

RA ファミリ、M16C ファミリ

M16C から RA への置き換えガイド クロック・低消費電力機能編

要旨

本アプリケーションノートでは、RA ファミリ、M16C ファミリのクロック機能・低消費電力機能の置き換えについて説明します。

対象デバイス

- RA ファミリ
- M16C ファミリ

M16C ファミリから RA ファミリへの置き換え例として、RA ファミリは RA6M3 グループを、M16C ファミリは M16C/65C シリーズを用いて説明しています。本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

RA ファミリと M16C ファミリ間で使用している用語が一部異なります。

クロックに関する用語の相違点を下表に示します。

表 RA ファミリと M16C ファミリ間の用語の相違点

項目	RA	M16C
CPU の動作クロック	システムクロック(ICLK)	CPU クロック
周辺機能の動作クロック	周辺モジュールクロック (PCLKA、PCLKB、PCLKC、 PCLKD)	周辺機能クロック (fC、fC32、fOCO40M、fOCO-F、 fOCO-S、f1)
メインクロック発振回路 の端子	EXTAL、XTAL	XIN、XOUT
消費電力を低減するモード	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	ウェイトモード ストップモード
周辺機能のレジスタ	I/O レジスタ	SFR

目次

1. クロック発生回路の相違点	3
2. クロックの機能および設定手順の相違点	5
2.1 メインクロックの発振安定待機時間の考え方	6
2.2 クロックの設定手順の相違点	6
2.2.1 メインクロックをシステムクロックに設定する手順の例	7
2.2.2 HOCO クロックをシステムクロックに設定する手順の例	9
2.2.3 サブクロックをシステムクロックに設定する手順の例	11
2.2.4 PLL クロックをシステムクロックに設定する手順の例	13
3. 低消費電力モードの相違点	15
3.1 スリープモード	15
3.2 ソフトウェアスタンバイモード	15
3.3 スヌーズモード	15
3.4 ディープソフトウェアスタンバイモード	15
4. 動作電力低減機能について	18
5. クロック周波数精度測定回路について	18
6. 発振停止検出機能について	19
7. WFI 命令のタイミング	19
8. RA ファミリユーザーズマニュアルハードウェア編の関連する章	19
9. 付録	20
9.1 M16C ファミリから RA ファミリへ置き換えるときのポイント	20
9.1.1 割り込み	20
9.1.2 入出力ポート	21
9.1.3 モジュールストップ機能	21
9.2 Flexible Software Package (FSP)	22
10. 参考ドキュメント	23
改訂記録	24

1. クロック発生回路の相違点

RA ファミリ、M16C ファミリのクロック発生回路の相違点を説明します。

RA ファミリと M16C ファミリでは、使用できるクロックの周波数に違いがあります。表 1-1 に各クロック周波数の相違点を示します。

RA ファミリでは、システムクロック、周辺モジュールクロック、フラッシュ用クロック(FlashIF クロック)、外部バスクロックに対して、それぞれ個別に分周を設定することができます。また、M16C ファミリでは、システムクロックと周辺モジュールクロック、フラッシュ用クロック、外部バスクロックは、同じクロックとなります。

図 1-1 に各クロックの選択イメージを示します。

表 1-1 各クロック周波数の相違点

項目		RA (RA6M3)	M16C
最大動作周波数	システムクロック	120MHz	32MHz
	周辺モジュールクロック	120MHz(PCLKA, PCLKD) 60MHz(PCLKB, PCLKC)	32MHz
	外部バスクロック	120MHz	32MHz ^(注 1)
	グラフィック LCD ピクセルクロック	54MHz (パラレル RGB) 60MHz (シリアル RGB)	—
周波数	メインクロック	8MHz~24MHz	2MHz~20MHz
	サブクロック	32.768kHz	32.768kHz~50kHz
	PLL	120MHz~240MHz	10MHz~32MHz
	高速オンチップオシレータ(HOCO)	16MHz/18MHz/20MHz	40MHz
	中速オンチップオシレータ(MOCO)	8MHz	—
	低速オンチップオシレータ(LOCO)	32.768kHz	125kHz
	USB クロック	48MHz	—
	CAN クロック	8~24MHz	—
	AGT クロック	32.768kHz	—
	RTC クロック	32.768kHz	—
	IWDT クロック	15kHz	—
	WDT 周期	約 128μs~4096sec ^(注 2)	約 16.384ms~33.6sec ^(注 3)
リセット解除後のクロック		中速オンチップオシレータ	低速オンチップオシレータ
リセット後の発振状態	メインクロック	停止	動作
	サブクロック	動作 ^(注 4)	停止
	高速オンチップオシレータ(HOCO)	動作/停止 ^(注 5)	停止
	中速オンチップオシレータ(MOCO)	動作 ^(注 4)	—
	低速オンチップオシレータ(LOCO)	動作	動作
フラッシュ用クロック		FlashIF クロック	CPU クロック

