

## 特徴

- 高性能、低消費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**131命令**(多くは1周期で実行)
  - 32個の1バイト長**汎用レジスタ**
  - 完全なステディック動作
  - 20MHz時、20MIPSに達する高速動作
  - 2周期実行の乗算命令
- 高耐久不揮発性メモリ部
  - 実装自己書き換え可能な32Kバイト(16K語)**フラッシュ メモリ**内蔵
  - 1Kバイトの**EEPROM**
  - 2Kバイトの内蔵**SRAM**
  - 書き換え回数: 10,000/フラッシュ, 100,000/EEPROM
  - **データ保持力**: 20年/85°C, 100年/25°C
  - 個別施錠ビットを持つ任意の**ブートコード領域**  
チップ内ブートプログラムによる実装書き換え  
真の書き込み中の読み出し動作
  - ソフトウェア保護用の設定可能な**施錠機能**
- JTAG (IEEE 1149.1準拠) インターフェース
  - JTAG規格に従った**境界走査(Boundary-Scan)**能力
  - 広範囲な**内蔵デバッグ機能**
  - **JTAGインターフェース経由でのフラッシュ、EEPROM、ヒューズ、施錠ビットのプログラミング**
- 内蔵周辺機能
  - 独立した前置分周器、比較機能付き2つの**8ビット タイマ/カウンタ**
  - 独立した前置分周器、比較、捕獲機能付き1つの**16ビット タイマ/カウンタ**
  - 専用発振器と**8ビット タイマ/カウンタ**による実時間計数器(RTC)
  - 4つのPWM
  - 8チャンネルの10ビット **A/D変換器**
  - 設定可能な直列**USART**
  - 主装置/従装置動作**SPI直列インターフェース**
  - 開始条件検出器付き**多用途直列インターフェース**
  - 設定可能な専用発振器付き**ウォッチドッグ タイマ**
  - **アナログ比較器**
  - **ピン変化での割り込み**と起動復帰
- 特殊マイクロ コントローラ機能
  - 電源ONリセット回路と設定可能な**低電圧検出器(BOD)**
  - **校正可能な内蔵RC発振器**
  - 外部及び内部の**割り込み**
  - **アイドル、A/D変換雑音低減、パワーセーブ、パワーダウン、スタンバイ**の5つの**低消費動作**
- I/Oと外周器 (ATmega325P:ATmega3250P)
  - 54:69ビットの**設定可能なI/O**
  - **64リードTQFP、64パッドQFN/MLF:100リードTQFP**
- 動作速度/電圧
  - ATmega325PV/ATmega3250PV:  
0~4MHz/1.8~5.5V      0~10MHz/2.7~5.5V
  - ATmega325P/ATmega3250P:  
0~10MHz/2.7~5.5V      0~20MHz/4.5~5.5V
- 温度範囲
  - -40~85°C (工業用)
- 超低消費電力
  - 活動動作  
1MHz/1.8V : 420µA
  - パワーダウン動作  
40nA/1.8V
  - パワーセーブ動作  
750nA/1.8V



8ビット **AVR®**  
マイクロ コントローラ  
実装書き換え可能な  
32Kバイト  
フラッシュ メモリ内蔵

ATmega325P  
ATmega325PV  
ATmega3250P  
ATmega3250PV

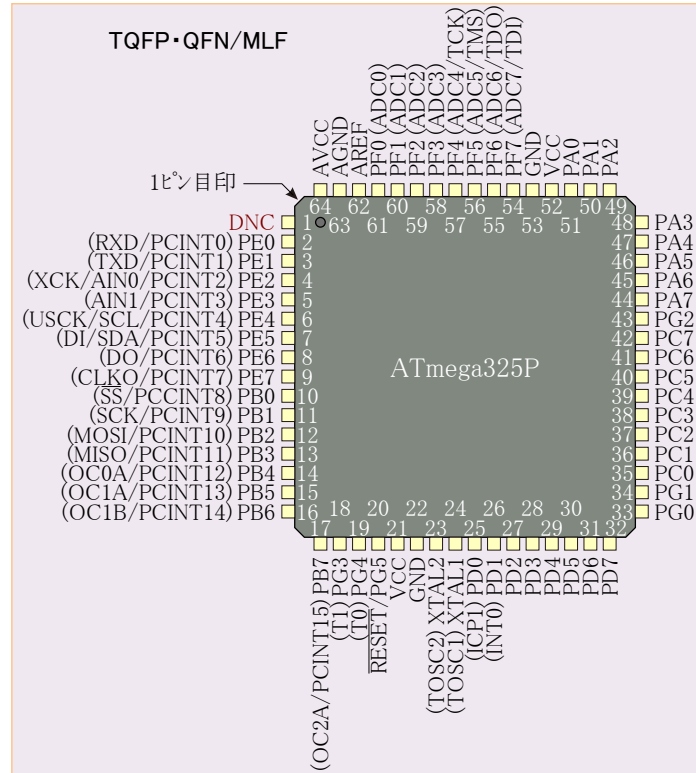
暫定

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

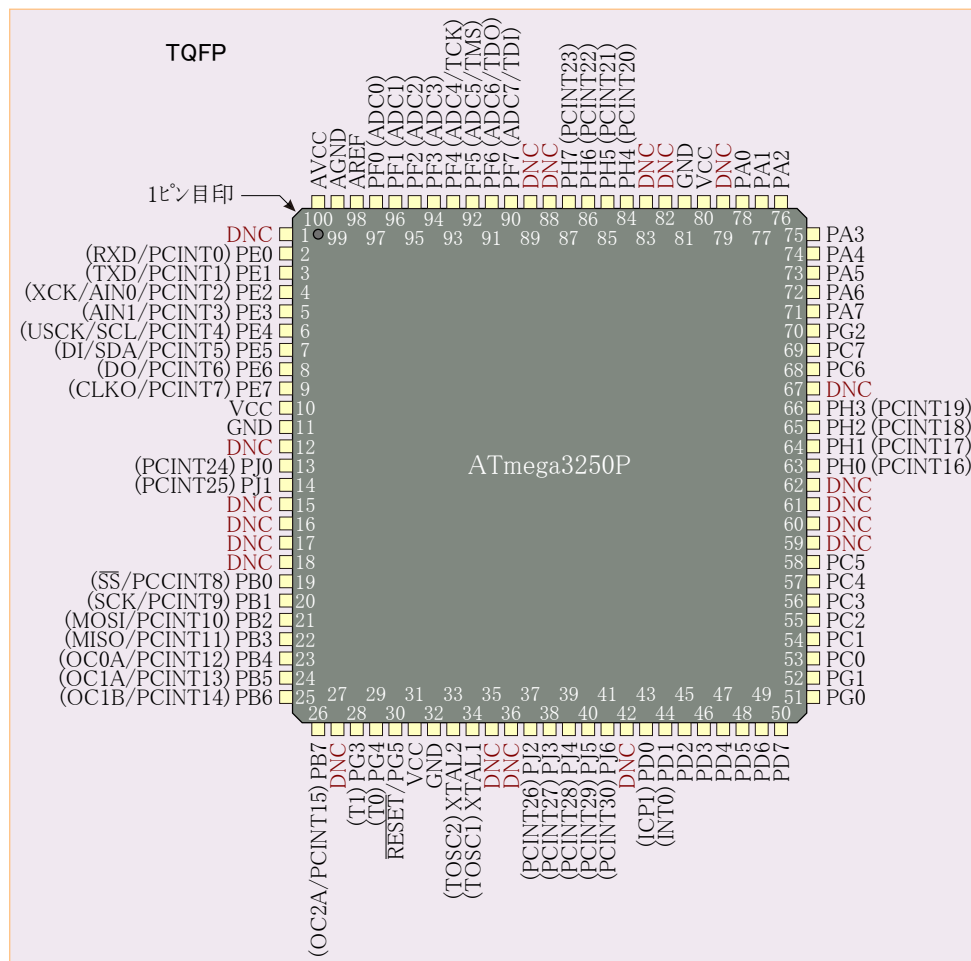
Rev. 8023F-07/09, 8023FJ10-05/23



## 1. ピン配置



**注:** QFN/MLF外周器底面中央の大パッドは金属で作られ、内部でGNDに接続されています。これは良好な機構的安定を保証するため、基板に貼り付けるか、半田付けされるべきです。この中央パッドが未接続の場合、基板から脱落するかもしれません。



### 1.1. お断り

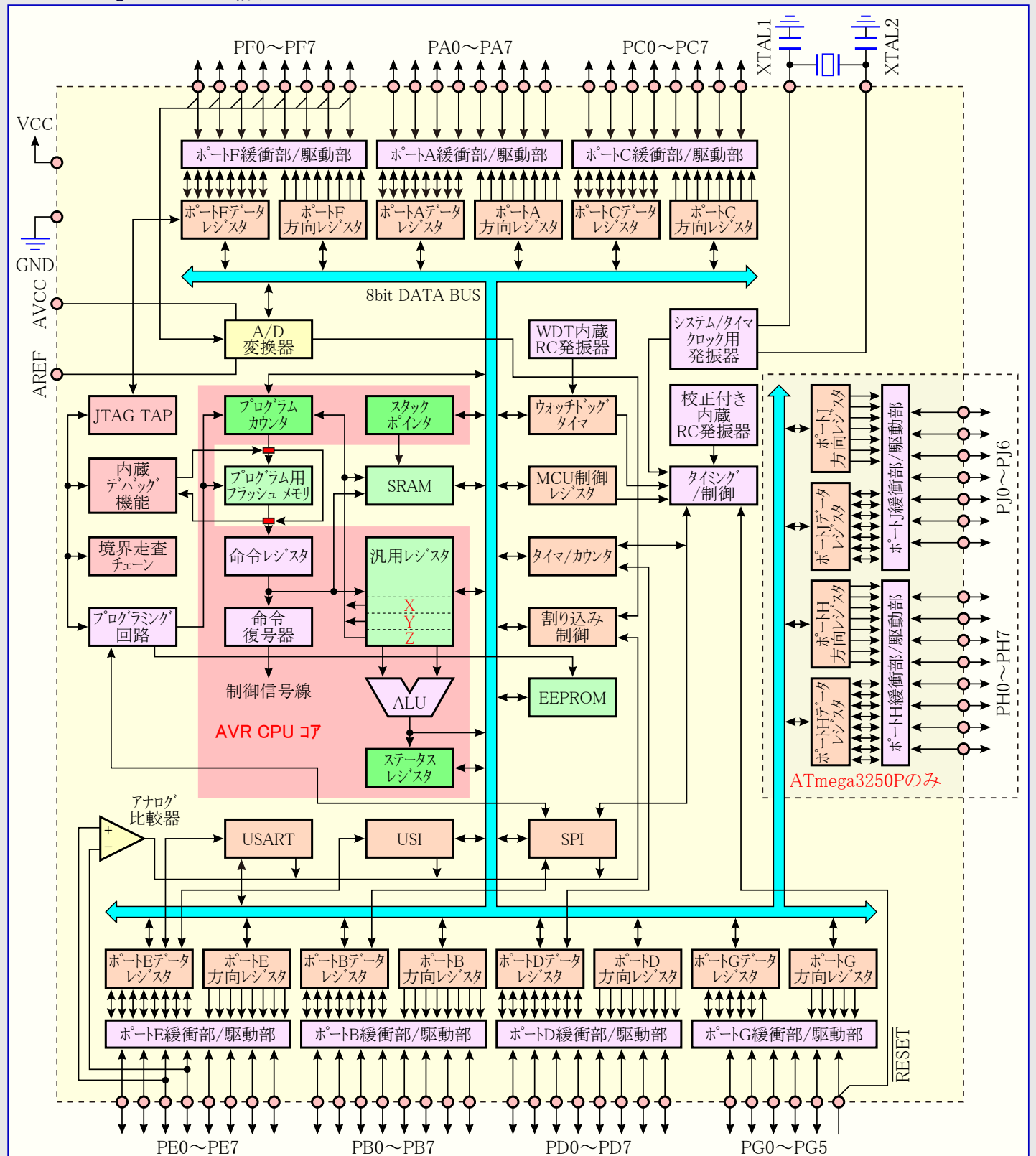
本データシート内で示された代表値はシミュレーションと同じ製法技術で製造された他のAVRマイクロコントローラの特性を基にしています。最小と最大の値はデバイスの特性が記載された後に利用可能になります。

## 2. 概要

ATmega325P/3250PはAVR RISC構造の低消費CMOS 8ビット マイクロ コントローラです。1周期で実行する強力な命令はMHzあたり1 MIPSにも達し、実行速度対電力消費の最適化が容易に行えます。

### 2.1. 構成図

図2-1. ATmega325P/3250P構成図



AVRは32個の汎用レジスタと豊富な命令群を兼ね備えています。32個の全レジスタはALU(Arithmetic Logic Unit)に直結され、レジスタ間命令は1クロック周期で実行されます。AVR構造は現状のCISC型マイクロコントローラに対し、最大10倍の単位処理量向上効果があります。

ATmega325P/3250Pは次の特徴、書き込み中読める能力を持つ32Kバイトの実装書き換え可能なフラッシュメモリと1KバイトのEEPROM、2KバイトのSRAM、54/69本の汎用入出力線、32個の汎用作業レジスタ、境界走査用のJTAGインターフェース、内蔵デバッグとプログラミング機能、比較動作を含む柔軟性のある3つのタイマ/カウンタ、内部及び外部割り込み、設定変更可能な直列USART、開始条件検出器付き多用途直列インターフェース(USI)、8チャネルの10ビットA/D変換器、設定変更可能な内部発振器付きウォッチドッグタイマ、SPI直列ポート、ソフトウェアで選べる5つの低消費動作機能を提供します。アイドル動作ではCPU動作を停止しますが、SRAM、タイマ/カウンタ、SPIポート、割り込み機能は有効で、動作を継続します。パワーダウン動作ではレジスタの内容は保護されますが、発振器が停止するため、以降のハードウェアリセットか割り込みまで他の全機能を禁止(無効に)します。パワーセーブ動作では非同期タイマが動作を継続し、デバイスのその他が停止中であっても基準タイマの継続走行を使用者に許します。A/D変換雑音低減動作ではA/D変換中の切り替え雑音を最小とするために、非同期タイマとA/D変換器を除く全ての周辺機能とCPUが停止します。スタンバイ動作ではクリスタル発振子/セラミック振動子用発振器が動作し、一方デバイスのその他は休止します。これは低消費電力と非常に速い起動の組み合わせを許します。

本デバイスはAtmelの高密度不揮発性メモリ技術を使って製造されています。内蔵の実装書き換え(ISP)可能なプログラム用フラッシュメモリは規定の不揮発性メモリ書き込み器、SPI直列インターフェース経由、AVRコア上ポートプログラムの実行によって再書き込みができます。ポートプログラムは応用領域フラッシュメモリ内の応用プログラムの読み込みに、どのインターフェースでも使えます。ポート領域フラッシュメモリ内のプログラムは真の「書き込み中の読み出し可」動作により、応用領域フラッシュメモリ更新中も実行を継続します。モノリシックチップ上の実装書き換え可能なフラッシュメモリと、8ビットRISC型CPUの組み合わせによるAtmel ATmega325P/3250Pは多くの組み込み制御の応用に対して高度な柔軟性と対費用効果をもたらす強力なマイクロコントローラです。

ATmega325P/3250P AVRはCコンパイラ、マクロアセンブラ、デバッグ、シミュレータ、インサーキットエミュレータ、評価キットを含む専用のプログラム及びシステム開発ツールで支援されます。

## 2.2. ATmega325PとATmega3250P間の違い

ATmega325PとATmega3250Pはピン数とピン配置が異なるだけです。表2-1は2つのデバイスについて各種形態の違いを一覧します。

表2-1. メモリ容量/形態対比表

デバイス名	フラッシュメモリ	EEPROM	SRAM	汎用入出力ピン数
ATmega325P	32Kバイト	1Kバイト	2Kバイト	53+1
ATmega3250P				68+1

## 2.3. ピン概要

次節はI/Oピンの特殊機能を記述します。

### 2.3.1. VCC

デジタル電源ピン。

### 2.3.2. GND

接地ピン。

### 2.3.3. PA7～PA0 (ポートA)

ポートAは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートA出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートAピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートAピンはHi-Zにされます。

### 2.3.4. PB7～PB0 (ポートB)

ポートBは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートB出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートBピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートBピンはHi-Zにされます。

ポートBは他のポートより優れた駆動能力を持ちます。

ポートBは43頁で一覧されるATmega325P/3250Pの様々な特殊機能も扱います。

### 2.3.5. PC7～PC0 (ポートC)

ポートCは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートC出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートCピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートCピンはHi-Zにされます。

### 2.3.6. PD7～PD0 (ポートD)

ポートDは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートD出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートDピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートDピンはHi-Zにされます。

ポートDは45頁で一覧されるATmega325P/3250Pの様々な特殊機能も扱います。

## 2.3.7. PE7～PE0 (ポートE)

ポートEは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートE出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートEピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートEピンはHi-Zにされます。

ポートEは[46頁](#)で一覧されるATmega325P/3250Pの様々な特殊機能も扱います。

## 2.3.8. PF7～PF0 (ポートF)

ポートFはA/D変換器へのアナログ入力として扱います。

A/D変換器が使われない場合、ポートFは8ビット双方向入出力ポートとしても扱います。ポートFピンは(ビット毎に選ばれる)内蔵プルアップ抵抗を提供できます。ポートF出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートFピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートFピンはHi-Zにされます。JTAGインターフェースが許可されると、リセットが起きてもPF7(TDI)、PF5(TMS)、PF4(TCK)ピンのプルアップ抵抗は活性(有効)にされます。

ポートFはJTAGインターフェース機能も扱います。

## 2.3.9. PG5～PG0 (ポートG)

ポートGは固定内蔵プルアップ抵抗付き1ビット入力(PG5)と(ビット毎に選ばれる)内蔵プルアップ抵抗付き5ビット双方向入出力(PG4～0)ポートです。ポートG出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートGピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートGピンはHi-Zにされます。

ポートGは[50頁](#)で一覧されるATmega325P/3250Pの様々な特殊機能も扱います。

## 2.3.10. PH7～PH0 (ポートH)

ポートHは(ビット毎に選ばれる)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートH出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートHピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートHピンはHi-Zにされます。

ポートHは[51頁](#)で一覧されるATmega3250Pの様々な特殊機能も扱います。

## 2.3.11. PJ6～PJ0 (ポートJ)

ポートJは(ビット毎に選ばれる)内蔵プルアップ抵抗付き7ビット双方向入出力ポートです。ポートJ出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力としてプルアップ抵抗が活性(有効)なら、外部的にLowへ引かれたポートJピンは電流を吐き出します。リセット条件が有効になるとクロックが動いていなくてもポートJピンはHi-Zにされます。

ポートJは[53頁](#)で一覧されるATmega3250Pの様々な特殊機能も扱います。

## 2.3.12. RESET

リセット入力。最小パルス幅より長いこのピンのLowレベルはクロックが動いていなくてもリセットを生成します。最小パルス幅は[195頁の「システムとリセットの特性」](#)で得られます。より短いパルスはリセットの生成が保証されません。

## 2.3.13. XTAL1

発振器反転増幅器への入力と内部クロック操作回路への入力。

## 2.3.14. XTAL2

発振器反転増幅器からの出力。

## 2.3.15. AVCC

AVCCはポートFとA/D変換器用供給電圧(電源)ピンです。例えば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、[VCCから低域通過濾波器を通して接続](#)されるべきです。

## 2.3.16. AREF

AREFはA/D変換器用アナログ基準(電圧)ピンです。



### 3. 資料

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

### 4. データ保持力

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

### 5. コード例について

この文書はデバイスの様々な部分の使用法を手短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。

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ファイルがインクルードされることが前提です。拡張I/O領域に配置したI/Oレジスタに対し、**IN**, **OUT**, **SBIS**, **SBIC**, **CBI**, **SBI**命令は拡張I/O領域へのアクセスを許す命令に置き換えられなければなりません。代表的には**SBRS**, **SBRC**, **SBR**, **CBR**命令と組み合わせた**LDS**, **STS**命令です。

## 6. AVR CPU コア

### 6.1. 概要

ここでは一般的なAVRコア構造について説明します。このCPUコアの主な機能は正しいプログラム実行を保証することです。従ってCPUはメモリアクセス、計算実行、周辺制御、割り込み操作ができなければなりません。

### 6.2. 構造概要

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

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

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

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

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

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全SRAM容量とSRAM使用量でのみ制限されます。全てのユーザープログラムはリセット処理ルーチンで(サブルーチン呼び出しや割り込みが実行される前に)、スタックポインタ(SP)を初期化しなければなりません。SPはI/O空間で読み書きアクセスが可能です。データ用SRAMはAVR構造で支援される5つの異なるアドレス指定種別を通して容易にアクセスできます。

AVR構図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。

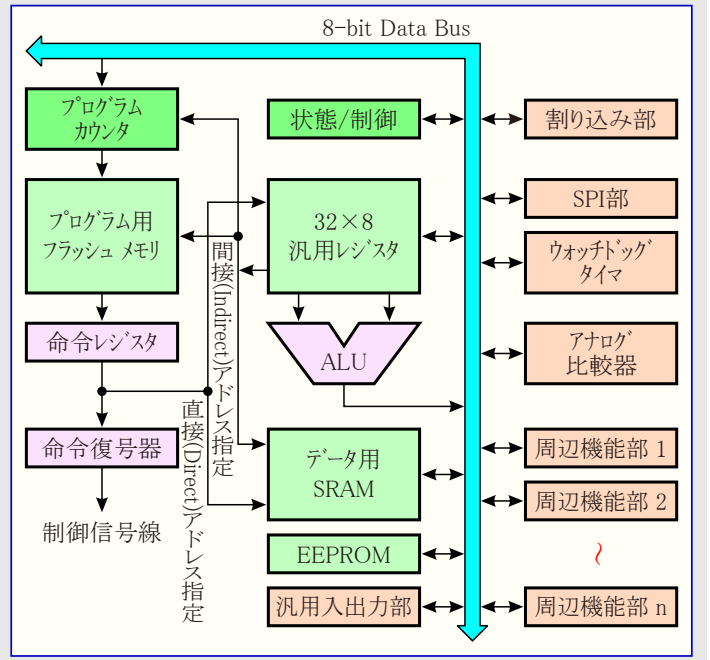
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(1ビット)があります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタ、SPI、他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリは直接またはレジスタファイルの次のデータ空間位置\$20～\$5Fとしてアクセスできます。加えてATmega325P/3250PにはST/STS/STDとLD/LDS/LDD命令だけ使えるSRAM内の\$60～\$FFに拡張I/O空間があります。

### 6.3. ALU (Arithmetic Logic Unit)

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

図6-1. AVR MCU構造



## 6.4. ステータス レジスタ

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

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

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

AVRのステータスレジスタ(SREG)は次のように定義されます。

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

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

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

#### • ビット6 – T : ビット変数 (Bit Copy Storage)

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

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

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

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

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

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

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

#### • ビット2 – N : 負フラグ (Negative Flag)

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

#### • ビット1 – Z : ゼロフラグ (Zero Flag)

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

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

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



## 6.5. 汎用レジスタ ファイル

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

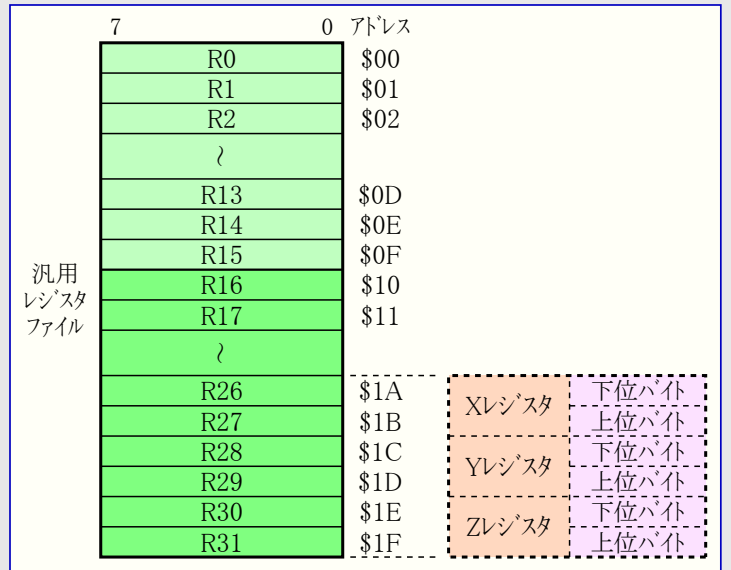
- 1つの8ビット出力オペランドと1つの8ビットの結果入力
- 2つの8ビット出力オペランドと1つの8ビットの結果入力
- 2つの8ビット出力オペランドと1つの16ビットの結果入力
- 1つの16ビット出力オペランドと1つの16ビットの結果入力

図6-2.はCPU内の32個の汎用作業レジスタの構造を示します。

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

図6-2.で示されるように各レジスタは使用者データ空間の最初の32位置へ直接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6-2. AVR CPU 汎用レジスタ構成図

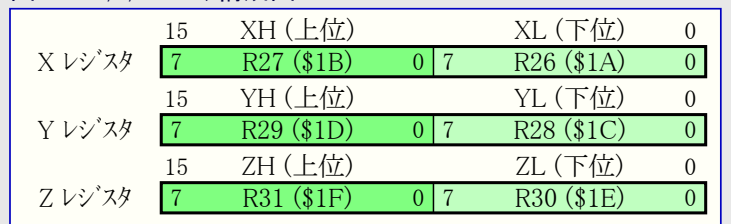


### 6.5.1. Xレジスタ, Yレジスタ, Zレジスタ

R26~R31レジスタには通常用途の使用にいくつかの追加機能があります。これらのレジスタはデータ空間の間接アドレス指定用の16ビットアドレスポインタです。3つのX,Y,Z間接アドレスレジスタは図6-3.で記載したように定義されます。

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

図6-3. X,Y,Zレジスタ構成図



## 6.6. スタック ポインタ

スタックは主に一時データの保存、局所変数の保存、割り込みとサブルーチン呼び出し後の戻りアドレスの保存に使われます。スタックが高位メモリから低位メモリへ伸長するように実行されることに注意してください。スタック ポインタレジスタは常にこのスタックの先頭(訳注:次に使われるべき位置)を指し示します。スタック ポインタはサブルーチンや割り込みのスタックが配置されるデータSRAMのスタック領域を指し示します。スタック 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サブルーチン呼び出しの実行や割り込みの許可の何れにも先立ってプログラムによって定義されなければなりません。初期スタック ポインタ値は内部SRAMの最終アドレスに等しく、スタック ポインタはSRAMの先頭以上に設定されなければなりません。12ページの図7-2.をご覧ください。

スタック ポインタの詳細については表6-1.をご覧ください。

表6-1. スタック ポインタ命令

命令	スタック ポインタ	内容
PUSH	-1	データがスタック上に押し込まれます。
CALL,ICALL,RCALL	-2	サブルーチン呼び出しまたは割り込みでの戻りアドレスがスタック上に押し込まれます。
POP	+1	データがスタックから引き出されます。
RET,RETI	+2	サブルーチンまたは割り込みからの復帰での戻りアドレスがスタックから引き出されます。

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

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

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	—	—	—	(SP12)	SP11	SP10	SP9	SP8	SPH
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	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	0	0	0	0	0	0	0	0	

### 6.7. 命令実行タイミング

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

図6-4.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1 MIPSを達成するための基本的なパイプラインの概念です。

図6-5.はレジスタファイルに対する内部タイミングの概念を示します。単一クロック周期で2つのレジスタオペランドを使うALU操作が実行され、その結果が転送先レジスタへ書き戻されます。

図6-4. 命令の取得と実行の並列動作

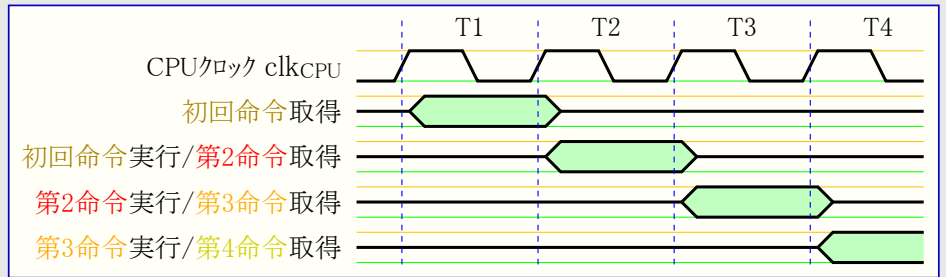
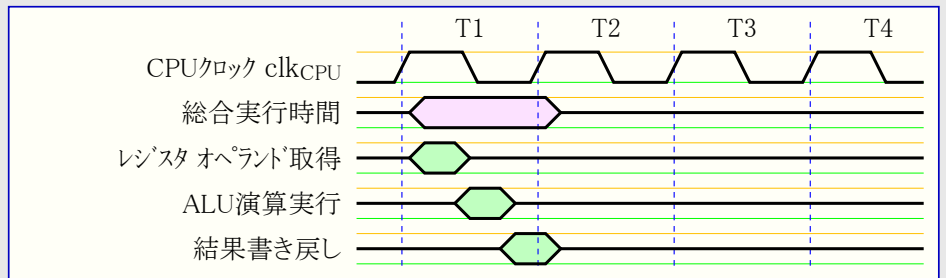


図6-5. 1周期ALU命令



### 6.8. リセットと割り込みの扱い

AVRは多くの異なる割り込み元を提供します。これらの割り込みと独立したリセットベクタ各々はプログラムメモリ空間内に独立したプログラムベクタを持ちます。全ての割り込みは割り込みを許可するために、ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない。個別の許可ビットを割り当てられます。BLB02またはBLB12 プート施錠ビットがプログラム(0)されると、プログラムカウンタ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172頁の「メモリプログラミング」章をご覧ください。

