

RX FAMILY ハードウェアマニュアルガイド (周辺機能編)

2026/02 REV 2.00
ルネサスエレクトロニクス株式会社

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て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 2020.10)

本資料の目的

- 本資料はハードウェアマニュアルに記載の機能の使い方を簡潔にわかりやすくまとめたものです。
- ハードウェアマニュアルと合わせてご使用になることで、機能に対する理解をより一層深めていただくことを目的に作成しています。
- アプリケーションノートに記載している章もございますので、より詳細な使用方法はページに記載のアプリケーションノートをご参照ください。
- 本資料はマニュアルに記載のすべての情報を網羅しているわけではございません。各機能の注意事項など詳細な情報は必ず該当製品のハードウェアマニュアルをご参照ください。

ハードウェアマニュアルガイド 周辺機能編一覧

本資料に掲載した機能情報を示します。カッコ内は参考にした機能名称です。機能は随時追加予定です。

■ 消費電力低減機能	ページ 05
■ 割り込みコントローラ	ページ 12
■ 外部バスコントローラ	ページ 36
■ I/Oポート	ページ 58
■ イベントリンクコントローラ(ELC)	ページ 69
■ マルチファンクションタイムパルスユニット(MTU3)	ページ 89
■ ポートアウトプットイネーブル(POE3)	ページ 110
■ 汎用PWMタイマ(GPTW)	ページ 121
■ ウォッチドッグタイマ/独立ウォッチドックタイマ(WDT/IWDT)	ページ 150

ハードウェアマニュアルガイド 電気的特性編に関しては[こちら](#)

消費電力低減機能

製品別消費電力低減機能

- 製品ごとの消費電力低減機能は以下の表となっています。

*RX26Tのみ製品群A

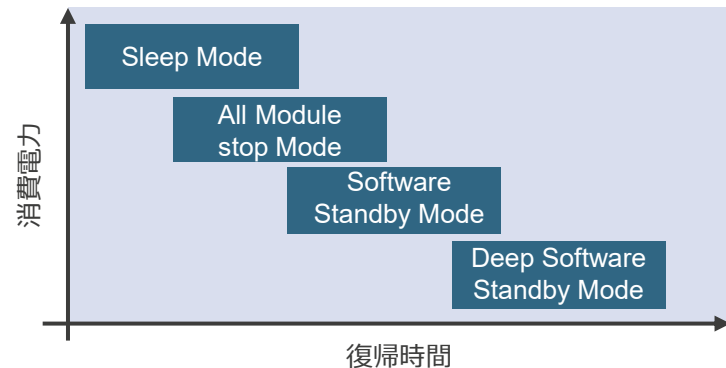
主な製品		スリープ	ディープスリープ	全モジュールクロックストップ	ソフトウェアスタンバイ	ディープソフトウェアスタンバイ	スヌーズ
製品群A	RX600,700シリーズ	✓	—	✓	✓	✓	—
製品群B	RX100,200シリーズ*	✓	✓	—	✓	—	✓

製品によりサポートするモードがない場合がありますので、詳細は各製品のユーザーズマニュアルハードウェア編をご参照ください。

製品群Aの消費電力低減機能概要

停止(保持) : 内部レジスタ値保持・内部状態は動作中断を示します
 停止(不定) : 内部レジスタ値不定・内部状態は電源オフを示します

低消費電力モード	CPU	発振器 / 電源管理				周辺モジュール						メモリ		I/Oポート
		Main clock	HOCO LOCO PLL	Sub-clock RTC LVD	POR	IWDT	WDT	USB	TMR POE	REMC	その他	SRAM	Standby RAM	
スリープ	停止 (保持)	選択可	選択可	選択可	動作	選択可	停止 (保持)	選択可	選択可	選択可	選択可	選択可	選択可	動作
全モジュール クロックストップ	停止 (保持)	選択可	選択可	選択可	動作	選択可	停止 (保持)	停止*2	選択可	停止 (保持)	停止 (保持)	停止 (保持)	停止 (保持)	停止 (保持)
ソフトウェア スタンバイ	停止 (保持)	選択可 または停止*1	停止	選択可	動作	選択可	停止 (保持)	停止*2	停止 (保持)	選択可	停止 (保持)	停止 (保持)	停止 (保持)	停止 (保持)
ディープ ソフトウェア スタンバイ	停止 (不定)	選択可 または停止*1	停止	選択可	動作	停止 (不定)	停止 (不定)	停止*3	停止 (不定)	選択可	停止 (不定)	停止 (不定)	停止 (保持/不定)	停止 (保持)



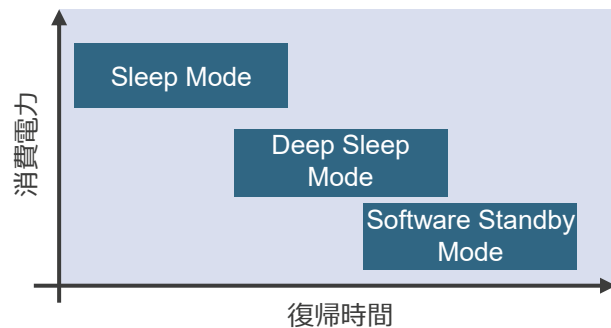
- *1 : 製品により異なります
- *2 : レジューム機能(は有効)
- *3 : レジューム機能の有効/無効を選択可

製品群Bの消費電力低減機能概要

停止(保持)：内部レジスタ値保持・内部状態は動作中断を示します

低消費電力モード	CPU	発振器 / 電源管理				周辺モジュール						SRAM	I/Oポート
		Main clock	HOCO LOCO PLL	Sub-clock RTC LVD	POR	IWDT	WDT	LPTコンパレータB REMC RTCCOUT出力 CLKOUT出力	DMAC	DTC	その他		
スリープ	停止(保持)	選択可	選択可	選択可	動作	選択可	停止(保持)	選択可	選択可	選択可	選択可	選択可	動作
ディープスリープ	停止(保持)	選択可	選択可	選択可	動作	選択可	停止(保持)	選択可	停止(保持)	停止(保持)	選択可	停止(保持)	動作
ソフトウェアスタンバイ	停止(保持)	停止	停止	選択可	動作	選択可	停止(保持)	選択可	停止(保持)	停止(保持)	停止(保持)	停止(保持)	停止(保持)
スヌーズ*1	停止(保持)	選択可	選択可	選択可	動作	選択可	停止(保持)	選択可	停止(保持)	選択可	選択可	選択可(保持)	動作

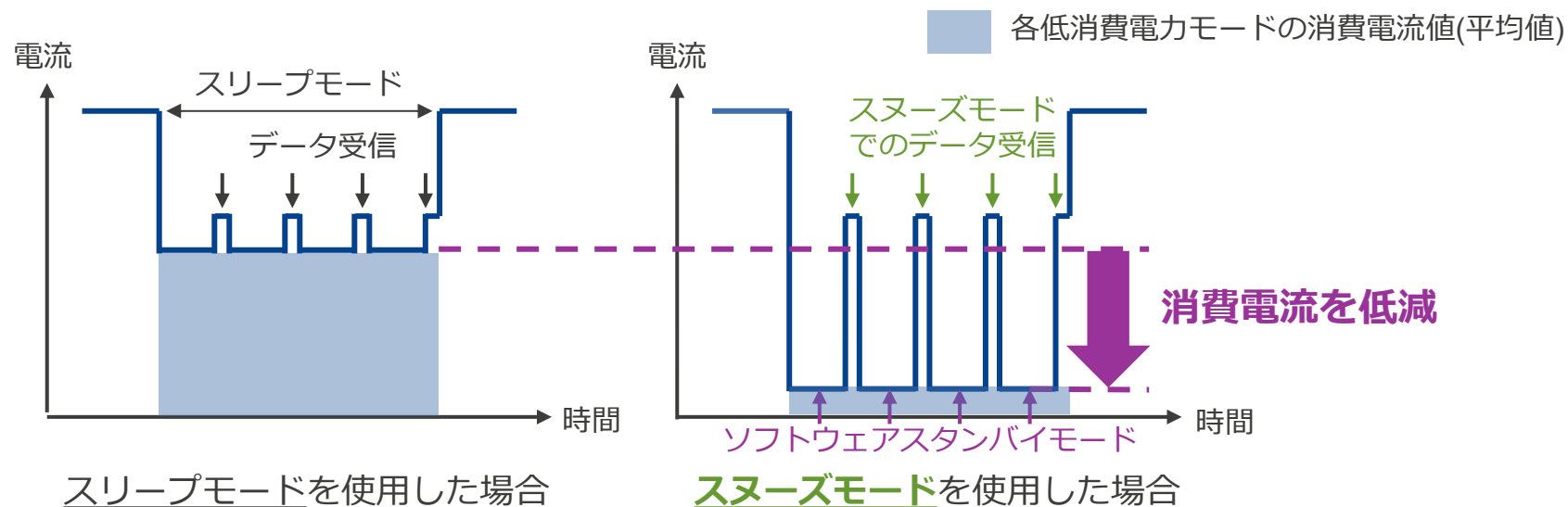
*1 スヌーズ機能については次ページで解説



スヌーズモード

- ソフトウェアスタンバイモード中に、一部の周辺機能を一時的に動作させることが可能（間欠動作）
- ソフトウェアスタンバイとスヌーズモードで間欠動作をすることでスリープモード動作に比べ、トータルの消費電流はmA→uAオーダーまで低減*が可能

*APN “RX ファミリ スヌーズモードの使用例 [R01AN5914](#)”での実測値



消費電力低減のコツ

ここでは、RX140を例に消費電力を低減するコツを紹介します。

①高速動作モード、中速動作モード、低速動作モードを活用する。

スリープモードやディープスリープモードに移行する際にシステムの動作周波数を下げることで、さらなる消費電力低減が可能です。

②使用していない周辺モジュールを停止する

全モジュールを動作させた場合と全モジュールを停止した場合では消費電流に大きな差が生じるので、不要なモジュールを停止することで消費電力を抑えることができます。

③I/Oポート

出力選択をしているI/Oポートは下記の処理を行うことで消費電流を低減することが可能です。

- プルアップ抵抗に接続している場合は出力レベルをHigh、プルダウン抵抗に接続している場合は出力レベルをLowにする。

例) スリープモード、全周辺モジュール動作時の消費電流比較 (typ値)

	高速動作 (48MHz)	中速動作 (8MHz)	低速動作 (32.768KHz)
消費電流	4.0 mA	1.5 mA	3.8 uA

例) 48MHz駆動スリープモード時における周辺モジュール動作/停止の消費電流比較 (typ値)

	全周辺モジュール動作	全周辺モジュール停止
消費電流	4.0 mA	1.4 mA

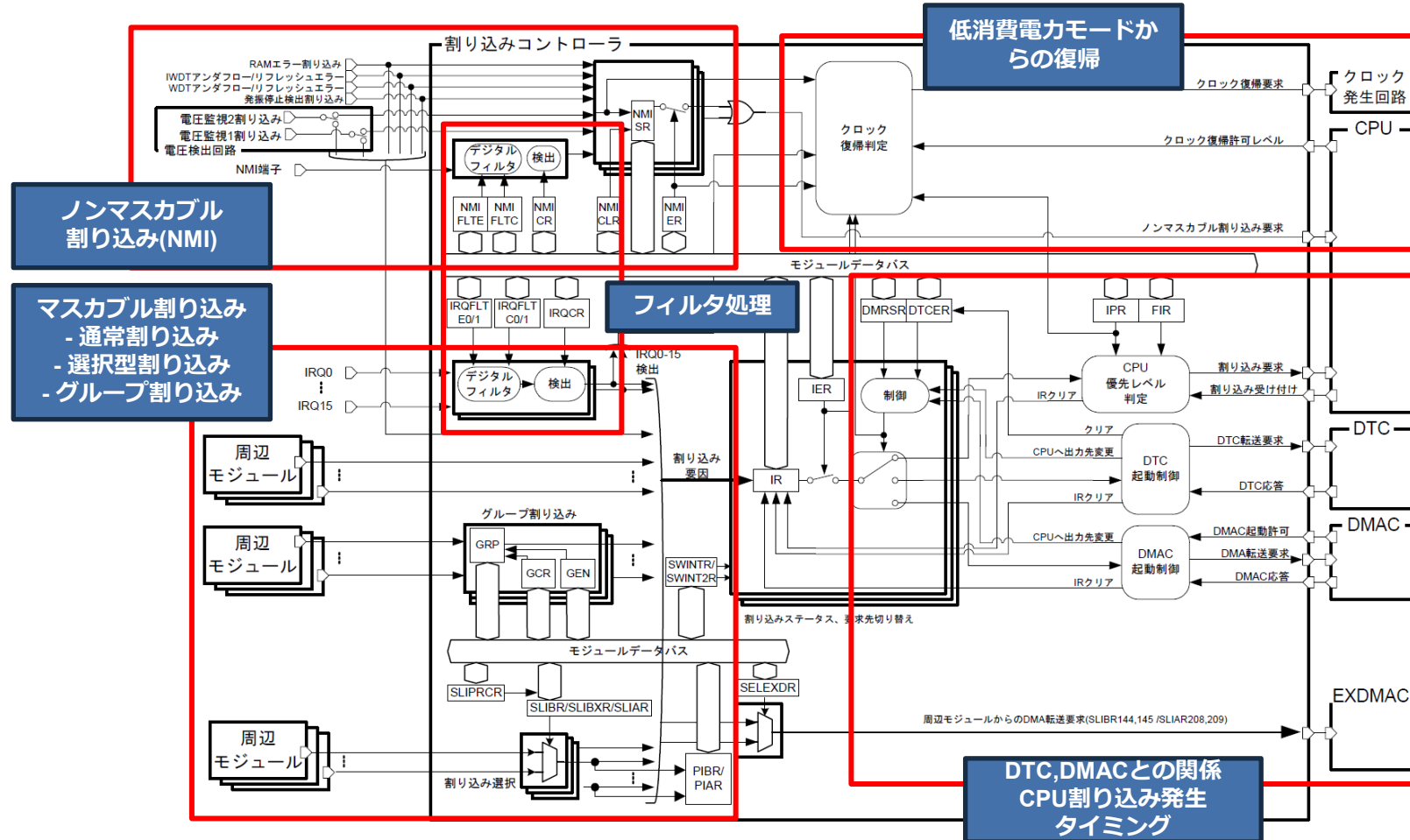
消費電力低減機能のアプリケーションノート

- 消費電力低減機能の詳細な使用方法は下記をご参照ください
 - RXファミリ 低消費電力モードへの移行例 [R01AN4347](#)
 - RXファミリ スヌーズモードの使用例 [R01AN5914](#)

割り込みコントローラ

割り込みコントローラ機能仕様一覧

割り込みコントローラの構成と5つのポイントを以下に示します。
本資料では5つのポイントを順に説明します。



ノンマスクابل割り込み

- CPUの異常を検知し、システムに致命的な障害が発生したと考えられる場合に発生する割り込みです。RXでは下記の割り込みをすべてまとめてノンマスクابل割り込みとして定義しています。

■ ノンマスクابل割り込みの種類

NMI端子

発振停止検出

WDT(IWDT)アンダフロー
/リフレッシュエラー

電圧監視1/2

倍精度浮動小数点例外

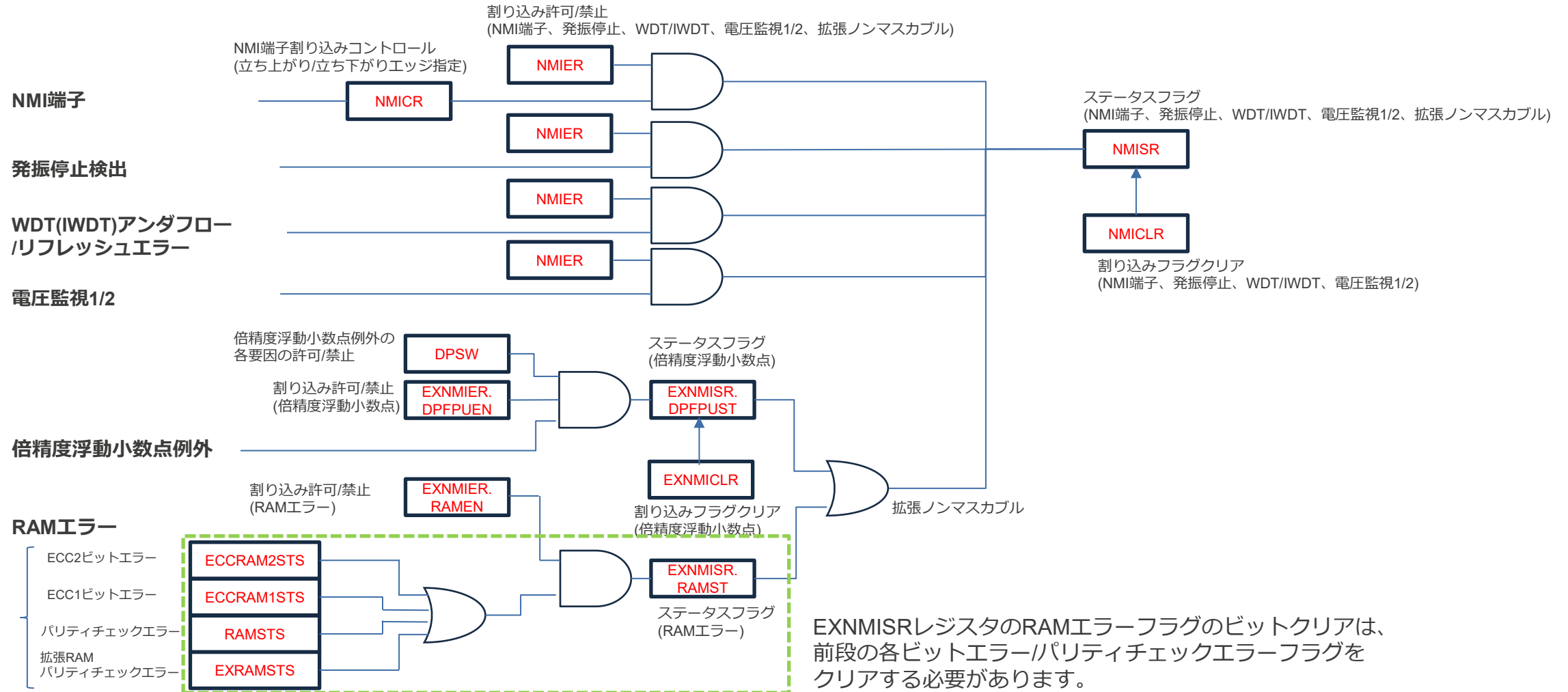
RAMエラー

- ノンマスクابل割り込みの割り込み優先レベルはすべて15(最高)です。
- ノンマスクابل割り込みは、PSWのIビットやIPLビットの影響を受けず、必ず受け付けられます。
- ノンマスクابل割り込みのベクタテーブルは例外ベクタテーブル上にあります。
- ノンマスクابل割り込みでDTCやDMACを起動することはできません。
- ノンマスクابل割り込みが発生した際はシステム異常が起きている可能性がありますので、割り込みから元の処理に戻らない仕組みをご検討ください。
- NMI端子以外はマスクابل割り込みとして使用することも可能です。(この場合のベクタテーブルは割り込みベクタテーブル上にあります)

ノンマスクابل割り込み

ノンマスクابل割り込み構造

ノンマスクابل割り込みの構造と設定が必要なレジスタ名を以下に示します。レジスタの詳細はマニュアルをご参照ください。



マスクابل割り込み

- IRQ端子または周辺機能(要求元)の割り込み要因により発生する割り込みです。
- 割り込み要因は割り込みベクタテーブルに割り当てられます。
- マスクابل割り込みにより、CPU割り込みの発生またはDTC/DMAC/EXDMAC転送を行います。
- マスクابل割り込みを発生させる場合、割り込み機能のレジスタだけでなく、周辺機能(要求元)、CPUのレジスタの設定も必要です。
- マスクابل割り込みには、固有の割り込みの他にグループ割り込みや選択型割り込みも存在します。

マスカブル割り込み

マスカブル割り込みのベクタテーブル

マスカブル割り込みのベクタテーブルについて下記に示します。

割り込み要求発生元	名称	ベクタ番号(IR)	割り込み検出方法	DMAC起動	DTC起動	IER	IPR
・	・	・	・	・	・	・	・
CMT0	CMI0	28	エッジ	○	○	IER03.IEN4	IPR004
・	・	・	・	・	・	・	・
RIIC1	RX11	50	エッジ	○	○	IER06.IEN2	IPR051
	TX11	51	エッジ	○	○	IER06.IEN3	IPR052
・	・	・	・	・	・	・	・
SCIO	RX10	58	エッジ	○	○	IER07.IEN2	IPR058
	TX10	59	エッジ	○	○	IER07.IEN3	IPR059
・	・	・	・	・	・	・	・
ICU (グループ割り込み)	GROUPBL0	110	レベル	○	○	IER0D.IEN6	IPR110
	GROUPBL1	111	レベル	○	○	IER0D.IEN7	IPR111
・	・	・	・	・	・	・	・
PERIB (選択型割り込み)	INTB128	128	エッジ			IER10.IEN0	IPR128
	INTB129	129	エッジ			IER10.IEN1	IPR129
・	・	・	・	・	・	・	・
・	・	・	・	・	・	・	・

単体割り込み、グループ割り込み、選択型割り込みは最大256個のベクタテーブルに振り分けられています。

単体割り込み：割り込み要求に対する対応を早急に行う必要がある割り込みを配置
(無条件トラップ、システム系(BSC,FCU,IRQ,CMT0,LVD,WDT,DTC/DMAC)、通信系(送信データエンプティ,受信完了)など)

グループ割り込み：割り込み要求に対する対応が急を要さない割り込みを配置
(通信系(送信完了、受信エラー)、AD系(コンペア割り込み))

選択型割り込み：複数ユニットを有している機能の割り込みを配置
(タイマ系、A/D系(変換終了割り込み))

各種割り込み要因の検出方法はエッジとレベルに分かれています。レベル割り込みの場合、その割り込み要因をクリアしない限り対応した割り込みフラグ(IRビット)は0クリアされませんのでご注意ください。特にレベル割り込み要因で構成したグループ割り込み(BLx、ALx)はすべての要因が0にならないとグループ割り込みのIRフラグが0になりません。

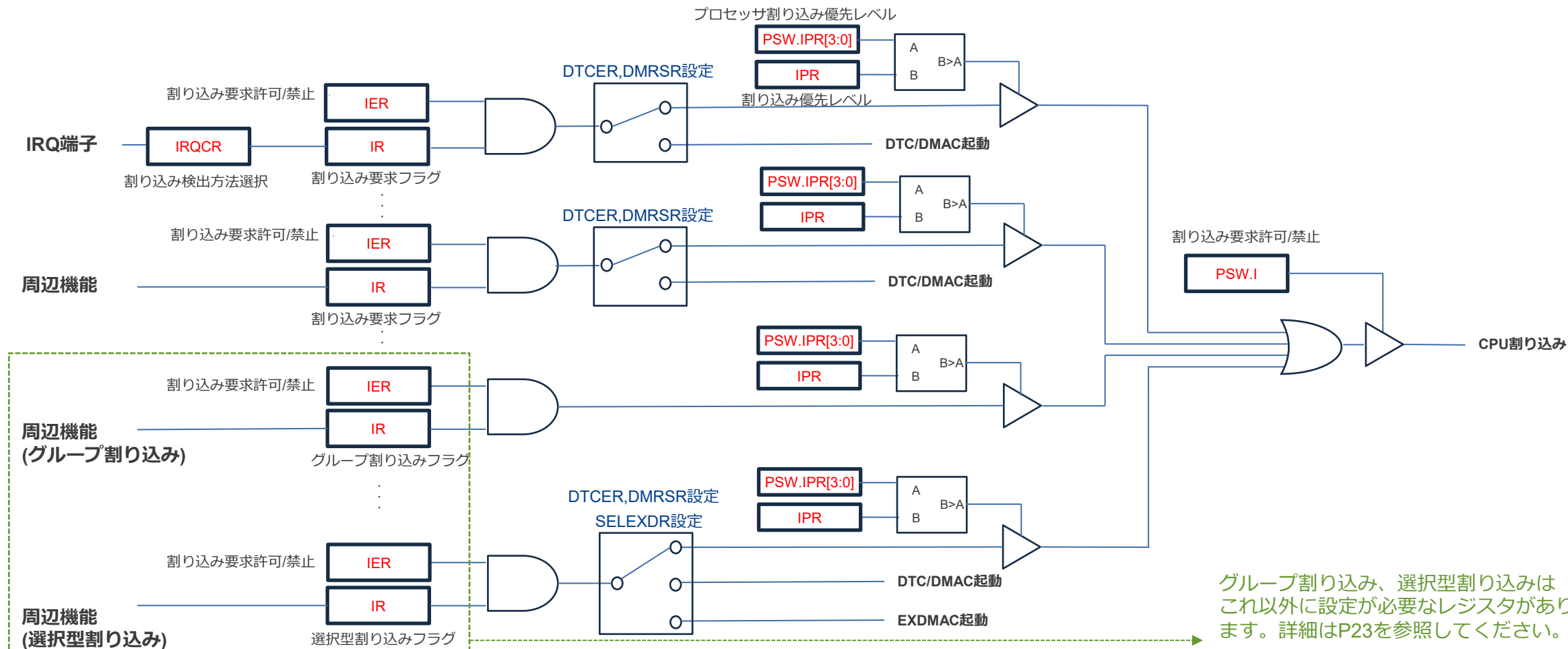
DMAC,DTC起動が可能な割り込みの場合、本項目で“○”と表記されています。逆に起動できない割り込みは“×”と表記されています。なお、DMAC、DTC起動できる割り込み種別はエッジ検出のみでレベル検出は使用できません。

各割り込み要因のCPU割り込み優先順は、IPRレジスタを使用することでベクタテーブルの順番に依存せず0~15の間で自由に設定できます。IPRで設定した優先レベルが同じ場合はベクタ番号の小さい順となります。

マスカブル割り込み

マスカブル割り込み構造

各種割り込み要因(NMI、IRQ、周辺機能)と各要因によるCPU割り込み、DTC/DMAC/EXDMAC転送の流れと必要なレジスタをまとめます。



マスカブル割り込み

グループ割り込み

- 通常、割り込み要因一つにつき固有のベクタテーブル番号が用意されており、それぞれIR、IER、IPRが設けられています。
- グループ割り込みは、複数の周辺モジュールの割り込み要求(最大32本)をグループ化して、まとめて一つの割り込み要求としています。ベクタテーブル番号やIR,IER,IPRもグループ割り込み用として用意されています。(グループ割り込みに割り付けられた個別の要因の割り込みフラグ、割り込み要求許可/禁止設定は専用のレジスタで管理します)
- グループ割り込みはエッジ割り込みをまとめたもの(IE0,BE0)とレベル割り込みをまとめたもの(BL0~2、AL0~1)があります。

割り込み要求発生元	名称	ベクタ番号(IR)	IER	IPR
CMT0	CMIO	28	IER03.IEN4	IPR004
RIIC1	RX11	50	IER06.IEN2	IPR051
	TX11	51	IER06.IEN3	IPR052
SCIO	RX10	58	IER07.IEN2	IPR058
	TX10	59	IER07.IEN3	IPR059
ICU	GROUPBL0	110	IER0D.IEN6	IPR110
	GROUPBL1	111	IER0D.IEN7	IPR111
	GROUPAL0	112	IER0E.IEN0	IPR112
	GROUPAL1	113	IER0E.IEN1	IPR113

ベクタテーブル例

グループ割り込みに割り付けられた個別要因の割り込みフラグ、割り込み要求許可/禁止はベクタテーブルではなく別レジスタで制御

グループ	割り込み要求元	名称	割り込みフラグ	割り込み要求許可/禁止設定
GROUPBL0	SCIO	TEI0	GRPBL0.IS0	GENBL0.EN0
		REI0	GRPBL0.IS1	GENBL0.EN1
SCIO	SCIO	TEI1	GRPBL0.IS2	GENBL0.EN2
		RX11	GRPBL0.IS3	GENBL0.EN3
QSPI	QSPI	QSPSSLI	GRPBL0.IS24	GENBL0.EN24
CAC	CAC	FERRI	GRPBL0.IS26	GENBL0.EN26
		MENDI	GRPBL0.IS27	GENBL0.EN27
DOC	DOC	OVFI	GRPBL0.IS28	GENBL0.EN28
		DOPCI	GRPBL0.IS29	GENBL0.EN29

グループ割り込み例

マスカブル割り込み グループ割り込み構造

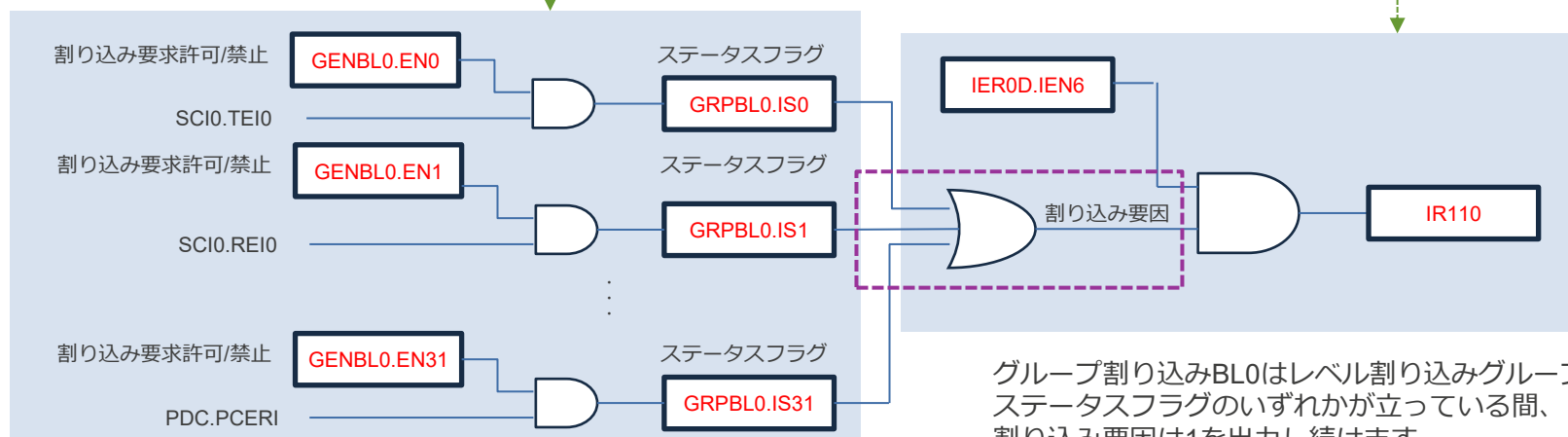
グループ割り込み構造を以下に示します。

グループ	割り込み要求元	名称	割り込みフラグ	割り込み要求許可/禁止設定
GROUPBL0	SCI0	TEI0	GRPBL0.IS0	GENBL0.EN0
		REI0	GRPBL0.IS1	GENBL0.EN1
	SCI1	TEI1	GRPBL0.IS2	GENBL0.EN2
		RXI1	GRPBL0.IS3	GENBL0.EN3
	⋮	⋮	⋮	⋮
	DOC	DOPCI	GRPBL0.IS29	GENBL0.EN29
PDC	PCFEI	GRPBL0.IS30	GENBL0.EN30	
	PCERI	GRPBL0.IS31	GENBL0.EN31	

グループ割り込み(BL0)

割り込み要求発生元	名称	ベクタ番号(IR)	IER
ICU	GROUPBL0	110	IER0D.IEN6
	GROUPBL1	111	IER0D.IEN7
	GROUPAL0	112	IER0E.IEN0
	GROUPAL1	113	IER0E.IEN1

ベクタテーブル



グループ割り込み構造

グループ割り込みBL0はレベル割り込みグループです。ステータスフラグのいずれかが立っている間、割り込み要因は1を出力し続けます。

マスカブル割り込み

グループ割り込み時の処理ルーチンの注意事項

- グループ割り込みの場合、割り込み処理内でグループに割り当てられたすべての割り込みフラグを確認し、フラグの立ったすべての割り込みに対してユーザ処理が必要になります。

グループ	割り込み要求元	名称	割り込みフラグ	割り込み要求許可/禁止設定
GROUPBL0	SCI0	TEI0	GRPBL0.IS0	GENBL0.EN0
		REI0	GRPBL0.IS1	GENBL0.EN1
	SCI1	TEI1	GRPBL0.IS2	GENBL0.EN2
		RXI1	GRPBL0.IS3	GENBL0.EN3
	QSPI	QSPSSLI	GRPBL0.IS24	GENBL0.EN24
	CAC	FERRI	GRPBL0.IS26	GENBL0.EN26
		MENDI	GRPBL0.IS27	GENBL0.EN27
		OVFI	GRPBL0.IS28	GENBL0.EN28
	DOC	DOPCI	GRPBL0.IS29	GENBL0.EN29

グループBL0割り込み要因

■グループ割り込み処理プログラム例

```
if(GRPBL0.IS0==1){  
    TEI0(送信完了)割り込みの処理  
  
}else if(GRPBL0.IS1 ==1){  
    ERI0(受信エラー)割り込みの処理  
  
}else if(GRPBL0.IS2 ==1){  
    TEI1(送信完了)割り込みの処理  
    .  
    .  
    .
```

マスカブル割り込み

選択型割り込み

- ある決められた割り込みベクタ番号に、複数の周辺モジュールの割り込み要因から任意の一つを選択して割り当てます。
- IR、IEN、IPRは割り当てられたベクタ番号に紐づいたものを使用します。
- 選択型割り込みに割り当てられなかった割り込み要因はCPU割り込みとして使用することができません。ただし、各要因毎に割り込みステータスフラグを用意しており、ポーリング用途で使用することができます。

割り込み要求発生元	名称	ベクタ番号(IR)	IEN	IPR
CMT0	CMIO	28	IER03.IEN4	IRP004
RIIC1	RXI1	50	IER06.IEN2	IPR051
	TXI1	51	IER06.IEN3	IPR052
SCIO	RXI0	58	IER07.IEN2	IPR058
	TXI0	59	IER07.IEN3	IPR059
PERIB	INTB128	128	IER10.IEN0	IPR128
	INTB129	129	IER10.IEN1	IPR129
	INTB130	130	IER10.IEN2	IPR130
	INTB131	131	IER10.IEN3	IPR131

ベクタテーブル例

割り込み要求元	名称	割り込みステータスフラグ
CMT2	CMIO	PIBR0.PIR0
CMT3	CMIO	PIBR0.PIR1
TMR0	CMIA0	PIBR0.PIR3
	CMIB0	PIBR0.PIR4
OVI0	OVI0	PIBR0.PIR5
	USBIO	PIBR7.PIR6
S12AD	S12ADI	PIBR8.PIR0
	S12GBDAI	PIBR8.PIR1
	S12GCDAI	PIBR8.PIR2

任意で選択

ポーリングで使用可能

ポーリングで使用可能

ポーリングで使用可能

選択型割り込みB要因一覧例

マスクブル割り込み

グループ割り込み、選択型割り込みレジスタ一覧

グループ割り込みと選択型割り込みレジスタの一覧を示します。

グループ割り込みレジスタ一覧

レジスタ名	詳細	グループ割り込みの種類と各レジスタ						
		IE0	BE0	BL0	BL1	BL2	AL0	AL1
割り込みステータスフラグレジスタ	グループ化された割り込み要因の各割り込み要求ステータスを表示	GRPIE0 *1	GRPBE0 *1	GRPBL0	GRPBL1	GRPBL2	GRPAL0	GRPAL1
割り込み要求許可レジスタ	グループ化された割り込み要因の許可/禁止設定	GENIE0	GENBE0	GENBL0	GENBL1	GENBL2	GENAL0	GENAL1
グループ割り込みクリアレジスタ	割り込みステータスフラグレジスタのクリアを実施	GCRIE0	GCRBE0	-	-	-	-	-

*1:フラグのクリアはグループ割り込みクリアレジスタで行います。

選択型割り込みレジスタ一覧

レジスタ名	詳細	選択型割り込みと各レジスタ	
		選択型割り込みA	選択型割り込みB
選択型割り込み要求レジスタ	選択型割り込みに分類された割り込み要因の各割り込み要求をソフトウェアでポーリングするためのレジスタ	PIBRn	PIARn
選択型割り込み要因選択レジスタ	選択型割り込みに割り当てられたベクタ番号に選択型割り込みに分類された割り込み要因を割り当て	SLIBXRn SLIBRn	SLIAXRn
選択型割り込み要因選択レジスタ書き込み保護レジスタ	選択側割り込み要因選択レジスタへの書き込みを保護するレジスタ	SLIPRCR	

マスクブル割り込み

DTC/DMAC、EXDMAC転送

- マスクブル割り込み要因(エッジ要因)によりDTC、DMACおよびEXDMACを起動することができます。
- DTC、DMACはIPRの影響を受けません。EXDMACはIPR、IERの影響を受けません。
- DTC、DMACは転送回数に応じて転送要因となった割り込みによるCPU割り込みを発生させることが可能です。この際のCPU割り込み要因はIPRの影響を受けます。
- DTC、DMAC、EXDMACは転送終了などに応じて個別の割り込みを発生させることができます。こちらの詳細はユーザーズマニュアルハードウェア編のDTC、DMAC,EXDMACの章を参照してください。

マスクブル割り込み

IRビットのクリアタイミング

割り込み		クリアタイミング
エッジ検出	CPU割り込み	<ul style="list-style-type: none">・割り込み要求を受け付けたとき・IRフラグに0を書いたとき
	DMAC/DTC転送	<ul style="list-style-type: none">・DTC,DMAC転送開始時*CPUでのIRフラグクリアは禁止
レベル検出	CPU割り込み	<ul style="list-style-type: none">・グループ割り込み以外 周辺モジュールの割り込み要求出力をクリア・グループ割り込み グループ割り込み要求許可レジスタのEnjビットが0、またはグループ割り込み要求レジスタIsjフラグがすべて0

マスクブル割り込み

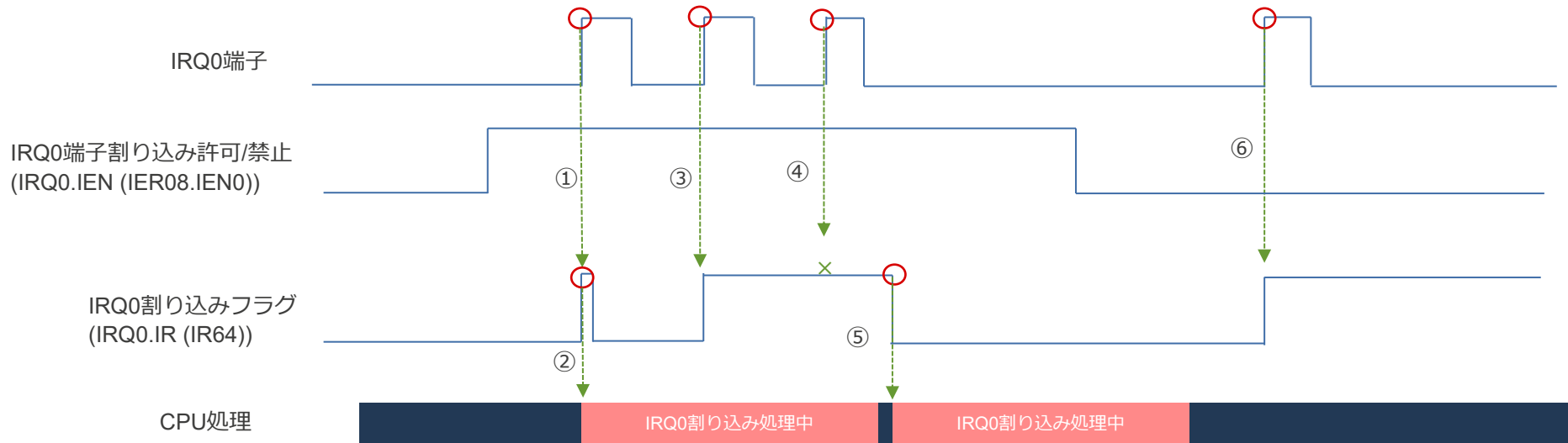
高速割り込み

- 高速割り込みは、CPU の割り込み応答を高速に実行できる割り込みで、割り込み要因のうちの 1 つだけを割り当てることができます。
- 高速割り込みの割り込み優先レベルは、IPRn.IPR[3:0] ビット(n = 割り込みベクタ番号) の設定にかかわらず、15 (最高) です。また、他のレベル15 の割り込み要因よりも優先的に受け付けられます。ただし、PSW.IPL[3:0] ビットの値が“1111b” (優先レベル15) の場合は、高速割り込みも受け付けられません。
- 割り込み要因を高速割り込みに割り当てするには、FIR.FVCT[7:0] ビットにその要因のベクタ番号を設定し、FIR.FIEN ビットを“1” (高速割り込みを許可) にしてください。

割り込み処理例

(IRQ端子：エッジ割り込み)

IRQ端子によるエッジ割り込みの動作例を示します。



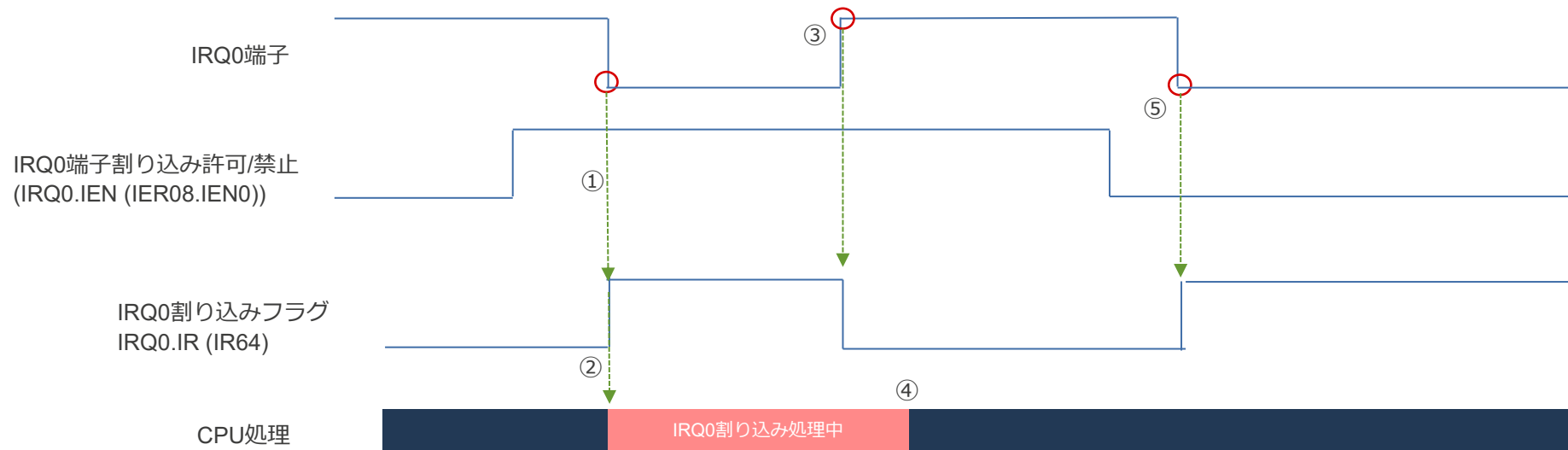
IRQ割り込み例：立ち上がりエッジ割り込み

- ① IRQ端子の立ち上がりエッジを検知。IRQ0.IRがセットされる。
- ② IRQ0によるCPU割り込み処理開始。IRQ0.IRビットは自動的にクリアされる。
- ③ 割り込み処理中、再度IRQ端子の立ち上がりエッジを検知。IRQ0.IRがセット(保留)される。
- ④ 再度IRQ端子の立ち上がりエッジを検知。しかし、IRQ0.IRビットはすでに保留中のため、この要求は破棄される。
- ⑤ IRQ0割り込み処理終了後、保留したIRQ0.IRビットにより再度IRQ0割り込みが発生。
- ⑥ IRQ端子の立ち上がりを検知。IRQ0.IRはセットされるがIRQ0.IENビットが0のためCPU割り込みは発生しない。

割り込み処理例

(IRQ端子：レベル割り込み)

IRQ端子によるレベル割り込みの動作例を示します。



IRQ割り込み例：Lowレベル割り込み

- ① IRQ端子のLowを検知。IRQ0.IRがセットされる。
- ② IRQ0によるCPU割り込み処理が開始 (エッジ割り込みと異なり、IRQ0.IRビットはクリアされない)
- ③ IRQ端子がHighに変化。これに伴いIRQ0.IRもクリアされる。
- ④ IRQ0.64ビットのクリアを確認後、IRQ0割り込み処理を終了。
- ⑤ IRQ0端子のLowを検知。IRQ0.IRビットがセットされるがIRQ0.IENビットが0のため割り込み処理は発生しない。

割り込み処理例

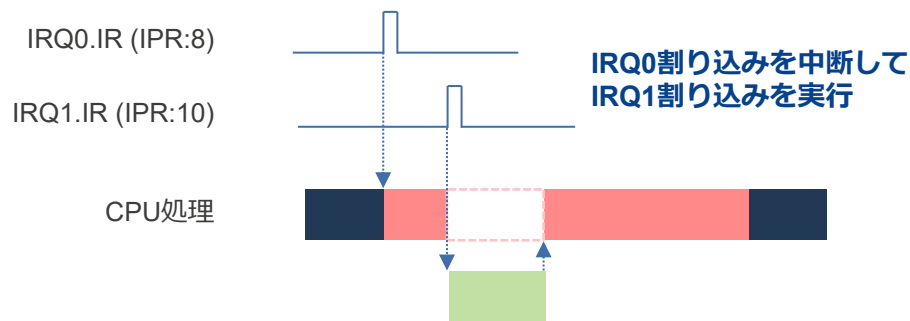
(IRQ端子：多重割り込み例、同時割り込み発生例)

割り込み処理の先頭でPSW.Iビットを1にすると多重割り込みを許可できます。多重割り込みを許可にすると、割り込み処理中の割り込み優先レベルより高い優先レベルの割り込み要求が発生した場合、この割り込みを受け付けます。

多重割り込みを許可した場合のIRQ端子割り込み動作例を示します。

IPRで設定した値が高い割り込みほど、優先レベルは高くなります。IPRの値が同じ場合はベクタ番号の小さい順になります。

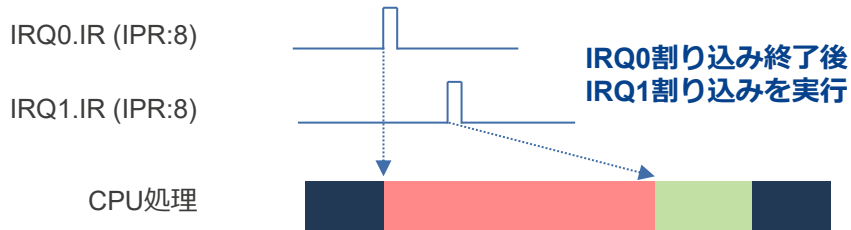
① 割り込み処理中、より高い優先レベルの割り込みが発生した場合



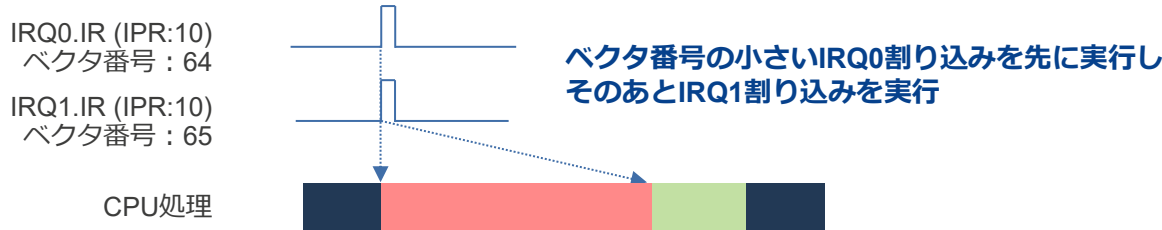
③ 同時に優先レベルの異なる割り込みが発生した場合



② 割り込み処理中、同じ優先レベルの割り込みが発生した場合



④ 同時に同じ優先レベルの割り込みが発生した場合



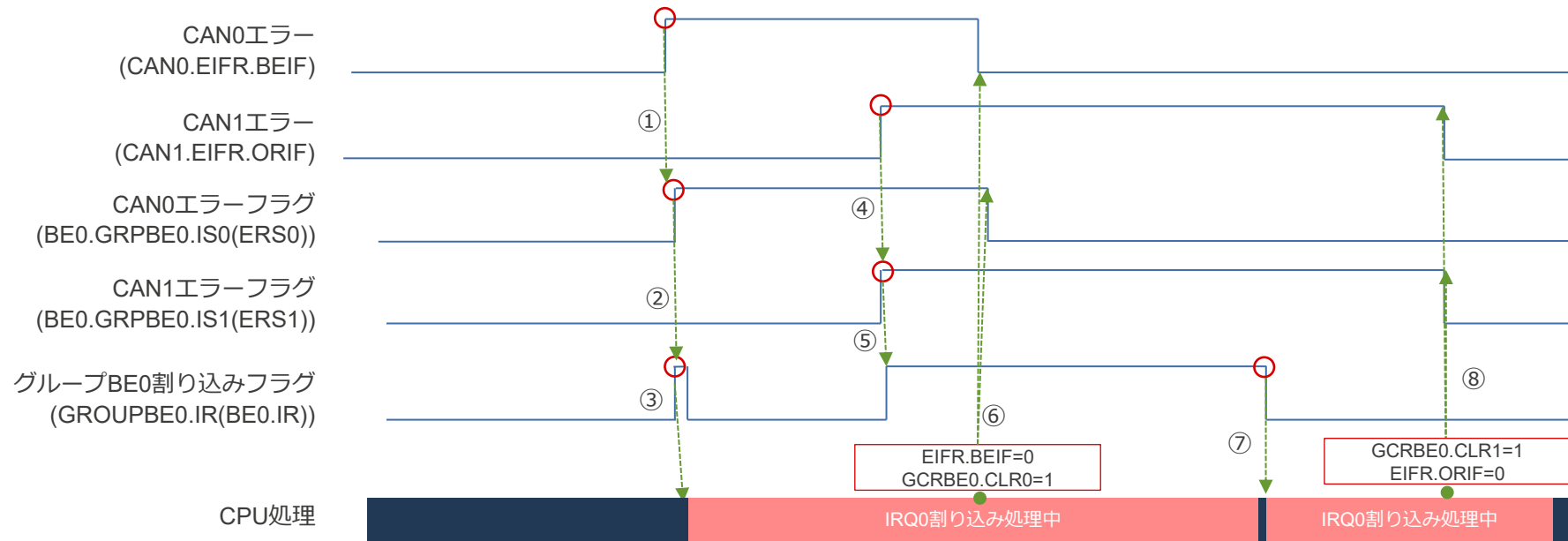
■ 通常処理 ■ 割り込み処理(IRQ0) ■ 割り込み処理(IRQ1)

割り込み発生例

グループ割り込み(エッジ割り込み)

