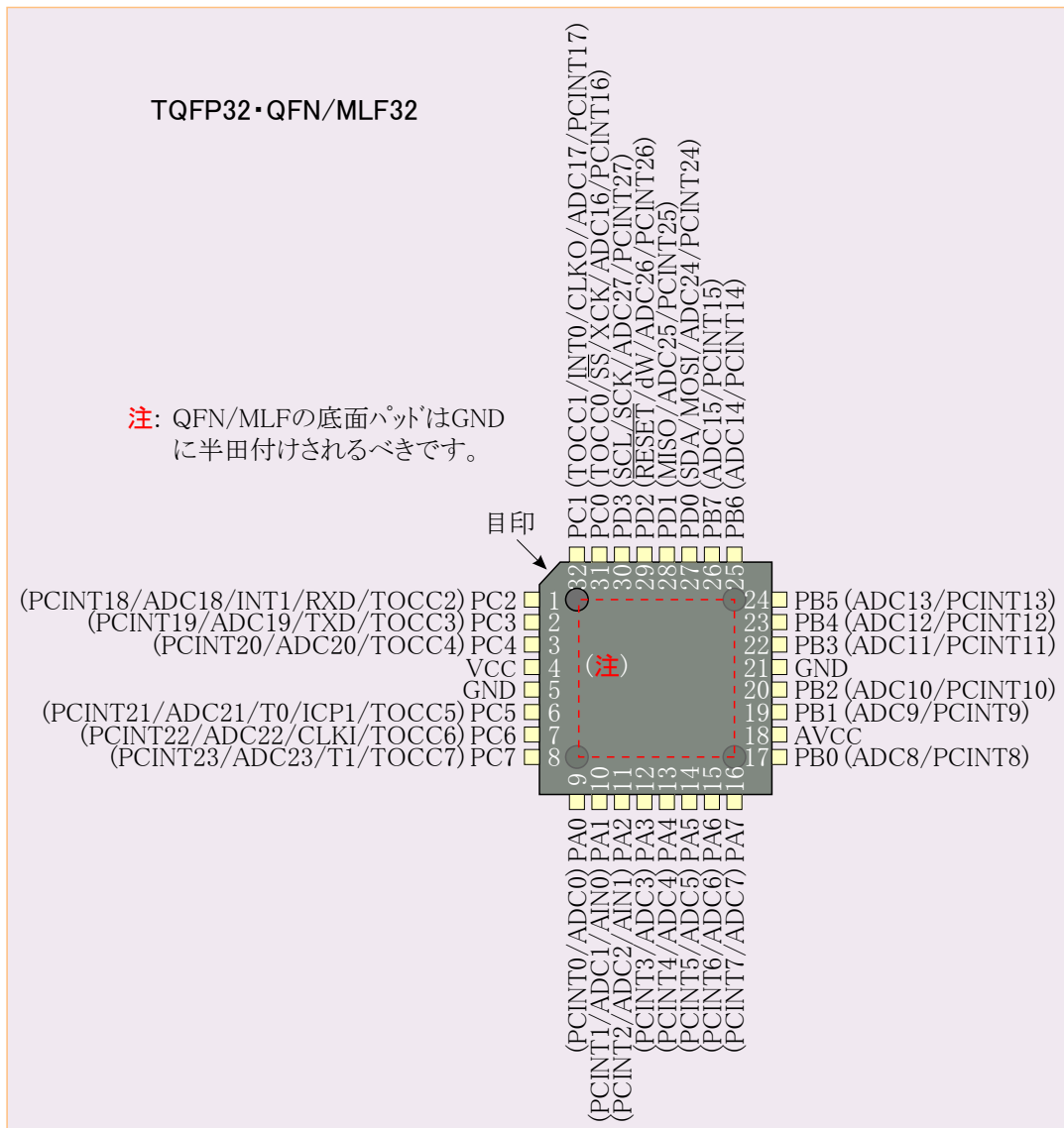


## 特徴

- 高性能、低消費Atmel® AVR® 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
  - 完全なスタティック動作
  - 20MHz時、20MIPSに達する高速動作
- データメモリと不揮発性プログラムメモリ
  - 実装書き換え可能な8Kバイト(4K語)フラッシュメモリ
    - 耐久性: 10,000回の書き換え周回
  - 実装書き換え可能な256バイトのEEPROM
    - 耐久性: 100,000回の書き換え周回
  - 512バイトの内蔵SRAM
  - 個別施錠ビットを持つ任意のブートコード領域
  - データ保持力: 20年/85°C, 100年/25°C
- 内蔵周辺機能
  - 各々2つのPWMチャネルを持つ、1つの8ビットタイマ/カウンタと1つの16ビットタイマ/カウンタ
  - 設定可能な超低電力ウォッチドッグタイマ
  - チップ上のアナログ比較器
  - 10ビットA/D変換器
    - 外部28と内部4のシングルエンド入力チャネル
  - 開始フレーム検出を持つ全2重USART
  - 主装置/従装置動作SPI直列インターフェース
  - 従装置I<sup>2</sup>C直列インターフェース
- 特殊マイクロコントローラ機能
  - 低電力アイドル、A/D変換雑音低減、パワーダウンの低消費動作形態
  - 強化した電源ONリセット回路
  - 供給電圧採取での設定可能な低電圧検出器(BOD)回路
  - 内部及び外部の割り込み元
    - 28ピンでのピン変化割り込み
  - 温度校正任意選択を持つ校正付き8MHz発振器
  - 校正付き32kHz超低電力発振器
  - 8つの入出力ピンでの高電流駆動能力
- I/Oと外圍器
  - 32リードTQFP、32パッドQFN/MLF : 28本の設定可能なI/O線
- 動作速度
  - 0~2MHz/1.7~5.5V
  - 0~4MHz/1.8~5.5V
  - 0~10MHz/2.7~5.5V
  - 0~20MHz/4.5~5.5V
- 代表消費電力(1MHz,1.8V)
  - 0.2mA (活動動作)
  - 30µA (アイドル動作)
  - 1µA (パワーダウン動作、WDT許可)
  - 0.1µA (パワーダウン動作,WDT禁止)

# 1. ピン配置



## 1.1. ピン説明

### 1.1.1. VCC

デジタル電源ピン。

### 1.1.2. AVCC

AVCCはA/D変換器と選択入出力ピン用の供給電圧(電源)ピンです。例えA/D変換が使われなくても、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、このピンは137頁の「雑音低減技術」で記述されるように低域通過濾波器を通してVCCに接続されることが推奨されます。

ポートAとポートBの全てのピンはAVCCによって給電されます。他の全ての入出力ピンはそれらの供給電圧をVCCから取ります。

### 1.1.3. GND

接地ピン。

### 1.1.4. RESET

リセット入力。最小パルス幅より長いこのピンのLowレベルは、リセットピンが禁止されていないならば、例えクロックが動作していなくてもリセットを生成します。最小パルス幅は168頁の表107.で与えられます。より短いパルスはリセットの生成が保証されません。

リセットピンは(弱い)入出力ピンとしても使うことができます。

### 1.1.5. PA7~PA0 (ポートA)

ポートAは(ビット毎に選択される)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。出力緩衝部は高い吸い込みと標準の吐き出し能力を持ちます。ポート駆動能力については166頁の表103.をご覧ください。

入力の時にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポート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、ポートピンはHi-Zになります。

このポートはピン変化割り込み、アナログ比較器、A/D変換器に関する交換ピン機能を持ちます。43頁の「交換ポート機能」をご覧ください。

### 1.1.6. PB7~PB0 (ポートB)

ポートBは(ビット毎に選択される)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。出力緩衝部は高い吸い込みと標準の吐き出し能力を持ちます。ポート駆動能力については166頁の表103.をご覧ください。

入力の時にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポート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、ポートピンはHi-Zになります。

このポートはピン変化割り込みとA/D変換器に関する交換ピン機能を持ちます。43頁の「交換ポート機能」をご覧ください。

### 1.1.7. PC7~PC0 (ポートC)

ポートCは(ビット毎に選択される)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。出力緩衝部は高い吸い込みと標準の吐き出し能力を持ちます。任意選択で、追加の高吸い込み能力を許可することができます。ポート駆動能力については166頁の表103.をご覧ください。

入力の時にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポート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、ポートピンはHi-Zになります。

このポートはピン変化割り込み、A/D変換器、タイマ/カウンタ、外部割り込み、直列インターフェースに関する交換ピン機能を持ちます。43頁の「交換ポート機能」をご覧ください。

### 1.1.8. PD3~PD0 (ポートD)

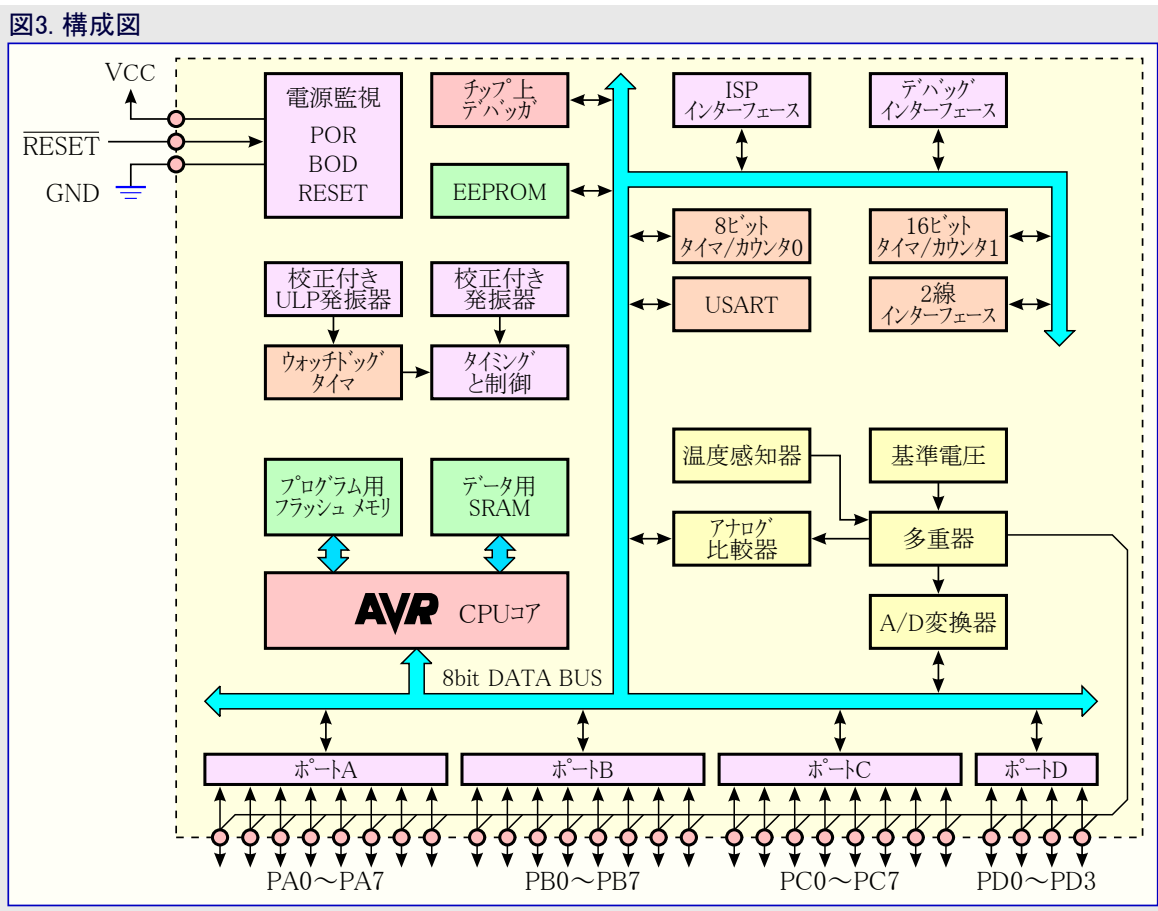
ポートDは(ビット毎に選択される)内蔵プルアップ抵抗付きの4ビットの双方向入出力ポートです。PD0とPD3の出力緩衝部は吸い込みと吐き出しの両能力で対称的な駆動特性を持ちます。PD1の出力駆動部は高い吸い込みと標準の吐き出し能力を持ち、一方PD2はそれがリセットピンとして使われるために弱い駆動特性を持ちます。ポート駆動能力については166頁の表103.をご覧ください。

入力の時にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポート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、ポートピンはHi-Zになります。

このポートはピン変化割り込み、A/D変換器、直列インターフェース、デバッグWIREに関する交換ピン機能を持ちます。43頁の「交換ポート機能」をご覧ください。

## 2. 概要

ATtiny828はAVR強化RISC構造を基にした低消費CMOS 8ビット マイクロ コントローラです。1周期での強力な命令の実行によってATtiny828はMHzあたり1MIPSに達する単位処理量を成し遂げ、処理速度対消費電力の最適化を設計者に許します。



AVRコアは32個の汎用作業レジスタと豊富な命令群の組み合わせです。32個の全レジスタはALU(Arithmetic Logic Unit)に直結され、1クロック周期で実行され、単一命令でアクセスされるべき2つの個別レジスタを許します。結果の構造は簡潔でコード効率的な一方で、現状のCISC型マイクロコントローラに対して最大10倍の単位処理量を達成します。

ATtiny828は以下の機能を提供します。

- 8Kバイトの実装書き込み可能なフラッシュメモリ
- 512バイトのSRAMデータメモリ
- 256バイトのEEPROMデータメモリ
- 28本の汎用入出力線
- 32個の汎用作業レジスタ
- 2つのPWMチャンネルを持つ1つの8ビットタイマ/カウンタ
- 2つのPWMチャンネルを持つ1つの16ビットタイマ/カウンタ
- 内部及び外部の割り込み
- 4つの内部と28個の外部チャンネルを持つ10ビットA/D変換器
- 内蔵発振器付き超低電力の設定可能なウォッチドッグタイマ
- 開始フレーム検出を持つ設定可能なUSART
- 従装置、I<sup>2</sup>C適合、2線インターフェース(TWI)
- 主装置/従装置直列周辺インターフェース(SPI)
- 校正付き8MHz発振器
- 校正付き32kHz超低電力発振器
- ソフトウェア選択可能な3つの節電動作形態

デバイスは節電用の以下の動作形態を含みます。

- **アイドル動作**：CPUを停止し、一方タイマ/カウンタ、A/D変換器、アナログ比較器、SPI、TWI、割り込み機構に機能の継続を許します。
- **A/D変換雑音低減動作**：A/D変換器を除く全I/O部とCPUを停止することによって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**：レジスタがそれらの内容を保ち、以降の割り込みまたはハードウェアリセットまで、全チップ機能が禁止されます。

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ます。プログラム用フラッシュメモリは通常の不揮発性メモリ書き込み器による直列インターフェースを通して、またはAVRコアで走行するチップ上のブートコードによって実装書き換えすることができます。ブートプログラムは応用プログラムをフラッシュメモリに設定するのにどのインターフェースをも使うことができます。ブート領域フラッシュメモリ内のソフトウェアは応用領域フラッシュメモリ更新中に実行する、真の「書き込み中の読み出し可」動作を提供します。

ATtiny828 AVRはCコンパイラ、マクロアセンブラ、プログラムデバッガ/シミュレータ、評価キットを含む完全なプログラム及びシステム開発ツールで支援されます。

## 3. 一般情報

### 3.1. 資料

包括的なデータシート、応用記述、ドライバ群と開発ツールの説明は<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

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

### 3.3. 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。

## 4. CPU コア

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

### 4.1. 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16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て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、本項内で後述されます。

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する能力があります。殆どのAVR命令は単一16ビット語(ワード)形式ですが、32ビット幅の命令も存在します。実際の命令1式はいくつかのデバイスが(全)命令1式の部分だけを実装するようになります。

プログラム用フラッシュメモリ空間はブートプログラム領域と応用プログラム領域の2つに分けられます。どちらの領域にも書き込み禁止や読み書き防止用の専用施錠ビットがあります。応用フラッシュメモリ領域内に書き込むSPM命令はブートプログラム領域内に属さ(存在し)なければなりません。

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4つの異なるアドレス指定種別を通して容易にアクセスできます。

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

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

I/Oメモリ空間は制御レジスタや他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリはデータ空間位置\$0000~\$003Fとしてアクセスできます。加えて、ATtiny828はST/STS/STDとLD/LDS/LDDの命令だけを使うことができる、データ空間内の\$0060~\$00FFの拡張I/O空間を持ちます。

### 4.2. ALU (Arithmetic Logic Unit)

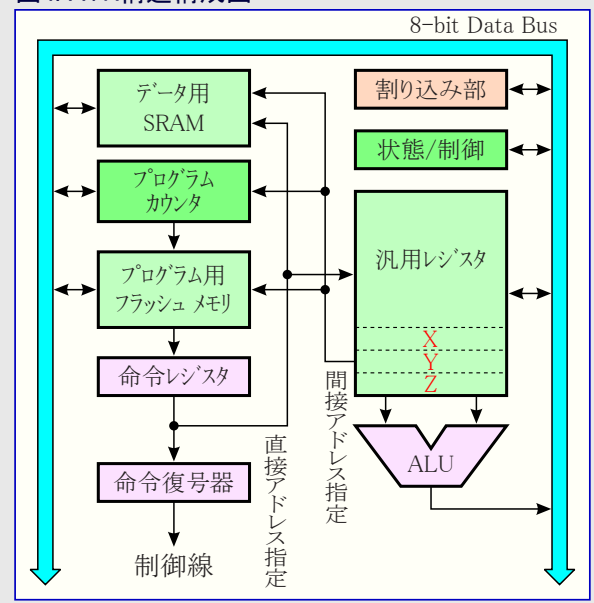
高性能なAVRのALUは16個全ての汎用レジスタに直接接続され動作します。汎用レジスタ間または汎用レジスタと即値間の演算操作は単一クロック周期内で実行されます。ALU操作は算術演算、論理演算、ビット操作の3つの主要な種類に大別されます。符号付きと符号なし両方の乗算と固定小数点形式を支援する乗算器(乗算命令)も提供する構造の実装(製品)もあります。より多くの情報については外部資料の「AVR命令一式」と204頁の「命令要約」項をご覧ください。

### 4.3. 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。より多くの情報については外部文書の「AVR命令一式」と204頁の「命令要約」章をご覧ください。

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。

図4. AVR構造構成図





## 4.4. 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5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への直接アクセスを持つ単一周期命令です。

図5.で示されるように各レジスタは使用者データ空間の最初の32位置へ直接的に配置することで、それらはデータ メモリ アドレスも割り当てられます。例えば物理的にSRAM位置として実装されていなくてもX,Y,Zレジスタ(ポインタ)がレジスタ ファイル内のどのレジスタの指示にも設定できるように、このメモリ構成は非常に柔軟なレジスタのアクセスを提供します。

図5. 汎用レジスタ構成図

7	0	アドレス	特殊機能
		R0	\$00
		R1	\$01
		R2	\$02
		⋮	
		R13	\$0D
		R14	\$0E
		R15	\$0F
		R16	\$10
		R17	\$11
		⋮	
		R26	\$1A
		R27	\$1B
		R28	\$1C
		R29	\$1D
		R30	\$1E
		R31	\$1F

Xレジスタ	下位8ビット	上位8ビット
	下位8ビット	上位8ビット
Yレジスタ	下位8ビット	上位8ビット
	下位8ビット	上位8ビット
Zレジスタ	下位8ビット	上位8ビット
	下位8ビット	上位8ビット

### 4.4.1. 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6.に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6. 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)

## 4.5. スタック ポインタ

スタックは主に一時データ、局所変数、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(SPHとSPL)レジスタは常にこのスタックの先頭(訳注:次に使われるべき位置)を指し示します。スタックが高位メモリから低位メモリへ伸長することに注意してください。これはPUSH命令がスタック ポインタ値を減少し、POP命令が増加することを意味します。

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置かれるデータSRAMのスタック領域を指し示します。このスタック空間はどのサブルーチン呼び出しの実行や割り込みの許可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格納されると-1され、POP命令でデータがスタックから引き出されると+1されます。それはサブルーチン呼び出しや割り込みで戻りアドレスがスタックに格納されると-2され、サブルーチンからの復帰(RET)命令や割り込みからの復帰(RETI)命令でアドレスがスタックから引き出されると+2されます。

AVRのスタック ポインタはI/O空間内の2つの8ビットレジスタとして実装されます。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デバイスでは全てのデータ メモリがSPLだけを使ってアドレス指定することができます。この場合、SPHレジスタは実装されません。

スタック ポインタはSRAMの最低アドレスである最低値、I/Oレジスタ領域の上の位置に設定されなければなりません。12頁の表3.をご覧ください。

## 4.6. 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)用に選択した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7はハーバ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1MIPSを達成するための基本的なパイプラインの概念です。

図8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7. 命令の取得と実行の並列動作

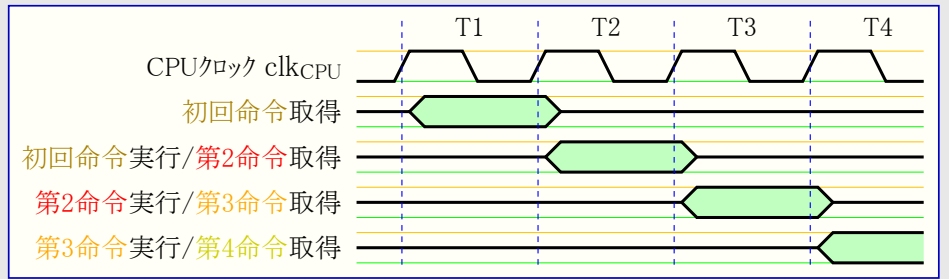
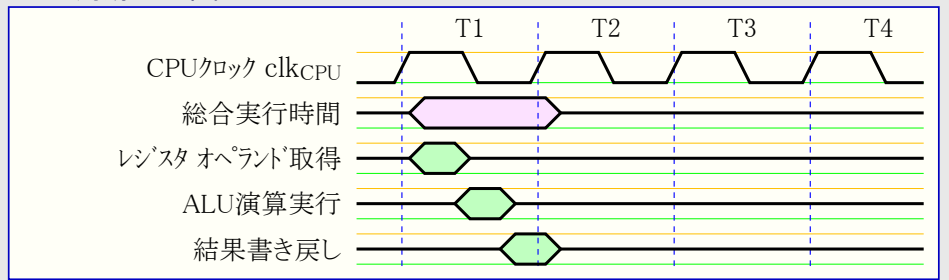


図8. 1周期ALU命令



## 4.7. 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。

ブート施錠ビット(BLB02またはBLB12)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151頁の「施錠ビット」項をご覧ください。

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32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。

割り込みベクタ表はMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットの設定(1)によってブートフラッシュ領域先頭へ移動することができます。より多くの情報については35頁の「MCUCR - MCU制御レジスタ」と32頁の「割り込み」を参照してください。リセットベクタもBOOTRSTヒューズのプログラム(0)によってブートフラッシュ領域先頭へ移動できます。145頁の「ブートローダ プログラムへの移行」をご覧ください。

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い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 */
_cli();               /* 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」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せん。

#### 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

```
_sei(); /* 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」をご覧ください。

### 4.7.1. 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3クロック周期要します。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。この増加は選択した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。

## 4.8. コア関係レジスタ

### 4.8.1. CCP – 構成設定変更保護レジスタ (Configuration Change Protection Register)

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

#### ● ビット7~0 – CCP7~0 : 構成設定変更保護 (Configuration Change Protection)

保護されたI/Oレジスタの内容を変更するため、CCPレジスタは最初に正しい識票を書かれなければなりません。CCPが書かれた後、保護されたI/Oレジスタは次からの4CPU命令周期中に書けます。これらの周期中は全ての割り込みが無視されます。これらの周期後に割り込みはCPUによって自動的に再び取り扱われて、どの保留割り込みもそれらの優先権に従って実行されます。

保護されたI/Oレジスタの識票が書かれると、CCP0は保護機能が許可されている限り1として読めます。CCP7~1は常に0として読めます。

表1. は認証に於ける識票を示します。

表1. 構成設定変更保護レジスタによって認証される識票

識票	適用群	説明
\$D8	CLKPR, MCUCR, WDTCSR (注)	保護されたI/Oレジスタ

注: WDTCSRではWDEとWDP3~0のビットだけが保護されます。

### 4.8.2. 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	

#### ● ビット9~0 – SP9~0 : スタックポインタ (Stack Pointer)

スタックポインタレジスタは上位側メモリ位置から下位側メモリ位置へ伸長するように実行されるスタックの先頭を指し示します。従ってスタックへのPUSH命令はスタックポインタを減らします。

データSRAM内のスタック空間は、どのサブルーチン呼び出し実行や割り込み許可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。

(訳補) ATtiny828の内蔵SRAMは512バイト(\$0100~\$02FF)です。従って、SRAM最終位置は\$02FF(0000 0010 1111 1111)です。

### 4.8.3. 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「**命令一式参考書**」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「**命令一式記述**」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「**命令一式記述**」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「**命令一式記述**」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「**命令一式記述**」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「**命令一式記述**」記述をご覧ください。

#### ● ビット0 – C : キャリー フラグ (Carry Flag)

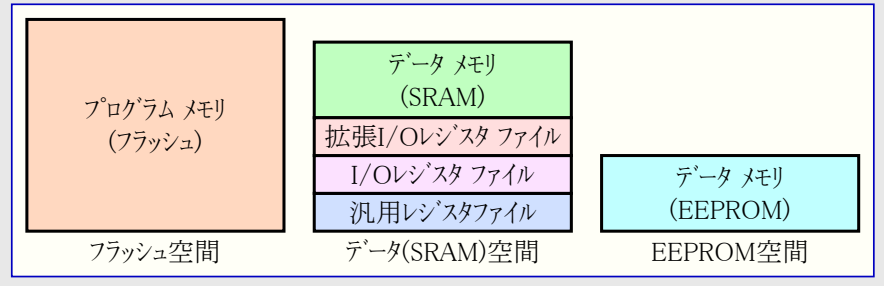
キャリー(C)フラグは算術及び論理操作でのキャリー(またはホロー)を示します。詳細情報については「**命令一式記述**」記述をご覧ください。

## 5. メモリ

AVR構造は独立したアドレス空間に各メモリ形式を配置してプログラムメモリとデータメモリを区別します。実行可能なコードは不揮発性プログラムメモリ(フラッシュ)に置かれ、一方データは揮発性(SRAM)または不揮発性メモリ(EEPROM)のどちらにも置くことができます。右の図9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9. メモリ概要



### 5.1. プログラムメモリ(フラッシュ)

ATtiny828はプログラム保存用に実装書き換え可能なチップ上の8Kバイトのフラッシュメモリを含みます。フラッシュメモリは不揮発性、換言するとそれらは例え給電されない時でも格納され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、フラッシュメモリは4096×16ビットとして構成されます。プログラムカウンタ(PC)は12ビット幅で、従って、下の表2で説明されるように、プログラムの全4096位置をアドレス指定する能力があります。

ソフトウェア保護のため、フラッシュプログラムメモリはブートプログラム領域と応用プログラム領域の2つに分けられます。より多くの詳細については147頁の「フラッシュメモリの自己プログラミング」と144頁の「応用とブートローダのフラッシュ領域」をご覧ください。

定数表はプログラムメモリのアドレス空間全体に置くことができます。204頁の「命令要約」でLPM(プログラムメモリ取得)とSPM(プログラムメモリ格納)の命令をご覧ください。プログラム用フラッシュメモリは156頁の「外部プログラミング」で記述されるように、外部装置からもプログラミングすることができます。

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を持ちます。

表2. プログラムメモリ(フラッシュ)の大きさ

デバイス	フラッシュ容量	アドレス範囲
ATtiny828	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3はATtiny828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。これらのメモリ領域は揮発性、換言すると、それらは電力が取り去れる時に情報を維持しません。

表3. データメモリとレジスタファイルの配置

デバイス	メモリ領域	容量	長アドレス(注1)	短アドレス(注2)
ATtiny828	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。このアドレス指定の形態はデータメモリとレジスタファイル全体を網羅します。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。このアドレス指定の形態はI/Oレジスタ領域の一部だけを網羅します。これはアドレスが命令語に組み込まれた命令によって使われます。

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

将来のデバイスとの互換性のため、アクセスされる場合、予約ビットは0が書かれるべきです。予約された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ます。この領域内のレジスタは主にデバイスのI/Oと周辺機能部の通信に使われます。データはIN, OUT, LD, STとその派生のような命令を用いて、I/O空間と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I/O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特有命令はI/Oレジスタ内の先頭位置を\$00とし、最後を\$3Fとしてアドレス指定します。

下位32個のレジスタ(アドレス範囲\$00～\$1F)はいくつかのビット特有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)が行われ、一方ビット条件分岐はSBIC, SBIS, SBRC, SBRSの命令を用いて容易く作成されます。

この領域のレジスタはLD/LDD/LDS/(LDI)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レジスタを\$0020で始まるアドレスで指定します。

204頁の「命令要約」をご覧ください。

ATtiny828はどんな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。200頁の「レジスタ要約」でGPOR0、GPOR1、GPOR2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のようなビット特有命令でアクセス可能なため、全域変数や状態フラグの格納に著しく有用です。

### 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。ATtiny828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/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。204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200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512位置は内部データSRAM用に予約されます。

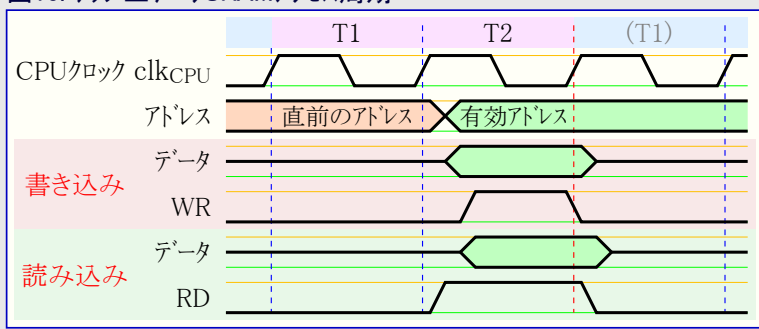
5つの利用可能なアドレス指定形態があります。

- 直接：このアドレス指定形態はデータ空間全体に届きます。
- 間接
- 変位付き間接：このアドレス指定形態はYまたはZのレジスタによって与えられる基本アドレスから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前に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 ファイルでレジスタR26~R31内に置かれます。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 ファイルでレジスタR26~R31内に置かれます。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 メモリを含む揮発性メモリ全体で使うことができます。内部SRAMは下の図10.で図解されるように、2clkCPU周期でアクセスされます。

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

図10. チップ上データSRAMアクセス周期



### 5.3. データ メモリ (EEPROM)

ATtiny828は256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することができる独立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空間に置かれます。

EEPROM配置は下の表4.で要約されます。

EEPROM操作計時に内部8MHz発振器が使われます。この発振器の周波数は21頁の「OSCCAL0 - 発振器0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。

激しく濾波した供給によって給電される時に供給電圧(VCC)は電力上昇と電力下降で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指定されるよりも低い供給電圧で走る状態にデバイスを置くかもしれません。このような状況での問題を避けるために15頁の「EEPROM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を持ちます。

表4. 不揮発性データメモリ(EEPROM)の大きさ

デバイス	EEPROM容量	アドレス範囲
ATtiny828	256バイト	\$00~\$FF

#### 5.3.1. プログラミング方法

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動作形態です。この動作形態では常に目標が書き込み前に消去されることを保証されますが、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つの違う操作で消去と書き込み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、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。これはシステムが時間に重要な操作を許す、代表的に始動と初期化時に時々行うことができます。



プログラミング方法はEEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPROM1,0)ビットを用いて選択されます。表5をご覧ください。書き込みと消去の時間はこの同じ表で与えられます。

EEPROMプログラミングは幾許かの時間がかかるため、応用は次を開始する前に1つ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(EEMPE)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位置を読むには下の手続きに従ってください。

- 他のEEPROM操作が進行中でないこと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(EEMPE)ビットをポーリングしてください。
- EEPROMアドレスレジスタ(EEARL)に目的アドレスを書いてください。
- 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。読み込み操作中、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、メモリ位置を消去するのに特別な手続きに従わなければなりません。EEPROM位置を消去するには下の手続きに従ってください。

- 他のEEPROM操作が進行中でないこと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(EEMPE)ビットをポーリングしてください。設定(1)の場合、解除(0)を待ってください。
- 自己プログラミング操作が進行中でないことを保証するためにSPM制御/状態レジスタ(SPMCSR)のSPM操作許可(SPMEN)ビットをポーリングしてください。設定(1)の場合、解除(0)を待ってください。この段階は応用がフラッシュメモリをプログラミングするブートローダを含む場合にだけ関連します、そうでなければ、この段階を省略することができます。
- EEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPROM1,0)ビットを書くことによって消去のためのプログラミング形態を設定してください。
- EEPROMアドレスレジスタ(EEARL)に目的アドレスを書いてください。
- 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(EEMPE)ビットを設定(1)することによって消去操作を開始してください。消去操作中、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。

EEMPEビットは消去操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で多忙の間、他のどのEEPROM操作を実行することも不可能です。

### 5.3.4. 書き込み

予期せぬEEPROM書き込みを防ぐため、メモリ位置を書くのに特別な手続き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は消去されなければなりません。これは同じ操作で、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終わるでしょう。

EEPROM位置を書くには下の手続きに従ってください。

- 他のEEPROM操作が進行中でないこと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(EEMPE)ビットをポーリングしてください。設定(1)の場合、解除(0)を待ってください。
- 自己プログラミング操作が進行中でないことを保証するためにSPM制御/状態レジスタ(SPMCSR)のSPM操作許可(SPMEN)ビットをポーリングしてください。設定(1)の場合、解除(0)を待ってください。この段階は応用がフラッシュメモリをプログラミングするブートローダを含む場合にだけ関連します、そうでなければ、この段階を省略することができます。
- EEPROM制御レジスタ(EECR)のEEPROMプログラミング種別(EEPROM1,0)ビットを書くことによって書き込みのためのプログラミング形態を設定してください。二者選択で、データは1操作で書くことができるか、または書き込み手続きを消去のみと書き込みのみに分割することができます。
- EEPROMアドレスレジスタ(EEARL)に目的アドレスを書いてください。
- EEPROMデータレジスタ(EEDR)に目的データを書いてください。
- 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(EEMPE)ビットを設定(1)することによって書き込み操作を開始してください。書き込み操作中、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。

EEMPEビットは書き込み操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で多忙の間、他のどのEEPROM操作も行うことが不可能です。

### 5.3.5. EEPROM化けの防止

低VCCの間中、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。

低供給電圧でEEPROM内のデータは以下の2つの点で不正にされ得ます。

- その他正当なEEPROMプログラミング手順の正しい動作を維持するの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。これは内部**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検出電圧が設計に対して不十分な場合、外部低VCC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、例えリセットが起きても、EEPROM書き込み操作は完了されます。

### 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。

本例はソフトウェア内にフラッシュブートローダが無い前提でもあります。そのようなコードが存在する場合、EEPROM書き込み関数は何れかが実行するSPM命令の完了も待たねばなりません。

#### 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<<EEP1) | (0<<EEP0) ;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 (訳注:無効)
             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<<EEP1) | (0<<EEP0); /* 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」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 (訳注:無効)
             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」をご覧ください。

## 5.4. メモリ関係レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$21 (\$41)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

#### ● ビット7~0 – EEAR7~0 : EEPROMアドレス (EEPROM Address)

EEPROMアドレスレジスタはアクセスされつつあるメモリ位置を示すのに読み/書き操作で必要とされます。

EEPROMデータバイトはメモリ範囲全体(0~(256-1))に渡って直線的にアドレス指定されます。これらのビットの初期値は未定義で従ってEEPROMがアクセスされる前に正当な値がこのレジスタに書かれなければなりません。

256バイトまたはそれ以下のEEPROMを持つデバイスは上位アドレスレジスタ(EEARH)が不要です。従ってこのようなデバイスで上位アドレスレジスタが省かれますが、互換性の問題のため、残りのレジスタは未だEEPROMアドレスレジスタの下位バイト(EEARL)として参照されます。

アドレスバイト全体を満たさないデバイス、換言すると256に等しくないEEPROMの大きさを持つデバイスは未使用位置に読み込み専用ビットを実装します。未使用ビットはアドレスレジスタの上位側に置かれ、それらは常に0を読みます。

### 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)で与えられたEEPROMアドレス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によって与えられたEEPROMアドレス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を読みます。

#### ● 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プログラム許可(EEPE)が書かれる時に起動される活動を定義します。データは新しいデータが書かれる前に前の値が自動的に消去される単一非分離操作で書くことができ、または消去と書き込みが2つの異なる操作に分けることができます。各種動作形態に対するプログラミング時間が表5.で示されます。

EEPE設定(1)時、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を除いて'00'にリセットされます。

表5. EEPROMプログラミング種別

EEPM1	EEPM0	プログラミング時間	動作
0	0	3.4ms	1操作での非分離操作(消去と書き込み)
0	1	1.8ms	消去のみ
1	0	1.8ms	書き込みのみ
1	1	–	(予約)

#### ● 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書き込みはステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているならば、EEPROM操作可割り込みを許可し、不揮発性メモリがプログラミング準備可の時に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書き込みが有効か否かの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書かれる(1)と、選択したアドレスでEEPROMがプログラミングされます。4クロック周期後にハードウェアがEEMPEビットを0に解除します。

EEMPEが0なら、EEPEビットは無効です。

● **ビット1 – EEPE : EEPROMプログラミング許可 (EEPROM Program Enable)**

これはEEPROMのプログラミング許可信号です。EEPEが(1)書かれる前にEEPROM主プログラム許可(EMPE)ビットが設定(1)されなければならず、さもなければEEPROMはプログラミングされません。

EEPEが(1)書かれると、EEPROMはEEPMMnビット設定に従ってプログラミングされます。EEPEが設定(1)されてしまうと、CPUは次の命令が実行される前に2周期間停止されます。書き込み(プログラミング)アクセス時間経過後、EEPEビットはハードウェアによって解除(0)されます。

EEPROM書き込み操作がフラッシュ、ヒューズビット、施錠ビットの全ての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。

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

これはEEPROMの読み込みストロブです。目的アドレスがEEPROMアドレスレジスタ(EEAR)で構成設定されてしまうと、EEPROM読み込み操作を起動するためにEEREビットが1を書かれなければなりません。

EEPROM読み出しアクセスは(その)1命令かかり、要求したデータは直ちに利用可能です。EEPROMが読まれる時にCPUは次の命令が実行される前に4周期間停止されます。

使用者は読み込み操作を開始する前にEEPEビットをホーリングすべきです。書き込み(プログラミング)操作が進行中の場合、EEPROMを読むことやアドレスレジスタ(EEAR)を変更することは不可能です。

**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	

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

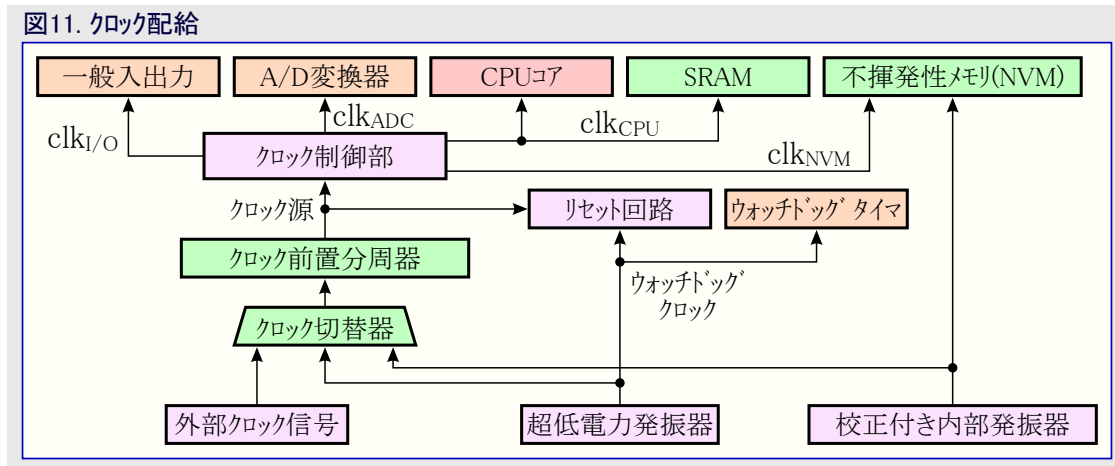
ビット	7	6	5	4	3	2	1	0	
\$2A (\$4A)	(MSB)							(LSB)	GPIOR1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

**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	

## 6. クロック体系

図11はATtiny828での主要なクロックシステムとそれらの配給を示します。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、23頁の「電力管理と休止形態動作」で記述されるように、各種休止動作形態と電力削減レジスタビットを使うことによって、使われていない単位部へのクロックを停止することができます。クロック体系は下で詳述されます。



### 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 - clkCPU

CPUクロックはAVRコアの動作と関係する系統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、システムレジスタ、SRAM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 - clkI/O

I/Oクロックはタイマ/カウンタのようなI/O部の大部分で使われます。I/Oクロックは外部割り込み部でも使われますが、いくつかの外部割り込みは例えばI/Oクロックが停止されても検出されるのをこのような割り込みに許す非同期論理回路によって検出されることに注意してください。

#### 6.1.3. NVMクロック - clkNVM

NVMクロックは不揮発性メモリ制御部の動作を制御します。NVM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 - clkADC

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る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。

### 6.2. クロック元

このデバイスはシステム クロック用に以下のどれも使うことができます。

- 外部クロック信号 (19頁をご覧ください。)
- 校正付き内部8MHz発振器 (19頁をご覧ください。)
- 内部32kHz超低電力(ULP)発振器 (19頁をご覧ください。)

クロック元は下の表6.で示されるようにCKSELヒューズを使って選ばれます。

表6. CKSELヒューズ ビットとデバイス クロック駆動任意選択

CKSEL1,0 (注1)	周波数	デバイス クロック駆動任意選択
0 x	どれも	外部クロック信号 (19頁をご覧ください。)
1 0	8MHz	校正付き内部8MHz発振器 (19頁をご覧ください。)(注2)
1 1	32kHz	内部32kHz超低電力(ULP)発振器 (19頁をご覧ください。)

注1: 全てのヒューズに対して"1"は非プログラム、"0"はプログラムを意味します。

注2: これが既定設定です。デバイスはこのヒューズの組み合わせで出荷されます。

CKSELヒューズ ビットはファームウェアによって読むことができますが(153頁の「ソフトウェアからの施錠、ヒューズ、識別データの読み出し」をご覧ください)、ファームウェアはヒューズ ビットを書くことができません。

デバイスがパワーダウンから起き上がる時に、命令実行開始前の安定な発振器動作を保证するために、始動時間に選択したクロック元が使われます。CPUがリセットから始まる時に、通常のデバイス動作が開始される前に安定な水準に達することを供給電圧に許す、付加遅延を生成するのに内部32kHz発振器が使われます。

システム クロック代替は以降の項で検討されます。



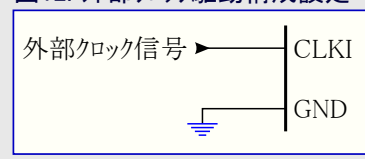
## 6.2.1. 外部クロック信号

外部クロック元からデバイスを駆動するには、下の図12.で示されるようにCLKIが接続されるべきです。このクロック元に対する始動時間は20頁の表7.で示されるように、SUTヒューズ<sup>1</sup>ビットによって決められます。

外部クロック周波数での急な変化を避けるためにMCUの安定動作を保証す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%よりも大きな周波数での変化は予測されない動きを引き起こし得ます。クロック周波数でのこのような変化中、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。

システムクロックでの大きな段階の変更に対する安定な操作はシステムクロック前置分周器を使う時に保証されます。「システムクロック前置分周器」をご覧ください。

図12. 外部クロック駆動構成設定



## 6.2.2. 校正付き内部8MHz発振器

内部8MHz発振器は外部部品なしで動作し、既定によって概ね8MHzの周波数を持つクロック元を提供します。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非常に正確に校正することができます。より多くの詳細については167頁の表104.と197頁の「内部発振器周波数」をご覧ください。

リセット中、ハードウェアが発振器校正0(OSCCAL0)レジスタ内に予めプログラムされた校正値を設定し、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104.で“工場校正”として参照されます。予めプログラムされた校正値の自動設定のより多くの情報については153頁の「校正バイト」項をご覧ください。

特に応用が狭い範囲の温度と電圧を許す時に、工場既定よりも高い精度に達することが可能です。ファームウェアは始動時や走行中のどちらでもOSCCAL0に校正データを再設定することができます。継続的な走行時校正法は電圧と温度の監視と検出したどの変化の補償もファームウェアに許します。「OSCCAL0 - 発振器校正0レジスタ」、138頁の「温度測定」、139頁の表76.をご覧ください。この校正の精度は表104.で“使用者校正”として参照されます。

発振器温度校正レジスタ(OSCTCAL0AとOSCTCAL0B)は発振器周波数の一時的な温度校正に使うことができます。「OSCTCAL0A - 発振器温度校正レジスタA」と「OSCTCAL0B - 発振器温度校正レジスタB」をご覧ください。

このクロック元に対する始動時間は20頁の表7.で示されるように、SUTヒューズ<sup>1</sup>ビットによって決められます。

## 6.2.3. 内部32kHz超低電力(ULP)発振器

内部32kHz発振器は外部部品なしで動作する低電力発振器です。概ね32kHzの周波数を持つクロック元を提供します。周波数は供給電圧、温度と一群の変化に依存します。精度の詳細については167頁の表105.をご覧ください。

リセット中、ハードウェアが発振器校正1(OSCCAL1)レジスタ内に予めプログラムされた校正値を設定し、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105.で“工場校正”として参照されます。予めプログラムされた校正値の自動設定のより多くの情報については153頁の「校正バイト」項をご覧ください。

この発振器がチップクロックとして使われる時に、未だウォッチドッグタイマとリセット時間超過に使われます。

このクロック元に対する始動時間は20頁の表7.で示されるように、SUTヒューズ<sup>1</sup>ビットによって決められます。

## 6.2.4. 既定クロック設定

このデバイスは以下のヒューズ設定で出荷されます。

- 校正付き内部8MHz発振器 (18頁の表6.でCKSELヒューズ<sup>1</sup>ビットをご覧ください。)
- 可能な最長始動時間 (20頁の表7.でSUTヒューズ<sup>1</sup>ビットをご覧ください。)
- システムクロック前置分周器を8に設定 (21頁のCKDIV8ヒューズ<sup>1</sup>ビットをご覧ください。)

既定設定は1MHzシステムクロックを与え、全ての使用者が実装または高電圧の書き込み器を使ってそれらを望むクロック元設定にすることができるのを保証します。

## 6.3. システムクロック前置分周器

ATtiny828のシステムクロックは「CLKPR - クロック前置分周レジスタ」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、clkNVM、clkI/O、clkADCは21頁の表8.で示された値によって分周されます。

### 6.3.1. 切り替え時間

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。

## 6.4. クロック出力緩衝部 (外部クロック出力)

本デバイスはシステム クロックをCLKOピンに出力することができます。この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に適します。このヒューズがプログラム(0)された時にI/Oピンの標準動作が無視され、このクロックがリセット間に出力されないことに注意してください。クロックがCLKOに出力される時は校正付き内部発振器を含むどのクロック元も選択できます。システム クロック前置分周器が使われると、出力されるのは分周されたシステム クロックです。

## 6.5. 始動時間

CKSELとSUTのヒューズ ビットは下の表7.で示されるように、デバイスの始動時間を定義します。

表7. CKSELとSUTのヒューズ ビット 対 デバイス始動時間

CKSEL	SUT	クロック元	パワーダウンから (注1,2)	リセットから (注3)
0 x	0 0	外部クロック信号	6 × CK	20 × CK
	0 1			20 × CK+4ms
	1 x			20 × CK+64ms
1 0 (注4)	0 0	内部8MHz発振器	6 × CK (注5)	20 × CK (注5)
	0 1			20 × CK+4ms
	1 x (注4)			20 × CK+64ms
1 1	0 0	内部32kHz発振器	6 × CK (注5)	20 × CK (注5)
	0 1			20 × CK+4ms
	1 x			20 × CK+64ms

注1: パワーダウン休止動作形態からのデバイス始動時間。

注2: 低電圧検出器(BOD)がソフトウェアによって禁止されていると、MCUがコード実行を続ける前にBODが正しく動作するのを保証するために、休止形態からの起き上がり時間は概ね60μsになります。

注3: リセット後のデバイス始動時間。

注4: デバイスはこの任意選択が選択されて出荷されます。

注5: リセットが禁止されている時に最小4ms。

## 6.6. クロック関係レジスタ

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

ビット (\$61)	7	6	5	4	3	2	1	0	
	-	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
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を読みます。

#### ● 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択したクロック元と内部システムクロック間の分周係数を定義します。これらのビットは応用の必要条件に合わせてクロック周波数を変えるために走行時に書くことができます。前置分周器はMCUへの主クロック入力を分周するため、これによって全ての同期周辺機能の速度が減じられます。分周係数は表8.で与えられます。

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。

表8. クロック前置分周器選択

CLKPS3	0								1								
CLKPS2	0				1				0				1				
CLKPS1	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 (注1)	2	4	8 (注2)	16	32	64	128	256	(予約)							

注1: これはCKDIV8ヒューズが非プログラム(1)にされた時の初期値です。

注2: これはCKDIV8ヒューズがプログラム(0)された時の初期値です。デバイスはCKDIV8ヒューズがプログラム(0)されて出荷されます。

クロック前置分周ビットの初期値はCKDIV8ヒューズによって決められます(152頁の表91.をご覧ください)。CKDIV8が非プログラム(1)にされる時にシステムクロック前置分周器は1に、プログラム(0)される時に8に設定されます。どの値もCKDIV8ヒューズビット設定に拘らずCLKPSビットに書くことができます。

CKDIV8がプログラム(0)される時にCLKPSビットの値は始動での8分周クロック係数を与えます。これは選択したクロック元が現在の動作条件下で許されるよりも高い周波数を持つ時に有用です。167頁の「速度」をご覧ください。

予期せぬクロック周波数の変更を防ぐため、CLKPSビットの変更するには保護された変更手順に従わなければなりません。

1. 保護されたI/Oレジスタの変更許可の識票を構成設定変更保護(CCP)レジスタに書いてください。
2. 4命令周期内に望む値でCLKPSビットを書いてください。

### 6.6.2. OSCCAL0 – 発振器校正レジスタ (Oscillator Calibration Register)

ビット (\$66)	7	6	5	4	3	2	1	0	
	CAL07	CAL06	CAL05	CAL04	CAL03	CAL02	CAL01	CAL00	OSCCAL0
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 – CAL07~0 : 発振器校正值 (Oscillator Calibration Value)

発振器校正レジスタは内部8MHz発振器の調整と発振器周波数からの変化除去処理に使われます。チップのリセット中に167頁の表104.で指定される工場校正された周波数を与える予めプログラムされた校正值が自動的にこのレジスタに書かれます。

応用ソフトウェアは発振器周波数を変更するために、このレジスタを書くことができます。この発振器は表104.で指定される周波数に校正することができます。この範囲外への校正は保証されません。

最低発振器周波数はこれらのビットを0に設定することによって達成されます。レジスタ値の増加は発振器周波数を増加します。代表的な周波数応答曲線が198頁の図182.で示されます。

この発振器がEEPROMとフラッシュメモリの書き込みアクセスに使われ、それによって書き込み時間が影響を及ぼされますことに注意してください。EEPROMやフラッシュメモリが書かれる場合、8.8MHz以上に校正しないでください。さもなければ、EEPROMやフラッシュメモリの書き込みが失敗するかもしれません。

MCUの安定な動作を保証するため、校正值は小さな間隔で変更されるべきです。或る周回から次へ2%以上の周波数での段階変更は予測不能な動きを引き起こし得ます。また、連続する2つのレジスタ値の差は\$20を超えるべきではありません。これらの限度を超える場合、MCUはクロック周波数変更の間にリセットを保たなければなりません。

### 6.6.3. OSCTCAL0A – 発振器温度校正レジスタA (Oscillator Temperature Calibration Register A)

ビット (\$F0)	7	6	5	4	3	2	1	0	
	発振器温度校正データ								OSCTCAL0A
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值								

#### ● ビット7~0 – 発振器温度校正值 (Oscillator Temperature Calibration Value)

温度校正值は校正付き8MHz発振器を調節して発振器周波数から温度変動を取り去るのに使うことができます。

### 6.6.4. OSCTCAL0B – 発振器温度校正レジスタB (Oscillator Temperature Calibration Register B)

ビット (\$F1)	7	6	5	4	3	2	1	0	
	発振器温度校正データ								OSCTCAL0B
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值								

#### ● ビット7~0 – 発振器温度校正值 (Oscillator Temperature Calibration Value)

温度校正值は校正付き8MHz発振器を調節して発振器周波数から温度変動を取り去るのに使うことができます。

### 6.6.5. OSCCAL1 – 発振器校正レジスタ (Oscillator Calibration Register)

ビット (\$67)	7	6	5	4	3	2	1	0	
	-	-	-	-	-	-	CAL11	CAL10	OSCCAL1
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	デバイス固有の校正值		

#### ● 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を読みます。

#### ● ビット1,0 – CAL11~0 : 発振器校正值 (Oscillator Calibration Value)

この発振器校正レジスタは内部32kHz発振器の調整と発振器周波数からの変化除去処理に使われます。チップのリセット中に167頁の表105で指定される工場校正された周波数を与える予めプログラムされた校正值が自動的にこのレジスタに書かれます。

応用ソフトウェアは発振器周波数を変更するために、このレジスタを書くことができます。この発振器は表105で指定される周波数に校正することができます。この範囲外への校正は保証されません。

最低発振器周波数はこれらのビットを0に設定することによって達成されます。レジスタ値の増加は発振器周波数を増加します。代表的な周波数応答曲線が199頁の図185で示されます。

MCUの安定な動作を保証するため、校正值は小さな間隔で変更されるべきです。或る周回から次へ2%以上の周波数での段階変更は予測不能な動きを引き起こし得ます。また、連続する2つのレジスタ値の差は\$20を超えるべきではありません。これらの限度を超える場合、MCUはクロック周波数変更の間にリセットを保たなければなりません。(訳注:原書での本節(前4行)は単純複写のために不適切な記述になっており、基本的に削除されるべきです。)



## 7. 電力管理と休止形態動作

高機能と産業的に先行するコード効率、低電力の応用に対してAVRマイクロコントローラを理想的に選択させます。加えて休止形態動作は応用で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動作を提供します。

### 7.1. 休止形態動作種別

18頁の図11はATtiny828の各種クロック系統とその配給を示します。この図は適切な休止形態動作選択の助けになります。表9は異なる休止形態動作と起き上がりに使える起動元を示します。

表9. 各休止形態動作に於ける動作クロック範囲と復帰起動要因

休止形態種別	発振器動作 許可された 主クロック 供給元	動作クロック範囲				復帰起動要因 (割り込み)						
		clk CPU	clk NVM	clk IO	clk ADC	ウォッチ ドッグ	INT <sub>n</sub> ピン変化	USART	従装置 TWI	A/D変換 完了	SPM/ EEPROM 準備可	その他 I/O
アイドル	○			○	○	○	○	②	○	○	○	○
A/D変換雑音低減	○				○	○	①	②	③	○	○	
パワーダウン						○	①	②	③			

①: INT<sub>n</sub> (n=0,1)についてはレベル割り込みのみです。

②: フレーム開始検出のみです。

③: アドレス一致割り込みのみです。

休止形態動作へ移行するには**休止形態制御レジスタ(SMCR)**の**休止許可(SE)ビット**が設定(1)され、**SLEEP**命令が実行されなければなりません。SMCRの**休止形態種別選択(SMn)ビット**は**SLEEP**命令によって活性(有効)にされる休止形態のどれかを選びます。一覧については表10をご覧ください。

MCUが休止形態動作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。デバイスが休止状態から起動するとき、レジスタファイルとSRAMの内容は変えられません。休止形態動作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35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

この休止形態は基本的にclk<sub>CPU</sub>とclk<sub>NVM</sub>を停止する一方、他のクロックに走行を許します。アイドル動作ではCPUが停止されますが、以下の周辺機能は動作を続けます。

- ウォッチドッグと割り込み機構
- アナログ比較器とA/D変換器
- USART、TWI、タイマ/カウンタ

アイドル動作はMCUにタイマ溢れ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き上がりも許します。アナログ比較器割り込みからの起き上がりが必要とされないなら、アナログ比較器はACSRAの**アナログ比較器禁止(ACD)ビット**を設定(1)することによって電源断にすることができます。130頁の「ACSRA - アナログ比較器制御/状態レジスタ」をご覧ください。これはアイドル動作での消費電力を削減します。

A/D変換が許可されているなら、この動作に移行すると自動的に変換が始まります。

#### 7.1.2. A/D変換雑音低減動作

この休止形態は基本的にclk<sub>IO</sub>、clk<sub>CPU</sub>、clk<sub>NVM</sub>を停止する一方、他のクロックの走行を許します。A/D変換雑音低減動作ではCPUが停止されますが、以下の周辺機能は動作を続けます。

- (許可されていれば)ウォッチドッグと外部割り込み
- A/D変換器
- USARTフレーム開始検出部とTWI

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許します。A/D変換器が許可されている場合、この動作に移行すると、変換が自動的に始まります。

以下の事象がMCUを起き上がらせ得ます。

- ウォッチドッグ リセット、外部リセット、低電圧検出(BOD)リセット
- INT0とINT1の外部レベル割り込み、ピン変化割り込み
- A/D変換完了割り込み、SPM/EEPROM操作可割り込み
- USARTフレーム開始検出、TWI従装置アドレス一致



### 7.1.3. パワーダウン動作

この休止形態は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。パワーダウン動作では発振器が停止される一方で以下の周辺機能は動作を続けます。

- (許可されていれば)ウォッチドッグ、外部割り込み

以下の事象がMCUを起き上がらせることができます。

- ウォッチドッグ リセット、外部リセット、低電圧検出(BOD)リセット
- INT0とINT1の外部レベル割り込み、ピン変化割り込み
- USARTフレーム開始検出、TWI従装置アドレス一致

## 7.2. 電力削減レジスタ

電力削減レジスタ(26頁の「PRR – 電力削減レジスタ」参照)は個別周辺機能へのクロックを停止することにより、消費電力を削減方法を提供します。周辺機能へのクロックが停止されると、以下のようになります。

- 周辺機能の現在の状態が固定化されます。
- 関連するレジスタは読み書きすることができません。
- 周辺機能によって使われる資源は専有されたままに留まります。

周辺機能は殆どの場合に於いてクロックを停止する前に禁止されるべきです。電力削減レジスタ(PRR)のビットを解除(0)することが周辺機能部を起し、停止前と同じ状態にします。

周辺機能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。例については179頁の「周辺機能部消費電流」をご覧ください。その他の休止形態動作ではクロックが予め停止されます。

## 7.3. 消費電力の最小化

これらはAVRが制御するシステムで消費電力の最小化を試みる時に考慮するためのそれぞれの検討点です。一般的に休止形態動作は可能な限り多く使われるべきで、休止形態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択されるべきです。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時、特別な考慮を必要とするでしょう。

### 7.3.1. A/D変換器 (ADC)

許可したなら、A/D変換器は全ての休止形態動作で許可されます。節電するため、休止形態動作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133頁の「A/D変換器」をご覧ください。

### 7.3.2. アナログ比較器

アイドル動作移行時に未使用ならばアナログ比較器は禁止されるべきです。A/D変換雑音削減動作移行時にアナログ比較器は禁止されるべきです。その他の休止形態動作ではアナログ比較器が自動的に禁止されます。けれども、アナログ比較器が入力として内部基準電圧を使うように構成設定される場合、アナログ比較器は全ての休止動作形態で禁止されるべきです。さもなければ、休止動作形態と無関係に内部基準電圧が許可されます。アナログ比較器の構成設定方法の詳細については130頁の「アナログ比較器」をご覧ください。

### 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BODPDヒューズによって許可されていると全休止形態動作で許可され、故に常時電力を消費します。これはより深い休止形態動作での総消費電流にとって重要な一因になります。応用に於いて低電圧検出器が必要とされる場合、この単位部は節電のために採取BOD動作形態に設定することもできます。低電圧検出器(BOD)設定法の詳細については28頁の「低電圧検出 (BOD)」をご覧ください。

### 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、またはA/D変換器によって必要とされる時に許可されます。これらの単位部が上の項で記述されるように禁止されるなら、内部基準電圧は禁止され、電力を消費しません。再びONに切り替わる時に、この出力が使われる前に使用者はこの基準電圧に始動を許さなければなりません。基準電圧が休止動作形態でONを維持されるなら、この出力は直ちに使うことができます。始動時間の詳細については168頁の表107内の内部ハントギャップ基準をご覧ください。

### 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動作で許可され、故に常時電力を消費します。これはより深い休止形態動作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29頁の「ウォッチドッグ タイマ」をご覧ください。

### 7.3.6. ポートピン

休止形態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変換器クロック( $clk_{ADC}$ )の両方が停止される休止形態動作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42頁の「[デジタル入力許可と休止形態動作](#)」をご覧ください。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(DIDR0~3)の書き込みによって禁止できます。詳細については132頁と141頁の「[DIDR0 - デジタル入力禁止レジスタ0](#)」をご覧ください。

## 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を読みます。

- ビット2,1 – SM1,0 : 休止形態種別選択 (Sleep Mode Select Bit 1 and 0)

これらのビットは表10.で示されるように、休止形態動作を選択します。

表10. 休止形態動作種別選択

SM1	SM0	休止形態動作種別
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を休止形態動作へ移行させるには、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の予期せぬ休止形態動作移行を避けるため、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. PRR – 電力削減レジスタ (Power Reduction Register)

電力削減レジスタは周辺機能クロック信号が禁止されることを許すことによって電力消費を減らす方法を提供します。

ビット	7	6	5	4	3	2	1	0	
(\$64)	PRTWI	–	PRTIM0	–	PRTIM1	PRSPI	PRUSART0	PRADC	PRR
Read/Write	R/W	R	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 – PRTWI : TWI電力削減 (Power Reduction Two-Wire Interface)

このビットへの1書き込みは2線インターフェース(TWI)部を停止します。

- 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を読みます。

- 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 : SPI電力削減 (Power Reduction SPI)

このビットへの1書き込みはこの単位部へクロックを停止することによって直列周辺インターフェース(SPI)部を停止します。再びSPIを起き上がらせる時に正しい動作を保証するためにSPIは再初期化されるべきです。

- ビット1 – PRUSART0 : USART0電力削減 (Power Reduction USART0)

このビットへの1書き込みはこの単位部へクロックを停止することによってUSART部を停止します。USART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。

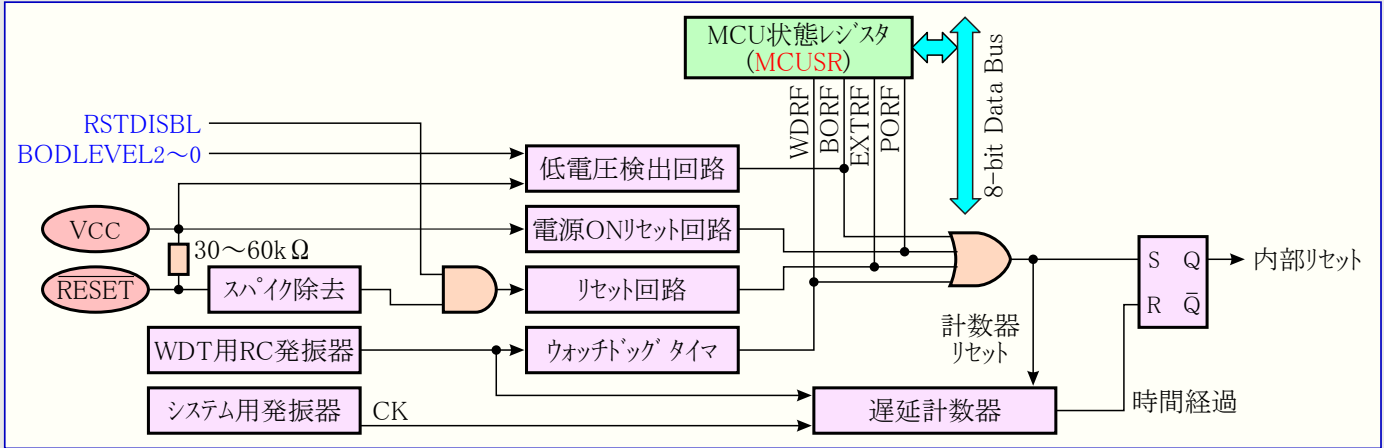
## 8. システム制御とリセット

### 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を配置することができます。これはリセットベクタが応用領域の一方、割り込みベクタがポート領域の場合やその逆も同様です

図13.の回路構成図はリセット論理回路を示します。リセット回路の電気的特性は168頁の「システムとリセットの特性」項で定義されます。

図13. 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。

### 8.2. リセット元

ATtiny828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>POA</sub>)**以下でMCUがリセットされます。
- **外部リセット** …… RESET機能許可時にRESETピンが**最小パルス幅**以上**Low**レベルに保たれると、MCUがリセットされます。
- **ウォッチドッグ リセット** …… ウォッチドッグ リセット動作が許可され、ウォッチドッグ タイマが終了すると、MCUがリセットされます。
- **低電圧リセット** …… 低電圧検出器(BOD)が許可され、供給電圧(VCC)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り、生成されます。検出電圧は168頁の「システムとリセットの特性」項で定義されます。POR信号はVCCが検出電圧以下の時は必ず活性(有効)にされます。POR回路は供給電圧異常検出は勿論、始動リセットの起動にも使えます。

電源ONリセット回路はデバイスが電源投入でリセットされることを保証します。電源ONリセット閾値電圧(V<sub>POR</sub>)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14. MCU始動、VCCに繋がれたRESET

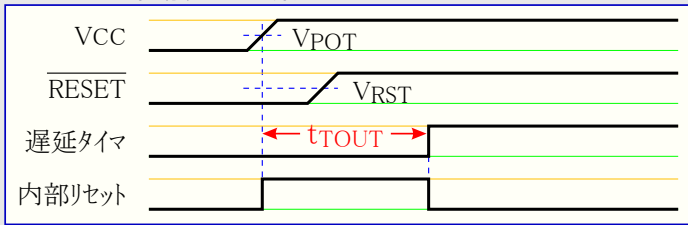
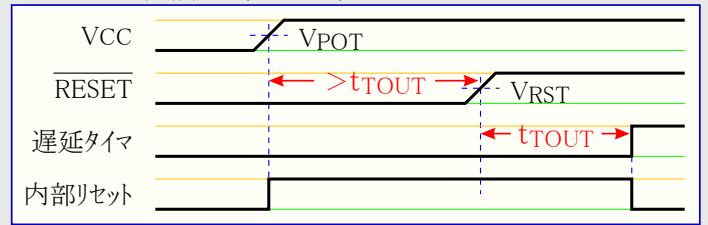


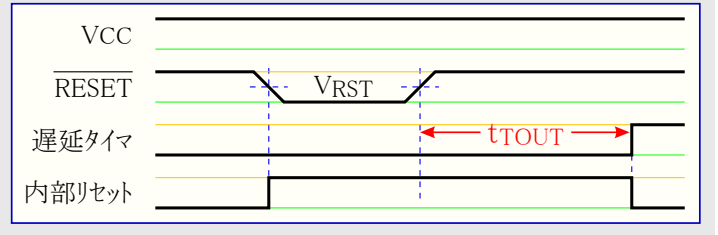
図15. MCU始動、外部的に延長されたRESET



### 8.2.2. 外部リセット

外部リセットは $\overline{\text{RESET}}$ 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(168頁の「システムとリセットの特性」項参照)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( $V_{\text{RST}}$ )に達すると(遅延タイマを起動し)、遅延タイマは遅延時間( $t_{\text{TOUT}}$ )経過後にMCUを始動します。

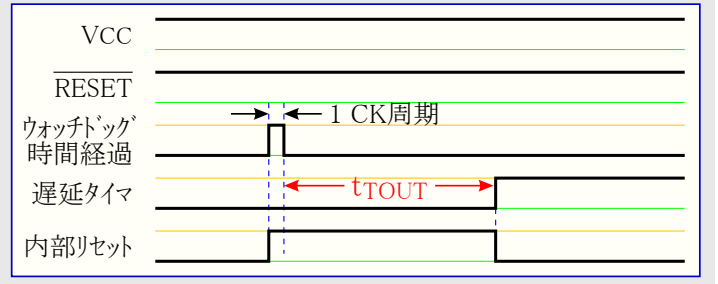
図16. 動作中の外部リセット



### 8.2.3. ウォッチドッグリセット

ウォッチドッグ時間経過時に(内部で)短いリセットパルスを生成します。遅延タイマはこのパルスの下降端で遅延時間( $t_{\text{TOUT}}$ )の計時を始めます。ウォッチドッグタイマの操作の詳細については29頁をご覧ください。

図17. 動作中のウォッチドッグリセット



### 8.2.4. 低電圧(ブラウンアウト)検出リセット

低電圧検出(BOD)回路はVCC水準が構成設定可能な検出レベル( $V_{\text{BOT}}$ )以上を維持することを監視します。BOD許可時、VCCが下降して検出時間の長さ( $t_{\text{BOD}}$ )の間、検出レベル以下に留まる時にBODリセットが与えられます。このリセットはVCCが再び検出レベル以上に上昇するまで活性を維持します。

BOD回路は電圧が検出時間( $t_{\text{BOD}}$ )の間、検出レベル以下に留まらなければ、VCCの降下を検出しません(168頁の「システムとリセットの特性」をご覧ください)。

BOD回路は以下のように3つの動作形態を持ちます。

- **禁止:** この動作形態ではVCCが監視されず、従って供給電力が安定を維持する応用だけに推奨されます。
- **許可:** この動作形態ではVCCレベルが継続的に監視されます。VCCが最低 $t_{\text{BOD}}$ 間、 $V_{\text{BOT}}$ 以下へ降下した場合に低電圧(Brown-out)リセットが生成されます。
- **採取:** この動作形態ではVCCが32kHz超低電力(ULP)発振器から配給される1kHzクロックの各負端で採取されます。各採取間でBODはOFFされます。この動作種別はBODが継続的に許可される動作形態に比べて消費電力を減らしますが、1kHzクロックの2つの正端間のVCCでの降下検出を怠ります。この動作形態で低電圧(Brown-out)が検出された場合、VCCが $V_{\text{BOT}}$ 以上に上昇するまでデハイスがリセットを維持するのを保証するために、BOD回路が許可動作形態に設定されます。BODはリセットが開放されてヒューズが読み込まれた後で採取動作形態に戻ります。

BOD動作形態はBODACTとBODPDのヒューズビットを用いて選択されます。BODACTヒューズビットは表11.で示されるように、活動動作とアイドル動作でBODがどう動作するかを決めます。

図18. 低電圧検出(BOD)リセット

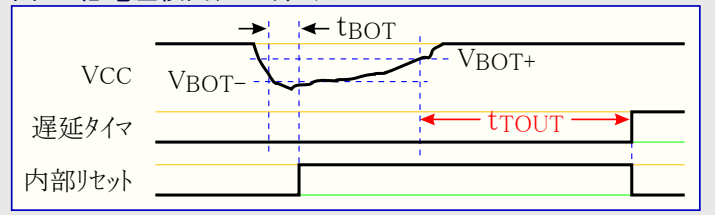


表11. 活動とアイドルの動作でのBOD動作構成設定

BODACT1	BODACT0	動作形態
0	0	(予約)
0	1	採取動作: 採取動作でBOD許可
1	0	許可動作: 継続的にBOD許可
1	1	禁止動作: BOD禁止

BODPDヒューズビットは表12.で示されるように、アイドル動作を除く全ての休止形態動作でBODがどう動作するかを決めます。

152頁の「ヒューズビット」をご覧ください。

表12. アイドル以外の休止形態でのBOD動作構成設定

BODPD1	BODPD0	動作形態
0	0	(予約)
0	1	採取動作: 採取動作でBOD許可
1	0	許可動作: 継続的にBOD許可
1	1	禁止動作: BOD禁止



### 8.3. 内部基準電圧

ATtiny828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。バンドギャップ電圧は供給電圧と温度で変化します。

#### 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を及ぼすかもしれない起動時間があります。この起動時間は、168ページの「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。

- 低電圧検出リセット許可時
- アナログ比較器内部基準電圧接続時
- A/D変換部動作許可時

従って低電圧検出(BOD)が許可されていないと、ACPMUXnの設定(=4)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

ウォッチドッグ タイマは32kHz超低電力(ULP)発振器からクロック駆動されます(19頁をご覧ください)。ウォッチドッグ タイマ前置分周器の制御によってウォッチドッグ リセット間隔は31頁の表15で示されるように調整できます。ウォッチドッグ リセット(WDR)命令はウォッチドッグ タイマをリセットします。ウォッチドッグ タイマはそれが禁止される時とチップ リセットが起こる時もリセットされます。10種の異なるクロック周期時間がこのリセット周期を決めるために選択できます。別のウォッチドッグ リセットなしにリセット周期が経過すると、ATtiny 828はリセットしてリセット ヘクタから実行します。

ウォッチドッグ タイマはリセットの代わりに割り込みを生成する設定にもできます。これはパワーダウン動作から起動するのにウォッチドッグを使う時に大変有用となり得ます。

予期せぬウォッチドッグ禁止や予期せぬ計時終了周期変更を防ぐため、2つの異なる安全レベルが表13で示されるWDTON構成設定ビットによって選択されます。詳細については次の「ウォッチドッグ タイマ構成設定変更手順」を参照してください。

図19. ウォッチドッグ タイマ構成図 (訳注:内容に合せ修正)

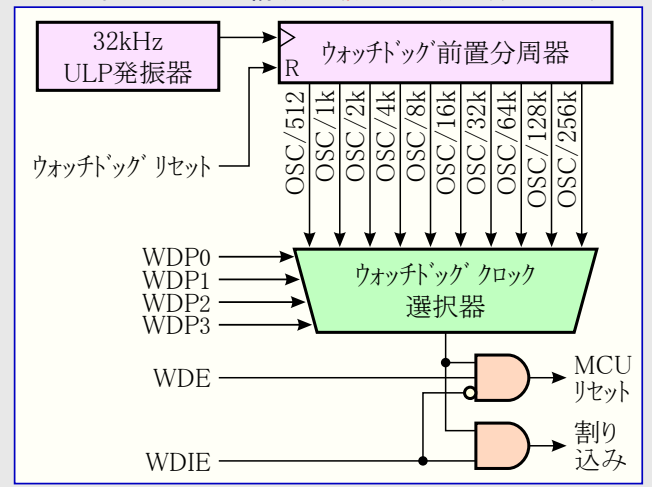


表13. WDTON構成設定ビットの設定によるウォッチドッグ機能設定

WDTON構成設定ビット	安全レベル	WDT初期状態	WDT禁止方法	計時完了時間変更方法
非プログラム(1)	1	禁止	時間制限変更手順	なし(常時可)
プログラム(0)	2	許可	なし(常時許可)	時間制限変更手順

#### 8.4.1. ウォッチドッグ タイマ構成設定変更用時間制限手順

ウォッチドッグ構成設定変更手順は安全レベル間で僅かに異なります。各レベルに対して独立した手順が記述されます。

##### ● 安全レベル1

この動作種別ではウォッチドッグ タイマが初めに禁止されますが、どんな制限もなくウォッチドッグ許可(WDE)ビットに1を書くことによって許可できます。許可したウォッチドッグ タイマを禁止する時に時間制限手順が必要とされます。許可したウォッチドッグ タイマを禁止するには次の手順に従わなければなりません。

1. 保護されたI/Oレジスタの変更許可用の識票を構成設定変更保護(CCP)レジスタに書いてください。
2. (次からの)4命令周期内に同じ操作(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)ビットを書いてください。

##### ● 安全レベル2

この動作種別ではウォッチドッグ タイマが常に許可され、WDEビットは常に1として読めます。ウォッチドッグ計時完了周期を変更する時に時間制限された変更が必要とされます。ウォッチドッグ計時完了周期を変更するには次の手順に従わなければなりません。

1. 保護されたI/Oレジスタの変更許可用の識票を構成設定変更保護(CCP)レジスタに書いてください。
2. (次からの)4命令周期内に同じ操作(命令)でWDP2~0ビットを書いてください。WDEビットに書かれた値は無関係です。

## 8.4.2. コード例

次のコード例はウォッチドッグ(WDT)をOFFに切り替える方法を示します。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:      WDR           ;ウォッチドッグ タイマリセット
              IN            R16, MCUSR      ;現MCUSR値を取得
              ANDI         R16, ~(1<<WDRF) ;WDRFビットのみ0値を取得
              OUT          MCUSR, R16      ;MCUSRのWDRFを解除(0)
              LDI         R16, $D8        ;変更許可識別値を取得
              OUT          CCP, R16       ;構成設定変更許可手順開始
              LDI         R16, (0<<WDE)   ;WDE論理0値を取得
              OUT          WDTCSR, R16    ;ウォッチドッグ禁止
              RET                    ;呼び出し元へ復帰
    
```

注: 5頁の「コード例」をご覧ください。

## 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	
\$35 (\$55)	–	–	–	–	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を読みます。

#### ● 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/状態レジスタ (Watchdog Timer Control and Status Register)

ビット (\$60)	7	6	5	4	3	2	1	0	WDTCSR
	WDIF	WDIE	WDP3	–	WDE	WDP2	WDP1	WDP0	
Read/Write	R/W	R/W	R/W	R	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 : ウォッチドッグ時間超過割り込み要求フラグ (Watchdog Timeout 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)の全体割り込み許可(I)ビットとウォッチドッグ割り込み許可(WDIE)が設定(1)されていると、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 : ウォッチドッグ時間超過割り込み許可 (Watchdog Timeout Interrupt Enable)

このビットが1を書かれ、ウォッチドッグリセット許可(WDE)ビットが解除(0)され、ステータスレジスタ(SREG)の全体割り込み許可(I)ビットが設定(1)されると、ウォッチドッグ時間超過割り込みが許可されます。この動作形態ではウォッチドッグ タイマで時間超過が起きた場合にリセットの代わりに対応する割り込みが実行されます。

WDEが設定(1)されると、時間経過発生時にハードウェアによってWDIEが自動的に解除(0)されます。これはウォッチドッグリセット保護を維持すると同時に割り込みを使うのに有用です。WDIEビット解除(0)後、次の時間経過がリセットを生成します。ウォッチドッグリセットを避け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。

表14. ウォッチドッグ タイマ構成設定

WDE	WDIE	ウォッチドッグ タイマの状態	時間超過での活動
0	0	停止	なし
0	1	走行	割り込み
1	0	走行	リセット
1	1	走行	割り込み

### ● ビット4 – Res : 予約 (Reserved)

このビットは予約されており、常に0を読みます。

### ● ビット3 – WDE : ウォッチドッグ許可 (Watchdog Enable)

WDEが論理1を書かれると、ウォッチドッグ タイマが許可され、WDEが論理0を書かれた場合にウォッチドッグ タイマ機能が禁止されます。

安全レベル2では、例え上で記述した方法でもウォッチドッグ タイマを禁止することが不可能です。29頁の「ウォッチドッグ タイマ構成設定変更用時間制限手順」をご覧ください。

安全レベル1では、WDEがMCUSRのウォッチドッグリセットフラグ(WDRF)によって無効化されます。WDRFの記述については30頁の「MCUSR – MCU状態レジスタ」をご覧ください。WDEを解除(0)するには、上で記述した手順でウォッチドッグを禁止する前に、WDRFが解除(0)されなければなりません。この特質は失敗が起こっている状況中の複数リセットと、失敗後の安全な始動を保証します。

ウォッチドッグ タイマを応用で使わないつもりなら、デバイスの初期化でウォッチドッグ禁止手順全体を行うことが重要です。例えばポイント暴走や低電圧状況によってウォッチドッグが予期せず許可された場合、デバイスは順に新しいウォッチドッグリセットを引き起こす、リセットにされるでしょう。この状況を避けるため、応用ソフトウェアは初期化ルーチン内でWDRFとWDE制御ビットを常に解除(0)すべきです。

### ● 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)

WDP3~0ビットはウォッチドッグ タイマが許可される時のウォッチドッグ タイマ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15.で示されます。

表15. 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	512	1k	2k	4k	8k	16k	32k	64k	128k	256k	(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: 選択したなら、1010未満の有効設定の1つが使われます。

これらのビットの予期せぬ変更を避けるため、以下の手順に従わなければなりません。

1. 必要とされる識票を構成設定変更保護(CCP)レジスタに書いてください。10頁をご覧ください。
2. (次からの)4命令周期内に望むビット値を書いてください。

## 9. 割り込み

この章は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」をご覧ください。

### 9.1. 割り込みベクタ

ATtiny828の割り込みベクタは下の表16.で記述されます。

表16. 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 (注1)	ラベル表記	割り込み元
1	\$0000 (注2)	RESET	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	WDT	ウォッチドッグ計時完了
9	\$0008	TIM1_CAPT	タイマ/カウンタ1捕獲発生
10	\$0009	TIM1_COMPA	タイマ/カウンタ1比較A一致
11	\$000A	TIM1_COMPB	タイマ/カウンタ1比較B一致
12	\$000B	TIM1_OVF	タイマ/カウンタ1溢れ
13	\$000C	TIM0_COMPA	タイマ/カウンタ0比較A一致
14	\$000D	TIM0_COMPB	タイマ/カウンタ0比較B一致
15	\$000E	TIM0_OVF	タイマ/カウンタ0溢れ
16	\$000F	SPI	SPI 転送完了
17	\$0010	USART0_RXS	USART0受信開始
18	\$0011	USART0_RXC	USART0受信完了
19	\$0012	USART0_DRE	USART0データレジスタ空
20	\$0013	USART0_TXC	USART0送信完了
21	\$0014	ADC_READY	A/D変換完了
22	\$0015	EE_RDY	EEPROM操作可
23	\$0016	ANA_COMP	アナログ比較器出力遷移
24	\$0017	TWI	2線インターフェース
25	\$0018	SPM_RDY	SPM命令操作可
26	\$0019	RESERVED	(予約)

**注1:** MCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)されると、割り込みベクタはブートフラッシュ領域先頭(部)へ移動されます。そして各割り込みベクタのアドレスは、この表のアドレスがブートフラッシュ領域の先頭アドレスに加算されます。

**注2:** BOOTRSTヒューズがプログラム(0)されると、デバイスはリセットでブートローダアドレスへ飛びます。145頁の「ブートローダプログラムへの移行」をご覧ください。

プログラムが決して割り込み元を許可しない条件では、割り込みベクタは使われず、結果としてこれらの位置に通常のプログラムコードを置くことができます。これはリセットベクタが応用領域の一方、割り込みベクタがブート領域、またはその逆の場合でも同様です。

ATtiny828での最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定は下のプログラム例で示されます。

アドレス	ラベル	命令	注釈
		. org \$0000	;次命令行開始アドレス
\$0000		RJMP RESET	;各種リセット
\$0001		RJMP INTO_ISR	;外部割り込み要求0
\$0002		RJMP INT1_ISR	;外部割り込み要求1
\$0003		RJMP PCINT0_ISR	;ピン変化0群割り込み要求
\$0004		RJMP PCINT1_ISR	;ピン変化1群割り込み要求
\$0005		RJMP PCINT2_ISR	;ピン変化2群割り込み要求
\$0006		RJMP PCINT3_ISR	;ピン変化3群割り込み要求
\$0007		RJMP WDT_ISR	;ウォッチドッグ計時完了
\$0008		RJMP TIM1_CAPT_ISR	;タイマ/カウンタ1捕獲発生
\$0009		RJMP TIM1_COMP_A_ISR	;タイマ/カウンタ1比較A一致
\$000A		RJMP TIM1_COMP_B_ISR	;タイマ/カウンタ1比較B一致
\$000B		RJMP TIM1_OVF_ISR	;タイマ/カウンタ1溢れ
\$000C		RJMP TIM0_COMP_A_ISR	;タイマ/カウンタ0比較A一致
\$000D		RJMP TIM0_COMP_B_ISR	;タイマ/カウンタ0比較B一致
\$000E		RJMP TIM0_OVF_ISR	;タイマ/カウンタ0溢れ
\$000F		RJMP SPI_ISR	;SPI転送完了
\$0010		RJMP USART0_RXS_ISR	;USART0受信開始
\$0011		RJMP USART0_RXC_ISR	;USART0受信完了
\$0012		RJMP USART0_DRE_ISR	;USART0データレジスタ空
\$0013		RJMP USART0_TXC_ISR	;USART0送信完了
\$0014		RJMP ADC_ISR	;A/D変換完了
\$0015		RJMP EE_RDY_ISR	;EEPROM操作可
\$0016		RJMP ANA_COMP_ISR	;アナログ比較器出力遷移
\$0017		RJMP TWI_ISR	;2線インターフェース
\$0018		RJMP SPM_RDY_ISR	;SPM命令操作可
\$0019		RJMP RESERVED	; (予約)
;			
\$001A	RESET:	<命令>	;主プログラム開始 (以下命令)

注: 5頁の「コード例」をご覧ください。

表17はBOOTRST(ヒューズ)とIVSEL(割り込みベクタ選択ビット)の様々な組み合わせに対するリセットと割り込みのベクタ配置を示します。

表17. リセットと割り込みのベクタ配置

BOOTRST	IVSEL	リセットベクタアドレス	割り込みベクタ先頭アドレス
非プログラム(1)	0	\$0000	\$0001
	1	\$0000	ブートリセットアドレス(注)+\$0001
プログラム(0)	0	ブートリセットアドレス(注)	\$0001
	1	ブートリセットアドレス(注)	ブートリセットアドレス(注)+\$0001