既定でのプログラムメモリ空間の最下位アドレスはリセットと割り込みの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できます。より多くの情報については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。163頁の「ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング」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まれます。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ず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, EEMWE     ;EEPROM主書き込み許可
SBI     EECR, EEWE      ;EEPROM書き込み開始
OUT     SREG, R16       ;ステータスレジスタを復帰
```

## C言語プログラム例

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

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

## アセンブリ言語プログラム例

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

## C言語プログラム例

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

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

### 6.8.1. 割り込み応答時間

許可した全てのAVR割り込みに対する割り込み実行応答は最小4クロック周期です。4クロック周期後、実際の割り込み処理ルーチンに対する**プログラムベクタアドレス**が実行されます。この4クロック周期時間中にプログラムカウンタ(PC)がスタック上に保存(プッシュ)されます。このベクタは標準的に割り込み処理ルーチンへの無条件分岐で、この分岐は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。この増加は選んだ休止形態からの起動時間に加えてです。

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

## 7. AVR メモリ

### 7.1. 概要

この項はATmega325P/3250Pの各種メモリを記述します。AVR構造にはプログラムメモリ空間とデータメモリ空間の2つの主なメモリ空間があります。加えてATmega325P/3250Pはデータ保存用EEPROMメモリが特徴です。3つのメモリ空間全ては一般的な直線的アドレスです。

### 7.2. 実装書き換え可能なプログラム用フラッシュメモリ

ATmega325P/3250Pはプログラム保存用に実装書き換え可能な32Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、このフラッシュメモリは16K×16ビットとして構成されます。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。

フラッシュメモリは最低10,000回の消去/書き込み回数の耐久性があります。ATmega325P/3250Pのプログラムカウンタ(PC)は14ビット幅、従って16Kプログラムメモリ位置のアドレス指定です。ブートプログラム領域の操作と関係するソフトウェア保護用ブート施錠ビットは163頁の「[ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング](#)」の詳細で記述されます。172頁の「[メモリプログラミング](#)」はSPI、JTAG、並列プログラミング動作でのフラッシュメモリプログラミングの詳細な記述を含みます。

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

命令の取得と実行のタイミング図は10頁の「[命令実行タイミング](#)」で示されます。

図7-1. プログラムメモリ配置図

応用プログラム用 フラッシュメモリ	\$0000
ブートプログラム用 フラッシュメモリ	\$3FFF

### 7.3. データ用SRAMメモリ

図7-2はATmega325P/3250PのSRAMメモリ構成方法を示します。

ATmega325P/3250PはINやOUT命令で予約した64位置で支援されるより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。SRAM(データ空間)内\$60～\$FFの拡張I/O空間に対してLD/LDS/LDDとST/STS/STD命令だけが使えます。

下位2304データメモリ位置はレジスタファイル、I/Oメモリ、拡張I/Oメモリ、データ用内蔵SRAMに充てます。先頭の32位置はレジスタファイル、次の64位置は標準I/Oメモリ、その次の160位置は拡張I/Oメモリ、そして次の2048位置はデータ用内蔵SRAMに充てます。

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

直接アドレス指定はデータ空間全体に届きます。

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

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

ATmega325P/3250Pの32個の汎用レジスタ、64個のI/Oレジスタ、160個の拡張I/Oレジスタ、2048バイトのデータ用内蔵SRAMはこれら全てのアドレス指定種別を通して全部アクセスできます。レジスタファイルは9頁の「[汎用レジスタファイル](#)」で記述されます。

図7-2. データメモリ配置図

		アドレス
レジスタファイル (32×8)	R0 ～ R31	\$0000 ～ \$001F
I/Oレジスタ (64×8)	\$00 ～ \$3F	\$0020 ～ \$005F
拡張I/Oレジスタ (160×8)	\$0060 ～ \$00FF	\$0060 ～ \$00FF
内蔵SRAM (2K×8)	\$0100 ～ \$08FF	\$0100 ～ \$08FF

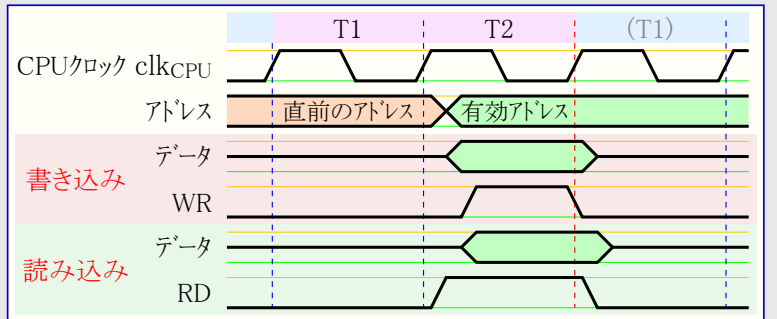
注: 赤字はI/Oアドレス

#### 7.3.1. データメモリアクセスタイミング

本節は内部メモリアクセスに対する一般的なアクセスタイミングの概念を記述します。データ用内蔵SRAMアクセスは図7-3.で記載されるように2 clk<sub>CPU</sub>周期で実行されます。

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

図7-3. データ用内蔵SRAMアクセス周期





## 7.4. データ用EEPROMメモリ

ATmega325P/3250Pは1KバイトのデータEEPROMを含みます。それは単一バイトが読み書きできる分離したデータ空間として構成されます。EEPROMは最低100,000回の消去/書き込み回数の耐久性があります。CPUとEEPROM間のアクセスは以降の[EEPROMアドレスレジスタ](#)、[EEPROMデータレジスタ](#)、[EEPROM制御レジスタ](#)で詳細に記述されます。

EEPROMへのSPI、JTAG、並列プログラミングの詳細記述については各々[183頁](#)、[186頁](#)、[175頁](#)をご覧ください。

### 7.4.1. EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROMの書き込み(訳注:原文はアクセス)時間は[15頁の表7-1](#)で与えられます。(書き込みは)自己タイミング機能ですが、使用者ソフトウェアは**次バイトが書ける時を検知**してください。使用者コードがEEPROMに書く命令を含む場合、いくつかの予防処置が取られなければならない。嚴重に濾波した電源では電源投入/切断でVCCが緩やかに上昇または下降しそうです。これはデバイスが何周期かの時間、使われるクロック周波数に於いて最小として示されるより低い電圧で走行する原因になります。これらの状態で問題を避ける方法の詳細については以下の「[EEPROMデータ化けの防止](#)」をご覧ください。

予期せぬEEPROM書き込みを防止するため、特別な書き込み手順に従わなければならない。この詳細については「[EEPROM制御レジスタ](#)」の記述を参照してください。

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

### 7.4.2. パワーダウン休止動作中のEEPROM書き込み

EEPROM書き込み動作が活動中に**パワーダウン休止動作**へ移行すると、EEPROM書き込み動作が継続し、EEPROM書き込み時間が過ぎ去ってしまう前に完了します。しかし、書き込み動作が完了されると、発振器が動作を継続し、結果としてデバイスはパワーダウン動作へ完全に移行しません。従ってパワーダウンへ移行する前に、EEPROM書き込み動作が完了される(**EEWE=0**)ことの確認が推奨されます。(訳補: パワーダウン移行後もEEPROM書き込みは正常に完了するが、その後発振器が止まらないことの注意)

### 7.4.3. EEPROMデータ化けの防止

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

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

EEPROMデータ化けは次の推奨設計によって容易に避けられます。

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

## 7.5. I/O メモリ (レジスタ)

ATmega325P/3250PのI/O空間定義は[217頁の「レジスタ要約」](#)で示されます。

ATmega325P/3250Pの全てのI/Oと周辺機能はI/O空間に配置されます。全てのI/O位置はI/O空間と32個の汎用作業レジスタ間のデータ転送を行う**LD/LDS/LDD**命令と**ST/STS/STD**命令によってアクセスされます。アドレス範囲\$00～\$1F内のI/Oレジスタは**SBI**命令と**CBI**命令の使用で直接的にビットアクセス可能です。これらのレジスタでは**SBIS**と**SBIC**命令の使用によって単一ビット値が検査できます。より多くの詳細については「[命令要約](#)」章を参照してください。I/O指定命令**IN**と**OUT**を使う時はI/Oアドレス\$00～\$3Fが使われなければならない。LD命令とST命令を使い、データ空間としてI/Oレジスタをアクセスする時はこれらのアドレスに\$20が加算されなければならない。

ATmega325P/3250Pは**IN**や**OUT**命令で予約された64位置で支援されるより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。SRAM(データ空間)内\$60～\$FFの拡張I/O領域に対してはLD/LDS/LDDとST/STS/STD命令だけが使えます。

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

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

I/Oと周辺制御レジスタは以降の項で説明されます。

ATmega325P/3250Pは3つの汎用I/Oレジスタを含みます。これらのレジスタはどの情報の格納にも使え、特に全体変数や状態フラグの格納に有用です。(I/O)アドレス範囲\$00～\$1Fの汎用I/Oレジスタは**SBI**、**CBI**、**SBIS**、**SBIC**命令の使用で直接ビットアクセスが可能です。



## 7.6. メモリ関係レジスタ

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

ビット	15	14	13	12	11	10	9	8	
\$22 (\$42)	–	–	–	–	–	–	EEAR9	EEAR8	EEARH
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	不定	不定	
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15～10 – 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9～0 – EEAR9～0 : EEPROMアドレス (EEPROM Address)

EEPROMアドレス レジスタ(EEARHとEEARL)は1KバイトEEPROM空間のEEPROMアドレスを指定します。EEPROMデータ バイトは0～1023間で直線的に配されます。EEARの初期値は不定です。EEPROMがアクセスされるであろう前に適切な値が書かれねばなりません。

### 7.6.2. EEPROMデータ レジスタ (EEPROM Data Register) EEDR

ビット	7	6	5	4	3	2	1	0	
\$20 (\$40)	(MSB)							(LSB)	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7～0 – EEDR7～0 : EEPROMデータ (EEPROM Data)

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

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

ビット	7	6	5	4	3	2	1	0	
\$1F (\$3F)	–	–	–	–	EERIE	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	不定	0	

#### • ビット7～4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

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

EERIEの1書き込みはステータス レジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、EEPROM操作可割り込みを許可します。EERIEの0書き込みは、この割り込みを禁止します。EEPROM操作可割り込みはEEWEが解除(0)されていると継続する割り込みを発生します。

#### • ビット2 – EEMWE : EEPROM主書き込み許可 (EEPROM Master Write Enable)

EEMWEビットはEEPROM書き込み許可(EEWE)ビットの1設定がEEPROM書き込みの原因となるかどうかを決めます。EEMWEが設定(1)されるとき、EEWEの1設定は選ばれたアドレスのEEPROMにデータを書きます。EEMWEが0の場合、EEWEの1設定は無効です。EEMWEがソフトウェアによって設定(1)されてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。EEPROM書き込み手順については「書き込み許可(EEWE)ビット」の記述をご覧ください。

#### • ビット1 – EEWE : EEPROM書き込み許可 (EEPROM Write Enable)

EEPROM書き込み許可信号(EEWE)はEEPROMへの書き込みスロープです。アドレスとデータが正確に設定されると、EEPROMへその値を書くためにEEWEビットは1を書かれなければなりません。論理1がEEWEへ書かれる前にEEPROM主書き込み許可(EEMWE)ビットは1を書かれなければならず、さもないとEEPROM書き込みは行われません。EEPROMを書くとき、次の手順に従うべきです(手順3.と4.の順番は重要ではありません)。

1. EEPROM書き込み許可(EEWE)ビットが0になるまで待ちます。
2. SPM命令制御/状態レジスタ(SPMCSR)のSPM操作許可(SPMEN)ビットが0になるまで待ちます。
3. 今回のEEPROMアドレスをEEPROMアドレス レジスタ(EEAR)に書きます。(任意、省略可)
4. 今回のEEPROMデータをEEPROMデータ レジスタ(EEDR)に書きます。(任意、省略可)
5. EEPROM制御レジスタ(EECR)のEEPROM主書き込み許可(EEMWE)ビットに1を、EEPROM書き込み許可(EEWE)ビットに0を同時に書きます。

6. EEMWEビット設定後4クロック周期内にEEPROM書き込み許可(EEMWE)ビットへ論理1を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMはプログラム(書き込みが)できません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを検査しなければなりません。2.はソフトウェアがフラッシュメモリをプログラム(書き込みを)することをCPUに許すブートローダを含む場合だけ関係します。フラッシュメモリが決してCPUによって更新されないなら、2.は省略できます。ブートプログラミングについての詳細に関しては163頁の「ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

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

書き込みアクセス時間が経過されると、EEPROM書き込み許可(EEMWE)ビットはハードウェアによって解除(0)されます。使用者ソフトウェアは次バイトを書く前にこのビットをポーリングし、0を待つことができます。EEMWEが設定(1)されてしまうと、次の命令が実行される前にCPUは2周期停止されます。

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

EEPROM読み込み許可信号(EERE)はEEPROMへの読み込みストロープです。EEARに正しいアドレスが設定されると、EEPROM読み出しを起動するためにEEREビットは論理1を書かれなければなりません。EEPROM読み出しアクセスは(その)1命令で行われ、要求したデータは直ちに利用できます。EEPROMが読まれるとき、次の命令が実行される前にCPUは4周期停止されます。

使用者は読み込み操作を始める前にEEMWEビットをポーリングすべきです。書き込み操作が実行中の場合、EEPROMアドレスレジスタ(EEAR)の変更もEEPROM読み込みもできません。

EEPROMアクセスの時間に校正済み内蔵RC発振器が使われます。表7-1.はCPUからのEEPROMアクセスに対する代表的な書き込み時間を示します。

表7-1. EEPROM書き込み時間

項目	校正付き内蔵RC 発振器周期数	代表
EEPROM書き込み(CPU)	27072	3.4ms

次のコード例はアセンブリ言語とC言語でのEEPROM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, EEMWE          ;EEPROM書き込み完了ならばスキップ
             RJMP    EEPROM_WR         ;以前のEEPROM書き込み完了まで待機
;
             OUT     EEARH, R18         ;EEPROMアドレス上位バイト設定
             OUT     EEARL, R17         ;EEPROMアドレス下位バイト設定
             OUT     EEDR, R16         ;EEPROM書き込み値を設定
             SBI     EECR, EEMWE       ;EEPROM主書き込み許可ビット設定
             SBI     EECR, EEMWE       ;EEPROM書き込み開始(書き込み許可ビット設定)
             RET                        ;呼び出し元へ復帰
```

### C言語プログラム例

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    while(EECR & (1<<EEMWE));          /* 以前のEEPROM書き込み完了まで待機 */
    EEAR = uiAddress;                   /* EEPROMアドレス設定 */
    EEDR = ucData;                      /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMWE);                 /* EEPROM主書き込み許可 */
    EECR |= (1<<EEMWE);                 /* EEPROM書き込み開始 */
}
```

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

#### アセンブリ言語プログラム例

```
EEPROM_RD:  SBIC    EECR, EWE        ;EEPROM書き込み完了ならばスキップ
             RJMP    EEPROM_RD      ;以前のEEPROM書き込み完了まで待機
;
             OUT     EEARH, R18      ;EEPROMアドレス上位バイト設定
             OUT     EEARL, R17      ;EEPROMアドレス下位バイト設定
             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 uiAddress)
{
    while(EECR & (1<<EWE));          /* 以前のEEPROM書き込み完了まで待機 */
    EEAR = uiAddress;                 /* 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 */
}
```

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

ビット	7	6	5	4	3	2	1	0	
\$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	

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

ビット	7	6	5	4	3	2	1	0	
\$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	

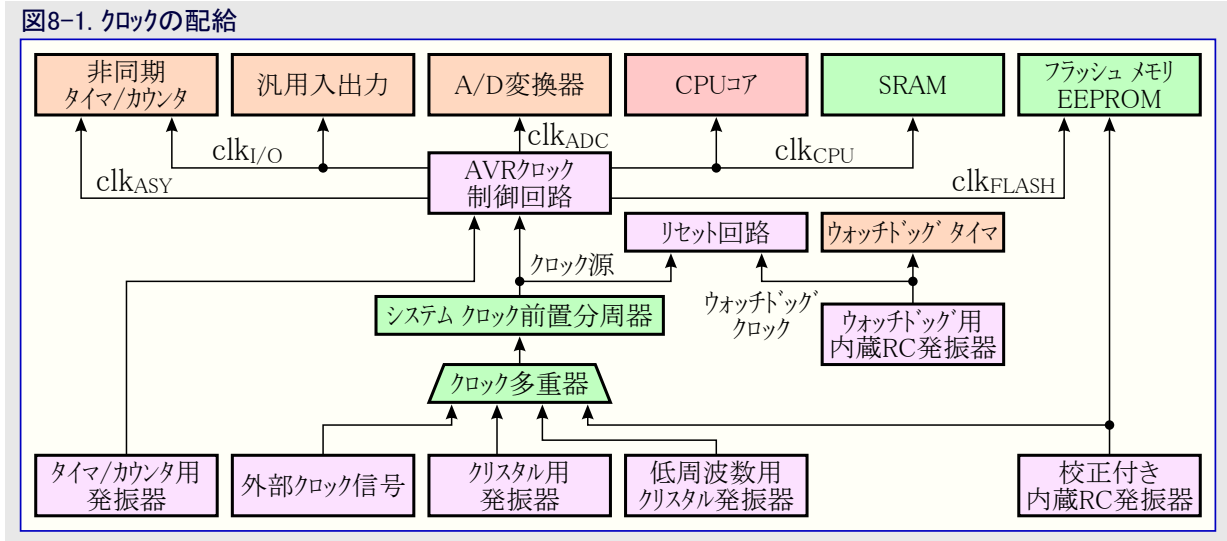
#### 7.6.6. 汎用I/Oレジスタ0 (General Purpose I/O Register 0) GPIOR0

ビット	7	6	5	4	3	2	1	0	
\$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	

## 8. システム クロックとクロック選択

### 8.1. クロック系統とその配給

図8-1はAVR内の主要なクロック系統とその配給を示します。全てのクロックが与えられた時間有効である必要はありません。消費電力低減のため、22頁の「電力管理と休止形態」で記述される各種休止形態の使用によって、使われない部分のクロックを停止することができます。クロック系統は以下で詳細に示されます。



#### 8.1.1. CPU クロック – clk<sub>CPU</sub>

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

#### 8.1.2. I/O クロック – clk<sub>I/O</sub>

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

#### 8.1.3. フラッシュ クロック – clk<sub>FLASH</sub>

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

#### 8.1.4. 非同期タイマ クロック – clk<sub>ASY</sub>

非同期タイマクロックは外部32kHzクロック用クリスタルから直接的にクロック駆動されるのを非同期タイマ/カウンタに許します。この専用クロック範囲はデバイスが休止形態作の時でも、このタイマ/カウンタの実時間計数器としての使用を許します。

#### 8.1.5. A/D変換クロック – clk<sub>ADC</sub>

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

## 8.2. クロック元

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

各クロック選択に対する各種選択は次項で得られます。CPUがパワーダウンまたはパワーセーブから起動するとき、選んだクロック元は命令実行開始前に安定な発振器動作を保証する起動時間に使われます。CPUがリセットから始まる時、これらは通常動作開始前に安定電圧へ達するのを電源に許す付加遅延です。ウォッチドッグ発振器はこの起動時間の実時間部のタイミングに使われます。各計時完了に使われるウォッチドッグ(WDT)発振器周期数は表8-2で示されます。198頁の「代表特性」で示されるように、ウォッチドッグ発振器の周波数は電圧に依存します。

表8-1. クロック種別選択

クロック種別	CKSEL3~0
外部水晶/セラミック発振子	1111~1000
外部低周波数水晶発振子	0111~0110
校正付き内蔵RC発振器	0010
外部クロック信号	0000
(予約)	00x1, 010x

注: 1=非プログラム、0=プログラム、x=0または1

表8-2. WDT発振器代表的計時完了値、周期数

VCC=3.0V	VCC=5.0V	周期数
4.3ms	4.1ms	4K (4096)
69ms	65ms	64K (65536)

## 8.3. 既定のクロック元

このデバイスはCKSEL=0010, SUT=10, CKDIV8=プログラム(0)で出荷されます。既定クロック元設定は最長起動時間で初期システムクロック8分周の内蔵RC発振器で、1MHzシステムクロックになります。この既定設定は全ての使用者が実装または並列書き込み器を使い、それらを希望したクロック元設定にできることを保証します。

## 8.4. クリスタル用発振器

XTAL1とXTAL2は図8-2で示されるように、チップ上の発振器としての使用に設定できる反転増幅器の各々、入力と出力です。クリスタル発振子またはセラミック振動子のどちらでも使えます。

C1とC2はクリスタル発振子とセラミック振動子の両方について常に等しくすべきです。このコンデンサの最適値は使うクリスタル発振子やセラミック振動子、浮遊容量の量、その環境の電磁雑音に依存します。クリスタル発振子使用に対するコンデンサ選択について初期の指針のいくつかは表8-3で与えられます。セラミック振動子については製造業者によって与えられたコンデンサ値が使われるべきです。

発振器は示された周波数範囲で最適化された4つの異なる動作ができます。この動作は、表8-3で示されたCKSEL3~1ヒューズによって選ばれます。

図8-2. クリスタル発振子接続図

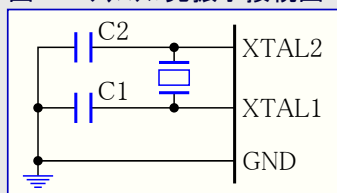


表8-3. クリスタル用発振器動作

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

注1: 本選択はクリスタル発振子ではなく、セラミック振動子でのみ使われるべきです。

CKSEL0ヒューズはSUT1,0ヒューズと共に、表8-4で示される起動遅延時間を選びます。

表8-4. クリスタル発振子/セラミック振動子用起動遅延時間選択表

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

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

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



## 8.5. 低周波数クリスタル用発振器

低周波数クリスタル用発振器は時計用32.768kHzクリスタルでの使用に最適化されています。クリスタル選択時、負荷容量とクリスタルの等価直列抵抗(ESR)が考慮されなければなりません。両値はクリスタル販売業者によって指定されます。ATmega325P/3250Pの発振器は非常に低い電力消費用に最適化されており、従って6.5pF,9pF,12.5pFでの推奨最大ESRに関してはクリスタル選択時に表8-5をご覧ください。

低周波数クリスタル用発振器は各TOSCピンで代表値6pFの内部負荷容量を提供します。各TOSCピンで必要とする外部容量(C)は次式を使って計算できます。

$$C = 2 \times CL - C_s$$

ここでCLはクリスタル製造業者によって指定された32.768kHz用の負荷容量で、Csは1つのTOSCピンに対する総浮遊容量です。

6pFより高い指定負荷容量(CL)のクリスタルは図8-2で記述したように、付加外部容量(コンデンサ)が必要です。

32.768kHzクリスタルに対する最適な負荷容量を見つけるには、クリスタルのデータシートを調べてください。

低周波数クリスタル発振器は表8-7で示されるように、CKSELヒューズを'0110'または'0111'に設定することによって選ばれなければなりません。起動時間は表8-6で示されるようにSUTヒューズによって決定されます。

表8-5. 時計用32.768kHzクリスタル用推奨最大ESR

クリスタル負荷容量 (CL: pF)	最大ESR (kΩ) (注1)
6.5	75
9.0	65
12.5	30

注1: 最大ESRは特性を基にした代表値です。

表8-6. 低周波数クリスタル発振器用起動遅延時間選択表

SUT1,0	リセットからの付加遅延時間 (VCC=5.0V)	推奨使用法
0 0	14×CK	高速上昇電源またはBOD許可
0 1	14×CK+4.1ms	低速上昇電源
1 0	14×CK+65ms	起動時の周波数安定
1 1		(予約)

表8-7. 低周波数クリスタル発振器用起動時間選択表

CKSEL3~0	パワーダウン、パワーセーブからの起動遅延時間	推奨使用法
0 1 1 0	1K×CK	(注1)
0 1 1 1	32K×CK	起動時の周波数安定

注1: これらの選択は起動時の周波数安定度が応用にとって重要でない場合だけ使われるべきです。

## 8.6. 校正付き内蔵RC発振器

既定による校正された内蔵RC発振器は概ね8.0MHzのクロックを供給します。電圧と温度に依存しますが、このクロックは使用者によって高精度な校正ができます。本デバイスはCKDIV8ヒューズがプログラム(0)で出荷されます。より多くの詳細については20頁の「システムクロック前置分周器」をご覧ください。

このクロックは表8-8で示されるようにCKSELヒューズのプログラミングによってシステムクロックとして選べます。選んだなら、外部部品なしで動作します。リセット中、ハードウェアが発振校正(OSCCAL)レジスタに予めプログラムされた校正値バイトを設定し、これによってRC発振器を自動的に校正します。この校正の精度は194頁の表26-1で工場校正として示されます。

ソフトウェアからOSCCALレジスタを変更することによって(発振校正レジスタ(OSCCAL)参照)、工場校正を使うよりも高い精度を得ることができます。この校正の精度は表26-1で使用者校正として示されます。

この発振器がチップ(システム)クロックとして使われるとき、ウォッチドッグ発振器は未だウォッチドッグタイマとリセット付加遅延タイマに使われます。予め設定された校正値のより多くの情報については174頁の「校正バイト」項をご覧ください。

この発振器が選ばれると、起動時間は表8-9で示されるようにSUTヒューズによって決定されます。XTAL1/TOSC1とXTAL2/TOSC2はタイマ/カウンタ用発振器ピンとして使えます(訳補: 共通性から本行追加)。

表8-8. 校正付き内蔵RC発振器動作種別 注: デバイスはこの選択で出荷されます。

CKSEL3~0	周波数範囲 (MHz)
0 0 1 0	7.3~8.1

注: 周波数範囲は暫定値で実際の値はTBDです。

注: この8MHz周波数がデバイス仕様(VCCに依存)を越える場合、内部周波数を8分周するためにCKDIV8ヒューズをプログラム(0)にできます。

表8-9. 校正付き内蔵RC発振器用起動遅延時間選択表

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

注1: デバイスはこの選択で出荷されます。

## 8.7. 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ためにXTAL1は図8-3.で示されるように駆動されるべきです。外部クロックでデバイスを走行するためにはCKSELヒューズが'0000'にプログラム(設定)されなければなりません。

このクロック元が選ばれると、起動時間は表8-11.で示されるようにSUTヒューズによって決定されます。

表8-10. 外部クロック信号動作

CKSEL3~0	周波数範囲
0 0 0 0	0~20MHz

図8-3. 外部クロック信号駆動接続図

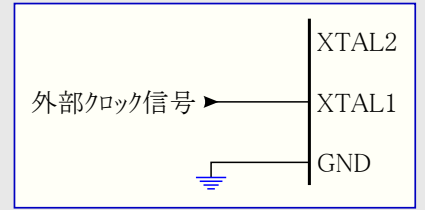


表8-11. 外部クロック信号駆動用起動遅延時間選択表

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

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

システムクロック前置分周器が安定な動作を保証しながら、内部クロック周波数の実行時変更の実現に使えることに注意してください。詳細については以下の「システムクロック前置分周器」を参照してください。

## 8.8. クロック出力緩衝部 (外部クロック出力)

CKOUTヒューズがプログラム(0)されると、システムクロックがCLKO(ピン)に出力されます。この動作はチップのクロックがシステム上の他の回路を駆動する時用です。このヒューズがプログラム(0)されると、I/Oピンの通常動作は無視され、このクロックはリセット中も出力されます。CLKOがクロック出力を扱うとき、校正付き内蔵RC発振器を含む何れのクロック元も選べます。システムクロック前置分周器が使われると、CKOUTヒューズがプログラム(0)された時の出力は分周したシステムクロックです。

## 8.9. タイマ/カウンタ用発振器

ATmega325P/3250Pは低周波数クリスタル用発振器とタイマ/カウンタ用発振器に対して同じクリスタル用発振器を使います。この発振器とクリスタルの必要条件については19頁の「低周波数クリスタル用発振器」をご覧ください。

ATmega325P/3250Pはタイマ/カウンタ用発振器(TOSC1とTOSC2)ピンとXTAL1,XTAL2ピンを共用します。タイマ/カウンタ用発振器使用時、システムクロックはこの発振器周波数の4倍を必要とします。これとピン共用のため、タイマ/カウンタ用発振器はシステムクロック元として校正付き内蔵RC発振器選択時にだけ使えます。

非同期状態レジスタ(ASSR)の外部クロック許可(EXCLK)ビットが論理1を書かれると、TOSC1に外部クロック元の印加が行えます。時計用32.768kHzクリスタルに代わる入力として外部クロックを選ぶ更なる記述については99頁の「タイマ/カウンタの非同期動作」をご覧ください。

## 8.10. システムクロック前置分周器

ATmega325P/3250Pのシステムクロックはクロック前置分周レジスタ(CLKPR)の設定によって分周できます。この特徴(機能)は必要とされる処理能力が低い時の消費電力削減に使えます。これは全クロック種別で使え、CPUと全同期周辺機能のクロック周波数に影響を及ぼします。clkCPU、clkFLASH、clkI/O、clkADCは表8-12.で示された値によって分周されます。

### 8.10.1. 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。

## 8.11. クロック関係用レジスタ

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

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

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

発振校正レジスタは発振器周波数の偏差処理を省くための内蔵発振器の調整に使われます。チップのリセット中、194頁の表26-1で指定されるように工場で校正された周波数を与える、予めプログラムされた値が本レジスタへ自動的に書かれます。応用ソフトウェアは発振器周波数を変更するために、このレジスタに書くことができます。この発振器は表26-1で指定される周波数に校正できます。この範囲外への校正は推奨されません。

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

CAL7ビットは発振器に対する操作範囲を決めます。このビットの(0)設定は低周波数範囲になり、(1)設定は高周波数範囲になります。この2つの周波数範囲は重複し、別の言葉では、OSCCAL=\$7F設定はOSCCAL=\$80設定より高い周波数になります。

CAL6～0ビットは選んだ範囲内の周波数調整に使われます。\$00設定はその範囲の最低周波数になり、\$7F設定はその範囲の最高周波数になります。

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

ビット (\$61)	7	6	5	4	3	2	1	0	
	CLKPCE	–	–	–	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	内容参照	内容参照	内容参照	内容参照	

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

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

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

これらのビットは選んだクロック元と内部システムクロック間の分周値を定義します。これらのビットは応用の必要条件に合わせた各種クロック周波数を実行時に書けます。分周値が使われると、分周器はMCUへの主クロックを分周し、全ての同期周辺機能の速度が減じられます。分周値は表8-12で与えられます。

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

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

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

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

表8-12. 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	2	4	8	16	32	64	128	256	(予約)						



## 9. 電力管理と休止形態

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

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

### 9.1. 休止形態種別

17頁の図8-1はATmega325P/3250Pの各種クロック系統とその配給を示します。この図は適切な休止形態を選ぶ助けになります。表9-1は各種休止形態、起動元、BOD禁止の可能性を示します。

表9-1. 各休止形態に於ける動作クロック範囲と復帰起動要因

休止種別	動作クロック範囲					動作発振器		復帰起動要因 (割り込み)						ソフトウェア BOD 禁止
	clkCPU	clkFLASH	clkI/O	clkADC	clkASY	主クロック 供給元	タイマ用 発振器	INT0 ピン変化 割り込み	USI 開始条件 検出	タイマ/カウンタ 2	SPM EEPROM 操作可	A/D 変換 完了	その他 I/O	
アイドル			○	○	○	○	②	○	○	○	○	○	○	
A/D変換雑音低減				○	○	○	②	③	○	②	○	○		
パワーダウン								③	○					○
パワーセーブ					○		○	③	○	○				○
スタンバイ(注1)						○		③	○					○

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

② タイマ/カウンタ2が非同期動作で走行する場合です。

③ INT0についてはレベル割り込みのみです。

休止形態の何れかへ移行するには**休止形態制御レジスタ(SMCR)**の**休止許可(SE)ビット**が論理1を書かれ、**SLEEP**命令が実行されなければなりません。SMCRの**休止種別選択(SM2~0)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25頁の表9-2をご覧ください。

MCUが休止形態中に許可した割り込みが起ると、MCUは起動します。その時にMCUは起動時間に加えて4周期停止され、割り込みルーチンを実行し、そして**SLEEP**命令の次の命令から実行を再開します。デバイスが休止から起動するとき、レジスタファイルとSRAMの内容は変えられません。休止形態中にリセットが起ると、MCUは起動し、リセットベクタから実行します。

### 9.2. 低電圧検出器(BOD)禁止

低電圧検出器(BOD)が173頁の表25-3のBODLEVELヒューズによって許可されていると、BODは休止期間中に電源電圧を活発に監視します。節電のため、休止形態のいくつかに対してソフトウェアによってBODを禁止することが可能です。表9-1をご覧ください。この休止形態電力消費はBODがヒューズによって全面的に禁止される時と同じ水準になるでしょう。BODがソフトウェアで禁止される場合、BOD機能は休止形態移行後、直ちにOFFされます。休止からの起動復帰で、BODは再び自動的に許可されます。これは休止期間中にVCCレベルが落ちた場合の安全な動作を保証します。

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

BOD禁止はMCU制御レジスタ(MCUCR)のビット6、**BOD休止(BODS)ビット**によって制御されます。25頁の「**MCU制御レジスタ(MCUCR)**」をご覧ください。このビットへの1書き込みは関連する休止形態でのBODをOFFにし、一方このビットの0はBOD活動(有効)を保ちます。既定設定はBOD活動維持、換言するとBODSは0に設定です。

BODSビットへの書き込みは許可ビットと時間制限手順によって制御されます。25頁の「**MCU制御レジスタ(MCUCR)**」をご覧ください。

### 9.3. アイドル動作

**休止種別選択(SM2~0)ビット**が'000'を書かれるとき、**SLEEP**命令はMCUをアイドル動作へ移行させ、CPUを停止しますが、SPI、USART、アナログ比較器、A/D変換器、多用途直列インターフェース(USI)、タイマ/カウンタ、ウォッチドッグ、割り込み機構の継続動作を許します。この休止形態は基本的にclkCPUとclkFLASHを停止する一方、他のクロックに走行を許します。

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

## 9.4. A/D変換雑音低減動作

休止種別選択(SM2~0)ビットが'001'を書かれるとき、**SLEEP**命令はMCUをA/D変換雑音低減動作へ移行させて、CPUを停止しますが、A/D変換器、外部割り込み、USIの開始条件検出、タイマ/カウンタ2、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkI/O、clkCPU、clkFLASHを停止する一方、他のクロックに走行を許します。

これはA/D変換に対する雑音環境を改善し、より高い分解能の測定を可能にします。A/D変換器が許可されている場合、この動作に移行すると、変換が自動的に始まります。A/D変換完了割り込みからの他、外部リセット、ウォッチドッグリセット、低電圧検出(BOD)リセット、USIの開始条件検出割り込み、タイマ/カウンタ2の割り込み、SPM/EEPROM操作割り込み、INT0の外部レベル割り込み、ピン変化割り込みだけが、A/D変換雑音低減動作からMCUを起動できます。

## 9.5. パワーダウン動作

SM2~0ビットが'010'を書かれるとき、**SLEEP**命令はMCUをパワーダウン動作へ移行させます。この動作では外部発振器が停止される一方、外部割り込み、多用途直列インターフェース(USI)の開始条件検出、ウォッチドッグ機能は(許可されていれば)継続して動作します。外部リセット、ウォッチドッグリセット、低電圧検出(BOD)リセット、USI開始条件検出割り込み、INT0の外部レベル割り込み、ピン変化割り込みだけがMCUを起動できます。この休止形態は基本的に生成した全クロックを停止し、非同期部の動作だけを許します。

レベルで起動した割り込みがパワーダウン動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。詳細については36頁の「外部割り込み」を参照してください。

パワーダウン動作から起動するとき、起動条件が起きてから起動の効果が現れるまで遅延があります。これは停止されてしまっている後の再始動と安定になることをクロックに許します。この起動(遅延)時間は18頁の「クロック元」で記述されるように、リセット遅延時間を定義するのと同じCKSELヒューズによって定義されます。

## 9.6. パワーセーブ動作

SM2~0ビットが'011'を書かれると、**SLEEP**命令はMCUをパワーセーブ動作へ移行させます。この動作は(次の)1つの例外を除いてパワーダウン動作と同じです。

タイマ/カウンタ2が許可される場合、それらは休止中でも走行(動作)を維持します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、タイマ/カウンタ2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ2溢れ割り込み許可(TOIE2)ビットまたは比較2割り込み許可(OCIE2A)ビットが設定(1)されるなら、デバイス是对応するどちらの割り込みからでも起動できます。

タイマ/カウンタ2が走行(動作)しないなら、パワーダウン動作をパワーセーブ動作の代わりにすることが推奨されます。

タイマ/カウンタ2はパワーセーブ動作で同期と非同期の両方でクロック駆動できます。タイマ/カウンタ2が非同期クロックを使わない場合、休止中、タイマ/カウンタ用発振器は停止されます。タイマ/カウンタ2が同期クロックを使わない場合、休止中、そのクロック元は停止されます。例えばパワーセーブ動作で同期クロックが走行しても、このクロックはタイマ/カウンタ2に対してだけ利用可能なことに注意してください。

## 9.7. スタンバイ動作

外部クリスタル発振子/セラミック振動子クロック種別が選ばれ、SM2~0ビットが'110'のとき、**SLEEP**命令はMCUをスタンバイ動作へ移行させます。この動作は(外部クリスタル用)発振器が走行(動作)を保たれる例外を除いてパワーダウン動作と同じです。デバイスはスタンバイ動作から6クロック周期で起動します。

## 9.8. 電力削減

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

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



## 9.9. 消費電力の最小化

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

### 9.9.1. A/D変換器 (ADC)

許可したなら、A/D変換器は全休止形態で許可されます。節電するため、休止形態の何れかへ移行する前にA/D変換器は禁止されるべきです。A/D変換器がOFFそして再びONに切り替えられると、次の(最初の)変換は延長された(初回)変換になります。A/D変換器操作の詳細については138頁の「[A/D変換器](#)」を参照してください。

### 9.9.2. アナログ比較器

アイドル動作へ移行するとき、アナログ比較器は使われないなら、禁止されるべきです。A/D変換雑音削減動作へ移行するとき、アナログ比較器は禁止されるべきです。その他の休止形態でのアナログ比較器は自動的に禁止されます。しかしアナログ比較器が入力として内部基準電圧を使う設定の場合、全休止形態でアナログ比較器は禁止されるべきです。さもなければ内部基準電圧は休止形態と無関係に許可されます。アナログ比較器設定法の詳細については136頁の「[アナログ比較器](#)」を参照してください。

### 9.9.3. 低電圧検出器 (BOD)

低電圧検出器(BOD)が応用で必要とされないなら、この単位部はOFFにされるべきです。低電圧検出器がBODLEVELヒューズによって許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。低電圧検出器(BOD)設定法の詳細については28頁の「[低電圧検出 \(BOD\)](#)」を参照してください。

### 9.9.4. 内部基準電圧

内部基準電圧は低電圧検出器(BOD)、アナログ比較器、A/D変換器によって必要とされる時に許可されます。これら単位部が上の項目で記述されたように禁止されると、内部基準電圧は禁止され、電力を消費しません。再び許可する場合、この出力が使われる前に使用者は基準電圧へ起動(安定時間)を与えなければなりません。基準電圧が休止形態でON保持される場合、この出力は直ちに使えます。起動時間の詳細については29頁の「[内部基準電圧](#)」を参照してください。

### 9.9.5. ウォッチドッグ タイマ

ウォッチドッグ タイマが応用で必要とされないなら、この単位部はOFFにされるべきです。ウォッチドッグ タイマが許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。ウォッチドッグ タイマ設定法の詳細については29頁の「[ウォッチドッグ タイマ](#)」を参照してください。

### 9.9.6. ポートピン

休止形態へ移行するとき、全てのポートピンは最小電力使用に設定されるべきです。最も重要なことはその時にピンが抵抗性負荷を駆動しないのを保証することです。I/Oクロック(clk<sub>I/O</sub>)とA/D変換クロック(clk<sub>ADC</sub>)の両方が停止される休止形態ではデバイスの入力緩衝部が禁止されます。これは必要とされない時に入力論理回路によって電力が消費されないことを保証します。いくつかの場合で入力論理回路は起動条件を検出するために必要とされ、その時は許可されます。どのピンが許可されるかの詳細については41頁の「[デジタル入力許可と休止形態](#)」を参照してください。入力緩衝部が許可され、入力信号が浮いている状態のままか、またはアナログ信号電圧がVCC/2付近の場合、入力緩衝部は過大な電力を消費するでしょう。

アナログ入力ピンに対するデジタル入力緩衝部は常に禁止されるべきです。入力ピンでのVCC/2付近のアナログ信号入力は活動動作でも重要な電流を引き起こし得ます。デジタル入力緩衝部はデジタル入力禁止レジスタ(DIDR0とDIDR1)の書き込みによって禁止できます。詳細については137頁の「[デジタル入力禁止レジスタ1\(DIDR1\)](#)」と147頁の「[デジタル入力禁止レジスタ0\(DIDR0\)](#)」を参照してください。

### 9.9.7. JTAGインターフェースと内蔵デバッグ機能 (OCD)

内蔵デバッグ機能がOCDENヒューズによって許可され、チップがパワーダウンまたはパワーセーブの休止形態へ移行すると、主クロック元は許可に留まります。これらの休止形態では、これが総消費電流にとって重要な一因になります。これを避けるには3通りの方法があります。

- OCDENヒューズを禁止 (非プログラム(1)設定)
- JTAGENヒューズを禁止 (非プログラム(1)設定)
- MCU制御レジスタ(MCUCR)のJTAG禁止(JTD)ビットへの1書き込み

JTAGインターフェースが許可され、JTAG TAP制御器がデータを移動していないと、TDOピンは浮き状態のままにされます。TDOピンに接続したハードウェアが論理レベルをプルアップしないなら、消費電力が増加するでしょう。走査チェーン内の次のデバイスのTDIピンがこの問題を避けるプルアップを含むことに注意してください。MCUCRのJTDビットに1を書くか、またはJTAGENヒューズを非プログラムのままにすることがJTAGインターフェースを禁止します。

## 9.10. 電力管理用レジスタ

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

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)	—	—	—	—	SM2	SM1	SM0	SE	SMCR
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3~1 – SM2~0 : 休止種別選択 (Sleep Mode Select Bit 2, 1 and 0)

これらのビットは表9-2.で示される利用可能な5つの休止形態の1つを選びます。

表9-2. 休止形態種別選択

SM2	SM1	SM0	休止形態種別
0	0	0	アイドル動作
0	0	1	A/D変換雑音低減動作
0	1	0	パワーダウン動作
0	1	1	パワーセーブ動作
1	0	0	(予約)
1	0	1	(予約)
1	1	0	スタンバイ動作 (注)
1	1	1	(予約)

注: スタンバイ動作は外部クリスタル発振子またはセラミック振動子でだけ利用できます。

#### • ビット0 – SE : 休止許可 (Sleep Enable)

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

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	BODS	BODSE	PUD	—	—	IVSEL	IVCE	MCUCR
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6 – BODS : BOD休止 (BOD Sleep)

休止間にBODをOFFにするために、BODSビットは論理1を書かれなければなりません。22頁の表9-1.をご覧ください。BODSビットの書き込みはMCUCRのBOD休止許可(BODSE)ビットと時間制限手順によって制御されます。関連する休止形態でBODを禁止するには最初にBODSとBODSEの両方が1に設定されなければなりません。その後BODSビットを設定するためにBODSが1に設定され、そして4クロック周期内にBODSEが0に設定されなければなりません。

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

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

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

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

ビット (\$64)	7	6	5	4	3	2	1	0	PRR
	–	–	–	–	PRTIM1	PRSPI	PRUSART0	PRADC	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7～4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

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

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

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

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

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

このビットへの1書き込みはその部分へのクロック停止によってUSARTを停止します。USARTの再起動時、USARTは正しい動作を保証するために再初期化されるべきです。

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

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

**注:** アナログ比較器は137頁の「アナログ比較器制御/状態レジスタ(ACSR)」のアナログ比較器禁止(ACD)ビットを使って禁止されます。

## 10. システム制御とリセット

### 10.1. 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JMP(絶対分岐)命令でしょう。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが配置できます。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10-1の回路構成図はリセット論理回路を示します。195頁の「システムとリセットの特性」はリセット回路の電気的特性を定義します。

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

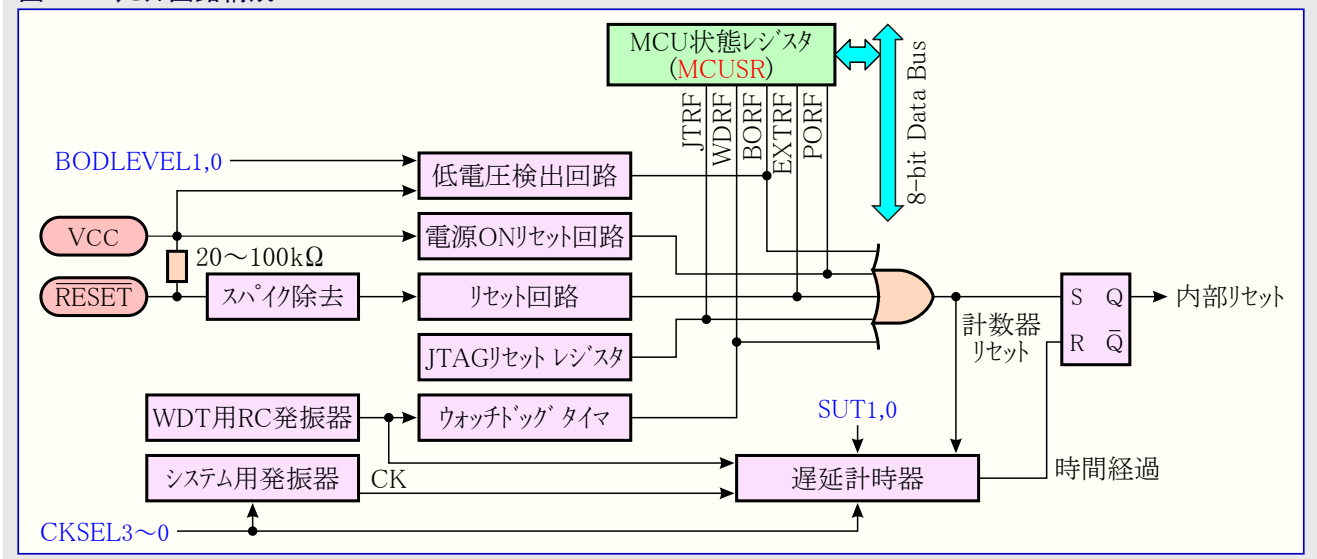
全てのリセット元が無効にされてしまった後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電源に許します。遅延タイマの遅延時間はCKSELヒューズを通して使用者によって定義されます。この遅延時間についての各種選択は18頁の「クロック元」で示されます。

### 10.2. リセット元

ATmega325P/3250Pには次の5つのリセット元があります。

- **電源ONリセット** ..... 供給電圧が電源ONリセット閾値電圧(V<sub>POT</sub>)以下でMCUがリセットされます。
- **外部リセット** ..... RESETピンが最小パルス幅以上Lowレベルに保たれると、MCUがリセットされます。
- **ウォッチドッグリセット** ... ウォッチドッグが許可され、ウォッチドッグタイマが終了すると、MCUがリセットされます。
- **低電圧リセット** ..... 低電圧検出が許可され、供給電圧(V<sub>CC</sub>)が低電圧検出電圧(V<sub>BOT</sub>)以下でMCUがリセットされます。
- **JTAG AVR リセット** .. JTAGシステムの走査チェーンの1つとしてリセットレジスタ内に論理1がある間中、MCUがリセットされます。詳細については152頁の「IEEE 1149.1 (JTAG) 境界走査」を参照してください。

図10-1. リセット回路構成



### 10.2.1. 電源ONリセット

電源ONリセット(POR)パルスはチップ上の検出回路によって生成されます。検出電圧は195頁の「システムとリセットの特性」で定義されます。POR信号はVCCが検出電圧以下の時は必ず活性(有効)にされます。POR回路は供給電圧異常検出は勿論、始動リセットの起動にも使えます。

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

図10-2. 内蔵電源ONリセット (RESETはVCCに接続)

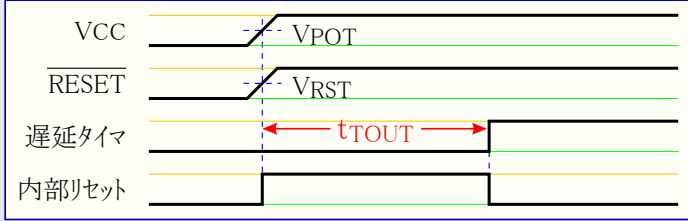
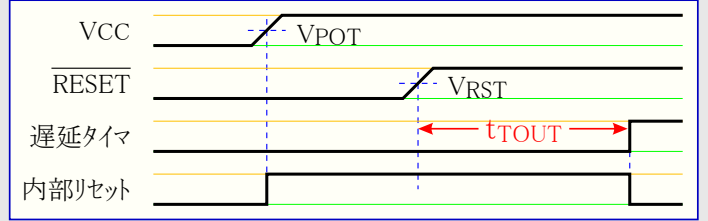


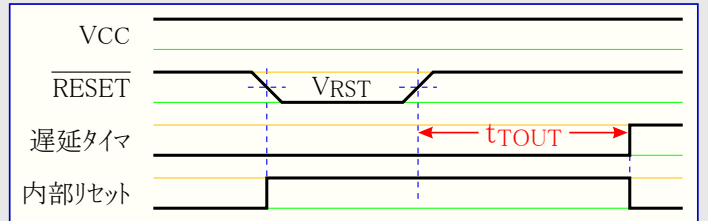
図10-3. 外部RESET信号による延長電源ONリセット



### 10.2.2. 外部リセット

外部リセットはRESETピンのLowレベルによって生成されます。クロックが動いていなくても、最小パルス幅(195頁の「システムとリセットの特性」参照)以上のリセットパルスはリセットを生成します。短すぎるパルスはリセット生成が保証されません。印加された信号の上昇がリセット閾値電圧( $V_{RST}$ )に達すると(遅延タイマを起動し)、遅延タイマは遅延時間( $t_{TOUT}$ )経過後にMCUを始動します。

図10-4. 動作中の外部リセット



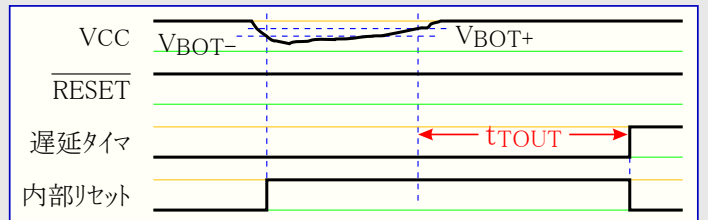
### 10.2.3. 低電圧(ブラウンアウト)検出リセット

ATmega325P/3250Pには固定化された起動(検出)電圧と比較することで動作中のVCCを監視するチップ上の低電圧検出(BOD)回路があります。BODの起動電圧はBODLEVELヒューズによって選べます。この起動電圧はスパイク対策BODを保証するためにヒステリシスを持ちます。検出電圧のヒステリシスは $V_{BOT+}=V_{BOT}+V_{HYST}/2$ 、 $V_{BOT-}=V_{BOT}-V_{HYST}/2$ と解釈されるべきです。

BODが許可され、VCCが起動電圧以下に下降すると(図10-5.の $V_{BOT-}$ )、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(図10-5.の $V_{BOT+}$ )、(遅延タイマが起動され)遅延タイマは遅延時間( $t_{TOUT}$ )経過後にMCUを始動します。

BOD回路は電圧が195頁の「システムとリセットの特性」で与えられる $t_{BOD}$ 時間より長く起動電圧以下に留まる場合のみ、VCCでの低下を検出します。

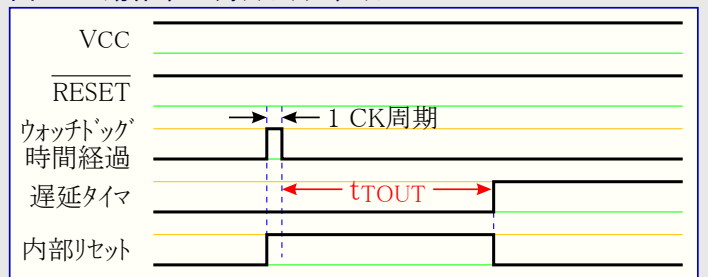
図10-5. 動作中の低電圧検出リセット



### 10.2.4. ウォッチドッグリセット

ウォッチドッグ時間経過時、(内部的に)1 CK周期幅の短いリセットパルスを生成します。本パルスの下降端で遅延タイマは遅延時間( $t_{TOUT}$ )の計時を始めます。ウォッチドッグタイマ操作の詳細については29頁を参照してください。

図10-6. 動作中のウォッチドッグリセット





## 10.3. 内部基準電圧

ATmega325P/3250P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。

### 10.3.1. 基準電圧許可信号と起動時間

この基準電圧には使われるべき方法に影響を及ぼすかもしれない起動時間があります。この起動時間は195頁の「システムとリセットの特性」で与えられます。節電のために、この基準電圧は常にONではありません。この基準電圧は次の状態中ONです。

1. 低電圧検出リセット許可時 (BODLEVELヒューズのプログラム(0)により)
2. アナログ比較器基準電圧接続時 (アナログ比較器制御/状態レジスタ(ACSR)の基準電圧入力選択(ACBG)=1)
3. A/D変換部動作許可時 (A/D変換制御/状態レジスタ(ADCSRA)のA/D動作許可(ADEN)=1)

従って低電圧検出(BOD)が許可されていないと、ACBGの設定(=1)またはA/D変換部許可(ADEN=1)後、使用者はアナログ比較器またはA/D変換器出力が使われる前に基準電圧へ起動時間を与えなければなりません。パワーダウン動作での消費電力を減らすため、使用者はパワーダウン動作へ移行する前に基準電圧がOFFされるのを保証することで上の3つの状態を避けられます。

## 10.4. ウォッチドッグ タイマ

ウォッチドッグ タイマは1MHzで動く独立したチップ上の発振器からクロック駆動されます。これはVCC=5Vでの代表値です。他のVCC電圧での代表値については「特性データ」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10-2.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にもリセットされます。このリセット周期を決めるのに8つの異なるクロック周期を選べます。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、ATmega325P/3250Pはリセットしてリセットベクタから実行します。ウォッチドッグ リセットの詳細タイミングについては28頁を参照してください。

予期せぬウォッチドッグ 禁止や予期せぬ計時周期変更を防ぐために、2つの異なる安全基準が表10-1.で示されるWDTONヒューズによって選ばれます。詳細については次の「ウォッチドッグ タイマ設定変更の時間制限手順」を参照してください。

図10-7. ウォッチドッグ タイマ構成図

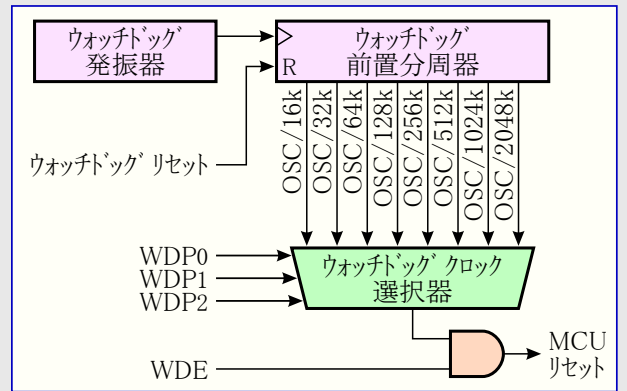


表10-1. WDTONヒューズ設定によるウォッチドッグ 機能設定

WDTON ヒューズ	安全基準	WDT初期状態	WDT禁止方法	計時時間変更方法
非プログラム(1)	1	禁止	時間制限	時間制限
プログラム(0)	2	許可	なし(常時許可)	時間制限

### 10.4.1. ウォッチドッグ タイマ設定変更の時間制限手順

ウォッチドッグ 設定変更手順は安全基準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。

#### 10.4.2. 安全基準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を書くことにより許可できます。ウォッチドッグ 計時完了周期を変更または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 タイマを禁止や、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. 同じ操作(命令)でウォッチドッグ 変更許可(WDCE)とWDEに論理1を書きます。WDEビットの直前の値に拘らず、論理1がWDEに書かれなければなりません。
2. 次からの4クロック周期内に同じ操作(命令)で欲したWDEとウォッチドッグ タイマ前置分周選択(WDP2~0)ビットを書きますが、WDCEビットは解除(0)されてです。

#### 10.4.3. 安全基準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として読みます。時間制限手順はウォッチドッグ 計時完了周期を変更する時に必要とされます。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. 同じ操作(命令)でWDCEとWDEに論理1を書きます。WDEビットが常に設定(1)されていても、WDEは時間制限手順を開始するために1を書かれなければなりません。
2. 次からの4クロック周期内に同じ操作(命令)で欲したWDP2~0ビットを書きますが、WDCEビットは解除(0)されてです。WDEビットに書かれた値は無関係です。

## 10.5. リセット関係用レジスタ

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

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

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

#### • ビット4 – JTRF : JTAG リセット フラグ (JTAG Reset Flag)

このビットはリセットがJTAG命令AVR\_RESETで選ばれたJTAG リセット レジスタ内の論理1によって引き起こされると設定(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。

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

ビット	7	6	5	4	3	2	1	0	
(\$60)	–	–	–	WDCE	WDE	WDP2	WDP1	WDP0	WDTCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7～5 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

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

ウォッチドッグ許可(WDE)ビットが論理0を書かれるとき、このビットは設定(1)されなければなりません。さもなければ、ウォッチドッグは禁止されません。一旦1を書かれると、ハードウェアが4クロック周期後、このビットを解除(0)します。ウォッチドッグ禁止手順についてはWDEビットの記述を参照してください。このビットは前置分周選択ビットを変更する時も設定(1)されなければなりません。29頁の「ウォッチドッグ タイマ設定変更の時間制限手順」をご覧ください。

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

ウォッチドッグ許可(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を書かれるとウォッチドッグ タイマ機能が禁止されます。WDEはウォッチドッグ変更許可(WDCE)ビットが論理レベル1を持つ場合だけ解除(0)できます。許可されているウォッチドッグ 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. 同じ操作(命令)内で、ウォッチドッグ変更許可(WDCE)ビットとウォッチドッグ許可(WDE)ビットに論理1を書きます。例え禁止操作を始める前のWDEが1に設定されていても、論理1がWDEに書かれなければなりません。
2. 次からの4クロック周期内で、論理0をWDEに書きます。これがウォッチドッグを禁止します。

安全基準2では上記の手順でもウォッチドッグ タイマを禁止することができません。29頁の「ウォッチドッグ タイマ設定変更の時間制限手順」をご覧ください。

## •ビット2～0 – WDP2～0 : ウォッチドッグ タイマ前置分周選択 (Watchdog Timer Prescaler 2,1 and 0)

このWDP2～0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 タイマの前置分周を決めます。異なる前置分周値と対応する計時完了周期は表10-2.に示されます。

表10-2. ウォッチドッグ前置分周選択

WDP2	WDP1	WDP0	WDT発振周期数	代表的な計時完了周期	
				VCC=3.0V	VCC=5.0V
0	0	0	16k (16,384)	17.1ms	16.3ms
0	0	1	32k (32,768)	34.3ms	32.5ms
0	1	0	64k (65,536)	68.5ms	65.0ms
0	1	1	128k (131,072)	0.14s	0.13s
1	0	0	256k (262,144)	0.27s	0.26s
1	0	1	512k (524,288)	0.55s	0.52s
1	1	0	1024k (1,048,576)	1.1s	1.0s
1	1	1	2048k (2,097,152)	2.2s	2.1s

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

### アセンブリ言語プログラム例

