✓	N/A
S1JA	✓	✓	GPIO	N/A	✓	N/A
S3A1	✓	✓	GPIO	N/A	✓	N/A
S3A3	✓	✓	GPIO	N/A	✓	N/A
S3A6	✓	✓	GPIO	N/A	✓	N/A
S3A7	✓	✓	GPIO	N/A	✓	N/A
S5D5	✓	✓	GPIO	N/A	✓	N/A
S5D9	✓	✓	GPIO	N/A	✓	N/A
S7G2	✓	✓	GPIO	N/A	✓	N/A

グループ名	オーバーランエラー検出	データフォーマット	マスタモード/スレーブモードの選択	クロックソース (内部/外部)	設定可能な極性/位相	割り込み要因のサポート
S124	✓	8ビット	マスタ	内部	✓	✓
S128	✓	8ビット	マスタ	内部	✓	✓
S1JA	✓	8ビット	マスタ	内部	✓	✓
S3A1	✓	8ビット	マスタ	内部	✓	✓
S3A3	✓	8ビット	マスタ	内部	✓	✓
S3A6	✓	8ビット	マスタ	内部	✓	✓
S3A7	✓	8ビット	マスタ	内部	✓	✓
S5D5	✓	8ビット	マスタ	内部	✓	✓
S5D9	✓	8ビット	マスタ	内部	✓	✓
S7G2	✓	8ビット	マスタ	内部	✓	✓

14.41 シリアルコミュニケーションインタフェース UART (SCI_UART)

UART HAL モジュールは、UART アプリケーションのためのハイレベルの API を提供し、`r_sci_uart` に実装されています。UART HAL モジュールは、MCU 上の SCI 周辺機能を使用します。ユーザー定義のコールバックを作成し、必要に応じてハードウェアハンドシェイクとデータ操作を管理できます。

UART HAL モジュールは、標準の UART プロトコルをサポートします。UART モード (SCI 上の UART) の SCI 周辺機能とともに使用される UART HAL モジュールは、(標準的な UART プロトコルに加えて) 次の機能をサポートします。

- 全二重 UART 通信
- 複数のチャネルを使用した同時通信
- 割り込み駆動型のデータ送信と受信
- イベントコードを引数として渡すユーザーコールバック関数の呼び出し
- 実行時のボーレートの変更
- UART トランザクション中のハードウェアリソースのロック
- CTS/RTS ハードウェアフロー制御 (関連付けられた I/O ポート端子を使用し、ユーザー定義コールバック関数でサポート)
- DTC モジュールとの統合
- 進行中のリード/ライト動作の中止
- 受信のみ中止
- 送信のみ中止
- 送信と受信の両方の中止

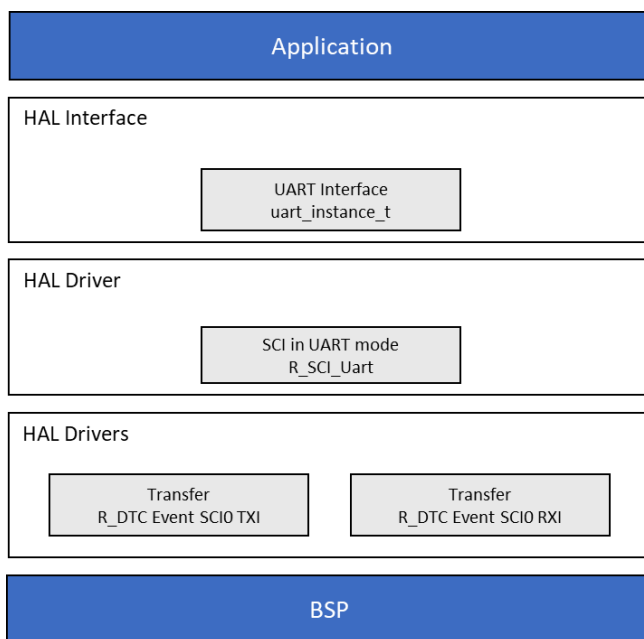


図 14.42 シリアルコミュニケーションインタフェース UART (SCI_UART) HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の SCI_UART の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	調歩同期式モード	クロック同期式モード	スマートカードインタフェースモード	簡易 I ² C モード	簡易 SPI モード
S124	✓	✓	☒	✓	✓
S128	✓	✓	☒	✓	✓
S1JA	✓	✓	☒	✓	✓
S3A1	✓	✓	☒	✓	✓
S3A3	✓	✓	☒	✓	✓
S3A6	✓	✓	☒	✓	✓
S3A7	✓	✓	☒	✓	✓
S5D5	✓	✓	☒	✓	✓
S5D9	✓	✓	☒	✓	✓
S7G2	✓	✓	☒	✓	✓

グループ名	転送速度の選択	データ長 (7ビット、8ビット、9ビット)	すべての割り込み要因のサポート	送信ストップビット (1ビット、2ビット)	偶数パリティ、奇数パリティ、パリティなし	受信エラー検出
S124	✓	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	ハードウェアフロー制御	送受信	アドレスマッチ	アドレスミスマッチ	スタートビットの検出	ブ레이크検出
S124	✓	レジスタ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S128	✓	レジスタ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S1JA	✓	レジスタ	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S3A1	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S3A3	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S3A6	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S3A7	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S5D5	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S5D9	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>
S7G2	✓	FIFO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	<input checked="" type="checkbox"/>

グループ名	クロックソース (内部/外部)	倍速モード	マルチプロセスサ通信	ノイズ除去	ビットレート変調機能	IrDA サポート
S124	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S128	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S1JA	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S3A1	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S3A3	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S3A6	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S3A7	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S5D5	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S5D9	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>
S7G2	✓	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	✓	✓	<input checked="" type="checkbox"/>

14.42 シリアルペリフェラルインタフェース (RSPI)

SPI HAL モジュールは、SPI プロトコルを使用した通信用の汎用 API です。このモジュールは、Synergy MCU で利用可能な SPI 周辺機能と SCI 周辺機能の両方をサポートし、`r_rsapi` と `r_sci_spi` に実装されています。このセクションでは、`r_rsapi` HAL モジュールについて説明します。このモジュールは SPI モジュールとも呼ばれます（以前は RSPI とも呼ばれていました）。SPI HAL モジュールは、標準 SPI マスタおよびスレーブモード通信機能をサポートします。転送イベントに対してコールバックが提供されています。

SPI HAL モジュールは、MCU のデータトランスファコントローラ (DTC) モジュールを組み込むことにより、データ転送サポートありで有効化されます。このモジュールは、CPU による各フレームに対する割り込み処理を必要とせず、DTC を介して SPI 転送を実行します。

SPI HAL モジュールは、以下の主要な機能をサポートします。

- ドライバーの初期化
- SPI 転送機能
 - 4 ワイヤ方式を使用する SPI 動作によるシリアル通信が可能
 - マスタモードとスレーブモードでシリアル通信が可能
 - シリアル転送クロックの極性の切り替え
 - シリアル転送クロックの位相の切り替え
- データ形式
 - MSB ファーストと LSB ファーストの選択が可能
 - 転送ビット長を 8、16、32 ビットから選択可能
- エラー検出
 - モード障害の検出
 - オーバーランエラーの検出
 - パリティエラーの検出
- SSL 制御機能
 - マスタモードでは、チャンネルごとに最大 4 つの SSL 信号 (SSLn0 から SSLn3) を内部的に選択できます
 - 外部ハードウェアスレーブ選択をマスタモードで使用できます
- 割り込み
 - RSPI 受信割り込み (受信バッファフル)
 - RSPI 送信割り込み (送信バッファエンブティ)
 - RSPI エラー割り込み (モード障害エラー、オーバーランエラー、パリティエラー)
- 遅延
 - SPI クロック遅延の追加
 - スレーブ選択ネゲート遅延の追加
 - 次アクセス遅延の追加

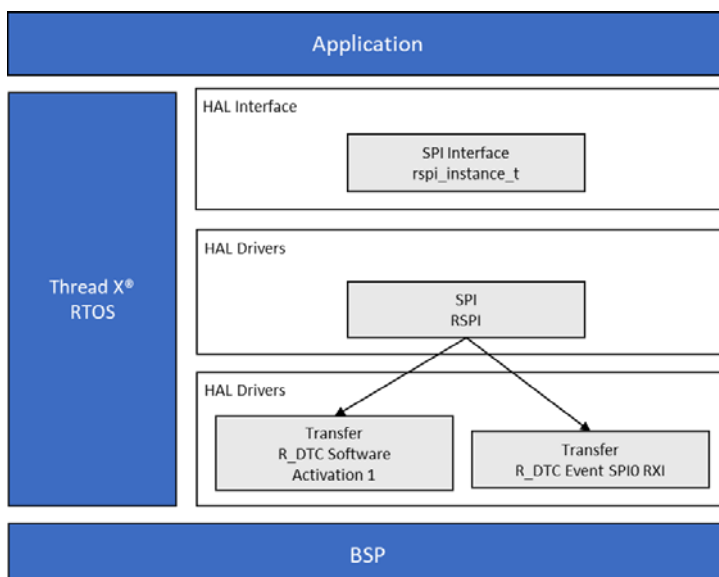


図 14.43 RSPI HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の RSPI の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	SPI 動作モード	クロック同期モード	全二重または送信のみの通信モードを選択可能	極性/位相の切り替え	マスターモード、スレーブモード	MSB ファースト/LSB ファーストの選択可能
S124	✓	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	転送ビット長 (8/16/32)	1ラウンドの送受信で最大4フレームを転送可能	ビットレート構成	ダブルバッファ構成	エラー検出	SSL制御機能
S124	✓	☒	✓	☒	✓	✓
S128	✓	☒	✓	☒	✓	✓
S1JA	✓	☒	✓	☒	✓	✓
S3A1	✓	☒	✓	☒	✓	✓
S3A3	✓	☒	✓	☒	✓	✓
S3A6	✓	☒	✓	☒	✓	✓
S3A7	✓	☒	✓	☒	✓	✓
S5D5	✓	☒	✓	☒	✓	✓
S5D9	✓	☒	✓	☒	✓	✓
S7G2	✓	☒	✓	☒	✓	✓

グループ名	最大8コマンドの転送	すべての割り込み要因	DTCサポート	ELC HAL ドライバによるイベントリンク	CMOS出力とオープンドレイン出力の切り替え機能	ループバックモード
S124	☒	✓	✓ (16ビット)	☒	✓	☒
S128	☒	✓	✓ (16ビット)	☒	✓	☒
S1JA	☒	✓	✓ (16ビット)	☒	✓	☒
S3A1	☒	✓	✓ (32ビット)	☒	✓	☒
S3A3	☒	✓	✓ (32ビット)	☒	✓	☒
S3A6	☒	✓	✓ (32ビット)	☒	✓	☒
S3A7	☒	✓	✓ (32ビット)	☒	✓	☒
S5D5	☒	✓	✓ (32ビット)	☒	✓	☒
S5D9	☒	✓	✓ (32ビット)	☒	✓	☒
S7G2	☒	✓	✓ (32ビット)	☒	✓	☒

グループ名	モジュールストップ機能	マルチマスタモード
S124	☒	☒
S128	☒	☒
S1JA	☒	☒
S3A1	☒	☒
S3A3	☒	☒
S3A6	☒	☒
S3A7	☒	☒
S5D5	☒	☒
S5D9	☒	☒
S7G2	☒	☒

14.43 セグメント LCD コントローラ (SLCDC)

セグメント LCD コントローラ (SLCDC) HAL モジュールは、セグメント LCD アプリケーションのためのハイレベルの API を提供し、`r_slcdc` に実装されています。セグメント LCD コントローラ HAL モジュールは、セグメント LCD にデータを表示し、表示したデータを変更します。セグメント LCD コントローラ HAL モジュールは、MCU 上のセグメント LCD コントローラモジュールを使用します。

SLCDC HAL モジュールは、セグメント LCD コントローラ (SLCDC) を使用して、データをセグメント LCD に表示します。ドライバーは LCD を初期化してデータを表示し、ドライブ電圧ジェネレータ、ディスプレイ波形、タイムスライス数、および LCD を稼働するバイアスメソッドを構成します。このモジュールでは、指定されたセグメントのセットにデータを表示するための関数、既存のセグメントデータを更新するための関数、ディスプレイを有効化および無効化するための関数、表示領域を設定するための関数、およびコントラストを調整するための関数を提供します。

- LCD ドライバー電圧ジェネレータには、内部昇圧方式または外部抵抗分割方式を選択可能
- 表示バイアスは 1/2 バイアス、1/3 バイアス、1/4 バイアスから選択可能
- 表示の時分割、スタティック、2 時分割、3 時分割、4 時分割、8 時分割から選択可能
- 表示波形は A 波形または B 波形から選択可能
- 表示データ領域は A パターン、B パターン、および点滅から選択可能。表示データ領域は切り替え可能
- RTC 周期割り込み (PRD) を使用して、A パターンまたは B パターンの点滅表示を生成
- 電圧ブースト回路を稼働すると生成される参照電圧を 16 ステップ (コントラスト調整) で調整

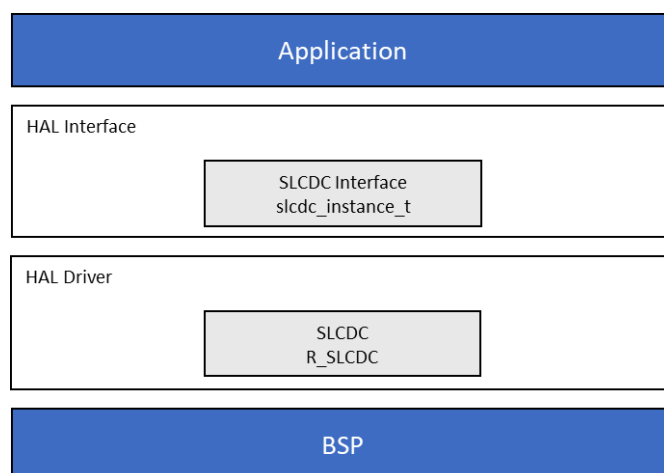


図 14.44 SLCDC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の SLCDC の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	表示波形（波形 A または波形 B）が選択可能	電圧生成回路：内部昇圧方式および外部抵抗分割方式	電圧生成回路：容量分割方式	LCD 点滅および表示機能	ソースクロックのサポート：メインクロック	ソースクロックのサポート：サブクロック
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	☒	✓	☒	☒
S3A3	✓	✓	☒	✓	☒	☒
S3A6	✓	✓	☒	✓	☒	☒
S3A7	✓	✓	☒	✓	☒	☒
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A	N/A	N/A

グループ名	表示波形（波形 A または波形 B）が選択可能	電圧生成回路：内部昇圧方式および外部抵抗分割方式	時分割モード：スタティック、4 時分割	時分割モード：1、2、3、8 時分割	表示バイアス：1/2、1/3、1/4 バイアス	コントラスト調整
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	✓	✓	✓	☒	✓	✓
S3A3	✓	✓	✓	☒	✓	✓
S3A6	✓	✓	✓	☒	✓	✓
S3A7	✓	✓	✓	☒	✓	✓
S5D5	N/A	N/A	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A	N/A	N/A

14.44 シリアルサウンドインタフェース (SSI)

I²S HAL ドライバモジュールは、汎用 I²S オーディオシリアル通信プロトコル用のハイレベルの API を提供します。マスタモードでの非圧縮オーディオデータの送受信に使用されます。

I²S マスタモードの SSI 周辺機能で使用する I²S ドライバは、標準的な I²S プロトコルに加えて、次の機能をサポートしています。

- 全二重通信 (SSI チャンネル 0 のみ可能)
- 割り込み駆動型のデータ送信と受信
- DTC 転送モジュールとの統合
- ユーザー定義のコールバックを作成し、追加データが必要な場合に対応可能

