

RZ/N1D グループ、 RZ/N1S グループ、 RZ/N1L グループ

ユーザーズマニュアル PWMTimer

RZ ファミリ

RZ/N シリーズ

本資料に記載の全ての情報は本資料発行時点のものであり、ルネサス エレクトロニクスは、予告なしに、本資料に記載した製品または仕様を変更することがあります。
ルネサス エレクトロニクスのホームページなどにより公開される最新情報をご確認ください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
 2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 4. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、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。
 7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。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。
 8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
 11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.4.0-1 2017.11)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

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

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。

製品ご使用上の注意事項

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

1. 静電気対策

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

2. 電源投入時の処置

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

3. 電源オフ時における入力信号

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

4. 未使用端子の処理

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

5. クロックについて

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

6. 入力端子の印加波形

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

7. リザーブアドレス（予約領域）のアクセス禁止

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

8. 製品間の相違について

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

このマニュアルの使い方

1. 目的と対象者

このマニュアルは、本マイコンのハードウェア機能と電気的特性をユーザに理解していただくためのマニュアルです。本マイコンを用いた応用システムを設計するユーザを対象にしています。このマニュアルを使用するには、電気回路、論理回路、マイクロコンピュータに関する基本的な知識が必要です。

このマニュアルは、大きく分類すると、製品の概要、CPU、システム制御機能、周辺機能、電気的特性、使用上の注意で構成されています。

本マイコンは、注意事項を十分確認の上、使用してください。注意事項は、各章の本文中、各章の最後、注意事項の章に記載しています。

改訂記録は旧版の記載内容に対して訂正または追加した主な箇所をまとめたものです。改訂内容すべてを記録したものではありません。詳細は、このマニュアルの本文でご確認ください。

RZ/N1 グループでは次のドキュメントを用意しています。

■ RZ/N1 関連ドキュメント一覧

ドキュメント名	資料番号
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ データシート	R01DS0323JJ****
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ ユーザーズマニュアル アーキテクチャ・製品データ編	R01UH0750JJ****
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ ユーザーズマニュアル システム・周辺機能 1 編	R01UH0751JJ****
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ ユーザーズマニュアル 周辺機能 2 編	R01UH0752JJ****
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ ユーザーズマニュアル R-IN Engine・イーサーネット編	R01UH0753JJ****
RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ ユーザーズマニュアル PWMTimer	R01UH0913JJ**** (本ユーザーズマニュアル)

2. レジスタの表記

各章において「レジスタの説明」には、ビットの並びを示すビット配置図とビットに設定する内容を説明するビット機能表があります。使用する記号、用語を以下に説明します。

X.X.X [レジスタ名]

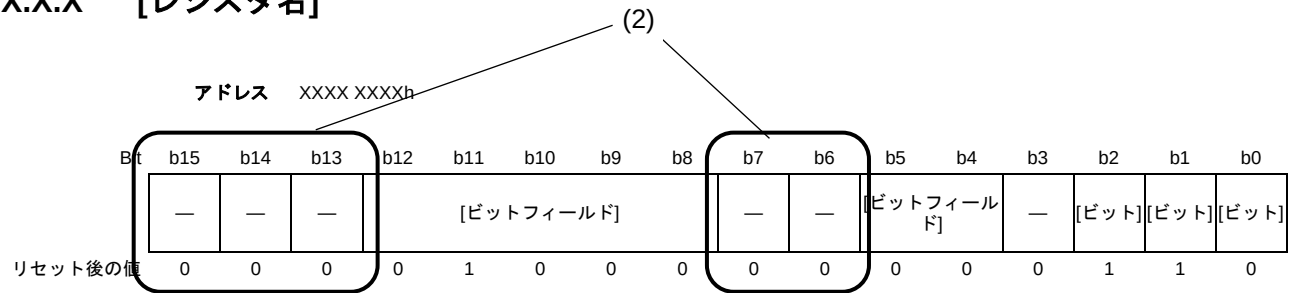


表 X.X [レジスタ名]レジスタの内容

ビット位置	ビット名	機能	R/W
b12~b8	[ビットフィールド]	[説明]	R/W
b5~b4	[ビットフィールド]	[説明] 2'b00 : Hi-Z 2'b01 : L 出力 上記以外 : 設定禁止	R/W
b2	[ビット]	[説明] 1'b0 : Hi-Z 1'b1 : 出力 (デフォルト値)	R/W
b1	[ビット]	[説明] 1'b0 : Hi-Z 1'b1 : 出力 (デフォルト値)	R/W
b0	[ビット]	[説明] 1'b0 : Hi-Z 1'b1 : 出力	R/W

- (1) R/W : 読み出し／書き込みとも有効です。
R/(W) : 読み出し／書き込みとも有効ですが、書き込みには制限があります。
制限の内容については、各レジスタの説明や注記を参照ください。
- R : 読み出しのみ有効です。書き込みは無効になります。
- W : 書き込みのみ有効です。読み出した値は保証されません。
- (2) 予約ビットです。書き込みを行う場合には、指定された値を書き込んでください。指定以外の値を書き込んだ場合の動作は保証されません。
- (3) 設定禁止。設定した場合の動作は保証されません。

3. 略語および略称の説明

略語／略称	フルスペル
AHB	Arm Advanced High-performance Bus
APB	Arm Advanced Peripheral Bus
AXI	Arm Advanced eXtensible Interface
bps	bits per second
CA7	Arm Cortex-A7 module
CM3	Arm Cortex-M3 module
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
Hi-Z	High Impedance
HSR	High-availability Seamless Redundancy
HW-RTOS	Hard Ware Real Time OS
I/O	Input/Output
INTC	Interrupt Controller
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connect
NoC	Network-on-Chip
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
UART	Universal Asynchronous Receiver/Transmitter
OTP	One Time Programmable
PTP	Precision Time Protocol
PRP	Parallel Redundancy Protocol
SoC	System On Chip

4. アクセスサイズの表記

アクセスサイズ：

8 ビット＝バイト

16 ビット＝ハーフワード

32 ビット＝ワード

Arm は Arm Limited（またはその子会社）の EU およびその他地域における登録商標です。
全ての商標・登録商標は各所有者の財産です。

目 次

第 1 章	PWMTimer	10
1.1	概要	10
1.2	信号インタフェース	12
1.3	レジスタマップ	13
1.4	レジスタの説明	15
1.4.1	rPWMTimer_FILTER_IN_[s] – FilterIN モジュール制御 (s = 0, 10)	15
1.4.2	rPWMTimer_FILTER_IN_[s] – FilterIN モジュール制御 (s = 2, 12)	17
1.4.3	rPWMTimer_FORCE_INPUT_[n] – FilterIN モジュール固定出力 (n = 0~1)	18
1.4.4	rPWMTimer_OUTPUTCTRL_[n] – OutputCtrl モジュール制御 (n = 0~3)	19
1.4.5	rPWMTimer_OUTPUTCTRL_FORCE_OUT_0 – OutputCtrl モジュール固定出力	20
1.4.6	rPWMTimer_OUTPUTCTRL_RELEASE_OUT – OutputCtrl モジュール固定出力解除	21
1.4.7	rPWMTimer_EVENT_MANAGER_INT_STATUS – グローバル割り込みステータス	22
1.4.8	rPWMTimer_EVENT_MANAGER_INT_MASK – グローバル割り込みマスク	23
1.4.9	rPWMTimer_PWM_CMP1_MASK – PWM CMP1 フラグマスク	24
1.4.10	rPWMTimer_PWM_OVERFLOW_MASK – PWM オーバーフローフラグマスク	25
1.4.11	rPWMTimer_PWM_UNDERFLOW_MASK – PWM アンダーフローフラグマスク	26
1.4.12	rPWMTimer_HWSTORE_OR[r]_STORE_MASK – PWM OR[r]トリガフラグマスク (r = 1, 3)	27
1.4.13	rPWMTimer_PWM_MODE_[k] – Timebase モード制御 (k = 0~1)	28
1.4.14	rPWMTimer_PWM_CLOCK_[q]_[k] – PWM モジュールクロック入力制御 (q = 0~1, k = 0~1)	30
1.4.15	rPWMTimer_PWM_UPDOWN_[d]_[k] – PWM モジュール Up/Down 入力制御 (d = 0~3, k = 0~1)	31
1.4.16	rPWMTimer_PWM_TRIG0_[k] – PWM モジュール Trig0 入力制御 (k = 0~1)	32
1.4.17	rPWMTimer_PWM_TRIG1_[k] – PWM モジュール Trig1 入力制御 (k = 0~1)	33
1.4.18	rPWMTimer_PWM_OUTPUTGEN[d]_[k] – PWM モジュール出力生成制御 (d = 0~3, k = 0~1)	34
1.4.19	rPWMTimer_PWM_BLOCK_CLOCK_[k] – PWM モジュール Block_Clock 制御 (k = 0~1)	35
1.4.20	rPWMTimer_TIMEBASE_[i]_[k] – 16 ビット Timebase 現カウンタ値 (i = 0~7, k = 0~1)	36
1.4.21	rPWMTimer_TIMEBASE_[n*2][n*2+1]_[k] – 32 ビットカスケード Timebase 現カウンタ値 (n = 0~3, k = 0~1)	37
1.4.22	rPWMTimer_TIMEBASE_[(n+1)*2-1][(n+1)*2]_[k] – 32 ビットカスケード Timebase 現カウンタ値 (n = 0~2, k = 0~1)	38
1.4.23	rPWMTimer_SCALEVALUE_[i]_[k] – カスケード値設定 (i = 0~7, k = 0~1)	39
1.4.24	rPWMTimer_TIMEBASE_MUX_[q]_[k] – コンペア/キャプチャ入力選択制御 (q = 0~1, k = 0~1)	40
1.4.25	rPWMTimer_HWSTORE_CONF_[i]_[k] – PWM モジュール Hardware Store トリガ制御 (i	

= 0~7, k = 0~1)	41
1.4.26 rPWMTimer_HWSTORE_MUX_[i]_[k] – PWM モジュール Hardware Store トリガ選択制御 (i = 0~7, k = 0~1)	42
1.4.27 rPWMTimer_HWSTORE_LOCK_[k] – オプションレジスタロックおよびロックステータス (k = 0~1)	43
1.4.28 rPWMTimer_HWSTORE_UNLOCK_[k] – オプションレジスタロック解除 (k = 0~1) ..	45
1.4.29 rPWMTimer_OPTIONREG[r]_[i]_[k] – 16 ビット Timebase オプションレジスタ[r]値 (i = 0 ~7, r = 0~3, k = 0~1)	47
1.4.30 rPWMTimer_OPTIONREG[r]_[n*2][n*2+1]_[k] – 32 ビットカスケード Timebase オプショ ンレジスタ[r]値 (n = 0~3, r = 0~3, k = 0~1)	48
1.4.31 rPWMTimer_OPTIONREG[r]_[(n+1)*2-1][(n+1)*2]_[k] – 32 ビットカスケード Timebase オ プションレジスタ[r]値 (n = 0~2, r = 0~3, k = 0~1)	49
1.4.32 rPWMTimer_PWM_SOFTSTART – PWM モジュールカウントイネーブルスタート	50
1.4.33 rPWMTimer_PWM_SOFTSTOP – PWM モジュールカウントイネーブルストップ	51
1.4.34 rPWMTimer_PWM_SOFTUPDOWN_0 – PWM モジュール Up/Down 制御	52
1.4.35 rPWMTimer_PWM_SOFTRESET – Timebase リセットトリガ	54
1.4.36 rPWMTimer_PWM_CMP1_FLAG_POLARITY – PWM CMP1 フラグ設定	55
1.4.37 rPWMTimer_PWM_CMP1_FLAG – PWM CMP1 フラグ	56
1.4.38 rPWMTimer_PWM_MAX_OVERFLOW – PWM オーバーフローフラグ	57
1.4.39 rPWMTimer_PWM_MIN_UNDERFLOW – PWM アンダーフローフラグ	58
1.4.40 rPWMTimer_HWSTORE_OR[r]_STORE_FLAG – PWM OR[r]トリガフラグ (r = 1, 3) ..	59
1.4.41 rPWMTimer_ROUTING_IN_[s] – PWMTimer の IN 入力選択 (s = 0, 2)	61
1.4.42 rPWMTimer_ROUTING_OUT_[n] – PWMTimer の OUT 出力選択 (n = 0~3)	62
1.4.43 rPWMTimer_ROUTING_OUTCTRL_0 – OutputCtrl0 モジュールの入力選択	63
1.4.44 rPWMTimer_ROUTING_OUTCTRL_1 – OutputCtrl1 モジュールの入力選択	64
1.4.45 rPWMTimer_ROUTING_OUTCTRL_2 – OutputCtrl2 モジュールの入力選択	65
1.4.46 rPWMTimer_ROUTING_OUTCTRL_3 – OutputCtrl3 モジュールの入力選択	66
1.4.47 rPWMTimer_ROUTING_OUTCTRL_4 – OutputCtrl4 モジュールの入力選択	67
1.4.48 rPWMTimer_ROUTING_OUTCTRL_5 – OutputCtrl5 モジュールの入力選択	68
1.4.49 rPWMTimer_ROUTING_OUTCTRL_6 – OutputCtrl6 モジュールの入力選択	69
1.4.50 rPWMTimer_ROUTING_OUTCTRL_7 – OutputCtrl7 モジュールの入力選択	70
1.4.51 rPWMTimer_ROUTING_OUTCTRL_8 – OutputCtrl8 モジュールの入力選択	71
1.4.52 rPWMTimer_ROUTING_OUTCTRL_9 – OutputCtrl9 モジュールの入力選択	72
1.4.53 rPWMTimer_ROUTING_OUTCTRL_10 – OutputCtrl10 モジュールの入力選択	73
1.4.54 rPWMTimer_ROUTING_OUTCTRL_11 – OutputCtrl11 モジュールの入力選択	74
1.4.55 rPWMTimer_ROUTING_OUTCTRL_12 – OutputCtrl12 モジュールの入力選択	75
1.4.56 rPWMTimer_ROUTING_OUTCTRL_13 – OutputCtrl13 モジュールの入力選択	76
1.4.57 rPWMTimer_ROUTING_OUTCTRL_14 – OutputCtrl14 モジュールの入力選択	77
1.4.58 rPWMTimer_ROUTING_OUTCTRL_15 – OutputCtrl15 モジュールの入力選択	78
1.5 動作説明	79
1.5.1 PWMTimer Top 内部接続および選択	79
1.5.2 PWMTimer Core	80

1.5.3	PWMTimer Core 内部接続	81
1.5.3.1	PWM モジュール接続	81
1.5.3.2	OutputCtrl モジュール接続	81
1.5.4	FilterIN モジュール	82
1.5.4.1	PolarIn	83
1.5.4.2	Bounce	83
1.5.4.3	Force & Enable	84
1.5.5	OutputCtrl モジュール	85
1.5.6	PWM モジュール	86
1.5.6.1	クロックプリスケラ	87
1.5.6.2	シングルカウンタモジュール	87
1.5.6.3	シングルカウンタモジュールのカスケード	97
1.5.6.4	シングルカウンタモジュールのマスタカウンタ機能	98
1.5.7	Event Manager モジュール	99
1.5.8	PWMTimer プログラミング	100
1.5.8.1	共通な初期設定例	100
1.5.8.2	コンペアマッチによる PWM 出力	102
1.5.8.3	マスタカウンタ	103
1.5.8.4	インプットキャプチャ	104
1.5.8.5	パルス測定	105

第1章 PWMTimer

1.1 概要

PWMTimer は、キャプチャおよびコンペアマッチ動作が可能な 16 ビットカウンタを 16 個搭載しています。

PWMTimer 機能：

- 6 個のフィルタ入力：
 - － バウンスフィルタ
 - － 極性選択
 - － Force & Enable 機能
 - － 40 本の外部入力 (PWM_IN)
- 16 個の 16 ビットカウンタを搭載し、カスケードにより 32 ビットカウンタとして使用可能：
 - － 他のカウンタとの同期動作が可能
 - － イベントによる動作制御：
 - ◆ クロック：外部入力およびクロックプリスケアラ
 - ◆ Up/Down：Max/Min 検出
 - ◆ キャプチャトリガ：外部入力および Max/Min 検出
 - － 以下構成の PWM モジュールを 2 個搭載：
 - ◆ 8×16 ビットカウンタ
 - ◆ 2×10 ビットのクロックプリスケアラ
 - ◆ 32 ビットカスケードカウンタ (16 ビットカウンタ×2)
 - ◆ 各カウンタの機能構成：
 - (1) キャプチャレジスタ (最大 2 個)
 - (2) コンペアレジスタ
 - (3) バッファ転送のレジスタ
 - (4) レンジモード (最大値の設定レジスタ)
 - － 同期の波形生成：
 - ◆ のこぎり波
 - ◆ 三角波
 - － イベントカウント
- 16 個の出力制御：
 - － Force & Enable 機能
 - － 極性選択
 - － 20 本の外部出力 (PWM_OUT)
- 割り込み制御
- 動作周波数：100MHz

PWMTimer TOP で使用可能な端子：

- PWM_IN[39:0]
- PWM_OUT[19:0]

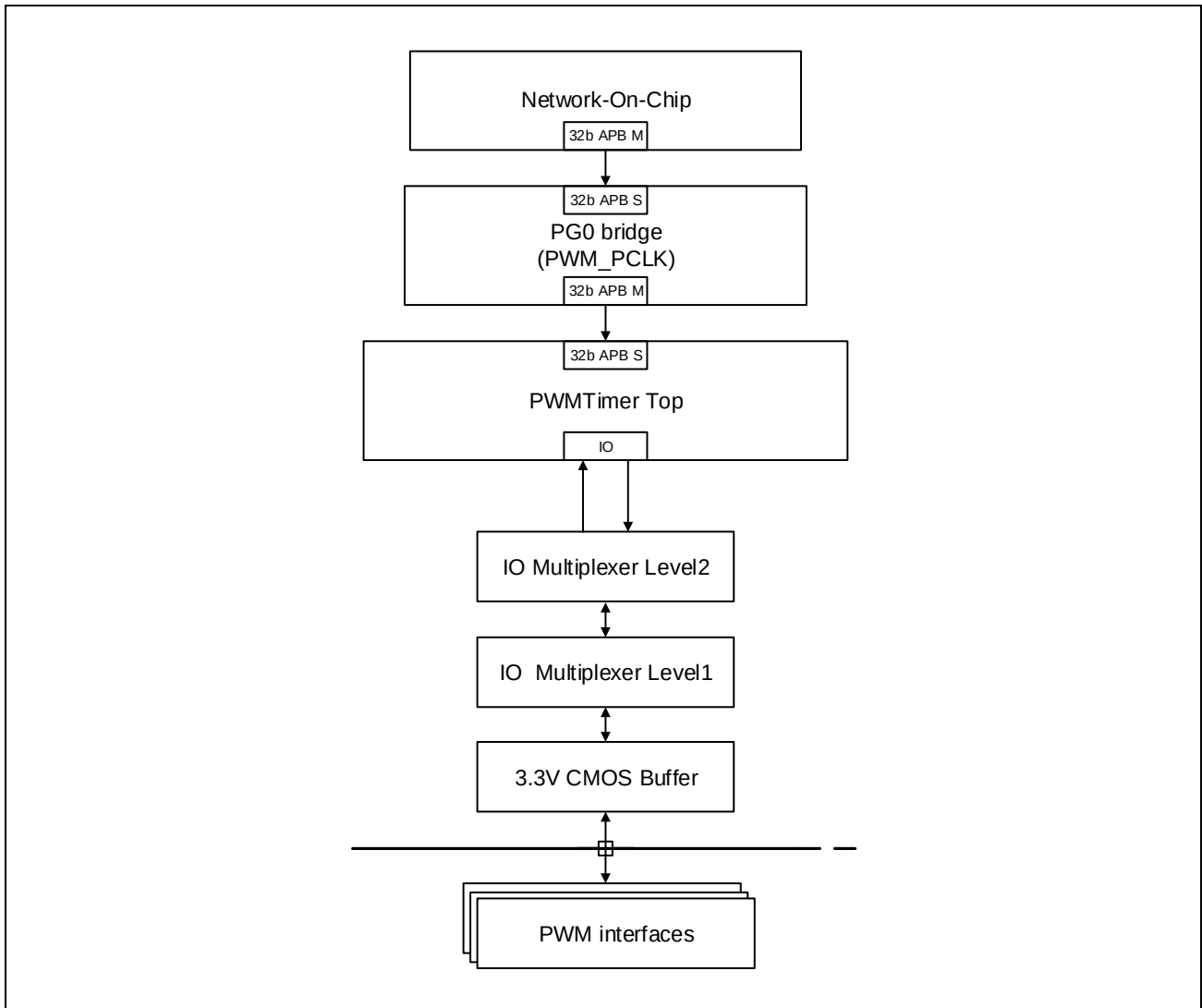


図 1.1 PWMTimer Top インタフェースおよび接続

1.2 信号インタフェース

表 1.1 信号インタフェース

信号名	入力 出力	説明
クロック		
PWM_PCLK	入力	内部バスクロック（APB）
割り込み		
PWM_Int	出力	レベル検出割り込み出力、アクティブ High
外部信号		
PWM_IN[39:0]	入力	PWMTimer Top 入力、PWMTimer Core 入力への任意の割り当て可能
PWM_OUT[19:0]	出力	PWMTimer Top 出力、PWMTimer Core 出力からの任意の割り当て可能

1.3 レジスタマップ

備 考

基本的に本項では、以下の添え字を使用します。

- k : PWM モジュール (k = 0~1)
- i : シングルカウンタモジュール (i = 0~7)
 - d : 2 個ずつのシングルカウンタモジュール (d = int(i/2), d = 0~3)
 - q : 4 個ずつのシングルカウンタモジュール (q = int(i/4), q = 0~1)
- r : オプションレジスタ : OR (r = 0~3)

表 1.2 レジスタマップ

アドレス	レジスタシンボル	レジスタ名
4006 8000h+2h×s	rPWMTimer_FILTER_IN_[s] (s = 0, 2, 10, 12)	FilterIN モジュール制御
4006 80A0h+4h×n	rPWMTimer_FORCE_INPUT_[n] (n = 0~1)	FilterIN モジュール固定出力
4006 8200h+4h×n	rPWMTimer_OUTPUTCTRL_[n] (n = 0~3)	OutputCtrl モジュール制御
4006 8330h	rPWMTimer_OUTPUTCTRL_FORCE_OUT_0	OutputCtrl モジュール固定出力
4006 8340h	rPWMTimer_OUTPUTCTRL_RELEASE_OUT	OutputCtrl モジュール固定出力解除
4006 8C00h	rPWMTimer_EVENT_MANAGER_INT_STATUS	グローバル割り込みステータス
4006 8C80h	rPWMTimer_EVENT_MANAGER_INT_MASK	グローバル割り込みマスク
4006 8CC4h	rPWMTimer_PWM_CMP1_MASK	PWM CMP1 フラグマスク
4006 8CC8h	rPWMTimer_PWM_OVERFLOW_MASK	PWM オーバーフローフラグマスク
4006 8CCCh	rPWMTimer_PWM_UNDERFLOW_MASK	PWM アンダーフローフラグマスク
4006 8CD0h+4h×r	rPWMTimer_HWSTORE_OR[r]_STORE_MASK (r = 1, 3)	PWM OR[r]トリガフラグマスク
4006 9000h+400h×k	rPWMTimer_PWM_MODE_[k] (k = 0~1)	Timebase モード制御
4006 9030h+400h×k +4h×q	rPWMTimer_PWM_CLOCK_[q]_[k] (q = 0~1, k = 0~1)	PWM モジュールクロック入力制御
4006 9040h+400h×k +4h×d	rPWMTimer_PWM_UPDOWN_[d]_[k] (d = 0~3, k = 0~1)	PWM モジュール Up/Down 入力制御
4006 9070h+400h×k	rPWMTimer_PWM_TRIG0_[k] (k = 0~1)	PWM モジュール Trig0 入力制御
4006 9080h+400h×k	rPWMTimer_PWM_TRIG1_[k] (k = 0~1)	PWM モジュール Trig1 入力制御
4006 90C0h+400h×k +4h×d	rPWMTimer_PWM_OUTPUTGEN[d]_[k] (d = 0~3, k = 0~1)	PWM モジュール出力生成制御
4006 90E0h+400h×k	rPWMTimer_PWM_BLOCK_CLOCK_[k] (k = 0~1)	PWM モジュール Block_Clock 制御
4006 9100h+400h×k +4h×i	rPWMTimer_TIMEBASE_[i]_[k] (i = 0~7, k = 0~1)	16 ビット Timebase 現カウンタ値
4006 9120h+400h×k +4h×n	rPWMTimer_TIMEBASE_[n*2][n*2+1]_[k] (n = 0~3, k = 0~1)	32 ビットカスケード Timebase 現カウンタ値
4006 9130h+400h×k +4h×n	rPWMTimer_TIMEBASE_[(n+1)*2-1][(n+1)*2]_[k] (n = 0~2, k = 0~1)	32 ビットカスケード Timebase 現カウンタ値
4006 9140h+400h×k +4h×i	rPWMTimer_SCALEVALUE_[i]_[k] (i = 0~7, k = 0~1)	カスケード値設定
4006 91A0h+400h×k +4h×q	rPWMTimer_TIMEBASE_MUX_[q]_[k] (q = 0~1, k = 0~1)	コンペアノキャプチャ入力選択制御
4006 91B0h+400h×k +4h×i	rPWMTimer_HWSTORE_CONF_[i]_[k] (i = 0~7, k = 0~1)	PWM モジュール Hardware Store トリガ制御
4006 91D0h+400h×k +4h×i	rPWMTimer_HWSTORE_MUX_[i]_[k] (i = 0~7, k = 0~1)	PWM モジュール Hardware Store トリガ選択制御

アドレス	レジスタシンボル	レジスタ名
4006 91F0h+400h×k	rPWMTimer_HWSTORE_LOCK_[k] (k = 0~1)	オプションレジスタロックおよびロックステータス
4006 91F4h+400h×k	rPWMTimer_HWSTORE_UNLOCK_[k] (k = 0~1)	オプションレジスタロック解除
4006 9200h+400h×k+40h×r+4h×i	rPWMTimer_OPTIONREG[r]_[i]_[k] (i = 0~7, r = 0~3, k = 0~1)	16 ビット Timebase オプションレジスタ[r]値
4006 9220h+400h×k+40h×r+4h×n	rPWMTimer_OPTIONREG[r]_[n*2][n*2+1]_[k] (n = 0~3, r = 0~3, k = 0~1)	32 ビットカスケード Timebase オプションレジスタ[r]値
4006 9230h+400h×k+40h×r+4h×n	rPWMTimer_OPTIONREG[r]_[(n+1)*2-1][(n+1)*2]_[k] (n = 0~2, r = 0~3, k = 0~1)	32 ビットカスケード Timebase オプションレジスタ[r]値
4006 A010h	rPWMTimer_PWM_SOFTSTART	PWM モジュールカウントイネーブルスタート
4006 A018h	rPWMTimer_PWM_SOFTSTOP	PWM モジュールカウントイネーブルストップ
4006 A040h	rPWMTimer_PWM_SOFTUPDOWN_0	PWM モジュール Up/Down 制御
4006 A050h	rPWMTimer_PWM_SOFTRESET	Timebase リセットトリガ
4006 A0D4h	rPWMTimer_PWM_CMP1_FLAG_POLARITY	PWM CMP1 フラグ設定
4006 A0DCh	rPWMTimer_PWM_CMP1_FLAG	PWM CMP1 フラグ
4006 A108h	rPWMTimer_PWM_MAX_OVERFLOW	PWM オーバーフローフラグ
4006 A10Ch	rPWMTimer_PWM_MIN_UNDERFLOW	PWM アンダーフローフラグ
4006 A1D0h+4h×r	rPWMTimer_HWSTORE_OR[r]_STORE_FLAG (r = 1, 3)	PWM OR[r]トリガフラグ
4006 B100h+4h×s	rPWMTimer_ROUTING_IN_[s] (s = 0, 2)	PWMTimer の IN 入力選択
4006 B200h+4h×n	rPWMTimer_ROUTING_OUT_[n] (n = 0~3)	PWMTimer の OUT 出力選択
4006 B800h	rPWMTimer_ROUTING_OUTCTRL_0	OutputCtrl0 モジュールの入力選択
4006 B804h	rPWMTimer_ROUTING_OUTCTRL_1	OutputCtrl1 モジュールの入力選択
4006 B808h	rPWMTimer_ROUTING_OUTCTRL_2	OutputCtrl2 モジュールの入力選択
4006 B80Ch	rPWMTimer_ROUTING_OUTCTRL_3	OutputCtrl3 モジュールの入力選択
4006 B810h	rPWMTimer_ROUTING_OUTCTRL_4	OutputCtrl4 モジュールの入力選択
4006 B814h	rPWMTimer_ROUTING_OUTCTRL_5	OutputCtrl5 モジュールの入力選択
4006 B818h	rPWMTimer_ROUTING_OUTCTRL_6	OutputCtrl6 モジュールの入力選択
4006 B81Ch	rPWMTimer_ROUTING_OUTCTRL_7	OutputCtrl7 モジュールの入力選択
4006 B820h	rPWMTimer_ROUTING_OUTCTRL_8	OutputCtrl8 モジュールの入力選択
4006 B824h	rPWMTimer_ROUTING_OUTCTRL_9	OutputCtrl9 モジュールの入力選択
4006 B828h	rPWMTimer_ROUTING_OUTCTRL_10	OutputCtrl10 モジュールの入力選択
4006 B82Ch	rPWMTimer_ROUTING_OUTCTRL_11	OutputCtrl11 モジュールの入力選択
4006 B830h	rPWMTimer_ROUTING_OUTCTRL_12	OutputCtrl12 モジュールの入力選択
4006 B834h	rPWMTimer_ROUTING_OUTCTRL_13	OutputCtrl13 モジュールの入力選択
4006 B838h	rPWMTimer_ROUTING_OUTCTRL_14	OutputCtrl14 モジュールの入力選択
4006 B83Ch	rPWMTimer_ROUTING_OUTCTRL_15	OutputCtrl15 モジュールの入力選択

