

ルネサス RA ファミリ

RA6 シリーズクイックデザインガイド

要旨

このドキュメントでは、よくある質問に回答し、ハードウェア マニュアルを見直さなければ見落とされる可能性のある MCU の細かな点を指摘するものです。本書は、ハードウェアマニュアルの代わりとなるものではなく、多くのエンジニアが自身の設計を開始する時、必要な重要な項目を紹介し、マニュアルを補完することを目的としています。また、アプリケーションの観点から、設計上の決定事項を論じています。

動作確認デバイス

RA6 シリーズ

目次

1. 電源	4
1.1 参考文献	5
2. エミュレータサポート	6
2.1 SWD インタフェース	6
2.2 JTAG インタフェース	7
2.3 SCI を使用したシリアルプログラミングインタフェース	8
2.4 SCI を使用したシリアルプログラミングインタフェース:TrustZone®をサポートするデバイス	9
2.5 マルチエミュレータインタフェース	10
2.6 エミュレータ接続のためのソフトウェア設定	11
2.6.1 SWD および JTAG インタフェース	11
2.6.2 トレースポート	11
3. MCU の動作モード	11
4. オプション設定メモリ	12
4.1 オプション設定メモリレジスタ	13
5. クロック発生回路	14
5.1 リセット条件	15
5.2 クロック周波数要件	15
5.2.1 USB 通信の要件	16
5.2.2 イーサネットコントローラの要件	16
5.2.3 内蔵 ROM またはデータフラッシュのプログラミングと消去に必要な要件	16
5.2.4 SDRAM コントローラの要件	16
5.3 クロック発生回路 (CGC) の低消費電力化	16
5.4 システムクロックコントロールレジスタの書き込み	16
5.5 クロック設定の例	17
5.6 HOCO の精度	17
5.7 フラッシュインタフェースクロック	17

5.8	基板設計	17
5.9	外部水晶振動子の選択	18
6.	リセット要件とリセット回路	18
6.1	端子リセット	19
6.2	パワーオンリセット	20
6.3	独立ウォッチドッグタイマのリセット	20
6.4	ウォッチドッグタイマリセット	20
6.5	電圧監視リセット	20
6.6	ディープソフトウェアスタンバイリセット	20
6.7	ソフトウェアリセット	21
6.8	その他のリセット	21
6.9	コールド/ウォームスタートの決定	21
6.10	リセット要因の特定	21
7.	TrustZone® 有効化	21
7.1	Arm® TrustZone® テクノロジーの実装	21
7.2	TrustZone のエミュレータサポート	22
8.	メモリ	24
8.1	SRAM	25
8.2	スタンバイ SRAM	25
8.3	周辺 I/O レジスタ	26
8.4	内蔵フラッシュメモリ	26
8.4.1	バックグラウンド動作	27
8.4.2	ID コード保護	27
8.4.3	デバイスライフサイクルマネジメント	28
8.4.4	フラッシュブロック保護	28
8.4.5	メモリ保護ユニット	28
8.5	外部メモリ	29
8.5.1	外部 16 ビットメモリデバイスの使用	29
8.5.2	SDRAM 初期化の例	29
8.6	データ整列	31
8.7	エンディアンの制限	31
9.	レジスタ書き込み保護	32
10.	I/O ポートの設定	32
10.1	マルチファンクションピン選択の設計方法	33
10.2	ポートを GPIO として設定して使用する方法	33
10.2.1	内蔵プルアップ	34
10.2.2	オープンドレイン出力	34
10.2.3	ポート駆動能力	34

10.3	ポート周辺機能の設定と使用.....	35
10.4	IRQ ピンの設定と使い方.....	36
10.5	未使用ピン.....	38
10.6	存在しないピン.....	38
10.7	電気的特性.....	38
11.	モジュール停止機能.....	38
12.	割り込み制御ユニット.....	39
13.	低消費電力.....	40
14.	外部バス.....	43
14.1	バス幅とマルチプレクス.....	43
14.2	バス信号の駆動能力.....	43
14.3	バスエラー.....	43
15.	一般的なレイアウトの実践.....	43
15.1	デジタルドメインとアナログドメインの比較.....	43
15.2	高速信号設計の考慮事項.....	44
15.3	信号グループの選択.....	45
16.	参照.....	45
	改訂記録.....	47

1. 電源

RA6 シリーズには、デジタル電源とアナログ電源があります。電源は次のピンを使用します。

表 1. デジタル電源

記号	名称	説明
VCC	電源	電源ピンです。システム電源に接続します。VCC ピンの近くに配置された 0.1 μF コンデンサを介して、VSS に接続します。
VSS	グラウンド	グラウンド
VCL	電源	VCL ピンの近くに 0.1 μF コンデンサを介して、VSS に接続します。
VCL0	電源	VCL0 ピンの近くに 0.1 μF のコンデンサを介して、VSS に接続します。
VBATT	バックアップ電源	バックアップ電源ピンです。VCC 断時、RTC およびサブクロック発振器に電源を供給します。VBATT ピンを使用しない場合は、VCC または VSS に接続してください。
VCC_USB	USB FS 電源	USB フルスピード用の電源ピンです。このピンは VCC に接続します。VCC_USB ピンの近くに配置された 0.1 μF コンデンサを介して、VSS_USB に接続します。
VSS_USB	USB FS グラウンド	USB フルスピード用のグラウンドピンです。このピンを VSS に接続します。
VCC_USBHS ¹	USB HS 電源	USB ハイスピード用の電源ピンです。このピンは VCC に接続します。VCC_USBHS ピンの近くに配置された 0.1 μF コンデンサを介して、VSS1_USBHS または VSS2_USBHS に接続します。
VSS1_USBHS ¹	USB HS グラウンド	USB ハイスピード用のグラウンドピンです。このピンは VSS に接続します。
VSS2_USBHS ¹	USB HS グラウンド	USB ハイスピード用のグラウンドピンです。このピンは VSS に接続します。

注: 1. USB HIGH-SPEED 周辺機器を備えたデバイスのみ。

表 2. アナログ電源

記号	名称	説明
AVCC0	アナログ電源	各モジュールのアナログ電圧供給ピンです。このピンを VCC ピンと同じ電圧に接続します。
AVSS0	アナロググラウンド	各モジュールのアナロググラウンドです。このピンを VSS ピンと同じ電圧に接続します。
VREFHO	12 ビット ADC 基準電圧	AN000~AN002 の 12 ビット A/D コンバータ (ユニット 0) およびサンプルホールド回路の基準電圧入力ピンです。これらの機能を使用しない場合、このピンを AVCC0 に接続してください。
VREFLO	12 ビット ADC 基準電圧	AN000~AN002 の 12 ビット A/D コンバータ (ユニット 0) およびサンプルホールド回路のアナログ基準グラウンドピンです。これらの機能を使用しない場合、このピンを VSS に接続してください。
VREFH	12 ビット ADC & DAC アナログ電源	12 ビット A/D コンバータ (ユニット 1)、AN100~AN102 のサンプルホールド回路、D/A コンバータ用の基準電圧入力ピンです。これらのモジュールのアナログ電源として使用されます。これらの機能を使用しない場合、このピンを AVCC0 に接続してください。

記号	名称	説明
VREFL	12 ビット ADC & DAC アナログ グラウンド	12 ビット A/D コンバータ (ユニット 1)、AN100~AN102 のサンプルホールド回路、D/A コンバータ用の基準電圧入力ピンです。これらのモジュールのアナログ電源として使用されます。これらの機能を使用しない場合、このピンを V _{SS} に接続してください。
AVCC_USBHS ¹	USB HS アナログ電源	USB HIGH-SPEED 用アナログ電源です
AVSS_USBHS ¹	USB HS アナロググランド	USB HIGH-SPEED 用アナロググランドピンです。PVSS_USBHS ピンと短絡してください。
PVSS_USBHS ¹	USB HS PLL 回路グランド	USB HIGH-SPEED 用 PLL 回路のグランドピンです。AVSS_USBHS ピンと短絡してください。
USBHS_RREF ¹	USB HS 電流リファレンス	USB HIGH-SPEED 用基準電源ピンです。AVSS_USBHS ピンと 2.2kΩ 抵抗 (±1%) を介して接続します。

注: 1 USB High-speed 周辺機器を備えたデバイスのみ。

1.1 参考文献

- RA ファミリの電源に関する詳細は、以下のドキュメントを参照してください:
- R01UH0884EJ0100 RA6M1 グループ, RA6M1 グループ ハードウェア編
- R01UH0885JJ0100 RA6M2 グループ, RA6M2 グループ ハードウェア編
- R01UH0886JJ0100 RA6M3 グループ, RA6M3 グループ ハードウェア編
- R01UH0890JJ0110 RA6M4 グループ, RA6M4 グループ ハードウェア編
- R01UH0891JJ0110 RA6M5 グループ, RA6M5 グループ ハードウェア編

Arm Cortex-M4 と Arm Cortex-M33 のデバイスでは、チャプター番号が異なる場合があります。

第 1 章「概要」では、各パッケージの電源ピンと推奨バイパスコンデンサを記載しています。

第 5 章「リセット」では、パワーオンリセットと他のリセットソースとの区別方法について説明します。

「低電圧検出」では、電源の監視に使用する低電圧検出回路の詳細を説明します。また、「オプション設定メモリ」では、起動時に自動的に低電圧検出 0 回路を有効にする方法について説明します。

「バッテリーバックアップ機能」では、RTC とサブクロックの発振器をバッテリーでバックアップする方法について説明します。オンチップ ADC (アナログ デジタル コンバータ) または DAC (デジタル アナログ コンバータ) を使用する場合は、「12 ビット A/D コンバータ (ADC12)」「12 ビット D/A コンバータ (DAC12)」を参照してください。

表 3. RA6 シリーズ ユーザーズマニュアル: ハードウェア

章名	説明
概要	各パッケージの電源ピンをリストアップし、終端抵抗とバイパスの注意事項をしめします。
リセット	パワーオンリセットについて説明し、他のリセットソースと区別する方法について説明します。
電圧検出回路	電源の監視に使用できる低電圧検出回路の詳細を説明します。
低電力モード	低電力モードを使用すると、電源電圧を下げる場合があります。動作モードが電源要件への影響については、本章を参照してください。
バッテリーバックアップ機能	RTC とサブクロックの発振器をバッテリーでバックアップする方法を記載します。
12 ビット A/D コンバータ、12 ビット D/A コンバータ	オンチップ A/D や D/A コンバータを使用する場合、これらの周辺機器のためにフィルタリングされた電源を提供する方法について、これらの章を参照してください。
クロック発生回路	PCB 設計の推奨を含め、利用可能なクロック構成と使用方法について詳細に説明します。

2. エミュレータサポート

RA6 シリーズは、SWD や JTAG 通信を使ったデバッグと、SCI 通信を使ったシリアルプログラミングの両方をサポートしています。本エミュレータを使用することで、デバッグとシリアルプログラミングを簡単に切り替えることができます。

SWD または JTAG エミュレータインタフェースは、ARM 標準の 10 ピンまたは 20 ピンソケットに接続する必要があります。SCI 通信を使用したシリアルプログラミング用に MD、TXD、RXD ピンが追加されています。

TrustZone®をサポートするデバイスでは、デバッグとシリアルプログラミングを使用するため、基板上で P300/SWCLK/TCK ピンと P201/MD ピンを有線 OR 回路での接続をお勧めします。

エミュレータは、製品開発やプロトタイプングには便利ですが、設計が生産に移行すると不要になる場合があります。エミュレータが不要になった場合は、関連する MCU ユーザーズマニュアル ハードウェア編の「未使用ピンの取り扱い」に従ってポートを設定してください。本書の 10.5 章も参照してください。

2.1 SWD インタフェース

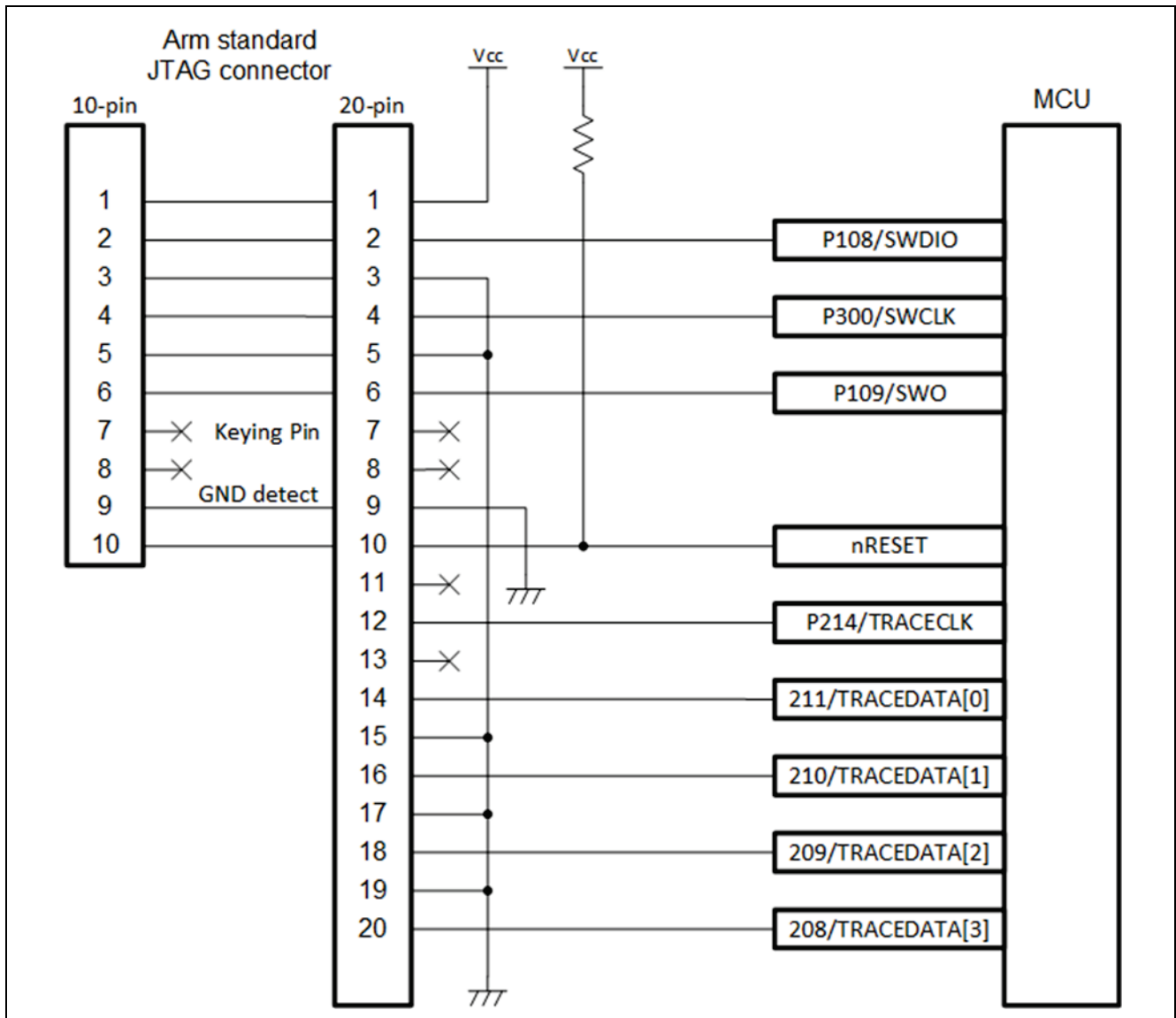


図 1. SWD インタフェースの接続

注 1: ユーザシステムのリセット回路の出力は、オープンコレクタである必要があります。

2.2 JTAG インタフェース

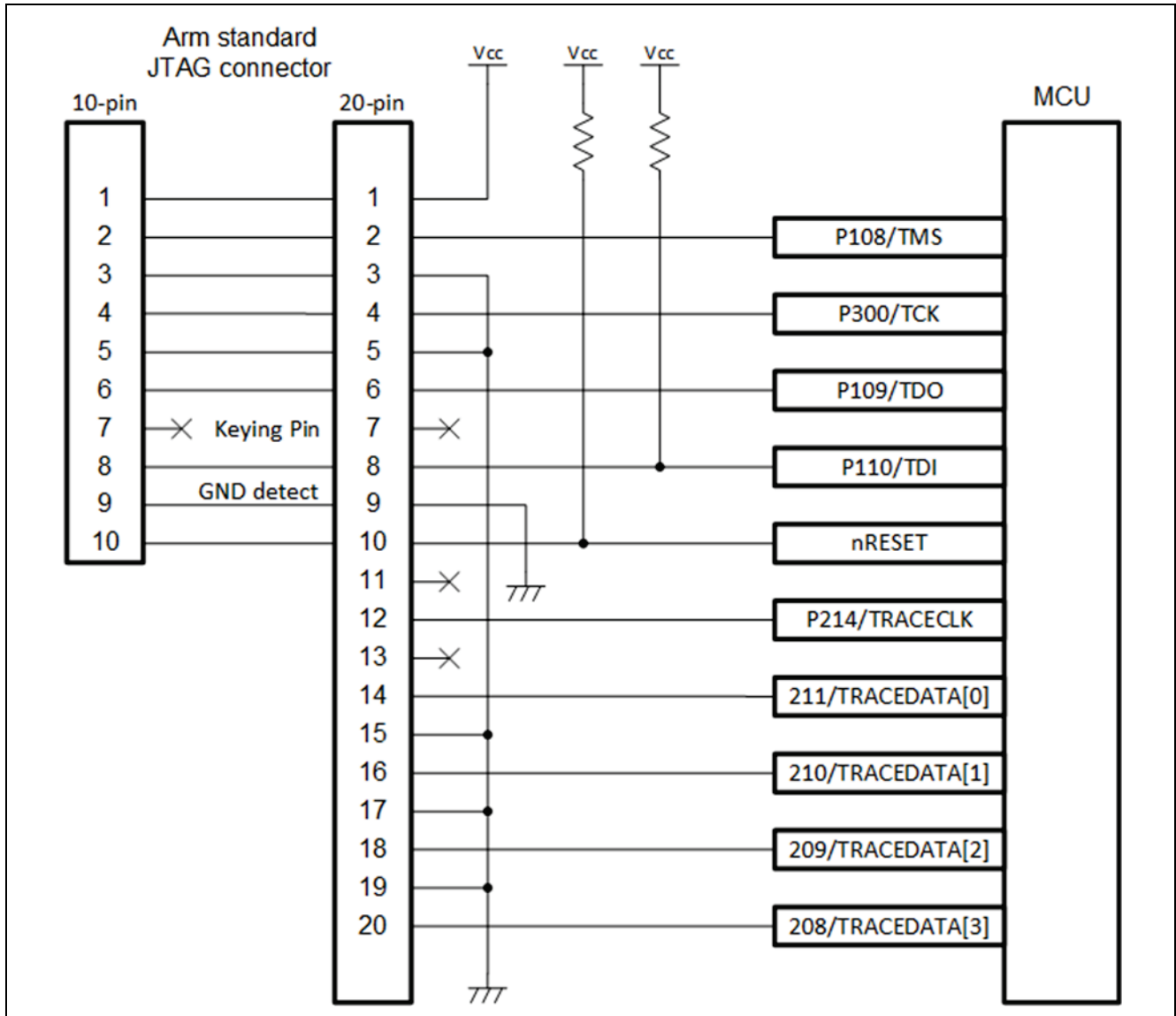


図 2. JTAG インタフェースの接続

- 注 : 1. ユーザシステムのリセット回路の出力はオープンコレクタである必要があります。
 2. TMS、TCK、TDO、TDI に 4.7 k Ω ~ 10 k Ω のプルアップを使用します。

