

特徴

- AVR[®] RISC構造の利用
- 高性能、低消費AVR[®] RISC構造
 - 強力な123命令(多くは1周期で実行)
 - 32個の1バイト長汎用レジスタ
 - 完全なステティック動作
 - 20MHz時、20MIPSに達する高速動作
- データメモリと不揮発性プログラムメモリ
 - 実装自己書き換え(ISP)可能な2Kバイト(1K語)フラッシュメモリ内蔵
10,000回の書き換え可能
 - 実装書き換え可能な128バイトのEEPROM
100,000回の書き換え可能
 - 128バイトの内蔵SRAM
 - プログラム用フラッシュメモリとデータ用EEPROM保護用の設定可能な施錠機能
- 内蔵周辺機能
 - 分離された前置分周器と比較動作付き1つの8ビットタイマ/カウンタ
 - 分離された前置分周器と比較、捕獲動作付き1つの16ビットタイマ/カウンタ
 - 4つのPWM出力
 - アナログ比較器
 - 設定可能な専用発振器付きウォッチドッグタイマ
 - 多用途直列インターフェース(USI)
 - 全二重USART
- 特殊マイクロコントローラ機能
 - デバッグWIRE内蔵デバッグ機能
 - SPIポート経由の実装書き込み
 - 外部及び内部の割り込み
 - アイドル、パワーダウン、スタンバイの3つの低消費動作
 - 強化した電源ONリセット回路
 - 設定可能な低電圧検出器(BOD)回路
 - 校正付き内蔵RC発振器
- I/Oと外圍器
 - 18ビットの設定可能なI/O
 - 20ピンPDIP、20リードSOIC、20ピンQFN/MLF
- 動作電圧
 - 1.8~5.5V (ATtiny2313V)
 - 2.7~5.5V (ATtiny2313)
- 動作速度
 - ATtiny2313V :
 - 0~4MHz/1.8~5.5V
 - 0~10MHz/2.7~5.5V
 - ATtiny2313 :
 - 0~10MHz/2.7~5.5V
 - 0~20MHz/4.5~5.5V
- 代表消費電力
 - 230μA (1MHz, 1.8V, 活動動作)
 - 20μA (32kHz, 1.8V, 活動動作)
 - 0.1μA未満 (1.8V, パワーダウン動作)



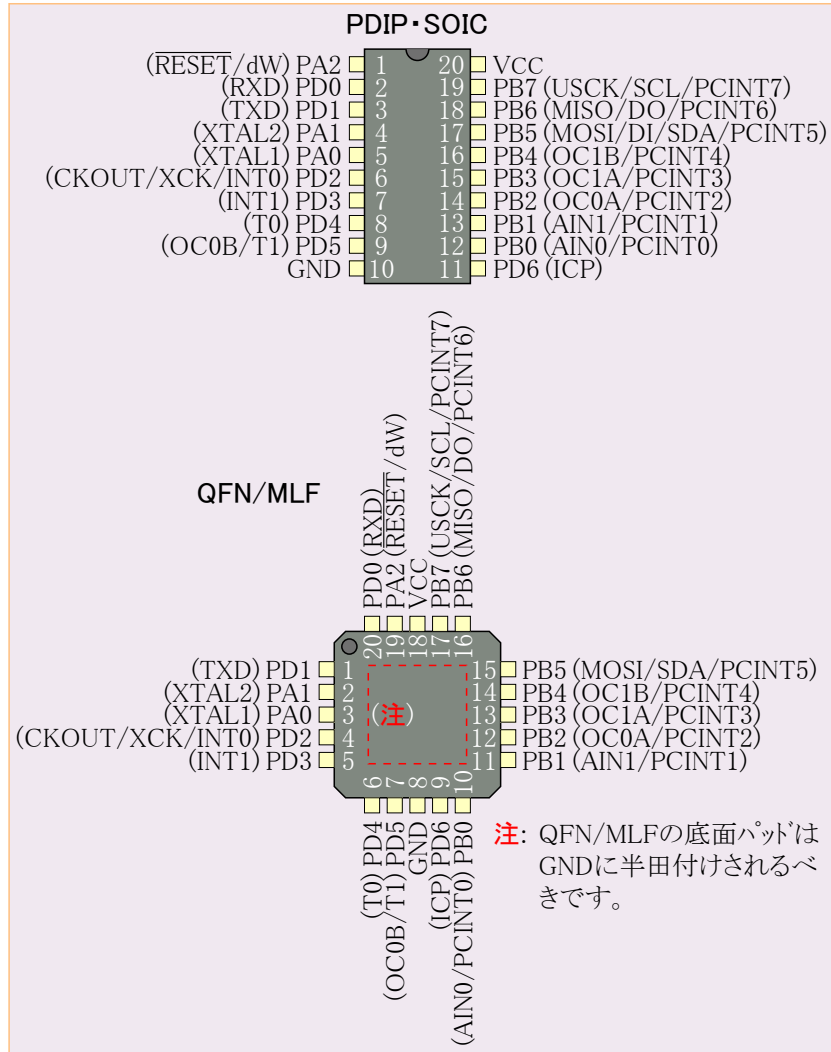
8ビット AVR[®]
 マイクロコントローラ
 実装書き換え可能な
 2Kバイト
 フラッシュメモリ内蔵

ATtiny2313
 ATtiny2313V

本書は一般の方々の便宜のため有志によって作成されたもので、Atmel社とは無関係であることを御承知ください。しおりの[はじめに]での内容にご注意ください。

Rev. 2543M-10/16, 2543MJ4-10/22

ピン配置

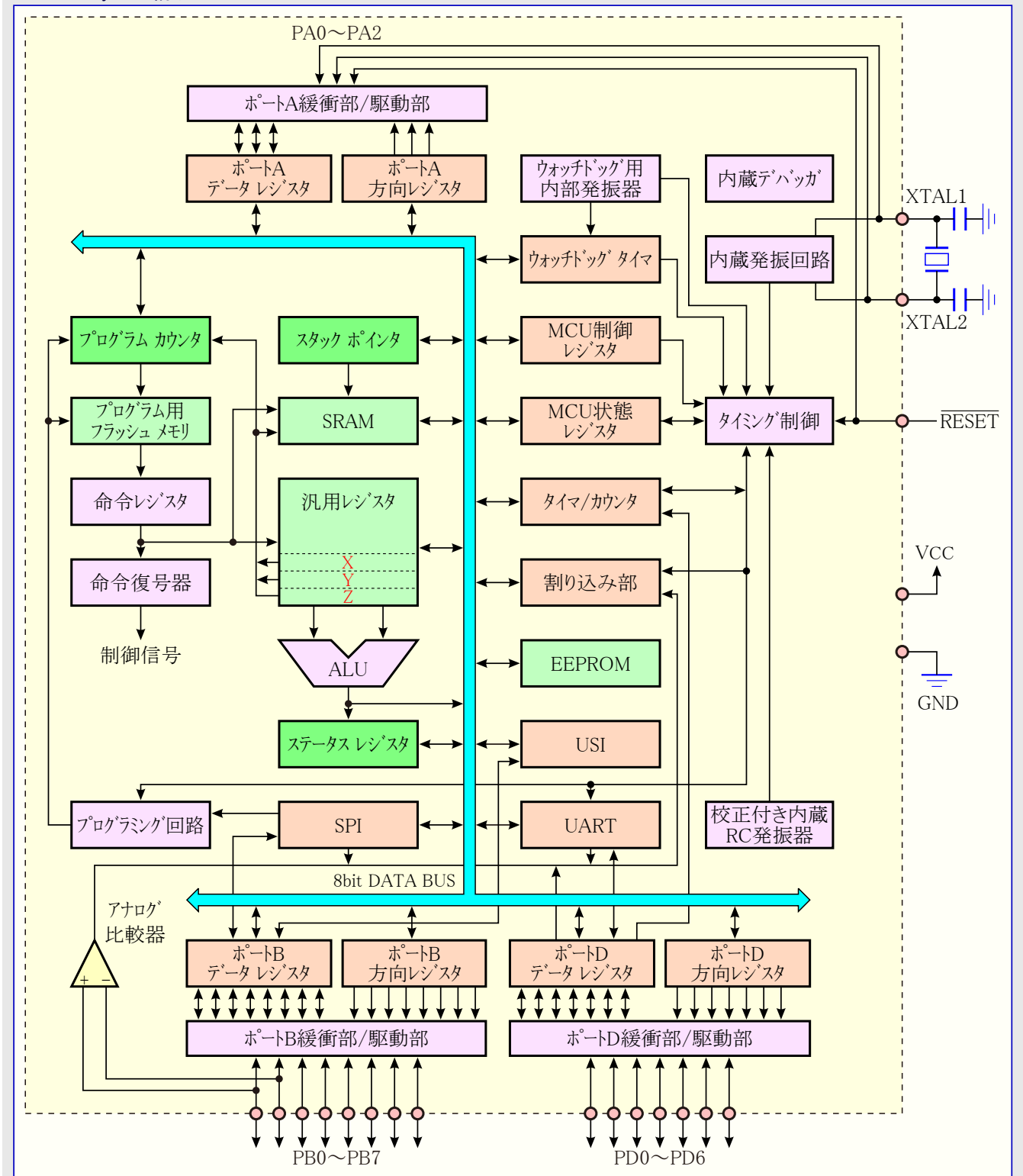


概要

ATtiny2313はAVR強化RISC構造を基にした低消費CMOS 8ビット マイクロ コントローラです。1周期での強力な命令の実行により、ATtiny2313はMHzあたり1 MIPSに達する単位処理量を成し遂げ、処理速度対消費電力の最適化を設計者に許します。

構成図

図2. ATtiny2313構成図



AVRコアは32個の汎用作業レジスタと豊富な命令群の組み合わせです。32個の全レジスタはALU(Arithmetic Logic Unit)に直結され、レジスタ間命令は1クロック周期で実行されます。AVR構造は現状のCISC型マイクロコントローラに対して最大10倍の単位処理量向上効果があります。

ATtiny2313は2Kバイトの実装書き込み可能なフラッシュメモリ、128バイトのEEPROM、128バイトのSRAM、18本の汎用入出力線、32個の汎用作業レジスタ、内蔵デバッグ用の単線インターフェース、比較動作付きの2つの柔軟なタイマ/カウンタ、内部及び外部割り込み、設定変更可能な直列USART、開始条件検出器付き多用途直列インターフェース(USD)、内蔵発振器付きの設定変更可能なウォッチドッグタイマ、ソフトウェアで選べる3つの低消費動作を提供します。アイドル動作はCPUを停止し、一方SRAM、タイマ/カウンタ、割り込み機構に機能の継続を許します。パワーダウン動作は発振器を停止しますがレジスタの内容を保護し、以降のハードウェアリセットか外部割り込みまで他の全機能を禁止します。スタンバイ動作では水晶発振子/セラミック振動子用発振器が動作する一方で、デバイスのその他は休止します。これは低消費電力と非常に速い起動の組み合わせを許します。

本デバイスはAtmelの高密度不揮発性メモリ技術を使って製造されています。内蔵の実装書き換え(ISP)可能なフラッシュメモリはプログラムメモリに使い、規定の不揮発性メモリ書き込み器かSPI直列インターフェース経由によって再書き込みができます。モリシックチップ上の実装書き換え可能なフラッシュメモリと拡張された8ビットRISC型CPUの組み合わせによるATtiny2313は、多くの組み込み制御の応用に対して高度な柔軟性と対費用効果をもたらす強力なマイクロコントローラです。

ATtiny2313 AVRはCコンパイラ、マクロアセンブラ、デバッグ、シミュレータ、インサーキットエミュレータ、評価キットを含む完全なプログラム及びシステム開発ツールで支援されます。

ピン概要

VCC

電源ピン。

GND

接地ピン。

PA2~PA0 (ポートA)

ポートAは(ビット単位で選ばれる)内蔵プルアップ抵抗付きの3ビット双方向入出力ポートです。ポートA出力緩衝部は共に高い吐き出し/吸い込み能力の対称駆動特性です。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートAピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが走行していなくても、ポートAピンはHi-Zにされます。

ポートAは34頁で示されるATtiny2313の様々な特殊機能も扱います。

PB7~PB0 (ポートB)

ポートBは(ビット単位で選ばれる)内蔵プルアップ抵抗付きの8ビット双方向入出力ポートです。ポートB出力緩衝部は共に高い吐き出し/吸い込み能力の対称駆動特性です。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートBピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが走行していなくても、ポートBピンはHi-Zにされます。

ポートBは35頁で示されるATtiny2313の様々な特殊機能も扱います。

PD6~PD0 (ポートD)

ポートDは(ビット単位で選ばれる)内蔵プルアップ抵抗付きの7ビット双方向入出力ポートです。ポートD出力緩衝部は共に高い吐き出し/吸い込み能力の対称駆動特性です。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートDピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが走行していなくても、ポートDピンはHi-Zにされます。

ポートDは37頁で示されるATtiny2313の様々な特殊機能も扱います。

RESET

リセット入力。例えクロックが走行していなくても、最小パルス幅より長いこのピンのLowレベルはリセットを生成します。最小パルス幅は120頁の表80.で与えられます。より短いパルスはリセットの生成が保証されません。リセット入力はPA2及びdW(デバッグワイヤ)との切り替え機能です。

XTAL1

発振器反転増幅器への入力と内部クロック操作回路への入力。XTAL1はPA0との切り替え機能です。

XTAL2

発振器反転増幅器からの出力。XTAL2はPA1との切り替え機能です。

一般情報

資料

包括的なデータシート、応用記述、開発ツール群は<http://www.atmel.com/avr>でのダウンロードで利用可能です。

コード例

この文書はデバイスの様々な部分の使用法を手短に示す簡単なコード例を含みます。これらのコード例はアセンブルまたはコンパイルに先立ってデバイス定義ヘッダファイルがインクルードされると仮定します。全てのCコンパイラ製造業者がヘッダファイル内にビット定義を含めるとは限らず、またCでの割り込みの扱いがコンパイラに依存することに注意してください。より多くの詳細についてはCコンパイラの資料で確認してください。

データ保持力

信頼性証明結果はデータ保持誤り率の反映を示し、20年以上/85°Cまたは100年以上/25°Cで1PPMよりずっと小さな値です。

AVR CPU コア

序説

本章はAVRコア構造を一般的に説明します。このCPUコアの主な機能は正しいプログラム実行を保証することです。従ってCPUはメモリ アクセス、計算実行、周辺制御、割り込み操作ができなければなりません。

構造概要

最大効率と平行処理のため、AVRはプログラムとデータに対してメモリとバスを分離するハーバード構造を使います。プログラムメモリ内の命令は単一段のパイプラインで実行されます。1命令の実行中に次の命令がプログラムメモリから事前取得されます。この概念は全部のクロック周期で命令実行を可能にします。プログラムメモリは実装書き換え可能なフラッシュメモリです。

高速レジスタファイルは1クロック周期アクセスの32個の8ビット長汎用レジスタを含みます。これは1クロック周期ALU(Arithmetic Logic Unit)操作を許します。代表的なALU操作では2つのオペランドがレジスタファイルからの出力で、1クロック周期内でその操作が実行され、その結果がレジスタファイルに書き戻されます。

32個中の6つのレジスタは効率的なアドレス計算ができるデータ空間アドレス指定用に3つの16ビット長間接アドレスポインタ用レジスタとして使われます。これらアドレスポインタの1つはプログラム用フラッシュメモリ内の定数表参照用アドレスポインタとしても使えます。これら16ビット長付加機能レジスタはX、Y、Zレジスタで、本章内で後述されます。

ALUはレジスタ間またはレジスタと定数間の算術及び論理操作を支援します。単一レジスタ操作もALUで実行できます。算術演算操作後、操作結果についての情報を反映するために、ステータスレジスタ(SREG)が更新されます。

プログラムの流れは条件/無条件分岐や呼び出し命令によって提供され、全アドレス空間を直接アドレス指定できます。AVR命令の多くは16ビット語(ワード)形式です。全てのプログラムメモリのアドレスは(訳注:定数のみを除き)16または32ビット長命令を含みます。

割り込みやサブルーチン呼び出し中、戻りアドレスを示すプログラムカウンタ(PC)はスタックに保存されます。スタックは一般的なデータ用SRAM上に実際には割り当てられ、従ってスタック容量は全SRAM容量とSRAM使用量でのみ制限されます。全てのユーザープログラムはリセット処理ルーチンで(サブルーチン呼び出しや割り込みが実行される前に)、スタックポインタ(SP)を初期化しなければなりません。SPはI/O空間で読み書きアクセスが可能です。データ用SRAMはAVR構造で支援される5つの異なるアドレス指定種別を通して容易にアクセスできます。

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

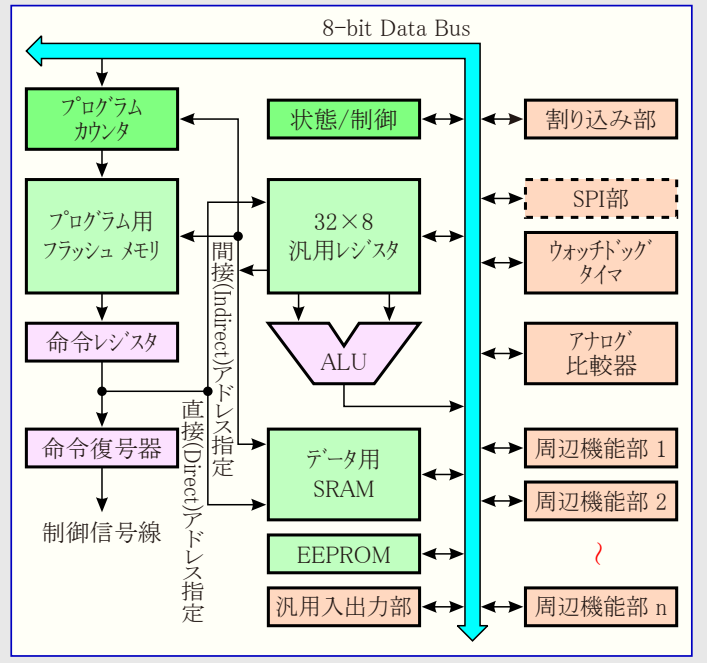
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(I)ビットがあります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタや他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリは直接またはレジスタファイルの次のデータ空間位置\$20~\$5Fとしてアクセスできます。

ALU (Arithmetic Logic Unit)

高性能なAVRのALUは32個の全汎用レジスタに直接接続され動作します。汎用レジスタ間または汎用レジスタと即値間の演算操作は単一クロック周期内で実行されます。ALU操作は算術演算、論理演算、ビット操作の3つの主要な種類に大別されます。符号付きと符号なし両方の乗算と固定小数点形式を支援する乗算器(乗算命令)も提供する構造の実装(製品)もあります。詳細記述については「命令要約」章をご覧ください。

図3. AVR構造構成図



ステータスレジスタ (Status Register) SREG

ステータスレジスタは最も直前に実行した演算命令の結果についての情報を含みます。この情報は条件処理を行うためのプログラムの流れ変更に使えます。ステータスレジスタは「命令一式手引書」で詳述したように、全てのALU操作後、更新されることに注目してください。これは多くの場合でそれ用の比較命令使用の必要をなくし、高速でより少ないコードに帰着します。

ステータスレジスタは割り込み処理ルーチン移行時の保存と割り込みからの復帰時の回復(復帰)が自動的に行われません。これはソフトウェアによって扱われなければなりません。

AVRのステータスレジスタ(SREG)は次のように定義されます。

ビット	7	6	5	4	3	2	1	0	
\$3F (\$5F)	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット7 - I: 全割り込み許可 (Global Interrupt Enable)

全割り込み許可ビットは割り込みが許可されるために設定(1)されなければなりません。その時の個別割り込み許可制御は独立した制御レジスタで行われます。全割り込み許可ビットが解除(0)されると、個別割り込み許可設定に拘らず、どの割り込みも許可されません。Iビットは割り込みが起こった後にハードウェアによって解除(0)され、後続の割り込みを許可するために、**RETI**命令によって設定(1)されます。Iビットは「命令一式手引書」で記述されるように**SEI**や**CLI**命令で応用(プログラム)によって設定(1)や解除(0)もできます。

• ビット6 - T: ビット変数 (Bit Copy Storage)

ビット複写命令、**BLD**(Bit LoaD)と**BST**(Bit STore)は操作したビットの転送元または転送先として、このTビットを使います。レジスタファイルのレジスタからのビットは**BST**命令によってTに複写でき、Tのビットは**BLD**命令によってレジスタファイルのレジスタ内のビットに複写できます。

• ビット5 - H: ハーフキャリーフラグ (Half Carry Flag)

ハーフキャリー(H)フラグはいくつかの算術操作でのハーフキャリーを示します。ハーフキャリーはBCD演算に有用です。詳細情報については「[命令要約](#)」記述をご覧ください。

• ビット4 - S: 符号 (Sign Bit, S= N Ex-OR V)

Sフラグは常に負(N)フラグと2の補数溢れ(V)フラグの排他的論理和です。詳細情報については「[命令要約](#)」記述をご覧ください。

• ビット3 - V: 2の補数溢れフラグ (2's Complement Overflow Flag)

2の補数溢れ(V)フラグは2の補数算術演算を支援します。詳細情報については「[命令要約](#)」記述をご覧ください。

• ビット2 - N: 負フラグ (Negative Flag)

負(N)フラグは算術及び論理操作での負の結果(MSB=1)を示します。詳細情報については「[命令要約](#)」記述をご覧ください。

• ビット1 - Z: ゼロフラグ (Zero Flag)

ゼロ(Z)フラグは算術及び論理操作でのゼロ(0)の結果を示します。詳細情報については「[命令要約](#)」記述をご覧ください。

• ビット0 - C: キャリーフラグ (Carry Flag)

キャリー(C)フラグは算術及び論理操作でのキャリー(または borrow)を示します。詳細情報については「[命令要約](#)」記述をご覧ください。

汎用レジスタ ファイル

このレジスタ ファイルはAVRの増強したRISC命令群用に最適化されています。必要な効率と柔軟性を達成するために、次の入出力機構がレジスタ ファイルによって支援されます。

- 1つの8ビット出力オペランドと1つの8ビットの結果入力
- 2つの8ビット出力オペランドと1つの8ビットの結果入力
- 2つの8ビット出力オペランドと1つの16ビットの結果入力
- 1つの16ビット出力オペランドと1つの16ビットの結果入力

図4.はCPU内の32個の汎用作業レジスタの構造を示します。

レジスタ ファイルを操作する殆どの命令は全てのレジスタに直接アクセスし、それらの殆どは単一周期命令です。

図4.で示されるように各レジスタは使用者データ空間の最初の32位置へ直接的に配置することで、それらはデータ メモリ アドレスも割り当てられます。例えば物理的にSRAM位置として実装されていなくてもX,Y,Zレジスタ(ポインタ)がレジスタ ファイル内のどのレジスタの指示にも設定できるように、このメモリ構成は非常に柔軟なレジスタのアクセスを提供します。

図4. AVR CPU 汎用レジスタ構成図

汎用レジスタファイル	7	0	アドレス	
R0			\$00	
R1			\$01	
R2			\$02	
⋮				
R13			\$0D	
R14			\$0E	
R15			\$0F	
R16			\$10	
R17			\$11	
⋮				
R26			\$1A	Xレジスタ
R27			\$1B	
R28			\$1C	Yレジスタ
R29			\$1D	
R30			\$1E	Zレジスタ
R31			\$1F	

Xレジスタ, Yレジスタ, Zレジスタ

R26～R31レジスタには通常用途の使用にいくつかの追加機能があります。これらのレジスタはデータ空間の間接アドレス指定用の16ビットアドレスポインタです。3つのX,Y,Z間接アドレスレジスタは図5.で記載したように定義されます。

種々のアドレス指定種別で、これらのアドレスレジスタは固定変位、自動増加、自動減少としての機能を持ちます(詳細については「命令一式手引書」をご覧ください)。

図5. X,Y,Zレジスタ構成図

	15	XH (上位)		XL (下位)	0
X レジスタ	7	R27 (\$1B)	0	7	R26 (\$1A)
	15	YH (上位)		YL (下位)	0
Y レジスタ	7	R29 (\$1D)	0	7	R28 (\$1C)
	15	ZH (上位)		ZL (下位)	0
Z レジスタ	7	R31 (\$1F)	0	7	R30 (\$1E)

スタックポインタ (Stack Pointer) SPL (SP)

スタックは主に一時データの保存、局所変数の保存、割り込みとサブルーチン呼び出し後の戻りアドレスの保存に使われます。スタックポインタレジスタは常にこのスタックの先頭(訳注:次に使われるべき位置)を指し示します。スタックが高位メモリから低位メモリへ伸長するように実行されることに注意してください。これはスタックへのPUSH命令はスタックポインタを減少するという意味です。

スタックポインタはサブルーチンや割り込みのスタックが配置されるデータSRAMのスタック領域を指し示します。データSRAM内のスタック空間はサブルーチン呼び出しの実行や割り込みの許可の何れにも先立ってプログラムによって定義されなければなりません。スタックポインタは\$60以上を指示するように設定されなければなりません。スタックポインタはPUSH命令でデータがスタックに格納されると-1され、サブルーチン呼び出しや割り込みで戻りアドレスがスタックに格納されると-2されます。スタックポインタはPOP命令でデータがスタックから引き出されると+1され、サブルーチンからの復帰(RET)命令や割り込みからの復帰(RETI)命令でアドレスがスタックから引き出されると+2されます。

AVRのスタックポインタはI/O空間内の2つの8ビットレジスタとして実装されます。実際に使われるビット数は(そのデバイス)実装に依存します。SPLだけが必要とされる程に小さいAVR構造の実装(デバイス)のデータ空間もあることに注意してください。その場合、SPHレジスタは存在しません。

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	-	-	-	-	-	-	-	-	SPH
Read/Write	R	R	R	R	R	R	R	R	
初期値	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	
ビット	7	6	5	4	3	2	1	0	
\$3D (\$5D)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	RAMEND	

命令実行タイミング

本項は命令実行の一般的なアクセスタイミングの概念を記述します。AVR CPUはチップ(デバイス)用に選んだクロック元から直接的に生成したCPUクロック(clk_{CPU})によって駆動されます。内部クロック分周は使われません。

図6.はハード構造と高速アクセスレジスタファイルの概念によって可能とされる並列の命令取得と命令実行を示します。これは機能対費用、機能対クロック、機能対電源部についての好結果と対応するMHzあたり1 MIPSを達成するための基本的なパイプラインの概念です。

図7.はレジスタファイルに対する内部タイミングの概念を示します。単一クロック周期で2つのレジスタオペラントを使うALU操作が実行され、その結果が転送先レジスタへ書き戻されます。

図6. 命令の取得と実行の並列動作

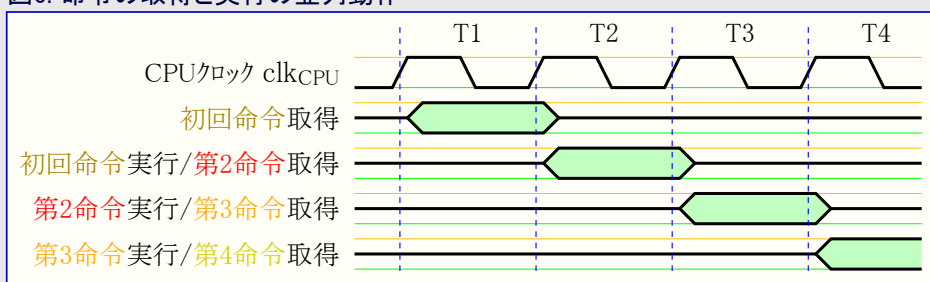
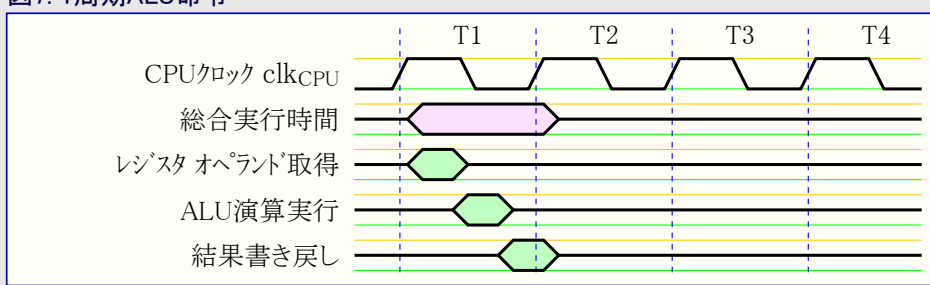


図7. 1周期ALU命令



リセットと割り込みの扱い

AVRは多くの異なる割り込み元を提供します。これらの割り込みと独立したリセットベクタ各々はプログラムメモリ空間内に独立したプログラムベクタを持ちます。全ての割り込みは割り込みを許可するため、ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない個別の許可ビットを割り当てられます。

プログラムメモリ空間の最下位アドレスは既定でリセットと割り込みベクタとして定義されます。ベクタの完全な一覧は29頁の「割り込み」で示されます。この一覧は各種割り込みの優先順位も決めます。下位側アドレスがより高い優先順位です。リセットが最高優先順位で、次が外部割り込み要求0(INT0)です。より多くの情報については29頁の「割り込み」を参照してください。

割り込みが起こると全割り込み許可(I)ビットが解除(0)され、全ての割り込みは禁止されます。使用者ソフトウェアは多重割り込みを許可するため、全割り込み許可(I)ビットへ論理1を書けます。その後全ての許可した割り込みが現在の割り込みルーチンで割り込めます。全割り込み許可(I)ビットは割り込みからの復帰(RETI)命令が実行されると、自動的に設定(1)されます。

根本的に2つの割り込み形式があります。1つ目の形式は割り込み要求フラグを設定(1)する事象によって起動されます。これらの割り込みでは割り込み処理ルーチンを実行するために、プログラムカウンタは対応する現実の割り込みベクタを指示し、ハードウェアが対応する割り込み要求フラグを解除(0)します。割り込み要求フラグは解除(0)されるべきフラグのビット位置へ論理1を書くことによっても解除(0)できます。対応する割り込み許可ビットが解除(0)されている間に割り込み条件が起こると、割り込み要求フラグが設定(1)され、割り込みが許可されるか、またはこのフラグがソフトウェアによって解除(0)されるまで記憶(保持)されます。同様に、全割り込み許可(I)ビットが解除(0)されている間に1つまたはより多くの割り込み条件が起こると、対応する割り込み要求フラグが設定(1)されて全割り込み許可(I)ビットが設定(1)されるまで記憶され、その(I=1)後で優先順に従って実行されます。

2つ目の割り込み形式は割り込み条件が存在する限り起動し(続)けます。これらの割り込みは必ずしも割り込み要求フラグを持っているとは限りません。割り込みが許可される前に割り込み条件が消滅すると、この割り込みは起動されません。

AVRが割り込みから抜け出すと常に主プログラムへ戻り、何れかの保留割り込みが扱われる前に1つ以上の命令を実行します。

ステータスレジスタ(SREG)は割り込みルーチンへ移行時の保存も、復帰時の再設定も自動的に行われなことに注意してください。これはソフトウェアによって扱われなければなりません。

割り込みを禁止するためにCLI命令を使うと、割り込みは直ちに禁止されます。CLI命令と同時に割り込みが起こっても、CLI命令後に割り込みは実行されません。次例は時間制限EEPROM書き込み手順中に割り込みを無効とするために、これがどう使えるかを示します。

アセンブリ言語プログラム例

```
IN      R16, SREG      ;ステータスレジスタを保存
CLI     ;EEPROM書き込み手順中割り込み禁止
SBI     EECR, EEMPE    ;EEPROM主書き込み許可
SBI     EECR, EEPE     ;EEPROM書き込み開始
OUT     SREG, R16     ;ステータスレジスタを復帰
```

C言語プログラム例

```
char cSREG;           /* ステータスレジスタ保存変数定義 */
cSREG = SREG;        /* ステータスレジスタを保存 */
__disable_interrupt(); /* EEPROM書き込み手順中割り込み禁止 */
EECR |= (1<<EEMPE); /* EEPROM主書き込み許可 */
EECR |= (1<<EEPE);  /* EEPROM書き込み開始 */
SREG = cSREG;       /* ステータスレジスタを復帰 */
```

割り込みを許可するためにSEI命令を使うと、次例で示されるようにどの保留割り込みにも先立ってSEI命令の次の命令が実行されます。

アセンブリ言語プログラム例

```
SEI     ;全割り込み許可
SLEEP  ;休止形態移行(割り込み待ち)
```

C言語プログラム例

```
__enable_interrupt(); /* 全割り込み許可 */
__sleep();           /* 休止形態移行(割り込み待ち) */
```

注: SLEEP命令までは割り込み禁止、保留割り込み実行前に休止形態へ移行します。

割り込み応答時間

許可した全てのAVR割り込みに対する割り込み実行応答は最小4クロック周期です。4クロック周期後、実際の割り込み処理ルーチンに対するプログラムベクタアドレスが実行されます。この4クロック周期時間中にプログラムカウンタ(PC)がスタック上に保存(プッシュ)されます。このベクタは標準的に割り込み処理ルーチンへの無条件分岐で、この分岐は2クロック周期(訳注:原文は3(JMP命令=3を想定、実際はRJMP命令=2))要します。複数周期命令実行中に割り込みが起こると、その割り込みが扱われる前に、この命令が完了されます。MCUが休止形態の時に割り込みが起こると、割り込み実行応答時間は4クロック周期増やされます。この増加は選んだ休止形態からの起動時間に加えてです。

割り込み処理ルーチンからの復帰は4クロック周期要します。これらの4クロック周期中、プログラムカウンタ(PC:2バイト)がスタックから取り戻され(ポップ)、スタックポインタは増加され(+2)、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されます。

AVR ATtiny2313のメモリ

この項はATtiny2313の各種メモリを記述します。AVR構造にはプログラムメモリ空間とデータメモリ空間の2つの主要なメモリ空間があります。加えてATtiny2313はデータ保存用EEPROMメモリが特徴です。3つのメモリ空間全ては一般的な直線的アドレスです。

実装書き換え可能なプログラム用フラッシュメモリ

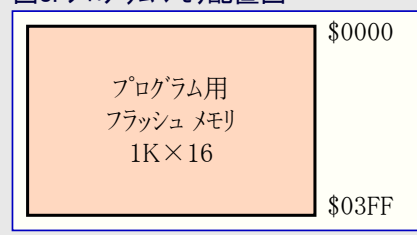
ATtiny2313はプログラム保存用に実装書き換え可能な2Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、このフラッシュメモリは1K×16ビットとして構成されます。

フラッシュメモリは最低10,000回の消去/書き込み回数の耐久性があります。ATtiny2313のプログラムカウンタ(PC)は10ビット幅、故に1Kプログラムメモリ位置のアドレス指定です。106頁の「[メモリプログラミング](#)」はSPIピンを使うフラッシュメモリの直列プログラミングの詳細な記述を含みます。

定数表は全てのプログラムメモリアドレス空間に配置できます。(LPM命令記述参照)

命令の取得と実行のタイミング図は9頁の「[命令実行タイミング](#)」で示されます。

図8. プログラムメモリ配置図



データ用SRAMメモリ

図9はATtiny2313のSRAMメモリ構成方法を示します。

下位224データメモリ位置はレジスタファイル、I/Oメモリ、内蔵データSRAMに充てます。最初の32位置はレジスタファイル、次の64位置は標準I/Oメモリに充て、最後の128位置は内蔵データSRAMに充てます。

直接、間接、変位付き間接、事前減少付き間接、事後増加付き間接の5つの異なるアドレス指定種別でデータメモリ(空間)を網羅します。レジスタファイル内のレジスタR26～R31は間接アドレス指定ポインタ用レジスタが特徴です。

直接アドレス指定はデータ空間全体に届きます。

変位付き間接動作はYまたはZレジスタで与えられる基準アドレスからの63アドレス位置に届きます。

自動の事前減少付きと事後増加付きのレジスタ間接アドレス指定動作を使う時に(使われる)X, Y, Zアドレスレジスタは減少(-1)または増加(+1)されます。

ATtiny2313の32個の汎用レジスタ、64個のI/Oレジスタ、128バイトのデータ用内蔵SRAMはこれら全てのアドレス指定種別を通して全部アクセスできます。レジスタファイルは8頁の「[汎用レジスタファイル](#)」で記述されます。

図9. データメモリ配置図

		アドレス
レジスタ ファイル (32)	R0	\$0000
	R1	\$0001
	⋮	
	R30	\$001E
	R31	\$001F
I/O レジスタ (64)	\$00	\$0020
	\$01	\$0021
	⋮	
	\$3E	\$005E
内蔵 SRAM (128×8)	\$3F	\$005F
	\$0060	\$0060
	\$0061	\$0061
	⋮	
	\$00DE	\$00DE
	\$00DF	\$00DF

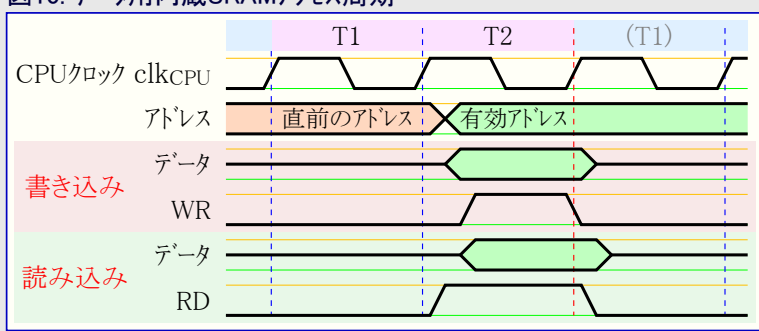
注: 赤字はI/Oアドレス

データメモリアクセスタイミング

この項は内部メモリアクセスに対する一般的なアクセスタイミングの概念を記述します。データ用内蔵SRAMアクセスは図10で記載されるように2 clkCPU周期で実行されます。

(訳注) 内蔵SRAMのアクセスを含む代表的な命令はT1, T2の2周期で実行され、T1で対象アドレスを取得/(算出)/確定し、T2で実際のアクセスが行われます。後続する(T1)は次の命令のT1です。

図10. データ用内蔵SRAMアクセス周期



データ用EEPROMメモリ

ATtiny2313は128バイトのデータ用EEPROMを含みます。それは単一バイトが読み書きできる分離したデータ空間として構成されます。EEPROMは最低100,000回の消去/書き込み回数の耐久性があります。CPUとEEPROM間のアクセスは以降の[EEPROMアドレスレジスタ](#)、[EEPROMデータレジスタ](#)、[EEPROM制御レジスタ](#)で詳細に記述されます。EEPROMの直列プログラミングの詳細な記述については115頁をご覧ください。

EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROMの消去/書き込み(訳注:原文はアクセス)時間は表1.で与えられます。(書き込みは)自己タイミング機能ですが、使用者ソフトウェアは**次バイトが書ける時を検知**してください。使用者コードがEEPROMに書く命令を含む場合、いくつかの予防処置が取られなければなりません。嚴重に濾波した電源では電源投入/切断でVCCが緩やかに上昇または下降しそうです。これはデバイスが何周期かの時間、使われるクロック周波数に於いて最小として示されるより低い電圧で走行する原因になります。これらの状態で問題を避ける方法の詳細については14頁の「[EEPROMデータ化けの防止](#)」をご覧ください。

予期せぬEEPROM書き込みを防止するため特別な書き込み手順に従わなければなりません。この詳細についてはEEPROM制御レジスタの説明と13頁の「[非分離バイト書き込み](#)」と「[分離バイト書き込み](#)」を参照してください(訳注:本行内容追加)。

EEPROMが読まれると、CPUは次の命令が実行される前に4クロック周期停止されます。EEPROMが書かれると、CPUは次の命令が実行される前に2クロック周期停止されます。

EEPROMアドレスレジスタ (EEPROM Address Register) EEAR

ビット	7	6	5	4	3	2	1	0	
\$1E (\$3E)	-	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEAR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	不定	不定	不定	不定	不定	不定	不定	

- **ビット7 - Res : 予約 (Reserved)**

このビットは予約されており、常に0として読みます。

- **ビット6~0 - EEAR6~0 : EEPROMアドレス (EEPROM Address)**

EEPROMアドレスレジスタ(EEAR)は128バイトEEPROM空間のEEPROMアドレスを指定します。EEPROMデータバイトは0~127間で直線的に配されます。EEARの初期値は不定です。EEPROMがアクセスされるであろう前に適切な値が書かれなければなりません。

EEPROMデータレジスタ (EEPROM Data Register) EEDR

ビット	7	6	5	4	3	2	1	0	
\$1D (\$3D)	(MSB)							(LSB)	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

- **ビット7~0 - EEDR7~0 : EEPROMデータ (EEPROM Data)**

EEPROM書き込み操作に対してEEDRはEEPROMアドレスレジスタ(EEAR)で与えたアドレスのEEPROMへ書かれるべきデータを含みます。EEPROM読み込み操作に対してEEDRはEEARで与えたアドレスのEEPROMから読み出したデータを含みます。

EEPROM制御レジスタ (EEPROM Control Register) EECR

ビット	7	6	5	4	3	2	1	0	
\$1C (\$3C)	-	-	EEPM1	EEPM0	EERIE	EEMPE	EEPE	EERE	EECR
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	不定	0	0	不定	0	

- **ビット7,6 - Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

•ビット5,4 – EEPM1,0 : EEPROMプログラミング種別 (EEPROM Programming Mode Bits)

EEPROMプログラミング種別ビット設定はEEPROMプログラミング許可(EEPE)書き込み時にどのプログラミング動作が起動されるかを定義します。1つの非分離操作(旧値消去と新値書き込み)、または2つの異なる操作として消去と書き込み操作を分離してデータをプログラムする(書く)ことが可能です。各動作に対するプログラミング時間は表1.で示されます。EEPEが設定(1)されている間はEEP Mnへのどの書き込みも無視されます。リセット中、EEP MnビットはEEPROMがプログラミング作業中を除いて'00'にリセットされます。

表1. EEPROMプログラミング種別

EEP M1	EEP M0	プログラミング時間	動作
0	0	3.4ms	1操作での消去と書き込み(非分離操作)
0	1	1.8ms	消去のみ
1	0	1.8ms	書き込みのみ
1	1	-	将来使用に予約

•ビット3 – EERIE : EEPROM操作可割り込み許可 (EEPROM Ready Interrupt Enable)

EERIEの1書き込みはステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、EEPROM操作可割り込みを許可します。EERIEの0書き込みは、この割り込みを禁止します。EEPROM操作可割り込みは不揮発性メモリ(フラッシュメモリとEEPROM)がプログラミングの準備可ならば継続する割り込みを発生します。

•ビット2 – EEMPE : EEPROM主プログラム許可 (EEPROM Master Program Enable)

EEMPEビットはEEPROMプログラム許可(EEPE)ビットの1書き込みが有効か無効かどちらかを決めます。

EEMPEが設定(1)されると、4クロック周期内のEEPE設定(1)は選んだアドレスのEEPROMをプログラムします。EEMPEが0なら、EEPE設定(1)は無効です。EEMPEがソフトウェアによって設定(1)されてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。

•ビット1 – EEPE : EEPROMプログラム許可 (EEPROM Program Enable)

EEPROMプログラム許可信号(EEPE)はEEPROMへのプログラミング許可信号です。EEPEが(1)を書かれると、EEPROMはEEP Mnビット設定に従ってプログラムされます。論理1がEEPEへ書かれる前にEEPROM主プログラム許可(EEMPE)ビットは1を書かれなければならない、さもなければEEPROM書き込み(消去)は行われません。書き込み(プログラミング)アクセス時間が経過されると、EEPROMプログラム許可(EEPE)ビットはハードウェアによって解除(0)されます。EEPEが設定(1)されてしまうと、次の命令が実行される前にCPUは2周期停止されます。

•ビット0 – EERE : EEPROM読み込み許可 (EEPROM Read Enable)

EEPROM読み込み許可信号(EERE)はEEPROMへの読み込みストローブです。EERに正しいアドレスが設定されると、EEPROM読み出しを起動するためにEEREビットは1を書かれなければなりません。EEPROM読み出しアクセスは(その)1命令で行われ、要求したデータは直ちに利用できます。EEPROMが読まれるとき、次の命令が実行される前にCPUは4周期停止されます。使用者は読み込み操作を始める前にEEPEビットをポーリングすべきです。書き込み(プログラム)操作実行中の場合、EEPROMアドレスレジスタ(EEAR)の変更もEEPROM読み込みもできません。

非分離バイトプログラミング

非分離バイトプログラミングの使用は最も簡単な動作です。EEPROMにバイトを書くとき、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEP Mnビットが'00'ならば、(EEMPEが1を書かれる後の4周期内の)EEPEの1書き込みは消去/書き込み動作を起動します。消去と書き込みの両周期は1操作で行われ、総プログラミング時間は表1.で与えられます。EEPEビットは消去と書き込み動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

分離バイトプログラミング

2つの異なる操作として消去と書き込み周期を分離することが可能です。これは或る時間制限(代表的には電源電圧不足)に対してシステムが短いアクセス時間を必要とする場合に有用かもしれません。この方法の優位性を得るため、書かれるべき位置が書き込み操作前に消去されてしまっていることが必要とされます。しかし、消去と書き込みが分離されるため、時間が重大な操作の実行をシステムが許す時(代表的には電源投入後)に消去操作を行うことが可能です。

消去

バイトを消去するにはアドレスがEEARに書かれなければなりません。EEP Mnビットが'01'なら、(EEMPEが1を書かれた後の4周期内の)EEPEの1書き込みは消去動作だけを起動します(プログラミング時間は表1.で与えられます)。EEPEビットは消去動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

書き込み

(特定)位置を書くため、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEP Mnビットが'10'なら、(EEMPEが1を書かれた後の4周期内の)EEPEの1書き込みは書き込み動作だけを起動します(プログラミング時間は表1.で与えられます)。EEPEビットは書き込み動作が完了されるまで設定(1)に留まります。書かれるべき位置が書き込み前に消去されてしまっていなければ、元の格納データは失ったと見做されなければなりません。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

EEPROMアクセスの時間に校正済み内蔵RC発振器が使われます。発振器周波数が18頁の「発振校正レジスタ – OSCCAL」で記述した必要条件内であることを確かめてください。

次のコード例はアセンブリ言語とC言語でのEEPROM消去、書き込み、または非分離書き込み関数を示します。本例は(例えば全割り込み禁止によって)割り込みが制御され、これらの関数実行中に割り込みが起きない前提です。

アセンブリ言語プログラム例

```
EEPROM_WR:  SBIC    EECR, EEPE                ;EEPROMプログラミング完了ならばスキップ
            RJMP    EEPROM_WR              ;以前のEEPROMプログラミング完了まで待機
;
            LDI    R18, (0<<EEPROM1) | (0<<EEPROM0) ;プログラミング種別値取得(本例は非分離)
            OUT    EECR, R18                ;対応プログラミング種別設定
            OUT    EEAR, R17                ;EEPROMアドレス設定
            OUT    EEDR, R16                ;EEPROM書き込み値を設定
            SBI    EECR, EEMPE              ;EEPROM主プログラム許可ビット設定
            SBI    EECR, EEPE              ;EEPROMプログラミング開始(プログラム許可ビット設定)
            RET                               ;呼び出し元へ復帰
```

C言語プログラム例

```
void EEPROM_write(unsigned char ucAddress, unsigned char ucData)
{
    while(EECR & (1<<EEPE));                /* 以前のEEPROMプログラミング完了まで待機 */
    EECR = (0<<EEPROM1) | (0<<EEPROM0);      /* 対応プログラミング種別設定 */
    EEAR = ucAddress;                        /* EEPROMアドレス設定 */
    EEDR = ucData;                          /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMPE);                      /* EEPROM主プログラム許可 */
    EECR |= (1<<EEPE);                      /* EEPROMプログラミング開始 */
}
```

次のコード例はアセンブリ言語とC言語でのEEPROM読み込み関数を示します。本例は割り込みが制御され、これらの関数実行中に割り込みが起きない前提です。

アセンブリ言語プログラム例

```
EEPROM_RD:  SBIC    EECR, EEPE                ;EEPROMプログラミング完了ならばスキップ
            RJMP    EEPROM_RD              ;以前のEEPROMプログラミング完了まで待機
;
            OUT    EEAR, R17                ;EEPROMアドレス設定
            SBI    EECR, EERE              ;EEPROM読み出し開始(読み込み許可ビット設定)
            IN     R16, EEDR                ;EEPROM読み出し値を取得
            RET                               ;呼び出し元へ復帰
```

C言語プログラム例

```
unsigned char EEPROM_read(unsigned char ucAddress)
{
    while(EECR & (1<<EEPE));                /* 以前のEEPROMプログラミング完了まで待機 */
    EEAR = ucAddress;                        /* EEPROMアドレス設定 */
    EECR |= (1<<EERE);                      /* EEPROM読み出し開始 */
    return EEDR;                            /* EEPROM読み出し値を取得, 復帰 */
}
```

EEPROMデータ化けの防止

低VCCの期間中、正しく動作するための供給電圧がCPUとEEPROMに対して低すぎるためにEEPROMデータが化け得ます。これらの問題はEEPROMを使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

EEPROMデータ化けは電圧が低すぎる時の2つの状態によって起こされ得ます。1つ目として、EEPROMへの通常の書き込み手順は正しく動作するための最低電圧が必要です。2つ目として、供給電圧が低すぎると、CPU自身が命令を間違って実行し得ます。

EEPROMデータ化けは次の推奨設計によって容易に避けられます。

不十分な供給電源電圧の期間中、AVRのRESETを活性(Low)に保ってください。これは内蔵低電圧検出器(BOD)を許可することによって行えます。内蔵BODの検出電圧が必要とした検出電圧と一致しない場合、外部低VCCリセット保護回路が使えます。書き込み動作実行中にリセットが起こると、この書き込み操作は供給電源電圧が充分ならば(継続)完了されます。

I/O メモリ (レジスタ)

ATtiny2313のI/O空間定義は141頁の「レジスタ要約」で示されます。

ATtiny2313の全てのI/Oと周辺機能はI/O空間に配置されます。全てのI/O位置はI/O空間と32個の汎用作業レジスタ間のデータ転送を行うLD/LDS/LDD命令とST/STS/STD命令によってアクセスされます。アドレス範囲\$00～\$1F内のI/OレジスタはSBI命令とCBI命令の使用で直接的にビットアクセス可能です。これらのレジスタではSBISとSBIC命令の使用によって単一ビット値が検査できます。より多くの詳細については「命令要約」章を参照してください。I/O指定命令INとOUTを使う時はI/Oアドレス\$00～\$3Fが使われなければなりません。LD命令とST命令を使い、データ空間としてI/Oレジスタをアクセスする時はこれらのアドレスに\$20が加算されなければなりません。

将来のデバイスとの共通性を保つため、アクセスされる場合、予約ビットは0が書かれるべきです。予約済みI/Oメモリアドレスは決して書かれるべきではありません。

状態フラグのいくつかはそれらへ論理1を書くことによって解除(0)されます。CBIとSBI命令は他の多くのAVRの様ではなく、指定ビットだけを操作し、従って状態フラグのようなものを含むレジスタに使えることに注意してください。CBIとSBI命令は(I/Oアドレス)\$00～\$1Fのレジスタでのみ動作します。

I/Oと周辺制御レジスタは以降の項で説明されます。

汎用I/Oレジスタ

ATtiny2313は3つの汎用I/Oレジスタを含みます。これらのレジスタはどの情報の格納にも使え、特に全体変数や状態フラグの格納に有用です。(I/O)アドレス範囲\$00～\$1Fの汎用I/OレジスタはSBI,CBI,SBIS,SBIC命令の使用で直接ビットアクセスが可能です。

汎用I/Oレジスタ2 (General Purpose I/O Register 2) GPIOR2

ビット	7	6	5	4	3	2	1	0	
\$15 (\$35)	(MSB)							(LSB)	GPIOR2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

汎用I/Oレジスタ1 (General Purpose I/O Register 1) GPIOR1

ビット	7	6	5	4	3	2	1	0	
\$14 (\$34)	(MSB)							(LSB)	GPIOR1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

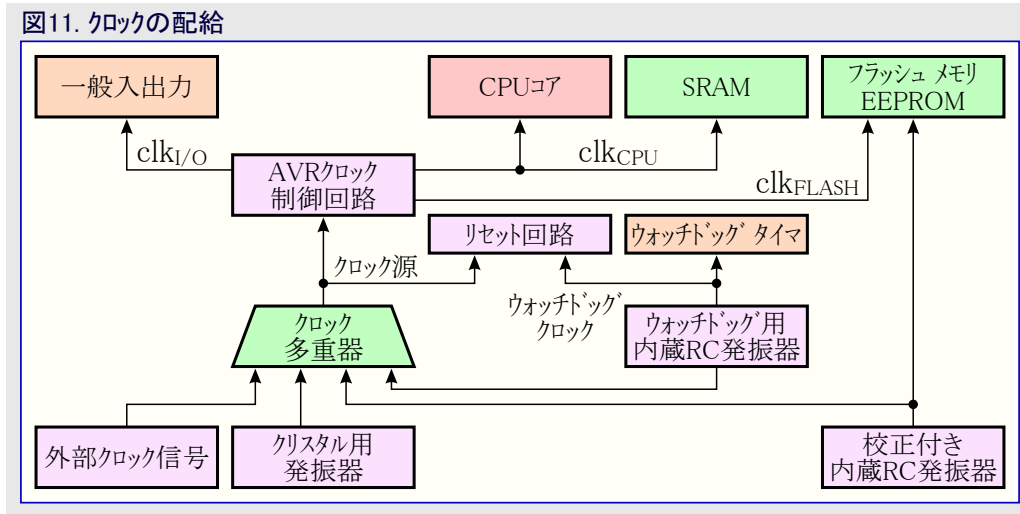
汎用I/Oレジスタ0 (General Purpose I/O Register 0) GPIOR0

ビット	7	6	5	4	3	2	1	0	
\$13 (\$33)	(MSB)							(LSB)	GPIOR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

システム クロックとクロック選択

クロック系統とその配給

図11.はAVR内の主要なクロック系統とその配給を示します。全てのクロックが与えられた時間有効である必要はありません。消費電力低減のため、21頁の「電力管理と休止形態」で記述される各種休止形態の使用によって、使われていない部分のクロックが停止できます。クロック系統は以下で詳述されます。



CPU クロック - clk_{CPU}

CPUクロックはAVRコアの動作と関係する系統の部分に配給されます。このような部分の例は汎用レジスタファイル、ステータスレジスタ、スタックポインタを保持するデータメモリです。CPUクロックの停止はコアが一般的な操作や計算を実行することを禁止します。

I/O クロック - clk_{I/O}

I/Oクロックはタイマ/カウンタやUSARTのようなI/O部の大部分で使われます。I/Oクロックは外部割り込み部でも使われますが、いくつかの外部割り込みは例えばI/Oクロックが停止されても検出されることをこのような割り込みに許す非同期論理回路によって検出されることに注意してください。多用途直列(USD)部の開始条件検出はclk_{I/O}が停止される時に非同期に実行され、全休止形態でUSI開始条件検出を可能とすることにも注意してください。

フラッシュ クロック - clk_{FLASH}

フラッシュクロックはフラッシュメモリインターフェースの動作を制御します。このフラッシュクロックは常にCPUクロックと同時に活動します。

クロック元

このデバイスには右で示されるようにフラッシュヒューズビットによって選択可能な後続のクロック元選択があります。選んだクロック元からのクロックはAVRクロック発生器への入力で、適切な単位部へ配給されます。

各クロック選択に対する様々な選択は次項で得られます。CPUがパワーダウンから起動するとき、選んだクロック元は命令実行開始前に安定な発振器動作を保証する起動時間に使われます。CPUがリセットから始まる時、これらは通常動作開始前に安定電圧へ達するのを電源に許す付加遅延です。ウォッチドッグ発振器は、この起動時間の実時間部のタイミングに使われます。各計時終了に使われるウォッチドッグ(WDT)発振器の各周期数は表3.で示されます。121頁の「代表性」で示されるようにウォッチドッグ発振器の周波数は電圧に依存します。

表2. クロック種別選択

クロック種別	CKSEL3~0
外部クリスタル/セラミック発振器	1111~1000
128kHz内部(WDT)発振器	0110
8MHz校正付き内蔵RC発振器	0100
4MHz校正付き内蔵RC発振器	0010
外部クロック信号	0000
(予約)	0xx1

注: 1=非プログラム、0=プログラム、x=0または1

表3. WDT発振器の代表的計時終了値、周期数

VCC=3.0V	VCC=5.0V	周期数
4.3ms	4.1ms	512
69ms	65ms	8K (8192)

既定のクロック元

このデバイスはCKSEL=0100, SUT=10, CKDIV8=プログラム(0)で出荷されます。従って既定クロック元設定は最長起動時間の8MHzで走行する内蔵RC発振器を8で前置分周し、結果として1MHzに帰着する初期システムクロックです。この既定設定は全ての使用者が実装または並列書き込み器を使って、それらを希望したクロック元設定にできることを保証します。

クリスタル発振器

XTAL1とXTAL2は図12.で示されるように、チップ上の発振器としての使用に設定できる反転増幅器の各々、入力と出力です。クリスタル発振子またはセラミック振動子のどちらでも使えます。

C1とC2はクリスタル発振子とセラミック振動子の両方について常に等しくすべきです。このコンデンサの最適値は使うクリスタル発振子やセラミック振動子、浮遊容量の量、その環境の電磁雑音に依存します。クリスタル発振子使用に対するコンデンサ選択について初期の指針のいくつかは表4.で与えられます。セラミック振動子については製造業者によって与えられたコンデンサ値が使われるべきです。

発振器は示された周波数範囲で最適化された4つの異なる動作ができます。この動作は、表4.で示されたCKSEL3~1ヒューズによって選ばれます。

図12. クリスタル発振子接続図

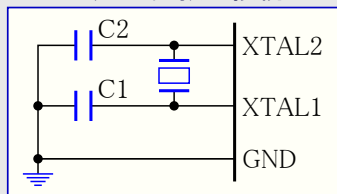


表4. クリスタル用発振器動作

CKSEL3~1	周波数範囲	推奨C1,2容量
100 (注)	0.4~0.9MHz	-
101	0.9~3.0MHz	12~22pF
110	3.0~8.0MHz	12~22pF
111	8.0~ MHz	12~22pF

注: この選択はクリスタル発振子ではなく、セラミック振動子でのみ使われるべきです。

CKSEL0ヒューズはSUT1,0ヒューズと共に、表5.で示される起動遅延時間を選びます。

表5. クリスタル発振子/セラミック振動子用起動遅延時間選択表

CKSEL0	SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間 (VCC=5.0V)	推奨使用法
0	0 0	258×CK (注1)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
	0 1	258×CK (注1)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	1 0	1K×CK (注2)	14×CK	外部セラミック振動子、低電圧検出(BOD)リセット許可
	1 1	1K×CK (注2)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
1	0 0	1K×CK (注2)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	0 1	16K×CK	14×CK	外部クリスタル発振子、低電圧検出(BOD)リセット許可
	1 0	16K×CK	14×CK+4.1ms	外部クリスタル発振子、高速上昇電源
	1 1	16K×CK	14×CK+65ms	外部クリスタル発振子、低速上昇電源

注1: これらの選択はデバイスの最高周波数付近での動作でないとき、応用にとって起動での周波数安定性が重要でない場合だけ使われるべきです。これらの選択はクリスタル発振子用ではありません。

注2: これらの選択はセラミック振動子での使用を意図され、起動での周波数安定性を保証します。デバイスの最高周波数付近での動作でないとき、応用にとって起動での周波数安定性が重要でない場合はクリスタル発振子も使えます。

校正付き内蔵RC発振器 (訳注:共通性のため内容を一部追加/修正)

校正された内蔵RC発振器は8MHzまたは4MHzクロックを供給します。この周波数は3V,25°Cでの公称値です。この周波数が(VCCに依存して)デバイスの仕様を越える場合、起動中に内部周波数を8分周するためにCKDIV8ヒューズがプログラム(0)されなければなりません。より多くの詳細については20頁の「システムクロック前置分周器」をご覧ください。このクロックは表6.で示されるようにCKSELヒューズのプログラミングによってシステムクロックとして選べます。選んだなら、外部部品なしで動作します。リセット中、ハードウェアが発振校正レジスタ(OSCCAL)に校正値バイトを設定し、これによってRC発振器を自動的に校正します。3V,25°Cでこの校正は公称周波数±10%以内の周波数を与えます。atmel.com/avrで利用可能な応用記述に記載された実行時校正法の使用で、与えられたどのVCCと温度でも±2%の精度を達成することができます。この発振器がチップ(システム)クロックとして使われるとき、ウォッチドッグ発振器は未だウォッチドッグタイマとリセット付加遅延タイマに使われます。予め設定された校正値のより多くの情報については、107頁の「校正バイト」項をご覧ください。このデバイスはCKDIV8ヒューズがプログラム(0)で出荷されます。

この発振器が選択されると、起動時間は表7.で示されるようにSUTヒューズによって決定されます。

表6. 校正付き内蔵RC発振器動作

CKSEL3~0	公称周波数 (MHz)
0 1 0 0	8.0 (注1)
0 0 1 0	4.0

注1: デバイスはこの選択で出荷されます。

表7. 校正付き内蔵RC発振器用起動遅延時間選択表

SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間	推奨使用法
0 0	6×CK	14×CK (注1)	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4.1ms	高速上昇電源
1 0 (注2)	6×CK	14×CK+65ms	低速上昇電源
1 1	(予約)		

注1: RSTDISBLヒューズがプログラム(0)されると、プログラミング動作への移行可を保証するため、付加遅延時間は14×CK+4msに増やされます。

注2: デバイスはこの選択で出荷されます。

発振校正レジスタ (Oscillator Calibration Register) OSCCAL

ビット	7	6	5	4	3	2	1	0	
\$31 (\$51)	-	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	デバイス固有の校正値							

• ビット7 - Res : 予約 (Reserved) (訳注:共通性のため追加)

このビットは予約されており、常に0として読みます。

• ビット6~0 - CAL6~0 : 発振校正値 (Oscillator Calibration Value)

このアドレスへの校正バイト書き込みは発振器周波数の偏差処理を省くために内蔵発振器を調整します。これはチップリセット中、自動的に行われます。OSCCALが0の時に最低利用可能周波数が選ばれます。このレジスタへ0以外の値を書くことは内蔵発振器の周波数を増加します。このレジスタへの\$7F書き込みは最高使用可能周波数にします。校正付き発振器はフラッシュメモリとEEPROMのアクセス時間に使われます。フラッシュメモリやEEPROMが書かれる場合、公称周波数より上へ10%を越えて校正してはいけません。そうでなければフラッシュメモリやEEPROM書き込みは失敗するかもしれません。この発振器は8MHzまたは4MHzへの校正を意図したものであることに注意してください。表8.で示されるような他の値への調整は保証されません。

表8. 内蔵RC発振器周波数範囲

OSCCAL値	公称周波数に対する割合	
	最小	最大
\$00	50 %	100 %
\$3F	75 %	150 %
\$7F	100 %	200 %

校正付き内蔵RC発振器を校正するとき、MCUの安定な動作を保証するために大きな段階での校正値変更を避けてください。或るクロック周期から次への2%より大きな周波数変化は予測されない事態を引き起こします。OSCCAL変更は各校正について\$20を越えるべきではありません。

外部クロック信号

外部クロック元からデバイス駆動するためにXTAL1は図13.で示されるように駆動されるべきです。外部クロックでデバイスを走行するためにはCKSELヒューズが'0000'にプログラム(設定)されなければなりません。

このクロック元が選ばれると、起動時間は表10.で示されるようにSUTヒューズによって決定されます。

表9. 外部クロック信号動作

CKSEL3~0	周波数範囲
0 0 0 0	0~16MHz

図13. 外部クロック信号駆動接続図

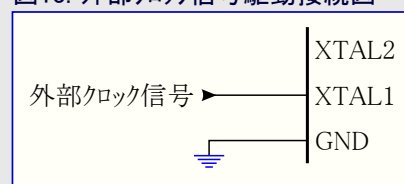


表10. 外部クロック信号駆動用起動遅延時間選択表

SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間 (VCC=5.0V)	推奨使用法
0 0	6×CK	14×CK	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4.1ms	高速上昇電源
1 0	6×CK	14×CK+65ms	低速上昇電源
1 1	(予約)		

外部クロックを供給するとき、MCUの安定な動作を保証するために供給したクロック周波数の急な変化を避けることが必要とされます。或るクロック周期から次への2%より大きな周波数変化は予測されない事態を引き起こします。このようなクロック周波数での変化中、MCUはリセットに保たれるのを保証することが必要とされます。

システムクロック前置分周器が安定な動作を保証しながら、内部クロック周波数の実行時変更の実現に使えることに注意してください。詳細については20頁の「システムクロック前置分周器」を参照してください(訳注:共通性のため本行追加)。

128kHz内部発振器

128kHz内部発振器は128kHzのクロックを供給する低電力発振器です。この周波数は3V,25°Cでの公称値です。このクロックはCKSELヒューズを'0110'にプログラミング(設定)することによってシステムクロックとして選べます。

このクロック元が選ばれると、起動時間は表11.で示されるようにSUTヒューズによって決定されます。

表11. 128kHz内部発振器用起動遅延時間選択表

SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間	推奨使用法
0 0	6×CK	14×CK (注1)	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4ms	高速上昇電源
1 0	6×CK	14×CK+64ms	低速上昇電源
1 1	(予約)		

注1: RSTDISBLヒューズがプログラム(0)されると、プログラミング動作への移行可を保証するため、付加遅延時間は14×CK+4msに増やされます。

システム クロック前置分周器

ATtiny2313のシステム クロックは**クロック前置分周レジスタ(CLKPR)**の設定によって分周できます。この特徴(機能)は必要とされる処理能力が低い時の消費電力削減に使えます。これは全クロック種別で使え、CPUと全同期周辺機能のクロック周波数に影響を及ぼします。clk_{CPU}、clk_{FLASH}、clk_{I/O}は**表12**で示された値によって分周されます。

前置分周器設定間を切り替えるとき、システム クロック前置分周器は中間(経過途中)の周波数が直前の設定に対応するクロック周波数または新規設定に対応するクロック周波数のどちらよりも高くなく、クロック系で不具合が起きないことを保証します。

前置分周器として実行するリプル カウンタは分周されないクロック周波数で走行し、CPUのクロック周波数より速いかもしれません。従って例え(カウンタ値が)読めるとしても、前置分周器の状態を決めることはできず、1から他へのクロック分周値切り替えを行う正確な時間は必ずしも予測できません。CLKPS値が書かれる時から新規クロック周波数が活性(有効)になる前にT1+T2~T1+2×T2間かかります。この間で2つの有効なクロック端が生成されます。ここでのT1は直前のクロック周期、T2は新規前置分周器設定に対応する周期です。

予期せぬクロック周波数の変更を防ぐため、CLKPSビットの変更は次の特別な書き込み手順に従わなければなりません。

1. **クロック分周値変更許可(CLKPCCE)ビット**に**1**、CLKPR内の他の全ビットに**0**を書きます。
2. (次からの)4周期以内にCLKPCCEビットを**0**とする欲したCLKPS値を書きます。

前置分周器設定変更時、書き込み手続きが割り込まれないことを保証するため、割り込みは禁止されなければなりません。

クロック前置分周レジスタ (Clock Prescale Register) CLKPR

ビット	7	6	5	4	3	2	1	0	
\$26 (\$46)	CLKPCCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	内容参照	内容参照	内容参照	内容参照	

• ビット7 – CLKPCCE : クロック分周値変更許可 (Clock Prescaler Change Enable)

CLKPSビットの変更を許可するためにCLKPCCEビットは論理**1**を書かれなければなりません。CLKPCCEビットは同時にCLKPRの他の全ビットが**0**を書かれる時だけ更新されます。CLKPCCEは書き込み後4クロック周期またはCLKPSビット書き込み時、ハードウェアによって解除(**0**)されます。この制限時間(4クロック周期)内のCLKPCCEビット再書き込みは制限時間の延長もCLKPCCEビットの解除(**0**)も行いません。

• ビット3~0 – CLKPS3~0 : クロック分周値選択 (Clock Prescaler Select Bits 3~0)

これらのビットは選んだクロック元と内部システム クロック間の分周値を定義します。これらのビットは応用の必要条件に合わせた各種クロック周波数を実行時に書けます。分周値が使われると、分周器はMCUへの主クロックを分周し、全ての同期周辺機能の速度が減じられます。分周値は**表12**で与えられます。

CKDIV8ヒューズがCLKPSビットの初期値を決めます。CKDIV8が非プログラム(**1**)にされると、CLKPSビットは'**0000**'にリセットされます。CKDIV8がプログラム(**0**)にされると、CLKPSビットは起動時に8分周を与える'**0011**'にリセットされます。現在の動作条件でデバイスの最高周波数より高い周波数のクロック元を選んだ場合、この機能が使われるべきです。CKDIV8ヒューズ設定に拘らず、どの値もCLKPSビットへ書けることに注意してください。応用ソフトウェアは現在の動作条件でデバイスの最高周波数より高い周波数のクロック元を選んだ場合、充分な分周値が選ばれることを保証しなければなりません。本デバイスはCKDIV8ヒューズがプログラム(**0**)で出荷されます。

表12. クロック前置分周器選択

CLKPS3	0								1							
	0				1				0				1			
CLKPS2	0				1				0				1			
CLKPS1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
CLKPS0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
分周値(数)	1	2	4	8	16	32	64	128	256	(予約)						

電力管理と休止形態

休止形態は応用でMCU内の未使用部を一時停止することを可能にし、それによって節電します。AVRは応用で必要な消費電力に仕立てることを使用者に許す様々な休止形態を提供します。

3つの休止形態の何れかへ移行するにはMCU制御レジスタ(MCUCR)の**休止許可(SE)ビット**が論理1を書かれ、**SLEEP**命令が実行されなければなりません。MCUCRの**休止種別選択(SM1,0)ビット**は**SLEEP**命令によって活性(有効)にされる休止形態(アイドル、パワーダウン、スタンバイ)のどれかを選びます。一覧については表13をご覧ください。MCUが休止形態中に許可した割り込みが起ると、MCUは起動します。その時にMCUは起動時間に加えて4周期停止され、割り込みルーチンを実行し、そして**SLEEP**命令の次の命令から実行を再開します。デバイスが休止から起動するとき、レジスタファイルとSRAMの内容は変更されません。休止形態中にリセットが起ると、MCUは起動し、リセットベクタから実行します。

16ページの図11はATtiny2313の各種クロック系統とその配給を示します。この図は適切な休止形態を選ぶ助けになります。

MCU制御レジスタ (MCU Control Register) MCUCR

このMCU制御レジスタは電力管理用の制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	PUD	SM1	SE	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット5 – SE : 休止許可 (Sleep Enable)

SLEEP命令が実行される時にMCUを休止形態へ移行させるには、休止許可(SE)ビットが論理1を書かれなければなりません。MCUの目的外休止形態移行を避けるため、**SLEEP**命令実行直前に休止許可(SE)ビットを設定(1)し、起動後直ちに解除(0)することが推奨されます。

• ビット6,4 – SM1,0 : 休止種別選択 (Sleep Mode Select Bits 1 and 0)

これらのビットは表13.で示される利用可能な3つの休止形態の1つを選びます。

表13. 休止形態種別選択

SM1	SM0	休止形態種別
0	0	アイドル動作
0	1	パワーダウン動作
1	0	スタンバイ動作
1	1	パワーダウン動作

注: スタンバイ動作は外部クリスタル発振子またはセラミック振動子での使用に対してだけ推奨されます。

アイドル動作

休止種別選択(SM1,0)ビットが'00'を書かれるとき、**SLEEP**命令はMCUをアイドル動作へ移行させ、CPUを停止しますが、USART、アナログ比較器、多用途直列インターフェース(USI)、タイマ/カウンタ、ウォッチドッグ、割り込み機構の継続動作を許します。この休止形態は基本的にclkCPUとclkFLASHを停止する一方、他のクロックに走行を許します。

アイドル動作はMCUにタイマ溢れやUSARTの送信完了などの内部割り込みだけでなく、外部で起動された割り込みからの起動も可能にします。アナログ比較器割り込みからの起動が必要とされないなら、**アナログ比較器制御/状態レジスタ(ACSR)のアナログ比較器禁止(ACD)ビット**を設定(1)することによってアナログ比較器を電源断にできます。これはアイドル動作での消費電力を削減します。

パワーダウン動作

SM1,0ビットが'01'または'11'を書かれると、**SLEEP**命令はMCUをパワーダウン動作へ移行させます。この動作では外部発振器が停止される一方、外部割り込み、多用途直列インターフェース(USI)の開始条件検出、ウォッチドッグ機能は(許可ならば)継続して動作します。外部リセット、ウォッチドッグリセット、低電圧検出(BOD)リセット、USI開始条件検出割り込み、INT0またはINT1の外部レベル割り込み、ピン変化割り込みだけがMCUを起動できます。この休止形態は基本的に生成した全てのクロックを停止し、非同期部の動作だけを許します。

レベルで起動した割り込みがパワーダウン動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。詳細については40ページの「外部割り込み」を参照してください。

パワーダウン動作から起動するとき、起動条件が起きてから起動の効果が現れるまで遅延があります。これは停止されてしまっている後の再始動と安定になることをクロックに許します。この起動(遅延)時間は16ページの「クロック元」で記述されるように、リセット遅延時間を定義するのと同じCKSELヒューズによって定義されます。

スタンバイ動作

外部クリスタル発振子/セラミック振動子クロック種別が選ばれ、SM1,0ビットが'10'の時に、**SLEEP**命令はMCUをスタンバイ動作へ移行させます。この動作は(外部クリスタル用)発振器が走行(動作)を保たれる例外を除いてパワーダウン動作と同じです。デバイスはスタンバイ動作から6クロック周期で起動します。



表14. 各休止形態に於ける動作クロック範囲と復帰起動要因

休止種別	動作クロック範囲			発振器動作 主クロック 供給元	復帰起動要因 (割り込み)				
	clk CPU	clk FLASH	clk IO		INT0,INT1 ピン変化	USI開始条件 検出	SPM/EEPROM 操作可	ウォッチ ドッグ	その他 I/O
アイドル			○	○	○	○	○	○	○
パワーダウン					②	○		○	
スタンバイ (注1)				○	②	○		○	

注1: クロック元として推奨した外部クリスタル発振子またはセラミック振動子が選ばれた場合だけです。

② INT0とINT1についてはレベル割り込みのみです。

消費電力の最小化

これらはAVRが制御するシステムで消費電力の最小化を試みる時に考慮するためのそれぞれの検討点です。一般的に休止形態は可能な限り多く使われるべきで、休止種別は動作するデバイスの機能が可能な限り少なくなるように選ばれるべきです。必要とされない全ての機能は禁止されるべきです。特に次の機能部は最低可能消費電力の達成を試みるとき、特別な考慮を必要とすでしょう。

アナログ比較器

アイドル動作へ移行するとき、アナログ比較器は使われないなら、禁止されるべきです。その他の休止形態でのアナログ比較器は自動的に禁止されます。しかしアナログ比較器が入力として内部基準電圧を使う設定の場合、全休止形態でアナログ比較器は禁止されるべきです。さもなければ内部基準電圧は休止形態と無関係に許可されます。アナログ比較器設定法の詳細については100頁の「[アナログ比較器](#)」を参照してください。

低電圧検出器 (BOD)

低電圧検出器(BOD)が応用で必要とされないなら、この単位部はOFFにされるべきです。低電圧検出器がBODLEVELヒューズによって許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。低電圧検出器(BOD)設定法の詳細については24頁の「[低電圧検出 \(BOD\)](#)」を参照してください。

内部基準電圧

内部基準電圧は低電圧検出器(BOD)とアナログ比較器によって必要とされる時に許可されます。これら単位部が上の項で記述されたように禁止されると、内部基準電圧は禁止され、電力を消費しません。再び許可する場合、この出力が使われる前に使用者は基準電圧へ起動(安定時間)を与えなければなりません。基準電圧が休止形態でON保持される場合、この出力は直ちに使えます。起動時間の詳細については25頁の「[内部基準電圧](#)」を参照してください。

ウォッチドッグ タイマ

ウォッチドッグ タイマが応用で必要とされないなら、この単位部はOFFにされるべきです。ウォッチドッグ タイマが許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。ウォッチドッグ タイマ設定法の詳細については26頁の「[ウォッチドッグ タイマ](#)」を参照してください(訳注:原書に対して参照先修正)。

ポートピン

休止形態へ移行するとき、全てのポートピンは最小電力使用に設定されるべきです。最も重要なことはその時にピンが抵抗性負荷を駆動しないのを保証することです。I/Oクロック(clk_{I/O})が停止される休止形態ではデバイスの入力緩衝部が禁止されます。これは必要とされない時に入力論理回路によって電力が消費されないことを保証します。いくつかの場合で入力論理回路は起動条件を検出するために必要とされ、その時は許可されます。どのピンが許可されるかの詳細については32頁の「[デジタル入力許可と休止形態](#)」を参照してください。入力緩衝部が許可され、入力信号が浮いている状態のままか、またはアナログ信号電圧がVCC/2付近の場合、入力緩衝部は過大な電力を消費するでしょう。