1.4 レジスタの説明

1.4.1 rPWMTimer_FILTER_IN_[s] — FilterIN モジュール制御 (s = 0, 10)

FilterIN モジュール : 0~1, 10~11 (Trig1, Trig0 用)

本レジスタには、ライトマスクの制御があります。

アドレス 4006 8000h+2h×s

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bMOST_SIGNIFICANT_BIT_REGISTER_ACCESS	—		bBOUNCE_STEP_[s+1]	bBOUNCE_THRESHOLD_[s+1]										—	bPOLAR_IN_[s+1]
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bLEAST_SIGNIFICANT_BIT_REGISTER_ACCESS	—		bBOUNCE_STEP_[s]	bBOUNCE_THRESHOLD_[s]										—	bPOLAR_IN_[s]
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.3 rPWMTimer_FILTER_IN_[s]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	bMOST_SIGNIFICANT_BIT_REGISTER_ACCESS	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの上位 16 ビットのライトイネーブルが有効となります。	W
b30~b29	予約ビット	初期値を保持してください。	R/W
b28	bBOUNCE_STEP_[s+1]	FilterIN[s+e]のバンスフィルタで使用されるクロック分周設定 1'b0 : PWM_PCLK/8 1'b1 : PWM_PCLK/2048 本ビット : e = 1 (FilterIN 1, 11)	R/W
b27~b18	bBOUNCE_THRESHOLD_[s+1]	FilterIN[s+e]のバンスフィルタ期間 10'd0 : バンスフィルタなし 10'd1 : 1 バンスクロック 10'd2 : 2 バンスクロック 10'dX : X バンスクロック 10'd1023 : 1023 バンスクロック 本ビット : e = 1 (FilterIN 1, 11)	R/W
b17	予約ビット	初期値を保持してください。	R/W
b16	bPOLAR_IN_[s+1]	FilterIN[s+e]の極性選択 1'b0 : ダイレクト 1'b1 : 反転 本ビット : e = 1 (FilterIN 1, 11)	R/W
b15	bLEAST_SIGNIFICANT_BIT_REGISTER_ACCESS	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの下位 16 ビットのライトイネーブルが有効となります。	W
b14~b13	予約ビット	初期値を保持してください。	R/W
b12	bBOUNCE_STEP_[s]	bBOUNCE_STEP_[s+1]を参照、本ビット : e = 0 (FilterIN 0, 10)	R/W
b11~b2	bBOUNCE_THRESHOLD_[s]	bBOUNCE_THRESHOLD_[s+1]を参照、本ビット : e = 0 (FilterIN 0, 10)	R/W

ビット位置	ビット名	機能	R/W
b1	予約ビット	初期値を保持してください。	R/W
b0	bPOLAR_IN_[s]	bPOLAR_IN_[s+1]を参照、本ビット : e = 0 (FilterIN 0, 10)	R/W

1.4.2 rPWMTimer_FILTER_IN_[s] — FilterIN モジュール制御 (s = 2, 12)

FilterIN モジュール : 2, 12 (ClockIn 用)

本レジスタには、ライトマスクの制御があります。

アドレス 4006 8000h+2h×s

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—														
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bLEAST_SIGNIFICANT_BIT_REGISTER_ACCESS	—		bBOUNCE_STEP_[s]	bBOUNCE_THRESHOLD_[s]										—	bPOLAR_IN_[s]
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.4 rPWMTimer_FILTER_IN_[s]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	予約ビット	初期値を保持してください。	W
b30~b16	予約ビット	初期値を保持してください。	R/W
b15	bLEAST_SIGNIFICANT_BIT_REGISTER_ACCESS	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの下位 16 ビットのライトイネーブルが有効となります。	W
b14~b13	予約ビット	初期値を保持してください。	R/W
b12	bBOUNCE_STEP_[s]	FilterIN[s]のバウンスフィルタで使用されるクロック分周設定 1'b0 : PWM_PCLK/8 1'b1 : PWM_PCLK/2048 (FilterIN 2, 12)	R/W
b11~b2	bBOUNCE_THRESHOLD_[s]	FilterIN[s]のバウンスフィルタ期間 10'd0 : バウンスフィルタなし 10'd1 : 1 バウンスクロック 10'd2 : 2 バウンスクロック 10'dX : X バウンスクロック 10'd1023 : 1023 バウンスクロック (FilterIN 2, 12)	R/W
b1	予約ビット	初期値を保持してください。	R/W
b0	bPOLAR_IN_[s]	FilterIN[s]の極性選択 1'b0 : ダイレクト 1'b1 : 反転 (FilterIN 2, 12)	R/W

1.4.3 rPWMTimer_FORCE_INPUT_[n] — FilterIN モジュール固定出力 (n = 0~1)

FilterIN モジュール : 0~2, 10~12

本レジスタには、ライトマスクの制御があります。

アドレス 4006 80A0h+4h×n

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	bMASK_SET_RESET_[n*10+2]	bFORCE_IN_[n*10+2]	—	bMASK_SET_RESET_[n*10+1]	bFORCE_IN_[n*10+1]	—	bMASK_SET_RESET_[n*10]	bFORCE_IN_[n*10]	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.5 1.4.2 rPWMTimer_FORCE_INPUT_[n]レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b30	予約ビット	読むと 0 が読み出されます。	R
b29	予約ビット	初期値を保持してください。	W
b28~b27	予約ビット	初期値を保持してください。	R/W
b26	予約ビット	初期値を保持してください。	W
b25~b24	予約ビット	初期値を保持してください。	R/W
b23	予約ビット	初期値を保持してください。	W
b22~b21	予約ビット	初期値を保持してください。	R/W
b20	予約ビット	初期値を保持してください。	W
b19~b18	予約ビット	初期値を保持してください。	R/W
b17	予約ビット	初期値を保持してください。	W
b16~b15	予約ビット	初期値を保持してください。	R/W
b14	予約ビット	初期値を保持してください。	W
b13~b12	予約ビット	初期値を保持してください。	R/W
b11	予約ビット	初期値を保持してください。	W
b10~b9	予約ビット	初期値を保持してください。	R/W
b8	bMASK_SET_RESET_[n*10+2]	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの 6~7 ビットのライトイネーブルが有効となります。	W
b7~b6	bFORCE_IN_[n*10+2]	FilterIN[n*10+e]の固定出力 2'b00 : 固定なし (ダイレクト入力) 2'b01 : 1 固定 2'b1X : 0 固定 本ビット : e = 2 (FilterIN 2, 12)	R/W
b5	bMASK_SET_RESET_[n*10+1]	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの 3~4 ビットのライトイネーブルが有効となります。	W
b4~b3	bFORCE_IN_[n*10+1]	bFORCE_IN_[n*10+2]を参照、本ビット : e = 1 (FilterIN 1, 11)	R/W
b2	bMASK_SET_RESET_[n*10]	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでの 0~1 ビットのライトイネーブルが有効となります。	W
b1~b0	bFORCE_IN_[n*10]	bFORCE_IN_[n*10+2]を参照、本ビット : e = 0 (FilterIN 0, 10)	R/W

1.4.4 rPWMTimer_OUTPUTCTRL_[n] — OutputCtrl モジュール制御 (n = 0~3)

OutputCtrl モジュール : 0~15

アドレス 4006 8200h + 4h × n

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	bBYPASS_MODE_ DE_[n*4+3]			—		bINVERTED_SET_ ET_[n*4+3]		—	bBYPASS_MODE_ DE_[n*4+2]			—		bINVERTED_SET_ ET_[n*4+2]	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	bBYPASS_MODE_ DE_[n*4+1]			—		bINVERTED_SET_ ET_[n*4+1]		—	bBYPASS_MODE_ DE_[n*4]			—		bINVERTED_SET_ ET_[n*4]	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.6 rPWMTimer_OUTPUTCTRL_[n] レジスタの内容

ビット位置	ビット名	機能	R/W
b31	予約ビット	初期値を保持してください。	R/W
b30	bBYPASS_MODE_[n*4+3]	OutputCtrl[n*4+e]のモード設定 本ビットは、1に設定してください。 本ビット : e = 3 (OutputCtrl 3, 7, 11, 15)	R/W
b29~b26	予約ビット	初期値を保持してください。	R/W
b25	bINVERTED_SET_[n*4+3]	OutputCtrl[n*4+e]の入力極性選択 1'b0 : ダイレクト 1'b1 : 反転 本ビット : e = 3 (OutputCtrl 3, 7, 11, 15)	R/W
b24~b23	予約ビット	初期値を保持してください。	R/W
b22	bBYPASS_MODE_[n*4+2]	bBYPASS_MODE_[n*4+3]を参照、本ビット : e = 2 (OutputCtrl 2, 6, 10, 14)	R/W
b21~b18	予約ビット	初期値を保持してください。	R/W
b17	bINVERTED_SET_[n*4+2]	bINVERTED_SET_[n*4+3]を参照、本ビット : e = 2 (OutputCtrl 2, 6, 10, 14)	R/W
b16~b15	予約ビット	初期値を保持してください。	R/W
b14	bBYPASS_MODE_[n*4+1]	bBYPASS_MODE_[n*4+3]を参照、本ビット : e = 1 (OutputCtrl 1, 5, 9, 13)	R/W
b13~b10	予約ビット	初期値を保持してください。	R/W
b9	bINVERTED_SET_[n*4+1]	bINVERTED_SET_[n*4+3]を参照、本ビット : e = 1 (OutputCtrl 1, 5, 9, 13)	R/W
b8~b7	予約ビット	初期値を保持してください。	R/W
b6	bBYPASS_MODE_[n*4]	bBYPASS_MODE_[n*4+3]を参照、本ビット : e = 0 (OutputCtrl 0, 4, 8, 12)	R/W
b5~b2	予約ビット	初期値を保持してください。	R/W
b1	bINVERTED_SET_[n*4]	bINVERTED_SET_[n*4+3]を参照、本ビット : e = 0 (OutputCtrl 0, 4, 8, 12)	R/W
b0	予約ビット	初期値を保持してください。	R/W

1.4.5 rPWMTimer_OUTPUTCTRL_FORCE_OUT_0 — OutputCtrl モジュール固定出力

OutputCtrl モジュール：0～15

本レジスタには、ライトマスクの制御があります。上位 16 ビットは、対応する下位 16 ビットのライトイネーブルとして使用されます。

アドレス 4006 8330h	
ビット	b31 b30 b29 b28 b27 b26 b25 b24 b23 b22 b21 b20 b19 b18 b17 b16
	bFORCE_OUT_MASK
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ビット	b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0
	bFORCE_OUT_15 bFORCE_OUT_14 bFORCE_OUT_13 bFORCE_OUT_12 bFORCE_OUT_11 bFORCE_OUT_10 bFORCE_OUT_9 bFORCE_OUT_8 bFORCE_OUT_7 bFORCE_OUT_6 bFORCE_OUT_5 bFORCE_OUT_4 bFORCE_OUT_3 bFORCE_OUT_2 bFORCE_OUT_1 bFORCE_OUT_0
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

表 1.7 rPWMTimer_OUTPUTCTRL_FORCE_OUT_0 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	bFORCE_OUT_MASK	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでそれぞれ下位ビットに対応したライトイネーブルが有効となります。	W
b15	bFORCE_OUT_15	OutputCtrl[e]の固定出力 1 書き込み：1 固定 0 書き込み：0 固定 本ビットは、上位ビットに対応したライトイネーブルがあります。 本ビット：e = 15 (OutputCtrl 15)	W
b14	bFORCE_OUT_14	上記参照、本ビット：e = 14 (OutputCtrl 14)	W
b13	bFORCE_OUT_13	上記参照、本ビット：e = 13 (OutputCtrl 13)	W
b12	bFORCE_OUT_12	上記参照、本ビット：e = 12 (OutputCtrl 12)	W
b11	bFORCE_OUT_11	上記参照、本ビット：e = 11 (OutputCtrl 11)	W
b10	bFORCE_OUT_10	上記参照、本ビット：e = 10 (OutputCtrl 10)	W
b9	bFORCE_OUT_9	上記参照、本ビット：e = 9 (OutputCtrl 9)	W
b8	bFORCE_OUT_8	上記参照、本ビット：e = 8 (OutputCtrl 8)	W
b7	bFORCE_OUT_7	上記参照、本ビット：e = 7 (OutputCtrl 7)	W
b6	bFORCE_OUT_6	上記参照、本ビット：e = 6 (OutputCtrl 6)	W
b5	bFORCE_OUT_5	上記参照、本ビット：e = 5 (OutputCtrl 5)	W
b4	bFORCE_OUT_4	上記参照、本ビット：e = 4 (OutputCtrl 4)	W
b3	bFORCE_OUT_3	上記参照、本ビット：e = 3 (OutputCtrl 3)	W
b2	bFORCE_OUT_2	上記参照、本ビット：e = 2 (OutputCtrl 2)	W
b1	bFORCE_OUT_1	上記参照、本ビット：e = 1 (OutputCtrl 1)	W
b0	bFORCE_OUT_0	上記参照、本ビット：e = 0 (OutputCtrl 0)	W

1.4.6 rPWMTimer_OUTPUTCTRL_RELEASE_OUT — OutputCtrl モジュール固定出力解除

アドレス 4006 8340h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bRELEASE_OUT_15	bRELEASE_OUT_14	bRELEASE_OUT_13	bRELEASE_OUT_12	bRELEASE_OUT_11	bRELEASE_OUT_10	bRELEASE_OUT_9	bRELEASE_OUT_8	bRELEASE_OUT_7	bRELEASE_OUT_6	bRELEASE_OUT_5	bRELEASE_OUT_4	bRELEASE_OUT_3	bRELEASE_OUT_2	bRELEASE_OUT_1	bRELEASE_OUT_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.8 rPWMTimer_OUTPUTCTRL_RELEASE_OUT レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	初期値を保持してください。	W
b15	bRELEASE_OUT_15	OutputCtrl[e]の固定出力解除 1 を書き込むと固定出力は解除、0 の書き込みは無効です。 初期出力は、固定値が解除されています。 本ビット : e = 15 (OutputCtrl 15)	W
b14	bRELEASE_OUT_14	上記参照、本ビット : e = 14 (OutputCtrl 14)	W
b13	bRELEASE_OUT_13	上記参照、本ビット : e = 13 (OutputCtrl 13)	W
b12	bRELEASE_OUT_12	上記参照、本ビット : e = 12 (OutputCtrl 12)	W
b11	bRELEASE_OUT_11	上記参照、本ビット : e = 11 (OutputCtrl 11)	W
b10	bRELEASE_OUT_10	上記参照、本ビット : e = 10 (OutputCtrl 10)	W
b9	bRELEASE_OUT_9	上記参照、本ビット : e = 9 (OutputCtrl 9)	W
b8	bRELEASE_OUT_8	上記参照、本ビット : e = 8 (OutputCtrl 8)	W
b7	bRELEASE_OUT_7	上記参照、本ビット : e = 7 (OutputCtrl 7)	W
b6	bRELEASE_OUT_6	上記参照、本ビット : e = 6 (OutputCtrl 6)	W
b5	bRELEASE_OUT_5	上記参照、本ビット : e = 5 (OutputCtrl 5)	W
b4	bRELEASE_OUT_4	上記参照、本ビット : e = 4 (OutputCtrl 4)	W
b3	bRELEASE_OUT_3	上記参照、本ビット : e = 3 (OutputCtrl 3)	W
b2	bRELEASE_OUT_2	上記参照、本ビット : e = 2 (OutputCtrl 2)	W
b1	bRELEASE_OUT_1	上記参照、本ビット : e = 1 (OutputCtrl 1)	W
b0	bRELEASE_OUT_0	上記参照、本ビット : e = 0 (OutputCtrl 0)	W

1.4.7 rPWMTimer_EVENT_MANAGER_INT_STATUS — グローバル割り込みステータス

アドレス 4006 8C00h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bINT_FROM_HWSTORE_OR3_STORE_FLAG	—	bINT_FROM_HWSTORE_OR1_STORE_FLAG	—	bINT_FROM_PWM_UNDERFLOW	bINT_FROM_PWM_OVERFLOW	bINT_FROM_PWM_CMP1_FLAG	—								
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.9 rPWMTimer_EVENT_MANAGER_INT_STATUS レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	読むと 0 が読み出されます。	R
b15	bINT_FROM_HWSTORE_OR3_STORE_FLAG	rPWMTimer_HWSTORE_OR3_STORE_FLAG レジスタの割り込みステータスマスクされていないフラグが立つと、本ビットは 1 にセットされます。マスクされていないフラグがすべて立っていない、またはフラグがすべてマスクされている場合、本ビットは 0 にセットされます。	R
b14	予約ビット	読むと 0 が読み出されます。	R
b13	bINT_FROM_HWSTORE_OR1_STORE_FLAG	rPWMTimer_HWSTORE_OR1_STORE_FLAG レジスタの割り込みステータス bINT_FROM_HWSTORE_OR3_STORE_FLAG を参照	R
b12	予約ビット	読むと 0 が読み出されます。	R
b11	bINT_FROM_PWM_UNDERFLOW	rPWMTimer_PWM_MIN_UNDERFLOW レジスタの割り込みステータス bINT_FROM_HWSTORE_OR3_STORE_FLAG を参照	R
b10	bINT_FROM_PWM_OVERFLOW	rPWMTimer_PWM_MAX_OVERFLOW レジスタの割り込みステータス bINT_FROM_HWSTORE_OR3_STORE_FLAG を参照	R
b9	bINT_FROM_PWM_CMP1_FLAG	rPWMTimer_PWM_CMP1_FLAG レジスタの割り込みステータス bINT_FROM_HWSTORE_OR3_STORE_FLAG を参照	R
b8～b0	予約ビット	読むと 0 が読み出されます。	R

1.4.8 rPWMTimer_EVENT_MANAGER_INT_MASK — グローバル割り込みマスク

アドレス 4006 8C80h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bMASK_FOR_HWSTORE_OR3_STORE_FLAG	—	bMASK_FOR_HWSTORE_OR1_STORE_FLAG	—	bMASK_FOR_PWM_MIN_UNDERFLOW	bMASK_FOR_PWM_MAX_OVERFLOW	bMASK_FOR_PWM_CMP1_FLAG	—								
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.10 rPWMTimer_EVENT_MANAGER_INT_MASK レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	読むと 0 が読み出されます。	R
b15	bMASK_FOR_HWSTORE_OR3_STORE_FLAG	rPWMTimer_HWSTORE_OR3_STORE_FLAG レジスタからの割り込みマスク 1'b0 : マスクあり、割り込み禁止 1'b1 : マスクなし、割り込み許可	R/W
b14	予約ビット	初期値を保持してください。	R/W
b13	bMASK_FOR_HWSTORE_OR1_STORE_FLAG	rPWMTimer_HWSTORE_OR1_STORE_FLAG レジスタからの割り込みマスク bMASK_FOR_HWSTORE_OR3_STORE_FLAG を参照	R/W
b12	予約ビット	初期値を保持してください。	R/W
b11	bMASK_FOR_PWM_MIN_UNDERFLOW	rPWMTimer_PWM_MIN_UNDERFLOW レジスタからの割り込みマスク bMASK_FOR_HWSTORE_OR3_STORE_FLAG を参照	R/W
b10	bMASK_FOR_PWM_MAX_OVERFLOW	rPWMTimer_PWM_MAX_OVERFLOW レジスタからの割り込みマスク bMASK_FOR_HWSTORE_OR3_STORE_FLAG を参照	R/W
b9	bMASK_FOR_PWM_CMP1_FLAG	rPWMTimer_PWM_CMP1_FLAG レジスタからの割り込みマスク bMASK_FOR_HWSTORE_OR3_STORE_FLAG を参照	R/W
b8～b0	予約ビット	初期値を保持してください。	R/W

1.4.9 rPWMTimer_PWM_CMP1_MASK — PWM CMP1 フラグマスク

アドレス 4006 8CC4h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM_CMP1_MASK_15	bPWM_CMP1_MASK_14	bPWM_CMP1_MASK_13	bPWM_CMP1_MASK_12	bPWM_CMP1_MASK_11	bPWM_CMP1_MASK_10	bPWM_CMP1_MASK_9	bPWM_CMP1_MASK_8	bPWM_CMP1_MASK_7	bPWM_CMP1_MASK_6	bPWM_CMP1_MASK_5	bPWM_CMP1_MASK_4	bPWM_CMP1_MASK_3	bPWM_CMP1_MASK_2	bPWM_CMP1_MASK_1	bPWM_CMP1_MASK_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.11 rPWMTimer_PWM_CMP1_MASK レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_CMP1_MASK_15	PWM[k] シングルカウンタ[i] の CMP1 フラグマスク 1'b0 : マスクあり、割り込み禁止 1'b1 : マスクなし、割り込み許可 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bPWM_CMP1_MASK_14	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bPWM_CMP1_MASK_13	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bPWM_CMP1_MASK_12	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bPWM_CMP1_MASK_11	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bPWM_CMP1_MASK_10	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bPWM_CMP1_MASK_9	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bPWM_CMP1_MASK_8	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bPWM_CMP1_MASK_7	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bPWM_CMP1_MASK_6	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bPWM_CMP1_MASK_5	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bPWM_CMP1_MASK_4	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bPWM_CMP1_MASK_3	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bPWM_CMP1_MASK_2	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bPWM_CMP1_MASK_1	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bPWM_CMP1_MASK_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.10 rPWMTimer_PWM_OVERFLOW_MASK — PWM オーバーフローフラグマスク

アドレス 4006 8CC8h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM_OVERFLOW_MASK_7_1	bPWM_OVERFLOW_MASK_6_1	bPWM_OVERFLOW_MASK_5_1	bPWM_OVERFLOW_MASK_4_1	bPWM_OVERFLOW_MASK_3_1	bPWM_OVERFLOW_MASK_2_1	bPWM_OVERFLOW_MASK_1_1	bPWM_OVERFLOW_MASK_0_1	bPWM_OVERFLOW_MASK_7_0	bPWM_OVERFLOW_MASK_6_0	bPWM_OVERFLOW_MASK_5_0	bPWM_OVERFLOW_MASK_4_0	bPWM_OVERFLOW_MASK_3_0	bPWM_OVERFLOW_MASK_2_0	bPWM_OVERFLOW_MASK_1_0	bPWM_OVERFLOW_MASK_0_0
リセット後の値	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

表 1.12 rPWMTimer_PWM_OVERFLOW_MASK レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_OVERFLOW_MASK_7_1	PWM[k] シングルカウンタ [i] からのオーバーフローフラグマスク 1'b0 : マスクあり、割り込み禁止 1'b1 : マスクなし、割り込み許可 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bPWM_OVERFLOW_MASK_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bPWM_OVERFLOW_MASK_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bPWM_OVERFLOW_MASK_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bPWM_OVERFLOW_MASK_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bPWM_OVERFLOW_MASK_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bPWM_OVERFLOW_MASK_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bPWM_OVERFLOW_MASK_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bPWM_OVERFLOW_MASK_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bPWM_OVERFLOW_MASK_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bPWM_OVERFLOW_MASK_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bPWM_OVERFLOW_MASK_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bPWM_OVERFLOW_MASK_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bPWM_OVERFLOW_MASK_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bPWM_OVERFLOW_MASK_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bPWM_OVERFLOW_MASK_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.11 rPWMTimer_PWM_UNDERFLOW_MASK — PWM アンダーフローフラグマスク

アドレス 4006 8CCCh

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM_UNDERFLOW_MASK_7_1	bPWM_UNDERFLOW_MASK_6_1	bPWM_UNDERFLOW_MASK_5_1	bPWM_UNDERFLOW_MASK_4_1	bPWM_UNDERFLOW_MASK_3_1	bPWM_UNDERFLOW_MASK_2_1	bPWM_UNDERFLOW_MASK_1_1	bPWM_UNDERFLOW_MASK_0_1	bPWM_UNDERFLOW_MASK_7_0	bPWM_UNDERFLOW_MASK_6_0	bPWM_UNDERFLOW_MASK_5_0	bPWM_UNDERFLOW_MASK_4_0	bPWM_UNDERFLOW_MASK_3_0	bPWM_UNDERFLOW_MASK_2_0	bPWM_UNDERFLOW_MASK_1_0	bPWM_UNDERFLOW_MASK_0_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.13 rPWMTimer_PWM_UNDERFLOW_MASK レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_UNDERFLOW_MASK_7_1	PWM[k] シングルカウンタ [i] からのアンダーフローフラグマスク 1'b0 : マスクあり、割り込み禁止 1'b1 : マスクなし、割り込み許可 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bPWM_UNDERFLOW_MASK_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bPWM_UNDERFLOW_MASK_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bPWM_UNDERFLOW_MASK_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bPWM_UNDERFLOW_MASK_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bPWM_UNDERFLOW_MASK_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bPWM_UNDERFLOW_MASK_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bPWM_UNDERFLOW_MASK_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bPWM_UNDERFLOW_MASK_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bPWM_UNDERFLOW_MASK_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bPWM_UNDERFLOW_MASK_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bPWM_UNDERFLOW_MASK_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bPWM_UNDERFLOW_MASK_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bPWM_UNDERFLOW_MASK_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bPWM_UNDERFLOW_MASK_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bPWM_UNDERFLOW_MASK_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.12 rPWMTimer_HWSTORE_OR[r]_STORE_MASK — PWM OR[r]トリガフラグマスク (r = 1, 3)

アドレス 4006 8CD0h + 4h × r

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bOR[r]_STORE_MASK_15	bOR[r]_STORE_MASK_14	bOR[r]_STORE_MASK_13	bOR[r]_STORE_MASK_12	bOR[r]_STORE_MASK_11	bOR[r]_STORE_MASK_10	bOR[r]_STORE_MASK_9	bOR[r]_STORE_MASK_8	bOR[r]_STORE_MASK_7	bOR[r]_STORE_MASK_6	bOR[r]_STORE_MASK_5	bOR[r]_STORE_MASK_4	bOR[r]_STORE_MASK_3	bOR[r]_STORE_MASK_2	bOR[r]_STORE_MASK_1	bOR[r]_STORE_MASK_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.14 rPWMTimer_HWSTORE_OR[r]_STORE_MASK レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bOR[r]_STORE_MASK_15	PWM[k]シングルカウンタ[i]の OR[r]トリガフラグマスク 1'b0 : マスクあり、割り込み禁止 1'b1 : マスクなし、割り込み許可 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bOR[r]_STORE_MASK_14	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bOR[r]_STORE_MASK_13	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bOR[r]_STORE_MASK_12	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bOR[r]_STORE_MASK_11	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bOR[r]_STORE_MASK_10	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bOR[r]_STORE_MASK_9	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bOR[r]_STORE_MASK_8	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bOR[r]_STORE_MASK_7	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bOR[r]_STORE_MASK_6	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bOR[r]_STORE_MASK_5	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bOR[r]_STORE_MASK_4	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bOR[r]_STORE_MASK_3	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bOR[r]_STORE_MASK_2	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bOR[r]_STORE_MASK_1	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bOR[r]_STORE_MASK_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.13 rPWMTimer_PWM_MODE_[k] — Timebase モード制御 (k = 0~1)