2.3 SCI を使用したシリアルプログラミングインタフェース

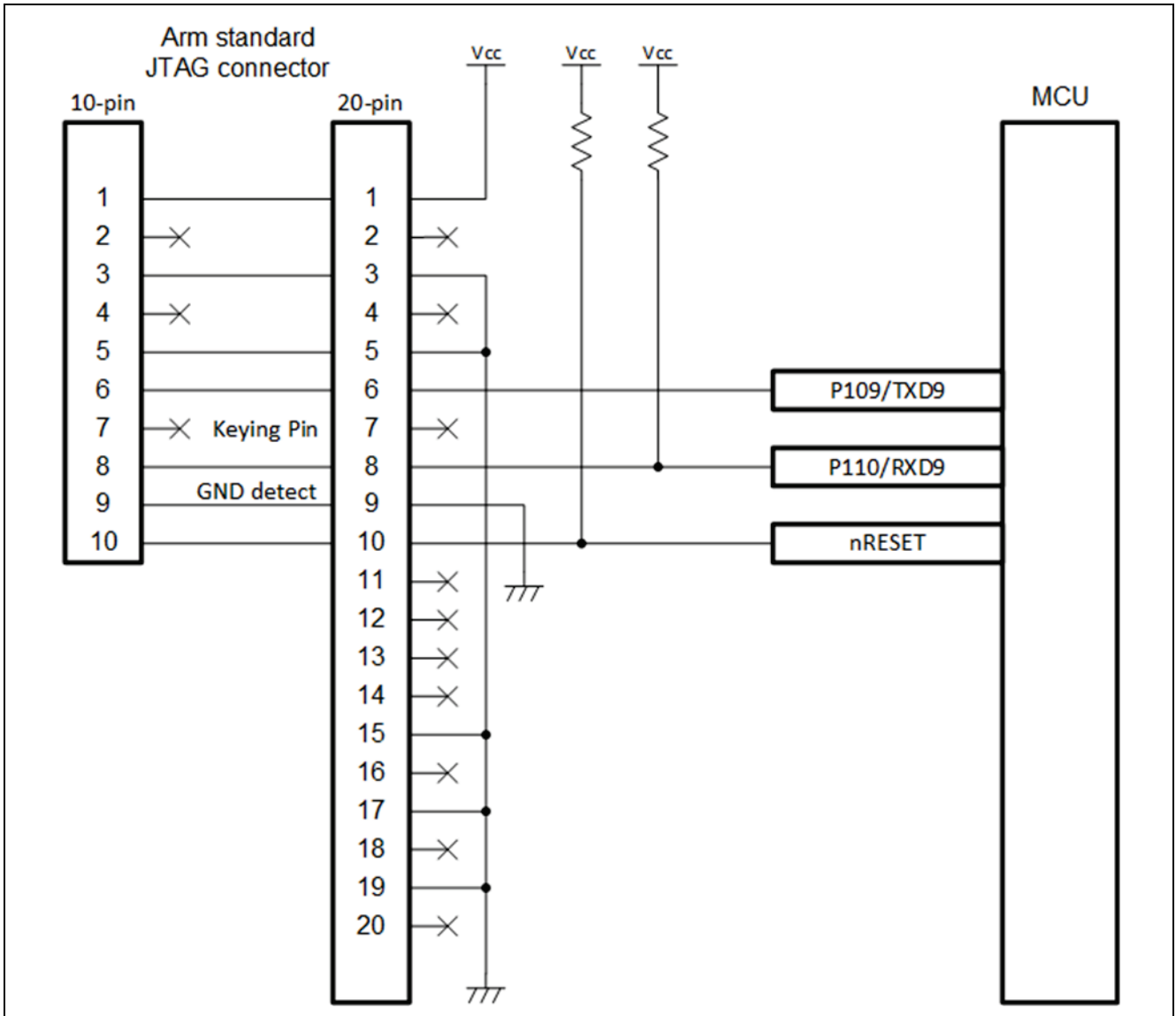


図 3. SCI を使用したシリアルプログラミングインタフェース

注：1. ユーザシステムのリセット回路出力は、オープンコレクタである必要があります。

2.4 SCI を使用したシリアルプログラミングインタフェース:TrustZone®をサポートするデバイス

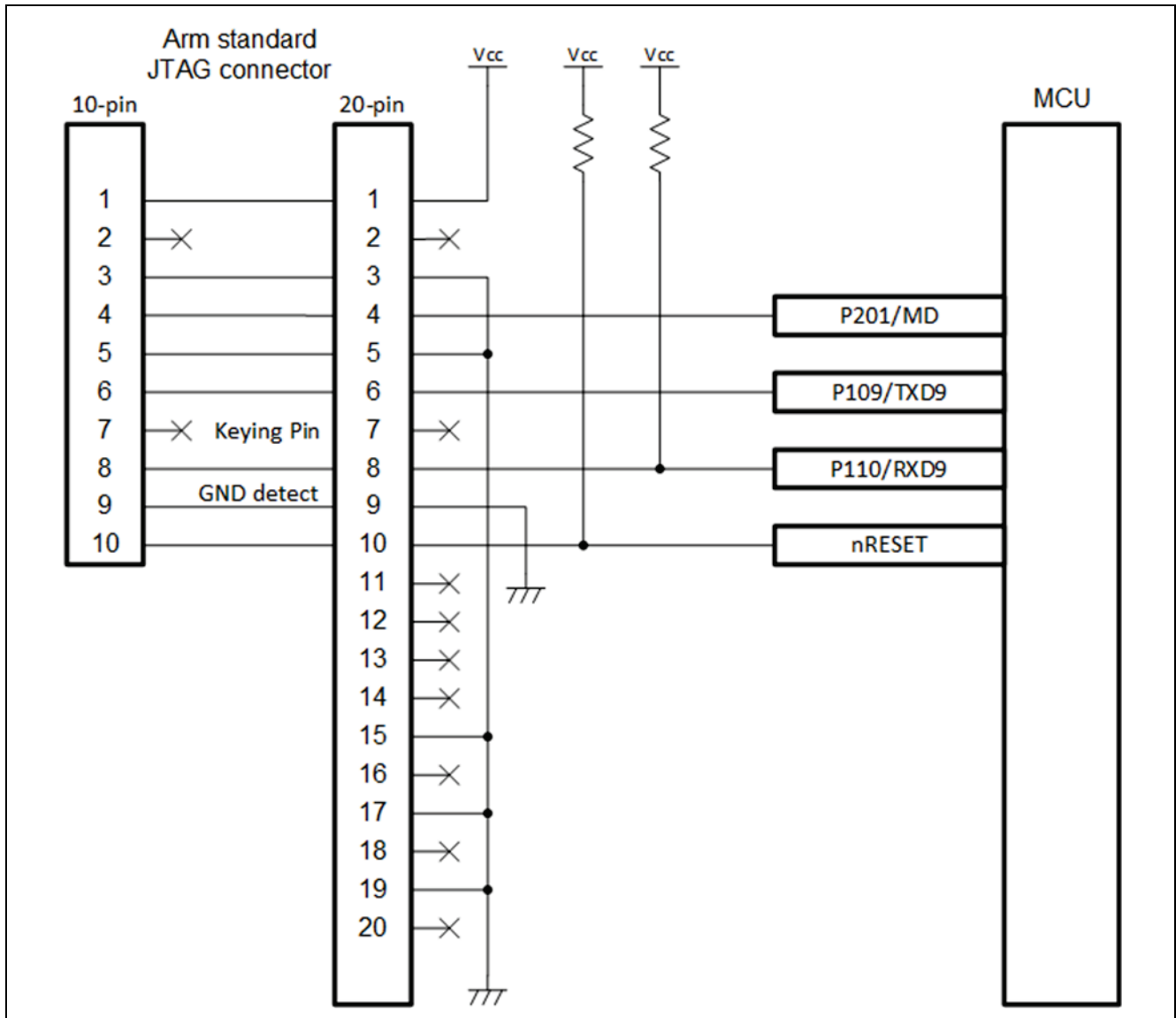


図 4. SCI を使用したシリアルプログラミングインタフェース:
TrustZone® サポート接続を備えたデバイス

- Notes: 1. ユーザシステムのリセット回路の出力はオープンコレクタである必要があります。
2. P201/MD は、TrustZone® IDAU バウンダリ レジスタ設定のプログラミングをサポートするために、Arm 標準 JTAG コネクタのピン 4 に接続する必要があります。

2.5 マルチエミュレータインタフェース

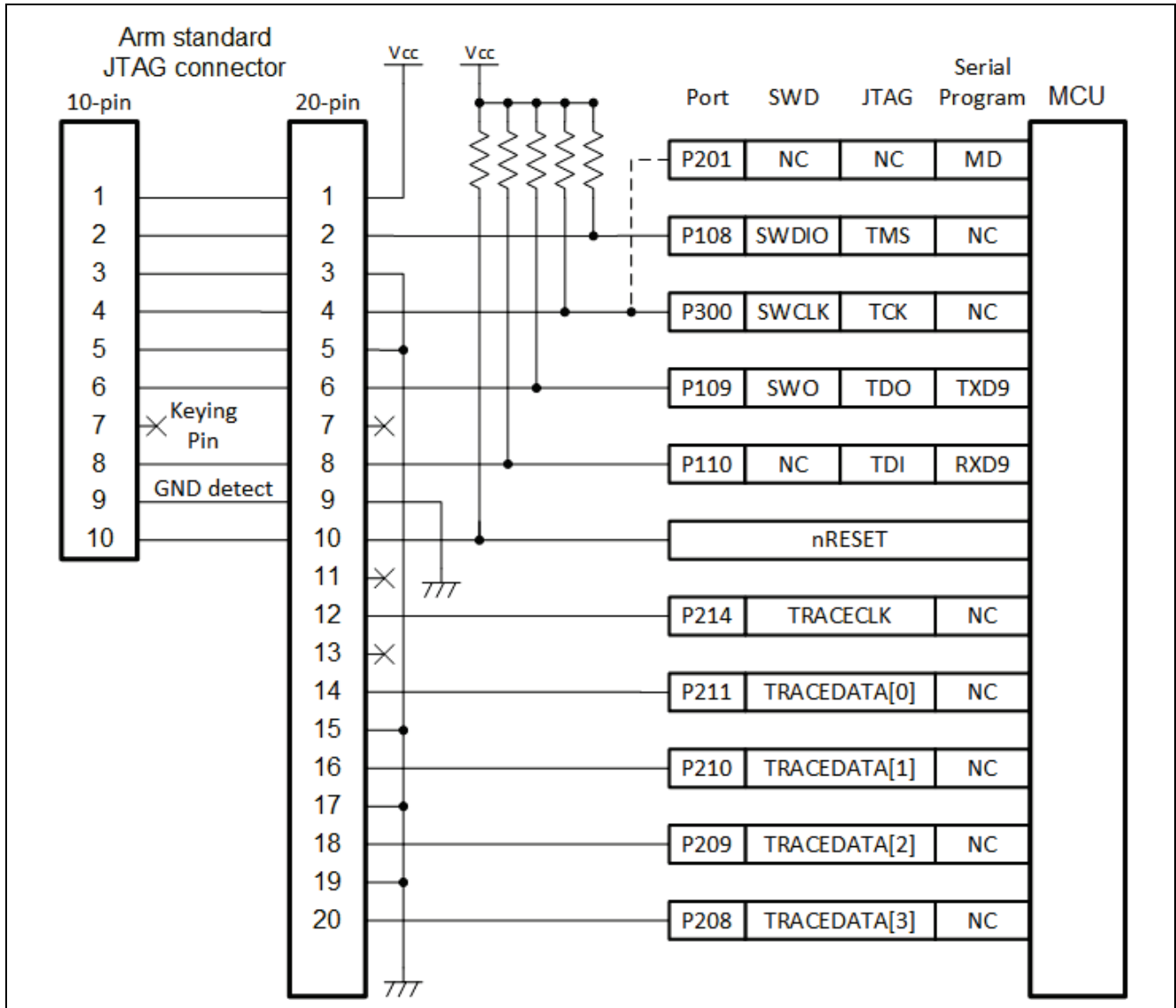


Figure 1. Multiple Emulator Interface Connections

- 注: 1. ターゲットのリセット回路は、オープン コレクタにしてください。nRESET 信号をプルアップしてください。パワーオンリセット回路の動作に影響を与えるため、この信号にコンデンサを接続しないでください。
2. TMS、TCK、TDO、TDI には 4.7kΩ ~ 10kΩ のプルアップを使用する。
3. TrustZone®対応機器でデバッグとシリアルプログラミングを併用する場合は、P201/MD と P300/SWCLK/TCK をワイヤード OR 回路で接続してください

2.6 エミュレータ接続のためのソフトウェア設定

2.6.1 SWD および JTAG インタフェース

SWD ピンと JTAG ピンは、リセット後のデフォルト状態です。

表 4 SWD/JTAG ピン

ピン	P108	P109	P110	P300
機能	TMS/SWDIO	TDO/SWO	TDI	TCK/SWCLK

2.6.2 トレースポート

4ビットのトレースポートインタフェースユニット (TPIU) とシリアルワイヤ出力 (SWO) は、RA6 デバイスでトレース出力を提供します。トレースポートおよびクロックは、デバッグスクリプトで使用する前に有効にする必要があります。トレースポート機能を使用する場合は、トレースピンを他の機能に使用しないでください。

表 5. トレースポート

ピン	P208	P209	P210	P211	P214
機能	TDATA3	TDATA2	TDATA1	TDATA0	TCLK

SEGGER J-Trace Pro を使用した RA6M3 上のトレースポートの使用例については、以下のリンクを参照してください。

https://wiki.segger.com/J-Link_Renesas_RA6M3

トレースポートは、ルネサス FSP の Pin コンフィギュレータを使用して実行時に有効にすることもできますが、この場合、一部のトレースデータが失われる可能性があります。

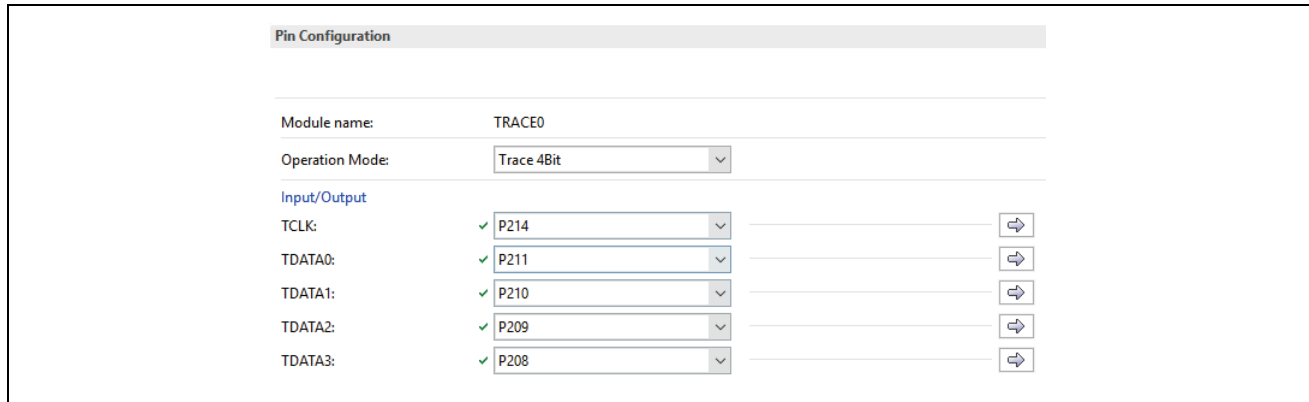


図 6. FSP コンフィギュレータを使用した時のトレースポートの有効化

3. MCU の動作モード

RA6 シリーズはリセット後に、シングルチップモードまたは SCI / USB ブートモードのいずれかのモードに入ることができます。ブートモードは、MD ピンによって選択されます。:

表 6. リセット後に有効な動作モード

動作モード	MD	内蔵フラッシュメモリ	外部バス
シングルチップモード	1	有効	無効
SCI/USB ブートモード	0	有効	無効

モード設定ピンによって決定される動作モードの移行。

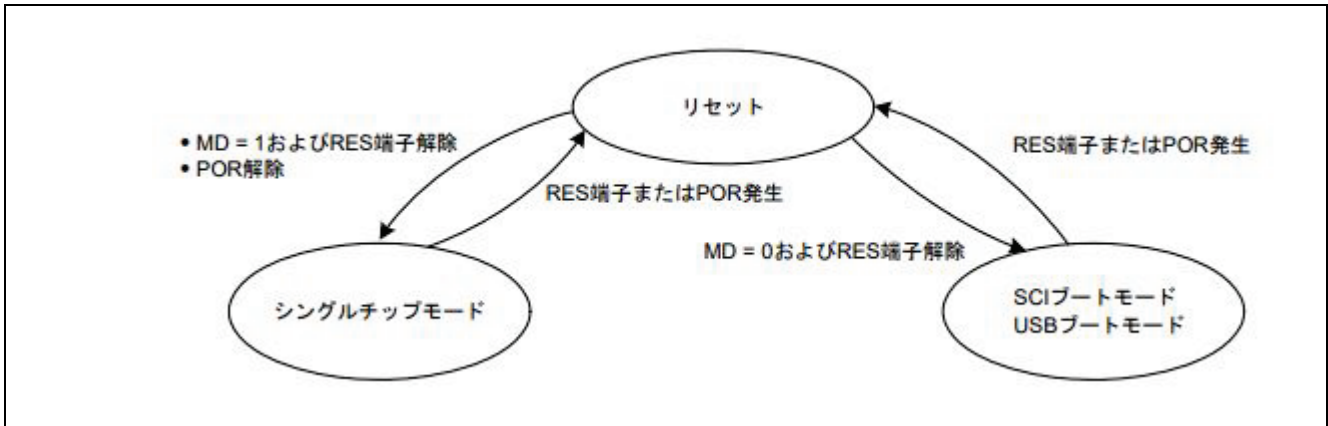


図7. モード設定ピンレベルと動作モード

典型的な MCU ブートモード回路には、MD ピンを VCC またはグラウンドに接続するためのジャンパと複数の抵抗で構成されます。

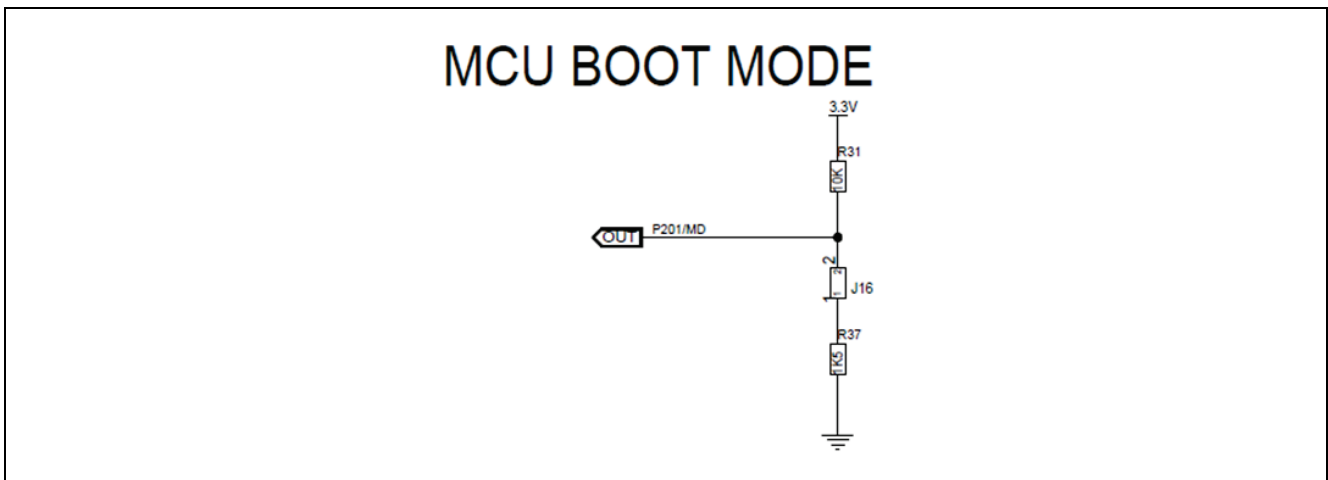


図8. MCU ブートモード選択の典型的な回路

4. オプション設定メモリ

オプション設定用メモリは、リセット後の MCU の状態を決定するものです。フラッシュメモリのコンフィギュレーション設定領域とプログラムフラッシュ領域に割り当てられています。設定方法は、2つの領域で異なります。Cortex-M33 ベースのデバイスでは、オプション設定メモリのサイズやレイアウトが異なる場合があります。