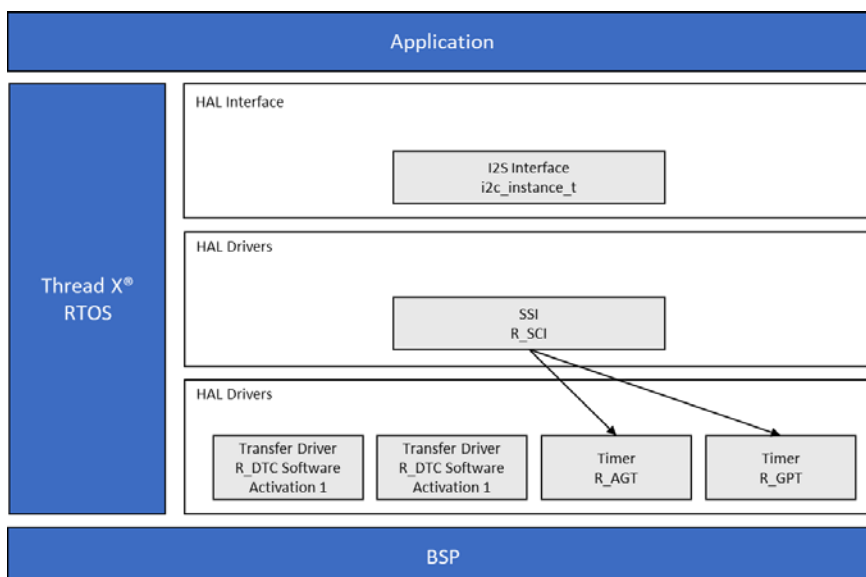


図 14.45 I²S HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の I²S の機能は以下のとおりです。

[記号の説明]

✓ : 使用可能 (テスト済み)

☒ : 使用不可 (未テスト/機能せず、またはその両方)

N/A : MCU のサポート対象外

グループ名	2 チャンネル	SSI フォーマット	MSB ファースト フォーマット	構成可能なシリアルビットクロック (16、32、48、および 64 のサンプリングレート)	オーディオ用マスタクロック端子からのマスタクロック入力 (AUDIO_CLK)	GPT 出力用マスタクロック端子からのマスタクロック入力 (GTIOC1A)
S124	N/A	N/A	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A	N/A	N/A
S1JA	N/A	N/A	N/A	N/A	N/A	N/A
S3A1	☒	☒	☒	✓	✓	☒
S3A3	☒	☒	☒	✓	✓	☒
S3A6	☒	☒	☒	✓	✓	☒
S3A7	☒	☒	☒	✓	✓	☒
S5D5	✓	☒	☒	✓	✓	☒
S5D9	✓	☒	☒	✓	✓	☒
S7G2	✓	☒	☒	✓	✓	☒

グループ名	ストップワード SSIWS を選択可能	通信エラーによる割り込みの受け付け	受信データフルによる割り込みの受け付け	送信データエンプティによる割り込みの受け付け	GTIOC1A との内 部接続
S124	☒	☒	☒	☒	N/A
S128	☒	☒	☒	☒	N/A
S1JA	N/A	N/A	N/A	N/A	N/A
S3A1	☒	✓	✓	✓	✓
S3A3	☒	✓	✓	✓	✓
S3A6	☒	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓
S5D5	☒	✓	✓	✓	✓
S5D9	☒	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓

14.45 シグマデルタ ADC (SDADC)

SDADC HAL モジュールでは、`r_sdadc` 上で A/D 変換用の ADC API を実装します。このモジュールは、MCU 上で利用可能な 24 ビット A/D コンバータ (SDADC24) 周辺機能をサポートしています。ユーザー定義のコールバックを作成して、新しいサンプルが利用可能になるたびにデータを処理することができます。

- 24 ビットシグマデルタ A/D コンバータ
- シングルスキャンまたは連続スキャン動作モード
- シングルエンド入力または差動入力
- 差動入力での最大 32 のゲイン
- 差動入力での構成可能なオーバーサンプリング比

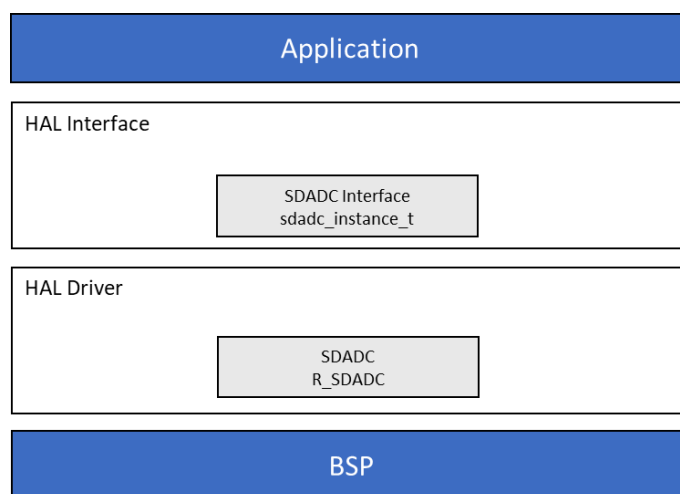


図 14.46 SDADC HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の SDADC の機能は以下のとおりです。

[記号の説明]

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	すべてのアナログチャネルのサポート	24 ビット	シングルスキャンモード	連続スキャンモード
S124	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A
S1JA	✓	✓	N/A	✓
S3A1	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A

グループ名	シングルエンド入力	作動入力	プログラマブルゲインアンプ	オーバーサンプリング比
S124	N/A	N/A	N/A	N/A
S128	N/A	N/A	N/A	N/A
S1JA	✓	✓	✓	✓
S3A1	N/A	N/A	N/A	N/A
S3A3	N/A	N/A	N/A	N/A
S3A6	N/A	N/A	N/A	N/A
S3A7	N/A	N/A	N/A	N/A
S5D5	N/A	N/A	N/A	N/A
S5D9	N/A	N/A	N/A	N/A
S7G2	N/A	N/A	N/A	N/A

14.46 ウォッチドッグタイマ (WDT)

WDT (ウォッチドッグタイマ) HAL モジュールは、WDT アプリケーションのためのハイレベルの API を提供し、`r_wdt` に実装されています。WDT HAL モジュールは、MCU 上の WDT 周辺機能を使用します。ユーザー定義のコールバックを作成し、イベント通知に応答できます。

- WDT がアンダーフローした場合、または許可されたリフレッシュウィンドウの外でリフレッシュされた場合には、次のいずれかのイベントが生じる可能性があります。
 - デバイスのリセット
 - NMI の生成
- ウォッチドッグタイマ (WDT) 周辺機能をサポート。WDT は外部クロックを使用
- WDT レジスタからレジスタスタートモードで設定可能
- リセット後の自動ハードウェア構成をサポート
- アプリケーションから開始可能

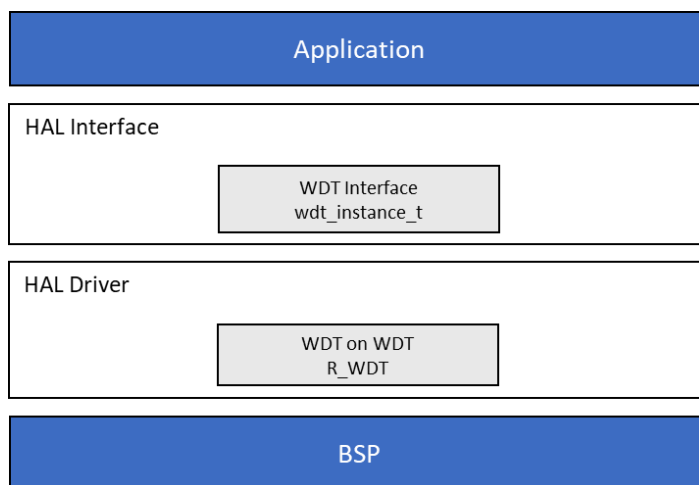


図 14.47 WDT HAL モジュールのブロック図

SSP でサポート対象およびサポート対象外の WDT の機能は以下のとおりです。

【記号の説明】

✓：使用可能（テスト済み）

☒：使用不可（未テスト/機能せず、またはその両方）

N/A：MCU のサポート対象外

グループ名	クロック分周比 (4、64、128、512、2048、8192)	カウントダウン	レジスタスタートモード	オートスタートモード	リセット出力	割り込み要求出力
S124	✓	✓	✓	✓	✓	✓
S128	✓	✓	✓	✓	✓	✓
S1JA	✓	✓	✓	✓	✓	✓
S3A1	✓	✓	✓	✓	✓	✓
S3A3	✓	✓	✓	✓	✓	✓
S3A6	✓	✓	✓	✓	✓	✓
S3A7	✓	✓	✓	✓	✓	✓
S5D5	✓	✓	✓	✓	✓	✓
S5D9	✓	✓	✓	✓	✓	✓
S7G2	✓	✓	✓	✓	✓	✓

グループ名	スリープモードカウント停止制御出力	ELC HAL ドライバーによるイベントリンク	ウィンドウ機能	カウンタ停止条件 (リセット、カウンタのアンダーフローまたはリフレッシュエラー発生)	リフレッシュ許可期間外でのリフレッシュ	カウンタ値の読み出し
S124	☒	☒	✓	✓	✓	✓
S128	☒	☒	✓	✓	✓	✓
S1JA	☒	☒	✓	✓	✓	✓
S3A1	☒	☒	✓	✓	✓	✓
S3A3	☒	☒	✓	✓	✓	✓
S3A6	☒	☒	✓	✓	✓	✓
S3A7	☒	☒	✓	✓	✓	✓
S5D5	☒	☒	✓	✓	✓	✓

グループ名	スリープモードカウント停止制御出力	ELC HAL ドライバーによるイベントリンク	ウィンドウ機能	カウンタ停止条件 (リセット、カウンタのアンダーフローまたはリフレッシュエラー発生)	リフレッシュ許可期間外でのリフレッシュ	カウンタ値の読み出し
S5D9	☒	☒	✓	✓	✓	✓
S7G2	☒	☒	✓	✓	✓	✓

15. ボードサポートパッケージ (BSP)

SSP には、DK-S7G2、SK-S7G2、PK-S5D9、DK-S3A7、DK-S128、DK-S124、PE-HMI1、TB-S5D5、TB-S3A6、TB-S3A3 の各ターゲットボード向けのボードサポートパッケージ (BSP) が含まれています。

BSP は、リセット後の MCU の制御をユーザーアプリケーション (main()関数など) 側に渡します。ユーザーアプリケーション側に制御を渡す前に、BSP はスタック、ヒープ、クロック、割り込み、C ランタイム環境をセットアップします。また、BSP はポート I/O 端子の設定およびセットアップをし、ボード固有の初期化も行います。SSP が提供する BSP の主な機能は以下のとおりです。

- 指定されているキットのサポート
- e² studio ISDE または IAR EW for Synergy を使用したカスタム BSP の作成
- スタート時のシステムの初期化と構成
- ソフトウェアとハードウェアのロック/ロック解除
- レジスタ保護
- CMSIS 対応
 - MCU 周辺機能の標準化された定義
 - MCU 機能にアクセスするための標準化されたアクセス関数
 - システム例外処理のための標準化された関数
 - システム初期化のための標準化された関数
 - クロック速度情報のための標準化されたソフトウェア変数

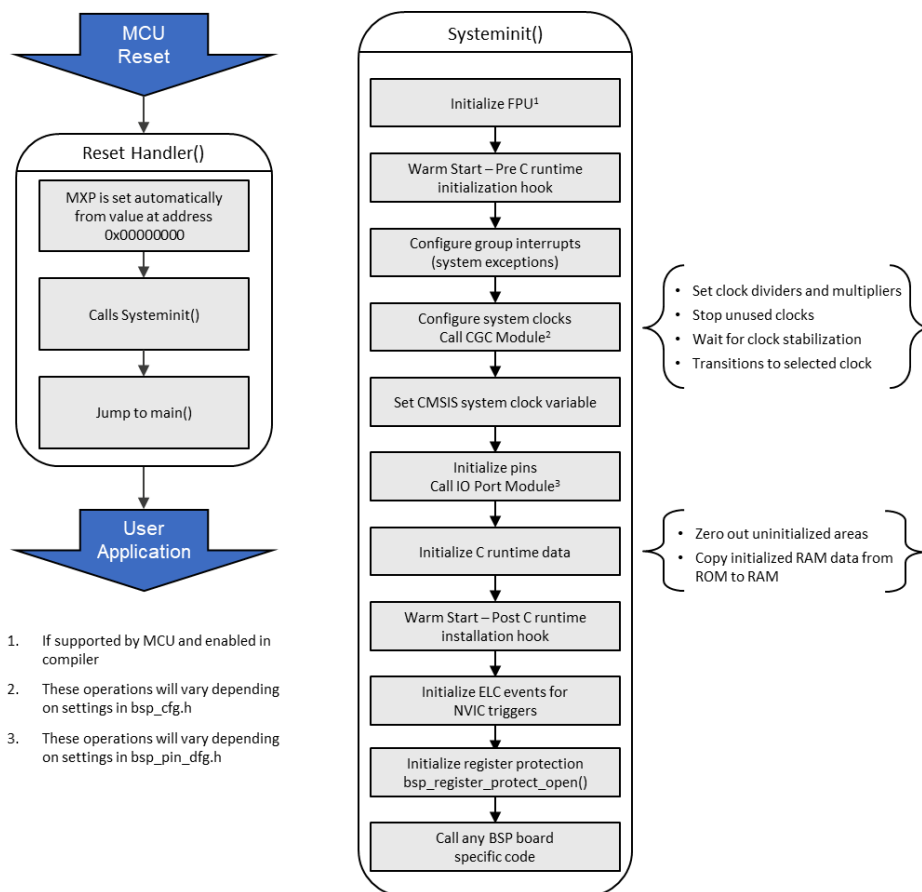


図 15.1 ボードサポートパッケージのイベントごとのワークフロー

16. 推定メモリサイズ

特定の設計に対して、必要なメモリサイズを推定することは容易ではありません。設計、コーディング、コンパイラのオプション、さらにはエラーチェックの方法によって、生成されるバイナリのサイズや実行時に必要なメモリサイズは異なります。以下の各セクションでは、開発時に、エンジニアが特定の機能を含める場合に必要なフラッシュメモリサイズを判断できるように、さまざまなコンフィギュレーションとオプションに関する情報を提供しています。

記載されている推定メモリサイズは、各モジュールの推定メモリ消費量を示しています。各モジュールのメモリ使用量には以下が含まれます。

- フラッシュメモリ使用量 = .text + .data

注：これらは推定メモリサイズです。ビルドオプションによっては、異なる結果になる可能性があります。これらの推定値の精度は +/-5% です。

ThreadX コンポーネントと X-Ware コンポーネントでは、コンパイラ時エラーチェックのオン/オフによって必要なメモリサイズは異なるため、本データシートでは両方の条件を記載しています。たとえば、ThreadX でエラーチェックをオフにした場合、必要なフラッシュメモリサイズ ("event" に対する) は以下のとおりです。

名称	フラッシュ
Event	1156 バイト

エラーチェックをオンにする場合は、上記のフラッシュメモリサイズ (単位：バイト) に以下のサイズを追加する必要があります。

名称	フラッシュ
Event	360 バイト

したがって、"event" ディレクトリのすべての機能が使用されている場合、この ThreadX 機能のために必要な合計フラッシュメモリサイズは 1516 バイトになります。

注：

- 本データシートに記載されているメモリサイズは、考えられるすべてのオプションには対応していません。
- 多様なコンパイラツールチェーンによってメモリサイズはさまざまに変化するため、Arm® Cortex M4 アーキテクチャでのコンパイルと、Arm® Cortex M0+でのコンパイルには差異が生じます。
- Arm® Cortex M4 とは、S3A3、S3A6、S3A7、S5D5、S5D9、S7G2 の各 Synergy MCU グループに含まれるデバイスを指します。
- Arm® Cortex M0+とは、S124 および S128 Synergy MCU グループに含まれるデバイスを指します。
- 弊社でフラッシュメモリサイズの計算に使用されているプロセスは以下のとおりです。ソースコードをコンパイルしてオブジェクトファイルを作成する。
以下のコマンドを使用してメモリフットプリントを生成する。

```
arm-none-eabi-size --format=Berkeley <filename>.o
```