注: ブートリセットアドレスは145頁の表82で示されます。



以下は次の場合に対するプログラム例です。

- BOOTRSTヒューズは非プログラム(1)
- ブート領域容量は2Kバイトに設定
- どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットを設定(1)

アドレス	ラベル	命令	注釈
\$0000	RESET:	. ORG \$0000 <命令> }	;アドレス \$0000 ;主プログラム開始 ;(以下命令)
		. ORG \$0C01	;アドレス \$0C01
\$0C01		RJMP INTO_ISR	;外部割り込み要求0
\$0C02		RJMP INT1_ISR }	;外部割り込み要求1
\$0C18		RJMP SPM_RDY_ISR	;SPM命令操作可
\$0C19		RJMP RESERVED	; (予約)

注: 5頁の「コード例」をご覧ください。

以下は次の場合に対するプログラム例です。

- BOOTRSTヒューズはプログラム(0)
- ブート領域容量は2Kバイトに設定

アドレス	ラベル	命令	注釈
		. ORG \$0001	;アドレス \$0001
\$0001		RJMP INTO_ISR	;外部割り込み要求0
\$0002		RJMP INT1_ISR }	;外部割り込み要求1
\$0018		RJMP SPM_RDY_ISR	;SPM命令操作可
\$0019		RJMP RESERVED }	; (予約) ;以下、プログラムなど
		. ORG \$0C00	;アドレス \$0C00
\$0C00	RESET:	<命令> }	;主プログラム開始 ;(以下命令)

注: 5頁の「コード例」をご覧ください。

以下は次の場合に対するプログラム例です。

- BOOTRSTヒューズはプログラム(0)
- ブート領域容量は2Kバイトに設定
- どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットを設定(1)

アドレス	ラベル	命令	注釈
		. ORG \$0C00	;アドレス \$0C00
\$0C00		RJMP RESET	;各種リセット (BOOTRSTヒューズ=0)
\$0C01		RJMP INTO_ISR	;外部割り込み要求0
\$0C02		RJMP INT1_ISR }	;外部割り込み要求1
\$0018		RJMP SPM_RDY_ISR	;SPM命令操作可
\$0019		RJMP RESERVED	; (予約)
;			
\$0C1A	RESET:	<命令> }	;主プログラム開始 ;(以下命令)

注: 5頁の「コード例」をご覧ください。

## 9.2. 外部割り込み

外部割り込みはINT0とINT1のピンまたはPCINTnピンの何れかによって起動されます。許可したなら、例えINTnまたはPCINTnピンが出力として設定されても、割り込みが起動することに注目してください。この特徴はソフトウェア割り込みを生成する方法を提供します。

ピン変化割り込みは以下のように起動します。

- **ピン変化0群割り込み(PCI0)**は許可したPCINT0～7ピンの何れかが切り替わる場合に起動します。
- **ピン変化1群割り込み(PCI1)**は許可したPCINT8～15ピンの何れかが切り替わる場合に起動します。
- **ピン変化2群割り込み(PCI2)**は許可したPCINT15～23ピンの何れかが切り替わる場合に起動します。
- **ピン変化3群割り込み(PCI3)**は許可したPCINT24～27ピンの何れかが切り替わる場合に起動します。

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、そしてそれはそれらの割り込みがアイドル動作以外の休止形態動作からもデバイス起動するのに使えることを意味します。

ピン変化割り込み(PCINT)が生成されるにはデバイスが活性なI/Oクロックを持たなければなりません。23頁の表9.で示されるようにI/Oクロック領域はアイドル動作で活性ですが、より深い休止形態動作では違います。アイドル動作よりも深い休止形態動作で、切り替えたピンはデバイスが完全に起こされるまでその切り替えた状態に留まります。起き上がり時間については20頁の表7.をご覧ください。起き上がり中にピンがその初期状態へ元に戻した場合、デバイスは未だ手続きを完了しますが、一度起きても割り込みを生成しません。

外部割り込みのINT0とINT1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。INT0とINT1はレベル起動として構成設定されて許可されると、そのピンがLowに保持される限り、割り込みが(継続的に)起動します。

INT0とINT1の上昇端または下降端の割り込みの認知は18頁の「クロック体系」で記述されるように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. Lowレベル割り込み

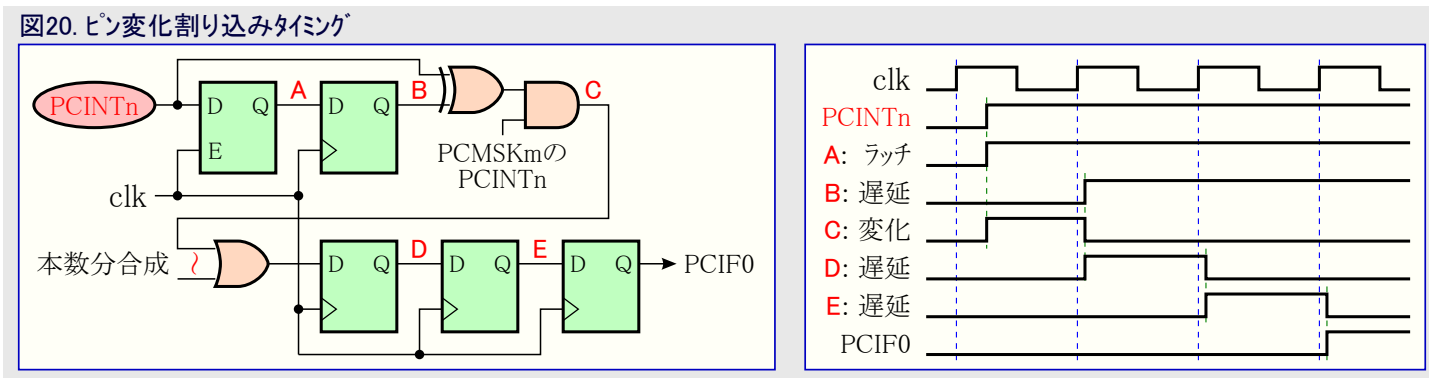
INT0やINT1のLowレベル割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態動作からもデバイスを起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態動作で停止されます。

レベル起動割り込みがパワーダウン動作からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが起動時間の最後に先立ち消滅すると、MCUは今までどおり起動しますが、割り込みが生成されません。起動時間は18頁の「クロック体系」で記述されるように定義されます。

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。

### 9.2.2. ピン変化割り込みタイミング

ピン変化割り込みのタイミング例は図20.で示されます。



## 9.3. 割り込み用レジスタ

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

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

- **ビット7～2.0 – Res : 予約 (Reserved)**

これらのビットは予約されており、常に0を読みます。

- **ビット1 – IVSEL : 割り込みベクタ選択 (Interrupt Vector Select)**

このビットが解除(0)されると、割り込みベクタはフラッシュメモリの始めに置かれます。このビットが設定(1)されると、割り込みベクタはポートロータ領域の先頭に移動されます。

ブート領域の開始アドレスはBOOTSZヒューズによって決められます。詳細については145頁の「ブートローダ構成設定」をご覧ください。

割り込みベクタがブートローダ領域に置かれ、ブート施錠ビットのBLB02がプログラム(0)される場合、応用領域から実行する間、割り込みが禁止されます。

割り込みベクタが応用領域に置かれ、ブート施錠ビットのBLB12がプログラム(0)される場合、ブートローダ領域から実行する間、割り込みが禁止されます。

このビットへの予期せぬ変更を避けるため、以下の手順に従わなければなりません。

1. 必要とされる識票を構成設定変更保護(CCP)レジスタに書いてください。10頁をご覧ください。
2. (次からの)4命令周期内に望むビット値をIVSELに書いてください。

### 9.3.2. PCMSK3 – ピン変化割り込み許可レジスタ3 (Pin Change Mask Register 3)

ビット (\$73)	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を読みます。

#### ● 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PCINTnビットは対応するI/Oピンのピン変化割り込みが許可されるかどうかを選びます。ピンでのピン変化割り込みはピンに対する許可ビット(PCINTn)とピン変化割り込み制御レジスタ(PCICR)の対応するn群許可(PCIEEn)ビットの設定(1)によって許可されます。

このビットが解除(0)されると、対応するピンのピン変化割り込みが禁止されます。

### 9.3.3. PCMSK2 – ピン変化割り込み許可レジスタ2 (Pin Change Mask Register 2)

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PCINTnビットは対応するI/Oピンのピン変化割り込みが許可されるかどうかを選びます。ピンでのピン変化割り込みはピンに対する許可ビット(PCINTn)とピン変化割り込み制御レジスタ(PCICR)の対応するn群許可(PCIEEn)ビットの設定(1)によって許可されます。

このビットが解除(0)されると、対応するピンのピン変化割り込みが禁止されます。

### 9.3.4. PCMSK1 – ピン変化割り込み許可レジスタ1 (Pin Change Mask Register 1)

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PCINTnビットは対応するI/Oピンのピン変化割り込みが許可されるかどうかを選びます。ピンでのピン変化割り込みはピンに対する許可ビット(PCINTn)とピン変化割り込み制御レジスタ(PCICR)の対応するn群許可(PCIEEn)ビットの設定(1)によって許可されます。

このビットが解除(0)されると、対応するピンのピン変化割り込みが禁止されます。

### 9.3.5. PCMSK0 – ピン変化割り込み許可レジスタ0 (Pin Change Mask Register 0)

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PCINTnビットは対応するI/Oピンのピン変化割り込みが許可されるかどうかを選びます。ピンでのピン変化割り込みはピンに対する許可ビット(PCINTn)とピン変化割り込み制御レジスタ(PCICR)の対応するn群許可(PCIEEn)ビットの設定(1)によって許可されます。

このビットが解除(0)されると、対応するピンのピン変化割り込みが禁止されます。

### 9.3.6. 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	
	-	-	-	-	ISC11	ISC10	ISC01	ISC00	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 – 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

#### ● ビット3,2 – ISC11,0 : 外部割り込み1条件制御 (Interrupt Sense Control, INT1)

#### ● ビット1,0 – ISC01,0 : 外部割り込み0条件制御 (Interrupt Sense Control, INT0)

外部割り込みのINT0とINT1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)の対応する割り込み許可(INT0またはINT1)ビットが設定(1)される場合のINT0とINT1のピンでの活動によって起動されます。割り込みを起動するのに必要とされる条件は表18.で定義されます。

表18. 外部割り込み感知制御

ISCn1	ISCn0	説明
0	0	INT0/INT1のLowレベルが割り込み要求を生成します。(注1)
0	1	INT0/INT1でのどの論理変化(両端も割り込み要求を生成します。(注2)
1	0	INT0/INT1の下降端が割り込み要求を生成します。(注2)
1	1	INT0/INT1の上昇端が割り込み要求を生成します。(注2)

**注1:** Lowレベル割り込みが選択される場合、割り込みを生成するのに、そのLowレベルは現在実行中の命令の完了まで保持されなければなりません。

**注2:** INT0/INT1ピン上の値はエッジを検出するのに先立って採取されます。エッジまたは切り替わりの割り込みが選択される場合、1クロック周期よりも長く持続するパルスが割り込みを生成します。より短いパルスは割り込みの生成が保証されません。

### 9.3.7. PCICR – ピン変化割り込み制御レジスタ (Pin Change Interrupt Control Register)

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 – 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

#### ● ビット3 – PCIE3 : ピン変化3群割り込み許可 (Pin Change Interrupt Enable 3)

この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)されると、ピン変化割り込み3が許可されます。許可したPCINT24~27ピンのどの変化もピン変化3群割り込みを引き起こします。32頁の表16.をご覧ください。

各ピンは個別に許可することができます。前頁の「PCMSK3 – ピン変化割り込み許可レジスタ3」をご覧ください。

#### ● ビット2 – PCIE2 : ピン変化2群割り込み許可 (Pin Change Interrupt Enable 2)

この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)されると、ピン変化割り込み2が許可されます。許可したPCINT16~23ピンのどの変化もピン変化2群割り込みを引き起こします。32頁の表16.をご覧ください。

各ピンは個別に許可することができます。前頁の「PCMSK2 – ピン変化割り込み許可レジスタ2」をご覧ください。

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

この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)されると、ピン変化割り込み1が許可されます。許可したPCINT8~15ピンのどの変化もピン変化1群割り込みを引き起こします。32頁の表16.をご覧ください。

各ピンは個別に許可することができます。前頁の「PCMSK1 – ピン変化割り込み許可レジスタ1」をご覧ください。

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

この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)されると、ピン変化割り込み0が許可されます。許可したPCINT0~7ピンのどの変化もピン変化0群割り込みを引き起こします。32頁の表16.をご覧ください。

各ピンは個別に許可することができます。前頁の「PCMSK0 – ピン変化割り込み許可レジスタ0」をご覧ください。

### 9.3.8. 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 – 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

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

この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)される時にINT1ピンに対する外部割り込みが許可されます。起動条件は外部割り込み制御レジスタA(EICRA)の割り込み条件制御1(ISC1n)ビットで設定されます。

例えINT1ピンが出力として設定されていても、このピンの活動は割り込み要求を引き起こします。

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

この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)される時にINT0ピンに対する外部割り込みが許可されます。起動条件は外部割り込み制御レジスタA(EICRA)の割り込み条件制御0(ISC0n)ビットで設定されます。

例えINT0ピンが出力として設定されていても、このピンの活動は割り込み要求を引き起こします。

### 9.3.9. 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 – 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

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

INT1ピンでの活動が割り込み要求を起動する時にこのビットが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(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処理ルーチンが実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。

INT1がレベル割り込みとして設定される時に、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ピンでの活動が割り込み要求を起動する時にこのビットが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(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処理ルーチンが実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。

INT0がレベル割り込みとして設定される時に、このフラグは常に解除(0)されます。

### 9.3.10. PCIFR – ピン変化割り込み要求フラグ レジスタ (Pin Change Interrupt Flag Register)

ビット	7	6	5	4	3	2	1	0	
\$1B (\$3B)	-	-	-	-	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 – 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

- 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時にこのビットが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(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実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。



- **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時にこのビットが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(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実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。

- **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)

PCINT8～15ピンの何れかの論理変化が割り込み要求を起動する時にこのビットが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(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実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。

- **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時にこのビットが設定(1)されます。**ステータスレジスタ(SREG)の全割り込み許可(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実行される時に解除(0)されます。代わりに、このフラグは論理1書き込みによっても解除(0)することができます。

## 10. 入出力ポート

### 10.1. 概要

全てのAVRのポートは標準デジタルI/Oポートとして使われる時に真の読み-変更-書き(リード-モディファイ-ライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方向をも無意識に変更することなく、1つのポートピンの方向を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。

このピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピンは個別に選択可能な、供給電圧で抵抗値が変化しないプルアップ抵抗を持ちます。全てのI/Oピンは図21.で示されるように、VCCとGNDの両方に保護ダイオードがあります。各値の完全な一覧については166頁の「電気的特性」を参照してください。

本項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PORTx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53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)、プルアップ許可レジスタ(PUEx)、入力レジスタ(PINx)の各ポートに対して、4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、方向レジスタ、プルアップ許可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。

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43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目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22.はここで属にPxnと呼ばれるI/Oポートピンの1つの機能説明を示します。

図21. 入出力ピン等価回路

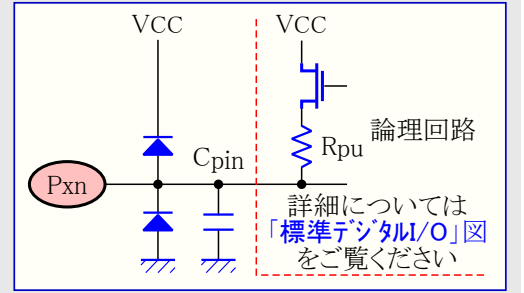
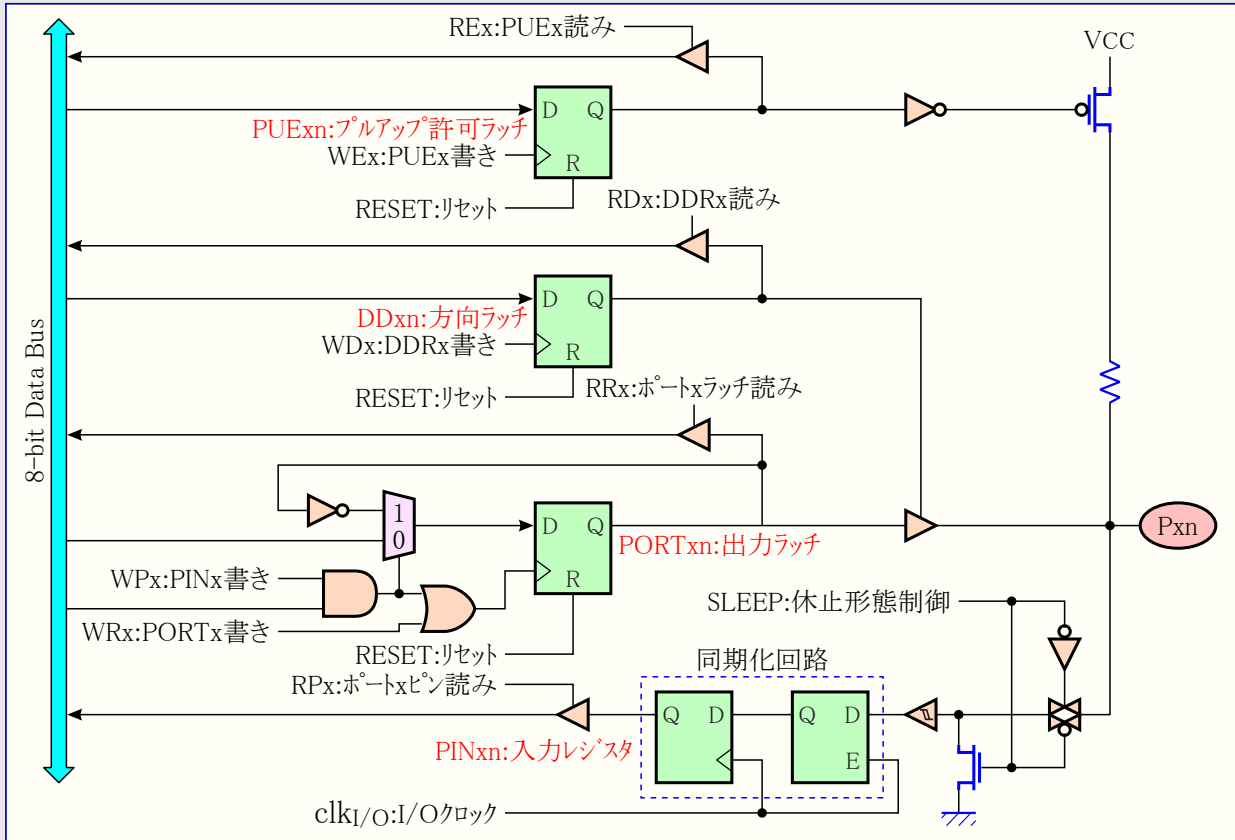


図22. 標準デジタル入出力回路構成



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

### 10.2.1. ピンの構成設定

各ポートピンはDDxn、PUExn、PORTxn、PINxnの4つのレジスタビットから成ります。53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位置、PUExnビットはPUEx I/O位置、PORTxnビットはPORTx I/O位置、PINxnビットはPINx I/O位置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択します。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。

PUExnが論理1を書かれた場合に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PUExnが論理0を書かれなければなりません。

表19.はピン値に対する制御信号を要約します。

DDxn	PORTxn	PUExn	入出力	プルアップ抵抗	備考
0	x	0	入力	なし	高インピーダンス (Hi-Z)
0	x	1	入力	あり	外部的にLowへ引かれた場合に電流を吐き出します。
1	0	0	出力	なし	Low (吸い込み)出力
1	0	1	出力	あり	<b>非推奨:</b> Low (吸い込み)出力、内部プルアップ活性(有効) 内部プルアップ抵抗を通して電流を吐き出し、常に電力を消費します。
1	1	0	出力	なし	High (吐き出し)出力
1	1	1	出力	あり	High (吐き出し)出力、内部プルアップ活性(有効)

ポートピンは例え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。

### 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. ピン値の読み込み

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

(図23.で)システムクロックの最初の上昇端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(透明)となります。この信号値はシステムクロックがLowになる時に保持(ラッチ)されます。それが続くクロックの上昇端でPINxnレジスタに取り込まれます。2つの矢印 $t_{pd,min}$ と $t_{pd,max}$ によって示されるように、ピン上の単一信号遷移は出現時点に依存して0.5~1.5システムクロック周期遅らされます。

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

図23. 外部供給ピン値読み込み時の同期化

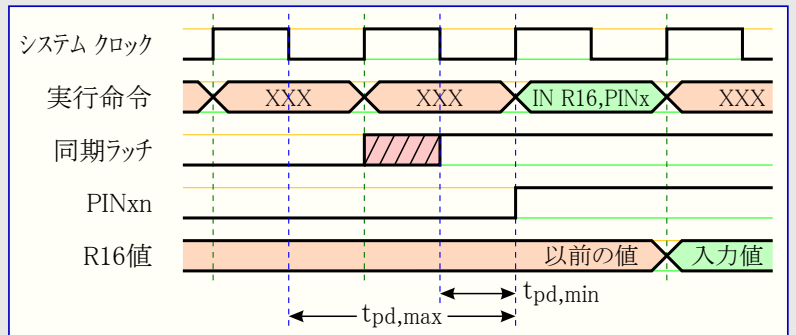
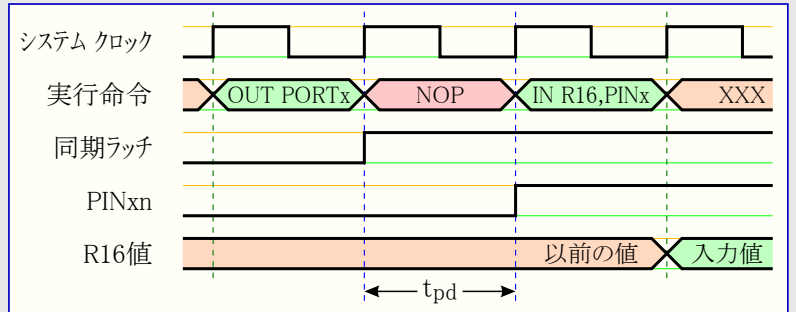


図24. プログラムで設定したピン値読み戻し時の同期化



## 10.2.4. デジタル入力許可と休止形態動作

図22.で示されるようにデジタル入力信号はシュミットトリガの入力をGNDにクランプできます。この図でSLEEPと印された信号は入力信号のいくつかが開放のまま、またはVCC/2付近のアナログ信号電圧を持つ場合の高消費電力を避けるため、パワーダウン動作で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43ページの「交換ポート機能」で記載されるように様々な他の交換機能によっても無視されます。

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動作から(復帰)再開する時に、これらの休止形態動作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。

## 10.2.5. 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ことの保証が推奨されます。例え上記のような深い休止形態動作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、外部の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。

## 10.2.6. プログラム例

次のコード例はポートBピンの0をHigh出力、1をLow出力、ポートピンの2に割り当てられたプルアップと共に2~3のポートピンを入力として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。

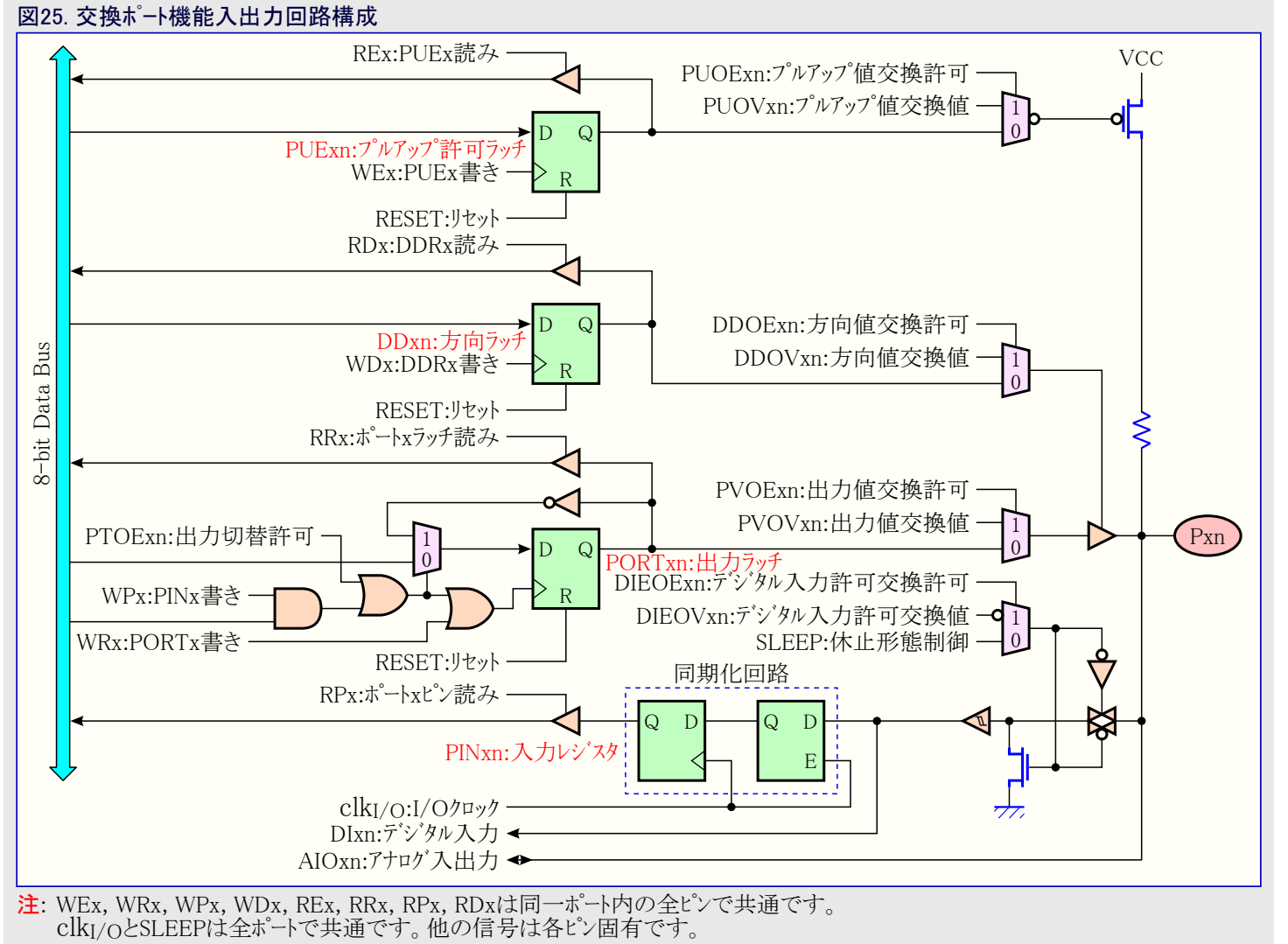
### アセンブリ言語プログラム例

```
~
LDI    R16, (1<PUEB2)           ;
LDI    R17, (1<<PB0)           ;PB2プルアップ指定値取得
LDI    R18, (1<<DDB1) | (1<<DDB0) ;PB1=0,PB0=1出力ビット値取得
OUT    PUEB, R16               ;PB1,PB0出力指定値取得
OUT    PORTB, R17              ;PB2プルアップ設定
OUT    DDRB, R18               ;PB1=0,PB0=1出力ビット値設定
NOP                                     ;入出力方向設定
IN     R16, PINB                ;同期化遅延対処
~                                     ;ピン値読み戻し
;
```

注: 5ページの「コード例」をご覧ください。

### 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。下の図25は単純化された40頁の図22.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVRマイクロコントローラシステムの全ポートピンに適用できる一般的な記述として取り扱います。



上の図の図解は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。いくつかの重複信号は全てのポートピンに存在しないかもしれませんが。



表20.は重複(交換)信号の機能一覧を示します。図25.で示すピンとポートは次表で示されません。重複(交換)信号は交換機能を持つ機能部で内部的に生成されます。

表20. 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、PUExn=1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、PUExnレジスタビット設定に関係なく、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なら、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、休止形態動作)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、休止形態動作)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。本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。

次節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の交換機能は表21.で示されます。

表21. ポートAの交換機能

ピン	機能	交換機能の説明
PA0	ADC0	A/D変換器(ADC)用入力チャネル
	PCINT0	ピン変化割り込み元
PA1	ADC1	A/D変換器(ADC)用入力チャネル
	AIN0	アナログ比較器の正入力(注1)
PA2	PCINT1	ピン変化割り込み元
	ADC2	A/D変換器(ADC)用入力チャネル
	AIN1	アナログ比較器の負入力(注1)
PA3	PCINT2	ピン変化割り込み元
	ADC3	A/D変換器(ADC)用入力チャネル
PA4	PCINT3	ピン変化割り込み元
	ADC4	A/D変換器(ADC)用入力チャネル
PA5	PCINT4	ピン変化割り込み元
	ADC5	A/D変換器(ADC)用入力チャネル
PA6	PCINT5	ピン変化割り込み元
	ADC6	A/D変換器(ADC)用入力チャネル
PA7	PCINT6	ピン変化割り込み元
	ADC7	A/D変換器(ADC)用入力チャネル
	PCINT7	ピン変化割り込み元

注1: アナログ比較器の機能を妨げるデジタルポート機能为了避免するため、ポートピンを内部プルアップをOFFにした入力として構成設定してください。

下の表22はポートの交換機能によって使われる再定義信号を要約します。信号どう使われるかの図解については43頁の図25をご覧ください。

表22. ポートAの再定義信号

ピン	信号	構成	ピン	信号	構成
PA0	PUOE	0	PA1	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT0・PCIE0)+ADC0D		DIEOE	(PCINT1・PCIE0)+ADC1D
	DIEOV	PCINT0・PCIE0		DIEOV	PCINT1・PCIE0
	DI	PCINT0入力		DI	PCINT1入力
AIO	ADC0入力	AIO	ADC1入力		
PA2	PUOE	0	PA3	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT2・PCIE0)+ADC2D		DIEOE	(PCINT3・PCIE0)+ADC3D
	DIEOV	PCINT2・PCIE0		DIEOV	PCINT3・PCIE0
	DI	PCINT2入力		DI	PCINT3入力
AIO	ADC2入力	AIO	ADC3入力		
PA4	PUOE	0	PA5	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT4・PCIE0)+ADC4D		DIEOE	(PCINT5・PCIE0)+ADC5D
	DIEOV	PCINT4・PCIE0		DIEOV	PCINT5・PCIE0
	DI	PCINT4入力		DI	PCINT5入力
AIO	ADC4入力	AIO	ADC5入力		
PA6	PUOE	0	PA7	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT6・PCIE0)+ADC6D		DIEOE	(PCINT7・PCIE0)+ADC7D
	DIEOV	PCINT6・PCIE0		DIEOV	PCINT7・PCIE0
	DI	PCINT6入力		DI	PCINT7入力
AIO	ADC6入力	AIO	ADC7入力		

### 10.3.2. ポートBの交換機能

ポートBの交換機能は表23.で示されます。

表23. ポートBの交換機能

ピン	機能	交換機能の説明
PB0	ADC8	A/D変換器(ADC)用入力チャンネル
	PCINT8	ピン変化割り込み元
PB1	ADC9	A/D変換器(ADC)用入力チャンネル
	PCINT9	ピン変化割り込み元
PB2	ADC10	A/D変換器(ADC)用入力チャンネル
	PCINT10	ピン変化割り込み元
PB3	ADC11	A/D変換器(ADC)用入力チャンネル
	PCINT11	ピン変化割り込み元
PB4	ADC12	A/D変換器(ADC)用入力チャンネル
	PCINT12	ピン変化割り込み元
PB5	ADC13	A/D変換器(ADC)用入力チャンネル
	PCINT13	ピン変化割り込み元
PB6	ADC14	A/D変換器(ADC)用入力チャンネル
	PCINT14	ピン変化割り込み元
PB7	ADC15	A/D変換器(ADC)用入力チャンネル
	PCINT15	ピン変化割り込み元

下の表24はポートの交換機能によって使われる再定義信号を要約します。信号どう使われるかの図解については43頁の図25をご覧ください。

表24. ポートBの再定義信号

ピン	信号	構成	ピン	信号	構成
PB0	PUOE	0	PB1	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT8・PCIE1)+ADC8D		DIEOE	(PCINT9・PCIE1)+ADC9D
	DIEOV	PCINT8・PCIE1		DIEOV	PCINT9・PCIE1
	DI	PCINT8入力		DI	PCINT9入力
AIO	ADC8入力	AIO	ADC9入力		
PB2	PUOE	0	PB3	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT10・PCIE1)+ADC10D		DIEOE	(PCINT11・PCIE1)+ADC11D
	DIEOV	PCINT10・PCIE1		DIEOV	PCINT11・PCIE1
	DI	PCINT10入力		DI	PCINT11入力
AIO	ADC10入力	AIO	ADC11入力		
PB4	PUOE	0	PB5	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT12・PCIE1)+ADC12D		DIEOE	(PCINT13・PCIE1)+ADC13D
	DIEOV	PCINT12・PCIE1		DIEOV	PCINT13・PCIE1
	DI	PCINT12入力		DI	PCINT13入力
AIO	ADC12入力	AIO	ADC13入力		
PB6	PUOE	0	PB7	PUOE	0
	PUOV	0		PUOV	0
	DDOE	0		DDOE	0
	DDOV	0		DDOV	0
	PVOE	0		PVOE	0
	PVOV	0		PVOV	0
	PTOE	0		PTOE	0
	DIEOE	(PCINT14・PCIE1)+ADC14D		DIEOE	(PCINT15・PCIE1)+ADC15D
	DIEOV	PCINT14・PCIE1		DIEOV	PCINT15・PCIE1
	DI	PCINT14入力		DI	PCINT15入力
AIO	ADC14入力	AIO	ADC15入力		

### 10.3.3. ポートCの交換機能

ポートCの交換機能は表25.で示されます。

表25. ポートCの交換機能

ピン	機能	交換機能の説明
PC0	TOCC0	タイマ/カウンタ比較出力チャンネル0 (注1)
	SS	SPI従装置選択入力 (注2)
	XCK	USART転送クロック
	ADC16	A/D変換器(ADC)用入力チャンネル
	PCINT16	ピン変化割り込み元
PC1	TOCC1	タイマ/カウンタ比較出力チャンネル1 (注1)
	INT0	外部割り込み要求0
	CLKO	システムクロック出力 (注3)
	ADC17	A/D変換器(ADC)用入力チャンネル
	PCINT17	ピン変化割り込み元
PC2	TOCC2	タイマ/カウンタ比較出力チャンネル2 (注1)
	INT1	外部割り込み要求1
	RXD	USART直列データ入力
	ADC18	A/D変換器(ADC)用入力チャンネル
	PCINT18	ピン変化割り込み元
PC3	TOCC3	タイマ/カウンタ比較出力チャンネル3 (注1)
	TXD	USART直列データ出力
	ADC19	A/D変換器(ADC)用入力チャンネル
	PCINT19	ピン変化割り込み元
PC4	TOCC4	タイマ/カウンタ比較出力チャンネル4 (注1)
	ADC20	A/D変換器(ADC)用入力チャンネル
	PCINT20	ピン変化割り込み元
PC5	TOCC5	タイマ/カウンタ比較出力チャンネル5 (注1)
	ICP1	捕獲起動入力ピン
	T0	タイマ/カウンタ0 クロック元
	ADC21	A/D変換器(ADC)用入力チャンネル
PC6	PCINT21	ピン変化割り込み元
	TOCC6	タイマ/カウンタ比較出力チャンネル6 (注1)
	CLKI	外部供給元用クロック入力
	ADC22	A/D変換器(ADC)用入力チャンネル
PC7	PCINT22	ピン変化割り込み元
	TOCC7	タイマ/カウンタ比較出力チャンネル7 (注1)
	T1	タイマ/カウンタ1 クロック元
	ADC23	A/D変換器(ADC)用入力チャンネル
	PCINT23	ピン変化割り込み元

注1: 85頁の「TOCPMSA1, TOCPMSA0 – タイマ/カウンタ比較出力ピン多重器選択レジスタ」をご覧ください。

注2: SPIが従装置として許可されると、このピンはピンのデータ方向ビットに拘らず、自動的に入力として構成設定されます。SPIが主装置として許可されると、データ方向の標準ピン制御が再開されます。

注3: CKOUTヒューズがプログラム(0)されると、ピン設定に拘らず、このピンにシステムクロックが出力されます。このクロックはデバイスがリセットの時にも出力されます。



下の表26.はポートの交換機能によって使われる再定義信号を要約します。信号どう使われるかの図解については43頁の図25.をご覧ください。

表26. ポートCの再定義信号

ピン	信号	構成
PC0	PUOE	0
	PUOV	0
	DDOE	SPE・MSTR
	DDOV	0
	PVOE	TOCC0OE+XCK_主(注1)
	PVOV	XCK_主装置(注1)・XCK出力+XCK_主装置(注1)・TOCC0出力
	PTOE	0
	DIEOE	(PCINT16・PCIE2)+ADC16D+(XCK_従装置(注2)・RXEN・SFDE)
	DIEOV	(PCINT16・PCIE2)+(XCK_従装置(注2)・RXEN・SFDE)
	DI	PCINT16/XCK/SS入力
AIO	ADC16入力	
PC1	PUOE	0
	PUOV	0
	DDOE	CKOUT(注3)
	DDOV	CKOUT(注3)
	PVOE	TOCC1OE+CKOUT(注3)
	PVOV	CKOUT(注3)・SYSTEM_CLOCK+CKOUT・TOCC1出力
	PTOE	0
	DIEOE	(PCINT17・PCIE2)+ADC17D+INT0
	DIEOV	PCINT17・PCIE2+INT0
	DI	PCINT17/INT0入力
AIO	ADC17入力	
PC2	PUOE	0
	PUOV	0
	DDOE	RXEN
	DDOV	0
	PVOE	TOCC2OE
	PVOV	TOCC2出力
	PTOE	0
	DIEOE	(PCINT18・PCIE2)+ADC18D+(RXEN・SFDE)+INT1
	DIEOV	PCINT18・PCIE2+(RXEN・SFDE)+INT1
	DI	PCINT18/INT1/RXD入力
AIO	ADC18入力	
PC3	PUOE	TXEN
	PUOV	0
	DDOE	TXEN
	DDOV	0
	PVOE	TOCC3OE+TXEN
	PVOV	TXEN・TXD出力+TXEN・TOCC3出力
	PTOE	0
	DIEOE	(PCINT19・PCIE2)+ADC19D
	DIEOV	PCINT19・PCIE2
	DI	PCINT19入力
AIO	ADC19入力	

注1: XCK\_主装置=UMSEL1・UMSEL0+UMSEL0・DDRC0

注2: XCK\_従装置=UMSEL1・UMSEL0・DDRC0

注3: CKOUTはこのヒューズビットがプログラム(0)される時に1です。

表26. (続き) ホートCの再定義信号

ピン	信号	構成
	PUOE	0
	PUOV	0
	DDOE	0
	DDOV	0
	PVOE	TOCC4OE
PC4	PVOV	TOCC4出力
	PTOE	0
	DIEOE	(PCINT20・PCIE2)+ADC20D
	DIEOV	PCINT20・PCIE2
	DI	PCINT20入力
	AIO	ADC20入力
	PUOE	0
	PUOV	0
	DDOE	0
	DDOV	0
	PVOE	TOCC5OE
PC5	PVOV	TOCC5出力
	PTOE	0
	DIEOE	(PCINT21・PCIE2)+ADC21D
	DIEOV	PCINT21・PCIE2
	DI	PCINT21/T0/ICP1入力
	AIO	ADC21入力
	PUOE	0
	PUOV	0
	DDOE	EXT_CLOCK(注4)
	DDOV	0
	PVOE	TOCC6OE+EXT_CLOCK(注4)
PC6	PVOV	TOCC6出力・EXT_CLOCK(注4)
	PTOE	0
	DIEOE	(PCINT22・PCIE2)+ADC22D+EXT_CLOCK(注4)
	DIEOV	(EXT_CLOCK(注4)・PCINT22・PCIE2)+(EXT_CLOCK(注4)・PWR_DOWN)
	DI	PCINT22/クロック入力
	AIO	ADC22入力
	PUOE	0
	PUOV	0
	DDOE	0
	DDOV	0
	PVOE	TOCC7OE
PC7	PVOV	TOCC7出力
	PTOE	0
	DIEOE	(PCINT23・PCIE2)+ADC23D
	DIEOV	PCINT23・PCIE2
	DI	PCINT23/T1入力
	AIO	ADC23入力

注4: EXT\_CLOCKは(CKSELピンス)によって外部クロックが選択されたことを意味します。

### 10.3.4. ポートDの交換機能

ポートDの交換機能は表27.で示されます。

表27. ポートDの交換機能

ピン	機能	交換機能の説明
PD0	MOSI	SPIの主装置出力/従装置入力 (注1)
	SDA	2線インターフェース(TWI)データ (注2)
	ADC24	A/D変換器(ADC)用入力チャンネル
	PCINT24	ピン変化割り込み元
PD1	MISO	SPIの主装置入力/従装置出力 (注3)
	ADC25	A/D変換器(ADC)用入力チャンネル
	PCINT25	ピン変化割り込み元
PD2	RESET	Low活性、外部リセット入力 (注4)
	dW	デバッグWIREの入出力
	ADC26	A/D変換器(ADC)用入力チャンネル
	PCINT26	ピン変化割り込み元
PD3	SCK	SPIの主装置クロック出力/従装置クロック入力 (注1)
	SCL	2線インターフェース(TWI)クロック (注5)
	ADC27	A/D変換器(ADC)用入力チャンネル
	PCINT27	ピン変化割り込み元

**注1:** SPIが従装置として許可されると、このピンはピンのデータ方向ビットに拘らず、自動的に入力として構成設定されます。SPIが主装置として許可されると、データ方向の標準ピン制御が再開されます。

**注2:** TWI従装置制御レジスタA(TWSCRA)のTWENが設定(1)されると、このピンはポートから切断されて、TWI用の直列データになります。この動作形態では、ピンがスルーレート制限と尖頭雑音濾波器を持つオープンドレイン回路によって駆動されます。

**注3:** SPIが主装置として許可されると、このピンはピンのデータ方向ビットに拘らず、自動的に入力として構成設定されます。SPIが従装置として許可されると、データ方向の標準ピン制御が再開されます。

**注4:** RSTDISBLヒューズを非プログラム(1)にすることによって許可されます。リセットピンとして使用时、ピンのプルアップ抵抗が活性にされ、出力駆動部とデジタル入力は不活性にされます。

**注5:** TWI従装置制御レジスタA(TWSCRA)のTWENが設定(1)されると、このピンはポートから切断されて、TWI用の直列クロックになります。この動作形態では、ピンがスルーレート制限と尖頭雑音濾波器を持つオープンドレイン回路によって駆動されます。

下の表28はポートの交換機能によって使われる再定義信号を要約します。信号どう使われるかの図解については43頁の図25をご覧ください。

表28. ポートDの再定義信号

ピン	信号	構成	ピン	信号	構成
PD0	PUOE	0	PD1	PUOE	0
	PUOV	0		PUOV	0
	DDOE	$\overline{\text{TWEN}} + (\text{SPE} \cdot \overline{\text{MSTR}})$		DDOE	$\text{SPE} \cdot \overline{\text{MSTR}}$
	DDOV	$\overline{\text{TWEN}} \cdot \overline{\text{SDA}}$ 出力		DDOV	0
	PVOE	$\overline{\text{TWEN}} + (\text{SPE} \cdot \overline{\text{MSTR}})$		PVOE	$\text{SPE} \cdot \overline{\text{MSTR}}$
	PVOV	$\overline{\text{TWEN}} \cdot \text{SPE} \cdot \overline{\text{MSTR}} \cdot \text{SPI}$ 主装置出力		PVOV	$\text{SPI}$ 従装置出力
	PTOE	0		PTOE	0
	DIEOE	$(\text{PCINT24} \cdot \text{PCIE3}) + \text{ADC24D}$		DIEOE	$(\text{PCINT25} \cdot \text{PCIE3}) + \text{ADC25D}$
	DIEOV	$\text{PCINT24} \cdot \text{PCIE3}$		DIEOV	$\text{PCINT25} \cdot \text{PCIE3}$
	DI	$\text{PCINT24}/\text{SPI}$ 従装置入力		DI	$\text{PCINT25}/\text{SPI}$ 主装置入力
AIO	$\text{ADC24}/\text{SDA}$ 入力	AIO	$\text{ADC25}$ 入力		
PD2	PUOE	$\overline{\text{RSTDISBL}}$ (注1)	PD3	PUOE	0
	PUOV	1		PUOV	0
	DDOE	$\overline{\text{RSTDISBL}}$ (注1)		DDOE	$\overline{\text{TWEN}} + (\text{SPE} \cdot \overline{\text{MSTR}})$
	DDOV	0		DDOV	$\overline{\text{TWEN}} \cdot \overline{\text{SCL}}$ 出力
	PVOE	$\overline{\text{RSTDISBL}}$ (注1)		PVOE	$\overline{\text{TWEN}} + (\text{SPE} \cdot \overline{\text{MSTR}})$
	PVOV	0		PVOV	$\overline{\text{TWEN}} \cdot \text{SPE} \cdot \overline{\text{MSTR}} \cdot \text{SCK}$ 出力
	PTOE	0		PTOE	0
	DIEOE	$(\text{PCINT26} \cdot \text{PCIE3}) + \text{ADC26D} + \overline{\text{RSTDISBL}}$ (注1)		DIEOE	$(\text{PCINT27} \cdot \text{PCIE3}) + \text{ADC27D}$
	DIEOV	$\text{PCINT26} \cdot \text{PCIE3} \cdot \overline{\text{RSTDISBL}}$ (注1)		DIEOV	$\text{PCINT27} \cdot \text{PCIE3}$
DI	$\text{PCINT26}$ 入力	DI	$\text{PCINT27}/\text{SCK}$ 入力		
AIO	$\text{ADC26}/\overline{\text{RESET}}$ 入力	AIO	$\text{ADC27}/\text{SCL}$ 入力		

注1: RSTDISBLはこのヒューズビットがプログラム(0)される時に1です。

## 10.4. I/Oポート用レジスタ

### 10.4.1. PHDE – ポート高駆動許可レジスタ (Port High Drive Enable Register)

ビット	7	6	5	4	3	2	1	0	
\$14 (\$34)	–	–	–	–	–	PHDEC	–	–	PHDE
Read/Write	R	R	R	R	R	R/W	R	R	
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を読みます。

- ビット2 – PHDEC : ポートC高駆動許可 (Port C High Drive Enable)

このビットが設定(1)されると、ポートCの追加高吸い込み能力が許可されます。

- ビット1,0 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

### 10.4.2. PUEd – ポートDプルアップ許可制御レジスタ (Port D Pull-up Enable Control Register)

ビット	7	6	5	4	3	2	1	0	
\$0F (\$2F)	–	–	–	–	PUEd3	PUEd2	PUEd1	PUEd0	PUEd
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を読みます。

- ビット3~0 – PUEd3~0 : ポートDプルアップ許可ビット (Pull-Up Enable Bits)

プルアップ許可ビット(PUEdn)が設定(1)されると、対応するポートピン(PDn)のプルアップ抵抗が許可されます。

### 10.4.3. 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	
\$0E (\$2E)	–	–	–	–	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
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を読みます。

- ビット3~0 – PORTD3~0 : ポートD出力データビット (Port Data Bits)

PDnピンが出力として構成設定されると、PORTDn設定(1)はPDnをHighに駆動します。PORTDn解除(0)はPDnをLowに駆動します。ピンが入力として構成設定されると、PORTDnビットの値は問題になりません。41頁の表19をご覧ください。

### 10.4.4. 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	
\$0D (\$2D)	–	–	–	–	DDD3	DDD2	DDD1	DDD0	DDRD
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を読みます。

- ビット3~0 – DDD3~0 : ポートD出力方向ビット (Data Direction Bits)

DDDnが設定(1)されると、PDnピンは出力として構成設定されます。DDDnが解除(0)されると、ピンは入力として構成設定されます。



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

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

- ビット7~4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

- ビット3~0 – PIND3~0 : ポートD入力データ (Port Input Data)

データ方向ビットの設定に拘らず、PDnポートピンの値はPINDnビットを通して読むことができます。

PINDnへの論理1書き込みはDDDnの値に拘らず、PORTDnの値を(交互に)切り替えます。

#### 10.4.6. PUEC – ポートCプルアップ許可制御レジスタ (Port C Pull-up Enable Control Register)

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	PUEC7	PUEC6	PUEC5	PUEC4	PUEC3	PUEC2	PUEC1	PUEC0	PUEC
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 – PUEC7~0 : ポートCプルアップ許可ビット (Pull-Up Enable Bits)

プルアップ許可ビット(PUECn)が設定(1)されると、対応するポートピン(PCn)のプルアップ抵抗が許可されます。

#### 10.4.7. 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	
\$0A (\$2A)	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	

- ビット7~0 – PORTC7~0 : ポートC出力データビット (Port Data Bits)

PCnピンが出力として構成設定されると、PORTCn設定(1)はPCnをHighに駆動します。PORTCn解除(0)はPCnをLowに駆動します。ピンが入力として構成設定されると、PORTCnビットの値は問題になりません。41頁の表19をご覧ください。

#### 10.4.8. 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	
\$09 (\$29)	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	

- ビット7~0 – DDC7~0 : ポートC出力方向ビット (Data Direction Bits)

DDCnが設定(1)されると、PCnピンは出力として構成設定されます。DDCnが解除(0)されると、ピンは入力として構成設定されます。

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

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	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	

- ビット7~0 – PINC7~0 : ポートC入力データ (Port Input Data)

データ方向ビットの設定に拘らず、PCnポートピンの値はPINCnビットを通して読むことができます。

PINCnへの論理1書き込みはDDCnの値に拘らず、PORTCnの値を(交互に)切り替えます。

#### 10.4.10. PUEB – ポートBプルアップ許可制御レジスタ (Port B Pull-up Enable Control Register)

ビット	7	6	5	4	3	2	1	0	
\$07 (\$27)	PUEB7	PUEB6	PUEB5	PUEB4	PUEB3	PUEB2	PUEB1	PUEB0	PUEB
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 – PUEB7~0 : ポートBプルアップ許可ビット (Pull-Up Enable Bits)

プルアップ許可ビット(PUEBn)が設定(1)されると、対応するポートピン(PBn)のプルアップ抵抗が許可されます。

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

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	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	

##### ● ビット7~0 – PORTB7~0 : ホートB出力データビット (Port Data Bits)

PBnピンが出力として構成設定されると、PORTBn設定(1)はPBnをHighに駆動します。PORTBn解除(0)はPBnをLowに駆動します。ピンが入力として構成設定されると、PORTBnビットの値は問題になりません。41頁の表19をご覧ください。

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

ビット	7	6	5	4	3	2	1	0	
\$05 (\$25)	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	

##### ● ビット7~0 – DDB7~0 : ホートB出力方向ビット (Data Direction Bits)

DDBnが設定(1)されると、PBnピンは出力として構成設定されます。DDBnが解除(0)されると、ピンは入力として構成設定されます。

#### 10.4.13. PINB – ホートB入力レジスタ (Port B Input Pins)

ビット	7	6	5	4	3	2	1	0	
\$04 (\$24)	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	

##### ● ビット7~0 – PINB7~0 : ホートB入力データ (Port Input Data)

データ方向ビットの設定に拘らず、PBnポートピンの値はPINBnビットを通して読むことができます。

PINBnへの論理1書き込みはDDBnの値に拘らず、PORTBnの値を(交互に)切り替えます。

#### 10.4.14. PUEA – ホートAプルアップ許可制御レジスタ (Port A Pull-up Enable Control Register)

ビット	7	6	5	4	3	2	1	0	
\$03 (\$23)	PUEA7	PUEA6	PUEA5	PUEA4	PUEA3	PUEA2	PUEA1	PUEA0	PUEA
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 – PUEA7~0 : ホートAプルアップ許可ビット (Pull-Up Enable Bits)

プルアップ許可ビット(PUEAn)が設定(1)されると、対応するポートピン(PAn)のプルアップ抵抗が許可されます。

#### 10.4.15. PORTA – ホートA出力レジスタ (Port A Data Register)

ビット	7	6	5	4	3	2	1	0	
\$02 (\$22)	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
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 – PORTA7~0 : ホートA出力データビット (Port Data Bits)

PAnピンが出力として構成設定されると、PORTAn設定(1)はPAnをHighに駆動します。PORTAn解除(0)はPAnをLowに駆動します。ピンが入力として構成設定されると、PORTAnビットの値は問題になりません。41頁の表19をご覧ください。

#### 10.4.16. DDRA – ホートA方向レジスタ (Port A Data Direction Register)

ビット	7	6	5	4	3	2	1	0	
\$01 (\$21)	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
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 – DDA7~0 : ホートA出力方向ビット (Data Direction Bits)

DDAnが設定(1)されると、PAnピンは出力として構成設定されます。DDAnが解除(0)されると、ピンは入力として構成設定されます。

### 10.4.17. PINA – ポートA入力レジスタ (Port A Input Pins)

ビット	7	6	5	4	3	2	1	0	
\$00 (\$20)	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

- ビット7~0 – PINA7~0 : ポートA入力データ (Port Input Data)

データ方向ビットの設定に拘らず、PAnポートピンの値はPINAnビットを通して読むことができます。

PINAnへの論理1書き込みはDDAnの値に拘らず、PORTAnの値を(交互に)切り替えます。

## 11. PWM付き8ビット タイマ/カウンタ

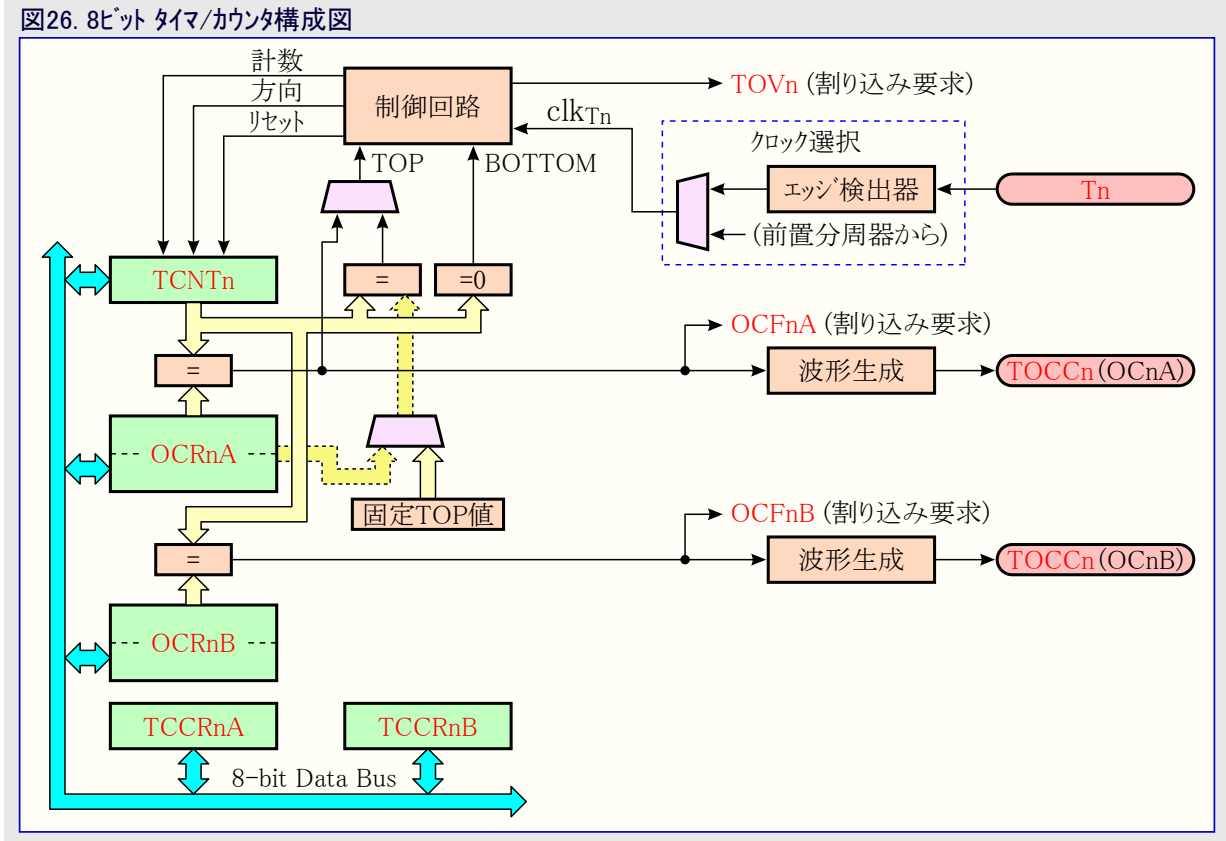
### 11.1. 特徴

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

### 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とPWM支援付きの汎用8ビット タイマ/カウンタ部です。それは正確なプログラム実行タイミング(事象管理)、波形生成を許します。

この8ビット タイマ/カウンタの単純化した構成図は図26で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(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65頁の「8ビット タイマ/カウンタ用レジスタ」で一覧されます。



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。また、85頁の「TOCPMSA1, TOCPMSA0 – タイマ/カウンタ比較出力ピン多重器選択レジスタ」と86頁の「TOCPMCOE – タイマ/カウンタ比較出力ピン多重器チャネル出力許可レジスタ」もご覧ください。

#### 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は本図で示されません。

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

2重緩衝化した比較レジスタ(OCR0AとOCR0B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC0AとOC0B)ピンでPWMまたは可変周波数出力を作成するための波形生成器によって使えます。詳細については59頁の「比較出力部」をご覧ください。この比較一致発生は比較一致割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF0AとOCF0B)も設定(1)します。

## 11.2.2. 定義

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

表29.の定義は本資料を通して広範囲に渡って使われます。

表29. 用語定義

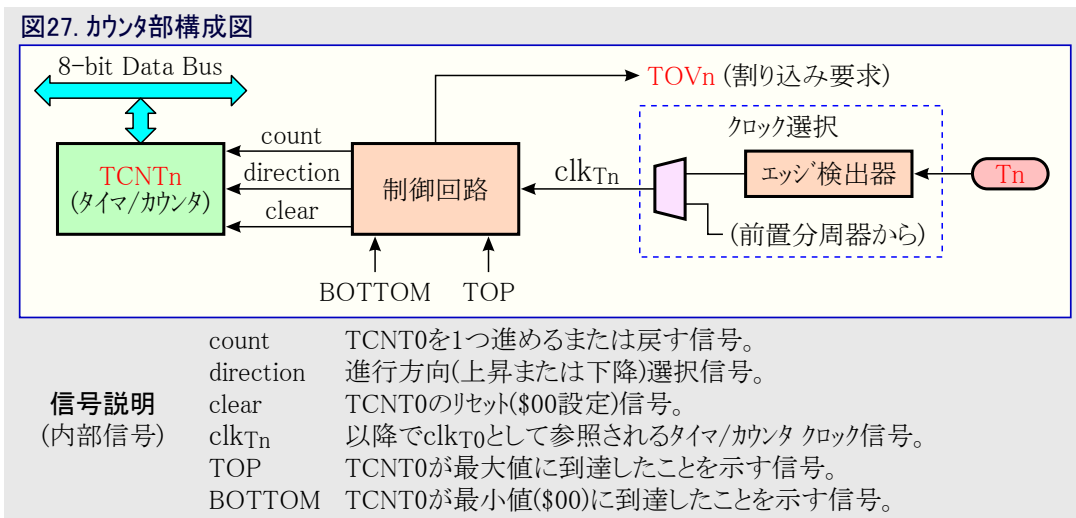
用語	意味
BOTTOM	タイマ/カウンタが\$00に到達した時。
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0A値に到達した時。この指定(TOP)値は動作種別に依存します。

## 11.3. クロック元

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR0B)に配置されたクロック選択(CS02~0)ビットによって制御されるクロック選択論理回路で選択されます。クロック元と前置分周器の詳細については89頁の「タイマ/カウンタ前置分周器」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27.は、このカウンタとその周辺環境の構成図を示します。



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

計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCR0A)に配置された波形生成種別(WGM01,0)ビットとタイマ/カウンタ制御レジスタB(TCCR0B)に配置された波形生成種別(WGM02)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOCR0A/OCR0B比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成については61頁の「動作種別」をご覧ください。

タイマ/カウンタ溢れ(TOV0)フラグはWGM02~0ビットによって選択さ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. 比較出力部

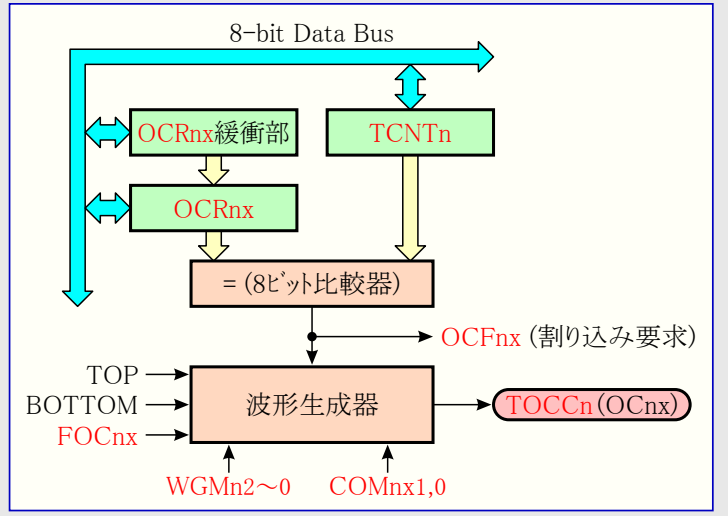
この8ビット比較器はTCNT0と比較レジスタ(OCR0AとOCR0B)を継続的に比較します。TCNT0がOCR0AまたはOCR0Bと等しければ、比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF0AまたはOCF0B)を設定(1)します。対応する割り込みが許可(I=1, OCIE0AまたはOCIE0B=1)されているならば、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットによって設定された動作種別に従った出力を生成するのにこの一致信号を使います。MAXとBOTTOM信号は動作種別(61頁の「動作種別」参照)のいくつかで両端値の特別な場合を扱うため、波形生成器によって使われます。

図28. は比較出力部の構成図を示します。

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

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

図28. 比較出力部構成図



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

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

### 11.5.2. TCNT0書き込みによる比較一致妨害

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

### 11.5.3. 比較一致部の使用

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

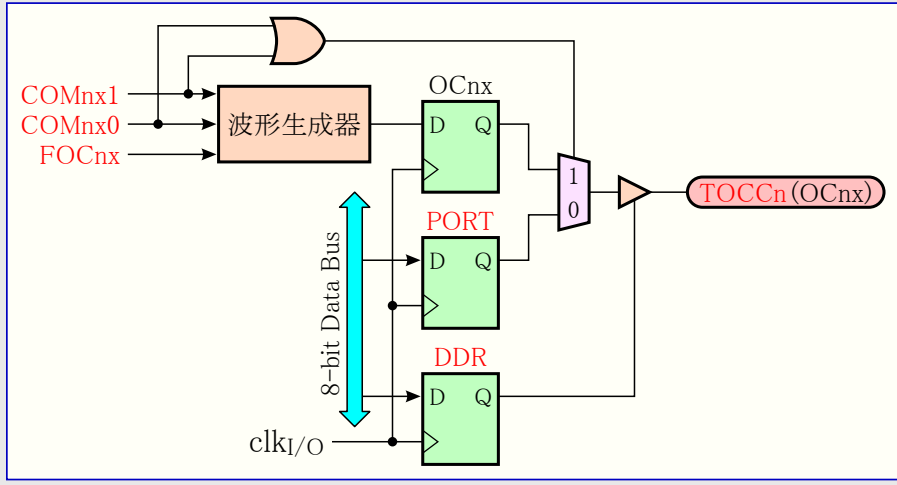
OC0xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC0x値を設定する一番簡単な方法は標準動作で強制変更(FOC0x)スロープビットを使うことです。波形生成動作種別間を変更する時でも、OC0x(内部)レジスタはその値を保ちます。

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

## 11.6. 比較一致出力部

比較出力選択(COM0x1,0)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力(OC0x)状態の定義にCOM0x1,0ビットを使います。またCOM0x1,0ビットはOC0xピン出力元を制御します。図29はCOM0x1,0ビット設定によって影響を及ぼされる論理回路の簡単化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM0x1,0ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。OC0xの状態を参照するとき、その参照はOC0xピンでなく内部OC0xレジスタに対してです。システムリセットが起こると、OC0xレジスタは'0'にリセットされます。

図29. 比較一致出力回路図



COM0x1,0ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力(OC0x)によって無効にされます。けれどもOC0xピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。OC0xピンに対するポート方向レジスタのビット(DDR\_OC0x)はOC0x値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は波形生成種別と無関係です。

比較出力ピン論理回路の設計は出力が許可される前のOC0x状態の初期化を許します。いくつかのCOM0x1,0ビット設定が或る種の動作種別に対して予約されることに注意してください。65頁の「8ビットタイマ/カウンタ用レジスタ」をご覧ください。

### 11.6.1. 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COM0x1,0ビットを違うふうに使います。全ての動作種別に対してCOM0x1,0=00設定は次の比較一致で実行すべきOC0xレジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については65頁の表30と表33を参照してください。高速PWM動作については65頁の表31と表34、位相基準PWMについては65頁の表32と表35を参照してください。

COM0x1,0ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(FOC0x)スローブビットを使うことによって直ちに効果を得ることを強制できます。「TCCR0B - タイマ/カウンタ制御レジスタB」をご覧ください。

## 11.7. 動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットの組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM0x1,0ビットは生成されるPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対するCOM0x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します(60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64頁の「タイマ/カウンタのタイミング」を参照してください(訳注:原文中の図番号省略)。

### 11.7.1. 標準動作

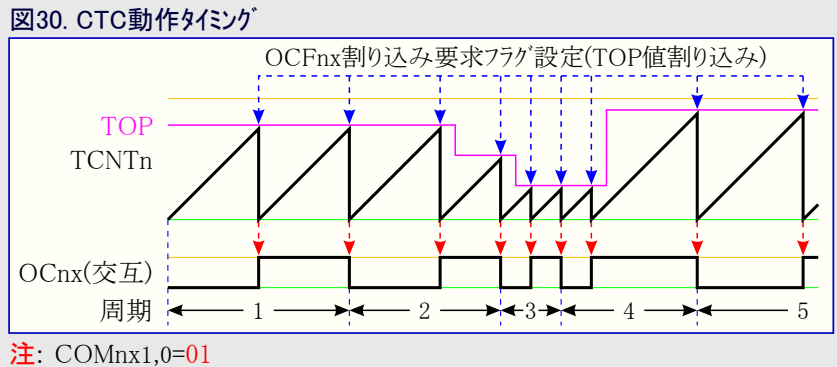
最も単純な動作種別が標準動作(WGM02~0=000)です。この動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV0)フラグはTCNT0が\$00になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV0フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのようになります。けれどもTOV0フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解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

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

### 11.7.2. 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(WGM02~0=010)ではOCR0Aがカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT0)値がOCR0Aと一致する時にカウンタが\$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30.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、そしてその後にはカウンタ(TCNT0)は解除(\$00)されます。



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

CTC動作で波形出力を生成するため、OC0A出力は比較出力選択(COM0A1,0)ビットを交互動作(=01)に設定することによって各比較一致での論理レベル交互切り替えに設定できます。OC0A値はそのピンに対するデータ方向が出力(DDR\_OC0A=1)に設定されない限りポートピンで見えないでしょう。生成された波形はOCR0Aが0(\$00)に設定される時に $f_{OC0A} = f_{clk\_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

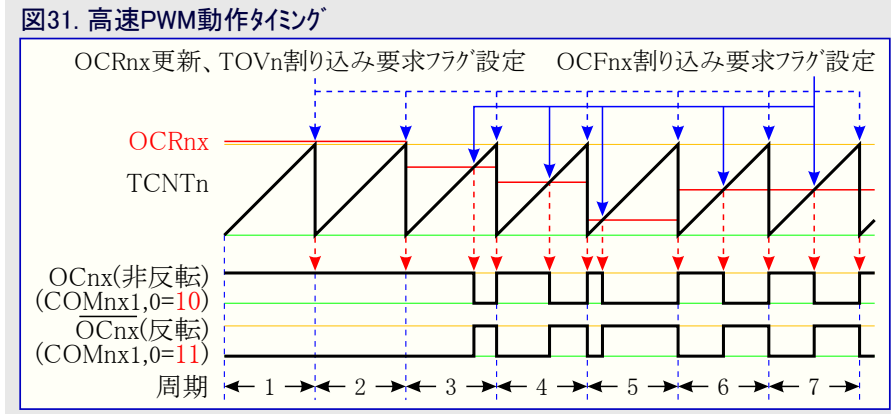
変数Nは前置分周数(1,8,64,256,1024)を表します。

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

### 11.7.3. 高速PWM動作

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

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がTOP値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は図31.で示されます。TCNT0値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。赤細線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF0x)はOCR0x=TOPを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。



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

高速PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0を'11'に設定することで生成できます。WGM02ビットが設定(1)なら、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(65頁の表31.と表34.をご覧ください)。実際のOC0x値はポートピンに対するデータ方向(DDR\_OC0x)が出力として設定される場合だけ見えるでしょう。PWM波形はTCNT0とOCR0x間の比較一致で、OC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$00、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0xレジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

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

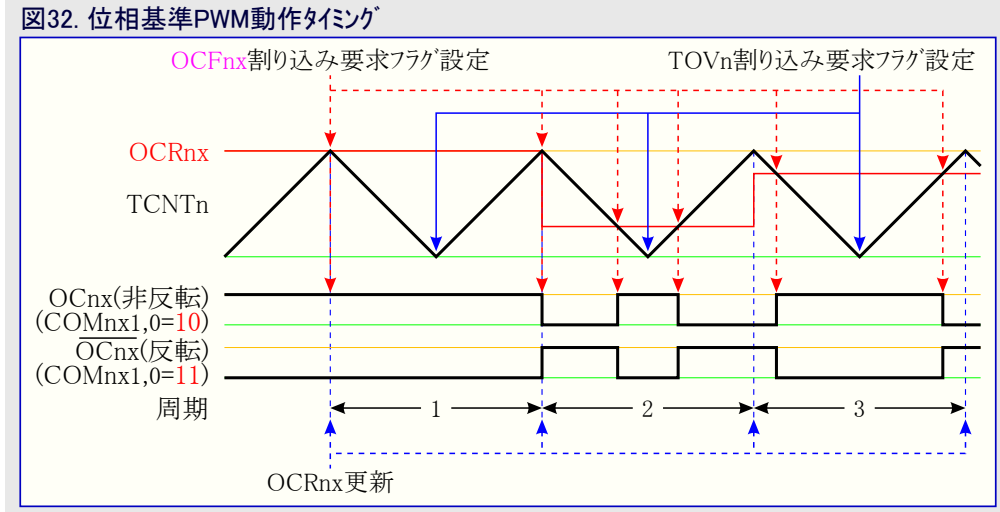
(訳補:WGM02~0=111の場合については、)高速PWM動作での(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC0A設定(COM0A1,0=01)によって達成できます。生成された波形はOCR0Aが0(\$00)に設定される時に $f_{OC0x} = f_{clk\_I/O} / 2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いて、CTC動作でのOC0A交互出力(COM0A1,0=01)と同じです。



### 11.7.4. 位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM02~0=001または101)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とします。カウンタはBOTTOMからTOPへそして次にTOPからBOTTOMへを繰り返し計数します。TOPはWGM02~0=001時に\$FF、WGM02~0=101時にOCR0Aとして定義されます。非反転比較出力動作(COM0x1,0=10)での比較出力(OC0x)は上昇計数中のTCNT0とOCR0xの比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM0x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準PWM動作でのカウンタはカウンタ値がTOPと一致するまで増加されます。カウンタはTOPに達すると計数方向を変更します。このTCNT0値は1タイマ/カウンタクロック周期間TOPと等しくなります。位相基準PWM動作のタイミング図は図32.で示されます。TCNT0値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。



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

位相基準PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0ビットを'11'に設定することで生成できます。WGM02ビットが設定(1)なら、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(65頁の表32.と表35.をご覧ください)。実際のOC0x値はそのポートピンに対するデータ方向(DDR\_OC0x)が出力として設定される場合だけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT0とOCR0x間の比較一致でOC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT0とOCR0x間の比較一致でOC0xレジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

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

図32.の第2周期のその出発点に於いて、例えば比較一致がないとしても、OCnxにはHighからLowへの遷移があります。この遷移点はBOTTOMを挟む対称を保証するためです。比較一致なしに遷移を生ずるのは2つの場合です。

- 図32.のようにOCR0xはTOPからその値を変更します。OCR0x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0x値よりも高い値から数え始め、そしてその理屈のために比較一致、それ故上昇途中で起こされるであろうOCnxの変更を逃します。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)



## 11.8. 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33は基本的なタイマ/カウンタ動作についてのタイミング データを含みます。この図は位相基準PWM動作以外の全ての動作種別でのMAX値近辺の計数の流れを示します。

図33. 前置分周なし(1/1)のタイマ/カウンタ タイミング

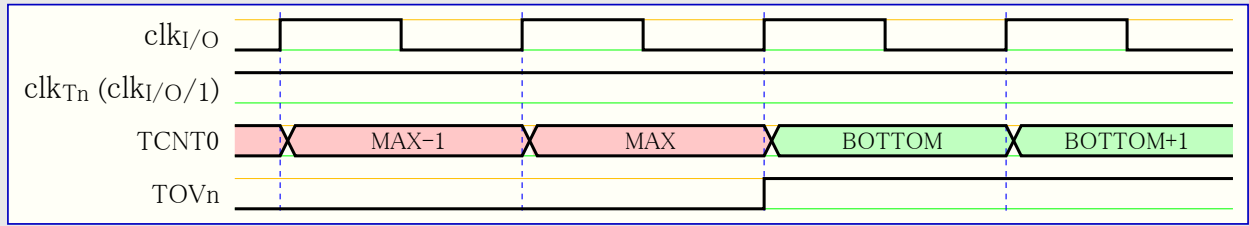


図34.は同じタイミング データを示しますが、前置分周器が許可されています。

図34. 前置分周器( $f_{clk_{I/O}}/8$ )のタイマ/カウンタ タイミング

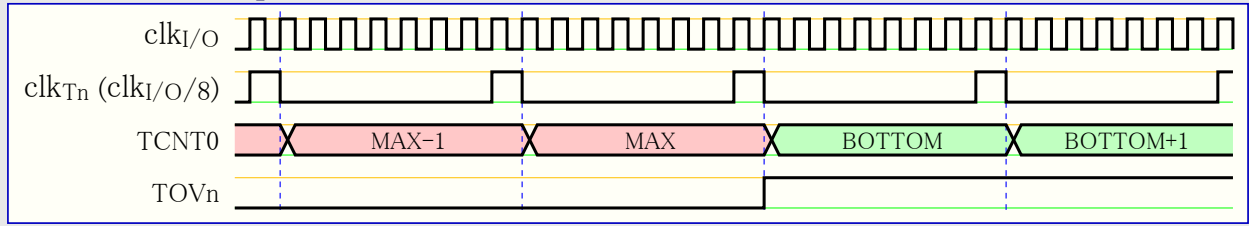


図35.はCTC動作とOCR0AがTOPのPWM動作を除く全動作種別でのOCF0Aと全動作種別でのOCF0Bの設定を示します。

図35. 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

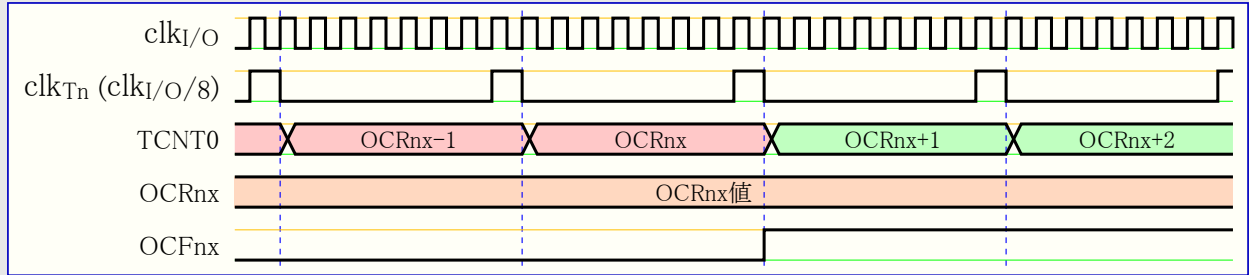
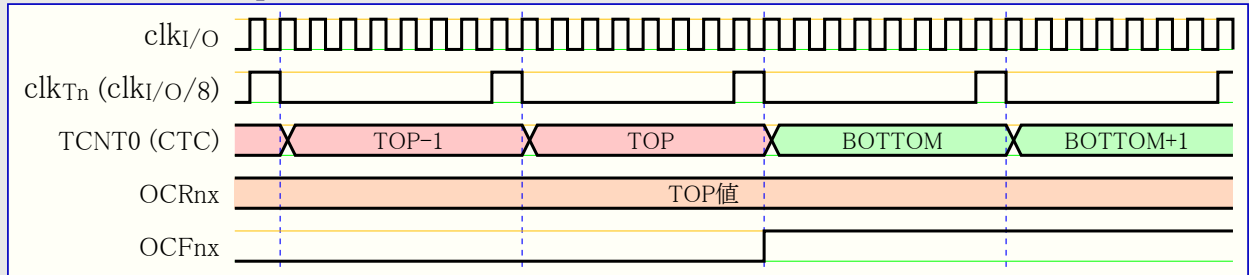


図36.はOCR0AがTOPの高速PWM動作と、CTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図36. 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



## 11.9. 8ビット タイマ/カウンタ0用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$24 (\$44)	COM0A1	COM0A0	COM0B1	COM0B0	–	–	WGM01	WGM00	TCCR0A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

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

表30.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0A1,0ビット機能を示します。

表31.はWGM02~0ビットが高速PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表32.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表30. 非PWM動作比較A出力選択

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

表31. 高速PWM動作比較A出力選択 (共通注意参照)

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

表32. 位相基準PWM動作比較A出力選択 (共通注意参照)

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

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

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

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

表33.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0B1,0ビット機能を示します。

表34.はWGM02~0ビットが高速PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表35.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表33. 非PWM動作比較B出力選択

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

表34. 高速PWM動作比較B出力選択 (共通注意参照)

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

表35. 位相基準PWM動作比較B出力選択 (共通注意参照)

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

**共通注意:** COM0x1が設定(1)され、対応するOCR0xがTOPと等しい時に、特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMまたはTOPでの設定(1)または解除(0)は行われます。より多くの詳細については62頁の「高速PWM動作」または63頁の「位相基準PWM動作」をご覧ください。(表31,32,34,35.各々での注:を纏めました。)

●ビット3,2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

●ビット1,0 – WGM01,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)

タイマ/カウンタ制御レジスタB(TCCR0B)で得られるWGM02ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元、使われるべき波形生成のどの形式かを制御します(表36参照)。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。61頁の「動作種別」をご覧ください。

表36. 波形生成種別選択

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

注: MAX=\$FF、BOTTOM=\$00です。

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

ビット	7	6	5	4	3	2	1	0	
\$25 (\$45)	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	TCCR0B
Read/Write	W	W	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7 – FOC0A : OC0A強制変更 (Force Output Compare A)

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

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

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

FOC0Aビットは常に0として読みます。

●ビット6 – FOC0B : OC0B強制変更 (Force Output Compare B)

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

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

FOC0Bスロープは何れの割り込みの生成も行いません。

FOC0Bビットは常に0として読みます。

●ビット5,4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

●ビット3 – WGM02 : 波形生成種別 (Waveform Generation Mode bit 2)

65頁の「TCCR0A – タイマ/カウンタ0制御レジスタA」のWGM01,0ビット記述をご覧ください。

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

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

表37. 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	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	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。

### 11.9.3. TCNT0 – タイマ/カウンタ0 (Timer/Counter0)

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。

### 11.9.4. OCR0A – タイマ/カウンタ0 比較Aレジスタ (Timer/Counter0 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やOC0Aピンでの波形出力を生成するのに使えます。

### 11.9.5. OCR0B – タイマ/カウンタ0 比較Bレジスタ (Timer/Counter0 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やOC0Bピンでの波形出力を生成するのに使えます。

### 11.9.6. 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 (\$6E)	7	6	5	4	3	2	1	0	
	-	-	-	-	-	OCIE0B	TOIE0	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を読みます。

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

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

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

TOIE0ビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0溢れ割り込みが許可されます。タイマ/カウンタ(TCNT0)溢れが起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFR0)でタイマ/カウンタ溢れ割り込み要求(TOV0)フラグが設定(1)されると、対応する割り込みが実行されます。

### 11.9.7. 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 \$15 (\$35)	7	6	5	4	3	2	1	0	
	-	-	-	-	-	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)**

このビットは予約されており、常に0を読みます。

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

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

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

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

このフラグの設定(1)はWGM02~0設定に依存します。66頁の表36をご覧ください。







### 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81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エッジを制御します。クロック元が選択されないとき、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。73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(130頁の「アナログ比較器」参照)のどちらかの外部(エッジで起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波器(雑音消去器)を含みます。

TOP値または最大タイマ/カウンタ値は、いくつかの動作種別で、OCR1A、ICR1、または一群の固定値のどれかによって定義できます。PWM動作でTOP値としてOCR1Aを使うと、OCR1AはPWM出力生成用に使えません。けれどもこの場合、TOP値は動作中に変更されるのをTOP値に許す2重緩衝にします。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

次の定義は本資料を通して広範囲に使われます。

表38. 用語定義

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

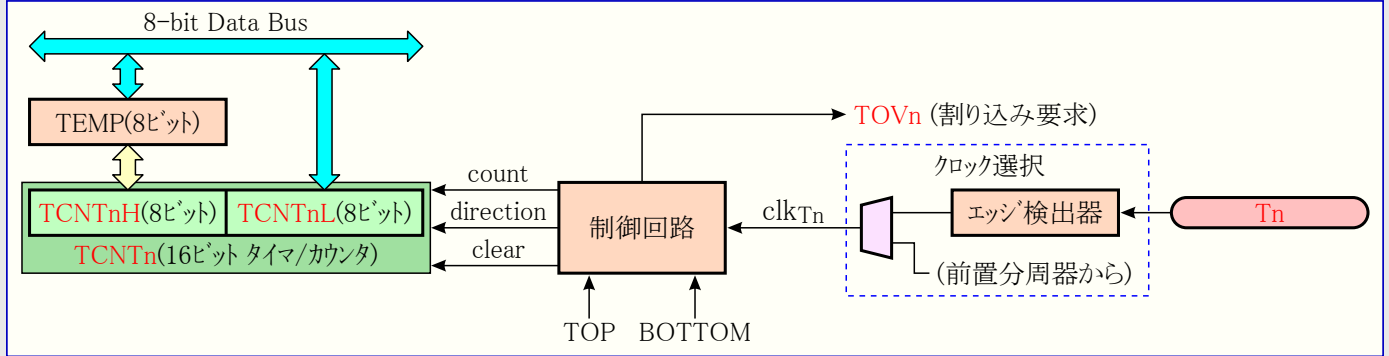
### 12.3. タイマ/カウンタのクロック元

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**クロック選択(CS12~0)**ビットによって制御されるクロック選択論理回路で選択されます。クロック元と前置分周器の詳細については89頁の「**タイマ/カウンタの前置分周器**」をご覧ください。

### 12.4. 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38はこのカウンタとその周辺の構成図を示します。

図38. カウンタ部構成図



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

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)ビットによって選択された内部または外部のクロック元から生成できます。クロック元が選択されない(CS12~0=000)時にカウンタは停止されます。けれどもTCNT1値はタイマ/カウンタ クロック(clkT1)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先権を持ちます)。

計数順序(方法)は**タイマ/カウンタ制御レジスタA(TCCR1A)**と**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**波形生成種別(WGM13~0)**ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては76頁の「**動作種別**」をご覧ください。

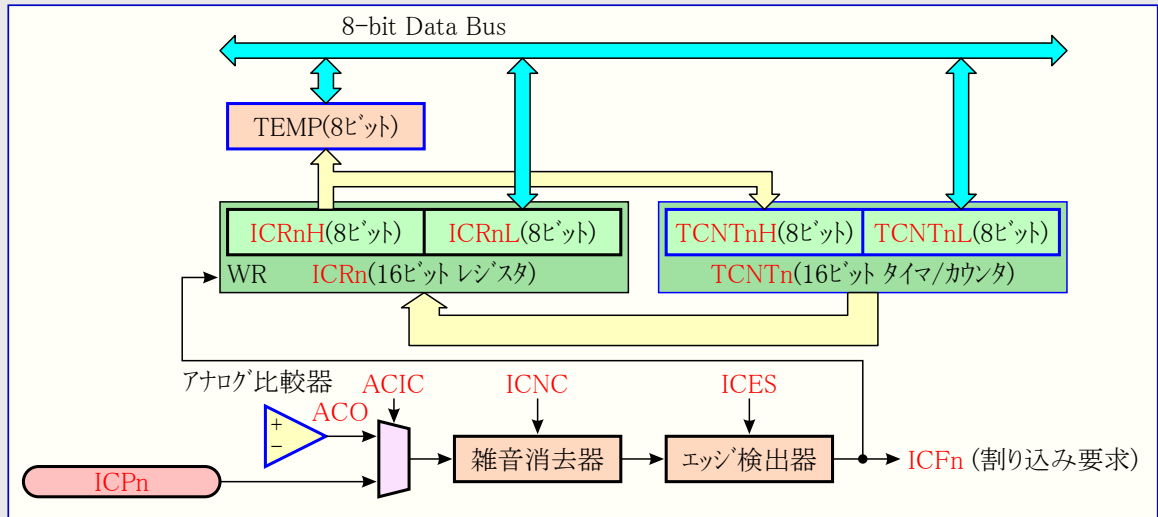
**タイマ/カウンタ溢れ(TOV1)フラグ**はWGM13~0ビットによって選択された動作種別に従って設定(=1)されます。TOV1はCPU割り込み発生に使えます。

## 12.5. 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性の計算に使えます。代わりに時間印は出来事の記録作成にも使えます。

捕獲入力部は図39.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。

図39. 捕獲入力部構成図



注: ACO, ACICはアナログ比較器制御/状態レジスタ(ACSRA)内のビットです。

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、その変化がエッジ検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF1)はTCNT1値がICR1に複写される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も解除(0)できます。

捕獲レジスタ(ICR1)の16ビット値読み込みは、初めに下位バイト(ICR1L)、その後に上位バイト(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にだけ書けます。これらの場合、TOP値がICR1に書かれ得る前に波形生成種別(WGM13~0)ビットが設定されなければなりません。ICR1に書く時は下位バイトがICR1Lに書かれる前に、上位バイトがICR1H I/O位置に書かれなければなりません。

16ビットレジスタアクセス法のより多くの情報については81頁の「16ビットレジスタのアクセス」を参照してください。

### 12.5.1. 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(ACSRA)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択されます。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)の両入力は、T1ピン(89頁の図50.参照)についてと同じ技法を使って採取されます。エッジ検出器も全く同じです。けれども雑音除去が許可されると、付加論理回路がエッジ検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されないなら、雑音除去器とエッジ検出器の入力が常に許可されることに注意してください。

捕獲入力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。

### 12.5.2. 雑音除去器

雑音除去器は雑音耐性を改善するために簡単なデジタル濾波器機構を使います。連続する採取が4段の深さのパイプラインで監視されます。信号は4つ全ての採取が等しい時にだけ変更を許されるエッジ検出器に行きます。

雑音除去器はタイマ/カウンタ制御レジスタB(TCCR1B)の捕獲入力雑音除去許可(ICNC1)ビットの設定(1)によって許可されます。許可時、雑音除去器はICR1が更新される前に入力に印加される変更に対して4システムクロック周期の追加遅延をもたらします。

雑音除去器はシステムクロックを使い、従って前置分周器によって影響を及ぼされません。

### 12.5.3. 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は新しい値で上書きされます。この場合、捕獲の結果は不正にされます。

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

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

外部信号のデューティ比測定は各捕獲後に起動端が変更されることを必要とします。検出端の変更はICR1が読まれてしまった後に可能な限り早く行われなければなりません。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:エッジ変更によってICF1が設定(1)されることを想定)。周波数のみの測定について(割り込み処理が使われる場合)、ICF1の解除(0)は必要とされません。

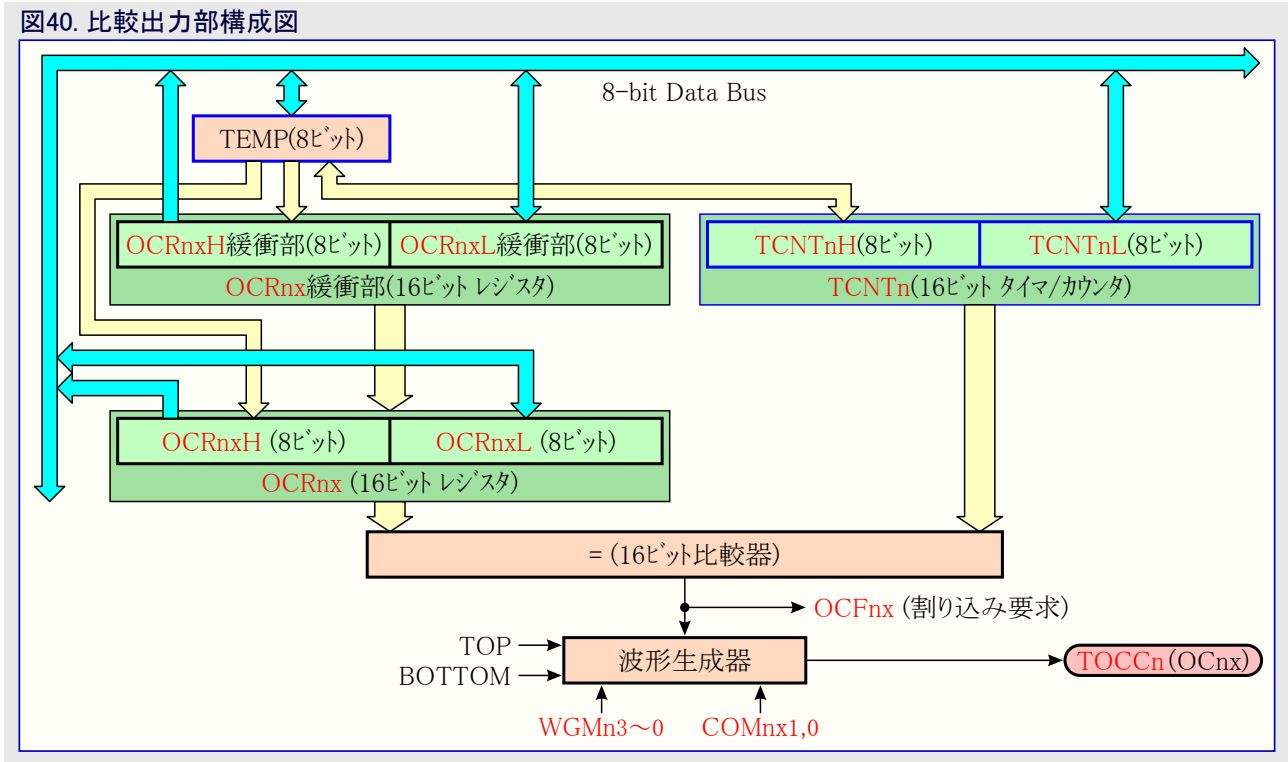
### 12.6. 比較出力部

この16ビット比較器は**TCNT1**と**比較レジスタ(OCR1x)**を継続的に比較します。TCNT1とOCR1xが等しければ、比較器は一致を指示します。この一致は**次のタイマ/カウンタクロック周期**で**比較割り込み要求フラグ(OCF1x)**を設定(1)します。許可(I=1, OCIE1x=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF1xは割り込みが実行されると自動的に解除(0)されます。代わりにOCF1xはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**によって設定された動作種別に従った出力を生成するのにこの一致信号を使います。TOPとBOTTOM信号は動作種別(76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40.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40. 比較出力部構成図





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

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

16ビットレジスタアクセス法のより多くの情報については81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。

### 12.6.1. 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)。

### 12.6.2. 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。

### 12.6.3. 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、TCNT1を変更する時に危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。可変TOP値のPWM動作でTOPに等しいTCNT1を書いてはいけません。(行った場合)TOPに対する比較一致は無視され、カウンタは\$FFFFへ(計数を)続けます。同様にカウンタが下降計数のとき、BOTTOMに等しいTCNT1値を書いてはいけません。

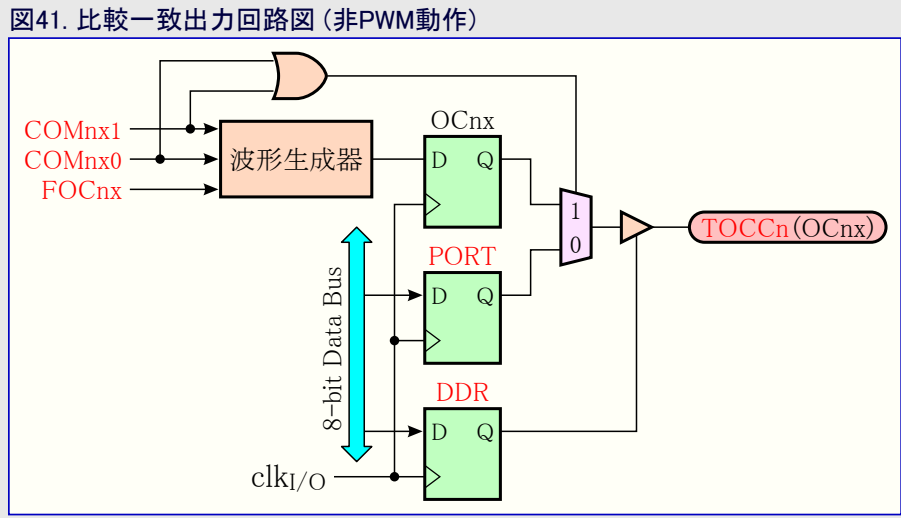
OC1xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)ストローブビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

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



## 12.7. 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41.は $COM1x1,0$ ビット設定によって影響される論理回路の単純化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。 $COM1x1,0$ ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。 $OC1x$ の状態を参照するとき、その参照は $OC1x$ ピンでなく内部 $OC1x$ レジスタに対してです。システムリセットが起ると、 $OC1x$ レジスタは0にリセットされます。



$COM1x1,0$ ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力( $OC1x$ )によって無効にされます。けれども $OC1x$ ピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。 $OC1x$ ピンに対するポート方向レジスタのビット(DDR\_ $OC1x$ )は $OC1x$ 値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は一般的に波形生成種別と無関係ですが、いくつかの例外があります。詳細については表39.、表40.、表41.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。83頁の「16ビットタイマ/カウンタ用レジスタ」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。

### 12.7.1. 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83頁の表39.を参照してください。高速PWM動作については83頁の表40.、位相基準PWMと位相/周波数基準PWMについては83頁の表41.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)スローブビットを使うことによって直ちに効果を得ることを強制できます。

## 12.8. 動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は波形生成種別(WGM13~0)ビットと比較出力選択(COM1x1,0)ビットの組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM1x1,0ビットは生成されたPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)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。75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80頁の「タイマ/カウンタのタイミング」を参照してください。

### 12.8.1. 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。

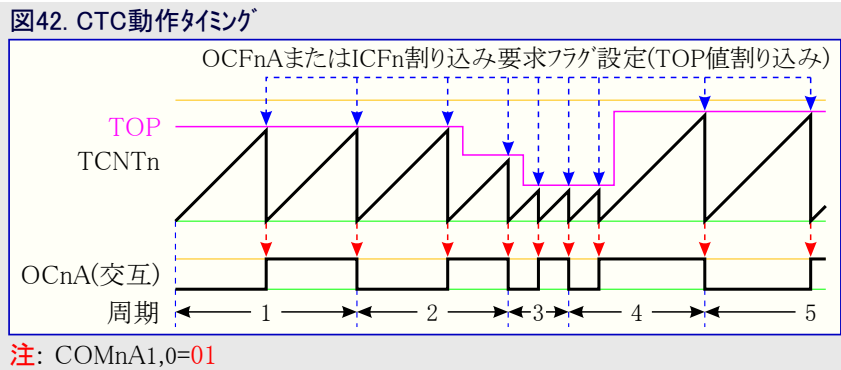
捕獲入力部は標準動作での使用が容易です。けれども外部の事象間の最大間隔がタイマ/カウンタの分解能(16ビット長)を越えてはならないことに気付いてください。事象間の間隔が長すぎる場合、捕獲部に対して分解能を拡張するために、タイマ/カウンタ溢れ割り込みまたは前置分周器が使われなければなりません。

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

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

CTC動作についてのタイミング図は図42.で示されます。カウンタ(TCNT1)値はOCR1AまたはICR1のどちらかで比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT1)は解除(\$0000)されます。



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

CTC動作で波形出力を生成するため、OC1A出力は比較出力選択(COM1A1,0)ビットを交互動作(=01)に設定することによって各比較一致での論理レベル交互切替に設定できます。OC1A値はそのピンに対するデータ方向が出力(DDR\_OC1A=1)に設定されない限り、ポートピンで見えないでしょう。生成された波形はOCR1Aが0(\$0000)に設定される時に $f_{OC1A} = f_{clk\_I/O} / 2$ 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。

### 12.8.3. 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、BOTTOMで設定(1)されます。反転出力動作(COM1x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準や位相/周波数基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

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

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

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)のどれかと一致するまで増加されます。そしてカウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$0000)されます。高速PWM動作のタイミング図は図43.

