

## 特徴

- 高性能、低消費AVR<sup>®</sup> 8ビット マイクロ コントローラ
- 進化したRISC構造
  - 強力な123命令(多くは1周期実行)
  - 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
- 高耐久不揮発性メモリ部
  - 実装自己書き換え可能な4K/8Kバイト(2K/4K語)フラッシュメモリ内蔵
  - 64/64バイトのEEPROM
  - 256/512バイトの内蔵SRAM
  - 書き換え回数: 10,000/フラッシュ, 100,000/EEPROM
  - データ保持力: 20年/85°C, 100年/25°C
  - 自己プログラミングフラッシュとEEPROMデータ保護用の設定可能な施錠機能
- 内蔵周辺機能
  - 独立した前置分周器と比較機能付き1つの8ビットタイマ/カウンタ
  - 独立した前置分周器、比較、捕獲機能付き1つの16ビットタイマ/カウンタ
  - 6または8チャンネルの10ビットA/D変換器
  - 主装置/従装置動作SPI直列インターフェース
  - バイト対応2線直列インターフェース(Philips I<sup>2</sup>C互換)
  - 設定可能な専用発振器付きウォッチドッグタイマ
  - アナログ比較器
  - ピン変化での割り込みと起動復帰
- 特殊マイクロコントローラ機能
  - デバッグWIRE内蔵デバッグシステム
  - SPIポート経由の実装書き込み(ISP)
  - 電源ONリセット回路と設定可能な低電圧検出器(BOD)
  - 校正可能な内蔵発振器
  - 外部及び内部の割り込み
  - アイドル、A/D変換雑音低減、パワーダウンの3つの低消費動作
  - チップ上の温度感知器
- I/Oと外圍器
  - 24ビットの設定可能なI/O
    - 28ピンPDIP、28ピンQFN
  - 28ビットの設定可能なI/O
    - 32リードTQFP、32ピンQFN、32球UFPGA
- 動作電圧
  - 1.8~5.5V
- 動作温度
  - -40~85°C
- 動作速度
  - 0~4MHz/1.8~5.5V
  - 0~8MHz/2.7~5.5V
  - 0~12MHz/4.5~5.5V
- 代表消費電力
  - 240μA (活動動作、1MHz, 1.8V)
  - 0.1μA (パワーダウン動作)



8ビット AVR<sup>®</sup>  
 マイクロ コントローラ  
 実装書き換え可能な  
 4/8Kバイト  
 フラッシュメモリ内蔵

ATtiny48  
 ATtiny88

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. 8008H-04/11, 8008HJ6-10/22

# 1. ピン配置

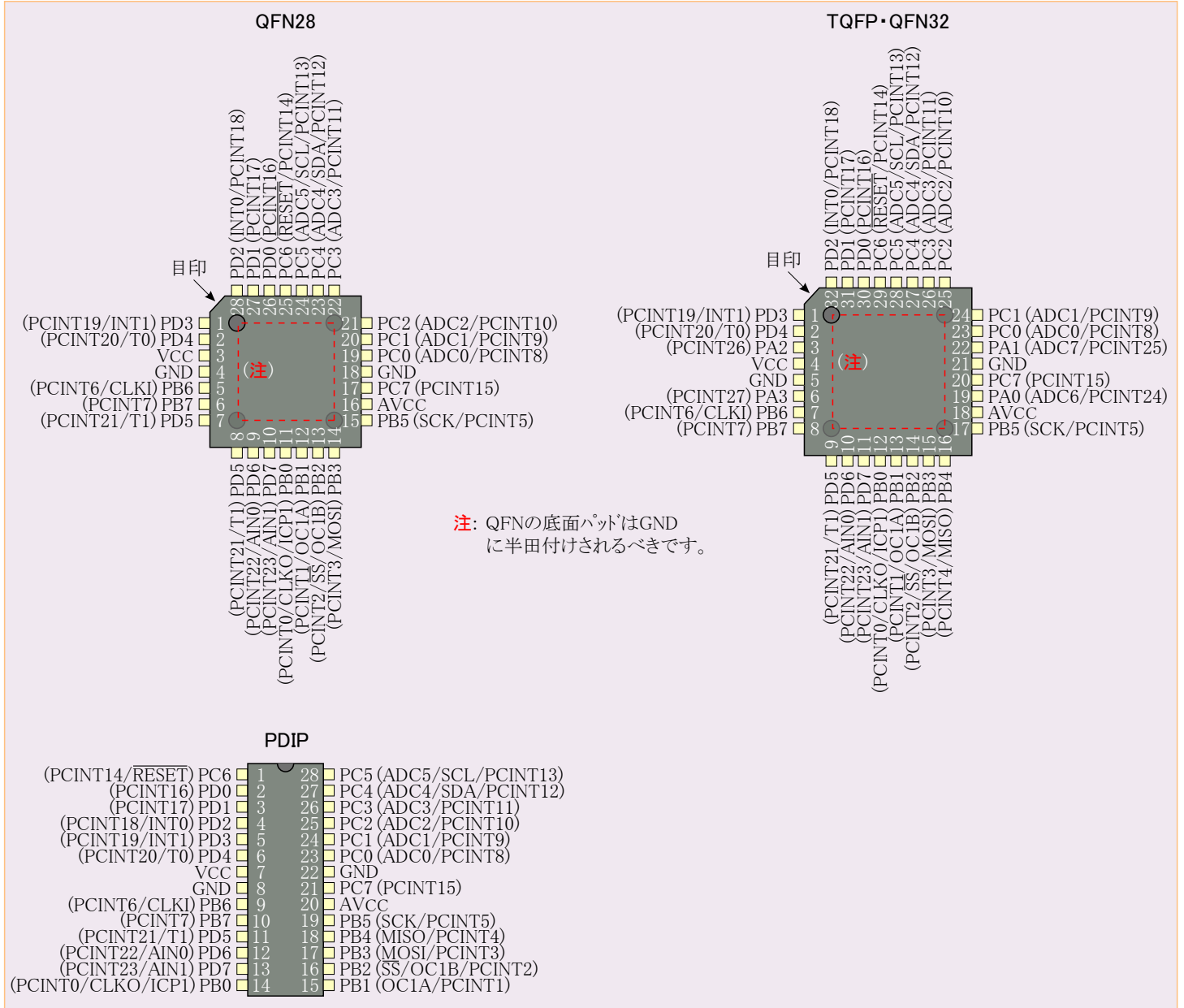


表1-1. 32UFBGALピン配列 (上面視、190頁参照)

	1	2	3	4	5	6
<b>A</b>	PD2	PD1	PC6	PC4	PC2	PC1
<b>B</b>	PD3	PD4	PD0	PC5	PC3	PC0
<b>C</b>	GND	PA2			PA1	GND
<b>D</b>	VCC	PA3			PC7	PA0
<b>E</b>	PB6	PD6	PB0	PB2	AVCC	PB5
<b>F</b>	PB7	PD5	PD7	PB1	PB3	PB4

## 1.1. ピン概要

### 1.1.1. VCC

デジタル電源ピン。

### 1.1.2. AVCC

AVCCは分離I/OピンとA/D変換器用供給電圧(電源)ピンです。例えA/D変換が使われなくても、このピンは外部的にVCCへ接続されるべきです。A/D変換が使われる場合、111頁の「[アナログ雑音低減技術](#)」で記述されるように、このピンは低域通過濾波器を通してVCCへ接続されることが推奨されます。

PC7,PC5~0と(32ピン外圍器での)PA1~0ピンはそれらの供給電圧をAVCCから受けます。他の全てのI/Oピンはそれらの供給電圧をVCCから取り入れます。

### 1.1.3. GND

接地ピン。

### 1.1.4. PA3~PA0 - (ポートA)

ポートAは32ピンTQFPと32ピンQFN/MLF外圍器での(ビット毎に選ばれる)内蔵プルアップ抵抗付きの4ビットの双方向入出力ポートです。PA3~0出力緩衝部は共に吐き出し/吸い込み能力の対称駆動特性です。入力として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き、クロックが動作していなくても、ポートAピンはHi-Zになります。

このポートは32リードTQFP、32ピンQFN、32球UFBGA外圍器でだけ利用可能です。

### 1.1.5. PB7~PB0 - (ポートB)

ポートBは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの8ビットの双方向入出力ポートです。ポートB出力緩衝部は共に吐き出し/吸い込み能力の対称駆動特性です。入力として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き、クロックが動作していなくても、ポートBピンはHi-Zになります。

[クロック選択ヒューズ設定](#)に依存し、PB6は内部クロック操作回路の入力として使えます。

ポートBの各特殊機能は43頁の「[ポートBの交換機能](#)」で詳しく述べられます。

### 1.1.6. PC7,PC5~PC0 - (ポートC)

ポートCは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの8ビットの双方向入出力ポートです。PC7とPC5~0出力緩衝部は共に吐き出し/吸い込み能力の対称駆動特性です。入力として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C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き、クロックが動作していなくても、ポートCピンはHi-Zになります。

### 1.1.7. PC6/RESET

RSTDISBLヒューズがプログラム(0)されると、PC6はI/Oピンとして使われます。PC6の電気的特性がポートCの他のピンのそれらと異なることに注意してください。

RSTDISBLヒューズが非プログラム(1)の場合、PC6はリセット入力として使われます。クロックが動作していなくても、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137頁の表22-3.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。

ポートCの各特殊機能は45頁の「[ポートCの交換機能](#)」で詳しく述べられます。

### 1.1.8. PD7~PD0 - (ポートD)

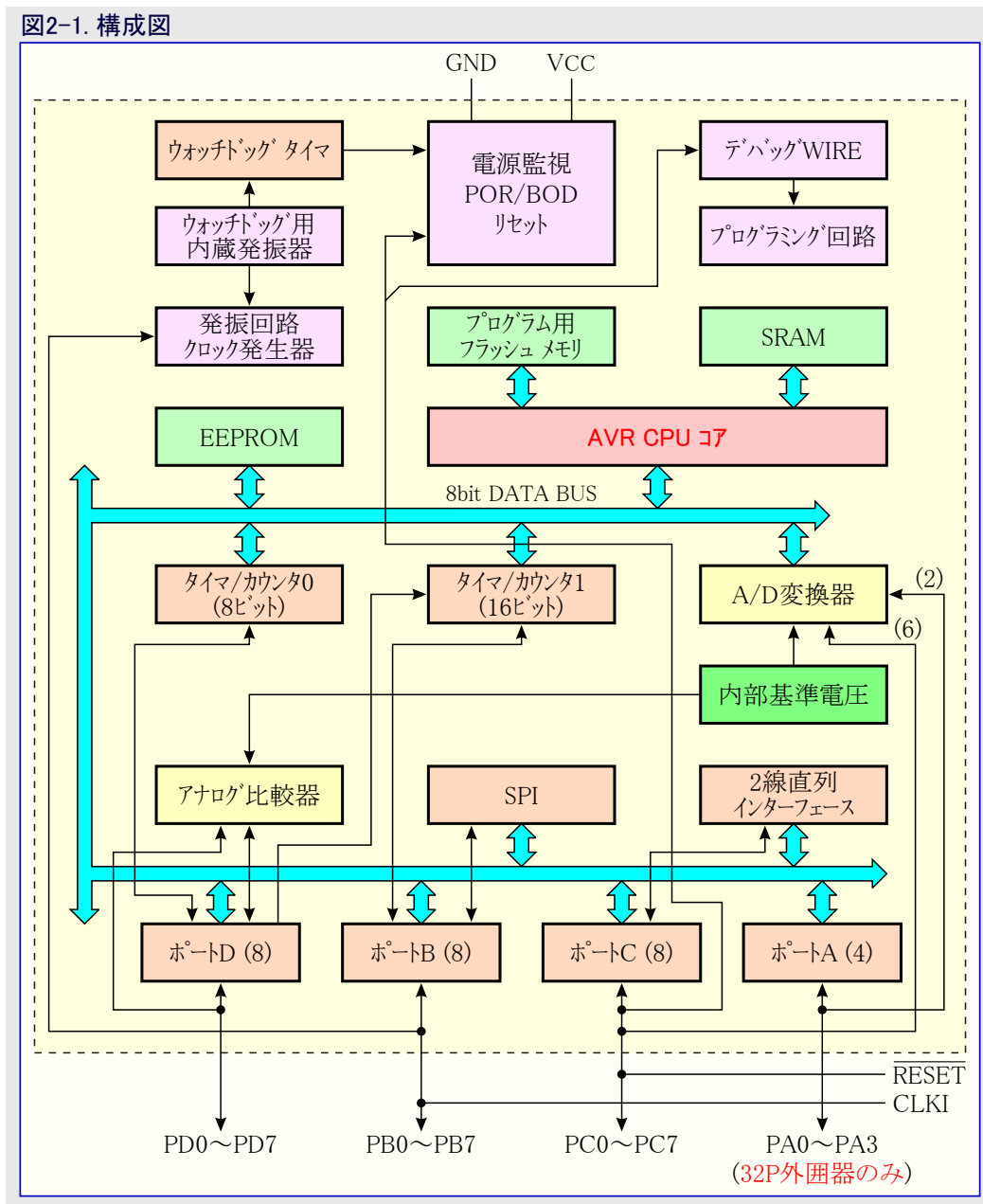
ポートDは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの8ビットの双方向入出力ポートです。PD7~4出力緩衝部は共に吐き出し/吸い込み能力の対称駆動特性で、一方PD3~0出力緩衝部は高い吸い込み能力を持ちます。入力としてプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートDピンには吐き出し電流が流れます。リセット条件が有効になるとき、クロックが動作していなくても、ポートDピンはHi-Zになります。

ポートDの各特殊機能は47頁の「[ポートDの交換機能](#)」で詳しく述べられます。

## 2. 概要

ATtiny48/88はAVR RISC構造の低消費CMOS 8ビットマイクロコントローラです。1周期で実行する強力な命令によって、ATtiny48/88は電力消費対処理速度の最適化をシステム設計者に許す、MHzあたり1 MIPSに達する単位処理量を成し遂げます。

### 2.1. 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を兼ね備えています。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10倍以上の単位処理量向上効果があります。

ATtiny48/88は以下の特徴を提供します。

- 4/8Kバイトの実装書き換え可能なフラッシュメモリ
- 64/64バイトのEEPROM
- 256/512バイトのSRAM
- 24(28ピン外圍器), 28(32ピン外圍器)本の汎用入出力線
- 32個の汎用レジスタ
- 比較動作を持つ柔軟な2つのタイマ/カウンタ
- 内部及び外部割り込み
- バイト志向の2線直列インターフェース

- SPI直列ポート
- 6(28ピン外圍器), 8(32ピン外圍器)チャンネルの10ビット A/D変換器
- 設定変更可能な内部発振器付きウォッチドッグ タイマ
- ソフトウェアで選べる3つの低消費動作機能

本デバイスは節電用に以下の動作形態を含みます。

- **アイドル動作** : CPUを停止しますが、タイマ/カウンタ、2線直列インターフェース、SPIポート、割り込み機能は有効で動作を継続します。
- **A/D変換雑音低減動作** : A/D変換器を除く周辺機能とCPUを停止し、A/D変換中の切り替え雑音を最小にします。
- **パワーダウン動作** : レジスタはそれらの内容を保ち、次の外部割り込みかハードウェア リセットまで全てのチップ機能が禁止(無効)されます。

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)可能なプログラム用フラッシュ メモリは規定の不揮発性メモリ書き込み器、SPI直列インターフェース経由、AVRコア上ポートプログラムの実行によって再書き込みができます。ポートプログラムはフラッシュ メモリ内の応用プログラムの読み込みによどのインターフェースでも使えます。モノリシック チップ上の自己実装書き換え可能なフラッシュ メモリと、8ビットRISC型CPUの組み合わせによるAtmel ATtiny48/88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。

ATtiny48/88 AVRはCコンパイラ、マクロ アセンブラ、デバッガ、シミュレータ、評価キットを含む専用のプログラム及びシステム開発ツールで支援されます。

## 2.2. ATtiny48とATtiny88間の違い

右の表2-1.で要約されるように、ATtiny48とATtiny88はメモリ容量だけが異なります。

表2-1. メモリ容量対比表

デバイス名	フラッシュ メモリ	EEPROM	SRAM
ATtiny48	4Kバイト	64バイト	256バイト
ATtiny88	8Kバイト	64バイト	512バイト

## 3. 一般情報

### 3.1. 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。

### 3.2. コード例について

この文書はデバイスの様々な部分の使用法を手短か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。

拡張I/O領域に配置したI/Oレジスタに対し、**IN**, **OUT**, **SBIS**, **SBIC**, **CBI**, **SBI**命令は拡張I/O領域へのアクセスを許す命令に置き換えられなければなりません。代表的には**SBRS**, **SBRC**, **SBR**, **CBR**命令と組み合わせた**LDS**, **STS**命令です。

### 3.3. 容量性接触感知

Atmel QTouchライブラリはAtmel AVRマイクロ コントローラ上の接触感知インターフェースのための使い易い解決策を提供します。QTouchライブラリはQTouch®とQMatrix®採取法用の支援を含みます。

接触感知は接触チャンネルと感知器を定義することで応用プログラミング インターフェース(API)を用いてQTouchライブラリをリンクすることによってどの応用にも容易に追加されます。そして応用はチャンネル情報を取得して接触感知器の状態を決めるためにAPIを呼び出します。

QTouchライブラリは無料でAtmelのウェブサイトからダウンロードすることができます。より多くの情報と実装の詳細についてはAtmelのウェブサイトからも入手可能なQTouchライブラリ使用者の手引きを参照してください。

### 3.4. 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。

### 3.5. お断り

本データシート内で示された代表値はシミュレーションと同じ製法技術で製造された他のAVRマイクロ コントローラの特性に基いています。

## 4. AVR CPU コア

### 4.1. 序説

ここでは一般的なAVRコア構造について説明します。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。

### 4.2. 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。殆どのAVR命令は16ビット語(ワード)形式ですが、32ビット命令もあります。いくつかのデバイスが命令一式の部分だけを実装するため、実際の命令一式は変化します。

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。

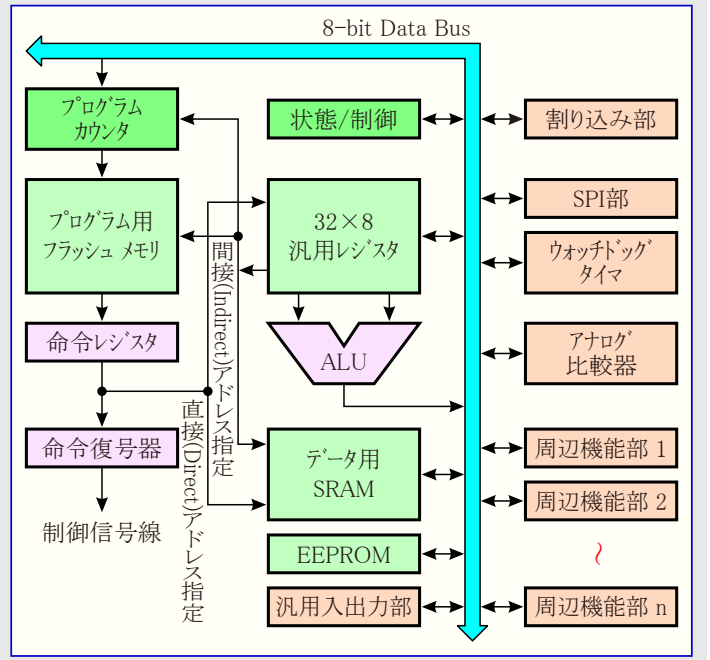
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(I)ビットがあります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタ、SPI、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。加えてATtiny48/88にはST/STS/STDとLD/LDS/LDD命令だけ使えるSRAM内の\$60~\$FFに拡張I/O空間があります。

### 4.3. ALU (Arithmetic Logic Unit)

高性能なAVRのALUは32個の全汎用レジスタとの直結で動作します。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。より多くの情報については外部文書の「AVR命令一式」と「命令要約」章をご覧ください。

図4-1. AVR構造構成図



## 4.4. ステータス レジスタ

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。

## 4.5. 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4-2.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4-2.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い場合、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4-2. 汎用レジスタ構成図

7	0	アドレス	特殊機能レジスタ
R0		\$00	
R1		\$01	
R2		\$02	
⋮			
R13		\$0D	
R14		\$0E	
R15		\$0F	
R16		\$10	
R17		\$11	
⋮			
R26		\$1A	Xレジスタ 下位8ビット
R27		\$1B	上位8ビット
R28		\$1C	Yレジスタ 下位8ビット
R29		\$1D	上位8ビット
R30		\$1E	Zレジスタ 下位8ビット
R31		\$1F	上位8ビット

### 4.5.1. 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4-3.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4-3. 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)

## 4.6. スタック ポインタ

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(SPHとSPL)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びることに注意してください。これはPUSH命令がスタック ポインタ値を減らし、POP命令が増すという意味です。

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 メモリの領域を指し示します。こ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。

このポインタはPUSH命令でデータがスタックに置かれる時に-1され、POP命令でデータが取得される時に+1されます。これはサブルーチン呼び出しや割り込み処理ルーチンへ飛ぶことによって戻りアドレスがスタックに置かれる時に-2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)によって戻りアドレスがスタックから取得される時に+2されます。

AVRのスタック ポインタは代表的に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。スタック ポインタの幅と実装されるビット数はデバイス依存です。いくつかのAVRデバイスではSPLだけを用いて全てのデータ メモリをアドレス指定することができます。その場合、SPHレジスタは実装されません。

スタック ポインタはI/Oレジスタ領域(SRAMの最低アドレスである最小値)以上を指示するように設定されなければなりません。11頁の表5-2.をご覧ください。

## 4.7. 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4-4.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、機能対電源部に関する好結果と対応する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4-5.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ドを使う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4-4. 命令の取得と実行の並列動作

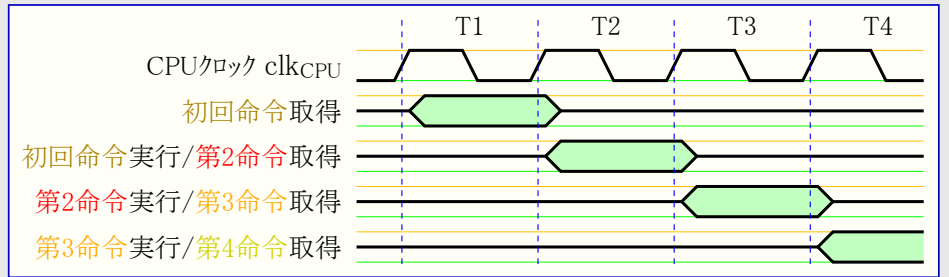
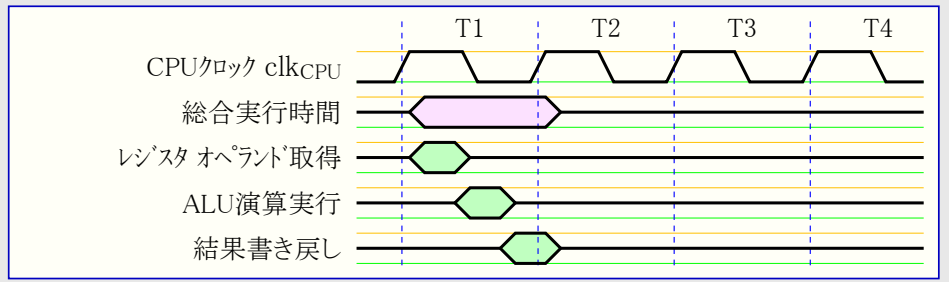


図4-5. 1周期ALU命令



## 4.8. 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に、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。LB2またはLB1施錠ビットがプログラム(0)されると、プログラムカウンタ値によっては割り込みが自動的に禁止されるかもしれません。この特質はソフトウェア保護を改善します。詳細については121頁の「施錠ビット、ヒューズビット、デバイス識票」章をご覧ください。

既定でのプログラムメモリ空間の最下位アドレスは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31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31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 */
```

**注:** 5頁の「コード例について」をご覧ください。

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。

**注:** 5頁の「コード例について」をご覧ください。

## 4.8.1. 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。

## 4.9. コア関係レジスタ

### 4.9.1. SPH,SPL (SP) – スタック ポインタ (Stack Pointer)

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	–	–	–	–	–	–	(SP9)	SP8	SPH
Read/Write	R	R	R	R	R	R	R/W	R/W	
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	

### 4.9.2. SREG – ステータス レジスタ (Status Register)

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。

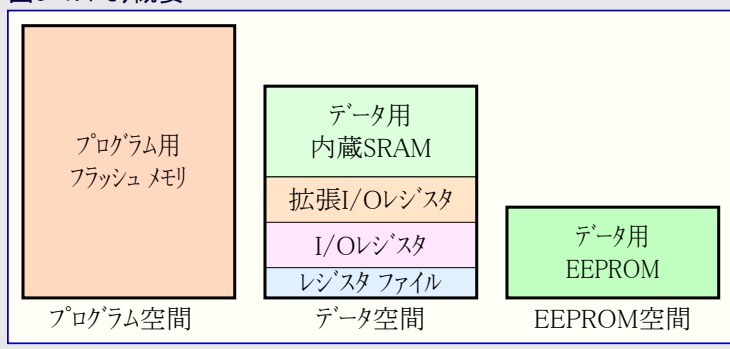
(訳補) 内蔵SRAMはATtiny48が256バイト(\$0100~\$01FF)、ATtiny88が512バイト(\$0100~\$02FF)です。従ってATtiny48ではSPHのSP9が利用できません。RAMENDはATtiny48が\$01FF(0000 0001 1111 1111)、ATtiny88が\$02FF(0000 0010 1111 1111)になります。

## 5. メモリ

AVR構造は独立したアドレス空間に各々のメモリ形式を配置し、プログラムメモリとデータメモリを区別します。実行可能コードは不揮発性プログラムメモリ(フラッシュ)に配置され、一方データは揮発性(SRAM)または不揮発性(EEPROM)のメモリに置くことができます。右の図5-1をご覧ください。

全てのメモリ空間は直線的で規則的です。

図5-1. メモリ概要



### 5.1. プログラムメモリ

ATtiny48/88はプログラム保存用に実装書き換え可能な4/8Kバイトのフラッシュメモリをチップ上を含みます。フラッシュメモリは不揮発性、換言するとそれらは例えば電力がない時でも格納した情報を維持します。

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2/4K×16ビットとして構成されます。プログラムカウンタ(PC)は11/12ビット幅で、従って下の表5-1.で説明されるように、2/4Kプログラムメモリ位置のアドレス指定能力があります。

定数表はプログラムメモリのアドレス空間全体に配置できます。「命令要約」のLPMとSPMの命令をご覧ください。フラッシュプログラムメモリは124頁の「外部プログラミング」で記述されるように外部装置からプログラミングすることもできます。

命令の取得と実行のタイミング図は8頁の「命令実行タイミング」で示されます。

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。

表5-1. プログラムメモリの大きさ

デバイス名	フラッシュ容量	アドレス範囲
ATtiny48	4Kバイト	\$0000~\$07FF
ATtiny88	8Kバイト	\$0000~\$0FFF

### 5.2. データメモリ (SRAM)とレジスタファイル

表5-2.はATtiny48/88のデータメモリとレジスタファイルがどう構成されるかを示します。これらのメモリは揮発性、換言するとそれらは電力が取り去られた時に情報を保持しません。

表5-2. データメモリとレジスタファイルの配置

デバイス名	メモリ領域	大きさ	長アドレス (注1)	短アドレス (注2)
ATtiny48	汎用レジスタファイル	32バイト	\$0000~\$001F	(利用不可)
	I/Oレジスタファイル	64バイト	\$0020~\$005F	\$00~\$3F
	拡張I/Oレジスタファイル	160バイト	\$0060~\$00FF	(利用不可)
	データ用SRAM	256バイト	\$0100~\$01FF	(利用不可)
ATtiny88	汎用レジスタファイル	32バイト	\$0000~\$001F	(利用不可)
	I/Oレジスタファイル	64バイト	\$0020~\$005F	\$00~\$3F
	拡張I/Oレジスタファイル	160バイト	\$0060~\$00FF	(利用不可)
	データ用SRAM	512バイト	\$0100~\$02FF	(利用不可)

**注1:** データアドレスとしても知られます。このアドレス指定形態はデータメモリとレジスタ領域全体を網羅します。アドレスは2語命令の16ビット領域に含まれます。

**注2:** 直接I/Oアドレスとしても知られます。このアドレス指定形態はレジスタ領域の一部だけを網羅します。アドレスが命令語に組み込まれている命令によって使われます。

512/768メモリ位置は汎用レジスタファイル、I/Oレジスタファイル、拡張I/Oレジスタファイル、内蔵データメモリ(SRAM)を含みます。

将来のデバイスとの両立性のため、アクセスされる場合、予約ビットは0を書かれるべきです。予約された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。

#### 5.2.1. 汎用レジスタファイル

最初の32位置は汎用レジスタファイル用に予約されています。これらのレジスタは7頁の「汎用レジスタファイル」で詳細に記述されます。

#### 5.2.2. I/Oレジスタファイル

汎用レジスタファイルに後続する、次の64位置はI/Oレジスタ用に予約されています。この領域内のレジスタは主にデバイスの入出力と周辺機能部との通信に使われます。データはIN, OUT, LD, STとそれらの派生のような命令を用いてI/O空間と汎用レジスタファイル間で転送することができます。

この領域内の全てのレジスタはINとOUTの命令でアクセスすることができます。これらのI/O特定命令はI/Oレジスタ領域の最初の位置を\$00、最後を\$3Fとしてアドレス指定します。

下位32個(アドレス範囲\$00~\$1F)のレジスタはいくつかのビット特定命令によってアクセス可能です。これらのレジスタでは、**SBI**と**CBI**を用いてビットが容易に設定(1)と解除(0)され、一方ビット条件分岐は**SBIC**,**SBIS**,**SBRC**,**SBRS**命令を用いて容易く作成されます。

この領域内のレジスタは**LD/LDD/LDI/LDS**と**ST/STD/STS**の命令でアクセスすることもできます。これらの命令は揮発性メモリ全体を1つのデータ空間として扱い、従ってI/Oレジスタのアドレスは\$20で始まります。

186頁の「**命令要約**」をご覧ください。

ATtiny48/88はどの情報の格納にも使うことができる3つの汎用I/Oレジスタも含まれます。183頁の「**レジスタ要約**」で**GPIOR0**,**GPIOR1**,**GPIOR2**をご覧ください。これらの汎用I/Oレジスタはそれらが**SBI**,**CBI**,**SBIC**,**SBIS**,**SBRC**,**SBRS**のようなビット特定命令でアクセス可能なので、特に全体変数や状態フラグの格納に有用です。

### 5.2.3. 拡張I/Oレジスタ ファイル

標準I/Oレジスタファイルに後続する、次の160位置は拡張I/Oレジスタ用に予約されます。ATtiny48/88は**IN**と**OUT**の命令でアドレス指定することができるよりも多くの周辺機能部を持つ複合マイクロコントローラです。拡張I/O領域内のレジスタは**LD/LDD/LDI/LDS**と**ST/STD/STS**の命令を用いてアクセスされなければなりません。186頁の「**命令要約**」をご覧ください。

I/Oレジスタの一覧については183頁の「**レジスタ要約**」をご覧ください。

### 5.2.4. データメモリ (SRAM)

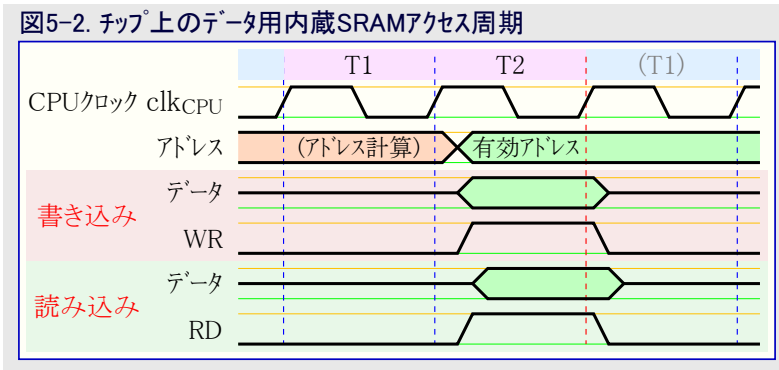
汎用レジスタファイルとI/Oレジスタファイルに後続する、残りの256/512位置はデータ用内蔵SRAM用に予約されます。

以下のように利用可能な5つのアドレス指定形態があります。

- 直接：このアドレス指定の形態はデータ空間全体に届きます。
- 間接：
  - 変位付き間接：このアドレス指定の形態はYまたはZレジスタによって与えられる基準アドレスから63アドレス位置に届きます。
  - 事前減少付き間接：この形態ではアクセスに先立ってアドレスレジスタが自動的に減少(-1)されます。アドレスポインタレジスタ(X,Y,Z)は汎用レジスタファイルのR26~R31に配置されます。7頁の「**汎用レジスタファイル**」をご覧ください。
  - 事後増加付き間接：この形態ではアクセス後にアドレスレジスタが自動的に増加(+1)されます。アドレスポインタレジスタ(X,Y,Z)は汎用レジスタファイルのR26~R31に配置されます。7頁の「**汎用レジスタファイル**」をご覧ください。

全てのアドレス指定形態は汎用レジスタファイル、I/Oレジスタファイル、データメモリを含み、揮発性メモリ全体に使うことができます。

内蔵SRAMは下の**図5-2**で図解されるように2 clkCPU周期でアクセスされます。



### 5.3. データメモリ (EEPROM)

ATtiny48/88は64バイトの不揮発性データメモリを含みます。このEEPROMは単一バイトで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。全てのアクセスレジスタはI/O空間に配置されます。

EEPROM配置は右の**表5-3**で要約されます。

**表5-3. 不揮発性データメモリ(EEPROM)の大きさ**

デバイス名	フラッシュ容量	アドレス範囲
ATtiny48/88	64バイト	\$00~\$3F

EEPROM操作の時間に内蔵8MHz発振器が使われます。この発振器の周波数は20頁の「**OSCCAL - 発振校正レジスタ**」で記述される必要条件内でなければなりません。

厳重に濾波した電源によって給電される時に電源投入/切断でVCCが緩やかに上昇または下降しそうです。緩やかな上昇と下降の時間はデバイスを指定されたよりも低い供給電圧で走行する状態に置かれるかもしれません。このような状況での問題を避けるには14頁の「**EEPROMデータ化けの防止**」をご覧ください。

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。

### 5.3.1. プログラミング法

EEPROMプログラミングには以下のように2つの方法があります。

- 非分離バイトプログラミング：これは単一操作で目標位置が消去されて書かれる、単純なプログラミング形態です。この操作形態では書く前に目標が常に消去されることが保証されますが、プログラミング時間がより長くなります。
- 分離バイトプログラミング：2つの異なる操作で消去と書き込みの周期を分けることが可能です。これは短いアクセス時間が必要とされる時、例えば供給電圧が降下する時に有用です。この方法を利用するために、目標位置はそれらへ書く前に消去されなければなりません。

プログラミング法はEEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPМ1とEEPМ0)ビットを用いて選ばれます。15頁の表5-4をご覧ください。書き込みと消去の時間が同じ表で与えられます。

EEPROMプログラミングが或る程度の時間がかかるため、応用は次の(操作)を開始する前に現在の操作の完了を待たなければなりません。これはEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットのポーリング、またはEEPROM操作可割り込み経由のどちらかによって行うことができます。EEPROM割り込みはEECRのEEPROM操作可割り込み許可(EERIE)ビットによって制御されます。

### 5.3.2. 読み込み

EEPROM位置を読むには以下の手順に従ってください。

- 他のEEPROM操作が進行中でないのを確実にするためにEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットをポーリングしてください。設定(1)ならば解除(0)を待ってください。
- 目的アドレスをEEPROMアドレスレジスタ(EEARH/EEARL)に書いてください。
- EEPROM制御レジスタ(EECR)のEEPROM読み込み許可(EERE)ビットを設定(1)することによって読み込み操作を開始してください。読み込み動作中、CPUは次の命令を実行する前に4クロック周期間停止されます。
- EEPROMデータレジスタ(EEDR)からデータを読んでください。

### 5.3.3. 消去

予期せぬEEPROM書き込みを防ぐために、メモリ位置を消去するのに特別な手順に従わなければなりません。EEPROM位置を消去するには以下の手順に従ってください。

- 他のEEPROM操作が進行中でないのを確実にするためにEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットをポーリングしてください。設定(1)ならば解除(0)を待ってください。
- EEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPМ1とEEPМ0)ビットを書くことによってプログラミング動作形態を消去に設定してください。
- 目的アドレスをEEPROMアドレスレジスタ(EEARH/EEARL)に書いてください。
- EEPROM制御レジスタ(EECR)のEEPROM主プログラム許可(EEMPE)を設定(1)することによって消去を許可してください。4クロック周期内にEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットを設定(1)することによって消去動作を開始してください。消去動作中、CPUは次の命令を実行する前に2クロック周期間停止されます。

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多忙の間、他のどんなEEPROM操作の実行も不可能です。

### 5.3.4. 書き込み

予期せぬEEPROM書き込みを防ぐために、メモリ位置を書くのに特別な手順に従わなければなりません。

EEPROMにデータを書く前に目的位置が消去されなければなりません。これは同一操作で、または分離操作の一部としてのどちらかで行うことができます。未消去EEPROM位置への書き込みは不正なデータに帰着します。

EEPROM位置を書くには以下の手順に従ってください。

- 他のEEPROM操作が進行中でないのを確実にするためにEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットをポーリングしてください。設定(1)ならば解除(0)を待ってください。
- EEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPМ1とEEPМ0)ビットを書くことによってプログラミング動作形態を設定してください。2者選択として、1操作でのデータ書き込みか、または消去のみと書き込みのみに分離する書き込み手順にできます。
- 目的アドレスをEEPROMアドレスレジスタ(EEARH/EEARL)に書いてください。
- 目的データをEEPROMデータレジスタ(EEDR)に書いてください。
- EEPROM制御レジスタ(EECR)のEEPROM主プログラム許可(EEMPE)を設定(1)することによって書き込みを許可してください。4クロック周期内にEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットを設定(1)することによって書き込み動作を開始してください。書き込み動作中、CPUは次の命令を実行する前に2クロック周期間停止されます。

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。デバイスがプログラミング多忙の間、他のどんなEEPROM操作の実行も不可能です。

### 5.3.5. 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。

低供給電圧に於いてEEPROMのデータは以下の2つのように不正にされ得ます。

- その他正当なEEPROMプログラミング手順の正しい操作を維持するために供給電圧が低すぎます。
- CPUに対して供給電圧が低すぎて、命令が不正に実行されるかもしれません。

EEPROMデータ化けは不十分な供給電源電圧の期間中、デバイスをリセットに保つことによって避けられます。これは内蔵**低電圧検出器(BOD)**を許可することによって容易に行えます。BOD検出基準が設計に対して充分でなければ、低VCCに対する外部リセット回路を使うことができます。

供給電圧が充分ならば、例えリセットが起きる時でも、EEPROM書き込み動作は完了されます。

### 5.3.6. プログラム例

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 R19, (0<<EEM1) | (0<<EEM0) ;プログラミング種別値取得(本例は非分離)
            OUT EECR, R19 ;対応プログラミング種別設定
            OUT EEARH, R18 ;EEPROMアドレス上位設定
            OUT EEARL, R17 ;EEPROMアドレス下位設定
            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 int 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<<EEM1) | (0<<EEM0); /* 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 */
}
```

注: 5頁の「コード例について」をご覧ください。

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 EEARH, R18 ;EEPROMアドレス上位設定
            OUT EEARL, R17 ;EEPROMアドレス下位設定
            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 int 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 */
}
```

注: 5頁の「コード例について」をご覧ください。

## 5.4. メモリ関係レジスタ

### 5.4.1. EEARH,EEARL (EEAR) – EEPROMアドレスレジスタ (EEPROM Address Register)

ビット	15	14	13	12	11	10	9	8	
\$-- (\$--)	–	–	–	–	–	–	–	–	EEARH
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$21 (\$41)	–	–	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
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	不定	不定	不定	不定	

- ビット15~6 – 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5~0 – EEAR5~0 : EEPROMアドレス (EEPROM Address)

EEPROMアドレスレジスタ(EEARHとEEARL)は64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0~63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ねばなりません。

### 5.4.2. EEDR – EEPROMデータレジスタ (EEPROM Data Register)

ビット	7	6	5	4	3	2	1	0	
\$20 (\$40)	(MSB)							(LSB)	EEDR
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 – 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。

### 5.4.3. EECR – EEPROM制御レジスタ (EEPROM Control Register)

ビット	7	6	5	4	3	2	1	0	
\$1F (\$3F)	–	–	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はEEPROMプログラミング許可(EEPE)が書かれた時に起動される活動を定義します。データは新しい値が書かれる前に直前の値が自動的に消去される単一非分離操作でプログラミングすることができ、または消去と書き込みを2つの異なる操作に分離することができます。各動作に対するプログラミング時間は表5-4.で示されます。

EEPEが設定(1)されていると、EEPmnへのどの書き込みも無視されます。

リセット中、EEPmnビットはEEPROMがプログラミング多忙中を除いて'00'にリセットされます。

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

このビットの1書き込みがEEPROM操作可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)ビットが設定(1)されているなら、不揮発性メモリがプログラミングに対して準備可の時にEEPROM操作可割り込みが起動されます。

このビットの0書き込みはEEPROM操作可割り込みを禁止します。

- 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)されて4クロック周期内にEEPEが設定(1)されると、選んだアドレスのEEPROMがプログラムされます。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。

表5-4. EEPROMプログラミング種別ビットとプログラミング時間

EEPM1	EEPM0	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

EEMPEが0なら、EEPE設定(1)は無効です。

• **ビット1 – EEPE : EEPROMプログラム許可 (EEPROM Program Enable) (訳注:本文を修正)**

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EEPm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、さもないとEEPROM書き込み(消去)は行われません。EEPROMを書くとき、次の手順に従うべきです(手順3.と4.の順番は重要ではありません)。

1. EEPROMプログラム許可(EEPE)ビットが0になるまで待ちます。
2. SPM制御/状態レジスタ(SPMCSR)の自己プログラミング許可(SELFPRGEN)ビットが0になるまで待ちます。
3. 今回のEEPROMアドレスをEEPROMアドレスレジスタ(EEAR)に書きます。(任意)
4. 今回のEEPROMデータをEEPROMデータレジスタ(EEDR)に書きます。(任意)
5. EEPROM制御レジスタ(EECR)のEEPROM主プログラム許可(EEMPE)ビットに1を、EEPROMプログラム許可(EEPE)ビットに0を同時に書きます。
6. EEMPEビット設定後4クロック周期内にEEPROMプログラム許可(EEPE)ビットへ論理1を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMはプログラム(書き込み)ができません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを検査しなければなりません。フラッシュメモリが決してCPUによって更新されないなら、2.は省略できます。

**警告:** 手順5.と6.間の割り込みはEEPROM主プログラム許可が時間超過するため、書き込み周期失敗になります。EEPROMをアクセスする割り込みルーチンが他のEEPROMアクセスを中断し、EEARかEEDRが変更されると、中断したEEPROMアクセスを失敗させます。これらの問題を避けるため、全ての手順中、ステータスレジスタ(SREG)の全割り込み許可(I)ビットは解除(0)されていることが推奨されます。

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)**

これはEEPROMの読み込みストロープです。目的アドレスがEEARに構成設定されると、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らない。

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が読まれる時に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。

**5.4.4. GPIOR2 – 汎用I/Oレジスタ2 (General Purpose I/O Register 2)**

ビット	7	6	5	4	3	2	1	0	
\$2B (\$4B)	(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	

このレジスタはどんな種類のデータの格納にも自由に使えるかもしれません。

**5.4.5. GPIOR1 – 汎用I/Oレジスタ1 (General Purpose I/O Register 1)**

ビット	7	6	5	4	3	2	1	0	
\$2A (\$4A)	(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	

このレジスタはどんな種類のデータの格納にも自由に使えるかもしれません。

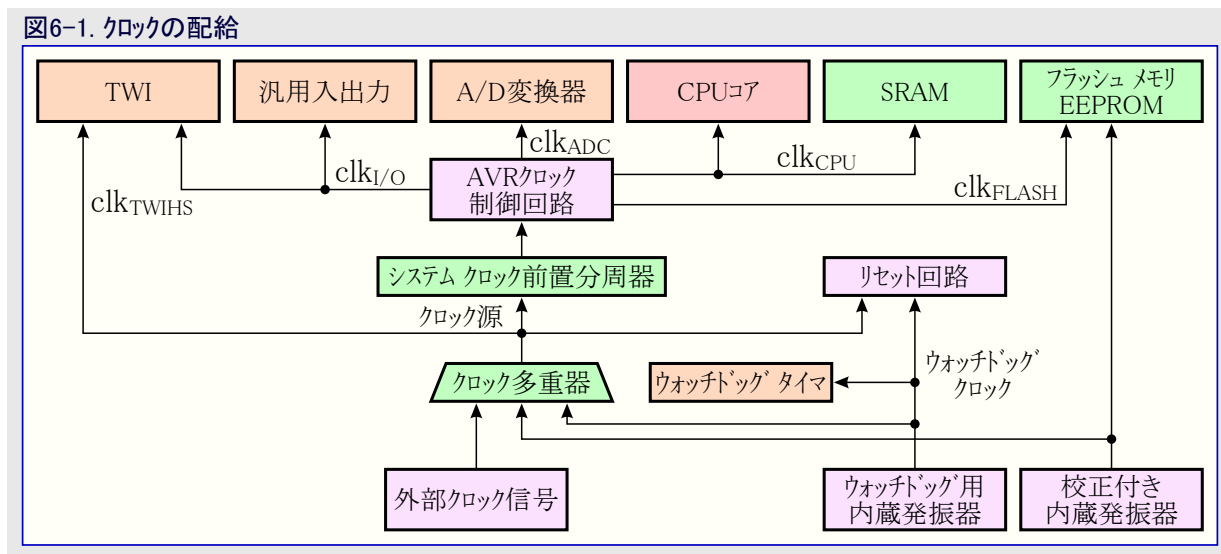
**5.4.6. GPIOR0 – 汎用I/Oレジスタ0 (General Purpose I/O Register 0)**

ビット	7	6	5	4	3	2	1	0	
\$1E (\$3E)	(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	

このレジスタはどんな種類のデータの格納にも自由に使えるかもしれません。

## 6. クロック体系

図6-1はAVR内の主要なクロックシステム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、使われない部分のクロックを停止することができます。クロックシステムは以下で詳細に示されます。



### 6.1. クロック副系統

クロック副系統が以下の項で詳述されます。

#### 6.1.1. CPU クロック - clk<sub>CPU</sub>

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。

#### 6.1.2. I/O クロック - clk<sub>I/O</sub>

I/Oクロックはタイマ/カウンタ、直列周辺インターフェース、外部割り込み部のようなI/O部の大部分で使われます。いくつかの外部割り込みは例えI/Oクロックが停止されてもそれらが認証されることを意味する、非同期論理回路によって検知されることに注意してください。2線インターフェースの開始条件検出が(例えclk<sub>I/O</sub>が停止されている時でも)全休止形態でのTWIアドレス認証動作を意味する非同期であることに注意してください。

#### 6.1.3. フラッシュ クロック - clk<sub>FLASH</sub>

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。

#### 6.1.4. A/D変換クロック - clk<sub>ADC</sub>

A/D変換器には専用のクロック範囲が提供されます。これはデジタル回路によって生成された雑音を低減するためにCPUとI/Oクロックの停止を許します。これはより正確なA/D変換結果を与えます。

#### 6.1.5. 高速TWIクロック - clk<sub>TWHHS</sub>

このTWIクロックは高速動作で操作される時の2線インターフェース部の動作を制御します。実際問題として、このクロックはデバイスのクロック源と同じです。87頁の「ビット速度発生器」をご覧ください。

### 6.2. クロック元

デバイスはシステムクロックに以下の供給元のどれかを使うことができます。

- 外部クロック信号 (18頁をご覧ください。)
- 校正付き内蔵8MHz発振器 (18頁をご覧ください。)
- 内部128kHz発振器 (18頁をご覧ください。)

クロック元はCKSELヒューズを用いて選ばれます。表6-1をご覧ください。

ヒューズビットのより多くの情報については122頁の「ヒューズビット」をご覧ください。

表6-1. デバイス クロック任意選択

CKSEL1,0 (注1)	デバイス クロック任意選択	周波数
0 0	外部クロック信号	0~12MHz
0 1	(予約)	-
1 0 (注2)	校正付き内蔵発振器	8MHz (注3)
1 1	128kHz内部(WDT)発振器	128kHz

注1: 全ヒューズに対し、“1”は非プログラム、“0”はプログラムを意味します。

注2: デバイスはこの任意選択で出荷されます。

注3: 周波数8MHzが(VCCに依存して)デバイスの仕様を超える場合、内部周波数を8分周するのにCKDIV8ヒューズをプログラム(0)にできます。

### 6.2.1. 外部クロック信号

外部クロック元からデバイスを駆動するにはCLKIが図6-2.で示されるように駆動されるべきです。このクロック元が選ばれると、起動時間は表6-2.で示されるようにSUTヒューズによって決定されます。

図6-2. 外部クロック駆動構成設定

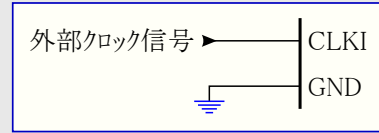


表6-2. 外部クロック信号駆動用起動遅延時間選択表

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)	電力条件
0 0	6 × CK	14 × CK	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)		

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19頁の「システムクロック前置分周器」を参照してください。

### 6.2.2. 校正付き内蔵8MHz発振器

既定による校正された内蔵発振器は概ね8.0MHzのクロックを供給します。電圧と温度に依存しますが、このクロックは使用者によって高精度な校正ができます。より多くの詳細については136頁の表22-1.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。より多くの詳細については19頁の「システムクロック前置分周器」をご覧ください。

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器校正(OSCCAL)レジスタに予めプログラムされた校正値バイトを設定し、これによって発振器を自動的に校正します。この校正の精度は表22-1.で工場校正として示されます。

ソフトウェアからOSCCALレジスタを変更することによって(発振器校正レジスタ(OSCCAL)参照)、工場校正を使うよりも高い精度を得ることができます。この校正の精度は表22-1.で使用者校正として示されます。

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ムとリセット付加遅延タイムに使われます。予め設定された校正値のより多くの情報については123頁の「校正バイト」項をご覧ください。

この発振器が選ばれると、起動時間は下表で示されるようにSUTヒューズによって決定されます。

表6-3. 校正付き内蔵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 (VCC=5.0V)	電力条件
0 0	6 × CK	14 × CK (注1)	低電圧検出リセット(BOD)許可
0 1	6 × CK	14 × CK+4ms	高速上昇電源
1 0 (注2)	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。

注2: デバイスはこの選択で出荷されます。

### 6.2.3. 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。この周波数は供給電圧、温度、不均一さに依存します。このクロックはCKSELヒューズを'11'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6-4.で示されるようにSUTヒューズによって決定されます。

表6-4. 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。

## 6.2.4. 既定のクロック元

このデバイスは8.0MHzの校正付き内蔵発振器でCKDIV8ヒューズがプログラム(0)され、結果として1.0MHzのシステムクロックで出荷されます。起動時間は計時完了周期が許可され、最大に設定されます。(CKSEL=10, SUT=10, CKDIV8=プログラム(0)) この既定設定は全ての使用者が利用可能なプログラミングインターフェースを用いて、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。

## 6.3. システムクロック前置分周器

ATtiny48/88はシステムクロック前置分周器を持ち、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。clkCPU、clkFLASH、clkI/O、clkADCは表6-6.で示された値によって分周されます。

### 6.3.1. 前置分周器設定切り替え

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(CLKPCE)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CLKPCE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。

## 6.4. クロック出力緩衝部 (外部クロック出力)

このデバイスはシステムクロックをCLKOピンに出力できます。本出力を許可するにはCKOUTヒューズがプログラム(0)されなければなりません。この動作はチップのクロックがシステム上の他の回路を駆動する時用です。このヒューズがプログラム(0)されると、I/Oピンの標準動作は無視され、このクロックはリセット中も出力されます。CLKOがクロック出力を扱うとき、校正付き内蔵発振器を含む何れのクロック元も選ばず、システムクロック前置分周器が使われた場合、分周されたシステムクロックが出力されます。

## 6.5. クロック起動手順

何れのクロック元も発振を開始するための十分なVCCと、それが安定であると考えられるのに先立って最低発振周期数が必要です。

十分なVCCを保証するためにその他全てのリセット元によってデバイスリセットが開放された後、デバイスは起動遅延時間(t<sub>OUT</sub>)の内部リセットを発生します。25頁からの「システム制御とリセット」はこの内部リセットに対する起動条件を記述します。この遅延(t<sub>OUT</sub>)はウォッチドッグ発振器で計時され、遅延周期数はSUTとCKSELのヒューズによって設定されます。選択可能な遅延は表6-5.で示されます。161頁の「ウォッチドッグ発振器周波数対動作電圧」と162頁の「ウォッチドッグ発振器周波数対動作温度」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。

遅延の主な目的はAVRが最小VCCを印加されるまでAVRをリセットに保つことです。この遅延は実電圧を監視しないので、VCC上昇時間よりも長い遅延を選ぶことが必要とされるべきです。これが不可能な場合、内部または外部の低電圧検出回路(BOD)が使われるべきです。BOD回路がリセットを開放する前に十分なVCCを保証するようから、起動遅延時間は禁止され得ます。低電圧検出回路(BOD)なしでの起動遅延時間の禁止は推奨されません。

この発振器はクロックが安定と考えられるのに先立って最低クロック数を発振することを必要とされます。内部リプルカウンタは発振器の出力クロックを監視し、与えられたクロック周期数に対して内部リセットを活性(有効)に保ちます。このリセットはその後に開放され、デバイスが実行を開始します。

クロックについての起動手順は計時完了遅延とデバイスがリセットから起動する時の起動時間の両方を含みます。パワーダウンから起動する時にVCCは十分な電圧であると認識され、起動時間だけが含められます。

表6-5. 始動手順時間長

CKSEL1,0	SUT1,0	WDT周期数	代表経過時間
0 0	0 0	0	0ms
1 0	0 1	4K (4096)	4ms
1 1	1 0	8K (8192)	64ms
	1 1	(予約)	(予約)
0 1	x x	(予約)	(予約)

## 6.6. クロック関係レジスタ

### 6.6.1. OSCCAL – 発振校正レジスタ (Oscillator Calibration Register)

ビット (\$66)	7	6	5	4	3	2	1	0	
	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	デバイス固有の校正値								

#### • ビット7~0 – CAL7~0 : 発振校正値 (Oscillator Calibration Value)

発振校正レジスタは発振器周波数の偏差処理を省くための内蔵発振器の調整に使われます。チップのリセット中、136頁の表22-1.で指定されるように工場で校正された周波数を与える、予めプログラムされた値が本レジスタへ自動的に書かれます。

応用ソフトウェアは発振器周波数を変更するためにOSCCALレジスタへ書くことができます。この発振器は表22-1.で指定される周波数に校正できます。この範囲外への校正は推奨されません。

この発振器はフラッシュメモリとEEPROMの書き込みアクセス時間に使われ、この書き込み時間がそれに応じて影響されることに注意してください。フラッシュメモリまたはEEPROMが書かれる場合、8.8MHzより高く校正してはいけません。そうでなければ、フラッシュメモリまたはEEPROM書き込みは失敗するかもしれません。

全てのレジスタビットが周波数用に使います。\$00設定が最低周波数、\$FF設定が最高周波数を生じます。

### 6.6.2. CLKPR – クロック前置分周レジスタ (Clock Prescaler Register)

ビット (\$61)	7	6	5	4	3	2	1	0	
	CLKPCE	-	-	-	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 – CLKPCE : 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CLKPCEビットは論理1を書かれなければなりません。CLKPCE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。CLKPCE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CLKPCEビット再書き込みは制限時間の延長もCLKPCEビットの解除(0)も行いません。

#### • ビット6~4 – 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3~0 – 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6-6.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)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。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。

表6-6. 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)							

## 7. 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。

許可したなら、低電圧検出器(BOD)は休止期間中、電源電圧を積極的に監視します。更なる節電のため、いくつかの休止形態種別でBODを禁止することが可能です。より多くの詳細については次頁の「ソフトウェア低電圧検出器(BOD)禁止」をご覧ください。

### 7.1. 休止形態種別

17頁の図6-1はATtiny48/88の各種クロック系統とその配給を示します。本図は適切な休止形態を選ぶ助けになります。表7-1は各種休止形態、それらの起動元とBOD禁止の可能性を示します。

表7-1. 各休止形態に於ける動作クロック範囲と復帰起動要因

休止形態	動作クロック範囲				動作発振器 主クロック 供給元	復帰起動要因 (割り込み)					
	clkCPU	clkFLASH	clkI/O	clkADC		INT1 INT0 ピン変化	TWI アドレス一致	EEPROM 操作可	A/D変換 完了	ウォッチドッグ	その他 I/O
アイドル	○	○	○	○	○	○	○	○	○	○	○
A/D変換雑音低減	○	○	○	○	○	①	○	○	○	○	○
パワーダウン	○	○	○	○	○	①	○	○	○	○	○

①: INT1とINT0についてはレベル割り込みだけです。

休止形態のどれかへ移行するには、**休止形態制御レジスタ(SMCR)**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。SMCR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、A/D変換雑音低減、パワーダウン)のどれかを選びます。一覧については23頁の表7-2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こると、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の内容は変えられません。休止形態中にリセットが起こると、MCUは起動し、リセットベクタから実行します。

レベルで起動した割り込みが起動復帰に使われる場合、MCUを起動(とMCUがその割り込み処理ルーチンへ移行)するには、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32頁の「外部割り込み」を参照してください。

#### 7.1.1. 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、SPI、アナログ比較器、A/D変換器、2線直列インターフェース、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SPI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。A/D変換が許可されるなら、この動作に移行すると変換が自動的に始まります。

#### 7.1.2. A/D変換雑音低減動作

**休止種別選択(SM1,0)ビット**が'01'を書かれるとき、**SLEEP**命令はMCUをA/D変換雑音低減動作へ移行させ、CPUを停止しますが、A/D変換器、外部割り込み、2線直列インターフェースのアドレス監視、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkI/O、clkCPU、clkFLASHを停止する一方、他のクロックに走行を許します。

これはA/D変換に対する雑音環境を改善し、より高い分解能の測定を可能にします。A/D変換器が許可されている場合、本動作に移行すると、変換が自動的に始まります。A/D変換完了割り込みからの他、外部リセット、ウォッチドッグシステムリセット、ウォッチドッグ割り込み、低電圧検出(BOD)リセット、2線直列インターフェースのアドレス一致割り込み、EEPROM操作可割り込み、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、A/D変換雑音低減動作からMCUを起動できます。

#### 7.1.3. パワーダウン動作

**休止種別選択(SM1,0)ビット**が'10'を書かれるとき、**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、2線直列インターフェースのアドレス監視、ウォッチドッグ機能は(許可されていれば)継続して動作します。外部リセット、ウォッチドッグシステムリセット、ウォッチドッグ割り込み、低電圧検出(BOD)リセット、2線直列インターフェースのアドレス一致割り込み、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32頁の「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17頁の「クロック元」で記述されるように、リセット付加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。



## 7.2. ソフトウェア低電圧検出器(BOD)禁止

低電圧検出器(BOD)が122頁の表20-4のBODLEVELヒューズによって許可されていると、BODは休止期間中に電源電圧を活発に監視します。節電のため、休止形態のいくつかに対してソフトウェアによってBODを禁止することが可能です。表7-1をご覧ください。この休止形態電力消費はBODがヒューズによって全面的に禁止される時と同じ水準になるでしょう。BODがソフトウェアで禁止される場合、BOD機能は休止形態移行後、直ちにOFFされます。休止からの起動復帰でBODは再び自動的に許可されます。これは休止期間中にVCC水準が落ちた場合の安全な動作を保証します。

BODが禁止されてしまうと、MCUがコードの実行を継続する前にBODが正しく動作することを保証するために、休止形態からの起動時間は概ね60 $\mu$ sになります。

BOD禁止はMCU制御レジスタ(MCUCR)のビット6、BOD休止(BODS)ビットによって制御されます。23頁の「MCUCR – MCU制御レジスタ」をご覧ください。このビットへの1書き込みは関連する休止形態でのBODをOFFにし、一方このビットの0はBOD活動(有効)を保ちます。既定設定はBOD活動維持、換言するとBODSは0に設定です。

BODSビットへの書き込みは許可ビットと時間制限手順によって制御されます。23頁の「MCUCR – MCU制御レジスタ」をご覧ください。

## 7.3. 消費電力の最小化

AVRが制御するシステムで消費電力の最小化を試みる時の考慮には様々な検討点があります。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るでしょう。

### 7.3.1. A/D変換器 (ADC)

許可なら、A/D変換器は全休止形態で許可されます。電力を節約するため、休止形態の何れかへ移行する前にA/D変換器は禁止されるべきです。A/D変換器がOFFそして再びONに切り替えられると、次の(最初の)変換は延長された(初回)変換になります。A/D変換器操作の詳細については107頁の「A/D変換器」を参照してください。

### 7.3.2. 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。A/D変換雑音削減動作へ移行するとき、アナログ比較器は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105頁の「アナログ比較器」を参照してください。

### 7.3.3. 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26頁の「低電圧検出 (BOD)」を参照してください。

### 7.3.4. 内部基準電圧

内部基準電圧は低電圧検出器(BOD)、アナログ比較器、A/D変換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27頁の「内部基準電圧」を参照してください。

### 7.3.5. 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27頁の「ウォッチドッグ タイマ」を参照してください。

### 7.3.6. ポートピン

休止動作へ移行するとき、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( $clk_{I/O}$ )とA/D変換クロック( $clk_{ADC}$ )の両方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40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(DIDR0とDIDR1)の書き込みによって禁止できます。詳細については106頁の「DIDR1 – デジタル入力禁止レジスタ1」と115頁の「DIDR0 – デジタル入力禁止レジスタ0」を参照してください。

### 7.3.7. 内蔵デバッグ機能 (dW)

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。

## 7.4. 電力管理用レジスタ

### 7.4.1. SMCR – 休止形態制御レジスタ (Sleep Mode Control Register)

この休止形態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	
\$33 (\$53)	-	-	-	-	-	SM1	SM0	SE	SMCR
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7~3 – 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2,1 – SM1,0 : 休止種別選択 (Sleep Mode Select Bit 1 and 0)**

これらのビットは表7-2.で示される利用可能な休止形態の1つを選びます。

表7-2. 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	A/D変換雑音低減動作
1	0	パワーダウン動作
1	1	(予約)

• **ビット0 – 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。

### 7.4.2. MCUCR – MCU制御レジスタ (MCU Control Register)

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	-	BODS	BODSE	PUD	-	-	-	-	MCUCR
Read/Write	R	R/W	R/W	R/W	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

• **ビット6 – BODS : BOD休止 (BOD Sleep)**

休止間にBODをOFFにするために、BODSビットは論理1を書かれなければなりません。21頁の表7-1.をご覧ください。BODSビットの書き込みはMCUCRのBOD休止許可(BODSE)ビットと時間制限手順によって制御されます。関連する休止形態でBODを禁止するには最初にBODSとBODSEの両方が1に設定されなければなりません。その後BODSビットを設定するためにBODSが1に設定され、そして4クロック周期内にBODSEが0に設定されなければなりません。

BODSビットはそれが設定された後の3クロック周期間活性(有効)です。SLEEP命令は実際の休止形態に対してBODをOFFにするために、BODSが活性(有効)の間に実行されなければなりません。BODSビットは3クロック周期後、自動的に解除(0)されます。

• **ビット5 – BODSE : BOD休止許可 (BOD Sleep Enable)**

BODSEはBOD休止(BODS)ビット記述で説明されるように、BODS制御ビットの設定を許可します。このBOD禁止は時間制限手順によって制御されます。

### 7.4.3. PRR – 電力削減レジスタ (Power Reduction Register)

電力削減レジスタは消費電力を削減するために個別周辺機能へのクロックを停止する方法を提供します。周辺機能は現状で固定化され、I/Oレジスタは読み込みも書き込みもできません。クロックを停止している時に周辺機能によって使われていた資源は占有されたままですので、その周辺機能は殆どの場合、クロックを停止する前に禁止されるべきです。周辺機能部の起動は電力削減レジスタ(PRR)のビットを解除(0)することによって行い、その周辺機能部を停止前と同じ状態にします。

周辺機能部の停止は全体に亘る重要な消費電力の削減のために活動動作とアイドル動作で使えます。その他の休止形態ではクロックが予め停止されます。

ビット (\$64)	7	6	5	4	3	2	1	0	PRR
	PRTWI	-	PRTIM0	-	PRTIM1	PRSPI	-	PRADC	
Read/Write	R/W	R	R/W	R	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット7 – PRTWI : 2線直列インターフェース電力削減 (Power Reduction TWI)**

このビットへの1書き込みはその部分へのクロック停止によって**2線直列インターフェース(TWI)**を停止します。TWIの再起動時、TWIは正しい動作を保証するために再初期化されるべきです。

- **ビット6,4,1 – 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5 – PRTIM0 : タイマ/カウンタ0電力削減 (Power Reduction Timer/Counter0)**

このビットへの1書き込みは**タイマ/カウンタ0**部を停止します。タイマ/カウンタ0が許可されると、停止前と同様に動作は継続します。

- **ビット3 – PRTIM1 : タイマ/カウンタ1電力削減 (Power Reduction Timer/Counter1)**

このビットへの1書き込みは**タイマ/カウンタ1**部を停止します。タイマ/カウンタ1が許可されると、停止前と同様に動作は継続します。

- **ビット2 – PRSPI : 直列周辺インターフェース電力削減 (Power Reduction Serial Peripheral Interface)**

デバッグWIRE内蔵デバッグ機能を使うなら、このビットは1を書かれるべきではありません。

このビットへの1書き込みはその部分へのクロック停止によって**直列周辺インターフェース(SPI)**を停止します。SPIの再起動時、SPIは正しい動作を保証するために再初期化されるべきです。

- **ビット0 – PRADC : A/D変換器電力削減 (Power Reduction ADC)**

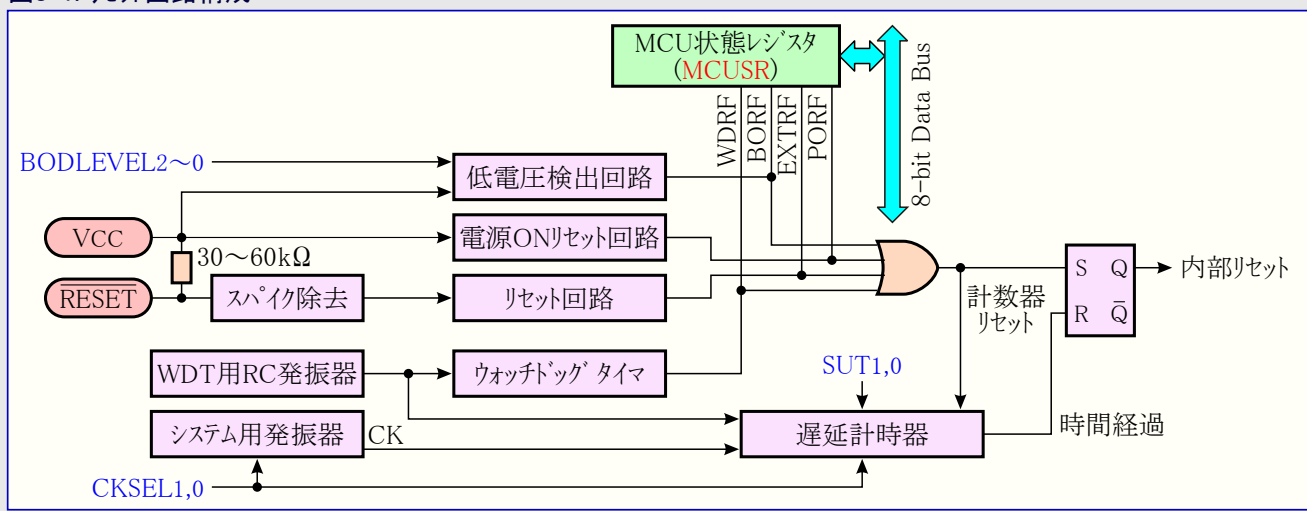
このビットへの1書き込みは**A/D変換器(ADC)**を停止します。A/D変換器は停止前に**禁止**されなければなりません。A/D変換器停止時、**アナログ比較器**は**ADC入力切替器**を使えません。

## 8. システム制御とリセット

### 8.1. 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、きっとリセット処理ルーチンへのRJMP(相対分岐)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8-1.の回路構成図はリセット論理回路を示します。表22-3.はリセット回路の電気的特性を定義します。

図8-1. リセット回路構成



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17頁の「クロック元」で示されます。

### 8.2. リセット元

ATtiny48/88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<sub>POT</sub>)以下、または供給電圧の急激な低下時にMCUがリセットされます。
- **外部リセット** …… RESETピンがパルス幅必要条件以上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** …… 低電圧検出が許可され、供給電圧(V<sub>CC</sub>)が低電圧検出電圧(V<sub>BOT</sub>)以下でMCUがリセットされます。

### 8.2.1. 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137頁の表22-3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には勿論、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8-2. 内蔵電源ONリセット (RESETはVCCに接続)

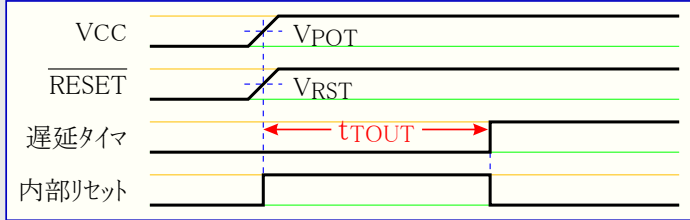
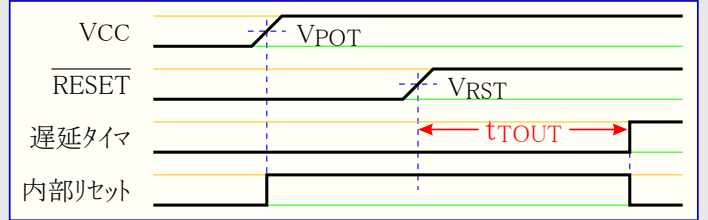


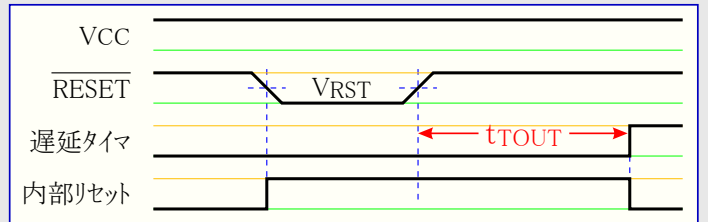
図8-3. 外部RESET信号による延長電源ONリセット



### 8.2.2. 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(137頁の表22-3.参照)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( $V_{RST}$ )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。外部リセットはRSTDISBLヒューズによって禁止できます。122頁の表20-4をご覧ください。

図8-4. 動作中の外部リセット



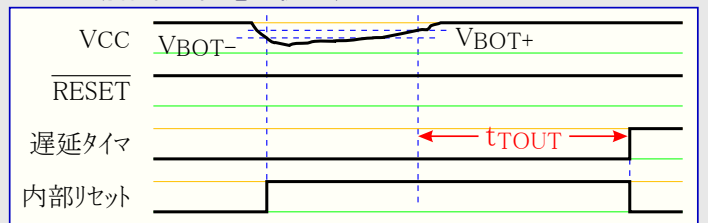
### 8.2.3. 低電圧(ブラウンアウト)検出リセット

ATtiny48/88には固定化された起動(検出)電圧と比較することで動作中の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、VCCが起動電圧以下に下降すると(図8-5.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8-5.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137頁の「システムとリセットの特性」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。

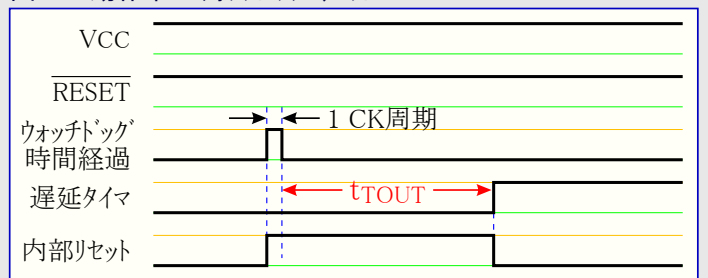
図8-5. 動作中の低電圧検出リセット



### 8.2.4. 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27頁を参照してください。

図8-6. 動作中のウォッチドッグ リセット



## 8.3. 内部基準電圧

ATtiny48/88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、A/D変換やアナログ比較器の入力としても使えます。

### 8.3.1. 基準電圧許可信号と起動時間

基準電圧には使われるべき方法に影響するかもしれない起動時間があります。この起動時間は137頁の「システムとリセットの特性」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 (BODLEVELヒューズのプログラム(0)により)
2. アナログ比較器内部基準電圧接続時 (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)
3. A/D変換部動作許可時 (A/D変換制御/状態レジスタ(ADCSRA)のA/D動作許可(ADEN)=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)またはA/D変換部許可(ADEN=1)後、使用者はアナログ比較器またはA/D変換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3つの状態を避けられます。

## 8.4. ウォッチドッグ タイマ

ATtiny48/88は強化されたウォッチドッグ タイマ(WDT)を持ちます。このウォッチドッグ 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。

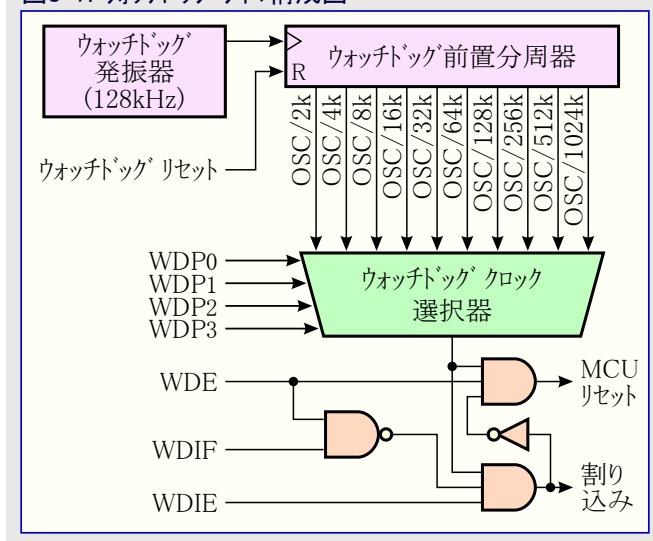
割り込み動作種別では、タイマ計時完了時にWDTが割り込みを生じます。この割り込みは休止形態からデバイスを起動するためや、一般的なシステムタイマとしても使えます。1つの例は或る動作に対して許された最大時間を制限することで、その動作が予測されたより長く走行する時に割り込みを生じます。システムリセット動作種別ではタイマ計時完了時にWDTがリセットを生じます。これは一般的にコード外走行の場合の中断を防止するのに使われます。3つ目の動作種別は先に割り込みを生じ、その後にシステムリセット動作種別に切り替えることで、他の2つの動作種別の組み合わせとなる、割り込み及びシステムリセット動作種別です。この動作種別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8-7. 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
            LDS   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
            STS   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
            STS   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: 5頁の「コード例について」をご覧ください。

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
            LDS   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
            STS   WDTCSR, R16  ;WDCEとWDEに論理1書き込み
            LDI   R16, (1<<WDE) | (1<<WDP2) | (1<<WDPO) ;WDE=1,計時間隔=0.5s値を取得
            STS   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<<WDPO); /* 0.5s監視間隔リセット動作開始 */
    __enable_interrupt();         /* 全割り込み許可 */
}
    
```

注: 5頁の「コード例について」をご覧ください。

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。

## 8.5. リセット関係レジスタ

### 8.5.1. MCUSR – MCU状態レジスタ (MCU Status Register)

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、使用者はプログラム内で可能な限り早くMCUSRを読み、そして解除(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、その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。

### 8.5.2. WDTCSR – ウォッチドッグ タイマ制御レジスタ (Watchdog Timer Control Register)

ビット (\$60)	7	6	5	4	3	2	1	0	WDTCSR
	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	
Read/Write	R/W	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 タイマが割り込みに設定され、ウォッチドッグ タイマ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 タイマ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 タイマ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 タイマ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8-1. ウォッチドッグ タイマ設定

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でプログラム、1で非プログラムに設定です。

• **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 タイマ前置分周選択 (Watchdog Timer Prescaler 3,2,1 and 0)**

このWDP3~0ビットはウォッチドッグ タイマが走行する時のウォッチドッグ タイマ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8-2.で示されます。

表8-2. ウォッチドッグ前置分周器選択

WDP3	0								1							
	0				1				0				1			
WDP2	0				1				0				1			
WDP1	0		1		0		1		0		1		0		1	
WDP0	0	1	0	1	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) (注)					
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						

注: 選んだ場合、1010未満の有効な設定の(どれか)1つが使われます。

## 9. 割り込み

本章はATtiny48/88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8頁の「リセットと割り込みの扱い」を参照してください。

### 9.1. 割り込みベクタ

表9-1. 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	PCINT0 (PCI0)	ピン変化0群割り込み要求
5	\$0004	PCINT1 (PCI1)	ピン変化1群割り込み要求
6	\$0005	PCINT2 (PCI2)	ピン変化2群割り込み要求
7	\$0006	PCINT3 (PCI3)	ピン変化3群割り込み要求
8	\$0007	ウォッチドッグ WDT	ウォッチドッグ計時完了
9	\$0008	タイマ/カウンタ1 CAPT	タイマ/カウンタ1捕獲発生
10	\$0009	タイマ/カウンタ1 COMPA	タイマ/カウンタ1比較A一致
11	\$000A	タイマ/カウンタ1 COMPB	タイマ/カウンタ1比較B一致
12	\$000B	タイマ/カウンタ1 OVF	タイマ/カウンタ1溢れ
13	\$000C	タイマ/カウンタ0 COMPA	タイマ/カウンタ0比較A一致
14	\$000D	タイマ/カウンタ0 COMPB	タイマ/カウンタ0比較B一致
15	\$000E	タイマ/カウンタ0 OVF	タイマ/カウンタ0溢れ
16	\$000F	SPI STC	SPI 転送完了
17	\$0010	A/D変換器 ADC	A/D変換完了
18	\$0011	EEPROM EE_RDY	EEPROM 操作可
19	\$0012	アナログ比較器 ANA_COMP	アナログ比較器出力遷移
20	\$0013	2線直列インターフェース TWI	2線直列インターフェース状態変化

プログラムが決して割り込み元を許可しない場合、割り込みベクタは使われず、その結果としてそれらの場所に通常のプログラムコードを置くことができます。

ATtiny48/88での代表的で一般的な割り込みベクタ アドレス構成設定が次のプログラム例で示されます。

### アセンブリ言語プログラム例

```

$0000      ORG      $0000          ;プログラム用フラッシュ メモリ先頭位置

$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    PCINT0        ;ピン変化0群割り込み要求
$0004      RJMP    PCINT1        ;ピン変化1群割り込み要求
$0005      RJMP    PCINT2        ;ピン変化2群割り込み要求
$0006      RJMP    PCINT3        ;ピン変化3群割り込み要求
$0007      RJMP    WDT_OVF       ;ウォッチドッグ計時完了
$0008      RJMP    TIM1_CAPT     ;タイマ/カウンタ1捕獲発生
$0009      RJMP    TIM1_COMP A   ;タイマ/カウンタ1比較A一致
$000A      RJMP    TIM1_COMP B   ;タイマ/カウンタ1比較B一致
$000B      RJMP    TIM1_OVF     ;タイマ/カウンタ1溢れ
$000C      RJMP    TIM0_COMP A   ;タイマ/カウンタ0比較A一致
$000D      RJMP    TIM0_COMP B   ;タイマ/カウンタ0比較B一致
$000E      RJMP    TIM0_OVF     ;タイマ/カウンタ0溢れ
$000F      RJMP    SPI_STC      ;SPI転送完了
$0010      RJMP    ADC          ;ADC変換完了
$0011      RJMP    EE_RDY       ;EEPROM操作可
$0012      RJMP    ANA_COMP     ;アナログ比較器出力遷移
$0013      RJMP    TWI          ;2線直列インターフェース状態変化
;
$0014      RESET:   "命令"       ;主プログラム開始
  
```

注: 5頁の「コード例について」をご覧ください。

## 9.2. 外部割り込み

外部割り込みはINT0,INT1ピンまたはPCINT0~27ピンの何れかによって起動されます。許可したなら、例えINT0,1またはPCINT0~27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。この特徴は以下のようなソフトウェア割り込みを生成する方法を提供します。

- PCI3ピン変化割り込みは許可したPCINT24~27ピンの何れかが切り替わると起動します。
- PCI2ピン変化割り込みは許可したPCINT16~23ピンの何れかが切り替わると起動します。
- PCI1ピン変化割り込みは許可したPCINT8~15ピンの何れかが切り替わると起動します。
- PCI0ピン変化割り込みは許可したPCINT0~7ピンの何れかが切り替わると起動します。

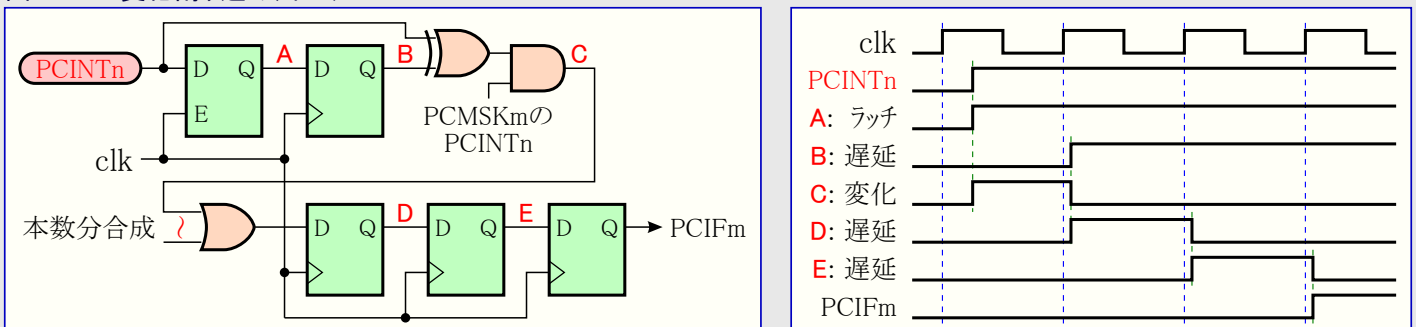
ピン変化割り込み許可レジスタn(PCMSK0,PCMSK1,PCMSK2,PCMSK3)は、どのピンがピン変化割り込み要因となるかを制御します。PCINT0~27でのピン変化割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイスを起動するのに使えることを意味します。

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。これは外部割り込み制御レジスタA(EICRA)で記述されるように構成設定されます。INT0またはINT1割り込みがレベル起動として設定、そして許可されると、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の上昇端や下降端割り込みの認知は17頁の「I/Oクロック - clk<sub>I/O</sub>」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。

### 9.2.1. 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9-1.で示されます。

図9-1. ピン変化割り込みタイミング



## 9.2.2. Lowレベル割り込み

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た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した場合、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19頁の表6-5.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。

デバイスが起動復帰する前に割り込みピン上のLowレベルが取り去られると、プログラム実行は割り込み処理ルーチンへ転換されませんが、**SLEEP**命令に続く命令から継続します。

## 9.3. 外部割り込み用レジスタ

### 9.3.1. EICRA – 外部割り込み制御レジスタA (External Interrupt Control Register A)

外部割り込み制御レジスタAは割り込み条件制御ビットを含みます。

ビット (\$69)	7	6	5	4	3	2	1	0	EICRA
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7~4 – 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3,2 – 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)ビットと外部割り込み許可レジスタ(EIMSK)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9-2.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ています。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の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なければなりません。

表9-2. 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ピンがLowレベルで発生。
0	1	INT1ピンの論理変化(両端)。
1	0	INT1ピンの下降端で発生。
1	1	INT1ピンの上昇端で発生。

- **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)ビットと外部割り込み許可レジスタ(EIMSK)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9-3.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ています。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の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なければなりません。

表9-3. 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レベルで発生。
0	1	INT0ピンの論理変化(両端)。
1	0	INT0ピンの下降端で発生。
1	1	INT0ピンの上昇端で発生。

### 9.3.2. EIMSK – 外部割り込み許可レジスタ (External Interrupt Mask Register)

ビット	7	6	5	4	3	2	1	0	
\$1D (\$3D)	-	-	-	-	-	-	INT1	INT0	EIMSK
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7~2 – 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 – 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。外部割り込み制御レジスタA(EICRA)の**割り込み条件制御1のビット(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0 – 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。外部割り込み制御レジスタA(EICRA)の**割り込み条件制御0のビット(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。

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

ビット	7	6	5	4	3	2	1	0	
\$1C (\$3C)	-	-	-	-	-	-	INTF1	INTF0	EIFR
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7~2 – 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 – 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)ビットと外部割り込み許可レジスタ(EIMSK)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0 – 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)ビットと外部割り込み許可レジスタ(EIMSK)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。

### 9.3.4. PCICR – ピン変化割り込み制御レジスタ (Pin Change Interrupt Control Register) (訳注:次頁下段参照)

ビット (\$68)	7	6	5	4	3	2	1	0	
	–	–	–	–	PCIE3	PCIE2	PCIE1	PCIE0	PCICR
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7~4 – 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 – PCIE3 : ピン変化3群割り込み許可 (Pin Change Interrupt Enable 3)**

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、PCIE3ビットが設定(1)されると、ピン変化3群割り込みが許可されます。許可したPCINT24~27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**PCI3割り込みベクタ**から実行されます。PCINT24~27ピンは**ピン変化割り込み許可レジスタ3(PCMSK3)**によって個別に許可されます。

• **ビット2 – PCIE2 : ピン変化2群割り込み許可 (Pin Change Interrupt Enable 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、PCIE2ビットが設定(1)されると、ピン変化2群割り込みが許可されます。許可したPCINT16~23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**PCI2割り込みベクタ**から実行されます。PCINT16~23ピンは**ピン変化割り込み許可レジスタ2(PCMSK2)**によって個別に許可されます。

• **ビット1 – PCIE1 : ピン変化1群割り込み許可 (Pin Change Interrupt Enable 1)**

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、PCIE1ビットが設定(1)されると、ピン変化1群割り込みが許可されます。許可したPCINT8~14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**PCI1割り込みベクタ**から実行されます。PCINT8~14ピンは**ピン変化割り込み許可レジスタ1(PCMSK1)**によって個別に許可されます。

• **ビット0 – PCIE0 : ピン変化0群割り込み許可 (Pin Change Interrupt Enable 0)**

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、PCIE0ビットが設定(1)されると、ピン変化0群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**PCI0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0(PCMSK0)**によって個別に許可されます。

### 9.3.5. PCIFR – ピン変化割り込み要求フラグ レジスタ (Pin Change Interrupt Flag Register) (訳注:次頁下段参照)

ビット (\$3B)	7	6	5	4	3	2	1	0	
	–	–	–	–	PCIF3	PCIF2	PCIF1	PCIF0	PCIFR
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7~4 – 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 – PCIF3 : ピン変化3群割り込み要求フラグ (Pin Change Interrupt Flag 3)**

PCINT24~27ピンの何れかの論理変化が割り込み要求を起動すると、PCIF3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)ビットとピン変化割り込み制御レジスタ(PCICR)の**ピン変化3群割り込み許可(PCIE3)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処理ルーチンが開始されると解除(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。

• **ビット2 – PCIF2 : ピン変化2群割り込み要求フラグ (Pin Change Interrupt Flag 2)**

PCINT16~23ピンの何れかの論理変化が割り込み要求を起動すると、PCIF2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)ビットとピン変化割り込み制御レジスタ(PCICR)の**ピン変化2群割り込み許可(PCIE2)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処理ルーチンが開始されると解除(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。

• **ビット1 – PCIF1 : ピン変化1群割り込み要求フラグ (Pin Change Interrupt Flag 1)**

PCINT8~14ピンの何れかの論理変化が割り込み要求を起動すると、PCIF1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)ビットとピン変化割り込み制御レジスタ(PCICR)の**ピン変化1群割り込み許可(PCIE1)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処理ルーチンが開始されると解除(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。