● : Data書き込みタイミング

- グループ割り込み(エッジ割り込み)の動作例を以下に示します。



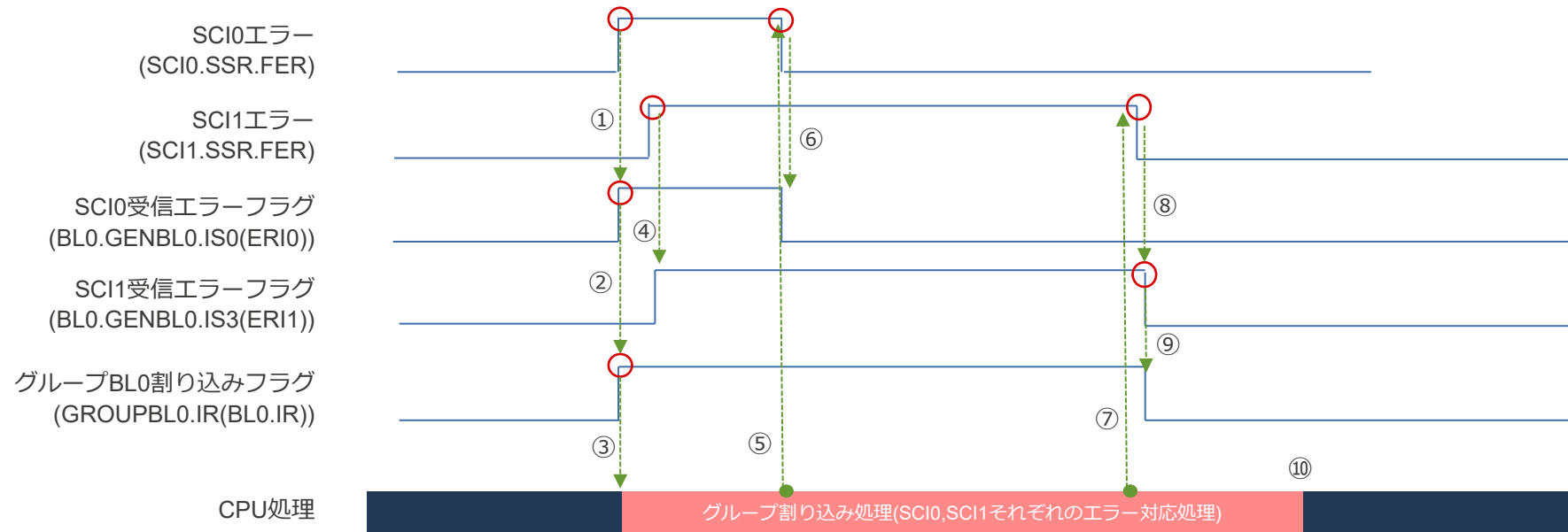
- ① CAN0エラーを検知。BE0.GRPBE0.IS0がセットされる。
- ② ①によりGROUP.IRがセットされる。
- ③ グループ割り込み処理開始。GROUP.IRビットは自動的にクリアされる。
- ④ グループ割り込み処理中にCAN1エラーも発生。BE0.GRPBE0.IS1がセットされる。
- ⑤ ④によりGROUP.IRがセット(保留)される。
- ⑥ グループ割り込み処理内で、CAN0.EIFR.BEIF並びにBE0.GROUPBE0.IS0がクリアされる。
- ⑦ グループ割り込み処理終了後、保留されたIRビットにより再度グループ割り込み発生。
- ⑧ グループ割り込み処理内でCAN1.EIFR.ORIF並びにBE0.GROUPBE0.IS1がクリアされる。

割り込み発生例

グループ割り込み(レベル割り込み)

● : Data書き込みタイミング

- グループ割り込み(レベル割り込み)の動作例を以下に示します。



フィルタ処理

- NMI入力端子およびIRQ入力端子はフィルタ処理によるノイズ除去が可能です。
- 3回連続でレベルが一致する入力信号のみを通過させます。
- ノイズフィルタ処理は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、スヌーズモード中は使用できません。

	デジタルフィルタ 有効/無効設定	デジタルフィルタ サンプリングクロック設定
NMI入力端子	NMIFLTE	NMIFLTC
IRQ入力端子	IRQFLTE _n	IRQFLTC _n

低消費電力モードからの復帰

- 各種低消費電力状態からの復帰に割り込みが使用できます。詳細はハードウェアマニュアルの低消費電力低減機能を参照してください。
- ノンマスカブル割り込み、マスカブル割り込みにより、低消費電力状態から復帰する場合、割り込み機能の各レジスタの設定において注意事項があります。詳細はハードウェアマニュアルの割り込みコントローラ、低消費電力状態からの復帰の章を参照してください。

割り込み処理ルーチンと注意事項

- 割り込み処理のルーチンと注意事項を示します。

高速割り込みと通常割り込みではPC、PSWの退避先が異なります。高速割り込みの退避先は専用レジスタになります。割り込み処理サイクルは高速割り込みの方が早くなります。

PSW.PMビットが1だった場合、割り込み処理開始時に**スーパーバイザモードに移行**します。割り込み処理終了時に**ユーザモードに移行**します。

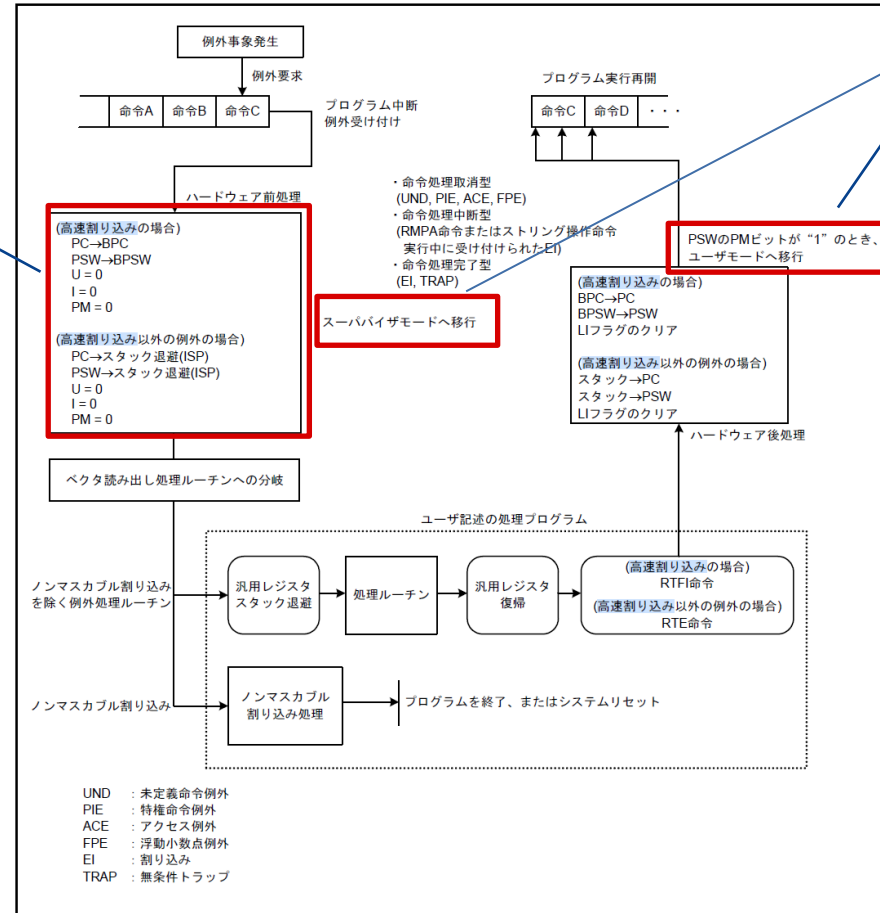


図 13.2 例外の処理手順の概要

割り込みコントローラのアプリケーションノート

割り込みコントローラの詳細な使用方法は下記をご参照ください

- RXファミリ 多重割り込みの使い方 [R01AN1954](#)
- RXファミリ 例外ベクタテーブルと選択型割り込みの使い方 [R01AN2178](#)

外部バスコントローラ

本章はRX72Mを参考に作成しておりますが、全製品の参考資料としてお使いいただけます。

バスコントローラ構成

■ RXファミリのバスコントローラ構成についてRX72Mを例として以下に示します。

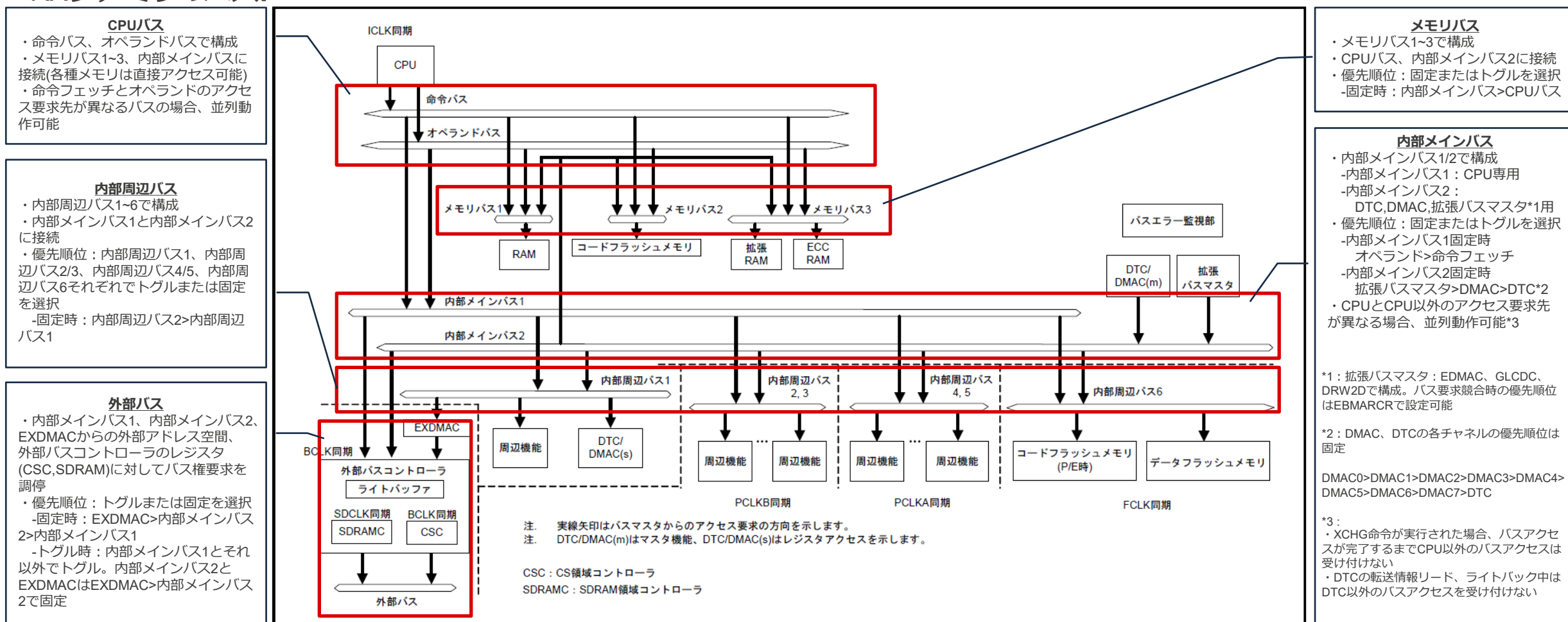


図 16.1 バスの構成図

内部周辺バスのライトバッファ、並列動作仕様について

- RXの内部周辺バスはライトバッファ機能を持っており、ライトアクセスの場合、動作の終了(実レジスタへのライト)を待たずに次の処理を実施できます。また、それぞれのバスマスタが異なるアクセス先にアクセスする場合、並列に動作できます。各仕様について以下にまとめます。
- ライトバッファの動作仕様
 - 連続するアクセス(要求元(バスマスタ)は問わず、1回目と2回目のアクセス先が同じ)：レジスタライトを待たず連続実行が可能
 - 同じ要求元による連続アクセス(1回目と2回目どちらか一方だけアクセス先が内部メモリ)
 - ②-1：要求元がCPUの場合：並列動作が可能
 - ②-2：要求元がDTC、DMAC、拡張バスマスタのいずれかの場合：レジスタライトを待たず連続実行が可能
 - 同じ要求元による連続アクセス(1回目と2回目のアクセス先がどちらも周辺バスの場合(ただし異なる番号の周辺バス))：レジスタライトまで待たされる
- 並列動作の動作仕様
 - 異なる要求元による連続アクセス(1回目と2回目のアクセス先が異なる)：並列動作が可能

No	一回目アクセス		二回目アクセス		結果
	要求元	アクセス先	要求元	アクセス先	
1	CPU	内部周辺バス1	CPU	内部周辺バス1	① 連続実行(レジスタライトは待たない)
2	DTC	メモリバス	DMAC	メモリバス	① 連続実行(レジスタライトは待たない)
3	CPU	内部周辺バス1	CPU	メモリバス	②-1 並列動作(パイプラインによる)
4	DTC	内部周辺バス1	DTC	メモリバス	②-2 連続実行 (内部周辺バス2のバスマスタは並列動作できないため)
5	CPU	内部周辺バス1	CPU	内部周辺バス2	③ レジスタライト終了まで待たされる
6	DMAC	内部周辺バス1	拡張バスマスタ	内部周辺バス2	③ レジスタライト終了まで待たされる
7	CPU	内部周辺バス1	拡張バスマスタ	メモリバス	④ 並列動作
8	DTC	内部周辺バス2	CPU	内部周辺バス1	④ 並列動作

内部周辺バスのライトバッファ、並列動作例

①連続実行(レジスタライトは待たずに次のバスアクセスを実行)

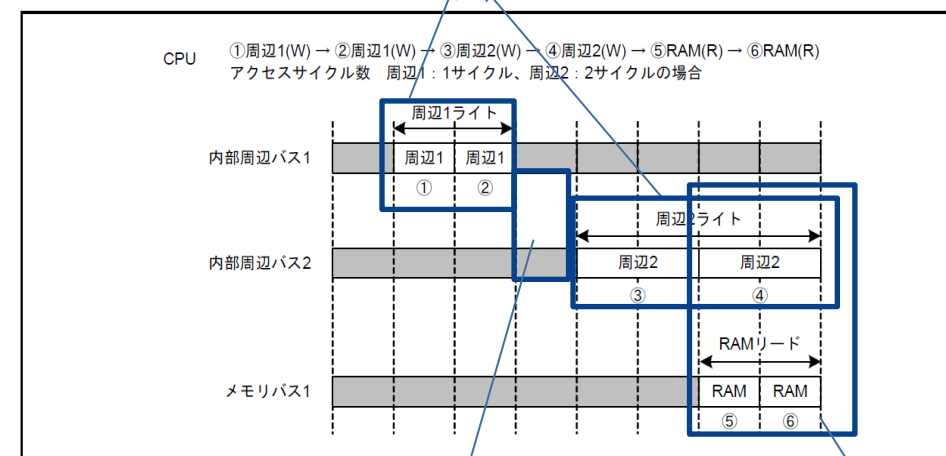


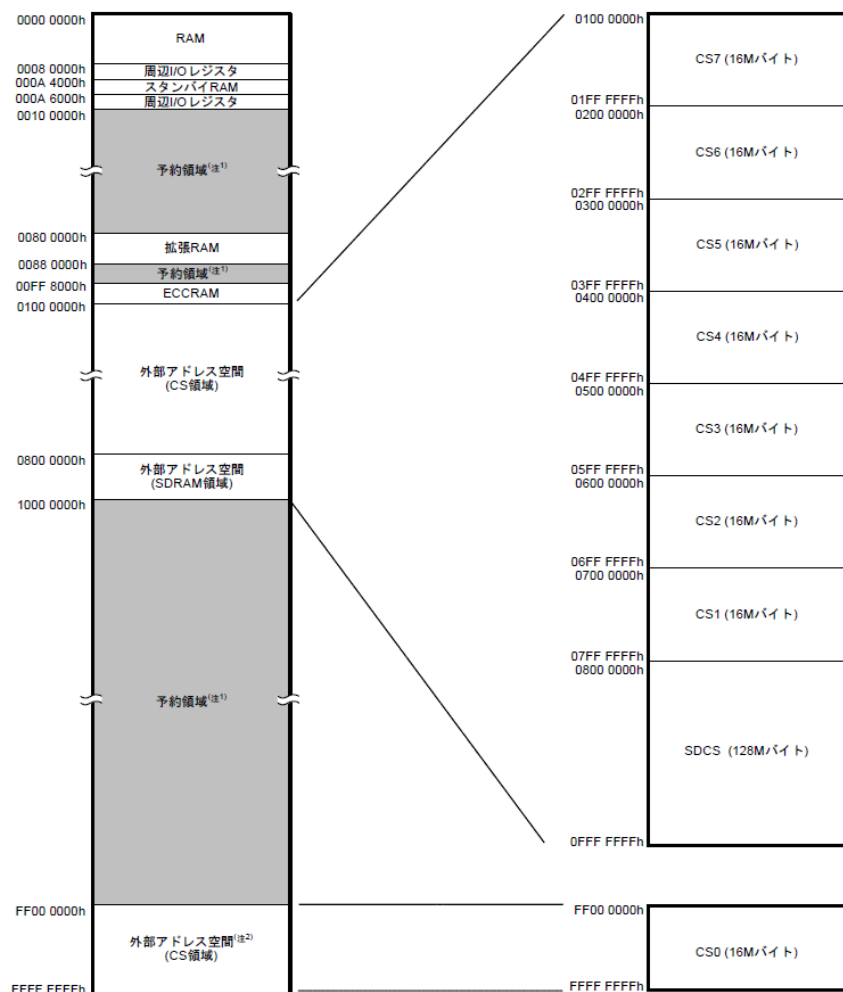
図 16.3 ライトバッファ機能

③レジスタライト待ち

②-1並列動作

内部周辺バスのライトバッファ制御

外部バス



外部アドレス空間とCS空間(内蔵ROM無効拡張モード)

外部バス空間は、複数のCS空間とSDRAM専用空間に分割されています。相手デバイスの仕様に合わせて、アクセス方法やウェイト設定を行い、出力波形を制御することでバス接続を行うことが可能になります。

■ CS空間

それぞれの空間において、個別に以下設定が可能です。

- ・アクセス方法の設定(セパレートバス or アドレス/データマルチプレクスバス)
- ・バス幅設定
- ・リカバリサイクル、サイクルウェイト、外部/内部ウェイト制御
- ・ライトアクセスモード設定(1ライトストローク or バイトストローク)
- ・エンディアン設定
- ・ページリード/ライト制御

CS0空間は内蔵ROM領域と排他になります。

■ SDRAM空間

SDRAM I/Fと接続できます。一部製品で対応しています。

外部バス 各モードと使用端子一覧

- 外部バスの各モードと使用可能な端子一覧を以下に示します。

*A0/D0~A15/D15はアドレス/データマルチプレクス専用端子です。
セパレートバスでアドレスとして使用することはできません。

端子	CS空間				SDRAM空間	非アクセス時の状態
	セパレート		アドレス/データマルチプレクス			
	バイトストロープ	1ライトストロープ	バイトストロープ	1ライトストロープ		
A23~A0	✓	✓			✓	
D31~D0	✓	✓			✓	ハイインピーダンス
A0/D0~A15/D15*			✓	✓		
BC0#~BC3#		✓		✓		High
CS0#~CS7#	✓	✓	✓	✓		High
RD#	✓	✓	✓	✓		High
WR0#/WR#	✓(WR0#として使用)	✓(WR#として使用)	✓(WR0#として使用)	✓(WR#として使用)		High
WR1#~WR3#	✓		✓			High
ALE			✓	✓		Low
WAIT#	✓	✓	✓	✓		
BCLK	✓	✓	✓	✓		
SDCLK					✓	
CKE					✓	
SDCS#					✓	High
RAS#					✓	High
CAS#					✓	High
WE#					✓	High
DQM0~DQM3					✓	

外部バス 各モードと設定レジスタ

- 外部バスの各モードと設定が必要なレジスタの一覧を示します。

レジスタ名および設定内容		CS空間	SDRAM空間	バスエラー関連	プライオリティ制御関連
CSnCR	バス動作許可/禁止、バス幅、エンディアン設定、アクセス方法(セパレート or マルチプレクス)	✓			
CSnREC	リカバリサイクル数(リード、ライト)	✓			
CSRECEEN	リカバリサイクル挿入許可/禁止	✓			
CSnMOD	ライトアクセスモード設定、外部ウェイト許可/禁止、ページリード/ライトアクセス許可/禁止	✓			
CSnWCR1,2	ウェイト設定(ノーマルライト/リード、ページライト/リード、CS延長サイクル(リード/ライト)、ライトデータ出力延長/出力ウェイト、アドレスサイクル、アサート(RD,WD,CS))	✓			
SDCCR	SDRAM動作許可/禁止		✓		
SDCMOD	SDRAM空間エンディアン設定		✓		
SDAMOD	SDRAM連続アクセス許可/禁止		✓		
SDSELF	セルフリフレッシュ有効/無効		✓		
SDRFCR	オートリフレッシュ関連		✓		
SDRFEN	オートリフレッシュ有効/無効		✓		
SDICR	SDRAM初期化シーケンス		✓		
SDIR	SDRAM初期化オートリフレッシュ間隔、回数、初期化プリチャージサイクル数		✓		
SDADR	SDRAMアドレスマルチプレクスシフト量設定		✓		
SDTR	SDRAMカラムレイテンシ、ライトリカバリ、ロウプリチャージ、ロウアクティブ期間		✓		
SDMOD	SDRAMモードレジスタ書き込み		✓		
SDSR	SDRAMアクセスステータス		✓		
BERCLR	バスエラーステータスクリア			✓	
BEREN	バスエラー監視許可/禁止			✓	
BERSR1,2	バスエラーステータス			✓	
BUSPRI	バスプライオリティ制御				✓
EBMAPCR	拡張バスマスタプライオリティ制御				✓

外部バス 各モードと設定レジスタ(CS空間詳細)

- CS空間における外部バスの各モードと設定が必要なレジスタ詳細を以下に示します。

W : Write, R : Read

レジスタ名および設定内容				セパレート	アドレス/データ マルチプレクス	ページ アクセス	リカバリ
説明	レジスタ名	ビット名	詳細				
動作モード 関連の設定	CSnCR	EXENB	外部バスの動作禁止/許可設定	✓	✓	-	-
		BSIZE[1:0]	外部バス幅選択	✓	✓	-	-
		EMODE	CS領域のエンディアン設定	✓	✓	-	-
		MPXEN	アクセス方法(セパレートバス、アドレス/データマルチプレクス)設定	✓	✓	-	-
	CSnMOD	WRMOD	ライトアクセスモード設定(バイトストロープ/1ライトストロープ)	✓	✓	-	-
		EWENB	外部ウェイトの許可/禁止設定	✓	✓	-	-
		PRENB	ページリードアクセス許可/禁止設定	-	-	✓(R)	-
		PWENB	ページライトアクセス許可/禁止設定	-	-	✓(W)	-
	PRMOD	ページリードアクセスモード選択(ノーマルアクセス互換モード/外部データリード連続アサート)	-	-	✓(R)	-	
リカバリ サイクル 関連の設定	CSnREC	RRCV[3:0]	リードリカバリ数の設定	-	-	-	✓(R)
		WRCV[3:0]	ライトリカバリ数の設定	-	-	-	✓(W)
	CSRECEEN	RCVEN0~7	セパレートバス用リカバリサイクルの許可/禁止	-	-	-	✓(R)
		RCVENM0~7	マルチプレクスバス用リカバリサイクルの許可/禁止	-	-	-	✓(W)
ウェイト 制御	CSnWCR1	CSPWAIT[4:0]	ページライトサイクルウェイト数設定	-	-	✓(W)	-
		CSPRWAIT[4:0]	ページリードサイクルウェイト数設定	-	-	✓(R)	-
		CSWAIT[4:0]	ノーマルライトサイクルウェイト数設定	✓(W)	✓(W)	-	-
		CSRWAIT[4:0]	ノーマルリードサイクルウェイト数設定	✓(R)	✓(R)	-	-
	CSnWCR2	CSROFF[2:0]	リード時CS延長サイクル数設定	✓(R)	✓(R)	-	-
		CSWOFF[2:0]	ライト時CS延長サイクル数設定	✓(W)	✓(W)	-	-
		WDOFF[2:0]	ライトデータ出力延長サイクル数設定	✓(W)	✓(W)	-	-
		AWAIT[1:0]	アドレスサイクルウェイト数設定		✓(W)	-	-
		RDON[2:0]	RDアサートウェイト数設定	✓(R)	✓(R)	-	-
		WRON[2:0]	WRアサートウェイト数設定	✓(W)	✓(W)	-	-
		WDON[2:0]	ライトデータ出力ウェイト数設定	✓(W)	✓(W)	-	-
		CSON[2:0]	CSアサートウェイト数設定	✓(R/W)	✓(R/W)	-	-

外部バスインタフェース端子設定方法について

- 外部バスインタフェースの端子設定はマルチピンファンクションコントローラ章に記載されています。
下記表のとおりMPCのレジスタを設定した後SYSCR0.EXBE=1にすることで外部バス端子として有効となります。

表 23.46 外部バスインタフェース設定方法 (1/4)

ポート	出力信号名	MPCのレジスタの設定	
		224ピン、176ピン	144ピン
P10	ALE	PFBCR1.ALEOE = 1, PFBCR1.ALES = 1	
P12	WR3#/BC3#	PFBCR0.WR32BC32E = 1	(非対応)
P13	WR2#/BC2#	PFBCR0.WR32BC32E = 1	(非対応)
P24	CS4#	PFCSE.CS4E = 1, PFCSS1.CS4S[1:0] = 10/11	
P25	CS5#	PFCSE.CS5E = 1, PFCSS1.CS5S[1:0] = 10/11	
P26	CS6#	PFCSE.CS6E = 1, PFCSS1.CS6S[1:0] = 10/11	
P27	CS7#	PFCSE.CS7E = 1, PFCSS1.CS7S[1:0] = 10/11	
P50	WR0#/WR#	—	
P51	WR1#/BC1#	PFBCR0.WR1BC1E = 1	
	WAIT#	PFBCR1.WAITS[1:0] = 11, PFBCR3.WAITS2 = 0	
P52	RD#	—	
P53	BCLK	—	
P54	ALE	PFBCR1.ALEOE = 1, PFBCR1.ALES = 0	
	D1[A1/D1]	PFBCR2.D1S[1:0] = 10	
P55	WAIT#	PFBCR1.WAITS[1:0] = 01, PFBCR3.WAITS2 = 0	
	D0[A0/D0]	PFBCR2.D0S[1:0] = 10	
P60	CS0#	PFCSE.CS0E = 1, PFCSS0.CS0S = 0	
P61	CS1#	PFCSE.CS1E = 1, PFCSS0.CS1S[1:0] = 00	
	SDCS#	PFBCR1.MDSDE = 1	(ピンなし)
	D0[A0/D0]	PFBCR2.D0S[1:0] = 11	
PA0	A0	PFBCR0.ADRLE = 1, CsnMOD.WRMOD = 0	
	BC0#	PFBCR0.ADRLE = 1, CsnMOD.WRMOD = 1	
	DQM2	PFBCR0.ADRLE = 1, SDCCR.EXENB = 1, SDCCR.BSIZE[1:0] = 01	

ここに記載のレジスタの通り設定してください。
他の機能端子として使用したい場合は、この設定と異なる設定をしてください。

レジスタが複数記載している場合すべてのレジスタを設定する必要があります。
いずれかが設定されていない場合、外部バス端子として有効になりませんが、他の機能端子として使用できます。

<Smart configuratorの設定>

ALEとD1：禁止
CS1#とSDCS#：禁止
SDCS#とD0：禁止
A0とBC0とDQM2：許可

“—”の記載があるものは、SYSCR0.EXBEビットを1にするだけでバス端子として有効になります。

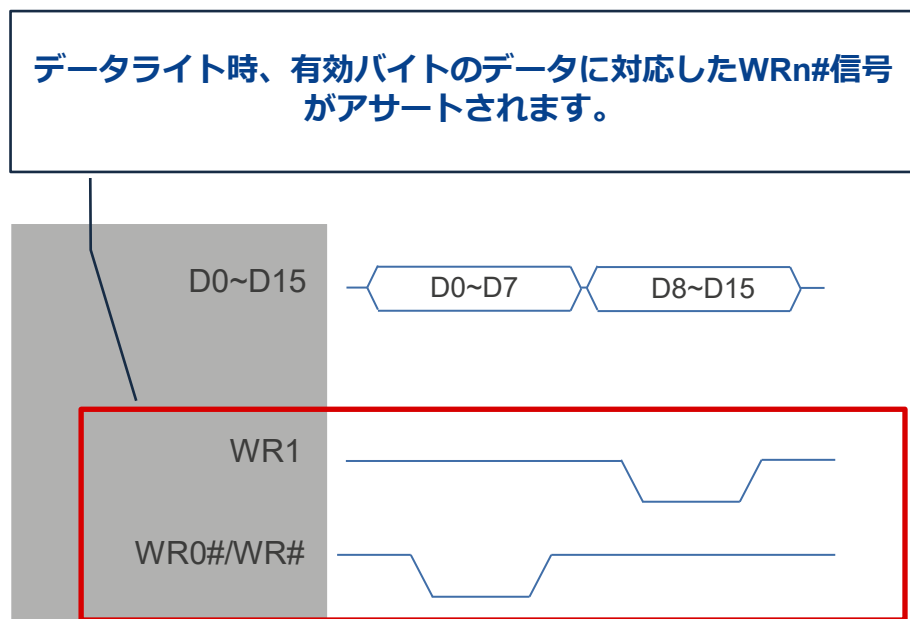
複数端子を選択した場合、動作に影響のケースにおいてはいずれかの端子のみ選択するようにしてください。
P61の場合、CS1#とSDCS#を同時にONするとSDRAM空間にアクセス時、CS1空間もアクティブになってしまう、誤動作してしまうため、同時設定は禁止です。

複数の異なるデバイスを接続しても、他の端子機能などにより排他制御ができる場合は、複数選択可能です。
PA0の場合、SDRAMアクセスでDQM2がアクティブになった場合BC0#端子もアクティブになりますが、該当するCS#がアクティブにならない限り誤動作は起こさないため同時設定可能です。

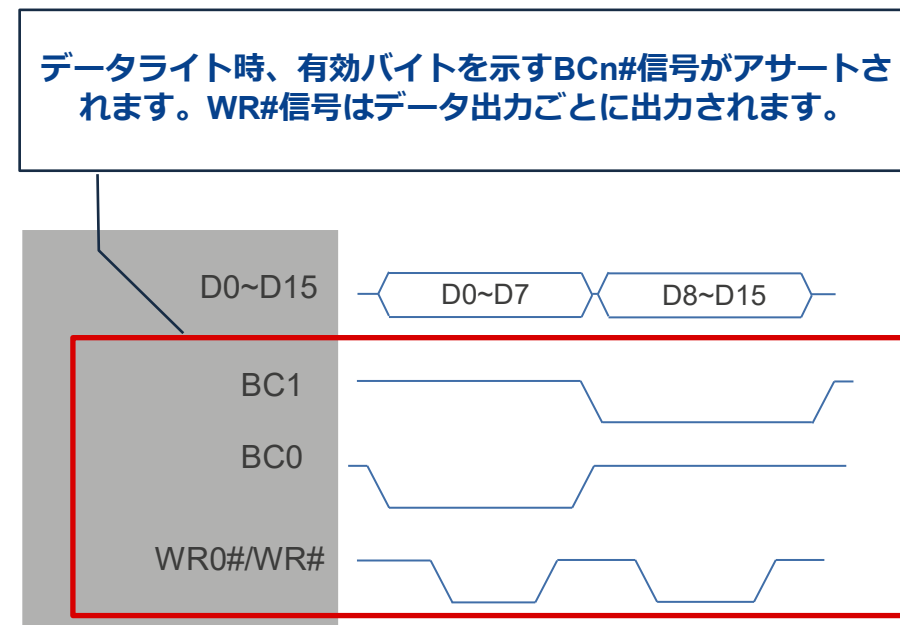
外部バス

CS空間(ライトアクセスモード)

- ライトアクセスモードはバイトストローブモードと1ライトストローブモードをサポートしています。それぞれの各信号出力例を以下に示します。



バイトストローブ ライトアクセス信号出力例
(16ビットバス空間 4n+1アドレスへの16ビットアクセス)



1ライトストローブ ライトアクセス信号出力例
(16ビットバス空間 4n+1アドレスへの16ビットアクセス)

エンディアン

- 外部バスはCS空間ごとにエンディアンを設定できます。また、設定したエンディアンおよびアクセスアドレスにより1度の転送要求で複数のアクセスが発生します。バス幅以上のデータサイズアクセスを行う場合、2回目以降のアクセスでページアクセスが発生することがあります。アクセス先の空間のバス幅に合わせたアクセスアドレス指定をすることで、アクセス回数の増加を抑えることができます。

設定したエンディアンによりデータのアクセス順が変わります。

データバス幅を超えるアクセスサイズのアクセスをするとページアクセスが発生することがあります。対象アクセスには(p)を表示しています。

データサイズ	アクセス番地	アクセス回数	バスサイクル	データ量	アドレス	WR1#/BC1# WR0#/BC0#			
						RD#			
						D15	D8 D7	D0	
8bit	4n	1回	1回目	8bit	4n	[7 0]			
	4n+1	1回	1回目	8bit	4n	[7 0]			
	4n+2	1回	1回目	8bit	4n+2	[7 0]			
	4n+3	1回	1回目	8bit	4n+2	[7 0]			
16bit	4n	1回	1回目	16bit	4n	[15 8]	[7 0]	[0]	
	4n+1	2回	1回目	8bit	4n	[7 0]			
			2回目	8bit	4n+2	[15 8]			
	4n+2	1回	1回目	16bit	4n+2	[15 8 7 0]			
	4n+3	2回	1回目	8bit	4n+2	[7 0]			
			2回目	8bit	4n+4	[15 8]			
32bit	4n	2回	1回目	16bit	4n	[15 8 7 0]			
			2回目	16bit	4n+2 (p)	[31 24 23 16]			
	4n+1	3回	1回目	8bit	4n	[7 0]			
			2回目	16bit	4n+2	[23 16 15 8]			
			3回目	8bit	4n+4	[31 24]			
	4n+2	2回	1回目	16bit	4n+2	[15 8 7 0]			
			2回目	16bit	4n+4	[31 24 23 16]			
	4n+3	3回	1回目	8bit	4n+2	[7 0]			
			2回目	16bit	4n+4	[23 16 15 8]			
			3回目	8bit	4n+6	[31 24]			
64bit	4n	4回	1回目	16bit	4n	[15 8 7 0]			
			2回目	16bit	4n+2 (p)	[31 24 23 16]			
			3回目	16bit	4n+4	[47 40 39 32]			
			4回目	16bit	4n+6 (p)	[63 56 55 48]			

CSnMOD.PRENB、PWENBビットでページアクセスを許可した場合、(p)で表示されたアクセスがページアクセスの対象となります。

図 16.8 16ビットバス空間のデータアライメント (リトルエンディアン)

データサイズ	アクセス番地	アクセス回数	バスサイクル	データ量	アドレス	WR1#/BC1# WR0#/BC0#			
						RD#			
						D15	D8 D7	D0	
8bit	4n	1回	1回目	8bit	4n	[7 0]			
	4n+1	1回	1回目	8bit	4n	[7 0]			
	4n+2	1回	1回目	8bit	4n+2	[7 0]			
	4n+3	1回	1回目	8bit	4n+2	[7 0]			
16bit	4n	1回	1回目	16bit	4n	[15 8 7 0]			
	4n+1	2回	1回目	8bit	4n	[15 8]			
			2回目	8bit	4n+2	[7 0]			
	4n+2	1回	1回目	16bit	4n+2	[15 8 7 0]			
	4n+3	2回	1回目	8bit	4n+2	[15 8]			
			2回目	8bit	4n+4	[7 0]			
32bit	4n	2回	1回目	16bit	4n	[31 24 23 16]			
			2回目	16bit	4n+2 (p)	[15 8 7 0]			
	4n+1	3回	1回目	8bit	4n	[31 24]			
			2回目	16bit	4n+2	[23 16 15 8]			
			3回目	8bit	4n+4	[7 0]			
	4n+2	2回	1回目	16bit	4n+2	[31 24 23 16]			
			2回目	16bit	4n+4	[15 8 7 0]			
	4n+3	3回	1回目	8bit	4n+2	[31 24]			
			2回目	16bit	4n+4	[23 16 15 8]			
			3回目	8bit	4n+6	[7 0]			
64bit	4n	4回	1回目	16bit	4n	[63 56 55 48]			
			2回目	16bit	4n+2 (p)	[47 40 39 32]			
			3回目	16bit	4n+4	[31 24 23 16]			
			4回目	16bit	4n+6 (p)	[15 8 7 0]			

CSnMOD.PRENB、PWENBビットでページアクセスを許可した場合、(p)で表示されたアクセスがページアクセスの対象となります。

図 16.9 16ビットバス空間のデータアライメント (ビッグエンディアン)

ノーマルリードタイミング(各種ウェイト設定)

- ノーマルリードタイミング例を以下に示します。接続デバイスに合わせてウェイトを挿入し、出力波形を制御してください。

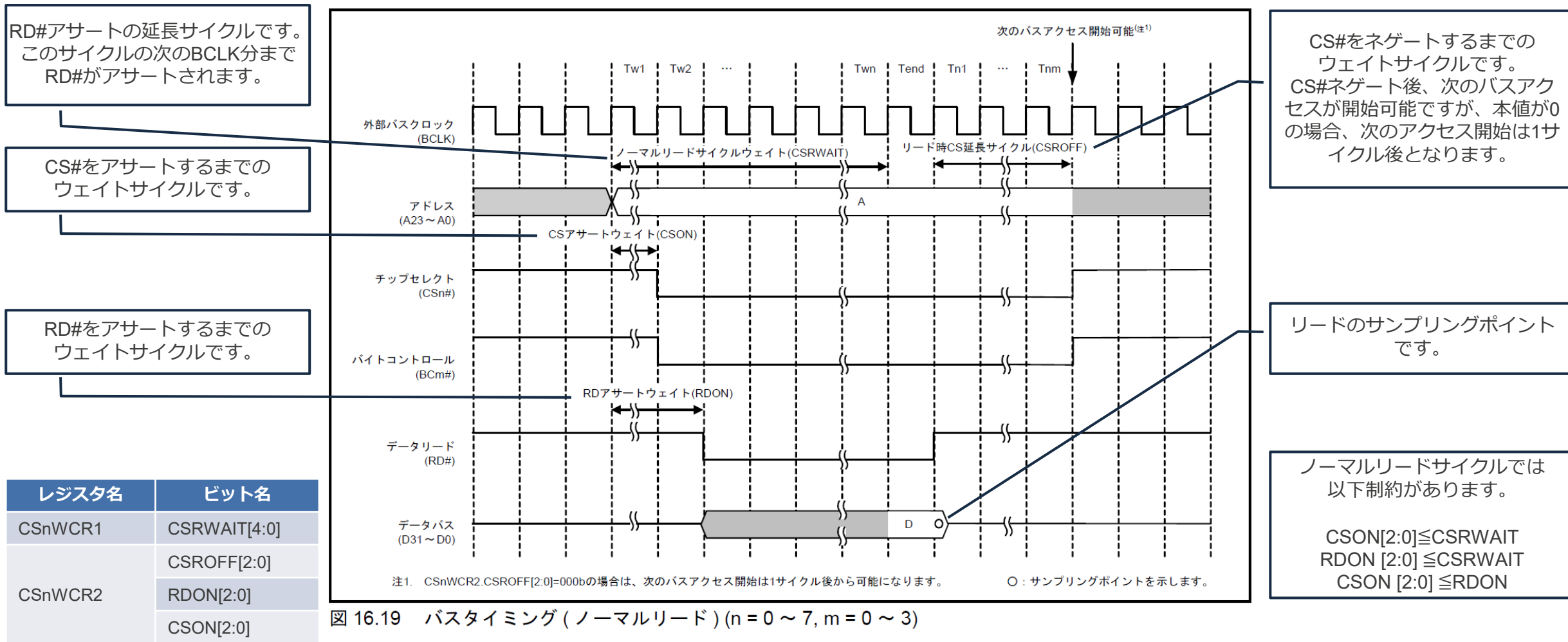


図 16.19 バスタイミング (ノーマルリード) (n = 0 ~ 7, m = 0 ~ 3)

ノーマルライトタイミング(各種ウェイト設定)

- ノーマルライトタイミング例を以下に示します。接続デバイスに合わせてウェイトを挿入し、出力波形を制御してください。

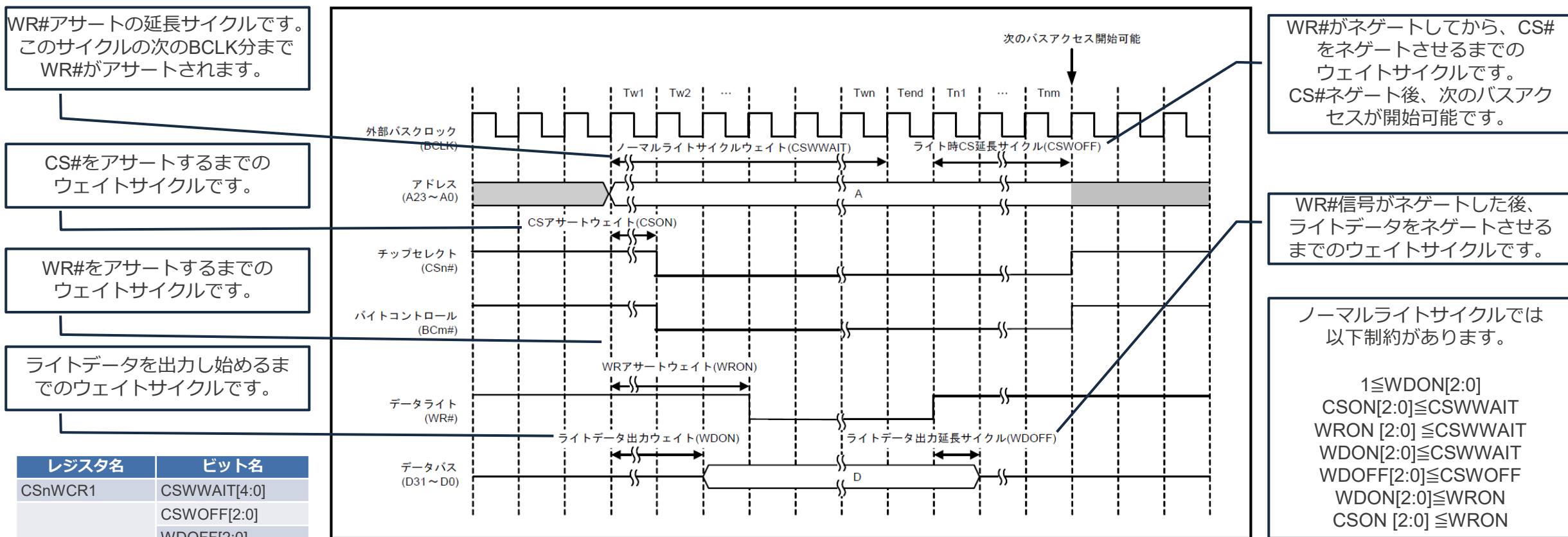


図 16.20 バスタイミング (ノーマルライト、1ライトストロブモード) (n = 0 ~ 7, m = 0 ~ 3)

レジスタ名	ビット名
CSnWCR1	CSWAIT[4:0]
	CSWOFF[2:0]
	WDOFF[2:0]
CSnWCR2	WRON[2:0]
	WDON[2:0]
	CSON[2:0]

外部バスクロックとBCLK端子

- 製品によっては、BCLK端子出力を内部のBCLKの1/2に設定する必要があります。この時の動作例と注意点を以下に示します。

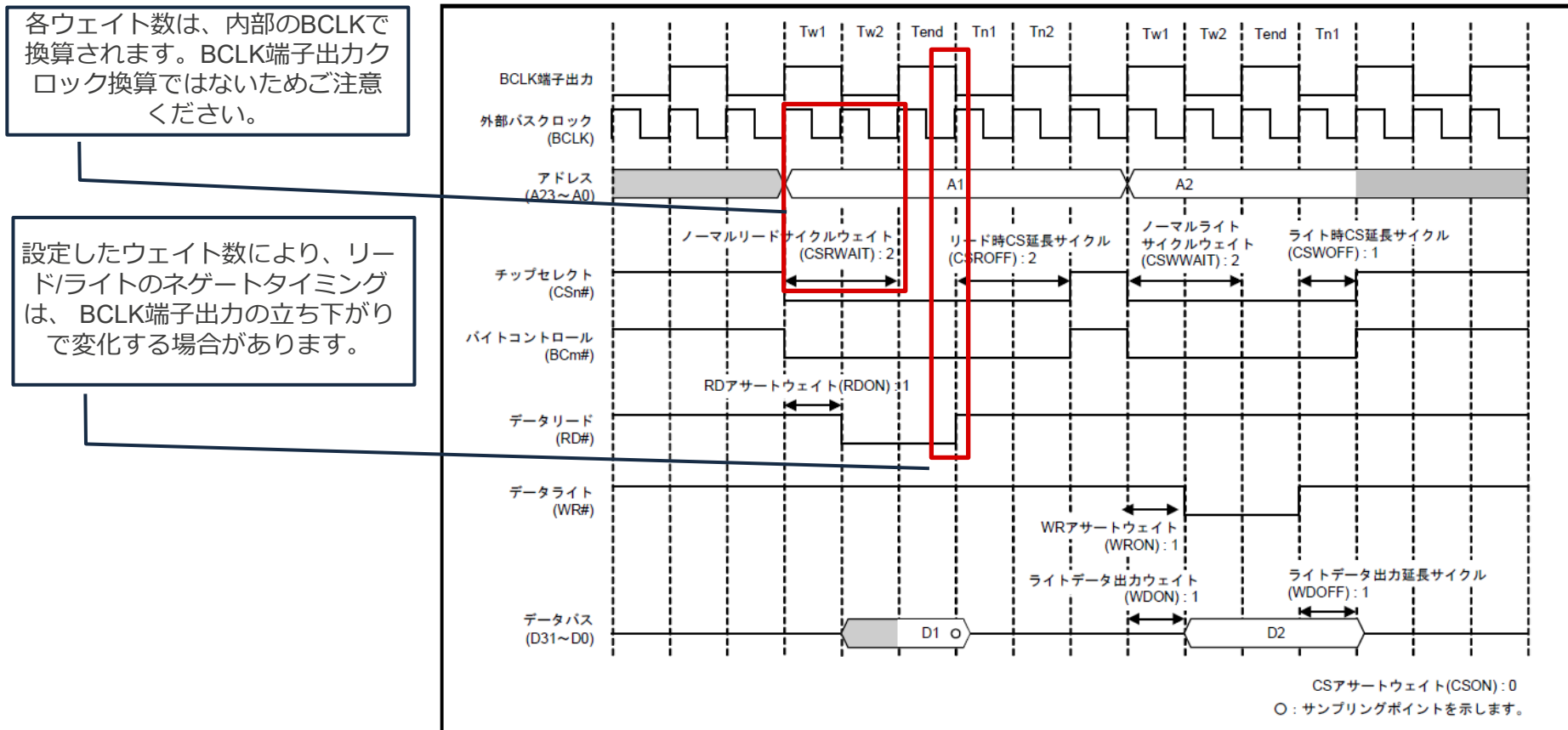


図 16.26 ノーマルアクセスの動作例 (BCLK 端子出力選択ビットで BCLK の 2 分周を設定した場合) (n = 0 ~ 7, m = 0 ~ 3)

ページアクセス(各種ウェイト設定)

- 1度の転送要求で複数のアクセスが発生したとき、ページアクセスが発生します。ページアクセスはデータアクセス間のウェイトをノーマルアクセスと異なるウェイト数にすることができます。

二回目以降のアクセス間において、ノーマルアクセスのノーマルリード(ライト)サイクルウェイトの代わりにページリード(ライト)サイクルウェイトが挿入されることがあります

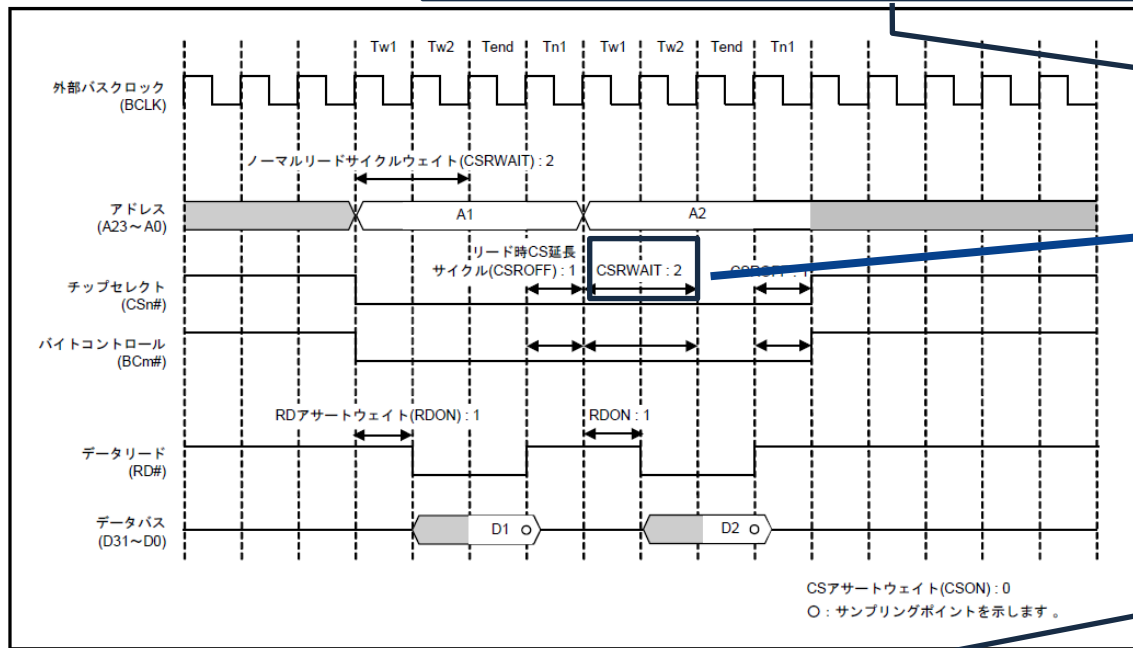


図 16.22 ノーマルリードアクセスの動作例 (1 転送要求に対して 2 回バスアクセスが発生する場合)
($n = 0 \sim 7, m = 0 \sim 3$)