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

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

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

ICR1がTOP値を定義するのに使われるとき、ICR1を更新する手順はOCR1Aの更新と異なります。ICR1は2重緩衝されません。これは前置分周なしまたは低い前置分周値でカウンタが走行している時にICR1が小さな値に変更される場合、書かれた新しいICR1値がTCNT1の現在値よりも小さくなる危険を意味します。その後の結果はカウンタが(その回の)TOP値での比較一致を失うことです。その後のカウンタは比較一致が起こせるのに先立って、MAX値(\$FFFF)へそして次に\$0000から始める計数をしなければならないでしょう。けれども、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。83頁の表40をご覧ください。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$0000、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でのOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

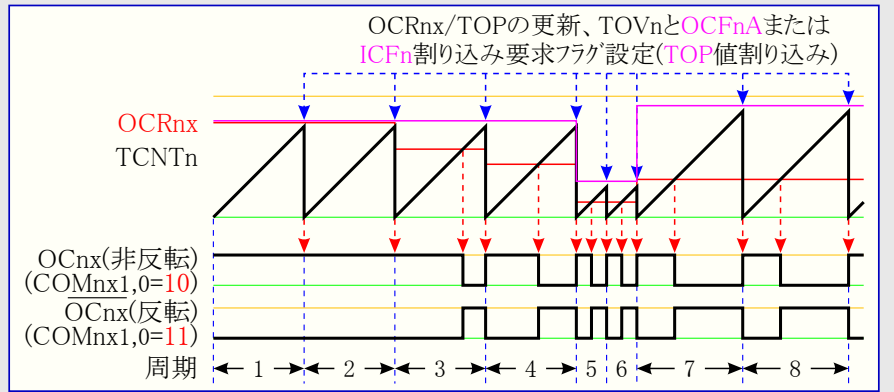
PWM出力周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

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

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

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

図43. 高速PWM動作タイミング





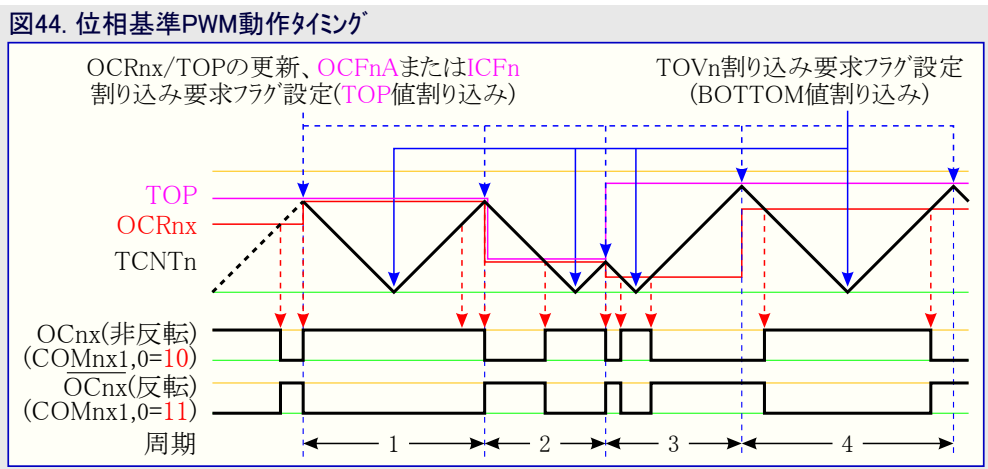
## 12.8.4. 位相基準PWM動作

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、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

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

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

位相基準PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0001,0010,0011)、ICR1値(WGM13~0=1010)またはOCR1A値(WGM13~0=1011)のどれかと一致するまで増やされます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期TOPと等しくなります。位相基準PWM動作のタイミング図は図44.で示されます。本図はOCR1AかICR1がTOPを定義するのに使われる時の位相基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。



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

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。固定TOP値を使う場合、どのOCR1xが書かれる時も、未使用ビットが0で隠(に置換)されることに注意してください。図44.で示される第3周期が図解するように、タイマ/カウンタが位相基準PWM動作で走行中にTOPを積極的に変更するのは、非対称出力で終わることが有り得ます。これに対する理由はOCR1xレジスタの更新時に見出せます。OCR1x更新はTOPで起きるので、PWM周期はTOPで始まりそして終わります。これは下降傾斜長が直前のTOP値によって決定され、一方上昇傾斜長は新しいTOP値で決定されることを意味します。これら2つの値(TOP)が違うとき、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値を使う時に2つの動作種別間に現実的な違いはありません。

位相基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます(83頁の表41.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

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

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

OCR1xの両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。TOP値定義にOCR1Aが使われ(WGM13~0=1011)、COM1A1,0=01なら、OC1A出力はデューティ比50%で交互に変化します。

### 12.8.5. 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、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準と位相/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44と図45参照)。

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間、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45.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。

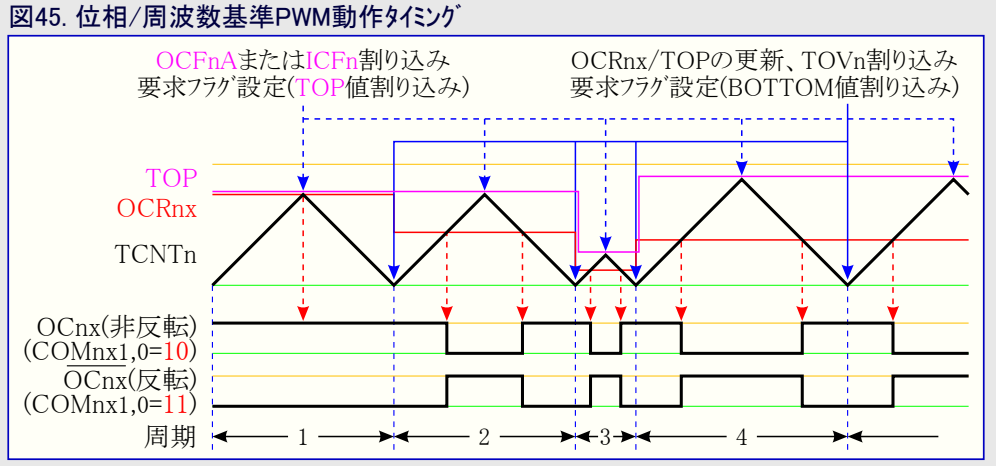


図45. 位相/周波数基準PWM動作タイミング

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

図45.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。

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(83頁の表41.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えます。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。

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

OCR1xの両端値は位相/周波数基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。TOP値定義にOCR1Aが使われ(WGM13~0=1001)、COM1A1,0=01なら、OC1A出力はデューティ比50%で交互に変化します。



## 12.9. 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重緩衝を使う動作種別のみ)の情報を含みます。図46は $OCF1x$ の設定についてのタイミング図を示します。

図46. 前置分周なし(1/1)のタイマ/カウンタ、 $OCF1x$ 設定 タイミング

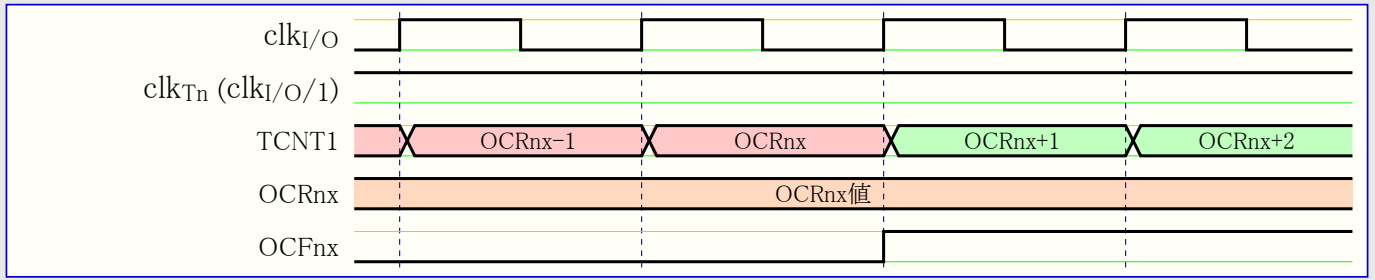


図47. は同じタイミング データを示しますが、前置分周器が許可されています。

図47. 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

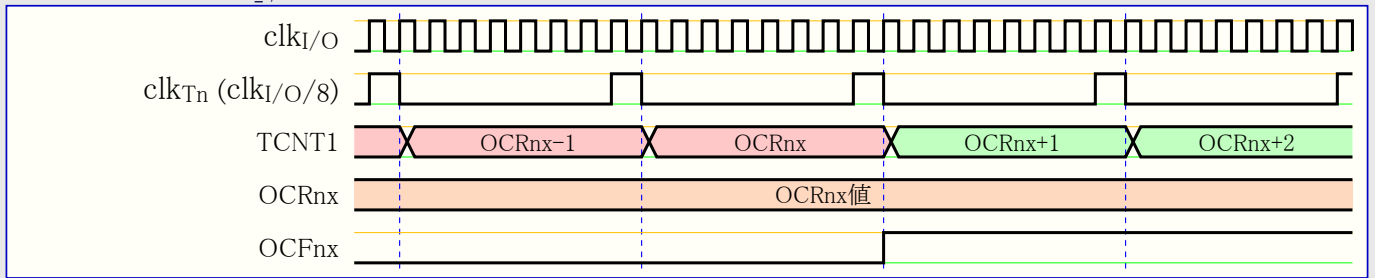


図48. は各動作種別でのTOP近辺の計数手順を示します。位相/周波数基準PWM動作使用時の $OCR_{nx}$ レジスタはBOTTOMで更新されます。タイミング図は同じになりますが、当然TOPはBOTTOMで、TOP-1はBOTTOM+1でなどのように置き換えられます。BOTTOMで $TOV1$ を設定(1)する動作種別についても、同様な名称変更が適用されます。

図48. 前置分周なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング

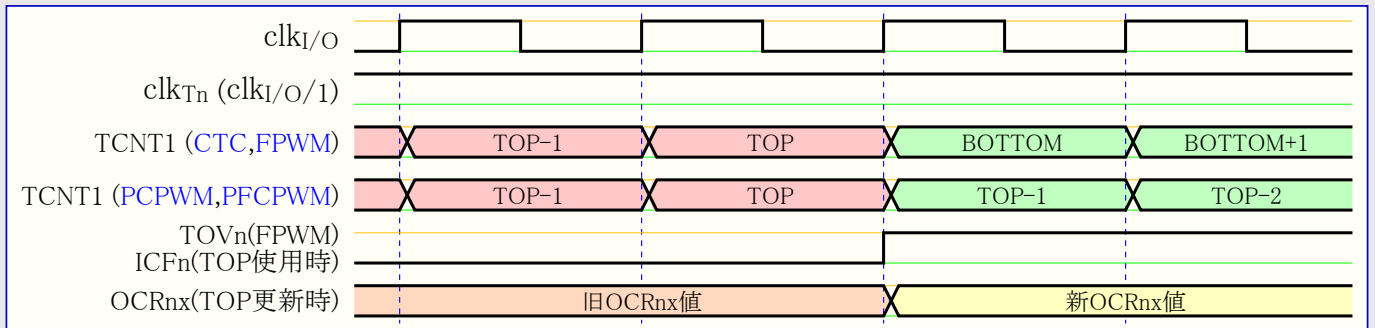
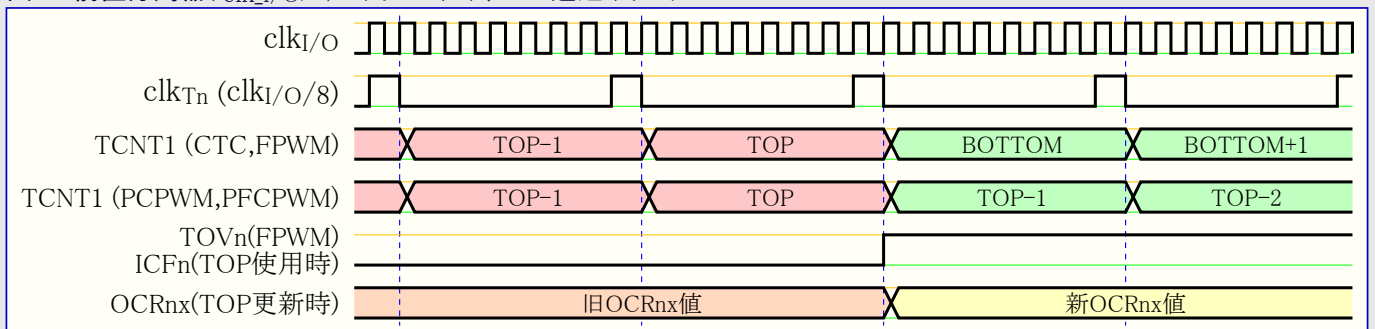


図49. は同じタイミング データを示しますが、前置分周器が許可されています。

図49. 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



## 12.10. 16ビットレジスタのアクセス

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

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

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

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

### アセンブリ言語プログラム例

```
~ ;
; [16ビット($01FF)書き込み]
LDI R17, $01 ; $01FFの上位バイト値取得
LDI R16, $FF ; $01FFの下位バイト値取得
OUT TCNT1H, R17 ; 上位バイト設定(一時レジスタ)
OUT TCNT1L, R16 ; 下位バイト設定(一時レジスタ⇒上位バイト)
; [16ビット読み込み]
IN R16, TCNT1L ; 下位バイト取得(上位バイト⇒一時レジスタ)
IN R17, TCNT1H ; 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; /* 16ビット($01FF)書き込み */
i = TCNT1; /* 16ビット読み込み */
~ /* */
```

**注:** 5頁の「コード例」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ことに注意することが重要です。16ビットレジスタをアクセスする2命令間で割り込みが起き、割り込みコードがその16ビットタイマ/カウンタレジスタの同じ若しくは他の何れかをアクセスすることによって一時レジスタを更新する場合、割り込み外のその後のアクセス結果は不正にされます。従って主コードと割り込みコードの両方が一時レジスタを更新するとき、主コードは16ビットアクセス中の割り込みを禁止しなければなりません。

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

注: 5頁の「コード例」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 */
    _CLI();                             /* 全割り込み禁止 */
    TCNT1 = i;                          /* TCNT1値を設定 */
    SREG = sreg;                       /* 全割り込み許可フラグ(I)を復帰 */
}
```

注: 5頁の「コード例」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。

### 12.10.1. 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。

## 12.11. 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、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39.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39. 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40.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40. 高速PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

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

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMでの設定(1)または解除(0)は実行されます。より多くの詳細については77頁の「高速PWM動作」をご覧ください。

表41.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41. 位相基準または位相/周波数基準PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

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

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。より多くの詳細については78頁の「位相基準PWM動作」をご覧ください。

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

タイマ/カウンタ制御レジスタB(TCCR1B)で得られるWGM13,2ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大計数(TOP)値供給元、使われるべき波形生成のどの形式かを制御します(表42.参照)。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。76頁の「動作種別」をご覧ください。

表42. 波形生成種別選択

番号	WGM13~0	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)に設定することが捕獲起動入力雑音消去器を活性(有効)にします。雑音消去器が有効にされると、捕獲起動入力(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のに、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、捕獲入力はそれによって4発振器(システムクロック)周期遅らされます。

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

このビットは出来事での捕獲を起動するのに使われる捕獲起動入力(ICP1)ピンのどちらかのエッジを選択します。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予約されています。将来のデバイスとの互換性を保証するため、このビットはレジスタが書かれる時に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)

タイマ/カウンタ制御レジスタA(TCCR1A)で得られるWGM11,0ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大計数(TOP)値供給元、使われるべき波形生成のどの形式かを制御します。

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択します。図46と図47をご覧ください。

表43. 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	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	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動作で扱う時にTCCR1Cが書かれる場合、これらのビットは0に設定されなければなりません。FOC1A/FOC1Bビットに論理1を書くと波形生成部で直ちに比較一致が強制されます。OC1x出力はCOM1x1,0ビット設定に従って変更されます。FOC1A/FOC1Bビットがスローブ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スローブ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)

これらのビットは将来の使用に予約されています。将来のデバイスとの互換性を保証するため、これらのビットはレジスタが書かれる時に0を書かれなければなりません。

12.11.4. TOCPMSA1, TOCPMSA0 – タイマ/カウンタ比較出力ピン多重器選択レジスタ (Timer/Counter Output Compare Pin Mux Selection Registers)

ビット (\$E9)	15	14	13	12	11	10	9	8	TOCPMSA1
	TOCC7S1	TOCC7S0	TOCC6S1	TOCC6S0	TOCC5S1	TOCC5S0	TOCC4S1	TOCC4S0	
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 (\$E8)	7	6	5	4	3	2	1	0	TOCPMSA0
	TOCC3S1	TOCC3S0	TOCC2S1	TOCC2S0	TOCC1S1	TOCC1S0	TOCC0S1	TOCC0S0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット15~0 – TOCCnS1,0 : タイマ/カウンタ比較出力チャネル選択 (Timer/Counter Output Compare Channel Select)

TOCCnS1とTOCCnS0のビットはどのタイマ/カウンタ比較出力が対応するTOCCnピンに配線されるかを選びます。右表で示されるように、2つのタイマ/カウンタは出力ピンに配線することができる4つの可能な比較出力を提供します。

表44. TOCCnピンに対するタイマ/カウンタ比較出力選択

TOCCnS1	TOCCnS0	TOCCn出力 (注)
0	0	OC0A
0	1	OC0B
1	0	OC1A
1	1	OC1B

注: 48頁の「ポートCの交換機能」をご覧ください。

### 12.11.5. TOCPMCOE – タイマ/カウンタ比較出力ピン多重器チャネル出力許可レジスタ (Timer/Counter Output Compare Mux Channel Output Enable)

ビット (\$E2)	7	6	5	4	3	2	1	0	
	TOCC7OE	TOCC6OE	TOCC5OE	TOCC4OE	TOCC3OE	TOCC2OE	TOCC1OE	TOCC0OE	TOCPMCOE
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 – TOCCnOE : タイマ/カウンタ比較出力チャネル出力許可 (Timer/Counter Output Compare Channel Output Enable)

これらのビットは比較出力動作が選択されているか否かに拘らず、対応するTOCCnピンで選択した比較出力チャネルを許可します。

### 12.11.6. 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	
	(MSB)								TCNT1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$84)	7	6	5	4	3	2	1	0	
								(LSB)	TCNT1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この2つのタイマ/カウンタ I/O位置(TCNT1HとTCNT1Lを合わせたTCNT1)は、読み書き両方についてタイマ/カウンタ部の16ビットカウンタに直接アクセスします。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。81頁の「16ビットレジスタのアクセス」をご覧ください。

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

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

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

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

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

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

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

この比較レジスタは容量が16ビットです。CPUがこれらのレジスタへ書く時に上位と下位の両バイトが同時に書かれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。81頁の「16ビットレジスタのアクセス」をご覧ください。

### 12.11.9. 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 (\$69)	15 (MSB)	14	13	12	11	10	9	8	ICR1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$68)	7	6	5	4	3	2	1	0 (LSB)	ICR1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この捕獲レジスタは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有されます。81頁の「16ビットレジスタのアクセス」をご覧ください。

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

ビット (\$6F)	7	6	5	4	3	2	1	0	TIMSK1
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7,6,4,3 – Res : 予約 (Reserved)

これらのビットは将来の使用に予約されています。将来のデバイスとの互換性を保証するため、これらのビットはレジスタが書かれる時に0を書かれなければなりません。

#### ● 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(32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(32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(32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(32頁の「割り込み」参照)が実行されます。

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

ビット \$16 (\$36)	7	6	5	4	3	2	1	0	TIFR1
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7,6,4,3 – Res : 予約 (Reserved)

これらのビットは将来の使用に予約されています。将来のデバイスとの互換性を保証するため、これらのビットはレジスタが書かれる時に0を書かれなければなりません。

#### ● 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**84頁の表42**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。

### 13. タイマ/カウンタの前置分周器

タイマ/カウンタ0とタイマ/カウンタ1は同じ前置分周器部を共有しますが、タイマ/カウンタは異なる前置分周器設定ができます。以下の記述は両タイマ/カウンタに適用します。これは一般名、n (n=0,1)として使われます。

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。

#### 13.1. 前置分周器リセット

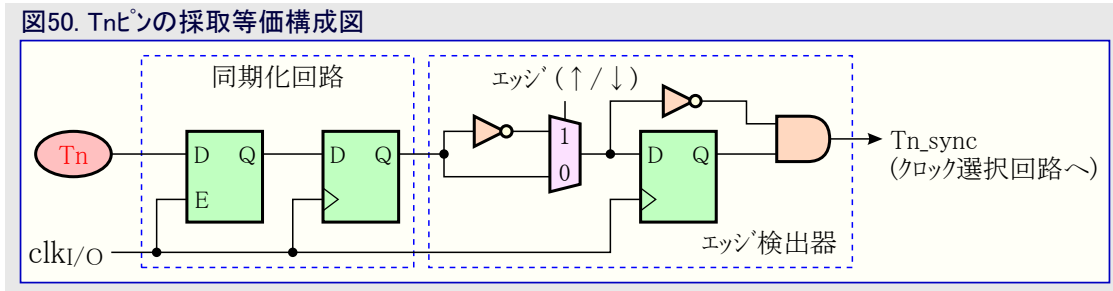
この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ( $T_n$ )によって共有されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが使われる状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステムクロック周期数は、Nが前置分周値(8, 64, 256, 1024)とすると、 $1 \sim N+1$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

プログラム実行にタイマ/カウンタを同期することに対して前置分周器リセットを使うことが可能です。しかし、同じ前置分周器を共有する他のタイマ/カウンタも前置分周を使う場合、注意が必要とされなければなりません。

#### 13.2. 外部クロック元

$T_n$ 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( $f_{clk\_Tn}$ )として使えます。この $T_n$ 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、その後エッジ検出器を通して通過されます。図50.は $T_n$ 同期化とエッジ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。

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



同期化とエッジ検出器論理回路は $T_n$ ピンへ印加されたエッジから計数器が更新されるまでに2.5~3.5システムクロック周期の遅延をもたらします。

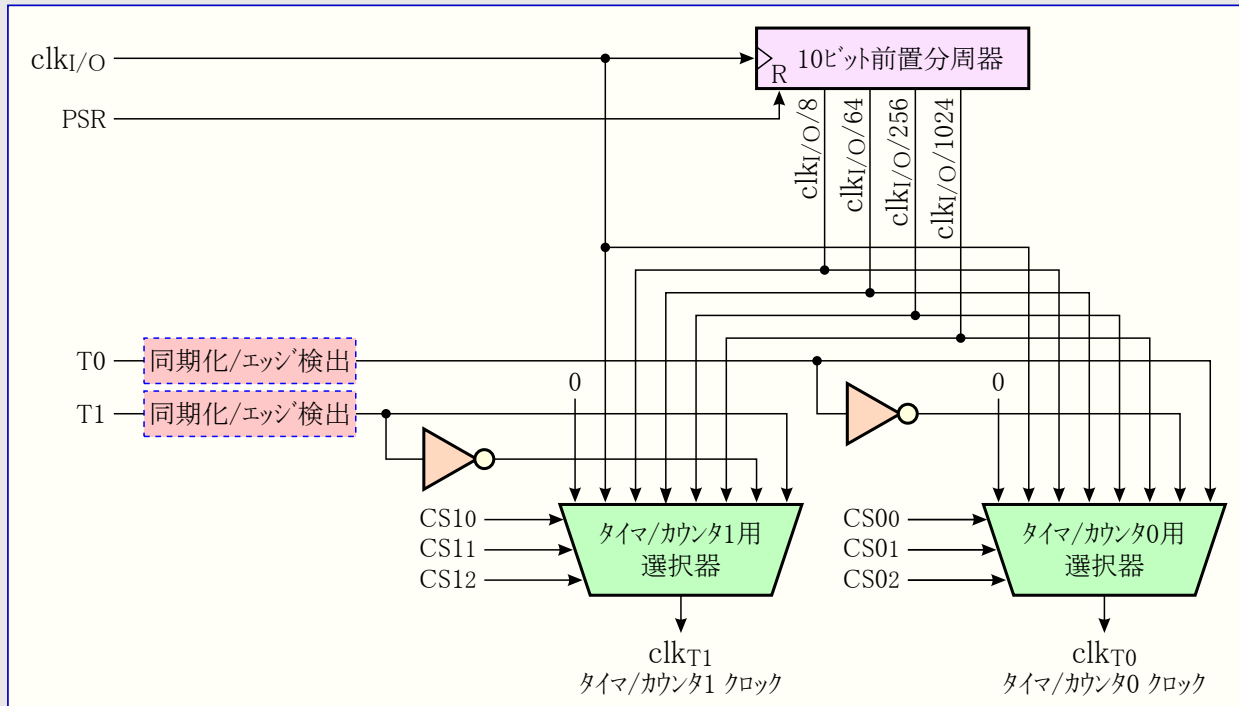
クロック入力の許可と禁止は $T_n$ が最低1システムクロック周期に対して安定してしまっている時に行われなければならない、さもなければ不正なタイマ/カウンタクロックパルスが生成される危険があります。

印加された外部クロックの各半周期は正しい採取を保証するために1システムクロック周期より長くなければなりません。この外部クロックは50%/50%デューティ比で与えられるものとして、システムクロック周波数の半分未満( $f_{EXTclk} < f_{clk\_I/O}/2$ )であることが保証されなければなりません。エッジ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51. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器構成図



注: 入力ピン(T0/T1)の同期化/エッジ検出論理回路は図50.で示されます。

### 13.3. タイマ/カウンタ前置分周器制御関係レジスタ

#### 13.3.1. 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	-	-	-	-	-	-	PSR	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PSR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。これはタイマ/カウンタが停止され、設定中に進行する危険なしに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、PSRビットはハードウェアによって解除(0)され、タイマ/カウンタが計数を始めます。

- ビット0 – PSR : タイマ/カウンタ前置分周器リセット (Prescaler Reset)

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。通常、TSMビットが設定(1)されている場合を除いて、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全てのタイマ/カウンタに影響を及ぼすことに注意してください(訳注:共通性から本行追加)。

## 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ATtiny828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。SPI単位部は図52.で図解されます。

SPI単位部を許可するために電力削減レジスタのPRSPIビットは0を書かれなければなりません。26頁の「PRR - 電力削減レジスタ」をご覧ください。

SPIでの主装置と従装置のCPU間相互接続は図53.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のに必要なクロックパルスを生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。これは通信を開始す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クロック発生器を始動し、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。

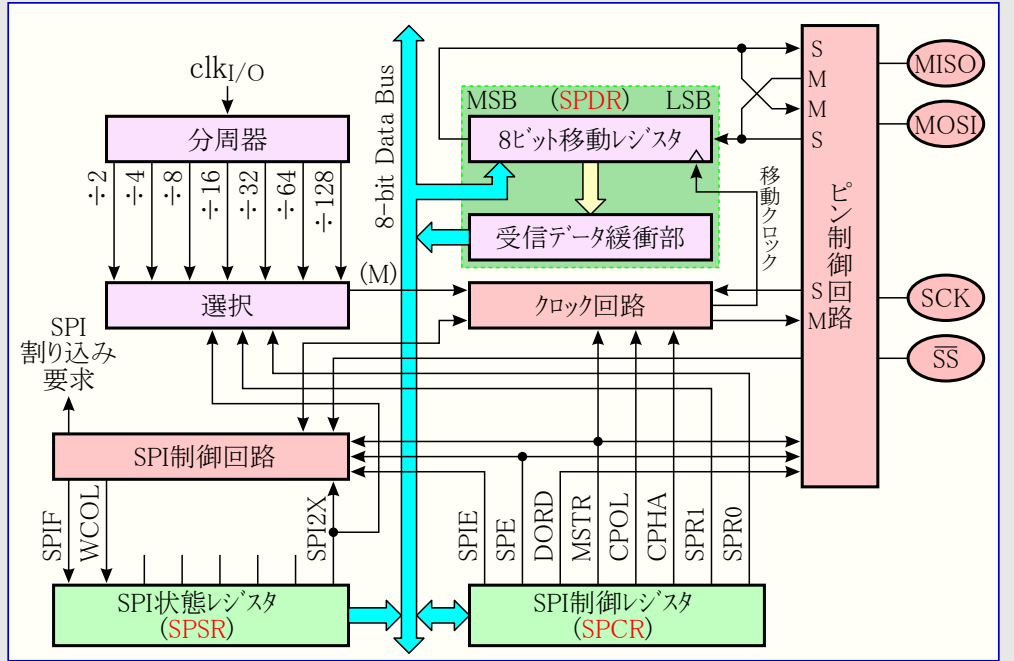
このシステムは送信側で単一緩衝、受信側で2重緩衝です。これは一連の移動全体が完了される前に送信されるべきバイトがSPIデータレジスタ(SPDR)へ書けないことを意味します。けれども、データを受信するとき、次のデータが完全に移動入力される前に受信したデータがSPIデータレジスタ(SPDR)から読まれなければなりません。さもなければ始めのバイトは失われます。

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、LowとHighの最小周期は以下であるべきです。

- Low周期 : 2 CPUクロック周期より長い
- High周期 : 2 CPUクロック周期より長い

SPIが許可されると、MOSI, MISO, SCK, SSピンのデータ方向は表45.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、43頁の「交換ポート機能」を参照してください。

図52. SPI構成図



注: SPIピン配置については2頁の「ピン配置」と48頁の表25.と51頁の表27.をご覧ください。

図53. SPI 主装置/従装置の接続

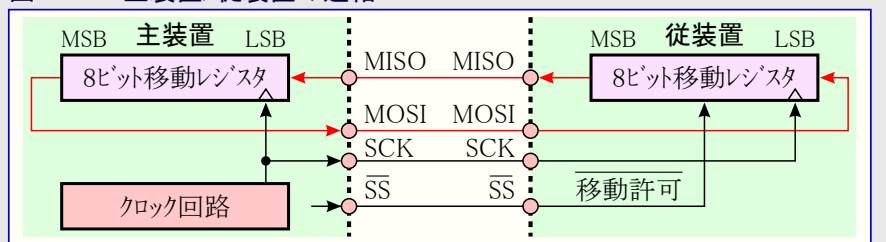


表45. SPIピン方向規定

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D方向レジスタ(DDRD)の指定	入力
MISO	入力	ポートD方向レジスタ(DDRD)の指定
MOSI	ポートD方向レジスタ(DDRD)の指定	入力
SS	ポートC方向レジスタ(DDRC)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「交換ポート機能」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PB3ピンに配置されるなら、DD\_MOSIはDDB3、DDR\_SPIはDDRBに置き換えま

#### 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: IN     R17, SPSR                          ;SPI状態レジスタ値取得
            SBRS   R17, 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」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:   IN     R16, SPSR                       ;SPI状態レジスタ値取得
            SBRS   R16, 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」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は出力になります。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。一旦 $\overline{SS}$ ピンがHighに駆動されると、SPI論理回路が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、従装置ビット計数器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データの取り落としをします。

### 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、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時に周辺回路によって $\overline{SS}$ 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択して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/従選択(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)されていること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54と図55.で示されます。

図54. SPIデータ転送形式 (CPHA=0)

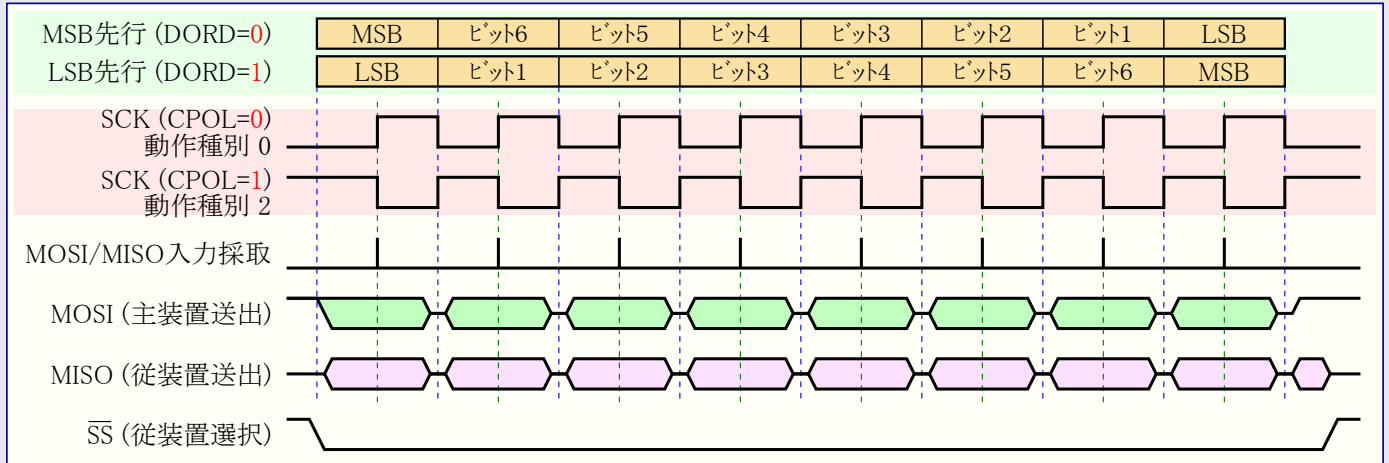
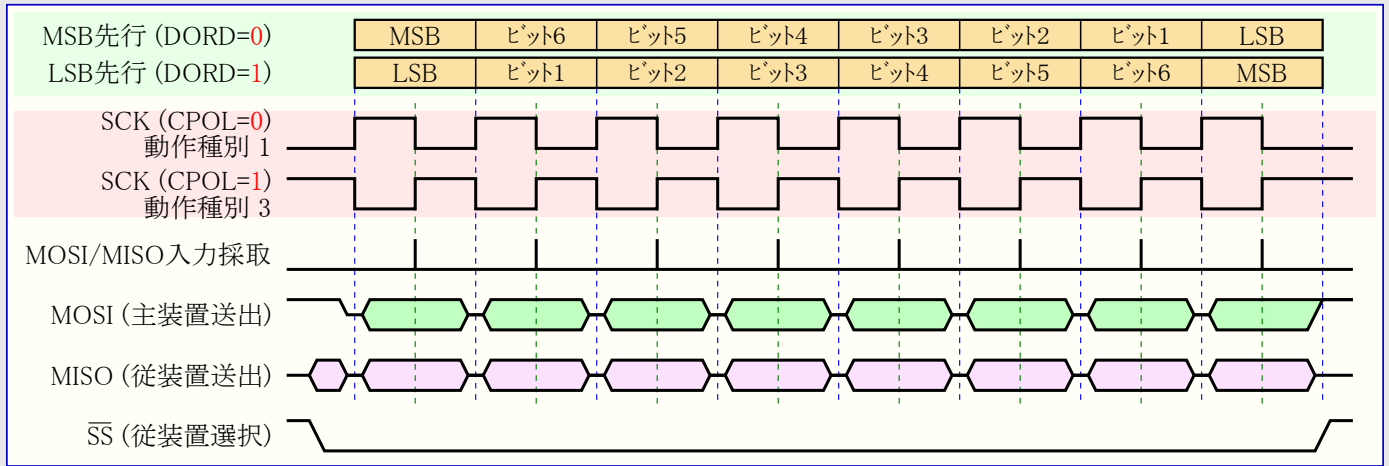


図55. SPIデータ転送形式 (CPHA=1)



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47.と表48.を要約する表46.で示されます。

表46. SPI動作形態

SPI動作形態番号	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)**

このビットが設定(1)されると、SPI割り込みが許可されます。**ステータスレジスタ(SREG)の全割り込み許可(1)ビット**が設定(1)なら、**SPI状態レジスタ(SPSR)のSPI割り込み要求フラグ(SPIF)**が設定(1)される時にSPI割り込み処理ルーチンが実行されます。

- **ビット6 – SPE : SPI許可 (SPI Enable)**

このビットが設定(1)されると、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)**

このビットが設定(1)されると、データ語のLSBが最初に転送されます。

このビットが解除(0)されると、データ語の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択します。 $\overline{SS}$ 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、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。その後には使用者は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)**

このビットが設定(1)されると、SCKはアイドル時にHighです。このビットが解除(0)されると、SCKはアイドル時にLowです。例については図54.と図55.を参照してください。CPOL機能は右で要約されます。

表47. 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54.と図55.を参照してください。CPHA機能は右で要約されます。

表48. 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 $f_{osc}$ 間の関連は下で要約されます。

表49. 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)**

このビットは直列転送完了時に設定(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(1)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時に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、その後にはSPIデータレジスタ(SPDR)に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)

このビットはデータ転送中にSPIデータレジスタ(SPDR)が書かれた時に設定(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 – 予約** (Reserved)

これらのビットは予約されており、常に0を読みます。

- **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、SPIが主装置動作の時にSCK速度(SCK周波数)が倍にされます(表49参照)。これは最小SCK周期が2CPU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時にSPIはfOSC(CPUクロック周波数)/4またはそれ以下での動作のみ保証されます。

### 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が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。

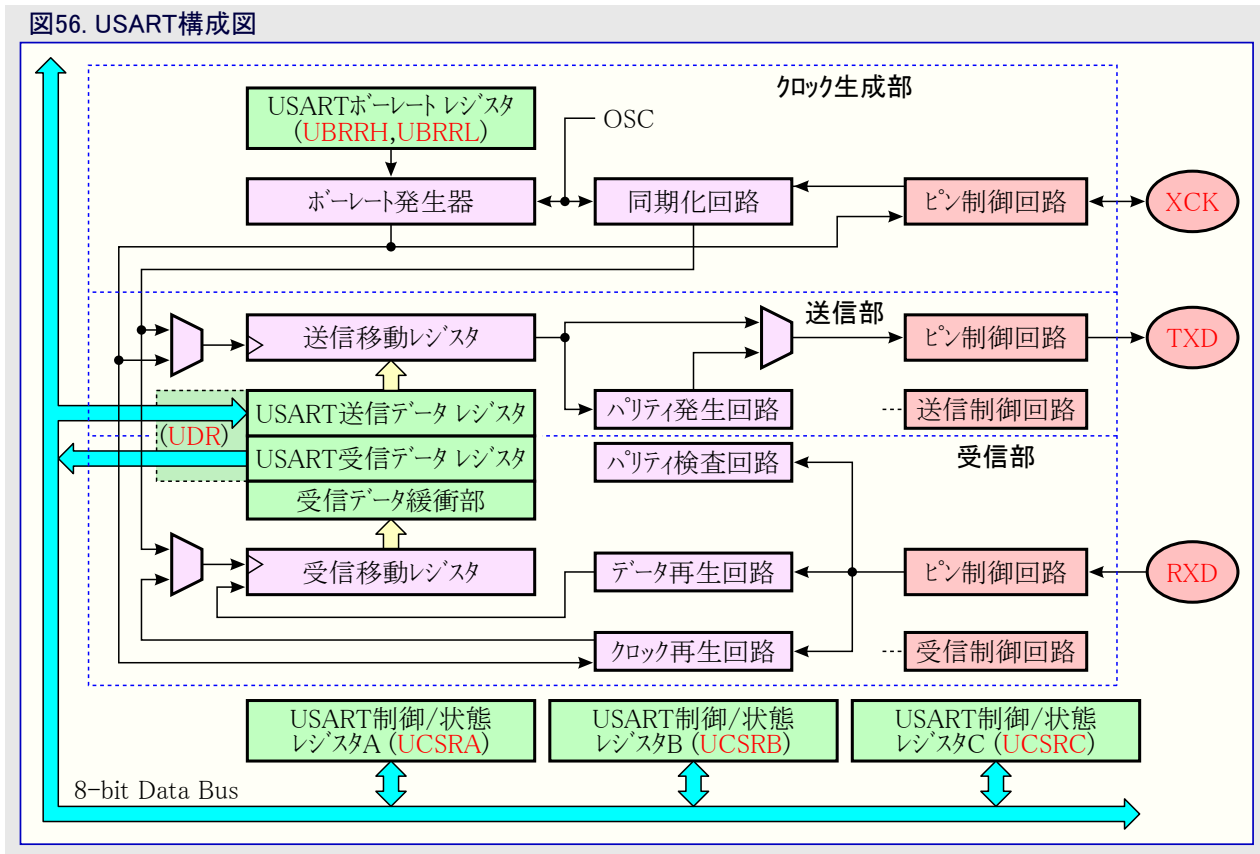
## 15. USART

### 15.1. 特徴

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

### 15.2. 概要

USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は高い柔軟性をもつ直列通信機能です。USARTの簡略構成図は図56.で示されます。CPUがアクセス可能なレジスタとI/Oピンは赤文字(訳注:原文は太字)で示されます。



USARTピン配置については2頁の「ピン配置」と43頁の「交換ポート機能」をご覧ください。

図56.内の構成図内の破線は(上から一覧にされる)以下のようなUSARTの3つの主要部分です。

- クロック生成部
- 送信部
- 受信部

クロック生成論理部はボーレート発生器と(同期従装置動作での外部クロック入力用の)同期化論理回路から成ります。転送クロック(XCK)ピンは同期転送動作でだけ使われます。

送信部は単一書き込み緩衝部(UDR)、直列移動レジスタ、パリティ発生器、異なる直列フレーム形式を扱うための制御論理回路から成ります。書き込み緩衝部はどんなフレーム間の遅れもなしにデータの継続転送を許します。

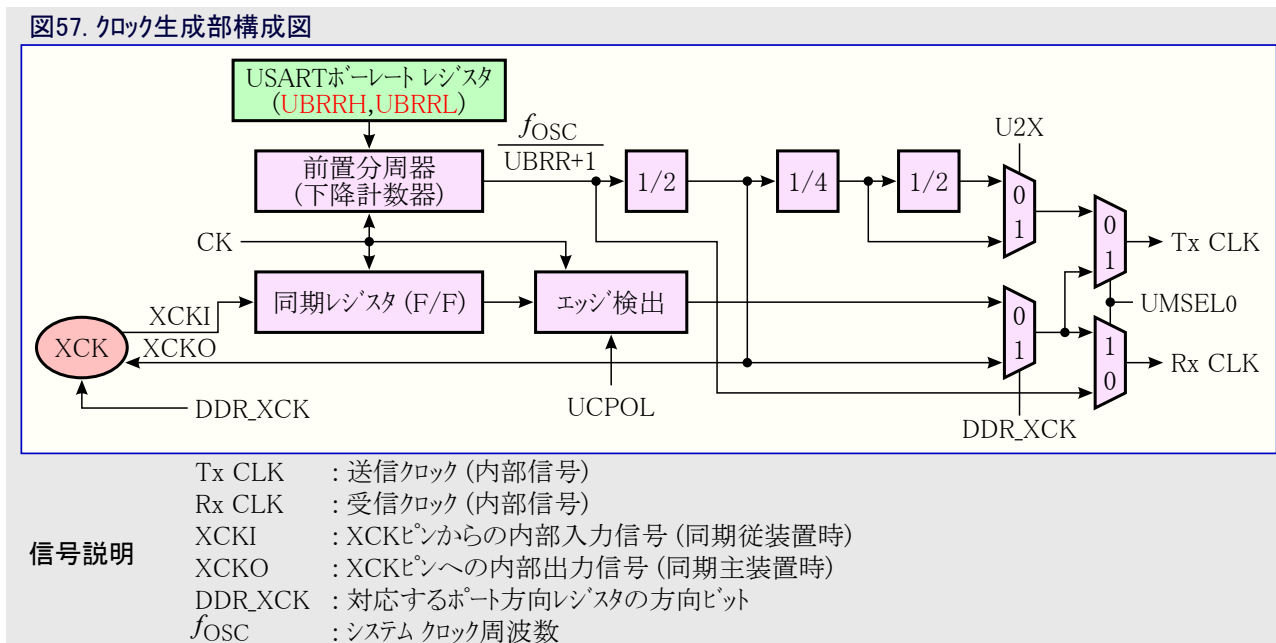
受信部はクロックとデータの再生部のため、USART部の最も複雑な部分です。再生部は非同期データ受信で使われます。再生部に加えて、受信部はパリティ検査器、制御論理回路、移動レジスタ、2重の受信緩衝部(UDR)を含みます。受信部は送信部と同じフレーム形式を支援し、以下の異常を検出することができます。

- フレーミング異常
- データ オーバーラン発生
- パリティ誤り

USARTを操作するためにUSART電力削減ビットが禁止されなければなりません。26頁の「PRR – 電力削減レジスタ」をご覧ください。

### 15.3. クロック生成

クロック生成論理回路は送受信部用基準クロックを作成します。クロック生成論理回路の構成図は図57.で示されます。



USARTは以下のような4つのクロック動作形態を支援します。

- 標準非同期動作
- 倍速非同期動作
- 主装置同期動作
- 従装置同期動作

USART動作種別選択(UMSEL0)ビット(112頁の「UCSRC – USART制御/状態レジスタC」をご覧ください。)は同期動作と非同期動作を選びます。非同期動作では速度が倍速許可(U2X)ビット(111頁の「UCSRA – USART制御/状態レジスタA」をご覧ください。)によって制御されます。

同期動作では、クロック元が内部(主装置動作)または外部(従装置動作)のどちらかかを、XCKピンが配置されるデータ方向レジスタ(DDR x)内のXCKピンの方向ビット(DDR\_XCK)が制御します。XCKピンは同期動作でだけ活性(有効)です。

#### 15.3.1. 内部クロック発生 – ボーレート発生器

内部クロック生成は非同期と同期主装置動作種別で使われます。本項の記述は図57.を参照してください。

USARTボーレートレジスタ(UBRR(UBRRH:UBRRL))と下降カウンタは設定可能な前置分周器またはボーレート発生器として機能するように接続されます。システムクロック( $f_{OSC}$ )で走行する下降カウンタは0への下降計数時毎またはUBRRLが書かれる時にUBRR値で設定されます。

1クロックはカウンタが0に達する毎に生成されます。これがボーレート発生器出力で $f_{OSC}/(UBRR+1)$ の周波数を持ちます。動作形態に依存して送信部はボーレート発生器出力を2,8,16分周します。ボーレート発生器出力は受信部クロックとデータの再生部によって直接使われます。けれども、再生部はUSART動作形態選択(UMSEL0)、倍速許可(U2X)、DDR\_XCKビットによって設定される動作形態に依存して2,8,16段を使う状態機構を使います。

表50.は内部的に生成したクロック元を使う各動作形態に於けるボーレート(bps)とUBRR値の計算式を含みます。

表50. ホールートレジスタ(UBRR)値計算式

動作種別	ホールート (注)	UBRR値
標準速非同期動作 (U2X=0)	$BAUD = \frac{f_{OSC}}{16 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{16 \times BAUD} - 1$
倍速非同期動作 (U2X=1)	$BAUD = \frac{f_{OSC}}{8 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{8 \times BAUD} - 1$
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{2 \times BAUD} - 1$

注: ホールートは転送速度(ビット/1秒)で定義されます。

BAUD : ホールート (bps)

UBRR : UBRRHとUBRRLレジスタ値 (0~4095)

$f_{OSC}$  : システム発振器クロック周波数

選択したシステム クロック周波数に対するいくつかのUBRR値の例が109頁の表53.で示されます。

### 15.3.2. 倍速動作 (U2X)

転送速度は倍速許可(U2X)ビット(111頁の「UCSRA – USART制御/状態レジスタ」をご覧ください。)を設定(1)することによって倍にすることができます。このビットの設定は非同期動作形態でだけ有効です。同期動作形態ではこのビットが解除(0)されるべきです。

このビットの設定(1)は事実上非同期通信に対する転送速度を倍にするホールート分周器の分周数を16から8に減らします。けれども、この場合に受信部が半分の採取数だけを使うことに注意してください。倍速動作ではデータとクロックの再生採取が16から8に減らされ、従ってより正確なホールート設定とシステム クロックが必要とされます。

送信部についての低下要因はありません。

### 15.3.3. 外部クロック

外部クロックは従装置同期動作形態で使われます。不確定レベル状態(マステーブル)の機会を最少とするためにXCKピンからの外部クロック入力同期化レジスタによって採取されます。その後同期化レジスタからの出力は送受信部で使われるのに先立ってエッジ検出器を通過します。この処理手順が2CPUクロック周期の遅延を持ち込み、従って最大外部(XCK)クロック周波数が次式によって制限されます。

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

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

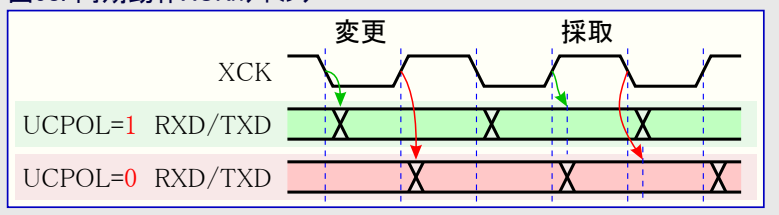
### 15.3.4. 同期クロック動作

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

データ採取とデータ変更に対してどちらのXCKクロック端が使われるかはXCK極性(UCPOL)ビット(112頁の「UCSRC – USART制御/状態レジスタC」をご覧ください。)で変更することができます。

図58.で示されるように、UCPOL設定(1)時、データはXCKの下降端で変更され、上昇端で採取されます。UCPOL解除(0)時、データはXCKの上昇端で変更され、下降端で採取されます。

図58. 同期動作XCKnタイミング



## 15.4. フレーム形式

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

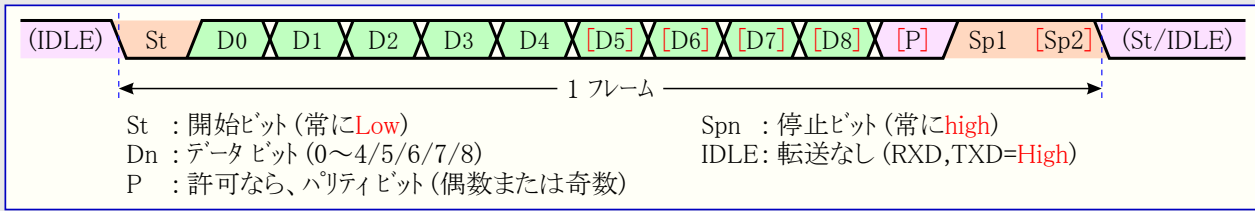
- 開始ビット : 1
- ビットデータ : 5、6、7、8、または9
- パリティビット : なし、奇数、または偶数パリティ
- 停止ビット : 1または2

フレームは最下位データビット(LSB)が後続する開始ビットで始まります。その後他のデータビット、最上位データビット(MSB)となる最後の1つが後続します。許可したなら、データビットの後、停止ビットの前にパリティビットが挿入されます。完全なフレームが送信されると、新規フレームによって直ちに後続されるか、または通信線をアイドル状態(high)に設定され得ます。

図59.は組み合わせ可能なフレーム形式を図解します。[ ]付きビットは任意選択です。



図59. 1フレームの構成



USARTによって使われるフレーム形式は以下のように、データ長選択(UCSZ2~0)ビット、パリティ選択(UPM1,0)ビット、停止ビット選択(USBS)ビット(112頁の「UCSRB – USART制御/状態レジスタ」と112頁の「UCSRC – USART制御/状態レジスタ」をご覧ください。)によって設定されます。

- USARTデータ長選択(UCSZ2~0)ビットはフレーム内のデータビット数を選びます。
- USARTパリティ形態(UPM1,0)ビットはパリティビットの形式を選びます。
- 1または2停止ビットのどちらかの選択は停止ビット選択(USBS)ビットによって行われます。受信部は第2停止ビットを無視します。従ってフレーミング異常(FE)は最初の停止ビットが0(Low)の場合にだけ検出されます。

受信部と送信部は同じ設定を使います。これらのどのビットの設定変更も、送受信部両方に対して進行中の通信を不正とすることに注意してください。

#### 15.4.1. パリティビット計算

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

$$PEVEN = D0 \text{ Ex-OR } D1 \text{ Ex-OR } D2 \text{ Ex-OR } \sim \text{ Ex-OR } D_{n-1}$$

$$PODD = D0 \text{ Ex-OR } D1 \text{ Ex-OR } D2 \text{ Ex-OR } \sim \text{ Ex-OR } D_{n-1} \text{ Ex-OR } 1$$

PEVEN : 偶数パリティを使うパリティビット  
 PODD : 奇数パリティを使うパリティビット  
 D<sub>n</sub> : データビットn

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

## 15.5. USARTの初期化

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

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

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

### アセンブリ言語プログラム例

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

### C言語プログラム例

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

**注:** 5頁の「コード例」をご覧ください。

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

## 15.6. データ送信 – USART送信部

USART送信部は送信許可(TXEN)ビット(112頁の「UCSRB – USART制御/状態レジスタB」をご覧ください。)を設定(1)することによって許可されます。送信部が許可されると、TXDピンの標準ピン動作はUSARTによって無視され、送信部の直列出力としての機能を与えられます。何かの送信を行う前に一度はポーレート、フレーム形式、動作種別が設定されなければなりません。同期動作が使われる場合、XCKピンの(受信)クロックは無視され、送信クロックとして使われます。

### 15.6.1. 5～8ビット データ フレーム送信

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

次のコード例はUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグのポーリングを基準とした簡単なUSART送信関数を示します。この関数が使われ得る前にUSARTが初期化されなければなりません。アセンブリ言語での送るべきデータはR16レジスタに格納されると仮定されます。

#### アセンブリ言語プログラム例

```
USART_Tx:  SBIS    UCSRA, UDRE          ;送信緩衝部空きでスキップ
           RJMP    USART_Tx          ;送信緩衝部空き待機
;
           OUT    UDR, R16           ;データ送信(送信開始)
           RET                          ;呼び出し元へ復帰
```

#### C言語プログラム例

```
void USART_Transmit(unsigned char data)
{
    while ( !(UCSRA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UDR = data;                          /* データ送信(送信開始) */
}
```

**注:** 5頁の「コード例」をご覧ください。

この関数は送信されるべき新規データを設定する前に、UDREの検査によって送信緩衝部が空になるのを単純に待ちます。送信緩衝部空き割り込みが使われる場合、その割り込み処理ルーチンがデータを緩衝部に書き込みます。

### 15.6.2. 9ビット データ フレーム送信

9ビットデータが使われる場合、データの低位バイトがUSARTデータレジスタ(UDR)に書かれるのに先立って第9ビットがUSART制御/状態レジスタB(UCSRB)の送信データビット8(TXB8)ビットに書かれなければなりません。次のコード例は9ビットデータを扱う送信関数を示します。アセンブリ言語での送るべきデータはR17:R16レジスタに格納されると仮定されます。

#### アセンブリ言語プログラム例

```
USART_Tx:  SBIS    UCSRA, UDRE          ;送信緩衝部空きでスキップ
           RJMP    USART_Tx          ;送信緩衝部空き待機
;
           CBI    UCSRB, TXB8         ;第9ビットを0に仮設定
           SBRC   R17, 0              ;送信すべき第9ビットが0でスキップ
           SBI    UCSRB, TXB8         ;第9ビットを1に設定
           OUT    UDR, R16           ;データ送信(送信開始)
           RET                          ;呼び出し元へ復帰
```

#### C言語プログラム例

```
void USART_Transmit(unsigned int data)
{
    while ( !(UCSRA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UCSRB &= ~(1<<TXB8);              /* TXB8を0に仮設定 */
    if (data & 0x0100) UCSRB |= (1<<TXB8); /* 第9ビットをR17からTXB8へ複写 */
    UDR = data;                          /* データ送信(送信開始) */
}
```

**注:** これらの送信関数は一般的な機能で書かれています。UCSRBの内容が静的(換言すると、UCSRBのTXB8ビットが初期化後に使われるだけ)ならば最適化できます。

5頁の「コード例」をご覧ください。

第9ビットは複数プロセッサ通信使用時のアドレス フレーム識別、また例えば同期として扱う他の規約で使うことができます。

### 15.6.3. 送信フラグと割り込み

USART送信部には状態を示す2つのフラグ、USARTデータレジスタ空き(UDRE)と送信完了(TXC)があります。両フラグは割り込みを発生するのに使えます。

USARTデータレジスタ空き(UDRE)フラグは送信緩衝部が新規データを受け取る準備ができていないかどうかを示します。このビットは送信緩衝部が空の時に設定(1)され、送信緩衝部が移動レジスタに未だ移動されてしまっていない送信されるべきデータを含む時に解除(0)されます。将来のデバイスとの共通性のため、常にこのビットに0を書いてください。

USART制御/状態レジスタB(UCSRB)でデータレジスタ空き割り込み許可(UDRIE)ビットが1を書かれると、(全割り込みが許可されているなら)UDREフラグが設定(1)されている限り、USARTデータレジスタ空き割り込みが実行されます。UDREはUSARTデータレジスタ(UDR)書き込みによって解除(0)されます。割り込み駆動データ送信が使われる時に、データレジスタ空き割り込みルーチンはUDREを解除(0)するために新規データをUDRに書くか、データレジスタ空き割り込みを禁止するかのどちらかを行わなければならない、さもなければ一旦割り込みルーチンを終了しても新しい割り込みが(継続的に)起こります。

送信完了(TXC)フラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部に新規データが現在存在しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行される時に自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)できます。TXCフラグは送信応用プログラムが送信完了後、直ちに通信回線を開放し、受信動作へ移行しなければならない(RS485規格のような)半二重(ハーフデュプレックス)通信インターフェースで有用です。

UCSRBで送信完了割り込み許可(TXCIE)ビットが設定(1)され、(全割り込みが許可されていれば)TXCフラグが設定(1)になる時にUSART送信完了割り込みが実行されます。送信完了割り込み使用時、割り込み実行時にこれが自動的に行われるため、割り込み処理ルーチンはTXCフラグを解除(0)しなくても構いません。

### 15.6.4. パリティ発生器

パリティ発生器は直列フレームデータに対するパリティビットを計算します。パリティビットが許可されると(UPM1=1)、送信部制御論理回路は送られているフレームの最終データビットと最初の停止ビット間にパリティビットを挿入します。

### 15.6.5. 送信の禁止

UCSRBのUSART送信許可(TXEN)ビットの解除(0)は送信部を禁止しますが、その変更は進行中と保留中のどの送信も完了される、換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータが全く無いのに先立って有効にはなりません。禁止されると、送信部はもはやTXDピンの(標準ピン機能)を無効にしません。

## 15.7. データ受信 - USART受信部

USART受信部は受信許可(RXEN)ビット(112頁の「UCSRB - USART制御/状態レジスタB」をご覧ください。)に(論理1を)書くことによって許可されます。受信部が許可されると、RXDピンの標準動作はUSARTによって無視され、受信部の直列入力としての機能を与えられます。何か直列受信が行われ得る前に一度はボーレート、フレーム形式、動作種別が設定されなければなりません。同期動作が使われる場合、XCKピンのクロックは転送クロックとして使われます。

### 15.7.1. 5~8ビットデータフレーム受信

受信部は有効な開始ビットを検出する時にデータ受信を開始します。開始ビットに続く各ビットはボーレートまたはXCKクロックで採取され、その後フレームの最初の停止ビットが受信されるまで受信移動レジスタ内へ移動されます。第2停止ビットは受信部によって無視されます。最初の停止ビットが受信される、換言すると、受信移動レジスタに完全なフレームが存在すると、その内容が受信緩衝部内へ移されます。受信緩衝部はUSARTデータレジスタ(UDR)を読むことによって読めます。8ビット未満のフレームを使う時にUDRから読むデータの上位ビットは0で覆われます。

次のコード例はUSART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグのポーリングを基準とした、簡単なUSART受信関数を示します。この関数が使われ得る前にUSARTが初期化されなければなりません。

#### アセンブリ言語プログラム例

```
USART_Rx:  SBIS    UCSRA, RXC          ;受信完了でスキップ
           RJMP    USART_Rx         ;受信完了待機
;
           IN     R16, UDR           ;受信データ取得
           RET                          ;呼び出し元へ復帰
```

#### C言語プログラム例

```
unsigned char USART_Receive(void)
{
    while ( !(UCSRA & (1<<RXC)) ); /* 受信完了待機 */
    return UDR;                    /* 受信データ取得 */
}
```

注: 5頁の「コード例」をご覧ください。

この関数は緩衝部を読んで値を戻す前に、RXCフラグの検査によって受信緩衝部にデータが存在するのを単純に待ちます。

## 15.7.2. 9ビットデータフレーム受信

9ビットデータが使われる場合、USARTデータレジスタ(UDR)から下位バイトを読むのに先立って第9ビットがUSART制御/状態レジスタB(UCSRB)の受信データビット8(RXB8)ビットから読まれなければなりません。この規則はフレーミング異常(FE)、オーバーラン発生(DOR)、パリティ誤り(UPE)状態フラグにも適用されます。UDR読み込みが受信緩衝部FIFOの状態、その結果、RXB8,FE,DOR,UPEビットの状態を変更する(切り替える)ため、UDRからのデータに先立って状態ビットが読まれなければなりません。

次のコード例は9ビットデータと状態ビット両方を扱う簡単なUSART受信関数を示します。

### アセンブリ言語プログラム例

```
USART_Rx: SBIS UCSRA, RXC ;受信完了でスキップ
          RJMP USART_Rx ;受信完了待機
;
          IN R18, UCSRA ;状態フラグ取得
          IN R17, UCSRB ;受信第9ビット取得
          IN R16, UDR ;受信データ取得
          ANDI R18, (1<<FE) | (1<<DOR) | (1<<UPE) ;受信異常検査
          BREQ USART_Rx_V ;異常なしで分岐
;
          LDI R17, -1 ;異常で-1値設定
          LDI R16, -1 ;
USART_Rx_V: LSR R17 ;RXB8ビットをビット0位置へ移動
          ANDI R17, $01 ;RXB8ビットのみ有効
          RET ;呼び出し元へ復帰
```

### C言語プログラム例

```
unsigned int USART_Receive(void)
{
    unsigned char status, resh, resl; /* 一時変数定義 */
    while ( !(UCSRA & (1<<RXC)) ); /* 受信完了待機 */
    status = UCSRA; /* 状態フラグ取得 */
    resh = UCSRB; /* 受信第9ビット取得 */
    resl = UDR; /* 受信データ取得 */
    if ( status & ((1<<FE) | (1<<DOR) | (1<<UPE)) ) return -1; /* 受信異常で-1値設定/復帰 */
    resh = (resh>>1) & 0x01; /* RXB8ビットのみ有効最下位へ */
    return ((resh<<8) | resl); /* 結果9ビットデータ取得/復帰 */
}
```

**注:** 5頁の「コード例」をご覧ください。

この受信関数例は何か評価を行う前に、全てのI/Oレジスタをレジスタファイルに読みます。これは読まれた緩衝部位置が可能な限り早く新規データを自由に受け入れられるため、最適な受信緩衝部利用になります。

## 15.7.3. 受信完了フラグと割り込み

USART受信部には受信部の状態を示す1つのフラグがあります。

USART受信完了(RXC)フラグは未読データが受信緩衝部に存在するかを示します。このフラグは受信緩衝部に未読データが存在する時に設定(1)され、受信緩衝部が空の(換言すると、何も未読データを含まない)時に解除(0)されます。受信部が禁止される場合(RXEN=0)、受信緩衝部が破棄され、その結果としてRXCフラグは0になります。

USART制御/状態レジスタB(UCSRB)でUSART受信完了割り込み許可(RXCIE)ビットが設定(1)されると、(全割り込みが許可されていれば)RXCフラグが設定(1)されている限り、USART受信完了割り込みが実行されます。割り込み駆動データ受信が使われる時に、受信完了割り込みルーチンはRXCフラグを解除(0)するためにUSARTデータレジスタ(UDR)から受信したデータを読まなければならない、さもなければ一旦割り込みルーチンを終了しても新しい割り込みが(継続的に)起きます。



## 15.7.4. 受信異常フラグ

USART受信にはフレーミング異常(FE)、データオーバーラン発生(DOR)、パリティ誤り(UPE)の3つの異常フラグがあります。全ての以上フラグはそれらが異常状態を示すフレームと共に受信緩衝部に置かれ、それらはUSART制御/状態レジスタ(UCSRA)経由で読めます。異常フラグの緩衝のため、USARTデータレジスタ(UDR)読み込みが緩衝部を変更するので、それらは受信緩衝部(UDR)の前に読まなければならないなりません。

異常フラグはソフトウェアによって変更できませんが、将来のUSART(機能)実装の上位互換性のため、UCSRAが書かれる時に全てのフラグは解除(0)されなければならないなりません。異常フラグはどれも割り込みを生成できません。

- **フレーミング異常(FE)フラグ**は受信緩衝部に格納された、次に読み込み可能なフレームの第1停止ビットの状態を示します。このフラグは停止ビットが正しく(Highとして)読まれた時に0で、停止ビットが不正(Low)だった時にこのフラグは1です。このフラグは同期外れ状態の検出、中断状態検出、規約操作に使えます。受信部が最初(第1)以外の全停止ビットを無視するため、このフラグはUSART制御/状態レジスタC(UCSRC)の停止ビット選択(USBS)ビット設定によって影響を及ぼされません。将来のデバイスとの共通性のため、このビットはUCSRA書き込み時、常に解除(0)しなければならないなりません。
- **データオーバーラン発生(DOR)フラグ**は受信緩衝部が一杯状態のためのデータ消失を示します。データオーバーランの状況は受信緩衝部(2フレーム)が一杯で、受信移動レジスタで待っている(次の)新規フレームデータがあり、(更に次の)新規開始ビットが検出される時に起きます。このフラグが設定(1)なら、最後にUDRから読んだフレームと次にUDRから読むフレーム間で1つ以上の直列フレーム損失があります。将来のデバイスとの共通性のため、このビットはUCSRA書き込み時、常に解除(0)しなければならないなりません。このフラグは受信されたデータが移動レジスタから受信緩衝部へ正常に移動された時に解除(0)されます。
- **パリティ誤り(UPE)フラグ**は受信緩衝部内の次のフレームがパリティ誤りを持つことを示します。パリティ検査が許可されていない場合、このフラグは常に0です。将来のデバイスとの共通性のため、このビットはUCSRA書き込み時、常に解除(0)しなければならないなりません。より多くの詳細については100頁の「パリティビット計算」と次の「パリティ検査器」をご覧ください。

## 15.7.5. パリティ検査器

パリティ検査器はパリティ種別上位ビット(UPM1)が設定(1)されると活性(有効)になります。実行されるべきパリティ検査の形式(偶数または奇数)はUPM0ビットによって選択されます。許可されると、パリティ検査器は到着フレーム内のデータビットのパリティを計算し、その結果と(受信)直列フレーム内のパリティビットを比較します。検査の結果は受信データ、停止ビットと共に受信緩衝部に格納されます。その後パリティ誤り(UPE)フラグはフレームにパリティ異常があるかを検査するため、ソフトウェアによって読むことができます。

パリティ検査が許可される場合、受信緩衝部から読むことができる次のフレームが受信時にパリティ異常を持つなら、UPEフラグが設定(1)されます。このビットはUSARTデータレジスタ(UDR)が読まれるまで有効です。

## 15.7.6. 受信の禁止

送信部と違い、受信部は直ちに禁止され、受信進行中のどのデータも失われます。禁止(USART制御/状態レジスタB(UCSRB)のUSART受信許可(RXEN)ビット=0)されると、受信部はもはやRXDポートピンの標準機能を無効にせず、FIFO緩衝部が破棄され、緩衝部内の残データが失われます。

## 15.7.7. 受信緩衝部の破棄

受信緩衝部FIFOは受信が禁止されると破棄、換言すると、緩衝部は内容を空にされます。未読データは失われます。例えば異常状態のため、通常動作中に緩衝部を破棄するには、USART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグが解除(0)されるまでUSARTデータレジスタ(UDR)を読んでください。

次のコード例は受信緩衝部の破棄方法を示します

### アセンブリ言語プログラム例

```
USART_Flush: SBIS    UCSRA, RXC                ;未読データありでスキップ
              RET                                ;未読データなしで復帰
;
              IN     R16, UDR                  ;データ受信
              RJMP   USART_Flush              ;未読データなしまで継続
```

### C言語プログラム例