• **ビット0 – PCIF0 : ピン変化0群割り込み要求フラグ (Pin Change Interrupt Flag 0)**

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、PCIF0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)ビットとピン変化割り込み制御レジスタ(PCICR)の**ピン変化0群割り込み許可(PCIE0)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処理ルーチンが開始されると解除(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。



### 9.3.6. PCMSK3 – ピン変化割り込み許可レジスタ3 (Pin Change Enable Mask 24~27) (訳注: 頁下段参照)

ビット (\$6A)	7	6	5	4	3	2	1	0	
	-	-	-	-	PCINT27	PCINT26	PCINT25	PCINT24	PCMSK3
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7~4 – 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3~0 – PCINT27~PCINT24 : ピン変化割り込み27~24許可 (Pin Change Enable Mask 27~24)

各PCINT24~27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。PCINT24~27とピン変化割り込み制御レジスタ(PCICR)のPCIE3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。PCINT24~27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。

### 9.3.7. PCMSK2 – ピン変化割り込み許可レジスタ2 (Pin Change Enable Mask 16~23)

ビット (\$6D)	7	6	5	4	3	2	1	0	
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	PCMSK2
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 – PCINT23~PCINT16 : ピン変化割り込み23~16許可 (Pin Change Enable Mask 23~16)

各PCINT16~23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。PCINT16~23とピン変化割り込み制御レジスタ(PCICR)のPCIE2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。PCINT16~23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。

### 9.3.8. PCMSK1 – ピン変化割り込み許可レジスタ1 (Pin Change Enable Mask 8~14)

ビット (\$6C)	7	6	5	4	3	2	1	0	
	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	PCMSK1
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 – PCINT15~PCINT8 : ピン変化割り込み15~8許可 (Pin Change Enable Mask 15~8)

各PCINT8~15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。PCINT8~15とピン変化割り込み制御レジスタ(PCICR)のPCIE1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。PCINT8~15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。

### 9.3.9. PCMSK0 – ピン変化割り込み許可レジスタ0 (Pin Change Enable Mask 0~7)

ビット (\$6B)	7	6	5	4	3	2	1	0	
	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	PCMSK0
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とピン変化割り込み制御レジスタ(PCICR)のPCIE0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。

(訳補) 28ピン外圍器にはPCINT24~27が存在しないため、PCIE3, PCIF3ビット及びPCMSK3レジスタが利用できません。

## 10. 入出力ポート

### 10.1. 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角をも無意識に変更することなく、1つのポートピンの方角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。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。全てのポートピンに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があります。全てのI/Oピンは図10-1.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134頁の「電気的特性」を参照してください。

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49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41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。

### 10.2. 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10-2.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10-1. 入出力ピン等価回路

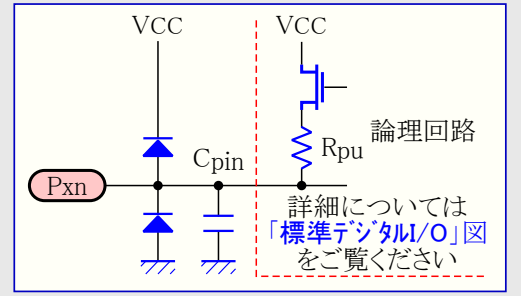
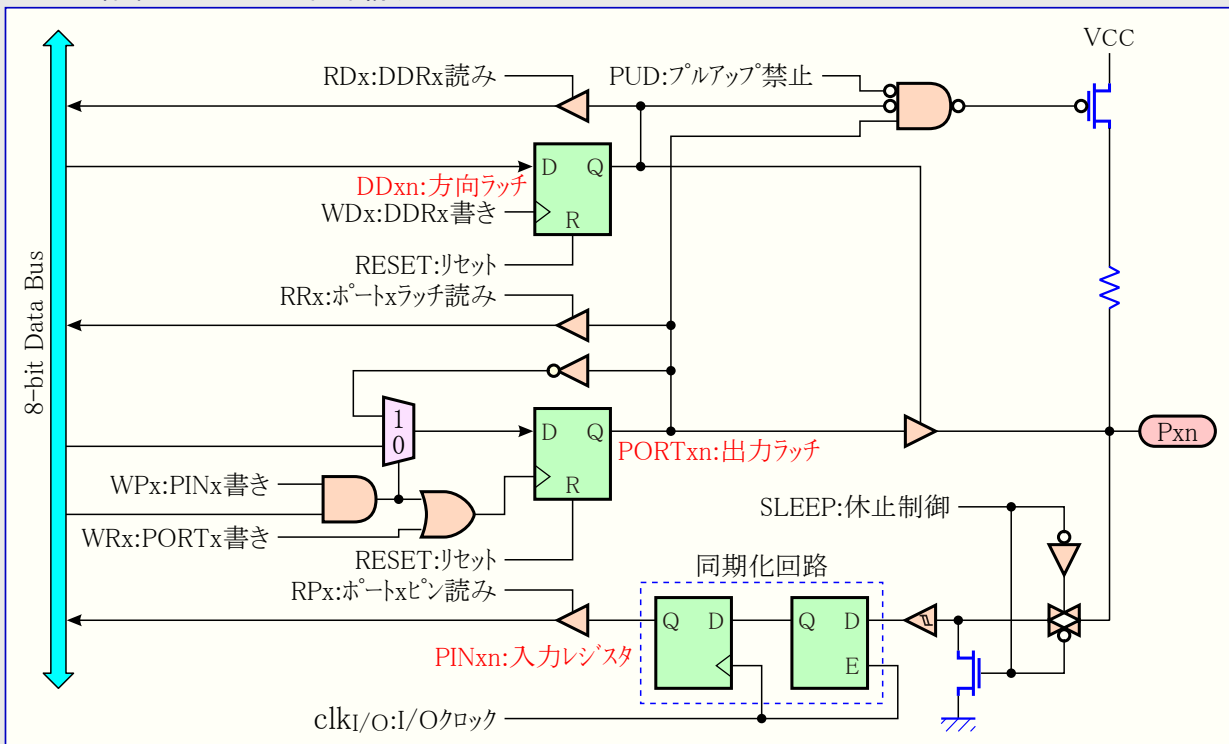


図10-2. 標準デジタル入出力回路構成



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。

### 10.2.1. 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。49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。

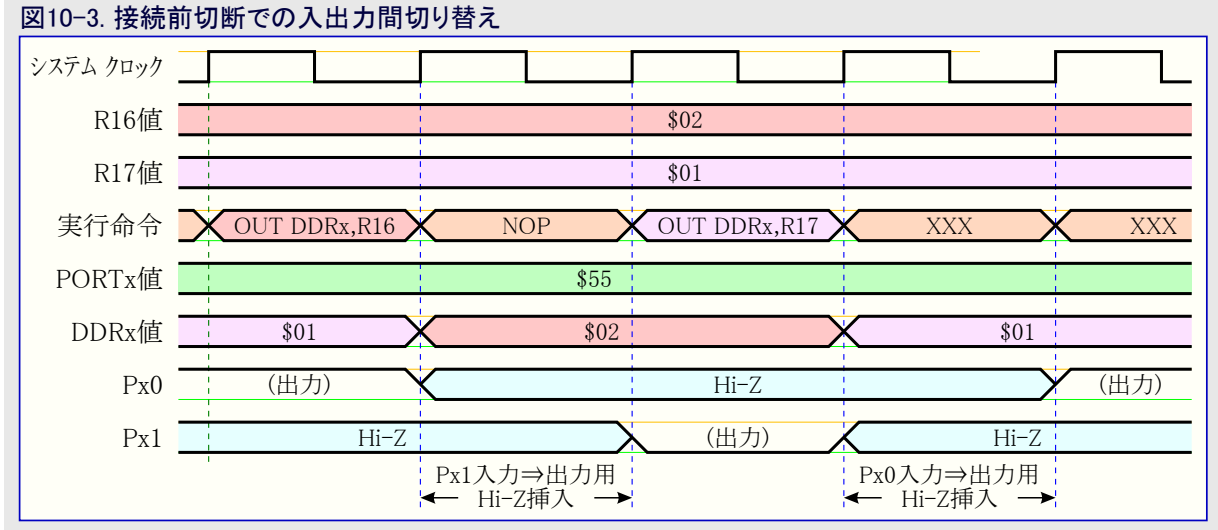
そのピンが出力ピンとして設定される時にPORTxnが論理**1**を書かれると、そのポートピンは**High(1)**に駆動されます。そのピンが出力ピンとして設定される時にPORTxnが論理**0**を書かれると、そのポートピンは**Low(0)**に駆動されます。

### 10.2.2. 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。

### 10.2.3. 接続前切断(Break-Before-Make)切り替え

接続前切断動作ではDDRxnを入力から出力へ切り換える時に1システムクロック持続する隣接Hi-Z区間が**図10-3**で示されるように導入されます。例えば、システムクロックが4MHzでDDRxnが出力にできるように書かれた場合、PORTxnの値がポートピンで見える前に250nsの隣接Hi-Z区間が導入されます。異常を避けるため、DDRxn最大切り替え周波数は2システムクロックが推奨されます。この接続前切断はポート単位動作で、ポート単位の接続前切断許可(BBMx)ビットによって活性にされます。BBMxビットの詳細については49頁の「**PORTCR - ポート制御レジスタ**」をご覧ください。DDRxnビットを出力から入力に切り替える時に隣接Hi-Z区間は導入されません。



### 10.2.4. 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10-1.はピン値に対する制御信号の一覧を示します。

**表10-1. ポートピンの設定**

DDxn	PORTxn	PUD (MCUCR) (注)	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 (シンク)出力
1	1	X	出力	なし	High (ソース)出力

注: または**ポート制御レジスタ(PORTCR)**の**ポート単位プルアップ禁止(PUDx)**ビット

10.2.5. 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10-2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10-4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10-4で)システムクロックの最初の上昇端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(トランスパレント)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10-5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10-4. 外部供給ピン値読み込み時の同期化

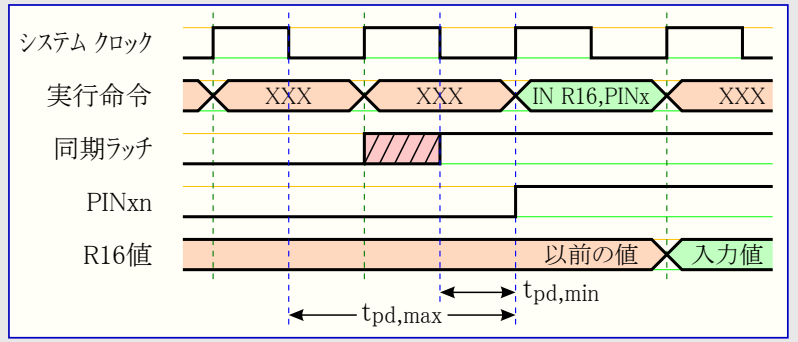
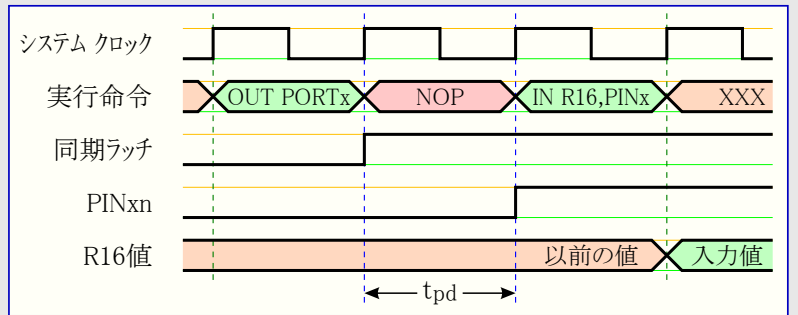


図10-5. プログラムで設定したピン値読み戻し時の同期化



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のに2つの一時レジスタが使われます。

### 10.2.6. デジタル入力許可と休止形態

図10-2.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**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41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ず。

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。

### 10.2.7. 未接続ピン

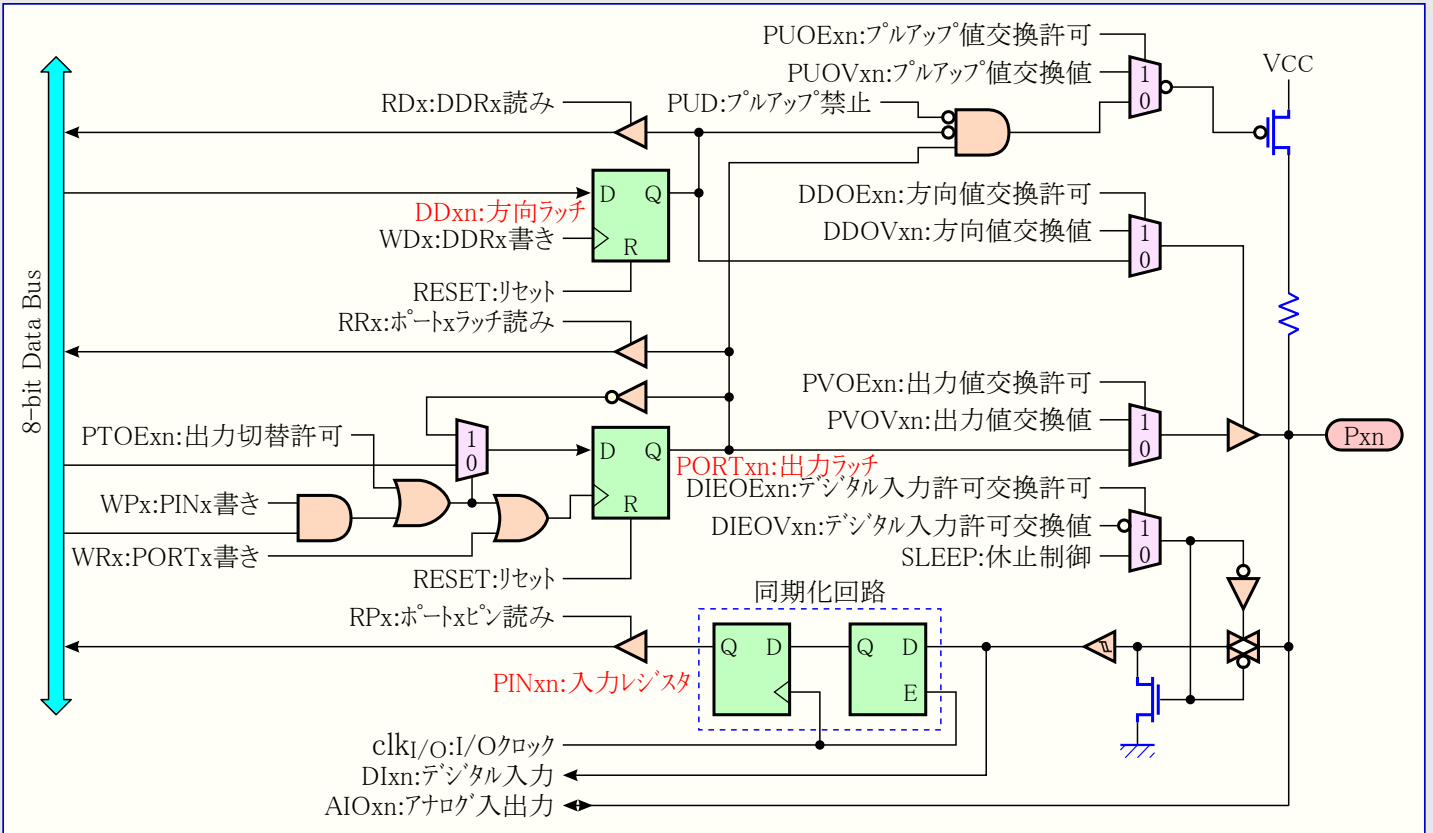
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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。

10.3. 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10-6は単純化された図10-2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10-6. 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10-2は重複(交換)信号の機能一覧を示します。図10-6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10-2. 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。

### 10.3.1. ポートAの交換機能

ポートAピンの交換機能は表10-3.で示されます。

表10-3. ポートAピンの交換機能

ポートピン	交換機能
PA3	PCINT27 (ピン変化割り込み27入力)
PA2	PCINT26 (ピン変化割り込み26入力)
PA1	ADC7 (A/D変換チャンネル7入力) PCINT25 (ピン変化割り込み25入力)
PA0	ADC6 (A/D変換チャンネル6入力) PCINT4 (ピン変化割り込み4入力)

交換ピンの設定は次のとおりです。

- PCINT27 – ポートA ビット3 : PA3  
PCINT27 : ピン変化割り込み27入力。
- PCINT26 – ポートA ビット2 : PA2  
PCINT26 : ピン変化割り込み26入力。
- ADC7/PCINT25 – ポートA ビット1 : PA1  
ADC7 : PA1はA/D変換チャンネル7入力として使えます。  
PCINT25 : ピン変化割り込み25入力。
- ADC6/PCINT24 – ポートA ビット0 : PA0  
ADC6 : PA0はA/D変換チャンネル6入力として使えます。  
PCINT24 : ピン変化割り込み24入力。

表10-4.はポートAの交換機能を41頁の図10-6.で示される交換信号に関連付けます。

表10-4. ポートA3~0の交換機能用交換信号

信号名	PA3/PCINT27	PA2/PCINT26	PA1/PCINT25	PA0/PCINT24
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIEOE	PCINT27・PCIE3	PCINT26・PCIE3	PCINT25・PCIE3	PCINT24・PCIE3
DIEOV	1	1	1	1
DI	PCINT27入力	PCINT26入力	PCINT25入力	PCINT24入力
AIO	-	-	ADC7入力	ADC6入力

## 10.3.2. ポートBの交換機能

ポートBピンの交換機能は表10-5.で示されます。

表10-5. ポートBピンの交換機能

ポートピン	交換機能
PB7	PCINT7 (ピン変化割り込み7入力)
PB6	CLKI (外部クロック信号入力) PCINT6 (ピン変化割り込み6入力)
PB5	SCK (SPI 直列クロック 主装置側出力/従装置側入力) PCINT5 (ピン変化割り込み5入力)
PB4	MISO (SPI 主装置側データ入力/従装置側データ出力) PCINT4 (ピン変化割り込み4入力)
PB3	MOSI (SPI 主装置側データ出力/従装置側データ入力) PCINT3 (ピン変化割り込み3入力)
PB2	$\overline{SS}$ (SPI 従装置選択入力) OC1B (タイマ/カウンタ1 比較B一致出力) PCINT2 (ピン変化割り込み2入力)
PB1	OC1A (タイマ/カウンタ1 比較A一致出力) PCINT1 (ピン変化割り込み1入力)
PB0	ICPI (タイマ/カウンタ1 捕獲起動入力) CLKO (システムクロック出力) 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。

- PCINT7 – ポートB ビット7 : PB7

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。

- CLKI/PCINT6 – ポートB ビット6 : PB6

CLKI : 外部クロック信号入力。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。

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。

PB6がクロックピンとして使われると、PORTB6, DDB6, PINB6は全て0を読みます。

- SCK/PCINT5 – ポートB ビット5 : PB5

SCK : SPIチャンネル用の主装置クロック出力、従装置クロック入力。SPIが従装置として許可されると、本ピンはポートB方向レジスタ(DDRB)のDDB5設定に拘らず、入力として設定されます。SPIが主装置として許可されると、本ピンのデータ方向はDDB5で制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB5で制御できます。

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。

- MISO/PCINT4 – ポートB ビット4 : PB4

MISO : SPIチャンネル用の主装置データ入力、従装置データ出力。SPIが主装置として許可されると、本ピンはポートB方向レジスタ(DDRB)のDDB4の設定に拘らず、入力として設定されます。SPIが従装置として許可されると、本ピンのデータ方向はDDB4で制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB4で制御できます。

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。

- MOSI/PCINT3 – ポートB ビット3 : PB3

MOSI : SPIチャンネル用の主装置データ出力、従装置データ入力。SPIが従装置として許可されると、本ピンはポートB方向レジスタ(DDRB)のDDB3設定に拘らず、入力として設定されます。SPIが主装置として許可されると、本ピンのデータ方向はDDB3で制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB3で制御できます。

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。

- $\overline{SS}$ /OC1B/PCINT2 – ポートB ビット2 : PB2

$\overline{SS}$  : SPI従装置選択入力。SPIが従装置として許可されると、本ピンはポートB方向レジスタ(DDRB)のDDB2の設定に拘らず、入力として設定されます。従装置としてこのピンがLowに駆動されるとSPI(機能)が活性化(有効に)されます。SPIが主装置として許可されると、このピンのデータ方向はDDB2で制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB2で制御できます。

OC1B : タイマ/カウンタ1の比較B一致出力。PB2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。この機能を扱うため、PB2ピンは出力として設定(DDB2=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。

• OC1A/PCINT1 – ホートB ビット1 : PB1

OC1A : タイマ/カウンタ1の比較A一致出力。PB1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。この機能を扱うため、PB1ピンは出力として設定(DDB1=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。

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。

• ICP1/CLKO/PCINT0 – ホートB ビット0 : PB0

ICP1 : タイマ/カウンタ1の捕獲起動入力。PB0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。

CLKO : システム クロック出力。分周したシステム クロックがPB0ピンに出力できます。分周したシステム クロックはCKOUTヒューズがプログラム(0)されると、PORTB0とDDB0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。

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10-6と表10-7はポートBの交換機能を41頁の図10-6で示される交換信号に関連付けます。SPI主装置入力とSPI従装置出力がMISOを構成し、一方MOSIはSPI主装置出力とSPI従装置入力に分けられます。

表10-6. ホートB7~4の交換機能用交換信号

信号名	PB7/PCINT7	PB6/CLKI/PCINT6	PB5/SCK/PCINT5	PB4/MISO/PCINT4
PUOE	0	INTOSC	SPE・MSTR	SPE・MSTR
PUOV	0	0	PORTB5・PUD	PORTB4・PUD
DDOE	0	INTOSC	SPE・MSTR	SPE・MSTR
DDOV	0	0	0	0
PVOE	0	0	SPE・MSTR	SPE・MSTR
PVOV	0	0	SCK出力	SPI従装置出力
PTOE	-	-	-	-
DIEOE	PCINT7・PCIE0	INTOSC+PCINT6・PCIE0	PCINT5・PCIE0	PCINT4・PCIE0
DIEOV	1	INTRC	1	1
DI	PCINT7入力	PCINT6入力	SCK/PCINT5入力	SPI主装置/PCINT4入力
AIO	-	クロック入力	-	-

注: INTOSCは校正付き内蔵発振器が(CKSELヒューズによって)選ばれることを意味します。

表10-7. ホートB3~0の交換機能用交換信号

信号名	PB3/MOSI/PCINT3	PB2/SS/OC1B/PCINT2	PB1/OC1A/PCINT1	PB0/ICP1/CLKO/PCINT0
PUOE	SPE・MSTR	SPE・MSTR	0	0
PUOV	PORTB3・PUD	PORTB2・PUD	0	0
DDOE	SPE・MSTR	SPE・MSTR	0	0
DDOV	0	0	0	0
PVOE	SPE・MSTR	OC1B許可	OC1A許可	0
PVOV	SPI主装置出力	OC1B	OC1A	0
PTOE	-	-	-	-
DIEOE	PCINT3・PCIE0	PCINT2・PCIE0	PCINT1・PCIE0	PCINT0・PCIE0
DIEOV	1	1	1	1
DI	SPI従装置/PCINT3入力	SPI SS/PCINT2入力	PCINT1入力	ICP1/PCINT0入力
AIO	-	-	-	-

## 10.3.3. ポートCの交換機能

ポートCピンの交換機能は表10-8.で示されます。

表10-8. ポートC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
PC7	PCINT15 (ピン変化割り込み15入力)	PC3	ADC3 (A/D変換チャンネル3入力)
PC6	RESET (リセットピン)		PCINT11 (ピン変化割り込み11入力)
	PCINT14 (ピン変化割り込み14入力)	PC2	ADC2 (A/D変換チャンネル2入力)
PC5	ADC5 (A/D変換チャンネル5入力)		PCINT10 (ピン変化割り込み10入力)
	SCL (2線直列バスクロック入出力)	PC1	ADC1 (A/D変換チャンネル1入力)
PCINT13 (ピン変化割り込み13入力)	PCINT9 (ピン変化割り込み9入力)		
PC4	ADC4 (A/D変換チャンネル4入力)	PC0	ADC0 (A/D変換チャンネル0入力)
	SDA (2線直列バスデータ入出力)		PCINT8 (ピン変化割り込み8入力)
PCINT12 (ピン変化割り込み12入力)			

交換ピンの設定は次のとおりです。

### • PCINT15 – ポートC ビット7 : PC7

PCINT15 : ピン変化割り込み15入力。PC7ピンは外部割り込み元としても扱えます。

### • RESET/PCINT14 – ポートC ビット6 : PC6

RESET : リセットピン。RSTDISBLヒューズがプログラム(0)されると、このピンは標準のI/Oピンとして機能し、デバイスは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。

PC6がリセットピンとして使われると、PORTC6, DDC6, PINC6は全て0を読みます。

PCINT14 : ピン変化割り込み14入力。PC6ピンは外部割り込み元としても扱えます。

### • SCL/ADC5/PCINT13 – ポートC ビット5 : PC5

SCL : 2線直列インターフェースクロック。2線直列インターフェースを許可するために2線直列インターフェース制御レジスタ(TWCR)の2線直列インターフェース動作許可(TWEN)ビットが設定(1)されると、PC5は(標準の)ポートから切り離されて、2線直列インターフェース用直列クロック入出力ピンになります。この動作では入力信号上の50nsより短いスパイク(瞬間雑音)を消去するためのスパイク除去器があり、スレーブ(上昇/下降速度)制限付きオープンドレイン駆動部によって駆動されます。

ADC5 : PC5はA/D変換チャンネル5入力としても使えます。この入力はデジタル電源を使うことに注意してください。

PCINT13 : ピン変化割り込み13入力。PC5ピンは外部割り込み元としても扱えます。

### • SDA/ADC4/PCINT12 – ポートC ビット4 : PC4

SDA : 2線直列インターフェースデータ。2線直列インターフェースを許可するために2線直列インターフェース制御レジスタ(TWCR)の2線直列インターフェース動作許可(TWEN)ビットが設定(1)されると、PC4は(標準の)ポートから切り離されて、2線直列インターフェース用直列データ入出力ピンになります。この動作では入力信号上の50nsより短いスパイク(瞬間雑音)を消去するためのスパイク除去器があり、スレーブ(上昇/下降速度)制限付きオープンドレイン駆動部によって駆動されます。

ADC4 : PC4はA/D変換チャンネル4入力としても使えます。この入力はデジタル電源を使うことに注意してください。

PCINT12 : ピン変化割り込み12入力。PC4ピンは外部割り込み元としても扱えます。

### • ADC3/PCINT11 – ポートC ビット3 : PC3

ADC3 : PC3はA/D変換チャンネル3入力としても使えます。この入力はアナログ電源を使うことに注意してください。

PCINT11 : ピン変化割り込み11入力。PC3ピンは外部割り込み元としても扱えます。

### • ADC2/PCINT10 – ポートC ビット2 : PC2

ADC2 : PC2はA/D変換チャンネル2入力としても使えます。この入力はアナログ電源を使うことに注意してください。

PCINT10 : ピン変化割り込み10入力。PC2ピンは外部割り込み元としても扱えます。

### • ADC1/PCINT9 – ポートC ビット1 : PC1

ADC1 : PC1はA/D変換チャンネル1入力としても使えます。この入力はアナログ電源を使うことに注意してください。

PCINT9 : ピン変化割り込み9入力。PC1ピンは外部割り込み元としても扱えます。

### • ADC0/PCINT8 – ポートC ビット0 : PC0

ADC0 : PC0はA/D変換チャンネル0入力としても使えます。この入力はアナログ電源を使うことに注意してください。

PCINT8 : ピン変化割り込み8入力。PC0ピンは外部割り込み元としても扱えます。

表10-9と表10-10はポートCの交換機能を41頁の図10-6.で示される交換信号に関連付けます。

表10-9. ポートC7~4の交換機能用交換信号

信号名	PC7/PCINT15	PC6/RESET/PCINT14	PC5/SCL/ADC5/PCINT13	PC4/SDA/ADC4/PCINT12
PUOE	0	RSTDISBL	TWEN	TWEN
PUOV	0	1	PORTC5·PUD	PORTC4·PUD
DDOE	0	RSTDISBL	TWEN	TWEN
DDOV	0	0	SCL出力	SDA出力
PVOE	0	0	TWEN	TWEN
PVOV	0	0	0	0
PTOE	-	-	-	-
DIEOE	PCINT15·PCIE1	RSTDISBL+PCINT14·PCIE1	ADC5D+PCINT13·PCIE1	ADC4D+PCINT12·PCIE1
DIEOV	1	RSTDISBL	PCINT13·PCIE1	PCINT12·PCIE1
DI	PCINT15入力	PCINT14入力	PCINT13入力	PCINT12入力
AIO	-	リセット入力	ADC5入力/SCL入力	ADC4入力/SDA入力

注: 許可されると、2線直列インターフェースはPC4とPC5ピン出力の slew-rate 制御を許可します。これは図で示されていません。  
 加えて、ポート図で示されるAIO出力と2線直列インターフェース(TWI)部間にスパイク除去器が接続されます。

表10-10. ポートC3~0の交換機能用交換信号

信号名	PC3/ADC3/PCINT11	PC2/ADC2/PCINT10	PC1/ADC1/PCINT9	PC0/ADC0/PCINT8
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIEOE	ADC3D+PCINT11·PCIE1	ADC2D+PCINT10·PCIE1	ADC1D+PCINT9·PCIE1	ADC0D+PCINT8·PCIE1
DIEOV	PCINT11·PCIE1	PCINT10·PCIE1	PCINT9·PCIE1	PCINT8·PCIE1
DI	PCINT11入力	PCINT10入力	PCINT9入力	PCINT8入力
AIO	ADC3入力	ADC2入力	ADC1入力	ADC0入力

## 10.3.4. ポートDの交換機能

ポートDピンの交換機能は表10-11.で示されます。

表10-11. ポートD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
PD7	AIN1 (アナログ比較器反転入力) PCINT23 (ピン変化割り込み23入力)	PD3	INT1 (外部割り込み1 入力) PCINT19 (ピン変化割り込み19入力)
PD6	AIN0 (アナログ比較器非反転入力) PCINT22 (ピン変化割り込み22入力)	PD2	INT0 (外部割り込み0 入力) PCINT18 (ピン変化割り込み18入力)
PD5	T1 (タイマ/カウンタ1 外部クロック入力) PCINT21 (ピン変化割り込み21入力)	PD1	PCINT17 (ピン変化割り込み17入力)
PD4	T0 (タイマ/カウンタ0 外部クロック入力) PCINT20 (ピン変化割り込み20入力)	PD0	PCINT16 (ピン変化割り込み16入力)

交換ピンの設定は次のとおりです。

- AIN1/PCINT23 – ポートD ビット7 : PD7

AIN1 : アナログ比較器反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

PCINT23 : ピン変化割り込み23入力。PD7ピンは外部割り込み元としても扱えます。

- AIN0/PCINT22 – ポートD ビット6 : PD6

AIN0 : **アナログ比較器**非反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

PCINT22 : ピン変化割り込み22入力。PD6ピンは外部割り込み元としても扱えます。

- T1/PCINT21 – ポートD ビット5 : PD5

T1 : **タイマ/カウンタ1**の外部クロック入力ピンです。

PCINT21 : ピン変化割り込み21入力。PD5ピンは外部割り込み元としても扱えます。

- T0/PCINT20 – ポートD ビット4 : PD4

T0 : **タイマ/カウンタ0**の外部クロック入力ピンです。

PCINT20 : ピン変化割り込み20入力。PD4ピンは外部割り込み元としても扱えます。

- INT1/PCINT19 – ポートD ビット3 : PD3

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

PCINT19 : ピン変化割り込み19入力。PD3ピンは外部割り込み元としても扱えます。

- INT0/PCINT18 – ポートD ビット2 : PD2

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

PCINT18 : ピン変化割り込み18入力。PD2ピンは外部割り込み元としても扱えます。

- PCINT17 – ポートD ビット1 : PD1

PCINT17 : ピン変化割り込み17入力。PD1ピンは外部割り込み元としても扱えます。

- PCINT16 – ポートD ビット0 : PD0

PCINT16 : ピン変化割り込み16入力。PD0ピンは外部割り込み元としても扱えます。

表10-12と表10-13はポートDの交換機能を41頁の図10-6.で示される交換信号に関連付けます。

表10-12. ポートD7~4の交換機能用交換信号

信号名	PD7/AIN1/PCINT23	PD6/AIN0/PCINT22	PD5/T1/PCINT21	PD4/T0/PCINT20
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIOE	PCINT23・PCIE2	PCINT22・PCIE2	PCINT21・PCIE2	PCINT20・PCIE2
DIOV	1	1	1	1
DI	PCINT23入力	PCINT22入力	T1/PCINT21入力	T0/PCINT20入力
AIO	AIN1入力	AIN0入力	-	-

表10-13. ポートD3~0の交換機能用交換信号

信号名	PD3/INT1/PCINT19	PD2/INT0/PCINT18	PD1/PCINT17	PD0/PCINT16
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIOE	INT1許可+PCINT19・PCIE2	INT0許可+PCINT18・PCIE2	PCINT17・PCIE2	PCINT16・PCIE2
DIOV	1	1	1	1
DI	INT1/PCINT19入力	INT0/PCINT18入力	PCINT17入力	RXD/PCINT16入力
AIO	-	-	-	-

## 10.4. I/Oポート用レジスタ

### 10.4.1. MCUCR – MCU制御レジスタ (MCU Control Register)

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	–	BODS	BODSE	PUD	–	–	–	–	MCUCR
Read/Write	R	R/W	R/W	R/W	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

- **ビット4 – 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38頁の「[ピンの設定](#)」をご覧ください。

### 10.4.2. PORTCR – ポート制御レジスタ (Port Control Register)

ビット	7	6	5	4	3	2	1	0	
\$12 (\$32)	BBMD	BBMC	BBMB	BBMA	PUDD	PUDC	PUDB	PUDA	PORTCR
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7~4 – BBMx : 接続前切断動作許可 (Break-Before-Make Mode Enable)**

これらのビットが1を書かれると、ポート単位の接続前切断動作が活性(有効)にされます。そして出力にするためのDDRxn書き込み時に仲介Hi-Z周期が挿入されます。更なる情報については38頁の「[接続前切断切り替え](#)」をご覧ください。

- **ビット3~0 – PUDx : ポート単位プルアップ禁止 (Port-Wise Pull-up Disable)**

これらのビットが1を書かれると、例え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、定義された入出力ポートでポート単位のプルアップが禁止されます。これらのポート単位プルアップ禁止ビットはMCU制御レジスタ(MCUCR)の全プルアップ禁止(PUD)ビットと論理和(OR)されます。この特徴についてより多くの詳細に関しては38頁の「[ピンの設定](#)」をご覧ください。

### 10.4.3. PORTA – ポートA出力レジスタ (Port A Data Register) (訳注: 頁下段参照)

ビット	7	6	5	4	3	2	1	0	
\$0E (\$2E)	–	–	–	–	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
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	

### 10.4.4. DDRA – ポートA方向レジスタ (Port A Data Direction Register) (訳注: 頁下段参照)

ビット	7	6	5	4	3	2	1	0	
\$0D (\$2D)	–	–	–	–	DDA3	DDA2	DDA1	DDA0	DDRA
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	

### 10.4.5. PINA – ポートA入力レジスタ (Port A Input Address) (訳注: 頁下段参照)

ビット	7	6	5	4	3	2	1	0	
\$0C (\$2C)	–	–	–	–	PINA3	PINA2	PINA1	PINA0	PINA
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	不定	不定	不定	不定	

### 10.4.6. PORTB – ポートB出力レジスタ (Port B Data Register)

ビット	7	6	5	4	3	2	1	0	
\$05 (\$25)	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	

(訳補) 28ピン外周器にはPA3~0が存在しないため、PORTA,DDRA,PINAレジスタは利用できません。

#### 10.4.7. DDRB – ポートB方向レジスタ (Port B Data Direction Register)

ビット	7	6	5	4	3	2	1	0	
\$04 (\$24)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
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	

#### 10.4.8. PINB – ポートB入力レジスタ (Port B Input Address)

ビット	7	6	5	4	3	2	1	0	
\$03 (\$23)	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	

#### 10.4.9. PORTC – ポートC出力レジスタ (Port C Data Register)

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	PORTC
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	

#### 10.4.10. DDRC – ポートC方向レジスタ (Port C Data Direction Register)

ビット	7	6	5	4	3	2	1	0	
\$07 (\$27)	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
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	

#### 10.4.11. PINC – ポートC入力レジスタ (Port C Input Address)

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	PINC
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	

#### 10.4.12. PORTD – ポートD出力レジスタ (Port D Data Register)

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
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	

#### 10.4.13. DDRD – ポートD方向レジスタ (Port D Data Direction Register)

ビット	7	6	5	4	3	2	1	0	
\$0A (\$2A)	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
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	

#### 10.4.14. PIND – ポートD入力レジスタ (Port D Input Address)

ビット	7	6	5	4	3	2	1	0	
\$09 (\$29)	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
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	

## 11. 8ビット タイマ/カウンタ0

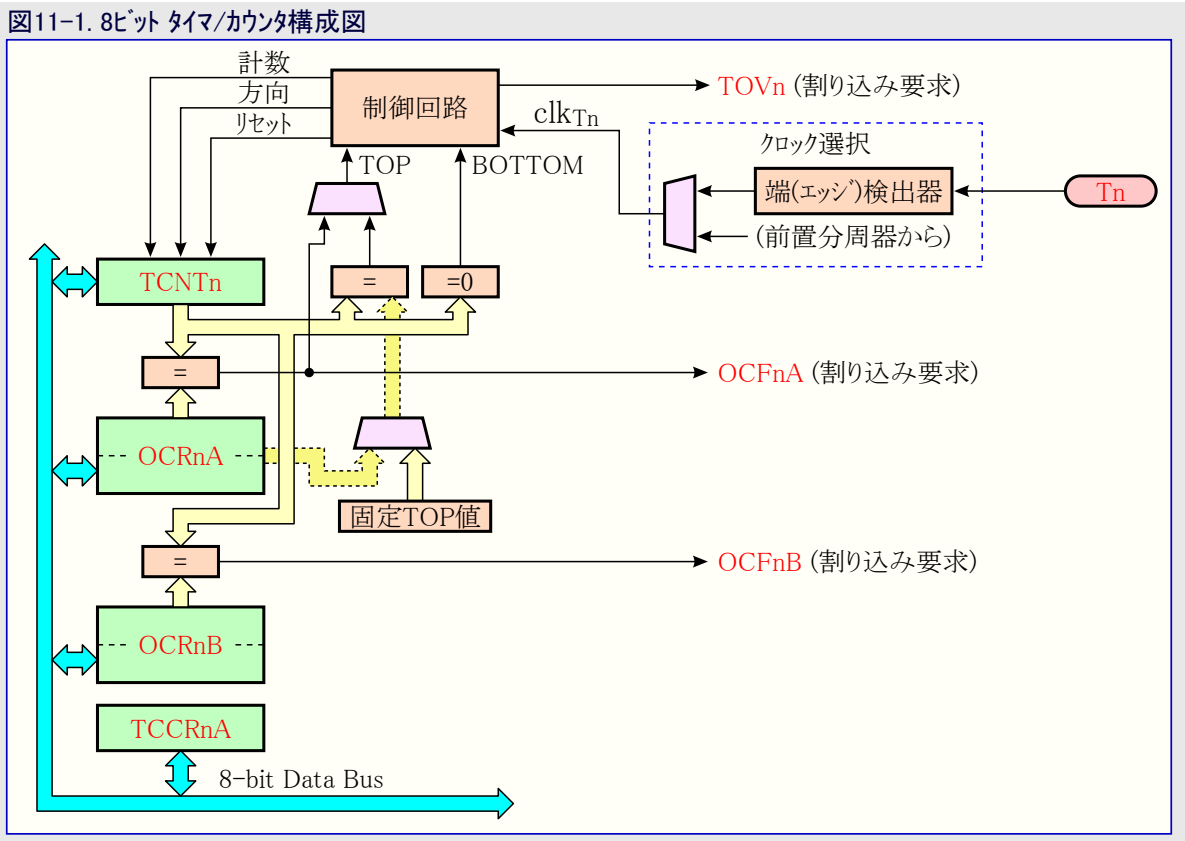
### 11.1. 特徴

- 2つの独立した比較出力部
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 3つの独立した割り込み (TOV0, OCF0A, OCF0B)

### 11.2. 概要

タイマ/カウンタ0は2つの独立した比較出力部付きの汎用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。この8ビット タイマ/カウンタの単純化した構成図は図11-1.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55頁の「8ビット タイマ/カウンタ0用レジスタ」で一覧されます。

24頁の「PRR – 電力削減レジスタ」のPRTIM0ビットはタイマ/カウンタ0部を許可するために0を書かれなければなりません。



#### 11.2.1. 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(TIFR0)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(TIMSK0)で個別に遮蔽(禁止)されます。TIFR0とTIMSK0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(clk\_T0)として参照されます。

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。

#### 11.2.2. 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、小文字の'x'は比較出力部のチャネル名を表し、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11-1.の定義は本文書を広範囲に通して使われます。

表11-1. 用語定義

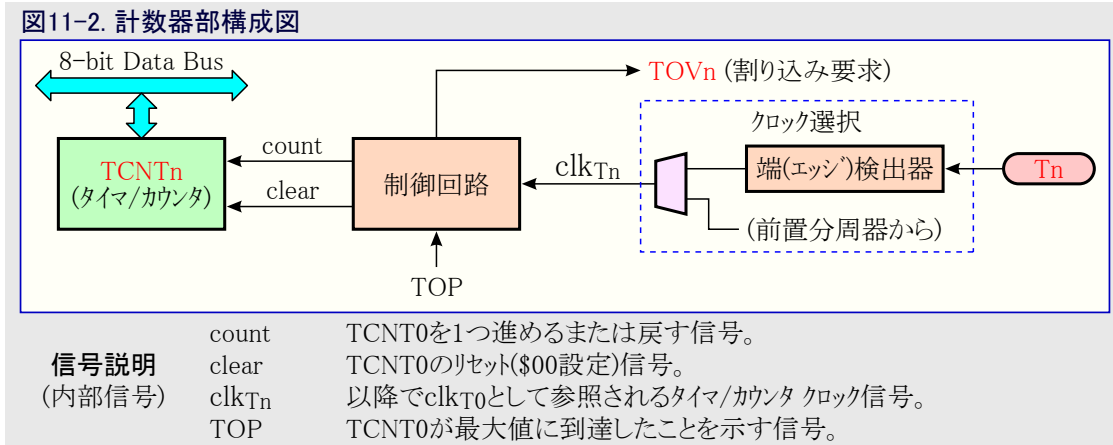
用語	意味
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0A値に到達した時。この指定(TOP)値は動作種別に依存します。

### 11.3. 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A(TCCR0A)**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76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。

### 11.4. 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11-2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って、カウンタは各タイマ/カウンタ クロック(clk<sub>T0</sub>)で解除(\$00)、増加(+1)、または減少(-1)されます。clk<sub>T0</sub>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(clk<sub>T0</sub>)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)。

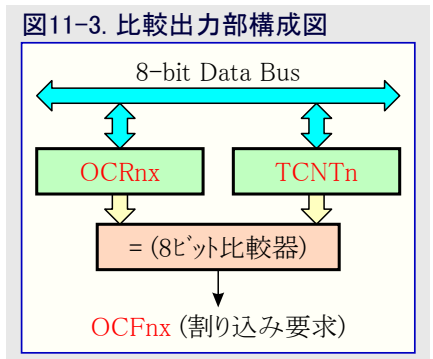
計数順序(方法)は**タイマ/カウンタ制御レジスタA(TCCR0A)**に配置された**比較一致解除許可(CTC0)ビット**の設定によって決定されます。進化した計数順序についてのより多くの詳細に関しては53頁の「**動作種別**」をご覧ください。

**タイマ/カウンタ溢れ(TOV0)フラグ**はCTC0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。

### 11.5. 出力比較部

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、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。

図11-3.は比較出力部の構成図を示します。



#### 11.5.1. 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。この特性はタイマ/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。

#### 11.5.2. 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)、不正な結果に終わります。

## 11.6. 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の動作は比較一致解除許可(CTC0)ビットによって定義されます。

タイミング情報の詳細については54頁の「タイマ/カウンタ0のタイミング」を参照してください。

### 11.6.1. 標準動作

最も単純な動作種別が標準動作(CTC0=0)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0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。

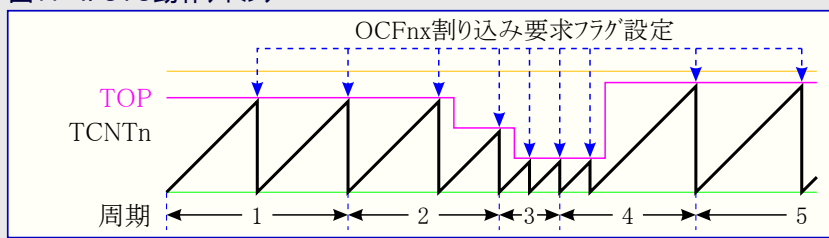
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。

### 11.6.2. 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(CTC0=1)ではOCR0Aがカウンタの分解能を操作するために使われます。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11-4.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。

図11-4. CTC動作タイミング



(訳注) 本図内の周期はOCFnx割り込み時に値反転を行った場合のそれに対する周期で、参考の意味だけです。

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\$00に近い値へTOPを変更することは、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。

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。

## 11.7. 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11-5.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11-5. 前置分周なし(1/1)のタイマ/カウンタ タイミング

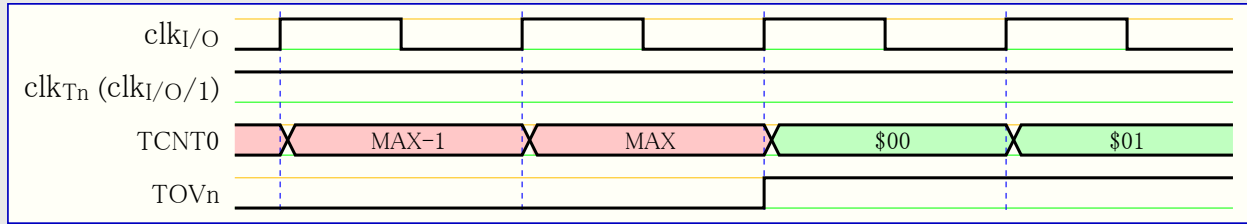


図11-6.は同じタイミング データを示しますが、前置分周器が許可されています。

図11-6. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ タイミング

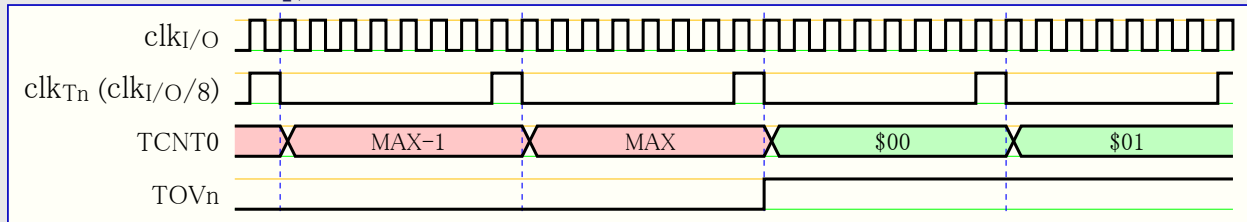


図11-7.はCTC動作を除く全動作種別でのOCF0Aと全動作種別でのOCF0Bの設定を示します。

図11-7. 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

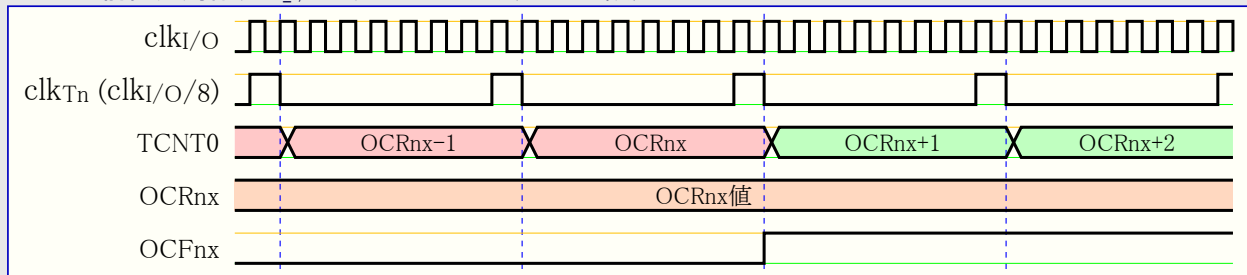
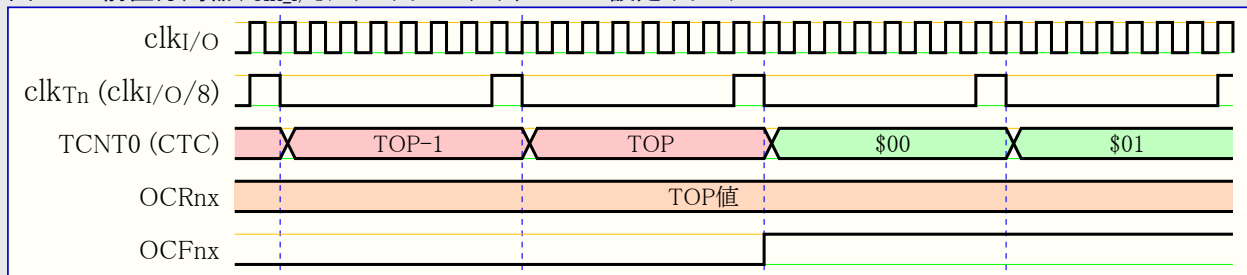


図11-8.はOCR0AがTOPのCTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図11-8. 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



## 11.8. 8ビット タイマ/カウンタ用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$25 (\$45)	–	–	–	–	CTC0	CS02	CS01	CS00	TCCR0A
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7~4 – 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 – CTC0 : 比較一致解除許可 (Clear Timer on Compare Match Mode)

このビットはカウンタの計数手順、最大カウンタ(TOP)値の供給元を制御します。表11-2をご覧ください。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)動作です。53頁の「動作種別」をご覧ください。

表11-2. CTC動作ビット内容

番号	CTC0	タイマ/カウンタ動作種別	TOP値	OCR0x更新時	TOV0設定時
0	0	標準動作	\$FF	即時	MAX
1	1	比較一致タイマ/カウンタ解除(CTC)動作	OCR0A	即時	MAX

注: MAX=\$FFです。

- 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11-3. タイマ/カウンタ0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0動作停止)
0	0	1	clkI/O (前置分周なし)
0	1	0	clkI/O/8 (8分周)
0	1	1	clkI/O/64 (64分周)
1	0	0	clkI/O/256 (256分周)
1	0	1	clkI/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0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。

### 11.8.2. TCNT0 – タイマ/カウンタ (Timer/Counter 0)

ビット	7	6	5	4	3	2	1	0	
\$26 (\$46)	(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。

### 11.8.3. OCR0A – タイマ/カウンタ 比較Aレジスタ (Timer/Counter 0 Output Compare A Register)

ビット	7	6	5	4	3	2	1	0	
\$27 (\$47)	(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を生成するのに使えます。

#### 11.8.4. OCR0B – タイマ/カウンタ 比較レジスタ (Timer/Counter 0 Output Compare B Register)

ビット	7	6	5	4	3	2	1	0	
\$28 (\$48)	(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を生成するのに使えます。

#### 11.8.5. TIMSK0 – タイマ/カウンタ0割り込み許可レジスタ (Timer/Counter 0 Interrupt Mask Register)

ビット	7	6	5	4	3	2	1	0	
(\$6E)	–	–	–	–	–	OCIE0B	OCIE0A	TOIE0	TIMSK0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~3 – 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2 – OCIE0B : タイマ/カウンタ比較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0(TCNT0)で比較一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)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 – OCIE0A : タイマ/カウンタ比較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0(TCNT0)で比較一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)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。

- ビット0 – TOIE0 : タイマ/カウンタ溢れ割り込み許可 (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0(TCNT0)溢れが起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)でタイマ/カウンタ0溢れ割り込み要求フラグ(TOV0)が設定(1)されると、対応する割り込みが実行されます。

#### 11.8.6. TIFR0 – タイマ/カウンタ0割り込み要求フラグレジスタ (Timer/Counter 0 Interrupt Flag Register)

ビット	7	6	5	4	3	2	1	0	
\$15 (\$35)	–	–	–	–	–	OCF0B	OCF0A	TOV0	TIFR0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~3 – 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2 – OCF0B : タイマ/カウンタ比較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0割り込み許可レジスタ(TIMSK0)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 – OCF0A : タイマ/カウンタ比較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0割り込み許可レジスタ(TIMSK0)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。

- ビット0 – TOV0 : タイマ/カウンタ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0割り込み許可レジスタ(TIMSK0)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。

55頁の表11-2.「CTC動作ビット内容」をご覧ください。

## 12. 16ビット タイマ/カウンタ

### 12.1. 特徴

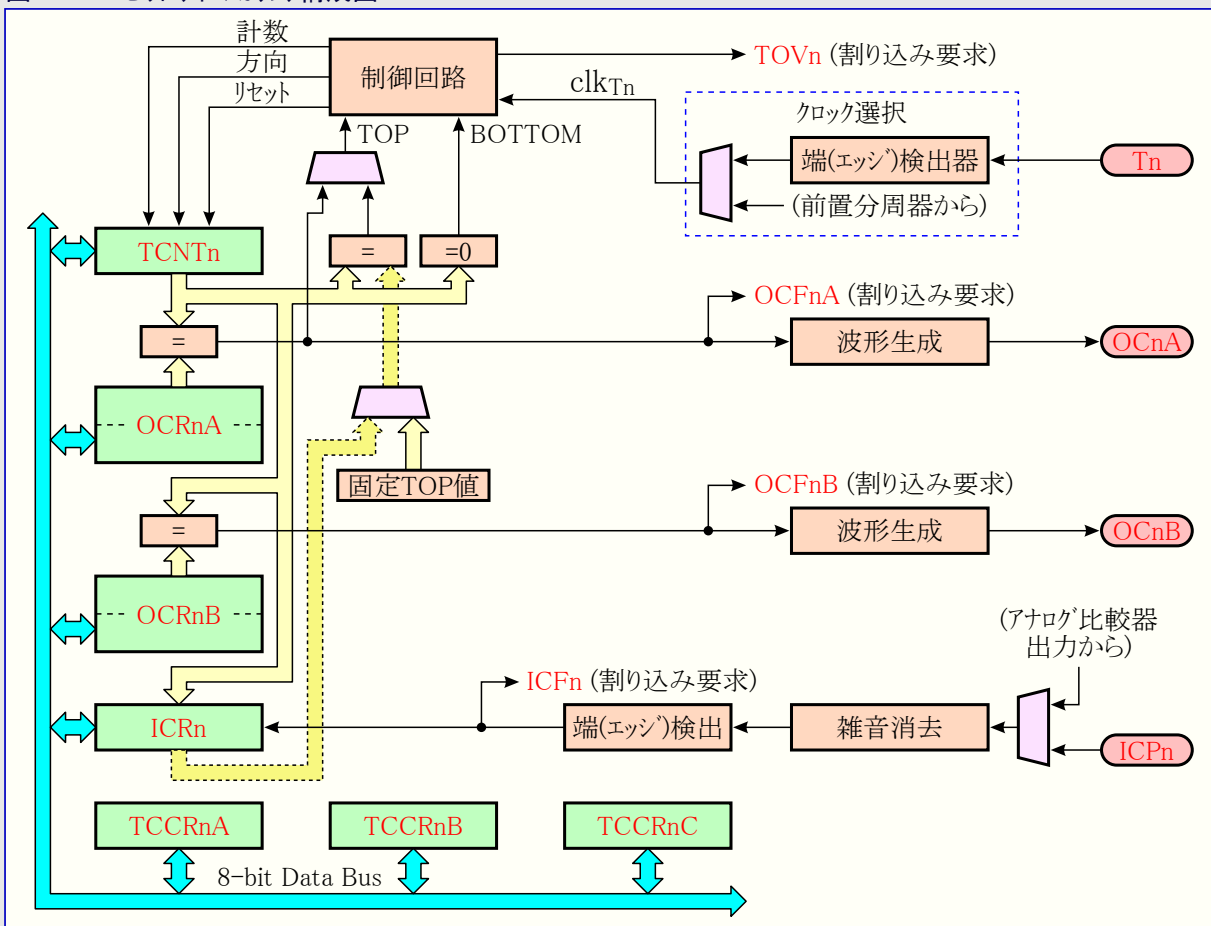
- 真の16ビット設計 (換言すれば16ビットPWMの許容)
- 2つの独立した比較出力部
- 2重緩衝の比較レジスタ
- 1つの捕獲入力部
- 捕獲入力雑音消去器
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 可変PWM周期
- 周波数発生器
- 外部事象計数器
- 4つの独立した割り込み (TOV1,OCF1A,OCF1B,ICF1)

### 12.2. 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、ここで小文字の'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。この16ビットタイマ/カウンタの単純化した構成図は図12-1.で示されます。

図12-1. 16ビット タイマ/カウンタ構成図



注: タイマ/カウンタ1のピンの配置と記述については2頁の「ピン配置」、43頁の表10-5、47頁の表10-11を参照してください。

実際のI/O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。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71頁の「16ビットタイマ/カウンタ用レジスタ」で示されます。

24頁の「PRR - 電力削減レジスタ」のPRTIM1ビットはタイマ/カウンタ1部を許可するために0を書かれなければなりません。

### 12.2.1. 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59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アクセスの制限はありません。割り込み要求信号はタイマ/カウンタ1割り込み要求レジスタ(TIFR1)で全て見えます。全ての割り込みはタイマ/カウンタ1割り込み許可レジスタ(TIMSK1)で個別に遮蔽(禁止)されます。TIFR1とTIMSK1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。63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(ICP1)ピンまたはアナログ比較器出力(105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。捕獲入力部はスパイク雑音を捕らえる機会を軽減するためにデジタル濾波器(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何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出力生成用に使えません。けれどもこの場合、TOP値は動作中に変更されることを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が代わりに使え、PWM出力として使われるべきOCR1Aを開放します。

### 12.2.2. 定義

次の定義は本文書を通じて広範囲に使われます。

表12-1. 用語定義

用語	意味
BOTTOM	計数器が\$0000に到達した時。
MAX	計数器が\$FFFF(65535)に到達した時。
TOP	計数器が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。

## 12.3. 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16ビットレジスタに両方が複写されます。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
;上位バイト設定(一時レジスタ)
;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)
;上位バイト取得(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 */
~ /* 16ビット読み込み */
/* */

```

**注:** 5頁の「コード例について」をご覧ください。

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の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 */
    __disable_interrupt();            /* 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: 5頁の「コード例について」をご覧ください。

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 */
    __disable_interrupt();            /* 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: 5頁の「コード例について」をご覧ください。

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。

### 12.3.1. 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。

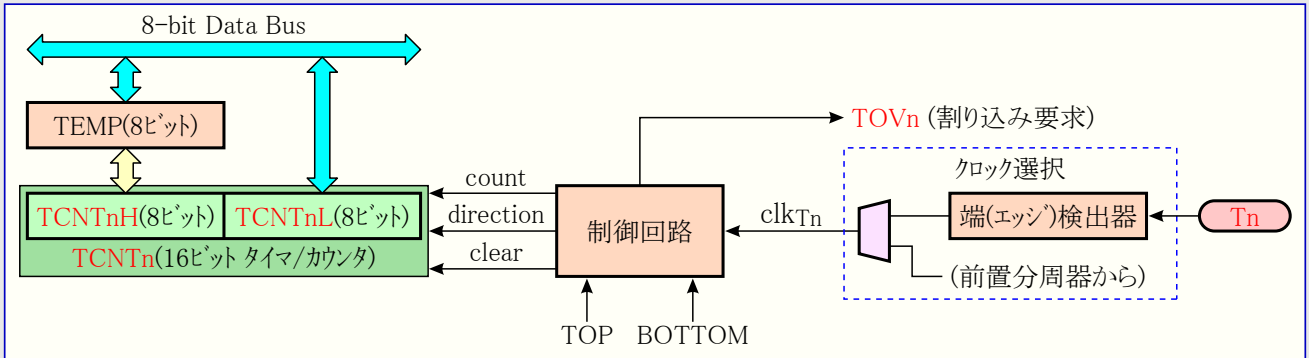
12.4. タイマ/カウンタのクロック

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によって選ばれます。クロック元と前置分周器の詳細については76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。

12.5. 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12-2はこのカウンタとその周辺の構成図を示します。

図12-2. 計数器部構成図



信号説明  
(内部信号)

count	TCNT1を1つ進めるまたは戻す信号。
direction	進行方向(上昇または下降)選択信号。
clear	TCNT1のリセット(\$0000設定)信号。
clkTn	以降でclkT1として参照されるタイマ/カウンタ クロック信号。
TOP	TCNT1が最大値に到達したことを示す信号。
BOTTOM	TCNT1が最小値(\$0000)に到達したことを示す信号。
TEMP	一時レジスタ。

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によって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時に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クロック周期内での16ビット カウンタ値全体の読み書きをCPUに許します。予測不能な結果を与える、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存して、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)。

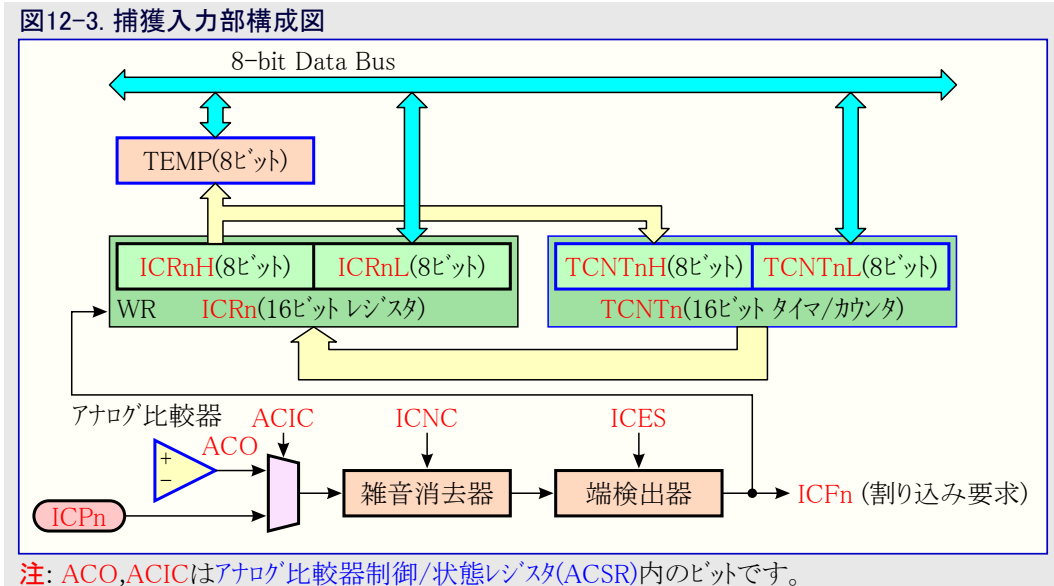
計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCR1A)とタイマ/カウンタ制御レジスタB(TCCR1B)に配置された波形生成種別(WGM13~0)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては66頁の「動作種別」をご覧ください。

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。

## 12.6. 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ICP1ピンまたは代わりにアナログ比較器部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徴の計算に使えます。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12-3.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(ICP1)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)検出器の設定を迫認すると、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に複写されるの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)、その後に上位バイト(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、この一時レジスタをアクセスします。

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だけ書けます。これらの場合、TOP値がICR1に書かれるであろう前に波形生成種別(WGM13~0)ビットが設定されなければなりません。ICR1に書くとき、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59頁の「16ビットレジスタのアクセス」を参照してください。

### 12.6.1. 捕獲起動元

捕獲入力部用の主な起動元は捕獲起動入力(ICP1)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(ICP1)ピンとアナログ比較器出力(ACO)の両入力(T1ピン(76頁の図13-1.参照))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。けれども雑音消去が許可されると、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、雑音消去器と端検出器の入力は常に許可されることに注意してください。

捕獲入力はICP1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。

### 12.6.2. 雑音消去器

雑音消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。雑音消去器の入力は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。

雑音消去器はタイマ/カウンタ制御レジスタB(TCCR1B)の捕獲入力雑音消去許可(ICNC1)ビットの設定(1)によって許可されます。許可した時に雑音消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。雑音消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。

### 12.6.3. 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は新しい値で書き込まれます。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何れかを扱うため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は(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。

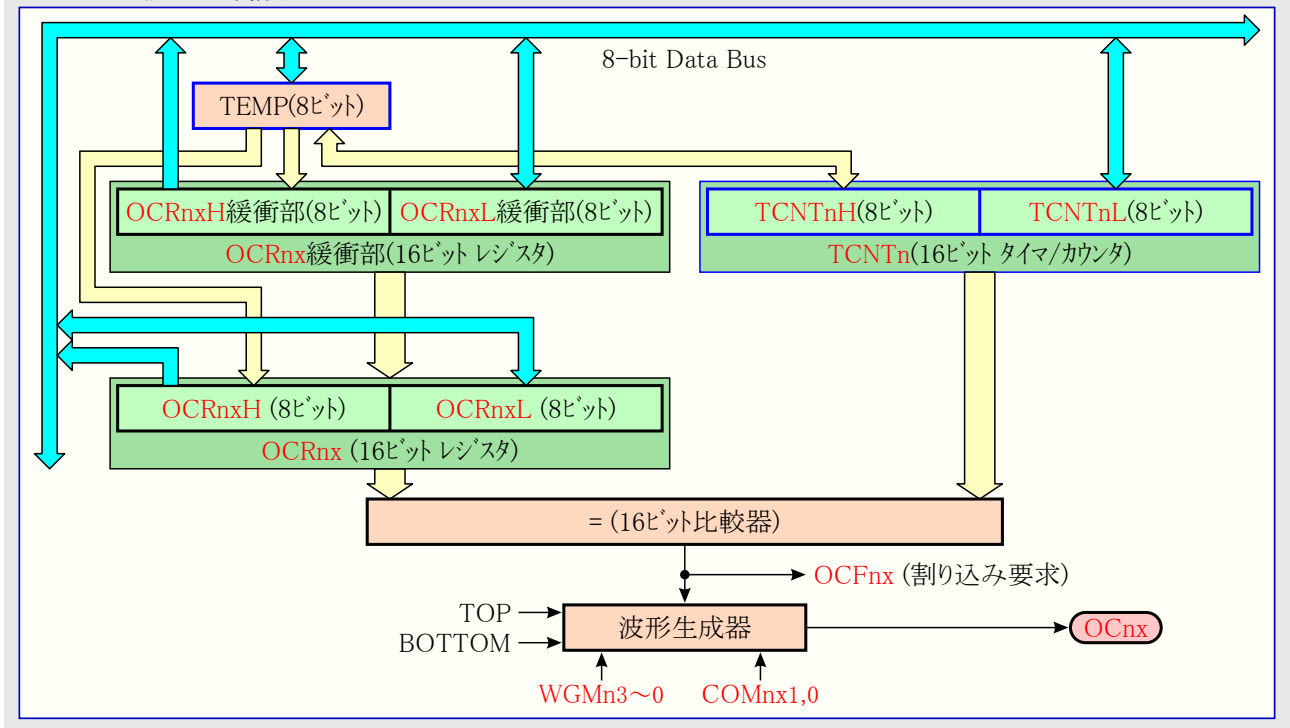
### 12.7. 比較出力部

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が一致を指示します。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が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(66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12-4.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12-4. 比較出力部構成図



OCR1xは12種類のパルス幅変調(PWM)の何れかを使う時に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)は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59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。

### 12.7.1. 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)。

### 12.7.2. 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。

### 12.7.3. 比較一致部の使用

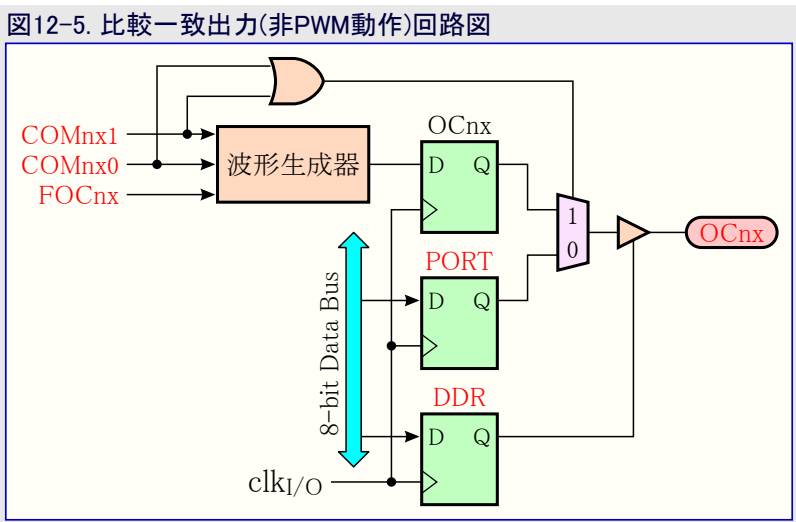
どの動作種別でのTCNT1書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、何れかの比較出力部を使う場合にTCNT1を変更する時は危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。可変TOP値のPWM動作でTOPに等しいTCNT1を書いてはいけません。(行った場合)TOPに対する比較一致は無視され、カウンタは\$FFFFへ(計数を)続けます。同様にカウンタが下降計数のとき、BOTTOMに等しいTCNT1値を書いてはいけません。

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。

## 12.8. 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12-5はCOM1x1,0ビット設定によって影響される論理回路の簡略化された図を示します。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12-2、表12-3、表12-4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。71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。

### 12.8.1. 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。全ての動作種別に対して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71頁の表12-2を参照してください。高速PWM動作については71頁の表12-3、位相基準PWMと位相/周波数基準PWMについては71頁の表12-4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。

## 12.9. 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、否か(反転または非反転PWM)のどちらかを制御します。非PWM動作に対する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。65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70頁の「**タイマ/カウンタのタイミング**」を参照してください。

### 12.9.1. 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。

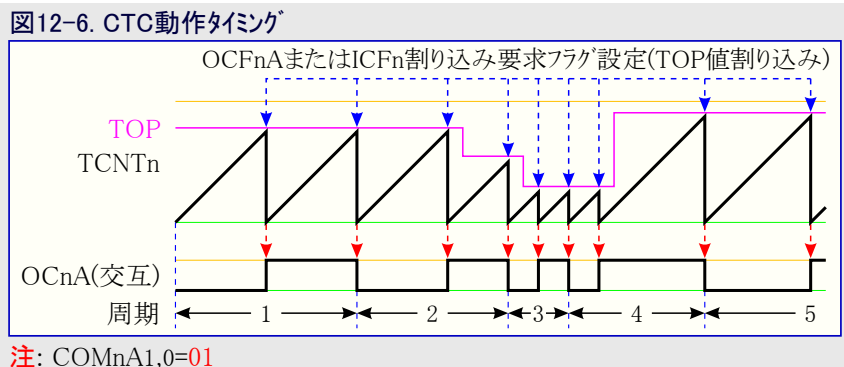
捕獲入力部は標準動作で使用が容易です。けれども外部の事象間の最大間隔がタイマ/カウンタの分解能(16ビット長)を越えてはならないことに気付いてください。事象間の間隔が長すぎる場合、捕獲部に対して分解能を拡張するためにタイマ/カウンタ溢れ割り込みまたは前置分周器が使われなければなりません。

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ために使えます。標準動作で波形を生成するため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。

### 12.9.2. 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、OC1A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と、カウンタは\$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動作に関するタイミング図は図12-6.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OC1A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。OCR1AまたはICR1に書かれた新しい値がTCNT1の現在値より低い(小さい)場合、カウンタは(その回の)比較一致を失います。その後カウンタは比較一致が起こ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りません。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を使う**高速PWM動作(WGM13~0=1111)**を使うことでしょ。

CTC動作で波形出力を生成するために、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)に設定される時に $f_{OC1A} = f_{clk\_I/O} / 2$ 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。

## 12.9.3. 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、TOP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、TOP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、または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(右)式を使うことによって計算できます。

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

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)の何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12-7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。本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時に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。その後カウンタは比較一致を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値を使う時に上手くなります。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。71頁の表12-3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。

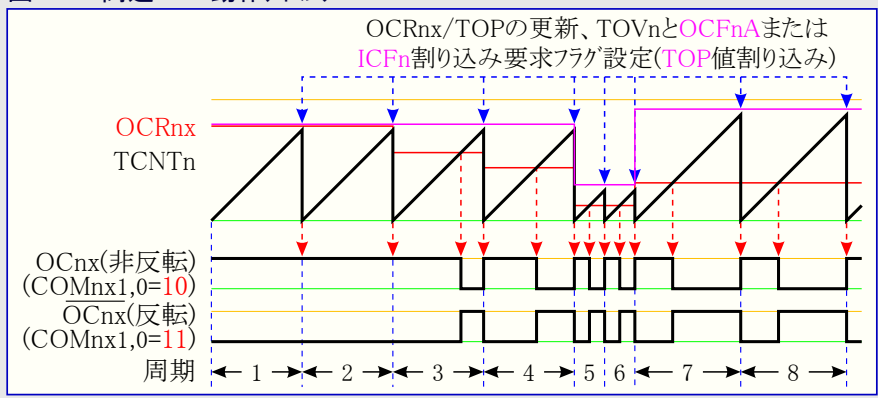
PWM出力周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

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

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。OCR1xがTOPに等しく設定されると、(COM1x1,0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。

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12-7. 高速PWM動作タイミング

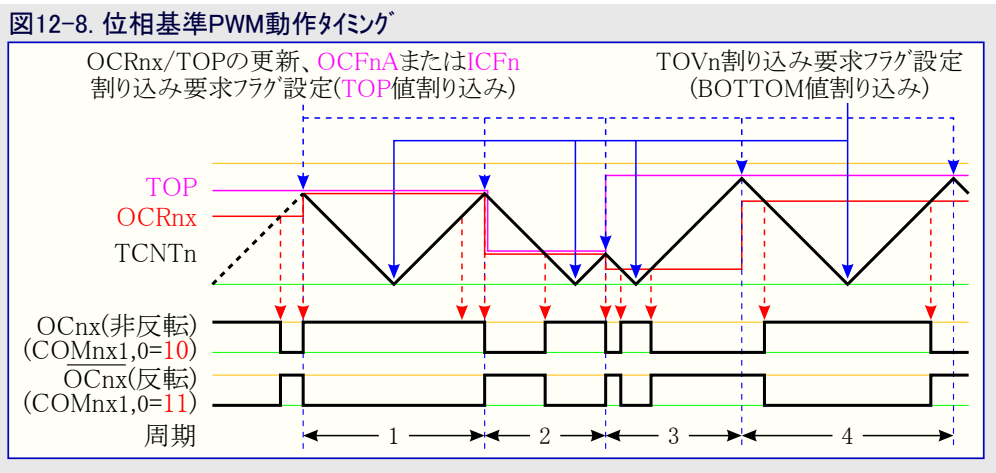


### 12.9.4. 位相基準PWM動作

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)の何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12-8.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値を変更する時にプログラムは新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12-8.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こと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)が違うとき、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値を使うとき、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(71頁の表12-4.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。

12.9.5. 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(訳補:TOPとBOTTOM)です(図12-8と図12-9参照)。

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12-9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。

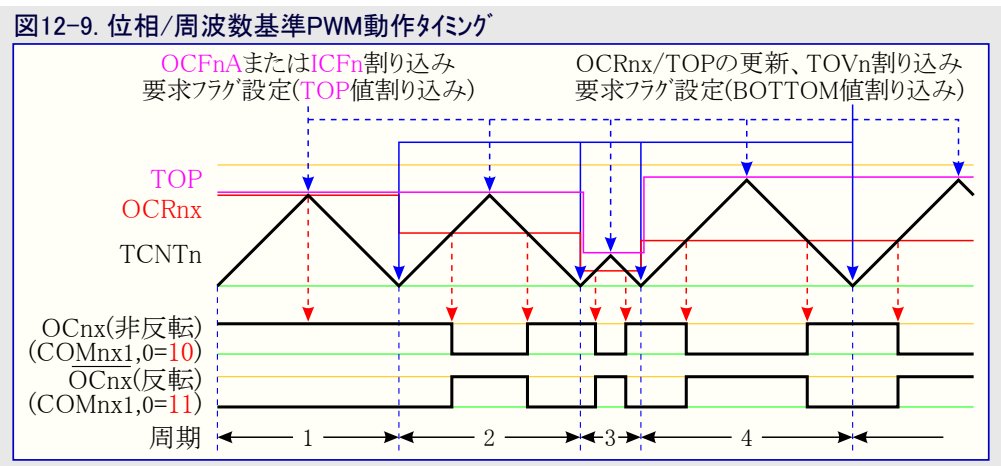


図12-9. 位相/周波数基準PWM動作タイミング

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値を変更する時にプログラムは新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12-9.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(71頁の表12-4.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あたります。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。



## 12.10. 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重緩衝を使う種別のみ)の情報を含まず、図12-10は $OCF1x$ の設定に関するタイミング図を示します。

図12-10. 前置分周なし(1/1)のタイマ/カウンタ、 $OCF1x$ 設定 タイミング

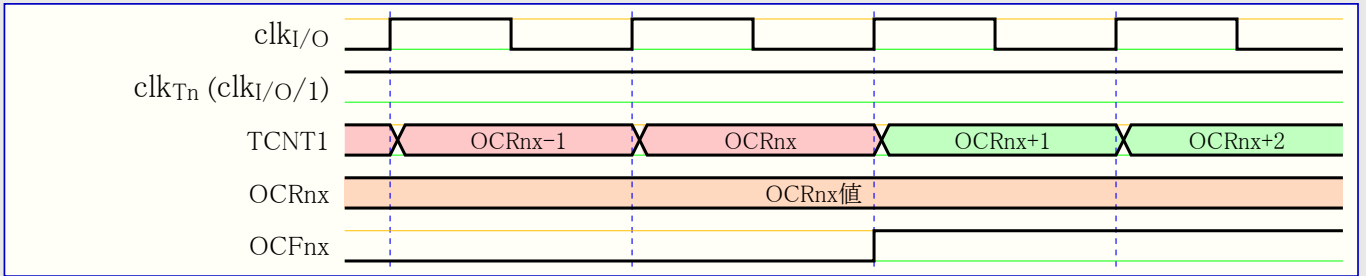


図12-11. は同じタイミング データを示しますが、前置分周器が許可されています。

図12-11. 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

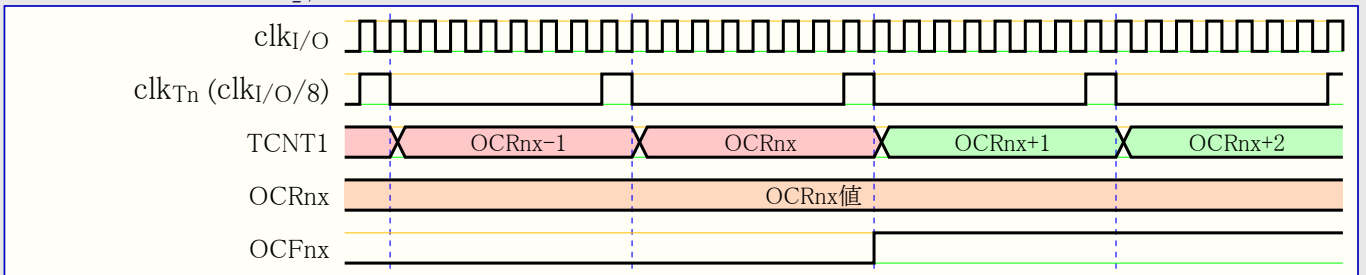


図12-12. は各動作種別でのTOP近辺の計数手順を示します。位相/周波数基準PWM動作使用時の $OCR1x$ レジスタはBOTTOMで更新されます。タイミング図は同じになりますが、当然TOPはBOTTOM、TOP-1はBOTTOM+1などのように置き換えられます。BOTTOMで $TOV1$ を設定(1)する動作種別についても、同様な名称変更が適用されます。

図12-12. 前置分周なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング

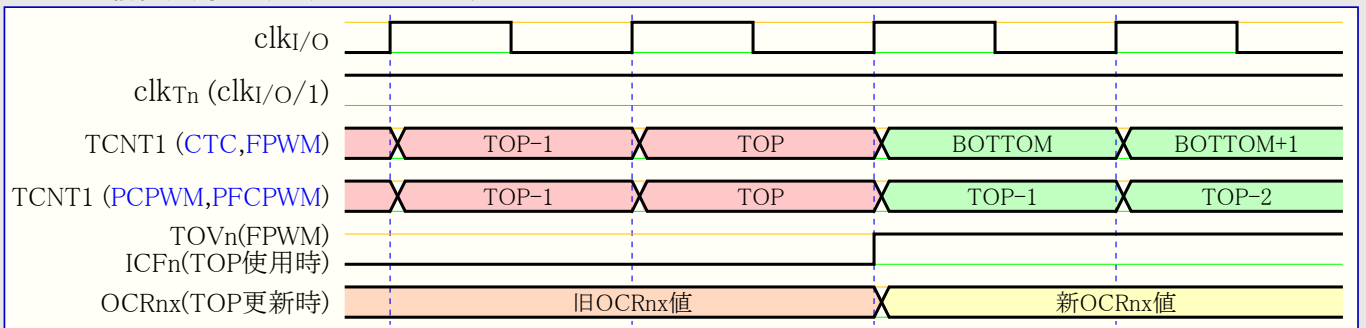
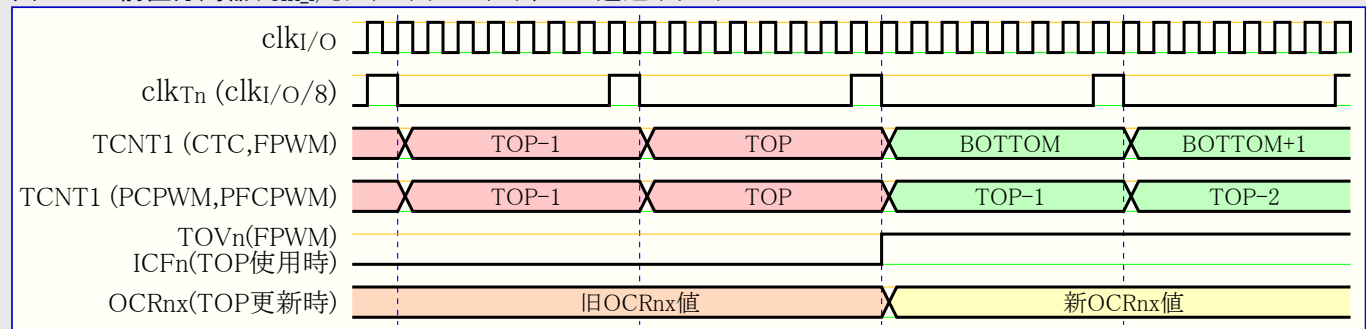


図12-13. は同じタイミング データを示しますが、前置分周器が許可されています。

図12-13. 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



## 12.11. 16ビット タイマ/カウンタ1用レジスタ

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

ビット (\$80)	7	6	5	4	3	2	1	0	TCCR1A
	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	
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に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12-2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12-2. 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12-3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12-3. 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、TOPでHighをOC1xピンへ出力 (非反転動作)
1	1	比較一致でHigh、TOP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、TOP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67頁の「高速PWM動作」をご覧ください。

表12-4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12-4. 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=10XX : 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68頁の「位相基準PWM動作」をご覧ください。

- 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 – 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12-5.参照)。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。66頁の「動作種別」をご覧ください。

表12-5. 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	TOP	TOP
6	0	1	1	0	9ビット高速PWM動作	\$01FF	TOP	TOP
7	0	1	1	1	10ビット高速PWM動作	\$03FF	TOP	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	TOP	TOP
15	1	1	1	1	高速PWM動作	OCR1A	TOP	TOP

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

ビット (\$81)	7	6	5	4	3	2	1	0	TCCR1B
	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	
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1雑音消去許可 (Input Capture1 Noise Canceler)

このビットを(1)に設定することは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(ICP1)ピンからの入力が濾波されます。この濾波器機能はそれが出力を更新することに対して連続4回等しく評価されたICP1ピンの採取を必要とします。雑音消去器が許可されると、捕獲入力はこのによって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ために使われる捕獲起動入力(ICP1)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)、ICP1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12-10と図12-11をご覧ください。

表12-6. 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 <sub>I/O</sub> (前置分周なし)
0	1	0	clk <sub>I/O</sub> /8 (8分周)
0	1	1	clk <sub>I/O</sub> /64 (64分周)
1	0	0	clk <sub>I/O</sub> /256 (256分周)
1	0	1	clk <sub>I/O</sub> /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。

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

ビット (\$82)	7	6	5	4	3	2	1	0	TCCR1C
	FOC1A	FOC1B	-	-	-	-	-	-	
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動作時にTCCR1Cが書かれるとき、これらのビットは0を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。

•ビット5~0 – 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。

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

ビット (\$85)	15	14	13	12	11	10	9	8	TCNT1H
	(MSB)								
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 (\$84)	7	6	5	4	3	2	1	0	TCNT1L
								(LSB)	
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。59ページの「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。



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

ビット (\$89)	15	14	13	12	11	10	9	8	OCR1AH
	(MSB)								
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 (\$88)	7	6	5	4	3	2	1	0	OCR1AL
								(LSB)	
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	

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

ビット (\$8B)	15	14	13	12	11	10	9	8	OCR1BH
	(MSB)								
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 (\$8A)	7	6	5	4	3	2	1	0	OCR1BL
								(LSB)	
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。59ページの「16ビットレジスタのアクセス」をご覧ください。

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

ビット (\$87)	15	14	13	12	11	10	9	8	ICR1H
	(MSB)								
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 (\$86)	7	6	5	4	3	2	1	0	ICR1L
								(LSB)	
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ICP1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ことを保証するため、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。59ページの「16ビットレジスタのアクセス」をご覧ください。

## 12.11.8. TIMSK1 – タイマ/カウンタ1割り込み許可レジスタ (Timer/Counter1 Interrupt Mask Register)

ビット (\$6F)	7	6	5	4	3	2	1	0	TIMSK1
	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1	
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット5 – 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1割り込み要求フラグレジスタ(TIFR1)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(31頁の「割り込み」参照)が実行されます。

### • ビット2 – 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1割り込み要求フラグレジスタ(TIFR1)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(31頁の「割り込み」参照)が実行されます。

### • ビット1 – 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1割り込み要求フラグレジスタ(TIFR1)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(31頁の「割り込み」参照)が実行されます。

### • ビット0 – 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1割り込み要求フラグレジスタ(TIFR1)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(31頁の「割り込み」参照)が実行されます。

## 12.11.9. TIFR1 – タイマ/カウンタ1割り込み要求フラグレジスタ (Timer/Counter1 Interrupt Flag Register)

ビット (\$16 (\$36))	7	6	5	4	3	2	1	0	TIFR1
	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1	
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット5 – 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)

ICP1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。

### • ビット2 – 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1 – 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0 – TOV1 : タイマ/カウンタ1溢れ割り込み要求フラグ (Timer/Counter1 Overflow Flag)

このフラグの設定(1)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72頁の表12-5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。

(注) 本頁レジスタ内のビット7,6,4,3は予約されており、常に0として読みます。

### 13. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器

51頁の「8ビット タイマ/カウンタ0」と57頁の「16ビット タイマ/カウンタ1(PWM付き)」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。

#### 13.1. 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。

#### 13.2. 前置分周器リセット

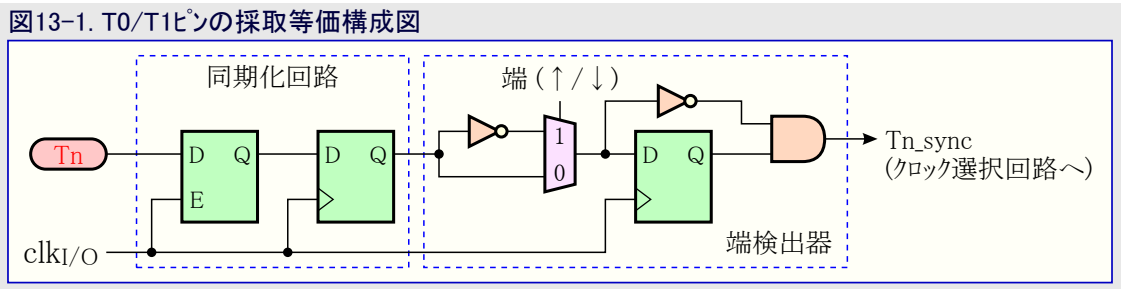
この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ0とタイマ/カウンタ1によって共有されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=010,011,100,101)時に前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステム クロック周期数は、Nが前置分周値(8, 64, 256, 1024)とすると、 $1 \sim N+1$  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。

#### 13.3. 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13-1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。



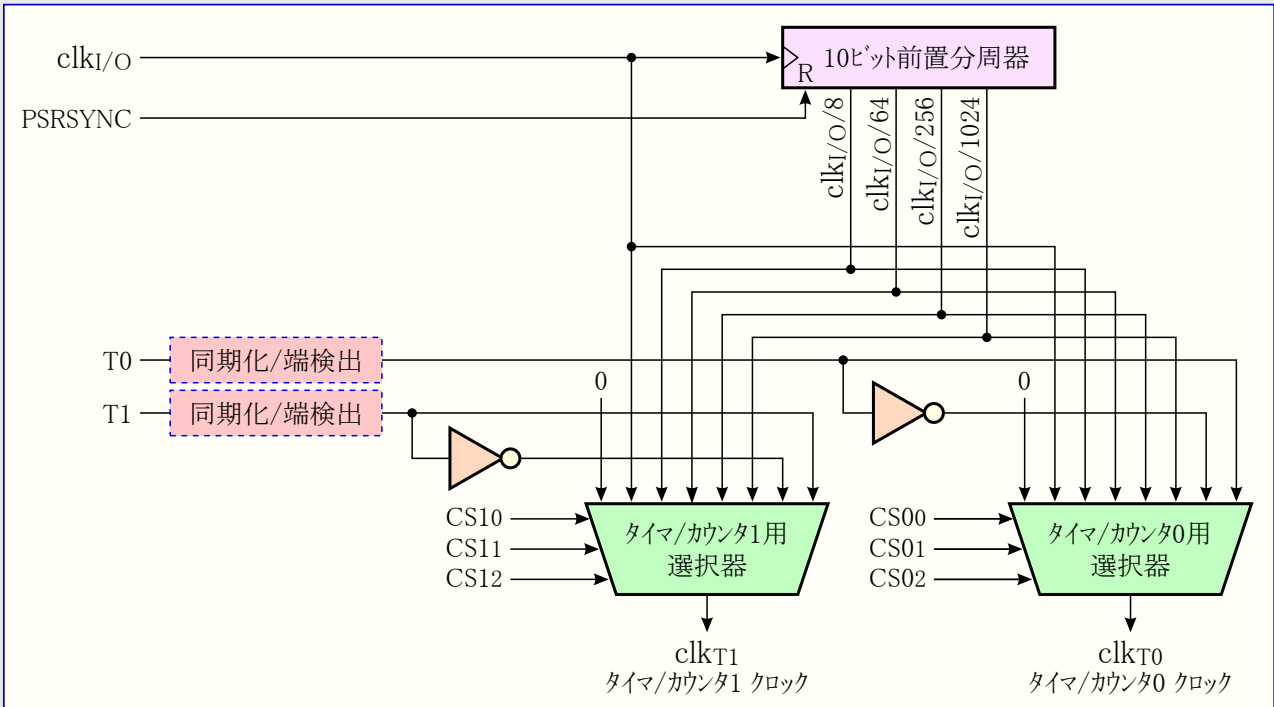
同期化と端検出器論理回路はT0/T1ピンへ印加された端からカウンタが更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

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%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13-2. 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13-1.で示されます。

### 13.4. タイマ/カウンタ前置分周器制御関係レジスタ

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

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

• **ビット7 – TSM : タイマ/カウンタ同時動作 (Timer/Counter Synchronization Mode)**

TSMビットへの1書き込みはタイマ/カウンタ同期化動作を活性(有効)にします。この動作でPSRSYNCへ書かれる値は保持され、従って対応する前置分周器リセット信号の有効を保持します。これは対応するタイマ/カウンタを停止し、設定中にそれらの1つが進行する危険なしに同じ値に構成設定できることを保証します。TSMビットが0を書かれると、PSRSYNCビットはハードウェアによって解除(0)され、同時にタイマ/カウンタが計数を始めます。

• **ビット6~1 – 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 – PSRSYNC : 同期系タイマ/カウンタ前置分周器リセット (Prescaler Reset Timer/Counter 1,0)**

このビットが1のとき、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。TSMビットが設定(1)されている場合を除き、通常、このビットはハードウェアによって直ちに解除(0)されます。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。

## 14. 直列周辺インターフェース (SPI: Serial Peripheral Interface)

### 14.1. 特徴

- 全二重3線同期データ転送
- 主装置/従装置動作
- LSB/MSB先行データ転送
- 設定変更可能な7つのビット速度
- 送信完了割り込み要求フラグ
- 送信上書きフラグ保護
- アイドル動作からの起動
- 倍速(CK/2)主装置SPI動作

### 14.2. 概要

直列周辺インターフェースはATtiny44/48と様々なAVRデバイスや周辺デバイス間の高速同期データ転送を許します。

24頁の「PRR - 電力削減レジスタ」でのPRSPIビットはSPI部を許可するために0を書かれなければなりません。

SPIでの主装置と従装置のCPU間相互連結は図14-2.で示されます。このシステムは2つの移動レジスタと主装置クロック発生器から成ります。SPI主装置は希望した従装置のSS(従装置選択)ピンをLowへ引き込む時に一群の通信を開始します。主装置と従装置は各々の移動レジスタに送出すべきデータを用意し、主装置はデータを交換するために必要なクロックパルスをSCK信号線に生成します。データは常にMOSI(Master Out Slave In)信号線を主装置から従装置へ、MISO(Master In Slave Out)信号線を従装置から主装置へ移動されます。各データパケット後、主装置はSS(従装置選択)ピンをHighへ引き上げることによって従装置と同期を取ります。

主装置として設定されると、SPIインターフェースにはSS信号線の自動制御がありません。これは通信を開始できるのに先立ってユーザーソフトウェアによって操作されなければなりません。これが行われると、SPIデータレジスタ(SPDR)へのバイト書き込みがSPIクロック発生器を開始し、ハードウェアが従装置内へ8ビットを移動します。1バイトの移動後、SPIクロック発生器は停止し、SPI状態レジスタ(SPSR)の転送完了フラグ(SPIF)を設定(1)します。SPI制御レジスタ(SPCR)でSPI割り込み許可(SPIE)ビットが設定(1)されていれば割り込みが要求されます。主装置はSPDR内へ次バイトを書くことによって次バイトの移動を継続、またはSS(従装置選択)信号線をHighへ引き上げることによってパケットの終了を指示することができます。最後の到着バイトはその後の使用のため、緩衝レジスタ内に保持されます。

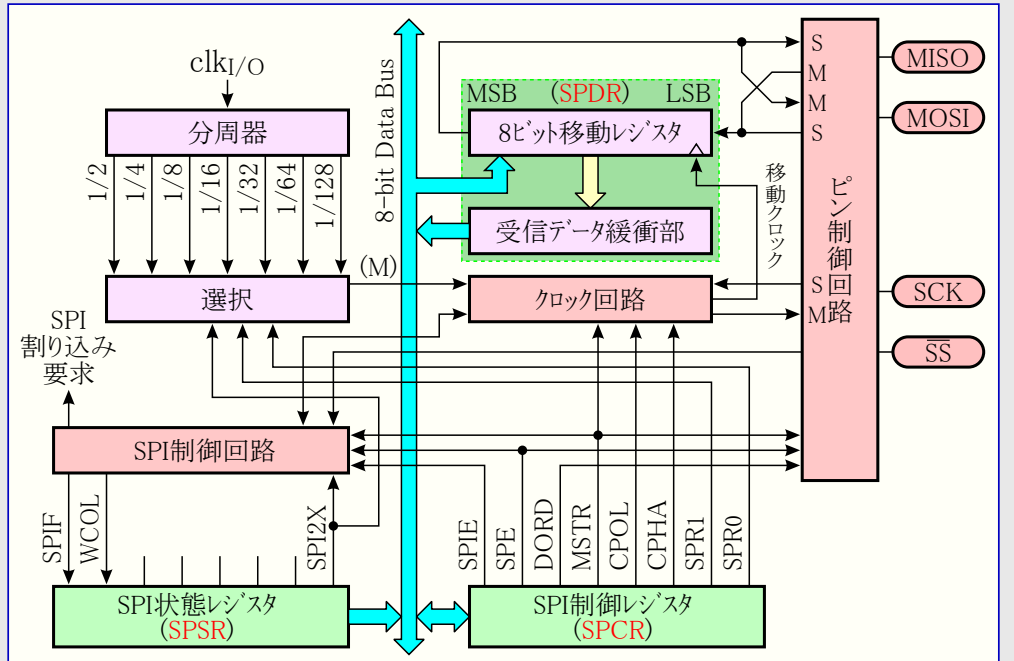
従装置として設定されると、SPIインターフェースはSSピンがHighに駆動される限り、MISOをHi-Zにした休止状態に留まります。この状態でプログラムはSPIデータレジスタ(SPDR)の内容を更新できますが、そのデータはSSピンがLowに駆動されるまでSCKピンでの到着クロックパルスによって移動出力されません。1バイトが完全に移動されてしまうと転送完了フラグ(SPIF)が設定(1)されます。SPCRでSPI割り込み許可(SPIE)ビットが設定(1)されていれば割り込みが要求されます。従装置は受信データを読む前にSPDR内へ送られるべき次のデータの配置を続けられます。最後の到着バイトはその後の使用のため、緩衝レジスタ内に保持されます。

このシステムは送信側単一緩衝、受信側2重緩衝です。これは一連の移動全体が完了される前に送信されるべきバイトがSPIデータレジスタ(SPDR)へ書けないことを意味します。けれども、データを受信するとき、次のデータが完全に移動入力される前に受信したデータがSPIデータレジスタ(SPDR)から読まなければならない。さもなければ始めのバイトが失われます。

SPI従装置動作では制御論理回路がSCKピンの到着信号を採取します。このクロック信号の正しい採取を保証するため、High/Lowの最小時間は各々2 CPUクロック周期より長くあるべきです。

SPIが許可されると、MOSI, MISO, SCK, SSピンのデータ方向は表14-1.に従って無視されます。自動的なポート無視のより多くの詳細については41頁の「交換ポート機能」を参照してください。

図14-1. SPI構成図



注: SPIピン配置については2頁の「ピン配置」と43頁の表10-5.を参照してください。

図14-2. SPI主装置/従装置の連結

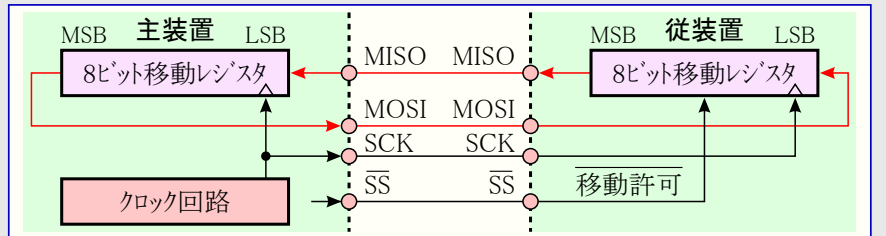


表14-1. SPIピン方向規定

ピン名	主装置時の方向規定	従装置時の方向規定
SCK	ポートB方向レジスタ(DDRB)の指定	入力
MISO	入力	ポートB方向レジスタ(DDRB)の指定
MOSI	ポートB方向レジスタ(DDRB)の指定	入力
SS	ポートB方向レジスタ(DDRB)の指定	入力

注: 使用者定義SPIピンの方向定義方法の詳細記述については43頁の「ポートBの交換機能」を参照してください。

次のコード例は主装置としてSPIを初期化する方法と簡単な送信を実行する方法を示します。例でのDDR\_SPIはSPIピンを制御する実際のポート方向レジスタに置き換えられなければなりません。DD\_MOSI, DD\_MISO, DD\_SCKはこれらのピンに対する実際のポート方向ビットに置き換えられなければなりません。例えばMOSIがPB5ピンに配置されるなら、DD\_MOSIはDDB5、DDR\_SPIはDDRBに置き換ええます。

## アセンブリ言語プログラム例