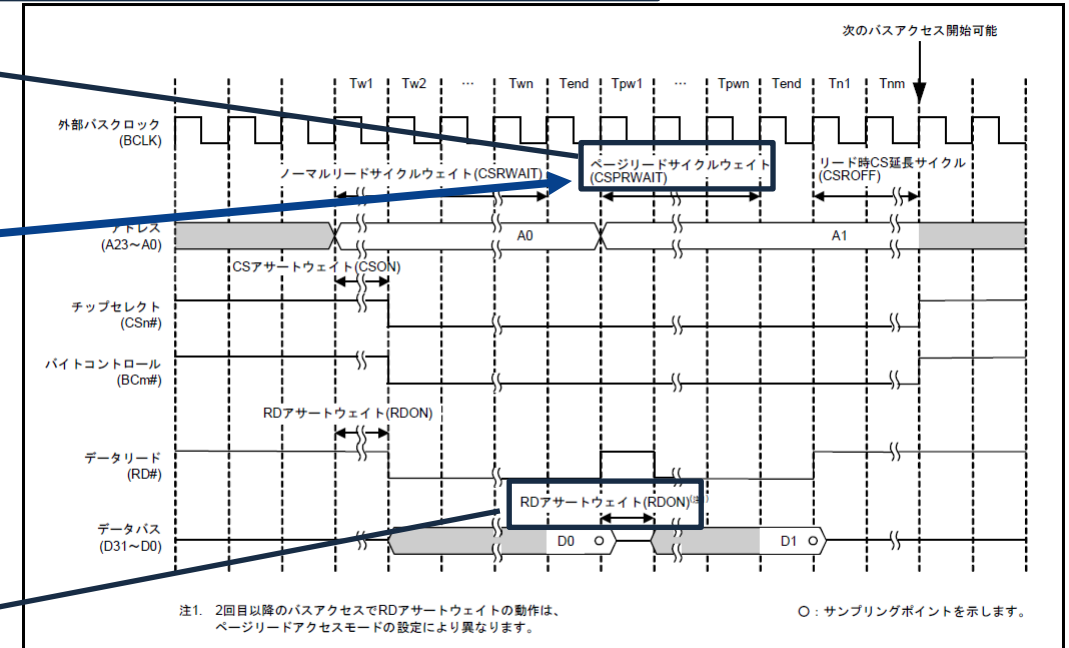


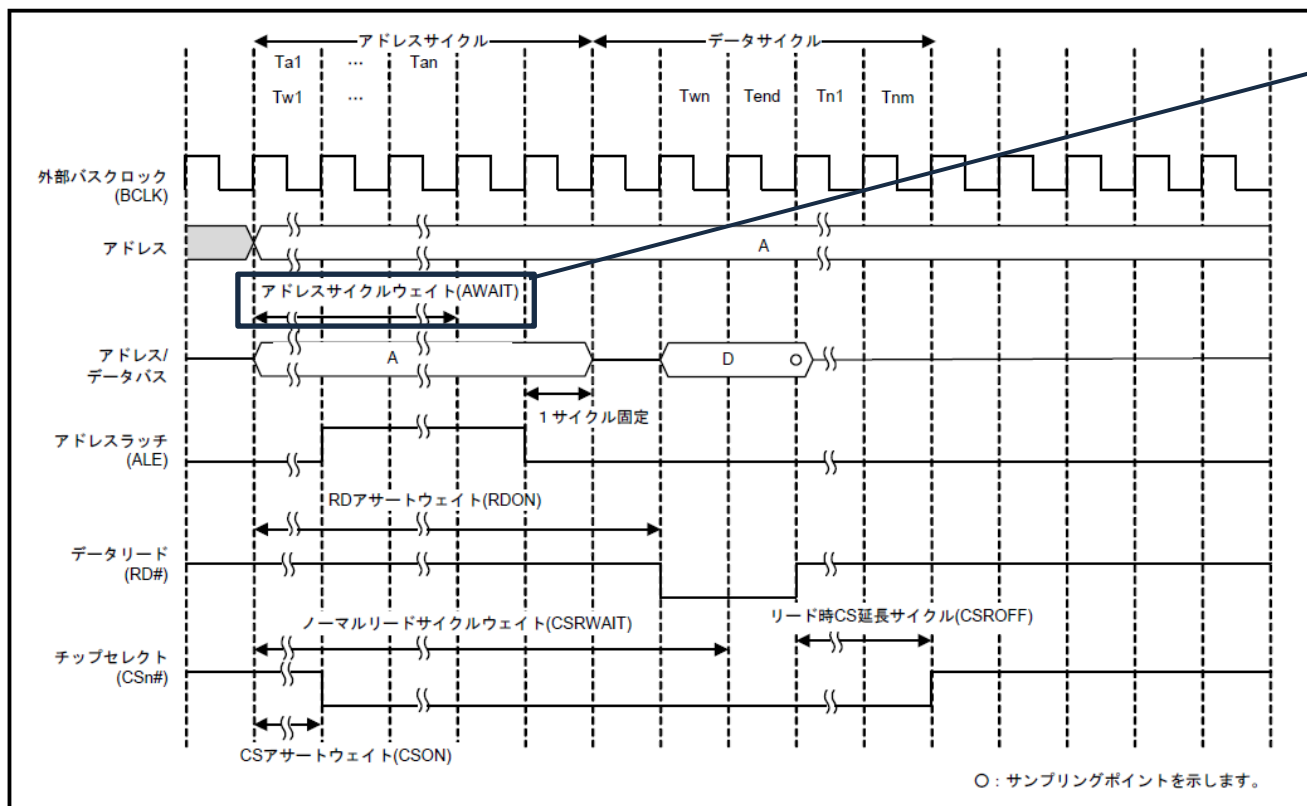
図 16.31 ページリードアクセスタイミング ($n = 0 \sim 7, m = 0 \sim 3$)

レジスタ名	ビット名
CSnWCR1	CSPWAIT[4:0](ライト時)
	CSPRWAIT[4:0](リード時)

WRアサートウェイトは1回目と同様に有効ですが、RDアサートウェイトは以下ようになります。
 SDnMOD.PRMODビットが0の場合：1回目と同様RDアサートが有効かつその間RD信号はネゲート
 SDnMOD.PRMODビットが1の場合：1回目と同様RDアサートが有効かつその間RD信号はアサート継続

アドレス/データマルチプレクス

- アドレス/データマルチプレクスアクセス例を以下に示します。接続デバイスに合わせてウェイトを挿入し、出力波形を制御してください。



最初のアドレスサイクルにおいてALEのネゲートタイミングを設定します。

アドレス/データマルチプレクスでは以下制約があります。

<p><リード時></p> <p>$CSON[2:0] \leq CSRWAIT$ $RDON[2:0] \leq CSRWAIT$ $CSON[2:0] \leq RDON$ $AWAIT[1:0] + 2 \leq RDON$ $CSON[2:0] \leq AWAIT$</p>	<p><ライト時></p> <p>$CSON \leq CSWWAIT$ $WRON[2:0] \leq CSWWAIT$ $WDON[2:0] \leq CSWWAIT$ $WDOFF[2:0] \leq CSWOFF$ $WDON[2:0] \leq WRON$ $CSON[2:0] \leq WRON$ $AWAIT[1:0] + 2 \leq WRON$ $AWAIT[1:0] + 2 \leq WDON$ $CSON[2:0] \leq AWAIT$</p>
---	---

<リード時>

レジスタ名	ビット名
CnWCR1	CSRWAIT[4:0]
CnWCR2	CSROFF[2:0]
	RDON[2:0]
	CSON[2:0]
	AWAIT[4:0]

<ライト時>

レジスタ名	ビット名
CnWCR1	CSWWAIT[4:0]
CnWCR2	CSWOFF[2:0]
	WDOFF[2:0]
	WRON[2:0]
	WDON[2:0]
	CSON[2:0]
	AWAIT[4:0]

図 16.37 アドレス/データマルチプレクス I/O インタフェース時のリードアクセスの動作例 (n = 0 ~ 7)

リカバリサイクル

- RXは外部アクセスと外部アクセスの間にリカバリサイクル(CS#端子ネゲート期間の延長)を設定することができます。接続デバイスに合わせてCSネゲート区間を設定してください。なお、バスマスタからの一回の転送要求に対して2回以上の外部バスアクセスが必要となる場合で、かつ以下リカバリサイクル挿入条件が成り立つ場合、途中のアクセスへのリカバリサイクル挿入は以下のようになります。
 - ノーマルアクセスの場合：
 - ページアクセス無効時：途中のアクセスにリカバリサイクルが挿入される
 - ページアクセス有効時：途中のアクセスにリカバリサイクルが挿入されない
 - アドレス/データマルチプレクスの場合：
 - ページアクセスの有効/無効にかかわらず途中のアクセスにリカバリサイクルが挿入される

アクセスの種類	外部アドレス空間	リカバリサイクルの挿入サイクル数設定	リカバリサイクル挿入の許可/禁止設定		
			セバレートバス	マルチプレクスバス	
リードアクセス後のリードアクセス	同じ領域	CSnREC.RRCV[3:0]*	CSRECEN.RCVEN0	CSRECEN.RCVENM0	
	異なる領域		CSRECEN.RCVEN1	CSRECEN.RCVENM1	
リードアクセス後のライトアクセス	同じ領域		CSRECEN.RCVEN2	CSRECEN.RCVENM2	
	異なる領域		CSRECEN.RCVEN3	CSRECEN.RCVENM3	
ライトアクセス後のリードアクセス	同じ領域		CSnREC.WRCV[3:0]	CSRECEN.RCVEN4	CSRECEN.RCVENM4
	異なる領域			CSRECEN.RCVEN5	CSRECEN.RCVENM5
ライトアクセス後のライトアクセス	同じ領域	CSRECEN.RCVEN6		CSRECEN.RCVENM6	
	異なる領域	CSRECEN.RCVEN7		CSRECEN.RCVENM7	

*CSnWCR2.CSROFF[2:0]=000b (リード時CS延長サイクルが0)の場合、次のバス開始サイクルは1サイクル後になります。本設定で“サイクルを挿入しない”としても、1サイクルはネゲートが入ります。

リカバリサイクル挿入例

- 以下にリカバリサイクルの挿入例を示します。

連続した外部空間アクセスの場合、
各アクセス間(CS端子ネゲート期間)でリカバ
リサイクルが挿入されます。

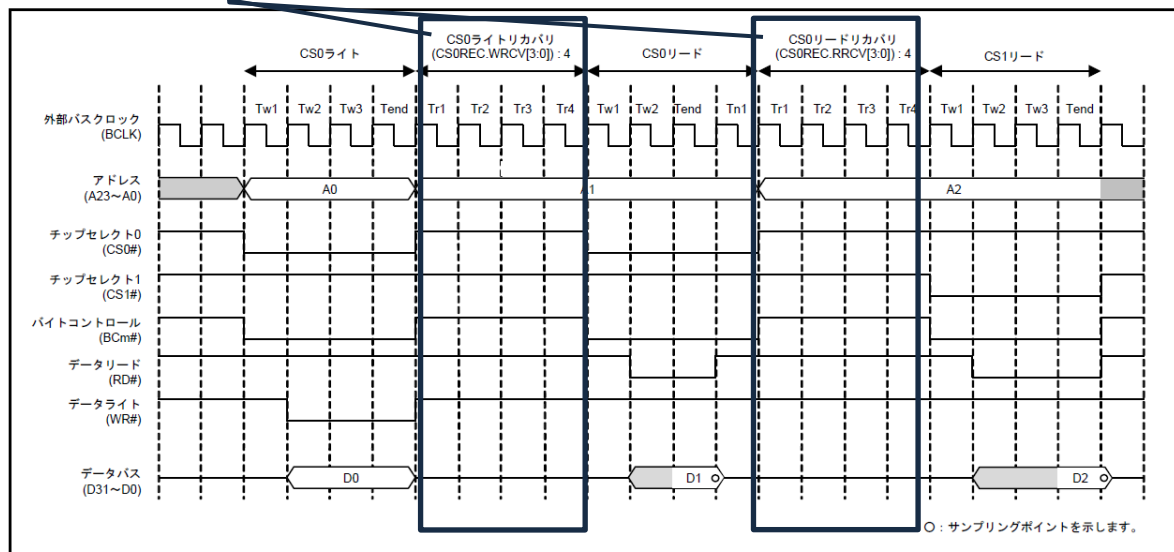


図 16.43 セパレートバスインタフェース時のリカバリサイクルの動作例 (m = 0 ~ 3)

条件によっては、バスマスタからの1回のアクセ
ス要求で二回以上のアクセスが発生する場合も
リカバリサイクルが挿入されます。

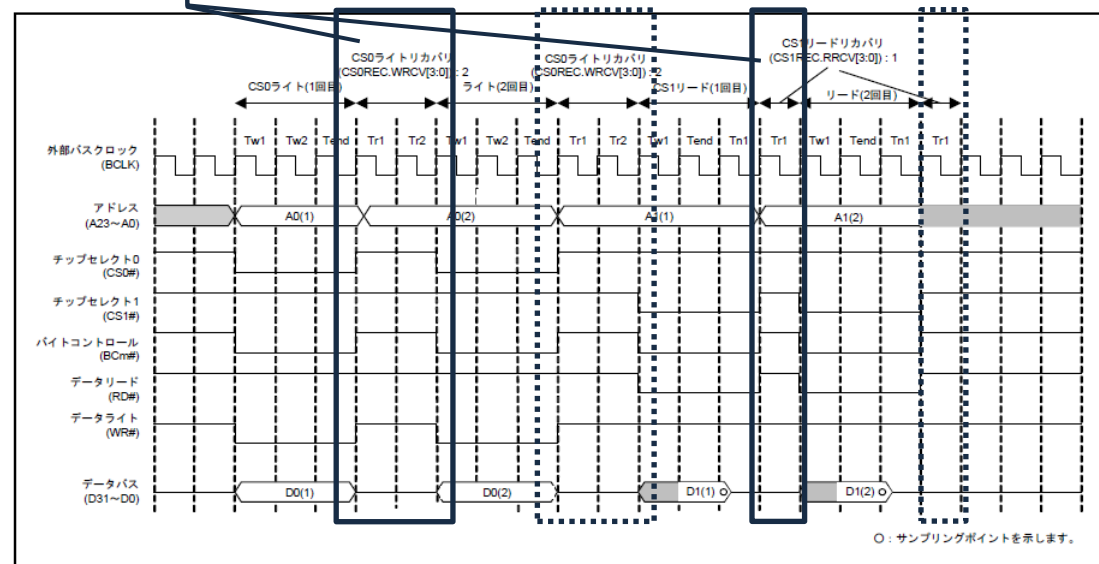


図 16.44 バスアクセスが分割された場合のリカバリサイクルの動作例 (セパレートバスインタフェース、ノーマルアクセスの場合) (m = 0 ~ 3)

SDRAM空間 レジスタ詳細

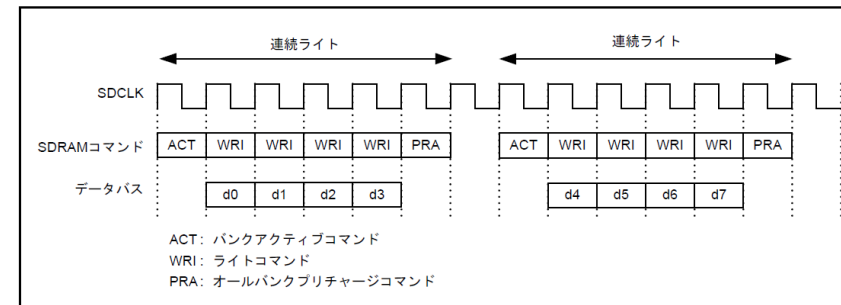
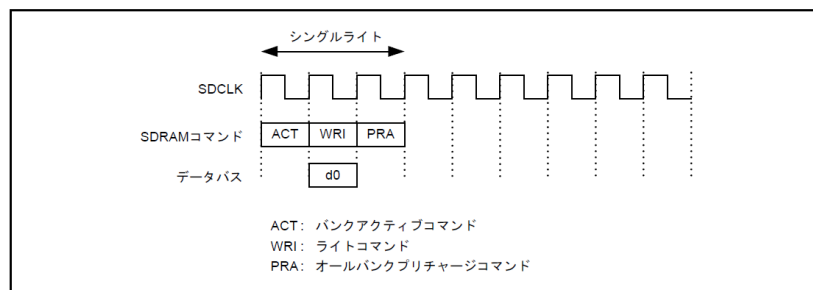
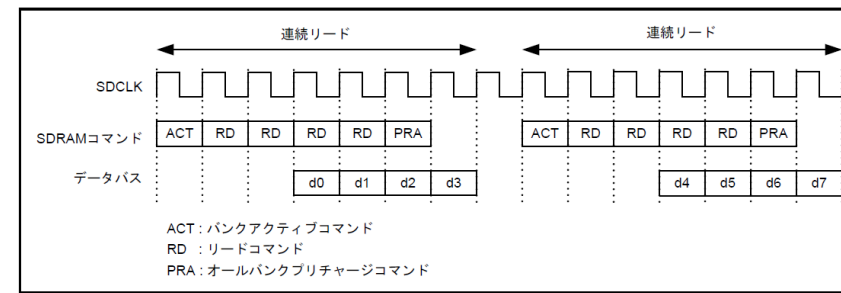
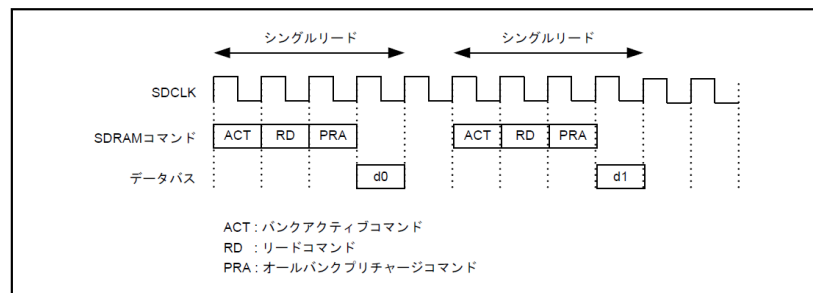
- SDRAM空間の各種レジスタの詳細とレジスタの設定可能条件について以下に示します。

レジスタ名および設定内容				レジスタ設定可能条件
基本設定	SDCCR	EXENB	SDRAM動作許可/禁止	
		BSIZE[1:0]	バス幅選択	
	SDCMOD	EMODE	SDRAM空間エンディアン設定	
	SDAMOD	BE	SDRAM連続アクセス許可/禁止	
初期化シーケンス	SDICR	INIRQ	SDRAM初期化シーケンス	SDRAMアクセス無効(SDCCR.EXENB=0) オートリフレッシュ無効(SDRFEN.RFEN=0) セルフリフレッシュ無効(SDSELF.SFEN=0)
リード/ライト設定	SDADR	MXC[1:0]	SDRAMアドレスマルチプレクスシフト量設定	
	SDTR	CL[2:0]	SDRAMカラムレイテンシサイクル数設定	セルフリフレッシュ動作中(SDSELF.SFEN=1) または
		WR	ライトリカバリサイクル数設定	SDRAMアクセス無効(SDCCR.EXENB=0)
		RP[2:0]	ロウプリチャージサイクル数設定	オートリフレッシュ無効(SDRFEN.RFEN=0)
		RCD[1:0]	カラムプリチャージサイクル数設定	セルフリフレッシュ無効(SDSELF.SFEN=0)
	RAS[2:0]	ロウアクティブ期間サイクル数設定		
リフレッシュ設定	SDSELF	SFEN	セルフリフレッシュ有効/無効	SDRAMアクセス無効(SDCCR.EXENB=0) オートリフレッシュ有効(SDRFEN.RFEN=1)
	SDRFCR	RFE[11:0]	オートリフレッシュ要求間隔サイクル設定	
		REFW[3:0]	オートリフレッシュ/セルフリフレッシュ解除サイクル数設定	セルフリフレッシュ無効(SDSELF.SFEN=0)
	SDRFEN	RFEN	オートリフレッシュ有効/無効	
	SDIR	ARF[3:0]	SDRAM初期化オートリフレッシュ間隔設定	
ARFC[3:0]		SDRAM初期化オートリフレッシュ回数設定	SDICR設定前、且つSDICRの書き換えと 同じ条件	
PRC[3:0]		SDRAM初期化プリチャージサイクル数設定		
モードレジスタ設定	SDMOD	MR[4:0]	SDRAMモードレジスタ書き込み	SDRAMアクセス無効(SDCCR.EXENB=0) セルフリフレッシュ無効(SDSELF.SFEN=0)
ステータス情報	SDSR	MRSST, INIST, SRFST	SDRAMアクセスステータス	

SDRAM リードライトアクセス

- SDRAMのリード/ライトアクセスはシングルアクセスと連続アクセスの二種類があります。連続アクセスの使用には制限があります。
- 連続アクセス条件(下記すべてを満たす場合) :
 - ① SDRAMアクセスモードレジスタの連続アクセスイネーブルビット(SDAMOD.BE)が1のとき*1
 - ② 同一のロウアドレスに対するアクセスが連続するとき
 - ③ EXDMACのクラスタ転送またはシングルアドレスモードブロック転送使用時
 - ④ EXDMACからの1転送データサイズが外部バス幅以下、かつ1回の転送要求でバスアクセスが1回で終わる場合

*1:EXDMAC以外のバスマスタからSDRAM領域へアクセスする場合、連続アクセス許可に設定することは禁止です。設定された場合の動作は保証しておりません。



オートリフレッシュ制御

- オートリフレッシュ制御に関する仕様を以下にまとめます。
 - オートリフレッシュサイクルは、SDRAM オートリフレッシュ制御レジスタのオートリフレッシュ動作有効ビット(SDRFEN.RFEN)に“1”にすると開始します。
以後**リフレッシュカウンタにより定期的にリフレッシュ要求が発生**し、オートリフレッシュサイクルが起動されます。
 - **リフレッシュ要求はリード/ライトアクセス中には受け付けられない**ため、オートリフレッシュサイクルの起動が待たされることがあります。
 - **連続アクセス中にオートリフレッシュ要求が発生した場合は、バスマスタからの1 転送要求に対するバスアクセスが終了した後に**、オートリフレッシュサイクルが開始されます。
 - **SDRAM へのアクセスとリフレッシュ要求が同時に発生した場合は、リフレッシュ要求が優先**されます。
 - **CS 領域アクセスとリフレッシュ要求は同時に行うことができます**。ただし、リフレッシュコマンド発行に必要な SDCS#/RAS#/CAS#/WE#/CKE 信号は、SDRAM アクセス専用の信号でなければなりません。
 - **オートリフレッシュサイクル開始後、再度SDRFEN.RFEN ビットを“1”にすると、リフレッシュ要求が発生**します。ただし、リード/ライトアクセス中の場合は、アクセス終了後にリフレッシュ要求が発生します。
 - **リフレッシュカウンタは、セルフリフレッシュ中は停止**します。セルフリフレッシュ復帰後は、**オートリフレッシュサイクルを起動した後、カウンタ値がリセットされカウントが再開**されます。
 - DMAC転送中にオートリフレッシュ要求が発生した場合、**DMAC転送を中断してオートリフレッシュサイクルが開始**されます。

バスエラー

- RXでは2種類のバスエラー(不正アドレスアクセスとタイムアウト)を検知できます。ただし、バスエラーを検知できるアドレス空間は限られます。以下にRX72Mでの例を示します。

アドレス	内容	種類			
		不正アドレスアクセス		タイムアウト	
		内蔵ROM有効	内蔵ROM無効	内蔵ROM有効	内蔵ROM無効
0000 0000h ~ 0007 FFFFh	メモリバス1	—	—	—	—
0008 0000h ~ 0008 7FFFh	内部周辺バス1	—	—	—	—
0008 8000h ~ 0009 FFFFh	内部周辺バス2	—	△	—	—
000A 0000h ~ 000B FFFFh	内部周辺バス3	—	△	—	—
000C 0000h ~ 000D FFFFh	内部周辺バス4	—	△	○	—
000E 0000h ~ 000F FFFFh	内部周辺バス5	—	△	—	—
0010 0000h ~ 0011 FFFFh	内部周辺バス6	予約領域	—	○	—
0012 0000h ~ 007F FFFFh			△	○	—
0080 0000h ~ 00FF FFFFh	メモリバス3	—	—	—	—
0100 0000h ~ 07FF FFFFh	外部バス (CS1 ~ CS7)	—	[IA]	—	[TO]
0800 0000h ~ 0FFF FFFFh	外部バス (SDRAM領域)	—	[IA]	—	—
1000 0000h ~ 7FFF FFFFh	予約領域	—	○	—	—
8000 0000h ~ FEFF FFFFh	メモリバス2	予約領域	—	○	—
FF00 0000h ~ FF7F FFFFh		外部バス (CS0)	—	[IA]	—
FF80 0000h ~ FFFF FFFFh			—	—	[TO]

— : バスエラーは発生しません。

△ : バスエラーは不定です。

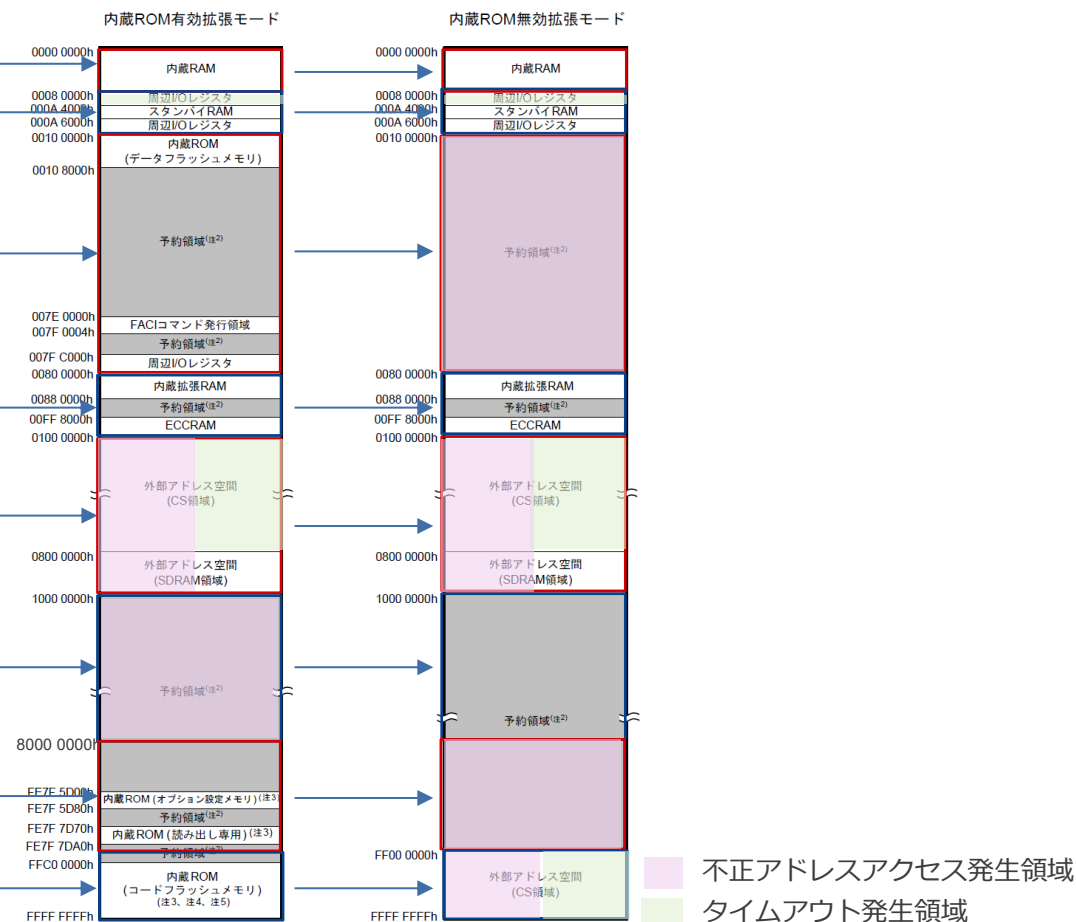
○ : バスエラーが発生します。

[IA] : これらの領域を動作禁止 (CSnCR.EXENB = 0 (n = 0 ~ 7), SDCCR.EXENB = 0) に設定している場合にバスエラーを検出します。

[TO] : 768 サイクル以内にバスアクセスが終了しない場合にバスエラーを検出します。

注 : 実装されるコードフラッシュメモリの容量は製品により異なります。製品ごとの仕様については、「64. フラッシュメモリ (FLASH)」を参照してください。

SDRAMは外部WAITがなく、768サイクルを越えるアクセスがないためタイムアウトが発生する要素はありません。



外部バスを使用したアプリケーションノート一覧

- 外部バスを使用したアプリケーションノートを準備しています。より詳細な使用方法は下記をご参照ください
 - RXファミリ 外部バスの使用例 [R01AN6594](#)
 - RXファミリ SDRAMC を使用した SDRAM のリードライト制御 [R01AN5441](#)

I/Oポート

本章はRX66Tを参考に作成しておりますが、全製品の参考資料としてお使いいただけます。

ポート機能一覧

ポート シンボル	ポート	入力プルアップ機能	オープンドレイン 出力機能	駆動能力切り替え機能	5Vトレラント
PORT0	P00, P01	○	○	通常 / 高駆動	—
PORT1	P10, P11	○	○	通常 / 高駆動	—
PORT2	P20 ~ P24, P27	○	○	通常 / 高駆動	—
PORT3	P30 ~ P33	○	○	通常 / 高駆動	—
	P36, P37	○	○	通常出力固定	—
PORT4	P40 ~ P47	○	○	通常出力固定	—
PORT5	P50 ~ P55	○	○	通常出力固定	—
PORT6	P60 ~ P65	○	○	通常出力固定	—
PORT7	P70	○	○	通常 / 高駆動	—
	P71 ~ P76	○	○	通常 / 高駆動 / 大電流出力	—
PORT8	P80, P82	○	○	通常 / 高駆動	—
	P81	○	○	通常 / 高駆動 / 大電流出力	—
PORT9	P90 ~ P95	○	○	通常 / 高駆動 / 大電流出力	—
	P96	○	○	通常 / 高駆動	—
PORTA	PA0 ~ PA5	○	○	通常 / 高駆動	—
PORTB	PB0, PB3, PB4, PB6, PB7	○	○	通常 / 高駆動	—
	PB1, PB2	○	○	通常出力固定	○
	PB5	○	○	通常 / 高駆動 / 大電流出力	—
PORTD	PD0 ~ PD2, PD4 ~ PD7	○	○	通常 / 高駆動	—
	PD3	○	○	通常 / 高駆動 / 大電流出力	—
PORTE	PE0, PE1, PE3 ~ PE5	○	○	通常 / 高駆動	—
	PE2	—	—	—	—
PORTN	PN6 (注1)	○	○	通常 / 高駆動	—
	PN7 (注2)	○	○	通常 / 高駆動	—

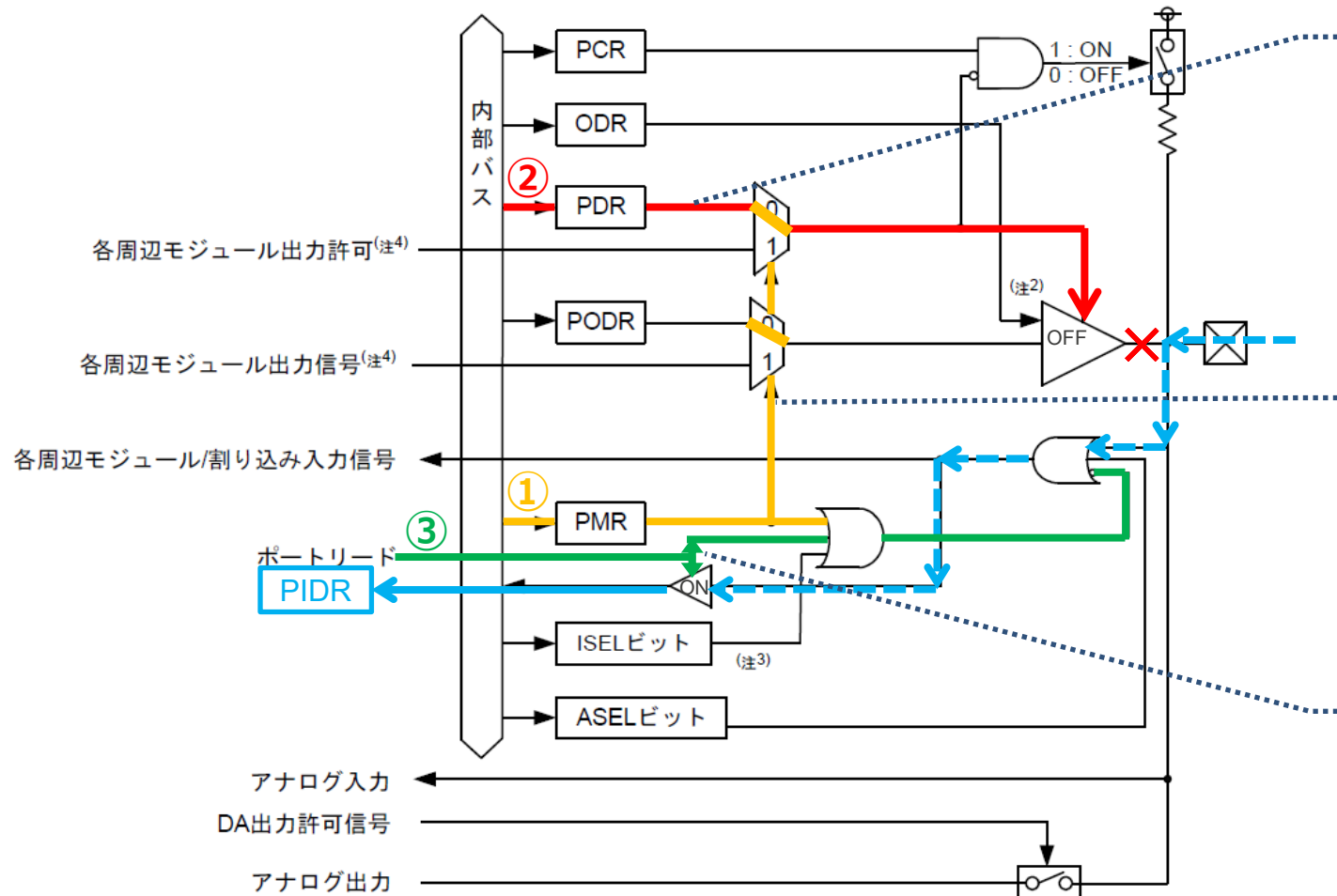
注1. 初期状態は入力プルアップです。
注2. 初期状態は入力プルダウンです。

以下ポート機能は、汎用入出力ポートと端子を共有している他の信号(シリアル等の周辺機能など)に対しても有効です。

- 入力プルアップ機能
- オープンドレイン出力機能
- 駆動能力切り替え機能
- 5Vトレラントの設定

*本機能はRX66Tを例として記載しています。

入出力ポートの動作 - 汎用入力ポート -

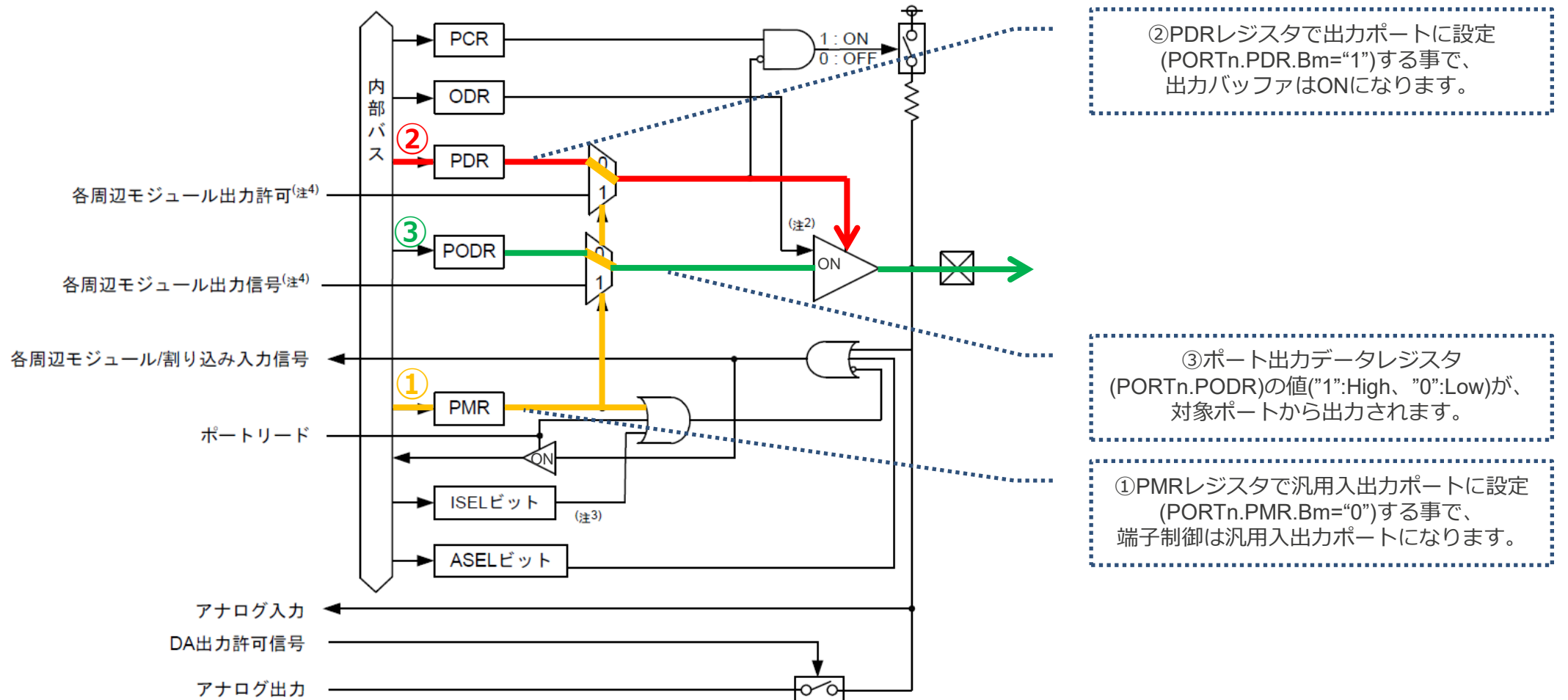


②PDRレジスタで入力ポートに設定 (PORTn.PDR.Bm="0")する事で、出力バッファはOFFになります。

①PMRレジスタで汎用入出力ポートに設定 (PORTn.PMR.Bm="0")する事で、端子制御は汎用入出力ポートになります。

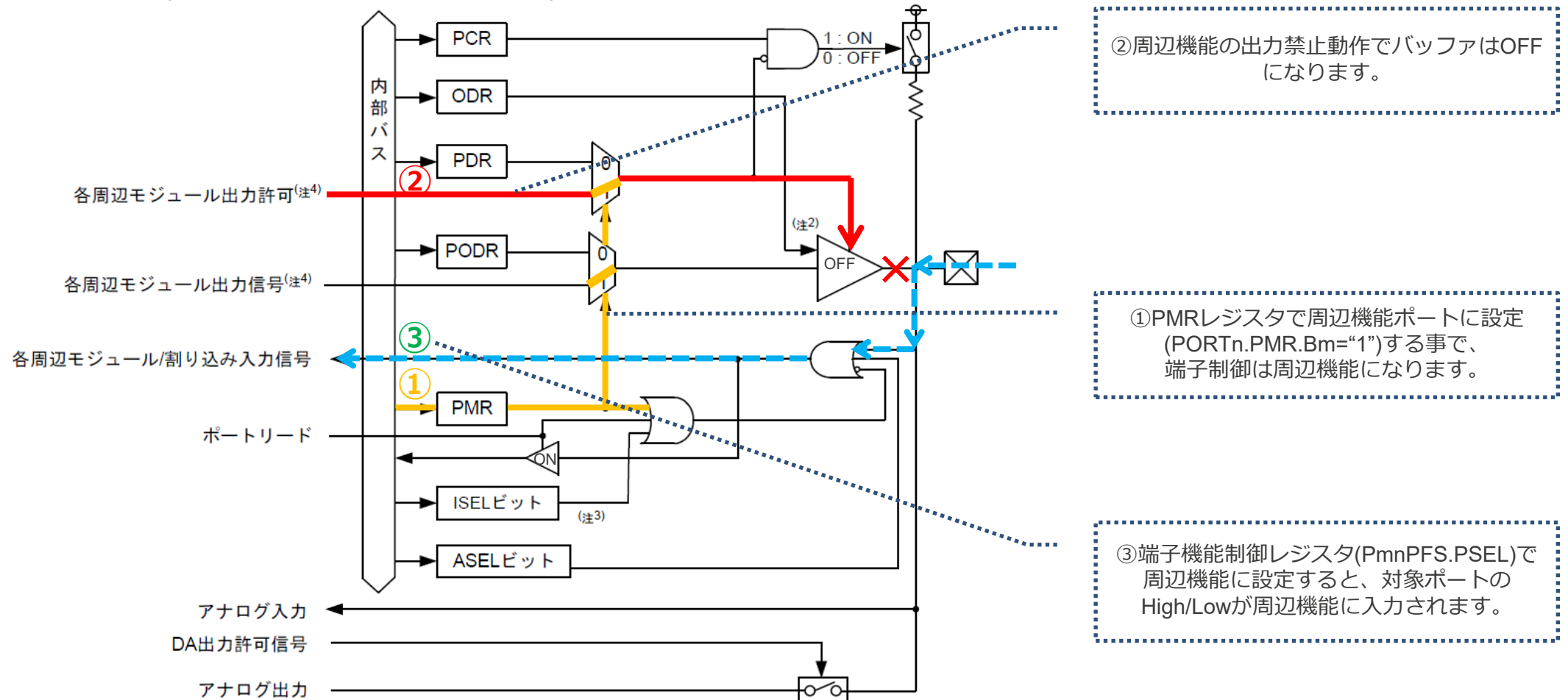
③ポートを読みに行く (PORTn.PIDR)、入力バッファがONになり、対象ポートの High/LowがPIDRにセットされます。