レジスタは、ユーザーズマニュアル ハードウェア編の「オプション設定メモリ」で詳細に説明されています。

これらのレジスタは、コードフラッシュメモリの一部を占めているため、お客様がこれらのアドレスに誤ってデータを保存(上書き)する可能性があります。

ユーザは、これらの場所に不要なデータが書き込まれないように注意する必要があります。そうでないと、チップの予期しない動作が発生する可能性があります。たとえば、フラッシュオプションレジスタの設定により、リセット直後に独立ウォッチドッグタイマ (IWDT) が有効になることがあります。プログラム ROM に誤って格納されたデータがオプション設定メモリレジスタと重なる場合、IWDT が誤ってオンになる可能性があります。これにより、デバッガとボードとの通信に問題が発生します。

下図は、Cortex-M4 デバイスである RA6M3 のオプション機能選択レジスタで構成されるオプション設定メモリです。

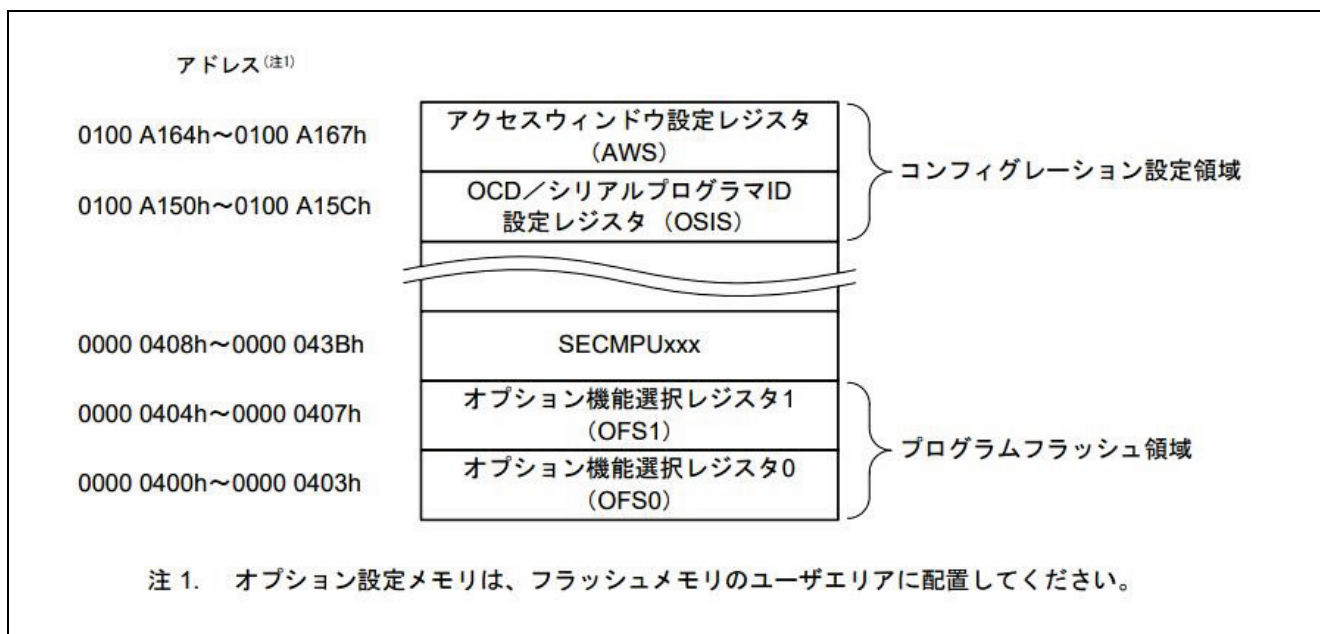


図 9. RA6M3 のオプション機能選択レジスタ

4.1 オプション設定メモリレジスタ

以下はオプション設定メモリレジスタの概要です。起動前に正しく設定されていることを確認してください。

- OFS0 レジスタ
 - 独立ウォッチドッグタイマ (IWDT) 自動起動
 - IWDT タイムアウト、周波数、ウィンドウ表示、割り込みタイプ、および低電力モードの動作
 - ウォッチドッグタイマ (WDT) 自動起動
 - WDT タイムアウト、周波数、ウィンドウ表示、および割り込みタイプ
- OFS1 レジスタ
 - リセット後の LVDO 有効化
 - リセット後の HOCO 起動

ルネサス FSP コンフィギュレータは、以下の RA6M3 MCU の図で示されるように、BSP 設定でオプションメモリの設定をサポートしています。

Summary BSP Clocks Pins Interrupts Event Links Stacks Components		
問題 コンソール プロパティ スマート・ブラウザー 検索		
EK-RA6M3		
Settings	プロパティ	値
	> R7FA6M3AH3CFC	
	> RA6M3	
	▼ RA6M3 Family	
	▼ OFS0 register settings	
	> Independent WDT	
	▼ WDT	
	Start Mode Select	Stop WDT after a reset (register-start mode)
	Timeout Period	16384 cycles
	Clock Frequency Division Ratio	128
	Window End Position	0% (no window end position)
	Window Start Position	100% (no window start position)
	Reset Interrupt Request	Reset
	Stop Control	Stop counting when entering Sleep mode
	▼ OFS1 register settings	
	Voltage Detection 0 Circuit Start	Voltage monitor 0 reset is disabled after reset
	Voltage Detection 0 Level	2.80 V
	HOCO Oscillation Enable	HOCO oscillation is disabled after reset
	▼ MPU	
	Enable or disable PC Region 0	Disabled
	PC0 Start	0xFFFFFFFF
	PC0 End	0xFFFFFFFF
	Enable or disable PC Region 1	Disabled
	PC1 Start	0xFFFFFFFF
	PC1 End	0xFFFFFFFF
	Enable or disable Memory Region 0	Disabled
	Memory Region 0 Start	0x00FFFFFF
	Memory Region 0 End	0x00FFFFFF
	Enable or disable Memory Region 1	Disabled
	Memory Region 1 Start	0x200FFFFC
	Memory Region 1 End	0x200FFFFF

図 10. RA6M3 MCU の FSP 設定におけるオプションメモリ設定

5. クロック発生回路

RA6 シリーズには 6 つの主要な発振器があります。これらのうち 5 つは、メインシステムクロックのソースとして使用することができます。残りの 1 つの発振器は独立ウォッチドッグタイマに割り当てられています。一般的なシステムでは、外部の水晶やクロックでメインクロックが駆動されます。この入力は PLL に送られ、PLL クロックに乗算され、メインシステムクロック (ICKL)、フラッシュクロック、周辺モジュールクロック、外部バスクロック、トレースクロック、USB クロックに分周されます。ICKL は、さらに CPU クロック、DMAC クロック、内部フラッシュクロック、SDRAM クロックに使用されます。クロック発生回路のブロック図については、ユーザーズマニュアル ハードウェア編の「クロック発生回路」を参照してください。

各クロックには、特定の許容範囲とタイミング値があります。周波数とクロックタイミングの仕様については、ユーザーズマニュアル ハードウェア編の「電気的特性」の「AC 特性」を参照してください。各クロック周波数間の関係については、ユーザーズマニュアル ハードウェア編の「クロック発生回路」を参照してください。

表 7. RA6 シリーズ オシレータ

発振器	入力ソース	周波数	主な用途
メインクロック	外部のセラミック発振器、水晶振動子 または 外部クロック	8 MHz to 24 MHz 最大 24 MHz	PLL 入力、メインシステムクロック、CLKOUT、CAN クロック、CAC クロック

サブクロック (SOSC)	外部水晶発振子	32.768 kHz	リアルタイム クロック、低電力モードのシステム クロック、CLKOUT、AGT クロック、CAC クロック
高速オンチップオシレータ (HOCO)	内蔵発振器	16/18/20 MHz	PLL 入力、メインシステムクロック、CLKOUT、CAC クロック
中速オンチップオシレータ (MOCO)	内蔵発振器	8 MHz	起動時のシステムクロック、CLKOUT、CAC クロック
低速オンチップオシレータ (LOCO)	内蔵発振器	32.768 kHz	低消費電力モード時、メイン発振停止検出中のメインシステムクロック、AGT クロック、CAC クロック、リアルタイムクロック
独立ウォッチドッグ (IWDG)	内蔵発振器	15 kHz	独立ウォッチドッグタイマ クロック

5.1 リセット条件

RA6 シリーズは、リセット後、中速オンチップオシレータ (MOCO) をメインクロックソースとして動作を開始します。リセット時に、メイン発振器と PLL はデフォルトでオフになっています。HOCO と IWDG は、オプション設定メモリの設定に応じてオンまたはオフにすることができます。(4章を参照)

5.2 クロック周波数要件

ICLK は常に BCLK 以上である必要があります。最小と最大の周波数を以下の表に示します。外部および内部クロックソースの仕様を含め、MCU ユーザーズマニュアル ハードウェア編の「クロック発生回路」の「概要」に詳細が記載されています。さらに詳細は、MCU ユーザーズマニュアル ハードウェア編の「電気的特性」の「AC 特性」にも記載されています。

表 8. Arm Cortex-M4 MCU 内部クロックの周波数範囲

	ICLK ¹	PCLKA ¹	PCLKB	PCLKC	PCLKD
最大周波数 [MHz]	120	120	60	60	120
最小周波数 [MHz]	-	-	-	-	-

¹ イーサネットコントローラを使用する場合、ICLK と PCLKA は 12.5MHz 以上とし、周波数は同じとすること

	FCLK ¹	BCLK	UCLK	CANCLK	SDCLK
最大周波数 [MHz]	60	120	48	24	120
最小周波数 [MHz]	-	-	48	8	-

¹ ROM やデータフラッシュの書き込みや消去を行う場合、FCLK は 4MHz 以上の周波数で動作する必要があります。

表 9. Arm Cortex-M33 MCU の内部クロックの周波数範囲

	ICLK ¹	PCLKA ¹	PCLKB	PCLKC	PCLKD
最大周波数 [MHz]	200	100	50	50	100
最小周波数 [MHz]	-	-	-	-	-

¹ イーサネットコントローラを使用する場合、ICLK と PCLKA は 12.5MHz 以上とし、周波数は同じとすること

	FCLK ¹	BCLK	USBCLK	CANCLK	OCTACKL
最大周波数 [MHz]	50	100	48	24	200
最小周波数 [MHz]	-	-	48	8	-

¹ ROM やデータフラッシュの書き込みや消去を行う場合、FCLK は 4MHz 以上の周波数で動作する必要があります。

5.2.1 USB 通信の要件

RA ファミリの一部にある USB 2.0 フルスピードモジュール (USBFS) および USB 2.0 ハイスピードモジュール (USBHS) では、48MHz の USB クロック信号 (UCLK または USBCLK) が必要です。USB を使用、且つ、PLL のクロックソースを HOCO が選択された場合、周波数ロックループ (FLL) 機能を有効にする必要があります。

USB を使用する場合、メインクロック発振器の周波数は以下の選択肢に限られます。

8 MHz、10 MHz、12 MHz、15 MHz、16 MHz、20 MHz、24 MHz

これは、クロック発生回路で利用可能な特定の分周比から USB モジュールに必要な 48 MHz クロックを生成するためです。

Arm® Cortex-M4 デバイスの場合、使用される分周器は、SCKDIVCR2 レジスタ内の UCK ビットの設定によって決まります。

Arm Cortex-M33 デバイスの場合、使用される分周器は、USBCKDIVCR レジスタ内の USBCKDIV ビットの設定によって決まります。

5.2.2 イーサネットコントローラの要件

Arm Cortex-M4 デバイスの場合、イーサネット コントローラ (EtherC) とイーサネット DMA コントローラ (EDMAC) を使用する場合、PCLKA (イーサネット) は ICLK と同じである必要があります、両方の周波数は 12.5 MHz から 120 MHz の範囲内である必要があります。

Arm Cortex-M33 デバイスで、Ethernet コントローラ (EtherC) とイーサネット DMA コントローラ (EDMAC) を使用する場合、PCLKH (イーサネット) は 12.5MHz~100MHz の範囲である必要があります。

5.2.3 内蔵 ROM またはデータフラッシュのプログラミングと消去に必要な要件

内蔵 ROM やデータフラッシュのプログラミングや消去を行うには、FCLK が 4MHz 以上である必要があります。

5.2.4 SDRAM コントローラの要件

SDCLK は、BCLK から供給されます。SDCLK をシステムクロック (ICLK) より高い周波数に設定しないでください。

5.3 クロック発生回路 (CGC) の低消費電力化

省電力のために、使用しないクロック (例: BCLK) の分周器を可能な限り最高値に設定し、クロックを使用しない場合は、適切なレジスタを設定してクロックが停止していることを確認してください。各クロックソースを制御するためのレジスタを下表に示します。

表 10. クロックソースコンフィギュレーションレジスタ

発振器	レジスタ	説明
メインクロック	MOSCCR	メインクロック発振器の起動/停止
サブクロック	SOSCCR	サブクロック発振器の起動/停止
高速オンチップ (HOCO)	HOCOGR	HOCO の起動/停止
中速オンチップ (MOCO)	MOCOGR	MOCO の起動/停止
低速オンチップ (LOCO)	LOCOGR	LOCO の起動/停止

5.4 システムクロックコントロールレジスタの書き込み

システムクロック分周コントロールレジスタ (SCKDIVCR)、システムクロック分周コントロールレジスタ 2 (SCKDIVCR2)、システムクロックソースコントロールレジスタ (SCKSCR) の個々のビットフィールドに書き込む際には注意が必要です。

SCKDIVCR、SCKDIVCR2 の値を低い分周比から高い分周比に変更する場合は、変更前に 750ns 以上待機してしてください。

高い分周比から低い分周比に変更する場合は、変更後 250ns 以上経過してから次の処理を開始してください。

SCKSCR の値を PLL から別のクロックソースに変更する場合は、変更後 750ns 以上待ってください。PLL 以外のクロックソースから PLL に変更する場合は、変更後 250ns 以上待ってから次の処理を開始してください。

待機時間を測定する推奨方法は、ソフトウェアで命令サイクルをカウントして行う方法です。必要な待機時間が経過するよう、使用条件を考慮してください。

5.5 クロック設定の例

ルネサス FSP では、以下のような RA6 シリーズ用のシンプルで視覚的なクロックコンフィギュレーションツールを提供しています。

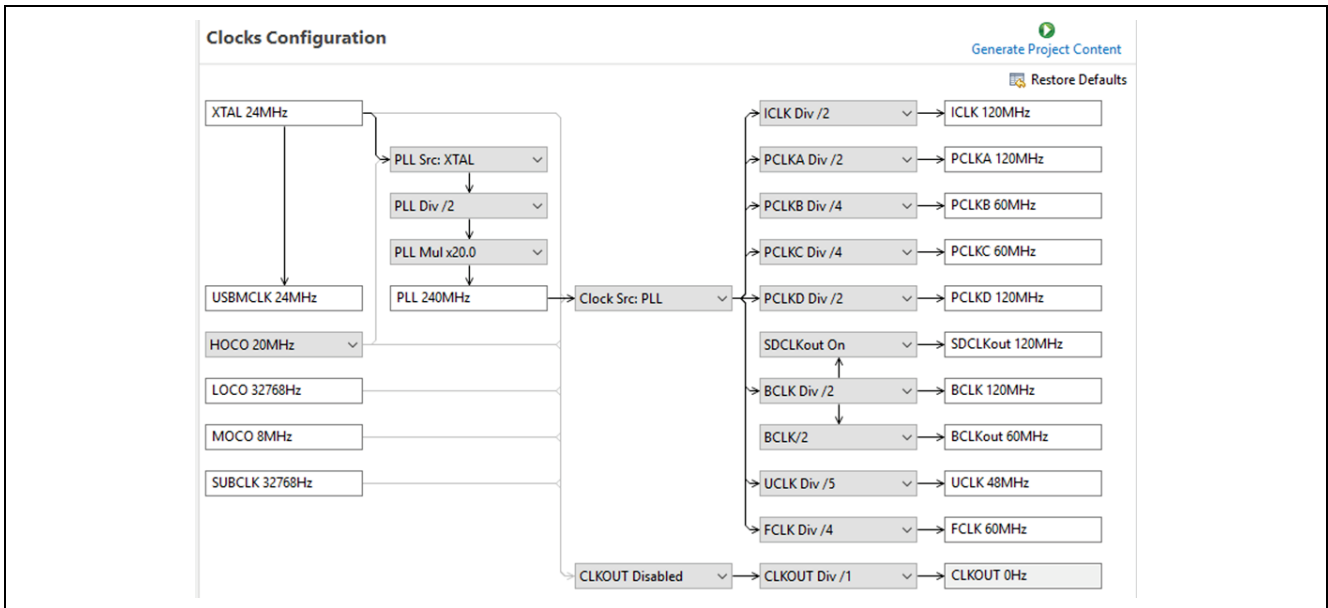


図 11. ルネサス FSP コンフィギュレータによるクロック設定

5.6 HOCO の精度

内部高速オンチップ発振器 (HOCO) は、16 MHz、18 MHz、20 MHz で動作し、精度は±2%以上です。HOCO の精度は、周波数ロックループ (FLL) 機能を有効にすることで改善でき、クロックの精度は±0.3%以上になります。詳細については、マニュアルの電気仕様を参照してください。

HOCO は PLL 回路の入力として使用することができます。HOCO をこのように使用する場合、外部発振器は必要ありません。これは、PCB 設計で部品数を減らす必要がある場合や、スペースの制約などがある場合には利点となります。ただし、クロックの精度による性能のトレードオフや制限があるため、アプリケーションに応じて評価する必要があります。