アドレス 4006 9000h + 400h × k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bPWM_RANGE_DMODE_7	bPWM_CASCADE_MODE_7	bPWM_MASTER_SINGLE_COUNTER_7	bPWM_RANGE_DMODE_6	bPWM_CASCADE_MODE_6	bPWM_MASTER_SINGLE_COUNTER_6	bPWM_RANGE_DMODE_5	bPWM_CASCADE_MODE_5	bPWM_MASTER_SINGLE_COUNTER_5	bPWM_RANGE_DMODE_4	bPWM_CASCADE_MODE_4	bPWM_MASTER_SINGLE_COUNTER_4	bPWM_RANGE_DMODE_3	bPWM_CASCADE_MODE_3	bPWM_MASTER_SINGLE_COUNTER_3	bPWM_RANGE_DMODE_2
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM_RANGE_DMODE_3	bPWM_CASCADE_MODE_3	bPWM_MASTER_SINGLE_COUNTER_3	bPWM_RANGE_DMODE_2	bPWM_CASCADE_MODE_2	bPWM_MASTER_SINGLE_COUNTER_2	bPWM_RANGE_DMODE_1	bPWM_CASCADE_MODE_1	—	bPWM_MASTER_SINGLE_COUNTER_1	bPWM_RANGE_DMODE_0	—	—	—	—	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.15 rPWMTimer_PWM_MODE_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31	bPWM_RANGE_DMODE_7	PWM[k] シングルカウンタ 7 のカウントモード 1'b0 : フリーランニングモード : 最大値は 16'hFFFF 1'b1 : レンジモード : 最大値はオプションレジスタ 0 で設定	R/W
b30	bPWM_CASCADE_MODE_7	PWM[k] シングルカウンタ 7 の機能モード 1'b0 : 16 ビットカウンタ、またはカスケードの下位 16 ビットカウンタ 1'b1 : カスケードの上位 16 ビットカウンタ (カスケード論理を有効)	R/W
b29~b28	bPWM_MASTER_SINGLE_COUNTER_7	PWM[k] シングルカウンタ 7 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 6 をマスタカウンタ 2'b10 : シングルカウンタ 5 をマスタカウンタ 2'b11 : シングルカウンタ 4 をマスタカウンタ	R/W
b27	bPWM_RANGE_DMODE_6	PWM[k] シングルカウンタ 6 のカウントモード bPWM_RANGE_DMODE_7 を参照	R/W
b26	bPWM_CASCADE_MODE_6	PWM[k] シングルカウンタ 6 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W
b25~b24	bPWM_MASTER_SINGLE_COUNTER_6	PWM[k] シングルカウンタ 6 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 5 をマスタカウンタ 2'b10 : シングルカウンタ 4 をマスタカウンタ 2'b11 : シングルカウンタ 3 をマスタカウンタ	R/W
b23	bPWM_RANGE_DMODE_5	PWM[k] シングルカウンタ 5 のカウントモード bPWM_RANGE_DMODE_7 を参照	R/W
b22	bPWM_CASCADE_MODE_5	PWM[k] シングルカウンタ 5 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W
b21~b20	bPWM_MASTER_SINGLE_COUNTER_5	PWM[k] シングルカウンタ 5 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 4 をマスタカウンタ 2'b10 : シングルカウンタ 3 をマスタカウンタ 2'b11 : シングルカウンタ 2 をマスタカウンタ	R/W
b19	bPWM_RANGE_DMODE_4	PWM[k] シングルカウンタ 4 のカウントモード bPWM_RANGE_DMODE_7 を参照	R/W
b18	bPWM_CASCADE_MODE_4	PWM[k] シングルカウンタ 4 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W

ビット位置	ビット名	機能	R/W
b17~b16	bPWM_MASTER_SINGLE_COUNTER_4	PWM[k]シングルカウンタ 4 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 3 をマスタカウンタ 2'b10 : シングルカウンタ 2 をマスタカウンタ 2'b11 : シングルカウンタ 1 をマスタカウンタ	R/W
b15	bPWM_RANGEDMODE_3	PWM[k]シングルカウンタ 3 のカウントモード bPWM_RANGEDMODE_7 を参照	R/W
b14	bPWM_CASCADE_MODE_3	PWM[k]シングルカウンタ 3 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W
b13~b12	bPWM_MASTER_SINGLE_COUNTER_3	PWM[k]シングルカウンタ 3 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 2 をマスタカウンタ 2'b10 : シングルカウンタ 1 をマスタカウンタ 2'b11 : シングルカウンタ 0 をマスタカウンタ	R/W
b11	bPWM_RANGEDMODE_2	PWM[k]シングルカウンタ 2 のカウントモード bPWM_RANGEDMODE_7 を参照	R/W
b10	bPWM_CASCADE_MODE_2	PWM[k]シングルカウンタ 2 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W
b9~b8	bPWM_MASTER_SINGLE_COUNTER_2	PWM[k]シングルカウンタ 2 のマスタカウンタ設定 2'b00 : マスタカウンタ無効 2'b01 : シングルカウンタ 1 をマスタカウンタ 2'b10 : シングルカウンタ 0 をマスタカウンタ 2'b11 : 予約	R/W
b7	bPWM_RANGEDMODE_1	PWM[k]シングルカウンタ 1 のカウントモード bPWM_RANGEDMODE_7 を参照	R/W
b6	bPWM_CASCADE_MODE_1	PWM[k]シングルカウンタ 1 の機能モード bPWM_CASCADE_MODE_7 を参照	R/W
b5	予約ビット	読むと 0 が読み出されます。	R
b4	bPWM_MASTER_SINGLE_COUNTER_1	PWM[k]シングルカウンタ 1 のマスタカウンタ設定 1'b0 : マスタカウンタ無効 1'b1 : シングルカウンタ 0 をマスタカウンタ	R/W
b3	bPWM_RANGEDMODE_0	PWM[k]シングルカウンタ 0 のカウントモード bPWM_RANGEDMODE_7 を参照	R/W
b2~b0	予約ビット	初期値を保持してください。	R/W

1.4.14 rPWMTimer_PWM_CLOCK_[q]_[k] — PWM モジュールクロック入力制御 (q = 0~1, k = 0~1)

アドレス 4006 9030h + 400h × k + 4h × q

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—		bEDGE_CLOCK [q*4+3]	—		bMUXEDCLOCK_ [q*4+3]		—		bEDGE_CLOCK [q*4+2]	—		bMUXEDCLOCK_ [q*4+2]			
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—		bEDGE_CLOCK [q*4+1]	—		bMUXEDCLOCK_ [q*4+1]		—		bEDGE_CLOCK [q*4]	—		bMUXEDCLOCK_ [q*4]			
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.16 rPWMTimer_PWM_CLOCK_[q]_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b30	予約ビット	読むと 0 が読み出されます。	R
b29~b28	bEDGE_CLOCK_ [q*4+3]	PWM[k] のクロック入力[q*4+e]のエッジ検出設定 2'b00 : ダイレクト入力 2'b01 : 立ち上がりエッジパルス 2'b10 : 立ち下がりエッジパルス 2'b11 : 立ち上がり／立ち下がりの両エッジパルス 本ビット : e = 3 (シングルカウンタ 3, 7)	R/W
b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	bMUXEDCLOCK_ [q*4+3]	PWM[k] のクロック入力[q*4+e]のクロック選択 3'b000 : 1 固定 (PWM_PCLK) 3'b001 : ClockIn.[k] 3'b110 : Block_Clock.0.[k] 3'b111 : Block_Clock.1.[k] それ以外 : 予約 本ビット : e = 3 (シングルカウンタ 3, 7)	R/W
b23~b22	予約ビット	読むと 0 が読み出されます。	R
b21~b20	bEDGE_CLOCK_ [q*4+2]	bEDGE_CLOCK_[q*4+3]を参照、本ビット : e = 2 (シングルカウンタ 2, 6)	R/W
b19	予約ビット	読むと 0 が読み出されます。	R
b18~b16	bMUXEDCLOCK_ [q*4+2]	bMUXEDCLOCK_[q*4+3]を参照、本ビット : e = 2 (シングルカウンタ 2, 6)	R/W
b15~b14	予約ビット	読むと 0 が読み出されます。	R
b13~b12	bEDGE_CLOCK_ [q*4+1]	bEDGE_CLOCK_[q*4+3]を参照、本ビット : e = 1 (シングルカウンタ 1, 5)	R/W
b11	予約ビット	読むと 0 が読み出されます。	R
b10~b8	bMUXEDCLOCK_ [q*4+1]	bMUXEDCLOCK_[q*4+3]を参照、本ビット : e = 1 (シングルカウンタ 1, 5)	R/W
b7~b6	予約ビット	読むと 0 が読み出されます。	R
b5~b4	bEDGE_CLOCK_ [q*4]	bEDGE_CLOCK_[q*4+3]を参照、本ビット : e = 0 (シングルカウンタ 0, 4)	R/W
b3	予約ビット	読むと 0 が読み出されます。	R
b2~b0	bMUXEDCLOCK_ [q*4]	bMUXEDCLOCK_[q*4+3]を参照、本ビット : e = 0 (シングルカウンタ 0, 4)	R/W

1.4.15 rPWMTimer_PWM_UPDOWN_[d]_[k] — PWM モジュール Up/Down 入力制御 (d = 0~3, k = 0~1)

アドレス 4006 9040h + 400h × k + 4h × d

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bUPDOWN_MODE_ [d*2+1]	bOREDDOWN_ [d*2+1]	—	—	—	—	—	—	—	bOREDUP_ [d*2+1]	—	—	—	—	—	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bUPDOWN_MODE_ [d*2]	bOREDDOWN_ [d*2]	—	—	—	—	—	—	—	bOREDUP_ [d*2]	—	—	—	—	—	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.17 rPWMTimer_PWM_UPDOWN_[d]_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b30	bUPDOWN_MODE_ [d*2+1]	PWM[k]の Up/Down 入力[d*2+e]のモード 2'b00 : Set/Reset モード : Set 信号で Up、Reset 信号で Down 2'b01 : 予約 2'b10 : Bypass モード : Up/Down = "Set 信号" 2'b11 : Bypass モード : Up/Down = "Reset 信号"の反転 Set 信号および Reset 信号は、「1.5.6.2(1)(b) Up/Down 入力」を参照してください。 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b29	bOREDDOWN_ [d*2+1]	PWM[k]の Up/Down 入力[d*2+e]、Reset 信号での MAX 検出 1'b0 : 0 固定 1'b1 : MAX.[d*2+e].in 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b28~b27	予約ビット	初期値を保持してください。	R/W
b26	予約ビット	読むと 0 が読み出されます。	R
b25~b23	予約ビット	初期値を保持してください。	R/W
b22	bOREDUP_ [d*2+1]	PWM[k]の Up/Down 入力[d*2+e]、Set 信号での MIN 検出 1'b0 : 0 固定 1'b1 : MIN.[d*2+e].in 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b21~b20	予約ビット	初期値を保持してください。	R/W
b19	予約ビット	読むと 0 が読み出されます。	R
b18~b16	予約ビット	初期値を保持してください。	R/W
b15~b14	bUPDOWN_MODE_ [d*2]	bUPDOWN_MODE_[d*2+1]を参照、本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W
b13	bOREDDOWN_ [d*2]	bOREDDOWN_[d*2+1]を参照、本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W
b12~b11	予約ビット	初期値を保持してください。	R/W
b10	予約ビット	読むと 0 が読み出されます。	R
b9~b7	予約ビット	初期値を保持してください。	R/W
b6	bOREDUP_ [d*2]	bOREDUP_[d*2+1]を参照、本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W
b5~b4	予約ビット	初期値を保持してください。	R/W
b3	予約ビット	読むと 0 が読み出されます。	R
b2~b0	予約ビット	初期値を保持してください。	R/W

1.4.16 rPWMTimer_PWM_TRIG0_[k] — PWM モジュール Trig0 入力制御 (k = 0~1)

アドレス 4006 9070h+400h×k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	bTRIG0_7			—	bTRIG0_6			—	bTRIG0_5			—	bTRIG0_4		
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	bTRIG0_3			—	bTRIG0_2			—	bTRIG0_1			—	bTRIG0_0		
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.18 rPWMTimer_PWM_TRIG0_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	予約ビット	読むと 0 が読み出されます。	R
b30~b28	bTRIG0_7	PWM[k]シングルカウンタ[i]の Trig0 入力 3'b000 : 0 固定 3'b001 : Trig0.[i] それ以外 : 予約 本ビット : i = 7	R/W
b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	bTRIG0_6	上記参照、本ビット : i = 6	R/W
b23	予約ビット	読むと 0 が読み出されます。	R
b22~b20	bTRIG0_5	上記参照、本ビット : i = 5	R/W
b19	予約ビット	読むと 0 が読み出されます。	R
b18~b16	bTRIG0_4	上記参照、本ビット : i = 4	R/W
b15	予約ビット	読むと 0 が読み出されます。	R
b14~b12	bTRIG0_3	上記参照、本ビット : i = 3	R/W
b11	予約ビット	読むと 0 が読み出されます。	R
b10~b8	bTRIG0_2	上記参照、本ビット : i = 2	R/W
b7	予約ビット	読むと 0 が読み出されます。	R
b6~b4	bTRIG0_1	上記参照、本ビット : i = 1	R/W
b3	予約ビット	読むと 0 が読み出されます。	R
b2~b0	bTRIG0_0	上記参照、本ビット : i = 0	R/W

1.4.17 rPWMTimer_PWM_TRIG1_[k] — PWM モジュール Trig1 入力制御 (k = 0~1)

アドレス 4006 9080h+400h×k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	bTRIG1_7			—	bTRIG1_6			—	bTRIG1_5			—	bTRIG1_4		
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	bTRIG1_3			—	bTRIG1_2			—	bTRIG1_1			—	bTRIG1_0		
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.19 rPWMTimer_PWM_TRIG1_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	予約ビット	読むと 0 が読み出されます。	R
b30~b28	bTRIG1_7	PWM[k]シングルカウンタ[i]の Trig1 入力 3'b000 : 0 固定 3'b001 : Trig1.[i] それ以外 : 予約 本ビット : i = 7	R/W
b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	bTRIG1_6	上記参照、本ビット : i = 6	R/W
b23	予約ビット	読むと 0 が読み出されます。	R
b22~b20	bTRIG1_5	上記参照、本ビット : i = 5	R/W
b19	予約ビット	読むと 0 が読み出されます。	R
b18~b16	bTRIG1_4	上記参照、本ビット : i = 4	R/W
b15	予約ビット	読むと 0 が読み出されます。	R
b14~b12	bTRIG1_3	上記参照、本ビット : i = 3	R/W
b11	予約ビット	読むと 0 が読み出されます。	R
b10~b8	bTRIG1_2	上記参照、本ビット : i = 2	R/W
b7	予約ビット	読むと 0 が読み出されます。	R
b6~b4	bTRIG1_1	上記参照、本ビット : i = 1	R/W
b3	予約ビット	読むと 0 が読み出されます。	R
b2~b0	bTRIG1_0	上記参照、本ビット : i = 0	R/W

1.4.18 rPWMTimer_PWM_OUTPUTGEN[d]_k — PWM モジュール出力生成制御 (d = 0~3, k = 0~1)

アドレス 4006 90C0h+400h×k+4h×d

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—			—			bCMP1_STOPPED_MODE_ [d*2+1]		—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			—			bCMP1_STOPPED_MODE_ [d*2]		—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.20 rPWMTimer_PWM_OUTPUTGEN[d]_kレジスタの内容

ビット位置	ビット名	機能	R/W
b31~b29	予約ビット	読むと 0 が読み出されます。	R
b28~b26	予約ビット	初期値を保持してください。	R/W
b25~b24	bCMP1_STOPPED_MODE_ [d*2+1]	カウンタインエーブル無効時の PWM[k] シングルカウンタ [d*2+e] の CMP1 出力 2'b00 : リセット (0 固定) 2'b01 : セット (1 固定) それ以外 : 予約 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b10	予約ビット	初期値を保持してください。	R/W
b9~b8	bCMP1_STOPPED_MODE_ [d*2]	bCMP1_STOPPED_MODE_[d*2+1] を参照 本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.19 rPWMTimer_PWM_BLOCK_CLOCK_[k] —
PWM モジュール Block_Clock 制御 (k = 0~1)

本レジスタの書き込みにより、プリスケアラはリスタートします。

アドレス 4006 90E0h+400h×k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—						bPWM_CLOCK_PRESCALER_1									
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—						bPWM_CLOCK_PRESCALER_0									
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.21 rPWMTimer_PWM_BLOCK_CLOCK_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b26	予約ビット	読むと 0 が読み出されます。	R
b25~b16	bPWM_CLOCK_PRESCALER_1	PWM[k]の Block_Clock[e]プリスケアラの分周設定 10'd0 : PWM_PCLK/2 10'd1 : PWM_PCLK/4 10'd2 : PWM_PCLK/6 10'dX : PWM_PCLK/((X+1)×2) 10'd1023 : PWM_PCLK/2048 本ビット : e = 1	R/W
b15~b10	予約ビット	読むと 0 が読み出されます。	R
b9~b0	bPWM_CLOCK_PRESCALER_0	上記参照、本ビット : e = 0	R/W

1.4.20 rPWMTimer_TIMEBASE_[i]_[k] —
16 ビット Timebase 現カウンタ値 (i = 0~7, k = 0~1)

アドレス 4006 9100h+400h×k+4h×i

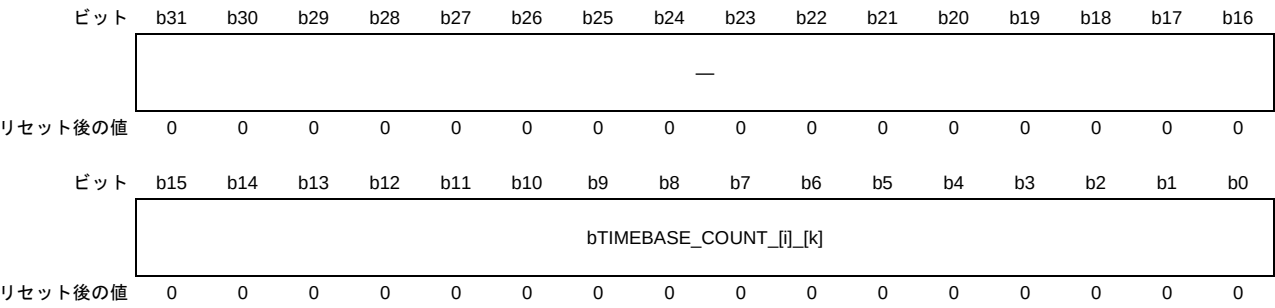


表 1.22 rPWMTimer_TIMEBASE_[i]_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	読むと 0 が読み出されます。	R
b15~b0	bTIMEBASE_COUNT_[i]_[k]	PWM[k]シングルカウンタ[i]の Timebase 現カウンタ値	R/W

1.4.21 rPWMTimer_TIMEBASE_[n*2][n*2+1]_[k] —
32 ビットカスケード Timebase 現カウンタ値 (n = 0~3, k = 0~1)

アドレス 4006 9120h+400h×k+4h×n

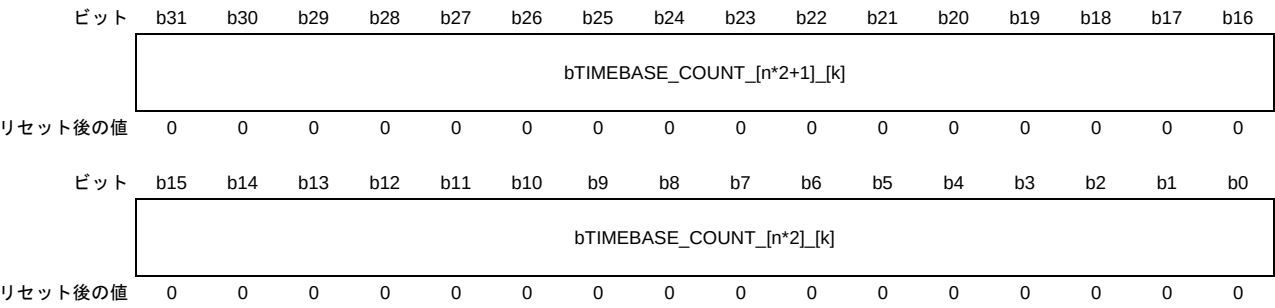


表 1.23 rPWMTimer_TIMEBASE_[n*2][n*2+1]_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	bTIMEBASE_COUNT_[n*2+1]_[k]	PWM[k] シングルカウンタ [n*2+e] の Timebase 現カウンタ値 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b15~b0	bTIMEBASE_COUNT_[n*2]_[k]	上記参照、本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W

1.4.22 rPWMTimer_TIMEBASE_ $[(n+1)*2-1] [(n+1)*2]_ [k]$ —
32 ビットカスケード Timebase 現カウンタ値 ($n = 0 \sim 2, k = 0 \sim 1$)

アドレス 4006 9130h+400h×k+4h×n

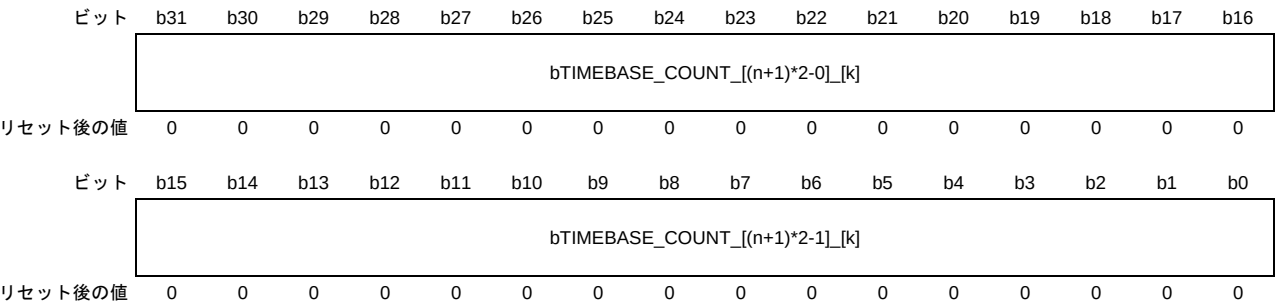


表 1.24 rPWMTimer_TIMEBASE_ $[(n+1)*2-1] [(n+1)*2]_ [k]$ レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	bTIMEBASE_COUNT_ $[(n+1)*2-0]_ [k]$	PWM[k] シングルカウンタ $[(n+1)*2-e]$ の Timebase 現カウンタ値 本ビット : $e = 0$ (シングルカウンタ 2, 4, 6)	R/W
b15～b0	bTIMEBASE_COUNT_ $[(n+1)*2-1]_ [k]$	上記参照、本ビット : $e = 1$ (シングルカウンタ 1, 3, 5)	R/W

1.4.23 rPWMTimer_SCALEVALUE_[i]_[k] —
カスケード値設定 (i = 0~7, k = 0~1)

アドレス 4006 9140h+400h×k+4h×i

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—								bSCALEVALUE_[i]_[k]							
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

表 1.25 rPWMTimer_SCALEVALUE_[i]_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b8	予約ビット	読むと 0 が読み出されます。	R
b7~b0	bSCALEVALUE_[i]_[k]]	PWM[k] シングルカウンタ [i] の Timebase カスケード値 注意) PWM[k] シングルカウンタ [i] のカスケード論理が有効な場合 (rPWMTimer_PWM_MODE_[k].bPWM_CASCADE_MODE_[i] = 1)、本ビットを 8'h00 に設定してください。それ以外は、8'h01 に設定してください。	R/W

1.4.24 rPWMTimer_TIMEBASE_MUX_[q]_[k] — コンペア／キャプチャ入力選択制御 (q = 0~1, k = 0~1)

アドレス 4006 91A0h+400h×k+4h×q

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—		—				bTIMEBASE_MUX_OR2_ [q*4+3]	bTIMEBASE_MUX_OR0_ [q*4+3]	—		—				bTIMEBASE_MUX_OR2_ [q*4+2]	bTIMEBASE_MUX_OR0_ [q*4+2]
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—		—				bTIMEBASE_MUX_OR2_ [q*4+1]	bTIMEBASE_MUX_OR0_ [q*4+1]	—		—				bTIMEBASE_MUX_OR2_ [q*4]	bTIMEBASE_MUX_OR0_ [q*4]
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.26 rPWMTimer_TIMEBASE_MUX_[q]_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b30	予約ビット	読むと 0 が読み出されます。	R
b29~b26	予約ビット	初期値を保持してください。	R/W
b25	bTIMEBASE_MUX_OR2_ [q*4+3]	シングルカウンタ[q*4+e]の OR2 へのバッファ転送有効 本ビットは、1 に設定してください。 本ビット : e = 3 (シングルカウンタ 3, 7)	R/W
b24	bTIMEBASE_MUX_OR0_ [q*4+3]	シングルカウンタ[q*4+e]の OR0 へのバッファ転送有効 本ビットは、1 に設定してください。 本ビット : e = 3 (シングルカウンタ 3, 7)	R/W
b23~b22	予約ビット	読むと 0 が読み出されます。	R
b21~b18	予約ビット	初期値を保持してください。	R/W
b17	bTIMEBASE_MUX_OR2_ [q*4+2]	bTIMEBASE_MUX_OR2_[q*4+3]を参照 本ビット : e = 2 (シングルカウンタ 2, 6)	R/W
b16	bTIMEBASE_MUX_OR0_ [q*4+2]	bTIMEBASE_MUX_OR0_[q*4+3]を参照 本ビット : e = 2 (シングルカウンタ 2, 6)	R/W
b15~b14	予約ビット	読むと 0 が読み出されます。	R
b13~b10	予約ビット	初期値を保持してください。	R/W
b9	bTIMEBASE_MUX_OR2_ [q*4+1]	bTIMEBASE_MUX_OR2_[q*4+3]を参照 本ビット : e = 1 (シングルカウンタ 1, 5)	R/W
b8	bTIMEBASE_MUX_OR0_ [q*4+1]	bTIMEBASE_MUX_OR0_[q*4+3]を参照 本ビット : e = 1 (シングルカウンタ 1, 5)	R/W
b7~b6	予約ビット	読むと 0 が読み出されます。	R
b5~b2	予約ビット	初期値を保持してください。	R/W
b1	bTIMEBASE_MUX_OR2_ [q*4]	bTIMEBASE_MUX_OR2_[q*4+3]を参照 本ビット : e = 0 (シングルカウンタ 0, 4)	R/W
b0	bTIMEBASE_MUX_OR0_ [q*4]	bTIMEBASE_MUX_OR0_[q*4+3]を参照 本ビット : e = 0 (シングルカウンタ 0, 4)	R/W

1.4.25 rPWMTimer_HWSTORE_CONF_[i]_[k] — PWM モジュール Hardware Store トリガ制御 (i = 0~7, k = 0~1)

アドレス 4006 91B0h+400h×k+4h×i

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	bHWSTORE_SYNC_TRIG_2	—	—	—	—
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	bHWSTORE_SYNC_TRIG_0	—	—	—	—
リセット後の値	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0

表 1.27 rPWMTimer_HWSTORE_CONF_[i]_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	予約ビット	初期値を保持してください。	R/W
b30~b28	予約ビット	読むと 0 が読み出されます。	R
b27~b25	予約ビット	初期値を保持してください。	R/W
b24~b21	予約ビット	読むと 0 が読み出されます。	R
b20	bHWSTORE_SYNC_TRIG_2	PWM[k]シングルカウンタ[i]のオプションレジスタ OR2 のトリガ同期設定 1'b0 : 非同期 1'b1 : Timebase クロック同期 キャプチャ値を転送する場合は 0 を、それ以外は 1 に設定してください。	R/W
b19~b16	予約ビット	初期値を保持してください。	R/W
b15~b14	予約ビット	読むと 0 が読み出されます。	R
b13	予約ビット	初期値を保持してください。	R/W
b12	予約ビット	読むと 0 が読み出されます。	R
b11~b8	予約ビット	初期値を保持してください。	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4	bHWSTORE_SYNC_TRIG_0	PWM[k]シングルカウンタ[i]のオプションレジスタ OR0 のトリガ同期設定 1'b0 : 非同期 1'b1 : Timebase クロック同期 キャプチャ値を転送する場合は 0 を、それ以外は 1 に設定してください。	R/W
b3~b0	予約ビット	初期値を保持してください。	R/W

1.4.26 rPWMTimer_HWSTORE_MUX_[i]_[k] — PWM モジュール Hardware Store トリガ選択制御 (i = 0~7, k = 0~1)