注 1 たゞし、25MHz を超えるとデータ出力保持時間が 0ns 以下になります。(VCC=5V の場合)

- 注 2 CPU の動作クロックが、PLL クロック 50MHz のとき最短、サブクロック 32.768kHz のとき最長にできます。
- 注 3 CPU の動作クロックが、PLL クロック 32MHz のとき最短、LOCO クロック 125kHz のとき最長にできます。
- 注 4 使用しない場合は、停止させる処理が必要です。
- 注 5 オプション機能選択レジスタ 1 の HOCO 発振有効ビット (OFS1.HOCOEN) でリセット後の状態を選択できます。

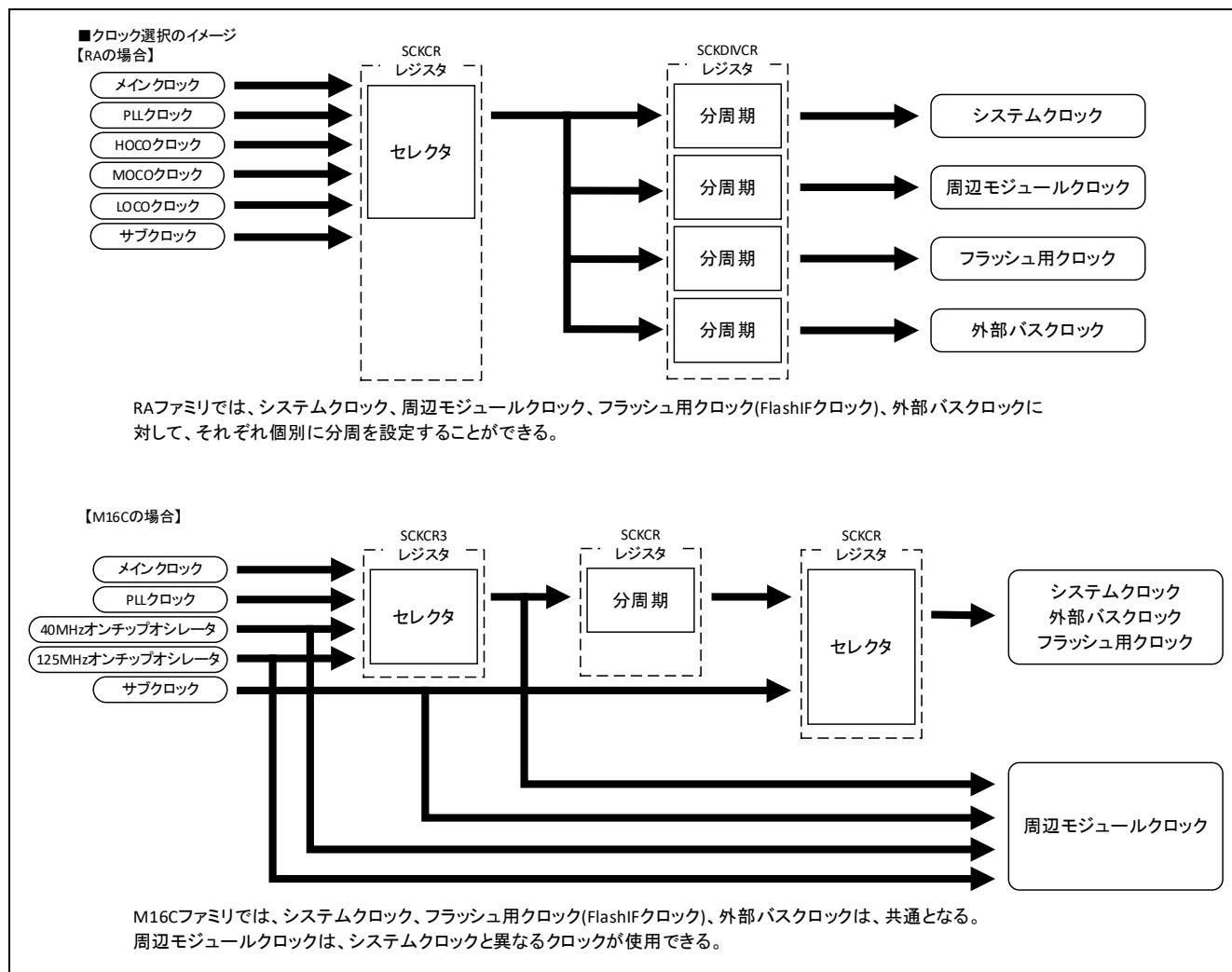


図 1-1 各クロックの選択イメージ

2. クロックの機能および設定手順の相違点

RA ファミリ、M16C ファミリのクロックの機能および設定手順の相違点を説明します。

RA ファミリでは、ウェイトコントロールレジスタが搭載されており、クロックの発振動作を開始させてから、CPU 内部にクロックを供給するまでの時間を調節することができます。これによって、クロックの発振が十分に安定した状態で CPU 内部に供給することができ、マイコンの誤動作を防ぎます。ウェイトコントロールレジスタは、低消費電力モードへの遷移後、復帰の際にも機能します。