```
void USART_Flush(void)
{
    unsigned char dummy;                /* 一時変数定義 */
    while ( UCSRA & (1<<RXC) ) dummy=UDR; /* 未読データ読み捨て */
}
```

注: 5頁の「コード例」をご覧ください。

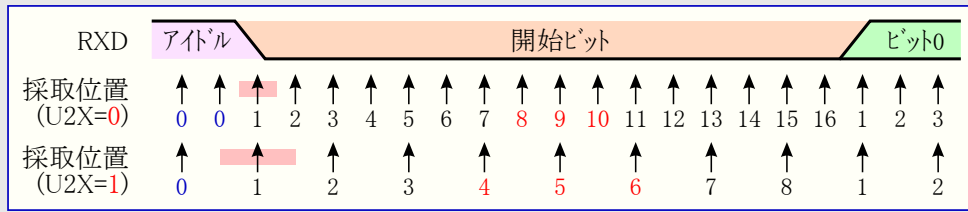
## 15.8. 非同期受信

USARTは非同期データ受信を扱うためのクロック再生とデータ再生部を含みます。クロック再生論理回路は内部的に生成したボーレートクロックをRXDピンに到着する非同期直列フレームに同期化するのに使われます。データ再生論理回路は到着ビット毎に低域通過の濾波と採取をし、それによって受信部の雑音耐性を改善します。非同期受信動作範囲は内部ボーレートクロック精度、到着フレーム速度、フレーム長のビット数に依存します。

### 15.8.1. 非同期クロック再生

クロック再生論理回路は内部クロックを到着直列フレームに同期化します。図60.は到着フレームの開始ビットの採取手順を図解します。採取速度は標準速動作でボーレートの16倍、倍速動作で8倍です。赤い帯(訳注:原文は水平矢印)は採取処理のための同期変量を図示します。倍速動作(U2X=1)を使う時の広い変量時間に注意してください。採取番号0はRXD信号がアイドル(換言すると、通信の動きなし)の時に実行される採取です。

図60. 開始ビットの採取

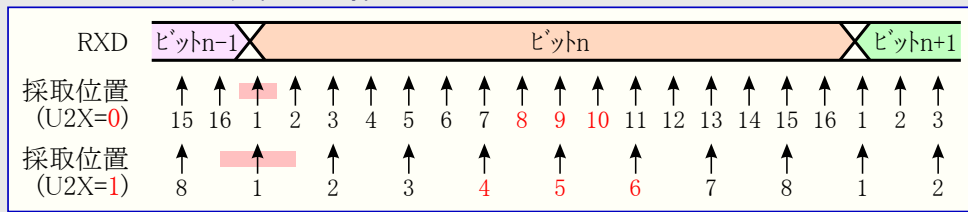


クロック再生論理回路がRXD信号線でHigh(アイドル)からLow(開始)の遷移を検出すると、開始ビット検出手順が開始されます。図60.で採取は赤数字(訳注:原文は箱枠内)の番号で示され、採取1は最初のLow採取を意味します。その後にクロック再生回路は有効な開始ビットが受信されるかを決定するのに、(標準速動作で)採取8,9,10、(倍速動作で)採取4,5,6を使います。これら3回で2回以上の採取が論理Highレベルの場合(多数決)、この開始ビットは尖頭雑音として捨てられ、受信部は次のHighからLowの遷移を探し始めます。けれども有効な開始ビットが検出された場合、クロック再生論理回路は同期化され、データ再生が開始されます。この同期化手順は各開始ビット毎に繰り返されます。

### 15.8.2. 非同期データ再生

受信部クロックが開始ビットに同期化されるとデータ再生が始まります。データ再生部は標準速動作で16段、倍速動作で8段の順列回路を使います。図61.はデータビットとパリティビットの採取を示します。各採取は再生部の各段に等しい番号を与えられます。

図61. データビットとパリティビットの採取



受信したビットの論理値の決定は受信したビット中央の3採取で論理値の多数決を取るによって行われます。図に於ける中央の3採取は赤数字(訳注:原文は箱枠内)の採取番号によって強調されます。多数決の手順は次のように行われます。2または全3採取がHighレベルならば受信したビットは論理1が記録されます。2または全3採取がLowレベルならば受信したビットは論理0が記録されます。この多数決手順はRXDピンの到着信号に対して低域通過濾波器(ローパスフィルタ)のように働きます。この再生手順はその後に完全なフレームが受信されるまで繰り返されます。これには最初の(第1)停止ビットを含みます。受信部がフレームの最初の停止ビットだけを使うことに注意してください。

図62.は停止ビットの採取と次フレームの最も早い開始ビット開始の可能性を示します。

図62. 停止ビットの採取と次の開始ビットの採取



停止ビットはフレーム内の他のビットと同じ多数決に従わなければなりません。停止ビットが論理0値と記録されると、フレーミング異常(FE)フラグが設定(1)されます。

新規フレームの開始ビットを示すHighからLowへの遷移は多数決に使った最後のビット後に実現できます。標準速動作に於ける最初のLowレベル採取は図62.のA点で有り得ます。倍速動作に於ける最初のLowレベルはB点に遅れます。C点は完全な長さの停止ビット(の終点)を示します。この早い開始ビット検出は受信部の動作範囲に影響します。

### 15.8.3. 非同期での動作範囲

受信部の動作範囲は受信したビット速度と内部的に生成したポーレート間の不一致に依存します。送信部が速すぎるまたは遅すぎるビット速度でフレームを送出したり、内部的に発生した受信部のポーレートが類似した基準周波数(表51.参照)を持たない場合、受信部は開始ビットでフレームを同期できません。

次式は到着データ速度と内部受信部ポーレート間の比率計算に役立ちます。

$$R_{slow} = \frac{(D+1) \times S}{S-1+D \times S+S_F} \quad R_{fast} = \frac{(D+2) \times S}{(D+1) \times S+S_M}$$

D : データとパリティのビット数 (5~10)  
 S : ビットあたりの採取数 (標準速=16、倍速=8)  
 S<sub>F</sub> : 多数決に使う最初の採取番号 (標準速=8、倍速=4)  
 S<sub>M</sub> : 多数決に使う中心の採取番号 (標準速=9、倍速=5)  
 R<sub>slow</sub> : は受信側ポーレートに対して許容できる最低受信ビット速度の比率です。  
 R<sub>fast</sub> : は受信側ポーレートに対して許容できる最高受信ビット速度の比率です。

表51.は許容できる最大受信部ポーレート誤差の一覧です。標準速動作には、より高いポーレート変動許容力があることに注目してください。

表51. 標準速と倍速での受信部ポーレート推奨最大許容誤差

D	標準速動作 (U2X=0)				倍速動作 (U2X=1)			
	R <sub>slow</sub> (%)	R <sub>fast</sub> (%)	総合許容誤差(%)	推奨許容誤差(%)	R <sub>slow</sub> (%)	R <sub>fast</sub> (%)	総合許容誤差(%)	推奨許容誤差(%)
5	93.20	106.67	-6.80~+6.67	±3.0	94.12	105.66	-5.88~+5.66	±2.5
6	94.12	105.79	-5.88~+5.79	±2.5	94.92	104.92	-5.08~+4.92	±2.0
7	94.81	105.11	-5.19~+5.11	±2.0	95.52	104.35	-4.48~+4.35	±1.5
8	95.36	104.58	-4.54~+4.58	±2.0	96.00	103.90	-4.00~+3.90	±1.5
9	95.81	104.14	-4.19~+4.14	±1.5	96.39	103.53	-3.61~+3.53	±1.5
10	96.17	103.78	-3.83~+3.78	±1.5	96.70	103.23	-3.30~+3.23	±1.0

注: Dはデータビット数とパリティビットの合計ビット数です。

(訳注) 原書は表61.に標準速、表62.に倍速を記載していますが、比較が容易なように表51.として纏めました。

受信部ポーレートの推奨最大許容誤差は最大総合許容誤差を送信部と受信部で等分割する仮定で作られています。

受信部ポーレート誤差に対して2つの起こり得る原因があります。

- 受信部のシステムクロックは供給電圧範囲と温度範囲に関して常に若干の不安定性があります。
- 2つ目の誤差元はより制御可能です。ポーレート発振器は欲したポーレートを取得するためにシステム周波数の正確な分周を常に行うことはできません。この場合、可能ならば受け入れ可能な低い誤差を与えるUBRR値が使われるべきです。

### 15.8.4. フレーム開始検出

USARTフレーム開始検出器は開始ビット検出時にMCUを起こすことができます。23頁の表9.をご覧ください。

RXDピンでHighからLowへの遷移が検出されると、内部8MHz発振器が給電されUSARTクロックが許可されます。ポーレートが内部8MHz発振器始動時間に関して充分遅ければ、始動後にデータフレームの残りを受信することができます。内部8MHz発振器の始動時間は供給電圧と温度で変化します。

USARTフレーム開始検出は非同期と同期の両動作形態で動きます。これは「UCSRD – USART制御/状態レジスタ」でフレーム開始検出許可(SFDE)ビットを(1)に書くことによって許可されます。USART開始割り込み許可(RXSIE)ビットが設定(1)されるなら、開始検出時直ちにUSART受信開始割り込みが生成されます。

開始割り込みなしでこの機能使用時、開始検出論理回路は内部8MHz発振器とUSARTクロックを活性化し、同時にフレームが受信されつつあるだけです。他のクロックは受信完了割り込みがMCUを起こすまで停止されたままです。

より多くの詳細については113頁の「UCSRD – USART制御/状態レジスタ」をご覧ください。

## 15.9. 複数プロセッサ通信動作

USART制御/状態レジスタA(UCSRA)での複数プロセッサ通信動作(MPCM)ビットの設定(1)はUSART受信部によって受信された到着フレームの選別機能を許可します。アドレス情報を含まないフレームは無視され、受信緩衝部に格納されません。これは同一直列バス経由で通信する複数MCUのシステムで、CPUによって扱われなければならない到着フレーム数を効果的に減らします。送信部はMPCMビット設定によって影響されませんが、複数プロセッサ通信動作を利用するシステムの一部の時は違うふうに使われなければなりません。

受信部が5~8データビットを含むフレームを受信するように設定されるなら、最初の停止ビットはデータまたはアドレス情報を含むフレームかどうかを示します。受信部が9データビットのフレームに設定されるなら、USART制御/状態レジスタB(UCSRB)の受信第9(RXB8)ビットがアドレスとデータのフレームを識別するのに使われます。フレーム種別(最初の停止または第9)ビットが1の時にフレームはアドレスを含みます。フレーム種別ビットが0の時にそのフレームはデータフレームです。

複数プロセッサ通信動作は主MCUからのデータを多くの従MCUで受信することを可能にします。これはどのMCUがアドレス指定されるかを検出するため、最初にアドレスフレームを調べることによって行われます。特定の従MCUがアドレス指定されたなら、そのMCUは後続するデータフレームを通常のように受信し、一方その他の従MCUは他のアドレスフレームが受信されるまで受信したフレームを無視します。

### 15.9.1. 複数プロセッサ通信動作(MPCM)の使い方

主MCUとして動作するMCUについては9ビットデータフレーム形式を使えます。UCSRBの送信第9(TXB8)ビットはアドレスフレーム時に設定(1)、またはデータフレーム時に解除(0)されて送信されなければなりません。この場合、従MCUは9ビットデータフレーム形式の使用に設定されなければなりません。

複数プロセッサ通信動作でデータを交換するのに次の手順が使われるべきです。

1. 全ての従MCUは複数プロセッサ通信動作です(UCSRAの複数プロセッサ通信動作(MPCM)ビットが設定(1))。
2. 主MCUはアドレスフレームを送り、全ての従装置がこのフレームを受信し、これを読みます。従CPUでは通常のようにUCSRAで受信完了(RXC)フラグが設定(1)されます。
3. 各従MCUはUSARTデータレジスタ(UDR)を読み、選択されたかを判定します。選択された場合はUCSRAのMPCMビットを解除(0)し、そうでなければ(非選択の場合は)MPCMビット設定を保ち、次のアドレスフレームを待ちます。
4. アドレス指定されたMCUは新規アドレスフレームが受信されるまで全データフレームを受信します。MPCMビットが未だ設定(1)されている他の従CPUはこのデータフレームを無視します。
5. 最後のデータフレームがアドレス指定されたMCUによって受信されると、アドレス指定されたMCUはMPCMビットを設定(1)し、主装置からの新規アドレスフレームを待ちます。以降、手順は2.からを繰り返します。

5~8ビットデータフレーム形式のどの使用も可能ですが、受信側が使うnとn+1ビットデータフレーム形式間を切り替えなければならないため非実用的です。これは送信部と受信部が同じデータビット長設定を使うため、全二重(フルデュプレックス)動作を困難にします。5~8ビットデータフレームが使われる場合、最初の停止ビットがフレーム種別を示すのに使われるので、送信部は2停止ビット使用に設定されなければなりません。

MPCMビットを設定(1)または解除(0)するのに読み-修正-書き(リード モデファイライト)命令(SBIとCBI)を使ってはいけません。MPCMビットは送信完了(TXC)フラグと同じI/O位置を共用しており、SBIまたはCBI命令を使うと偶然に解除(0)されるかもしれません。

(訳注) ATtiny828ではUCSRAのI/OアドレスがSBI,CBI命令適用範囲外なので上記記述は不適切です。但し、命令の組み合わせによる同様処理の場合に対して、上記注意の本意は適切(有効)です。



## 15.10. ホーレート設定例

非同期動作に対して一般的に使われるホーレートは表53.のUBRR設定を使うことによって生成することができます。目的のホーレートに対して設定誤差1.5%(標準速:U2X=0)、1.0%(倍速:U2X=1)以上を赤字で示します(訳注:原書の本文は0.5%未満を太字)。より高い誤差率でも受け入れ可能ですが、特に長い直列フレーム(107頁の「非同期での動作範囲」参照)で誤差率が高いと、受信部は雑音耐性が低下します。誤差率は次式を使って計算されます。

$$\text{誤差率(\%)} = \left( \frac{\text{UBRR設定ホーレート(最近似値)}}{\text{目的のホーレート}} - 1 \right) \times 100(\%)$$

表53. Xtal、ホーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ホーレート (bps)	1MHz				1.8432MHz				2MHz				2.4576MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	51	0.2	103	0.2	95	0.0	191	0.0	103	0.2	207	0.2	127	0.0	255	0.0
2400	25	0.2	51	0.2	47	0.0	95	0.0	51	0.2	103	0.2	63	0.0	127	0.0
4800	12	0.2	25	0.2	23	0.0	47	0.0	25	0.2	51	0.2	31	0.0	63	0.0
9600	6	-7.0	12	0.2	11	0.0	23	0.0	12	0.2	25	0.2	15	0.0	31	0.0
14400	3	8.5	8	-3.5	7	0.0	15	0.0	8	-3.5	16	2.1	10	-3.0	20	1.6
19200	2	8.5	6	-7.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	0.0	15	0.0
28800	1	8.5	3	8.5	3	0.0	7	0.0	3	8.5	8	-3.5	4	6.7	10	-3.0
38400	1	-18.6	2	8.5	2	0.0	5	0.0	2	8.5	6	-7.0	3	0.0	7	0.0
57600	0	8.5	1	8.5	1	0.0	3	0.0	1	8.5	3	8.5	2	-11.1	4	6.7
76800	0	-18.6	1	-18.6	1	-25.0	2	0.0	1	-18.6	2	8.5	1	0.0	3	0.0
115.2k	-	-	0	8.5	0	0.0	1	0.0	0	8.5	1	8.5	0	33.3	2	-11.1
230.4k	-	-	0	-45.7	0	-50.0	0	0.0	0	-45.7	0	8.5	0	-33.3	0	33.3
250k	-	-	-	-	-	-	0	-7.8	-	-	0	0.0	-	-	0	22.9
最高速	62500		125k		115.2k		230.4k		125k		250k		153.6k		307.2k	

ホーレート (bps)	3.2768MHz				3.6864MHz				4MHz				4.608MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	170	-0.2	340	0.1	191	0.0	384	0.0	207	0.2	416	-0.1	239	0.0	479	0.0
2400	84	0.4	170	-0.2	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0
4800	42	-0.8	84	0.4	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0
9600	20	1.6	42	-0.8	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0
14400	13	1.6	27	1.6	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0
19200	10	-3.0	20	1.6	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0
28800	6	1.6	13	1.6	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0
38400	4	6.7	10	-3.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	-6.3	14	0.0
57600	3	-11.1	6	1.6	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0
76800	2	-11.1	4	6.7	2	0.0	5	0.0	2	8.5	6	-7.0	3	-6.3	7	-6.3
115.2k	1	-11.1	3	-11.1	1	0.0	3	0.0	1	8.5	3	8.5	2	-16.7	4	0.0
230.4k	0	-11.1	1	-11.1	0	0.0	1	0.0	0	8.5	1	8.5	0	25.0	2	-16.7
250k	-	-	1	-18.1	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2
500k	-	-	0	-18.1	-	-	0	-7.8	-	-	0	0.0	0	-42.4	0	15.2
1M	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	-42.4
最高速	204.8k		409.6k		230.4k		460.8k		250k		500k		288k		576k	

注: 最高速はUBRR=0、誤差=0.0%です。以降の周波数は次頁へ続く。

(訳注) 原書では頁割された表毎に表番号63.~66.となっていますが、共通性から纏めて表53.としました。原書に対して数種の発振周波数を追加しました。



表53 (続き). Xtal、ホーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ホーレート (bps)	4.9152MHz				6.144MHz				7.3728MHz				8MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	255	0.0	511	0.0	319	0.0	639	0.0	383	0.0	767	0.0	416	-0.1	832	0.0
2400	127	0.0	255	0.0	159	0.0	319	0.0	191	0.0	383	0.0	207	0.2	416	-0.1
4800	63	0.0	127	0.0	79	0.0	159	0.0	95	0.0	191	0.0	103	0.2	207	0.2
9600	31	0.0	63	0.0	39	0.0	79	0.0	47	0.0	95	0.0	51	0.2	103	0.2
14400	20	1.6	42	-0.8	26	-1.2	52	0.6	31	0.0	63	0.0	34	-0.8	68	0.6
19200	15	0.0	31	0.0	19	0.0	39	0.0	23	0.0	47	0.0	25	0.2	51	0.2
28800	10	-3.0	20	1.6	12	2.6	26	-1.2	15	0.0	31	0.0	16	2.1	34	-0.8
38400	7	0.0	15	0.0	9	0.0	19	0.0	11	0.0	23	0.0	12	0.2	25	0.2
57600	4	6.7	10	-3.0	6	-4.8	12	2.6	7	0.0	15	0.0	8	-3.5	16	2.1
76800	3	0.0	7	0.0	4	0.0	9	0.0	5	0.0	11	0.0	6	-7.0	12	0.2
115.2k	2	-11.1	4	6.7	2	11.1	6	-4.8	3	0.0	7	0.0	3	8.5	8	-3.5
230.4k	1	33.3	2	-11.1	1	-16.7	2	11.1	1	0.0	3	0.0	1	8.5	3	8.5
250k	1	22.9	2	-18.1	1	-23.2	2	2.4	1	-7.8	3	0.0	1	0.0	3	0.0
500k	0	-38.6	0	22.9	0	-23.2	1	-23.2	0	-7.8	1	-7.8	0	0.0	1	0.0
1M	-	-	0	-38.6	-	-	0	-23.2	-	-	0	-7.8	-	-	0	0.0
最高速	307.2k		614.4k		384k		768k		460.8k		921.6k		500k		1M	

ホーレート (bps)	9.216MHz				9.8304MHz				10MHz				11.0592MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	479	0.0	959	0.0	511	0.0	1023	0.0	520	-0.0	1041	-0.0	575	0.0	1151	0.0
2400	239	0.0	479	0.0	255	0.0	511	0.0	259	0.2	520	-0.0	287	0.0	575	0.0
4800	119	0.0	239	0.0	127	0.0	255	0.0	129	0.2	259	0.2	143	0.0	287	0.0
9600	59	0.0	119	0.0	63	0.0	127	0.0	64	0.2	129	0.2	71	0.0	143	0.0
14400	39	0.0	79	0.0	42	-0.8	84	0.4	42	0.9	86	-0.2	47	0.0	95	0.0
19200	29	0.0	59	0.0	31	0.0	63	0.0	32	-1.4	64	0.2	35	0.0	71	0.0
28800	19	0.0	39	0.0	20	1.6	42	-0.8	21	-1.4	42	0.9	23	0.0	47	0.0
38400	14	0.0	29	0.0	15	0.0	31	0.0	15	1.8	32	-1.4	17	0.0	35	0.0
57600	9	0.0	19	0.0	10	-3.0	20	1.6	10	-1.4	21	-1.4	11	0.0	23	0.0
76800	7	6.7	14	0.0	7	0.0	15	0.0	7	1.8	15	1.8	8	0.0	17	0.0
115.2k	4	0.0	9	0.0	4	6.7	10	-3.0	4	8.5	10	-1.4	5	0.0	11	0.0
230.4k	2	-16.7	4	0.0	2	-11.1	4	6.7	2	-9.6	4	8.5	2	0.0	5	0.0
250k	1	15.2	4	-7.8	1	-23.2	4	-1.7	2	-16.7	4	0.0	2	-7.8	5	-7.8
500k	0	15.2	1	15.2	0	22.9	1	22.9	0	25.0	2	-16.7	0	38.2	2	-7.8
1M	0	-42.4	0	15.2	0	-38.6	0	22.9	0	-37.5	0	25.0	0	-30.9	0	38.2
最高速	576k		1.152M		614.4k		1.2288M		625k		1.25M		691.2k		1.3824M	

ホーレート (bps)	14.7456MHz				16MHz				18.4320MHz				20MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	767	0.0	1533	0.0	832	0.0	1666	0.0	959	0.0	1919	0.0	1041	0.0	2082	0.0
2400	383	0.0	767	0.0	416	-0.1	832	0.0	479	0.0	959	0.0	520	0.0	1041	0.0
4800	191	0.0	383	0.0	207	0.2	416	-0.1	239	0.0	479	0.0	259	0.2	520	0.0
9600	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0	129	0.2	259	0.2
14400	63	0.0	127	0.0	68	0.6	138	-0.1	79	0.0	159	0.0	86	-0.2	173	-0.2
19200	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0	64	0.2	129	0.2
28800	31	0.0	63	0.0	34	-0.8	68	0.6	39	0.0	79	0.0	42	0.9	86	-0.2
38400	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0	32	-1.4	64	0.2
57600	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0	21	-1.4	42	0.9
76800	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0	15	1.7	32	-1.4
115.2k	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0	10	-1.4	21	-1.4
230.4k	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0	4	8.5	10	-1.4
250k	3	-7.8	6	5.3	3	0.0	7	0.0	4	-7.8	8	2.4	4	0.0	9	0.0
500k	1	-7.8	3	-7.8	1	0.0	3	0.0	1	15.2	4	-7.8	1	25.0	4	0.0
1M	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2	0	25.0	1	25.0
最高速	921.6k		1.8432M		1M		2M		1.152M		2.304M		1.25M		2.5M	

注: 最高速はUBRR=0、誤差=0.0%です。

## 15.11. USART用レジスタ

### 15.11.1. UDR – USARTデータレジスタ (USART I/O Data Register)

ビット (\$C6)	7	6	5	4	3	2	1	0	
	R/TXB7	R/TXB6	R/TXB5	R/TXB4	R/TXB3	R/TXB2	R/TXB1	R/TXB0	UDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

同じI/Oアドレスを共用するUSART受信データ緩衝とUSART送信データ緩衝のレジスタはUSARTデータレジスタまたはUDRとして参照されます。UDRへのデータ書き込みは送信データ緩衝レジスタ(TXB)へ行き、UDR読み込みは受信データ緩衝レジスタ(RXB)の内容を返します。

5～7ビットデータに対する上位未使用ビットは送信部によって無視され、受信部によって0に設定されます。

送信緩衝部はUCSRAで送信データレジスタ空き(UDRE)フラグが設定(1)される時にだけ書くことができます。UDREフラグが設定(1)されない時にUDRへ書かれたデータは無視されます。送信部が許可されて送信緩衝部にデータが書かれると、送信部は送信移動レジスタが空きの時にそこへデータを設定(移動)します。その後データはTXDピンで直列送信されます。

受信緩衝部は2段のFIFOから成ります。このFIFOは受信緩衝部がアクセスされると必ず状態を切り替えます。この受信緩衝部の動きのため、この位置をアクセスするのに読み-修正-書き(リード モデファイライト)命令(SBIとCBI)が使われるべきではありません。ビット検査(SBICとSBIS)命令もFIFOの状態を換えるので、これらの命令を使う時にも注意されるべきです(訳注:本レジスタがこれらの命令の適用可能アドレス範囲外のため、この注意は不適切ですが、複数命令で同様動作を行う場合は注意が必要です)。

### 15.11.2. UCSRA – USART制御/状態レジスタA (USART Control and Status Register A)

ビット (\$C0)	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
初期値	0	0	1	0	0	0	0	0	

#### ● ビット7 – RXC : USART受信完了フラグ (USART Receive Complete)

このフラグは受信緩衝部に未読データがある時に設定(1)され、受信緩衝部が空の(換言すると、どんな未読データも含まない)時に解除(0)されます。受信部が禁止されると、受信緩衝部が破棄され、その結果RXCフラグは0になります。このフラグは受信完了割り込みを発生するのに使うことができます(USART制御/状態レジスタB(UCSRB)の受信完了割り込み許可(RXCIE)ビットをご覧ください)。

#### ● ビット6 – TXC : USART送信完了フラグ (USART Transmit Complete)

このフラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部(UDR)に新規データが現存しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行される時に自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)することができます。このフラグは送信完了割り込みを発生することができます(UCSRBの送信完了割り込み許可(TXCIE)ビットをご覧ください)。

#### ● ビット5 – UDRE : USART送信データレジスタ空きフラグ (USART Data Register Empty)

このフラグは送信緩衝部(UDR)が新規データを受け取り準備可を示します。UDREが1ならば緩衝部は空で、従って書かれる準備可です。UDREフラグは送信データレジスタ空き割り込みを発生することができます(UCSRBの送信データレジスタ空き割り込み許可(UDRIE)ビットをご覧ください)。

送信部が準備可であることを示すため、UDREはリセット後に設定(1)されます。

#### ● ビット4 – FE : フレーミング異常フラグ (Framing Error)

このフラグは受信緩衝部内の次データが受信時にフレーミング異常の場合(換言すると、受信緩衝部内で次データの最初の(第1)停止ビットがLowの時に)設定(1)されます。このフラグは受信緩衝部(UDR)が読まれるまで有効です。受信したデータの停止ビットがHighの時にFEフラグは0です。

レジスタ書き込み時に常にこのビットを0に設定してください。

#### ● ビット3 – DOR : データオーバーラン発生フラグ (Data OverRun)

このビットはオーバーラン状態が検出される場合に設定(1)されます。データオーバーランは受信緩衝部(2フレーム分)が一杯で、受信移動レジスタ内で新規フレームが待機中に新規開始ビットが検出される時に起きます。このビットは受信緩衝部(UDR)が読まれるまで有効です。

レジスタ書き込み時に常にこのビットを0に設定してください。

#### ● ビット2 – UPE : パリティ誤りフラグ (USART Parity Error)

このビットは受信緩衝部内の次データが受信時にパリティ異常があり、その時点でパリティ検査が許可されていた(UPM1=1)場合に設定(1)されます。このビットは受信緩衝部(UDR)が読まれるまで有効です。

レジスタ書き込み時に常にこのビットを0に設定してください。

#### ● ビット1 – U2X : 倍速許可 (Double the USART Transmission Speed)

このビットは非同期動作でだけ有効です。同期動作使用時、このビットに0を書いてください。

このビットの1書き込みはボーレート分周器の分周値を16から8に減らし、事実上、非同期通信に対して転送速度を倍にします。

● **ビット0 – MPCM : 複数プロセッサ通信動作 (Multi-Processor Communication Mode)**

このビットは複数プロセッサ通信動作を許可します。このビットが1を書かれると、USART受信部によって受信したアドレス情報を含まない全到着フレームが無視されます。送信部はMPCMによって影響を及ぼされません。より多くの詳細情報については108頁の「複数プロセッサ通信動作」をご覧ください。

**15.11.3. UCSRB – USART制御/状態レジスタB (USART Control and Status Register B)**

ビット (\$C1)	7	6	5	4	3	2	1	0	UCSRB
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

● **ビット7 – RXCIE : 受信完了割り込み許可 (Receive Complete Interrupt Enable)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグでの割り込みを許可します。

USART受信完了割り込みはRXCIEビット、ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUCSRAのRXCフラグが設定(1)される場合にだけ生成されます。

● **ビット6 – TXCIE : 送信完了割り込み許可 (Transmit Complete Interrupt Enable)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の送信完了(TXC)フラグでの割り込みを許可します。

USART送信完了割り込みはTXCIEビット、SREGの全割り込み許可(I)ビットとUCSRAのTXCフラグが設定(1)される場合にだけ生成されます。

● **ビット5 – UDRIE : 送信データレジスタ空き割り込み許可 (Transmit Data Register Empty Interrupt Enable)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグでの割り込みを許可します。USART送信データレジスタ空き割り込みはUDRIEビット、SREGの全割り込み許可(I)ビットとUCSRAのUDREフラグが設定(1)される場合にだけ生成されます。

● **ビット4 – RXEN : 受信許可 (Receiver Enable)**

このビットへの1書き込みはUSART受信(部)を許可します。許可されると、受信部はRXDピンに対する標準ポート動作を無効にします。受信禁止は受信緩衝部を破棄し、フレーミング異常(FE)、オーバラン(DOR)、パリティ誤り(UPE)のフラグを無効にします。

● **ビット3 – TXEN : 送信許可 (Transmitter Enable)**

このビットへの1書き込みはUSART送信(部)を許可します。許可されると、送信部はTXDピンに対する標準ポート動作を無効にします。送信禁止(TXENの0書き込み)は進行中と保留中の送信が完了される、換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない時まで有効になりません。禁止されると、送信部はもはやTXDポート(の標準I/O機能)を無効にしません。

● **ビット2 – UCSZ2 : データビット長選択2 (Character Size)**

USART制御/状態レジスタC(UCSRC)のUCSZ1,0ビットと組み合わせたUCSZ2ビットは送信部と受信部で使うフレームのデータビット数(Character size)を設定します。

● **ビット1 – RXB8 : 受信データビット8 (Receive Data Bit 8)**

RXB8は9ビットデータでの直列フレーム操作時に受信したフレームの第9データビット(ビット8)です。UDRから下位ビットを読む前に読まれなければなりません。

● **ビット0 – TXB8 : 送信データビット8 (Transmit Data Bit 8)**

TXB8は9ビットデータでの直列フレーム操作時に送信されるべきデータの第9データビット(ビット8)です。UDRへ下位ビットを書く前に書かれなければなりません。

**15.11.4. UCSRC – USART制御/状態レジスタC (USART Control and Status Register C)**

ビット (\$C2)	7	6	5	4	3	2	1	0	UCSRC
	UMSEL1	UMSEL0	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	
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	1	1	0	

● **ビット7,6 – UMSEL1,0 : USART動作選択 (USART Mode Select)**

これらのビットは表57.で示されるようにUSART動作形態を選びます。

表57. UMSELビット設定

UMSEL1	UMSEL0	動作種別
0	0	非同期USART
	1	同期USART
1	0	(予約)
	1	主装置SPI (MSPIM) (注)

注: 主装置SPI動作(MSPIM)操作の完全な記述については115頁の「USARTでのSPI動作」をご覧ください。

● **ビット5,4 – UPM1,0 : パリティ選択 (Parity Mode)**

これらのビットはパリティの発生と検査の許可と形式を設定します。許可した場合、送信部は各フレーム内で送信したデータビットのパリティを自動的に生成して送出します。受信部は到着データに対してパリティ値を生成し、それをUPM0設定と比較します。不一致が検出された場合、USART制御/状態レジスタ(UCSRA)でパリティ誤り(UPE)フラグが設定(1)されます。

表58. パリティ動作選択

UPM1	UPM0	パリティ動作
0	0	禁止
0	1	(予約)
1	0	偶数パリティ許可
1	1	奇数パリティ許可

● **ビット3 – USBS : 停止ビット選択 (Stop Bit Select)**

このビットは送信部によって挿入される停止ビット数を選びます。受信部はこの設定を無視します(訳補:常に第1停止ビットだけが有効)。

表59. USBビット設定

USBS	停止ビット数
0	1ビット
1	2ビット

● **ビット2,1 – UCSZ1,0 : データビット長選択 (Character Size)**

USART制御/状態レジスタB(UCSRB)のUCSZ2ビットと共にUCSZ1,0ビットは送信部と受信部で使うフレームのデータビット数(Character size)を設定します。表60をご覧ください。

表60. UCSZビット設定

UCSZ2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
データビット数	5ビット	6ビット	7ビット	8ビット	(予約)	(予約)	(予約)	9ビット

● **ビット0 – UCPOL : クロック極性選択 (Clock Polarity)**

このビットは同期動作にだけ使われます。非同期動作が使われる時は、このビットに0を書いてください。UCPOLビットはデータ出力変更、データ入力採取、同期クロック(XCK)間の関係を設定します。

表61. クロック極性設定

UCPOL	送信データ変更 (TXDピン出力)	受信データ採取 (RXDピン入力)
0	XCKの上昇端	XCKの下降端
1	XCKの下降端	XCKの上昇端

**15.11.5. UCSRD – USART制御/状態レジスタD (USART Control and Status Register D)**

ビット (\$C3)	7	6	5	4	3	2	1	0	UCSRD
	RXSIE	RXS	SFDE	-	-	-	-	-	
Read/Write	R/W	R/W	R	R	R	R	R	R	
初期値	0	0	1	0	0	0	0	0	

● **ビット7 – RXSIE : 受信開始割り込み許可 (USART RX Start Interrupt Enable)**

このビットへの1書き込みは受信開始(RXS)フラグでの割り込みを許可します。休止動作形態でこのビットは、RXDピンで開始条件が検出された時にMCUを起こすことができるフレーム開始割り込みを許可します。

USART受信開始割り込みはRXSIEビット、ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUCSRDのRXSフラグが設定(1)される場合にだけ生成されます。

● **ビット6 – RXS : 受信開始 (USART RX Start)**

このフラグはRXDピンで開始条件が検出された時に設定(1)されます。受信開始割り込み許可(RXSIE)ビットとSREGの全割り込み許可(I)ビットが設定(1)されていれば、このフラグが設定(1)される時に受信開始割り込みが生成されます。このフラグはRXSビット位置に論理1を書くことによって解除(0)することができます。

フレーム開始検出器が許可(RXSIE=1)され、全割り込み許可(I)ビットが設定(1)されていれば、受信開始割り込みは全ての休止動作形態からMCUを起こします。

● **ビット5 – SFDE : フレーム開始検出許可 (Start Frame Detection Enable)**

このビットへの1書き込みはUSARTフレーム開始動作を許可します。フレーム開始検出器は開始条件、換言すると、RXDピンでHigh(アイドル)からLow(開始)への遷移が検出される時に休止動作形態からMCUを起こすことができます。

表62. USARTフレーム開始検出動作

SFDE	RXSIE	RXCIE	説明
0	x	x	フレーム開始検出禁止。
1	0	0	(予約)
1	0	1	フレーム開始検出許可。RXCフラグは全ての休止動作形態からMCUを起こします。
1	1	0	フレーム開始検出許可。RXSフラグは全ての休止動作形態からMCUを起こします。
1	1	1	フレーム開始検出許可。RXCとRXSの両フラグは全ての休止動作形態からMCUを起こします。

より多くの情報については107頁の「フレーム開始検出」をご覧ください。

●ビット4~0 - 予約 (Reserved)

これらのビットは予約されており、常に0を読みます。

15.11.6. UBRRH, UBRL (UBRR) - USARTボーレートレジスタ (USART Baud Rate Register)

ビット (\$C5)	15	14	13	12	11	10	9	8	
	-	-	-	-	UBRR11	UBRR10	UBRR9	UBRR8	UBRRH
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$C4)	7	6	5	4	3	2	1	0	
	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0	UBRRL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット15~12 - 予約 (Reserved)

これらのビットは将来使用するために予約されています。将来のデバイスとの共通性のため、これらのビットはUBRRHが書かれる時に解除(0)されなければなりません。

●ビット11~0 - UBRR11~0 : ボーレート分周値 (USART Baud Rate Register)

これはUSARTのボーレートを含む12ビットレジスタです。UBRRHがUSARTボーレートの上位4ビットを含み、UBRRLが下位8ビットを含みます。送信部と受信部によって進行中の転送はボーレートが変更される場合に不正にされます。UBRRL書き込みはボーレート前置分周器の更新を直ちに始めます。



## 16. USARTでのSPI動作

### 16.1. 特徴

- 全二重動作、3線同期データ転送
- 主装置動作
- 4つ全てのSPI転送形式(動作種別0,1,2,3)支援
- LSBまたはMSB先行データ転送(データ順設定)
- 順列動作(2重緩衝)
- 高分解能ボーレート発振器
- 高速動作( $f_{XCKmax} = f_{CK}/2$ )
- 柔軟な割り込み生成

### 16.2. 概要

USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は主装置SPI互換動作に設定できます。

USART動作選択(UMSEL1,0)ビットの11設定は主装置SPI(MSPIM)論理回路でのUSARTを許可します。この動作形態でのSPI主装置制御論理回路はUSART資源を直接制御します。これらの資源には送受信の移動レジスタと緩衝部、ボーレート発生器を含みます。パリティ発生/検査論理回路、データ/クロック再生論理回路、送受信制御論理回路は禁止されます。USART送受信論理回路は普通のSPI転送制御論理回路に置き換えられます。けれどもピン制御論理回路と割り込み生成論理回路は両動作形態で全く同じです。

I/Oレジスタ位置は両動作形態で同じです。けれどもMSPIM使用時、制御レジスタの一部の機能が変わります。

### 16.3. クロック生成

クロック生成論理回路は送信部と受信部に対する基準クロックを生成します。USARTのMSPIM動作形態に対しては内部クロック生成(換言すると主装置動作)だけが支援されます。従って、USARTでMSPIMを正しく動かすにはXCKピンが配置されたデータ方向レジスタ(DDR<sub>x</sub>)が出力(DDR\_XCK=1)として設定されなければなりません。なるべくならDDR\_XCKはUSARTでのMSPIMが許可(換言するとTXENとRXENが設定(1))される前に設定されるべきです。

MSPIM動作で使われる内部クロック生成はUSART同期主装置動作と同一です。従ってボーレートやUBRR設定は同じ式を使って計算できます。表63をご覧ください。

表63. ボーレートレジスタ(UBRR)値計算式

動作種別	ボーレート計算 (注)	UBRR値計算
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{2 \times BAUD} - 1$

注: ボーレートは転送速度(ビット/1秒)で定義されます。

BAUD : ボーレート (bps)

UBRR : UBRRHとUBRRLレジスタ値 (0~4095)

$f_{OSC}$  : システム発振器クロック周波数

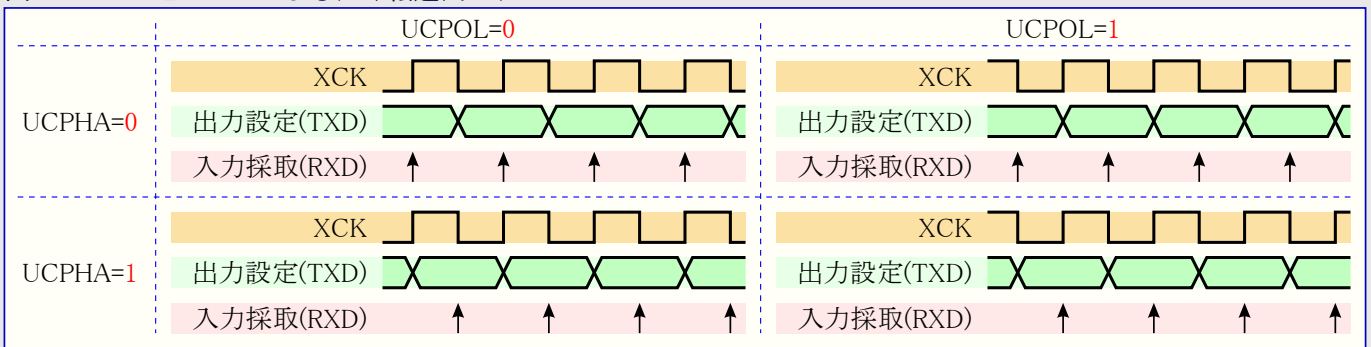
### 16.4. SPIデータ形態とタイミング

直列データに関してはクロック位相(UCPHA)とクロック極性(UCPOL)制御ビットによって決定されるXCK(SCK)位相と極性で4つの組み合わせがあります。このデータ転送タイミング図は図63.で示されます。データビットは安定のためにデータ信号に対して十分な時間を保証するXCK信号の反対端で移動出力と(入力)ラッチが行われます。UCPOLとUCPHAの機能は表64.で要約されます。これらビットのどの変更も送受信両方に対して実行中の通信を不正とすることに注意してください。

表64. UCPOL,UCPHA機能動作

SPI動作種別番号	UCPOL	UCPHA	XCK(SCK)先行端	XCK(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

図63. UCPHAとUCPOLによるデータ転送タイミング



## 16.5. フレーム形式

MSPIMの直列フレームは8データビット/1フレームに固定されます。USARTでのMSPIM動作は2つの有効なフレーム形式を持ちます。

- MSB先行 8ビット データ
- LSB先行 8ビット データ

フレームは最下位(LSB)または最上位(MSB)のデータビットで始まります。その後次のデータビットが、それによって最上位(MSB)または最下位(LSB)データビットで終わる合計8ビットまで続きます。完全なフレームが送信されると、新規フレームが直ちに後続するか、または通信線がアイドル状態(**high**)に設定されるでしょう。

USART制御/状態レジスタ(UCSRC)のデータ順選択(UDORD)ビットはUSARTでのMSPIMによって使われるフレーム形式を設定します。送受信部は同じ設定を使います。これらビットのどれかの設定変更が送受信部両方に対して実行中の通信を不正にすることに注意してください。

16ビットデータ通信はUDRに2バイトデータを書くことによって達せられます。その後のUSART送信完了割り込み(TXC)はこの16ビット値が移動出力されてしまったことを示します。

### 16.5.1. USART MSPIM初期化

USARTでのMSPIM動作はどれかの通信が行われ得るのに先立って初期化されなければなりません。標準的な初期化手順は使用方法に依存するボーレート設定、主装置動作構成設定、フレーム形式設定、送受信部許可から成ります。送信部だけが独立して操作できません。割り込み駆動USART操作に関して初期化を行う時にステータスレジスタの全割り込み許可(I)ビットが解除(0)される(そして全割り込みが禁止される)べきです。

**注:** XCK出力の初期化を直ちに保証するために、ボーレートレジスタ(UBRR)は送信部が許可される時に0でなければなりません。標準動作のUSART操作と逆に、UBRRは送信部が許可されて最初の送信が開始される前に、希望する値を書かれなければなりません。この初期化がリセット後直に行なわれるなら、UBRRが0にリセットされているので、送信部許可前にUBRRを0に設定することは必要ありません。

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

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

#### アセンブリ言語プログラム例

```
USART_Init: CLR    R18                ;0値取得
             OUT    UBRRH, R18        ;ボーレート設定(上位バイト)=0
             OUT    UBRL, R18         ;ボーレート設定(下位バイト)=0 (クロック停止)
             SBI    XCK_DDR, XCK      ;XCKポートピン出力設定
             LDI    R18, (1<<UMSEL1) | (1<<UMSEL0) | (0<<UCPHA) | (0<<UCPOL)
             ;動作種別値を取得
             OUT    UCSRC, R18        ;MSPI,データ種別0設定
             LDI    R18, (1<<RXEN) | (1<<TXEN)
             ;送受信許可値を取得
             OUT    UCSRB, R18        ;送受信許可
             OUT    UBRRH, R17        ;ボーレート設定(上位バイト)
             OUT    UBRL, R16        ;ボーレート設定(下位バイト)
             RET                       ;呼び出し元へ復帰
```

#### C言語プログラム例

```
void USART_Init(unsigned int baud)
{
    UBRR = 0;                /* ボーレート設定(クロック停止) */
    XCK_DDR |= (1<<XCK);    /* XCKポートピン出力設定 */
    UCSRC = (1<<UMSEL1) | (1<<UMSEL0) | (0<<UCPHA) | (0<<UCPOL);
    ;動作種別値を取得
    UCSRB = (1<<RXEN) | (1<<TXEN); /* 送受信許可 */
    UBRR = baud;           /* ボーレート設定 */
}
```

**注:** 5頁の「コード例」をご覧ください。

## 16.6. データ転送

USARTでのMSPI動作の使用は送信部が許可されること、換言するとUSART制御/状態レジスタ(UCSRB)で送信許可(TXEN)ビットが1に設定される必要があります。送信部が許可されると、TXDピンの標準ピン動作は無視され、送信部の直列出力としての機能を与えられます。受信部の許可は任意選択で、UCSRBの受信許可(RXEN)ビットの設定(=1)によって行なわれます。受信部が許可されると、RXDピンの標準ピン動作は無視され、受信部の直列入力としての機能を与えられます。XCKは両方の場合で転送クロックとして使われます。

初期化後、USARTはデータ転送を行なう準備が整います。データ転送はUSARTデータレジスタ(UDR)I/O位置に書くことによって開始されます。送信部が転送クロックを制御するため、これは送受信データ両方の状態についてです。UDRに書かれたデータは移動レジスタが新規フレームを送る準備が整った時に送信緩衝部から移動レジスタへ移されます。

**注:** 入力緩衝部に於いて送信されたデータバイト数との同期を保つために、送信された各バイトに対して一度、UDRが読まれない限りなりません。入力緩衝部操作は通常のUSART動作と同じで、換言すると、オーバーランが起きると、緩衝部内の先頭ではなく最後に受信したデータが失われます。これは第1バイト、第2、第3、第4バイトの順で4バイトのデータが送信され、全ての転送が完了される前にDDRが読まれない場合、第1バイトではなく、受信されるべき第3バイトが失われます。

次のコード例はUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグのポーリングに基いた、簡単なUSARTでのMSPI M転送関数を示します。この関数が使われ得る前にUSARTが初期化されなければなりません。アセンブリ言語での送るべきデータはR16レジスタに格納されると仮定され、受信したデータは関数復帰後、同じR16レジスタで利用できます。

この関数は送信されるべき新規データを設定する前に、UDREの検査によって送信緩衝部が空になるのを単純に待ちます。その後緩衝部を読んで値を戻す前に、RXCフラグの検査によって受信緩衝部にデータが存在するのを待ちます。

### アセンブリ言語プログラム例

```
USART_MSPI: SBIS    UCSRA, UDRE          ;送信緩衝部空きでスキップ
             RJMP    USART_MSPI        ;送信緩衝部空き待機
;
             OUT     UDR, R16           ;データ送信(送信開始)
USART_SPI:  SBIS    UCSRA, RXC          ;受信完了でスキップ
             RJMP    USART_SPI        ;受信完了待機
;
             IN     R16, UDR           ;受信データ取得
             RET                          ;呼び出し元へ復帰
```

### C言語プログラム例