アドレス 4006 91D0h + 400h × k + 4h × i

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—				bHWSTORE_TRIG2_GENSY				bHWSTORE_EDGE_OR3	bHWSTORE_EDGE_OR2	bHWSTORE_EDGE_OR1	bHWSTORE_EDGE_OR0				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—		bHWSTORE_MUXED_TRIG_OR3	—		bHWSTORE_MUXED_TRIG_OR2	—		bHWSTORE_MUXED_TRIG_OR1	—		bHWSTORE_MUXED_TRIG_OR0				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.28 rPWMTimer_HWSTORE_MUX_[i]_[k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b28	予約ビット	初期値を保持してください。	R
b27~b24	bHWSTORE_TRIG2_GENSY	PWM[k] シングルカウンタ [i] の Trig2 入力選択 4'b0000 : Trig2 = 予約 4'b0001 : Trig2 = MIN.i.in 4'b0010 : Trig2 = MAX.i.in 4'b0011 : Trig2 = MAX.i.in および MIN.i.in それ以外 : Trig2 = 予約	R/W
b23~b22	bHWSTORE_EDGE_OR3	PWM[k] シングルカウンタ [i] のオプションレジスタ OR[r] の入力トリガエッジ選択 2'b00 : ダイレクト入力 2'b01 : 立ち上がりエッジパルス 2'b10 : 立ち下がりエッジパルス 2'b11 : 立ち上がり／立ち下がりの両エッジパルス 本ビット : r = 3	R/W
b21~b20	bHWSTORE_EDGE_OR2	bHWSTORE_EDGE_OR3 を参照、本ビット : r = 2	R/W
b19~b18	bHWSTORE_EDGE_OR1	bHWSTORE_EDGE_OR3 を参照、本ビット : r = 1	R/W
b17~b16	bHWSTORE_EDGE_OR0	bHWSTORE_EDGE_OR3 を参照、本ビット : r = 0	R/W
b15~b14	予約ビット	初期値を保持してください。	R
b13~b12	bHWSTORE_MUXED_TRIG_OR3	PWM[k] シングルカウンタ [i] のオプションレジスタ OR[r] 入力トリガ選択 2'b00 : 0 固定 (トリガなし) 2'b01 : Trig0 2'b10 : Trig1 2'b11 : Trig2 本ビット : r = 3	R/W
b11~b10	予約ビット	初期値を保持してください。	R
b9~b8	bHWSTORE_MUXED_TRIG_OR2	bHWSTORE_MUXED_TRIG_OR3 を参照、本ビット : r = 2	R/W
b7~b6	予約ビット	初期値を保持してください。	R
b5~b4	bHWSTORE_MUXED_TRIG_OR1	bHWSTORE_MUXED_TRIG_OR3 を参照、本ビット : r = 1	R/W
b3~b2	予約ビット	初期値を保持してください。	R
b1~b0	bHWSTORE_MUXED_TRIG_OR0	bHWSTORE_MUXED_TRIG_OR3 を参照、本ビット : r = 0	R/W

1.4.27 rPWMTimer_HWSTORE_LOCK_[k] — オプションレジスタロックおよびロックステータス (k = 0~1)

アドレス 4006 91F0h + 400h × k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bHWST ORE_O R3_LO CK_7	bHWST ORE_O R2_LO CK_7	bHWST ORE_O R1_LO CK_7	bHWST ORE_O R0_LO CK_7	bHWST ORE_O R3_LO CK_6	bHWST ORE_O R2_LO CK_6	bHWST ORE_O R1_LO CK_6	bHWST ORE_O R0_LO CK_6	bHWST ORE_O R3_LO CK_5	bHWST ORE_O R2_LO CK_5	bHWST ORE_O R1_LO CK_5	bHWST ORE_O R0_LO CK_5	bHWST ORE_O R3_LO CK_4	bHWST ORE_O R2_LO CK_4	bHWST ORE_O R1_LO CK_4	bHWST ORE_O R0_LO CK_4
リセット後の値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bHWST ORE_O R3_LO CK_3	bHWST ORE_O R2_LO CK_3	bHWST ORE_O R1_LO CK_3	bHWST ORE_O R0_LO CK_3	bHWST ORE_O R3_LO CK_2	bHWST ORE_O R2_LO CK_2	bHWST ORE_O R1_LO CK_2	bHWST ORE_O R0_LO CK_2	bHWST ORE_O R3_LO CK_1	bHWST ORE_O R2_LO CK_1	bHWST ORE_O R1_LO CK_1	bHWST ORE_O R0_LO CK_1	bHWST ORE_O R3_LO CK_0	bHWST ORE_O R2_LO CK_0	bHWST ORE_O R1_LO CK_0	bHWST ORE_O R0_LO CK_0
リセット後の値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 1.29 rPWMTimer_HWSTORE_LOCK_[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	bHWSTORE_OR3_L OCK_7	シングルカウンタ[i]のオプションレジスタ OR[r]のロックステータス リードステータス： 1'b0：オプションレジスタはロック解除状態 1'b1：オプションレジスタはロック状態 1を書き込むとオプションレジスタはロック、0の書き込みは無効です。 本ビット：i = 7, r = 3 (シングルカウンタ 7 の OR3) ロック解除は、「1.4.28 rPWMTimer_HWSTORE_UNLOCK_[k] — オプションレジスタロック解除 (k = 0~1)」を参照してください。	R/W
b30	bHWSTORE_OR2_L OCK_7	上記参照、本ビット：i = 7, r = 2 (シングルカウンタ 7 の OR2)	R/W
b29	bHWSTORE_OR1_L OCK_7	上記参照、本ビット：i = 7, r = 1 (シングルカウンタ 7 の OR1)	R/W
b28	bHWSTORE_OR0_L OCK_7	上記参照、本ビット：i = 7, r = 0 (シングルカウンタ 7 の OR0)	R/W
b27	bHWSTORE_OR3_L OCK_6	上記参照、本ビット：i = 6, r = 3 (シングルカウンタ 6 の OR3)	R/W
b26	bHWSTORE_OR2_L OCK_6	上記参照、本ビット：i = 6, r = 2 (シングルカウンタ 6 の OR2)	R/W
b25	bHWSTORE_OR1_L OCK_6	上記参照、本ビット：i = 6, r = 1 (シングルカウンタ 6 の OR1)	R/W
b24	bHWSTORE_OR0_L OCK_6	上記参照、本ビット：i = 6, r = 0 (シングルカウンタ 6 の OR0)	R/W
b23	bHWSTORE_OR3_L OCK_5	上記参照、本ビット：i = 5, r = 3 (シングルカウンタ 5 の OR3)	R/W
b22	bHWSTORE_OR2_L OCK_5	上記参照、本ビット：i = 5, r = 2 (シングルカウンタ 5 の OR2)	R/W
b21	bHWSTORE_OR1_L OCK_5	上記参照、本ビット：i = 5, r = 1 (シングルカウンタ 5 の OR1)	R/W
b20	bHWSTORE_OR0_L OCK_5	上記参照、本ビット：i = 5, r = 0 (シングルカウンタ 5 の OR0)	R/W
b19	bHWSTORE_OR3_L OCK_4	上記参照、本ビット：i = 4, r = 3 (シングルカウンタ 4 の OR3)	R/W
b18	bHWSTORE_OR2_L OCK_4	上記参照、本ビット：i = 4, r = 2 (シングルカウンタ 4 の OR2)	R/W
b17	bHWSTORE_OR1_L OCK_4	上記参照、本ビット：i = 4, r = 1 (シングルカウンタ 4 の OR1)	R/W

ビット位置	ビット名	機能	R/W
b16	bHWSTORE_OR0_L OCK_4	上記参照、本ビット : i = 4, r = 0 (シングルカウンタ 4 の OR0)	R/W
b15	bHWSTORE_OR3_L OCK_3	上記参照、本ビット : i = 3, r = 3 (シングルカウンタ 3 の OR3)	R/W
b14	bHWSTORE_OR2_L OCK_3	上記参照、本ビット : i = 3, r = 2 (シングルカウンタ 3 の OR2)	R/W
b13	bHWSTORE_OR1_L OCK_3	上記参照、本ビット : i = 3, r = 1 (シングルカウンタ 3 の OR1)	R/W
b12	bHWSTORE_OR0_L OCK_3	上記参照、本ビット : i = 3, r = 0 (シングルカウンタ 3 の OR0)	R/W
b11	bHWSTORE_OR3_L OCK_2	上記参照、本ビット : i = 2, r = 3 (シングルカウンタ 2 の OR3)	R/W
b10	bHWSTORE_OR2_L OCK_2	上記参照、本ビット : i = 2, r = 2 (シングルカウンタ 2 の OR2)	R/W
b9	bHWSTORE_OR1_L OCK_2	上記参照、本ビット : i = 2, r = 1 (シングルカウンタ 2 の OR1)	R/W
b8	bHWSTORE_OR0_L OCK_2	上記参照、本ビット : i = 2, r = 0 (シングルカウンタ 2 の OR0)	R/W
b7	bHWSTORE_OR3_L OCK_1	上記参照、本ビット : i = 1, r = 3 (シングルカウンタ 1 の OR3)	R/W
b6	bHWSTORE_OR2_L OCK_1	上記参照、本ビット : i = 1, r = 2 (シングルカウンタ 1 の OR2)	R/W
b5	bHWSTORE_OR1_L OCK_1	上記参照、本ビット : i = 1, r = 1 (シングルカウンタ 1 の OR1)	R/W
b4	bHWSTORE_OR0_L OCK_1	上記参照、本ビット : i = 1, r = 0 (シングルカウンタ 1 の OR0)	R/W
b3	bHWSTORE_OR3_L OCK_0	上記参照、本ビット : i = 0, r = 3 (シングルカウンタ 0 の OR3)	R/W
b2	bHWSTORE_OR2_L OCK_0	上記参照、本ビット : i = 0, r = 2 (シングルカウンタ 0 の OR2)	R/W
b1	bHWSTORE_OR1_L OCK_0	上記参照、本ビット : i = 0, r = 1 (シングルカウンタ 0 の OR1)	R/W
b0	bHWSTORE_OR0_L OCK_0	上記参照、本ビット : i = 0, r = 0 (シングルカウンタ 0 の OR0)	R/W

1.4.28 rPWMTimer_HWSTORE_UNLOCK[k] — オプションレジスタロック解除 (k = 0~1)

アドレス 4006 91F4h + 400h × k

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bHWSTORE_OR3_UNLOCK_7	bHWSTORE_OR2_UNLOCK_7	bHWSTORE_OR1_UNLOCK_7	bHWSTORE_OR0_UNLOCK_7	bHWSTORE_OR3_UNLOCK_6	bHWSTORE_OR2_UNLOCK_6	bHWSTORE_OR1_UNLOCK_6	bHWSTORE_OR0_UNLOCK_6	bHWSTORE_OR3_UNLOCK_5	bHWSTORE_OR2_UNLOCK_5	bHWSTORE_OR1_UNLOCK_5	bHWSTORE_OR0_UNLOCK_5	bHWSTORE_OR3_UNLOCK_4	bHWSTORE_OR2_UNLOCK_4	bHWSTORE_OR1_UNLOCK_4	bHWSTORE_OR0_UNLOCK_4
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bHWSTORE_OR3_UNLOCK_3	bHWSTORE_OR2_UNLOCK_3	bHWSTORE_OR1_UNLOCK_3	bHWSTORE_OR0_UNLOCK_3	bHWSTORE_OR3_UNLOCK_2	bHWSTORE_OR2_UNLOCK_2	bHWSTORE_OR1_UNLOCK_2	bHWSTORE_OR0_UNLOCK_2	bHWSTORE_OR3_UNLOCK_1	bHWSTORE_OR2_UNLOCK_1	bHWSTORE_OR1_UNLOCK_1	bHWSTORE_OR0_UNLOCK_1	bHWSTORE_OR3_UNLOCK_0	bHWSTORE_OR2_UNLOCK_0	bHWSTORE_OR1_UNLOCK_0	bHWSTORE_OR0_UNLOCK_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.30 rPWMTimer_HWSTORE_UNLOCK[k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31	bHWSTORE_OR3_UNLOCK_7	シングルカウンタ[i]のオプションレジスタ OR[r]のロック解除 1 を書き込むとロック解除、0 の書き込みは無効です。 本ビット : i = 7, r = 3 (シングルカウンタ 7 の OR3)	W
b30	bHWSTORE_OR2_UNLOCK_7	上記参照、本ビット : i = 7, r = 2 (シングルカウンタ 7 の OR2)	W
b29	bHWSTORE_OR1_UNLOCK_7	上記参照、本ビット : i = 7, r = 1 (シングルカウンタ 7 の OR1)	W
b28	bHWSTORE_OR0_UNLOCK_7	上記参照、本ビット : i = 7, r = 0 (シングルカウンタ 7 の OR0)	W
b27	bHWSTORE_OR3_UNLOCK_6	上記参照、本ビット : i = 6, r = 3 (シングルカウンタ 6 の OR3)	W
b26	bHWSTORE_OR2_UNLOCK_6	上記参照、本ビット : i = 6, r = 2 (シングルカウンタ 6 の OR2)	W
b25	bHWSTORE_OR1_UNLOCK_6	上記参照、本ビット : i = 6, r = 1 (シングルカウンタ 6 の OR1)	W
b24	bHWSTORE_OR0_UNLOCK_6	上記参照、本ビット : i = 6, r = 0 (シングルカウンタ 6 の OR0)	W
b23	bHWSTORE_OR3_UNLOCK_5	上記参照、本ビット : i = 5, r = 3 (シングルカウンタ 5 の OR3)	W
b22	bHWSTORE_OR2_UNLOCK_5	上記参照、本ビット : i = 5, r = 2 (シングルカウンタ 5 の OR2)	W
b21	bHWSTORE_OR1_UNLOCK_5	上記参照、本ビット : i = 5, r = 1 (シングルカウンタ 5 の OR1)	W
b20	bHWSTORE_OR0_UNLOCK_5	上記参照、本ビット : i = 5, r = 0 (シングルカウンタ 5 の OR0)	W
b19	bHWSTORE_OR3_UNLOCK_4	上記参照、本ビット : i = 4, r = 3 (シングルカウンタ 4 の OR3)	W
b18	bHWSTORE_OR2_UNLOCK_4	上記参照、本ビット : i = 4, r = 2 (シングルカウンタ 4 の OR2)	W
b17	bHWSTORE_OR1_UNLOCK_4	上記参照、本ビット : i = 4, r = 1 (シングルカウンタ 4 の OR1)	W
b16	bHWSTORE_OR0_UNLOCK_4	上記参照、本ビット : i = 4, r = 0 (シングルカウンタ 4 の OR0)	W
b15	bHWSTORE_OR3_UNLOCK_3	上記参照、本ビット : i = 3, r = 3 (シングルカウンタ 3 の OR3)	W

ビット位置	ビット名	機能	R/W
b14	bHWSTORE_OR2_UNLOCK_3	上記参照、本ビット : i = 3, r = 2 (シングルカウンタ 3 の OR2)	W
b13	bHWSTORE_OR1_UNLOCK_3	上記参照、本ビット : i = 3, r = 1 (シングルカウンタ 3 の OR1)	W
b12	bHWSTORE_OR0_UNLOCK_3	上記参照、本ビット : i = 3, r = 0 (シングルカウンタ 3 の OR0)	W
b11	bHWSTORE_OR3_UNLOCK_2	上記参照、本ビット : i = 2, r = 3 (シングルカウンタ 2 の OR3)	W
b10	bHWSTORE_OR2_UNLOCK_2	上記参照、本ビット : i = 2, r = 2 (シングルカウンタ 2 の OR2)	W
b9	bHWSTORE_OR1_UNLOCK_2	上記参照、本ビット : i = 2, r = 1 (シングルカウンタ 2 の OR1)	W
b8	bHWSTORE_OR0_UNLOCK_2	上記参照、本ビット : i = 2, r = 0 (シングルカウンタ 2 の OR0)	W
b7	bHWSTORE_OR3_UNLOCK_1	上記参照、本ビット : i = 1, r = 3 (シングルカウンタ 1 の OR3)	W
b6	bHWSTORE_OR2_UNLOCK_1	上記参照、本ビット : i = 1, r = 2 (シングルカウンタ 1 の OR2)	W
b5	bHWSTORE_OR1_UNLOCK_1	上記参照、本ビット : i = 1, r = 1 (シングルカウンタ 1 の OR1)	W
b4	bHWSTORE_OR0_UNLOCK_1	上記参照、本ビット : i = 1, r = 0 (シングルカウンタ 1 の OR0)	W
b3	bHWSTORE_OR3_UNLOCK_0	上記参照、本ビット : i = 0, r = 3 (シングルカウンタ 0 の OR3)	W
b2	bHWSTORE_OR2_UNLOCK_0	上記参照、本ビット : i = 0, r = 2 (シングルカウンタ 0 の OR2)	W
b1	bHWSTORE_OR1_UNLOCK_0	上記参照、本ビット : i = 0, r = 1 (シングルカウンタ 0 の OR1)	W
b0	bHWSTORE_OR0_UNLOCK_0	上記参照、本ビット : i = 0, r = 0 (シングルカウンタ 0 の OR0)	W

1.4.29 rPWMTimer_OPTIONREG[r][i][k] —
16 ビット Timebase オプションレジスタ[r]値 (i = 0~7, r = 0~3, k = 0~1)

アドレス 4006 9200h+400h×k+40h×r+4h×i

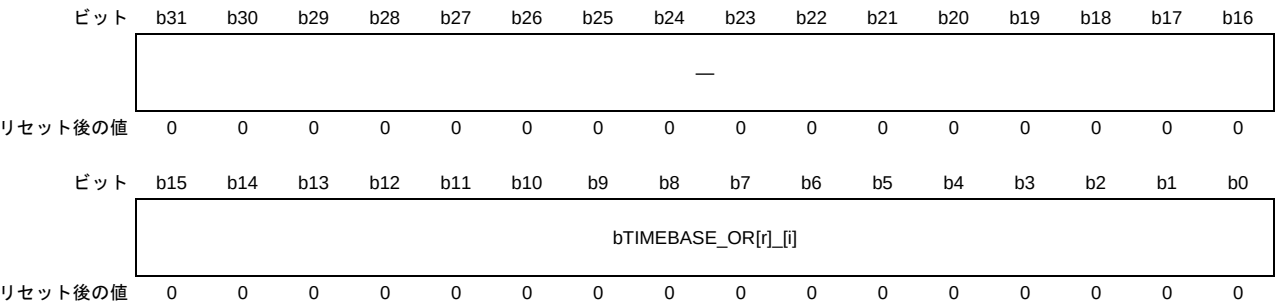


表 1.31 rPWMTimer_OPTIONREG[r][i][k]レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	読むと 0 が読み出されます。	R
b15~b0	bTIMEBASE_OR[r][i]]	PWM[k]シングルカウンタ[i]のオプションレジスタ OR[r]値	R/W

1.4.30 rPWMTimer_OPTIONREG[r]_[n*2][n*2+1]_ [k] —
32 ビットカスケード Timebase オプションレジスタ[r]値 (n = 0~3, r = 0~3, k = 0~1)

アドレス 4006 9220h+400h×k+40h×r+4h×n

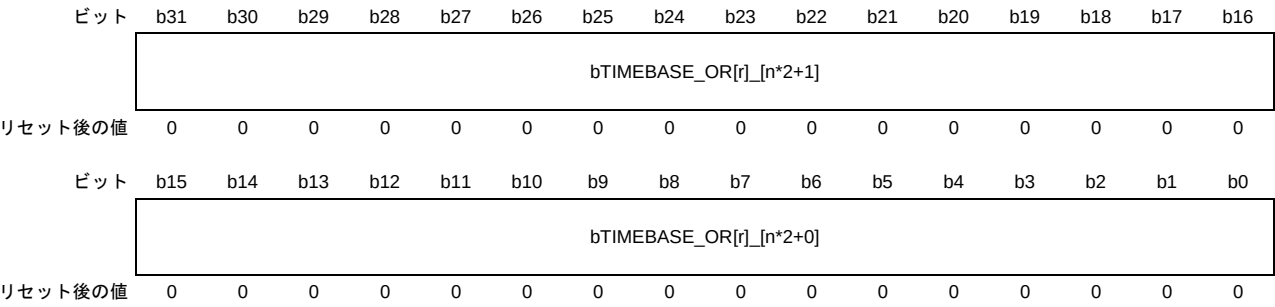


表 1.32 rPWMTimer_OPTIONREG[r]_[n*2][n*2+1]_ [k] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	bTIMEBASE_OR[r]_[n*2+1]	PWM[k] シングルカウンタ[n*2+e] のオプションレジスタ OR[r] 値 本ビット : e = 1 (シングルカウンタ 1, 3, 5, 7)	R/W
b15~b0	bTIMEBASE_OR[r]_[n*2+0]	上記参照、本ビット : e = 0 (シングルカウンタ 0, 2, 4, 6)	R/W

1.4.31 rPWMTimer_OPTIONREG[r]_[(n+1)*2-1][(n+1)*2]_k — 32 ビットカスケード Timebase オプションレジスタ[r]値 (n = 0~2, r = 0~3, k = 0~1)

アドレス 4006 9230h + 400h × k + 40h × r + 4h × n

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	bTIMEBASE_OR[r]_[(n+1)*2-0]															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bTIMEBASE_OR[r]_[(n+1)*2-1]															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.33 rPWMTimer_OPTIONREG[r]_[(n+1)*2-1][(n+1)*2]_k レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	bTIMEBASE_OR[r]_[(n+1)*2-0]	PWM[k] シングルカウンタ [(n+1)*2-e] のオプションレジスタ OR[r] 値 本ビット : e = 0 (シングルカウンタ 2, 4, 6)	
b15~b0	bTIMEBASE_OR[r]_[(n+1)*2-1]	上記参照、本ビット : e = 1 (シングルカウンタ 1, 3, 5)	

1.4.32 rPWMTimer_PWM_SOFTSTART — PWM モジュールカウントイネーブルスタート

アドレス 4006 A010h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bSOFT START _7_1	bSOFT START _6_1	bSOFT START _5_1	bSOFT START _4_1	bSOFT START _3_1	bSOFT START _2_1	bSOFT START _1_1	bSOFT START _0_1	bSOFT START _7_0	bSOFT START _6_0	bSOFT START _5_0	bSOFT START _4_0	bSOFT START _3_0	bSOFT START _2_0	bSOFT START _1_0	bSOFT START _0_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.34 rPWMTimer_PWM_SOFTSTART レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	W
b15	bSOFTSTART_7_1	PWM[k]シングルカウンタ[i]のスタートトリガ 1 を書き込むとカウントイネーブルは有効、0 の書き込みは無効です。 本ビット：i = 7, k = 1 (PWM1 のシングルカウンタ 7)	W
b14	bSOFTSTART_6_1	上記参照、本ビット：i = 6, k = 1 (PWM1 のシングルカウンタ 6)	W
b13	bSOFTSTART_5_1	上記参照、本ビット：i = 5, k = 1 (PWM1 のシングルカウンタ 5)	W
b12	bSOFTSTART_4_1	上記参照、本ビット：i = 4, k = 1 (PWM1 のシングルカウンタ 4)	W
b11	bSOFTSTART_3_1	上記参照、本ビット：i = 3, k = 1 (PWM1 のシングルカウンタ 3)	W
b10	bSOFTSTART_2_1	上記参照、本ビット：i = 2, k = 1 (PWM1 のシングルカウンタ 2)	W
b9	bSOFTSTART_1_1	上記参照、本ビット：i = 1, k = 1 (PWM1 のシングルカウンタ 1)	W
b8	bSOFTSTART_0_1	上記参照、本ビット：i = 0, k = 1 (PWM1 のシングルカウンタ 0)	W
b7	bSOFTSTART_7_0	上記参照、本ビット：i = 7, k = 0 (PWM0 のシングルカウンタ 7)	W
b6	bSOFTSTART_6_0	上記参照、本ビット：i = 6, k = 0 (PWM0 のシングルカウンタ 6)	W
b5	bSOFTSTART_5_0	上記参照、本ビット：i = 5, k = 0 (PWM0 のシングルカウンタ 5)	W
b4	bSOFTSTART_4_0	上記参照、本ビット：i = 4, k = 0 (PWM0 のシングルカウンタ 4)	W
b3	bSOFTSTART_3_0	上記参照、本ビット：i = 3, k = 0 (PWM0 のシングルカウンタ 3)	W
b2	bSOFTSTART_2_0	上記参照、本ビット：i = 2, k = 0 (PWM0 のシングルカウンタ 2)	W
b1	bSOFTSTART_1_0	上記参照、本ビット：i = 1, k = 0 (PWM0 のシングルカウンタ 1)	W
b0	bSOFTSTART_0_0	上記参照、本ビット：i = 0, k = 0 (PWM0 のシングルカウンタ 0)	W

1.4.33 rPWMTimer_PWM_SOFTSTOP — PWM モジュールカウントイネーブルストップ

アドレス 4006 A018h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bSOFTSTOP_7_1	bSOFTSTOP_6_1	bSOFTSTOP_5_1	bSOFTSTOP_4_1	bSOFTSTOP_3_1	bSOFTSTOP_2_1	bSOFTSTOP_1_1	bSOFTSTOP_0_1	bSOFTSTOP_7_0	bSOFTSTOP_6_0	bSOFTSTOP_5_0	bSOFTSTOP_4_0	bSOFTSTOP_3_0	bSOFTSTOP_2_0	bSOFTSTOP_1_0	bSOFTSTOP_0_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.35 rPWMTimer_PWM_SOFTSTOP レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	W
b15	bSOFTSTOP_7_1	PWM[k]シングルカウンタ[i]のストップトリガ 1 を書き込むとカウントイネーブルは無効、0 の書き込みは無効です。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	W
b14	bSOFTSTOP_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	W
b13	bSOFTSTOP_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	W
b12	bSOFTSTOP_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	W
b11	bSOFTSTOP_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	W
b10	bSOFTSTOP_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	W
b9	bSOFTSTOP_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	W
b8	bSOFTSTOP_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	W
b7	bSOFTSTOP_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	W
b6	bSOFTSTOP_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	W
b5	bSOFTSTOP_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	W
b4	bSOFTSTOP_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	W
b3	bSOFTSTOP_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	W
b2	bSOFTSTOP_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	W
b1	bSOFTSTOP_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	W
b0	bSOFTSTOP_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	W

1.4.34 rPWMTimer_PWM_SOFTUPDOWN_0 — PWM モジュール Up/Down 制御

本レジスタには、ライトマスクの制御があります。上位 16 ビットは、対応する下位 16 ビットのライトイネーブルとして使用されます。

アドレス 4006 A040h	
ビット	b31 b30 b29 b28 b27 b26 b25 b24 b23 b22 b21 b20 b19 b18 b17 b16
	bSOFTUPDOWN_MASK
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ビット	b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0
	bSOFTUPDOWN_7_1 bSOFTUPDOWN_6_1 bSOFTUPDOWN_5_1 bSOFTUPDOWN_4_1 bSOFTUPDOWN_3_1 bSOFTUPDOWN_2_1 bSOFTUPDOWN_1_1 bSOFTUPDOWN_0_1 bSOFTUPDOWN_7_0 bSOFTUPDOWN_6_0 bSOFTUPDOWN_5_0 bSOFTUPDOWN_4_0 bSOFTUPDOWN_3_0 bSOFTUPDOWN_2_0 bSOFTUPDOWN_1_0 bSOFTUPDOWN_0_0
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

表 1.36 rPWMTimer_PWM_SOFTUPDOWN_0 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	bSOFTUPDOWN_MASK	読むと 0 が読み出されます。 本ビットに 1 を書き込むと、同アクセスでそれぞれ下位ビットに対応したライトイネーブルが有効となります。	W
b15	bSOFTUPDOWN_7_1	PWM[k] シングルカウンタ[i] のカウンタ Up/Down 出力 1'b0 : 0 : カウントダウン 1'b1 : 1 : カウントアップ 本ビットは、上位ビットに対応したライトイネーブルがあります。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7) 備考) ソフトウェアでカウンタの方向を設定するときは、Set/Reset モードを使用し、Set/Reset 入力を 0 固定してください。詳細は、「1.5.6.2(1)(b) Up/Down 入力」を参照してください。	W
b14	bSOFTUPDOWN_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	W
b13	bSOFTUPDOWN_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	W
b12	bSOFTUPDOWN_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	W
b11	bSOFTUPDOWN_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	W
b10	bSOFTUPDOWN_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	W
b9	bSOFTUPDOWN_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	W
b8	bSOFTUPDOWN_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	W
b7	bSOFTUPDOWN_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	W
b6	bSOFTUPDOWN_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	W
b5	bSOFTUPDOWN_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	W
b4	bSOFTUPDOWN_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	W
b3	bSOFTUPDOWN_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	W

ビット位置	ビット名	機能	R/W
b2	bSOFTUPDOWN_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	W
b1	bSOFTUPDOWN_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	W
b0	bSOFTUPDOWN_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	W

1.4.35 rPWMTimer_PWM_SOFTRESET — Timebase リセットトリガ

アドレス 4006 A050h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bSOFT RESET _7_1	bSOFT RESET _6_1	bSOFT RESET _5_1	bSOFT RESET _4_1	bSOFT RESET _3_1	bSOFT RESET _2_1	bSOFT RESET _1_1	bSOFT RESET _0_1	bSOFT RESET _7_0	bSOFT RESET _6_0	bSOFT RESET _5_0	bSOFT RESET _4_0	bSOFT RESET _3_0	bSOFT RESET _2_0	bSOFT RESET _1_0	bSOFT RESET _0_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.37 rPWMTimer_PWM_SOFTRESET レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	W
b15	bSOFTRESET_7_1	PWM[k]シングルカウンタ[i]のリセットトリガ 1 を書き込むとカウンタはリセット、0 の書き込みは無効です。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	W
b14	bSOFTRESET_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	W
b13	bSOFTRESET_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	W
b12	bSOFTRESET_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	W
b11	bSOFTRESET_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	W
b10	bSOFTRESET_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	W
b9	bSOFTRESET_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	W
b8	bSOFTRESET_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	W
b7	bSOFTRESET_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	W
b6	bSOFTRESET_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	W
b5	bSOFTRESET_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	W
b4	bSOFTRESET_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	W
b3	bSOFTRESET_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	W
b2	bSOFTRESET_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	W
b1	bSOFTRESET_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	W
b0	bSOFTRESET_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	W