ウェイトコントロールレジスタの設定と発振安定待機時間の考え方は、「2.1 メインクロックの発振安定待機時間の考え方」で説明します。

RA ファミリ、M16C ファミリのクロック設定の手順は、「2.2 クロックの設定手順の相違点」で説明します。

表 2-1 にクロック設定手順の掲載する項を示します。

表 2-1 クロック設定手順の掲載する項

内容	掲載している項
メインクロックをシステムクロックに設定	2.2.1
HOCO クロックをシステムクロックに設定する手順	2.2.2
サブクロックをシステムクロックに設定する手順	2.2.3
PLL クロックをシステムクロックに設定する手順	2.2.4

2.1 メインクロックの発振安定待機時間の考え方

RA ファミリのメインクロックの発振安定待機時間の考え方を説明します。

メインクロックのウェイトコントロールレジスタ(MOSCWTCR)は、「発振子メーカーが推奨するメインクロックの発振安定時間以上の値」を設定します。

メインクロック発振安定待機時間は、ユーザがソフトウェアによって待つ必要があります。ソフトウェアループなどを作成して、十分な時間待つようにしてください。発振安定フラグレジスタをもつマイコンをご使用の場合は、それぞれの発振安定フラグを確認することで、発振の安定を確認することができます。メインクロックの発振安定待機時間は「MOSCWTCR レジスタで設定したサイクル期間の 2 倍以上」待つことを推奨します。

図 2-1 にメインクロックの発振安定待機時間の考え方を示します。

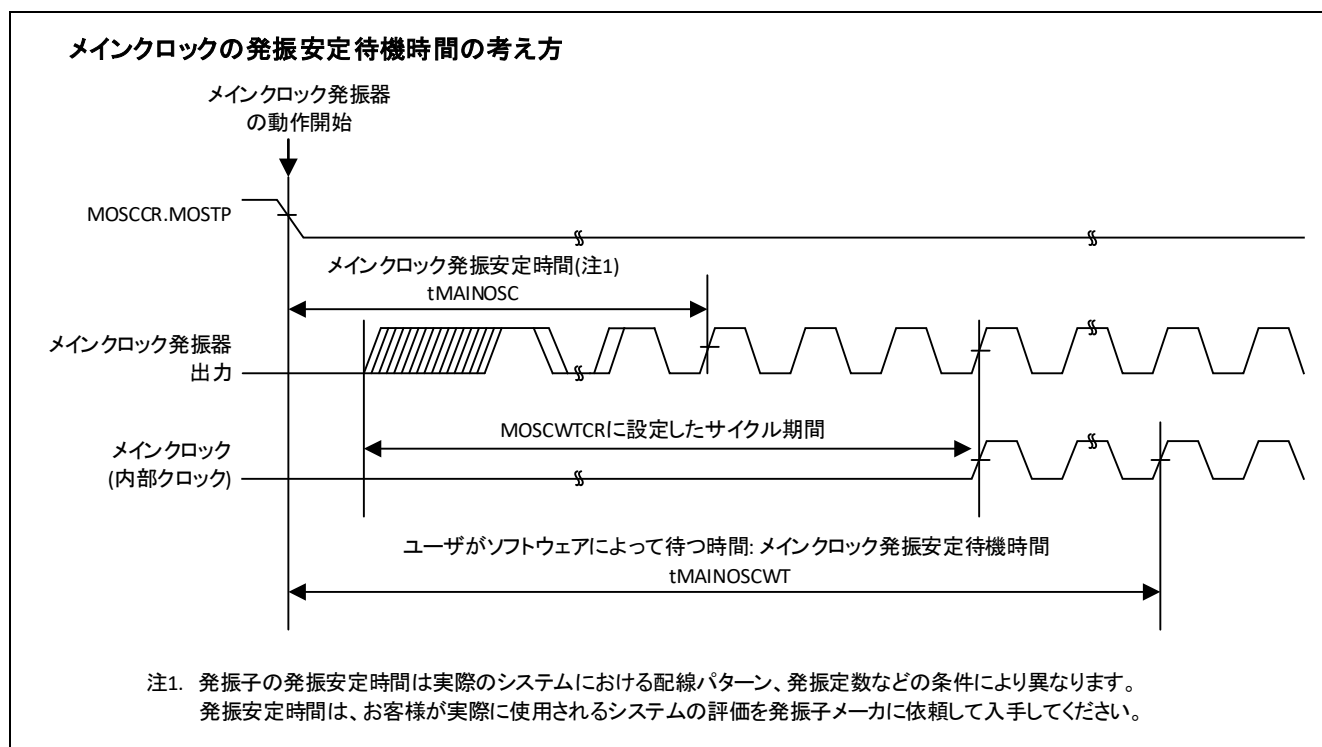


図 2-1 メインクロックの発振安定待機時間の考え方

2.2 クロックの設定手順の相違点

以下に同一設定条件下で RA ファミリ、M16C ファミリのリセット解除後のクロック設定手順の相違点を示します。

2.2.1 メインクロックをシステムクロックに設定する手順の例

メインクロックをシステムクロックに設定する手順(サブクロックをシステムクロックにも RTC にも使用しない場合)の例を示します。

表 2-2 にシステムクロックの設定条件(メインクロック)、表 2-3 にメインクロックに設定する手順の相違点を示します。

表 2-2 システムクロックの設定条件(メインクロック)