```

WDT_OFF:      WDR                      ;ウォッチドッグ タイマ リセット
              IN      R16, WDTCR        ;現WDTCR値を取得
              ORI     R16, (1<<WDCE) | (1<<WDE) ;WDCEとWDE論理1値を設定
              OUT     WDTCR, R16        ;WDCEとWDEに論理1書き込み
              LDI     R16, (0<<WDE)     ;WDE論理0値を取得
              OUT     WDTCR, R16        ;ウォッチドッグ禁止
              RET                       ;呼び出し元へ復帰
    
```

### C言語プログラム例

```

void WDT_off(void)
{
    __watchdog_reset();                /* ウォッチドッグ タイマ リセット */
    WDTCR |= (1<<WDCE) | (1<<WDE);    /* WDCEとWDEに論理1書き込み */
    WDTCR = 0x00;                     /* ウォッチドッグ禁止 */
}
    
```

**注:** 6頁の「コード例について」をご覧ください。

## 11. 割り込み

### 11.1. 概要

この項はATmega325P/3250Pによって実行される割り込み操作の詳細を記述します。AVR割り込み操作の一般説明については10頁の「リセットと割り込みの扱い」を参照してください。

### 11.2. ATmega325P/3250Pの割り込みベクタ

表11-1. リセットと割り込みのベクタ

ベクタ番号	プログラム アドレス (注2)	発生元	備考
1	\$0000 (注1)	リセット	電源ON, WDT, BOD等の各種リセット
2	\$0002	INT0	外部割り込み要求0
3	\$0004	PCINT0	ピン変化0群割り込み要求 (PCI0)
4	\$0006	PCINT1	ピン変化1群割り込み要求 (PCI1)
5	\$0008	TIMER2_COMP	タイマ/カウンタ2比較一致
6	\$000A	TIMER2_OVF	タイマ/カウンタ2溢れ
7	\$000C	TIMER1_CAPT	タイマ/カウンタ1捕獲発生
8	\$000E	TIMER1_COMPA	タイマ/カウンタ1比較A一致
9	\$0010	TIMER1_COMPB	タイマ/カウンタ1比較B一致
10	\$0012	TIMER1_OVF	タイマ/カウンタ1溢れ
11	\$0014	TIMER0_COMP	タイマ/カウンタ0比較一致
12	\$0016	TIMER0_OVF	タイマ/カウンタ0溢れ
13	\$0018	SPI_STC	SPI 転送完了
14	\$001A	USART0_RX	USART 受信完了
15	\$001C	USART0_UDRE	USART 送信緩衝部空き
16	\$001E	USART0_TX	USART 送信完了
17	\$0020	USI_START	USI 開始条件検出
18	\$0022	USI_OVERFLOW	USI 計数器溢れ
19	\$0024	ANALOG_COMP	アナログ比較器出力遷移
20	\$0026	ADC	A/D変換完了
21	\$0028	EE_READY	EEPROM 操作可
22	\$002A	SPM_READY	SPM命令操作可
23	\$002C (注3)	NOT_USED	(予約)
24	\$002E (注3)	PCINT2	ピン変化2群割り込み要求 (PCI2)
25	\$0030 (注3)	PCINT3	ピン変化3群割り込み要求 (PCI3)

**注1:** 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。163頁の「ブート ローダ 支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

**注2:** 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3:** これら(NOT\_USED, PCINT2, PCINT3)はATmega3250Pにだけ存在します。

表11-2. 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ベクタの配置を示します。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置けます。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。

表11-2. リセットと割り込みベクタの配置

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	\$0002
	1	\$0000	ブート領域先頭アドレス+\$0002
プログラム(0)	0	ブート領域先頭アドレス	\$0002
	1	ブート領域先頭アドレス	ブート領域先頭アドレス+\$0002

**注:** ブート領域先頭アドレスは170頁の表24-6.で示されます。



ATmega325P/3250Pでの最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000		JMP RESET	;各種リセット
\$0002		JMP INTO	;外部割り込み要求0
\$0004		JMP PCINT0	;ピン変化0群割り込み要求
\$0006		JMP PCINT1	;ピン変化1群割り込み要求
\$0008		JMP TIMER2_COMP	;タイマ/カウンタ2比較一致
\$000A		JMP TIMER2_OVF	;タイマ/カウンタ2溢れ
\$000C		JMP TIMER1_CAPT	;タイマ/カウンタ1捕獲発生
\$000E		JMP TIMER1_COMPA	;タイマ/カウンタ1比較A一致
\$0010		JMP TIMER1_COMPB	;タイマ/カウンタ1比較B一致
\$0012		JMP TIMER1_OVF	;タイマ/カウンタ1溢れ
\$0014		JMP TIERMO_COMP	;タイマ/カウンタ0比較一致
\$0016		JMP TIERMO_OVF	;タイマ/カウンタ0溢れ
\$0018		JMP SPI_STC	;SPI転送完了
\$001A		JMP USART0_RX	;USART 受信完了
\$001C		JMP USART0_UDRE	;USART 送信緩衝部空
\$001E		JMP USART0_TX	;USART 送信完了
\$0020		JMP USI_START	;USI 開始条件検出
\$0022		JMP USI_OVERFLOW	;USI 計数器溢れ
\$0024		JMP ANALOG_COMP	;アナログ比較器出力遷移
\$0026		JMP ADC	;ADC変換完了
\$0028		JMP EE_READY	;EEPROM操作可
\$002A		JMP SPM_READY	;SPM命令操作可
\$002C		JMP PC	;未使用(予約) (ATmega3250P時のみ)
\$002E		JMP PCINT2	;ピン変化2群割り込み要求 (ATmega3250P時のみ)
\$0030		JMP PCINT3	;ピン変化3群割り込み要求 (ATmega3250P時のみ)
;			
\$0032	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$0033		OUT SPH, R16	;スタック ポインタ上位を初期化
\$0034		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0035		OUT SPL, R16	;スタック ポインタ下位を初期化
}			
;以下、I/O初期化など			

BOOTRSTヒューズが非プログラム(1)、ブート領域容量が4Kバイトに設定され、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)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (応用プログラム開始)
\$0001		OUT SPH, R16	;スタック ポインタ上位を初期化
\$0002		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0003		OUT SPL, R16	;スタック ポインタ下位を初期化
}			
;以下、I/O初期化など			
	.ORG	\$3802	;ブート プログラム領域が4Kバイトの場合
\$3802		JMP INTO	;外部割り込み要求0
\$3804		JMP PCINT0	;ピン変化0群割り込み要求
}			
\$382A		JMP SPM_READY	;SPM命令操作可
}			
;(これ以降はATmega3250P時のみ)			
\$3830		JMP PCINT3	;ピン変化3群割り込み要求

BOOTRSTヒューズがプログラム(0)、ブート領域容量が4Kバイトに設定される時の最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		. ORG    \$0002	; 割り込みベクタ先頭
\$0002		JMP    INT0	; 外部割り込み要求0
\$0004		JMP    PCINT0	; ピン変化0群割り込み要求
		}	
\$002A		JMP    SPM_READY	; SPM命令操作可
		}	; (これ以降\$0030まではATmega3250P時のみ)
\$0030		JMP    PCINT3	; ピン変化3群割り込み要求
		}	; 以下、プログラムなど
		. ORG    \$3800	; ブート プログラム領域が4Kバイトの場合
\$3800	RESET:	LDI    R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$3801		OUT    SPH, R16	; スタック ポインタ上位を初期化
\$3802		LDI    R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$3803		OUT    SPL, R16	; スタック ポインタ下位を初期化
		}	; 以下、I/O初期化など

BOOTRSTヒューズがプログラム(0)、ブート領域容量が4Kバイトに設定され、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)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		. ORG    \$3800	; ブート プログラム領域が4Kバイトの場合
\$3800		JMP    RESET	; 各種リセット (BOOTRSTヒューズ=0)
\$3802		JMP    INT0	; 外部割り込み要求0
\$3804		JMP    PCINT0	; ピン変化0群割り込み要求
		}	
\$382A		JMP    SPM_READY	; SPM命令操作可
		}	; (これ以降\$3830まではATmega3250P時のみ)
\$3830		JMP    PCINT3	; ピン変化3群割り込み要求
;			
\$382C/\$3832	RESET:	LDI    R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$382D/\$3833		OUT    SPH, R16	; スタック ポインタ上位を初期化
\$382E/\$3834		LDI    R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$382F/\$3835		OUT    SPL, R16	; スタック ポインタ下位を初期化
		}	; 以下、I/O初期化など

### 11.2.1. 応用領域とブート領域間の割り込みベクタ移動

MCU制御レジスタ(MCUCR)は割り込みベクタ表の配置を制御します。

## 11.3. ベクタ移動用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	BODS	BODSE	PUD	—	—	IVSEL	IVCE	MCUCR
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1 – IVSEL : 割り込みベクタ選択 (Interrupt Vector Select)

IVSEL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先頭に配置されます。このビットが設定(1)されると、割り込みベクタはフラッシュメモリのブートローダ領域の始まりへ移動されます。ブートフラッシュ領域先頭の実際のアドレスはBOOTSZヒューズによって決定されます。詳細については163頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」章を参照してください。割り込みベクタ表の予期せぬ変更を防ぐため、IVSELビットの変更は特別な書き込み手順に従わなければなりません。

1. 割り込みベクタ変更許可(IVCE)ビットに1を書きます。
2. 4周期内に、欲した値をIVSELに書き、同時に0をIVCEへ書きます。

この手順が実行される間、割り込みは自動的に禁止されます。割り込みはIVCEが設定(1)される周期で禁止され、後続のIVSELに書く命令の後まで禁止されたままです。IVSELが書かれなければ、割り込みは4周期間禁止されたままです。ステータスレジスタ(SREG)の全割り込み許可(I)ビットはこの自動禁止によって影響されません。

**注:** 割り込みベクタがブートローダ領域に配置され、BLB02ブート施錠ビットがプログラム(0)されると、応用領域から実行する間中、割り込みが禁止されます。割り込みベクタが応用領域に配置され、BLB12ブート施錠ビットがプログラム(0)されると、ブートローダ領域から実行する間中、割り込みが禁止されます。ブート施錠ビットの詳細については163頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」を参照してください。

#### • ビット0 – IVCE : 割り込みベクタ変更許可 (Interrupt Vector Change Enable)

IVCEビットは割り込みベクタ選択(IVSEL)ビットの変更を許可するために論理1を書かれなければなりません。IVCEはIVSELが書かれる時、またはIVCEが書かれた後の4周期後、ハードウェアによって解除(0)されます。上記IVSELで説明されるようにIVCEビットの設定(1)は割り込みを(一時的に)禁止します。以下のコード例をご覧ください。

#### アセンブリ言語プログラム例

```

MOVE_IVT:    IN      R16, MCUCR           ; 現MCUCR値取得
              MOV     R17, R16           ; 現MCUCR値複写
              ORI     R16, (1<<IVCE)     ; IVCE論理1値を取得
              OUT     MCUCR, R16         ; IVCEに論理1書き込み
              ORI     R17, (1<<IVSEL)     ; IVSEL論理1値を取得
              OUT     MCUCR, R17         ; ブート領域へ割り込みベクタを移動
              RET                        ; 呼び出し元へ復帰
    
```

#### C言語プログラム例

```

void Move_interrupts(void)
{
    uchar temp;                          /* 一時定数定義 */
    temp = MCUCR;                        /* 現MCUCR値取得 */
    MCUCR = temp | (1<<IVCE);            /* IVCEに論理1書き込み */
    MCUCR = temp | (1<<IVSEL);           /* ブート領域へ割り込みベクタを移動 */
}
    