```
void USART_MSPI_Transfer(unsigned int data)
{
    while ( !(UCSRA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UDR = data;                          /* データ送信(送信開始) */
    while ( !(UCSRA & (1<<RXC)) );    /* 受信完了待機 */
    return UDR;                          /* 受信データ取得 */
}
```

**注:** 5頁の「コード例」をご覧ください。

### 16.6.1. 送受信フラグと割り込み

USARTのMSPI動作での受信完了(RXC)、送信完了(TXC)、送信データレジスタ空き(UDRE)フラグと対応する割り込みは通常のUSART操作と機能的に同一です。けれども受信異常状態フラグ(FE,DOR,UPE)は使えず、常に0を読みます。

### 16.6.2. 送受信の禁止

USARTのMSPI動作での送受信部の禁止は通常のUSART操作と機能的に同一です。

## 16.7. AVR SPIとの互換性

USARTでのMSPI動作は次に関してSPIと完全な互換性があります。

- 主装置動作タイミング図
- クロック極性選択(UCPOL)ビットはSPIのSCK極性選択(CPOL)ビットと機能的に同じです。
- クロック位相選択(UCPHA)ビットはSPIのSCK位相選択(CPHA)ビットと機能的に同じです。
- データ順選択(UDORD)ビットはSPIのデータ順選択(DORD)ビットと機能的に同じです。

けれどもUSARTでのMSPI動作がUSART資源を再使用するため、USARTでのMSPI動作はSPIと比較して多少異なります。加えて制御レジスタビットの差異、主装置動作だけがUSARTでのMSPI動作によって支援されること、2つの部間で異なる次の特質があります。

- USARTでのMSPI動作は送信部の(2重)緩衝部を含みます。SPIは緩衝部を持ちません。
- USARTのMSPI動作での受信部は追加の緩衝段を含みます。
- SPIの上書き(WCOL)ビットはUSARTでのMSPI動作に含まれません。
- SPIの倍速許可(SPI2X)ビットは含まれません。しかし、対応するボーレートレジスタ(UBRR)設定によって同じ効果が達せられます。
- 割り込みタイミングに互換性はありません。
- USARTでのMSPI動作が主装置動作だけのため、ピン制御が異なります。

USARTのMSPI動作とSPIでのピンは表65.で示されます。

表65. USARTでのMSPIMとSPIのピン比較

USART MSPIM	SPI	注釈
TXD	MOSI	主装置出力のみ
RXD	MISO	主装置入力のみ
XCK	SCK	機能的に同一
該当なし	$\overline{SS}$	USARTでのMSPIMで未支援

## 16.8. MSPIMでのUSART用レジスタ

以下の節はUSARTを使うSPI操作で使われるレジスタを記述します。

### 16.8.1. UDR – USART MSPIMデータレジスタ (USART I/O Data Register)

MSPIMでのUSARTデータレジスタ(UDR)の機能と説明は通常のUSART操作と同一です。111ページの「[UDR – USARTデータレジスタ](#)」をご覧ください。

### 16.8.2. UCSRA – USART MSPIM制御/状態レジスタA (USART Control and Status Register A)

ビット (\$C0)	7	6	5	4	3	2	1	0	UCSRA
	RXC	TXC	UDRE	–	–	–	–	–	
Read/Write	R	R/W	R	R	R	R	R	R	
初期値	0	0	1	0	0	0	0	0	

#### ● ビット7 – RXC : USART受信完了フラグ (USART Receive Complete)

このフラグは受信緩衝部に未読データがある時に設定(1)され、受信緩衝部が空の(換言すると、どんな未読データも含まない)時に解除(0)されます。受信部が禁止されると、受信緩衝部が破棄され、その結果RXCフラグが0になります。RXCフラグは受信完了割り込みを発生するのに使うことができます(USART制御/状態レジスタB(UCSRB)の受信完了割り込み許可(RXCIE)ビットをご覧ください)。

#### ● ビット6 – TXC : USART送信完了フラグ (USART Transmit Complete)

このフラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部(UDR)内に新規データが現在全く存在しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行される時に自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)することができます。TXCフラグは送信完了割り込みを発生することができます(UCSRBの送信完了割り込み許可(TXCIE)ビットをご覧ください)。

#### ● ビット5 – UDRE : USART送信データレジスタ空きフラグ (USART Data Register Empty)

UDREフラグは送信緩衝部(UDR)が新規データを受け取り準備可かどうかを示します。UDREが1ならば緩衝部は空で、従って書かれる準備が整っています。UDREフラグは送信緩衝部空き割り込みを発生することができます(UCSRBの送信データレジスタ空き割り込み許可(UDRIE)ビットをご覧ください)。UDREフラグは送信部が準備可であることを示すためにリセット後に設定(1)されます。

#### ● ビット4~0 – 予約 (Reserved in MSPI mode)

MSPI動作での時にこれらのビットは将来の使用に予約されています。将来のデバイスとの共通性のため、これらのビットはUCSRAが書かれる時に0を書かれなければなりません。

### 16.8.3. UCSRB – USART MSPIM制御/状態レジスタB (USART Control nad Status Register B)

ビット (\$C1)	7	6	5	4	3	2	1	0	UCSRB
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	–	–	–	
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R	R	
初期値	0	0	0	0	0	0	0	0	

#### ● ビット7 – RXCIE : 受信完了割り込み許可 (Receive Complete Interrupt Enable)

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグでの割り込みを許可します。USART受信完了割り込みはRXCIEビットが1を書かれ、ステータスレジスタ(SREG)で全割り込み許可(I)ビットが1を書かれ、UCSRAでRXCフラグが設定(1)される場合にだけ生成されます。

#### ● ビット6 – TXCIE : 送信完了割り込み許可 (Transmit Complete Interrupt Enable)

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の送信完了(TXC)フラグでの割り込みを許可します。USART送信完了割り込みはTXCIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでTXCフラグが設定(1)される場合にだけ生成されます。

#### ● ビット5 – UDRIE : 送信データレジスタ空き割り込み許可 (Transmit Data Register Empty Interrupt Enable)

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグでの割り込みを許可します。USART送信データレジスタ空き割り込みはUDRIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでUDREフラグが設定(1)される場合にだけ生成されます。

#### ● ビット4 – RXEN : 受信許可 (Receiver Enable)

このビットへの1書き込みはMSPIMでのUSART受信(部)を許可します。受信部は許可時にRXDピンの標準ポート動作を無効にします。受信禁止は受信緩衝部を破棄します。MSPI動作で受信(部)だけを許可すること(換言すると、RXEN=1とTXEN=0の設定)は転送クロックを制御するのが送信部で主装置動作だけが支援されるため、意味を持ちません。



● **ビット3 – TXEN : 送信許可 (Transmitter Enable)**

このビットへの**1**書き込みがUSART送信(部)を許可します。送信部は許可時にTXDピンの標準ポート動作を無効にします。送信(部)の禁止(TXENの**0**書き込み)は進行中と保留中の送信が完了される、換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない時まで有効になりません。禁止時、送信部はもはやTXDポート(の標準I/O機能)を無効にしません。

● **ビット2~0 – 予約 (Reserved in MSPI mode)**

MSPI動作での時にこれらのビットは将来の使用に予約されています。将来のデバイスとの共通性のため、これらのビットはUCSRB書き込み時に**0**を書かれなければなりません。

**16.8.4. UCSRC – USART MSPIM制御/状態レジスタC (USART Control nad Status Register C)**

ビット (\$C2)	7	6	5	4	3	2	1	0	UCSRC
	UMSEL1	UMSEL0	–	–	–	UDORD	UCPHA	UCPOL	
Read/Write	R/W	R/W	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	1	1	0	

● **ビット7,6 – UMSEL1,0 : USART動作選択 (USART Mode Select)**

このビットは表66で示されるようにUSART動作種別を選びます。通常USART操作の完全な記述については112頁の「UCSRC – USART制御/状態レジスタC」をご覧ください。主装置SPI動作(MSPIM)は両UMSELビットが**1**に設定される時に許可されます。データ順選択(UDORD)、クロック位相選択(UCPHA)、クロック極性選択(UCPOL)はMSPIMが許可されるのと同じ書き込み操作で設定することができます。

表66. UMSELビット設定

UMSEL1	UMSEL0	動作形態
0	0	非同期動作
	1	同期動作
1	0	(予約)
	1	主装置SPI (MSPIM)

● **ビット5~3 – 予約 (Reserved in MSPI mode)**

MSPI動作での時にこれらのビットは将来の使用に予約されています。将来のデバイスとの共通性のため、これらのビットはUCSRC書き込み時に**0**を書かれなければなりません。

● **ビット2 – UDORD : データ順選択 (Data Order)**

**1**に設定時、データ語のLSBが最初に転送されます。**0**に設定時、データ語のMSBが最初に転送されます。詳細については116頁の「フレーム形式」をご覧ください。

● **ビット1 – UCPHA : クロック位相選択 (Clock Phase)**

UCPHAビット設定はデータがXCK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。詳細については115頁の「SPIデータ形態とタイミング」をご覧ください。

● **ビット0 – UCPOL : クロック極性選択 (Clock Polarity)**

UCPOLビットはXCKクロックの極性を設定します。UCPOLとクロック位相選択(UCPHA)ビット設定の組み合わせがデータ転送のタイミングを決めます。詳細については115頁の「SPIデータ形態とタイミング」をご覧ください。

**16.8.5. UBRRH, UBRL (UBRR) – USART MSPIMボーレートレジスタ (USART Baud Rate Register)**

MSPIMでのボーレートレジスタ(UBRR)の機能と説明は通常のUSART操作と同一です。114頁の「UBRRH, UBRL (UBRR) – USARTボーレートレジスタ」をご覧ください。

## 17. I<sup>2</sup>C適合2線従装置インターフェース (TWI: I<sup>2</sup>C Compatible, Two Wire Slave Interface)

### 17.1. 特徴

- Phillips社I<sup>2</sup>C適合
- (条件付き)SMBus適合
- 低システム クロックで100kHzと400kHzを支援
- スレーブ制限された出力駆動部
- 雑音消去を提供する入力濾波器
- ハードウェアでの7ビットと一斉呼び出しのアドレス認証
- アドレス遮蔽または2重アドレス一致用のアドレス遮蔽レジスタ
- 10ビット アドレス指定支援
- 無制限の従装置アドレス数を提供する任意選択のソフトウェア アドレス認証
- パワーダウン動作を含む全休止形態動作での動作
- SMBusアドレス解決規約(ARP)に対する支援を許す従装置調停

### 17.2. 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。I<sup>2</sup>C適合と条件付きSMBus適合です(124頁の「SMBusでの適合性」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か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。

ATtiny828のTWI単位部は従装置の機能だけを実装しています。バス上の協調損失、異常、衝突、クロック保持はハードウェアで検出され、独立した状態フラグで示されます。

7ビットと一斉アドレス呼び出しがハードウェアで実装されています。10ビットアドレスも支援されます。専用のアドレス遮蔽レジスタは第2のアドレス一致レジスタまたはアドレスの範囲での一致のための従装置アドレスに対する遮蔽レジスタとして働くことができます。従装置論理回路はパワーダウン動作を含む全ての休止形態動作で動作を継続します。これはTWIアドレス一致での休止からの起動を従装置に許します。アドレス一致を禁止し、代わりにソフトウェアでこれを扱うことも可能です。これは多数のアドレスに対する検知と応答を従装置に許します。自動起動操作のために簡便動作を許可することができ、ソフトウェアの複雑さを低減します。

TWI単位部は開始条件、停止条件、バス衝突、バス異常を検知するための情報を収集するバス状況論理回路を含みます。バス状況論理回路はパワーダウン動作を含む全ての休止形態で動作を継続します。

### 17.3. 一般的なTWIバスの概念

2線インターフェース(TWI)は直列クロック線(SCL)と直列データ線(SDA)から成る簡素な2線双方向バスを提供します。この2線は開放コレクタ (トレン)線(ワイヤードAND)で、プルアップ抵抗器だけがバスを駆動するために必要とされる外部部品です。このプルアップ抵抗は接続された装置がバスを駆動しない時の信号線にHighレベルを供給します。プルアップ抵抗の代替として定電流源が使えます。

TWIバスは直列バスでの複数装置内部接続の簡単で効率的な方法です。バスに接続された装置は主または従装置にでき、主装置がバスと全ての通信を制御します。

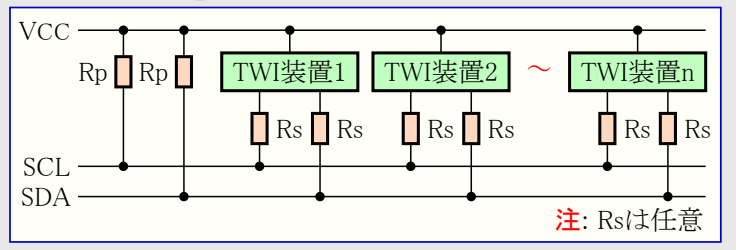
図64.はTWIバス形態を図解します。

バスに接続した全ての従装置に固有のアドレスが割り当てられ、主装置は従装置をアドレス指定するのにこれを使ってデータ転送処理を始めます。7ビットまたは10ビットのアドレス指定が使えます。

多数の主装置が同じバスに接続でき、これは複数主装置環境と呼ばれます。与えられた時間で1つの主装置だけがバスを自身のものにできるので、主装置間のバス所有権を解決するために調停機構が提供されます。

デバイスは主装置と従装置の両論理回路を含み、2つ以上のアドレスに応答することによって複数従装置の擬似動作ができます。

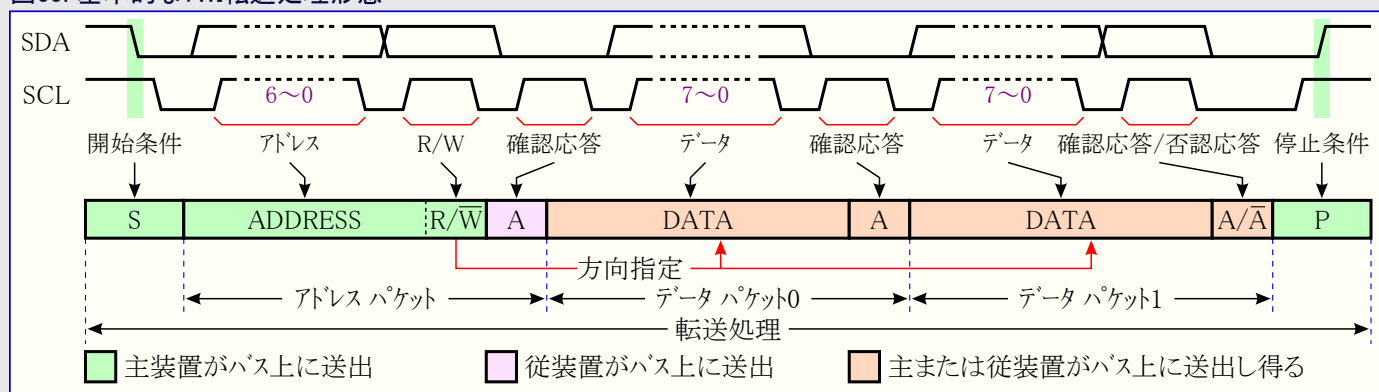
図64. TWIバス形態



主装置がバス上に**開始条件(S)**を発行することによって転送処理の開始を指示します。それから、従装置アドレス(ADDRESS)と主装置がデータを読みまたは書きどちらをしたいのかを示す(R/W)を持つアドレスパケットが送られます。データパケット(DATA)が転送された後、転送処理終了のためにバス上へ**停止条件(P)**を発行します。受信側は各バイトに対して**確認応答(A)**または**否認応答( $\bar{A}$ )**を行わなければなりません。

図65.はTWI転送処理を示します。

図65. 基本的なTWI転送処理形態



主装置が転送処理に対してクロック信号を供給しますが、バスに接続された装置はクロック速度を下げるためにクロックのLow区間を引き伸ばすことが許されます。

### 17.3.1. 電気的特性

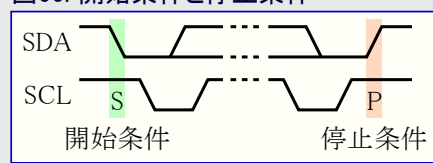
TWIはI<sup>2</sup>CとSMBusの電気的仕様とタイミングに従っています。169頁の「**2線直列インターフェース特性**」と124頁の「**SMBusでの適合性**」をご覧ください。

### 17.3.2. 開始条件と停止条件

転送処理の始まり(**開始条件**)と終り(**停止条件**)を記すために2つの独特なバス状態が使用されます。主装置はSCL線をHighに保持すると同時にSDA線でHighからLowへの遷移を示すことによって**開始条件(S)**を発行します。主装置はSCL線をHighに保持すると同時にSDA線をLowからHighへの遷移を示すことによって**停止条件(P)**を発行し、それによって転送処理を完了します。

単一転送処理の間に複数の**開始条件**が発行され得ます。**停止条件**に直接後続しない**開始条件**は**再送開始条件(Sr)**と名付けられます。

図66. 開始条件と停止条件

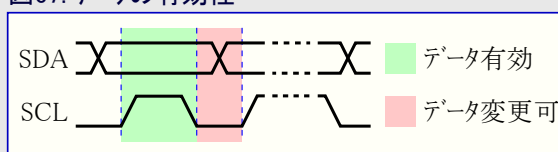


### 17.3.3. ビット転送

図67.で図解されるように、SDA線で転送されるビットはSCL線のHigh区間全体に対して安定でなければなりません。従ってSDA値はクロックのLow区間の間でだけ変更できます。これはTWI単位部でのハードウェアで保証されます。

ビット転送の組み合わせがアドレスとデータの packets の編成に帰着します。これらの packets は最上位ビット先行転送の8データビット(1バイト)と**確認(ACK)**または**否認(NACK)**の応答の単一ビットから成ります。アドレス指定された装置は9クロック周期の間に、SCL線をLowに引くことによってACKで、SCL線をHighのままにして置くことによってNACKで合図します。

図67. データの有効性



### 17.3.4. アドレスパケット

**開始条件**後、読み/書き(R/W)ビットが後続する7ビットアドレスが送出されます。これは常に主装置によって送出されます。そのアドレスを認証する従装置は次のSCLクロックでSDA線をLowに引くことによってアドレスの**確認応答(ACK)**を行い、一方他の全ての従装置はTWI線の開放を維持して次の**開始条件**とアドレスを待ちます。7ビットアドレス、R/Wビット、応答ビットの組み合わせがアドレスパケットです。各**開始条件**に対して1つのアドレスパケットだけが与えられ、これは10ビットアドレスが使われる時です。

R/Wは転送処理の方向を指定します。R/WビットがLowなら、主装置書き込み転送処理を示し、従装置のそのアドレスの**確認応答**後に主装置はそのデータを送出します。逆の主装置読み込み操作については、従装置がそのアドレスの**確認応答**後にデータ送出を開始します。

### 17.3.5. データパケット

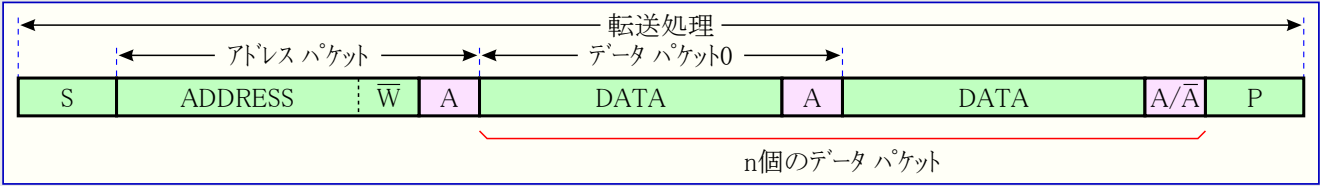
データパケットはアドレスパケットまたは他のデータパケットに続きます。全てのデータパケットは1つのデータバイトと応答ビットから成る9ビット長です。直前のアドレスパケット内の方向ビットがデータが転送される方向を決めます。

### 17.3.6. 転送処理

転送処理は開始条件から停止条件までの完全な転送で、その間のどの再送開始条件も含まれます。TWI規格は3つの基本転送処理種別、主装置書き込み、主装置読み込み、組み合わせの転送処理を定義しています。

図68.は主装置書き込み転送処理を図解します。主装置は開始条件(S)によって転送処理を開始し、方向ビットを0に設定したアドレスパケット(ADDRESS+W)がそれに後続します。

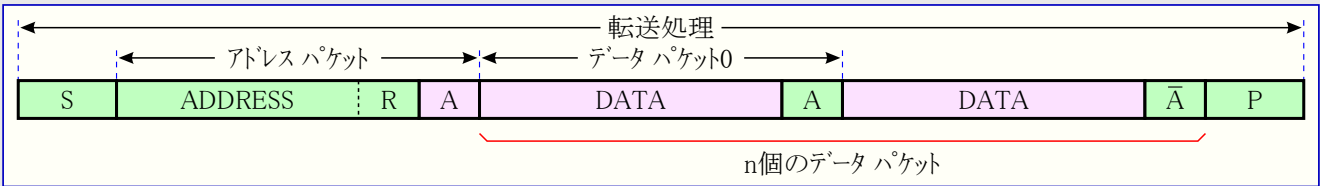
図68. 主装置書き込み転送処理



従装置のアドレス確認応答が与えられると、主装置はデータ(DATA)の送信を開始でき、従装置は各バイトで確認応答(ACK)または否認応答(NACK) (A/A-bar)を行います。送信すべきデータパケットがない場合、主装置はアドレスパケット直後に停止条件(P)を発行することによって転送処理を終了します。転送できるデータパケット数に制限はありません。従装置がデータに否認応答(NACK)で合図した場合、主装置は従装置がこれ以上データを受信できず、転送処理を終了すると認識しなければなりません。

図69.は主装置読み込み転送処理を図解します。主装置は開始条件(S)によって転送処理を開始し、方向ビットを1に設定したアドレスパケット(ADDRESS+R)がそれに後続します。アドレス指定された従装置は転送処理の継続を許す主装置に対してアドレスの確認応答をしなければなりません。

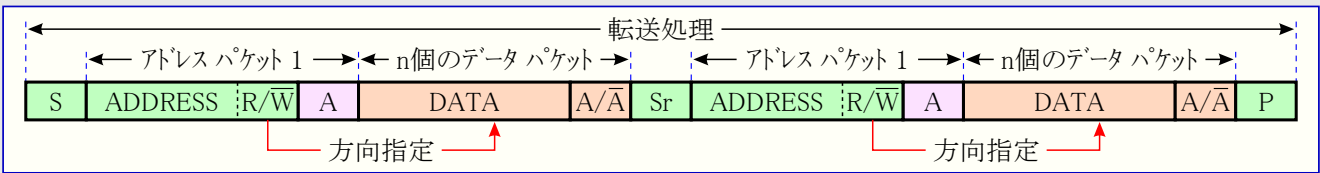
図69. 主装置読み込み転送処理



従装置のアドレス確認応答が与えられると、主装置は従装置からのデータ受信を開始できます。転送できるデータパケット数に制限はありません。従装置がデータを送信する一方で、主装置は各データバイト後に確認応答(ACK)または否認応答(NACK)で合図します。主装置は停止条件を発行する前に否認応答(NACK)で転送を終了します。

図70.は組み合わせ転送処理を図解します。組み合わせ転送処理は再送開始条件(Sr)によって分離された多数の読み込みと書き込みの転送処理から成ります。

図70. 組み合わせ転送処理



### 17.3.7. クロックとクロック伸長

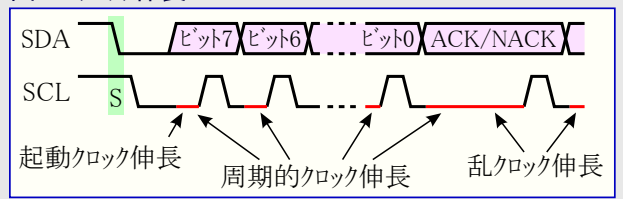
バスに接続された全ての装置はクロック周波数全体を低下するため、またはデータ処理の間に待ち状態を挿入するために、クロックのLow区間を伸長することが許されています。クロックの伸長を必要とする装置はSCL線上のLowレベル検出後にその線をLowに強制保持することによってこれを行えます。

クロック伸長は図71.で示されるように3つの形式に定義できます。デバイスが休止形態動作で開始条件が検出された場合、クロックはデバイスに対する起動時間の間、引き伸ばされます。

従装置はビット単位で周期的にクロックを伸長することによってバス周波数を低下できます。けれども、それによってバス全体性能が低下されます。主と従の両装置は応答(ACK/NACK)ビットの前後を基本にバイト単位で任意にクロックを引き伸ばせます。これは到着処理、出力データ準備、または重要な作業の別時間実行のための時間を提供します。

従装置がクロックを伸長する場合は、従装置の準備が整うまで主装置が待ち状態を強制され、その逆も同様です。

図71. クロック伸長





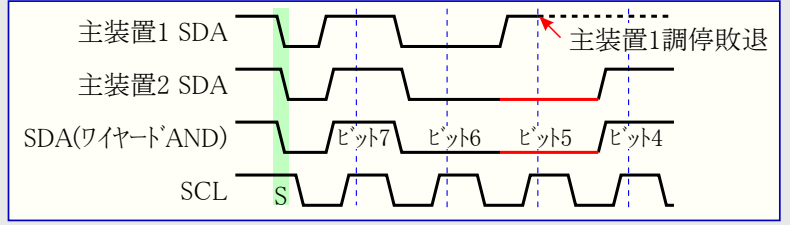
### 17.3.8. 調停

主装置はバスのアイドルを検出した場合にだけバス転送処理を開始できます。TWIバスが複数主装置バスなので、2つの装置が同時に転送処理を始めることが可能です。これは複数の主装置が同時にバスを所有する結果になります。これは調停の仕組みを使って解決され、そしてこれはSDA線でHighレベルを送信できなかった場合にその主装置がバスの制御を失います。調停で敗れた主装置はその後にバス所有権要求を試みる前に、バスがアイドルになるまで待たなければなりません(換言すると**停止条件待機**)。従装置は調停手順に関係しません。

図72.は、2つのTWI主装置がバス所有権を争う例を示します。両装置が**開始条件**を発行できますが、主装置1がHighレベル(ビット5)の送信を試み、同時に主装置2がLowレベルを送信する時に主装置1が調停に敗れます。

再送開始条件とデータビット、停止条件とデータビット、または再送開始条件と停止条件の間の調停は許されず、ソフトウェアによる特別な扱いが必要です。

図72. TWI調停

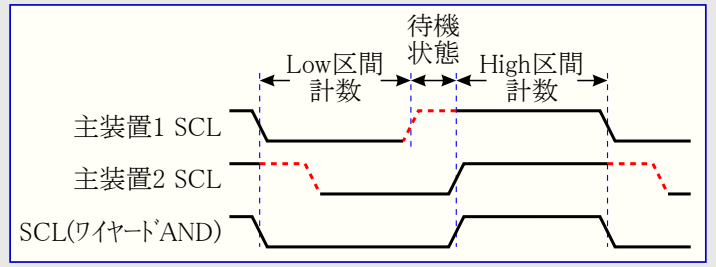


### 17.3.9. 同期化

複数の主装置が同時にSCL線の制御を試みる状況を解決するのにクロック同期化手法が必要です。この方法は直前に記述されたクロック伸長で使われるのと同じ原理に基づいています。図73.は2つの主装置がバスクロックに関する制御を競争する例を示します。SCL線は2つの主装置のクロック出力のワイヤードANDの結果です。

SCL線のHighからLowへの遷移がバス上の全ての主装置に対してLowを強制し、それらはそれぞれのクロックLow区間タイミングを開始します。このクロックLow区間のタイミング長は主装置間で変わり得ます。主装置(この場合は主装置1)はLow区間を完了すると、SCL線を開放します。けれども、SCL線は全ての主装置がSCLを解放する前にはHighになりません。従ってSCL線は最長Low区間を持つ装置(主装置2)によってLowを維持されます。より短いLow区間を持つ装置はクロックが開放されるまで待機状態を挿入しなければなりません。全ての装置でSCLが開放されてHighになる時に、全て主装置がそれらのHigh区間を始めます。最初にHigh区間を完了した装置(主装置1)がクロック線をLowに強制し、そしてこの手順が繰り返されます。その結果は最短クロック区間を持つ装置がHigh区間を決め、一方クロックのLow区間は最長クロック区間によって決められることです。

図73. クロック同期化



### 17.3.10. SMBusでの適合性

他の何れかのI<sup>2</sup>C適合インターフェースとで既知の適合性の問題があるため、TWI装置をSMBus装置に接続する前に設計者は注意すべきです。SMBus環境での使用に対して、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I/O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。40頁の図21.をご覧ください。これはSMBus仕様の必要条件に矛盾します。結果として、供給電圧がAVRから取り去られてはならないか、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。電力断と休止動作形態は供給電圧が存続するなら問題ありません。
- TWIのデータ保持時間はSMBusに対して指定されるものよりも小さい値です。TWSCRAのTWI SDA保持時間許可(TWSHE)ビットは保持時間を増すのに使うことができます。126頁の「TWSCRA – TWI従装置制御レジスタ」をご覧ください。
- SMBusは低速限界を持ち、一方I<sup>2</sup>Cにはありません。AVRがSMBus環境での主装置の時に、より低いバス速度がSMBus従装置で時間超過を起動するため、バス速度が仕様以下に落ちないことを確実にしなければなりません。AVRが従装置として構成設定される場合、TWI単位部が時間超過を確認しないため、バス固着の可能性がります。

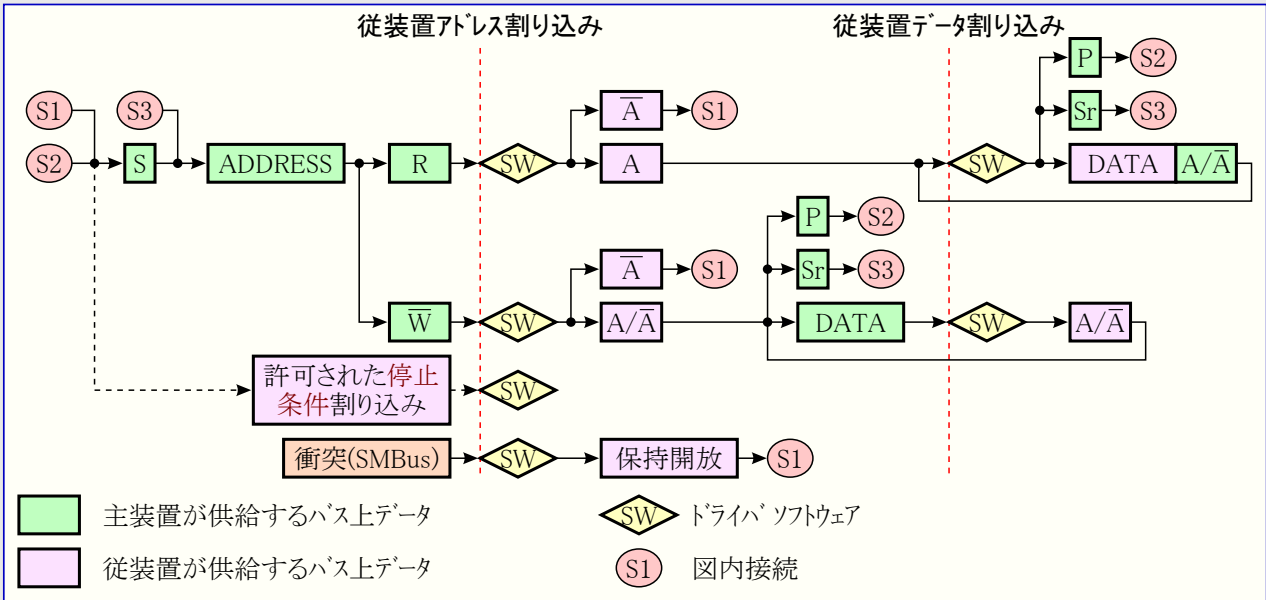


## 17.4. TWI従装置動作

TWI従装置は各バイト後の任意選択の割り込みを持つバイト志向です。独立したデータ割り込みとアドレス/停止割り込みがあります。割り込みフラグはTWI割り込みの起動、またはポーリング操作に使うことができます。受信した(ACK)確認応答/(NACK)否認応答(TWRA)、クロック保持(TWCH)、衝突(TWC)、バス異常(TWBE)、読み/書き方向(TWDIR)を示す専用の状態フラグがあります。

割り込み要求フラグが設定(1)されると、SCL線はLowを強制されます。これは応答または何れかのデータを扱う時間を従装置に与え、殆どの場合はソフトウェアの介入を必要とするでしょう。図74.はTWI従装置動作を示します。菱形シンボル(SW)はソフトウェアの介入を必要とする場所を示します。

図74. TWI従装置動作



生成される割り込み数は殆どの条件を自動的に扱うことによって最小に保たれています。自動起動操作とソフトウェアの複雑さを低減するために**簡便動作**を許可にできます。

全てのアドレスに応答することを従装置に許すために**無差別動作(TWPME)**を許可にできます。

### 17.4.1. アドレス パケット受信

TWI従装置が正しく構成設定されていると、検出されるべき**開始条件**を待ちます。これが起きると、継続してアドレスバイトが受信されてアドレス一致論理回路によって調べられ、従装置は正しいアドレスに**確認応答(ACK)**します。受信したアドレスが不一致なら、従装置はアドレスに**否認応答**せず、新しい**開始条件**を待ちます。

**開始条件**に後続する有効なアドレス パケットが検出される時に、**従装置アドレス/停止割り込み要求フラグ(TWASIF)**が設定(1)されます。一斉呼び出しアドレスもこのフラグを設定(1)します。

**開始条件**直後の**停止条件**は不正操作で、**バス異常(TWBE)フラグ**が設定(1)されます。

**R/W方向(TWDIR)フラグ**はアドレスと共に受信された方向ビットを反映します。これは現在進行中の操作形式を決めるためにソフトウェアによって読むことができます。

R/W方向ビットとバス状況に依存して、以降のアドレス パケットで4つの別個の状態の1つ(4つの内の1つ)が起きます。異なる状態はソフトウェアで扱われなければなりません。

#### 17.4.1.1. 状態1: アドレス パケット受け入れ - 方向ビット=1

R/W方向(TWDIR)フラグが設定(1)されている場合、これは主装置読み込み操作を示します。SCL線がLowを強制され、バス クロックを引き伸ばします。従装置によって**確認応答(ACK)**が送出される場合、従装置ハードウェアは送信のためのデータが必要なことを示す**データ割り込み要求フラグ(TWDIF)**を設定(1)します。従装置によって**否認応答(NACK)**が送出される場合、従装置は新しい**開始条件**とアドレス一致を待ちます。

#### 17.4.1.2. 状態2: アドレス パケット受け入れ - 方向ビット=0

R/W方向(TWDIR)フラグが解除(0)されている場合、これは主装置書き込み操作を示します。SCL線がLowを強制され、バス クロックを引き伸ばします。従装置によって**確認応答(ACK)**が送出される場合、従装置は受信されるべきデータを待ちます。その後データ、**再送開始条件**、**停止条件**が受信され得ます。**否認応答(NACK)**が示された場合、従装置は新しい**開始条件**とアドレス一致を待ちます。

#### 17.4.1.3. 状態3: 衝突

従装置がHighレベルまたは**否認応答(NACK)**を送出できない場合、**衝突(TWC)フラグ**が設定(1)され、従装置からのデータと応答の出力が禁止されます。クロック保持は開放されます。**開始条件**と**再送開始条件**は受け入れられます。

#### 17.4.1.4. 状態4: 停止条件受信

動作は1つの例外付きで先の**状態1**または**状態2**と同じです。**停止条件**が受信されると、アドレス一致発生ではなく**停止条件**を示す**従装置アドレス/停止割り込み要求フラグ(TWASIF)**が設定(1)されます。

#### 17.4.2. データパケット受信

従装置は解除(0)されたR/W方向ビットと共にアドレスパケットが成功裏に受信された時に知ります。これの応答後、従装置はデータを受信する準備を整えなければなりません。データパケットが受信されると、データ割り込み要求フラグ(TWDIF)が設定(1)され、従装置は**確認応答(ACK)**または**否認応答(NACK)**を示さなければなりません。NACK提示後、従装置は**停止条件**または**再送開始条件**を期待して待たなければなりません。

#### 17.4.3. データパケット送信

従装置は設定(1)されたR/W方向ビットと共にアドレスパケットが成功裏に受信された時に知ります。その後に従装置データ(TWSD)レジスタへの書き込みによって送出手始められます。データパケット送信が完了されると、**データ割り込み要求フラグ(TWDIF)**が設定(1)されます。主装置が**否認応答(NACK)**を提示する場合、従装置はデータ送信を停止し、**停止条件**または**再送開始条件**を期待して待たなければなりません。

### 17.5. TWI用レジスタ

#### 17.5.1. TWSCRA – TWI 従装置制御レジスタA (TWI Slave Control Register A)

ビット (\$B8)	7	6	5	4	3	2	1	0	TWSCRA
	TWSHE	–	TWDIE	TWASIE	TWEN	TWSIE	TWPME	TWSME	
Read/Write	R/W	R	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 – TWSHE : TWI SDA保持時間許可 (TWI SDA Hold Time Enable)**  
このビットが設定(1)されると、SCLの負端に対するSDAの内部保持時間が許可されます。
- **ビット6 – 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5 – TWDIE : TWIデータ割り込み許可 (TWI Data Interrupt Enable)**  
このビットが設定(1)され、割り込みが許可されていると、TWI従装置状態レジスタA(TWSSRA)の**データ割り込み要求フラグ(TWDIF)**が設定(1)される時にTWI割り込みが生成されます。
- **ビット4 – TWASIE : TWIアドレス/停止割り込み許可 (TWI Address/Stop Interrupt Enable)**  
このビットが設定(1)され、割り込みが許可されていると、TWI従装置状態レジスタA(TWSSRA)の**アドレス/停止割り込み要求フラグ(TWASIF)**が設定(1)される時にTWI割り込みが生成されます。
- **ビット3 – TWEN : 2線インターフェース許可 (Two-Wire Interface Enable)**  
このビットが設定(1)されると、従装置2線インターフェースが許可されます。
- **ビット2 – TWSIE : TWI停止条件割り込み許可 (TWI Stop Interrupt Enable)**  
停止条件割り込み許可(TWSIE)ビットの設定(1)は**停止条件**が検出された時にTWI従装置状態レジスタA(TWSSRA)の**アドレス/停止割り込み要求フラグ(TWASIF)**を設定(1)します。
- **ビット1 – TWPME : TWI無差別動作許可 (TWI Promiscuous Mode Enable)**  
このビットが設定(1)されると、従装置アドレス一致論理回路は全ての受信アドレスに応答します。このビットが解除(0)されると、アドレス一致論理回路は自身のアドレスとしてどのアドレスを認証するかを決めるのに**TWI従装置アドレス(TWSA)レジスタ**を使います。
- **ビット0 – TWSME : TWI簡便動作許可 (TWI Smart Mode Enable)**  
このビットが設定(1)されると、TWI従装置は**TWI従装置データ(TWSD)レジスタ**読み込み直後に応答動作が送出される簡便動作に移行します。応答動作は**TWI従装置制御レジスタB(TWSCR B)**の**応答動作(TWAA)ビット**によって定義されます。  
このビットが解除(0)されると、応答動作はTWSCR Bの**TWI指令(TWCMDn)ビット**が'1x'に書かれた後で送出されます。

#### 17.5.2. TWSCR B – TWI 従装置制御レジスタB (TWI Slave Control Register B)

ビット (\$B9)	7	6	5	4	3	2	1	0	TWSCR B
	–	–	–	–	–	TWAA	TWCMD1	TWCMD0	
Read/Write	R	R	R	R	R	R/W	W	W	
初期値	0	0	0	0	0	0	0	0	

- **ビット7~3 – 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2 – TWAA : TWI応答動作 (TWI Acknowledge Action)

このビットは主装置からアドレスまたはデータバイトが受信された後の従装置の応用の動きを定義します。TWI従装置制御レジスタA(TWSCRA)のTWI簡便動作許可(TWSME)ビットに依存して、応答動作はTWI指令(TWCMDn)ビットへ有効な指令が書かれた時か、またはデータレジスタ(TWSD)が読まれた時のどちらかで実行されます。応答動作はアドレス一致後にTWIアドレス/停止割り込み要求フラグ(TWASIF)を、または主装置送信中にTWIデータ割り込み要求フラグ(TWDIF)を解除(0)する場合にも実行されます。詳細については表67をご覧ください。

表67. TWI従装置の応答動作

TWAA	動作	TWSME	実行時
0	ACK送出	0	TWCMDnビットが10または11を書かれる時
		1	TWSDが読まれる時
1	NACK送出	0	TWCMDnビットが10または11を書かれる時
		1	TWSDが読まれる時

● ビット1,0 – TWCMD1,0 : TWI指令 (TWI Command)

これらのビット書き込みは表68.によって定義されるような従装置動作を起動します。動作の形式はTWI従装置割り込みフラグのTWIアドレス/停止割り込み要求フラグ(TWASIF)とTWIデータ割り込み要求フラグ(TWDIF)に依存します。応答動作は従装置が主装置からデータバイトまたはアドレスバイトを受信する時にだけ実行されます。

表68. TWI従装置指令

TWCMD1,0	TWDIR	動作
0 0	x	動作なし
0 1	x	(予約)
1 0	完全な転送処理に使用	
	0	応答動作を実行し、そして何れかの開始条件または再送開始条件を待ちます。
	1	何れかの開始条件または再送開始条件を待ちます。
1 1	アドレスバイト(アドレス/停止割り込み要求フラグ(TWASIF)=1)への応答で使用	
	0	応答動作を実行し、そして次バイトを受信します。
	1	応答動作を実行し、そしてデータ割り込み要求フラグ(TWDIF)を設定(1)します。
	データバイト(データ割り込み要求フラグ(TWDIF)=1)への応答で使用	
	0	応答動作を実行し、そして次バイトを受信します。
	1	動作なし

TWCMDnビット書き込みはSCL線を自動的に開放してTWIクロック保持(TWCH)ビットと従装置割り込みを解除(0)します。TWI応答動作(TWAA)とTWCMDnのビットは同時に書くことができます。その後、指令が起動される前に応答動作が実行されます。TWCMDnビットは瞬時発行され(スロープ信号)、常に0を読みます。

17.5.3. TWSSRA – TWI 従装置状態レジスタ (TWI Slave Status Register A)

ビット (\$BA)	7	6	5	4	3	2	1	0	TWSSRA
	TWDIF	TWASIF	TWCH	TWRA	TWC	TWBE	TWDIR	TWAS	
Read/Write	R/W	R/W	R	R	R/W	R/W	R	R	
初期値	0	0	0	0	0	0	0	0	

● ビット7 – TWDIF : TWIデータ割り込み要求フラグ (TWI Data Interrupt Flag)

このフラグはデータバイトが成功裏に受信された、換言すると動作中に衝突またはバス異常が起きない時に設定(1)されます。このフラグが設定(1)されると、従装置はSCL線をLowに強制し、TWIクロック周期を引き伸ばします。SCL線はこの割り込み要求フラグの解除(0)によって開放されます。

このビットへの1書き込みはこのフラグを解除(0)します。このフラグはTWI従装置制御レジスタB(TWSCRb)のTWI指令(TWCMDn)ビットへの有効な指令書き込み時にも自動的に解除(0)されます。

● ビット6 – TWASIF : TWIアドレス/停止割り込み要求フラグ (TWI Address/Stop Interrupt Flag)

このフラグは有効なアドレスが受信されたことを従装置が検知した時、または送信衝突が検出された時に設定(1)されます。このフラグが設定(1)されると、従装置はSCL線をLowに強制し、TWIクロック周期を引き伸ばします。SCL線はこの割り込み要求フラグの解除(0)によって開放されます。

TWI従装置制御レジスタA(TWSCRA)のTWIアドレス/停止割り込み許可(TWASIE)ビットが設定(1)なら、バス上の停止条件もTWASIFを設定(1)します。停止条件と開始条件間の最小バス開放時間よりもシステムクロックが速い場合に停止条件がこのフラグを設定(1)します。

このビットへの1書き込みはこのフラグを解除(0)します。このフラグはTWI従装置制御レジスタB(TWSCRb)のTWI指令(TWCMDn)ビットへの有効な指令書き込み時にも自動的に解除(0)されます。

#### ● ビット5 – TWCH : TWIクロック保持 (TWI Clock Hold)

このビットは従装置がSCL線をLowに保持する時に設定(1)されます。

このビットは読み込み専用で、データ割り込み要求フラグ(TWDIF)またはアドレス/停止割り込み要求フラグ(TWASIF)が設定(1)される時に設定(1)されます。このビットは割り込み要求フラグの解除(0)とSCL線の開放によって間接的に解除(0)することができます。

#### ● ビット4 – TWRA : TWI受信応答 (TWI Receive Acknowledge)

このビットは主装置からの最終受信応答ビットを含みます。

このビットは読み込み専用です。0の時は主装置からの最終応答ビットが確認応答(ACK)で、1の時は最終応答ビットが否認応答(NACK)です。

#### ● ビット3 – TWC : TWI衝突 (TWI Collision)

このビットは従装置がHighのデータビット、否認応答(NACK)ビットを転送できなかった時に設定(1)されます。衝突が検出された場合、従装置は通常動作を開始し、データと応答の出力を禁止します。Low値はSDA線上に全く移動出力されません。

このビットはこれへの1書き込みによって解除(0)されます。このビットは開始条件または再送開始条件が検出される時にも自動的に解除(0)されます。

#### ● ビット2 – TWBE : TWIバス異常 (TWI Bus Error)

このビットは転送中に不正なバス状態が起きた場合に設定(1)されます。不正なバス状態は再送開始条件または停止条件が検出され、直前の開始条件からのビット数が9の倍数でない場合に起きます。

このビットはこれへの1書き込みによって解除(0)されます。

検出されるべきバス異常に関して、システムクロック周波数はSCL周波数の最低4倍でなければなりません(訳注:共通性から本行追加)。

#### ● ビット1 – TWDIR : TWI読み/書き方向 (TWI Read/Write Direction)

このビットは主装置から受信した最後のアドレスパケットからの方向ビットを示します。このビットが1の時は主装置読み込み動作が進行中です。このビットが0の時は主装置書き込み動作が進行中です。

#### ● ビット0 – TWAS : TWIアドレス/停止条件 (TWI Address or Stop)

このビットはアドレス/停止割り込み要求フラグ(TWASIF)が最後に設定(1)されたのが何故かを示します。0なら、停止条件が引き起こしたTWASIFの設定(1)です。1なら、アドレス検出が引き起こしたTWASIFの設定(1)です。

### 17.5.4. TWSA – TWI 従装置アドレスレジスタ (TWI Slave Address Register)

ビット (\$BC)	7	6	5	4	3	2	1	0	TWSA
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	

従装置アドレスレジスタは主装置が従装置をアドレス指定しているかを判断するための従装置アドレス一致論理回路によって使われるTWI従装置アドレスを含みます。7または10ビットのアドレス認証形態使用時、上位7ビット(TWSA7~1)が従装置アドレスを表し、最下位ビット(TWSA0)は一斉呼び出しアドレス認証に使われません。TWSA0の設定(1)は一斉呼び出しアドレス認証論理回路を許可します。

10ビットアドレス使用時、アドレス一致論理回路は10ビットアドレスの最初のバイトのハードウェアアドレス認証を支援するだけです。TWSA7~1='11110nn'設定で、'nn'は従装置アドレスのビット9と8を表します。次の受信バイトが10ビットアドレスのビット7~0で、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。

有効なアドレスバイトが受信されたことをアドレス一致論理回路が検知すると、アドレス/停止割り込み要求フラグ(TWASIF)が設定(1)され、読み/書き方向(TWDIR)フラグが更新されます。

従装置制御レジスタA(TWSCRA)の無差別動作許可(TWPME)ビットが設定(1)なら、アドレス一致論理回路はTWIバス上に送信された全てのアドレスに応答します。TWSAレジスタはこの動作形態で使われません。



### 17.5.5. TWSD – TWI 従装置データレジスタ (TWI Slave Data Register)

ビット (\$BD)	7	6	5	4	3	2	1	0	TWSD
	TWSD7	TWSD6	TWSD5	TWSD4	TWSD3	TWSD2	TWSD1	TWSD0	
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	

データ(TWSD)レジスタはデータを送受信する時に使われます。データ転送中、データはバスとTWSDレジスタ間で移動されます。従って、データレジスタはバイト転送中にアクセスできません。これはハードウェアで保護されています。データレジスタは従装置によってSCL線がLowを保持している時、換言すると**クロック保持(TWCH)フラグ**が設定(1)されている時にだけアクセスできます。

主装置が従装置からデータを読む時に送出すべきデータがTWSDレジスタに書かれなければなりません。バイト転送は主装置が従装置からデータバイトのクロック駆動を始める時に開始されます。続いて従装置は主装置からの応答ビットを受信します。そして**データ割り込み要求フラグ(TWDIF)**とTWCHが設定(1)されます。

主装置が従装置にデータを書く時に、1バイトがデータレジスタに受信された時にTWDIFとTWCHが設定(1)されます。**簡便動作**が許可の場合、データレジスタ読み込みは**TWI従装置制御レジスタB(TWSCR)**の**応答動作(TWAA)ビット**によって設定されるようにバス操作を起動します。

簡便動作でのTWSDのアクセスは従装置割り込み要求フラグとTWCHフラグを解除(0)します。

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

ビット (\$BB)	7	6	5	4	3	2	1	0	TWSAM
	TWSAM7	TWSAM6	TWSAM5	TWSAM4	TWSAM3	TWSAM2	TWSAM1	TWAE	
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7~1 – ADDRMask7~1 : TWIアドレス遮蔽 (TWI Address Mask)

これらのビットはアドレス許可(TWAE)ビット設定に依存して、第2アドレス一致レジスタまたはアドレス遮蔽レジスタとして働きます。

TWAEが0に設定される場合、TWSAMは7ビットの従装置アドレス遮蔽を格納することができます。TWSAM内の各ビットは**従装置アドレス(TWSA)レジスタ**内の対応するアドレスビットを遮蔽(禁止)することができます。遮蔽ビットが1の場合に到着アドレスビットとTWSA内の対応するビット間のアドレス一致が無視されます。換言すると、遮蔽されたビットは常に一致します。

TWAEが1に設定される場合、TWSAMはTWSAレジスタに加えて第2従装置アドレスを格納することができます。この動作形態では従装置がTWSAの1つとTWSAMの別の1つで2つの独自のアドレスに一致します。

#### ● ビット0 – TWAE : TWIアドレス許可 (TWI Address Enable)

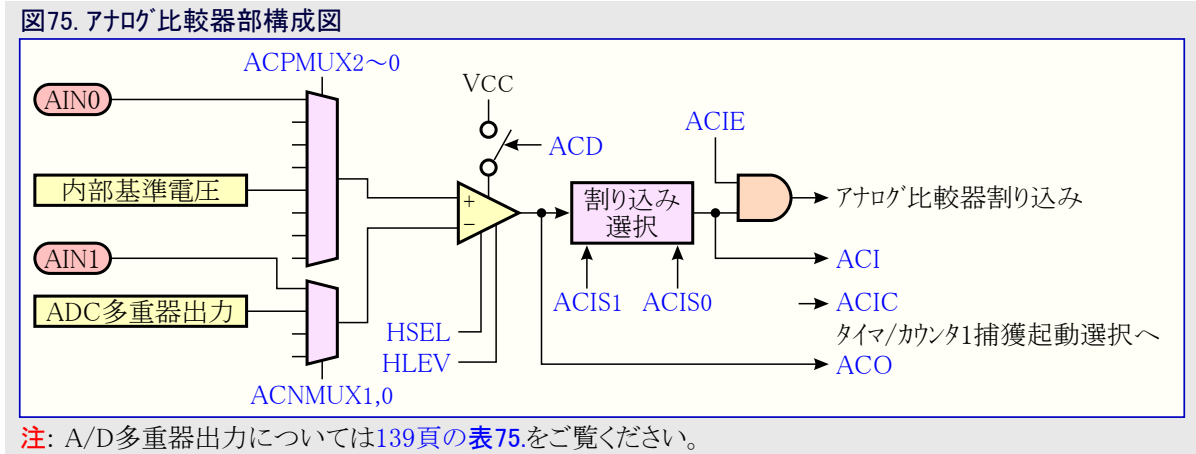
既定によるこのビットは0で、TWSAMビットはTWSAレジスタに対するアドレス遮蔽として働きます。このビットが1に設定される場合、従装置アドレス一致論理回路はTWSAとTWSAM内の2つの独自のアドレスに回答します。



## 18. 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。AIN0ピンの電圧がAIN1ピンの電圧よりも高い時にアナログ比較器制御/状態レジスタ(ACSRA)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75.で示されます。



ピン配置については2頁の「ピン配置」をご覧ください。

アナログ比較器への正入力信号に対して以下の任意選択が利用可能です。

- AIN0ピン
- ハットキャップ内部基準電圧

アナログ比較器への負入力信号に対して以下の任意選択が利用可能です。

- AIN1ピン
- A/D変換器(ADC)多重器出力

ADC入力多重器を使うにはA/D変換器電力削減(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)内のPRADCビットを解除(0)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26頁の「PRR - 電力削減レジスタ」をご覧ください。

### 18.1. アナログ比較器用レジスタ

#### 18.1.1. ACSRA - 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	ACPMUX2	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSRA
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)することができます。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ACSRA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、このビットが変更される時に割り込みが起り得ます。

##### ● ビット6 - ACPMUX2 : アナログ比較器正入力多重器 (Analog Comparator Positive Input Multiplexer)

ACPMUX1とACPMUX0と共にこれらのビットはアナログ比較器の正入力に対する供給元を選びます。次頁の「ACSRB - アナログ比較器制御/状態レジスタB」をご覧ください。

##### ● ビット5 - ACO : アナログ比較器出力 (Analog Comparator Output)

アナログ比較器の出力は同期化され、その後直接ACOへ接続されます。この同期化は1~2クロック周期の遅延をもたらします。

##### ● ビット4 - ACI : アナログ比較器割り込み要求フラグ (Analog Comparator Interrupt Flag)

このビットは比較器出力での出来事がACSRA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。ACSRA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)

このビットが設定(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が活性(有効)にされます。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。

●ビット2 – ACIC : アナログ比較器捕獲起動許可 (Analog Comparator Input Capture Enable)

このビットが設定(1)されると、タイマ/カウンタ1の捕獲機能はアナログ比較器によって起動することができます。そして比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音消去機能とエッジ選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。比較器にタイマ/カウンタ1捕獲割り込みを起動させるには、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(87頁の「TIMSK1 – タイマ/カウンタ1割り込み許可レジスタ」をご覧ください)。

このビットが解除(0)されると、アナログ比較器と捕獲機能間の接続は全く存在しません。

●ビット1,0 – ACIS1,0 : アナログ比較器割り込み条件 (Analog Comparator Interrupt Mode Select)

これらのビットは比較器のどの事象がアナログ比較器割り込みを起動するのかを決めます。各種設定は表69.で示されます。

これらのビットを変更する時にアナログ比較器割り込みは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69. ACIS1,0設定

ACIS1	ACIS0	割り込み形態
0	0	比較器出力の変移(トグル)
0	1	(予約)
1	0	比較器出力の下降端
1	1	比較器出力の上昇端

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

ビット	7	6	5	4	3	2	1	0	
\$2F (\$4F)	HSEL	HLEV	–	–	ACNMUX1	ACNMUX0	ACPMUX1	ACPMUX0	ACSRB
Read/Write	R/W	R/W	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7 – HSEL : ヒステリシス許可 (Hysteresis Select)

このビットが論理1を書かれると、アナログ比較器のヒステリシスが許可されます。ヒステリシスレベルはHLEVビットによって選択されます。

●ビット6 – HLEV : ヒステリシス レベル選択 (Hysteresis Level)

ヒステリシス許可(HSEL)ビット経由で許可されると、ヒステリシスレベルは表70.で示されるようにHLEVビットを使って設定することができます。

表70. アナログ比較器ヒステリシス レベル選択

HSEL	HLEV	アナログ比較器のヒステリシス
0	x	不許可
1	0	20mV
	1	50mV

●ビット5,4 – 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 – ACNMUX1,0 : アナログ比較器負入力多重器 (Analog Comparator Negative Input Multiplexer)

これらのビットは右の表71.で示されるように、アナログ比較器の負入力に対する供給元を選びます。

表71. アナログ比較器負入力用供給元選択

ACNMUX1	ACNMUX0	アナログ比較器負入力
0	0	AIN1ピン
0	1	A/D変換器(ADC)多重器出力
1	0	(予約)
1	1	(予約)

●ビット1,0 – ACPMUX1,0 : アナログ比較器正入力多重器 (Analog Comparator Positive Input Multiplexer)

ACPMUX2と共に、これらのビットは右の表72.で示されるように、アナログ比較器の正入力に対する供給元を選びます。

表72. アナログ比較器正入力用供給元選択

ACPMUX2	ACPMUX1	ACPMUX0	アナログ比較器正入力
0	0	0	AIN0ピン
0	0	1	(予約)
0	1	0	(予約)
0	1	1	(予約)
1	0	0	内部バンドキャップ基準電圧
1	0	1	(予約)
1	1	0	(予約)
1	1	1	(予約)

### 18.1.3. 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	DIDR0
	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	
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2 – ADC2D : ADC2/AIN1デジタル入力緩衝部禁止 (ADC2/AIN1 Digital input buffer disable)**

アナログ入力として使うけれどもデジタル入力として必要としない時に、このビットに論理1を書くことによってデジタル入力緩衝部の電力消費を減らすことができます。このビットが設定(1)されると、AIN1ピンのデジタル入力緩衝部が禁止され、対応するピンレジスタビット(PA2)は常に0を読みます。

- **ビット1 – ADC1D : ADC1/AIN0デジタル入力緩衝部禁止 (ADC1/AIN0 Digital input buffer disable)**

アナログ入力として使うけれどもデジタル入力として必要としない時に、このビットに論理1を書くことによってデジタル入力緩衝部の電力消費を減らすことができます。このビットが設定(1)されると、AIN0ピンのデジタル入力緩衝部が禁止され、対応するピンレジスタビット(PA1)は常に0を読みます。

## 19. A/D変換器

### 19.1. 特徴

- 10ビット分解能
- 積分非直線性誤差1LSB
- 絶対精度±2LSB
- 15μsの変換時間
- 最大分解能で15kSPS(採取/s)
- 28チャンネルのシングルエンド入力多重器内蔵
- 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
- 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
- 休止動作雑音低減機能

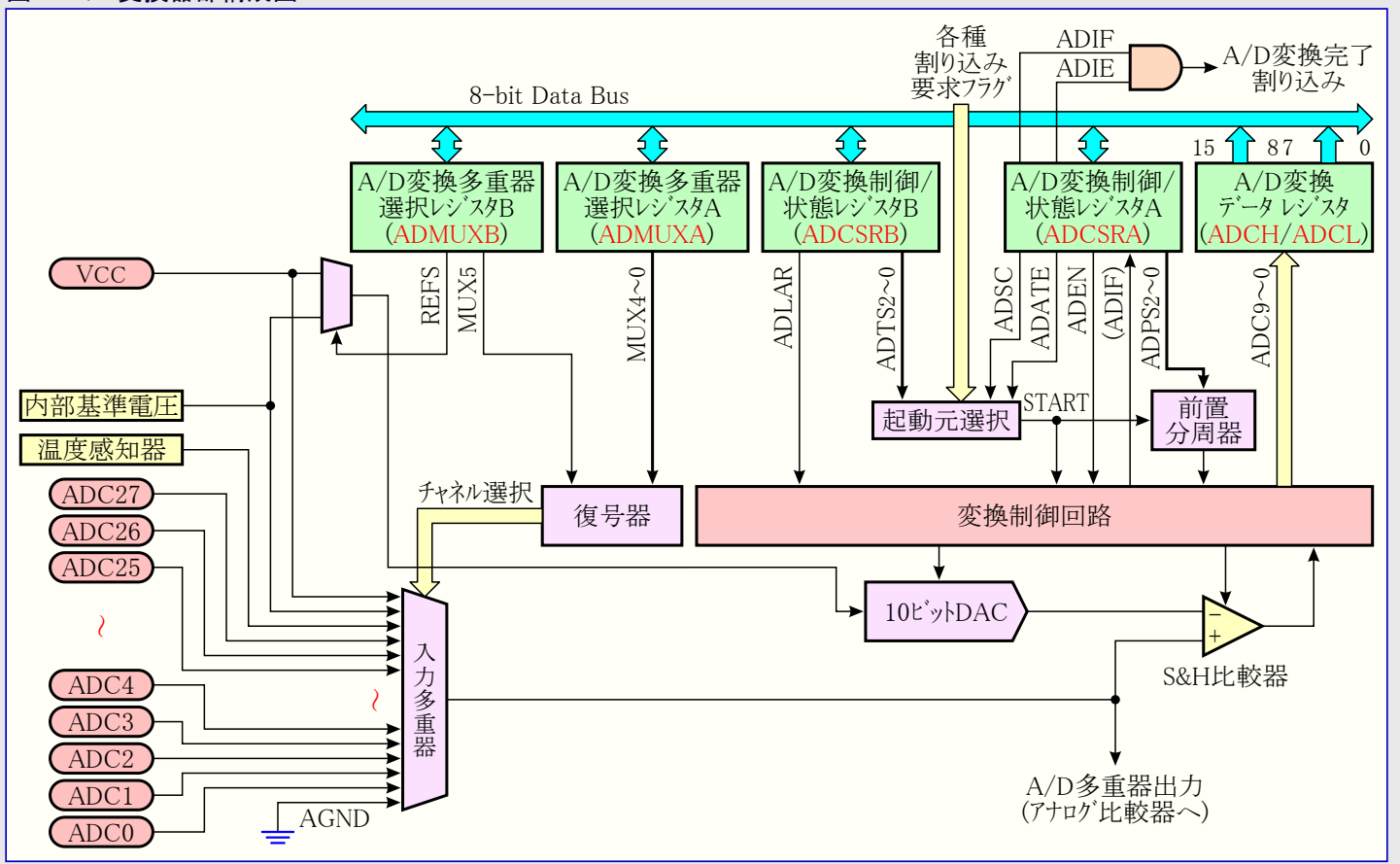
### 19.2. 概要

ATtiny828は10ビット逐次比較A/D変換器(ADC)が特徴です。ADCは28のシングルエンド入力ピンと、内部温度感知器、内部基準電圧、アナログGND、供給電源電圧から来るシングルエンド電圧チャンネルの4つの内部からの電圧測定をADCに許す32チャンネルのアナログ多重器に配線されます。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76.で示されます。

公称1.1Vの内部基準電圧がチップ上で提供されます。代わりにシングルエンドチャンネルに対する基準電圧としてVCCを使うことができます。

図76. A/D変換器部構成図



### 19.3. 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26頁の「PRR - 電力削減レジスタ」をご覧ください。

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、節電をする休止動作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B(ADMUXB)の基準電圧選択(REFS)ビット書き込みによって選択されます。選択はVCC供給ピンと1.1V内部基準電圧です。

アナログ入力チャネルはA/D多重器選択レジスタ(ADMUXAとADMUXB)のチャネル選択(MUX5~0)ビットへの書き込みによって選択されます。GNDと固定基準電圧(1.1V内蔵基準電圧(V<sub>BG</sub>))だけでなく、どのADC入力ピンもA/D変換器のシングルエンド入力として選択できます。

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、ADCSR<sub>B</sub>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る。さもなければ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。

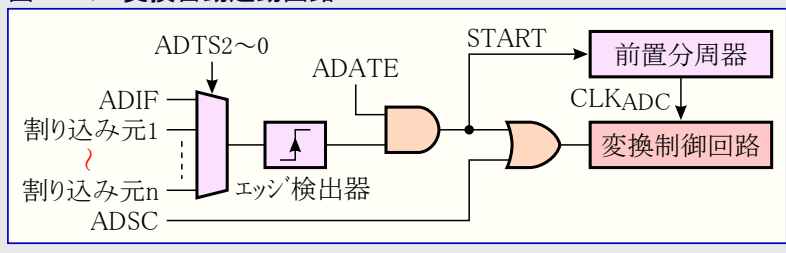
A/D変換部には変換完了時に起動できる自身の割り込みがあります。A/DデータレジスタへのA/D変換器アクセスがADCLとADCHの読み込み間で禁止されている場合、例えその変換結果が失われても割り込みは起動します。

### 19.4. 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(26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ネルが選択されると、A/D変換部はそのチャ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(ADCSR<sub>B</sub>)のA/D変換起動元選択(ADTS<sub>2~0</sub>)ビットの設定によって選択されます(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)。選択した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(1)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77. 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かかわらず、変換中は1として読めます。

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137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)



## 19.5. 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78.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。

通常の変換は次頁の表73.で要約されるように、15変換クロック周期で行われます。下の図79.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26変換クロック周期で行われます。

図78. A/D変換前置分周器部構成

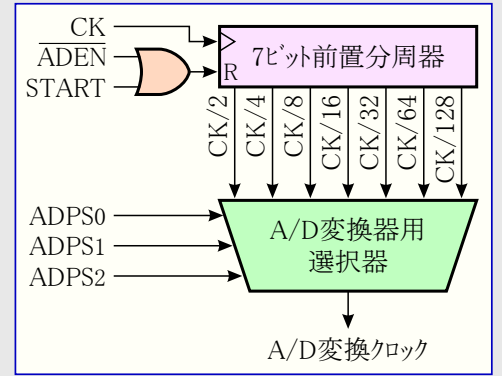
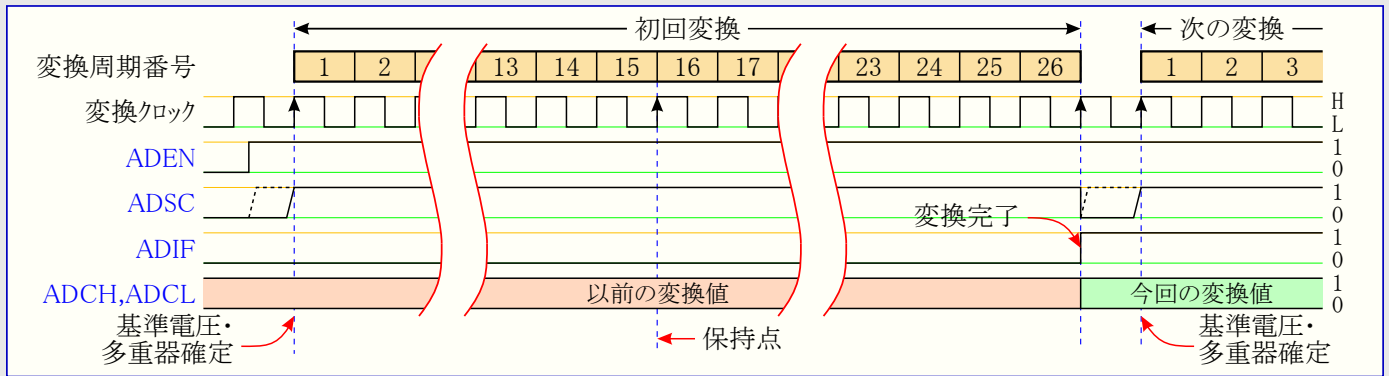
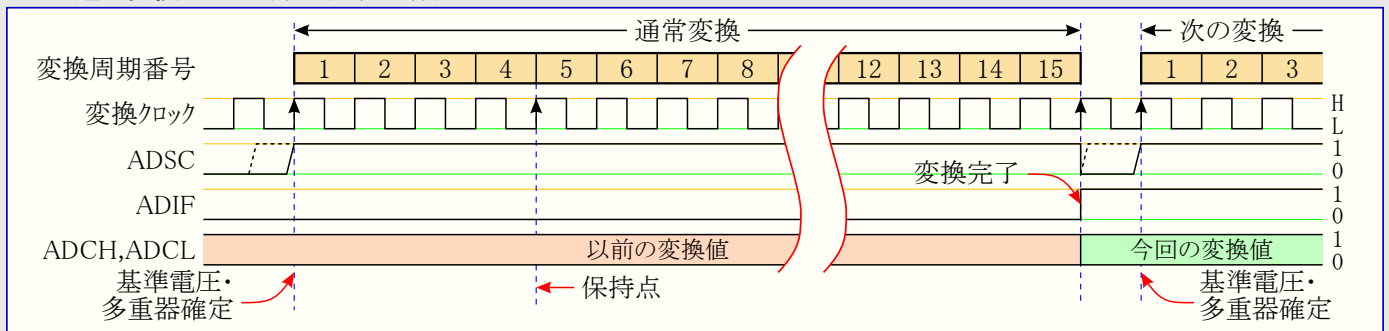


図79. 初回変換タイミング (単独変換動作)



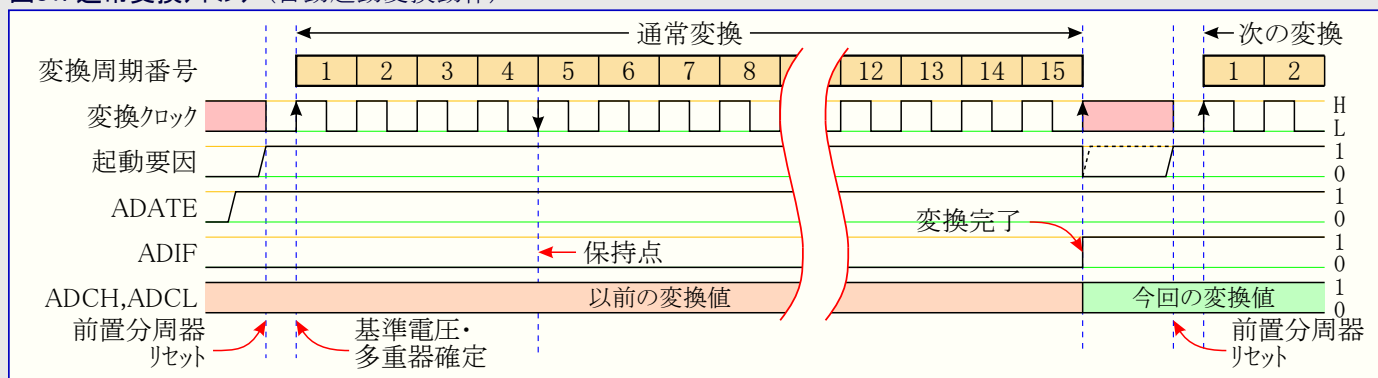
実際の採取&保持(保持開始点)は通常変換の開始後4変換クロック周期、初回変換の開始後15変換クロック周期で行われます。図80.をご覧ください。変換が完了すると、結果がA/Dデータレジスタ(ADCH,ADCL)に書かれ、ADCSRAのA/D変換完了割り込み要求フラグ(ADIF)が設定(1)されます。単独変換動作(ADATE=0)では同時にADCSRAのA/D変換開始(ADSC)ビットが解除(0)されます。その後にソフトウェアは再びADSCを設定(1)でき、新規変換は変換クロックの最初の上昇端で開始されます。

図80. 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、下の図81.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、4.5変換クロック周期で採取が行われます。同期化論理回路(エッジ検出器)に対して追加の2CPUクロック周期が費やされます。

図81. 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82.をご覧ください。

変換時間の要約については表73.をご覧ください。

図82. 連続変換動作タイミング

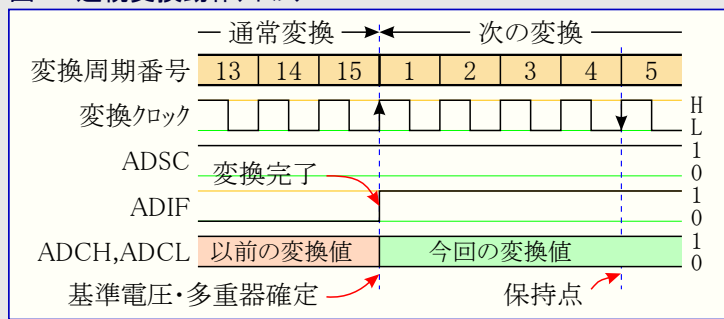


表73. A/D変換時間

変換種別	保持点	変換時間
初回変換	15	26
通常変換	4	15
自動起動変換	4.5	15.5
連続変換	4	15

注: 保持点の値は変換開始からの変換クロック数です。

## 19.6. チャネル変更と基準電圧選択

A/D多重器選択レジスタ(ADMUXA,ADMUXB)のチャネル選択(MUX5~0)ビットと基準電圧選択(REFS)ビットはCPUが不規則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)前の最後の交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ADMUXAとADMUXBの更新時に特別な注意が祓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ADCSRA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。ADCSRA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つで更新すると、新設定は次のA/D変換に影響を及ぼします。

### 19.6.1. 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択さ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択してください。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選択を変更する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択してください。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、その後にチャネル選択を変更することです。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。

## 19.6.2. 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または1.1V内部基準電圧のどちらかとして選択できます。1.1V内部基準電圧は内部増幅器を通して内部ハントキャップ基準電圧(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。

## 19.7. 雑音低減機能

A/D変換部は**休止動作**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動作へ移行する時に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、このような休止動作へ移行する前にADENへ**0**を書くことが推奨されます。

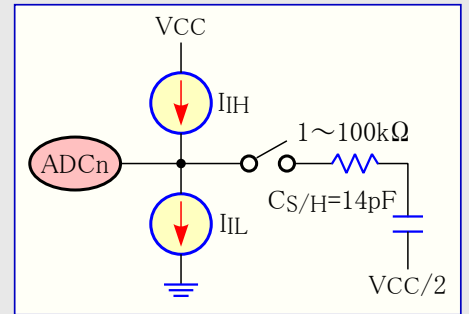
## 19.8. アナログ入力回路

シングルエンド入力チャネルのアナログ回路は**図83**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入力として選択さ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。そのチャネルが選択されると、(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。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、使用者は緩やかに変化する信号を持つ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83. 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。

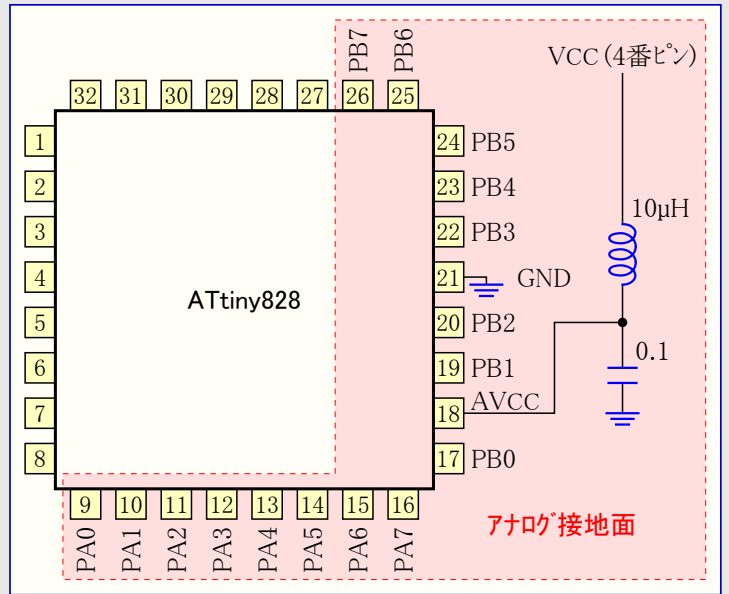
## 19.9. アナログ雑音低減技術

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。変換精度が重要な時は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に、これらを変換進行中に決して切り替えてはなりません。
- 可能な限りVCCとGNDピンの近くに、パスコンを配置してください。
- アナログ供給電圧(AVCC)ピンは**図84**で示されるようにLC網を経由してデジタル供給電圧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時は「**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。正しい外部パ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84. A/D変換部電源接続



## 19.10. 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。最低値コードは0として読み、最高値コードは $2^n-1$ 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。

### ● オフセット誤差 - 図85.

最初の遷移点(\$000から\$001)に於いて理想遷移点(差0.5 LSB)と比較した偏差です。理想値は0LSBです。

### ● 利得誤差 - 図86.

オフセット誤差補正後の最後の遷移点(\$3FEから\$3FF)に於いて理想遷移点(最大差1.5LSB以下)と比較した偏差です。理想値は0LSBです。

### ● 積分非直線性誤差 (INL) - 図87.

オフセット誤差と利得誤差補正後の全ての遷移点に於いて理想遷移点と比較した最大偏差です。理想値は0LSBです。

### ● 微分非直線性誤差 (DNL) - 図88.

実際のコードの幅(隣接する2つの遷移点間)に於いて理想コード幅(1LSB)と比較した最大偏差です。理想値は0LSBです。

### ● 量子化誤差

有限数のコードで入力電圧を量子化するため、1LSB幅となる入力電圧範囲は同じ値のコードになります。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に於いて理想遷移点と比較した最大偏差です。これは、オフセット誤差、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85. オフセット誤差

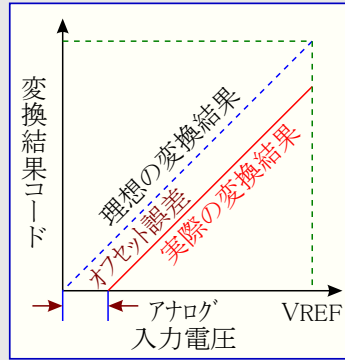


図86. 利得誤差

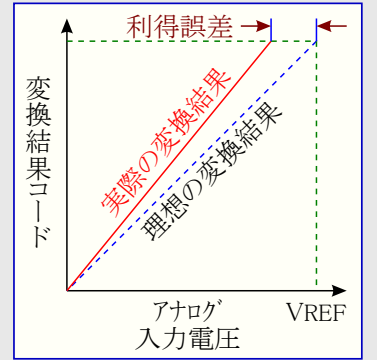


図87. 積分非直線性誤差

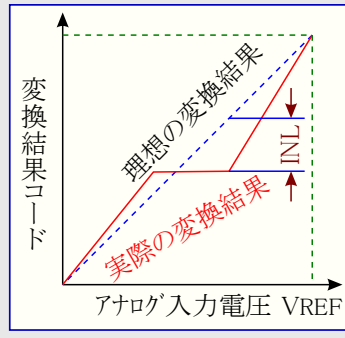
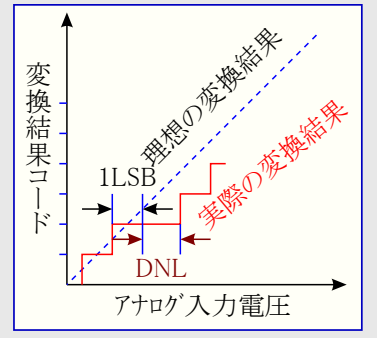


図88. 微分非直線性誤差



## 19.11. 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。結果は右のとおりです。ここで $V_{IN}$ は選択した入力ピンの電圧で、 $V_{REF}$ は選択した基準電圧です(139頁の表75と表76をご覧ください)。\$000はアナログGNDを表し、\$3FFは選択した基準電圧-1LSBを表します。結果は\$03FF~\$0000の片側形式で表されます(訳補:±ではありません)。

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

## 19.12. 温度測定

温度測定はシングルエンドADCチャンネルに連結されるチップ上の温度感知器に基きます。温度感知器はA/Dチャンネル選択(MUX)ビット書き込みによって許可されます。温度感知器測定でのADC基準電圧供給元に内部1.1V基準電圧も選択されなければなりません。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を測定するのに単独変換動作(ADATE=0)で使用することができます。

測定した電圧は表74で記載されるように温度に対して直線関係を持ちます。感度は概ね1LSB/°Cで、精度は使用者校正の方法に依存します。代表的に、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74で記載した値は代表値です。けれども、製法変化のため、温度感知器の出力電圧は或るチップと別のチップで変わります。より正確な結果を達成するため、温度測定は応用ソフトウェアで校正することができます。校正ソフトウェアは次式を使って行なうことができます。

表74. 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)	235	300	360

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

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。より高い精度が必要とされる処では、2つの温度での測定に基づいて傾斜計数が評価されるべきです。



## 19.13. A/D変換用レジスタ

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

ビット (\$7C)	7	6	5	4	3	2	1	0	
	-	-	-	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUXA
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4~0 – MUX4~0 : A/Dチャンネル選択 (Analog Channel Selectin Bits)

これらのビットはA/D多重器選択レジスタB(ADMUXB)のMUX5と共にどのアナログ入力がA/D変換器に接続されるかを選びます。表75をご覧ください。

表75. シングル エント'入力チャンネル選択

MUX5~0	アナログ入力チャンネル	MUX5~0	アナログ入力チャンネル	MUX5~0	アナログ入力チャンネル	MUX5~0	アナログ入力チャンネル
00 0000	ADC0 (PA0)	00 1000	ADC8 (PB0)	01 0000	ADC16 (PC0)	01 1000	ADC24 (PD0)
00 0001	ADC1 (PA1)	00 1001	ADC9 (PB1)	01 0001	ADC17 (PC1)	01 1001	ADC25 (PD1)
00 0010	ADC2 (PA2)	00 1010	ADC10 (PB2)	01 0010	ADC18 (PC2)	01 1010	ADC26 (PD2)
00 0011	ADC3 (PA3)	00 1011	ADC11 (PB3)	01 0011	ADC19 (PC3)	01 1011	ADC27 (PD3)
00 0100	ADC4 (PA4)	00 1100	ADC12 (PB4)	01 0100	ADC20 (PC4)	01 1100	0V (GND)
00 0101	ADC5 (PA5)	00 1101	ADC13 (PB5)	01 0101	ADC21 (PC5)	01 1101	1.1V (V <sub>BG</sub> ) (注1)
00 0110	ADC6 (PA6)	00 1110	ADC14 (PB6)	01 0110	ADC22 (PC6)	01 1110	温度感知器 (注2)
00 0111	ADC7 (PA7)	00 1111	ADC15 (PB7)	01 0111	ADC23 (PC7)	01 1111	供給電圧(VCC)
1x xxxx	(予約) (未接続)						

注1: 内部基準電圧への切り換え後、A/D変換器は測定が安定する前に1msの安定時間が必要です。これ以前に開始された変換は信頼できません。A/D変換器は安定時間中に許可されなければなりません。

注2: 138頁の「温度測定」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。

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

ビット (\$7D)	7	6	5	4	3	2	1	0	
	-	-	REFS	-	-	-	-	MUX5	ADMUXB
Read/Write	R	R	R/W	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,6,4~1 – Res : 予約 (Reserved Bits)

これらのビットは予約されており、常に0を読みます。

- ビット5 – REFS : 基準電圧選択 (Reference Selection Bit)

このビットは表76.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76. A/D変換部の基準電圧選択

REFS	基準電圧
0	アナログ基準電圧としてVCC使用
1	内部1.1V基準電圧

- ビット0 – MUX5 : A/Dチャンネル選択 (Analog Channel Selectin Bit)

このビットはA/D多重器選択レジスタA(ADMUXA)のMUX4~0と共にどのアナログ入力がA/D変換器に接続されるかを選びます。表75.をご覧ください。

### 19.13.3. 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15に代わって26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**として読めます。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変換器は選択した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変換起動元選択(ADTS2~0)ビット設定によって選択されます。

● **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(ADIE)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。

● **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**が書かれ、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77. A/D変換器前置分周器選択

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
分周係数	2	2	4	8	16	32	64	128

19.13.4. 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	
	-	-	-	-	ADLAR	ADTS2	ADTS1	ADTS0	ADCSR
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 Bits)**

これらは予約ビットです。将来のデバイスとの共通性のため、常にこれらのビットへ**0**を書いてください。

● **ビット3 – 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次頁の「A/Dデータレジスタ」をご覧ください。

● **ビット2~0 – ADTS2~0 : A/D変換自動起動元選択 (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**が書かれると、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。変換は選択した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78. A/D変換自動起動元選択

ADTS2	ADTS1	ADTS0	起動元	ADTS2	ADTS1	ADTS0	起動元
0	0	0	連続変換動作	1	0	0	タイマ/カウンタ0溢れ
0	0	1	アナログ比較器	1	0	1	タイマ/カウンタ1比較B一致
0	1	0	外部割り込み要求0	1	1	0	タイマ/カウンタ1溢れ
0	1	1	タイマ/カウンタ0比較A一致	1	1	1	タイマ/カウンタ1捕獲要求

### 19.13.5. 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									
ビット (\$79)	15	14	13	12	11	10	9	8	
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
ビット (\$78)	7	6	5	4	3	2	1	0	
	ADC1	ADC0	-	-	-	-	-	-	ADCL

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変換制御/状態レジスタB(ADCSRB)の左揃え選択(ADLAR)ビットとA/D多重器選択レジスタ(ADMUXA,ADMUXB)のA/Dチャンネル選択(MUX5~0)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。

#### ● ADC9~0 : A/D変換結果 (ADC Conversion result)

これらのビットは138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。

### 19.13.6. 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)

アナログ信号がADCnに印加され、そのピンのデジタル入力が必要ない時に、消費電力を減らすためにADCnDビットが設定(1)されるべきです。ADCnDビット設定(1)は対応する(ADCn)ピンのデジタル入力緩衝部を禁止します。ADCnD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(PINxn)は常に0を読みます。

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

ビット (\$7F)	7	6	5	4	3	2	1	0	
	ADC15D	ADC14D	ADC13D	ADC12D	ADC11D	ADC10D	ADC9D	ADC8D	DIDR1
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 – ADC15D~ADC8D : ADC15~8デジタル入力禁止 (ADC15~8 Digital Input Disable)

アナログ信号がADCnに印加され、そのピンのデジタル入力が必要ない時に、消費電力を減らすためにADCnDビットが設定(1)されるべきです。ADCnDビット設定(1)は対応する(ADCn)ピンのデジタル入力緩衝部を禁止します。ADCnD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(PINxn)は常に0を読みます。

### 19.13.8. DIDR2 – デジタル入力禁止レジスタ2 (Digital Input Disable Register 2)

ビット (\$DE)	7	6	5	4	3	2	1	0	
	ADC23D	ADC22D	ADC21D	ADC20D	ADC19D	ADC18D	ADC17D	ADC16D	DIDR2
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 – ADC23D~ADC16D : ADC23~16デジタル入力禁止 (ADC23~16 Digital Input Disable)

アナログ信号がADCnに印加され、そのピンのデジタル入力が必要ない時に、消費電力を減らすためにADCnDビットが設定(1)されるべきです。ADCnDビット設定(1)は対応する(ADCn)ピンのデジタル入力緩衝部を禁止します。ADCnD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(PINxn)は常に0を読みます。

### 19.13.9. DIDR3 – デジタル入力禁止レジスタ3 (Digital Input Disable Register 3)

ビット (\$DF)	7	6	5	4	3	2	1	0	
	–	–	–	–	ADC27D	ADC26D	ADC25D	ADC24D	DIDR3
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7～0 – ADC27D～ADC24D : ADC27～24デジタル入力禁止 (ADC27～24 Digital Input Disable)

アナログ信号がADCnに印加され、そのピンのデジタル入力が必要ない時に、消費電力を減らすためにADCnDビットが設定(1)されるべきです。ADCnDビット設定(1)は対応する(ADCn)ピンのデジタル入力緩衝部を禁止します。ADCnD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(PINxn)は常に0を読みます。

## 20. デバッグWIRE 内蔵デバッグ システム

### 20.1. 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

### 20.2. 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1本線の双方向インターフェースを使います。

### 20.3. 物理インターフェース

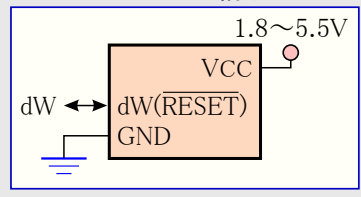
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

図89.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ヒューズで選択したクロック元です。

デバッグWIREが使われるシステム的设计時、正しい動作のために次の注意点が厳守されなければなりません。

- dW/(RESET)線のプルアップ抵抗は10k~20kΩの範囲でなければなりません。けれどもこのプルアップ抵抗は任意です。
- RESETピンのVCCへの直接的な接続では動作しません。
- RESETピンに挿入したコンデンサはデバッグWIRE使用時、切断されなければなりません。
- 全ての外部リセット元は切断されなければなりません。

図89. デバッグWIRE構成図



### 20.4. 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。

### 20.5. 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機能を正確エミュレートします。CPUが停止される時にデバッグ(AVR Studio)経由でいくつかのI/Oレジスタをアクセスする間、注意が妨げられなければなりません。この制限の詳細説明についてはデバッグWIRE資料をご覧ください。

デバッグWIREインターフェースはデバッグがシステムクロックに同期する必要があることを意味する、非同期です。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。

### 20.6. デバッグWIRE用レジスタ

次項はデバッグWIREで使うレジスタを記述します。

#### 20.6.1. 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。

## 21. ブートローダと書き込み中読み込み(Read-While-Write)を持つ自己プログラミング

### 21.1. 特徴

- MCUに应用メモリの消去、書き込み、書き換えを許す自己プログラミング
- 効率的な読み-変更-書き(リード-モディファイ-ライト)支援
- 書き込み中読める(Read-While-Write)自己プログラミングに対する支援
- 可変の大きさを持つブートローダ領域
- 更なるアクセスに対して应用メモリに安全な閉鎖を許す施錠ビット
- 高い安全と柔軟な保護の仕組みを許す独立したブート施錠ビット
- リセットベクタ設定用の独立したヒューズ
- 最適化されたフラッシュページ容量
- 効率的なコード手法

### 21.2. 概要

ブートローダ支援は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真の書き込み中の読み出しが可能な自己プログラミング機構を提供します。この特徴はフラッシュメモリに常駐するブートローダプログラムを使うMCUによって制御される柔軟な应用ソフトウェア更新を可能にします。ブートローダプログラム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。ブートローダ領域内のプログラムコードはブートローダメモリを含むフラッシュメモリ全体を書く能力を持ちます。従ってブートローダは自身をも変更でき、この機能がそれ以上必要とされないなら、そのコードから自身を消去することもできます。ブートローダメモリの容量はヒューズで設定可能で、ブートローダは個別に設定可能な2組の独立したブート施錠ビットを持ちます。これは異なる保護レベルを選択する独特な柔軟性を使用者に与えます。

### 21.3. 応用とブートローダのフラッシュ領域

フラッシュメモリは应用領域とブートローダ領域(BLS)の2つの主な領域で構成されます。図91をご覧ください。各領域の容量は145頁の表82で示されるようにBOOTSZヒューズによって設定されます。これら2つの領域は個別の施錠ビットの組を持つため、異なる保護レベルを持てます。

#### 21.3.1. 应用領域

应用領域は应用コードを格納するのに使われるフラッシュメモリの部分です。应用領域用保護レベルは应用ブート施錠ビット(ブート施錠ビット0)によって選択することができます。151頁の表87をご覧ください。

应用領域から実行される時にSPM命令が禁止されるので、应用領域はどんなブートローダコードも決して格納し得ません。

#### 21.3.2. ブートローダ領域 (BLS)

应用領域が应用コード格納用に使われるのに対して、SPM命令はブートローダ領域(BLS)から実行する時にだけプログラミングを始められるので、ブートローダソフトウェアはBLSに格納されなければなりません。

SPM命令はBLS自身を含む全てのフラッシュメモリにアクセスできます。BLS用保護レベルはブートローダ施錠ビット(ブート施錠ビット1)によって選択することができます。151頁の表88をご覧ください。

### 21.4. フラッシュメモリの書き込み中に読み出し可能な領域と不能な領域

ブートローダによってソフトウェアが更新する間、CPUは書き込み中読み込み(RWW)で動くか、または操作中に停止されるかのどちらかです。動作の形式はどちらのアドレスがプログラミングされつつあるかに依存します。

应用とブートローダの領域に加えて、フラッシュメモリは書き込み中読み出し可能な(RWW)領域と書き込み中読み出し不能な(NRWW)領域の2つの固定領域にも分けられます。RWWとNRWWの領域間の主な違いは次のとおりです。

- RWW領域側に配置されたページを消去または書く時に、NRWW領域はその動作中に読むことができます。
- NRWW領域側に配置されたページを消去または書く時に、その全ての動作中にCPUは停止されます。

ブートローダ動作中、ソフトウェアはRWW領域に置かれたコードを決して読むことができません。下をご覧ください。

表79. 書き込み中読み出し機能

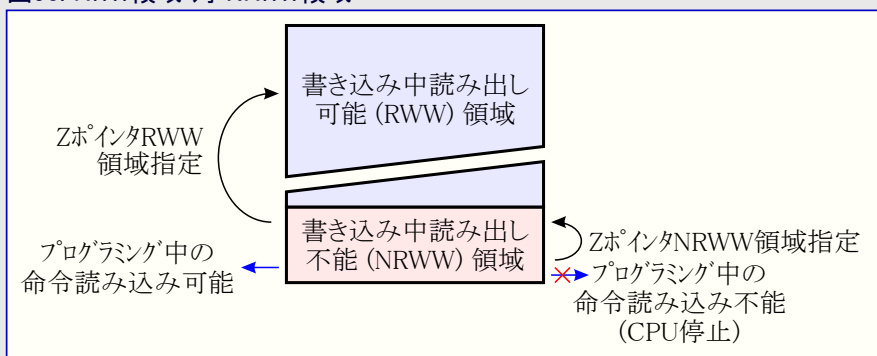
プログラミング中にZポインタで指定される領域	プログラミング中に読める領域	CPU動作	RWW機能支援
RWW領域	NRWW領域	通常動作	あり
NRWW領域	なし	停止	なし



用語「書き込み中読み込み(RWW)領域」は「ブートローダ」によってソフトウェアが更新する間に読まれるつつある領域ではなく、プログラミングされ(換言すると、消去または書かれ)つつある領域を参照します。

RWWとNRWWの領域は表82.で定義され、図90.で図解されます。

図90. RWW領域 対 NRWW領域



### 21.4.1. RWW – 書き込み中読み出し可能領域

ブートローダがRWW領域側のページをプログラミングする時にフラッシュメモリからコードを読むことが可能ですが、NRWW領域に置かれたコードだけです。

プログラミング中、ソフトウェアはRWW領域が読まれないことを保証しなければなりません。ソフトウェアがプログラミング中に(例えば、CALL, JMP, LPM系命令または割り込みによって)RWW領域側に置かれたコードを読もうとすると、そのソフトウェアは未知の状態へ行き着くかもしれません。これを避けるために割り込みは禁止、または常にNRWW領域に置かれるBLSへ移動のどちらかにされるべきです。RWW領域が読み出しに対して妨げられている限り、SPM命令制御/状態レジスタ(SPMCSR)のRWW領域多忙(RWWSB)ビットが設定(1)されます。プログラミング完了後、RWW領域に置かれたコードを読む前にRWWSBはソフトウェアによって解除(0)されなければなりません。

### 21.4.2. NRWW – 書き込み中読み出し不能領域

NRWW領域に置かれたコードはブートローダがRWW領域内のページを更新する時に読めます。ブートローダがNRWW領域を更新する時に、CPUは全てのページ消去またはページ書き込み動作中に停止されます。

## 21.5. ブートローダプログラムへの移行

ブートローダへの移行は応用プログラムから分岐(Jump)または呼び出し(Call)によって起きます。これはUSARTやSPI経由で受信した指令のような起点によって始められるかもしれませんが。代わりに、リセット後にリセットベクタがブートローダ領域開始アドレスを指示するようにブートルセット(BOOTRST)ヒューズをプログラム(0)することができ、この場合、リセット後にブートローダが開始されます。下の表80.をご覧ください。

表80. ブートルセットヒューズ

BOOTRST (注)	リセットベクタ
0	ブートローダ開始アドレス (表82.をご覧ください。)
1	応用リセットアドレス (\$0000)

注: “0”はプログラムを意味し、“1”は非プログラムを意味します。

応用コードが設定された(書かれた)後、ブートプログラムは応用コードの実行を始めることができます。

このヒューズがMCU自身によって変更できないことに注意してください。これは一旦BOOTRSTヒューズがプログラム(0)されると、リセットベクタは常にブートローダを指し、このヒューズが直列または並列のプログラミングインターフェースを通してのみ変更できることを意味します。

### 21.6. ブートローダ構成設定

フラッシュメモリの書き込み中読み出し可能な(RWW)領域と書き込み中読み出し不能な(NRWW)領域は、表81.で示されるように一定です。これら2つの領域の詳細については前頁の「応用とブートローダのフラッシュ領域」をご覧ください。

表81. フラッシュメモリのRWW領域とNRWW領域

領域	フラッシュページ数	アドレス範囲
書き込み中読み出し可能(RWW)領域	96	\$0000～\$0BFF
書き込み中読み出し不能(NRWW)領域	32	\$0C00～\$0FFF

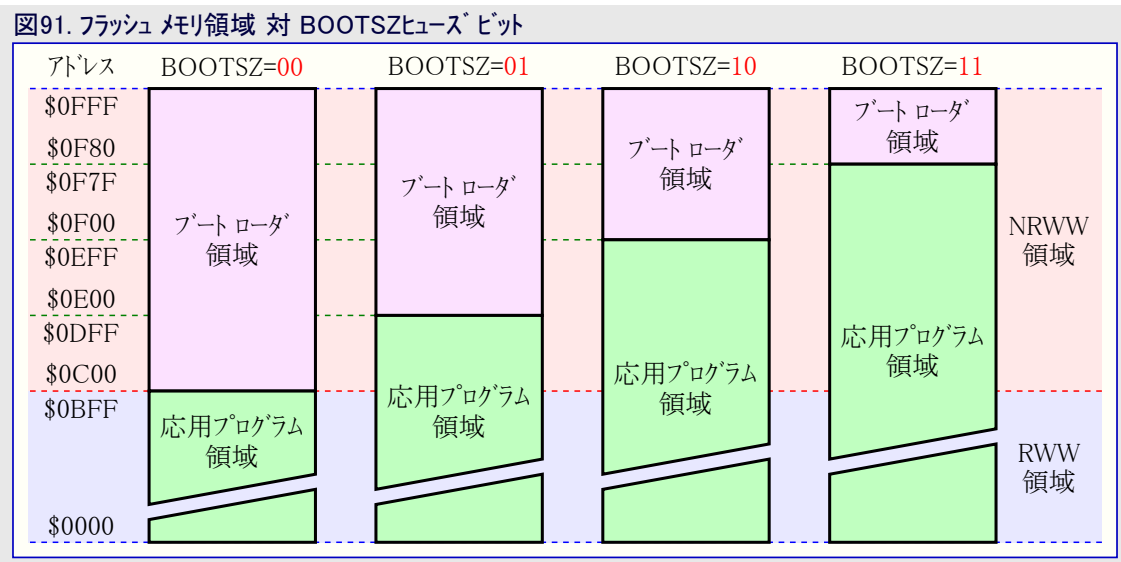
応用とブートローダの領域の大きさは表82.で示されるように変更することができます。

表82. 応用とブートローダの領域の大きさ設定

BOOTSZ1	BOOTSZ0	応用領域		ブートローダ領域		
		開始	終了	開始	終了	大きさ(語)
0	0	\$0000	\$0BFF	\$0C00	\$0FFF	1024
0	1		\$0DFF	\$0E00		512
1	0		\$0EFF	\$0F00		256
1	1		\$0F7F	\$0F80		128

注: ブートローダ領域の開始=ブートルセットアドレス

ブートローダ領域任意選択は下の図91.で図解されます。



## 21.7. ブートローダ施錠ビット

ブートローダは独立して設定することができる、2つの独立した施錠ビットの組を持ちます。これは異なる保護の水準を選ぶため、使用者に独特な柔軟性を与えます。

使用者は以下のどれをも選ぶことができます。

- MCUによるソフトウェア更新からフラッシュメモリ全体を保護
- MCUによるソフトウェア更新からブートローダフラッシュ領域だけを保護
- MCUによるソフトウェア更新から応用フラッシュ領域だけを保護
- フラッシュメモリ全体でソフトウェア更新を許可

施錠ビットのより多くの詳細については151頁の「施錠ビット」をご覧ください。

### 21.7.1. SPMによるブートローダ施錠ビットのプログラミング

ブートローダと一般の施錠ビットを設定するのは次のとおりです。

- R0に望むデータを書いてください。解除(0)ビットは対応する施錠ビットがプログラム(0)されるべきであることを示します。下のR0のビット割り当てをご覧ください。
- SPM命令制御/状態レジスタ(SPMCSR)に“x0001001”を書いてください。
- 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。

施錠ビットのプログラミング中、R0の内容は下で示されるように扱われます。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	LB2	LB1

操作中、Zポインタの値は無視されますが、将来との共通性のため、(施錠ビット読み出しに使われるのと同じ)\$0001でZポインタを設定することが推奨されます。将来との共通性のため、施錠ビット書き込み時に、R0のビット7,6を1に設定することも推奨されます。

施錠ビットをプログラミングするとき、操作中に全てのフラッシュメモリを読むことができます。

フラッシュメモリのアクセスに影響を及ぼすブートローダ施錠ビットの各種設定法については151頁の表87と表88をご覧ください。施錠ビットの配置については151頁の「施錠ビット」をご覧ください。

### 21.7.2. ブートローダ領域(BLS)の更新

BLSに更新を許す、ブートローダ施錠ビットのBLB11が非プログラム(1)のままにされる場合に特別な注意が払われなければなりません。BLSへの偶然の書き込みは更なるソフトウェア更新を不可能にする、ブートローダ全体を不正にし得ます。ブートローダソフトウェアが更新不要の場合、ソフトウェアによる変更からBLSを保護するためにブート施錠ビットのBLB11をプログラム(0)することが推奨されます。

## 21.8. フラッシュメモリの自己プログラミング

デバイスはMCU自身によってプログラムコードを書き込みと読み出しをするため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プログラムメモリ内にコードを書き(プログラム)、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をも使用することができます。

プログラムメモリはページ単位形式で更新されます。一時ページ緩衝部に格納されたデータでページをプログラミングする前に、ページは消去されなければなりません。一時ページ緩衝部はSPM命令を用いて一度に1語で満たされ、緩衝部はページ消去指令の前、またはページ消去とページ書き込みの操作間のどちらかで満たすことができます。

### 手段1 (ページ消去前の緩衝部格納)

1. 一時ページ緩衝部を満たしてください。
2. ページ消去を実行してください。
3. ページ書き込みを実行してください。

### 手段2 (ページ消去後の緩衝部格納)

1. ページ消去を実行してください。
2. 一時ページ緩衝部を満たしてください。
3. ページ書き込みを実行してください。

手段1を使う場合、ブートローダは、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-モディファイライト)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、格納中に旧データを読むことができません。

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りは(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。

一時ページ緩衝部は乱順でアクセスすることができます。ページ消去とページ書き込みの両操作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重要です。

### 21.8.1. 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、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得ます(156頁の表94をご覧ください)。下位側ビットから成る1つの領域はページ内の語(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。これは下の図92.で示されます。

図92. SPM操作中のフラッシュメモリのアドレス指定

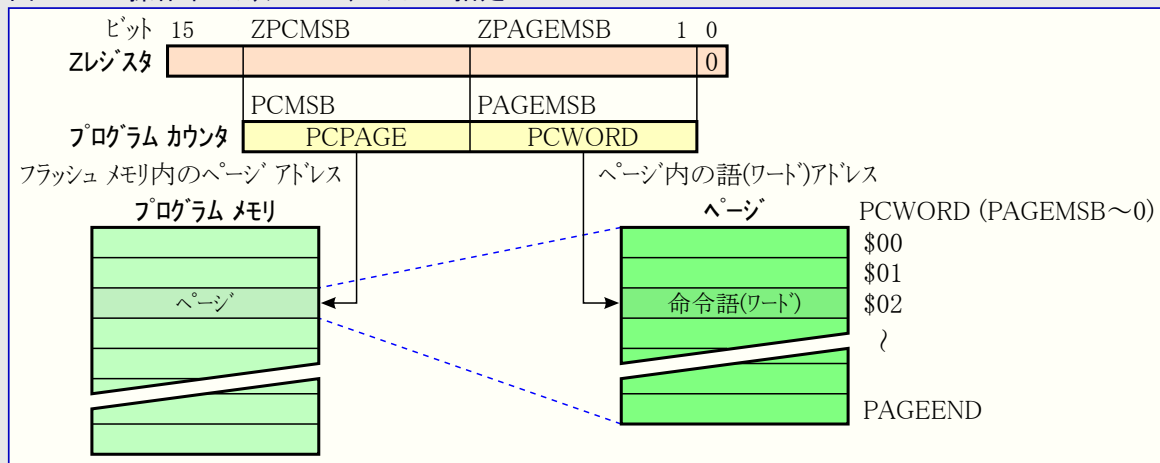


図92.で使われた変数は次の表83.で説明されます。

表83. フラッシュメモリのアドレス指定で使変数

変数	説明
PCPAGE	プログラムカウンタページアドレス。語のページを選択し、ページ消去とページ書き込みの操作で使われます。156頁の表94をご覧ください。
PCMSB	プログラムカウンタの最上位ビット。156頁の表94をご覧ください。
ZPCMSB	PCMSBに割り当てられるZレジスタ内のビット。Z0が使われないため、ZPCMSB=PCMSB+1。ZPCMSBを超えるZレジスタは無視されます。
PCWORD	プログラムカウンタ語(ワード)アドレス。ページ内の語を選びます。これは一時緩衝部を満たすのに使われ、ページ書き込み操作中は0でなければなりません。156頁の表94をご覧ください。
PAGEMSB	1つのページ内で語をアドレス指定するのに使われる最上位ビット。
ZPAGEMSB	PAGEMSBに割り当てられるZレジスタ内のビット。Z0が使われないため、ZPAGEMSB=PAGEMSB+1。

ページ消去とページ書き込みの操作が独立してアドレス指定されることに注意してください。従って、ブートローダはページ消去とページ書き込みの両操作で同じページをアドレス指定することが非常に重要です。

例えZレジスタの最下位ビット(Z0)がSPMに対して0であるべきでも、LPM命令はフラッシュメモリをバイト単位でアドレス指定し、バイト選択ビットとしてZ0を使うことに注意すべきです。

Zポインタを使わないSPM操作はブートローダ施錠ビット設定だけです。この操作でZレジスタの内容は無視され、無効です。

一旦プログラミング操作が開始されると、アドレスはラッチされ、Zポインタは他の操作に使うことができます。

### 21.8.2. ページ消去

ページ消去を実行するには次のとおりです。

- Zポインタにアドレスを構成設定してください。
- SPM命令制御/状態レジスタ(SPMCSR)に'00000011'を書いてください。
- 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。

上の時間制限手順中に割り込みが起こる場合に4周期アクセスを保証することができません。非分断操作を保証するためにSPMCSR書き込み前に割り込みが禁止されるべきです。

RWW領域のページ消去実行時、その操作中にNRWW領域を読むことができます。NRWW領域のページ消去実行時、その操作中にCPUは停止されます。

### 21.8.3. ページ設定

命令語(ワード)を(一時ページ緩衝部に)書くには次のとおりです。

- Zポインタにアドレスを構成設定してください。
- R1:R0にデータを構成設定してください。
- SPMCSRに'00000001'を書いてください。
- 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内のデータのアドレス指定に使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのRWWSREビット書き込みによって自動的に消去されます。システムリセット後も消去されます。

一時緩衝部の消去なし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。

### 21.8.4. ページ書き込み

ページ書き込みを行うには次のとおりです。

- Zポインタにアドレスを構成設定してください。
- SPMCSRに'00000101'を書いてください。
- 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。この操作中に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。

RWW領域のページ消去実行時、その操作中にNRWW領域を読むことができます。NRWW領域のページ消去実行時、その操作中にCPUは停止されます。

### 21.8.5. SPM割り込み

SPM操作可割り込みが許可された場合、SPMCSRのSPMENビットが解除(0)されている時に絶えず続くSPM操作可割り込みを生成します。これはソフトウェアでSPMCSRをポーリングする代わりにこの割り込みが使えることを意味します。

SPM操作可割り込みを使用時、読み出しに対して防がれる時に割り込みがRWW領域をアクセスするのを避けるために、割り込みベクタはブートローダ領域へ移動されるべきです。割り込み(ベクタ)の移動は32頁の「割り込み」章で記述されます。

### 21.8.6. EEPROMプログラミング中の時にSPMCSR書き込み不可

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ご注意してください。ソフトウェアからのヒューズと施錠ビットの読み出しも、EEPROM書き込み動作中に妨げられます。使用者はSPM命令制御/状態レジスタ(SPMCSR)へ書く前に、EEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビットを調べて、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。

### 21.8.7. 自己プログラミング中のRWW領域読み込み不可

自己プログラミング中(ページ消去やページ書き込み)、RWW領域は読み出しに対して常に防がれ、従ってソフトウェアは操作中のアドレス指定からこの領域を保護すべきです。SPMCSRのRWWSBはRWW領域が多忙である限り設定(1)されます。

自己プログラミング中、割り込みベクタ表はブートローダ領域(BLS)へ移動されるべきです。35頁の「MCUCR – MCU制御レジスタ」の割り込みベクタ選択(IVSEL)ビットをご覧ください。代わりに、割り込みは禁止されるべきです。

プログラミング完了後、RWW領域をアドレス指定する前に、使用者ソフトウェアはRWWSREビットの書き込みによってRWWSBを解除(0)しなければなりません。

## 21.9. フラッシュメモリ化けの防止

低VCCの期間中、CPUとフラッシュメモリの正しい動作に対して供給電圧が低すぎるためにフラッシュメモリのプログラムが不正にされ得ます。これらの問題はフラッシュメモリを使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

フラッシュメモリのプログラム化けは電圧が低すぎる時の2つの状態によって起こされます。1つ目としてフラッシュメモリへの通常の書き込み手順は正しく動作するための最低電圧が必要です。2つ目として供給電圧が低すぎると、CPU自身が命令を間違えて実行し得ます。

フラッシュメモリ化けは次の推奨設計によって容易に避けられます(1つは必須)。

- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアをパワーダウン休止動作に保ってください。これはCPUが命令の復号と実行を試みるのを防ぎ、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。
- そのシステムでブートローダ更新が必要ない場合、どんなブートローダソフトウェア更新をも防ぐためにブートローダ施錠ビットをプログラム(0)してください。

### 21.10. 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

フラッシュメモリアクセスは校正された内蔵RC発振器を用いて計時されます。CPUに対する代表的なフラッシュメモリプログラミング時間は表84.で示されます。

表84. SPM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。



## 21.11. 自己プログラミング用レジスタ

### 21.11.1. SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register) SPMCSR

このレジスタはメモリプログラミング操作を制御するのに必要とされる制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$37 (\$57)	SPMIE	RWWSB	RSIG	RWWSRE	RWFLB	PGWRT	PGERS	SPMEN	SPMCSR
Read/Write	R/W	R	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 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIE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)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みはSPM操作許可(SPMEN)ビットが解除(0)されている限り実行されます。

#### ● ビット6 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)

このビットが設定(1)されると、RWW領域はアクセスできません。

このビットはRWW領域に自己プログラミング(ページ消去またはページ書き込み)操作が開始される時に設定(1)されます。

このビットは自己プログラミング完了後にRWWSREビットが1を書かれる場合に解除(0)されます。このビットはページ設定操作が開始される時に自動的に解除(0)されます。

#### ● ビット5 – RSIG : デバイス識票銘刻読み込み (Read Device Signature Imprint Table)

SPMCSRでRSIGとSPMENが設定(1)された後の3クロック周期内のLPM命令実行はデバイス識票銘刻部から(Zポインタ値に依存する)選択したデータを転送先レジスタ内に返します。詳細については153頁の「デバイス識票銘刻部」をご覧ください。

#### ● ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域はこの領域がプログラミングされつつある時に読み出しに対して防がれます(RWWSBビットをご覧ください)。この領域を再許可するためにソフトウェアは最初にプログラミングが完了されるまで待たねばなりません(SPMENビットをご覧ください)。その後、RWW領域はSPM命令発行後の4クロック周期内のRWWSREとSPMENのビットの同時書き込みによって許可されます。

RWW領域はページ消去またはページ書き込みでフラッシュメモリが多忙の間、再許可することができません(SPMENをご覧ください)。フラッシュメモリ設定中にRWWSREビットが書かれた場合、その操作は失敗し、設定したデータは失われます。

#### ● ビット3 – RWFLB : ヒューズ/施錠ビット読み書き (Read/Write Fuse and Lock Bits)

RWFLBとSPMENのビットが設定(1)された後3クロック内のLPM命令は(ZポインタのZ0によって)施錠ビットまたはヒューズ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153頁の「ソフトウェアからの施錠、ヒューズ、識票データの読み出し」をご覧ください。

このビットがSPMENと同時に1を書かれた場合、次の4クロック周期内のSPM命令は、R0のデータに従ってフート施錠ビットとメモリ施錠ビットを設定します。R1のデータとZポインタのアドレスは無視されます。このビットは施錠ビットが設定されてしまった時、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。

#### ● ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に1を書かれた場合、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了、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。NRWW領域がアドレス指定されると、ページ全体の書き込み動作中に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SPMENと同時に1を書かれた場合、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了、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。NRWW領域がアドレス指定されると、ページ全体の消去中にCPUは停止されます。

#### ● ビット0 – SPMEN : SPM操作許可 (Store Program Memory Enable)

このビットは次の4クロック周期間のSPM命令を許可します。このビットがRSIG、RWWSRE、RWFLB、PGWRT、PGERSのどれかと共に1を設定された場合、続くSPM命令は他で記述されるように特別な意味を持ちます。

SPMENだけが書かれると、続くSPM命令はZポインタによってアドレス指定された一時ページ緩衝部へR1:R0の値を格納します。Zポインタの最下位ビットは無視されます。SPMENビットはSPM命令の完了、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、SPMENビットはその動作が完了されるまで1に留まります。

## 22. 施錠ビット、ヒューズビット、デバイス識票

### 22.1. 施錠ビット

ATtiny828は表85.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85. 施錠ビットバイト

名称	ビット番号	説明	参照	既定値 (注)
-	7			1 (非プログラム)
-	6			1 (非プログラム)
BLB12	5			1 (非プログラム)
BLB11	4	ブート施錠ビット	表88. 表87.	1 (非プログラム)
BLB02	3			1 (非プログラム)
BLB01	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	表86.	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)のままにする、または表86.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86. 一般施錠ビット保護形態

施錠ビット (注1)		保護形態
LB2	LB1	
1	1	メモリ施錠機能は全く許可されません。
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2)
0	1	(予約)
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2)

注1: 0はプログラム、1は非プログラムを意味します。

注2: 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

一般読み書き施錠ビットはLPM/SPMによる読み書きを制御しません。SPMとLPMの応用とブートローダの領域へのアクセスは表87.と表88.で一覧にされるブート施錠ビットによって制御されます。

表87. ブート施錠ビット保護形態 (応用領域)

施錠ビット (注1)		保護形態
BLB02	BLB01	
1	1	SPMやLPMの応用領域アクセスに全く制限はありません許可されません。
1	0	SPMは応用領域への書き込みを許されません。
0	1	ブートローダ領域からのLPM実行は応用領域からの読み込みを許されません。割り込みベクタがブートローダ領域に置かれる場合、応用領域から実行する間、割り込みが禁止されます。
0	0	SPMは応用領域への書き込みが許されず、ブートローダ領域からのLPM実行は応用領域からの読み込みが許されません。割り込みベクタがブートローダ領域に置かれる場合、応用領域から実行する間、割り込みが禁止されます。

注1: 0はプログラム、1は非プログラムを意味します。

表88. ブート施錠ビット保護形態 (ブートローダ領域)

施錠ビット (注1)		保護形態
BLB12	BLB11	
1	1	SPMやLPMのブートローダ領域アクセスに全く制限はありません許可されません。
1	0	SPMはブートローダ領域への書き込みを許されません。
0	1	応用領域からのLPM実行はブートローダ領域からの読み込みを許されません。割り込みベクタが応用領域に置かれる場合、ブートローダ領域から実行する間、割り込みが禁止されます。
0	0	SPMはブートローダ領域への書き込みが許されず、応用領域からのLPM実行はブートローダ領域からの読み込みが許されません。割り込みベクタがブートローダ領域に置かれる場合、ブートローダ領域から実行する間、割り込みが禁止されます。

注1: 0はプログラム、1は非プログラムを意味します。

ブート施錠ビットはソフトウェアによって、または直列や並列のプログラミング動作で設定(0)することができますが、それらはチップ消去指令によってだけ解除(1)することができます。146頁の「SPMによるブート施錠ビットのプログラミング」をご覧ください。

## 22.2. ヒューズ ビット

ヒューズ ビットは表89、表90、表91で記述されます。プログラムされたヒューズが0として読まれることに注意してください。

表89. 拡張ヒューズ バイト

名称	ビット	使用	参照	既定値
BODPD1	7	デバイスがアイドル以外の休止動作形態の時の低電圧検出(BOD)動作	28頁	1 (非プログラム)
BODPD0	6	形態を設定		1 (非プログラム)
BODACT1	5	デバイスが活動またはアイドル動作時の低電圧検出(BOD)動作形態を	28頁	1 (非プログラム)
BODACT0	4	設定		1 (非プログラム)
-	3	-		1 (非プログラム)
BOOTSZ1	2	ブート ロータ領域の大きさを設定	145頁	1 (非プログラム)
BOOTSZ0	1			1 (非プログラム)
BOOTrST	0	ブートリセット ベクタを定義	145頁	1 (非プログラム)

表90. ヒューズ 上位バイト

名称	ビット	使用	参照	既定値
RSTDISBL	7	外部リセットを禁止 (注1)	51頁	1 (非プログラム)
DWEN	6	デバッグ WIRE を許可 (注1)	143頁	1 (非プログラム)
SPIEN	5	デバイスへの直列プログラミングを許可 (注2)		0 (プログラム) (注3)
WDTON	4	ウォッチドッグ タイマを定期的にON設定	29頁	1 (非プログラム)
EESAVE	3	チップ消去操作中にEEPROM内容を保護	157頁	1 (非プログラム) (注4)
BODLEVEL2	2	低電圧検出(BOD)リセット検出電圧設定	168頁	1 (非プログラム)
BODLEVEL1	1			1 (非プログラム)
BODLEVEL0	0			1 (非プログラム)

注1: 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て、直列インターフェース経由の更なるプログラミングの可能性を放棄します。このヒューズ ビットは並列プログラミング法を使用して非プログラム(1)にすることができます(156頁をご覧ください)。

注2: このヒューズ ビットは直列プログラミング形態でアクセスできません。

注3: この設定はSPI(直列)プログラミングを許可します。

注4: この設定はEEPROMを保護しません。

表91. ヒューズ 下位バイト

名称	ビット	使用	参照	既定値
CKDIV8	7	クロックを8分周 (注1)	21頁	0 (プログラム)
CKOUT	6	ポートピンでシステム クロックを出力	20頁	1 (非プログラム)
SUT1	5	システム始動時間を設定	20頁	1 (非プログラム) (注2)
SUT0	4			0 (プログラム) (注2)
-	3	-		1 (非プログラム)
-	2	-		1 (非プログラム)
CKSEL1	1	クロック元を選択	18頁	1 (非プログラム) (注3)
CKSEL0	0			0 (プログラム) (注3)

注1: 低電圧でのこのヒューズの非プログラム(1)は過剰クロック駆動に帰着するかもしれません。デバイス速度対供給電圧については167頁の「速度」項をご覧ください。

注2: この設定は既定クロック元に対する最大始動時間に帰着します。

注3: この設定は校正付き内部8MHz発振器を選択します。

ヒューズ ビットは施錠ビット1(LB1)がプログラム(0)される時に固定されます。従って、ヒューズ ビットは施錠ビットの前にプログラミング(書き込み)されなければなりません。

ヒューズ ビットはチップ消去によって影響を及ぼされません。

### 22.2.1. 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。ヒューズは標準動作形態での電源投入でもラッチされます。

## 22.3. デバイス識票銘刻部

デバイス識票銘刻部はデバイス識票と発振器校正データのような種々雑多なデバイス情報の格納に使われる専用のメモリ領域です。表92.で略述されるように、このメモリ領域の殆どは内部使用に予約されています。

デバイス自身がLPM命令でデータを読む時にはバイトアドレスが使われます。外部書き込み装置は語(ワード)アドレスを使わなければなりません。

表92. デバイス識票銘刻部の内容

語アドレス (外部)	バイトアドレス (内部)	説明
\$00	\$00	識票バイト0 (注1)
	\$01	内部8MHz発振器用校正データ (OSCCAL0) (注2)
\$01	\$02	識票バイト1 (注1)
	\$03	発振器温度校正データ (OSCTCAL0A)
\$02	\$04	識票バイト2 (注1)
	\$05	発振器温度校正データ (OSCTCAL0B)
\$03	\$06	(予約)
	\$07	内部32kHz発振器用校正データ (OSCCAL1) (注2)
\$04~\$15	\$08~	(予約)
	~\$2B	(予約)
\$16	\$2C	温度感知器用校正データ (利得) (注3,4)
	\$2D	温度感知器用校正データ (変位(オフセット)) (注3,5)
\$17~\$3F	\$2E~	(予約)
	~\$7F	(予約)

注1: より多くの情報については下の「識票バイト」をご覧ください。

注2: より多くの情報については下の「校正バイト」をご覧ください。

注3: 138頁の「温度測定」をご覧ください。校正データはATtiny828Rデバイスでだけ有効です。

注4: 符号なし2の補数固定小数点 [0~(255/128)]

注5: 符号付き2の補数整数 [-127~+128]

### 22.3.1. 識票バイト

全てのAtmelマイクロ コントローラはデバイスを識別する3バイトの識票符号を持ちます。この符号は直列と並列の両プログラミング動作で、またデバイスが施錠されている時も読むことができます。

識票バイトはデバイスファームウェアによって読むこともできます。「ソフトウェアからの施錠、ヒューズ、識票データの読み出し」項をご覧ください。

この3バイトの識票バイトはデバイス識票銘刻部と呼ばれる分離された空間に存在します。ATtiny828用の識票バイトは表93.で与えられます。

表93. デバイス識票バイト

部品番号	識票バイト		
	0	1	2
ATtiny828	\$1E	\$93	\$14

### 22.3.2. 校正バイト

表92.で示されるようにATtiny828のデバイス識票銘刻部は内部発振器用の校正データを含みます。リセット中、校正付き発振器の正しい周波数を保証するために、校正データが校正レジスタ(OSCCAL0,OSCCAL1)へ自動的に複写されます。21頁の「OSCCAL0 - 発振器校正レジスタ」と22頁の「OSCCAL1 - 発振器校正レジスタ」をご覧ください。

校正バイトはデバイスファームウェアによって読むこともできます。「ソフトウェアからの施錠、ヒューズ、識票データの読み出し」項をご覧ください。

## 22.4. ソフトウェアからの施錠、ヒューズ、識票データの読み出し

ヒューズと施錠ビットはデバイスファームウェアによって読むことができます。プログラム(0)されたヒューズと施錠ビットは0を、非プログラム(1)にされたものは1として読みます。151頁の「施錠ビット」と152頁の「ヒューズビット」をご覧ください。

加えて、ファームウェアはデバイス識票銘刻部からデータを読むこともできます。「デバイス識票銘刻部」をご覧ください。



### 22.4.1. 施錠ビット読み出し

SPMCSRでフラッシュ/施錠ビット読み書き(RWFLB)とSPM操作許可(SPMEN)のビットが設定(1)された後の3CPU周期内でLPM命令が発行された後、転送先レジスタ内に施錠ビット値が返されます。RWFLBとSPMENのビットは施錠ビット読み出しの完了で、または3CPU周期内でLPM命令が実行されないか、または4CPU周期内でSPM命令が実行されない場合、自動的に解除(0)されます。通常、RWFLBとSPMENのビットが解除(0)されるのはLPMの作用です。

施錠ビットを読むには以下の手続きに従ってください。

1. Zポイントに\$0001を格納してください。
2. SPM命令制御/状態レジスタ(SPMCSR)でRWFLBとSPMENのビットを設定(1)してください。
3. 3クロック周期内にLPM命令を発行してください。
4. LPM転送先レジスタから施錠ビット値を読んでください。

成功なら、転送先レジスタの内容は次のとおりです。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	BLB12	BLB11	BLB02	BLB01	LB2	LB1

より多くの情報については151頁の「施錠ビット」をご覧ください。

ブートローダ施錠ビットの各種設定がフラッシュアクセスにどう影響を及ぼすかについては146頁の「ブートローダ施錠ビット」をご覧ください。

### 22.4.2. ヒューズビット読み出し

ヒューズバイトを読む方法はアドレスが異なるだけで上記の施錠ビット読み出しと同様です。

ヒューズ下位バイト(FLB)を読むには以下の手続きに従ってください。

1. Zポイントに\$0000を格納してください。
2. SPMCSRでRWFLBとSPMENのビットを設定(1)してください。
3. 3クロック周期内にLPM命令を発行してください。
4. LPM転送先レジスタからFLB値を読んでください。

成功なら、転送先レジスタの内容は次のとおりです。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	-	-	FLB1	FLB0

ヒューズ下位バイトの詳細記述と配置については152頁の表91をご覧ください。

ヒューズ上位バイト(FHB)を読むにはZポイント内のアドレスを\$0003に置き換え、上の手続きを繰り返してください。成功なら、転送先レジスタの内容は次のとおりです。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

ヒューズ上位バイトの詳細記述と配置については152頁の表90をご覧ください。

ヒューズ拡張バイト(FEB)を読むにはZポイント内のアドレスを\$0002に置き換え、直前の手続きを繰り返してください。成功なら、転送先レジスタの内容は次のとおりです。

ビット	7	6	5	4	3	2	1	0
Rd	FEB7	FEB6	FEB5	FEB4	-	FEB2	FEB1	FEB0

ヒューズ拡張バイトの詳細記述と配置については152頁の表89をご覧ください。

### 22.4.3. デバイス識票銘刻部読み出し

デバイス識票銘刻部の内容を読むには以下の手続きに従ってください。

1. Zポイントに銘刻指示子を格納してください。
2. SPMCSRでデバイス識票銘刻読み込み(RSIG)とSPM操作許可(SPMEN)のビットを設定(1)してください。
3. 3クロック周期内にLPM命令を発行してください。
4. LPM転送先レジスタから表データ値を読んでください。

成功なら、転送先レジスタの内容は153頁の「デバイス識票銘刻部」項で記述されたとおりです。

以下のプログラム例をご覧ください。



### アセンブリ言語プログラム例