以降の表には、GCC -O2 と IAR -OH (速度) の両方を使用した SSP v1.4.0 ビルドのフラッシュメモリサイズがバイト単位で示されています。メモリサイズはビルドオプションによって異なります。また、記載されている値は推定値であり、その値を保証するものではありません。このデータを 1 か所にまとめて記載することにより、エンジニアは任意の機能に対するフラッシュメモリ要件をすばやく特定し、各種機能の要件を集約して基礎となる Renesas Synergy プラットフォームに必要なサイズを簡単に見極めることができます。

16.1 GCC ROM の推定メモリサイズ

表 16.1 各種 MCU グループの GCC ROM 結果まとめ

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
_bsp	bsp_cache	-	8	8	112	112	112	112	112	112	112
_bsp	bsp_clocks	-	100	100	320	332	328	320	413	429	468
_bsp	bsp_common	-	92	92	84	84	84	84	84	84	84
_bsp	bsp_common_leds	-	24	24	32	32	32	32	32	32	32
_bsp	bsp_delay	-	100	100	100	100	100	100	100	100	100
_bsp	bsp_feature	-	248	276	308	308	308	316	272	272	272
_bsp	bsp_fmi_R7FS124*	-	1024	-	-	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS128*	-	-	1024	-	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A1*	-	-	-	1024	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A3*	-	-	-	-	1024	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A6*	-	-	-	-	-	1024	-	-	-	-
_bsp	bsp_fmi_R7FS3A7*	-	-	-	-	-	-	1024	-	-	-
_bsp	bsp_fmi_R7FS5D5*	-	-	-	-	-	-	-	1024	-	-
_bsp	bsp_fmi_R7FS5D9*	-	-	-	-	-	-	-	-	1024	-
_bsp	bsp_fmi_R7FS7G2*	-	-	-	-	-	-	-	-	-	1024
_bsp	bsp_group_irq	-	468	516	612	612	612	612	612	612	612
_bsp	bsp_hw_locks	-	220	232	308	344	336	336	432	432	432
_bsp	bsp_init	-	4	4	4	4	4	4	16	16	28
_bsp	bsp_irq	-	96	96	96	96	96	96	96	96	96
_bsp	bsp_leds	-	14	14	16	16	16	16	12	14	16
_bsp	bsp_locking	-	300	300	308	308	308	308	308	308	308
_bsp	bsp_module_stop	-	250	270	274	290	290	274	282	282	282
_bsp	bsp_qspi	-	-	-	1013	1013	-	1013	1009	1009	1021
_bsp	bsp_register_protection	-	188	188	204	204	204	204	204	204	204
_bsp	bsp_rom_registers	-	24	76	76	76	76	76	76	76	76
_bsp	bsp_sbrk	-	68	68	68	68	68	68	68	68	68
_bsp	bsp_sdram	-	-	-	-	-	-	-	336	336	336
adc	r_adc	-	3920	3920	3556	3556	3556	3552	3552	3552	3552
adc_periodic	sf_adc_periodic	-	1060	1060	1068	1068	1068	1068	1068	1068	1068

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
agt	r_agt	-	2286	2286	1982	1982	1982	1978	1978	1978	1978
analog	r_acmphs	-	-	-	-	-	-	-	-	-	-
analog	r_acmplp	-	-	-	-	-	-	-	-	-	-
analog	r_opamp	-	-	872	-	-	-	-	-	-	-
analog	r_sdadc	-	-	-	-	-	-	-	-	-	-
audio_playback	sf_audio_playback	-	3001	3001	2849	2849	2849	2849	2849	2849	2849
audio_playback_dac	sf_audio_playback_hw_dac	-	752	840	788	788	788	788	788	788	788
audio_playback_i2s	sf_audio_playback_hw_i2s	-	-	-	268	268	268	268	268	268	268
audio_record_adc	sf_audio_record_adc	-	436	-	-	388	388	388	388	388	388
ble	ble_rl78g1d_if	-	-	-	-	-	-	-	-	-	8693
ble	ble_rl78g1d_prf_anp_if	-	-	-	-	-	-	-	-	-	1095
ble	ble_rl78g1d_prf_blp_if	-	-	-	-	-	-	-	-	-	1363
ble	ble_rl78g1d_prf_fmp_if	-	-	-	-	-	-	-	-	-	525
ble	ble_rl78g1d_prf_hid_if	-	-	-	-	-	-	-	-	-	2044
ble	ble_rl78g1d_prf_hrp_if	-	-	-	-	-	-	-	-	-	1283
ble	ble_rl78g1d_prf_hrp_if	-	-	-	-	-	-	-	-	-	1350
ble	ble_rl78g1d_prf_if	-	-	-	-	-	-	-	-	-	1912
ble	ble_rl78g1d_prf_paps_if	-	-	-	-	-	-	-	-	-	1187
ble	ble_rl78g1d_prf_pxp_if	-	-	-	-	-	-	-	-	-	899
ble	ble_rl78g1d_prf_scps_if	-	-	-	-	-	-	-	-	-	668
ble	ble_rl78g1d_prf_tip_if	-	-	-	-	-	-	-	-	-	1228
ble	ble_serial	-	-	-	-	-	-	-	-	-	1407
ble	bmi	-	-	-	-	-	-	-	-	-	1505
ble	rble_api_anpc	-	-	-	-	-	-	-	-	-	552
ble	rble_api_anps	-	-	-	-	-	-	-	-	-	336
ble	rble_api_blpc	-	-	-	-	-	-	-	-	-	788
ble	rble_api_blps	-	-	-	-	-	-	-	-	-	328
ble	rble_api_cppc	-	-	-	-	-	-	-	-	-	1044
ble	rble_api_cpps	-	-	-	-	-	-	-	-	-	968
ble	rble_api_cscpc	-	-	-	-	-	-	-	-	-	824
ble	rble_api_cscps	-	-	-	-	-	-	-	-	-	312
ble	rble_api_fmpl	-	-	-	-	-	-	-	-	-	256

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
ble	rble_api_fmpt	-	-	-	-	-	-	-	-	-	132
ble	rble_api_gap	-	-	-	-	-	-	-	-	-	1968
ble	rble_api_gatt	-	-	-	-	-	-	-	-	-	1104
ble	rble_api_glpc	-	-	-	-	-	-	-	-	-	992
ble	rble_api_glps	-	-	-	-	-	-	-	-	-	556
ble	rble_api_hgbh	-	-	-	-	-	-	-	-	-	1408
ble	rble_api_hghd	-	-	-	-	-	-	-	-	-	504
ble	rble_api_hgrh	-	-	-	-	-	-	-	-	-	1560
ble	rble_api_hrhc	-	-	-	-	-	-	-	-	-	708
ble	rble_api_hrps	-	-	-	-	-	-	-	-	-	228
ble	rble_api_hrpc	-	-	-	-	-	-	-	-	-	924
ble	rble_api_hrps	-	-	-	-	-	-	-	-	-	364
ble	rble_api_hrpc	-	-	-	-	-	-	-	-	-	1112
ble	rble_api_hrpc	-	-	-	-	-	-	-	-	-	864
ble	rble_api_paspc	-	-	-	-	-	-	-	-	-	464
ble	rble_api_pasps	-	-	-	-	-	-	-	-	-	292
ble	rble_api_pxpm	-	-	-	-	-	-	-	-	-	468
ble	rble_api_pxpr	-	-	-	-	-	-	-	-	-	144
ble	rble_api_rscpc	-	-	-	-	-	-	-	-	-	928
ble	rble_api_rscps	-	-	-	-	-	-	-	-	-	384
ble	rble_api_sm	-	-	-	-	-	-	-	-	-	652
ble	rble_api_sppc	-	-	-	-	-	-	-	-	-	396
ble	rble_api_spps	-	-	-	-	-	-	-	-	-	220
ble	rble_api_tipc	-	-	-	-	-	-	-	-	-	644
ble	rble_api_tips	-	-	-	-	-	-	-	-	-	480
ble	rble_api_vs	-	-	-	-	-	-	-	-	-	992
ble	rble_host	-	-	-	-	-	-	-	-	-	1464
ble	rble_if_api_cb	-	-	-	-	-	-	-	-	-	31334
ble	rscip	-	-	-	-	-	-	-	-	-	320
ble	rscip_cntl	-	-	-	-	-	-	-	-	-	3436
ble	rscip_uart	-	-	-	-	-	-	-	-	-	1447
ble	sf_ble_rl78g1d	-	-	-	-	-	-	-	-	-	2610

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
ble	sf_ble_rl78g1d_onboard_profile	-	-	-	-	-	-	-	-	-	960
ble	timer	-	-	-	-	-	-	-	-	-	348
block_media_sdmmc	sf_block_media_qspi	-	-	-	-	844	-	844	844	844	844
block_media_sdmmc	sf_block_media_ram	-	-	-	-	364	-	364	-	364	364
block_media_sdmmc	sf_block_media_sdmmc	-	-	-	515	515	-	515	515	515	515
cac	r_cac	-	2272	2272	1912	1912	1912	1908	1908	1908	1908
can	hw_can	-	1880	1880	1620	1620	1620	1620	1620	1620	1620
can	r_can	-	2892	2892	2464	2464	2464	2464	2464	2464	2464
cellular	cellular_serial	-	-	-	-	-	813	813	813	813	813
cellular	sf_cellular_cat1	-	-	-	-	-	4772	4772	4772	4772	4772
cellular	sf_cellular_cat1_socket	-	-	-	-	-	636	636	636	636	636
cellular	sf_cellular_cat1_socket_private	-	-	-	-	-	4951	4951	4951	4951	4951
cellular	sf_cellular_cat3	-	-	-	-	-	1892	1892	1892	1892	1892
cellular	sf_cellular_cat3_socket	-	-	-	-	-	612	612	612	612	612
cellular	sf_cellular_cat3_socket_private	-	-	-	-	-	4888	4888	4888	4888	4888
cellular	sf_cellular_common	-	-	-	-	-	1648	1648	1648	1648	1648
cellular	sf_cellular_common_private	-	-	-	-	-	9395	9395	9395	9395	9395
cellular	sf_cellular_qctlicatm1	-	-	-	-	-	-	-	-	2800	2800
cellular	sf_cellular_qctlicatm1_socket	-	-	-	-	-	-	-	-	696	696
cellular	sf_cellular_qctlicatm1_socket_private	-	-	-	-	-	-	-	-	5165	5165
cgc	hw_cgc	-	2942	2942	3002	3002	3002	3002	3002	3002	3002
cgc	r_cgc	-	3410	3410	3170	3170	3170	3170	3170	3170	3170
console	sf_console	-	-	-	-	-	-	-	-	-	2283
ctsu	hw_ctsu_common	-	588	-	-	596	596	596	-	596	-
ctsu	sf_touch_ctsu	-	4312	-	-	4212	4212	4212	-	4212	-
ctsu	sf_touch_ctsu_button	-	1888	-	-	1964	1964	1964	-	1964	-
ctsu	sf_touch_ctsu_slider	-	-	-	-	2840	-	2840	-	-	-
ctsu	sf_touch_panel_i2c	-	-	-	-	-	-	-	-	1063	1063
ctsu	touch_panel_i2c_ft5x06	-	-	-	-	-	-	-	-	295	295

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
ctsu	touch_panel_i2c_sx8654	-	-	-	-	-	-	-	-	604	604
ctsu	r_ctsu	-	13889	-	-	13961	13961	13957	-	13957	-
dac	r_dac	-	644	644	592	592	592	592	592	592	592
dac8	r_dac8	-	-	816	-	724	-	-	-	-	-
dave/2d	dave_64bitoperation	-	-	-	-	-	-	-	-	0	0
dave/2d	dave_base	-	-	-	-	-	-	-	-	21	21
dave/2d	dave_blit	-	-	-	-	-	-	-	-	1272	1272
dave/2d	dave_box	-	-	-	-	-	-	-	-	3516	3516
dave/2d	dave_circle	-	-	-	-	-	-	-	-	1312	1312
dave/2d	dave_context	-	-	-	-	-	-	-	-	2992	2992
dave/2d	dave_curve	-	-	-	-	-	-	-	-	1104	1104
dave/2d	dave_dlist	-	-	-	-	-	-	-	-	3840	3840
dave/2d	dave_driver	-	-	-	-	-	-	-	-	2718	2718
dave/2d	dave_edge	-	-	-	-	-	-	-	-	1648	1648
dave/2d	dave_errorcodes	-	-	-	-	-	-	-	-	1184	1184
dave/2d	dave_gradient	-	-	-	-	-	-	-	-	2132	2132
dave/2d	dave_hardware	-	-	-	-	-	-	-	-	300	300
dave/2d	dave_line	-	-	-	-	-	-	-	-	8744	8744
dave/2d	dave_math	-	-	-	-	-	-	-	-	264	264
dave/2d	dave_memory	-	-	-	-	-	-	-	-	224	224
dave/2d	dave_pattern	-	-	-	-	-	-	-	-	464	464
dave/2d	dave_perfcoun	-	-	-	-	-	-	-	-	200	200
dave/2d	dave_polyline	-	-	-	-	-	-	-	-	2116	2116
dave/2d	dave_quad	-	-	-	-	-	-	-	-	3260	3260
dave/2d	dave_rbuffer	-	-	-	-	-	-	-	-	1796	1796
dave/2d	dave_render	-	-	-	-	-	-	-	-	4304	4304
dave/2d	dave_texture	-	-	-	-	-	-	-	-	3700	3700
dave/2d	dave_triangle	-	-	-	-	-	-	-	-	2496	2496
dave/2d	dave_utility	-	-	-	-	-	-	-	-	1072	1072
dave/2d	dave_viewport	-	-	-	-	-	-	-	-	600	600
dave/2d	dave_wedge	-	-	-	-	-	-	-	-	1196	1196
dave/2d	sf_tes_2d_drw_base	-	-	-	-	-	-	-	-	275	275

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
dave/2d	sf_tes_2d_drw_irq	-	-	-	-	-	-	-	-	320	320
dave/2d	sf_tes_2d_drw_memory	-	-	-	-	-	-	-	-	229	229
dmac	r_dmac	-	-	-	1924	1924	1924	1936	1936	1936	1936
doc	r_doc	-	984	984	808	808	808	804	804	804	804
dtc	r_dtc	-	1821	1821	1669	1669	1669	1685	1685	1685	1685
el_fx	sf_el_fx	-	-	-	536	536	-	536	536	536	536
el_gx	sf_el_gx	-	-	-	-	-	-	-	-	1458	1458
el_nx	nx_hw_init	-	-	-	-	-	-	-	920	920	920
el_nx	nx_renesas_synergy	-	-	-	-	-	-	-	2376	2376	2376
el_nx	sf_el_nx_comms	-	-	-	-	-	-	-	1534	1534	1534
el_ux	sf_el_ux_comms	-	918	918	878	878	878	878	878	878	878
el_ux	usb_irq_veneer	-	28	28	32	32	32	32	32	96	96
el_ux	usbfs_irq_veneer	-	60	60	68	68	68	68	68	68	136
el_ux	ux_dcd_synergy_buffer_empty_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_dcd_synergy_buffer_notready_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_dcd_synergy_buffer_read	-	80	80	76	76	76	76	76	76	76
el_ux	ux_dcd_synergy_buffer_ready_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_dcd_synergy_buffer_write	-	116	116	112	112	112	112	112	112	112
el_ux	ux_dcd_synergy_current_endpoint_change	-	140	140	140	140	140	140	140	140	140
el_ux	ux_dcd_synergy_data_buffersize	-	64	64	60	60	60	60	60	60	60
el_ux	ux_dcd_synergy_endpoint_create	-	424	424	424	424	424	424	424	424	424
el_ux	ux_dcd_synergy_endpoint_destroy	-	152	152	152	152	152	152	152	152	152
el_ux	ux_dcd_synergy_endpoint_nak_set	-	32	32	24	24	24	24	24	24	24
el_ux	ux_dcd_synergy_endpoint_reset	-	104	104	100	100	100	100	100	100	100
el_ux	ux_dcd_synergy_endpoint_stall	-	52	52	48	48	48	48	48	48	48
el_ux	ux_dcd_synergy_endpoint_status	-	28	28	20	20	20	20	20	20	20
el_ux	ux_dcd_synergy_fifo_port_change	-	60	60	60	60	60	60	60	60	60
el_ux	ux_dcd_synergy_fifo_read	-	800	800	672	672	672	672	672	1096	1096
el_ux	ux_dcd_synergy_fifo_write	-	432	432	344	344	344	344	344	552	552
el_ux	ux_dcd_synergy_fifo_write	-	800	800	700	700	700	700	700	828	828
el_ux	ux_dcd_synergy_frame_number_get	-	20	20	20	20	20	20	20	20	20
el_ux	ux_dcd_synergy_function	-	136	136	84	84	84	84	84	84	84