5.7 フラッシュインタフェースクロック

フラッシュインタフェースクロック (FCLK) は、内部フラッシュ (ROM およびデータフラッシュ) のプログラムや消去、データフラッシュからの読み取り時の動作クロックとして使用されます。したがって、FCLK の周波数設定は、データフラッシュからの読み取りにかかる時間に直接影響します。もしユーザのプログラムがデータフラッシュから読み取りを行うか、内部フラッシュのプログラムや消去を行う場合には、最大の FCLK 周波数を使用することが推奨されます。

FCLK の周波数設定は、ROM からの読み取りや RAM への読み書きには影響を与えないことに注意してください。

5.8 基板設計

CGC の使用方法とボード設計の推奨事項については、ユーザーズマニュアル ハードウェア編の「クロック発生回路 (CGC)」の「使用上の注意」を参照してください。

一般的に、水晶振動子とその負荷容量を MCU のクロックピン (XTAL / EXTAL、XCIN / XCOUT) にできるだけ近くに配置してください。水晶振動子と MCU の間に他の信号を配線しないでください。また、使用する VIA の数を最小限に抑えてください。

5.9 外部水晶振動子の選択

外部の水晶振動子は、メインクロック ソースとして使用することができます。外部水晶振動子は、MCU の EXTAL と XTAL にピンに接続されます。外部水晶振動子の周波数は、メインクロック発振器の周波数範囲内である必要があります。

水晶振動子の選択は、各基板設計に大きく依存します。RA6 シリーズでの使用に適した水晶振動子の選択肢が豊富にあるため、選択された水晶振動子の電気的特性を慎重に評価し、具体的な実装要件を決定する必要があります。

以下の図は、水晶振動子接続の典型的な例を示しています。

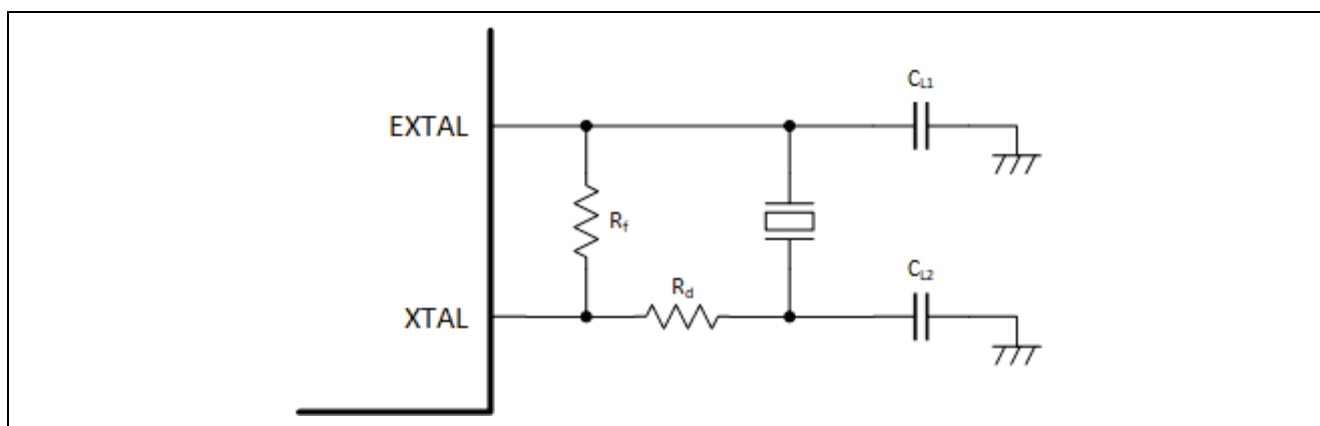


図 12. 水晶振動子の接続例

水晶振動子と関連するコンデンサを選択する際には、慎重な評価が必要です。水晶振動子メーカーが推奨する場合、外部フィードバック抵抗 (R_f) とダンピング抵抗 (R_d) を追加することができます。

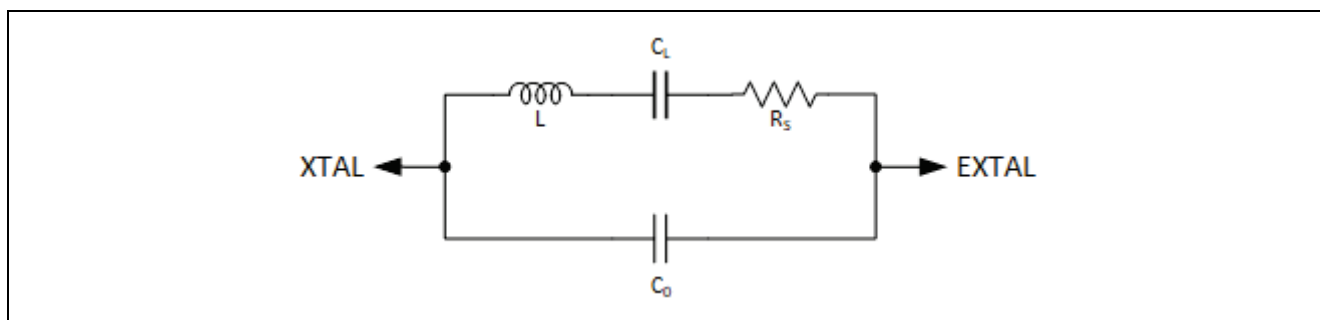


図 13. 水晶振動子の等価回路

CL1 および CL2 のコンデンサ値の選択は、内部クロックの精度に影響を与えます。CL1 および CL2 の値の影響を理解するためには、上記の図の水晶振動子の等価回路を使用して回路をシミュレーションする必要があります。より正確な結果を得るためには、水晶振動子コンポーネント間の配線に関連する浮遊容量も考慮してください。

6. リセット要件とリセット回路

リセットには 14 種類あります。これらのリセットは、Arm® Cortex-M4 デバイスと Arm® Cortex-M33 デバイスの間でわずかに異なります。

表 11. Arm® Cortex-M4 デバイスのリセット

リセット名	ソース
ピンリセット	RES 端子への入力電圧が Low
パワーオンリセット	VCC 上昇 (電圧検出: VPOR)
独立ウォッチドッグタイマリセット	IWDT がアンダーフローまたはリフレッシュエラー
ウォッチドッグタイマリセット	WDT のアンダーフローまたはリフレッシュエラー
電圧監視 0 リセット	VCC の下降 (監視電圧: Vdet0)
電圧監視 1 リセット	VCC の下降 (監視電圧: Vdet1)
電圧監視 2 リセット	VCC の下降 (監視電圧: Vdet2)
SRAM パリティエラーリセット	SRAM パリティ エラー検出
SRAM ECC エラーリセット	SRAM ECC エラー検出
バスマスタ MPU エラーリセット	バスマスタ MPU エラー検出
バススレーブ MPU エラーリセット	バススレーブ MPU エラー検出
スタックポインタエラーリセット	スタックポインタエラー検出
ディープソフトウェアスタンバイリセット	割り込みによるディープソフトウェアスタンバイモードの解除
ソフトウェアリセット	レジスタ設定

表 12. Arm® Cortex-M33 デバイスの s リセット

リセット名	ソース
ピンリセット	RES 端子への入力電圧が Low
パワーオンリセット	VCC 上昇 (電圧検出: VPOR)
独立ウォッチドッグ タイマーリセット	IWDT がアンダーフローまたはリフレッシュエラー
ウォッチドッグ タイマーリセット	WDT のアンダーフローまたはリフレッシュエラー
電圧監視 0 リセット	VCC の下降 (監視電圧: Vdet0)
電圧監視 1 リセット	VCC の下降 (監視電圧: Vdet1)
電圧監視 2 リセット	VCC の下降 (監視電圧: Vdet2)
SRAM パリティ エラー リセット	SRAM パリティ エラー検出
SRAM ECC エラー リセット	SRAM ECC エラー検出 SRAM
バスマスタ MPU エラーリセット	バスマスタ MPU エラー検出
トラストゾーンエラーリセット	トラストゾーンエラー検出
キャッシュパリティエラーリセット	キャッシュパリティエラー検出
ディープ ソフトウェア スタンバイリセット	割り込みによるディープソフトウェアスタンバイモードの解除
ソフトウェアリセット	レジスタ設定

6.1 端子リセット

RES#ピンに Low レベルが入力されると、すべての処理が中止され、MCU はリセット状態に入ります。実行中の MCU をリセットするには、指定されたリセットパルス幅の間、RES#を Low レベルで保持する必要があります。詳細なタイミング要件については、ユーザーズマニュアル ハードウェア編の「電気的特性」の「リセットタイミング」を参照してください。また、デバッグサポートに関するリセット回路の詳細については、本書の 2 章、「エミュレータサポート」を参照してください。

正しいリセットを行うために POR 回路が内部的に RES#信号を Low レベルで保持するため、外部キャパシタを RES#信号に使用する必要はありません。このプロセスを開始するには、最小限のリセットパルスが必要です。

6.2 パワーオンリセット

パワーオンリセット (POR) を発生させる条件は2つあります:

1. 電源投入時に RES#ピンがハイレベルの場合。
2. VCC が VPOR 以下の状態で RES# ピンがハイレベルにある場合。

VCC がパワーオンリセット電圧 (VPOR) を超え、パワーオンリセット時間 (tPOR) を経過した後、チップはパワーオンリセット状態から解除されます。パワーオンリセット時間は、外部電源と MCU を安定させるための時間です。電圧レベルやタイミングの詳細については、ユーザーズマニュアル ハードウェア編の「電気的特性」の「POR および LVD 特性」を参照してください。

POR 回路は、VCC と同時に RES#が高い状態であることに依存していますので、リセットピンにコンデンサを配置しないでください。これにより、RES#の立ち上がり時間が VCC に対して遅くなり、POR 回路がパワーオンの状態を正しく認識できなくなる可能性があります。

電源 (VCC) が VPOR 以下になるかその時点で RES#ピンが High 状態の場合、パワーオンリセットが発生します。VCC が VPOR を超え、tPOR が経過した後、チップはパワーオン状態から解放されます。

パワーオンリセット後、RSTSRO の PORF ビットは 1 に設定されます。ピンリセットの後、PORF は 0 にクリアされます。

6.3 独立ウォッチドッグタイマのリセット

独立ウォッチドッグタイマ (IWDG) により発生する内部リセットです。

IWDG がアンダーフローすると、独立ウォッチドッグタイマリセットをオプションで発生可能 (NMI を生成することも可能)。そして RSTSRI の IWDTRF ビットが 1 に設定されます。短い遅延 (通常 320µs) の後、IWDG リセットがキャンセルされます。

6.4 ウォッチドッグタイマリセット

これはウォッチドッグタイマ (WDT) によって発生する内部リセットです。

WDT がアンダーフローすると、ウォッチドッグタイマリセットがオプションで発生可能 (NMI を生成することも可能)。そして RSTSRI の WDTRF ビットが 1 に設定されます。短い遅延 (通常 320µs) の後、WDT リセットがキャンセルされます。

6.5 電圧監視リセット

RA6 シリーズには、MCU がブラウンアウト中に安全な動作を維持するための回路が含まれています。オンボードコンパレータは、電源電圧を3つの基準電圧 (Vdet0、Vdet1、Vdet2) に対して比較します。供給電圧が各基準電圧を下回ると、割り込みまたはリセットが発生可能です。検出電圧 Vdet0、Vdet1、Vdet2 は、それぞれ3つのレベルから選択可能です。

後に Vcc が Vdet0、Vdet1、Vdet2 を上回ると、安定化時間が経過した後に電圧監視リセットから解除されます。低電圧検出はパワーオンリセット後に無効になります。電圧監視は、オプション機能レジスタ OFS1 を使用することで有効にすることができます。詳細については、ユーザーズマニュアル ハードウェア編の「低電圧検出 (LVD)」を参照してください。

LVD リセット後、RSTSRO の LVDnRF (n = 0、1、2) ビットが 1 に設定されます。

6.6 ディープソフトウェアスタンバイリセット

ディープソフトウェアスタンバイモードが割り込みで解除された時に発生する内部リセットです。

ディープソフトウェアスタンバイモードが解除されると、ディープソフトウェアスタンバイリセットが発生し、クロック発振が開始されます。割り込みを受信すると、ディープスタンバイ解除待機時間 (tDSBYWT 34-35 クロック) が経過した後、リセットが解除され、通常の処理が開始される。ディープソフトウェアスタンバイモードの詳細については、ユーザーズマニュアル ハードウェア編の「低電力モード」を参照してください。

ディープソフトウェアスタンバイリセット後、RSTSRO の DPSRSTF ビットが 1 に設定されます。

6.7 ソフトウェアリセット

SWRR レジスタに 0xA501 を書き込むことで発生する内部リセットです。ソフトウェアリセット使用時の内部リセット時間は、最大 960 μ s です。ソフトウェアリセットを使用する場合、ソフトウェアリセットコマンドを発行する前に、ウォッチドッグタイマが有効であることを確認してください。

ソフトウェアリセットが発生すると、RSTSR1 の SWRF ピンが 1 に設定され、短い遅延（通常 320 μ s）の後、ソフトウェアリセットがキャンセルされます。

6.8 その他のリセット

MCU 内のほとんどの周辺機能は、特定の障害条件下でリセットを発生することができます。これらのリセットを有効にするためには、ハードウェア構成は必要ありません。各周辺機能に対してリセットが発生する条件の詳細については、ユーザーズマニュアル ハードウェア編の関連する章を参照してください。

6.9 コールド/ウォームスタートの決定

RA6 シリーズでは、リセット処理の原因をユーザが判断することができます。RSTSR2 の CWSF フラグは、パワーオンリセットによるリセット処理（コールドスタート）か、動作中のリセット信号入力によるリセット処理（ウォームスタート）かを示しています。

パワーオンリセットが発生した場合、フラグは 0 に設定されます。それ以外の場合は 0 になりません。ソフトウェア経由で 1 が書き込まれると、フラグは 1 に設定されます。0 が書き込まれた場合でも、0 に設定されません。

6.10 リセット要因の特定

RA6 シリーズでは、ユーザはリセット信号の発生元を特定することができます。リセットの原因を特定するには、RSTSRO、RSTSR1、RSTSR0 を読み取ります。フロー図については、ユーザーズマニュアル ハードウェア編の「リセット発生要因の特定」を参照してください。

以下のコードサンプルは、ルネサス FSP の CMSIS ベースのレジスタ構造を用いて、ソフトウェアリセット、ディープソフトウェアスタンバイ、パワーオンリセットによるリセットが発生したかどうかを判定する方法です。

```
/* Deep Software Standby Reset */
if(1 == R_SYSTEM->RSTSRO_b.DPSRSTF)
{
    /* Do something */
}

/* Power on Reset */
if(1 == R_SYSTEM->RSTSRO_b.PORF)
{
    /* Do something */
}

/* Software Reset */
if(1 == R_SYSTEM->RSTSR1_b.SWRF)
{
    /* Do something */
}
```

7. TrustZone® 有効化

7.1 Arm® TrustZone® テクノロジーの実装

RA6M4 および RA6M5 などの一部の RA6 シリーズには、Arm® TrustZone® (TZ) セキュリティ機能が含まれています。TZ の詳細については、Arm のドキュメント (<https://developer.arm.com/ip-products/security-ip/trustzone>) または適切なユーザーズマニュアル ハードウェア編を参照してください。

Arm TrustZone 技術は、MCU とアプリケーションをセキュアパーティションとノンセキュアパーティションに分けられます。セキュアアプリケーションは、セキュアおよびノンセキュアメモリ、リソースにアクセスできます。ノンセキュアアプリケーションは、ノンセキュアメモリおよびリソースに加えて、ノンセキュアコーラブル (NSC) 領域に配置された一連のベニアを介してセキュアリソースにアクセスできます。これにより、ノンセキュアパーティションから呼び出されたときにセキュアコードのアクセスポイントが1つ確保されます。MCU はデフォルトでセキュアパーティションを起動します。CPU のセキュリティ状態はセキュアまたはノンセキュアのいずれかになります。MCU コードフラッシュ、データフラッシュ、SRAM はセキュア (S)、ノンセキュア (NS) の領域に分割されます。コードフラッシュと SRAM には、さらに NSC (ノンセキュアコーラブル) と呼ばれる領域があります。これらのメモリセキュリティ属性は、デバイスのライフサイクルが SSD (Secure Software Debug) 状態のときに、SCI または USB ブートモードコマンドを介して不揮発性メモリに設定されます。メモリセキュリティ属性は、アプリケーション実行前に IDAU (実装定義属性ユニット) ペリフェラルとメモリコントローラに読み込まれ、アプリケーションコードで更新されることはありません。

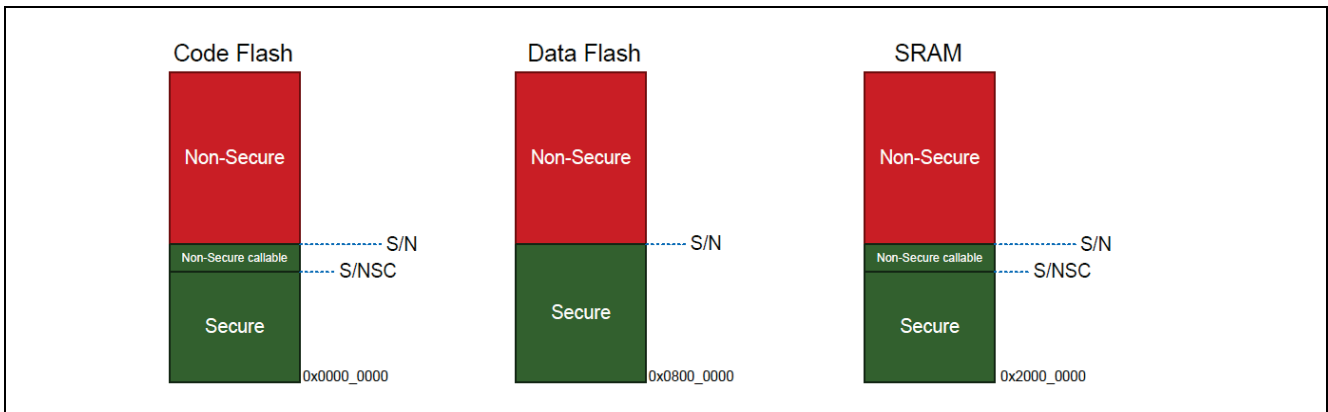


図 14. セキュア領域とノンセキュア領域

注: すべての外部メモリアクセスはノンセキュアと見なされます

フラッシュと SRAM は、セキュア、ノンセキュア、ノンセキュアコーラブルに分けられます。ノンセキュア領域からのすべてのセキュアメモリアクセスは、ノンセキュアコーラブルゲートウェイを経由し、特定のセキュアゲートウェイ (SG) アセンブラ命令を対象にしなければなりません。これにより、セキュア API へのアクセスが固定され、サブファンクションへの呼び出しなどが防止されます。SG 命令を対象にできなかった場合、TZ 例外が発生します。TZ 対応コンパイラは、CMSE 拡張機能を使用して、NSC ベニアの生成を自動的に管理します。参考資料:

- RA Arm® TrustZone® Tooling Primer
- Securing Data at Rest Using the Arm® TrustZone®
- Security Design with Arm® TrustZone® - IP Protection
- Establishing and Protecting Device Identity using SCE9 and Arm® TrustZone®
- Device Lifecycle Management Key Installation
- Arm TrustZone Security section in the relevant MCU Hardware User' s Manual.