```
SPI_M_Init: LDI    R17, (1<<DD_MOSI) | (1<<DD_SCK)      ;MOSI, SCK=出力、他は入力値を取得
            OUT    DDR_SPI, R17                        ;MOSI, SCK=出力、他は入力に設定
            LDI    R17, (1<<SPE) | (1<<MSTR) | (1<<SPRO) ;SPI許可、主装置、16分周値を取得
            OUT    SPCR, R17                          ;SPI許可、主装置、16分周に設定
            RET                                         ;呼び出し元へ復帰

SPI_M_Tx:   OUT    SPDR, R16                          ;データ(R16)送信開始
SPI_M_Tx_W: SBIS   SPSR, SPIF                          ;転送完了ならばスキップ
            RJMP  SPI_M_Tx_W                          ;転送完了まで待機
;
            RET                                         ;呼び出し元へ復帰
```

## C言語プログラム例

```
void SPI_MasterInit(void)
{
    DDR_SPI = (1<<DD_MOSI) | (1<<DD_SCK);          /* MOSI, SCK=出力、他は入力に設定 */
    SPCR = (1<<SPE) | (1<<MSTR) | (1<<SPRO);        /* SPI許可、主装置、16分周に設定 */
}

void SPI_MasterTransmit(char cData)
{
    SPDR = cData;                                  /* データ送信開始 */
    while(!(SPSR & (1<<SPIF)));                    /* 転送完了まで待機 */
}
```

注: 5頁の「コード例について」をご覧ください。

次のコード例は従装置としてSPIを初期化する方法と簡単な受信を実行する方法を示します。

## アセンブリ言語プログラム例

```
SPI_S_Init: LDI    R17, (1<<DD_MISO)                  ;MISO出力、他は入力値を取得
            OUT    DDR_SPI, R17                      ;MISO出力、他は入力に設定
            LDI    R17, (1<<SPE)                     ;SPI許可値を取得
            OUT    SPCR, R17                          ;SPI許可設定
            RET                                         ;呼び出し元へ復帰

SPI_S_Rx:   SBIS   SPSR, SPIF                          ;受信(転送)完了ならばスキップ
            RJMP  SPI_S_Rx                          ;受信(転送)完了まで待機
;
            IN     R16, SPDR                          ;受信データを取得
            RET                                         ;呼び出し元へ復帰
```

## C言語プログラム例

```
void SPI_SlaveInit(void)
{
    DDR_SPI = (1<<DD_MISO);                          /* MISO出力、他は入力に設定 */
    SPCR = (1<<SPE);                                  /* SPI許可設定 */
}