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_dcd_synergy_initialize	-	846	846	750	750	750	750	750	1094	1094
el_ux	ux_dcd_synergy_initialize_complete	-	152	152	136	136	136	136	136	136	136
el_ux	ux_dcd_synergy_interrupt_handler	-	744	744	656	656	656	656	656	704	704
el_ux	ux_dcd_synergy_register_clear	-	16	16	16	16	16	16	16	16	16
el_ux	ux_dcd_synergy_register_read	-	12	12	8	8	8	8	8	8	8
el_ux	ux_dcd_synergy_register_set	-	16	16	12	12	12	12	12	12	12
el_ux	ux_dcd_synergy_register_write	-	12	12	8	8	8	8	8	8	8
el_ux	ux_dcd_synergy_transfer_callback	-	716	716	752	752	752	752	752	752	752
el_ux	ux_dcd_synergy_transfer_request	-	1068	1068	1148	1148	1148	1148	1148	1164	1164
el_ux	ux_hcd_synergy_asynch_queue_process	-	240	240	208	208	208	208	208	208	208
el_ux	ux_hcd_synergy_asynch_queue_process_bemp	-	324	324	316	316	316	316	316	316	316
el_ux	ux_hcd_synergy_asynch_queue_process_brdy	-	508	508	488	488	488	488	488	488	488
el_ux	ux_hcd_synergy_asynch_queue_process_nrdy	-	220	220	228	228	228	228	228	228	228
el_ux	ux_hcd_synergy_asynch_queue_process_sign	-	56	56	60	60	60	60	60	60	60
el_ux	ux_hcd_synergy_asynch_schedule	-	92	92	80	80	80	80	80	80	80
el_ux	ux_hcd_synergy_asynchronous_endpoint_create	-	68	68	60	60	60	60	60	60	60
el_ux	ux_hcd_synergy_asynchronous_endpoint_destroy	-	128	128	120	120	120	120	120	120	120
el_ux	ux_hcd_synergy_buffer_empty_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_hcd_synergy_buffer_notready_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_hcd_synergy_buffer_read	-	88	88	88	88	88	88	88	88	88
el_ux	ux_hcd_synergy_buffer_ready_interrupt	-	76	76	84	84	84	84	84	84	84
el_ux	ux_hcd_synergy_buffer_write	-	128	128	140	140	140	140	140	140	140
el_ux	ux_hcd_synergy_bulk_endpoint_create	-	368	368	380	380	380	380	380	500	500
el_ux	ux_hcd_synergy_bulk_int_td_add	-	228	228	232	232	232	232	232	232	232
el_ux	ux_hcd_synergy_control_endpoint_create	-	136	136	140	140	140	140	140	140	140
el_ux	ux_hcd_synergy_control_td_add	-	364	364	364	364	364	364	364	364	364
el_ux	ux_hcd_synergy_controller_disable	-	12	12	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_current_endpoint_change	-	116	116	124	124	124	124	124	124	124
el_ux	ux_hcd_synergy_data_buffer_size	-	44	44	40	40	40	40	40	40	40
el_ux	ux_hcd_synergy_ed_obtain	-	64	64	60	60	60	60	60	60	60
el_ux	ux_hcd_synergy_ed_td_clean	-	24	24	24	24	24	24	24	24	24
el_ux	ux_hcd_synergy_endpoint_nak_set	-	32	32	24	24	24	24	24	24	24

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_hcd_synergy_endpoint_reset	-	12	12	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_entry	-	400	400	380	380	380	380	380	380	380
el_ux	ux_hcd_synergy_fifo_port_change	-	72	72	64	64	64	64	64	64	64
el_ux	ux_hcd_synergy_fifo_read	-	732	732	624	624	624	624	624	712	712
el_ux	ux_hcd_synergy_fifoc_write	-	328	328	292	292	292	292	292	488	488
el_ux	ux_hcd_synergy_fifod_write	-	484	484	436	436	436	436	436	568	568
el_ux	ux_hcd_synergy_frame_number_get	-	88	88	92	92	92	92	92	92	92
el_ux	ux_hcd_synergy_frame_number_set	-	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_initialize	-	1280	1276	1168	1168	1168	1164	1164	1460	1460
el_ux	ux_hcd_synergy_interrupt_endpoint_create	-	392	392	392	392	392	392	392	392	392
el_ux	ux_hcd_synergy_interrupt_handler	-	1056	1056	968	968	968	968	968	1020	1020
el_ux	ux_hcd_synergy_iso_queue_process	-	180	180	152	152	152	152	152	152	152
el_ux	ux_hcd_synergy_iso_queue_process_bemp	-	212	212	220	220	220	220	220	220	220
el_ux	ux_hcd_synergy_iso_queue_process_brdy	-	276	276	264	264	264	264	264	264	264
el_ux	ux_hcd_synergy_iso_queue_process_nrdy	-	152	152	152	152	152	152	152	152	152
el_ux	ux_hcd_synergy_iso_schedule	-	92	92	80	80	80	80	80	80	80
el_ux	ux_hcd_synergy_iso_td_add	-	136	136	132	132	132	132	132	132	132
el_ux	ux_hcd_synergy_isochronous_endpoint_create	-	408	408	396	396	396	396	396	620	620
el_ux	ux_hcd_synergy_isochronous_td_obtain	-	64	64	60	60	60	60	60	60	60
el_ux	ux_hcd_synergy_least_traffic_list_get	-	68	68	68	68	68	68	68	68	68
el_ux	ux_hcd_synergy_periodic_endpoint_destroy	-	132	132	128	128	128	128	128	128	128
el_ux	ux_hcd_synergy_periodic_schedule	-	76	76	72	72	72	72	72	72	72
el_ux	ux_hcd_synergy_periodic_tree_create	-	164	164	144	144	144	144	144	144	144
el_ux	ux_hcd_synergy_port_disable	-	36	36	36	36	36	36	36	36	36
el_ux	ux_hcd_synergy_port_enable	-	68	68	68	68	68	68	68	68	68
el_ux	ux_hcd_synergy_port_reset	-	136	136	120	120	120	120	120	120	120
el_ux	ux_hcd_synergy_port_resume	-	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_port_status_get	-	88	88	88	88	88	88	88	88	88
el_ux	ux_hcd_synergy_port_suspend	-	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_down_port	-	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_on_port	-	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_root_hubs	-	4	4	4	4	4	4	4	4	4

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_hcd_synergy_register_clear	-	12	12	12	12	12	12	12	12	12
el_ux	ux_hcd_synergy_register_read	-	8	8	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_register_set	-	12	12	12	12	12	12	12	12	12
el_ux	ux_hcd_synergy_register_write	-	8	8	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_regular_td_obtain	-	96	96	96	96	96	96	96	96	96
el_ux	ux_hcd_synergy_request_bulk_transfer	-	224	224	216	216	216	216	216	340	340
el_ux	ux_hcd_synergy_request_control_transfer	-	528	528	508	508	508	508	508	508	508
el_ux	ux_hcd_synergy_request_interrupt_transfer	-	76	76	72	72	72	72	72	72	72
el_ux	ux_hcd_synergy_request_isochronous_transfer	-	312	312	288	288	288	288	288	288	288
el_ux	ux_hcd_synergy_request_transfer	-	68	68	72	72	72	72	72	72	72
el_ux	ux_hcd_synergy_td_add	-	24	24	24	24	24	24	24	24	24
el_ux	ux_hcd_synergy_transfer_abort	-	244	244	244	244	244	244	244	244	244
elc	r_elc	-	336	336	312	312	312	312	312	312	312
external_irq	sf_external_irq	-	484	484	476	476	476	476	476	476	476
flash	hw_codeflash	-	1652	1652	1612	1612	1612	1612	-	-	-
flash	hw_codeflash_extra	-	444	444	408	408	408	408	-	-	-
flash	hw_dataflash	-	1132	1132	1048	1048	1048	1048	-	-	-
flash	hw_flash_common	-	24	24	28	28	28	28	-	-	-
flash	hw_flash_hp	-	-	-	-	-	-	-	3012	3012	2996
flash	hw_flash_lp	-	1116	1116	1180	1180	1180	1180	-	-	-
flash_hp	r_flash_hp	-	-	-	-	-	-	-	3106	3106	3106
flash_lp	r_flash_lp	-	3334	3334	3118	3118	3118	3118	-	-	-
fmi	r_fmi	-	1216	1216	1152	1152	1152	1156	1156	1156	1156
glcd	r_glcd	-	-	-	-	-	-	-	-	6431	6431
gpt	r_gpt	-	2744	2744	2368	2368	2368	2372	2416	2416	2416
gpt_input_capture	r_gpt_input_capture	-	1716	1716	1536	1536	1536	1536	1536	1536	1536
i2c	sf_i2c	-	-	-	-	1200	1200	1200	1200	-	1200
icu	r_icu	-	1004	1004	912	912	912	912	912	912	912
ioport	r_ioport	-	1654	1654	1780	1780	1780	1780	1780	1780	1780
iwdt	r_iwdt	-	611	611	591	591	591	591	591	591	591
jpeg	r_jpeg_common	-	-	-	-	-	-	-	-	96	96
jpeg	r_jpeg_decode	-	-	-	-	-	-	-	-	2364	2364

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
jpeg	r_jpeg_encode	-	-	-	-	-	-	-	-	1968	1968
jpeg	sf_jpeg_decode	-	-	-	-	-	-	-	-	1176	1176
kint	r_kint	-	989	989	853	853	853	853	853	853	853
lpm	hw_lpm_common	-	104	-	-	-	-	92	-	-	92
lpm	hw_lpm_s124	-	24	-	-	-	-	-	-	-	-
lpm	hw_lpm_s3a7	-	-	-	-	-	-	24	-	-	-
lpm	hw_lpm_s7g2	-	-	-	-	-	-	-	-	-	100
lpm	hw_lpmv2_s124	-	584	-	-	-	-	0	-	-	-
lpm	hw_lpmv2_s128	-	-	584	-	-	-	0	-	-	-
lpm	hw_lpmv2_s1ja	-	-	-	-	-	-	0	-	-	-
lpm	hw_lpmv2_s3a1	-	-	-	600	-	-	0	-	-	-
lpm	hw_lpmv2_s3a3	-	-	-	-	600	-	0	-	-	-
lpm	hw_lpmv2_s3a6	-	-	-	-	-	592	0	-	-	-
lpm	hw_lpmv2_s3a7	-	-	-	-	-	-	600	-	-	-
lpm	hw_lpmv2_s5d5	-	-	-	-	-	-	0	1140	-	-
lpm	hw_lpmv2_s5d9	-	-	-	-	-	-	0	-	1140	-
lpm	hw_lpmv2_s7g2	-	-	-	-	-	-	0	-	-	1140
lpm	r_lpm	-	860	-	-	-	-	848	-	-	1304
lpmv2	r_lpmv2	-	220	220	216	216	216	216	216	216	216
lvd	r_lvd	-	1968	1968	1676	1676	1676	1672	1672	1672	1672
message	sf_message	-	1404	1404	1320	1320	1320	1320	1320	1320	1320
pdcc	r_pdc	-	-	-	-	-	-	-	1784	-	1784
power_profiles	sf_power_profiles	-	748	-	-	-	-	764	-	-	764
power_profiles	sf_power_profiles_v2	-	-	-	-	-	-	580	-	-	-
qspi	r_qsapi	-	-	-	-	920	-	920	920	920	920
riic	r_riic	-	5112	5112	5164	5164	5164	5160	5160	5160	5160
riic_slave	r_riic_slave	-	2528	2528	2560	2560	2560	2560	2560	2560	2560
rsapi	r_rsapi	-	3960	3960	3732	3732	3732	3736	3736	3736	3736
rtc	r_rtc	-	4892	-	-	-	-	4412	-	-	4376
sci_i2c	r_sci_i2c	-	5081	5081	5009	5009	5009	5009	5009	5069	5069
sci_spi	r_sci_spi	-	3576	3576	3276	3276	3276	3268	3268	3268	3268
sci_uart	r_sci_uart	-	4566	4566	4302	4302	4302	4302	4302	4302	4302

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
sdmmc	r_sdmmc	-	-	-	5300	5300	-	5308	5308	5308	5308
sf_audio_record_i2s	sf_audio_record_i2s	-	-	-	576	576	576	576	576	576	576
slcdc	r_slcdc	-	-	-	1292	1292	1292	1292	-	-	-
spi	sf_spi	-	1456	1456	1448	1448	1448	1448	1448	1448	1448
spi	spi_hcd	-	-	-	-	-	-	-	-	-	2903
ssi	r_ssi	-	-	-	4745	4745	4745	4745	4745	4745	4745
system	startup_<mcu>	-	80	-	-	-	-	-	-	-	-
system	startup_S128	-	-	92	-	-	-	-	-	-	-
system	startup_S1JA	-	-	-	-	-	-	-	-	-	-
system	startup_S3A1	-	-	-	80	-	-	-	-	-	-
system	startup_S3A3	-	-	-	-	80	-	-	-	-	-
system	startup_S3A6	-	-	-	-	-	80	-	-	-	-
system	startup_S3A7	-	-	-	-	-	-	80	-	-	-
system	startup_S5D5	-	-	-	-	-	-	-	80	-	-
system	startup_S5D9	-	-	-	-	-	-	-	-	80	-
system	startup_S7G2	-	-	-	-	-	-	-	-	-	80
system	storerecall	-	-	-	-	-	-	-	-	-	0
system	system_S124	-	392	-	-	-	-	-	-	-	-
system	system_S128	-	-	352	-	-	-	-	-	-	-
system	system_S1JA	-	-	-	-	-	-	-	-	-	-
system	system_S3A1	-	-	-	396	-	-	-	-	-	-
system	system_S3A3	-	-	-	-	476	-	-	-	-	-
system	system_S3A6	-	-	-	-	-	976	-	-	-	-
system	system_S3A7	-	-	-	-	-	-	476	-	-	-
system	system_S5D5	-	-	-	-	-	-	-	1388	-	-
system	system_S5D9	-	-	-	-	-	-	-	-	1492	-
system	system_S7G2	-	-	-	-	-	-	-	-	-	1644
telnet	sf_comms_telnet	-	-	-	-	-	-	-	1486	1486	1486
thread_monitor	sf_thread_monitor	-	1139	1139	1063	1063	1063	1063	1063	1063	1063
uart	sf_uart_comms	-	1190	1190	1162	1162	1162	1162	1162	1166	1166
ul_gx	gx_display_driver_synergy_dave2d	-	-	-	-	-	-	-	-	14095	14095
ul_gx	gx_display_driver_synergy_dave2d_8bit_palette	-	-	-	-	-	-	-	-	5760	5760