1.4.36 rPWMTimer_PWM_CMP1_FLAG_POLARITY — PWM CMP1 フラグ設定

アドレス 4006 A0D4h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bCMP1_FLAG_EDGE_15	bCMP1_FLAG_EDGE_14	bCMP1_FLAG_EDGE_13	bCMP1_FLAG_EDGE_12	bCMP1_FLAG_EDGE_11	bCMP1_FLAG_EDGE_10	bCMP1_FLAG_EDGE_9	bCMP1_FLAG_EDGE_8	bCMP1_FLAG_EDGE_7	bCMP1_FLAG_EDGE_6	bCMP1_FLAG_EDGE_5	bCMP1_FLAG_EDGE_4	bCMP1_FLAG_EDGE_3	bCMP1_FLAG_EDGE_2	bCMP1_FLAG_EDGE_1	bCMP1_FLAG_EDGE_0
リセット後の値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 1.38 rPWMTimer_PWM_CMP1_FLAG_POLARITY レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bCMP1_FLAG_EDGE_15	PWM[k]シングルカウンタ[i]の CMP1 フラグエッジ選択 1'b1 : CMP1 の立ち上がりエッジでフラグセット 1'b0 : CMP1 の立ち下がりエッジでフラグセット 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bCMP1_FLAG_EDGE_14	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bCMP1_FLAG_EDGE_13	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bCMP1_FLAG_EDGE_12	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bCMP1_FLAG_EDGE_11	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bCMP1_FLAG_EDGE_10	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bCMP1_FLAG_EDGE_9	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bCMP1_FLAG_EDGE_8	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bCMP1_FLAG_EDGE_7	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bCMP1_FLAG_EDGE_6	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bCMP1_FLAG_EDGE_5	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bCMP1_FLAG_EDGE_4	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bCMP1_FLAG_EDGE_3	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bCMP1_FLAG_EDGE_2	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bCMP1_FLAG_EDGE_1	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bCMP1_FLAG_EDGE_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.37 rPWMTimer_PWM_CMP1_FLAG — PWM CMP1 フラグ

本レジスタは、PWMTimer 割り込みを生成用に Event Manager で使用されます。詳細は、「**1.5.7 Event Manager モジュール**」を参照してください。

アドレス		4006 A0DCh														
ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM_CMP1_FLAG_15	bPWM_CMP1_FLAG_14	bPWM_CMP1_FLAG_13	bPWM_CMP1_FLAG_12	bPWM_CMP1_FLAG_11	bPWM_CMP1_FLAG_10	bPWM_CMP1_FLAG_9	bPWM_CMP1_FLAG_8	bPWM_CMP1_FLAG_7	bPWM_CMP1_FLAG_6	bPWM_CMP1_FLAG_5	bPWM_CMP1_FLAG_4	bPWM_CMP1_FLAG_3	bPWM_CMP1_FLAG_2	bPWM_CMP1_FLAG_1	bPWM_CMP1_FLAG_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.39 rPWMTimer_PWM_CMP1_FLAG レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_CMP1_FLAG_15	PWM[k] シングカウンタ[i]からの CMP1 ステータスフラグ CMP1 フラグエッジ設定により、1 にセットされます。 1 を書き込むとフラグクリア、0 の書き込みは無効です。 本ビット：i = 7, k = 1 (PWM1 のシングカウンタ 7)	R/W
b14	bPWM_CMP1_FLAG_14	上記参照、本ビット：i = 6, k = 1 (PWM1 のシングカウンタ 6)	R/W
b13	bPWM_CMP1_FLAG_13	上記参照、本ビット：i = 5, k = 1 (PWM1 のシングカウンタ 5)	R/W
b12	bPWM_CMP1_FLAG_12	上記参照、本ビット：i = 4, k = 1 (PWM1 のシングカウンタ 4)	R/W
b11	bPWM_CMP1_FLAG_11	上記参照、本ビット：i = 3, k = 1 (PWM1 のシングカウンタ 3)	R/W
b10	bPWM_CMP1_FLAG_10	上記参照、本ビット：i = 2, k = 1 (PWM1 のシングカウンタ 2)	R/W
b9	bPWM_CMP1_FLAG_9	上記参照、本ビット：i = 1, k = 1 (PWM1 のシングカウンタ 1)	R/W
b8	bPWM_CMP1_FLAG_8	上記参照、本ビット：i = 0, k = 1 (PWM1 のシングカウンタ 0)	R/W
b7	bPWM_CMP1_FLAG_7	上記参照、本ビット：i = 7, k = 0 (PWM0 のシングカウンタ 7)	R/W
b6	bPWM_CMP1_FLAG_6	上記参照、本ビット：i = 6, k = 0 (PWM0 のシングカウンタ 6)	R/W
b5	bPWM_CMP1_FLAG_5	上記参照、本ビット：i = 5, k = 0 (PWM0 のシングカウンタ 5)	R/W
b4	bPWM_CMP1_FLAG_4	上記参照、本ビット：i = 4, k = 0 (PWM0 のシングカウンタ 4)	R/W
b3	bPWM_CMP1_FLAG_3	上記参照、本ビット：i = 3, k = 0 (PWM0 のシングカウンタ 3)	R/W
b2	bPWM_CMP1_FLAG_2	上記参照、本ビット：i = 2, k = 0 (PWM0 のシングカウンタ 2)	R/W
b1	bPWM_CMP1_FLAG_1	上記参照、本ビット：i = 1, k = 0 (PWM0 のシングカウンタ 1)	R/W
b0	bPWM_CMP1_FLAG_0	上記参照、本ビット：i = 0, k = 0 (PWM0 のシングカウンタ 0)	R/W

1.4.38 rPWMTimer_PWM_MAX_OVERFLOW — PWM オーバーフローフラグ

本レジスタは、PWMTimer 割り込みを生成用に Event Manager で使用されます。詳細は、「1.5.7 Event Manager モジュール」を参照してください。

アドレス 4006 A108h	
ビット	b31 b30 b29 b28 b27 b26 b25 b24 b23 b22 b21 b20 b19 b18 b17 b16
	—
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ビット	b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0
	bPWM_OVERFLOW_FLAG_7_1 bPWM_OVERFLOW_FLAG_6_1 bPWM_OVERFLOW_FLAG_5_1 bPWM_OVERFLOW_FLAG_4_1 bPWM_OVERFLOW_FLAG_3_1 bPWM_OVERFLOW_FLAG_2_1 bPWM_OVERFLOW_FLAG_1_1 bPWM_OVERFLOW_FLAG_0_1 bPWM_OVERFLOW_FLAG_7_0 bPWM_OVERFLOW_FLAG_6_0 bPWM_OVERFLOW_FLAG_5_0 bPWM_OVERFLOW_FLAG_4_0 bPWM_OVERFLOW_FLAG_3_0 bPWM_OVERFLOW_FLAG_2_0 bPWM_OVERFLOW_FLAG_1_0 bPWM_OVERFLOW_FLAG_0_0
リセット後の値	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

表 1.40 rPWMTimer_PWM_MAX_OVERFLOW レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_OVERFLOW_FLAG_7_1	PWM[k] シングルカウンタ [i] からのオーバーフローフラグ オーバーフロー時に、1 にセットされます。 1 を書き込むとフラグクリア、0 の書き込みは無効です。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bPWM_OVERFLOW_FLAG_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bPWM_OVERFLOW_FLAG_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bPWM_OVERFLOW_FLAG_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bPWM_OVERFLOW_FLAG_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bPWM_OVERFLOW_FLAG_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bPWM_OVERFLOW_FLAG_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bPWM_OVERFLOW_FLAG_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bPWM_OVERFLOW_FLAG_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bPWM_OVERFLOW_FLAG_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bPWM_OVERFLOW_FLAG_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bPWM_OVERFLOW_FLAG_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bPWM_OVERFLOW_FLAG_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bPWM_OVERFLOW_FLAG_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bPWM_OVERFLOW_FLAG_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bPWM_OVERFLOW_FLAG_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.39 rPWMTimer_PWM_MIN_UNDERFLOW — PWM アンダーフローフラグ

本レジスタは、PWMTimer 割り込みを生成用に Event Manager で使用されます詳細は、「**1.5.7 Event Manager モジュール**」を参照してください。

アドレス		4006 A10Ch														
ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bPWM UNDER FLOW FLAG_7_1	bPWM UNDER FLOW FLAG_6_1	bPWM UNDER FLOW FLAG_5_1	bPWM UNDER FLOW FLAG_4_1	bPWM UNDER FLOW FLAG_3_1	bPWM UNDER FLOW FLAG_2_1	bPWM UNDER FLOW FLAG_1_1	bPWM UNDER FLOW FLAG_0_1	bPWM UNDER FLOW FLAG_7_0	bPWM UNDER FLOW FLAG_6_0	bPWM UNDER FLOW FLAG_5_0	bPWM UNDER FLOW FLAG_4_0	bPWM UNDER FLOW FLAG_3_0	bPWM UNDER FLOW FLAG_2_0	bPWM UNDER FLOW FLAG_1_0	bPWM UNDER FLOW FLAG_0_0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.41 rPWMTimer_PWM_MIN_UNDERFLOW レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b16	予約ビット	初期値を保持してください。	R/W
b15	bPWM_UNDERFLOW_FLAG_7_1	PWM[k] シングルカウンタ [i] からのアンダーフローフラグ アンダーフロー時に、1 にセットされます。 1 を書き込むとフラグクリア、0 の書き込みは無効です。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bPWM_UNDERFLOW_FLAG_6_1	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bPWM_UNDERFLOW_FLAG_5_1	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bPWM_UNDERFLOW_FLAG_4_1	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bPWM_UNDERFLOW_FLAG_3_1	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bPWM_UNDERFLOW_FLAG_2_1	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bPWM_UNDERFLOW_FLAG_1_1	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bPWM_UNDERFLOW_FLAG_0_1	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bPWM_UNDERFLOW_FLAG_7_0	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bPWM_UNDERFLOW_FLAG_6_0	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bPWM_UNDERFLOW_FLAG_5_0	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bPWM_UNDERFLOW_FLAG_4_0	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bPWM_UNDERFLOW_FLAG_3_0	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bPWM_UNDERFLOW_FLAG_2_0	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bPWM_UNDERFLOW_FLAG_1_0	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W
b0	bPWM_UNDERFLOW_FLAG_0_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.40 rPWMTimer_HWSTORE_OR[r]_STORE_FLAG — PWM OR[r]トリガフラグ (r = 1, 3)

本レジスタは、PWMTimer 割り込みを生成用に Event Manager で使用されます。詳細は、「1.5.7 Event Manager モジュール」を参照してください。

アドレス 4006 A1D0h + 4h × r

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—															
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bOR[r] STORE FLAG 15	bOR[r] STORE FLAG 14	bOR[r] STORE FLAG 13	bOR[r] STORE FLAG 12	bOR[r] STORE FLAG 11	bOR[r] STORE FLAG 10	bOR[r] STORE FLAG 9	bOR[r] STORE FLAG 8	bOR[r] STORE FLAG 7	bOR[r] STORE FLAG 6	bOR[r] STORE FLAG 5	bOR[r] STORE FLAG 4	bOR[r] STORE FLAG 3	bOR[r] STORE FLAG 2	bOR[r] STORE FLAG 1	bOR[r] STORE FLAG 0
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.42 rPWMTimer_HWSTORE_OR[r]_STORE_FLAG レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b16	予約ビット	初期値を保持してください。	R/W
b15	bOR[r]_STORE_FLAG_15	PWM[k]シングルカウンタ[i]の OR[r]トリガフラグ OR[r]_TRIG.[i]信号の立ち上がりエッジで 1 にセットされます。 1 を書き込むとフラグクリア、0 の書き込みは無効です。 本ビット : i = 7, k = 1 (PWM1 のシングルカウンタ 7)	R/W
b14	bOR[r]_STORE_FLAG_14	上記参照、本ビット : i = 6, k = 1 (PWM1 のシングルカウンタ 6)	R/W
b13	bOR[r]_STORE_FLAG_13	上記参照、本ビット : i = 5, k = 1 (PWM1 のシングルカウンタ 5)	R/W
b12	bOR[r]_STORE_FLAG_12	上記参照、本ビット : i = 4, k = 1 (PWM1 のシングルカウンタ 4)	R/W
b11	bOR[r]_STORE_FLAG_11	上記参照、本ビット : i = 3, k = 1 (PWM1 のシングルカウンタ 3)	R/W
b10	bOR[r]_STORE_FLAG_10	上記参照、本ビット : i = 2, k = 1 (PWM1 のシングルカウンタ 2)	R/W
b9	bOR[r]_STORE_FLAG_9	上記参照、本ビット : i = 1, k = 1 (PWM1 のシングルカウンタ 1)	R/W
b8	bOR[r]_STORE_FLAG_8	上記参照、本ビット : i = 0, k = 1 (PWM1 のシングルカウンタ 0)	R/W
b7	bOR[r]_STORE_FLAG_7	上記参照、本ビット : i = 7, k = 0 (PWM0 のシングルカウンタ 7)	R/W
b6	bOR[r]_STORE_FLAG_6	上記参照、本ビット : i = 6, k = 0 (PWM0 のシングルカウンタ 6)	R/W
b5	bOR[r]_STORE_FLAG_5	上記参照、本ビット : i = 5, k = 0 (PWM0 のシングルカウンタ 5)	R/W
b4	bOR[r]_STORE_FLAG_4	上記参照、本ビット : i = 4, k = 0 (PWM0 のシングルカウンタ 4)	R/W
b3	bOR[r]_STORE_FLAG_3	上記参照、本ビット : i = 3, k = 0 (PWM0 のシングルカウンタ 3)	R/W
b2	bOR[r]_STORE_FLAG_2	上記参照、本ビット : i = 2, k = 0 (PWM0 のシングルカウンタ 2)	R/W
b1	bOR[r]_STORE_FLAG_1	上記参照、本ビット : i = 1, k = 0 (PWM0 のシングルカウンタ 1)	R/W

ビット位置	ビット名	機能	R/W
b0	bOR[r]_STORE_FLAG_0	上記参照、本ビット : i = 0, k = 0 (PWM0 のシングルカウンタ 0)	R/W

1.4.41 rPWMTimer_ROUTING_IN_[s] — PWMTimer の IN 入力選択 (s = 0, 2)

PWMTimer 入力 IN 設定 : 0~2, 10~12

アドレス 4006 B100h+4h×s

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—		—													bROUTING_INP UT_IN_[2+s*5]
リセット後の値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bROUTING_INPUT_IN_[2+s*5]				bROUTING_INPUT_IN_[1+s*5]						bROUTING_INPUT_IN_[0+s*5]					
リセット後の値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

表 1.43 rPWMTimer_ROUTING_IN_[s] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b30	予約ビット	読むと 0 が読み出されます。	R
b29~b18	予約ビット	初期値を保持してください。	R/W
b17~b12	bROUTING_INPUT_I N_[2+s*5]	PWM[s/2]のクロック入力選択 (ClockIn) PWMTimer Core 入力 IN[e+s*5]設定 6'd0 : PWMTimer Core IN = PWM_IN[0] 6'd1 : PWMTimer Core IN = PWM_IN[1] 6'dX : PWMTimer Core IN = PWM_IN[X] (X = 0~39) 6'd39 : PWMTimer Core IN = PWM_IN[39] 6'dY : 予約 (Y = 40~63) 本ビット : e = 2 (PWMTimer Core 入力 IN 2, 12) 備考) リセット後の値は、PWMTimer Core 入力 IN と同じ[e+s*5]になります。	R/W
b11~b6	bROUTING_INPUT_I N_[1+s*5]	PWM[s/2]のキャプチャトリガ 0 の入力選択 (Trig0) 上記参照、本ビット : e = 1 (PWMTimer Core 入力 IN 1, 11)	R/W
b5~b0	bROUTING_INPUT_I N_[0+s*5]	PWM[s/2]のキャプチャトリガ 1 の入力選択 (Trig1) 上記参照、本ビット : e = 0 (PWMTimer Core 入力 IN 0, 10)	R/W

1.4.42 rPWMTimer_ROUTING_OUT_[n] — PWMTimer の OUT 出力選択 (n = 0~3)

PWMTimer 出力 PWM_OUT 設定 : 0~19

アドレス 4006 B200h+4h×n

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—			bROUTING_OUTPUT_OUT_[4+n*5]					—	bROUTING_OUTPUT_OUT_[3+n*5]					—	bROUTING_OUTPUT_OUT_[2+n*5]
リセット後の値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bROUTING_OUTPUT_OUT_[2+n*5]			—	bROUTING_OUTPUT_OUT_[1+n*5]					—	bROUTING_OUTPUT_OUT_[0+n*5]					
リセット後の値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

表 1.44 rPWMTimer_ROUTING_OUT_[n] レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b29	予約ビット	読むと 0 が読み出されます。	R
b28~b24	bROUTING_OUTPUT_OUT_[4+n*5]	PWM_OUT[e+n*5]出力設定 5'd0 : PWM_OUT = PWMTimer Core OUT[0] 5'd1 : PWM_OUT = PWMTimer Core OUT[1] 5'dX : PWM_OUT = PWMTimer Core OUT[X] (X = 0~15) 5'd15 : PWM_OUT = PWMTimer Core OUT[15] 5'dY : 予約 (Y = 16~31) 本ビット : e = 4 (PWM_OUT 4, 9, 14, 19) 備考) リセット後の値は、PWM_OUT と同じ[e+n*5]になります。	R/W
b23	予約ビット	読むと 0 が読み出されます。	R
b22~b18	bROUTING_OUTPUT_OUT_[3+n*5]	上記参照、本ビット : e = 3 (PWM_OUT 3, 8, 13, 18)	R/W
b17	予約ビット	読むと 0 が読み出されます。	R
b16~b12	bROUTING_OUTPUT_OUT_[2+n*5]	上記参照、本ビット : e = 2 (PWM_OUT 2, 7, 12, 17)	R/W
b11	予約ビット	読むと 0 が読み出されます。	R
b10~b6	bROUTING_OUTPUT_OUT_[1+n*5]	上記参照、本ビット : e = 1 (PWM_OUT 1, 6, 11, 16)	R/W
b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	bROUTING_OUTPUT_OUT_[0+n*5]	上記参照、本ビット : e = 0 (PWM_OUT 0, 5, 10, 15)	R/W

1.4.43 rPWMTimer_ROUTING_OUTCTRL_0 — OutputCtrl0 モジュールの入力選択

アドレス 4006 B800h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_0					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.45 rPWMTimer_ROUTING_OUTCTRL_0 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_0	OutputCtrl の入力設定 5'h00 : OutputCtrl0.Set = 1'b0 5'h03 : OutputCtrl0.Set = CMP1.0.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.44 rPWMTimer_ROUTING_OUTCTRL_1 — OutputCtrl1 モジュールの入力選択

アドレス 4006 B804h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_1					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.46 rPWMTimer_ROUTING_OUTCTRL_1 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_1	OutputCtrl の入力設定 5'h00 : OutputCtrl1.Set = 1'b0 5'h05 : OutputCtrl1.Set = CMP1.1.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.45 rPWMTimer_ROUTING_OUTCTRL_2 — OutputCtrl2 モジュールの入力選択

アドレス 4006 B808h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_2				—			—					
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.47 rPWMTimer_ROUTING_OUTCTRL_2 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_2	OutputCtrl の入力設定 5'h00 : OutputCtrl2.Set = 1'b0 5'h07 : OutputCtrl2.Set = CMP1.2.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.46 rPWMTimer_ROUTING_OUTCTRL_3 — OutputCtrl3 モジュールの入力選択

アドレス 4006 B80Ch

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_3					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.48 rPWMTimer_ROUTING_OUTCTRL_3 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_3	OutputCtrl の入力設定 5'h00 : OutputCtrl3.Set = 1'b0 5'h09 : OutputCtrl3.Set = CMP1.3.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.47 rPWMTimer_ROUTING_OUTCTRL_4 — OutputCtrl4 モジュールの入力選択

アドレス 4006 B810h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_4					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.49 rPWMTimer_ROUTING_OUTCTRL_4 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_4	OutputCtrl の入力設定 5'h00 : OutputCtrl4.Set = 1'b0 5'h0B : OutputCtrl4.Set = CMP1.4.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.48 rPWMTimer_ROUTING_OUTCTRL_5 — OutputCtrl5 モジュールの入力選択

アドレス 4006 B814h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_5				—			—					
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.50 rPWMTimer_ROUTING_OUTCTRL_5 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	予約ビット	初期値を保持してください。	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b8	bROUTING_OUTCTRL_SET_5	OutputCtrl の入力設定 5'h00 : OutputCtrl5.Set = 1'b0 5'h0D : OutputCtrl5.Set = CMP1.5.0 それ以外 : 予約	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.49 rPWMTimer_ROUTING_OUTCTRL_6 — OutputCtrl6 モジュールの入力選択

アドレス 4006 B818h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_6					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.51 rPWMTimer_ROUTING_OUTCTRL_6 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_6	OutputCtrl の入力設定 5'h00 : OutputCtrl6.Set = 1'b0 5'h0F : OutputCtrl6.Set = CMP1.6.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.50 rPWMTimer_ROUTING_OUTCTRL_7 — OutputCtrl7 モジュールの入力選択

アドレス 4006 B81Ch

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_7					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.52 rPWMTimer_ROUTING_OUTCTRL_7 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_7	OutputCtrl の入力設定 5'h00 : OutputCtrl7.Set = 1'b0 5'h11 : OutputCtrl7.Set = CMP1.7.0 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.51 rPWMTimer_ROUTING_OUTCTRL_8 — OutputCtrl8 モジュールの入力選択

アドレス 4006 B820h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_8					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.53 rPWMTimer_ROUTING_OUTCTRL_8 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_8	OutputCtrl の入力設定 5'h00 : OutputCtrl8.Set = 1'b0 5'h03 : OutputCtrl8.Set = CMP1.0.1 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.52 rPWMTimer_ROUTING_OUTCTRL_9 — OutputCtrl9 モジュールの入力選択

アドレス 4006 B824h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_9					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.54 rPWMTimer_ROUTING_OUTCTRL_9 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	予約ビット	初期値を保持してください。	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b8	bROUTING_OUTCTRL_SET_9	OutputCtrl の入力設定 5'h00 : OutputCtrl9.Set = 1'b0 5'h05 : OutputCtrl9.Set = CMP1.1.1 それ以外 : 予約	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.53 rPWMTimer_ROUTING_OUTCTRL_10 —
OutputCtrl10 モジュールの入力選択

アドレス 4006 B828h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_10					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.55 rPWMTimer_ROUTING_OUTCTRL_10 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	予約ビット	初期値を保持してください。	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b8	bROUTING_OUTCTRL_SET_10	OutputCtrl の入力設定 5'h00 : OutputCtrl10.Set = 1'b0 5'h07 : OutputCtrl10.Set = CMP1.2.1 それ以外 : 予約	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.54 rPWMTimer_ROUTING_OUTCTRL_11 —
OutputCtrl11 モジュールの入力選択

アドレス 4006 B82Ch

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_11					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.56 rPWMTimer_ROUTING_OUTCTRL_11 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	予約ビット	初期値を保持してください。	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b8	bROUTING_OUTCTRL_SET_11	OutputCtrl の入力設定 5'h00 : OutputCtrl11.Set = 1'b0 5'h09 : OutputCtrl11.Set = CMP1.3.1 それ以外 : 予約	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.55 rPWMTimer_ROUTING_OUTCTRL_12 —
OutputCtrl12 モジュールの入力選択

アドレス 4006 B830h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_12					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.57 rPWMTimer_ROUTING_OUTCTRL_12 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_12	OutputCtrl の入力設定 5'h00 : OutputCtrl12.Set = 1'b0 5'h0B : OutputCtrl12.Set = CMP1.4.1 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.56 rPWMTimer_ROUTING_OUTCTRL_13 —
OutputCtrl13 モジュールの入力選択

アドレス 4006 B834h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_13					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.58 rPWMTimer_ROUTING_OUTCTRL_13 レジスタの内容

ビット位置	ビット名	機能	R/W
b31~b27	予約ビット	読むと 0 が読み出されます。	R
b26~b24	予約ビット	初期値を保持してください。	R/W
b23~b21	予約ビット	読むと 0 が読み出されます。	R
b20~b16	予約ビット	初期値を保持してください。	R/W
b15~b13	予約ビット	読むと 0 が読み出されます。	R
b12~b8	bROUTING_OUTCTRL_SET_13	OutputCtrl の入力設定 5'h00 : OutputCtrl13.Set = 1'b0 5'h0D : OutputCtrl13.Set = CMP1.5.1 それ以外 : 予約	R/W
b7~b5	予約ビット	読むと 0 が読み出されます。	R
b4~b0	予約ビット	初期値を保持してください。	R/W

1.4.57 rPWMTimer_ROUTING_OUTCTRL_14 —
OutputCtrl14 モジュールの入力選択

アドレス 4006 B838h

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_14					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.59 rPWMTimer_ROUTING_OUTCTRL_14 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_14	OutputCtrl の入力設定 5'h00 : OutputCtrl14.Set = 1'b0 5'h0F : OutputCtrl14.Set = CMP1.6.1 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.4.58 rPWMTimer_ROUTING_OUTCTRL_15 —
OutputCtrl15 モジュールの入力選択

アドレス 4006 B83Ch

ビット	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—					—			—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ビット	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—			bROUTING_OUTCTRL_SET_15					—			—				
リセット後の値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 1.60 rPWMTimer_ROUTING_OUTCTRL_15 レジスタの内容

ビット位置	ビット名	機能	R/W
b31～b27	予約ビット	読むと 0 が読み出されます。	R
b26～b24	予約ビット	初期値を保持してください。	R/W
b23～b21	予約ビット	読むと 0 が読み出されます。	R
b20～b16	予約ビット	初期値を保持してください。	R/W
b15～b13	予約ビット	読むと 0 が読み出されます。	R
b12～b8	bROUTING_OUTCTRL_SET_15	OutputCtrl の入力設定 5'h00 : OutputCtrl15.Set = 1'b0 5'h11 : OutputCtrl15.Set = CMP1.7.1 それ以外 : 予約	R/W
b7～b5	予約ビット	読むと 0 が読み出されます。	R
b4～b0	予約ビット	初期値を保持してください。	R/W

1.5 動作説明

1.5.1 PWMTimer Top 内部接続および選択

40 本の外部入力同期され、PWMTimer Core の入力信号に任意の割り当てが可能です。また、同じ外部入力を使用することも可能です。

PWMTimer Core の出力信号は、外部出力 20 本に任意の割り当てが可能です。

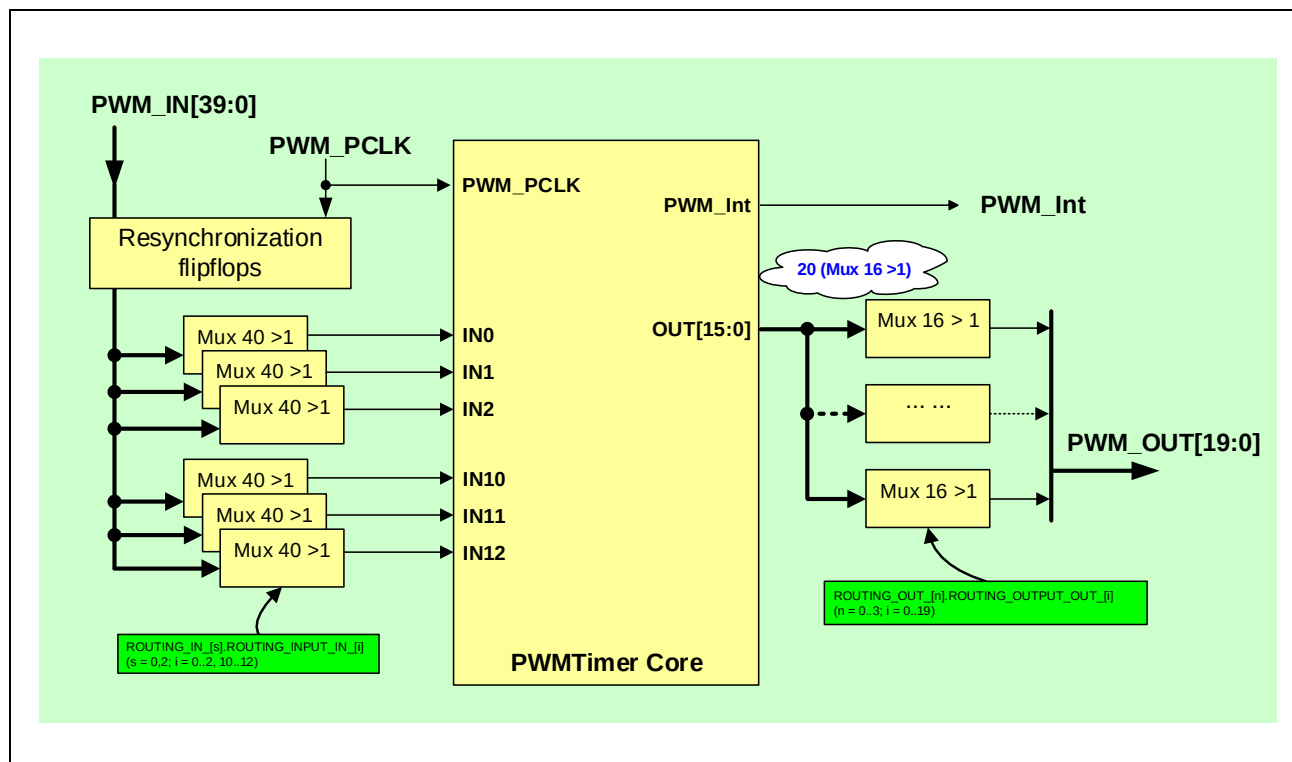


図 1.2 PWMTimer Top の概要

詳細については、以下のレジスタを参照してください。

rPWMTimer_ROUTING_IN_[s] (s = 0, 2), rPWMTimer_ROUTING_OUT_[n] (n = 0~3)