アナログ入力ピンに対するデジタル入力緩衝部は常に禁止されるべきです。入力ピンでのVCC/2付近のアナログ信号入力は活動動作でも重要な電流を引き起こし得ます。デジタル入力緩衝部はデジタル入力禁止レジスタ(DIDR)の書き込みによって禁止できます。詳細については101頁の「[デジタル入力禁止レジスタ - DIDR](#)」を参照してください。

内蔵デバッグ機能 (dW) (訳注:共通性から追加)

内蔵デバッグ機能がDWENヒューズによって許可され、チップが休止形態へ移行すると、主クロック元は許可に留まり、従って常に電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。

システム制御とリセット

AVRのリセット

リセット中、全てのI/Oレジスタはそれらの初期値に設定され、プログラムはリセットベクタから実行を開始します。リセットベクタに配置される命令は、きつとリセット処理ルーチンへの無条件相対分岐(RJMP)命令でしょう。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが配置できます。図14.の回路構成図はリセット論理回路を示します。表80.はリセット回路の電気的特性を定義します。

AVRのI/Oポートはリセット元が有効になると直ちにそれらの初期状態にリセットされます。これはどのクロック元の走行も必要ありません。

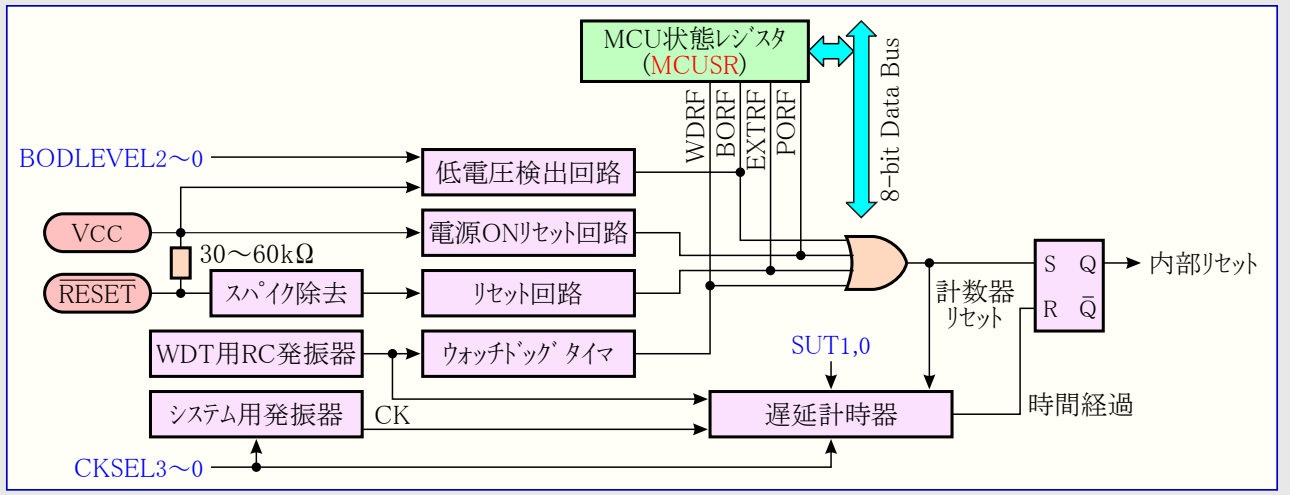
全てのリセット元が無効にされてしまった後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電源に許します。遅延タイマの遅延時間はSUTヒューズとCKSELヒューズを通して使用者によって定義されます。この遅延時間についての各種選択は16頁の「クロック元」で示されます。

リセット元

ATtiny2313には次の4つのリセット元があります。

- **電源ONリセット** …… 供給電圧が電源ONリセット閾値電圧(V_{POT})以下でMCUがリセットされます。
- **外部リセット** …… RESETピンが最小パルス幅以上Lowレベルに保たれると、MCUがリセットされます。
- **ウォッチドッグリセット** …… ウォッチドッグシステムリセット動作が許可され、ウォッチドッグタイマが終了すると、MCUがリセットされます。
- **低電圧リセット** …… 低電圧検出器(BOD)が許可され、供給電圧(V_{CC})が低電圧検出電圧(V_{BOT})以下でMCUがリセットされます。

図14. リセット回路構成



電源ONリセット

電源ONリセット(POR)パルスはチップ上の検出回路によって生成されます。検出電圧は表80.で定義されます。POR信号はVCCが検出電圧以下の時は必ず活性(有効)にされます。POR回路は供給電圧異常検出は勿論、始動リセットの起動にも使えます。

電源ONリセット回路はデバイスが電源投入でリセットされることを保証します。電源ONリセット閾値電圧(V_{POT})への到達はVCCの上昇後にデバイスがどのくらいリセットを保つかを決める遅延計数器(タイマ)を起動します。VCCがこの検出電圧以下に低下すると、リセット信号はどんな遅延もなく再び有効にされます。

図15. 内蔵電源ONリセット (RESETはVCCに接続)

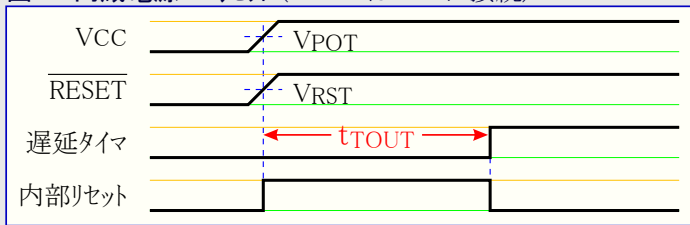
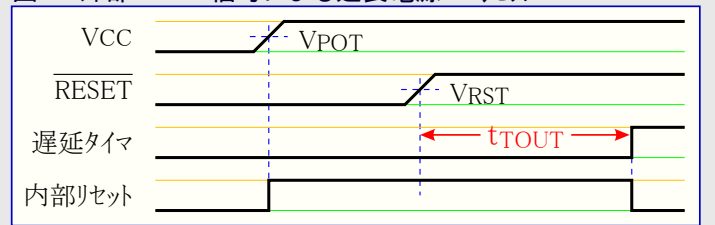


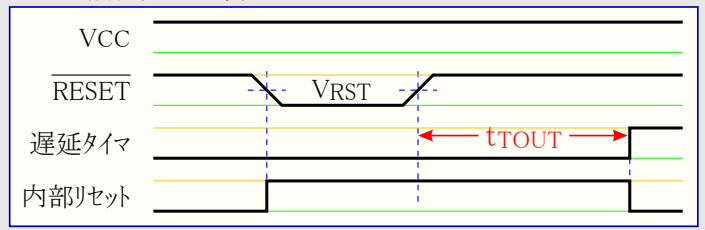
図16. 外部RESET信号による延長電源ONリセット



外部リセット

外部リセットはRESETピンのLowレベルによって生成されます。クロックが動いていなくても、最小パルス幅(表80参照)以上のリセットパルスはリセットを生成します。短すぎるパルスはリセット生成が保証されません。印加された信号の上昇がリセット閾値電圧(VRST)に達すると(遅延タイマを起動し)、遅延タイマは遅延時間(t_{TOUT})経過後にMCUを始動します。

図17. 動作中の外部リセット



低電圧(ブラウンアウト)検出リセット

ATtiny2313には固定された起動(検出)電圧と比較することによって動作中のVCCを監視するチップ上の低電圧検出(BOD)回路があります。BODの起動電圧はBODLEVELヒューズによって選べます。この起動電圧はスパイク対策BODを保証するためにヒステリシスを持ちます。検出電圧のヒステリシスは $V_{BOT+}=V_{BOT}+V_{HYST}/2$ 、 $V_{BOT-}=V_{BOT}-V_{HYST}/2$ と解釈されるべきです。

BODが許可されて、VCCが起動電圧以下に下降すると(図18の V_{BOT-})、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(図18の V_{BOT+})、(遅延タイマが起動され)遅延タイマは遅延時間(t_{TOUT})経過後にMCUを始動します。

BOD回路は電圧が表16.で与えられる t_{BOD} 時間より長く起動電圧以下に留まる場合のみ、VCCでの低下を検出します。

図18. 動作中の低電圧検出リセット

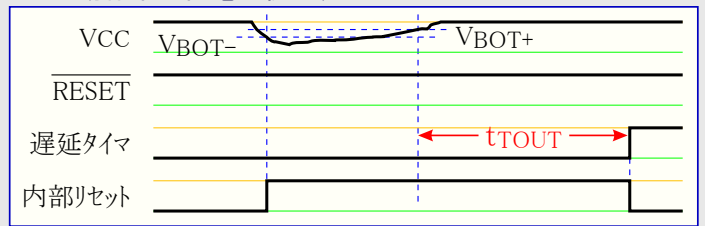


表16. 低電圧検出(BOD)特性

シンボル	項目	最小	代表	最大	単位
V_{HYST}	低電圧検出ヒステリシス電圧		50		mV
t_{BOD}	最小低電圧検出時間		2		μ s

表15. BODLEVELヒューズ (V_{BOT}) 設定 (注1)

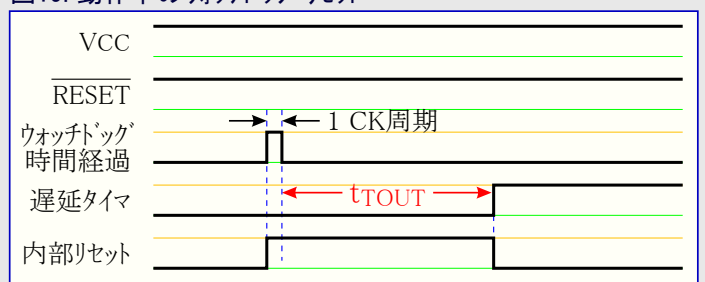
BODLEVEL2~0	最小	代表	最大	単位
1 1 1	低電圧検出(BOD)リセット禁止			
1 1 0		1.8		V
1 0 1		2.7		
1 0 0		4.3		
0 0 0 ~ 0 1 1	(予約)			

注1: いくつかのデバイスで V_{BOT} が公称最低動作電圧以下の可能性があります。この状態のデバイスについては製造検査中、 $VCC=V_{BOT}$ に落として検査されています。これはマイクロコントローラの正しい動作がもはや保証されない電圧になる前に低電圧検出(BOD)リセットが起きることを保証します。この検査はATtiny2313VについてBODLEVEL=110を、ATtiny2313についてBODLEVEL=101を使って実行されます。

ウォッチドッグリセット

ウォッチドッグ時間経過時、(内部的に)1 CK周期幅の短いリセットパルスを生成します。本パルスの下降端で遅延タイマは遅延時間(t_{TOUT})の計時を始めます。ウォッチドッグタイマ操作の詳細については26頁を参照してください。

図19. 動作中のウォッチドッグリセット



MCU状態レジスタ (MCU Status Register) MCUSR

MCU状態レジスタはどのリセット元がMCUリセットを起こしたかの情報を提供します。

ビット	7	6	5	4	3	2	1	0	
\$34 (\$54)	-	-	-	-	WDRF	BORF	EXTRF	PORF	MCUSR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	内容参照	内容参照	内容参照	内容参照	

• ビット7~4 - Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

• ビット3 - WDRF : ウォッチドッグ リセット フラグ (Watchdog Reset Flag)

このビットはウォッチドッグ リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

• ビット2 - BORF : 低電圧リセット フラグ (Brown-Out Reset Flag)

このビットは低電圧リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

• ビット1 - EXTRF : 外部リセット フラグ (External Reset Flag)

このビットは外部リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

• ビット0 - PORF : 電源ONリセット フラグ (Power-on Reset Flag)

このビットは電源ONリセットが起こると設定(1)されます。このビットはこのフラグへの論理0書き込みによってのみリセット(0)されます。

リセット条件の確認にリセットフラグを使うために使用者はプログラム内で可能な限り早くMCUSRを読み、それから解除(0)すべきです。別のリセットが起こる前にこのレジスタが解除(0)されると、リセット元はリセットフラグを調べることによって得られます。

内部基準電圧

ATtiny2313は内部基準電圧が特徴です。この基準電圧は低電圧検出(BOD)に使われ、アナログ比較器入力としても使えます。

基準電圧許可信号と起動時間

この基準電圧には使われるべき方法に影響を及ぼすかもしれない起動時間があります。この起動時間は表17.で与えられます。節電のために、この基準電圧は常にONではありません。この基準電圧は次の状態中ONです。

1. 低電圧検出リセット許可時 (BODLEVEL2~0ビュースのプログラム(0)により)
2. アナログ比較器基準電圧接続時 (アナログ比較器制御/状態レジスタ(ACSR)の基準電圧入力選択(ACBG)=1)

従って低電圧検出(BOD)が許可されていないと、ACBGの設定(=1)後、使用者はアナログ比較器出力が使われる前に、基準電圧へ起動時間を与えなければなりません。パワーダウン動作での消費電力を減らすため、使用者はパワーダウン動作へ移行する前に基準電圧がOFFされるのを保証することで上の2つの状態を避けられます。

表17. 内部基準電圧特性

シンボル	項目	条件	最小	代表	最大	単位
V _{BG}	基準電圧	VCC=2.7V TA=25°C	1.0	1.1	1.2	V
t _{BG}	起動時間			40	70	μs
I _{BG}	消費電流				15	μA

ウォッチドッグ タイマ

ATtiny2313は強化されたウォッチドッグ タイマ(WDT)を持ちます。主な機能を次に示します。

- 独立した内蔵発振器からのクロック駆動
- 3つの動作種別
 - 割り込み
 - システム リセット
 - 割り込みとシステム リセット
- 選択可能な16ms～8sの計時完了時間
- 安全動作用のウォッチドッグ常時ONハートウェア ヒューズ

ウォッチドッグ タイマ(WDT)はチップ上の独立した128kHz発振器の間隔で計時するタイマです。WDTは計数器が与えられた計時完了値に達した時に割り込みまたはシステム リセットを生じます。通常動作では計時完了値へ達する前に計数器を再始動するために、システムはウォッチドッグ リセット (WDR) 命令を使う必要があります。システムが計数器を再始動しなければ、割り込みまたはシステム リセットが起こるでしょう。

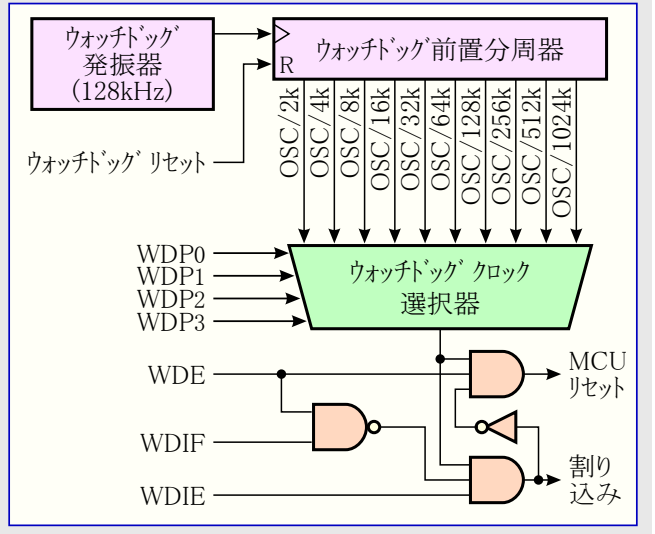
割り込み動作種別では、タイマ計時完了時にWDTが割り込みを生じます。この割り込みは休止形態からデバイスを起動するためや、一般的なシステム タイマとしても使えます。1つの例は或る動作に対して許された最大時間を制限することで、その動作が予測されたより長く走行する時に割り込みを生じます。システム リセット動作種別ではタイマ計時完了時にWDTがリセットを生じます。これは一般的にコード外走行の場合の中断を防止するのに使われます。3つ目の動作種別は先に割り込みを生じ、その後にシステム リセット動作種別に切り替えることで、他の2つの動作種別の組み合わせとなる、割り込み及びシステム リセット動作種別です。この動作種別は例えばシステム リセットに先立って重要なパラメータを保存することによって安全な停止を許します。

ウォッチドッグ常時ON(WDTON)ヒューズのプログラム(0)はウォッチドッグ タイマをシステム リセット動作種別に強制します。このヒューズのプログラム(0)でシステム リセット動作(WDE)ビットと割り込み動作(WDIE)ビットは各々、'1'と'0'に固定されます。

更にプログラム保護を保証するためにウォッチドッグ設定の変更は時間制限手順に従わなければなりません。システム リセット許可(WDE)の解除と計時完了時間設定の変更についての手順は次のとおりです。

1. 同じ操作(命令)でウォッチドッグ変更許可(WDCE)とWDEに論理1を書きます。WDEビットの直前の値に拘らず、論理1がWDEに書かれなければなりません。
2. 次からの4クロック周期内に同じ操作(命令)で欲したWDEとウォッチドッグ タイマ前置分周選択(WDP3～0)ビットを書きますが、WDCEビットは解除(0)されてです。これは1操作(命令)で行わなければなりません。

図20. ウォッチドッグ タイマ構成図



次のコード例はウォッチドッグ タイマをOFFに切り替えるアセンブリ言語とC言語の関数を示します。本例は(例えば全割り込み禁止によって)割り込みが制御され、それ故これらの関数実行中に割り込みが起きない前提です。

アセンブリ言語プログラム例

```

WDT_OFF:    CLI                ;全割り込み禁止
            WDR                ;ウォッチドッグ タイマ リセット
            IN     R16, MCUSR    ;MCUSR値を取得
            ANDI   R16, ~(1<<WDRF) ;WDRF論理0値を取得
            OUT   MCUSR, R16    ;ウォッチドッグ リセット フラグ(WDRF)解除
            IN     R16, WDTCSR   ;現WDTCSR値を取得(他ビット保護用)
            ORI    R16, (1<<WDCE) | (1<<WDE) ;WDCEとWDE論理1値を設定
            OUT   WDTCSR, R16   ;WDCEとWDEに論理1書き込み
            LDI   R16, (0<<WDE) ;WDE論理0値を取得
            OUT   WDTCSR, R16   ;ウォッチドッグ禁止
            SEI                ;全割り込み許可
            RET                ;呼び出し元へ復帰
    
```

C言語プログラム例

```

void WDT_off(void)
{
    __disable_interrupt();          /* 全割り込み禁止 */
    __watchdog_reset();            /* ウォッチドッグ タイマ リセット */
    MCUSR &= ~(1<<WDRF);           /* ウォッチドッグ リセット フラグ(WDRF)解除 */
    WDTCSR |= (1<<WDCE) | (1<<WDE); /* WDCEとWDEに論理1書き込み */
    WDTCSR = 0x00;                /* ウォッチドッグ禁止 */
    __enable_interrupt();         /* 全割り込み許可 */
}
    
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

注: ウォッチドッグが偶然に許可されると(例えばポインタの逸脱や低電圧(ブラウンアウト)状態)、デバイスはリセットし、ウォッチドッグは許可に留まります。コードがウォッチドッグ操作の初期設定をしなければ、これは計時完了の無限繰り返しを引き起こすかもしれません。この状態を避けるため、応用ソフトウェアは例えウォッチドッグが使われなくても、初期化ルーチンで**WDRFフラグ**と**WDE制御ビット**を常に解除(0)すべきです。

次のコード例はウォッチドッグ タイマの計時完了値変更用のアセンブリ言語とC言語の関数を示します。

アセンブリ言語プログラム例

```

WDT_PRS:    CLI                ;全割り込み禁止
            WDR                ;ウォッチドッグ タイマ リセット
            IN     R16, WDTCSR   ;現WDTCSR値を取得(他ビット保護用)
            ORI    R16, (1<<WDCE) | (1<<WDE) ;WDCEとWDE論理1値を設定
            OUT   WDTCSR, R16   ;WDCEとWDEに論理1書き込み
            LDI   R16, (1<<WDE) | (1<<WDP2) | (1<<WDP0) ;WDE=1,計時間隔=0.5s値を取得
            OUT   WDTCSR, R16   ;0.5s監視間隔リセット動作開始
            SEI                ;全割り込み許可
            RET                ;呼び出し元へ復帰
    
```

C言語プログラム例

```

void WDT_off(void)
{
    __disable_interrupt();          /* 全割り込み禁止 */
    __watchdog_reset();            /* ウォッチドッグ タイマ リセット */
    WDTCSR |= (1<<WDCE) | (1<<WDE); /* WDCEとWDEに論理1書き込み */
    WDTCSR = (1<<WDE) | (1<<WDP2) | (1<<WDP0); /* 0.5s監視間隔リセット動作開始 */
    __enable_interrupt();         /* 全割り込み許可 */
}
    
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

注: ウォッチドッグ タイマ前置分周選択(WDP3~0)ビットの変更がより短い計時完了周期になってしまう結果になり得るため、ウォッチドッグ タイマはWDPビットのどんな変更にも先立ってリセット(WDR命令)されるべきです。

ウォッチドッグ タイム制御/状態レジスタ (Watchdog Timer Control Register) WDTCSR

ビット	7	6	5	4	3	2	1	0	
\$21 (\$41)	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	WDTCSR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	不定	0	0	0	

• ビット7 – WDIF : ウォッチドッグ割り込み要求フラグ (Watchdog Interrupt Flag)

ウォッチドッグ タイムが割り込みに設定され、ウォッチドッグ タイムで計時完了が起こると、本ビットが設定(1)されます。対応する割り込み処理ベクタを実行すると、WDIFはハードウェアによって解除(0)されます。代わりにWDIFはこのフラグへの論理1書き込みによっても解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとウォッチドッグ割り込み許可(WDIE)が設定(1)されていれば、ウォッチドッグ計時完了割り込みが実行されます。

• ビット6 – WDIE : ウォッチドッグ割り込み許可 (Watchdog Interrupt Enable)

このビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、ウォッチドッグ割り込みが許可されます。この設定(=1)との組み合わせでウォッチドッグリセット許可(WDE)ビットが解除(0)されると、割り込み動作種別になり、ウォッチドッグ タイムで計時完了が起こると、対応する割り込みが実行されます。

WDEが設定(1)されると、ウォッチドッグ タイムは割り込み及びシステムリセット動作種別になります。ウォッチドッグ タイムでの最初の計時完了がウォッチドッグ割り込み要求(WDIF)フラグを設定(1)します。対応する割り込みベクタの実行はハードウェアによってWDIEとWDIFを自動的に解除(0)します。これは割り込みを使う間のウォッチドッグリセット保護を維持するのに有用です。割り込み及びシステムリセット動作種別に留まるには、各割り込み後にWDIEが設定(1)されなければなりません。然しながら、ウォッチドッグシステムリセット動作種別の安全機能に危険を及ぼすかもしれないため、これは割り込み処理ルーチン自身内で行われるべきではありません。次の計時完了に先立って割り込みが実行されない場合、システムリセットが適用(実行)されます。

表18. ウォッチドッグ タイム設定

WDTON	WDE	WDIE	動作種別	計時完了での動作
1	0	0	停止	なし
1	0	1	割り込み	割り込み
1	1	0	システムリセット	リセット
1	1	1	割り込み及びシステムリセット	割り込み、その後システムリセット動作種別
0	x	x	システムリセット	リセット

注: WDTONヒューズは、0がプログラム、1が非プログラムを意味します。

• ビット4 – WDCE : ウォッチドッグ変更許可 (Watchdog Change Enable)

このビットはウォッチドッグリセット許可(WDE)と前置分周器ビットの変更用の時間制限手順で使われます。WDEビットの解除(0)や前置分周器ビット変更のため、WDCEは設定(1)されなければなりません。

一旦1を書かれると、4クロック周期後にハードウェアがWDCEを解除(0)します。

• ビット3 – WDE : ウォッチドッグリセット許可 (Watchdog System Reset Enable)

WDEはMCU状態レジスタ(MCUSR)のウォッチドッグリセットフラグ(WDRF)によって無効にされます。これはWDRFが設定(1)されると、WDEが常に設定(1)されることを意味します。WDEを解除(0)するにはWDRFが先に解除(0)されなければなりません。この特徴は失敗を引き起こす状態中の複数リセットと失敗後の安全な起動を保証します。

• ビット5,2~0 – WDP3~0 : ウォッチドッグ タイム前置分周選択 (Watchdog Timer Prescaler 3,2,1 and 0)

このWDP3~0ビットはウォッチドッグ タイムが走行する時のウォッチドッグ タイムの前置分周を決めます。各種前置分周値と対応する計時完了周期は表19.で示されます。

表19. ウォッチドッグ前置分周選択

WDP3	0				1							
	0		1		0		1		0		1	
WDP2	0		1		0		1		0		1	
WDP1	0	1	0	1	0	1	0	1	0	1	0	1
WDP0	0	1	0	1	0	1	0	1	0	1	0	1
WDT発振周期数	2k	4k	8k	16k	32k	64k	128k	256k	512k	1024k	(予約)	
代表的計時完了周期 (VCC=5V)	16ms	32ms	64ms	0.125s	0.25s	0.5s	1.0s	2.0s	4.0s	8.0s		

割り込み

本章はATtiny2313によって実行される割り込み操作の詳細を記述します。AVR割り込み操作の一般説明については9頁の「リセットと割り込みの扱い」を参照してください。

ATtiny2313の割り込みベクタ

表20. リセットと割り込みのベクタ

ベクタ番号	プログラム アドレス	発生元	備考
1	\$0000	リセット	電源ON, WDT, BOD等の各種リセット
2	\$0001	INT0	外部割り込み要求0
3	\$0002	INT1	外部割り込み要求1
4	\$0003	タイマ/カウンタ1 CAPT	タイマ/カウンタ1捕獲発生
5	\$0004	タイマ/カウンタ1 COMPA	タイマ/カウンタ1比較A一致
6	\$0005	タイマ/カウンタ1 OVF1	タイマ/カウンタ1溢れ
7	\$0006	タイマ/カウンタ0 OVF0	タイマ/カウンタ0 溢れ
8	\$0007	USART RX	USART 受信完了
9	\$0008	USART UDRE	USART 送信緩衝部空き
10	\$0009	USART TX	USART 送信完了
11	\$000A	アナログ比較器 ANA_COMP	アナログ比較器出力遷移
12	\$000B	PCINT (PCI)	ピン変化割り込み要求
13	\$000C	タイマ/カウンタ1 COMPB	タイマ/カウンタ1比較B一致
14	\$000D	タイマ/カウンタ0 COMPA	タイマ/カウンタ0比較A一致
15	\$000E	タイマ/カウンタ0 COMPB	タイマ/カウンタ0比較B一致
16	\$000F	USI START	USI 開始条件検出
17	\$0010	USI OVF	USI 計数器溢れ
18	\$0011	EEPROM EE_RDY	EEPROM 操作可
19	\$0012	ウォッチドッグ WDT	ウォッチドッグ計時完了

プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが配置できます(注: 共通性から本行追加)。ATtiny2313での最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

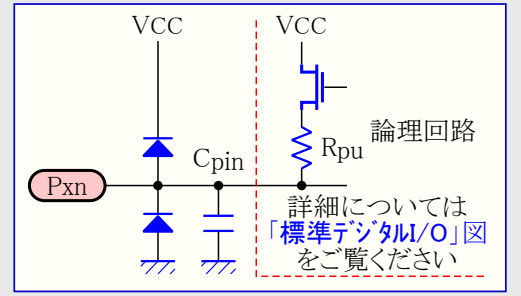
アドレス	ラベル	命令	注釈
\$0000		RJMP RESET	;各種リセット
\$0001		RJMP EXT_INT0	;外部割り込み要求0
\$0002		RJMP EXT_INT1	;外部割り込み要求1
\$0003		RJMP TIM1_CAPT	;タイマ/カウンタ1捕獲発生
\$0004		RJMP TIM1_COMPA	;タイマ/カウンタ1比較A一致
\$0005		RJMP TIM1_OVF	;タイマ/カウンタ1溢れ
\$0006		RJMP TIM0_OVF	;タイマ/カウンタ0溢れ
\$0007		RJMP USART_RXC	;USART 受信完了
\$0008		RJMP USART_UDRE	;USART 送信緩衝部空き
\$0009		RJMP USART_TX	;USART 送信完了
\$000A		RJMP ANA_COMP	;アナログ比較器出力遷移
\$000B		RJMP PCINT	;ピン変化割り込み要求
\$000C		RJMP TIM1_COMPB	;タイマ/カウンタ1 比較B一致
\$000D		RJMP TIM0_COMPA	;タイマ/カウンタ0 比較A一致
\$000E		RJMP TIM0_COMPB	;タイマ/カウンタ0 比較B一致
\$000F		RJMP USI_STRT	;USI 開始条件検出
\$0010		RJMP USI_OVF	;USI 計数器溢れ
\$0011		RJMP EE_RDY	;EEPROM操作可
\$0012		RJMP WDT_OVF	;ウォッチドッグ計時完了
;			
\$0013	RESET:	LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0014		OUT SPL, R16	;スタック ポインタ(下位)を初期化
)	;以下、I/O初期化など

入出力ポート

序説

全てのAVRのポートは標準デジタルI/Oポートとして使われる時に真の読み-変更-書き(リード・モディファイライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方向をも無意識に変更することなく、1つのポートピンの方向を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。このピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピンは個別に選択可能な、供給電圧で抵抗値が変化しないプルアップ抵抗を持っています。全てのI/Oピンは図21.で示されるようにVCCとGNDの両方に保護ダイオードを持っています。各値の完全な一覧については118頁の「電気的特性」を参照してください。

図21. 入出力ピン等価回路



本章内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記されたPORTxnがポートBのビット3に対してはPORTB3)が使われなければなりません。物理的なI/Oレジスタとビット位置は39頁の「I/Oポート用レジスタ」で一覧されます。

各々1つの出力レジスタ(PORTx)、方向レジスタ(DDRx)、入力レジスタ(PINx)の各ポートに対して、3つI/Oメモリアドレス位置が割り当てられます。入力レジスタのI/O位置は読むだけで、一方出力レジスタと方向レジスタは読み書き(両方)です。けれどもPINxレジスタのビットへの論理1書き込みは、出力レジスタの対応ビット値を(1/0)反転する結果になります。加えてMCU制御レジスタ(MCUCR)のプルアップ禁止(PUD)ビットが設定(1)されると、全ポートで全ビットに対してプルアップ機能を禁止します。

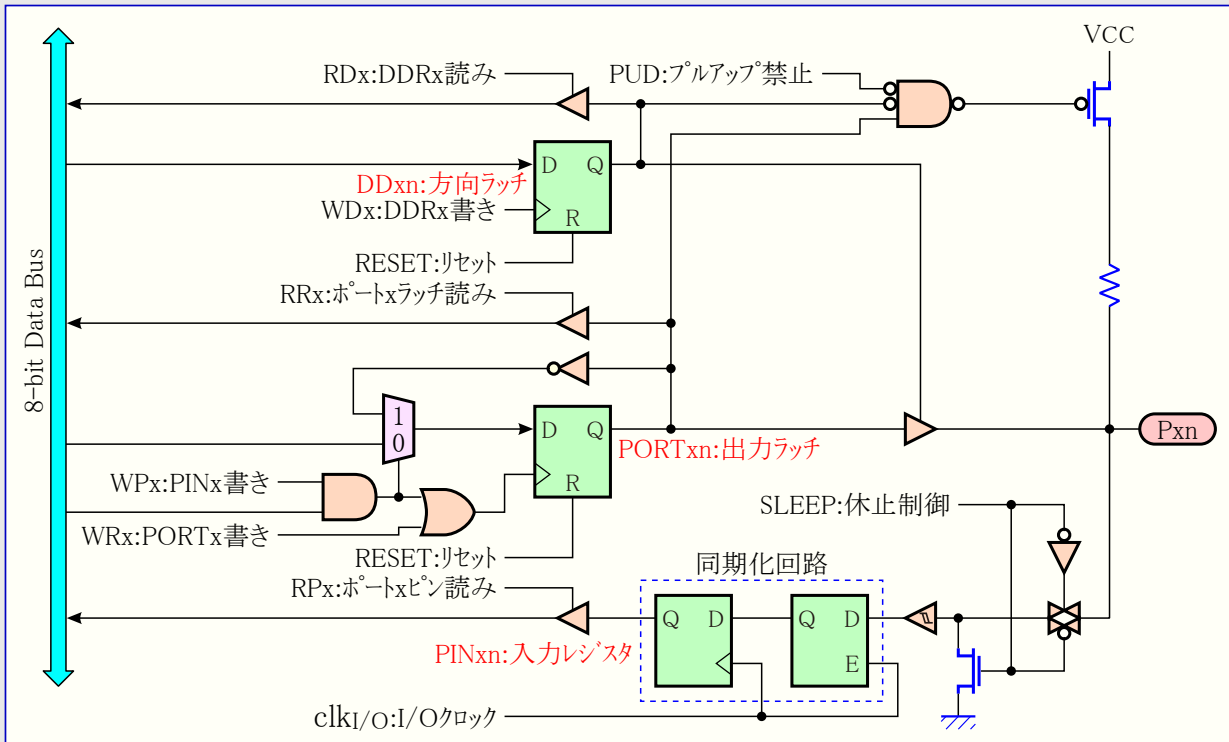
標準デジタルI/OとしてのI/Oポートの使用は次の「標準デジタル入出力としてのポート」で記述されます。多くのポートピンはデバイスの周辺機能用の交換機能と多重化されます。ポートピンとの各交換機能のインターフェース法は33頁の「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項を参照してください。

ポートピンのいくつかの交換機能の許可は、そのポート内の他のピンの標準デジタル入出力としての使用に影響しないことに注意してください。

標準デジタル入出力としてのポート

このポートは任意の内部プルアップ付き双方向I/Oポートです。図22.はここで属にPxnと呼ばれるI/Oポートピンの1つの機能説明を示します。

図22. 標準デジタル入出力回路構成



注: WRx, WPx, WDx, RRx, RPx, RDxは同一ポート内の全ピンで共通です。
clkI/O, SLEEP, PUDは全ポートで共通です。

ピンの設定

各ポートピンは3つのレジスタビットDDxn、PORTxn、PINxnから成ります。39頁の「I/Oポート用レジスタ」で示されるように、DDxnビットはDDRx I/Oアドレス、PORTxnビットはPORTx I/Oアドレス、PINxnビットはPINx I/Oアドレスでアクセスされます。

DDRxレジスタ内のDDxnビットはそのピンの方向を選びます。DDxnが論理1を書かれるとPxnは出力ピンとして設定されます。DDxnが論理0を書かれるとPxnは入力ピンとして設定されます。

そのピンが入力ピンとして設定される時にPORTxnが論理1を書かれると、プルアップ抵抗が活性(有効)にされます。プルアップ抵抗をOFFに切り替えるにはPORTxnが論理0を書かれるか、またはそのピンが出力ピンとして設定されなければなりません。ポートピンは例えばクロックが動いていなくても、リセット条件が活性(有効)になるとHi-Zにされます。

そのピンが出力ピンとして設定される時にPORTxnが論理1を書かれると、そのポートピンはHigh(1)に駆動されます。そのピンが出力ピンとして設定される時にPORTxnが論理0を書かれると、そのポートピンはLow(0)に駆動されます。

ピンの出力交互切り替え

PINxnへの論理1書き込みはDDRxnの値に拘らず、PORTxnの値を反転切り替えます。SBI命令がポート内の1ビットの反転切り替えに使えることに注目してください。

入出力間の切り替え

Hi-Z入力(DDxn=0, PORTxn=0)とHigh出力(DDxn=1, PORTxn=1)間の切り替え時、プルアップ許可入力(DDxn=0, PORTxn=1)またはLow出力(DDxn=1, PORTxn=0)のどちらかの中間状態が生じるに違いありません。通常、高インピーダンス環境は強力なHigh(吐き出し)駆動部とプルアップ間の違いに気付かないので、プルアップが許可された状態は十分受け入れられます。この事例でないなら、全ポートの全プルアップを禁止するために、MCU制御レジスタ(MCUCR)のプルアップ禁止(PUD)ビットが設定(1)できます。

プルアップ入力とLow出力間の切り替えは同じ問題を発生します。使用者は中間状態としてHi-Z入力(DDxn=0, PORTxn=0)またはHigh出力(DDxn=1, PORTxn=1)のどちらかを使わなければなりません。

表21.はピン値に対する制御信号の一覧を示します。

表21. ポートピンの設定

DDxn	PORTxn	PUD (MCUCR)	入出力	プルアップ抵抗	備考
0	0	X	入力	なし	高インピーダンス (Hi-Z)
0	1	0	入力	あり	Pxnに外部からLowを入力すると吐き出し電流が流れます。
0	1	1	入力	なし	高インピーダンス (Hi-Z)
1	0	X	出力	なし	Low (吸い込み)出力
1	1	X	出力	なし	High (吐き出し)出力

ピン値の読み込み

DDxn方向ビットの設定に関係なく、ポートピンはPINxnレジスタビットを通して読めます。図22.で示されるようにPINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。図23.は外部的に加えられたピン値を読む時の同期化タイミング図を示します。伝播遅延の最小と最大は各々 $t_{pd,min}$ と $t_{pd,max}$ で示されます。

(図23.で)システムクロックの最初の上昇端の直後から始まるクロック周期を考察してください。このラッチはクロックがLowの時に閉じ、クロックがHighの時に同期ラッチ信号の斜線部分で示されるように通過(トランスペアレント)となります。この信号値はシステムクロックがLowになる時に保持(ラッチ)されます。それが続くクロックの上昇端でPINxnレジスタに取り込まれます。2つの矢印 $t_{pd,min}$ と $t_{pd,max}$ によって示されるように、ピン上の単一信号遷移は出現時点に依存して0.5~1.5システムクロック周期遅らされます。

ソフトウェアが指定したピン値を読み戻す時は、図24.で示されるようにNOP命令が挿入されなければなりません。OUT命令はシステムクロックの上昇端で同期ラッチを設定します。この場合、同期化回路を通過する遅延時間(t_{pd})は1システムクロック周期です。

図23. 外部供給ピン値読み込み時の同期化

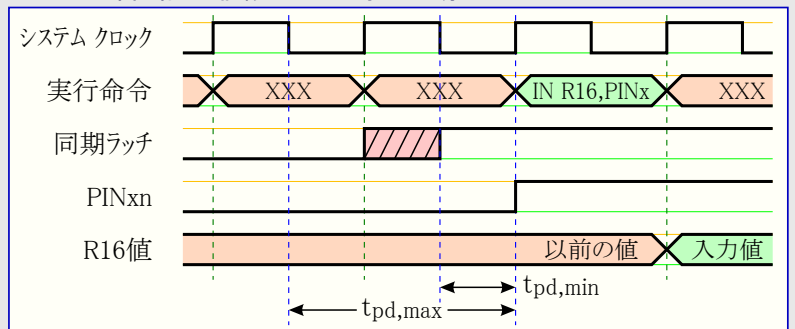
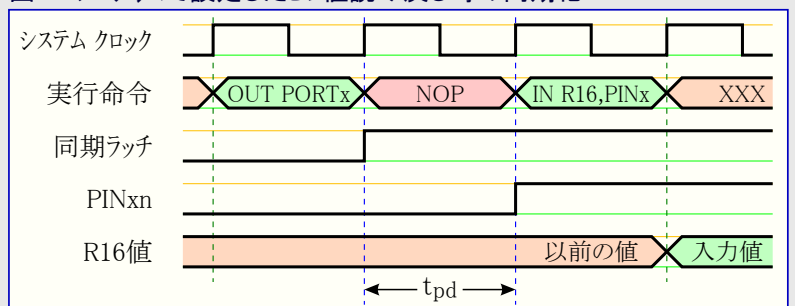


図24. プログラムで設定したピン値読み戻し時の同期化



次のコード例はポートBピンの0と1をHigh出力、2と3をLow出力、6と7をプルアップ指定として4～7を入力に設定する方法を示します。結果のピン値が再び読み戻されますが、前記で検討されたように、いくつかのピンへ直前に指定された値を読み戻すことができるようにNOP命令が挿入されます。

アセンブリ言語プログラム例

```

~
LDI    R16, (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0)    ;プルアップとHigh値を取得
LDI    R17, (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0) ;出力ビット値を取得
OUT    PORTB, R16    ;プルアップとHigh値を設定
OUT    DDRB, R17    ;入出力方向を設定
NOP    ;同期化遅延対処
IN     R16, PINB    ;ピン値読み戻し
~
;

```

C言語プログラム例

```

unsigned char i;
~
PORTB = (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0);    /* */
DDRB = (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0); /* プルアップとHigh値を設定 */
__no_operation(); /* 入出力方向を設定 */
i = PINB; /* 同期化遅延対処 */
~
/* ピン値読み戻し */
/* */

```

注: アセンブリ言語プログラムについてはプルアップがピン0,1,6,7に設定されてから、ビット0と1の強力なHigh駆動部としての再定義、ビット2と3のLow駆動部としての定義、方向ビットが正しく設定されるまでの時間を最小とするために2つの一時レジスタが使われます。

デジタル入力許可と休止形態

図22.で示されるようにデジタル入力信号はシュミットトリガの入力をGNDにクランプできます。この図でSLEEPと印された信号は入力信号のいくつかが開放のまま、またはVCC/2付近のアナログ信号電圧を持つ場合の高消費電力を避けるため、**パワーダウン動作**と**スタンバイ動作**でMCU休止制御器によって設定(1)されます。

SLEEPは外部割り込みピンとして許可されたポートピンに対しては無視されます。外部割り込み要求が許可されないなら、SLEEPは他のピンについてと同様に有効です。SLEEPは33頁の「**交換ポート機能**」で記載されるように様々な他の交換機能によっても無視されません。

外部割り込みが許可されていない”上昇端、下降端または論理変化(両端)割り込み”として設定された非同期外部割り込みピンに論理1が存在すると、上で言及した休止形態から(復帰)再開する時に、これらの休止形態に於けるクランプが要求された論理変化を生ずるので、対応する外部割り込み要求フラグが設定(=1)されます。

未接続ピン (訳注:共通性のため本項追加)

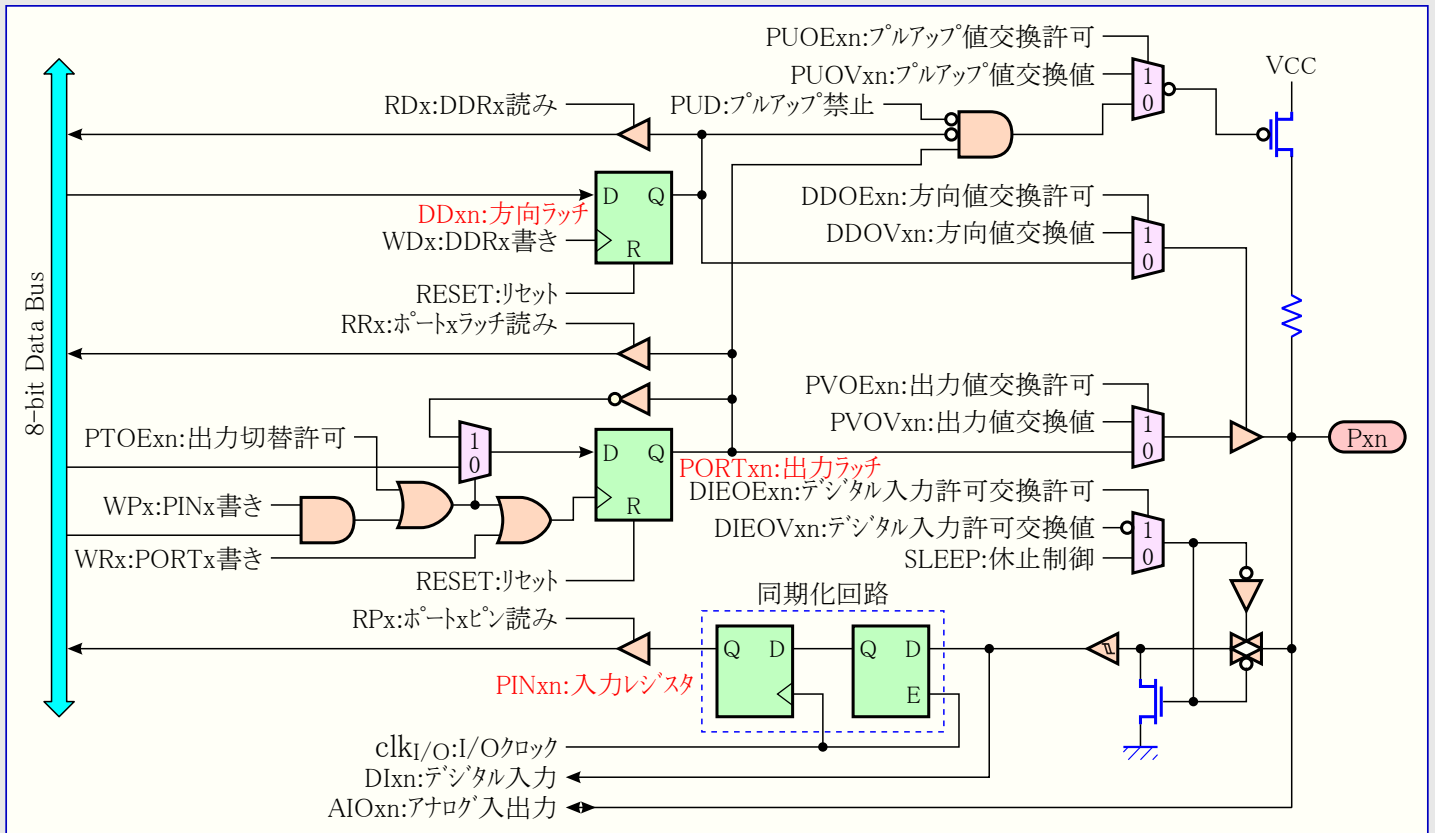
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

未使用ピンの定義されたレベルを保証する最も簡単な方法は内部プルアップを許可することです。この場合、リセット中のプルアップは禁止されます。リセット中の低消費電力が重要なら、外部プルアップまたはプルダウンを使うことが推奨されます。未使用ピンを直接GNDまたはVCCに接続することは、ピンが偶然に出力として設定されると過電流を引き起こす可能性があるため推奨されません。

交換ポート機能

多くのポートピンには標準デジタル入出力に加え交換機能があります。図25.は単純化された図22.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVR マイクロ コントローラシステムの全ポートピンに適用できる一般的な記述として取り扱います。

図25. 交換ポート機能入出力回路構成



注: WRx, WPx, WDx, RRx, RPx, RDxは同一ポート内の全ピンで共通です。
clkI/O, SLEEP, PUDは全ポートで共通です。他の信号は各ピン固有です。

表22.は重複(交換)信号の機能一覧を示します。図25.で示すピンとポートは次表で示されません。重複(交換)信号は交換機能を持つ機能部で内部的に生成されます。

表22. 交換機能用交換信号の一般定義

信号略名	信号名	意味
PUOE	プルアップ値交換許可	1で、プルアップ許可はPUOV信号で制御され、0の場合、DDxn=0, PORTxn=1, PUD=0でプルアップが許可されます。
PUOV	プルアップ値交換値	PUOE=1時、DDxn, PORTxn, PUDの値に関係なく、プルアップの有(1)/無(0)を指定します。
DDOE	方向値交換許可	1で、出力駆動部はDDOV信号で制御され、0の場合、DDxnレジスタ値で制御されます。
DDOV	方向値交換値	DDOE=1時、DDxnレジスタ値に関係なく、出力駆動部のON(1)/OFF(0)を制御します。
PVOE	出力値交換許可	1で出力駆動部がONなら、ポート値はPVOV信号で制御されます。出力駆動部がONで0の場合、ポート値はPORTxnレジスタ値で制御されます。
PVOV	出力値交換値	PVOE=1時、PORTxnレジスタ値に関係なく、ポート値を制御(1/0)します。
PTOE	出力切替許可	PTOE=1時、PORTxnレジスタ値が反転します。
DIEOE	デジタル入力許可交換許可	1で、デジタル入力許可はDIEOV信号で制御され、0の場合、MCUの状態(活動動作、休止形態)によって決定されます。
OIEOV	デジタル入力許可交換値	DIEOE=1時、MCUの状態(活動動作、休止形態)に関係なく、デジタル入力を許可(1)/禁止(0)します。
DI	デジタル入力	交換機能用デジタル入力です。この信号は図上でシュミットトリガ出力に接続されていますが、これは同期化前となります。この信号はクロックとしての使用を除き、各交換機能自身が同期化します。
AIO	アナログ入出力	交換機能用アナログ入出力です。この信号はピンに直接接続され、双方向使用ができます。

以降の項は交換機能に関連する重複(交換)信号と各ポートの交換機能を簡単に記述します。更に先の詳細については交換機能の記述を参照してください。

MCU制御レジスタ (MCU Control Register) MCUCR

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	PUD	SM1	SE	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – PUD : プルアップ禁止 (Pull-up Disable)

このビットが1を書かれると、例えばDDxnとPORTxnレジスタがプルアップを許可(DDxn=0, PORTxn=1)に設定されていても、I/Oポートのプルアップは禁止されます。この特徴についてより多くの詳細に関しては31頁の「[ピンの設定](#)」をご覧ください。

ポートAの交換機能

ポートAピンの交換機能は表23.で示されます。

表23. ポートAピンの交換機能

ポートピン	交換機能
PA2	RESET (外部リセット入力) dW (デバッグWIRE入出力)
PA1	XTAL2 (システムクロック用発振増幅器出力)
PA0	XTAL1 (システムクロック用発振増幅器入力または外部クロック信号入力)

交換ピンの設定は次のとおりです。(訳注:共通性のため本項追加)

•RESET – ポートAビット2 : PA2

RESET : リセットピン。RSTDISBLヒューズがプログラム(0)されると、このピンは通常のI/Oピンとして機能し、デバイスがリセット元として電源ONリセットと低電圧リセットに頼らなければなりません。RSTDISBLヒューズが非プログラム(1)にされると、このピンにリセット回路が接続され、このピンはI/Oピンとして使えません。

PA2がリセットピンとして使われると、PORTA2, DDA2, PINA2は全て0を読みます。

dW : デバッグWIRE双方向通信ピン。デバッグWIREによる内蔵デバッグ時の双方向通信に使います。dWピンとして使われると、このピンは他の機能ピンとして使えません。

•XTAL2 – ポートAビット1 : PA1

XTAL2 : チップ(システム)クロック発振器ピン2。クリスタル用発振器クロックピンとして使われます。クロックピンとして使われると、このピンはI/Oピンとして使えません。

PA1がクロックピンとして使われると、PORTA1, DDA1, PINA1は全て0を読みます。

•XTAL1 – ポートAビット0 : PA0

XTAL1 : チップ(システム)クロック発振器ピン1。クリスタル用発振器クロックピンまたは外部クロック入力ピンとして使われます。これらのピンとして使われると、このピンはI/Oピンとして使えません。

PA0がクロックピンとして使われると、PORTA0, DDA0, PINA0は全て0を読みます。

ポートBの交換機能

ポートBピンの交換機能は表24.で示されます。

表24. ポートBピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PB7	USCK (3線動作USIクロック入出力) SCL (2線動作USI(TWI)クロック入出力) PCINT7 (ピン変化割り込み7入力)	PB3	OC1A (タイマ/カウンタ1 比較A一致/PWM出力) PCINT3 (ピン変化割り込み3入力)
PB6	DO (3線動作USIデータ出力) PCINT6 (ピン変化割り込み6入力)	PB2	OC0A (タイマ/カウンタ0 比較A一致/PWM出力) PCINT2 (ピン変化割り込み2入力)
PB5	DI (3線動作USIデータ入力) SDA (2線動作USI(TWI)データ入出力) PCINT5 (ピン変化割り込み5入力)	PB1	AIN1 (アナログ比較器反転入力) PCINT1 (ピン変化割り込み1入力)
PB4	OC1B (タイマ/カウンタ1 比較B一致/PWM出力) PCINT4 (ピン変化割り込み4入力)	PB0	AIN0 (アナログ比較器非反転入力) PCINT0 (ピン変化割り込み0入力)

交換ピンの設定は次のとおりです。

• USCK/SCL/PCINT4 – ポートB ビット7 : PB7

USCK : 3線動作USIのクロック入出力。

SCL : 2線動作USI(TWI)のクロック入出力。

PCINT7 : ピン変化割り込み7入力。PB7ピンは外部割り込み元としても扱えます。

• DO/PCINT6 – ポートB ビット6 : PB6

DO : 3線動作USIのデータ出力。

PCINT6 : ピン変化割り込み6入力。PB6ピンは外部割り込み元としても扱えます。

• DI/SDA/PCINT5 – ポートB ビット5 : PB5

DI : 3線動作USIのデータ入力。3線動作USIは標準ポート機能を無効にしないので、ピンは入力として設定されなければなりません。

SDA : 2線動作USI(TWI)のデータ入出力。

PCINT5 : ピン変化割り込み5入力。PB5ピンは外部割り込み元としても扱えます。

• OC1B/PCINT4 – ポートB ビット4 : PB4

OC1B : タイマ/カウンタ1の比較B一致出力。PB4ピンはタイマ/カウンタ1の比較B一致用外部出力として扱えます。この機能を扱うにはこのピンが出力として設定(DDB4=1)されなければなりません。このOC1Bピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT4 : ピン変化割り込み4入力。PB4ピンは外部割り込み元としても扱えます。

• OC1A/PCINT3 – ポートB ビット3 : PB3

OC1A : タイマ/カウンタ1の比較A一致出力。PB3ピンはタイマ/カウンタ1の比較A一致用外部出力として扱えます。この機能を扱うにはこのピンが出力として設定(DDB3=1)されなければなりません。このOC1Aピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT3 : ピン変化割り込み3入力。PB3ピンは外部割り込み元としても扱えます。

• OC0A/PCINT2 – ポートB ビット2 : PB2

OC0A : タイマ/カウンタ0の比較A一致出力。PB2ピンはタイマ/カウンタ0の比較A一致用外部出力として扱えます。この機能を扱うにはこのピンが出力として設定(DDB2=1)されなければなりません。このOC0Aピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT2 : ピン変化割り込み2入力。PB2ピンは外部割り込み元としても扱えます。

• AIN1/PCINT1 – ポートB ビット1 : PB1

AIN1 : アナログ比較器の反転入力。アナログ比較器の機能を妨げるデジタルポート機能を避けるため、内部プルアップをOFFにした入力としてポートピンを設定してください。

PCINT1 : ピン変化割り込み1入力。PB1ピンは外部割り込み元としても扱えます。

• AIN0/PCINT0 – ポートB ビット0 : PB0

AIN0 : アナログ比較器の非反転入力。アナログ比較器の機能を妨げるデジタルポート機能を避けるため、内部プルアップをOFFにした入力としてポートピンを設定してください。

PCINT0 : ピン変化割り込み0入力。PB0ピンは外部割り込み元としても扱えます。

表25.と表26.はポートBの交換機能を33頁の図25.で示される交換信号に関連付けます。(訳注:原書無効行削除)

表25. ポートB7~4の交換機能用交換信号

信号名	PB7/USCK/SCL/PCINT7	PB6/DO/PCINT6	PB5/DI/SDA/PCINT5	PB4/OC1B/PCINT4
PUOE	2線USI	0	0	0
PUOV	0	0	0	0
DDOE	2線USI	0	2線USI	0
DDOV	(SCL保持+PORTB7)+DDB7	0	(SDA+PORTB5)・DDB5	0
PVOE	2線USI・DDB7	3線USI	2線USI・DDB5	OC1B許可
PVOV	0	DO	0	OC1B
PTOE	USI許可	0	0	0
DIEOE	(PCIE・PCINT7)+USISIE	PCIE・PCINT6	(PCIE・PCINT5)+USISIE	PCIE・PCINT4
DIEOV	1	1	1	1
DI	USCK/SCL/PCINT7入力	PCINT6入力	DI/SDA/PCINT5入力	PCINT4入力
AIO	-	-	-	-

表26. ポートB3~0の交換機能用交換信号 (訳注:矛盾点を一部修正)

信号名	PB3/OC1A/PCINT3	PB2/OC0A/PCINT2	PB1/AIN1/PCINT1	PB0/AIN0/PCINT0
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	OC1A許可	OC0A許可	0	0
PVOV	OC1A	OC0A	0	0
PTOE	0	0	0	0
DIEOE	PCIE・PCINT3	PCIE・PCINT2	(PCIE・PCINT1)+AIN1D	(PCIE・PCINT0)+AIN0D
DIEOV	1	1	1	1
DI	PCINT3入力	PCINT2入力	PCINT1入力	RXD/PCINT0入力
AIO	-	-	AIN1入力	AIN0入力

注: AIN0DとAIN1Dは101頁の「デジタル入力禁止レジスタ - DIDR」で記述されます。

ポートDの交換機能

ポートDピンの交換機能は表27.で示されます。

表27. ポートDピンの交換機能

ポートピン	交換機能
PD6	ICP (タイマ/カウンタ1 捕獲起動入力)
PD5	OC0B (タイマ/カウンタ0 比較B一致/PWM出力) T1 (タイマ/カウンタ1 外部クロック入力)
PD4	T0 (タイマ/カウンタ0 外部クロック入力)
PD3	INT1 (外部割り込み1 入力)
PD2	INT0 (外部割り込み0 入力) XCK (USART 外部クロック入出力) CKOUT (システムクロック出力)
PD1	TXD (USART 送信データ出力)
PD0	RXD (USART 受信データ入力)

交換ピンの設定は次のとおりです。

- ICP – ポートD ビット6 : PD6

ICP : タイマ/カウンタ1の捕獲起動入力。PD6ピンはタイマ/カウンタ1用捕獲起動入力ピンとして動作できます。

- OC0B/T1 – ポートD ビット5 : PD5

OC0B : タイマ/カウンタ0の比較B一致出力。PD5ピンはタイマ/カウンタ0の比較B一致用外部出力として扱えます。この機能を扱うにはこのピンが出力として設定(DDD5=1)されなければなりません。このOC0Bピンはタイマ機能のPWM動作出力ピンでもあります。

T1 : タイマ/カウンタ1外部クロック入力はタイマ/カウンタ1制御レジスタ(TCCR1B)のCS12,1ビットを設定(1)することによって許可されます。

- T0 – ポートD ビット4 : PD4

T0 : タイマ/カウンタ0外部クロック入力はタイマ/カウンタ0制御レジスタ(TCCR0B)のCS02,1ビットを設定(1)することによって許可されます。

- INT1 – ポートD ビット3 : PD3

INT1 : 外部割り込み1入力。PD3ピンはMCUへの外部割り込み元として扱えます。

- INT0/XCK/CKOUT – ポートD ビット2 : PD2

INT0 : 外部割り込み0入力。PD2ピンはMCUへの外部割り込み元として扱えます。

XCK : USARTの外部クロック入出力。ポートD方向レジスタ(DDRD)のDDD2は、このクロックが入力(DDD2=0)または出力(DDD2=1)のどちらかを制御します。XCKピンはUSARTが同期種別で動作する時だけ活動します(有効です)。

CKOUT : システムクロック出力。分周したシステムクロックはPD2ピンに出力できます。分周したシステムクロックはCKOUTピンのプルアップがプログラム(0)されると、PORTD2とDDD2設定に拘らず、出力されます。これはリセット中にも出力されます。

- TXD – ポートD ビット1 : PD1

TXD : 送信データ(USART用データ出力ピン)。USART送信部が許可されると、ピンはDDRDのDDD1値に拘らず出力として設定されます。

- RXD – ポートD ビット0 : PD0

RXD : 受信データ(USART用データ入力ピン)。USART受信部が許可されると、このピンはDDRDのDDD0の値に拘らず入力として設定されます。USARTがこのピンを入力に強制するとき、プルアップは未だPORTD0ビットによって制御できます。

表28.と表29.はポートDの交換機能を33頁の図25.で示される交換信号に関連付けます。

表28. ポートD6~4の交換機能用交換信号

信号名	PD6/ICP	PD5/OC1B/T1	PD4/T0
PUOE	0	0	0
PUOV	0	0	0
DDOE	0	0	0
DDOV	0	0	0
PVOE	0	OC0B許可	0
PVOV	0	OC0B	0
PTOE	0	0	0
DIOE	ICP許可	T1許可	T0許可
DIOV	1	1	1
DI	ICP入力	T1入力	T0入力
AIO	-	-	-

表29. ポートD3~0の交換機能用交換信号

信号名	PD3/INT1	PD2/INT0/XCK/CKOUT	PD1/TXD	PD0/RXD
PUOE	0	0	TXEN	RXEN
PUOV	0	0	0	PORTD0・ $\overline{\text{PUD}}$
DDOE	0	0	TXEN	RXEN
DDOV	0	0	1	0
PVOE	0	XCK出力許可	TXEN	0
PVOV	0	XCK	TXD	0
PTOE	-	0	0	0
DIOE	INT1許可	INT0許可・XCK入力設定	0	0
DIOV	1	1	0	0
DI	INT1入力	INT0/XCK入力	-	RXD入力
AIO	-	-	-	-

I/Oポート用レジスタ

ポートA出力レジスタ (Port A Data Register) PORTA

ビット	7	6	5	4	3	2	1	0	
\$1B (\$3B)	-	-	-	-	-	PORTA2	PORTA1	PORTA0	PORTA
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートA方向レジスタ (Port A Data Direction Register) DDRA

ビット	7	6	5	4	3	2	1	0	
\$1A (\$3A)	-	-	-	-	-	DDA2	DDA1	DDA0	DDRA
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートA入力レジスタ (Port A Input Address) PINA

ビット	7	6	5	4	3	2	1	0	
\$19 (\$39)	-	-	-	-	-	PINA2	PINA1	PINA0	PINA
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

ポートB出力レジスタ (Port B Data Register) PORTB

ビット	7	6	5	4	3	2	1	0	
\$18 (\$38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートB方向レジスタ (Port B Data Direction Register) DDRB

ビット	7	6	5	4	3	2	1	0	
\$17 (\$37)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDR B
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートB入力レジスタ (Port B Input Address) PINB

ビット	7	6	5	4	3	2	1	0	
\$16 (\$36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

ポートD出力レジスタ (Port D Data Register) PORTD

ビット	7	6	5	4	3	2	1	0	
\$12 (\$32)	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートD方向レジスタ (Port D Data Direction Register) DDRD

ビット	7	6	5	4	3	2	1	0	
\$11 (\$31)	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDR D
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ポートD入力レジスタ (Port D Input Address) PIND

ビット	7	6	5	4	3	2	1	0	
\$10 (\$30)	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	



外部割り込み

外部割り込みはINT0,INT1ピンまたはPCINT0~7ピンの何れかによって起動されます。許可したなら、例えばINT0,1またはPCINT0~7ピンが出力として設定されても、割り込みが起動することに注目してください。この特徴はソフトウェア割り込みを生成する方法を提供します。ピン変化割り込みPCIは許可したPCINT0~7の何れかが切り替わると起動します。ピン変化割り込み許可レジスタ(PCMSK)は、どのピンがピン変化割り込み要因となるかを制御します。PCINT0~7でのピン変化割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイスを起動するのに使えることを意味します。

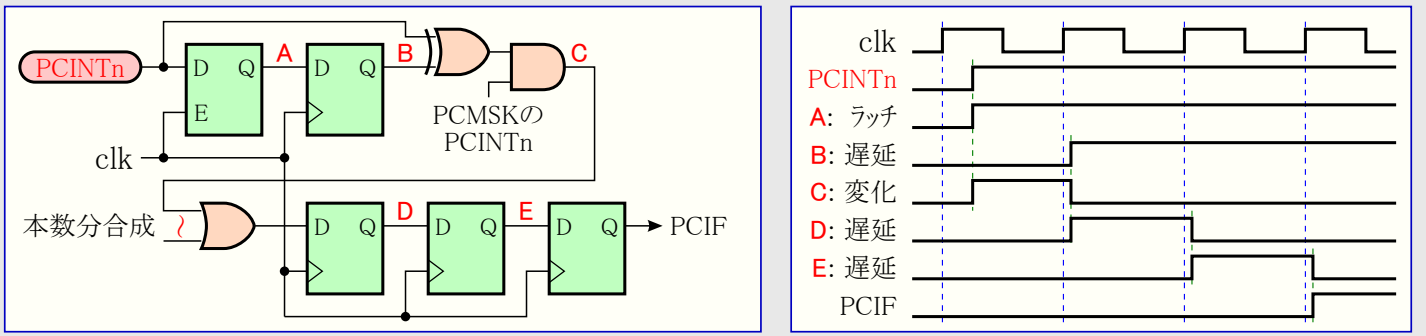
INT0とINT1割り込みは上昇端または下降端(含む両端)、またはLowレベルによって起動できます。これはMCU制御レジスタ(MCUICR)の詳述で示される設定です。INT0またはINT1割り込みがレベル起動として設定、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT0またはINT1の上昇端または下降端割り込みの認知は16頁の「クロックシステムとその配給」で記述されるI/Oクロックの存在を必要とすることに注意してください。INT0とINT1のLowレベル割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイスを起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

レベル起動割り込みが「パワーダウン」動作からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが起動時間の最後に先立って消滅すると、MCUは今までどおり起動しますが、割り込みが生成されません。起動時間は16頁の「システムクロックとクロック選択」で示されるようにSUTヒューズとCKSELヒューズで定義されます。

ピン変化割り込みタイミング

ピン変化割り込みの例は図26.で示されます。

図26. ピン変化割り込みタイミング



MCU制御レジスタ (MCU Control Register) MCUICR

MCU制御レジスタは割り込み条件制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	PUD	SM1	SE	SM0	ISC11	ISC10	ISC01	ISC00	MCUICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット3,2 – ISC11,0 : 外部割り込み1条件制御 (Interrupt Sense Control 1 bit1 and 0)

外部割り込み1はステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み許可レジスタ(GIMSK)の外部割り込み1許可(INT1)ビットが設定(1)される場合のINT1外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT1ピンの端(エッジ)とレベルは表30.で定義されます。INT1ピンの値は端検出に先立って採取されます。端または論理変化割り込みが選ばれる場合、1クロック周期よりも長く留まるパルスは割り込みを生成します。より短いパルスは割り込み発生が保証されません。Lowレベル割り込みが選ばれる場合、そのLowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表30. 外部割り込み1(INT1)割り込み条件

ISC11	ISC10	割り込み発生条件
0	0	INT1ピンのLowレベル
0	1	INT1ピンの論理変化(両端)
1	0	INT1ピンの下降端
1	1	INT1ピンの上昇端

• ビット1,0 – ISC01,0 : 外部割り込み0条件制御 (Interrupt Sense Control 0 bit1 and 0)

外部割り込み0はステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み許可レジスタ(GIMSK)の外部割り込み0許可(INT0)ビットが設定(1)される場合のINT0外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT0ピンの端(エッジ)とレベルは表31.で定義されます。INT0ピンの値は端検出に先立って採取されます。端または論理変化割り込みが選ばれる場合、1クロック周期よりも長く留まるパルスは割り込みを生成します。より短いパルスは割り込み発生が保証されません。Lowレベル割り込みが選ばれる場合、そのLowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表31. 外部割り込み0(INT0)割り込み条件

ISC01	ISC00	割り込み発生条件
0	0	INT0ピンがLowレベル
0	1	INT0ピンの論理変化(両端)
1	0	INT0ピンの下降端
1	1	INT0ピンの上昇端

一般割り込み許可レジスタ (General Interrupt Mask Register) GIMSK

ビット	7	6	5	4	3	2	1	0	
\$3B (\$5B)	INT1	INT0	PCIE	-	-	-	-	-	GIMSK
Read/Write	R/W	R/W	R/W	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – INT1 : 外部割り込み1許可 (External Interrupt Request 1 Enable)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、INT1ビットが設定(1)されると、INT1外部ピン割り込みが許可されます。MCU制御レジスタ(MCUCR)の割り込み条件制御1のビット1と0(ISC11,0)はこの外部割り込みがINT1ピンの上昇端、下降端、両端、またはLowレベルのどれで活性(有効)にされるかを定義します。例えばINT1ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求1に対応する割り込みはINT1割り込みベクタから実行されます。

•ビット6 – INT0 : 外部割り込み0許可 (External Interrupt Request 0 Enable)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、INT0ビットが設定(1)されると、INT0外部ピン割り込みが許可されます。MCU制御レジスタ(MCUCR)の割り込み条件制御0のビット1と0(ISC01,0)はこの外部割り込みがINT0ピンの上昇端、下降端、両端、またはLowレベルのどれで活性(有効)にされるかを定義します。例えばINT0ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求0に対応する割り込みはINT0割り込みベクタから実行されます。

•ビット5 – PCIE : ピン変化割り込み許可 (Pin Change Interrupt Enable)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIEビットが設定(1)されると、ピン変化割り込みが許可されます。許可したPCINT0~7ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みはPCI割り込みベクタから実行されます。PCINT0~7ピンはピン変化割り込み許可レジスタ(PCMSK)によって個別に許可されます。

外部割り込み要求フラグ レジスタ (External Interrupt Flag Register) EIFR

ビット	7	6	5	4	3	2	1	0	
\$3A (\$5A)	INTF1	INTF0	PCIF	-	-	-	-	-	EIFR
Read/Write	R/W	R/W	R/W	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – INTF1 : 外部割り込み1要求フラグ (External Interrupt Flag 1)

INT1ピン上の端(エッジ)または論理変化が割り込み要求を起動すると、INTF1が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み許可レジスタ(GIMSK)の外部割り込み1許可(INT1)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。INT1がレベル割り込みとして設定されると、このフラグは常に解除(0)されます。

•ビット6 – INTF0 : 外部割り込み0要求フラグ (External Interrupt Flag 0)

INT0ピン上の端(エッジ)または論理変化が割り込み要求を起動すると、INTF0が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み許可レジスタ(GIMSK)の外部割り込み0許可(INT0)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。INT0がレベル割り込みとして設定されると、このフラグは常に解除(0)されます。

•ビット5 – PCIF : ピン変化割り込み要求フラグ (Pin Change Interrupt Flag)

PCINT0~7ピンの何れかの論理変化が割り込み要求を起動すると、PCIFが設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み許可レジスタ(GIMSK)のピン変化割り込み許可(PCIE)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

ピン変化割り込み許可レジスタ (Pin Change Mask) PCMSK

ビット	7	6	5	4	3	2	1	0	
\$20 (\$40)	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	PCMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7~0 – PCINT7~PCINT0 : ピン変化割り込み7~0許可 (Pin Change Enable Mask 7~0)

各PCINT0~7ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT0~7と一般割り込み許可レジスタ(GIMSK)のPCIEが設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT0~7が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

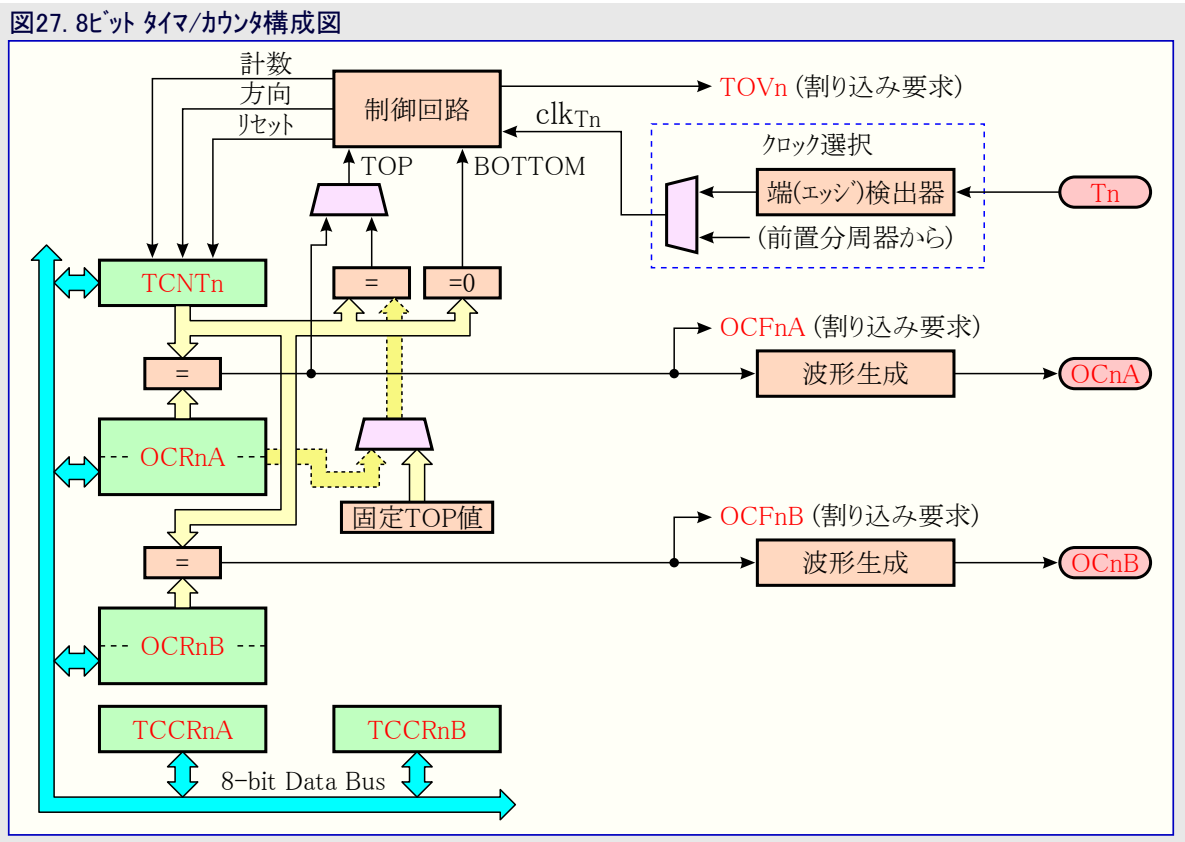
8ビット タイマ/カウンタ0 (PWM付き)

タイマ/カウンタ0は2つの独立した比較出力部とPWM支援付きの汎用8ビット タイマ/カウンタ部です。それは正確なプログラム実行タイミング(事象管理)、波形生成を許します。主な特徴は次に示されます。

- 2つの独立した比較出力部
- 2重緩衝の比較レジスタ
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 可変PWM周期
- 周波数発生器
- 3つの独立した割り込み (TOV0,OCF0A,OCF0B)

概要

この8ビット タイマ/カウンタの単純化した構成図は図27.で示されます。I/Oピンの実際の配置については2頁の「ピン配置」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注:原文は太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は50頁の「8ビット タイマ/カウンタ用レジスタ」で一覧されます。



関係レジスタ

タイマ/カウンタ(TCNT0)と比較レジスタ(OCR0AとOCR0B)は8ビットのレジスタです。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKは本図で示されません。

このタイマ/カウンタは内部的、前置分周器経由、またはT0ピンの外部クロック元によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加(または減少)するのに使うクロック元を制御します。クロック元が選ばれないと、タイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタ クロック(clkT0)として参照されます。

2重緩衝化した比較レジスタ(OCR0AとOCR0B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC0AとOC0B)ピンでPWMまたは可変周波数出力を作成するための波形生成器によって使えます。詳細については44頁の「比較出力部」をご覧ください。この比較一致発生は比較一致割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF0AとOCF0B)も設定(1)します。

定義

本章でのレジスタとビット参照の多くは一般形で書かれます。小文字の'n'はタイマ/カウンタ番号、この場合は0で置き換えます。小文字のxは比較出力部のチャンネル名を表し、この場合はAまたはBです。然しながらプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えばタイマ/カウンタ0のカウンタ値のアクセスに対してのTCNT0のように)。

表32.の定義は本文書を通して広範囲に渡って使われます。

表32. 用語定義

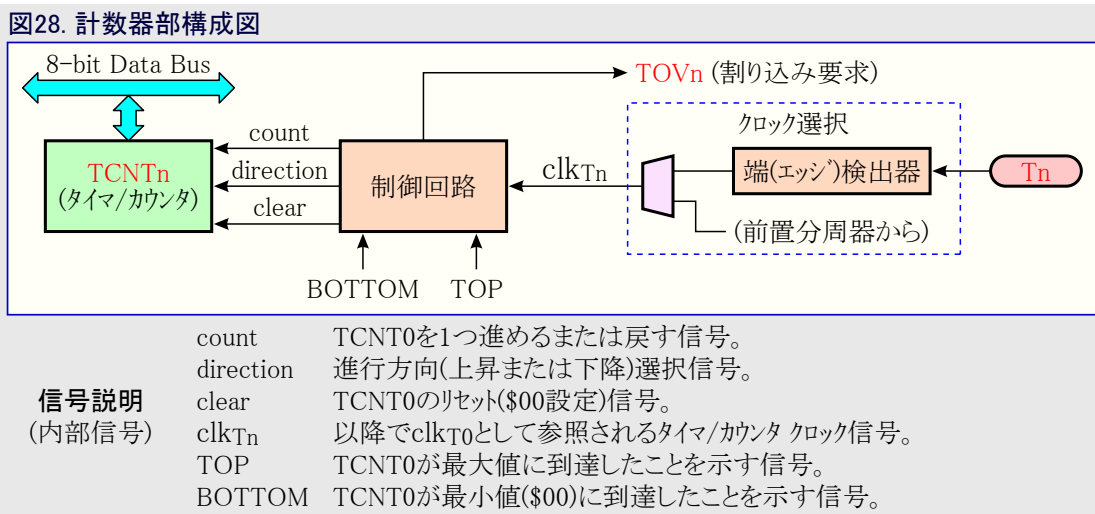
用語	意味
BOTTOM	タイマ/カウンタが\$00に到達した時。
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0A値に到達した時。この指定(TOP)値は動作種別に依存します。

タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR0B)に配置されたクロック選択(CS02~0)ビットによって制御されるクロック選択論理回路で選ばれます。クロック元と前置分周器の詳細については54頁の「タイマ/カウンタ0とタイマ/カウンタ1の前置分周器」をご覧ください。

計数器部

8ビット タイマ/カウンタの主な部分は設定可能な双方向カウンタ部です。図28.は、このカウンタとその周辺環境の構成図を示します。



使った動作種別に依存して、カウンタは各タイマ/カウンタ クロック(clkT0)で解除(\$00)、増加(+1)、または減少(-1)されます。clkT0はクロック選択(CS02~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS02~0=000)と、タイマ/カウンタは停止されます。けれどもTCNT0値はタイマ/カウンタ クロック(clkT0)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先順位を持ちます)。

計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCR0A)に配置された波形生成種別(WGM01,0)ビットとタイマ/カウンタ制御レジスタB(TCCR0B)に配置された波形生成種別(WGM02)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOCR0A/OCR0B比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に関しては46頁の「動作種別」をご覧ください。

タイマ/カウンタ溢れ(TOV0)フラグはWGM02~0ビットによって選ばれた動作種別に従って設定(1)されます。TOV0はCPU割り込み発生に使えます。

比較出力部

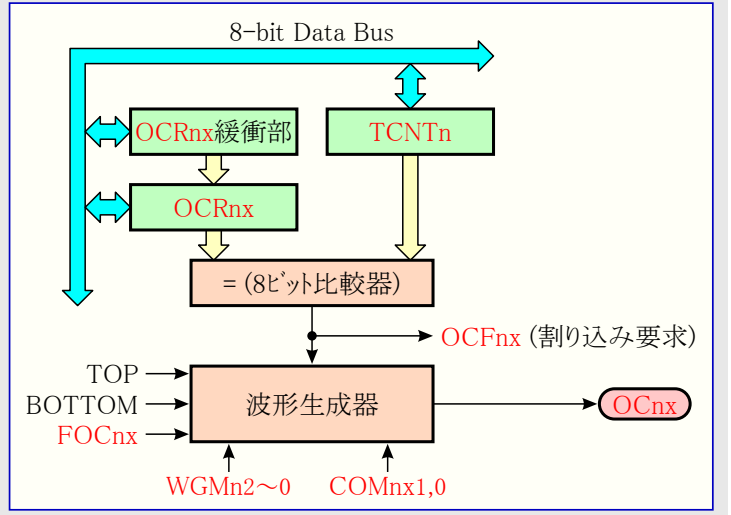
この8ビット比較器はTCNT0と比較レジスタ(OCR0AとOCR0B)を継続的に比較します。TCNT0がOCR0AまたはOCR0Bと等しければ、比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF0AまたはOCF0B)を設定(1)します。対応する割り込みが許可(I=1, OCIE0AまたはOCIE0B=1)されているならば、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットによって設定された動作種別に従った出力を生成するのにこの一致信号を使います。MAXとBOTTOM信号は動作種別(46頁の「動作種別」参照)のいくつかで両端値の特別な場合を扱うため、波形生成器によって使われます。

図29.は比較出力部の構成図を示します。

OCR0xはパルス幅変調(PWM)のどれかを使う時に2重緩衝化されます。標準動作と比較一致タイマ/カウンタ解除(CTC)動作については2重緩衝動作が禁止されます。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR0xレジスタの更新を同期化します。この同期化は奇数長、非対称PWMパルスの発生を防ぎ、それによって不具合なしの出力を作成します。

OCR0xのアクセスは複雑なように思えますが決してそんなことはありません。2重緩衝動作が許可されるとCPUはOCR0x緩衝部をアクセスし、禁止されるとOCR0xレジスタを直接アクセスします。

図29. 比較出力部構成図



(訳注) ここでは比較レジスタ全体をOCR0x、OCR0xを構成する緩衝部分をOCR0x緩衝部、実際の比較に使われるレジスタ本体部分をOCR0xレジスタとして記述しています。他の部分での記述でも特に必要がある場合はこの記述方法を適用します。

強制比較出力

非PWM波形生成動作での比較器の一致出力は、強制変更(FOC0x)ビットに1を書くことによって強制(変更)できます。比較一致の強制は比較割り込み要求フラグ(OCF0x)の設定(1)やタイマ/カウンタの再設定/解除を行いませんが、OC0xピンは実際の比較一致が起きた場合と同様に更新されます(COM0x1,0ビット設定がOC0xピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

TCNT0書き込みによる比較一致妨害

TCNT0への全てのCPU書き込みは、例えタイマ/カウンタが停止されていても、次のタイマ/カウンタクロック周期で起こるどんな比較一致をも妨げます。この特質はタイマ/カウンタクロックが許可されている時に割り込みを起動することなく、TCNT0と同じ値に初期化されることをOCR0xに許します。

比較一致部の使用

どの動作種別でのTCNT0書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、比較出力部を使う場合、TCNT0を変更する時に危険を伴います。TCNT0に書かれた値がOCR0x値と同じ場合に比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。同様にタイマ/カウンタが下降計数のとき、BOTTOMに等しいTCNT0値を書いてはいけません。

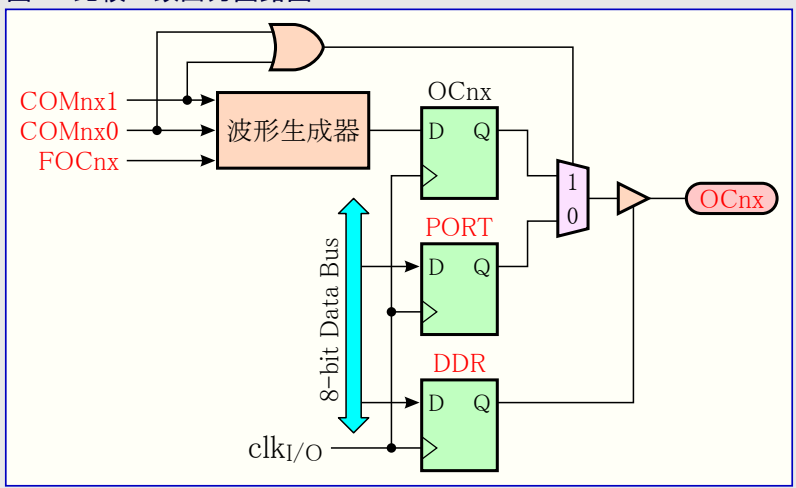
OC0xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC0x値を設定する一番簡単な方法は標準動作で強制変更(FOC0x)スローブビットを使うことです。波形生成動作種別間を変更する時でも、OC0x(内部)レジスタはその値を保ちます。

比較出力選択(COM0x1,0)ビットが比較値(OCR0x)と共に2重緩衝されないことに気付いてください。COM0x1,0ビットの変更は直ちに有効となります。

比較一致出力部

比較出力選択($COM0x1,0$)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力($OC0x$)状態の定義に $COM0x1,0$ ビットを使います。また $COM0x1,0$ ビットは $OC0x$ ピン出力元を制御します。図30.は $COM0x1,0$ ビット設定によって影響を及ぼされる論理回路の簡易化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。 $COM0x1,0$ ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。 $OC0x$ の状態を参照するとき、その参照は $OC0x$ ピンでなく内部 $OC0x$ レジスタに対してです。システムリセットが起ると、 $OC0x$ レジスタは'0'にリセットされます。

図30. 比較一致出力回路図



$COM0x1,0$ ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力($OC0x$)によって無効にされます。けれども $OC0x$ ピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。 $OC0x$ ピンに対するポート方向レジスタのビット(DDR_OC0x)は $OC0x$ 値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は波形生成種別と無関係です。

比較出力ピン論理回路の設計は出力が許可される前の $OC0x$ 状態の初期化を許します。いくつかの $COM0x1,0$ ビット設定が或る種の動作種別に対して予約されることに注意してください。50頁の「8ビットタイマ/カウンタ0用レジスタ」をご覧ください。

比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作で $COM0x1,0$ ビットを違うふうに使います。全ての動作種別に対して $COM0x1,0=00$ 設定は次の比較一致で実行すべき $OC0x$ レジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については50頁の表33.と表36.を参照してください。高速PWM動作については50頁の表34.と表37.、位相基準PWMについては50頁の表35.と表38.を参照してください。

$COM0x1,0$ ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(FOC0x)ストローブビットを使うことによって直ちに効果を得ることを強制できます。

動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットの組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM0x1,0ビットは生成されるPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対するCOM0x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します(45頁の「比較一致出力部」をご覧ください)。

タイミング情報の詳細については49頁の「タイマ/カウンタのタイミング」を参照してください(訳注:原文中の図番号省略)。

標準動作

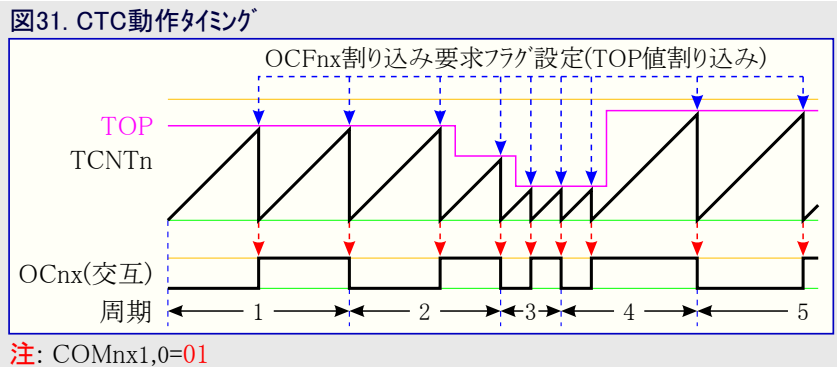
最も単純な動作種別が標準動作(WGM02~0=000)です。この動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV0)フラグはTCNT0が\$00になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV0フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのようになります。けれどもTOV0フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解能はソフトウェアによって増やせます。標準動作での考慮に特別な場合はなく、新しいカウンタ値は何時でも書けます。

比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM02~0=010)ではOCR0Aがカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT0)値がOCR0Aと一致すると、カウンタは\$00に解除されます。OCR0Aはカウンタに対するTOP値、従って分解能も定義します。この動作種別はより大きい比較一致出力周波数の制御を許します。それは外部の出来事の計数操作も簡単にします。

CTC動作についてのタイミング図は図31.で示されます。カウンタ(TCNT0)値はTCNT0とOCR0A間で比較一致が起こるまで増加し、そしてその後カウンタ(TCNT0)は解除(\$00)されます。



OCF0Aフラグを使うことにより、タイマ/カウンタ値がTOP値に達する時毎に割り込みが生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。けれども前置分周なしまたは低い前置分周値でカウンタが走行している時にBOTTOMと近い値にTOPを変更することは、CTC動作が2重緩衝機能を持たないために注意して行わなければなりません。OCR0Aに書かれた新しい値がTCNT0の現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こるのに先立って最大値(\$FF)へ、そして次に\$00から始める計数をしなければならないでしょう。

CTC動作で波形出力を生成するため、OC0A出力は比較出力選択(COM0A1,0)ビットを交互動作(=01)に設定することによって各比較一致での論理レベル交互切り替えに設定できます。OC0A値はそのピンに対するデータ方向が出力(DDR_OC0A=1)に設定されない限りポートピンで見えないでしょう。生成された波形はOCR0Aが0(\$00)に設定されるとき、 $f_{OC0A} = f_{clk_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

$$f_{OCnx} = \frac{f_{clk_I/O}}{2 \times N \times (1 + OCRnx)}$$

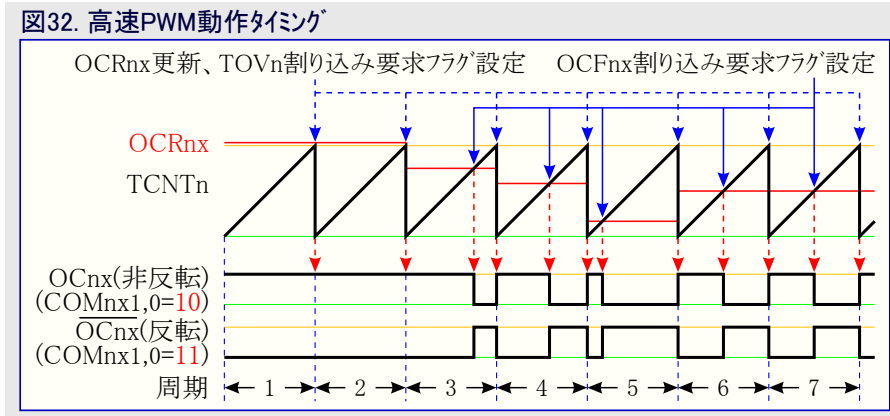
変数Nは前置分周数(1,8,64,256,1024)を表します。

標準動作と同じように、タイマ/カウンタ溢れ(TOV0)フラグはカウンタがMAXから\$00へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

高速PWM動作

高速パルス幅変調(PWM)動作(WGM02~0=011または111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。TOPはWGM02~0=011時に\$FF、WGM02~0=111時にOCR0Aとして定義されます。非反転比較出力動作(COM0x1,0=10)での比較出力(OC0x)はTCNT0とOCR0x間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM0x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。この単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がTOP値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は図32.で示されます。TCNT0値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。赤細線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF0x)はOCR0x=TOPを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。



タイマ/カウンタ溢れ(TOV0)フラグはカウンタがTOPに到達する時毎に設定(1)されます。割り込みが許可されるなら、その割り込み処理ルーチンは比較値を更新するのに使えます。

高速PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0を'11'に設定することで生成できます。WGM02ビットが設定(1)ならば、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(50頁の表34.と表37.をご覧ください)。実際のOC0x値はポートピンに対するデータ方向(DDR_OC0x)が出力として設定される場合だけ見えてでしょう。PWM波形はTCNT0とOCR0x間の比較一致で、OC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$00、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0xレジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

$$f_{OCnxPWM} = \frac{f_{clk_I/O}}{N \times (1 + TOP)}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

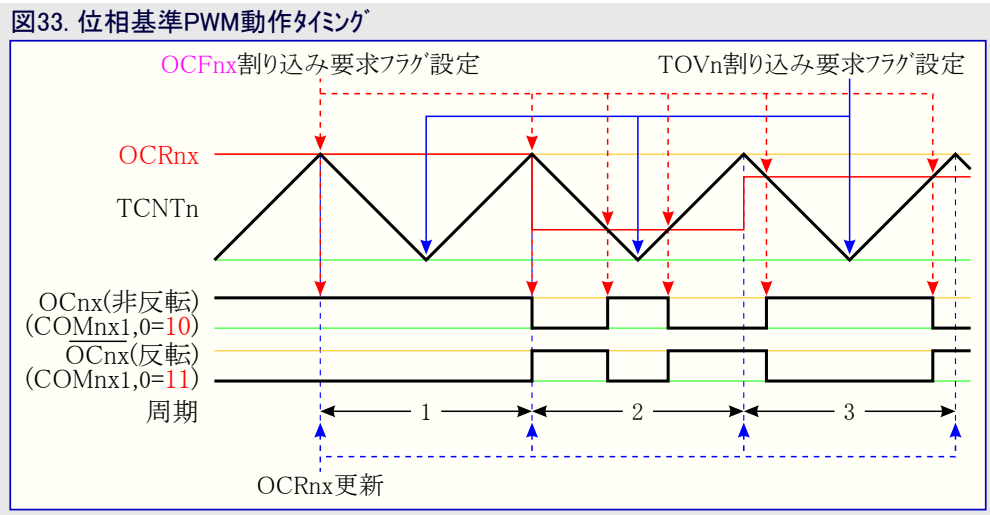
OCR0xの両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合にあたります。OCR0xがBOTTOM(\$00)と等しく設定されると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。OCR0xがTOPに等しく設定されると、(COM0x1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

(訳補:WGM02~0=111の場合については、)高速PWM動作での(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC0A設定(COM0A1,0=01)によって達成できます。生成された波形はOCR0Aが0(\$00)に設定される時に $f_{OC0x} = f_{clk_I/O} / 2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いて、CTC動作でのOC0A交互出力(COM0A1,0=01)と同じです。

位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM02~0=001または101)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とします。カウンタはBOTTOMからTOPへそして次にTOPからBOTTOMへを繰り返し計数します。TOPはWGM02~0=001時に\$FF、WGM02~0=101時にOCR0xとして定義されます。非反転比較出力動作(COM0x1,0=10)での比較出力(OC0x)は上昇計数中のTCNT0とOCR0xの比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM0x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準PWM動作でのカウンタはカウンタ値がTOPと一致するまで増加されます。カウンタはTOPに達すると計数方向を変更します。このTCNT0値は1タイマ/カウンタクロック周期間TOPと等しくなります。位相基準PWM動作のタイミング図は図33.で示されます。TCNT0値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。



タイマ/カウンタ溢れ(TOV0)フラグはタイマ/カウンタがBOTTOMに到達する時毎に設定(1)されます。この割り込み要求フラグはカウンタがBOTTOM値に到達する毎に割り込みを発生するために使えます。

位相基準PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0ビットを'11'に設定することで生成できます。WGM02ビットが設定(1)なら、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(50頁の表35.と表38.をご覧ください)。実際のOC0x値はそのポートピンに対するデータ方向(DDR_OC0x)が出力として設定される場合だけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT0とOCR0x間の比較一致でOC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT0とOCR0x間の比較一致でOC0xレジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

$$f_{OCnxPCPWM} = \frac{f_{clk_I/O}}{2 \times N \times TOP}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

OCR0xの両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR0xがBOTTOM(\$00)に等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。

図33.の第2周期のその出発点に於いて、例えば比較一致がないとしても、OCnxにはHighからLowへの遷移があります。この遷移点はBOTTOMを挟む対称を保証するためです。比較一致なしに遷移を生ずるのは2つの場合です。

- 図33.のようにOCR0xはTOPからその値を変更します。OCR0x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0x値よりも高い値から数え始め、そしてその理屈のために比較一致、それ故上昇途中で起こされるであろうOCnxの変更を逃します。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)

タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従って以下の図でタイマ/カウンタクロック(clk_{T0})がクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時の情報を含みます。図34は基本的なタイマ/カウンタ動作についてのタイミングデータを含みます。この図は位相基準PWM動作以外の全ての動作種別でのMAX値近辺の計数の流れを示します。

図34. 前置分周なし(1/1)のタイマ/カウンタ タイミング

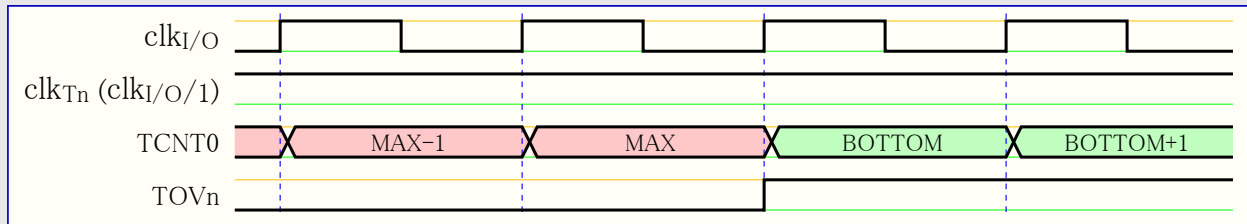


図35は同じタイミング データを示しますが、前置分周器が許可されています。

図35. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ タイミング

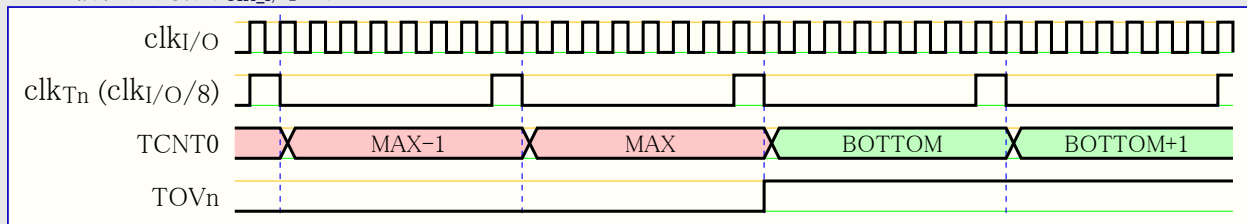


図36はCTC動作とOCR0AがTOPのPWM動作を除く全動作種別でのOCF0Aと全動作種別でのOCF0Bの設定を示します。

図36. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、OCF0x設定 タイミング

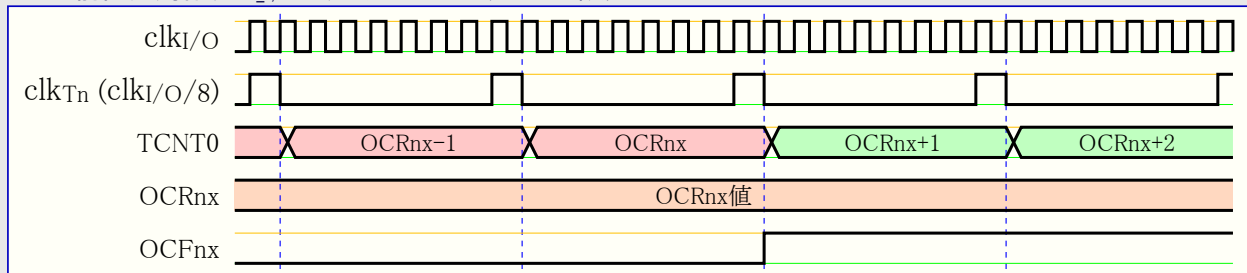
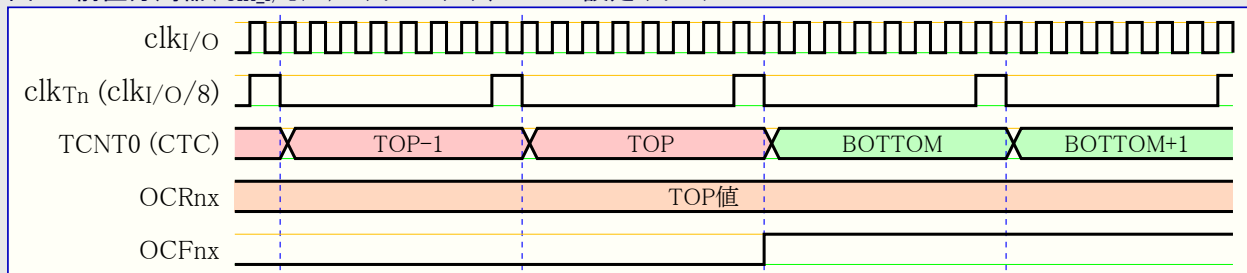


図37はOCR0AがTOPの高速PWM動作と、CTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図37. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、OCF0A設定 タイミング



8ビット タイマ/カウンタ0用レジスタ

タイマ/カウンタ0制御レジスタA (Timer/Counter 0 Control Register A) TCCR0A

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	COM0A1	COM0A0	COM0B1	COM0B0	–	–	WGM01	WGM00	TCCR0A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7,6 – COM0A1,0 : 比較A出力選択 (Compare Match A Output Mode bit 1 and 0)

これらのビットはOC0A比較出力ピンの動作を制御します。COM0A1,0ビットの1つまたは両方が1を書かれると、OC0A出力はそのI/Oピンの通常ポート機能を無効にし、そのI/Oピンに接続されます。けれども出力駆動部を許可するため、OC0Aピンに対応するポート方向レジスタ(DDR)のビットが設定(1)されなければならないことに注意してください。

OC0Aがピンに接続されるとき、COM0A1,0ビットの機能はWGM02~0ビット設定に依存します。

表33.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0A1,0ビット機能を示します。

表34.はWGM02~0ビットが高速PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表35.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表33. 非PWM動作比較A出力選択

COM0A1	COM0A0	意味
0	0	標準ポート動作 (OC0A切断)
0	1	比較一致でOC0Aピントグル(交互)出力
1	0	比較一致でOC0Aピン Lowレベル出力
1	1	比較一致でOC0Aピン Highレベル出力

表34. 高速PWM動作比較A出力選択 (共通注意参照)

COM0A1	COM0A0	意味
0	0	標準ポート動作 (OC0A切断)
0	1	WGM02=0 : 標準ポート動作 (OC0A切断) WGM02=1 : 比較一致でOC0Aピントグル(交互)出力
1	0	比較一致でLow、BOTTOMでHighをOC0Aピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC0Aピンへ出力 (反転動作)

表35. 位相基準PWM動作比較A出力選択 (共通注意参照)

COM0A1	COM0A0	意味
0	0	標準ポート動作 (OC0A切断)
0	1	WGM02=0 : 標準ポート動作 (OC0A切断) WGM02=1 : 比較一致でOC0Aピントグル(交互)出力
1	0	上昇計数時の比較一致でLow、下降計数時の比較一致でHighをOC0Aピンへ出力
1	1	上昇計数時の比較一致でHigh、下降計数時の比較一致でLowをOC0Aピンへ出力

•ビット5,4 – COM0B1,0 : 比較B出力選択 (Compare Match B Output Mode bit 1 and 0)

これらのビットはOC0B比較出力ピンの動作を制御します。COM0B1,0ビットの1つまたは両方が1を書かれると、OC0B出力はそのI/Oピンの通常ポート機能を無効にし、そのI/Oピンに接続されます。けれども出力駆動部を許可するため、OC0Bピンに対応するポート方向レジスタ(DDR)のビットが設定(1)されなければならないことに注意してください。

OC0Bがピンに接続されるとき、COM0B1,0ビットの機能はWGM02~0ビット設定に依存します。

表36.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0B1,0ビット機能を示します。

表37.はWGM02~0ビットが高速PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表38.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表36. 非PWM動作比較B出力選択

COM0B1	COM0B0	意味
0	0	標準ポート動作 (OC0B切断)
0	1	比較一致でOC0Bピントグル(交互)出力
1	0	比較一致でOC0Bピン Lowレベル出力
1	1	比較一致でOC0Bピン Highレベル出力

表37. 高速PWM動作比較B出力選択 (共通注意参照)

COM0B1	COM0B0	意味
0	0	標準ポート動作 (OC0B切断)
0	1	(予約)
1	0	比較一致でLow、BOTTOMでHighをOC0Bピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC0Bピンへ出力 (反転動作)

表38. 位相基準PWM動作比較B出力選択 (共通注意参照)

COM0B1	COM0B0	意味
0	0	標準ポート動作 (OC0B切断)
0	1	(予約)
1	0	上昇計数時の比較一致でLow、下降計数時の比較一致でHighをOC0Bピンへ出力
1	1	上昇計数時の比較一致でHigh、下降計数時の比較一致でLowをOC0Bピンへ出力

共通注意: COM0x1が設定(1)され、対応するOCR0xがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMまたはTOPでの設定(1)または解除(0)は行われます。より多くの詳細については47頁の「高速PWM動作」または48頁の「位相基準PWM動作」をご覧ください。(表34.,35.,37.,38.各々での注:を纏めました。)

• **ビット3,2 - Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

• **ビット1,0 - WGM01,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)**

タイマ/カウンタ制御レジスタB(TCCR0B)で得られるWGM02ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元、使われるべき波形生成のどの形式かを制御します(表39参照)。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。46頁の「動作種別」をご覧ください。

表39. 波形生成種別選択

番号	WGM02	WGM01	WGM00	タイマ/カウンタ動作種別	TOP値	OCR0x更新時	TOV0設定時
0	0	0	0	標準動作	\$FF	即時	MAX
1	0	0	1	8ビット位相基準PWM動作	\$FF	TOP	BOTTOM
2	0	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR0A	即時	MAX
3	0	1	1	8ビット高速PWM動作	\$FF	BOTTOM	MAX
4	1	0	0	(予約)	-	-	-
5	1	0	1	位相基準PWM動作	OCR0A	TOP	BOTTOM
6	1	1	0	(予約)	-	-	-
7	1	1	1	高速PWM動作	OCR0A	BOTTOM	TOP

注: MAX=\$FF、BOTTOM=\$00です。

タイマ/カウンタ0制御レジスタB (Timer/Counter0 Control Register B) TCCR0B

ビット	7	6	5	4	3	2	1	0	
\$33 (\$53)	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	TCCR0B
Read/Write	W	W	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7 - FOC0A : OC0A強制変更 (Force Output Compare A)**

FOC0AビットはWGM02~0ビットが非PWM動作を指示する時だけ有効です。

けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR0Bが書かれる場合、このビットは0に設定されなければなりません。FOC0Aビットに論理1を書くと、波形生成部で直ちに比較一致が強制されます。OC0A出力はCOM0A1,0ビット設定に従って変更されます。FOC0Aビットがスロープとして実行されることに注意してください。従って強制した比較の効果を決めるのはCOM0A1,0ビットに存在する値です。

FOC0Aスロープは何れの割り込みの生成もTOPとしてOCR0Aを使う比較一致タイマ解除(CTC)動作でのタイマ/カウンタの解除(\$00)も行いません。

FOC0Aビットは常に0として読みます。

• **ビット6 - FOC0B : OC0B強制変更 (Force Output Compare B)**

FOC0BビットはWGM02~0ビットが非PWM動作を指示する時だけ有効です。

けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR0Bが書かれる場合、このビットは0に設定されなければなりません。FOC0Bビットに論理1を書くと、波形生成部で直ちに比較一致が強制されます。OC0B出力はCOM0B1,0ビット設定に従って変更されます。FOC0Bビットがスロープとして実行されることに注意してください。従って強制した比較の効果を決めるのはCOM0B1,0ビットに存在する値です。

FOC0Bスロープは何れの割り込みの生成も行いません。

FOC0Bビットは常に0として読みます。

• **ビット5,4 - Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

• **ビット3 - WGM02 : 波形生成種別 (Waveform Generation Mode bit 2)**

50頁の「タイマ/カウンタ制御レジスタA - TCCR0A」のWGM01,0ビット記述をご覧ください。

•ビット2~0 – CS02~0 : クロック選択0 (Clock Select0, bit 2,1 and 0)

この3つのクロック選択ビットはタイマ/カウンタ(TCNT0)によって使われるクロック元を選びます。

表40. タイマ/カウンタ入力クロック選択

CS02	CS01	CS00	意味
0	0	0	停止 (タイマ/カウンタ動作停止)
0	0	1	clk _{I/O} (前置分周なし)
0	1	0	clk _{I/O} /8 (8分周)
0	1	1	clk _{I/O} /64 (64分周)
1	0	0	clk _{I/O} /256 (256分周)
1	0	1	clk _{I/O} /1024 (1024分周)
1	1	0	T0ピンの下降端 (外部クロック)
1	1	1	T0ピンの上昇端 (外部クロック)

外部ピン(クロック)動作がタイマ/カウンタに対して使われる場合、例えT0ピンが出力として設定されても、T0ピンの遷移はカウンタをクロック駆動します。この特徴がソフトウェアに計数制御を許します。

タイマ/カウンタ (Timer/Counter0) TCNT0

ビット	7	6	5	4	3	2	1	0	
\$32 (\$52)	(MSB)							(LSB)	TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

このタイマ/カウンタレジスタは読み書き両方の操作について、タイマ/カウンタ部の8ビットカウンタに直接アクセスします。TCNT0への書き込みは次のタイマ/カウンタクロックでの比較一致を妨害(除去)します。カウンタが走行中にカウンタ(TCNT0)を変更することはTCNT0とOCR0x間の比較一致消失の危険を誘発します。

タイマ/カウンタ 比較Aレジスタ (Timer/Counter0 Output Compare A Register) OCR0A

ビット	7	6	5	4	3	2	1	0	
\$36 (\$56)	(MSB)							(LSB)	OCR0A
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この比較レジスタは継続的にカウンタ(TCNT0)値と比較される8ビットの値を含みます。一致は比較一致割り込みやOC0Aピンでの波形出力を生成するのに使えます。

タイマ/カウンタ 比較Bレジスタ (Timer/Counter0 Output Compare B Register) OCR0B

ビット	7	6	5	4	3	2	1	0	
\$3C (\$5C)	(MSB)							(LSB)	OCR0B
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この比較レジスタは継続的にカウンタ(TCNT0)値と比較される8ビットの値を含みます。一致は比較一致割り込みやOC0Bピンでの波形出力を生成するのに使えます。

タイマ/カウンタ割り込み許可レジスタ (Timer/Counter Interrupt Mask Register) TIMSK

ビット	7	6	5	4	3	2	1	0	
\$39 (\$59)	TOIE1	OCIE1A	OCIE1B	–	ICIE1	OCIE0B	TOIE0	OCIE0A	TIMSK
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット4 – Res : 予約 (Reserved)**

このビットは予約されており、常に0として読みます。

- **ビット2 – OCIE0B : タイマ/カウンタ0比較B割り込み許可 (Timer/Counter Output Compare Match B Interrupt Enable)**

OCIE0Bビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較B一致割り込みが許可されます。タイマ/カウンタ(TCNT0)で比較一致が起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFR)で比較B割り込み要求フラグ(OCF0B)が設定(1)されると、対応する割り込みが実行されます。

- **ビット1 – TOIE0 : タイマ/カウンタ0溢れ割り込み許可 (Timer/Counter Overflow Interrupt Enable)**

TOIE0ビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0溢れ割り込みが許可されます。タイマ/カウンタ(TCNT0)溢れが起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFR)でタイマ/カウンタ溢れ割り込み要求フラグ(TOV0)フラグが設定(1)されると、対応する割り込みが実行されます。

- **ビット0 – OCIE0A : タイマ/カウンタ0比較A割り込み許可 (Timer/Counter Output Compare Match A Interrupt Enable)**

OCIE0Aビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが許可されます。タイマ/カウンタ(TCNT0)で比較一致が起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFR)で比較A割り込み要求フラグ(OCF0A)が設定(1)されると、対応する割り込みが実行されます。

タイマ/カウンタ割り込み要求フラグレジスタ (Timer/Counter Interrupt Flag Register) TIFR

ビット	7	6	5	4	3	2	1	0	
\$38 (\$58)	TOV1	OCF1A	OCF1B	–	ICF1	OCF0B	TOV0	OCF0A	TIFR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット4 – Res : 予約 (Reserved)**

このビットは予約されており、常に0として読みます。

- **ビット2 – OCF0B : タイマ/カウンタ0比較B割り込み要求フラグ (Timer/Counter, Output Compare B Match Flag)**

OCF0Bビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0B)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF0Bはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0Bは解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSK)のタイマ/カウンタ0比較B一致割り込み許可(OCIE0B)ビット、OCF0Bが設定(1)されると、タイマ/カウンタ0比較B一致割り込みが実行されます。

- **ビット1 – TOV0 : タイマ/カウンタ0溢れ割り込み要求フラグ (Timer/Counter Overflow Flag)**

TOV0ビットはタイマ/カウンタ(TCNT0)溢れが起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、TOV0はハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもTOV0は解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSK)のタイマ/カウンタ0溢れ割り込み許可(TOIE0)ビット、TOV0が設定(1)されると、タイマ/カウンタ0溢れ割り込みが実行されます。

このフラグの(1)設定はWGM02~0設定に依存します。51頁の「表39. 波形生成種別選択」を参照してください。

- **ビット0 – OCF0A : タイマ/カウンタ0比較A割り込み要求フラグ (Timer/Counter, Output Compare A Match Flag)**

OCF0Aビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0A)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF0Aはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0Aは解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSK)のタイマ/カウンタ0比較A一致割り込み許可(OCIE0A)ビット、OCF0Aが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが実行されます。

タイマ/カウンタ0とタイマ/カウンタ1の前置分周器

タイマ/カウンタ0とタイマ/カウンタ1は同じ前置分周器部を共有しますが、タイマ/カウンタは異なる前置分周器設定ができます。以下の記述はタイマ/カウンタ0とタイマ/カウンタ1の両方に適用されます。

内部クロック元

タイマ/カウンタはシステムクロック(CSn2~0=001設定)によって直接的にクロック駆動できます。これはシステムクロック周波数($f_{clk_I/O}$)と等しいタイマ/カウンタ最大クロック周波数での最高速動作を提供します。選択で前置分周器からの4つのタップの1つがクロック元として使えます。この前置分周したクロックは $f_{clk_I/O}/8$, $f_{clk_I/O}/64$, $f_{clk_I/O}/256$, $f_{clk_I/O}/1024$ の何れかの周波数です。

前置分周器リセット

この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ0とタイマ/カウンタ1によって共有されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが使われる状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステムクロック周期数はNが前置分周値(8, 64, 256, 1024)とすると、 $1 \sim N+1$ システムクロック周期になり得ます。

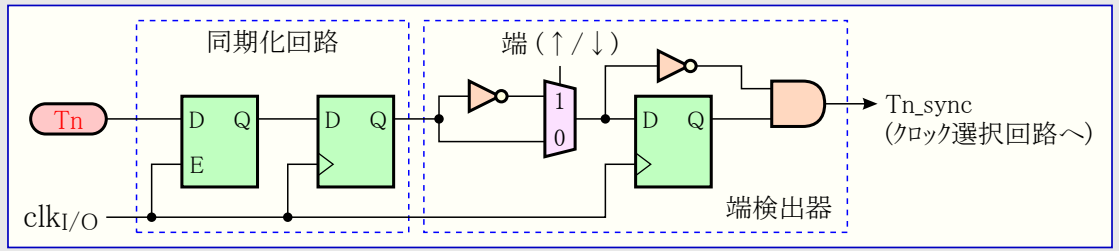
プログラム実行にタイマ/カウンタを同期することに対して前置分周器リセットを使うことが可能です。しかし、同じ前置分周器を共有する他のタイマ/カウンタも前置分周を使う場合、注意が必要とされなければなりません。前置分周器リセットはそれが接続される全タイマ/カウンタに関する前置分周器周期に影響を及ぼします。

外部クロック元

T0/T1ピンに印加された外部クロック元はタイマ/カウンタクロック(f_{clk_T0}/f_{clk_T1})として使えます。このT0/T1ピンはピン同期化論理回路によって全てのシステムクロック周期に一度採取されます。この同期化(採取)された信号はその後に端(エッジ)検出器を通して通過されます。図38はT0/T1同期化と端検出器論理回路の機能等価構成図を示します。レジスタは内部システムクロック($f_{clk_I/O}$)の上昇端でクロック駆動されます。ラッチは内部システムクロックのHigh区間で通過(Low区間で保持)です。

端検出器は上昇端(CSn2~0=111)または下降端(CSn2~0=110)の検出毎に1つの clk_{T0}/clk_{T1} パルスを生成します。

図38. T0/T1ピンの採取等価構成図



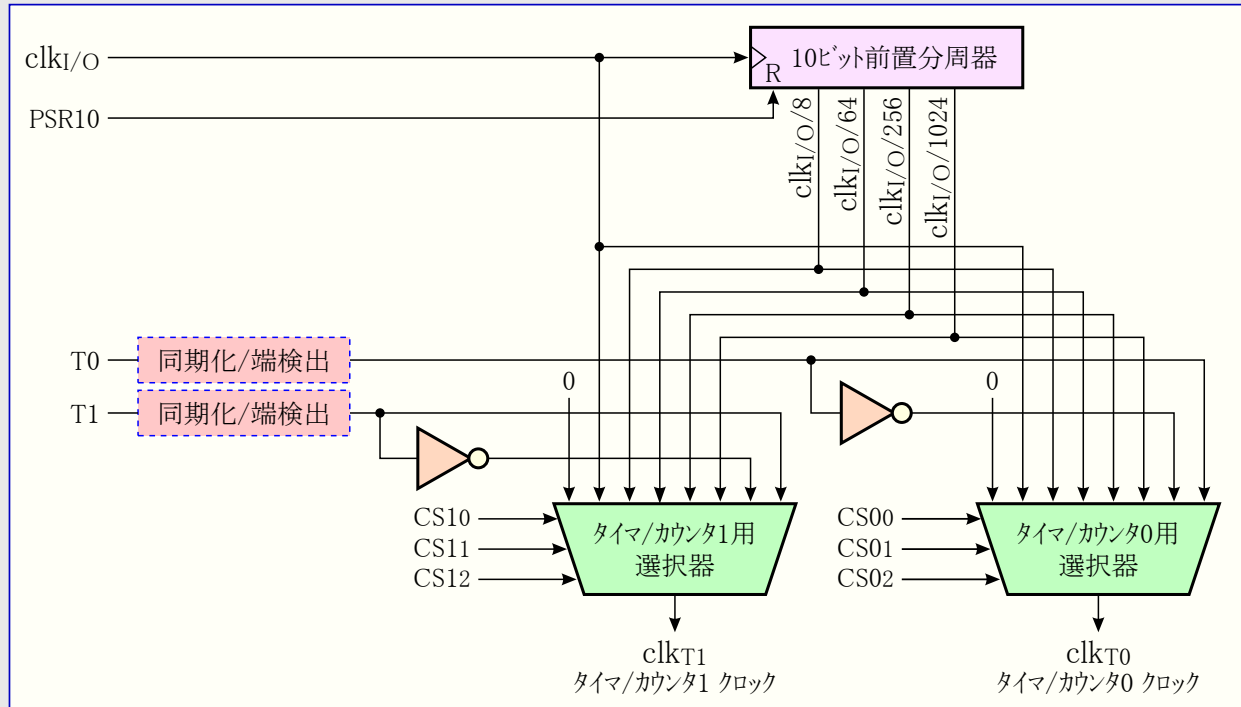
同期化と端検出器論理回路はT0/T1ピンへ印加された端から計数器が更新されるまでに2.5~3.5システムクロック周期の遅延をもたらします。

クロック入力の許可と禁止はT0/T1が最低1システムクロック周期に対して安定してしまっている時に行われなければならない、さもなければ不正なタイマ/カウンタクロックパルスが生成される危険があります。

印加された外部クロックの各半周期は正しい採取を保証するために1システムクロック周期より長くなければなりません。この外部クロックは50%/50%デューティ比で与えられるものとして、システムクロック周波数の半分未満($f_{EXTclk} < f_{clk_I/O}/2$)であることが保証されなければなりません。端検出器が採取を使うため、検出できる外部クロックの最大周波数は採取周波数の半分です(ナイキストの標本化定理)。然しながら、発振元(水晶発振子、セラミック振動子、コンデンサ)公差によって引き起こされたシステムクロック周波数やデューティ比の変動のため、外部クロック元の最大周波数は $f_{clk_I/O}/2.5$ 未満が推奨されます。

外部クロック元は前置分周できません。

図39. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器部構成図



注: 入力ピン(T0/T1)の同期化/端(エッジ)検出論理回路は図38.で示されます。

一般タイマ/カウンタ制御レジスタ (General Timer/Counter Control Register) GTCCR

ビット	7	6	5	4	3	2	1	0	
\$23 (\$43)	-	-	-	-	-	-	-	PSR10	GTCCR
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~1 - Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

- ビット0 - PSR10 : タイマ/カウンタ1,0前置分周器リセット (Prescaler Reset Timer/Counter 1,0)

このビットが1の時にタイマ/カウンタ0とタイマ/カウンタ1の前置分周器はリセットします。通常、このビットはハードウェアによって直ちに解除(0)されません。タイマ/カウンタ0とタイマ/カウンタ1は同じ前置分周器を共用し、この前置分周器のリセットが両方のタイマ/カウンタに影響を及ぼすことに注意してください。

16ビット タイマ/カウンタ1

この16ビット タイマ/カウンタ部は正確なプログラム実行タイミング(事象管理)、波形生成、信号タイミング計測を許します。主な特徴は次に示されます。

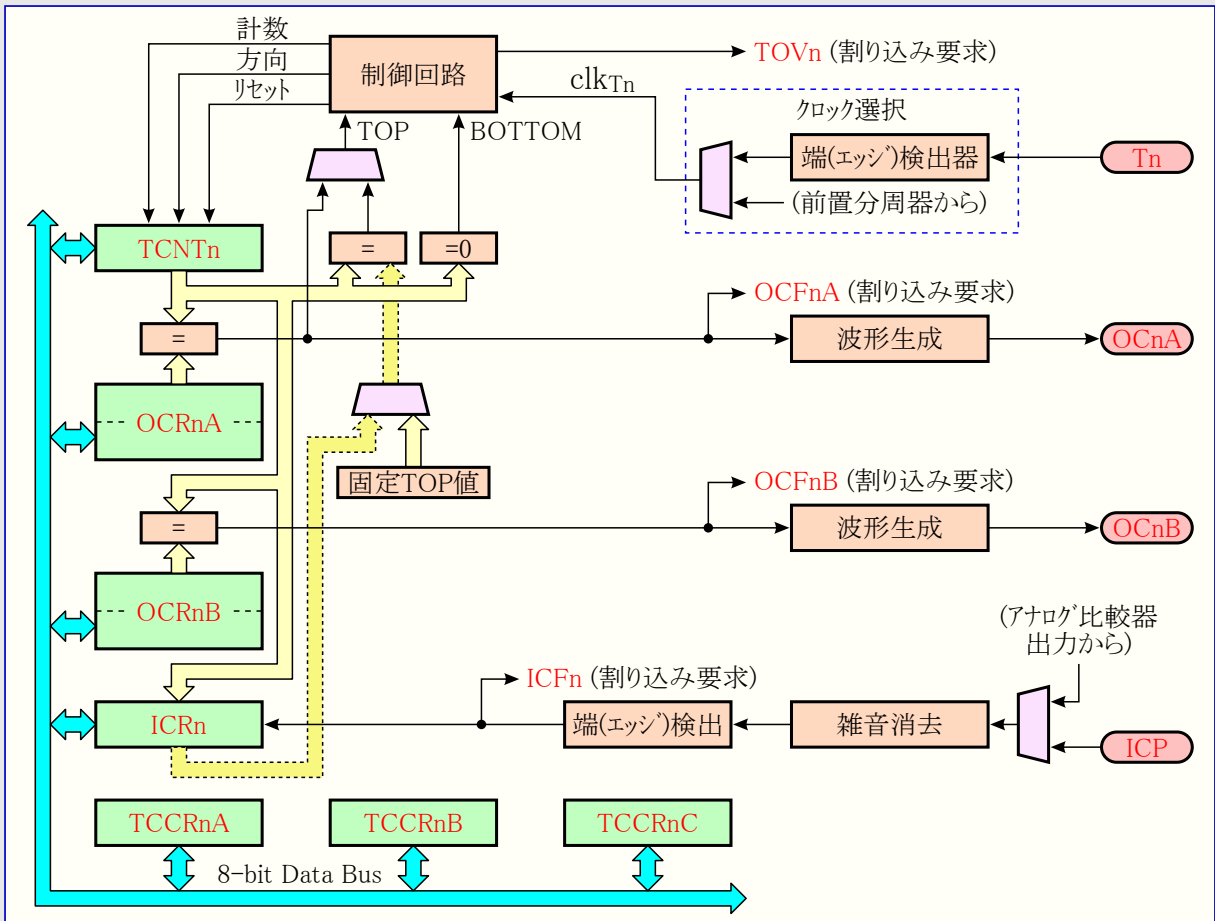
- 真の16ビット設計 (換言すれば16ビットPWMの許容)
- 2つの独立した比較出力部
- 2重緩衝の比較レジスタ
- 1つの捕獲入力部
- 捕獲入力雑音消去器
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 可変PWM周期
- 周波数発生器
- 外部事象計数器
- 4つの独立した割り込み (TOV1,OCF1A,OCF1B,ICF1)

概要

本文書でのレジスタとビット参照の多くは一般形で書かれます。小文字の'n'はタイマ/カウンタ番号、小文字の'x'は比較出力部の置き換えです。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えばタイマ/カウンタ1のカウンタ値に対するアクセスのTCNT1のように)。

この16ビット タイマ/カウンタの単純化した構成図は図40.で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤字(訳注: 原文太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は70頁の「16ビット タイマ/カウンタ1用レジスタ」で示されます。

図40. 16ビット タイマ/カウンタ構成図



注: タイマ/カウンタ1のピンの配置については2頁の「ピン配置」を参照してください。

関係レジスタ

タイマ/カウンタ(TCNT1)、比較レジスタ(OCR1A,OCR1B)、捕獲レジスタ(ICR1)は全て16ビットレジスタです。16ビットレジスタをアクセスするとき、特別な手順に従わなければなりません。これらの手順は58頁の「16ビットレジスタのアクセス」項で記述されます。タイマ/カウンタ1制御レジスタ(TCCR1A,TCCR1B,TCCR1C)は8ビットレジスタで、CPUアクセス制限はありません。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKはこの図で示されません。

このタイマ/カウンタは内部的、前置分周器経由、またはT1ピンの外部クロック元によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加(または減少)するのに使うクロック元と端(エッジ)を制御します。クロック元が選ばれないと、タイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkT1)として参照されます。

2重緩衝化した比較レジスタ(OCR1A,OCR1B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC1A,OC1B)ピンでPWMまたは可変周波数出力を生成するための波形生成器によって使えます。62頁の「比較出力部」をご覧ください。この比較一致発生は比較出力割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF1A,OCF1B)も設定(1)します。

捕獲レジスタ(ICR1)は捕獲起動(ICP)ピンまたはアナログ比較器出力(100頁の「アナログ比較器」参照)のどちらかの外部(端で起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波部(雑音除去器)を含みます。

TOP値または最大タイマ/カウンタ値は、いくつかの動作種別で、OCR1A、ICR1、または一群の固定値のどれかによって定義できます。PWM動作でTOP値としてOCR1Aを使うと、OCR1AはPWM出力生成用に使いません。けれどもこの場合、TOP値は動作中に変更されるのをTOP値に許す2重緩衝化します。固定的なTOP値が必要とされる場合、ICR1が代わりに使い、PWM出力として使われるべきOCR1Aを開放します。

定義

次の定義は本文書を通して広範囲に使われます。

表41. 用語定義

用語	意味
BOTTOM	カウンタが\$0000に到達した時。
MAX	カウンタが\$FFFF(65535)に到達した時。
TOP	カウンタがTOP値に到達した時(計数動作での最大値と等しくなった時)。TOP値は固定値(\$00FF,\$01FF,\$03FF)、OCR1A値、ICR1値の何れか1つを指定できます。この指定は動作種別に依存します。

互換性

この16ビットタイマ/カウンタは旧版の16ビット AVR タイマ/カウンタから改良更新されてしまっています。この16ビットタイマ/カウンタは次の点に関して以前の版と完全な互換性があります。

- ・タイマ/カウンタ割り込み関係レジスタを含め、16ビットタイマ/カウンタに関連する全てのI/Oレジスタアドレスの位置。
- ・タイマ/カウンタ割り込み関係レジスタを含め、16ビットタイマ/カウンタに関連する全てのI/Oレジスタ内のビット位置。
- ・割り込みベクタ。

次の制御ビットは変更された名前を持ちますが、同じ機能とレジスタ位置を持ちます。

- ・PWM10はWGM10に変更。
- ・PWM11はWGM11に変更。
- ・CTC1はWGM12に変更。

次の制御ビットが16ビットタイマ/カウンタ制御レジスタ内に追加されます。

- ・FOC1AとFOC1BがTCCR1Aに追加。
- ・WGM13がTCCR1Bに追加。

この16ビットタイマ/カウンタにはいくつかの特別な状況で互換性に影響を及ぼす改良点があります。

16ビットレジスタのアクセス

TCNT1, OCR1A, OCR1B, ICRI1は8ビットバス経由でAVR CPUによってアクセスできる16ビットレジスタです。この16ビットレジスタは2回の読みまたは書き操作を使ってバイトアクセスされなければなりません。16ビットタイマ/カウンタは16ビットアクセスの上位バイトの一時保存用に1つの8ビットレジスタを持ちます。16ビットタイマ/カウンタ内の全ての16ビットレジスタ間で、この同じ一時レジスタが共用されます。下位バイトアクセスが16ビット読み書き動作を起動します。16ビットレジスタの下位バイトがCPUによって書かれると、一時レジスタに保存した上位バイトと書かれた下位バイトは同じクロック周期で両方が16ビットレジスタに複写されます。16ビットレジスタの下位バイトがCPUによって読まれると、16ビットレジスタの上位バイトは下位バイトが読まれるのと同じクロック周期で一時レジスタに複写されます。

全ての16ビットアクセスが上位バイトに対して一時レジスタを使う訳ではありません。OCR1AとOCR1Bの16ビットレジスタ読み込みは一時レジスタの使用に関係しません。

16ビット書き込みを行うために上位バイトは下位バイトに先立って書かれなければなりません。16ビット読み込みについては下位バイトが上位バイトの前に読まれなければなりません。

次のコード例は割り込みが一時レジスタを更新しないことが前提の16ビットタイマ/カウンタレジスタのアクセス法を示します。OCR1A, OCR1B, ICRI1レジスタのアクセスに対して同じ原理が直接的に使えます。C言語を使う時はコンパイラが16ビットアクセスを扱うことに注意してください。

アセンブリ言語プログラム例

```

~
LDI    R17, $01           ;
LDI    R16, $FF          ;[16ビット($01FF)書き込み]
OUT    TCNT1H, R17       ;$01FFの上位バイト値取得
OUT    TCNT1L, R16       ;$01FFの下位バイト値取得
IN     R16, TCNT1L       ;上位バイト設定(一時レジスタ)
IN     R17, TCNT1H       ;下位バイト設定(一時レジスタ⇒上位バイト)
~
IN     R16, TCNT1L       ;[16ビット読み込み]
IN     R17, TCNT1H       ;下位バイト取得(上位バイト⇒一時レジスタ)
~

```

C言語プログラム例

```

unsigned int i;
~
TCNT1 = 0x1FF;           /* */
i = TCNT1;               /* 16ビット($01FF)書き込み */
~

```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

16ビットレジスタアクセスが非分断操作であることに注意することが重要です。16ビットレジスタをアクセスする2命令間で割り込みが起き、割り込みコードがその16ビットタイマ/カウンタレジスタの同じ若しくは他の何れかをアクセスすることによって一時レジスタを更新する場合、割り込み外のその後のアクセス結果は不正にされます。従って主コードと割り込みコードの両方が一時レジスタを更新するとき、主コードは16ビットアクセス中の割り込みを禁止しなければなりません。

次のコード例はTCNT1レジスタ内容の非分断読み込み法を示します。同じ原理を使うことにより、OCR1A,OCR1B,ICR1のどの読み込みも行えます。

アセンブリ言語プログラム例

```
RD_TCNT1:   IN      R18, SREG           ; 現全割り込み許可フラグ(I)を保存
            CLI                    ; 全割り込み禁止
            IN      R16, TCNT1L        ; TCNT1下位バイト取得(上位バイト⇒一時レジスタ)
            IN      R17, TCNT1H        ; TCNT1上位バイト取得(一時レジスタ)
            OUT     SREG, R18          ; 全割り込み許可フラグ(I)を復帰
            RET                       ; 呼び出し元へ復帰
```

C言語プログラム例

```
unsigned int TIM16_Read_TCNT1(void)
{
    unsigned char sreg;                /* ステータスレジスタ一時保存変数定義 */
    unsigned int i;                    /* TCNT1読み出し変数定義 */
    sreg = SREG;                       /* 現全割り込み許可フラグ(I)を保存 */
    _CLI();                             /* 全割り込み禁止 */
    i = TCNT1;                          /* TCNT1値を取得 */
    SREG = sreg;                       /* 全割り込み許可フラグ(I)を復帰 */
    return i;                          /* TCNT1値で呼び出し元へ復帰 */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

次のコード例はTCNT1レジスタ内容の非分断書き込み法を示します。同じ原理を使うことにより、OCR1A,OCR1B,ICR1のどの書き込みも行えます。

アセンブリ言語プログラム例

```
WR_TCNT1:   IN      R18, SREG           ; 現全割り込み許可フラグ(I)を保存
            CLI                    ; 全割り込み禁止
            OUT     TCNT1H, R17        ; TCNT1上位バイト設定(一時レジスタ)
            OUT     TCNT1L, R16        ; TCNT1下位バイト設定(一時レジスタ⇒上位バイト)
            OUT     SREG, R18          ; 全割り込み許可フラグ(I)を復帰
            RET                       ; 呼び出し元へ復帰
```

C言語プログラム例

```
void TIM16_Write_TCNT1(unsigned int i)
{
    unsigned char sreg;                /* ステータスレジスタ一時保存変数定義 */
    unsigned int i;                    /* TCNT1書き込み変数定義 */
    sreg = SREG;                       /* 現全割り込み許可フラグ(I)を保存 */
    _CLI();                             /* 全割り込み禁止 */
    TCNT1 = i;                          /* TCNT1値を設定 */
    SREG = sreg;                       /* 全割り込み許可フラグ(I)を復帰 */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

アセンブリ言語コード例はR17:R16レジスタ対がTCNT1へ書かれるべき値を含むことが必要です。

上位バイト一時レジスタの再使用

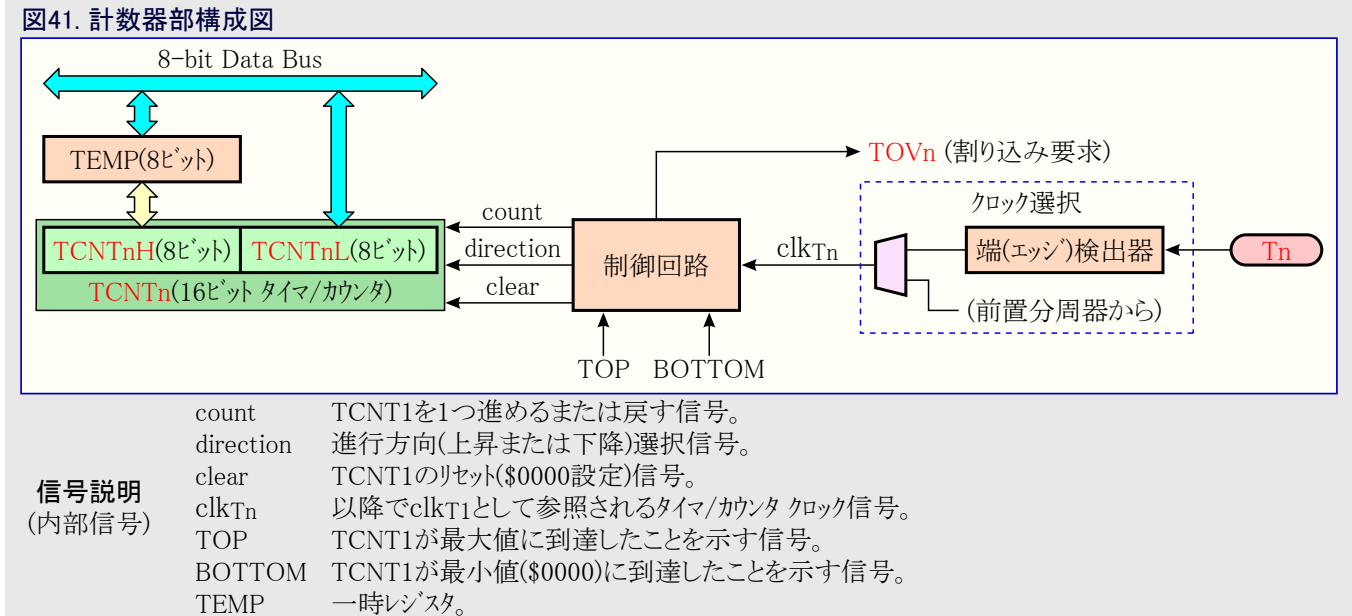
書かれる全レジスタについて上位バイトが同じ複数16ビットレジスタ書き込みなら、上位バイトは1度書かれることだけが必要です。けれども直前で記述した非分断操作の同じ規則が、この場合にも適用されることに注意してください。

タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**クロック選択(CS12~0)**ビットによって制御されるクロック選択論理回路で選ばれます。クロック元と前置分周器の詳細については54頁の「**タイマ/カウンタ0とタイマ/カウンタ1の前置分周器**」をご覧ください。

計数器部

16ビットタイマ/カウンタの主な部分は設定可能な16ビット双方向カウンタ部です。図41はこのカウンタとその周辺の構成図を示します。



この16ビットカウンタはカウンタの上位8ビットを含むカウンタ上位(TCNT1H)と下位8ビットを含むカウンタ下位(TCNT1L)の2つの8ビット I/Oメモリ位置に配置されます。TCNT1HレジスタはCPUによる間接的なアクセスのみできます。CPUがTCNT1H I/O位置をアクセスするとき、CPUは上位バイト一時レジスタ(TEMP)をアクセスします。この一時レジスタはTCNT1Lが読まれる時にTCNT1H値で更新され、TCNT1Lが書かれる時にTCNT1Hは一時レジスタ値で更新されます。これは8ビットデータバス経由で1クロック周期内の16ビットカウンタ値全体の読み書きをCPUに許します。予測不能な結果を生じる、カウンタが計数中の時のTCNT1書き込みの特別な場合に注意することが重要です。この特別な場合はそれらが重要となる項で記述されます。

使った動作種別に依って、カウンタは各タイマ/カウンタクロック(clkT1)でリセット(\$0000)、増加(+1)、または減少(-1)されます。clkT1はクロック選択(CS12~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS12~0=000)時にカウンタは停止されます。けれどもTCNT1値はタイマ/カウンタクロック(clkT1)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先権を持ちます)。

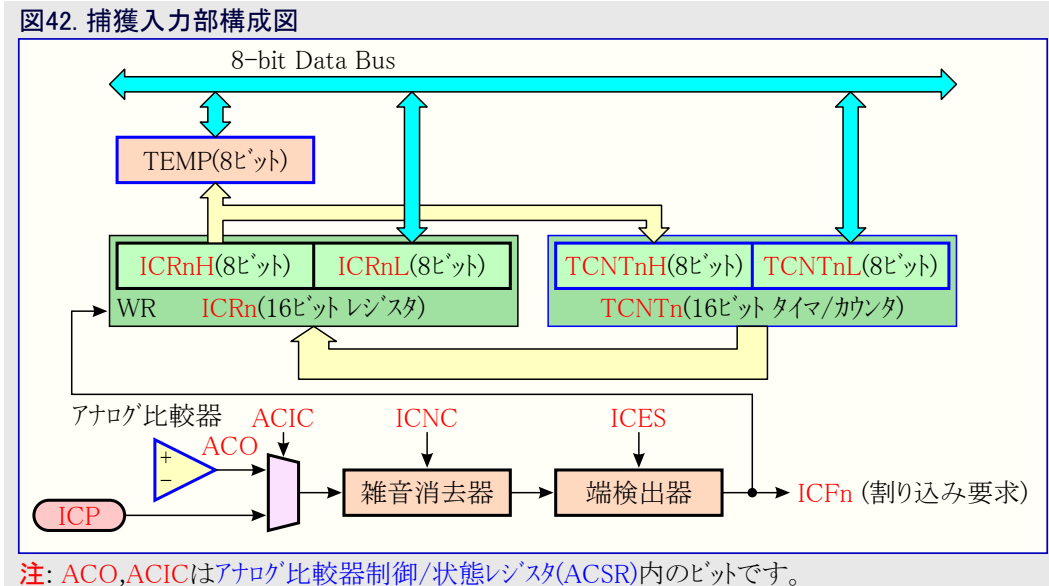
計数順序(方法)は**タイマ/カウンタ制御レジスタA(TCCR1A)**と**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**波形生成種別(WGM13~0)**ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては65頁の「**動作種別**」をご覧ください。

タイマ/カウンタ溢れ(TOV1)フラグはWGM13~0ビットによって選ばれた動作種別に従って設定(1)されます。TOV1はCPU割り込み発生に使えます。

捕獲入力部

タイマ/カウンタは外部の出来事を捕獲でき、発生時間を示す時間印(タイマ/カウンタ値)を与える捕獲入力部と合体します。出来事または複数の出来事を示す外部信号はICPピンまたは代わりにアナログ比較器経由で印加できます。時間印はその後、周波数、デューティ比、印加された信号の他の特性の計算に使えます。代わりに時間印は出来事の記録作成にも使えます。

捕獲入力部は図42.で示される構成図によって図解されます。直接的な捕獲入力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号を示します。



捕獲起動入力(ICP)ピン若しくは代わりにアナログ比較器出力(ACO)で論理レベルの変化(出来事)が起き、その変化が端(エッジ)検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF1)はTCNT1値がICR1に複写されると同じシステムクロックで設定(1)されます。許可(I=1, ICIE1=1)ならば捕獲割り込み要求フラグは捕獲割り込みを発生します。ICF1は割り込みが実行されると自動的に解除(0)されます。代わりにこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。

捕獲レジスタ(ICR1)の16ビット値読み込みは、初めに下位バイト(ICR1L)、その後上位バイト(ICR1H)を読むことによって行われます。下位バイトが読まれる時に上位バイトが上位バイト一時レジスタ(TEMP)に複写されます。CPUがICR1H I/O位置を読むと、この一時レジスタにアクセスします。

ICR1はカウンタのTOP値定義にICR1を利用する波形生成種別を使う時にだけ書けます。これらの場合、TOP値がICR1に書かれ得る前に波形生成種別(WGM13~0)ビットが設定されなければなりません。ICR1に書く時は下位バイトがICR1Lに書かれる前に、上位バイトがICR1H I/O位置に書かれなければなりません。

16ビットレジスタアクセス法のより多くの情報については58頁の「16ビットレジスタのアクセス」を参照してください。

捕獲起動元

捕獲入力部用の主な起動元は捕獲起動入力(ICP)ピンです。タイマ/カウンタ1は捕獲入力部用起動元としてアナログ比較器出力を代わりに使えます。アナログ比較器はアナログ比較器制御/状態レジスタ(ACSR)のアナログ比較器捕獲起動許可(ACIC)ビットの設定(1)によって起動元として選ばれます。起動元を変更することが捕獲を起動し得ることに気付いてください。従って捕獲割り込み要求フラグ(ICF1)は、その変更後に解除(0)されなければなりません。

捕獲起動入力(ICP)ピンとアナログ比較器出力(ACO)の両入力は、T1ピン(54頁の図38.参照)についてと同じ技法を使って採取されます。端検出器も全く同じです。けれども雑音除去が許可されると、付加論理回路が端検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されないなら、雑音除去器と端検出器の入力が常に許可されることに注意してください。

捕獲入力はICPピンのポートを制御することによってソフトウェアで起動できます。

雑音除去器

雑音除去器は簡単なデジタル濾波器機構を使うことによって雑音耐性を改善します。雑音除去器の入力は4採取に渡って監視され、端検出器によって使われる方向転換となる出力を変更するためには4回全てが同じでなければなりません。

雑音除去器はタイマ/カウンタ制御レジスタB(TCCR1B)の捕獲入力雑音除去許可(ICNC1)ビットの設定(1)によって許可されます。許可したとき、雑音除去器は入力に印加した変更からICR1の更新までに4システムクロック周期の追加遅延をもたらします。雑音除去器はシステムクロックを使い、従って前置分周器によって影響されません。

捕獲入力の使用

捕獲入力機能を使う主な要求(目的)は入って来る出来事に対して十分なプロセッサ能力を当てがうことです。2つの出来事間の時間が際どいとして、次の出来事が起こる前に捕獲した**捕獲レジスタ(ICR1)**の値をプロセッサが読めなかった場合、ICR1は新しい値で上書きされます。この場合、捕獲の結果は不正にされます。

捕獲割り込みを使う時にICR1は割り込み処理ルーチンで可能な限り早く読まれるべきです。捕獲割り込みが相対的に高い優先順位であっても、最大割り込み応答時間は他の割り込み要求のどれかを扱うのに必要とされる最大クロック周期数に依存します。

動作中に**TOP値(分解能)**が積極的に変更されるとき、どの動作種別での捕獲入力部の使用も推奨されません。

外部信号のデューティ比測定は各捕獲後に起動端が変更されることを必要とします。検出端の変更はICR1が読まれてしまった後に可能な限り早く行われなければなりません。端(エッジ)の変更後、**捕獲割り込み要求フラグ(ICF1)**はソフトウェア(I/Oビット位置への論理1書き込み)によって解除(0)されなければなりません(訳補:端変更によってICF1が設定(1)されることを想定)。周波数のみの測定について(割り込み処理が使われる場合)、ICF1の解除(0)は必要とされません。

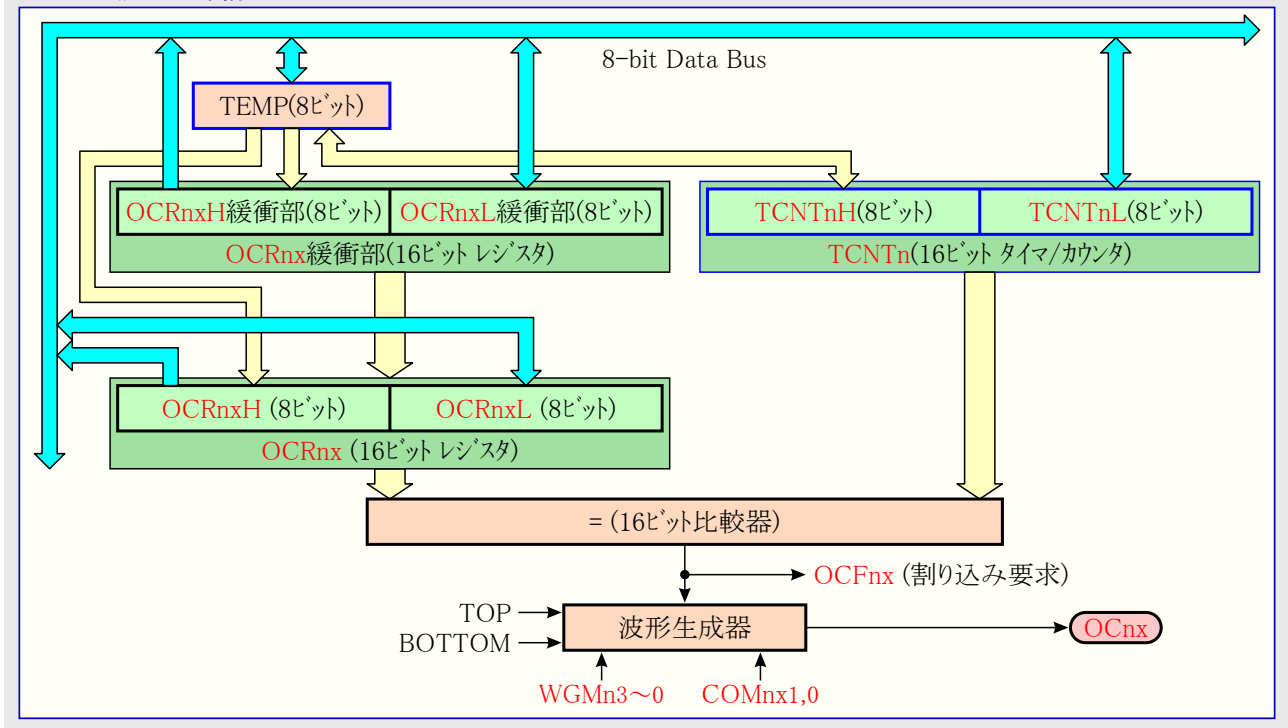
比較出力部

この16ビット比較器は**TCNT1**と**比較レジスタ(OCR1x)**を継続的に比較します。TCNT1とOCR1xが等しければ、比較器は一致を指示します。この一致は**次のタイマ/カウンタクロック周期**で**比較割り込み要求フラグ(OCF1x)**を設定(1)します。許可(I=1, OCIE1x=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF1xは割り込みが実行されると自動的に解除(0)されます。代わりにOCF1xはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**によって設定された動作種別に従った出力を生成するのにこの一致信号を使います。TOPとBOTTOM信号は動作種別(65頁の「動作種別」参照)のいくつかで両端値の特別な場合を扱うため、波形生成器によって使われます。

比較A出力部の特殊な特性はタイマ/カウンタのTOP値(換言するとカウンタの分解能)定義を許します。カウンタの分解能に加え、TOP値は波形生成器によって生成された波形の周期時間を定義します。

図43. は比較出力部の構成図を示します。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対してはn=1)、小文字の'x'は比較出力部(AまたはB)を表します。直接的な比較出力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。

図43. 比較出力部構成図



OCR1xは12種類のパルス幅変調(PWM)のどれかを使う時に2重緩衝化されます。標準動作と比較一致タイマ/カウンタ解除(CTC)動作については2重緩衝動作が禁止されます。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR1xレジスタの更新を同期化します。この同期化は奇数長、非対称PWMパルスの発生を防ぎ、それによって不具合なしの出力を作成します。

OCR1xのアクセスは複雑なように思えますが決してそんなことはありません。2重緩衝動作が許可されるとCPUはOCR1x緩衝部をアクセスし、禁止されるとOCR1xレジスタを直接アクセスします。OCR1x(緩衝部またはレジスタ)の内容は書き込み操作によってのみ変更されます(タイマ/カウンタはTCNT1やICR1のようにOCR1xを自動的に更新しません)。従ってOCR1xは上位バイト一時レジスタ(TEMP)経由で読まれません。けれども他の16ビットレジスタをアクセスする時のように下位バイトを先に読むのは良い習慣です。OCR1x書き込みは16ビット全ての比較が継続的に行われるため、一時レジスタ経由で行われなければなりません。上位バイト(OCR1xH)が先に書かれなければなりません。上位バイトI/O位置がCPUによって書かれると、一時レジスタは書かれた値で更新されます。その後下位バイト(OCR1xL)が下位8ビットを書かれると、(一時レジスタ内の)上位バイトは(下位バイト書き込みと)同じシステムクロック周期でOCR1x緩衝部またはOCR1xレジスタのどちらかに複写されます。

16ビットレジスタアクセス法のより多くの情報については58頁の「16ビットレジスタのアクセス」を参照してください。

(訳注) ここでは比較nxレジスタ全体をOCRnx、OCRnxを構成する緩衝部分をOCRnx緩衝部、実際の比較に使われるレジスタ本体部分をOCRnxレジスタとして記述しています。他の部分での記述でも特に必要がある場合はこの記述方法を適用します。

強制比較出力

非PWM波形生成動作での比較器の一致出力は強制変更(FOC1x)ビットに1を書くことによって強制(変更)できます。比較一致の強制は比較割り込み要求フラグ(OCF1x)の設定(1)やタイマ/カウンタの再設定/解除を行いませんが、OC1xピンは実際の比較一致が起きた場合と同様に更新されます(COM1x1,0ビット設定がOC1xピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

TCNT1書き込みによる比較一致妨害

TCNT1への全てのCPU書き込みは、例えタイマ/カウンタが停止されていても、次のタイマ/カウンタクロック周期で起こるどんな比較一致をも妨げます。この特質はタイマ/カウンタクロックが許可されている時に、割り込みを起動することなく、TCNT1と同じ値に初期化されることをOCR1xに許します。

比較一致部の使用

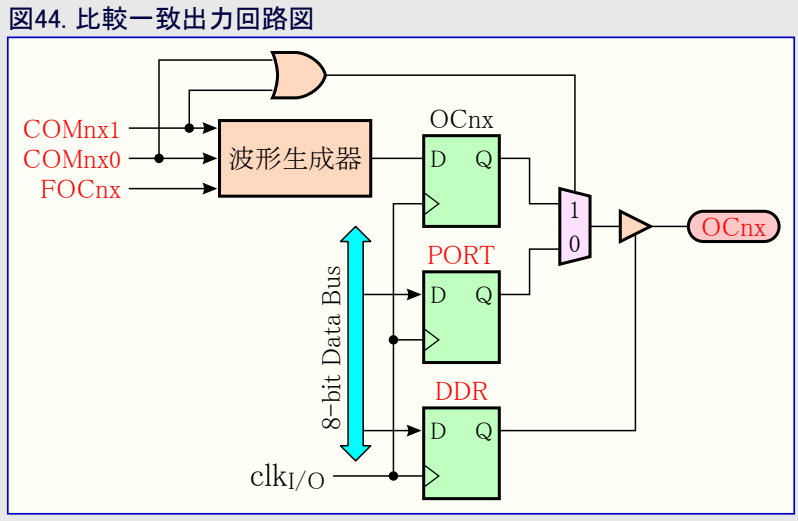
どの動作種別でのTCNT1書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、何れかの比較出力部を使う場合、TCNT1を変更する時に危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。可変TOP値のPWM動作でTOPに等しいTCNT1を書いてはいけません。(行った場合)TOPに対する比較一致は無視され、カウンタは\$FFFFへ(計数を)続けます。同様にカウンタが下降計数のとき、BOTTOMに等しいTCNT1値を書いてはいけません。

OC1xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)ストローブビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

比較出力選択(COM1x1,0)ビットが比較値(OCR1x)と共に2重緩衝されないことに気付いてください。COM1x1,0ビットの変更は直ちに有効となります。

比較一致出力部

比較出力選択(COM1x1,0)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力(OC1x)状態の定義にCOM1x1,0ビットを使います。次にCOM1x1,0ビットはOC1xピン出力元を制御します。図44.はCOM1x1,0ビット設定によって影響される論理回路の簡易化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM1x1,0ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。OC1xの状態を参照するとき、その参照はOC1xピンでなく内部OC1xレジスタに対してです。システムリセットが起ると、OC1xレジスタは0にリセットされます。



COM1x1,0ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力(OC1x)によって無効にされます。けれどもOC1xピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。OC1xピンに対するポート方向レジスタのビット(DDR_OC1x)はOC1x値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は一般的に波形生成種別と無関係ですが、いくつかの例外があります。詳細については表42、表43、表44を参照してください。

比較出力ピン論理回路の設計は出力が許可される前のOC1x状態の初期化を許します。いくつかのCOM1x1,0ビット設定が或る種の動作種別に対して予約されることに注意してください。70頁の「16ビットタイマ/カウンタ1用レジスタ」をご覧ください。

COM1x1,0ビットは捕獲入力部での何の効果もありません。

比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作でCOM1x1,0ビットを違うふうに使います。全動作種別に対してCOM1x1,0=00設定は次の比較一致で実行すべきOC1xレジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については70頁の表42を参照してください。高速PWM動作については70頁の表43、位相基準PWMと位相/周波数基準PWMについては70頁の表44を参照してください。

COM1x1,0ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(FOC1x)ストップビットを使うことによって直ちに効果を得ることを強制できます。

動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**の組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM1x1,0ビットは生成されたPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対してのCOM1x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します。64頁の「**比較一致出力部**」をご覧ください。

タイミング情報の詳細については69頁の「**タイマ/カウンタのタイミング**」を参照してください。

標準動作

最も単純な動作種別が標準動作(WGM13~0=0000)です。この動作種別での計数方向は常に上昇(+)で、カウンタの解除は実行されません。カウンタは16ビット最大値(MAX=\$FFFF)を通過すると単に範囲を超え、そして\$0000(BOTTOM)から再び始めます。通常動作での**タイマ/カウンタ溢れ(TOV1)フラグ**はTCNT1が\$0000になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV1フラグは設定(1)のみで解除(0)されないことを除いて第17ビットのようになります。けれどもTOV1フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解能はソフトウェアによって増やせます。標準動作での考慮に特別な場合はなく、新しいカウンタ値は何時でも書けます。

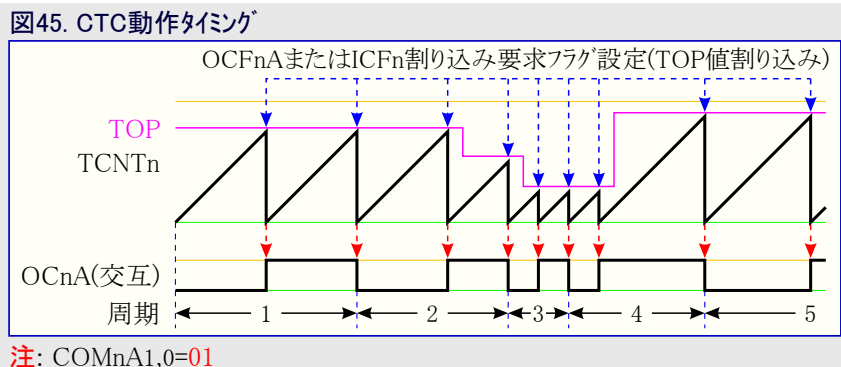
捕獲入力部は標準動作での使用が容易です。けれども外部の事象間の最大間隔がタイマ/カウンタの分解能(16ビット長)を越えてはならないことに気付いてください。事象間の間隔が長すぎる場合、捕獲部に対して分解能を拡張するために、タイマ/カウンタ溢れ割り込みまたは前置分周器が使われなければなりません。

比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM13~0=0100または1100)ではOCR1AまたはICR1がカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT1)値がOCR1A(WGM13~0=4)またはICR1(WGM13~0=12)のどちらかと一致する時にカウンタが\$0000に解除されます。OCR1AまたはICR1はカウンタに対する**TOP値**、従って分解能も定義します。この動作種別はより大きい比較一致出力周波数の制御を許します。それは外部の出来事の計数操作も簡単にします。

CTC動作についてのタイミング図は図45.で示されます。カウンタ(TCNT1)値はOCR1AまたはICR1のどちらかで比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT1)は解除(\$0000)されます。



TOP値を定義するのに使われるレジスタに対してOCF1AまたはICF1のどちらかを使うことにより、カウンタ値がTOP値に到達する時毎に割り込みが生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。けれども前置分周なしまたは低い前置分周値でカウンタが走行している時にBOTTOMと近い値にTOPを変更するのは、CTC動作が2重緩衝機能を持たないために注意して行わなければなりません。OCR1AまたはICR1に書かれた新しい値がTCNT1の現在値よりも低い(小さい)場合、カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こせるのに先立って、最大値(\$FFFF)へそして次に\$0000から始める計数をしなければならぬでしょう。多くの場合でこの特性は好ましくありません。OCR1Aが2重緩衝されるので、代替はTOPを定義するのにOCR1Aを用いる**高速PWM動作(WGM13~0=1111)**を使うことでしよう。

CTC動作で波形出力を生成するため、OC1A出力は**比較出力選択(COM1A1,0)ビット**を交互動作(=01)に設定することによって各比較一致での論理レベル交互切替に設定できます。OC1A値はそのピンに対するデータ方向が出力(DDR_OC1A=1)に設定されない限り、ポートピンで見えないでしょう。生成された波形はOCR1Aが0(\$0000)に設定されるとき、 $f_{OC1A} = f_{clk_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

$$f_{OCnA} = \frac{f_{clk_I/O}}{2 \times N \times (1 + OCRnA)}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

標準動作と同じように、**タイマ/カウンタ溢れ割り込み要求(TOV1)フラグ**はカウンタがMAXから\$0000へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

高速PWM動作

高速パルス幅変調(PWM)動作(WGM13~0=0101,0110,0111,1110,1111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)はTCNT1とOCR1x間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM1x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準や位相/周波数基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

高速PWMのPWM分解能は8,9,10ビットに固定、若しくはOCR1AかICR1のどちらかによって定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は次式を使うことによって計算できます。

$$R_{PWM} = \frac{\log(TOP+1)}{\log 2}$$

高速PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0101,0110,0111)、ICR1値(WGM13~0=1110)またはOCR1A値(WGM13~0=1111)のどれかと一致するまで増加されます。そしてカウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$0000)されます。高速PWM動作のタイミング図は図46.