入出力ポートの動作 - 汎用出力ポート -



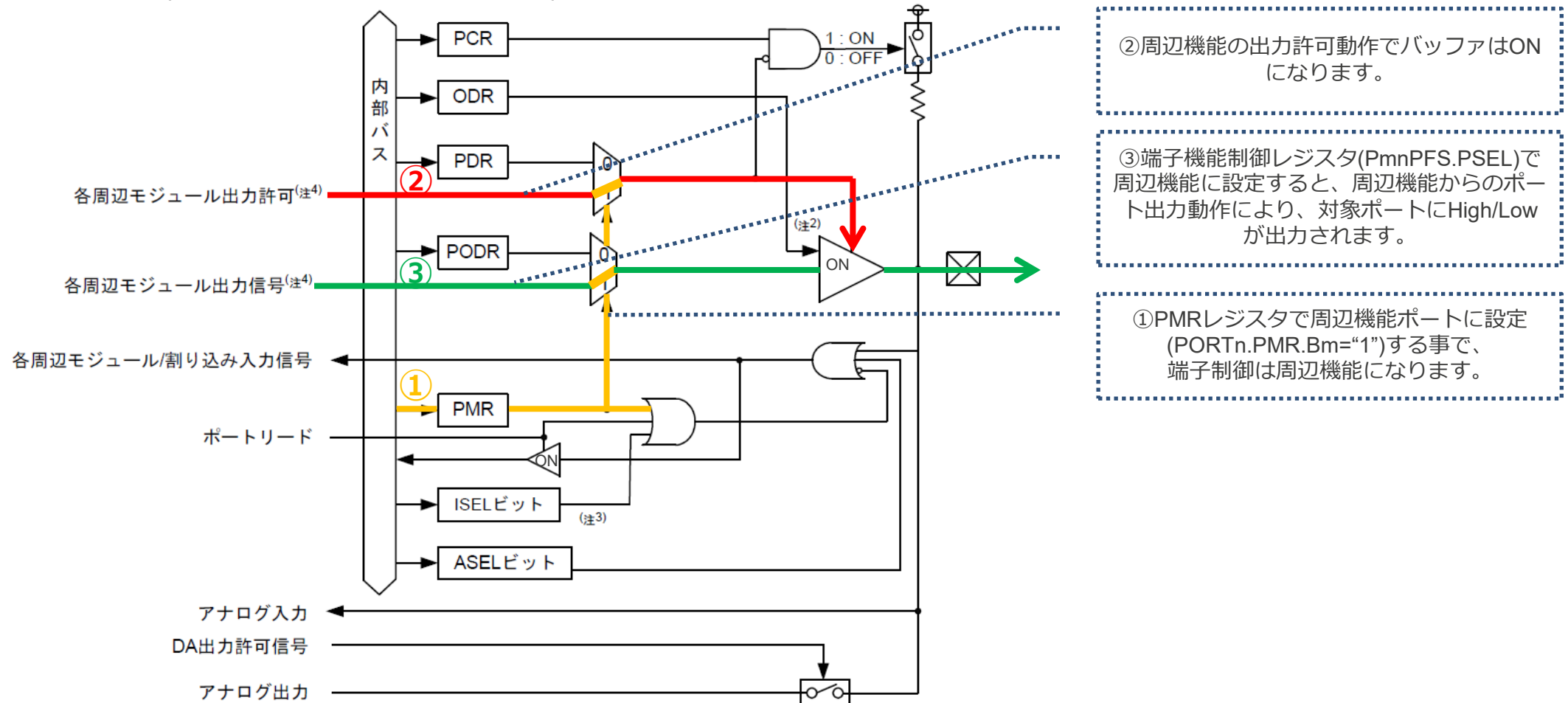
入出力ポートの動作 - 周辺機能入力 -

- 周辺機能の入出力機能(タイマ等のデジタルポート)に設定し、かつ周辺機能が入力動作の場合

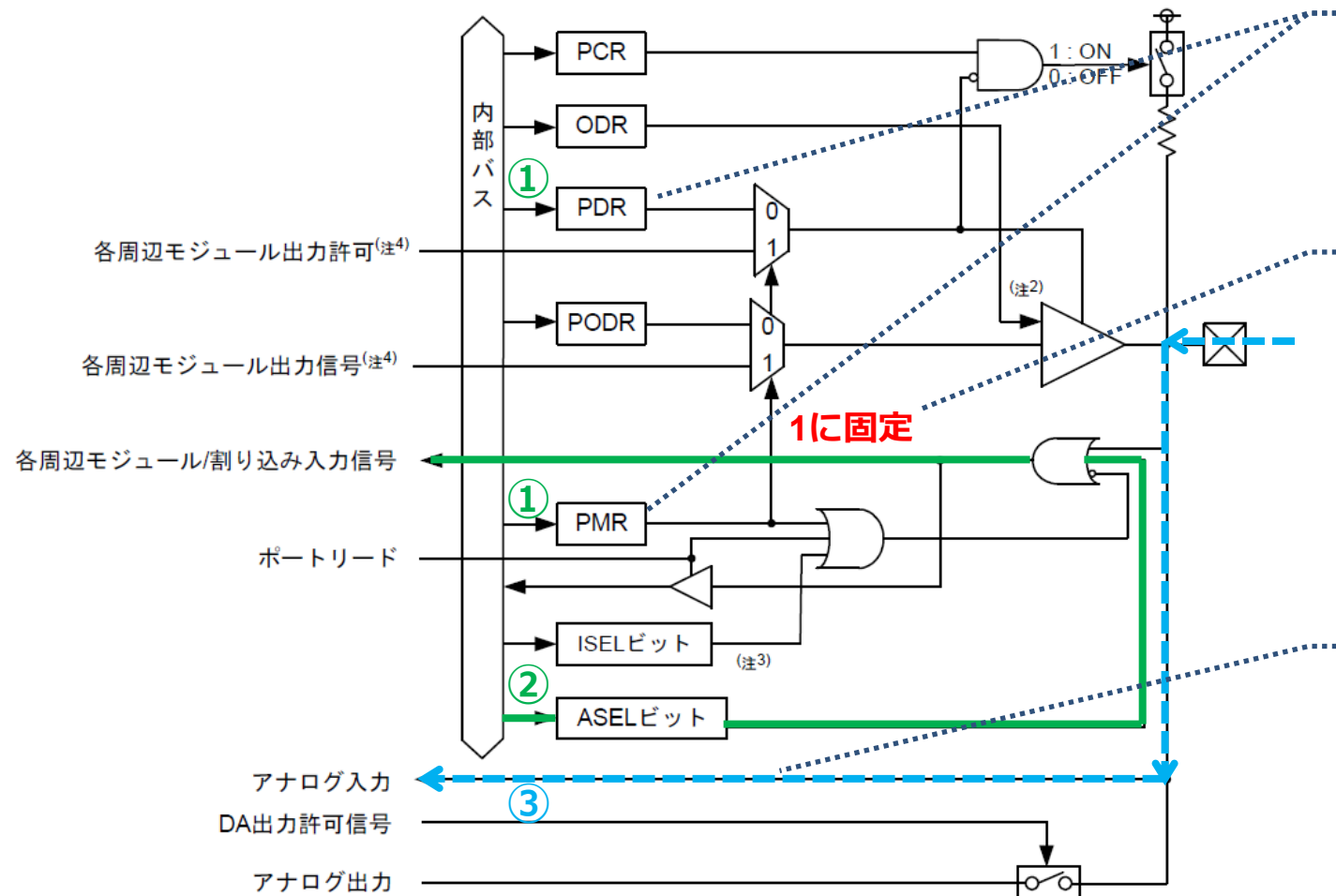


入出力ポートの動作 - 周辺機能出力 -

- 周辺機能の入出力機能(タイマ等のデジタルポート)に設定し、かつ周辺機能が出力動作の場合



入出力ポートの動作 - アナログ機能 入力 -

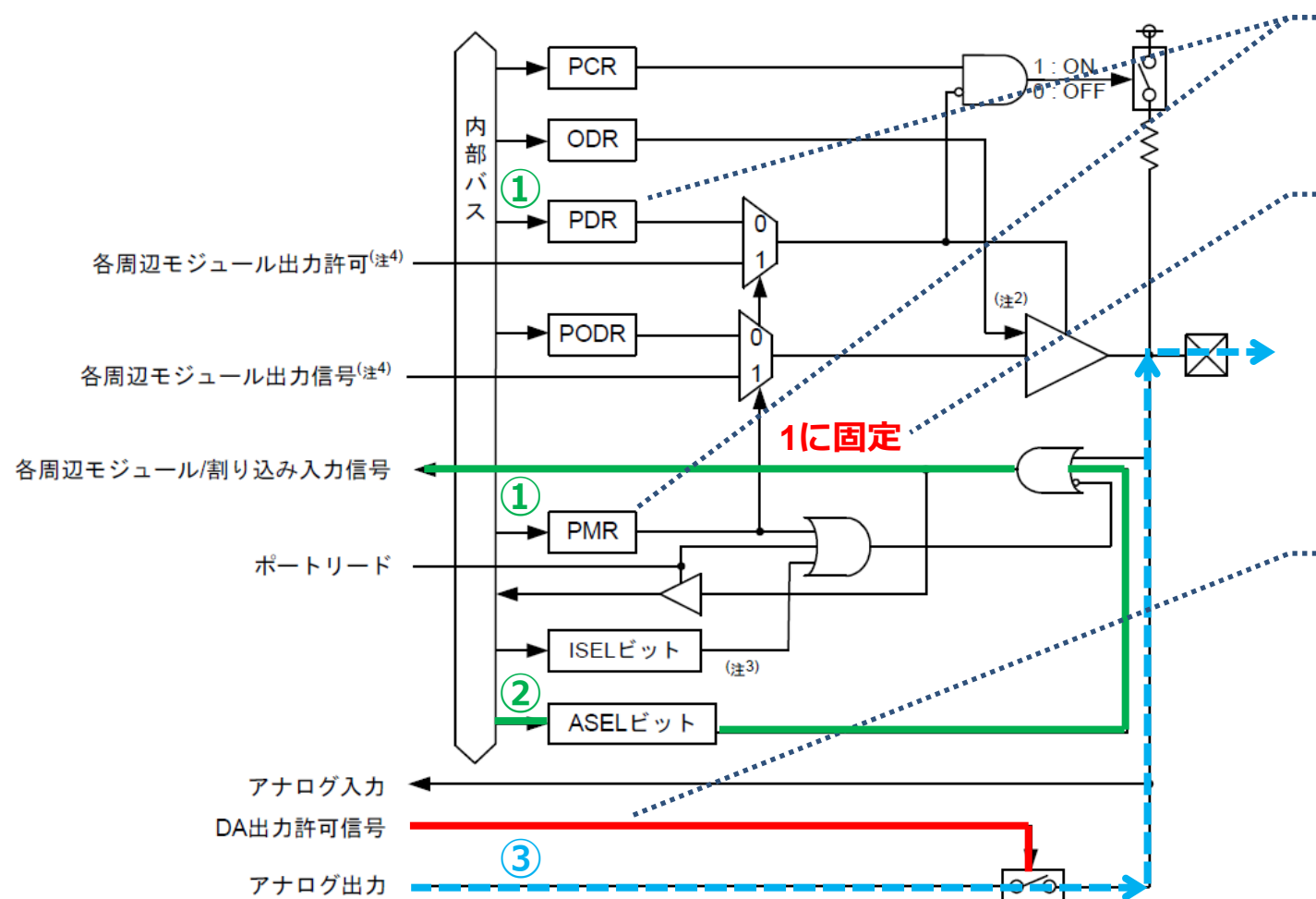


①ポートモードレジスタ(PMR)を“0”、ポート方向レジスタ(PDR)を“0”にし、当該端子を汎用入力ポートにして下さい。

②端子機能制御レジスタ(PmnPFS.ASEL)でアナログポートに設定すると、内部デジタル入力信号を“1”に固定します。

②アナログ入力動作を開始(A/Dのサンプリング動作など)すると、アナログ入力を実施します。

入出力ポートの動作 - アナログ機能 出力 -

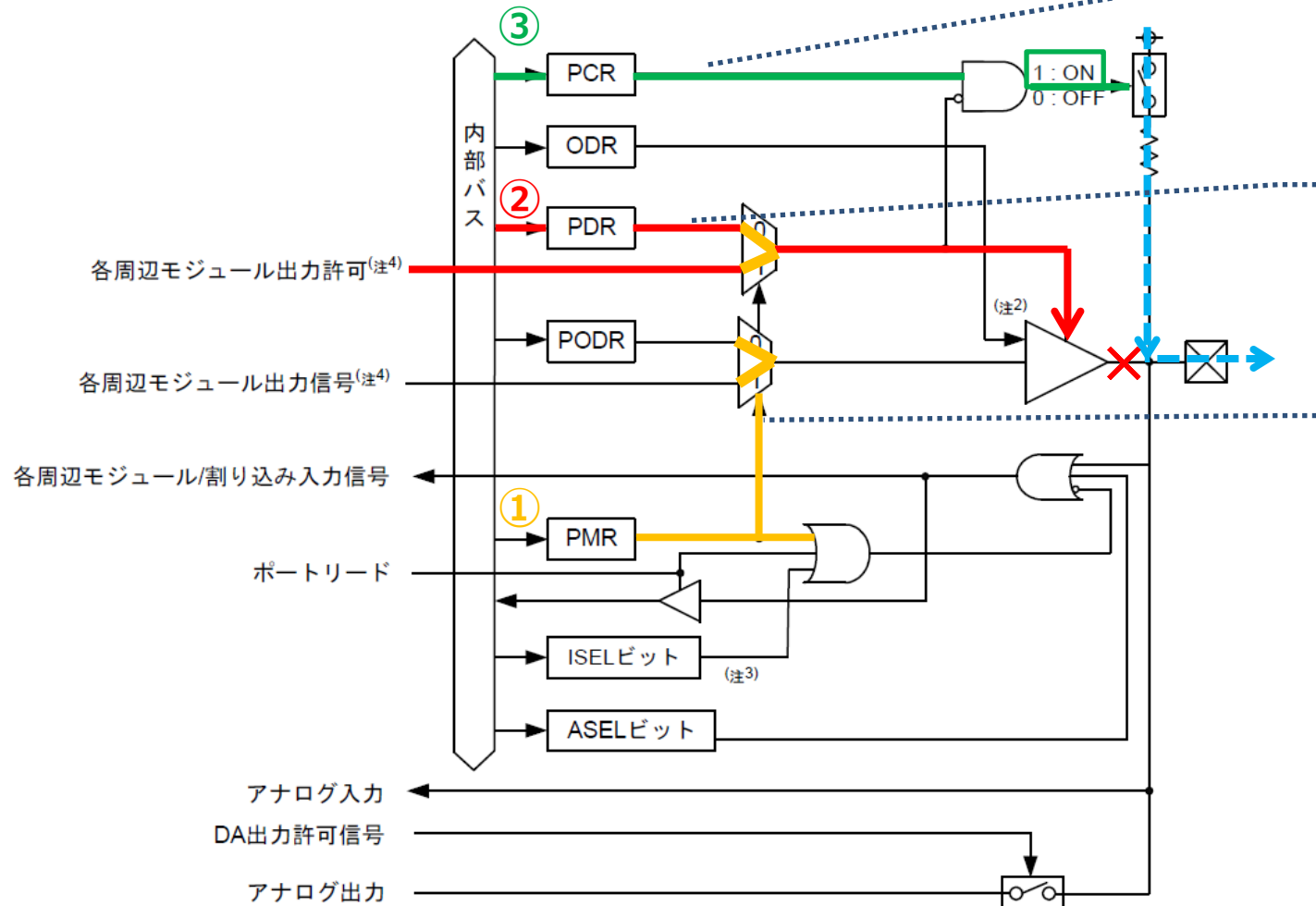


①ポートモードレジスタ(PMR)を“0”、ポート方向レジスタ(PDR)を“0”にし、当該端子を汎用入力ポートにしてください。

②端子機能制御レジスタ(PmnPFS.ASEL)でアナログポートに設定すると、内部デジタル入力信号を“1”に固定します。

③D/A 出力先選択レジスタ(DADSELR.OUTDAn)を“1”にセットすると、対象ポートからD/A信号が出力されます。

入出力ポートの動作 - プルアップ -

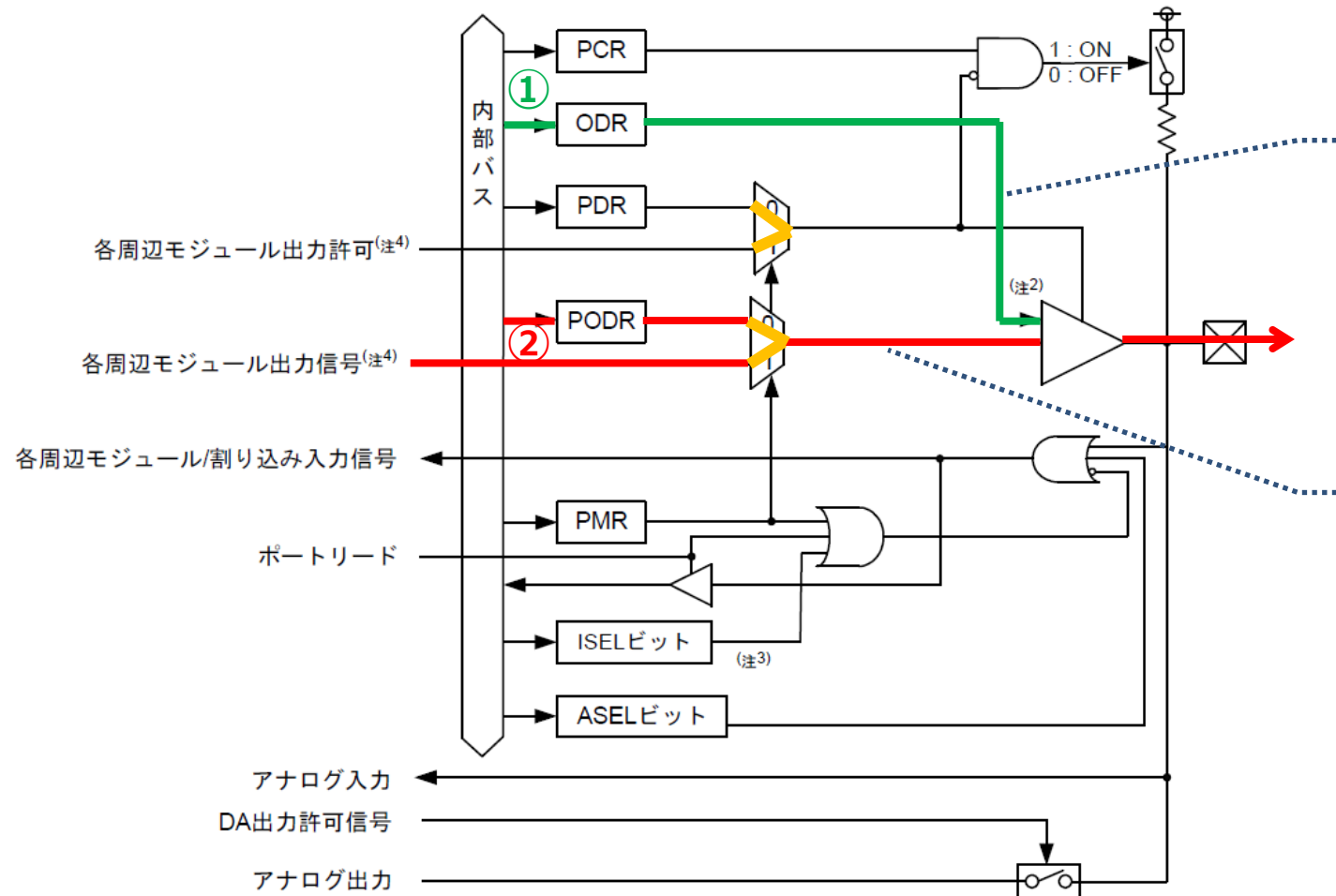


③ ①と②を設定後、プルアップ制御レジスタ(PCR)を"1"にする事でプルアップONになります(プルアップコントロールは入力ポート時にONとしてください)

② PDRレジスタで入力ポートに設定 (PORTn.PDR.Bm="0"), もしくは周辺機能の入力動作では出力バッファがOFFになります。

① PMRレジスタで汎用入出力ポート、もしくは周辺機能に設定します (PORTn.PMR.Bm="0" or "1")

入出力ポートの動作 - Nチャネルオープンドレイン -



①オープンドレイン制御レジスタ (PORTn.ODRm)を設定する事で、端子機能をNチャネルオープンドレイン端子に設定します。

②Nチャネルオープンドレイン端子に設定した端子は、“High”出力すると、N-MOSがOFFし、端子レベルはHi-Z、“Low”出力するとN-MOSがONし、端子レベルはLowとなるNチャネルオープンドレイン動作を行います。汎用出力、及び周辺機能動作の両方でご使用頂けます。

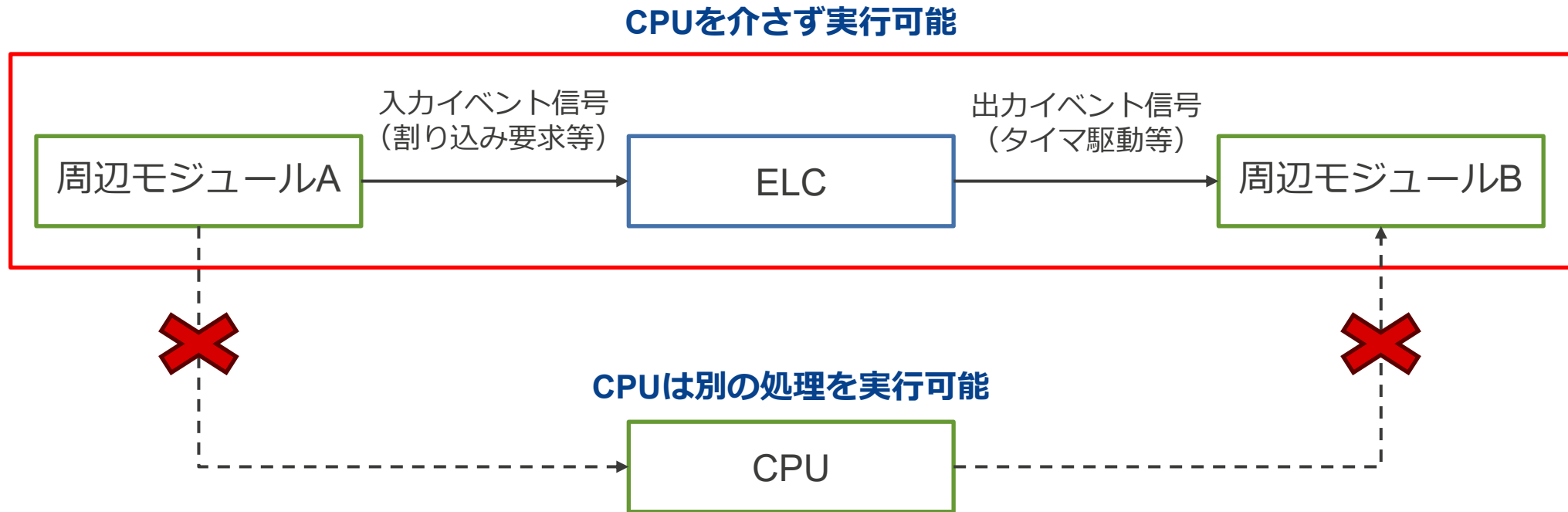
入出力ポートの設定 注意事項

- 共通
 - PORTm.PIDRレジスタを読むと、PORTm.PDRレジスタ、PORTm.PMRレジスタの値に関係なく端子の状態が読めます。
 - プルアップは、端子が入力状態のときにPORTm.PCRレジスタが“1”のビットに対応する端子の入力プルアップ抵抗が有効になります。リセット中はプルアップ抵抗が無効になります。
 - 入力プルアップ機能、オープンドレイン出力機能、駆動能力切り替え機能、5Vトレラントの設定は、汎用入出力ポートと端子を共有している他の信号に対しても有効です。
- 割り込み端子
 - ISELビットは、IRQ入力端子(外部端子割り込み)として使用する場合に設定します。周辺機能と組み合わせて使用することも可能です。ただし、同じ番号のIRQnを2つ以上の端子で許可することは禁止です。
- アナログポート
 - ASEL ビットでアナログ端子として設定する場合、ポートモードレジスタ(PORTm.PMR) の当該ビットおよびポート方向レジスタ(PORTm.PDR)の当該ビットを“0” にして当該端子を汎用入力にし、PmnPFS.ASEL ビットを“1” にしてください。このとき、端子状態を読むことはできません。
- 周辺機能ポート
 - PmnPFS.PSEL[5:0]ビットの変更は、PMR.Bnビットが“0”の状態で行ってください。
- その他
 - RIIC、RI3Cをアサインしたポートは、PCR.Bnビットを“0”にしてください (RIIC、RI3C以外の周辺機能出力では自動的にプルアップがOFFになります)。

イベントリンクコントローラ(ELC)

ELC概要

イベントリンクコントローラ(ELC)とは、周辺モジュールからのイベント信号をトリガとし、CPUを介さずに別の周辺モジュールを起動させる機能です。あるモジュールのイベントを起因に他の様々な機能を連動的に動作させることが可能です。



ELCとCPU割り込みの比較

イベントリンクコントローラを使用するケースとCPU割り込みを使用するケースの比較を以下に示します。

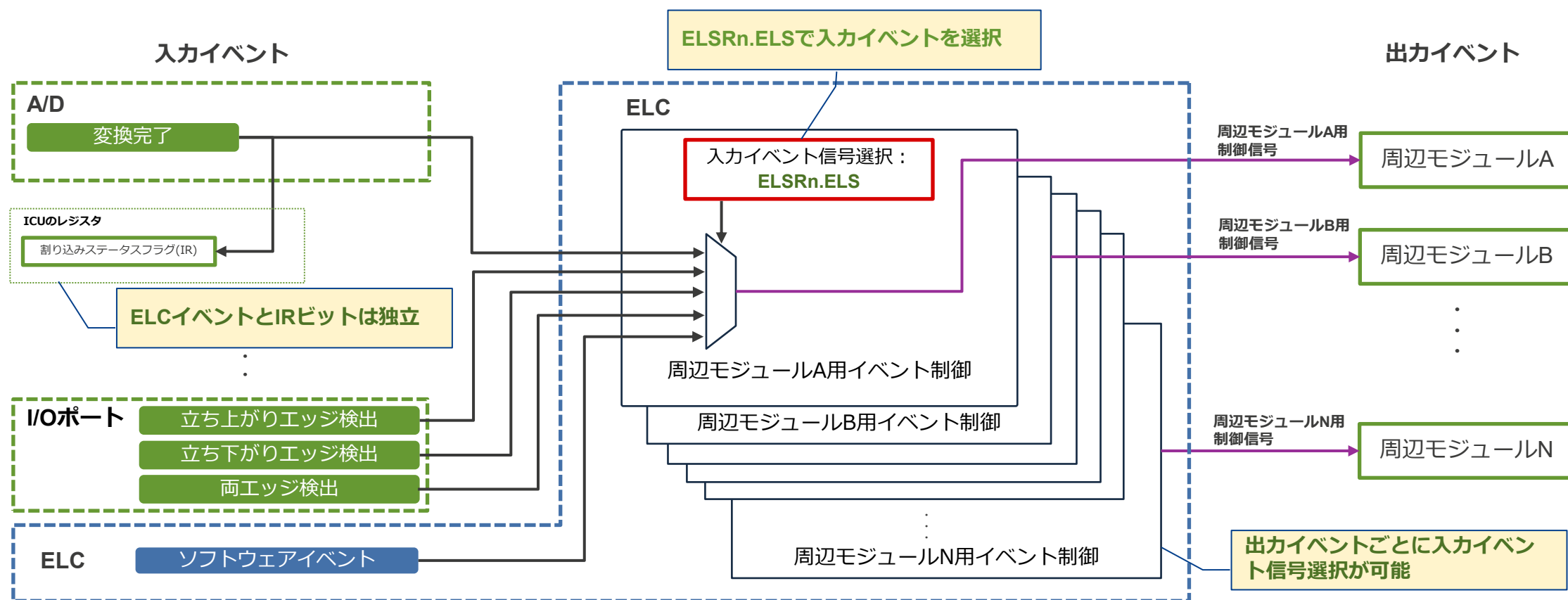
青字：メリット

項目	ELC	CPU割り込み
CPU占有	なし	あり
1 入力イベントに対する 同時出力イベント数	複数可能	複数可能 (ただし、命令によりひとつずつ実行)
複数入力イベント発生時の 出力イベント対応数	複数の出力イベントを 並列に対応可能	優先度の高い順に1つずつ 順番に対応可能

ELCの入カイベントと出カイベントの接続方法

イベントリンクコントローラの構造について下記に示します。

ELCは各出カイベントで一つの制御レジスタ(ELSRn)を保有しており、1つの入カイベントで複数の出カイベントを起動できます。入カイベントと割り込みステータスフラグ(IRビット)は独立しており、CPU割り込みへの影響は与えません。



ELCの入カイベントと出カイベントまとめ

入カイベントと出カイベントを下記にまとめます。(下記は一例になります。詳細は各RX製品の「ELC」章をご確認ください。)

入カイベント：ELCを起動させるイベントとなる機能



出カイベント：入カイベントを受けてELCが実行できる機能



ELC使用時の関連レジスタ

ELCはELCRレジスタとELSRnレジスタで制御できます。青枠のイベント使用時はその他のELCレジスタ設定が追加が必要です。赤枠の機能はELCの設定以外に各機能内のレジスタ設定が必要です。(次ページ参照)

入カイベント

タイマ ■ GPTW ・コンペアマッチ ・オーバフロー ・アンダフロー ■ AD変換開始要求 ・UVW相入力エッジ検出 ■ MTU, TPU ・コンペアマッチ ・オーバフロー ・アンダフロー ■ TMR ・コンペアマッチ ・オーバフロー ■ CMT1, CMTW0, LPT ・コンペアマッチ			
通信 ■ SCI5, RSCI11, RSCI10 ・受信エラー/エラーシグナル検出 ・受信データフル ・送信データエンプティ ・送信完了 ・受信データ一致/不一致(RSCI10のみ) ・有効エッジ検出(RSCI10のみ) ■ RIIC0, RIICHS0, RI3C0 ・受信エラー/イベント発生 ・受信データフル ・送信データエンプティ ・送信終了(RI3C0を除く) ■ EPTPC ・STCAタイマエッジ検出 ■ RSPI0, RSPiA0 ・エラー ・アイドル ・受信バッファフル ・送信バッファエンプティ ・通信/送信完了 ■ ESC ・SYNC割り込み検出			
システム ■ クロック発生回路 ・発振停止検出 ■ LVD1/2 ・電圧検出 ■ IWDT ・アンダフロー ・リフレッシュエラー ■ RTC ・周期イベント			
データ転送、演算 ■ DAMC, DTC ・転送終了 ■ DOC ・演算条件成立 ■ コンパレータ ・比較結果変化			
アナログ ■ 12bit A/D ・変換終了 ・比較条件成立/不成立	I/Oポート ■ PortB, PortE ・入力エッジ検出	その他 ■ ELC ・ソフトウェアトリガ(ソフトウェアイベント)	

ELCレジスタ

- ELCR: ELC有効無効設定
- ELSRn: 入カイベント設定
- ELOPn: タイマ系の出カイベント設定
- PGR, PDFBn, PGCn, PELm: I/Oポート関連の入出カイベント設定
- ELSEGR: ソフトウェアイベント設定

出カイベント

タイマ ■ GPTW ・カウントスタート ・カウントストップ ・カウンタクリア ・カウンタアップ ・カウンタダウン ・キャプチャ動作 ■ MTU, TPU ・カウントスタート ・カウントリスタート ・インプットキャプチャ ■ CMT1, CMTW0, TMR ・カウントスタート ・カウントリスタート ・イベントカウンタ		
システム ■ クロック ・LOC0(低速オンチップオシレータ)へ切り替え ■ IRQ ・CPU割り込み要求 ※ ※: CPU割り込み要求で利用できる入カイベント信号はI/Oポート、ELC、DOCのみです。(一部製品はLPTも含まれます。)		
アナログ ■ 12bit A/D ・AD変換開始 ■ 24bit Δ-Σ A/D ・AD変換開始 ■ 12bit D/A ・DA変換開始 ■ DSMIF ・DSMIFトリガ (測定データをレジスタに転送)		
I/Oポート ■ PortB, PortE ・ポート出力(Low/High/トグル/バッファ値/ローテート) ・ポート読み出し		
タッチセンサ ■ CTSU ・外部トリガで計測開始	ポートアウトプットイネーブル ■ POE ・出力端子をハイインピーダンスに設定	

ELC使用時の関連レジスタ

ELC関連以外の各機能側で設定が必要なレジスタを下記に示します。

入カイベント

タイマ

■GPTW

- ・A/D変換開始要求
- GTINTAD.ADTRnUEN** : イベント発生条件設定 (アップカウントのコンペアマッチ)
- GTINTAD.ADTRnDEN** : イベント発生条件設定 (ダウンカウントのコンペアマッチ)

通信

■ESC

- ・SYNCn信号
- ESCICR.SYNCnC** : エッジ条件設定 (立ち上がり/立下り)

出カイベント

タイマ

■GPTW

- ・カウントスタート
- GTSSR** : カウントスタート要因設定
- ・カウントストップ
- GTPSR** : カウントストップ要因設定
- ・カウントクリア
- GTCSR** : カウントクリア要因設定
- ・カウントアップ
- GTUPSR** : カウントアップ要因設定
- ・カウントダウン
- GTDNSR** : カウントダウン要因設定
- ・キャプチャ動作
- GTICnSR** : インพุットキャプチャn要因設定

システム

■クロック

- ・LOCO(低速オンチップオシレータ)へ切り替え
- RSTCKCR.RSTCKEN** : 本イベント使用時、ビットを0に設定

アナログ

■12bit A/D

- ・A/D変換開始
- ADSTRGR.TRSA_n** : A/D変換開始トリガ選択
- ADGCTRGR.TRSC** : A/D変換開始トリガ選択 (グループC専用)

■24bit Δ - Σ A/D

- ・A/D変換開始
- MR.TRGMD** : A/D変換開始トリガ選択

タッチセンサ

■CTSU

- ・外部トリガで計測開始
- CTSUCR0.CTSUCAP** : 計測開始トリガ選択

ELCの設定方法

I/Oポート：シングルポートとポートグループ

I/Oポートによる入力/出力イベントでは「シングルポート」と「ポートグループ」の2種類あります。

- ・シングルポート：I/Oポートの端子とイベントを1対1で接続する機能です。
- ・ポートグループ：I/Oポートの複数端子を1グループとしてイベントと接続する機能です。

以下のように、「ポートグループ」と「シングルポート」で設定するレジスタが異なりますのでご注意ください。

シングルポート

シングルポート設定：PELm

ポートグループ

グループ端子選択：PGRn

ポートグループ設定：PGCn

ポートグループ用バッファ：PDBFn

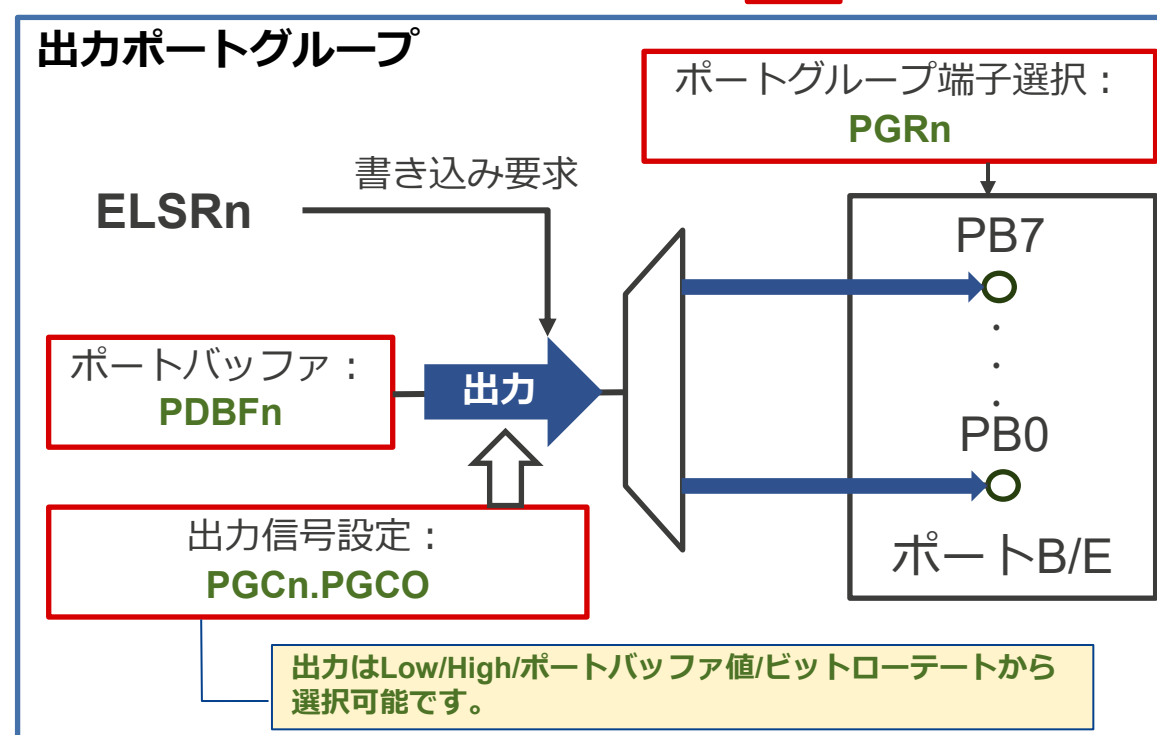
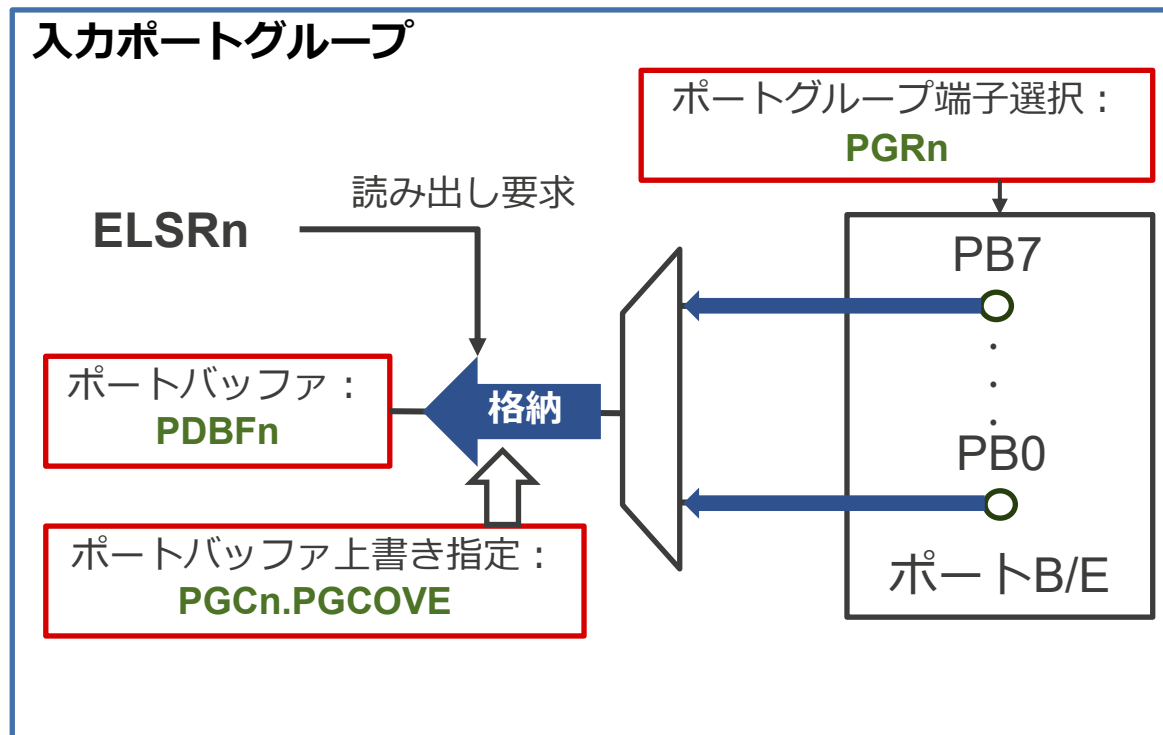
ELCの設定方法

I/Oポート：ポートグループの出カイベント

ポートグループの出カイベントは以下の2種類があります。

- ・ 入力ポートグループ：イベントが入力されると**PGRn**で選択した端子のレベルをポートバッファ(**PDBFn**)に転送する。
(**PGCn.PGCOVE**で一度だけ格納するか、上書きして何度も格納するかを選択する)
- ・ 出力ポートグループ：イベントが入力されると**PGCn.PGCO**で設定した条件で**PGRn**で選択した端子から出力する
(**PGCn.PGCO**で出力方法を選択する)

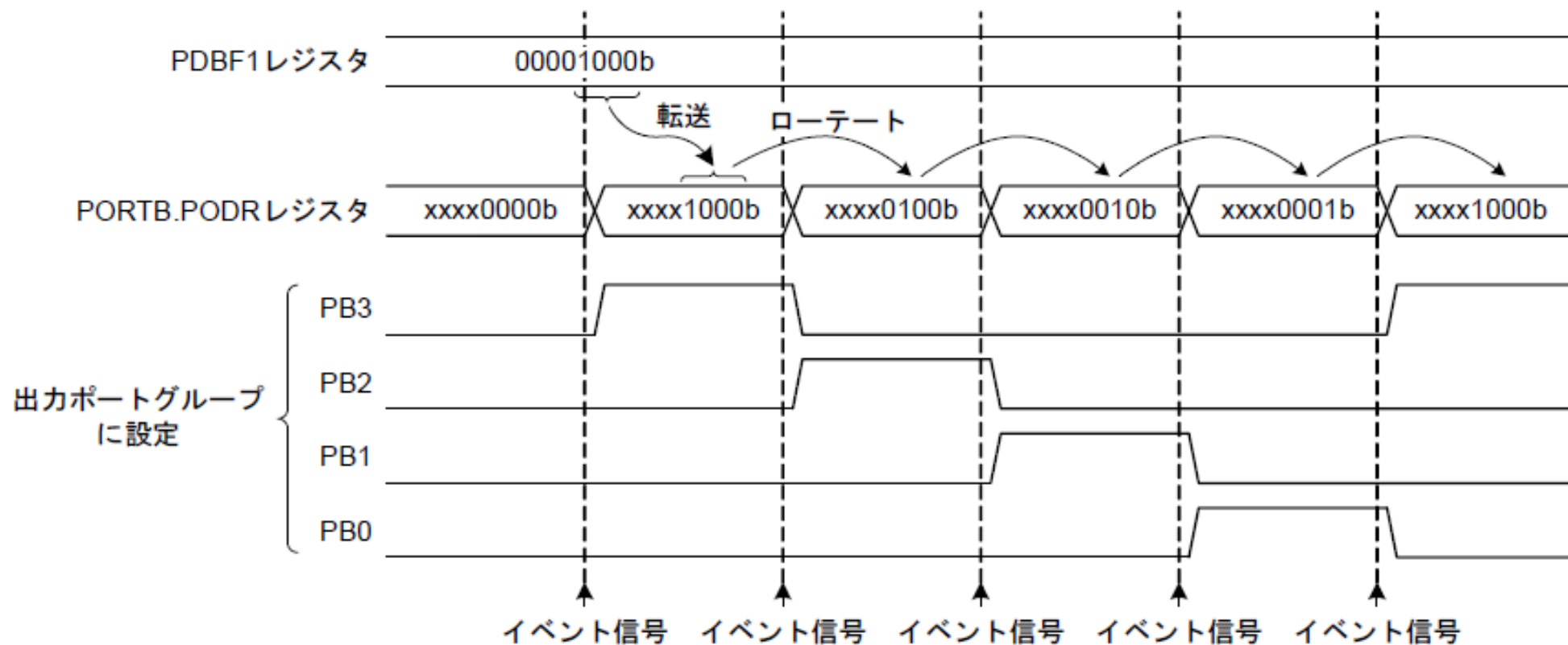
 : 設定が必要なレジスタ



ELCの設定方法

補足)ビットローテート出力とは

出力イベントをポートグループで使用する際、PDBFnレジスタに設定したデータ値を入力イベント発生ごとにMSB→LSBにローテートしながら各ポートから値を出力します。例えば、下記のようにPDBFnレジスタを00001000bと設定すると、イベント入力ごとにHighを出力するポートを切り替えさせることが可能です。



ELC設定の補足及び注意事項

■補足事項

- ELCはスリープモードなどのCPU停止状態でも動作可能です。
ただし、ELC章の「使用上の注意事項」にありますようにELCと対象の周辺モジュールへのクロック供給が停止するモードでは動作しませんのでご注意ください。
- ELCに使用する入カイベントはDMACやDTCの転送要因としても同時に使用可能です。

■注意事項

- ELC使用時の注意点がELC章の「使用上の注意事項」にありますので、必ずご一読ください。
- 使用する周辺モジュール側の「使用上の注意事項」も必ずご一読ください。
- ICU割り込みを出カイベントとする場合、使用できる入カイベントに制限があります。
詳細はハードウェアマニュアルをご確認ください。

活用例

ELCは様々な用途に応用可能で、以下のような活用法がございます。

① : CPUを介さずに周辺モジュールを実行する

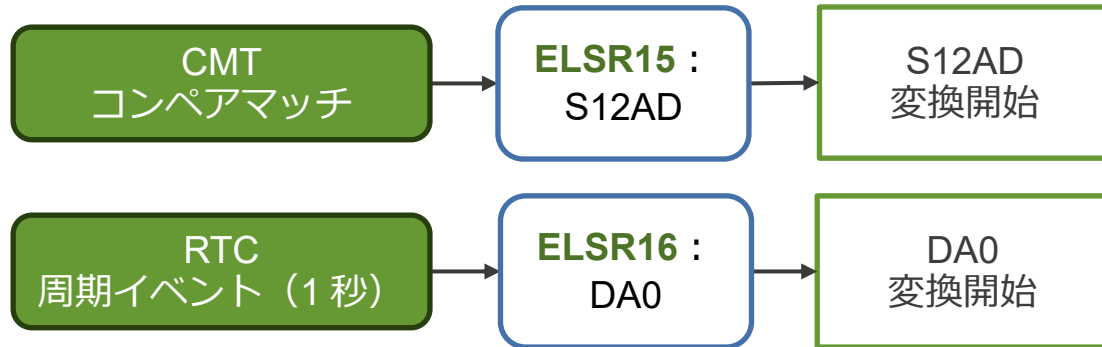
② : 1つのイベント信号で異なる複数の周辺モジュールを実行する

③ : 周辺モジュールを数珠つなぎしてシーケンスで実行する

活用例はRX72Mを参考に作成しておりますが、全製品の参考資料としてお使いいただけます。

活用例①： CPUを介さずに周辺モジュールを実行する

ELCの特性を活かし、CPUを介在させず一定周期で周辺モジュールを実施する例を示します。



例1：コンペアマッチタイマやRTCの周期イベントで12ビットA/DやD/Aの変換開始

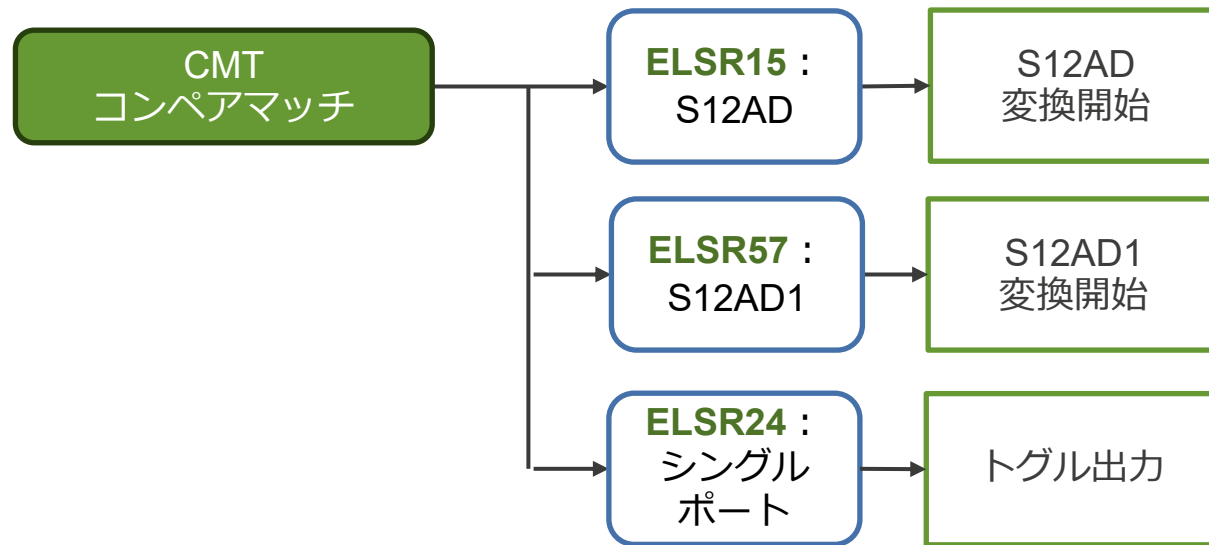


例2：一定周期でのビットローテート出力 (PDBFレジスタに設定した値をMSBからLSBへローテート出力)

活用例②：

1つのイベント信号で異なる複数の周辺モジュールを実行する

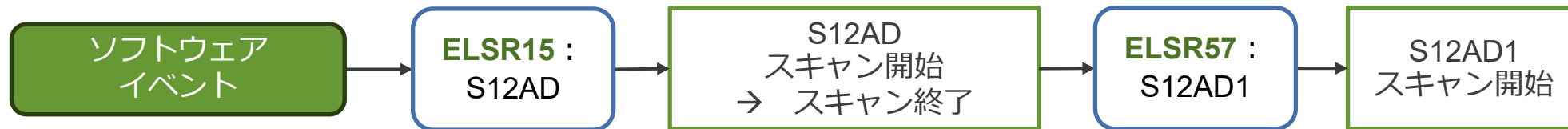
ELCの特性を活かし、1つの入力イベント信号で複数の周辺モジュールを起動させる例を示します。



例：CMTのコンペアマッチでS12AD、S12AD1、I/O出力(トグル)を同時駆動

活用例③： 周辺モジュールを数珠つなぎしてシーケンスで実行する

ELCの特性を活かし、1つの入カイベント信号で複数の周辺モジュールを連続して起動させる例を示します。



例：ソフトウェアイベントでS12ADを変換開始、S12AD変換終了を入カイベントとしてS12AD1の変換開始

応用例

具体的な応用例を以降のページで紹介します。

応用例① : CPUを介さずにSCI受信時データ転送とタイムアウト再設定を行う

応用例② : CPUを介さずにキー・スキャンを行う

応用例③ : CPUを介さずに周期的にAD変換を行い、変換結果をRAMに保存する

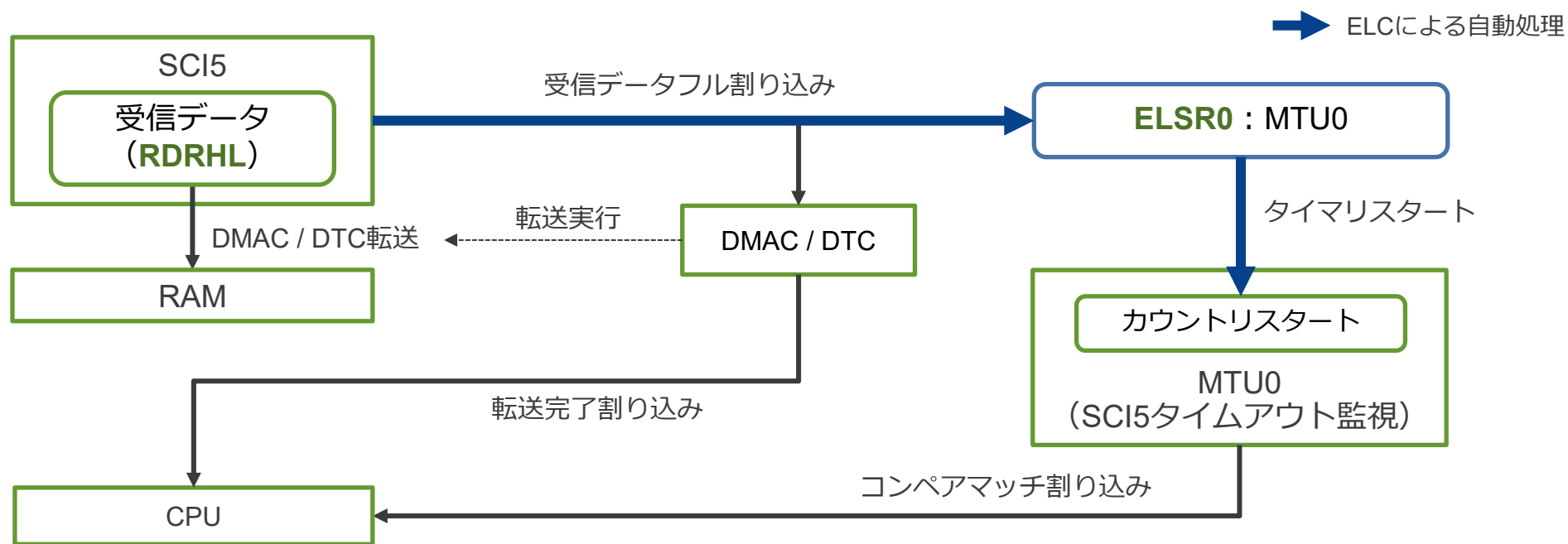
応用例はRX72Mを参考に作成しておりますが、全製品の参考資料としてお使いいただけます。

応用例①：

CPUを介さずにSCI受信時データ転送とタイムアウト再設定を行う

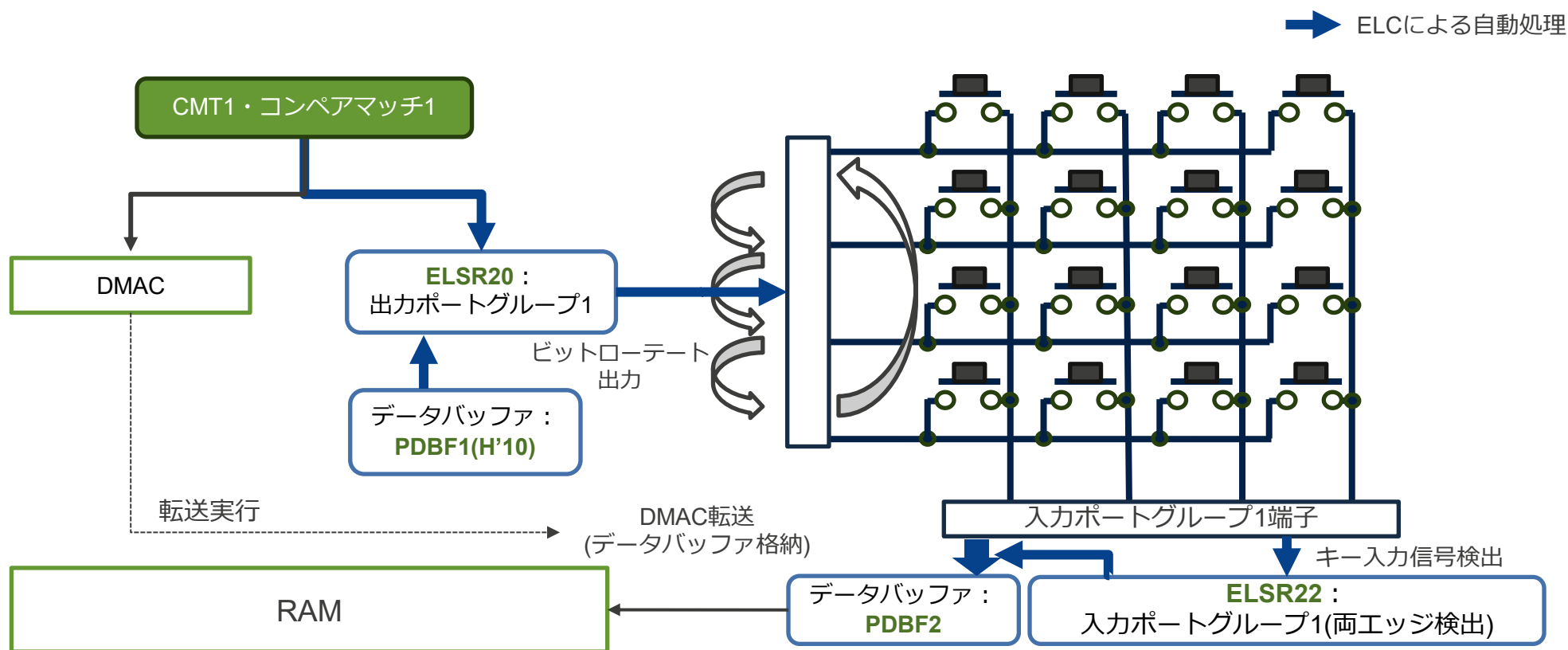
一定間隔でSCI受信が必要なシステムにおいて、SCI受信後次の受信終了までのタイムアウト検知をMTUを使用して自動的に行う例になります。CPUは一切介在しません。

SCIの受信データフル割り込みをELCの入カイベントとし、ELCがMTUのタイマカウントをリスタートさせます。タイマは受信間隔に合わせてコンペアマッチするように設定しています。リスタートが間に合わない場合はコンペアマッチ割り込みが発生しタイムアウトしたことをCPUに知らせます。なお、受信データフル割り込みを転送要因としてDMAC / DTCによるRAMへの受信データ転送を同時に行います。



応用例②： CPUを介さずにキー・スキャンを行う

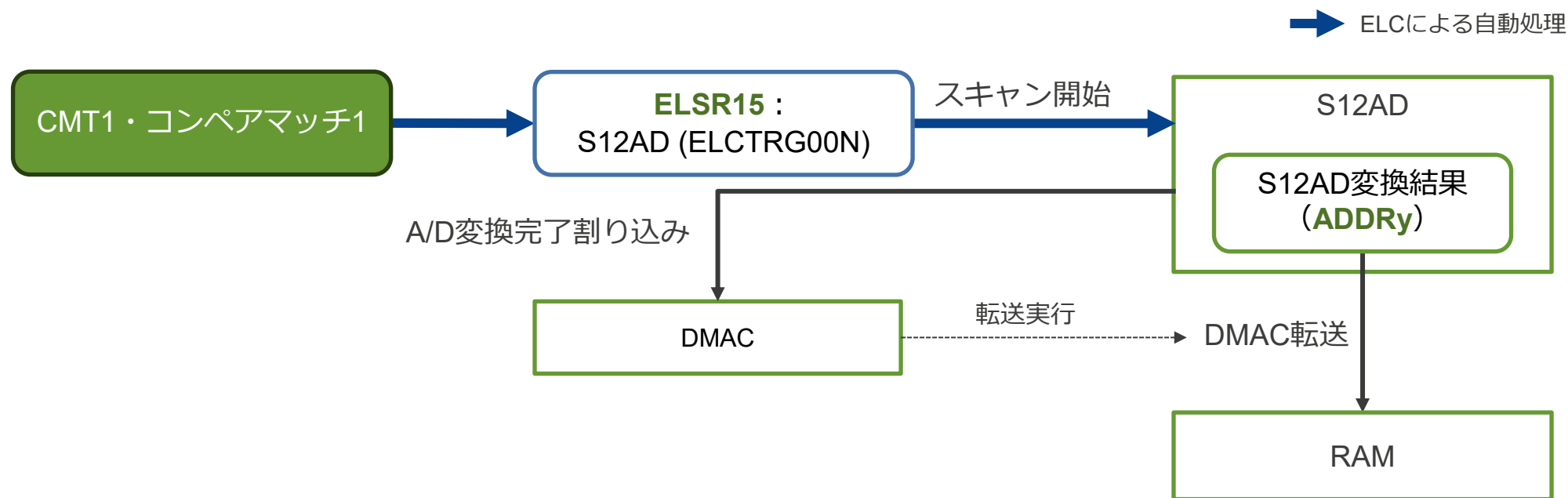
キースキャン機能を実現する例になります。入力イベントのCMTコンペアマッチが発生するとELCが起動し、出力ポートグループをローテート出力させ、行を順番に有効にしていきます。さらにこの入力イベントを転送要因としてDMACがデータバッファ(PDBF2)からRAMにデータ転送を行います。列側の操作では、ボタンの押下により入力ポートグループ端子が変化し入力イベントが発生します。これによりELCが起動し、入力ポートグループ端子の値をデータバッファ(PDBF1)に格納します。