備 考

本章の図においては、レジスタ名およびビット名を一部省略しています。

1.5.2 PWMTimer Core

PWMTimer Core のモジュール接続関係を示します。

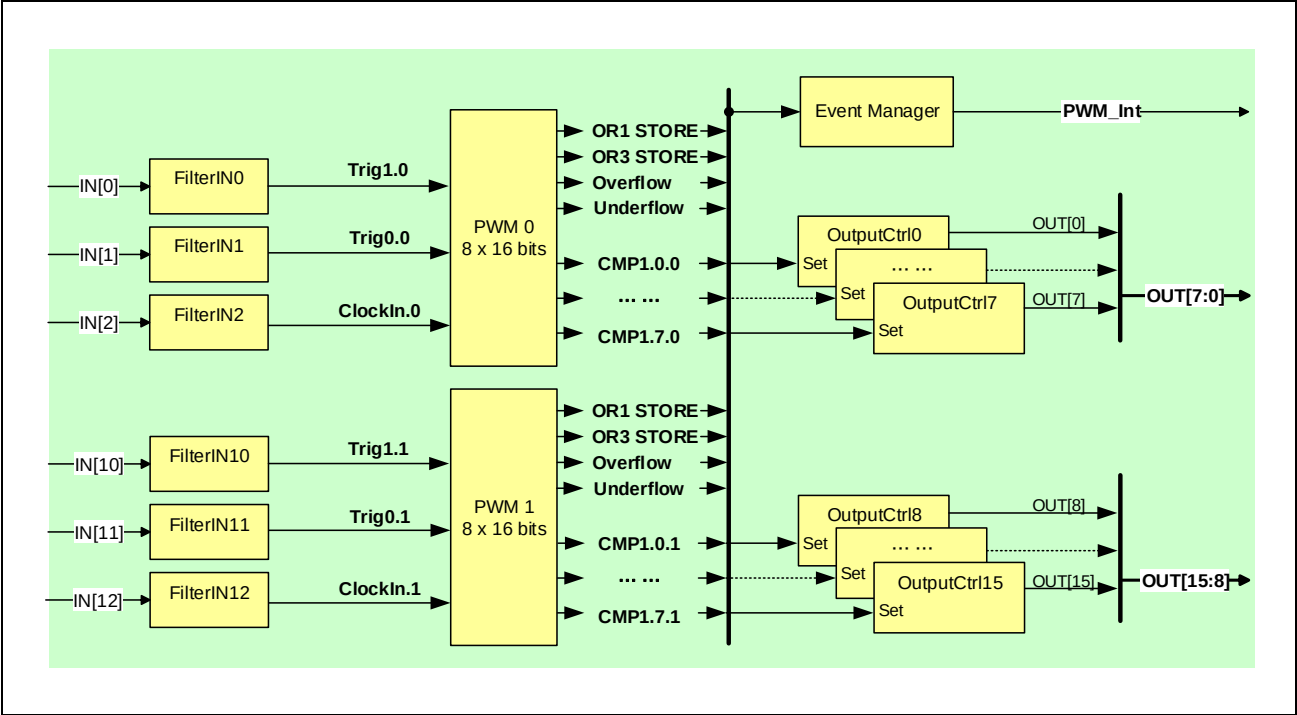


図 1.3 PWMTimer Core の概要

1.5.3 PWMTimer Core 内部接続

PWMTimer Core 信号と PWM モジュールおよび OutputCtrl モジュールの内部接続を示します。
「1.5.2 PWMTimer Core」を参照してください。

1.5.3.1 PWM モジュール接続

入力信号および機能については、「1.5.6 PWM モジュール」を参照してください。各 PWM モジュールは、クロック入力と 2 本のトリガ入力を使用可能です。

表 1.61 PWM 入力接続

FilterIN 出力	PWM0 入力先
FilterIN0	Trig1.0
FilterIN1	Trig0.0
FilterIN2	ClockIn.0
FilterIN 出力	PWM1 入力先
FilterIN10	Trig1.1
FilterIN11	Trig0.1
FilterIN12	ClockIn.1

1.5.3.2 OutputCtrl モジュール接続

入力信号および機能については、「1.5.5 OutputCtrl モジュール」を参照してください。

表 1.62 OutputCtrl 入力接続

PWM 0 出力	OutputCtrl 入力先
CMP1.0.0 (シングルカウンタ 0)	OutputCtrl0.Set
CMP1.1.0 (シングルカウンタ 1)	OutputCtrl1.Set
CMP1.2.0 (シングルカウンタ 2)	OutputCtrl2.Set
CMP1.3.0 (シングルカウンタ 3)	OutputCtrl3.Set
CMP1.4.0 (シングルカウンタ 4)	OutputCtrl4.Set
CMP1.5.0 (シングルカウンタ 5)	OutputCtrl5.Set
CMP1.6.0 (シングルカウンタ 6)	OutputCtrl6.Set
CMP1.7.0 (シングルカウンタ 7)	OutputCtrl7.Set
PWM 1 出力	OutputCtrl 入力先
CMP1.0.1 (シングルカウンタ 0)	OutputCtrl8.Set
CMP1.1.1 (シングルカウンタ 1)	OutputCtrl9.Set
CMP1.2.1 (シングルカウンタ 2)	OutputCtrl10.Set
CMP1.3.1 (シングルカウンタ 3)	OutputCtrl11.Set
CMP1.4.1 (シングルカウンタ 4)	OutputCtrl12.Set
CMP1.5.1 (シングルカウンタ 5)	OutputCtrl13.Set
CMP1.6.1 (シングルカウンタ 6)	OutputCtrl14.Set
CMP1.7.1 (シングルカウンタ 7)	OutputCtrl15.Set

詳細については、以下のレジスタを参照してください。

rPWMTimer_ROUTING_OUTCTRL_[n] (n = 0～15)

1.5.4 FilterIN モジュール

PWMTimer のフィルタ入力について説明します。

PWMTimer 入力の IN0～IN2, IN10～IN12 は、FilterIN モジュールと 1 対 1 に接続されています。

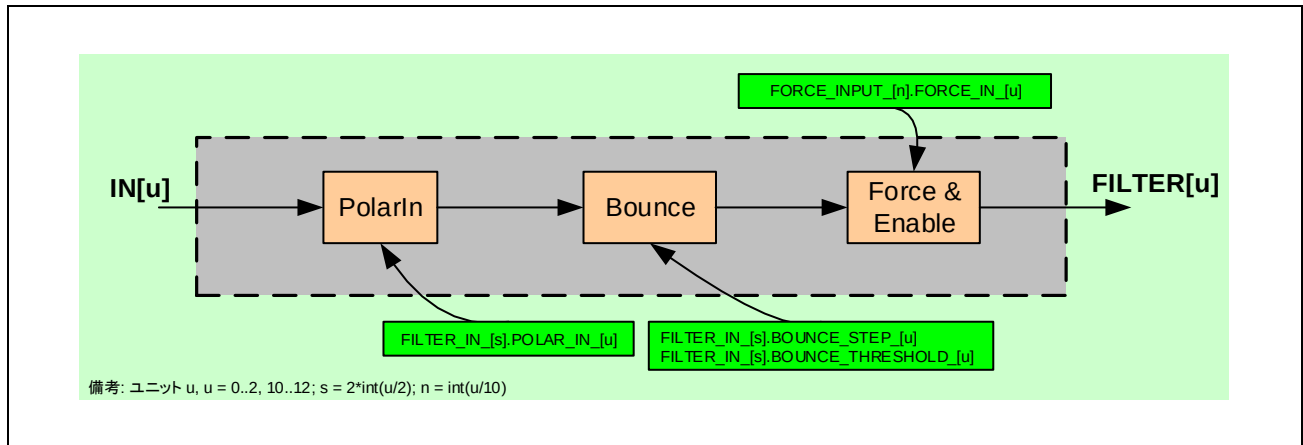


図 1.4 FilterIN モジュール

FilterIN モジュールは、後述する 3 つの機能があります。

詳細については、以下のレジスタを参照してください。

rPWMTimer_FILTER_IN_[s] (s = 0, 2, 10, 12), rPWMTimer_FORCE_INPUT_[n] (n = 0～1)

1.5.4.1 PolarIn

外部からの入力信号の極性を切り替えます。

詳細については、以下のレジスタを参照してください。

rPWMTimer_FILTER_IN[s] (s = 0, 2, 10, 12)

1.5.4.2 Bounce

2 種類のクロックを使用することにより、高速入力 は 80ns 刻みで 0~81.84us、低速入力 は 20.48us 刻みで 0~20.95104ms までのバウンスフィルタ値を設定することが可能です。

Bounce の入力変化で、フィルタカウンタはリセットされます。

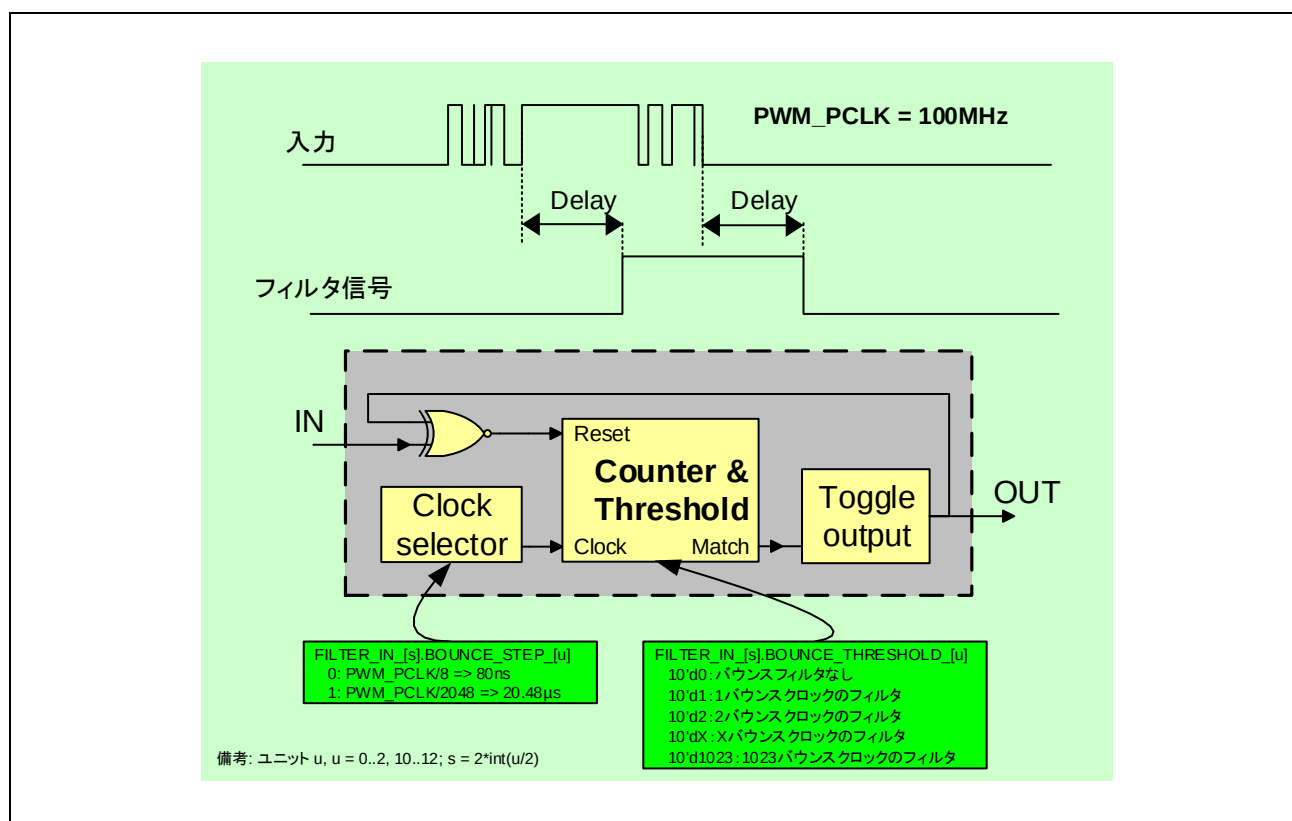


図 1.5 Bounce

詳細については、以下のレジスタを参照してください。

rPWMTimer_FILTER_IN[s] (s = 0, 2, 10, 12)

1.5.4.3 Force & Enable

入力信号を 1 または 0 に固定出力することが可能です。

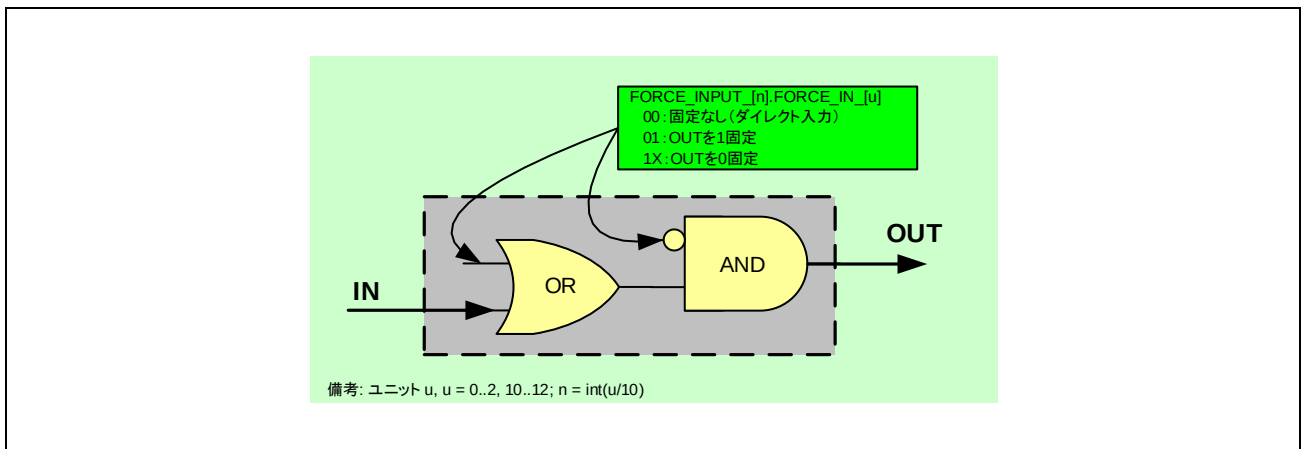


図 1.6 Force & Enable

詳細については、以下のレジスタを参照してください。

rPWMTimer_FORCE_INPUT_[n] (n = 0~1)

備 考

Force & Enable は、OutputCtrl モジュールでも使用しています。

1.5.5 OutputCtrl モジュール

本モジュールは、PWMTimer 出力を生成します。

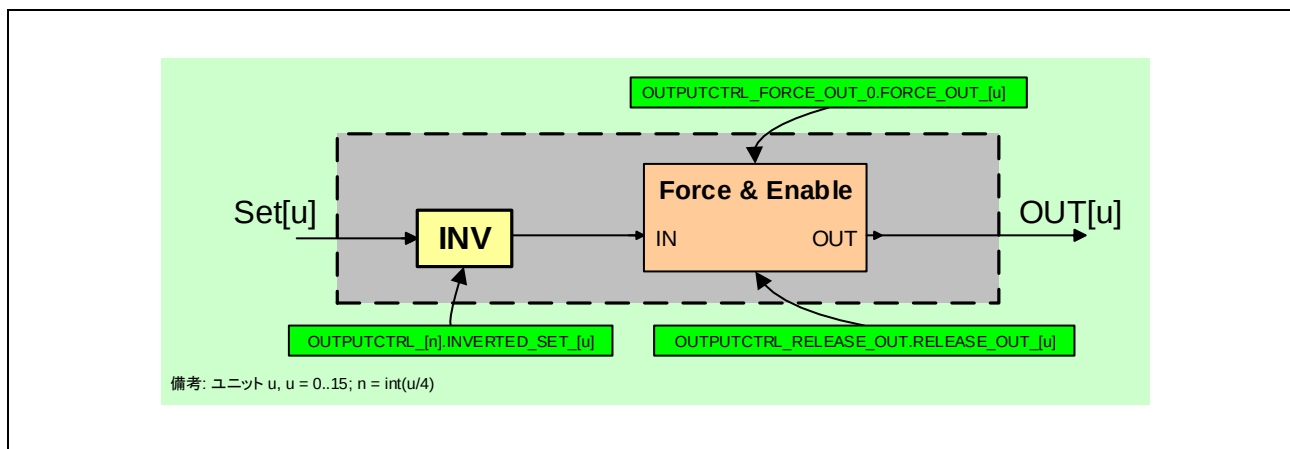


図 1.7 OutputCtrl モジュール

固定出力の解除状態（リセット後）では、“Force & Enable”の IN 値が直接 OUT に出力されます。
出力信号は PWM_PCLK クロックで同期され、反転もしくは、1 または 0 に固定することが可能です。

詳細については、以下のレジスタを参照してください。

rPWMTimer_OUTPUTCTRL_[n] (n = 0~3), rPWMTimer_OUTPUTCTRL_FORCE_OUT_0,
rPWMTimer_OUTPUTCTRL_RELEASE_OUT

1.5.6 PWM モジュール

PWM モジュールは、入力信号を共有した 32 ビットカスケードが可能な 16 ビットカウンタ群です。

- 各 PWM は、8 個のシングルカウンタおよび 2 個のクロックプリスケーラを搭載
- 各シングルカウンタは、次段のカウンタとカスケード接続することで 32 ビットカウンタを生成可能
- 各シングルカウンタは、「1.5.6.2(3) Outputs Generator」で詳述する機能で出力信号を制御

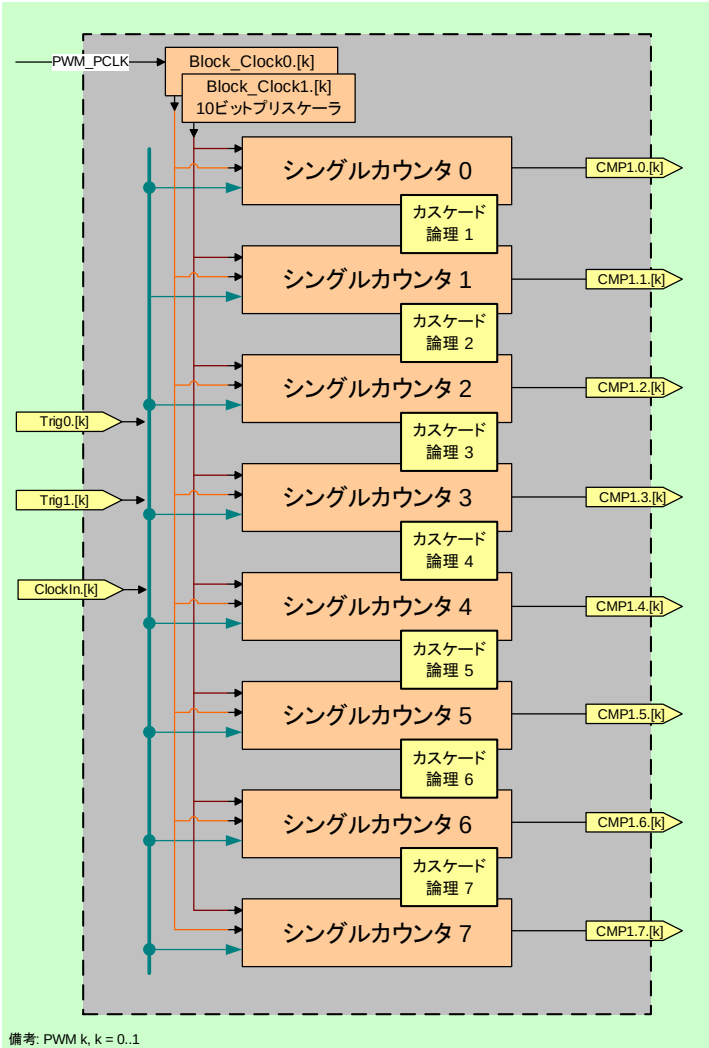


図 1.8 PWM モジュール

備 考

基本的に本項では、以下の添え字を使用します。

- k : PWM モジュール (k = 0~1)
- i : シングルカウンタモジュール (i = 0~7)

1.5.6.1 クロックプリスケアラ

各 PWM は、クロックプリスケアラ (Block_Clock0.[k], Block_Clock1.[k]) を 2 個搭載しています。本プリスケアラは、PWM_PCLK クロックを 2~2048 分周したクロックを生成します。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_BLOCK_CLOCK_[k] (k = 0~1)

1.5.6.2 シングルカウンタモジュール

カスケードモード時、MAX.[i], MIN.[i], CMP1.[i]信号はカスケード論理で生成されます。CarryIn.[i]信号は、カスケード時のみ使用されます。

Timebase カウンタの各動作は、外部入力、内部信号およびソフトウェア要求により制御されます。

- スタート、ストップ、およびリセット：ソフトウェア要求
- クロック：外部入力 (ClockIn) または PWM_PCLK プリスケアラ
- カウント方向：Max/Min 検出およびソフトウェア要求
- キャプチャトリガ：Trig0, Trig1 または内部トリガ Trig2 (Max/Min 検出)
- バッファ転送トリガ：Trig0, Trig1 または内部トリガ Trig2 (Max/Min 検出)

キャプチャおよびバッファ転送の制御は、「1.5.6.2(2)(c) Hardware Store トリガ機能」に詳述されています。

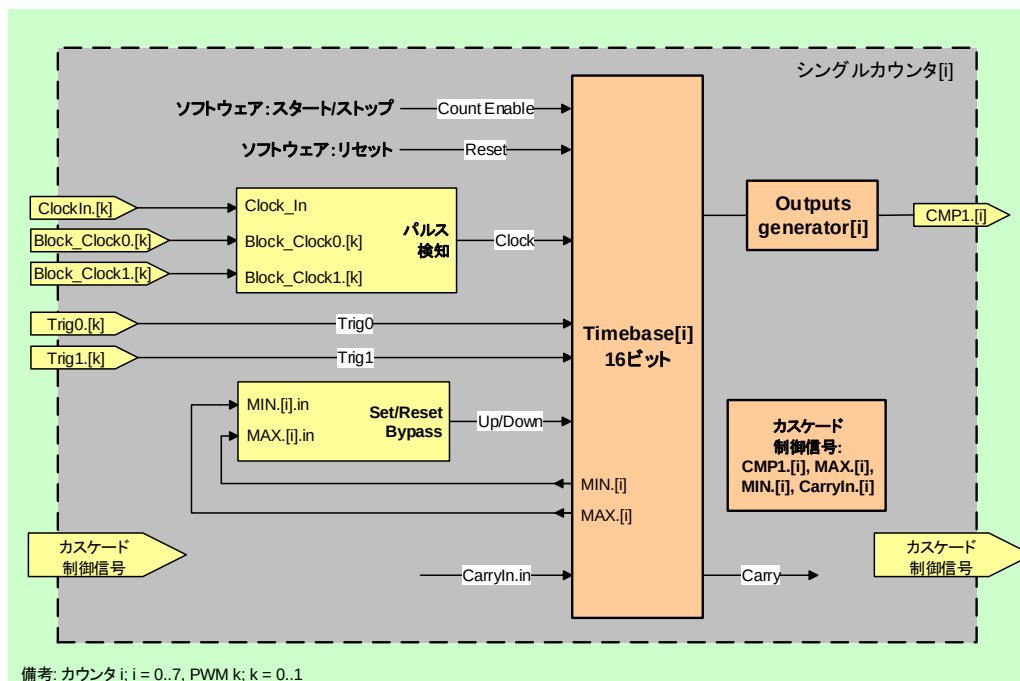


図 1.9 シングルカウンタモジュール

(1) シングルカウンタ入力選択

シングルカウンタに入力可能な外部信号を示します。

表 1.63 シングルカウンタの外部信号入力

シングルカウンタの信号	外部信号 (PWM k, k = 0~1)
Clock	ClockIn.[k]
Trig0	Trig0.[k]
Trig1	Trig1.[k]

(a) クロック入力

カウント動作は、以下の選択が可能です。

- PWM_PCLK 周期
- クロックプリスケアラ (エッジ検出を使用)
- ClockIn.[k]入力 (エッジ検出を使用)

エッジ検出なしの ClockIn.[k]入力を使用することで、パルス幅測定が可能です。

PWM_PCLK 周期でカウントするには、rPWMTimer_PWM_CLOCK_[q]_[k].bMUXEDCLOCK_[i] = 3'b000 を設定する必要があります。

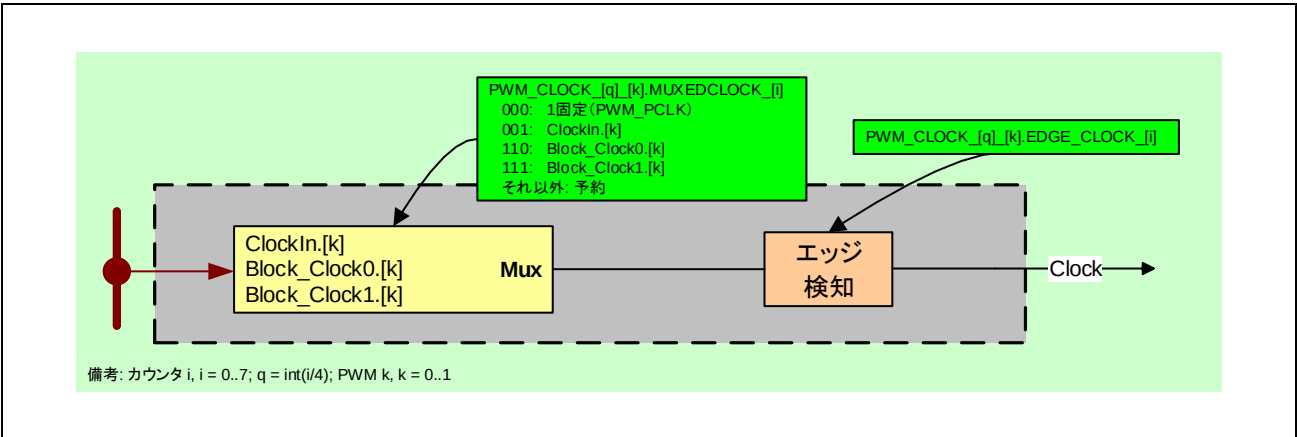


図 1.10 クロック入力

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_CLOCK_[q]_[k] (q = 0~1, k = 0~1)

(b) Up/Down 入力

Set/Reset または Bypass モードを使用し、Timebase カウンタの方向（Up/Down）を制御します。

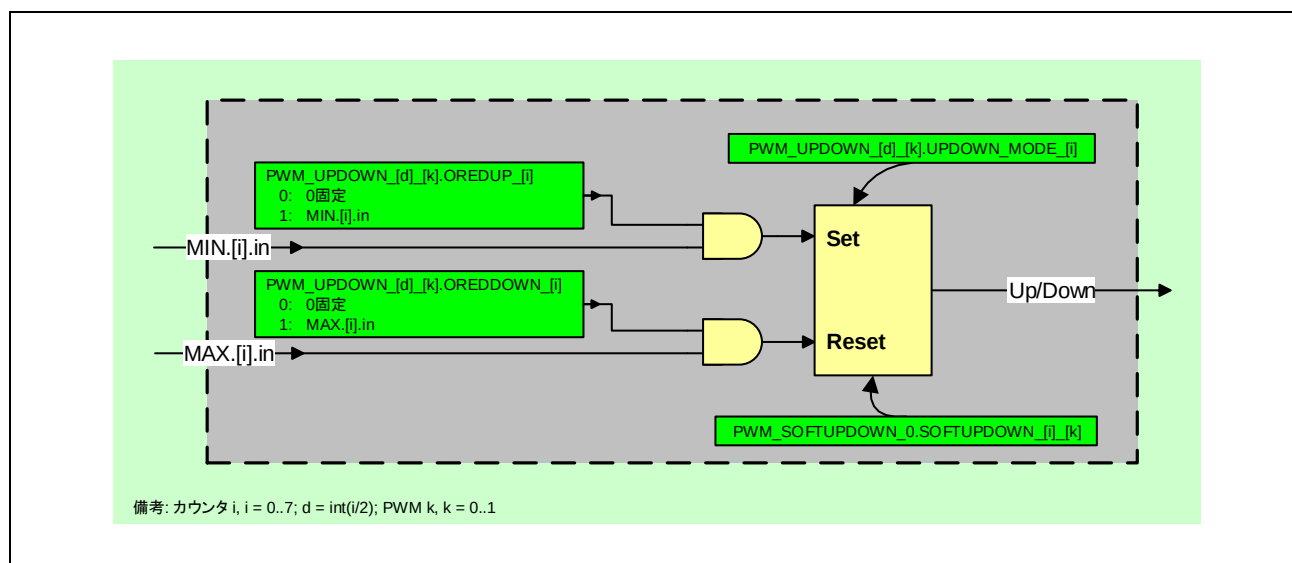


図 1.11 Up/Down 入力

Up/Down 信号の初期値は、Up となります。

Max/Min 検出入力を使用する場合は、Set/Reset モードに設定してください。

Set/Reset モードで同時に入力が発生した場合、Up/Down 信号は Up になります。

備 考

ソフトウェアでカウンタの方向を切り替える場合、Set/Reset モードに設定し、それぞれの入力を 0 固定してください。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_UPDOWN_[d]_[k] ($d = 0..3$, $k = 0..1$), rPWMTimer_PWM_SOFTUPDOWN_0