char SPI_SlaveReceive(void)
{
    while(!(SPSR & (1<<SPIF)));                      /* 受信(転送)完了まで待機 */
    return SPDR;                                      /* 受信データと共に復帰 */
}
```

注: 5頁の「コード例について」をご覧ください。

## 14.3. $\overline{SS}$ ピンの機能

### 14.3.1. 従装置動作

SPIが従装置として設定されると、従装置選択( $\overline{SS}$ )ピンは常に入力です。 $\overline{SS}$ がLowに保たれるとSPIは活性に(作動)され、使用者によってそのように設定されていればMISOが出力になります。他の全てのピンは入力です。 $\overline{SS}$ がHighに駆動されると、出力として使用者設定され得るMISOを除く全てのピンは入力、SPIは非活動で、それは到着データを受信しないことを意味します。SPI論理回路は一旦 $\overline{SS}$ ピンがHighに駆動されると、リセットすることに注意してください。

この $\overline{SS}$ ピンはパケット/バイト同期に対して、従装置ビットカウンタが主装置クロック発生器との同期を保つのに有用です。 $\overline{SS}$ ピンがHighに駆動されると、SPI従装置は直ちに送受信論理回路をリセットし、それは移動レジスタ内で部分的に受信したどのデータも取り落とします。

### 14.3.2. 主装置動作

SPIが主装置(SPI制御レジスタ(SPCR)の主装置許可(MSTR)ビット=1)として設定されると、 $\overline{SS}$ ピンの方向は使用者が決められます。

$\overline{SS}$ が出力として設定されると、このピンはSPIシステムに影響を及ぼされない標準出力ピンです。代表的にはこのピンがSPI従装置の $\overline{SS}$ ピンを駆動するでしょう。

$\overline{SS}$ が入力として設定されると、SPI主装置動作を保証するためにそれはHighに保持されなければなりません。 $\overline{SS}$ ピンが入力として定義されたSPI主装置として設定されるとき、周辺回路によって $\overline{SS}$ ピンがLowに駆動されると、SPIシステムは他の主装置が従装置として選んでデータ送信を始めると解釈します。バスの衝突を避けるためにSPIシステムは次の動作を行います。

1. SPCRで主装置許可(MSTR)ビットが解除(0)され、SPIシステムは従装置になります。SPIシステムが従装置になる結果としてMOSIとSCKピンが入力になります。
2. SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)され、そしてSPI割り込みが許可(SPCRのSPIE=1)され、且つ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)なら、割り込みルーチンが実行されます。

従って割り込み駆動SPI送信が主装置動作で使われ、 $\overline{SS}$ がLowに駆動される可能性があるとき、その割り込み(処理)はMSTRビットが未だ設定(1)されているのを常に検査すべきです。MSTRビットが従装置選択によって解除(0)されてしまっていると、それはSPI主装置動作を再び許可するため、使用者によって設定(1)されなければなりません。

### 14.4. データ転送形式

直列データに関してはSPI制御レジスタ(SPCR)のSCK位相(CPHA)とSCK極性(CPOL)制御ビットによって決定されるSCK位相と極性で4つの組み合わせがあります。このSPIデータ転送形式は図14-3.と図14-4.で示されます。

図14-3. SPIデータ転送形式 (CPHA=0)

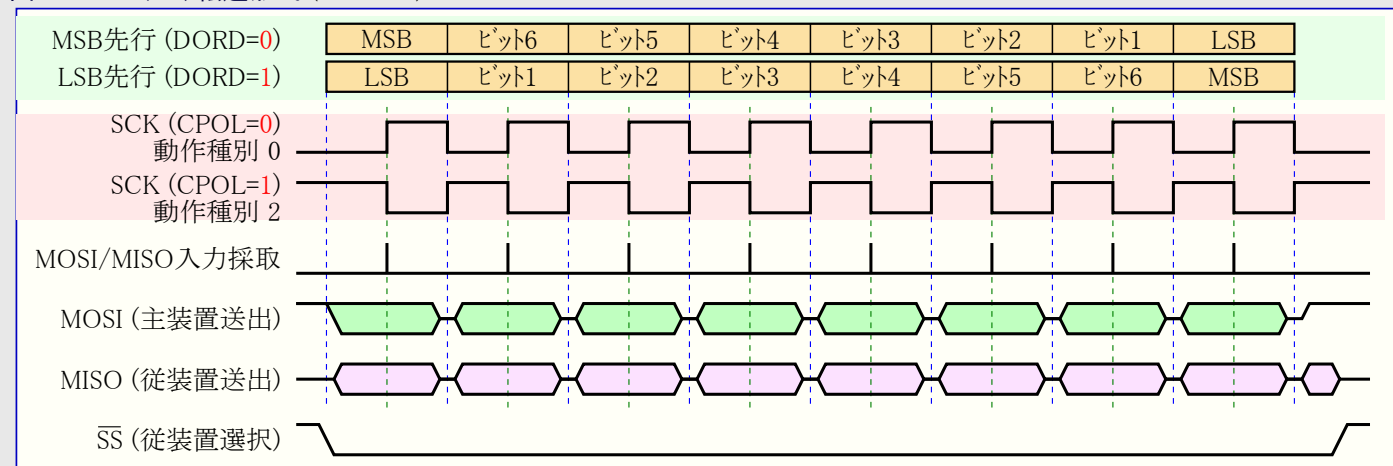
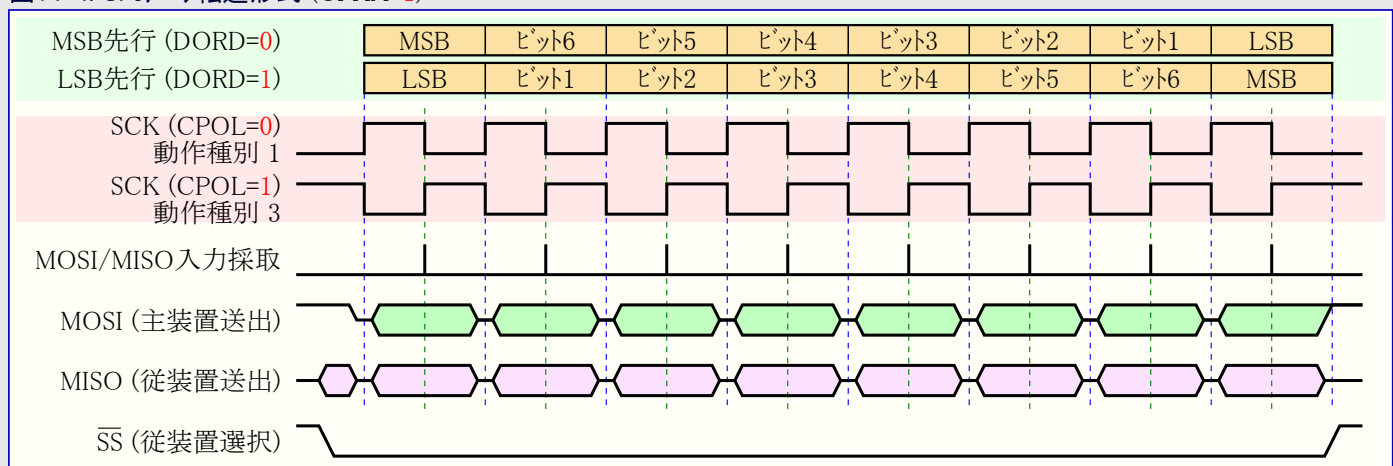


図14-4. SPIデータ転送形式 (CPHA=1)



データビットは安定のためデータ信号に対して十分な時間を保証するSCK信号の反対端で移動出力と(入力)ラッチが行われます。これは表14-2.で行われるように表14-3.と表14-4.を要約することによって明解にされます。

表14-2. CPOL,CPHA機能動作

SPI動作種別番号	CPOL	CPHA	SCK先行端	SCK後行端
0	0	0	入力採取/上昇端	出力設定/下降端
1	0	1	出力設定/上昇端	入力採取/下降端
2	1	0	入力採取/下降端	出力設定/上昇端
3	1	1	出力設定/下降端	入力採取/上昇端

## 14.5. SPI用レジスタ

### 14.5.1. SPCR – SPI制御レジスタ (SPI Control Register)

ビット	7	6	5	4	3	2	1	0	
\$2C (\$4C)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
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 – SPIE : SPI割り込み許可 (SPI 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)されて、SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)されると、このビットがSPI割り込みを実行させます。

#### • ビット6 – SPE : SPI許可 (SPI Enable)

SPEビットが1を書かれるとSPIが許可されます。どのSPI操作を許可するにも、このビットが設定(1)されなければなりません。

#### • ビット5 – DORD : データ順選択 (Data Order)

DORDビットが1を書かれるとデータ語のLSBが最初に転送されます。DORDビットが0を書かれるとMSBが最初に転送されます。

#### • ビット4 – MSTR : 主装置/従装置選択 (Master/Slave Select)

このビットは1を書かれると主装置動作、論理0を書かれると従装置動作を選びます。 $\overline{SS}$ が入力として設定され、MSTRが設定(1)の間にLowへ駆動されると、MSTRが解除(0)されてSPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)になります。その後使用者はSPI主装置動作を再び許可するためにMSTRを設定(1)しなければなりません。

#### • ビット3 – CPOL : SCK極性選択 (Clock Polarity)

このビットが1を書かれると、アイドル時にSCKはHighです。CPOLが0を書かれると、アイドル時にSCKはLowです。例については図14-3と図14-4を参照してください。CPOL機能は右で要約されます。

表14-3. CPOL機能動作

CPOL	SCK先行端	SCK後行端
0	上昇端	下降端
1	下降端	上昇端

#### • ビット2 – CPHA : SCK位相選択 (Clock Phase)

このSCK位相選択(CPHA)ビットの設定はデータがSCKの先行(先)端または後行(後)端で採取/(設定)されるかを決めます。例については図14-3と図14-4を参照してください。CPHA機能は右で要約されます。

表14-4. CPHA機能動作

CPHA	SCK先行端	SCK後行端
0	入力採取	出力設定
1	出力設定	入力採取

#### • ビット1,0 – SPR1,0 : SPIクロック選択 (SPI Clock Rate Select 1 and 0)

これら2ビットは主装置として設定されたデバイスのSCK速度を制御します。従装置でのSPR1とSPR0は無効です。SCKと(システム)発振器クロック周波数 $f_{osc}$ 間の関連は次表で示されます。

表14-5. SCK速度選択 ( $f_{osc}$ =CPUクロック周波数)

SPR1	0	0	1	1
SPR0	0	1	0	1
SPI2X	1	0	1	0
SCK周波数	$f_{osc}/2$	$f_{osc}/4$	$f_{osc}/8$	$f_{osc}/16$
	$f_{osc}/32$	$f_{osc}/64$	$f_{osc}/128$	

### 14.5.2. SPSR – SPI状態レジスタ (SPI Status Register)

ビット	7	6	5	4	3	2	1	0	
\$2D (\$4D)	SPIF	WCOL	-	-	-	-	-	SPI2X	SPSR
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 – SPIF : SPI割り込み要求フラグ (SPI Interrupt Flag)

直列転送が完了すると、このSPIFフラグが設定(1)されます。全割り込みが許可(ステータスレジスタ(SREG)の全割り込み許可(I)ビット=1)されて、SPI制御レジスタ(SPCR)でSPI割り込み許可(SPIE)ビットが設定(1)されるなら、割り込みが生成されます。SPIが主装置動作の時に $\overline{SS}$ ピンが入力でLowに駆動されるなら、これもこのSPIFフラグを同様に設定(1)します。対応する割り込み処理ベクタを実行するとき、SPIFはハードウェアによって解除(0)されます。代わりにSPIFが設定(1)されたSPI状態レジスタ(SPSR)を始めに読み、その後SPIデータレジスタ(SPCR)をアクセスすることによっても、SPIFフラグは解除(0)されます。

- **ビット6 – WCOL : 上書き発生フラグ (Write Collision Flag)**

データ転送中にSPIデータレジスタ(SPDR)が書かれると、このWCOLビットが設定(1)されます。WCOLビット(とSPIFビット)はWCOLが設定(1)されたSPI状態レジスタ(SPSR)を始めに読み、その後にSPIデータレジスタ(SPDR)をアクセスすることによって解除(0)されます。

- **ビット5~1 – Res : 予約 (Reserved Bit)**

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 – SPI2X : SPI倍速許可 (Double SPI Speed Bit)**

このビットが論理1を書かれると、SCK速度(SCK周波数)はSPIが主装置動作のとき、倍にされます(表14-5参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定されるとき、SPIは $f_{OSC}$ (CPUクロック周波数)/4またはそれ以下の動作のみ保証されます。

ATtiny48/88のSPIインターフェースはフラッシュメモリやEEPROMの書き換え(読み書き)にも使われます。直列プログラミングと照合については131頁をご覧ください。

### 14.5.3. SPDR – SPIデータレジスタ (SPI Data Register)

ビット	7	6	5	4	3	2	1	0	
\$2E (\$4E)	(MSB)							(LSB)	SPDR
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	

SPIデータレジスタはSPI移動レジスタとレジスタファイル(汎用レジスタ)間のデータ転送に使われる読み書き可能なレジスタです。このレジスタへの書き込みはデータ送信を開始します。このレジスタの読み込みは移動レジスタの受信緩衝部読み出しを引き起こします。

## 15. 2線インターフェース (TWI:Two Wire Interface)

### 15.1. 特徴

- Phillips社I<sup>2</sup>C適合
- SMBus適合(条件付)
- 2本のバス信号線のみ必要な、単純ながら強力な柔軟な通信インターフェース
- 主装置動作と従装置動作の両方を支援
- 送信装置または受信装置として動作可能
- 7ビットのアドレス空間が128までの異なる従装置アドレスを許容
- 複数主装置の調停支援
- 従装置動作で400kHzまでのデータ転送速度
- 上昇/下降(スルーレート)制限された出力駆動回路
- バス信号線のスパイクを排除する雑音消去回路
- 一斉呼び出しを含む完全に設定変更可能な従装置アドレスの支援
- AVRが休止形態の時にアドレス認証(一致)が起動

24頁の「PRR - 電力削減レジスタ」のPRTWIビットは2線直列インターフェース部を許可するために0を書かれなければなりません。

### 15.2. 概要

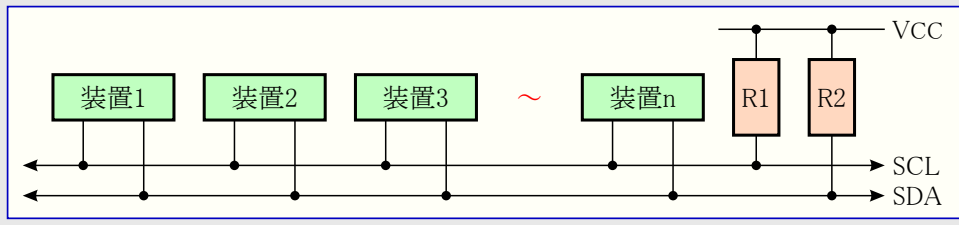
2線インターフェース(TWI)は2線だけを使う双方向バス通信です。TWIはI<sup>2</sup>Cと条件付でSMBusに適合します(101頁の「SMBusとの適合性」をご覧ください)。

バスに接続されたデバイスは主装置または従装置として動作しなければなりません。主装置はバス上の従装置をアドレス指定することによってデータ転送処理を始め、データの送信または受信のどちらを望むかを知らせます。1つのバスは多数の主装置を持って、そして同時に2つ以上の主装置が送信を試みる場合の優先権を調停手順が取り扱います。

### 15.3. バス定義

2線直列インターフェース(TWI)は代表的なマイクロコントローラ応用に対して理想的に適応されています。TWI通信規約は2本の双方向バス信号線、データ用1本(SDA)とクロック用1本(SCL)だけを使って128個までの異なる装置の相互接続をシステム設計者に許します。バスを実現するために必要とされる外部ハードウェアはTWIバス信号線各々に1本ずつのプルアップ抵抗だけです。バスに接続した全ての装置は個別のアドレスを持ち、バス衝突を解決する機構は本質的にTWI通信規約で行います。

図15-1. 2線直列(TWI)バス構成



#### 15.3.1. TWI用語定義

次の定義は本章で度々使われます。

表15-1. TWI用語定義

用語	意味
主装置	送信の開始と終了する装置。主装置はSCLクロックも生成します。
従装置	主装置によって指定された装置。
送信装置	バス上にデータを送り出す装置。
受信装置	バスからデータを読み込む装置。

#### 15.3.2. 電氣的な相互接続

図15-1.で描かれたように両バス信号線はプルアップ抵抗を通して正供給電圧に接続されます。全てのTWI準拠装置のバス駆動部はオープンドレインかオープンコレクタです。これはインターフェースの動作のために重要なワイアードAND機能を実現します。TWIバス信号線のLowレベルは1つまたはより多くのTWI装置の0出力時に生成されます。Highレベルは全TWI装置がHi-Z出力時の出力で、プルアップ抵抗で信号線をHighへ引き上げさせます。TWIバスに接続した全てのAVRデバイスはどんなバス動作も許すために電源が供給されなければならないことに注意してください。

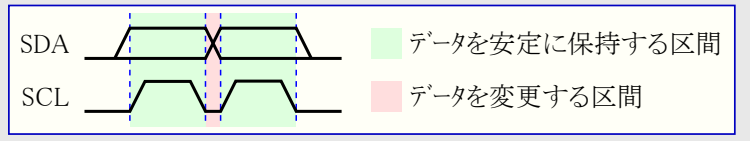
このバスに接続できる装置数は7ビットの従装置アドレス空間と400pFのバス容量制限によってのみ制限されます。TWIの電氣的特性の詳細仕様は139頁の「2線直列インターフェース特性」で与えられます。そこで与えられる2組の異なる仕様は1つがバス速度100kHz以下に関するもので、もう1つはバス速度400kHzまでに関して有効です。

## 15.4. データ転送とフレーム形式

### 15.4.1. ビット転送

TWIバスに転送される各データビットはクロック信号線のパルスを伴います。データ信号線のレベルはクロック信号線がHighの時に安定していなければなりません。この規則の例外は**開始条件**と**停止条件**の生成だけです。

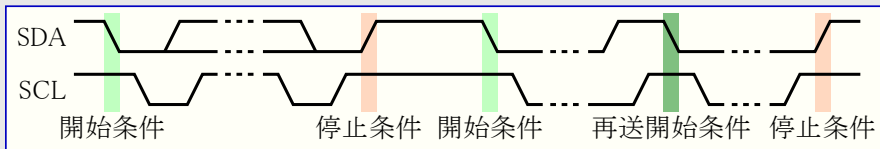
図15-2. データの有効性



### 15.4.2. 開始条件と停止条件

主装置がデータ転送の開始と終了を行います。転送は主装置がバスに**開始条件**を起こすと開始され、主装置が**停止条件**を起こすと終了されます。**開始条件**と**停止条件**間はバスが使用中と考えられ、他の主装置はバスの制御獲得を試みるべきではありません。**開始条件**と**停止条件**間で新規**開始条件**が起こされると特別な状態が起きます。これは**再送開始条件**として引用され、主装置がバスの制御を手放さずに新規転送を始めたい時に使われます。**再送開始条件**後、バスは次の**停止条件**まで使用中と考えられます。これは開始動作に関して全く同じで、従って特記事項を除いて本データシートの残りに対して**開始条件**と**再送開始条件**の両方の記述に**開始条件**が使われます。下で描かれるように**開始条件**と**停止条件**はSCL信号線がHighの時のSDA信号線のレベル変更によって指示されます。

図15-3. 開始条件、再送開始条件、停止条件 バス タイミング



### 15.4.3. アドレス パケット形式

TWIバスに送信した全てのアドレス パケットは7ビットのアドレスビット、1ビットの方向(Read/Write)制御ビット、1ビットの応答ビットから成る9ビットです。方向(R/W)ビットが設定(1)されると読み出し操作が実行され、さもなければ書き込み操作が実行されるべきです。従装置がアドレス指定されたことを認証すると、9番目のSCL(ACK)周期でSDAをLowへ引くことによって確認応答すべきです。アドレス指定された従装置が忙しいまたはその他の理由で主装置の要求を扱えない場合、**確認応答(ACK)**クロック周期でSDA信号線をHighのままにすべきです。主装置はその後に**停止条件**または新規転送を始めるために**再送開始条件**を送出できます。従装置アドレスと方向(R/W)ビットから成るアドレス パケットは各々SLA+RまたはSLA+Wと呼ばれます。

アドレスビットの最上位ビット(MSB)が最初に送信されます。従装置アドレスは設計者によって自由に割り当てられますが、アドレス**0000 000**は一斉呼び出し用に予約されています。

一斉呼び出しが起こされると、全従装置は**確認応答(ACK)**周期でSDA信号線をLowにすることによって応答すべきです。一斉呼び出しは主装置がシステム内のそれぞれの従装置に同じ通信内容を送信したい時に使われます。一斉呼び出しアドレスに続き**W**(方向が書き込み)ビットがバスに送信されると、一斉呼び出しに**応答**する設定の全ての従装置は**ACK**周期でSDA信号線をLowに引き込みます。そして後続のデータ パケットは一斉呼び出しに**確認応答**した全従装置によって受信されます。一斉呼び出しアドレスに続き**R**(方向が読み出し)ビットの送信は、従装置それぞれが異なるデータの送信を始める場合の衝突の原因となるので意味がないことに注意してください。

**1111 xxx**形式の全アドレスは将来の目的のために予約されるべきです(訳補: I2C規格のアドレス拡張他)。

### 15.4.4. データ パケット形式

TWIバスに送信した全てのデータ パケットは1バイトのデータと1ビットの応答ビットから成る9ビットです。データ転送中、主装置はクロックと**開始条件**、**停止条件**を生成し、一方受信装置は受信に**応答**する責任があります。**確認応答(ACK)**は受信装置が9番目のSCL周期中にSDA信号線をLowに引き込むことによって示されます。受信装置がSDA信号線をHighのままにすると**NACK**を示します。受信装置が最終バイトを受信したとき、または何らかの理由でこれ以上のバイトを受信できないとき、最終バイト後に**NACK**を送ることによって送信装置へ通知すべきです。データバイトの最上位(MSB)ビットが最初に送信されます。

図15-4. アドレス パケット形式

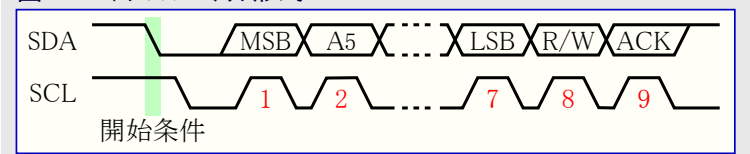
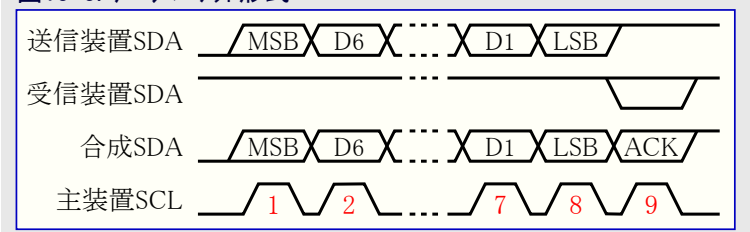


図15-5. データ パケット形式

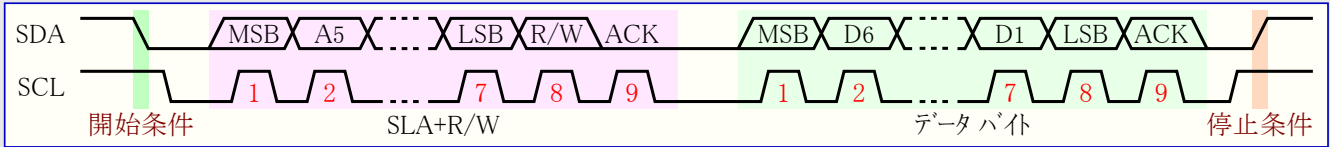


### 15.4.5. 転送内でのアドレスパケットとデータパケットの組み合わせ

転送は基本的に開始条件、SLA+R/W、1つ以上のデータパケット、停止条件から成ります。開始条件に続く停止条件から成る空の通信内容は規則違反です。SCL信号線のワイアードANDが主装置と従装置間のハンドシェイクに使えることに注目してください。従装置はSCL信号線をLowに引き込むことによってSCLのLow期間を引き伸ばせます。これは主装置が従装置に対して速すぎるクロック速度設定、または従装置がデータ送信間の処理に追加時間を必要とする場合に有用です。従装置がSCLのLow期間を延長することは主装置によって決められるSCLのHigh期間に影響しません。同様に従装置はSCLのデューティ比(Low期間)を延長することによってTWIデータ転送速度を落とせます。

図15-6.は代表的なデータ転送を示します。様々なデータは応用ソフトウェアによって実装されたソフトウェア規約に依存し、SLA+R/Wと停止条件間に送信できることに注意してください。

図15-6. 代表的なデータ転送



### 15.5. 複数主装置バスシステムの調停と同期

TWI規約は多数主装置のバスシステムを許します。例え2つ以上の主装置が同時に送信を始めても、送信が通常のように続行することを保証するために特別な手段が講じられます。複数主装置のシステムでは2つの問題が起こります。

- 送信を完了するために1つの主装置だけを許す方法が実現されなければなりません。他の全ての主装置は(自身が行っている従装置)選択手順を失った(失敗した)ことに気付く時、送信を止めるべきです。この選択手順は調停(アビトレーション)と呼ばれます。競合する主装置は調停(従装置選択)手順を失ったことに気付くと、勝ち残った主装置によってアドレス指定されるかどうかを調べるため、直ちに従装置動作へ切り替えるべきです。複数の主装置が同時に送信を始めた事実は従装置で検知できるべきではありません。換言すると、バスに転送されているデータが不正にされてはなりません。
- 違う主装置が異なるSCL周波数を使うかもしれません。同期確定手順で送信が続行するために、全主装置からの直列クロックを同期化する方法が考案されなければなりません。これは調停手順を容易にします。

バス信号線のワイアードANDはこれらの問題の両方の解決に用いられます。全ての主装置からの直列クロックはワイアードANDされ、最短High期間の主装置の1つからに等しいHigh期間の合成クロックを生成します。合成クロックのLow期間は最長Low期間の主装置のLow期間に等しくなります。全ての主装置がSCL信号線を監視する、実際には合成SCL信号線がHighまたはLowになるとき、各々SCLのHighとLow経過時間の計時を始めることに注意してください。

調停は全ての主装置がデータ出力後にSDA信号線を継続的に監視することによって実行されます。SDA信号線から読んだ値がその主装置の出力した値と一致しない場合、調停に敗れます。主装置がSDAにHigh値を出力し、同時に他の主装置がLow値を出力する時のみ調停に敗れるかもしれないことに注意してください。敗れた主装置は直ちに従装置動作へ移行し、勝ち残った主装置によってアドレス指定されるかを検査すべきです。SDA信号線はHighのままにすべきですが、敗れた主装置は現在のデータ若しくはアドレスパケットの最後までクロック信号を生成することを許されます。調停は唯一の主装置が残るまで継続され、多くのビットを必要とするかもしれません。多くの主装置が同じ従装置をアドレス指定しようとすると、調停はデータパケットへ続くでしょう。

図15-7. 複数主装置間でのSCL同期化

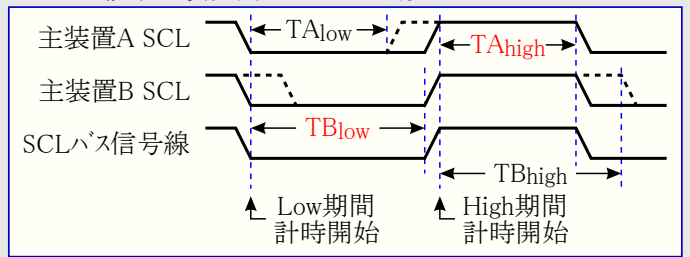
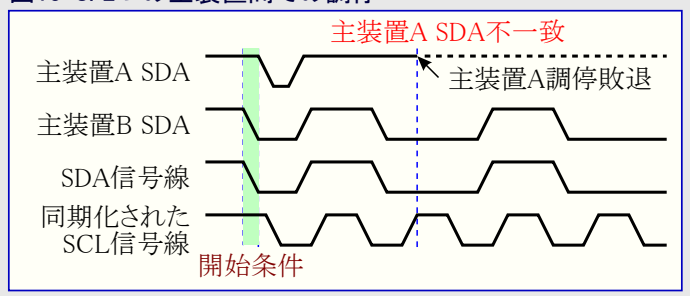


図15-8. 2つの主装置間での調停



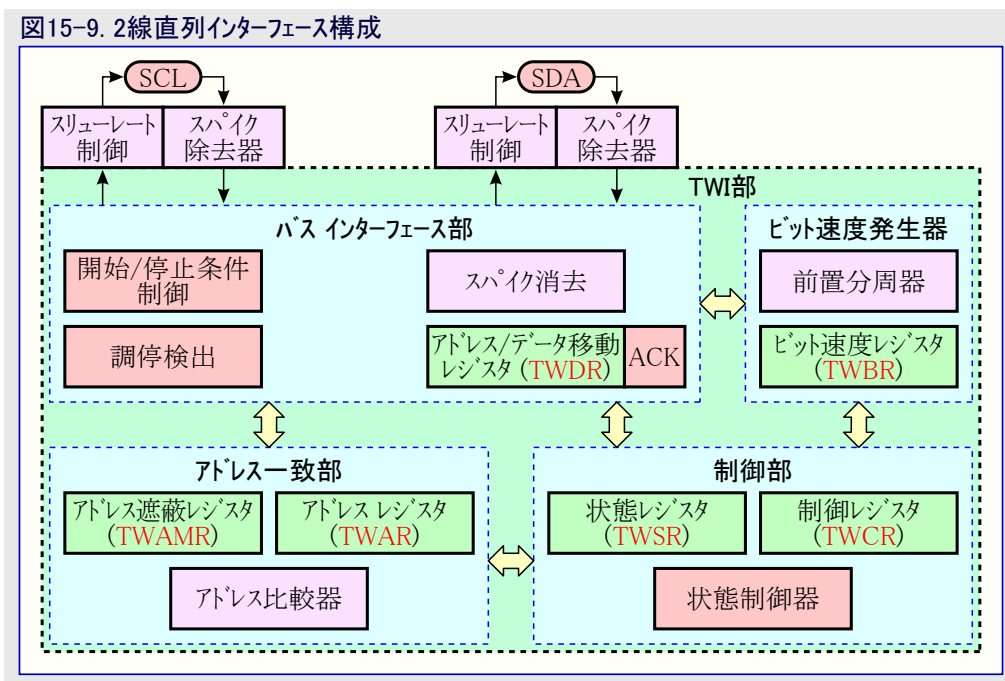
調停が次の状態間で許されないことに注意してください。

- 再送開始条件とデータビット間
- 停止条件とデータビット間
- 再送開始条件と停止条件間

これらの違法な調停状態を決して起こさないよう保証するのは使用者ソフトウェアの責任です。これは複数主装置システムでの全てのデータ転送は同じ構成、SLA+R/Wとデータパケットを使われなければならないことを意味します。言葉を変えると、全ての送信は同じデータパケット数を含まなければならない、さもなければ調停の結果は不定にされます。

## 15.6. TWI部の概要

図15-9.で示されるようにTWI部は様々な部分から成ります。赤文字で示された(訳注:原文は太線で描かれた)全てのレジスタはAVRデータバスを通してアクセス可能です。



### 15.6.1. SCLとSDAピン

これらのピンはAVR TWIをMCUシステムのその他とインターフェースします。出力駆動部はTWI仕様に適合させるためのスレーブ(上昇/下降)制限器を含みます。入力段は50nsよりも短いスパイクを除去するスパイク除去部を含みます。「入出力ポート」章で説明したようにAVRパッドの内部プルアップはSCLとSDAピンに対応するポートのビットを設定(1)することによって許可できることに注目してください。内部プルアップはいくつかのシステムで外部抵抗の必要をなくせます。

### 15.6.2. ビット速度発生器

この部分は主装置動作で動く時のSCL周期を制御します。SCL周期はTWIビット速度レジスタ(TWBR)とTWI状態レジスタ(TWSR)の前置分周器ビットの設定によって制御されます。従装置動作はビット速度や前置分周器設定と関係ありませんが、従装置でのCPUクロック周波数はSCL周波数よりも最低16倍高くなければなりません。従装置がSCLのLow期間を延長するかもしれず、これによって平均TWIバスクロック周波数が減少することに注意してください。

TWIは104頁の「TWIHSR - TWI高速レジスタ」で記述されるように、高速動作で動くように設定することができます。高速動作ではTWIがシステムクロックを使い、一方標準動作では同じシステムクロックの前置分周版に頼ります。使うクロック信号に依存してSCL周波数は次式の1つに従って生成されます。

標準動作SCL周波数 = $\frac{\text{clkI/O}}{16 + 2 \times (\text{TWBR}) \times \text{前置分周値}}$	clkI/O : 前置分周されたシステムクロック(17頁の図6-1.参照)
	clkTWIHS : システムクロック(17頁の図6-1.参照)
高速動作SCL周波数 = $\frac{\text{clkTWIHS}}{16 + 2 \times (\text{TWBR}) \times \text{前置分周値}}$	TWBR : TWIビット速度レジスタ値
	前置分周値 : TWI状態レジスタ内TWPSで指定(103頁の表15-7.参照)

注: TWI主装置動作ではTWBRが10またはそれ以上でなければなりません。

### 15.6.3. バスインターフェース部

この部分はデータとアドレスの移動レジスタ(TWDR)、開始条件/停止条件制御器、調停検出回路を含みます。TWDRは送信されるべきアドレスまたはデータバイト、若しくは受信したアドレスまたはデータバイトを含みます。8ビットのTWDRに加えバスインターフェース部は送信されるべきまたは受信した(N)ACKビットを含むレジスタも含みます。この(N)ACKレジスタは応用ソフトウェアによって直接的にアクセスできません。けれどもTWI制御レジスタ(TWCR)を操作することにより、受信時に設定(1)または解除(0)できます。送信装置動作時、受信した(N)ACKビットの値はTWSRの値によって判定できます。

開始条件/停止条件制御器は開始条件、再送開始条件、停止条件の生成と検出に対して責任があります。開始条件/停止条件制御器はAVR MCUが主装置によってアドレス指定されるとMCUを起動できる休止形態の1つの時でも、開始条件または停止条件を検出できます。

TWIが主装置として送信を始められると、調停検出ハードウェアは調停が進行中かを定めるために送信の試行を継続的に監視します。TWIが調停に敗れた場合、制御部に通知されます。その後正しい処置が行われ、適切な状態符号が生成されます。

#### 15.6.4. アドレス一致部

アドレス一致部は受信したアドレスバイトがTWIアドレスレジスタ(TWAR)の7ビットアドレスと一致するかを検査します。TWARで一斉呼び出し検出許可(TWGCE)ビットが1を書かれると、全ての到着アドレスビットは一斉呼び出しアドレスに対しても比較されます。アドレス一致を制御部は通知され、正しい処置を行うことを許します。TWIはTWI制御レジスタ(TWCR)の設定によってそのアドレスへの応答をするかもしれないし、しないかもしれません。アドレス一致部はAVR MCUが主装置によってアドレス指定されるとMCUを起動できる休止形態の1つの時でもアドレスを比較できます。TWIがパワーダウン動作でのアドレス一致でCPUを起動中に他の割り込み(例えばINT0)が起これば、TWIは動作を停止してアイドル状態で復帰します。これが何らかの問題の原因なら、パワーダウン動作へ移行する時にTWIアドレス一致だけが割り込みを許可されることを保証してください。

#### 15.6.5. 制御部

制御部はTWIバスを監視し、TWI制御レジスタ(TWCR)の設定に従った応答を生成します。応用に注意を要求する事象がTWIバスで起これば、TWI割り込み要求フラグ(TWINT)が有効にされます。次のクロック周期でTWI状態レジスタ(TWSR)は事象を示す状態符号で更新されます。TWI割り込み要求フラグが有効にされる時にだけ、TWSRは適切な状態情報を含みます。他の全ての時でTWSRは適切な状態情報が利用できないことを示す特別な状態符号を含みます。TWINTフラグが設定(1)されている限り、SCL信号線はLowに保たれます。これは続くTWI送信を許す前に(現状)処理完了を応用ソフトウェアに許します。

TWI割り込み要求フラグ(TWINT)は次の場合に設定(1)されます。

- 開始条件または再送開始条件送信後
- SLA+R/W送信後
- アドレスバイト送信後
- 調停に敗れた後
- 自身の従装置アドレスまたは一斉呼び出しによってアドレス指定された後
- データバイト受信後
- 従装置として未だアドレス指定されている間の停止条件または再送開始条件受信後
- 不正な開始条件または停止条件のためバス異常が起きた時

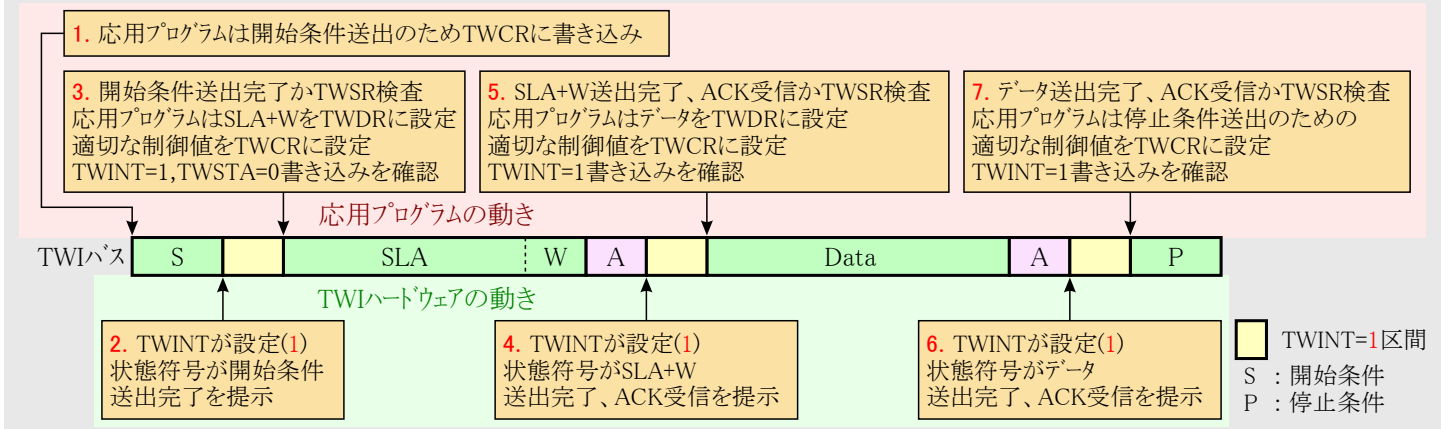
### 15.7. TWIの使用法

AVR TWIはパルソ志向で割り込みが基本です。割り込みはパルソの受信や開始条件の送出のような全てのパルソの事象後に起こります。TWIは割り込みが基本のため、アプリケーションソフトウェアはTWIパルソ転送中に他の操作を続行するために開放されます。ステータスレジスタ(SREG)の全割り込み許可(0)ビットと共にTWI制御レジスタ(TWCR)のTWI割り込み許可(TWIE)ビットは、TWCRのTWI割り込み要求フラグ(TWINT)の設定(1)が割り込み要求を発生すべきかどうか決めることをアプリケーションソフトウェアに許します。TWIEビットが解除(0)されると、アプリケーションソフトウェアはTWIパルソの動きを検知するためにTWINTフラグをポーリングしなければなりません。

TWINTフラグが設定(1)されると、TWIは動作を終え、アプリケーションソフトウェアの応答を待ちます。この場合、TWI状態レジスタ(TWSR)はTWIパルソの現在の状態を示す値を含みます。そしてアプリケーションソフトウェアはTWCRとTWDRの操作により、TWIが次のTWIパルソ周期で何を行うべきかを決定できます。

図15-10はアプリケーションソフトウェアがTWIハードウェアにどうインターフェースできるかの簡単な例です。この例では主装置が単一データパルソを従装置に送信しようとしています。この内容はかなり大雑把ですので、より詳細な説明が本項の後に続きます。希望した動きを実現する簡単なコード例も示されます。

図15-10. 代表的な送信でのアプリケーションプログラムとTWIのインターフェース



1. TWI送信の最初の段階は**開始条件**を送出することです。これはTWIハードウェアに開始条件送出を命じる特別な値をTWCR内に書くことによって行います。どんな値を書くかは後で記述されます。けれども、書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みは、このフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。アプリケーションソフトウェアがTWINTを解除(0)した後、TWIは直ちに**開始条件**の送出を始めます。
2. 開始条件が送出されてしまうと、TWCRでTWINTフラグが設定(1)され、TWSRは**開始条件**が正常に送出されてしまったことを示す状態符号に更新されます。
3. アプリケーションソフトウェアは**開始条件**が正常に送信されたのを確認するためにTWSRの値を直ぐに検査すべきです。TWSRが他を示している場合、アプリケーションソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、アプリケーションソフトウェアはTWDRに**SLA+W**を設定しなければなりません。TWDRがアドレスとデータの両方に使われることを思い出してください。TWDRが希望した**SLA+W**に設定されてしまった後、TWDRにある**SLA+W**の送信をTWIハードウェアへ命じる特別な値がTWCRに書かれなければなりません。どんな値を書くかは後で記述されます。けれども書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みがこのフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。アプリケーションソフトウェアがTWINTを解除(0)した後、TWIは直ちにアドレスパケットの送信を始めます。
4. アドレスパケットが送信されてしまうと、TWCRでTWINTフラグが設定(1)され、TWSRはアドレスパケットが正常に送信されたことを示す状態符号に更新されます。この状態符号は従装置がパケットに**応答**したかどうかも反映します。
5. アプリケーションソフトウェアはアドレスパケットが正常に送信され、期待された**ACK**ビット値であることを確認するためにTWSRの値を直ぐに検査すべきです。TWSRが他を示している場合、アプリケーションソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、アプリケーションソフトウェアはTWDRにデータを設定しなければなりません。その後、TWDRにあるデータパケットの送信をTWIハードウェアへ命じる特別な値がTWCRに書かれなければなりません。どんな値を書くかは後で記述されます。けれども書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みがこのフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。アプリケーションソフトウェアがTWINTを解除(0)した後、TWIは直ちにデータパケットの送信を始めます。
6. データパケットが送信されてしまうと、TWCR内のTWINTフラグが設定(1)され、TWSRはデータパケットが正常に送信されたことを示す状態符号に更新されます。この状態符号は従装置がパケットに**応答**したかどうかも反映します。
7. アプリケーションソフトウェアはデータパケットが正常に送信され、期待された**ACK**ビットの値であることを確認するためにTWSRの値を直ぐに検査すべきです。TWSRが他を示している場合、アプリケーションソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、アプリケーションソフトウェアは**停止条件**の送出をTWIハードウェアへ命じる特別な値をTWCRに書かなければなりません。どんな値を書くかは後で記述されます。けれども書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みがこのフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。アプリケーションソフトウェアがTWINTを解除(0)した後、TWIは直ちに**停止条件**の送出を始めます。**停止条件**が送出されてしまった後にTWINTが設定(1)されないことに注意してください。

この例は簡単とはいえ、全てのTWI送信に関係した原理を示しています。これらは次のように要約できます。

- TWIが動作を終了して応用(ソフトウェア)の反応を予想する時に**TWINTフラグ**が設定(1)されます。SCL信号線はTWINTが解除(0)されるまで**Low**に引き込まれます。
- TWINTフラグが設定(1)されたなら、使用者は次のTWIバス周期に関連した値で(必要な)全てのTWIレジスタを更新しなければなりません。例で示されるように**TWDR**は次のTWIバス周期で送信されるべき値を設定されなければなりません。
- (必要な)全てのTWIレジスタを更新し、その他保留中の応用ソフトウェアの処理が完了してしまった後に**TWCR**が書かれます。TWCR書き込み時、TWINTビットが設定(1)されるべきです。TWINTへの1書き込みはこのフラグを解除(0)します。TWCR設定によってどの動作が指定されても、TWIはその(TWINT=0)後に実行を始めます。

次にアセンブリ言語とC言語の実装例が与えられます。以下のコードは例えばインクルードファイルの使用により、様々な定義が作成されてしまっている前提であることに注意してください。

	アセンブリ言語プログラム例	C言語プログラム例	注釈
1.	LDI R16, (1<<TWINT)   (1<<TWSTA)   (1<<TWEN) OUT TWCR, R16	TWCR = (1<<TWINT)   (1<<TWSTA)   (1<<TWEN);	; 開始条件送出
2.	WAIT1: IN R16, TWCR SBRS R16, TWINT RJMP WAIT1	while (!(TWCR & (1<<TWINT)));	; TWINT=1まで待機 ; (開始条件送出完了待機)
3.	IN R16, TWSR ANDI R16, \$F8 CPI R16, START BRNE ERROR	if ((TWSR & 0xF8) != START) ERROR();	; TWI状態レジスタ値検査 ; 前置分周選択ビットの遮蔽 ; STARTと異なる状態符号で ; 異常処理へ
	LDI R16, SLA_W OUT TWDR, R16 LDI R16, (1<<TWINT)   (1<<TWEN) OUT TWCR, R16	TWDR = SLA_W; TWCR = (1<<TWINT)   (1<<TWEN);	; TWDRにSLA+W設定 ; アドレス送信開始のため ; TWCRのTWINTを解除(0)
4.	WAIT2: IN R16, TWCR SBRS R16, TWINT RJMP WAIT2	while (!(TWCR & (1<<TWINT)));	; TWINT=1まで待機 ; (SLA+W送出完了と ; ACK/NACK受信完了待機)
5.	IN R16, TWSR ANDI R16, \$F8 CPI R16, MT_SLA_ACK BRNE ERROR	if ((TWSR & 0xF8) != MT_SLA_ACK) ERROR();	; TWI状態レジスタ値検査 ; 前置分周選択ビットの遮蔽 ; MT_SLA_ACKと違う状態符号で ; 異常処理へ
	LDI R16, DATA OUT TWDR, R16 LDI R16, (1<<TWINT)   (1<<TWEN) OUT TWCR, R16	TWDR = DATA; TWCR = (1<<TWINT)   (1<<TWEN);	; TWDRにデータ設定 ; データ送信開始のため ; TWCRのTWINTを解除(0)
6.	WAIT3: IN R16, TWCR SBRS R16, TWINT RJMP WAIT3	while (!(TWCR & (1<<TWINT)));	; TWINT=1まで待機 ; (データ送出完了と ; ACK/NACK受信完了待機)
7.	IN R16, TWSR ANDI R16, \$F8 CPI R16, MT_DATA_ACK BRNE ERROR	if ((TWSR & 0xF8) != MT_DATA_ACK) ERROR();	; TWI状態レジスタ値検査 ; 前置分周選択ビットの遮蔽 ; MT_DATA_ACKと違う状態符号で ; 異常処理へ
	LDI R16, (1<<TWINT)   (1<<TWSTO)   (1<<TWEN) OUT TWCR, R16	TWCR = (1<<TWINT)   (1<<TWSTO)   (1<<TWEN);	; 停止条件送出

注: 5頁の「コード例について」をご覧ください。

## 15.8. 転送種別

TWIは4つの主な動作種別の1つで動けます。これらは送信主装置(MT)、受信主装置(MR)、送信従装置(ST)、受信従装置(SR)と名付けられます。これら種別の多くは同じ応用に使えます。例えば、TWI方式のEEPROM内にデータを書くのにTWIはMT動作を、EEPROMからデータを読み戻すのにMR動作を使えます。システム内に他の主装置が存在する場合、それらのいくつかがTWIにデータを送信するかもしれず、するとSR動作が使われるでしょう。どの動作種別が適正かを決めるのは応用ソフトウェアです。

次項はこれら動作種別の各々を記述します。起こり得る状態符号は各動作種別のデータ伝送詳細図に沿って示されます。これらの図は次の略号を含みます。

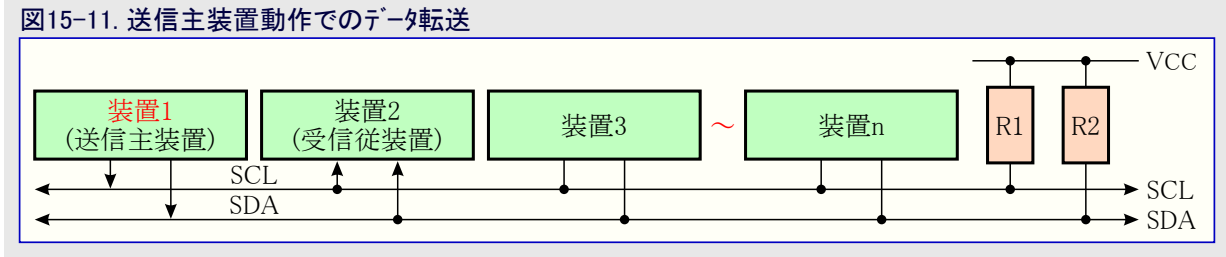
S	開始(START)条件
Rs	再送開始(REPEATED START)条件
R	読み出し指定ビット (SDA=High)
W	書き込み指定ビット (SDA=Low)
A	確認応答(ACK)ビット (SDA=Low)
$\bar{A}$	非確認応答(NACK)ビット (SDA=High)
Data	8ビット データ バイト
P	停止(STOP)条件
SLA	従装置アドレス

**図15-12. ~18.**内の楕円(訳注:原文は円)はTWI制御レジスタ(TWCR)のTWI割り込み要求フラグ(TWINT)が設定(1)されたことを示すのに使われます。この楕円内の番号は前置分周選択ビットが0で遮蔽されたTWI状態レジスタ(TWSR)に保持した状態符号を表します。これら位置での動きはTWI転送の継続または完了が応用(ソフトウェア)によって行われなければなりません。TWI転送はソフトウェアによってTWINTフラグが解除(0)されるまで一時停止されます。

TWI割り込み要求フラグ(TWINT)が設定(1)される時のTWI状態レジスタ(TWSR)の状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対する必要なソフトウェア動作や後続の直列転送の詳細は表15-2. ~5.で与えられます。これらの表に於いて前置分周選択ビットが0で遮蔽されていることに注意してください。

### 15.8.1. 送信主装置動作

送信主装置動作では何バイトかのデータが受信従装置へ送信されます(図15-11.参照)。主装置動作へ移行するには**開始条件**が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。**SLA+W**が送信されると送信主装置(MT)へ移行し、**SLA+R**が送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は**前置分周選択ビット**が0か、または0で遮蔽されることが前提です。



**開始条件**はTWCRに次の値を書くことによって送出されます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。TWSTAは**開始条件**を送出するために1を書かれねばならず、TWINTはTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに**開始条件**を生成します。**開始条件**が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、TWSRの状態符号が\$08(表15-2.参照)になります。送信主装置へ移行するには**SLA+W**が送信されなければなりません。これはTWDRに**SLA+W**を書くことによって行います。その後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

**SLA+W**が送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$18,\$20,\$38です。これら状態符号の各々に対する適切な動作は表15-2.で詳述されます。

**SLA+W**が正常に送信されてしまうと、データパケットが送信されるべきです。これはTWDRにデータバイトを書くことによって行われます。TWDRはTWINTが1の時にだけ書かれなければなりません。さもなければ、そのアクセスは破棄され、TWCRで**上書き発生(TWWC)フラグ**が設定(1)されます。TWDR更新後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

最後のバイトが送られてしまうまでこの手順が繰り返され、この転送は**停止条件**または**再送開始条件**を生成することによって終了されます。**停止条件**はTWCRに次の値を書くことによって生成されます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	1	X	1	0	X

**再送開始条件**はTWCRに次の値を書くことによって生成されます。

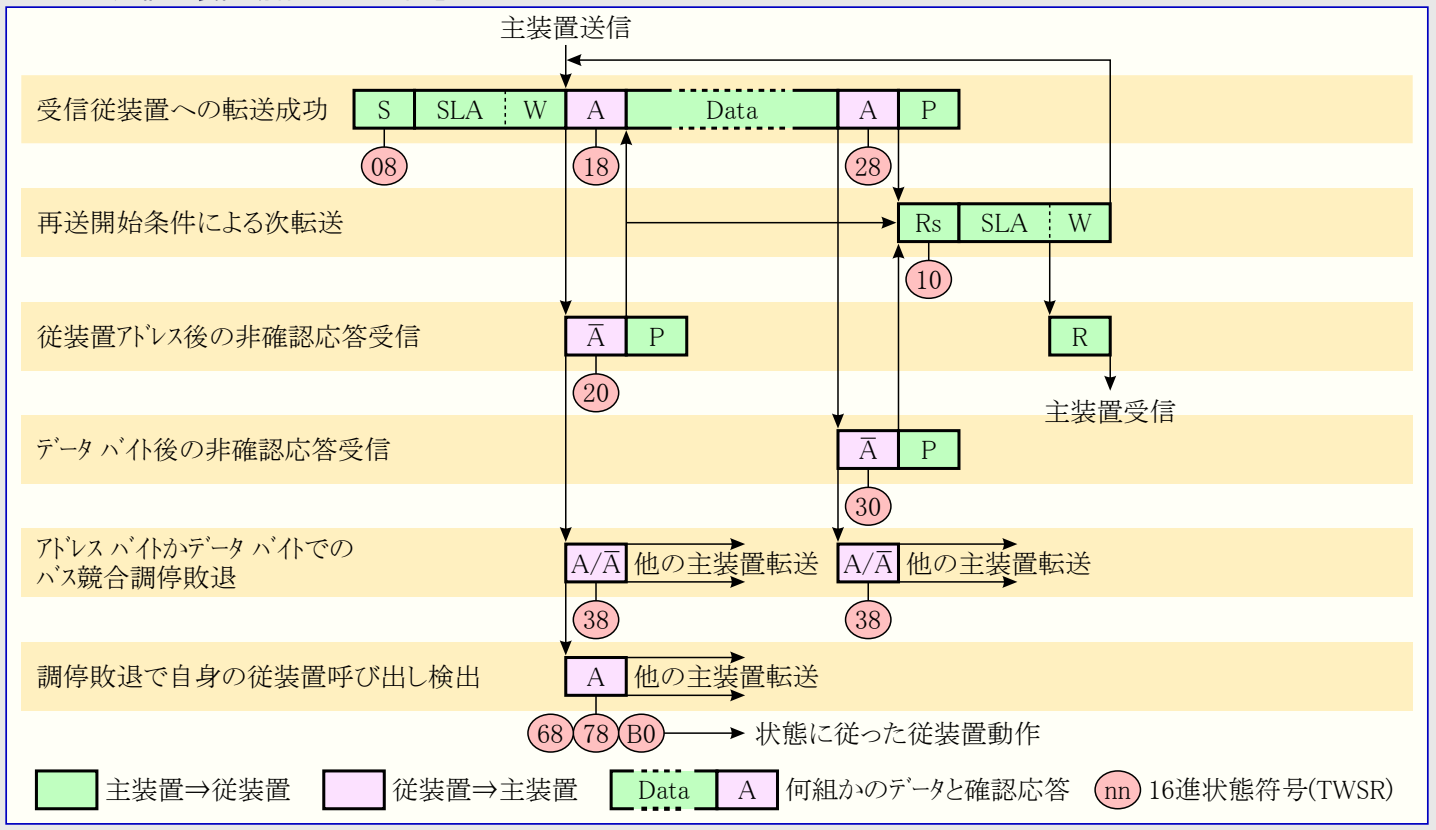
ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

**再送開始条件**(状態符号\$10)後、2線直列インターフェースは**停止条件**を送出せずに再び同じ従装置または新しい従装置にアクセスできます。**再送開始条件**は主装置がバスの制御を失わずに送信主装置、受信主装置間の切り替えを可能にします(訳注:原文では従装置も含まれていますが、基本動作に対して不適切なため削除しました)。

表15-2. 送信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

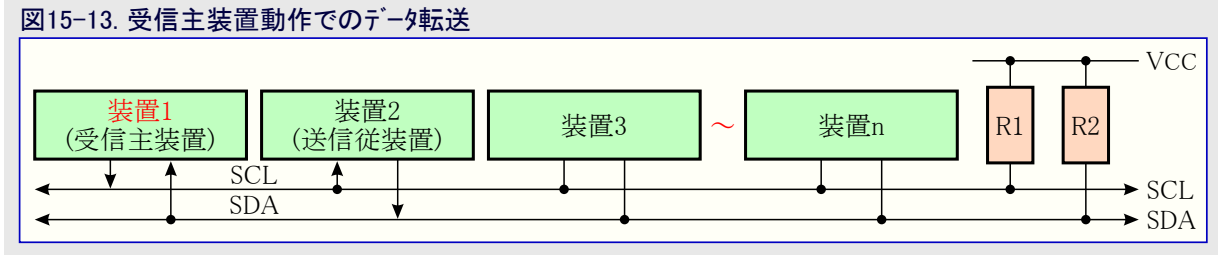
状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの対応				TWCR設定によるハードウェア動作	
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT		TWEA
\$08	開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
		SLA+R設定	0	0	1	X	SLA+R送信、受信主装置動作へ移行
\$18	SLA+W送信 ACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
\$20	SLA+W送信 NACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
\$28	データバイト送信 ACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
\$30	データバイト送信 NACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
\$38	SLA+W, データバイトで バス競合調停敗退	なし	0	0	1	X	バス開放、未指定従装置動作へ移行
			1	0	1	X	バス開放時に開始条件送信

図15-12. 送信主装置動作の形式と状態



## 15.8.2. 受信主装置動作

受信主装置動作では何バイトかのデータが送信従装置から受信されます(図15-13参照)。主装置動作へ移行するには**開始条件**が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。**SLA+W**が送信されると送信主装置(MT)へ移行し、**SLA+R**が送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は**前置分周選択ビット**が0か、または0で遮蔽されることが前提です。