応用例③ :

CPUを介さずに周期的にA/D変換を行い、変換結果をRAMに保存する

ELCとDMACを組み合わせることで、周期的にA/D変換処理から変化データ格納までCPUを介さず行うことができます。入カイベントであるCMTのコンペアマッチによりELCはA/D変換を開始させます。また、A/D変換終了割り込みを転送要因としてDMACによりA/D変換結果をRAMに保存します。DMACはフリーランモードにすることで、転送回数の再設定をする必要がなくなります。



ELCを使用したアプリケーションノート一覧

ELCを使用したアプリケーションノートを準備しています。より詳細な使用方法は下記をご参照ください。

- RX200シリーズ ELCによるLVDとLOCOのリンク動作 [R01AN1099](#)
- RXファミリ 既存の周辺機能を応用したタンパ検知方法 [R01AN7654](#)

MTU

タイマ比較

RXが搭載する各タイマの比較を以下に示します。

*製品により異なります

動作モード	GPT/GPTW	MTU3	TPU	TMR	CMT	CMTW
ビット幅	16/32	16	16	8	16	32
最大チャンネル数*	10	9	6	4	4	2
最大PWM出力本数*	20	14	15	4	-	2
最大動作クロック	ICLKと同等	ICLKと同等	PCLK	PCLK	PCLK	PCLK
PWM出力	○	○	○	○	-	○
相補PWM出力	○	○	-	-	-	-
正相/逆相独立コンペア動作	○	-	-	-	-	-
入力キャプチャ	○	○	○	-	-	○
位相計数モード	○	○	○	-	-	-
アウトプットコンペア	○	○	○	○	-	○
外部クロックカウント	○	○	○	○	-	-
フリーラン動作	○	○	○	○	○	○
コンペアマッチ動作	○	○	○	○	○	○
高分解能出力制御	○	-	-	-	-	-

MTU機能一覧

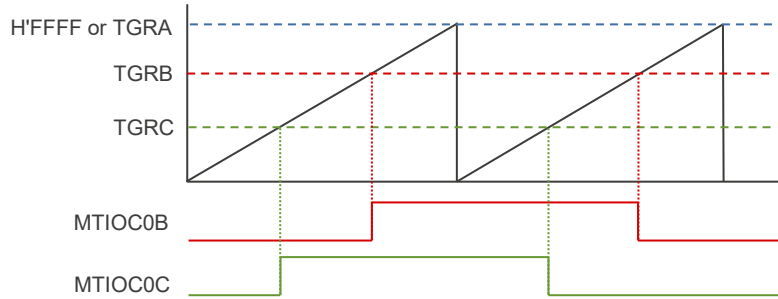
MTUが保有する動作モードと、バッファ動作をサポートするレジスタ、オプション動作の一覧を下記に示します

動作モード一覧	バッファ動作対象レジスタ (MTU0,3,4,6,7,9)	オプション動作	該当チャネル
ノーマルモード アウトプットコンペア インプットキャプチャ	TGR(アウトプットコンペア/ インプットキャプチャ)	同期動作 ノイズフィルタ A/D変換開始要求/A/D変換ディレイド	MTU0,1,2,3,4,6,7,9 *A/D変換開始要求はMTU4,7のみ
PWM1/2モード アウトプットコンペア インプットキャプチャ	TGR(アウトプットコンペア/ インプットキャプチャ)	同期動作 ノイズフィルタ A/D変換開始要求/A/D変換ディレイド	MTU0,1,2,3,4,6,7,9 *A/D変換開始要求はMTU4,7のみ
リセット同期PWMモード	TGR(アウトプットコンペア) TOCR2(出力制御)	同期動作 ノイズフィルタ A/D変換開始要求/A/D変換ディレイド AC同期モータ駆動	MTU3,4,6,7 *A/D変換開始要求はMTU4,7のみ *AC同期モータ駆動はMTU3,4のみ
相補PWMモード	TGR*(アウトプットコンペア/ インプットキャプチャ) *ダブルバッファ対応 TOCR2(出力制御) TCDR(周期レジスタ)	同期動作 ノイズフィルタ A/D変換開始要求/A/D変換ディレイド AC同期モータ駆動 割り込み間引き	MTU3,4,6,7 *A/D変換開始要求はMTU4,7のみ *AC同期モータ駆動はMTU3,4のみ
位相計数モード	TGR(アウトプットコンペア/ インプットキャプチャ)	ノイズフィルタ	MTU1,2
位相計数モード (カスケード動作)	TGR(アウトプットコンペア/ インプットキャプチャ)	ノイズフィルタ	MTU1,2

ノーマルモード、PWMモード1、PWMモード2の出力波形

ノーマルモード

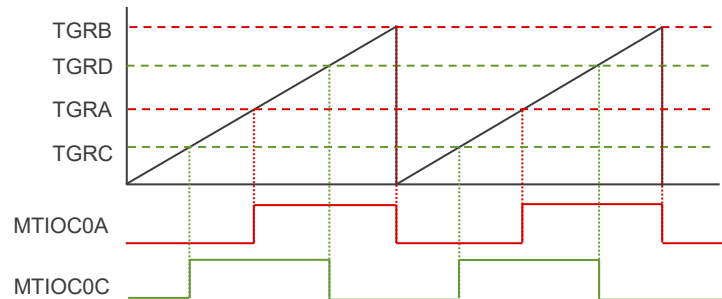
- ・ TCNTクリア条件：H'FFFFまたはTGRレジスタ
- ・ PWM出力波形：TGRに対応した波形を出力
- ・ Duty：50% duty波形のみ



例：MTU0 ノーマルモード動作例(TGRB, TBRCはトグル動作)

PWMモード1

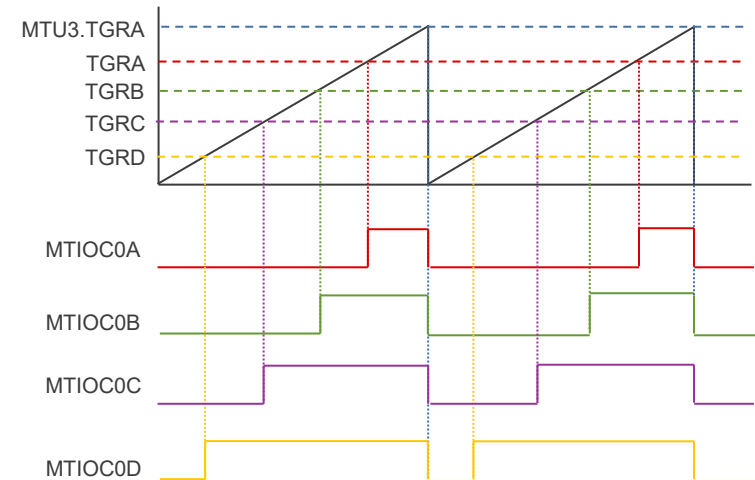
- ・ TCNTクリア条件：いずれかのTGRレジスタ
- ・ PWM出力波形：TGRAとTGRB、TGRCとTGRDの組み合わせで波形出力
- ・ Duty：0%~100%までのPWM波形出力可能



例：MTU0 PWMモード1動作例(TGRBでカウンタクリア)

PWMモード2

- ・ TCNTクリア条件：いずれかのTGRレジスタまたは他モードのTGRレジスタ
- ・ PWM出力波形：クリア条件はTCNTカウンタクリア固定、周期は各TGRレジスタ
- ・ Duty：0%~100%までのPWM波形出力可能

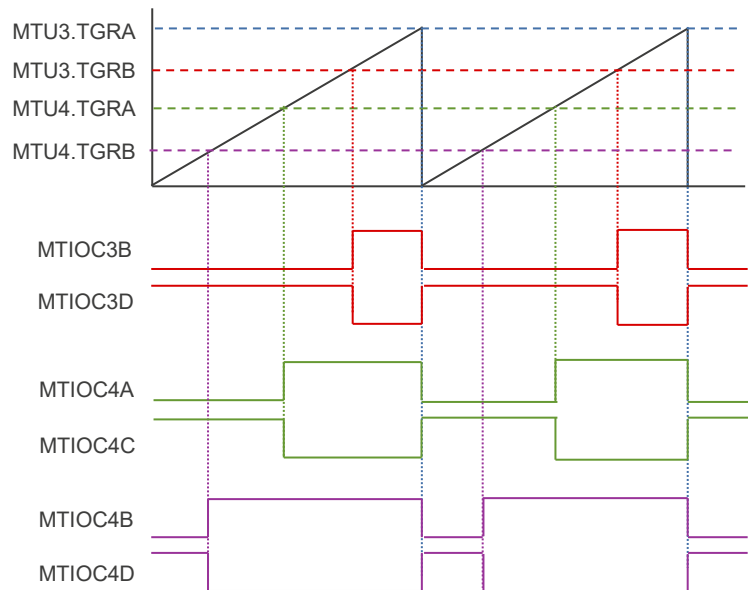


例：MTU0 PWMモード2動作例(TCNTカウンタクリアはモード1で動作のMTU3.TGRA)

リセット同期PWMモード、相補PWMモードの出力波形

リセット同期PWMモード

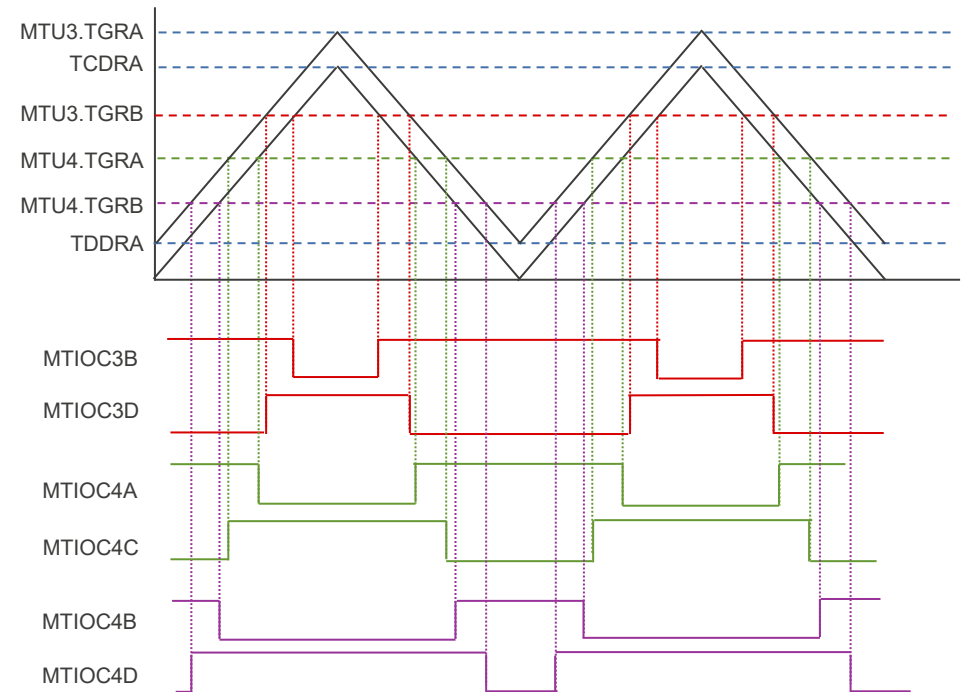
- TCNTクリア条件 : MTU3(6).TGRAレジスタ
- PWM出力波形 : MTU3(6).TGRB、MTU4(7).TGRA、MTU4(7).TGRBのコンペアマッチとカウンタクリアでトグル出力(正相、逆相の二相出力)
- Duty : 0%~100%までのPWM波形出力可能
- デッドタイム : なし



例 : MTU3/4 リセット同期PWMモード動作例

相補PWMモード

- TCNTクリア条件 : MTU3(6).TGRAレジスタ
- PWM出力波形 : MTU3(6).TGRB、MTU4(7).TGRA、MTU4(7).TGRBのコンペアマッチでトグル出力(正相、逆相の二相出力)
- Duty : 0%~100%までのPWM波形出力可能
- デッドタイム : TDDRAに設定した値分のデッドタイムを保持

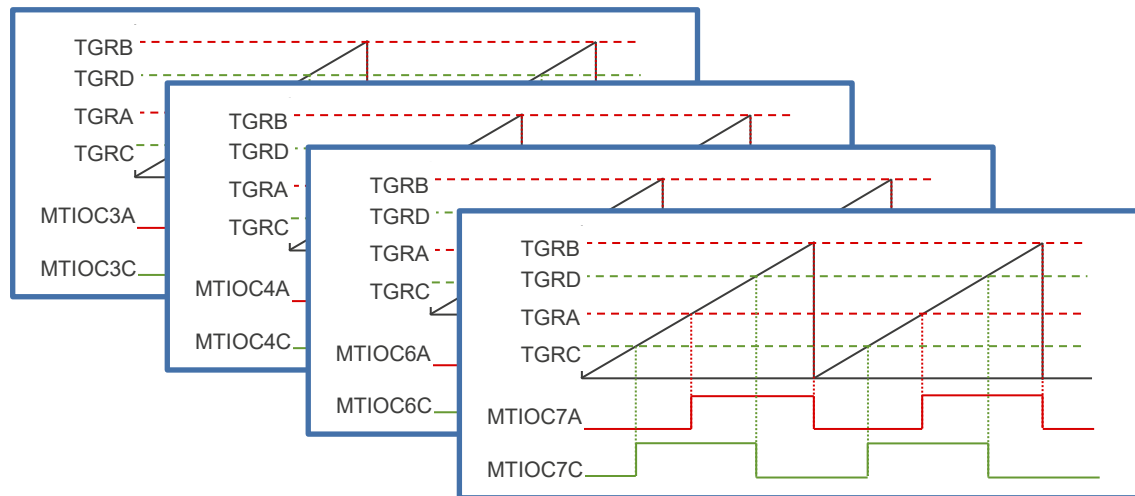


例 : MTU3/4 相補PWMモード動作例

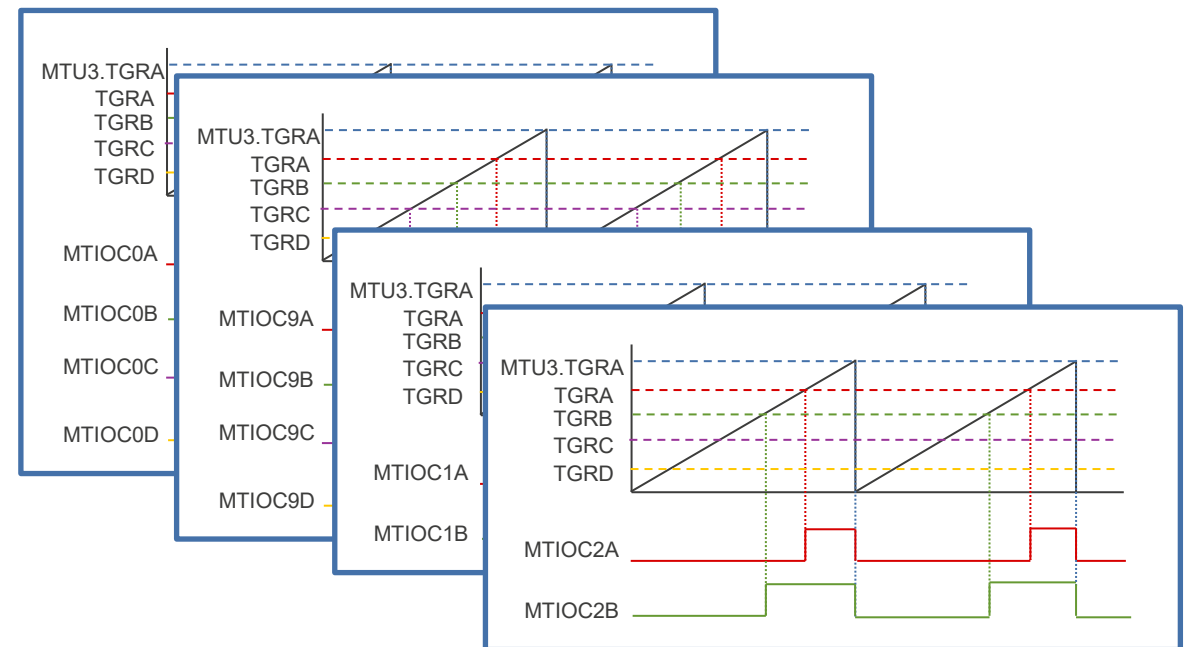
最大PWM波形出力本数

最大PWM波形出力本数はPWMモード1で14本*、PWMモード2で最大12本*となります。
PWMモード1とPWMモード2を混在させると、最大20本*の出力が可能となります。

*製品により変わります



PWMモード1(MTU3、MTU4、MTU6、MTU7)計8本



PWMモード2 (MTU0~MTU2、MTU9)計12本

*TCNTカウントクリアはモード1で動作のMTUn.TGRA)

n = 3,4,6,7

ノーマルモード、PWMモード1/2 関連レジスタ

: 設定が必要なレジスタ

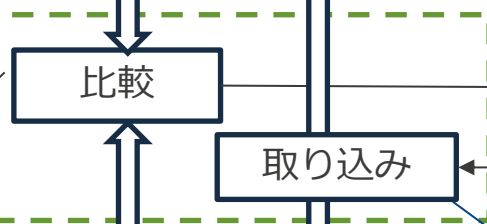
クロックソース

PCLK MTCLKA
 PCLK/2 MTCLKB
 PCLK/4 MTCLKC
 PCLK/8 MTCLKD
 PCLK/16
 PCLK/32
 PCLK/64
 PCLK/256
 PCLK/1024

タイマコントロール
 動作モード設定 : **TMDR1**
 レジスタアクセス設定 : **TRWERx**
 カウントスタート : **TSTR**

周期設定 : **TCR, TCR2**
 カウント方向設定 : **TSR**

カウンタ : **TCNT**



コンペア/キャプチャ : **TGR**

動作モードやTCNTカウンタを制御します

割り込み設定 : **TIER**

- 割り込み TGIX
- ・コンペアマッチ割り込み
 - ・入力キャプチャ割り込み
 - ・オーバフロー割り込み
 - ・アンダフロー割り込み

端子設定 : **TIOR**
 出力 : **TOER**
 入力 : **TICCR**



TGRの設定値とTCNTカウンタが一致したタイミングで、出力処理します。

インพุットキャプチャ要因を検出したタイミングでTCNTの値がキャプチャされ、TGRに格納されます

リセット同期PWMモード 関連レジスタ

: 設定が必要なレジスタ

クロックソース
 PCLK MTCLKA
 PCLK/2 MTCLKB
 PCLK/4 MTCLKC
 PCLK/8 MTCLKD
 PCLK/16
 PCLK/32
 PCLK/64
 PCLK/256
 PCLK/1024

タイマコントロール
 動作モード設定: **TMDR1**
 レジスタアクセス設定: **TRWERx**
 カウントスタート: **TSTR**

周期設定: **TCR, TCR2**

カウンタ: **TCNT**

比較

コンペア/キャプチャ: **TGR**

動作モードやTCNTカウンタを制御します

割り込み設定: **TIER**

割り込み TGIx
 ・コンペアマッチ割り込み
 ・オーバフロー割り込み
 ・アンダフロー割り込み

端子設定: **TIOR***
 出力: **TOER, TOCR1/2**

○ 出力端子 MTIOCxx

TGRの設定値とTCNTカウンタが一致した
 タイミングで、出力処理します。

*TIOR'00を設定してください

相補PWMモード 関連レジスタ

: 設定が必要なレジスタ

クロックソース
 PCLK MTCLKA
 PCLK/2 MTCLKB
 PCLK/4 MTCLKC
 PCLK/8 MTCLKD
 PCLK/16
 PCLK/32
 PCLK/64
 PCLK/256
 PCLK/1024

タイマコントロール
 動作モード設定 : **TMDR1**
 レジスタアクセス設定 : **TRWERx**
 カウントスタート : **TSTR**

周期設定 : **TCR, TCR2, TCDR**
 コンペアマッチクリア設定 : **TWCR**
 デッドタイム設定 : **TDDR, TDER**

カウンタ : **TCNT, TCNTS**

比較

コンペア/キャプチャ : **TGR**

動作モードやTCNTカウンタを制御します

割り込み設定 : **TIER**

割り込み TGix
 ・コンペアマッチ割り込み
 ・オーバフロー割り込み
 ・アンダフロー割り込み

端子設定 : **TIOR***
 出力 : **TOER, TOCR1/2**

○ 出力端子 MTIOCxx

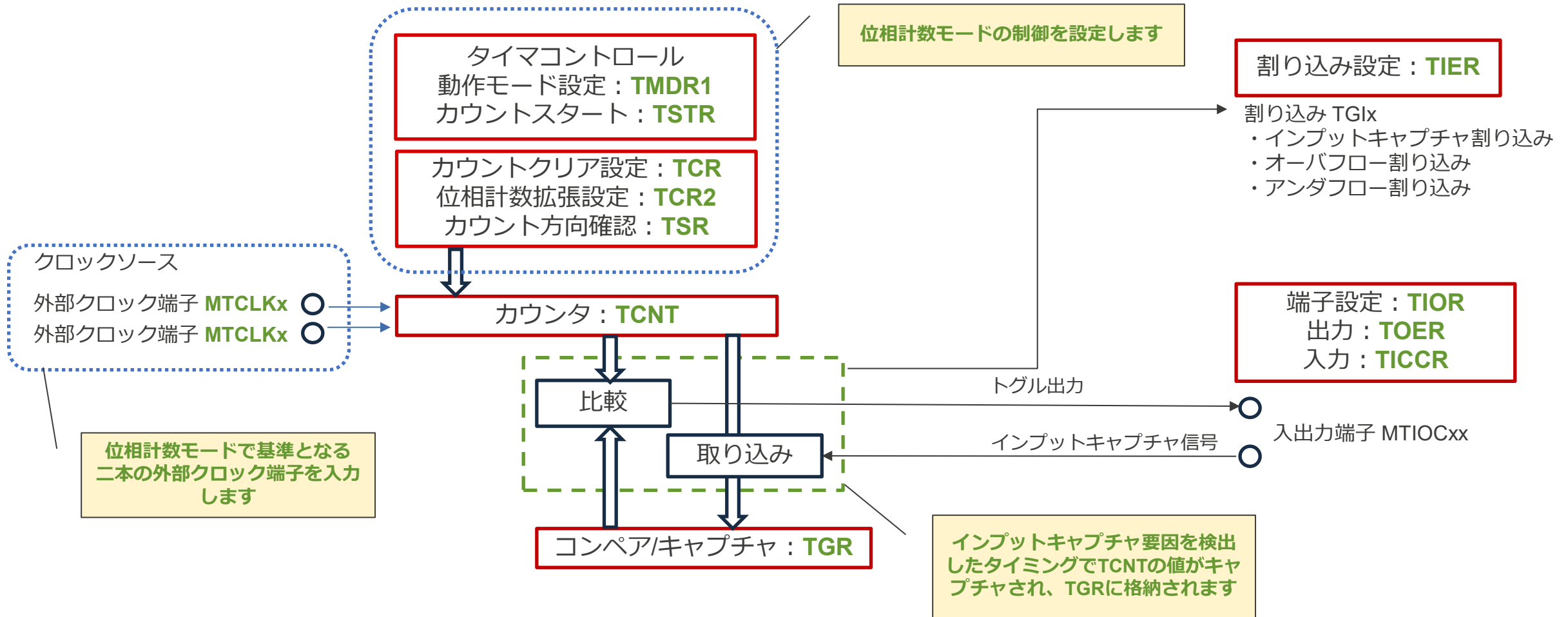
TGRの設定値とTCNTカウンタが一致したタイミングで、出力処理します。

PWM波形出力

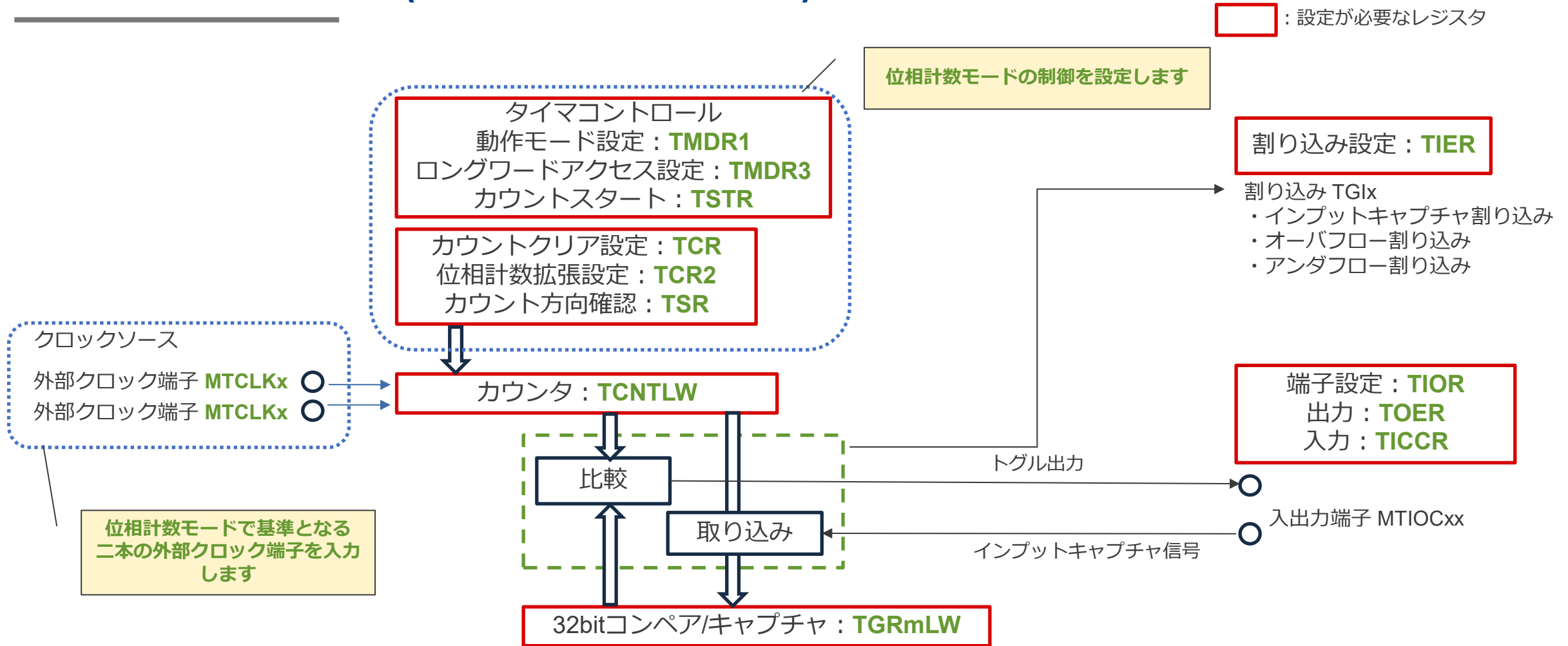
*TIOR'00を設定してください

位相計数モード 関連レジスタ

 : 設定が必要なレジスタ



位相計数モード(カスケード動作) 関連レジスタ



*TCR, TCR2, TMDR, TIORレジスタはMTU1のみ有効

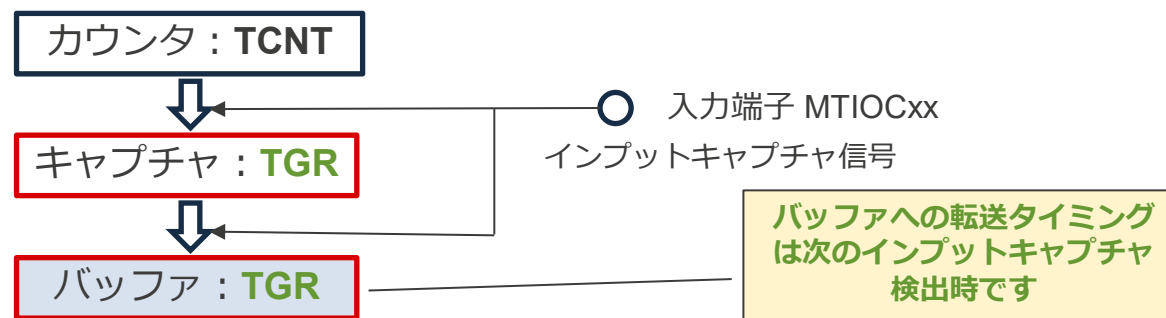
バッファ動作 関連レジスタ (1/2)

: 追加で設定が必要なレジスタ
 : バッファレジスタ

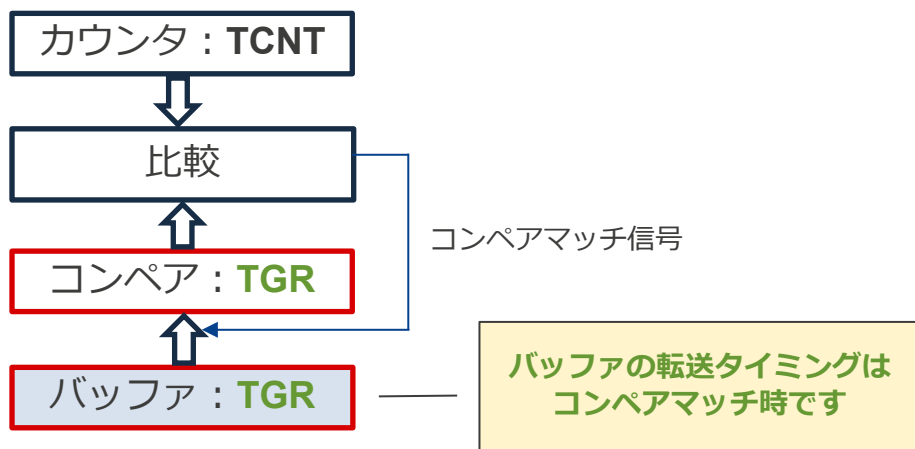
■ ノーマルモード、PWMモード 1/2、位相計数モード共通設定

タイマコントロール
バッファ設定 : **TMDR1**

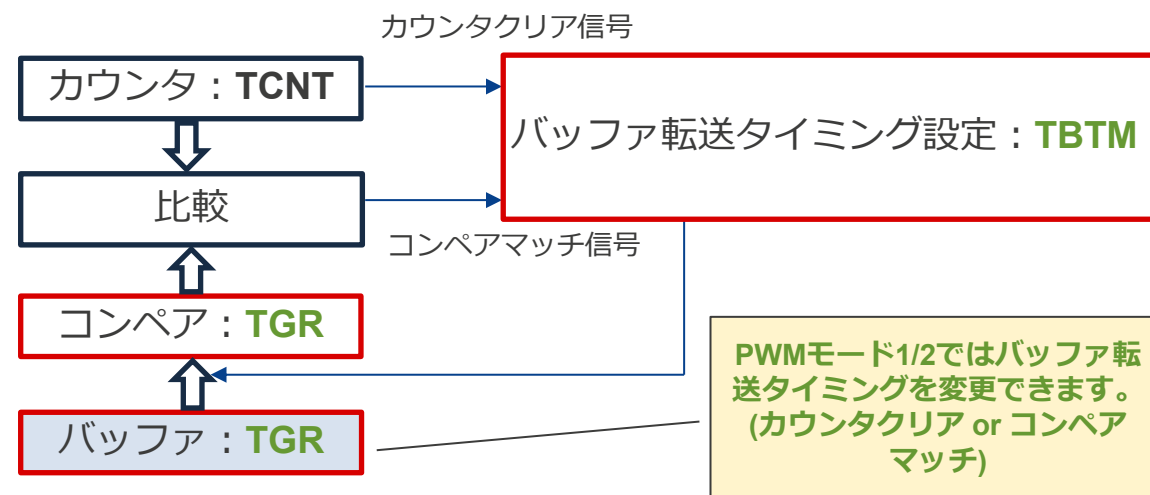
■ インพุットキャプチャのバッファ動作 (ノーマルモード、PWMモード1/2、位相計数モード)



■ アウツプツコンペアのバッファ動作 (ノーマルモード、位相計数モード)



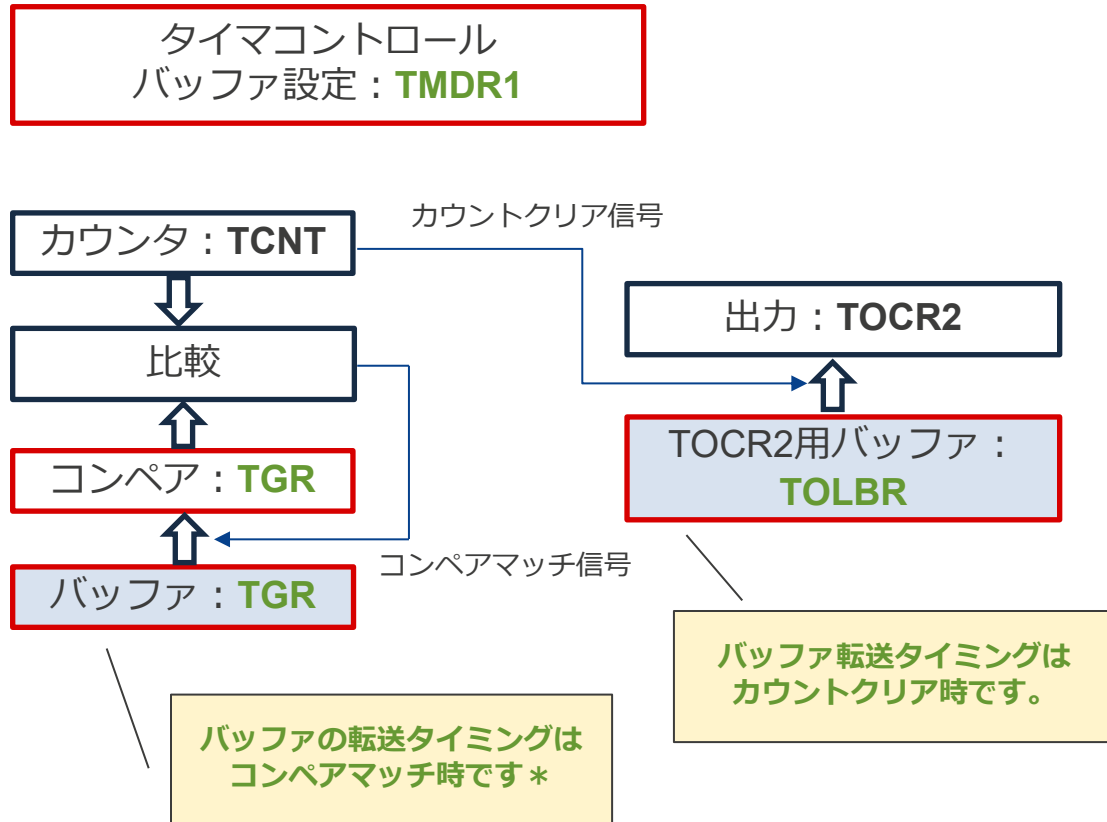
■ アウツプツコンペアのバッファ動作(PWMモード1/2)



バッファ動作 関連レジスタ (2/2)

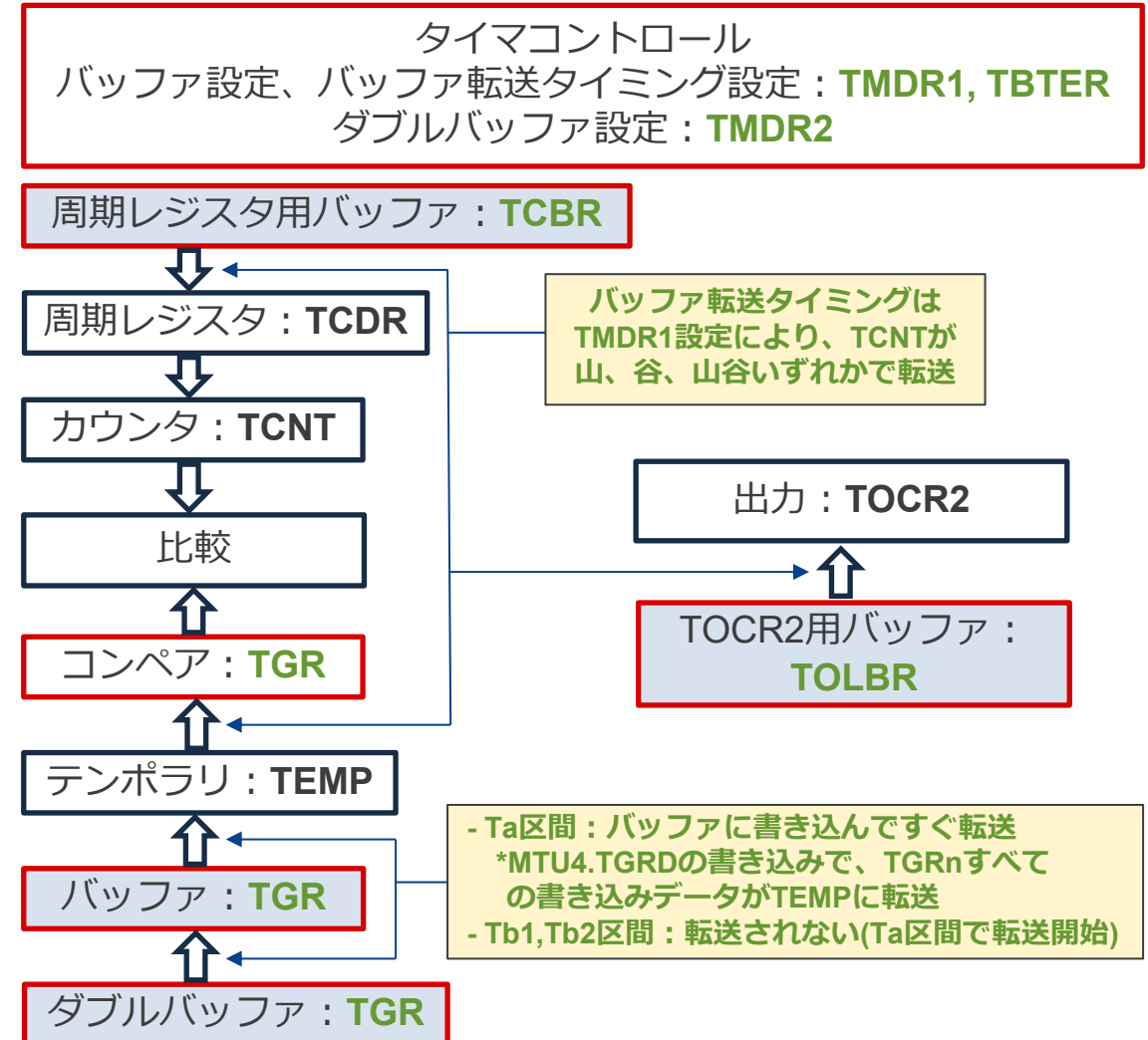
□ : 追加で設定が必要なレジスタ
 ■ : バッファレジスタ

■リセット同期PWMモードのバッファ動作



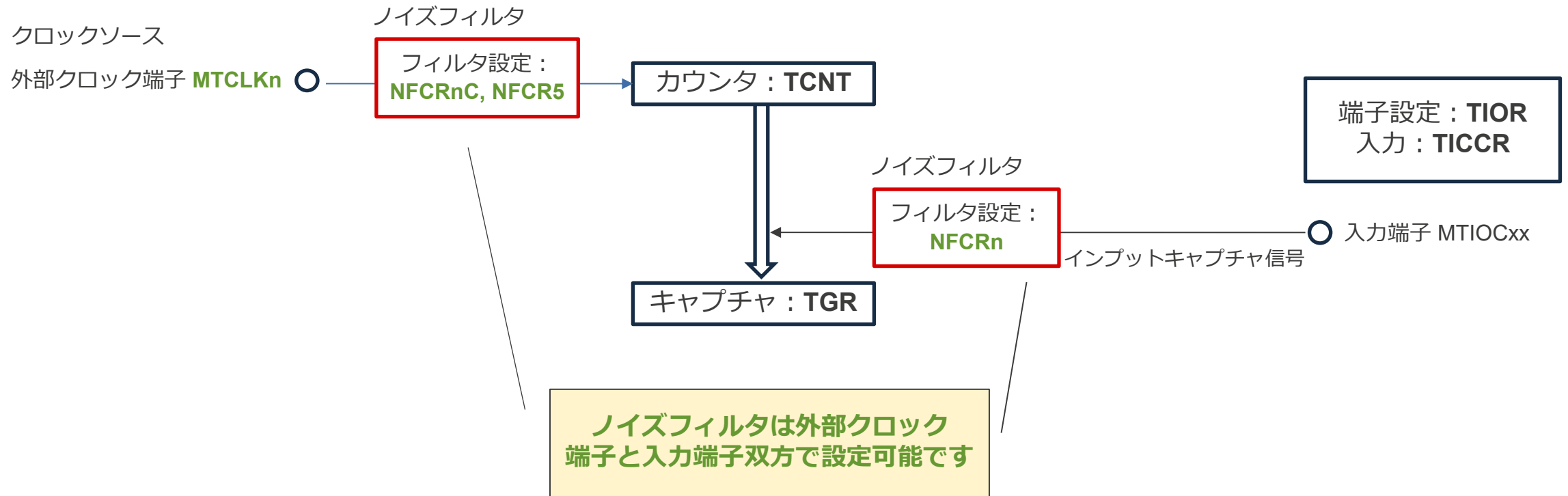
*リセット同期PWMモードの場合、周期用およびデューティ用のバッファはともに周期レジスタのコンペアタイミングでのみバッファ転送されます。

■相補PWMモードのバッファ動作



ノイズフィルタ 関連レジスタ(全モード共通)

: 設定が必要なレジスタ



同期動作 関連レジスタ

(ノーマルモード、PWMモード1/2、リセット同期PWMモード)

 : 設定が必要なレジスタ

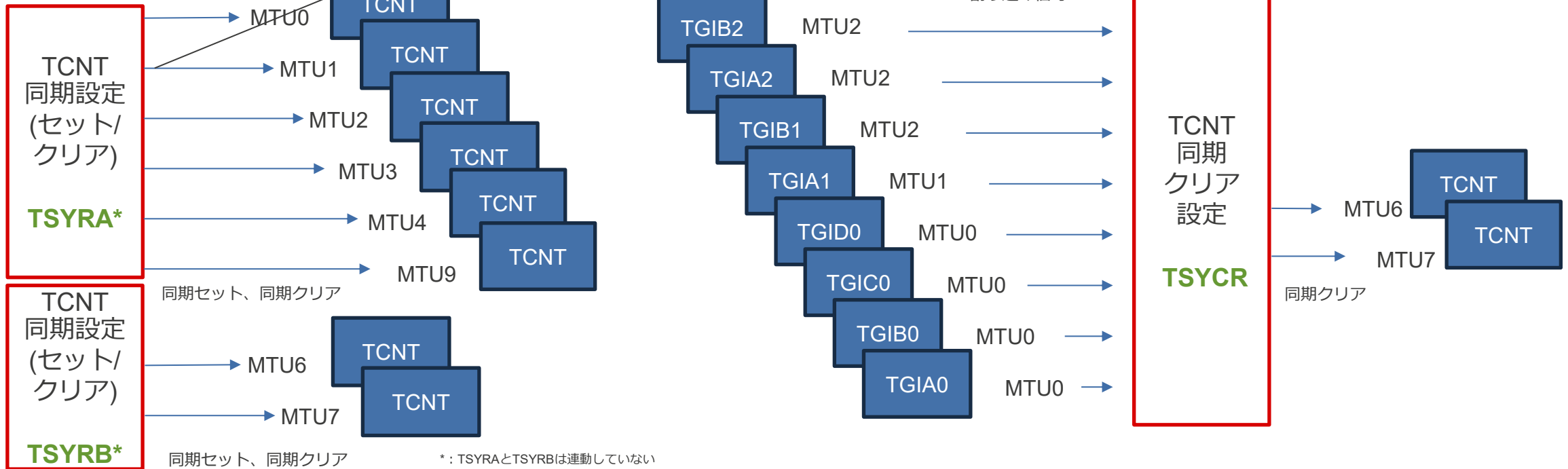
同期スタート

タイマコントロール
シンクロスタート設定 : **TCSYSTR**

同期するチャンネルを選択します。選択された
TCNTカウントを同期セット/クリアできます。

MTU6,MTU7は同期クリアのみ設定可能です。
同期起因のMTU0~2の割り込みを選択します

同期セット/クリア



同期動作 関連レジスタ

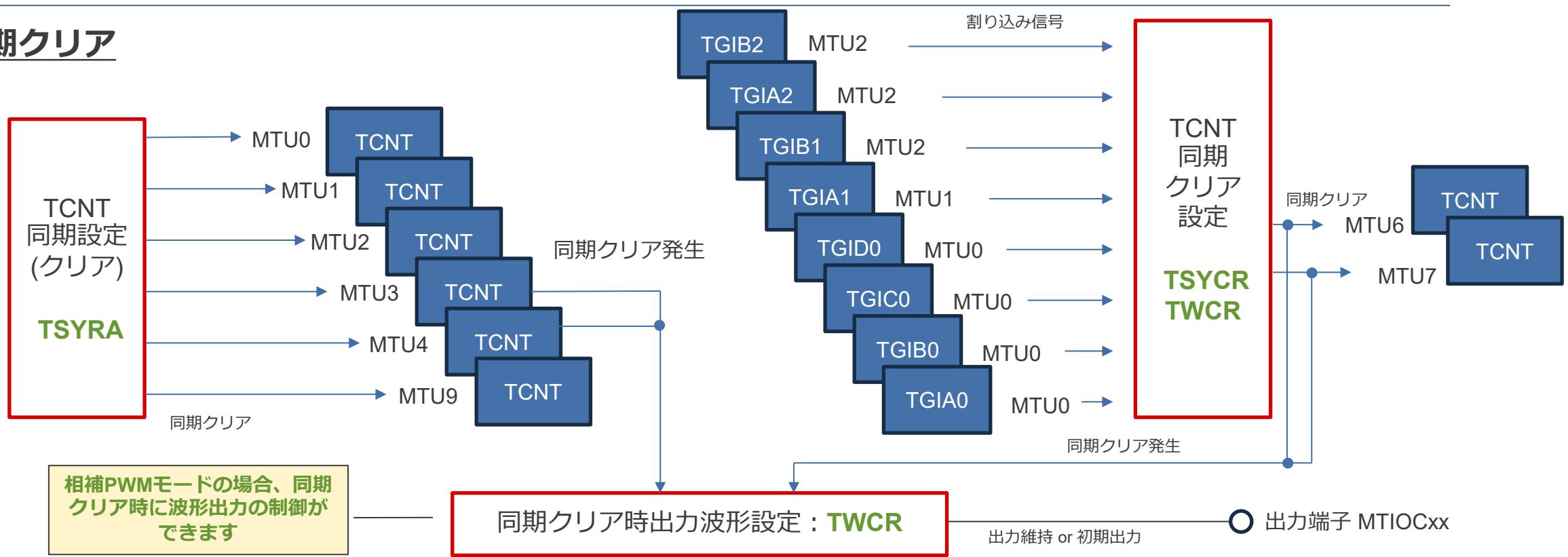
(相補PWMモード)