項目		条件
システムクロック		メインクロック
分周	システムクロック	分周なし
	周辺モジュールクロック	分周なし
	FlashIF クロック	分周なし
	外部バスクロック	分周なし
メインクロック	発振器発振源	発振子
	ウェイト時間	8194μs
サブクロック		使用しない

表 2-3 メインクロックに設定する手順の相違点

手順		RA (RA6M3)	M16C
1	プロテクト解除	R_SYSTEM->PRCR = 0xA503;	prc0 = 1;
2	サブクロックの停止	R_SYSTEM->SOSCCR = 0x01; while (0x01 != R_SYSTEM->SOSCCR) { }	—
3	メインクロック発振器のドライブ能力を設定	R_SYSTEM->MOMCR = 0x00;	cm15 = 1;
4	メインクロック発振器のウェイトコントロールレジスタを設定	R_SYSTEM->MOSCWTCR = 0x07; ^(注1)	—
5	メインクロック発振器を動作	R_SYSTEM->MOSCCR = 0x00; while (0x00 != R_SYSTEM->MOSCCR) { }	cm05 = 0;
6	発振安定待ち	while (R_SYSTEM->OSCSF_b.MOSCSF != 1) { }	メインクロック発振安定待ち
7	分周の設定	R_SYSTEM->SCKDIVCR = 0x00000000; while (0x00000000 != R_SYSTEM->SCKDIVCR) { }	—

手順		RA (RA6M3)	M16C
8	システムクロックの切り替え	<pre>R_SYSTEM->SCKCR = 0x03; while (0x03 != R_SYSTEM->SCKCR) { }</pre>	<pre>cm06 = 1; cm11 = 0; cm21 = 0; cm07 = 0;</pre>
9	分周の設定	—	<pre>cm17 = 1; cm06 = 0; cm1 = cm1 ^ (0xC0); cm16 = 0;</pre>
10	プロテクト設定	<pre>R_SYSTEM->PRCR = 0xA500;</pre>	<pre>prc0 = 0;</pre>

注 1 発振子メーカーが推奨するメインクロックの発振安定時間以上の値を設定してください。例は、24MHz のメインクロックの発振安定時間が 4.2ms の場合に、約 8.19ms 待つようにした場合の設定値です。

2.2.2 HOCO クロックをシステムクロックに設定する手順の例

HOCO クロックをシステムクロックに設定する手順(サブクロックをシステムクロックにも RTC にも使用しない場合)を示します。

表 2-4 にシステムクロックの設定条件(HOCO クロック)、表 2-5 に HOCO クロックに設定する手順の相違点を示します。

表 2-4 システムクロックの設定条件(HOCO クロック)

項目		条件
システムクロック		HOCO クロック
分周	システムクロック	分周なし
	周辺モジュールクロック	2 分周
	FlashIF クロック	2 分周
	外部バスクロック	2 分周
HOCO	周波数	20MHz(RA) ^(注1) / 40MHz(M16C)
	ウェイト時間	274.32μs
	リセット後の発振	リセット後、HOCO 発振が無効 (OFS1 レジスタの HOCOEN ビットが“1”)
サブクロック		使用しない

注1 オプション機能選択レジスタ 1 の HOCO 周波数設定 0 ビット (OFS1. HOCOFRQ0) で選択できません。

表 2-5 HOCO クロックに設定する手順の相違点

手順		RA (RA6M3)	M16C
1	プロテクト解除	R_SYSTEM->PRCR = 0xA503;	prc0 = 1;
2	サブクロックの停止	R_SYSTEM->SOSCCR = 0x01; while (0x01 != R_SYSTEM->SOSCCR) { }	—
3	HOCO のウェイトコントロールレジスタの設定	R_SYSTEM->HOCOWTCR = 0x06;	—
4	HOCO を動作	R_SYSTEM->HOCOCR = 0x00;	fra00 = 1;
5	発振安定待ち	while (R_SYSTEM->OSCSF_b.HOCOSF != 1) { }	40MHz オンチップオシレータが安定するまで(tsu(fOCO40M))待つ
6	分周の設定	R_SYSTEM->SCKDIVCR = 0x10011111; while (0x10011111 != R_SYSTEM->SCKDIVCR) { }	—
7	システムクロックの切り替え	R_SYSTEM->SCKCR = 0x00; while (0x00 != R_SYSTEM->SCKCR) { }	cm06 = 1; fra01 = 1; cm21 = 1; cm07 = 0;