**開始条件**はTWCRに次の値を書くことによって送出されます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。TWSTAは**開始条件**を送出するために1を書かれねばならず、TWINTはTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに**開始条件**を生成します。**開始条件**が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、TWSRの状態符号が\$08(表15-3参照)になります。受信主装置へ移行するには**SLA+R**が送信されなければなりません。これはTWDRに**SLA+R**を書くことによって行います。その後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

**SLA+R**が送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$38,\$40,\$48です。これら状態符号の各々に対する適切な動作は表15-3で詳述されます。

ハードウェアによってTWINTフラグが設定(1)されると、受信したデータがTWDRから読めます。この手順は最後のバイトが受信されてしまうまで繰り返されます。最後のバイトが受信されてしまった後、受信主装置は最後に受信したデータバイト後の**NACK**送信によって送信従装置へ通知すべきです。この転送は**停止条件**または**再送開始条件**を生成することによって終了されます。**停止条件**はTWCRに次の値を書くことによって生成されます。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	1	X	1	0	X

**再送開始条件**はTWCRに次の値を書くことによって生成されます。

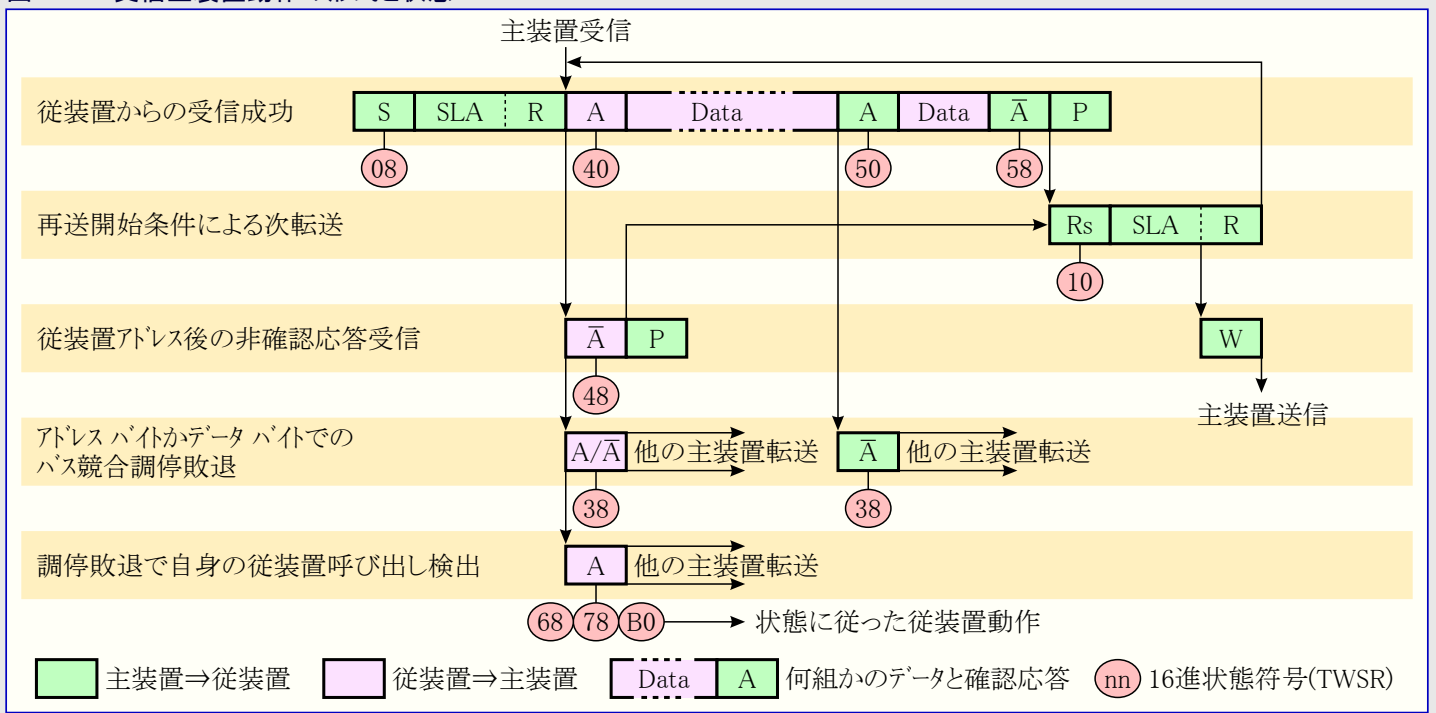
ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

**再送開始条件**(状態符号\$10)後、2線直列インターフェースは**停止条件**を送出せずに再び同じ従装置または新しい従装置にアクセスできます。**再送開始条件**は主装置がバスの制御を失わずに送信主装置、受信主装置間の切り替えを可能にします(訳注:原文では従装置も含まれていますが、基本動作に対して不適切なため削除しました)。

表15-3. 受信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの対応					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$08	開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
		SLA+W設定	0	0	1	X	SLA+W送信、送信主装置動作へ移行
\$38	SLA+Rで調停敗退 またはNACK受信	なし	0	0	1	X	バス開放、未指定従装置動作へ移行
			1	0	1	X	バス開放時に開始条件送信
\$40	SLA+R送信 ACK受信	なし	0	0	1	0	データ受信、NACK応答
			0	0	1	1	データ受信、ACK応答
\$48	SLA+R送信 NACK受信	なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0
\$50	データバイト受信 ACK応答	データ取得	0	0	1	0	データ受信、NACK応答
			0	0	1	1	データ受信、ACK応答
\$58	データバイト受信 NACK応答	データ取得	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0

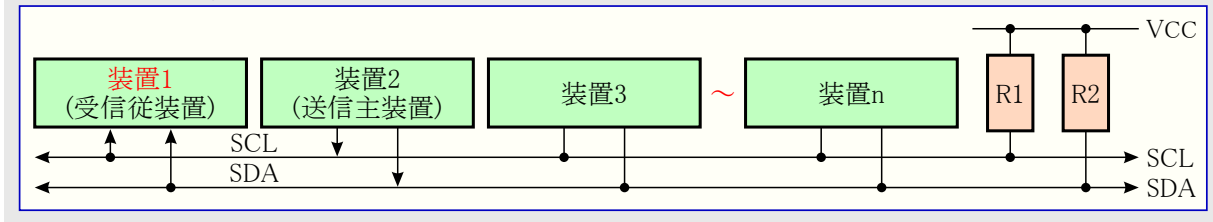
図15-14. 受信主装置動作の形式と状態



### 15.8.3. 受信従装置動作

受信従装置動作では何バイトかのデータが送信主装置から受信されます(図15-15参照)。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。

図15-15. 受信従装置動作でのデータ転送



受信従装置動作を始めるにはTWARとTWCRが次のように初期化されなければなりません。

ビット	7	6	5	4	3	2	1	0
TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
設定値	装置自身の従装置アドレス							1/0

上位7ビットは主装置によってアドレス指定される時に2線直列インターフェースが応答するアドレスです。最下位(TWGCE)ビットが設定(1)されるなら、TWIは一斉呼び出し(\$00)に応答し、さもなければ一斉呼び出しアドレスを無視します。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	0	1	0	0	0	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために1を書かれなければなりません。TWEAは装置自身の従装置アドレスまたは一斉呼び出しアドレスの確認応答(ACK)を許可するために1を書かれなければなりません。TWSTAとTWSTOは0を書かれなければなりません。

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可なら、一斉呼び出しアドレス)とそれに続くデータ方向ビットによってアドレス指定されるまで待ちます。方向ビットが0(W)ならばTWIは受信従装置で動作し、さもなければ1(R)ならば送信従装置へ移行されます。自身の従装置アドレスとWビットが受信されてしまった後にTWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表15-4で詳述されます。受信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$68,\$78参照)

転送中にTWEAビットがリセット(0)されると、TWIは次に受信したデータバイト後のSDAに非確認応答(NACK)(SDA=High)を返します。これは従装置がこれ以上受信できないことを示すのに使えます。TWEAが0の間中、TWIは自身の従装置アドレスに応答しませんが、2線直列バスは未だ監視され、アドレス認証はTWEAの設定(1)によって何時でも再開できます。これはTWEAビットがTWIを2線直列バスから一時的に隔離するのに使えることを意味します。

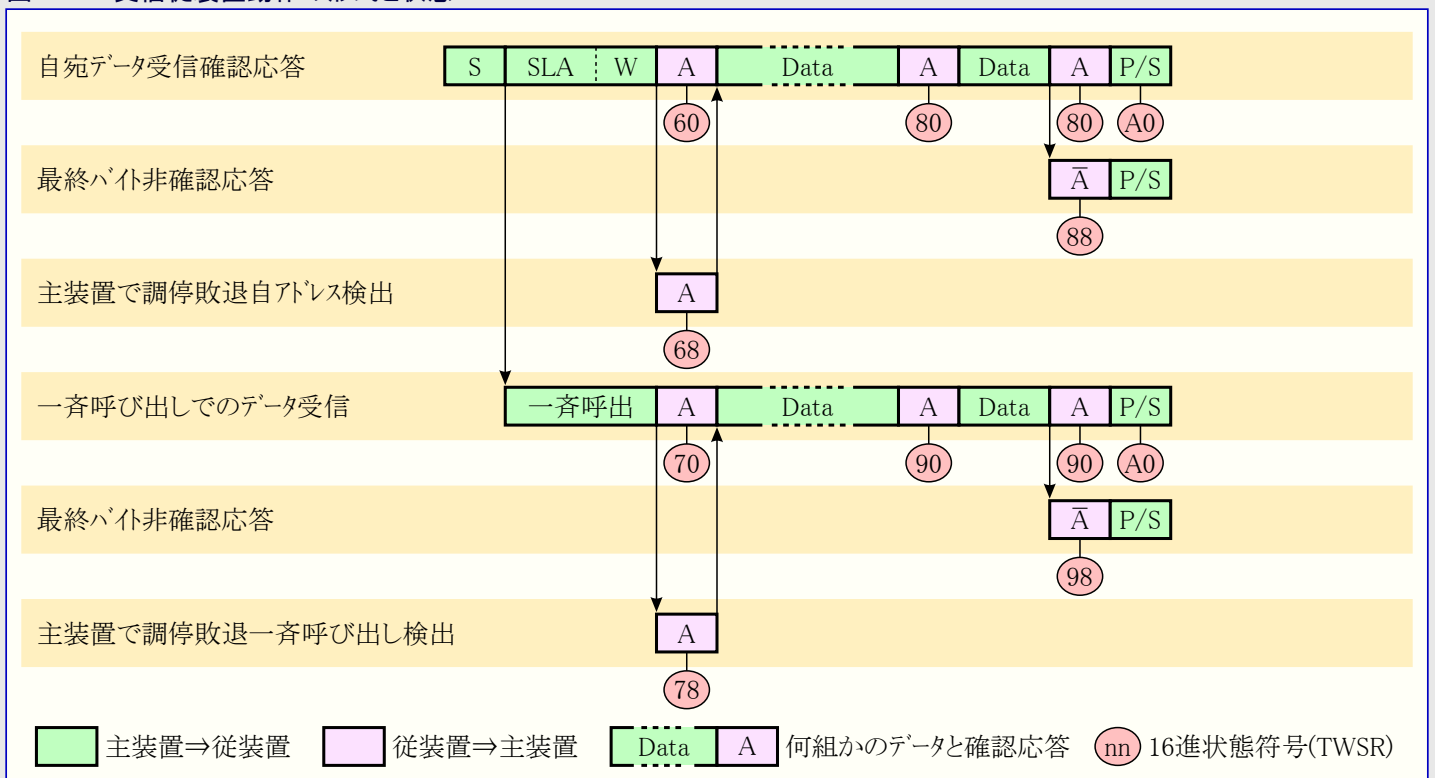
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが1書き込みによって解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないことに気付いてください。

これらの(アイドル動作を除く)休止形態から起動すると、2線直列インターフェースデータレジスタ(TWDR)がバスで渡す最後のバイトを反映しないことに注意してください。

表15-4. 受信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

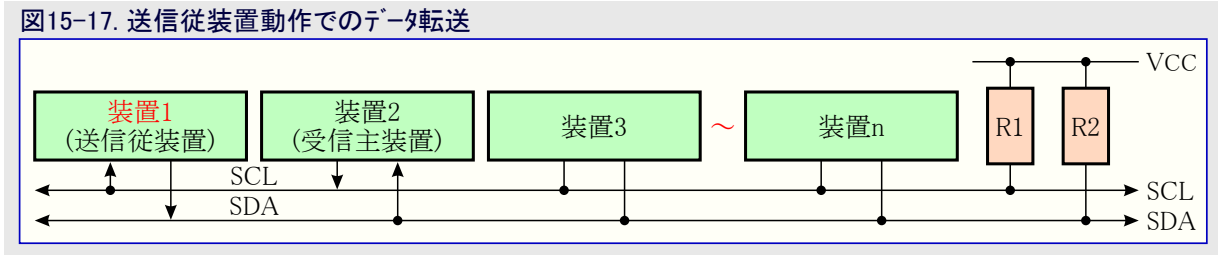
状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの応答				TWCR設定によるハードウェア動作	
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT		TWEA
\$60	自宛SLA+W受信 ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$68	主装置のSLA+R/Wで 調停敗退/自宛SLA+W 受信/ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$70	一斉呼び出し受信 ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$78	主装置のSLA+R/Wで 調停敗退/一斉呼び 出し受信/ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$80	自宛データバイト受信 ACK応答	データ取得	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$88	自宛データバイト受信 NACK応答	データ取得	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$90	一斉呼び出しのデータ バイト受信/ACK応答	データ取得	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$98	一斉呼び出しのデータ バイト受信 NACK応答	データ取得	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$A0	自指定中の 停止条件または 再送開始条件検出	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信

図15-16. 受信従装置動作の形式と状態



### 15.8.4. 送信従装置動作

送信従装置動作では何バイトかのデータが送信主装置へ送信されます(図15-17参照)。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。



送信従装置動作を始めるにはTWARとTWCRが次のように初期化されなければなりません。

ビット	7	6	5	4	3	2	1	0
TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
設定値	装置自身の従装置アドレス							1/0

上位7ビットは主装置によってアドレス指定される時に2線直列インターフェースが応答するアドレスです。最下位(TWGCE)ビットが設定(1)されるなら、TWIは一斉呼び出し(\$00)に応答し、さもなければ一斉呼び出しアドレスを無視します。

ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	0	1	0	0	0	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために1を書かれなければなりません。TWEAは装置自身の従装置アドレスまたは一斉呼び出しアドレスの確認応答(ACK)を許可するために1を書かれなければなりません。TWSTAとTWSTOは0を書かれなければなりません。

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可ならば一斉呼び出しアドレス)とそれに続くデータ方向ビットによってアドレス指定されるまで待機します。方向ビットが1(R)ならばTWIは送信従装置で動作し、さもなければ0(W)ならば受信従装置へ移行されます。自身の従装置アドレスとRビットが受信されてしまった後、TWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表15-5で詳述されます。送信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$B0参照)

転送中にTWEAビットが0を書かれると、TWIは転送の最後のバイトを送信します。受信主装置が最終バイト後にACKまたはNACKのどちらを送信するかによって状態\$C0か\$C8へ移行します。TWIはアドレス指定されていない従装置動作に切り替えられ、主装置が転送を続ける場合、その主装置を無視します。従って受信主装置は直列データとして全て1を受信します。従装置が最後のバイトを送信(TWEAが0で主装置からのNACKを予測したとしても、主装置が(ACK送信によって)追加データバイトを要求すると状態\$C8へ移行します。

TWEAが0の間中、TWIは自身の従装置アドレスに応答しませんが、2線直列バスは未だ監視され、アドレス認証はTWEAの設定(1)によって何時でも再開できます。これはTWEAビットがTWIを2線直列バスから一時的に隔離するのに使えることを意味します。

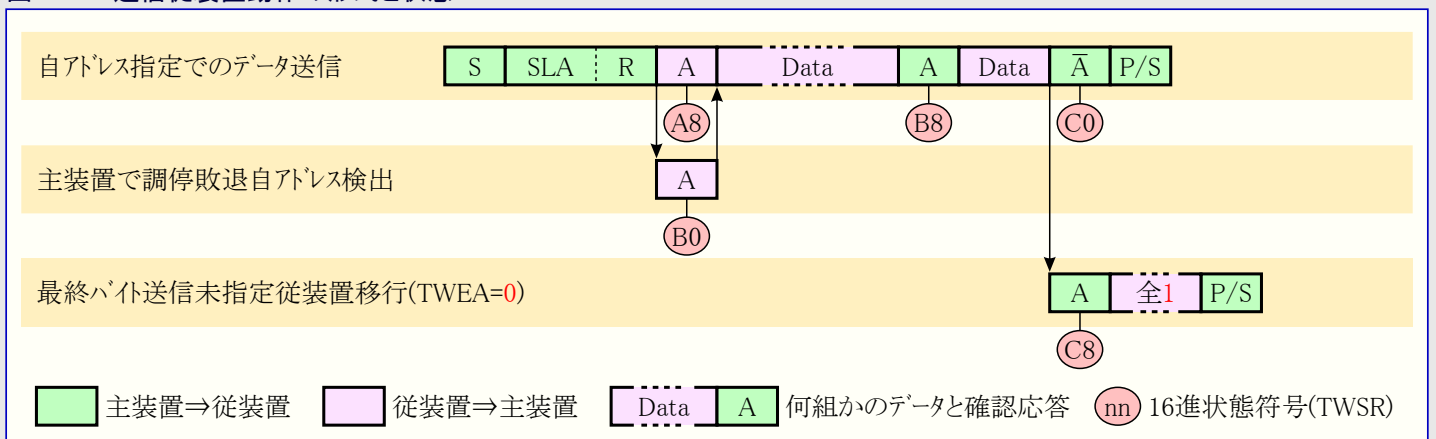
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが(1書き込みによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないことに気付いてください。

これらの(アイドル動作を除く)休止形態から起動すると、2線直列インターフェースデータレジスタ(TWDR)がバスで渡す最後のバイトを反映しないことに注意してください。

表15-5. 送信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$A8	自宛SLA+R受信 ACK応答	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$B0	主装置のSLA+R/Wで 調停敗退/自宛SLA+ R受信/ACK応答	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$B8	データバイト送信 ACK受信	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$C0	データバイト送信 NACK受信	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$C8	最終データバイト送信 (TWEA=0) ACK受信	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信

図15-18. 送信従装置動作の形式と状態



### 15.8.5. その他の状態

定義したTWI状態に従わない2つの状態符号があります。表15-6をご覧ください。

状態\$F8はTWI割り込み要求フラグ(TWINT)が設定(1)されないので適切な情報が利用できないことを示します。これは他の状態間でTWIが直列転送に関係しない時に起きます。

状態\$00は2線直列バス転送中にバス異常が起きたことを示します。バス異常はフレーム形式の不正な位置で開始(START)条件または停止(STOP)条件が起きる時に起きます。このような不正位置の例はアドレスバイト、データバイト、確認応答(ACK)ビットの直列転送中です。バス異常が起きるとTWINTが設定(1)されます。バス異常から回復するには停止(STOP)条件生成許可(TWSTO)ビットが設定(1)され、TWINTが論理1書き込みによって解除(0)されなければなりません。これはTWIをアドレス指定されていない従装置動作にさせ、TWSTOビットを解除(0)させます(TWCRの他のビットは影響されません)。SDAとSCL信号進は開放され、停止条件は送出されません。

表15-6. その他の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$F8	適切な状態情報なし TWINT=0	なし	-	-	-	-	待機または現在の転送続行
\$00	不正な開始条件/停止条件でのバス異常	なし	0	1	1	X	停止条件を送出せずにバスを開放 TWSTO=0

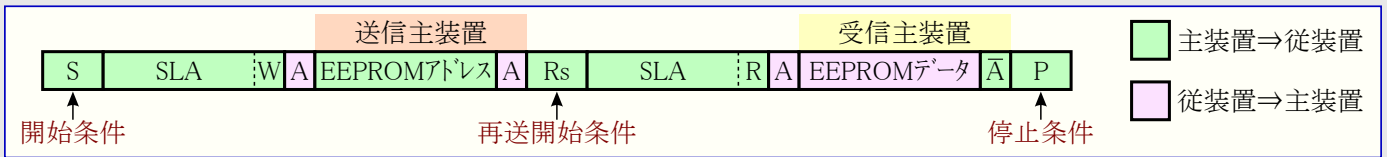
### 15.8.6. 各種TWI動作種別の組み合わせ

いくつかの場合で望んだ動作を満たすために各々のTWI動作種別は組み合わせられなければなりません。例えば直列EEPROMからのデータ読み出しを考えてください。一般的にこのような転送は次の段階を含みます。

1. 転送が開始されなければなりません。
2. EEPROMは読み出すべき場所を指示されなければなりません。
3. 読み出しが実行されなければなりません。
4. 転送が終了されなければなりません。

データが主装置から従装置へとその逆の両方向へ転送されることに注意してください。主装置はどの場所を読みたいかを従装置に指示しなければならず、送信主装置動作の使用を必要とします。その後にデータを従装置から読まねばならず、受信主装置動作の使用を意味します。従って転送方向が切り替えられなければなりません。主装置はこれら全ての段階中にバスの制御を保持しなければならず、この手順は排他的(非分断)操作として行われるべきです。複数主装置システムでこの原則に違反すると、他の主装置が②と③段階間でEEPROM内のデータポインタを変更するかもしれず、(元の)主装置は不正なデータ位置を読むでしょう。このような転送方向の切り替えはアドレスバイトの送信とデータの受信間で再送開始条件を送出することによって成し遂げられます。再送開始条件後も主装置はバスの占有権を保持します。次の図はこの転送の流れを示します。

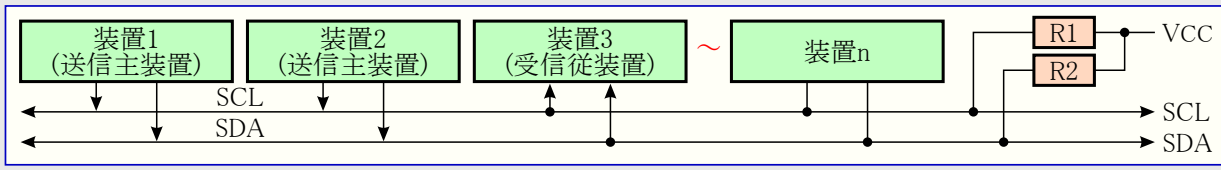
図15-19. 直列EEPROMアクセスでの各種TWI動作種別の組み合わせ



### 15.9. 複数主装置システムでのバス競合と調停

複数の主装置が同じバスに接続されると、それらの1つまたはそれ以上によって同時に送信が開始されるかもしれません。TWIは主装置の1つが転送を続けることを許され、手順内でデータが失われないような方法でこのような状態が扱われることを標準で保証します。2つの主装置が受信従装置へデータを送信することを試みる場合の調停状況の例は以下で図示されます。

図15-20. バスの競合調停例

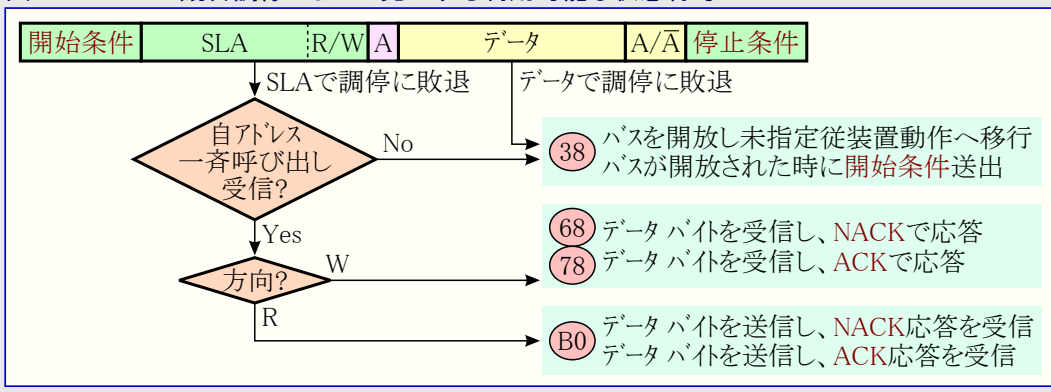


以下で示されるように様々な異なる状況が調停中に起こるかもしれません。

- 複数の主装置が同じ従装置に全く同じ通信を実行する場合。この場合、主/従装置のどれもがバスの衝突について知りません。
- 複数の主装置が異なるデータまたは方向ビット(R/W)で同じ従装置をアクセスする場合。この場合、R/Wビットまたはデータビットのどちらかで調停が起きます。他の主装置がSDAに0を出力する間に1を出力しようとする主装置が調停に敗れます。敗れた主装置はアプリケーションソフトウェアの処置によって未指定従装置動作に切り替えるか、またはバスが開放になるまで待って新規開始条件を送出します。
- 複数の主装置が異なる従装置をアクセスする場合。この場合、SLAビット内で調停が起きます。他の主装置がSDAに0を出力する間に1を出力しようとする主装置が調停に敗れます。SLA内で調停に敗れた主装置は勝った主装置によってアドレス指定されるかを検査するために従装置動作へ切り替えます。アドレス指定されると、R/Wビットの値によって受信従装置(SR)動作または送信従装置(ST)動作へ切り替えます。アドレス指定されないなら、アプリケーションソフトウェアの処置によって未指定従装置動作に切り替えるか、またはバスが開放になるまで待って新規開始条件を送出します。

これは図15-21.で要約されます。利用可能な状態符号は楕円(訳注:原文は円)で与えられます。

図15-21. バスの競合調停によって発生する利用可能な状態符号



## 15.10. SMBusとの適合性

他のどのI<sup>2</sup>C適合インターフェースとでも既知の互換性の問題があるため、設計者はTWIデバイスがSMBusデバイスへ接続する前に知るべきです。SMBus環境での使用に関して以下が注意されるべきです。

- 2線インターフェースのそれらを含むAVRの全ての入出力ピンは供給電圧とGNDの両方に保護ダイオードを持ちます。37頁の図10-1をご覧ください。これはSMBus仕様の必要条件と矛盾します。結果として、供給電圧はAVRから取り除かれてはならず、さもなければ保護ダイオードがバス線を引き下げるでしょう。パワーダウンと休止形態は供給電圧が残っているなら問題ありません。
- TWIのデータ保持時間はSMBusに対する指定よりも小さくなっています。
- SMBusは低速限度を持ち、一方はI<sup>2</sup>C持ちません。SMBus環境での主装置の時に、より遅いバス速度がSMBus従装置で時間超過を起動するため、AVRはバス速度が仕様以下に低下しないことを確実にしなければなりません。AVRが従装置に構成設定された場合、TWI単位部が時間超過を識別しないため、バス固定化の可能性があります。

## 15.11. TWI用レジスタ

### 15.11.1. TWBR – TWIビット速度レジスタ (TWI Bit Rate Register)

ビット (\$B8)	7	6	5	4	3	2	1	0	
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
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 – TWBR7~0 : TWIビット速度選択 (TWI Bit Rate Register)

TWBRはビット速度発生器用の分周値を選びます。ビット速度発生器は主装置動作でのSCLクロック周波数を生成する周波数分周器です。ビット速度の計算については87頁の「[ビット速度発生器](#)」をご覧ください。

TWIが主装置動作で動く場合、TWBRは10またはそれ以上でなければなりません。

### 15.11.2. TWCR – TWI制御レジスタ (TWI Control Register)

ビット (\$BC)	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

TWCRはTWI動作の制御に使われます。TWIの許可、バス上に開始条件を印加することによる主装置のアクセス開始、受信装置の応答生成、停止条件の生成、バスにデータを送出するためのTWIデータレジスタ(TWDR)への書き込み中のバスの一時停止制御に使われます。TWDRがアクセス不能の間にTWDRへ書き込もうとする場合の上書き発生も示します。

#### • ビット7 – TWINT : TWI割り込み要求フラグ (TWI Interrupt Flag)

このビットはTWIが現在の作業を終了し、応用ソフトウェアの応答が予測される時、ハードウェアによって設定(1)されます。TWI制御レジスタ(TWCR)のTWI割り込み許可(TWIE)ビットと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、MCUはTWI割り込みベクタへ飛びます。TWINTフラグが設定(1)の間、SCLのLow期間は引き伸ばされます。TWINTフラグはソフトウェアによる論理1書き込みによって解除(0)されなければなりません。このフラグが割り込みルーチンを実行する時に自動的に解除(0)されないことに注意してください。このフラグの解除(0)がTWIの動作を始めるので、このフラグの解除(0)前にTWIアドレスレジスタ(TWAR)、TWIデータレジスタ(TWDR)、TWI状態レジスタ(TWSR)への全てのアクセスが完了していなければならないことにも注意してください。

#### • ビット6 – TWEA : 確認応答(ACK)許可 (TWI Enable Acknowledge Bit)

TWEAビットは確認応答(ACKパルス)の生成を制御します。TWEAビットが1を書かれ、次の条件に合致すると、TWIバスにACKパルスが生成されます。

- 装置が自分用の従装置アドレスを受信した場合。
- TWIアドレスレジスタ(TWAR)の一斉呼び出し検出許可(TWGCE)ビットが設定(1)されている時に一斉呼び出しを受信した場合。
- 主受信装置または従受信装置動作でデータバイトを受信した場合。

TWEAビットに0を書くことにより、一時的且つ仮想的に装置を2線直列バスから切り離せます。アドレス認証はその後に再びTWEAビットへ1を書くことによって再開できます。

#### • ビット5 – TWSTA : 開始(START)条件生成許可 (TWI START Condition Bit)

2線直列バスの主装置になるのを欲する時に応用はTWSTAビットに1を書きます。TWIハードウェアはバスが利用可能かを検査し、開放ならばバスに開始条件を生成します。しかし、バスが未開放の場合、TWIは停止条件が検出されるまで待ち、その後にバス所有権を要求する新規開始条件を生成します。TWSTAは開始条件が送出されてしまった時にソフトウェアで解除(0)されなければなりません。

#### • ビット4 – TWSTO : 停止(STOP)条件生成許可 (TWI STOP Condition Bit)

主装置動作でTWSTOビットに1を書くことが2線直列バスに停止条件を生成します。停止条件がバスで実行されると、TWSTOビットは自動的に解除(0)されます。従装置動作でのTWSTOビットの設定(1)は異常状態からの回復に使えます。これは停止条件を生成しますが、TWIは明確に指定されていない従装置動作に戻り、SCL、SDA信号線をHi-Z状態に開放します。

#### • ビット3 – TWWC : TWI上書き発生フラグ (TWI Write Collision Flag)

TWI割り込み要求フラグ(TWINT)が0の時にTWIデータレジスタ(TWDR)への書き込みを試みると、このTWWCフラグが設定(1)されます。このフラグはTWINTが1の時のTWDR書き込みによって解除(0)されます。

#### • ビット2 – TWEN : TWI動作許可 (TWI Enable Bit)

TWENビットはTWI動作を許可し、TWIインターフェースを活性(有効)にします。TWENが1を書かれると、TWIはSCL、SDAピンに接続したI/Oピンを制御できるようになり、スパイク濾波器とスレーブ制限器を許可します。このビットが0を書かれると、TWIがOFFにされ、どんな進行中の動作にも関係なく、全てのTWI送信が終了されます。

#### • ビット1 – Res : 予約 (Reserved Bit)

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0 – TWIE : TWI 割り込み許可 (TWI 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)されていると、TWI割り込み要求フラグ(TWINT)が1である限り、TWI割り込み要求が活性に(発生)されます。

**15.11.3. TWSR – TWI 状態レジスタ (TWI Status Register)**

ビット (\$B9)	7	6	5	4	3	2	1	0	TWSR
	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	1	1	1	1	1	0	0	0	

• **ビット7~3 – TWS7~3 : TWI 状態 (TWI Status)**

これら5ビットはTWI論理回路と2線直列バスの状態を反映します。各種状態符号は91頁の「転送種別」で記述されます。TWSRから読む値が5ビットの状態符号と2ビットの前置分周値の両方を含むことに注意してください。応用設計者は状態ビットを検査する時に前置分周器ビットを0で隠すべきです。これは前置分周器設定に関係なく状態検査を行います。この手法は特記事項を除いてこのデータシート内で使われます。

• **ビット2 – Res : 予約 (Reserved Bit)**

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1,0 – TWPS1,0 : TWI 前置分周器選択 (TWI Prescaler Bits)**

これらのビットは読み書きでき、ビット速度の前置分周器を制御します。

ビット速度を計算するには87頁の「ビット速度発生器」をご覧ください。TWPS1,0の値はこの式で使われます。

表15-7. TWIビット速度前置分周器選択

TWPS1	0	0	1	1
TWPS0	0	1	0	1
分周値	1	4	16	64

**15.11.4. TWDR – TWI データ レジスタ (TWI Data Register)**

ビット (\$BB)	7	6	5	4	3	2	1	0	TWDR
	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	1	1	1	1	1	1	1	1	

送信動作でのTWDRは送信されるべき次のデータバイトを含みます。受信動作でのTWDRは最後に受信したバイトを含みます。TWDRはTWIがバイトを移動する手順でない間に書き込み可能です。これはTWI制御レジスタ(TWCR)のTWI割り込み要求フラグ(TWINT)がハードウェアによって設定(1)されると起きます。最初のTWI割り込みが起こる前にデータレジスタ(TWDR)は使用者によって初期化できないことに注意してください。TWDRのデータはTWINTが安定して設定(1)されている限り存続します。データが移動出力される間、バスのデータが同時に移動入力されます。TWI割り込みによる休止形態からの起動後を除いて、TWDRは常にバスに現れる最後のバイトを含みます。この例外の場合のTWDRの内容は不定です。バス調停に敗れた場合の主装置から従装置への移行でもデータは失われません。確認応答(ACK)ビットの扱いはTWI論理回路によって自動的に制御され、CPUはACKビットを直接的にアクセスできません。

• **ビット7~0 – TWD7~0 : TWI データ (TWI Data)**

これら8ビットは送信されるべき次のデータバイト、または2線直列バスで最後に受信したデータバイトを構成します。

**15.11.5. TWAR – TWI(従装置)アドレス レジスタ (TWI (Slave) Address Register)**

ビット (\$BA)	7	6	5	4	3	2	1	0	TWAR
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	1	1	1	1	1	1	1	0	

TWARは従装置の送受信装置として設定した時にTWIが応答する7ビット従装置アドレスを(TWAR上位7ビットに)設定されるべきで、主装置動作では必要とされません。複数主装置のシステムでは、他の主装置によって従装置としてアドレス指定され得る主装置に於いて、TWARは設定されなければなりません。

TWARの最下位ビット(TWGCE)は一斉呼び出しアドレス(\$00)認証の許可に使われます。これらは受信した直列アドレスで従装置アドレス(と許可ならば一斉呼び出しアドレス)を捜す関連アドレス比較器です。一致が見つかるとう割り込み要求が生成されます。

• **ビット7~1 – TWA6~0 : TWI 従装置アドレス (TWI (Slave) Address)**

これら7ビットはTWI部の従装置アドレスを構成します。

• **ビット0 – TWGCE : 一斉呼び出し検出許可 (TWI General Call Recognition Enable Bit)**

設定(1)なら、このビットは2線直列バスを伝って与えられる一斉呼び出しの認証(検出)を許可します。



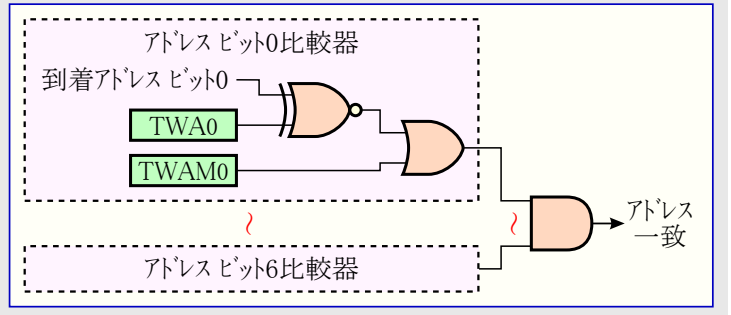
### 15.11.6. TWAMR – TWI(従装置)アドレス遮蔽レジスタ (TWI (Slave) Address Mask Register)

ビット (\$BD)	7	6	5	4	3	2	1	0	
	TWAM6	TWAM5	TWAM4	TWAM3	TWAM2	TWAM1	TWAM0	–	TWAMR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
初期値	0	0	0	0	0	0	0	0	0

#### • ビット7~1 – TWAM6~0 : TWI従装置アドレス遮蔽 (TWI (Slave) Address Mask)

TWAMRは7ビットの従装置アドレス遮蔽値を格納できます。TWAMR内の各ビットはTWI(従装置)アドレスレジスタ(TWAR)内の対応するアドレスビットを遮蔽(禁止)します。遮蔽ビットが1に設定されると、その後のアドレス一致論理回路は到着アドレスビットとTWAR内の対応ビット間の比較を無視します。図15-22はアドレス一致論理回路を詳細に示します。

図15-22. TWIアドレス一致論理 構成図



#### • ビット0 – Res : 予約 (Reserved Bit)

このビットは予約されており、常に0として読みます。

### 15.11.7. TWHSR – TWI高速レジスタ (TWI High Speed Register)

ビット (\$BE)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	–	TWHS	TWHSR
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	0

#### • ビット7~1 – Res : 予約 (Reserved Bit)

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 – TWHS : TWI高速許可 (TWI High Speed Enable)

このビットに1を書くことによってTWI高速動作が許可されます。この動作ではTWIクロックとして分周されないシステムクロックが選ばれます。17頁の図6-1をご覧ください。

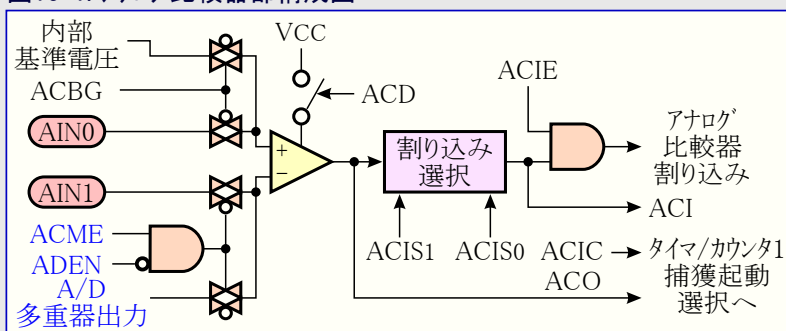
TWI高速動作はI/Oクロック周波数( $clk_{I/O}$ )よりも正確に2倍高い高速クロック( $clk_{TWIHS}$ )を必要とします。これは2倍低く前置分周されるI/Oクロック周波数( $clk_{I/O}$ )を使用者が保証しなければなりません。例えば、供給元クロックして8MHz発振器が選ばれた場合、使用者はシステムクロック(従って、とI/Oクロック)を4MHzに下げる尺度で前置分周器を設定しなければなりません。クロックシステムについてのより多くの情報に関しては17頁の「クロック体系」をご覧ください。

## 16. 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。この比較器の出力はタイマ/カウンタ1捕獲機能を起動するように設定することができます。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16-1.で示されます。

ADC入力多重器を使うには**ADC電力削減(PRADC)ビット**が禁止されなければなりません。これは電力削減レジスタ(PRR)のPRADCビットを解除することによって行われます。より多くの詳細については24頁の「**PRR - 電力削減レジスタ**」をご覧ください。

図16-1. アナログ比較器部構成図



注: A/D多重器出力については表16-1.をご覧ください。  
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47頁の表10-11.を参照してください。

### 16.1. アナログ比較器入力選択

アナログ比較器への反転入力を取り替えるのにADC7~0のどれかを選ぶことができます。A/D変換の多重器がこの入力選択に使われ、従ってこの機能を利用するためにA/D変換部がOFF(動作禁止)にされなければなりません。ADCSRBの**アナログ比較器多重器許可(ACME)ビット**が設定(1)され、A/D変換部がOFF(ADCSRAのADENビットが0)にされていれば、表16-1.で示されるように**ADMUXのチャネル選択(MUX2~0)ビット**がアナログ比較器への反転入力を取り替えるための入力ピンを選びます。ACMEが解除(0)またはADENが設定(1)されると、AIN1がアナログ比較器への反転入力に印加されます。

表16-1. アナログ比較器反転入力選択

ACME	ADEN	MUX2~0	アナログ比較器反転入力	備考
0	x	x x x	AIN1	
1	0	0 0 0	ADC0	
		0 0 1	ADC1	
		0 1 0	ADC2	
		0 1 1	ADC3	
		1 0 0	ADC4	
		1 0 1	ADC5	
		1 1 0	ADC6	TQFP, QFN/MLF32
		1 1 1	ADC7	外圍器のみ

### 16.2. アナログ比較器用レジスタ

#### 16.2.1. ADCSRB - A/D変換制御/状態レジスタB (ADC Control and Status Register B)

ビット (\$7B)	7	6	5	4	3	2	1	0	
	-	ACME	-	-	-	ADTS2	ADTS1	ADTS0	ADCSRB
Read/Write	R	R/W	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット6 - ACME** : アナログ比較器多重器許可 (Analog Comparator Multiplexer Enable)

このビットが論理1を書かれ、A/D変換部がOFF(ADCSRAのADENビットが0)にされると、A/D変換の多重器がアナログ比較器への反転入力を選びます。このビットが論理0を書かれると、AIN1がアナログ比較器の反転入力に印加されます。このビットの詳細な記述については上記の「**アナログ比較器入力選択**」をご覧ください。

### 16.2.2. ACSR – アナログ比較器 制御/状態レジスタ (Analog Comparator Control and Status Register)

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	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き得ます。

• **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置き換わります。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とき、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。27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、その後に直接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除去機能と端(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せん。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1割り込み許可レジスタ(TIMSK1)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かを決めます。各種設定は表16-2.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16-2. 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(両端)
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

### 16.2.3. DIDR1 – デジタル入力禁止レジスタ1 (Digital Input Disable Register 1)

ビット	7	6	5	4	3	2	1	0	
(\$7F)	-	-	-	-	-	-	AIN1D	AIN0D	DIDR1
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7~2 – 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 – 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に、このビットは論理1を書かれるべきです。

## 17. A/D変換器

### 17.1. 特徴

- 10ビット分解能
- 積分非直線性誤差1 LSB
- 絶対精度±2 LSB
- 14μsの変換時間
- 最大分解能で15kSPS(採取/s)
- 6チャンネルのシングルエンド入力多重器内蔵
  - ・ 32リードTQFP、32パッドQFN、32球UFBGAでの2つの追加入力
- 温度感知器入力チャンネル
- A/D変換結果読み出しに対する任意の左揃え
- 0~VCC A/D変換入力電圧範囲
- 選択可能な1.1V A/D変換基準電圧
- 連続と単独の変換動作
- 割り込み元の自動起動によるA/D変換開始
- A/D変換完了割り込み
- 休止形態雑音低減機能

### 17.2. 概要

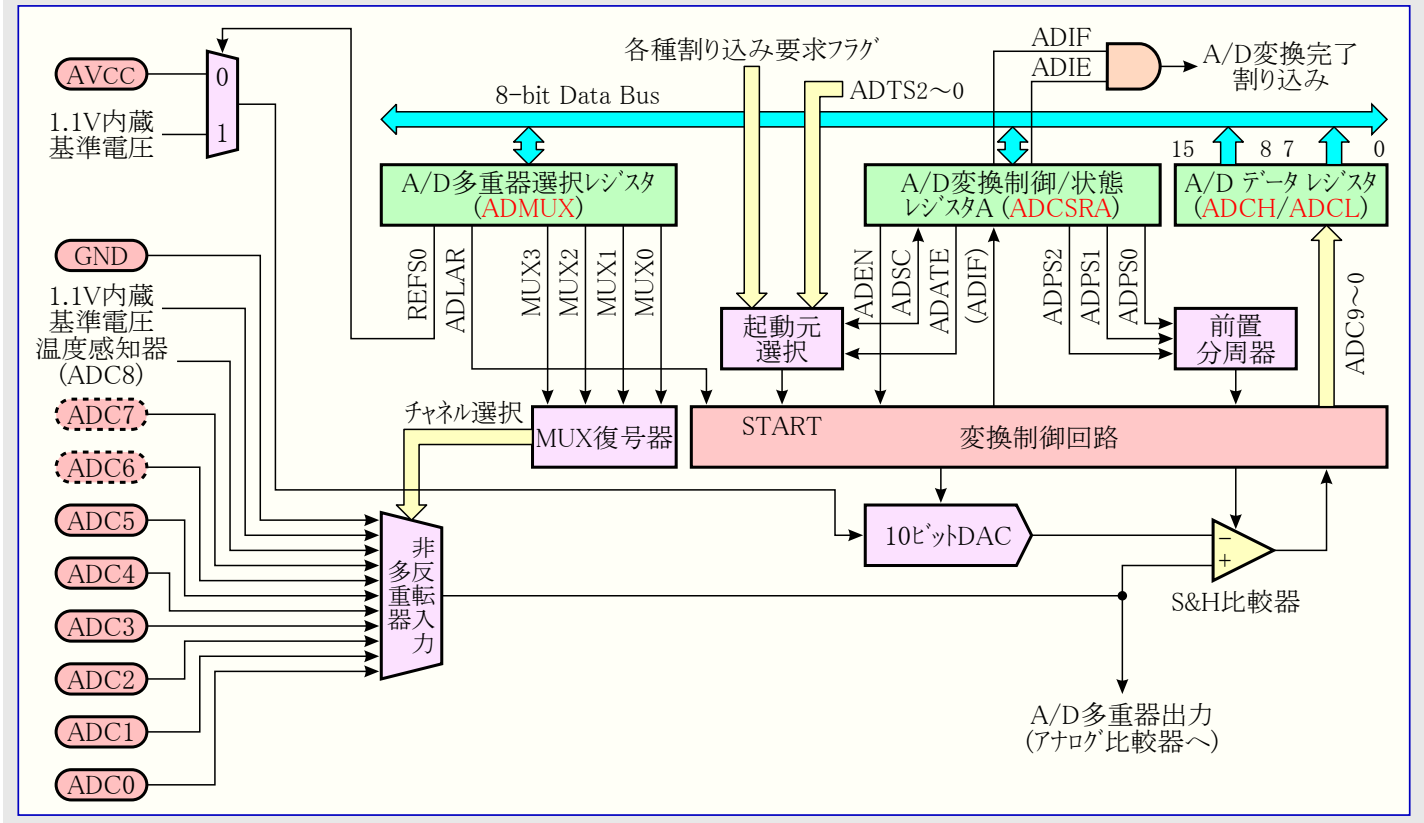
ATtiny48/88は10ビット逐次比較A/D変換器(ADC)が特徴です。ADCは6つ(または32リード/パッド/球の外圍器では8つ)のシングルエンド入力ピンと、内部温度感知器から来るシングルエンド電圧チャンネルの1つの内部からの電圧測定をADCに許す9チャンネルのアナログ多重器に配線されています。シングルエンド電圧入力は0V(GND)が基準です。

このA/D変換器はA/D変換器への入力電圧が変換中に一定の値で保持されることを保証する採取&保持(S/H)回路を含みます。A/D変換部の構成図は図17-1.で示されます。

ADC用に独立したアナログ電圧供給ピン(AVCC)があります。VCCとAVCCの電圧供給ピン間の電圧差は0.3Vを越えてはいけません。アナログ電圧供給ピンの接続方法は111頁の「雑音低減技術」項をご覧ください。

公称1.1Vの内部基準電圧がチップ上で提供されます。代わりに基準電圧としてAVCCを使うことができます。

図17-1. A/D変換器部構成図



### 17.3. 操作

A/D変換部を使えるようにするため、ADC電力削減(PRADC)ビットが禁止されなければなりません。これは電力削減レジスタ(PRR)のPRADCビットを解除することによって行われます。より多くの詳細については24頁の「PRR - 電力削減レジスタ」をご覧ください。

A/D変換部は逐次比較を通してアナログ入力電圧を10ビットのデジタル値に変換します。最小値はGNDを表し、最大値は基準電圧を表します。A/D変換器基準電圧はA/D多重器選択レジスタ(ADMUX)の基準電圧選択(REFS0)ビット書き込みによって選ばれます。選択はAVCC供給ピンと1.1V内部基準電圧です。

アナログ入力チャンネルはA/D多重器選択(ADMUX)レジスタのチャンネル選択(MUX3~0)ビットへの書き込みによって選ばれます。GNDと固定基準電圧(1.1V内蔵基準電圧(VBG))だけでなく、どのADC入力ピン(ADC7~0)もA/D変換器のシングルエンド入力として選べます。

A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットの設定(1)によって(動作が)許可されます。基準電圧と入力チャンネルの選択はADENが設定(1)されるまで実施しません。ADENが解除(0)されているとA/D変換部が電力を消費しないので、節電をする休止形態へ移行する前にA/D変換部をOFFに切り替えることが推奨されます。

A/D変換部はA/Dデータレジスタ(ADCH,ADCL)で示される10ビットの結果を生成します。既定では、この結果は右揃え(16ビットのビット0側10ビット)で表されますが、ADMUXで左揃え選択(ADLAR)ビットを設定(1)することにより、任意で左揃え(16ビットのビット15側10ビット)で表せます。

この結果が左揃え補正され、8ビットを越える精度が必要とされない場合はADCHだけを読むことで足ります。さもないとデータレジスタの内容が同じ変換に属すること(からの結果)を保証するため、ADCLが初めに、次にADCHが読まれなければなりません。一度ADCLが読まれると、A/D変換器からのA/Dデータレジスタ(ADCH,ADCL)アクセスが阻止されます。これはADCLが読まれてしまい、ADCHが読まれる前に変換が完了すると、どちらのレジスタ(ADCH,ADCL)も更新されず、その変換からの結果が失われることを意味します。ADCHが読まれると、ADCH,ADCLへのA/D変換器アクセスが再び許可されます。

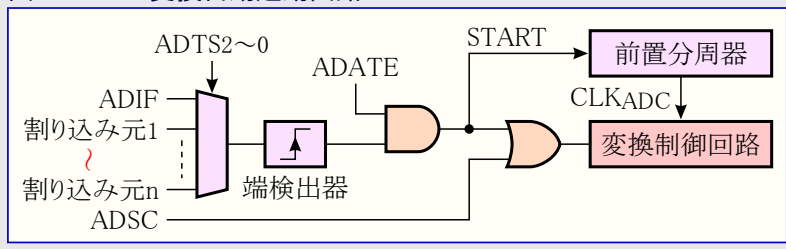
A/D変換部には変換完了時に起動できる自身の割り込みがあります。A/DデータレジスタへのA/D変換器アクセスがADCLとADCHの読み込み間で禁止されている場合、例えその変換結果が失われても割り込みは起動します。

### 17.4. 変換の開始

電力削減レジスタ(PRR)のA/D変換電力削減(PRADC)ビットを解除(0)することによってADCが給電されることを保証してください(24頁の「PRR - 電力削減レジスタ」をご覧ください)。単独変換はADCSRAで変換開始(ADSC)ビットに論理1を書くことによって開始されます。このビットは変換が進行中である限り1に留まり、変換が完了される時にハードウェアによって解除(0)されます。変換が進行中に違う入力チャンネルが選ばれると、A/D変換部はそのチャンネル変更を実行する前に現在の変換を済ませます。

代わりに、変換は様々な起動元によって自動的に起動できます。自動起動はA/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットの設定(1)によって許可されます。起動元はA/D変換制御/状態レジスタB(ADCSRB)のA/D変換起動元選択(ADTS2~0)ビットの設定によって選ばれます(起動元の一覧についてはADTSビットの記述をご覧ください)。選んだ起動信号上に上昇端が起きると、A/D変換用前置分周器がリセットして変換が開始されます。これは一定間隔での変換開始の方法を提供します。変換完了時、起動信号が未だ設定(1)されている場合、新規の変換は開始されません。変換中にこの起動信号上で別の上昇端が起きると、その端(エッジ)は無視されます。指定した割り込みが禁止またはステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)でも、割り込み要求フラグが設定(1)されることに注意してください。従って割り込みを起さずに変換が起動できます。けれども次の割り込み要因で新規変換を起動するために、割り込み要求フラグは解除(0)されなければなりません。

図17-2. A/D変換自動起動回路



起動元としてA/D変換完了割り込み要求フラグ(ADIF)を使うことは、A/D変換器に実行中の変換が完了されると直ぐに新規変換を開始させます。そのためにA/D変換器は連続動作で動き、継続的な採取(変換)とA/Dデータレジスタ更新を行います。最初の変換はADCSRAでADSCビットに論理1を書くことによって始めなければなりません。この動作でのA/D変換器はA/D変換完了割り込み要求フラグ(ADIF)が解除(0)されるかどうかにかかわらず、連続的な変換を実行します。

自動起動が許可されている場合、ADCSRAのADSCビットに論理1を書くことによって単独変換を開始できます。ADSCは変換が進行中かを決めるためにも使えます。ADSCビットは変換がどう開始されたかに拘らず、変換中は1として読みます。

変換はA/D変換雑音低減機能の使用によっても開始され得ます。この機能はアイドル休止動作とA/D変換雑音低減休止動作中に変換を許可します。詳細については111頁の「雑音低減機能」をご覧ください。(訳注:共通性から2行追加)

17.5. 前置分周と変換タイミング

既定での逐次比較回路は最大分解能を得るのに50~200kHzの入力クロック周波数を必要とします。10ビットよりも低い分解能が必要とされるなら、A/D変換器への入力クロック周波数はより高い採取速度を得るために200kHzよりも高くできます。1MHzよりも高い入力クロック周波数の使用は推奨されません。

図17-3.で図解されるように、A/D変換部は100kHz以上のどんなCPUクロックからも受け入れ可能なA/D変換クロック周波数を生成する前置分周器を含みます。この前置分周器はA/D変換制御/状態レジスタ(ADCSRA)のA/Dクロック選択(ADPS2~0)ビットによって設定されます。前置分周器はADCSRAでA/D許可(ADEN)ビットの設定(1)によってA/D変換部がONにされた瞬間から計数を始めます。前置分周器はADENビットが設定(1)される限り走行を維持し、ADENが0の時は継続的にリセットします。

ADCSRAのA/D変換開始(ADSC)ビットの設定(1)によってシングルエンド入力の変換を起動すると、その変換は直後の変換クロックの上昇端で始まります。

通常の変換は13変換クロック周期で行われます。下の図17-4.で示されるように、A/D変換部がONされる(ADCSRAのADEN=1)後の最初の変換はアナログ回路を初期化するために25変換クロック周期で行われます。

図17-3. A/D変換前置分周器部構成

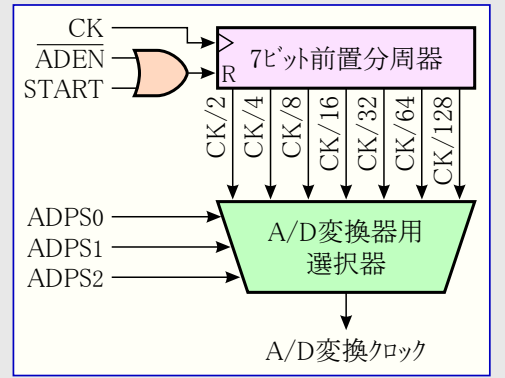
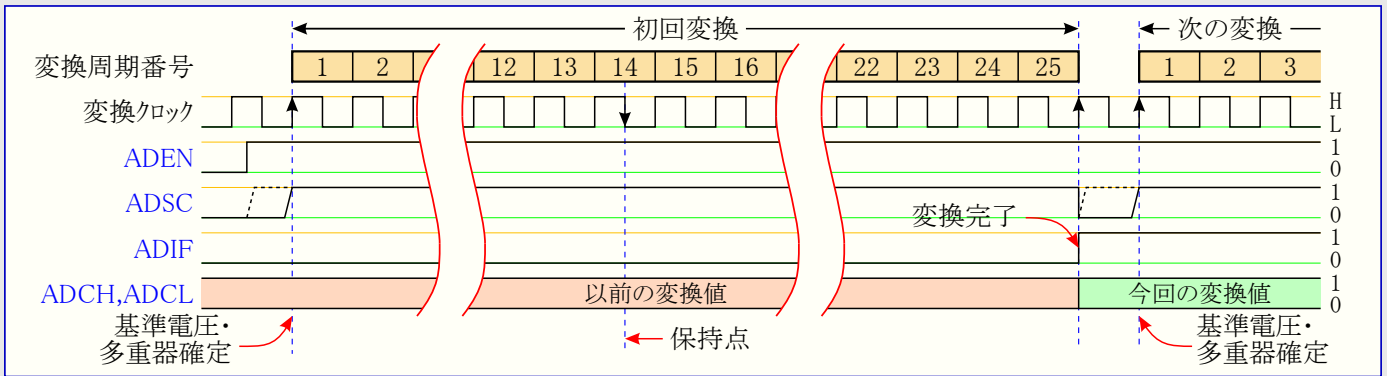
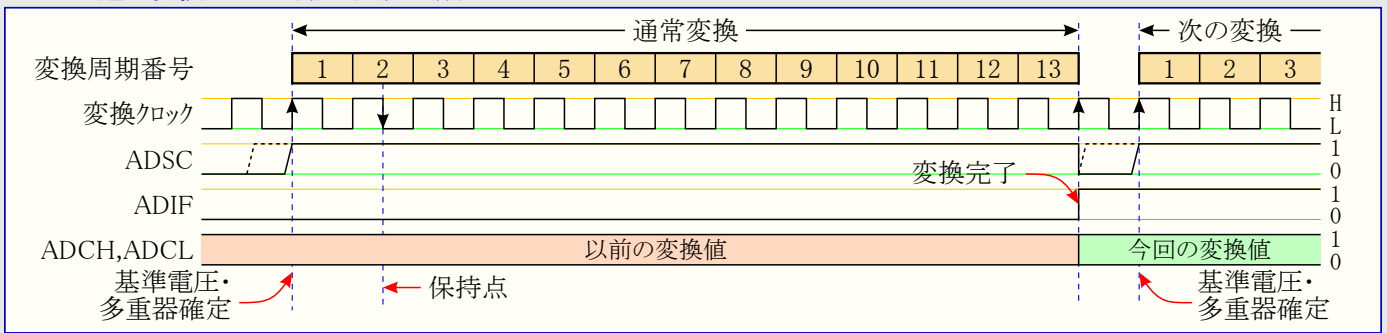


図17-4. 初回変換タイミング (単独変換動作)



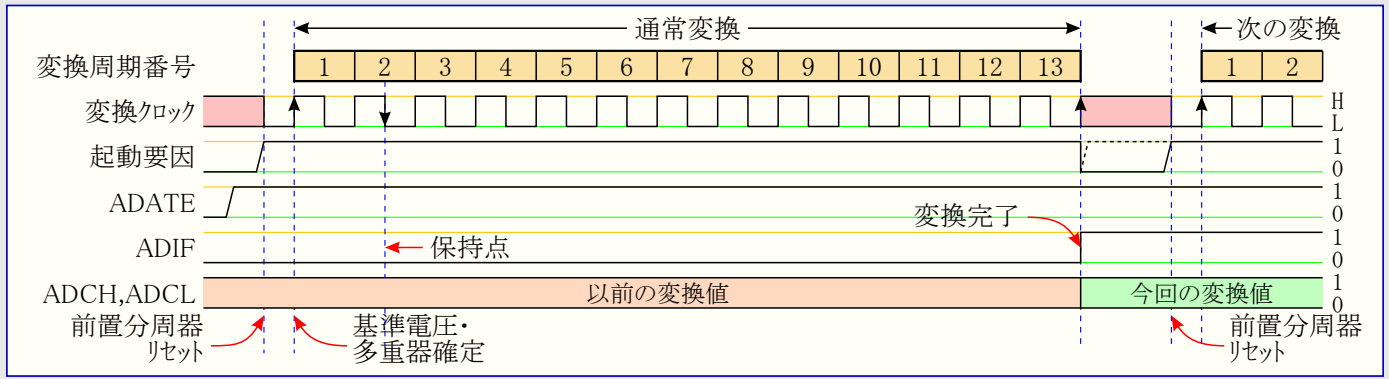
実際の採取&保持(保持開始点)は通常変換の開始後1.5変換クロック周期、初回変換の開始後13.5変換クロック周期で行われます。図17-5.をご覧ください。変換が完了すると、結果がA/Dデータレジスタ(ADCH,ADCL)に書かれ、ADCSRAのA/D変換完了割り込み要求フラグ(ADIF)が設定(1)されます。単独変換動作(ADATE=0)では同時にADCSRAのA/D変換開始(ADSC)ビットが解除(0)されます。その後ソフトウェアは再びADSCを設定(1)でき、新規変換は変換クロックの最初の上昇端で開始されます。

図17-5. 通常変換タイミング (単独変換動作)