```
DSIT_read:  LDI    ZH, 0           ;表指示子上位バイト値設定
            LDI    ZL, 1       ;表指示子下位バイト値設定
            LDI    R17, (1<<RSIG) | (1<<SPMEN) ;RSIG,SPMEN=1値取得
            OUT    SPMCSR, R17 ;RSIG,SPMEN=1設定
            LPM    R16, Z       ;$0001位置の識票値取得
            RET                  ;呼び出し元へ復帰
```

**注:** 5頁の「コード例」をご覧ください。

## 23. 外部プログラミング

本章はATtiny828内のフラッシュメモリ、EEPROM、施錠ビット、ヒューズビットの書き込みと検証の方法を記述します。

### 23.1. メモリの要素

フラッシュメモリの要素は下の表94.で要約されます。

表94. フラッシュメモリの要素

デバイス	フラッシュ容量	ページ容量	PCWORD (注)	ページ数	PCPAGE (注)	PCMSB (注)
ATtiny828	4K語(8Kバイト)	32語	PC4~0	128	PC11~5	11

注: 148頁の表83.をご覧ください。

EEPROMの要素は下の表95.で要約されます。

表95. EEPROMの要素

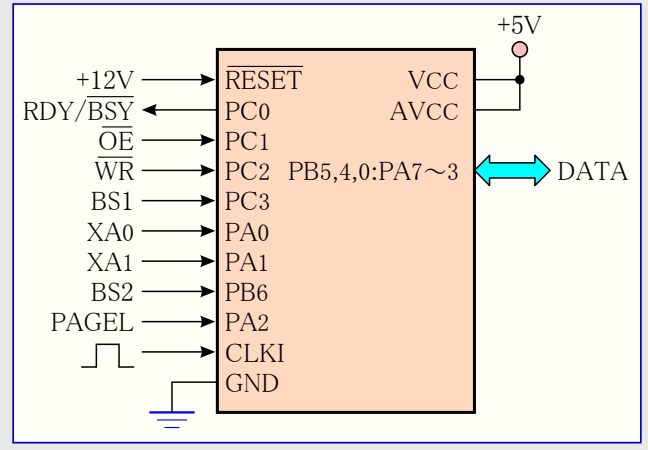
デバイス	EEPROM容量	ページ容量	PCWORD (注)	ページ数	PCPAGE (注)	EEAMSB
ATtiny828	256バイト	4バイト	EEA1~0	64	EEA7~2	7

注: 148頁の表83.をご覧ください。

### 23.2. 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93.で図解されます。

図93. 並列プログラミング信号



信号は下の表96.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96. プログラミング動作で使われるピンと信号の名前

信号名	ピン名	入出力	機能
RDY/BSY	PC0	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	PC1	入力	出力許可(負論理)
WR	PC2	入力	書き込みパルス(負論理)
BS1	PC3	入力	上位/下位バイト選択1 (0:下位, 1:上位) (一般用)
XA0	PA0	入力	XTAL動作ビット0
XA1	PA1	入力	XTAL動作ビット1
BS2	PB6	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)
PAGEL	PA2	入力	プログラムメモリとEEPROMのデータを一時ページ緩衝部に設定
DATA	PB5,4,0:PB7~3 (MSB~LSB)	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であるべきです。

特記事項を除いて、パルス幅は最低250nsと仮定されます。

表97. プログラミング動作移行時に使用されるピン値

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98.で示されます。

表98. 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99.で示されます。

表99. 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		

### 23.2.1. 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、156頁の表97.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 20µs以内にVCCが最低1.7Vに達することを保証してください。
- ④ 20~60µs待ち、RESETに11.5~12.5Vを印加します。
- ⑤ 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。
- ⑥ 如何なる並列プログラミング指令を与えるのにも先立って少なくとも300µs間待ちます。
- ⑦ デバイスの電源を落とすか、RESETピンを0Vに持つ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、156頁の表97.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、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に持つてくることによってプログラミング動作を抜けます。

### 23.2.2. 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。

### 23.2.3. 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。

### 23.2.4. フラッシュ メモリ書き込み

フラッシュ メモリは156ページの表94.で示されるようにページで構成されます。フラッシュ メモリを書く時にプログラム データは先に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)を選択します。
- 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択します。
- 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択します。
- ② PAGESLに正パルスを与えます。これは語データを一時ページ緩衝部にラッチ(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を繰り返す

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で、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。これは次ページの図94.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)を選択します。
- ③ DATAにアドレス上位バイト(\$00~\$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

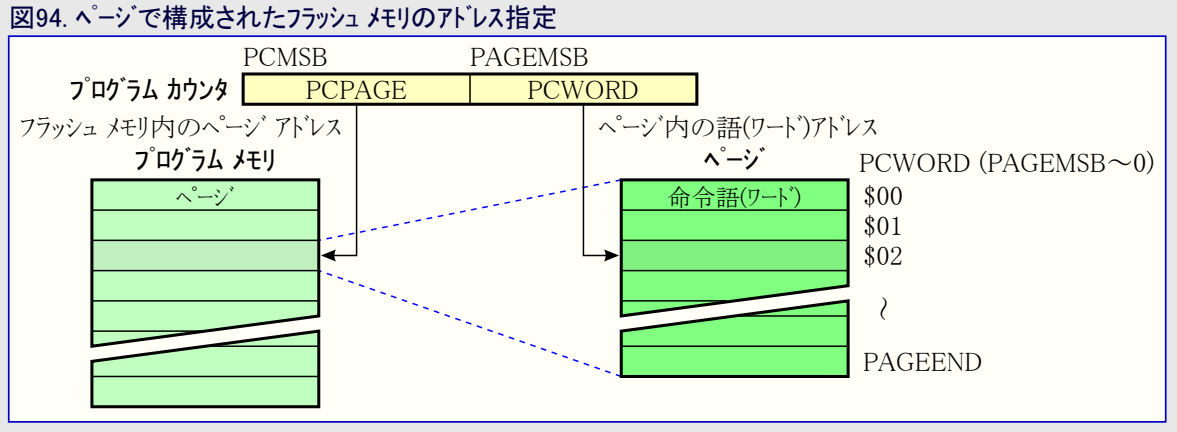
- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/ $\overline{BSY}$ がLow(0)になります。
- ③ RDY/ $\overline{BSY}$ がHigh(1)になるまで待ちます。(信号波形については次ページの図95.をご覧ください。)

#### 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を繰り返す

#### J. ページ書き込み終了

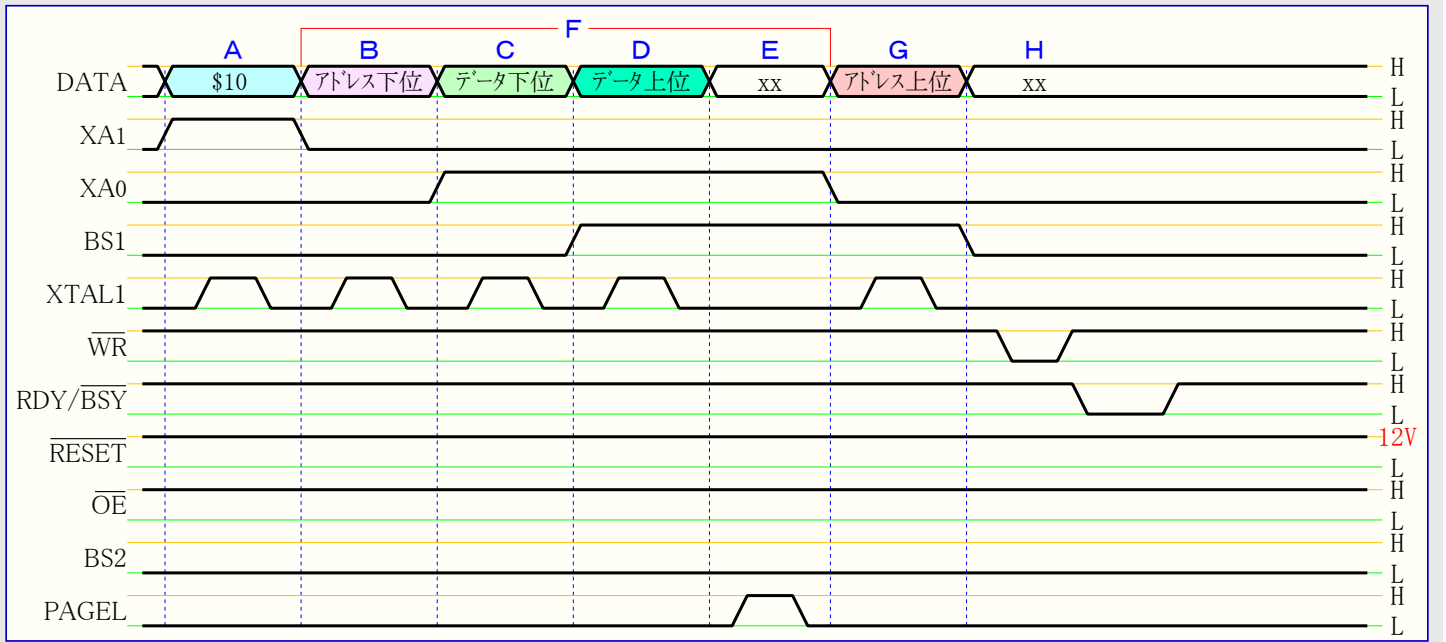
- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ CLKIに正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

フラッシュ ページのアドレス指定は下の図94.で図解されます。使われるシンボルは148頁の表83.で記述されます。



フラッシュ プログラミング 波形は図95.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95. フラッシュ メモリ書き込み波形





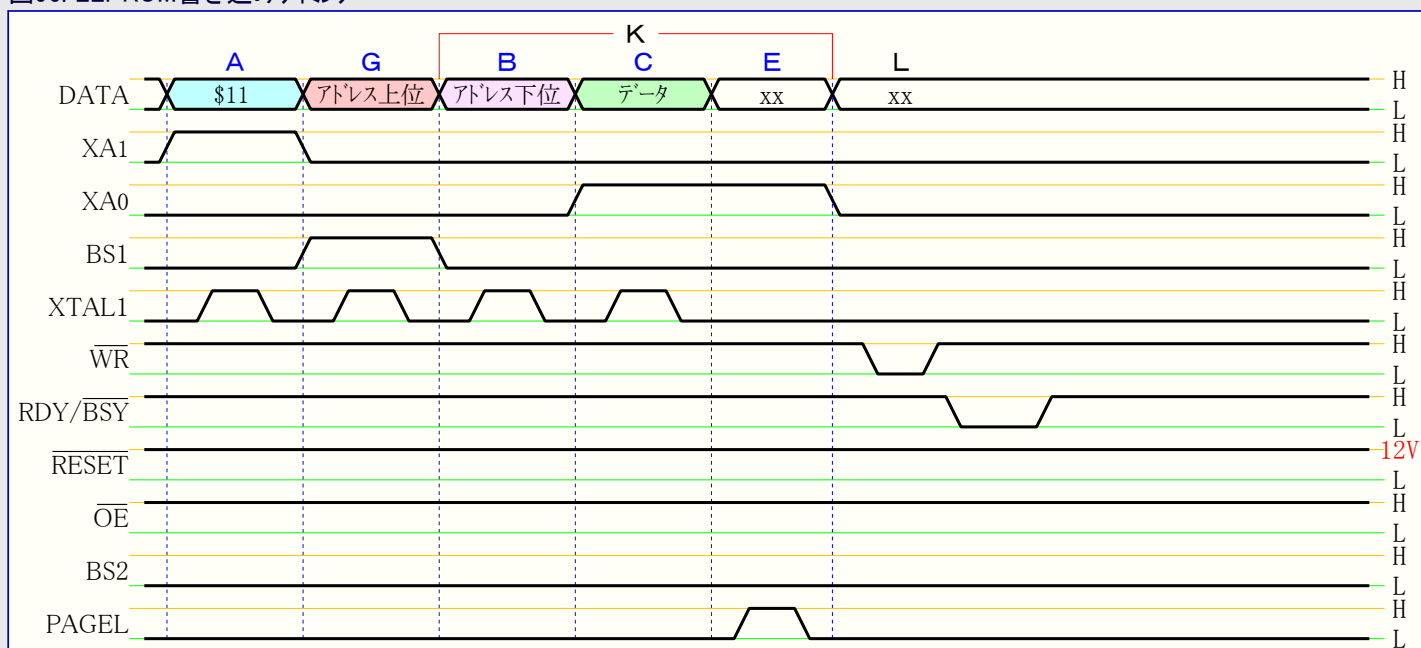
### 23.2.5. EEPROM書き込み

EEPROMは156頁の表95.で示されるようにページで構成されます。EEPROMを書く時にデータはページ緩衝部にラッチされます。これは同時に書かれることをデータの1ページに許します。EEPROMデータメモリの書き込み方法は次のとおりです。(指令、アドレス、データ設定の詳細については158頁の「フラッシュメモリの書き込み」をご覧ください。)

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~\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. 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。(信号波形については図96.をご覧ください。)

EEPROMのプログラミング波形は図96.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96. EEPROM書き込みタイミング



### 23.2.6. フラッシュメモリ読み出し

フラッシュメモリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については158頁の「フラッシュメモリの書き込み」を参照)

1. フラッシュメモリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$0F)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~\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。フラッシュメモリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュメモリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

### 23.2.7. 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158頁の「フラッシュメモリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$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~\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。EEPROMのバイトデータが直ぐにDATAで読めます。
5.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

### 23.2.8. 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158頁の「フラッシュメモリの書き込み」を参照)

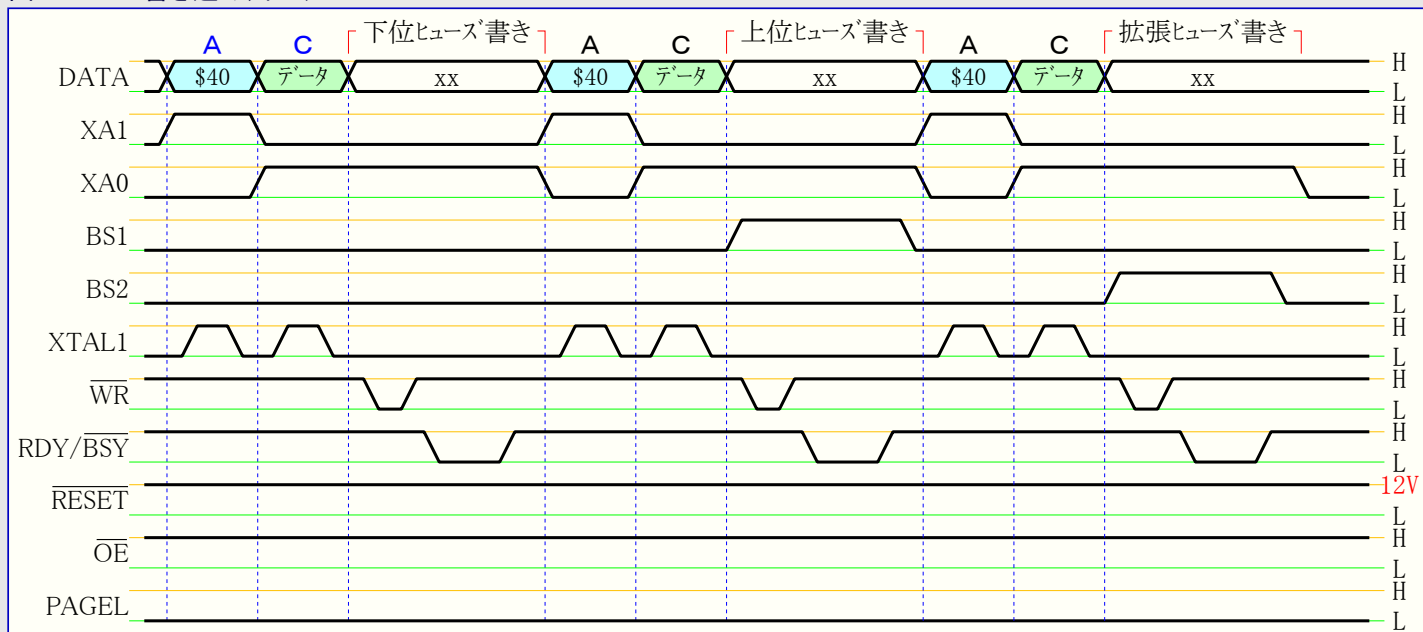
1. ヒューズビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュメモリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4.  $\overline{WR}$ に負パルスを与え、RDY/BSYがHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選択します。

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)

ヒューズプログラミング波形は図97.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97. ヒューズ書き込みタイミング



### 23.2.9. 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158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。LB1とLB2がプログラム(0)される場合、どんな外部的なプログラミング動作形態によっても施錠ビットをプログラミングすることができません。(「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。

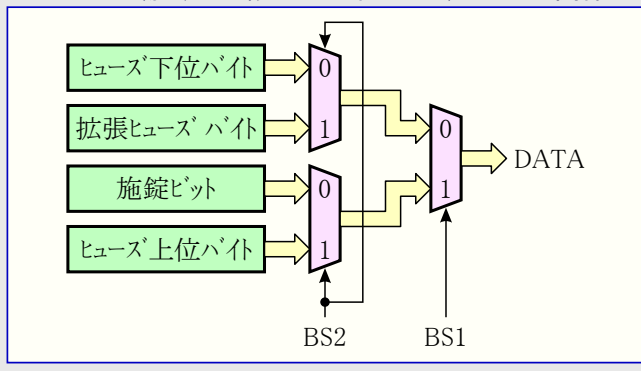
### 23.2.10. ヒューズビットと施錠ビットの読み出し

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158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98.で図解されます。

図98. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



### 23.2.11. 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158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。選択した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。

### 23.2.12. 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158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(\$xx)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。

## 23.3. 直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングすることができます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され得るのに先立って**プログラミング許可命令**が初めに実行されることを必要とします。

直列プログラミング信号と接続は右の図99.で図解されます。ピン割り当ては表100.で一覧にされます。

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。

- 直列クロックの最小Low区間： $f_{CK} < 12\text{MHz}$ 時に $> 2\text{CPUクロック周期}$ 、 $f_{CK} \geq 12\text{MHz}$ 時に $3\text{CPUクロック周期}$
- 直列クロックの最小High区間： $f_{CK} < 12\text{MHz}$ 時に $> 2\text{CPUクロック周期}$ 、 $f_{CK} \geq 12\text{MHz}$ 時に $3\text{CPUクロック周期}$

### 23.3.1. ピン配置

ピン割り当ては表100.で一覧にされます。全てのデバイスが内部SPIインターフェース用の専用SPIピンを使うとは限らないことに注意してください。

表100. 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	PD0	入力	直列データ入力
MISO	PD1	出力	直列データ出力
SCK	PD3	入力	直列クロック

### 23.3.2. プログラミング手順

ATtiny828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。ATtiny828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173頁の図107.と図108.をご覧ください。

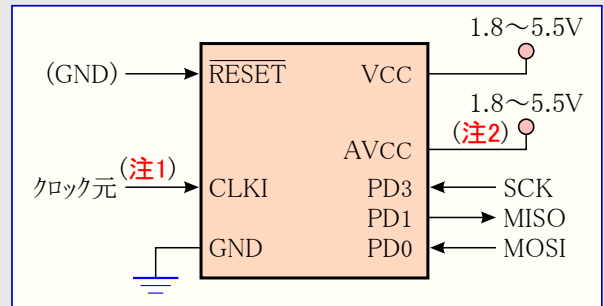
直列プログラミング動作でのATtiny828のプログラミングと照合は次手順が推奨されます(表101.の命令一式をご覧ください)。

- 電源投入手順：RESETとSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加してください。
  - 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、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 $t_{RST} + 2\text{ CPUクロック周期}$ でなければなりません。RESETピンの最小パルス幅の定義については168頁の表107.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、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156頁の表94.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+1ビットとデータを供給することによって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(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低 $t_{WD\_FLASH}$ (表102.参照)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供給することによって1バイト単位で書かれます。EEPROMのメモリ位置は新規データが書かれるのに先立って始めて自動的に消去されます。ポーリング(BSY/RDY)が使われない場合、次のバイトを行う前に最低 $t_{WD\_EEPROM}$ (表102.参照)待たなければなりません。

**ページ単位:** EEPROMの1ページは**EEPROMページ設定命令**と共にアドレスの下位2ビットとデータを供給することによって1バイトづつ設定されます。EEPROMページはアドレスの上位6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95.参照)を行う前に最低、 $t_{WD\_EEPROM}$ (表102.参照)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。

図99. 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:  $VCC - 0.3V < AVCC < VCC + 0.3V$ ですが、AVCCは常に1.7~5.5V内にすべきです。

6. どのメモリ位置も選択したアドレスの内容を直列出力MISOに読み戻す読み出し命令を使うことによって照合できます。
7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。
8. 電源OFF手順 (必要とされるならば) : RESETをHigh(1)に設定してください。VCC電源をOFFにしてください。

### 23.3.3. 直列プログラミング命令一式

表101.と次頁の図100.は命令一式記述します。

表101. 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。
設定系命令					
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。
読み出し命令					
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	
ヒューズ下位読み出し	\$50	\$00	\$00	ヒューズ下位	
ヒューズ上位読み出し	\$58	\$08	\$00	ヒューズ上位	
拡張ヒューズ読み出し	\$50	\$08	\$00	拡張ヒューズ	
校正バイト読み出し	\$38	\$00	\$00	校正バイト	
書き込み命令					
フラッシュ ページ書き込み (注3)	\$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 (注4)	\$AC	\$E0	\$00	施錠ビット値	
ヒューズ下位書き込み (注4)	\$AC	\$A0	\$00	ヒューズ下位	
ヒューズ上位書き込み (注4)	\$AC	\$A8	\$00	ヒューズ上位	
拡張ヒューズ書き込み (注4)	\$AC	\$A4	\$00	拡張ヒューズ	

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2: アドレス系バイトについては次頁の表を参照してください。

注3: プログラム用メモリにアクセスする命令は語(ワード)アドレスを使います。このアドレスはページ範囲内で乱順にできます。

注4: 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。

注: ・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。

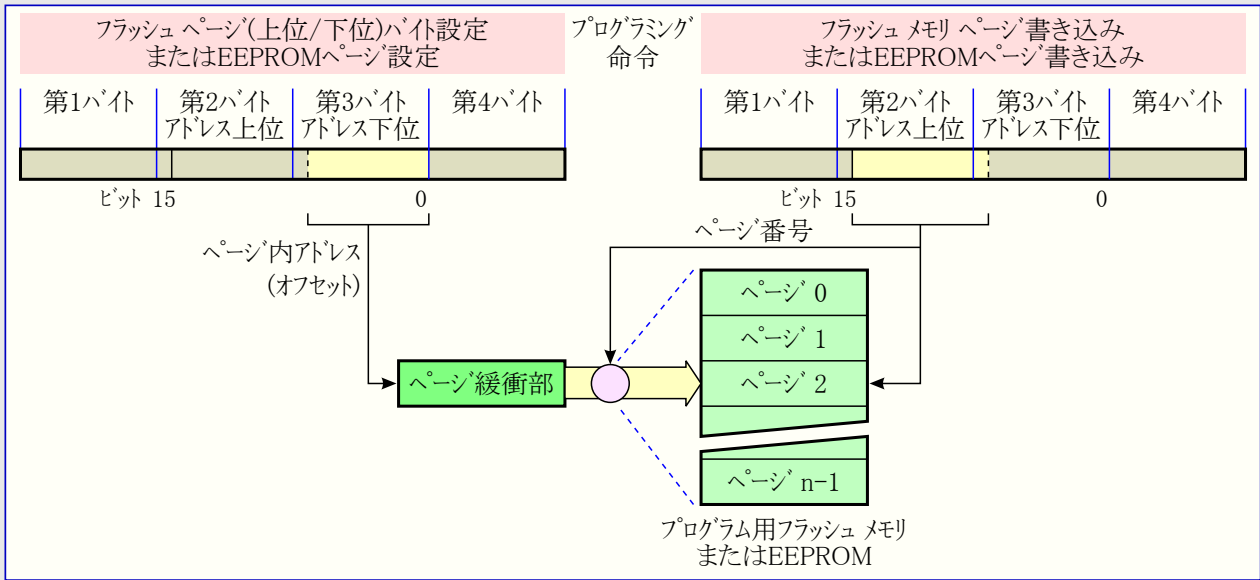
多忙/準備可検査バイト データ出力のLSBが1なら、プログラミング操作が未だ保留(動作中)です。次の命令が実行される前に本ビットが0に戻るまで待ってください。

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。図100.をご覧ください。



図100. 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。ATtiny828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	ATtiny828 : L=PC4~0
EEPROMページ内バイト設定	0000 0000	0000 00LL	ATtiny828 : L=EEA1~0
フラッシュメモリ読み出し	0000 HHHH	LLLL LLLL	ATtiny828 : H=PC11~8,L=PC7~0
EEPROM読み出し	0000 0000	LLLL LLLL	ATtiny828 : L=EEA7~0
フラッシュページ書き込み	0000 HHHH	LLL0 0000	ATtiny828 : H=PC11~8,L=PC7~5
EEPROMバイト書き込み	0000 0000	LLLL LLLL	ATtiny828 : L=EEA7~0
EEPROMページ書き込み	0000 0000	LLLL LL00	ATtiny828 : L=EEA7~2

### 23.4. 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102.で一覧にされます。

表102. 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_FUSE	4.5ms	ヒューズ書き込み
tWD_FLASH	4.5ms	フラッシュメモリ書き込み
tWD_EEPROM	3.6ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

(訳注)  
共通性から  
tWD\_FUSE  
項目を追加

## 24. 電気的特性

### 24.1. 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。

### 24.2. DC特性

表103. DC特性 TA=-40°C~85°C

シンボル	項目	条件	最小	代表 (注1)	最大	単位
VIL	Lowレベル入力電圧	VCC=1.7~2.4V	-0.5		0.2VCC (注2)	
		VCC=2.4~5.5V	-0.5		0.3VCC (注2)	
VIH	Highレベル入力電圧 (RESETを除く)	VCC=1.7~2.4V	0.7VCC (注3)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注3)		VCC+0.5	
	Highレベル入力電圧 (RESET)	VCC=1.7~5.5V	0.9VCC (注3)		VCC+0.5	
VOL	I/OとしてのRESETピン (注6) Lレベル出力電圧 (注4) (注5)	IOL=2mA, VCC=5V			0.6	V
		IOL=1mA, VCC=3V			0.5	
		IOL=0.4mA, VCC=1.8V			0.4	
	標準吸い込みI/Oピン (注7) Lレベル出力電圧 (注4) (注5)	IOL=10mA, VCC=5V			0.6	
		IOL=5mA, VCC=3V			0.5	
		IOL=2mA, VCC=1.8V			0.4	
	高吸い込みI/Oピン (注8) Lレベル出力電圧 (注4) (注5)	IOL=20mA, VCC=5V			0.6	
		IOL=10mA, VCC=3V			0.5	
		IOL=4mA, VCC=1.8V			0.4	
	追加高吸い込みI/Oピン (注9) Lレベル出力電圧 (注4) (注5)	IOL=20mA, VCC=5V			0.6	
		IOL=20mA, VCC=3V			0.6	
		IOL=8mA, VCC=1.8V			0.5	
VOH	RESETピンを除く (注6) Hレベル出力電圧 (注4) (注5)	IOH=-10mA, VCC=5V	4.3			
		IOH=-5mA, VCC=3V	2.5			
		IOH=-2mA, VCC=1.8V	1.4			
IIL	I/OピンLowレベル入力漏れ絶対電流	VCC=5.5V		<0.05	1	μA
IIH	I/OピンHighレベル入力漏れ絶対電流	確実なH/L範囲		<0.05	1	
IAC	アナログ比較器入力漏れ電流	VCC=5V, VIN=VCC/2	-50		50	nA
RRST	RESETピンプルアップ抵抗	VCC=5.5V, Low入力	30		60	kΩ
RPU	I/Oピンプルアップ抵抗	VCC=5.5V, Low入力	20		50	kΩ
ICC	活動動作消費電流 (注10)	VCC=2V, 1MHz		0.2	0.4	mA
		VCC=3V, 4MHz		1.2	2	
		VCC=5V, 8MHz		3.9	5	
	アイドル動作消費電流 (注10)	VCC=2V, 1MHz		0.03	0.1	
		VCC=3V, 4MHz		0.2	0.4	
		VCC=5V, 8MHz		0.9	1.5	
パワーダウン動作消費電流 (注11)	VCC=3V, WDT許可			1.8	4	μA
	VCC=3V, WDT禁止			0.1	2	

注1: 25°Cでの代表値です。

注2: Lowレベルの認識が保証される最高電圧です。

注3: Highレベルの認識が保証される最低電圧です。

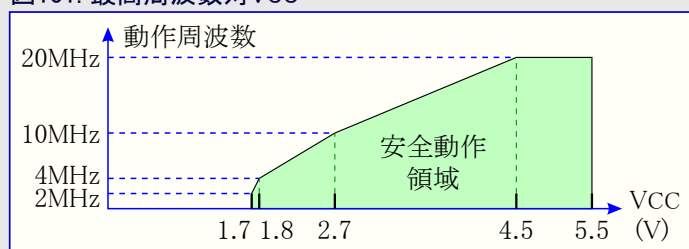
注4~注11は次頁へ

- 注4:** 安定状態(非過渡時)下でI/Oポートは検査条件よりも多くの吸い込み/吐き出しの電流を流すことができますが、ポートAとポートBの合計電流は100mAを超えてはいけません。また、ポートCとポートDの合計電流は120mAを超えてはいけません。ピンまたはポートの電流が与えられた限度を超える場合、VOL/VOHは仕様と合致することが保証されません。
- 注5:** ピンは与えられた供給電圧で一覧にされるそれらよりも多くの吸い込み/吐き出しの電流を流すことは保証されません。
- 注6:** RESETピンはプログラミング動作での操作と移行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I/Oピンと比べて弱い駆動能力を持ちます。188頁の「I/Oとしてのリセットピン」と192頁の「I/Oとしてのリセットピン」をご覧ください。
- 注7:** 標準吸い込み能力を持つポート：PD0,PD3。
- 注8:** 高吸い込み能力を持つポート：PA7~0,PB7~0,PC7~0,PD1。
- 注9:** 追加高吸い込み能力を持つポート：PC7~0。53頁の「PHDE - ポート高駆動許可レジスタ」をご覧ください。
- 注10:** 外部クロックと24頁の「消費電力の最小化」で記述された方法を使った結果。電力削減は完全に許可(PRR=\$FF)され、I/O駆動なしです。
- 注11:** 低電圧検出器(BOD)禁止。

## 24.3. 速度

デバイスの最高動作周波数は供給電圧(VCC)に依存します。供給電圧と最高動作周波数の関連は図101.で示されるように区分的線形です。

図101. 最高周波数対VCC



## 24.4. クロック特性

### 24.4.1. 校正付き8MHz発振器の精度

工場既定校正よりも高い精度に内部8MHz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197頁の「内部発振器周波数」で得られます。

表104. 校正付き内部8MHz発振器の校正精度

校正種別	目標周波数	VCC	温度	校正精度 (注1)
工場校正	8.0MHz	3V	25°C	±2% (注2)
使用者校正 (注3)	7.3~8.1MHz内	1.7~5.5V内	-40~85°C内	±10% (注2)
				±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)

**注2:** 2者選択については206頁のデバイス注文符号をご覧ください。

**注3:** ATtiny828Rデバイスで利用不可

### 24.4.2. 校正付き32kHz発振器の精度

工場既定校正よりも高い精度に内部32kHz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197頁の「内部発振器周波数」で得られます。

表105. 校正付き内部32kHz発振器の校正精度

校正種別	目標周波数	VCC	温度	校正精度
工場校正	32kHz	1.7~5.5V	-40~85°C	±30%

### 24.4.3. 外部クロック信号駆動

図102. 外部クロック駆動波形

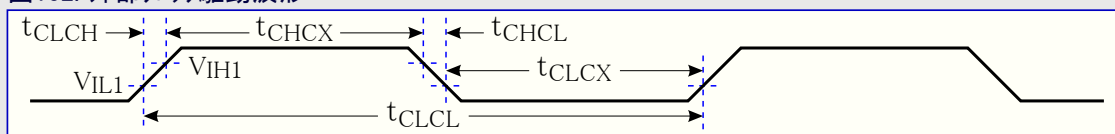


表106. 外部クロック駆動特性

シンボル	項目	VCC=1.7~5.5V		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	最小	最大	