```

## 12. 外部割り込み

### 12.1. 概要

外部割り込みはINT0ピンまたはPCINT0～30ピン(注2)の何れかによって起動されます。許可したなら、例えばINT0またはPCINT0～30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PCI0は許可したPCINT0～7の何れかが切り替わると起動します。同様にPCI1,PCI2,PCI3は各々PCINT8～15, PCINT16～23,PCINT24～30に対応します(誤注:前行と本行とに修正)。ピン変化割り込み許可レジスタn(PCMSK0,PCMSK1,PCMSK2(注1),PCMSK3(注1))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。PCINT0～30のピン変化割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイスを起動するのに使えることを意味します。

INT0割り込みは上昇端または下降端(含む両端)、またはLowレベルによって起動できます。これは外部割り込み制御レジスタA(EICRA)の詳述で示される設定です。INT0割り込みがレベル起動として設定、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT0の上昇端または下降端割り込みの認知は、17頁の「クロックシステムとその配給」で記述されるI/Oクロックの存在を必要とすることに注意してください。INT0のLowレベル割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からも、デバイスを起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

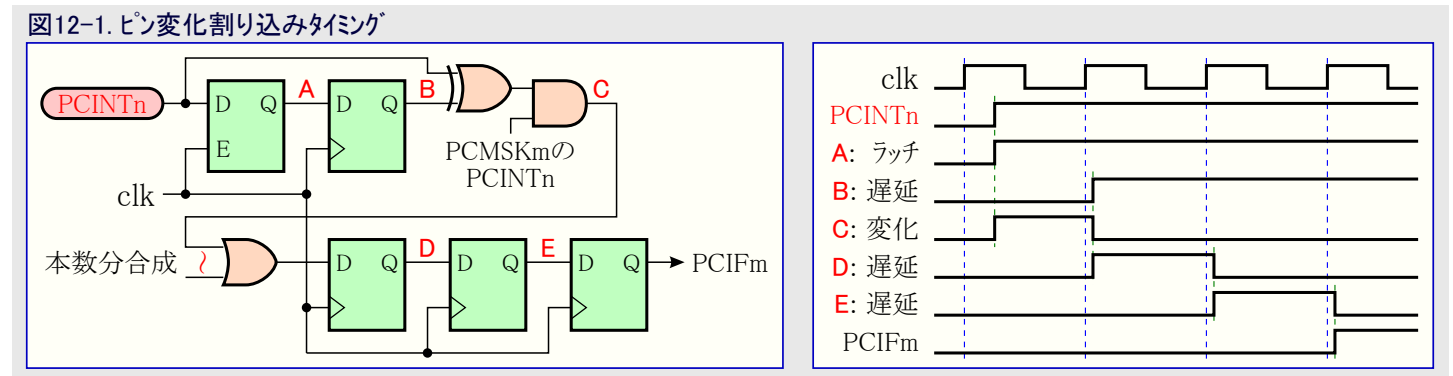
レベル起動割り込みがパワーダウン動作からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが起動時間の最後に先立って消滅すると、MCUは今までどおり起動しますが、割り込みが生成されません。起動時間は17頁の「システムクロックとクロック選択」で示されるようにSUTヒューズとCKSELヒューズで定義されます。

注1: PCMASK2とPCMSK3はATmega3250Pにだけ存在します。

注2: PCINT30～16はATmega3250Pにだけ存在します。PCINT15～0はATmega325Pにも存在します。詳細については2頁の「ピン配置」と「外部割り込み用レジスタ」をご覧ください。

### 12.2. ピン変化割り込みタイミング

ピン変化割り込みの例は図12-1.で示されます。



### 12.3. 外部割り込み用レジスタ

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

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	
	—	—	—	—	—	—	ISC01	ISC00	EICRA
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

外部割り込み0はステータスレジスタ(SREG)の全割り込み許可(I)ビットと外部割り込み許可レジスタ(EIMSK)の外部割り込み0許可(INT0)ビットが設定(1)される場合のINT0外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT0ピンの端(エッジ)とレベルは表12-1.で定義されます。INT0ピンの値は端検知前に採取されています。端または論理変化(両端)割り込みが選ばれる場合、1クロック周期よりも長く留まるパルスは割り込みを生成します。より短いパルスは割り込みの発生が保証されません。Lowレベル割り込みが選ばれる場合、Lowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表12-1. 外部割り込み0(INT0)割り込み条件

ISC01	ISC00	INT0ピン割り込み発生条件
0	0	Lowレベル。
0	1	論理変化(両端)
1	0	下降端
1	1	上昇端



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

ビット	7	6	5	4	3	2	1	0	
\$1D (\$3D)	(PCIE3)	(PCIE2)	PCIE1	PCIE0	—	—	—	INT0	EIMSK
Read/Write	R/W	R/W	R/W	R/W	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット7 – PCIE3 : ピン変化3群割り込み許可 (Pin Change Interrupt Enable 3)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE3ビットが設定(1)されると、ピン変化3群割り込みが許可されます。許可したPCINT24～30ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みは[PCI3割り込みベクタ](#)から実行されます。PCINT24～30ピンは[ピン変化割り込み許可レジスタ3\(PCMSK3\)](#)によって個別に許可されます。

本ビットはATmega325Pでは予約ビットで、常に0が書かれるべきです。

### • ビット6 – PCIE2 : ピン変化2群割り込み許可 (Pin Change Interrupt Enable 2)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE2ビットが設定(1)されると、ピン変化2群割り込みが許可されます。許可したPCINT16～23ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みは[PCI2割り込みベクタ](#)から実行されます。PCINT16～23ピンは[ピン変化割り込み許可レジスタ2\(PCMSK2\)](#)によって個別に許可されます。

本ビットはATmega325Pでは予約ビットで、常に0が書かれるべきです。

### • ビット5 – PCIE1 : ピン変化1群割り込み許可 (Pin Change Interrupt Enable 1)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE1ビットが設定(1)されると、ピン変化1群割り込みが許可されます。許可したPCINT8～15ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みは[PCI1割り込みベクタ](#)から実行されます。PCINT8～15ピンは[ピン変化割り込み許可レジスタ1\(PCMSK1\)](#)によって個別に許可されます。

### • ビット4 – PCIE0 : ピン変化0群割り込み許可 (Pin Change Interrupt Enable 0)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE0ビットが設定(1)されると、ピン変化0群割り込みが許可されます。許可したPCINT0～7ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みは[PCI0割り込みベクタ](#)から実行されます。PCINT0～7ピンは[ピン変化割り込み許可レジスタ0\(PCMSK0\)](#)によって個別に許可されます。

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

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、INT0ビットが設定(1)されると、INT0外部ピン割り込みが許可されます。外部割り込み制御レジスタA(EICRA)の割り込み条件制御0のビット1と0(ISC01,ISC00)は、この外部割り込みがINT0ピンの上昇端、下降端、両端、またはLowレベルのどれで活性(有効)にされるかを定義します。例えばINT0ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求0に対応する割り込みは[INT0割り込みベクタ](#)から実行されます。

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

ビット	7	6	5	4	3	2	1	0	
\$1C (\$3C)	(PCIF3)	(PCIF2)	PCIF1	PCIF0	—	—	—	INTF0	EIFR
Read/Write	R/W	R/W	R/W	R/W	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット7 – PCIF3 : ピン変化3群割り込み要求フラグ (Pin Change Interrupt Flag 3)

PCINT24～30ピンの何れかの論理変化が割り込み要求を起動すると、PCIF3が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと外部割り込み許可レジスタ(EIMSK)の[ピン変化3群割り込み許可\(PCIE3\)](#)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

本ビットはATmega325Pでは予約ビットで、常に0として読みます。

### • ビット6 – PCIF2 : ピン変化2群割り込み要求フラグ (Pin Change Interrupt Flag 2)

PCINT16～23ピンの何れかの論理変化が割り込み要求を起動すると、PCIF2が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと外部割り込み許可レジスタ(EIMSK)の[ピン変化2群割り込み許可\(PCIE2\)](#)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

本ビットはATmega325Pでは予約ビットで、常に0として読みます。

### • ビット5 – 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と、PCIF1が設定(1)になります。[ステータスレジスタ\(SREG\)の全割り込み許可\(I\)ビット](#)と外部割り込み許可レジスタ(EIMSK)の[ピン変化1群割り込み許可\(PCIE1\)](#)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

#### •ビット4 – PCIF0 : ピン変化0群割り込み要求フラグ (Pin Change Interrupt Flag 0)

PCINT0～7ピンの何れかの論理変化が割り込み要求を起動すると、PCIF0が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと外部割り込み許可レジスタ(EIMSK)のピン変化0群割り込み許可(PCIE0)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

#### •ビット0 – INTF0 : 外部割り込み0要求フラグ (External Interrupt Flag0)

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

#### 12.3.4. ピン変化割り込み許可レジスタ3 (Pin Change Enable Mask 24～30) PCMSK3 (注)

ビット (\$73)	7	6	5	4	3	2	1	0	
	–	PCINT30	PCINT29	PCINT28	PCINT27	PCINT26	PCINT25	PCINT24	PCMSK3
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### •ビット7～0 – PCINT30～ PCINT24 : ピン変化割り込み30～24許可 (Pin Change Enable Mask 30～24)

各PCINT24～30ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT24～30と外部割り込み許可レジスタ(EIMSK)のPCIE3が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT24～30が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

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

ビット (\$6D)	7	6	5	4	3	2	1	0	
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	PCMSK2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### •ビット7～0 – PCINT23～ PCINT16 : ピン変化割り込み23～16許可 (Pin Change Enable Mask 23～16)

各PCINT16～23ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT16～23と外部割り込み許可レジスタ(EIMSK)のPCIE2が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT16～23が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

注: PCMSK2とPCMSK3はATmega3250Pにだけ存在します。

#### 12.3.6. ピン変化割り込み許可レジスタ1 (Pin Change Enable Mask 8～15) PCMSK1

ビット (\$6C)	7	6	5	4	3	2	1	0	
	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	PCMSK1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### •ビット7～0 – PCINT15～ PCINT8 : ピン変化割り込み15～8許可 (Pin Change Enable Mask 15～8)

各PCINT8～15ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT8～15と外部割り込み許可レジスタ(EIMSK)のPCIE1が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT8～15が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

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

ビット (\$6B)	7	6	5	4	3	2	1	0	
	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	PCMSK0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

各PCINT0～7ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT0～7と外部割り込み許可レジスタ(EIMSK)のPCIE0が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT0～7が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

## 13. 入出力ポート

### 13.1. 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。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。ポートBは他のポートより高いピン駆動能力を持ちますが、全てのピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピンは個別に選択可能な、供給電圧で抵抗値が変化しないプルアップ抵抗を持っています。全てのI/Oピンは図13-1.で示されるようにVCCとGNDの両方に保護ダイオードを持ちます。各値の完全な一覧については193頁の「電気的特性」を参照してください。「絶対最大定格」のピン電圧を超える場合、それに応じて制限されないなら、これに起因する電流がデバイスに損害を及ぼし得ます。

本章内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記された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55頁の「I/Oポート用レジスタ」で一覧されます。

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

標準デジタルI/OとしてのI/Oポートの使用は次の「標準デジタル入出力としてのポート」で記述されます。多くのポートピンはデバイスの周辺機能用の交換機能と多重化されます。ポートピンとの各交換機能のインターフェース法は42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。

### 13.2. 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13-2.はここで属にPx<sub>n</sub>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13-1. 入出力ピン等価回路

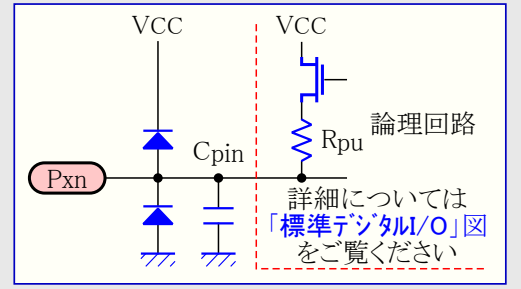
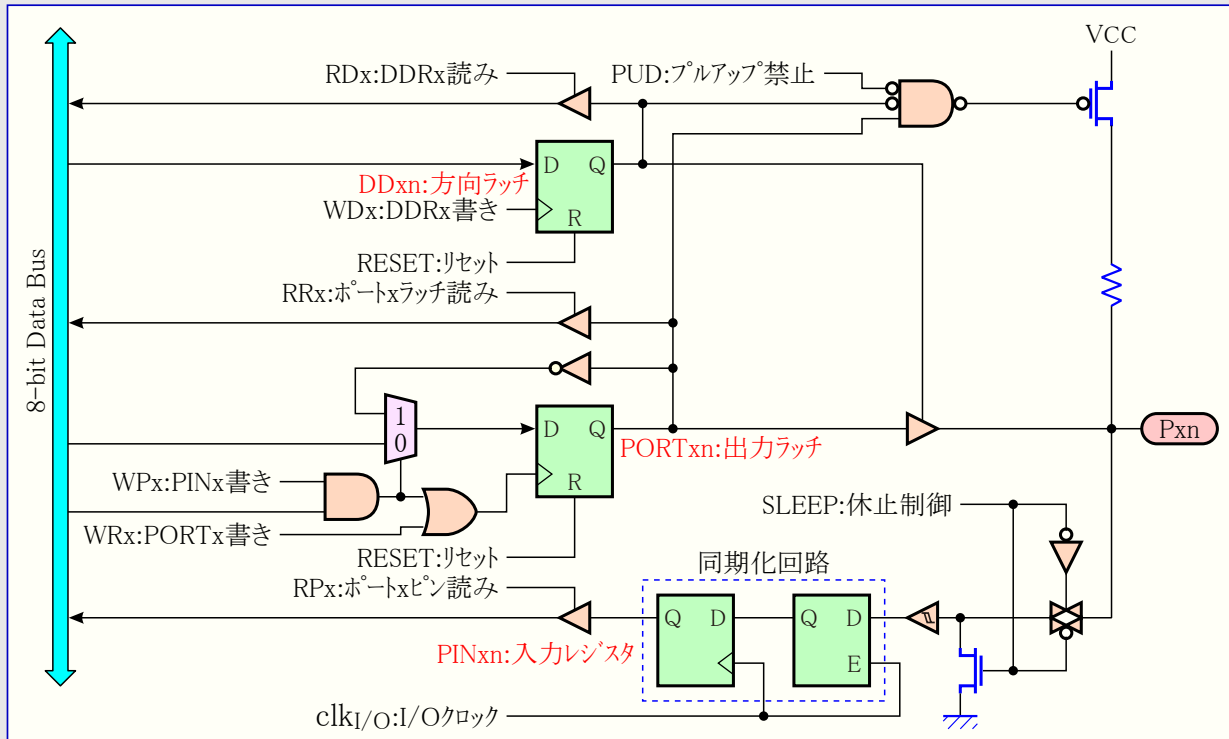


図13-2. 標準デジタル入出力回路構成



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

### 13.2.1. ピンの設定

各ポートピンは3つのレジスタビットDDxn、PORTxn、PINxnから成ります。55頁の「I/Oポート用レジスタ」で示されるように、DDxnビットはDDRx I/Oアドレス、PORTxnビットはPORTx I/Oアドレス、PINxnビットはPINx I/Oアドレスでアクセスされます。

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

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

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

### 13.2.2. 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。

### 13.2.3. 入出力間の切り替え

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

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

表13-1. はピン値に対する制御信号の一覧を示します。

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

### 13.2.4. 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13-2.で示されるようにPINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。図13-3.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13-3.で)システムクロックの最初の上昇端直後から始まるクロック周期を考察してください。このラッチはクロックがLowの時に閉じ、クロックがHighの時に同期ラッチ信号の斜線部分で示されるように通過(トランスペアレント)となります。この信号値はシステムクロックがLowになる時に保持(ラッチ)されます。それが続くクロックの上昇端でPINxnレジスタに取り込まれます。2つの矢印 $t_{pd,min}$ と $t_{pd,max}$ によって示されるように、ピン上の単一信号遷移は出現時点に依存して0.5~1.5システムクロック周期遅らされます。

ソフトウェアが指定したピン値を読み戻す時は、図13-4.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13-3. 外部供給ピン値読み込み時の同期化

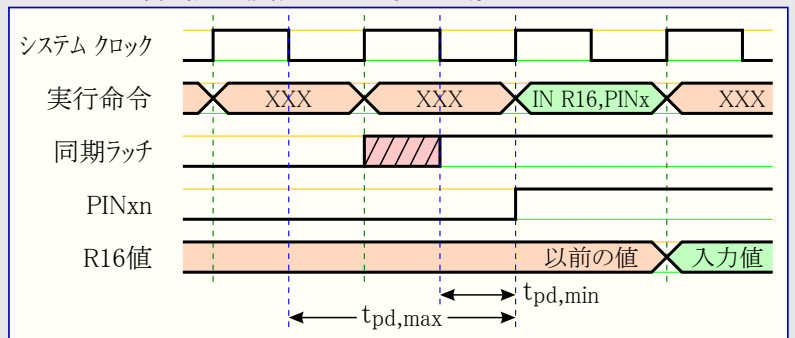
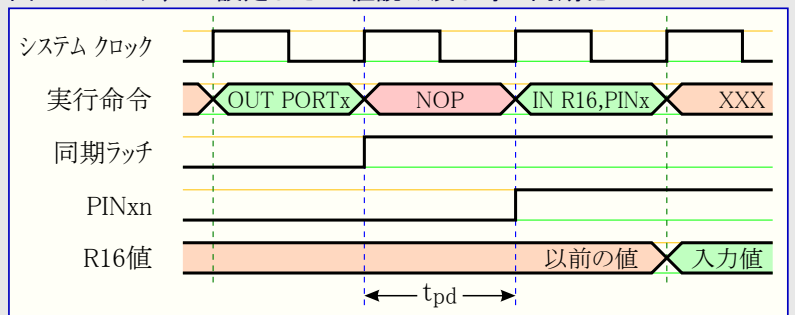


図13-4. プログラムで設定したピン値読み戻し時の同期化





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

## アセンブリ言語プログラム例