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
wdt	r_wdt	-	767	767	747	747	747	747	747	747	747
wifi	api_init	-	-	-	-	-	-	-	-	-	493
wifi	api_ioctl	-	-	-	-	-	-	-	-	-	2952
wifi	api_stack_offload	-	-	-	-	-	-	-	-	-	12952
wifi	api_txrx	-	-	-	-	-	-	-	-	-	723
wifi	api_wmi_rx	-	-	-	-	-	-	-	-	-	2163
wifi	cust_api_init	-	-	-	-	-	-	-	-	-	470
wifi	cust_api_ioctl	-	-	-	-	-	-	-	-	-	5572
wifi	cust_api_stack_offload	-	-	-	-	-	-	-	-	-	4534
wifi	cust_api_stack_txrx	-	-	-	-	-	-	-	-	-	268
wifi	cust_api_txrx	-	-	-	-	-	-	-	-	-	244
wifi	cust_api_wmi_rx	-	-	-	-	-	-	-	-	-	1650
wifi	cust_driver_main	-	-	-	-	-	-	-	-	-	320
wifi	cust_driver_netbuf	-	-	-	-	-	-	-	-	-	1165
wifi	cust_spi_hcd	-	-	-	-	-	-	-	-	-	758
wifi	custom_qcom_api	-	-	-	-	-	-	-	-	-	480
wifi	driver_diag	-	-	-	-	-	-	-	-	-	594
wifi	driver_init	-	-	-	-	-	-	-	-	-	1321
wifi	driver_main	-	-	-	-	-	-	-	-	-	1318
wifi	driver_netbuf	-	-	-	-	-	-	-	-	-	84
wifi	driver_txrx	-	-	-	-	-	-	-	-	-	538
wifi	Dset	-	-	-	-	-	-	-	-	-	380
wifi	dset_api	-	-	-	-	-	-	-	-	-	592
wifi	enet_irq_veneer	-	-	-	-	-	-	-	56	56	56
wifi	ether_phy	-	-	-	-	-	-	-	876	876	904
wifi	gt202_debug	-	-	-	-	-	-	-	-	-	807
wifi	gt202_util	-	-	-	-	-	-	-	-	-	386
wifi	gt202_wifi_ctrl	-	-	-	-	-	-	-	-	-	1751
wifi	gt202_wifi_task_thread	-	-	-	-	-	-	-	-	-	1328
wifi	Htc	-	-	-	-	-	-	-	-	-	1617
wifi	hw_api	-	-	-	-	-	-	-	-	-	334
wifi	Osal	-	-	-	-	-	-	-	-	-	840

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
wifi	qcom_api	-	-	-	-	-	-	-	-	-	7440
wifi	qcom_legacy	-	-	-	-	-	-	-	-	-	56
wifi	rcv_aggr	-	-	-	-	-	-	-	-	-	359
wifi	ring_buffer	-	-	-	-	-	-	-	-	-	320
wifi	sf_wifi_gt202	-	-	-	-	-	-	-	-	-	3153
wifi	sf_wifi_gt202_onchip_stack	-	-	-	-	-	-	-	-	-	376
wifi	sf_wifi_gt202_socket	-	-	-	-	-	-	-	-	-	1127
wifi	sf_wifi_nsal_nx	-	-	-	-	-	-	-	-	-	1468
wifi	Util	-	-	-	-	-	-	-	-	-	196
wifi	Wmi	-	-	-	-	-	-	-	-	-	3164

16.2 IAR ROM の推定メモリサイズ

表 16.2 各 MCU グループの IAR ROM 結果まとめ

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
_bsp	bsp_cache	92	8	8	96	96	96	96	88	96	96
_bsp	bsp_clocks	212	80	80	280	288	284	280	357	373	424
_bsp	bsp_common	82	86	86	86	86	86	86	86	86	86
_bsp	bsp_common_leds	20	24	24	24	24	24	24	24	24	24
_bsp	bsp_delay	94	102	102	94	94	94	94	94	94	94
_bsp	bsp_feature	244	228	252	280	280	280	284	244	244	244
_bsp	bsp_fmi_R7FS124*	-	1024	-	-	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS128*	-	-	1024	-	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS1JA*	1024	-	-	-	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A1*	-	-	-	1024	-	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A3*	-	-	-	-	1024	-	-	-	-	-
_bsp	bsp_fmi_R7FS3A6*	-	-	-	-	-	1024	-	-	-	-
_bsp	bsp_fmi_R7FS3A7*	-	-	-	-	-	-	1024	-	-	-
_bsp	bsp_fmi_R7FS5D5*	-	-	-	-	-	-	-	1024	-	-
_bsp	bsp_fmi_R7FS5D9*	-	-	-	-	-	-	-	-	1024	-
_bsp	bsp_fmi_R7FS7G2*	-	-	-	-	-	-	-	-	-	1024
_bsp	bsp_group_irq	364	300	320	384	384	384	384	384	384	384
_bsp	bsp_hw_locks	300	275	290	385	430	420	420	540	540	540
_bsp	bsp_init	2	2	2	4	4	2	4	14	14	20
_bsp	bsp_irq	92	92	92	88	88	88	88	88	88	88
_bsp	bsp_leds	16	16	16	16	16	16	16	12	16	16
_bsp	bsp_locking	236	242	242	232	232	232	232	232	232	232
_bsp	bsp_module_stop	256	252	244	240	276	276	240	248	248	248
_bsp	bsp_qspi	-	-	-	940	940	-	940	940	940	952
_bsp	bsp_register_protection	152	156	156	160	160	160	160	160	160	160
_bsp	bsp_rom_registers	76	24	76	76	76	76	76	76	76	76
_bsp	bsp_sbrk	0	0	0	0	0	0	0	0	0	0
_bsp	bsp_sdram	-	-	-	-	-	-	-	236	236	236
adc	r_adc	3896	3936	3936	3856	3856	3856	3852	3852	3852	3852

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
adc_periodic	sf_adc_periodic	956	1004	1004	968	968	968	968	968	968	968
agt	r_agt	2030	2094	2094	1922	1922	1922	1918	1918	1918	1918
analog	r_acmphs	964	-	-	-	-	-	-	-	-	-
analog	r_acmplp	1188	-	-	-	-	-	-	-	-	-
analog	r_opamp	812	-	840	-	-	-	-	-	-	-
analog	r_sdadc	2692	-	-	-	-	-	-	-	-	-
audio_playback	sf_audio_playback	2736	2900	2900	2766	2766	2766	2766	2766	2766	2766
audio_playback_dac	sf_audio_playback_hw_dac	700	676	748	728	728	728	728	728	728	728
audio_playback_i2s	sf_audio_playback_hw_i2s	-	-	-	236	236	236	236	236	236	236
audio_record_adc	sf_audio_record_adc	-	376	-	-	360	360	360	360	360	360
ble	ble_rl78g1d_if	-	-	-	-	-	-	-	-	-	8269
ble	ble_rl78g1d_prf_anp_if	-	-	-	-	-	-	-	-	-	878
ble	ble_rl78g1d_prf_blp_if	-	-	-	-	-	-	-	-	-	1060
ble	ble_rl78g1d_prf_fmp_if	-	-	-	-	-	-	-	-	-	492
ble	ble_rl78g1d_prf_hid_if	-	-	-	-	-	-	-	-	-	1660
ble	ble_rl78g1d_prf_hrp_if	-	-	-	-	-	-	-	-	-	984
ble	ble_rl78g1d_prf_htp_if	-	-	-	-	-	-	-	-	-	1040
ble	ble_rl78g1d_prf_if	-	-	-	-	-	-	-	-	-	1732
ble	ble_rl78g1d_prf_paps_if	-	-	-	-	-	-	-	-	-	962
ble	ble_rl78g1d_prf_pxp_if	-	-	-	-	-	-	-	-	-	768
ble	ble_rl78g1d_prf_scps_if	-	-	-	-	-	-	-	-	-	528
ble	ble_rl78g1d_prf_tip_if	-	-	-	-	-	-	-	-	-	1080
ble	ble_serial	-	-	-	-	-	-	-	-	-	2328
ble	bmi	-	-	-	-	-	-	-	-	-	1408
ble	rble_api_anpc	-	-	-	-	-	-	-	-	-	560
ble	rble_api_anps	-	-	-	-	-	-	-	-	-	312
ble	rble_api_blpc	-	-	-	-	-	-	-	-	-	752
ble	rble_api_blps	-	-	-	-	-	-	-	-	-	360
ble	rble_api_cppc	-	-	-	-	-	-	-	-	-	1012
ble	rble_api_cppps	-	-	-	-	-	-	-	-	-	900
ble	rble_api_cscpc	-	-	-	-	-	-	-	-	-	822
ble	rble_api_cscps	-	-	-	-	-	-	-	-	-	348

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
ble	rble_api_fmpl	-	-	-	-	-	-	-	-	-	256
ble	rble_api_fmpt	-	-	-	-	-	-	-	-	-	132
ble	rble_api_gap	-	-	-	-	-	-	-	-	-	1940
ble	rble_api_gatt	-	-	-	-	-	-	-	-	-	1208
ble	rble_api_glpc	-	-	-	-	-	-	-	-	-	992
ble	rble_api_glps	-	-	-	-	-	-	-	-	-	612
ble	rble_api_hgbh	-	-	-	-	-	-	-	-	-	1400
ble	rble_api_hghd	-	-	-	-	-	-	-	-	-	460
ble	rble_api_hgrh	-	-	-	-	-	-	-	-	-	1552
ble	rble_api_hrpc	-	-	-	-	-	-	-	-	-	692
ble	rble_api_hrps	-	-	-	-	-	-	-	-	-	262
ble	rble_api_htpc	-	-	-	-	-	-	-	-	-	876
ble	rble_api_htpt	-	-	-	-	-	-	-	-	-	380
ble	rble_api_lnpc	-	-	-	-	-	-	-	-	-	1076
ble	rble_api_lnps	-	-	-	-	-	-	-	-	-	904
ble	rble_api_paspc	-	-	-	-	-	-	-	-	-	472
ble	rble_api_pasps	-	-	-	-	-	-	-	-	-	276
ble	rble_api_pxpm	-	-	-	-	-	-	-	-	-	488
ble	rble_api_pxpr	-	-	-	-	-	-	-	-	-	136
ble	rble_api_rscpc	-	-	-	-	-	-	-	-	-	896
ble	rble_api_rscps	-	-	-	-	-	-	-	-	-	400
ble	rble_api_sm	-	-	-	-	-	-	-	-	-	536
ble	rble_api_sppc	-	-	-	-	-	-	-	-	-	412
ble	rble_api_spps	-	-	-	-	-	-	-	-	-	212
ble	rble_api_tipc	-	-	-	-	-	-	-	-	-	632
ble	rble_api_tips	-	-	-	-	-	-	-	-	-	532
ble	rble_api_vs	-	-	-	-	-	-	-	-	-	940
ble	rble_host	-	-	-	-	-	-	-	-	-	3340
ble	rble_if_api_cb	-	-	-	-	-	-	-	-	-	25072
ble	rscip	-	-	-	-	-	-	-	-	-	252
ble	rscip_cntl	-	-	-	-	-	-	-	-	-	2898
ble	rscip_uart	-	-	-	-	-	-	-	-	-	2080

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
ble	sf_ble_rl78g1d	-	-	-	-	-	-	-	-	-	2536
ble	sf_ble_rl78g1d_onboard_profile	-	-	-	-	-	-	-	-	-	872
ble	timer	-	-	-	-	-	-	-	-	-	321
block_media_sdmmc	sf_block_media_qspi	-	-	-	-	848	-	848	848	848	848
block_media_sdmmc	sf_block_media_ram	-	-	-	-	352	-	352	-	352	352
block_media_sdmmc	sf_block_media_sdmmc	-	-	-	492	492	-	492	492	492	492
cac	r_cac	1820	1804	1804	1864	1864	1864	1860	1860	1860	1860
can	hw_can	1468	1480	1480	1292	1292	1292	1292	1292	1292	1292
can	r_can	2378	2454	2454	2320	2320	2320	2320	2320	2320	2320
cellular	cellular_serial	-	-	-	-	-	712	712	712	712	712
cellular	sf_cellular_cat1	-	-	-	-	-	5568	5568	5568	5568	5568
cellular	sf_cellular_cat1_socket	-	-	-	-	-	548	548	548	548	548
cellular	sf_cellular_cat1_socket_private	-	-	-	-	-	6552	6552	6552	6552	6552
cellular	sf_cellular_cat3	-	-	-	-	-	1744	1744	1744	1744	1744
cellular	sf_cellular_cat3_socket	-	-	-	-	-	524	524	524	524	524
cellular	sf_cellular_cat3_socket_private	-	-	-	-	-	6328	6328	6328	6328	6328
cellular	sf_cellular_common	-	-	-	-	-	1400	1400	1400	1400	1400
cellular	sf_cellular_common_private	-	-	-	-	-	10016	10016	10016	10016	10016
cellular	sf_cellular_qctlcatm1	-	-	-	-	-	-	-	-	-	-
cellular	sf_cellular_qctlcatm1_socket	-	-	-	-	-	-	-	-	-	-
cellular	sf_cellular_qctlcatm1_socket_private	-	-	-	-	-	-	-	-	-	-
cgc	hw_cgc	2490	2538	2538	2496	2496	2496	2496	2496	2496	2496
cgc	r_cgc	2924	3044	3044	2984	2984	2984	2984	2984	2984	2984
console	sf_console	-	-	-	-	-	-	-	-	-	2272
ctsu	hw_ctsu_common	-	432	-	-	392	392	392	-	392	-
ctsu	sf_touch_ctsu	-	4256	-	-	4208	4208	4208	-	4208	-
ctsu	sf_touch_ctsu_button	-	2056	-	-	1976	1976	1976	-	1976	-
ctsu	sf_touch_ctsu_slider	-	-	-	-	2648	-	2648	-	-	-
ctsu	sf_touch_panel_i2c	-	-	-	-	-	-	-	-	1065	1065
ctsu	touch_panel_i2c_ft5x06	-	-	-	-	-	-	-	-	280	280
ctsu	touch_panel_i2c_sx8654	-	-	-	-	-	-	-	-	572	572
ctsu	r_ctsu	-	12508	-	-	11820	11820	11812	-	11812	-

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
dac	r_dac	588	624	624	596	596	596	596	596	596	596
dac8	r_dac8	776	-	804	-	732	-	-	-	-	-
dave/2d	dave_64bitoperation	-	-	-	-	-	-	-	-	0	0
dave/2d	dave_base	-	-	-	-	-	-	-	-	24	24
dave/2d	dave_blit	-	-	-	-	-	-	-	-	1024	1024
dave/2d	dave_box	-	-	-	-	-	-	-	-	1972	1972
dave/2d	dave_circle	-	-	-	-	-	-	-	-	1140	1140
dave/2d	dave_context	-	-	-	-	-	-	-	-	2392	2392
dave/2d	dave_curve	-	-	-	-	-	-	-	-	612	612
dave/2d	dave_dlist	-	-	-	-	-	-	-	-	3396	3396
dave/2d	dave_driver	-	-	-	-	-	-	-	-	2780	2780
dave/2d	dave_edge	-	-	-	-	-	-	-	-	1416	1416
dave/2d	dave_errorcodes	-	-	-	-	-	-	-	-	1180	1180
dave/2d	dave_gradient	-	-	-	-	-	-	-	-	498	498
dave/2d	dave_hardware	-	-	-	-	-	-	-	-	276	276
dave/2d	dave_line	-	-	-	-	-	-	-	-	6368	6368
dave/2d	dave_math	-	-	-	-	-	-	-	-	256	256
dave/2d	dave_memory	-	-	-	-	-	-	-	-	218	218
dave/2d	dave_pattern	-	-	-	-	-	-	-	-	320	320
dave/2d	dave_perfcoun	-	-	-	-	-	-	-	-	168	168
dave/2d	dave_polyline	-	-	-	-	-	-	-	-	1716	1716
dave/2d	dave_quad	-	-	-	-	-	-	-	-	2754	2754
dave/2d	dave_rbuffer	-	-	-	-	-	-	-	-	1660	1660
dave/2d	dave_render	-	-	-	-	-	-	-	-	2978	2978
dave/2d	dave_texture	-	-	-	-	-	-	-	-	2612	2612
dave/2d	dave_triangle	-	-	-	-	-	-	-	-	1942	1942
dave/2d	dave_utility	-	-	-	-	-	-	-	-	1098	1098
dave/2d	dave_viewport	-	-	-	-	-	-	-	-	568	568
dave/2d	dave_wedge	-	-	-	-	-	-	-	-	1050	1050
dave/2d	sf_tes_2d_drw_base	-	-	-	-	-	-	-	-	300	300
dave/2d	sf_tes_2d_drw_irq	-	-	-	-	-	-	-	-	336	336
dave/2d	sf_tes_2d_drw_memory	-	-	-	-	-	-	-	-	208	208