手順		RA (RA6M3)	M16C
8	分周の設定	—	cm17 = 1; cm06 = 0; cm1 = cm1 ^ (0xC0);
9	プロテクト設定	R_SYSTEM->PRCR = 0xA500;	prc0 = 0;

2.2.3 サブクロックをシステムクロックに設定する手順の例

サブクロックをシステムクロックに設定する手順(サブクロックを RTC に使用しない場合)を示します。表 2-6 にシステムクロックの設定条件(サブクロック)、表 2-7 にサブクロックに設定する手順の相違点を示します。

表 2-6 システムクロックの設定条件(サブクロック)

項目		条件
システムクロック		サブクロック
分周	システムクロック	分周なし
	周辺モジュールクロック	分周なし
	FlashIF クロック	分周なし ^(注 1)
	外部バスクロック	分周なし
サブクロック	発振器駆動能力	通常駆動

注 1 サブクロックがシステムクロックのときは、フラッシュ書き換えできません。

表 2-7 サブクロックに設定する手順の相違点

手順		RA (RA6M3)	M16C
1	プロテクト解除	R_SYSTEM->PRCR = 0xA503;	prc0 = 1;
2	サブクロックを接続するポートを入力ポートに設定	—	pu21 = 0; pd8_6 = 0; pd8_7 = 0;
3	サブクロックの発振安定待ち	サブクロックの発振安定待機時間待ち ^(注 1)	サブクロック発振安定時間を待つ
4	サブクロック発振器を停止	R_SYSTEM->SOSCCR = 0x01; while (0x01 != R_SYSTEM->SOSCCR) { }	—
5	サブクロック発振器停止待ち	サブクロックの 5 サイクル経過待ち	—
6	サブクロックのドライブ能力を設定	R_SYSTEM->SOMCR = 0x00;	cm03 = 1;
7	サブクロック発振器を動作	R_SYSTEM->SOSCCR = 0x00; while (0x00 != R_SYSTEM->SOSCCR) { }	cm04 = 1;
8	サブクロックの発振安定待ち	サブクロックの発振安定待機時間待ち ^(注 1)	サブクロック発振安定時間を待つ
9	分周の設定	R_SYSTEM->SCKDIVCR = 0x00000000; while (0x00000000 != R_SYSTEM->SCKDIVCR) { }	—

手順		RA (RA6M3)	M16C
10	システムクロックの切り替え	<pre>R_SYSTEM->SCKCR = 0x04; while (0x04 != R_SYSTEM->SCKCR) { }</pre>	<pre>cm16 = 1; cm1 = cm1^(0xC0); cm06 = 1; cm07 = 1;</pre>
11	プロテクト設定	<pre>R_SYSTEM->PRCR = 0xA500;</pre>	<pre>prc0 = 0;</pre>

注 1 サブクロック発振安定待機時間は、「SOSCWTCR レジスタで設定したサイクル期間の 2 倍以上」の時間をソフトウェアで待ってください。

2.2.4 PLL クロックをシステムクロックに設定する手順の例

PLL クロックをシステムクロックに設定する手順(サブクロックをシステムクロックにも RTC にも使用しない場合)を示します。

表 2-8 にシステムクロックの設定条件(PLL クロック)、表 2-9 に PLL クロックに設定する手順の相違点を示します。

表 2-8 システムクロックの設定条件(PLL クロック)

項目		条件
システムクロック		PLL クロック
分周	システムクロック	2 分周
	周辺モジュールクロック	4 分周
	FlashIF クロック	4 分周
	外部バスクロック	4 分周
メインクロック	発振器発振源	発振子
	ウェイト時間	8194μs
PLL クロック	PLL 入力分周比／通倍率	2 分周 20 通倍(RA) / 2 分周 8 通倍(M16C)
サブクロック		使用しない

表 2-9 PLL クロックに設定する手順の相違点

手順		RA (RA6M3)	M16C
1	プロテクト解除	R_SYSTEM->PRCR = 0xA503;	prcr = 0x01;
2	サブクロックの停止	R_SYSTEM->SOSCCR = 0x01; while (0x01 != R_SYSTEM->SOSCCR) { }	—
3	メインクロック発振器のドライブ能力を設定	R_SYSTEM->MOMCR = 0x00;	cm15 = 1;
4	メインクロック発振器のウェイトコントロールレジスタを設定	R_SYSTEM->MOSCWTCR = 0x07; ^(注1)	—
5	メインクロック発振器を動作	R_SYSTEM->MOSCCR = 0x00; while (0x00 != R_SYSTEM->MOSCCR) { }	cm05 = 0;
6	発振安定待ち	while (R_SYSTEM->OSCSF_b.MOSCSF != 1) { }	メインクロック発振安定待ち
7	PLL 入力分周比および PLL クロックソース、周波数通倍率を設定	R_SYSTEM->PLLCCR = 0x2701; ^(注2)	plc0 = 0x14; ^(注2)