```

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

```

## C言語プログラム例

```

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

```

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

### 13.2.5. デジタル入力許可と休止形態

図13-2.で示されるようにデジタル入力信号はシュミットトリガの入力をGNDにクランプできます。この図で**SLEEP**と印された信号は入力信号のいくつかが開放のまま、またはVCC/2付近のアナログ信号電圧を持つ場合の高消費電力を避けるため、**パワーダウ**動作、**パワーセーブ**動作、**スタンバイ**動作でMCU休止制御器によって設定(1)されます。

**SLEEP**は外部割り込みピンとして許可されたポートピンに対しては無視されます。外部割り込み要求が許可されないなら、**SLEEP**は他のピンについてと同様に有効です。**SLEEP**は42頁の「**交換ポート機能**」で記載されるように様々な他の交換機能によっても無視されます。

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から(復帰)再開する時に、これらの休止形態に於けるクランプが要求された論理変化を生ずるので、対応する外部割り込み要求フラグが設定(=1)されます。

### 13.2.6. 未接続ピン

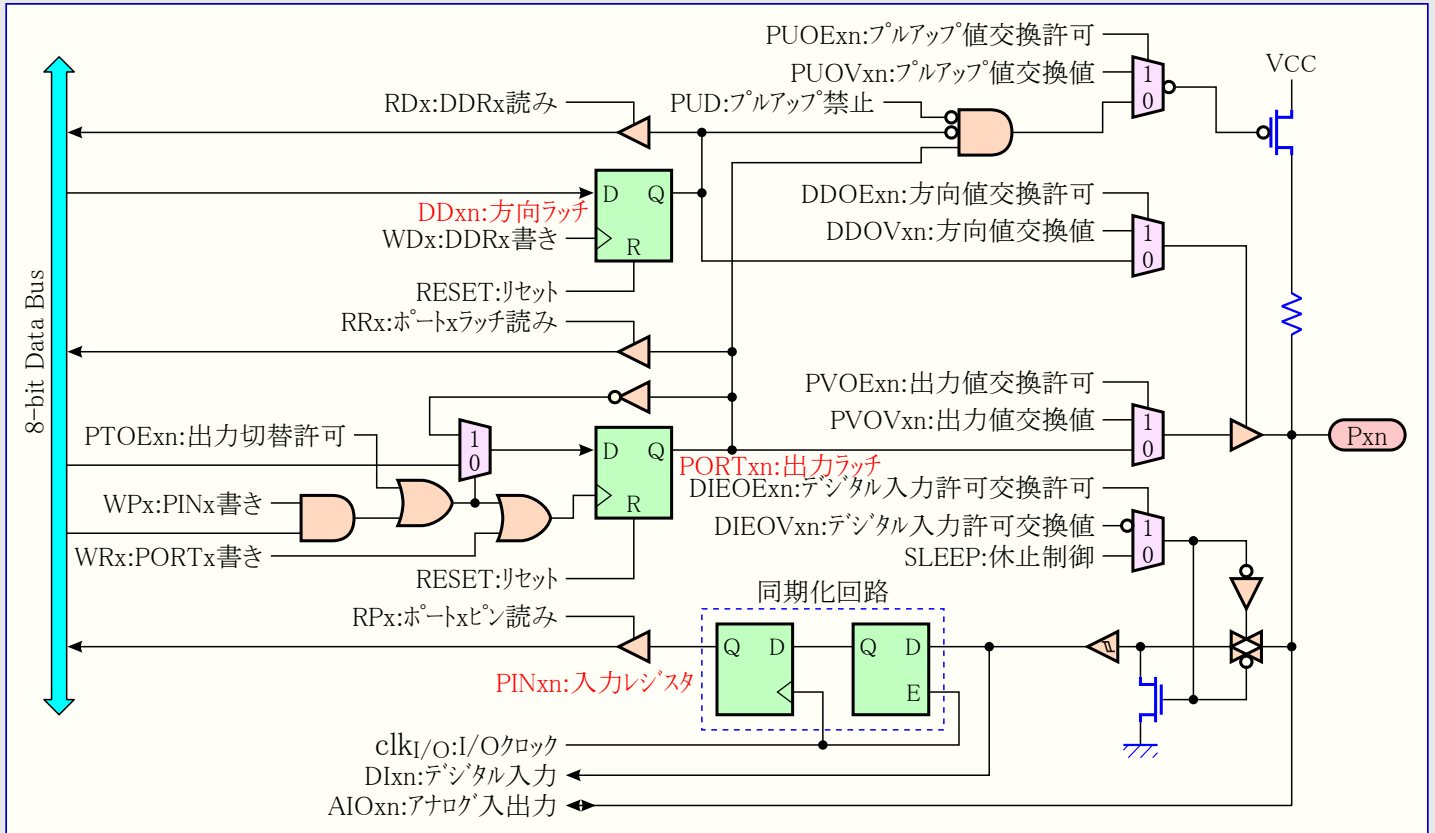
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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

### 13.3. 交換ポート機能

多くのポートピンには標準デジタル入出力に加え交換機能があります。図13-5は単純化された図13-2.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVRマイクロコントローラシステムの全ポートピンに適用できる一般的な記述として取り扱います。

図13-5. 交換ポート機能入出力回路構成



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

表13-2は重複(交換)信号の機能一覧を示します。図13-5.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13-2. 交換機能用交換信号の一般定義

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

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

## 13.3.1. ポートBの交換機能

ポートBピンの交換機能は表13-3.で示されます。

表13-3. ポートBピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PB7	OC2A (タイマ/カウンタ2 比較一致/PWM出力) PCINT15 (ピン変化割り込み15入力)	PB3	MISO (SPI 主側データ入力/従側データ出力) PCINT11 (ピン変化割り込み11入力)
PB6	OC1B (タイマ/カウンタ1 比較B一致/PWM出力) PCINT14 (ピン変化割り込み14入力)	PB2	MOSI (SPI 主側データ出力/従側データ入力) PCINT10 (ピン変化割り込み10入力)
PB5	OC1A (タイマ/カウンタ1 比較A一致/PWM出力) PCINT13 (ピン変化割り込み13入力)	PB1	SCK (SPI 直列クロック 主側出力/従側入力) PCINT9 (ピン変化割り込み9入力)
PB4	OC0A (タイマ/カウンタ0 比較一致/PWM出力) PCINT12 (ピン変化割り込み12入力)	PB0	$\overline{SS}$ (SPI 従装置選択入力) PCINT8 (ピン変化割り込み8入力)

交換ピンの設定は次のとおりです。

### • OC2A/PCINT15 – ポートB ビット7 : PB7

OC2A : タイマ/カウンタ2の比較A一致出力。PB7ピンはタイマ/カウンタ2の比較A一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(**DDRB7=1**)されなければなりません。このOC2Aピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT15 : ピン変化割り込み15入力。PB7ピンは外部割り込み元としても扱えます。

### • OC1B/PCINT14 – ポートB ビット6 : PB6

OC1B : タイマ/カウンタ1の比較B一致出力。PB6ピンはタイマ/カウンタ1の比較B一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(**DDRB6=1**)されなければなりません。このOC1Bピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT14 : ピン変化割り込み14入力。PB6ピンは外部割り込み元としても扱えます。

### • OC1A/PCINT13 – ポートB ビット5 : PB5

OC1A : タイマ/カウンタ1の比較A一致出力。PB5ピンはタイマ/カウンタ1の比較A一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(**DDRB5=1**)されなければなりません。このOC1Aピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT13 : ピン変化割り込み13入力。PB5ピンは外部割り込み元としても扱えます。

### • OC0A/PCINT12 – ポートB ビット4 : PB4

OC0A : タイマ/カウンタ0の比較一致出力。PB4ピンはタイマ/カウンタ0の比較一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(**DDRB4=1**)されなければなりません。このOC0Aピンはタイマ機能のPWM動作用出力ピンでもあります。

PCINT12 : ピン変化割り込み12入力。PB4ピンは外部割り込み元としても扱えます。

### • MISO/PCINT11 – ポートB ビット3 : PB3

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

PCINT11 : ピン変化割り込み11入力。PB3ピンは外部割り込み元としても扱えます。

### • MOSI/PCINT10 – ポートB ビット2 : PB2

MOSI : SPI用の主装置データ出力、従装置データ入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB2の設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB2によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB2によって制御できません。

PCINT10 : ピン変化割り込み10入力。PB2ピンは外部割り込み元としても扱えます。

### • SCK/PCINT9 – ポートB ビット1 : PB1

SCK : SPI用の主装置クロック出力、従装置クロック入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB1の設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB1によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB1によって制御できません。

PCINT9 : ピン変化割り込み9入力。PB1ピンは外部割り込み元としても扱えます。

•  $\overline{SS}$ /PCINT8 – ポートB ビット0 : PB0

$\overline{SS}$  : SPI従装置選択入力。SPIが従装置として許可されると、このピンはDDRBのDDB0設定に拘らず、入力として設定されます。従装置としてこのピンがLowに駆動されると、SPI(機能)が活性化(有効に)されます。SPIが主装置として許可されると、このピンのデータ方向はDDB0によって制御されます。このピンがSPIにより入力を強制される時にプルアップは未だPORTBのPORTB0によって制御できます。

PCINT8 : ピン変化割り込み8入力。PB0ピンは外部割り込み元としても扱えます。

表13-4.と表13-5.はポートBの交換機能を42頁の図13-5.で示される交換信号に関連付けます。SPI主装置入力とSPI従装置出力がMISOを構成し、一方MOSIはSPI従装置出力とSPI従装置入力に分けられます。

表13-4. ポートB7～4の交換機能用交換信号

信号名	PB7/OC2A/PCINT15	PB6/OC1B/PCINT14	PB5/OC1A/PCINT13	PB4/OC0A/PCINT12
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	OC2A許可	OC1B許可	OC1A許可	OC0A許可
PVOV	OC2A	OC1B	OC1A	OC0A
PTOE	–	–	–	–
DIOE	PCIE1・PCINT15	PCIE1・PCINT14	PCIE1・PCINT13	PCIE1・PCINT12
DIOV	1	1	1	1
DI	PCINT15入力	PCINT14入力	PCINT13入力	PCINT12入力
AIO	–	–	–	–

表13-5. ポートB3～0の交換機能用交換信号

信号名	PB3/MISO/PCINT11	PB2/MOSI/PCINT10	PB1/SCK/PCINT9	PB0/ $\overline{SS}$ /PCINT8
PUOE	SPE・MSTR	SPE・ $\overline{MSTR}$	SPE・ $\overline{MSTR}$	SPE・ $\overline{MSTR}$
PUOV	PORTB3・ $\overline{PUD}$	PORTB2・ $\overline{PUD}$	PORTB1・ $\overline{PUD}$	PORTB0・ $\overline{PUD}$
DDOE	SPE・MSTR	SPE・MSTR	SPE・MSTR	SPE・MSTR
DDOV	0	0	0	0
PVOE	SPE・ $\overline{MSTR}$	SPE・MSTR	SPE・MSTR	0
PVOV	SPI従装置出力	SPI主装置出力	SCK出力	0
PTOE	–	–	–	–
DIOE	PCIE1・PCINT11	PCIE1・PCINT10	PCIE1・PCINT9	PCIE1・PCINT8
DIOV	1	1	1	1
DI	SPI主装置/PCINT11入力	SPI従装置/PCINT10入力	SCK/PCINT9入力	SPI $\overline{SS}$ /PCINT8入力
AIO	–	–	–	–



## 13.3.2. ポートDの交換機能

ポートDピンの交換機能は表13-6.で示されます。

表13-6. ポートDピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PD7	–	PD3	–
PD6	–	PD2	–
PD5	–	PD1	INT0 (外部割り込み0 入力)
PD4	–	PD0	ICP1 (タイマ/カウンタ1 捕獲起動入力)

交換ピンの設定は次のとおりです。

- INT0 – ポートD ビット1 : PD1

INT0 : 外部割り込み0入力。PD1ピンはMCUへの外部割り込み元として扱えます。

- ICP1 – ポートD ビット0 : PD0

ICP1 : タイマ/カウンタ1の捕獲起動入力。PD0ピンはタイマ/カウンタ1用捕獲起動入力ピンとして動作できます。

表13-7.はポートDの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-7. ポートD3～0の交換機能用交換信号

信号名	PD3	PD2	PD1/INT0	PD0/ICP1
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	–	–	–	–
DIOE	0	0	INT0許可	0
DIOV	0	0	INT0許可	0
DI	–	–	INT0入力	ICP1入力
AIO	–	–	–	–

### 13.3.3. ポートEの交換機能

ポートEピンの交換機能は表13-8.で示されます。

表13-8. ポートEピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PE7	CLKO (システムクロック出力) PCINT7 (ピン変化割り込み7入力)	PE3	AIN1 (アナログ比較器反転入力) PCINT3 (ピン変化割り込み3入力)
PE6	DO (3線動作USIデータ出力) PCINT6 (ピン変化割り込み6入力)	PE2	AIN0 (アナログ比較器非反転入力) XCK (USART 外部クロック入出力) PCINT2 (ピン変化割り込み2入力)
PE5	DI (3線動作USIデータ入力) SDA (2線動作USI(TWI)データ入出力) PCINT5 (ピン変化割り込み5入力)	PE1	TXD (USART 送信データ出力) PCINT1 (ピン変化割り込み1入力)
PE4	USCK (3線動作USIクロック入出力) SCL (2線動作USI(TWI)クロック入出力) PCINT4 (ピン変化割り込み4入力)	PE0	RXD (USART 受信データ入力) PCINT0 (ピン変化割り込み0入力)

交換ピンの設定は次のとおりです。

- CLKO/PCINT7 – ポートE ビット7 : PE7

CLKO : システム クロック出力。分周したシステム クロックがPE7ピンに出力できます。分周したシステム クロックはCKOUTヒューズがプログラム(0)されると、PORTE7とDDE7設定に拘らず、出力されます。これはリセット中にも出力されます。

PCINT7 : ピン変化割り込み7入力。PE7ピンは外部割り込み元としても扱えます。

- DO/PCINT6 – ポートE ビット6 : PE6

DO : 3線動作USIのデータ出力。

PCINT6 : ピン変化割り込み6入力。PE6ピンは外部割り込み元としても扱えます。

- DI/SDA/PCINT5 – ポートE ビット5 : PE5

DI : 3線動作USIのデータ入力。

SDA : 2線動作USI(TWI)のデータ入出力。

PCINT5 : ピン変化割り込み5入力。PE5ピンは外部割り込み元としても扱えます。

- USCK/SCL/PCINT4 – ポートE ビット4 : PE4

USCK : 3線動作USIのクロック入出力。

SCL : 2線動作USI(TWI)のクロック入出力。

PCINT4 : ピン変化割り込み4入力。PE4ピンは外部割り込み元としても扱えます。

- AIN1/PCINT3 – ポートE ビット3 : PE3

AIN1 : アナログ比較器の反転入力。このピンはアナログ比較器の反転入力に直接接続されます。

PCINT3 : ピン変化割り込み3入力。PE3ピンは外部割り込み元としても扱えます。

- AIN0/XCK/PCINT2 – ポートE ビット2 : PE2

AIN0 : アナログ比較器の非反転入力。このピンはアナログ比較器の非反転入力に直接接続されます。

XCK : USARTの外部クロック入出力。ポートE方向レジスタ(DDRE)のDDE2は、このクロックが入力(DDE2=0)または出力(DDE2=1)のどちらかを制御します。XCKピンはUSARTが同期種別で動作する時だけ活動します(有効です)。

PCINT2 : ピン変化割り込み2入力。PE2ピンは外部割り込み元としても扱えます。

- TXD/PCINT1 – ポートE ビット1 : PE1

TXD : 送信データ(USART用データ出力ピン)。USART送信部が許可されると、ピンはDDREのDDE1値に拘らず出力として設定されます。

PCINT1 : ピン変化割り込み1入力。PE1ピンは外部割り込み元としても扱えます。

- RXD/PCINT0 – ポートE ビット0 : PE0

RXD : 受信データ(USART用データ入力ピン)。USART受信部が許可されると、このピンはDDREのDDE0の値に拘らず入力として設定されます。USARTがこのピンを入力に強制するとき、プルアップは未だPORTE0ビットによって制御できます。

PCINT0 : ピン変化割り込み0入力。PE0ピンは外部割り込み元としても扱えます。

表13-9.と表13-10.はポートEの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-9. ポートE7～4の交換機能用交換信号

信号名	PE7/PCINT7/(CLKO)	PE6/DO/PCINT6	PE5/DI/SDA/PCINT5	PE4/USCK/SCL/PCINT4
PUOE	0	0	2線USI	2線USI
PUOV	0	0	0	0
DDOE	CKOUT	0	2線USI	2線USI
DDOV	1	0	(SDA+PORTE5)・DDE5	(SCL保持+PORTE4)・DDE4
PVOE	CKOUT	3線USI	2線USI・DDE5	2線USI・DDE4
PVOV	clk <sub>i</sub> /O	DO	0	0
PTOE	–	–	0	USITC
DIEOE	PCIE0・PCINT7	PCIE0・PCINT6	(PCIE0・PCINT5)+USISIE	(PCIE0・PCINT4)+USISIE
DIEOV	1	1	1	1
DI	PCINT7入力	PCINT6入力	DI/SDA/PCINT5入力	USCK/SCL/PCINT4入力
AIO	–	–	–	–

注: CKOUTピュースがプログラム(0)ならばCKOUTが論理1です。

表13-10. ポートE3～0の交換機能用交換信号

信号名	PE3/AIN1/PCINT3	PE2/AIN0/XCK/PCINT2	PE1/TXD/PCINT1	PE0/RXD/PCINT0
PUOE	0	0	TXEN	RXEN
PUOV	0	0	0	PORTE0・PUD
DDOE	0	0	TXEN	RXEN
DDOV	0	0	1	0
PVOE	0	XCK出力許可	TXEN	0
PVOV	0	XCK	TXD	0
PTOE	–	–	–	–
DIEOE	(PCIE0・PCINT3)+AIN1D	(PCIE0・PCINT2)+AIN0D	PCIE0・PCINT1	PCIE0・PCINT0
DIEOV	PCIE0・PCINT3	PCIE0・PCINT2	1	1
DI	PCINT3入力	XCK/PCINT2入力	PCINT1入力	RXD/PCINT0入力
AIO	AIN1入力	AIN0入力	–	–

注: AIN0DとAIN1Dは137頁の「デジタル入力禁止レジスタ1(DIDR1)」で記述されます。

### 13.3.4. ポートFの交換機能

ポートFピンには表13-11.に示されるようにA/D変換用アナログ入力としての交換機能があります。ポートFピンのいくつかが出力として設定される場合、A/D変換実行中の時にそれらを切り替えないことが重要です。これは変換結果を不正にするかもしれません。JTAGインターフェースが許可されると、リセットが起きてもPF7(TDI)、PF5(TMS)、PF4(TCK)ピンのプルアップ抵抗は活性(有効)にされます。

表13-11. ポートFピンの交換機能

ポートピン	交換機能
PF7	ADC7 (A/D変換アナログ入力チャネル7) TDI (JTAG 検査データ入力)
PF6	ADC6 (A/D変換アナログ入力チャネル6) TDO (JTAG 検査データ出力)
PF5	ADC5 (A/D変換アナログ入力チャネル5) TMS (JTAG 検査種別選択入力)
PF4	ADC4 (A/D変換アナログ入力チャネル4) TCK (JTAG クロック入力)
PF3	ADC3 (A/D変換アナログ入力チャネル3)
PF2	ADC2 (A/D変換アナログ入力チャネル2)
PF1	ADC1 (A/D変換アナログ入力チャネル1)
PF0	ADC0 (A/D変換アナログ入力チャネル0)

交換ピンの設定は次のとおりです。

- ADC7/TDI – ポートF ビット7 : PF7

ADC7 : A/D変換チャネル7入力。

TDI : JTAG検査データ入力。命令レジスタまたはデータレジスタ(走査チェーン)に移動入力される直列入力データです。JTAGインターフェースが許可されると、このピンはI/Oピンとして使えません。

- ADC6/TDO – ポートF ビット6 : PF6

ADC6 : A/D変換チャネル6入力。

TDO : JTAG検査データ出力。命令レジスタまたはデータレジスタから移動出力される直列出力データです。JTAGインターフェースが許可されたなら、このピンはI/Oピンとして使えません。データを移動出力するTAP状態に於いてTDOピンは動的に駆動します。他の状態ではこのピンがHighに引かれます。

- ADC5/TMS – ポートF ビット5 : PF5

ADC5 : A/D変換チャネル5入力。

TMS : JTAG検査種別選択。このピンはTAP(検査入出力ポート)制御器状態機構を通しての操作に使われます。JTAGインターフェースが許可されると、このピンはI/Oピンとして使えません。

- ADC4/TCK – ポートF ビット4 : PF4

ADC4 : A/D変換チャネル4入力。

TCK : JTAGクロック。JTAG操作はTCKに同期します。JTAGインターフェースが許可されると、このピンはI/Oピンとして使えません。

- ADC3 – ポートF ビット3 : PF3

ADC3 : A/D変換チャネル3入力。

- ADC2 – ポートF ビット2 : PF2

ADC2 : A/D変換チャネル2入力。

- ADC1 – ポートF ビット1 : PF1

ADC1 : A/D変換チャネル1入力。

- ADC0 – ポートF ビット0 : PF0

ADC0 : A/D変換チャネル0入力。



表13-12.と表13-13.はポートFの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-12. ポートF7～4の交換機能用交換信号

信号名	PF7/ADC7/TDI	PF6/ADC/TDO	PF5/ADC5/TMS	PF4/ADC4/TCK
PUOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
PUOV	1	1	1	1
DDOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
DDOV	0	IR移動+DR移動	0	0
PVOE	0	JTAGEN	0	0
PVOV	0	TDO	0	0
PTOE	–	–	–	–
DIOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
DIOV	0	0	0	0
DI	–	–	–	–
AIO	ADC7入力/TDI入力	ADC6入力	ADC5入力/TMS入力	ADC4入力/TCK入力

表13-13. ポートF3～0の交換機能用交換信号

信号名	PF3/ADC3	PF2/ADC2	PF1/ADC1	PF0/ADC0
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	–	–	–	–
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	–	–	–	–
AIO	ADC3入力	ADC2入力	ADC1入力	ADC0入力

### 13.3.5. ポートGの交換機能

ポートGピンの交換機能は表13-14.で示されます。

表13-14. ポートGピンの交換機能

ポートピン	交換機能
PG5	RESET (外部リセット入力)
PG4	T0 (タイマ/カウンタ0 外部クロック入力)
PG3	T1 (タイマ/カウンタ1 外部クロック入力)
PG2	–
PG1	–
PG0	–

注: PG5は入力専用で、常にプルアップが有効です。RSTDISBLヒューズについては表25-3.をご覧ください。

交換ピンの設定は次のとおりです。

- RESET – ポートGビット5 : PG5

RESET : 外部リセット入力。RSTDISBLヒューズがプログラム(0)されると、PG5は常時プルアップの入力として機能します。

- T0 – ポートGビット4 : PG4

T0 : タイマ/カウンタ0の外部クロック入力ピンです。

- T1 – ポートGビット3 : PG3

T1 : タイマ/カウンタ1の外部クロック入力ピンです。

表13-15.はポートGの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-15. ポートG4~0の交換機能用交換信号

信号名	PG4/T0	PG3/T1	PG2	PG1	PG0
PUOE	0	0	0	0	0
PUOV	0	0	0	0	0
DDOE	0	0	0	0	0
DDOV	0	0	0	0	0
PVOE	0	0	0	0	0
PVOV	0	0	0	0	0
PTOE	–	–	–	–	–
DIEOE	0	0	0	0	0
DIEOV	0	0	0	0	0
DI	T0入力	T1入力	–	–	–
AIO	–	–	–	–	–

## 13.3.6. ポートHの交換機能

ポートHはATmega3250Pにのみ存在します。交換機能は次のとおりです。

表13-16. ポートHピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PH7	PCINT23 (ピン変化割り込み23入力)	PH3	PCINT19 (ピン変化割り込み19入力)
PH6	PCINT22 (ピン変化割り込み22入力)	PH2	PCINT18 (ピン変化割り込み18入力)
PH5	PCINT21 (ピン変化割り込み21入力)	PH1	PCINT17 (ピン変化割り込み17入力)
PH4	PCINT20 (ピン変化割り込み20入力)	PH0	PCINT16 (ピン変化割り込み16入力)

交換ピンの設定は次のとおりです。

- PCINT23 – ポートH ビット7 : PH7  
PCINT23 : ピン変化割り込み23入力。PH7ピンは外部割り込み元としても扱えます。
- PCINT22 – ポートH ビット6 : PH6  
PCINT22 : ピン変化割り込み22入力。PH6ピンは外部割り込み元としても扱えます。
- PCINT21 – ポートH ビット5 : PH5  
PCINT21 : ピン変化割り込み21入力。PH5ピンは外部割り込み元としても扱えます。
- PCINT20 – ポートH ビット4 : PH4  
PCINT20 : ピン変化割り込み20入力。PH4ピンは外部割り込み元としても扱えます。
- PCINT19 – ポートH ビット3 : PH3  
PCINT19 : ピン変化割り込み19入力。PH3ピンは外部割り込み元としても扱えます。
- PCINT18 – ポートH ビット2 : PH2  
PCINT18 : ピン変化割り込み18入力。PH2ピンは外部割り込み元としても扱えます。
- PCINT17 – ポートH ビット1 : PH1  
PCINT17 : ピン変化割り込み17入力。PH1ピンは外部割り込み元としても扱えます。
- PCINT16 – ポートH ビット0 : PH0  
PCINT16 : ピン変化割り込み16入力。PH0ピンは外部割り込み元としても扱えます。

表13-17と表13-18はポートHの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-17. ポートH7～4の交換機能用交換信号

信号名	PH7/PCINT23	PH6/PCINT22	PH5/PCINT21	PH4/PCINT20
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	–	–	–	–
DIOE	PCIE2・PCINT23	PCIE2・PCINT22	PCIE2・PCINT21	PCIE2・PCINT20
DIOV	0	0	0	0
DI	PCINT23入力	PCINT22入力	PCINT21入力	PCINT20入力
AIO	–	–	–	–

表13-18. ポートH3～0の交換機能用交換信号

信号名	PH3/PCINT19	PH2/PCINT18	PH1/PCINT17	PH0/PCINT16
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	–	–	–	–
DIOE	PCIE2・PCINT19	PCIE2・PCINT18	PCIE2・PCINT17	PCIE2・PCINT16
DIOV	0	0	0	0
DI	PCINT19入力	PCINT18入力	PCINT17入力	PCINT16入力
AIO	–	–	–	–



## 13.3.7. ポートJの交換機能

ポートJはATmega3250Pにのみ存在します。交換機能は次のとおりです。

表13-19. ポートJピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
		PJ3	PCINT27 (ピン変化割り込み27入力)
PJ6	PCINT30 (ピン変化割り込み30入力)	PJ2	PCINT26 (ピン変化割り込み26入力)
PJ5	PCINT29 (ピン変化割り込み29入力)	PJ1	PCINT25 (ピン変化割り込み25入力)
PJ4	PCINT28 (ピン変化割り込み28入力)	PJ0	PCINT24 (ピン変化割り込み24入力)

交換ピンの設定は次のとおりです。

- PCINT30 – ポートJ ビット6 : PJ6  
PCINT30 : ピン変化割り込み30入力。PJ6ピンは外部割り込み元としても扱えます。
- PCINT29 – ポートJ ビット5 : PJ5  
PCINT29 : ピン変化割り込み29入力。PJ5ピンは外部割り込み元としても扱えます。
- PCINT28 – ポートJ ビット4 : PJ4  
PCINT28 : ピン変化割り込み28入力。PJ4ピンは外部割り込み元としても扱えます。
- PCINT27 – ポートJ ビット3 : PJ3  
PCINT27 : ピン変化割り込み27入力。PJ3ピンは外部割り込み元としても扱えます。
- PCINT26 – ポートJ ビット2 : PJ2  
PCINT26 : ピン変化割り込み26入力。PJ2ピンは外部割り込み元としても扱えます。
- PCINT25 – ポートJ ビット1 : PJ1  
PCINT25 : ピン変化割り込み25入力。PJ1ピンは外部割り込み元としても扱えます。
- PCINT24 – ポートJ ビット0 : PJ0  
PCINT24 : ピン変化割り込み24入力。PJ0ピンは外部割り込み元としても扱えます。

表13-20.と表13-21.はポートJの交換機能を42頁の図13-5.で示される交換信号に関連付けます。

表13-20. ポートJ6～4の交換機能用交換信号

信号名		PJ6/PCINT30	PJ5/PCINT29	PJ4/PCINT28
PUOE		0	0	0
PUOV		0	0	0
DDOE		0	0	0
DDOV		0	0	0
PVOE		0	0	0
PVOV		0	0	0
PTOE		–	–	–
DIOE		PCIE3・PCINT30	PCIE3・PCINT29	PCIE3・PCINT28
DIOV		0	0	0
DI		PCINT30入力	PCINT29入力	PCINT28入力
AIO		–	–	–

表13-21. ポートJ3～0の交換機能用交換信号

信号名	PJ3/PCINT27	PJ2/PCINT26	PJ1/PCINT25	PJ0/PCINT24
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	–	–	–	–
DIOE	PCIE3・PCINT27	PCIE3・PCINT26	PCIE3・PCINT25	PCIE3・PCINT24
DIOV	0	0	0	0
DI	PCINT27入力	PCINT26入力	PCINT25入力	PCINT24入力
AIO	–	–	–	–

## 13.4. I/Oポート用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	BODS	BODSE	PUD	—	—	IVSEL	IVCE	MCUCR
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4 – PUD : プルアップ禁止 (Pull-up Disable)

このビットが1を書かれると、例えばDDxnとPORTxnレジスタがプルアップを許可(DDxn=0, PORTxn=1)に設定されていても、I/Oポートのプルアップは禁止されます。この特徴についてより多くの詳細に関しては40頁の「[ピンの設定](#)」をご覧ください。

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

ビット	7	6	5	4	3	2	1	0	
\$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	

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

ビット	7	6	5	4	3	2	1	0	
\$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	

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

ビット	7	6	5	4	3	2	1	0	
\$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	

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

ビット	7	6	5	4	3	2	1	0	
\$05 (\$25)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$04 (\$24)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$03 (\$23)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

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

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	PORTC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$07 (\$27)	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	PINC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

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

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$0A (\$2A)	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$09 (\$29)	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

#### 13.4.14. ポートE出力レジスタ (Port E Data Register) PORTE

ビット	7	6	5	4	3	2	1	0	
\$0E (\$2E)	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	PORTE
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	

#### 13.4.15. ポートE方向レジスタ (Port E Data Direction Register) DDRE

ビット	7	6	5	4	3	2	1	0	
\$0D (\$2D)	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	DDRE
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	

#### 13.4.16. ポートE入力レジスタ (Port E Input Address) PINE

ビット	7	6	5	4	3	2	1	0	
\$0C (\$2C)	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0	PINE
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	



## 13.4.17. ポートF出力レジスタ (Port F Data Register) PORTF

ビット	7	6	5	4	3	2	1	0	
\$11 (\$31)	PORTF7	PORTF6	PORTF5	PORTF4	PORTF3	PORTF2	PORTF1	PORTF0	PORTF
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	

## 13.4.18. ポートF方向レジスタ (Port F Data Direction Register) DDRF

ビット	7	6	5	4	3	2	1	0	
\$10 (\$30)	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0	DDRF
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	

## 13.4.19. ポートF入力レジスタ (Port F Input Address) PINF

ビット	7	6	5	4	3	2	1	0	
\$0F (\$2F)	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0	PINF
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	

## 13.4.20. ポートG出力レジスタ (Port G Data Register) PORTG

ビット	7	6	5	4	3	2	1	0	
\$14 (\$34)	—	—	—	PORTG4	PORTG3	PORTG2	PORTG1	PORTG0	PORTG
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	

## 13.4.21. ポートG方向レジスタ (Port G Data Direction Register) DDRG

ビット	7	6	5	4	3	2	1	0	
\$13 (\$33)	—	—	—	DDG4	DDG3	DDG2	DDG1	DDG0	DDRG
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	

## 13.4.22. ポートG入力レジスタ (Port G Input Address) PING

ビット	7	6	5	4	3	2	1	0	
\$12 (\$32)	—	—	PING5	PING4	PING3	PING2	PING1	PING0	PING
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	不定	不定	不定	不定	不定	

## 13.4.23. ポートH出力レジスタ (Port H Data Register) PORTH (注)

ビット	7	6	5	4	3	2	1	0	
(\$DA)	PORTH7	PORTH6	PORTH5	PORTH4	PORTH3	PORTH2	PORTH1	PORTH0	PORTH
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	

## 13.4.24. ポートH方向レジスタ (Port H Data Direction Register) DDRH (注)

ビット	7	6	5	4	3	2	1	0	
(\$D9)	DDH7	DDH6	DDH5	DDH4	DDH3	DDH2	DDH1	DDH0	DDRH
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	

注: これらのレジスタはATmega3250Pでだけ利用できます。

#### 13.4.25. ポートH入力レジスタ (Port H Input Address) PINH (注)

ビット (\$D8)	7	6	5	4	3	2	1	0	
	PINH7	PINH6	PINH5	PINH4	PINH3	PINH2	PINH1	PINH0	PINH
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	

#### 13.4.26. ポートJ出力レジスタ (Port J Data Register) PORTJ (注)

ビット (\$DD)	7	6	5	4	3	2	1	0	
	–	PORTJ6	PORTJ5	PORTJ4	PORTJ3	PORTJ2	PORTJ1	PORTJ0	PORTJ
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### 13.4.27. ポートJ方向レジスタ (Port J Data Direction Register) DDRJ (注)

ビット (\$DC)	7	6	5	4	3	2	1	0	
	–	DDJ6	DDJ5	DDJ4	DDJ3	DDJ2	DDJ1	DDJ0	DDRJ
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### 13.4.28. ポートJ入力レジスタ (Port J Input Address) PINJ (注)

ビット (\$DB)	7	6	5	4	3	2	1	0	
	–	PINJ6	PINJ5	PINJ4	PINJ3	PINJ2	PINJ1	PINJ0	PINJ
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	不定	不定	不定	不定	不定	不定	不定	

注: これらのレジスタはATmega3250Pでだけ利用できます。

## 14. 8ビット タイマ/カウンタ0 (PWM付き)

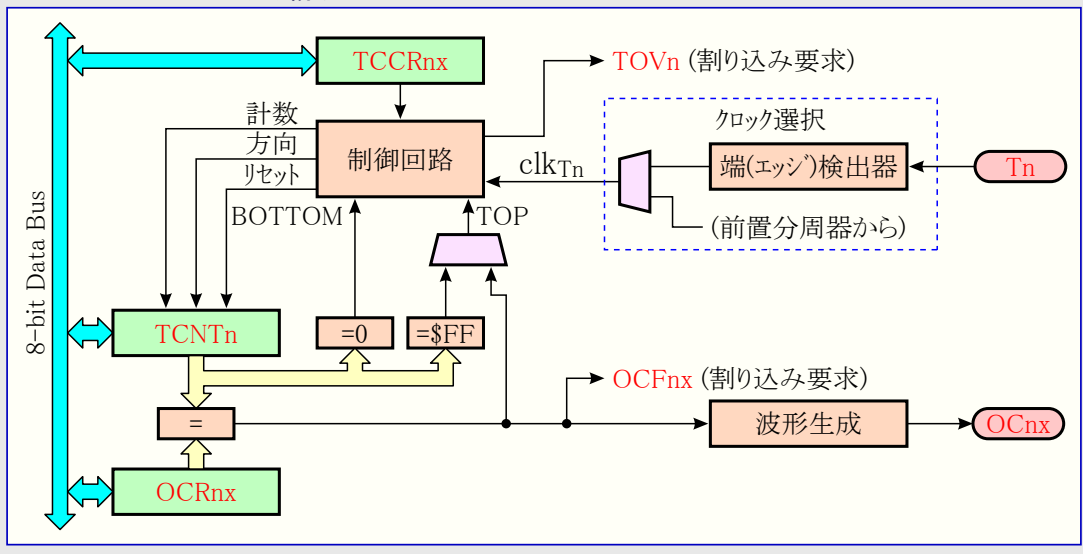
### 14.1. 特徴

- 単一比較部付き計数器
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 周波数発生器
- 外部事象計数器
- クロック用10ビット前置分周器
- 溢れと比較一致割り込み (TOV0とOCF0A)

### 14.2. 概要

タイマ/カウンタ0は単一比較部付き汎用8ビットタイマ/カウンタ部です。この8ビットタイマ/カウンタの簡易化した構成図は図14-1.で示されます。I/Oピンの実際の配置については2頁の「ピン配置」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注:原文は太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は69頁の「8ビット タイマ/カウンタ0用レジスタ」で一覧されます。

図14-1. 8ビット タイマ/カウンタ構成図



#### 14.2.1. 関係レジスタ

タイマ/カウンタ(TCNT0)と比較レジスタ(OCR0A)は8ビットのレジスタです。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR0)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK0)で個別に遮蔽(禁止)されます。TIFR0とTIMSK0はこの図で示されません。

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

2重緩衝化した比較レジスタ(OCR0A)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)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。61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)も設定(1)します。

#### 14.2.2. 定義

本章でのレジスタとビット参照の多くは一般形で書かれます。小文字の'n'はタイマ/カウンタ番号、この場合は0で置き換えます。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です。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えばタイマ/カウンタ0のカウント値のアクセスに対してのTCNT0のように)。

表14-1.の定義は本文書を通して広範囲に渡って使われます。

表14-1. 用語定義

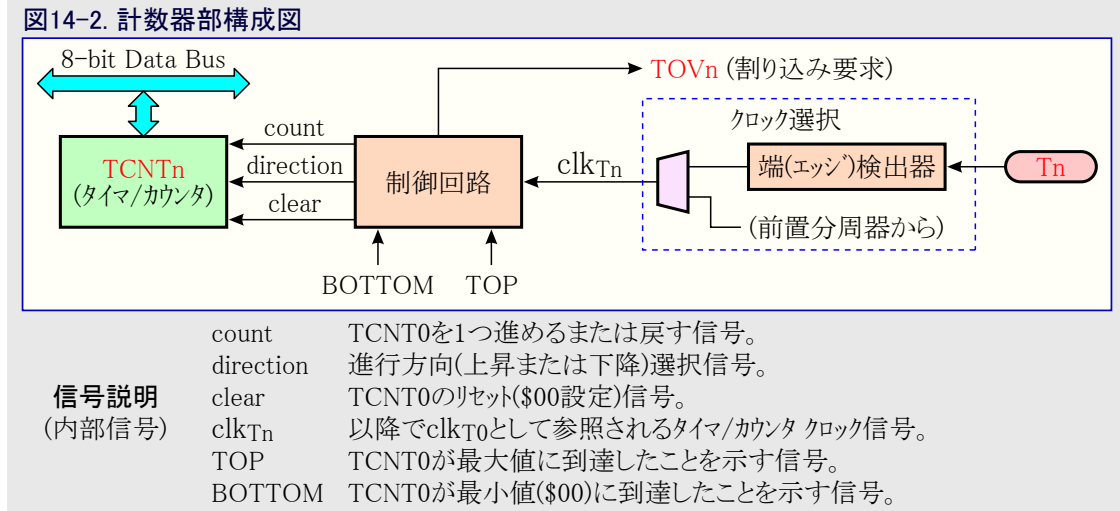
用語	意味
BOTTOM	タイマ/カウンタが\$00に到達した時。
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0A値に到達した時。この指定(TOP)値は動作種別に依存します。

### 14.3. タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ0制御レジスタ(TCCR0A)**に配置された**クロック選択(CS02~0)ビット**によって制御されるクロック選択論理回路で選ばれます。クロック元と前置分周器の詳細については67頁の「**タイマ/カウンタ0とタイマ/カウンタ1の前置分周器**」をご覧ください。

### 14.4. 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14-2は、このカウンタとその周辺環境の構成図を示します。



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

計数順序(方法)は**タイマ/カウンタ制御レジスタ(TCCR0A)**に配置された**波形生成種別(WGM01,0)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OC0A比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に関しては63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WGM01,0ビットによって選ばれた動作種別に従って設定(1)されます。TOV0はCPU割り込み発生に使えます。



## 14.5. 比較出力部

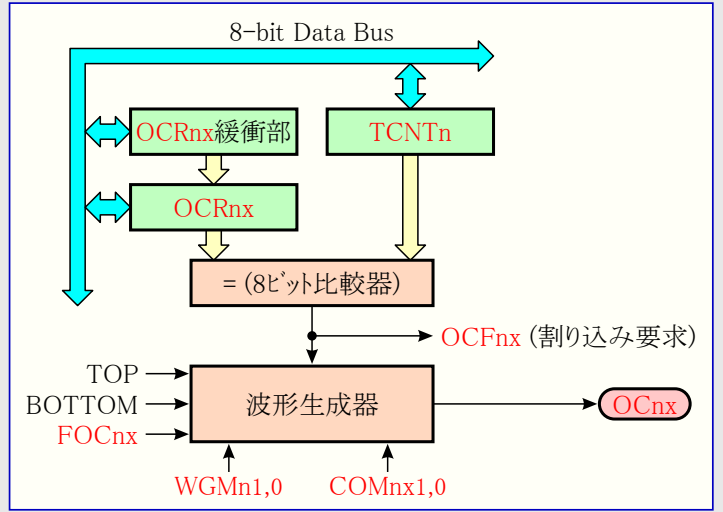
この8ビット比較器はTCNT0と比較レジスタ(OCR0A)を継続的に比較します。TCNT0とOCR0Aが等しければ、比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で**比較割り込み要求フラグ(OCF0A)**を設定(1)します。許可(I=1, OCIE0A=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF0Aは割り込みが実行されると自動的に解除(0)されます。代わりにOCF0Aはこのビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM01,0)ビット**と**比較出力選択(COM0A1,0)ビット**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(63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14-3.は比較出力部の構成図を示します。

OCR0Aはパルス幅変調(PWM)のどれかを使う時に2重緩衝化されます。標準動作と比較一致タイマ/カウンタ解除(CTC)動作については2重緩衝動作が禁止されます。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR0Aレジスタの更新を同期化します。この同期化は奇数長さや非対称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。

OCR0A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OCR0A緩衝部をアクセスし、禁止されるとOCR0Aレジスタを直接アクセスします。

図14-3. 比較出力部構成図



### 14.5.1. 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(FOC0A)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(OCF0A)**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、OC0A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(**COM0A1,0ビット**設定がOC0Aピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

### 14.5.2. 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、例えばタイマ/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OCR0Aに許します。

### 14.5.3. 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OCR0A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。

OC0A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。OC0A値を設定する一番簡単な方法は標準動作で**強制変更(FOC0A)スローブビット**を使うことです。波形生成動作種別間を変更する時であっても、OC0A(内部)レジスタはその値を保ちます。

**比較出力選択(COM0A1,0)ビット**が比較値(OCR0A)と共に2重緩衝されないことに気付いてください。COM0A1,0ビットの変更は直ちに有効となります。



## 14.7. 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(WGM01,0)ビットと比較出力選択(COM0A1,0)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。COM0A1,0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COM0A1,0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(62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66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)。

### 14.7.1. 標準動作

最も単純な動作種別が標準動作(WGM01,0=00)です。本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。

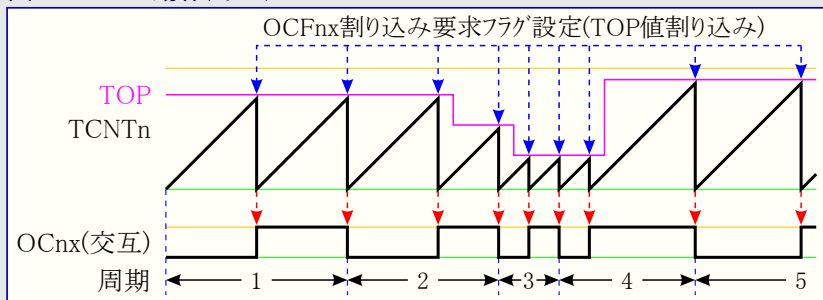
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

### 14.7.2. 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(WGM01,0=10)ではOCR0Aがカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT0)値がOCR0Aと一致すると、カウンタは\$00に解除されます。OCR0Aはカウンタに対するTOP値、従って分解能も定義します。この動作種別はより大きい比較一致出力周波数の制御を許します。それは外部の出来事の計数操作も簡単にします。

CTC動作についてのタイミング図は図14-5.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、そしてその後にカウンタ(TCNT0)は解除(\$00)されます。

図14-5. CTC動作タイミング



注: COMnx1,0=01

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

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

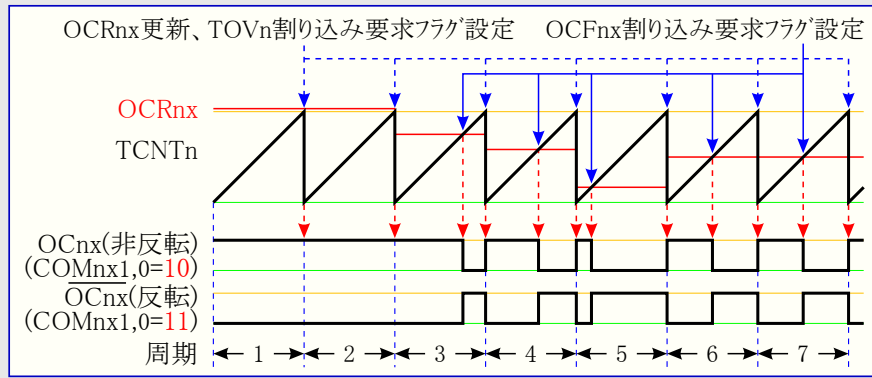
標準動作と同じように、タイマ/カウンタ溢れ(TOV0)フラグはカウンタがMAXから\$00へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

### 14.7.3. 高速PWM動作

高速パルス幅変調(PWM)動作(WGM01,0=11)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MAX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(COM0A1,0=10)での比較出力(OC0A)はTCNT0とOCR0A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(COM0A1,0=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MAX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14-6.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。細い赤線はOCR0A値を示し、TCNT0値との交点(接点)がTCNT0とOCR0A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(OCF0A)はOCR0A=MAX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)。

図14-6. 高速PWM動作タイミング



タイマ/カウンタ溢れ(TOV0)フラグはカウンタがMAX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OC0AピンでのPWM波形の生成を許します。COM0A1,0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COM0A1,0を'11'に設定することで生成できます(69頁の表14-4.をご覧ください)。実際のOC0A値はポートピンに対するデータ方向(DDR\_OC0A)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OCR0A間の比較一致でOC0A(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、MAXからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0A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。

$$f_{OCnxPWM} = \frac{f_{clk\_I/O}}{N \times 256}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

OCR0A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。OCR0AがBOTTOM(\$00)と等しく設定されると、出力はMAX+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。OCR0AがMAXに等しく設定されると、(COM0A1,0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。

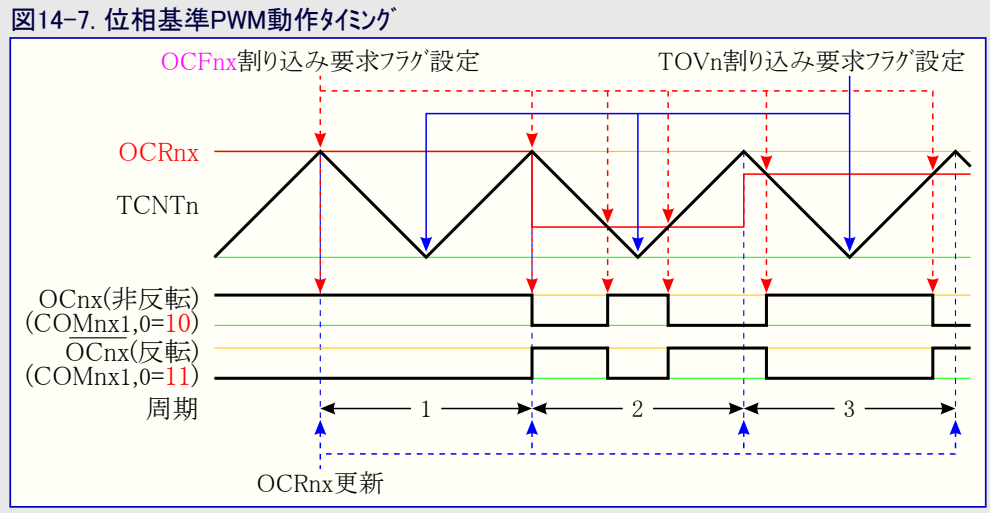
(訳注:以下は本動作種別がTOP=MAX(\$FF)で固定のため基本的に不適切です。) 高速PWM動作で(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC0A設定(COM0A1,0=01)によって達成できます。生成された波形はOCR0Aが0(\$00)に設定される時に $f_{OC0A} = f_{clk\_I/O} / 2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いて、CTC動作でのOC0A交互出力(COM0A1,0=01)と同じです。



## 14.7.4. 位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM01,0=01)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MAXへそして次にMAXからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM0A1,0=10)での比較出力(OC0A)は上昇計数中のTCNT0とOCR0A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(COM0A1,0=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動作のPWM分解能は8ビットに固定されます。位相基準PWM動作でのカウンタはカウンタ値がMAXと一致するまで増加されます。カウンタはMAX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MAXと等しくなります。位相基準PWM動作のタイミング図は図14-7で示されます。TCNT0値はタイミング図に於いて両傾斜動作(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OCR0A値を示し、TCNT0値との交点(接点)がTCNT0とOCR0A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OC0AピンでのPWM波形の生成を許します。COM0A1,0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COM0A1,0ビットを'11'に設定することで生成できます(69頁の表14-5をご覧ください)。実際のOC0A値はそのポートピンに対するデータ方向(DDR\_OC0A)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OCR0A間の比較一致でOC0A(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OCR0A間の比較一致でOC0A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。

$$f_{OCnxPCPWM} = \frac{f_{clk\_I/O}}{N \times 510}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

OCR0A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あたります。非反転PWM動作ではOCR0AがBOTTOM(\$00)に等しく設定されると出力は定常的なLow、MAX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14-7.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14-7.のようにOCR0AはMAXからその値を変更します。OCR0A値がMAX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)MAX(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OCR0A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、MAX位置で(直前がHならば)H→L遷移が生じます。)



## 14.8. 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( $\text{clk}_{T0}$ )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14-8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14-8. 前置分周なし(1/1)のタイマ/カウンタ タイミング

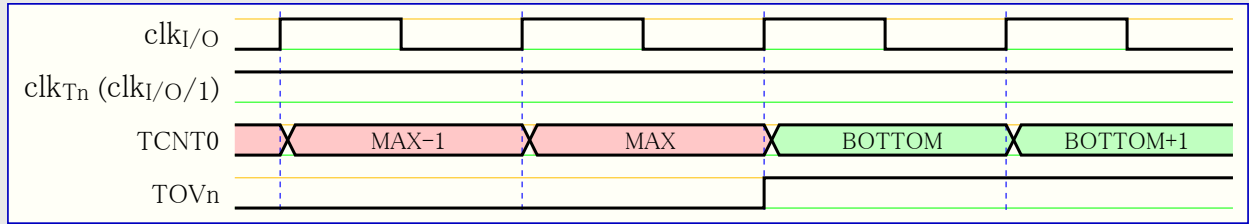


図14-9. は同じタイミング データを示しますが、前置分周器が許可されています。

図14-9. 前置分周器( $f_{\text{clk}_{I/O}/8}$ )のタイマ/カウンタ タイミング

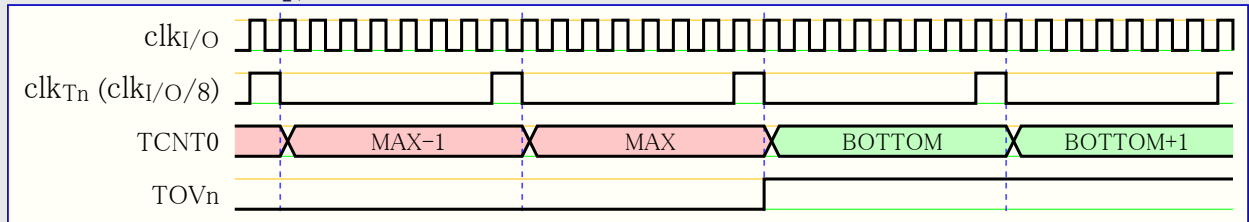


図14-10. はCTC動作を除く全ての動作種別でのOCF0Aの設定を示します。

図14-10. 前置分周器( $f_{\text{clk}_{I/O}/8}$ )のタイマ/カウンタ、OCF0A設定 タイミング

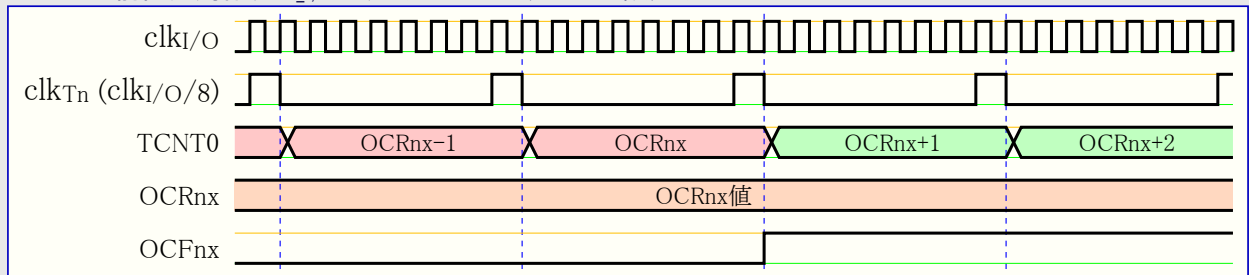
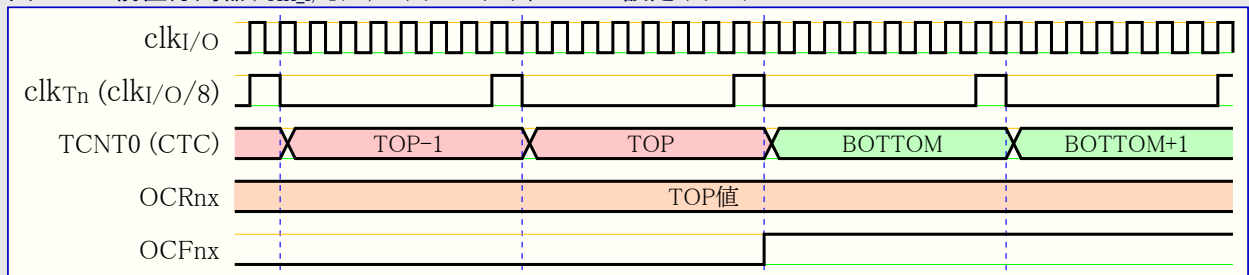


図14-11. はCTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図14-11. 前置分周器( $f_{\text{clk}_{I/O}/8}$ )のタイマ/カウンタ、OCF0A設定 タイミング



## 14.9. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器

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

### 14.9.1. 内部クロック元

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

### 14.9.2. 前置分周器リセット

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

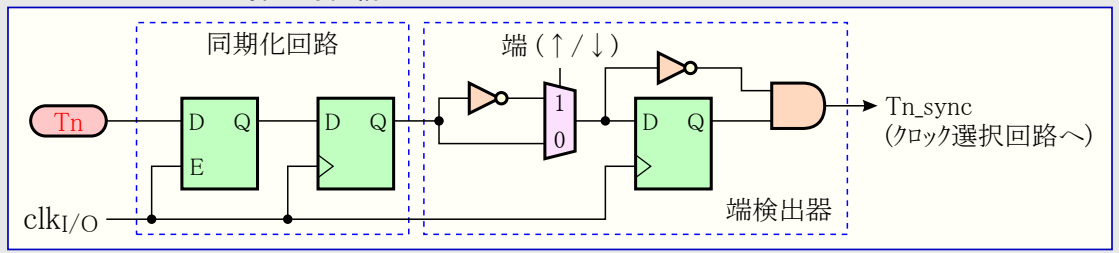
プログラム実行にタイマ/カウンタを同期することに対して前置分周器リセットを使うことが可能です。しかし、同じ前置分周器を共有する他のタイマ/カウンタも前置分周を使う場合、注意が必要とされなければなりません。前置分周器リセットはそれが接続される全タイマ/カウンタについての前置分周器周期に影響を及ぼします。

### 14.9.3. 外部クロック元

T0/T1ピンに印加された外部クロック元はタイマ/カウンタクロック( $f_{clk\_T0}/f_{clk\_T1}$ )として使えます。このT0/T1ピンはピン同期化論理回路によって全てのシステムクロック周期に一度採取されます。この同期化(採取)された信号はその後端(エッジ)検出器を通して通過されます。図14-12はT0/T1同期化と端検出器論理回路の機能等価構成図を示します。レジスタは内部システムクロック( $f_{clk\_I/O}$ )の上昇端でクロック駆動されます。ラッチは内部システムクロックのHigh区間で通過(Low区間で保持)です。

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

図14-12. T0/T1ピンの採取等価構成図



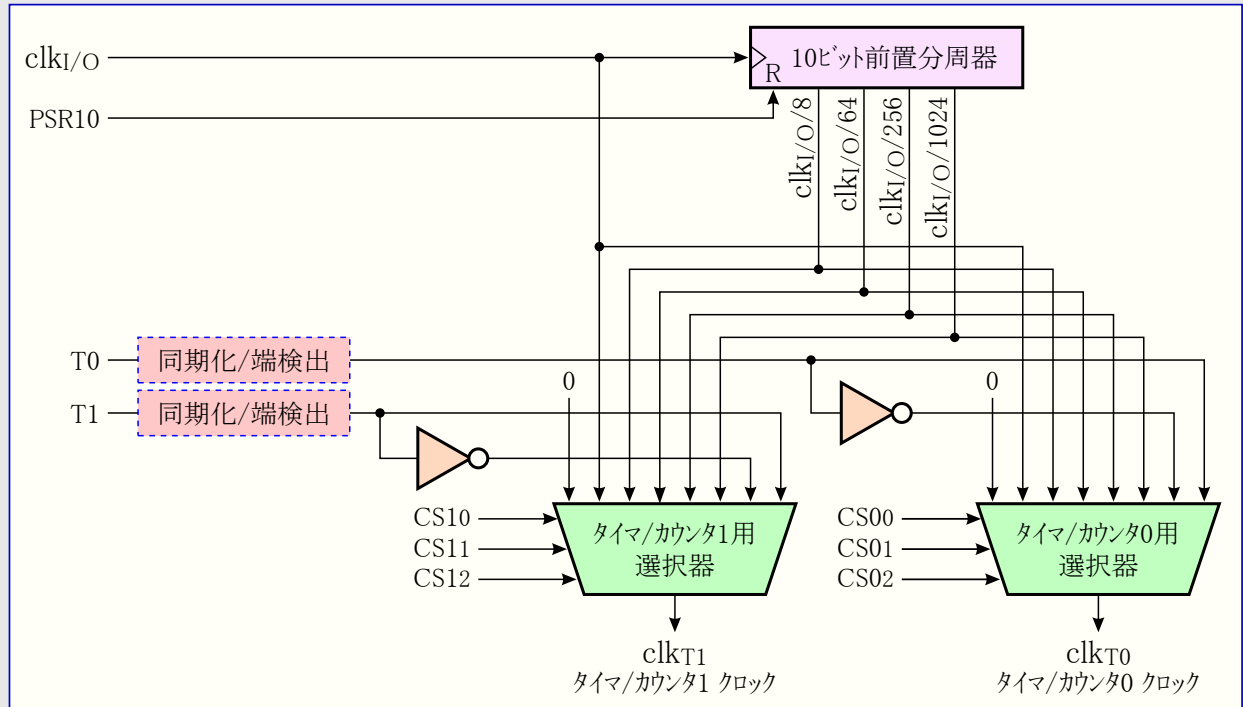
同期化と端検出器論理回路はT0/T1ピンへ印加された端から計数器が更新されるまでに2.5~3.5システムクロック周期の遅延をもたらします。

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

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

外部クロック元は前置分周できません。

図14-13. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器部構成図



注: 入力ピン(T0/T1)の同期化/端(エッジ)検出論理回路は図14-12.で示されます。

## 14.10. 8ビット タイマ/カウンタ0用レジスタ

### 14.10.1. タイマ/カウンタ0制御レジスタA (Timer/Counter0 Control Register A) TCCR0A

ビット	7	6	5	4	3	2	1	0	
\$24 (\$44)	FOC0A	WGM00	COM0A1	COM0A0	WGM01	CS02	CS01	CS00	TCCR0A
Read/Write	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 – FOC0A : OC0A強制変更 (Force Output Compare 0A)

FOC0AビットはWGM01,0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TCCR0A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ビットがスト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スト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3,6 – WGM01,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)

これらのビットはカウンタの計数順序(方向)、最大(TOP)カウンタ値の供給元、使われるべき波形生成のどの形式かを制御します。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、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。表14-2と63頁の「動作種別」をご覧ください。

表14-2. 波形生成種別選択

番号	WGM01 (CTC0)	WGM00 (PWM0)	タイマ/カウンタ動作種別	TOP値	OCR0A 更新時	TOV0 設定時
0	0	0	標準動作	\$FF	即時	MAX
1	0	1	位相基準PWM動作	\$FF	TOP	BOTTOM
2	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR0A	即時	MAX
3	1	1	高速PWM動作	\$FF	BOTTOM	MAX

注: CTC0とPWM0ビット定義名は旧名称です。WGM01,0定義を使ってください。しかし、これらのビットの機能と位置は旧版のタイマ/カウンタと一致します。

#### • ビット5,4 – COM0A1,0 : 比較0A出力選択 (Compare Match 0A 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WGM01,0ビット設定に依存します。

表14-3はWGM01,0ビットが標準動作またはCTC動作(即ちPWM以外)に設定される時のCOM0A1,0ビット機能を示します。

表14-4はWGM01,0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14-5はWGM01,0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14-3. 非PWM動作での比較出力選択

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14-4. 高速PWM動作での比較出力選択

COM0A1	COM0A0	意味
0	0	標準ポート動作 (OC0A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

注: COM0A1が設定(1)され、OCR0AがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、TOPでの設定(1)や解除(0)は行われます。より多くの詳細については65頁の「位相基準PWM動作」をご覧ください。

表14-4. 高速PWM動作での比較出力選択

COM0A1	COM0A0	意味
0	0	標準ポート動作 (OC0A切断)
0	1	(予約)
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)

注: COM0A1が設定(1)され、OCR0A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)や解除(0)は行われます。より多くの詳細については64頁の「高速PWM動作」をご覧ください。

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

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

表14-6. 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、例えば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。

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

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OCR0A間の比較一致消失の危険を誘発します。

#### 14.10.3. タイマ/カウンタ0 比較レジスタ (Timer/Counter0 Output Compare A Register) OCR0A

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。



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

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

### • ビット1 – OCIE0A : タイマ/カウンタ0比較A割り込み許可 (Timer/Counter0 Output Compare Match A Interrupt Enable)

OCIE0Aビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが許可されます。タイマ/カウンタ0で比較一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)で比較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/Counter0 Overflow Interrupt Enable)

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

## 14.10.5. 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) TIFR0

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

### • ビット1 – OCF0A : タイマ/カウンタ0比較A割り込み要求フラグ (Timer/Counter0, Output Compare A Match Flag)

OCF0Aビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0A)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF0Aはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0Aは解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ0割り込み許可レジスタ(TIMSK0)のタイマ/カウンタ0比較一致割り込み許可(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/Counter0 Overflow Flag)

TOV0ビットはタイマ/カウンタ(TCNT0)溢れが起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、TOV0はハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもTOV0は解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ0割り込み許可レジスタ(TIMSK0)のタイマ/カウンタ0溢れ割り込み許可(TOIE0)ビット、TOV0が設定(1)されると、タイマ/カウンタ0溢れ割り込みが実行されます。位相基準PWM動作ではタイマ/カウンタ0が\$00で計数方向を変える時にこのビットが設定(1)されます。

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

ビット \$23 (\$43)	7	6	5	4	3	2	1	0	
	TSM	–	–	–	–	–	PSR2	PSR10	GTCCR
Read/Write	R/W	R	R	R	R	R	R/W	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ビットに1を書くことがタイマ/カウンタ同期化動作を活性(有効)にします。この動作でPSR2とPSR10へ書かれる値は保持され、従って対応する前置分周器リセット信号の有効を保持します。これは対応するタイマ/カウンタを停止し、設定中にそれらの1つが進行する危険なしに同じ値に設定できるのを保証します。TSMビットが0を書かれると、PSR2とPSR10ビットはハードウェアによって解除(0)され、同時にタイマ/カウンタが計数を始めます。

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

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

## 15. 16ビット タイマ/カウンタ1

### 15.1. 特徴

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

### 15.2. 概要

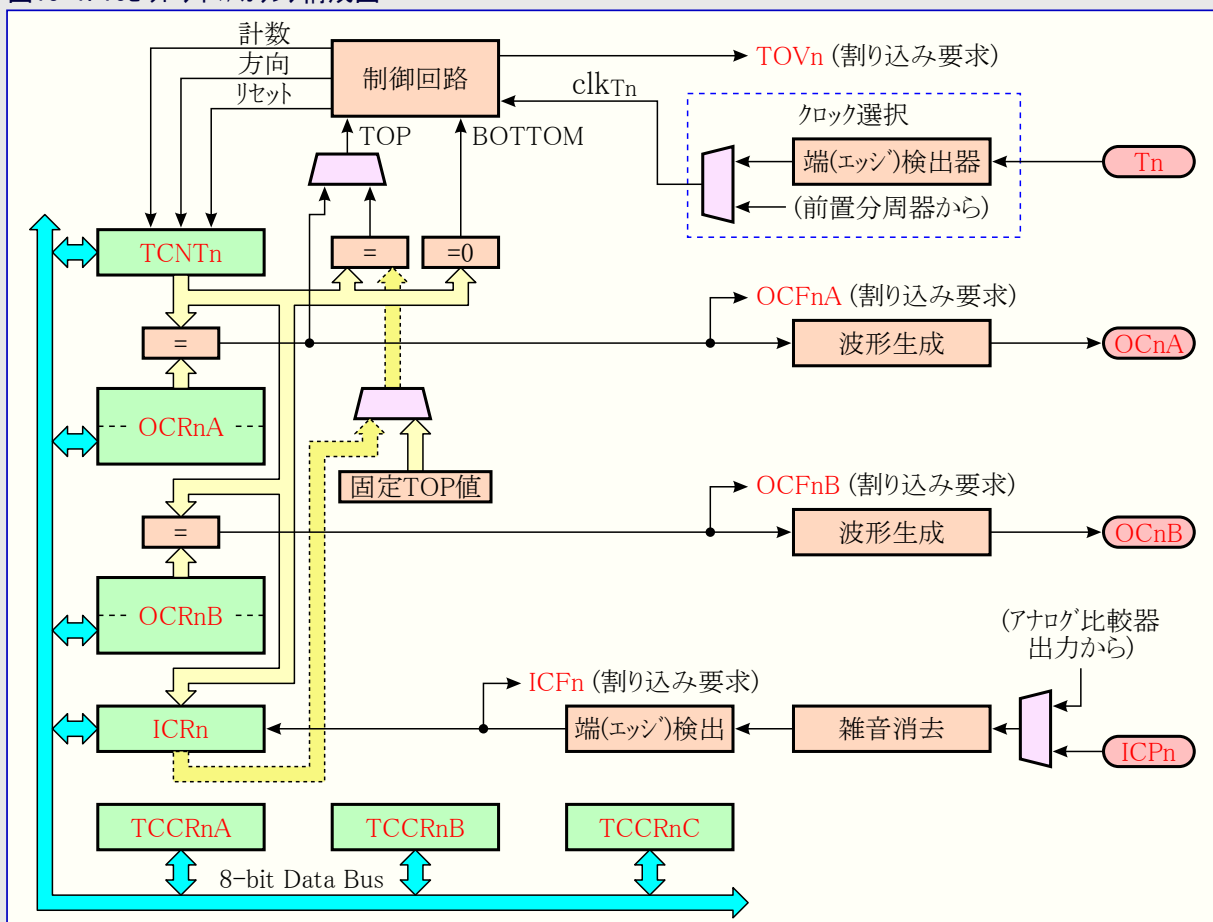
この16ビット タイマ/カウンタ部は正確なプログラム実行タイミング(事象管理)、波形生成、信号タイミング計測を許します。

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

この16ビット タイマ/カウンタの簡略化した構成図は図15-1.で示されます。実際のI/Oピンの配置については2頁の「ピン配置」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注: 原文太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は86頁の「16ビット タイマ/カウンタ用レジスタ」で示されます。

26頁の「電力削減レジスタ(PRR)」のPRTIM1ビットはタイマ/カウンタ1部を許可するために0を書かれなければなりません。

図15-1. 16ビット タイマ/カウンタ構成図



注: タイマ/カウンタ1のピンの配置と記述については2頁の「ピン配置」、43頁の表13-3、45頁の表13-6、50頁の表13-14.を参照してください。

## 15.2.1. 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74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端(エッジ)を制御します。クロック元が選ばれないと、タイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkT1)として参照されます。

2重緩衝化した比較レジスタ(OCR1A,OCR1B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC1A,OC1B)ピンでPWMまたは可変周波数出力を生成するための波形生成器によって使えます。78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(136頁の「アナログ比較器」参照)のどちらかの外部(端で起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波器(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出力生成用に使いません。けれどもこの場合、TOP値は動作中に変更されるのをTOP値に許す2重緩衝化します。固定的なTOP値が必要とされる場合、ICR1が代わりに使い、PWM出力として使われるべきOCR1Aを開放します。

## 15.2.2. 定義

次の定義は本文書を通して広範囲に使われます。

表15-1. 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。

## 15.2.3. 互換性

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

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

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

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

次の制御ビットが16ビットタイマ/カウンタ制御レジスタ内に追加されます。

- FOC1AとFOC1Bが(新設の)TCCR1Cに追加。
- WGM13がTCCR1Bに追加。

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

### 15.3. 16ビットレジスタのアクセス

TCNT1, OCR1A, OCR1B, ICRI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, ICRIレジスタのアクセスに対して同じ原理が直接的に使えます。C言語を使う時はコンパイラが16ビットアクセスを扱うことに注意してください。

#### アセンブリ言語プログラム例