$1/t_{CLCL}$	クロック周波数	0	2	0	4	0	8	0	12	MHz
$t_{CLCL}$	クロック周期	500		250		125		83		
$t_{CHCX}$	Highレベル時間	200		100		40		20		ns
$t_{CLCX}$	Lowレベル時間	200		100		40		20		
$t_{CLCH}$	上昇時間		2.0		2.0		1.6		0.5	$\mu$ s
$t_{CHCL}$	下降時間		2.0		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		2	%

### 24.5. システムとリセットの特性

表107. リセットと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
$V_{RST}$	RESETピン閾値電圧		$0.2V_{CC}$		$0.9V_{CC}$	V
$t_{RST}$	RESETピンでの最小リセットパルス幅	$V_{CC}=1.8V$		2		
		$V_{CC}=3V$		0.7		$\mu$ s
		$V_{CC}=5V$		0.4		
$V_{BG}$	基準電圧	$V_{CC}=2.7V, T_A=25^\circ C$	1.0	1.1	1.2	V

注: 値は指針だけです。

#### 24.5.1. 電源ONリセット

表108. 強化された電源ONリセットの特性 ( $T_A=-40^\circ C \sim 85^\circ C$ )

シンボル	項目	最小	代表	最大	単位
$V_{POR}$	電源ONリセット開放閾値電圧 (注1)	1.1	1.4	1.6	V
$V_{POA}$	電源ONリセット活性閾値電圧 (注2)	0.6	1.3	1.6	
$SR_{ON}$	電源投入時上昇率	0.01			V/ms

注: 値は指針だけです。

注1: 電圧上昇時にデバイスがリセットから開放される閾値電圧です。

注2: 供給電圧が $V_{POA}$ 未満でなければ電源ONリセットは動作しません(電圧下降時)。

#### 24.5.2. 低電圧検出 (BOD)

表109. BODLEVELヒューズ符号化( $V_{BOT}$ )

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

## 24.6. 温度感知器

表110. 工場校正での温度感知器の精度

シンボル	項目	条件	最小	代表	最大	単位
ATS	精度	VCC=4V, TA=25~85°C		10		°C

注: ・ファームウェアは工場校正値に基づいて温度を計算します。

・最小と最大の値は保証されません。より高い精度が求められる場合は最寄のAtmel営業所にお問い合わせください。

## 24.7. 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111.で一覧にされるパラメータは製造に於いて検査されません。シンボルは図103.を参照します。

図103. 2線直列バス タイミング

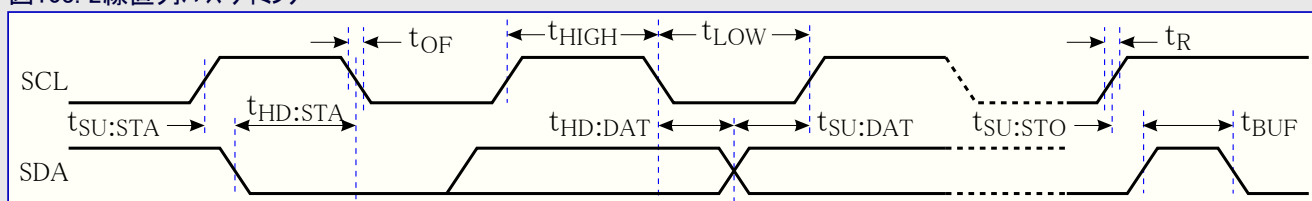


表111. 2線直列インターフェース特性

シンボル	項目	条件	最小	最大	単位
VIL	Lowレベル入力電圧		-0.5	0.3VCC	V
VIH	Highレベル入力電圧		0.7VCC	VCC+0.5	
VHYS	シュミットトリガ入力ヒステリシス電圧	VCC ≥ 2.7V VCC < 2.7V	0.05VCC 0	-	
VOL	Lowレベル出力電圧	IOL=3mA, VCC ≥ 2.7V IOL=2mA, VCC < 2.7V	0	0.4	ns
tSP	入力パルス最小幅(尖頭消去濾波)		0	50	
fSCL	SCLクロック周波数 (注)		0	400	kHz
tHD:STA	(再送)開始条件保持時間		0.6	-	μs
tLOW	SCLクロックLowレベル時間		1.3	-	
tHIGH	SCLクロックHighレベル時間		0.6	-	
tSU:STA	再送開始条件準備時間		0.6	-	ns
tHD:DAT	データ保持時間		0	0.9	
tSU:DAT	データ準備時間		100	-	μs
tSU:STO	停止条件準備時間		0.6	-	
tBUF	停止条件→開始条件間バス開放時間		1.3	-	

注: f<sub>CK</sub>=CPUクロック周波数



## 24.8. A/D変換器特性

表112. A/D変換器特性 (TA=-40°C~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、微分非直線性、量子化、利得、オフセットの各誤差を含む)	変換クロック=200kHz		2		
		変換クロック=1MHz		3		
		VCC=4V VREF=4V 変換クロック=200kHz 雑音低減動作		1.5		LSB
		変換クロック=1MHz 雑音低減動作		2.5		
	積分非直線性誤差(INL) (オフセット、利得校正後精度)			1		
	微分非直線性誤差(DNL)	VCC=4V, VREF=4V 変換クロック=200kHz		0.5		
	利得誤差			2.5		
	(絶対)オフセット誤差			1.5		
	変換時間	連続変換動作(自由走行変換)	15		300	μs
	変換クロック周波数		50		1000	kHz
V <sub>IN</sub>	入力電圧		GND		VREF	V
	入力周波数帯域			38.5		kHz
R <sub>AIN</sub>	アナログ入力インピーダンス			100		MΩ
	A/D変換出力		0		1023	LSB

## 24.9. アナログ比較器特性

表113. 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
V <sub>AIO</sub>	入力変位(オフセット)電圧	VCC=5V, V <sub>in</sub> =VCC/2		<10	40	mV
I <sub>LAC</sub>	入力漏れ電流	VCC=5V, V <sub>in</sub> =VCC/2	-50		50	nA
t <sub>APD</sub>	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		
t <sub>DPD</sub>	デジタル伝播遅延	VCC=1.7~5.5V		1	2	clk

## 24.10. 並列プログラミング特性

図104. 並列プログラミング タイミング (一般的な必要条件)

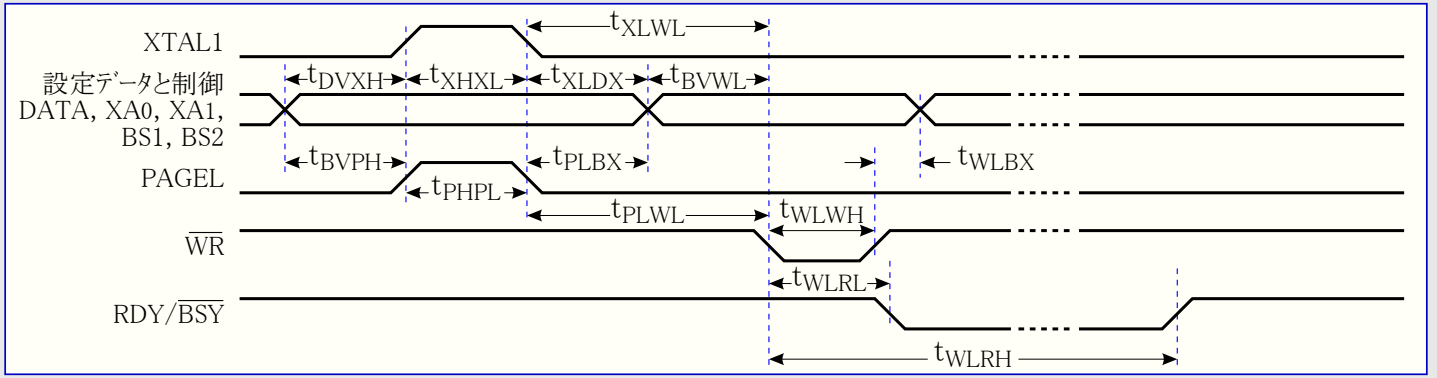
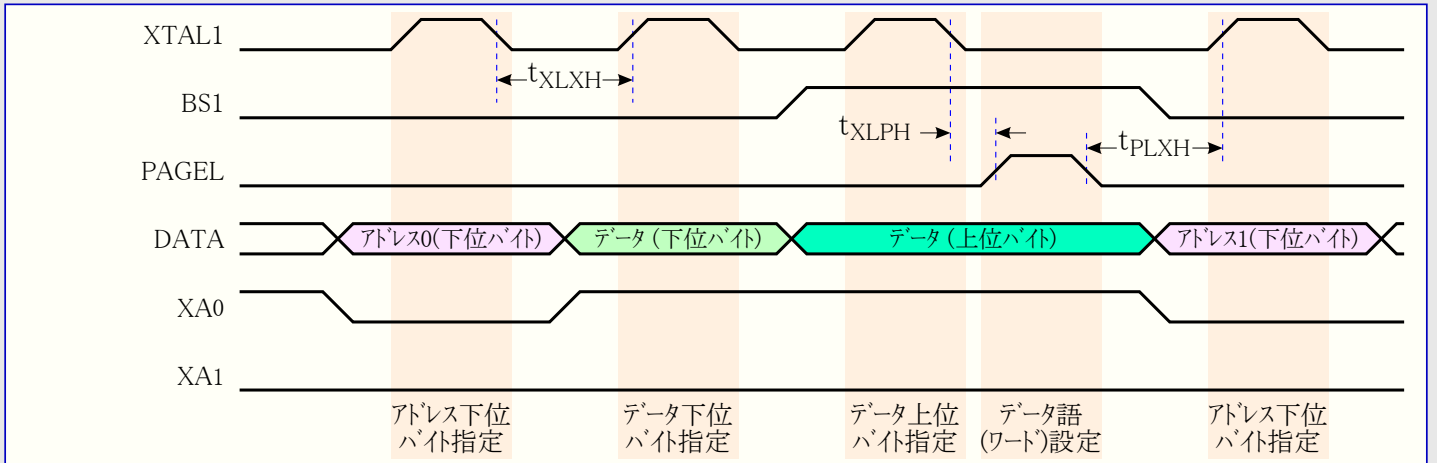
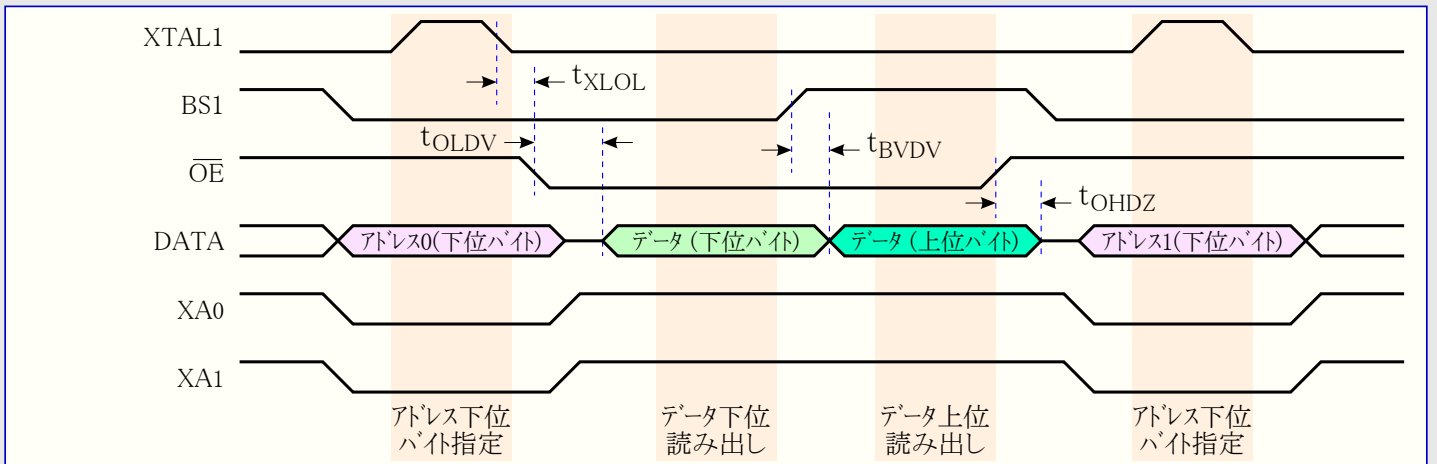


図105. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図104.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106. 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104.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114. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
V <sub>PP</sub>	プログラミング許可電圧	11.5		12.5	V
I <sub>PP</sub>	プログラミング許可電流			250	μA
t <sub>DVXH</sub>	XTAL1 ↑ に対するデータと制御の準備時間	67			
t <sub>XLXH</sub>	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
t <sub>XHXL</sub>	XTAL1 Highパルス幅	150			
t <sub>XLDX</sub>	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
t <sub>XLWL</sub>	XTAL1パルス ↓ 後の $\overline{WR}$ ↓ 待機時間	0			
t <sub>XLPH</sub>	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
t <sub>PLXH</sub>	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			ns
t <sub>BVPH</sub>	PAGELパルス ↑ に対するBS1準備時間	67			
t <sub>PHPL</sub>	PAGEL Highパルス幅	150			
t <sub>PLBX</sub>	PAGELパルス ↓ 後のBS1保持時間	67			
t <sub>WLBX</sub>	$\overline{WR}$ パルス ↓ 後のBS1,BS2保持時間	67			
t <sub>PLWL</sub>	PAGELパルス ↓ 後の $\overline{WR}$ パルス ↓ 待機時間	67			
t <sub>BVWL</sub>	$\overline{WR}$ パルス ↓ に対するBS1準備時間	67			
t <sub>WLWH</sub>	$\overline{WR}$ Lowパルス幅	150			
t <sub>WLRL</sub>	$\overline{WR}$ パルス ↓ 後のRDY/ $\overline{BSY}$ ↓ 遅延時間	0		1	μs
t <sub>WLRH</sub>	書き込み時間 ( $\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注1)	3.7		4.5	ms
t <sub>WLRH_CE</sub>	チップ消去時間 ( $\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注2)	7.5		9	ms
t <sub>XLCL</sub>	XTAL1パルス ↓ 後の $\overline{OE}$ ↓ 待機時間	0			
t <sub>BVDV</sub>	BS1有効からのDATA遅延時間	0		250	ns
t <sub>OLDV</sub>	$\overline{OE}$ ↓ 後のDATA出力遅延時間			250	
t <sub>OHDZ</sub>	$\overline{OE}$ ↑ 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。

## 24.11. 直列プログラミング特性

図107. 直列プログラミング タイミング

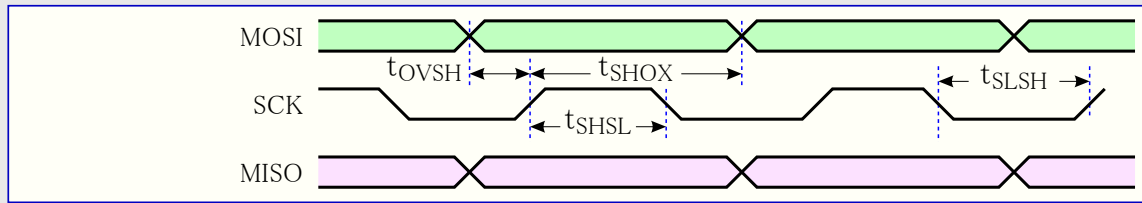


図108. 直列プログラミング波形

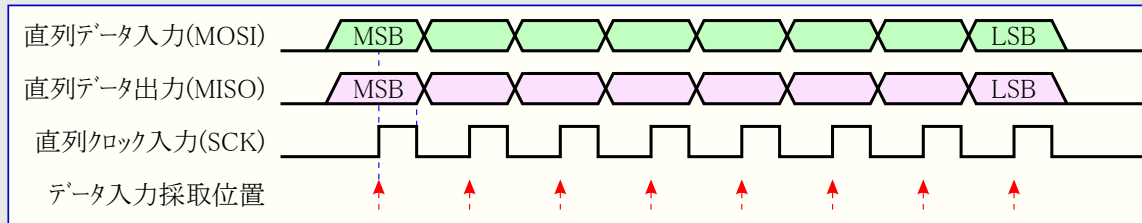


表115. 直列プログラミング特性 (TA=-40°C~85°C、VCC=1.7~5.5V、他の特記事項を除く)

シンボル	項目	最小	代表	最大	単位
1/tCLCL	発振器周波数	1.7~4.5V	0	4	MHz
		4.5~5.5V	0	20	
tCLCL	発振器周期	1.7~4.5V	250		
		4.5~5.5V	50		
tSHSL	SCKパルスHレベル幅	2tCLCL (注)			ns
tSLSH	SCKパルスLレベル幅	2tCLCL (注)			
tOVSH	SCK ↑ に対するMOSI準備時間	tCLCL			
tSHOX	SCK ↑ に対するMOSI保持時間	2tCLCL			

注:  $f_{CK} < 12\text{MHz}$  に対して  $2t_{CLCL}$ 、 $f_{CK} \geq 12\text{MHz}$  に対して  $3t_{CLCL}$

## 25. 代表特性

本項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。特性付けの間、デバイスは検査限界よりも高い周波数で動作していますが、それらは注文コードが示すよりも高い周波数での正しい機能が保証される訳ではありません。

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。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。

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が、**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。**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。

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。

### 25.1. 活動動作消費電流

図109. 活動動作消費電流 対 低周波数 (100kHz~1MHz)

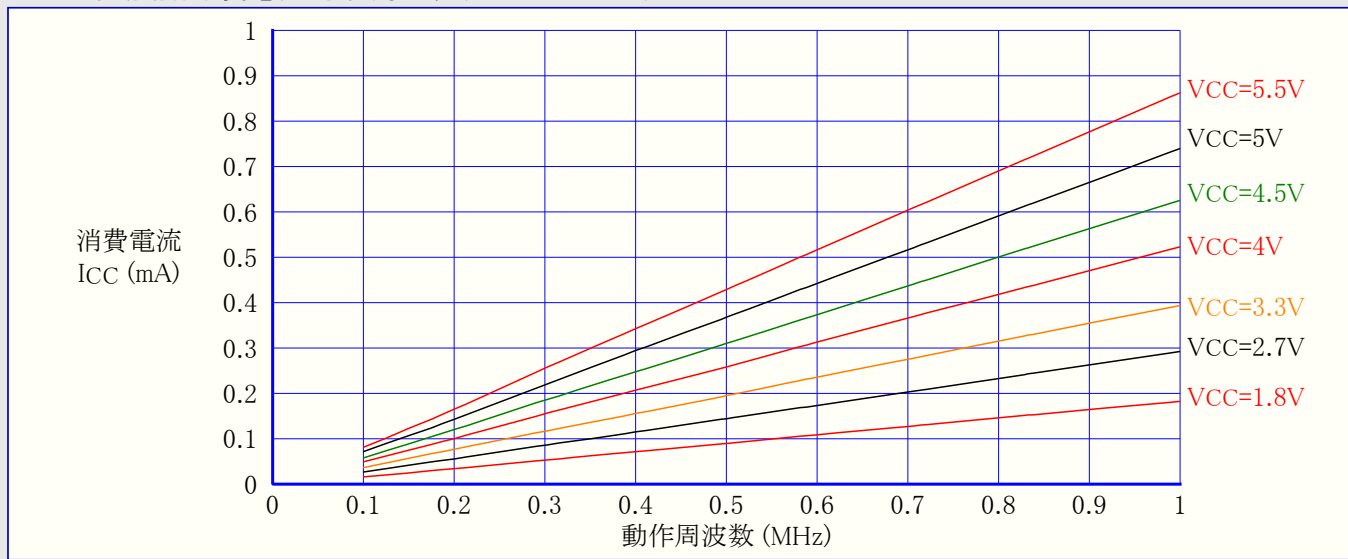


図110. 活動動作消費電流 対 周波数 (1MHz~20MHz)

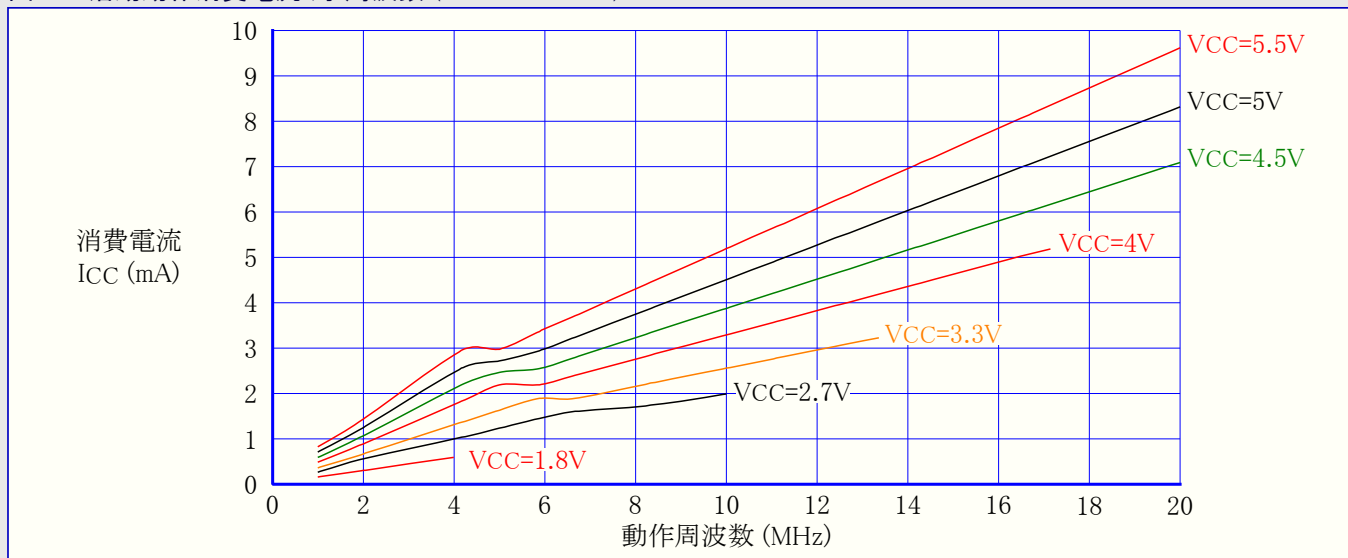




図111. 活動動作消費電流 対 動作電圧 (内部発振器,8MHz)

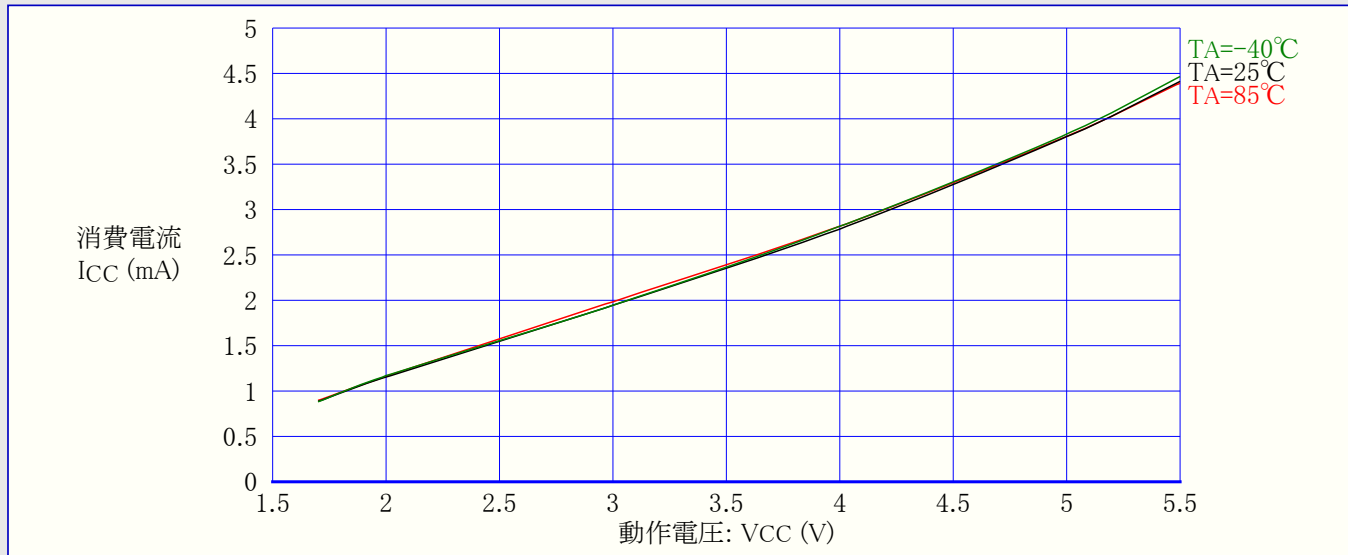


図112. 活動動作消費電流 対 動作電圧 (内部発振器,8MHzの8前置分周=1MHz)

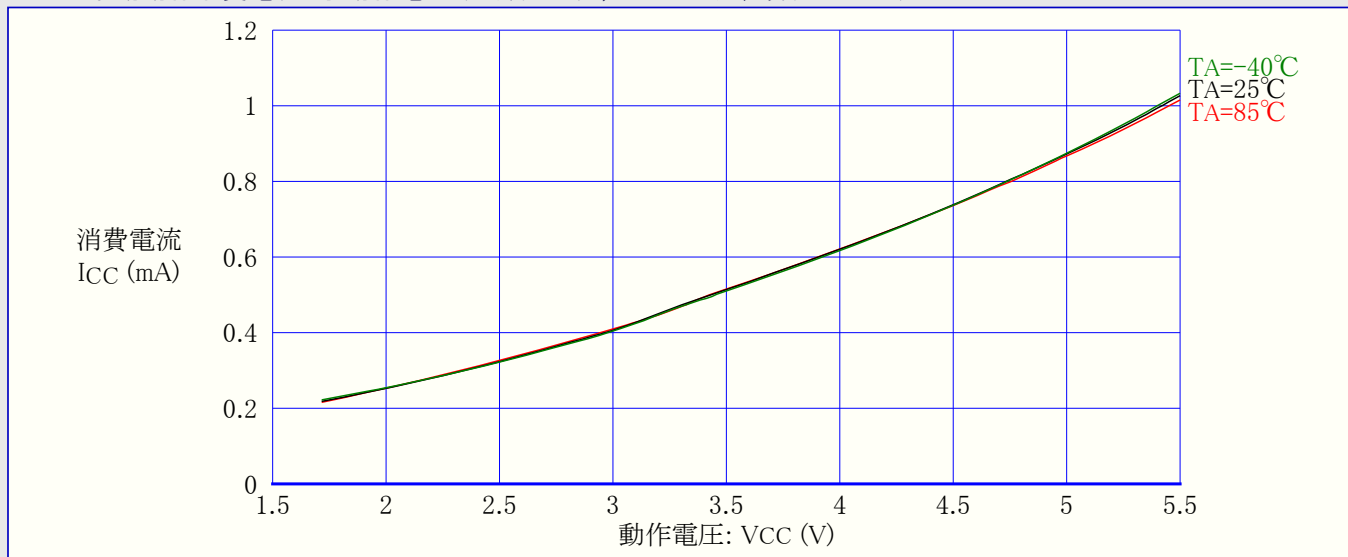
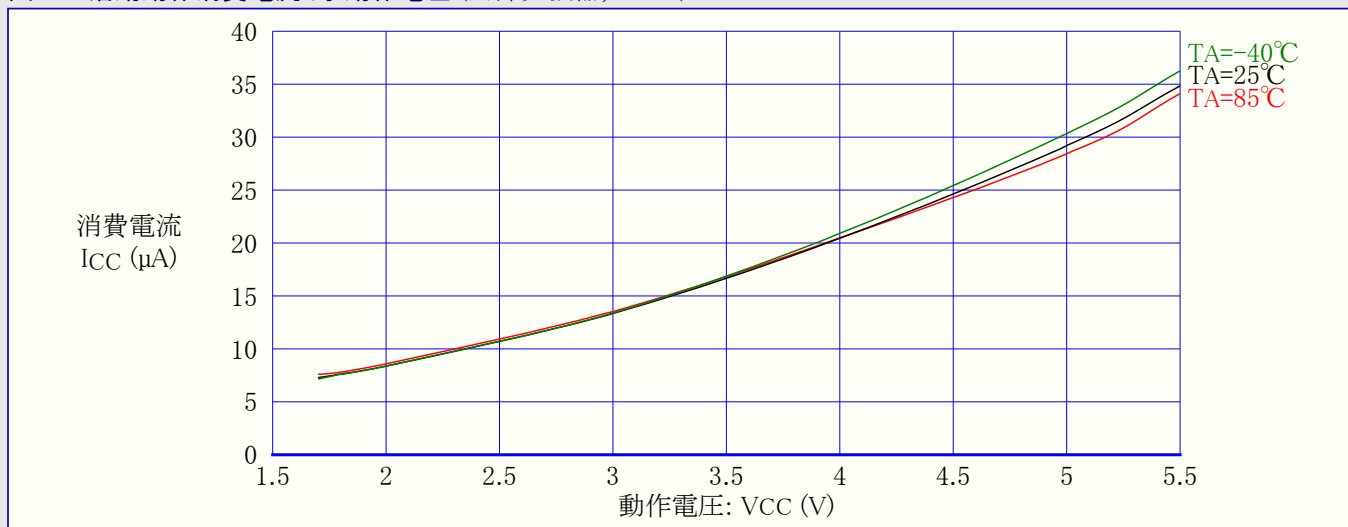


図113. 活動動作消費電流 対 動作電圧 (内部発振器,32kHz)



## 25.2. アイドル動作消費電流

図114. アイドル動作消費電流 対 低周波数 (100kHz~1MHz)

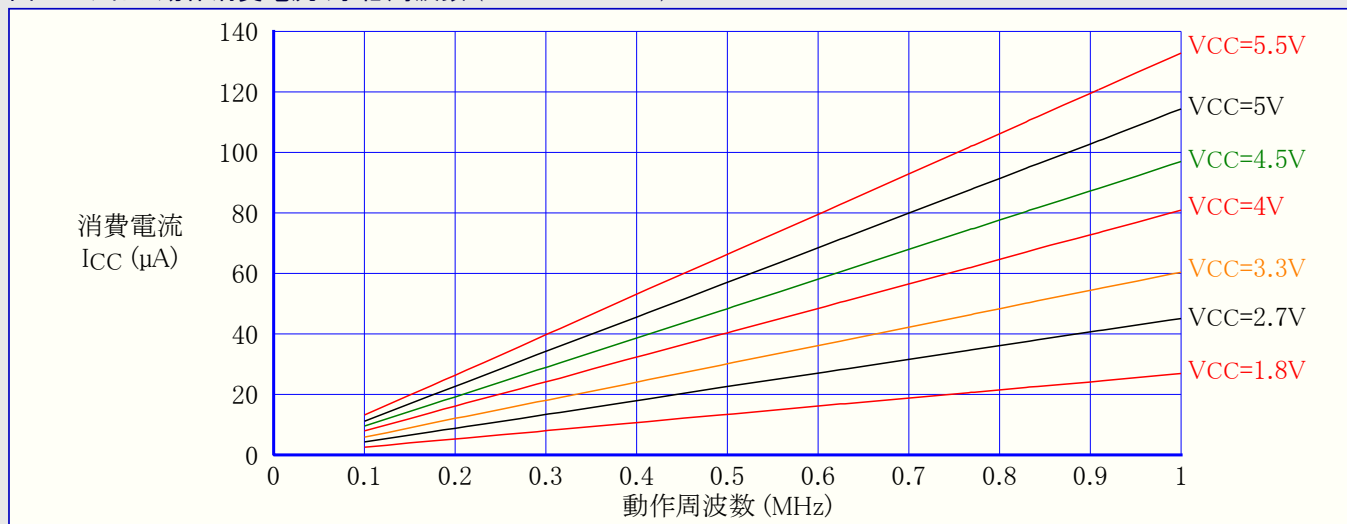


図115. アイドル動作消費電流 対 周波数 (1MHz~20MHz)

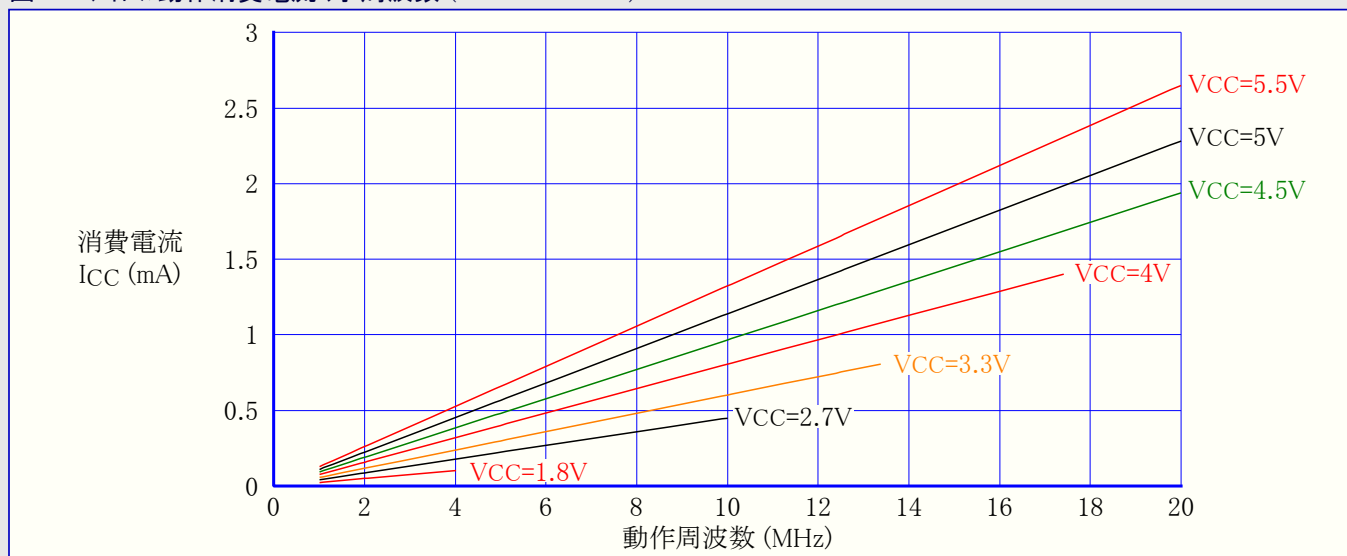


図116. アイドル動作消費電流 対 動作電圧 (内部発振器, 8MHz)

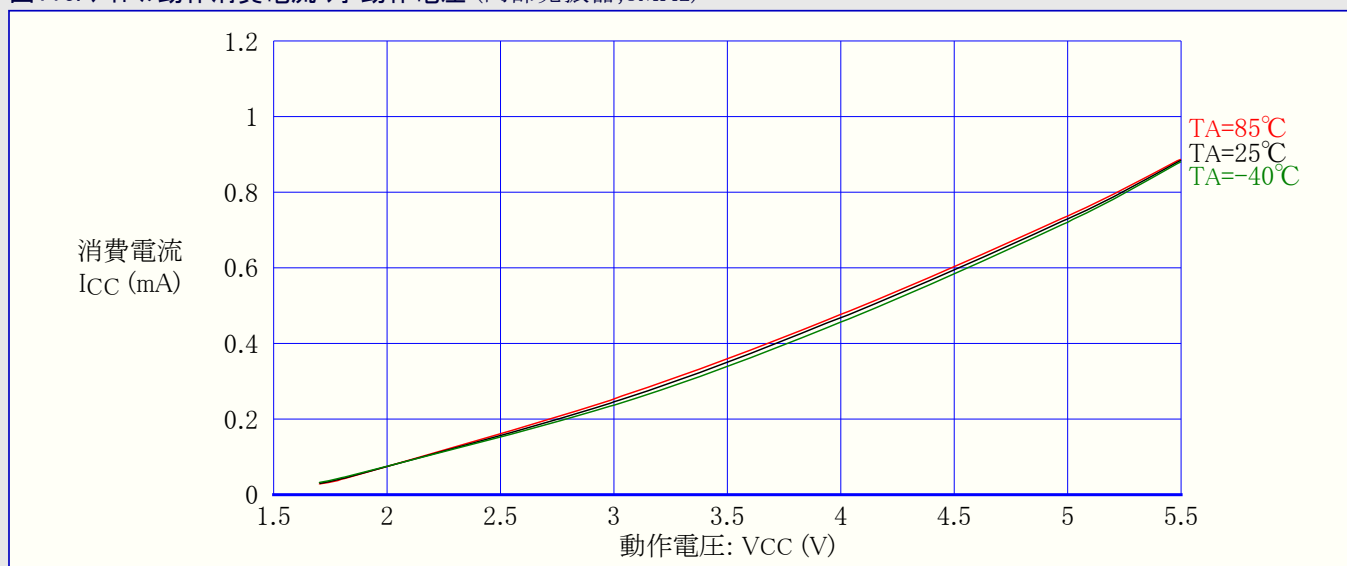


図117. アイドル動作消費電流 対 動作電圧 (内部発振器,8MHzの8前置分周=1MHz)

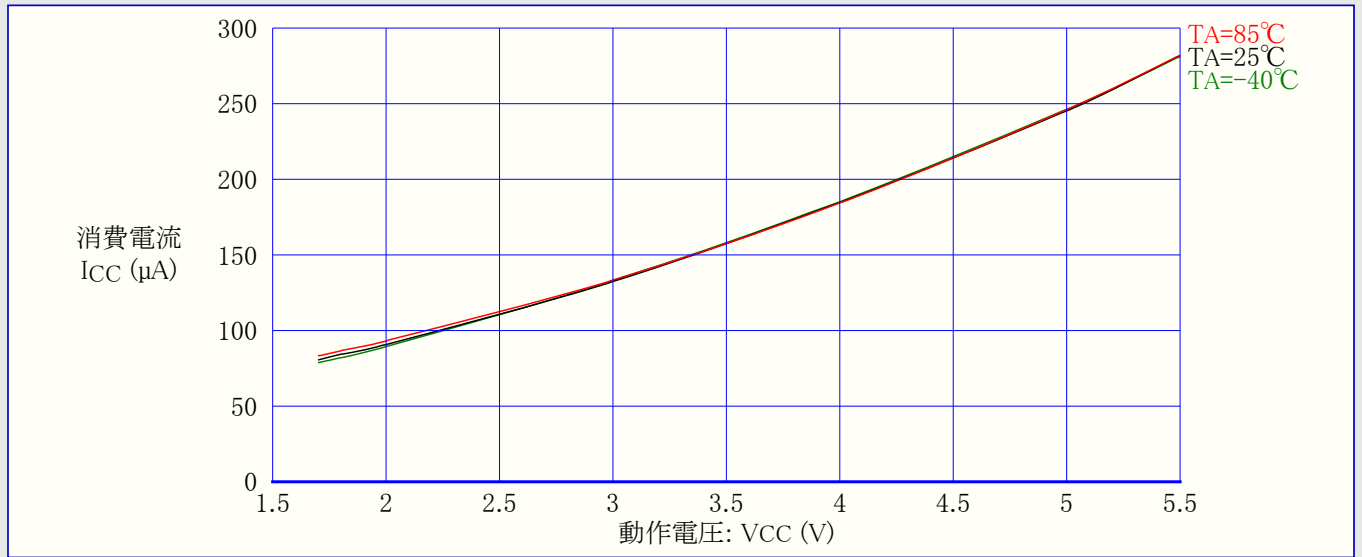
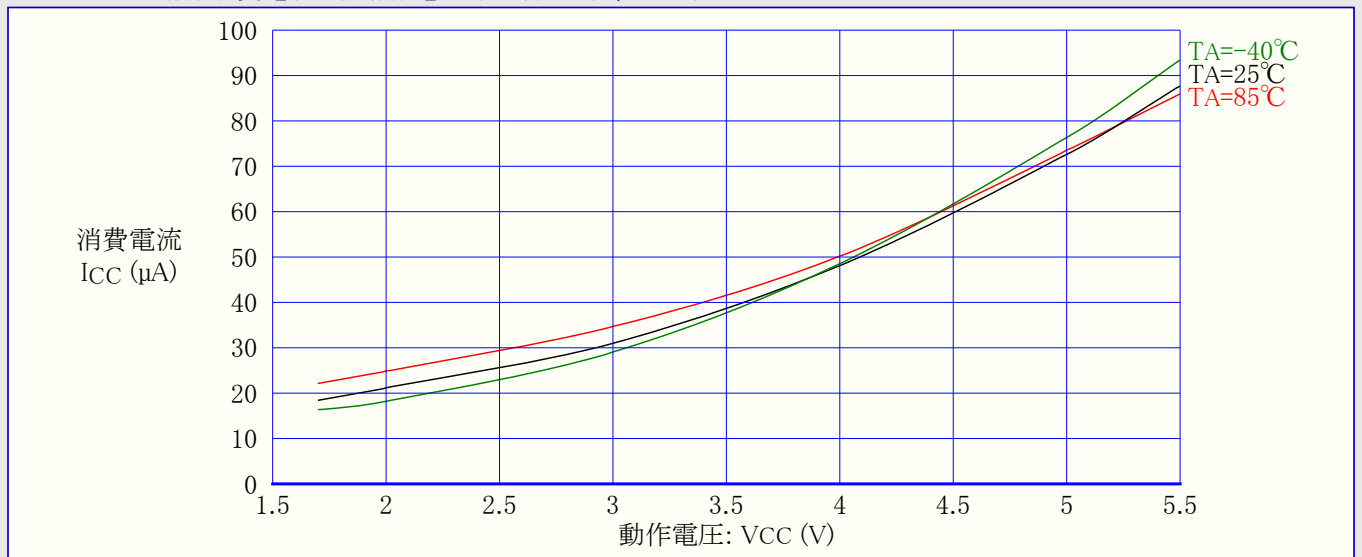


図118. アイドル動作消費電流 対 動作電圧 (内部発振器,32kHz)



### 25.3. パワーダウン動作消費電流

図119. 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)

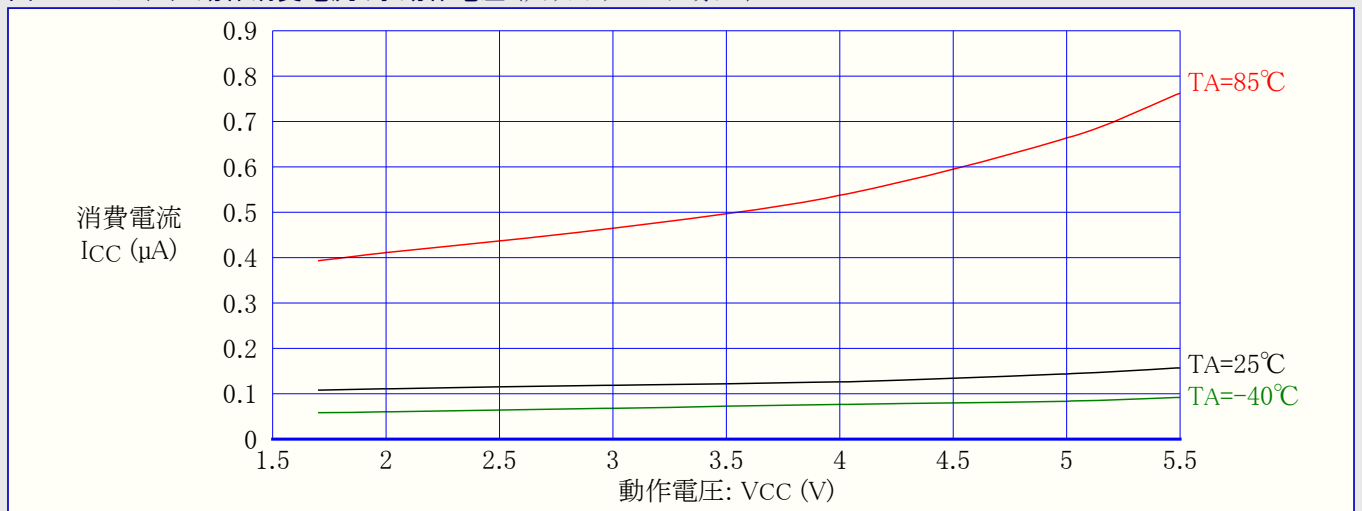
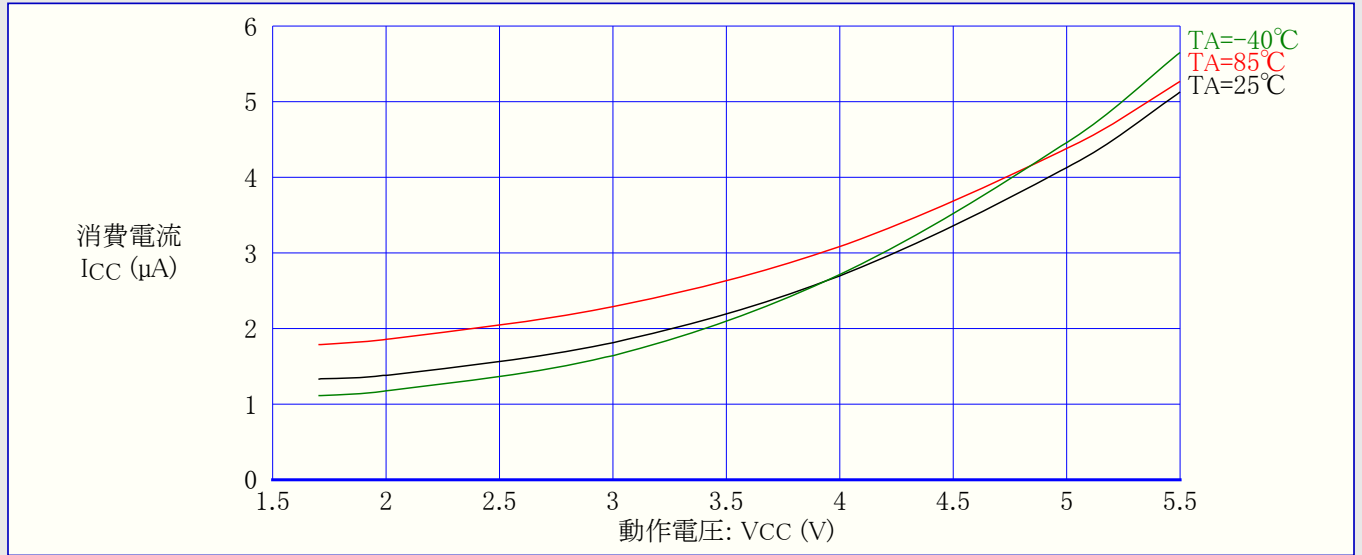


図120. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



## 25.4. リセット消費電流

図121. リセット消費電流 対 周波数 (100kHz~1MHz, プルアップ電流を除く)

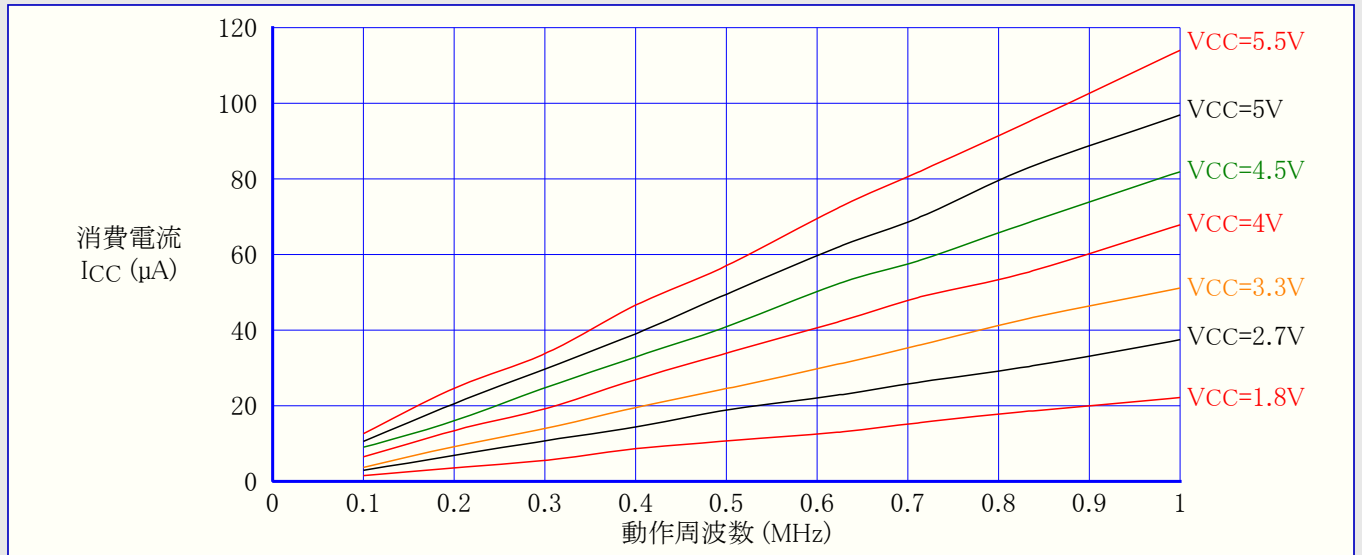


図122. リセット消費電流 対 周波数 (1MHz~20MHz, プルアップ電流を除く)

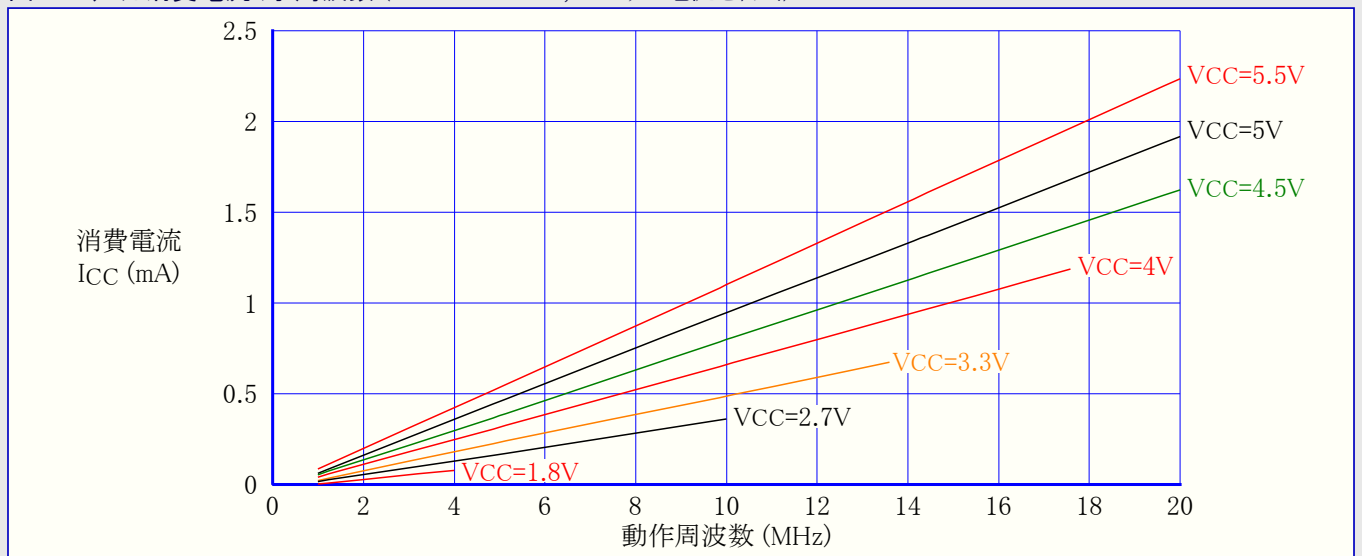
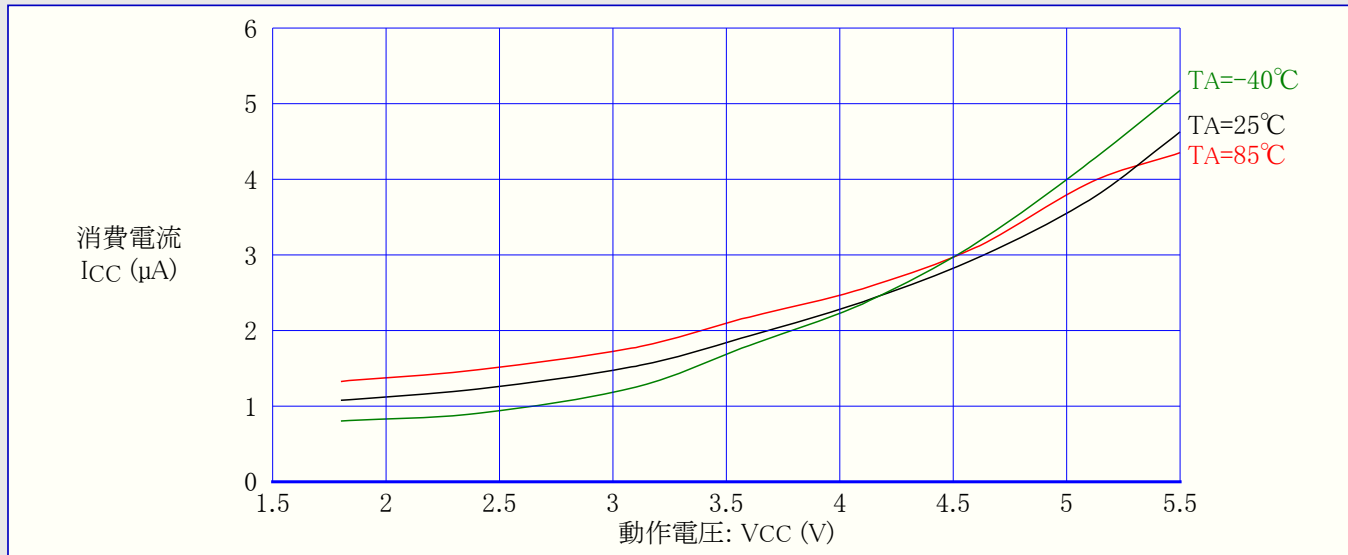


図123. リセット消費電流 対 動作電圧 (RESETフルアアップ電流を除く, クロックなし)



## 25.5. 周辺機能部消費電流

図124. 4MHzでの周辺機能の消費電流 対 動作電圧 (TA=25°C)

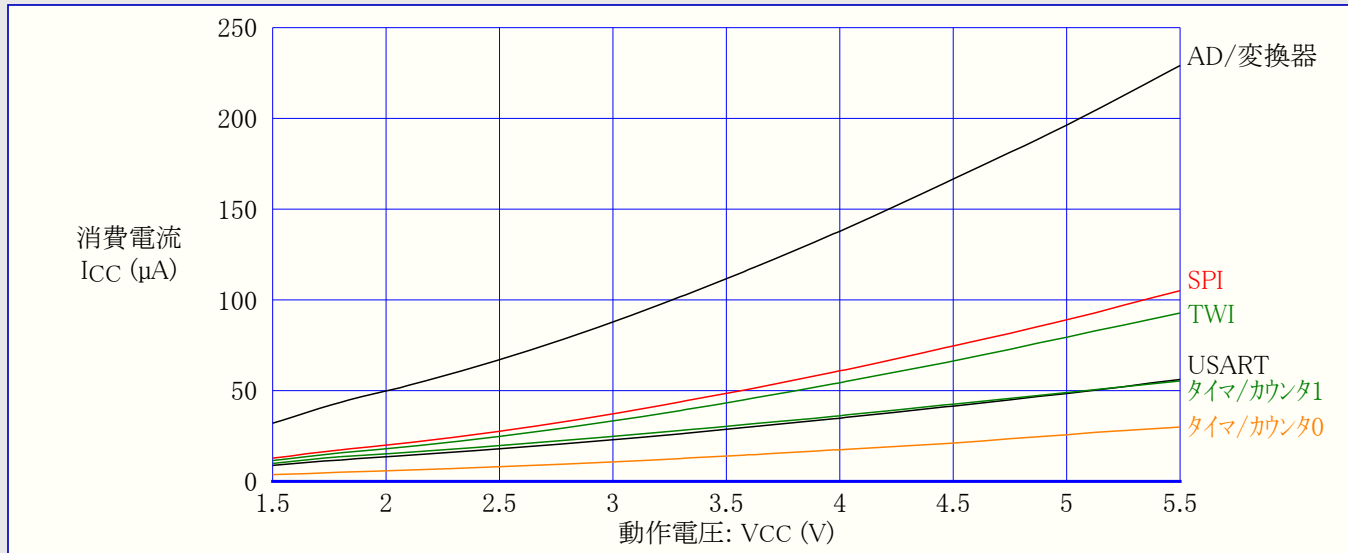


図125. 3Vでの周辺機能の消費電流 対 動作周波数 (TA=25°C)

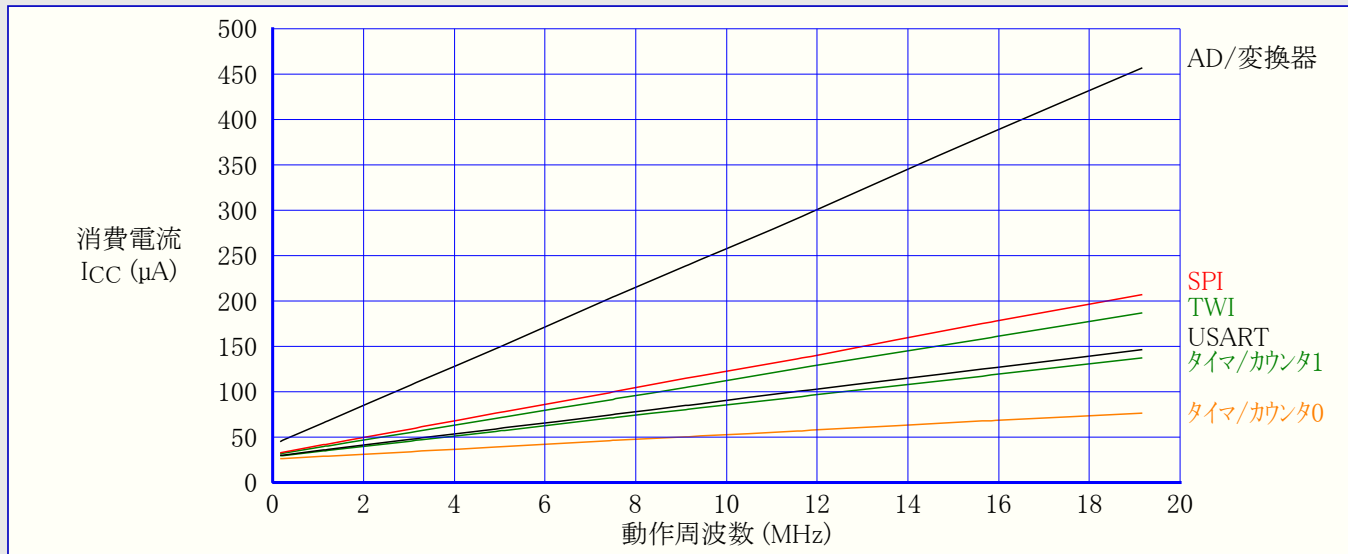




図126. ウォッチドッグ タイマ消費電流 対 動作電圧

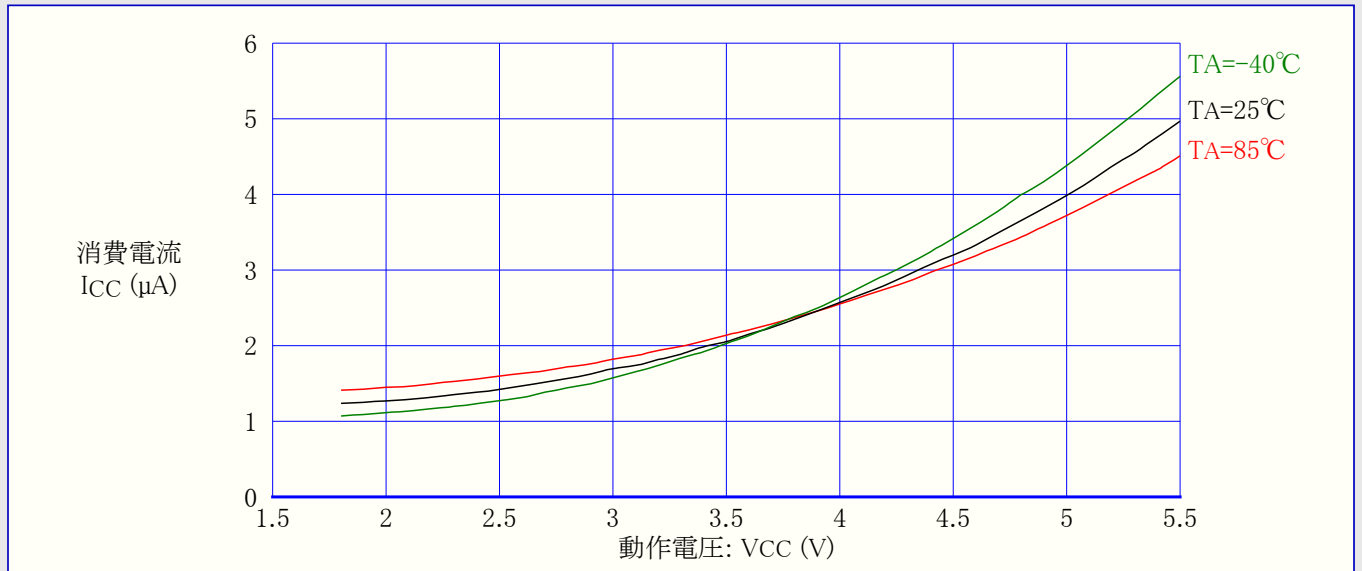


図127. 低電圧検出器(BOD)消費電流 対 動作電圧

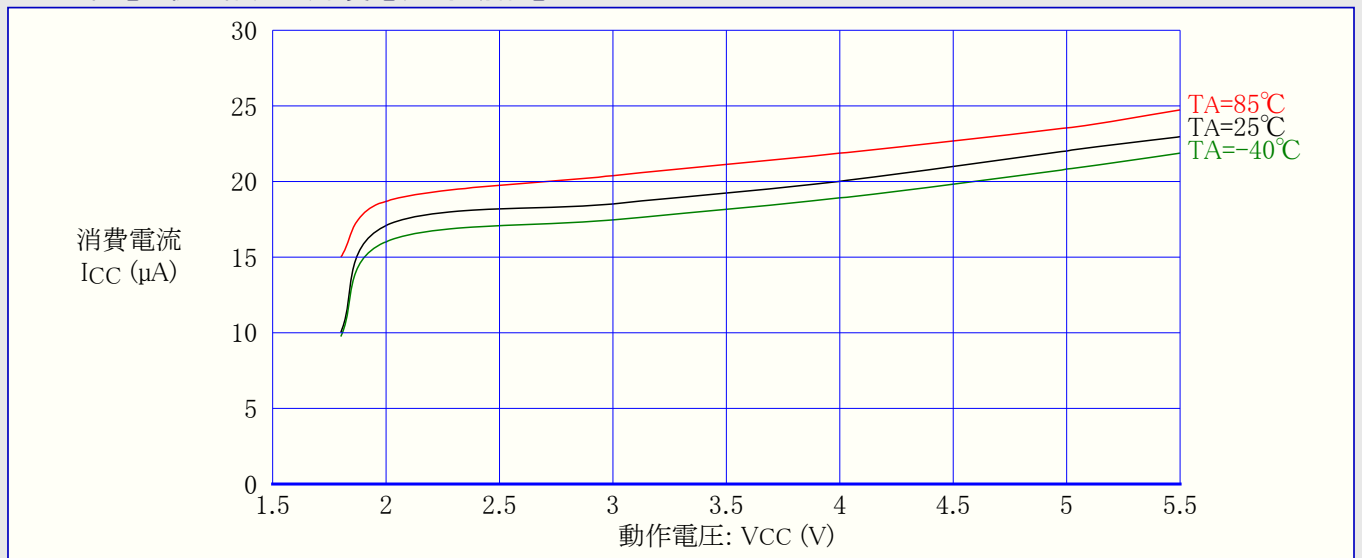
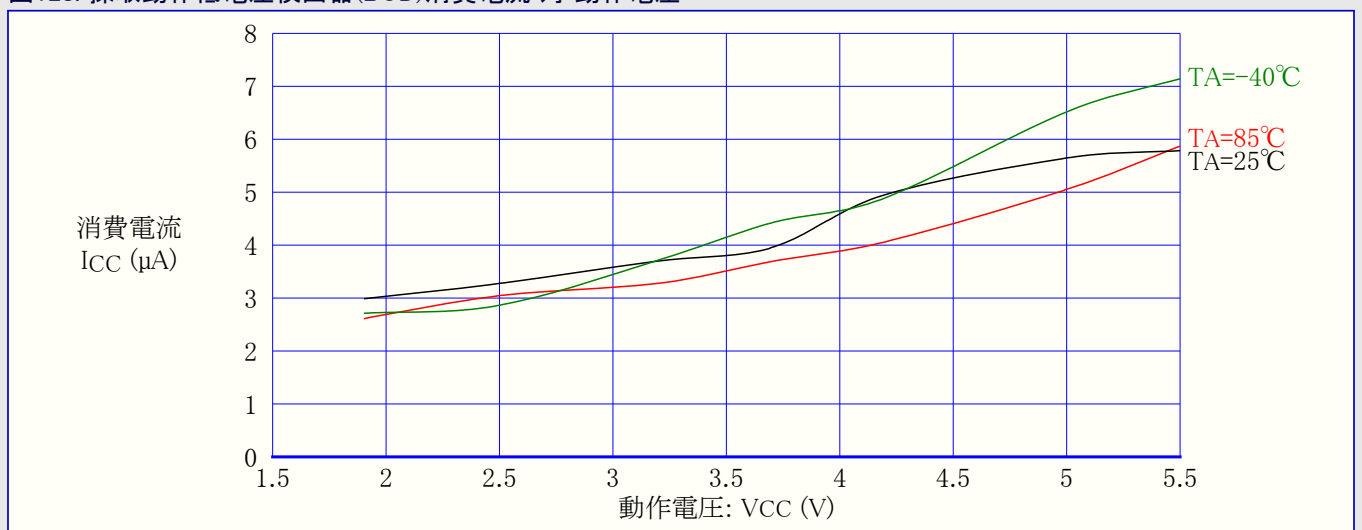


図128. 採取動作低電圧検出器(BOD)消費電流 対 動作電圧



## 25.6. プルアップ抵抗

### 25.6.1. 入出力ピン

図129. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

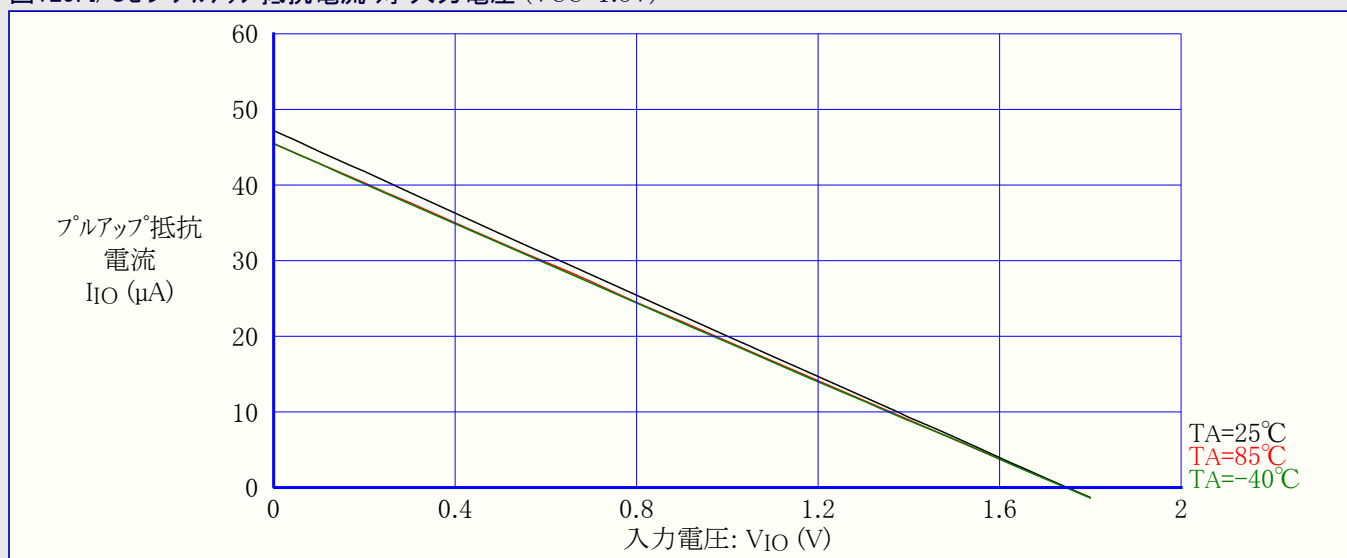


図130. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

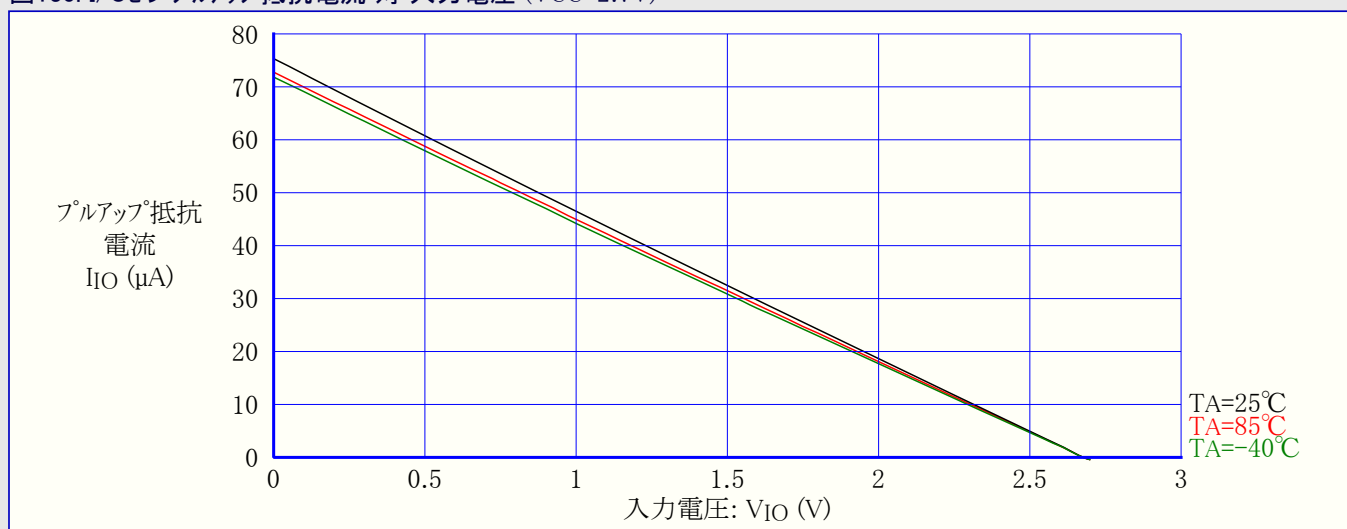
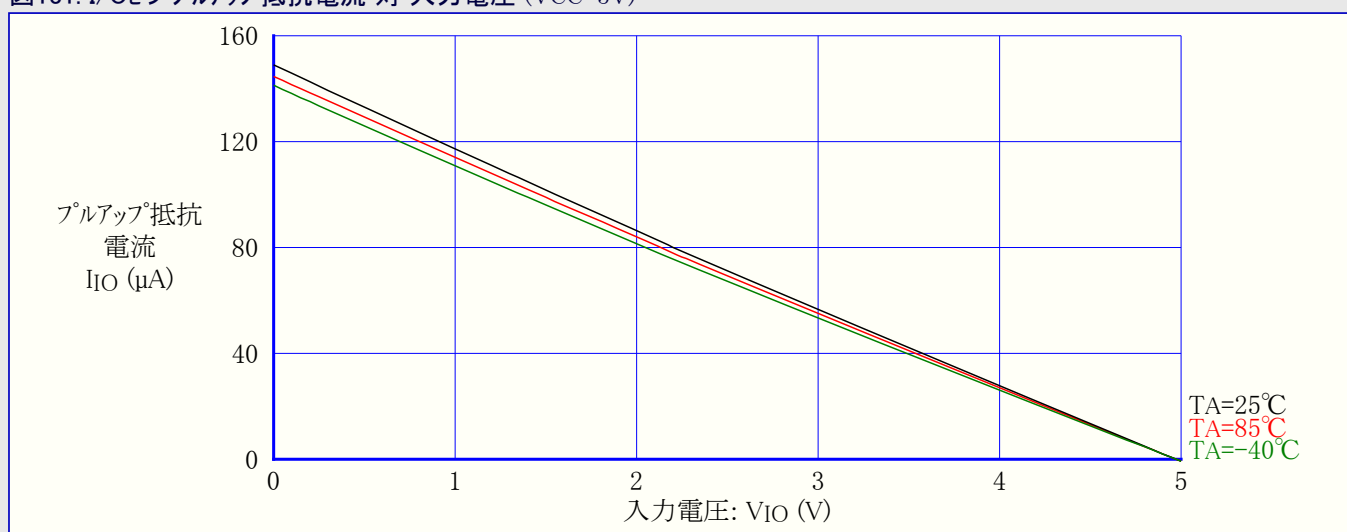


図131. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)



## 25.6.2. リセットピン

図132. RESETプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

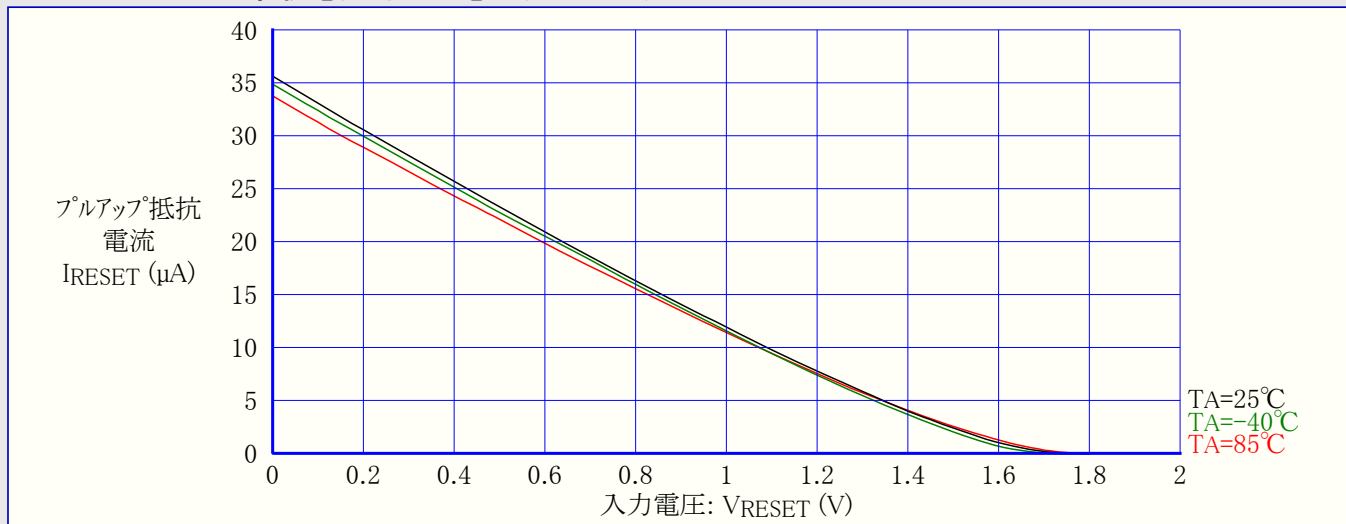


図133. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

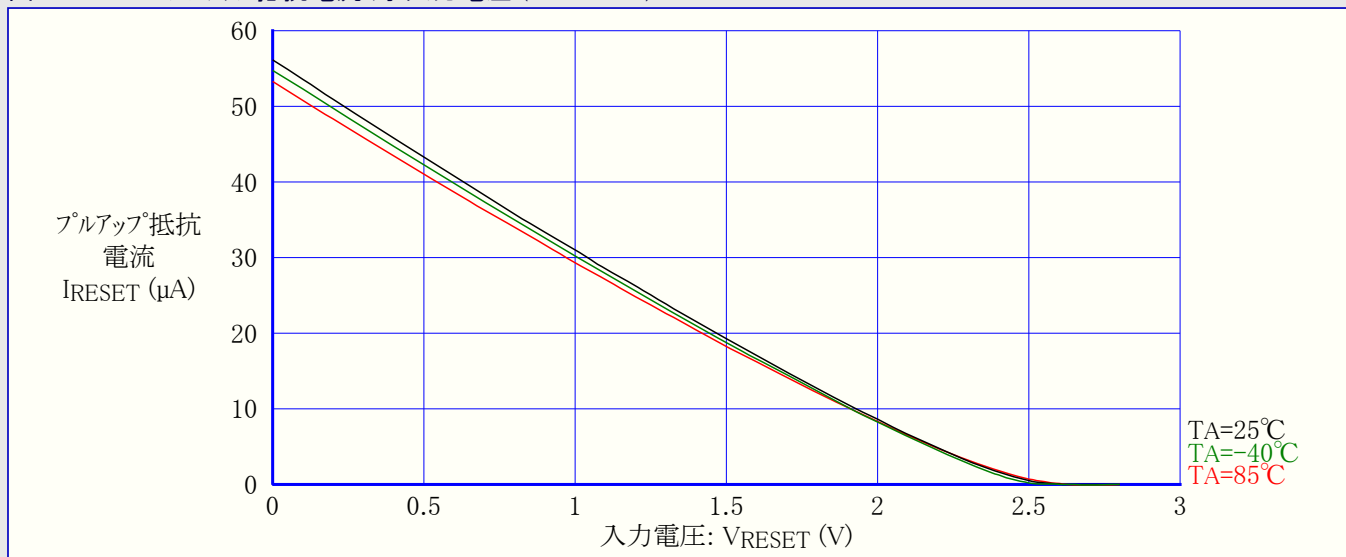
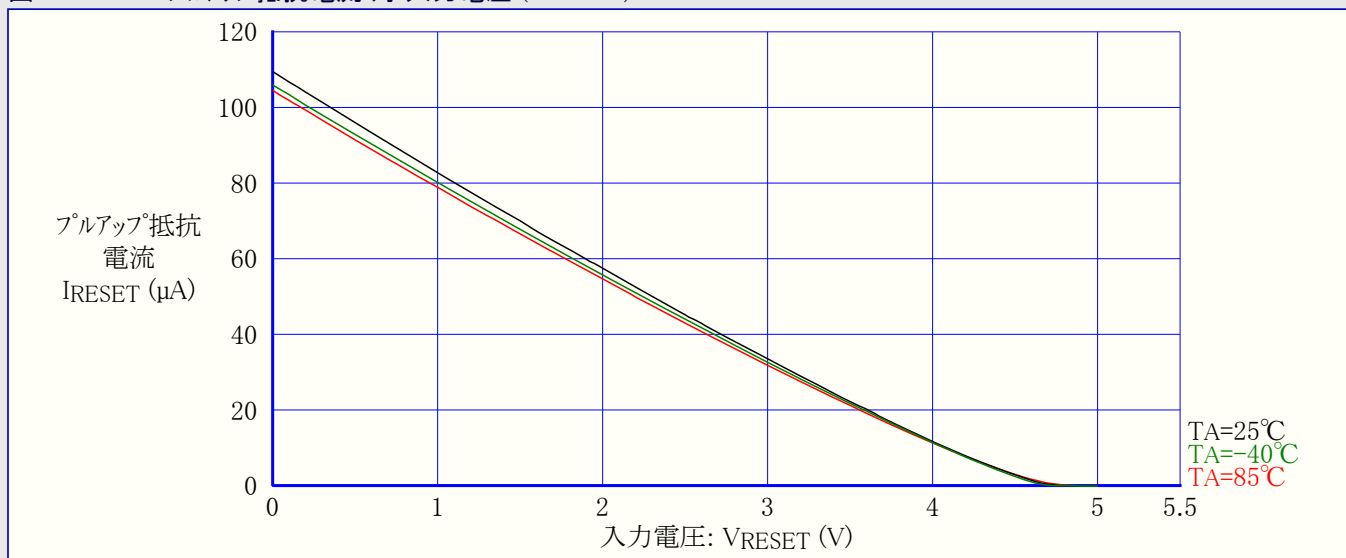


図134. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)



## 25.7. 入力閾値

### 25.7.1. 入出力ピン

図135. I/Oピン入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

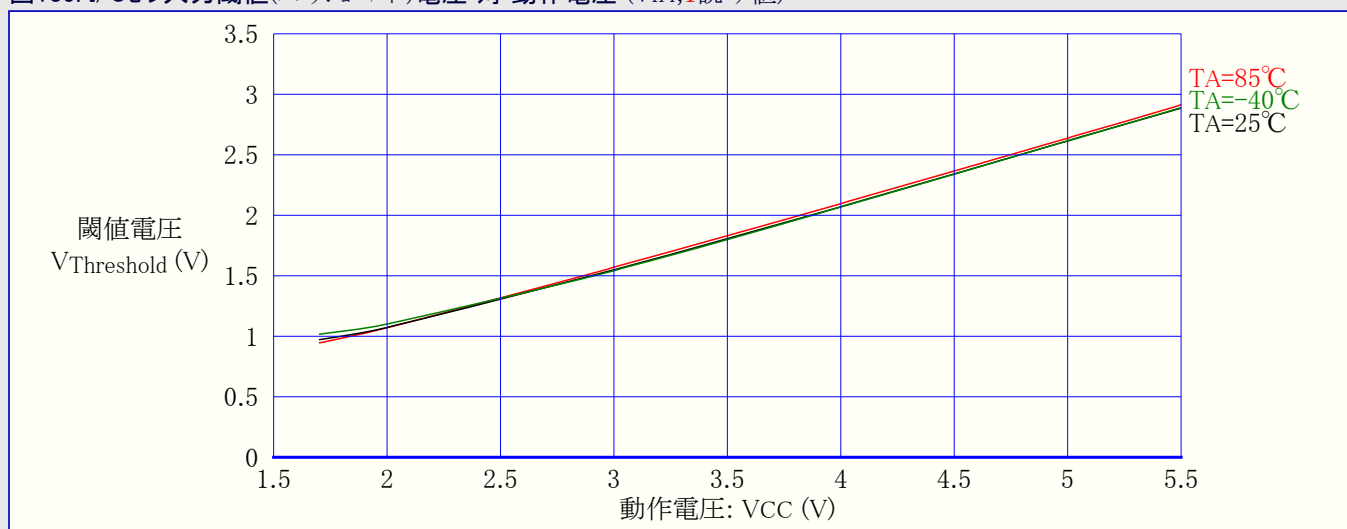


図136. I/Oピン入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

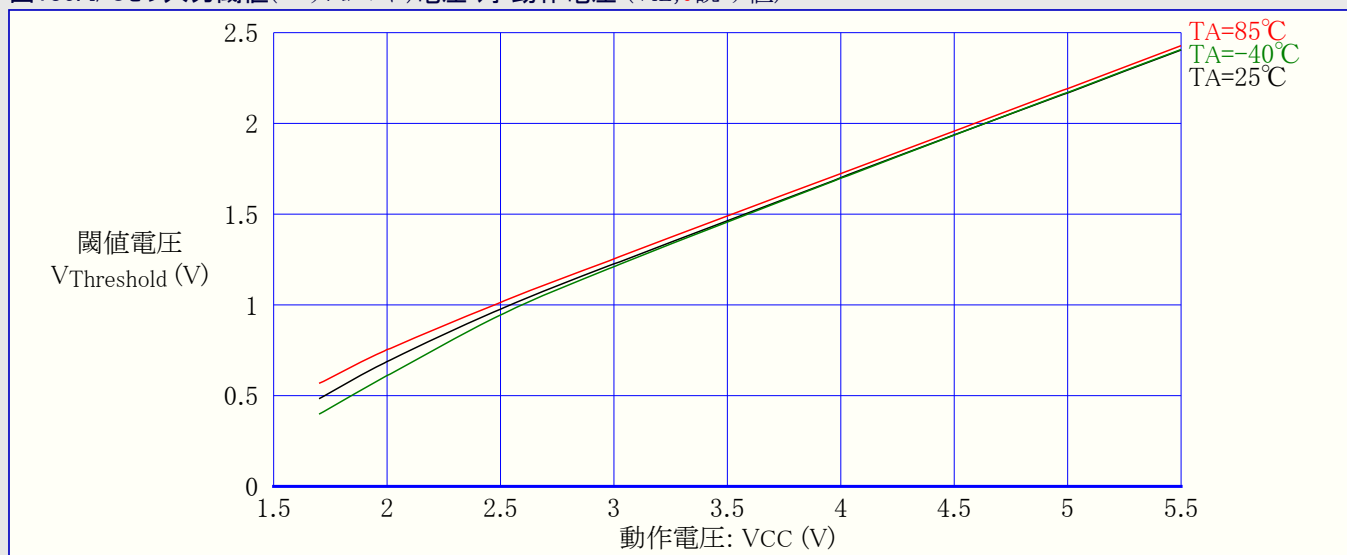
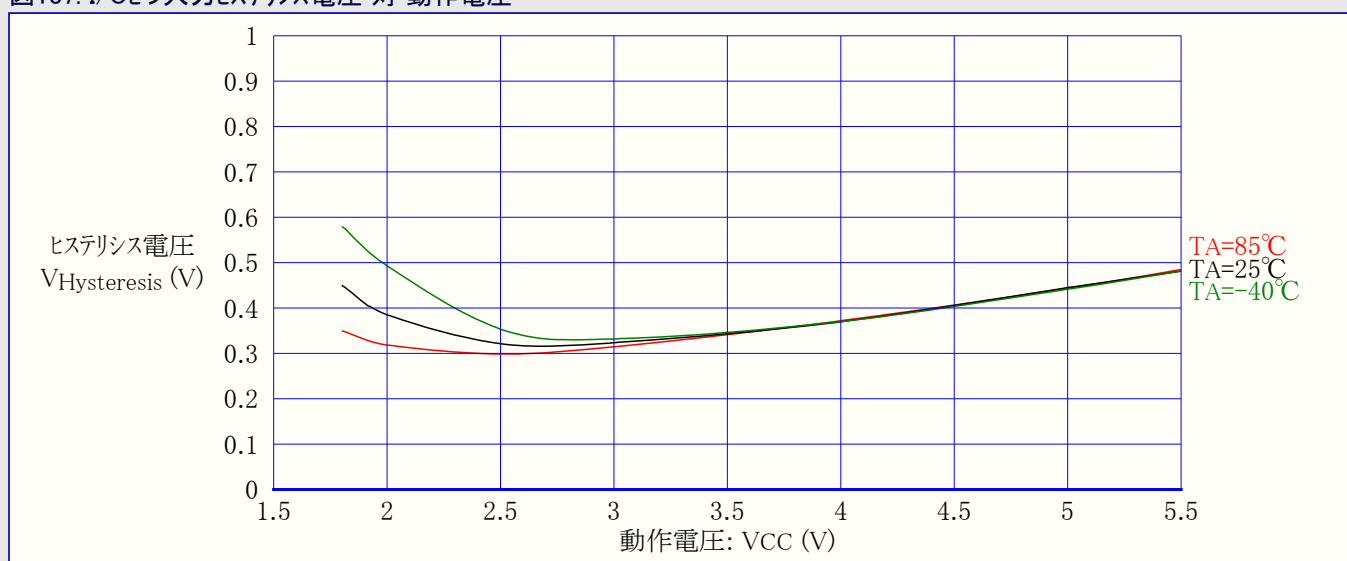


図137. I/Oピン入力ヒステリシス電圧 対 動作電圧



## 25.7.2. TWIピン

図138. TWI用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)

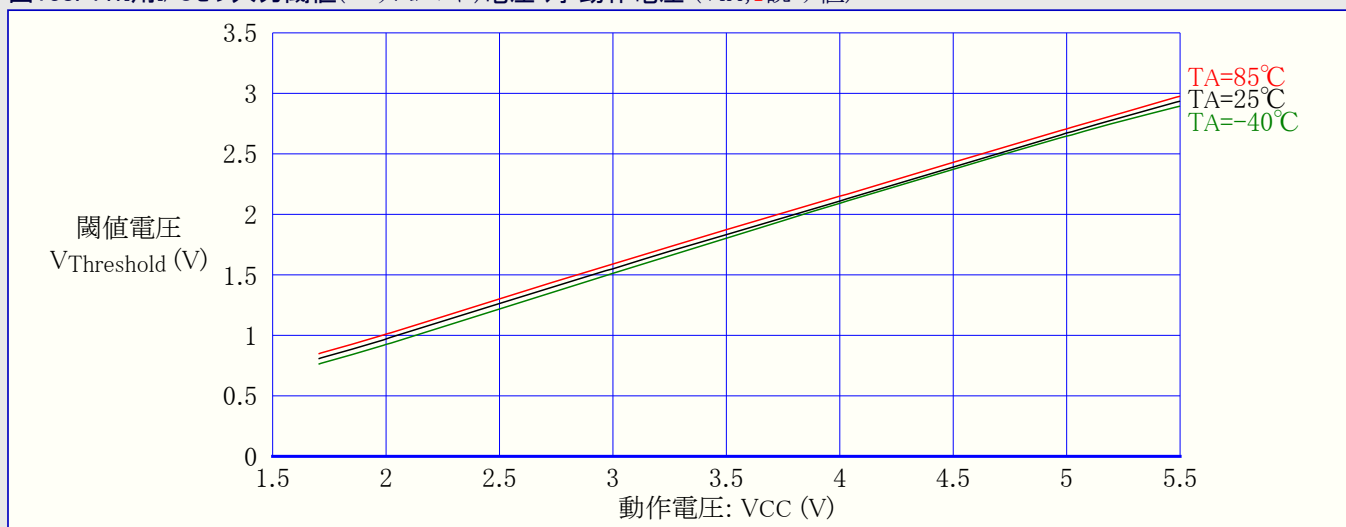


図139. TWI用I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

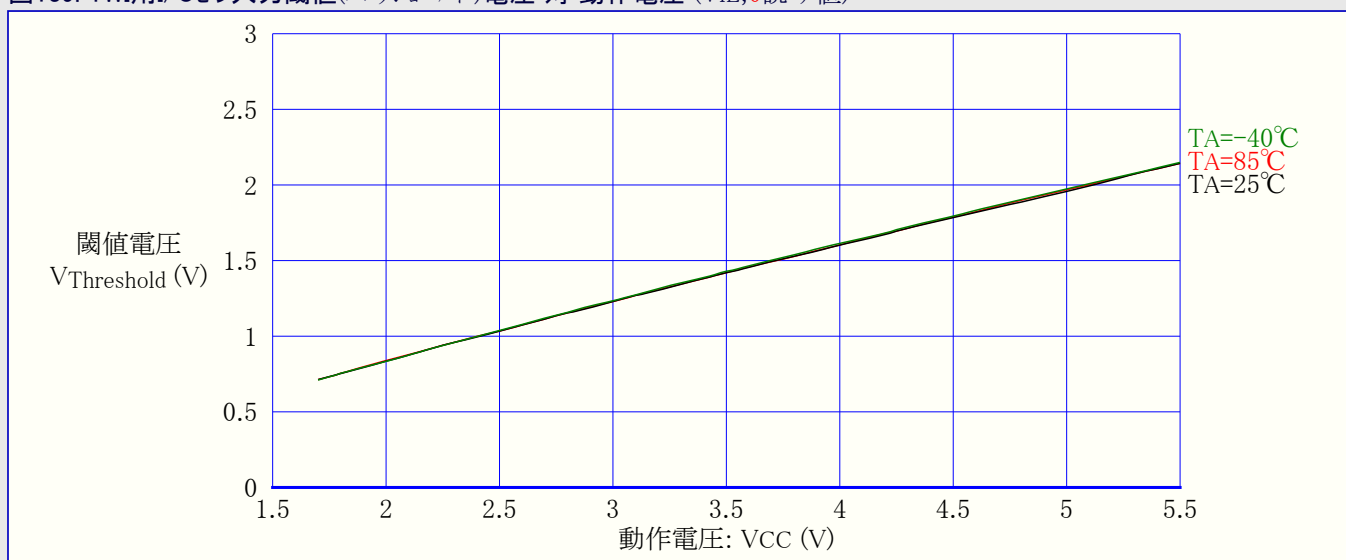
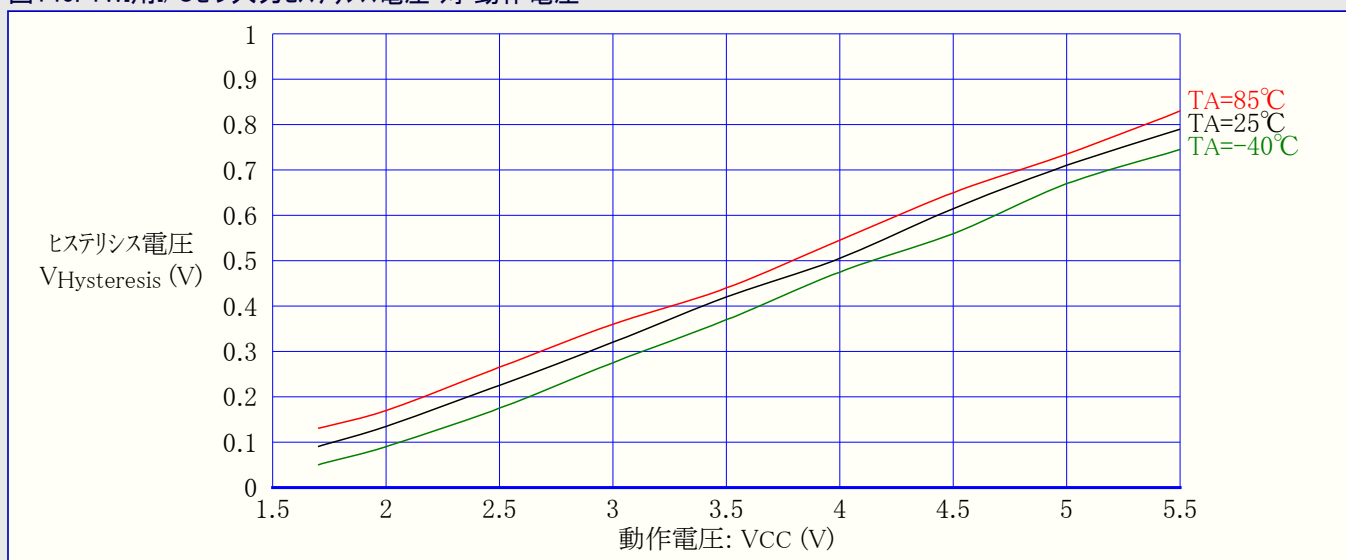


図140. TWI用I/Oピン入力ヒステリシス電圧 対 動作電圧





### 25.7.3. I/Oとしてのリセットピン

図141. 入出力としてのRESET入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IH,1}$ 読み値)

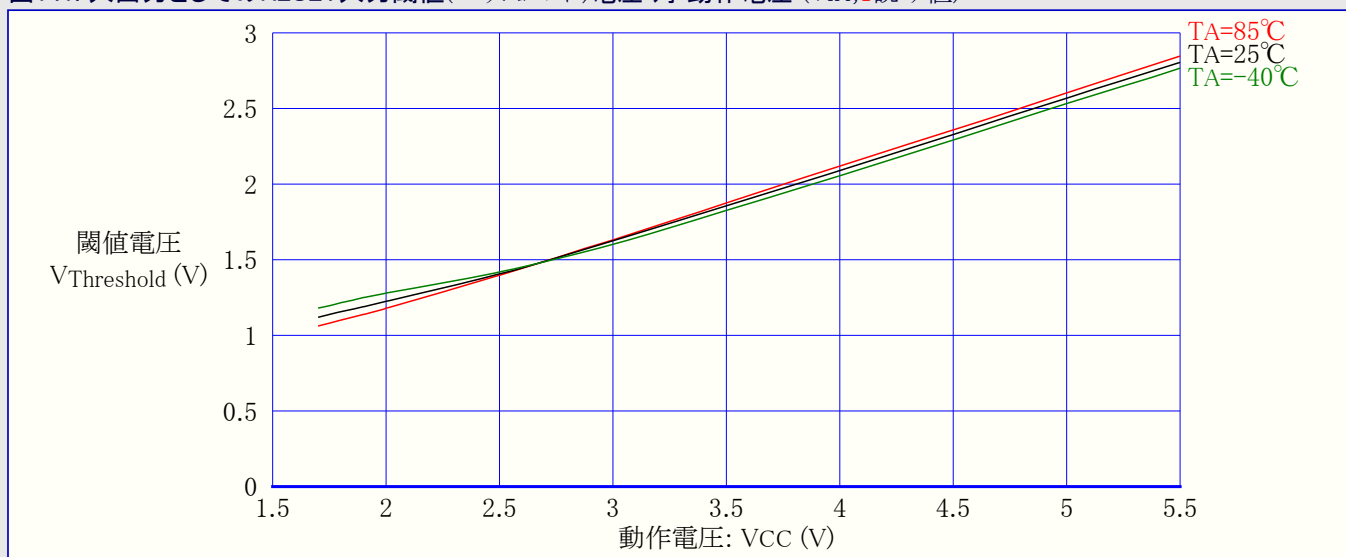


図142. 入出力としてのRESET入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IL,0}$ 読み値)

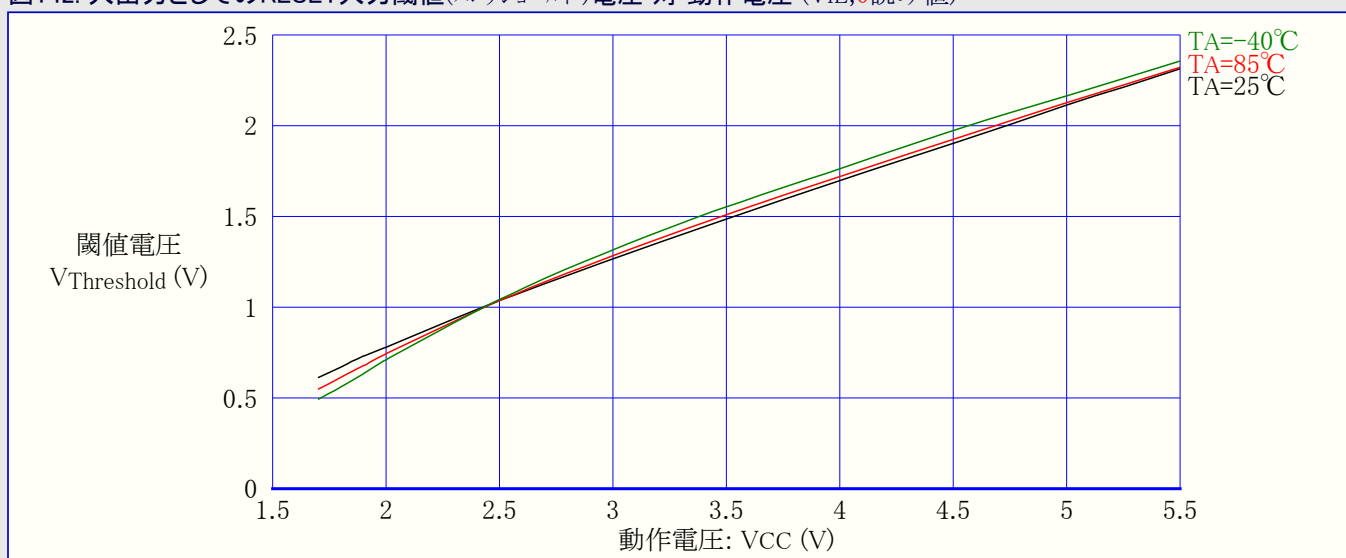
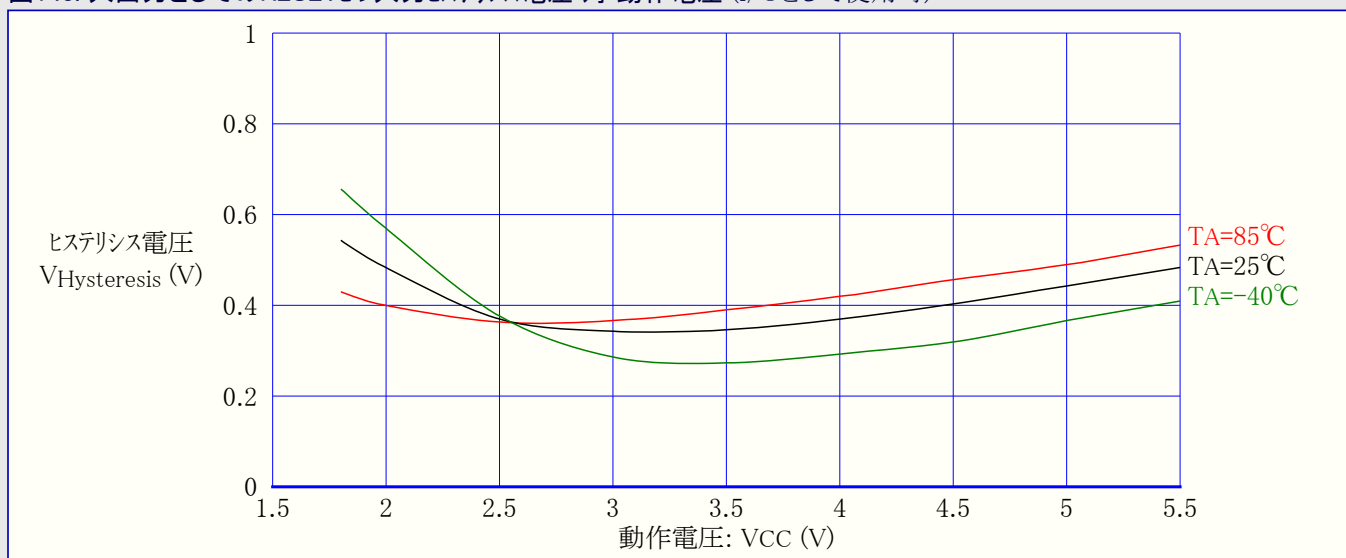


図143. 入出力としてのRESETピン入力ヒステリシス電圧 対 動作電圧 (I/Oとして使用時)



## 25.7.4. リセットピン

図144. 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)

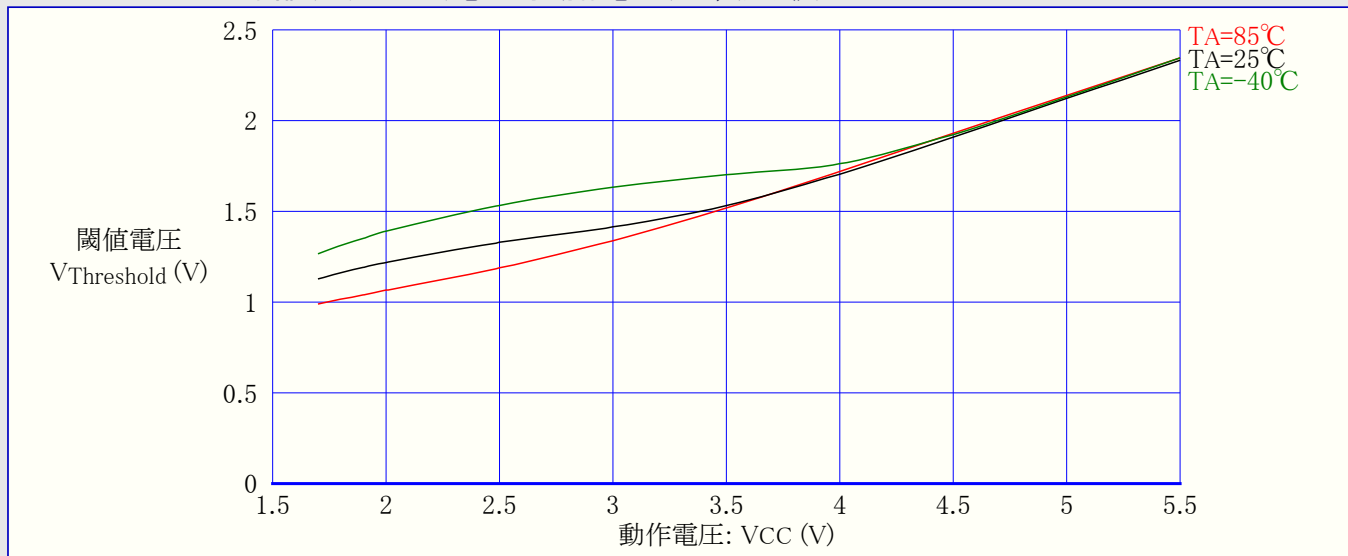


図145. RESETピン入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL,0}$ 読み値)

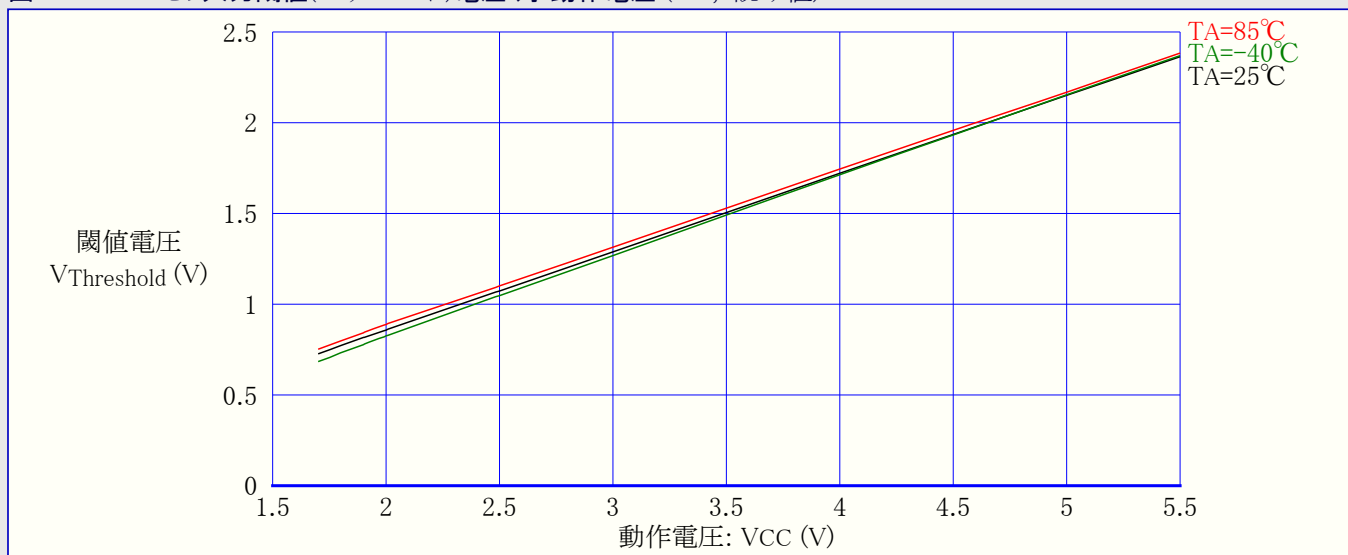
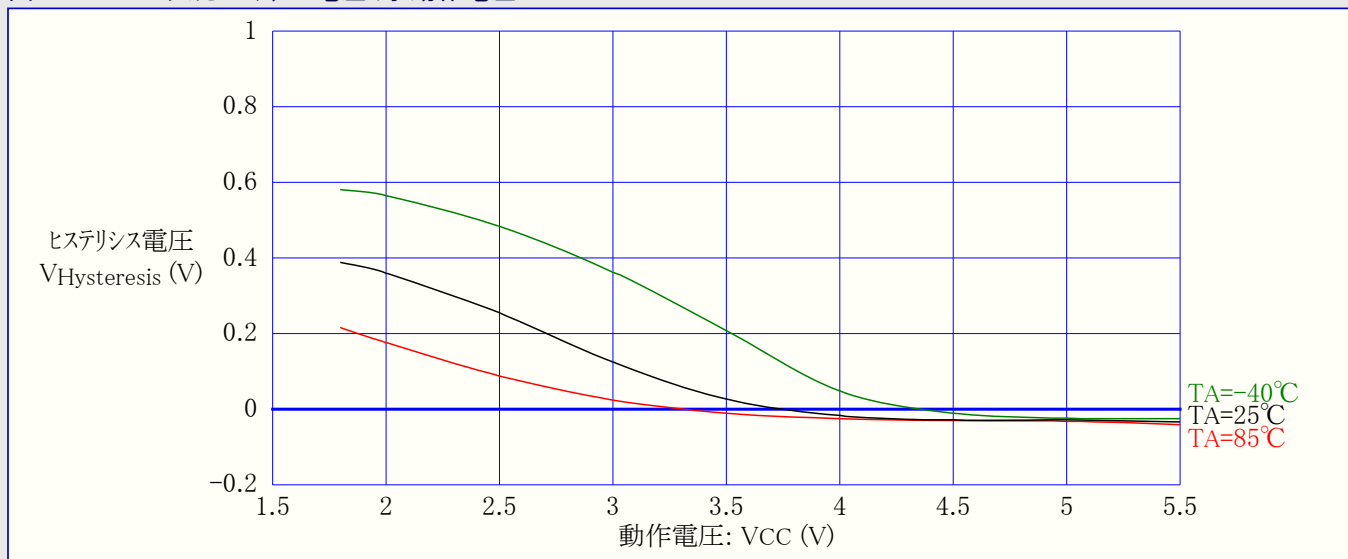


図146. RESET入力ヒステリシス電圧 対 動作電圧



## 25.8. 吐き出し電流能力

### 25.8.1. 入出力ピン

図147. I/Oピン出力電圧 対 吐き出し電流 (VCC=1.8V)

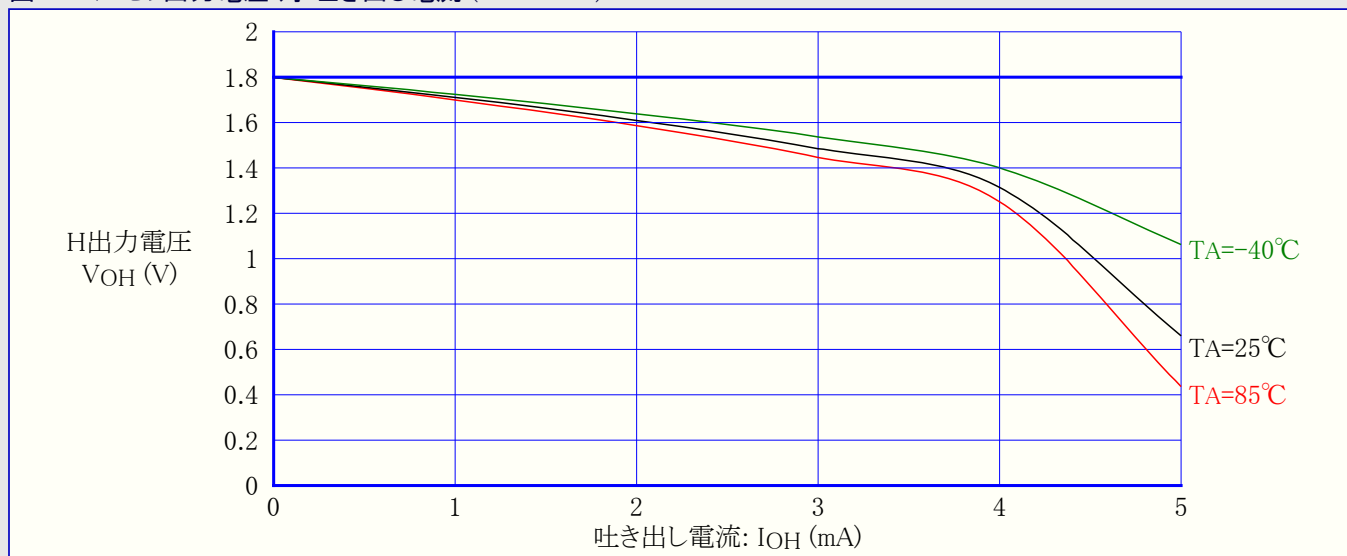


図148. I/Oピン出力電圧 対 吐き出し電流 (VCC=3V)

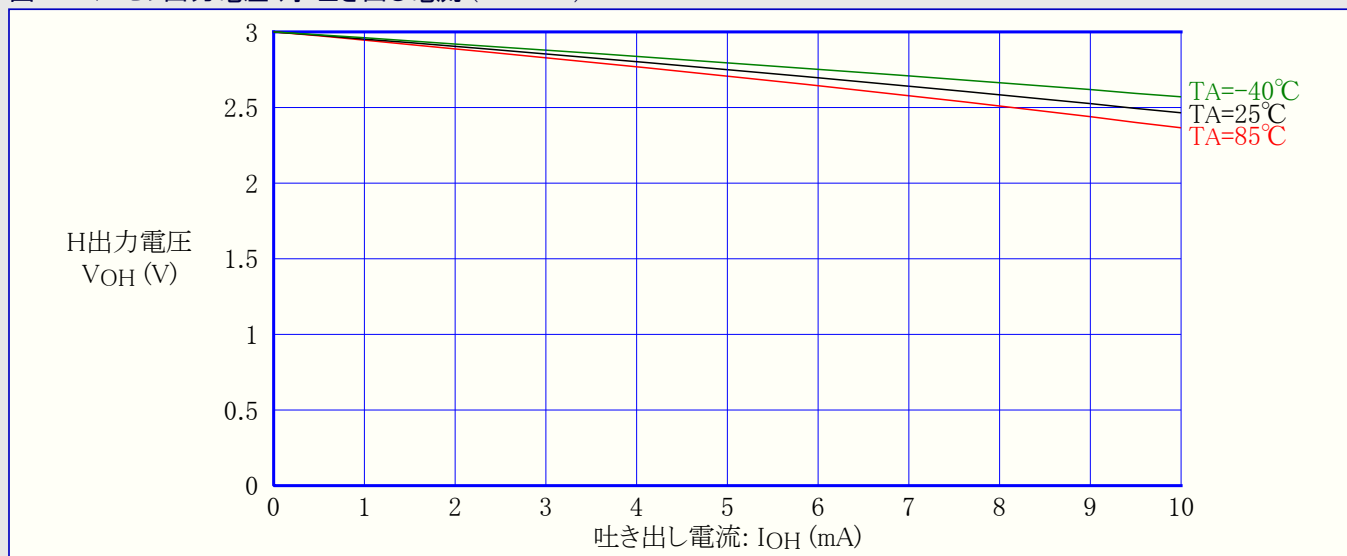
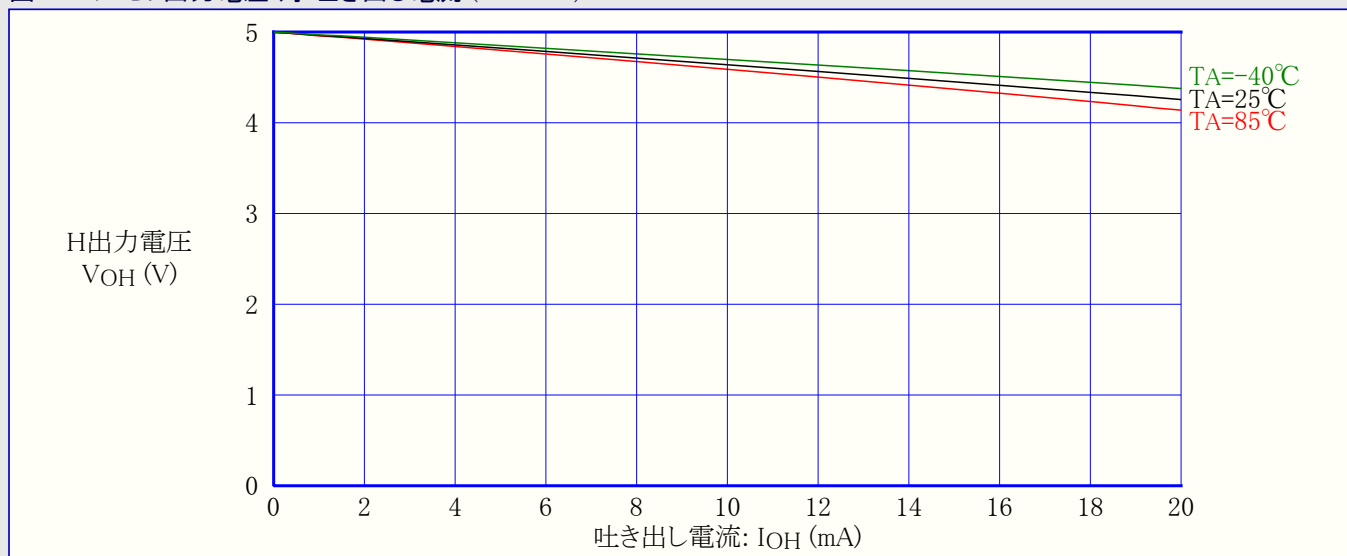


図149. I/Oピン出力電圧 対 吐き出し電流 (VCC=5V)



## 25.8.2. I/Oとしてのリセットピン

図150. 入出力としてのRESETピン出力電圧 対 吐き出し電流 (VCC=1.8V)

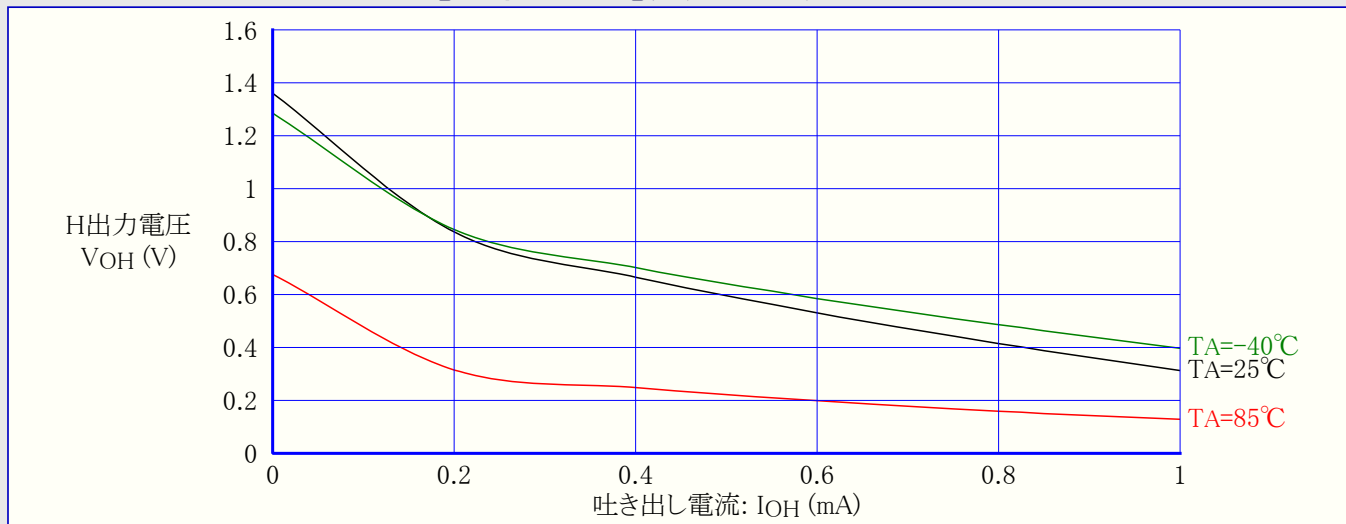


図151. 入出力としてのRESETピン出力電圧 対 吐き出し電流 (VCC=3V)

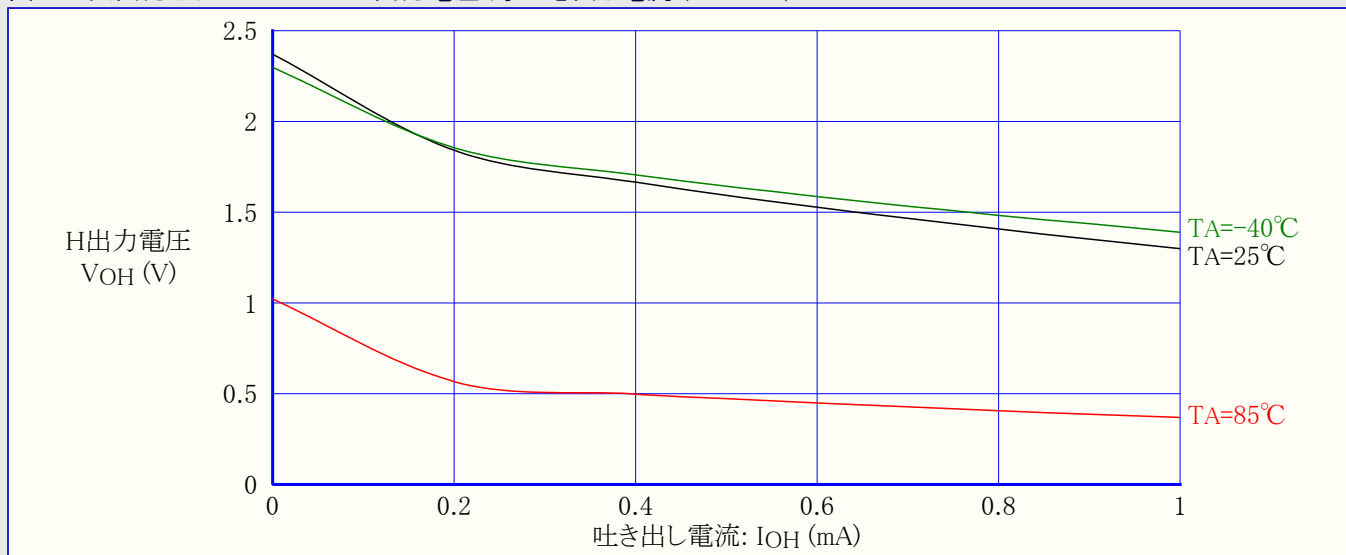
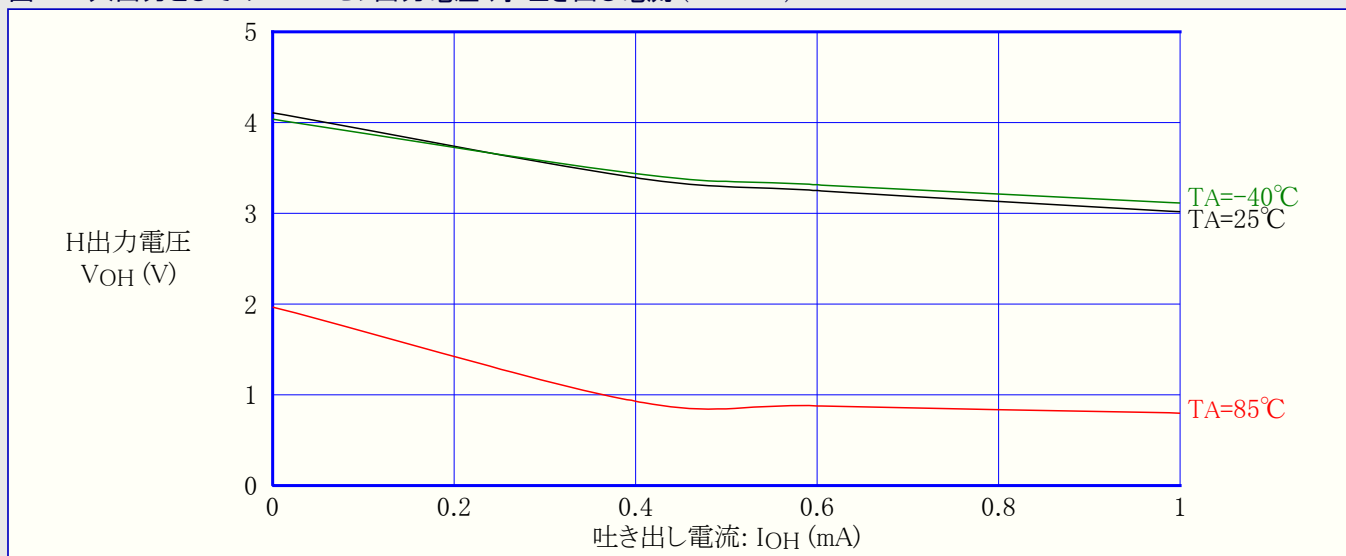


図152. 入出力としてのRESETピン出力電圧 対 吐き出し電流 (VCC=5V)



## 25.9. 吸い込み電流能力

### 25.9.1. 標準吸い込み能力の入出力ピン

図153. 標準I/Oピン出力電圧 対 吸い込み電流 (VCC=1.8V)

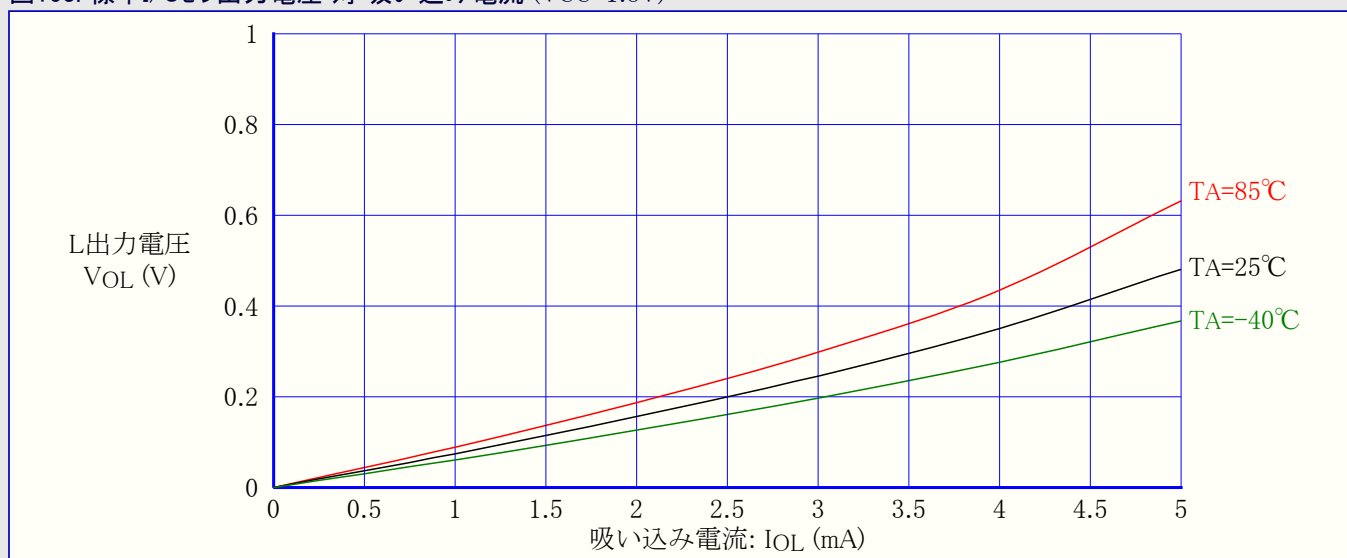


図154. 標準I/Oピン出力電圧 対 吸い込み電流 (VCC=3V)

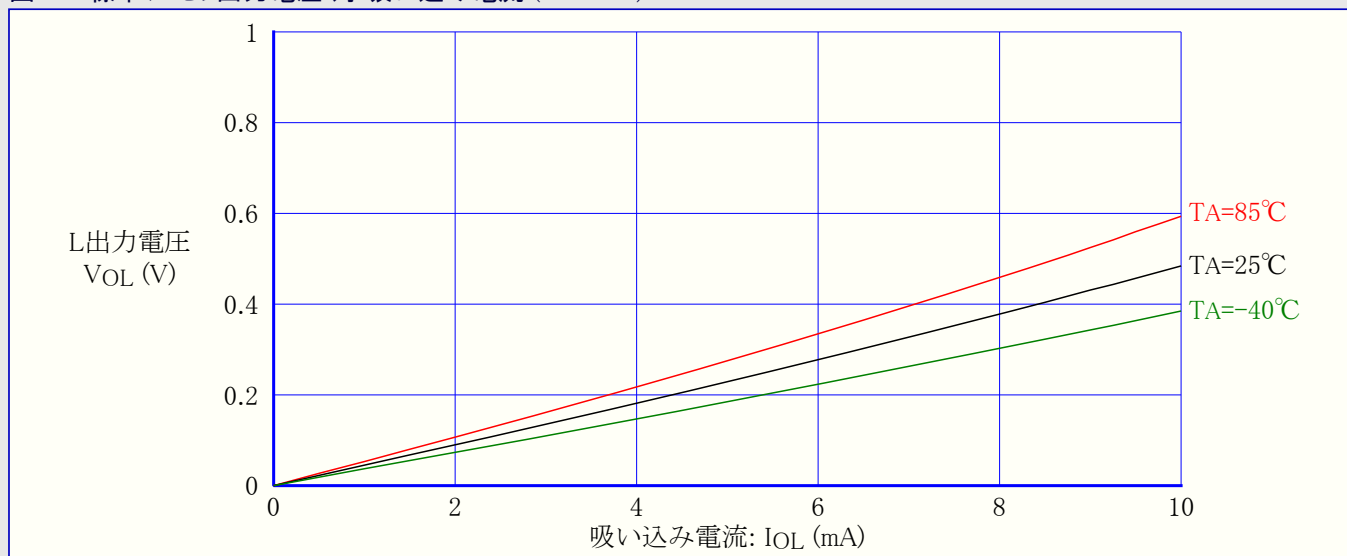
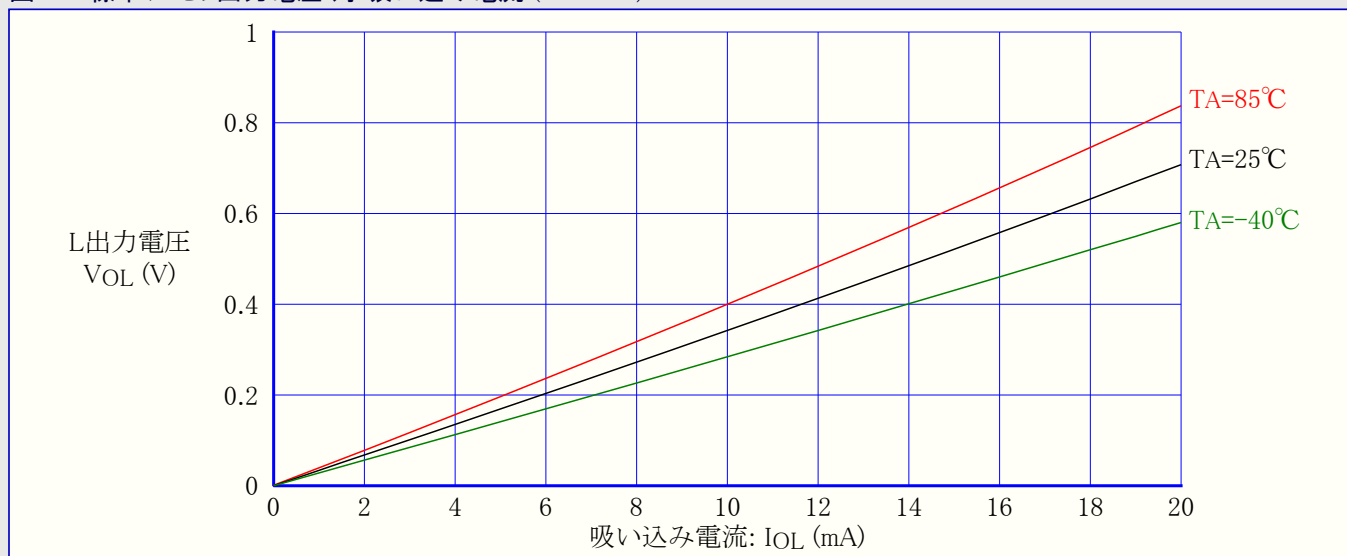


図155. 標準I/Oピン出力電圧 対 吸い込み電流 (VCC=5V)





## 25.9.2. 高吸い込み能力の入出力ピン

図156. 高シフト/OEピン出力電圧 対 吸い込み電流 (VCC=1.8V)

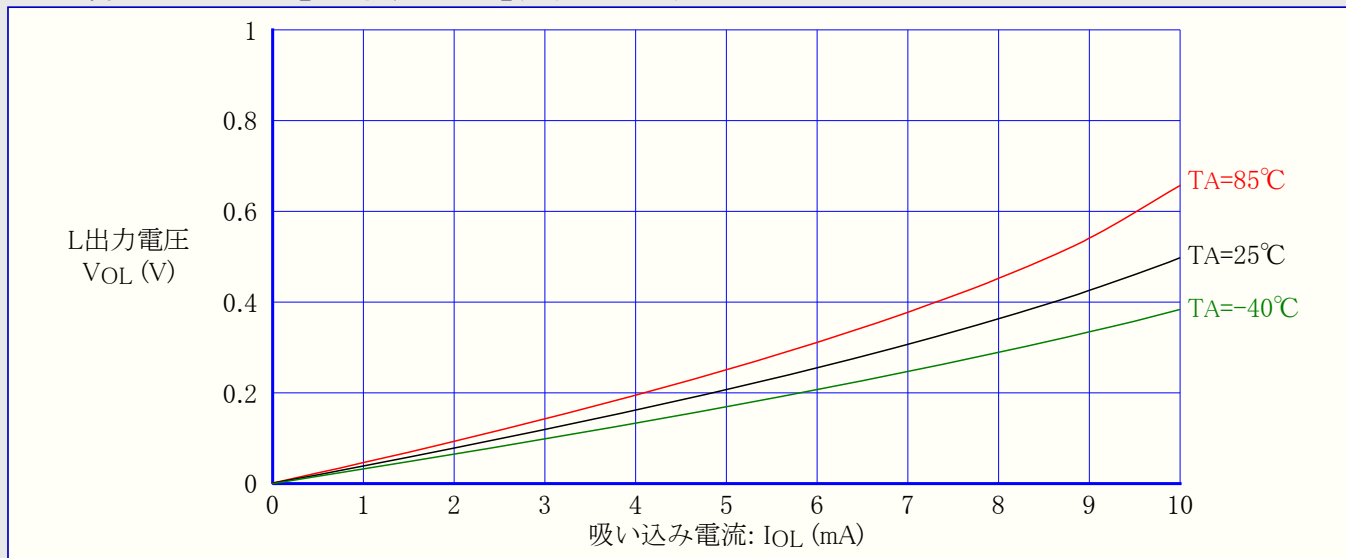


図157. 高シフト/OEピン出力電圧 対 吸い込み電流 (VCC=3V)

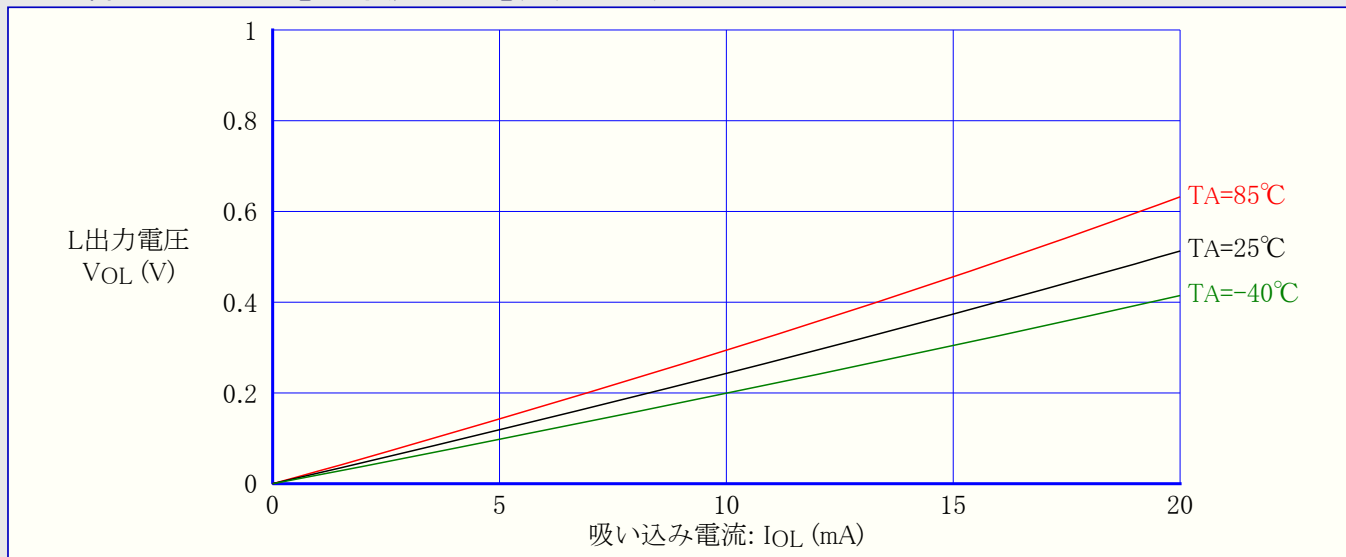
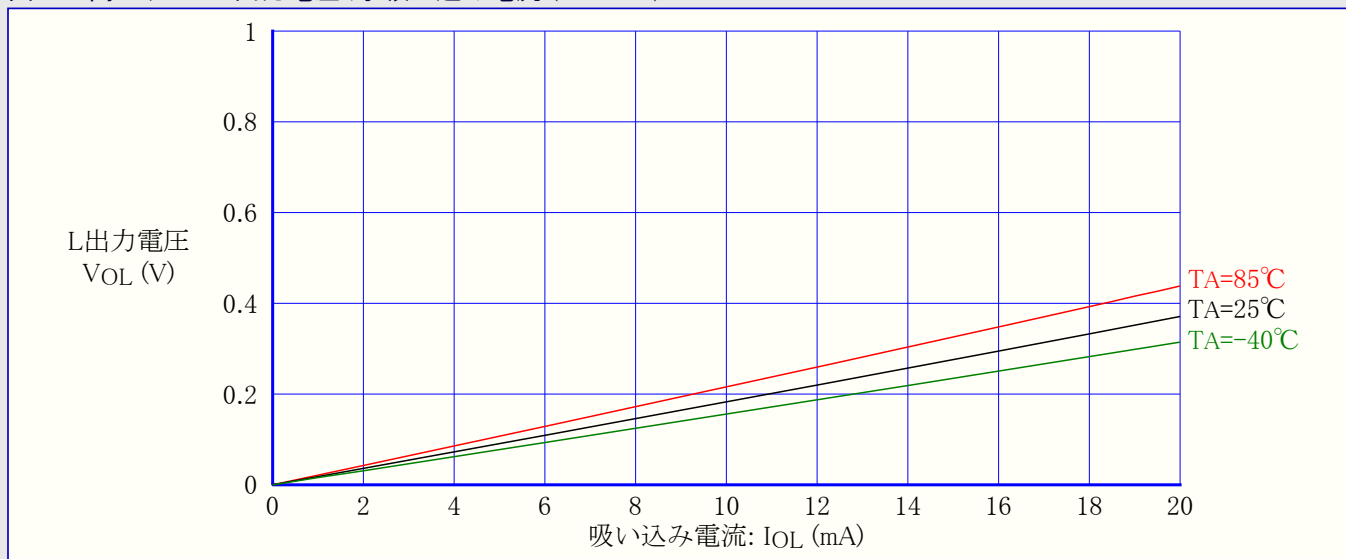


図158. 高シフト/OEピン出力電圧 対 吸い込み電流 (VCC=5V)



### 25.9.3. 追加高吸い込み能力の入出力ピン

図159. 追加高シク/オペン出力電圧 対 吸い込み電流 (VCC=1.8V)

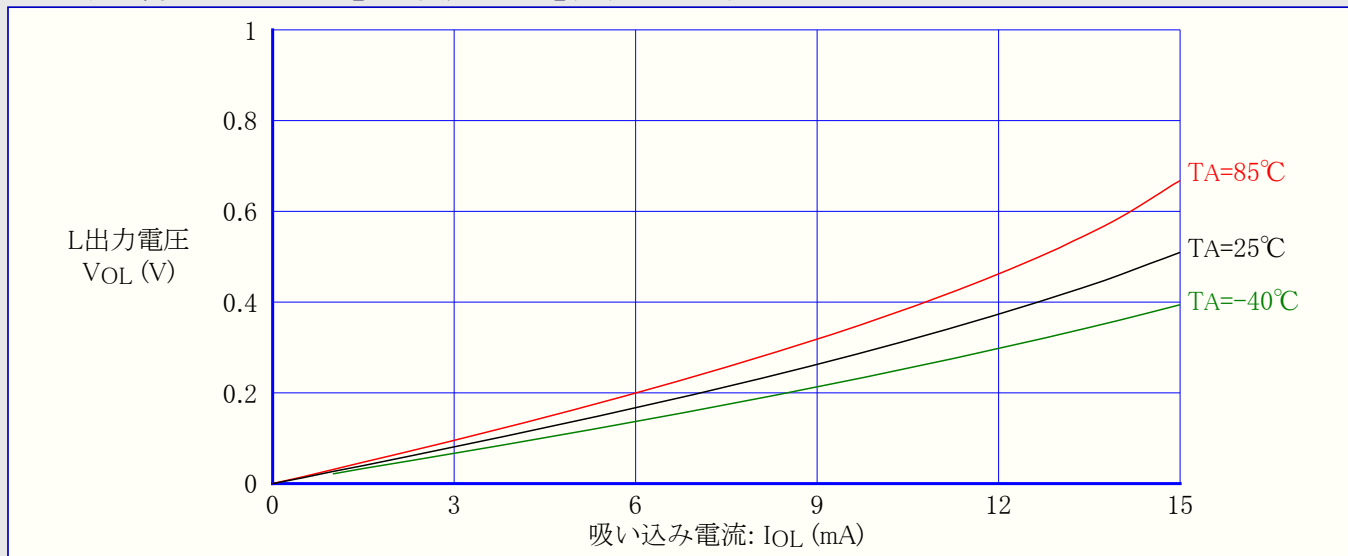


図160. 追加高シク/オペン出力電圧 対 吸い込み電流 (VCC=3V)

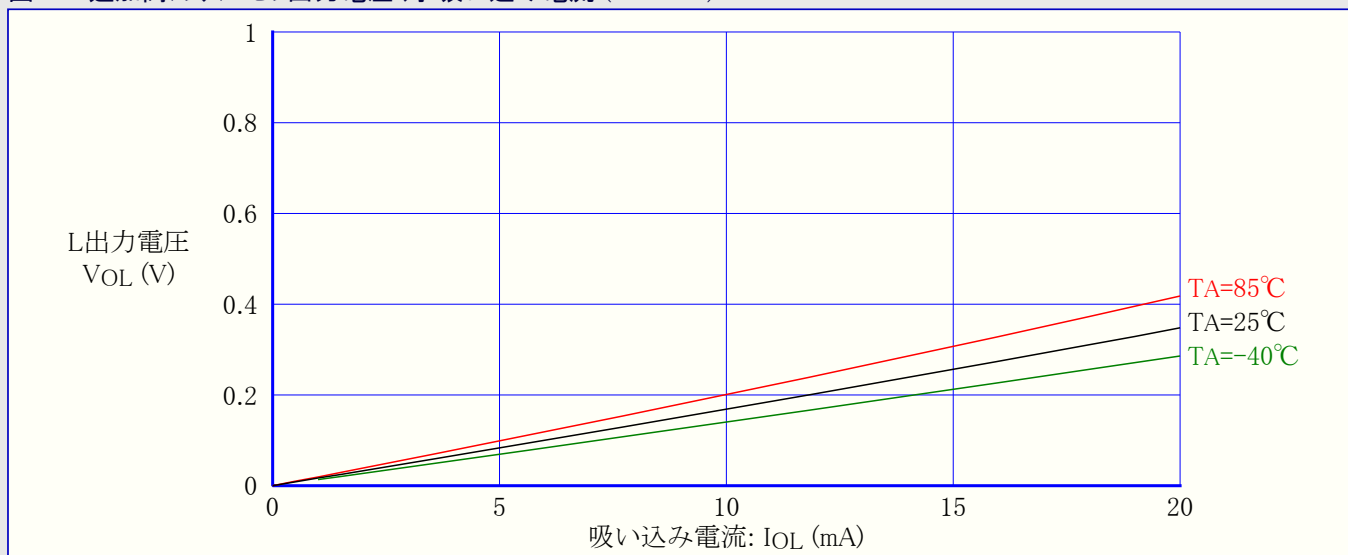
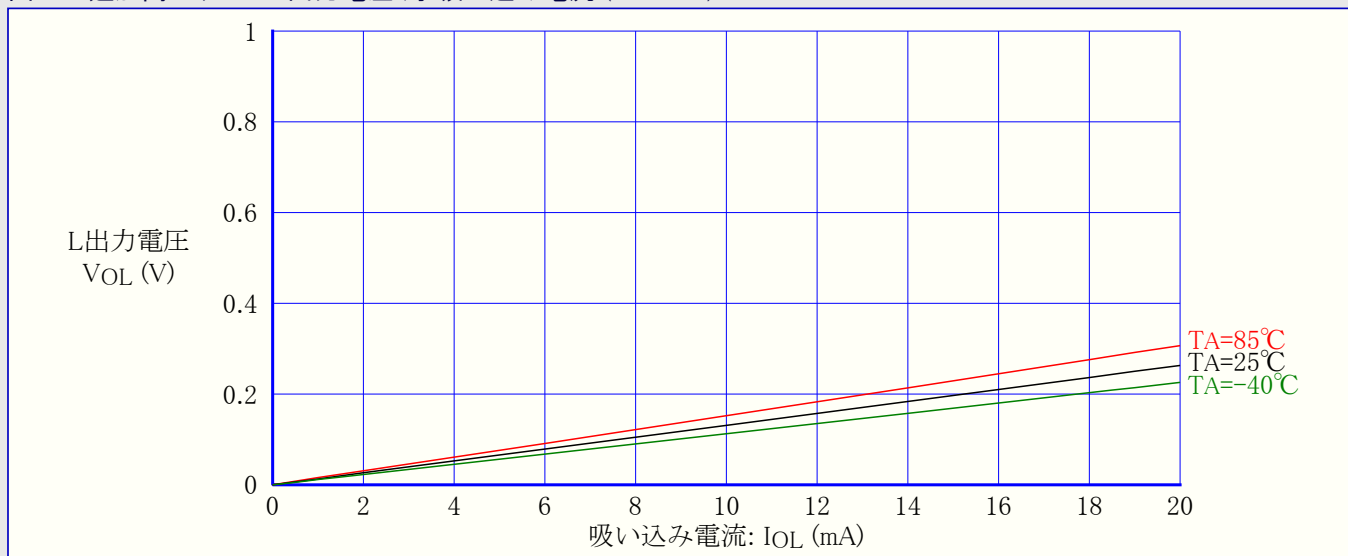


図161. 追加高シク/オペン出力電圧 対 吸い込み電流 (VCC=5V)



### 25.9.4. I/Oとしてのリセットピン

図162. 入出力としてのRESETピン出力電圧 対 吸い込み電流 (VCC=1.8V)

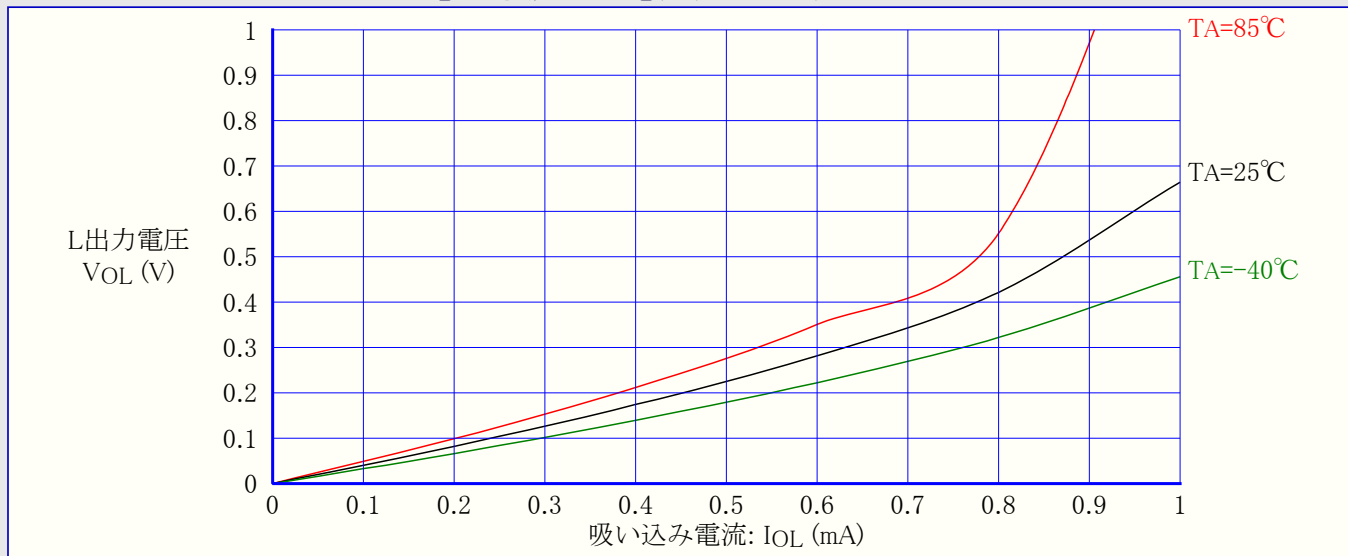


図163. 入出力としてのRESETピン出力電圧 対 吸い込み電流 (VCC=3V)

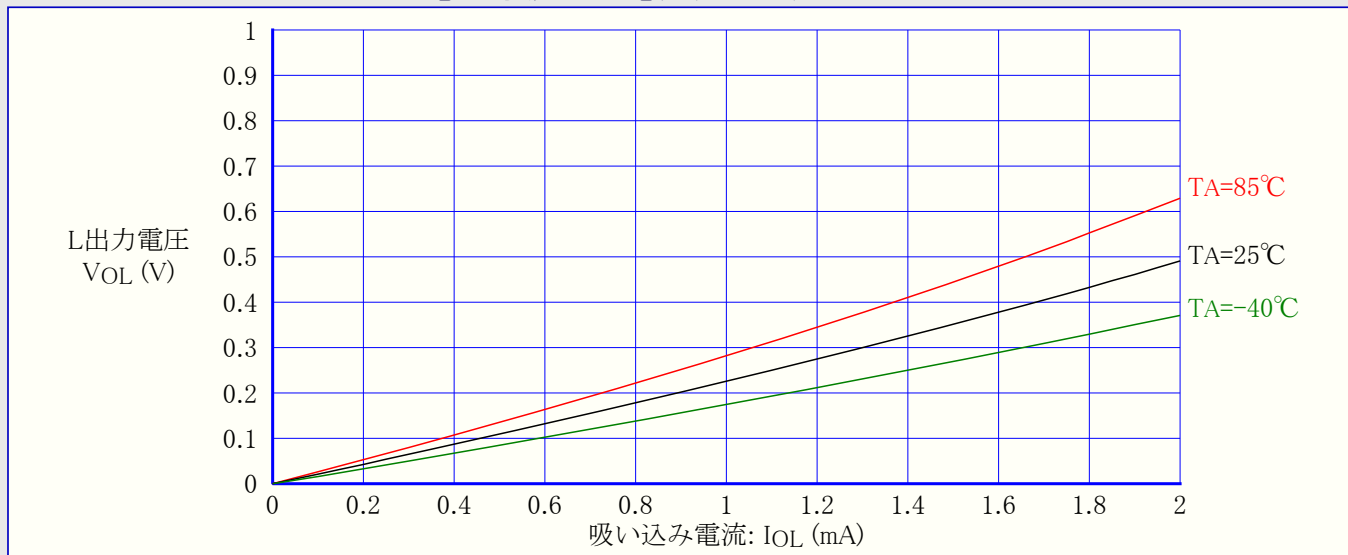
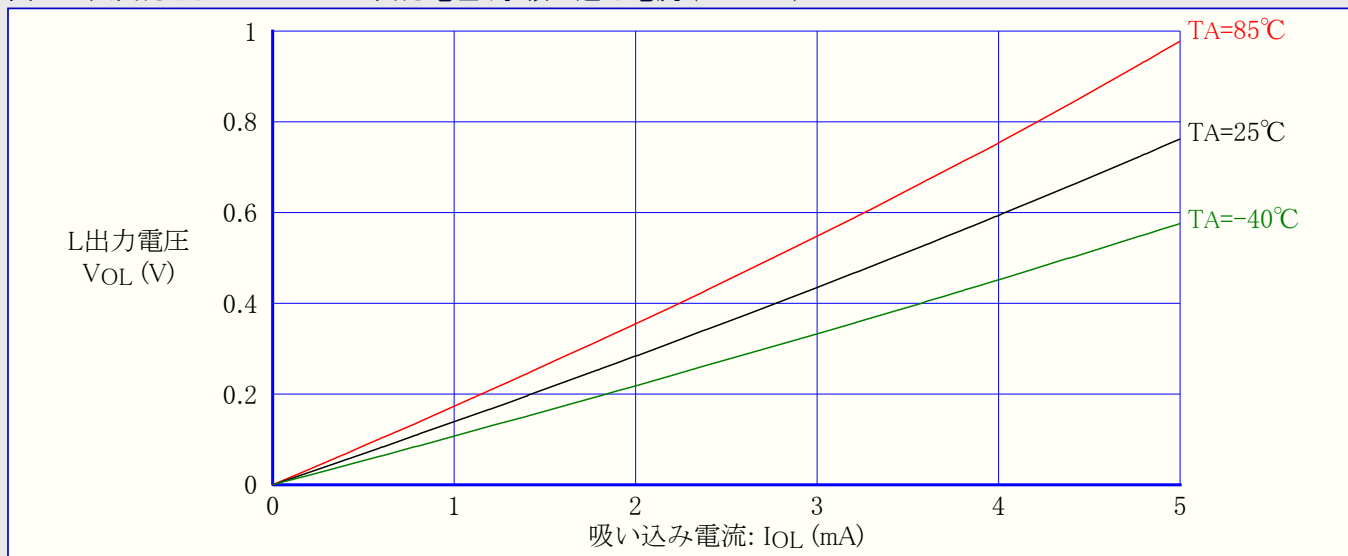


図164. 入出力としてのRESETピン出力電圧 対 吸い込み電流 (VCC=5V)



## 25.10. 低電圧検出器 (BOD)

図165. 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)

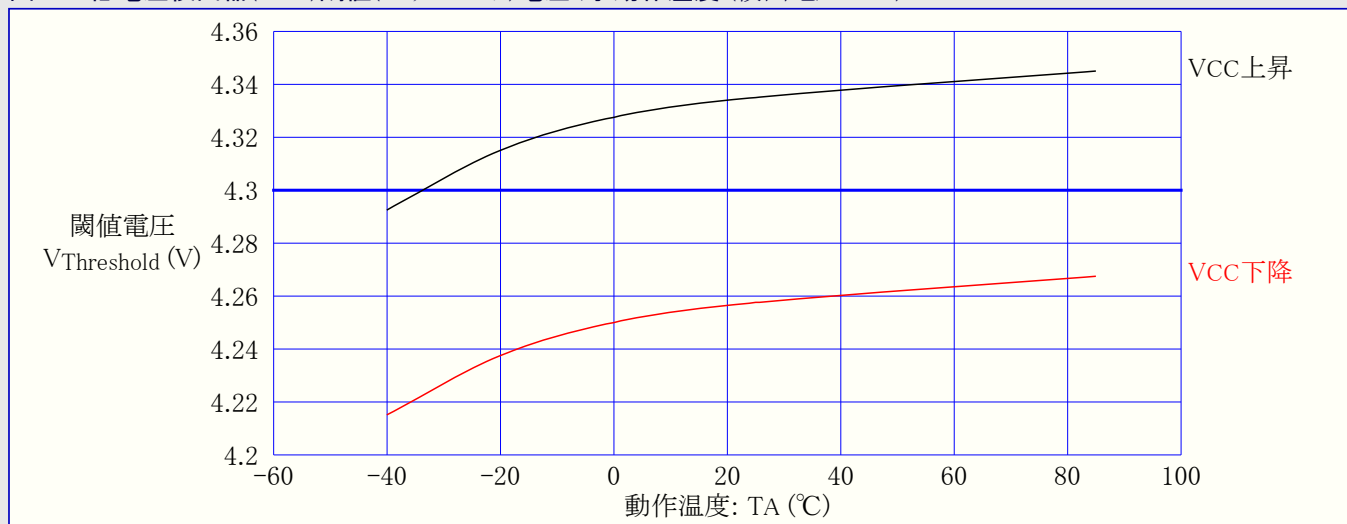


図166. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)

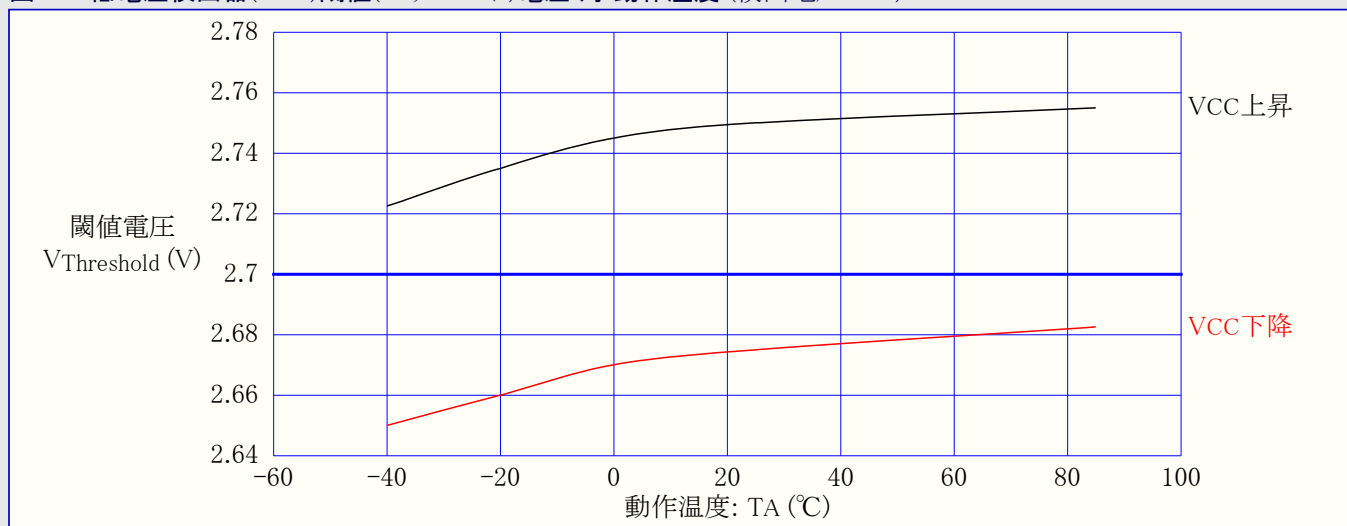


図167. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧1.8V)

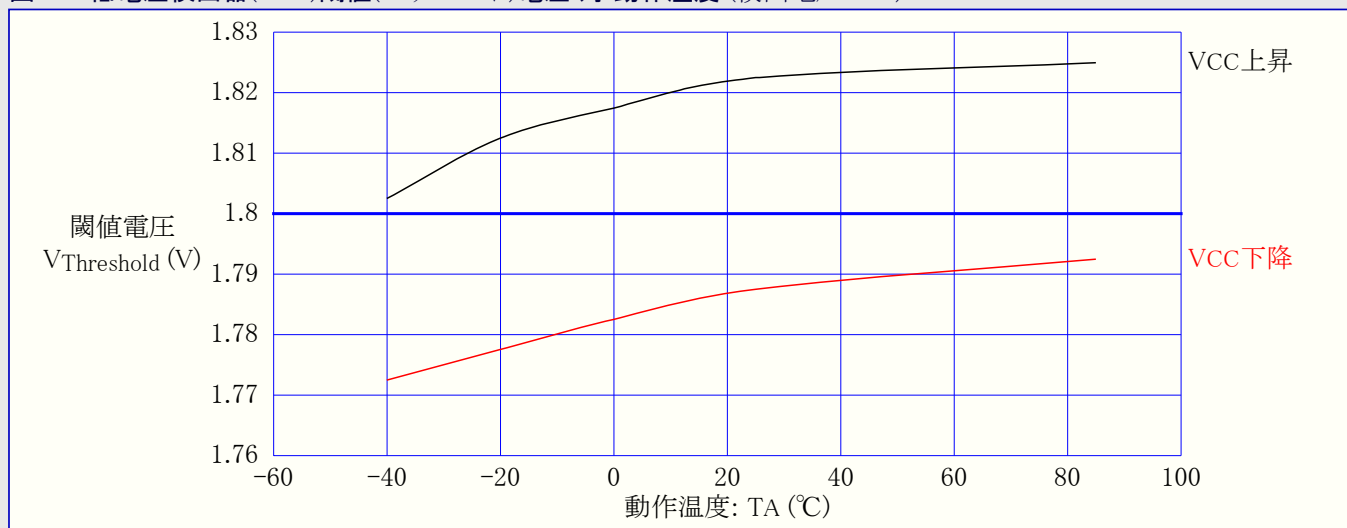


図168. 採取動作低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧4.3V)

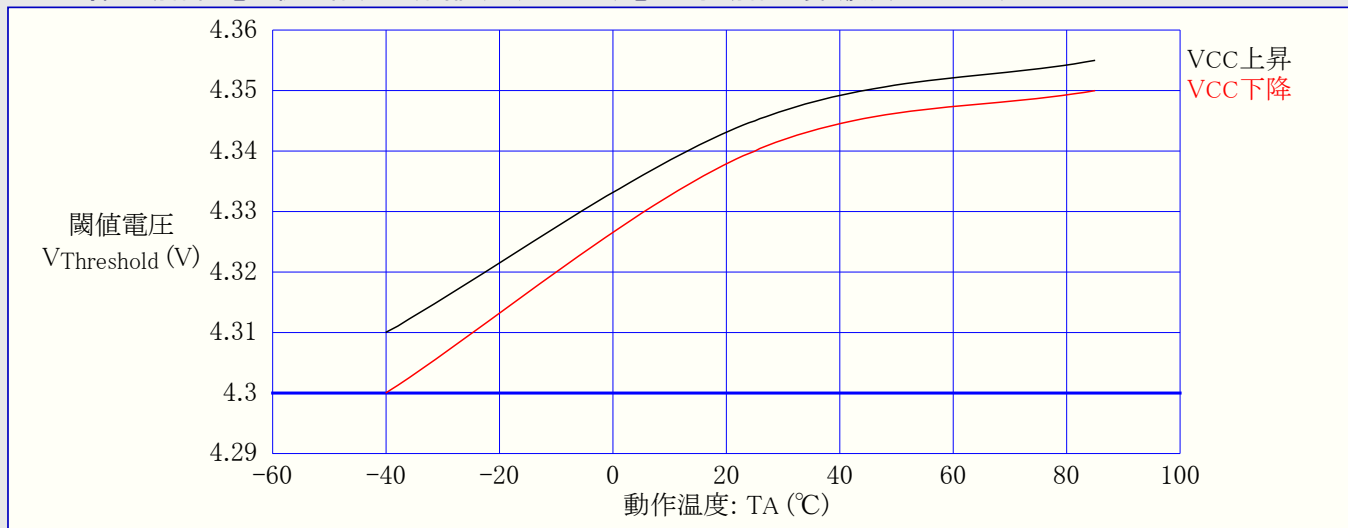


図169. 採取動作低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧2.7V)

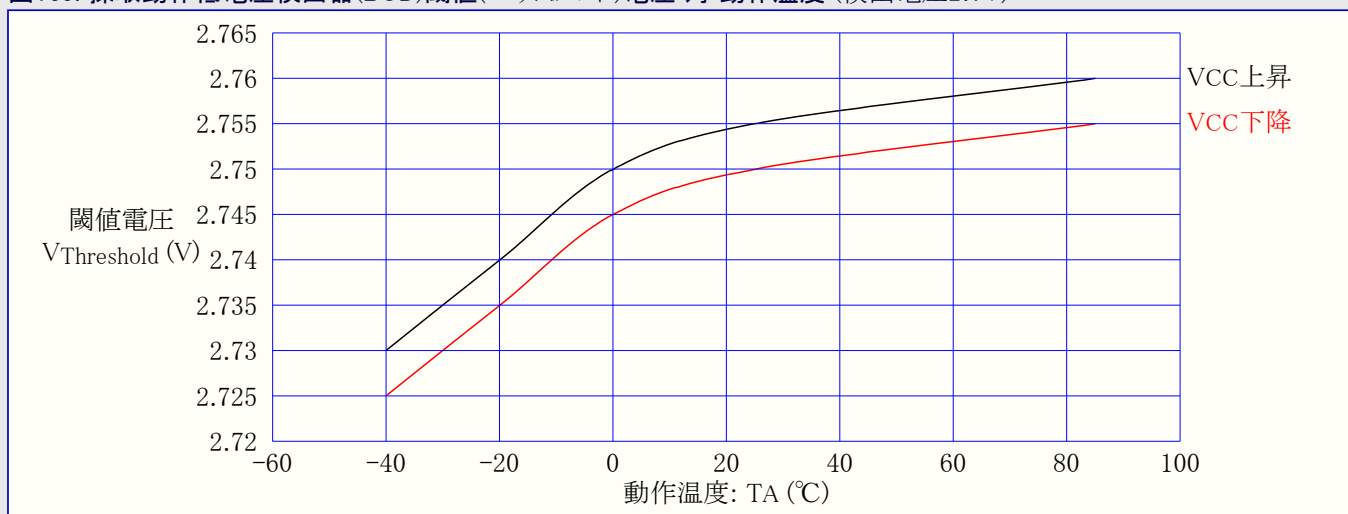
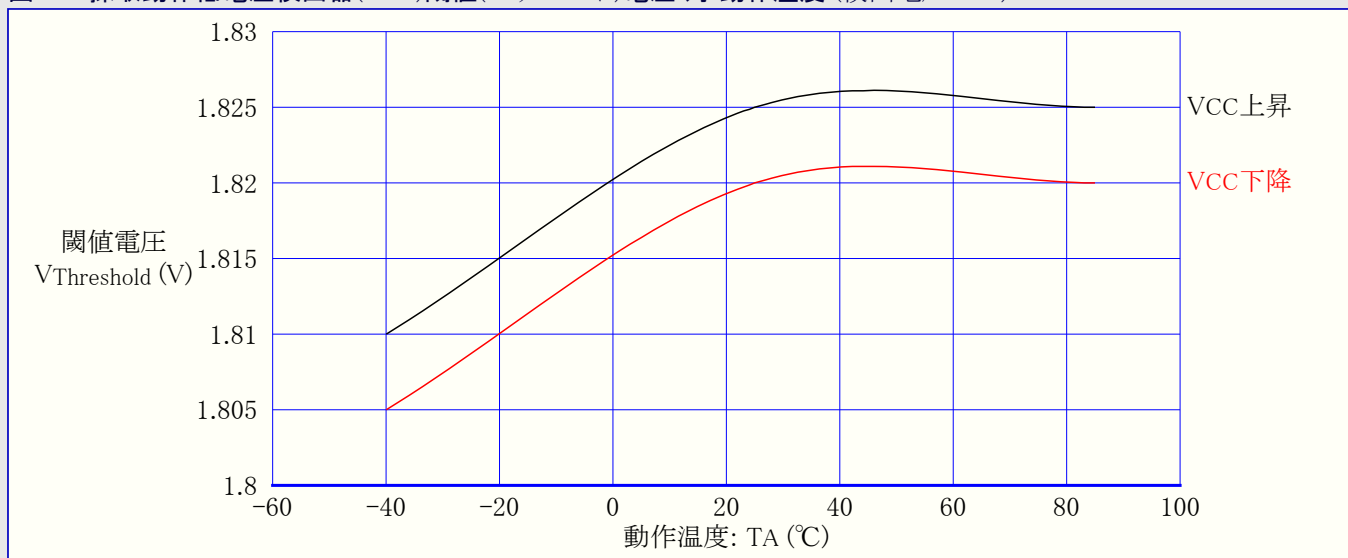


図170. 採取動作低電圧検出器(BOD)閾値(スレッショルド)電圧 対 動作温度 (検出電圧1.8V)





## 25.11. バントギャップ電圧

図171. 内部バントギャップ電圧 対 動作電圧

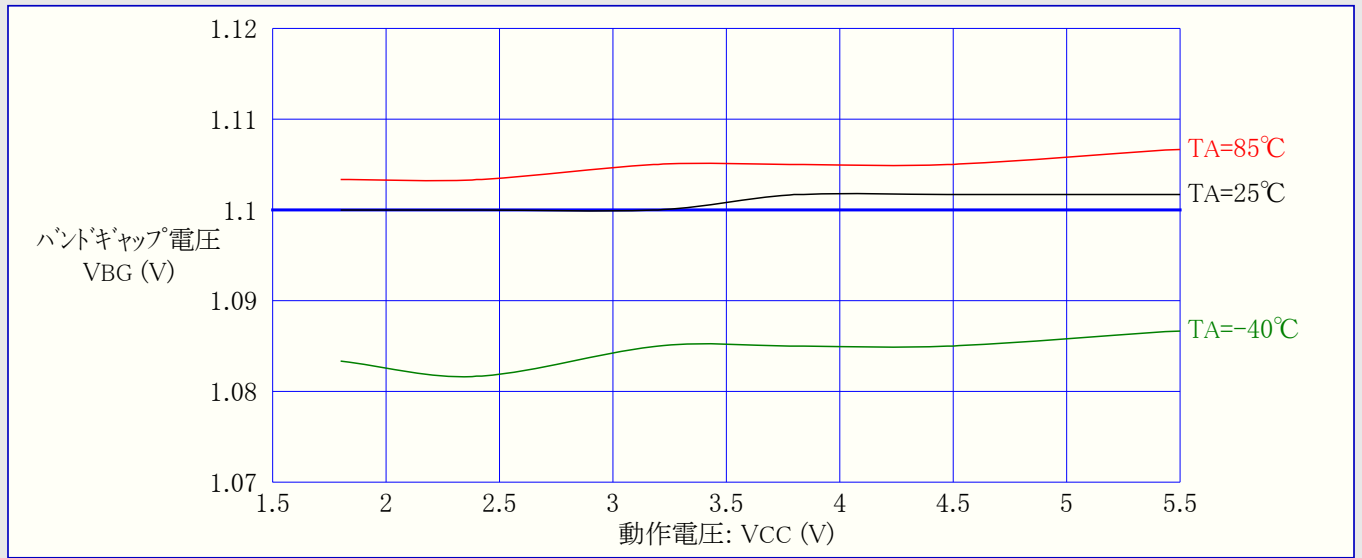
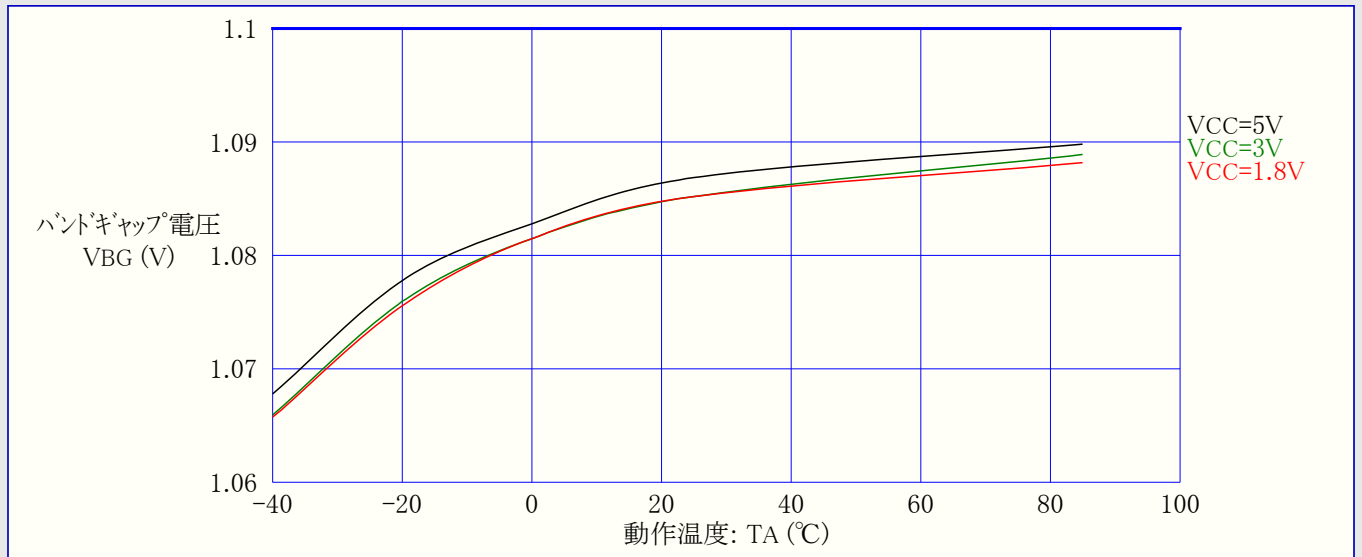


図172. 内部バントギャップ電圧 対 動作温度



## 25.12. リセット

図173. 電源ONリセット(POR)起動レベル 対 動作温度

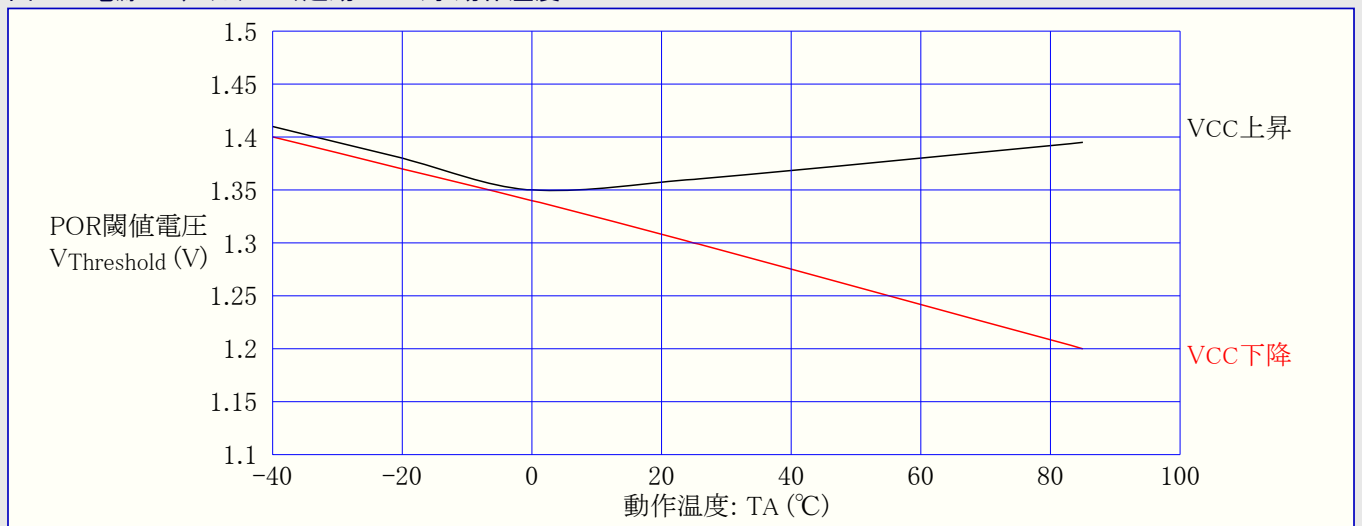
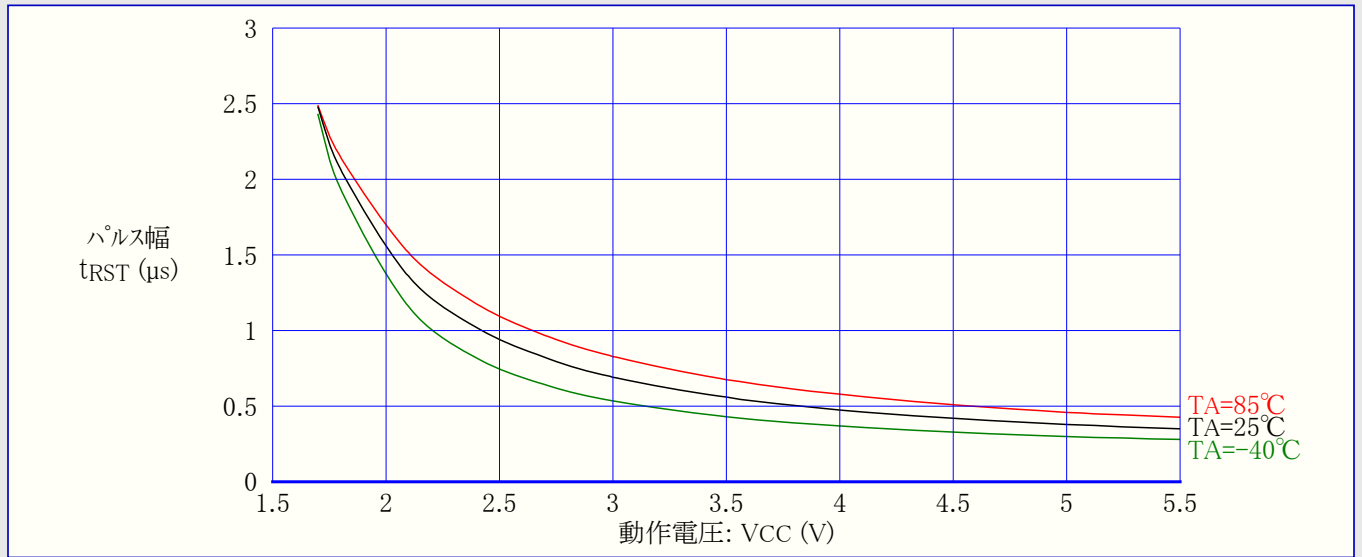


図174. 最小リセットパルス幅 対 動作電圧



### 25.13. アナログ比較器オフセット

図175. アナログ比較器オフセット電圧 対 入力電圧 (VCC=5V)

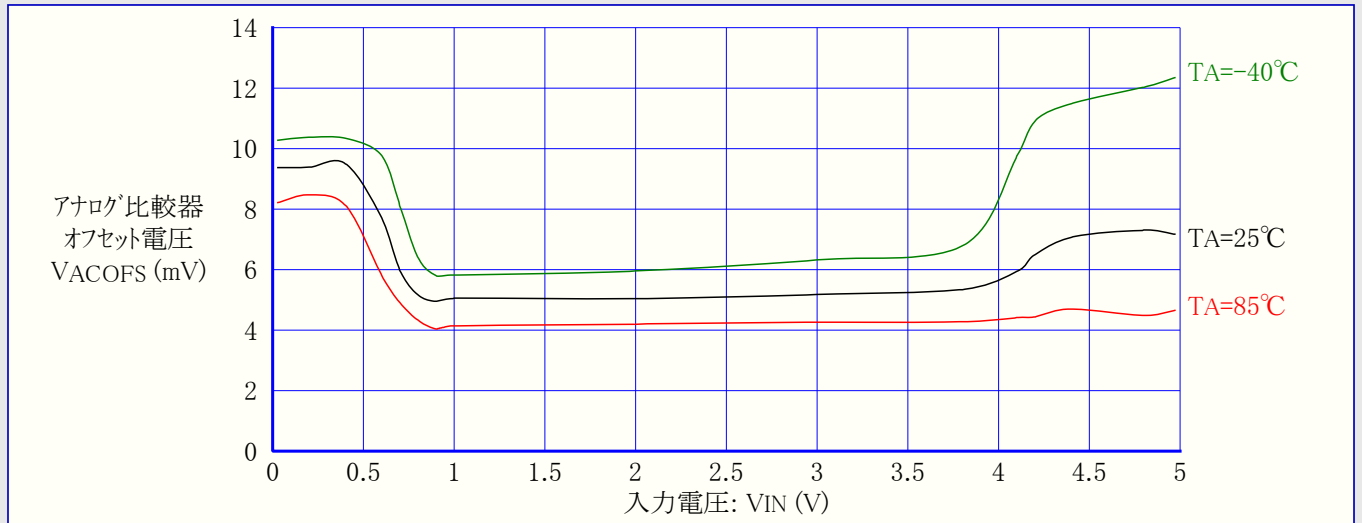


図176. アナログ比較器オフセット電圧 対 動作電圧 (VIN=1.1V)

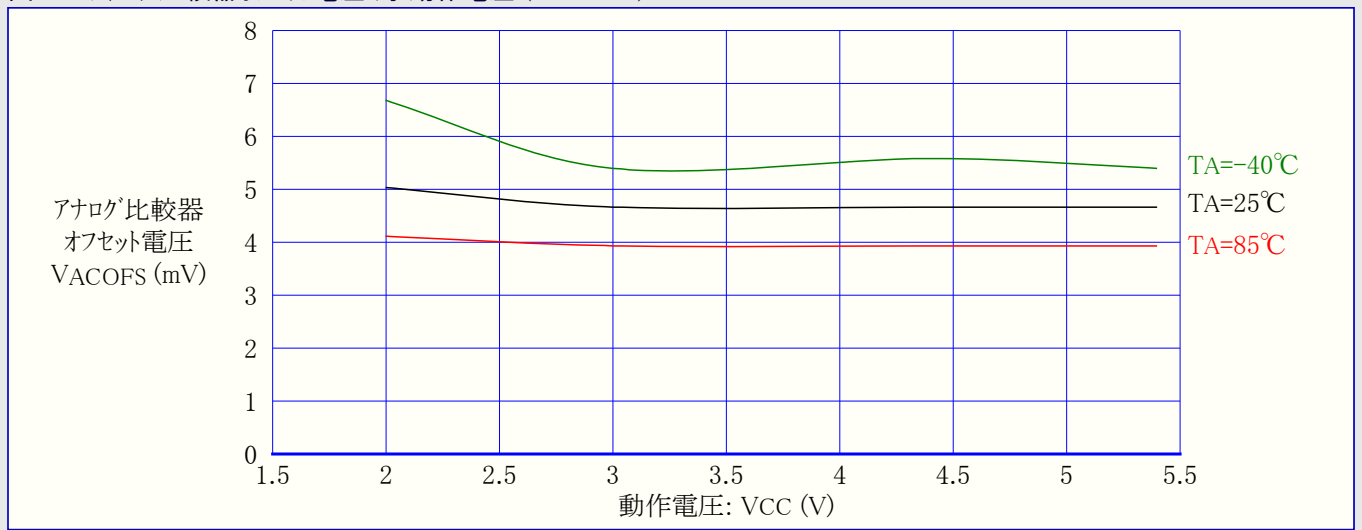
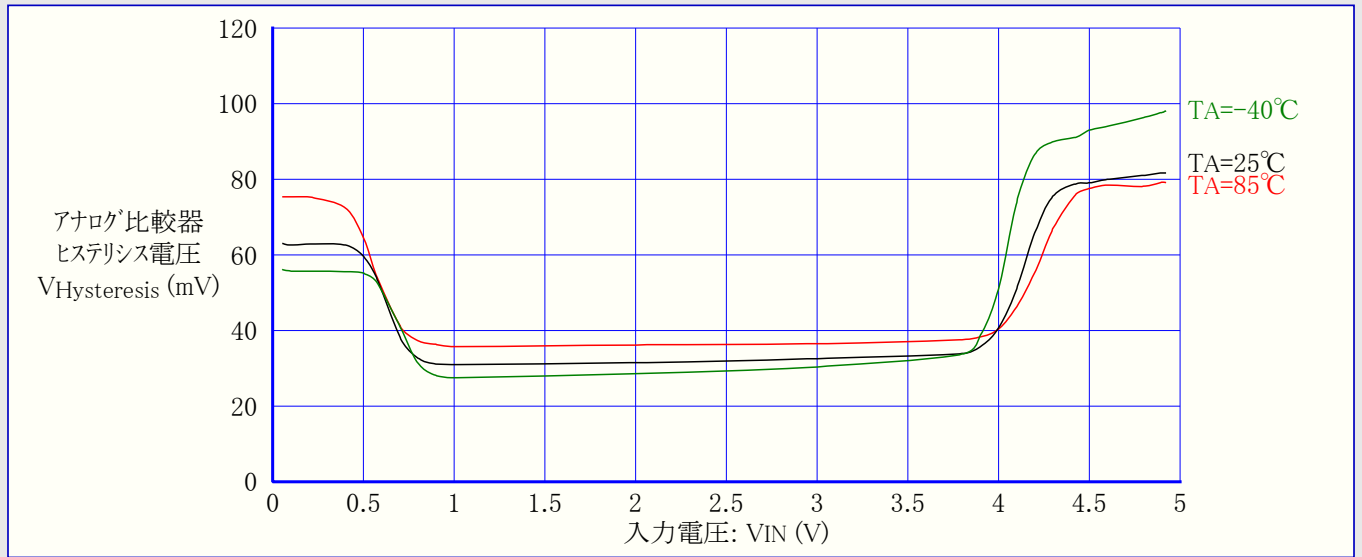


図177. アナログ比較器ヒステリシス電圧 対 入力電圧 (VCC=5V)



## 25.14. 内部発振器周波数

### 25.14.1. CKDIV8許可8MHz発振器

図178. CKDIV8許可8MHz校正付き発振器周波数 対 動作電圧 (1点校正)

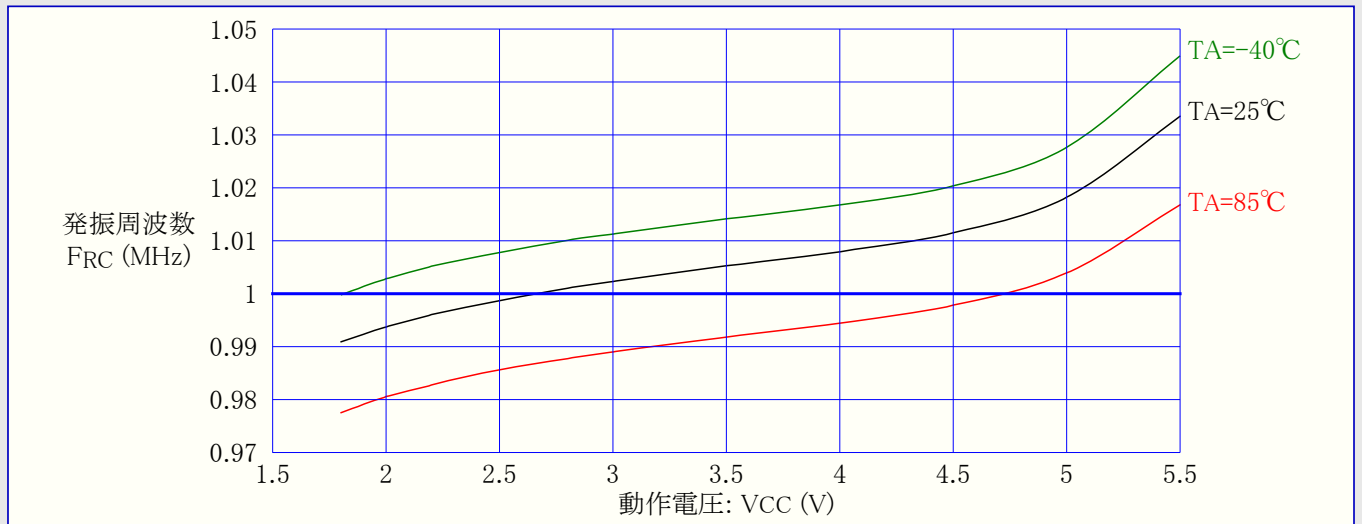


図179. CKDIV8許可8MHz校正付き発振器周波数 対 動作電圧 (2点校正)

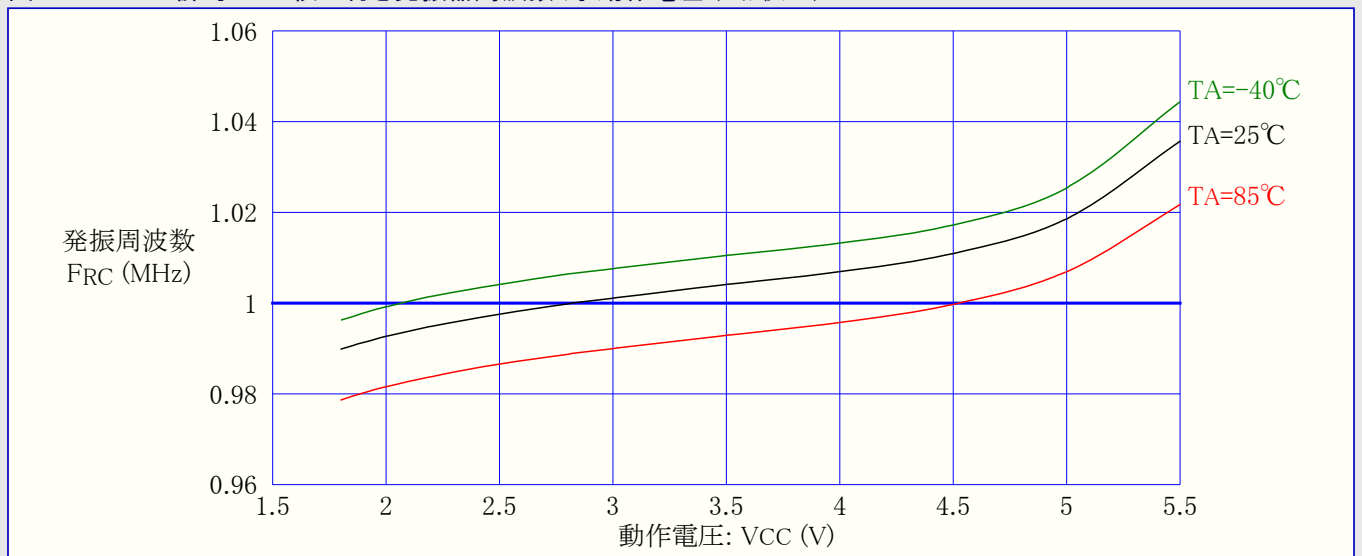


図180. CKDIV8許可8MHz校正付き発振器周波数 対 動作温度 (1点校正)

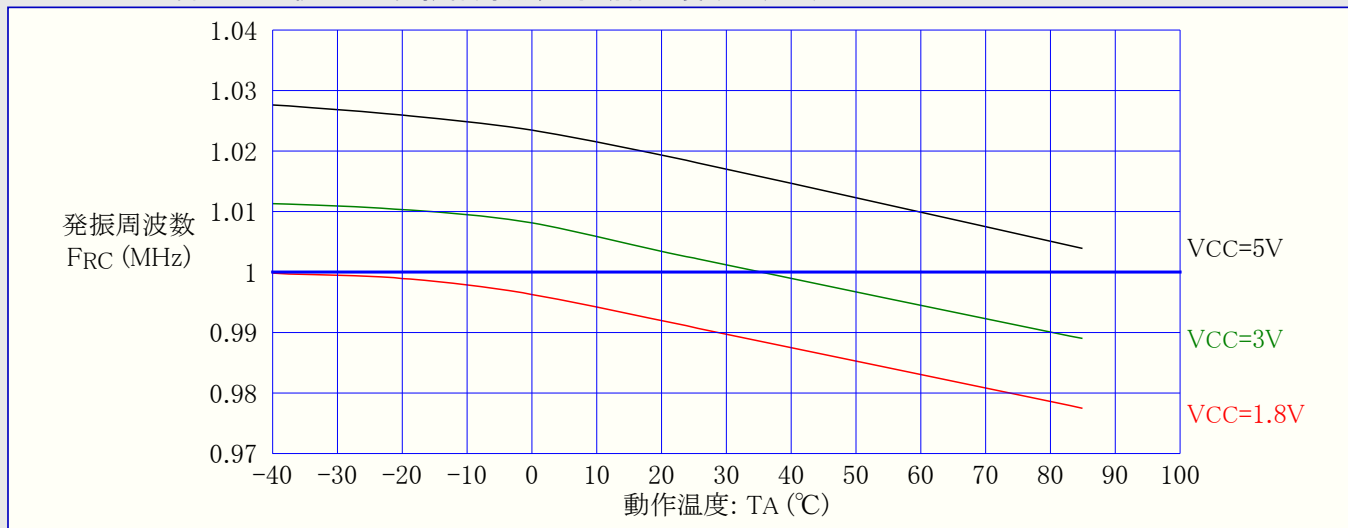


図181. CKDIV8許可8MHz校正付き発振器周波数 対 動作温度 (2点校正)

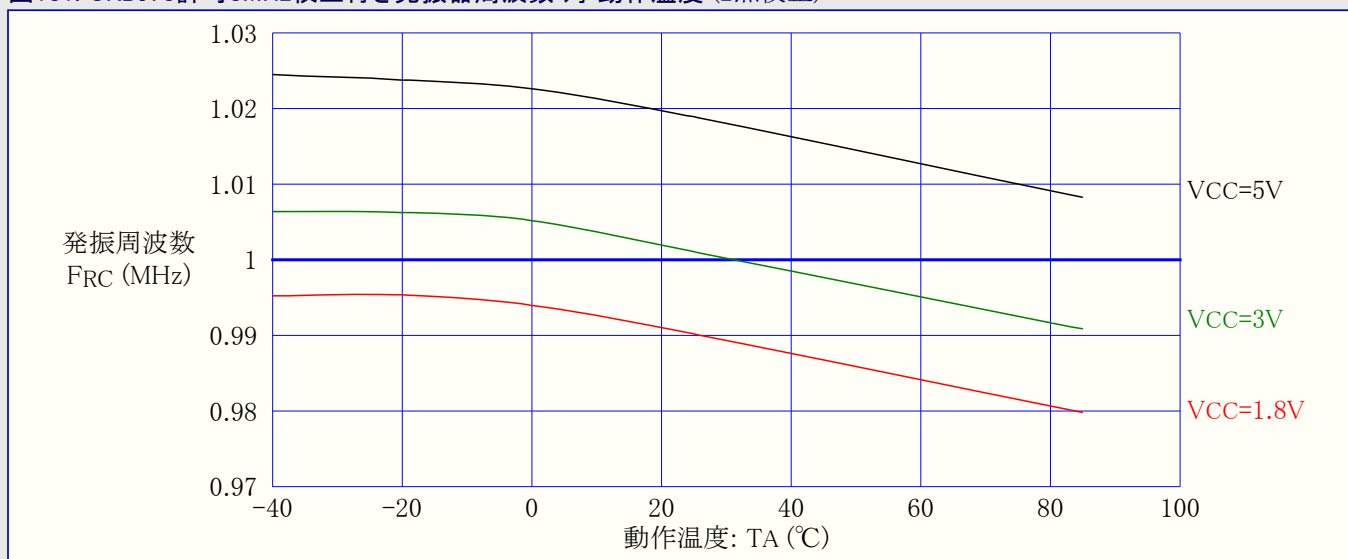
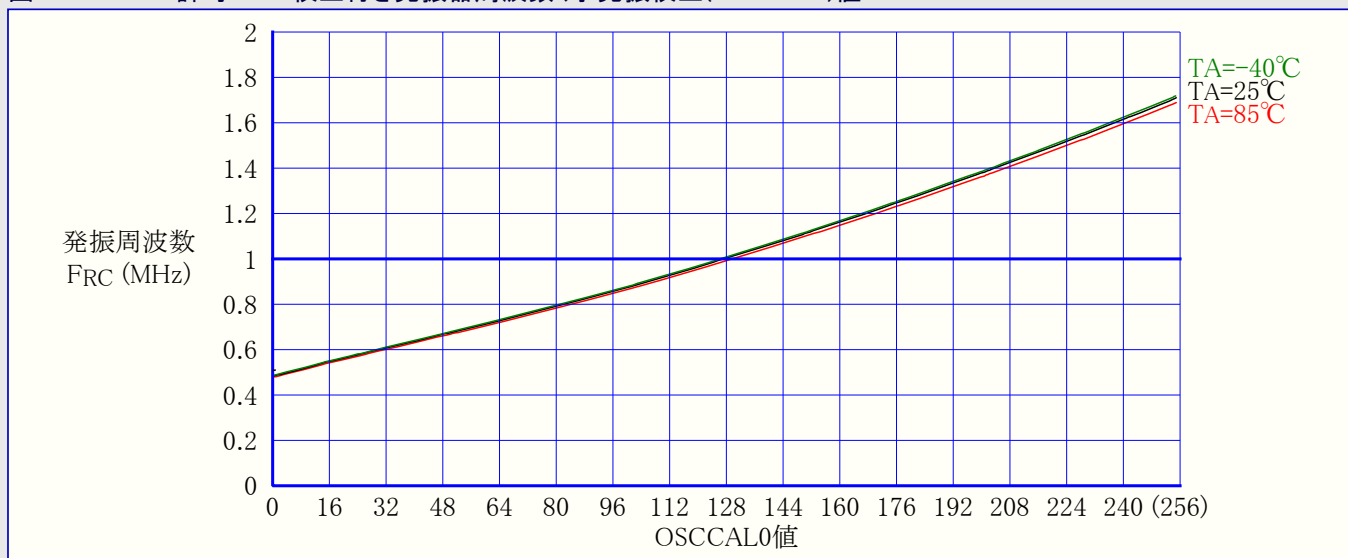


図182. CKDIV8許可8MHz校正付き発振器周波数 対 発振校正(OSCCAL0)値



### 25.14.2. 32kHz超低電力(ULP)発振器

図183. 32kHz超低電力(ULP)発振器周波数 対 動作電圧

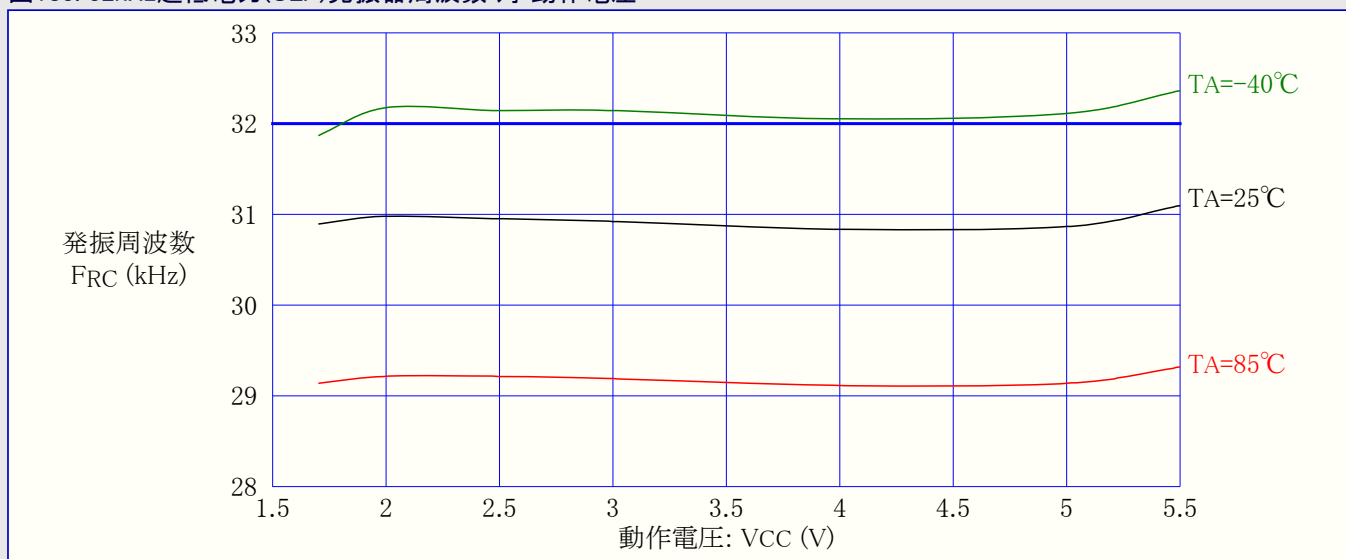


図184. 32kHz超低電力(ULP)発振器周波数 対 動作温度

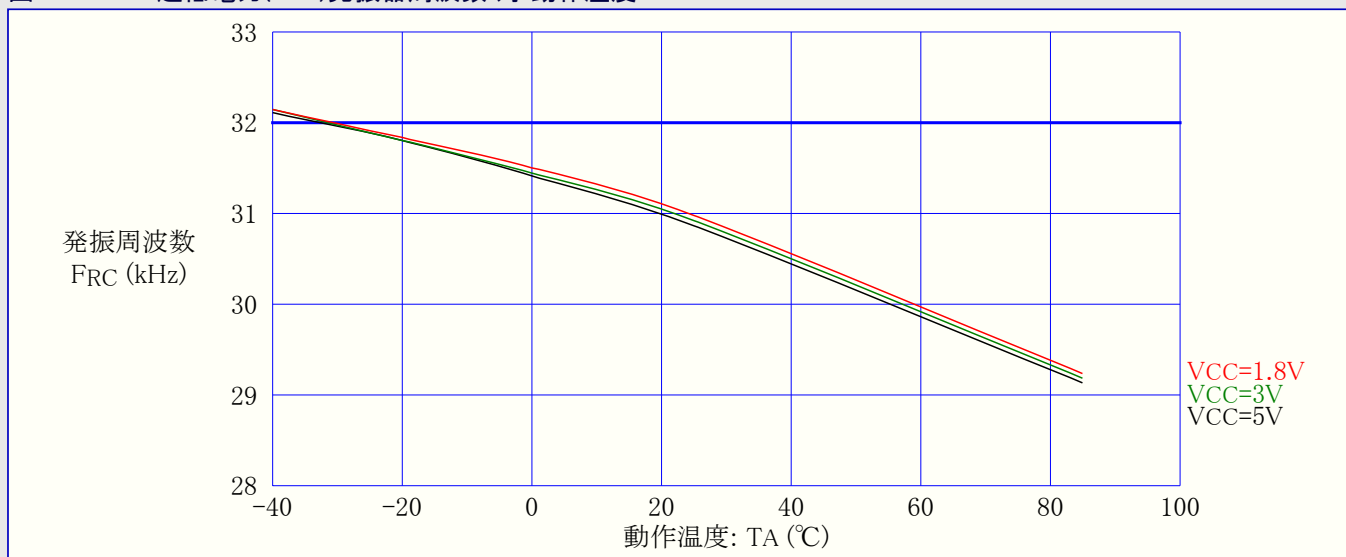
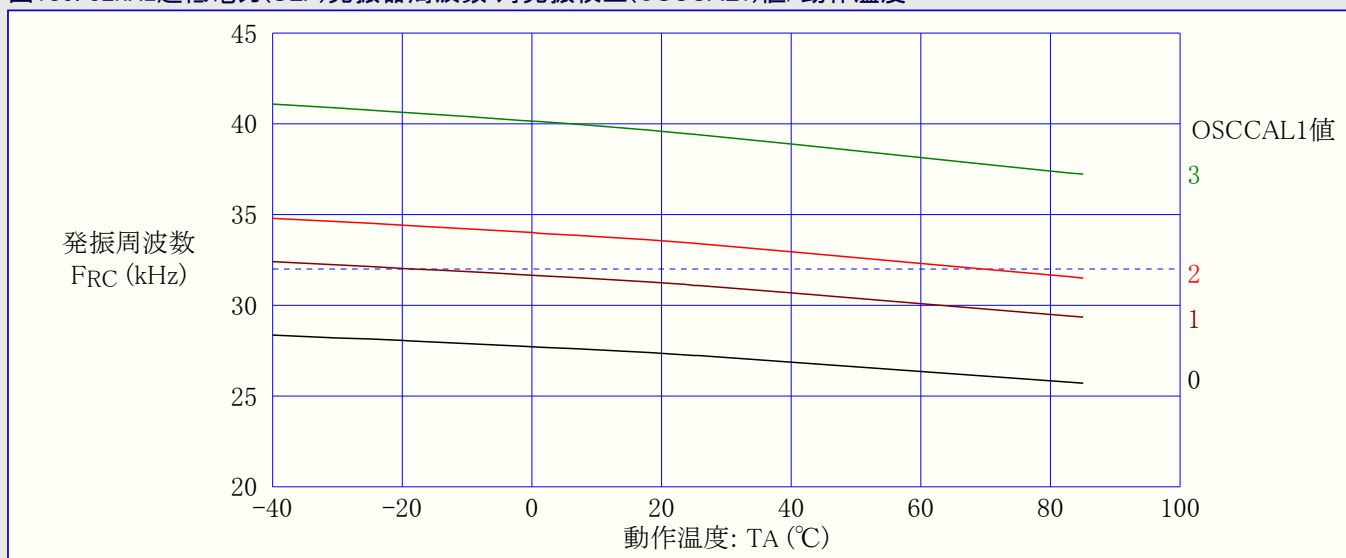


図185. 32kHz超低電力(ULP)発振器周波数 対 発振校正(OSCCAL1)値/動作温度





## 26. レジスタ要約

拡張1/Oレジスタ領域 (1/3)

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									
(\$FE)	予約									
(\$FD)	予約									
(\$FC)	予約									
(\$FB)	予約									
(\$FA)	予約									
(\$F9)	予約									
(\$F8)	予約									
(\$F7)	予約									