7.2 TrustZone のエミュレータサポート

ルネサスは、SWD や JTAG 通信によるデバッグと SCI 通信によるシリアルプログラミングの両方に対応したエミュレータを提供しています。このエミュレータを使用することで、デバッグとシリアルプログラミングを簡単に切り替えることができます。このエミュレータを使用した場合の 10 ピンソケット、20 ピンソケットのピン配置は以下の通りです。SWD と JTAG のピン配列は Arm®標準であり、SCI 通信によるシリアルプログラミングのために MD、TXD、RXD ピンが追加されています。

TrustZone IDAU 境界レジスタ設定をプログラムするには、シリアルプログラミングインタフェースを使用する必要があります。デバッグとシリアルプログラミングの両方を使用するには、基板上で P300 / SWCLK / TCK と P201 / MD ピンをワイヤード OR 回路で接続することを推奨します。

表 13. エミュレータのピン割り当て

ピン番号	SWD	JTAG	SCI を使用したシリアルプログラミング
1	VCC	VCC	VCC
2	P108/SWDIO	P108/SWDIO	NC
4	P300/SWCLK P201/MD によるワイヤード OR	P300/TCK P201/MD によるワイヤード OR	P201/MD
6	P109/SWO/TXD9	P109/TDO/TXD9	P109/TDO/TXD9
8	P110/SWO/RXD9	P110/TDI/RXD9	P110/TDI/RXD9
9	GNDdetect	GNDdetect	GNDdetect
10	nRESET	nRESET	nRESET
12	P214/TCLK	P214/TCLK	NC
14	P211/TDATA[0]	P211/TDATA[0]	NC
16	P210/TDATA[1]	P210/TDATA[1]	NC
18	P209/TDATA[2]	P209/TDATA[2]	NC
20	P208/TDATA[3]	P208/TDATA[3]	NC
3, 5, 15, 17, 19	GND	GND	GND
7	NC	NC	NC
11, 13	NC	NC	NC

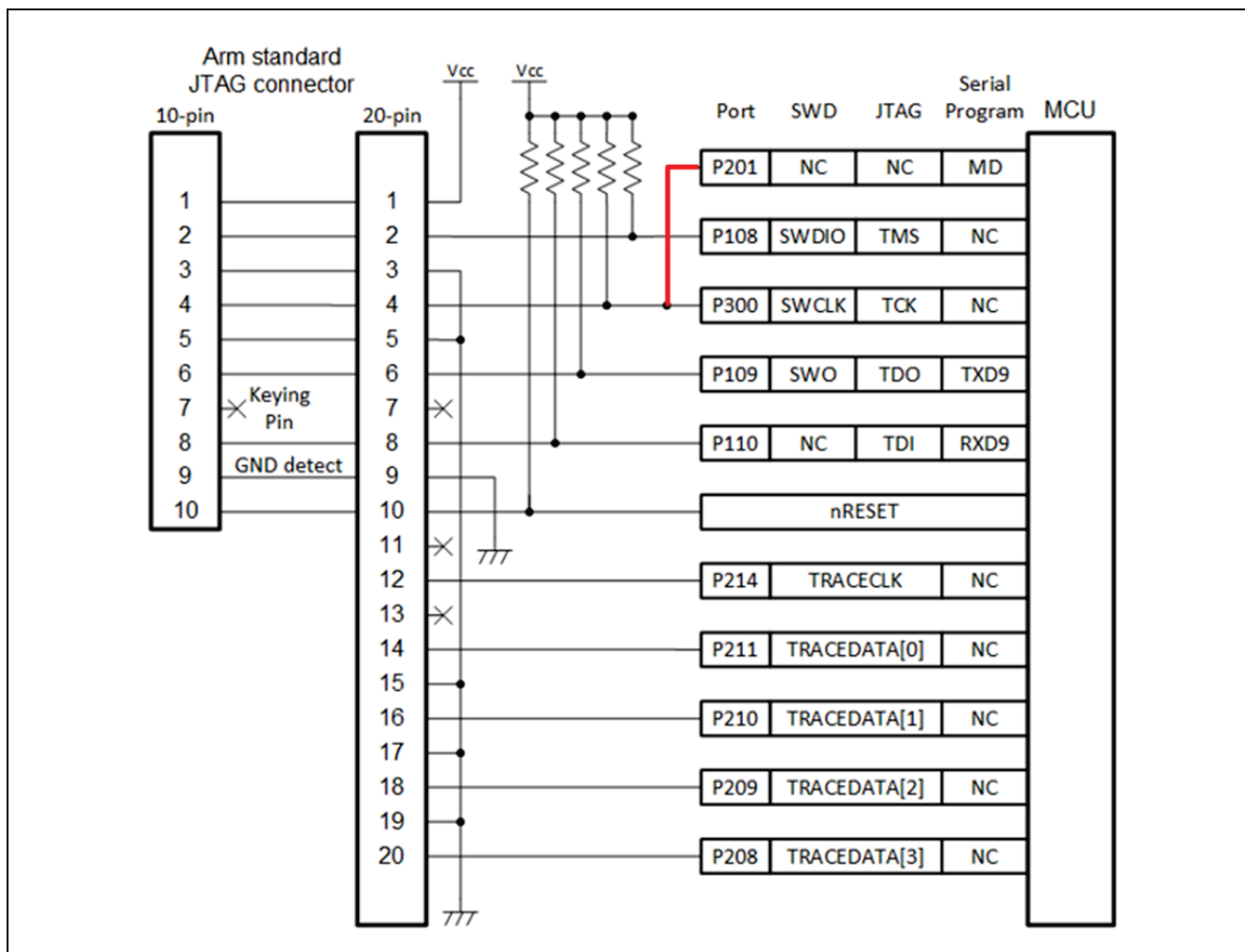


図 15. TrustZone をサポートする MCU のエミュレータ接続

8. メモリ

RA6 シリーズは、プログラム、データ、および外部メモリバスを含む、0000 0000h から FFFF FFFFh に及ぶ 4GB のリニアアドレス空間をサポートしています。このファミリの一部には、外部メモリバスに接続された SDRAM デバイスへのアクセスを可能にする SDRAM コントローラが含まれています。プログラムフラッシュメモリとデータメモリはアドレス空間を共有しており、それぞれにアクセスするためには別々のバスが使用され、プログラムとデータの同時アクセスが可能になり、性能が向上します。メモリマップ内には、オンチップ RAM、周辺 I/O レジスタ、プログラム ROM、データフラッシュ、外部メモリの領域が含まれます。

FFFF FFFFh	Cortex [®] -M4用システム領域
E000 0000h	予約領域 (注2)
9800 0000h	外部アドレス空間 (SDRAM領域)
9000 0000h	予約領域 (注2)
8800 0000h	外部アドレス空間 (CS領域)
8000 0000h	予約領域 (注2)
6800 0000h	外部アドレス領域 (SPI領域)
6000 0000h	予約領域 (注2)
4080 0000h	フラッシュ/I/Oレジスタ
407F C000h	予約領域 (注2)
407F B1A0h	内蔵フラッシュ(オプション設定メモリ) ^(注4)
407F B17Ch	予約領域 (注2)
407F 0000h	フラッシュ/I/Oレジスタ
407E 0000h	予約領域 (注2)
4011 0000h	内蔵フラッシュ (データフラッシュ)
4010 0000h	周辺I/Oレジスタ
4000 0000h	予約領域 (注2)
2010 0000h	スタンバイSRAM
200F E000h	予約領域 (注2)
2008 0000h	SRAM0, SRAM1
2000 0000h	SRAMHS領域
1FFE 0000h	予約領域 (注2)
0280 0000h	メモリマッピング領域
0200 0000h	予約領域 (注2)
0100 A168h	内蔵フラッシュ (オプション設定メモリ)
0100 A150h	予約領域 (注2)
0100 8000h	内蔵フラッシュ (オプション設定メモリ)
0100 7000h	予約領域 (注2)
0020 0000h	内蔵フラッシュ (プログラムフラッシュ) (読み出しのみ) ^(注1) ^(注2)
0000 0000h	

図 16. RA6M3 メモリ マップ

8.1 SRAM

RA6 シリーズには、パリティビットチェックまたは ECC（エラー訂正コード）を備えたオンチップの高速 SRAM モジュールが用意されています。SRAM0 の最初の 32 KB の領域は ECC の対象となります。他の領域ではパリティチェックが実行されます。以下の表には、SRAM の仕様が示されています。SRAM モジュールの数と容量はデバイスによって異なります。詳細については、ユーザーズマニュアル ハードウェア編を参照してください。

表 53.1 SRAM の仕様

項目	内容 (ECC なしの場合)	内容 (ECC ありの場合)	内容 (SRAMHS の場合)
SRAM 容量	SRAM0 : 224KB SRAM1 : 256KB	SRAM0 (ECC 領域) : 32KB	SRAMHS : 128KB
SRAM アドレス (注2)	SRAM0 : 2000 8000h ~ 2003 FFFFh SRAM1 : 2004 0000h ~ 2007 FFFFh	SRAM0 (ECC 領域) : 2000 0000h ~ 2000 7FFFh	SRAMHS : 1FFE 0000h ~ 1FFF FFFFh
アクセス (注1)	リードサイクルにウェイトステートがデフォルトで挿入されています。ICLK 周波数が 60MHz より高い (最高 120MHz) 場合、ウェイトステートが必要になります。ICLK 周波数が 60MHz 以下の場合、ウェイトステートは必要ありません。詳細は、53.4 使用上の注意事項を参照してください。	リードサイクルにウェイトステートがデフォルトで挿入されています。ICLK 周波数が 60MHz より高い (最高 120MHz) 場合、ウェイトステートが必要になります。ICLK 周波数が 60MHz 以下の場合、ウェイトステートは必要ありません。詳細は、53.4 使用上の注意事項を参照してください。	SRAMHS へのアクセスは、常にウェイトステートなしになります。
データ保持	ディープソフトウェアスタンバイモード時のデータ保持機能なし		
モジュールストップ機能	モジュールストップ状態に設定して消費電力を削減		
パリティ	偶数パリティ (データ : 8 ビット、パリティ : 1 ビット)	パリティなし	偶数パリティ (データ : 8 ビット、パリティ : 1 ビット)
エラーチェック機能	偶数パリティエラーチェック	2 ビット以下の誤り検出	偶数パリティエラーチェック

注 1. 詳細は、53.3.7 アクセスサイクルを参照してください。

注 2. Cortex®-M4 プロセッサは、Arm®v7 非整列アクセスに対応しています。本製品では、SRAMHS と SRAM0 は互いに隣接しており、両者間にはアクセス境界があります。そのため SRAMHS と SRAM0 をメモリ空間の連続領域として使用する場合、境界をまたぐアクセスは、意図しないデータアクセスとなる恐れがあるので、行わないでください。詳細は、ARM® Cortex®-M4 Processor Technical Reference Manual を参照してください。

図 17. RA6M3 の SRAM 仕様例

8.2 スタンバイ SRAM

RA6 シリーズ は、ディープソフトウェアスタンバイ モードでデータを保持するオンチップ SRAM を提供します。以下の表は、スタンバイ RAM の仕様を示しています。

ディープソフトウェアスタンバイモードでスタンバイ SRAM に電源が供給するには、DPSBYCR.DEEPCUT [1 : 0] ビットを有効にします。DPSBYCR.DEEPCUT [1:0] ビットが 00b に設定されている場合、スタンバイ SRAM のデータはディープソフトウェアスタンバイモードで保持されます。DPSBYCR.DEEPCUT [1 : 0] ビットの詳細については、11 章「低消費電力モード」を参照してください。

表 54.1 スタンバイ SRAM の仕様

項目	内容
SRAM 容量	8KB
SRAM アドレス	200F E000h ~ 200F FFFFh
アクセス	アクセス回数は、ICLK と PCLKB との周波数関係に依存します。詳細は、54.2.4 アクセスサイクルを参照してください。
データ保持	ディープソフトウェアスタンバイモード時、データを保持可能
パリティ	偶数パリティ（データ：8ビット、パリティ：1ビット）
モジュールストップ機能	モジュールストップ状態に設定して消費電力を削減

図 18. RA6M3 スタンバイ SRAM 仕様の例

ルネサス FSP の低消費電力モード（Low Power Mode）ドライバには、以下の図に示すように、スタンバイ SRAM への電力供給を切断するか保持するかのオプションが用意されています。低消費電力モードドライバの API を呼び出して、選択した設定を MCU レジスタに書き込む必要があります。

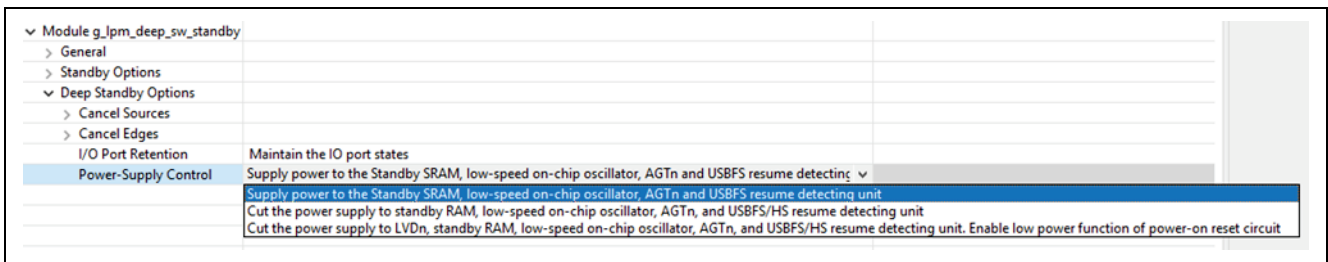


図 19. ルネサス FSP コンフィギュレータを使用したスタンバイ SRAM への電源供給の有効化/無効化

8.3 周辺 I/O レジスタ

周辺 I/O レジスタのブロックは、デバイスと現在の動作モードに応じてメモリマップ上のさまざまな場所に表示されます。ほとんどの周辺 I/O レジスタは、アドレス 4000 0000h から 400F FFFFh までの領域を占有しますが、デバイスによっては位置やサイズが異なる場合があります。詳細については、ユーザーズマニュアル ハードウェア編を参照してください。詳細は「I/O レジスタ」の付録や各周辺機能のレジスタ説明に記載されています。この領域は、すべての動作モードで常に利用可能なレジスタを含んでいます。フラッシュメモリへのアクセスを制御するフラッシュ I/O レジスタは、407E 0000h~407E FFFFh と 407F C000h~407F FFFFh の 2 領域を占有します。

ルネサス FSP は、特定のデバイスのすべての周辺 I/O レジスタを簡単にアクセスできる I/O データ構造にマッピングする CMSIS データ構造の C ヘッダファイルを含んでいます。

8.4 内蔵フラッシュメモリ

RA6 シリーズには、コードフラッシュとデータフラッシュの 2 つのフラッシュメモリセクションがあり、デバイスによってサイズやプログラマブル可能回数が異なります。フラッシュコントロールユニット（FCU）は、フラッシュメモリの書き込みと消去を制御します。フラッシュアプリケーションコマンドインタフェース（FACI）は、指定された FACI コマンドに従って FCU を制御します。コードフラッシュは、ユーザーアプリケーションコードと定数データを保存するように設計されています。データフラッシュは、構成パラメータ、ユーザ設定、ログデータなど、随時更新される情報を保存するように設計されています。データフラッシュ領域の書き込みと消去の単位は、コードフラッシュの場合よりはるかに小さい（データフラッシュの場合は 2 バイト、コードフラッシュの場合は 128 バイト）。

データフラッシュ領域とコードフラッシュ領域の両方は、アプリケーションコードによってプログラムや消去が可能です。これにより、外部のプログラミングツールを接続しなくてもフィールドでファームウェアの更新が可能になります。

ルネサス FSP は、コードフラッシュメモリとデータフラッシュメモリの両方に対して HAL 層ドライバを提供しています。

以下の表は、コードフラッシュメモリとデータフラッシュメモリの仕様の例を示しています。

表 55.1 コードフラッシュメモリとデータフラッシュメモリの仕様

項目	コードフラッシュメモリの仕様	データフラッシュメモリの仕様
メモリ容量	• 最大2MB	64KB
リードサイクル	<ul style="list-style-type: none"> • 80MHz < ICLK周波数 ≤ 120MHz キャッシュヒット：1サイクル キャッシュミス：3サイクル • 40MHz < ICLK周波数 ≤ 80MHz キャッシュヒット：1サイクル キャッシュミス：2サイクル • ICLK周波数 ≤ 40MHz キャッシュヒット：1サイクル キャッシュミス：1サイクル 	ワード、バイトアクセス時にはFCLK 7サイクルでのリード（FCLK周波数は最高60MHz）
イレース後の値	FFh	不定
プログラム/イレース方式	<ul style="list-style-type: none"> • FACLコマンド発行領域（407E 0000h）に設定したFACLコマンドで、コードフラッシュメモリ/データフラッシュメモリのプログラム/イレースが可能 • 専用フラッシュメモリプログラムによるシリアルインタフェース通信を介したプログラム（シリアルプログラミング） • ユーザプログラムによるフラッシュメモリのプログラム（セルフプログラミング） 	
セキュリティ機能	フラッシュメモリの不正改ざん/不正リードを防止	
プロテクション機能	フラッシュメモリの誤オーバーライトを防止	
BGO（バックグラウンドオペレーション）機能	<ul style="list-style-type: none"> • コードフラッシュメモリのプログラム中にコードフラッシュメモリの読み出しが可能（注1） • データフラッシュメモリのプログラム中にコードフラッシュメモリの読み出しが可能 	
プログラム/イレース単位	<ul style="list-style-type: none"> • ユーザ領域へのプログラム：128バイト単位 • ユーザ領域のイレース：ブロック単位 	<ul style="list-style-type: none"> • データ領域へのプログラム：4/8/16バイト単位 • データ領域のイレース：64/128/256バイト単位
その他の機能	セルフプログラミング中の割り込み受け付け可能	
	本MCUの初期設定でフラッシュメモリ拡張領域（オプションバイト）の設定可能	
オンボードプログラミング（4種類）	シリアルプログラミングモード（SCIブートモード）でのプログラム <ul style="list-style-type: none"> • 調歩同期式シリアルインタフェース（SCI9）を使用 • 転送速度は自動調整 シリアルプログラミングモード（USBブートモード）でのプログラム <ul style="list-style-type: none"> • USBFSを使用 • 特別なハードウェアが不要で、PCと直結可能 オンチップデバッグモードによるプログラム <ul style="list-style-type: none"> • JTAG/SWDインタフェースを使用 • 専用ハードウェアは必要なし ユーザプログラム中のコードフラッシュメモリ/データフラッシュメモリ書き換えルーチンによるプログラム <ul style="list-style-type: none"> • システムをリセットすることなくコードフラッシュメモリ/データフラッシュメモリのプログラムが可能 	