```

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

```

#### C言語プログラム例

```

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

```

**注:** 6頁の「コード例について」をご覧ください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

16ビットレジスタアクセスが非分断操作である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 */
    __disable_interrupt();             /* 全割り込み禁止 */
    i = TCNT1;                         /* TCNT1値を取得 */
    SREG = sreg;                      /* 全割り込み許可フラグ(I)を復帰 */
    return i;                          /* TCNT1値で呼び出し元へ復帰 */
}
```

**注:** 6頁の「コード例について」をご覧ください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

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

## アセンブリ言語プログラム例

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

## C言語プログラム例

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

**注:** 6頁の「コード例について」をご覧ください。

アセンブリ言語コード例はR17:R16レジスタ対がTCNT1へ書かれるべき値を含む必要があります。

### 15.3.1. 上位バイト一時レジスタの再使用

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



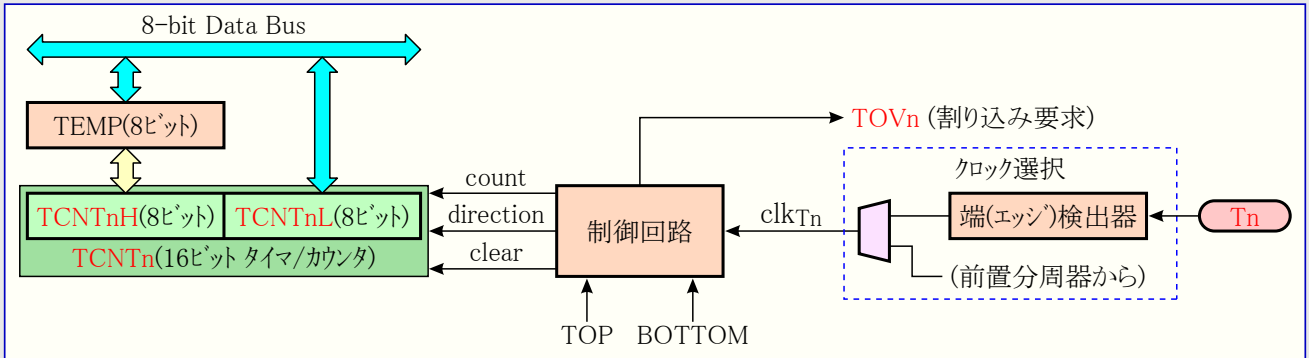
## 15.4. タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**クロック選択(CS12~0)**ビットによって制御されるクロック選択論理回路で選ばれます。クロック元と前置分周器の詳細については67頁の「**タイマ/カウンタ0とタイマ/カウンタ1の前置分周器**」をご覧ください。

## 15.5. 計数器部

16ビット タイマ/カウンタの主な部分は設定可能な16ビット双方向カウンタ部です。図15-2はこのカウンタとその周辺の構成図を示します。

図15-2. 計数器部構成図



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

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

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

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81頁の「**動作種別**」をご覧ください。

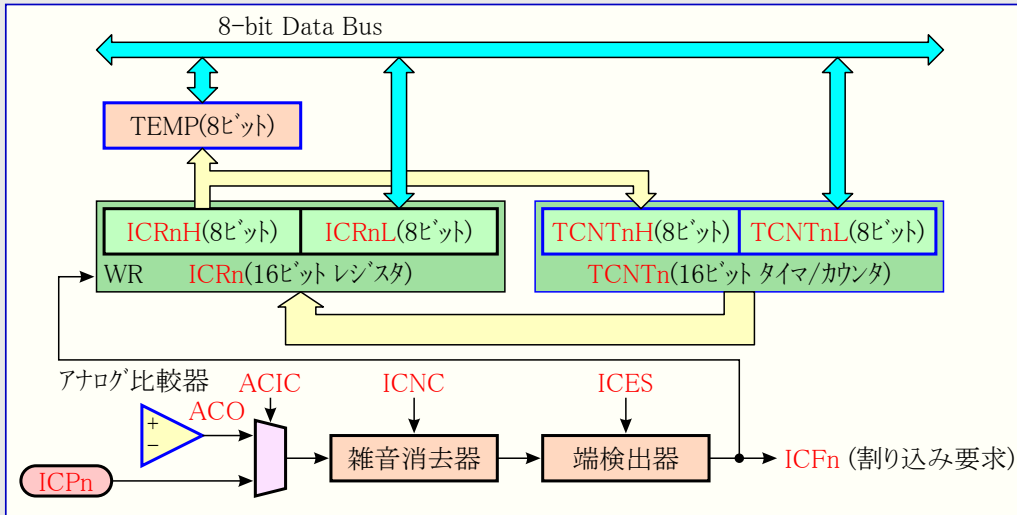
**タイマ/カウンタ溢れ(TOV1)フラグ**はWGM13~0ビットによって選ばれた動作種別に従って設定(1)されます。TOV1はCPU割り込み発生に使えます。

## 15.6. 捕獲入力部

タイマ/カウンタは外部の出来事を捕獲でき、発生時間を示す時間印(タイマ/カウンタ値)を与える捕獲入力部と合体します。出来事または複数の出来事示す外部信号は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15-3.で示される構成図によって図解されます。直接的な捕獲入力部の部分でない構成図の要素は青枠(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15-3. 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(ACSR)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、その変化が端(エッジ)検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF1)はTCNT1値がICR1に複写される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、この一時レジスタにアクセスします。

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

16ビットレジスタアクセス法のより多くの情報については74頁の「16ビットレジスタのアクセス」を参照してください。

### 15.6.1. 捕獲起動元

捕獲入力部用の主な起動元は捕獲起動入力(ICP1)ピンです。タイマ/カウンタ1は捕獲入力部用起動元としてアナログ比較器出力を代わりに使えます。アナログ比較器はアナログ比較器制御/状態レジスタ(ACSR)のアナログ比較器捕獲起動許可(ACIC)ビットの設定(1)によって起動元として選ばれます。起動元を変更することが捕獲を起動し得ることに気付いてください。従って捕獲割り込み要求フラグ(ICF1)は、その変更後に解除(0)されなければなりません。

捕獲起動入力(ICP1)ピンとアナログ比較器出力(ACO)の両入力、T1ピン(67頁の図14-12参照)についてと同じ技法を使って採取されます。端検出器も全く同じです。けれども雑音消去が許可されると、付加論理回路が端検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されないなら、雑音消去器と端検出器の入力が常に許可されることに注意してください。

捕獲入力はICP1ピンのポートを制御することによってソフトウェアで起動できます。

### 15.6.2. 雑音消去器

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

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

### 15.6.3. 捕獲入力の使用

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

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

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

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

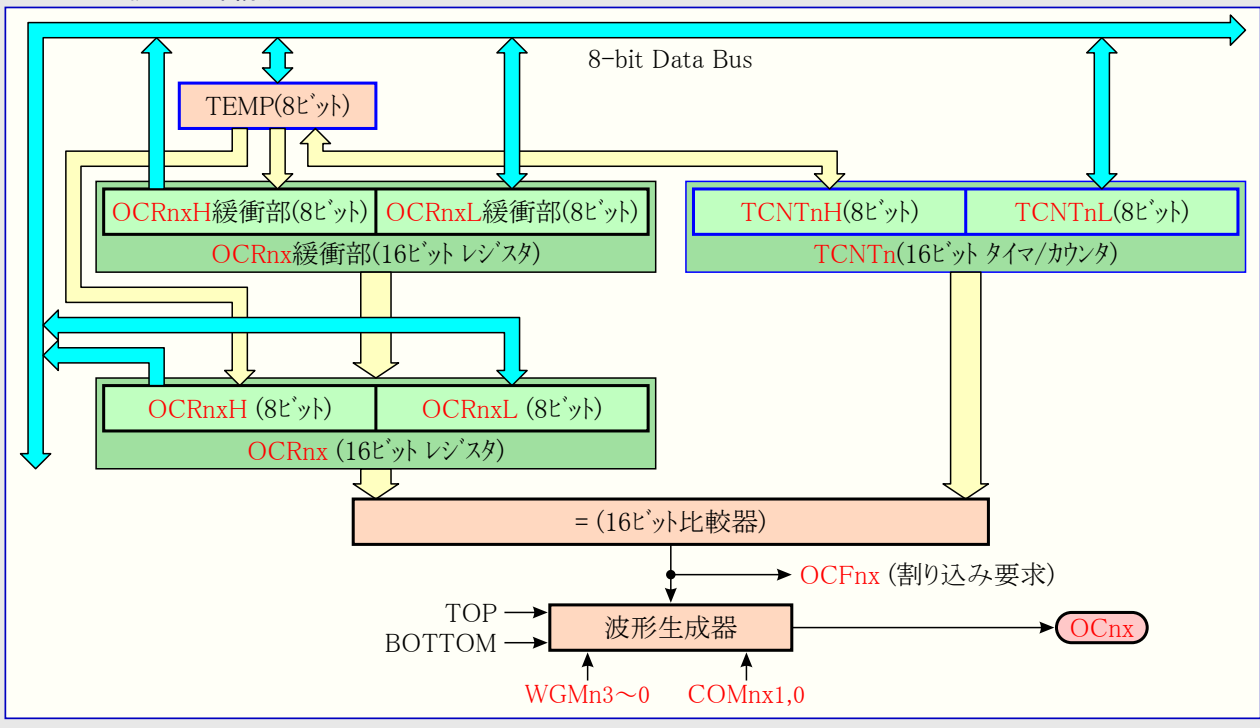
### 15.7. 比較出力部

この16ビット比較器は**TCNT1**と**比較レジスタ(OCR1x)**を継続的に比較します。TCNT1とOCR1xが等しければ、比較器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(81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15-4.は比較出力部の構成図を示します。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対してはn=1)、小文字の'x'は比較出力部(AまたはB)を表します。直接的な比較出力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。

図15-4. 比較出力部構成図



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

OCR1xのアクセスは複雑なように思えますが決してそんなことはありません。2重緩衝動作が許可されるとCPUはOCR1x緩衝部をアクセスし、禁止されるとOCR1xレジスタを直接アクセスします。OCR1x(緩衝部またはレジスタ)の内容は書き込み操作によってのみ変更されます(タイマ/カウンタはTCNT1やICR1のようにOCR1xを自動的に更新しません)。従ってOCR1xは上位バイト一時レジスタ(TEMP)経由で読まれません。けれども他の16ビットレジスタをアクセスする時のように下位バイトを先に読むのは良い習慣です。OCR1x書き込みは16ビット全ての比較が継続的に行われるため、一時レジスタ経由で行われなければなりません。上位バイト(OCR1xH)が先に書かれなければなりません。上位バイトI/O位置がCPUによって書かれると、一時レジスタは書かれた値で更新されます。その後に下位バイト(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74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。

## 15.7.1. 強制比較出力

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

## 15.7.2. TCNT1書き込みによる比較一致妨害

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

## 15.7.3. 比較一致部の使用

どの動作種別でのTCNT1書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、何れかの比較出力部を使う場合、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。

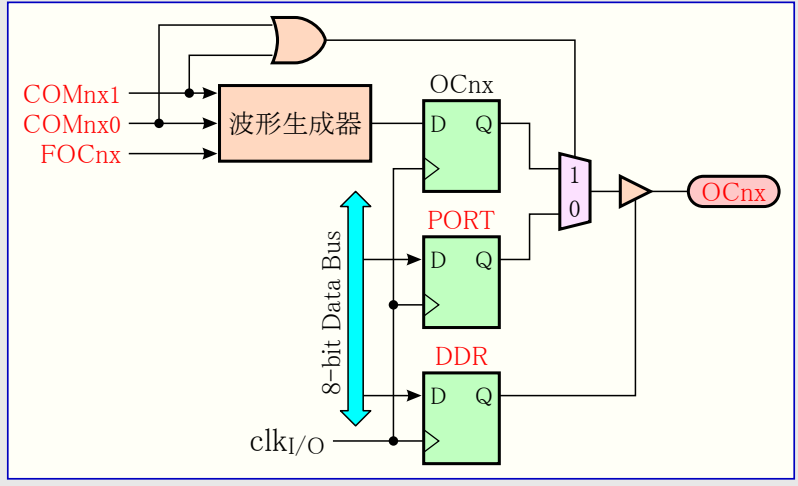
比較出力選択(COM1x1,0)ビットが比較値(OCR1x)と共に2重緩衝されないことに気付いてください。COM1x1,0ビットの変更は直ちに有効となります。



## 15.8. 比較一致出力部

比較出力選択( $COM1x1,0$ )ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力( $OC1x$ )状態の定義に $COM1x1,0$ ビットを使います。次に $COM1x1,0$ ビットは $OC1x$ ピン出力元を制御します。図15-5は $COM1x1,0$ ビット設定によって影響される論理回路の簡略化した図を示します。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。

図15-5. 比較一致出力回路図



$COM1x1,0$ 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15-2、表15-3、表15-4を参照してください。

比較出力ピン論理回路の設計は出力が許可される前の $OC1x$ 状態の初期化を許します。いくつかの $COM1x1,0$ ビット設定が或る種の動作種別に対して予約されることに注意してください。86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。

### 15.8.1. 比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作で $COM1x1,0$ ビットを違うふうに使います。全ての動作種別に対して $COM1x1,0=00$ 設定は次の比較一致で実行すべき $OC1x$ レジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については86頁の表15-2を参照してください。高速PWM動作については86頁の表15-3、位相基準PWMと位相/周波数基準PWMについては86頁の表15-4を参照してください。

$COM1x1,0$ ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更( $FOC1x$ )ス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。



## 15.9. 動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**の組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM1x1,0ビットは生成されたPWM出力が反転されるべきか否か(反転または非反転PWM)のどちらかを制御します。非PWM動作に対してのCOM1x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します。80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85頁の「**タイマ/カウンタのタイミング**」を参照してください。

### 15.9.1. 標準動作

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

捕獲入力部は標準動作での使用が容易です。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。

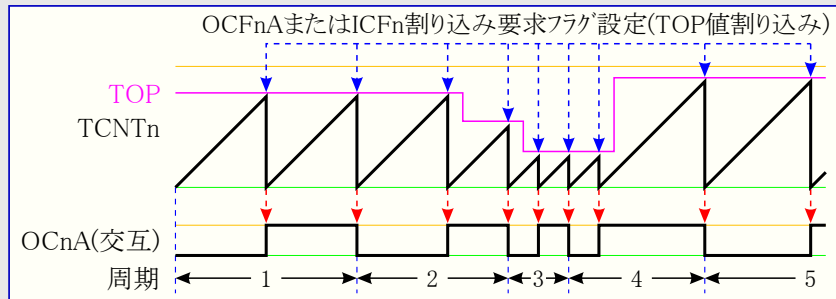
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

### 15.9.2. 比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM13~0=0100または1100)では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15-6.で示されます。カウンタ(TCNT1)値はOCR1AまたはICR1のどちらかで比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT1)は解除(\$0000)されます。

図15-6. CTC動作タイミング



注: COMnA1,0=01

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

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

### 15.9.3. 高速PWM動作

高速パルス幅変調(PWM)動作(WGM13~0=0101,0110,0111,1110,1111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)はTCNT1とOCR1x間の比較一致で解除(0)され、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。

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15-7で示されます。本図はOCR1AかICR1がTOPを定義するのに使われる時の高速PWM動作を示します。TCNT1値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。本図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

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

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

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

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

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

高速PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます。86頁の表15-3をご覧ください。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合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。

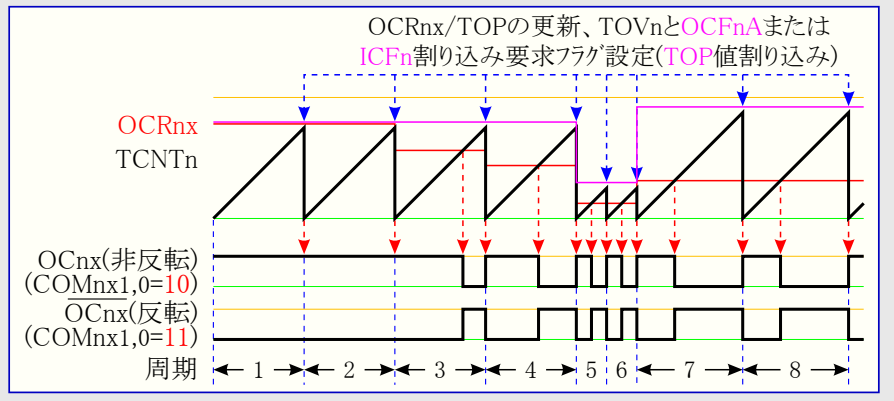
PWM出力周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

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

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)定常的なLowまたはHigh出力に終わるでしょう。

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

図15-7. 高速PWM動作タイミング



## 15.9.4. 位相基準PWM動作

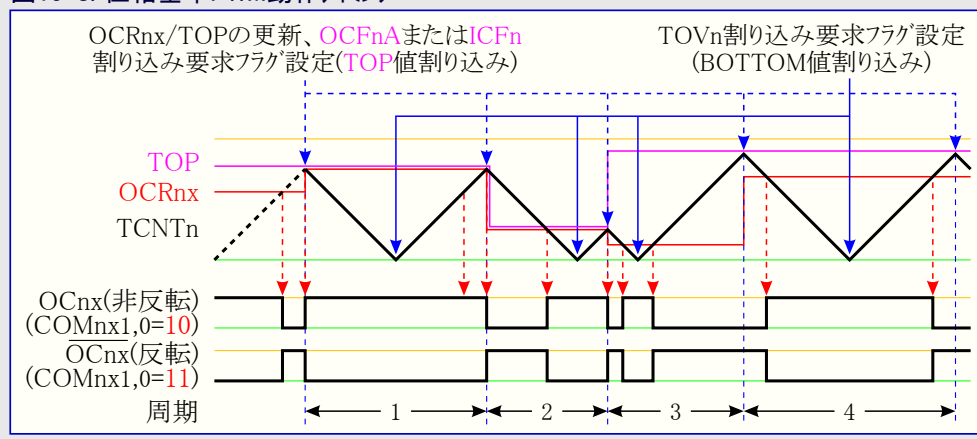
位相基準パルス幅変調(PWM)動作(WGM13~0=0001,0010,0011,1010,1011)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とした位相/周波数基準PWMと似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾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