(\$F6)	予約									
(\$F5)	予約									
(\$F4)	予約									
(\$F3)	予約									
(\$F2)	予約									
(\$F1)	OSCTCAL0B	発振器温度校正レジスタB値								22
(\$F0)	OSCTCAL0A	発振器温度校正レジスタA値								22
(\$EF)	予約									
(\$EE)	予約									
(\$ED)	予約									
(\$EC)	予約									
(\$EB)	予約									
(\$EA)	予約									
(\$E9)	TOCPMSA1	TOCC7S1	TOCC7S0	TOCC6S1	TOCC6S0	TOCC5S1	TOCC5S0	TOCC4S1	TOCC4S0	85
(\$E8)	TOCPMSA0	TOCC3S1	TOCC3S0	TOCC2S1	TOCC2S0	TOCC1S1	TOCC1S0	TOCC0S1	TOCC0S0	85
(\$E7)	予約									
(\$E6)	予約									
(\$E5)	予約									
(\$E4)	予約									
(\$E3)	予約									
(\$E2)	TOCPMCOE	TOCC7OE	TOCC6OE	TOCC5OE	TOCC4OE	TOCC3OE	TOCC2OE	TOCC1OE	TOCC0OE	86
(\$E1)	予約									
(\$E0)	予約									
(\$DF)	DIDR3	-	-	-	-	ADC27D	ADC26D	ADC25D	ADC24D	142
(\$DE)	DIDR2	ADC23D	ADC22D	ADC21D	ADC20D	ADC19D	ADC18D	ADC17D	ADC16D	141
(\$DD)	予約									
(\$DC)	予約									
(\$DB)	予約									
(\$DA)	予約									
(\$D9)	予約									
(\$D8)	予約									
(\$D7)	予約									
(\$D6)	予約									
(\$D5)	予約									
(\$D4)	予約									
(\$D3)	予約									
(\$D2)	予約									
(\$D1)	予約									
(\$D0)	予約									
(\$CF)	予約									
(\$CE)	予約									
(\$CD)	予約									
(\$CC)	予約									
(\$CB)	予約									
(\$CA)	予約									
(\$C9)	予約									
(\$C8)	予約									
(\$C7)	予約									
(\$C6)	UDR	USART データレジスタ								111,119
(\$C5)	UBRRH	-	-	-	-	USART ホールレートレジスタ上位 (UBRR11~8)				114,120
(\$C4)	UBRRL	USART ホールレートレジスタ下位 (UBRR7~0)								114,120
(\$C3)	UCSRD	RXSIE	RXS	SFDE	-	-	-	-	-	113
(\$C2)	UCSRC	UMSEL1	UMSEL0	UPM1	UPM0	USBS	UCSZ1 /UDORD	UCSZ0 /UCPHA	UCPOL	112,120
(\$C1)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	112,119
(\$C0)	UCSRA	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM	111,119

拡張1/レジスタ領域 (2/3)

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
(\$BF)	予約									
(\$BE)	予約									
(\$BD)	TWSD									129
(\$BC)	TWSA									128
(\$BB)	TWSAM								TWAE	129
(\$BA)	TWSSRA	TWDIF	TWASIF	TWCH	TWRA	TWC	TWBE	TWDIR	TWAS	127
(\$B9)	TWSCRIB	-	-	-	-	-	TWAA	TWCMD1	TWCMD0	126
(\$B8)	TWSCRA	TWSHE	-	TWDIE	TWASIE	TWEN	TWSIE	TWPMIE	TWSME	126
(\$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)	OCR1BH									86
(\$8A)	OCR1BL									86
(\$89)	OCR1AH									86
(\$88)	OCR1AL									86
(\$87)	ICR1H									87
(\$86)	ICR1L									87
(\$85)	TCNT1H									86
(\$84)	TCNT1L									86
(\$83)	予約									
(\$82)	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-	85
(\$81)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	84
(\$80)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	83

拡張I/Oレジスタ領域 (3/3)

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
(\$7F)	DIDR1	ADC15D	ADC14D	ADC13D	ADC12D	ADC11D	ADC10D	ADC9D	ADC8D	141
(\$7E)	DIDR0	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	132,141
(\$7D)	ADMUXB	-	-	REFS	-	-	-	-	MUX5	139
(\$7C)	ADMUXA	-	-	-	MUX4	MUX3	MUX2	MUX1	MUX0	139
(\$7B)	ADCSRB	-	-	-	-	ADLAR	ADTS2	ADTS1	ADTS0	140
(\$7A)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	139
(\$79)	ADCH	A/Dデータレジスタ上位バイト (ADC9~8またはADC9~2)								141
(\$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)	PCMSK3	-	-	-	-	PCINT27	PCINT26	PCINT25	PCINT24	36
(\$72)	予約									
(\$71)	予約									
(\$70)	予約									
(\$6F)	TIMSK1	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1	87
(\$6E)	TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0	68
(\$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)	予約									
(\$69)	EICRA	-	-	-	-	ISC11	ISC10	ISC01	ISC00	37
(\$68)	PCICR	-	-	-	-	PCIE3	PCIE2	PCIE1	PCIE0	37
(\$67)	OSCCAL1	-	-	-	-	-	-	CAL11	CAL10	21
(\$66)	OSCCAL0	CAL07	CAL06	CAL05	CAL04	CAL03	CAL02	CAL01	CAL00	22
(\$65)	予約									
(\$64)	PRR	PRTWI	-	PRTIM0	-	PRTIM1	PRSPI	PRUSART0	PRADC	26
(\$63)	予約									
(\$62)	予約									
(\$61)	CLKPR	-	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	21
(\$60)	WDTCR	WDIF	WDIE	WDP3	-	WDE	WDP2	WDP1	WDP0	31

**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と異なり、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うことができます。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。ATtiny828はINとOUT命令で予約した64位置で支援することができるよりも多くの周辺部(機能)を持つ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	11	
\$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	SPMIE	RWWSB	RSIG	RWWSRE	RWFLB	PGWRT	PGERS	SPMEN	150	
\$36 (\$56)	CCP	構成設定変更保護ハイト									10
\$35 (\$55)	MCUCR	-	-	-	-	-	-	IVSEL	-	35	
\$34 (\$54)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	30	
\$33 (\$53)	SMCR	-	-	-	-	-	SM1	SM0	SE	26	
\$32 (\$52)	予約										
\$31 (\$51)	DWDR	デバッグWIRE データレジスタ									143
\$30 (\$50)	ACSR	ACD	ACPMUX2	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	130	
\$2F (\$4F)	ACSRB	HSEL	HLEV	ACLPL	-	ACNMUX1	ACNMUX0	ACPMUX1	ACPMUX0	131	
\$2E (\$4E)	SPDR	SPI データレジスタ									96
\$2D (\$4D)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	95	
\$2C (\$4C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	95	
\$2B (\$4B)	GPIOR2	汎用I/Oレジスタ2									17
\$2A (\$4A)	GPIOR1	汎用I/Oレジスタ1									17
\$29 (\$49)	予約										
\$28 (\$48)	OCR0B	タイマ/カウンタ0 比較Bレジスタ									67
\$27 (\$47)	OCR0A	タイマ/カウンタ0 比較Aレジスタ									67
\$26 (\$46)	TCNT0	タイマ/カウンタ0									67
\$25 (\$45)	TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	66	
\$24 (\$44)	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	65	
\$23 (\$43)	GTCCR	TSM	-	-	-	-	-	-	PSR	90	
\$22 (\$42)	予約										
\$21 (\$41)	EEARL	EEPROMアドレスレジスタ下位バイト (EEAR7~0)									16
\$20 (\$40)	EEDR	EEPROMデータレジスタ									16
\$1F (\$3F)	EEDR	-	-	EEDR1	EEDR0	EEDR2	EEDR3	EEDR4	EEDR5	16	
\$1E (\$3E)	GPIOR0	汎用I/Oレジスタ0									17
\$1D (\$3D)	EIMSK	-	-	-	-	-	-	INT1	INT0	38	
\$1C (\$3C)	EIFR	-	-	-	-	-	-	INTF1	INTF0	38	
\$1B (\$3B)	PCIFR	-	-	-	-	PCIF3	PCIF2	PCIF1	PCIF0	38	
\$1A (\$3A)	予約										
\$19 (\$39)	予約										
\$18 (\$38)	予約										
\$17 (\$37)	予約										
\$16 (\$36)	TIFR1	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1	87	
\$15 (\$35)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0	68	
\$14 (\$34)	PHDE	-	-	-	-	-	PHDEC	-	-	53	
\$13 (\$33)	予約										
\$12 (\$32)	予約										
\$11 (\$31)	予約										
\$10 (\$30)	予約										
\$0F (\$2F)	PUED	-	-	-	-	PUED3	PUED2	PUED1	PUED0	53	
\$0E (\$2E)	PORTD	-	-	-	-	PORTD3	PORTD2	PORTD1	PORTD0	53	
\$0D (\$2D)	DDRD	-	-	-	-	DDD3	DDD2	DDD1	DDD0	53	
\$0C (\$2C)	PIND	-	-	-	-	PIND3	PIND2	PIND1	PIND0	54	
\$0B (\$2B)	PUEC	PUEC7	PUEC6	PUEC5	PUEC4	PUEC3	PUEC2	PUEC1	PUEC0	54	
\$0A (\$2A)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	54	
\$09 (\$29)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	54	
\$08 (\$28)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	54	
\$07 (\$27)	PUEB	PUEB7	PUEB6	PUEB5	PUEB4	PUEB3	PUEB2	PUEB1	PUEB0	54	
\$06 (\$26)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	55	
\$05 (\$25)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	55	
\$04 (\$24)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	55	
\$03 (\$23)	PUEA	PUEA7	PUEA6	PUEA5	PUEA4	PUEA3	PUEA2	PUEA1	PUEA0	55	
\$02 (\$22)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	55	
\$01 (\$21)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	55	
\$00 (\$20)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	56	

(訳注) 原書本位置の注意は前頁に移動しました。

## 27. 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, $PC \leftarrow PC + 2or3$	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, $PC \leftarrow PC + 2or3$	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, $PC \leftarrow PC + 2or3$	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, $PC \leftarrow PC + 2or3$	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, $PC \leftarrow PC + 2or3$	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,T	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



## 28. 注文情報

デバイス	速度 (注1)	電源電圧	温度範囲	外囲器 (注2)	精度 (注3)	注文符号 (注4)
ATtiny828	20MHz	1.7~5.5V	工業用 (-40°C~85°C) (注5)	32A	±10%	ATtiny828-AU
					±2%	ATtiny828R-AU
					±10%	ATtiny828-AUR
					±2%	ATtiny828R-AUR
				32M1-A	±10%	ATtiny828-MU
					±2%	ATtiny828R-MU
					±10%	ATtiny828-MUR
					±2%	ATtiny828R-MUR

注1: 速度対供給電圧については167頁の「速度」をご覧ください。

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: 内部発振器の精度を示します。167頁の表104をご覧ください。

注4: 符号は以下を示します。

- U: 半光沢錫。
- R: テープとリール。

注5: ウェハー(チップ単体)形状で出荷することもできます。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。

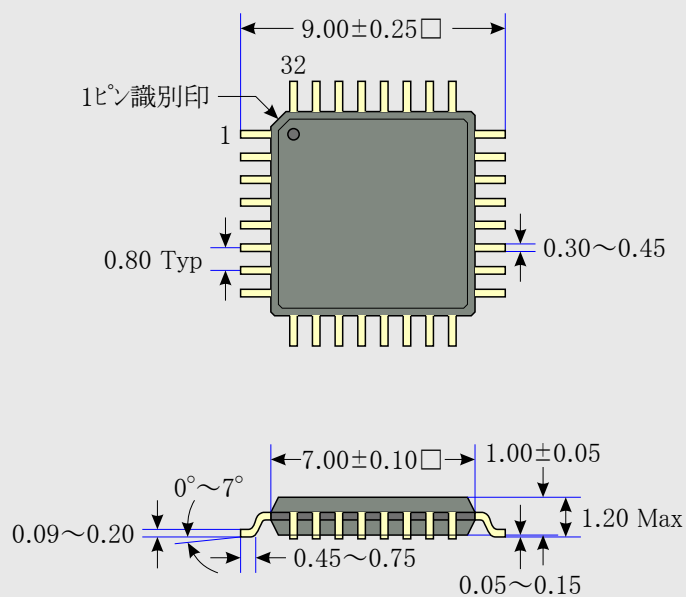
### 外囲器形式

32A	32リード 1.0mm厚 プラスティック4方向平板外囲器 (TQFP)
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)

## 29. 外囲器情報

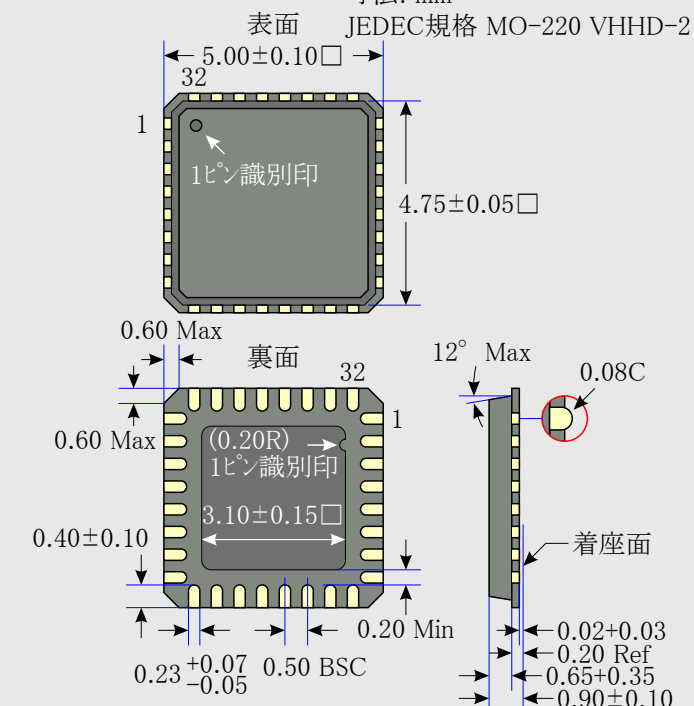
### 29.1. 32A

32リード 0.8mmピッチ プラスティック4方向平板外囲器 (TQFP)  
寸法: mm  
JEDEC規格 MS-026 ABA



### 29.2. 32M1-A

32パッド 0.5mmピッチ 4方向平板リードなし外囲器 (QFN/MLF)  
寸法: mm  
JEDEC規格 MO-220 VHHD-2



## 30. 障害情報

この章の改訂番号は対応するATtiny828デバイスの改訂版を参照してください。

- ULP発振器禁止時のポートピン制限

828-A

### 1. ULP発振器禁止時のポートピン制限 (828-A)

PD3ポートピンは超低電力(ULP)発振器が走行していない時に信頼できる入力として実行することが保証されません。加えて、このピンはULP発振器が禁止される時に内部的にプルダウンされます。PD3がSCLとSCKの信号によって使われるため、TWIとSPIの使用はULPが走行しない時に制限されるかもしれません。

#### 対策/対処

ULP発振器は必要とされる時に自動的に活性化されます。PD3を入力またはTWI/SPIのクロック信号として使うにはウォッチドッグ タイマを活性化してください。ウォッチドッグ タイマはULP発振器を自動的に許可します。

### 31. 改訂履歴

文書改訂	日付	注釈
8371A	2012年8月	初版文書公開

## 目次

特徴	1	11.8. タイマ/カウンタのタイミング	64
1. ピン配置	2	11.9. 8ビット タイマ/カウンタ0用レジスタ	65
1.1. ピン説明	3	12. 16ビット タイマ/カウンタ1	69
2. 概要	4	12.1. 特徴	69
3. 一般情報	5	12.2. 概要	69
3.1. 資料	5	12.3. タイマ/カウンタのクロック元	71
3.2. コード例	5	12.4. 計数器部	71
3.3. データ保持力	5	12.5. 捕獲入力部	72
4. CPU コア	6	12.6. 比較出力部	73
4.1. 構造概要	6	12.7. 比較一致出力部	75
4.2. ALU (Arithmetic Logic Unit)	6	12.8. 動作種別	76
4.3. ステータスレジスタ	6	12.9. タイマ/カウンタのタイミング	80
4.4. 汎用レジスタファイル	7	12.10. 16ビットレジスタのアクセス	81
4.5. スタックポインタ	7	12.11. タイマ/カウンタ1用レジスタ	83
4.6. 命令実行タイミング	8	13. タイマ/カウンタの前置分周器	89
4.7. リセットと割り込みの扱い	8	13.1. 前置分周器リセット	89
4.8. コア関係レジスタ	10	13.2. 外部クロック元	89
5. メモリ	12	13.3. タイマ/カウンタ前置分周器用レジスタ	90
5.1. プログラムメモリ (フラッシュ)	12	14. SPI (直列周辺インターフェース)	91
5.2. データメモリ (SRAM)とレジスタファイル	12	14.1. 特徴	91
5.3. データメモリ (EEPROM)	13	14.2. 概要	91
5.4. メモリ関係レジスタ	16	14.3. SSピンの機能	93
6. クロック体系	18	14.4. データ転送形式	94
6.1. クロック副系統	18	14.5. SPI用レジスタ	95
6.2. クロック元	18	15. USART	97
6.3. システムクロック前置分周器	19	15.1. 特徴	97
6.4. クロック出力緩衝部 (外部クロック出力)	20	15.2. 概要	97
6.5. 始動時間	20	15.3. クロック生成	98
6.6. クロック関係レジスタ	21	15.4. フレーム形式	99
7. 電力管理と休止形態動作	23	15.5. USARTの初期化	100
7.1. 休止形態動作種別	23	15.6. データ送信 - USART送信部	101
7.2. 電力削減レジスタ	24	15.7. データ受信 - USART受信部	103
7.3. 消費電力の最小化	24	15.8. 非同期受信	106
7.4. 電力管理用レジスタ	26	15.9. 複数プロセッサ通信動作	108
8. システム制御とリセット	27	15.10. ホールレート設定例	109
8.1. AVRのリセット	27	15.11. USART用レジスタ	111
8.2. リセット元	27	16. USARTでのSPI動作	115
8.3. 内部基準電圧	29	16.1. 特徴	115
8.4. ウォッチドッグタイマ	29	16.2. 概要	115
8.5. リセット関係レジスタ	30	16.3. クロック生成	115
9. 割り込み	32	16.4. SPIデータ形態とタイミング	115
9.1. 割り込みベクタ	32	16.5. フレーム形式	116
9.2. 外部割り込み	35	16.6. データ転送	117
9.3. 割り込み用レジスタ	35	16.7. AVR SPIとの互換性	118
10. 入出力ポート	40	16.8. MSPIMでのUSART用レジスタ	119
10.1. 概要	40	17. I <sup>2</sup> C適合2線従装置インターフェース	121
10.2. 標準デジタル入出力としてのポート	40	17.1. 特徴	121
10.3. 交換ポート機能	43	17.2. 概要	121
10.4. I/Oポート用レジスタ	53	17.3. 一般的なTWIバスの概念	121
11. PWM付き8ビット タイマ/カウンタ0	57	17.4. TWI従装置動作	125
11.1. 特徴	57	17.5. TWI用レジスタ	126
11.2. 概要	57	18. アナログ比較器	130
11.3. クロック元	58	18.1. アナログ比較器用レジスタ	130
11.4. 計数器部	58	19. A/D変換器	133
11.5. 比較出力部	59	19.1. 特徴	133
11.6. 比較一致出力部	60	19.2. 概要	133
11.7. 動作種別	61	19.3. 操作	134
		19.4. 変換の開始	134

19.5.	前置分周と変換タイミング	135	25.8.	吐き出し電流能力	187
19.6.	チャネル変更と基準電圧選択	136	25.9.	吸い込み電流能力	189
19.7.	雑音低減機能	137	25.10.	低電圧検出器 (BOD)	193
19.8.	アナログ入力回路	137	25.11.	ハントキャップ電圧	195
19.9.	アナログ雑音低減技術	137	25.12.	リセット	195
19.10.	A/D変換の精度定義	138	25.13.	アナログ比較器オフセット	196
19.11.	A/D変換の結果	138	25.14.	内部発振器周波数	197
19.12.	温度測定	138	26.	レジスタ要約	200
19.13.	A/D変換用レジスタ	139	27.	命令要約	204
20.	デバッグWIRE内蔵デバッグ機能	143	28.	注文情報	206
20.1.	特徴	143	29.	外圍器情報	206
20.2.	概要	143	30.	障害情報	207
20.3.	物理インターフェース	143	31.	改訂履歴	208
20.4.	ソフトウェア中断点	143			
20.5.	デバッグWIREの制限	143			
20.6.	デバッグWIRE用レジスタ	143			
21.	ブートローダと書き込み中読み込み(Read-While-Write)を持つ自己プログラミング	144			
21.1.	特徴	144			
21.2.	概要	144			
21.3.	応用とブートローダのフラッシュ領域	144			
21.4.	書き中に読みが可能な領域と不能な領域	144			
21.5.	ブートローダプログラムへの移行	145			
21.6.	ブートローダ構成設定	145			
21.7.	ブートローダ施錠ビット	146			
21.8.	フラッシュメモリの自己プログラミング	147			
21.9.	フラッシュメモリ化けの防止	149			
21.10.	SPM命令使用時フラッシュプログラミング時間	149			
21.11.	自己プログラミング用レジスタ	150			
22.	施錠ビット、ヒューズビット、デバイス識票	151			
22.1.	施錠ビット	151			
22.2.	ヒューズビット	152			
22.3.	デバイス識票銘刻部	153			
22.4.	ソフトウェアからの施錠、ヒューズ、 識票データの読み出し	153			
23.	外部プログラミング	156			
23.1.	メモリの要素	156			
23.2.	並列プログラミング	156			
23.3.	直列プログラミング	163			
23.4.	フラッシュとEEPROM用のプログラミング時間	165			
24.	電気的特性	166			
24.1.	絶対最大定格	166			
24.2.	DC特性	166			
24.3.	速度	167			
24.4.	クロック特性	167			
24.5.	システムとリセットの特性	168			
24.6.	温度感知器	169			
24.7.	2線直列インターフェース特性	169			
24.8.	A/D変換器特性	170			
24.9.	アナログ比較器特性	170			
24.10.	並列プログラミング特性	171			
24.11.	直列プログラミング特性	173			
25.	代表特性	174			
25.1.	活動動作消費電流	174			
25.2.	アイドル動作消費電流	176			
25.3.	パワーダウン動作消費電流	177			
25.4.	リセット消費電流	178			
25.5.	周辺機能部消費電流	179			
25.6.	プルアップ抵抗	181			
25.7.	入力閾値	183			



Enabling Unlimited Possibilities®

**Atmel Corporation**

1600 Technology Drive  
San Jose, CA 95110  
USA  
TEL (+1)(408) 441-0311  
FAX (+1)(408) 487-2600  
[www.atmel.com](http://www.atmel.com)

**Atmel Asia Limited**

Unit 01-5 & 16, 19F  
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 G.K.**

141-0032 東京都品川区  
大崎1-6-4  
新大崎勧業ビル 16F  
アトメル ジャパン合同会社  
TEL (+81)(3)-6417-0300  
FAX (+81)(3)-6417-0370

© 2012 Atmel Corporation. 不許複製 / 改訂:8371A-AVR-08/2012

Atmel®, Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®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。

**お断り:** 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイトに表示する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえばAtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© HERO 2022.

本データシートはAtmelのATtiny828英語版データシート(改訂8371A-08/2012)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力、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、一部の節/項番号が異なります。