注 1. 書き込みできる範囲の組み合わせに制約があります。表 55.11 を参照してください。

図 20. RA6M3 MCU のコードフラッシュメモリおよびデータフラッシュメモリの仕様

注：コードフラッシュの消去状態は FFh ですが、データフラッシュの消去状態は不定です。

8.4.1 バックグラウンド動作

RA6 シリーズは、コードフラッシュとデータフラッシュのバックグラウンドオペレーションをサポートしています。つまり、プログラムまたは消去が開始された場合でも、ユーザは操作中のメモリ領域以外のメモリの実行やアクセスを継続できます。例えば、データフラッシュメモリが消去またはプログラムされている間に CPU がコードフラッシュからアプリケーションコードを実行することができます。またコードフラッシュメモリが消去またはプログラムされている間に、CPU が SRAM からアプリケーションコードを実行することができます。ただし、この規則の例外として、データフラッシュはコードフラッシュのプログラムまたは消去中にアクセスできません。

8.4.2 ID コード保護

Cortex-M4 コアを搭載した RA6 シリーズは、オプション設定メモリ領域に ID コードとして使用される 128 ビットメモリを搭載しています。この ID コードを空白 (0xFF) にすると、保護機能は有効にならず、ブートモードやオンチップデバッグを使用して MCU にアクセスすることが可能になります。ID コードが設定され

ている場合、ユーザはこれらのモードへのアクセスを制御することができます。ユーザは、常に接続を拒否するか、一致する ID コードが入力されたときに接続を許可するかを選択することができます。詳細は、RA6 MCU ユーザーズマニュアル ハードウェア編の「OCD/シリアルプログラマ ID 設定レジスタ (OSIS)」と「ID コードプロテクション」を参照してください。

ルネサス FSP のコンフィギュレータは、ID コードコードプロテクションの設定を行うためのオプションを提供しています。

▼ RA6M3 Family	
> OFS0 register settings	
> OFS1 register settings	
> MPU	
> Clocks	
ID Code Mode	Unlocked (Ignore ID)
ID Code (32 Hex Characters)	FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
> RA Common	

図 21. ルネサス FSP コンフィギュレータを使用した ID コード設定

注) ID コード保護設定は、間違えると MCU へのアクセスがブロックされる可能性があるため、慎重に取り扱う必要があります。

8.4.3 デバイスライフサイクルマネジメント

Cortex-M33 コアを搭載した RA6 シリーズは、デバイスライフサイクルマネジメント (DLM) を備えています。DLM とは、製品が初期段階から開発、生産、そして最終的に使用済みとなるまでのプロセスを管理することです。RA ファミリ MCU のデバッグ機能とシリアルプログラミング機能は、デバイスライフサイクルの状態によって定義されます。詳細については、「Device Lifecycle Management Key Installation」のアプリケーションノートを参照してください。

8.4.4 フラッシュブロック保護

Cortex-M33 コアを搭載した RA6 MCU には、フラッシュブロック保護機能があり、セキュアまたは非セキュアなソフトウェアによるイレースや再プログラムからセキュアまたは非セキュアなフラッシュ領域を保護します。保護は、セキュア、非セキュアなソフトウェアの両方に対して行われることに注意してください。

詳細については、Arm® TrustZone®アプリケーションノートを使用した、保存データの保護に関する章「フラッシュブロック保護の設定」を参照してください。

8.4.5 メモリ保護ユニット

Cortex-M4 コアを搭載した RA6 シリーズには、メモリ保護ユニット (MPU) があります。これらは、様々な MCU 領域を不正なアクセスから保護する機能を持っています。選択肢には、書き込みと読み取りを許可する、書き込みを禁止する、書き込みと読み取りを禁止するものがあります。特定のメモリアドレスで対応する定数値を設定することで、これらのオプションのうちいずれかを選択します。詳細については、MCU ユーザーズマニュアル ハードウェア編の「メモリ保護ユニット」を参照してください。

▼ RA6M3 Family	
> OFS0 register settings	
> OFS1 register settings	
▼ MPU	
Enable or disable PC Region 0	Disabled
PC0 Start	0xFFFFFFFFC
PC0 End	0xFFFFFFFF
Enable or disable PC Region 1	Disabled
PC1 Start	0xFFFFFFFFC
PC1 End	0xFFFFFFFF
Enable or disable Memory Region 0	Disabled
Memory Region 0 Start	0x00FFFFFFC
Memory Region 0 End	0x00FFFFFF
Enable or disable Memory Region 1	Disabled
Memory Region 1 Start	0x200FFFFFFC
Memory Region 1 End	0x200FFFFFF
Enable or disable Memory Region 2	Disabled
Memory Region 2 Start	0x407FFFFFFC
Memory Region 2 End	0x407FFFFFF
Enable or disable Memory Region 3	Disabled
Memory Region 3 Start	0x400DFFFFFFC
Memory Region 3 End	0x400DFFFFFF

図 22. ルネサス FSP コンフィギュレータを使用した MPU セットアップ

注: MPU 設定は、MCU 領域へのアクセスをブロックする可能性のあるミスを防ぐために慎重に処理する必要があります。

8.5 外部メモリ

RA6 シリーズには、外部メモリやデバイスに接続するための外部データバスが含まれています。一部のデバイスには、最大 128 M バイトの外部 SDRAM の使用を可能にする内蔵 SDRAM コントローラも搭載されています。8 つのプログラマブルチップセレクトは、チップセレクトごとに設定可能な複数のオプションを提供し、様々な外部デバイスに接続することができます。メモリマップの外部チップセレクト領域は、アドレス 0x60000000 から始まります。詳細については、ユーザーズマニュアル ハードウェア編を参照してください。

8.5.1 外部 16 ビットメモリデバイスの使用

バイトセレクトラインを持つ外部 16 ビットメモリデバイスを接続する場合、MCU の A1 をメモリの A0 に接続し、MCU の A0 をバイトセレクトラインに接続してください。

8.5.2 SDRAM 初期化の例

ルネサス FSP は、すべての外部バス制御レジスタをマップする CMSIS データ構造内の C ヘッダファイルを提供しています。以下の関数は、ルネサス FSP の CMSIS レジスタ構造体を使用して SDRAM メモリコントローラを初期化する例です。

```
void bsp_sdram_init (void)
{
    /** Delay at least 100uS after SDCLK active */
    R_BUS->SoftwareDelay(100U, BSP_DELAY_UNITS_MICROSECONDS);

    /** Setting for SDRAM initialization sequence */
    #if (BSP_PRV_SDRAM_TRP < 3)
        R_BUS->SDRAM.SDIR_b.PRC = 3U;
    #else
        R_BUS->SDRAM.SSDIR_b.PRC = BSP_PRV_SDRAM_TRP - 3U;
    #endif

    while(R_BUS->SDRAM.SDSR)
    {
        /* According to h/w manual, need to confirm that all the status bits in
        SDSR are 0 before SDIR modification. */
    }

    R_BUS->SDRAM.SDIR_b.ARFC = BSP_PRV_SDRAM_SDIR_REF_TIMES;

    while(R_BUS->SDRAM.SDSR)
    {
        /* According to h/w manual, need to confirm that all the status bits in
        SDSR are 0 before SDIR modification. */
    }

    #if (BSP_PRV_SDRAM_TRFC < 3)
        R_BUS->SDRAM.SDIR_b.ARFI = 0U;
    #else
        R_BUS->SDRAM.SDIR_b.ARFI = BSP_PRV_SDRAM_TRFC - 3U;
    #endif

    while(R_BUS->SDRAM.SDSR)
    {
        /* According to h/w manual, need to confirm that all the status bits in
        SDSR are 0 before SDICR modification. */
    }

    /** Start SDRAM initialization sequence.
    * Following operation is automatically done when set SDICR.INIRQ bit.
    * Perform a PRECHARGE ALL command and wait at least tRP time.
    * Issue an AUTO REFRESH command and wait at least tRFC time.
    * Issue an AUTO REFRESH command and wait at least tRFC time.
    */
    R_BUS->SDRAM.SDICR_b.INIRQ = 1U;
    while(R_BUS->SDRAM.SDSR_b.INIST)
    {
        /* Wait the end of initialization sequence. */
    }

    /** Setting for SDRAM controller */
    R_BUS->SDRAM.SDCCR_b.BSIZE = BSP_PRV_SDRAM_BUS_WIDTH; /* set
SDRAM bus width */
    R_BUS->SDRAM.SDAMOD_b.BE = BSP_PRV_SDRAM_CONTINUOUS_ACCESSMODE; /*
enable continuous access */
    R_BUS->SDRAM.SDCMOD_b.EMODE = BSP_PRV_SDRAM_ENDIAN_MODE; /* set
endian mode for SDRAM address space */
}
```

```

while(R_BUS->SDRAM.SDSR)
{
    /* According to h/w manual, need to confirm that all the status bits in
SDSR are 0 before SDMOD modification. */
}

/** Using LMR command, program the mode register */
R_BUS->SDRAM.SDMOD = (((uint16_t) (BSP_PRV_SDRAM_MR_WB_SINGLE_LOC_ACC <<
9)
    | (uint16_t) (BSP_PRV_SDRAM_MR_OP_MODE          << 7))
    | (uint16_t) (BSP_PRV_SDRAM_CL                  << 4))
    | (uint16_t) (BSP_PRV_SDRAM_MR_BT_SEQUENTIAL   << 3))
    | (uint16_t) (BSP_PRV_SDRAM_MR_BURST_LENGTH    << 0));

/** wait at least tMRD time */
while(R_BUS->SDRAM.SDSR_b.MRSST)
{
    /* Wait until Mode Register setting done. */
}

/** Set timing parameters for SDRAM */
R_BUS->SDRAM.SDTR_b.RAS = BSP_PRV_SDRAM_TRAS - 1U;      /* set ACTIVE-to-
PRECHARGE command cycles*/
R_BUS->SDRAM.SDTR_b.RCD = BSP_PRV_SDRAM_TRCD - 1U;      /* set ACTIVE to
READ/WRITE delay cycles */
R_BUS->SDRAM.SDTR_b.RP  = BSP_PRV_SDRAM_TRP  - 1U;      /* set PRECHARGE
command period cycles */
R_BUS->SDRAM.SDTR_b.WR  = BSP_PRV_SDRAM_TWR  - 1U;      /* set write
recovery cycles */
R_BUS->SDRAM.SDTR_b.CL  = BSP_PRV_SDRAM_CL;   /* set SDRAM
column latency cycles */

/** Set row address offset for target SDRAM */
R_BUS->SDRAM.SDADR_b.MXC = BSP_PRV_SDRAM_SDADR_ROW_ADDR_OFFSET - 8U;

R_BUS->SDRAM.SDRFCR_b.REFW = (uint16_t) (BSP_PRV_SDRAM_TRFC - 1U); /*
set Auto-Refresh issuing cycle */
R_BUS->SDRAM.SDRFCR_b.RFC  = BSP_PRV_SDRAM_REF_CMD_INTERVAL - 1U; /*
set Auto-Refresh period */

/** Start Auto-refresh */
R_BUS->SDRAM.SDRFEN_b.RFEN = 1U;

/** Enable SDRAM access */
R_BUS->SDRAM.SDCCR_b.EXENB = 1U;
}

```

8.6 データ整列

データの整列に制限はありません。MCUは、奇数のメモリ位置でのバイト、ワード、ロングアクセスを実行できます。データアクセスを揃えることが最適であるにしても、必須ではありません。

8.7 エンディアンの制限

Cortex-M コア上でコードを実行するためには、メモリ空間はリトルエンディアンである必要があります。

9. レジスタ書き込み保護

レジスタライトプロテクト機能により、ソフトウェアエラーによる上書きから重要なレジスタを保護します。保護するレジスタは、プロテクトレジスタ (PRCR) で設定されます。表 14 は、PRCR ビットと保護されるレジスタの関連を示しています。

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
PRKEY[7:0]								-	-	-	-	PRC3	-	PRC1	PRC0

図 23. PRCR レジスタ

表 14. PRCR 保護ビット

PRCR ビット	概要
PRC0	<ul style="list-style-type: none"> クロック発生回路関連レジスタ: SCKDIVCR, SCKDIVCR2, SCKSCR, PLLCCR, PLLCR, BCKCR, MOSCCR, HOCOCR, MOCOCR, CKOCR, TRCKCR, OSTDCR, OSTDSR, EBCKOCR, SDCKOCR, MOCOUTCR, HOCOUTCR, MOSCWTCR, MOMCR, SOSCCR, SOMCR, LOCOCR, LOCOUTCR, HOCOWTCR, FLLCR1, FLLCR2
PRC1	<ul style="list-style-type: none"> ローパワーモードに関連するレジスタ: SBYGR, SNZCR, SNZEDCR, SNZREQCR, OPCCR, SOPCCR, DPSBYGR, DPSIERO-3, DPSIFRO-3, DPSIEGRO-2, SYOCDGR, STCONR Registers related to the battery backup function: VBTBKRn (n = 0 to 511), VBT1CTLR
PRC3	<ul style="list-style-type: none"> LVD に関連するレジスタ: LVD1CR1, LVD1SR, LVD2CR1, LVD2SR, LVCMPGR, LVDLVLR, LVD1CR0, LVD2CR0
PRKEY[7:0]	これらのビットは、PRCR レジスタへの書き込みアクセスを制御します。PRCR レジスタを変更するには、上位 8 ビットに A5h を書き込み、下位 8 ビットに希望する値を 16 ビットユニットとして書き込んでください。

ルネサス FSP は、レジスタ書き込み保護を有効および無効にする 2 つの API (R_BSP_RegisterProtectEnable および R_BSP_RegisterProtectDisable) を提供します。

10. I/O ポートの設定

ユーザーズマニュアル ハードウェア編の「I/O ポート」には、ペリフェラルの選択と他のレジスタ設定に基づく正確なピン構成が記載されています。一般的な情報は以下のように記載されています。

リセット後、各ピンは設定が適用されるまでそのピンのデフォルト状態になることに注意することが重要です。一部のピンが望ましくない状態になる可能性がある短い期間があります。これはどの設定方法を使用しているかに関係なく当てはまります。ユーザは、これが他のシステム機能にどのように影響するかを含め、各アプリケーションに与える影響を考慮する必要があります。

10.1 マルチファンクションピン選択の設計方法

RA6 シリーズの MCU のほとんどのポートは、複数の周辺機能を実行することができます。ルネサスから提供されるピン設定ツールなどのツールを使用して、RA6 シリーズのポート選択を支援することができます。複数の周辺機能が必要な場合は、以下の設計戦略を使用して、ポート機能の選択に役立ててください。

- まず、1つのポートオプションだけで周辺機能を割り当てます。例えば、デバッグ機能のトレースデータ信号には、1つのポートオプションしかありません。この機能が必要な場合は、まずこれらのポートを割り当てます。
- 次に、限られたポートオプションで周辺機能を割り当てます。例えば、QSPI 周辺機能をサポートするデバイスでは、通常、各 QSPI 信号に対して2つのオプションしかありません。
- 複数のポート オプションを持つ周辺機能を最後に割り当てます。一例としては、通常、多数の利用可能なポートオプションを持つシリアル通信インタフェース (SCI) が挙げられます。
- 一部の周辺機能のポートオプションは交換可能ですが、他のものは論理的なグループで割り当てる必要があります。例えば、IIC 周辺機能には、信号名にサフィックス “_A” を持つポートと、サフィックス “_B” を持つポートがあります。周辺機能に対して同じサフィックスを持つポートを選択する必要があります。他の周辺機能には、この種のサフィックスがなく、ポートは交換可能に割り当てることができます。例えば、USBFS 周辺機能の USB_VBUSEN 信号です。このドキュメントの 15.3 章 も参照してください。

10.2 ポートを GPIO として設定して使用する方法

ポートを GPIO として使用するには、ポートコントロールレジスタ (PCNTR1) を使用する方法と、PmnPFS レジスタを使用する方法の2つがあります。

方法 1: ポートレジコントロールスタ (PCNTR1)

- ポートコントロールレジスタ 1 (PCNTR1) のポートディレクションビット (PDRn) に “1” を書き込むことで、ピンを出力として選択します。
- ポートディレクションビット (PDRn) は、読み書き可能です。1 “に設定すると、そのピンは出力として選択されます。I/O ポートのデフォルトの状態は “0” (入力) です。RA6 シリーズでは、ポートディレクションレジスタを読み出すことができます。
- 対応するポートコントロールレジスタ (PCNTR1) のポート出力データビット (PODRn) は、読み書き可能です。PODR を読み取ると、出力データラッチの状態 (ピンレベルではない) が読み取られます
- ポートコントロールレジスタ 2 (PCNTR2) のポート入力ビット (PIDRn) は読み取り専用です。ピンの状態を読み取るには、PCNTR2 レジスタ内の PIDRn ビットを読み取ります。

方法 2: ポート mn ピン機能選択 (PmnPFS) レジスタ

- ポートモードレジスタ (PMR) は、個々のピンが GPIO として機能するか、周辺ピンとして機能するかを指定するために読み書きされます。リセット時には、すべての PMR レジスタが 0 に設定され、すべてのピンが GPIO として機能します。PMR レジスタが 1 に設定されると、対応するピンは周辺機能に使用されません。周辺機能は、そのピンの MPC 設定で定義されます。
- ピンを出力に設定する場合、ポートの所望の出力値を最初にデータラッチに書き込み、その後、方向レジスタを出力に設定することが推奨されています。すべてのシステムで重要ではありませんが、これにより、設定中のポートに意図しない出力グリッチが発生するのを防ぎます。

一般的に、ポートを設定するために PCNTR1 を使用すると、より高速なアクセスが可能になりますが、使用可能な構成機能が少なくなります。PmnPFS レジスタを使用すると、より多くの構成機能が利用可能になりますが、アクセス速度が遅くなります。

ルネサス FSP では、以下に示すように、リセット後に GPIO ピンを構成するためのピンコンフィギュレーションが提供されています。GPIO が構成された後は、FSP の HAL レイヤー API を使用して制御することができます。

The screenshot shows the 'Pin Configuration' window for module P706. The 'P706 Configuration' section is expanded, displaying the following settings:

- Module name: P706
- Symbolic Name: (empty field)
- Comment: (empty field)
- Port Capabilities: IRQ0: IRQ07, SCI3: RXD_MISO, SCI3: SCL, SDHI1: CD, USBHS0: OVRCURB
- P706 Configuration:
 - Mode: Output mode (Initial Low)
 - Pull up: None
 - IRQ: None
 - Drive Capacity: Low
 - Output type: CMOS
- Chip input/output: P706: GPIO

図 24. FSP コンフィギュレータ を使用して P706 を出力および Low として設定

10.2.1 内蔵プルアップ

- ポート 0 から 9、A および B のほとんどのピンには、内蔵プルアップ抵抗を有効にするオプションがあります。プルアップは、各ポート mn ピン機能選択 (PmnPFS) レジスタのプルアップ (PCR) ビットによって制御されます。各 PmnPFS レジスタの PCR ビットは、ポート上の対応するピンを制御します。
- ピンは、まず PmnPFS レジスタの関連ビットで入力として設定される必要があります。PCR ビットを「1」に設定するとプルアップが有効になり、「0」に設定するとプルアップが無効となります。
- リセット後、一部を除き PCR レジスタは 0 にクリアされ、すべてのプルアップ抵抗は無効となります。
- 外部バスピン、GPIO 出力ピン、周辺機能出力ピンとして指定された場合は、自動的にプルアップがオフになります。