 : 設定が必要なレジスタ

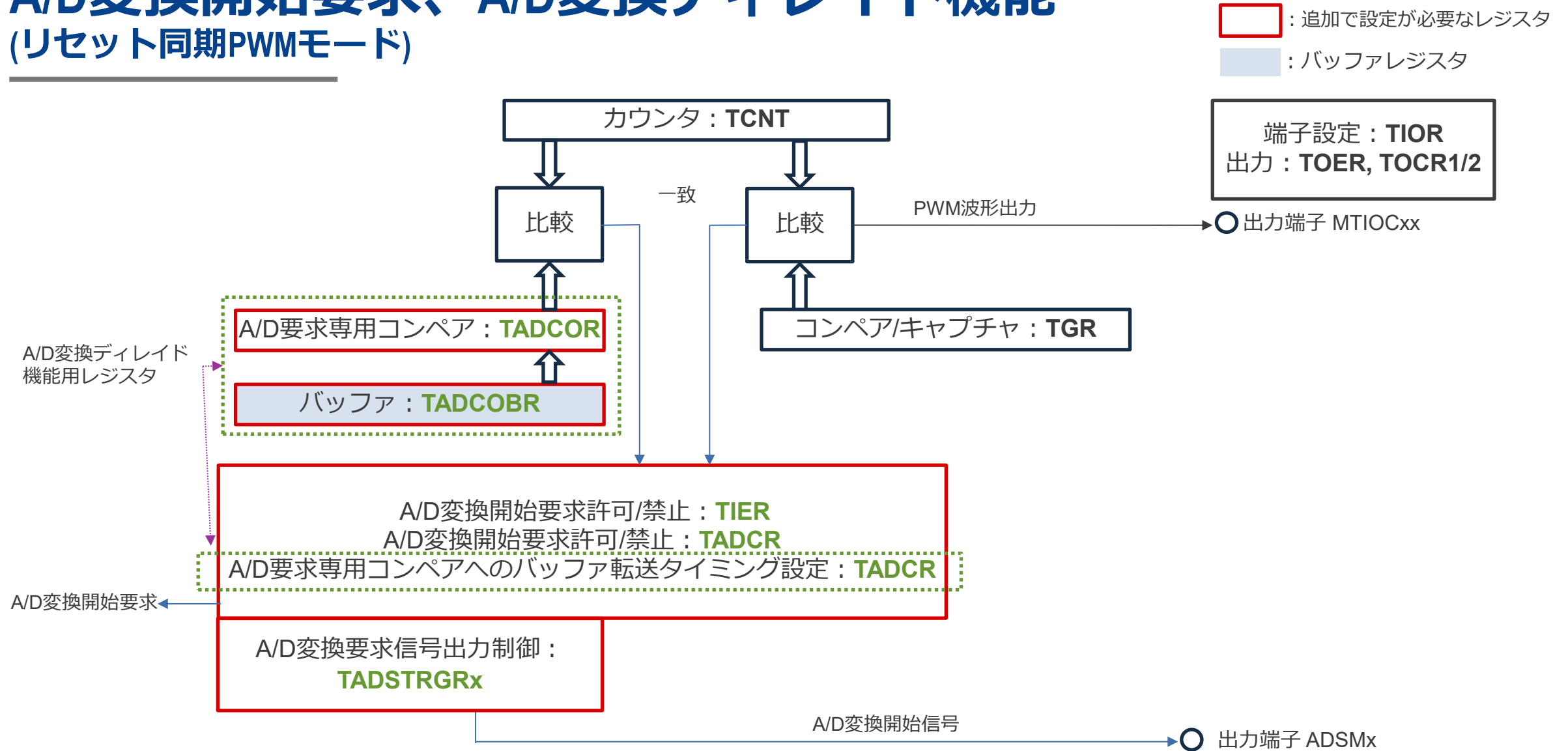
同期スタート

タイマコントロール
 シンクロスタート設定 : **TCSYSTR**

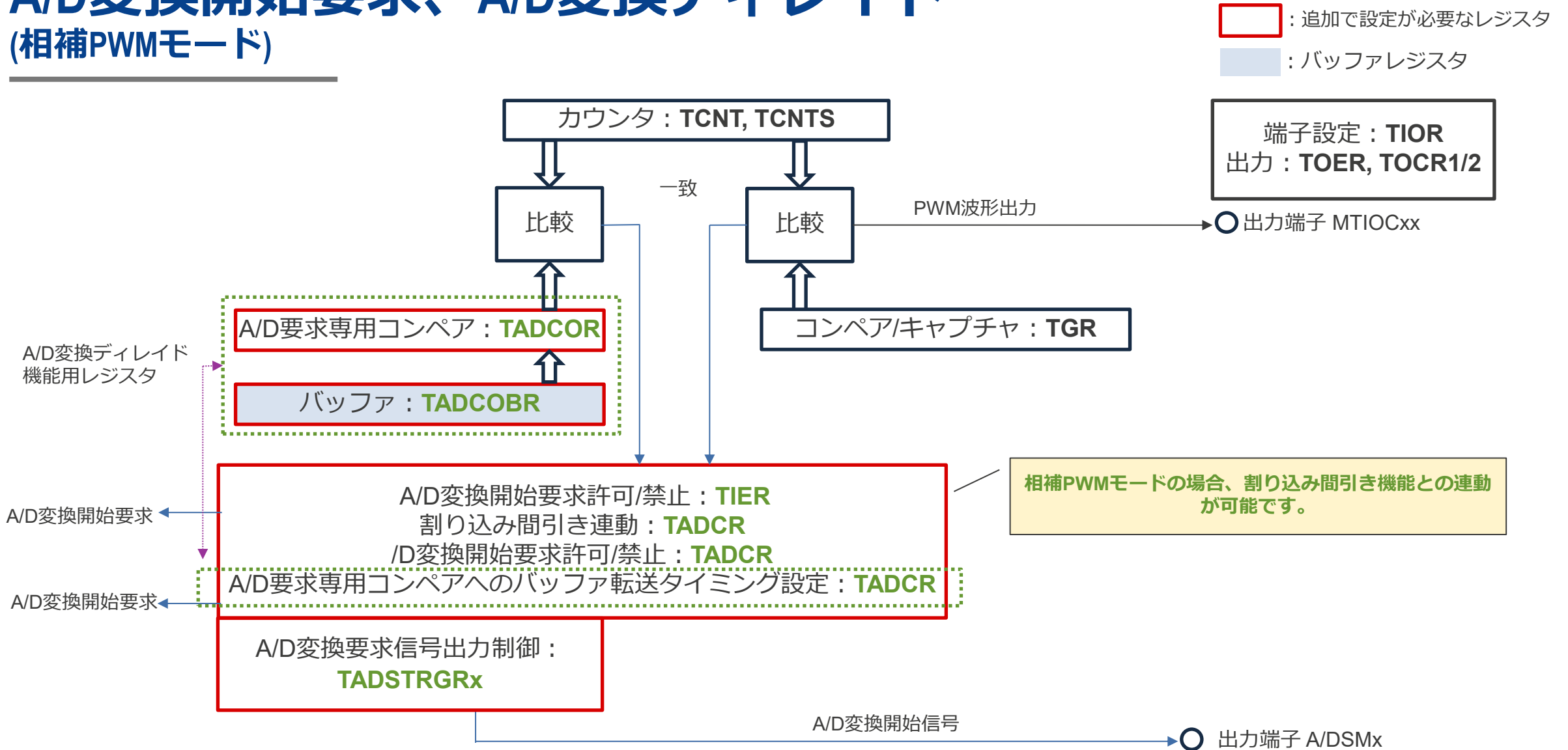
同期クリア



A/D変換開始要求、A/D変換ディレイド機能 (リセット同期PWMモード)

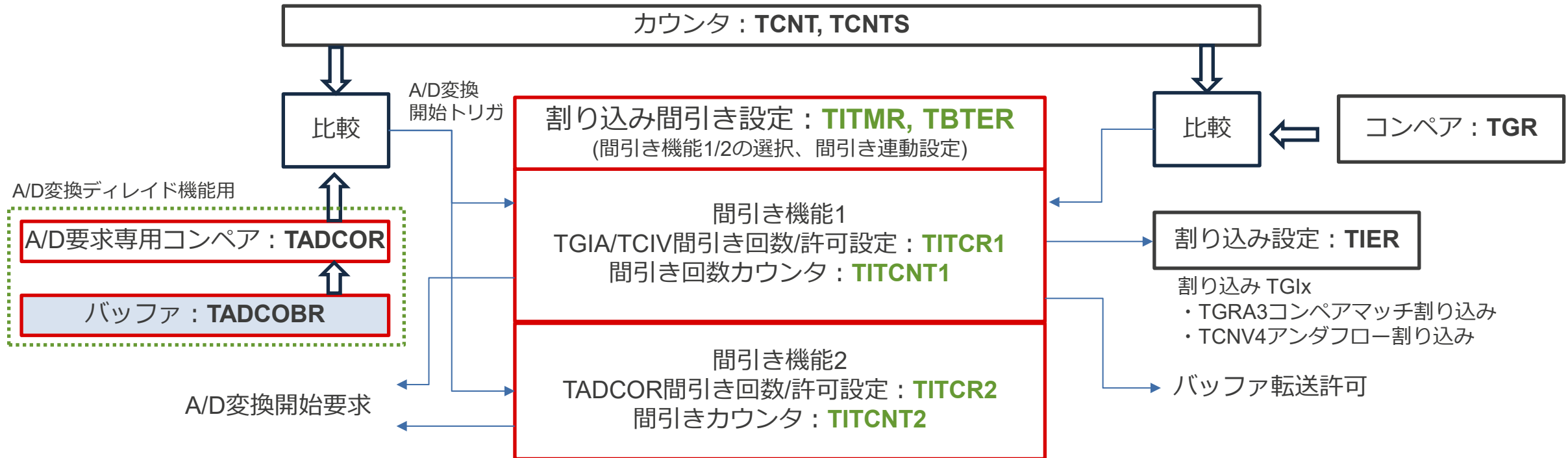


A/D変換開始要求、A/D変換ディレイド (相補PWMモード)



割り込み間引き (相補PWMモードのみ)

: 追加で設定が必要なレジスタ
 : バッファレジスタ



割り込み間引き1 : TGI(カウンタの山)、TCIV(カウンタの谷)の割り込みを間引くことが可能。
 また、バッファ動作や、AD変換ディレイド機能と連動し、バッファ転送期間やAD変換開始要求タイミングを限定することが可能。
 (TGI、TCIVの割り込み可能期間のみバッファ転送やAD変換開始要求可能)

割り込み間引き2 : ADディレイド機能によるAD変換トリガを間引きカウンタでカウントし、カウントが0になったらA/Dモジュールに開始要求発行

MTUを使用したアプリケーションノート一覧

MTUを使用したアプリケーションノートを多数準備しています。より詳細な使用方法は下記をご参照ください

- RXファミリ MTU3/GPTWを用いたPWM出力方法 [R01AN5995](#)
- RXファミリ MTU3/GPTWを用いた相補PWMの0%、100%近傍出力動作例 [R01AN6539](#)
- RXファミリ MTU3/GPTWを用いたA/D変換開始要求ディレイド機能使用例 [R01AN6643](#)
- RXファミリ MTU2/MTU3を用いたパルス周期測定機能使用例 [R01AN6644](#)
- RXファミリ MTU2/MTU3を用いたパルス幅測定機能使用例 [R01AN6748](#)
- RXファミリ MTU3/GPTWを用いた位相計数モード使用例 [R01AN6387](#)
- RXファミリ MTU3/GPTWを用いた同期動作 [R01AN6282](#)

POE3

ポートアウトプットイネーブル(POE)機能仕様一覧

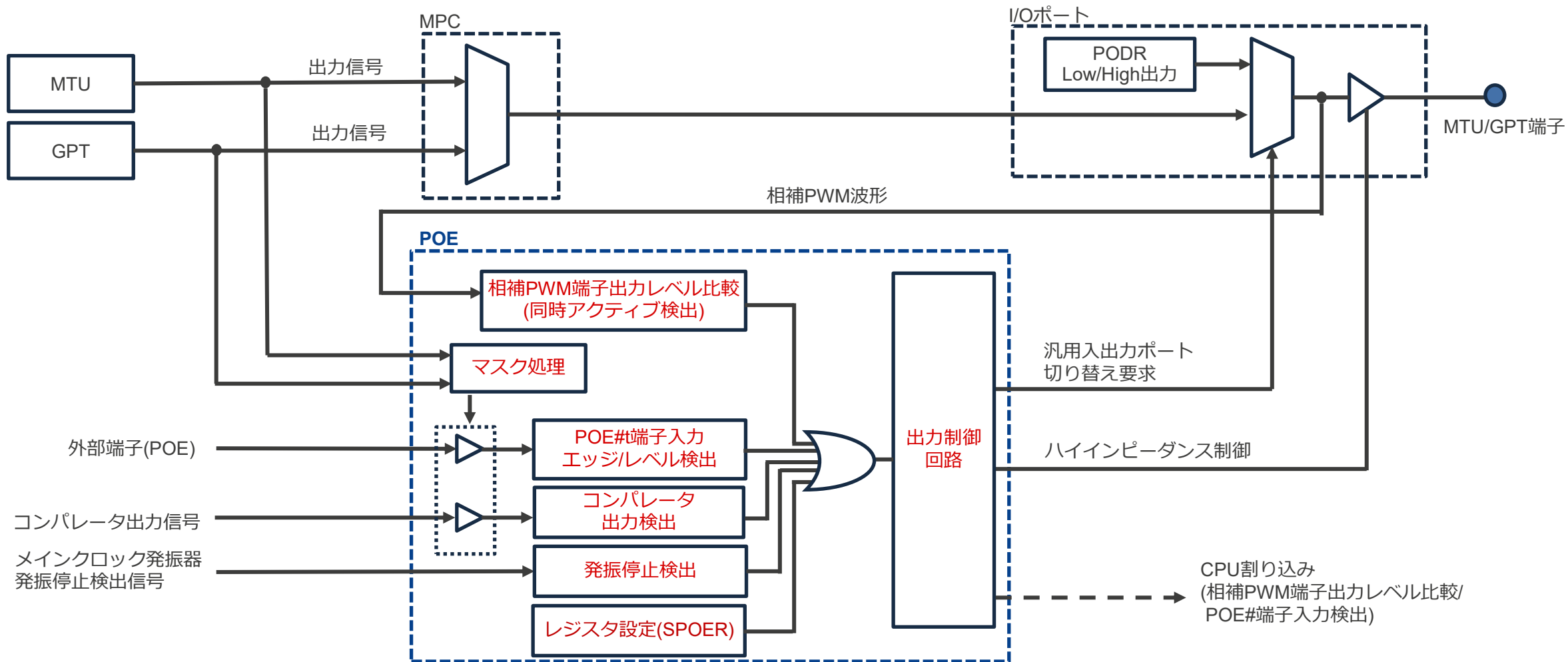
POE機能は、出力停止条件を検知すると、対象端子をハイインピーダンスまたは汎用入出力ポートに自動で切り替え、PWM出力を停止する機能です。POE機能の仕様一覧を以下に示します。(製品によって異なる場合があります。詳細はハードウェアマニュアルをご参照ください)

POEの仕様

出力停止時条件を検知時の出力停止対象端子の状態		ハイインピーダンスまたは汎用入出力ポートに切り替え
出力停止条件	条件1：相補PWM出力端子の出力レベル比較	正相および逆相の出力レベルを監視し、短絡を停止条件に設定可能(対象端子はP.113参照)
	条件2：POE#端子入力の変化	POE0,4,8~12#端子のトリガ入力(立ち下がりエッジまたはLow検出)を停止条件として設定可能 * 入力信号の論理反転機能により立ち上がりエッジまたはHigh検出も可能
	条件3：コンパレータ出力検出	コンパレータ機能からの出力信号を停止条件として設定可能
	条件4：発振停止検出(メインクロック)	メインクロックの発振停止検出を停止条件として設定可能
	条件5：レジスタ設定(SPOER)	ソフトウェアによる停止条件を発行可能
出力停止対象端子		P.60参照
出力停止条件ごとの出力停止対象端子	相補PWM出力端子の出力レベル比較(出力端子の短絡)時	出力レベル比較結果がMTU端子の場合：MTUの対象端子すべて 出力レベル比較結果がGPT端子の場合：GPTの対象端子すべて
	POE#端子入力の変化時	全てのMTUおよびGPT対象端子から選択
	コンパレータ出力検出時	全てのMTUおよびGPT対象端子から選択
	発振停止検出時(メインクロック)	全てのMTUおよびGPT対象端子から選択
	レジスタ設定(SPOER)時	全てのMTUおよびGPT対象端子から選択
出力条件(POE#端子入力変化、コンパレータ出力検出)のマスク処理		MTU/GPT出力端子状態により、POE#入力端子変化、コンパレータ出力検出の条件をマスク可能 対象端子はP.60を参照
割り込み発生要因		POE#端子入力時 相補端子の出力レベル比較時

POE端子検出機能システムブロック図

POE端子検出機能全体のシステムブロック図を下記に示します。

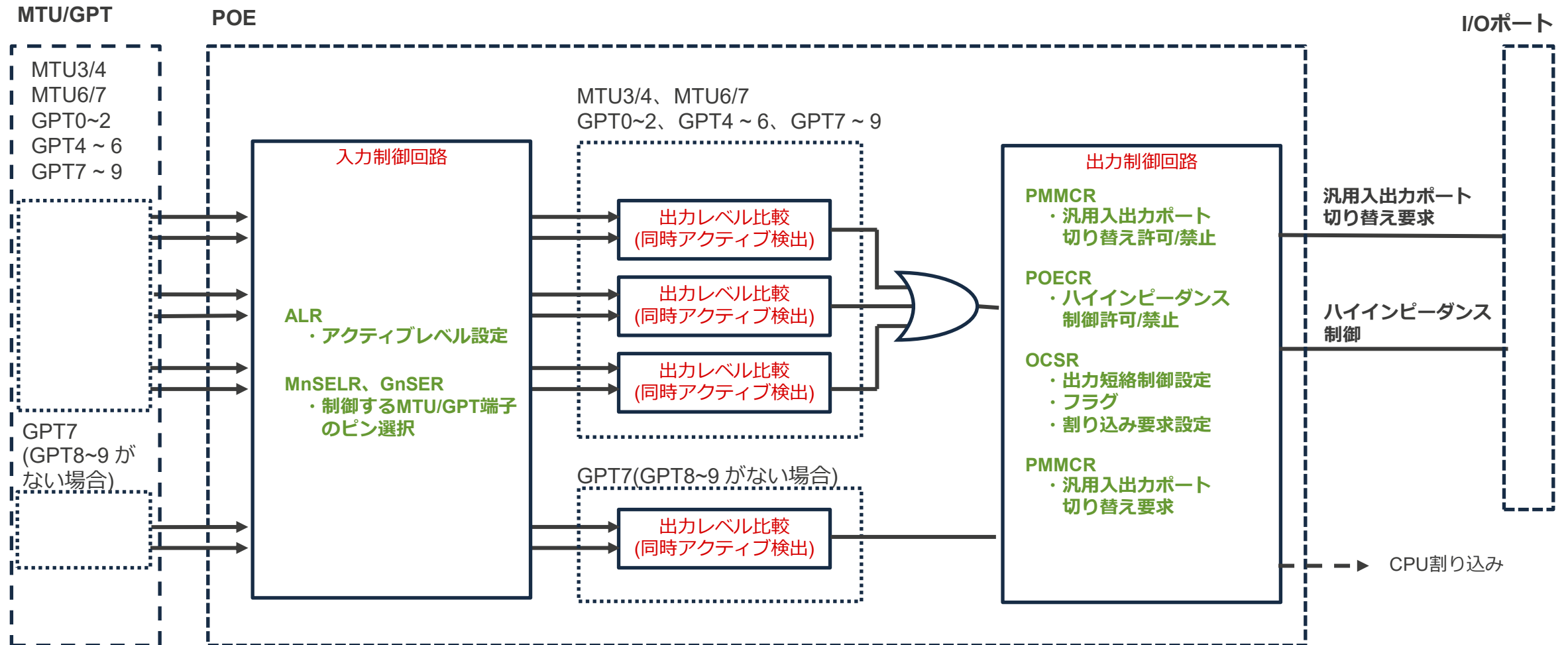


マスク処理、出力レベル比較、停止制御対象端子まとめ

MTU機能端子		マスク処理端子	出力レベル比較端子	停止制御対象端子	MTU機能端子		マスク処理端子	出力レベル比較端子	停止制御対象端子
MTU0	MTIOC0A	○		○	MTU9	MTIOC9A	○		○
	MTIOC0B	○		○		MTIOC9B	○		○
	MTIOC0C	○		○		MTIOC9C	○		○
	MTIOC0D	○		○		MTIOC9D	○		○
MTU1	MTIOC1A	○							
	MTIOC1B	○							
MTU2	MTIOC2A	○			GPTW機能端子				
	MTIOC2B	○			GPTW0	GTIOC0A	○	○	○
MTU3	MTIOC3A	○			GPTW1	GTIOC0B	○	○	○
	MTIOC3B	○	○	○		GTIOC1A	○	○	○
	MTIOC3C	○		○	GPTW2	GTIOC1B	○	○	○
	MTIOC3D	○	○	○		GTIOC2A	○	○	○
MTU4	MTIOC4A	○	○	○	GPTW3	GTIOC2B	○	○	○
	MTIOC4B	○	○	○		GTIOC3A	○		○
	MTIOC4C	○	○	○	GPTW4	GTIOC3B	○		○
	MTIOC4D	○	○	○		GTIOC4A	○	○	○
MTU6	MTIOC6A	○			GPTW5	GTIOC4B	○	○	○
	MTIOC6B	○	○	○		GTIOC5A	○	○	○
	MTIOC6C	○			GPTW6	GTIOC5B	○	○	○
	MTIOC6D	○	○	○		GTIOC6A	○	○	○
MTU7	MTIOC7A	○	○	○	GPTW7	GTIOC6B	○	○	○
	MTIOC7B	○	○	○		GTIOC7A	○	○	○
	MTIOC7C	○	○	○	GPTW8	GTIOC7B	○	○	○
	MTIOC7D	○	○	○		GTIOC8A	○	○	○
					GPTW9	GTIOC8B	○	○	○
						GTIOC9A	○	○	○
						GTIOC9B	○	○	○

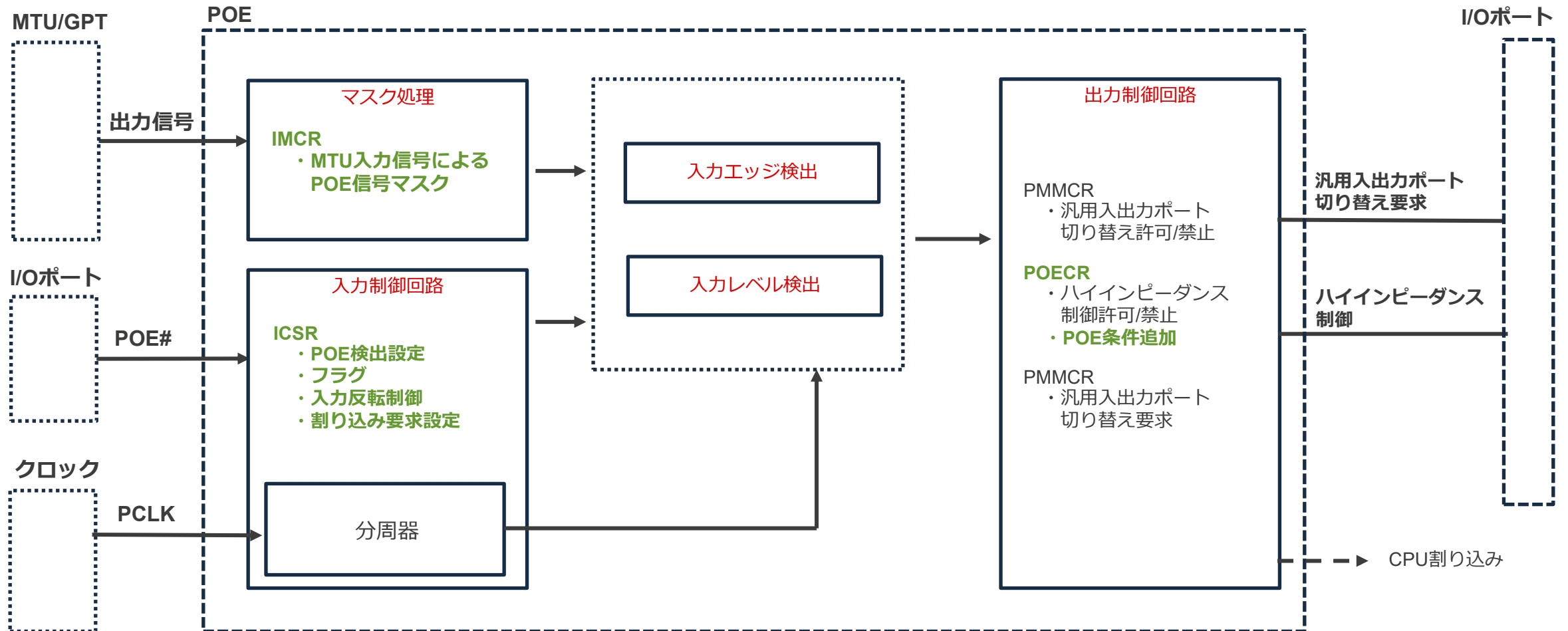
相補PWM出力端子の出力レベル比較を停止条件とする場合

相補PWM出力端子の出力レベル比較を停止条件とする場合の制御ブロックと設定が必要なレジスタを下記に示します。



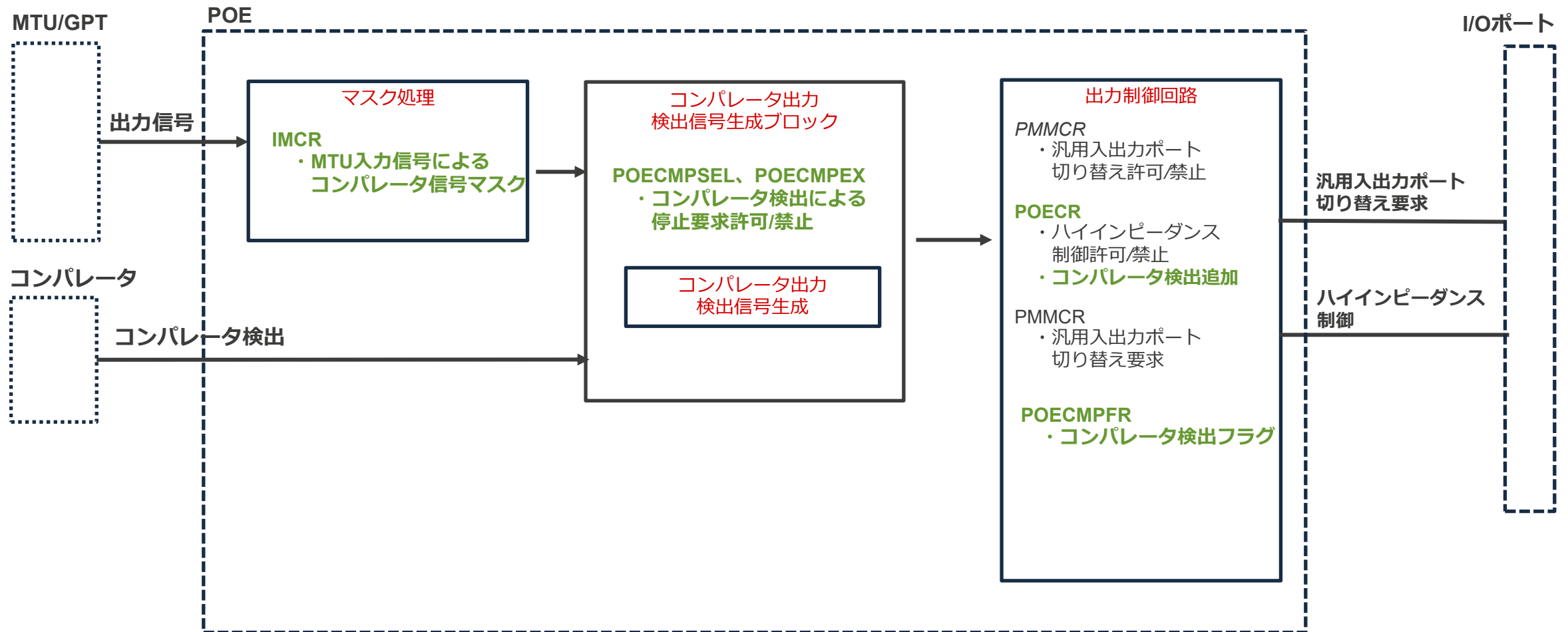
POE#端子入力を停止条件とする場合

POE#端子入力を停止条件とする場合の制御ブロックと設定が必要なレジスタを下記に示します。



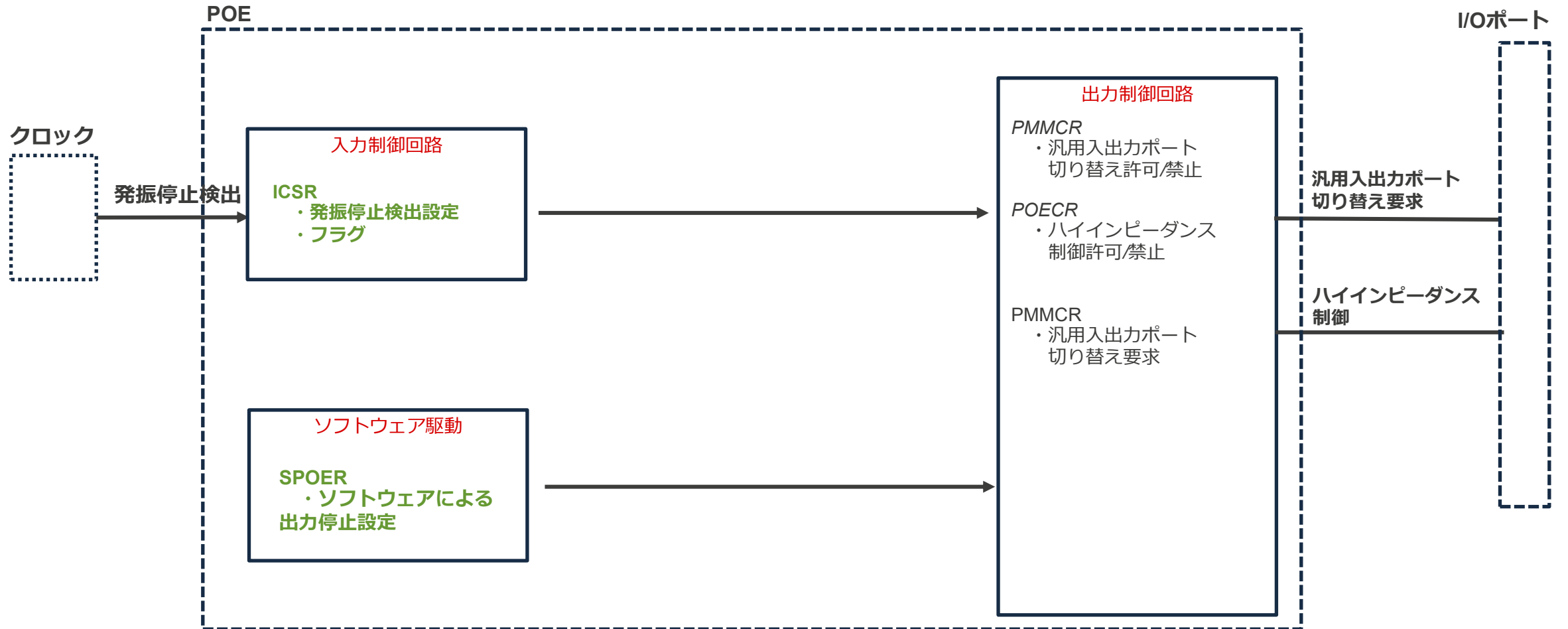
コンパレータ出力信号を停止条件とする場合

コンパレータ出力信号を停止条件とする場合の制御ブロックと設定が必要なレジスタを下記に示します。



発振停止検出信号、ソフトウェアによる制御の場合

発振停止検出信号またはソフトウェアによる停止条件とする場合の制御ブロックと設定が必要なレジスタを下記に示します。



POE出力の解除条件

■ 入力レベル検出で出力が停止した端子は以下のいずれかで解除されます

- ・リセットで初期状態に戻す
- ・ICSRn.POExF フラグをクリアする

*ICSRn.POExM[3:0] ビットでLow サンプリングに設定している場合は、POEn# 端子からHigh を入力してHigh を検出した後でないと、フラグに対して“0” を書いても無効となりフラグは“0” になりません。

■ 出力短絡検出で出力が停止した端子は以下のいずれかで解除されます

- ・リセットで初期状態に戻す
- ・OCSRn.OSFn フラグを“0” にする

* 端子から非アクティブレベルを出力するようにした後でないと、フラグに対して“0” を書いても無効となりフラグは“0” になりません。非アクティブレベル出力は、MTU の場合はカウント動作停止後に初期出力状態にすることで、GPTW の場合は「動作中の異常などによる端子の初期化」の手順で行うことができます。

■ コンパレータ出力検出で出力が停止した端子は以下のいずれかで解除されます

- ・リセットで初期状態に戻す
- ・POECMPFR.CjFLAG フラグ(j = 0 ~ 5) を“0” にする

* POECMPFR.CjFLAG フラグを“0” にする場合は、コンパレータ出力検出を行ったアナログ入力信号が適正値に戻ったことをA/D 変換実施等で確認した後に実施してください。アナログ入力信号が適正値に戻ったことを確認しないで当該フラグをクリアしたときで、かつコンパレータが正転出力の場合にアナログ入力信号が基準電圧よりも高い、または、コンパレータが反転出力の場合にアナログ入力信号が基準電圧よりも低いままだった場合、前述のPOECMPFR.CjFLAG フラグは再び“1” にはなりませんのでご注意ください。

■ 発振停止検出で出力が停止した端子は以下のいずれかで解除されます

- ・リセットで初期状態に戻す
- ・SYSTEM.OSTDSR.OSTDF フラグを“0” にしてICSR6.OSTSTF フラグを“0” にする

注意事項

- 製品によっては、リセット後からMTU3,4,6,7端子のハイインピーダンス化や一部POE端子による出力条件が有効になっているものがあります。また、MTU3,4,6,7端子を使用しない場合(該当端子を他モジュールとして使用する場合)も、ハイインピーダンス化が有効になる製品もあります。これを問題とする場合、リセット後に該当のPOE_CR2レジスタやMnSERレジスタをクリアする必要があります。詳細はハードウェアマニュアルをご参照ください。

POEを使用したアプリケーションノート一覧

POEを使用したアプリケーションノートを準備しています。より詳細な使用方法は下記をご参照ください

- RX24Uグループ POE3AとMTU3dによるPWM出力停止と復帰方法 [R01AN4140](#)
- RX63Tグループ コンパレータ検出によるPOEのPWM出力遮断方法について [R01AN1404](#)

GPTW

GPT/GPTW 主動作の概要

■PWM出力動作

GTCCR_xで設定したDutyとGTPRで設定した周期でGTIOCN_{A/B}にPWM出力する動作。モードは以下から選択。

- のこぎり波PWMモード：1回/1周期のコンペアマッチでPWMを生成。アップ/ダウンのカウンタ動作選択が可能。
- のこぎり波ワンショットパルスモード：2回/1周期のコンペアマッチでPWMを生成。アップ/ダウンのカウンタ動作選択が可能。左右非対称の相補PWM出力が可能。
- 三角波PWMモード1：カウンタモードを三角波にし、アップ時とダウン時のコンペアマッチ値が同一となるPWMを生成。バッファ動作は谷固定。左右対称の相補PWM出力が可能。
- 三角波PWMモード2：三角波PWMモード1に対し、バッファ動作を山と谷の両方にし、アップ時とダウン時の各々のコンペアマッチ値を設定可能にしたモード。左右非対称の相補PWMが出力可能。
- 三角波PWMモード3：三角波PWMモード2に対し、谷割り込みのみでアップ時とダウン時の各々のコンペアマッチ値を設定可能にしたモード。左右非対称の相補PWM出力が可能。

■イベントカウンタ動作

ELCのイベント信号、もしくはGTETR_{Gx}、GTIOCN_x端子で選択したエッジによりカウンタする動作。カウンタソースは以下から選択。

- ELCのイベント信号によるカウンタ動作
- GTETR_{Gx}端子から選択したエッジ(立ち上がり、立ち下がり、両エッジ)によるカウンタ動作
- GTIOCN_x端子から選択したエッジ(立ち上がり、立ち下がり、両エッジ)によるカウンタ動作
- GTIOCN_{A/B}の組み合わせで、片方をレベル判定にし、もう一方をエッジ(立ち上がり、立ち下がり、両エッジ)によるカウンタ動作(位相計数機能)主に、ホールICやレゾルバのカウンタとして使用しモータ制御の位置制御に使用される。

■インプットキャプチャ動作

ELCのイベント信号、もしくはGTETR_{Gx}、GTIOCN_x端子で選択したエッジによりインプットキャプチャする動作。インプットキャプチャ時のカウンタ動作はのこぎり波、三角波から選択。インプットキャプチャ要因は以下から選択。

- ELCのイベント信号
- GTETR_{Gx}端子によるエッジ(立ち上がり、立ち下がり、両エッジ)
- GTIOCN_x端子によるエッジ(立ち上がり、立ち下がり、両エッジ)、またはGTIOCN_{A/B}の組み合わせ要因(レベル判定とエッジ(立ち上がり、立ち下がり、両エッジ)判定の論理積)

GPT/GPTW 各機能の概要

■デッドタイム自動設定機能

GTCCRAレジスタ値とデッドタイム値(GTDVU, GTDVD レジスタ値) からデッドタイム付き逆相波形用のコンペアマッチ値をGTCCRBレジスタに**自動設定する機能**。
のこぎり波ワンショットパルスモードと、三角波PWM モード1/2/3で使用可能。
デッドタイム値(GTDVU, GTDVD レジスタ値)を超えた値をGTCCRAレジスタに設定した場合、デッドタイムエラーが発生し、正相と逆相の波形を整形。
詳細は、以降ページの「デッドタイム自動設定機能におけるコンペア値設定のご注意」を参照。

■デューティ0/100%出力

GTUDDTYC.OADTY[1:0], OBDTY[1:0] ビットの値を変更することで、設定した次の周期からGTIOCnA/B端子の出力デューティを0%、もしくは100% にする機能。
GTIOCnAの0%/100%出力設定はGTUDDTYC.OADTY[1:0]で、 GTIOCnBの0%/100%出力設定はGTUDDTYC.OBDTY[1:0]で各々設定しますので、
GTIOCnA/Bの出力は、同レベル、異レベルの出力が可能。

■同期機能

各チャンネルのカウント動作(スタート、ストップ、クリア動作)を同期動作させる機能。応用として、3チャンネルを同期動作させた三相相補PWM出力や、カウントスタート前に各チャンネルのGTCNT カウンタ値を設定しておくことで、各チャンネル間に位相差をつけたカウントスタートなど有り。同期動作は、以下ソフトウェアとハードウェア要因が有る。

- ・ソフトウェアによる同期
同期スタートはGTSTRレジスタ、同期ストップはGTSTPLレジスタ、同期クリアはGTCLRレジスタに同期するチャンネルを設定する事で同期動作可能。GTSTR、GTSTP、GTCLRは各チャンネル毎にレジスタがあるが、共通レジスタの為、どのチャンネルのレジスタに設定しても動作。
- ・ハードウェアによる同期
同期スタートはGTSSRレジスタ、同期ストップはGTPSRレジスタ、同期クリアはGTCSRレジスタに同じハードウェア要因をチャンネル毎に設定する事で同期動作可能。

■間引き機能

GTITCレジスタの設定により、カウンタ(GTCNT)のオーバフロー(GTPRレジスタのコンペアマッチ)/アンダフローの割り込み(GTCIV/GTCIU)を 指定した最大7回分間引く(割り込みを発生させなくする)ことが可能。また、他の割り込み、およびA/D 変換開始要求をGTCIV/GTCIU割り込み間引き機能と連動して間引くことが可能。さらに拡張割り込み間引き機能により、通常の間引き機能と独立して、2つの間引きカウンタにより最大2種の選択した対象割り込みやA/D変換開始要求を最大16回間引きが可能。

GPT/GPTW 動作内容に対する機能一覧

GPT/GPTWの以下機能は、チャンネル毎に設定します。 三相相補PWM出力などはチャンネル間同期制御により機能を実現します。

動作内容		アウトプット コンペア (GTIOR, GTIOA/B[4:0])	カウン ト方 向設 定	デッドタイム 自動設定 (GTDTCR,GTDVU/D ←GTDBU/D)	デューティ 0/100%出力 (GTUDDTYC)	同期機能 (GTSTR,GTSTP, GTCLR, GTSSR,GTCSR,GT CSR)	バッファ 機能 (GTBER)	A/D変換 開始トリガ (GTADTR ←GTADTBR ←GTADTDBR)	間引き機能 (GTITC, GTEITC, GTEITLI1/I2/B)
PWM出力動作 (GTCR.TPCS[3:0]]クロック設定、 GTCR.MD[2:0] モード設定)*1	のこぎり波PWMモード	○ *2	○ *6	-	○	○ *8	○	○	○
	のこぎり波ワンショット パルスモード								
	三角波PWMモード1		-	○					
	三角波PWMモード2		*3						
	三角波PWMモード3								
イベントカウン ト動作 (GTUPSR,GTDNS R設定イベント)*1	ELCイベント or 外/部イベント (GTETRGA~D or GTIOcNA/B)	△ *4	△ *7	-	-		△ *9	-	-
	位相計数機能 (GTIOcNA/B)	- *5							
インプットキャ プチャ動作 (GTICASR/GTIC BSR設定要因)*2	のこぎり波(PMW/ワン ショット)	-	○ *6	-	-	○	○	-	-
	三角波(PWMモード 1./2/3)	-	- *3	-	-	○	○	-	-

*1: GTUPSR,GTDNSR≠0x00の場合、強制的にイベントカウント動作となり、GTCR.TPCS[3:0]、GTCR.MD[2:0]、およびGTUDDTYC.UDの設定(PWM出力動作)は無効となります。

*2: GTICASR/GTICBSR=0で GTIORを設定するとアウトプットコンペアになります。(GTICASR/GTICBSR≠0の場合はインプットキャプチャになり、アウトプットコンペアに使用出来ません)

*3: GTUDDTYC.UDは、 カウントストップ中にGTUDDTYC.UDF="1"とする事でカウントスタート時のみ有効となります。これは谷からスタート(アップカウント)させるか、山からスタート(ダウンカウント)させるかの設定のみ有効となります。

*4: GTIOR.GTIOA/B[4:0]の設定の内、周期の終わりで波形変化する設定は無効となります。なお、GTIOcNA/B端子を使用する場合、片方をイベントカウント、片方をアウトプットコンペアに設定可能です。

*5: イベントカウントにGTIOcNA/Bを使用する為、アウトプットコンペア出力は不可となります。

*6: GTUDDTYC.UDでカウント方向を設定します。

*7: GTUPSR,GTDNSRの設定によりアップ/ダウン動作となります。

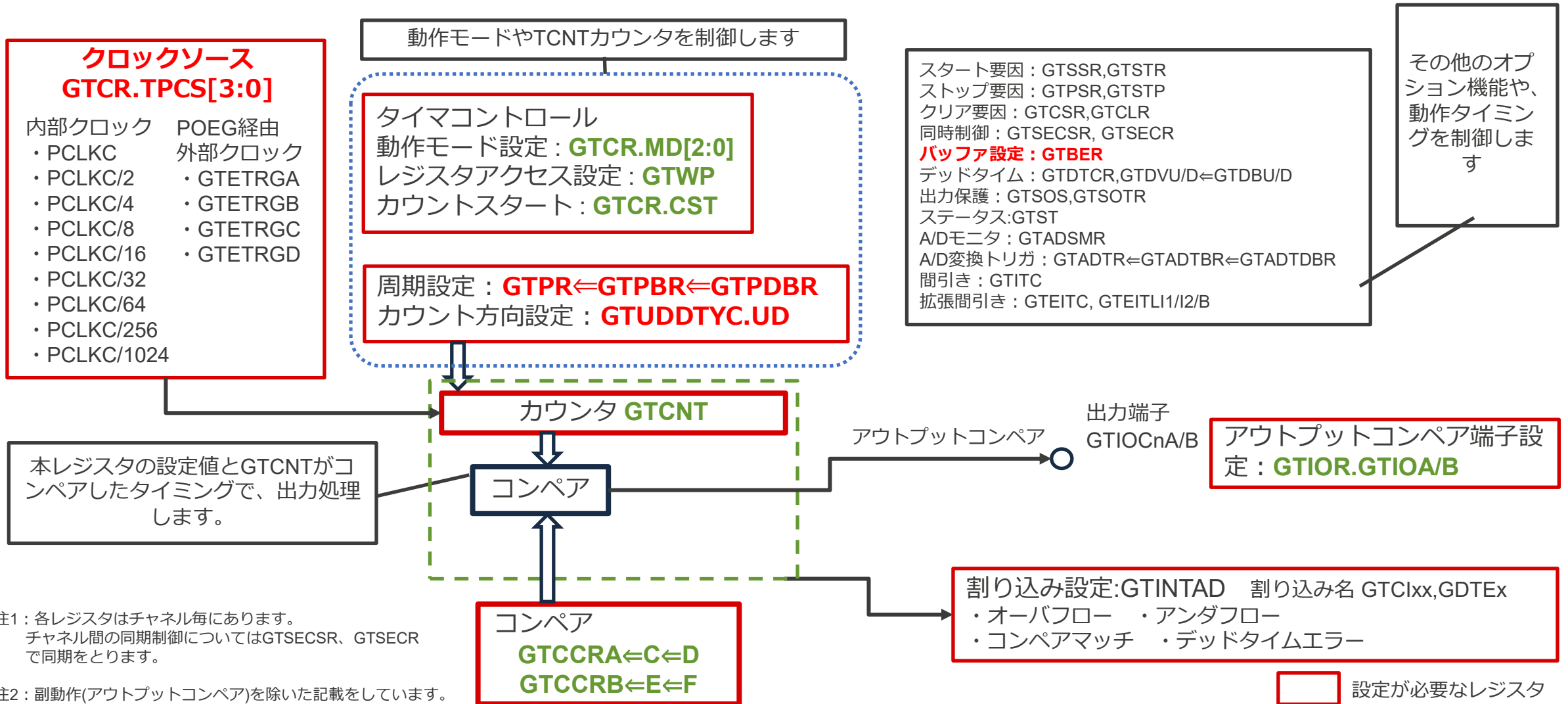
*8: GTCLRによるソフトウェアクリア動作は、GTCR.MD[2:0]がのこぎり波でダウンカウントのみ有効となります。

*9: 周期、コンペアのみ有効となります。

本内容は、RX66TのGPTWをベースに記載していますが、他のGPT/GPTW搭載品でも参考となります。

PWM出力動作 レジスタ一覧

(のこぎり波PWM、のこぎり波ワンショットパルス、三角波PWMモード1,2,3)



PWM出力動作 レジスタ設定

(のこぎり波PWM、のこぎり波ワンショットパルス、三角波PWMモード1/2/3)

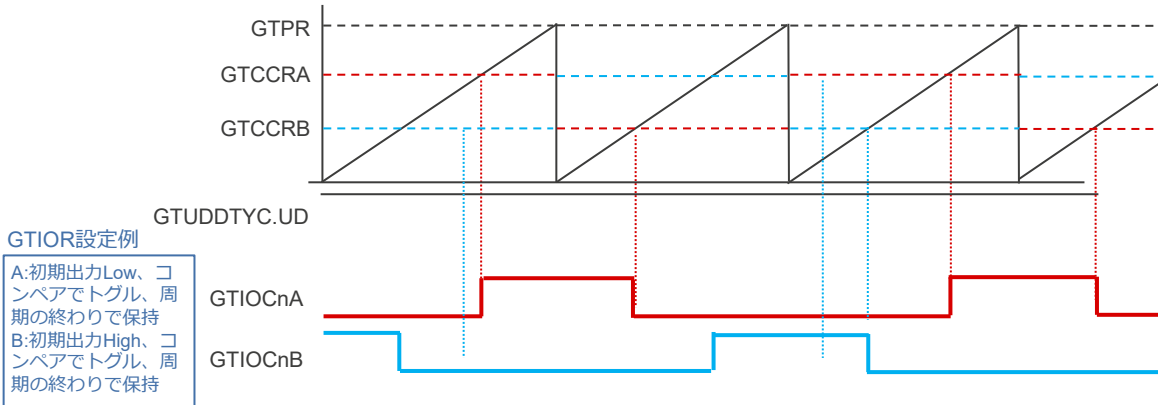
設定の流れ

		のこぎり波 PWM	のこぎり波 ワンショット	のこぎり波ワンショット (デッドタイム自動設定)	三角波PWM モード1/2	三角波PWMモード1/2 (デッドタイム自動設定)	三角波PWM モード3	三角波PWMモード3 (デッドタイム自動設定)
1	動作モード設定 GTCR.MD[2:0]	✓	✓	✓	✓	✓	✓	✓
2	カウント方向設定 GTUDDTYC	✓	✓	✓	(アップ・ダウンカウン트의為、設定不要)	(アップ・ダウンカウン트의為、設定不要)	(アップ・ダウンカウン트의為、設定不要)	(アップ・ダウンカウン트의為、設定不要)
3	カウントクロック設定 GTCR.TPCS[3:0]	✓	✓	✓	✓	✓	✓	✓
4	周期設定 GTPR	✓	✓	✓	✓	✓	✓	✓
5	カウンタ初期値設定 GTCNT	✓	✓	✓	✓	✓	✓	✓
6	GTIOCnm端子設定 GTIOR.GTIOA[4:0], GTIOB[4:0]	✓	✓	✓	✓	✓	✓	✓
7	GTIOCnm端子出力許可設定 GTIOR.OAE, OBE	✓	✓	✓	✓	✓	✓	✓
8	バッファ動作設定 GTBER.CCRA[1:0], CCRB[1:0]	✓	-	-	✓	✓	-	-
9	コンペアマッチ値設定 GTCCRA,GTCCRB	✓	-	-	✓	(GTCCRAを設定)	-	-
10	バッファ値設定 GTCCRD,GTCCRC,GTCCRF,GTCCRE	-	(GTCCRA,GTCCRBに強制転送する値をセット)	(GTCCRAに強制転送する値をGTCCRCとGTCCRDにセット)	-	-	(GTCCRA,GTCCRBに強制転送する値をセット)	(GTCCRAに強制転送する値をGTCCRCとGTCCRDにセット)
11	バッファ値強制転送 GTBER.CCRSWT=1	-	(強制転送でGTCCRA,GTCCRBに値をセット)	(強制転送する事でGTCCRAに値をセット)	-	-	(強制転送する事でGTCCRA,GTCCRBに値をセット)	(強制転送する事でGTCCRAに値をセット)
12	バッファ値設定 GTCCRD,GTCCRC,GTCCRF,GTCCRE	✓	✓	✓	✓	(GTCCRC,GTCCRDを設定)	✓	✓
13	デッドタイム自動設定機能設定 GTDTCR.TDE	-	-	✓	-	✓	-	✓
14	デッドタイム用バッファ動作設定 GTDTCR.TDBUE, TDBDE	-	-	✓	-	✓	-	✓
15	デッドタイム値設定 GTDVU,GTDVD (GTDTCR.TDFER=1にする事でGTCVUと同一値をGTDVDにセットします)	-	-	✓	-	✓	-	✓
16	デッドタイム用バッファ値設定 GTDBU, GTDBD	-	-	✓	-	✓	-	✓
17	カウントスタート GTCE.CST=1	✓	✓	✓	✓	✓	✓	✓
18	バッファ値設定 GTCCRC,GTCCRE,GTCCRD,GTCCRF	✓	✓	✓	✓	✓	✓	✓

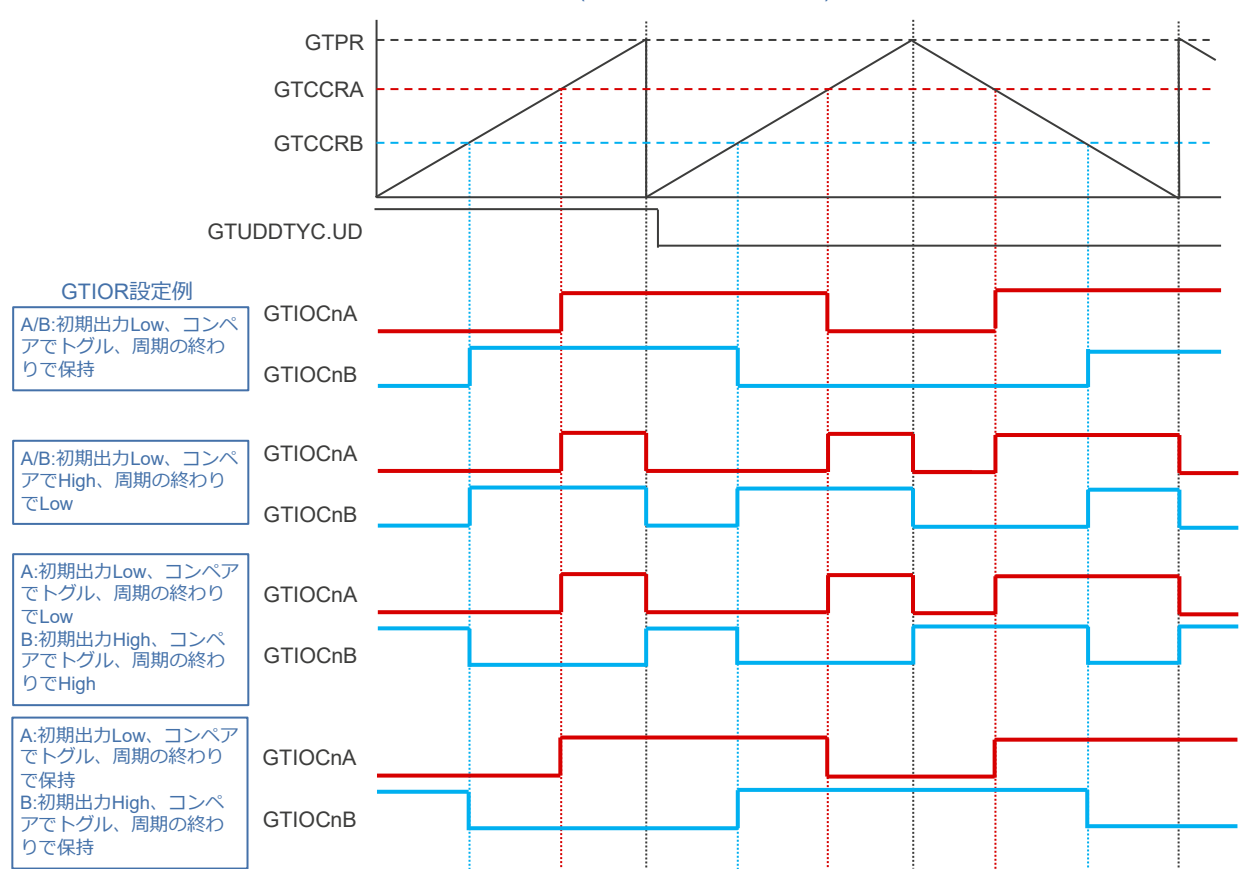
のこぎり波PWMモード

1回/周期のコンペアマッチによりGTIOCnAとGTIOCnB からPWM出力するモードです。各々のコンペア値により変化タイミングを制御します。デッドタイム自動設定は使用できません。