で示されます。本図はOCR1AかICR1がTOPを定義するのに使われる時の高速PWM動作を示します。TCNT1値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

タイマ/カウンタ溢れ割り込み要求(TOV1)フラグはカウンタがTOPに到達する時毎に設定(1)されます。加えて、OCR1AかICR1のどちらかがTOP値を定義するのに使われると、OCF1AまたはICF1割り込み要求フラグはTOV1が設定(1)されるのと同じタイマ/カウンタクロック周期で設定(1)されます。これらの割り込みの1つが許可されるなら、その割り込み処理ルーチンはTOPと比較値を更新するのに使えます。

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。固定TOP値を使う場合、どのOCR1xが書かれる時も、未使用ビットが0で隠(に置換)されることに注意してください。

ICR1がTOP値を定義するのに使われるとき、ICR1を更新する手順はOCR1Aの更新と異なります。ICR1は2重緩衝されません。これは前置分周なしまたは低い前置分周値でカウンタが走行している時にICR1が小さな値に変更される場合、書かれた新しいICR1値がTCNT1の現在値よりも小さくなる危険を意味します。その後の結果はカウンタが(その回の)TOP値での比較一致を失うことです。その後のカウンタは比較一致が起こせるのに先立って、MAX値(\$FFFF)へそして次に\$0000から始める計数をしなければならないでしょう。しかし、OCR1Aは2重緩衝されます。この特徴は何時でも書かれることをOCR1AのI/O位置に許します。OCR1A I/O位置が書かれると、書かれた値はOCR1A緩衝部に置かれます。OCR1A(比較)レジスタはその後にTCNT1がTOPと一致した次のタイマ/カウンタクロック周期にOCR1A緩衝部の値で更新されます。この更新はTCNT1の解除(\$0000)やTOV1の設定(1)と同じタイマ/カウンタクロック周期で行われます。

TOPを定義するのにICR1を使うことは決まったTOP値を使う時に上手いきます。ICR1を使うことにより、OC1AでのPWM出力を生成するためにOCR1Aが自由に使えます。けれども基準PWM周波数が(TOP値を変更することによって)動的に変更される場合、OCR1Aが2重緩衝機能のため、TOPとしてOCR1Aを使うことは明らかに良い選択です。

高速PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます。70頁の表43をご覧ください。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$0000、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でのOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

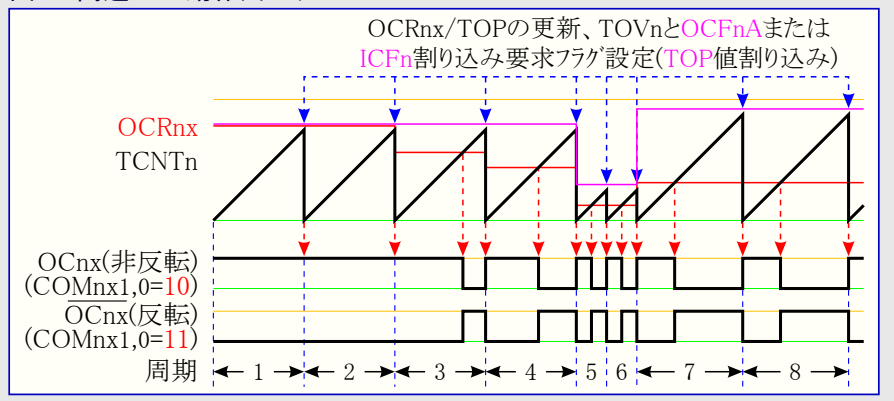
PWM出力周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

$$f_{OCnxPWM} = \frac{f_{clk_I/O}}{N \times (1 + TOP)}$$

OCR1xの両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合に当たります。OCR1xがBOTTOM(\$0000)に等しく設定されると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。TOPに等しいOCR1x設定は(COM1x1,0ビットによって設定される出力極性に依存して)定期的なLowまたはHigh出力に終わるでしょう。

高速PWM動作で(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC1A設定(COM1A1,0=01)によって達成できます。これはTOP値を定義するのにOCR1Aが使われる(WGM13~0=1111)の場合にだけ適用されます。生成された波形はOCR1Aが0(\$0000)に設定される時に $f_{OC1A} = f_{clk_I/O} / 2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いて、CTC動作でのOC1A交互出力(COM1A1,0=01)と同じです。

図46. 高速PWM動作タイミング



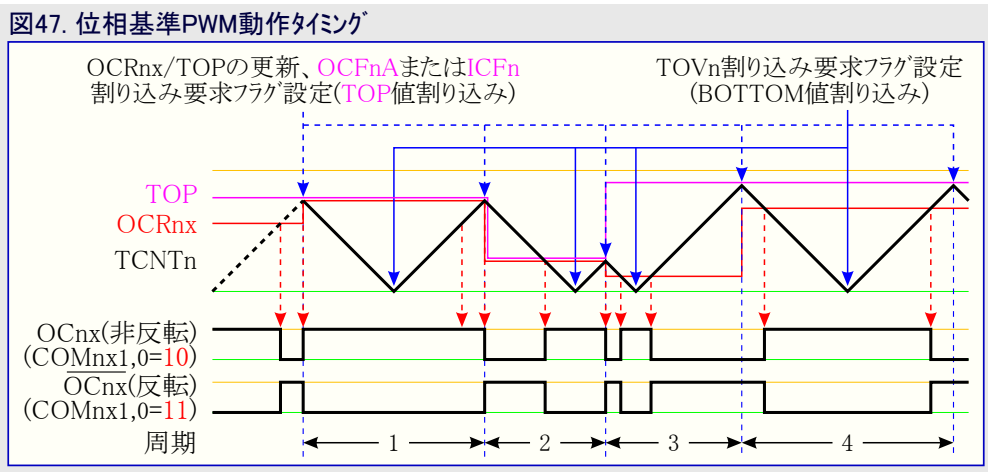
位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM13~0=0001,0010,0011,1010,1011)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とした位相/周波数基準PWMと似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準PWM動作のPWM分解能は8,9,10ビットに固定、若しくはOCR1AかICR1のどちらかによって定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は次式を使うことによって計算できます。

$$R_{PCPWM} = \frac{\log(TOP+1)}{\log 2}$$

位相基準PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0001,0010,0011)、ICR1値(WGM13~0=1010)またはOCR1A値(WGM13~0=1011)のどれかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期TOPと等しくなります。位相基準PWM動作のタイミング図は図47.で示されます。本図はOCR1AかICR1がTOPを定義するのに使われる時の位相基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。



タイマ/カウンタ溢れ(TOV1)フラグはカウンタがBOTTOMに到達する時毎に設定(1)されます。OCR1AかICR1のどちらかがTOP値を定義するのに使われるとき、OCF1AまたはICF1割り込み要求フラグはOCR1xレジスタが(TOPに於いて)2重緩衝値で更新されるのと同じタイマ/カウンタクロック周期によって設定(1)されます。これらの割り込み要求フラグはカウンタがTOPまたはBOTTOM値に到達する毎に割り込みを発生するのに使えます。

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。固定TOP値を使う場合、どのOCR1xが書かれる時も、未使用ビットが0で隠(に置換)されることに注意してください。図47.で示される第3周期が図解するようにタイマ/カウンタが位相基準PWM動作で走行中にTOPを積極的に変更するのは、非対称出力で終わることが有り得ます。これに対する理由はOCR1xレジスタの更新時に見出せます。OCR1x更新はTOPで起きるので、PWM周期はTOPで始まりそして終わります。これは下降傾斜長が直前のTOP値によって決定され、一方上昇傾斜長は新しいTOP値で決定されることを意味します。これら2つの値(TOP)が違う時にその周期の2つの傾斜長は異なるでしょう。この長さの相違が出力での非対称な結果を生じます。

タイマ/カウンタが走行中にTOP値を変更する場合、位相基準PWM動作の代わりに位相/周波数基準PWM動作を使うことが推奨されます。一定のTOP値を使う時に2つの動作種別間に現実的な違いはありません。

位相基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます(70頁の表44.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

$$f_{OCnxPCPWM} = \frac{f_{clk_I/O}}{2 \times N \times TOP}$$

OCR1xの両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。

TOP値定義にOCR1Aが使われ(WGM13~0=1011)、COM1A1,0=01なら、OC1A出力はデューティ比50%で交互に変化します。

位相/周波数基準PWM動作

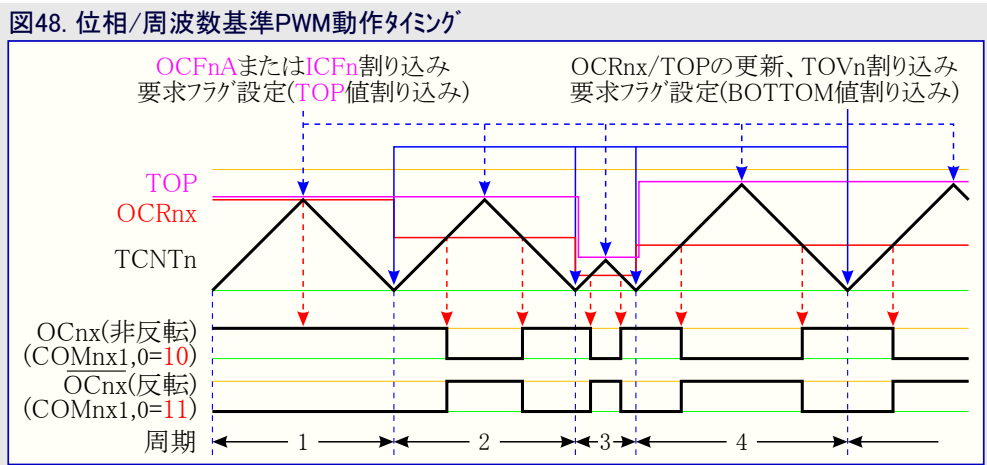
位相/周波数基準パルス幅変調(PWM)動作(WGM13~0=1000,1001)は高分解能で正しい位相と周波数のPWM波形生成選択を提供します。位相/周波数基準PWM動作は両傾斜(三角波)動作を基準とした**位相基準PWM**と似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後TOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準と位相/周波数基準PWM動作間の主な違いはOCR1xレジスタがOCR1x緩衝部によって更新される時(訳注:TOPとBOTTOM)です(図47と図48参照)。

位相/周波数基準PWM動作のPWM分解能はOCR1AかICR1のどちらかで定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は次式を使うことによって計算できます。

$$R_{PFCPWM} = \frac{\log(TOP+1)}{\log 2}$$

位相/周波数基準PWM動作でのタイマ/カウンタはタイマ/カウンタ値がICR1値(WGM13~0=1000)かOCR1A値(WGM13~0=1001)のどちらかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期、TOPと等しくなります。位相/周波数基準PWM動作のタイミング図は図48で示されます。この図はOCR1AかICR1がTOPを定義するのに使われる時の位相/周波数基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。



タイマ/カウンタ溢れ(TOV1)フラグはOCR1xレジスタが(BOTTOMに於いて)2重緩衝値で更新されるのと同じタイマ/カウンタクロック周期で設定(1)されます。OCR1AかICR1のどちらかがTOP値を定義するのに使われるとき、OCF1AまたはICF1割り込み要求フラグはタイマ/カウンタがTOPに到達する時毎に設定(1)されます。これらの割り込み要求フラグはカウンタがTOPまたはBOTTOM値に到達する毎に割り込みを発生するのに使えます。

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。

図48が示すように、生成された出力は位相基準PWM動作と異なり、全ての周期で対称です。OCR1xレジスタがBOTTOMで更新されるため、上昇と下降の傾斜長は常に等しくなります。これが対称出力パルス、従って正しい周波数を与えます。

TOPを定義するのにICR1を使うことは決まったTOP値を使う時に上手くいきます。ICR1を使うことにより、OC1AでのPWM出力を生成するためにOCR1Aが自由に使えます。けれども基準PWM周波数が(TOP値を変更することによって)動的に変更される場合、OCR1Aが2重緩衝機能のため、TOPとしてOCR1Aを使うことは明らかに良い選択です。

位相/周波数基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます(70頁の表44をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

位相/周波数基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

$$f_{OCnxPFCPWM} = \frac{f_{clk_1/O}}{2 \times N \times TOP}$$

OCR1xの両端値は位相/周波数基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。

TOP値定義にOCR1Aが使われ(WGM13~0=1001)、COM1A1,0=01なら、OC1A出力はデューティ比50%で交互に変化します。

タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従ってタイマ/カウンタクロック(clk_{T1})が下図のクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時、そしてOCR1xレジスタがOCR1x緩衝値で更新される時(2重緩衝部を使う動作種別のみ)の情報を含まず、図49はOCF1xの設定についてのタイミング図を示します。

図49. 前置分周なし(1/1)のタイマ/カウンタ、OCF1x設定 タイミング

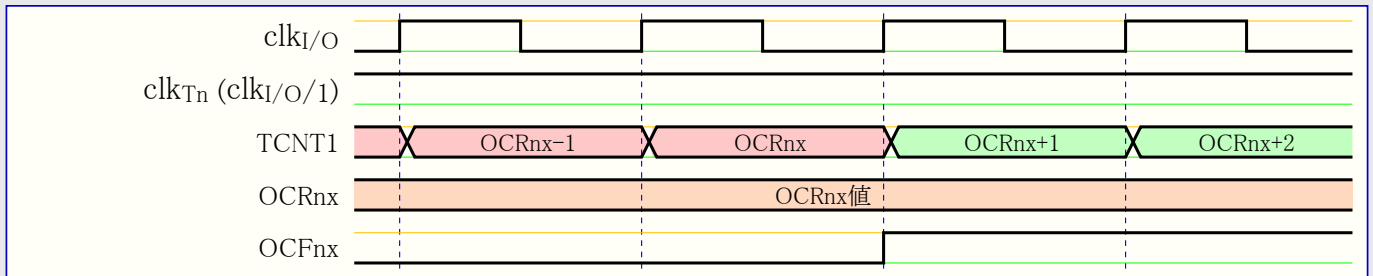


図50. は同じタイミング データを示しますが、前置分周器が許可されています。

図50. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、OCF1x設定 タイミング

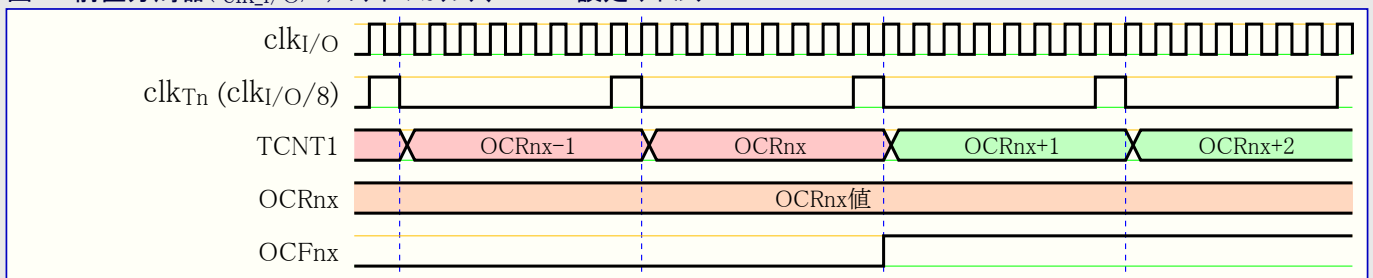


図51. は各動作種別でのTOP近辺の計数手順を示します。位相/周波数基準PWM動作使用時のOCR1xレジスタはBOTTOMで更新されます。タイミング図は同じになりますが、当然TOPはBOTTOMで、TOP-1はBOTTOM+1でなどのように置き換えられます。BOTTOMでTOV1を設定(1)する動作種別についても、同様な名称変更が適用されます。

図51. 前置分周器なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング

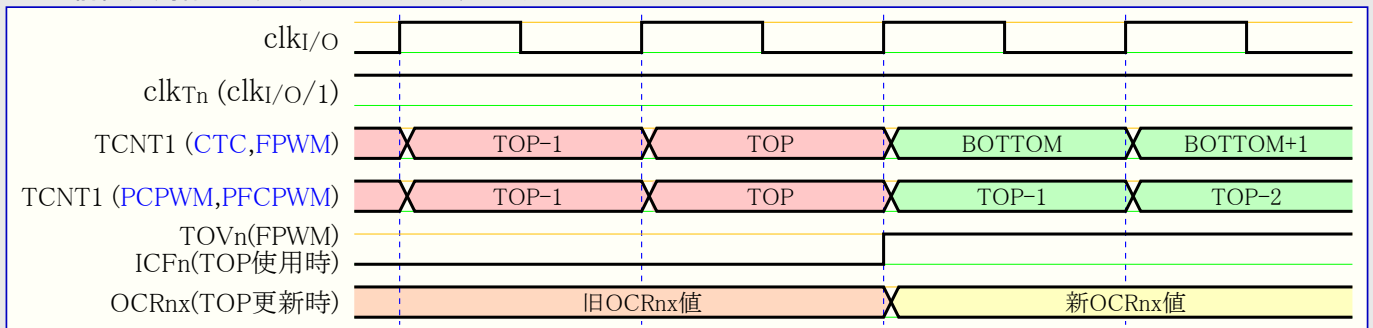
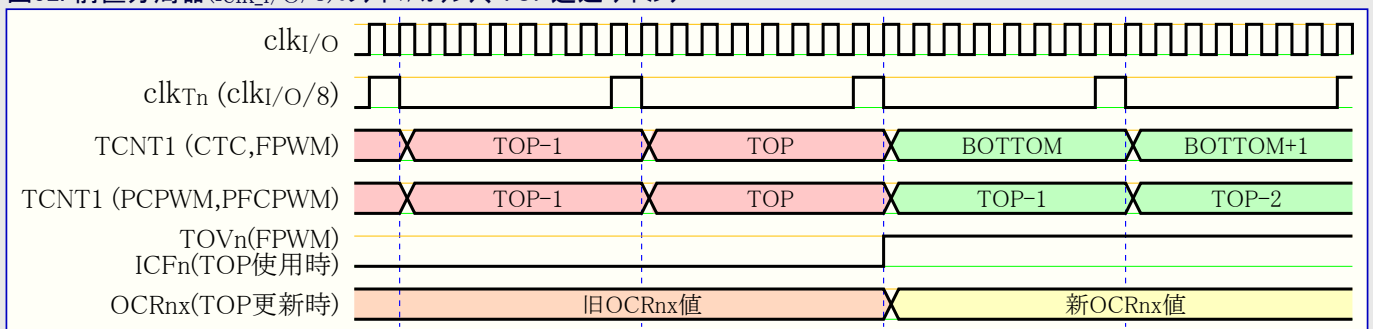


図52. は同じタイミング データを示しますが、前置分周器が許可されています。

図52. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、TOP近辺 タイミング



16ビット タイマ/カウンタ1用レジスタ

タイマ/カウンタ1制御レジスタA (Timer/Counter1 Control Register A) TCCR1A

ビット	7	6	5	4	3	2	1	0	
\$2F (\$4F)	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,6 – COM1A1,0 : 比較1A出力選択 (Compare Output Mode1A bit 1 and 0)
- ビット5,4 – COM1B1,0 : 比較1B出力選択 (Compare Output Mode1B bit 1 and 0)

COM1A1,0とCOM1B1,0は各々OC1AとOC1B比較出力ピンの動作を制御します。COM1A1,0ビットの1つまたは両方が1を書かれると、OC1A出力はそのI/Oピンの通常ポート機能を無効にし、そのI/Oピンに接続されます。COM1B1,0ビットの1つまたは両方が1を書かれると、OC1B出力はそのI/Oピンの通常ポート機能を無効にし、そのI/Oピンに接続されます。けれども出力駆動部を許可するため、OC1AまたはOC1Bピンに対応するポート方向レジスタ(DDR)のビットが設定(1)されなければならないことに注意してください。

OC1AまたはOC1Bがピンに接続されるとき、COM1x1,0ビットの機能はWGM13~0ビット設定に依存します。表42.はWGM13~0ビットが標準動作またはCTC動作(つまり非PWM)に設定される時のCOM1x1,0ビット機能を示します。

表42. 非PWM動作での比較出力選択 (注: xはAまたはB)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	比較一致でOC1xピントグル(交互)出力
1	0	比較一致でOC1xピン Lowレベル出力
1	1	比較一致でOC1xピン Highレベル出力

表43.はWGM13~0ビットが高速PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表43. 高速PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	WGM13~0=111X : 比較一致でOC1Aピントグル(交互)出力、OC1Bは標準ポート動作(OC1B切断) WGM13~0上記以外: 標準ポート動作 (OC1x切断)
1	0	比較一致でLow、BOTTOMでHighをOC1xピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC1xピンへ出力 (反転動作)

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMでの設定(1)または解除(0)は実行されます。より多くの詳細については66頁の「高速PWM動作」をご覧ください。

表44.はWGM13~0ビットが位相基準または位相/周波数基準PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表44. 位相基準または位相/周波数基準PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	WGM13~0=10X1 : 比較一致でOC1Aピントグル(交互)出力、OC1Bは標準ポート動作(OC1B切断) WGM13~0上記以外: 標準ポート動作 (OC1x切断)
1	0	上昇計数時の比較一致でLow、下降計数時の比較一致でHighをOC1xピンへ出力
1	1	上昇計数時の比較一致でHigh、下降計数時の比較一致でLowをOC1xピンへ出力

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。より多くの詳細については67頁の「位相基準PWM動作」をご覧ください。

•ビット1,0 – WGM11,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)

タイマ/カウンタ制御レジスタB(TCCR1B)で得られるWGM13,2ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大計数(TOP)値供給元、使われるべき波形生成のどの形式かを制御します(表45.参照)。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と3形式のパルス幅変調(PWM)動作です。65頁の「動作種別」をご覧ください。

表45. 波形生成種別選択

番号	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	タイマ/カウンタ動作種別	TOP値	OCR1x更新時	TOV1設定時
0	0	0	0	0	標準動作	\$FFFF	即値	MAX
1	0	0	0	1	8ビット位相基準PWM動作	\$00FF	TOP	BOTTOM
2	0	0	1	0	9ビット位相基準PWM動作	\$01FF	TOP	BOTTOM
3	0	0	1	1	10ビット位相基準PWM動作	\$03FF	TOP	BOTTOM
4	0	1	0	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR1A	即値	MAX
5	0	1	0	1	8ビット高速PWM動作	\$00FF	BOTTOM	TOP
6	0	1	1	0	9ビット高速PWM動作	\$01FF	BOTTOM	TOP
7	0	1	1	1	10ビット高速PWM動作	\$03FF	BOTTOM	TOP
8	1	0	0	0	位相/周波数基準PWM動作	ICR1	BOTTOM	BOTTOM
9	1	0	0	1	位相/周波数基準PWM動作	OCR1A	BOTTOM	BOTTOM
10	1	0	1	0	位相基準PWM動作	ICR1	TOP	BOTTOM
11	1	0	1	1	位相基準PWM動作	OCR1A	TOP	BOTTOM
12	1	1	0	0	比較一致タイマ/カウンタ解除(CTC)動作	ICR1	即値	MAX
13	1	1	0	1	(予約)	-	-	-
14	1	1	1	0	高速PWM動作	ICR1	BOTTOM	TOP
15	1	1	1	1	高速PWM動作	OCR1A	BOTTOM	TOP

注: CTC1とPWM11,0ビット定義名は旧名です。WGM12~0定義を使ってください。しかし、これらのビットの機能と位置は旧版のタイマ/カウンタと一致します。

タイマ/カウンタ1制御レジスタB (Timer/Counter1 Control Register B) TCCR1B

ビット	7	6	5	4	3	2	1	0	
\$2E (\$4E)	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – ICNC1 : 捕獲起動入力雑音消去許可 (Input Capture1 Noise Canceler)

このビットを(1)に設定することが捕獲起動入力雑音消去器を活性(有効)にします。雑音消去器が有効にされると、捕獲起動入力(ICP)ピンからの入力が濾波されます。この濾波機能はそれが出力を更新するのに、連続4回等しく評価されたICPピンの採取を必要とします。雑音消去器が許可されると、捕獲入力はそのによって4発振器(システムクロック)周期遅らされます。

•ビット6 – ICES1 : 捕獲起動入力端選択 (Input Capture1 Edge Select)

このビットは出来事での捕獲を起動するのに使われる捕獲起動入力(ICP)ピンのどちらかの端(エッジ)を選びます。ICES1ビットが0を書かれると起動動作として下降(負)端が使われ、ICES1ビットが1を書かれると上昇(正)端が捕獲を起動します。

捕獲がICES1設定に従って起動されると、カウンタ値が捕獲レジスタ(ICR1)に複写されます。この出来事は捕獲入力割り込み要求フラグ(ICF1)も設定(1)し、そしてこれは、この割り込みが許可されていれば捕獲入力割り込みを起こすのに使えます。

ICR1がTOP値として使われると(TCCR1AとTCCR1Bに配置されたWGM13~0ビットの記述をご覧ください)、ICPが切り離され、従って捕獲入力機能は禁止されます。

•ビット5 – Res : 予約 (Reserved)

このビットは将来の使用に対して予約されています。将来のデバイスとの共通性を保つため、TCCR1Bが書かれるとき、このビットは0を書かれなければなりません。

•ビット4,3 – WGM13,2 : 波形生成種別 (Waveform Generation Mode bit 3 and 2)

TCCR1AのWGM11,0ビット記述をご覧ください。



•ビット2~0 – CS12~0 : クロック選択1 (Clock Select1, bit 2,1 and 0)

この3つのクロック選択ビットはタイマ/カウンタ(TCNT1)によって使われるべきクロック元を選びます。図49.と図50.をご覧ください。

表46. タイマ/カウンタ1入力クロック選択

CS12	CS11	CS10	意味
0	0	0	停止 (タイマ/カウンタ1動作停止)
0	0	1	clk _{I/O} (前置分周なし)
0	1	0	clk _{I/O} /8 (8分周)
0	1	1	clk _{I/O} /64 (64分周)
1	0	0	clk _{I/O} /256 (256分周)
1	0	1	clk _{I/O} /1024 (1024分周)
1	1	0	T1ピンの下降端 (外部クロック)
1	1	1	T1ピンの上昇端 (外部クロック)

タイマ/カウンタ1に対して外部ピン(クロック)動作が使われる場合、例えT1ピンが出力として設定されても、T1ピンの遷移はカウンタをクロック駆動します。この特性はソフトウェアに計数制御を許します。

タイマ/カウンタ1制御レジスタC (Timer/Counter1 Control Register C) TCCR1C

ビット	7	6	5	4	3	2	1	0	
\$22 (\$42)	FOC1A	FOC1B	-	-	-	-	-	-	TCCR1C
Read/Write	W	W	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – FOC1A : OC1A強制変更 (Force Output Compare 1A)

•ビット6 – FOC1B : OC1B強制変更 (Force Output Compare 1B)

FOC1A/FOC1BビットはWGM13~0ビットが非PWM動作を指示する時だけ有効です。けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR1Cが書かれる場合、これらのビットは0に設定されなければなりません。FOC1A/FOC1Bビットに論理1を書くと波形生成部で直ちに比較一致が強制されます。OC1x出力はCOM1x1,0ビット設定に従って変更されます。FOC1A/FOC1Bビットがスロープとして実行されることに注意してください。それによって強制された比較の効果を決めるのはCOM1x1,0ビットに存在する値です。

FOC1A/FOC1Bスロープは何れの割り込みの生成もTOPとしてOCR1Aを使う比較一致タイマ解除(CTC)動作でのタイマ/カウンタの解除(\$0000)も行いません。

FOC1A/FOC1Bビットは常に0として読みます。

タイマ/カウンタ1 (Timer/Counter1) TCNT1H,TCNT1L (TCNT1)

ビット	15	14	13	12	11	10	9	8	
\$2D (\$4D)	(MSB)								TCNT1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$2C (\$4C)								(LSB)	TCNT1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この2つのタイマ/カウンタ I/O位置(TCNT1HとTCNT1Lを合わせたTCNT1)は、読み書き両方についてタイマ/カウンタ部の16ビットカウンタに直接アクセスします。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。58頁の「16ビットレジスタのアクセス」をご覧ください。

カウンタが走行中にカウンタ(TCNT1)を変更することはOCR1xの1つとTCNT1間の比較一致消失の危険を誘発します。

TCNT1への書き込みは全ての比較部に対して次のタイマ/カウンタクロックでの比較一致を妨害(除去)します。

タイマ/カウンタ1 比較レジスタ (Timer/Counter1 Output Compare Register A) OCR1AH,OCR1AL (OCR1A)

ビット	15	14	13	12	11	10	9	8	
\$2B (\$4B)	(MSB)								OCR1AH
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$2A (\$4A)								(LSB)	OCR1AL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

タイマ/カウンタ1 比較レジスタ (Timer/Counter1 Output Compare Register B) OCR1BH,OCR1BL (OCR1B)

ビット	15	14	13	12	11	10	9	8	
\$29 (\$49)	(MSB)								OCR1BH
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$28 (\$48)								(LSB)	OCR1BL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この比較レジスタは継続的にカウンタ(TCNT1)値と比較される16ビット値を含みます。一致は比較一致割り込みやOC1xピンでの波形出力を生成するのに使えます。

この比較レジスタは容量が16ビットです。CPUがこれらのレジスタへ書く時に上位と下位の両バイトが同時に書かれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。58頁の「16ビットレジスタのアクセス」をご覧ください。

タイマ/カウンタ1 捕獲レジスタ (Timer/Counter1 Input Capture Register) ICR1H,ICR1L (ICR1)

ビット	15	14	13	12	11	10	9	8	
\$25 (\$45)	(MSB)								ICR1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$24 (\$44)								(LSB)	ICR1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この捕獲レジスタはICPピン(またはタイマ/カウンタ1については任意のアナログ比較器出力)で出来事が起こる毎にカウンタ(TCNT1)値で更新されます。この捕獲レジスタはタイマ/カウンタのTOP値を定義するのに使えます。

この捕獲レジスタは容量が16ビットです。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読まれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。58頁の「16ビットレジスタのアクセス」をご覧ください。

タイマ/カウンタ割り込み許可レジスタ (Timer/Counter Interrupt Mask Register) TIMSK

ビット	7	6	5	4	3	2	1	0	
\$39 (\$59)	TOIE1	OCIE1A	OCIE1B	-	ICIE1	OCIE0B	TOIE0	OCIE0A	TIMSK
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット7 – TOIE1 : タイマ/カウンタ1溢れ割り込み許可 (Timer/Counter1 Overflow Interrupt Enable)

このビットが1を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1溢れ割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置されたタイマ/カウンタ1溢れ割り込み要求フラグ(TOV1)が設定(1)されると、対応する割り込みベクタ(29頁の「割り込み」参照)が実行されます。

• ビット6 – OCIE1A : タイマ/カウンタ1比較A割り込み許可 (Timer/Counter1 Output Compare A Match Interrupt Enable)

このビットが1を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1比較A一致割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置された比較1A割り込み要求フラグ(OCF1A)が設定(1)されると、対応する割り込みベクタ(29頁の「割り込み」参照)が実行されます。

• ビット5 – OCIE1B : タイマ/カウンタ1比較B割り込み許可 (Timer/Counter1 Output Compare B Match Interrupt Enable)

このビットが1を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1比較B一致割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置された比較1B割り込み要求フラグ(OCF1B)が設定(1)されると、対応する割り込みベクタ(29頁の「割り込み」参照)が実行されます。

• ビット3 – ICIE1 : タイマ/カウンタ1捕獲割り込み許可 (Timer/Counter1 Input Capture Interrupt Enable)

このビットが1を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1捕獲割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置された捕獲割り込み要求フラグ(ICF1)が設定(1)されると、対応する割り込みベクタ(29頁の「割り込み」参照)が実行されます。

タイマ/カウンタ割り込み要求フラグレジスタ (Timer/Counter Interrupt Flag Register) TIFR

ビット	7	6	5	4	3	2	1	0	
\$38 (\$58)	TOV1	OCF1A	OCF1B	-	ICF1	OCF0B	TOV0	OCF0A	TIFR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット7 – TOV1 : タイマ/カウンタ1溢れ割り込み要求フラグ (Timer/Counter1 Overflow Flag)

このフラグの(1)設定はWGM13~0ビット設定に依存します。標準またはCTC動作でのTOV1フラグはタイマ/カウンタ1溢れ時に設定(1)されます。他のWGM13~0ビット設定を使う時のTOV1フラグ動作については71頁の表45を参照してください。

タイマ/カウンタ1溢れ割り込みベクタが実行されると、TOV1は自動的に解除(0)されます。代わりにこのビット位置へ論理1を書くことによってもTOV1は解除(0)できます。

• ビット6 – OCF1A : タイマ/カウンタ1比較A割り込み要求フラグ (Timer/Counter1, Output Compare A Match Flag)

このフラグはカウンタ(TCNT1)値が比較Aレジスタ(OCR1A)と一致した後(次)のタイマ/カウンタクロック周期で設定(1)されます。

強制的な比較出力(FOC1A)ストロブがOCF1Aフラグを設定(1)しないことに注意してください。

比較A一致割り込みベクタが実行されると、OCF1Aは自動的に解除(0)されます。代わりにこのビット位置へ論理1を書くことによってもOCF1Aは解除(0)できます。

• ビット5 – OCF1B : タイマ/カウンタ1比較B割り込み要求フラグ (Timer/Counter1, Output Compare B Match Flag)

このフラグはカウンタ(TCNT1)値が比較Bレジスタ(OCR1B)と一致した後(次)のタイマ/カウンタクロック周期で設定(1)されます。

強制的な比較出力(FOC1B)ストロブがOCF1Bフラグを設定(1)しないことに注意してください。

比較B一致割り込みベクタが実行されると、OCF1Bは自動的に解除(0)されます。代わりにこのビット位置へ論理1を書くことによってもOCF1Bは解除(0)できます。

• ビット3 – ICF1 : タイマ/カウンタ1捕獲割り込み要求フラグ (Timer/Counter1, Input Capture Flag)

ICPピンに捕獲の事象が起こると、このフラグが設定(1)されます。捕獲レジスタ(ICR1)がWGM13~0によってTOP値として設定されると、ICF1フラグはカウンタがTOP値に到達する時に設定(1)されます。

捕獲割り込みベクタが実行されると、ICF1は自動的に解除(0)されます。代わりにこのビット位置へ論理1を書くことによってもICF1は解除(0)できます。

USART

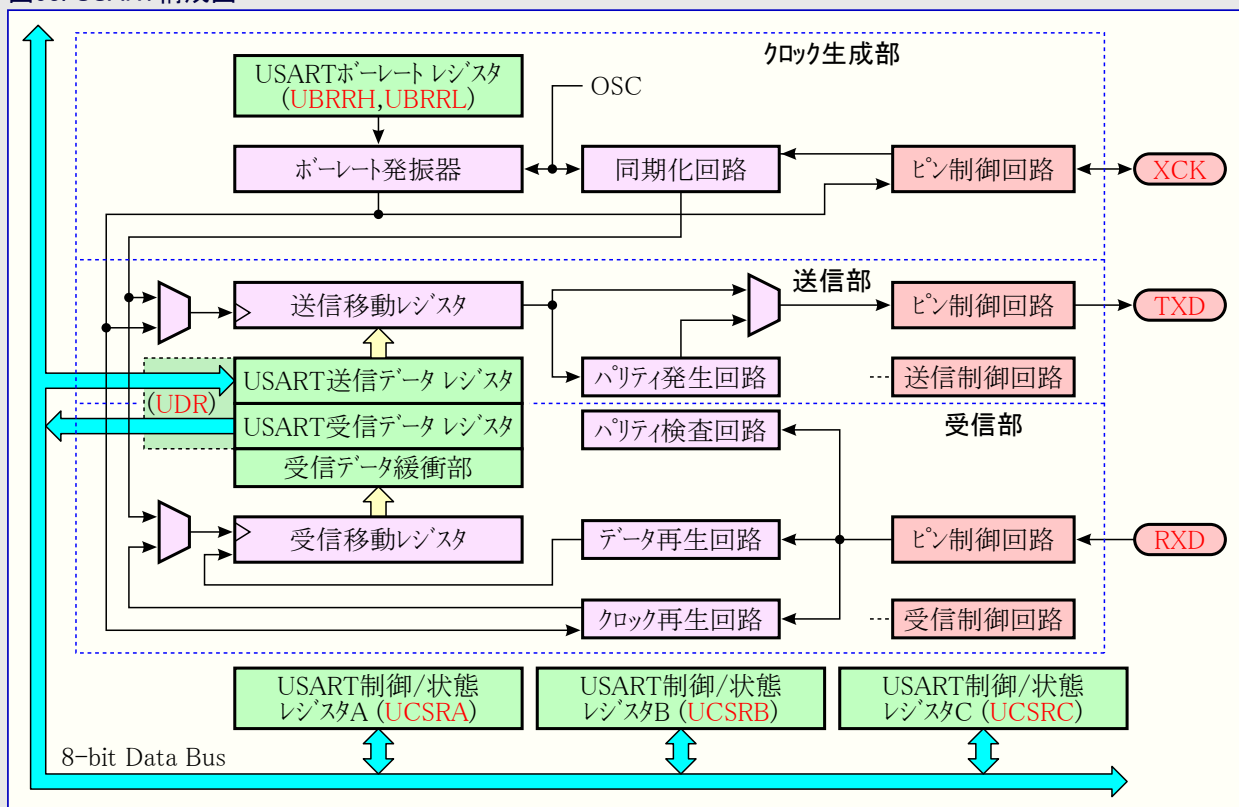
USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は高い柔軟性をもつ直列通信機能です。主な特徴を次に示します。

- 全二重動作 (独立した送受信レジスタ)
- 同期または非同期動作
- 同期クロック駆動された主装置/従装置動作
- 高分解能ボーレート発振器
- 5, 6, 7, 8または9ビットデータと1または2停止ビットの直列フレームの支援
- ハードウェアによって支援された奇数または偶数パリティの生成と検査
- データオーバーラン検出
- フレーミング異常検出
- 不正開始ビット検出とデジタル低域通過濾波器を含む雑音除去器
- 受信完了、送信完了、送信データレジスタ空きの3つの分離した割り込み
- 複数プロセッサ通信機能
- 倍速非同期通信動作

概要

USARTの簡便化した構成図は図53.で示されます。CPUがアクセス可能なレジスタとI/Oピンは赤文字(訳注:原文は太字)で示されます。

図53. USART構成図



注: USARTピン配置については2頁の「ピン配置」、38頁の表29.を参照してください。

構成図内の破線はUSARTの3つの主要部分、(上から)クロック生成部、送信部、受信部を分けます。制御レジスタは全部によって共有されます。クロック生成論理部はボーレート発振器と同期従装置動作によって使われる外部クロック入力に対する同期化論理回路から成ります。転送クロック(XCK)ピンは同期転送動作だけで使われます。送信部は単一書き込み緩衝部(UDR)、直列移動レジスタ、パリティ発生器、異なる直列フレーム形式を扱うための制御論理回路から成ります。書き込み緩衝部はどんなフレーム間の遅れもなくデータの継続転送を許します。受信部はクロックとデータの再生部のため、USART部の最も複雑な部分です。再生部は非同期データ受信で使われます。再生部に加えて、受信部は2重の受信緩衝部(UDR)、移動レジスタ、パリティ検査器、制御論理回路を含みます。受信部は送信部と同じフレーム形式を支援し、フレーミング異常、データオーバーラン発生、パリティ誤りを検知できます。

UARTとの互換性

USARTはAVRのUARTと次の項目に関して完全な互換性があります。

- 全てのUSART側レジスタでのビット位置
- ボーレート生成
- 送信操作
- 送信緩衝の動作
- 受信操作

けれども受信緩衝動作にはいくつかの特別な場合で互換性に影響を及ぼす2つの改良点があります。

- 第2受信緩衝部が追加されました。2つの緩衝レジスタは循環型FIFO緩衝部として動作します。従ってUSARTデータレジスタ(UDR)は到着データ毎に一度だけ読まれなければなりません。より重要なのは異常フラグ(FEとDOR)と第9データビット(RXB8)が受信緩衝部内のデータと共に緩衝されることの実事です。従って状態ビットは常にUSARTデータレジスタ(UDR)が読まれる前に読まれなければなりません。さもなければ緩衝部の状態が失われるため、異常情報も失われます。
- 受信部移動レジスタは第3緩衝段のように動けます。これは緩衝レジスタが一杯の場合、新規開始ビットが検出されるまで直列移動レジスタ(図53参照)内に留まるのを受信したデータに許すことによって行われます。従ってUSARTはデータオーバーラン(DOR)異常条件により耐えます。

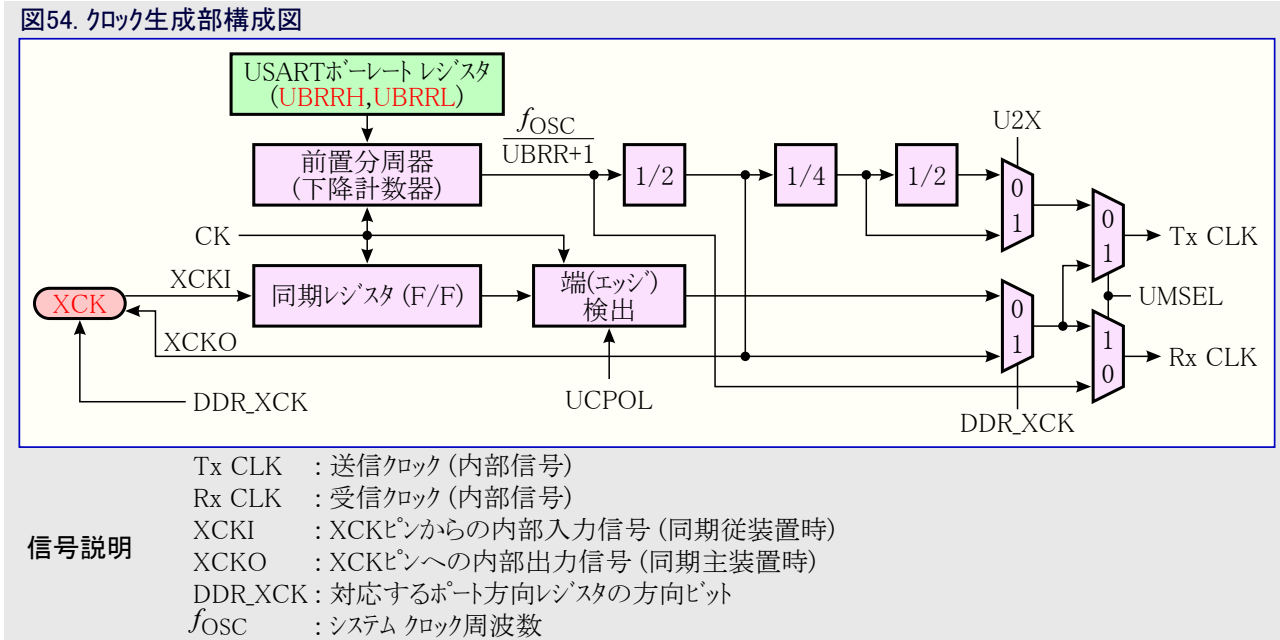
次の制御ビットは名称変更されましたが、機能とレジスタ(ビット)位置は同じです。

- 9ビット選択(CHR9)ビットはデータ長選択ビット2(UCSZ2)へ変更
- オーバーラン発生(OR)フラグはオーバーラン発生(DOR)フラグへ変更
- パリティ誤り(PE)フラグはパリティ誤り(UPE)フラグへ変更 (訳注:原書で欠落のため追加)

クロック生成

クロック生成論理回路は送受信部用基準クロックを生成します。USARTは標準非同期、倍速非同期、同期主装置、同期従装置の4つのクロック動作種別を支援します。USART制御/状態レジスタC(UCSRC)のUSART動作種別選択(UMSEL)ビットは同期動作と非同期動作のどちらかを選びます。倍速動作(非同期動作のみ)はUSART制御/状態レジスタA(UCSRA)にある倍速許可(U2X)ビットによって制御されます。同期動作(UMSEL=1)を使うとき、XCKピンに対する方向制御ビット(DDR_XCK)はクロック元が内部(主装置動作)または外部(従装置動作)のどちらかを制御します。このXCKピンは同期動作を使う時だけ活性(有効)です。

図54.はクロック生成論理回路の構成図を示します。



ボーレート発振器での内部クロック発生

内部クロック生成は非同期と同期主装置動作種別に対して使われます。本項の記述は図54.を参照してください。

USARTボーレートレジスタ(UBRR(UBRRH:UBRRL))と下降カウンタは設定変更可能な前置分周器またはボーレート発振器として機能するように接続されます。システムクロック(f_{OSC})で走行する下降カウンタは0への下降計数時毎またはUBRRLレジスタが書かれる時にUBRR値で設定されます。1クロック(パルス)はカウンタが0に達する毎に生成されます。このクロックがボーレート発振器出力($=f_{OSC}/(UBRR+1)$)です。送信部は動作種別に依存してボーレート発振器出力を2,8,16分周します。ボーレート発振器出力は受信部クロックとデータ再生部によって直接使われます。しかし、再生部はUSART動作種別選択(UMSEL)、倍速許可(U2X)、DDR_XCKビットの状態によって設定される動作種別に依存して2,8,16段を使う順次処理回路を使います。

表47.は内部的に生成したクロック元を使う各動作種別に於けるボーレート(bps)とUBRR値の計算式を含みます。

表47. ボーレートレジスタ(UBRR)値計算式

動作種別	ボーレート計算式	UBRR値計算式
標準速非同期動作 (U2X=0)	$BAUD = \frac{f_{OSC}}{16 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{16 \times BAUD} - 1$
倍速非同期動作 (U2X=1)	$BAUD = \frac{f_{OSC}}{8 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{8 \times BAUD} - 1$
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{2 \times BAUD} - 1$

注: ボーレートは転送速度(ビット/1秒)で定義されます。

- BAUD : ボーレート (bps)
- UBRR : UBRRHとUBRRLレジスタ値 (0~4095)
- f_{OSC} : システム発振器クロック周波数

いくつかのシステムクロック周波数に対するいくつかのUBRR値の例は表55.で得られます(90~91頁参照)。

倍速動作 (U2X)

転送速度はUSART制御/状態レジスタA(UCSRA)で倍速許可(U2X)ビットを設定(1)することによって倍にできます。このビットの設定は非同期動作に対してだけ有効です。同期動作を使うとき、このビットは0に設定してください。

このビットの設定(1)は事実上、非同期通信に対する転送速度を倍にするボーレート分周器の分周数を16から8に減じます。けれども受信部がデータ採取とクロック再生に対して半分の(16から8に減じた)採取数をこの場合だけ使い、従ってこの動作種別が使われる時はシステムクロックとボーレート設定の精度がより必要とされることに注意してください。送信部についての低下要因はありません。

外部クロック

外部クロックは同期従装置動作種別によって使われます。本項の記述での詳細については図54を参照してください。

XCKピンからの外部クロック入力是不確定レベル状態(マスタペル)の機会を最少とするために同期化レジスタによって採取されます。同期化レジスタからの出力は送受信部で使い得るのに先立って端(エッジ)検出器を通過しなければなりません。この処理手順が2 CPUクロック周期の遅延を持ち込み、このため最大外部XCKクロック周波数は次式によって制限されます。

$$f_{XCK} < \frac{f_{OSC}}{4}$$

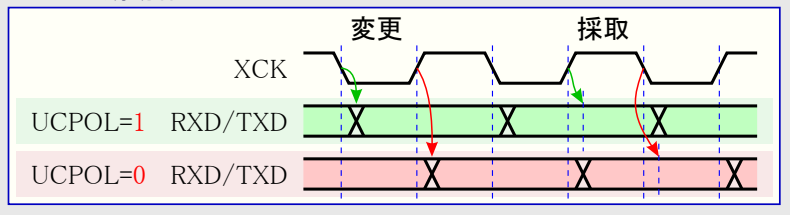
f_{OSC} がシステムクロック元の安定度に依存することに注意してください。従って周波数変動によるデータ消失の可能性を避けるため、いくらかの余裕分を追加することが推奨されます。

同期クロック動作

同期動作が使われる(UMSEL=1)とき、XCKピンはクロック入力(従装置)またはクロック出力(主装置)のどちらかとして使われます。データ採取またはデータ変更とクロック端間の依存性は同じです。基本原則はデータ出力(TXD)が変更される端と反対のXCKクロック端でデータ入力(RXD)が採取されることです。

USART制御/状態レジスタC(UCSRC)のXCK極性(UCPOL)ビットはデータ採取とデータ変更に対してどちらのクロック端が使われるかを選びます。図55.で示されるようにUCPOLが0のとき、データはXCKの上昇端で変更され、下降端で採取されます。UCPOLが設定(1)の場合、データはXCKの下降端で変更され、上昇端で採取されます。

図55. 同期動作XCKタイミング



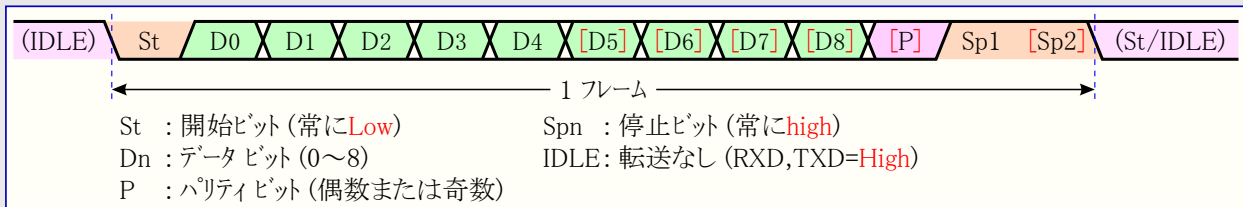
フレーム形式

1つの直列フレームは複数のデータビットと同期ビット(開始ビット、停止ビット)、任意の異常検査用パリティビットで定義されます。USARTは有効なフレーム形式として以下の組み合わせ30種全てを受け入れます。

- 1 開始ビット
- 5, 6, 7, 8, 9 ビット データ
- 奇数または偶数パリティビット、またはなし
- 1 または 2 停止ビット

フレームは最下位データビット(LSB)が次に続く開始ビットで始まります。その後次にデータビットが最後の最上位データビット(MSB)まで(最大合計9ビット続きます。許可したなら、パリティビットがデータビットの後、停止ビットの前に挿入されます。完全なフレームが送信されると、新規フレームによって直ちに後続されるか、または通信線をアイドル状態(high)に設定できます。図56.は組み合わせ可能なフレーム形式を図解します。[]付きビットは任意選択です。

図56. 17フレームの構成



USARTによって使われるフレーム形式はUSART制御/状態レジスタBとC(UCSRB, UCSRC)でデータ長選択(UCSZ2~0)ビット、パリティ選択(UPM1,0)ビット、停止ビット選択(USBS)ビットによって設定されます。受信部と送信部は同じ設定を使います。これらのどのビットの設定変更も、送受信部両方に対して進行中の通信を不正とすることに注意してください。

データ長選択(UCSZ2~0)ビットはフレーム内のデータビット数を選びます。パリティ選択(UPM1,0)ビットはパリティビットの許可と種別(奇/偶)を設定します。1または2停止ビットのどちらかの選択は停止ビット選択(USBS)ビットによって行います。受信部は第2停止ビットを無視します。従ってフレーミング異常(FE)は最初の停止ビットが0(Low)の場合にだけ検出されます。

パリティビットの計算

パリティビットは全データビットの排他的論理和(Ex-OR)を行うことによって計算されます。奇数パリティが使われる場合は排他的論理和の結果が反転されます。パリティビットとデータビットの関係は次のとおりです。

$$\begin{aligned} \text{偶数パリティビット} &= D_0 \text{ Ex-OR } D_1 \text{ Ex-OR } D_2 \text{ Ex-OR } \sim \text{Ex-OR } D_{n-1} \\ \text{奇数パリティビット} &= D_0 \text{ Ex-OR } D_1 \text{ Ex-OR } D_2 \text{ Ex-OR } \sim \text{Ex-OR } D_{n-1} \text{ Ex-OR } 1 \end{aligned} \quad n: \text{データビット長}$$

使った場合、パリティビットは直列フレームの最後のデータビットと最初の停止ビット間に配置されます。

USARTの初期化

何れかの通信が行えるのに先立ってUSARTは初期化されなければなりません。標準的な初期化手順は使用方法に依存するボーレート設定、フレーム形式設定、送受信部許可から成ります。割り込み駆動USART操作に関して初期化を行うとき、[ステータスレジスタの全割り込み許可\(0\)ビット](#)は解除(0)される(そして全割り込みが禁止される)べきです。

ボーレートまたはフレーム形式の変更を伴う再初期化を行う前には、レジスタが変更される期間中に進行中の送信がないことを確実にしてください。[USART制御/状態レジスタA\(UCSRA\)の送信完了\(TXC\)フラグ](#)は送信部の全転送完了検査に使い、[受信完了\(RXC\)フラグ](#)は受信緩衝部内の未読データ有無検査に使えます。この目的(次送信タイミング)にTXCフラグが使われる場合、各々の送信([USARTデータレジスタ\(UDR\)](#)が書かれる)前にTXCフラグが解除(0)されなければならないことに注意してください。

次の簡単なUSART初期化コード例が示すアセンブリ言語とC言語の関数は機能的に同じです。この例は固定フレーム形式でボーリングを使う(割り込み不許可)非同期動作と仮定します。ボーレート(UBRR)値は関数の引数として与えられます。アセンブリ言語でのボーレート引数はR17:R16レジスタに格納されると仮定されます。

アセンブリ言語プログラム例

```
USART_Init:  OUT    UBRRH, R17                ;ボーレート設定(上位バイト)
             OUT    UBRRL, R16             ;ボーレート設定(下位バイト)
             LDI    R16, (1<<USBS) | (3<<UCSZ0) ;フレーム形式値を取得
             OUT    UCSRC, R16             ;フレーム形式設定(8ビット,2停止ビット)
             LDI    R16, (1<<RXEN) | (1<<TXEN) ;送受信許可値を取得
             OUT    UCSRB, R16             ;送受信許可
             RET                             ;呼び出し元へ復帰
```

C言語プログラム例

```
#define FOSC 1843200 /* MCUクロック周波数 */
#define BAUD 9600 /* 目的USARTボーレート速度 */
#define MYUBRR FOSC/16/BAUD-1 /* 目的UBRR値 */

void main(void)
{
    ~
    USART_Init(MYUBRR); /* USART初期化 */
    ~
}

void USART_Init(unsigned int baud)
{
    UBRRH = (unsigned char) (baud>>8); /* ボーレート設定(上位バイト) */
    UBRRL = (unsigned char) baud; /* ボーレート設定(下位バイト) */
    UCSRC = (1<<USBS) | (3<<UCSZ0); /* フレーム形式設定(8ビット,2停止ビット) */
    UCSRB = (1<<RXEN) | (1<<TXEN); /* 送受信許可 */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

割り込みの禁止や引数としてフレーム形式を含めるなどで、より進化した初期化ルーチンが作成できます。けれども多くの応用はボーレートや制御レジスタの固定した設定が使われ、これらの応用形式での初期化コードは主ルーチンに直接置けるか、または他のI/Oの初期化コードと併せられます。

USARTのデータ送信

USART送信部はUSART制御/状態レジスタB(UCSRB)で送信許可(TXEN)ビットを設定(1)することによって許可されます。送信部が許可されると、TXDピンの標準ピン動作はUSARTによって無視され、送信部の直列出力としての機能を与えられます。何か送信を行う前に一度はホーレート、フレーム形式、動作種別が設定されなければなりません。同期動作が使われる場合、XCKピンの(受信)クロックは無視され、送信クロックとして使われます。

5～8ビット データ フレーム送信

データ送信は送信されるべきデータを送信緩衝部に設定することによって開始されます。CPUはUSARTデータレジスタ(UDR)I/O位置へ書くことによって送信緩衝部に設定できます。送信緩衝部内のデータは移動レジスタが新規フレームを送る準備が整った時に移動レジスタへ移されます。移動レジスタはアイドル状態(送信進行中以外)、または直前のフレームの最後の停止ビット送信後、直ちに新規データが設定されます。移動レジスタが新規データを設定されると、ホーレートレジスタ(UBRRH:UBRRL)と倍速許可(U2X)ビット、また動作種別によってはXCKピンによって与えられる速度で1つの完全なフレームを転送します。8ビット未満のフレームを使うとき、UDRに書かれた上位ビットは無視されます。

次のコード例はUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグのポーリングを基準とした簡単なUSART送信関数を示します。この関数が使われ得る前にUSARTが初期化されなければなりません。アセンブリ言語での送るべきデータはR16レジスタに格納されると仮定されます。

アセンブリ言語プログラム例

```
USART_Tx:  SBIS    UCSRA, UDRE          ;送信緩衝部空きでスキップ
           RJMP   USART_Tx           ;送信緩衝部空き待機
;
           OUT    UDR, R16           ;データ送信(送信開始)
           RET                                ;呼び出し元へ復帰
```

C言語プログラム例

```
void USART_Transmit(unsigned char data)
{
    while ( !(UCSRA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UDR = data;                          /* データ送信(送信開始) */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

この関数は送信されるべき新規データを設定する前に、UDREの検査によって送信緩衝部が空になるのを単純に待ちます。送信緩衝部空き割り込みが使われる場合、その割り込み処理ルーチンがデータを緩衝部内に書きます。

9ビット データ フレーム送信

9ビットデータが使われる場合(UCSZ2~0=111)、データの下位バイトがUSARTデータレジスタ(UDR)に書かれる前に、第9ビットがUSART制御/状態レジスタB(UCSRB)の送信データビット8(TXB8)ビットに書かれなければなりません。次のコード例は9ビットデータを扱う送信関数を示します。アセンブリ言語での送るべきデータはR17:R16レジスタに格納されると仮定されます。

アセンブリ言語プログラム例

```
USART_Tx:  SBIS    UCSRA, UDRE          ;送信緩衝部空きでスキップ
           RJMP   USART_Tx           ;送信緩衝部空き待機
;
           CBI    UCSRB, TXB8         ;第9ビットを0に仮設定
           SBRC   R17, 0              ;送信すべき第9ビットが0でスキップ
           SBI    UCSRB, TXB8         ;第9ビットを1に設定
           OUT    UDR, R16           ;データ送信(送信開始)
           RET                                ;呼び出し元へ復帰
```

C言語プログラム例

```
void USART_Transmit(unsigned int data)
{
    while ( !(UCSRA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UCSRB &= ~(1<<TXB8);               /* TXB8を0に仮設定 */
    if (data & 0x0100) UCSRB |= (1<<TXB8); /* 第9ビットをR17からTXB8へ複写 */
    UDR = data;                          /* データ送信(送信開始) */
}
```

注: これらの送信関数は一般的な機能で書かれています。UCSRBの内容が静的(換言すると、UCSRBのTXB8ビットが初期化後に使われるだけ)ならば最適化できます。

このコード例はデバイス定義ファイルがインクルードされることが前提です。

第9ビットは複数プロセッサ通信使用時のアドレスフレーム識別、また例えば同期として扱う他の規約で使うことができます。

送信フラグと割り込み

USART送信部には状態を示す2つのフラグ、USARTデータレジスタ空き(UDRE)と送信完了(TXC)があります。両フラグは割り込みを発生するのに使えます。

USARTデータレジスタ空き(UDRE)フラグは送信緩衝部が新規データを受け取る準備ができていないかどうかを示します。このビットは送信緩衝部が空の時に設定(1)され、送信緩衝部が移動レジスタに未だ移動されてしまっていない送信されるべきデータを含む時に解除(0)されます。将来のデバイスとの共通性のため、**USART制御/状態レジスタA(UCSRA)**に書くとき、常にこのビットに0を書いてください。

USART制御/状態レジスタB(UCSRB)で**データレジスタ空き割り込み許可(UDRIE)ビット**が1を書かれると、(全割り込みが許可されているならば)UDREフラグが設定(1)されている限り、USARTデータレジスタ空き割り込みが実行されます。UDREは**USARTデータレジスタ(UDR)**書き込みによって解除(0)されます。割り込み駆動データ送信が使われるとき、データレジスタ空き割り込みルーチンはUDREを解除(0)するために新規データをUDRに書くか、データレジスタ空き割り込みを禁止するかのどちらかを行わなければならない、さもなければ一旦割り込みルーチンを終了しても新しい割り込みが(継続的に)起こります。

送信完了(TXC)フラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部に新規データが現在存在しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行される時、自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)できます。TXCフラグは送信応用プログラムが送信完了後、直ちに通信回線を開放し、受信動作へ移行しなければならない(RS485規格のような)半二重(ハーフデュプレックス)通信インターフェースで有用です。

UCSRBで**送信完了割り込み許可(TXCIE)ビット**が設定(1)され、(全割り込みが許可されていれば)TXCフラグが設定(1)になる時にUSART送信完了割り込みが実行されます。送信完了割り込みが使われるとき、割り込み処理ルーチンはTXCフラグを解除(0)しなくてもよく、これは割り込みが実行される時、自動的に行われます。

パリティ発生器

パリティ発生器は直列フレームデータに対する**パリティビットを計算**します。パリティビットが許可されると(**UPM1=1**)、送信部制御論理回路は送られているフレームの最終データビットと最初の停止ビット間にパリティビットを挿入します。

送信の禁止

送信部の禁止(UCSRBの**USART送信許可(TXEN)ビット=0**)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない時)まで有効になりません。禁止されると、送信部はもはやTXDピンの(標準ピン機能)を無効にしません。

USARTのデータ受信

USART受信部は**USART制御/状態レジスタB(UCSRB)**で**受信許可(RXEN)ビット**に1を書くことによって許可されます。受信部が許可されると、RXDピンの標準ピン動作はUSARTによって無視され、受信部の直列入力としての機能を与えられます。何か直列受信が行われ得る前に一度はポーレート、フレーム形式、動作種別が設定されなければなりません。同期動作が使われる場合、XCKピンのクロックは転送クロックとして使われます。

5~8ビットデータフレーム受信

受信部は有効な開始ビットを検出する時にデータ受信を開始します。開始ビットに続く各ビットはポーレートまたはXCKクロックで採取され、フレームの最初の停止ビットが受信されるまで受信移動レジスタ内へ移動されます。第2停止ビットは受信部によって無視されます。最初の停止ビットが受信されると(換言すると、受信移動レジスタに完全なフレームが存在すると)、この移動レジスタの内容は受信緩衝部内へ移されます。受信緩衝部はUSARTデータレジスタ(UDR)I/O位置を読むことによって読めます。8ビット未満のフレームを使うとき、UDRから読むデータの上位ビットは0で覆われます。

次のコード例は**USART制御/状態レジスタA(UCSRA)**の**受信完了(RXC)フラグ**のポーリングを基にした簡単なUSART受信関数を示します。この関数が使われ得る前にUSARTが初期化されなければなりません。

アセンブリ言語プログラム例

```
USART_Rx:  SBIS    UCSRA, RXC          ;受信完了でスキップ
           RJMP    USART_Rx         ;受信完了待機
;
           IN     R16, UDR          ;受信データ取得
           RET                    ;呼び出し元へ復帰
```

C言語プログラム例

```
unsigned char USART_Receive(void)
{
    while ( !(UCSRA & (1<<RXC)) ); /* 受信完了待機 */
    return UDR;                    /* 受信データ取得 */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

この関数は緩衝部を読んで値を戻す前に、RXCフラグの検査によって受信緩衝部にデータが存在するのを単純に待ちます。

9ビットデータフレーム受信

9ビットデータが使われる場合(UCSZ2~0=111)、USARTデータレジスタ(UDR)から下位バイトを読む前に、第9ビットがUSART制御/状態レジスタB(UCSRB)の受信データビット8(RXB8)ビットから読まなければならない。この規則はフレーミング異常(FE)、オーバーラン発生(DOR)、パリティ誤り(UPE)状態フラグにも適用します。USART制御/状態レジスタA(UCSRA)から状態情報を読み、その後UDRからデータを読んでもください。UDR I/O位置を読むことが受信FIFO緩衝部の状態を切り替え、その結果、FIFO内に保管されるRXB8,FE,DOR,UPEビット全てが切り替わります。

次のコード例は9ビットデータと状態ビット両方を扱う簡単なUSART受信関数を示します。

アセンブリ言語プログラム例

```

USART_Rx:  SBIS   UCSRA, RXC           ;受信完了でスキップ
           RJMP  USART_Rx         ;受信完了待機
;
           IN   R18, UCSRA        ;状態フラグ取得
           IN   R17, UCSRB        ;受信第9ビット取得
           IN   R16, UDR          ;受信データ取得
           ANDI R18, (1<<FE) | (1<<DOR) | (1<<UPE) ;受信異常検査
           BREQ USART_Rx_V       ;異常なしで分岐
;
           LDI  R17, -1           ;異常で-1値設定
           LDI  R16, -1           ;
USART_Rx_V: LSR   R17              ;RXB8ビットをビット0位置へ移動
           ANDI R17, $01         ;RXB8ビットのみ有効
           RET                    ;呼び出し元へ復帰
    
```

C言語プログラム例

```

unsigned int USART_Receive(void)
{
    unsigned char status, resh, resl; /* 一時変数定義 */
    while ( !(UCSRA & (1<<RXC)) ); /* 受信完了待機 */
    status = UCSRA; /* 状態フラグ取得 */
    resh = UCSRB; /* 受信第9ビット取得 */
    resl = UDR; /* 受信データ取得 */
    if ( status & ((1<<FE) | (1<<DOR) | (1<<UPE)) ) return -1; /* 受信異常で-1値設定/復帰 */
    resh = (resh>>1) & 0x01; /* RXB8ビットのみ有効最下位へ */
    return ((resh<<8) | resl); /* 結果9ビットデータ取得/復帰 */
}
    
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

この受信関数例は何か評価を行う前に、全てのI/Oレジスタをレジスタファイルに読みます。これは読まれた緩衝部位置が可能な限り早く新規データを自由に受け入れられるため、最適な受信緩衝部利用になります。

受信完了フラグと割り込み

USART受信部には受信部の状態を示す1つのフラグがあります。

USART受信完了(RXC)フラグは未読データが受信緩衝部に存在するかを示します。このフラグは受信緩衝部に未読データが存在する時に1で、受信緩衝部が空の(換言すると、何も未読データを含まない)時に0です。受信部が禁止される場合(RXEN=0)、受信緩衝部が破棄され、その結果としてRXCフラグは0になります。

USART制御/状態レジスタB(UCSRB)でUSART受信完了割り込み許可(RXCIE)ビットが設定(1)されると、(全割り込みが許可されていれば)RXCフラグが設定(1)されている限り、USART受信完了割り込みが実行されます。割り込み駆動データ受信が使われるとき、受信完了割り込みルーチンはRXCフラグを解除(0)するためにUSARTデータレジスタ(UDR)から受信したデータを読まなければならない、さもなければ一旦割り込みルーチンを終了しても新しい割り込みが(継続的に)起きます。

受信異常フラグ

USART受信には3つの異常フラグ、フレーミング異常(FE)、データオーバーラン発生(DOR)、パリティ誤り(UPE)があります。全てが**USART制御/状態レジスタA(UCSRA)**を読むことによってアクセスできます。異常フラグに対する共通点は異常状態を示しているフレームと共に受信緩衝部に配置されることです。異常フラグが緩衝されるため、**USARTデータレジスタ(UDR)** I/O位置を読むことが緩衝部読み出し位置を切り替えるので、UCSRAは受信緩衝部(UDR)の前に読まなければならない。異常フラグに対するその他の共通点はソフトウェアがこのフラグ位置へ書き込みを行うことによって変更できないことです。しかし、将来のUSART(機能)実装の上位互換性のため、UCSRAが書かれるとき、全てのフラグは**0**に設定されなければならない。異常フラグはどれも割り込みを生成できません。

フレーミング異常(FE)フラグは受信緩衝部に格納された、次に読み込み可能なフレームの第1停止ビットの状態を示します。FEフラグは停止ビットが正しく(Highとして)読まれた時に**0**で、停止ビットが不正(Low)だった時にFEフラグは**1**です。このビットは同期外れ状態の検出、中断状態検出、規約での操作に使えます。受信部が最初(第1)以外の全停止ビットを無視するため、FEフラグは**USART制御/状態レジスタC(UCSRC)の停止ビット選択(USBS)ビット**設定によって影響を及ぼされません。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットを**0**に設定してください。

データオーバーラン発生(DOR)フラグは受信部緩衝部が一杯状態のためのデータ消失を示します。データオーバーランは受信緩衝部(2フレーム)が一杯で、(次の)新規フレームデータが受信移動レジスタで待っており、(更に次の)新規開始ビットが検出される時に起きます。DORフラグが設定(**1**)なら、最後にUDRから読んだフレームと次にUDRから読むフレーム間で1つ以上の直列フレームが失われています。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットに**0**を書いてください。DORフラグは受信されたデータが移動レジスタから受信緩衝部へ正常に移動された時に解除(**0**)されます。

パリティ誤り(UPE)フラグは受信緩衝部内の次フレームで、受信時にパリティ異常があったことを示します。パリティ検査が許可されていない場合、UPEフラグは常に**0**が読めます。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットを**0**に設定してください。より多くの詳細については**79頁の「パリティビットの計算」と次の「パリティ検査器」**をご覧ください。

パリティ検査器

パリティ検査器はパリティ種別上位ビット(**UPM1**)が設定(**1**)されると活性(有効)になります。実行されるべきパリティ検査の形式(偶数または奇数)は**UPM0**ビットによって選ばれます。許可されると、パリティ検査器は到着フレーム内のデータビットのパリティを計算し、その結果と(受信)直列フレーム内のパリティビットを比較します。検査の結果は受信データ、停止ビットと共に受信緩衝部に格納されます。その後パリティ誤り(UPE)フラグはフレームにパリティ異常があるかを検査するため、ソフトウェアによって読むことができます。

UPEフラグは受信緩衝部から読まれ得る次フレームで受信時にパリティ異常があり、その時点でパリティ検査が許可されていた(**UPM1=1**)場合に設定(**1**)されます。このビットはUSARTデータレジスタ(UDR)が読まれるまで有効です。

受信の禁止

送信部と対照的に受信部の禁止は即時です。従って受信進行中のデータは失われます。禁止されると(換言すると、**USART制御/状態レジスタB(UCSRB)のUSART受信許可(RXEN)ビット**が**0**に設定)、受信部はもはやRXDポートピンの標準機能を無効にしません。受信FIFO緩衝部は受信部が禁止されると破棄されます。緩衝部内の残データは失われます。

受信緩衝部の破棄

受信FIFO緩衝部は受信が禁止されると破棄(換言すると、緩衝部は内容を空に)されます。未読データは失われます。例えば異常状態のため、通常動作中に緩衝部が破棄されなければならない場合、**USART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグ**が解除(**0**)されるまでUSARTデータレジスタ(UDR) I/O位置を読んでください。次のコード例は受信緩衝部の破棄方法を示します。