10.2.2 オープンドレイン出力

- 出力として設定されたピンは通常、CMOS 出力として動作します。
- ポート 0 から 9、A および B のほとんどのピンは、NMOS オープンドレイン出力として設定するオプションがあります。
- 各ポート mn ピン機能選択 (PmnPFS) レジスタにおける N チャンネルオープンドレイン制御 (NCDOR) ビットは、オープンドレインモードで動作するピンを制御します。各レジスタの該当するビットを「1」に設定すると、出力がオープンドレインになります。該当するビットを「0」に設定すると、ポートが CMOS 出力になります。

10.2.3 ポート駆動能力

- 1 から 9 ポート、A ポート、B ポートの各ピン (P200 および P201 を除く) には、低、中、高起動出力を有効にするオプションがあります。駆動能力の切り替えは、各 Port mn Pin Function Select (PmnPFS) レジスタの駆動能力制御ビット DSCR によって制御されます。
- リセット時には、すべての DSCR ビットが 0 にクリアされているため、すべてのピンが低駆動能力に設定されます。「00」以外の値を設定すると、選択したピンの出力の駆動能力が変更されます。
- 全てのピンの最大の出力合計値は 80mA です。
- 駆動能力の違いを以下に示します：

典型的な出力ピン	DSCR[1:0]	駆動能力	平均 (mA)	最大値 (mA)
ピンあたりの許容出力電流	0 0	低駆動	2.0	4.0
ピンあたりの許容出力電流	0 1	中駆動	4.0	8.0
無効な設定；使用しないでください	1 0	-	-	-
1 ピンあたり許容される出力電流	1 1	高駆動	16	32

ハイドライブ出力ピン	DSCR[1:0]	駆動能力	平均 (mA)	最大値 (mA)
ピンあたりの許容出力電流	0 0	低駆動	2.0	4.0
ピンあたりの許容出力電流	0 1	中駆動	4.0	8.0
無効な設定；使用しないでください	1 0	-	-	-
ピンあたりの許容出力電流	1 1	高駆動	20	40

出力の駆動能力は、ボード設計全体の性能に重要な影響を与えることがあります。以下の点を考慮して、各出力の駆動能力を選択する必要があります。

- 全てのピンを低駆動能力（デフォルト）に設定して性能を評価することを推奨します。
- ボードレイアウトによっては、中または高駆動能力に設定されたピンは、より高い EMI 放射を引き起こす可能性があります。
- 長いトレースの場合、信号が受信機に正しく伝播するためには、より高駆動能力が必要な場合があります。

10.3 ポート周辺機能の設定と使用

ポート mn ピン機能選択レジスタ (PmnPFS) は、各ポートの特性を設定するために使用されます。PSEL ビットは、ポートごとに選択された周辺機能を選択します

- ほとんどのピンに複数の機能があるため、RA6 シリーズにはピン機能制御レジスタ (PmnPFS) があり、ピンに割り当てられた機能を変更することができます。
- 各ピンには、それぞれ独自の PmnPFS レジスタがあります。
- 各 PmnPFS レジスタは、ピンを周辺機能 (PSEL ビット)、IRQ 入力ピン (ISEL ビット)、またはアナログ入力ピン (ASEL ビット) として使用することを可能にします。ASEL ビットが「1」（ピンをアナログ入力ピンとして使用）に設定されている場合、ピンの PMR ビットは GPIO 使用のために設定され、ピンの PDR ビットは入力に設定される必要があります。
- ユーザーズマニュアル ハードウェア編の「I/O ポート」の「各製品の周辺機器選択設定」を参照してください。
- 周辺機器のピンに予期しないエッジが入出力されないようにするためには、ピンの PmnPFS レジスタを変更する前に、対象のピンのポート モード制御 (PMR) ビットをクリアするようにしてください。
- すべての PmnPFS レジスタは、リセット時に書き込み禁止になっています。これらのレジスタに書き込むためには、最初に書き込み保護レジスタ (PWPR) を使用して書き込みを有効にする必要があります。
- PmnPFS レジスタを設定する際には、1つの機能が複数のピンに割り当てられないように注意する必要があります。ユーザはこれを行うべきではありませんが、MCU は許可します。この場合、ピン上の機能は未定義になります。
- 外部バス、イーサネットコントローラ、または USB を使用する場合、これらの周辺機器を使用する前に、MPC に追加のレジスタを設定する必要があります。
- 以下の図は、FSP ピン設定を使用して QSPI ピンを有効にする例を示しています。

Pin Configuration	
Module name:	QSPI0
Usage:	For QSPI, same Pin Group Recommended
Pin Group Selection:	_A only
Operation Mode:	Quad
Input/Output	
QSPCLK:	✓ P500
QSSL:	✓ P501
QI00:	✓ P502
QI01:	✓ P503
QI02:	✓ P504
QI03:	✓ P505

図 25. ルネサス FSP の Pin Configurator を使用した QSPI ピンの有効化

10.4 IRQ ピンの設定と使い方

- 特定のポートピンは、外部端子割り込み（IRQ）として使用することができます。お使いの MCU でどのピンが使用可能かについては、ユーザーズマニュアル ハードウェア編の「I/O ポート」の各製品の周辺機器選択設定を参照してください。
- 一部の IRQ ピンには末尾に「-DS」があるものがあります（例：IRQ1-DS）。「-DS」は、このピンがディープソフトウェアスタンバイモードから MCU を起動するために使用できることを示します。
- 注：IRQn と IRQn-DS を同時に使用することはできません。末尾に-DS が有る/無いで同じ番号の割り込みは内部的に同じ割り込みに接続されますが、外部のピン接続が異なります。
- ポートピンを IRQ ピンとして使用するためには、ピンの PFS レジスタの割り込み入力機能選択ビット（ISEL）を“1”に設定する必要があります。”
- ピンは IRQ と周辺機器の両方に同時に使用することができます。これを有効にするには、ピンの PFS レジスタの ISEL ビットと PSEL ビットの両方を設定する必要があります。
- 同じ番号の IRQ 機能は、1 つのピンでのみ有効にする必要があります。
- IRQ ピンは、以下の検出で割り込みをトリガーすることができます。
 - Low レベル
 - 立ち下がリエッジ
 - 立ち上がりエッジ
 - 立ち上がりエッジと立ち下がりの両エッジ
 どのトリガーを使用するかは、IRQ 制御レジスタ（IRQCRi）を使用して選択されます。
- IRQ ピンにはデジタルフィルタリングが有効です。このフィルタは、4 つの選択可能なクロックレート（PCLK、PCLK/8、PCLK/32、PCLK/64）のうちの 1 つで信号を繰り返しサンプリングすることに基づいています。このフィルタは、短いパルス（フィルタレートで 3 サンプル未満の High または Low パルス）をフィルタリングします。このフィルタは、これらのラインのリングやノイズをフィルタリングするのに便利ですが、機械的なスイッチバウンスのような長いイベントをフィルタリングするには、あまりにも速すぎます。フィルタリングを有効にすると、IRQ を認識するまでに短いレイテンシ（フィルタリング時間）が追加されます。
- デジタルフィルタリングは、各 IRQ ピンに対して独立して有効にすることができます。これは、IRQCRi レジスタの IRQ ピンデジタルフィルタイネーブル（FLTEN）ビットを各 IRQ に設定することで実現します。
- デジタルフィルタリング用のクロックレートは各 IRQ ピンで独立して設定可能です。これは、IRQCRi レジスタの IRQ ピンデジタルフィルタ設定（FCLKSEL[1:0]）ビットを各 IRQ に設定することで可能です。
- 図 26、図 27 にルネサス FSP を使用した IRQ ピンのイネーブルとコンフィギュレーションの例を示します。

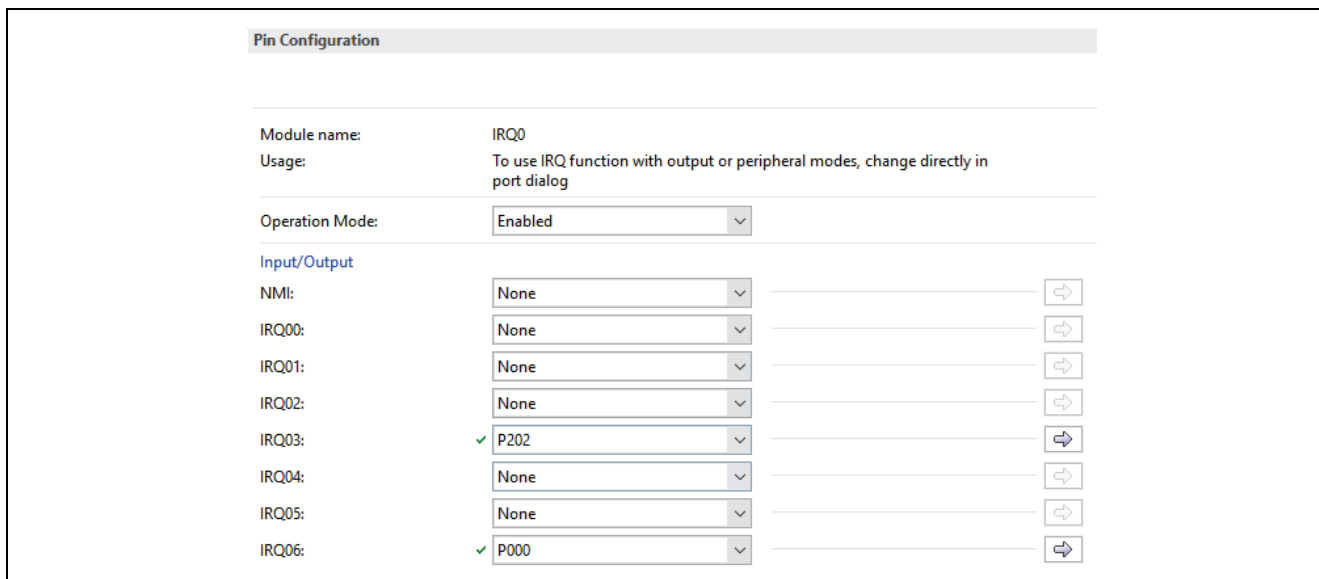


図 26. ルネサス FSP のピン設定ツールを使用し、P202 と P000 を IRQ03、IRQ06 として有効にする場合

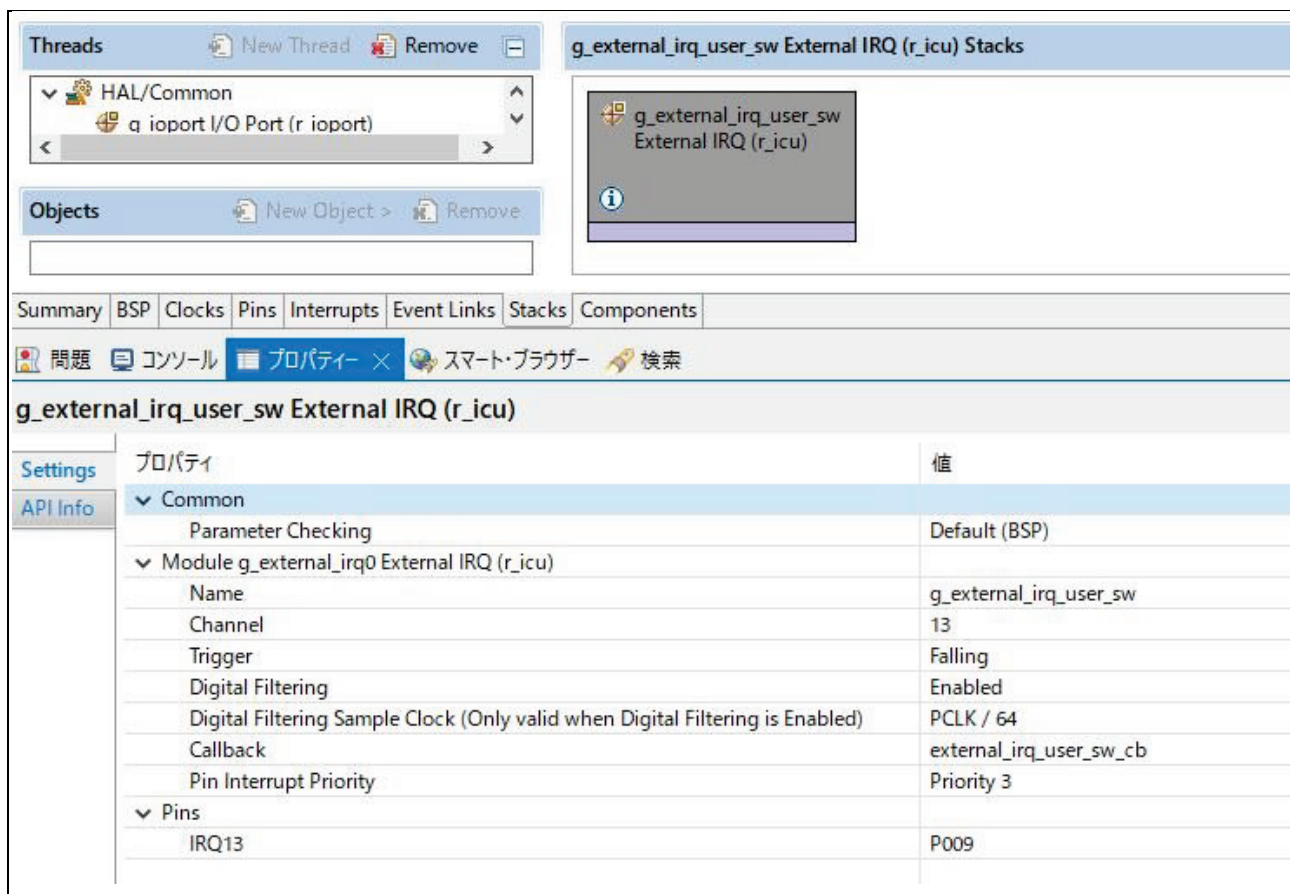


図 27. ルネサス FSP コンフィギュレータを使用した IRQ13 の設定 例

10.5 未使用ピン

注：特定のピンには特定の終端処理が必要な場合があります。MCU の I/O ポートについては、ユーザーズマニュアル ハードウェア編の「未使用ピンの処理」を参照して、特定の推奨事項を確認してください。

フローティング状態の未使用ピンは余分な電力を消費し、システムがノイズの問題により影響を受けやすくなる可能性があります。未使用ピンは、以下の方法のいずれかを使用して終端処理を行ってください。

1. 最初の選択肢は、ピンを入力に設定し（リセット後のデフォルト状態）、抵抗を使ってピンを Vcc または Vss に接続することです。どちらの接続でも MCU に差はありませんが、システムノイズの観点からは有利になる場合があります。Vss は最も一般的な選択でしょう。ピンを出力に設定するためにポートの方向レジスタに誤って書き込むと、出力がショートする可能性があるため、ピンを直接 Vcc または Vss に接続しないようにしてください。
2. 第 2 の方法は、ピンを出力に設定することです。ピンのレベルは High でも Low でもかまいませんが、ピンを出力に設定し、出力を Low にすることで、ピンを内部でグランドプレーンに接続します。これにより、システム全体のノイズ対策になる可能性があります。未使用のピンを出力に設定することの欠点は、ポートの設定をソフトウェア制御で行う必要があることです。MCU がリセットされ、方向レジスタが出力に設定されるまでの間、ピンはフローティング入力となり、余分な電流が流れる可能性があります。この間、余分な電流を許容できるのであれば、最初の方法で必要だった外部抵抗が不要になります。
3. ピンを入力として残し、外部抵抗で終端する方法は、MCU の多くのポートで利用可能な内部プルアップを使用します。これは、ピンを出力に設定するのと同じ制限（プログラムでポートを設定する必要があります）がありますが、デバイスがピンを駆動しないため、グランド、隣接ピン、Vcc への不慮のピンの短絡の影響を制限することができます。

10.6 存在しないピン

各 RA6 シリーズは、総ピン数が異なる複数のパッケージサイズで提供されています。その MCU グループの最大パッケージ（通常 176 ピン）より小さいパッケージでは、PDR レジスタの存在しないポートの対応ビットを「1」（出力）に、PODR レジスタを「0」に設定します。ユーザは、ユーザーズマニュアル ハードウェア編の「I/O ポート」の「I/O ポートの仕様」表を確認することで、各 MCU パッケージでどのポートが利用可能かを確認できます。例えば、ポート 1 の 0 と 1 のピンは、176 ピンパッケージでのみ利用可能です。存在しないピンを追加で処理する必要はないことに注意してください。

10.7 電気的特性

通常の GPIO ポートでは、CMOS レベルの入力 ($\text{High} \geq 0.8 * V_{cc}$, $\text{Low} \leq 0.2 * V_{cc}$) が必要です。一部の GPIO ポートにはシュミットトリガー入力があり、入力要件は若干異なります。詳細については、ユーザーズマニュアル ハードウェア編の「電気的特性」を参照してください。

11. モジュール停止機能

電力効率を最大化するため、RA6 シリーズの MCU では、モジュール停止制御レジスタ (MSTPCR_i, i=A, B, C, D, E) への書き込みによって、搭載の周辺機能を個別に停止することができます。モジュールが停止すると、モジュールレジスタへのアクセスは不可能になります。

リセット後、DMAC、DTC、SRAM を除くほとんどのモジュールがモジュール停止状態になります。詳細については、ユーザーズマニュアル ハードウェア編を参照してください。

周辺機器のレジスタにアクセスする前に、MSTPCR_i レジスタの対応するビットに '0' を書き込むことによって、停止モードから取り出して有効にする必要があります。

周辺機器は、MSTPCR_i レジスタの適切なビットに '1' を書き込むことで停止させることができます。

ルネサス FSP の HAL ドライバは、モジュールのスタート/ストップ機能を自動的に処理します。

12. 割り込み制御ユニット

割り込みコントローラユニット (ICU) は、どのイベント信号を NVIC、DTC、および DMAC モジュールにリンクさせるかを制御します。また、ICU はノンマスクابل割り込みの制御も行います。図 28 に ICU の仕様の例を、図 29 に I/O ピンから IRQ_i イベントを立ち上げる機能の例を示す。各 RA6 MCU グループの詳細については、「ユーザーズマニュアル ハードウェア編」を参照してください。