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
dmac	r_dmac	-	-	-	1760	1760	1760	1764	1764	1764	1764
doc	r_doc	832	872	872	800	800	800	796	796	796	796
dtc	r_dtc	1860	1888	1888	1960	1960	1960	1980	2108	2108	2108
el_fx	sf_el_fx	-	-	-	464	464	-	464	464	464	464
el_gx	sf_el_gx	-	-	-	-	-	-	-	-	1348	1348
el_nx	nx_hw_init	-	-	-	-	-	-	-	908	908	908
el_nx	nx_renesas_synergy	-	-	-	-	-	-	-	2224	2224	2224
el_nx	sf_el_nx_comms	-	-	-	-	-	-	-	1508	1508	1508
el_ux	sf_el_ux_comms	748	776	776	756	756	756	756	756	756	756
el_ux	usb_irq_veneer	12	16	16	12	12	12	12	12	72	72
el_ux	usbfs_irq_veneer	32	36	36	34	34	34	34	34	34	84
el_ux	ux_dcd_synergy_buffer_empty_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_dcd_synergy_buffer_notready_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_dcd_synergy_buffer_read	68	68	68	66	66	66	66	66	66	66
el_ux	ux_dcd_synergy_buffer_ready_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_dcd_synergy_buffer_write	114	118	118	114	114	114	114	114	114	114
el_ux	ux_dcd_synergy_current_endpoint_change	166	172	172	166	166	166	166	166	166	166
el_ux	ux_dcd_synergy_data_buffersize	66	68	68	64	64	64	64	64	64	64
el_ux	ux_dcd_synergy_endpoint_create	386	386	386	448	448	448	448	448	448	448
el_ux	ux_dcd_synergy_endpoint_destroy	156	156	156	154	154	154	154	154	154	154
el_ux	ux_dcd_synergy_endpoint_nak_set	20	26	26	20	20	20	20	20	20	20
el_ux	ux_dcd_synergy_endpoint_reset	102	106	106	100	100	100	100	100	100	100
el_ux	ux_dcd_synergy_endpoint_stall	42	46	46	42	42	42	42	42	42	42
el_ux	ux_dcd_synergy_endpoint_status	32	32	32	32	32	32	32	32	32	32
el_ux	ux_dcd_synergy_fifo_port_change	54	56	56	52	52	52	52	52	52	52
el_ux	ux_dcd_synergy_fifo_read	862	890	890	858	858	858	858	858	1094	1094
el_ux	ux_dcd_synergy_fifoc_write	338	350	350	280	280	280	280	280	412	412
el_ux	ux_dcd_synergy_fifod_write	702	730	730	698	698	698	698	698	804	804
el_ux	ux_dcd_synergy_frame_number_get	20	20	20	20	20	20	20	20	20	20
el_ux	ux_dcd_synergy_function	76	94	94	82	82	82	82	82	82	82
el_ux	ux_dcd_synergy_initialize	804	808	808	752	752	752	756	756	1104	1104
el_ux	ux_dcd_synergy_initialize_complete	132	132	132	124	124	124	124	124	124	124

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_dcd_synergy_interrupt_handler	660	676	676	624	624	624	624	624	668	668
el_ux	ux_dcd_synergy_register_clear	16	16	16	16	16	16	16	16	16	16
el_ux	ux_dcd_synergy_register_read	12	12	12	10	10	10	10	10	10	10
el_ux	ux_dcd_synergy_register_set	16	16	16	14	14	14	14	14	14	14
el_ux	ux_dcd_synergy_register_write	12	12	12	10	10	10	10	10	10	10
el_ux	ux_dcd_synergy_transfer_callback	636	656	656	654	654	654	654	654	654	654
el_ux	ux_dcd_synergy_transfer_request	922	900	900	808	808	808	808	808	832	832
el_ux	ux_hcd_synergy_asynch_queue_process	196	200	200	216	216	216	216	216	216	216
el_ux	ux_hcd_synergy_asynch_queue_process_bemp	290	300	300	290	290	290	290	290	290	290
el_ux	ux_hcd_synergy_asynch_queue_process_brdy	438	442	442	428	428	428	428	428	428	428
el_ux	ux_hcd_synergy_asynch_queue_process_nrdy	194	200	200	194	194	194	194	194	194	194
el_ux	ux_hcd_synergy_asynch_queue_process_sign	54	56	56	56	56	56	56	56	56	56
el_ux	ux_hcd_synergy_asynch_schedule	66	78	78	70	70	70	70	70	70	70
el_ux	ux_hcd_synergy_asynchronous_endpoint_create	62	66	66	62	62	62	62	62	62	62
el_ux	ux_hcd_synergy_asynchronous_endpoint_destroy	116	122	122	114	114	114	114	114	114	114
el_ux	ux_hcd_synergy_buffer_empty_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_hcd_synergy_buffer_notready_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_hcd_synergy_buffer_read	78	78	78	76	76	76	76	76	76	76
el_ux	ux_hcd_synergy_buffer_ready_interrupt	76	76	76	78	78	78	78	78	78	78
el_ux	ux_hcd_synergy_buffer_write	112	114	114	110	110	110	110	110	110	110
el_ux	ux_hcd_synergy_bulk_endpoint_create	414	430	430	414	414	414	414	414	528	528
el_ux	ux_hcd_synergy_bulk_int_td_add	218	222	222	212	212	212	212	212	212	212
el_ux	ux_hcd_synergy_control_endpoint_create	146	152	152	138	138	138	138	138	138	138
el_ux	ux_hcd_synergy_control_td_add	334	338	338	330	330	330	330	330	330	330
el_ux	ux_hcd_synergy_controller_disable	10	10	10	10	10	10	10	10	10	10
el_ux	ux_hcd_synergy_current_endpoint_change	110	116	116	106	106	106	106	106	106	106
el_ux	ux_hcd_synergy_data_buffer_size	36	38	38	44	44	44	44	44	44	44
el_ux	ux_hcd_synergy_ed_obtain	56	56	56	56	56	56	56	56	56	56
el_ux	ux_hcd_synergy_ed_td_clean	20	20	20	20	20	20	20	20	20	20
el_ux	ux_hcd_synergy_endpoint_nak_set	20	26	26	20	20	20	20	20	20	20
el_ux	ux_hcd_synergy_endpoint_reset	8	8	8	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_entry	328	330	330	356	356	356	356	356	356	356

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_hcd_synergy_fifo_port_change	58	60	60	58	58	58	58	58	58	58
el_ux	ux_hcd_synergy_fifo_read	604	620	620	628	628	628	628	628	714	714
el_ux	ux_hcd_synergy_fifoc_write	258	270	270	252	252	252	252	252	416	416
el_ux	ux_hcd_synergy_fifod_write	460	472	472	456	456	456	456	456	580	580
el_ux	ux_hcd_synergy_frame_number_get	76	76	76	84	84	84	84	84	84	84
el_ux	ux_hcd_synergy_frame_number_set	2	2	2	2	2	2	2	2	2	2
el_ux	ux_hcd_synergy_initialize	1160	1172	1176	1124	1124	1124	1124	1124	1392	1392
el_ux	ux_hcd_synergy_interrupt_endpoint_create	412	436	436	406	406	406	406	406	406	406
el_ux	ux_hcd_synergy_interrupt_handler	940	964	964	904	904	904	904	904	944	944
el_ux	ux_hcd_synergy_iso_queue_process	136	140	140	126	126	126	126	126	126	126
el_ux	ux_hcd_synergy_iso_queue_process_bemp	202	206	206	204	204	204	204	204	204	204
el_ux	ux_hcd_synergy_iso_queue_process_brdy	258	270	270	258	258	258	258	258	258	258
el_ux	ux_hcd_synergy_iso_queue_process_nrdy	144	146	146	144	144	144	144	144	144	144
el_ux	ux_hcd_synergy_iso_schedule	66	78	78	70	70	70	70	70	70	70
el_ux	ux_hcd_synergy_iso_td_add	138	138	138	132	132	132	132	132	132	132
el_ux	ux_hcd_synergy_isochronous_endpoint_create	402	412	412	400	400	400	400	400	536	536
el_ux	ux_hcd_synergy_isochronous_td_obtain	56	56	56	56	56	56	56	56	56	56
el_ux	ux_hcd_synergy_least_traffic_list_get	62	62	62	66	66	66	66	66	66	66
el_ux	ux_hcd_synergy_periodic_endpoint_destroy	126	132	132	128	128	128	128	128	128	128
el_ux	ux_hcd_synergy_periodic_schedule	66	68	68	60	60	60	60	60	60	60
el_ux	ux_hcd_synergy_periodic_tree_create	292	296	296	292	292	292	292	292	292	292
el_ux	ux_hcd_synergy_port_disable	34	34	34	34	34	34	34	34	34	34
el_ux	ux_hcd_synergy_port_enable	70	70	70	70	70	70	70	70	70	70
el_ux	ux_hcd_synergy_port_reset	132	132	132	120	120	120	120	120	120	120
el_ux	ux_hcd_synergy_port_resume	4	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_port_status_get	90	90	90	82	82	82	82	82	82	82
el_ux	ux_hcd_synergy_port_suspend	4	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_down_port	4	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_on_port	4	4	4	4	4	4	4	4	4	4
el_ux	ux_hcd_synergy_power_root_hubs	2	2	2	2	2	2	2	2	2	2
el_ux	ux_hcd_synergy_register_clear	12	12	12	14	14	14	14	14	14	14
el_ux	ux_hcd_synergy_register_read	8	8	8	8	8	8	8	8	8	8

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
el_ux	ux_hcd_synergy_register_set	12	12	12	12	12	12	12	12	12	12
el_ux	ux_hcd_synergy_register_write	8	8	8	8	8	8	8	8	8	8
el_ux	ux_hcd_synergy_regular_td_obtain	88	88	88	92	92	92	92	92	92	92
el_ux	ux_hcd_synergy_request_bulk_transfer	194	200	200	176	176	176	176	176	284	284
el_ux	ux_hcd_synergy_request_control_transfer	448	460	460	438	438	438	438	438	438	438
el_ux	ux_hcd_synergy_request_interrupt_transfer	72	74	74	68	68	68	68	68	68	68
el_ux	ux_hcd_synergy_request_isochronous_transfer	272	288	288	242	242	242	242	242	242	242
el_ux	ux_hcd_synergy_request_transfer	52	62	62	52	52	52	52	52	52	52
el_ux	ux_hcd_synergy_td_add	22	24	24	22	22	22	22	22	22	22
el_ux	ux_hcd_synergy_transfer_abort	236	240	240	230	230	230	230	230	230	230
elc	r_elc	288	296	296	304	304	304	304	304	304	304
external_irq	sf_external_irq	400	436	436	420	420	420	420	420	420	420
flash	hw_codeflash	1344	1360	1360	1348	1348	1348	1348	-	-	-
flash	hw_codeflash_extra	376	378	378	320	320	320	320	-	-	-
flash	hw_dataflash	888	900	900	876	876	876	876	-	-	-
flash	hw_flash_common	20	26	26	22	22	22	22	-	-	-
flash	hw_flash_hp	-	-	-	-	-	-	-	2726	2726	2666
flash	hw_flash_lp	1120	1128	1128	1124	1124	1124	1124	-	-	-
flash_hp	r_flash_hp	-	-	-	-	-	-	-	2834	2834	2834
flash_lp	r_flash_lp	2822	2994	2994	2846	2846	2846	2846	-	-	-
fmi	r_fmi	1476	1496	1496	1444	1444	1444	1448	1448	1448	1448
glcd	r_glcd	-	-	-	-	-	-	-	-	4916	4916
gpt	r_gpt	2292	2368	2368	2108	2108	2108	2112	2164	2164	2164
gpt_input_capture	r_gpt_input_capture	1508	1520	1520	1520	1520	1520	1520	1520	1520	1520
i2c	sf_i2c	-	-	-	-	1120	1120	1120	1120	-	1120
icu	r_icu	952	944	944	896	896	896	892	892	892	892
ioport	r_ioport	1234	1270	1270	1346	1346	1346	1346	1346	1346	1346
iwdt	r_iwdt	528	564	564	568	568	568	568	568	568	568
jpeg	r_jpeg_common	-	-	-	-	-	-	-	-	52	52
jpeg	r_jpeg_decode	-	-	-	-	-	-	-	-	2256	2256
jpeg	r_jpeg_encode	-	-	-	-	-	-	-	-	2040	2040
jpeg	sf_jpeg_decode	-	-	-	-	-	-	-	-	984	984

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
kint	r_kint	872	864	864	864	864	864	860	860	860	860
lpm	hw_lpm_common	-	100	-	-	-	-	96	-	-	96
lpm	hw_lpm_s124	-	44	-	-	-	-	-	-	-	-
lpm	hw_lpm_s3a7	-	-	-	-	-	-	42	-	-	-
lpm	hw_lpm_s7g2	-	-	-	-	-	-	-	-	-	88
lpm	hw_lpmv2_s124	0	528	-	-	-	-	0	-	-	-
lpm	hw_lpmv2_s128	-	-	528	-	-	-	0	-	-	-
lpm	hw_lpmv2_s1ja	512	-	-	-	-	-	0	-	-	-
lpm	hw_lpmv2_s3a1	-	-	-	528	-	-	0	-	-	-
lpm	hw_lpmv2_s3a3	-	-	-	-	528	-	0	-	-	-
lpm	hw_lpmv2_s3a6	-	-	-	-	-	512	0	-	-	-
lpm	hw_lpmv2_s3a7	-	-	-	-	-	-	528	-	-	-
lpm	hw_lpmv2_s5d5	-	-	-	-	-	-	0	936	-	-
lpm	hw_lpmv2_s5d9	-	-	-	-	-	-	0	-	936	-
lpm	hw_lpmv2_s7g2	-	-	-	-	-	-	0	-	-	936
lpm	r_lpm	-	708	-	-	-	-	712	-	-	1024
lpmv2	r_lpmv2	220	232	232	220	220	220	220	220	220	220
lvd	r_lvd	1768	1772	1772	1660	1660	1660	1656	1656	1656	1656
message	sf_message	1244	1304	1304	1248	1248	1248	1248	1248	1248	1248
pdcc	r_pdc	-	-	-	-	-	-	-	1680	-	1680
power_profiles	sf_power_profiles	-	648	-	-	-	-	636	-	-	636
power_profiles	sf_power_profiles_v2	-	-	-	-	-	-	512	-	-	-
qspi	r_qspi	-	-	-	-	980	-	980	980	980	980
riic	r_riic	4608	4868	4868	4712	4712	4712	4704	4704	4704	4704
riic_slave	r_riic_slave	2206	2306	2306	2228	2228	2228	2224	2224	2224	2224
rspi	r_rsapi	3484	3776	3776	3480	3480	3480	3480	3480	3480	3480
rtc	r_rtc	-	4064	-	-	-	-	3676	-	-	3676
sci_i2c	r_sci_i2c	4152	4260	4260	4320	4320	4320	4312	4312	4312	4312
sci_spi	r_sci_spi	2912	3052	3052	2964	2964	2964	2960	2960	2960	2960
sci_uart	r_sci_uart	3892	4144	4144	3916	3916	3916	3912	3912	3912	3912
sdmmc	r_sdmmc	-	-	-	6280	6280	-	6284	6284	6284	6284
sf_audio_record_i2s	sf_audio_record_i2s	-	-	-	540	540	540	540	540	540	540