- ・周期 : GTPR
- ・PWM出力波形 : GTIOR設定、GTCCRA/GTCCRBのコンペアに対応した波形を出力
- ・カウント方向 : アップカウント (GTUDDTYC.UD=1)



- ・周期 : GTPR
- ・PWM出力波形 : GTIOR設定、GTCCRA/GTCCRBのコンペアに対応した波形を出力
- ・カウント方向 : アップカウント→ダウンカウント (GTUDDTYC.UD=1→0)



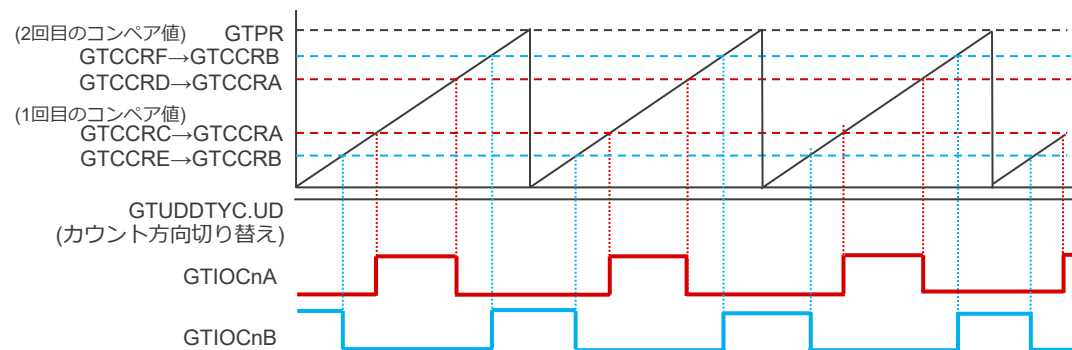
のこぎり波ワンショットパルスモード

2回/周期のコンペアマッチによりGTIOcN_AとGTIOcN_BからPWM出力するモードです。デッドタイム自動設定、または各々のコンペア値により変化タイミングを制御します。

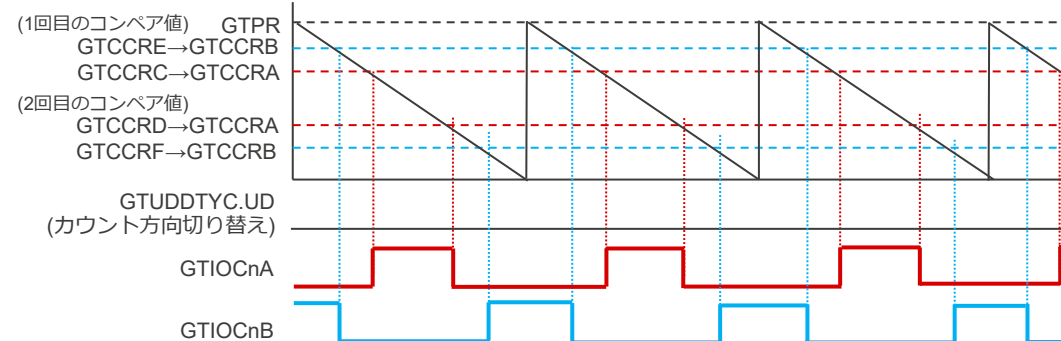
GTIOR設定例

- A:初期出力Low、コンペアでトグル、周期の終わりでLow
- B:初期出力High、コンペアでトグル、周期の終わりでHigh
- ・周期 : GTPR
- ・PWM出力波形 : GTIOR設定、GTCCRA/GTCCRBのコンペアに対応した波形を出力

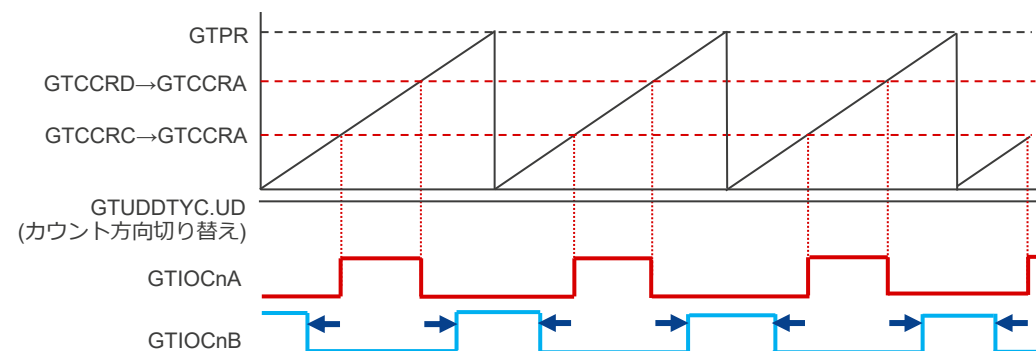
正相/逆相コンペアマッチ、アップカウント時



正相/逆相コンペアマッチ、ダウンカウント時

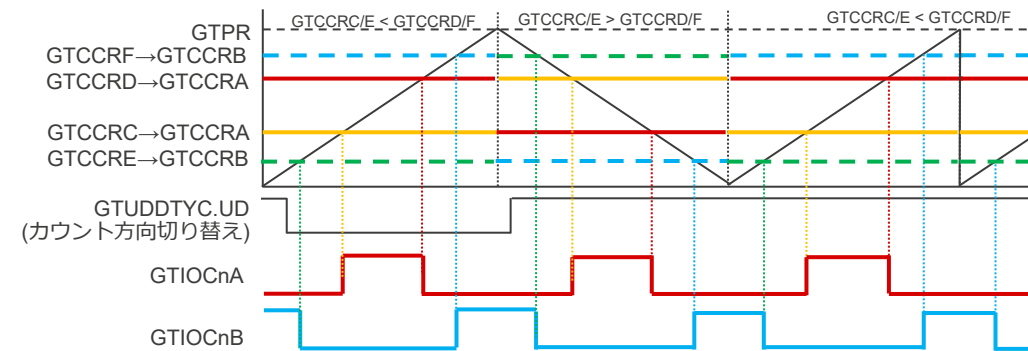


正相コンペアマッチ、逆相デッドタイム自動設定、アップカウント時



左右自動デッドタイム(GTCCRBの値を自動設定)

正相/逆相コンペアマッチ、アップ/ダウンカウント切り替え時



割り込み発生タイミング

(のこぎり波PWM、のこぎり波ワンショットパルス、三角波PWMモード1/2/3)

のこぎり波、三角波の割り込み発生タイミングを以下に示します。

■ のこぎり波

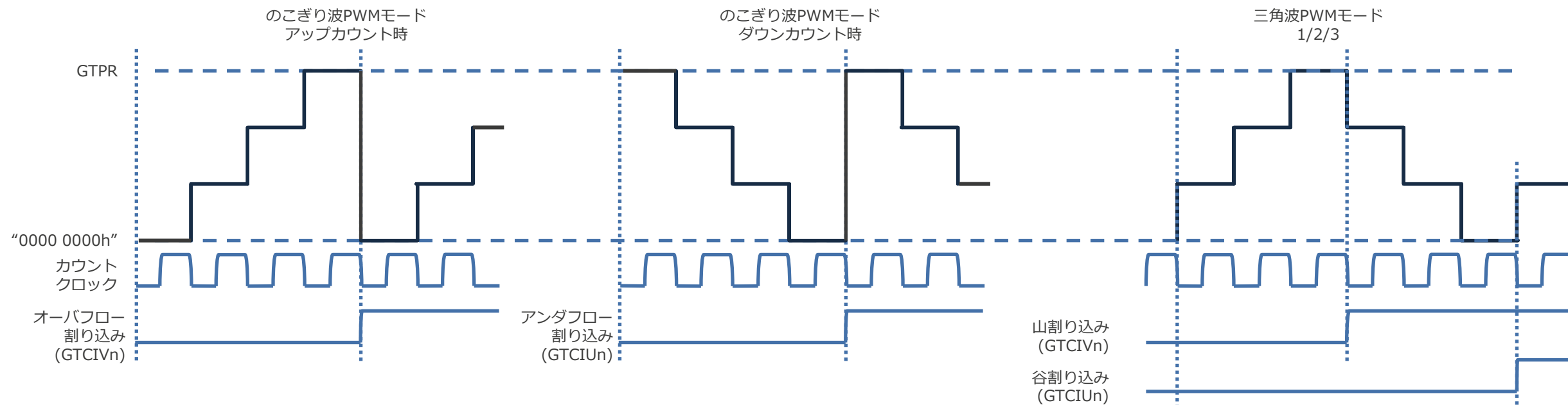
アップカウント時の割り込み(GTICV) : GTCNT カウンタ値がGTPR レジスタ値から“0000 0000h” (オーバフロー)になったとき

ダウンカウント時の割り込み(GTICU) : GTCNT カウンタ値が“0000 0000h” からGTPR レジスタ値(アンダフロー)になったとき

■ 三角波

山割り込み(GTICV) : GTCNT カウンタ値がGTPR レジスタ値からGTPR レジスタ値-1 になったとき

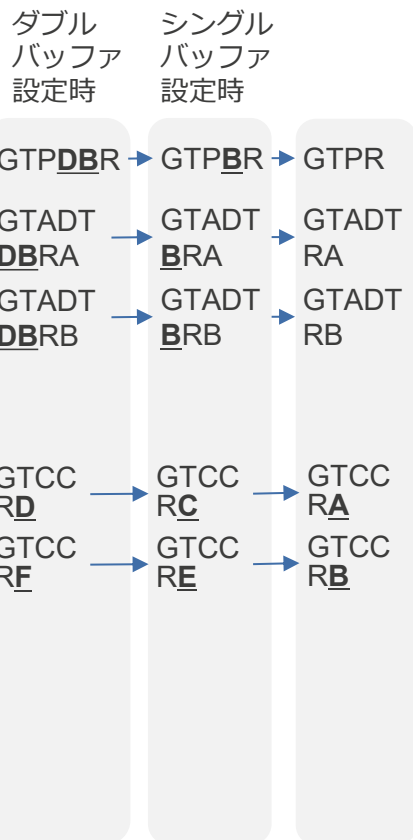
谷割り込み(GTICU) : GTCNT カウンタ値が“0000 0000h” から“0000 0001h” になったとき



バッファと転送タイミング (のこぎり波PWM、のこぎり波ワンショットパルス)

バッファ(ダブルバッファ)の値は、山または谷のタイミングで次のレジスタに転送されます。バッファ転送タイミングは各モードにより異なります。

のこぎり波PWM出力モード



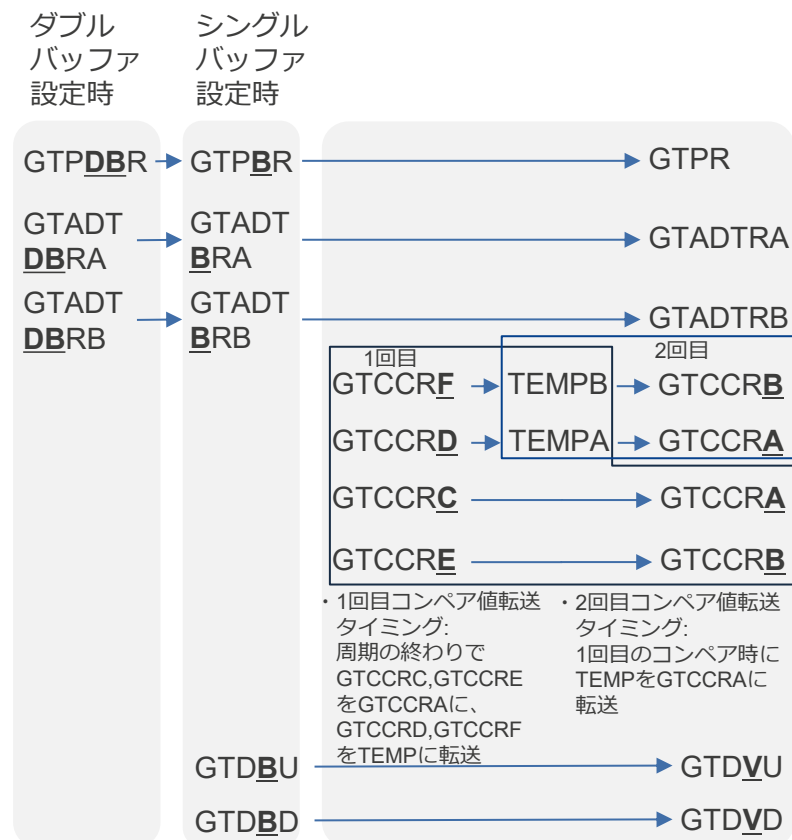
GTBER.DB[1]=0,
GTBER.PR[1:0]=01,1xで
GTPRのバッファレジスタ
(GTPBR,GTPDBR)が
有効

GTBER.DB[2]=0,
GTBER.ADTTA/B=0,1で
GTADTRA/Bのバッファレジスタ
(GTADTBRA/B, GTADTDBRA/B)
が有効

GTBER.DB[0]=0,
GTBER.CCRA[1:0]/CCRB[1:0]=01,1
XでGTCCRA/Bのバッファレジスタ
(GTCCRC/D,E/F)が有効
(のこぎり波ワンショットパルスモード
では、GTBER.DB[0]の設定は無効と
なり、強制動作となります)

GTBER.DB[3]=0,
GDTDCR.TDBUE=1,
GDTDCR.TDBDE=1で
GTDVU/Dのバッファレジスタ
(GTDBU/D)が有効

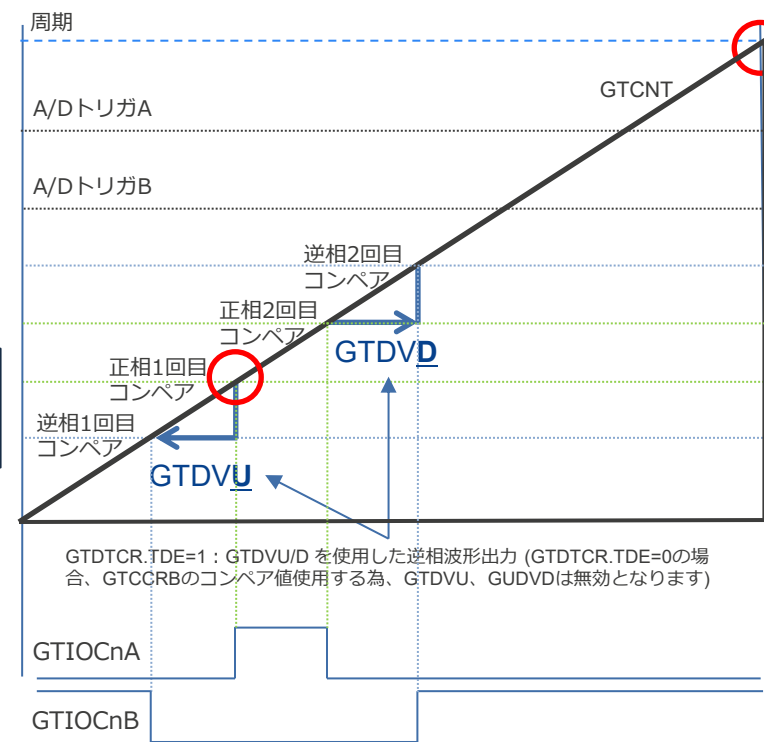
のこぎり波ワンショットパルスモード



デッドタイム自動付加機能使用時(GTDCR.TDE=1)
アップ時: GTCCRB=GTCCRA-GTDVU
ダウン時: GTCCRB=GTCCRA-GTDVD

バッファ転送タイミング

- ・オーバーフロー(アップカウント時)
- ・アンダフロー(ダウンカウント時)
- ・一回目のコンペア(のこぎり波ワンショットパルスモードのみ)



PWM出力動作 のこぎり波ワンショットパルスモードにおけるカウント方向切り替えについて

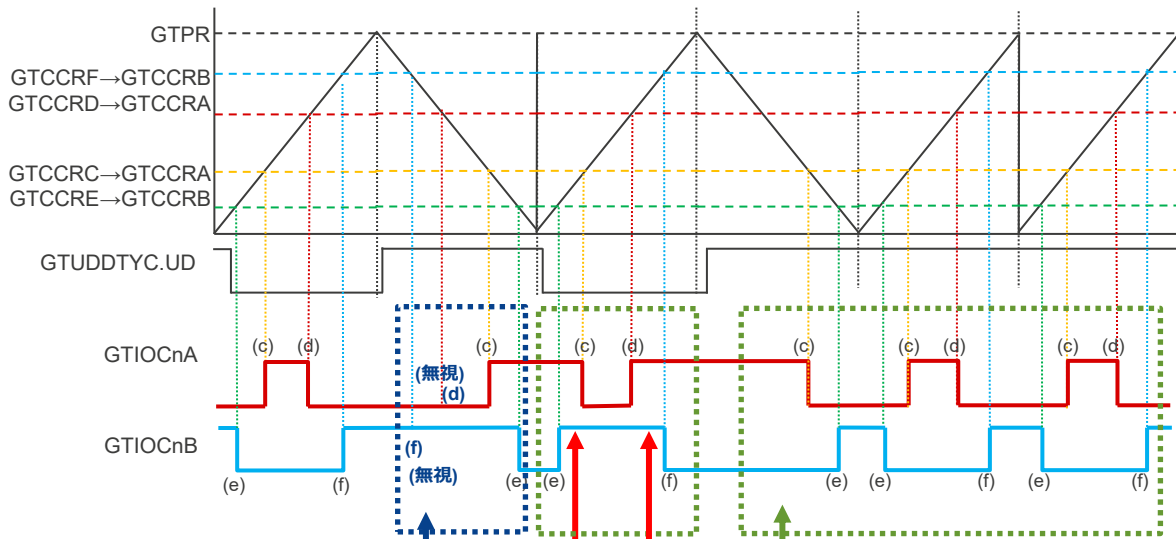
ワンショットパルスモードでは、コンペア順が以下の通り決まっています。

■コンペアの順番：正相：GTCCRC → GTCCRD, 逆相：GTCCRE → GTCCRF

その為、**カウント方向を切り替える場合は、コンペアマッチ値を正しく設定しないと、コンペアが2回/1周期 発生せず正相と逆相がActiveになる（ショートする）**場合がありますので、ご注意ください。

- 設定内容
 - ・カウント方向：アップ→ダウン→アップ→ダウン→アップカウント (GTUDDTYC.UD=1→0→1→0→1)
 - ・GTIOR設定：A:初期出力Low、コンペアでトグル、周期の終わりで保持、B:初期出力High、コンペアでトグル、周期の終わりで保持

■正相/逆相ともコンペアマッチ時のカウント方向切り替え波形例

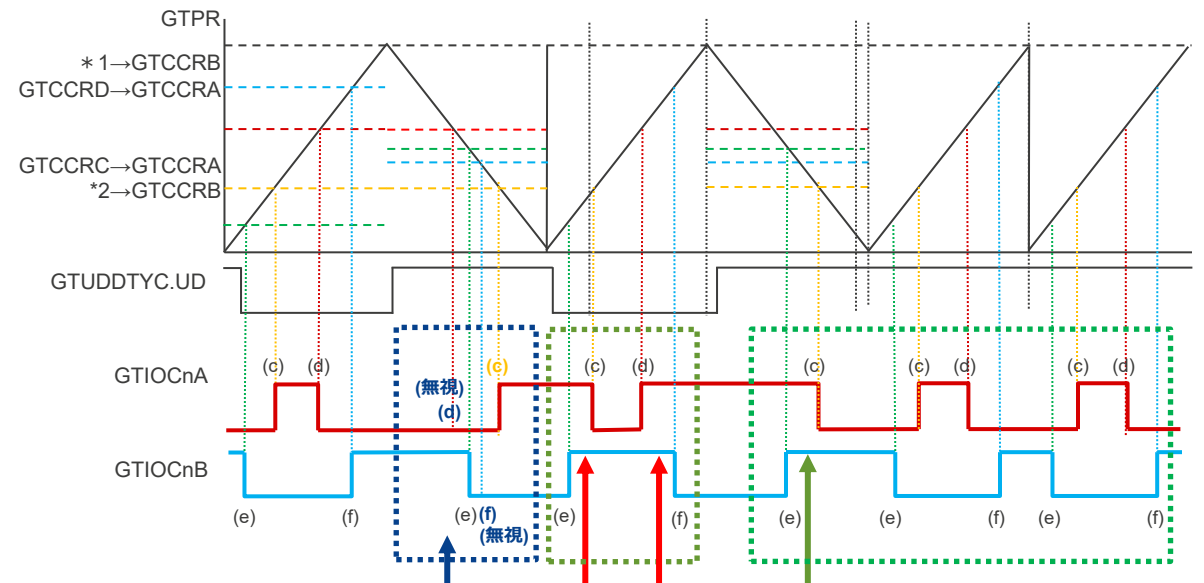


コンペア順のルールから外れるためd,fのコンペアが無視
また、正相と逆相のコンペア順が入れ替り、ショートする

次の周期は、
波形が逆転

再度コンペアを1回のみ発生させると、
次の周期から正常に戻る

■正相はコンペアマッチ、逆相はデッドタイム自動設定時のカウント方向切り替え波形例



コンペア順のルールから外れるため、d,fのコンペアが無視

次の周期は、
波形が逆転

再度コンペアを1回のみ発生させると、
次の周期から正常に戻る

*1:アップカウント時:(GTCCRA+GTDVD)→GTCCRB, ダウンカウント時:(GTCCRA-GTDVD)
*2:アップカウント時:(GTCCRA-GTDVU)→GTCCRB, ダウンカウント時:(GTCCRA+GTDVU)

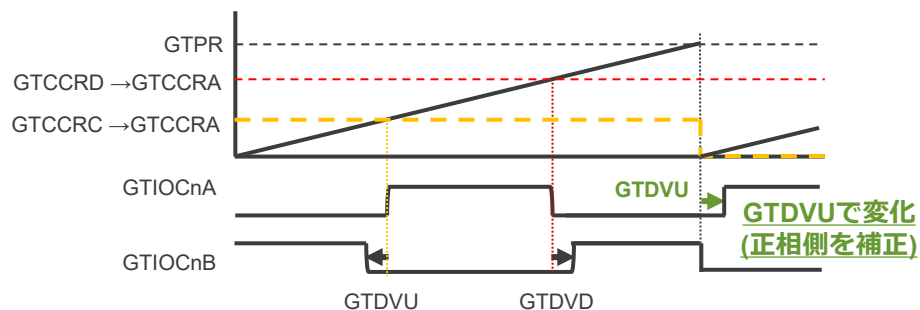
のこぎり波ワンショットパルスモードにおけるデッドタイムエラー時の動作 (デッドタイム自動設定におけるコンペア値設定のご注意)

通常デッドタイム補正は逆相波形(GTIOCnB端子)が対象ですが、デッドタイムエラー発生時(デッドタイムが確保されないコンペア値が正相側で設定された場合)、**正相波形(GTIOCnA端子)のコンペアマッチを補正**します。

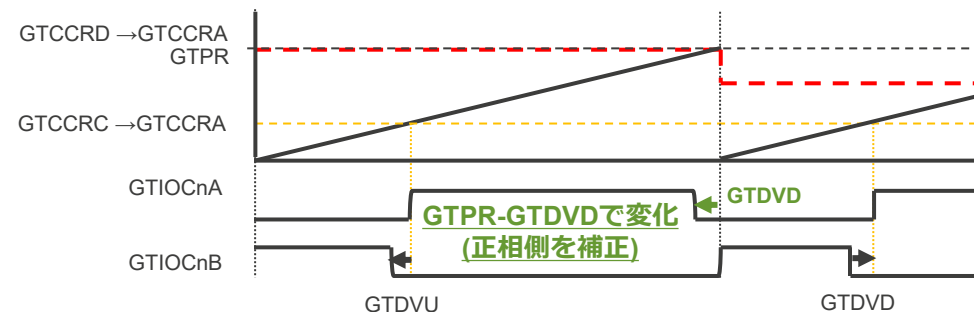
GTIOR設定

- A:初期出力Low、コンペアでトグル、周期の終わりで保持
- B:初期出力High、コンペアでトグル、周期の終わりで保持

・アップカウント時のデッドタイムエラー

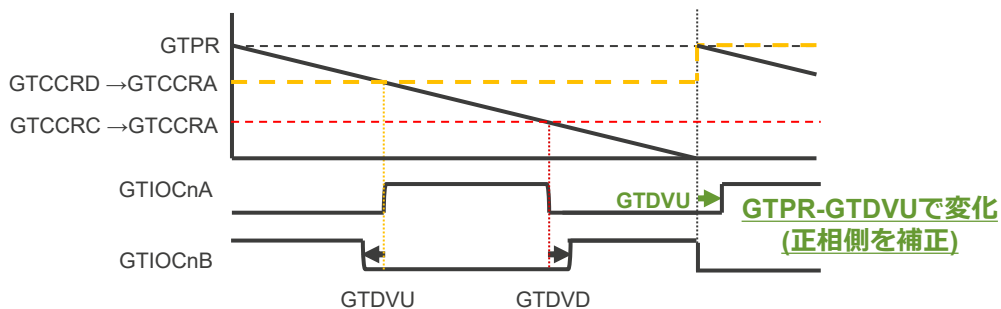


**GTCCRC(GTCCRA)がカウント = 0で変化するケース
($GTCCRA - GTDVU < 0$ でデッドタイムエラー)**

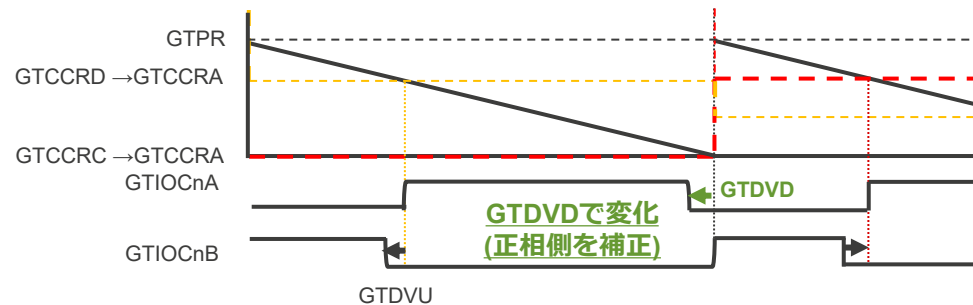


**GTCCRD(GTCCRA)がカウント = GPTRで変化するケース
($GTCCRA + GTDVD > GPTR$ でデッドタイムエラー)**

・ダウンカウント時のデッドタイムエラー



**GTCCRD(GTCCRA)がカウント=GPTRで変化するケース
($GTCCRA + GTDVU > GPTR$ でデッドタイムエラー)**



**GTCCRC(GTCCRA)がカウント = 0で変化するケース
($GTCCRA - GTDVD < 0$ でデッドタイムエラー)**

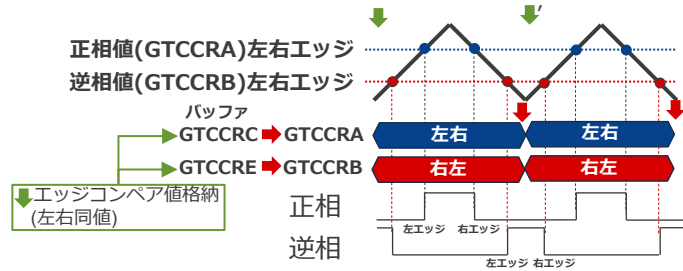
三角波PWMモード 1/2/3 の出力波形

4回/周期のコンペアマッチによりGTIOcNAとGTIOcNB からPWM出力するモードです。デッドタイム自動設定、または各々のコンペア値により変化タイミングを制御します。

三角波PWMモード1(谷転送)

谷でコンペア値を設定 (左右対称)

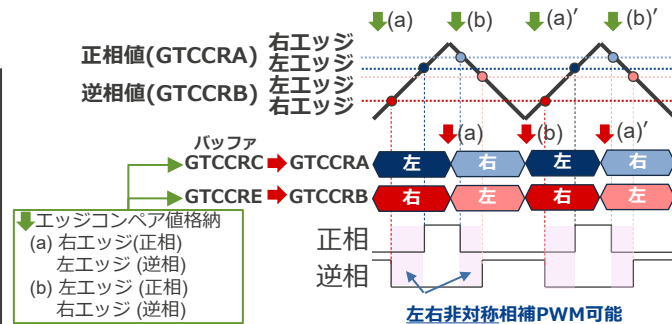
■ 正相/逆相コンペアマッチ制御



正相/逆相とも1回/週のコンペア値設定で
アップ/ダウンカウンツ両方に同一値を設定

三角波PWMモード2 (山谷転送)

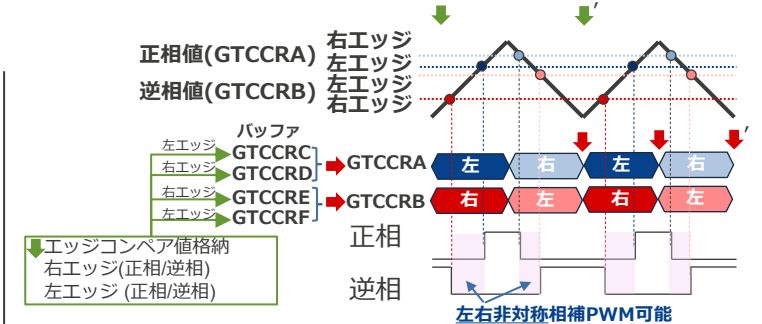
山谷各々でコンペア値を設定 (左右非対称)



正相/逆相とも2回/週のコンペア値設定で
アップ/ダウンカウンツ各々個別のコンペア値を設定

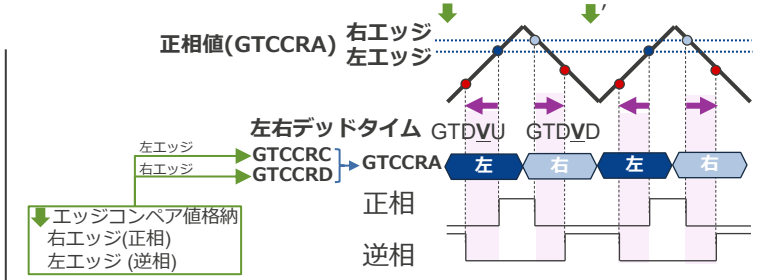
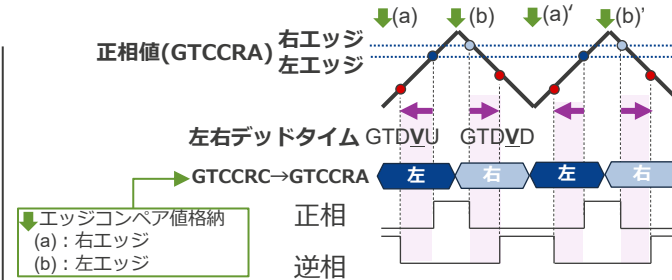
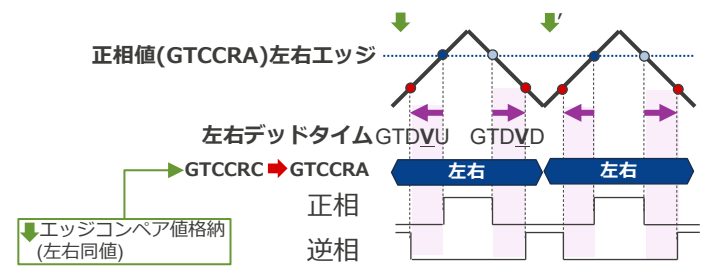
三角波PWMモード3(谷64bit転送)

谷でアップ/ダウン両方のコンペア値を設定 (左右非対称)



正相/逆相とも1回/週のコンペア値設定で
アップ/ダウンカウンツ各々個別のコンペア値を設定

■ 正相 : コンペアマッチ制御、逆相 : 自動設定 (デッドタイム自動設定)



逆相のコンペア値は、左右のデッドタイム設定値に応じて自動設定(CPU負荷軽減)

三角波PWMモード1/2/3 割り込み発生タイミング

三角波PWMモード1/2/3で割り込みが発生するポイントを以下に示します。

割り込み要因

①GTCIV n

GTCNTカウンタのオーバフロー

②GTCIA n (波形生成用)

GTCCRAのインプット/コンペアマッチ

③GTCIB n (波形生成用)

GTCCRBのインプット/コンペアマッチ

④GDTE n

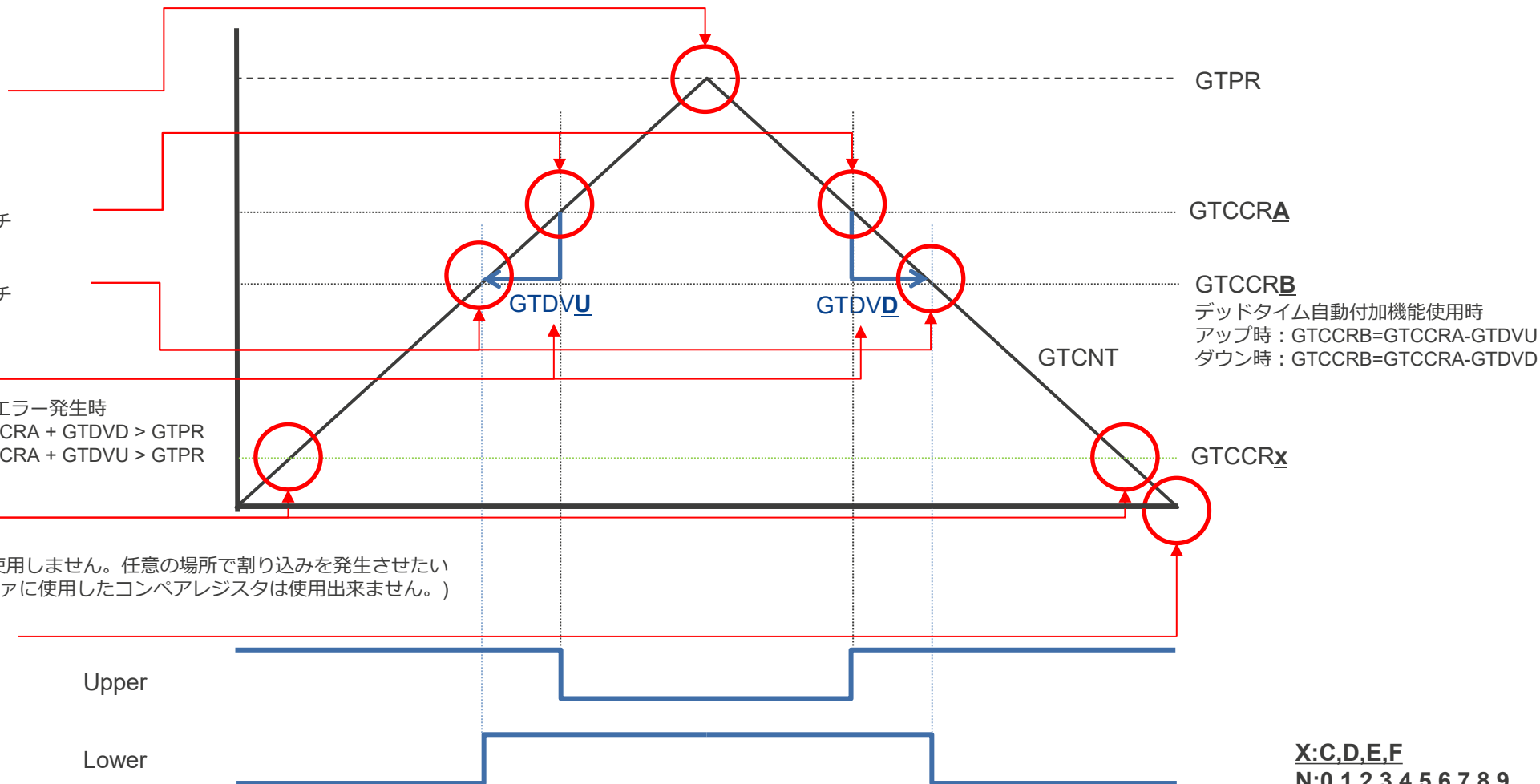
デッドタイム自動付加機能使用で以下エラー発生時
 アップ時 : $GTCCRA - GTDVU < 0$, $GTCCRA + GTDVD > GTPR$
 ダウン時 : $GTCCRA - GTDVD < 0$, $GTCCRA + GTDVU > GTPR$

⑤GTCIx n (波形生成以外)

その他GTCCR_xコンペアマッチ
 (本コンペアレジスタは波形成型には使用しません。任意の場所で割り込みを発生させたいなどの時に使用下さい。ただしバッファに使用したコンペアレジスタは使用出来ません。)

⑥GTCIU n

GTCNTカウンタのアンダフロー



X:C,D,E,F
 N:0,1,2,3,4,5,6,7,8,9

三角波PWMモード1/2/3 バッファ転送タイミング

バッファ(ダブルバッファ)の値は、山または谷のタイミングで次のレジスタに転送されます。バッファ転送タイミングは各モードにより異なります。

三角波PWM出力モード1(谷でバッファ転送) 三角波PWM出力モード2(山谷でバッファ転送)

ダブル
バッファ
設定時

シングル
バッファ
設定時

GTPDBR → GTPBR → GTPR

GTADTD
BRA → GTADT
BRA → GTAD
TRA

GTADTD
BRB → GTADT
BRB → GTAD
TRB

GTCCRD → GTCCRC → GTCCRA

GTCCRF → GTCCRE → GTCCRB

GTBER.DB[1]=0, GTBER.PR[1:0]=01,1xで
GTPRのバッファレジスタ(GTPBR,GTPDBR)が
有効

GTBER.DB[2]=0, GTBER.ADTT/A/B=0,1で
GTADTRA/Bのバッファレジスタ
(GTADTBRA/B, GTADTDBRA/B)が有効

GTBER.DB[0]=0,
GTBER.CCRA[1:0]/CCRB[1:0]=01,1xで
GTCCRA/Bのバッファレジスタ
(GTCCRC/D,E/F)が有効
(三角波PWM出力モード3、のこぎり波ファンショッ
トパルスモードでは、GTBER.DB[0]の設定は無
効となり、強制動作となります)

GTBER.DB[3]=0, GTDTCR.TDBUE=1,
GTDTCR.TDBDE=1 で
GTDVU/Dのバッファレジスタ(GTDBU/D)が有効

三角波PWM出力モード3 (山谷でバッファ転送)

ダブル
バッファ
設定時

シングル
バッファ
設定時

GTPDBR → GTPBR → GTPR

GTADTD
BRA → GTADT
BRA → GTAD
TRA

GTADTD
BRB → GTADT
BRB → GTAD
TRB

GTCCRC → GTCCRA

GTCCRD → GTCCRA

GTCCRE → GTCCRB

GTCCRF → GTCCRB

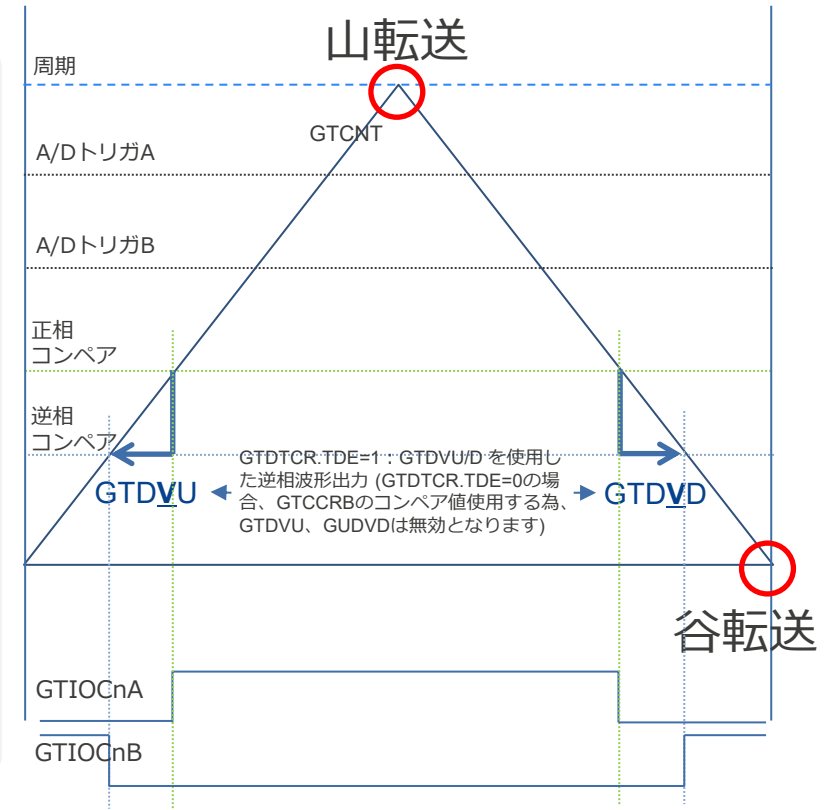
- ・アップ時使用コンペア:
GTCCRC,GTCCRE
- ・ダウン時使用コンペア:
GTCCRD,GTCCRF

GTDBU → GTDVU

GTDBD → GTDVD

デッドタイム自動付加機能使用時(GTDTCR.TDE=1)
アップ時: GTCCRB=GTCCRA-GTDVU
ダウン時: GTCCRB=GTCCRA-GTDVD

○ バッファ転送ポイント



三角波PWMモード1/2/3

デッドタイム自動設定機能におけるコンペア値設定のご注意

三角波PWMモード1/2/3 デッドタイム自動設定機能では、コンペアの発生条件により、デッドタイムエラー補正または出力保護機能が働きます。これにより、正相/逆相波形が補正されますのでご注意ください。下記に保護内容とコンペア発生条件および補正内容、復帰条件をまとめます。詳細は次ページ以降で説明します。

動作条件：デッドタイム自動設定機能有効時

保護内容	コンペア発生条件	発生時の波形補正内容	復帰条件	参照ページ
デッドタイムエラー補正	アップカウント時のコンペア値が $GTCCRA-GTDVU \leq 0000h$	正相:GTDVU+1でコンペア出力 逆相:GTCNTが1でトグル波形出力	発生条件が解消された時	デッドタイム自動設定機能におけるコンペア値設定のご注意 三角波PWMモード1/2/3におけるデッドタイムエラー時の動作
	ダウンカウント時のコンペア値が $GTCCRA-GTDVD < 0000h$	正相:GTDVDでコンペア出力 逆相:GTCNTが0でトグル波形出力		
出力保護機能	$GTCCRA=0000h$	<ul style="list-style-type: none"> 谷転送で発生した場合 正相:GTDVU+1でコンペア出力 逆相:GTCNTが1でトグル波形出力 山転送で発生した場合 正相:GTDVDでコンペア出力 逆相:GTCNTが0でトグル波形出力 	0 < GTCCRA < GTPR、または GTDTCR.TDE = 0(デッドタイム自動設定機能OFF)となった 周期のダウンカウントから復帰、 または、カウントストップした場合。	デッドタイム自動設定機能におけるコンペア値設定のご注意 三角波PWMモード1/2/3における出力保護機能 GTCCRA=0000hの場合
	$GTCCRA \geq GTPR$	<ul style="list-style-type: none"> 谷転送で発生した場合 正相:出力保持 逆相:出力保持 山転送で発生した場合 正相:出力保持 逆相:出力保持 	<ul style="list-style-type: none"> 谷転送で発生した場合 0 < GTCCRA < GTPR、または GTDTCR.TDE = 0(デッドタイム自動設定機能無効)となった 周期のアップカウントから復帰。 山転送で発生した場合 0 < GTCCRA < GTPR、または GTDTCR.TDE = 0(デッドタイム自動設定機能無効)となった 周期のダウンカウントから復帰。 カウントストップした場合。 	デッドタイム自動設定機能におけるコンペア値設定のご注意 三角波PWMモード1/2/3における出力保護機能 GTCCRA ≥ GTPRの場合

三角波PWMモード1/2/3におけるデッドタイムエラー時の動作 (デッドタイム自動設定機能におけるコンペア値設定のご注意)

デッドタイムが確保されないコンペア値を設定した場合、以下のPWM出力波形に整形します。
通常デッドタイム補正は逆相波形が対象ですが、デッドタイムが確保されないコンペア値を設定した場合、正相のコンペア値を補正します。

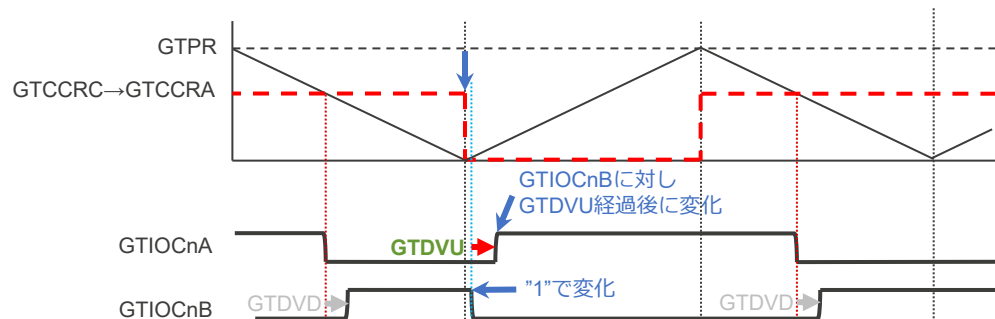
GTIOR設定

A:初期出力Low、コンペアでトグル、周期の終わりで保持
B:初期出力High、コンペアでトグル、周期の終わりで保持
正相：コンペアマッチ、逆相：デッドタイム自動設定機能

↓: デッドタイムエラー発生

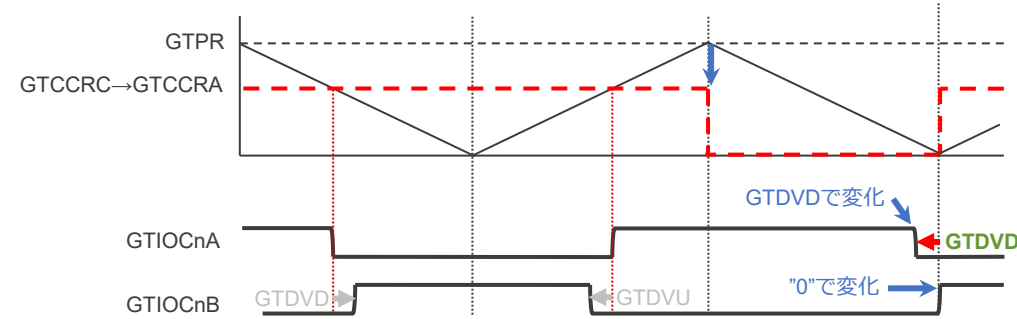
■ アップカウント時のデッドタイムエラー (GTCCRA-GTDVU ≤ 0000h)

逆相：“1”で変化 正相：GTDVU+1で変化

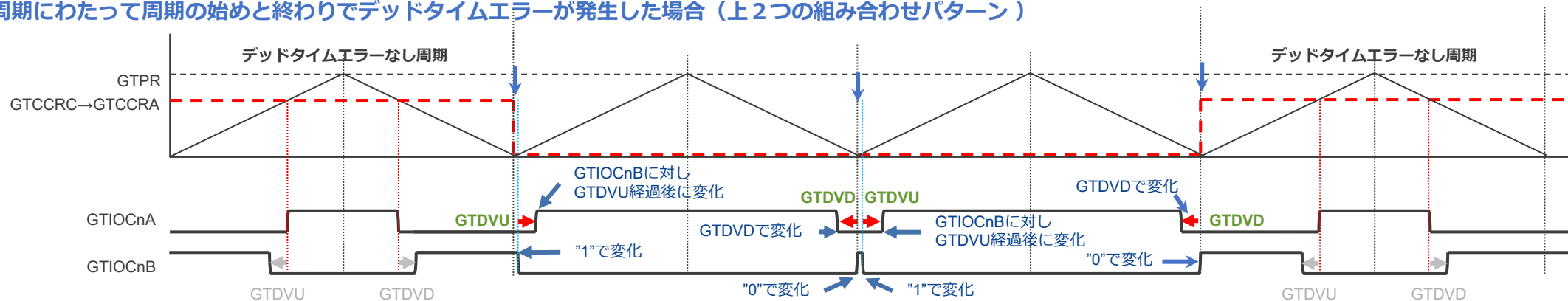


■ ダウンカウント時のデッドタイムエラー (GTCCR - GTDVD < 0000hの場合)

逆相：“0”で変化 正相：GTDVDで変化



■ 2周期にわたって周期の始めと終わりでデッドタイムエラーが発生した場合 (上2つの組み合わせパターン)

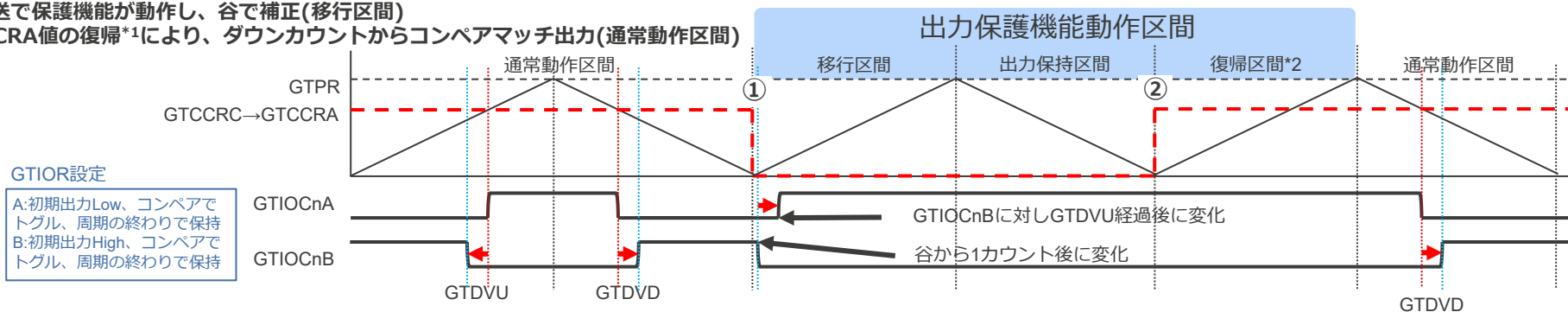


三角波PWMモード1/2/3における出力保護機能動作(GTCCRA=0000Hの場合) (デッドタイム自動設定機能におけるコンペア値設定のご注意)