(c) それ以外の入力機能**• カウントイネーブル機能**

Timebase クロック（カウントクロック）と同期して、カウントイネーブルを有効または無効にします。

- スタートトリガ（カウンタイネーブル有効）：ソフトウェア要求
- ストップトリガ（カウンタイネーブル無効）：ソフトウェア要求

カウントイネーブルが有効な場合、次の Timebase クロックでトリガが発生します。イネーブル制御で同時に両トリガが発生した場合、カウンタはストップします。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_SOFTSTART, rPWMTimer_PWM_SOFTSTOP

• リセット機能

Timebase クロック（カウントクロック）と同期して、カウンタをリセットします。

- リセットトリガ：ソフトウェア要求

カウントイネーブルが有効な場合、次の Timebase クロックでトリガが発生します。カウントイネーブルが無効な場合は、ただちにリセットされます。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_SOFTRESET

• Trig0 および Trig1 入力

Trig0 および Trig1 入力は、キャプチャトリガ／バッファ転送トリガとして、「**1.5.6.2(2)(c) Hardware Store トリガ機能**」で使用されます。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_TRIG0_[k] (k = 0～1), rPWMTimer_PWM_TRIG1_[k] (k = 0～1)

(2) Timebase モジュール

Timebase モジュールは、カウンタの主要部になります。キャプチャもしくはコンペア機能として使用可能です。また、カウント、コンパレータ、キャプチャ、リセット、およびバッファ転送を制御し、CMP1, MAX, MIN Underflow, Overflow, Carry 信号を生成します。

Timebase 構成および機能：

- カウンタ値：現カウンタ値
- 16 ビット演算：カウンタ値を 1 ずつ増減
- オプションレジスタ：コンペア、キャプチャ、最大値、バッファ転送用として 4 個のレジスタ
- 16 ビットコンパレータ部：カウンタ値とオプションレジスタ 2 とのコンペアマッチ
- レンジモード：最大値としてオプションレジスタ 0 を使用したカウントモード制御
- Hardware Store トリガ機能：バッファ転送およびキャプチャ制御
- フラグ生成：オーバーフローおよびアンダーフローのフラグ制御

たとえば、Max 信号を使用して、Up/Down 方向を切り替えることが可能です。最大値に到達するとすぐに Down 方向に切り替わり、このときオーバーフローフラグは発生しません。

カスケードモード用に、隣接する COUNT_1[i][15:0] の 32 ビットアクセスが可能です。詳細については、以下のレジスタを参照してください。

rPWMTimer_TIMEBASE_1[i]_k, rPWMTimer_TIMEBASE_01_1[k], rPWMTimer_TIMEBASE_23_1[k],
rPWMTimer_TIMEBASE_45_1[k], rPWMTimer_TIMEBASE_67_1[k], rPWMTimer_TIMEBASE_12_1[k],
rPWMTimer_TIMEBASE_34_1[k], rPWMTimer_TIMEBASE_56_1[k] (i = 0~7, k = 0~1)

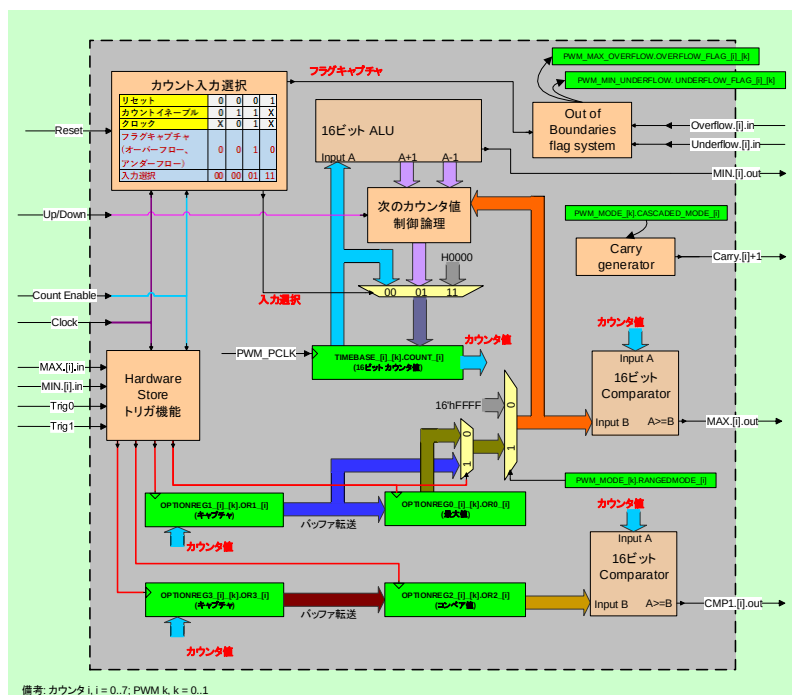


図 1.12 Timebase モジュール

(a) Timebase モジュールの信号生成

Timebase モジュールの信号生成：

- Max 検出：Timebase カウンタ = 最大値（レンジモード）／16'hFFFF（フリーランニングモード）
- Min 検出：Timebase カウンタ = 16'h0000
- オーバーフロー検出：Max 検出かつカウントアップ状態
- アンダーフロー検出：Min 検出かつカウントダウン状態
- コンペアマッチ：Timebase カウンタ $\geq$ コンペアレジスタ

Timebase モジュールの信号生成例を示します。

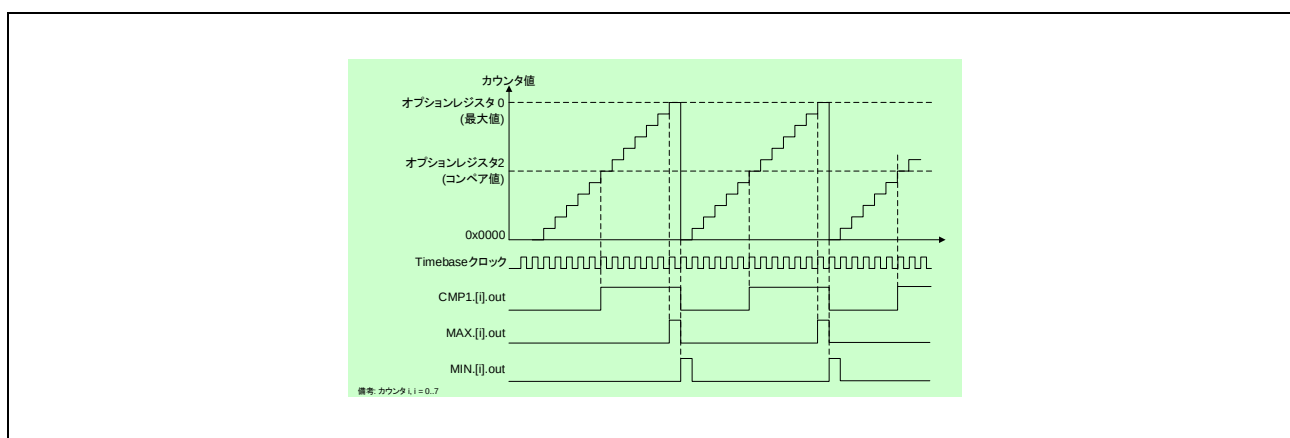


図 1.13 のこぎり波の信号生成

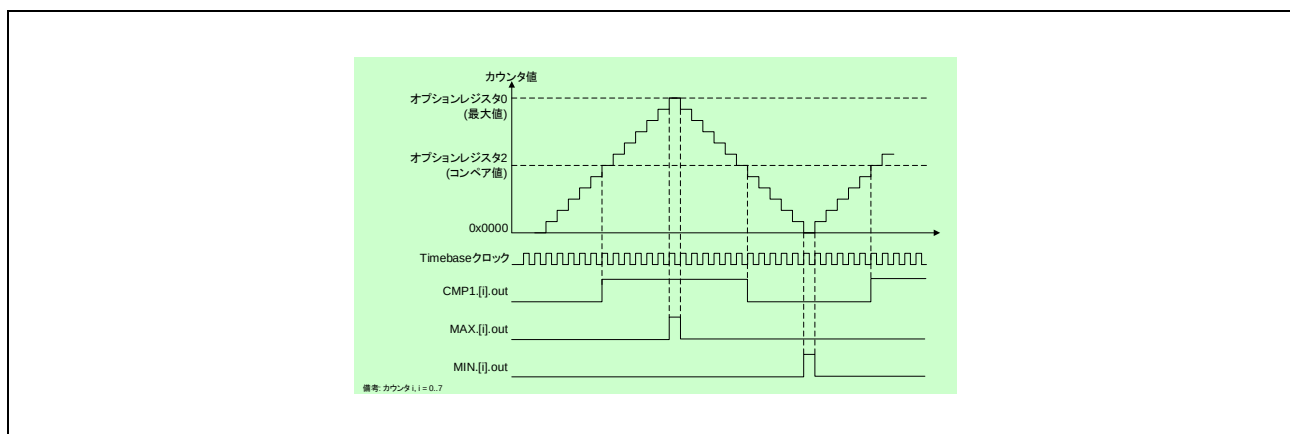


図 1.14 三角波の信号生成

オーバーフロー／アンダーフロー検出によって、割り込み生成が可能です。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_MAX_OVERFLOW, rPWMTimer_PWM_MIN_UNDERFLOW

(b) オプションレジスタのシステム

各 Timebase モジュールに 4 個のオプションレジスタがあり、各機能に使用可能です。

表 1.64 オプションレジスタ機能

オプションレジスタ (OR)	コンペアマッチ	キャプチャ	最大値	バッファ転送
OR0_[i]	—	—	使用可能	OR1_[i]から転送
OR1_[i]	—	使用可能	—	OR0_[i]へ転送
OR2_[i]	CMP1_[i]	—	—	OR3_[i]から転送
OR3_[i]	—	使用可能	—	OR2_[i]へ転送

機能説明：

- コンペアマッチ：COUNT_[i] $\geq$ OR2_[i]ならば、内部信号 CMP1_[i]に 1 をセット
- キャプチャ：トリガにより、COUNT_[i]を OR1_[i]/OR3_[i]に格納
- 最大値：レンジモードの場合、OR0_[i]により Overflow_[i], MAX_[i]信号を制御し、アンダーフロー検出で OR0_[i]からの最大値を COUNT_[i]にロード
- バッファ転送：トリガにより、OR1_[i]を OR0_[i]へ転送
- バッファ転送：トリガにより、OR3_[i]を OR2_[i]へ転送

キャプチャおよびバッファ転送のトリガは、Hardware Store トリガ機能で制御されます。

バッファ転送の設定については、以下のレジスタを参照してください。

rPWMTimer_TIMEBASE_MUX_[q]_[k] (q = 0~1, k = 0~1)

オプションレジスタについては、以下のレジスタを参照してください。

rPWMTimer_OPTIONREG[r]_[i]_[k], rPWMTimer_OPTIONREG[r]_01_[k], rPWMTimer_OPTIONREG[r]_23_[k],
rPWMTimer_OPTIONREG[r]_45_[k], rPWMTimer_OPTIONREG[r]_67_[k], rPWMTimer_OPTIONREG[r]_12_[k],
rPWMTimer_OPTIONREG[r]_34_[k], rPWMTimer_OPTIONREG[r]_56_[k] (r = 0~3, i = 0~7, k = 0~1)

(c) Hardware Store トリガ機能

Hardware Store トリガ機能は、各オプションレジスタのトリガを処理します。本トリガは、レジスタの設定に従い、キャプチャおよびバッファ転送に使用されます。

使用可能なトリガ入力：

- Max 検出 (Trig2 を使用)
- Min 検出 (Trig2 を使用)
- 外部入力 (Trig0 または Trig1 を使用)

Timebase クロック同期トリガと非同期トリガ：

同期モード (OR0_[i], OR2_[i]のみ使用可能) では、トリガは次の Timebase クロックが入るまで保留されます。カウントイネーブルが“無効”に切り替わると、保留されているトリガはただちに転送されます。キャプチャ値を転送する場合は非同期モードを、それ以外は同期モードに設定してください。

ソフトウェアによるトリガのロック：

- ロック：rPWMTimer_HWSTORE_LOCK_[k].bHWSTORE_OR[r]_LOCK_[i] (r = 0..3, i = 0..7, k = 0..1)
- ロック解除：
rPWMTimer_HWSTORE_UNLOCK_[k].bHWSTORE_OR[r]_UNLOCK_[i] (r = 0..3, i = 0..7, k = 0..1)
- リセット後のトリガ：ロック状態

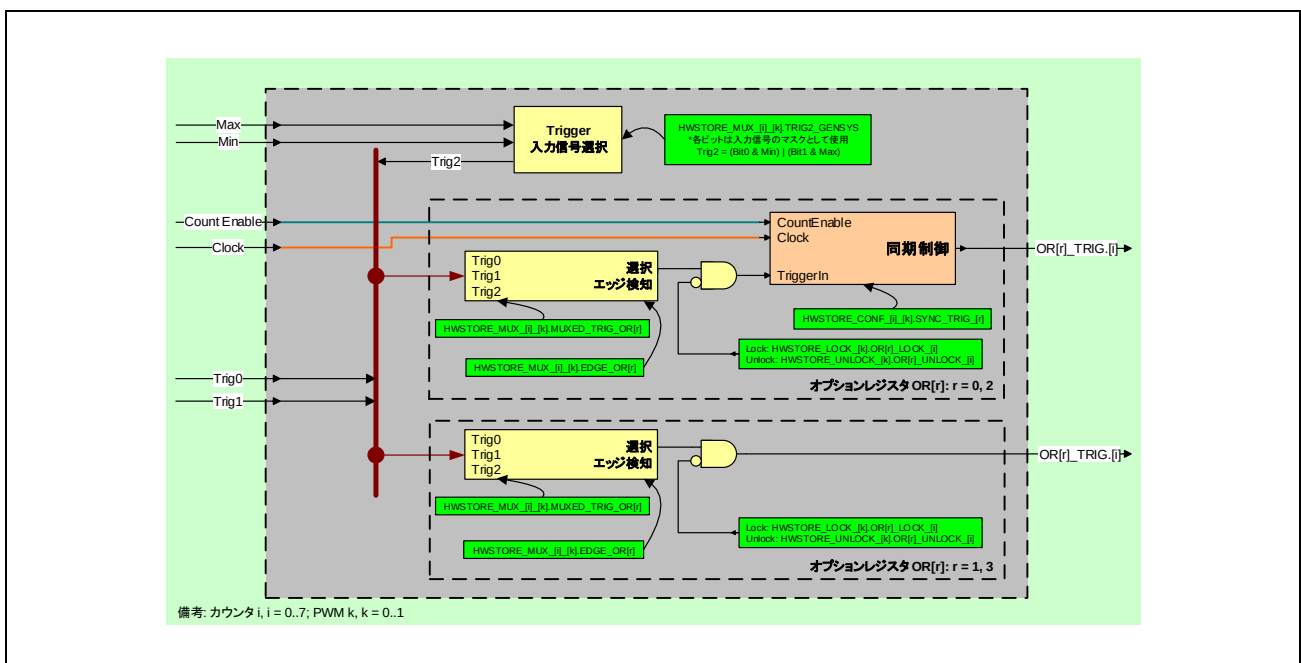


図 1.15 Hardware Store トリガ機能

詳細については、以下のレジスタを参照してください。

rPWMTimer_HWSTORE_CONF_[i]_[k], rPWMTimer_HWSTORE_MUX_[i]_[k],
rPWMTimer_HWSTORE_LOCK_[k], rPWMTimer_HWSTORE_UNLOCK_[k] (i = 0~7, k = 0~1)

(d) レンジモード

レンジモードでは、オプションレジスタ 0 の値がカウンタの最大値になります。COUNT[i]とオプションレジスタ 0 を比較し、MAX.[i], Overflow.[i]信号は生成されます。本モードにおいては、オーバーフローで 0x0000 を COUNT[i]に、アンダーフローで最大値（オプションレジスタ 0）を COUNT[i]にロードします。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_MODE_[k] (k = 0~1)

(3) Outputs Generator

Outputs Generator は、PWM モジュールの出力信号を制御します。

CMP1.[i]出力信号：

- カウントイネーブル = 1
 - ダイレクト：Timebase からのコンペアマッチ信号（CMP1[i].in）
- カウントイネーブル = 0（無効時）
 - リセット： 0 固定
 - セット： 1 固定

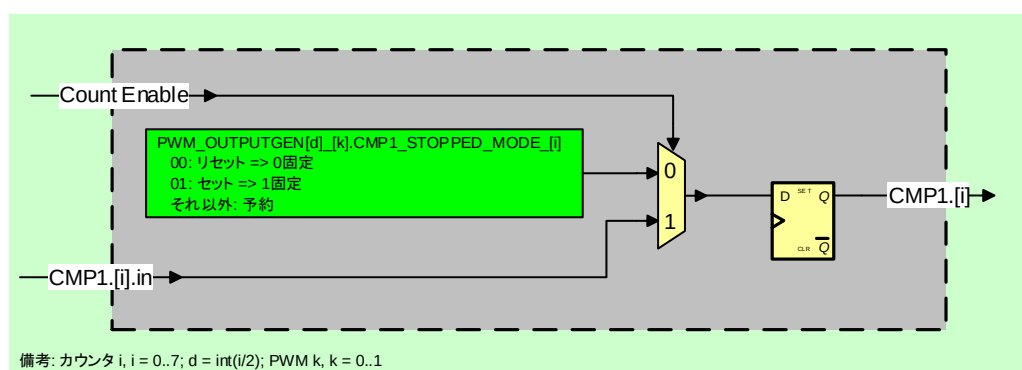


図 1.16 Outputs Generator

CMP1.[i]信号は、PWM_PCLK クロックで同期化されます。

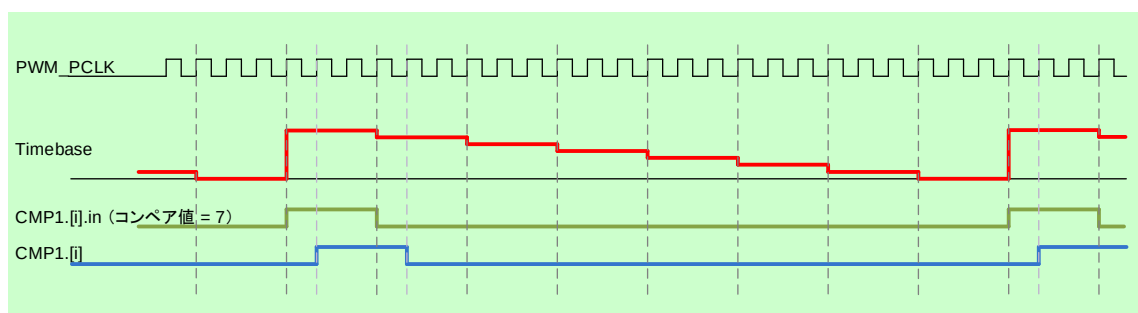


図 1.17 Outputs Generator タイミングチャート

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_OUTPUTGEN[d].[k] (d = 0~3, k = 0~1)

1.5.6.3 シングルカウンタモジュールのカスケード

PWM モジュールは、2 個のシングルカウンタを使用して 32 ビット長のカウンタを生成することが可能です。

カスケードモードで入力信号を設定する場合は、それぞれ 2 つのシングルカウンタに設定する必要があります。たとえば、カスケードカウンタでクロックに外部入力を使用する場合、それぞれのシングルカウンタにクロック入力を設定してください。

カスケードモード時は、関連する信号はカスケード論理で生成され、それぞれのシングルカウンタ入力となります。

キャリーは、上位カウンタの内部イネーブル信号として使用され、カウンタ値を増減します。

キャリーが 1 にセットされた場合：

- 下位カウンタ値は、16'h0000 から 16'hFFFF へ（ダウンカウンタ時）
- 下位カウンタ値は、16'hFFFF から 16'h0000 へ（アップカウンタ時）
- 32 ビットカウンタ値は、0 から最大値へ（ダウンカウンタ時）
- 32 ビットカウンタ値は、最大値から 0 へ（アップカウンタ時）

カスケードモード使用時の注意：

- シングルカウンタの入力選択とモード（レンジ／フリーランニング）は、カスケード接続しているそれぞれ 2 つのカウンタで同じ設定にする必要があります。

備 考

カスケード論理[i]は前カウンタ[i-1]を下位 16 ビットとして、32 ビットカウンタを構成します。32 ビットのカウンタを生成するときは、上位 16 ビットカウンタ[i]のみカスケードモードを有効にしてください。また、上位 16 ビットカウンタのカスケード値（rPWMTimer_SCALEVALUE_[i]_[k]）を 0 に設定してください。

詳細については、以下のレジスタを参照してください。

rPWMTimer_PWM_MODE_[k] (k = 0～1)

1.5.6.4 シングルカウンタモジュールのマスタカウンタ機能

rPWMTimer_PWM_MODE_ $[k]$ ($k = 0 \sim 1$) レジスタにより設定可能な、マスタカウンタ機能があります。

本機能を有効にしたシングルカウンタ「S」は、選択したシングルカウンタ「M」のカウンタ値に置き換わります。シングルカウンタ「S」は選択したシングルカウンタ「M」のスレーブとなり、カウンタ値の変更はできなくなります。コンペアおよびキャプチャ機能においても、シングルカウンタ「M」からの現カウンタ値が適用されます。カウンタ値とともに、オーバーフロー、アンダーフロー、Max、および Min 信号もマスタカウンタからスレーブへと転送されます。

本機能は間接接続が可能です。元になるマスタカウンタ値が伝わるため、最大 8 個のシングルカウンタで、使用する機能（コンペアおよびキャプチャ）にすべて同じカウンタ値を適用することが可能です。

例) シングルカウンタ 2 は、シングルカウンタ 0 をマスタとし、シングルカウンタ 3 は、シングルカウンタ 2 をマスタとすれば、シングルカウンタ 3 は、シングルカウンタ 0 が実質のマスタとなります。以下の図に示します。

本機能を使用する場合は、スレーブとマスタで以下の制限があります。

- 同じモード設定（レンジモード、カスケードモード）
- クロック、カウントイネーブル、および Up/Down 信号は同じ設定
- ソフトウェアリセットは、同アクセスで実行

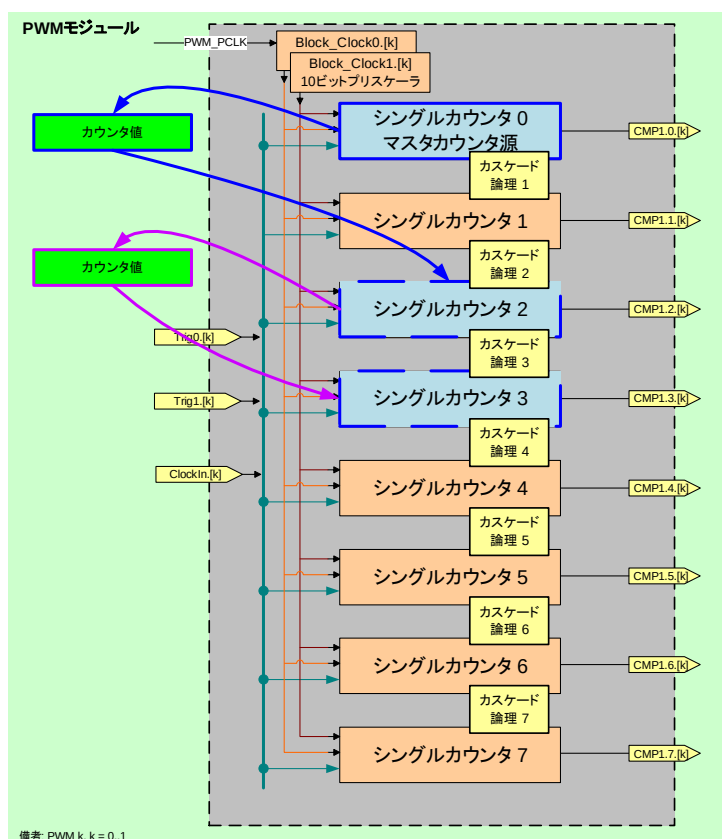
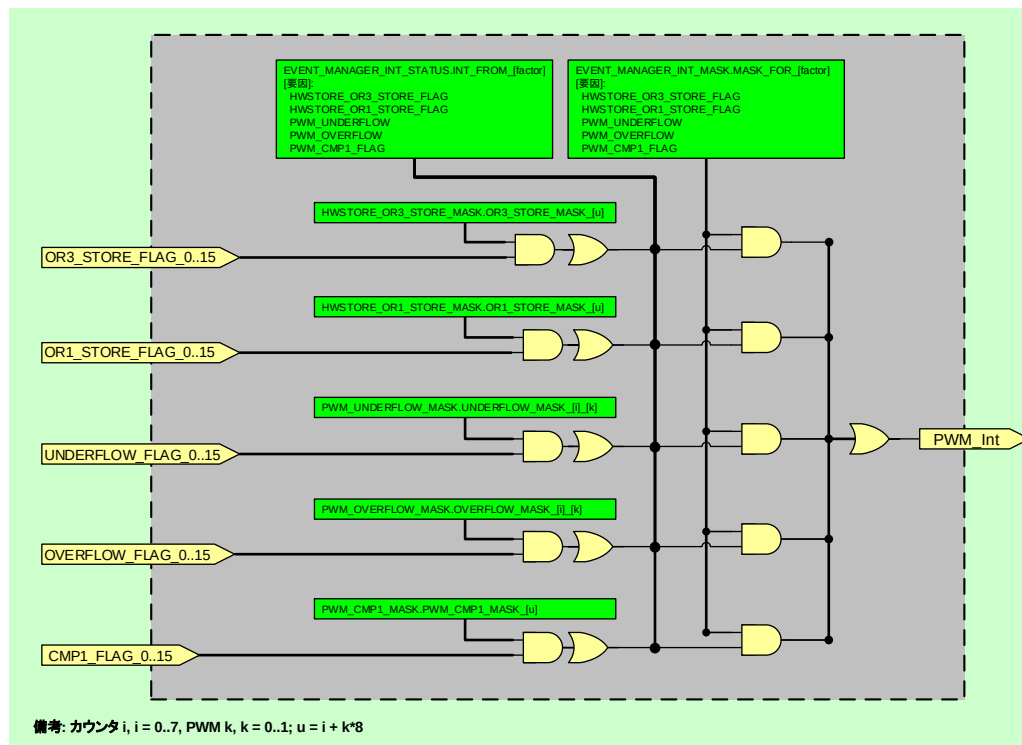


図 1.18 マスタカウンタ機能の例

1.5.7 Event Manager モジュール

Event Manager モジュールは、PWMTimer 割り込みを生成します。

割り込み生成可能な要因を下図に示します。Event Manager モジュールは、フラグを集約し、マスクされていないフラグによって PWMTimer 割り込みを発生します。



備考 フラグがセットされると、ソフトウェアによりフラグをクリアする必要があります。

図 1.19 Event Manager モジュール

Event Manager には、各要因フラグが入力され、2 レベルでマスクが可能です。

フラグのステータスは、各要因のマスクレジスタ設定によりグローバル割り込みステータスレジスタに反映されます。グローバル割り込みマスクレジスタによって、グローバル割り込みステータスはマスクされ、割り込みが生成されます。割り込みを禁止するためには、2 レベルのどちらかでマスクするか、要因自体をクリアする必要があります。

詳細については、以下のレジスタを参照してください。

rPWMTimer_EVENT_MANAGER_INT_STATUS, rPWMTimer_PWM_CMP1_FLAG_POLARITY,
rPWMTimer_PWM_CMP1_FLAG, rPWMTimer_PWM_MAX_OVERFLOW,
rPWMTimer_PWM_MIN_UNDERFLOW, rPWMTimer_HWSTORE_OR1_STORE_FLAG,
rPWMTimer_HWSTORE_OR3_STORE_FLAG, rPWMTimer_EVENT_MANAGER_INT_MASK,
rPWMTimer_PWM_CMP1_MASK, rPWMTimer_PWM_OVERFLOW_MASK,
rPWMTimer_PWM_UNDERFLOW_MASK, rPWMTimer_HWSTORE_OR1_STORE_MASK,
rPWMTimer_HWSTORE_OR3_STORE_MASK