自動起動が使われると、図17-6.で示されるように前置分周器は起動要因発生時にリセットされます。これは起動要因から変換開始までの一定の遅延を保証します。この動作での採取&保持は起動要因となる信号の上昇後、2変換クロック周期で採取が行われます。同期化論理回路(端(エッジ)検出器)に対して追加の3 CPUクロック周期が費やされます。

図17-6. 通常変換タイミング (自動起動変換動作)



連続変換動作(ADATE=1)では変換完了後直ちに新規変換が開始され、一方ADSCは1に留まります。図17-7.をご覧ください。変換時間の要約については表17-1.をご覧ください。

図17-7. 連続変換動作タイミング

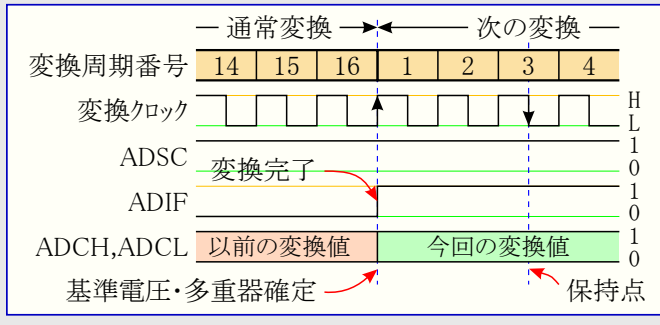


表17-1. A/D変換時間

変換種別	保持点	変換時間
初回変換	13.5	25
シングルエンド入力通常変換	1.5	13
自動起動変換	2	13.5
連続変換	2.5	14

注: 変換時間を除く各値は変換開始からの変換クロック数です。

## 17.6. チャネル変更と基準電圧選択

A/D多重器選択レジスタ(ADMUX)のチャネル選択(MUX3~0)ビットと基準電圧選択(REFS0)ビットはCPUが乱順にアクセスするための一時レジスタを通して単独緩衝されます。これはチャネルと基準電圧の選択が変換中の安全なところでだけ行うのを保証します。チャネルと基準電圧の選択は変換が開始されるまで継続的に更新されます。一旦変換が始まると、A/D変換器に対して十分な採取/変換時間を保証するためにチャネルと基準電圧の選択は固定されます。継続的な更新は変換完了(ADCSRAのADIF=1)前の最後の変換クロック周期で再開します。ADCSRAの変換開始(ADSC)ビットが書かれた後の次の変換クロックの上昇端で変換が始まることに注意してください。従って使用者はADSC書き込み後、1変換クロック周期(経過)まで新しいチャネルまたは基準電圧選択値をADMUXに書かないことが推奨されます。

自動起動が使われる場合、起動要因の正確な時間は確定できません。変換が新規設定によって影響されるように制御するにはADMUXの更新時に特別な注意が払われなければなりません。

ADCSRAのA/D許可(ADEN)とA/D変換自動起動許可(ADATE)の両方が1を書かれると、何時でも割り込みが起き得ます。この期間でADMUXが変更されると、使用者は次の変換が旧設定または新設定どちらを基準にされるかを知ることができません。ADMUXは次の方法で安全に更新できます。

- ADENまたはADATEが解除(0)されているとき。
- 変換開始後、最低1変換クロック周期経過後の変換中。
- 変換後から、変換起動元として使った割り込みフラグが解除(0)される直前まで。

これら条件の1つでADMUXを更新すると、新設定は次のA/D変換に影響を及ぼします。

### 17.6.1. A/D入力チャネル

チャネル変更する時に使用者は正しいチャネルが選ばれるのを保証するために次の指針を守るべきです。

単独変換動作では常に変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法はチャネル変更の前に変換が完了するまで待つことです。

連続変換動作では常に最初の変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法は最初の変換が完了するまで待ち、その後にチャネル変更をすることです。既に次の変換が自動的に開始されているので、次の結果は直前のチャネル選択を反映します。それに続く変換は新しいチャネル選択を反映します。

## 17.6.2. A/D変換基準電圧

このA/D変換基準電圧(VREF)はA/D変換に対する変換範囲を示します。VREFを越えるシングルエンド入力チャネルは\$3FFで打ち切る符号に帰着します。VREFはVCC、1.1V内部基準電圧、外部AREFピンのどれかとして選べます。1.1V内部基準電圧は内部増幅器を通して内部ハントギャップ基準電圧(VBG)から生成されます。

基準電圧源切り替え後の最初のA/D変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

## 17.7. 雑音低減機能

A/D変換部は**休止形態**中の変換を可能にする雑音低減機能が特徴です。これはCPUコアと他の周辺I/Oが誘導した雑音を減らします。この機能は**A/D変換雑音低減動作**と**アイドル動作**で使えます。この機能を使うには次の手順が使われるべきです。

1. A/D変換部が許可(ADEN=1)され、変換中でない(ADSC=0)ことを確認してください。単独変換動作が選択(ADATE=0)され、且つA/D変換完了割り込みが許可(ADIE=1)されていなければなりません。
2. A/D変換雑音低減(またはアイドル)動作に移行してください。一旦CPUが停止されてしまうと、A/D変換部は変換を始めます。
3. A/D変換完了前に他の割り込みが起こらなければ、A/D変換完了割り込みはCPUを起動してA/D変換完了割り込みルーチンを実行します。A/D変換完了前に他の割り込みがCPUを起動すると、その割り込みが実行され、A/D変換完了割り込み要求はA/D変換完了時に生成されます。CPUは新規**SLEEP**命令が実行されるまで活動動作に留まります。

アイドル動作とA/D変換雑音低減動作を除く他の休止形態へ移行する時にA/D変換部が自動的にOFFへ切り替えられないことに注意してください。使用者は余分な消費電力を避けるため、このような休止形態へ移行する前にADENへ**0**を書くことが推奨されます。

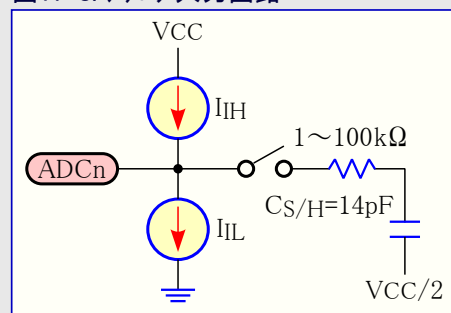
## 17.8. アナログ入力回路

シングルエンド入力チャネルのアナログ回路は**図17-8**で図示されます。ADCnに印加したアナログ(信号)源はそのチャネルがADC入力として選ばれているかどうかにかかわらず、ピン容量とそのピンの漏れ電流に左右されます。そのチャネルが選ばれると、(アナログ)信号源は直列抵抗(入力経路の合成抵抗)を通してS/Hコンデンサを駆動しなければなりません。

A/D変換部は概ね10kΩ若しくはそれ以下の出力インピーダンスのアナログ信号用に最適化されています。このような(アナログ)信号源が使われるなら、採取時間は無視してもよいでしょう。より高いインピーダンスの(アナログ)信号源が使われる場合、採取時間は広範囲に変化し得るS/Hコンデンサを充電するために(アナログ)信号源がどれくらいの時間を必要とするかに依存します。緩やかに変化する信号で、必要とされるS/Hコンデンサへの充放電を最小とするため、使用者は低インピーダンス信号源だけを使うことが推奨されます。

特定できない信号の渦からの歪を避けるために、どのチャネルに対してもナイキスト周波数( $f_{ADC}/2$ )よりも高い信号成分が存在すべきではありません。使用者はADC入力として信号を印加する前に、低域通過濾波器(ローパスフィルタ)で高い周波数成分を取り除くことが推奨されます。

図17-8. アナログ入力回路



**注:** 図内のコンデンサ容量はS/Hコンデンサとデバイス内の何れかの浮遊容量または寄生容量を含む合計容量を叙述します。与えられた値は最悪(最大)値です。

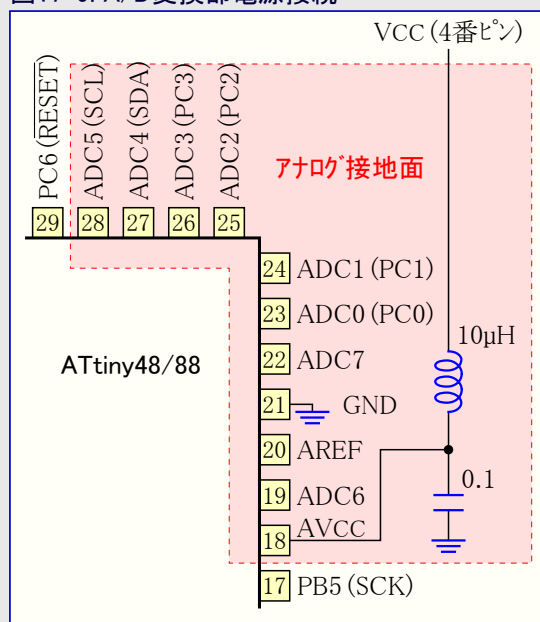
## 17.9. アナログ雑音低減技術

デバイス内外のデジタル回路がアナログ測定の精度に影響を及ぼすかもしれないEMIを発生します。変換精度が重要なとき、次の技法を適用することによって雑音レベルを低減できます。

- アナログ信号経路を可能な限り最短にしてください。
- アナログ信号経路がアナログGND面上を走ることを確認してください。
- アナログ信号経路を高速切り替えデジタル信号線から充分離すことを守ってください。
- 何れかのADCポートピンがデジタル出力として使われる場合、これらを変換進行中に決して切り替えてはなりません。
- アナログ電圧供給(AVCC)ピンは**図17-9**で示されるようにLC濾波器を経由してデジタル供給電圧(VCC)に接続されるべきです。

高いA/D変換精度が必要とされるとき、「**雑音低減機能**」で記述されるようにA/D変換雑音低減動作の使用が推奨されます。これは特にシステムクロック周波数が1MHz以上、または「**温度測定**」で記述されるようにA/D変換が内部温度感知器読み込みに使われる場合です。正しい外部バスコン配置での良いシステム設計はA/D変換雑音低減動作の必要性を低減します。

図17-9. A/D変換部電源接続



(訳注) 図17-9.は原図に対し修正され、ピン名は部分的に省略されています。

## 17.10. A/D変換の精度定義

シングルエンド入力電圧のnビットA/D変換はGNDとVREF間を $2^n$ で直線的に変換します。最低値符号は0として読み、最高値符号は $2^n - 1$ として読みます。以下の各種パラメータは理想状態からの偏差を表します。

### • 変位(オフセット)誤差 - 図17-10.

最初の遷移点(\$000から\$001)で理想遷移点(差0.5 LSB)と比べた偏差です。理想値は0 LSBです。

### • 利得誤差 - 図17-11.

変位誤差補正後の最後の遷移点(\$3FEから\$3FF)で理想遷移点(最大差1.5 LSB以下)と比べた偏差です。理想値は0 LSBです。

### • 積分非直線性誤差 (INL) - 図17-12.

変位誤差と利得誤差補正後の全ての遷移点で理想遷移点と比べた最大偏差です。理想値は0 LSBです。

### • 微分非直線性誤差 (DNL) - 図17-13.

実際の符号の幅(隣接する2つの遷移点間)で理想符号幅(1 LSB)と比べた最大偏差です。理想値は0 LSBです。

### • 量子化誤差

有限数の符号で入力電圧を量子化するため、1 LSB幅となる入力電圧範囲は同じ値の符号になります。この値は常に $\pm 0.5$  LSBです。

### • 絶対精度

補正しない全ての遷移点で理想遷移点と比べた最大偏差です。これは、変位誤差、利得誤差、差動誤差、非直線誤差の影響の合成です。理想値は $\pm 0.5$  LSBです。

図17-10. 変位(オフセット)誤差

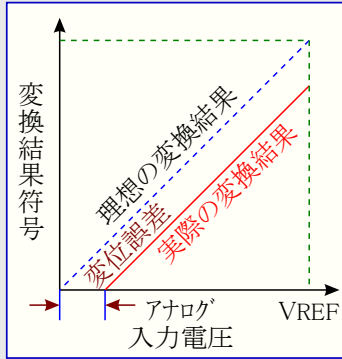


図17-11. 利得誤差

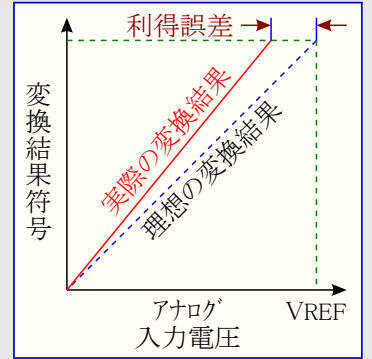


図17-12. 積分非直線性誤差

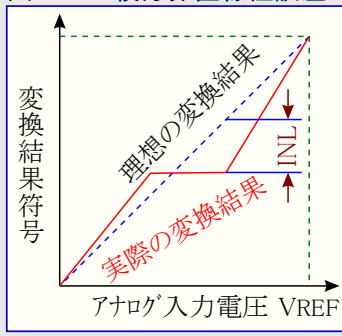
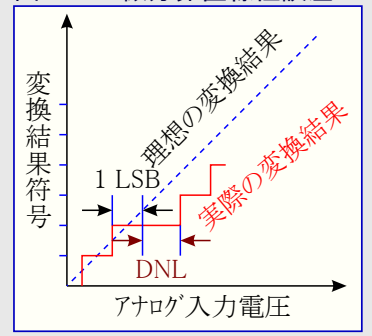


図17-13. 微分非直線性誤差



## 17.11. A/D変換の結果

変換完了(ADIF=1)後、変換結果はA/Dデータレジスタ(ADCH, ADCL)で得られます。

シングルエンド入力変換での結果は右式で示されます。

$$ADC = \frac{V_{IN} \times 1024}{V_{REF}}$$

$V_{IN}$ は選んだ入力ピンの電圧で、 $V_{REF}$ は選んだ基準電圧です(113頁の表17-3と表17-4をご覧ください)。 $\$000$ はアナログGNDを表し、 $\$3FF$ は選んだ基準電圧-1 LSBを表します。

## 17.12. 温度測定

温度測定はシングルエンドADC8チャンネルに連結されるチップ上の温度感知器に基づきます。A/D多重器選択(ADMUX)レジスタのチャンネル選択(MUX3~0)ビットへの'1000'書き込みによるADC8チャンネル選択がこの温度感知器を許可します。温度感知器測定に対する推奨A/D変換器基準電圧源は1.1V内部基準電圧です。温度感知器が許可されると、A/D変換器は温度感知器上の電圧測定を単独変換動作(ADATE=0)で使うことができます。

測定した電圧は表17-2で記述されたように温度に対して直線的関係を持ちます。感度は概ね1 LSB/°Cで、精度は使用者校正に依存します。代表的には、室温校正との仮定で、1点温度校正後の測定精度は $\pm 10^\circ\text{C}$ です。より良い精度は校正に対して2つの温度点を使うことで達せられます。

表17-2に記載した値は代表値です。然しながら製法変化のため、温度感知器出力電圧は或るチップと別のチップで変化します。より正確な結果達成を可能とするために、温度測定は応用ソフトウェアで校正できます。校正ソフトウェアは次式を使って行なえます。

表17-2. 温度対感知器出力電圧 (代表値)

温度(°C)	-40°C	+25°C	+85°C
A/D変換値(LSB)	230	300	370

$$T(\text{温度}) = k \times (\text{ADCH} \ll 8 | \text{ADCL}) - T_{OS}$$

ここでADCH:ADCLはA/D変換器データレジスタ、 $k$ は固定傾斜係数、 $T_{OS}$ は温度感知器変位(オフセット)値です。代表的に係数 $k$ は1.0に大変近く、1点校正ではこの係数を省略できるかもしれません。高精度が要求される場合、傾斜係数は2つ温度での測定を基に評価されるべきです。

### 17.13. A/D変換用レジスタ

#### 17.13.1. ADMUX – A/D多重器選択レジスタ (ADC Multiplexer Select Register)

ビット (\$7C)	7	6	5	4	3	2	1	0	
	-	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R	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7,4 – 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6 – REFS0 : 基準電圧選択 (Reference Selection Bit)

このビットは表17-3.で示されるようにA/D変換器の基準電圧を選びます。このビットが変換中に変更されると、その変更はこの変換が完了する(ADCSRAのADIF=1)まで実施しません。

表17-3. A/D変換部の基準電圧選択

REFS0	基準電圧
0	内部1.1V基準電圧
1	AVCC基準電圧

- ビット5 – ADLAR : 左揃え選択 (ADC Left Adjust Result)

ADLARビットはA/Dデータレジスタ内の変換結果の配置に影響を及ぼします。結果を左揃えにするにはADLARに1を書いてください。さもなければ結果は右揃えです。ADLARビットの変更はどんな進行中の変換にも拘らず、直ちにA/Dデータレジスタの内容に影響を及ぼします。このビットの完全な記述については115頁の「A/Dデータレジスタ」をご覧ください。

- ビット3~0 – MUX3~0 : A/Dチャネル選択 (Analog Channel Select Bits 3~0)

これらのビットの値はA/D変換器に接続されるアナログ入力を選びます。シングルエンドチャネルADC8選択は温度測定を許可します。詳細については表17-4.をご覧ください。これらのビットが変換中に変更される場合、その変更はこの変換が完了する(ADCSRAのADIF=1)まで実施しません。

表17-4. アナログ入力チャネル選択

MUX3~0	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001~1101	1110	1111
アナログ入力チャネル	ADC0	ADC1	ADC2	ADC3	ADC4	ADC5	(ADC6)	(ADC7)	ADC8	(予約)	1.1V	0V
備考	PC0	PC1	PC2	PC3	PC4	PC5	PA0	PA1	温度 <sup>(1)</sup>		V <sub>BG</sub> <sup>(2)</sup>	GND

注1: 温度測定については112頁の「温度測定」をご覧ください。

注2: 内部基準電圧切り替え後、A/D変換器は測定が安定するのに先立って1msの安定時間が必要です。これに先立つ変換開始は信頼できないかもしれません。A/D変換器はこの安定時間中に許可されなければなりません。

(訳注) PDIPとQFN/MLF28外囲器は6チャネルのため、ADC6,7はありません。

#### 17.13.2. ADCSRA – A/D制御/状態レジスタA (ADC Control and Status Register A)

ビット (\$7A)	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
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 – ADEN : A/D許可 (ADC Enable)

このビットに1を書くことがA/D変換部(動作)を許可します。0を書くことによってA/D変換部は(電源が)OFFにされます。変換が進行中にA/D変換部をOFFにすることはその変換を(途中)終了します。

- ビット6 – ADSC : A/D変換開始 (ADC Start Conversion)

単独変換動作で各変換を始めるにはこのビットへ1を書いてください。連続変換動作で最初の変換を始めるにはこのビットへ1を書いてください。A/D変換部が許可される(ADEN=1)と同時にADSCが書かれるか、またはA/D変換部が許可されてしまった後にADSCが書かれた後の初回変換は、通常の13に代わって25変換クロック周期で行います。この初回変換はA/D変換部の初期化を実行します。

ADSCは変換が進行中である限り1として読みます。変換が完了すると0に戻ります。このビットへの0書き込みは無効です。

- ビット5 – ADATE : A/D変換自動起動許可 (ADC Auto Trigger Enable)

このビットが1が書かれると、A/D変換の自動起動が許可されます。A/D変換器は選んだ起動信号の上昇端で変換を開始します。この起動元はA/D変換制御/状態レジスタB(ADCSR)のA/D変換起動要因選択(ADTS2~0)ビット設定によって選ばれます。



•ビット4 – ADIF : A/D変換完了割り込み要求フラグ (ADC Interrupt Flag)

A/D変換が完了してA/Dデータレジスタが更新される時にこのフラグが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(1)ビットとA/D変換完了割り込み許可(ADIF)ビットが設定(1)されていれば、A/D変換完了割り込みが実行されます。対応する割り込み処理ベクタを実行する時にADIF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に論理1を書くことによってもADIFは解除(0)されます。ADCSRAで読み-変更-書き(リード モディファイ ライト)を行うと、保留中の割り込みが禁止され得ることに注意してください。これはSBI,CBI命令が使われる場合にも適用されます(訳注:アドレス範囲外のため、本行は不適切です)。

•ビット3 – ADIE : A/D変換完了割り込み許可 (ADC Interrupt Enable)

このビットが1を書かれ、SREGの全割り込み許可(1)ビットが設定(1)されていると、A/D変換完了割り込みが活性に(許可)されます。このビットが解除(0)されると、この割り込みは禁止されます(訳注:共通性のため本行追加)。

•ビット2~0 – ADPS2~0 : A/D変換クロック選択 (ADC Prescaler Select Bits)

これらのビットはXTAL(システム)周波数とA/D変換部への入力クロック間の分周値を決めます。

表17-5. A/D変換クロック選択 (CK=システム クロック)

ADPS2	0	0	0	0	1	1	1	1
ADPS1	0	0	1	1	0	0	1	1
ADPS0	0	1	0	1	0	1	0	1
A/D変換クロック	CK/2	CK/2	CK/4	CK/8	CK/16	CK/32	CK/64	CK/128

17.13.3. ADCSRB – A/D変換 制御/状態レジスタB (ADC Control and Status Register B)

ビット (\$7B)	7	6	5	4	3	2	1	0	
	-	ACME	-	-	-	ADTS2	ADTS1	ADTS0	ADCSRB
Read/Write	R	R/W	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7,5~3 – Res : 予約 (Reserved Bits)

これら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のため、ADCSRB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2~0 – ADTS2~0 : A/D変換自動起動要因選択 (ADC Auto Trigger Source)

A/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットが1を書かれると、これらのビット値はどの起動元がA/D変換を起動するのかわらを選択します。ADATEが解除(0)されると、ADTS2~0設定は無効です。変換は選んだ割り込みフラグの上昇端によって起動されます。解除(0)されている起動元から設定(1)されている起動元への切り替えが起動信号上に上昇端を生成することに注意してください。ADCSRAのA/D許可(ADEN)ビットが設定(1)されているなら、これが変換を開始させます。連続変換動作(ADTS2~0=0)への切り替えは、例えばA/D変換完了割り込み要求フラグが設定(1)されていても、起動事象を引き起こしません。

表17-6. A/D変換自動起動元選択

ADTS2	ADTS1	ADTS0	起動元
0	0	0	連続変換動作
0	0	1	アナログ比較器
0	1	0	外部割り込み要求0
0	1	1	タイマ/カウンタ0比較A一致
1	0	0	タイマ/カウンタ0溢れ
1	0	1	タイマ/カウンタ1比較B一致
1	1	0	タイマ/カウンタ1溢れ
1	1	1	タイマ/カウンタ1捕獲要求

## 17.13.4. ADCH,ADCL – A/Dデータレジスタ (ADC Data Register)

ADLAR=0時									
ビット (\$79)	15	14	13	12	11	10	9	8	
	-	-	-	-	-	-	ADC9	ADC8	ADCH
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$78)	7	6	5	4	3	2	1	0	
	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ADLAR=1時									
15	14	13	12	11	10	9	8		
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
7	6	5	4	3	2	1	0		
	ADC1	ADC0	-	-	-	-	-	-	ADCL

### • ADC9~0 : A/D変換結果 (ADC Conversion result)

これらのビットは112頁の「A/D変換の結果」で詳述されるように変換での結果を表します。A/D変換が完了すると、その結果がこれら2つのレジスタで得られます。

ADCLが読まれると、A/DデータレジスタはADCHが読まれるまで更新されません。従ってこの結果が左揃えで且つ8ビットを越える精度が必要とされないなら、ADCHを読むことで用が足ります。さもなければADCLが先に、その後にADCHが読まれなければなりません。

A/D多重器選択レジスタ(ADMUX)の左揃え選択(ADLAR)ビットとA/Dチャネル選択(MUX3~0)ビットはこのレジスタから結果を読む方法に影響を及ぼします。ADLARが設定(1)されると結果は左揃えにされます。ADLARが解除(0)されると(既定)、結果は右揃えにされます。

## 17.13.5. DIDR0 – デジタル入力禁止レジスタ0 (Digital Input Disable Register 0)

ビット (\$7E)	7	6	5	4	3	2	1	0	
	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	DIDR0
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 – ADC7D~ADC0D : ADC7~0 デジタル入力禁止 (ADC7~0 Digital Input Disable)

このビットが論理1を書かれると、対応するADCn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。アナログ信号がADCnピンに印加され、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、そのビットは論理1を書かれるべきです。

## 18. デバッグWIRE 内蔵デバッグ システム

### 18.1. 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
- 不揮発性メモリのプログラミング

### 18.2. 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のプログラミングのための単線の双方向インターフェースを使います。

### 18.3. 物理インターフェース

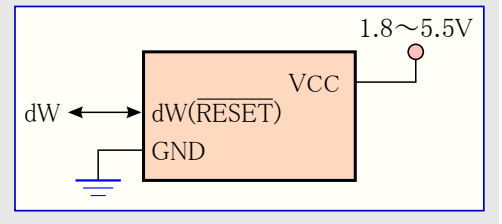
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

図18-1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の設計時、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よりも小さくしてはなりません。この抵抗はデバッグWIRE機能の必要条件ではありません。
- 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18-1. デバッグWIRE構成図



### 18.4. 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時、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。

### 18.5. 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機能を正確に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。

デバッグWIREインターフェースは非同期で、それはデバッグがシステムクロックに同期する必要を意味します。システムクロックがソフトウェア(例えばCLKPSビット書き込み)によって変更されると、デバッグWIRE経由通信は失敗するかもしれません。また、100kHz未満のクロック周波数は通信問題を引き起こすかもしれません。

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。

### 18.6. デバッグWIRE用レジスタ

#### 18.6.1. DWDR - デバッグWIRE データレジスタ (debugWIRE Data Register)

ビット	7	6	5	4	3	2	1	0	
\$31 (\$51)	(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。

## 19. 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命令は既定で禁止ですが、SELPFRGENヒューズを(0)にプログラミングすることで許可にできます。

プログラムメモリはページ単位形式で更新されます。ページ一時緩衝部に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.は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。

### 19.0.1. 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。

### 19.0.2. 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CTPBBビット(=1)書き込みにより、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。

### 19.0.3. 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'を書いて、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。

## 19.1. 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

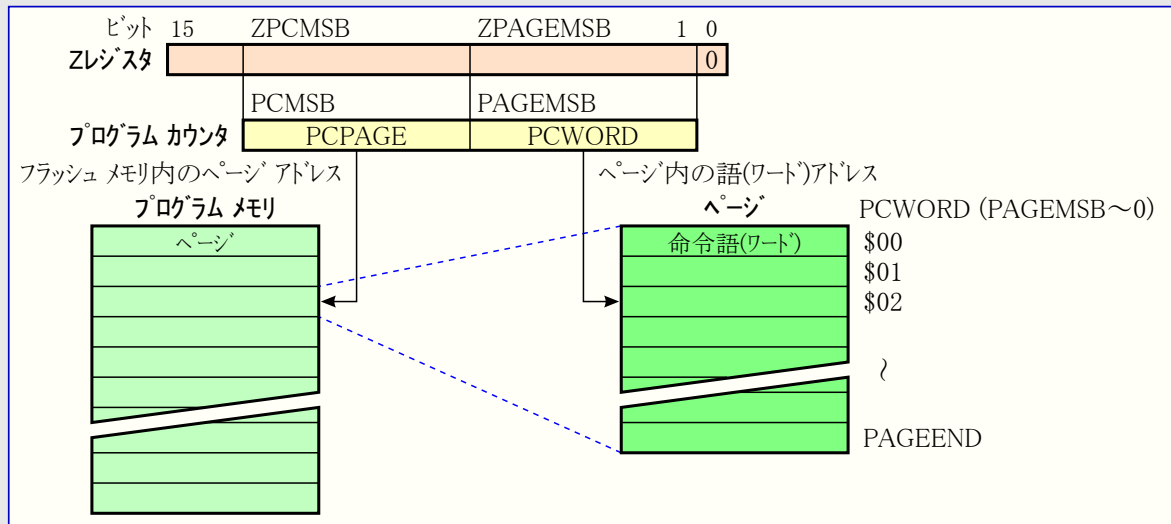
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

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(124頁の「メモリパラメータ」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19-1.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は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19-1. 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124頁の表21-1.で一覧されます。

### 19.1.1. 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書き込み動作中に妨げられます。使用者はEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)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。

### 19.1.2. 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とヒューズ/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

ヒューズ下位バイ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バイト(FLB)の値が転送先レジスタに格納されます。ヒューズ下位バイトの配置と詳細な記述については122頁の表20-5.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、ヒューズ上位バイトを読むには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バイト(FHB)の値が転送先レジスタに格納されます。ヒューズ上位バイトの配置と詳細な記述については122頁の表20-4.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

ヒューズ拡張バイトはZポインタに\$0002を設定することによって読むことができます。SELFPRGENとRFLBビットがSPMCSRで設定(1)された後、3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拡張バイト(FEB)の値が転送先レジスタに格納されます。ヒューズ拡張バイトの配置と詳細な記述については122頁の表20-3.を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	-	-	EFB0

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。

### 19.1.3. 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ー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自身が命令を誤って実行し得ます。

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のを防ぎ、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。

### 19.1.4. 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19-1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19-1. 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、施錠ビット書き込み)	3.7ms	4.5ms

注: 最小と最大の時間は(項目の)個別操作毎に対してです。

## 19.2. 自己プログラミング用レジスタ

### 19.2.1. SPMCSR – SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register)

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)	–	RWWSB	–	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 – 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 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)**

このビットは書き中読み可(Read-While-Write)を支援するデバイスとの互換用です。ATtiny48/88では常に0として読みます。

- **ビット5 – 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ポイントの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118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ポイント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ポイント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ポイント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。この名称重複が存在する他のデバイスと異なり、AVR Studio 4.16でのデバイス定義インクルードファイルではヒューズビットのSELFPRGEN定義が注釈になっています。故にこの名称を使う場合は十分な注意と確認が必要です。

## 20. 施錠ビット、ヒューズビット、デバイス識票

### 20.1. 施錠ビット

ATtiny48/88は表20-1.で一覧にされるプログラムとデータのメモリ施錠ビットを提供します。

表20-1. 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		1 (非プログラム)
-	6		1 (非プログラム)
-	5		1 (非プログラム)
-	4		1 (非プログラム)
-	3		1 (非プログラム)
-	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。

施錠ビットは非プログラム(1)のままか、または表20-2.で一覧される付加機能を得るためにプログラム(0)にすることができます。

表20-2. 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。

注2: 0はプログラム、1は非プログラムを意味します。

施錠ビットはチップ消去指令でのみ1に消去することができます。

ATtiny48/88は独立したブートローダ領域を持ちません。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。

## 20.2. ヒューズ ビット

ATtiny48/88には3つのヒューズ ビットがあります。表20-3.~5.は全てのヒューズの概略機能とヒューズ ビッ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20-3. ヒューズ 拡張バイト一覧

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。 (注1)	1 (非プログラム) 自己プログラミング不許可

注1: SPM命令許可。117頁の「フラッシュの自己プログラミング」をご覧ください。

表20-4. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
RSTDISBL	7	PC6がI/OピンかまたはRESETピンかを選びます。 (注1,2)	1 (非プログラム) PC6はRESETピン
DWEN	6	デバッグWIRE機能許可。 (注2)	1 (非プログラム) デバッグWIRE不許可
SPIEN	5	直列プログラミング許可。 (注3)	0 (プログラム) 直列プログラミング許可
WDTON	4	ウォッチドッグ タイム常時有効。 (注4)	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	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
BODLEVEL2	2		1 (非プログラム)
BODLEVEL1	1	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注5)	1 (非プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: RSTDISBLヒューズの記述については45頁の「ポートCの交換機能」をご覧ください。

注2: このヒューズ ビットのプログラム(0)はRESETピンの機能を変更し、直列インターフェース経由での更なるプログラミングを不可能にします。ヒューズ ビットは並列プログラミング法を使って非プログラム(1)にできます。

注3: 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4: 詳細については30頁の「WDTCSR - ウォッチドッグ タイム制御レジスタ」をご覧ください。

注5: BODLEVELヒューズの符号化については137頁の表22-4.をご覧ください。

表20-5. 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。 (注4)	0 (プログラム) 8分周
CKOUT	6	システム クロック出力許可。 (注3)	1 (非プログラム) 不許可
SUT1	5	起動時間選択。 (注1)	1 (非プログラム)
SUT0	4		0 (プログラム)
-	3		1 (非プログラム)
-	2		1 (非プログラム)
CKSEL1	1	クロック種別選択。 (注2)	1 (非プログラム)
CKSEL0	0		0 (プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については18頁の表6-3.をご覧ください。

注2: CKSEL1,0の既定設定は8MHz校正付き内蔵発振器になります。詳細については17頁の表6-1.をご覧ください。

注3: CKOUTはポートB0に出力するのをシステム クロックに許します。詳細については19頁の「クロック出力緩衝部」をご覧ください。

注4: 詳細については19頁の「システム クロック前置分周器」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。

### 20.2.1. 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。

### 20.3. 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確認する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。ATtiny48/88用の識票バイトは表20-6.で与えられます。

表20-6. デバイスの識別番号(ID)

部品番号	識票バイトアドレス		
	\$0000	\$0001	\$0002
ATtiny48	\$1E	\$92	\$09
ATtiny88	\$1E	\$93	\$11

### 20.4. 校正バイト

ATtiny48/88は内蔵発振器用に1バイトの校正値を持ちます。このバイトは識票アドレス空間でアドレス\$0000の上位バイトにあります。リセット中、[校正付き内蔵発振器](#)の正しい周波数を保証するために、このバイトが[発振校正\(OSCCAL\)レジスタ](#)へ自動的に書かれます。

## 21. 外部プログラミング

本章はATtiny48/88のフラッシュメモリ、EEPROM、**施錠ビット**、**ヒューズビット**のプログラミングと照合の方法を記述します。

### 21.1. メモリパラメータ

フラッシュメモリのパラメータは下の表21-1.で要約されます。

表21-1. フラッシュメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
ATtiny48	2K語 (4Kバイト)	32語	PC4~0	64	PC10~5	10
ATtiny88	4K語 (8Kバイト)	32語	PC4~0	128	PC11~5	11

EEPROMのパラメータは下の表21-2.で要約されます。

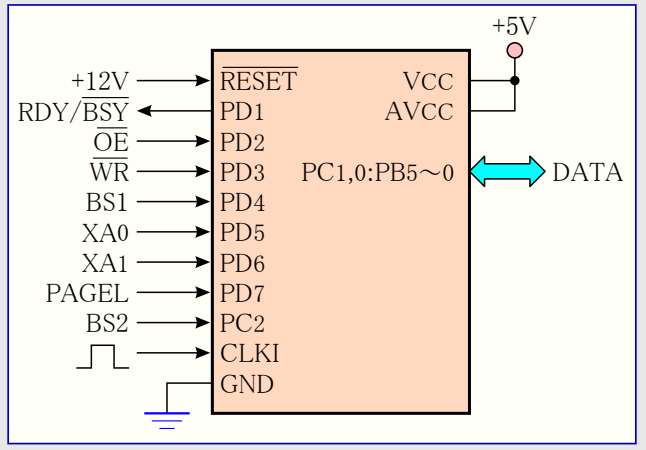
表21-2. EEPROMメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
ATtiny48	64バイト	4バイト	EEA1~0	16	EEA5~2	5
ATtiny88	64バイト	4バイト	EEA1~0	16	EEA5~2	5

### 21.2. 並列プログラミング

並列プログラミング信号と接続は右の図21-1.で図解されます。

図21-1. 並列プログラミング構成図



信号は下の表21-3.で記述されます。下表で示されないピンはピン名によって参照されます。

表21-3. 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)
BS1	PD4	入力	上位/下位バイト選択1 (0:下位, 1:上位)      (一般用)
XA0	PD5	入力	CLKI動作ビット0
XA1	PD6	入力	CLKI動作ビット1
PAGEL	PD7	入力	プログラムメモリとEEPROMデータをページ一時緩衝部に設定
BS2	PC2	入力	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	PC1,0:PB5~0	入出力	双方向データバス (OE=Low時出力)

注: VCC-0.3V<AVCC<VCC+0.3Vですが、AVCCは常に4.5~5.5V内にすべきです。特記事項を除いて、パルス幅は最低250nsと仮定されます。

表21-4. プログラミング動作移行時のピン値

ピン名	シンボル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

XA0とXA1ピンはCLKI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21-5.で示されます。

表21-5. XA0とXA1の符号化(機能)

XA1	XA0	CLKIパルス時の動作
0	0	フラッシュまたはEEPROMのアドレス設定 (上位/下位はBS1で指定)
0	1	データ設定 (フラッシュの上位/下位はBS1で指定)
1	0	指令設定
1	1	アイドル (動作なし)

WRまたはOEパルスを送るとき、設定した指令が実行される動作を決めます。各種指令は表21-6.で示されます。

表21-6. 指令バイトのビット符号化

指令バイト	指令の機能	指令バイト	指令の機能
\$80 (1000 0000)	チップ消去	\$08 (0000 1000)	識別バイト、校正バイト読み出し
\$40 (0100 0000)	ヒューズビット書き込み	\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$20 (0010 0000)	施錠ビット書き込み	\$02 (0000 0010)	フラッシュメモリ読み出し
\$10 (0001 0000)	フラッシュメモリ書き込み	\$03 (0000 0011)	EEPROM読み出し
\$11 (0001 0001)	EEPROM書き込み		

### 21.2.1. 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、RESETピンを0V、124頁の表21-4.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。それから20μs以内にVCCが最低1.8Vに達することを保証してください。
- ③ 20~60μs待ち、RESETに11.5~12.5Vを印加します。
- ④ Prog\_enable識別のラッチを確実にするため、高電圧が印加されてしまった後、最低10μs、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少なくとも300μs間待ちます。
- ⑥ デバイスの電源を落とすか、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、RESETピンを0V、124頁の表21-4.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、REESTに11.5~12.5Vを印加します。
- ④ Prog\_enable識別のラッチを確実にするため、高電圧が印加されてしまった後、最低10μs、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、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。

### 21.2.2. 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。この考慮は識別バイト読み出しにも適用されます。

### 21.2.3. チップ消去

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。チップ消去はフラッシュメモリ、EEPROM、施錠ビットの全てを消去します。EESAVEヒューズがプログラム(0)されている場合、EEPROMは消去されません。

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」指令は以下のように設定されます。

- 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 CLKI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 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/BSYがLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/BSYがHigh(1)になるまで待ちます。



#### 21.2.4. フラッシュ メモリ書き込み (次頁の図21-3.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(124頁の表21-1.参照)。フラッシュ メモリへ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 CLKI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 CLKI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 CLKI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 CLKI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. 語(ワード)データをページ一時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイトを選びます。
- ② PAGELに正パルスを与えます。これは語データをページ一時緩衝部にラッチ(設定)します。