アセンブリ言語プログラム例

```
USART_Flush: SBIS    UCSRA, RXC          ;未読データありでスキップ
              RET                          ;未読データなしで復帰
;
              IN     R16, UDR             ;データ受信
              RJMP   USART_Flush         ;未読データなしまで継続
```

C言語プログラム例

```
void USART_Flush(void)
{
    unsigned char dummy;                /* 一時変数定義 */
    while ( UCSRA & (1<<RXC) ) dummy=UDR; /* 未読データ読み捨て */
}
```

注: このコード例はデバイス定義ファイルがインクルードされることが前提です。

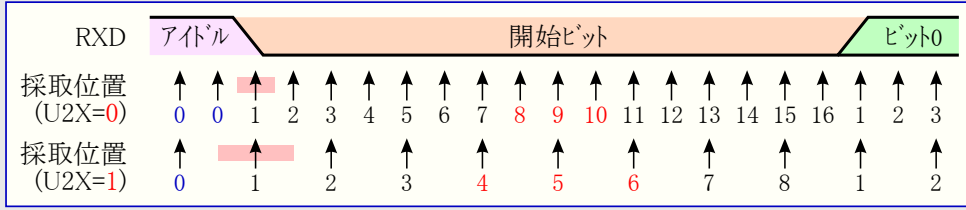
非同期受信

USARTは非同期データ受信を扱うためのクロック再生とデータ再生部を含みます。クロック再生論理回路は内部的に生成したボーレートクロックをRXDピンに到着する非同期直列フレームに同期化するために使われます。データ再生論理回路は到着ビット毎に低域通過の濾波と採取をし、それによって受信部の雑音耐性を改善します。非同期受信動作範囲は内部ボーレートクロック精度、到着フレーム速度、フレーム長のビット数に依存します。

非同期クロック再生

クロック再生論理回路は内部クロックを到着直列フレームに同期化します。図57.は到着フレームの開始ビット採取手順を図解します。採取速度は標準動作でボーレートの16倍、倍速動作で8倍です。赤帯(訳注:原文は水平矢印)は採取処理のための同期変量を図示します。倍速動作(U2X=1)を使う時の広い変量時間に注意してください。採取番号0はRXD信号がアイドル(換言すると、通信の動きなし)の時に行われる採取です。

図57. 開始ビットの採取

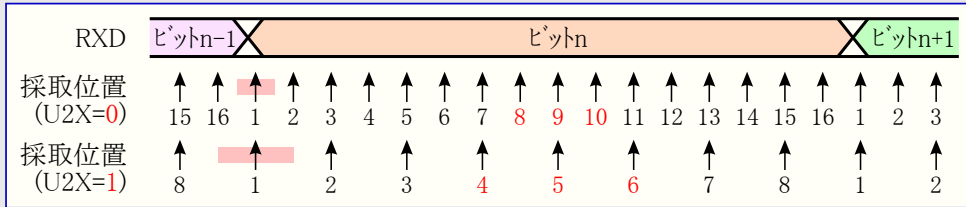


クロック再生論理回路がRXD信号線でHigh(アイドル)からLow(開始)の遷移を検出すると、開始ビット検出手順が開始されます。図で示されるように採取1は最初のLow採取を意味します。その後クロック再生回路は有効な開始ビットが受信されるかを定めるために、標準動作に対して採取8,9,10、倍速動作に対して採取4,5,6(図の赤字(訳注:原文は箱枠内))で示された採取番号)を使います。これら3回で2回以上の採取が論理Highレベルの場合(多数決)、この開始ビットは尖頭雑音として捨てられ、受信部は次のHighからLowの遷移を探し始めます。けれども有効な開始ビットが検出されると、クロック再生論理回路は同期化され、データ再生が開始されます。この同期化手順は各開始ビット毎に繰り返されます。

非同期データ再生

受信部クロックが開始ビットに同期化されるとデータ再生が始まります。データ再生部は標準動作で16段、倍速動作で8段の順列回路を使います。図58.はデータビットとパリティビットの採取を示します。各採取は再生部の各段に等しい番号を与えられます。

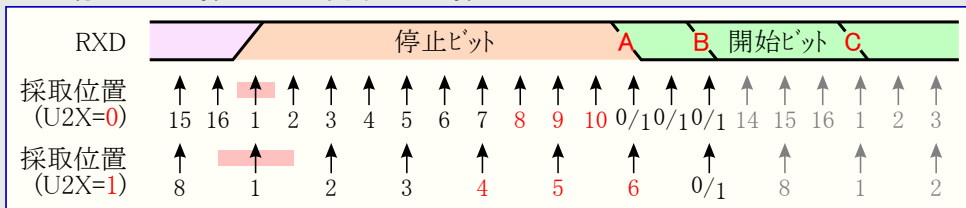
図58. データビットとパリティビットの採取



受信したビットの論理値の決定は受信したビット中央の3採取で論理値の多数決を取るによって行われます。この中央の3採取は図上の赤字(訳注:原文は箱枠内)の採取番号によって強調されます。多数決の手順は次のように行われます。2または全3採取がHighレベルならば受信したビットは論理1が記録されます。2または全3採取がLowレベルならば受信したビットは論理0が記録されます。この多数決手順はRXDピンの到着信号に対して低域通過濾波器(ローパスフィルタ)のように働きます。この再生手順はその後に完全なフレームが受信されるまで繰り返されます。これには最初の(第1)停止ビットを含みます。受信部がフレームの最初の停止ビットだけを使うことに注意してください。

図59.は停止ビットの採取と次フレームの最も早い開始ビット開始の可能性を示します。

図59. 停止ビットの採取と次の開始ビットの採取



フレーム内の他のビットに対して行われるのと同じ多数決が停止ビットにも行われます。停止ビットが論理0値と記録されると、フレーミング異常(FE)フラグが設定(1)されます。

新規フレームの開始ビットを示すHighからLowへの遷移は多数決に使った最後のビット後に実現できます。標準動作での最初のLowレベル採取は図59.のA点で有り得ます。倍速動作での最初のLowレベルはB点に遅れます。C点は完全な長さの停止ビット(の終点)を示します。この早い開始ビット検出は受信部の動作範囲に影響します。

非同期での動作範囲

受信部の動作範囲は受信したビット速度と内部的に生成したボーレート間の不一致に依存します。送信部が速すぎるまたは遅すぎるビット速度でフレームを送出したり、内部的に発生した受信部のボーレートが類似した(表48.参照)基準周波数を持たない場合、受信部は開始ビットでフレームを同期できません。

次式は到着データ速度と内部受信部ボーレート間の比率計算に使えます。

$$R_{slow} = \frac{(D+1) \times S}{S-1+D \times S+S_F} \quad R_{fast} = \frac{(D+2) \times S}{(D+1) \times S+S_M}$$

D : データとパリティのビット数 (5~10)
 S : ビットあたりの採取数 (標準速=16、倍速=8)
 S_F : 多数決に使う最初の採取番号 (標準速=8、倍速=4)
 S_M : 多数決に使う中心の採取番号 (標準速=9、倍速=5)
 R_{slow} : は受信側ボーレートに対して許容できる最低受信ビット速度の比率です。
 R_{fast} : は受信側ボーレートに対して許容できる最高受信ビット速度の比率です。

表48.は許容できる最大受信部ボーレート誤差一覧です。標準速動作には、より高いボーレート変動許容力があることに注目してください。

表48. 標準速と倍速での受信部ボーレート推奨最大許容誤差

D	標準速動作 (U2X=0)				倍速動作 (U2X=1)			
	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)
5	93.20	106.67	-6.80~+6.67	±3.0	94.12	105.66	-5.88~+5.66	±2.5
6	94.12	105.79	-5.88~+5.79	±2.5	94.92	104.92	-5.08~+4.92	±2.0
7	94.81	105.11	-5.19~+5.11	±2.0	95.52	104.35	-4.48~+4.35	±1.5
8	95.36	104.58	-4.54~+4.58	±2.0	96.00	103.90	-4.00~+3.90	±1.5
9	95.81	104.14	-4.19~+4.14	±1.5	96.39	103.53	-3.61~+3.53	±1.5
10	96.17	103.78	-3.83~+3.78	±1.5	96.70	103.23	-3.30~+3.23	±1.0

注: Dはデータビット数とパリティビットの合計ビット数です。

(訳注) 原書は表48.に標準速、表49.に倍速を記載していますが、比較が容易なように表48.として纏めました。

受信部ボーレートの推奨最大許容誤差は最大総合許容誤差を送信部と受信部で等分割するという仮定の下で作られました。

受信部ボーレート誤差に対して2つの起こり得る原因があります。受信部のシステムクロック(XTAL)は供給電圧範囲と温度範囲に関して常に若干の不安定性があります。システムクロックを生成するのに水晶発振子を使う時は殆ど問題ありませんが、(セラミック)振動子でのシステムクロックは振動子偏差に依存して2%を越えて異なるかもしれません。2つ目の誤り原因はより制御可能です。ボーレート発振器は欲しいボーレートを得るためにシステム周波数の正確な分周を常に行うことはできません。この場合、可能ならば受け入れ可能な低い誤差を与えるUBRR値が使えます。

複数プロセッサ通信動作

USART制御/状態レジスタA(UCSRA)での**複数プロセッサ通信動作(MPCM)ビット**の設定(1)はUSART受信部によって受信された到着フレームの選別機能を許可します。アドレス情報を含まないフレームは無視され、受信緩衝部に格納されません。これは同一直列バス経由で通信する複数MCUのシステムで、CPUによって扱われなければならない到着フレーム数を効果的に減らします。送信部はMPCMビット設定によって影響されませんが、複数プロセッサ通信動作を利用するシステムの一部の時は違うふうに使われなければなりません。

受信部が5~8データビットを含むフレームを受信するように設定されるなら、最初の停止ビットはデータまたはアドレス情報を含むフレームかどうかを示します。受信部が9データビットのフレームに設定されるなら、**USART制御/状態レジスタB(UCSRB)の受信第9(RXB8)ビット**がアドレスとデータのフレームを識別するのに使われます。フレーム種別(最初の停止または第9)ビットが1の時にフレームはアドレスを含みます。フレーム種別ビットが0の時にそのフレームはデータフレームです。

複数プロセッサ通信動作は主MCUからのデータを多くの従MCUで受信することを可能にします。これはどのMCUがアドレス指定されるかを検出するため、最初にアドレスフレームを調べることによって行われます。特定の従MCUがアドレス指定されたなら、そのMCUは後続するデータフレームを通常のように受信し、一方その他の従MCUは他のアドレスフレームが受信されるまで受信したフレームを無視します。

複数プロセッサ通信の使用法

主MCUとして動作するMCUは9ビットデータフレーム形式(**UCSZ=7**)を使えます。UCSRBの**送信第9(TXB8)ビット**はアドレスフレーム時に設定(1)、またはデータフレーム時に解除(0)されて送信されなければなりません。この場合、従MCUは9ビットデータフレーム形式の使用に設定されなければなりません。

複数プロセッサ通信動作でデータを交換するのに次の手順が使われるべきです。

1. 全ての従MCUは複数プロセッサ通信動作です(UCSRAの**複数プロセッサ通信動作(MPCM)ビット**が設定(1))。
2. 主MCUはアドレスフレームを送り、全ての従装置がこのフレームを受信し、これを読みます。従CPUでは通常のようにUCSRAで**受信完了(RXC)フラグ**が設定(1)されます。
3. 各従MCUは**USARTデータレジスタ(UDR)**を読み、選ばれたかを判定します。選ばれた場合はUCSRAのMPCMビットを解除(0)し、そうでなければ(非選択の場合は)MPCMビット設定を保ち、次のアドレスフレームを待ちます。
4. アドレス指定されたMCUは新規アドレスフレームが受信されるまで全データフレームを受信します。MPCMビットが未だ設定(1)されている他の従CPUはこのデータフレームを無視します。
5. 最後のデータフレームがアドレス指定されたMCUによって受信されると、アドレス指定されたMCUはMPCMビットを設定(1)し、主装置からの新規アドレスフレームを待ちます。以降、手順は2.からを繰り返します。

5~8ビットデータフレーム形式のどの使用も可能ですが、受信側が使う n と $n+1$ ビットデータフレーム形式間を切り替えなければならないため非実用的です。これは送信部と受信部が同じデータビット長設定を使うため、全二重(フルデュプレックス)動作を困難にします。5~8ビットデータフレームが使われる場合、最初の停止ビットがフレーム種別を示すのに使われるので、送信部は2停止ビット使用(**USBS=1**)に設定されなければなりません。

MPCMビットを設定(1)または解除(0)するのに読み-修正-書き(リード モデファイライト)命令(**SBI**と**CBI**)を使ってはいけません。MPCMビットは**送信完了(TXC)フラグ**と同じI/O位置を共用しており、**SBI**または**CBI**命令を使うと偶然に解除(0)されるかもしれません。

USART用レジスタ

USARTデータレジスタ (USART I/O Data Register) UDR

ビット	7	6	5	4	3	2	1	0	
\$0C (\$2C)	R/TXB7	R/TXB6	R/TXB5	R/TXB4	R/TXB3	R/TXB2	R/TXB1	R/TXB0	UDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

同じI/Oアドレスを共用するUSART受信データ緩衝レジスタとUSART送信データ緩衝レジスタはUSARTデータレジスタまたはUDRとして引用しました。送信データ緩衝レジスタ(TXB)はUDRレジスタ位置に書かれるデータの転送先です。UDRレジスタ位置読み込みは受信データ緩衝レジスタ(RXB)の内容を返します。

5〜7ビットデータでの上位未使用ビットは送信部によって無視され、受信部によって0に設定されます。

送信緩衝部はUCSRAで送信データレジスタ空き(UDRE)フラグが設定(1)される時にだけ書けます。UDREフラグが設定(1)されない時にUDRへ書かれたデータはUSART送信部によって無視されます。送信部が許可されて送信緩衝部にデータが書かれると、送信部は送信移動レジスタが空きの時にそのデータを送信移動レジスタへ設定(移動)します。その後データはTXDピンで直列送信されます。

受信緩衝部は2段のFIFOから成ります。このFIFOは受信緩衝部がアクセスされる毎にその状態を切り替えます。この受信緩衝部の動きのため、この位置に読み-修正-書き(リード モデファイライト)命令(SBIとCBI)を使えません。ビット検査(SBICとSBIS)命令もFIFOの状態を換えるので、これらの命令を使う時は注意してください。

USART制御/状態レジスタA (USART Control and Status Register A) UCSRA

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
初期値	0	0	1	0	0	0	0	0	

• ビット7 - RXC : USART受信完了フラグ (USART Receive Complete)

このフラグは受信緩衝部に未読データがある時に設定(1)され、受信緩衝部が空の(換言すると、どんな未読データも含まない)時に解除(0)されます。受信部が禁止されると、受信緩衝部が破棄され、その結果RXCフラグは0になります。RXCフラグは受信完了割り込みを発生するのに使えます(USART制御/状態レジスタB(UCSRB)の受信完了割り込み許可(RXCIE)ビットをご覧ください)。

• ビット6 - TXC : USART送信完了フラグ (USART Transmit Complete)

このフラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部(UDR)に新規データが現状存在しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行される時、自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)できます。TXCフラグは送信完了割り込みを発生できます(UCSRBの送信完了割り込み許可(TXCIE)ビットをご覧ください)。

• ビット5 - UDRE : USART送信データレジスタ空きフラグ (USART Data Register Empty)

UDREフラグは送信緩衝部(UDR)が新規データを受け取る準備ができているかどうかを示します。UDREが1ならば緩衝部は空で、従って書かれる準備ができています。UDREフラグは送信緩衝部空き割り込みを発生できます(UCSRBの送信データレジスタ空き割り込み許可(UDRIE)ビットをご覧ください)。送信部が準備できているのを示すため、リセット後のUDREは設定(1)です。

• ビット4 - FE : フレーミング異常フラグ (Framing Error)

受信緩衝部の次データが受信した時にフレーミング異常(換言すると、受信緩衝部で次データの最初の(第1)停止ビットがLow)だった場合、このビットが設定(1)されます。このフラグは受信緩衝部(UDR)が読まれるまで有効です。受信したデータの停止ビットがHighの時にFEフラグは0です。UCSRAに書くとき、常にこのビットを0に設定してください。

• ビット3 - DOR : データオーバーラン発生フラグ (Data OverRun)

このビットはオーバーラン状態が検出されると設定(1)されます。受信緩衝部(2フレーム分)が一杯で、新規フレームが受信移動レジスタ内で待機中に新規開始ビットが検出されるとデータオーバーランが起こります。UCSRAに書くとき、常にこのビットを0に設定してください。

• ビット2 - UPE : パリティ誤りフラグ (USART Parity Error)

受信緩衝部の次データが受信した時にパリティ異常があり、その時点でパリティ検査が許可されていれば(UPM1=1)、このビットが設定(1)されます。このフラグは受信緩衝部(UDR)が読まれるまで有効です。UCSRAに書くとき、常にこのビットを0に設定してください。

• ビット1 - U2X : 倍速許可 (Double the USART Transmission Speed)

このビットは非同期動作でだけ有効です。同期動作を使うとき、このビットに0を書いてください。

このビットに1を書くことはボーレート分周器の分周値を16から8に減らして事実上、非同期通信の転送速度を倍にします。

• ビット0 - MPCM : 複数プロセッサ通信動作 (Multi-Processor Communication Mode)

このビットは複数プロセッサ通信動作を許可します。MPCMビットが1を書かれると、USART受信部で受信したアドレス情報を含まない全到着フレームは無視されます。送信部はMPCM設定に影響されません。より多くの詳細情報については86頁の「複数プロセッサ通信動作」をご覧ください。

USART制御/状態レジスタB (USART Control and Status Register B) UCSRB

ビット	7	6	5	4	3	2	1	0	
\$0A (\$2A)	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – RXCIE : 受信完了割り込み許可 (Receive Complete Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグでの割り込みを許可します。USART受信完了割り込みはRXCIEビットが1を書かれ、ステータスレジスタ(SREG)で全割り込み許可(I)ビットが1を書かれ、UCSRAでRXCフラグが設定(1)される場合にだけ生成されます。

•ビット6 – TXCIE : 送信完了割り込み許可 (Transmit Complete Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の送信完了(TXC)フラグでの割り込みを許可します。USART送信完了割り込みはTXCIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでTXCフラグが設定(1)される場合にだけ生成されます。

•ビット5 – UDRIE : 送信データレジスタ空き割り込み許可 (Transmit Data Register Empty Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグでの割り込みを許可します。USART送信データレジスタ空き割り込みはUDRIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでUDREフラグが設定(1)される場合にだけ生成されます。

•ビット4 – RXEN : 受信許可 (Receiver Enable)

このビットに1を書くことはUSART受信(部)を許可します。受信部は許可されるとRXDピンの標準ポート動作を無効にします。受信(部)を禁止することは受信緩衝部を破棄し、フレーミング異常(FE)、オーバラン(DOR)、パリティ誤り(UPE)のフラグを無効にします。

•ビット3 – TXEN : 送信許可 (Transmitter Enable)

このビットに1を書くことはUSART送信(部)を許可します。送信部は許可されるとTXDピンの標準ポート動作を無効にします。送信(部)の禁止(TXEN=0書き込み)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない)まで有効になりません。禁止したとき、送信部はもはやTXDポート(の標準I/O機能)を無効にしません。

•ビット2 – UCSZ2 : データビット長選択2 (Character Size)

USART制御/状態レジスタC(UCSRC)のUCSZ1,0ビットと組み合わせたUCSZ2ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

•ビット1 – RXB8 : 受信データビット8 (Receive Data Bit 8)

RXB8は9ビットデータでの直列フレーム操作時に受信したフレームの第9データビット(ビット8)です。UDRから下位ビットを読む前に読んでください。

•ビット0 – TXB8 : 送信データビット8 (Transmit Data Bit 8)

TXB8は9ビットデータでの直列フレーム操作時に送信されるべきデータの第9データビット(ビット8)です。UDRへ下位ビットを書く前に書いてください。

USART制御/状態レジスタC (USART Control and Status Register C) UCSRC

ビット	7	6	5	4	3	2	1	0	
\$03 (\$23)	–	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	1	1	0	

•ビット7 – Res : 予約 (Reserved) (訳注: 共通性のため本項追加)

このビットは将来使用するために予約されています。将来のデバイスとの共通性を保つため、UCSRCが書かれるとき、このビットは0が書かれなければなりません。

•ビット6 – UMSEL : USART動作選択 (USART Mode Select)

このビットは非同期と同期の動作種別のどちらかを選びます。

表50. USART動作選択

UMSEL	動作種別
0	非同期動作
1	同期動作

•ビット5,4 – UPM1,0 : パリティ選択 (Parity Mode)

これらのビットはパリティの発生と検査の許可と種別を設定します。許可した場合、送信部は各フレーム内での送信したデータビットのパリティを自動的に生成して送出します。受信部は到着データからパリティ値を生成し、UPM0設定と比較します。不一致が検出されると、USART制御/状態レジスタA(UCSRA)でパリティ誤り(UPE)フラグが設定(1)されます。

表51. パリティ選択

UPM1	UPM0	パリティ動作
0	0	禁止
0	1	(予約)
1	0	偶数パリティ許可
1	1	奇数パリティ許可

•ビット3 – USBS : 停止ビット選択 (Stop Bit Select)

このビットは送信部によって挿入される停止ビット数を選びます。受信部はこの設定を無視します(**訳補**:常に第1停止ビットだけが有効)。

表52. 停止ビット選択

USBS	停止ビット数
0	1ビット
1	2ビット

•ビット2,1 – UCSZ1,0 : データビット長選択 (Character Size)

USART制御/状態レジスタB(UCSRB)のUCSZ2ビットと組み合わせたUCSZ1,0ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

表53. データビット長選択

UCSZ2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
データビット数	5ビット	6ビット	7ビット	8ビット	(予約)	(予約)	(予約)	9ビット

•ビット0 – UCPOL : クロック極性選択 (Clock Polarity)

このビットは同期動作に対してだけ使われます。非同期動作が使われるとき、このビットに0を書いてください。UCPOLビットは同期クロック(XCK)、データ出力変更、データ入力採取間の関係を設定します。

表54. XCKクロック極性選択

UCPOL	送信データ変更 (TXDピン出力)	受信データ採取 (RXDピン入力)
0	XCKの上昇端	XCKの下降端
1	XCKの下降端	XCKの上昇端

USARTボーレートレジスタ (USART Baud Rate Register) UBRRH, UBRL (UBRR)

ビット	15	14	13	12	11	10	9	8	
\$02 (\$22)	–	–	–	–	UBRR11	UBRR10	UBRR9	UBRR8	UBRRH
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$09 (\$29)	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0	UBRRL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット15~12 – Res : 予約 (Reserved Bits)

これらのビットは将来使用するために予約されています。将来のデバイスとの共通性のため、UBRRHが書かれるとき、これらのビットは0が書かれなければなりません。

•ビット11~0 – UBRR11~0 : ボーレート分周値 (USART Baud Rate Register)

これはUSARTのボーレートを含む12ビットレジスタです。UBRRHがUSARTボーレートの上位4ビットを含み、UBRRLが下位8ビットを含みます。ボーレートが変更されると、送受信部で進行中の転送は不正にされます。UBRRL書き込みはボーレート前置分周器の更新を直ちに始めます。

ボーレート設定例

標準的な水晶発振子やセラミック振動子の周波数に対して最も一般的に使われる非同期動作のボーレートは表55のUBRR設定を使うことによって生成できます。目的のボーレートに対して設定誤差1.5%(標準速:U2X=0)、1.0%(倍速:U2X=1)以上を赤字で示します(訳注:原書の本文は0.5%未満を太字)。より高い誤差率でも受け入れ可能ですが、特に長い直列フレーム(85頁の「非同期での動作範囲」参照)で誤差率が高いと、受信部は雑音耐性が低下します。誤差率は次式を使って計算されます。

$$\text{誤差率(\%)} = \left(\frac{\text{UBRR設定ボーレート(最近似値)}}{\text{目的のボーレート}} - 1 \right) \times 100(\%)$$

表55. Xtal、ボーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ボーレート (bps)	1MHz				1.8432MHz				2MHz				2.4576MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	51	0.2	103	0.2	95	0.0	191	0.0	103	0.2	207	0.2	127	0.0	255	0.0
2400	25	0.2	51	0.2	47	0.0	95	0.0	51	0.2	103	0.2	63	0.0	127	0.0
4800	12	0.2	25	0.2	23	0.0	47	0.0	25	0.2	51	0.2	31	0.0	63	0.0
9600	6	-7.0	12	0.2	11	0.0	23	0.0	12	0.2	25	0.2	15	0.0	31	0.0
14400	3	8.5	8	-3.5	7	0.0	15	0.0	8	-3.5	16	2.1	10	-3.0	20	1.6
19200	2	8.5	6	-7.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	0.0	15	0.0
28800	1	8.5	3	8.5	3	0.0	7	0.0	3	8.5	8	-3.5	4	6.7	10	-3.0
38400	1	-18.6	2	8.5	2	0.0	5	0.0	2	8.5	6	-7.0	3	0.0	7	0.0
57600	0	8.5	1	8.5	1	0.0	3	0.0	1	8.5	3	8.5	2	-11.1	4	6.7
76800	0	-18.6	1	-18.6	1	-25.0	2	0.0	1	-18.6	2	8.5	1	0.0	3	0.0
115.2k	-	-	0	8.5	0	0.0	1	0.0	0	8.5	1	8.5	0	33.3	2	-11.1
230.4k	-	-	0	-45.7	0	-50.0	0	0.0	0	-45.7	0	8.5	0	-33.3	0	33.3
250k	-	-	-	-	-	-	0	-7.8	-	-	0	0.0	-	-	0	22.9
最高速	62500		125k		115.2k		230.4k		125k		250k		153.6k		307.2k	

ボーレート (bps)	3.2768MHz				3.6864MHz				4MHz				4.608MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	170	-0.2	340	0.1	191	0.0	384	0.0	207	0.2	416	-0.1	239	0.0	479	0.0
2400	84	0.4	170	-0.2	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0
4800	42	-0.8	84	0.4	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0
9600	20	1.6	42	-0.8	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0
14400	13	1.6	27	1.6	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0
19200	10	-3.0	20	1.6	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0
28800	6	1.6	13	1.6	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0
38400	4	6.7	10	-3.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	-6.3	14	0.0
57600	3	-11.1	6	1.6	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0
76800	2	-11.1	4	6.7	2	0.0	5	0.0	2	8.5	6	-7.0	3	-6.3	7	-6.3
115.2k	1	-11.1	3	-11.1	1	0.0	3	0.0	1	8.5	3	8.5	2	-16.7	4	0.0
230.4k	0	-11.1	1	-11.1	0	0.0	1	0.0	0	8.5	1	8.5	0	25.0	2	-16.7
250k	-	-	1	-18.1	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2
500k	-	-	0	-18.1	-	-	0	-7.8	-	-	0	0.0	0	-42.4	0	15.2
1M	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	-42.4
最高速	204.8k		409.6k		230.4k		460.8k		250k		500k		288k		576k	

注: 最高速はUBRR=0、誤差=0.0%です。以降の周波数は次頁へ続く。

(訳注) 原書では頁割された表毎に表番号55.~58.となっていますが、共通性から纏めて表55.としました。原書に対して数種の発振周波数を追加しました。

表55 (続き). Xtal、ホーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ホーレート (bps)	4.9152MHz				6.144MHz				7.3728MHz				8MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	255	0.0	511	0.0	319	0.0	639	0.0	383	0.0	767	0.0	416	-0.1	832	0.0
2400	127	0.0	255	0.0	159	0.0	319	0.0	191	0.0	383	0.0	207	0.2	416	-0.1
4800	63	0.0	127	0.0	79	0.0	159	0.0	95	0.0	191	0.0	103	0.2	207	0.2
9600	31	0.0	63	0.0	39	0.0	79	0.0	47	0.0	95	0.0	51	0.2	103	0.2
14400	20	1.6	42	-0.8	26	-1.2	52	0.6	31	0.0	63	0.0	34	-0.8	68	0.6
19200	15	0.0	31	0.0	19	0.0	39	0.0	23	0.0	47	0.0	25	0.2	51	0.2
28800	10	-3.0	20	1.6	12	2.6	26	-1.2	15	0.0	31	0.0	16	2.1	34	-0.8
38400	7	0.0	15	0.0	9	0.0	19	0.0	11	0.0	23	0.0	12	0.2	25	0.2
57600	4	6.7	10	-3.0	6	-4.8	12	2.6	7	0.0	15	0.0	8	-3.5	16	2.1
76800	3	0.0	7	0.0	4	0.0	9	0.0	5	0.0	11	0.0	6	-7.0	12	0.2
115.2k	2	-11.1	4	6.7	2	11.1	6	-4.8	3	0.0	7	0.0	3	8.5	8	-3.5
230.4k	1	33.3	2	-11.1	1	-16.7	2	11.1	1	0.0	3	0.0	1	8.5	3	8.5
250k	1	22.9	2	-18.1	1	-23.2	2	2.4	1	-7.8	3	0.0	1	0.0	3	0.0
500k	0	-38.6	0	22.9	0	-23.2	1	-23.2	0	-7.8	1	-7.8	0	0.0	1	0.0
1M	-	-	0	-38.6	-	-	0	-23.2	-	-	0	-7.8	-	-	0	0.0
最高速	307.2k		614.4k		384k		768k		460.8k		921.6k		500k		1M	

ホーレート (bps)	9.216MHz				9.8304MHz				10MHz				11.0592MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	479	0.0	959	0.0	511	0.0	1023	0.0	520	-0.0	1041	-0.0	575	0.0	1151	0.0
2400	239	0.0	479	0.0	255	0.0	511	0.0	259	0.2	520	-0.0	287	0.0	575	0.0
4800	119	0.0	239	0.0	127	0.0	255	0.0	129	0.2	259	0.2	143	0.0	287	0.0
9600	59	0.0	119	0.0	63	0.0	127	0.0	64	0.2	129	0.2	71	0.0	143	0.0
14400	39	0.0	79	0.0	42	-0.8	84	0.4	42	0.9	86	-0.2	47	0.0	95	0.0
19200	29	0.0	59	0.0	31	0.0	63	0.0	32	-1.4	64	0.2	35	0.0	71	0.0
28800	19	0.0	39	0.0	20	1.6	42	-0.8	21	-1.4	42	0.9	23	0.0	47	0.0
38400	14	0.0	29	0.0	15	0.0	31	0.0	15	1.8	32	-1.4	17	0.0	35	0.0
57600	9	0.0	19	0.0	10	-3.0	20	1.6	10	-1.4	21	-1.4	11	0.0	23	0.0
76800	7	6.7	14	0.0	7	0.0	15	0.0	7	1.8	15	1.8	8	0.0	17	0.0
115.2k	4	0.0	9	0.0	4	6.7	10	-3.0	4	8.5	10	-1.4	5	0.0	11	0.0
230.4k	2	-16.7	4	0.0	2	-11.1	4	6.7	2	-9.6	4	8.5	2	0.0	5	0.0
250k	1	15.2	4	-7.8	1	-23.2	4	-1.7	2	-16.7	4	0.0	2	-7.8	5	-7.8
500k	0	15.2	1	15.2	0	22.9	1	22.9	0	25.0	2	-16.7	0	38.2	2	-7.8
1M	0	-42.4	0	15.2	0	-38.6	0	22.9	0	-37.5	0	25.0	0	-30.9	0	38.2
最高速	576k		1.152M		614.4k		1.2288M		625k		1.25M		691.2k		1.3824M	

ホーレート (bps)	14.7456MHz				16MHz				18.4320MHz				20MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	767	0.0	1533	0.0	832	0.0	1666	0.0	959	0.0	1919	0.0	1041	0.0	2082	0.0
2400	383	0.0	767	0.0	416	-0.1	832	0.0	479	0.0	959	0.0	520	0.0	1041	0.0
4800	191	0.0	383	0.0	207	0.2	416	-0.1	239	0.0	479	0.0	259	0.2	520	0.0
9600	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0	129	0.2	259	0.2
14400	63	0.0	127	0.0	68	0.6	138	-0.1	79	0.0	159	0.0	86	-0.2	173	-0.2
19200	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0	64	0.2	129	0.2
28800	31	0.0	63	0.0	34	-0.8	68	0.6	39	0.0	79	0.0	42	0.9	86	-0.2
38400	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0	32	-1.4	64	0.2
57600	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0	21	-1.4	42	0.9
76800	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0	15	1.7	32	-1.4
115.2k	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0	10	-1.4	21	-1.4
230.4k	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0	4	8.5	10	-1.4
250k	3	-7.8	6	5.3	3	0.0	7	0.0	4	-7.8	8	2.4	4	0.0	9	0.0
500k	1	-7.8	3	-7.8	1	0.0	3	0.0	1	15.2	4	-7.8	1	25.0	4	0.0
1M	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2	0	25.0	1	25.0
最高速	921.6k		1.8432M		1M		2M		1.152M		2.304M		1.25M		2.5M	

注: 最高速はUBRR=0、誤差=0.0%です。



多用途直列インターフェース (USI: Universal Serial Interface)

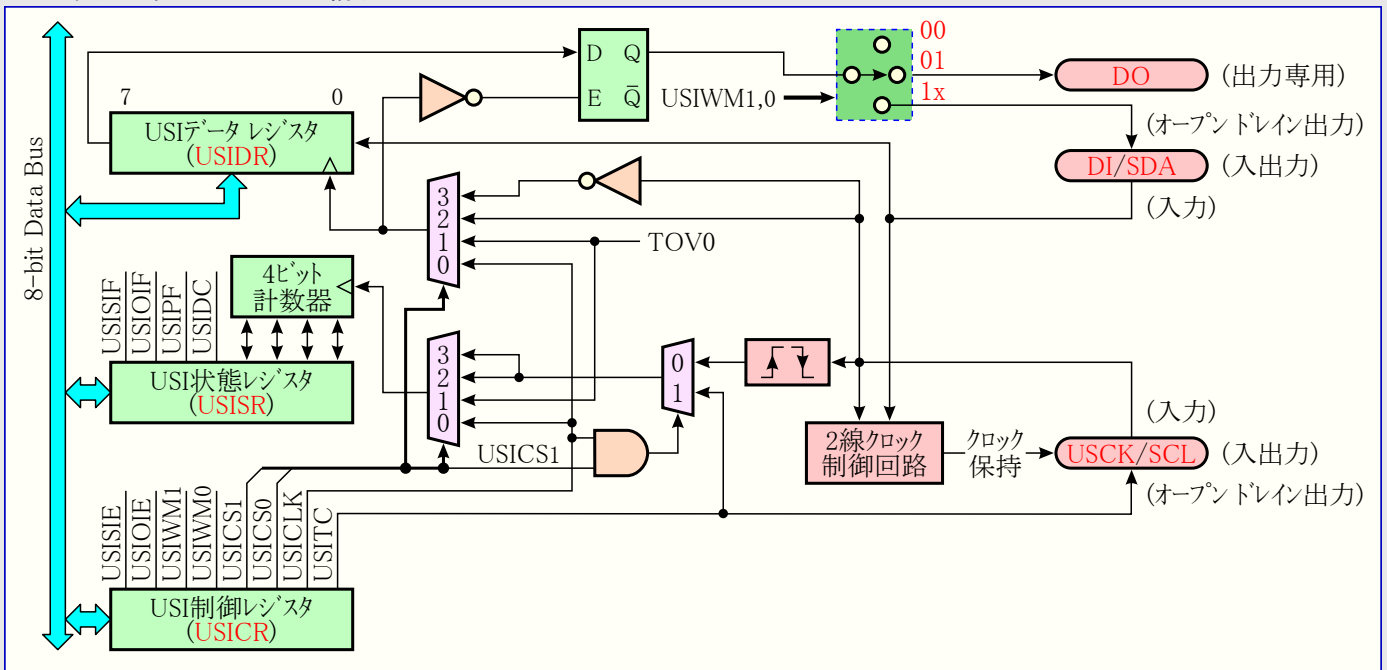
多用途直列インターフェース(USI)は直列通信で必要とされる基本的なハードウェア資源(機能)を提供します。最低限の制御ソフトウェアとの組み合わせで、USIはソフトウェアのみを元とした方法よりも少ないコード量と重要な高い転送レートを許します。プロセッサ負荷を最小とするために割り込みが含まれます。USIの主な特徴は次のとおりです。

- 2線同期データ転送 (主装置または従装置: $f_{SCLmax}=f_{CK}/16$)
- 3線同期データ転送 (主装置: $f_{SCKmax}=f_{CK}/2$ 、従装置: $f_{SCKmax}=f_{CK}/4$)
- データ受信割り込み
- アイドル動作からの起動復帰
- 2線動作でのパワーダウン動作を含む全休止形態からの起動復帰
- 割り込み能力を持つ2線動作開始条件検出器

概要

USIの簡便化した構成図は図60.で示されます。実際のI/Oピン配置は2頁の「ピン配置」を参照してください。I/OビットとI/Oピンを含みCPUがアクセス可能なI/Oレジスタは赤文字(訳注:原文は太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は97頁の「USI用レジスタ」で一覧されます。

図60. 多用途直列インターフェース構成図



8ビットの移動レジスタ(USIDR)はデータバス経由で直接的にアクセス可能で、入って来るデータと出て行くデータを含みます。このレジスタは緩衝部を持たないので、データが失われないことを保証するためにデータは可能な限り早く読まなければなりません。最上位ビットは動作種別設定に依存して2つの出力ピンの1つに接続されます。トランスペアレント(透過型)ラッチは直列(移動)レジスタ出力と出力ピン間に挿入され、データの変更をデータ入力採取と逆のクロック端に遅らせます。直列入力(種別)設定に拘らず、常にデータ入力(DI)ピンから採取されます。

4ビット計数器はデータバス経由で読み書き両方ができ、溢れ割り込みを生成できます。この計数器と直列(移動)レジスタは同じクロック元によって同時にクロック駆動されます。これは受信または送信したビット数を数え、転送完了時に割り込みの生成を計数器に許します。外部クロック元が選ばれると、この計数器が両クロック端で計数することに注意してください。この場合、計数器はビット数ではなく、端(エッジ)数を数えます。クロックは3つの異なるクロック元(USCKピン、タイマ/カウンタ0溢れ、ソフトウェア)から選べます。

2線クロック制御部(回路)は2線バスで開始条件が検出される時に割り込みを発生できます。開始条件検出後、または計数器溢れ後、クロックピンをLowに保持することによって待ち状態を生成することもできます。

機能説明

3線動作

USIの3線動作は直列周辺インターフェース(SPI)の動作種別0と1に準拠していますが、従装置選択(SS)ピン機能を持ちません。けれども、この特性(機能)は必要ならばソフトウェアで実現できます。この動作で使われるピン名はDI,DO,USCKです。

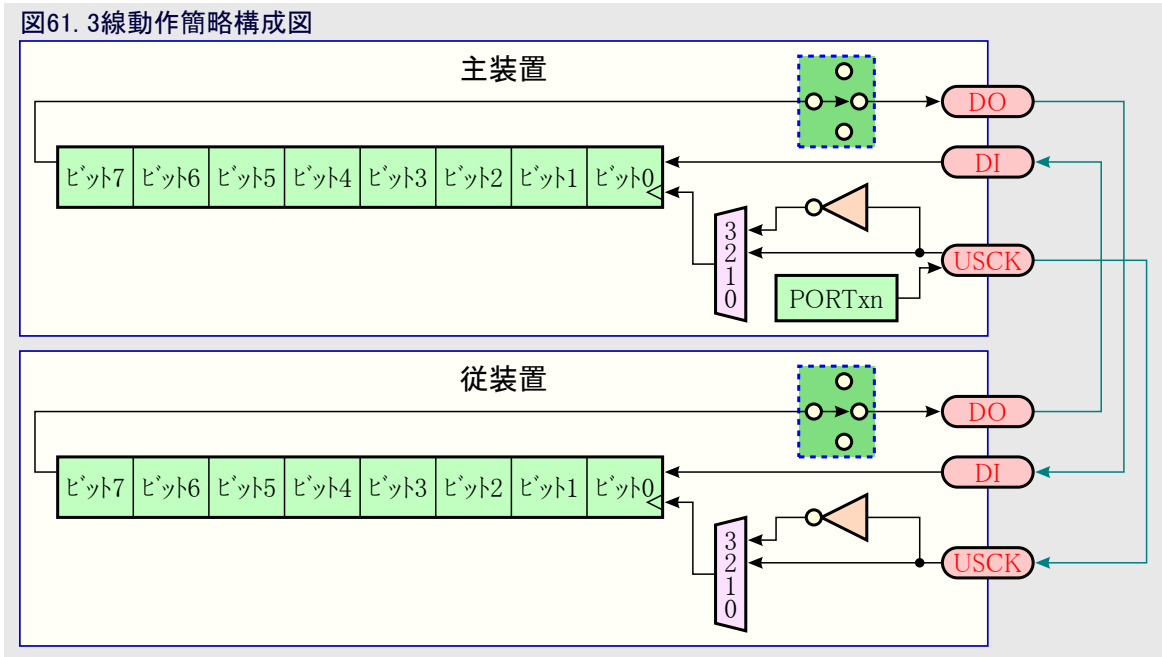
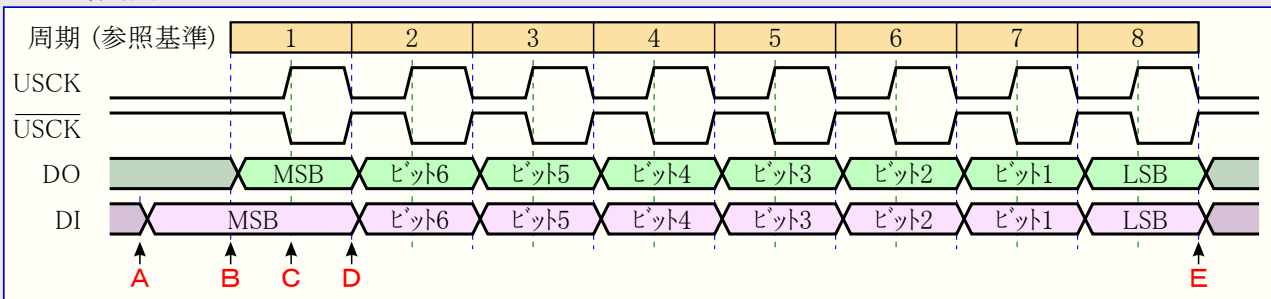


図61は3線動作での2つのUSI部（一方は主装置、他方は従装置）動作を示します。このような方法で連結された2つの移動レジスタは8 USCKクロック後、各々のレジスタのデータが交換されます。この同じクロック(USCK)がUSIの4ビットカウンタも増加(+1)します。従って**计数器溢れ割り込み要求フラグ(USIOIF)**は転送が完了される時を決めるのに使えます。このクロックはPORTレジスタ経由USCKピン1/0交互切り替え、またはUSICRのUSITCビットへの1書き込みによる主装置装置ソフトウェアによって生成されます。

図62. 3線動作タイミング



3線動作タイミングは図62で示されます。図の最上部はUSCK周期参照基準です。これらの各周期に対して1ビットが**USI移動レジスタ(USIDR)**に移動されます。USCKタイミングは両方の外部クロック動作について示されます。外部クロック動作0(USICS0=0)でのDIは上昇端で採取され、DOは下降端で変更(USIDRが1つ移動)されます。外部クロック動作1(USICS0=1)は外部クロック動作0に対して逆端を使い、換言すると下降端でデータを採取、上昇端で出力を変更します。USIクロック動作種別はSPIデータ動作種別0と1に対応します。

このタイミング図(図62.)を参照すると、バス転送は次の手順を含みます。

1. 主装置と従装置はデータ出力を設定し、使用規約に依存して出力駆動部を許可します(AとB)。この出力は直列データレジスタへの送信されるべきデータ書き込みによって設定します。出力の許可はポート方向レジスタの対応するビット設定によって行います。A点とB点はどんな特別な順番もありますが、両方共にデータが採取されるC点よりも最低1/2 USCK周期前でなければならないことに注意してください。これはデータ設定の必要条件を満足させるのを保証するために行わなければならないことです。4ビットカウンタは0にリセットします。
2. 主装置はUSCK線をソフトウェアで2度切り替えることによってクロックパルスを生成します(CとD)。主装置と従装置のデータ入力(DI)ピンのビット値は最初の端(エッジ)(C)でUSIによって採取され、データ出力は逆端(D)で変更されます。4ビットカウンタは両端で計数します。
3. レジスタ(バイト)転送完了のために手順2.が8回繰り返されます。
4. 8クロックパルス(換言すると16クロック端)後、カウンタが溢れて転送完了を示します。転送したバイトデータは新規転送が開始され得るのに先立って直ぐに処置されなければなりません。溢れ割り込みはプロセッサがアイドル動作に設定されているなら、プロセッサを起動復帰します。使用規約によって、従装置は直ぐに出力をHi-Z設定にできます。

SPI主装置操作例

次のコードはSPI主装置としてのUSI使用法を実際に示します。

アセンブリ言語プログラム例

```

SPIM:      OUT     USIDR, R16                ;送信データを設定
           LDI     R16, (1<<USIOIF)        ;USIOIFビットのみ1値を取得
           OUT     USISR, R16              ;フラグ解除/計数器初期化
           LDI     R16, (1<<USIWM0) | (1<<USICS1) | (1<<USICLK) | (1<<USITC) ;3線動作クロック生成値を取得
;
SPIM_LP:   OUT     USICR, R16                ;SCKクロック端発生
           SBIS    USISR, USIOIF            ;計数器溢れでスキップ
           RJMP    SPIM_LP                 ;計数器溢れまで継続
;
           IN      R16, USIDR                ;受信データを取得
           RET                                  ;呼び出し元へ復帰
    
```

このコードは8命令(+RET)のみ使用の容量最適化です。このコード例はDOとUSCKピンがDDR_Bで出力として許可される仮定です。この関数が呼び出される前にR16レジスタへ格納した値は従装置に転送され、転送が完了された時に従装置から受信したデータがR16レジスタに格納されて戻ります。

2行目と3行目の命令は計数器溢れ割り込み要求フラグ(USIOIF)を解除(0)し、USI 4ビット計数器値を解除(=0)します。4行目と6行目の命令は3線動作、上昇端クロック、USITCスローブ計数、SCK出力交互切り替え(PORTB7)を設定します。この繰り返しは16回繰り返されます。

次のコードは最高速($f_{SCK}=f_{CK}/2$)でのSPI主装置としてのUSI使用法を実際に示します。

アセンブリ言語プログラム例

```

SPIM_F:    OUT     USIDR, R16                ;送信データを設定
           LDI     R16, (1<<USIWM0) | (0<<USICS0) | (1<<USITC)        ;3線動作初期値を取得
           LDI     R17, (1<<USIWM0) | (0<<USICS0) | (1<<USITC) | (1<<USICLK) ;3線動作クロック生成値を取得
;
           OUT     USICR, R16                ;MSB転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット6転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット5転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット4転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット3転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット2転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;ビット1転送
           OUT     USICR, R17                ;
           OUT     USICR, R16                ;LSB転送
           OUT     USICR, R17                ;
;
           IN      R16, USIDR                ;受信データを取得
           RET                                  ;呼び出し元へ復帰
    
```

SPI従装置操作例

次のコードはSPI従装置としてのUSI使用法を実際に示します。

```

アセンブリ言語プログラム例
init:    LDI    R16, (1<<USIWM0) | (1<<USICS1)    ;3線動作,外部クロック値を取得
        OUT    USICR, R16                        ;3線動作,外部クロック設定

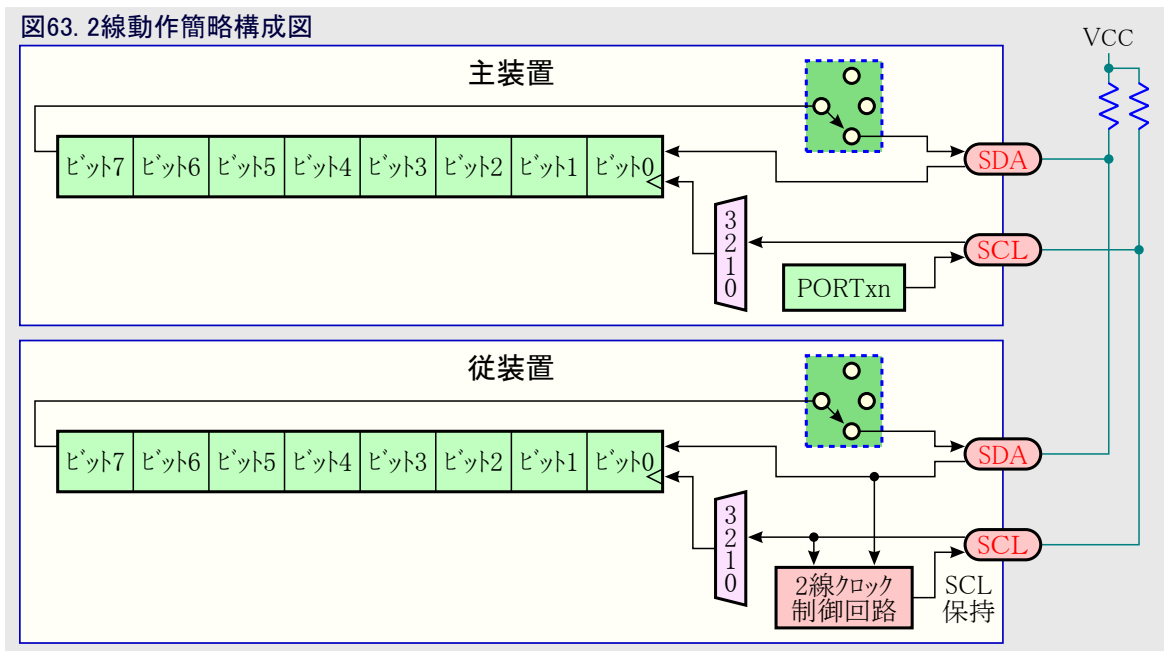
SPIS:    OUT    USIDR, R16                        ;送信データを設定
        LDI    R16, (1<<USIOIF)                  ;USIOIFビットのみ1値を取得
        OUT    USISR, R16                        ;フラグ解除/計数器初期化
;
SPIS_LP: SBIS   USISR, USIOIF                      ;計数器溢れでスキップ
        RJMP  SPIS_LP                            ;計数器溢れまで継続
;
        IN    R16, USIDR                          ;受信データを取得
        RET                                       ;呼び出し元へ復帰
    
```

このコードは8命令(+RET)のみ使用の容量最適化です。このコード例はDDRBLレジスタでDOピンが出力、USCKピンが入力として設定されると仮定します。この関数が呼び出される前にR16レジスタへ格納した値は主装置に転送され、転送が完了された時に主装置から受信したデータがR16レジスタに格納されて戻ります。

最初の2命令は初期化用だけで、一度だけ実行されるのを必要とすることに注意してください。これらの命令は3線動作と上昇端クロックを設定します。この繰り返しはUSI計数器溢れフラグが設定(1)されるまで繰り返されます。

2線動作

USIの2線動作はI²C(TWI)バス規約に準拠していますが、出力のスレーブ制限と入力雑音濾波器がありません。この動作で使われるピン名はSCLとSDAです。

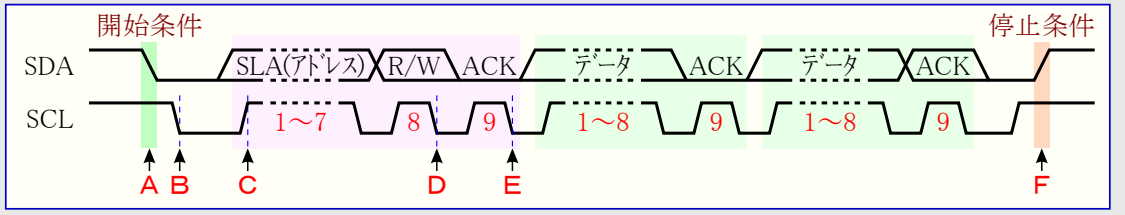


このクロックが(4ビット)計数器も増加(+1)するため、**計数器溢れ(USIOIF)**は転送が完了されたことを示すのに使えます。このクロックは主装置によるPORTBLレジスタ経由のSCLピン交互切り替えによって生成されます。

データ方向は物理層によって与えられません。データの流れを制御するためにTWIバスで使ったような或る種の規約が実装されなければなりません。

このタイミング図(図64.)を参照すると、バス転送は次の手順を含みます。

図64. 代表的な2線動作タイミング



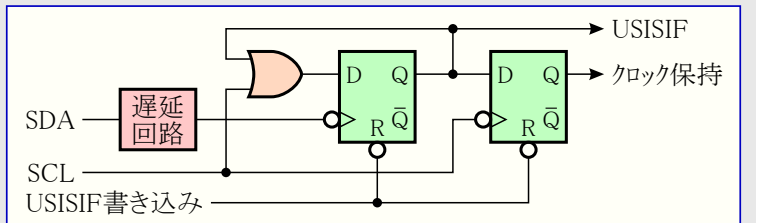
1. 開始条件は主装置によるSCL線High中のSDA線Low強制によって生成されます(A)。SDAは移動レジスタのビット7への0書き込みか、またはPORTレジスタで対応ビットを0に設定のどちらかによって強制実行できます。出力が許可されるにはデータ方向(DDR)レジスタのビットが1に設定されなければならないことに注意してください。従装置の開始条件検出器論理回路(図65.参照)が開始条件を検出してUSISIFフラグを設定(1)します。このフラグは必要ならば割り込みを発生できます。
2. 加えて開始条件検出器は主装置がSCL線で上昇端を強制してしまった後にSCL線をLowに保持します(B)。これはアドレスを受信するために移動レジスタを設定する前に休止形態から起動復帰または他の処理を完了することを従装置に許します。これ(保持解除)は(4ビット)計数器をリセット(=0)して開始条件検出フラグ(USISIF)を解除(0)することによって行われます。
3. 主装置は転送されるべき最初のビットを設定してSCL線を開放します(C)。従装置はSCLクロックの上昇端でデータを採取して移動レジスタへ移動します。
4. 従装置アドレスとデータ方向(R/W)を含む8ビットが転送された後、従装置の(4ビット)計数器が溢れてSCL線がLowを強制されます(D)。従装置は主装置の1つがアドレス指定してしまわない場合、SCL線を開放して新規開始条件を待ちます。
5. 従装置がアドレス指定されると、再びSCL線をLowに保持する前の応答(ACK)周期の間中、SDA線をLowに保持します(換言するとD点でSCLを開放する前に(4ビット)計数器レジスタは14に設定されなければなりません)。R/Wビットに依存して主装置か従装置は出力を許可します。このビットが設定(1)の場合、主装置読み込み動作が進行中です(換言すると従装置がSDA線を駆動します)。従装置は応答後(E)にSCL線をLowに保持することができます。
6. 主装置によって停止条件(F)か新規開始条件(再送開始条件)が与えられるまで全て同方向で複数バイトが直ぐに転送できます。従装置はより多くのデータを受信できない場合、最後に受信したバイトデータに回答しません。主装置が読み込み動作を行うとき、最後のバイトが転送された後の応答ビットをLowに強制することによってこの動作を終了しなければなりません。

開始条件検出器

開始条件検出器は図65.で示されます。SDA線はSCL線の有効な採取を保証するために(50~300nsの範囲で)遅延されます。開始条件検出器は2線動作でだけ許可されます。

開始条件検出器は非同期に動作し、従ってパワーダウン休止動作からプロセッサを起動できます。けれども使う規約がSCL保持時間で制限を持つかもしれません。従ってこの場合にこの特徴(機能)を使う時にCKSELヒューズで設定する発振器起動時間(16頁の「クロックシステムとその配給」参照)も考慮内に取り入れられなければなりません。より多くの詳細については97頁のUSISIFフラグ記述を参照してください。

図65. 開始条件検出器回路構成



USIでの代替使用

USI部を直列通信に使わないなら、柔軟な設計のため、代替処理を行うように設定できます。

半二重非同期データ転送

3線動作で移動レジスタを使うことによってソフトウェアだけよりも簡潔で高性能のUART実装が可能です。

4ビット計数器

4ビット計数器は溢れ割り込みとの自立型計数器として使えます。この計数器が外部的にクロック駆動される場合、両クロック端が増加(+1)を発生することに注意してください。

12ビット計数器

USIの4ビット計数器とタイマ/カウンタ0の組み合わせは12ビット計数器としての使用を許します。

端(エッジ)起動外部割り込み

4ビット計数器を最大値(\$F)に設定することによって追加外部割り込みとして機能できます。溢れフラグと割り込み許可ビットはこの外部割り込みに対して使われます。この特徴(機能)はUSICS1ビットによって選ばれ(=1)ます。

ソフトウェア割り込み

4ビット計数器の溢れ割り込みはクロックストロブによって起動するソフトウェア割り込みとして使えます。

USI用レジスタ

USIデータレジスタ (USI Data Register) USIDR

ビット	7	6	5	4	3	2	1	0	
\$0F (\$2F)	(MSB)							(LSB)	USIDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

USIは直列(移動)レジスタの緩衝をしません、換言するとデータレジスタ(USIDR)をアクセスする時に直列レジスタは直接的にアクセスされます。レジスタが書かれるのと同じ周期で直列クロックが起こると、レジスタは書いた値を含み、移動は実行されません。(左)移動動作はUSIC S1,0ビット設定に従って実行されます。この移動動作は外部クロック端、**タイマ/カウンタ0溢れ**により、または**USICLKストローブ**ビットを使うソフトウェアによって直接的に制御できます。2または3線動作種別が選ばれない(USIWM1,0=00)時でも、外部データ入力(DI/SDA)と外部クロック入力(USCK/SCL)の両方が未だ移動レジスタによって使われ得ることに注意してください。

動作種別に従って使う出力ピン(DOまたはSDA)は出力ラッチを経由してデータレジスタの最上位ビット(ビット7)に接続されます。この出力ラッチは外部クロック元が選ばれる(USICS1=1)時に直列クロック周期の前半中が透過(トランスペアレント)で、内部クロック元が使われる(USICS1=0)時に定常的な透過です。ラッチが透過である限り、新規MSBが書かれると、出力は直ちに更新されます。このラッチは違う(逆の)クロック端でデータ入力が採取され、データ出力が変更されるのを保証します。

移動レジスタからの出力データを許可するために、ピンに対応するデータ方向レジスタ(DDxn)が1に設定されなければならないことに注意してください。

USI状態レジスタ (USI Status Register) USISR

ビット	7	6	5	4	3	2	1	0	
\$0E (\$2E)	USISIF	USIOIF	USIPF	USIDC	USICNT3	USICNT2	USICNT1	USICNT0	USISR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この状態レジスタは割り込み要求フラグ、状態フラグ、計数値を含みます。

• ビット7 - USISIF : 開始条件検出割り込み要求フラグ (Start Condition Interrupt Flag)

2線動作が選ばれるとき、**開始条件**が検出されると、USISIFフラグが設定(1)されます。出力禁止動作または3線動作が選ばれ、(USICS1,0=11且つUSICLK=0)または(USICS1,0=10且つUSICLK=0)の時にUSCKピンのどの端(エッジ)もがこのフラグを設定(1)します。

ステータスレジスタ(SREG)の全割り込み許可(I)ビットと**USI制御レジスタ(USICR)**で**開始条件検出許可(USISIE)ビット**が設定(1)されている間中にこのフラグが設定(1)されると、割り込みが生成されます。このフラグはUSISIFビットへの論理1書き込みによってのみ解除(0)されます。このビットの解除(0)は2線動作でのSCLの**開始条件保持**を解除(開放)します。

開始条件検出割り込みは全ての**休止形態**からプロセッサを起動します。

• ビット6 - USIOIF : 計数器溢れ割り込み要求フラグ (Counter Overflow Interrupt Flag)

このフラグは4ビット計数器が溢れると(換言すると15から0への遷移で)設定(1)されます。SREGの全割り込み許可(I)ビットとUSICRで**計数器溢れ割り込み許可(USIOIE)ビット**が設定(1)されている間にこのフラグが設定(1)されると、割り込みが生成されます。本フラグはUSIOIFビットへの論理1書き込みによってのみ解除(0)されます。本ビットの解除(0)は2線動作でのSCLの**計数器溢れ保持**を解除(開放)します。計数器溢れ割り込みは**アイドル休止動作**からプロセッサを起動します。

• ビット5 - USIPF : 停止条件検出フラグ (Stop Condition Flag)

2線動作が選ばれた時に**停止条件**が検出されると、USIPFフラグが設定(1)されます。このフラグはこのビットへの1書き込みによって解除(0)されます。これが割り込み要求フラグでないことに注意してください。この合図は2線バス主装置調停の実装時に有用です。

• ビット4 - USIDC : 出力データ衝突フラグ (Data Output Collision Flag)

移動レジスタのビット7(MSB)が物理ピン値と異なる時にこのビットは論理1です。このフラグは2線動作が使われる時にだけ有効です。この合図は2線バス主装置調停の実装時に有用です。

• ビット3~0 - USICNT3~0 : 計数器値 (Counter Value)

これらのビットは現在の4ビット計数器値を反映します。この4ビット計数器値はCPUによって直接的に読み書きできます。

この4ビット計数器は外部クロック端検出器、**タイマ/カウンタ0溢れ**、**USICLK**または**USITC**ストローブビットを使うソフトウェアのどれかによって生成した各クロックに対して1つ増加します。このクロック元はUSICS1,0ビット設定に依存します。外部クロック動作に対してUSITCストローブビットへの書き込みによって生成されることをクロックに許す、特別な特徴が付加されます。この特徴は外部クロック元(USICS1=1)設定中のUSICLKへの1書き込みによって許可されます。

出力禁止動作(USIWM1,0=00)が選ばれている時でも、外部クロック入力(USCK/SCL)が未だ計数器によって使われることに注意してください。

USI制御レジスタ (USI Control Register) USICR

ビット	7	6	5	4	3	2	1	0	
\$0D (\$2D)	USISIE	USIOIE	USIWM1	USIWM0	USICS1	USICS0	USICLK	USITC	USICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	W	W	
初期値	0	0	0	0	0	0	0	0	

この制御レジスタは割り込み許可制御、動作種別設定、クロック選択設定、クロック ストローブを含みます。

•ビット7 – USISIE : 開始条件検出割り込み許可 (Start Condition Interrupt Enable)

このビットを1に設定することが**開始条件**検出割り込みを許可します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUSISIEビットが設定(1)される時に保留割り込みなら、これは直ちに実行されます。より多くの詳細については97頁の**開始条件検出割り込み要求フラグ(USISIF)**記述を参照してください。

•ビット6 – USIOIE : 計数器溢れ割り込み許可 (Counter Overflow Interrupt Enable)

このビットを1に設定することが計数器溢れ割り込みを許可します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUSIOIEビットが設定(1)される時に保留割り込みなら、これは直ちに実行されます。より多くの詳細については97頁の**計数器溢れ割り込み要求フラグ(USIOIF)**記述を参照してください。

•ビット5,4 – USIWM1,0 : 動作種別選択 (Wire Mode)

これらのビットは使われるべき動作種別(2または3線動作)を設定します。基本的に出力の機能だけがこれらのビットによって影響を及ぼされます。データとクロック入力は選んだ動作種別によって影響を及ぼされず、常に同じ機能を持ちます。従って出力が禁止される時でも移動レジスタと計数器は外部的にクロック駆動でき、入力データが採取されます。USI操作とUSIWM1,0間の関連は表59.で要約されます。

表59. USI操作とUSIWM1,0の関係

USIWM1	USIWM0	説明
0	0	出力、クロック保持、開始条件検出器は禁止。ポートピンは標準として動作。
0	1	3線動作。DO,DI,USCKピンを使用。 データ出力(DO)ピンはこの動作でPORTレジスタの対応ビットを無視します。けれども対応するDDRビットは未だデータ方向を制御します。ポートピンが入力として設定(DDxn=0)されると、そのピンのプルアップはPORTビットによって制御されます。 データ入力(DI)と直列クロック(USCK)ピンは標準ポート操作に影響を及ぼしません。主装置として動作する時のクロックパルスはデータ方向が出力に設定されている(DDxn=1)間のPORTレジスタの交互切り替えによってソフトウェアで生成されます。USICRのUSITCビットがこの目的に使えます。
1	0	2線動作。SDA(DI)とSCL(USCK)ピンを使用。(注1) 直列データ(SDA)と直列クロック(SCL)ピンは双方向でオープンドレイン出力駆動を使います。この出力駆動部はDDRレジスタでSDAとSCLに対応するビットの設定(=1)によって許可されます。 SDAピンに対して出力駆動部が許可されると、出力駆動部は移動レジスタの出力またはPORTレジスタの対応ビットが0ならばSDA線をLowに強制します。さもないければ、SDA線は駆動されません(換言すると開放されます)。SCLピン出力駆動部が許可されると、SCL線はPORTレジスタの対応ビットが0なら、または開始条件検出器によってLowを強制されます。さもないければSCL線は駆動されません。 SCL線は出力が許可され、開始条件検出器が 開始条件 を検出すると、Lowに保持されます。 開始条件検出フラグ(USISIF) の解除(0)がその線を開放します。SDAとSCLピン入力はこの動作の許可によって影響を及ぼされません。SDAとSCLピンのプルアップは2線動作で禁止されます。
1	1	2線動作。SDAとSCLピンを使用。 SCL線は計数器溢れが起こる時にもLowへ保持され、 計数器溢れフラグ(USIOIF) が解除(0)されるまでLowに保持されることを除いて、上で記述される2線動作と同じ動作です。

注1: DIとUSCKピンは動作種別間での混乱を避けるため、各々直列データ(SDA)と直列クロック(SCL)に改称されます。

•ビット3,2 – USICS1,0 : クロック選択 (Clock Source Select)

これらのビットは移動レジスタと計数器に対するクロック元を設定します。外部クロック元(USCK/SCL)を使う時にデータ出力ラッチはデータ入力(DI/SDA)の採取の逆端で出力が変更されるのを保証します。タイマ/カウンタ0溢れまたはソフトウェア ストローブが選ばれると、出力ラッチは透過で、従って出力は直ちに変更されます。USICS1,0ビットの解除(=00)がソフトウェア ストローブ選択を許可します。この選択を使う時のUSICLKビットへの1書き込みは移動レジスタと計数器の両方をクロック駆動します。外部クロック元(USICS1=1)に対してUSICLKビットはもはやストローブとして使われませんが、外部クロックとUSITCストローブ ビットによるソフトウェア クロック間を選びます。

表60.は移動レジスタと4ビット計数器に対して使われるクロック元とUSICS1,0、USICLK設定間の関連を示します。

表60. USICS1,0とUSICLK設定との関係

USICS1	USICS0	USICLK	移動レジスタ クロック元	4ビット計数器クロック元
0	0	0	クロックなし	クロックなし
		1	ソフトウェア クロック ストローブ (USICLK)	ソフトウェア クロック ストローブ (USICLK)
	1	X	タイマ/カウンタ0溢れ	タイマ/カウンタ0溢れ
1	0	0	外部クロック上昇端	外部クロック両端
	1		外部クロック下降端	
	0	1	外部クロック上昇端	ソフトウェア クロック ストローブ (USITC)
	1		外部クロック下降端	

•ビット1 – USICLK : クロック ストローブ (Clock Strobe)

ソフトウェア ストローブ選択が選ばれるように行うことによってUSICS1,0ビットがゼロ(=00)に設定されるなら、このビット位置への1書き込みは移動レジスタを1段移動して計数器を1つ増加します。このクロック ストローブが実行される時、直ちに、換言すると同じ命令周期内で出力が替わります。移動レジスタ内に移動した値は直前の命令周期で採取されます。このビットは0として読みます。

外部クロック元が選ばれる(USICS1=1)と、USICLK機能はクロック ストローブからクロック選択レジスタに変更されます。この場合のUSICLKビットの設定(=1)は4ビット計数器に対するクロック元としてのUSITCストローブを選びます(表60.参照)。

•ビット0 – USITC : クロック値切り替え (Toggle Clock Port Pin)

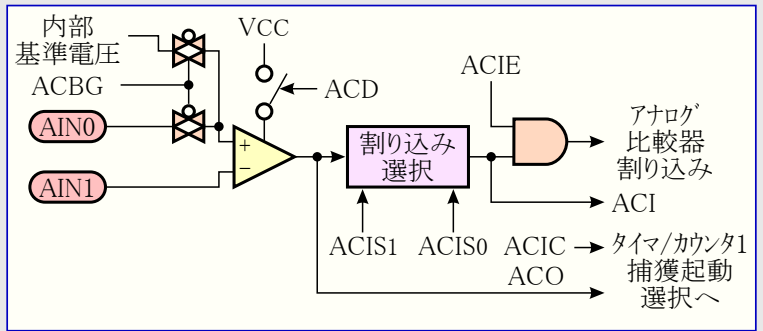
このビット位置に1を書くことはUSCK/SCL値を0から1、または1から0のどちらかへ切り替えます。この切り替えはデータ方向レジスタでの設定と無関係ですが、PORT値がピンで見られるべきなら、DDB7が出力として設定(=1)されなければなりません。この特徴は主装置を実現する時の容易なクロック生成を許します。このビットは0として読みます。

外部クロック元が選ばれ(USICS1=1)、USICLKビットが1に設定されると、USITCストローブ ビットへの書き込みは直接的に4ビット計数器をクロック駆動します。これは主装置として動作する時に転送が行われる時の早い検出を許します。

アナログ比較器

アナログ比較器は非反転入力AIN0ピンと反転入力AIN1ピンの入力値を比較します。非反転AIN0ピンの電圧が反転AIN1ピンの電圧よりも高い時にACSRのアナログ比較器出力(ACO)ビットが設定(1)されます。この比較器出力はタイマ/カウンタ1の捕獲機能を起動するように設定できます。加えて、この比較器はアナログ比較器専用の、独立した割り込みを起動できます。使用者は比較器出力の上昇端、下降端、またはその両方での割り込み起動を選べます。この比較器とその周辺論理回路の構成図は図66.で示されます。

図66. アナログ比較器部構成図



注: アナログ比較器のピン配置については2頁の「ピン配置」と36頁の表26.を参照してください。

アナログ比較器 制御/状態レジスタ (Analog Comparator Control and Status Register) ACSR

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	0	0	0	0	0	

• ビット7 – ACD : アナログ比較器禁止 (Analog Comparator Disable)

このビットが論理1を書かれると、アナログ比較器への電力がOFFにされます。このビットはアナログ比較器をOFFにするために何時でも設定(1)できます。これは活動動作やアイドル動作で電力消費を削減します。ACDビットを変更する時にACSRでアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、このビットが変更される時に割り込みが起り得ます。

• ビット6 – ACBG : 基準電圧選択 (Analog Comparator Bandgap Select)

このビットが設定(1)されると、内部基準電圧(=公称1.1V)がアナログ比較器への非反転入力に取って代わります。このビットが解除(0)されると、AIN0がアナログ比較器の非反転入力に印加されます。内部基準電圧がアナログ比較器入力として使われる時に電圧の安定に一定時間を必要とします。安定(待機を)しない場合、最初は不正値を与えるかもしれません。25頁の「内部基準電圧」をご覧ください。

• ビット5 – ACO : アナログ比較器出力 (Analog Comparator Output)

アナログ比較器の出力は同期化され、その後直接ACOへ接続されます。この同期化は1~2クロック周期の遅延をもたらします。

• ビット4 – ACI : アナログ比較器割り込み要求フラグ (Analog Comparator Interrupt Flag)

このビットは比較器出力での出来事がACSRのアナログ比較器割り込み条件(ACIS1,0)ビットによって定義した割り込み方法で起動する時に設定(1)されます。ACSRのアナログ比較器割り込み許可(ACIE)ビットが設定(1)され、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みルーチンが実行されます。対応する割り込み処理ベクタを実行すると、ACIはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもACIは解除(0)されます。

• ビット3 – ACIE : アナログ比較器割り込み許可 (Analog Comparator Interrupt Enable)

ACIEビットが論理1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みが活性(有効)にされます。論理0を書かれると、この割り込みは禁止されます。

• ビット2 – ACIC : アナログ比較器捕獲起動許可 (Analog Comparator Input Capture Enable)

論理1を書かれると、このビットはアナログ比較器によって起動されるタイマ/カウンタ1の捕獲機能を許可します。この場合、比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音消去機能と端(エッジ)選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。論理0を書かれると、アナログ比較器と捕獲機能間の接続は存在しません。比較器がタイマ/カウンタ1捕獲割り込みを起動するには、タイマ/カウンタ割り込み許可レジスタ(TIMSK)の捕獲割り込み許可(ICIE1)ビットが設定(1)されなければなりません。

• ビット1,0 – ACIS1,0 : アナログ比較器割り込み条件 (Analog Comparator Interrupt Mode Select)

これらのビットは比較器のどの事象がアナログ比較器割り込みを起動するのかを決めます。各種設定は表61.で示されます。

ACIS1,ACIS0ビットを変更する時にACSRのアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、これらのビットが変更される時に割り込みが起き得ます。

表61. アナログ比較器割り込み条件選択

ACIS1	ACIS0	割り込み発生条件
0	0	比較器出力の変移(トグル)
0	1	(予約)
1	0	比較器出力の下降端
1	1	比較器出力の上昇端

デジタル入力禁止レジスタ (Digital Input Disable Register) DIDR

ビット	7	6	5	4	3	2	1	0	
\$01 (\$21)	-	-	-	-	-	-	AIN1D	AIN0D	DIDR
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット1,0 – AIN1D,AIN0D : AIN1,AIN0 デジタル入力禁止 (AIN1,AIN0 Digital Input Disable)

このビットが論理1を書かれると、AIN1/0ピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、対応するポート入力レジスタのビット(PINx)は常に0として読みます。AIN1/0ピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされない時にデジタル入力緩衝部での消費電力を削減するため、このビットは論理1を書かれるべきです。

デバッグWIRE 内蔵デバッグ システム

特徴

- 完全なプログラムの流れ制御
- RESETピンを除くデジタルとアナログ両方でのチップ全機能のエミュレート
- 実時間(リアルタイム)動作
- シンボリックデバッグ支援 (アセンブリ及びC言語または他の高位言語)
- 無制限数のプログラム中断点(ブレークポイント:ソフトウェア中断点使用)
- 邪魔しない動作
- 実デバイスと同じ電気的特性
- 自動設定システム
- 高速動作
- 不揮発性メモリプログラミング

概要

デバッグWIRE内蔵デバッグシステムは、CPUでのAVR命令実行、プログラムの流れ制御、各種不揮発性メモリプログラミングのための1本線の双方向インターフェースを使います。

物理インターフェース

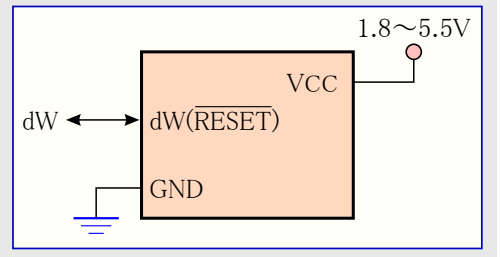
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

図67.はエミュレータと許可したデバッグWIREでの対象MCUとの接続の図を示します。システムクロックはデバッグWIREによって影響を及ぼされず、常にCKSELヒューズで選んだクロック元です。

デバッグWIREが使われるシステム的设计時、正しい動作のために次の注意点が厳守されなければなりません。

- dW/(RESET)線のプルアップ抵抗は10kΩよりも大きくなければなりません。とは言え、プルアップ抵抗は任意選択です。
- RESETピンのVCCへの直接的な接続では動作しません。
- RESETピンに挿入したコンデンサはデバッグWIRE使用時、切断されなければなりません。
- 全ての外部リセット元は切断されなければなりません。

図67. デバッグWIRE構成図



ソフトウェア中断点 (ブレークポイント)

デバッグWIREはAVRのBREAK命令によってプログラムメモリの中断点を支援します。AVR Studio®での中断点設定はプログラムメモリにBREAK命令を挿入します。BREAK命令で置換した(元の)命令は保存されます。プログラム実行が継続されると、プログラムメモリから継続される前に保存した命令が実行されます。一時停止(ブレーク)はプログラムにBREAK命令を置くことによって手動で挿入できます。

フラッシュメモリは中断点の変更される度毎に再書き換えされなければなりません。これはデバッグWIREインターフェースを通してAVR Studioによって自動的に操作されます。従って中断点の使用はフラッシュメモリのデータ保持力を低下させます。デバッグ目的に使ったデバイスは最終顧客へ出荷すべきではありません。

デバッグWIREの制限

デバッグWIRE通信(dW)ピンは物理的に外部リセット(RESET)と同じピンに配置されます。従ってデバッグWIREが許可されると、外部リセット元が支援されません。

デバッグWIREシステムは全速度、換言するとCPUのプログラムが走行する時に全I/O機能を正確エミュレートします。CPUが停止される時にデバッグ(AVR Studio)経由でいくつかのI/Oレジスタをアクセスする間、注意が妨げられなければなりません。この制限の詳細説明についてはデバッグWIRE資料をご覧ください。

プログラム(0)にしたDWENヒューズは全休止形態でクロック系のいくつかの部分の走行を許可します。これは休止間中の消費電力を増加します。従ってDWENヒューズはデバッグWIREが使われない場合、禁止されるべきです。