GTCCRA=0000hを設定した場合、出力保護機能により以下のPWM出力波形に整形します。
通常デッドタイム付加は逆相波形が対象ですが、GTCCRA =0000hを設定した場合、正相のコンペア値を補正します。

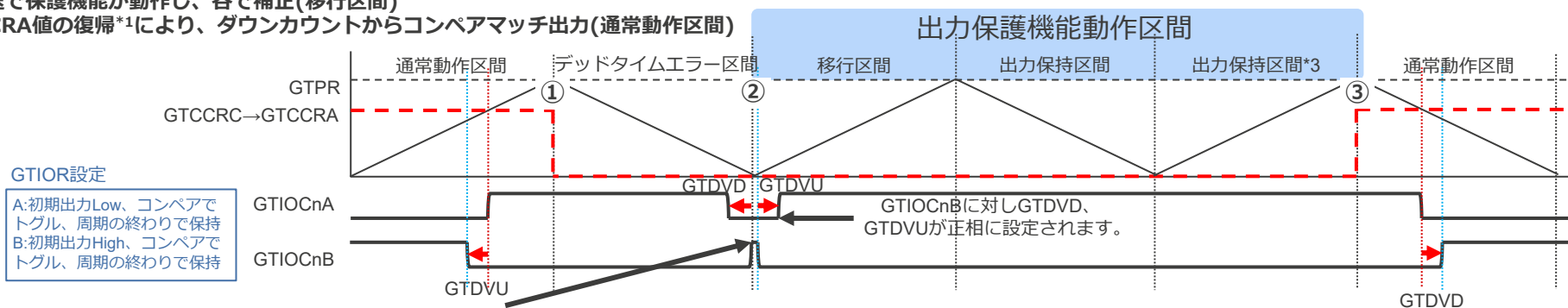
■ 谷転送でGTCCRA=0000hとした場合

- ①谷転送で保護機能が動作し、谷で補正(移行区間)
- ②GTCCRA値の復帰*1により、ダウンカウントからコンペアマッチ出力(通常動作区間)



■ 山転送でGTCCRA=0000hとした場合

- ①山転送でデッドタイムエラーが発生
- ②谷転送で保護機能が動作し、谷で補正(移行区間)
- ③GTCCRA値の復帰*1により、ダウンカウントからコンペアマッチ出力(通常動作区間)



谷で変化した後、谷から1カウント後に変化する為、1カウントクロック幅の波形を出力します。

*1:復帰条件は、 $0 < GTCCRAレジスタ < GTPRレジスタ$ 、カウントストップ、GTDTCR.TDEビット = 0の何れかになります。

*2:山で復帰した場合、復帰区間ではなく、出力保持区間となります。

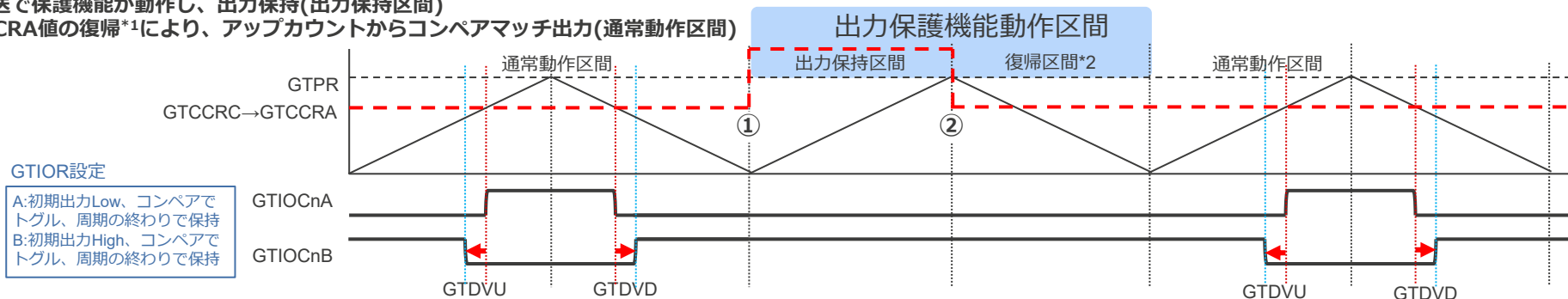
*3:谷で復帰した場合、出力保持区間ではなく、復帰区間となります。

三角波PWMモード1/2/3における出力保護機能動作($GTCCRA \geq GTPR$ の場合) (デッドタイム自動設定機能におけるコンペア値設定のご注意)

$GTCCRA \geq GTPR$ を設定した場合、出力保護機能により以下のPWM出力波形に整形します。

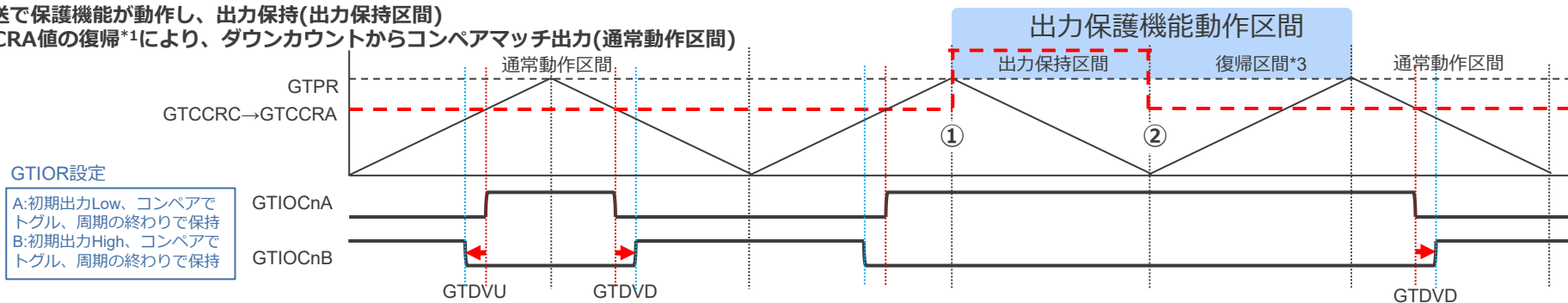
■ 谷転送で $GTCCRA \geq GTPR$ とした場合

- ① 谷転送で保護機能が動作し、出力保持(出力保持区間)
- ② $GTCCRA$ 値の復帰*1により、アップカウントからコンペアマッチ出力(通常動作区間)



■ 山転送で $GTCCRA \geq GTPR$ とした場合

- ① 山転送で保護機能が動作し、出力保持(出力保持区間)
- ② $GTCCRA$ 値の復帰*1により、ダウンカウントからコンペアマッチ出力(通常動作区間)

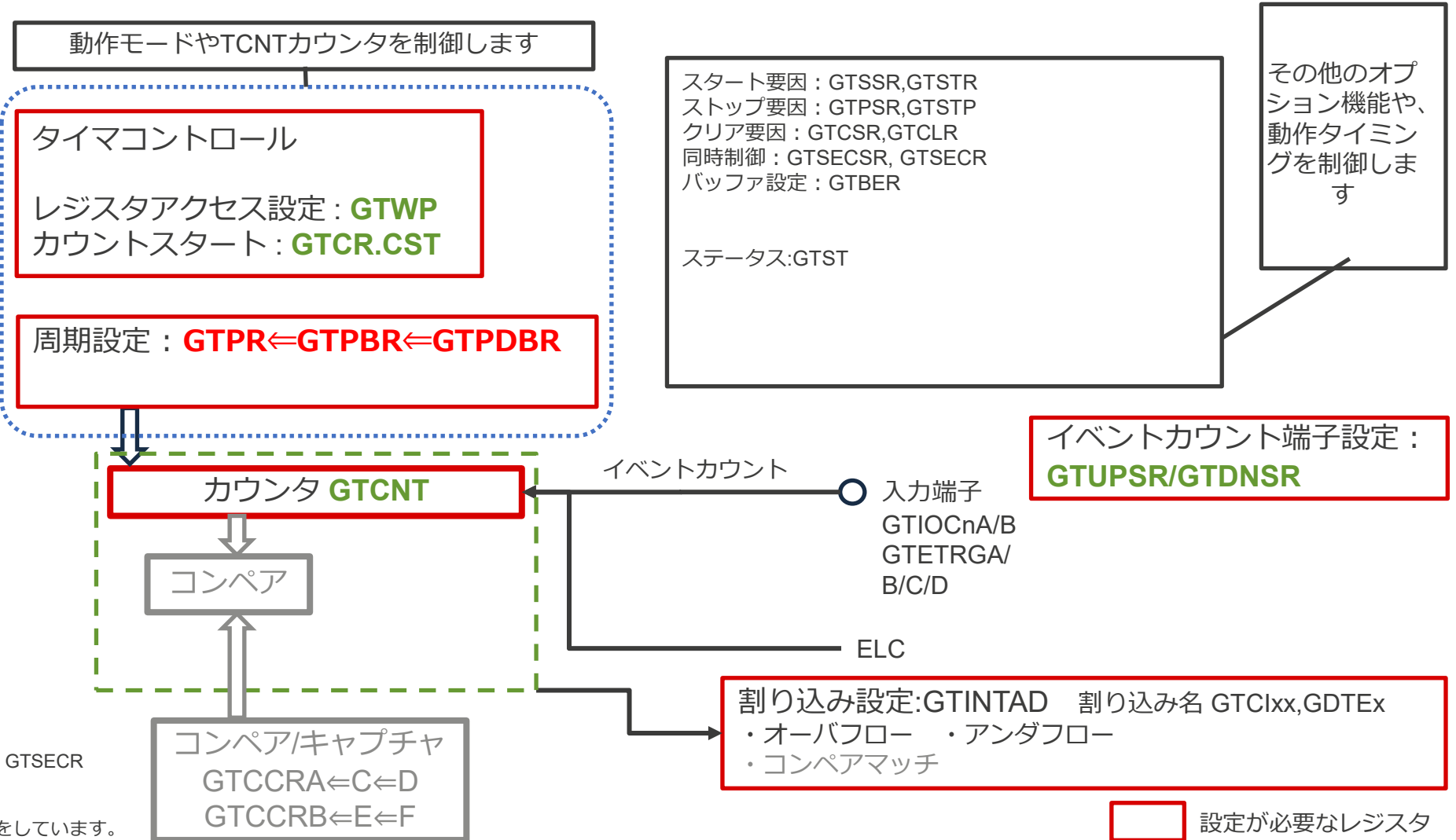


*1:復帰条件は、 $0 < GTCCRA$ レジスタ $< GTPR$ レジスタ、カウントストップ、 $GTDCR.TDE$ ビット = 0の何れかになります。

*2:谷で復帰した場合、復帰区間ではなく、出力保持区間となります。

*3:山で復帰した場合、復帰区間ではなく、出力保持区間となります。

イベントカウント動作 レジスタ一覧



注1: 各レジスタはチャンネル毎にあります。
チャンネル間の同期制御についてはGTSECSR、GTSECR
で同期をとります。

注2: 副動作(アウトプットコンペア)を除いた記載をしています。

イベントカウント動作 レジスタ設定

設定の流れ



設定の流れ		イベントカウント	位相係数機能
1	カウント アップ・ダウン要因 の設定 GTUPSR, GTDNSR	<ul style="list-style-type: none"> ● カウントアップ時 : GTUPSRを設定 ● カウントダウン時 : GTDNSRを設定 ● カウントアップ・ダウン時 : GTUPSR, GTDNSRを設定 ● カウント要因 <ul style="list-style-type: none"> ・ GTETRGA~Dのエッジ : 立上り=01 立下り=10 両エッジ=11 ・ GTIOcNA立上りエッジ : GTIOcNBがLowの時=01 GTIOcNBがHighの時=10 GTIOcNBに関係なく=11 ・ GTIOcNA立下りエッジ : GTIOcNBがLowの時=01 GTIOcNBがHighの時=10 GTIOcNBに関係なく=11 ・ GTIOcNB立上りエッジ : GTIOcNAがLowの時=01 GTIOcNAがHighの時=10 GTIOcNAに関係なく=11 ・ GTIOcNB立下りエッジ : GTIOcNAがLowの時=01 GTIOcNAがHighの時=10 GTIOcNAに関係なく=11 ・ ELC A~Hイベント : ELCイベントによるカウント 	<ul style="list-style-type: none"> ● 位相計数機能 : GTUPSR, GTDNSRを設定 ● カウント要因 <ul style="list-style-type: none"> ・ モード1 : GTUPSR=6900h, GTDNSR=9600h GTUPSR : GTIOcNA立上りエッジ=01 GTIOcNA立下りエッジ=10 GTIOcNB立上りエッジ=10 GTIOcNB立下りエッジ=01 GTDNSR : GTIOcNA立上りエッジ=10 GTIOcNA立下りエッジ=01 GTIOcNB立上りエッジ=01 GTIOcNB立下りエッジ=10 ・ モード2 : GTUPSR=0800h, GTDNSR=0400h or GTUPSR=0200h, GTDNSR=0100h ・ モード3 : GTUPSR=0800h, GTDNSR=8000h or GTUPSR=0200h, GTDNSR=2000h or GTUPSR=0A00h, GTDNSR=A000h ・ モード4 : GTUPSR=0600h, GTDNSR=9000h ・ モード5 : GTUPSR=0800h, GTDNSR=0400h <p>詳細のカウントアップ、ダウン条件は、 「イベントカウント動作 位相計数機能 (GTIOcNA / GTIOcNB)」を参照</p>
2	周期設定 GTPR	✓	✓
3	カウンタ初期値設定 GTCNT	✓	✓
4	カウントスタート GTCE.CST=1	✓	✓

イベントカウント動作

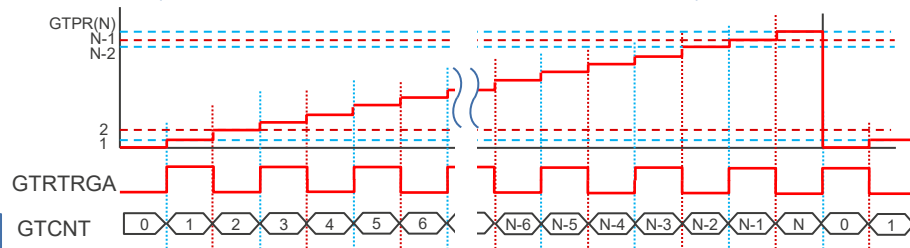
イベントカウント (ELCイベント OR 外部トリガ (GTETRGA~D. GTIOCNA / GTIOCNB))

GTUPSR/GTDBSRで設定したイベントでカウントアップ/ダウンするモードです。内部クロックはGTCR.TPCS[3:0]で選択するカウントクロックに同期して動作します。GTCR. MD[2:0]で選択する動作モード、GTCR.TPCS[3:0]で選択するカウントクロック、GTUDDTYC.UDで選択するカウント方向の設定による動作は無効となります。

- ・周期 : GTPR
- ・イベント : GTUPSR(アップカウントイベント用)で選択するイベント
- ・カウント方向 : アップカウント (GTUPSRはアップカウントイベントを設定する為)

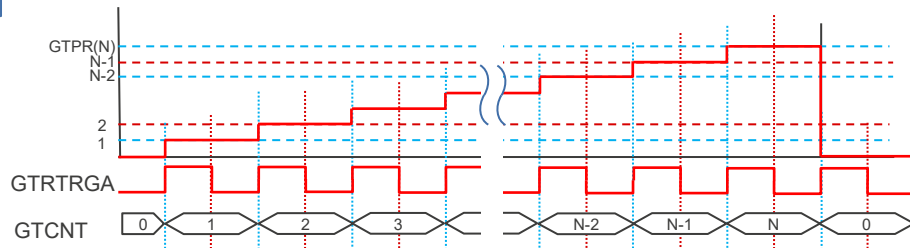
GTETRGAの
両エッジ選択時

GTUPSR.USGTRGAR=1,
GTUPSR.USGTRGAF=1



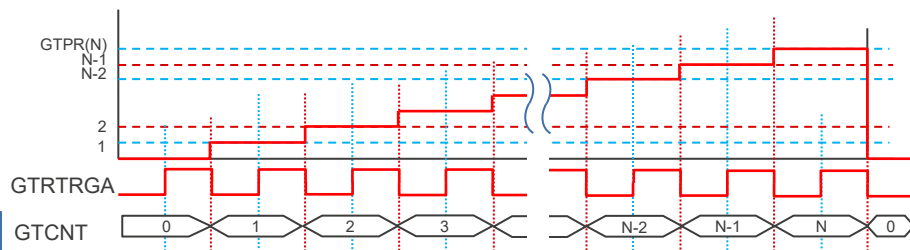
GTETRGAの
Riseエッジ選択時

GTUPSR.USGTRGAR=0,
GTUPSR.USGTRGAF=1



GTETRGAの
Fallエッジ選択時

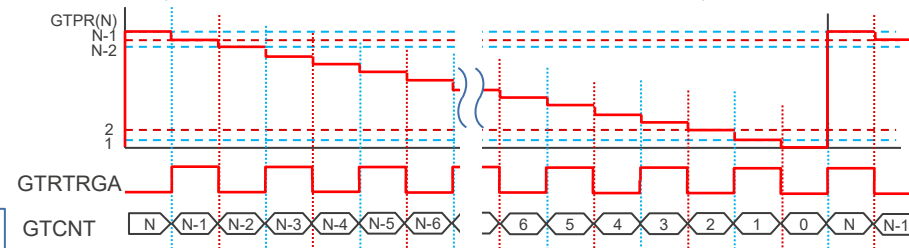
GTUPSR.USGTRGAR=1,
GTUPSR.USGTRGAF=0



- ・周期 : GTPR
- ・イベント : GTDNSR (ダウンカウントイベント用)で選択するイベント
- ・カウント方向 : ダウンカウント (GTDNSRはダウンカウントイベントを設定する為)

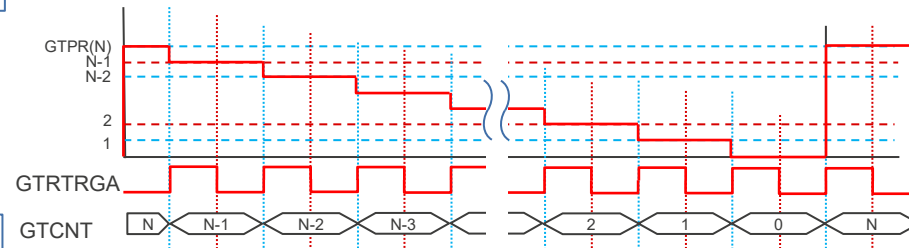
GTETRGAの
両エッジ選択時

GTDNSR.DSGTRGAR=1,
GTDNSR.DSGTRGAF=1



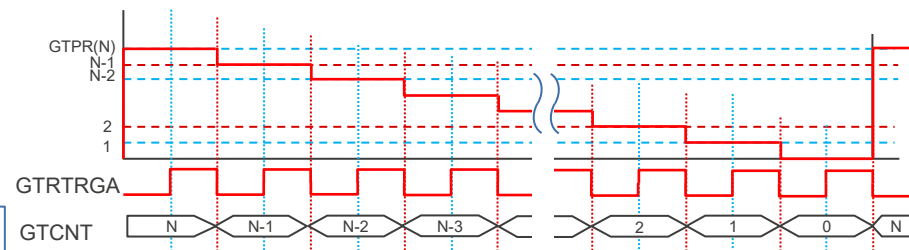
GTETRGAの
Riseエッジ選択時

GTDNSR.DSGTRGAR=0,
GTDNSR.DSGTRGAF=1



GTETRGAの
Fallエッジ選択時

GTDNSR.DSGTRGAR=1,
GTDNSR.DSGTRGAF=0



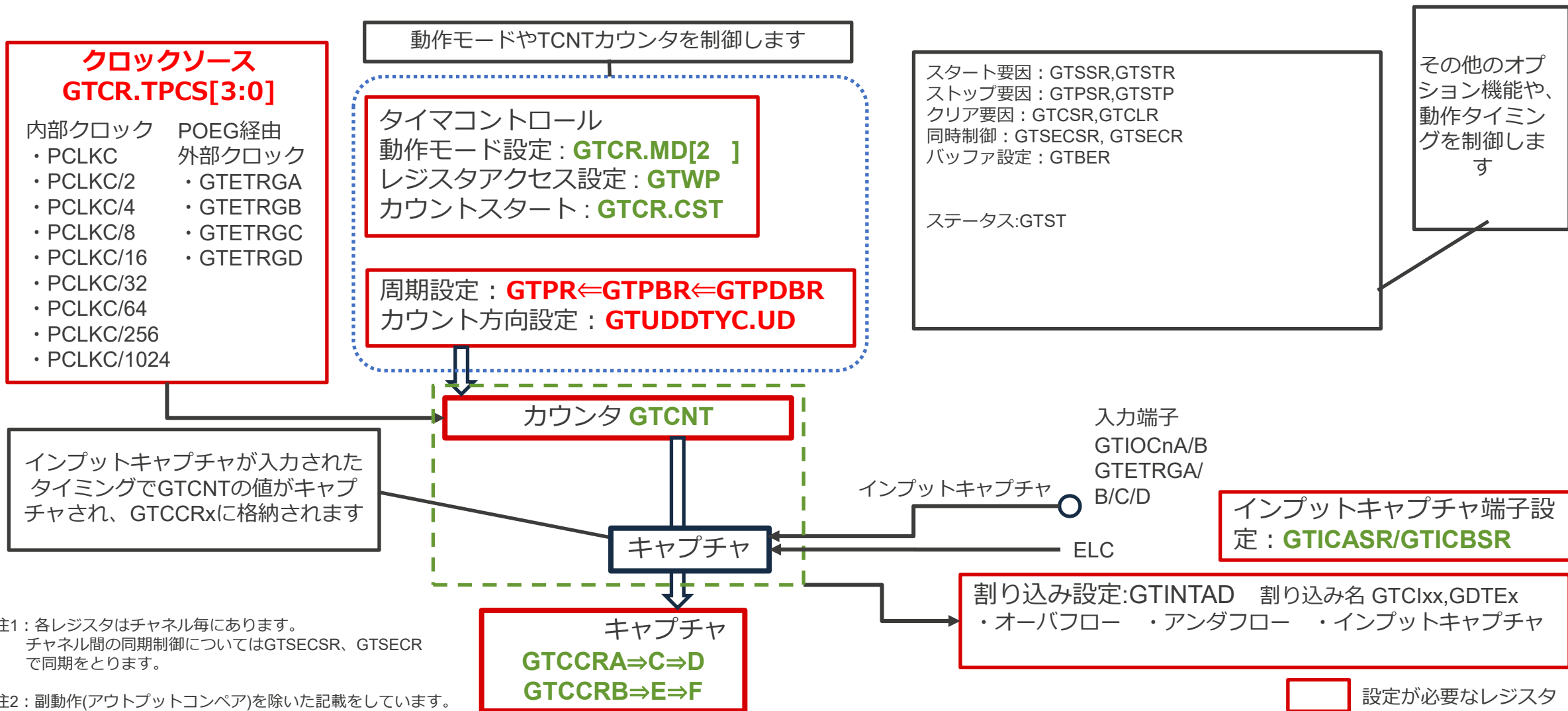
イベントカウント動作

位相計数機能 (GTIOCNA / GTIOCNB)

- パルス列入力/2相エンコーダの計測に最適な位相係数モードを搭載
 - 位相差のある2本のパルス信号(A相信号をGTIOCnA端子、B相信号をGTIOCnB端子)を入力し、エッジ数をカウント（加算/減算）します。
 - 位相計数機能は、5モード有り、様々なパルス入力およびエンコーダ入力に対応しています。

カウント方向	モード1		モード2		モード3		モード4		モード5	
	カウント条件	カウントエッジ	カウント条件	カウントエッジ	カウント条件	カウントエッジ	カウント条件	カウントエッジ	カウント条件	カウントエッジ
加算 (正回転)	位相 A相 > B相	A相、B相 各々の 立上り/立下り	B相がHigh	A相の立下り	B相がHigh	A相の立下り	位相 A相 > B相	B相の 立上り/立下り	A相がHigh/Low	B相の立下り
				A相の立上り		A相の立上り			B相がHigh/Low	A相の立下り
			A相の両エッジ		A相の両エッジ					
減算 (逆回転)	位相 A相 < B相	A相、B相 各々の 立上り/立下り	B相がLow	A相の立下り	A相がHigh	B相の立下り	位相 A相 < B相	B相の 立上り/立下り	なし	なし
				A相の立上り		B相の立上り				
			A相の両エッジ		B相の両エッジ					
									なし	

インプットキャプチャ動作時 レジスタ関連



インプットキャプチャ動作 レジスタ設定

設定の流れ



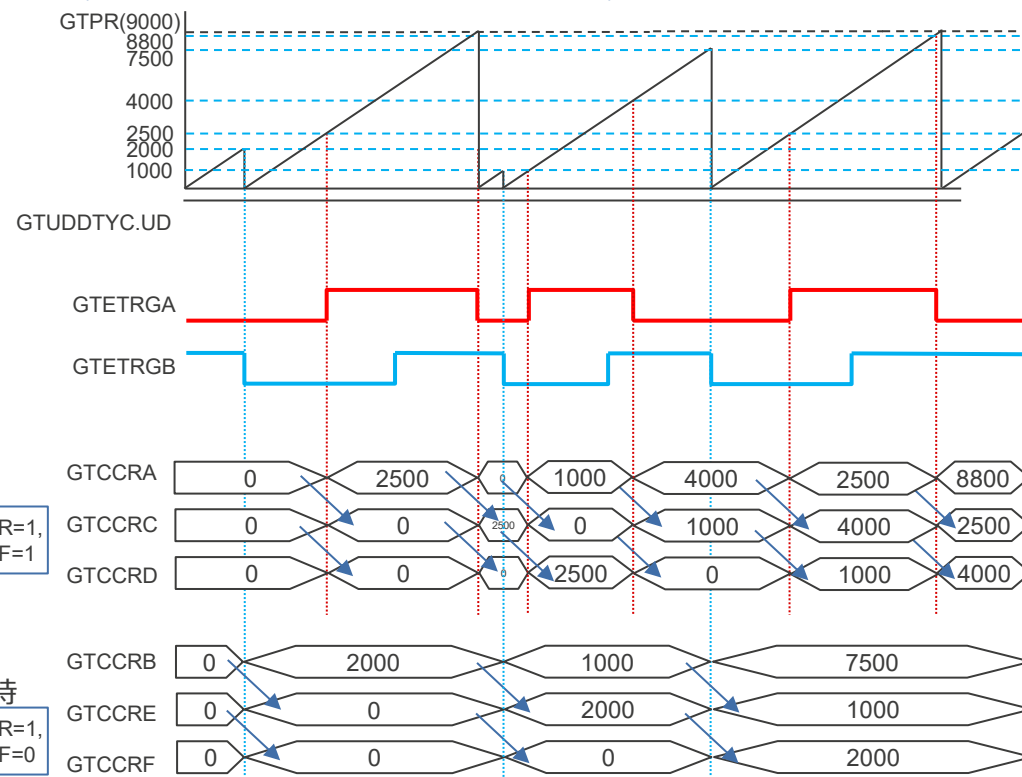
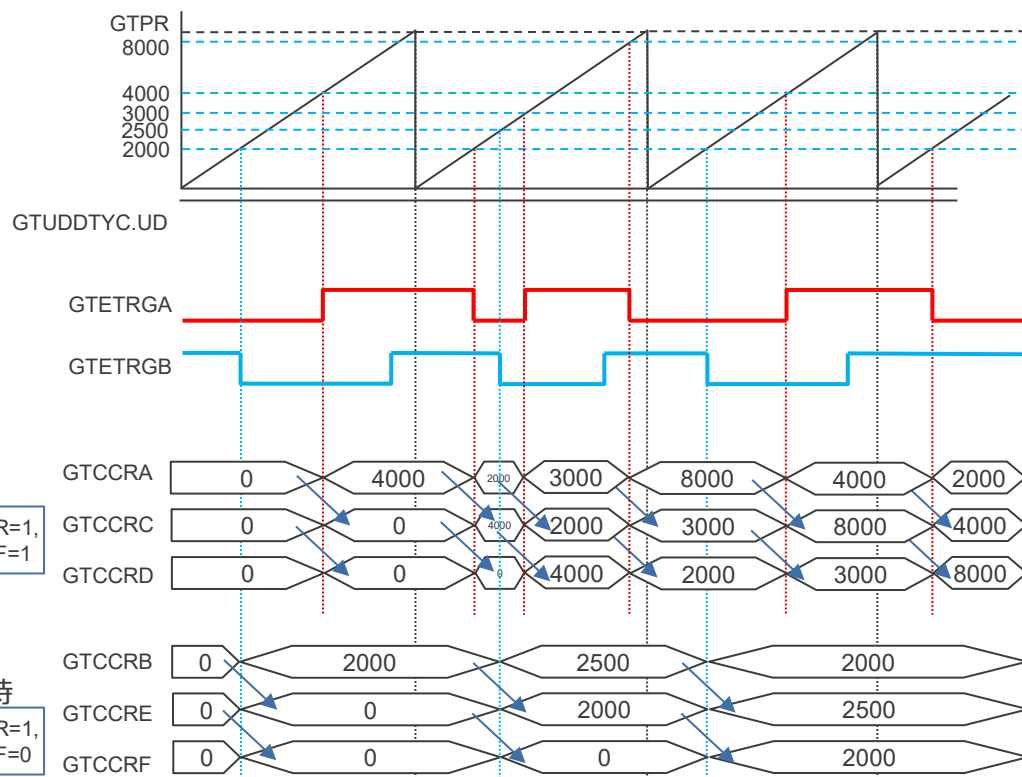
設定の流れ		インプットキャプチャ
1	動作モード設定 GTCR.MD[2:0]	GTCR.MD[2]のみ有効 (のこぎり波か三角波のカウントによるインプットキャプチャ動作) になります。
2	カウント方向設定 GTUDDTYC.UD	のこぎり波の場合、GTUDDTYC.UDでカウントアップかダウンを選択します。 三角波の場合、GTUDDTYC.UDは、カウントスタート時のみ有効となります。これは、谷からスタート(アップカウント)させるか、山からスタート(ダウンカウント)させるかの設定のみ有効となります。
3	カウントクロック設定 GTCR.TPCS[3:0]	✓
4	周期設定 GTPR	✓
5	カウンタ初期値設定 GTCNT	✓
6	インプットキャプチャ要因設定 GTICASR, GTICBSR	<ul style="list-style-type: none"> ● GTCCRALレジスタへのインプットキャプチャ：GTICASRを設定 ● GTCCRBレジスタへのインプットキャプチャ：GTICBSRを設定 ● GTICASR, GTICBSRのインプットキャプチャ要因 <ul style="list-style-type: none"> ・ GTETRGA~Dのエッジ <ul style="list-style-type: none"> : 立上り=01 立下り=10 両エッジ=11 ・ GTIOcNA立上りエッジ <ul style="list-style-type: none"> : GTIOcNBがLowの時=01 GTIOcNBがHighの時=10 GTIOcNBに関係なく=11 ・ GTIOcNA立下りエッジ <ul style="list-style-type: none"> : GTIOcNBがLowの時=01 GTIOcNBがHighの時=10 GTIOcNBに関係なく=11 ・ GTIOcNB立上りエッジ <ul style="list-style-type: none"> : GTIOcNAがLowの時=01 GTIOcNAがHighの時=10 GTIOcNAに関係なく=11 ・ GTIOcNB立下りエッジ <ul style="list-style-type: none"> : GTIOcNAがLowの時=01 GTIOcNAがHighの時=10 GTIOcNAに関係なく=11 ・ ELC A~Hイベント <ul style="list-style-type: none"> : ELCイベントによるインプットキャプチャ
7	カウントスタート GTCE.CST=1	✓

インプットキャプチャ動作

- GTICASR/GTICBSRで設定したイベントでインプットキャプチャするモードです。インプットキャプチャモードでは、GTCR.MDビットで設定したのこぎり波カウントか三角波カウントによるインプットキャプチャ動作になります。GTICASR/GTICBSR=0で GTIORを設定するとアウトプットコンペアになり、インプットキャプチャには使用出来ません。

- ・周期 : GTPR
- ・イベント : GTICASR/GTICBSRで選択するイベント
- ・カウント方向 : GTCR.MD[2]がのこぎり波の場合、GTUDDTYC.UDで設定したカウント方向 (GTCR.MD[2]が三角波の場合、アップ/ダウンカウントとなります。)

- ・周期 : GTPR
- ・イベント : GTICASR/GTICBSRで選択するイベント
- ・カウント方向 : GTCR.MD[2]がのこぎり波の場合、GTUDDTYC.UDで設定したカウント方向
- ・クリア設定 : カウンタクリア要因にインプットキャプチャ要因Bと同様の要因を設定 (GTCR.CSGTRGBR=1, GTCR.CSGTRGBF=0)



ご注意事項

- GTCNT更新の優先順位は以下になります。
CPU書き込み(GTCNTカウンタ書き込み / GTCLRレジスタ書き込み) > GTCRレジスタで設定されたハードウェア要因によるクリア > GTUPSR, GTDNSRレジスタで設定されたハードウェア要因によるカウントアップ / カウントダウン > カウント動作
- GTUPSR レジスタによるカウントアップとGTDNSR レジスタによるカウントダウンが競合した場合、GTCNTは更新されません。
- GTCNTの更新とCPU 読み出しが競合した場合、更新前のGTCNTデータが読み出されます。

- GTCCRA~F レジスタへの書き込みとインプットキャプチャ/ バッファ転送が競合した場合、インプットキャプチャ/ バッファ転送よりGTCCRA~F レジスタへの書き込みが優先されます。
- インプットキャプチャとカウンタへのCPU 書き込みまたはハードウェア要因によるカウンタの更新が競合した場合、更新前のカウンタの値がキャプチャされます。
- GTCCRA~F レジスタの更新とCPU 読み出しが競合した場合、更新前のGTCCRA~Fデータが読み出されます。

- バッファ転送とGTPR レジスタへの書き込みが競合した場合、バッファ転送よりGTPR レジスタへの書き込みが優先されます。
- GTPR レジスタの更新とCPU 読み出しが競合した場合、更新前のGTPRデータが読み出されます。

- バッファ転送とGTADTRm レジスタへの書き込みが競合した場合、バッファ転送よりGTADTRm レジスタへの書き込みが優先されます。
- GTADTRA/B レジスタの更新とCPU 読み出しが競合した場合、更新前のGTADTRA/Bデータが読み出されます。

- バッファ転送とGTDVm レジスタへの書き込みが競合した場合、バッファ転送よりGTDVm レジスタへの書き込みが優先されます。
- GTDVU/D レジスタの更新とCPU 読み出しが競合した場合、更新前のGTDVU/Dデータが読み出されます。

- GTSSR, GTPSR レジスタで設定したハードウェア要因によるスタート/ストップとCPU 書き込み(GTCR レジスタ書き込み/GTSTR レジスタ書き込み/GTSTP レジスタ書き込み) が競合した場合、CPU 書き込みが優先されます。
- GTSSR レジスタで設定したハードウェア要因によるスタートとGTPSR レジスタで設定したハードウェア要因によるストップが競合した場合、CST ビットの状態は変わりません。
- CST ビットの更新とCPU 読み出し(GTCR レジスタ読み出し/GTSTR レジスタ読み出し/GTSTP レジスタ読み出し) が競合した場合、更新前のデータが読み出されます。

GPTを使用したアプリケーションノート一覧

GPTを使用したアプリケーションノートを多数準備しています。より詳細な使用方法は下記をご参照ください

- RXファミリ MTU3/GPTWを用いたPWM出力方法 [R01AN5995](#)
- RXファミリ MTU3/GPTWを用いた相補PWMの0%、100%近傍出力動作例 [R01AN6539](#)
- RXファミリ MTU3/GPTWを用いたA/D変換開始要求ディレイド機能使用例 [R01AN6643](#)
- RXファミリ MTU3/GPTWを用いた位相計数モード使用例 [R01AN6387](#)
- RXファミリ MTU3/GPTWを用いた同期動作 [R01AN6282](#)
- GPT 三相のこぎり波相補 PWM [R01AN0818](#)
- GPT 非対称三角波相補 PWM [R01AN0750](#)

GPTを使用したアプリケーションノート一覧

モータ・電源向け

GPTを使用したアプリケーションノートを多数準備しています。より詳細な使用方法は下記をご参照ください

- 永久磁石同期モータのセンサレスベクトル制御 - MCK 用 [R01AN6858](#)
- 永久磁石同期モータのセンサレスベクトル制御 RX66T 実装編 [R01AN4244](#)
- 永久磁石同期モータのセンサレスベクトル制御 RX72T 実装編 [R01AN4720](#)
- 永久磁石同期モータのセンサレスベクトル制御 (4 モータ制御) RX72T 実装編 [R01AN5980](#)
- 永久磁石同期モータのセンサレスベクトル制御 Evaluation System for BLDC Motor 用 [R01AN6307](#)

- 永久磁石同期モータのエンコーダベクトル制御 - MCK 用 [R01AN6857](#)
- 永久磁石同期モータのエンコーダベクトル制御 RX66T 実装編 [R01AN4245](#)
- 永久磁石同期モータのエンコーダベクトル制御 RX72T 実装編 [R01AN5499](#)
- 永久磁石同期モータのエンコーダベクトル制御 (3 モータ制御) RX72T 実装編 [R01AN5981](#)

- 永久磁石同期モータの磁気センサ/誘導センサ ベクトル制御 [R01AN6317](#)
- 永久磁石同期モータのホール/センサレスベクトル制御 - 2 モータ駆動 [R01AN6068](#)
- サーボ制御向けサンプルプログラム [R01AN6911](#)

- SiC パワー素子を用いた 3 レベルインバータ電源制御 [R01AN6486](#)
- デジタル電力変換 (LLC 共振コンバータ (DC-DC コンバータ)) [R01AN7118](#)
- デジタル電力変換 (UPS (CCM インターリーブ PFC、チョップパ方式 DC-DC コンバータ)) [R01AN6465](#)

ウォッチドッグタイマ/ 独立ウォッチドッグタイマ (WDT/IWDT)

WDTとIWDTの機能比較

WDT

IWDT

WDTとIWDTの機能差分を以下に示します。

青文字箇所：機能差分














項目	WDT	IWDT
カウントソース	周辺モジュールクロック	IWDT専用クロック
クロック分周比	4/64/128/512/2048/8192	1/16/32/64/128/256
カウント開始条件	<ul style="list-style-type: none">・オートスタートモード：リセット解除後、自動的にカウント開始・レジスタスタートモード：リフレッシュ動作により、カウント開始	<ul style="list-style-type: none">・オートスタートモード：リセット解除後、自動的にカウント開始・レジスタスタートモード：リフレッシュ動作により、カウント開始
カウント停止条件	<ul style="list-style-type: none">・リセット・低消費電力状態・アンダフロー、リフレッシュエラー発生時 (レジスタスタートモード時のみ)	<ul style="list-style-type: none">・リセット・低消費電力状態(レジスタ設定による)・アンダフロー、リフレッシュエラー発生時 (レジスタスタートモード時のみ)
ウィンドウ機能	あり	あり
リセット出力要因	<ul style="list-style-type: none">・ダウンカウンタがアンダフローしたとき・リフレッシュ許可期間以外でリフレッシュを行った場合 (リフレッシュエラー)	<ul style="list-style-type: none">・ダウンカウンタがアンダフローしたとき・リフレッシュ許可期間以外でリフレッシュを行った場合 (リフレッシュエラー)
ノンマスカブル割り込み/ 割り込み要因	<ul style="list-style-type: none">・ダウンカウンタがアンダフローしたとき・リフレッシュ許可期間以外でリフレッシュを行った場合 (リフレッシュエラー)	<ul style="list-style-type: none">・ダウンカウンタがアンダフローしたとき・リフレッシュ許可期間以外でリフレッシュを行った場合 (リフレッシュエラー)
カウンタ値の読み出し	可能	可能
イベントリンク機能	なし	あり
出力信号(内部信号)	<ul style="list-style-type: none">・リセット出力・割り込み要求出力	<ul style="list-style-type: none">・リセット出力・割り込み要求出力・スリープモードカウント停止制御出力
低消費電力モード時	停止	動作/停止 (選択可能)

WDT設定一覧

WDT

ウォッチドッグタイマの動作モードはオートスタートモードとレジスタスタートモードがあります。
各モードを動作させるためにはオプション設定メモリ(OFS)またはWDTのレジスタを設定する必要があります。
各動作モードで設定が必要なレジスタを以下に示します。

 : OFSの設定
 : WDTの設定

	オートスタートモード	レジスタスタートモード
開始モード設定	 オートスタートモード有効 OFS0.WDTSTRT	 レジスタスタートモード有効 OFS0.WDTSTRT
タイマ設定	 タイムアウト期間選択 OFS0.WDTPRS	 タイムアウト期間選択 WDTCR.TOPS
	 クロック分周比選択 OFS0.WDTCKS	 クロック分周比選択 WDTCR.CKS
ウィンドウモード設定	 ウィンドウ終了位置選択 OFS0.WDTRPES	 ウィンドウ終了位置選択 WDTCR.RPES
	 ウィンドウ開始位置選択 OFS0.WDTRPSS	 ウィンドウ開始位置選択 WDTCR.RPSS
暴走検知時の 割り込み出力設定	 リセット割り込み要求選択 OFS0.WDTRSTIRQS	 リセット割り込み要求選択 WDTCR.RSTIRQS
カウントスタート	リセット解除後に自動スタート	 リフレッシュ動作 WDT.WDTRR "00h"を書き込み、"FFh"を書き込む

IWDT設定一覧

IWDT

ウォッチドッグタイマの動作モードはオートスタートモードとレジスタスタートモードがあります。
各モードを動作させるためにはオプション設定メモリ(OFS)またはWDTのレジスタを設定する必要があります。
各動作モードで設定が必要なレジスタを以下に示します。

 : OFSの設定

 : IWDTの設定

オートスタートモード

レジスタスタートモード

開始モード設定

オートスタートモード有効
OFS0.IWDTSTRT

レジスタスタートモード有効
OFS0.IWDTSTRT

タイマ設定

タイムアウト期間選択
OFS0.IWDTTOPS[1:0]

タイムアウト期間選択
IWDTCR.TOPS[1:0]

クロック分周比選択
OFS0.IWDTCKS[3:0]

クロック分周比選択
IWDTCR.CKS[3:0]

ウィンドウモード設定

ウィンドウ終了位置選択
OFS0.IWDRPES[1:0]

ウィンドウ終了位置選択
IWDTCR.RPES[1:0]

ウィンドウ開始位置選択
OFS0.IWDRPSS[1:0]

ウィンドウ開始位置選択
IWDTCR.RPSS[1:0]

暴走検知時の
割り込み出力設定

リセット割り込み要求選択
OFS0.IWDRSTIRQS

リセット割り込み要求選択
IWDRCCR.RSTIRQS

スリープモード時の
カウント停止設定

スリープモードカウント停止制御選択
OFS0.IWDTSLCSTP

スリープモードカウント停止制御選択
IWDTCSSTPR.SLCSTP

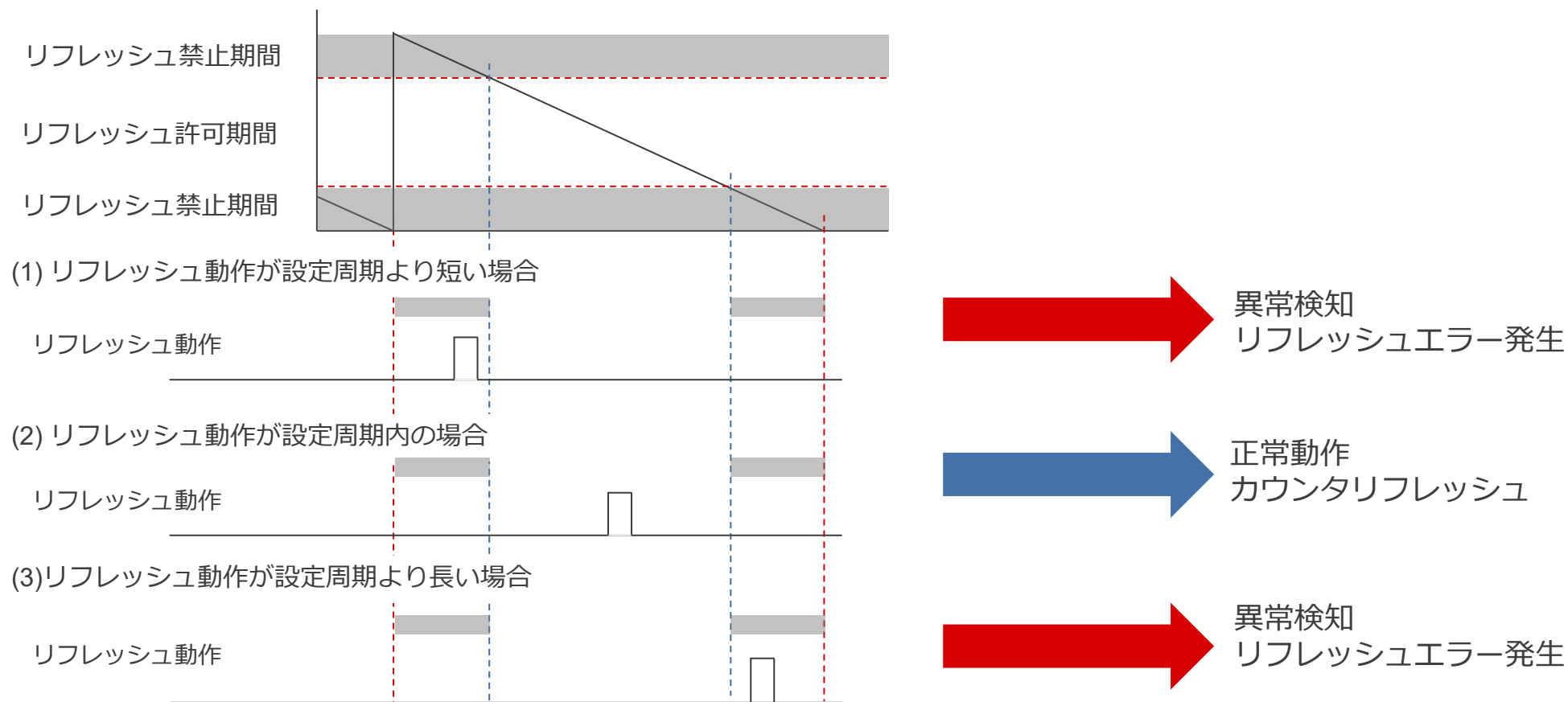
カウントスタート

リセット解除後に自動スタート

リフレッシュ動作
IWDT.IWDTRR
"00h"を書き込み、"FFh"を書き込む

機能説明：ウィンドウモード

ウィンドウモードは、リフレッシュ動作の許可期間と禁止期間を設定します。
禁止期間中にリフレッシュ動作が行われると、リフレッシュエラーが発生し、システムが暴走していると判断します。
CPUの暴走に加え、処理周期のズレも検知できるため、周期性や信頼性の要求が高い用途に適した機能です。



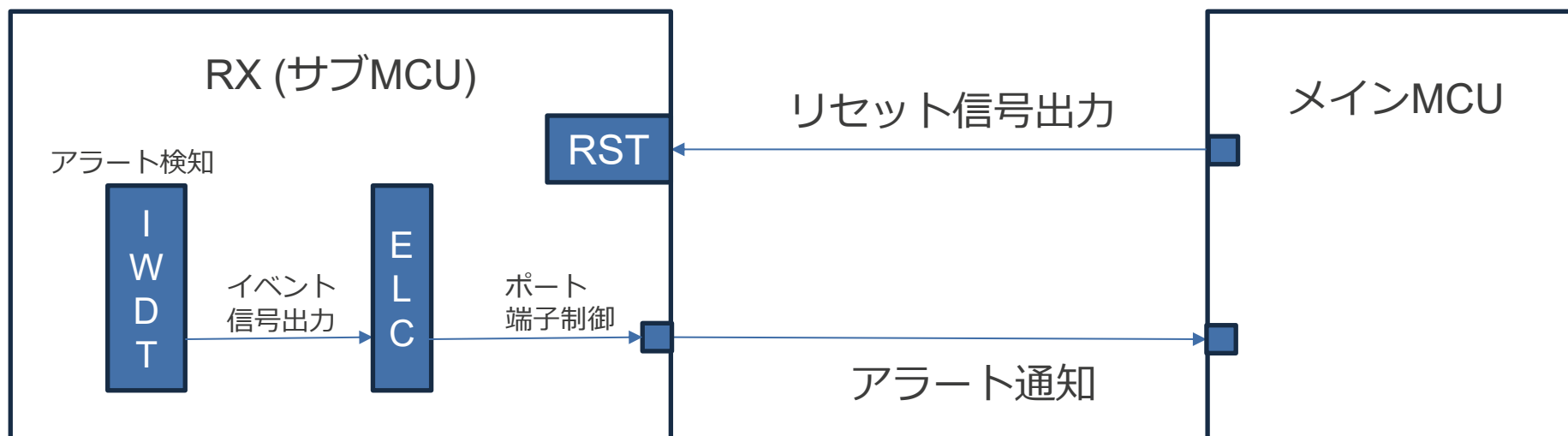
応用例：

イベントリンクコントローラ(ELC)を使用して外部のメインMCUへエラー通知を出力

IWDT

IWDTでは、リフレッシュエラーやアンダーフローが発生した際にELCへの信号出力が可能で、CPUを介さずにELCを使用してI/Oポートから外部信号を出力できます。

下図はIWDTでアラート検知をし、外部のメインMCUにアラート通知を送り、メインMCUからハードウェアリセットをかける方法です。メインMCUは、サブMCUの状態の確認やアラート発生回数の把握が可能となります。



※ELCを使用する場合、ノンマスクブル割り込み要求または割り込み要求出力を許可（IWDTRCR.RSTIRQS = 0）する必要があります。

注意事項

WDT

IWDT

- WDT, IWDTを動作させるとカウント停止条件以外での停止はできません。停止条件は以下の通りです。
 - リセット
 - 低消費電力状態(IWDTはレジスタの設定による)
 - アンダフロー、リフレッシュエラー発生時(レジスタスタートモード時のみ)
- エミュレータでデバッグする場合、ブレーク中はWDT、IWDTのカウントが停止します。停止したカウントはプログラム実行時に再開します。

改訂履歴

Revision	Date	Contents
1.00	2024/12	初版発行
2.00	2026/2	機能追加 ・消費電力低減機能 ・割り込みコントローラ ・外部バスコントローラ ・汎用PWMタイマ