1.5.8 PWMTimer プログラミング

1.5.8.1 共通な初期設定例

推奨する基本動作の初期設定を以下に示します。

- OutputCtrl モード有効

各ビットを 1'b1 に設定 : rPWMTimer_OUTPUTCTRL_[n].bBYPASS_MODE_[u] (u = 0~15, n = 0~3)

rPWMTimer_OUTPUTCTRL_[n] = 0x40404040

- シングルカウンタ Trig0 入力有効

各ビットを 3'b001 に設定 : rPWMTimer_PWM_TRIG0_[k].bTRIG0_[i] (i = 0~7, k = 0~1)

rPWMTimer_PWM_TRIG0_[k] = 0x11111111

- シングルカウンタ Trig1 入力有効

各ビットを 3'b001 に設定 : rPWMTimer_PWM_TRIG1_[k].bTRIG1_[i] (i = 0~7, k = 0~1)

rPWMTimer_PWM_TRIG1_[k] = 0x11111111

- オプションレジスタのバッファ転送有効

各ビットを 1'b1 に設定 : PWMTimer_TIMEBASE_MUX_[q]_[k].bTIMEBASE_MUX_OR[r]_[i]
(r = 0, 2, i = 0~7, q = 0~1, k = 0~1)

rPWMTimer_TIMEBASE_MUX_[q]_[k] = 0x03030303

- オプションレジスタのトリガ設定 (キャプチャ値をバッファ転送する場合)

各ビットを 1'b0 に設定 : rPWMTimer_HWSTORE_CONF_[i]_[k].bHWSTORE_SYNC_TRIG_[r]
(r = 0, 2, i = 0~7, k = 0~1)

rPWMTimer_HWSTORE_CONF_[i]_[k] = 0x00000000

- オプションレジスタのロック解除

各ビットを 1'b1 に設定 : rPWMTimer_HWSTORE_UNLOCK_[k].bHWSTORE_OR[r]_UNLOCK_[i]
(r = 0~3, i = 0~7, k = 0~1)

rPWMTimer_HWSTORE_UNLOCK_[k] = 0xFFFFFFFF

- OutputCtrl の入力有効

各ビットを $(5'h03 + [n \bmod 8] * 2)$ に設定 :

rPWMTimer_ROUTING_OUTCTRL_[n].bROUTING_OUTCTRL_SET_[n] (n = 0~15)

rPWMTimer_ROUTING_OUTCTRL_0/rPWMTimer_ROUTING_OUTCTRL_8 = 0x00000300

rPWMTimer_ROUTING_OUTCTRL_1/rPWMTimer_ROUTING_OUTCTRL_9 = 0x00000500

rPWMTimer_ROUTING_OUTCTRL_2/rPWMTimer_ROUTING_OUTCTRL_10 = 0x00000700

rPWMTimer_ROUTING_OUTCTRL_3/rPWMTimer_ROUTING_OUTCTRL_11 = 0x00000900

rPWMTimer_ROUTING_OUTCTRL_4/rPWMTimer_ROUTING_OUTCTRL_12 = 0x00000B00

rPWMTimer_ROUTING_OUTCTRL_5/rPWMTimer_ROUTING_OUTCTRL_13 = 0x00000D00

rPWMTimer_ROUTING_OUTCTRL_6/rPWMTimer_ROUTING_OUTCTRL_14 = 0x00000F00

rPWMTimer_ROUTING_OUTCTRL_7/rPWMTimer_ROUTING_OUTCTRL_15 = 0x00001100

1.5.8.2 コンペアマッチによる PWM 出力

コンペアマッチ信号（カウンタ値 $\geq$ コンペア値）を、PWM_OUT から出力可能です。

この例では、レンジモードでアップカウンタを動作させ、コンペアマッチ信号を出力しています。カウンタ値がオプションレジスタ 0（最大値）の値に達すると、オーバーフロー要因により PWM_Int から割り込みが発生します。

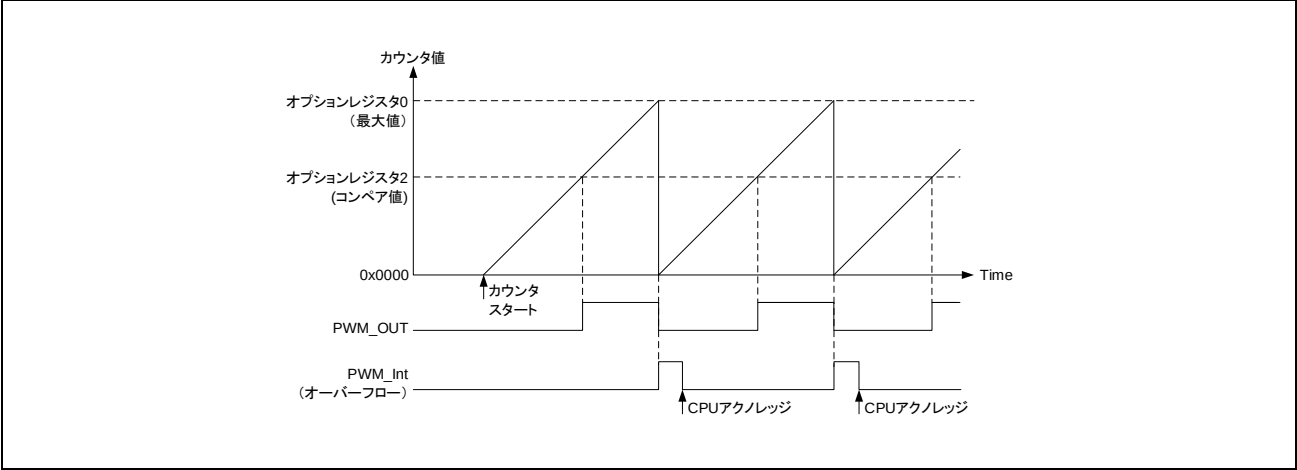


図 1.20 PWM 出力動作例

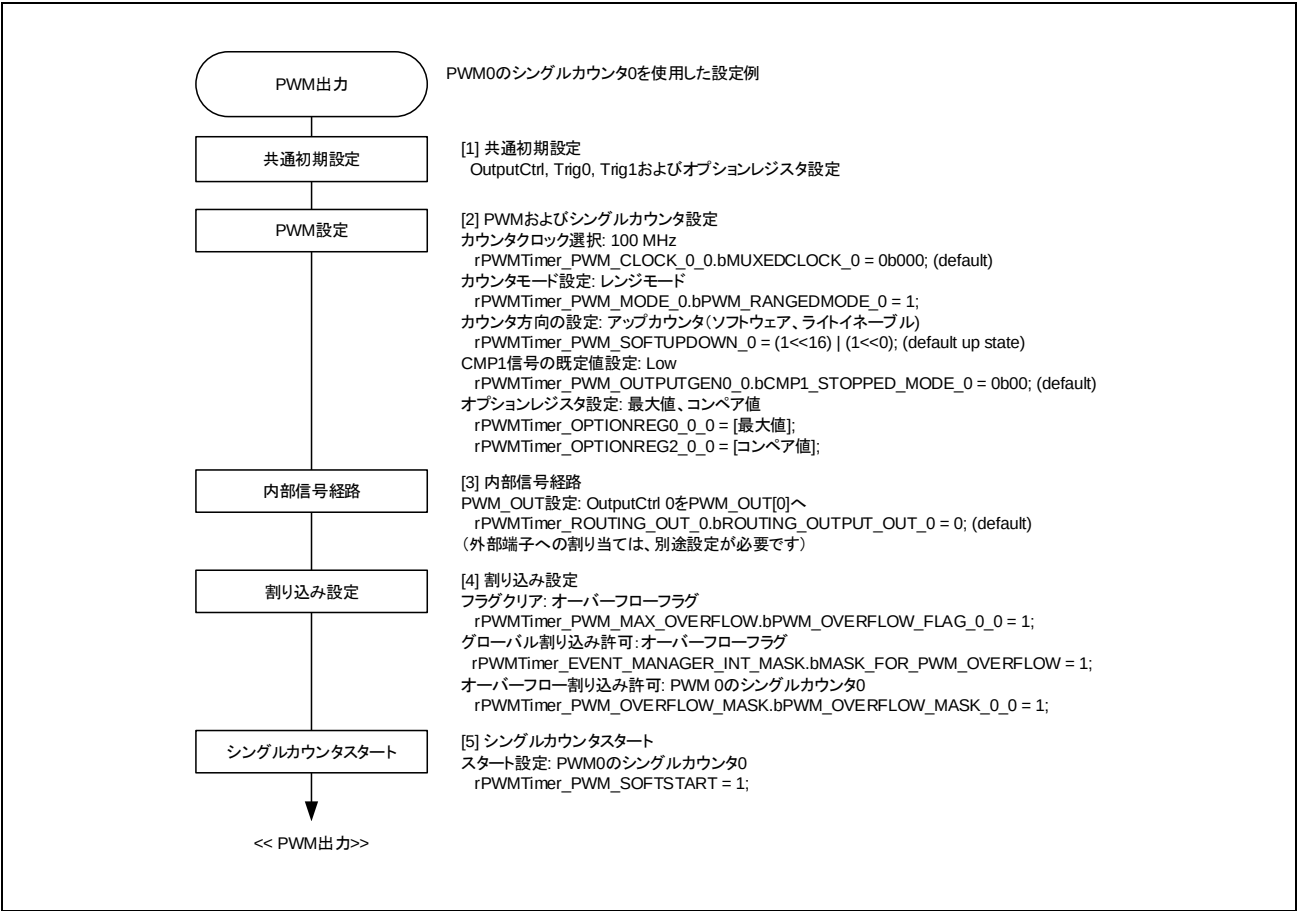


図 1.21 PWM 出力設定例

1.5.8.3 マスタカウンタ

マスタカウンタ機能を使用して、キャプチャ入力またはコンペアマッチ出力の数を増やすことが可能です。詳細は、「1.5.6.4 シングルカウンタモジュールのマスタカウンタ機能」を参照してください。

この例では、マスタカウンタ機能で 3 つのカウンタを同期動作させ、それぞれのコンペアマッチ信号を PWM_OUT から出力しています。

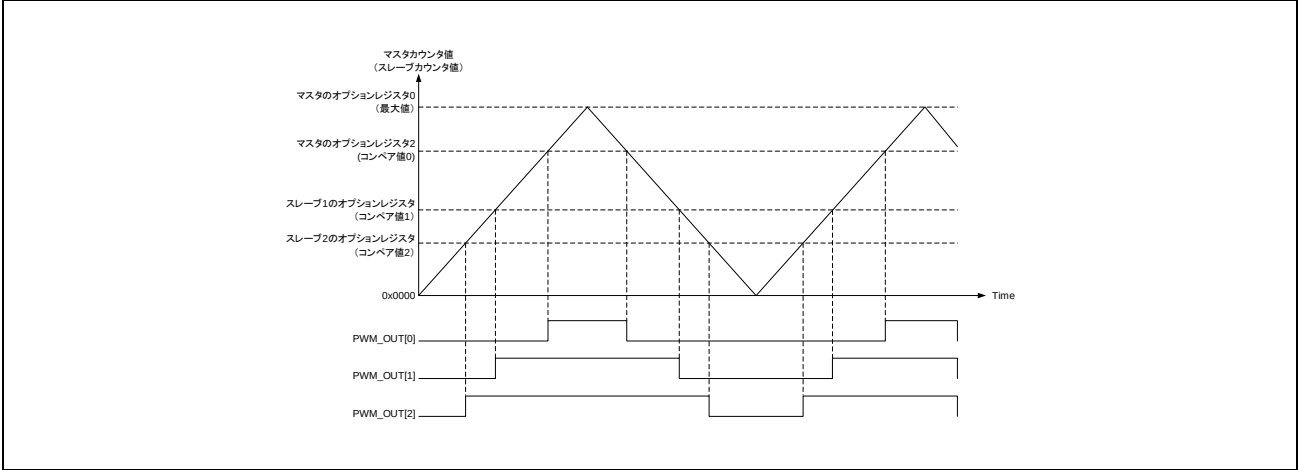


図 1.22 マスタカウンタ動作例

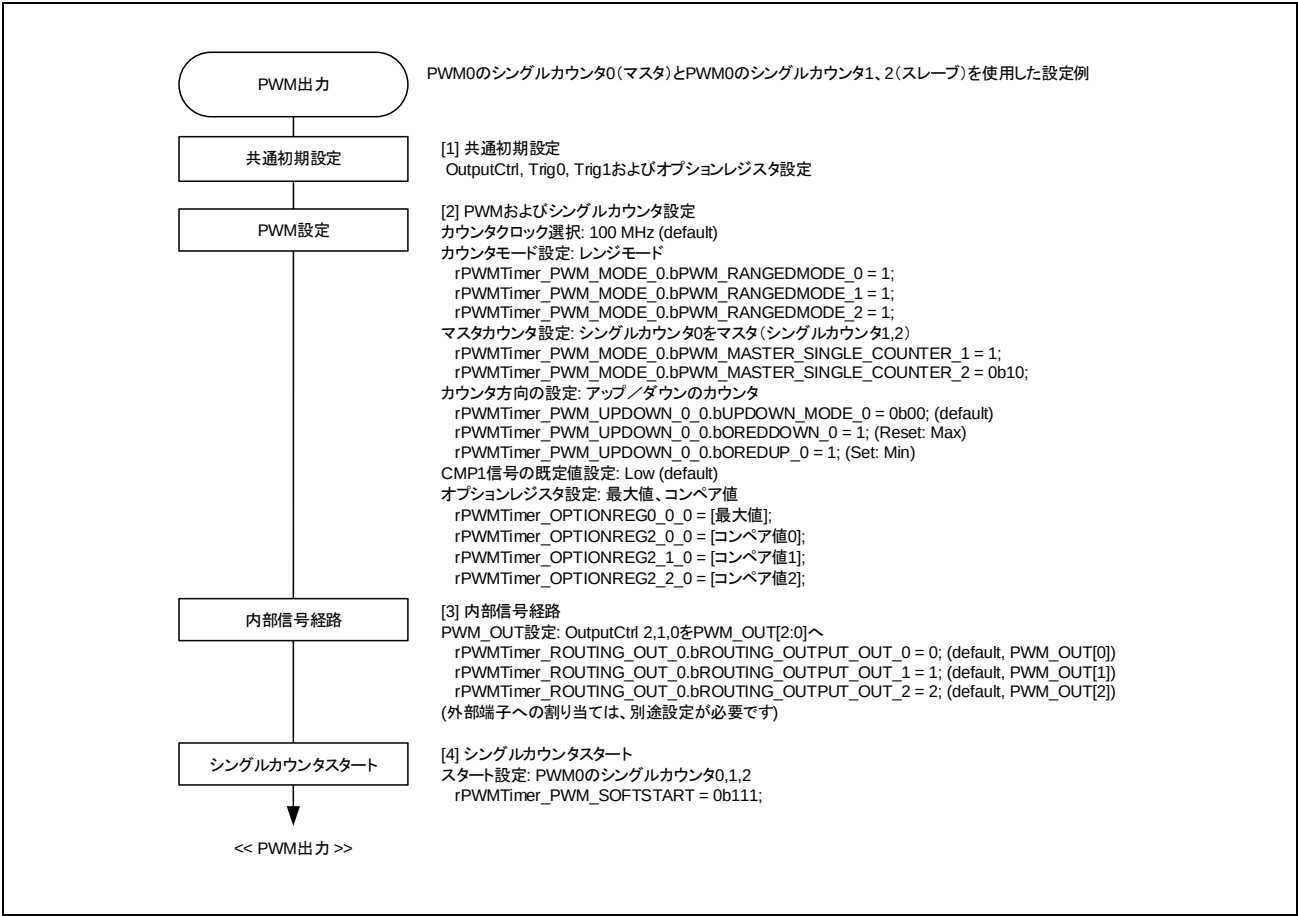


図 1.23 マスタカウンタ設定例

1.5.8.4 インプットキャプチャ

PWM_IN をトリガ入力として、カウンタ値のキャプチャおよびバッファ転送が可能です。

この例では、PWM_IN の立ち上がりエッジで、オプションレジスタ 3 にカウンタ値をキャプチャし、オプションレジスタ 2 へバッファ転送しています。

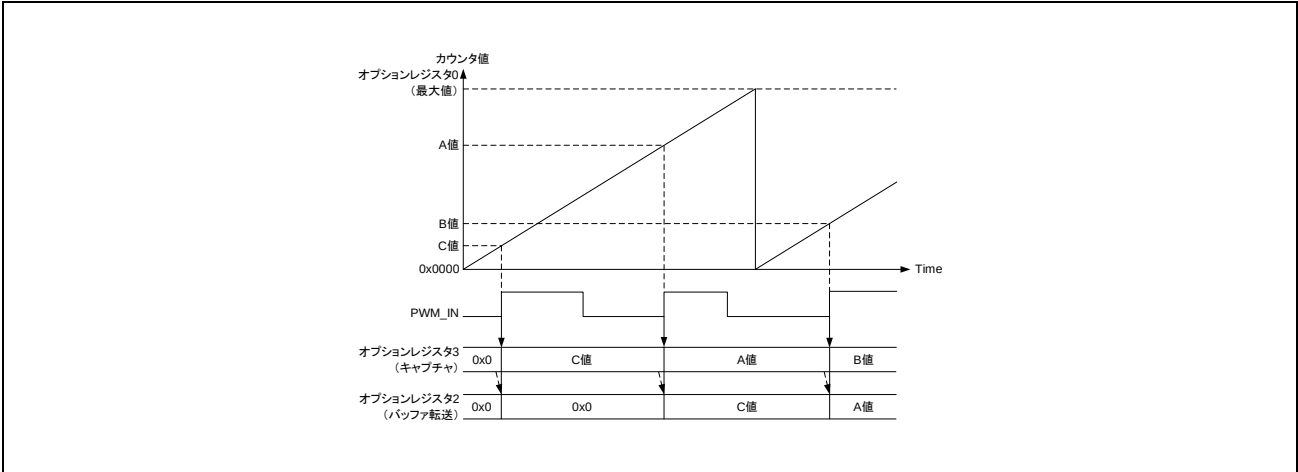


図 1.24 インプットキャプチャ動作例

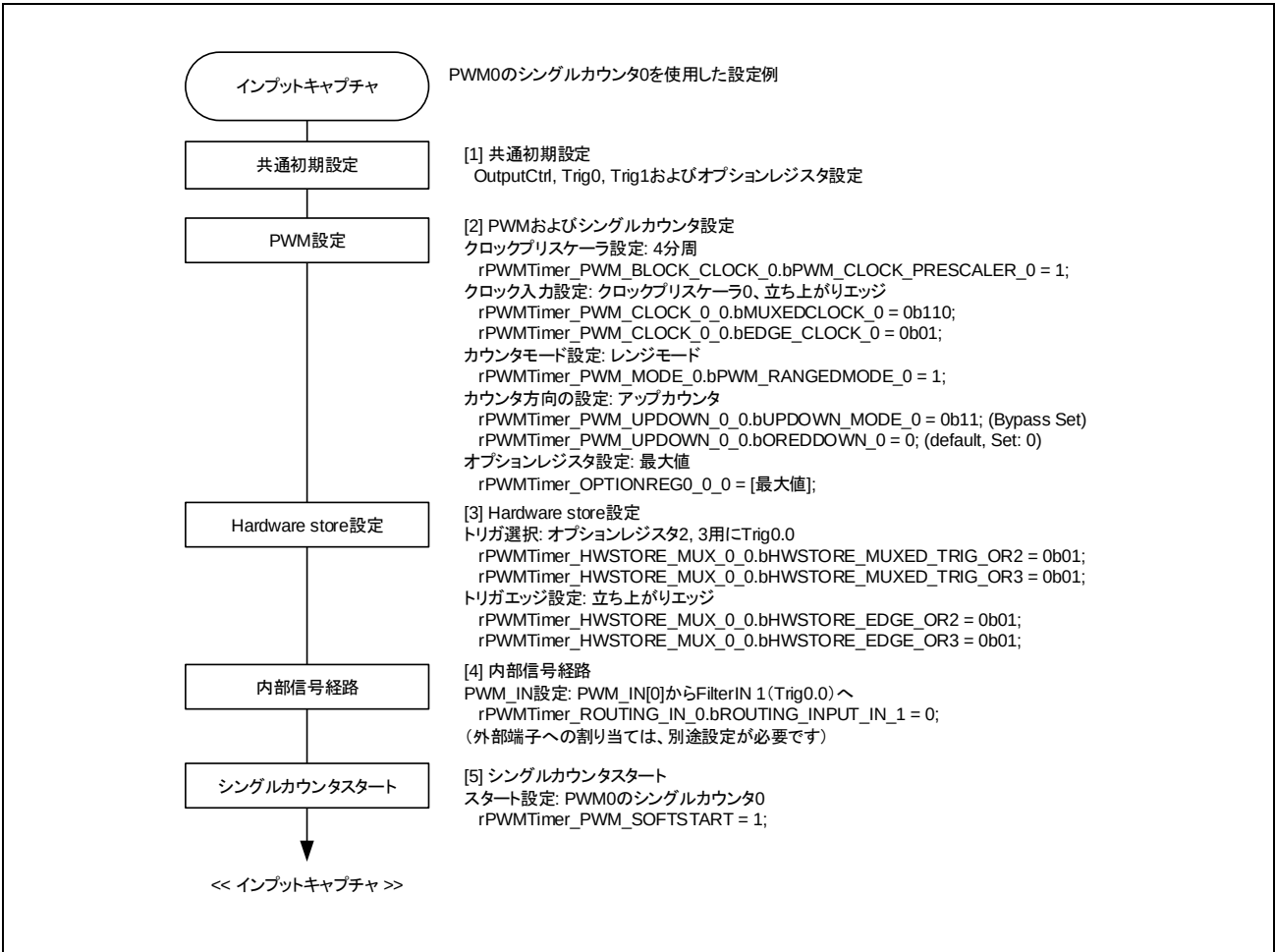


図 1.25 インプットキャプチャ設定例

1.5.8.5 パルス測定

PWM_IN をエッジ検出なしのクロック入力として使用することで、パルス幅の測定が可能になります。

この例では、クロック入力とトリガ入力に同じ PWM_IN を割り当て、トリガの立ち下がりエッジでキャプチャおよびバッファ転送をしています。オプションレジスタ 3 のトリガ割り込み (OR3_STORE_FLAG) 時に、オプションレジスタ 3 値からオプションレジスタ 2 値を引くことでパルス幅の測定が可能です。

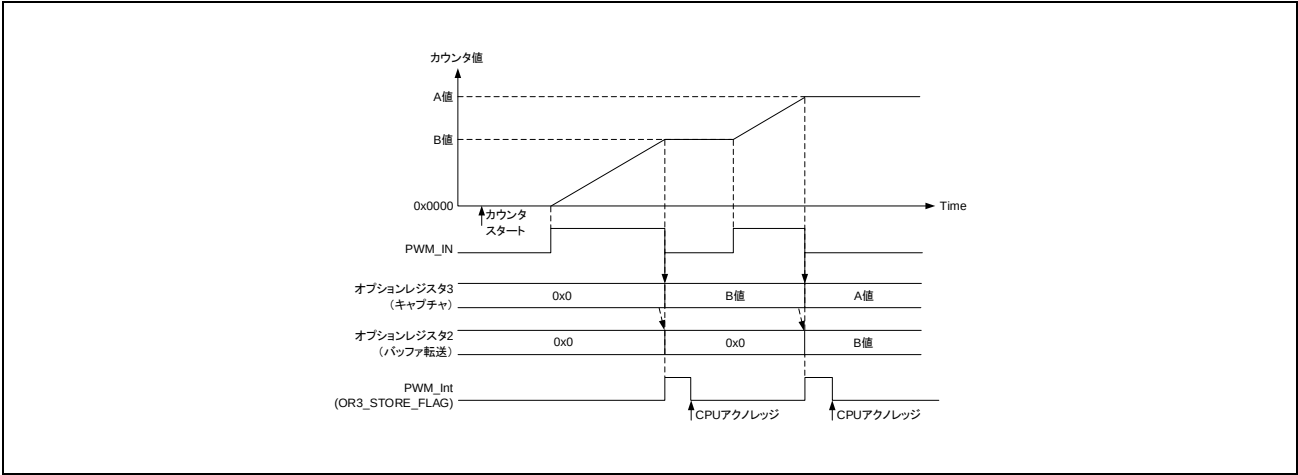


図 1.26 パルス測定動作例

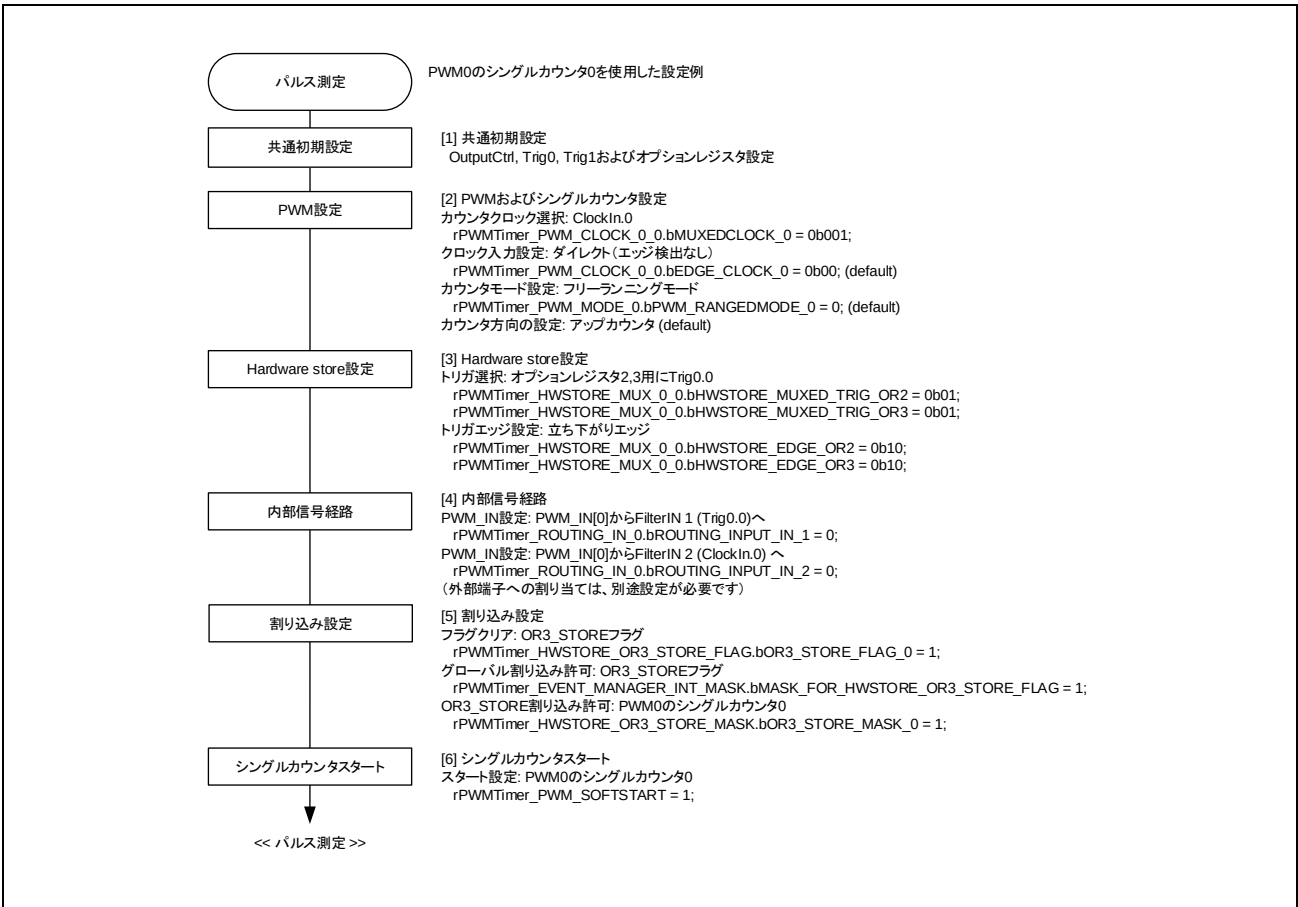


図 1.27 パルス測定設定例

改訂記録	RZ/N1Dグループ、RZ/N1Sグループ、RZ/N1Lグループ ユーザーズマニュアル PWMTimer
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2020.09.30	—	初版発行

RZ/N1Dグループ、RZ/N1Sグループ、RZ/N1Lグループ
ユーザーズマニュアル PWMTimer

発行年月日 2020年09月30日 Rev.1.00

発行 ルネサス エレクトロニクス株式会社

〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

RZ/N1D グループ、RZ/N1S グループ、RZ/N1L グループ



ルネサスエレクトロニクス株式会社

R01UH0913JJ0100