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

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

位相基準PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0001,0010,0011)、ICR1値(WGM13~0=1010)またはOCR1A値(WGM13~0=1011)のどれかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値はタイマ/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15-8.で示されます。この図はOCR1AかICR1がTOPを定義するのに使われる時の位相基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

図15-8. 位相基準PWM動作タイミング



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15-8.で示される第3周期が図解するように、タイマ/カウンタが位相基準PWM動作で走行中にTOPを積極的に変更するのは、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(86頁の表15-4.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合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。

$$f_{OCnxPCPWM} = \frac{f_{clk1/O}}{2 \times N \times TOP}$$

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。



### 15.9.5. 位相/周波数基準PWM動作

位相/周波数基準パルス幅変調(PWM)動作(WGM13~0=1000,1001)は高分解能で正しい位相と周波数のPWM波形生成選択を提供します。位相/周波数基準PWM動作は両傾斜(三角波)動作を基準とした**位相基準PWM**と似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後に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。

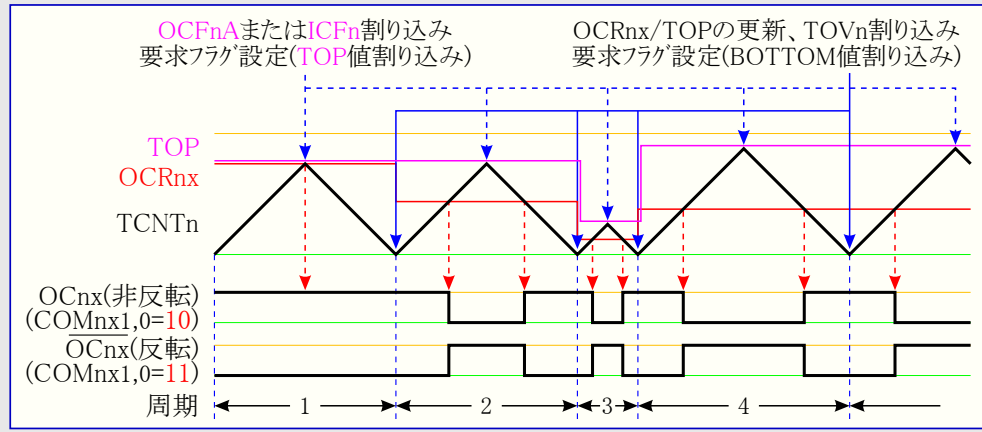
位相基準と位相/周波数基準PWM動作間の主な違いはOCR1xレジスタがOCR1x緩衝部によって更新される時(訳補:TOPとBOTTOM)です(図15-8と図15-9参照)。

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

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

位相/周波数基準PWM動作でのタイマ/カウンタはタイマ/カウンタ値がICR1値(WGM13~0=1000)かOCR1A値(WGM13~0=1001)のどちらかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期間、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15-9で示されます。この図はOCR1AかICR1がTOPを定義するのに使われる時の位相/周波数基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

図15-9. 位相/周波数基準PWM動作タイミング



タイマ/カウンタ溢れ(TOV1)フラグはOCR1xレジスタが(BOTTOMに於いて)2重緩衝値で更新されるのと同じタイマ/カウンタクロック周期で周期(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15-9が示すように、生成された出力は位相基準PWM動作と異なり、全ての周期で対称です。OCR1xレジスタがBOTTOMで更新されるため、上昇と下降の傾斜長は常に等しくなります。これが対称出力パルス、従って正しい周波数を与えます。

TOPを定義するのにICR1を使うことは決まったTOP値を使う時に上手くなります。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(86頁の表15-4をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

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

$$f_{OCnxPFCPWM} = \frac{f_{clk\_1/O}}{2 \times N \times TOP}$$

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

## 15.10. タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従ってタイマ/カウンタクロック( $clk_{T1}$ )が下図のクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時、そして $OCR1x$ レジスタが $OCR1x$ 緩衝値で更新される時(2重緩衝部を使う動作種別のみ)の情報を含まず。図15-10は $OCF1x$ の設定についてのタイミング図を示します。

図15-10. 前置分周なし(1/1)のタイマ/カウンタ、 $OCF1x$ 設定 タイミング

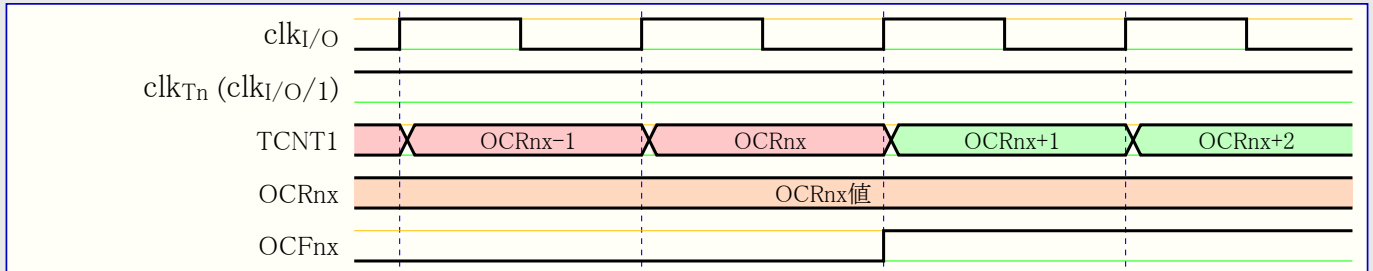


図15-11. は同じタイミング データを示しますが、前置分周器が許可されています。

図15-11. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、 $OCF1x$ 設定 タイミング

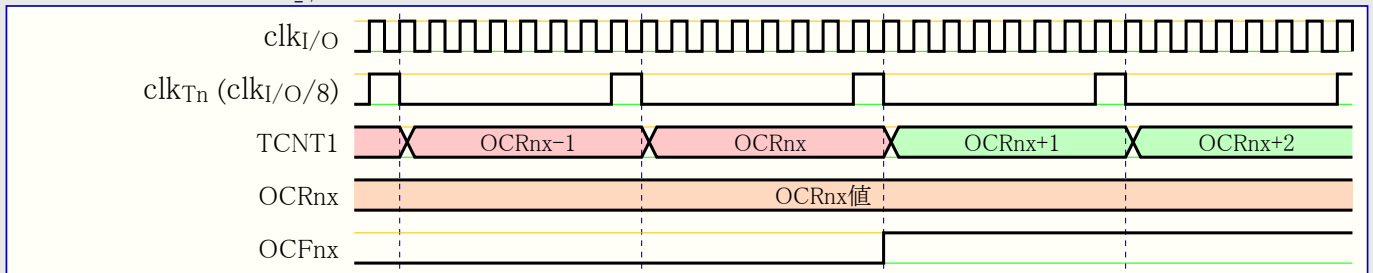


図15-12. は各動作種別でのTOP近辺の計数手順を示します。位相/周波数基準PWM動作使用時の $OCR_{nx}$ レジスタはBOTTOMで更新されます。タイミング図は同じになりますが、当然TOPはBOTTOMで、TOP-1はBOTTOM+1でなどのように置き換えられます。BOTTOMで $TOV1$ を設定(1)する動作種別についても、同様な名称変更が適用されます。

図15-12. 前置分周なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング

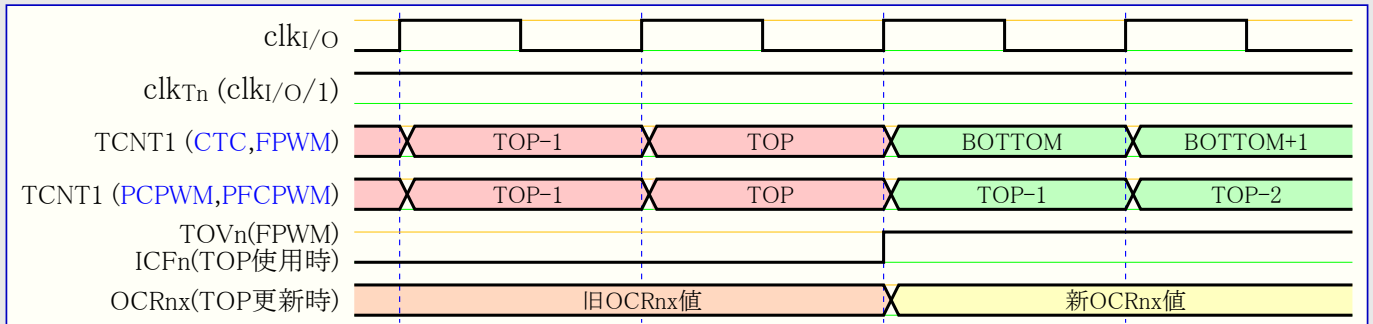
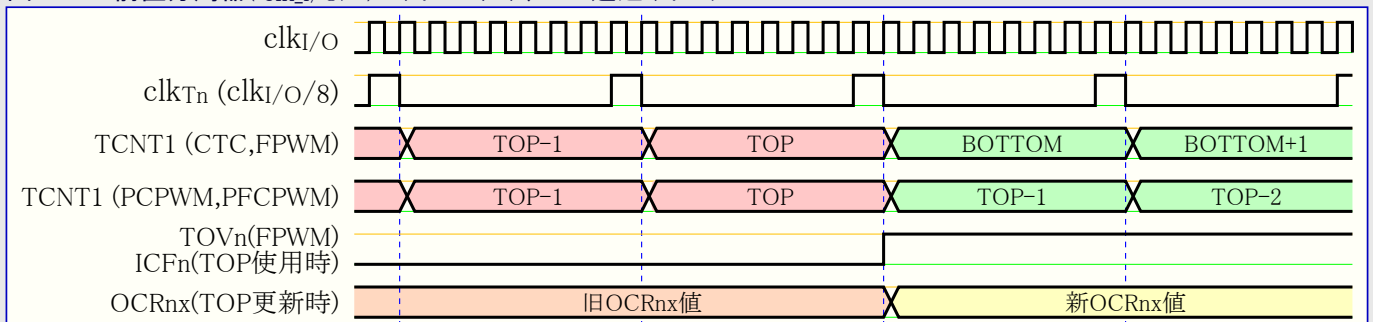


図15-13. は同じタイミング データを示しますが、前置分周器が許可されています。

図15-13. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、TOP近辺 タイミング





## 15.11. 16ビット タイマ/カウンタ1用レジスタ

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

ビット (\$80)	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	–	–	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

OC1AまたはOC1Bがピンに接続されるとき、COM1x1,0ビットの機能はWGM13~0ビット設定に依存します。表15-2はWGM13~0ビットが標準動作またはCTC動作(つまり非PWM)に設定される時のCOM1x1,0ビット機能を示します。

表15-2. 非PWM動作での比較出力選択 (注: xはAまたはB)

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

表15-3はWGM13~0ビットが高速PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表15-3. 高速PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	WGM13~0=111X : 比較一致でOC1Aピントグル(交互)出力、OC1Bは標準ポート動作(OC1B切断) WGM13~0上記以外 : 標準ポート動作 (OC1x切断)
1	0	比較一致でLow、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82頁の「高速PWM動作」をご覧ください。

表15-4はWGM13~0ビットが位相基準または位相/周波数基準PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表15-4. 位相基準または位相/周波数基準PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	WGM13~0=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83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15-5参照)。タイマ/カウンタ部により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。81頁の「動作種別」をご覧ください。

表15-5. 波形生成種別選択

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

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

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

ビット (\$81)	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### •ビット7 – ICNC1 : 捕獲起動入力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)ピンからの入力が増幅されます。この増幅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)ピンのどちらかの端(エッジ)を選びます。ICES1ビットが0を書かれると起動動作として下降(負)端が使われ、ICES1ビットが1を書かれると上昇(正)端が捕獲を起動します。

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

ICR1がTOP値として使われると(TCCR1AとTCCR1Bに配置されたWGM13~0ビットの記述をご覧ください)、ICP1が切り離され、従って捕獲入力機能は禁止されます。

### •ビット5 – Res : 予約 (Reserved)

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

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

TCCR1AのWGM11,0ビット記述をご覧ください。

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

この3つのクロック選択ビットはタイマ/カウンタ(TCNT1)によって使われるべきクロック元を選びます。図15-10と図15-11をご覧ください。

表15-6. タイマ/カウンタ1入力クロック選択

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

タイマ/カウンタ1に対して外部ピン(クロック)動作が使われる場合、例えば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。

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

ビット (\$82)	7	6	5	4	3	2	1	0	
	FOC1A	FOC1B	–	–	–	–	–	–	TCCR1C
Read/Write	W	W	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

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

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

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

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

FOC1A/FOC1Bビットは常に0として読みます。

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

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。74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。

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

ビット (\$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	

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

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。74頁の「16ビットレジスタのアクセス」をご覧ください。

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

ビット (\$87)	15	14	13	12	11	10	9	8	
	(MSB)								ICR1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$86)	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用されます。74頁の「16ビットレジスタのアクセス」をご覧ください。

### 15.11.8. タイマ/カウンタ1割り込み許可レジスタ (Timer/Counter1 Interrupt Mask Register) TMSK1

ビット (\$6F)	7	6	5	4	3	2	1	0	
	–	–	ICIE1	–	–	OCIE1B	OCIE1A	TOIE1	TMSK1
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット5 – ICIE1 : タイマ/カウンタ1捕獲割り込み許可 (Timer/Counter1 Input Capture Interrupt Enable)**

このビットが1を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1捕獲割り込みが許可されます。タイマ/カウンタ1割り込み要求フラグレジスタ(TIFR1)に配置された捕獲割り込み要求フラグ(ICF1)が設定(1)されると、対応する割り込みベクタ(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1割り込み要求フラグレジスタ(TIFR1)に配置された比較1B割り込み要求フラグ(OCF1B)が設定(1)されると、対応する割り込みベクタ(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1割り込み要求フラグレジスタ(TIFR1)に配置された比較1A割り込み要求フラグ(OCF1A)が設定(1)されると、対応する割り込みベクタ(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1割り込み要求フラグレジスタ(TIFR1)に配置されたタイマ/カウンタ1溢れ割り込み要求フラグ(TOV1)が設定(1)されると、対応する割り込みベクタ(32頁の「割り込み」参照)が実行されます。

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

ビット \$16 (\$36)	7	6	5	4	3	2	1	0	
	–	–	ICF1	–	–	OCF1B	OCF1A	TOV1	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)ストローブが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)ストローブが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(1)設定はWGM13~0ビット設定に依存します。標準またはCTC動作でのTOV1フラグはタイマ/カウンタ1溢れ時に設定(1)されます。他のWGM13~0ビット設定を使う時のTOV1フラグ動作については87頁の表15-5を参照してください。

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



## 16. 8ビット タイマ/カウンタ2 (PWM, 非同期動作付き)

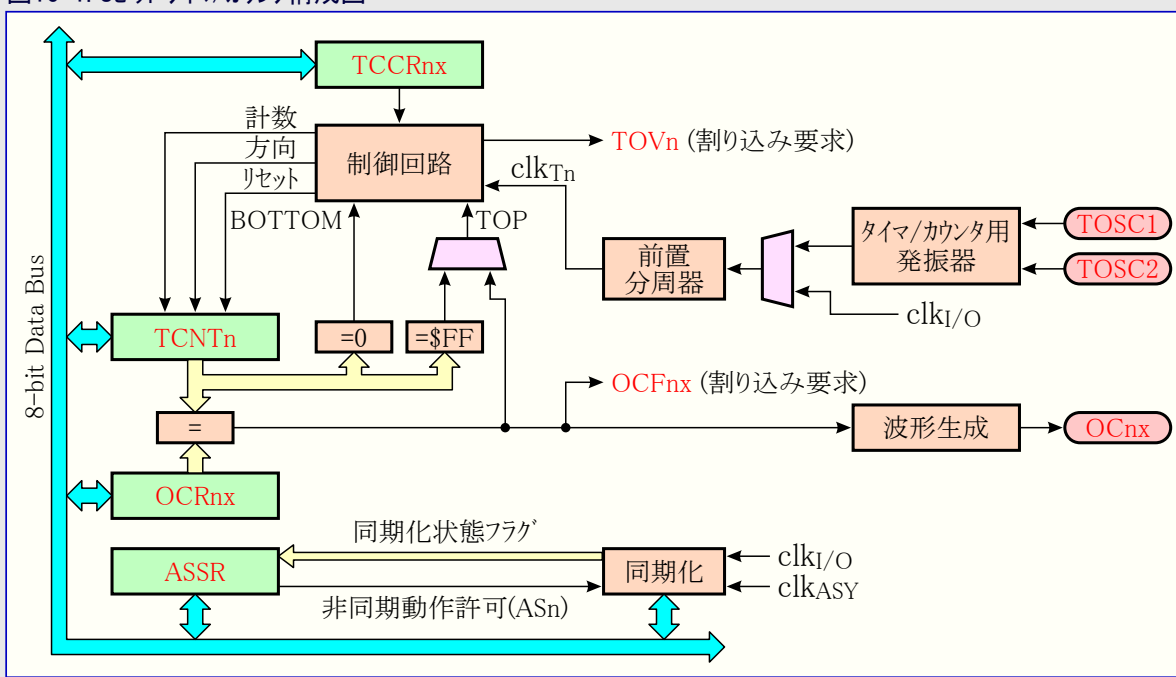
### 16.1. 特徴

- 単一比較部付き計数器
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 周波数発生器
- クロック用10ビット前置分周器
- 溢れと比較一致割り込み (TOV2とOCF2A)
- I/O(システム)クロックに依存しない時計用外部32kHzクリスタルからのクロック駆動可能

## 16.2. 概要

タイマ/カウンタ2は単一比較部付き汎用8ビットタイマ/カウンタ部です。この8ビットタイマ/カウンタの簡単化した構成図は図16-1.で示されます。I/Oピンの実際の配置については2頁の「ピン配置」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤字(訳注:原文は大文字)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101頁の「8ビットタイマ/カウンタ2用レジスタ」で一覧されます。

図16-1. 8ビット タイマ/カウンタ構成図



### 16.2.1. 関係レジスタ

タイマ/カウンタ(TCNT2)と比較レジスタ(OCR2A)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(TIFR2)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(TIMSK2)で個別に遮蔽(禁止)されます。TIFR2とTIMSK2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、または本章内後ろで詳述されるようにTOSC1/2ピンから**非同期**にクロック駆動されます。非同期動作は**非同期状態レジスタ(ASSR)**によって制御されます。クロック選択論理部はタイマ/カウンタが値を増加(または減少)するのに使うクロック元を制御します。クロック元が選ばれないと、このタイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clk<sub>T</sub>)として参照されます。

2重緩衝化した比較レジスタ(OCR2A)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(OC2A)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。詳細については93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(OCF2A)も設定(1)します。

### 16.2.2. 定義

本文書でのレジスタとビット参照の多くは一般形で書かれます。小文字の'n'はタイマ/カウンタ番号、この場合は2で置き換えます。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えばタイマ/カウンタ2のカウント値のアクセスに対するTCNT2のように)。

表16-1.の定義は本文書を通して広範囲に渡って使われます。

表16-1. 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OCR2A値に到達した時。この指定(TOP)値は動作種別に依存します。

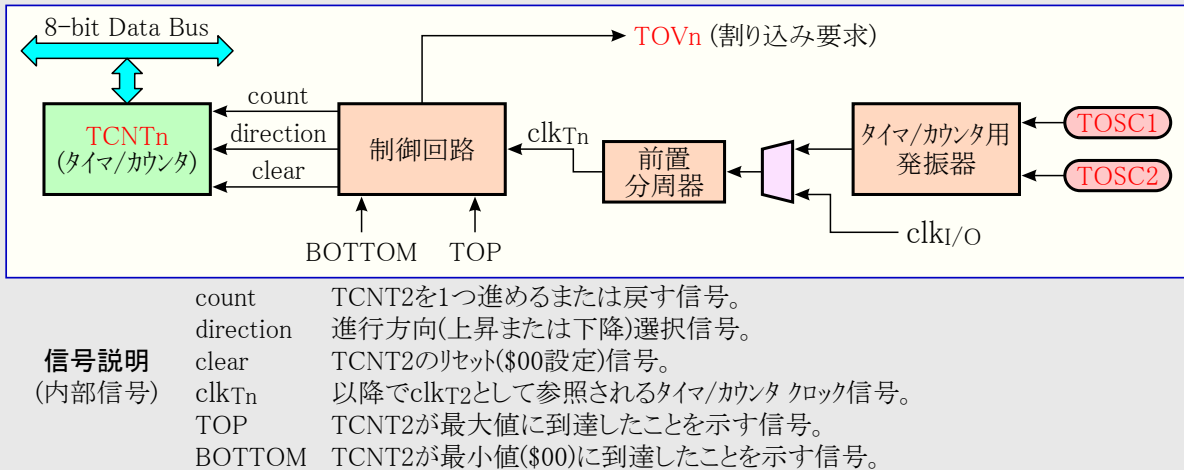
### 16.3. タイマ/カウンタのクロック

このタイマ/カウンタは内部同期または外部非同期クロック元によってクロック駆動できます。既定のクロック元( $clk_{T2}$ )はMCUクロック( $clk_{I/O}$ )と同じです。非同期状態レジスタ(ASSR)の**非同期動作許可(AS2)ビット**が論理1を書かれると、クロック元はTOSC1とTOSC2に繋がったタイマ/カウンタ用発振器から取得されます。非同期動作の詳細については103頁の「**非同期状態レジスタ(ASSR)**」をご覧ください。クロック元と前置分周器の詳細については100頁の「**タイマ/カウンタ2の前置分周器**」をご覧ください。

### 16.4. 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16-2は、このカウンタとその周辺環境の構成図を示します。

図16-2. 計数器部構成図



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( $clk_{T2}$ )で解除(\$00)、増加(+1)、または減少(-1)されます。 $clk_{T2}$ はクロック選択(CS22~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS22~0=000)時にタイマ/カウンタは停止されます。けれどもTCNT2値はタイマ/カウンタ クロック( $clk_{T2}$ )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(TCCR2A)**に配置された**波形生成種別(WGM21,0)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OC2A比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に関しては95頁の「**動作種別**」をご覧ください。

**タイマ/カウンタ溢れ(TOV2)フラグ**はWGM21,0ビットによって選ばれた動作種別に従って設定(1)されます。TOV2はCPU割り込み発生に使えます。

## 16.5. 比較出力部

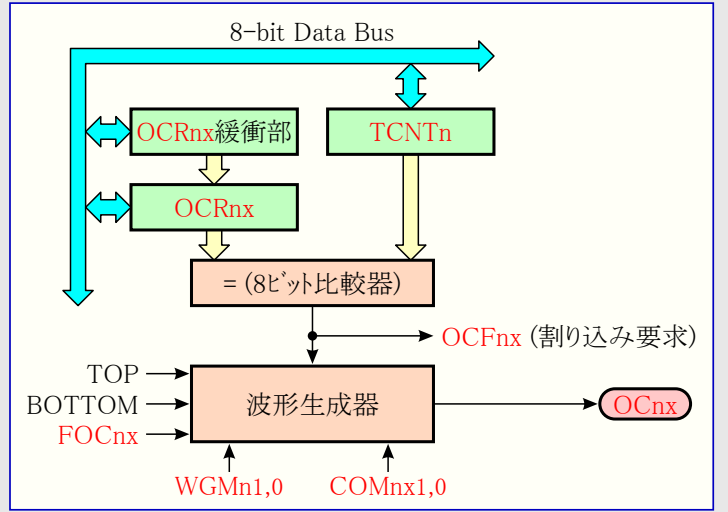
この8ビット比較器はTCNT2と比較レジスタ(OCR2A)を継続的に比較します。TCNT2とOCR2Aが等しければ、比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で**比較割り込み要求フラグ(OCF2A)**を設定(1)します。許可(I=1, OCIE2A=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF2Aは割り込みが実行されると自動的に解除(0)されます。代わりにOCF2Aはこのビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM21,0)ビット**と**比較出力選択(COM2A1,0)ビット**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(95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16-3.は比較出力部の構成図を示します。

OCR2Aは**パルス幅変調(PWM)**の何れかを使う時に2重緩衝化されます。標準動作と**比較一致タイマ/カウンタ解除(CTC)**動作については2重緩衝動作が禁止されます。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR2Aレジスタの更新を同期化します。この同期化は奇数長や非対称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。

OCR2A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OCR2A緩衝部をアクセスし、禁止されるとOCR2Aレジスタを直接アクセスします。

図16-3. 比較出力部構成図



### 16.5.1. 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(FOC2A)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(OCF2A)**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、OCR2A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(COM2A1,0ビット設定がOC2Aピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

### 16.5.2. TCNT2書き込みによる比較一致妨害

TCNT2への全てのCPU書き込みは、例えばタイマ/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、TCNT2と同じ値に初期化されることをOCR2Aに許します。

### 16.5.3. 比較一致部の使用

どの動作種別でのTCNT2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、TCNT2を変更する時に危険を伴います。TCNT2に書かれた値がOCR2A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TCNT2値を書いてはいけません。

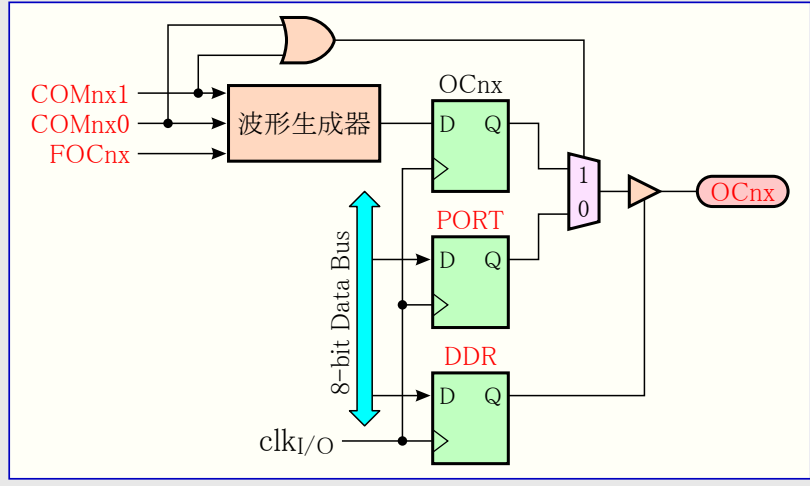
OCR2Aの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OCR2A値を設定する一番簡単な方法は標準動作で**強制変更(FOC2A)スローブビット**を使うことです。波形生成動作種別間を変更する時であっても、OCR2A(内部)レジスタはその値を保ちます。

**比較出力選択(COM2A1,0)ビット**が比較値(OCR2A)と共に2重緩衝されないことに気付いてください。COM2A1,0ビットの変更は直ちに有効となります。

## 16.6. 比較一致出力部

比較出力選択(COM2A1,0)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(OC2A)状態の定義にCOM2A1,0ビットを使います。またCOM2A1,0ビットはOC2Aピン出力元を制御します。図16-4はCOM2A1,0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。COM2A1,0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。OC2Aの状態を参照するとき、その参照はOC2Aピンでなく内部OC2Aレジスタに対してです。

図16-4. 比較一致出力回路図



COM2A1,0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(OC2A)によって無効にされます。けれどもOC2A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。OC2Aピンに対するポート方向レジスタのビット(DDR\_OC2A)はOC2A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OC2A状態の初期化を許します。いくつかのCOM2A1,0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。101頁の「8ビットタイマ/カウンタ2用レジスタ」をご覧ください。

### 16.6.1. 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COM2A1,0ビットを違うふうに使います。全ての動作種別に対してCOM2A1,0=00設定は次の比較一致で実行すべきOC2A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101頁の表16-3を参照してください。高速PWM動作については101頁の表16-4、位相基準PWMについては101頁の表16-5を参照してください。

COM2A1,0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(FOC2A)スロー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。

## 16.7. 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(WGM21,0)ビットと比較出力選択(COM2A1,0)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。COM2A1,0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COM2A1,0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(94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98頁の「タイマ/カウンタ2のタイミング」を参照してください。

### 16.7.1. 標準動作

最も単純な動作種別が標準動作(WGM21,0=00)です。本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(TOV2)フラグはTCNT2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TOV2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TOV2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。

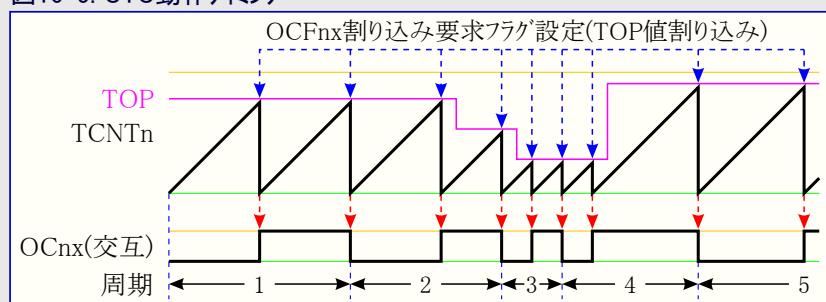
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

### 16.7.2. 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(WGM21,0=10)ではOCR2A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(TCNT2)値がOCR2Aと一致すると、カウンタは\$00に解除されます。OCR2A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16-5.で示されます。カウンタ(TCNT2)値はTCNT2とOCR2A間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(TCNT2)は解除(\$00)されます。

図16-5. CTC動作タイミング



注: COMnx1,0=01

OCF2A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。OCR2Aに書かれた新しい値がTCNT2の現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こるのに先立って、最大値(\$FF)へそして次に\$00から始める計数をしなければならないでしょう。

CTC動作で波形出力を生成するため、OC2A出力は比較出力選択(COM2A1,0)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。OC2A値はそのピンに対するデータ方向が出力(DDR\_OC2A=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OCR2Aが0(\$00)に設定される時に $f_{OC2A} = f_{clk\_I/O} / 2$ 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(1,8,32,64,128,256,1024)を表します。

標準動作と同じように、タイマ/カウンタ溢れ(TOV2)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。

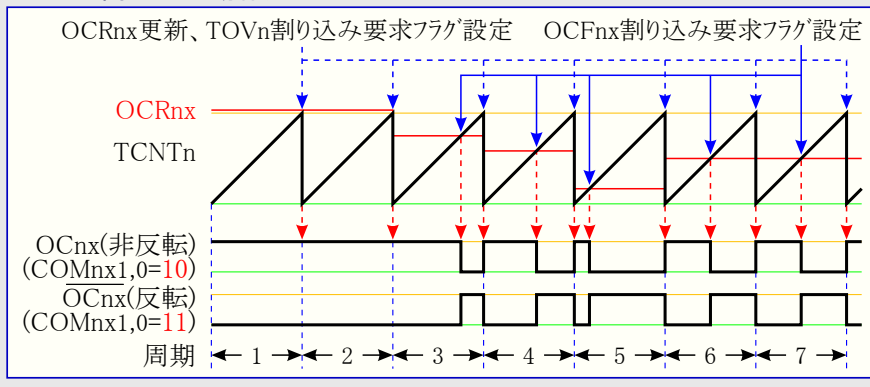


### 16.7.3. 高速PWM動作

高速パルス幅変調(PWM)動作(WGM21,0=11)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MAXまで計数し、その後にBOTTOMから再び始めます。非反転比較出力動作(COM2A1,0=10)での比較出力(OC2A)は、TCNT2とOCR2A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(COM2A1,0=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MAX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16-6.で示されます。TCNT2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OCR2A値を示し、TCNT2値との交点(接点)がTCNT2とOCR2A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(OCF2A)はOCR2A=MAX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)。

図16-6. 高速PWM動作タイミング



タイマ/カウンタ溢れ(TOV2)フラグはカウンタがMAX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OC2AピンでのPWM波形の生成を許します。COM2A1,0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COM2A1,0を'11'に設定することで生成できます(101頁の表16-4.をご覧ください)。実際のOC2A値はポートピンに対するデータ方向(DDR\_OC2A)が出力として設定される場合にだけ見えるでしょう。PWM波形はTCNT2とOCR2A間の比較一致でOC2A(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、MAXからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC2A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。

$$f_{OCnxPWM} = \frac{f_{clk\_I/O}}{N \times 256}$$

変数Nは前置分周数(1,8,32,64,128,256,1024)を表します。

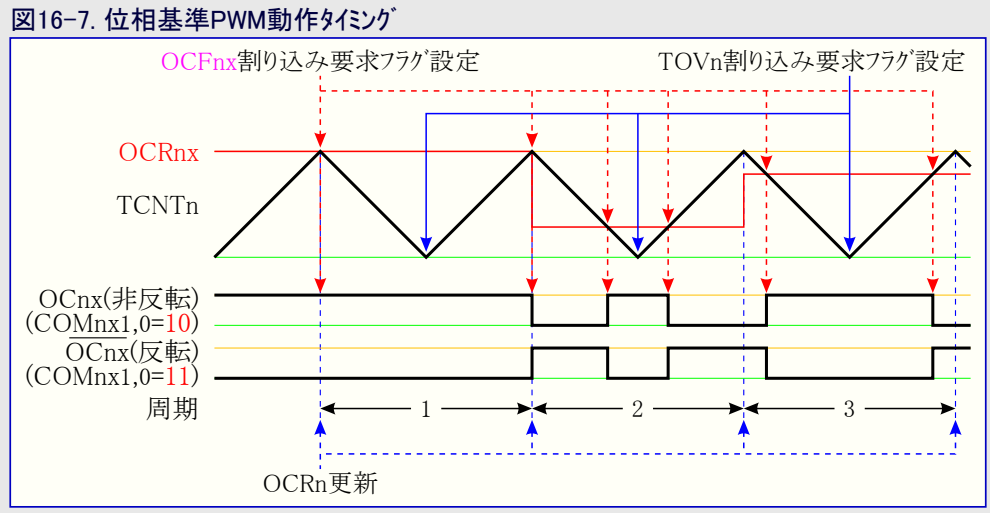
OCR2Aの両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合に当たります。OCR2AがBOTTOM(\$00)と等しく設定されると、出力はMAX+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。OCR2AがMAXに等しく設定されると、(COM2A1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