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
slcdc	r_slcdc	-	-	-	1336	1336	1336	1336	-	-	-
spi	sf_spi	-	-	-	-	-	-	-	-	-	-
spi	spi_hcd	-	-	-	-	-	-	-	-	-	2572
ssi	r_ssi	-	-	-	3980	3980	3980	3976	3976	3976	3976
system	startup_S124	-	80	-	-	-	-	-	-	-	-
system	startup_S128	-	-	5212	-	-	-	-	-	-	-
system	startup_S1JA	92	-	-	-	-	-	-	-	-	-
system	startup_S3A1	-	-	-	9296	-	-	-	-	-	-
system	startup_S3A3	-	-	-	-	9296	-	-	-	-	-
system	startup_S3A6	-	-	-	-	-	80	-	-	-	-
system	startup_S3A7	-	-	-	-	-	-	9296	-	-	-
system	startup_S5D5	-	-	-	-	-	-	-	80	-	-
system	startup_S5D9	-	-	-	-	-	-	-	-	80	-
system	startup_S7G2	-	-	-	-	-	-	-	-	-	80
system	storerecall	-	-	-	-	-	-	-	-	-	0
system	system_S124	-	414	-	-	-	-	-	-	-	-
system	system_S128	-	-	366	-	-	-	-	-	-	-
system	system_S1JA	342	-	-	-	-	-	-	-	-	-
system	system_S3A1	-	-	-	406	-	-	-	-	-	-
system	system_S3A3	-	-	-	-	486	-	-	-	-	-
system	system_S3A6	-	-	-	-	-	990	-	-	-	-
system	system_S3A7	-	-	-	-	-	-	486	-	-	-
system	system_S5D5	-	-	-	-	-	-	-	1402	-	-
system	system_S5D9	-	-	-	-	-	-	-	-	1506	-
system	system_S7G2	-	-	-	-	-	-	-	-	-	1658
telnet	sf_comms_telnet	-	-	-	-	-	-	-	1400	1400	1400
thread_monitor	sf_thread_monitor	976	1032	1032	1028	1028	1028	1028	1028	1028	1028
uart	sf_uart_comms	980	1056	1056	1004	1004	1004	1004	1004	1004	1004
ul_gx	gx_display_driver_synergy_dave2d	-	-	-	-	-	-	-	-	10488	10488
ul_gx	gx_display_driver_synergy_dave2d_8bit_palette	-	-	-	-	-	-	-	-	5268	5268
wdt	r_wdt	708	748	748	752	752	752	752	752	752	752
wifi	api_init	-	-	-	-	-	-	-	-	-	460

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
wifi	api_ioctl	-	-	-	-	-	-	-	-	-	2816
wifi	api_stack_offload	-	-	-	-	-	-	-	-	-	12880
wifi	api_txrx	-	-	-	-	-	-	-	-	-	716
wifi	api_wmi_rx	-	-	-	-	-	-	-	-	-	1876
wifi	cust_api_init	-	-	-	-	-	-	-	-	-	420
wifi	cust_api_ioctl	-	-	-	-	-	-	-	-	-	5126
wifi	cust_api_stack_offload	-	-	-	-	-	-	-	-	-	3904
wifi	cust_api_stack_txrx	-	-	-	-	-	-	-	-	-	224
wifi	cust_api_txrx	-	-	-	-	-	-	-	-	-	244
wifi	cust_api_wmi_rx	-	-	-	-	-	-	-	-	-	1472
wifi	cust_driver_main	-	-	-	-	-	-	-	-	-	416
wifi	cust_driver_netbuf	-	-	-	-	-	-	-	-	-	1149
wifi	cust_spi_hcd	-	-	-	-	-	-	-	-	-	616
wifi	custom_qcom_api	-	-	-	-	-	-	-	-	-	396
wifi	driver_diag	-	-	-	-	-	-	-	-	-	528
wifi	driver_init	-	-	-	-	-	-	-	-	-	1320
wifi	driver_main	-	-	-	-	-	-	-	-	-	9464
wifi	driver_netbuf	-	-	-	-	-	-	-	-	-	76
wifi	driver_txrx	-	-	-	-	-	-	-	-	-	508
wifi	dset	-	-	-	-	-	-	-	-	-	616
wifi	dset_api	-	-	-	-	-	-	-	-	-	840
wifi	enet_irq_veneer	-	-	-	-	-	-	-	44	44	44
wifi	ether_phy	-	-	-	-	-	-	-	800	800	828
wifi	gt202_debug	-	-	-	-	-	-	-	-	-	896
wifi	gt202_util	-	-	-	-	-	-	-	-	-	308
wifi	gt202_wifi_ctrl	-	-	-	-	-	-	-	-	-	1482
wifi	gt202_wifi_task_thread	-	-	-	-	-	-	-	-	-	1324
wifi	htc	-	-	-	-	-	-	-	-	-	1560
wifi	hw_api	-	-	-	-	-	-	-	-	-	288
wifi	osal	-	-	-	-	-	-	-	-	-	8896
wifi	qcom_api	-	-	-	-	-	-	-	-	-	6070
wifi	qcom_legacy	-	-	-	-	-	-	-	-	-	60

参考資料

グループ	モジュールまたはコンポーネント	S1JA	S124	S128	S3A1	S3A3	S3A6	S3A7	S5D5	S5D9	S7G2
wifi	rcv_aggr	-	-	-	-	-	-	-	-	-	356
wifi	ring_buffer	-	-	-	-	-	-	-	-	-	296
wifi	sf_wifi_gt202	-	-	-	-	-	-	-	-	-	2912
wifi	sf_wifi_gt202_onchip_stack	-	-	-	-	-	-	-	-	-	364
wifi	sf_wifi_gt202_socket	-	-	-	-	-	-	-	-	-	1008
wifi	sf_wifi_nsal_nx	-	-	-	-	-	-	-	-	-	1230
wifi	util	-	-	-	-	-	-	-	-	-	184
wifi	wmi	-	-	-	-	-	-	-	-	-	2852

16.3 暗号化の推定メモリサイズ

表 16.3 各 MCU グループの暗号化 (r_sce) ROM 結果まとめ

lib crypto gcc cm4_s7g2		lib crypto iar cm4_s7g2		libcrypto gcc cm4_s5d9		lib crypto iar cm4_s5d9		lib crypto gcc cm4_s5d5		lib crypto iar cm4_s5d5	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
aes	580	aes	672	aes	580	aes	672	aes	580	aes	672
aes128	1656	aes128	1808	aes128	1656	aes128	1808	aes128	1656	aes128	1808
aes192	1472	aes192	1616	aes192	1472	aes192	1616	aes192	1472	aes192	1616
aes256	1656	aes256	1808	aes256	1656	aes256	1808	aes256	1656	aes256	1808
arc4	376	arc4	376	arc4	376	arc4	376	arc4	376	arc4	376
dsa	48	dsa	48	dsa	48	dsa	48	dsa	48	dsa	48
dsa1024	208	dsa1024	246	dsa1024	208	dsa1024	246	dsa1024	208	dsa1024	246
dsa2048	416	dsa2048	492	dsa2048	416	dsa2048	492	dsa2048	416	dsa2048	492
ecc	48	ecc	48	ecc	48	ecc	48	ecc	48	ecc	48
ecc192	544	ecc192	536	ecc192	544	ecc192	536	ecc192	544	ecc192	536
ecc192hrk	400	ecc192hrk	384	ecc192hrk	400	ecc192hrk	384	ecc192hrk	400	ecc192hrk	384
ecc256	544	ecc256	536	ecc256	544	ecc256	536	ecc256	544	ecc256	536
ecc256hrk	396	ecc256hrk	384	ecc256hrk	396	ecc256hrk	384	ecc256hrk	396	ecc256hrk	384
hash	24	hash	24	hash	24	hash	24	hash	24	hash	24
hrk	5800	hrk	6136	hrk	5800	hrk	6136	hrk	5800	hrk	6136
interface	12	interface	14	interface	12	interface	14	interface	12	interface	14
key	1120	key	1076	key	1120	key	1076	key	1120	key	1076
md5	80	md5	74	md5	80	md5	74	md5	80	md5	74
r_sce	196	r_sce	188	r_sce	196	r_sce	188	r_sce	196	r_sce	188
rsa	504	rsa	488	rsa	504	rsa	488	rsa	504	rsa	488
rsa1024	528	rsa1024	504	rsa1024	528	rsa1024	504	rsa1024	528	rsa1024	504
rsa2048	476	rsa2048	512	rsa2048	476	rsa2048	512	rsa2048	476	rsa2048	512
s1	0	s1	0	s1	0	s1	0	s1	0	s1	0
s3	0	s3	0	s3	0	s3	0	s3	0	s3	0
s5	0	s5	0	s5	740	s5	704	s5	740	s5	704
s7	740	s7	704	s7	0	s7	0	s7	0	s7	0
sha1	76	sha1	80	sha1	76	sha1	80	sha1	76	sha1	80
sha256	76	sha256	80	sha256	76	sha256	80	sha256	76	sha256	80
tdes	108	tdes	108	tdes	108	tdes	108	tdes	108	tdes	108
tdes192	212	tdes192	246	tdes192	212	tdes192	246	tdes192	212	tdes192	246
trng	292	trng	252	trng	292	trng	252	trng	292	trng	252
		sc32									

参考資料

lib crypto gcc cm4_s3a7		lib crypto iar cm4_s3a7		lib crypto gcc cm4_s3a6		lib crypto iar cm4_s3a6		lib crypto gcc cm4_s3a3		lib crypto iar cm4_s3a3	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
aes	580	aes	672	aes	580	aes	672	aes	580	aes	672
aes128	1656	aes128	1808	aes128	1656	aes128	1808	aes128	1656	aes128	1808
aes256	1656	aes256	1808	aes256	1656	aes256	1808	aes256	1656	aes256	1808
ecc	48	ecc	48	ecc	48	ecc	48	ecc	48	ecc	48
ecc192	544	ecc192	536	ecc192	544	ecc192	536	ecc192	544	ecc192	536
ecc192hrk	400	ecc192hrk	384	ecc192hrk	400	ecc192hrk	384	ecc192hrk	400	ecc192hrk	384
ecc256	544	ecc256	536	ecc256	544	ecc256	536	ecc256	544	ecc256	536
ecc256hrk	396	ecc256hrk	384	ecc256hrk	396	ecc256hrk	384	ecc256hrk	396	ecc256hrk	384
hrk	3912	hrk	4238	hrk	3912	hrk	4238	hrk	3912	hrk	4238
interface	12	interface	14	interface	12	interface	14	interface	12	interface	14
key	908	key	872	key	908	key	872	key	908	key	872
r_sce	196	r_sce	188	r_sce	196	r_sce	188	r_sce	196	r_sce	188
s1	0	s1	0	s1	0	s1	0	s1	0	s1	0
s3	380	s3	356	s3	380	s3	356	s3	380	s3	356
s5	0	s5	0	s5	0	s5	0	s5	0	s5	0
s7	0	s7	0	s7	0	s7	0	s7	0	s7	0
trng	292	trng	252	trng	292	trng	252	trng	292	trng	252

lib crypto gcc cm0_s128		lib crypto iar cm0_s128		lib crypto gcc cm0_s124		lib crypto iar cm0_s124	
Name	ROM	Name	ROM	Name	ROM	Name	ROM
aes	440	aes	608	aes	440	aes	608
aes128	392	aes128	436	aes128	392	aes128	436
aes192	392	aes256	436	aes192	392	aes256	436
aes256	392	ecc	56	aes256	392	ecc	56
ecc	56	ecc192	848	ecc	56	ecc192	848
ecc192	624	ecc192hrk	584	ecc192	624	ecc192hrk	584
ecc192hrk	456	ecc256	848	ecc192hrk	456	ecc256	848
ecc256	624	ecc256hrk	584	ecc256	624	ecc256hrk	584
ecc256hrk	448	hrk	1804	ecc256hrk	448	hrk	1804
hrk	1744	interface	24	hrk	1744	interface	24
interface	20	key	764	interface	20	key	764
key	856	r_sce	224	key	856	r_sce	224
r_sce	220	s1	224	r_sce	220	s1	224
s1	148	s3	0	s1	148	s3	0
s3	0	s5	0	s3	0	s5	0

libcrypto iar cm23_s1ja	
Name	ROM
aes	592
aes128	440
aes256	440
ecc	56
ecc192	832
ecc192hrk	576
ecc256	832
ecc256hrk	576
hrk	1808
interface	20
key	756
r_sce	224
s1	200
s3	0
s5	0

lib crypto gcc cm0_s128		lib crypto iar cm0_s128		lib crypto gcc cm0_s124		lib crypto iar cm0_s124	
Name	ROM	Name	ROM	Name	ROM	Name	ROM
s5	0	s7	0	s5	0	s7	0
s7	0	trng	268	s7	0	trng	268
trng	292			trng	292		

libcrypto iar cm23_s1ja	
Name	ROM
s7	0
trng	268

16.4 DSP ライブラリの推定メモリサイズ

表 16.4 各 MCU グループの DSP (Arm) ライブラリ ROM 結果まとめ

DSP_Lib gcc cm4		DSP_Lib iar cm4		DSP_Lib gcc cm0 plus		DSP_Lib ia rcm0 plus		DSP_Lib iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
abs	692	abs	476	abs	196	abs	178	abs	176
add	512	add	368	add	280	add	256	add	254
biquad	6368	biquad	5174	biquad	4324	biquad	4030	biquad	4028
bitreversal	500	bitreversal	490	bitreversal	552	bitreversal	470	bitreversal	466
bitreversal2	192	bitreversal2	192	bitreversal2	80	bitreversal2	80	bitreversal2	80
cfft	14844	cfft	12894	cfft	21404	cfft	17834	cfft	17810
cmplx	5372	cmplx	3858	cmplx	2304	cmplx	1838	cmplx	1828
common	216778	common	216780	common	216778	common	216780	common	216780
const	1056	const	1056	const	1056	const	1056	const	1056
conv	20868	conv	14942	conv	17444	conv	14318	conv	14278
copy	332	copy	216	copy	92	copy	72	copy	64
correlate	10012	correlate	6986	correlate	8728	correlate	7020	correlate	6998
cos	276	cos	268	cos	312	cos	288	cos	288
dct4	328216	dct4	327888	dct4	327532	dct4	327314	dct4	327314
dot	680	dot	460	dot	332	dot	198	dot	190
fill	240	fill	190	fill	56	fill	62	fill	52
fir	18192	fir	13524	fir	10908	fir	9122	fir	8932
float	624	float	560	float	248	float	224	float	224
iir	1932	iir	1464	iir	1168	iir	1016	iir	1006
lms	4508	lms	3326	lms	3080	lms	2584	lms	2554
mat	8088	mat	6214	mat	7276	mat	6364	mat	6312
max	720	max	492	max	248	max	180	max	180
mean	384	mean	334	mean	196	mean	164	mean	152

DSP_Lib gcc cm4		DSP_Lib iar cm4		DSP_Lib gcc cm0 plus		DSP_Lib ia rcm0 plus		DSP_Lib iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
min	720	min	492	min	248	min	180	min	180
mult	700	mult	504	mult	288	mult	264	mult	262
negate	424	negate	334	negate	184	negate	172	negate	168
offset	428	offset	322	offset	272	offset	250	offset	250
pid	276	pid	218	pid	440	pid	348	pid	340
power	488	power	342	power	212	power	178	power	170
q15	360	q15	248	q15	88	q15	84	q15	80
q31	328	q31	256	q31	80	q31	78	q31	74
q7	388	q7	262	q7	88	q7	84	q7	80
rfft	166896	rfft	166578	rfft	167904	rfft	167468	rfft	167462
rms	408	rms	314	rms	324	rms	274	rms	270
scale	880	scale	652	scale	364	scale	340	scale	332
shift	932	shift	636	shift	356	shift	300	shift	292
sin	1136	sin	1076	sin	1332	sin	1264	sin	1264
sqrt	484	sqrt	464	sqrt	560	sqrt	564	sqrt	548
std	620	std	512	std	548	std	460	std	452
sub	516	sub	366	sub	280	sub	258	sub	258
var	644	var	530	var	480	var	362	var	358

16.5 X-Ware の推定メモリサイズ

表 16.5 各 MCU グループの X-Ware ROM 結果まとめ

FileX (fx)

fx gcc cm0plus		fx iar cm0plus		fx gcc cm4		fx iar cm4		fx iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
directory	12932	directory	11764	directory	12100	directory	10890	directory	11246
fault	0	fault	0	fault	0	fault	0	fault	0
file	10088	file	9402	file	9424	file	8238	file	9118
media	7755	media	6806	media	7471	media	6322	media	6592
partition	392	partition	382	partition	376	partition	354	partition	348
ram	220	ram	160	ram	204	ram	156	ram	160
system	986	system	636	system	870	system	592	system	600
trace	0	trace	0	trace	0	trace	0	trace	0
unicode	4720	unicode	4442	unicode	4484	unicode	4272	unicode	4220
utility	5796	utility	5136	utility	5440	utility	4530	utility	4898