手順		RA (RA6M3)	M16C
8	PLL クロック発振器のウェイトコントロールレジスタを設定	—	—
9	PLL を動作	R_SYSTEM->PLLCR = 0x00;	plc07 = 1;
10	発振安定待ち	while (R_SYSTEM->OSCSF_b.PLLSF != 1) { }	PLL クロックが安定するまで(tsu(PLL)) 待つ
11	分周の設定	R_SYSTEM->SCKDIVCR = 0x21022222; while (0x21022222 != R_SYSTEM->SCKDIVCR) { }	—
12	システムクロックの切り替え	R_SYSTEM->SCKCR = 0x05; while (0x05 != R_SYSTEM->SCKCR) { }	cm06 = 1; cm11 = 1; cm21 = 0; cm07 = 0
13	分周の設定	—	cm17 = 1; cm06 = 0; cm1 = cm1 ^ (0xC0); cm16 = 0;
14	プロテクト設定	R_SYSTEM->PRCR = 0xA500;	prc0 = 0;

注 1 発振子メーカーが推奨するメインクロックの発振安定時間以上の値を設定してください。例は、24MHz のメインクロックの発振安定時間が 4.2ms の場合に、弊社推奨の約 2 倍の約 8.19ms 待つようにした場合の設定値です。

注 2 RA ファミリでは、PLL 入力分周比選択ビットによる分周後の周波数を 8~24MHz に、周波数通倍率設定ビットによる通倍後の周波数を 120~240MHz の範囲に入るように設定してください。M16C ファミリでは、基準周波数カウンタ設定ビットによる分周後の周波数は 2MHz~5MHz に、PLL 通倍率選択ビットによる通倍後の周波数は f(PLL)の範囲内にしてください。

3. 低消費電力モードの相違点

RA ファミリでは、消費電力を低減させる機能として、複数の低消費電力モードを用意しています。ここでは、RA6M3にある、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の4つのモードと、M16C ファミリのウェイトモード、ストップモードとの相違点を説明します。

表 3-1 に RA ファミリと M16C ファミリの低消費電力モードの相違点を示します。

3.1 スリープモード

スリープモードは、M16C ファミリのウェイトモードと類似した機能で、CPU を停止させるモードです。RA ファミリでは、スリープモードからの復帰時に、自動的にクロックソースを切り替える機能があります。

3.2 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は、M16C ファミリのストップモードと類似した機能で、CPU とほとんどの周辺機能、および、発振器を停止させるモードです。

3.3 スヌーズモード

ソフトウェアスタンバイモード時に、スヌーズ制御回路がスヌーズ要求を受信すると、MCU はスヌーズモードへ遷移します。このモードでは、CPU が復帰していなくても一部の周辺モジュールは動作します。

3.4 ディープ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は、CPU、周辺機能、および、発振器の電源供給を停止させるモードです。電源供給を停止させるため、大幅に消費電流を小さくすることができます。また、このモードでも、リアルタイムクロックを動作させることができます。ディープソフトウェアスタンバイモードの解除は、リセットでのみ行うことができます。

表 3-1 RA ファミリと M16C ファミリの低消費電力モードの相違点

項目	RA (RA6M3)				M16C	
消費電力	大 → 小				大 → 小	
モード	スリープモード	スヌーズモード	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	ウェイトモード	ストップモード
CPU	－(保持)	－(保持)	－(保持)	－(不定)	－(保持)	－(保持)
メインクロックその他のクロック	○	－	－	－	○	－(保持)
サブクロック	○	○	○	○	○	－(保持)
RAM	○	○	－(保持)	－(不定)	○	○
スタンバイ SRAM	○	○	－(保持)	－(保持/不定)	機能なし	機能なし
フラッシュメモリ	○	－(保持)	－(保持)	－(保持)	○	－(保持)
DMA コントローラ	○	動作禁止	－(保持)	－(不定)	機能なし	機能なし
データトランスファコントローラ	○	○	－(保持)	－(不定)	機能なし	機能なし
USB	○	動作禁止	－(保持)	－(保持/不定)	機能なし	機能なし
ウォッチドッグタイマ	○	－(保持)	－(保持)	－(不定)	○	－(保持)
独立ウォッチドッグタイマ	○	○	○	－(不定)	機能なし	機能なし
リアルタイムクロック	○	○	○	○	○	－(保持)
低消費電力非同期汎用タイマ	○	○	○	○	機能なし	機能なし
12 ビット A/D コンバータ	○	○	－(保持)	－(不定)	機能なし	機能なし
プログラマブルゲインアンプ	○	○	○	－(不定)	機能なし	機能なし
12 ビット D/A コンバータ	○	○	－(保持)	－(不定)	機能なし	機能なし
静電容量式タッチセンシングユニット	○	○	－(保持)	－(不定)	機能なし	機能なし
データ演算回路	○	○	－(保持)	－(不定)	機能なし	機能なし