##### F. 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~E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21-2.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。

##### G. 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~\$07/\$0F)を設定します。
- ④ CLKI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。

##### H. ページ書き込み

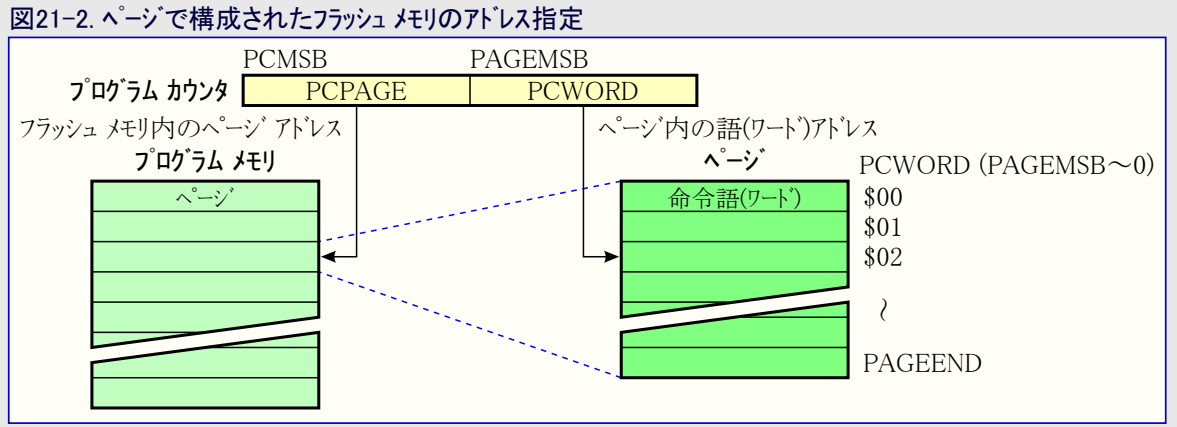
- ① BS1をLow(0)に設定します。
- ②  $\overline{WR}$ に負パルスを与えます。これはデータのページ全体の書き込みを開始します。 $\overline{RDY}/\overline{BSY}$ がLow(0)になります。
- ③  $\overline{RDY}/\overline{BSY}$ がHigh(1)になるまで待ちます。

##### I. 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~Hを繰り返し

##### J. ページ書き込み終了

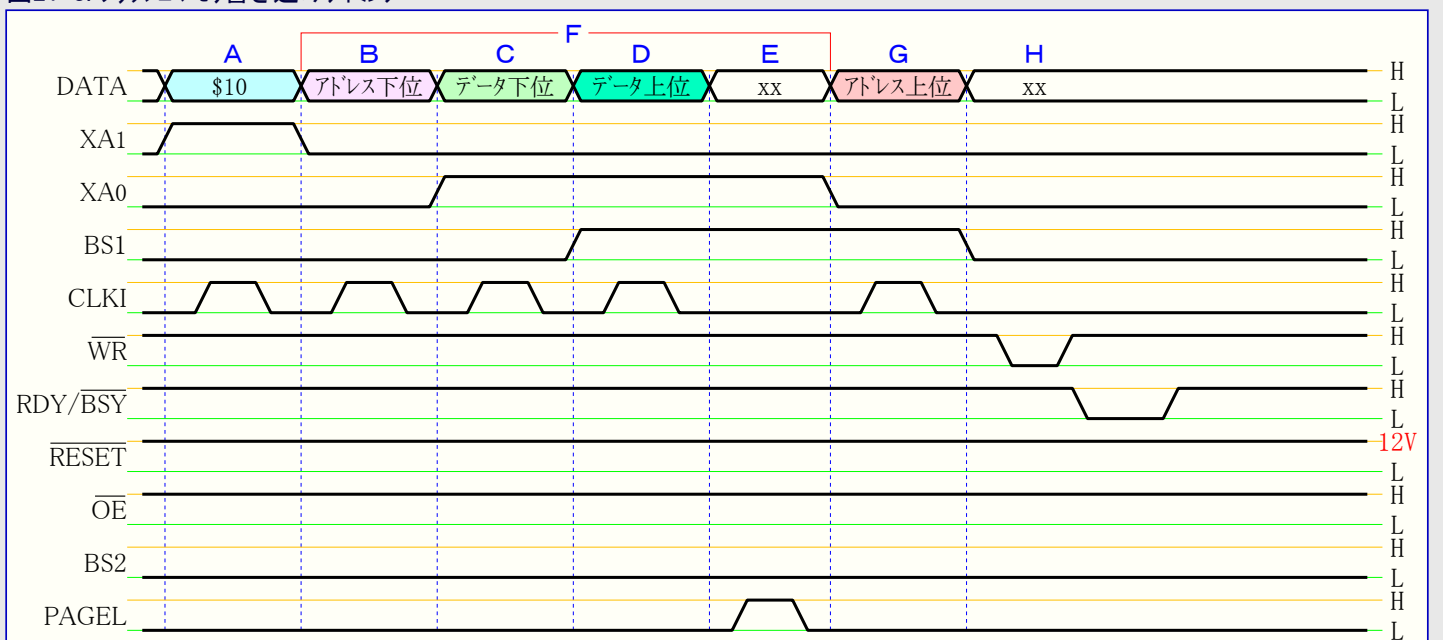
- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ CLKIに正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

フラッシュ ページのアドレス指定は下の図21-2. で図解されます。使われるシンボルは124頁の表21-1. で記述されます。



フラッシュプログラミング波形は図21-3. で図解され、そこでのxxは“無関係”を意味し、文字は先に記述されたプログラミング段階を参照します。

**図21-3. フラッシュメモリ書き込みタイミング**



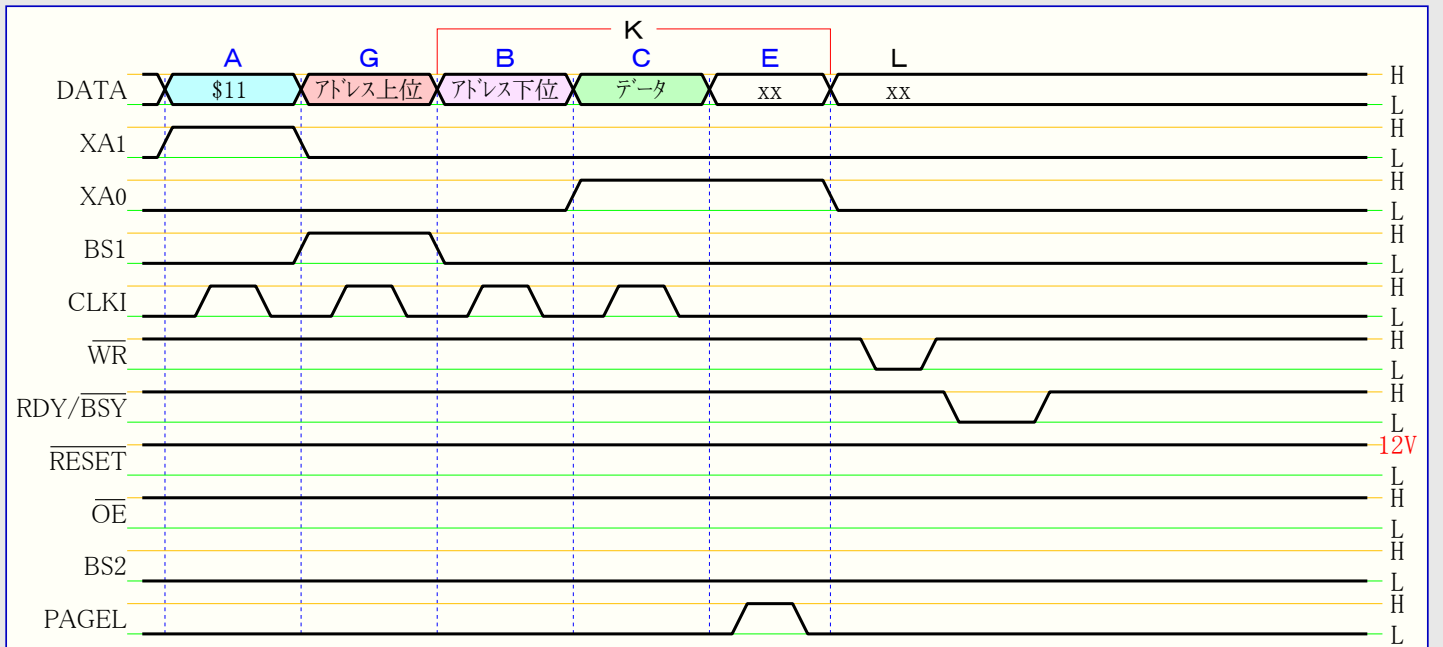
### 21.2.5. EEPROM書き込み

EEPROMはページで構成されます(124頁の表21-2参照)。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。データ用EEPROM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126頁の「フラッシュメモリの書き込み」を参照。図21-4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. アドレス上位バイト(\$00)を設定します。(「フラッシュメモリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$3F)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. 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)
5. データをラッチします(PAGELに正パルスを与えます)。(「フラッシュメモリ書き込み」のEを参照)
- K. 緩衝部全体が満たされるまで3~5を繰り返します。
- L. EEPROMページ書き込み
  - ① BS1をLow(0)に設定します。
  - ②  $\overline{WR}$ 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/ $\overline{BSY}$ がLow(0)になります。
  - ③ 次のページを書く前に、RDY/ $\overline{BSY}$ がHigh(1)になるまで待ちます。

EEPROMプログラミング波形は図21-4.で図解され、そこでのxxは“無関係”を意味し、文字は先に記述されたプログラミング段階を参照します。

図21-4. EEPROM書き込みタイミング



## 21.2.6. フラッシュ メリ読み出し

フラッシュ メ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126頁の「フラッシュ メリの書き込み」を参照)

1. フラッシュ メリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$07/\$0F)を設定します。(「フラッシュ メリ書き込み」のG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 メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。フラッシュ メリ語(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 メリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. OEをHigh(1)に設定します。DATAはHi-Zになります。

## 21.2.7. 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126頁の「フラッシュ メリの書き込み」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 メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$3F)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、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。
5. OEをHigh(1)に設定します。DATAはHi-Zになります。

## 21.2.8. 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126頁の「フラッシュ メリの書き込み」を参照)

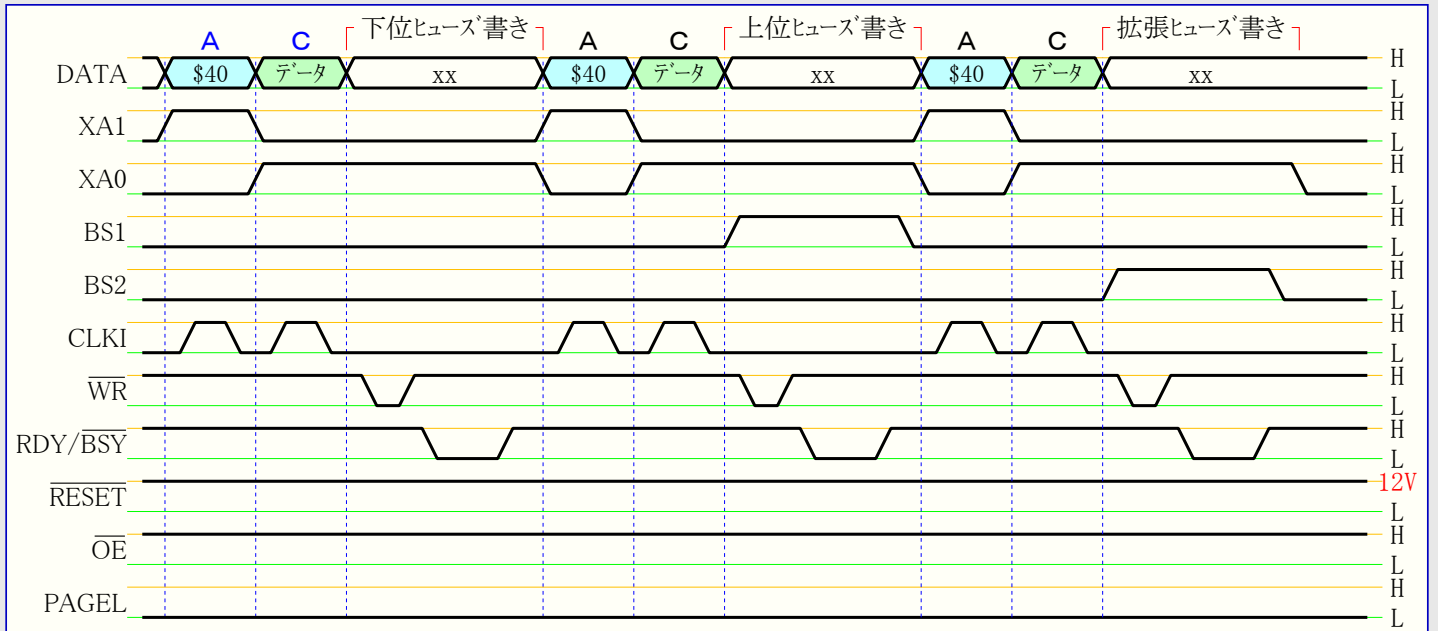
1. ヒューズ ビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュ メリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4. WRに負ハルスを与え、RDY/BSYがHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選びます。

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)

ヒューズのプログラミング波形は図21-5.で図解され、そこでのxxは“無関係”を意味し、文字は先に記述されたプログラミング段階を参照します。

図21-5. ヒューズ書き込みタイミング



### 21.2.9. 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126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。

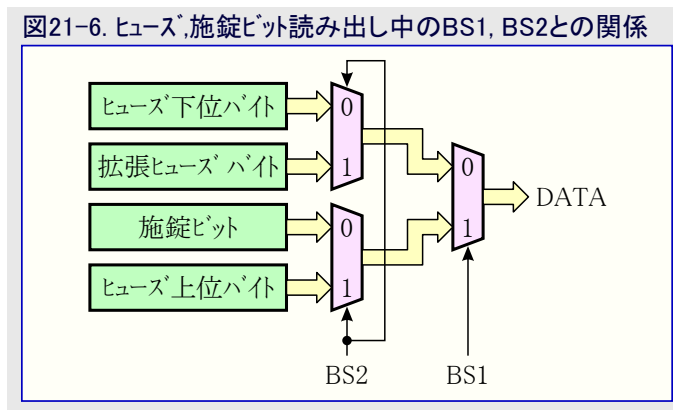
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

### 21.2.10. 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126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。

ヒューズと施錠ビットの割り当ては下の図21-6.で図解されます。



### 21.2.11. 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126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。

### 21.2.12. 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126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。

### 21.3. 直列プログラミング

フラッシュメモリとEEPROMの両方は $\overline{\text{RESET}}$ 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。 $\overline{\text{RESET}}$ 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。

直列プログラミング信号と接続は右の図21-7.で図解されます。ピン割り当ては表21-7.で図解されます。

EEPROMをプログラミングする時に自動消去期間が自動書き込み動作内に組み入れられるので(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せん。

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。

$f_{\text{CK}} < 12\text{MHz}$  : Low区間 > 2 CPUクロック周期      $f_{\text{CK}} < 12\text{MHz}$  : High区間 > 2 CPUクロック周期  
 $f_{\text{CK}} \geq 12\text{MHz}$  : Low区間 > 3 CPUクロック周期      $f_{\text{CK}} \geq 12\text{MHz}$  : High区間 > 3 CPUクロック周期

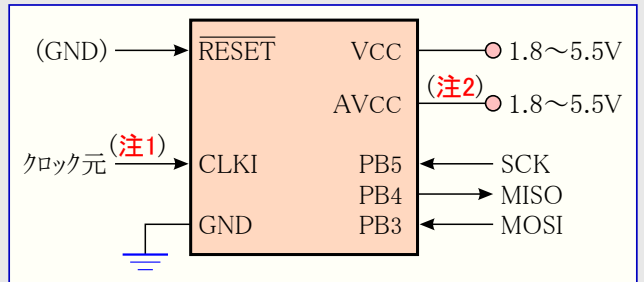
#### 21.3.1. ピン配置

ピン割り当ては表21-7.で一覧にされます。

表21-7. 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	PB3	入力	直列データ入力
MISO	PB4	出力	直列データ出力
SCK	PB5	入力	直列クロック

図21-7. 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、CLKIピンにクロック元を接続する必要はありません。

注2:  $V_{\text{CC}} - 0.3\text{V} < AV_{\text{CC}} < V_{\text{CC}} + 0.3\text{V}$ ですが、AVCCは常に1.8~5.5V内にすべきです。

#### 21.3.2. プログラミング手順

ATtiny48/88への直列データ書き込み時、データはSCKの上昇端でクロック駆動されます。ATtiny48/88からのデータ読み込み時、データはSCKの下降端でクロック駆動されます。タイミングの詳細については143頁の図22-9.と図22-10.をご覧ください。

直列プログラミング動作でのATtiny48/88のプログラミングと照合は次手順が推奨されます(表21-8.の命令一式参照)。

- 電源投入手順:  $\overline{\text{RESET}}$ とSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加してください。
  - 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、 $\overline{\text{RESET}}$ は正パルスを与えられなければなりません。パルスの幅は最低 $t_{\text{RST}} + 2$  CPUクロック周期でなければなりません。 $\overline{\text{RESET}}$ ピンの最小パルス幅の定義については137頁の表22-3.をご覧ください。
- 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。
- 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、 $\overline{\text{RESET}}$ 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。
- 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124頁の表21-1.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5/5+1ビットとデータを提供することによって1バイトづつ設定されます。
  - 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6/7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( $\overline{\text{BSY}}/\overline{\text{RDY}}$ )が使われない場合、使用者は次のページを行う前に最低 $t_{\text{WD\_FLASH}}$ (表21-9.参照)待たねばなりません。フラッシュ書き込み操作完了前の(ポーリング以外の)直列プログラミングインターフェースでのアクセスは不正な書き込み結果になり得ます。
- 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提供することによって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( $\overline{\text{BSY}}/\overline{\text{RDY}}$ )が使われない場合、次のバイトを行う前に最低 $t_{\text{WD\_EEPROM}}$ (表21-9.参照)待たねばなりません。

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4/4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、次ページ(表21-2.参照)を行う前に最低tWD\_EEPROM(表21-9.参照)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。

- 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。
- 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。
- 電源OFF手順(必要とされるならば): RESETをHigh(1)に設定してください。VCC電源をOFFにしてください。

### 21.3.3. 直列プログラミング命令一式

表21-8.と次頁の図21-8.は命令一式を記述します。

表21-8. 直列プログラミング命令一式

命令	命令形式				備考
	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	\$AC	\$53	\$00	\$00	
チップ消去	\$AC	\$80	\$00	\$00	
多忙/準備可検査	\$F0	\$00	\$00	状態値	最下位ビットが多忙フラグ。
<b>設定系命令</b>					
拡張アドレス設定 (注1)	\$4D	\$00	拡張アドレス	\$00	
フラッシュ ページ内上位バイト設定	\$48	アドレス上位	アドレス下位	上位バイト	
フラッシュ ページ内下位バイト設定	\$40	アドレス上位	アドレス下位	下位バイト	
EEPROMページ内バイト設定	\$C1	\$00	ページ内位置	バイト	注:ページ内指示以外のビットは0。
<b>読み出し命令</b>					
フラッシュ メモリ上位バイト読み出し	\$28	アドレス上位	アドレス下位	上位バイト	
フラッシュ メモリ下位バイト読み出し	\$20	アドレス上位	アドレス下位	下位バイト	
EEPROM読み出し	\$A0	アドレス上位	アドレス下位	バイト	
施錠ビット読み出し	\$58	\$00	\$00	施錠ビット値	
識票バイト読み出し	\$30	\$00	アドレス	識票バイト	
ヒューズ下位ビット読み出し	\$50	\$00	\$00	ヒューズ下位	
ヒューズ上位ビット読み出し	\$58	\$08	\$00	ヒューズ上位	
ヒューズ拡張ビット読み出し	\$50	\$08	\$00	ヒューズ拡張	
校正バイト読み出し	\$38	\$00	\$00	校正バイト	
<b>書き込み命令</b>					
フラッシュ ページ書き込み	\$4C	アドレス上位	アドレス下位	\$00	
EEPROMバイト書き込み	\$C0	アドレス上位	アドレス下位	バイト	
EEPROMページ書き込み	\$C2	アドレス上位	アドレス下位	\$00	注:ページ指示以外のアドレスビットは0。
施錠ビット書き込み	\$AC	\$E0	\$00	施錠ビット値	
ヒューズ下位ビット書き込み	\$AC	\$A0	\$00	ヒューズ下位	
ヒューズ上位ビット書き込み	\$AC	\$A8	\$00	ヒューズ上位	
ヒューズ拡張ビット書き込み	\$AC	\$A4	\$00	ヒューズ拡張	

注1: 全命令が全デバイスで利用可能な訳ではありません。

注: 施錠ビットとヒューズ値はプログラムが0、非プログラムが1です。将来との互換性のため、未使用のヒューズと施錠ビットは非プログラム(1)にすべきです。

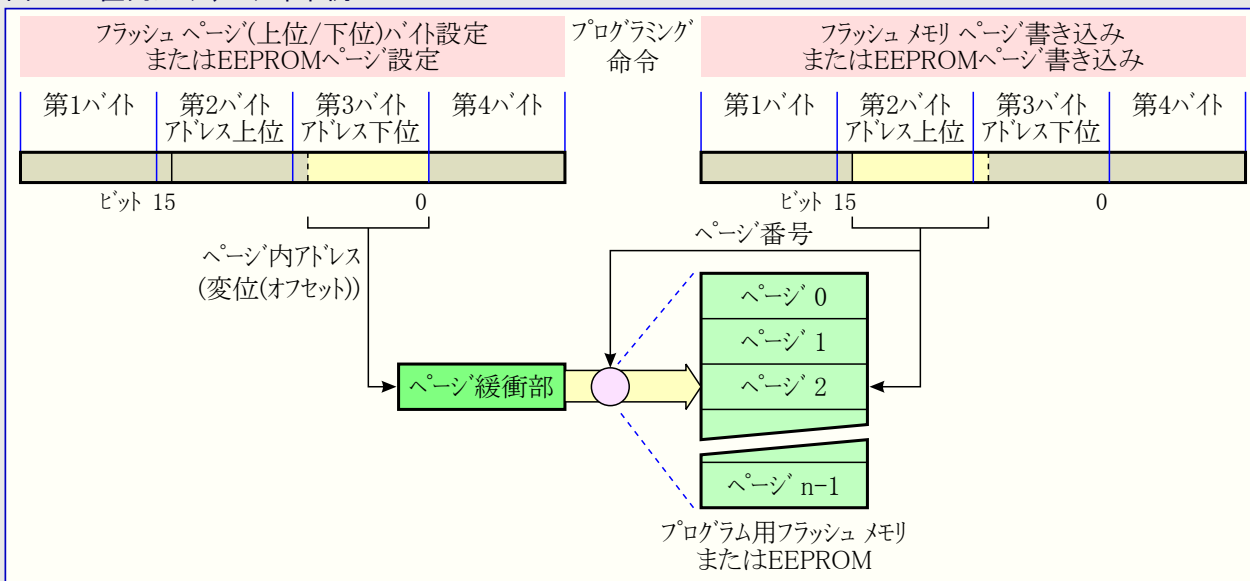
- ヒューズ、施錠ビット、識票バイト、校正バイト、ページ容量については対応項を参照してください。
- プログラミングと書き込み器に関する応用記述については <http://www.atmel.com/avr> をご覧ください。
- 第4バイトの赤背景はホスト読み込み(デバイス出力)を示します。

多忙/準備可検査バイト データ出力のLSBが1なら、プログラミング操作が未だ保留(動作中)です。次の命令が実行される前に本ビットが0に戻るまで待ってください。

同じページ内で、下位バイト データは上位バイト データに先行して格納されなければなりません。

データがページ緩衝部に格納された後にEEPROMページをプログラムしてください。次頁の図21-8.をご覧ください。

図21-8. 直列プログラミング命令例



(訳補) フラッシュメモリ、EEPROM、ページ緩衝部内のアドレス(位置)指定に使われるビットはメモリ容量とページ構成に依存します。ATtiny48/88でのこれらの指定方法は次表で要約されます。

表B. アドレス(第2,3バイト)指定法

命令	第2バイト	第3バイト	備考
拡張アドレス設定			該当命令なし
フラッシュページ内バイト設定	0000 0000	000L LLLL	ATtiny48/88 : L=PC4~0
EEPROMページ内バイト設定	0000 0000	0000 00LL	ATtiny48/88 : L=EEA1~0
フラッシュメモリ読み出し	0000 0HHH 0000 HHHH	LLLL LLLL	ATtiny48 : H=PC10~8,L=PC7~0 ATtiny88 : H=PC11~8,L=PC7~0
EEPROM読み出し	0000 0000	00LL LLLL	ATtiny48/88 : L=EEA5~0
フラッシュページ書き込み	0000 0HHH 0000 HHHH	LLL0 0000 LLL0 0000	ATtiny48 : H=PC10~8,L=PC7~5 ATtiny88 : H=PC11~8,L=PC7~5
EEPROMバイト書き込み	0000 0000	00LL LLLL	ATtiny48/88 : L=EEA5~0
EEPROMページ書き込み	0000 0000	00LL LL00	ATtiny48/88 : L=EEA5~2

### 21.4. フラッシュとEEPROM用のプログラミング時間

フラッシュメモリとEEPROMの待ち時間は表21-9.で一覧にされます。

表21-9. ヒューズ、フラッシュ、EEPROMの次位置書き込み前の待ち時間

シンボル	最低待機時間	備考
tWD_FLASH	4.5ms	フラッシュメモリ(含むヒューズ)書き込み
tWD_EEPROM	3.6ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

## 22. 電気的特性

### 22.1. 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。

### 22.2. 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
V <sub>IL</sub>	Lowレベル入力電圧 (RESETピンを除く)	VCC=1.8~2.4V	-0.5		0.2VCC (注2)	V
		VCC=2.4~5.5V	-0.5		0.3VCC (注2)	
	Lowレベル入力電圧 (RESETピン) (注3)	VCC=1.8~5.5V	-0.5		0.2VCC (注2)	
V <sub>IH</sub>	Highレベル入力電圧 (RESETピンを除く)	VCC=1.8~2.4V	0.7VCC (注4)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注4)		VCC+0.5	
	Highレベル入力電圧 (RESETピン) (注3)	VCC=1.8~5.5V	0.9VCC (注4)		VCC+0.5	
V <sub>OL</sub>	Lレベル出力電圧 (高吸い込みI/Oピンと RESETピンを除く) (注5,6)	IOL=10mA, VCC=5V			0.7	
		IOL=5mA, VCC=3V			0.5	
		IOL=2mA, VCC=1.8V			0.4	
	Lレベル出力電圧 (高吸い込みI/Oピン) (注5,7)	IOL=20mA, VCC=5V				0.7
		IOL=10mA, VCC=3V				0.5
		IOL=4mA, VCC=1.8V				0.4
Lレベル出力電圧 (I/O(PC6)としての RESETピン) (注3,5,6)	IOL=2mA, VCC=5V				0.7	
	IOL=1mA, VCC=3V				0.5	
	IOL=0.4mA, VCC=1.8V				0.4	
V <sub>OH</sub>	Hレベル出力電圧 (高吐き出しI/Oピンと RESETピンを除く) (注6,8)	IOH=-10mA, VCC=5V	4.3			
		IOH=-5mA, VCC=3V	2.5			
		IOH=-2mA, VCC=1.8V	1.4			
	Hレベル出力電圧 (高吐き出しI/Oピン) (注7,8)	IOH=-10mA, VCC=5V	4.3			
		IOH=-5mA, VCC=3V	2.5			
		IOH=-2mA, VCC=1.8V	1.4			
I <sub>IL</sub>	I/OピンLowレベル入力漏れ電流	VCC=5.5V		<0.05	1 (注9)	μA
I <sub>IH</sub>	I/OピンHighレベル入力漏れ電流	確実なH/L範囲		<0.05	1 (注9)	
R <sub>PU</sub>	I/Oピンプルアップ抵抗	VCC=5.5V, Low入力	20		50	kΩ
	RESETピンプルアップ抵抗		30		60	
I <sub>CC</sub>	活動動作消費電流 (注10)	VCC=2V, 1MHz		0.2	0.4	mA
		VCC=3V, 4MHz		1.4	2.5	
		VCC=5V, 8MHz		4.5	8	
	アイドル動作消費電流 (注10)	VCC=2V, 1MHz		0.03	0.1	
		VCC=3V, 4MHz		0.25	0.6	
		VCC=5V, 8MHz		1	2	
パワーダウン動作消費電流 (注11)	VCC=3V, WDT有効		4	10	μA	
	VCC=3V, WDT禁止		<0.2	2		

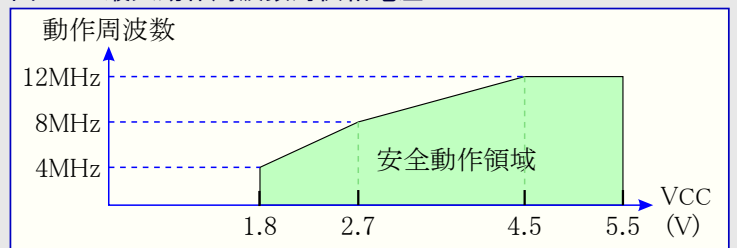
(注1)~(注11)は次頁へ

- 注1:** 25°Cでの代表値です。
- 注2:** “最大”はLowレベルの認識が保証される最高電圧を意味します。
- 注3:** これらのパラメータは製造で検査されていません。
- 注4:** “最小”はHighレベルの認識が保証される最低電圧を意味します。
- 注5:** 各I/Oポートが安定状態(非過渡時)下で検査条件(VCC=5Vで10mA,VCC=3Vで5mA,VCC=2Vで2mA)よりも多く吸い込めるとは言え、以下が厳守されなければなりません。
- ポートA3,0、B7~0、C7、D7~5に対する全IOLの合計が100mAを超えるべきではありません。
  - ポートA2,1、C6~0、D4~0に対する全IOLの合計が100mA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は関連する仕様書での値を超えるかもしれません。ピンは記載した検査条件よりも大きな電流を吸い込むことを保証されません。
- 注6:** RESETピンは(高電圧)プログラミング動作に移行して操作する時に高電圧を許容しなければならず、その結果として通常のI/Oピンと比べると弱い駆動能力を持ちます。(155頁から始まる)図23-34~39をご覧ください。
- 注7:** 高吐き出しI/OピンはPD0,PD1,PD2,PD3です。
- 注8:** 各I/Oポートが安定状態(非過渡時)下で検査条件(VCC=5Vで10mA,VCC=3Vで5mA,VCC=2Vで2mA)よりも多く吐き出せるとは言え、以下が厳守されなければなりません。
- ポートA3,2、B7~0、C6、D7~0に対する全IOHの合計が100mAを超えるべきではありません。
  - ポートA1,0、C7,5~0に対する全IOHの合計が100mA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は関連する仕様書での値を超えるかもしれません。ピンは記載した検査条件よりも大きな電流を吐き出すことを保証されません。
- 注9:** これらは検査環境の漏れ電流に対する評価の検査限界です。実際のデバイスの漏れ電流はより少量です。
- 注10:** 22頁の「消費電力の最小化」で記述された方法を用いた外部クロックでの値です。電力削減が許可(PRR=\$FF)され、I/Oの駆動はありません。
- 注11:** 低電圧検出(BOD)禁止での測定です。

### 22.3. 速度

デバイスの最大動作周波数は供給電圧(VCC)に依存します。供給電圧と最大動作周波数の関係は図22-1.で示されるように区分的線形です。最高周波数対VCC曲線は1.8~4.5V間で直線です。

図22-1. 最大動作周波数対供給電圧



## 22.4. クロック特性

### 22.4.1. 校正付き内蔵発振器精度

工場既定校正よりも高い精度に内蔵発振器を手動校正することが可能です。この発振器周波数が温度と電圧に依存することに注意してください。電圧と温度の特性は162頁の図23-55と図23-56と182頁の図23-112と図23-113.で得られます。

表22-1. 校正付き内蔵発振器の校正精度

校正種別	周波数	VCC	温度	校正精度 (注1)
工場校正	8.0MHz	3V	25°C	±10%
使用者校正	7.3~8.1MHz内の固定周波数	1.8~5.5V内の固定電圧	-40~85°C内の固定温度	±1%

注1: 校正点での発振器周波数精度(固定温度と固定電圧)

### 22.4.2. 外部クロック特性

図22-2. 外部クロック駆動波形

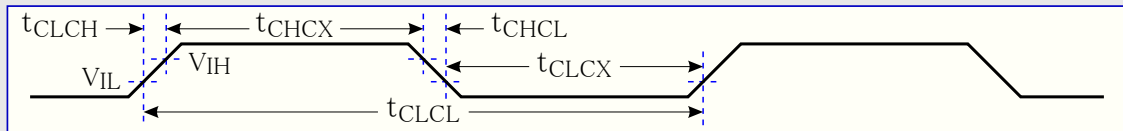


表22-2. 外部クロック特性

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	2	0	6	0	12	MHz
$t_{CLCL}$	クロック周期	500		166		83		ns
$t_{CHCX}$	Highレベル時間	240		80		40		
$t_{CLCX}$	Lowレベル時間	240		80		40		
$t_{CLCH}$	上昇時間		2.0		1.6		0.5	μs
$t_{CHCL}$	下降時間		2.0		1.6		0.5	
$\Delta t_{CLCL}$	隣接クロック周期間の変化率		2		2		2	%

注: 詳細については18頁の「外部クロック信号」を参照してください。

22.5. システムリセットの特性

表22-3. 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		1.5		V
	下降時電源ONリセット閾値電圧 (注1)			1.2		
VRST	RESETピン閾値電圧		0.2VCC		0.9VCC	V
tRST	RESETピンでの最小パルス幅	VCC=1.8V	2			μs
		VCC=3V	0.7			
		VCC=5V	0.4			
VHYST	低電圧検出(BOD)ヒステシス電圧			50		mV
tBOD	最小低電圧検出(BOD)時間			2		μs
VBG	基準電圧	VCC=5V TA=25°C	1.0	1.1	1.2	V
tBG	起動時間			40	70	μs
IBG	消費電流				15	μA

注1: 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。

注: 値は指針の意味だけで、実際の値はTBDです。

表22-4. BODLEVELヒューズ (VBOT) 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	1.7	1.8	2.0	V
1 0 1	2.5	2.7	2.9	
1 0 0	4.1	4.3	4.5	
0 x x	(予約)			

注1: いくつかのデバイスでVBOT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=VBOTに落として検査されます。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。

22.6. アナログ比較器特性

表22-5. アナログ比較器特性 (TA=-40°C~85°C)

シンボル	項目	条件	最小	代表	最大	単位
VAIO	入力変位(オフセット)電圧	VCC=5V, Vin=VCC/2		<10	40	mV
ILAC	入力漏れ電流	VCC=5V, Vin=VCC/2	-50		50	nA
tAPD	アナログ伝播遅延 (飽和から僅かな過駆動へ)	VCC=2.7V		750		ns
		VCC=4.0V		500		
	アナログ伝播遅延 (大きな段階変化)	VCC=2.7V		100		
		VCC=4.0V		75		
tDPD	デジタル伝播遅延	VCC=1.8~5.5V		1	2	clkI/O

注: 全てのパラメータはシミュレーション結果に基づいており、製造に於いて検査されません。

## 22.7. A/D変換器特性

表22-6. A/D変換特性 (TA=-40~85°C)

シンボル	項目	条件	最小	代表	最大	単位
	分解能				10	ビット
	絶対精度 (積分非直線性誤差、 微分非直線性誤差、 量子化誤差、利得誤差、 変位(オフセット)誤差を含む)	VCC=4V VREF=4V	変換クロック=200kHz	2		LSB
			変換クロック=1MHz	3		
			変換クロック=200kHz 雑音低減動作	1.5		
			変換クロック=1MHz 雑音低減動作	2.5		
	積分非直線性誤差	VCC=4V、VREF=4V		1		
	微分非直線性誤差			0.5		
	利得誤差		変換クロック=200kHz	2.5		
	変位(オフセット)(ゼロ)誤差			1.5		
	変換時間	連続変換動作	13		260	μs
	変換クロック周波数		0.05		1	MHz
AVCC	アナログ供給電圧		VCC-0.3		VCC+0.3	V
VREF	基準電圧		VINT		AVCC	
VIN	入力電圧		GND		VREF	
	入力周波数帯域			38.5		kHz
VINT	内蔵基準電圧		1.0	1.1	1.2	V
RREF	基準電圧入力インピーダンス			32		kΩ
RAIN	アナログ入力インピーダンス			100		MΩ

注: 絶対最小/最大AVCCは1.8/5.5Vです。

### 22.8. 2線直列インターフェース特性

以下のデータはシミュレーションと特性付けに基づきます。表22-7.で一覧にされたパラメータは製品で検査されません。シンボルは図22-3.を参照してください。

図22-3. 2線直列バス タイミング

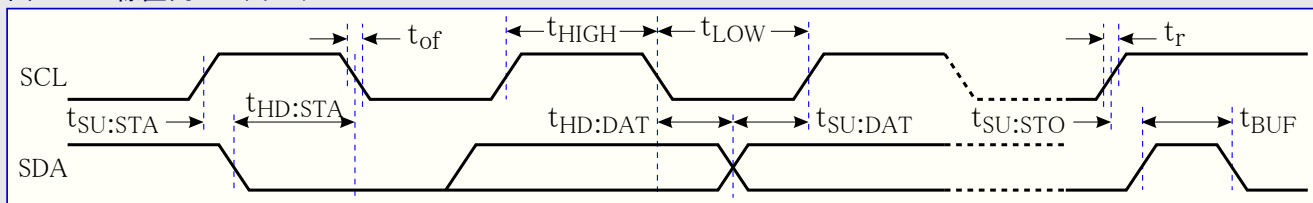


表22-7. 2線直列インターフェース必要条件

シンボル	項目	条件	最小	最大	単位
V <sub>IL</sub>	Lowレベル入力電圧		-0.5	0.3V <sub>CC</sub>	V
V <sub>IH</sub>	Highレベル入力電圧		0.7V <sub>CC</sub>	V <sub>CC</sub> +0.5	
V <sub>HYS</sub>	シュミットトリガ入力ヒステリシス電圧	V <sub>CC</sub> ≥ 2.7V V <sub>CC</sub> < 2.7V	0.05V <sub>CC</sub> 0	- -	
V <sub>OL</sub>	Lowレベル出力電圧	I <sub>OL</sub> =3mA	0	0.4	
t <sub>SP</sub>	入力パルス最小幅(尖頭除去濾波)		0	50	ns
f <sub>SCL</sub>	SCLクロック周波数 (注1)	f <sub>CK</sub> > max(16f <sub>SCL</sub> , 250kHz)	0	400	kHz
t <sub>HD:STA</sub>	(再送)開始条件保持時間		0.6	-	μs
t <sub>LOW</sub>	SCLクロックLowレベル時間		1.3	-	
t <sub>HIGH</sub>	SCLクロックHighレベル時間		0.6	-	
t <sub>SU:STA</sub>	再送開始条件準備時間		0.6	-	
t <sub>HD:DAT</sub>	データ保持時間		0	0.9	
t <sub>SU:DAT</sub>	データ準備時間		100	-	ns
t <sub>SU:STO</sub>	停止条件準備時間		0.6	-	
t <sub>BUF</sub>	停止条件→開始条件間バス開放時間		1.3	-	μs

注1: f<sub>CK</sub>はCPU(システム)クロック周波数です。

## 22.9. SPIタイミング特性

図22-4. SPI タイミング必要条件 (主装置動作)

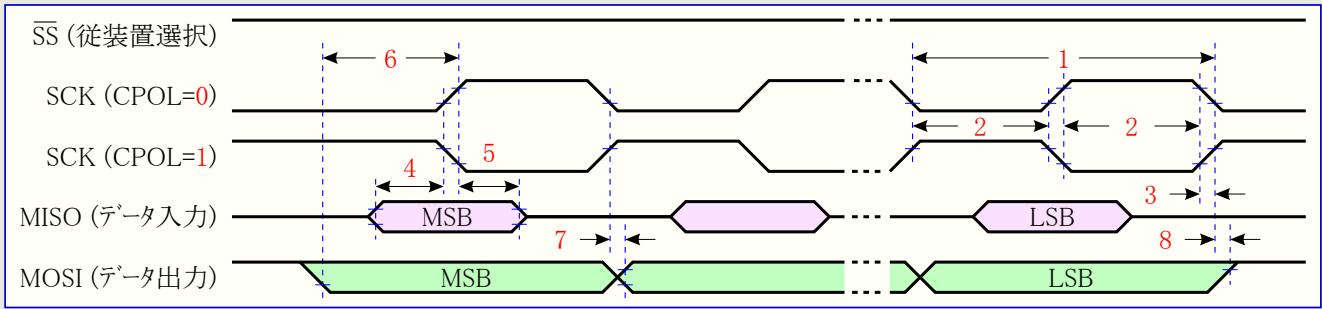


図22-5. SPI タイミング必要条件 (従装置動作)

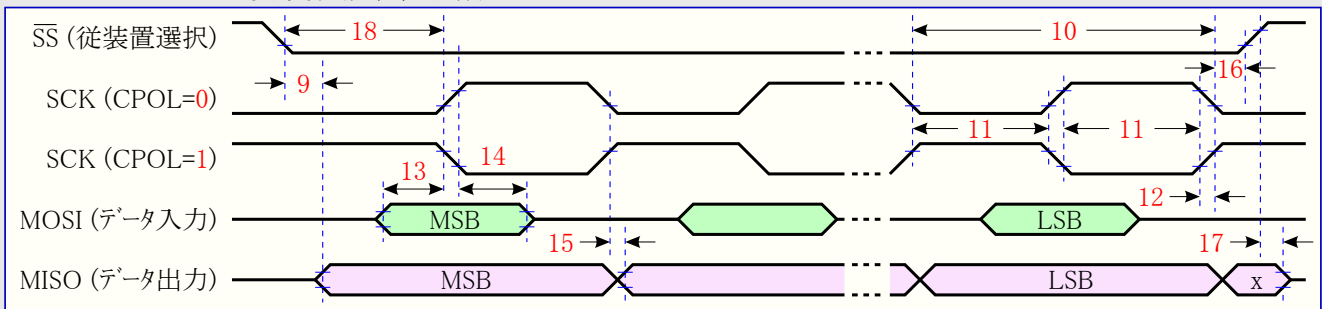


表22-8. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表14-5.参照		ns
2	SCK High/Low期間	主装置		50%デューティ比		
3	SCK上昇/下降時間	主装置		3.6		
4	入力データ 準備時間	主装置		10		
5	入力データ 保持時間	主装置		10		
6	出力からSCK変移時間	主装置		0.5t <sub>SCK</sub>		
7	SCKからの出力遅延時間	主装置		10		
8	SCKからのHigh出力時間	主装置		10		
9	SS ↓からの出力遅延時間	従装置		15		μs
10	SCK周期	従装置	4t <sub>CK</sub>			
11	SCK High/Low期間 (注)	従装置	2t <sub>CK</sub>			ns
12	SCK上昇/下降時間	従装置			1.6	
13	入力データ 準備時間	従装置	10			
14	入力データ 保持時間	従装置	t <sub>CK</sub>			
15	SCKからの出力遅延時間	従装置		15		
16	SCKからのSS ↑遅延時間	従装置	20			
17	SS ↑からの出力Hi-Z遅延時間	従装置		10		
18	SS ↓からのSCK遅延時間	従装置	20			

注: SPIプログラミングでの最小SCK High/Low期間は、2t<sub>CLCL</sub>(t<sub>CK</sub> < 12MHz)、3t<sub>CLCL</sub>(t<sub>CK</sub> ≥ 12MHz)です。

22.10. 並列プログラミング特性

図22-6. 並列プログラミング タイミング (一般的な必要条件)

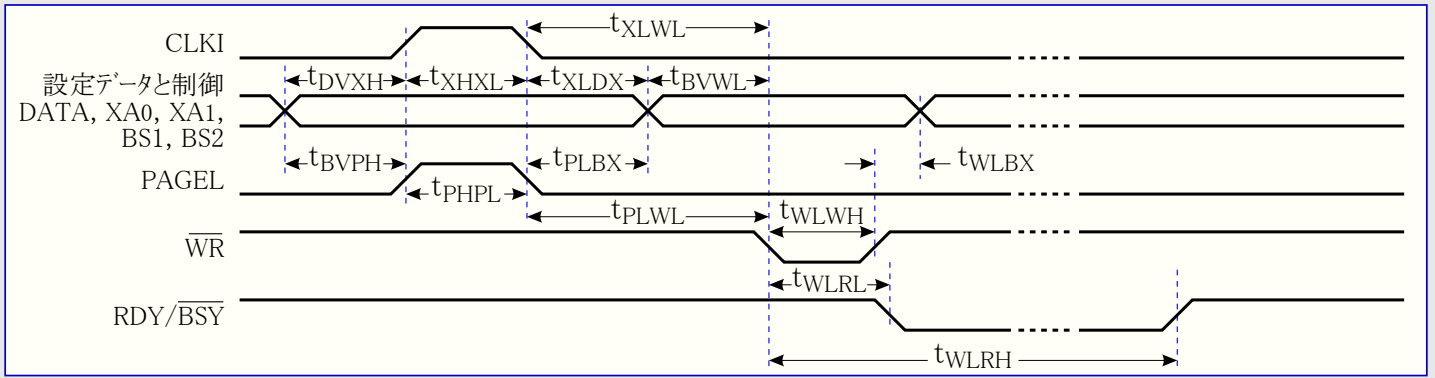
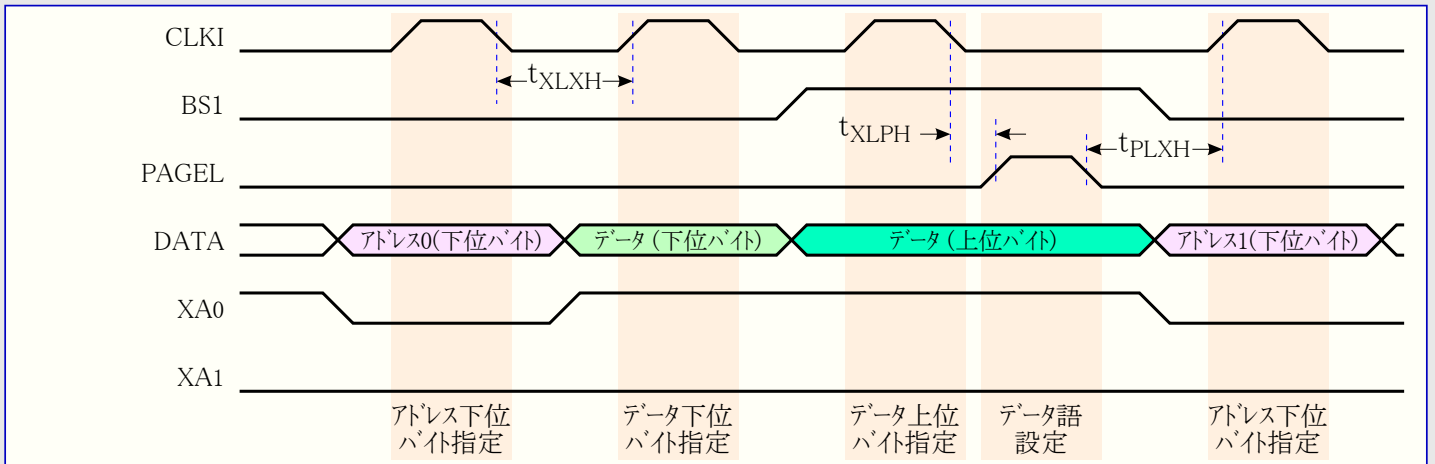
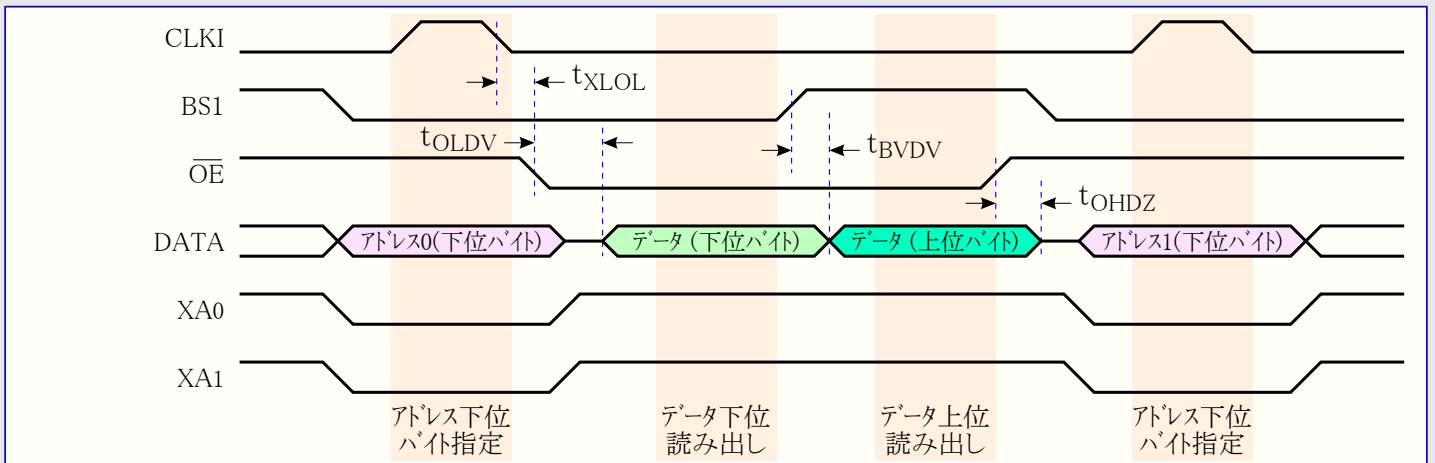


図22-7. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図22-6.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22-8. 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22-6.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22-9. 並列プログラミング特性 (TA=25°C, VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
VPP	プログラミング許可電圧	11.5		12.5	V
IPP	プログラミング許可電流			250	μA
tDVXH	CLKI ↑ に対するデータと制御の準備時間	67			ns
tXLXH	CLKI ↓ から次CLKI ↑ までの待機時間	200			
tXHXL	CLKI Highパルス幅	150			
tXLDX	CLKIパルス ↓ 後のデータと制御の保持時間	67			
tXLWL	CLKIパルス ↓ 後のWR ↓ 待機時間	0			
tXLPH	CLKIパルス ↓ 後のPAGELパルス ↑ 待機時間	0			
tPLXH	PAGELパルス ↓ 後のCLKIパルス ↑ 待機時間	150			
tBVPH	PAGELパルス ↑ に対するBS1準備時間	67			
tPHPL	PAGEL Highパルス幅	150			
tPLBX	PAGELパルス ↓ 後のBS1保持時間	67			
tWLBX	WRパルス ↓ 後のBS1,BS2保持時間	67			
tPLWL	PAGELパルス ↓ 後のWRパルス ↓ 待機時間	67			
tBVWL	WRパルス ↓ に対するBS1準備時間	67			
tWLWH	WR Lowパルス幅	150			
tWLRH	WRパルス ↓ 後のRDY/BSY ↓ 遅延時間	0		1	
tWLRH	書き込み時間 (WR ↓ からRDY/BSY ↑) (注1)	3.7		4.5	ms
tWLRH_CE	チップ消去時間 (WR ↓ からRDY/BSY ↑) (注2)	7.5		9	
tXLCL	CLKIパルス ↓ 後のOE ↓ 待機時間	0			ns
tBVDV	BS1有効からのDATA遅延時間	0		250	
tOLDV	OE ↓ 後のDATA出力遅延時間			250	
tOHDZ	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。

22.11. 直列プログラミング特性

図22-9. 直列プログラミング タイミング

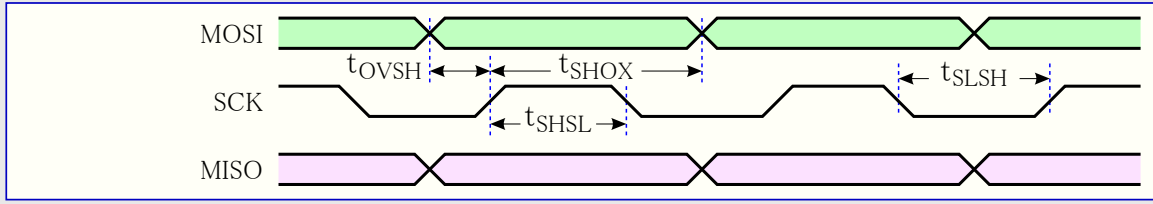


図22-10. 直列プログラミング バイト通信波形

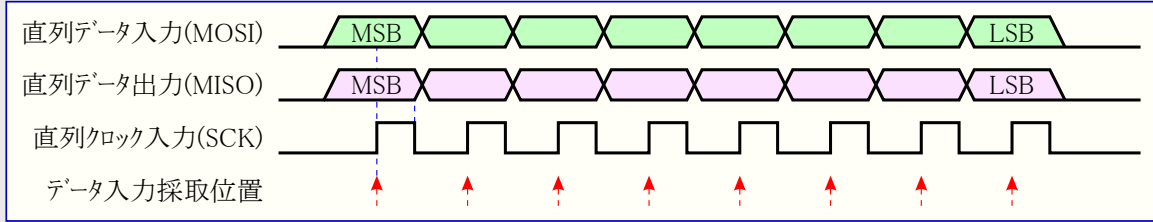


表22-10. 直列プログラミング特性 (別記を除いて、TA=-40℃～85℃, VCC=1.8～5.5V)

シンボル	項目	最小	代表	最大	単位
1/tCLCL	発振器周波数	2.7～4.5V	0	6	MHz
		4.5～5.5V	0	12	
tCLCL	発振器周期	2.7～4.5V	166		ns
		4.5～5.5V	83		
tSHSL	SCKパルスHレベル幅	2tCLCL			
tSLSH	SCKパルスLレベル幅	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			

## 23. 代表特性

本章内に含まれたデータは主に同じ製法と設計法の類似デバイスの特徴付けとシミュレーションに基づいています。従って、このデータはデバイスがどう反応するかについての指標として扱われるべきです。

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)  $C_L(\text{負荷容量}) \times V_{CC}(\text{動作電圧}) \times f_{sw}(\text{I/Oピンの平均切り替え周波数})$  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。

### 23.1. ATtiny48

#### 23.1.1. 活動動作消費電流

図23-1. 活動動作消費電流 対 周波数 (100kHz~1MHz)

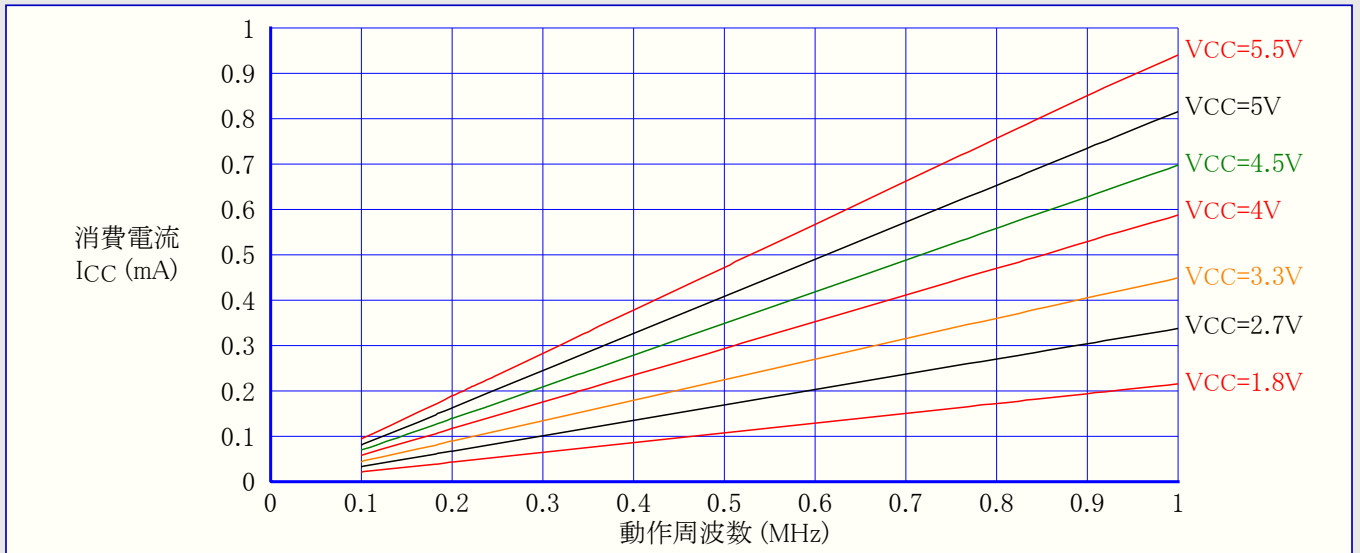


図23-2. 活動動作消費電流 対 周波数 (1MHz~12MHz)

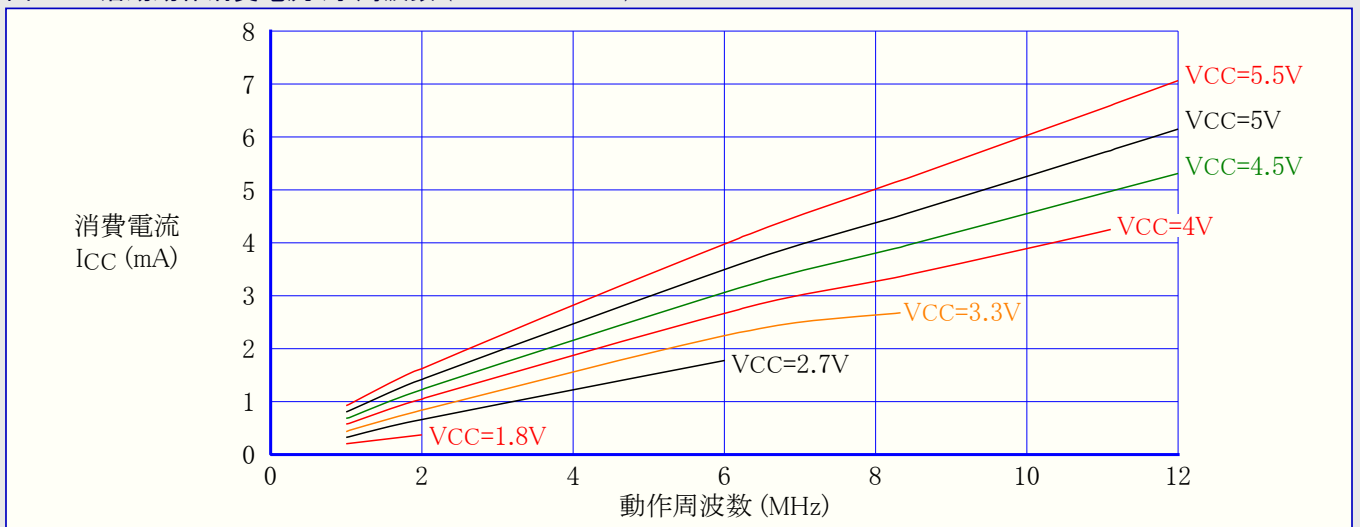


図23-3. 活動動作消費電流 対 動作電圧 (内蔵発振器, 8MHz)

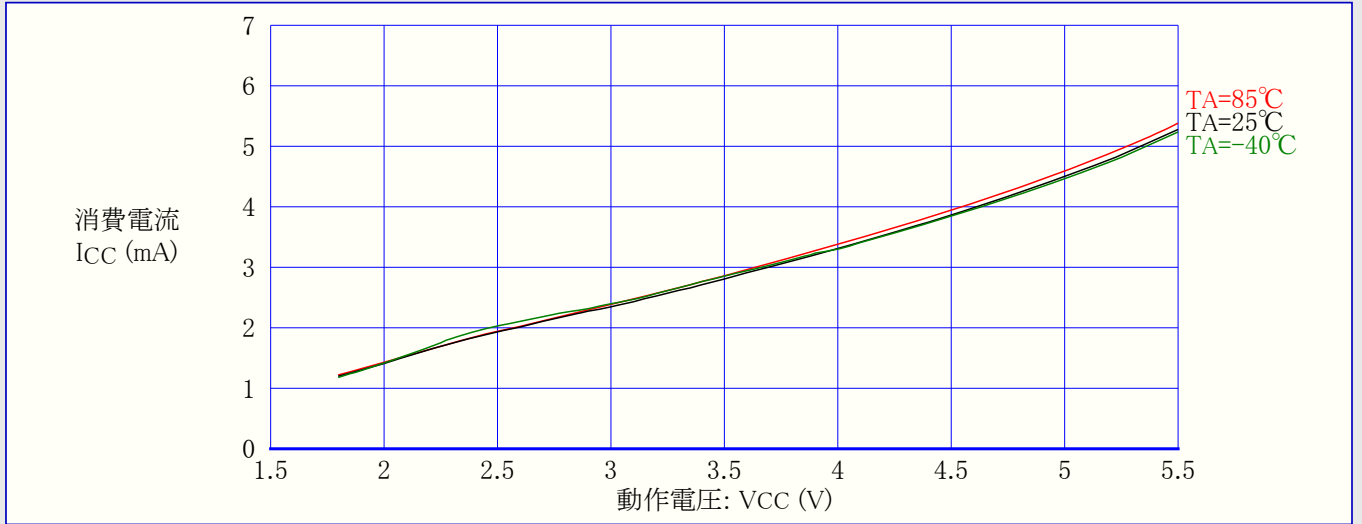


図23-4. 活動動作消費電流 対 動作電圧 (内蔵発振器, 1MHz)

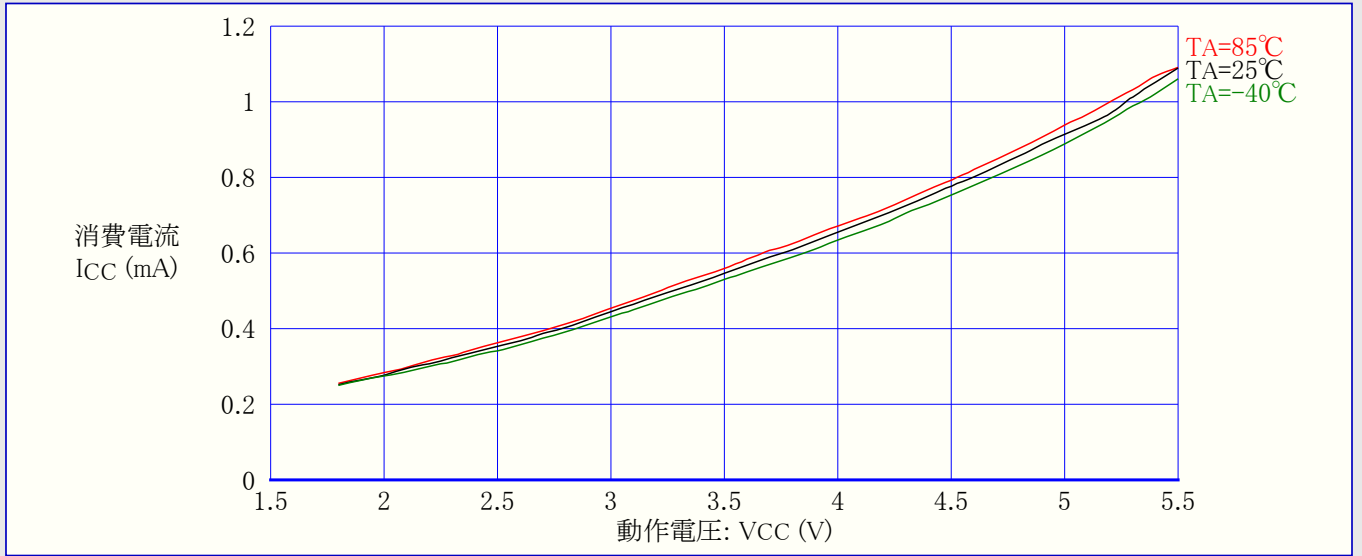
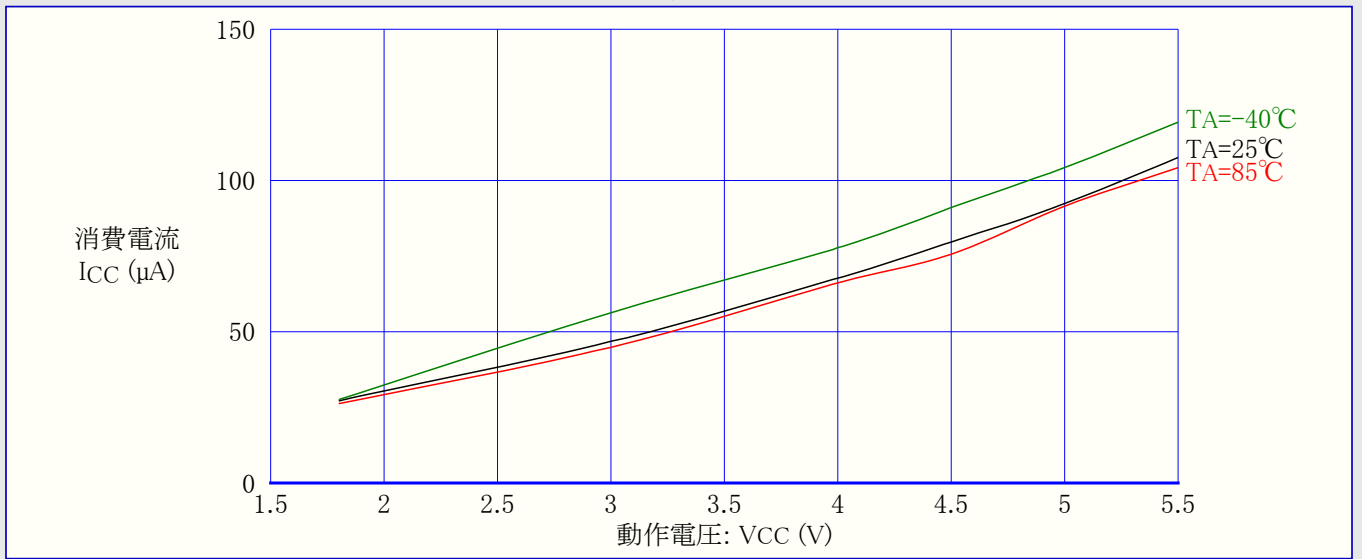


図23-5. 活動動作消費電流 対 動作電圧 (内蔵WDT発振器, 128kHz)



### 23.1.2. アイドル動作消費電流

図23-6. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

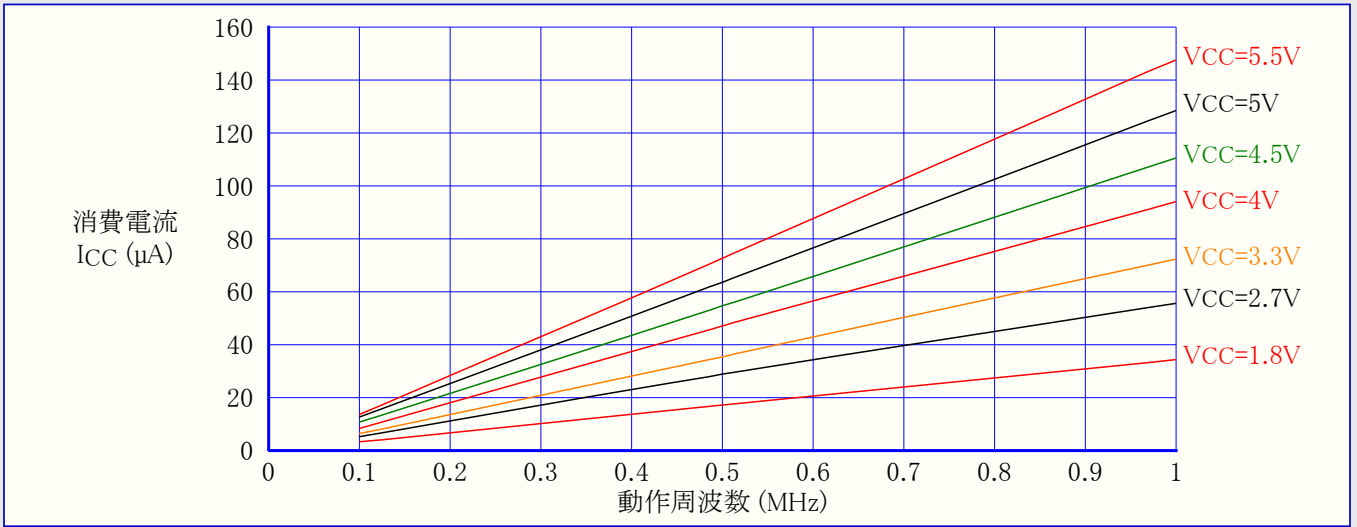


図23-7. アイドル動作消費電流 対 周波数 (1MHz~12MHz)

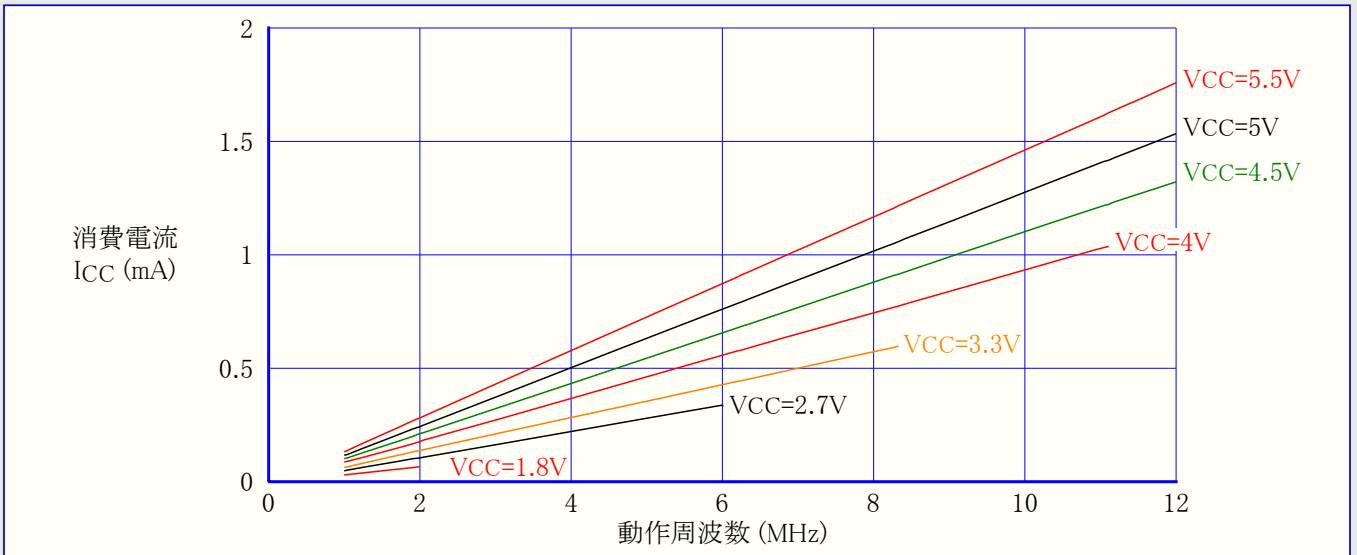


図23-8. アイドル動作消費電流 対 動作電圧 (内蔵発振器, 8MHz)

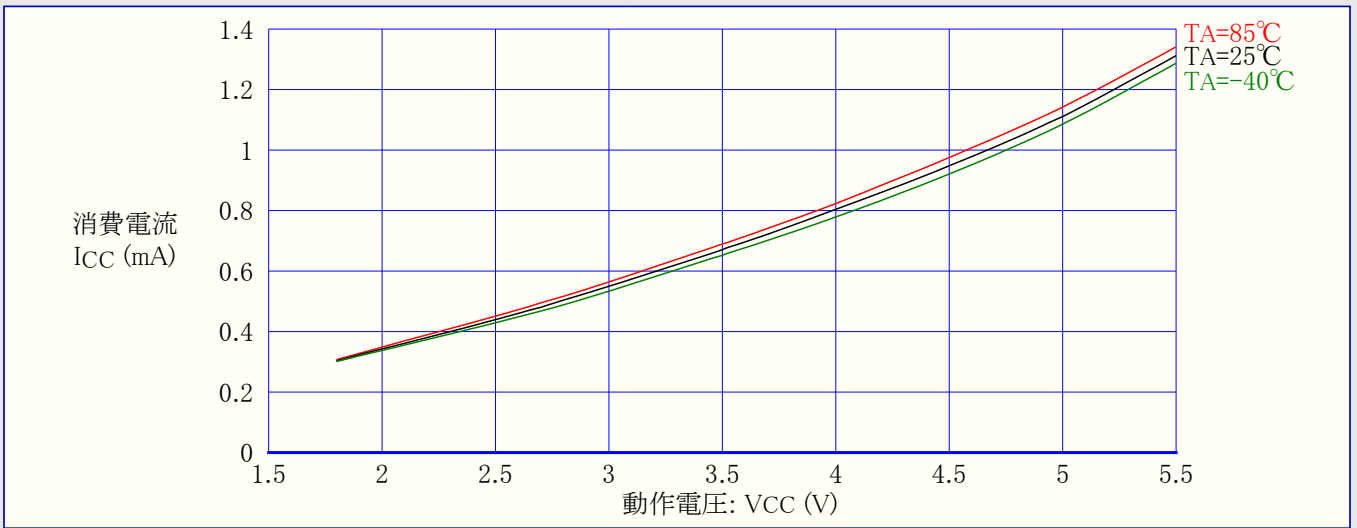


図23-9. アイドル動作消費電流 対 動作電圧 (内蔵発振器, 1MHz)

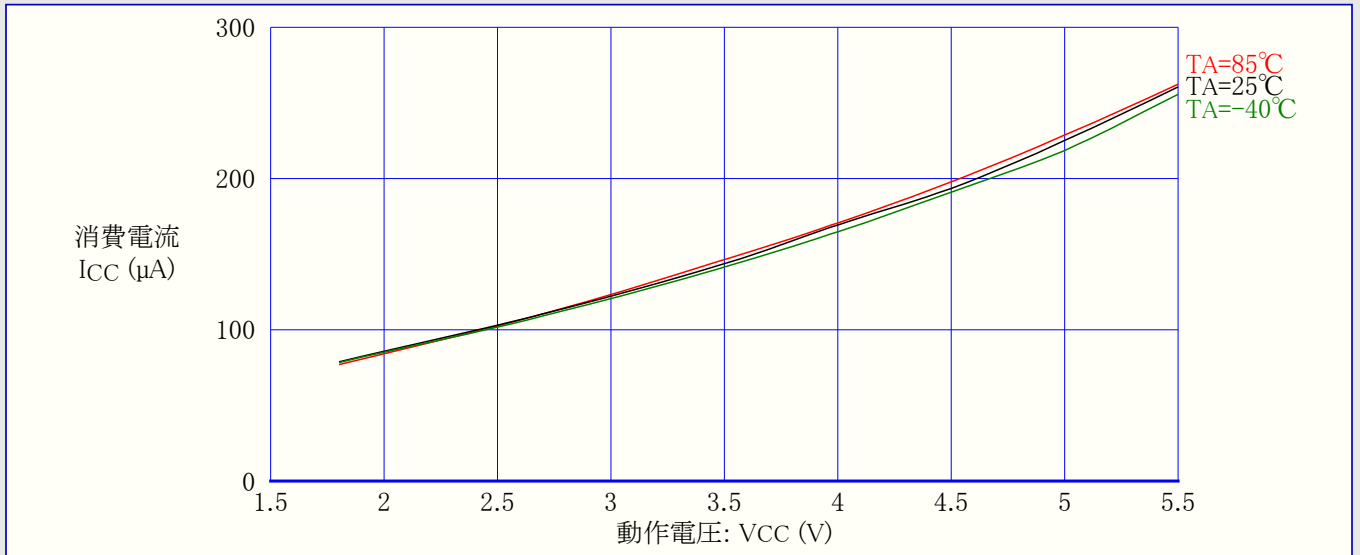
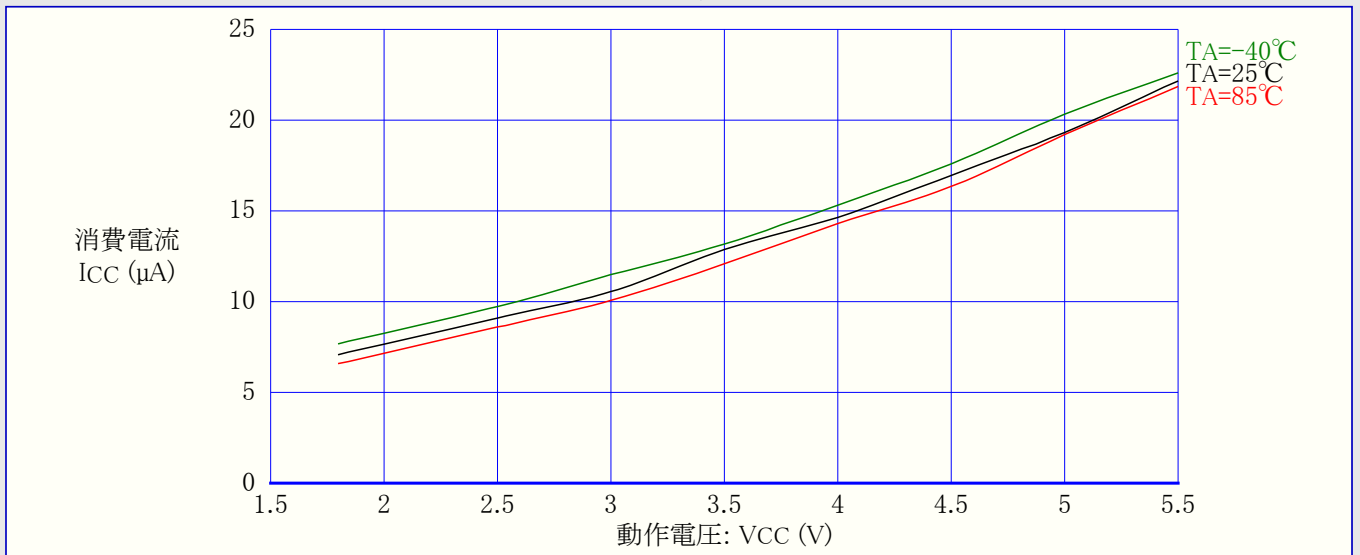


図23-10. アイドル動作消費電流 対 動作電圧 (内蔵WDT発振器, 128kHz)



23.1.3. パワーダウン動作消費電流

図23-11. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)

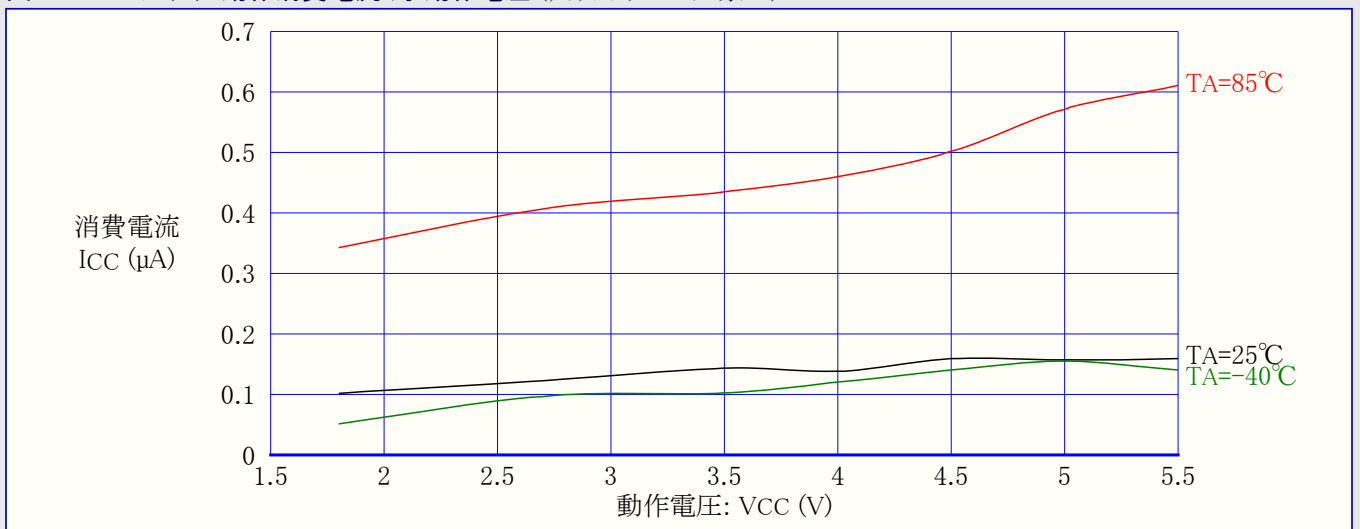
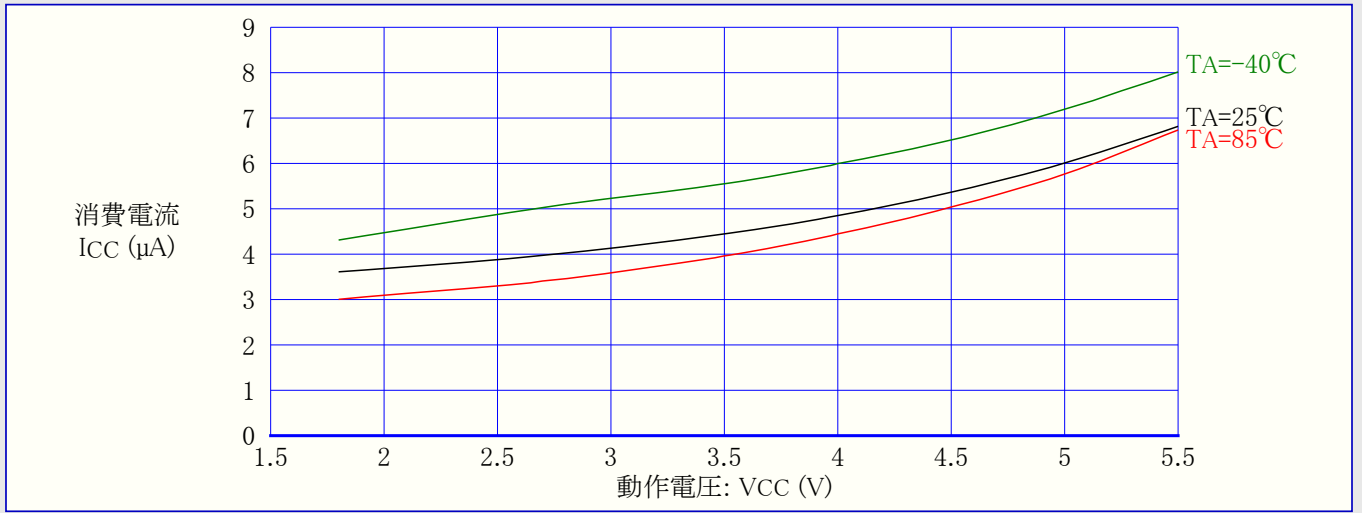


図23-12. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



### 23.1.4. リセット消費電流

図23-13. リセット消費(供給)電流 対 周波数 (100kHz~1MHz)

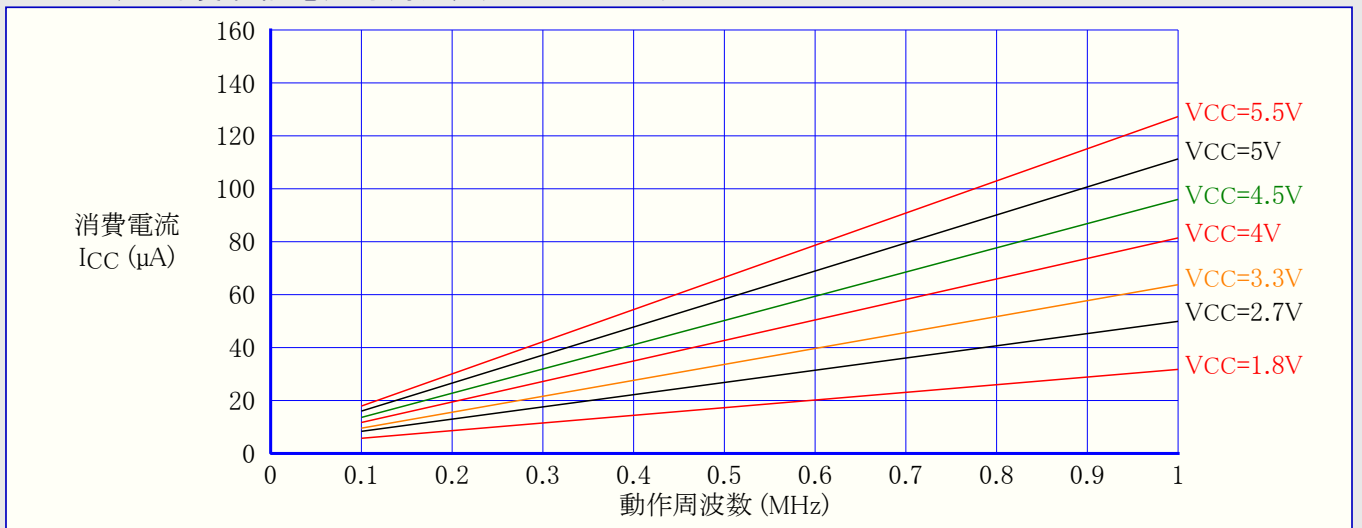
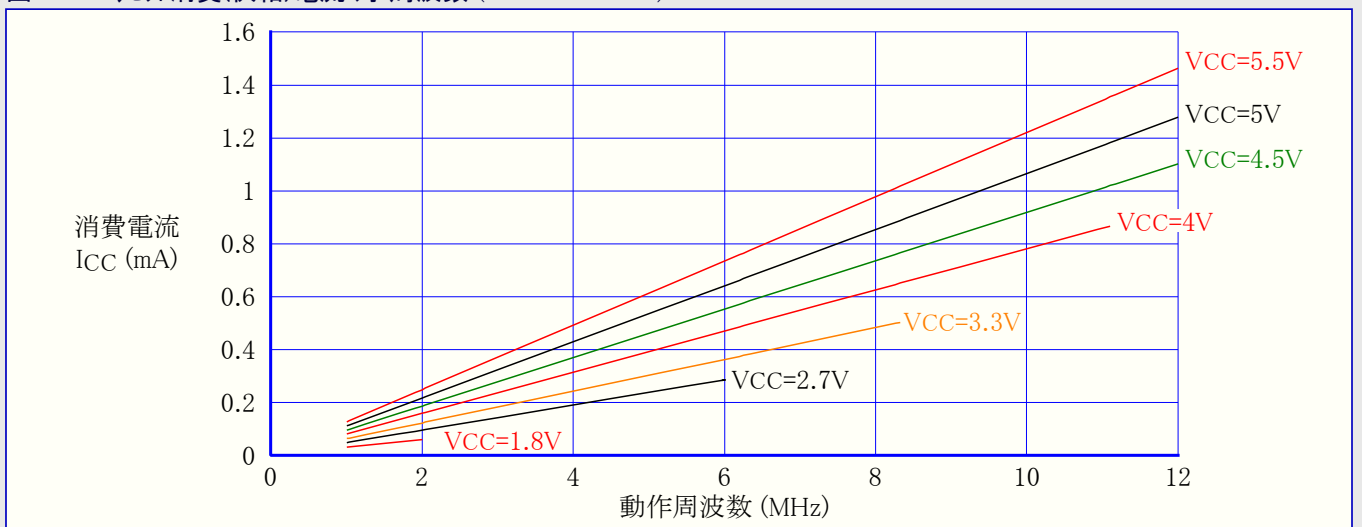


図23-14. リセット消費(供給)電流 対 周波数 (1MHz~12MHz)



23.1.5. 周辺機能部消費電流

図23-15. 低電圧検出器(BOD)消費電流 対 動作電圧

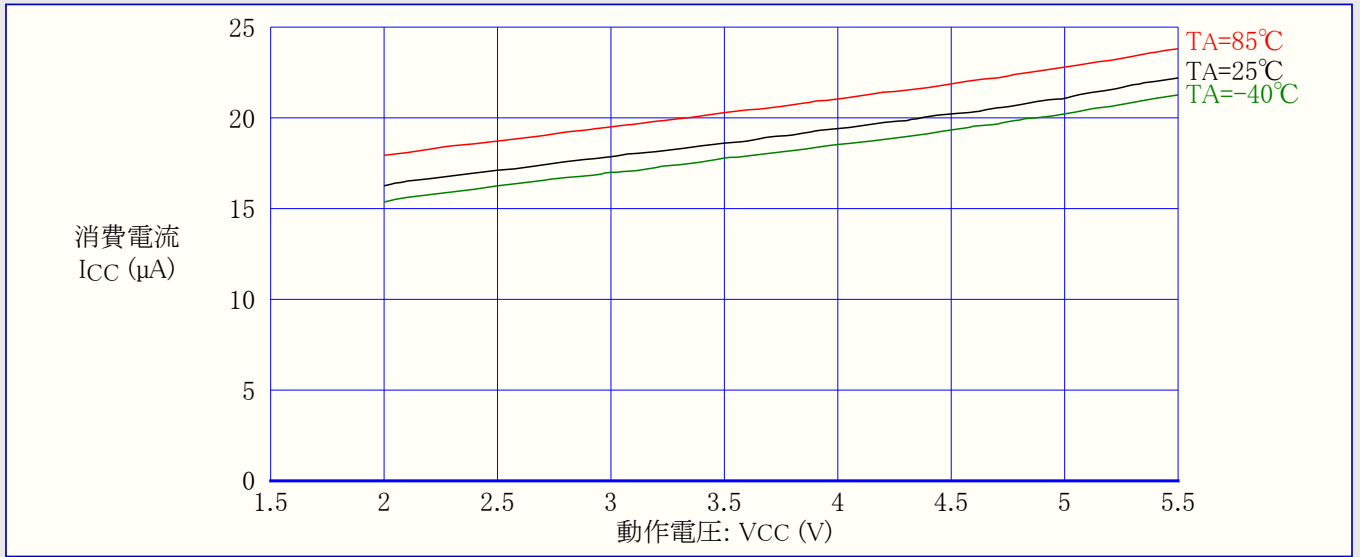


図23-16. A/D変換器消費電流 対 動作電圧 (AREF=AVCC)

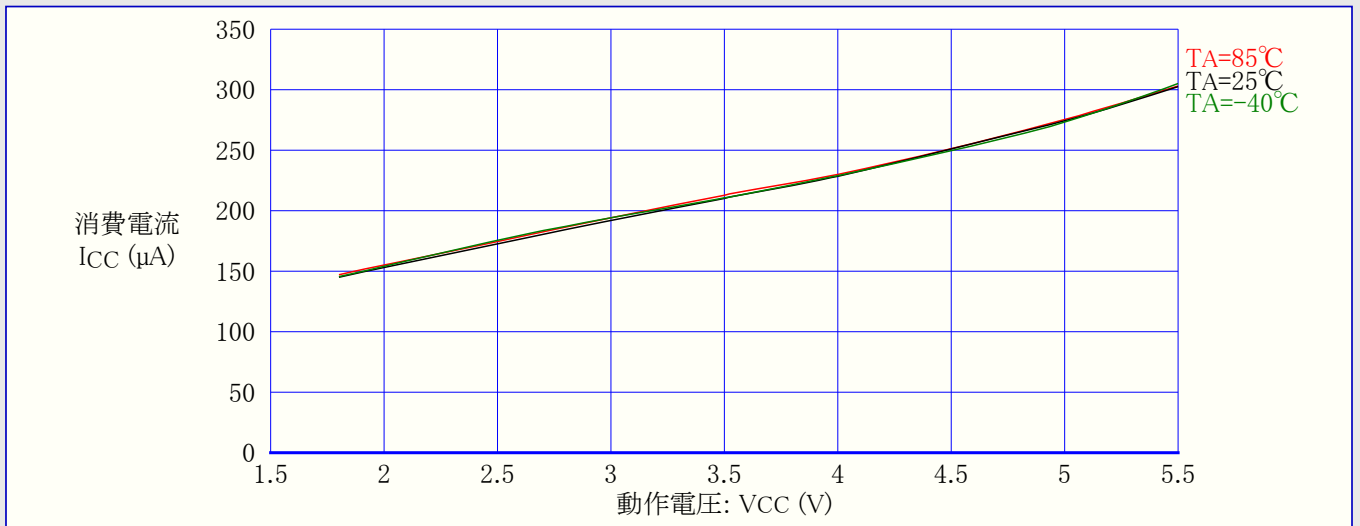


図23-17. アナログ比較器消費電流 対 動作電圧

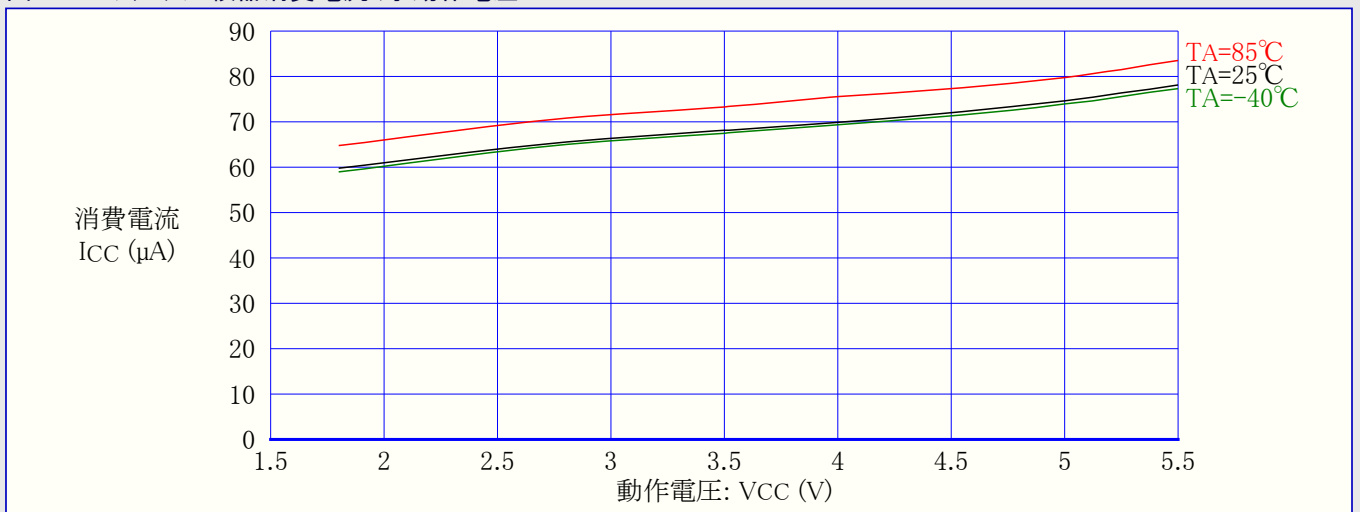
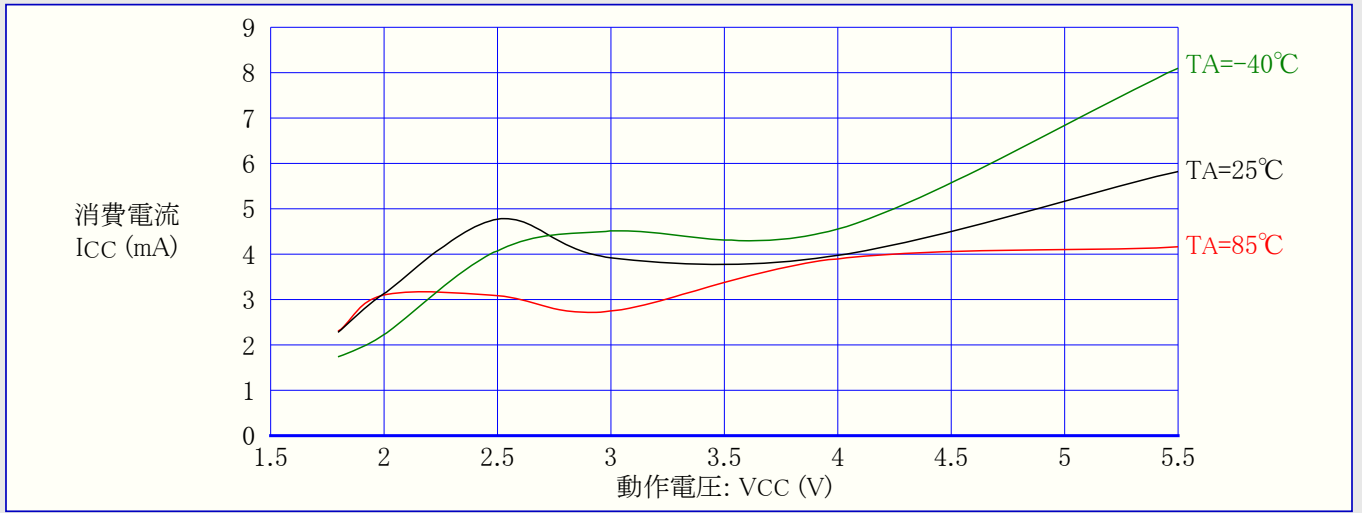


図23-18. プログラミング電流 対 動作電圧



23.1.6. プルアップ抵抗

図23-19. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

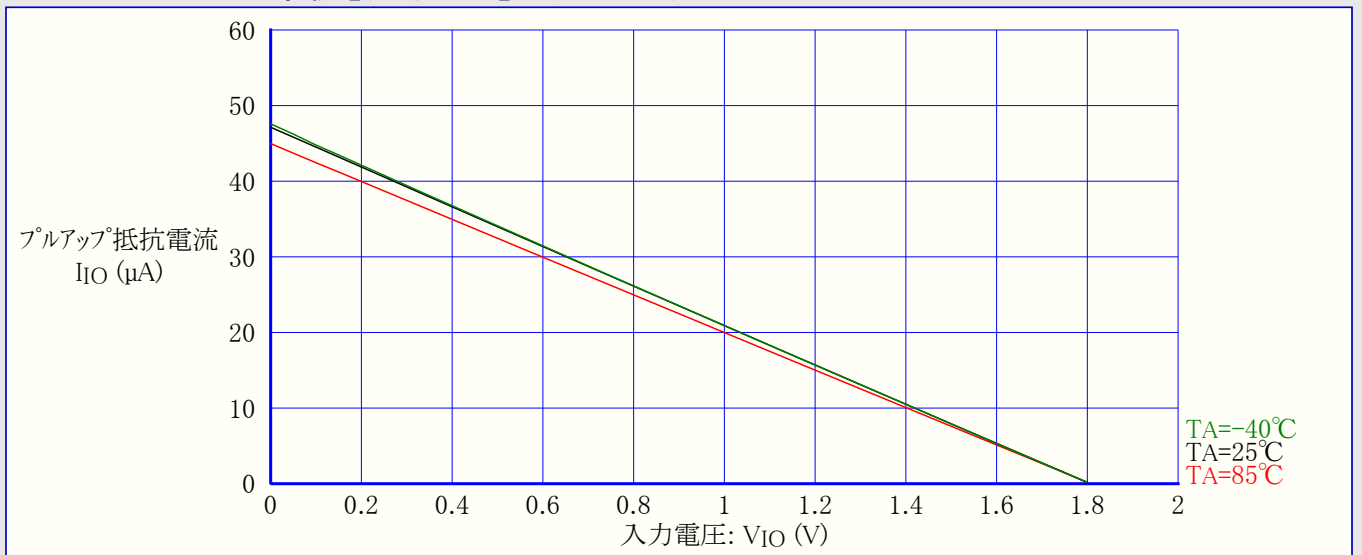


図23-20. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

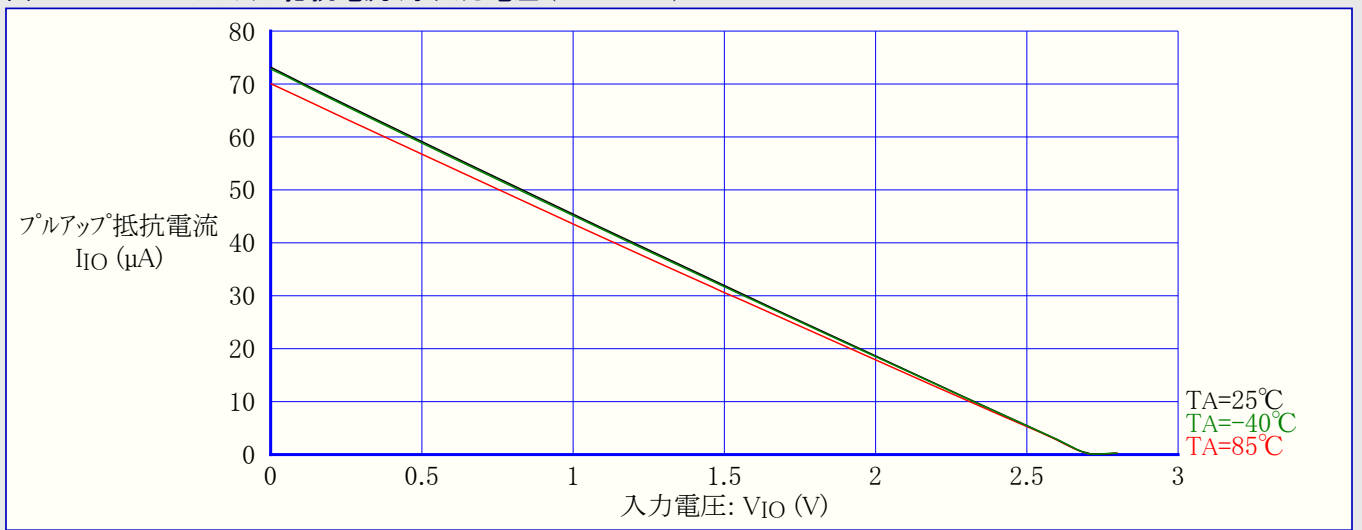


図23-21. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

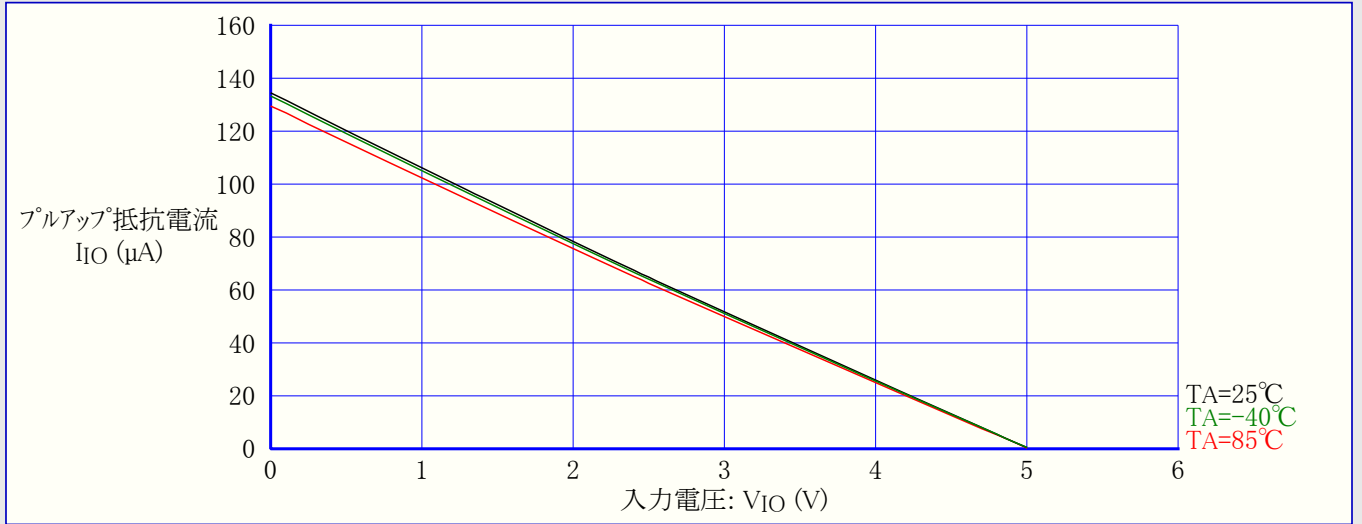


図23-22. RESETプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

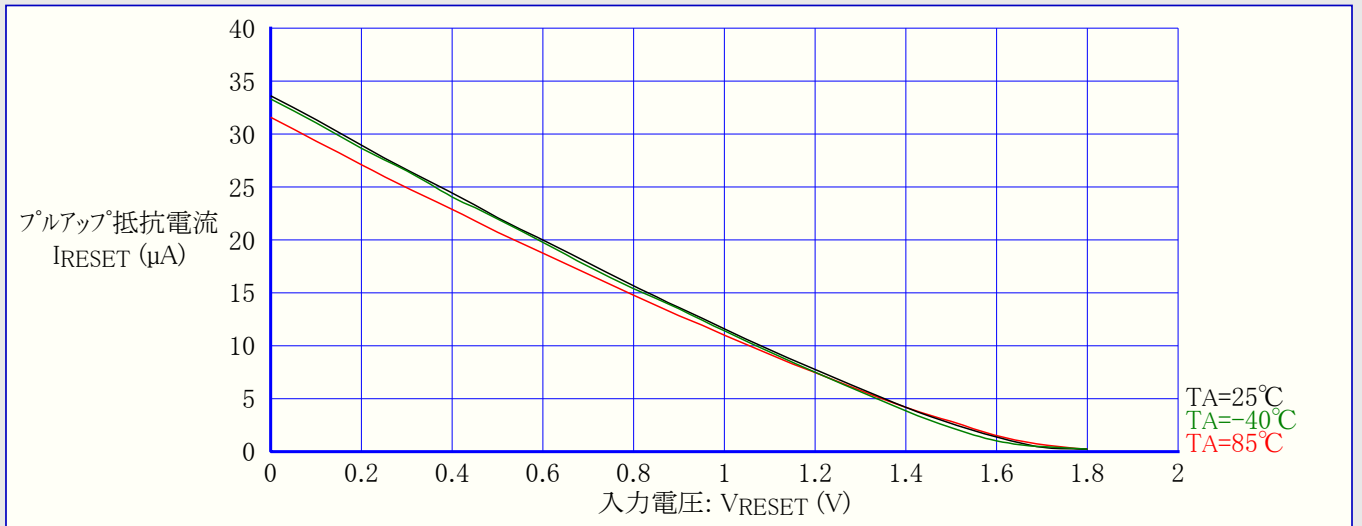


図23-23. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

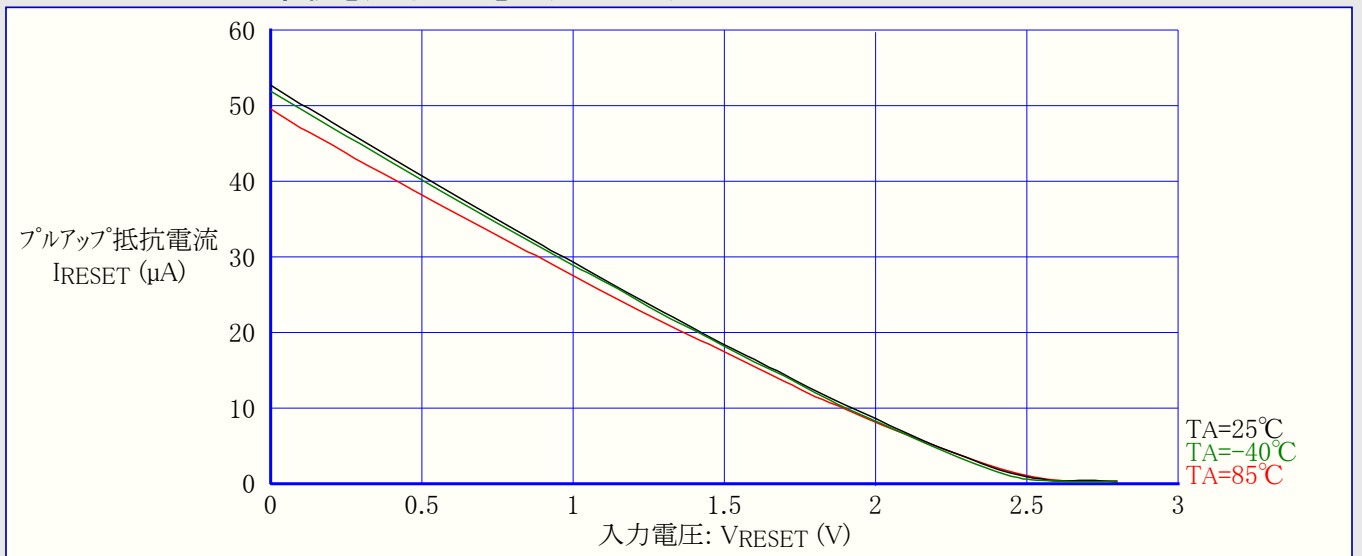
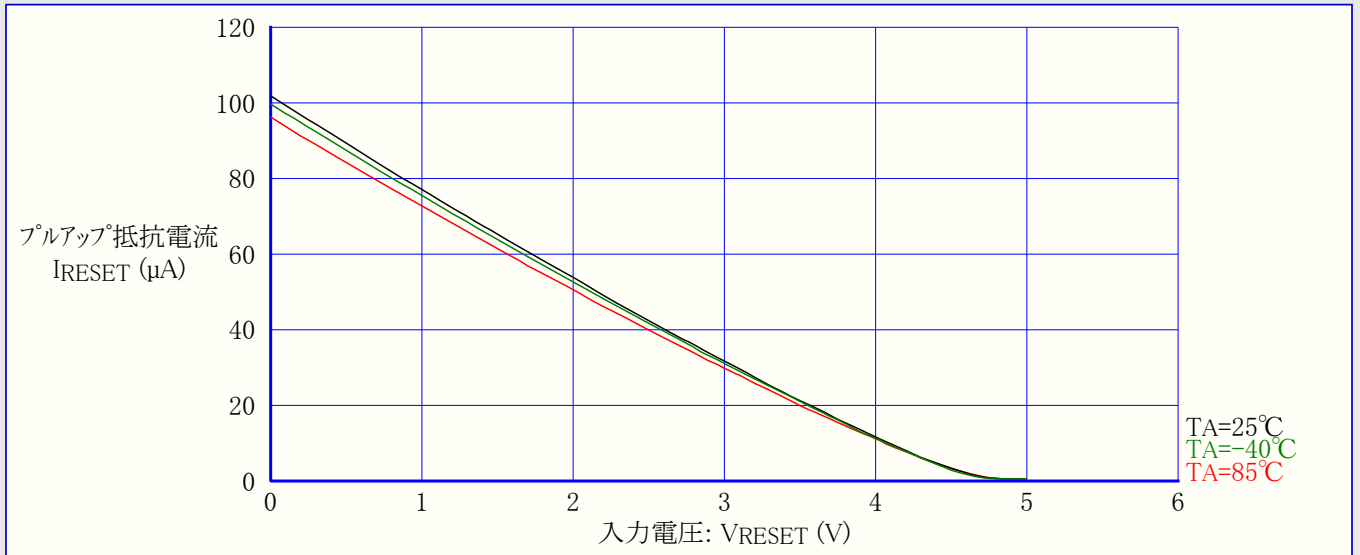


図23-24. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)



### 23.1.7. 出力駆動部能力

図23-25. 高シンク駆動I/Oピン 出力電圧 対 吸い込み電流 (VCC=1.8V)

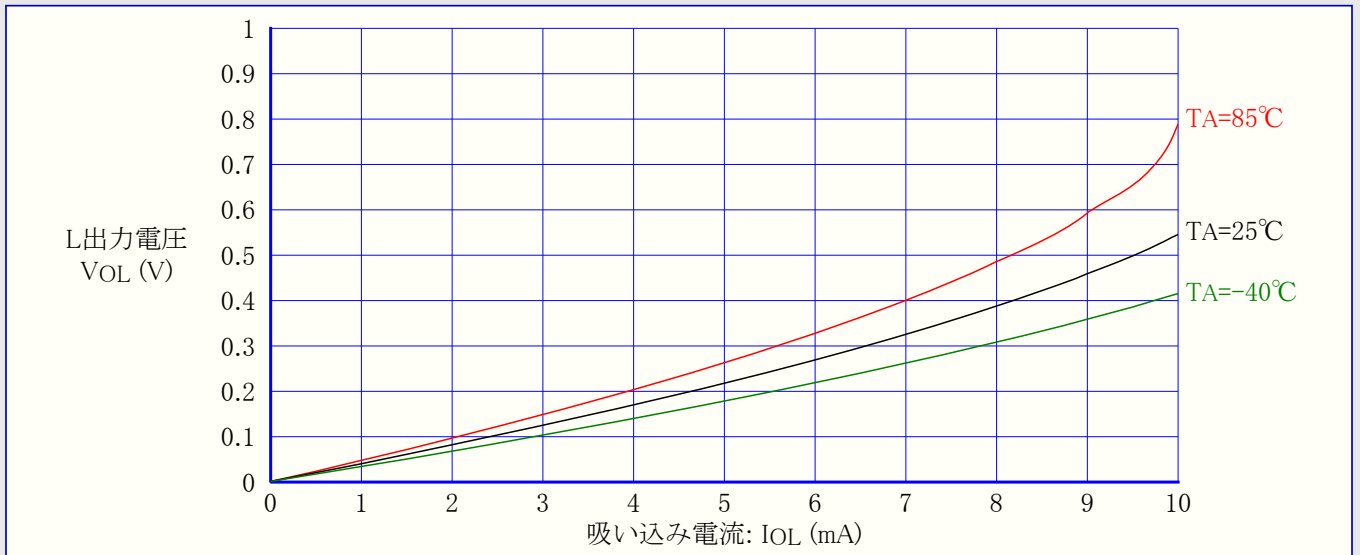


図23-26. 高シンク駆動I/Oピン 出力電圧 対 吸い込み電流 (VCC=3V)

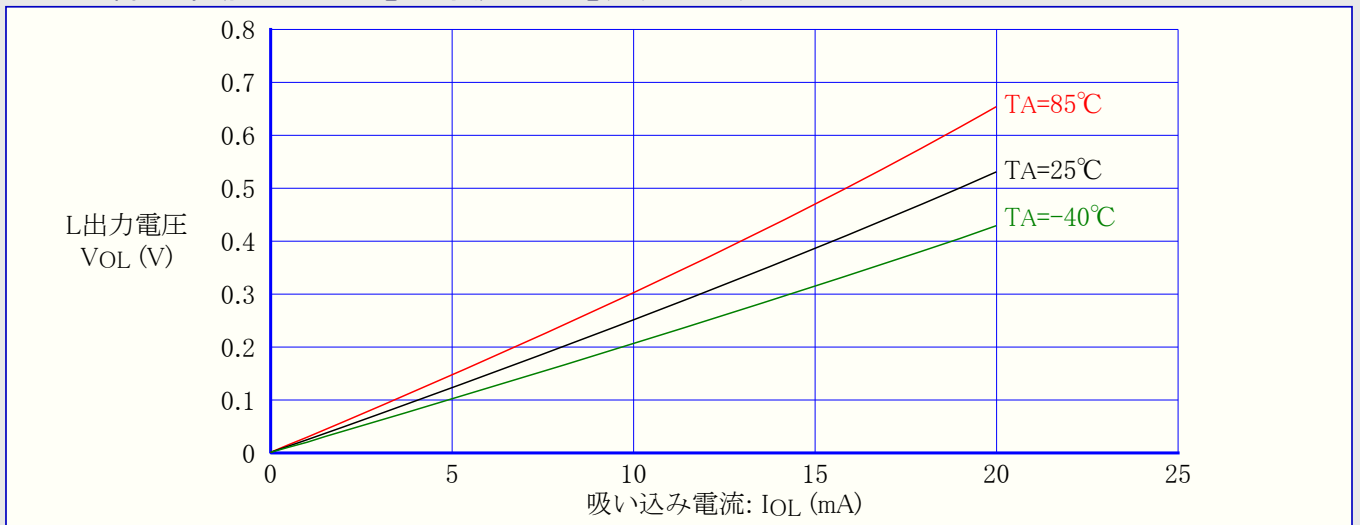


図23-27. 高シンク駆動I/Oピン出力電圧 対 吸い込み電流 (VCC=5V)

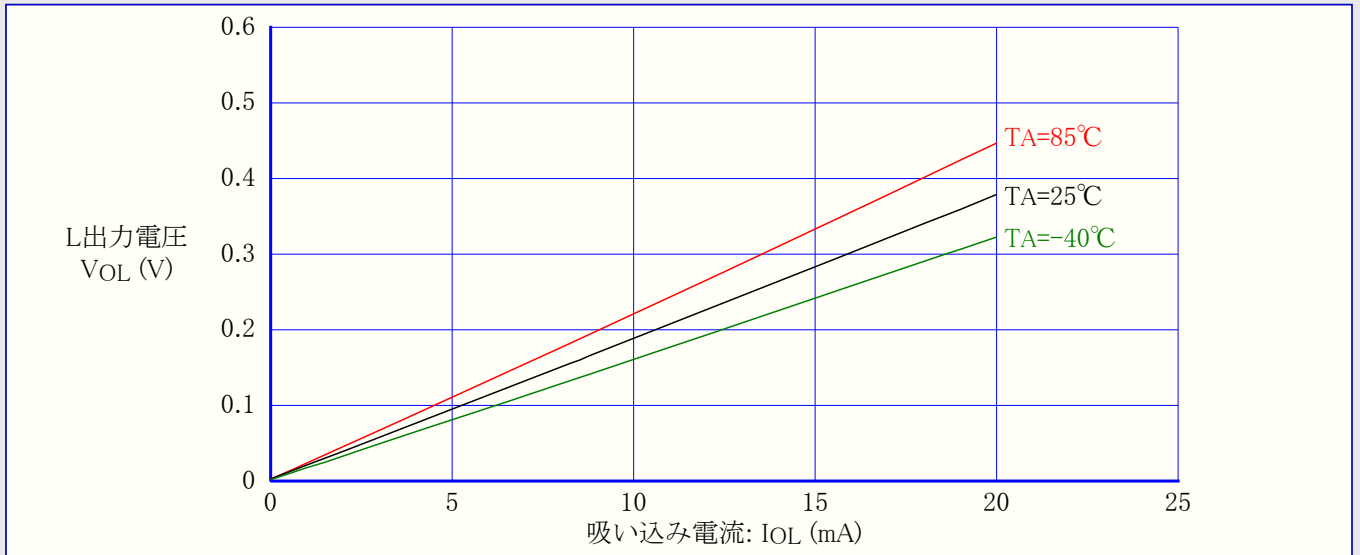


図23-28. 標準I/Oピン出力電圧 対 吸い込み電流 (VCC=1.8V)

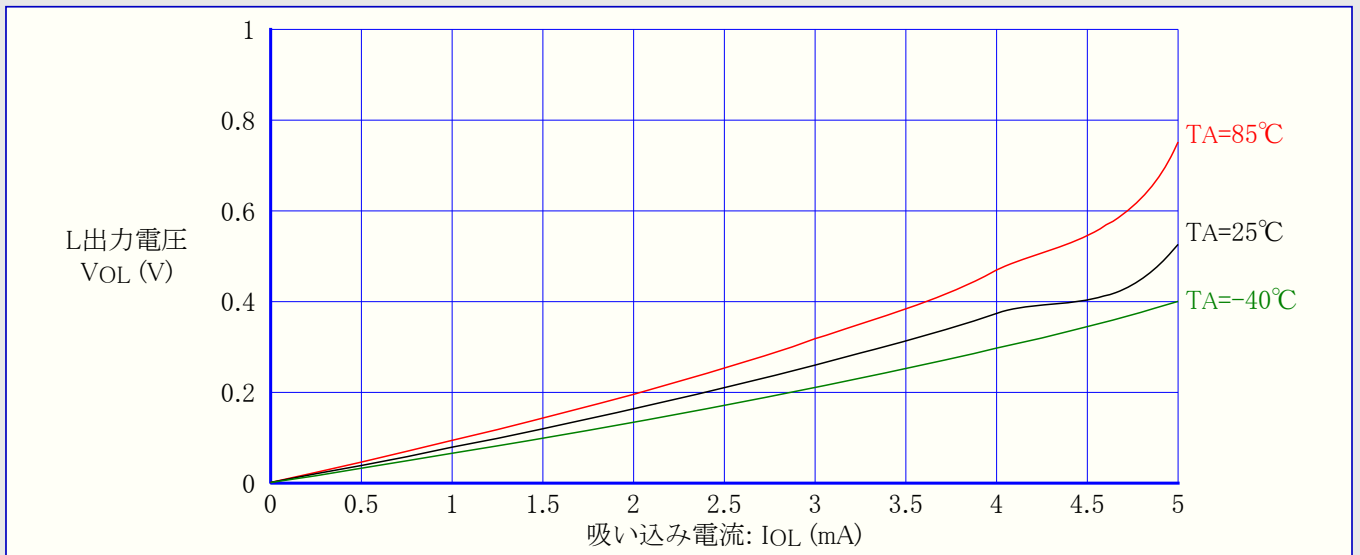


図23-29. 標準I/Oピン出力電圧 対 吸い込み電流 (VCC=3V)

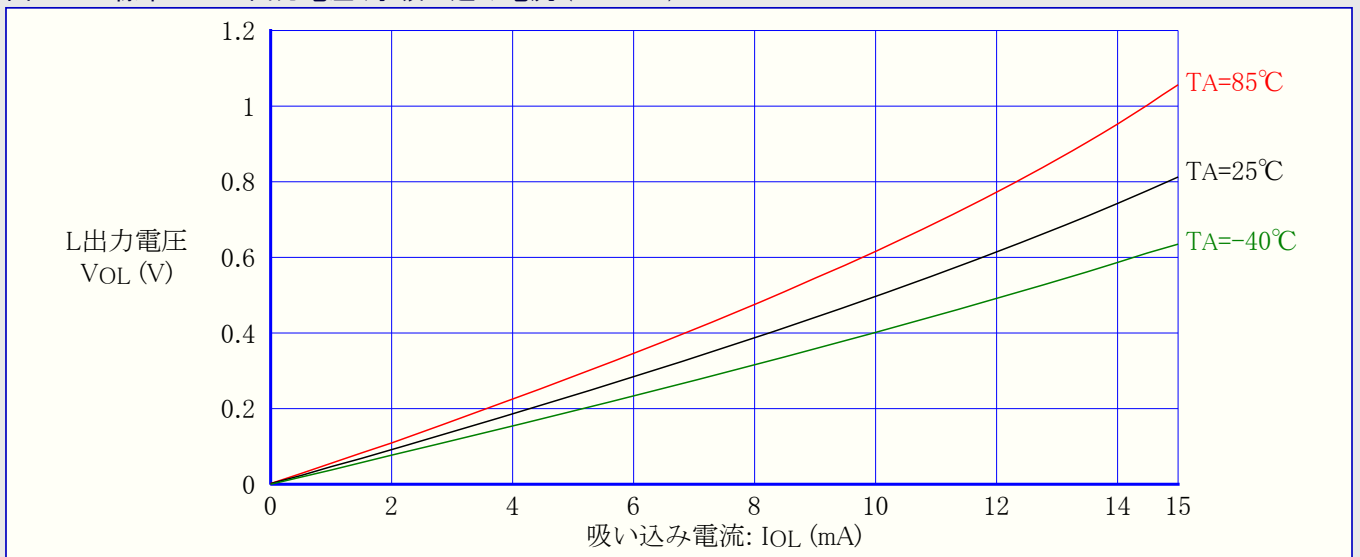


図23-30. 標準I/Oピン 出力電圧 対 吸い込み電流 (VCC=5V)

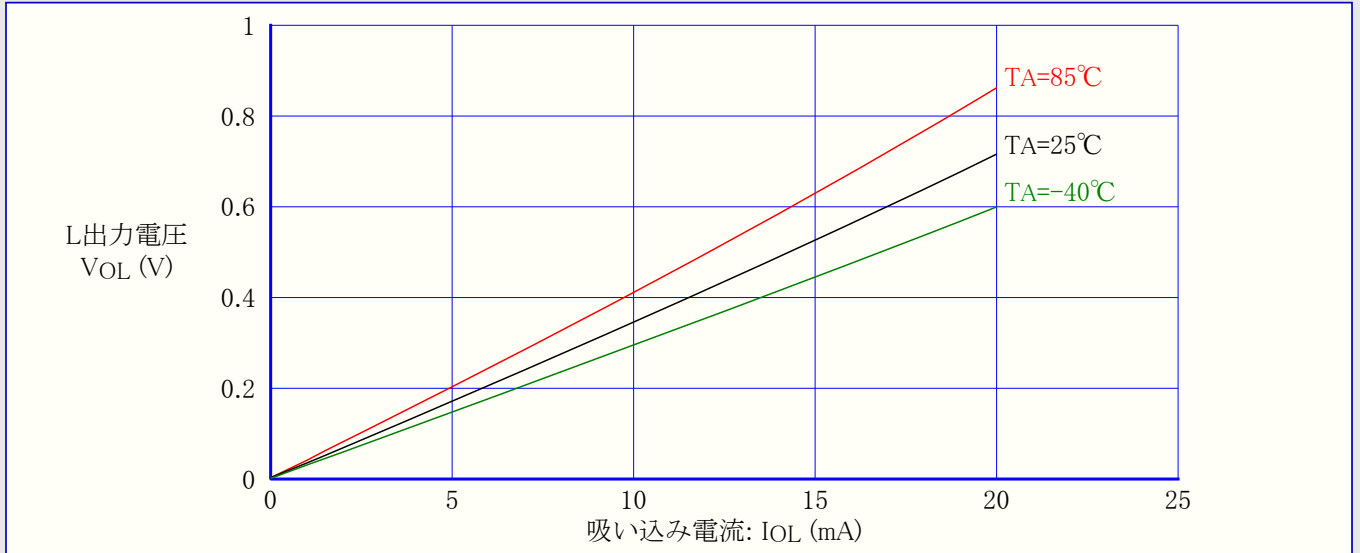


図23-31. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=1.8V)

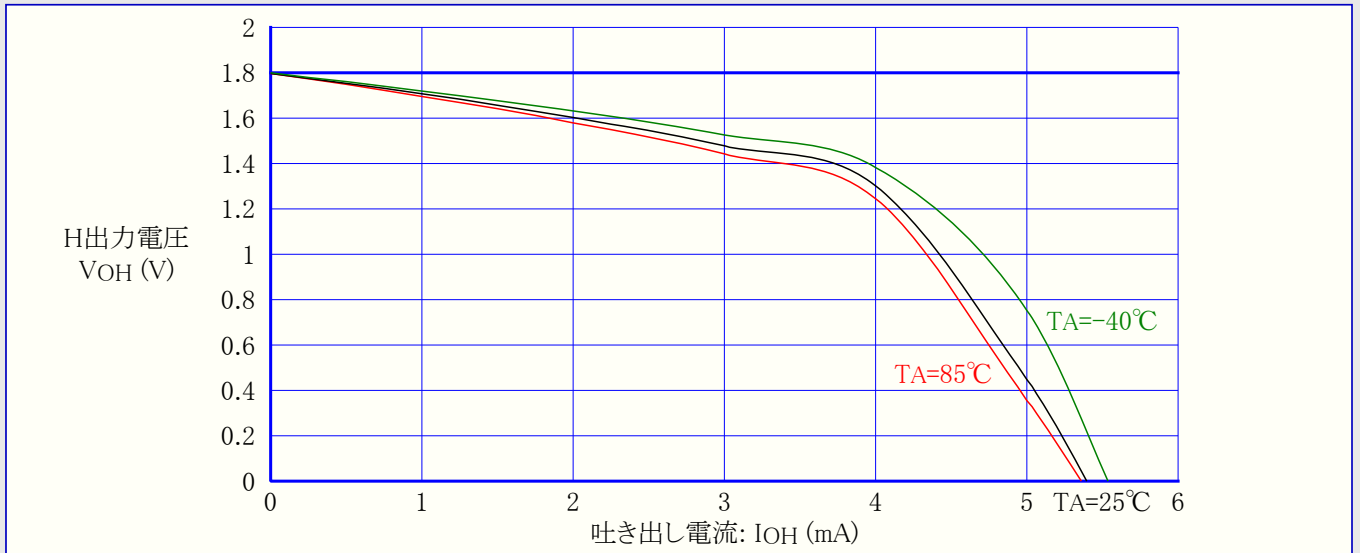


図23-32. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=3V)

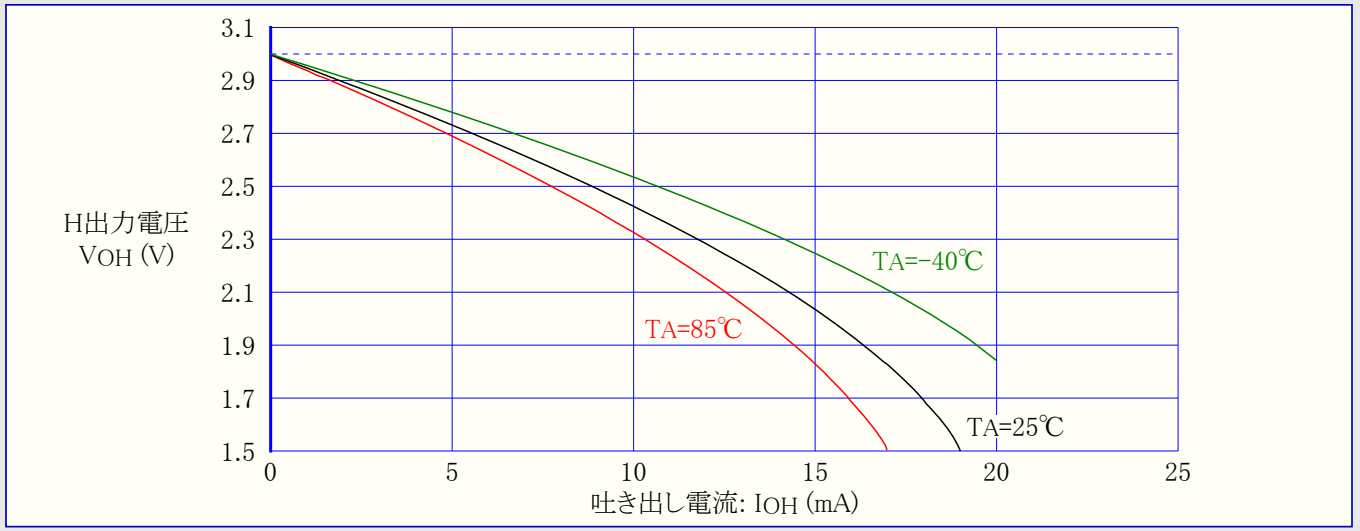


図23-33. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=5V)

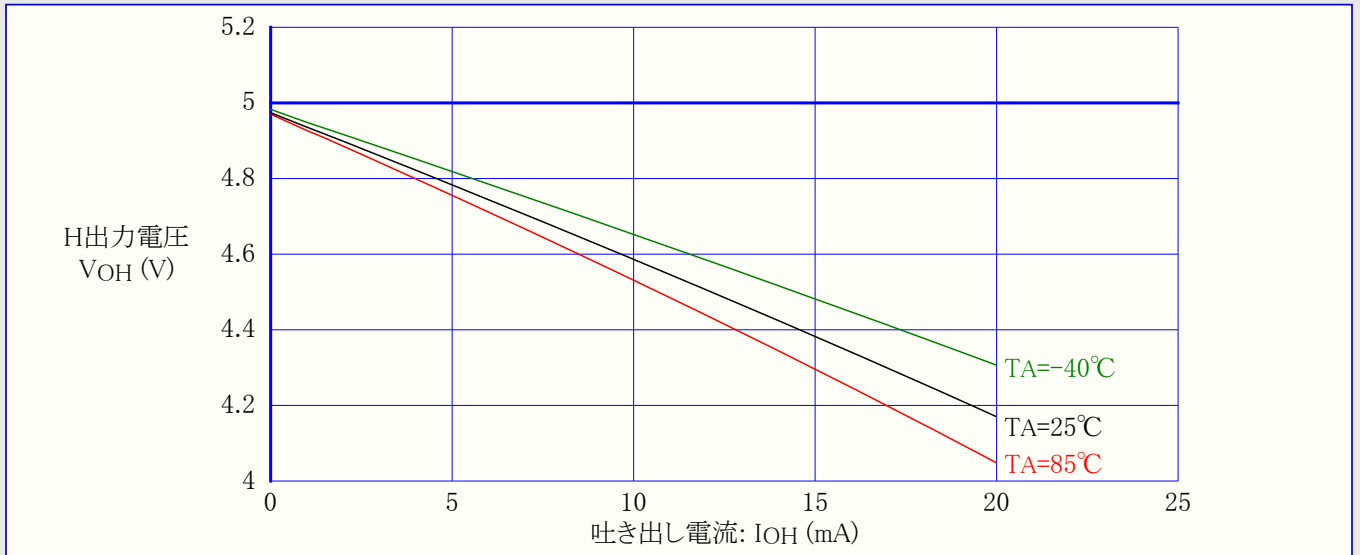


図23-34. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=1.8V)

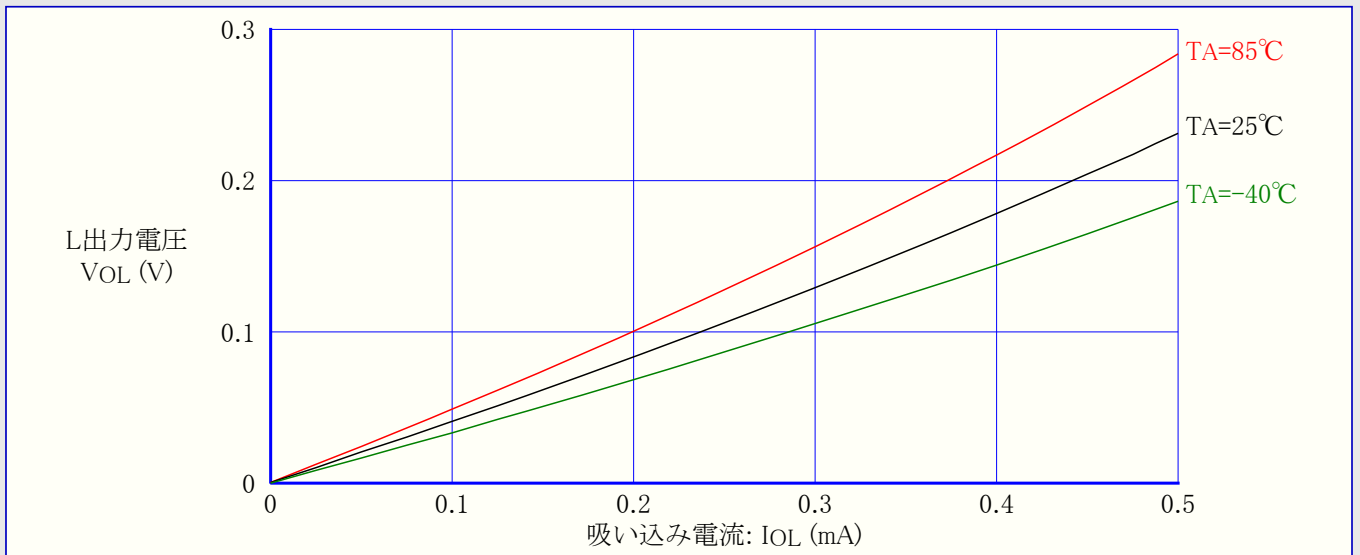


図23-35. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=3V)

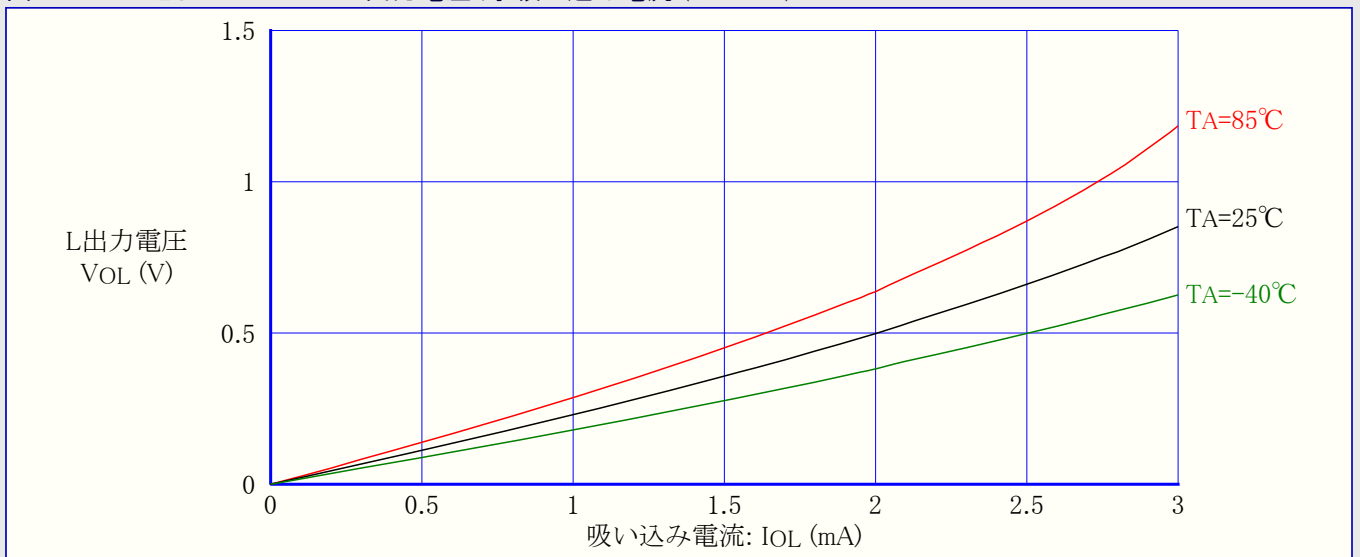


図23-36. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=5V)

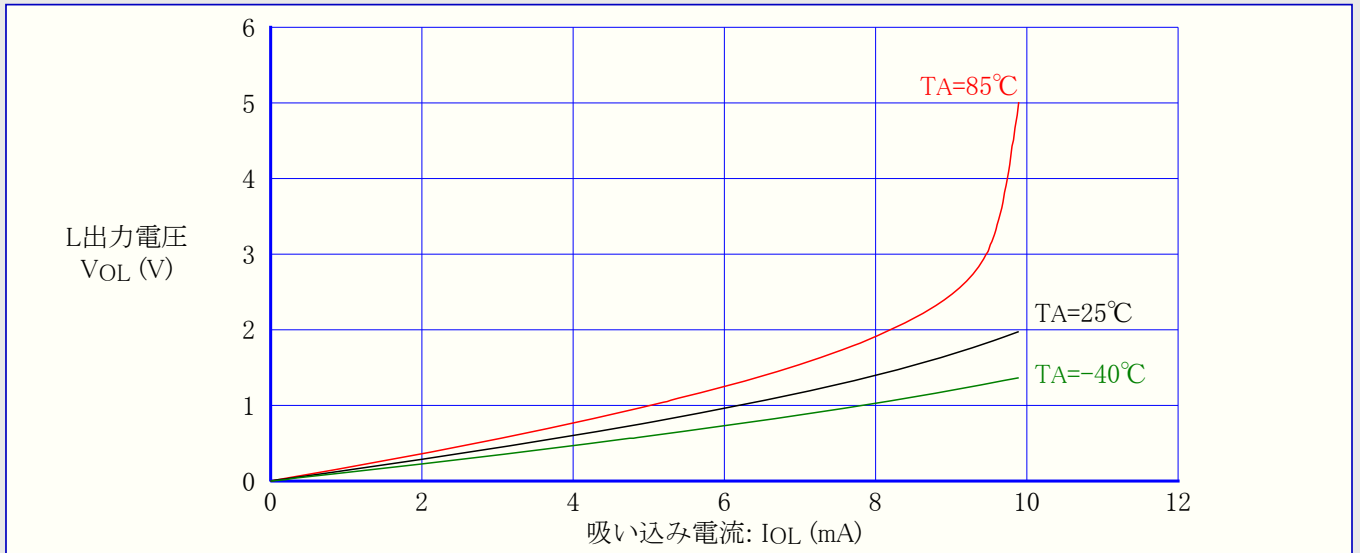


図23-37. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=1.8V)

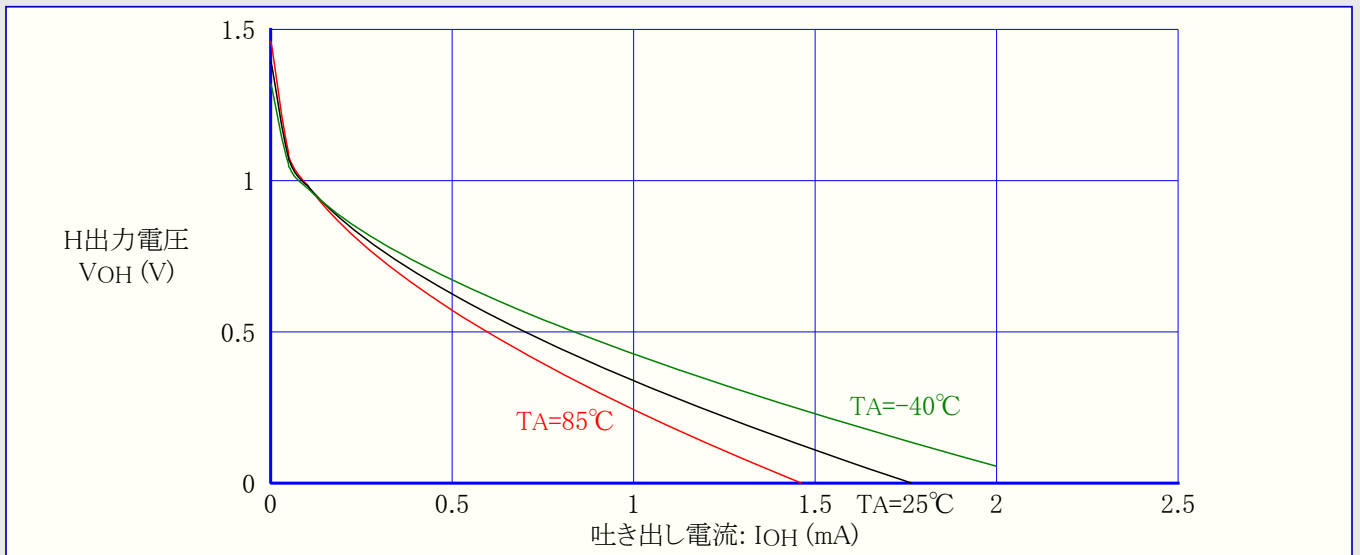


図23-38. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=3V)

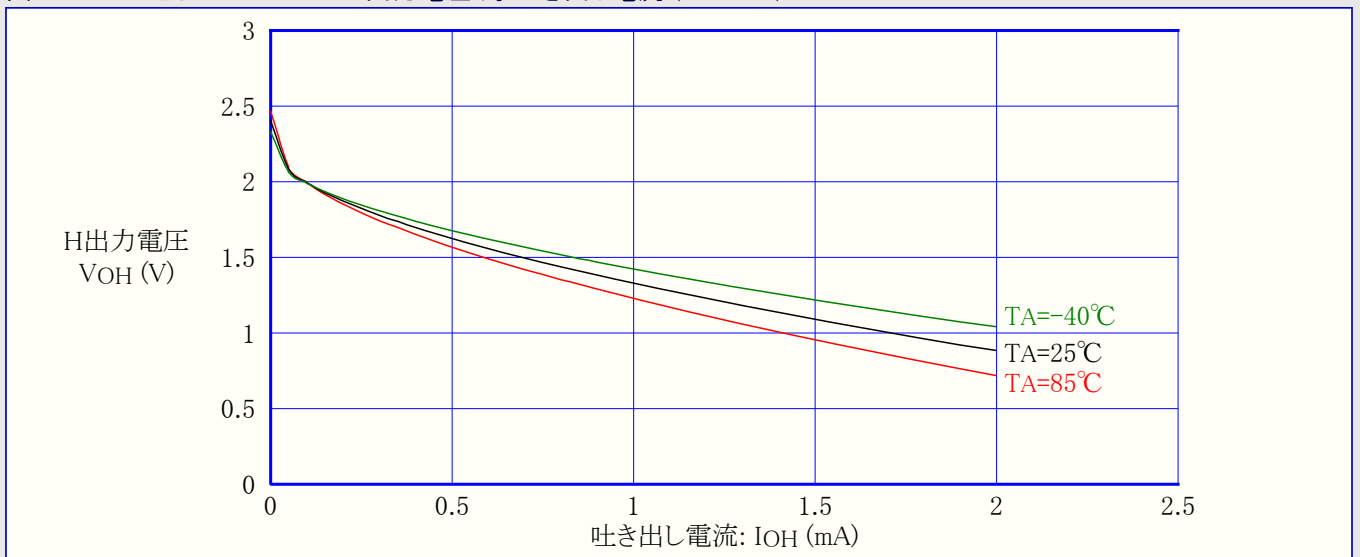
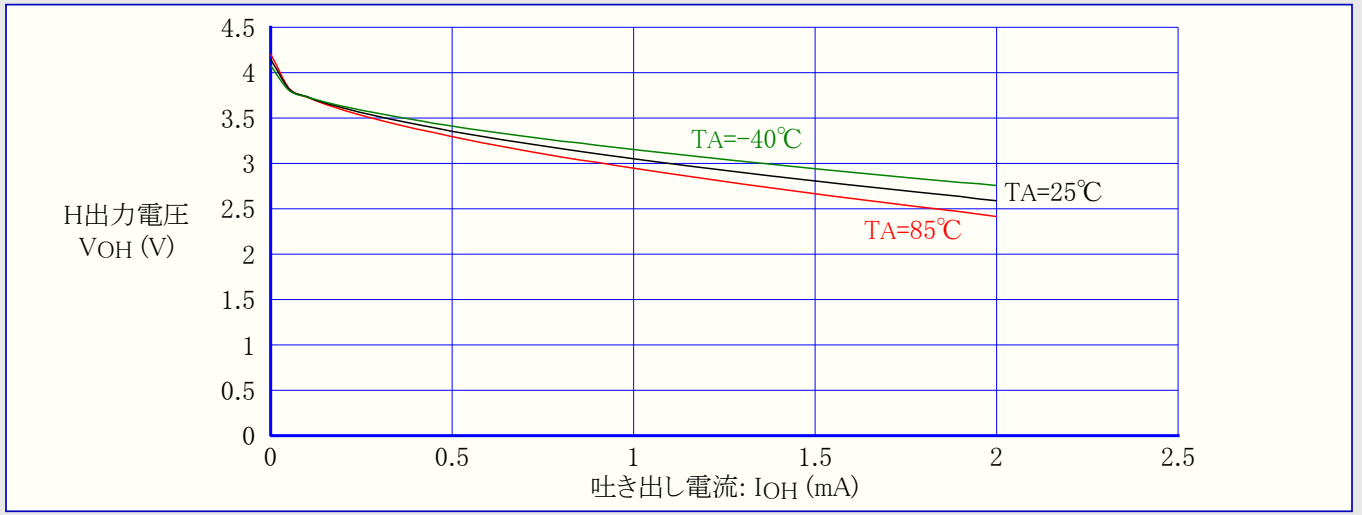


図23-39. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=5V)



23.1.8. 入力閾値とヒステリシス

図23-40. I/Oピン入力閾値(スレッシュホルツ)電圧 対 動作電圧 (VIH, 1読み値)