デバッグWIREに関連するI/Oメモリ内のレジスタ

次項はデバッグWIREで使うレジスタを記述します。

デバッグWIRE データレジスタ (debugWIRE Data Register) DWDR

ビット	7	6	5	4	3	2	1	0	
\$-- (\$--)	(MSB)							(LSB)	DWDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

DWDRはMCU内で走行するプログラムからデバッグへの通信チャネルを提供します。このレジスタはデバッグWIREでだけアクセス可能で、従って通常動作で一般目的レジスタとして使えません。

フラッシュメモリの自己プログラミング

本デバイスはMCU自身によるプログラムコードのダウンロードとアップロード用の自己プログラミング機構を提供します。自己プログラミングはフラッシュメモリ内にコードを書き(プログラム)、コードを読み、またはプログラムメモリからコードを読むのに、利用可能なデータインターフェースと関連する規約のどれもが使えます。SPM命令は既定で禁止されますが、SELFPRGENヒューズのプログラム(0)によって許可することができます。

プログラムメモリはページ単位形式で更新されます。ページ一時緩衝部へ格納したデータでページを書く前にそのページは消去されなければなりません。ページ一時緩衝部はSPM命令使用時毎の1語(ワード)で満たされ、この緩衝部はページ消去命令前、またはページ消去とページ書き込み操作間のどちらかで満たすことができます。

手段1 (ページ消去前の一時緩衝部格納)

- ・ページ一時緩衝部を満たしてください。
- ・ページ消去を実行してください。
- ・ページ書き込みを実行してください。

手段2 (ページ消去後の一時緩衝部格納)

- ・ページ消去を実行してください。
- ・ページ一時緩衝部を満たしてください。
- ・ページ書き込みを実行してください。

ページの一部の変更だけが必要な場合、消去前にページの残す部分は(例えばページ一時緩衝部に)保存されなければならない、その後に変更して書かれます。手段1.を使う場合、初めにページを読んで必要な変更を行い、その後に変更したデータを書き戻すことをユーザーソフトウェアに許す効率的な読み-修正-書き(リード モデファイ ライト)機能をデバイスが提供します。手段2.が使われる場合、ページが既に消去されているため、格納中の旧データを読むことができません。ページ一時緩衝部は乱順でアクセスできます。ページ消去とページ書き込み操作の両方で使われるページアドレスは同じページをアドレス指定することが非常に重要です。

SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタにアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'0000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGEに書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- ・ページ消去中、CPUは停止されます。

ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタにアドレス、R1:R0にデータを設定してSPMCSRに'0000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORDの内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのページ一時緩衝部消去(CTPB)ビット書き込みによって自動的に消去されます。システムリセット後も消去されています。一時緩衝部を消去せずに各アドレスへ複数回書くことができないことに注意してください。

SPMページ設定操作の途中でEEPROMが書かれると、設定した全データが失われます。

ページ書き込みの実行

ページ書き込みを行うにはZポインタにアドレスを設定してSPMCSRに'0000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGEに書かれなければなりません。この操作の間、Zポインタの他のビットは0を書かれなければなりません。

- ・ページ書き込み中、CPUは停止されます。

自己プログラミング中のフラッシュメモリのアドレス指定

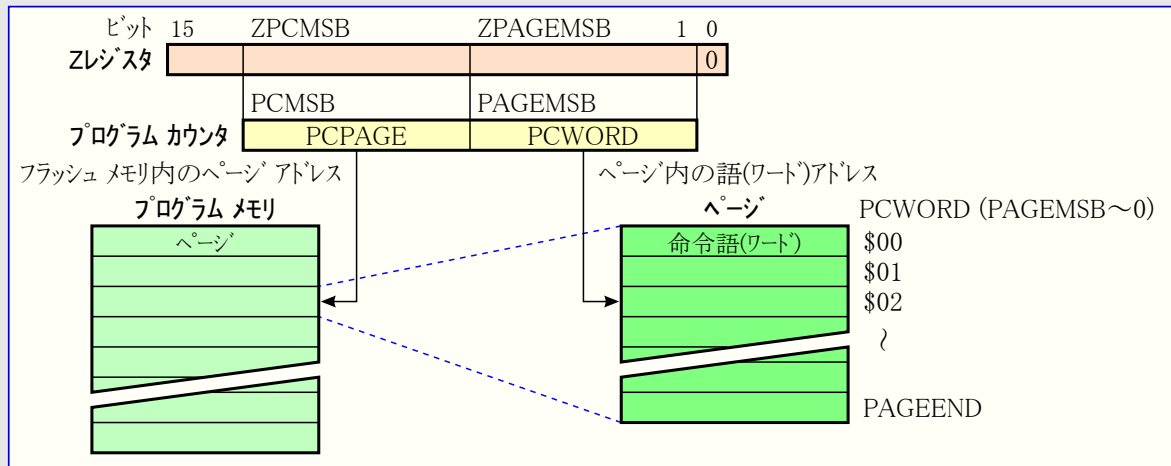
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

ビット	15	14	13	12	11	10	9	8
ZH (R31)	Z15	Z14	Z13	Z12	Z11	Z10	Z9	Z8
ビット	7	6	5	4	3	2	1	0
ZL (R30)	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

フラッシュメモリがページで構成されるため(107頁の表68参照)、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つようになり扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは次頁の図68.で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってソフトウェアがページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。

LPM命令はアドレスを格納するのにZポインタを使います。この命令はフラッシュメモリのバイト単位をアドレス指定するので、Zポインタの最下位ビット(Z0)も使われます。

図68. SPM操作中のフラッシュメモリのアドレス指定



注: 図内で使った各変数は107頁の表68.で一覧されます。

SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register) SPMCSR

このレジスタはプログラムメモリ操作を制御するために必要とする制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$37 (\$57)	-	-	-	CTPB	RFLB	PGWRT	PGERS	SELFPRGEN	SPMCSR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7~5 - Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

• **ビット4 - CTPB : ページ一時緩衝部消去 (Clear Temporary Page Buffer)**

ページ一時緩衝部を満たしている間にCTPBビットが1を書かれると、ページ一時緩衝部は消去され、データが失われます。

• **ビット3 - RFLB : ヒューズ/施錠ビット読み込み (Read Fuse and Lock Bits)**

SPMCSRでRFLBとSELFPRGENが設定(1)された後の3クロック周期内のLPM命令は(ZポインタのZ0によって)ヒューズビットまたは施錠ビットのどちらかを転送先レジスタに読みます。詳細については105頁の「ソフトウェアからのヒューズビットと施錠ビットの読み出し」をご覧ください。

• **ビット2 - PGWRT : ページ書き込み (Page Write)**

このビットがSELFPRGENと同時に1を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ全体の書き込み動作中、CPUは停止されます。

• **ビット1 - PGERS : ページ消去 (Page Erase)**

このビットがSELFPRGENと同時に1を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ全体の消去中、CPUは停止されます。

• **ビット0 - SELFPRGEN : 自己プログラミング許可 (Self Programming Enable)**

このビットは次の4クロック周期間SPM命令を許可します。このビットがCTPB,RFLB,PGWRT,PGERSのどれかと共に1を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。SELFPRGENだけが書かれると、続くSPM命令はZポインタでアドレス指定したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SELFPRGENビットはSPM命令の完了で、または4クロック周期内にSPM命令が実行されない場合自動的に解除(0)されます。ページ消去とページ書き込み中、SELFPRGENビットはその動作が完了されるまで1に留まります。

下位5ビットに10001, 01001, 00101, 00011, 00001以外のどんな組み合わせを書いても無効です。

(訳注) ビット0の名称SELFPRGENはヒューズビットに同一名が存在します。状況によっては不都合かもしれません。従ってソフトウェアに於ける本ビットの参照は、他の多くのAVRと同様にSPMENの使用が推奨されます。AVR Studioでのデバイス定義インクルードファイルでは既に本ビットに対してシンボルSPMENが定義されており、SELFPRGENはヒューズビット定義になっています。

SPM命令での書き込み時のEEPROM書き込みによる妨害

EEPROM書き込み動作がフラッシュメモリへの全ソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EPE)ビットを検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前に、このビットが解除(0)されているのを確認することが推奨されます。

ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビットの両方を読むことが可能です。施錠ビットを読むにはZポインタに\$0001を設定し、SPMCSRのSELFPRGENとフラッシュ/施錠ビット読み込み(RFLB)ビットを設定(1)してください。SELFPRGENとRFLBビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SELFPRGENとRFLBビットは施錠ビット読み出しの完了で、または3 CPU周期内にLPM命令が実行されないか、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SELFPRGENとRFLBビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	-	LB2	LB1

ヒューズ下位ビットを読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポインタに\$0000を設定してSPMCSRのSELFPRGENとRFLBビットを設定(1)してください。SELFPRGENとRFLBビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ下位ビット(FLB)の値が転送先レジスタに格納されます。ヒューズ下位ビットの配置と詳細な記述については107頁の表67を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様にヒューズ上位ビットを読むにはZポインタに\$0003を設定してください。SELFPRGENとRFLBビットがSPMCSRで設定(1)された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ上位ビット(FHB)の値が転送先レジスタに格納されます。ヒューズ上位ビットの配置と詳細な記述については106頁の表66を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

プログラム(0)されたヒューズと施錠ビットは0として読みます。非プログラム(1)にされたヒューズと施錠ビットは1として読みます。

フラッシュメモリデータ化けの防止

低VCCの期間中、CPUとフラッシュメモリの正しい動作に対して供給電圧が低すぎるためにフラッシュメモリのプログラムが不正にされ得ます。これらの問題はフラッシュメモリを使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

フラッシュメモリのプログラム化けは電圧が低すぎる時の2つの状態によって起こされます。1つ目としてフラッシュメモリへの通常の書き込み手順は正しく動作するための最低電圧が必要です。2つ目として供給電圧が低すぎると、CPU自身が命令を間違えて実行し得ます。

フラッシュメモリ化けは次の推奨設計によって容易に避けられます(1つは必須)。

- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアをパワーダウン休止動作に保ってください。これはCPUが命令の復号と実行を試みることを防ぎ、SPMCSR従ってフラッシュメモリを予期せぬ書き込みから効果的に保護します。

SPM命令使用時のフラッシュメモリ用プログラミング(書き込み)時間

校正された内蔵RC発振器がフラッシュメモリアクセス時間に使われます。表62はCPUからのフラッシュメモリアクセスに対する代表的なプログラミング時間を示します。

表62. SPM命令によるフラッシュメモリのプログラミング時間

項目	最小	最大
SPM命令によるフラッシュ書き込み (ページ消去、ページ書き込み)	3.7ms	4.5ms

メモリプログラミング

本章はATtiny2313のメモリプログラミングについての各種方法を記述します。

プログラムメモリとデータメモリ用施錠ビット

ATtiny2313は非プログラム(1)のままか、表64.で一覧される付加機能を得るためにプログラム(0)できる2つの施錠ビットを提供します。この施錠ビットはチップ消去指令でのみ1に消去できます。

SELFPRGENヒューズがプログラム(0)されると、SPM命令はフラッシュメモリ全体に対して許可され、さもなければ禁止されます(訳注:追加)。

表63. 施錠ビットバイトの内容

名称	ビット番号	意味	既定値(注)
-	7~2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表64. 施錠ビットの保護種別

メモリ施錠ビット			保護種別
LB 種別	LB2	LB1	直列、並列、またはデバッグWIRE経由プログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズビットが固定されます。(注1)

注: 0はプログラム、1は非プログラムを意味します。

注1: 施錠ビットを書く前にヒューズビットを書いてください。デバッグWIREは全面的に禁止されます。

ヒューズビット

ATtiny2313には3つのヒューズバイトがあります。表65.~67.は全てのヒューズの概略機能とヒューズバイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

表65. 拡張ヒューズバイト一覧

名称	ビット	意味	既定値
-	7~1		1 (非プログラム)
SELFPRGEN	0	自己プログラミング機能許可。	1 (非プログラム) 自己プログラミング不許可

注1: SPM命令を許可します。103頁の「フラッシュメモリの自己プログラミング」をご覧ください。

表66. ヒューズ上位バイト一覧

名称	ビット	意味	既定値
DWEN(注3)	7	デバッグWIRE機能許可。	1 (非プログラム) デバッグWIRE不許可
EESAVE	6	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
SPIEN(注1)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON(注2)	4	ウォッチドッグ タイム常時有効。	1 (非プログラム) WDTはWDTCSRで許可
BODLEVEL2	3	低電圧検出(BOD)リセットの制御と検出電圧選択。(注4)	1 (非プログラム)
BODLEVEL1	2		1 (非プログラム)
BODLEVEL0	1		1 (非プログラム)
RSTDISBL(注5)	0	PA2がI/OピンかまたはRESETピンかを選びます。	1 (非プログラム) PA2はRESETピン

注1: SPIENヒューズは直列プログラミングでアクセスできません。

注2: 詳細については28頁の「ウォッチドッグ タイム制御/状態レジスタ - WDTCSR」をご覧ください。

注3: 施錠ビットの設定に拘らず、DWENヒューズがプログラム(0)された製品の出荷は決してありません。プログラム(0)されたDWENヒューズは全休止形態でクロック系の一部を許可します。これは消費電力の増加になってしまいます。

注4: BODLEVELヒューズの符号化については24頁の表15.をご覧ください。

注5: RSTDISBLヒューズの記述については34頁の「ポートAの交換機能」をご覧ください。

表67. ヒューズ下位バイト一覧

名称	ビット	意味	既定値
CKDIV8	7	システムクロック 8分周選択。	0 (プログラム) 8分周
CKOUT	6	システムクロック出力許可。	1 (非プログラム) 不許可
SUT1	5	起動時間選択。	1 (非プログラム)
SUT0	4		0 (プログラム)
CKSEL3	3	クロック種別選択。	0 (プログラム)
CKSEL2	2		1 (非プログラム)
CKSEL1	1		0 (プログラム)
CKSEL0	0		0 (プログラム)

注1: SUT1,0の既定値は既定クロック元に対して最大起動時間になります。詳細については18頁の表7をご覧ください。

注2: 既定のCKSEL3~0設定は8MHz校正付き内蔵RC発振器になります。詳細については18頁の表6をご覧ください。

ヒューズビットの状態はチップ消去によって影響されません。施錠ビット1(LB1)がプログラム(0)されると、ヒューズビットが固定されることに注意してください。施錠ビットをプログラム(0)する前にヒューズビットをプログラミング(書き込み)してください。

ヒューズのラッチ

ヒューズ値はデバイスがプログラミング動作へ移行する時にラッチされ、ヒューズ値の変更はデバイスがプログラミング動作を去るまで無効です。これは一旦プログラム(0)されると直ぐに効果があるEESAVEヒューズには適用されません。ヒューズは通常動作での電源投入でもラッチされません。

識票バイト

全てのAtmelマイクロコントローラはデバイスを識別する3バイトの識票符号を持ちます。この符号は直列と並列の両プログラミング動作で、またデバイスが施錠されていても読めます。この3バイトは分離された空間に存在します。ATtiny2313の識票バイトを右に示します。

- ① \$000 : \$1E 製造業者Atmelを示します。
- ② \$001 : \$91 フラッシュメモリ容量2Kバイトを示します。
- ③ \$002 : \$0A ②値\$91と合せ、ATtiny2313を示します。

校正バイト

ATtiny2313の識票領域は内蔵RC発振器用に1バイトの校正データを持ちます。このバイトはアドレス\$0000の上位バイトに存在します。校正付き内蔵RC発振器の正しい周波数を保証するため、リセット中にこのバイトが18頁の「発振校正レジスタ - OSCCAL」へ自動的に書かれ(設定されます)。

ページ容量

表68. フラッシュメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
1K語 (2Kバイト)	16語	PC3~0	64	PC9~4	9

表69. EEPROMメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
128バイト	4バイト	EEA1,0	32	EEA6~2	6

並列プログラミング

この項はATtiny2313のプログラム用フラッシュメモリ、データ用EEPROM、メモリの**施錠ビット**、**ヒューズビット**の並列プログラミングと照合の方法を記述します。特記事項を除いて、パルス幅は最低250nsと仮定されます。

信号名

本章でATtiny2313のいくつかのピンは並列プログラミング中の機能を表す信号名で参照されます。**図69**と**表70**をご覧ください。後続の表で記載されないピンはピン名によって参照されます。

XA0とXA1ピンはXTAL1ピンが正パルスを与えられる時に実行される動作を決めます。このビット符号化は**表72**で示されます。

WRまたはOEパルス送出時、設定された指令が実行される動作を決めます。各種指令は**表73**で示されます。

図69. 並列プログラミング構成図

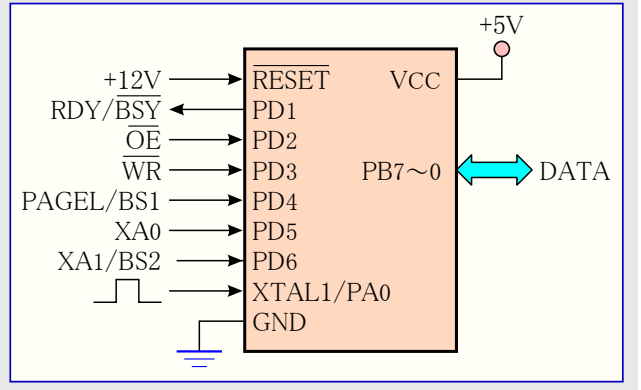


表70. 信号名とピン名の関係

信号名	ピン名	入出力	機能
RDY/BSY	PD1	出力	0(Low) : 多忙(プログラミング中) 1(High) : 準備可(指令受付可)
OE	PD2	入力	出力許可(負論理)
WR	PD3	入力	書き込みパルス(負論理)
PAGEL/BS1 (注)	PD4	入力	ページ一時緩衝部に設定 / 上位/下位バイト選択1 (0:下位, 1:上位) (一般用)
XA0	PD5	入力	XTAL動作ビット0
XA1/BS2 (注)	PD6	入力	XTAL動作ビット1 / 上位/下位バイト選択2 (0:下位, 1:上位) (ヒューズビット用)
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

注: このピンは2つの異なる制御信号に対して使われます。以降の記述で、通常1つの信号(名)だけが参照されます。例えば、「BS1に正パルスを与える」は「PAGEL/BS1に正パルスを与える」と同等です。

表71. プログラミング動作移行用のピン値

ピン名	シンボル	値
XA1/BS2	Prog_enable[3]	0
XA0	Prog_enable[2]	0
PAGEL/BS1	Prog_enable[1]	0
WR	Prog_enable[0]	0

表72. XA0とXA1の符号化(機能)

XA1	XA0	XTAL1パルス時の動作
0	0	フラッシュまたはEEPROMのアドレス取得 (上位/下位はBS1で指定)
0	1	データ取得 (フラッシュ時の上位/下位はBS1で指定)
1	0	指令取得
1	1	アイドル (動作なし)

注: XA1=1, XA0=1は非動作(アイドル)です。XTAL1がパルス駆動されない限り、指令、アドレス、データのレジスタは無変化に留まります。従ってBS2がXA1と多重されても、以降で記述されるようなBS2使用は問題ありません。BS2はヒューズ読み出し(OE=Low)の時だけ利用(1)され、(このとき)XTAL1はパルス駆動されません。

表73. 指令バイトのビット符号化

指令バイト	指令の機能
\$80 (1000 0000)	チップ消去
\$40 (0100 0000)	ヒューズビット書き込み
\$20 (0010 0000)	施錠ビット書き込み
\$10 (0001 0000)	フラッシュメモリ書き込み
\$11 (0001 0001)	EEPROM書き込み
\$08 (0000 1000)	識票バイト、校正バイト読み出し
\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$02 (0000 0010)	フラッシュメモリ読み出し
\$03 (0000 0011)	EEPROM読み出し

(訳注) 原書本位置の表74.は共通性のため「直列プログラミング」項へ移動しました。

並列プログラミング手順

並列プログラミング動作への移行

次に示す方法がデバイスを(高電圧)並列プログラミング動作にします。

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、108頁の表71.で一覧されるProg_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5～5.5Vを印加します。
- ③ 20 μ s以内にVCCが最低1.8Vに達することを保証してください。
- ④ 20～60 μ s待ち、 $\overline{\text{RESET}}$ に11.5～12.5Vを印加します。
- ⑤ Prog_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 μ s、Prog_enableピンを無変化に保ちます。
- ⑥ 如何なる並列プログラミング指令を与えるのにも先立って少なくとも300 μ s間待ちます。
- ⑦ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

VCCの上昇時間が上で示した必要条件を完全に満たせない場合、次の代替手順が使えます。

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、108頁の表71.で一覧されるProg_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5～5.5Vを印加します。
- ③ VCCを監視し、0.9～1.1Vに達したら直ぐ、 $\overline{\text{RESET}}$ に11.5～12.5Vを印加します。
- ④ Prog_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 μ s、Prog_enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立ってVCCが実際に4.5～5.5Vに達するまで待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

効率的なプログラミングへの考慮

設定した指令とアドレスはプログラミング中、維持されます。効率的なプログラミングを行うために次を考慮すべきです。

- ・ 複数のメモリ領域を読み書きする時に指令設定は一度だけ必要です。
- ・ チップ消去後のフラッシュメモリと(EESAVEヒューズがプログラム(0)されている場合を除き)EEPROM全体の内容は\$FFなので、値が\$FFのデータ書き込みを飛ばします。
- ・ アドレス上位バイトはフラッシュメモリで新規256語(ワード)枠、EEPROMで新規256バイト枠の読み書き前に一度だけ必要です。この考慮は識票バイト読み出しにも適用されます。

チップ消去

チップ消去はフラッシュメモリ、EEPROM(注1)、施錠ビットを消去します。施錠ビットはプログラムメモリが完全に消去されてしまうまでリセット(消去)されません。ヒューズビットは変更されません。チップ消去はフラッシュメモリやEEPROMが再書き込みされる前に実行されなければなりません。

注1: EESAVEヒューズがプログラム(0)されていると、EEPROMはチップ消去中、保護されます。

「チップ消去」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$80(1000 0000)に設定します。これはチップ消去指令です。
- ④ XTAL1に正パルスを与えます。これはチップ消去指令を設定します。
- ⑤ $\overline{\text{WR}}$ に負パルスを与えます。これはチップ消去を開始します。RDY/ $\overline{\text{BSY}}$ がLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/ $\overline{\text{BSY}}$ がHigh(1)になるまで待ちます。

フラッシュメモリ書き込み (図71.タイミングを参照)

フラッシュメモリはページで構成されます(107頁の表68参照)。フラッシュメモリを書く時にプログラムデータはページ緩衝部にラッチされます。これは同時に書かれることをプログラムデータの1ページに許します。次の手順は完全なフラッシュメモリの書き込み方法を記述します。

A. 「フラッシュメモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュメモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュメモリ書き込み指令を設定します。

B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選びます。
- ③ DATAにアドレス下位バイト(\$00~\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス下位バイトを設定します。

C. データ下位バイト設定

- ① XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ② DATAにデータ下位バイト(\$00~\$FF)を設定します。
- ③ XTAL1に正パルスを与えます。これはデータ下位バイトを設定します。

D. データ上位バイト設定

- ① BS1をHigh(1)に設定します。これは上位バイトを選びます。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00~\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはデータ上位バイトを設定します。

E. 緩衝部全体が満たされるか、または(必要な)全てのデータが緩衝部内に設定されるまで、B~Dを繰り返し

アドレス内の下位ビットがページ内の語(ワード)位置を指示する一方、上位ビットがフラッシュメモリ内のページをアドレス指定します。これは図70.で図示されます。ページ内の語アドレスに8ビット未満が必要とされる場合(ページ容量<256)、アドレス下位バイトの最上位(側)ビットがページ書き込み実行時のページアドレスに使われることに注意してください。

F. アドレス上位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選びます。
- ③ DATAにアドレス上位バイト(\$00~\$03)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイトを設定します。

G. ページ書き込み

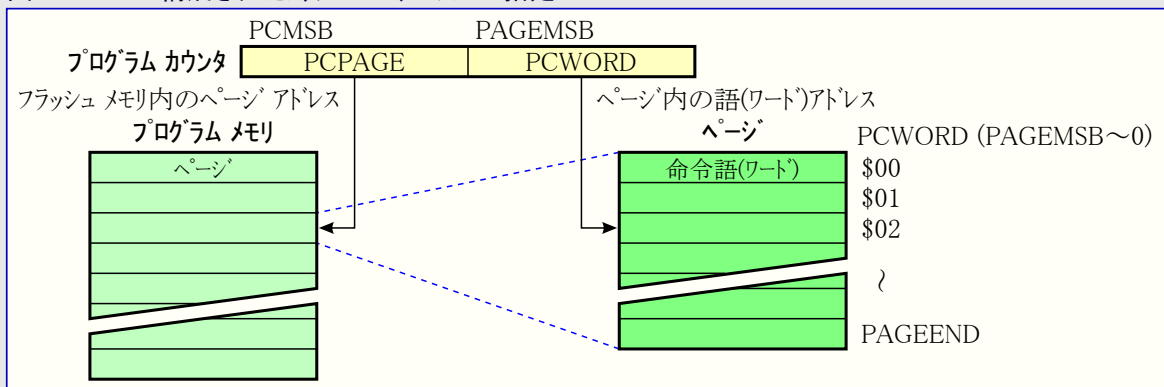
- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/ \overline{BSY} がLow(0)になります。
- ③ RDY/ \overline{BSY} がHigh(1)になるまで待ちます。

H. フラッシュメモリ全部が書かれるか、または(必要な)全データが書かれてしまうまで、B~Gを繰り返し

I. ページ書き込み終了

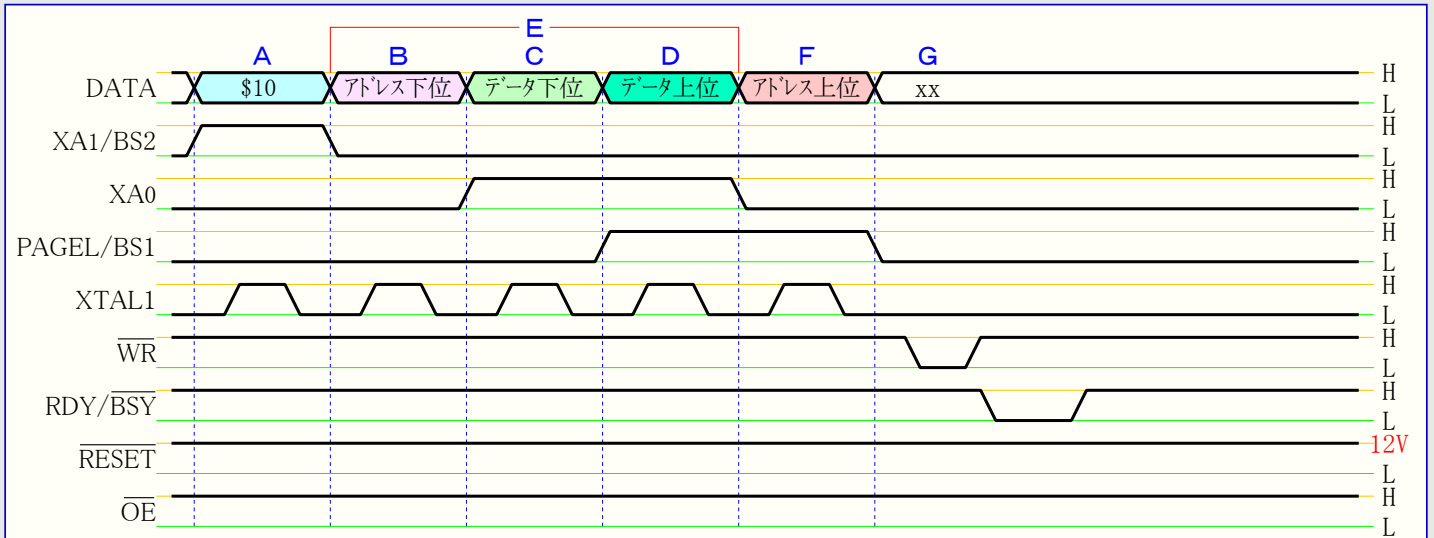
- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図70. ページで構成されたフラッシュメモリのアドレス指定



注: PCPAGEとPCWORDは107頁の表68.で一覧されます。

図71. フラッシュメモリ書き込みタイミング



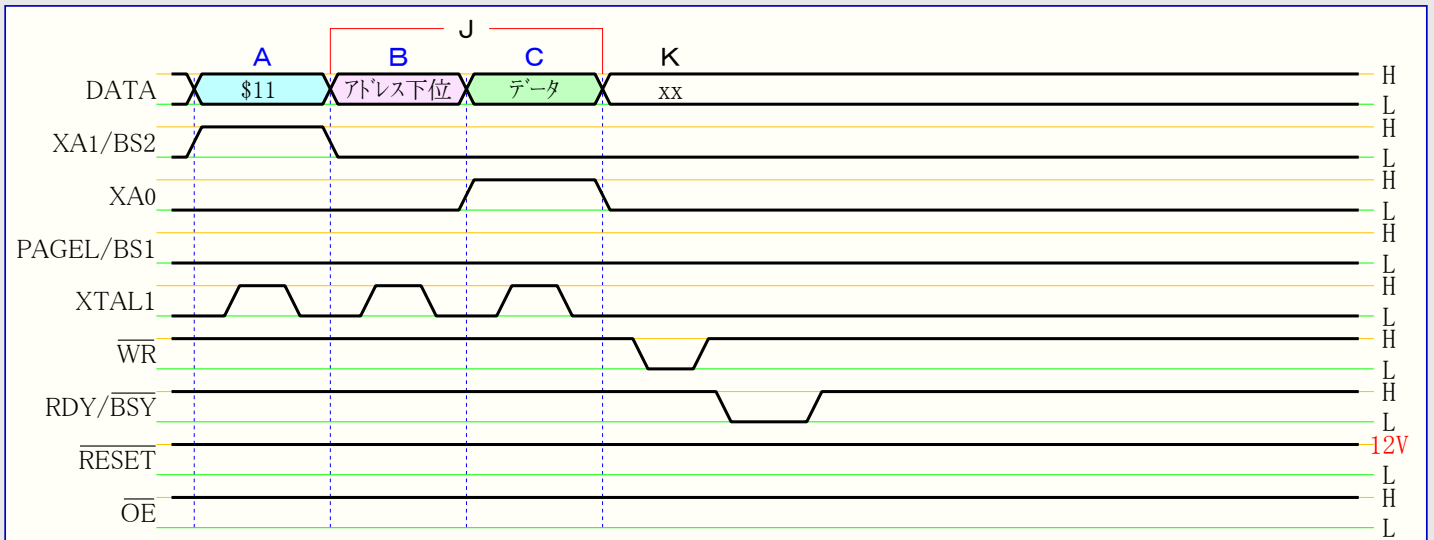
注: xx値は無関係です。A～Gは前記プログラミングを参照してください。

EEPROM書き込み

EEPROMはページで構成されます(107頁の表69参照)。EEPROMを書く時にデータはページ緩衝部にラッチされます。これは同時に書かれることをデータの1ページに許します。EEPROMデータメモリの書き込み方法は次のとおりです。(指令、アドレス、データ設定の詳細については110頁の「フラッシュメモリの書き込み」を参照。図72タイミング参照。)

1. EEPROM書き込み指令\$11(0001 0001)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス下位バイト(\$00～\$7F)を設定します。(「フラッシュメモリ書き込み」のBを参照)
3. データバイト(\$00～\$FF)を設定します。(「フラッシュメモリ書き込み」のCを参照)
- J. 緩衝部全体が満たされるまで2～3を繰り返します。
- K. EEPROMページ書き込み
 - ① BS1をLow(0)に設定します。
 - ② WRに負パルスを与えます。これはEEPROMページ書き込みを開始します。RDY/BSYがLow(0)になります。
 - ③ 次のページを書く前に、RDY/BSYがHigh(1)になるまで待ちます。

図72. EEPROM書き込みタイミング



フラッシュメモリ読み出し

フラッシュメモリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. フラッシュメモリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$03)を設定します。(「フラッシュメモリ書き込み」のFを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. BS1をLow(0)、 \overline{OE} をLow(0)に設定します。フラッシュメモリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュメモリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. \overline{OE} をHigh(1)に設定します。DATAはHi-Zになります。

EEPROM読み出し

データ用EEPROMの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$7F)を設定します。(「フラッシュメモリ書き込み」のBを参照)
3. BS1をLow(0)、 \overline{OE} をLow(0)に設定します。EEPROMのバイトデータが直ぐにDATAで読めます。
4. \overline{OE} をHigh(1)に設定します。DATAはHi-Zになります。

ヒューズビット書き込み (訳注:原書での拡張/上位/下位ヒューズ項を1つに纏めました。)

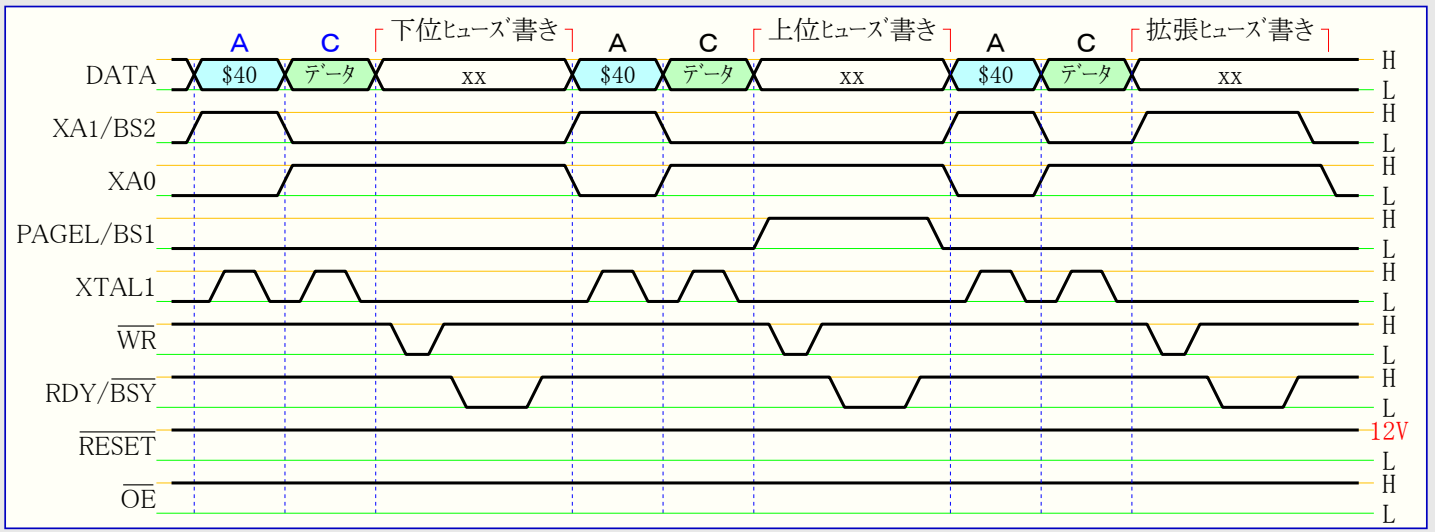
各ヒューズバイトの書き込み方法は次のとおりです。(指令とデータ設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュメモリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4. \overline{WR} に負パルスを与え、RDY/ \overline{BSY} がHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選びます。

表A. ヒューズバイト対応BS1,BS2設定

ヒューズバイト	BS1	BS2
拡張バイト	Low(0)	High(1)
上位バイト	High(1)	Low(0)
下位バイト	Low(0)	Low(0)

図73. ヒューズ書き込みタイミング



施錠ビット書き込み

施錠ビットの書き込み方法は次のとおりです。(指令とデータ設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. データ下位バイト(としてデータ)を設定します。0=プログラム,1=無変化です。LB保護種別3が設定(LB1とLB2がプログラム(0)されると、どの外部的なプログラミング動作種別によっても施錠ビットはプログラミングできません。(「フラッシュメモリの書き込み」のCを参照)
3. \overline{WR} に負パルスを与え、RDY/BSYがHighになるまで待ちます。

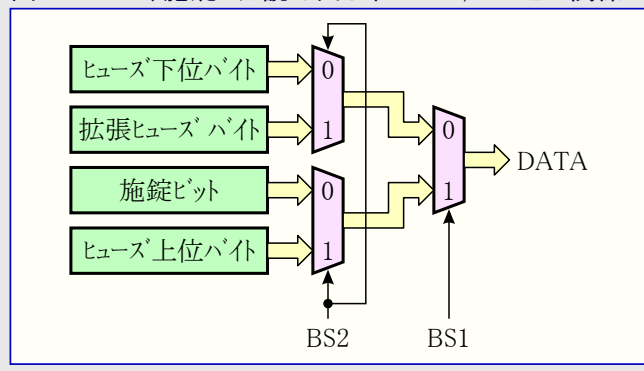
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

ヒューズビットと施錠ビットの読み出し

ヒューズビットと施錠ビットの読み出し方法は次のとおりです。(指令設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズビットと施錠ビットの読み出し指令\$04(0000 0100)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. BS1とBS2をLow(0)、 \overline{OE} をLow(0)に設定します。ヒューズ下位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
3. BS1とBS2をHigh(1)、 \overline{OE} をLow(0)に設定します。ヒューズ上位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
4. BS1をLow(0)、BS2をHigh(1)、 \overline{OE} をLow(0)に設定します。拡張ヒューズビットの状態が直ぐにDATAで読めます。(0=プログラム)
5. BS1をhigh(1)、BS2をLow(0)、 \overline{OE} をLow(0)に設定します。施錠ビットの状態が直ぐにDATAで読めます。(0=プログラム)
6. \overline{OE} をHigh(1)に設定します。DATAはHi-Zになります。

図74. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



識票バイト読み出し

識票バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をLow(0)、 \overline{OE} をLow(0)に設定します。選んだ識票バイトが直ぐにDATAで読めます。
4. \overline{OE} をHigh(1)に設定します。DATAはHi-Zになります。

校正バイト読み出し

校正バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については110頁の「フラッシュメモリの書き込み」を参照)

1. 校正バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイトに(\$00~\$01)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をHigh(1)、 \overline{OE} をLow(0)に設定します。校正バイトが直ぐにDATAで読めます。
4. \overline{OE} をHigh(1)に設定します。DATAはHi-Zになります。

並列プログラミング特性

図75. 並列プログラミング タイミング (一般的な必要条件)

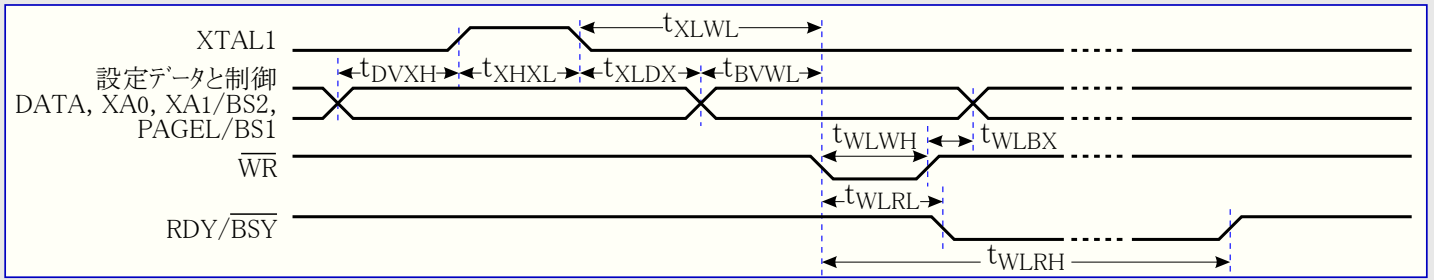
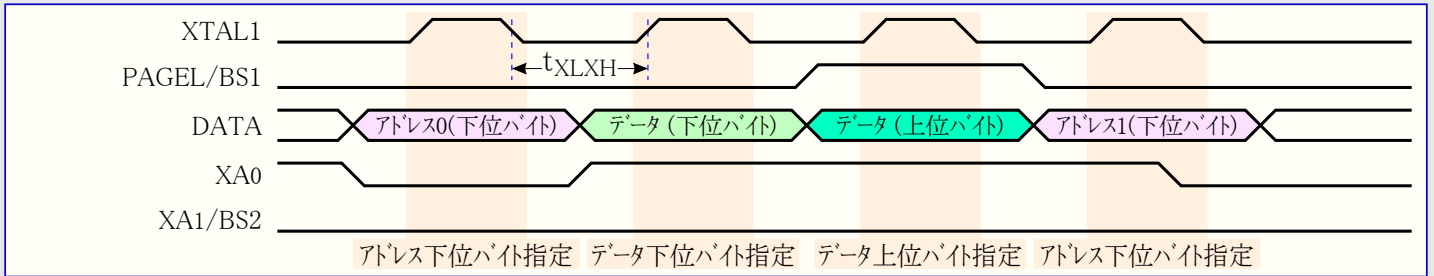
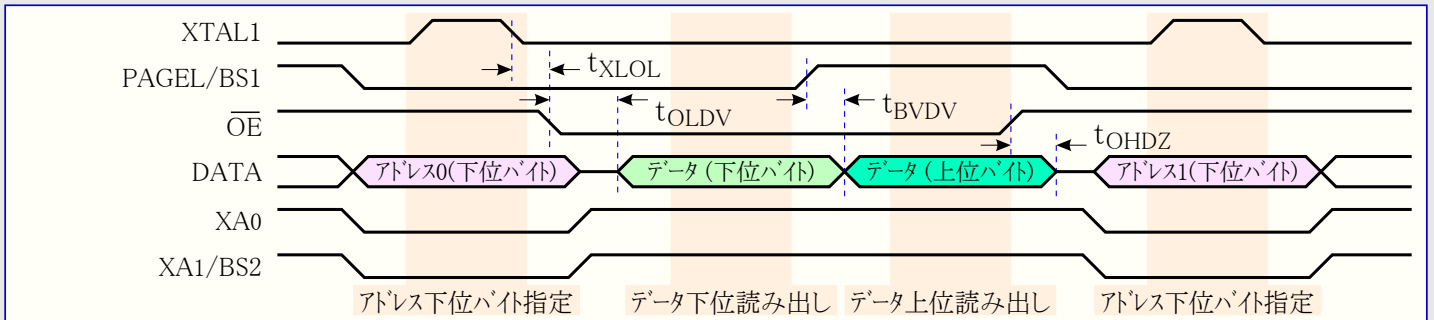


図76. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図75.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHXL} 、 t_{XLDX})は設定操作にも適用されます。

図77. 並列プログラミング タイミング (同一ページ読み出しでの必要条件)



注: 図75.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHXL} 、 t_{XLDX})は読み出し操作にも適用されます。

表75. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
V _{PP}	プログラミング許可電圧	11.5		12.5	V
I _{PP}	プログラミング許可電流			250	μA
t _{DVXH}	XTAL1 ↑ に対するデータと制御の準備時間	67			ns
t _{XLXH}	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
t _{XHXL}	XTAL1 Highパルス幅	150			
t _{XLDX}	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
t _{XLWL}	XTAL1パルス ↓ 後のWR ↓ 待機時間	0			
t _{WLBX}	WRパルス ↓ 後のBS1,BS2保持時間	67			
t _{BVWL}	WRパルス ↓ に対するBS1準備時間	67			
t _{WLWH}	WR Lowパルス幅	150			μs
t _{WLRH}	書き込み時間 (WR ↓ からRDY/BSY ↑)	0		1	
t _{WLRH}	書き込み時間 (WR ↓ からRDY/BSY ↑)	3.7		4.5	ms
t _{WLRH_CE}	チップ消去時間 (WR ↓ からRDY/BSY ↑)	7.5		9	
t _{XLCL}	XTAL1パルス ↓ 後のOE ↓ 待機時間	0			ns
t _{BVDV}	BS1有効からのDATA遅延時間	0		250	
t _{OLDV}	OE ↓ 後のDATA出力遅延時間			250	
t _{OHDZ}	OE ↑ 後のDATA Hi-Z遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され得るのに先立って**プログラミング許可命令**が初めに実行されることを必要とします。**注意**、表74.でSPIプログラミング用のピン配置が一覧されます。全てのデバイスが内部SPIインターフェースに対する専用SPIピンを使うとは限りません。

直列プログラミング用ピン配置

表74. 直列プログラミング用ピン配置

信号名	ピン名	入出力	機能
MOSI	PB5	入力	直列データ入力
MISO	PB6	出力	直列データ出力
SCK	PB7	入力	直列クロック

EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(直列プログラミングのみ)、**チップ消去命令**を初めに実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全てのメモリ位置の内容を\$FFにします。

CKSELヒューズによる有効なクロックが与えられなければなりません。直列クロック(SCK)入力のLowとHighの区間の最小値は次のように定義されます。

$f_{CK} < 12\text{MHz}$: Low区間	> 2 CPUクロック周期	$f_{CK} < 12\text{MHz}$: High区間	> 2 CPUクロック周期
$f_{CK} \geq 12\text{MHz}$: Low区間	> 3 CPUクロック周期	$f_{CK} \geq 12\text{MHz}$: High区間	> 3 CPUクロック周期

直列プログラミング手順

ATtiny2313に直列データを書く時にデータはSCKの上昇端で行われ、ATtiny2313から読む時にデータはSCKの下降端で行われます。タイミングの詳細については図79., 図80.と表78.をご覧ください。

直列プログラミング動作でのATtiny2313のプログラミングと照合は次の手順が推奨されます(4バイト命令形式は表77.を参照)。

1. 電源投入手順:

RESETとSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加します。いくつかのシステムで電源投入中、SCKがLowに保持されるのを書き込み器が保証できません。この場合、SCKがLow(0)に設定されてしまった後、RESETは最低2 CPUクロック周期幅の正パルスを与えられなければなりません。

2. 最低20ms待ち、MOSIピンに**プログラミング許可命令**を送ることによって直列プログラミングを許可してください。

3. 通信が同期を外していると、直列プログラミング命令は動作しません。同期していると、プログラミング許可命令の第3バイト送出時に第2バイト(\$53)が戻ります。この戻りが正しいかどうかによらず、命令の4バイト全てが送信されなければなりません。\$53が戻らない場合、RESETに正パルスを与え、新規プログラミング許可命令を行ってください。

4. フラッシュメモリは1ページ単位で書かれます。ページ容量は107頁の表68.で得られます。このメモリページはページ設定命令と共にアドレスの下位4+1ビットとデータを供給することによって1バイトずつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。プログラムメモリのページはアドレスの上位6ビットを含む**ページ書き込み命令**の設定によって(フラッシュメモリに)格納されます。ポーリング(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低tWD_FLASH(表76.参照)待たなければなりません。

注: 何れの(フラッシュ、EEPROM、施錠ビット、ヒューズ)書き込み操作が完了される前にポーリング(読み)以外の命令が加えられると、不正な書き込みに終わるかもしれません(訳注:共通性のため原書の該当行に代わり本行追加)。

5. EEPROMはページ単位かバイト単位のどちらかでプログラミングできます。

バイト単位: EEPROMは適切な**EEPROM書き込み命令**と共にアドレスとデータを供給することによって1バイト単位で書かれます。EEPROMのメモリ位置は新規データが書かれるのに先立って始めて自動的に消去されます。ポーリング(BSY/RDY)が使われない場合、次のバイトを行う前に最低tWD_EEPROM(表76.参照)待たなければなりません。

ページ単位: EEPROMの1ページは**EEPROMページ設定命令**と共にアドレスの下位2ビットとデータを供給することによって1バイトずつ設定されます。EEPROMページはアドレスの上位5ビットを含む**EEPROMページ書き込み命令**によって(EEPROMに)格納されます。EEPROMページアクセス使用時、EEPROMページ設定命令で設定したバイト位置だけが変更されます。残りの位置は無変化で留まります。ポーリング(BSY/RDY)が使われない場合、次のページを行う前に最低tWD_EEPROM(表76.参照)待たなければなりません。チップ消去されたデバイスでの\$FFデータは書かれる必要がありません。

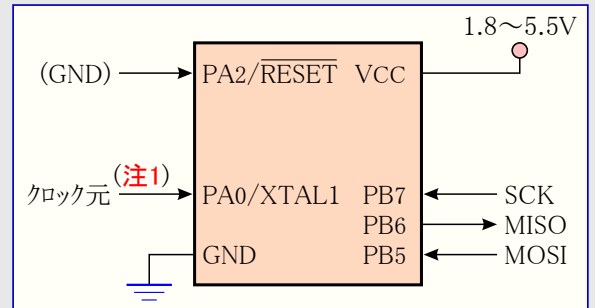
6. どのメモリ位置も選んだアドレスの内容を直列出力MISOに読み戻す**読み出し命令**を使うことによって照合できます。

7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。

8. 電源OFF手順 (必要とされるならば)

- RESETをHigh(1)に設定します。
- VCC電源をOFFにします。

図78. 直列プログラミング構成図



注1: デバイスが内蔵発振器で動作する場合、XTAL1ピンにクロック元を接続する必要はありません。

表76. ヒューズ、フラッシュ、EEPROM次位置書き込み前の待機時間

シンボル	最低待機時間	備考
tWD_FUSE	4.5ms	ヒューズ書き込み
tWD_FLASH	4.5ms	フラッシュメモリ書き込み
tWD_EEPROM	4.0ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

図79. 直列プログラミングバイト通信波形

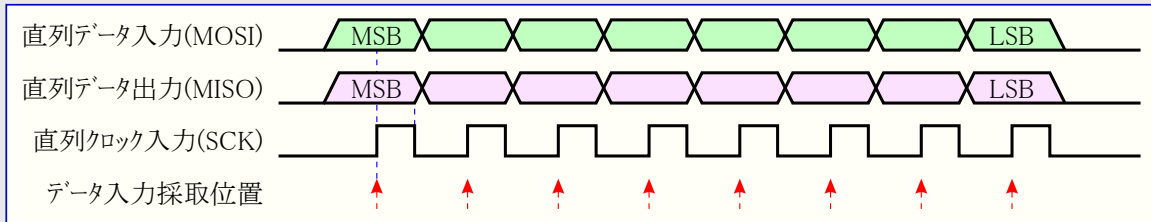


表77. 直列プログラミング命令一式

命令	命令形式				動作
	第1バイト	第2バイト	第3バイト	第4バイト	
プログラミング許可	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	RESET=Low後のプログラミングを許可します。
チップ消去	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	フラッシュメモリ,EEPROM,施錠ビットを消去します。
フラッシュメモリ読み出し	0010 P000	0000 00HH	LLLL LLLL	RRRR RRRR	アドレスH:LのP(H/L)バイトを読み出します。
フラッシュページ設定	0100 P000	000x xxxx	xxxx LLLL	WWWW WWWW	緩衝部アドレスLのP(H/L)バイトに書き込みます。
フラッシュページ書き込み	0100 1100	0000 00HH	LLLL xxxx	xxxx xxxx	アドレスH:Lのページに書き込みます。
EEPROM読み出し	1010 0000	000x xxxx	xLLL LLLL	RRRR RRRR	アドレスLのバイトを読み出します。
EEPROMバイト書き込み	1100 0000	000x xxxx	xLLL LLLL	WWWW WWWW	アドレスLのバイトに書き込みます。
EEPROMページ設定	1100 0001	0000 0000	0000 00LL	WWWW WWWW	緩衝部アドレスLのバイトに書き込みます。
EEPROMページ書き込み	1100 0010	00xx xxxx	xLLL LL00	xxxx xxxx	アドレスLのページに書き込みます。
施錠ビット読み出し	0101 1000	0000 0000	xxxx xxxx	xxxx xxRR	施錠ビットを読み出します。
施錠ビット書き込み	1010 1100	111x xxxx	xxxx xxxx	1111 11WW	(106頁の表63.参照)に書き込みます。
ヒューズ下位読み出し	0101 0000	0000 0000	xxxx xxxx	RRRR RRRR	ヒューズ下位ビットを読み出します。
ヒューズ下位書き込み	1010 1100	1010 0000	xxxx xxxx	WWWW WWWW	(107頁の表67.参照)に書き込みます。
ヒューズ上位読み出し	0101 1000	0000 1000	xxxx xxxx	RRxR RRRR	ヒューズ上位ビットを読み出します。
ヒューズ上位書き込み	1010 1100	1010 1000	xxxx xxxx	WWOW WWWW	(106頁の表66.参照)に書き込みます。
拡張ヒューズ読み出し	0101 0000	0000 1000	xxxx xxxx	xxxx xxxR	拡張ヒューズビットを読み出します。
拡張ヒューズ書き込み	1010 1100	1010 0100	xxxx xxxx	xxxx xxxW	(106頁の表65.参照)に書き込みます。
識票バイト読み出し	0011 0000	000x xxxx	xxxx xxLL	RRRR RRRR	アドレスLの識票バイトを読み出します。
校正バイト読み出し	0011 1000	000x xxxx	0000 000L	RRRR RRRR	アドレスLの校正バイトを読み出します。
多忙/準備可検査	1111 0000	0000 0000	xxxx xxxx	xxxx xxxR	R=1で多忙、他の操作前に0まで待機します。

H = アドレス上位バイトのビット L = アドレス下位バイトのビット P = 0=下位バイト、1=上位バイト
R = 読み出しデータ (MCU出力) W = 書き込みデータ (MCU入力) x = 0または1 (無意味/不定)

直列プログラミング特性

図80. 直列プログラミング タイミング

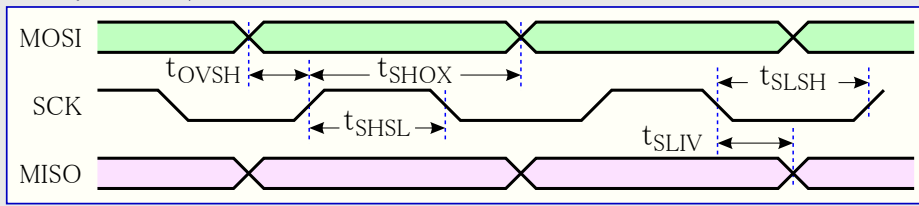


表78. 直列プログラミング特性 (特記条件を除いて、TA=-40°C~85°C, VCC=2.7~5.5V)

シンボル	項目	最小	代表	最大	単位
1/tCLCL	発振器周波数	2.7~5.5V	0	10	MHz
		4.5~5.5V	0	20	
tCLCL	発振器周期	2.7~5.5V	100		ns
		4.5~5.5V	50		
tSHSL	SCKパルスHレベル幅 (注1)	2tCLCL			
tSLSH	SCKパルスLレベル幅 (注1)	2tCLCL			
tOVSH	SCK↑に対するMOSI準備時間	tCLCL			
tSHOX	SCK↑に対するMOSI保持時間	2tCLCL			
tSLIV	SCK↓に対するMISO出力遅延時間			100	

注1: $f_{CK} < 12\text{MHz}$ 時 $2t_{CLCL}$ 、 $f_{CK} \geq 12\text{MHz}$ 時 $3t_{CLCL}$ 。

電気的特性

絶対最大定格 (警告)

動作温度	-55°C ~ +125°C
保存温度	-65°C ~ +150°C
RESETを除くピン許容電圧	-0.5V ~ VCC+0.5V
RESETピン許容電圧	-0.5V ~ +13.0V
最大動作電圧	6.0V
入出力ピン出力電流	40.0mA
消費電流	200.0mA

(警告)

絶対最大定格を超える負担はデバイスに定常的な損傷を与えます。絶対最大定格は負担の定格を示すためだけのもので、この値または、この仕様書の動作特性で示された値を超える条件で動作することを示すものではありません。長時間の最大定格での使用はデバイスの信頼性を損なう場合があります。

DC特性

TA=-40°C~85°C, VCC=1.8V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表 (注1)	最大	単位
VIL	Lowレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	-0.5		0.2VCC (注2)	V
		VCC=2.4~5.5V	-0.5		0.3VCC (注2)	
VIL1	Lowレベル入力電圧 (XTAL1)	VCC=1.8~5.5V	-0.5		0.1VCC (注2)	
VIL2	Lowレベル入力電圧 (RESET)	VCC=1.8~5.5V	-0.5		0.2VCC (注2)	
VIL3	Lowレベル入力電圧 (I/OとしてのRESET)	VCC=1.8~2.4V	-0.5		0.2VCC (注2)	
		VCC=2.4~5.5V	-0.5		0.3VCC (注2)	
VIH	Highレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	0.7VCC (注3)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注3)		VCC+0.5	
VIH1	Highレベル入力電圧 (XTAL1)	VCC=1.8~2.4V	0.8VCC (注3)		VCC+0.5	
		VCC=2.4~5.5V	0.7VCC (注3)		VCC+0.5	
VIH2	Highレベル入力電圧 (RESET)	VCC=1.8~5.5V	0.9VCC (注3)		VCC+0.5	
VIH3	Highレベル入力電圧 (I/OとしてのRESET)	VCC=1.8~2.4V	0.7VCC (注3)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注3)		VCC+0.5	
VOL	Lowレベル出力電圧 (注4)	I/OとしてのRESET以外 IOL=20mA, VCC=5V			0.7	
		IOL=10mA, VCC=3V			0.5	
VOL3	RESET=I/O	TBD			TBD	
VOH	Highレベル出力電圧 (注5)	I/OとしてのRESET以外 IOH=-20mA, VCC=5V	4.2			
		IOH=-10mA, VCC=3V	2.5			
VOH3	RESET=I/O	TBD	TBD			
IIL	I/OピンLowレベル入力漏れ電流	VCC=5.5V			1	μA
IIH	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			1	
RRST	RESETピンプルアップ抵抗		30		60	kΩ
RPU	I/Oピンプルアップ抵抗		20		50	
ICC	活動動作消費電流	VCC=2V, 1MHz			0.35	mA
		VCC=3V, 4MHz			2	
		VCC=5V, 8MHz			6	
	アイドル動作消費電流	VCC=2V, 1MHz		0.08	0.2	
		VCC=3V, 4MHz		0.41	1	
		VCC=5V, 8MHz		1.6	3	
パワーダウン動作消費電流	VCC=3V, WDT有効		<3	6	μA	
	VCC=3V, WDT禁止		<0.5	2		
VACIO	アナログ比較器入力変位(オフセット)電圧	VCC=5V, Vin=VCC/2		<10	40	mV
IACLK	アナログ比較器入力漏れ電流		-50		50	
tACPD	アナログ比較器伝播遅延時間	VCC=2.7V		750		ns
		VCC=5.0V		500		

(注1)~(注5)は次頁へ

注1: 25°Cでの代表値です。

注2: Lowレベルの認識が保証される最高電圧です。

注3: Highレベルの認識が保証される最低電圧です。

注4: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=5Vで10mA、VCC=3Vで5mA)よりも多くの吸い込み電流を流すことができますが、次の条件を厳守してください。

1. 全ポートのIOLの合計が60mAを超えるべきではありません。

IOLが検査条件を超える場合、VOLも仕様書での値を超えます。表の検査条件よりも大きな吸い込み電流を流すことは保証されません。

注5: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=5Vで10mA、VCC=3Vで5mA)よりも多くの吐き出し電流を流すことができますが、次の条件を厳守してください。

1. 全ポートのIOHの合計が60mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件よりも大きな吐き出し電流を流すことは保証されません。

外部クロック特性

図81. 外部クロック駆動波形

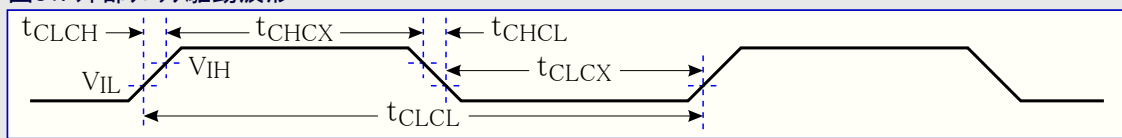


表79. 外部クロック特性

シンボル	項目	VCC=1.8~5.5V		VCC=2.7~5.5V		VCC=4.5~5.5V		単位
		最小	最大	最小	最大	最小	最大	
$1/t_{CLCL}$	クロック周波数	0	4	0	10	0	20	MHz
t_{CLCL}	クロック周期	250		100		50		ns
t_{CHCX}	Highレベル時間	100		40		20		
t_{CLCX}	Lowレベル時間	100		40		20		
t_{CLCH}	上昇時間		2.0		1.6		0.5	μs
t_{CHCL}	下降時間		2.0		1.6		0.5	
Δt_{CLCL}	隣接クロック周期間の変化率		2		2		2	%

最高速と動作電圧

最高周波数は動作電圧に依存します。図82と図83.で示されるように最高周波数対動作電圧曲線は1.8~2.7Vと2.7~4.5V間で直線です。

図82. ATtiny2313Vの最高周波数対VCC

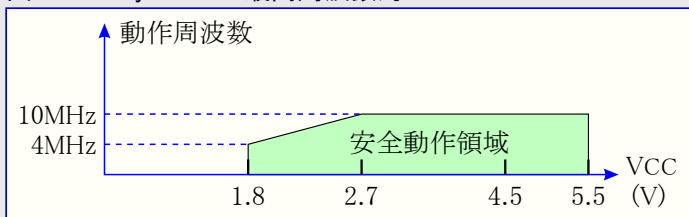
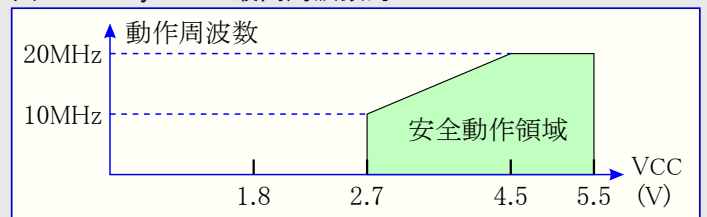


図83. ATtiny2313の最高周波数対VCC



システムとリセットの特性

表80. リセット、低電圧検出(BOD)、内部基準電圧の特性

シンボル	項目	条件	最小	代表	最大	単位
VPOT	上昇時電源ONリセット閾値電圧	TA=-40~85°C	0.7	1.0	1.4	V
	下降時電源ONリセット閾値電圧 (注)		0.05	0.9	1.3	
VPONSR	電源ON電圧傾斜率		0.01		4.5	V/ms
VRST	RESETピン閾値電圧		0.2VCC		0.9VCC	V
tRST	リセットパルス幅		2.5			μs
VHYST	低電圧検出ヒステリシス電圧			50		mV
tBOD	最小低電圧検出時間			2		μs
VBG	基準電圧	VCC=2.7V TA=25°C	1.0	1.1	1.2	V
tBG	起動時間			40	70	μs
IBG	消費電流				10	μA

注: 供給電圧がこの電圧以下にならないと、上昇時の電源ONリセットは動作しません。

代表特性

以下の図は代表的な特性を示します。これらの図は製造中に検査されていません。全ての消費電流測定は全I/Oピンを入力として設定した内部プルアップ許可で行われています。電源幅振幅の方形波発振器がクロック源として使われています。

パワーダウン動作での消費電力はクロック選択と無関係です。

消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。支配的な要素は動作電圧と動作周波数です。

容量性負荷のピンの引き込み電流は(1つのピンに対して) $CL(\text{負荷容量}) \times VCC(\text{動作電圧}) \times f(\text{I/Oピンの平均切り替え周波数})$ として推測できます。

デバイスは検査範囲よりも高い周波数特性を示します。デバイスは注文番号が示す周波数よりも高い周波数での機能特性を保証されません。

ウォッチドッグタイマ許可のパワーダウン動作での消費電流とウォッチドッグタイマ禁止のパワーダウン動作での消費電流間の違いは、ウォッチドッグタイマによって引き込んだ(消費した)差電流を表します。

活動動作消費電流

図84. 活動動作消費電流 対 周波数 (100kHz~1MHz)

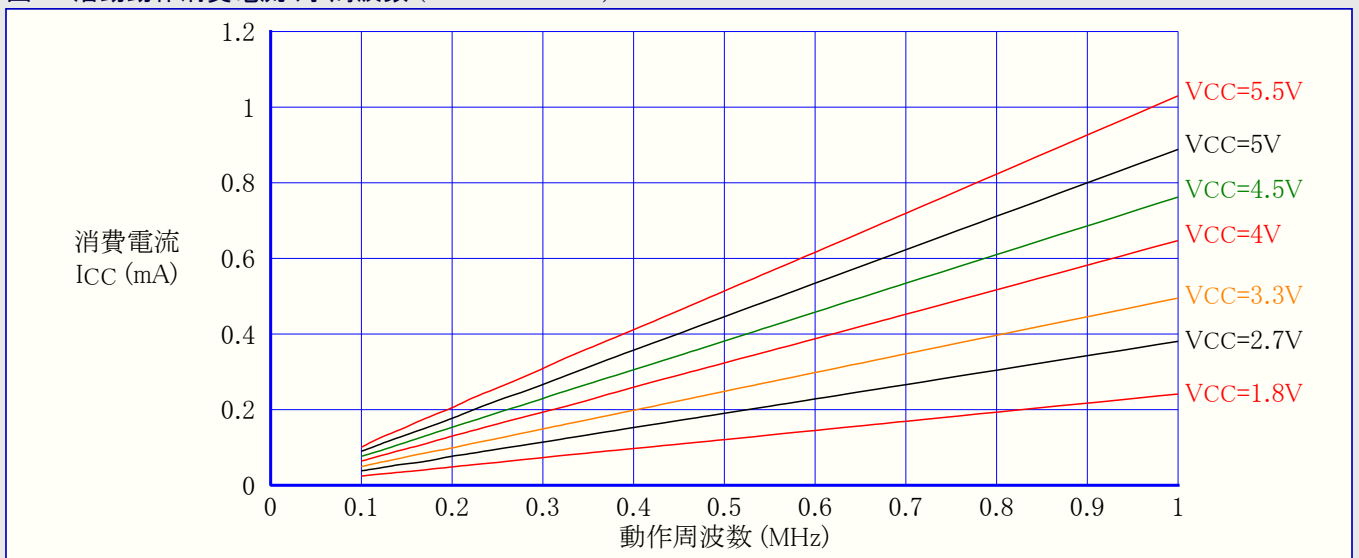


図85. 活動動作消費電流 対 周波数 (1MHz~20MHz)

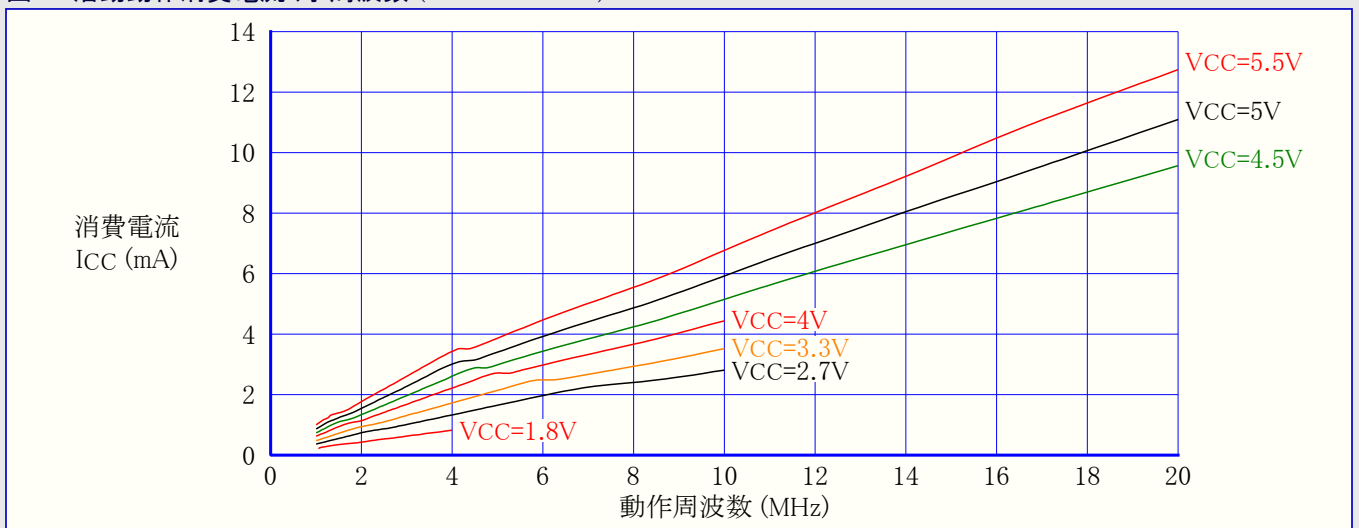


図86. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

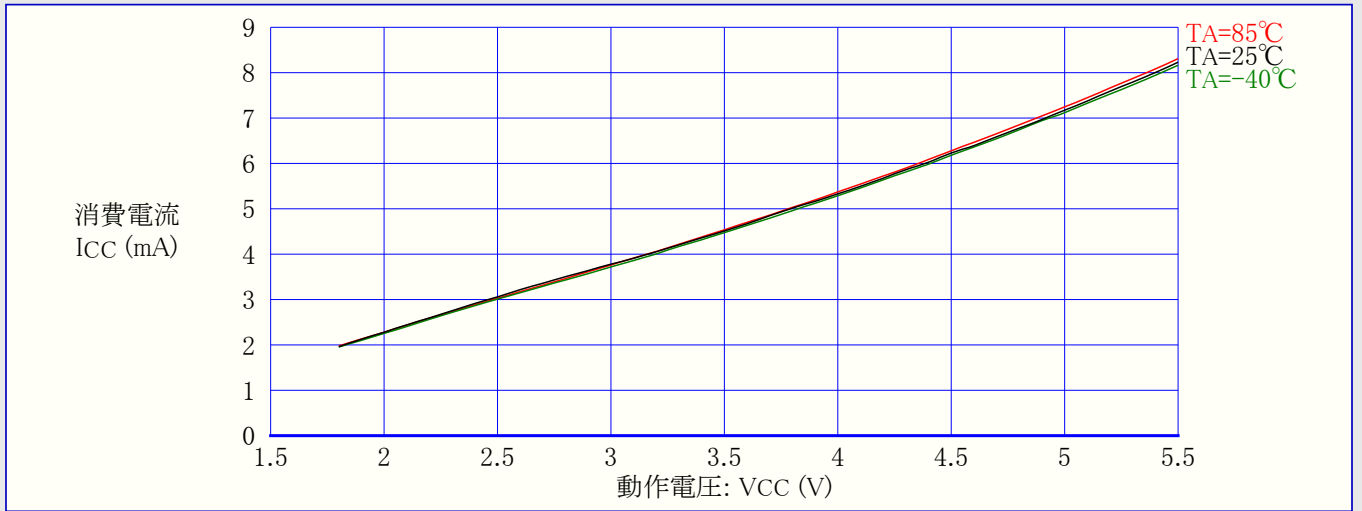


図87. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,4MHz)

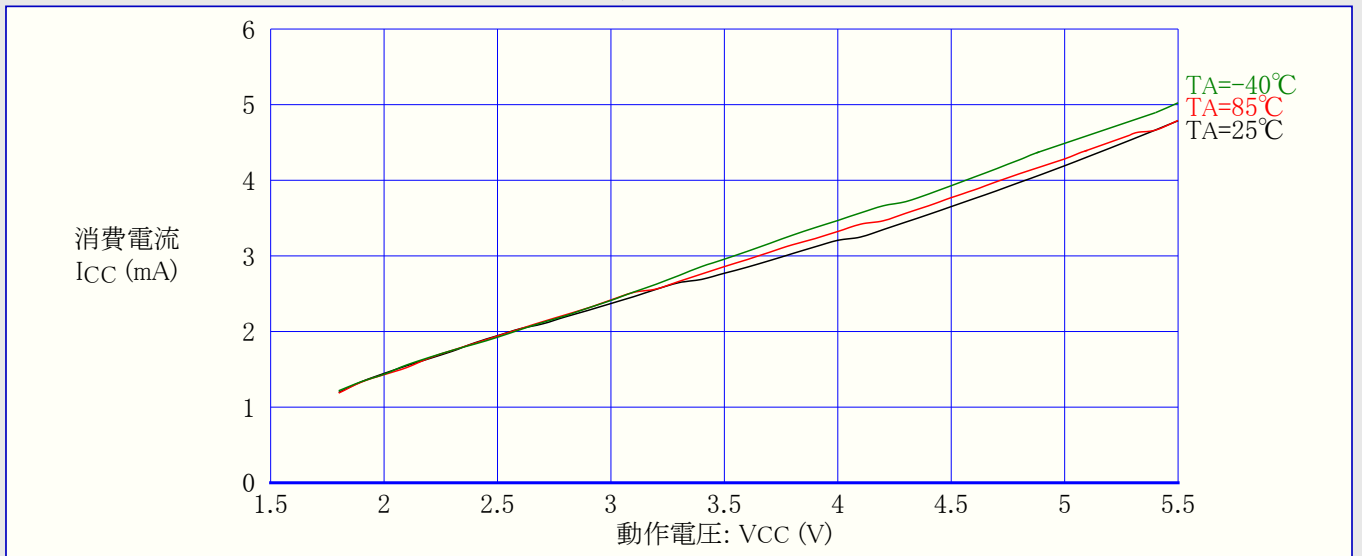


図88. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)

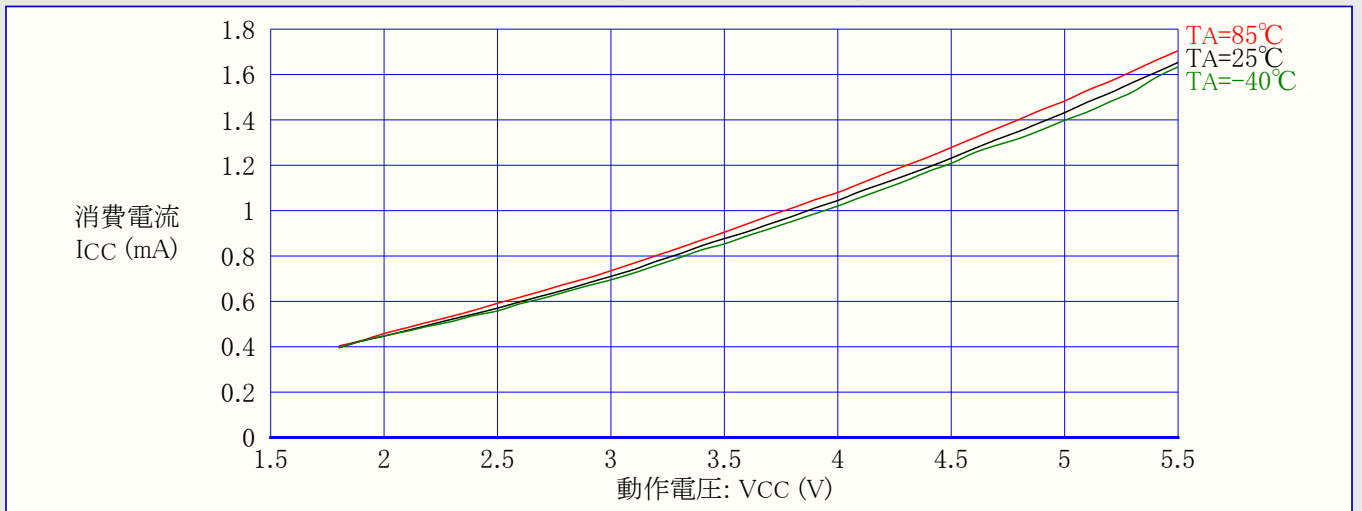


図89. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),500kHz)