FileX エラーチェック (fxe)

fxe gcc cm0plus		fxe iar cm0plus		fxe gcc cm4		fxe iar cm4		fxe iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
directory	1044	directory	980	directory	924	directory	934	directory	866
fault	0	fault	0	fault	0	fault	0	fault	0
file	1616	file	1452	file	1420	file	1404	file	1356
media	1048	media	988	media	916	media	944	media	892
system	228	system	168	system	184	system	138	system	144
unicode	648	unicode	640	unicode	564	unicode	672	unicode	572

GUIX (gx)

gx gcc cm0plus		gx iar cm0plus		gx gcc cm4		gx iar cm4		gx iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
accordion	1440	accordion	1330	accordion	1444	accordion	1326	accordion	1296
animation	2912	animation	2434	animation	2736	animation	2334	animation	2360
binres	4160	binres	3680	binres	3304	binres	3572	binres	3656
brush	40	brush	36	brush	48	brush	36	brush	36

gx gcc cm0plus		gx iar cm0plus		gx gcc cm4		gx iar cm4		gx iar cm23	
Name	ROM	Name	ROM	Name	ROM	Name	ROM	Name	ROM
button	856	button	652	button	756	button	648	button	652
canvas	7508	canvas	6324	canvas	7032	canvas	6104	canvas	6130
checkbox	604	checkbox	608	checkbox	596	checkbox	578	checkbox	590
circular	1256	circular	1170	circular	1240	circular	1130	circular	1130
context	608	context	590	context	588	context	552	context	556
display	116353	display	100020	display	108572	display	92784	display	98766
drop	788	drop	736	drop	756	drop	730	drop	722
horizontal	3240	horizontal	2770	horizontal	2852	horizontal	2662	horizontal	2722
icon	848	icon	774	icon	808	icon	762	icon	762
image	12881	image	12644	image	11933	image	12046	image	12374
line	556	line	510	line	544	line	492	line	496
menu	778	menu	764	menu	842	menu	754	menu	744
monochrome	88	monochrome	84	monochrome	88	monochrome	84	monochrome	84
multi	9158	multi	8048	multi	8182	multi	7520	multi	7880
numeric	472	numeric	450	numeric	460	numeric	436	numeric	434
pixelmap	1576	pixelmap	1486	pixelmap	1592	pixelmap	1472	pixelmap	1430
popup	92	popup	100	popup	96	popup	100	popup	98
progress	886	progress	796	progress	858	progress	798	progress	762
prompt	336	prompt	342	prompt	340	prompt	330	prompt	328
radial	2022	radial	1766	radial	1830	radial	1728	radial	1738
radio	480	radio	486	radio	472	radio	464	radio	472
screen	144	screen	136	screen	144	screen	140	screen	136
scroll	2532	scroll	2326	scroll	2544	scroll	2302	scroll	2222
scrollbar	2208	scrollbar	1992	scrollbar	2152	scrollbar	1950	scrollbar	1958
single	5188	single	4570	single	4836	single	4464	single	4536
slider	1416	slider	1282	slider	1444	slider	1308	slider	1272
sprite	592	sprite	574	sprite	604	sprite	546	sprite	556
string	248	string	230	string	252	string	236	string	220
system	25540	system	24844	system	25428	system	24614	system	24598
text	2432	text	2198	text	2276	text	2096	text	2144
touch	1228	touch	1552	touch	1220	touch	1390	touch	1518

参考資料

gx gcc cm0plus	
Name	ROM
tree	2556
utility	36440
vertical	3252
widget	6800
window	1948

gx iar cm0plus	
Name	ROM
tree	2278
utility	28330
vertical	2812
widget	6008
window	1690

gx gcc cm4	
Name	ROM
tree	2368
utility	34000
vertical	2864
widget	6284
window	1696

gx iar cm4	
Name	ROM
tree	2212
utility	27292
vertical	2686
widget	5824
window	1602

gx iar cm23	
Name	ROM
tree	2226
utility	27914
vertical	2764
widget	5826
window	1638

GUIX エラーチェック (gxe)

gxe gcc cm0plus	
Name	ROM
accordion	304
animation	336
binres	48
brush	168
button	316
canvas	2716
checkbox	412
circular	468
context	1564
display	672
drop	564
horizontal	900
icon	516
image	292
line	324
menu	400
multi	1896
numeric	856
pixelmap	828
progress	868
prompt	704
radial	672
radio	192

gxe iar cm0plus	
Name	ROM
accordion	280
animation	316
binres	40
brush	160
button	286
canvas	2600
checkbox	368
circular	408
context	1492
display	620
drop	532
horizontal	856
icon	420
image	268
line	284
menu	372
multi	1744
numeric	776
pixelmap	772
progress	784
prompt	668
radial	620
radio	160

gxe gcc cm4	
Name	ROM
accordion	288
animation	296
binres	32
brush	160
button	268
canvas	2584
checkbox	372
circular	452
context	1428
display	640
drop	572
horizontal	912
icon	468
image	268
line	308
menu	412
multi	1892
numeric	884
pixelmap	836
progress	856
prompt	704
radial	664
radio	180

gxe iar cm4	
Name	ROM
accordion	264
animation	280
binres	30
brush	148
button	262
canvas	2492
checkbox	354
circular	398
context	1356
display	592
drop	500
horizontal	820
icon	406
image	264
line	276
menu	364
multi	1686
numeric	744
pixelmap	754
progress	760
prompt	664
radial	592
radio	154

gxe iar cm23	
Name	ROM
accordion	272
animation	270
binres	28
brush	164
button	266
canvas	2596
checkbox	362
circular	400
context	1516
display	624
drop	524
horizontal	844
icon	408
image	264
line	284
menu	372
multi	1728
numeric	768
pixelmap	736
progress	780
prompt	664
radial	616
radio	150

gxe gcc cm0plus	
Name	ROM
screen	348
scroll	1060
scrollbar	352
single	1316
slider	916
sprite	472
string	316
system	2192
text	1328
tree	732
utility	968
vertical	928
widget	4340
window	1692

gxe iar cm0plus	
Name	ROM
screen	340
scroll	976
scrollbar	348
single	1248
slider	860
sprite	428
string	292
system	2084
text	1216
tree	688
utility	870
vertical	888
widget	4160
window	1612

gxe gcc cm4	
Name	ROM
screen	320
scroll	1032
scrollbar	344
single	1296
slider	888
sprite	448
string	320
system	2064
text	1284
tree	732
utility	896
vertical	940
widget	4232
window	1608

gxe iar cm4	
Name	ROM
screen	312
scroll	944
scrollbar	328
single	1164
slider	840
sprite	400
string	288
system	1952
text	1188
tree	660
utility	812
vertical	848
widget	4008
window	1536

gxe iar cm23	
Name	ROM
screen	340
scroll	968
scrollbar	348
single	1248
slider	844
sprite	416
string	288
system	2100
text	1216
tree	684
utility	782
vertical	876
widget	4156
window	1592

NetX (nx)

nx gcc cm0plus	
Name	ROM
arp	3812
icmp	1368
igmp	1904
ip	8905
packet	1692
ram	856
rarp	768
system	178
tcp	14976
trace	0
udp	3560

nx iar cm0plus	
Name	ROM
arp	3360
icmp	1258
igmp	2056
ip	7588
packet	1516
ram	880
rarp	724
system	156
tcp	13102
trace	0
udp	3254

nx gcc cm4	
Name	ROM
arp	3520
icmp	1344
igmp	1704
ip	8113
packet	1636
ram	712
rarp	720
system	182
tcp	14392
trace	0
udp	3392

nx iar cm4	
Name	ROM
arp	3138
icmp	1254
igmp	1892
ip	6942
packet	1460
ram	828
rarp	676
system	156
tcp	12438
trace	0
udp	3046

nx iar cm23	
Name	ROM
arp	3230
icmp	1220
igmp	2012
ip	7290
packet	1456
ram	852
rarp	688
system	156
tcp	12694
trace	0
udp	3124

NetX (nxe) エラーチェック

nxe gcc cm0plus	
Name	ROM
arp	868
icmp	284
igmp	676
ip	2604
packet	1000
rarp	232
tcp	2768
udp	1856

nxe iar cm0plus	
Name	ROM
arp	856
icmp	272
igmp	660
ip	2528
packet	880
rarp	228
tcp	2700
udp	1802

nxe gcc cm4	
Name	ROM
arp	796
icmp	248
igmp	600
ip	2476
packet	924
rarp	196
tcp	2624
udp	1724

nxe iar cm4	
Name	ROM
arp	816
icmp	252
igmp	584
ip	2338
packet	828
rarp	200
tcp	2496
udp	1618

nxe iar cm23	
Name	ROM
arp	796
icmp	248
igmp	604
ip	2404
packet	792
rarp	212
tcp	2484
udp	1668

NetX Duo (nxd) IPv4 および IPv6

nxd gcc cm4	
Name	ROM
arp	3884
icmp	1440
icmpv4	832
icmpv6	5488
igmp	1696
invalidate	100
ip	11073
ipv4	1016
ipv6	3788
nd	1200
packet	1704
ram	1168
rarp	768
system	281
tcp	14364
trace	0
udp	2956

nxd iar cm4	
Name	ROM
arp	3460
icmp	1318
icmpv4	780
icmpv6	5092
igmp	1714
invalidate	96
ip	9692
ipv4	826
ipv6	3408
nd	1188
packet	1600
ram	1156
rarp	708
system	240
tcp	12790
trace	0
udp	2756

NetX Duo (nxd) IPv6 のみ

nxd gcc cm4		nxd iar cm4	
Name	ROM	Name	ROM
icmp	264	icmp	250
icmpv6	36	icmpv6	34
ip	184	ip	180
ipv6	3820	ipv6	3426
nd	340	nd	314
tcp	548	tcp	496
udp	612	udp	578

NetX Duo IPv6 エラーチェック (nxde)

nxde gcc cm4		nxde iar cm4	
Name	ROM	Name	ROM
icmp	328	icmp	332
icmpv6	68	icmpv6	68
ip	404	ip	384
ipv6	832	ipv6	812
nd	388	nd	368
tcp	220	tcp	216
udp	608	udp	576

NetX Duo IPv4 エラーチェック (nxe)

nxe gcc cm4		nxe iar cm4	
Name	ROM	Name	ROM
arp	944	arp	976
icmp	248	icmp	252
igmp	688	igmp	704
ip	3092	ip	2924
ipv4	184	ipv4	192
packet	948	packet	852
rarp	196	rarp	200
tcp	2700	tcp	2572
udp	1732	udp	1630

el iar cm23	
Name	ROM
md5	2468

ThreadX (tx)

tx gcc cm0plus	
Name	ROM
block	992
byte	1404
event	1280
initialize	376
isr	8
misra	0
mutex	1720
queue	2036
semaphore	1004
thread	3965
time	40
timer	1416
trace	120

tx iar cm0plus	
Name	ROM
block	936
byte	1332
event	1150
initialize	360
isr	4
misra	0
mutex	1596
queue	1832
semaphore	922
thread	3386
time	40
timer	1230
trace	120

tx gcc cm4	
Name	ROM
block	1004
byte	1352
event	1180
initialize	392
isr	8
misra	0
mutex	1700
queue	1876
semaphore	940
thread	3573
time	40
timer	1368
trace	104

tx iar cm4	
Name	ROM
block	900
byte	1288
event	1080
initialize	280
isr	4
misra	0
mutex	1538
queue	1756
semaphore	896
thread	3358
time	40
timer	1198
trace	104

tx iar cm23	
Name	ROM
block	896
byte	1284
event	1100
initialize	360
isr	4
misra	0
mutex	1544
queue	1776
semaphore	888
thread	3296
time	40
timer	1182
trace	96

ThreadX (txe) エラーチェック

txe gcc cm0plus	
Name	ROM
block	448
byte	488
event	408
mutex	460
queue	596
semaphore	452
thread	912
timer	432

txe iar cm0plus	
Name	ROM
block	404
byte	456
event	372
mutex	432
queue	544
semaphore	408
thread	812
timer	396

txe gcc cm4	
Name	ROM
block	404
byte	444
event	360
mutex	420
queue	548
semaphore	396
thread	824
timer	408

txe iar cm4	
Name	ROM
block	388
byte	436
event	356
mutex	404
queue	532
semaphore	372
thread	792
timer	376

txe iar cm23	
Name	ROM
block	384
byte	432
event	352
mutex	408
queue	524
semaphore	368
thread	764
timer	372

USBX (ux)

ux gcc cm0plus	
Name	ROM
device	4755
host	6930
system	382
trace	0
utility	1784

ux iar cm0plus	
Name	ROM
device	4268
host	6356
system	376
trace	0
utility	1698

ux gcc cm4	
Name	ROM
device	4367
host	6750
system	382
trace	0
utility	1812

ux iar cm4	
Name	ROM
device	4032
host	6180
system	400
trace	0
utility	1632

ux iar cm23	
Name	ROM
device	4184
host	6174
system	376
trace	0
utility	1614

ux_device_class_cdc_acm gcc cm0plus	
Name	ROM
device	1475

ux_device_class_cdc_acm iar cm0plus	
Name	ROM
device	1296

ux_device_class_cdc_acm gcc cm4	
Name	ROM
device	1327

ux_device_class_cdc_acm iar cm4	
Name	ROM
device	1268

ux_device_class_cdc_acm iar cm23	
Name	ROM
device	1270

ux_device_class_cdc_ecm gcc cm0plus	
Name	ROM
device	2326

ux_device_class_cdc_ecm iar cm0plus	
Name	ROM
device	2050

ux_device_class_cdc_ecm gcc cm4	
Name	ROM
device	2102

ux_device_class_cdc_ecm iar cm4	
Name	ROM
device	1962

ux_device_class_cdc_ecm iar cm23	
Name	ROM
device	2010

ux_device_class_hid gcc cm0plus	
Name	ROM
device	1339

ux_device_class_hid iar cm0plus	
Name	ROM
device	1196

ux_device_class_hid gcc cm4	
Name	ROM
device	1187

ux_device_class_hid iar cm4	
Name	ROM
device	1176

ux_device_class_hid iar cm23	
Name	ROM
device	1176

ux_device_class_rndis gcc cm0plus	
Name	ROM
device	3580

ux_device_class_rndis iar cm0plus	
Name	ROM
device	3054

ux_device_class_rndis gcc cm4	
Name	ROM
device	3272

ux_device_class_rndis iar cm4	
Name	ROM
device	2878

ux_device_class_rndis iar cm23	
Name	ROM
device	2994

ux_device_class_storage gcc cm0plus	
Name	ROM
device	7242

ux_device_class_storage iar cm0plus	
Name	ROM
device	6446

ux_device_class_storage gcc cm4	
Name	ROM
device	6926

ux_device_class_storage iar cm4	
Name	ROM
device	6420

ux_device_class_storage iar cm23	
Name	ROM
device	6416

ux_network_driver gcc cm0plus	
Name	ROM
network	760

ux_network_driver iar cm0plus	
Name	ROM
network	770

ux_host_class_audio gcc cm4	
Name	ROM
host	3439

ux_host_class_audio iar cm4	
Name	ROM
host	3154

ux_network_driver iar cm23	
Name	ROM
network	716

参考資料

サポート窓口とお問合せ先

サポート窓口

<https://synergycastle.renesas.com/support>

技術的な質問に関するお問合せ先

<https://www.renesas.com/ja-jp/support/contact.html>

すべての商標および登録商標は、それぞれの所有者に帰属します。

参考資料

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.02	2018.08.08	—	第 1.02 版発行 英文版 Synergy Software Package (SSP) v1.4.0 Software Descriptive Datasheet (資料番号 R01DS0329EU0102、リビジョン Rev.1.02、発行日 2018 年 5 月 18 日) を翻訳

参考資料

Synergy Software Package (SSP) v1.4.0 データシート (参考資料)

発行年月日 2018年8月8日 Rev.1.02

発行 ルネサス エレクトロニクス株式会社
〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■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：<https://www.renesas.com/contact/>

Renesas Synergy™プラットフォーム
Synergy Software Package (SSP) v1.4.0
(参考資料)