項目	内容	
割り込み	周辺機能割り込み	<ul style="list-style-type: none"> 周辺モジュールからの割り込み 要因数：315 (イベントリスト番号 64～511 から要因を選択)
	外部端子割り込み	<ul style="list-style-type: none"> 割り込み検出：Low レベル、立ち下がりエッジ、立ち上がりエッジ、両エッジ これらの検出法は要因ごとに1つ設定可能 デジタルフィルタ機能をサポート 16 要因 (IRQ0～IRQ15 端子からの割り込み)
	DTC および DMAC 制御	割り込み要因によって DTC と DMAC の起動が可能 (注1)
	NVIC への割り込み要因	96 要因
ノンマスクابل割り込み (注2)	NMI 端子割り込み	<ul style="list-style-type: none"> NMI 端子からの割り込み 割り込み検出：立ち下がりエッジまたは立ち上がりエッジ デジタルフィルタ機能をサポート
	発振停止検出割り込み (注3)	メイン発振器の停止を検出したときの割り込み
	WDT アンダーフロー／リフレッシュエラー (注3)	ダウン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時の割り込み
	IWDT アンダーフロー／リフレッシュエラー (注3)	ダウン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時の割り込み
	電圧監視1割り込み (注3)	低電圧検出1回路 (LVD1) の電圧監視割り込み
	電圧監視2割り込み (注3)	低電圧検出2回路 (LVD2) の電圧監視割り込み
	RPEST	SRAM パリティエラー発生時の割り込み
	RECCST	SRAM ECC エラー発生時の割り込み
	BUSST	MPU バススレーブエラー発生時の割り込み
	BUSMST	MPU バスマスタエラー発生時の割り込み
	SPEST	CPU スタックポインタモニタによる割り込み
低消費電力モードからの復帰 (注4)	<ul style="list-style-type: none"> スリープモード：ノンマスクابل割り込みまたはその他の割り込み要因によって復帰 ソフトウェアスタンバイモード：ノンマスクابل割り込みによって復帰 割り込みは WUPEN レジスタで選択可能 スヌーズモード：ノンマスクابل割り込みによって復帰 割り込みは SELSR0 および WUPEN レジスタで選択可能 	

図 28. RA6M3 ICU の仕様例

端子名	入出力	機能
NMI	入力	ノンマスクابل割り込み要求端子
IRQ0～IRQ15	入力	外部割り込み要求端子

図 29. RA6M3 ICU の I/O ピンの例

下図は、ルネサス FSP コンフィギュレータを使用して、割り込みの有効化と設定を行う場合の例です。ICU と割り込みは、HAL ドライバの構成の一部として、FSP を通して設定します。

Settings	プロパティ	値
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	Pin Output Support	Disabled
	Write Protect Enable	Disabled
	Clock Source	PCLKD
	▼ Module gpt0_timer_sw_debounce_filter Timer, General PWM (r_gpt)	
	> General	
	> Output	
	> Input	
	▼ Interrupts	
	Callback	gpt0_timer_debounce_filter_cb
	Overflow/Crest Interrupt Priority	Priority 11
	Capture A Interrupt Priority	Disabled
	Capture B Interrupt Priority	Disabled
	Underflow/Trough Interrupt Priority	Disabled
	> Extra Features	
	> Pins	

図 30. GTP0 オーバーフロー割り込みを有効にし、割り込みサービスルーチンによって呼び出されるユーザーコールバック関数を設定

13. 低消費電力

RA6 シリーズには、消費電力を削減するためのいくつかの機能があります。これには、クロック分周器の設定、EBCLK 出力制御、モジュールの停止、ノーマルモードでの電源制御モードの選択、および低消費電力モードへの遷移が含まれます。詳細については、ユーザーズマニュアル ハードウェア編の「低消費電力モード」を参照してください。

RA6 シリーズは、MCU グループによって 4 種類の低消費電力モードをサポートしています。これらは以下の通りです

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

消費電力削減のために利用できる機能の概要は、次の表のとおりです。

表 15. 低消費電力モードの仕様

アイテム	仕様
クロック信号の変更による消費電力の削減	周波数分周比は、システムクロック (ICLK)、周辺モジュールクロック (PCLKH、PCLKL、PCLKADC、PCLKGPT)、外部バスクロック (BCLK)、フラッシュインタフェースクロック (FCLK) に対して、独立して選択できます。*1
EBCLK 出力制御	BCLK 出力またはハイレベル出力を選択できます。
モジュールストップ	各ペリフェラルモジュールについて、個別に機能を停止できます
低消費電力モード	<ul style="list-style-type: none"> スリープモード ソフトウェアスタンバイモード スヌーズモード ディープソフトウェアスタンバイモード
電源制御モード	電源制御モード3つの動作電源制御モード: <ul style="list-style-type: none"> ハイスピードモード ロースピードモード サブオシレータースピードモード
TrustZone フィルタ*2	セキュリティ属性を各レジスタに設定できます

注: 1. 詳細はユーザーズハードウェアマニュアルの「クロック発生回路」を参照してください
2. TrustZone セキュリティ機能をサポートするデバイスの場合

以下の表は、低消費電力モードへの移行条件、CPU と周辺モジュールの状態、および各モードの解除方法を示しています。

表 16. 低消費電力モード

動作状態*1	スリープモード	全モジュールクロック停止モード	ソフトウェアスタンバイモード	ディープソフトウェアスタンバイモード
遷移条件	SBYCR. SSBY=0 時に WFI 命令	SBYCR. SSBY=1 かつ DPSBYCR. DPSBY=0 時に WFI 命令	スヌーズ要求トリガー SNZCR. SNZE=11	SBYCR. SSBY=1 かつ DPSBYCR. DPSBY=1 時に WFI 命令
解除方法	すべての割り込み。このモード内で使用可能な全てのリセット	このモードで定義された割り込み。このモードで使用可能なりセット。	このモードで定義された割り込み。このモードで使用可能なりセット。	このモードで定義された割り込み。このモードで使用可能なりセット。
割り込み解除後の状態	プログラム実行状態 (割り込み処理中)	プログラム実行状態 (割り込み処理中)	プログラム実行状態 (割り込み処理中)	リセット状態
リセット解除後の状態	リセット状態	リセット状態	リセット状態	リセット状態

注: 1. 詳細については、ユーザーズマニュアル ハードウェア編の表「各低電力モードの動作条件」を参照してください。

RA6 シリーズには、通常モードおよびスリープモードで低消費電力動作するように MCU を設定するレジスタ設定が含まれています。これらのモードは動作電力制御モードと呼ばれ、OPCCR レジスタによって制御されます。

以下は、動作電力消費制御モードと、各モード下での最大許容クロッキングおよび電圧レベルの概要です。

表 17. 各動作消費電力制御モードで使用可能な発振器

モード	発振器						
	PLL, PLL2	高速オンチップオシレータ	中速オンチップオシレータ	低速オンチップオシレータ	メインクロック発振器	サブクロック発振器	IWDT 専用オンチップオシレータ
High-Speed	可能	可能	可能	可能	可能	可能	可能
Low-Speed	不可能	可能	可能	可能	可能	可能	可能
Bubosc-Speed	不可能	不可能	不可能	可能	不可能	可能	可能

注:OPCCR レジスタの値を低消費電力動作モードのいずれかに設定することは可能ですが、クロックや電圧レベルを所望のモードの要件に合わせて設定する必要があります。そうでない場合、OPCCR レジスタの設定は消費電力を下げる効果を持ちません。

最も低い電力数値を実現するために、クロック生成回路には可能な限り大きな分周器を使用する。

低消費電力モードは、RES ピンリセット、電源オンリセット、電圧モニタリセット、周辺機器割り込みなど、様々な割り込みソースによって解除されます。異なる低消費電力モードの割り込みソースのリストについては、ユーザズマニュアル ハードウェア編の「低消費電力モード」を参照してください。

スヌーズモードのみ、ソフトウェアスタンバイモードからスヌーズモードに移行するためのスヌーズ要求がトリガーとなります。他の低消費電力モードへの遷移は、スタンバイ制御レジスタ (SBYCR) を適切に設定した上で WFI 命令を実行することで行われます。

ルネサス FSP は、低消費電力モード (低消費電力モード) ドライバとドライバーコンフィグレータを提供しており、低消費電力モード、ウェイクソース/キャンセルソースなどを設定することができます。

プロパティ	値
▼ Common	
Parameter Checking	Default (BSP)
Standby Limit	Disabled
▼ Module g_lpm_deep_sw_standby Low Power Modes (
▼ General	
Name	g_lpm_deep_sw_standby
Low Power Mode	Deep Software Standby mode
Output port state in standby and deep standby	No change
▼ Standby Options	
> Wake Sources	
> Snooze End Sources	
Snooze Request Source	RXD0 falling edge
DTC state in Snooze Mode	Disabled
Snooze Cancel Source	None
▼ Deep Standby Options	
> Cancel Sources	
> Cancel Edges	
I/O Port Retention	Maintain the IO port states
Power-Supply Control	Supply power to the Standby SRAM, low-speed on-chip oscillator, AGTn and USBFS resume detect...

図 31. ルネサス FSP コンフィギュレータによる低消費電力モードの設定

FSP コンフィギュレータで特定の低消費電力モードモードを設定した後、低消費電力モードドライバの API を使用して低消費電力モードドライバを初期化し、設定された低消費電力モードモードに MCU を配置することができます:

```
/* Open LPM driver and initialize LPM mode */
err = R_LPM_Open(&g_lpm_ctrl_instance_ctrls[g_lpm_transition_pos],
                &g_lpm_ctrl_instance_cfgs[g_lpm_transition_pos]);
/* Handle error */
if (FSP_SUCCESS != err)
{
    return (err);
}
/* Transition to configured LPM mode */
err = lpm_mode_enter(g_lpm_transition_sequence[g_lpm_transition_pos]);
/* Handle error */
if (FSP_SUCCESS != err)
{
    return (err);
}
```

14. 外部バス

RA6 シリーズは、外部バスコントローラを有します。RA6 シリーズの中には、SDRAM コントローラを内蔵しているものもあります。

14.1 バス幅とマルチプレクス

外部メモリ領域のアクセス幅は 8 ビットまたは 16 ビットに設定することができます。幅の設定は、CSnCR レジスタまたは SDC 制御レジスタ (SDCCR) の BSIZE ビットを設定することにより、チップセレクトごとに設定されます。CSnCR レジスタの MPXEN ビットを設定することにより、チップセレクト領域のアドレスとデータラインをマルチプレクスすることができます。

14.2 バス信号の駆動能力

外部メモリ領域を使用する場合、バス信号を制御するピンは、高速設定時に高駆動能力出力になるように設定する必要があります。ピンの駆動容量の設定については、「I/O ポート」の「ポート mn ピン機能選択レジスタ」、および「MCU ユーザーズマニュアル ハードウェア編」の「電気特性」を参照してください。

14.3 バスエラー

各バスでは、以下のようなエラーが発生する可能性があります：

- 不正なアドレスアクセス
- バスマスタ MPU エラー
- トラストゾーン フィルターエラー (Trustzone をサポートする MCU のみ)
- 各スレーブ IP から送信されたバスエラー

バスエラーが発生すると、動作は保証されず、エラーは要求元のマスター IP に返されます。各マスターで発生するバスエラーは、BUSnERRADD および BUSnERRSTAT レジスタに格納されます。これらのレジスタは、リセットによってのみクリアする必要があります。詳細については、ユーザーズマニュアル ハードウェア編の「バスエラーアドレスレジスタ (BUSnERRADD)」と「バスエラーステータスレジスタ (BUSnERRSTAT)」を参照してください。

注：DMAC と DTC はバスエラーを受信しないため、バスエラーによる影響はありません。

15. 一般的なレイアウトの実践

15.1 デジタルドメインとアナログドメインの比較

RA6 シリーズは、主に 3 種類のピン機能を備えています：電源、デジタル、アナログです。

一般に、電源ピンは電圧と基準入力専用であり、複数の機能を持つことはありません。電源ピンは通常、MCU 内の特定の部分、つまりドメインに特化しています。例えば、MCU の主電源電圧は、デジタルコア、デジタル周辺機能、デジタル I/O ピンの多くに電力を供給することになります。デジタルドメインは、デジタル回路、デジタル I/O ピン、関連する電源ピンで構成されます。アナログ機能用に指定された電源ピン

(AVCCO、関連する AVSSO など) は、デジタルドメイン回路とは別に、MCU 内の特定のアナログ回路に電力を供給します。アナログドメインは、アナログ回路、アナログ I/O ピン、関連する電源ピンから構成されません。

デジタル信号は通常、周期的なクロックに関連する繰り返しのスイッチングパターンです。デジタル信号の遷移は、比較的鋭いエッジになる傾向があり、遷移間の High または Low レベルが安定しています。各信号は、指定された時間内に、論理状態として許容できる電圧レベルで安定している必要があります。信号の状態は、通常、所定のクロック間隔でサンプリングされ、関連するデータ信号を評価するためにクロックのエッジ遷移を使用します。デジタル信号の電圧レベルのわずかな変動は、レベルが指定された範囲内にとどまる限り、通常は許容されます。しかし、デジタル信号に対する大きな外部からの影響は、データがサンプリングされる瞬間に誤った論理状態になる可能性があります。

アナログ信号は通常は全く異なっています。アナログ信号は周期的かもしれませんが、アナログ信号の評価は通常、論理状態ではなく、一定範囲内の電圧の測定です。アナログ信号の電圧レベルは、特定のトリガーイベントに基づいてサンプリングされ、その結果得られる測定値は、MCU 内のアナログ回路を使用して処理されます。アナログ測定の精度は、サンプリングされた電圧レベルの精度に直接関係しています。アナログ入力信号の電圧レベルをわずかも変化させる可能性のある不要な外的影響は、測定の精度に影響を与える可能性があります。

RA6 シリーズの I/O ピンは非常に多重化されているため、多くの I/O ピンはアナログまたはデジタル機能のどちらかにも使用することができます。このため、デジタルとアナログの機能が重複し、データエラーが発生することがあります。

デジタルとアナログの信号ドメイン間で起こりうる問題を最小限に抑えるために、以下のガイドラインを考慮してください。

- I/O ピンの機能を割り当てる際には、アナログピンとデジタルピンが物理的にできるだけ分離されるようにピン機能を選択してください。
- 各アナログ信号はできるだけ他の信号から分離してください。
- PCB 配線は、各アナログ信号をできるだけ分離してください。同じ領域に他の信号（アナログまたはデジタル）を配線しないようにしてください。
- アナログ電源電圧とアナログ基準電圧には、適切な AC フィルタが含まれていることを確認します。これは、MCU 電圧ピンの近くにある推奨コンデンサや、適切な誘導フィルタの形式であるかもしれません。目的は、電圧供給と基準電圧をほぼリップルのない状態で提供することです。
- PCB 設計で専用の電源層を使用する場合、アナログ電圧領域でデジタル信号を配線しないようにし、デジタル電圧領域でアナログ信号を配線しないようにしてください。

高感度なアプリケーションでは、回路設計が性能に及ぼす影響を理解するために、シミュレーションツールを使って特定の設計を評価することが強く推奨されます。例えば、精密センサー設計や非常に高速なデジタルバスインタフェースなどのアプリケーションが含まれます。各周辺機能の特定の要件については、ユーザーズマニュアル ハードウェア編の「電気的特性」を参照してください。

15.2 高速信号設計の考慮事項

デジタル信号のクロック速度が上がると、外部刺激の影響がより重要になることがあります。一部の周辺機能は「高速」デジタル信号として分類されることがあります。高速デジタル信号に対しては、追加の設計上の考慮が必要です。

クロストークとは、ある信号の遷移が、近くにある別の信号に誘導的な影響を与える状態のことです。このクロストークの影響が十分強い場合、最初の信号が 2 番目の信号のエラーを引き起こす可能性があります。クロストークの影響を軽減するために、以下の一般的な PCB 配線ガイドラインを使用してください。

- 同じ配線層で配線された信号間に十分なスペースを確保します。通常、同じデジタルグループの信号間は最低 1 トレース幅以上、異なるデジタルグループの信号間は最低 3~5 トレース幅以上のスペースを確保します。
- 同一配線層上のクロック信号とデータ信号の間には、十分なスペースを確保します。通常、クロックとその他のデジタル信号の間には、最低でも 3~5 トレース幅のスペースを確保します。
- 隣接する配線層でのデジタル信号の並列配線は避けてください。隣接する信号層で配線する必要がある場合は、可能な限り、直交交差を使用するようにしてください。

可能であれば、信号層の間に電源層またはグランド層を使用して、PCB 信号層を分離してください。電源層やグランド層の固体銅は、デジタル信号の「シールド」として機能することができます。

標準化されたインタフェースには、それぞれ特定の要件があります。PCB が信号クロストーク問題を回避するように設計されていることを確認するために、設計の各インタフェースの関連規格を参照することを強くお勧めします。

15.3 信号グループの選択

一部のピン名には、末尾に信号グループを示す_A、_B、_C が追加されています。IIC、SPI、SSIE、ETHERC、SDHI など特定の周辺機能を割り当てる場合は、同じ末尾を持つ機能ピンを選択してください。ユーザーズマニュアル ハードウェア編の「電気的特性」に示されている AC タイミング特性を信号グループごとに測定する場合があります。信号グループが混在している場合、その周辺機器は機能しない可能性があります。記載されている AC タイミング特性は適用されない場合があります。

周辺機能のピン名に信号グループに末尾_A、_B がない場合、各機能の信号に対して最も使いやすいピン配置を選択することが安全です。

ユーザーズマニュアル ハードウェア編「I/O ポート」の「周辺機器選択設定」、「PmnPFS レジスタ設定に関する注意事項」を参照してください。

16. 参照

以下の文書はこのクイック設計ガイドの作成に使用されました：

参照	文書番号	概要
1	R01UH0884JJ0100	ルネサス RA6M1 グループ、ユーザーズマニュアル：ハードウェア編
2	R01UH0885JJ0100	ルネサス RA6M2 グループ、ユーザーズマニュアル：ハードウェア編
3	R01UH0886JJ0100	ルネサス RA6M3 グループ、ユーザーズマニュアル：ハードウェア編
4	R01UH0890JJ0110	ルネサス RA6M4 グループ、ユーザーズマニュアル：ハードウェア編
5	R01UH0891JJ0110	ルネサス RA6M5 グループ、ユーザーズマニュアル：ハードウェア編
6	R20AN0577JG0101	RA Arm® TrustZone® Tooling Primer
7	R11AN0467JU0100	Renesas RA Security Design with Arm TrustZone – IP Protection
8	R11AN0468JU0100	Renesas RA Securing Data at Rest Using the Arm® TrustZone®
9	R11AN0475JU0100	Establishing and Protecting Device Identity using SCE9 and Arm® TrustZone®
10	R11AN0469JU0110	Renesas RA Family Device Lifecycle Management Key Installation

ウェブサイトとサポート

以下のバニティ URL にアクセスして、RA ファミリの主要な要素を学び、コンポーネントや関連するドキュメントをダウンロードし、サポートを受けることができます。

RA 製品情報	www.renesas.com/ra
RA 製品サポート フォーラム	www.renesas.com/ra/forum
RA フレキシブル ソフトウェア パッケージ	www.renesas.com/FSP
ルネサスのサポート	www.renesas.com/support

改訂記録

改訂版	日付	概要	
		ページ	要約
1.00	Nov. 20. 23	-	初版リリース

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。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、お客様の責任において、お客様の機器・システムを設計ください。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みます。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. R 当社製品または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. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 5. 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。
 6. 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。
 7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、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。
 9. 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。
 10. 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。
 11. 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。
 12. 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なります。
 13. 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。
 14. 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。

(Rev.5.0-1 October 2020)

本社

豊洲フォレシア、豊洲 3-2-24、
135-0061 東京都江東区豊洲 3-2-24 豊洲フォレシア

www.renesas.com

商標

Renesas および Renesas ロゴは、ルネサス エレクトロニクス株式会社の商標です。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。

お問い合わせ先

製品、技術、ドキュメントの最新版、最寄りの営業所などに関する詳しい情報は、こちらをご覧ください：www.renesas.com/contact/。