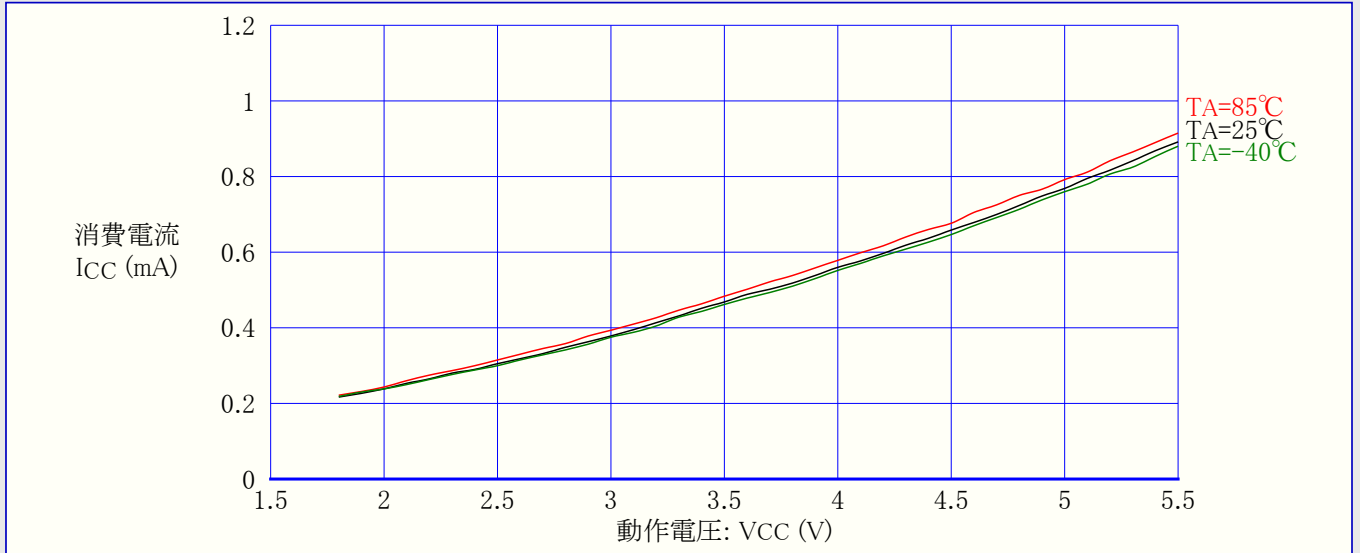
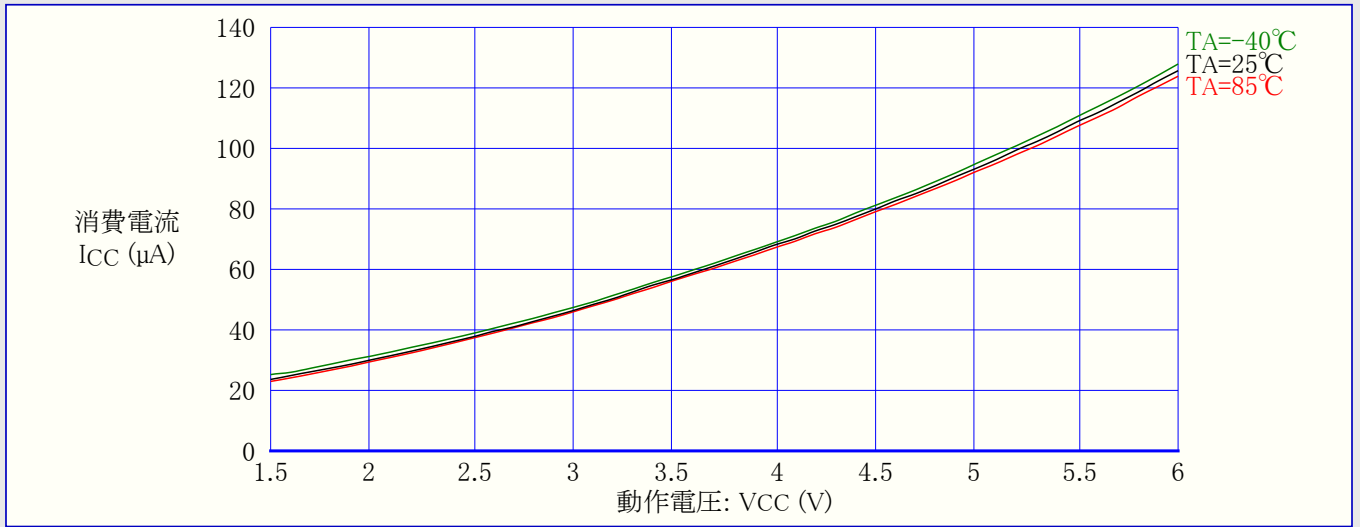


図90. 活動動作消費電流 対 動作電圧 (内蔵WDT発振器,128kHz)



アイドル動作消費電流

図91. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

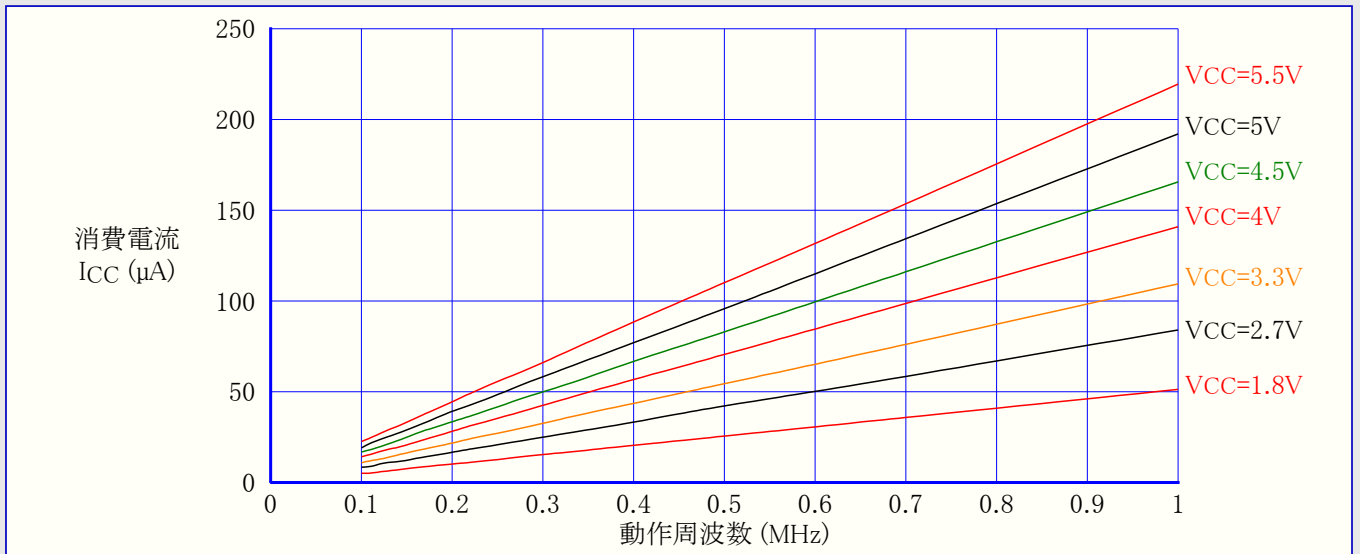


図92. アトル動作消費電流 対 周波数 (1MHz~20MHz)

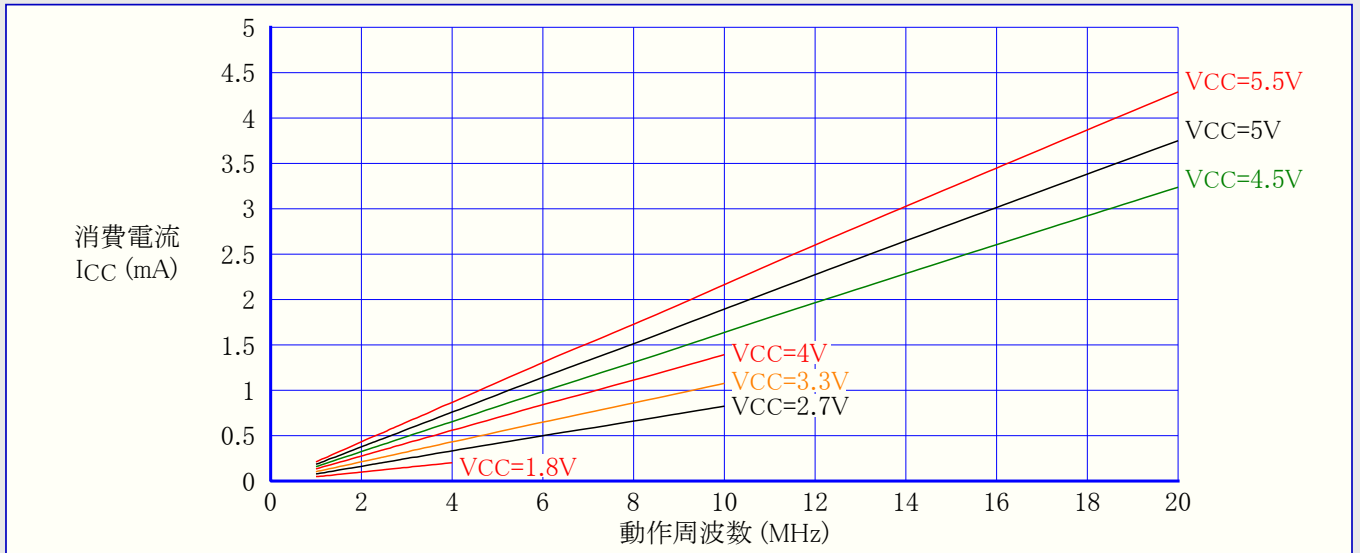


図93. アトル動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

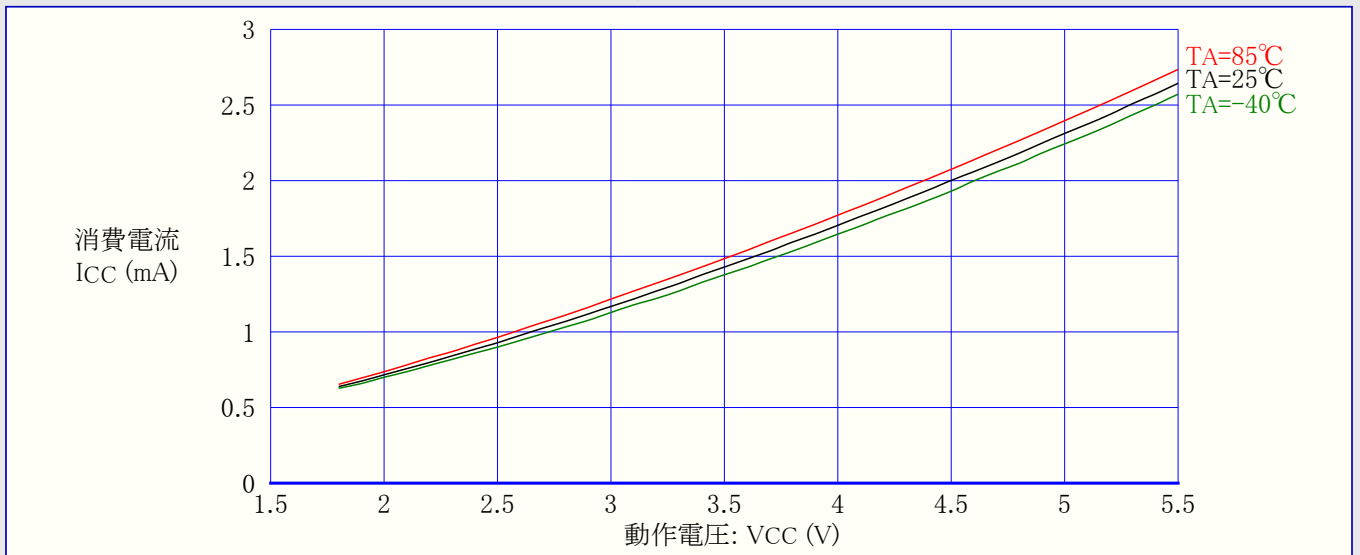


図94. アトル動作消費電流 対 動作電圧 (内蔵RC発振器,4MHz)

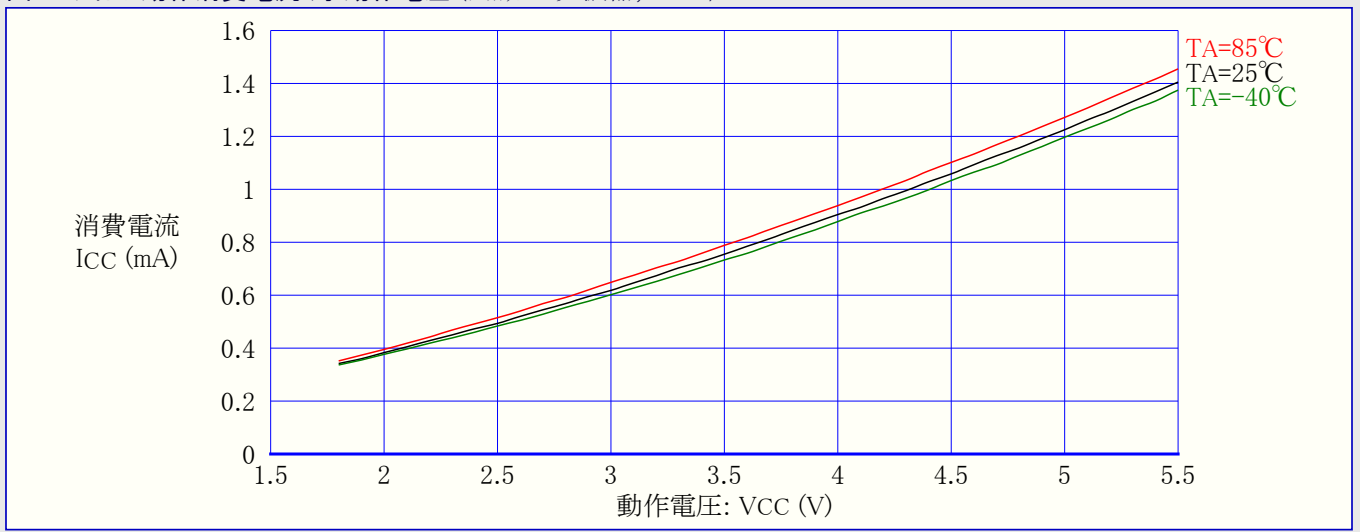


図95. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)

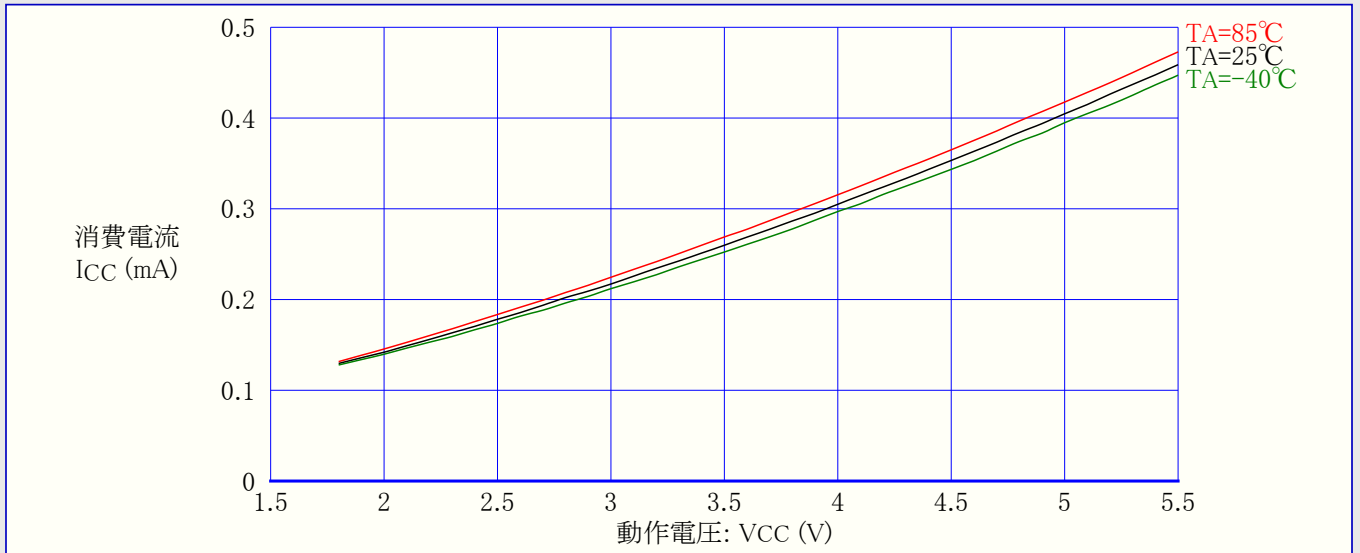


図96. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),500kHz)

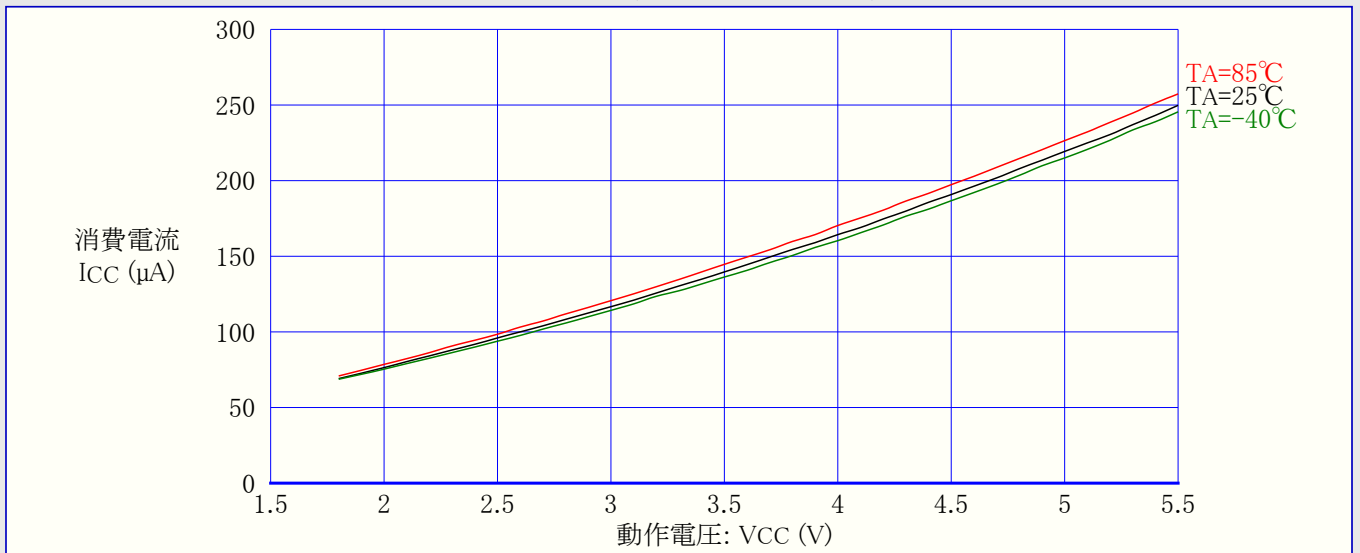
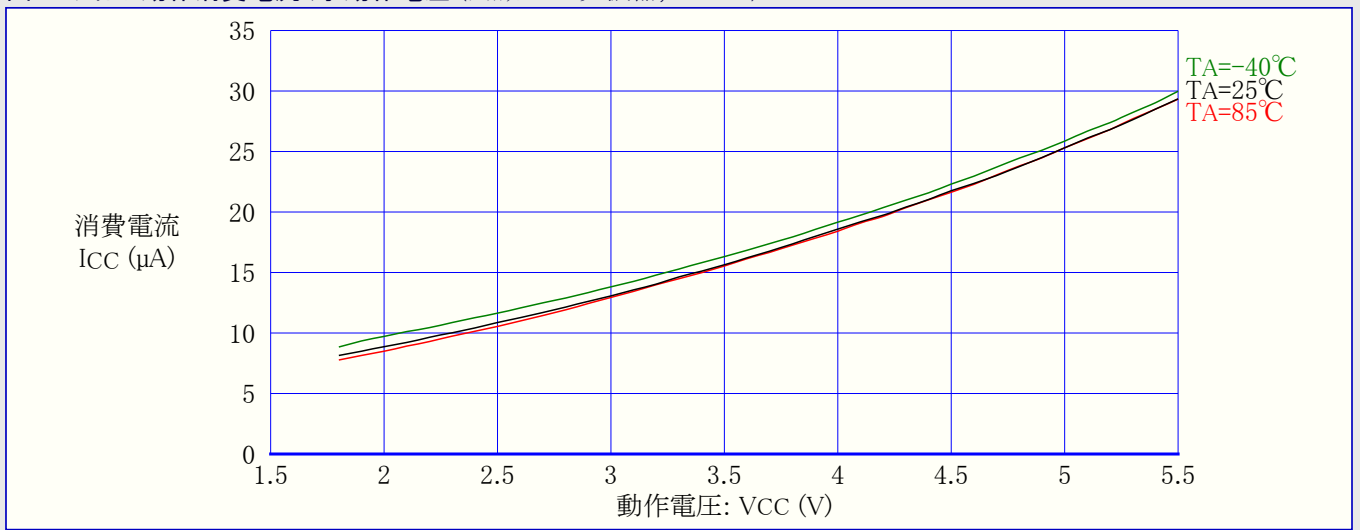


図97. アイドル動作消費電流 対 動作電圧 (内蔵WDT発振器,128kHz)



パワーダウン動作消費電流

図98. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイム禁止)

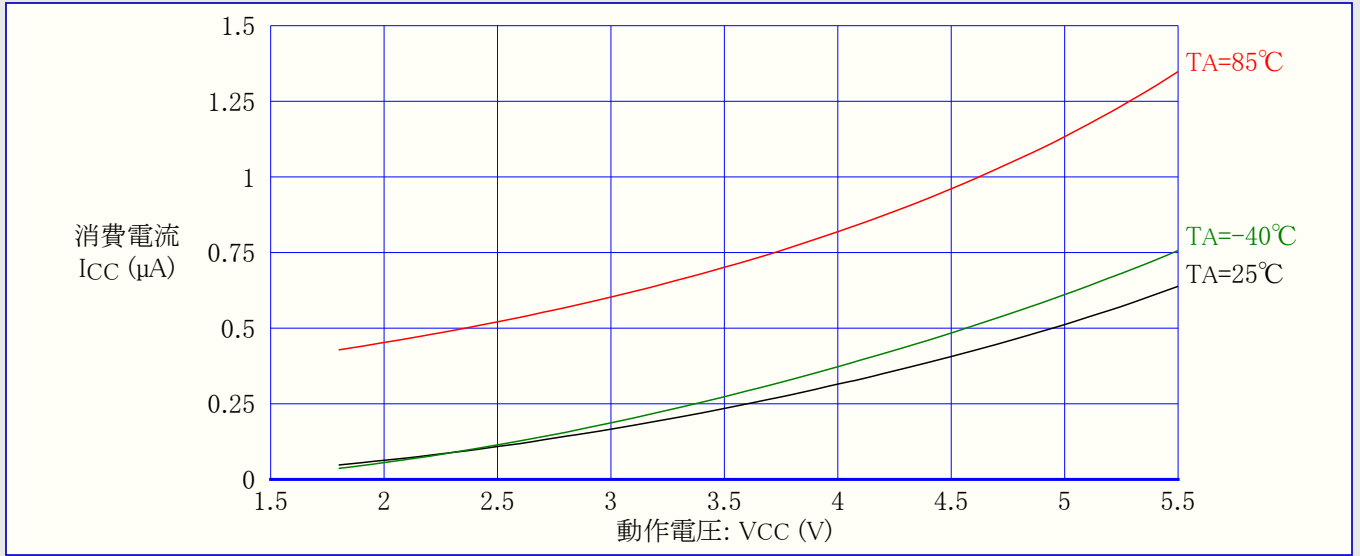
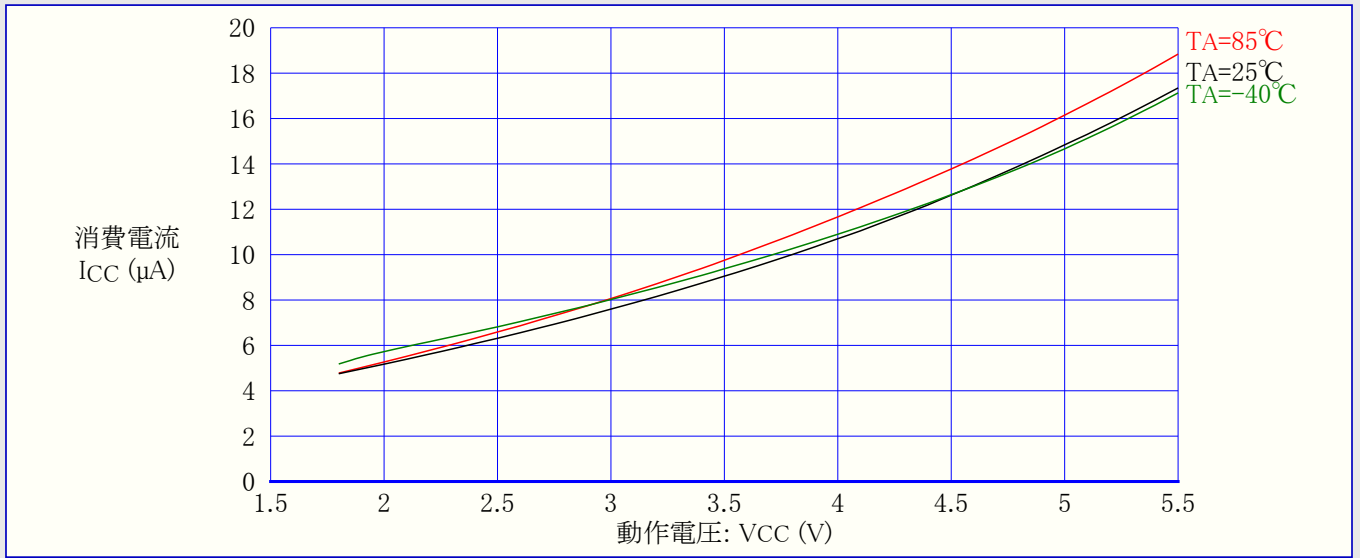
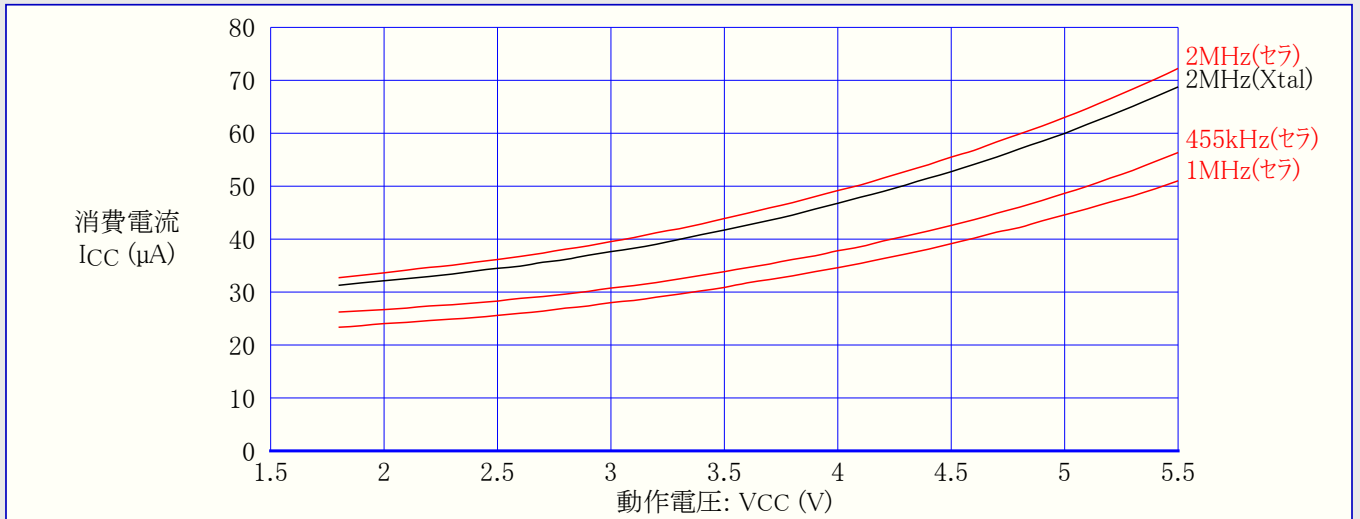


図99. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイム許可)



スタンバイ動作消費電流

図100. スタンバイ動作消費電流 対 動作電圧



注: セラはセラミック振動子

ピンプルアップ

図101. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

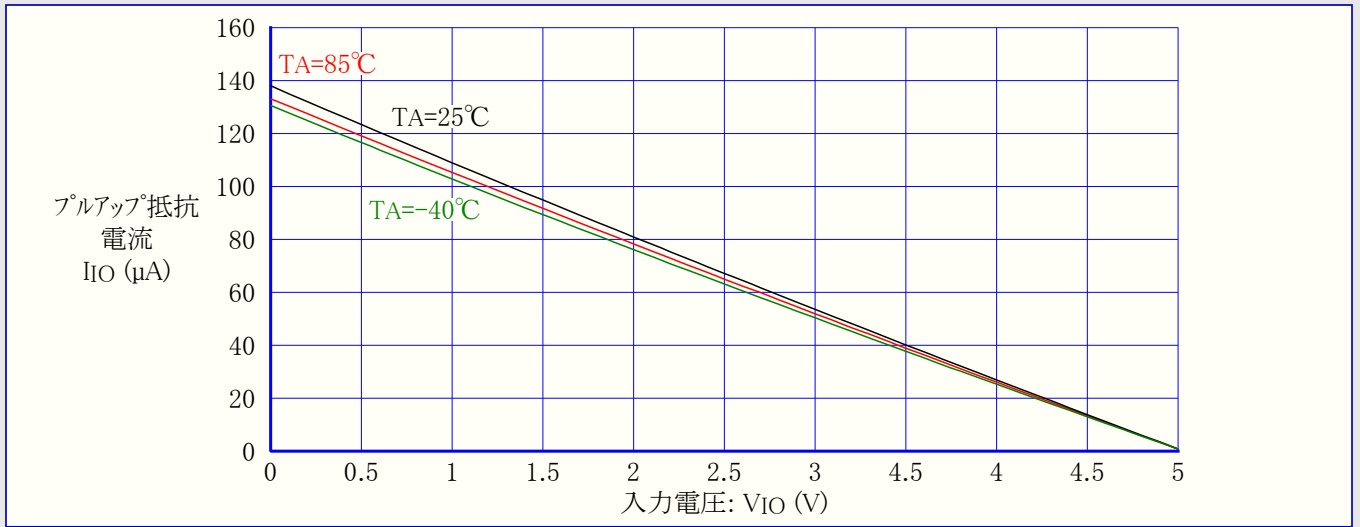


図102. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

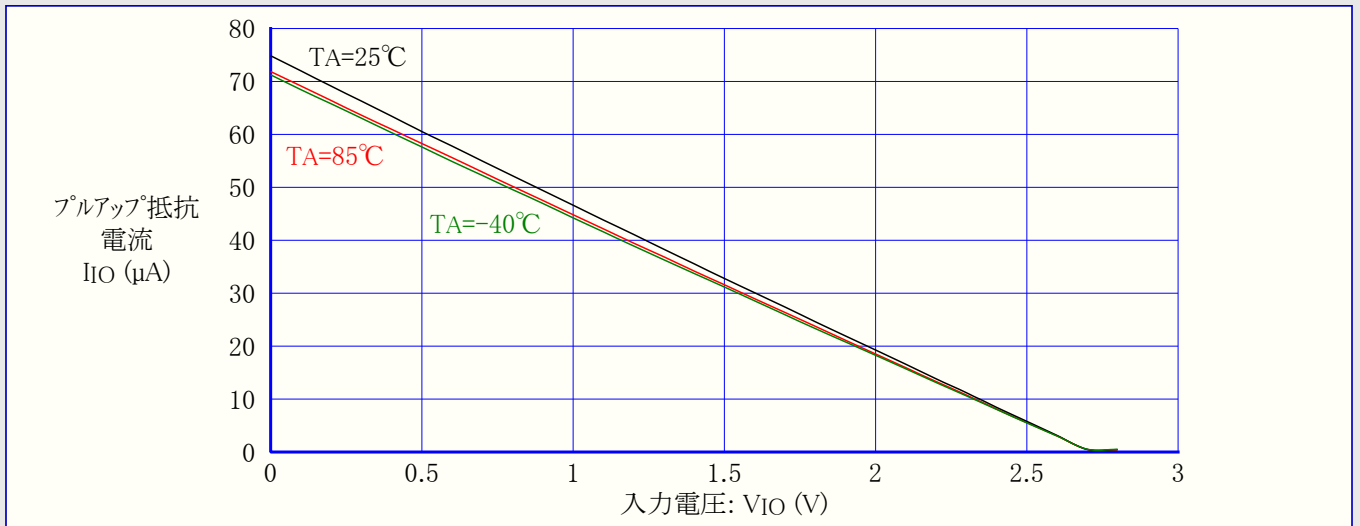


図103. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)

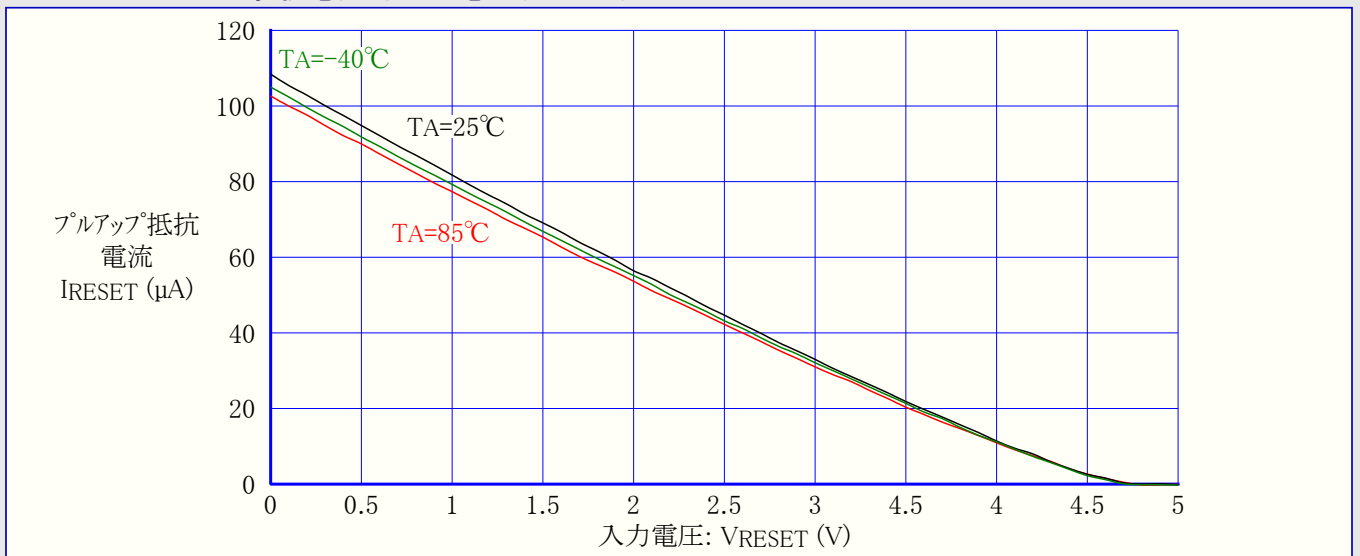
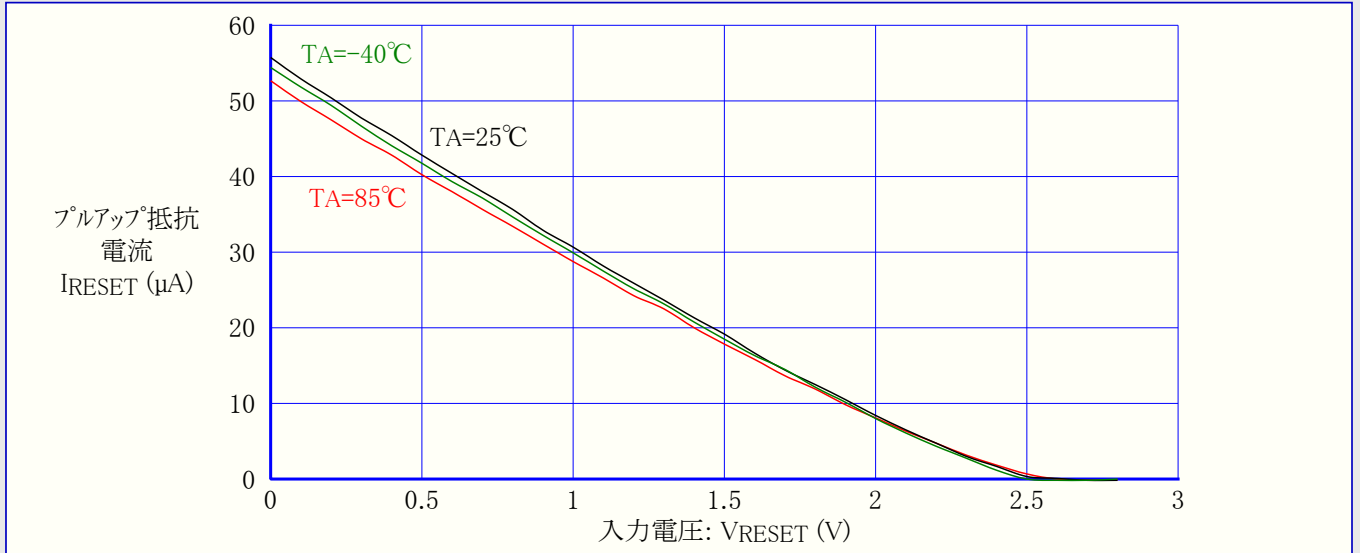


図104. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)



ピン駆動能力

図105. I/Oピン 吐き出し電流 対 出力電圧 (VCC=5V)

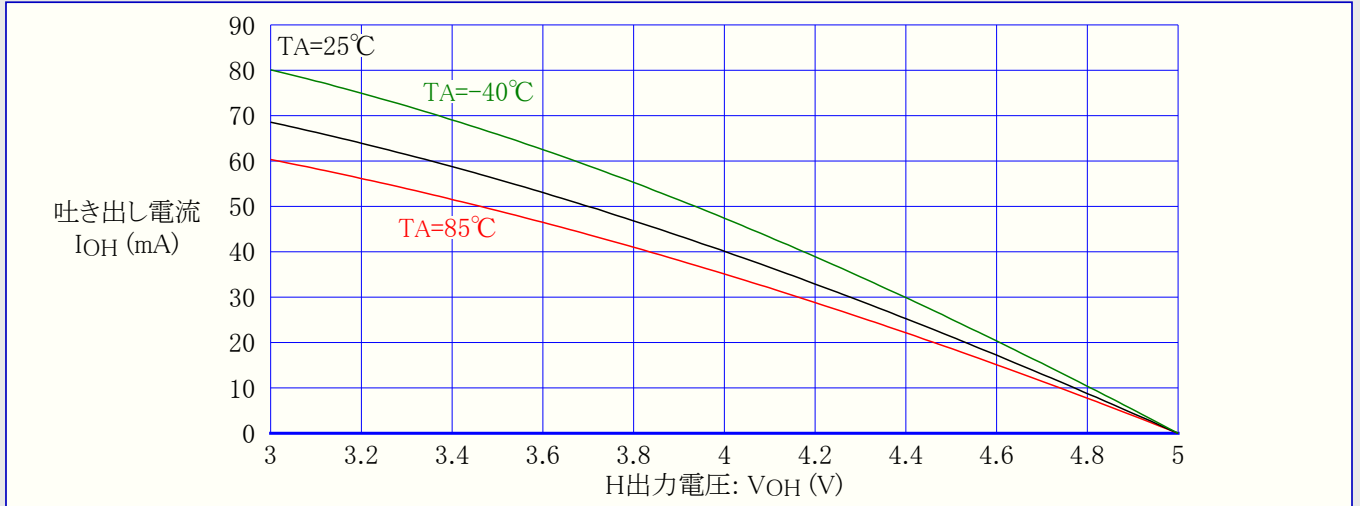


図106. I/Oピン 吐き出し電流 対 出力電圧 (VCC=2.7V)

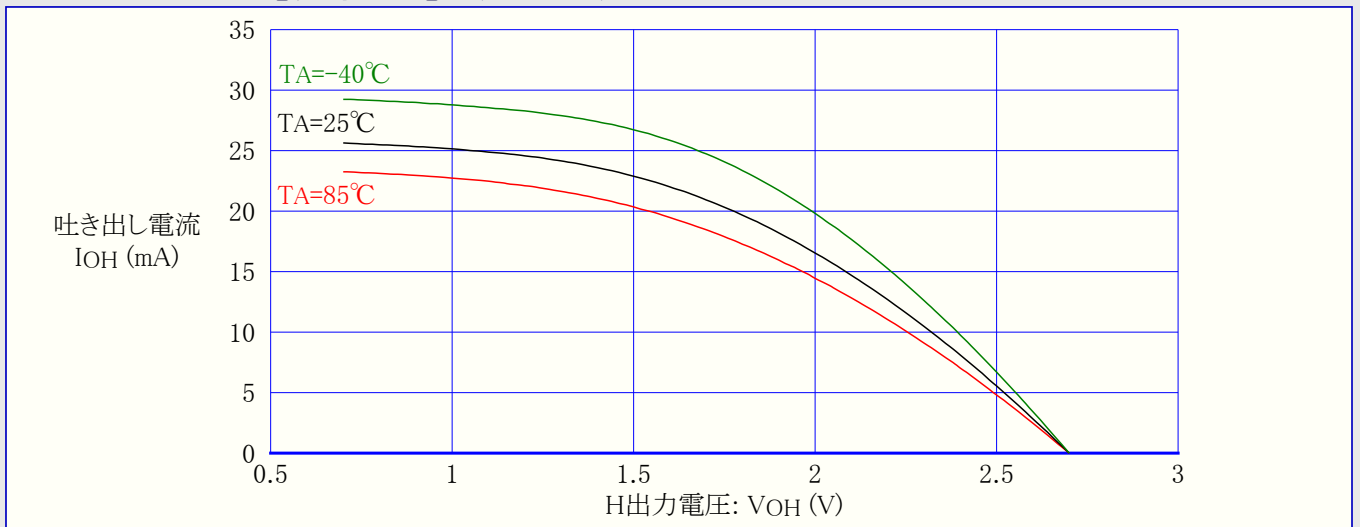


図107. I/Oピン 吐き出し電流 対 出力電圧 (VCC=1.8V)

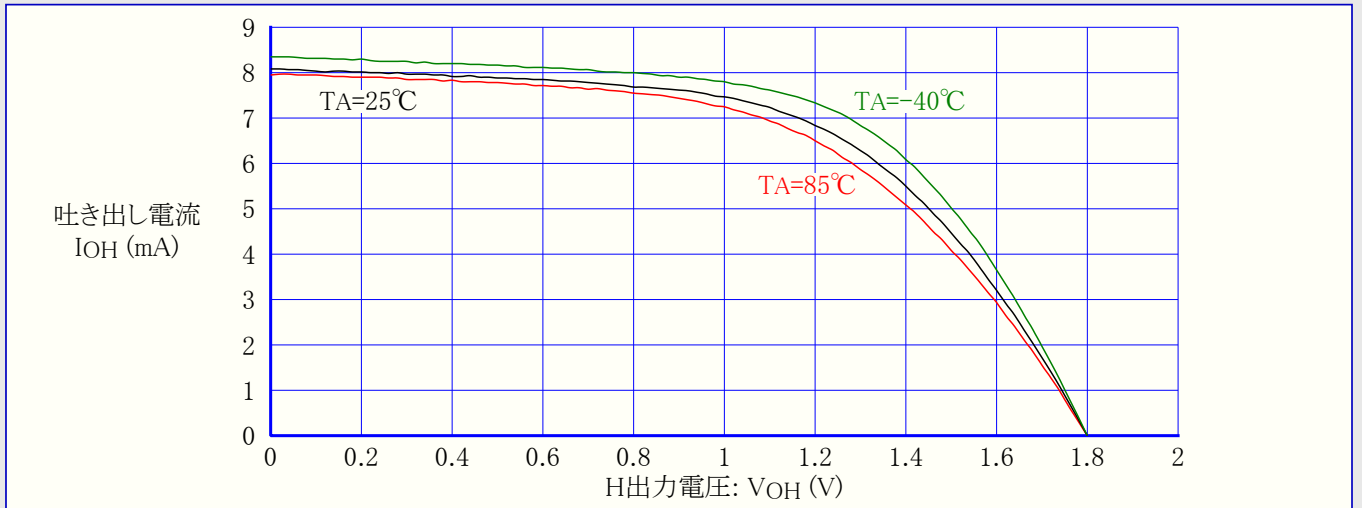


図108. I/Oピン 吸い込み電流 対 出力電圧 (VCC=5V)

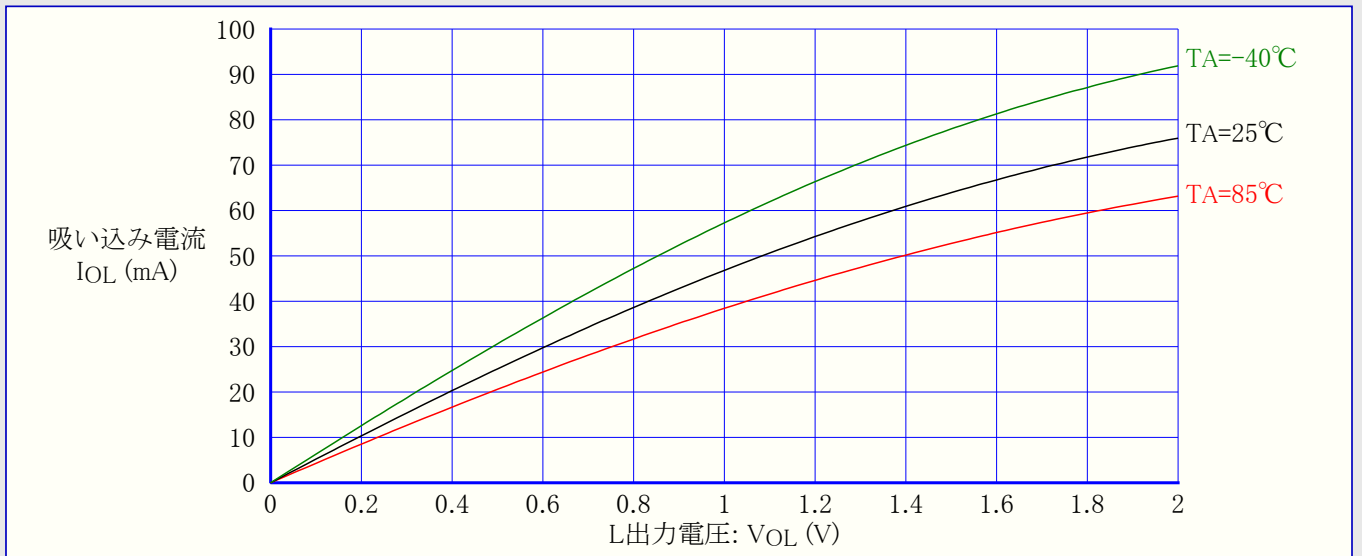


図109. I/Oピン 吸い込み電流 対 出力電圧 (VCC=2.7V)

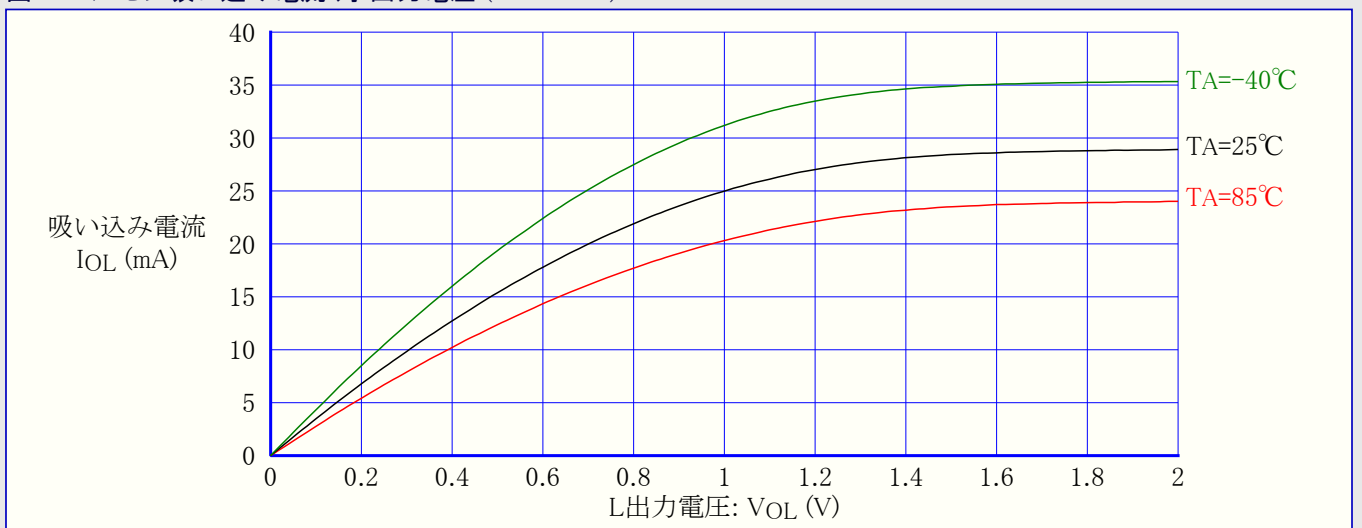


図110. I/Oピン 吸い込み電流 対 出力電圧 (VCC=1.8V)

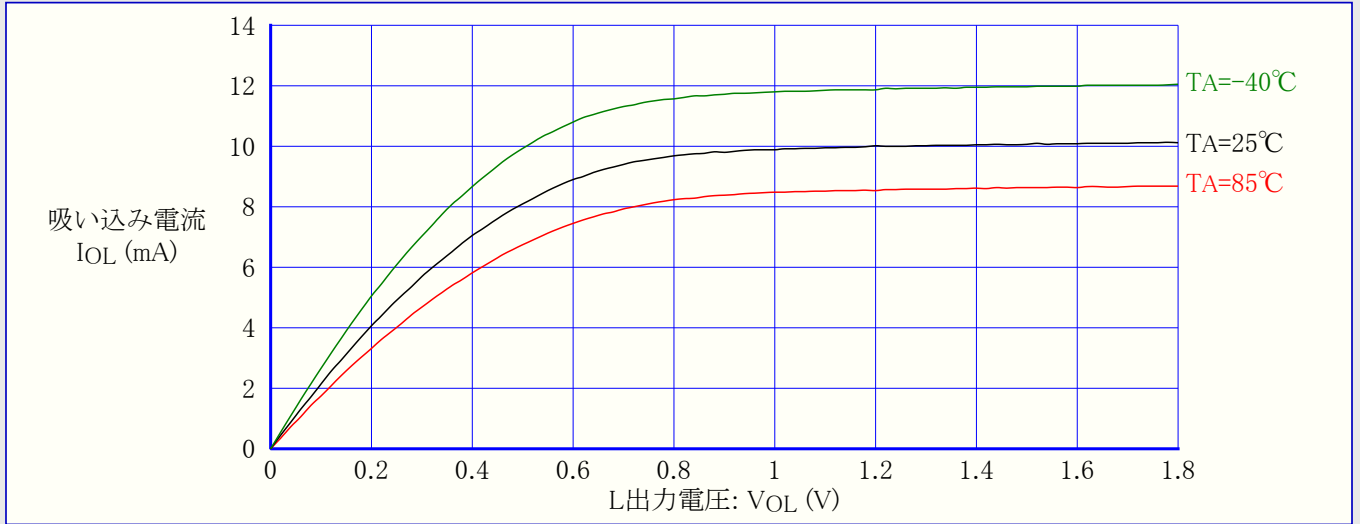


図111. I/OとしてのRESETピン 吐き出し電流 対 出力電圧 (VCC=5V)

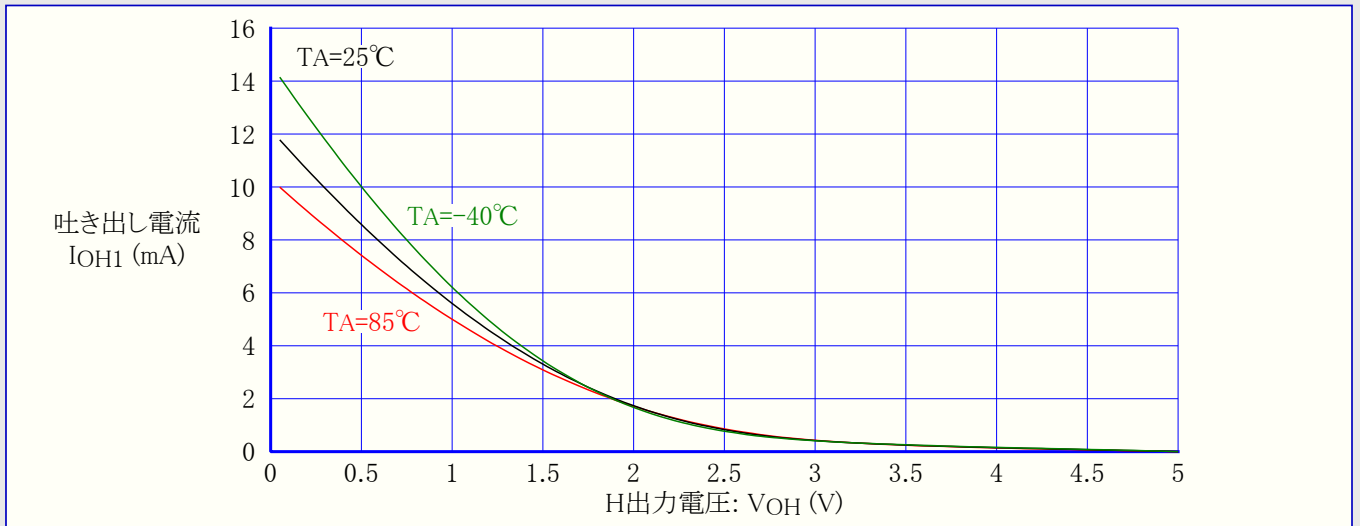


図112. I/OとしてのRESETピン 吐き出し電流 対 出力電圧 (VCC=2.7V)

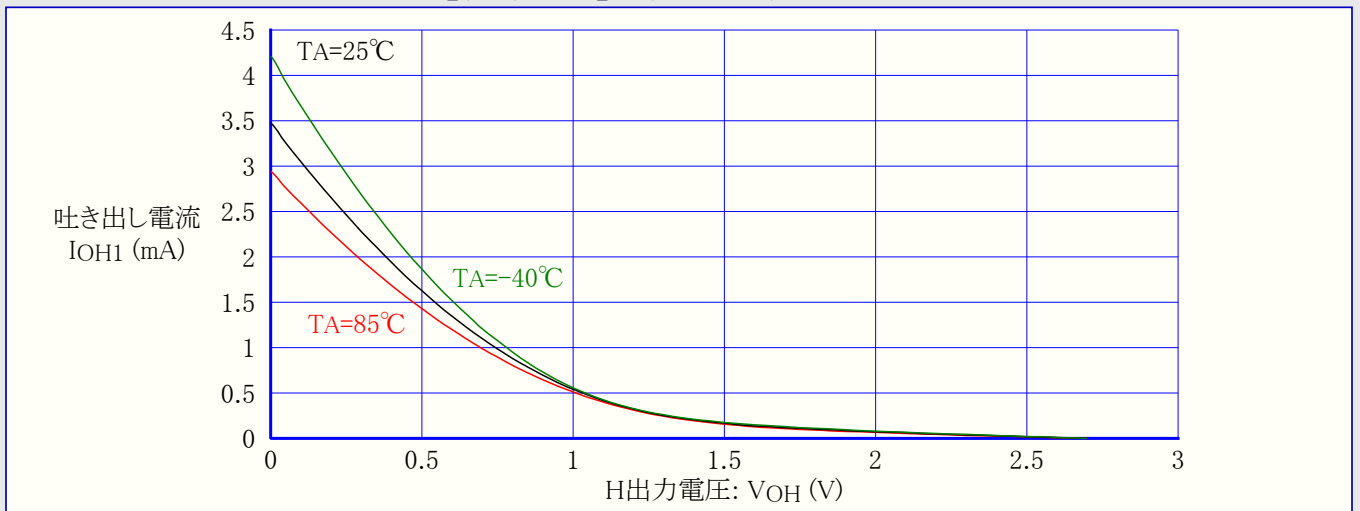


図113. I/OとしてのRESETピンの吐き出し電流 対 出力電圧 (VCC=1.8V)

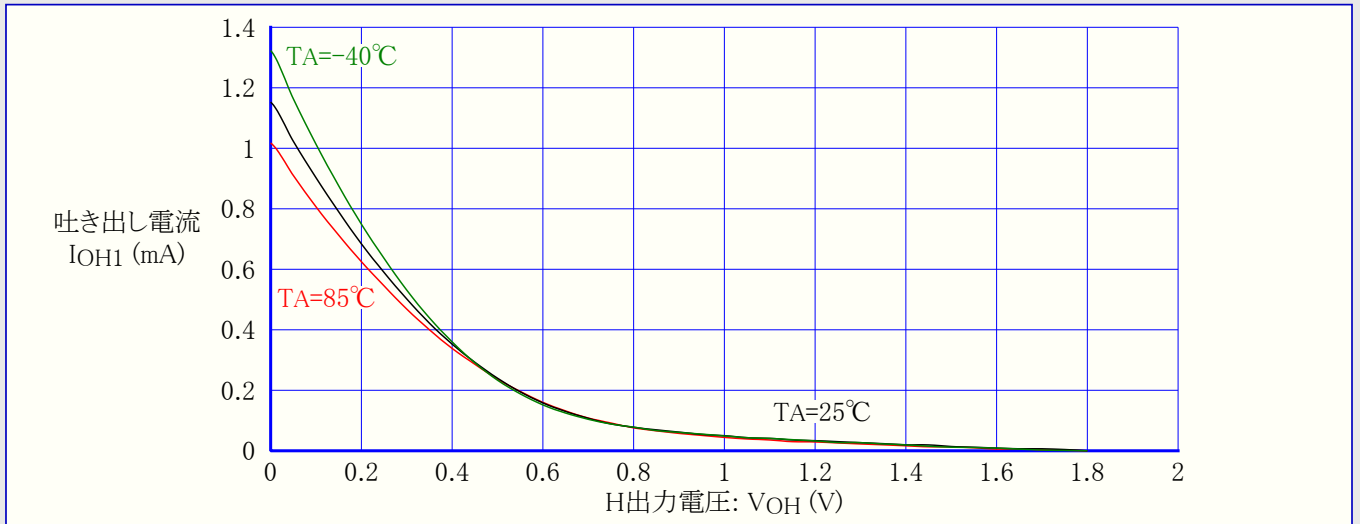


図114. I/OとしてのRESETピンの吸い込み電流 対 出力電圧 (VCC=5V)

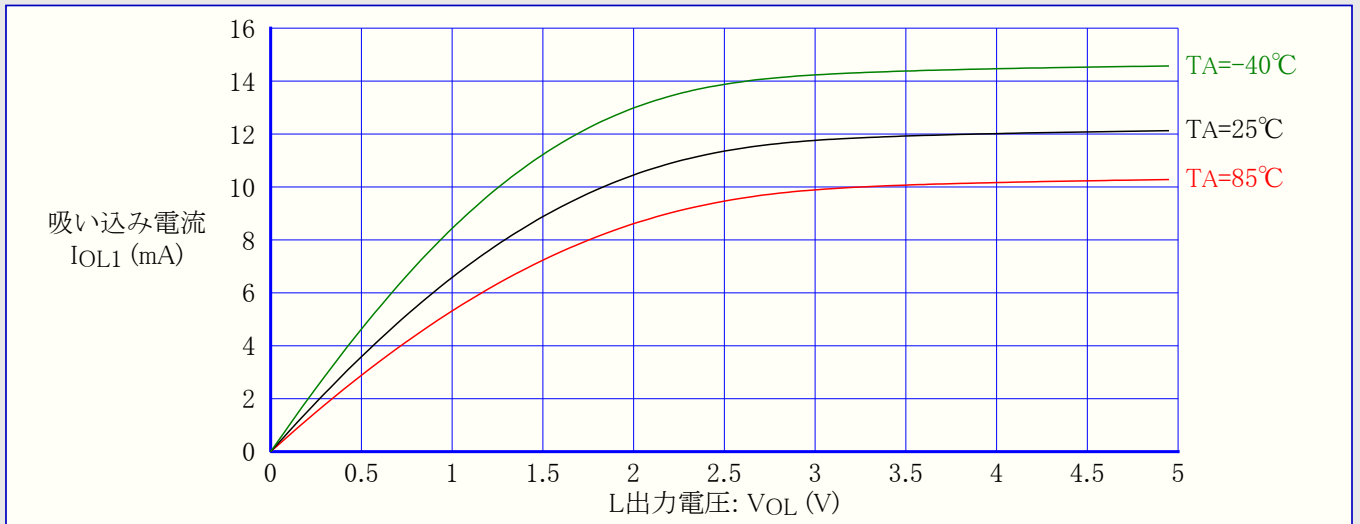


図115. I/OとしてのRESETピンの吸い込み電流 対 出力電圧 (VCC=2.7V)

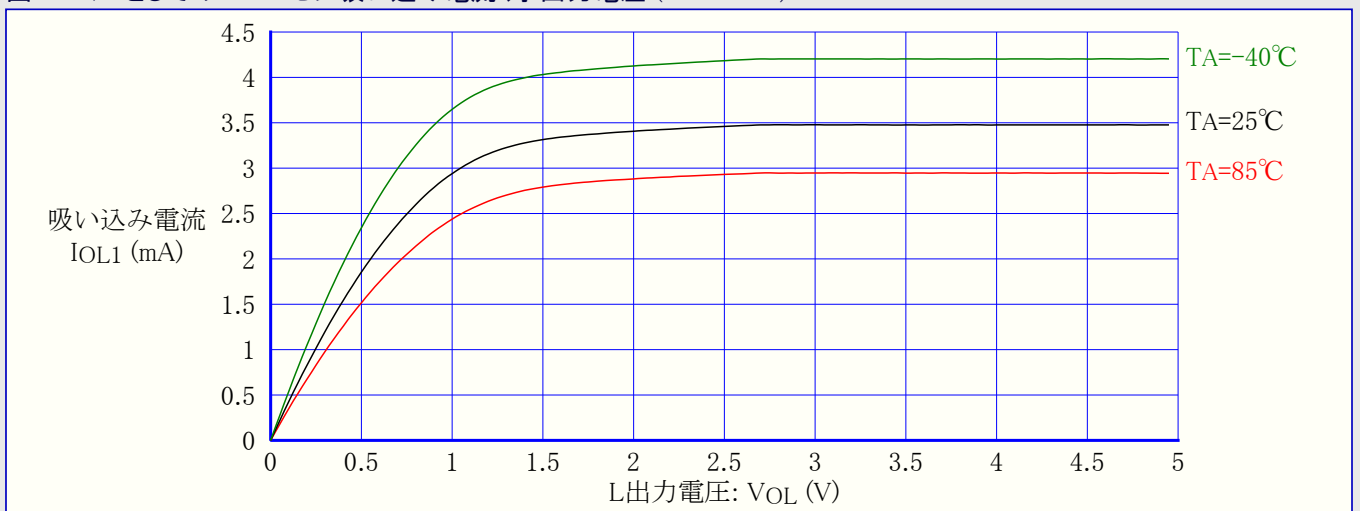
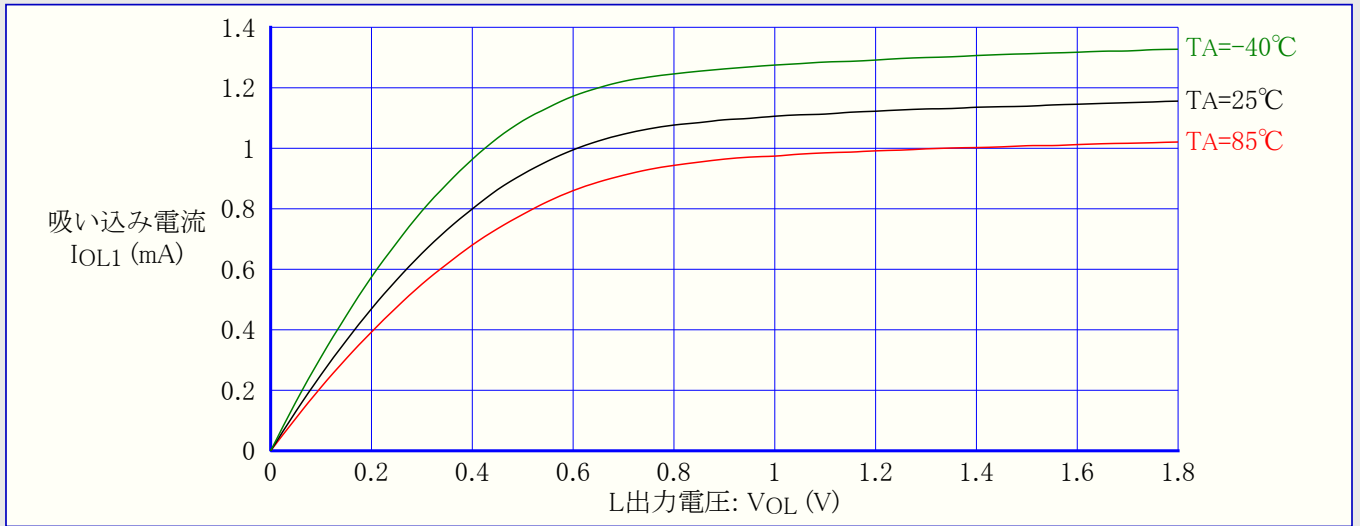


図116. I/OとしてのRESETピン 吸い込み電流 対 出力電圧 (VCC=1.8V)



ピン 閾値とヒステリシス

図117. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IH} , 1読み値)

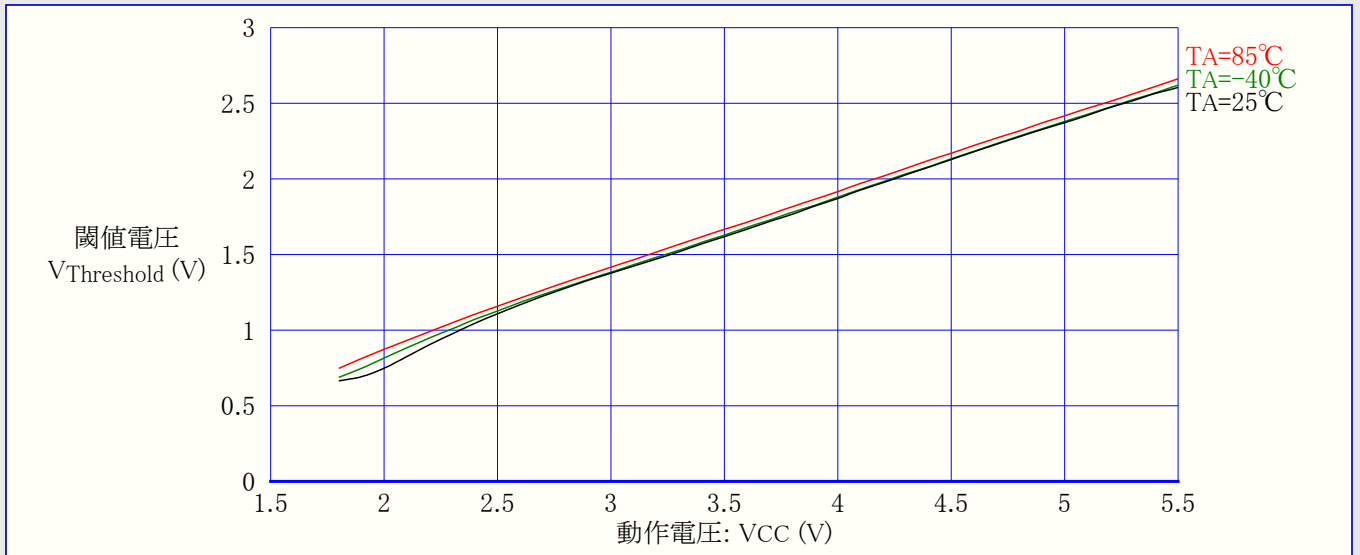


図118. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IL} , 0読み値)

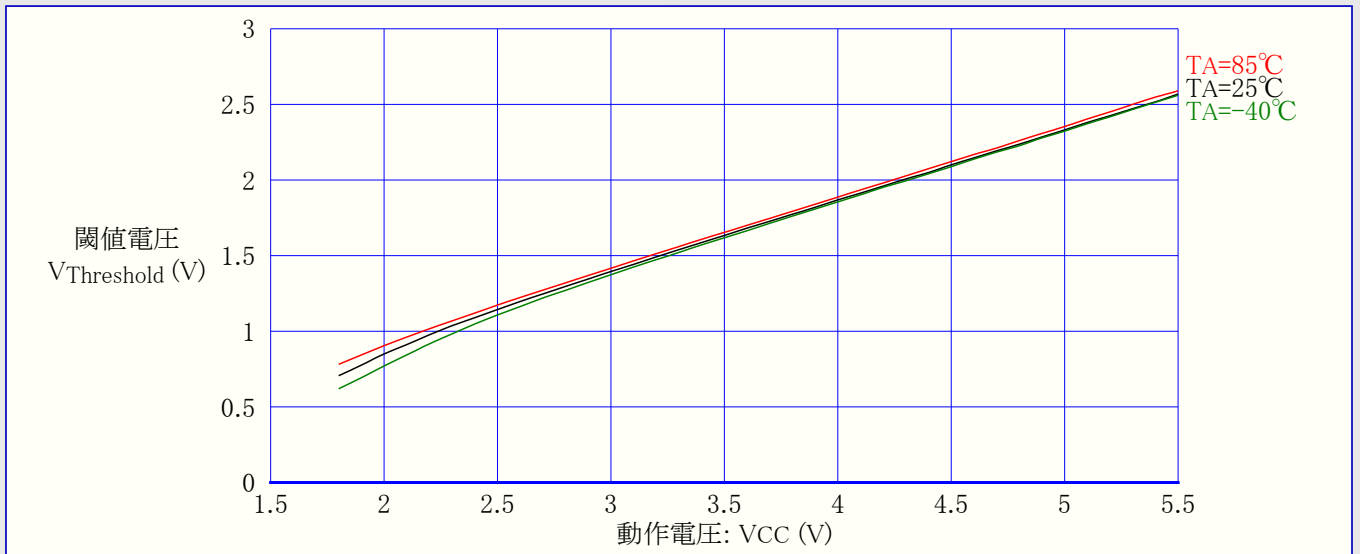


図119. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ($V_{IH,1}$ 読み値)

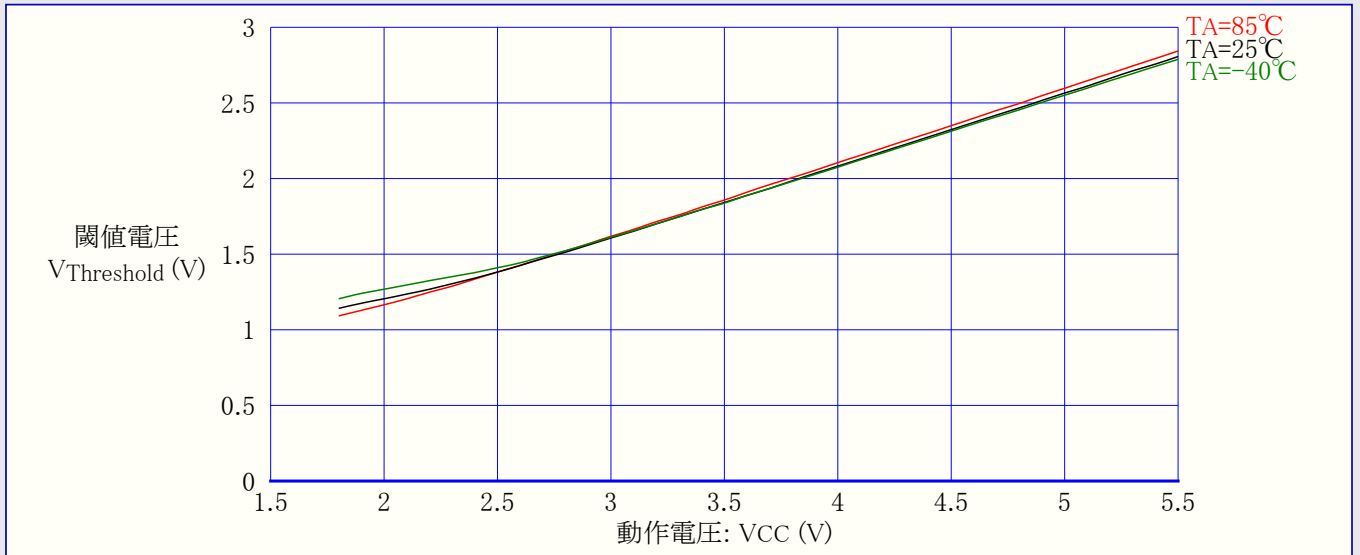


図120. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ($V_{IL,0}$ 読み値)

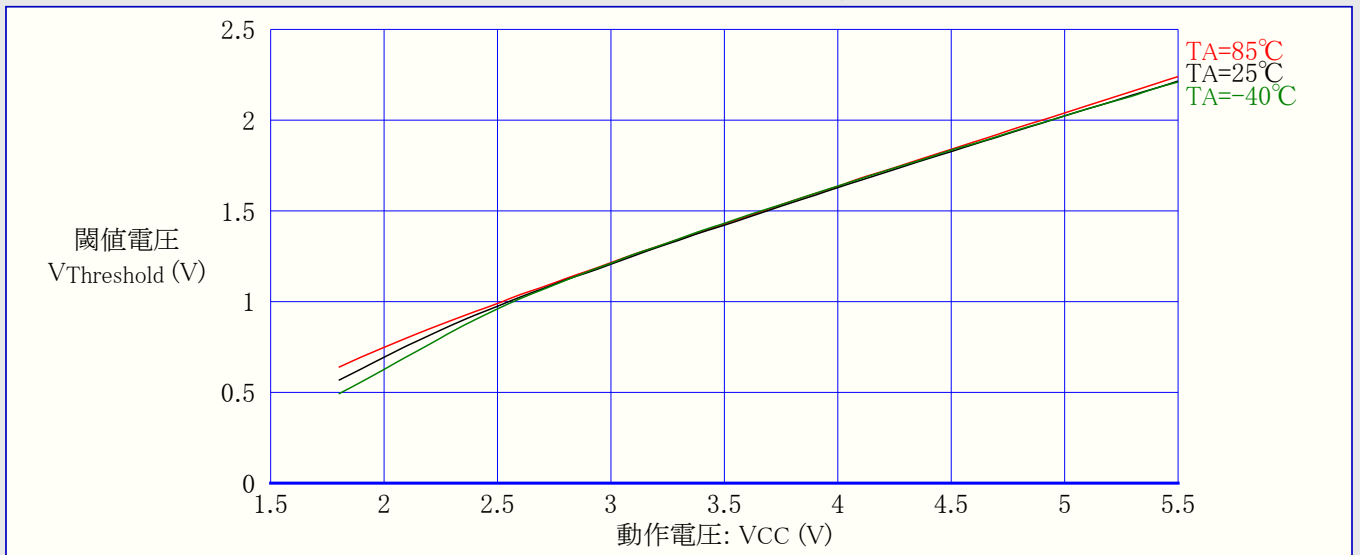


図121. I/OとしてのRESETピン入力ヒステリシス電圧 対 動作電圧

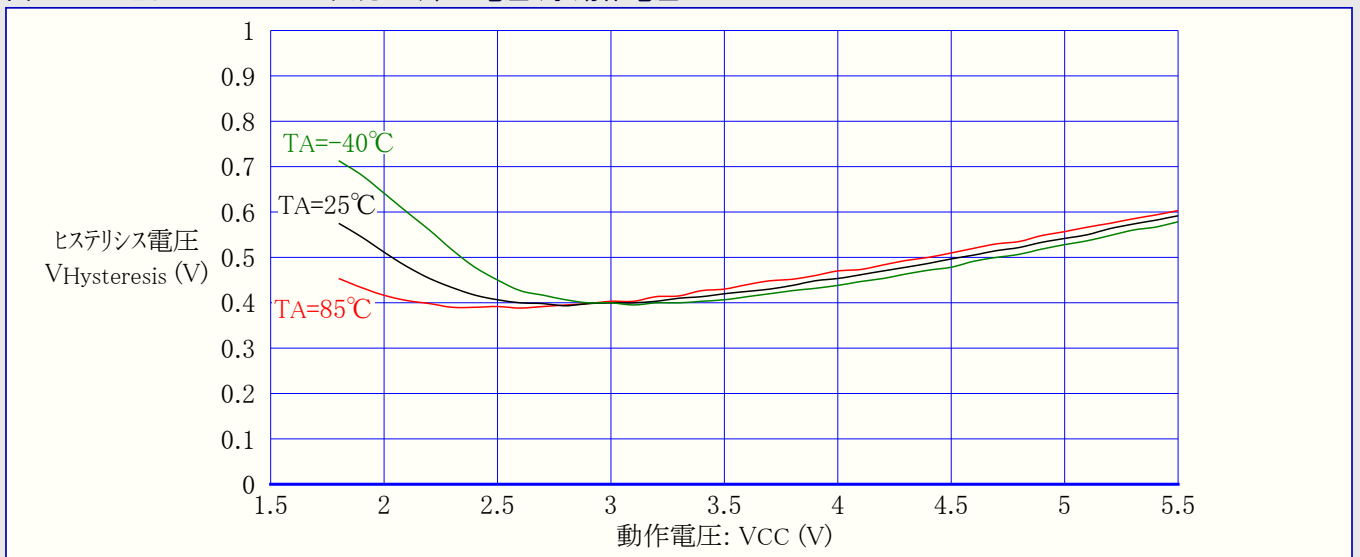


図122. RESET入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IH} , 1読み値)