項目	RA (RA6M3)				M16C	
消費電力	大 → 小				大 → 小	
モード	スリープモード	スヌーズモード	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	ウェイトモード	ストップモード
シリアルコミュニケーションインタフェース 0	○	○	－(保持)	－(不定)	機能なし	機能なし
シリアルコミュニケーションインタフェース 1~9	○	動作禁止	－(保持)	－(不定)	機能なし	機能なし
I2C バスインタフェース 0	○	○	○	－(不定)	機能なし	機能なし
I2C バスインタフェース 1, 2	○	動作禁止	－(保持)	－(不定)	機能なし	機能なし
イベントリンクコントローラ	○	○	－(保持)	－(不定)	機能なし	機能なし
高速アナログコンパレータ	○	○	○	－(不定)	機能なし	機能なし
IRQn (n=0~15) 端子割り込み	○	○	○	－(不定)	機能なし	機能なし
NMI、IRQn-DS (n=0~14) 端子割り込み	○	○	○	○	機能なし	機能なし
キー割り込み機能	○	○	○	－(不定)	機能なし	機能なし
その他の周辺機能	○	動作禁止	－(保持)	－(不定)	○	－(保持)

○：動作可能、－：停止、(保持)：内部レジスタの内容が保持されます、(不定)：内部レジスタの内容が不定です

注1 電源供給を停止するため、CPU、内蔵周辺機能（RTC アラーム、RTC 周期、USB サスペンド／レジューム検出部は除く）のレジスタ内容、RAM（スタンバイ SRAM は除く）のデータが不定となります。

4. 動作電力低減機能について

RA ファミリには、動作中の消費電力を抑える機能として、M16C ファミリにはない低消費電力機能が搭載されています。

低消費電力機能には、High-speed モード、Low-speed モード、Subosc-speed があり、低速になるほど消費電力を抑えられます。モードごとに使用できる電源電圧、クロック、周波数が異なりますので、使用条件に応じて、適切なモードを選択してください。クロックを低速にする場合と高速にする場合で、動作電力制御モードを変更する手順が異なります。

- クロックを低速にして CPU の消費電力を抑えたい場合
 1. クロックソース、分周比の切り替え設定する。
 2. 動作電力制御モードを変更する。
- クロックを高速にして CPU を高速で動作させたい場合
 1. 動作電力制御モードを変更する。
 2. クロックソース、分周比の切り替え設定する。

M16C ファミリの通常動作モードに、高速モード、中速モード、低速モードなどの類似の用語がありますが、これは、CPU が動作するクロックの違いを指すもので、RA ファミリのものと異なります。

5. クロック周波数精度測定回路について

RA ファミリには、M16C にはないクロックの周波数を監視し、周波数の異常を検出できる機能が搭載されています。RA6M3 には、クロック周波数精度測定回路(CAC)が搭載されています。

クロック周波数精度測定回路(CAC)は、MCU 外部から入力される基準信号や他のクロックソースをもとにクロックの周波数を監視し、測定の終了または設定した範囲を外れた場合に割り込みを発生する機能です。

例えば、内蔵のオンチップオシレータでサブクロックの周波数を監視し、サブクロックに異常が発生し停止した場合に、割り込みを発生させることができます。

6. 発振停止検出機能について

クロックの発振停止機能の相違点を示します。

RA ファミリと M16C ファミリで発振停止検出後のクロックなど一部機能が異なります。

表 6-1 に発振停止検出機能の相違点を示します。

表 6-1 発振停止検出機能の相違点

発振停止時のクロック	発振停止検出後のクロック	
	RA (RA6M3)	M16C
メインクロック	中速オンチップオシレータ	低速オンチップオシレータ
サブクロック	変わらない (サブクロックのまま)	
低速オンチップオシレータ	変わらない (低速オンチップオシレータのまま)	
中速オンチップオシレータ	変わらない (中速オンチップオシレータのまま)	
高速オンチップオシレータ	変わらない (高速オンチップオシレータのまま)	
PLL クロック	変わらない (PLL クロックのまま ^(注 1))	変わらない (PLL クロックのまま ^(注 1))

注 1 ただし、周波数は、自励発振周波数となります。

7. WFI 命令のタイミング

RA ファミリの低消費電力モードへの遷移には、WFI 命令を実行する必要があります。

WFI 命令は、I/O レジスタと CS/SDRAM 領域の書き込みが完了する前に実行されることがあり、その場合、低消費電力モードへの遷移に遷移しないなどの意図しない動作を起こす恐れがあります。これは、I/O レジスタまたは CS/SDRAM 領域への書き込み直後に WFI 命令が実行された場合に生じます。この問題を避けるには、書き込まれたレジスタまたは CS/SDRAM 領域を読み戻して、書き込みの完了を確認してください。例として、MSTPCRB レジスタを WFI 命令の実行前に読み込むと、I/O レジスタへの書き込みを完了する期間を確保することが可能です。

8. RA ファミリユーザズマニュアルハードウェア編の関連する章

M16C ファミリから RA ファミリに置き換えるときは、ユーザズマニュアルハードウェア編の以下の章を参考にしてください。

- クロック発生回路
- 消費電力低減機能
- レジスタライトプロテクション
- リアルタイムクロック (RTC)