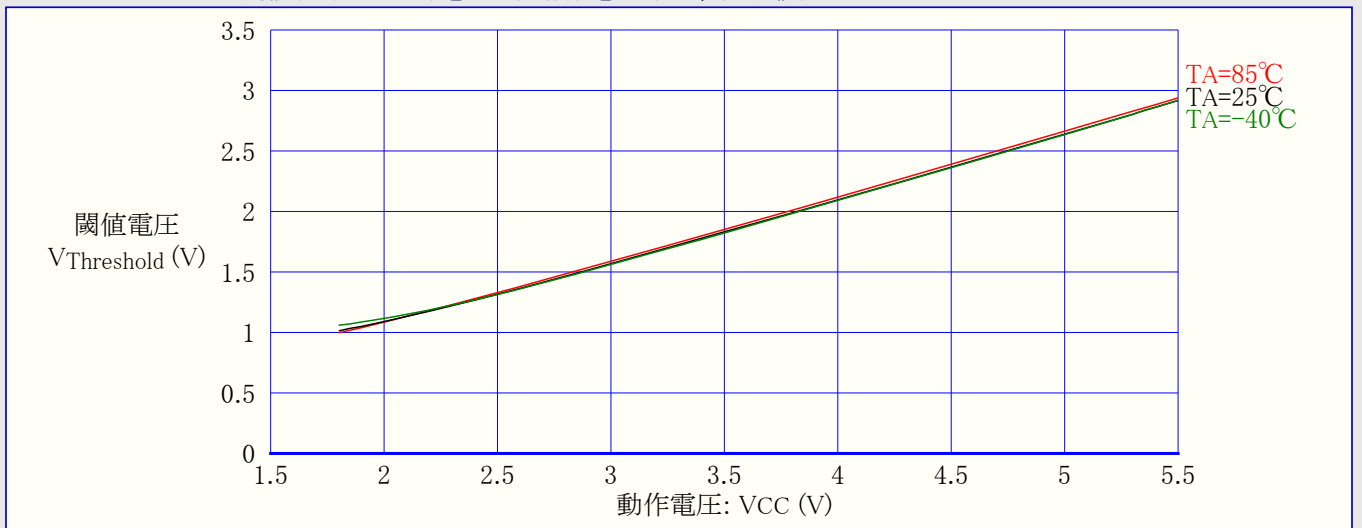


図23-41. I/Oピン入力閾値(スレッシュホルツ)電圧 対 動作電圧 (VIL, 0読み値)

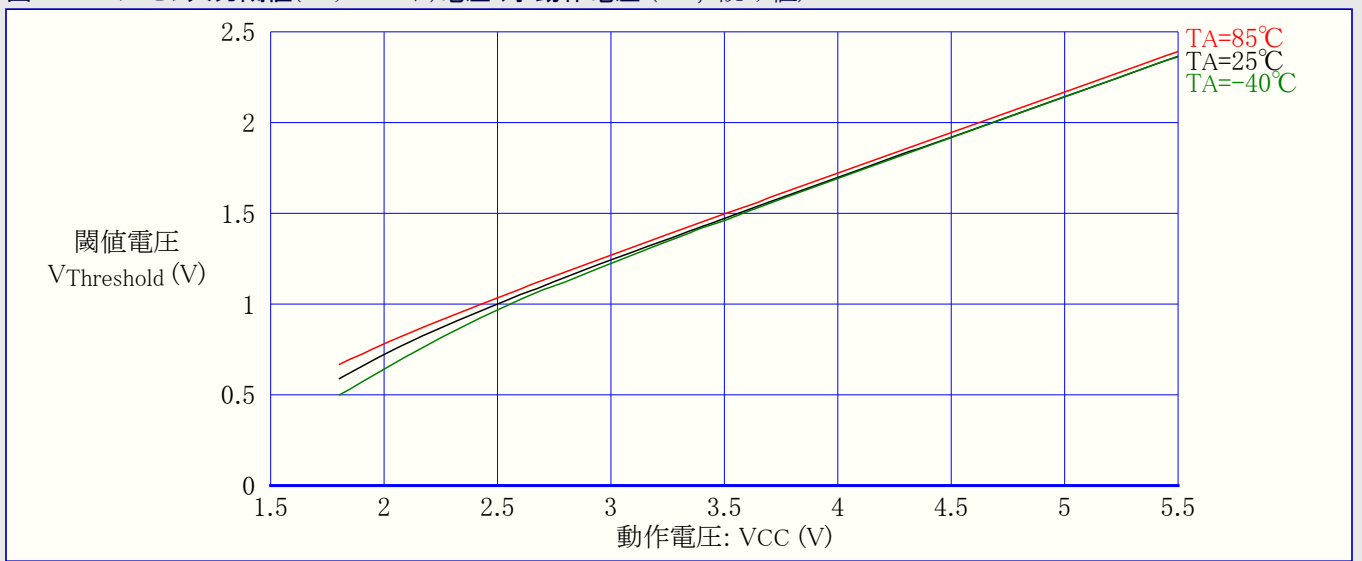


図23-42. I/Oピン入力ヒステリシス電圧 対 動作電圧

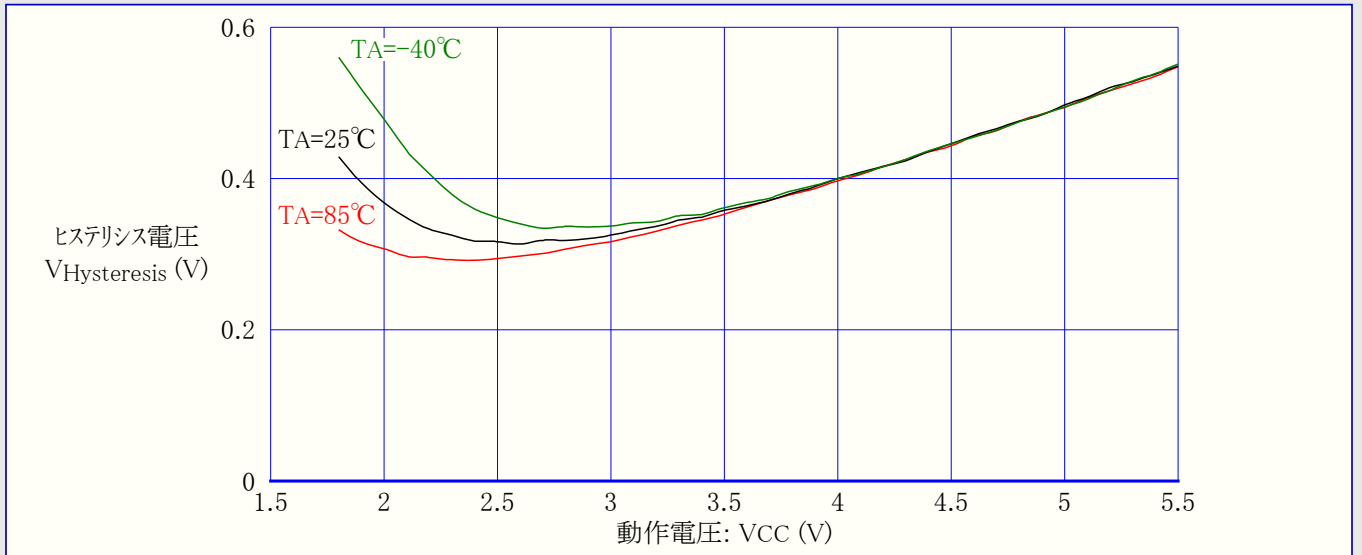


図23-43. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

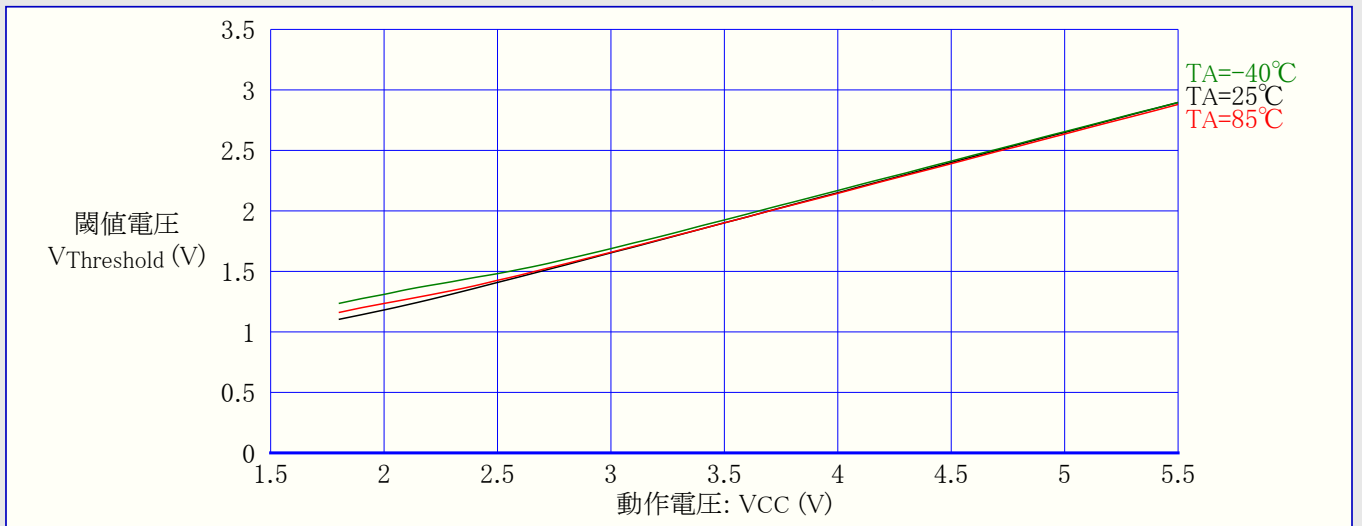


図23-44. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

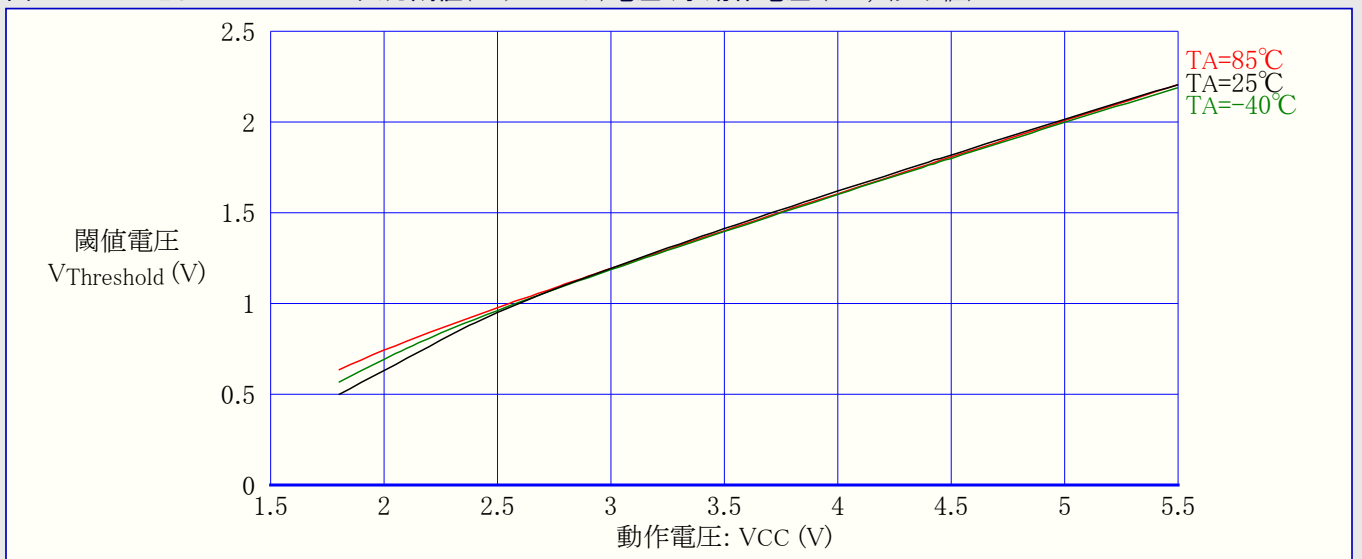
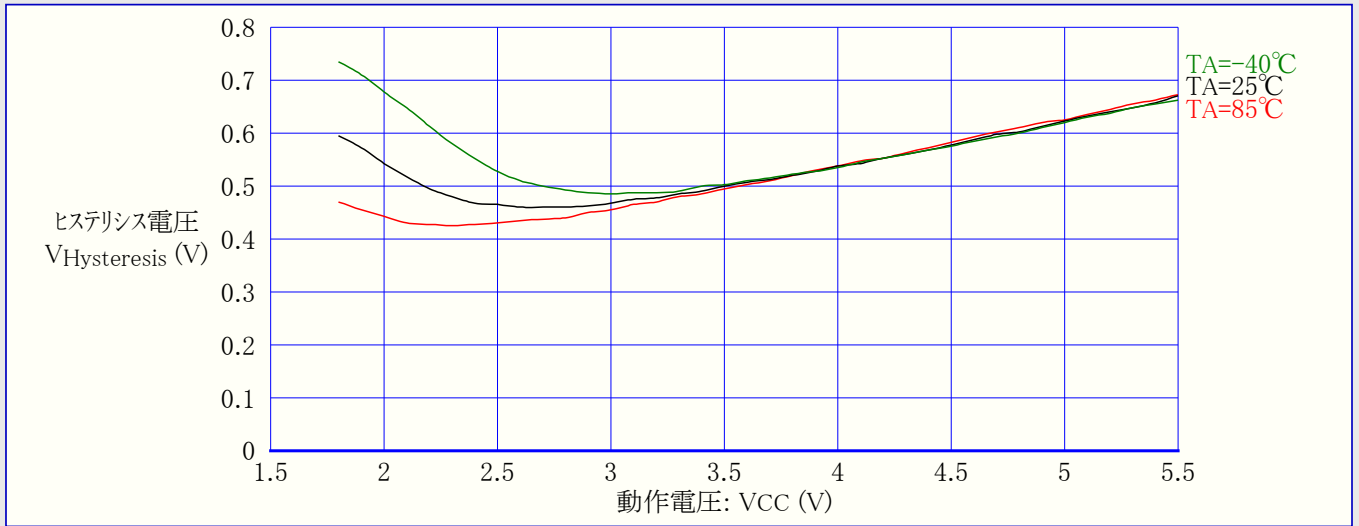


図23-45. I/OとしてのRESETピン入力ヒステリシス電圧 対 動作電圧



23.1.9. 低電圧検出器(BOD)、バンドキャップ、リセット

図23-46. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧4.3V)

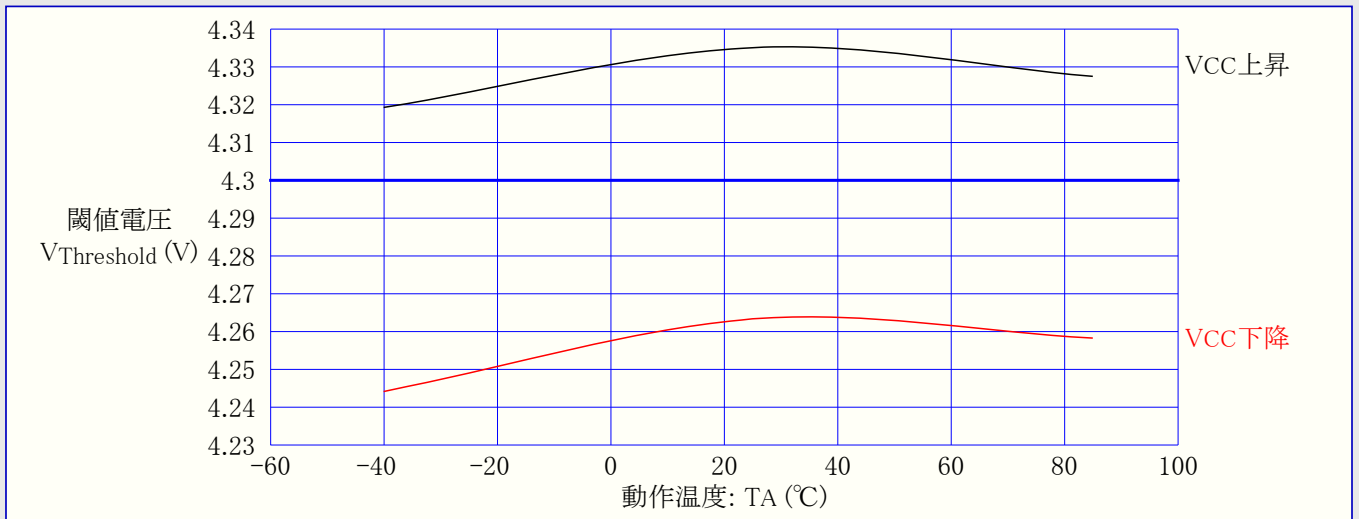


図23-47. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)

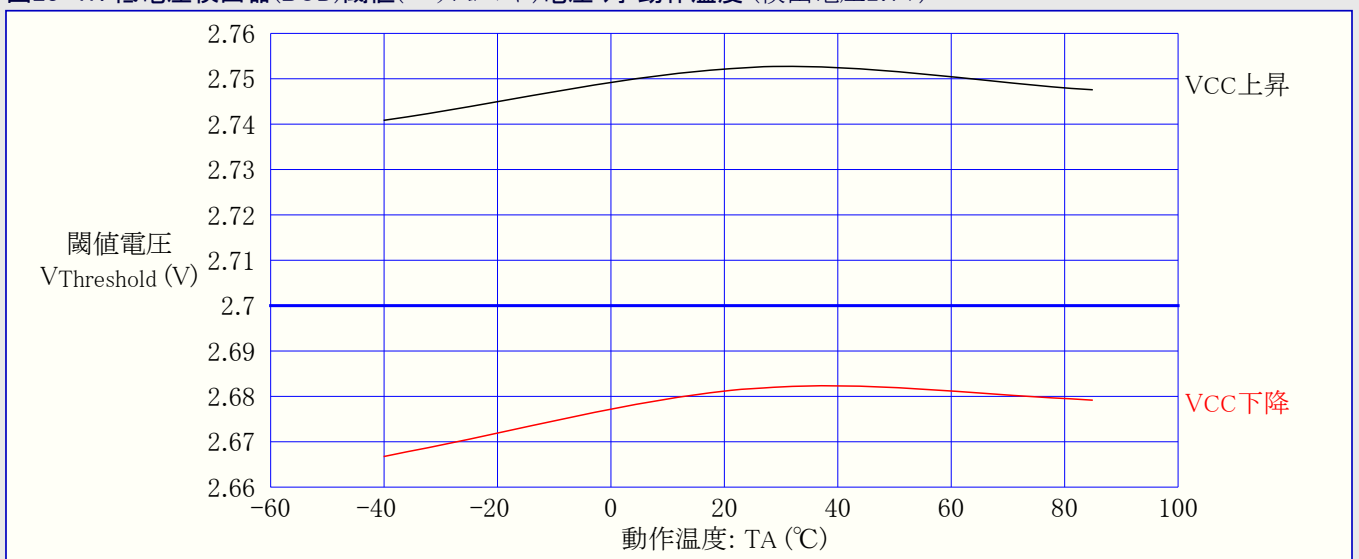


図23-48. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧1.8V)

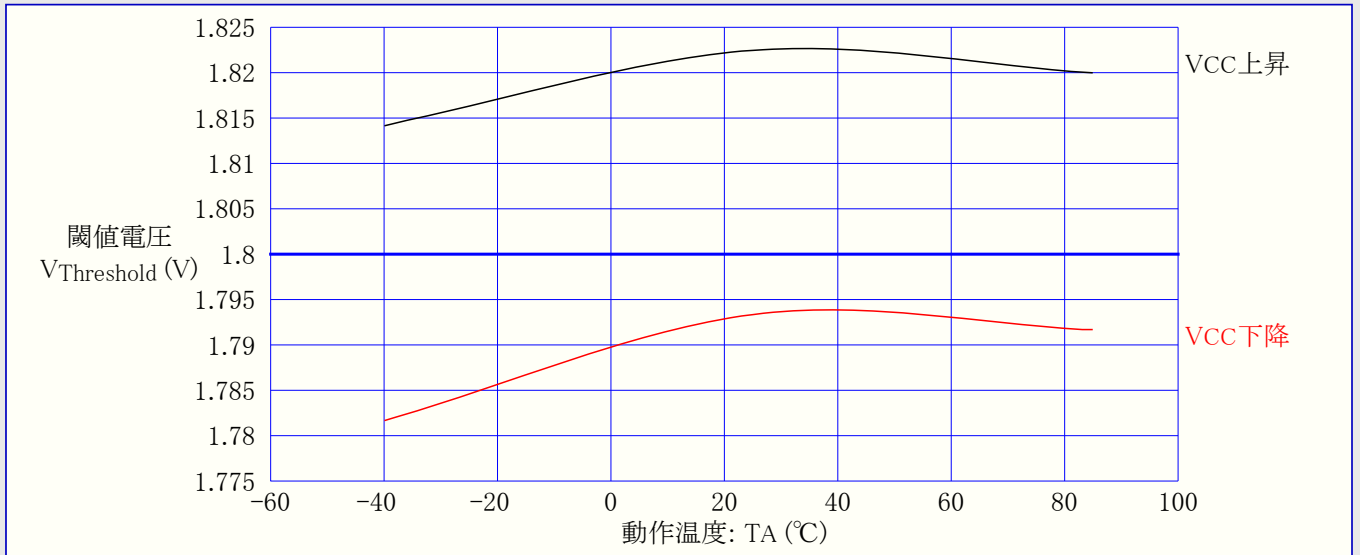


図23-49. RESETピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (VIH,1読み値)

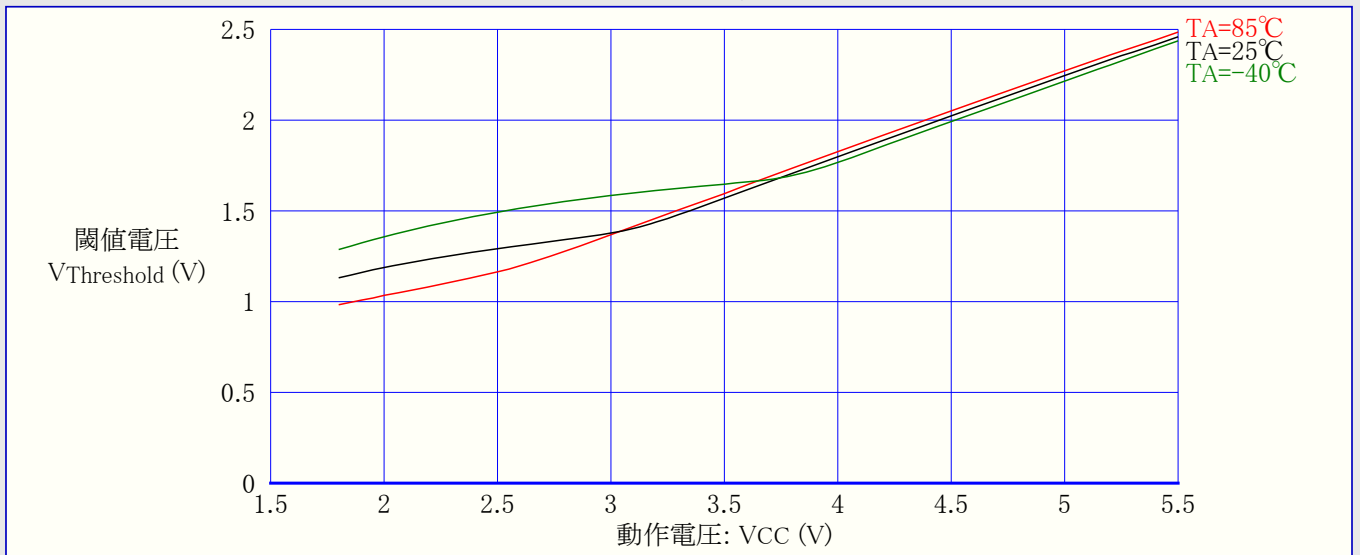


図23-50. RESETピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (VIL,0読み値)

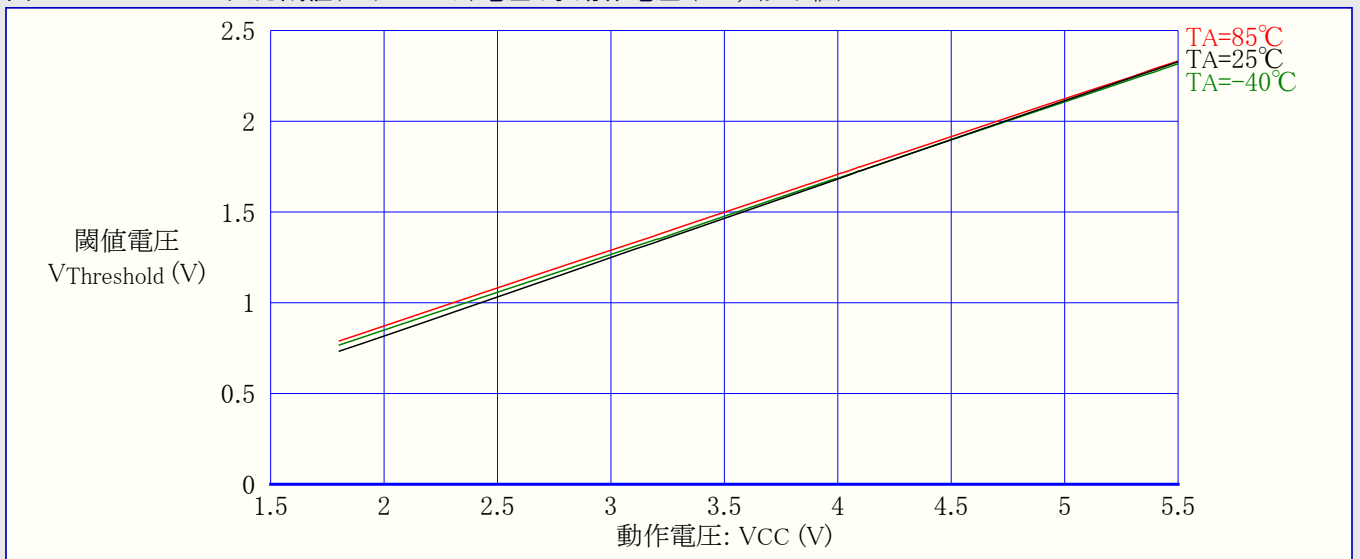


図23-51. I/OとしてのRESETピン入力ヒステリシス電圧 対 動作電圧

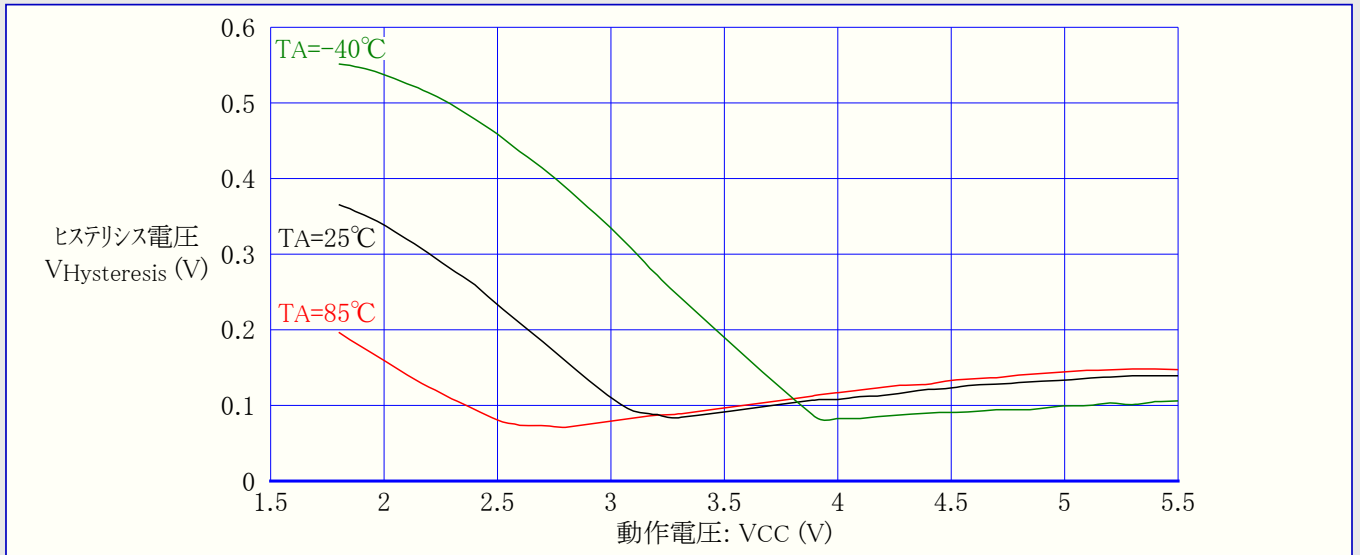
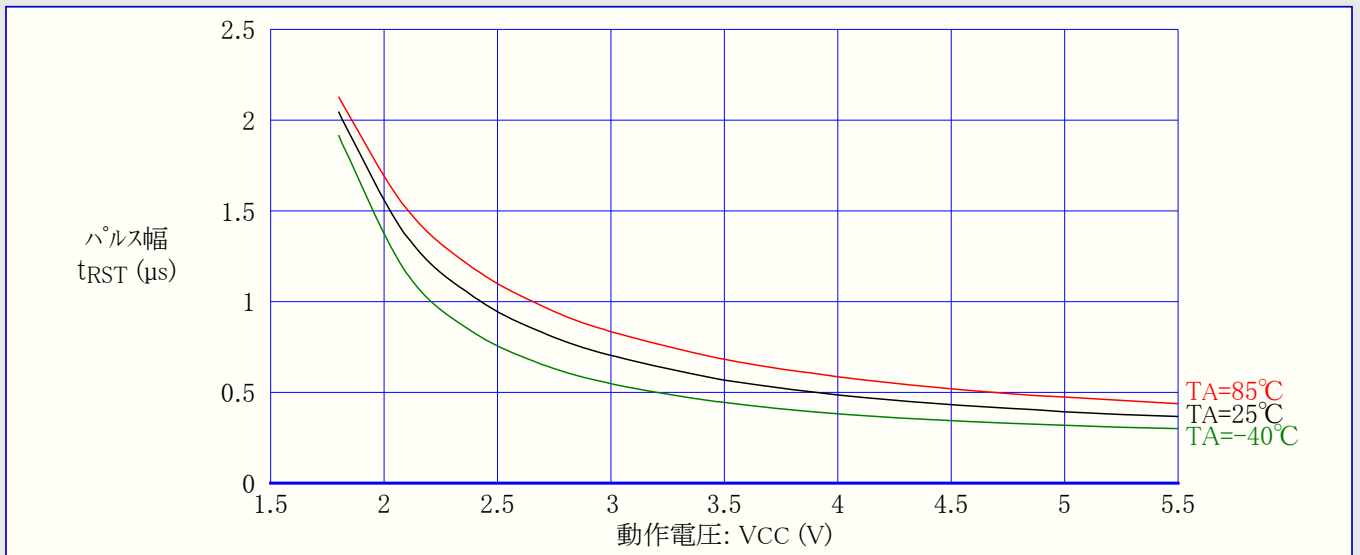


図23-52. 最小リセットパルス幅 対 動作電圧



23.1.10. 内部発振器周波数

図23-53. 128kHzウォッチドッグ発振器周波数 対 動作電圧

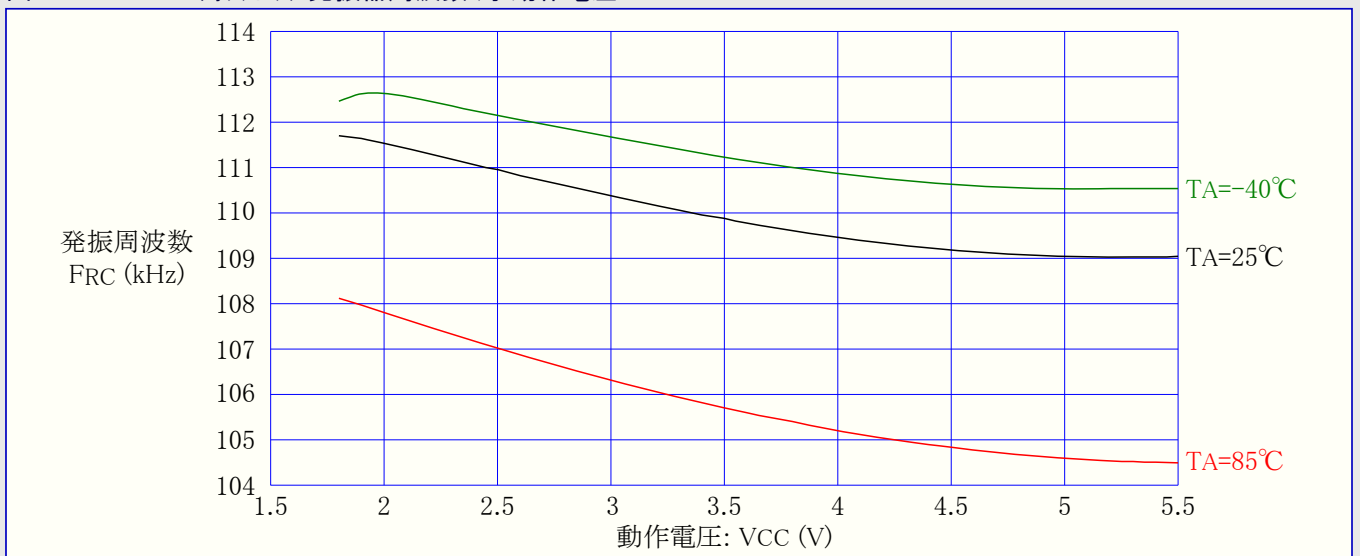


図23-54. 128kHzウオッチバック発振器周波数 対 動作温度

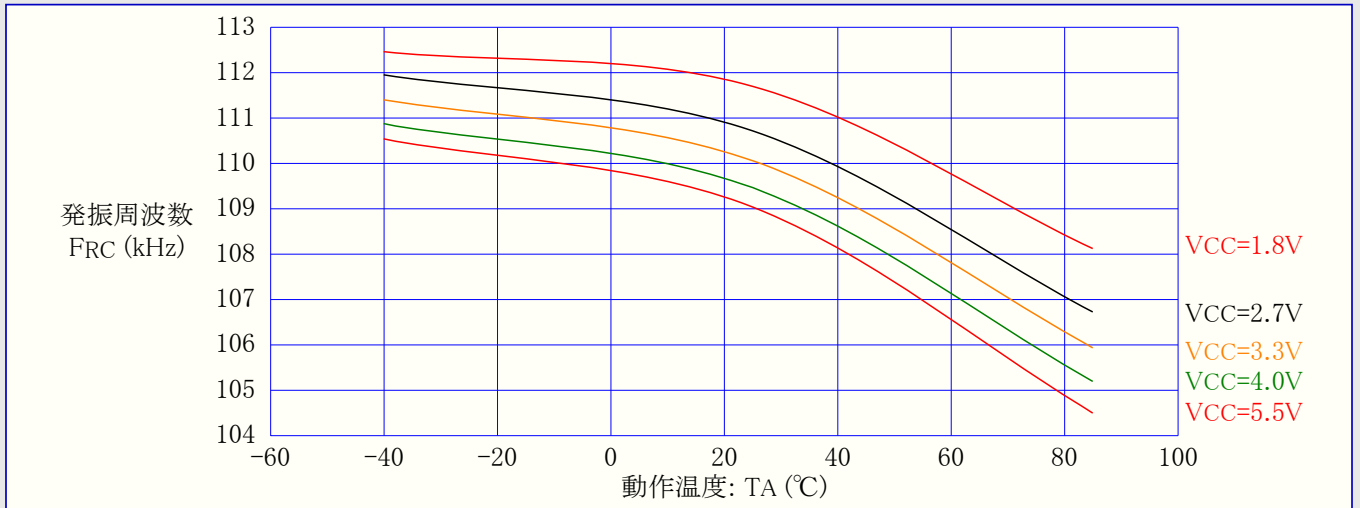


図23-55. 校正済み8MHz内蔵発振器周波数 対 動作電圧

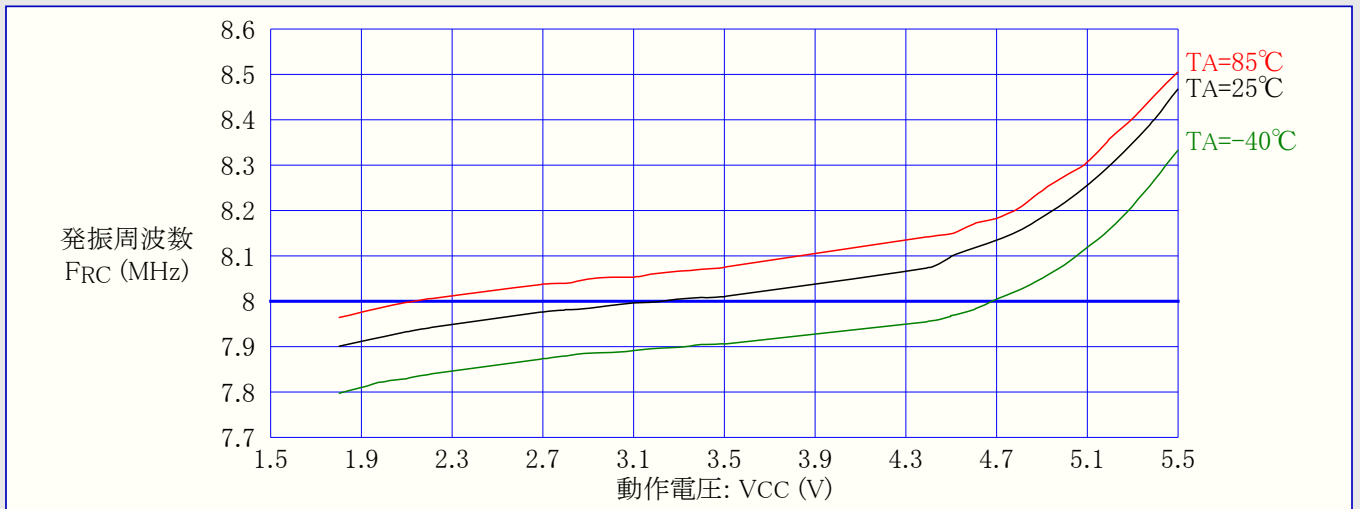


図23-56. 校正済み8MHz内蔵発振器周波数 対 動作温度

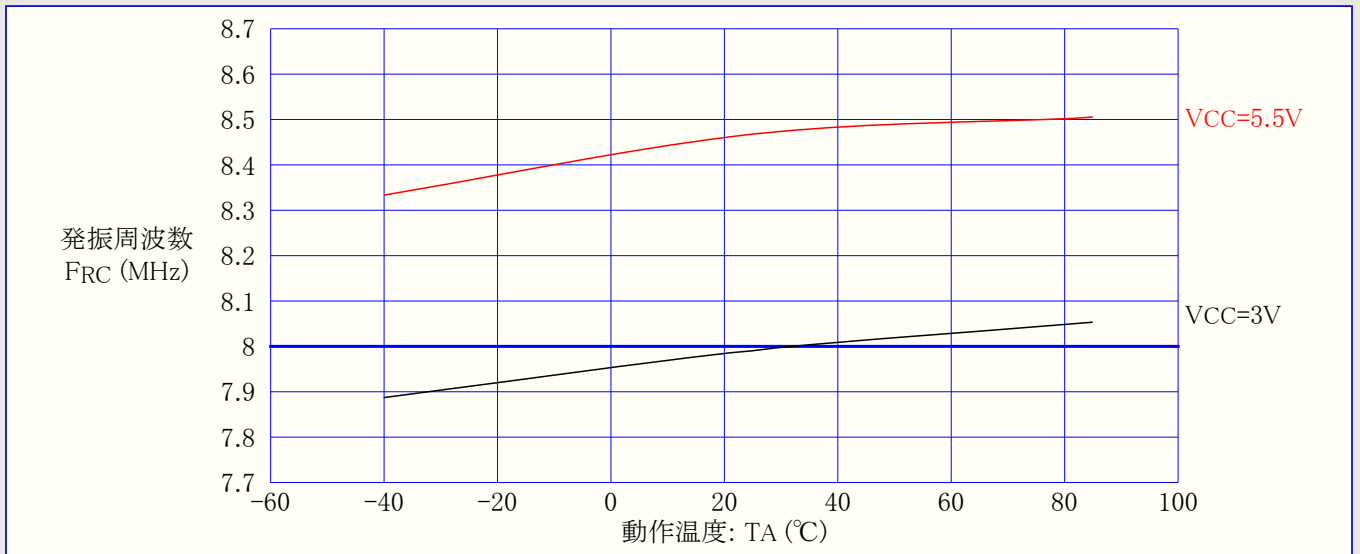
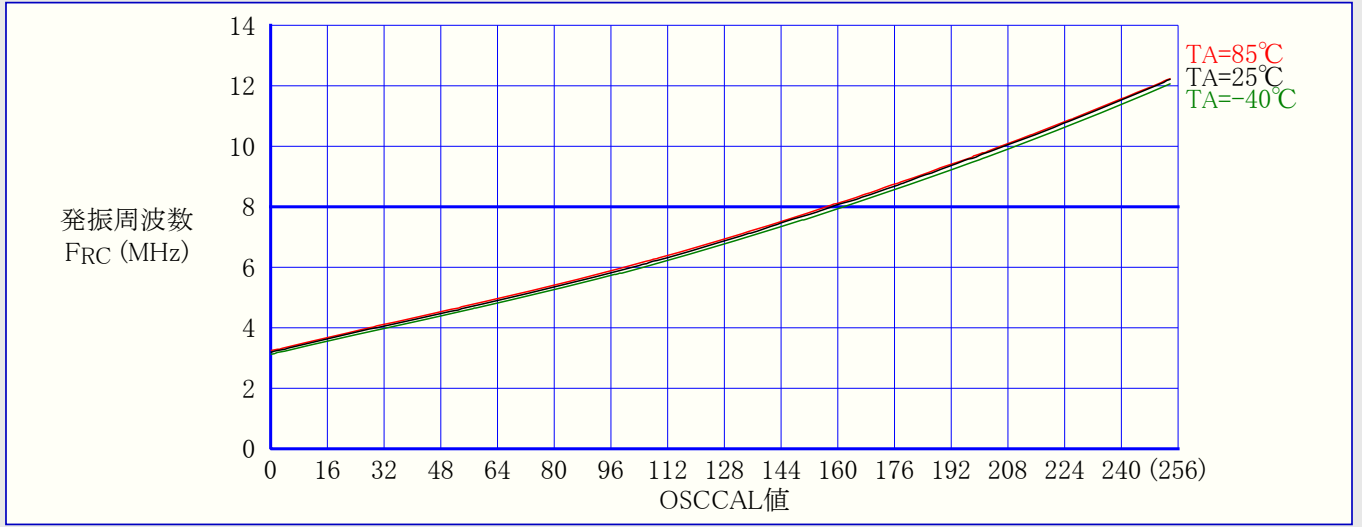


図23-57. 校正付き8MHz内蔵発振器周波数 対 発振校正(OSCCAL)値



## 23.2. ATtiny88

### 23.2.1. 活動動作消費電流

図23-58. 活動動作消費電流 対 周波数 (100kHz~1MHz)

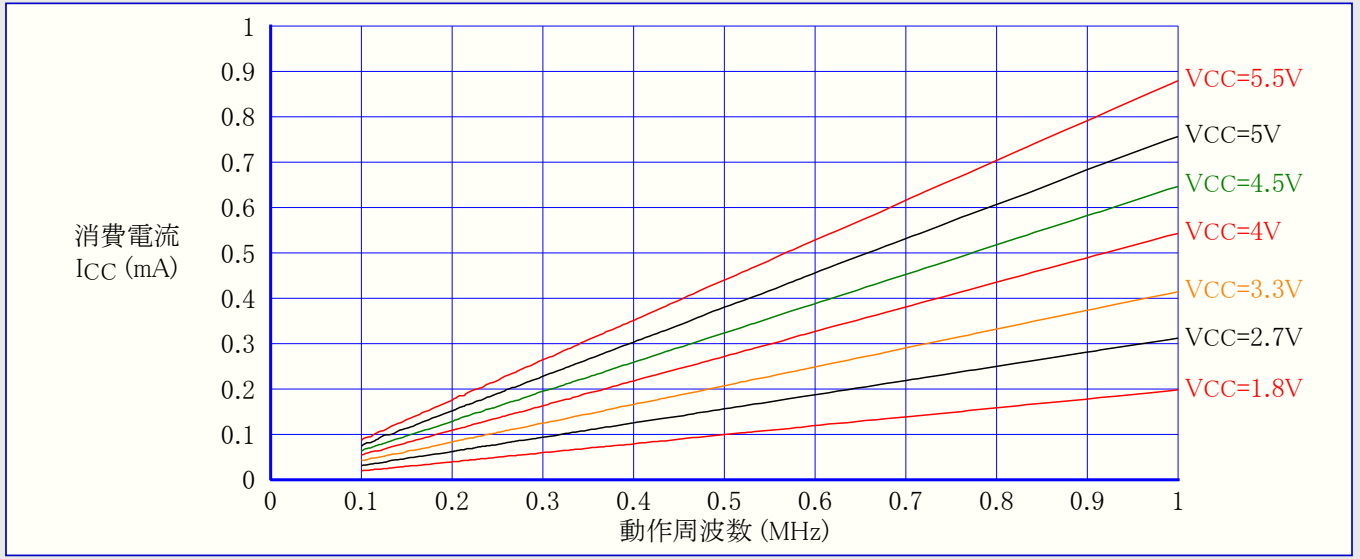


図23-59. 活動動作消費電流 対 周波数 (1MHz~12MHz)

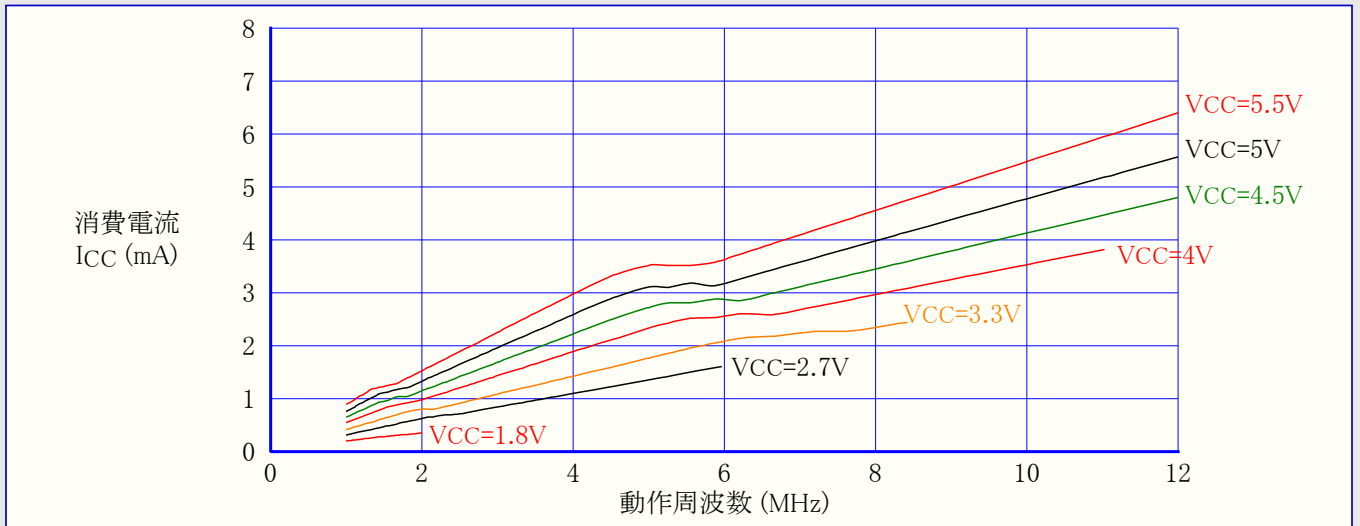


図23-60. 活動動作消費電流 対 動作電圧 (内蔵発振器, 8MHz)

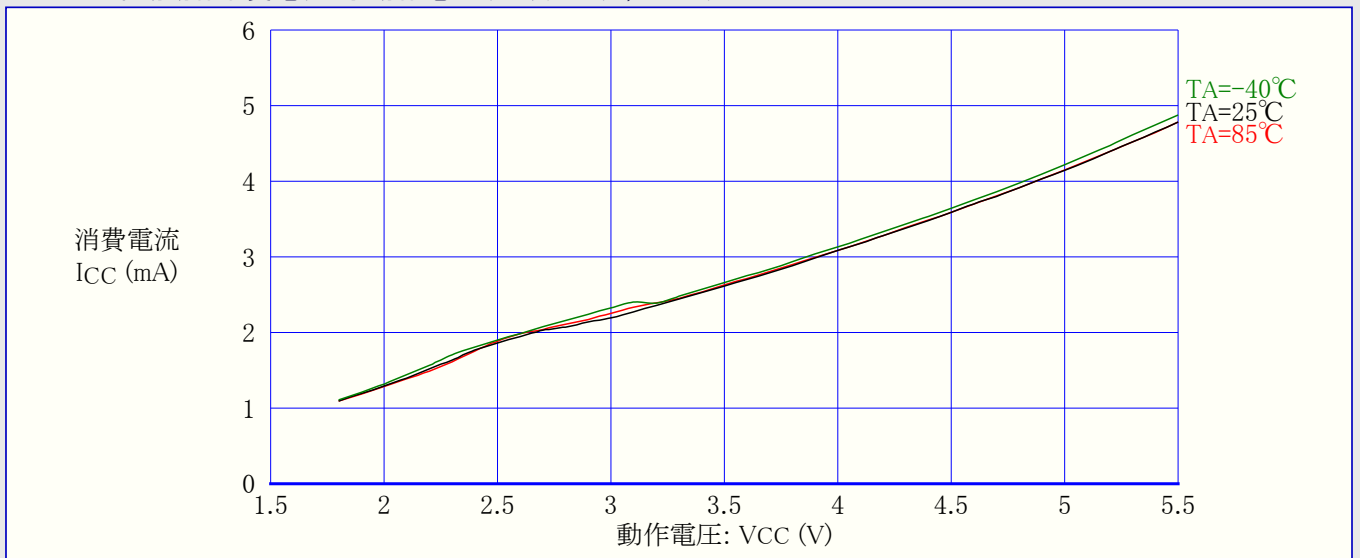


図23-61. 活動動作消費電流 対 動作電圧 (内蔵発振器, 1MHz)

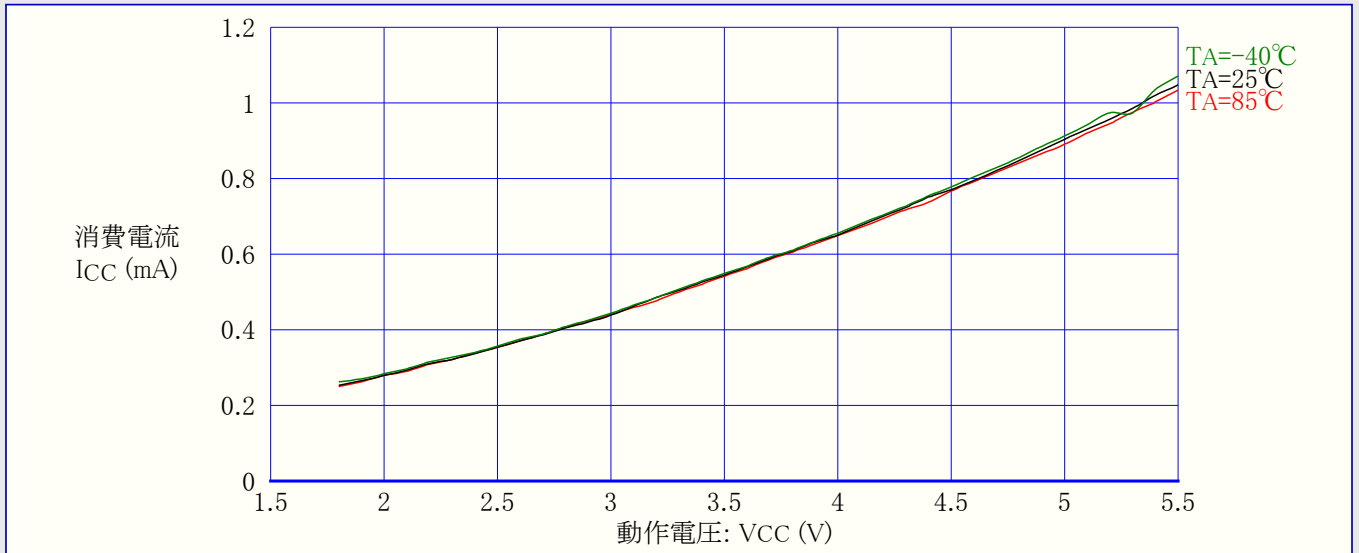
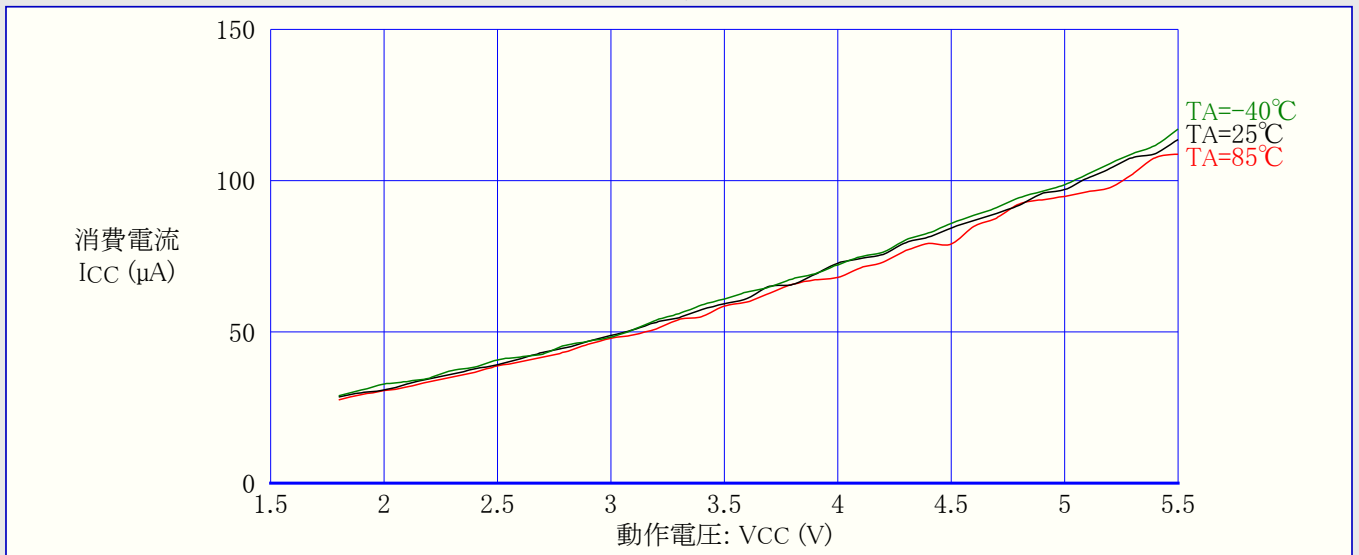


図23-62. 活動動作消費電流 対 動作電圧 (内蔵WDT発振器, 128kHz)



23.2.2. アイドル動作消費電流

図23-63. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

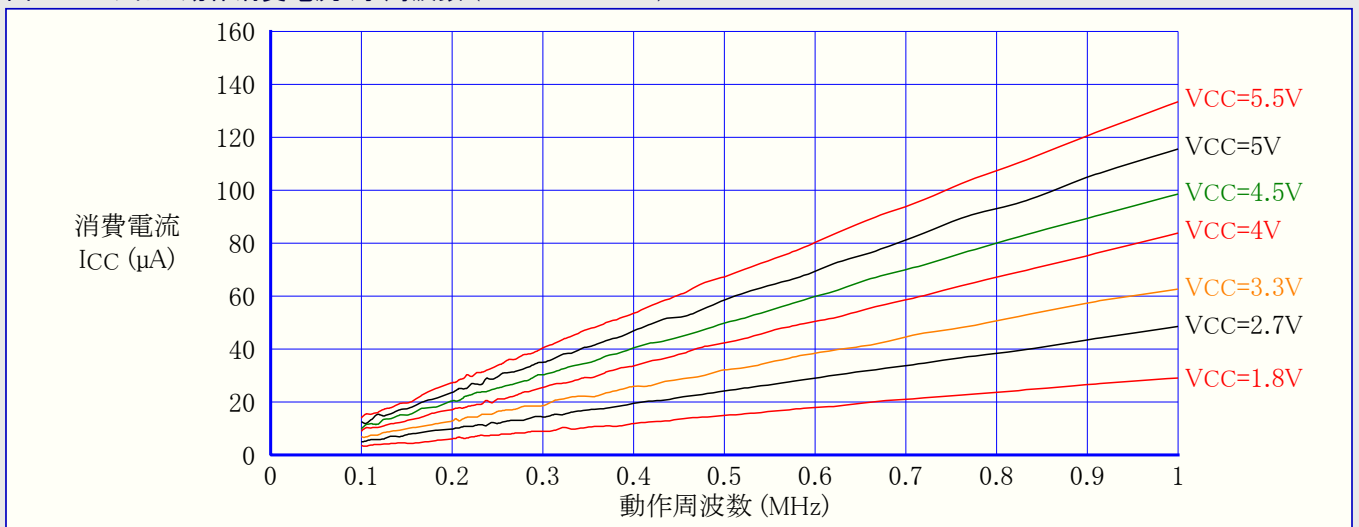


図23-64. アトル動作消費電流 対 周波数 (1MHz~12MHz)

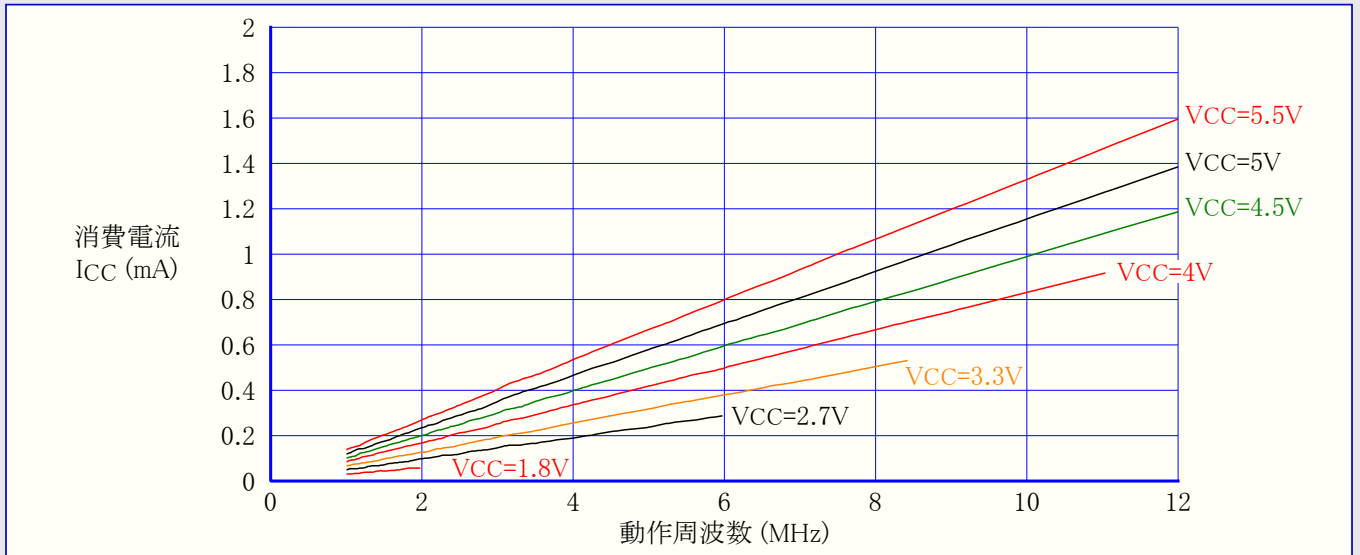


図23-65. アトル動作消費電流 対 動作電圧 (内蔵発振器, 8MHz)

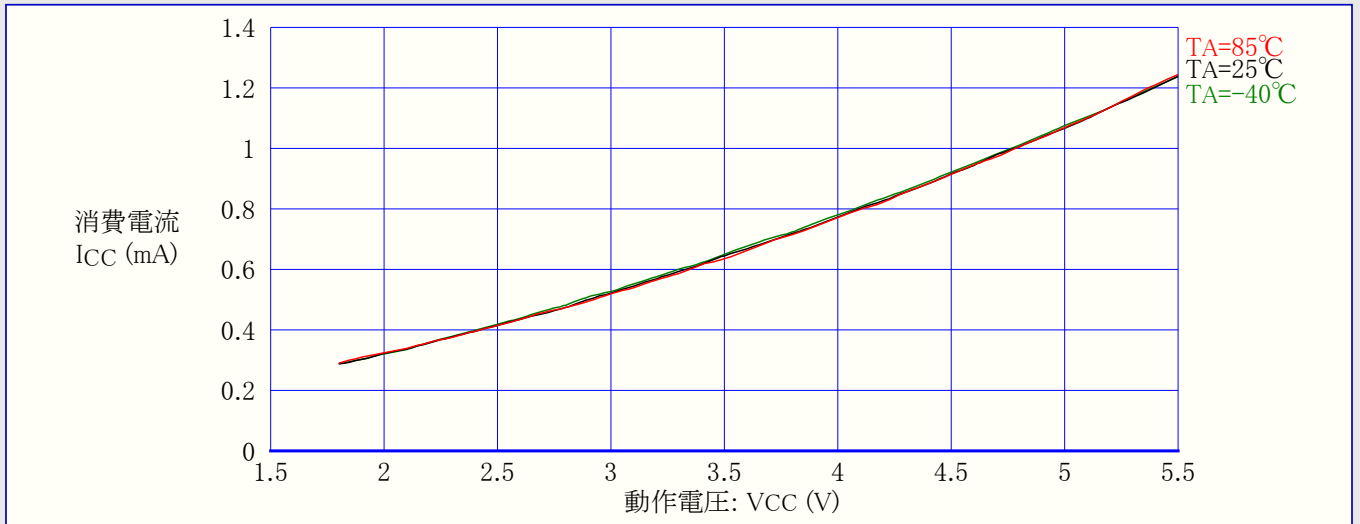


図23-66. アトル動作消費電流 対 動作電圧 (内蔵発振器, 1MHz)

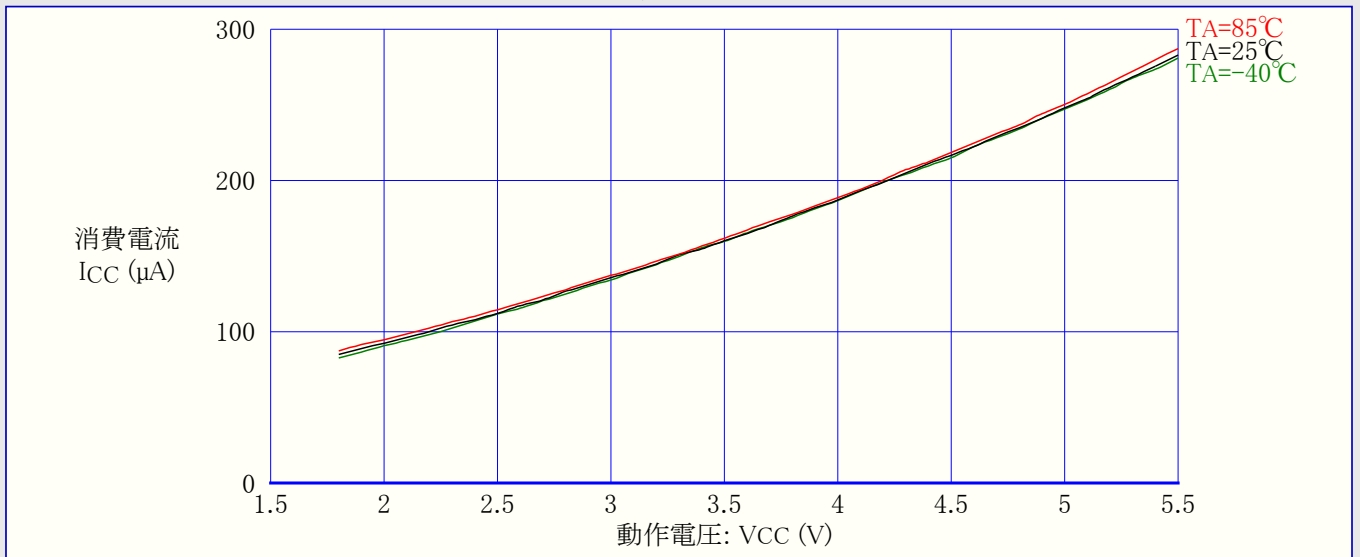
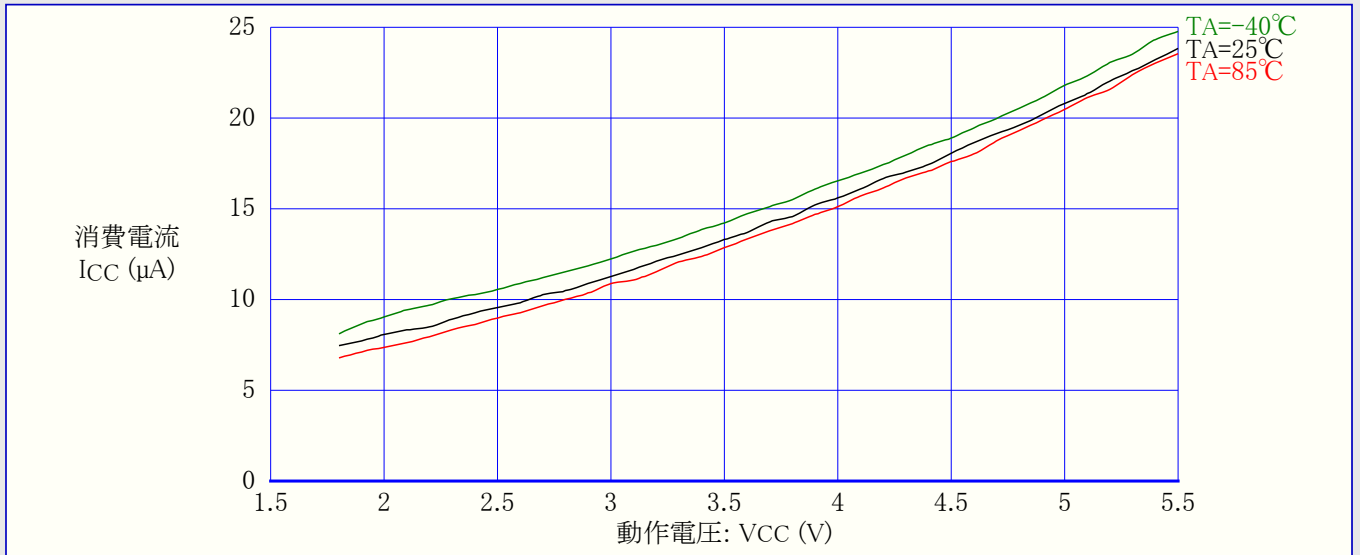


図23-67. アイドル動作消費電流 対 動作電圧 (内蔵WDT発振器, 128kHz)



23.2.3. パワーダウン動作消費電流

図23-68. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)

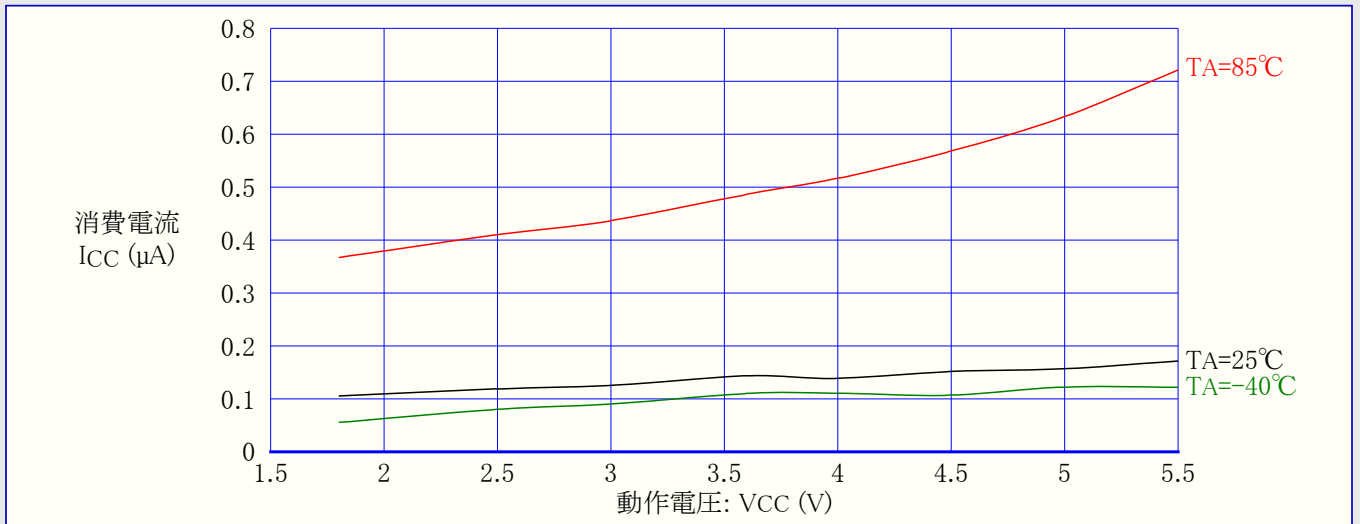
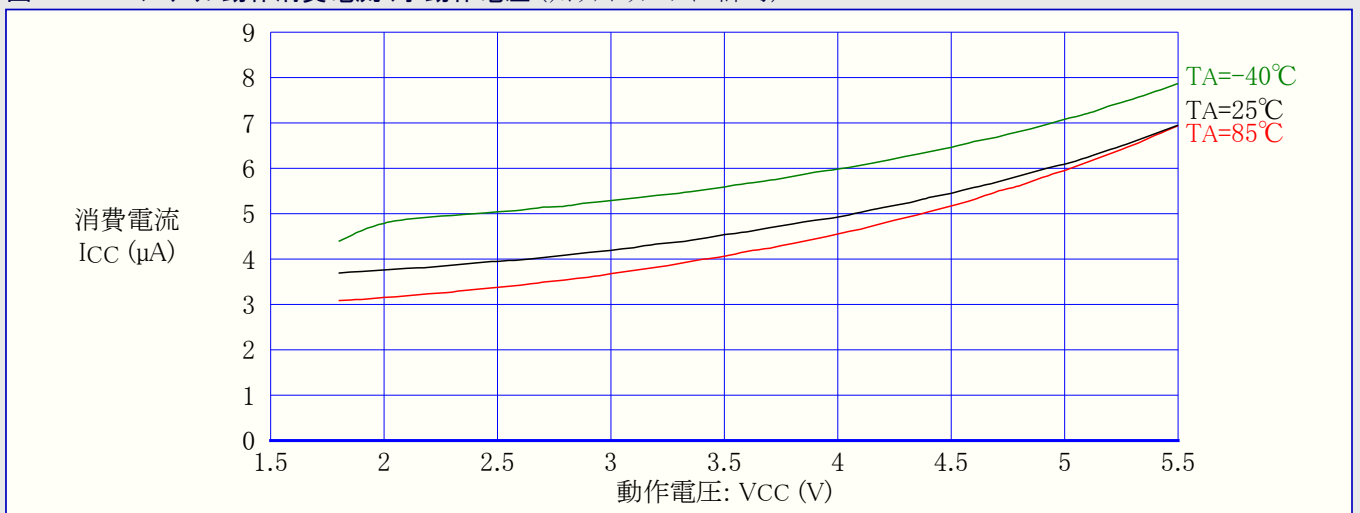


図23-69. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



### 23.2.4. リセット消費電流

図23-70. リセット消費(供給)電流 対 周波数 (100kHz~1MHz)

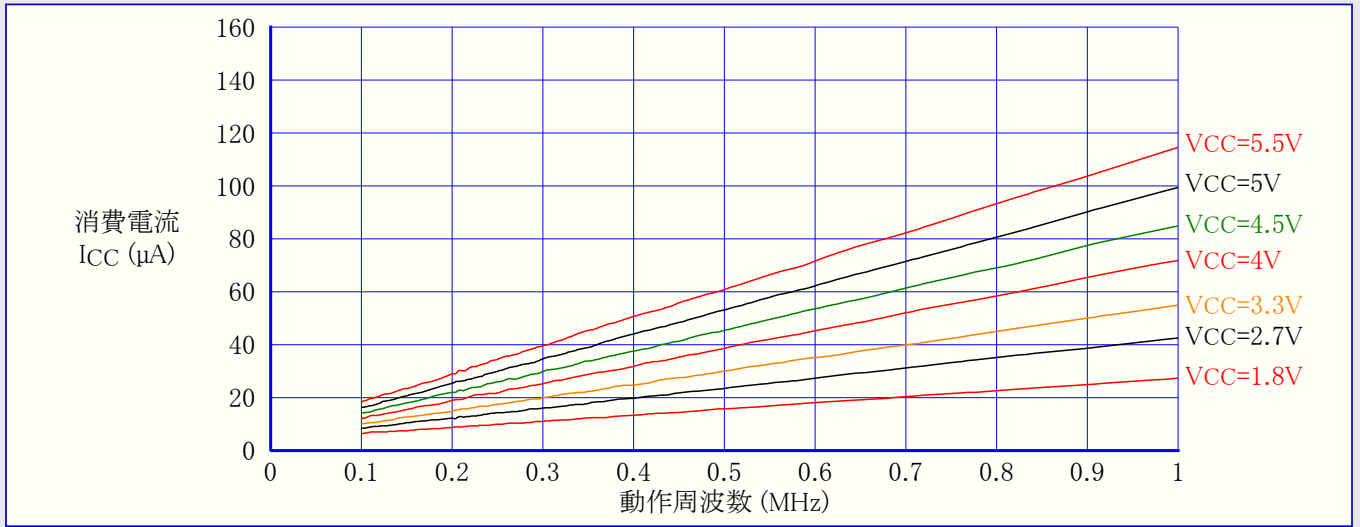
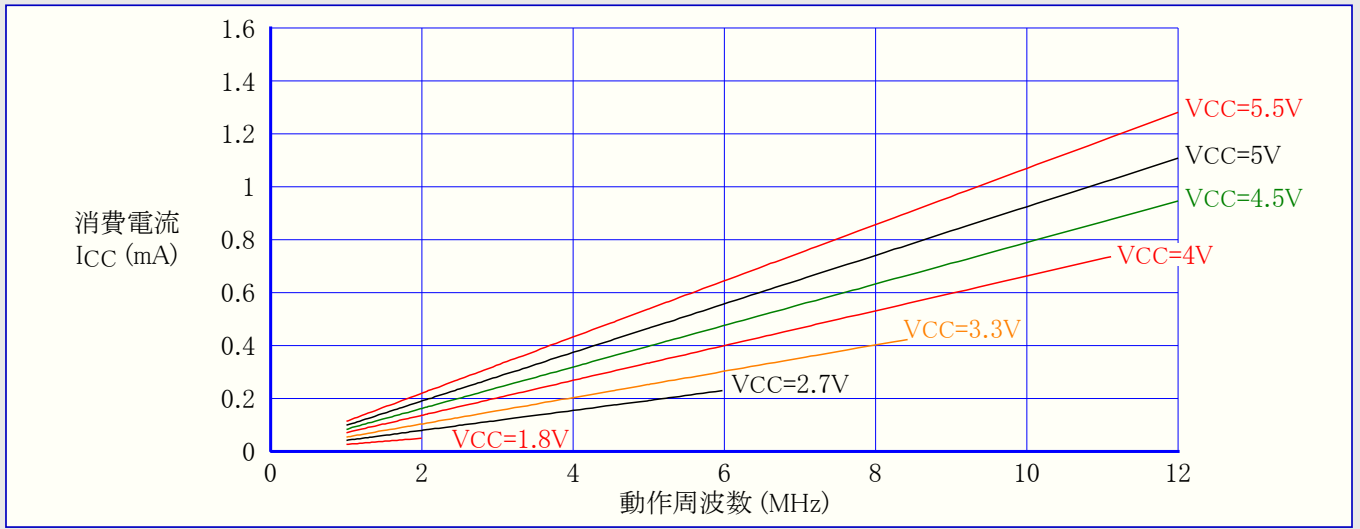


図23-71 リセット消費(供給)電流 対 周波数 (1MHz~12MHz)



### 23.2.5. 周辺機能部消費電流

図23-72. 低電圧検出器(BOD)消費電流 対 動作電圧

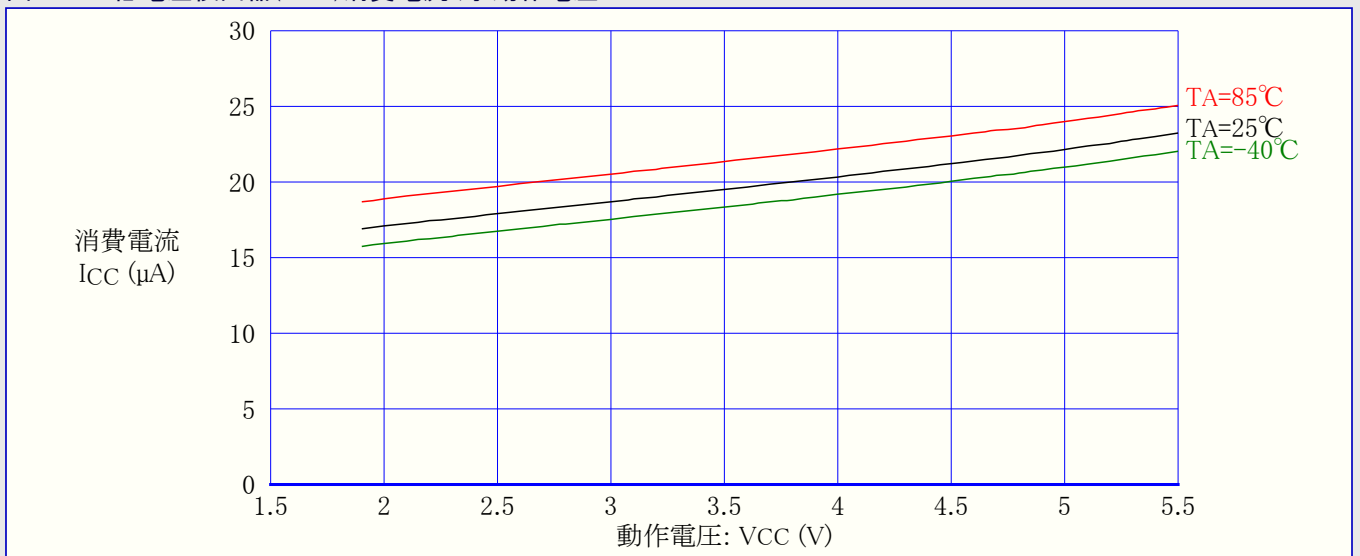


図23-73. A/D変換器消費電流 対 動作電圧 (AREF=AVCC)

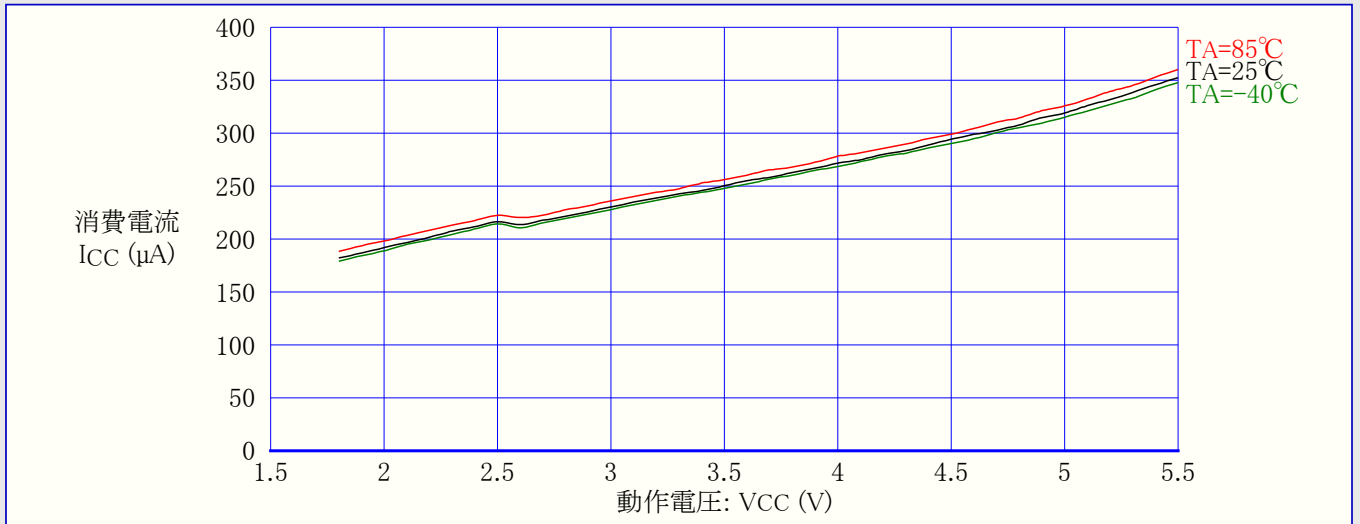


図23-74. アナログ比較器消費電流 対 動作電圧

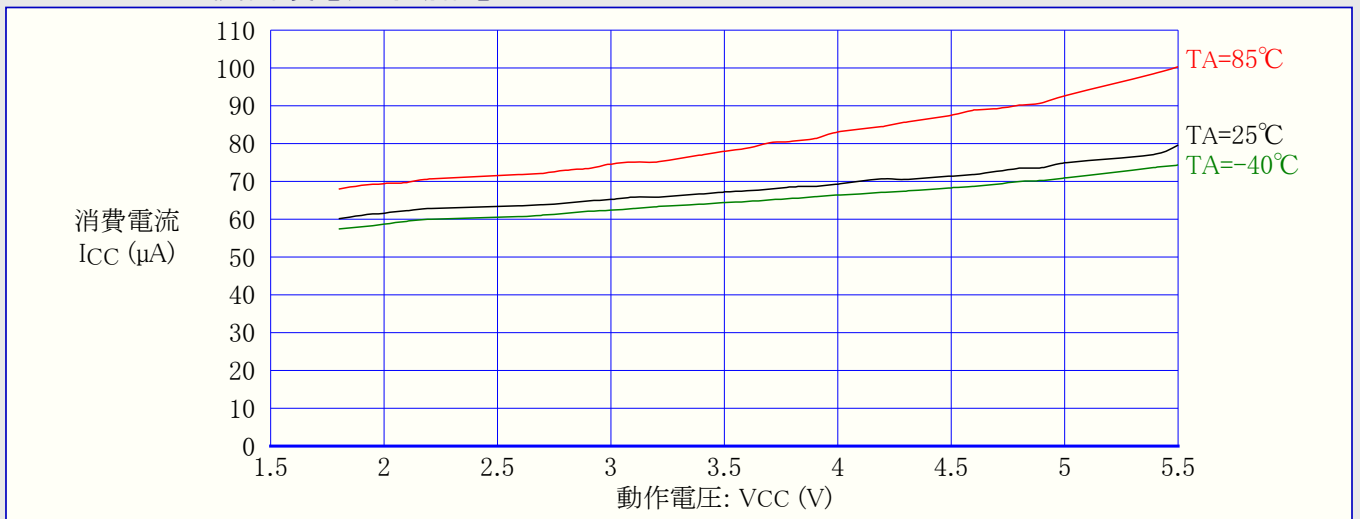
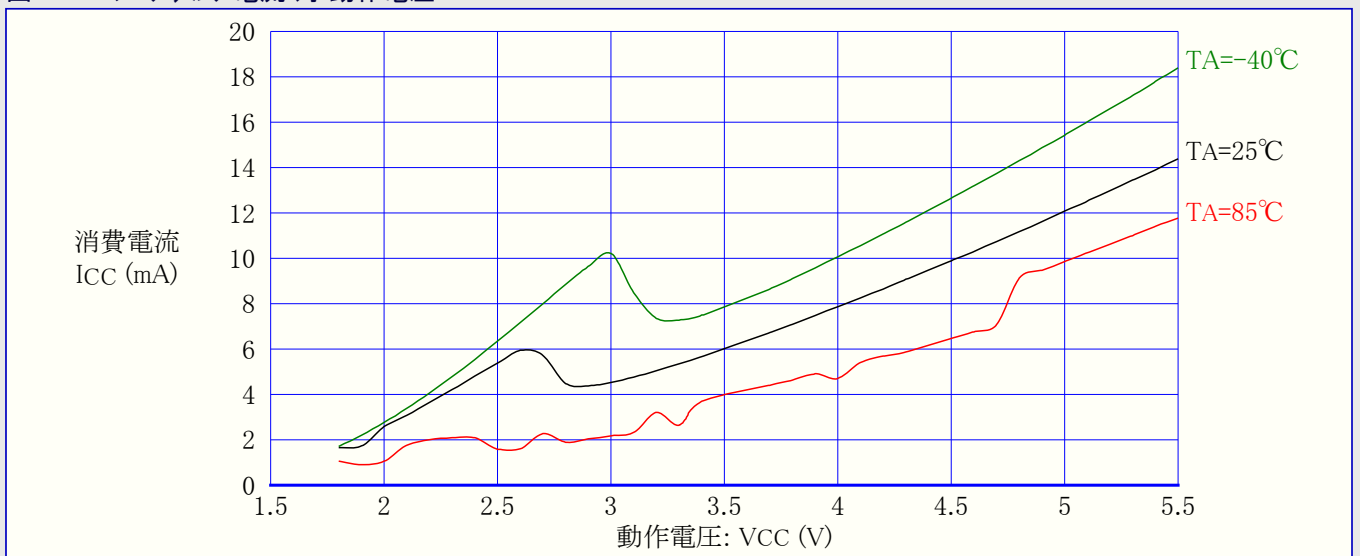


図23-75. フロタリング電流 対 動作電圧



### 23.2.6. プルアップ抵抗

図23-76. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

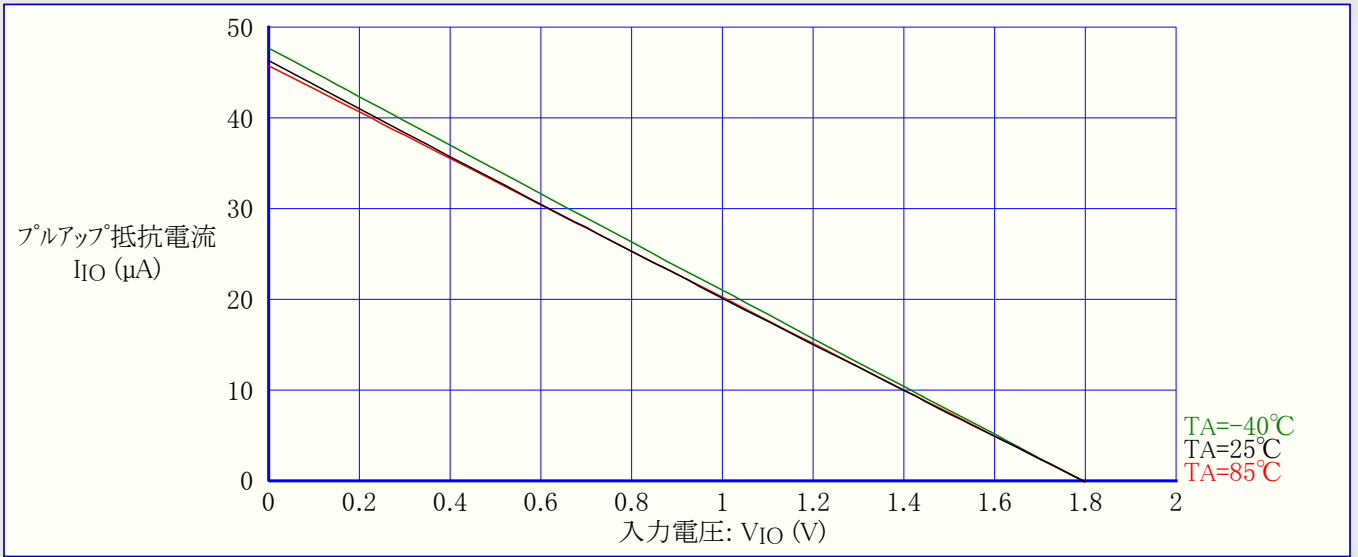


図23-77. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

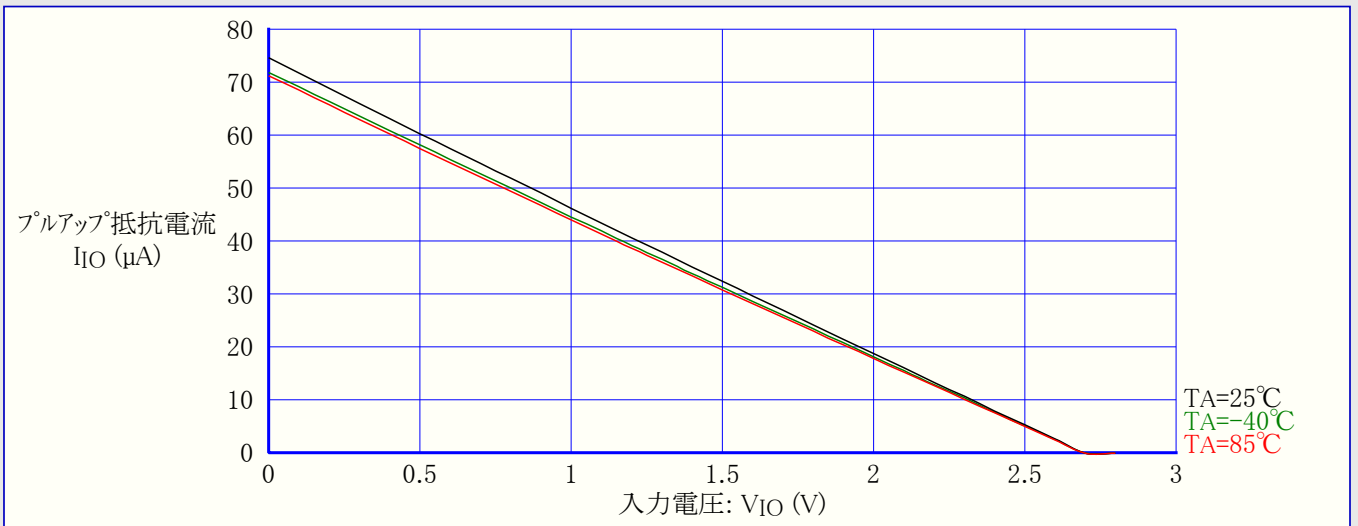


図23-78. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

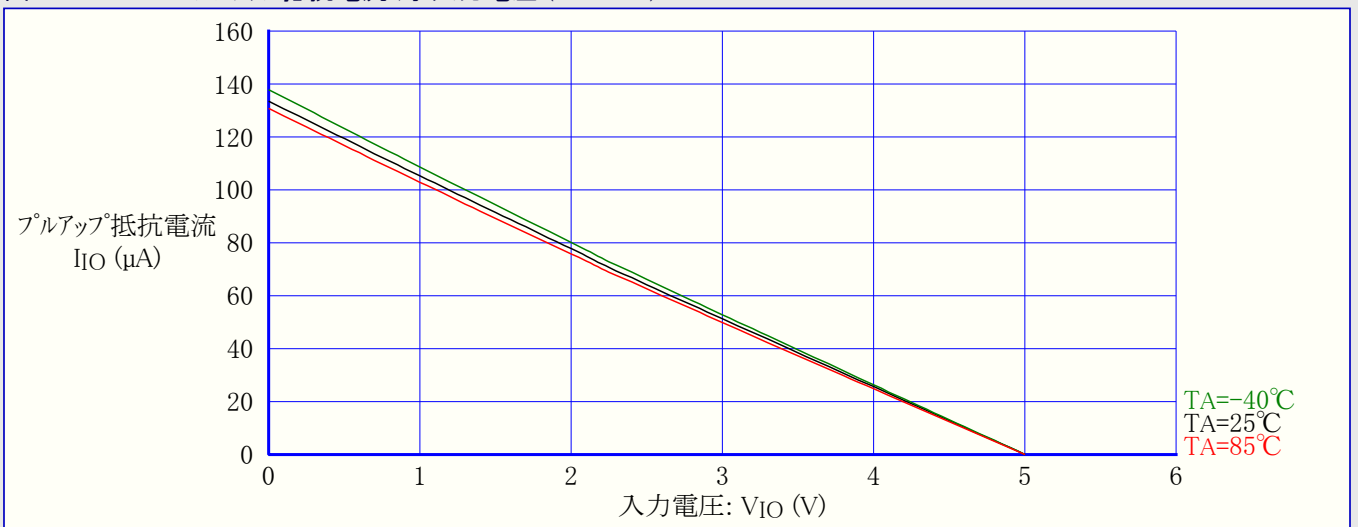


図23-79. RESETプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

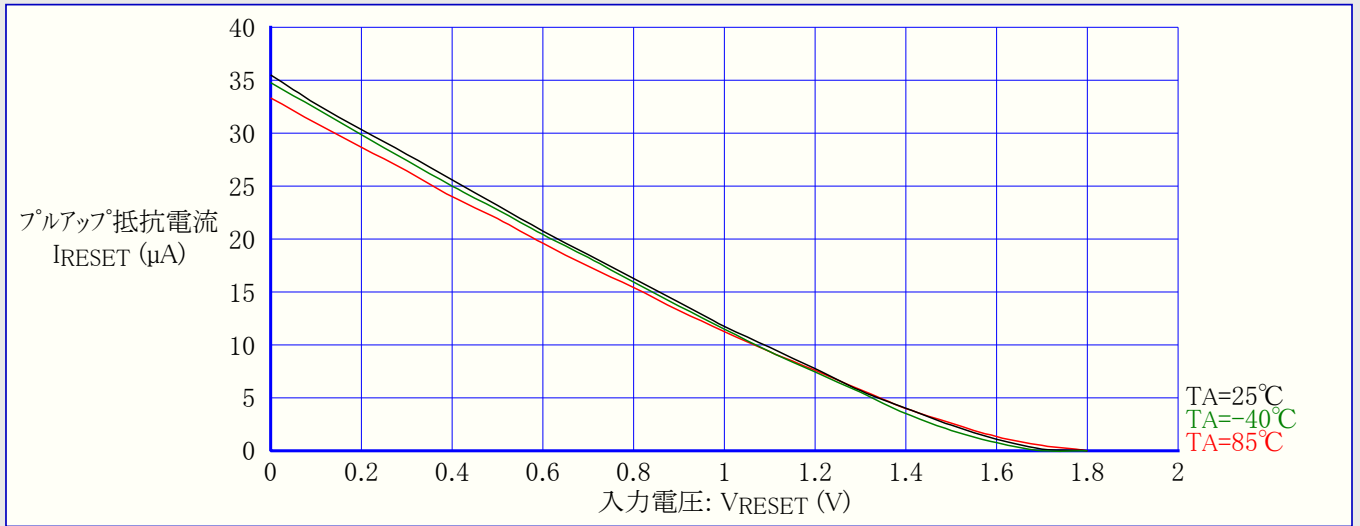


図23-80. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

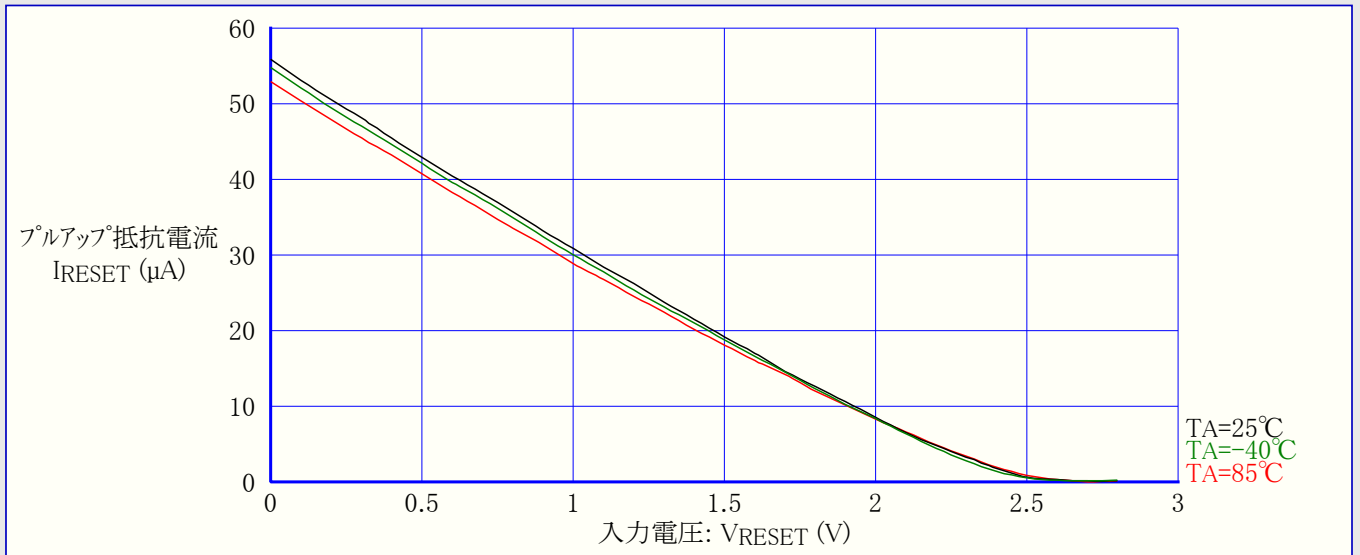
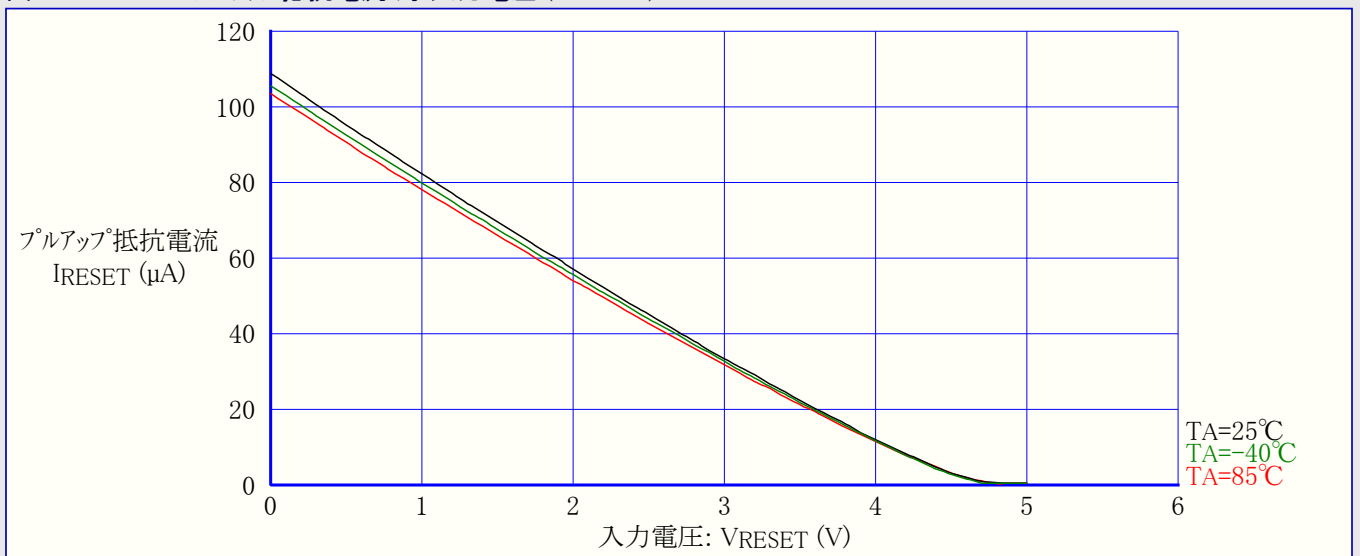


図23-81. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)



### 23.2.7. 出力駆動部能力

図23-82. 高シク駆動I/Oピン出力電圧対 吸い込み電流 (VCC=1.8V)

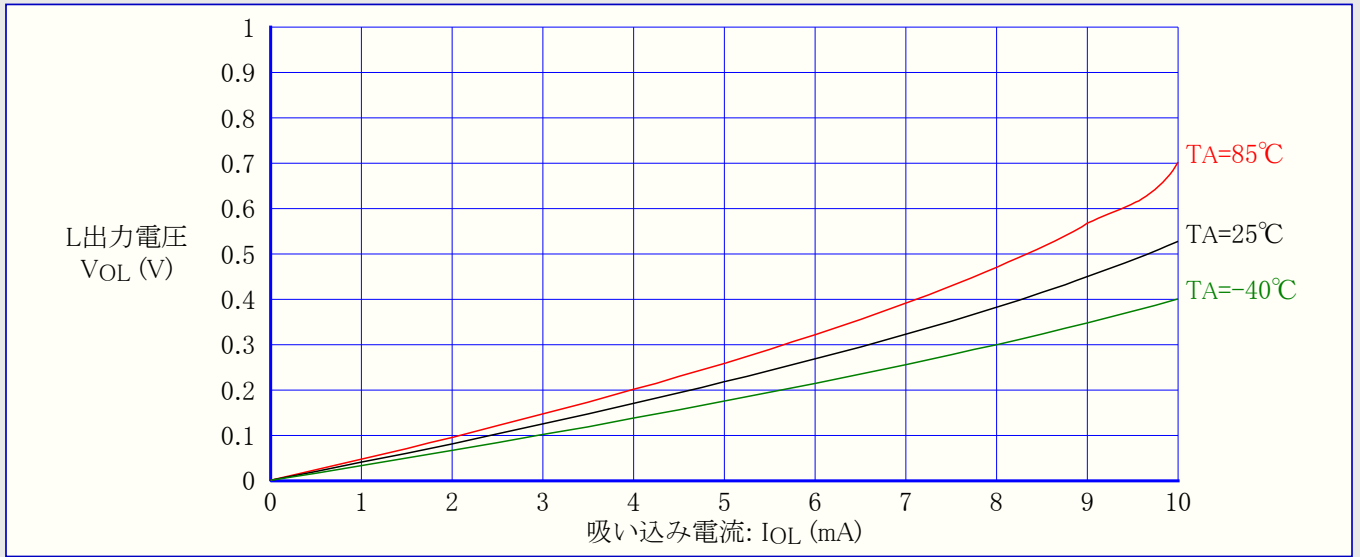


図23-83. 高シク駆動I/Oピン出力電圧対 吸い込み電流 (VCC=3V)

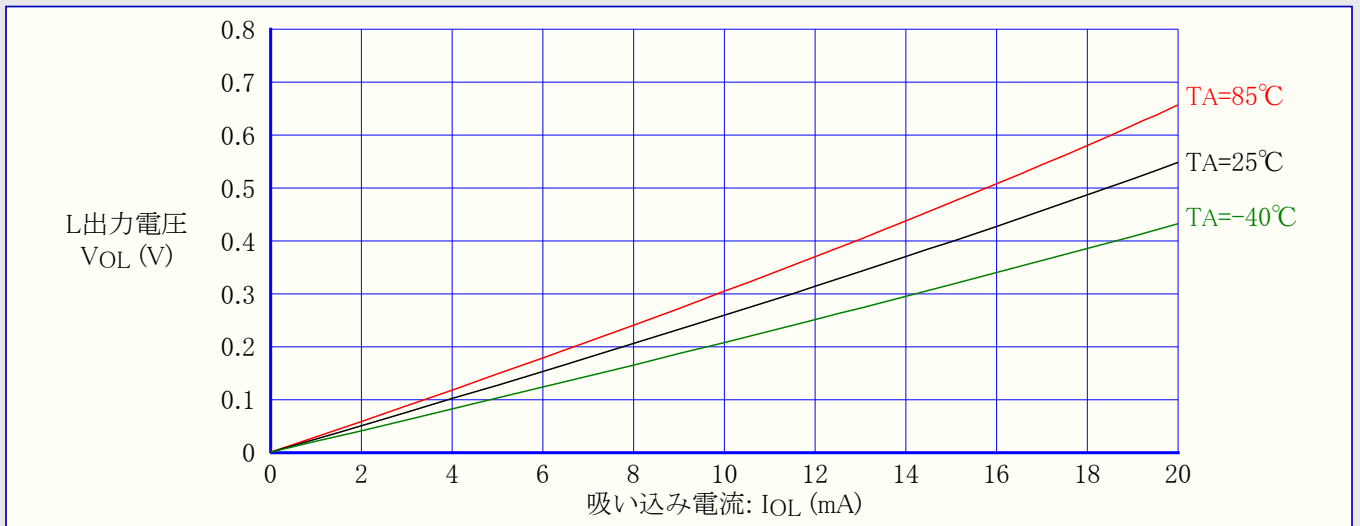


図23-84. 高シク駆動I/Oピン出力電圧対 吸い込み電流 (VCC=5V)

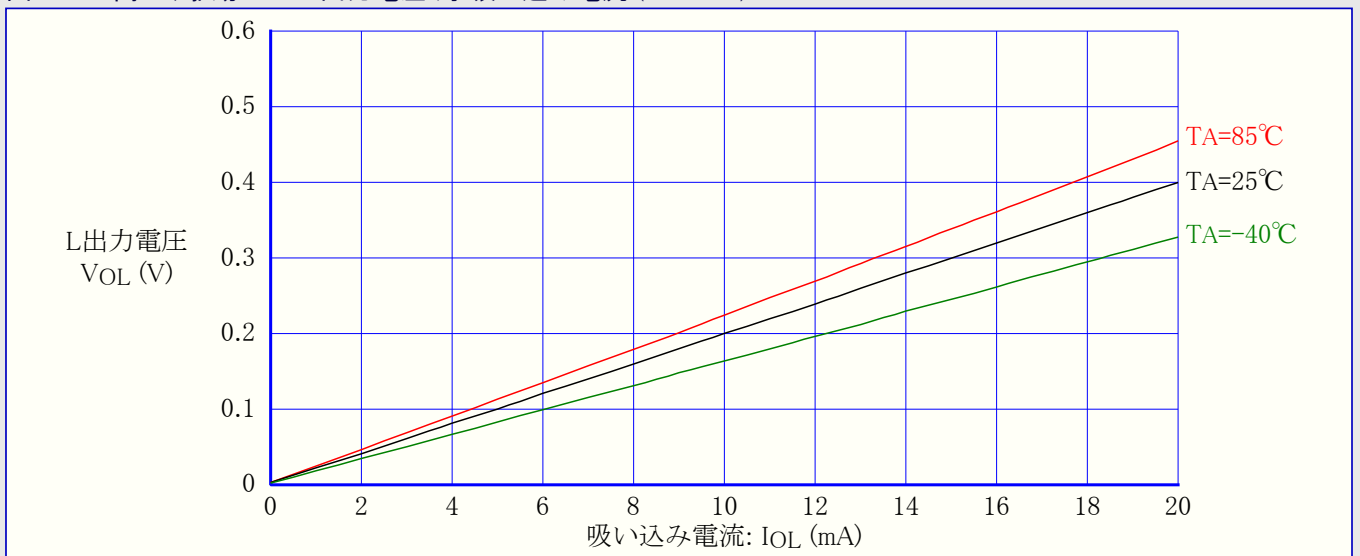


図23-85. 標準I/Oピン 出力電圧 対 吸い込み電流 (VCC=1.8V)

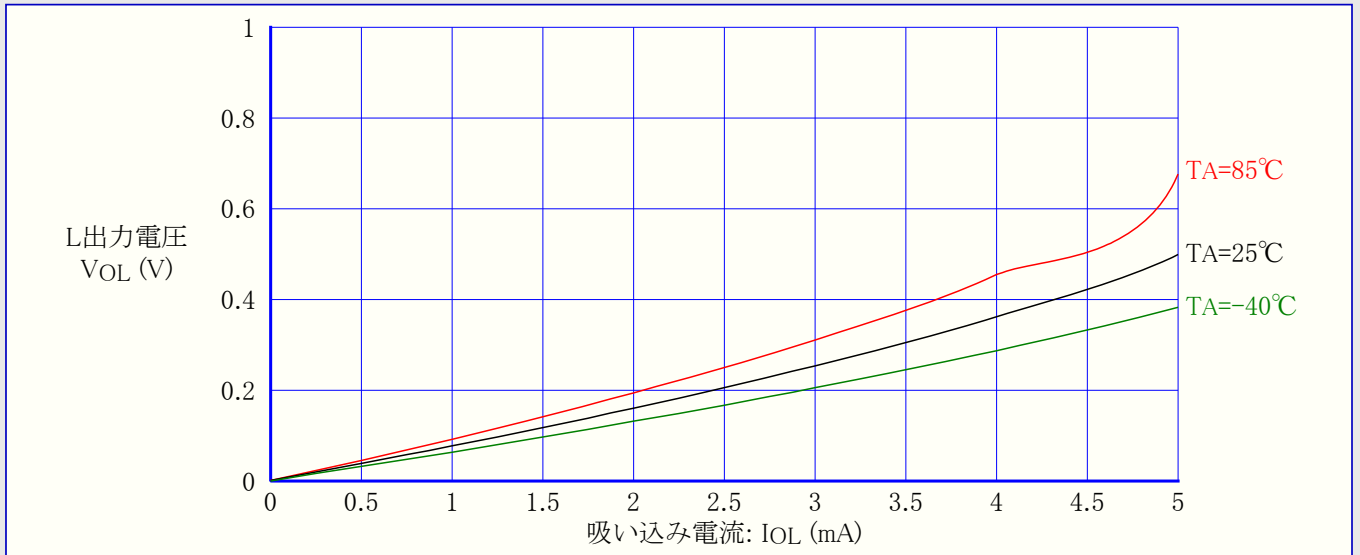


図23-86. 標準I/Oピン 出力電圧 対 吸い込み電流 (VCC=3V)

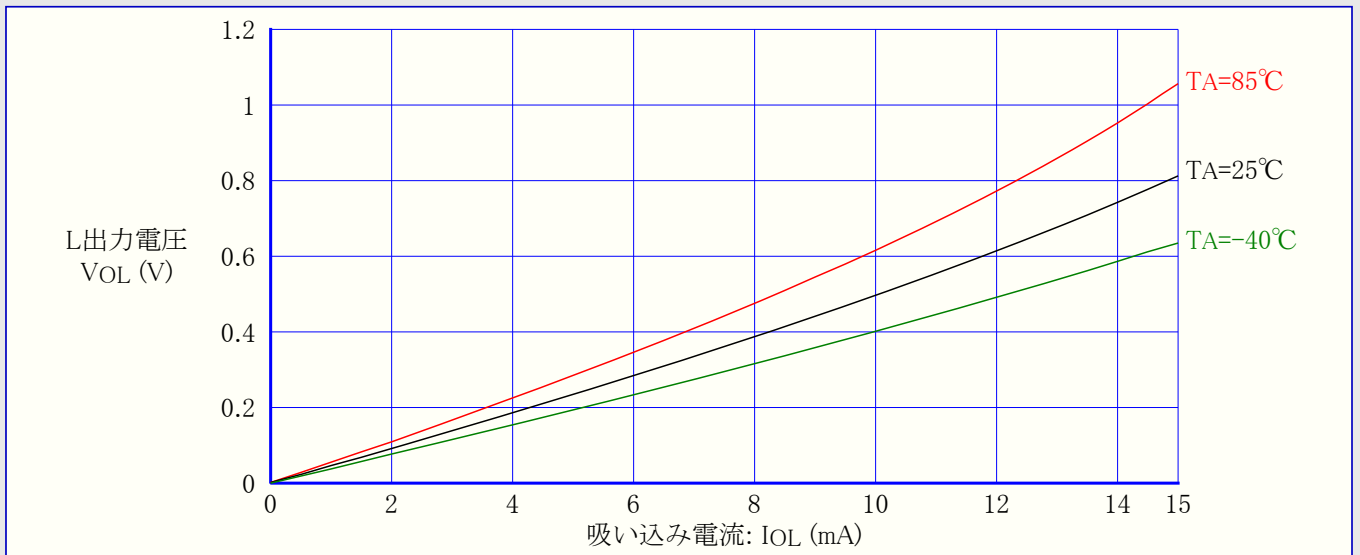


図23-87. 標準I/Oピン 出力電圧 対 吸い込み電流 (VCC=5V)

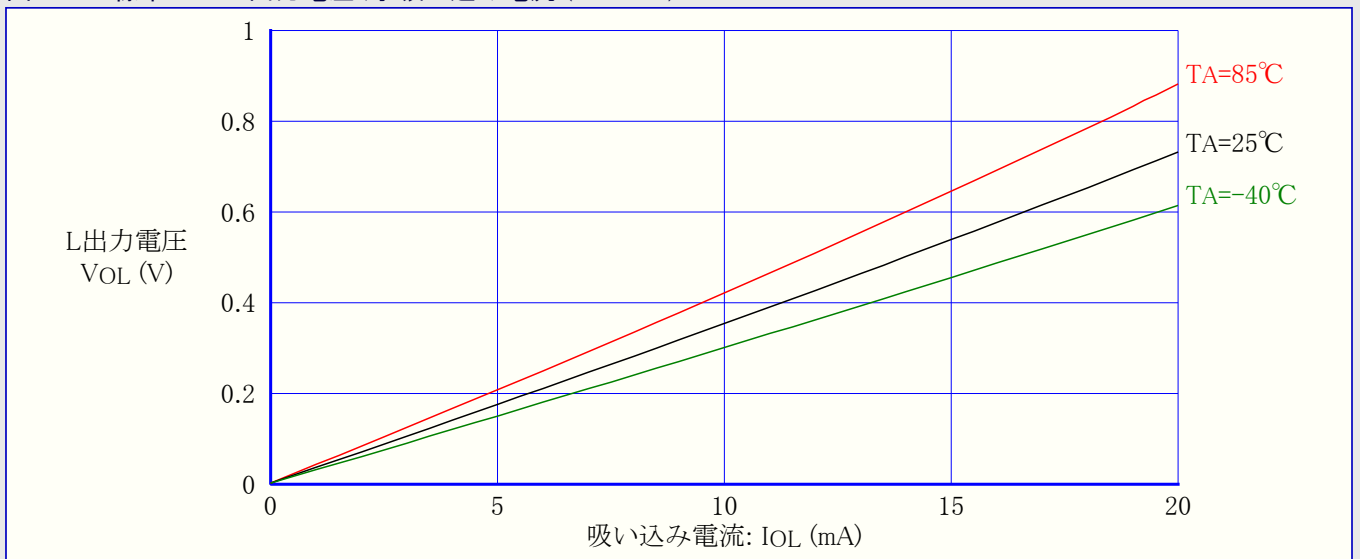


図23-88. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=1.8V)

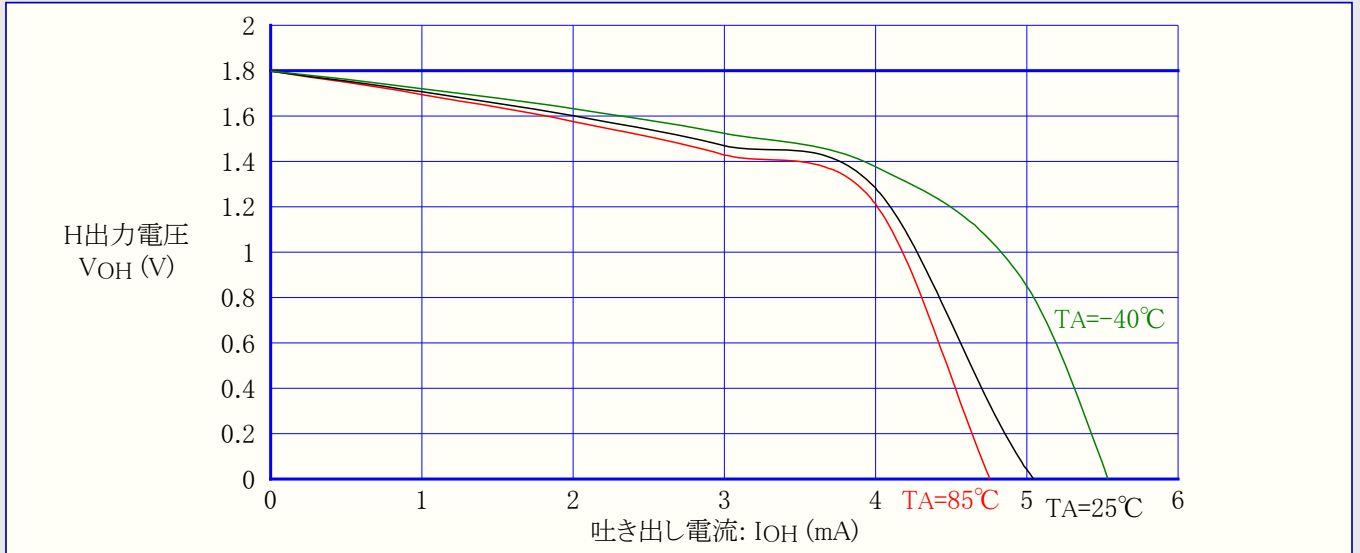


図23-89. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=3V)

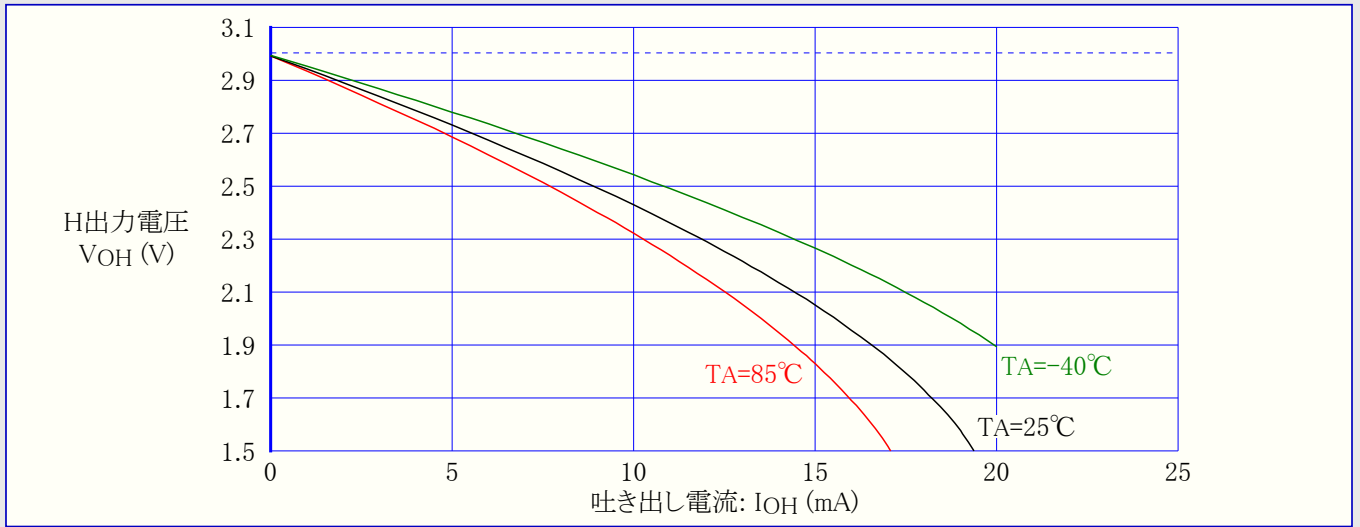


図23-90. 標準I/Oピン 出力電圧 対 吐き出し電流 (VCC=5V)

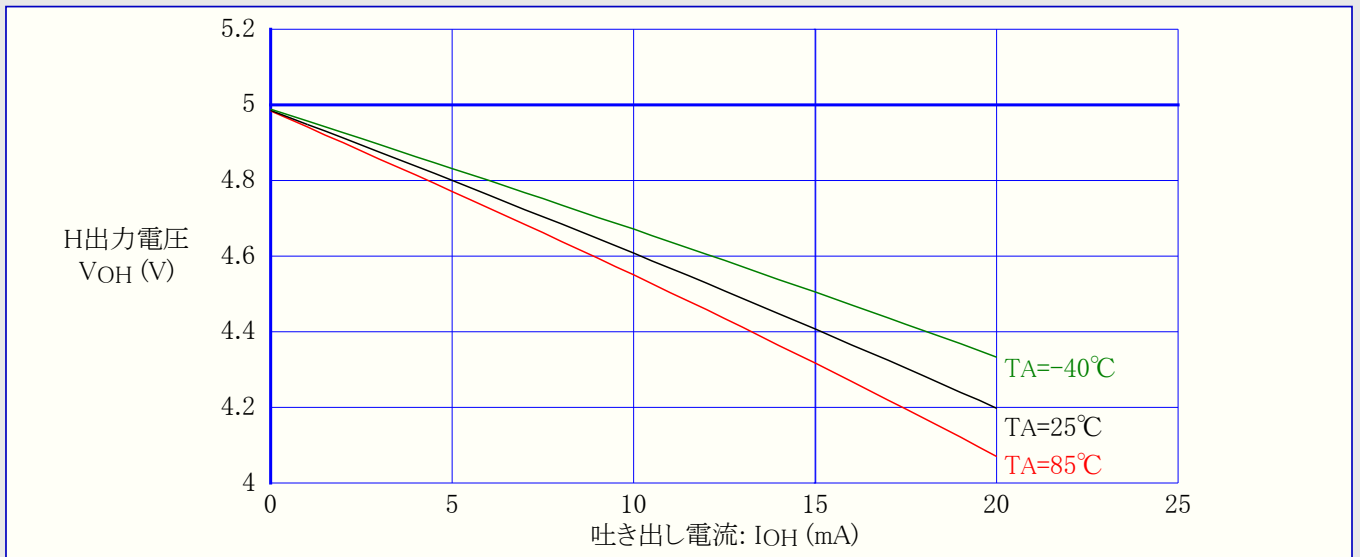


図23-91. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=1.8V)

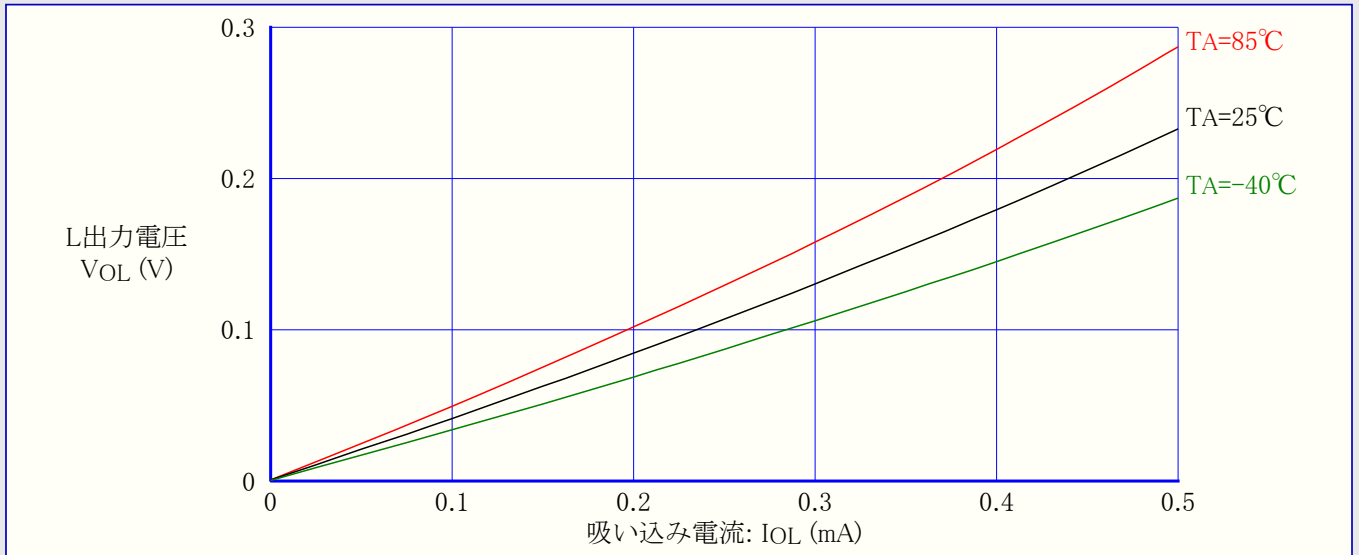


図23-92. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=3V)

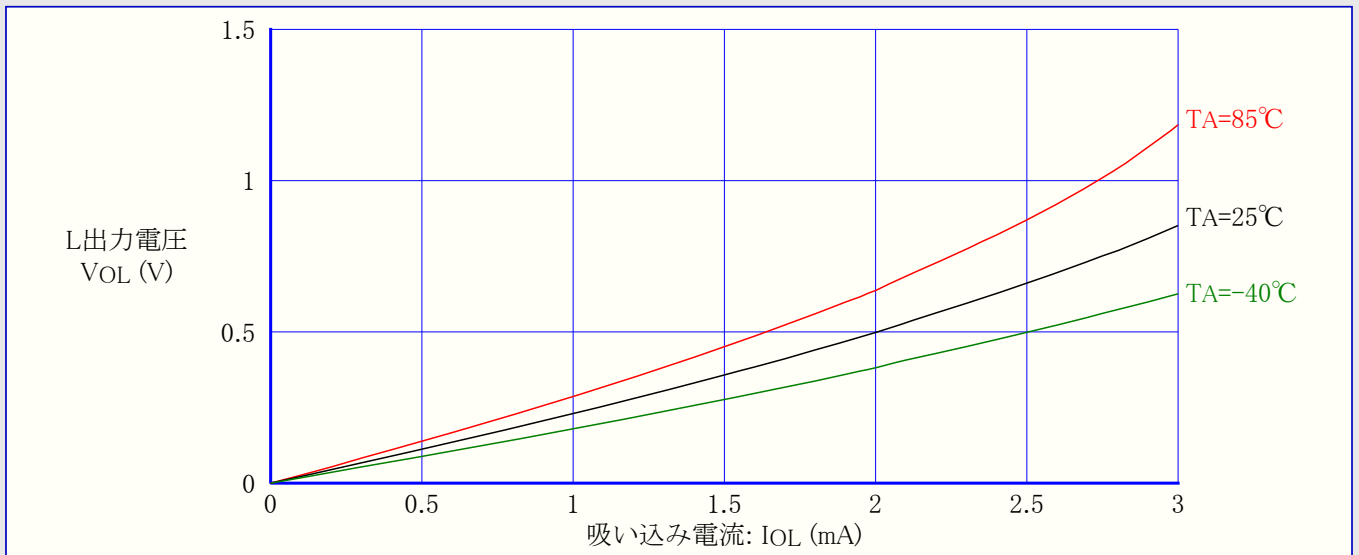


図23-93. I/OとしてのRESETピン 出力電圧 対 吸い込み電流 (VCC=5V)

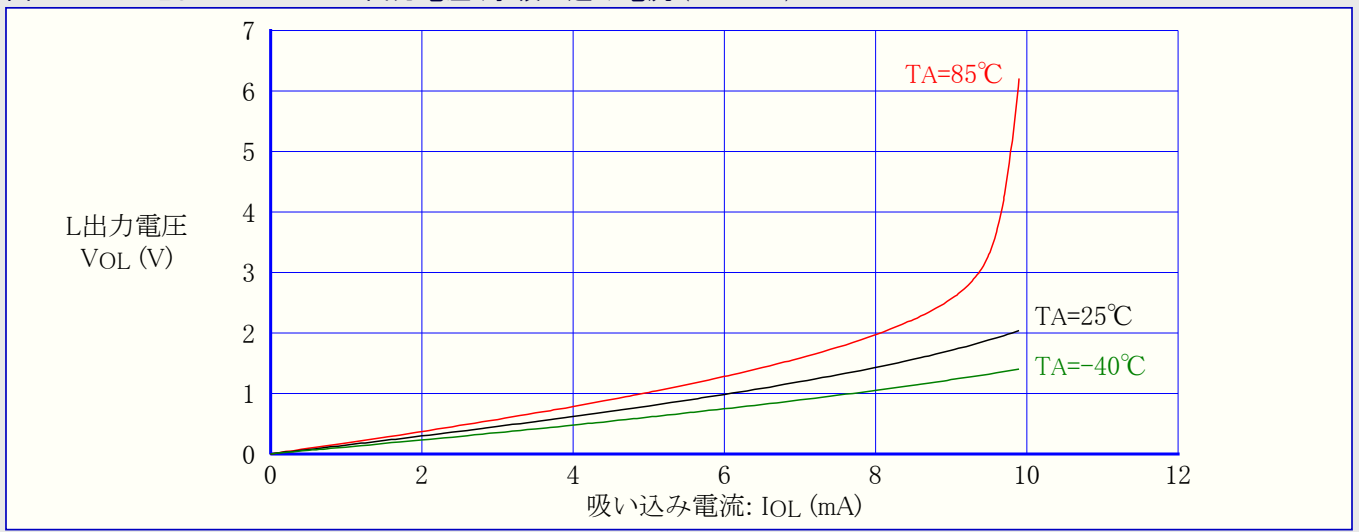


図23-94. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=1.8V)

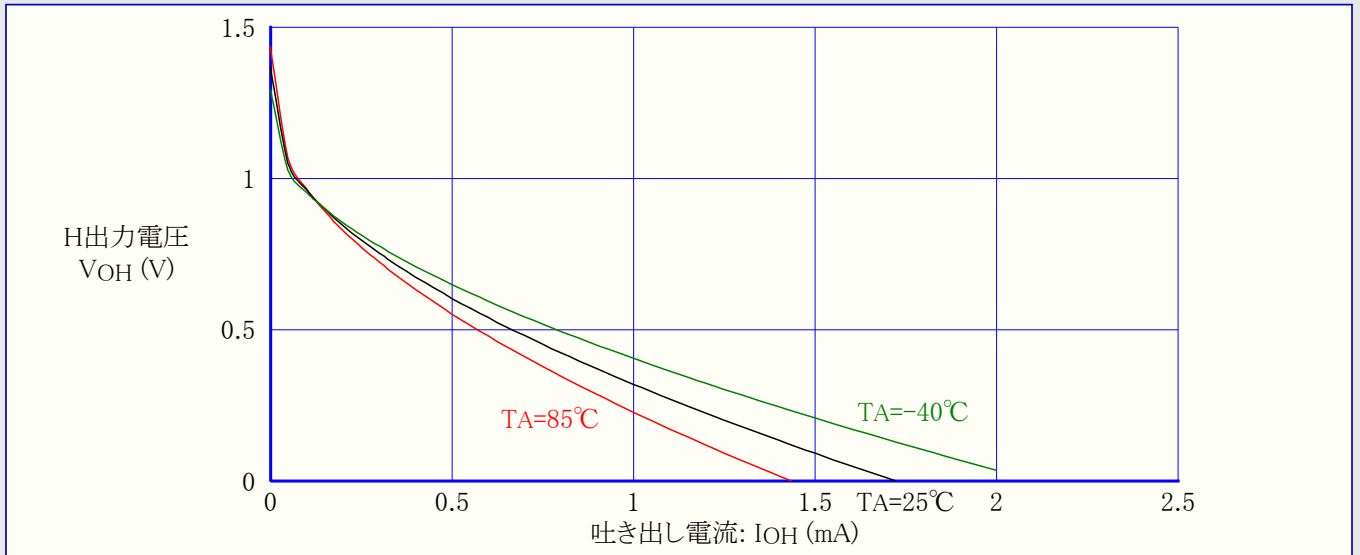


図23-95. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=3V)

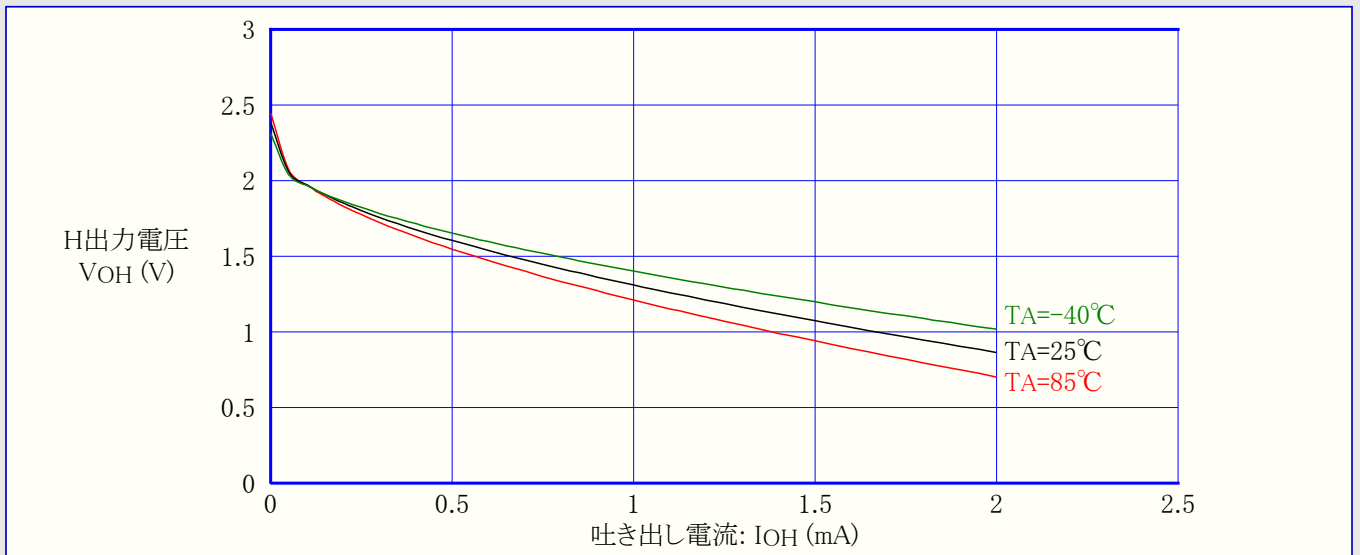
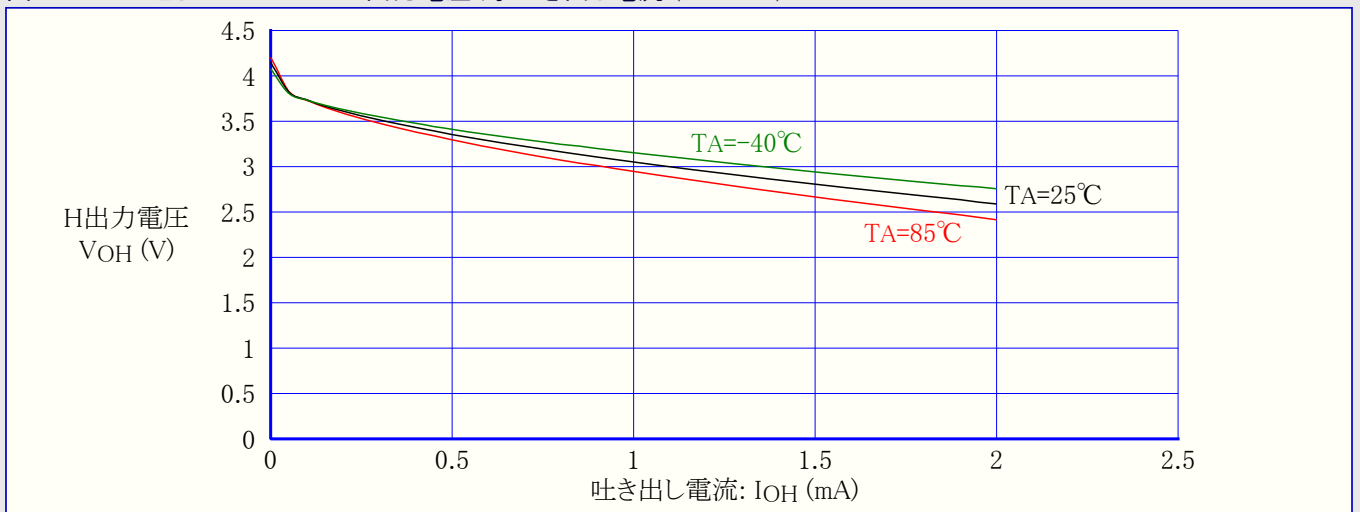


図23-96. I/OとしてのRESETピン 出力電圧 対 吐き出し電流 (VCC=5V)



23.2.8. 入力閾値とヒステリシス

図23-97. 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)

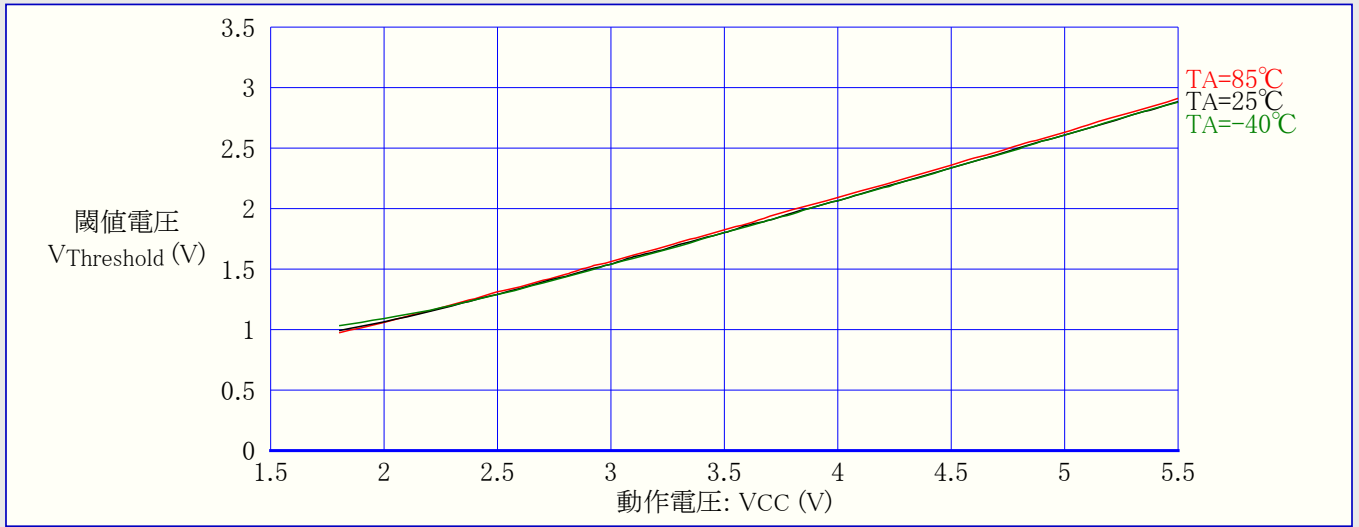


図23-98. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL,0}$ 読み値)

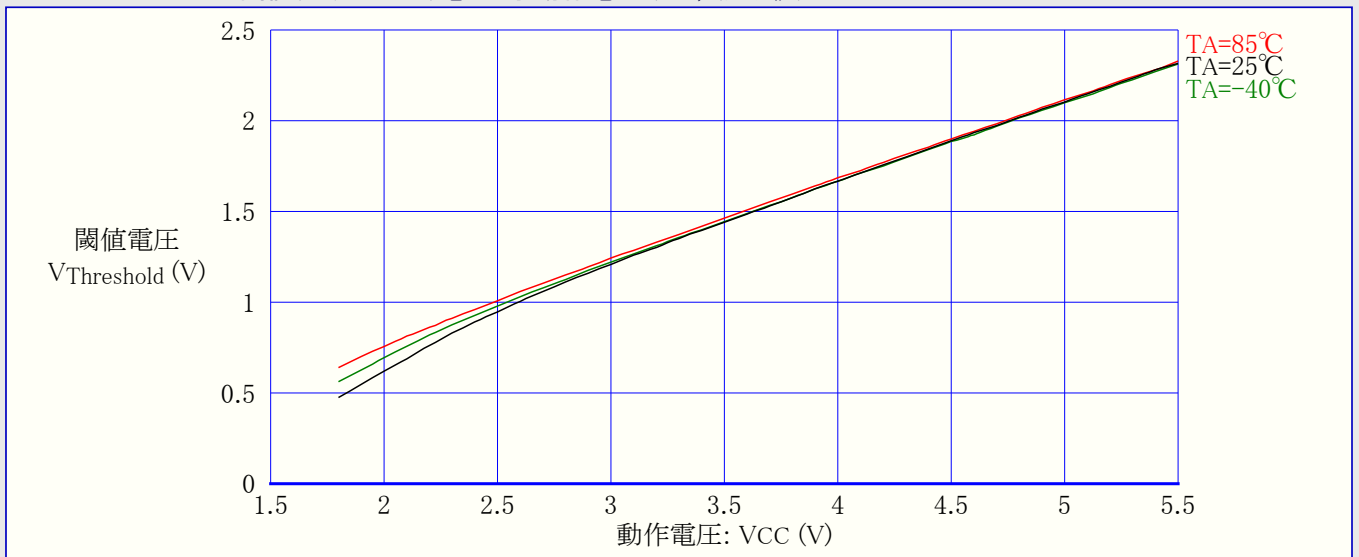


図23-99. I/Oピン入力ヒステリシス電圧 対 動作電圧

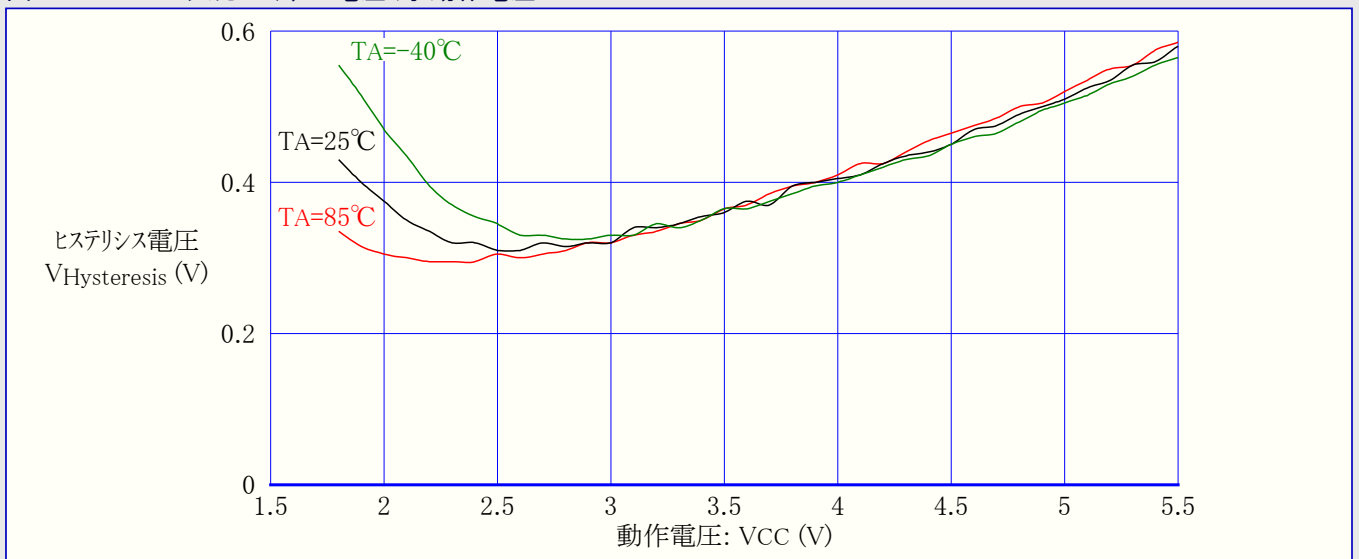


図23-100. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

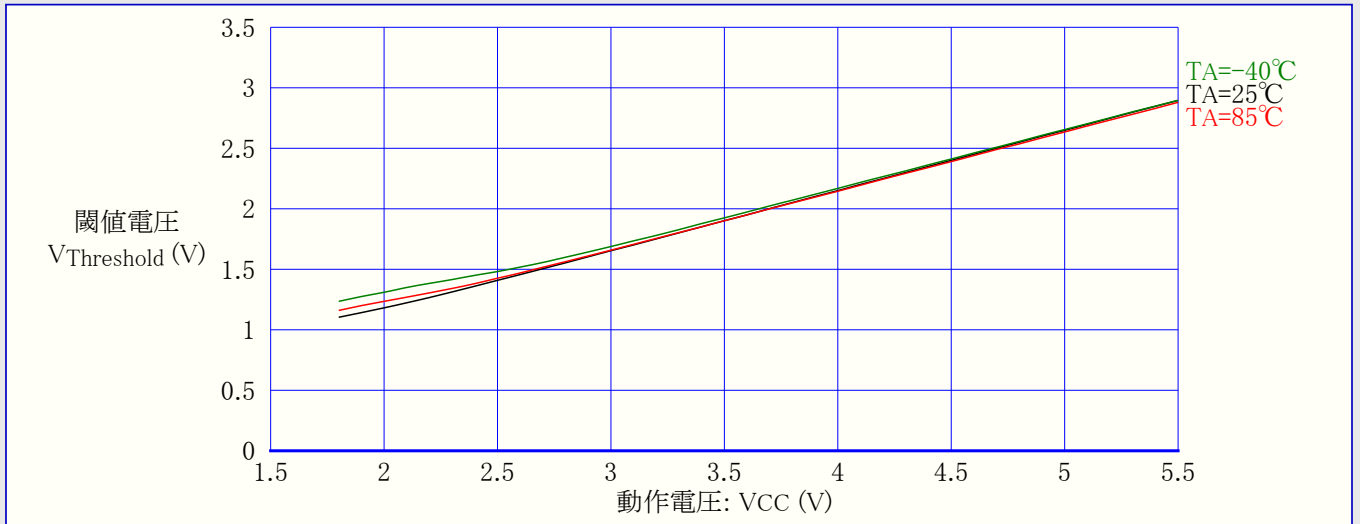


図23-101. I/OとしてのRESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

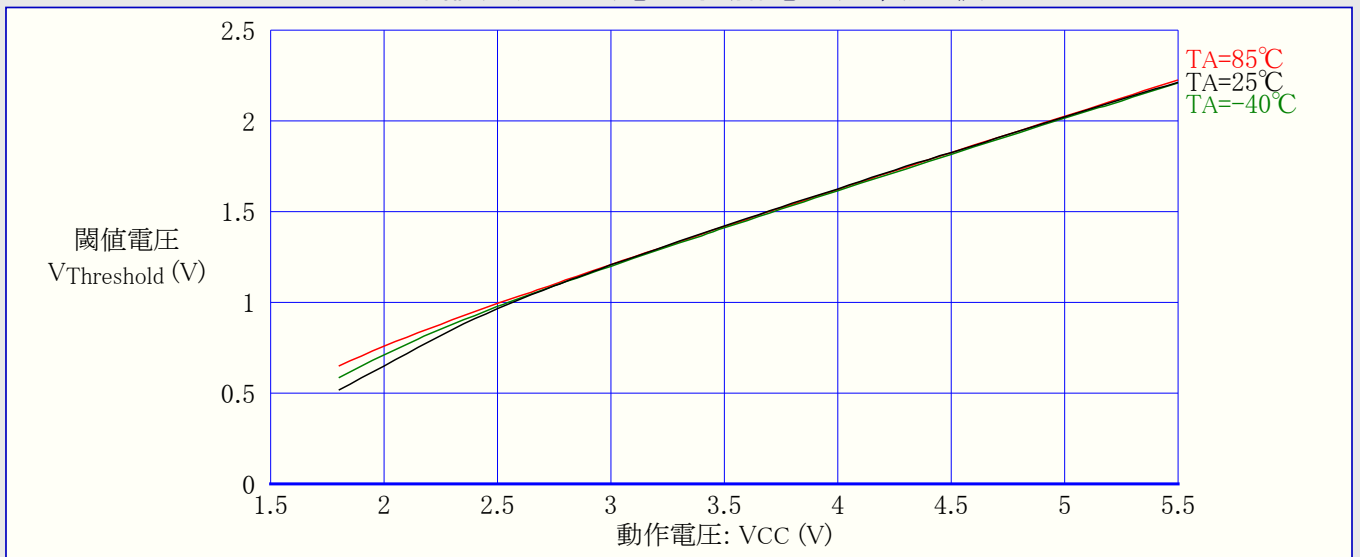
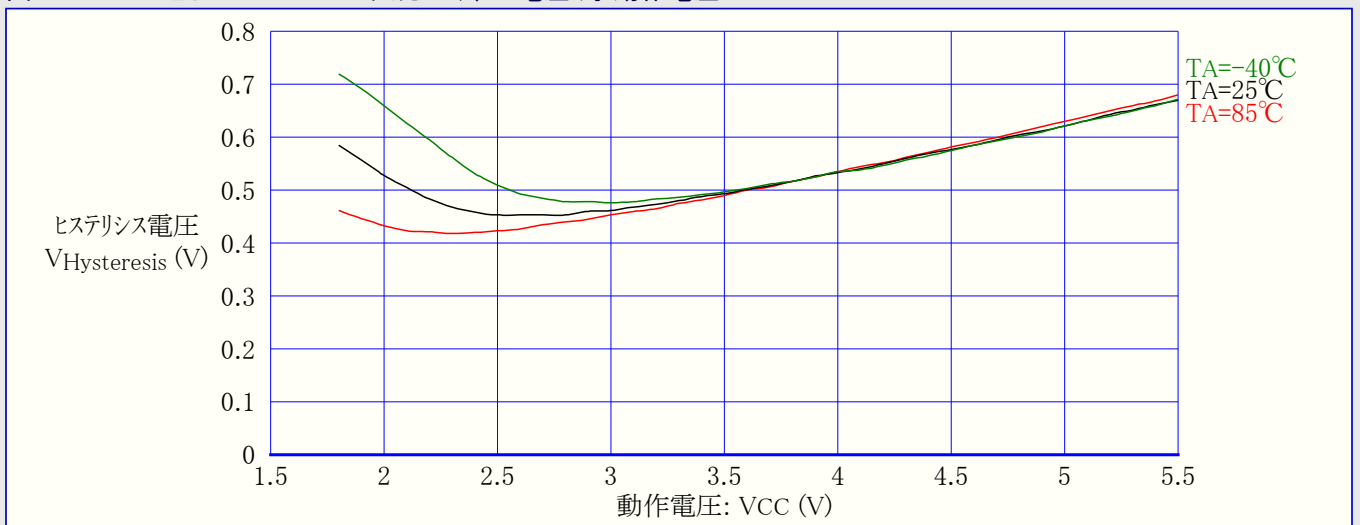


図23-102. I/OとしてのRESETピン入力ヒステリシス電圧 対 動作電圧



23.2.9. 低電圧検出器(BOD)、バンドキャップ、リセット

図23-103. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧4.3V)

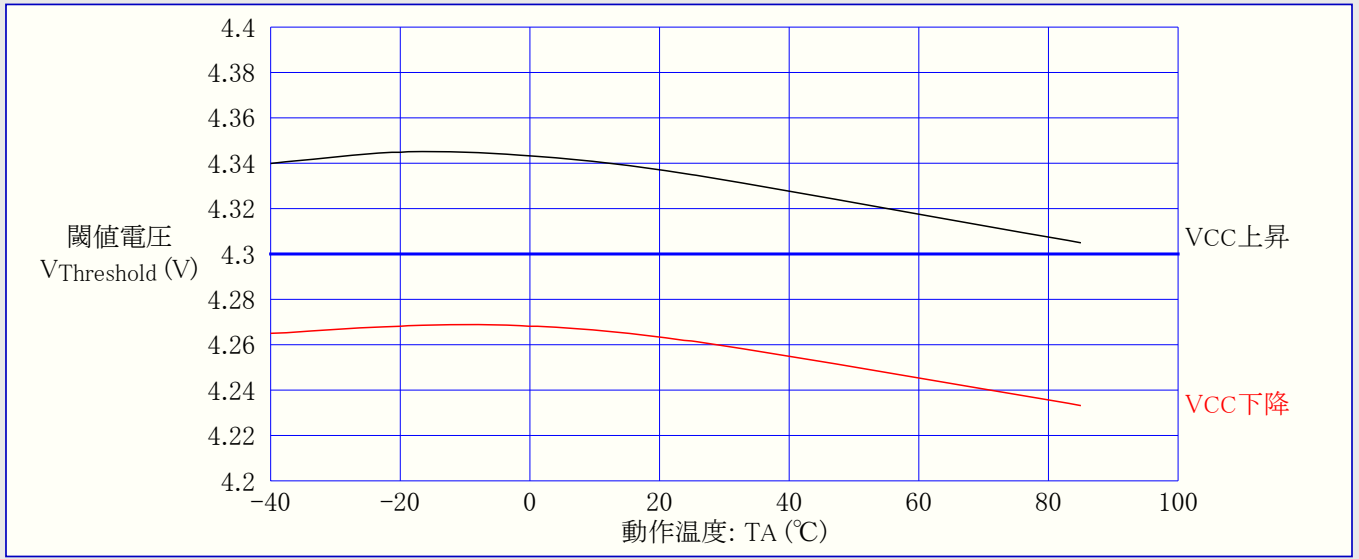


図23-104. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)

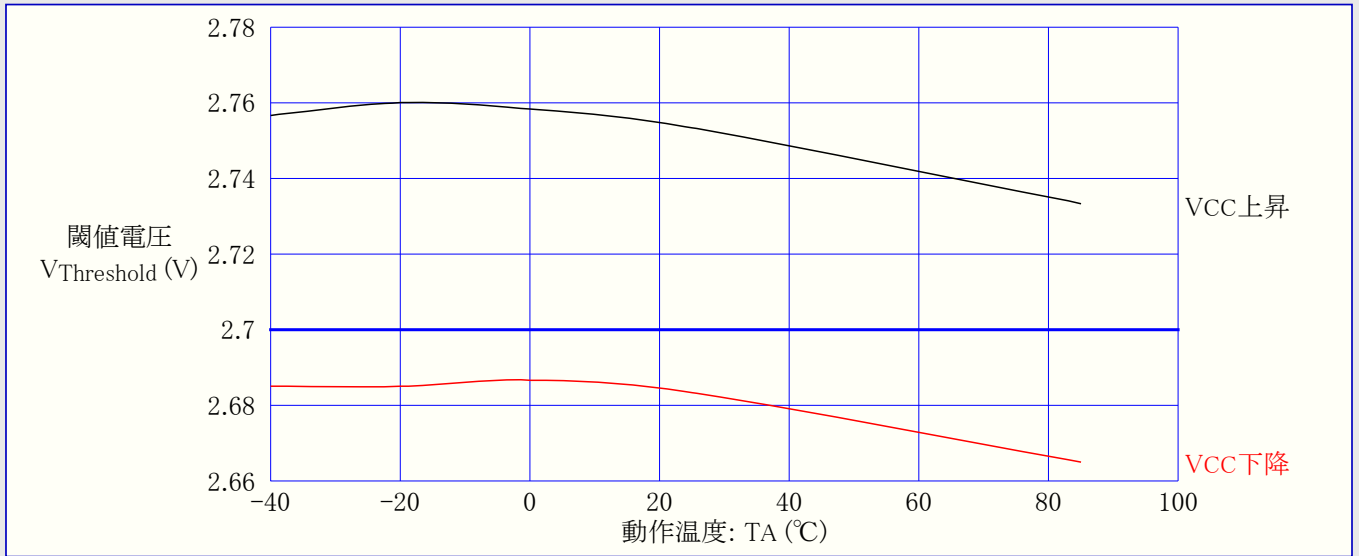


図23-105. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧1.8V)

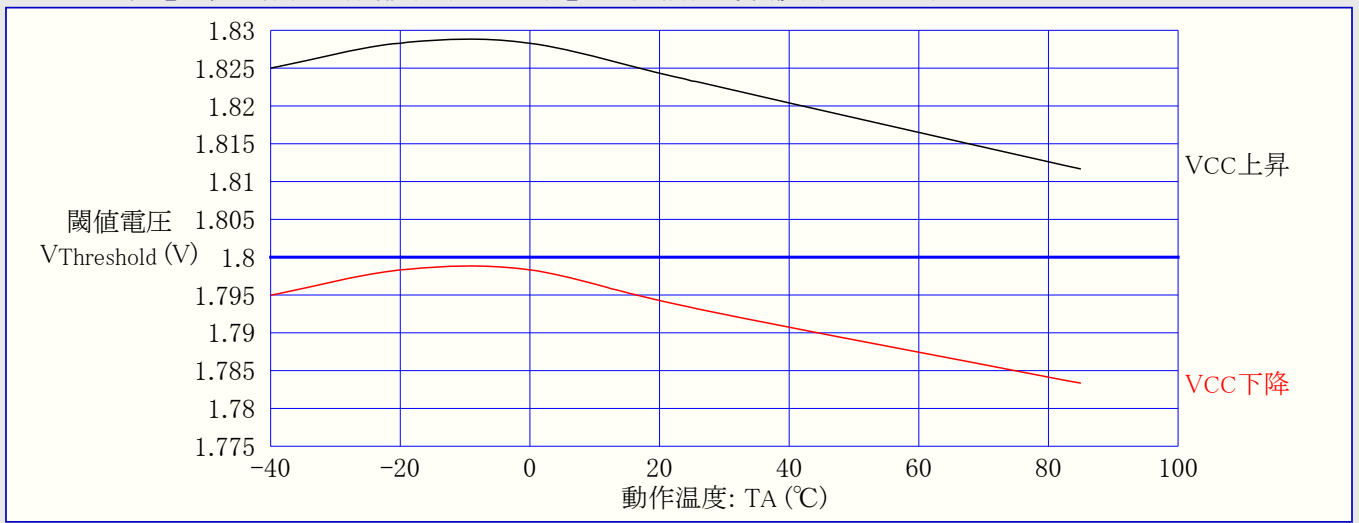


図23-106. RESETピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIH,1読み値)

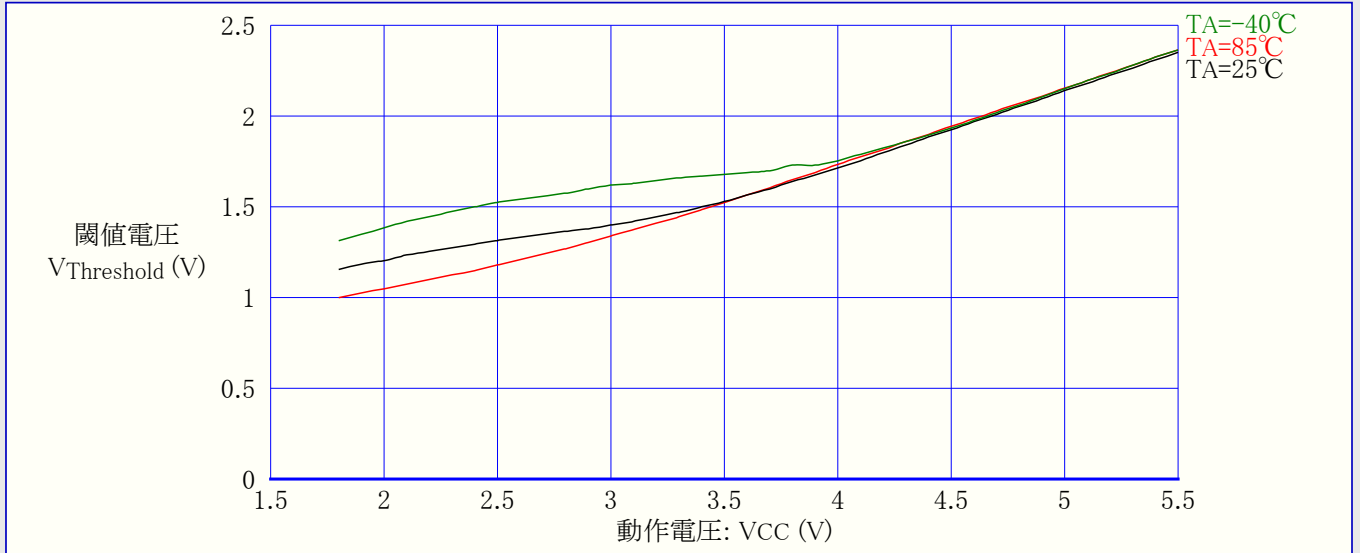


図23-107. RESETピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIL,0読み値)

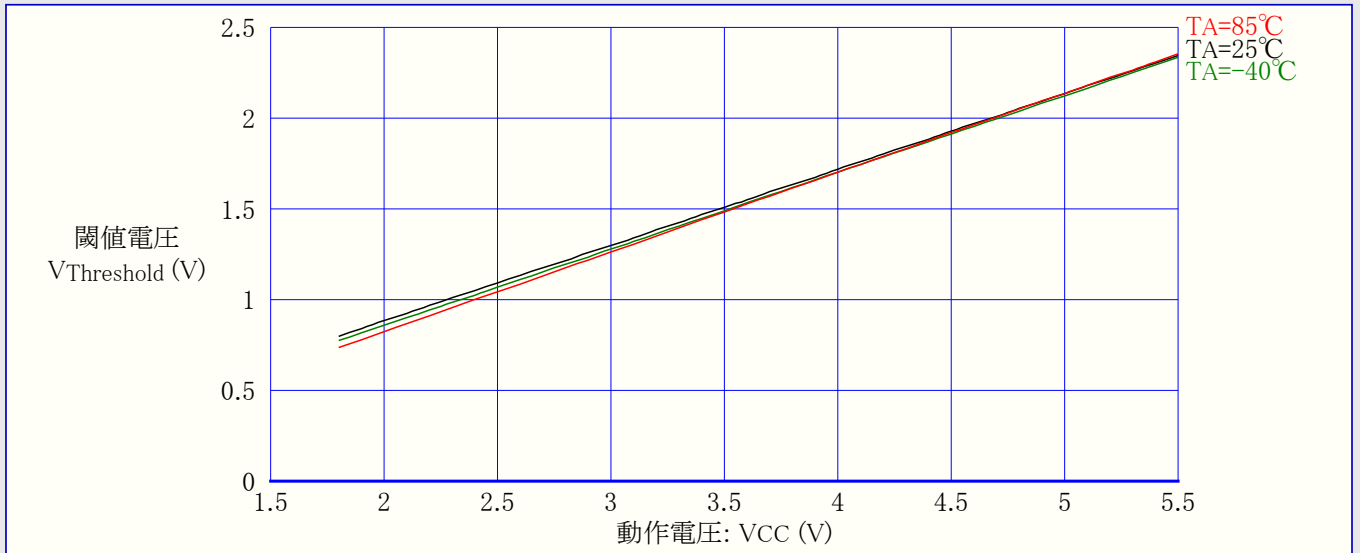


図23-108. I/OとしてのRESETピン入力ヒステリシス電圧 対 動作電圧

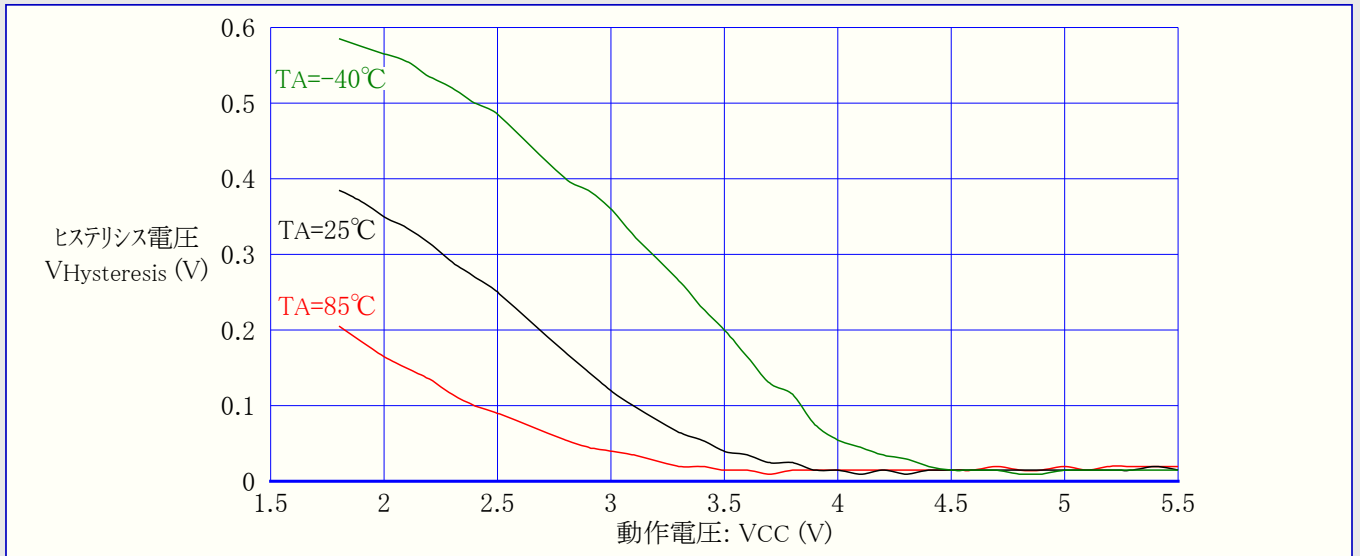
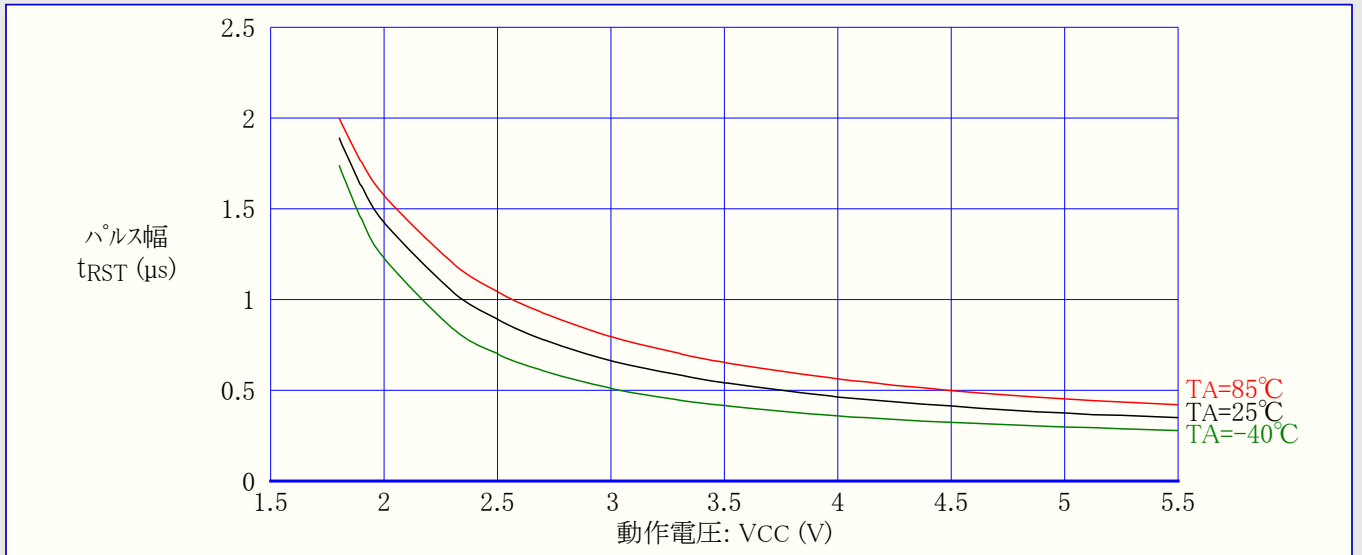


図23-109. 最小リセットパルス幅 対 動作電圧



23.2.10. 内部発振器周波数

図23-110. 128kHzウォッチドッグ発振器周波数 対 動作電圧

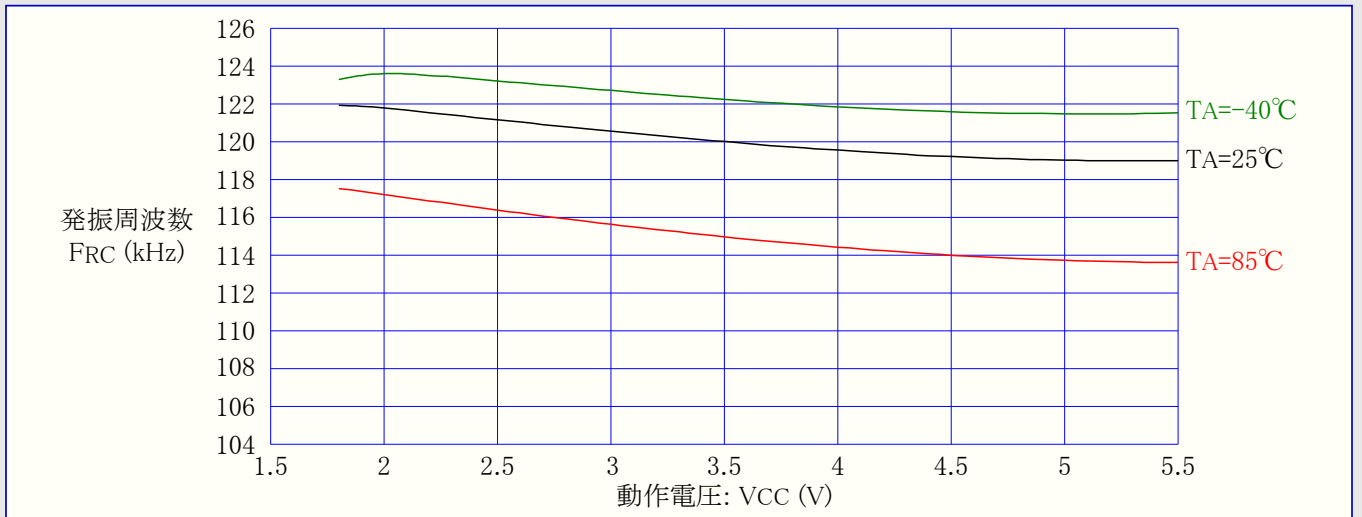


図23-111. 128kHzウォッチドッグ発振器周波数 対 動作温度

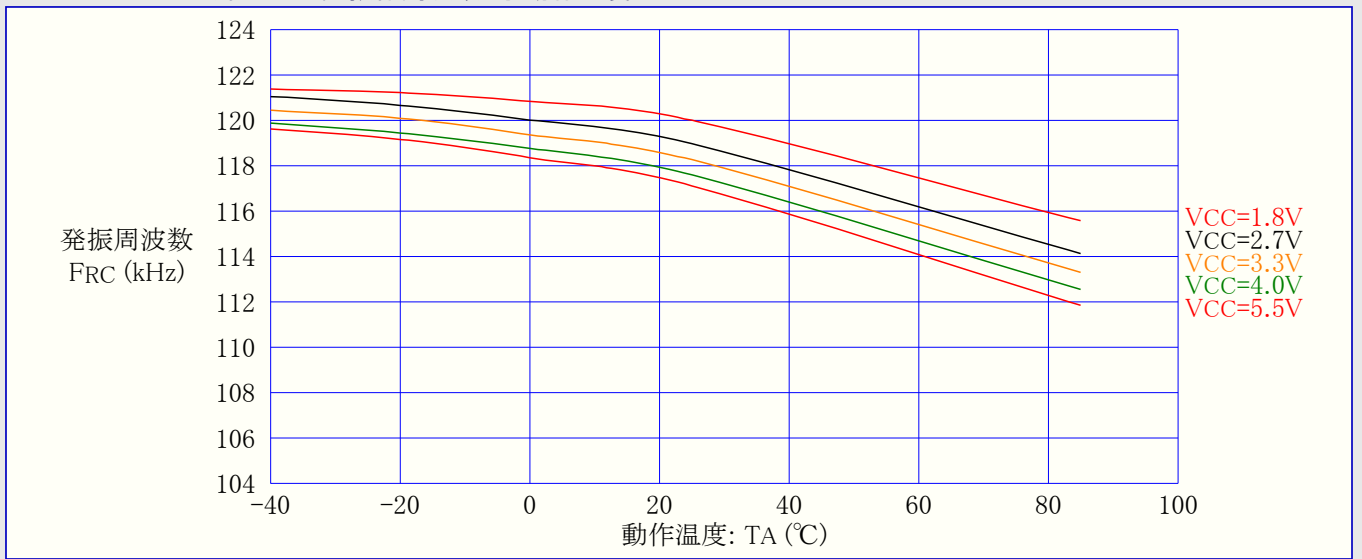


図23-112. 校正済み8MHz内蔵発振器周波数 対 動作電圧

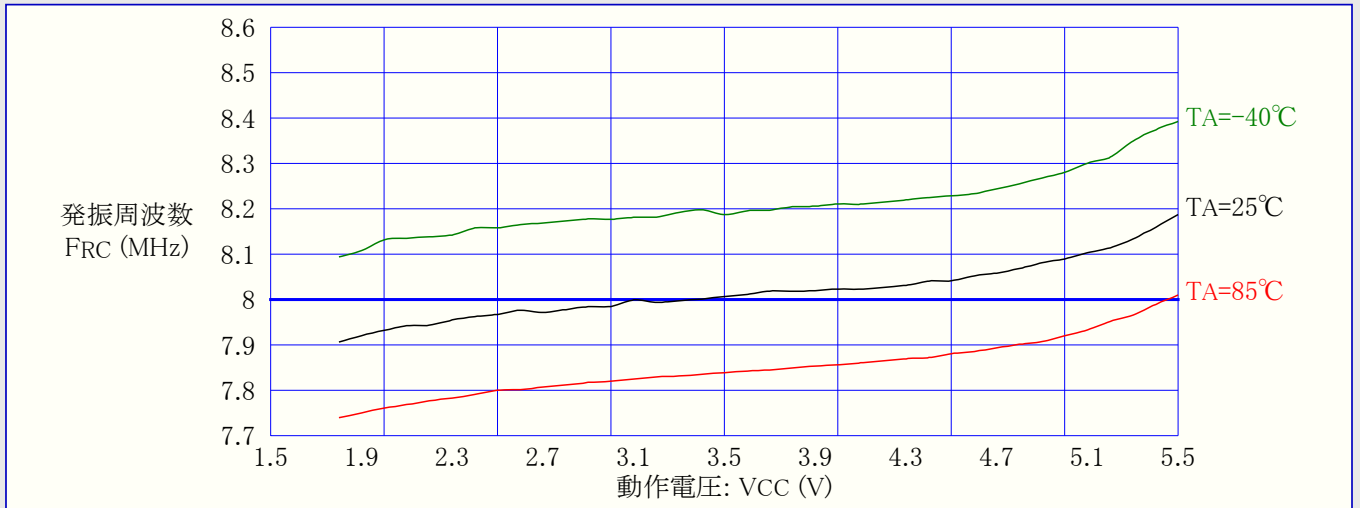


図23-113. 校正済み8MHz内蔵発振器周波数 対 動作温度

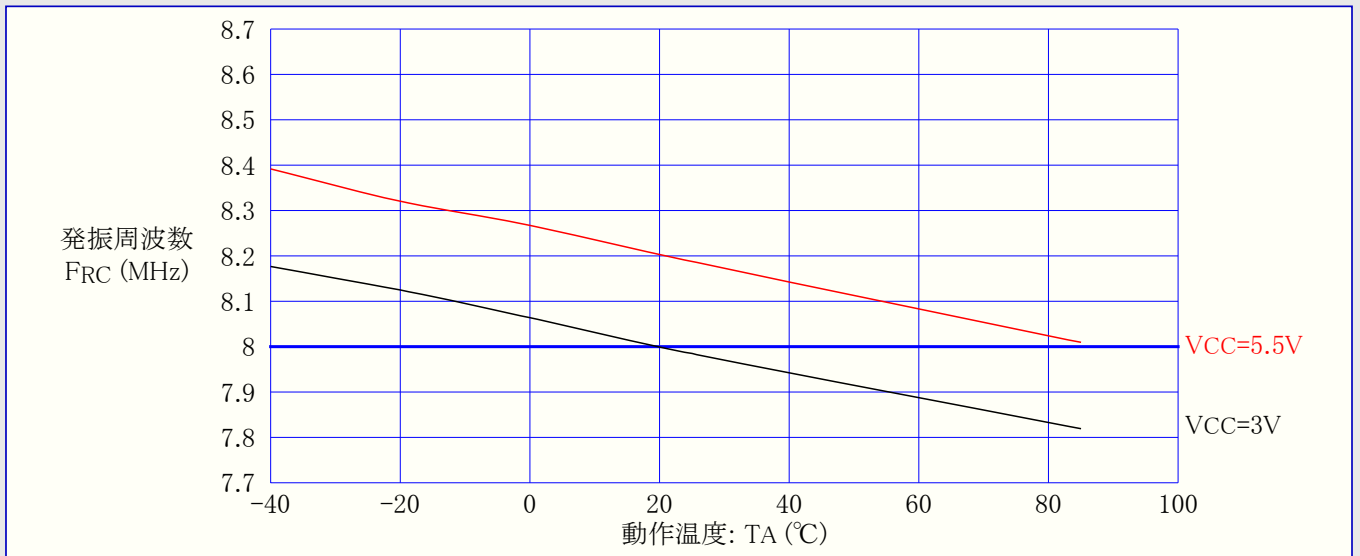
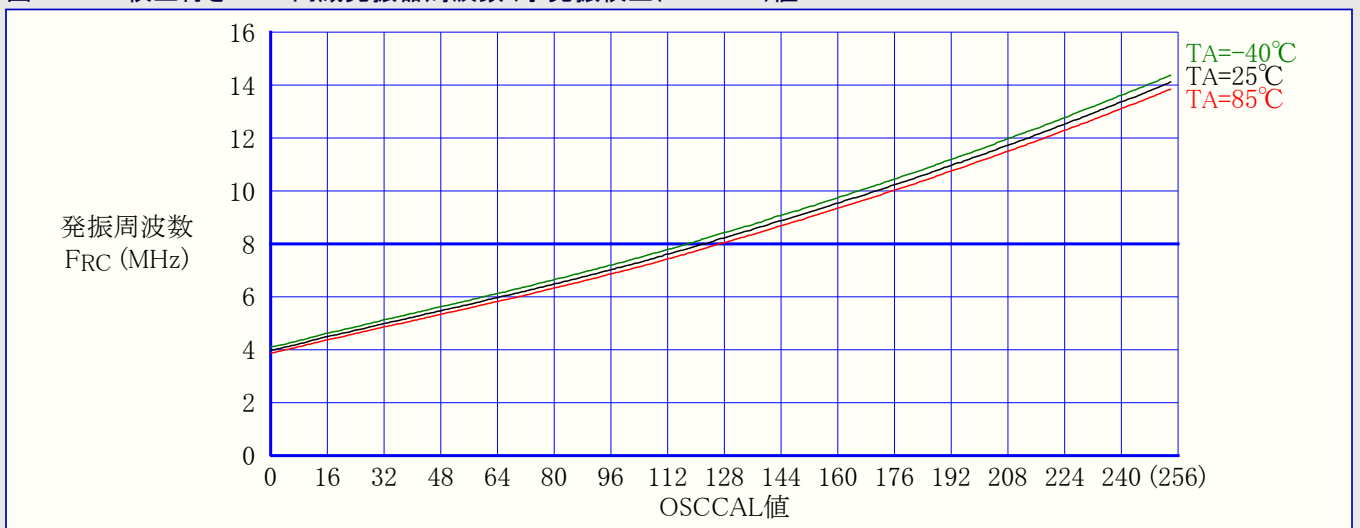


図23-114. 校正付き8MHz内蔵発振器周波数 対 発振校正(OSCCAL)値



24. レジスタ要約

拡張1/レジスタ領域 (1/2)

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
(~\$FF)	予約									
(\$BF)	予約									
(\$BE)	TWHSR	-	-	-	-	-	-	-	TWHS	104
(\$BD)	TWAMR	TWAM6	TWAM5	TWAM4	TWAM3	TWAM2	TWAM1	TWAM0	-	104
(\$BC)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	102
(\$BB)	TWDR	2線直列インターフェースデータレジスタ								103
(\$BA)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	103
(\$B9)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	-	TWPS1	TWPS0	103
(\$B8)	TWBR	2線直列インターフェースビット速度レジスタ								102
(\$B7)	予約									
(\$B6)	予約									
(\$B5)	予約									
(\$B4)	予約									
(\$B3)	予約									
(\$B2)	予約									
(\$B1)	予約									
(\$B0)	予約									
(\$AF)	予約									
(\$AE)	予約									
(\$AD)	予約									
(\$AC)	予約									
(\$AB)	予約									
(\$AA)	予約									
(\$A9)	予約									
(\$A8)	予約									
(\$A7)	予約									
(\$A6)	予約									
(\$A5)	予約									
(\$A4)	予約									
(\$A3)	予約									
(\$A2)	予約									
(\$A1)	予約									
(\$A0)	予約									
(\$9F)	予約									
(\$9E)	予約									
(\$9D)	予約									
(\$9C)	予約									
(\$9B)	予約									
(\$9A)	予約									
(\$99)	予約									
(\$98)	予約									
(\$97)	予約									
(\$96)	予約									
(\$95)	予約									
(\$94)	予約									
(\$93)	予約									
(\$92)	予約									
(\$91)	予約									
(\$90)	予約									
(\$8F)	予約									
(\$8E)	予約									
(\$8D)	予約									
(\$8C)	予約									
(\$8B)	OCRIBH									74
(\$8A)	OCRIBL									74
(\$89)	OCRIAH									74
(\$88)	OCRIAL									74
(\$87)	ICRIH									74
(\$86)	ICRIL									74
(\$85)	TCNTIH									73
(\$84)	TCNTIL									73
(\$83)	予約									
(\$82)	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-	73
(\$81)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	72
(\$80)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	71



拡張I/Oレジスタ領域 (2/2)

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	寛
(\$7F)	DIDR1	-	-	-	-	-	-	AIN1D	AIN0D	106
(\$7E)	DIDR0	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	115
(\$7D)	予約									
(\$7C)	ADMUX	-	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	113
(\$7B)	ADCSRB	-	ACME	-	-	-	ADTS2	ADTS1	ADTS0	105,114
(\$7A)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	113
(\$79)	ADCH	A/Dデータレジスタ上位バイト (ADC9~8またはADC9~2)								115
(\$78)	ADCL	A/Dデータレジスタ下位バイト (ADC7~0またはADC1~0)								
(\$77)	予約									
(\$76)	予約									
(\$75)	予約									
(\$74)	予約									
(\$73)	予約									
(\$72)	予約									
(\$71)	予約									
(\$70)	予約									
(\$6F)	TIMSK1	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1	75
(\$6E)	TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0	56
(\$6D)	PCMSK2	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	36
(\$6C)	PCMSK1	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	36
(\$6B)	PCMSK0	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	36
(\$6A)	PCMSK3	-	-	-	-	PCINT27	PCINT26	PCINT25	PCINT24	36
(\$69)	EICRA	-	-	-	-	ISC11	ISC10	ISC01	ISC00	33
(\$68)	PCICR	-	-	-	-	PCIE3	PCIE2	PCIE1	PCIE0	35
(\$67)	予約									
(\$66)	OSCCAL	内蔵RC発振器 発振校正値レジスタ								20
(\$65)	予約									
(\$64)	PRR	PRTWI	-	PRTIM0	-	PRTIM1	PRSPI	-	PRADC	24
(\$63)	予約									
(\$62)	予約									
(\$61)	CLKPR	CLKPCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	20
(\$60)	WDTCSR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	30

注: 暗背景のレジスタとビットは28ピン外圍器で利用できません。

- 注意:**
- 将来のデバイスとの共通性のため、アクセスされる場合の予約ビットは0を書かれるべきです。予約した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内のI/Oレジスタは**SBI**と**CBI**命令を使う直接的なビットアクセスが可能です。これらのレジスタでは**SBIS**と**SBIC**命令を使うことによって単一ビット値が検査できます。
  - 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の様ではなく、**CBI**と**SBI**命令は指定ビットだけを操作し、故にこのような状態フラグを含むレジスタで使えます。**CBI**と**SBI**命令は\$00~\$1Fのレジスタだけで動作します。
  - 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**と**ST**命令を使ってデータ空間としてI/Oレジスタをアドレス指定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。ATtiny48/88は**IN**と**OUT**命令で予約した64位置で支援できるよりも多くの周辺部(機能)の複合マイクロコントローラです。SRAM(データ空間)内の拡張I/O空間は**ST/STS/STD**と**LD/LDS/LDD**命令だけが使えます。

標準I/Oレジスタ領域

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	10
\$3E (\$5E)	SPH	-	-	-	-	-	-	(SP9)	SP8	10
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	
\$3C (\$5C)	予約									
\$3B (\$5B)	予約									
\$3A (\$5A)	予約									
\$39 (\$59)	予約									
\$38 (\$58)	予約									
\$37 (\$57)	SPMCSR	-	RWWSB	-	CTPB	RFLB	PGWRT	PGERS	SELFPRGEN	120
\$36 (\$56)	予約									
\$35 (\$55)	MCUCR	-	BODS	BODSE	PUD	-	-	-	-	23,49
\$34 (\$54)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	29
\$33 (\$53)	SMCR	-	-	-	-	-	SM1	SM0	SE	23
\$32 (\$52)	予約									
\$31 (\$51)	DWDR	デバッグWIRE データレジスタ								116
\$30 (\$50)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	106
\$2F (\$4F)	予約									
\$2E (\$4E)	SPDR	SPI データレジスタ								83
\$2D (\$4D)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	82
\$2C (\$4C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	82
\$2B (\$4B)	GPIOR2	汎用I/Oレジスタ2								16
\$2A (\$4A)	GPIOR1	汎用I/Oレジスタ1								16
\$29 (\$49)	予約									
\$28 (\$48)	OCR0B	タイマ/カウンタ0 比較レジスタ								56
\$27 (\$47)	OCR0A	タイマ/カウンタ0 比較Aレジスタ								55
\$26 (\$46)	TCNT0	タイマ/カウンタ0								55
\$25 (\$45)	TCCR0A	-	-	-	-	CTC0	CS02	CS01	CS00	55
\$24 (\$44)	予約									
\$23 (\$43)	GTCCR	TSM	-	-	-	-	-	-	PSRSYNC	77
\$22 (\$42)	予約									
\$21 (\$41)	EEARL	-	-	EEPROMアドレスレジスタ下位バイト (EEAR5~0)						15
\$20 (\$40)	EEDR	EEPROMデータレジスタ								15
\$1F (\$3F)	EECR	-	-	EPM1	EPM0	EERIE	EEMPE	EEPE	EERE	15
\$1E (\$3E)	GPIOR0	汎用I/Oレジスタ0								16
\$1D (\$3D)	EIMSK	-	-	-	-	-	-	INT1	INT0	34
\$1C (\$3C)	EIFR	-	-	-	-	-	-	INTF1	INTF0	34
\$1B (\$3B)	PCIFR	-	-	-	-	PCIF3	PCIF2	PCIF1	PCIF0	35
\$1A (\$3A)	予約									
\$19 (\$39)	予約									
\$18 (\$38)	予約									
\$17 (\$37)	予約									
\$16 (\$36)	TIFR1	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1	75
\$15 (\$35)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0	56
\$14 (\$34)	予約									
\$13 (\$33)	予約									
\$12 (\$32)	PORTCR	BBMD	BBMC	BBMB	BBMA	PUDD	PUDC	PUDB	PUDA	49
\$11 (\$31)	予約									
\$10 (\$30)	予約									
\$0F (\$2F)	予約									
\$0E (\$2E)	PORTA	-	-	-	-	PORTA3	PORTA2	PORTA1	PORTA0	49
\$0D (\$2D)	DDRA	-	-	-	-	DDA3	DDA2	DDA1	DDA0	49
\$0C (\$2C)	PINA	-	-	-	-	PINA3	PINA2	PINA1	PINA0	49
\$0B (\$2B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	50
\$0A (\$2A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	50
\$09 (\$29)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	50
\$08 (\$28)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	50
\$07 (\$27)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	50
\$06 (\$26)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	50
\$05 (\$25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	49
\$04 (\$24)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	50
\$03 (\$23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	50
\$02 (\$22)	予約									
\$01 (\$21)	予約									
\$00 (\$20)	予約									

注: 暗背景のレジスタとビットは28ピン外圍器で利用できません。( )付きビットはATtiny88でのみ利用可能です。

(訳注) 原書本位置の注意は前頁に移動しました。



## 25. 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
<b>算術、論理演算命令</b>					
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
<b>分岐命令</b>					
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
SBR	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
<b>データ移動命令</b>					
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
<b>ビット関係命令</b>					
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,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,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
<b>MCU制御命令</b>					
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

## 26. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号 (注1,2,3)	外囲器	動作範囲
ATtiny48	12	1.8~5.5V	ATtiny48-MMU	28M1	工業用 (-40°C~85°C)
			ATtiny48-MMUR		
			ATtiny48-MMH		
			ATtiny48-MMHR		
			ATtiny48-PU	28P3	
			ATtiny48-AU	32A	
			ATtiny48-AUR		
			ATtiny48-CCU	32CC1	
			ATtiny48-CCUR		
			ATtiny48-MU	32M1-A	
ATtiny48-MUR					
ATtiny88	12	1.8~5.5V	ATtiny88-MMU	28M1	
			ATtiny88-MMUR		
			ATtiny88-MMH		
			ATtiny88-MMHR		
			ATtiny88-PU	28P3	
			ATtiny88-AU	32A	
			ATtiny88-AUR		
			ATtiny88-CCU	32CC1	
			ATtiny88-CCUR		
			ATtiny88-MU	32M1-A	
ATtiny88-MUR					

**注1:** 符号識別子は次のとおりです。

- H : ニッケル・パラジウム・金(NiPdAu)仕上げ。
- U : 半光沢錫。
- R : テープとリール。

**注2:** 全ての外囲器は鉛フリー、ハロゲン化合物フリーで完全に安全で、これらは有害物質使用制限に関する欧州指令(RoHS指令)に適合します。

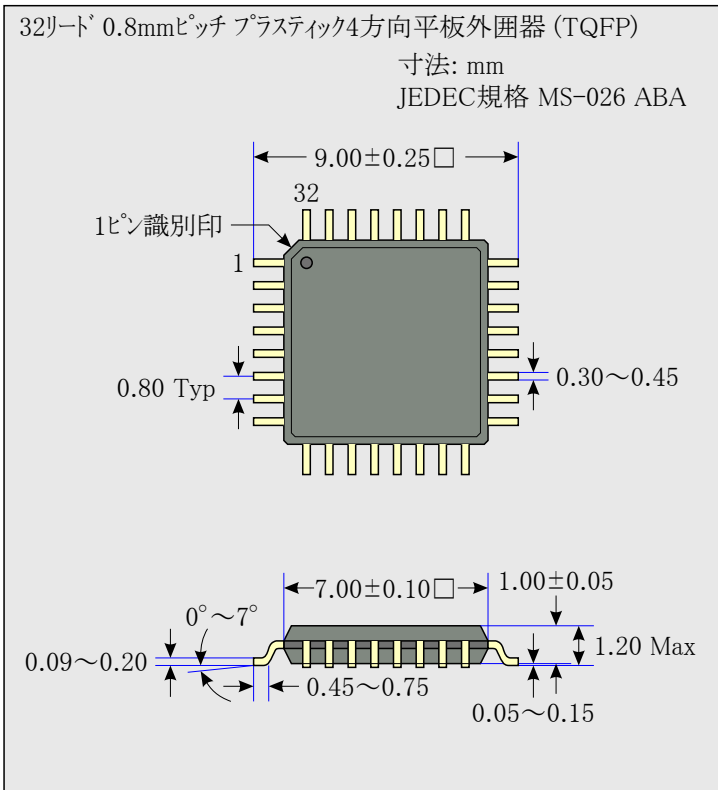
**注3:** このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

### 外囲器形式

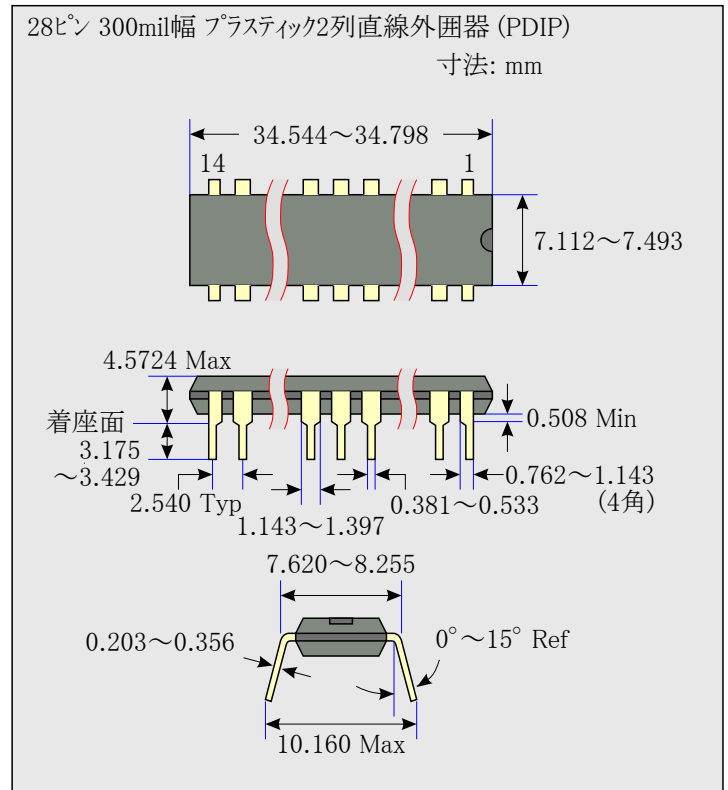
28M1	28パッド 4×4×1mm 0.45mmピッチ 4方向平板リードなし外囲器 (QFN)
28P3	28ピン 300mil幅 プラスティック2列直線外囲器 (PDIP)
32A	32リード 1.0mm厚 プラスティック4方向平板外囲器 (TQFP)
32CC1	32球(6×6配列) 4×4×0.6mm 0.50mmピッチ 超薄型密ピッチ球格子配列外囲器 (UFBGA)
32M1-A	32パッド 5×5×1mm 0.5mmピッチ 4方向平板リードなし外囲器 (QFN)

27. 外圍器情報

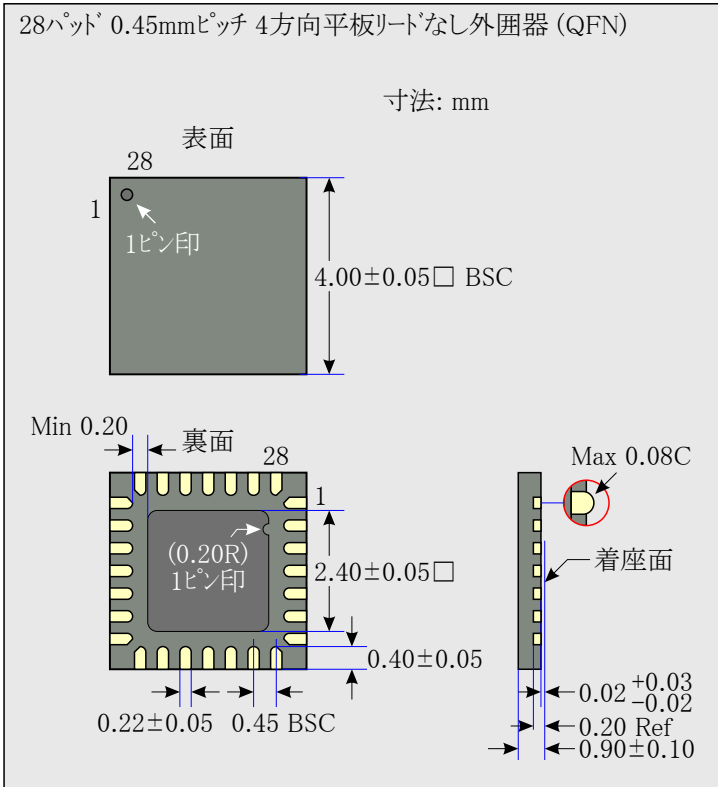
27.1. 32A



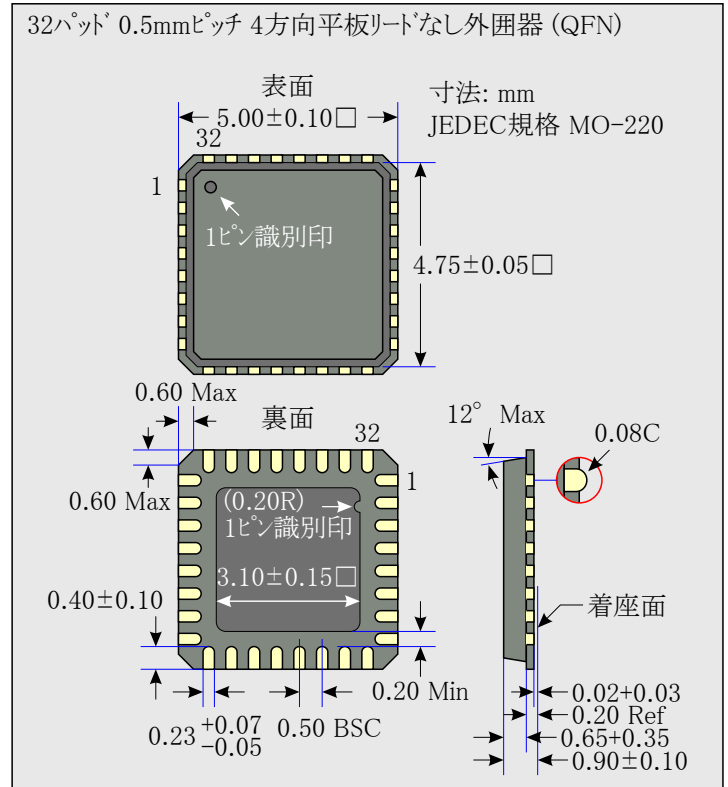
27.2. 28P3



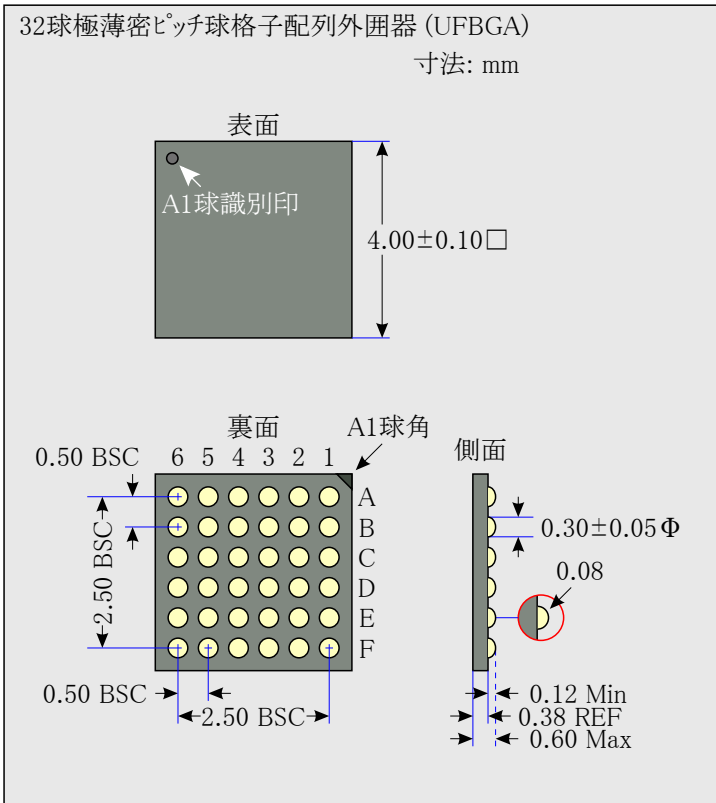
27.3. 28M1



27.4. 32M1-A



## 27.5. 32CC1



## 28. 障害情報

この章の改訂番号はATtiny48/88デバイスの改訂版を参照してください。

ATtiny48改訂A,B : 試供されていません。

ATtiny48改訂C : 既知の障害はありません。

ATtiny88改訂A : 試供されていません。

ATtiny88改訂B,C : 既知の障害はありません。

## 29. データシート改訂履歴

本章内の頁参照が本文書の現状版を参照していることに注意してください。

### 29.1. 改訂8008A – 2008年6月

1. 初版

### 29.2. 改訂8008B – 2008年6月

1. 37頁で「入出力ポート」の「序説」を更新
2. 134頁の「DC特性」を更新
3. 144頁に「代表特性」を追加

### 29.3. 改訂8008C – 2009年3月

1. 項更新
  - ・ 1頁の「特徴」
  - ・ 8頁の「リセットと割り込みの扱い」
  - ・ 15頁の「EECR – EEPROM制御レジスタ」
  - ・ 84頁の「特徴」
  - ・ 87頁の「ビット速度発生器」
  - ・ 102頁の「TWBR – TWI ビット速度レジスタ」
  - ・ 104頁の「TWSR – TWI 高速レジスタ」
  - ・ 105頁の「アナログ比較器」
  - ・ 107頁の「概要」
  - ・ 108頁の「操作」
  - ・ 108頁の「変換の開始」
  - ・ 130頁の「施錠ビット書き込み」
  - ・ 134頁の「絶対最大定格」
  - ・ 134頁の「DC特性」
  - ・ 135頁の「速度」
  - ・ 183頁の「レジスタ要約」
2. 項追加
  - ・ 17頁の「高速TWIクロック-clktwihS」
  - ・ 137頁の「アナログ比較器特性」
3. 17頁の図6-1を更新
4. リード枠合成に於ける変更を反映するために188頁の注文符号を更新

### 29.4. 改訂8008D – 2010年3月

1. 代表特性図分離、164頁に「23.2. ATtiny88」項を追加
2. 更新
  - ・ 3頁の「1.1. ピン説明」項、ポートD、「吸い込みと吐き出し」と「高い吸い込み」文字補正
  - ・ TBD修正のため、18頁の表6-3を補正(訳注:後に削除)
  - ・ TBD修正のため、18頁の「6.2.3. 128kHz内部発振器」項を補正
  - ・ 27頁の「8.4. ウォッチドッグ タイム」項を更新
  - ・ 135頁の「22.2. DC特性」項で注5と8のTBDを更新
3. 追加
  - ・ 1頁の「特徴」、2頁の「ピン配置」、188頁の「26. 注文情報」章、189頁の「27. 外圍器情報」でのUFBGA外圍器(32CC1)
  - ・ レジスタ要約への相互参照で全てのレジスタ説明表でのアドレス
  - ・ 188頁の「26. 注文情報」章でのテープとルール

### 29.5. 改訂8008E – 2010年5月

1. 183頁の24章「レジスタ要約」でアドレス\$3EにSPHを追加
2. 正しい外圍器図で189頁の27.1.項「28M1」を更新

### 29.6. 改訂8008F – 2010年6月

1. 134頁の「22.2. DC特性」の表で注の1と10を更新
2. 190頁の「27.5. 32CC1」で外圍器図を更新
3. データシート全体を通してビット表記法を更新、例えばCS0:0からCS0[2:0]へ (訳注:本書では表記法が異なるため無効)

## 29.7. 改訂8008G – 2011年4月

### 1. 更新

- ・ 4頁の「構成図」
- ・ 11頁の「メモリ」
- ・ 17頁の「クロック体系」
- ・ 121頁の「施錠ビット、ヒューズビット、デバイス識別票」
- ・ 124頁の「外部プログラミング」
- ・ 135頁の「速度」
- ・ 139頁の「2線直列インターフェース特性」

### 2. 追加

- ・ 5頁の「容量性接触感知」
- ・ 10頁の「コア関係レジスタ」
- ・ 84頁の「概要」
- ・ 101頁の「SMBusとの適合性」

### 3. 文書の状況を“暫定”から“最終”に変更

## 29.8. 改訂8008H – 2011年4月

### 1. 更新

- ・ 188頁の「注文情報」、テープとリールの符号-MMURを追加

# 目次

特徴	1	11.2. 概要	51
1. ピン配置	2	11.3. タイマ/カウンタのクロック	52
1.1. ピン説明	3	11.4. 計数器部	52
2. 概要	4	11.5. 出力比較部	52
2.1. 構成図	4	11.6. 動作種別	53
2.2. ATtiny48とATtiny88間の違い	5	11.7. タイマ/カウンタのタイミング	54
3. 一般情報	5	11.8. 8ビット タイマ/カウンタ0用レジスタ	55
3.1. 資料	5	12. 16ビット タイマ/カウンタ1	57
3.2. コード例について	5	12.1. 特徴	57
3.3. 容量性接触感知	5	12.2. 概要	57
3.4. データ保持力	5	12.3. 16ビットレジスタのアクセス	59
3.5. お断り	5	12.4. タイマ/カウンタのクロック	61
4. AVR CPU コア	6	12.5. 計数器部	61
4.1. 序説	6	12.6. 捕獲入力部	62
4.2. 構造概要	6	12.7. 比較出力部	63
4.3. ALU (Arithmetic Logic Unit)	6	12.8. 比較一致出力部	65
4.4. ステータスレジスタ	7	12.9. 動作種別	66
4.5. 汎用レジスタファイル	7	12.10. タイマ/カウンタのタイミング	70
4.6. スタックポインタ	7	12.11. 16ビット タイマ/カウンタ1用レジスタ	71
4.7. 命令実行タイミング	8	13. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器	76
4.8. リセットと割り込みの扱い	8	13.1. 内部クロック元	76
4.9. コア関係レジスタ	10	13.2. 前置分周器リセット	76
5. メモリ	11	13.3. 外部クロック元	76
5.1. プログラムメモリ	11	13.4. タイマ/カウンタ前置分周器制御用レジスタ	77
5.2. データメモリ(SRAM)とレジスタファイル	11	14. SPI (直列周辺インターフェース)	78
5.3. データメモリ(EEPROM)	12	14.1. 特徴	78
5.4. メモリ関係レジスタ	15	14.2. 概要	78
6. クロック体系	17	14.3. SSピンの機能	80
6.1. クロック副系統	17	14.4. データ転送形式	81
6.2. クロック元	17	14.5. SPI用レジスタ	82
6.3. システムクロック前置分周器	19	15. 2線インターフェース (TWI)	84
6.4. システムクロック出力緩衝部	19	15.1. 特徴	84
6.5. クロック起動手順	19	15.2. 概要	84
6.6. クロック関係レジスタ	20	15.3. バス定義	84
7. 電力管理と休止形態	21	15.4. データ転送とフレーム形式	85
7.1. 休止形態種別	21	15.5. 複数主装置バスシステムの調停と同期	86
7.2. ソフトウェア低電圧検出器(BOD)禁止	21	15.6. TWI部の概要	87
7.3. 消費電力の最小化	22	15.7. TWIの使用法	89
7.4. 電力管理用レジスタ	23	15.8. 転送種別	91
8. システム制御とリセット	25	15.9. 複数主装置システムでのバス競合と調停	100
8.1. AVRのリセット	25	15.10. SMBusとの適合性	101
8.2. リセット元	25	15.11. TWI用レジスタ	102
8.3. 内部基準電圧	27	16. アナログ比較器	105
8.4. ウォッチドッグタイマ	27	16.1. アナログ比較器入力選択	105
8.5. リセット関係レジスタ	29	16.2. アナログ比較器用レジスタ	105
9. 割り込み	31	17. A/D変換器	107
9.1. 割り込みベクタ	31	17.1. 特徴	107
9.2. 外部割り込み	32	17.2. 概要	107
9.3. 外部割り込み用レジスタ	33	17.3. 操作	108
10. 入出力ポート	37	17.4. 変換の開始	108
10.1. 序説	37	17.5. 前置分周と変換タイミング	109
10.2. 標準デジタル入出力としてのポート	37	17.6. チャネル変更と基準電圧選択	110
10.3. 交換ポート機能	41	17.7. 雑音低減機能	111
10.4. I/Oポート用レジスタ	49	17.8. アナログ入力回路	111
11. 8ビットタイマ/カウンタ0	51	17.9. アナログ雑音低減技術	111
11.1. 特徴	51	17.10. A/D変換の精度定義	112
		17.11. A/D変換の結果	112

17.12.	温度測定	112
17.13.	A/D変換用レジスタ	113
18.	デバッグWIRE内蔵デバッグ機能	116
18.1.	特徴	116
18.2.	概要	116
18.3.	物理インターフェース	116
18.4.	ソフトウェア中断点	116
18.5.	デバッグWIREの制限	116
18.6.	デバッグWIRE用レジスタ	116
19.	フラッシュの自己プログラミング	117
19.1.	自己プログラミングでのフラッシュアドレス指定	117
19.2.	自己プログラミング用レジスタ	120
20.	施錠ビット、ヒューズビット、デバイス識票	121
20.1.	施錠ビット	121
20.2.	ヒューズビット	122
20.3.	識票バイト	123
20.4.	校正バイト	123
21.	外部プログラミング	124
21.1.	メモリパラメータ	124
21.2.	並列プログラミング	124
21.3.	直列プログラミング	131
21.4.	フラッシュとEEPROM用のプログラミング時間	133
22.	電気的特性	134
22.1.	絶対最大定格	134
22.2.	DC特性	134
22.3.	速度	135
22.4.	クロック特性	136
22.5.	システムとリセットの特性	137
22.6.	アナログ比較器特性	137
22.7.	A/D変換器特性	138
22.8.	2線直列インターフェース特性	139
22.9.	SPI タイミング特性	140
22.10.	並列プログラミング特性	141
22.11.	直列プログラミング特性	143
23.	代表特性	144
23.1.	ATtiny48	144
23.2.	ATtiny88	164
24.	レジスタ要約	183
25.	命令要約	186
26.	注文情報	188
27.	外圍器情報	189
28.	障害情報	191
29.	データシート改訂履歴	192



## 本社

### *Atmel Corporation*

2325 Orchard Parkway  
San Jose, CA 95131  
USA  
TEL 1(408) 441-0311  
FAX 1(408) 487-2600

## 国外営業拠点

### *Atmel Asia Limited*

Unit 1-5 & 16, 19/F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
HONG KONG  
TEL (852) 2245-6100  
FAX (852) 2722-1369

### *Atmel Munich GmbH*

Business Campus  
Parking 4  
D-85748 Garching b. Munich  
GERMANY  
TEL (+49) 89-31970-0  
FAX (+49) 89-3194621

### *Atmel Japan*

141-0032 東京都品川区  
大崎1-6-4  
新大崎勸業ビル 16F  
アトメル ジャパン株式会社  
TEL (+81)(3)-6417-0300  
FAX (+81)(3)-6417-0370

## 製品窓口

### ウェブサイト

[www.atmel.com](http://www.atmel.com)

### 技術支援

[avr@atmel.com](mailto:avr@atmel.com)

### 販売窓口

[www.atmel.com/contacts](http://www.atmel.com/contacts)

### 文献請求

[www.atmel.com/literature](http://www.atmel.com/literature)

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のウェブサイトに位置する販売の条件と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、または利益の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。

© Atmel Corporation 2011. 不許複製 Atmel®、AVR®とそれらの組み合わせ、AVR®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

## © HERO 2022.

本データシートはAtmelのATtiny48/88英語版データシート(改訂8008H-04/11)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。

原書に対して若干構成が異なるため、一部の項番号が異なります。