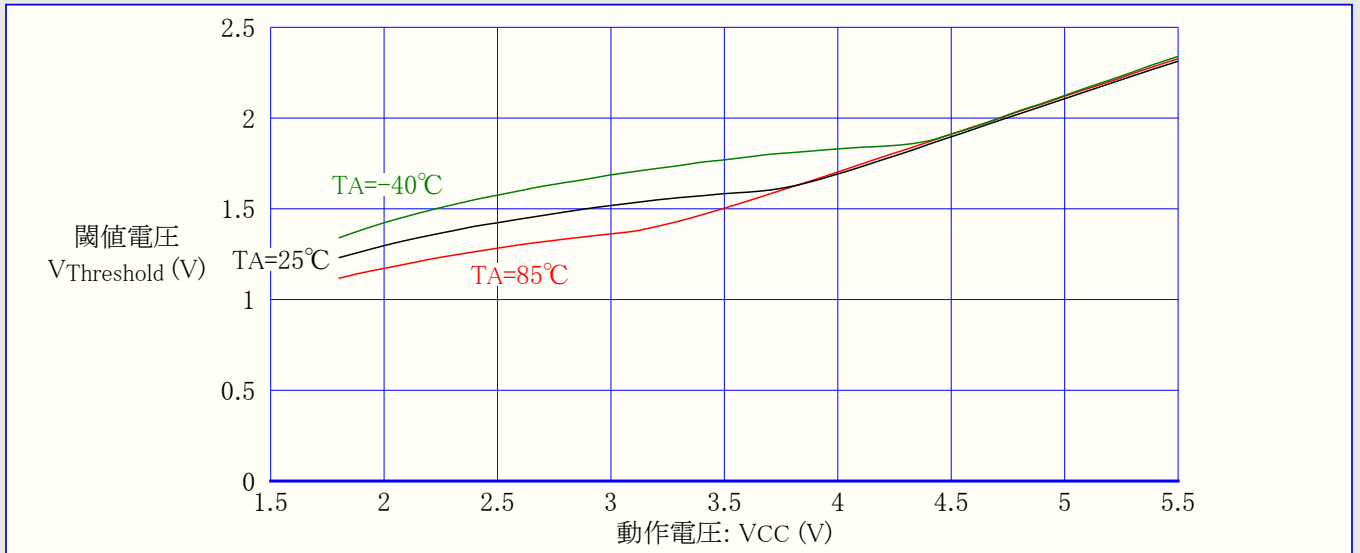


図123. RESET入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IL} , 0読み値)

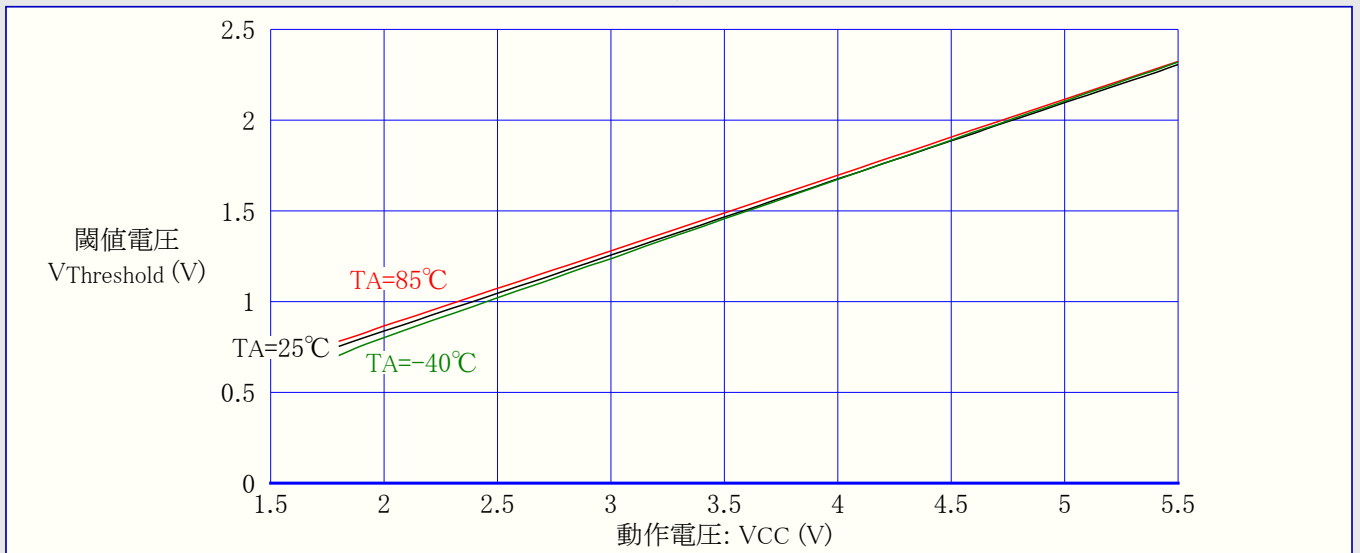
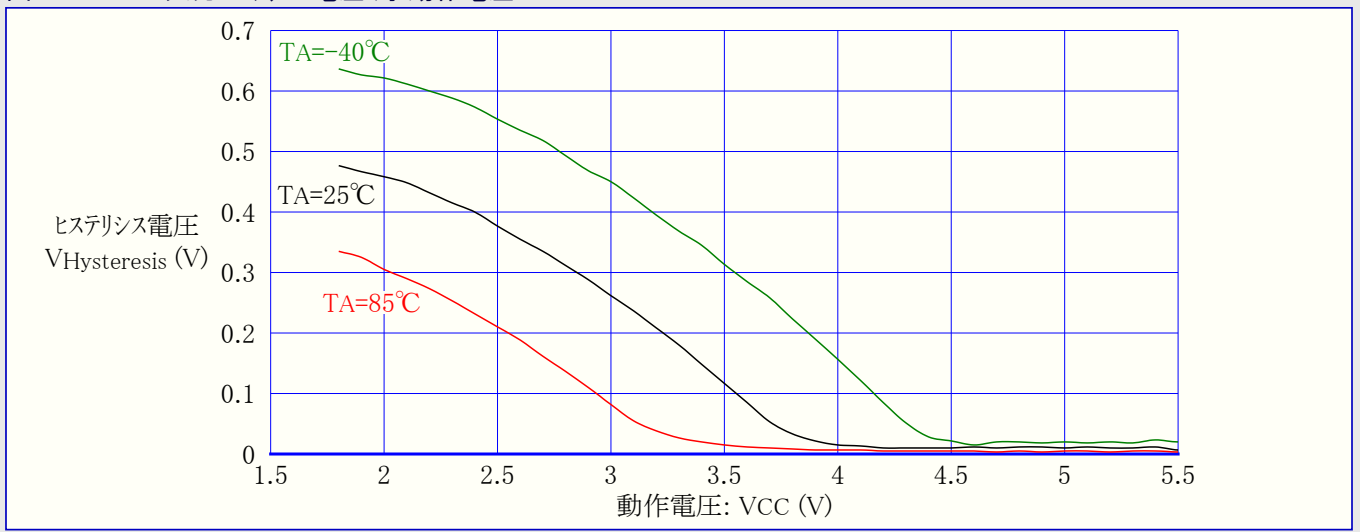


図124. RESET入力ヒステリシス電圧 対 動作電圧



低電圧検出器(BOD)閾値とアナログ比較器変位(オフセット)

図125. 低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧4.3V)

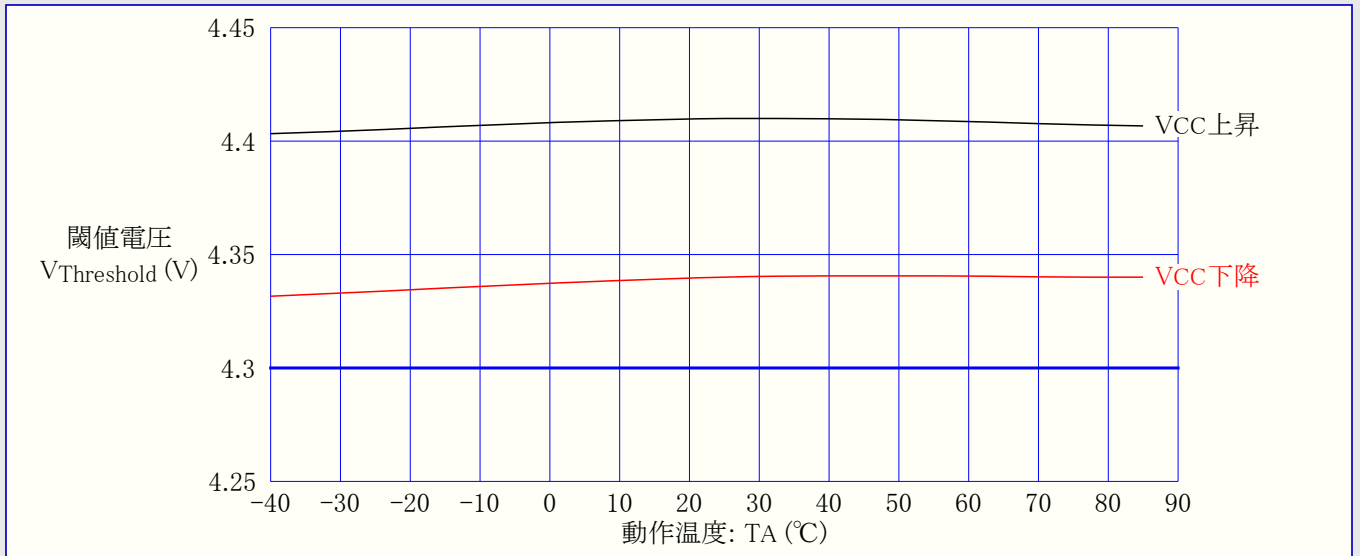


図126. 低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧2.7V)

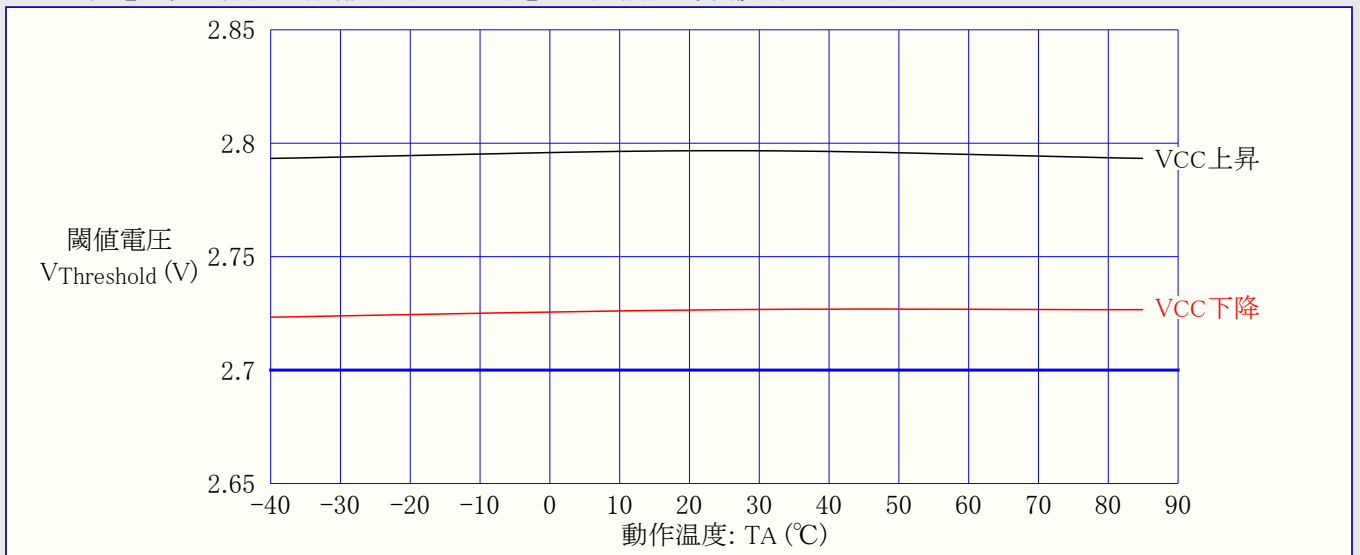


図127. 低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧1.8V)

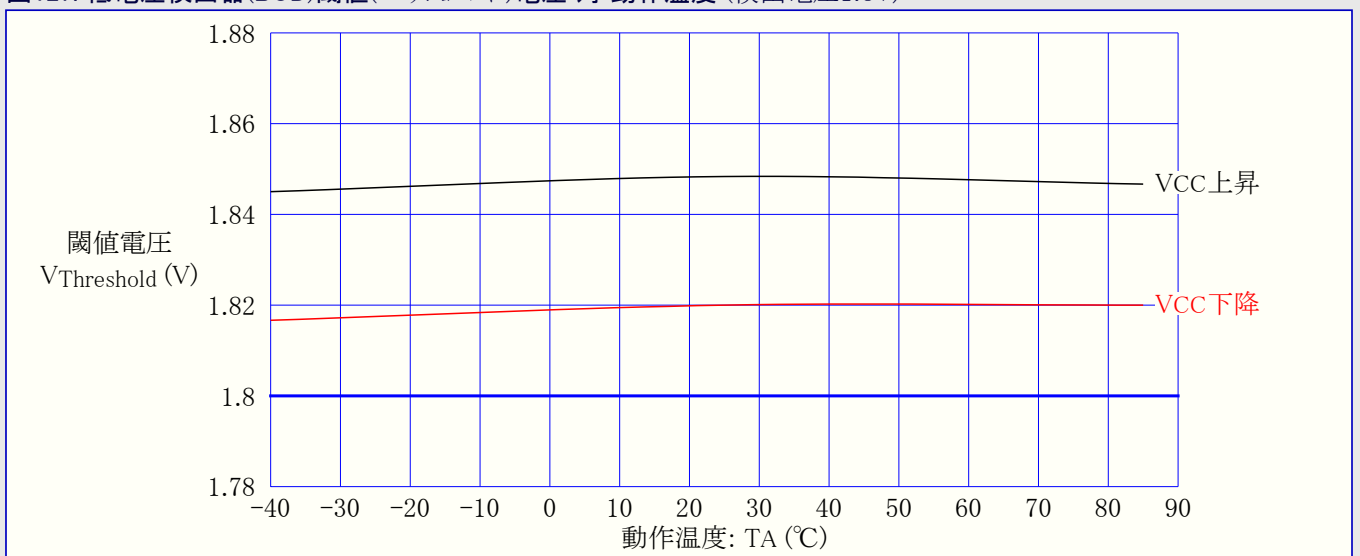
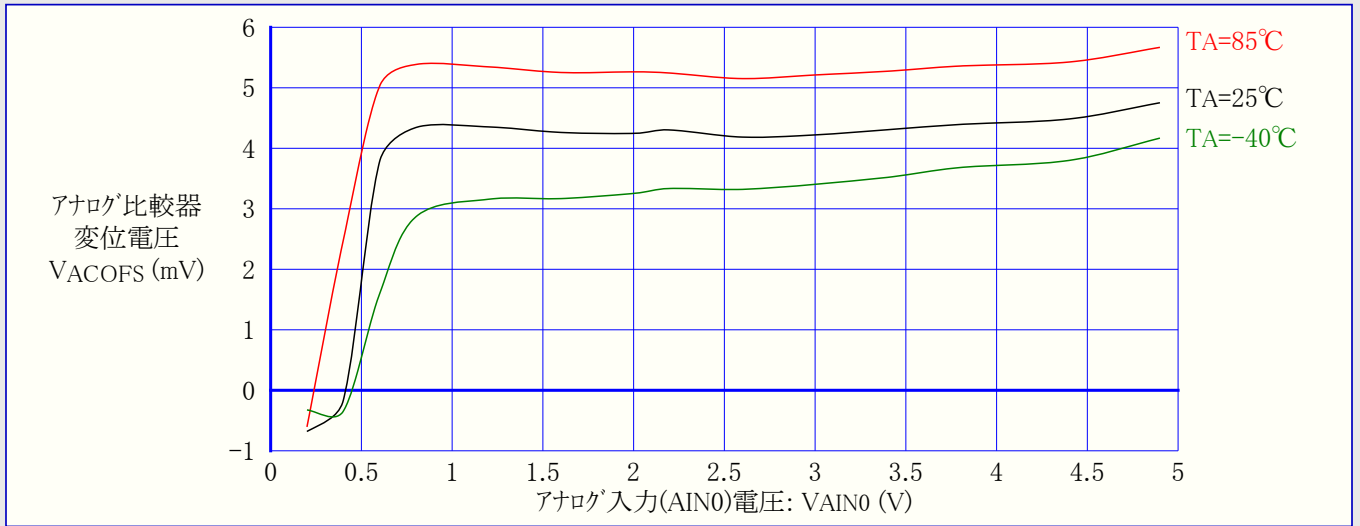


図128. アナログ比較器変位(オフセット)電圧 対 アナログ入力電圧 (VCC=5V)



内部発振器周波数

図129. 128kHzウォッチドッグ発振器周波数 対 動作電圧

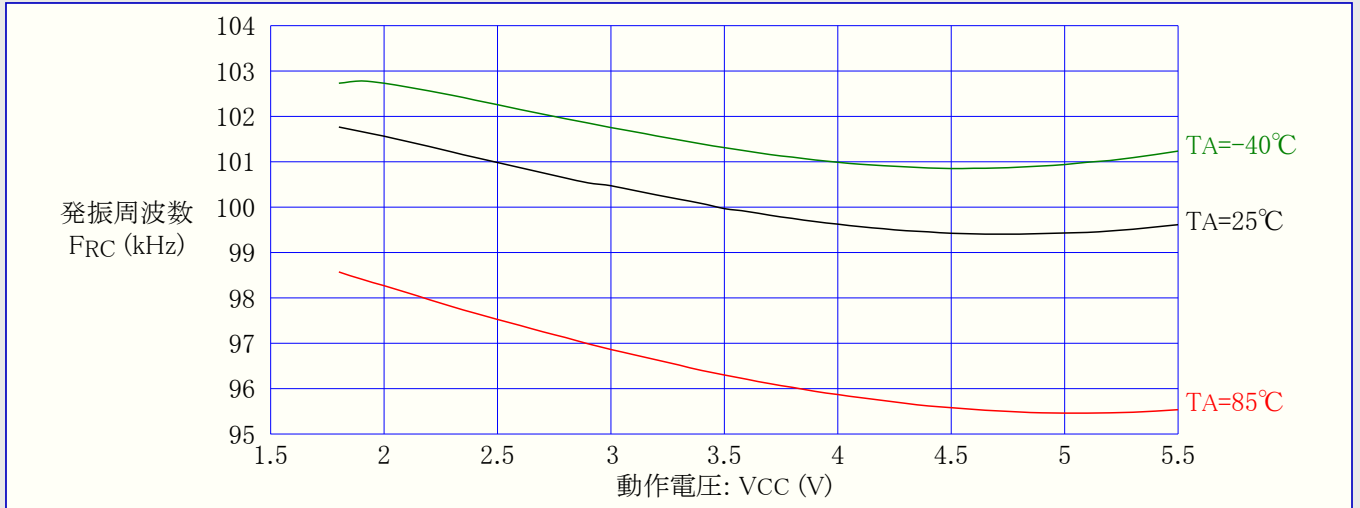


図130. 128kHzウォッチドッグ発振器周波数 対 動作温度

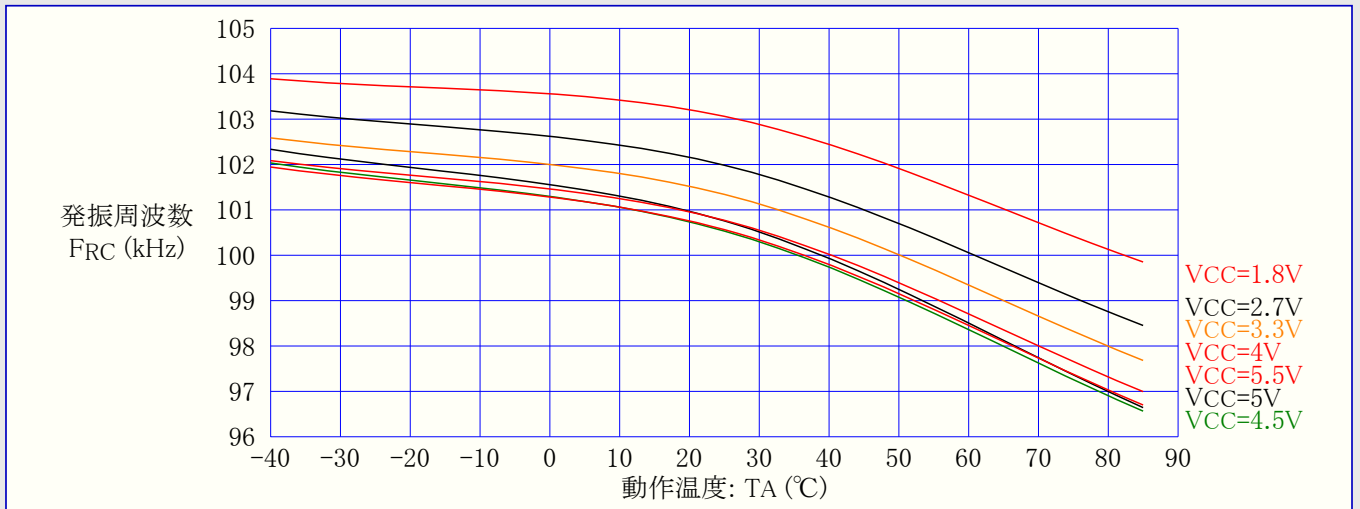


図131. 校正済み8MHz内蔵RC発振器周波数 対 動作温度

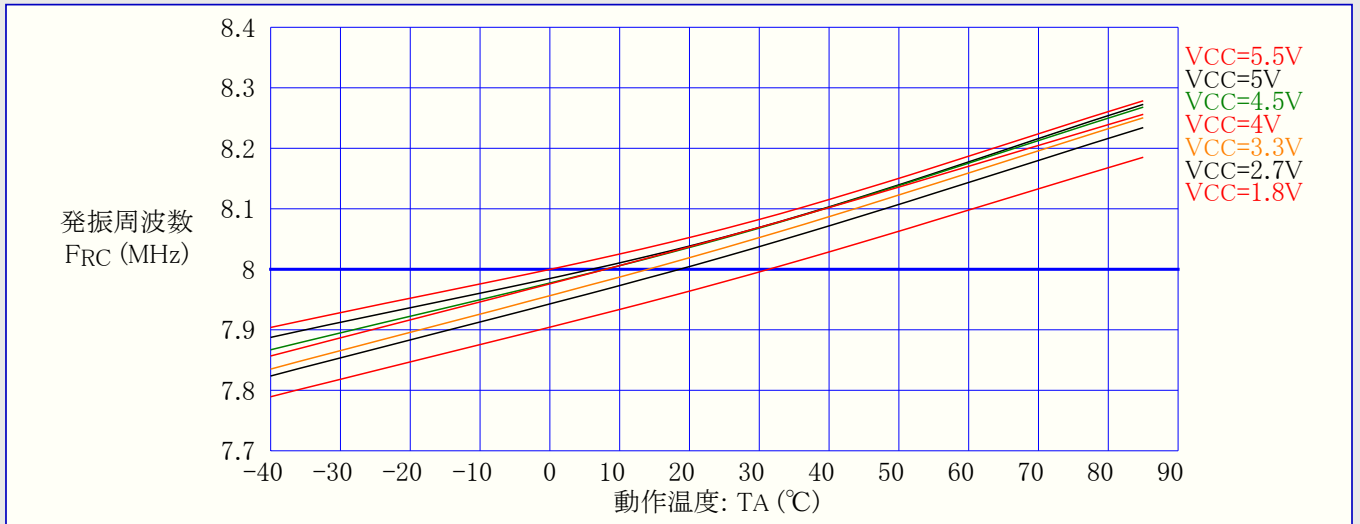


図132. 校正済み8MHz内蔵RC発振器周波数 対 動作電圧

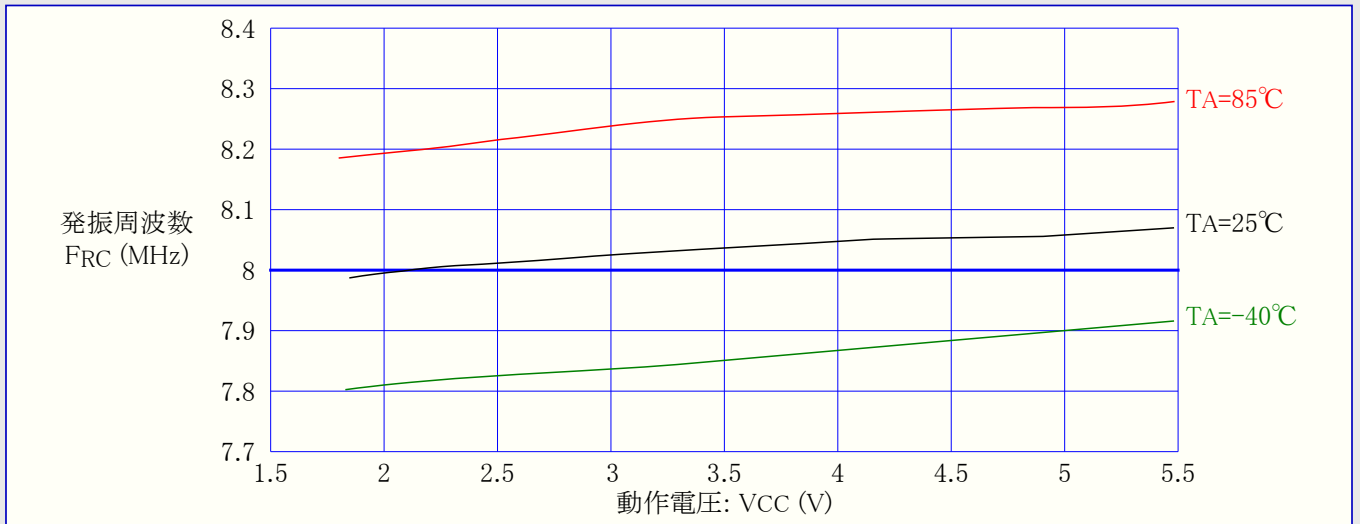


図133. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値

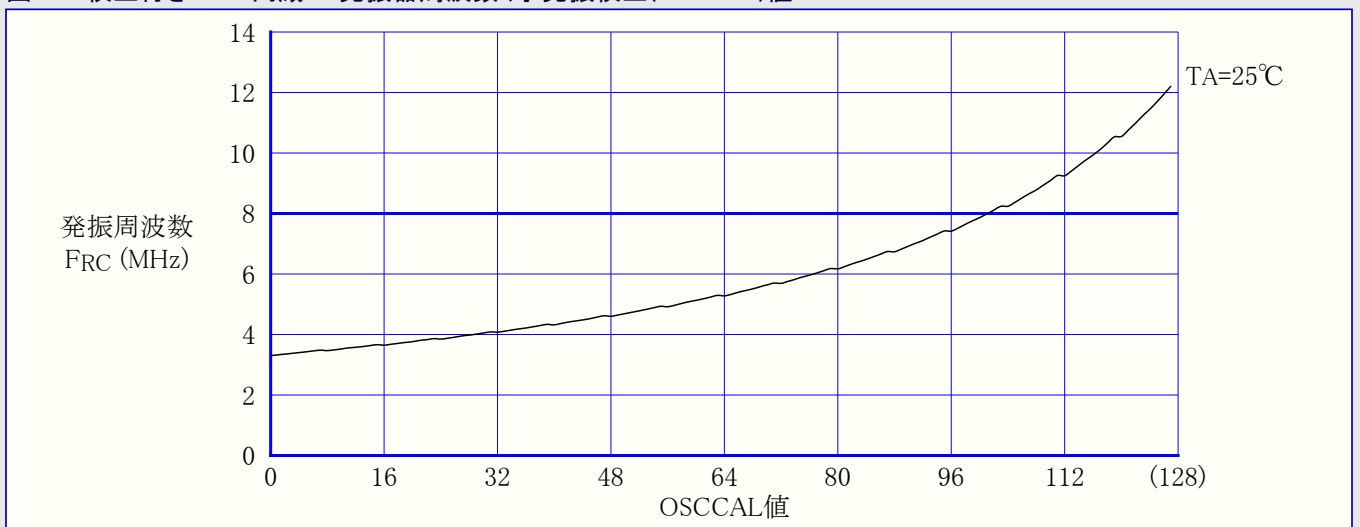


図134. 校正済み4MHz内蔵RC発振器周波数 対 動作温度

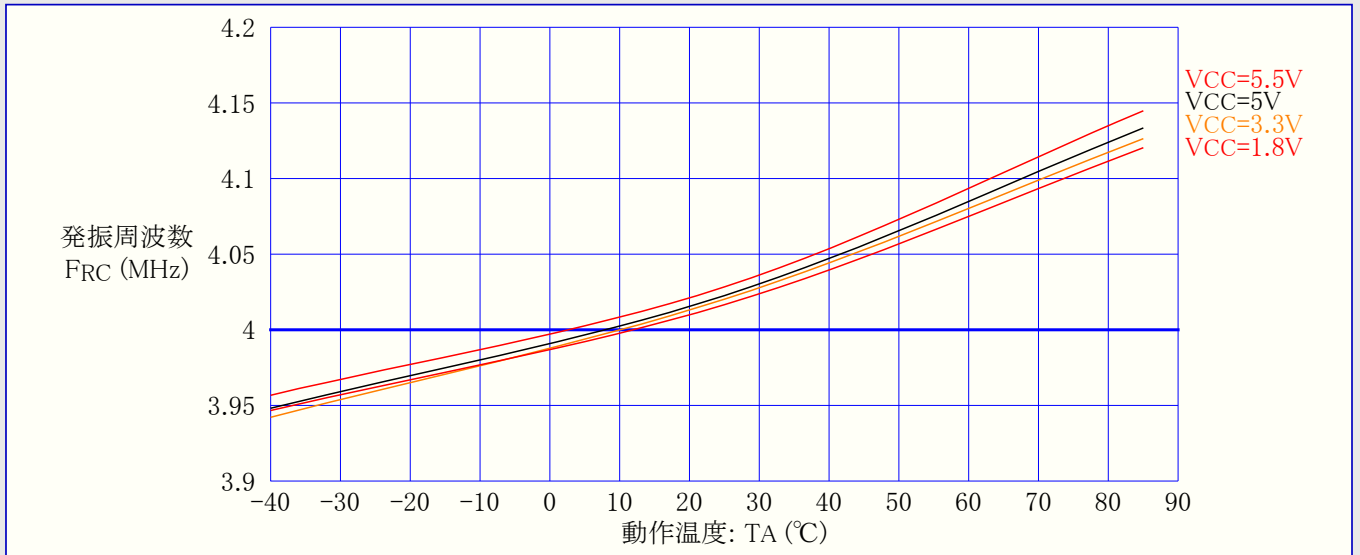


図135. 校正済み4MHz内蔵RC発振器周波数 対 動作電圧

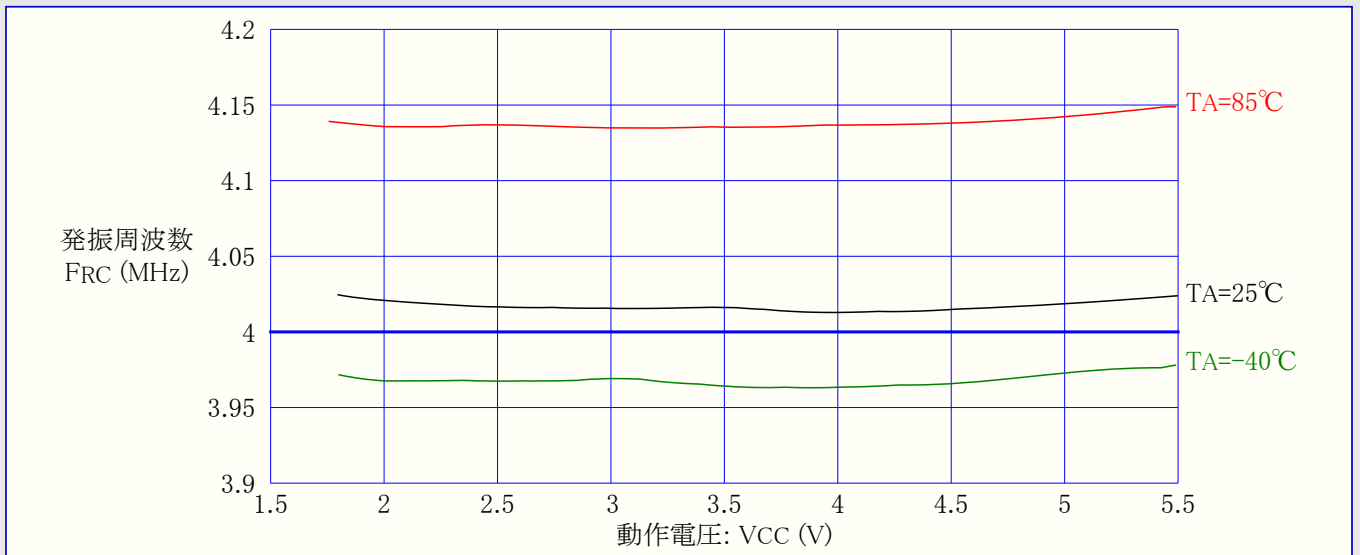
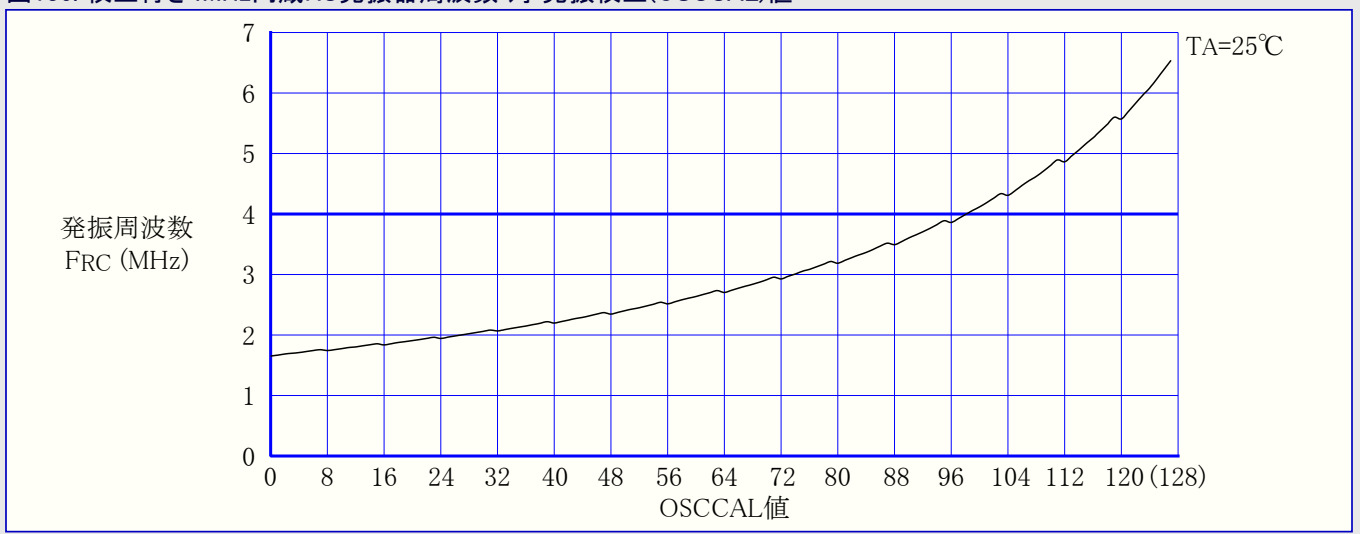


図136. 校正付き4MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値



周辺機能部消費電流

図137. 低電圧検出器(BOD)消費電流 対 動作電圧

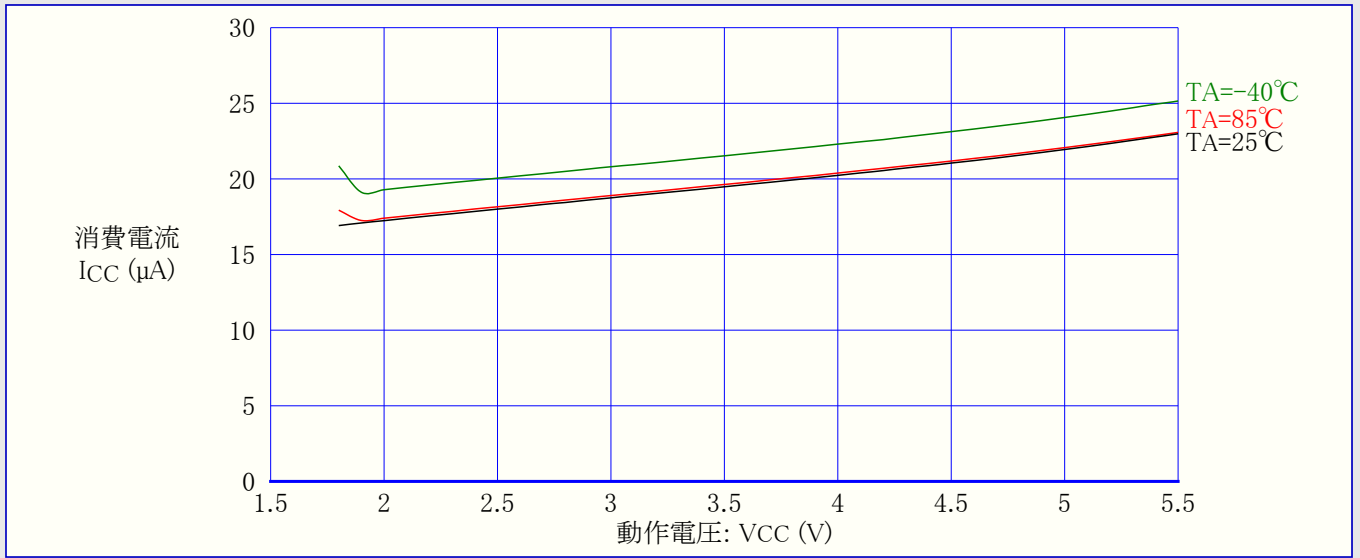


図138. アナログ比較器消費電流 対 動作電圧

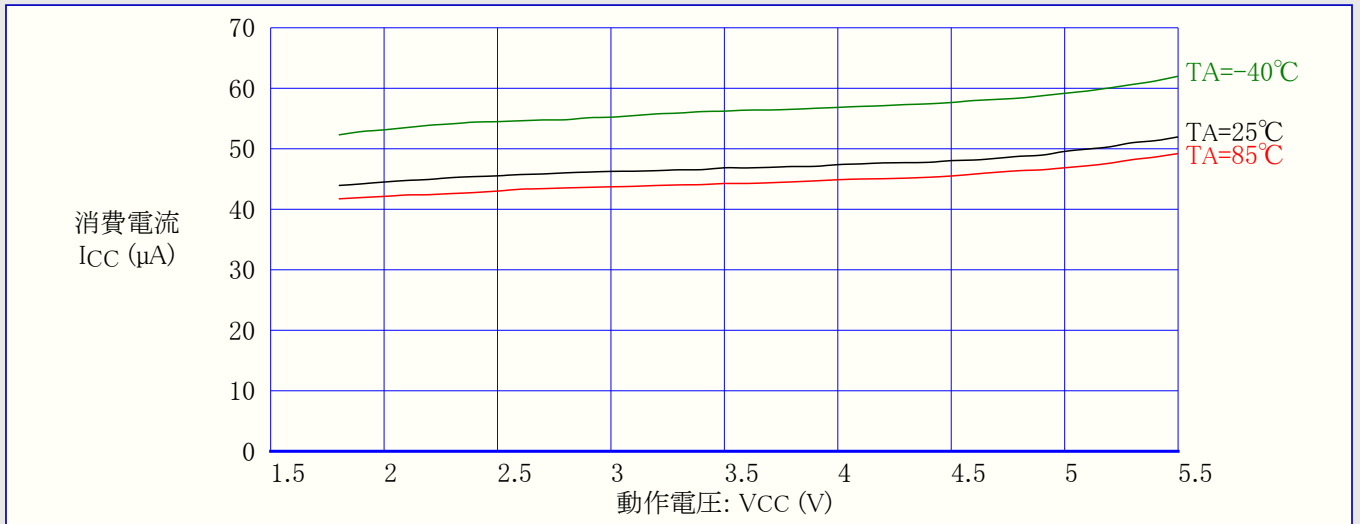
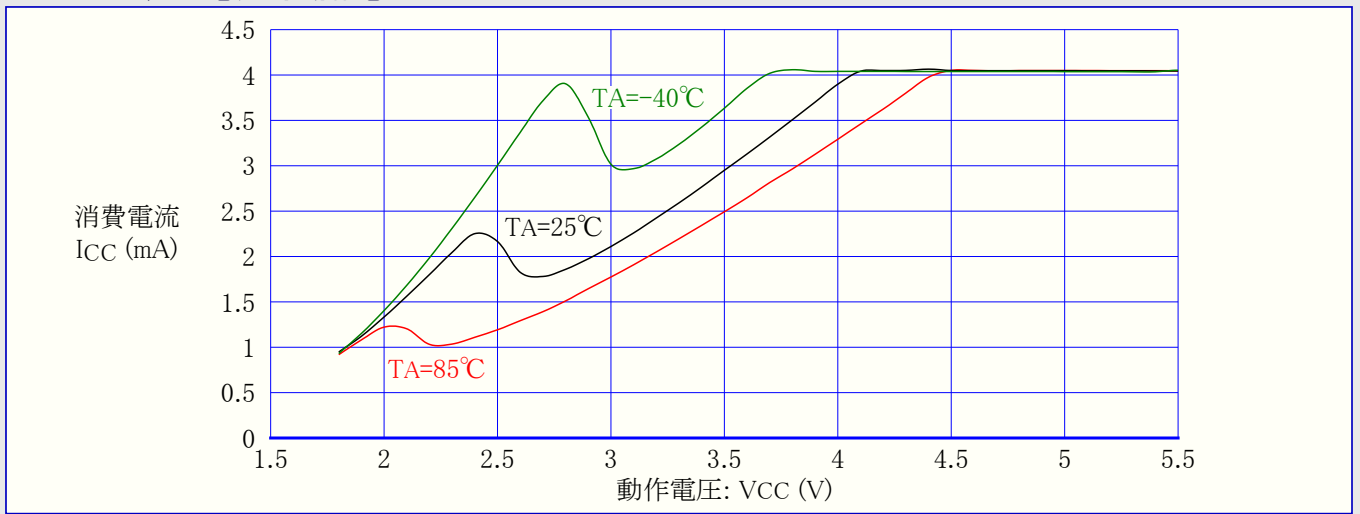


図139. プログラミング電流 対 動作電圧



リセット消費電流とリセットパルス幅

図140. リセット消費(供給)電流 対 周波数 (100kHz~1MHz、RESETプルアップ電流を除く)

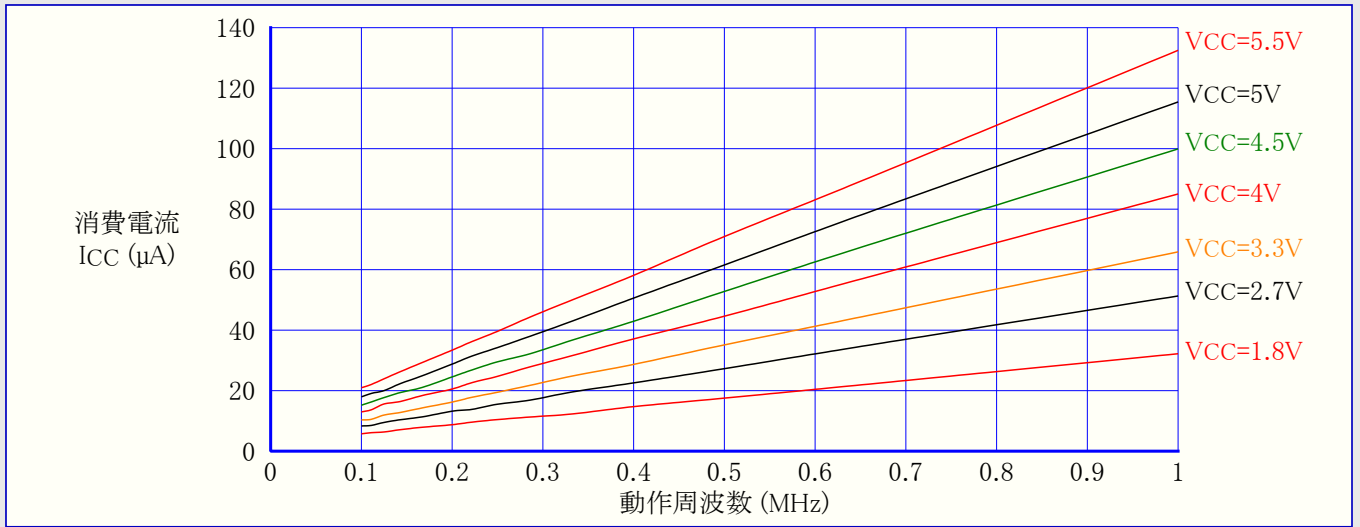


図141. リセット消費(供給)電流 対 周波数 (1MHz~20MHz、RESETプルアップ電流を除く)

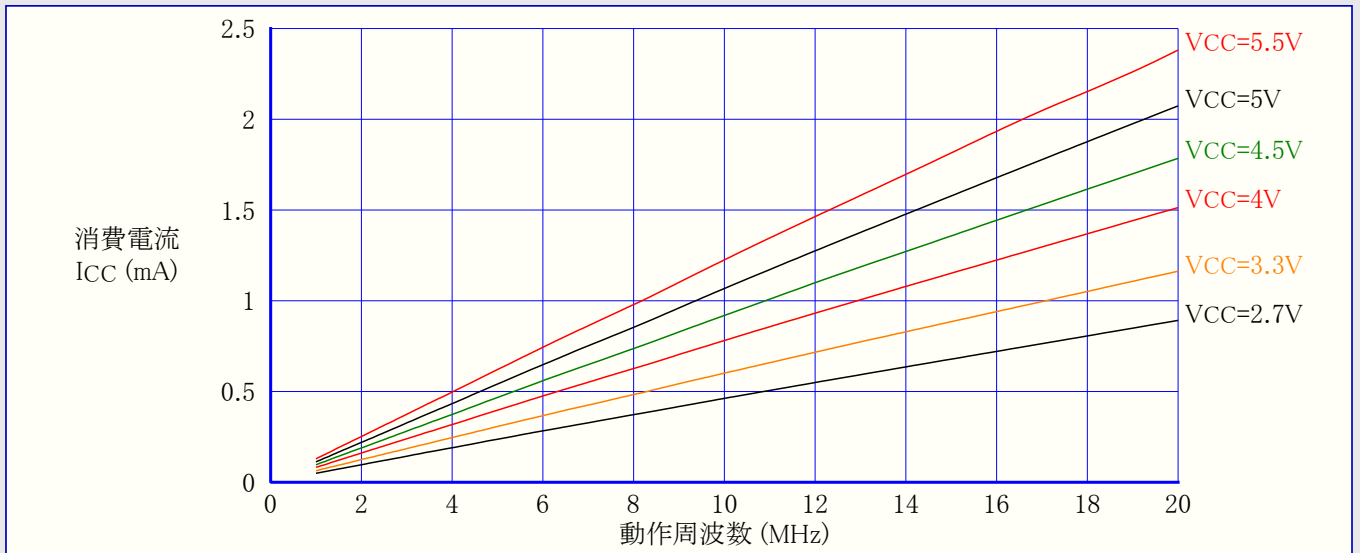
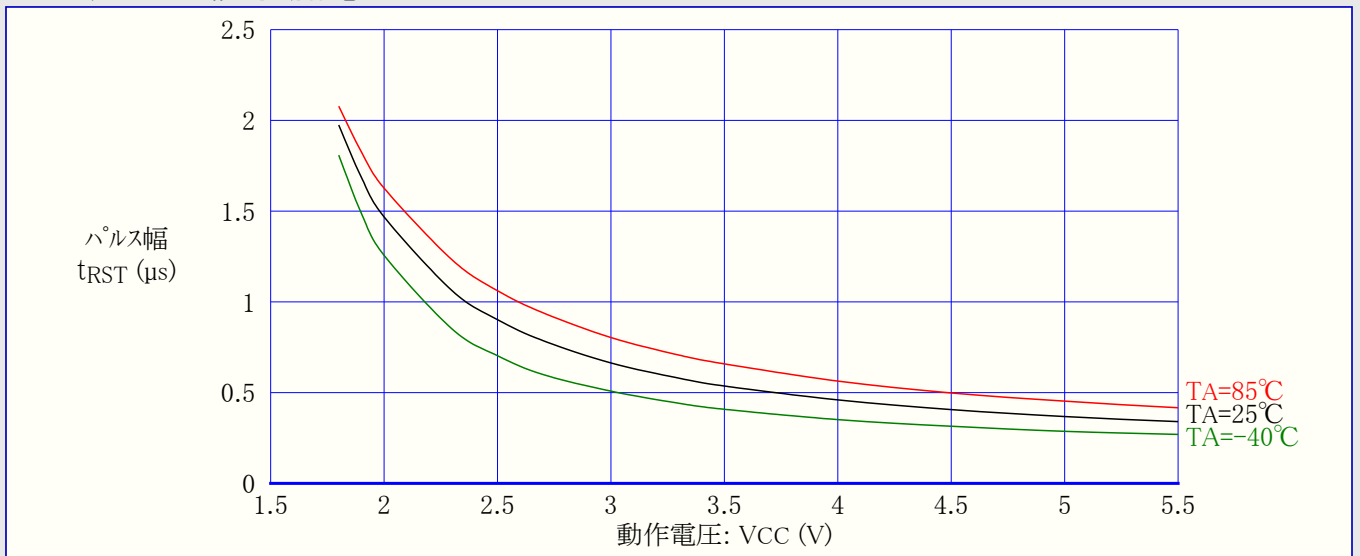


図142. リセットパルス幅 対 動作電圧



レジスタ要約

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁	
\$3F (\$5F)	SREG	I	T	H	S	V	N	Z	C	7	
\$3E (\$5E)	予約										
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	8	
\$3C (\$5C)	OCR0B	タイマ/カウンタ0 比較レジスタ								52	
\$3B (\$5B)	GIMSK	INT1	INT0	PCIE	-	-	-	-	-	41	
\$3A (\$5A)	EIFR	INTF1	INTF0	PCIF	-	-	-	-	-	41	
\$39 (\$59)	TIMSK	TOIE1	OCIE1A	OCIE1B	-	ICIE1	OCIE0B	TOIE0	OCIE0A	74,53	
\$38 (\$58)	TIFR	TOV1	OCF1A	OCF1B	-	ICF1	OCF0B	TOV0	OCF0A	74,53	
\$37 (\$57)	SPMCSR	-	-	-	CTPB	RFLB	PGWRT	PGERS	SELFPRGEN	104	
\$36 (\$56)	OCR0A	タイマ/カウンタ0 比較レジスタ								52	
\$35 (\$55)	MCUCR	PUD	SM1	SE	SM0	ISC11	ISC10	ISC01	ISC00	34,21,40	
\$34 (\$54)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	25	
\$33 (\$53)	TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	51	
\$32 (\$52)	TCNT0	タイマ/カウンタ0								52	
\$31 (\$51)	OSCCAL	-	内蔵RC発振器 発振校正レジスタ								18
\$30 (\$50)	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	50	
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	70	
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	71	
\$2D (\$4D)	TCNT1H	タイマ/カウンタ1 上位バイト								72	
\$2C (\$4C)	TCNT1L	タイマ/カウンタ1 下位バイト								72	
\$2B (\$4B)	OCR1AH	タイマ/カウンタ1 比較レジスタ上位バイト								73	
\$2A (\$4A)	OCR1AL	タイマ/カウンタ1 比較レジスタ下位バイト								73	
\$29 (\$49)	OCR1BH	タイマ/カウンタ1 比較レジスタ上位バイト								73	
\$28 (\$48)	OCR1BL	タイマ/カウンタ1 比較レジスタ下位バイト								73	
\$27 (\$47)	予約										
\$26 (\$46)	CLKPR	CLKPCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	20	
\$25 (\$45)	ICR1H	タイマ/カウンタ1 捕獲レジスタ上位バイト								73	
\$24 (\$44)	ICR1L	タイマ/カウンタ1 捕獲レジスタ下位バイト								73	
\$23 (\$43)	GTCCR	-	-	-	-	-	-	-	PSR10	55	
\$22 (\$42)	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-	72	
\$21 (\$41)	WDTCR	WDFR	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	28	
\$20 (\$40)	PCMSK	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	41	
\$1F (\$3F)	予約										
\$1E (\$3E)	EEAR	EEPROMアドレスレジスタ (EEAR6~0)								12	
\$1D (\$3D)	EEDR	EEPROMデータレジスタ								12	
\$1C (\$3C)	EEDR	-	-	EEM1	EEM0	EERIE	EEMPE	EEPE	EERE	12	
\$1B (\$3B)	PORTA	-	-	-	-	-	PORTA2	PORTA1	PORTA0	39	
\$1A (\$3A)	DDRA	-	-	-	-	-	DDA2	DDA1	DDA0	39	
\$19 (\$39)	PINA	-	-	-	-	-	PINA2	PINA1	PINA0	39	
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	39	
\$17 (\$37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	39	
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	39	
\$15 (\$35)	GPOR2	汎用I/Oレジスタ2								15	
\$14 (\$34)	GPOR1	汎用I/Oレジスタ1								15	
\$13 (\$33)	GPOR0	汎用I/Oレジスタ0								15	
\$12 (\$32)	PORTD	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	39	
\$11 (\$31)	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	39	
\$10 (\$30)	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	39	
\$0F (\$2F)	USIDR	USI データレジスタ								97	
\$0E (\$2E)	USISR	USISIF	USIOIF	USIPF	USIDC	USICNT3	USICNT2	USICNT1	USICNT0	97	
\$0D (\$2D)	USICR	USISIE	USIOIE	USIWM1	USIWM0	USICS1	USICS0	USICLK	USITC	98	
\$0C (\$2C)	UDR	USART データレジスタ								87	
\$0B (\$2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM	87	
\$0A (\$2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	88	
\$09 (\$29)	UBRRL	USART ホールレートレジスタ下位 (UBRR7~0)								89	
\$08 (\$28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	100	
\$07 (\$27)	予約										
\$06 (\$26)	予約										
\$05 (\$25)	予約										
\$04 (\$24)	予約										
\$03 (\$23)	UCSRC	-	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	88	
\$02 (\$22)	UBRRH	-	-	-	-	USART ホールレートレジスタ上位 (UBRR11~8)				89	
\$01 (\$21)	DIDR	-	-	-	-	-	-	AIN1D	AIN0D	101	
\$00 (\$20)	予約										

注: 将来のデバイスとの共通性のため、予約ビットへ書く場合は0を書くべきです。予約I/Oメモリアドレスは決して書かれるべきではありません。

- アドレス範囲\$00~\$1FのI/OレジスタはCBIとSBI命令の使用で直接アクセス可能です。これらのレジスタの単一ビットはCBISとSBIS命令の使用によって検査できます。
- いくつかの状態ビットはそれらへ論理1を書くことによって解除(0)されます。他の多くのAVRと異なり、CBIとSBI命令は指定ビットだけ操作し、従ってこのような状態フラグを含むレジスタで使えることに注意してください。CBIとSBI命令は\$00~\$1Fのレジスタだけで動作します。



命令要約

ニーモニック	オペラント	意味	動作	フラグ	クロック
算術、論理演算命令					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,I	1
NEG	Rd	2の補数	$Rd \leftarrow \$00 - Rd$	I,T,H,S,V,N,Z,C	1
SBR	Rd,K	汎用レジスタの(複数)ビット設定(1)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
CBR	Rd,K	汎用レジスタの(複数)ビット解除(0)	$Rd \leftarrow Rd \text{ AND } (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レジスタの増加(+1)	$Rd \leftarrow Rd + 1$	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	$Rd \leftarrow Rd - 1$	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レジスタのゼロとマイナス検査	$Rd \leftarrow Rd \text{ AND } Rd$	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レジスタの全0設定(=\$00)	$Rd \leftarrow Rd \text{ EOR } Rd$	I,T,H,0,0,0,I,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	$Rd \leftarrow \$FF$	I,T,H,S,V,N,Z,C	1
分岐命令					
RJMP	k	相対分岐	$PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	2
IJMP		Zレジスタ間接分岐	$PC \leftarrow Z$	I,T,H,S,V,N,Z,C	2
RCALL	k	相対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	3
ICALL		Zレジスタ間接サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow Z$	I,T,H,S,V,N,Z,C	3
RET		サブルーチンからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
RETI		割り込みからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
CPSE	Rd,Rr	汎用レジスタ間比較、一致でスキップ	Rd=Rrなら、 $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	$Rd - Rr$	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	$Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	$Rd - K$	I,T,H,S,V,N,Z,C	1
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ	Rr(b)=0なら、 $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBRS	Rr,b	汎用レジスタのビットが設定(1)でスキップ	Rr(b)=1なら、 $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ	P(b)=0なら、 $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBIS	P,b	I/Oレジスタのビットが設定(1)でスキップ	P(b)=1なら、 $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
BRBS	s,k	ステータスフラグが設定(1)で分岐	SREG(s)=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRBC	s,k	ステータスフラグが解除(0)で分岐	SREG(s)=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BREQ	k	一致で分岐	Z=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRNE	k	不一致で分岐	Z=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCS	k	キャリーフラグが設定(1)で分岐	C=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCC	k	キャリーフラグが解除(0)で分岐	C=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRSH	k	符号なしの \geq で分岐	C=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしの $<$ で分岐	C=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	N=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	N=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの \geq で分岐	(N EOR V)=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLT	k	符号付きの $<$ で分岐	(N EOR V)=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	H=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	H=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	T=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	T=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	V=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	V=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	I=1なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	I=0なら、 $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ
 b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)

ニーモニック	オペランド	意味	動作	フラグ	クロック
データ移動命令					
MOV	Rd,Rr	汎用レジスタ間の複写	$Rd \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
MOVW	Rd,Rr	汎用レジスタ対間の複写	$Rd+1:Rd \leftarrow Rr+1:Rr$	I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	$Rd \leftarrow K$	I,T,H,S,V,N,Z,C	1
LD	Rd,X	Xレジスタ間接での取得	$Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,X+	事後増加付きXレジスタ間接での取得	$Rd \leftarrow (X), X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-X	事前減少付きXレジスタ間接での取得	$X \leftarrow X - 1, Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y	Yレジスタ間接での取得	$Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上 (任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上 (事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
ビット関係命令					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	I,T,H,S,V,1,I,I	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLC		キャリーフラグを解除(0)	$C \leftarrow 0$	I,T,H,S,V,N,Z,0	1
SEN		負フラグを設定(1)	$N \leftarrow 1$	I,T,H,S,V,1,Z,C	1
CLN		負フラグを解除(0)	$N \leftarrow 0$	I,T,H,S,V,0,Z,C	1
SEZ		ゼロフラグを設定(1)	$Z \leftarrow 1$	I,T,H,S,V,N,1,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,0,C	1
SEI		全割り込み許可	$I \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLI		全割り込み禁止	$I \leftarrow 0$	0,T,H,S,V,N,Z,C	1
SES		符号フラグを設定(1)	$S \leftarrow 1$	I,T,H,1,V,N,Z,C	1
CLS		符号フラグを解除(0)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	$V \leftarrow 1$	I,T,H,S,1,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	$V \leftarrow 0$	I,T,H,S,0,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,1,H,S,V,N,Z,C	1
CLT		一時フラグを解除(0)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,1,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
MCU制御命令					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグWIRE機能専用	I,T,H,S,V,N,Z,C	N/A

注文情報

速度(MHz)	電源電圧	注文符号 (注2,4)	外囲器	動作範囲
10 (注3)	1.8~5.5V	ATtiny2313V-10PU	20P3	工業用 (-40°C~85°C)
		ATtiny2313V-10SU	20S	
		ATtiny2313V-10SUR		
		ATtiny2313V-10MU	20M1	
		ATtiny2313V-10MUR		
20 (注3)	2.7~5.5V	ATtiny2313-20PU	20P3	
		ATtiny2313-20SU	20S	
		ATtiny2313-20SUR		
		ATtiny2313-20MU	20M1	
		ATtiny2313-20MUR		

注: このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

注2: 有害物質使用制限に関する欧州指令(RoHS指令)適合の鉛フリー製品。またハロゲン化合物フリーで完全に安全です。

注3: 速度と電源電圧の関係については119頁の図82と図83をご覧ください。

注4: 符号は以下を示します。

- U : 半光沢錫
- R : テープとリール

外囲器形式

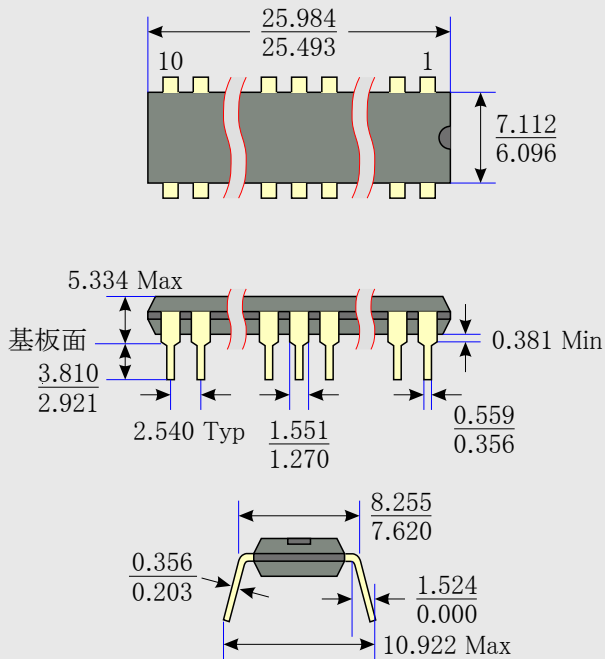
20P3	20ピン 300mil幅 プラスティック2列直線外囲器 (PDIP)
20S	20リード 300mil幅 プラスティック小外形外囲器 (SOIC)
20M1	20パッド 4×4×0.8mm 0.5mmピッチ 4方向平板リードなし/小リード枠外囲器 (QFN/MLF)

外圍器情報

20P3

20ピン 300mil幅 プラスティック2列直線外圍器 (PDIP)

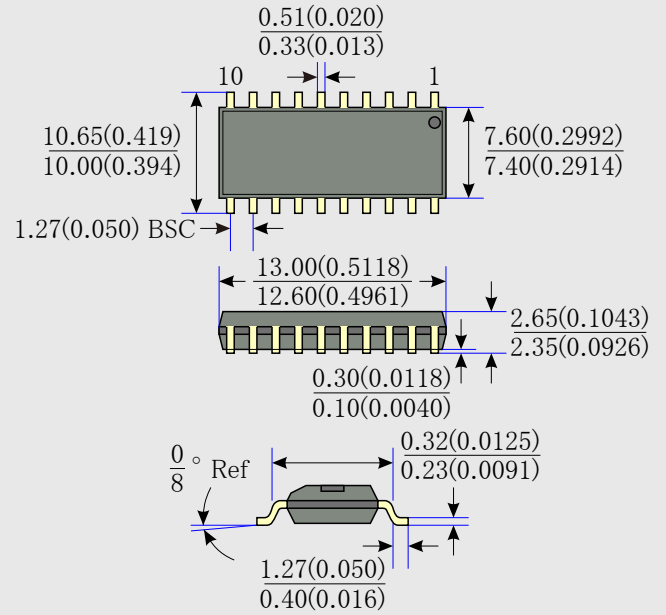
寸法: mm
JEDEC規格 MS-001 AD



20S

20リード 300mil幅 プラスティック小外形外圍器 (SOIC)

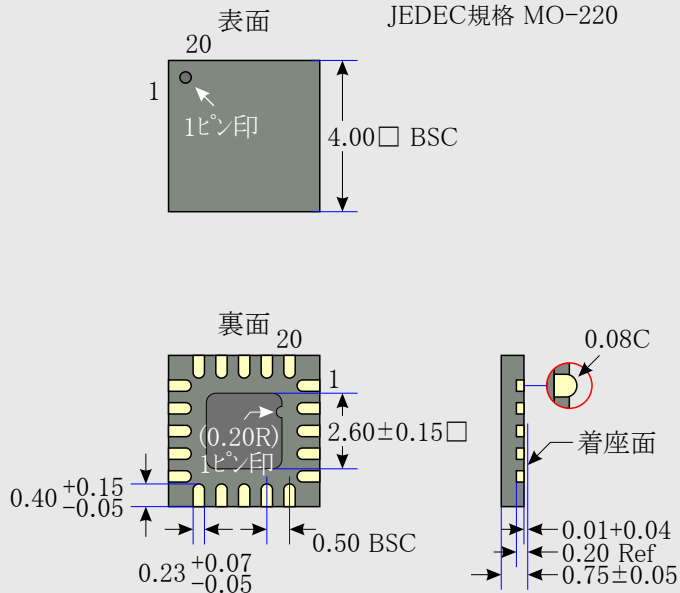
寸法: mm、()内はインチ
JEDEC規格 MS-013 AC



20M1

20パット 0.5mmピッチ 4方向平板リードなし/小リード枠外圍器 (QFN/MLF)

寸法: mm
JEDEC規格 MO-220



データシート改訂履歴

本章内の参照頁番号は本文書が参照されることに注意してください。本章内の改訂番号は文書の改訂番号を参照してください。

改訂2543A – 2003年9月

1. 初版

改訂2543B – 2003年9月

1. 1頁の「特徴」でUARTをUSARTに修正、動作速度と消費電力を更新
2. 2頁の「ピン配置」を更新
3. 119頁の表79.と120頁の表80.を更新
4. 115頁の「直列プログラミング手順」で項目5.を更新
5. 118頁の「電気的特性」を更新
6. 119頁の図82.を更新、図83.を追加
7. 141頁の「レジスタ要約」でSFIORをGTCCRに変更
8. 144頁の「注文情報」を更新
9. 148頁の「障害情報」に新規障害を追加

改訂2543C – 2003年12月

1. 18頁の「校正付き内蔵RC発振器」を更新

改訂2543D – 2004年3月

1. 16頁の表2.を更新
2. 26頁からの「ウォッチドッグ タイマ」を置換
3. 119頁に「最高速と動作電圧」を追加
4. 115頁の「直列プログラミング手順」を更新
5. 138頁の暫定図136.でmAをμAに変更
6. 144頁の「注文情報」を更新(MLF外圍器選択を削除)
7. 145頁の「外圍器情報」で「20P3」を更新
8. Cコード例を更新
9. 自己プログラミング許可のSPMENをSELFPRGENに改名

改訂2543E – 2004年4月

1. 最大動作速度を変更
・12MHzを10MHzへ ・24MHzを20MHzへ
2. 2頁の「ピン配置」で図を更新
3. 144頁の「注文情報」を更新
4. 119頁に「最高速と動作電圧」を更新
5. 121頁からの「代表特性」を更新

改訂2543F – 2004年8月

1. 1頁の「特徴」を更新
2. 35頁の「ポートBの交換機能」を更新
3. 106頁の「校正バイト」を更新
4. 107頁の表68.と表69.を「ページ容量」節内へ移動
5. 109頁の「並列プログラミング動作への移行」を更新
6. 115頁の「直列プログラミング手順」を更新
7. 116頁の表77.を更新
8. 118頁の「DC特性」を更新
9. 121頁からの「代表特性」を更新
10. PCINT15をPCINT7、EEMWEをEEMPE、EEWEをEEPEへ変更

改訂2543G – 2004年10月

1. 1頁の「特徴」を更新
2. 2頁の「ピン配置」を更新
3. 144頁の「注文情報」を更新

4. 145頁の「外圍器情報」を更新
5. 148頁の「障害情報」を更新

改訂2543H – 2005年2月

1. 18頁の表6、107頁の表67、119頁の表79、120頁の表80を更新
2. 16頁の「既定のクロック元」でCKSEL既定値を8MHzに変更
3. 144頁の「注文情報」を更新
4. 119頁の「プログラミング動作への移行」、110頁の「フラッシュメモリ書き込み」、111頁の「EEPROM書き込み」を更新
5. MLCF外圍器を“4方向平板リードなし/小リード枠外圍器(QFN/MLF)”に更新

改訂2543I – 2006年4月

1. 誤植修正
2. 2頁の「ピン配置」を更新
3. 5頁に「資料」を追加
4. 16頁の「既定のクロック元」を更新
5. 19頁の「128kHz内部発振器」を更新
6. 21頁の「電力管理と休止形態」を更新
7. 16頁の表3、21頁の表13、22頁の表14、28頁の表18、40頁の表30、117頁の表78を更新
8. 40頁の「外部割り込み」を更新
9. 41頁の「ビット7～0 – PCINT7～0 : ピン変化割り込み7～0許可」を更新
10. 100頁の「ビット6 – ACBG : 内部基準電圧選択」を更新
11. 106頁の「校正バイト」を更新
12. 118頁の「DC特性」を更新
13. 141頁のレジスタ要約を更新
14. 144頁の「注文情報」を更新
15. OCNxA, OCNxB, OCF1xAの存在を各々OCFnxA, OCFnxB, OCF1xAへ変更

改訂2543J – 2009年11月

1. 雛形更新
2. デバイス状況を“新規設計に推奨されません”に変更
3. 8頁の「スタックポインタ」を更新
4. 21頁の「表13. 休止形態種別選択」を更新
5. 107頁の「校正バイト」を(1バイトの校正データに)更新

改訂2543K – 2010年3月

1. 148頁の「障害情報」に改訂Cデバイスの“既知の障害はありません”を追加

改訂2543L – 2010年8月

1. 表紙から“新規設計に推奨されません”の文章を削除
2. 144頁の「注文情報」にテープとリール品の番号を追加
3. データシート改訂履歴で文献番号不一致を修正

改訂2543M – 2016年10月

1. 表紙からデバイス状況の“暫定”を削除
2. 5頁に「データ保持力」を追加
3. 18頁の表7と19頁の表11を更新
4. 23頁の図14を更新 (訳注:実際には無変化)
5. リセット特性表を23頁の「リセット元」項から「電気的特性」章で117頁の新「システムとリセットの特性」項へ移動して更新
6. 28頁の「ウォッチドッグ タイム制御/状態レジスタ – WDTCSR」表題を更新
7. 107頁の表67を更新
8. 113頁の「識別バイト読み出し」を更新
9. 115頁の図78を更新
10. 136頁の図128を更新
11. 裏表紙を更新

障害情報

この章の改訂番号はATtiny2313デバイスの改訂版を参照してください。

改訂A

改訂A版は試供されていません。

改訂B

- ・ 単独消去操作後の不正値読み込み B
- ・ 並列プログラミングが動作しない B
- ・ ウォッチドッグ タイマ割り込みが禁止 B
- ・ 1.9V未満でのEEPROM書き込み不可 B

改訂C

既知の障害はありません。

1. 単独消去操作後の不正値読み込み (B)

2.7V以下の電圧印加に於いて、消去操作のみで消去されたEEPROM位置はプログラム(\$00)として読むかもしれません。

対策/対処

単独消去後にEEPROM位置を読む必要がある場合、その位置を消去するために値として\$FFの**非分離書き込み**を使ってください。どんな場合も単独書き込み操作は予定どおりに使えます。従って消去された位置がプログラミング以前に読めない限り、特別な考慮の必要はありません。

2. 並列プログラミングが動作しない (B)

並列プログラムは正常に機能しません。この理由は次の動作種別の1つが選ばれた場合にデバイスの再プログラミングが不可能になるからです。

- ・ 実装書き込み(直列プログラミング)禁止 (SPIEN=非プログラム(1))
- ・ リセット禁止 (RSTDISBL=プログラム(0))

対策/対処

直列プログラミングは正しく動きます。上の2つの動作種別を避ける事により、デバイスは直列で再プログラミングできます。

3. ウォッチドッグ タイマ割り込みが禁止 (B)

ウォッチドッグ タイマ割り込み要求フラグ(WDIF)が次の計時完了発生に先立って解除(0)されなければウォッチドッグが禁止され、割り込み要求フラグは自動的に解除(0)されます。これは割り込みのみの動作種別にだけ当てはまります。ウォッチドッグ計時完了に続く割り込みでウォッチドッグがデバイスリセットに設定されれば、デバイスは正しく動作します。

対策/対処

次のウォッチドッグ計時完了が起こる前に、先行する計時完了発生での処理が常に十分な時間であることに注意してください。これは十分に長い計時完了周期を選ぶことによって行われます。

4. 1.9V未満でのEEPROM書き込み不可 (B)

1.9V未満のVCCでのEEPROM書き込みは失敗するかもしれません。

対策/対処

VCCが1.9V未満の時はEEPROMを書かないでください。

目次

特徴	1	タイマ/カウンタのタイミング	49
ピン配置	2	8ビット タイマ/カウンタ0用レジスタ	50
概要	3	タイマ/カウンタ0とタイマ/カウンタ1の前置分周器	54
構成図	3	16ビット タイマ/カウンタ1	56
ピン説明	4	概要	56
一般情報	5	16ビット レジスタのアクセス	58
資料	5	タイマ/カウンタのクロック	60
コード例	5	計数器部	60
データ保持力	5	捕獲入力部	61
AVR CPU コア	6	比較出力部	62
序説	6	比較一致出力部	64
構造概要	6	動作種別	65
ALU (Arithmetic Logic Unit)	6	タイマ/カウンタのタイミング	69
ステータスレジスタ	7	16ビット タイマ/カウンタ1用レジスタ	70
汎用レジスタファイル	8	USART	75
スタックポインタ	8	概要	75
命令実行タイミング	9	クロック生成	77
リセットと割り込みの扱い	9	フレーム形式	78
AVR ATtiny2313のメモリ	11	USARTの初期化	79
実装書き換え可能なプログラム用フラッシュメモリ	11	USARTのデータ送信	80
データ用SRAMメモリ	11	USARTのデータ受信	81
データ用EEPROMメモリ	12	非同期受信	84
I/Oメモリ (レジスタ)	15	複数プロセッサ通信動作	86
システム クロックとクロック選択	16	USART用レジスタ	87
クロックシステムとその配給	16	ポーレート設定例	90
クロック元	16	多用途直列インターフェース (USI)	92
既定のクロック元	16	概要	92
クリスタル用発振器	17	機能説明	93
校正付き内蔵RC発振器	18	USIでの代替使用	96
外部クロック信号	19	USI用レジスタ	97
128kHz内部発振器	19	アナログ比較器	100
システムクロック前置分周器	20	デバッグWIRE内蔵デバッグ機能	102
電力管理と休止形態	21	特徴	102
アイドル動作	21	概要	102
パワーダウン動作	21	物理インターフェース	102
スタンバイ動作	22	ソフトウェア中断点	102
消費電力の最小化	22	デバッグWIREの制限	102
システム制御とリセット	23	デバッグWIRE関連のI/Oメモリ内レジスタ	102
内部基準電圧	25	フラッシュの自己プログラミング	103
ウォッチドッグタイマ	26	自己プログラミングでのフラッシュアドレス指定	103
割り込み	29	メモリプログラミング	106
ATtiny2313の割り込みベクタ	29	プログラムメモリとデータメモリ用施錠ビット	106
入出力ポート	30	ヒューズビット	106
序説	30	識票バイト	107
標準デジタル入出力としてのポート	30	校正バイト	107
交換ポート機能	33	ページ容量	107
I/Oポート用レジスタ	39	並列プログラミング	108
外部割り込み	40	並列プログラミング手順	109
ピン変化割り込みタイミング	40	直列プログラミング	115
8ビット タイマ/カウンタ0 (PWM)	42	電气的特性	118
概要	42	絶対最大定格	118
タイマ/カウンタのクロック	43	DC特性	118
計数器部	43	外部クロック特性	119
比較出力部	44	最高速と動作電圧	119
比較一致出力部	45	システムとリセットの特性	120
動作種別	46	代表特性	121



活動動作消費電流	121
アイドル動作消費電流	123
パワーダウン動作消費電流	126
スタンバイ動作消費電流	126
ピンプルアップ	127
ピン駆動能力	128
ピン 閾値とヒステリシス	132
低電圧検出器(BOD)閾値と アナログ比較器変位(オフセット)	135
内部発振器周波数	136
周辺機能部消費電流	139
リセット消費電流とリセット パルス幅	140
レジスタ要約	141
命令要約	142
注文情報	144
外圍器情報	145
データシート改訂履歴	146
障害情報	148

Atmel®, Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®, AVR®とその他は米国及び他の国に於けるAtmel Corporationの登録商標または商標です。他の用語と製品名は一般的に他の商標です。

お断り: 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイトに表示する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえAtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

安全重視、軍用、車載応用のお断り: Atmel製品はAtmelが提供する特別に書かれた承諾を除き、そのような製品の機能不全が著しく人に危害を加えたり死に至らしめることがかなり予期されるどんな応用(“安全重視応用”)に対しても設計されず、またそれらとの接続にも使用されません。安全重視応用は限定なしで、生命維持装置とシステム、核施設と武器システムの操作の装置やシステムを含みます。Atmelによって軍用等級として特に明確に示される以外、Atmel製品は軍用や航空宇宙の応用や環境のために設計も意図もされていません。Atmelによって車載等級として特に明確に示される以外、Atmel製品は車載応用での使用のために設計も意図もされていません。

© HERO 2022.

本データシートはAtmelのATtiny2313英語版データシート(改訂2543M-10/2016)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力は、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。