9. 付録

9.1 M16C ファミリから RA ファミリへ置き換えるときのポイント

M16C ファミリから RA ファミリへ置き換えるときのポイントについて、以下に示します。

9.1.1 割り込み

RA ファミリでは、下記の条件を満たすときに割り込みを受け付けることができます。

- 割り込みセットイネーブルレジスタ (NVIC) を設定する。
- IELSRn.IELS[8:0]ビットを割り込み要因として設定する。

表 9-1 に、RA ファミリと M16C ファミリの割り込みの発生条件についての比較表を示します。

表 9-1 RA ファミリと M16C ファミリの割り込みの発生条件についての比較表

項目	RA (RA6M3)	M16C
I フラグ	-	I フラグを“1”(許可)にすると、マスカブル割り込みの受け付けが許可されます。
割り込み要求フラグ	周辺機能から割り込み要求があると、“1”(割り込み要求あり)になります。	
割り込み優先レベル	NVIC->IP レジスタで設定します。	ILVL2~ILVL0 ビットで設定します。
割り込み要求許可	NVIC->ISER レジスタで設定します。	-
周辺機能の割り込み許可	R_ICU->IELSR レジスタで設定します。	-

詳細は、ユーザーズマニュアル ハードウェア編の割り込みコントローラユニット(ICU)、CPU、使用する周辺機能の章を参照ください。

Arm® NVIC の内部レジスタについては、ARM® Cortex®-M4 Processor Technical Reference Manual (ARM DDI 0439D) を参照してください。

9.1.2 入出力ポート

RA ファミリでは、周辺機能の入出力信号を端子に割り当てるには、I/O ポートの PFS レジスタの設定を行う必要があります。

RA ファミリの端子の入出力制御を行う前に以下の 2 つのビットの設定を行ってください。

- PSEL ビット：該当端子に割り当てる周辺機能の選択
- PMR ビット：該当端子に汎用入出力ポート/周辺機能を割り当てるかの選択

表 9-2 に RA ファミリと M16C ファミリの周辺機能端子の入出力設定についての比較表を示します。

表 9-2 RA ファミリと M16C ファミリの周辺機能端子の入出力設定についての比較表

機能	RA (RA6M3)	M16C
端子の機能選択	PSEL ビットを設定することで、周辺機能の入出力を複数の端子から選択して割り付けることができます。	M16C ファミリにはありません。 ^(注 1) 各周辺機能のモードを設定すると、周辺機能の入出力端子として割り付けられます。
汎用入出力ポート/周辺機能の切り換え	PMR ビットを設定することで、対象端子を I/O ポートとして使用するか、周辺機能として使用するかを選択できます。	

注 1 M32C ファミリ、R32C ファミリには、同様の機能のレジスタがあります。

詳細は、ユーザーズマニュアル ハードウェア編の I/O ポートの章を参照ください。

9.1.3 モジュールストップ機能

RA ファミリでは、周辺モジュールごとに機能を停止させることが可能です。

使用しない周辺モジュールをモジュールストップ状態へ遷移させることで、消費電力を低減することができます。

リセット解除後は、一部を除く全てのモジュールがモジュールストップ状態になっています。

モジュールストップ状態のモジュールのレジスタは、読み書きできません。

詳細は、ユーザーズマニュアル ハードウェア編の消費電力低減機能の章を参照ください。

9.2 Flexible Software Package (FSP)

Renesas RA ファミリの Flexible Software Package (FSP) は、RA ファミリを用いた組み込みシステムを開発するためのソフトウェアパッケージです。

プログラミングとデバッグを簡単かつ迅速に行うための直感的なコンフィグレータおよびインテリジェントなコード生成をサポートします。

- RA Flexible Software Package Documentation
[Clock Generation Circuit \(r_cgc\)](#)

10. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RA6M3 グループ ユーザーズマニュアル ハードウェア編 (R01UH0886)

M16C/65C グループ ユーザーズマニュアル ハードウェア編 (R01UH0093)

RA6M3 グループ、M16C/65C グループ以外の製品をご使用の場合は、それぞれの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。

(最新版をルネサスエレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新の情報をルネサスエレクトロニクスホームページから入手してください。)

ユーザーズマニュアル：開発環境

Renesas Flexible Software Package (FSP) User's Manual (R11UM0155EU, <https://renesas.github.io/fsp/>)

M16C シリーズ, R8C ファミリ C コンパイラパッケージ V5.45

C コンパイラユーザーズマニュアル Rev.3.00

(最新版をルネサスエレクトロニクスホームページから入手してください。)

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

www.renesas.com

お問合せ先

www.renesas.com/contact/

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2022.06.24	-	初版発行

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違くと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

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. 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）、交通管制（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、これに限られません。）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おりません。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しします。
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 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

商標について

ルネサスおよびルネサスロゴはルネサス 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。

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/