(訳注:以下は本動作種別がTOP=MAX(\$FF)で固定のため基本的に不適切です。) 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OC2A設定(COM2A1,0=01)によって達成できます。生成された波形はOCR2Aが\$0(\$00)に設定される時に $f_{OC2A} = f_{clk\_I/O} / 2$ 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OC2A交互出力(COM2A1,0=01)と同じです。

## 16.7.4. 位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM21,0=01)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)動作に基きます。カウンタはBOTTOMからMAXへそして次にMAXからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM2A1,0=10)での比較出力(OC2A)は上昇計数中のTCNT2とOCR2A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(COM2A1,0=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動作のPWM分解能は8ビットに固定されます。位相基準PWM動作でのカウンタはカウンタ値がMAXと一致するまで増加されます。カウンタはMAXに到達すると計数方向を変えます。このTCNT2値は1タイマ/カウンタクロック周期間MAXと等しくなります。位相基準PWM動作のタイミング図は図16-7で示されます。TCNT2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OCR2A値を示し、TCNT2値との交点(接点)がTCNT2とOCR2A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(TOV2)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のに使えます。

位相基準PWM動作での比較部はOC2AピンでのPWM波形の生成を許します。COM2A1,0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COM2A1,0ビットを'11'に設定することで生成できます(101頁の表16-5をご覧ください)。実際のOC2A値はそのポートピンに対するデータ方向(DDR\_OC2A)が出力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TCNT2とOCR2A間の比較一致でOC2A(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TCNT2とOCR2A間の比較一致でOC2A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。

$$f_{OCnxPCPWM} = \frac{f_{clk\_I/O}}{N \times 510}$$

変数Nは前置分周数(1,8,32,64,128,256,1024)を表します。

OCR2A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あたります。非反転PWM動作ではOCR2AがBOTTOM(\$00)に等しく設定されると出力は定常的なLow、MAX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16-7.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16-7.のようにOCR2AはMAXからその値を変更します。OCR2A値がMAX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)MAX(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OCR2A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、MAX位置で(直前がHならば)H→L遷移が生じます。)

## 16.8. タイマ/カウンタのタイミング

以下の図は同期動作でのタイマ/カウンタを示し、従ってタイマ/カウンタクロック( $clk_{Tn}$ )が計数許可信号として示されます。非同期動作では $clk_{I/O}$ がタイマ/カウンタ用発振器(TOSC)クロックによって置換されるべきです。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16-8は基本的なタイマ/カウンタ動作についてのタイミング データを示します。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16-8. 前置分周なし(1/1)のタイマ/カウンタ タイミング

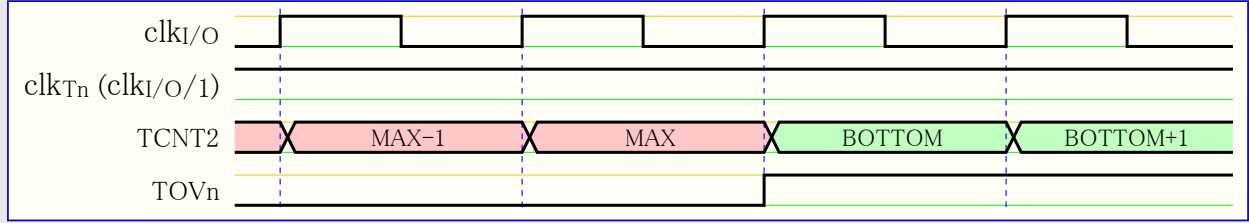


図16-9は同じタイミング データを示しますが、前置分周器が許可されています。

図16-9. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ タイミング

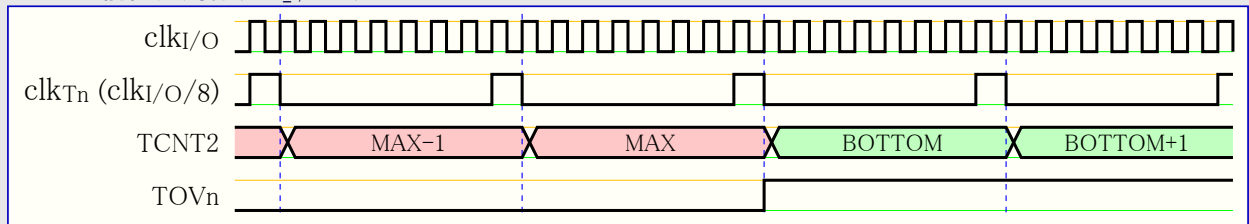


図16-10はCTC動作を除く全ての動作種別でのOCF2Aの設定を示します。

図16-10. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF2A設定 タイミング

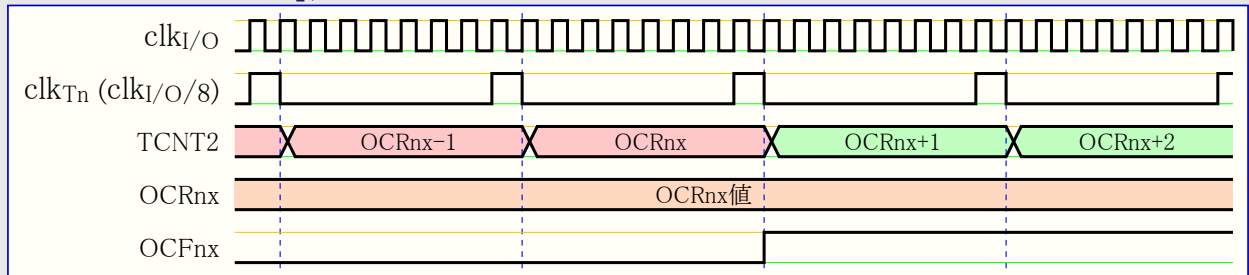
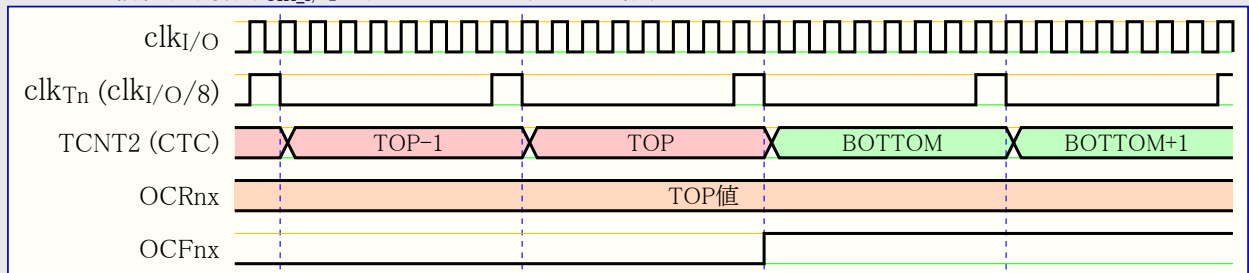


図16-11はCTC動作でのTCNT2の解除とOCF2Aの設定を示します。

図16-11. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF2A設定 タイミング



## 16.9. タイマ/カウンタ2 非同期動作

タイマ/カウンタ2が非同期で動くとき、いくつかの考慮が成されなければなりません。

- **警告:** タイマ/カウンタ2の同期/非同期クロック駆動間を切り替えると、**タイマ/カウンタ2(TCNT2)**、**タイマ/カウンタ2比較レジスタ(OCR2A)**、**タイマ/カウンタ2制御レジスタ(TCCR2A)**が不正となるかもしれません。クロック元を切り替える安全な手順を次に示します。
  1. **タイマ/カウンタ2割り込み許可レジスタ(TIMSK2)**の**OCIE2A**と**TOIE2**の解除(0)により、タイマ/カウンタ2割り込みを禁止します。
  2. **非同期状態レジスタ(ASSR)**の**非同期動作許可(AS2)**設定によってクロック元を適切に選びます。
  3. **TCNT2**,**OCR2A**,**TCCR2A**に新しい値を書きます。
  4. 非同期動作へ切り替えるには、**TCN2UB**, **OCR2UB**, **TCR2UB**について(=0まで)待機します。
  5. **タイマ/カウンタ2割り込み要求フラグ レジスタ(TIFR2)**の**OCF2A**と**TOV2**フラグを解除(0)します。
  6. 必要とされるなら、割り込みを許可します。
- CPU主クロック周波数はタイマ発振器周波数の4倍よりも高くなければなりません。
- **TCNT2**,**OCR2A**,**TCCR2A**レジスタの1つに書くとき、その値は一時レジスタへ転送され、**TOSC1**で2つの上昇端後、(実レジスタ)に設定されます。一時レジスタの内容がそれらの転送先へ転送されてしまう前に新しい値を書くべきではありません。記載された3つのレジスタの各々がそれら個別の一時レジスタを持ちます。それは、例えば**TCNT2**書き込みが**OCR2A**書き込みの実行を妨げないことを意味します。転送先レジスタへの転送が起きたことを検知するため、非同期状態レジスタ(**ASSR**)は実装されました。
- **TCNT2**,**OCR2A**,**TCCR2A**に書いた後で**パワーセーブ動作**または**A/D変換雑音低減動作**へ移行するとき、デバイスを起動するのにタイマ/カウンタ2が使われる場合、移動者は書かれたレジスタが更新されてしまうまで待たなければなりません。さもなければMCUは変更が有効になる前に**休止形態**へ移行するでしょう。タイマ/カウンタ2比較一致割り込みがデバイスを起動するのに使われる場合、**TCNT2**または**OCR2A**書き込み中の比較機能が禁止されるため、これは非常に重要です。書き込み周期が完了されず、**OCR2UB**が0に戻る前にMCUが**休止形態**へ移行すると、デバイスは決して比較一致割り込みを受け取らず、そしてMCUは起動しないでしょう。
- **パワーセーブ**または**A/D変換雑音低減動作**からデバイスを起動するのにタイマ/カウンタ2が使われる場合、移動者がこれらの動作へ(復帰後)再移行することを欲するなら、予防処置を講じなければなりません。割り込み論理回路はリセットするのに1 TOSC1周期必要です。起動と休止形態再移行間の時間が1 TOSC1周期未満の場合、割り込みが起きず、デバイスは起動に失敗するでしょう。**パワーセーブ**または**A/D変換雑音低減動作**再移行前の時間が充分であるかどうか迷うなら、1 TOSC1周期が経過されることを保証するのに次の手順が使えます。
  1. **TCNT2**,**OCR2A**,**TCCR2A**に値を書きます。
  2. 非同期状態レジスタ(**ASSR**)の対応する更新中フラグが0に戻るまで待ちます。
  3. **パワーセーブ**または**A/D変換雑音低減動作**へ移行します。
- **非同期動作が選ばれる**と、タイマ/カウンタ2用32.768kHz発振器は**パワーダウン動作**と**スタンバイ動作**を除いて常に動作します。電源投入リセット、**パワーダウン**または**スタンバイ動作**から起動後、この発振器が安定するのに1秒程度かかるかもしれないことを移動者は承知すべきです。電源投入、**パワーダウン**または**スタンバイ動作**から起動後、タイマ/カウンタ2を使う前に少なくとも1秒待機することが推奨されます。この発振器の移動またはクロック信号が**TOSC**ピンに印加されるかのどちらかに拘らず、起動時の不安定なクロック信号のため、**パワーダウン**または**スタンバイ動作**からの起動復帰後、タイマ/カウンタ2の全レジスタの内容が失われたと見做されなければなりません。
- タイマ/カウンタ2が非同期でクロック駆動される時の**パワーセーブ**または**A/D変換雑音低減動作**から起動の説明。割り込み条件が合致すると、タイマ/カウンタ2クロックの次の周期で起動処理が開始され、プロセッサがカウンタ値を読めるのに先立ってタイマ/カウンタは常に最低1進行されます。起動後MCUは4周期停止され、割り込み処理ルーチンを実行し、そして**SLEEP**命令の次の命令から実行を再開します。
- **パワーセーブ動作**から起動直後の**TCNT2**の読み込みは不正な結果を得るかもしれません。**TCNT2**が非同期**TOSC**クロックでクロック駆動されるため、**TCNT2**読み込みは内部I/Oクロック領域に同期したレジスタを通して行われなければなりません。同期化は**TOSC**クロックの全上昇端で行われます。**パワーセーブ動作**から起動し、I/Oクロック(**clkI/O**)が再び活性(有効)になると、**TCNT2**は**TOSC**クロックの次の上昇端まで以前(休止形態移行前)の値を読むでしょう。**パワーセーブ動作**から起動後の**TOSC**クロックの位相は起動時間に依存するため本質的に特定できません。従って**TCNT2**読み込みに対する推奨手順は次のとおりです。
  1. **OCR2A**または**TCCR2A**のどちらかに何か値を書きます。
  2. 非同期状態レジスタ(**ASSR**)の対応する更新中フラグが解除(0)されるまで待ちます。
  3. **TCNT2**を読みます。
- 非同期動作の間中、非同期タイマ用割り込み要求フラグの同期化は3プロセッサ周期+1タイマ周期かかります。従ってプロセッサが割り込み要求フラグ設定の原因となったタイマ値を読めるのに先立って、このタイマは最低1、進行されます。比較出力ピンはタイマクロックで変更され、プロセッサクロックに同期されません。



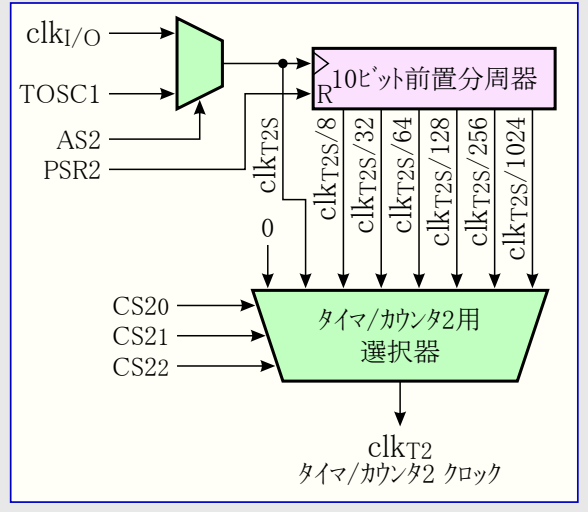
## 16.10. タイマ/カウンタ2の前置分周器

タイマ/カウンタ2用クロック元の名前は $\text{clk}_{T2S}$ です。既定での $\text{clk}_{T2S}$ は主システムI/Oクロック( $\text{clk}_{I/O}$ )に接続されます。**タイマ/カウンタ2非同期状態レジスタ(ASSR)の非同期クロック(AS2)ビット**の設定(1)により、タイマ/カウンタ2はTOSC1ピンから非同期にクロック駆動されます。これは実時間計数器(RTC)としてのタイマ/カウンタ2の使用を可能にします。AS2が設定(1)されると、TOSC1とTOSC2ピンは主クロック発振器から切り離されます。クリスタル発振子はタイマ/カウンタ2用の独立したクロック元として扱うため、TOSC1とTOSC2ピン間に接続できます。この発振器は32.768kHzクリスタル発振子で使うために最適化されています。TOSC1に外部クロック信号を印加する場合、ASSRの**外部クロック許可(EXCLK)ビット**が設定(1)されなければなりません。

タイマ/カウンタ2に対して可能な前置分周済み選択は $\text{clk}_{T2S}/8$ ,  $\text{clk}_{T2S}/32$ ,  $\text{clk}_{T2S}/64$ ,  $\text{clk}_{T2S}/128$ ,  $\text{clk}_{T2S}/256$ ,  $\text{clk}_{T2S}/1024$ です。加えて0(停止)は勿論 $\text{clk}_{T2S}$ も選択可能です。

**一般タイマ/カウンタ制御レジスタ(GTCCR)のタイマ/カウンタ2 前置分周器リセット(PSR2)ビット**の設定(1)は前置分周器をリセットします。これは予測可能な前置分周器での操作を使用者に許します。

図16-12. タイマ/カウンタ2 前置分周器部構成





## 16.11. 8ビット タイマ/カウンタ2用レジスタ

### 16.11.1. タイマ/カウンタ2制御レジスタA (Timer/Counter2 Control Register A) TCCR2A

ビット (\$B0)	7	6	5	4	3	2	1	0	
	FOC2A	WGM20	COM2A1	COM2A0	WGM21	CS22	CS21	CS20	TCCR2A
Read/Write	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 – FOC2A : OC2A強制変更 (Force Output Compare 2A)

FOC2AビットはWGM21,0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TCCR2A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。FOC2A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。OC2A出力はCOM2A1,0ビット設定に従って変更されます。FOC2Aビットがストローブ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COM2A1,0ビットに存在する値です。

FOC2Aストローブは何れの割り込みの生成もTOPとしてOCR2A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。

FOC2Aビットは常に0として読みます。

#### • ビット3,6 – WGM21,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)

これらのビットはカウンタの計数順序(方向)、最大(TOP)カウンタ値の供給元、使われるべき波形生成のどの形式かを制御します。タイマ/カウンタ部により支援される動作種別は標準動作、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。表16-2.と95頁の「動作種別」をご覧ください。

表16-2. 波形生成種別選択

番号	WGM21 (CTC2)	WGM20 (PWM2)	タイマ/カウンタ動作種別	TOP値	OCR2A 更新時	TOV2 設定時
0	0	0	標準動作	\$FF	即時	MAX
1	0	1	位相基準PWM動作	\$FF	TOP	BOTTOM
2	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR2A	即時	MAX
3	1	1	高速PWM動作	\$FF	BOTTOM	MAX

注: CTC2とPWM2ビット定義名は旧名称です。WGM21,0定義を使ってください。しかし、これらのビットの機能と位置は旧版のタイマ/カウンタと一致します。

#### • ビット5,4 – COM2A1,0 : 比較2A出力選択 (Compare Match 2A Output Mode bit 1 and 0)

これらのビットはOC2A比較出力ピンの動作を制御します。COM2A1,0ビットの1つまたは両方が1を書かれると、OC2A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、OC2A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。

OC2Aがピンに接続されるとき、COM2A1,0ビットの機能はWGM21,0ビット設定に依存します。

表16-3.はWGM21,0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COM2A1,0ビット機能を示します。

表16-4.はWGM21,0ビットが高速PWM動作に設定される時のCOM2A1,0ビットの機能を示します。

表16-5.はWGM21,0ビットが位相基準PWM動作に設定される時のCOM2A1,0ビットの機能を示します。

表16-3. 非PWM動作での比較出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A切断)
0	1	比較一致でOC2Aピン トグル(交互)出力
1	0	比較一致でOC2Aピン Lowレベル出力
1	1	比較一致でOC2Aピン Highレベル出力

表16-4. 高速PWM動作での比較出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A切断)
0	1	(予約)
1	0	比較一致でLow、BOTTOMでHighをOC2Aピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC2Aピンへ出力 (反転動作)

注: COM2A1が設定(1)され、OCR2A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)や解除(0)は行われます。より多くの詳細については96頁の「高速PWM動作」をご覧ください。

表16-5. 位相基準PWM動作での比較出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A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OC2A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OC2Aピンへ出力

注: COM2A1が設定(1)され、OCR2AがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、TOPでの設定(1)や解除(0)は行われます。より多くの詳細については97頁の「位相基準PWM動作」をご覧ください。

• ビット2~0 – CS22~0 : クロック選択2 (Clock Select2, 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(TCNT2)によって使われるクロック元を選びます。表16-6をご覧ください。

表16-6. タイマ/カウンタ2入力クロック選択

CS22	CS21	CS20	意味
0	0	0	停止 (タイマ/カウンタ2動作停止)
0	0	1	clk <sub>T2S</sub> (前置分周なし)
0	1	0	clk <sub>T2S</sub> /8 (8分周)
0	1	1	clk <sub>T2S</sub> /32 (32分周)
1	0	0	clk <sub>T2S</sub> /64 (64分周)
1	0	1	clk <sub>T2S</sub> /128 (128分周)
1	1	0	clk <sub>T2S</sub> /256 (256分周)
1	1	1	clk <sub>T2S</sub> /1024 (1024分周)

### 16.11.2. タイマ/カウンタ2 (Timer/Counter2) TCNT2

ビット (\$B2)	7 (MSB)	6	5	4	3	2	1	0 (LSB)	TCNT2
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。TCNT2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(TCNT2)を変更することは、TCNT2とOCR2A間の比較一致消失の危険を誘発します。

### 16.11.3. タイマ/カウンタ2 比較Aレジスタ (Timer/Counter2 Output Compare Register) OCR2A

ビット (\$B3)	7 (MSB)	6	5	4	3	2	1	0 (LSB)	OCR2A
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(TCNT2)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OC2Aピンでの波形出力を生成するのに使えます。

## 16.11.4. タイマ/カウンタ2非同期状態レジスタ (Timer/Counter2 Asynchronous Status Register) ASSR

ビット (\$B6)	7	6	5	4	3	2	1	0	
	–	–	–	EXCLK	AS2	TCN2UB	OCR2UB	TCR2UB	ASSR
Read/Write	R	R	R	R/W	R/W	R	R	R	
初期値	0	0	0	0	0	0	0	0	

### • ビット4 – EXCLK : 外部クロック信号許可 (Enable External Clock Input)

EXCLKが1を書かれ、非同期クロックが選ばれると、外部クロック入力緩衝部が許可され、32kHzクリスタルの代わりに外部クロックがタイマ発振器1(TOSC1)ピンへ入力できます。EXCLKへの書き込みは非同期動作が選ばれる前に行うべきです。クリスタル発振器はこのビットが0の時だけに走行(動作)することに注意してください。

### • ビット3 – AS2 : タイマ/カウンタ2非同期動作許可 (Asynchronous Timer/Counter2)

AS2が0を書かれると、タイマ/カウンタ2はI/Oクロック( $clk_{I/O}$ )からクロック駆動されます。AS2が1を書かれると、タイマ/カウンタ2はタイマ発振器(TOSC1,TOSC2)ピンに接続されたクリスタル発振器からクロック駆動されます。AS2の値が変更されると、[タイマ/カウンタ2\(TCNT2\)](#)、[比較2レジスタ\(OCR2A\)](#)、[タイマ/カウンタ2制御レジスタA\(TCCR2A\)](#)の内容は不正にされるかもしれません。

### • ビット2 – TCN2UB : タイマ/カウンタ2更新中フラグ (Timer/Counter2 Update Busy)

タイマ/カウンタ2が非同期に動き、タイマ/カウンタ2(TCNT2)が書かれると、このビットが設定(1)になります。TCNT2が一時保存レジスタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理0はTCNT2が新しい値で更新される用意ができたことを示します。

### • ビット1 – OCR2UB : 比較2レジスタ更新中フラグ (Output Compare Register2 Update Busy)

タイマ/カウンタ2が非同期に動き、比較2レジスタ(OCR2A)が書かれると、このビットが設定(1)になります。OCR2Aが一時保存レジスタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理0はOCR2Aが新しい値で更新される用意ができたことを示します。

### • ビット0 – TCR2UB : タイマ/カウンタ2制御レジスタ更新中フラグ (Timer/Counter2 Control Register Update Busy)

タイマ/カウンタ2が非同期に動き、タイマ/カウンタ2制御レジスタA(TCCR2A)が書かれると、このビットが設定(1)になります。TCCR2Aが一時保存レジスタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理0はTCCR2Aが新しい値で更新される用意ができたことを示します。

更新中フラグが設定(1)中に3つのタイマ/カウンタ2 レジスタの何れかに書き込みが実行されると、更新された値は不正にされ、予期せぬ割り込みを起こす原因になるかもしれません。

TCNT2、OCR2A、TCCR2A読み込みについての機構は異なります。TCNT2を読む時は実際のタイマ/カウンタ値が読まれ、OCR2AまたはTCCR2Aを読む時は一時保存レジスタの値が読まれます。

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

ビット \$23 (\$43)	7	6	5	4	3	2	1	0	
	TSM	–	–	–	–	–	PSR2	PSR10	GTCCR
Read/Write	R/W	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット1 – PSR2 : タイマ/カウンタ2 前置分周器リセット (Prescaler Reset Timer/Counter2)

このビットが1の時にタイマ/カウンタ2の前置分周器はリセットします。通常、このビットはハードウェアによって直ちに解除(0)されます。タイマ/カウンタ2が非同期動作の時にこのビットが(1)を書かれると、このビットは前置分周器がリセットされてしまうまで1に留まります。TSMビットが設定(1)される場合、このビットはハードウェアによって解除(0)されません。タイマ/カウンタ同期(同時)動作の記載については71頁の「[ビット7 – TSM : タイマ/カウンタ同時動作](#)」の記述を参照してください。

### 16.11.6. タイマ/カウンタ2割り込み許可レジスタ (Timer/Counter2 Interrupt Mask Register) TIMSK2

ビット (\$70)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	OCIE2A	TOIE2	TIMSK2
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット1 – OCIE2A : タイマ/カウンタ2比較A割り込み許可** (Timer/Counter2 Output Compare A Match Interrupt Enable)

OCIE2A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2比較一致割り込みが許可されます。タイマ/カウンタ2で比較一致が起こる、換言するとタイマ/カウンタ2割り込み要求フラグレジスタ(TIFR2)でタイマ/カウンタ2比較一致割り込み要求フラグ(OCF2A)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 – TOIE2 : タイマ/カウンタ2溢れ割り込み許可** (Timer/Counter2 Overflow Interrupt Enable)

TOIE2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2溢れ割り込みが許可されます。タイマ/カウンタ2溢れが起こる、換言するとタイマ/カウンタ2割り込み要求フラグレジスタ(TIFR2)でタイマ/カウンタ2溢れ割り込み要求(TOV2)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。

### 16.11.7. タイマ/カウンタ2割り込み要求フラグレジスタ (Timer/Counter2 Interrupt Flag Register) TIFR2

ビット \$17 (\$37)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	OCF2A	TOV2	TIFR2
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット1 – OCF2A : タイマ/カウンタ2比較A割り込み要求フラグ** (Timer/Counter2, Output Compare A Match Flag)

OCF2Aビットは比較一致がタイマ/カウンタ(TCNT2)と比較レジスタ(OCR2A)間で起こる時に設定(1)されます。対応する割り込み処理を実行するとき、OCF2A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OCF2A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2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ2比較一致割り込み許可(OCIE2A)ビット、OCF2Aが設定(1)されると、タイマ/カウンタ2比較一致割り込みが実行されます。

• **ビット0 – TOV2 : タイマ/カウンタ2溢れ割り込み要求フラグ** (Timer/Counter2 Overflow Flag)

TOV2ビットはタイマ/カウンタ(TCNT2)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を実行するとき、TOV2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TOV2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2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ2溢れ割り込み許可(TOIE2)ビット、TOV2が設定(1)されると、タイマ/カウンタ2溢れ割り込みが実行されます。PWM動作ではタイマ/カウンタ2が\$00で計数方向を変える時にこのビットが設定(1)されます。







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

#### アセンブリ言語プログラム例

```
SPI_M_Init: LDI    R17, (1<<DD_MOSI) | (1<<DD_SCK)      ;MOSI, SCK=出力、他は入力値を取得
            OUT    DDR_SPI, R17                        ;MOSI, SCK=出力、他は入力に設定
            LDI    R17, (1<<SPE) | (1<<MSTR) | (1<<SPR0)  ;SPI許可、主装置、16分周値を取得
            OUT    SPCR, R17                          ;SPI許可、主装置、16分周に設定
            RET                                         ;呼び出し元へ復帰

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

#### C言語プログラム例

```
void SPI_MasterInit(void)
{
    DDR_SPI = (1<<DD_MOSI) | (1<<DD_SCK);          /* MOSI, SCK=出力、他は入力に設定 */
    SPCR = (1<<SPE) | (1<<MSTR) | (1<<SPR0);        /* 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:** 6頁の「コード例について」をご覧ください。

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

#### アセンブリ言語プログラム例

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

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

#### C言語プログラム例

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

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

**注:** 6頁の「コード例について」をご覧ください。

## 17.3. $\overline{SS}$ ピンの機能

### 17.3.1. 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どのデータも取り落とします。

### 17.3.2. 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んでデータ送信を始めると解釈します。バスの衝突を避けるためにSPIシステムは次の動作を行います。

1. SPCRで主/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。

## 17.4. 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17-3と図17-4.で示されます。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17-2.で行われるように表17-3.と表17-4.を要約することによって明解にされます。

表17-2. CPOL,CPHA機能動作

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

図17-3. SPIデータ転送形式 (CPHA=0)

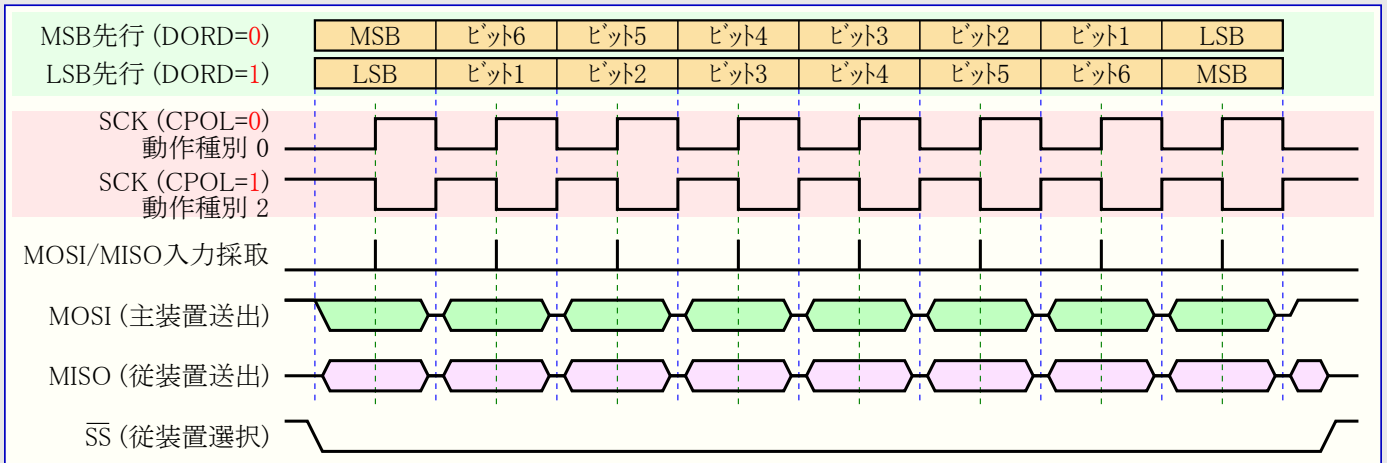
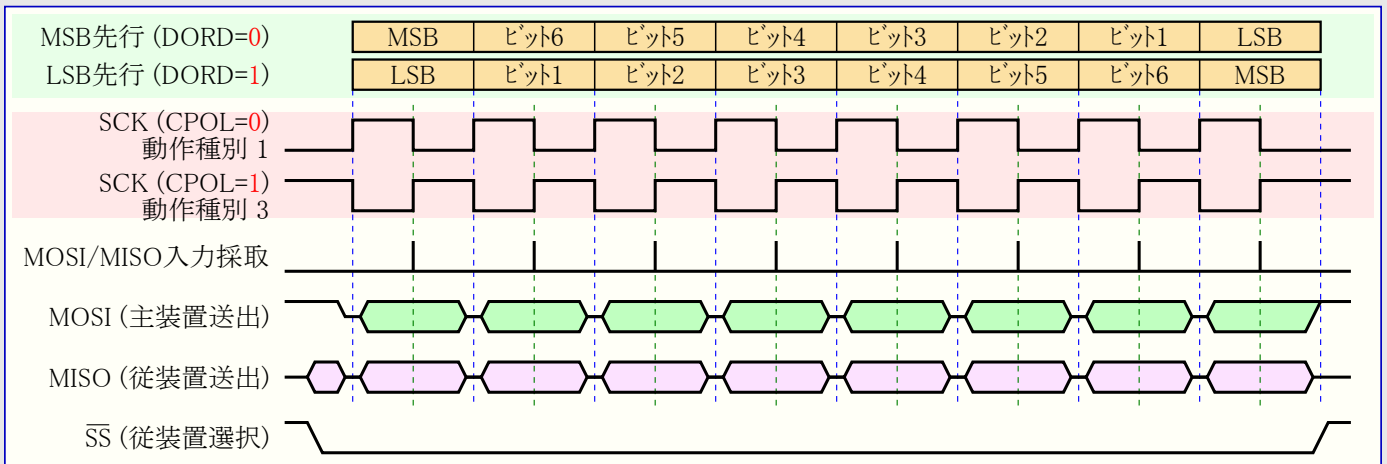


図17-4. SPIデータ転送形式 (CPHA=1)



## 17.5. SPI用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$2C (\$4C)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7 – SPIE : SPI割り込み許可 (SPI Interrupt Enable)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されて、SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)されると、このビットがSPI割り込みを実行させます。

- ビット6 – SPE : SPI許可 (SPI Enable)

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

- ビット5 – DORD : データ順選択 (Data Order)

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

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

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

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

このビットが1を書かれると、アイドル時にSCKはHighです。CPOLが0を書かれると、アイドル時にSCKはLowです。例については図17-3.と図17-4.を参照してください。CPOL機能は右で要約されます。

表17-3. CPOL機能動作

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

- ビット2 – CPHA : SCK位相選択 (Clock Phase)

このSCK位相選択(CPHA)ビットの設定はデータがSCKの先行(先)端または後行(後)端で採取/(設定)されるかを決めます。例については図17-3.と図17-4.を参照してください。CPHA機能は右で要約されます。

表17-4. CPHA機能動作

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

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

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

表17-5. SCK速度選択 ( $f_{osc}$ =CPUクロック周波数)

SPR1	0		0		1		1	
SPR0	0		1		0		1	
SPI2X	1	0	1	0	1	0	1	0
SCK周波数	fosc/2	fosc/4	fosc/8	fosc/16	fosc/32	fosc/64		fosc/128

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

ビット	7	6	5	4	3	2	1	0	
\$2D (\$4D)	SPIF	WCOL	–	–	–	–	–	SPI2X	SPSR
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7 – SPIF : SPI割り込み要求フラグ (SPI Interrupt Flag)

直列転送が完了すると、このSPIFフラグが設定(1)されます。全割り込みが許可(ステータスレジスタ(SREG)の全割り込み許可(I)ビット=1)されて、SPI制御レジスタ(SPCR)でSPI割り込み許可(SPIE)ビットが設定(1)されるなら、割り込みが生成されます。SPIが主装置動作の時に $\overline{SS}$ ピンが入力でLowに駆動されるなら、これもこのSPIFフラグを同様に設定(1)します。対応する割り込み処理ベクタを実行する時に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SPIデータレジスタ(SPD R)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)

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

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

これらのビットは予約されており、常に0として読みます。

• **ビット0 – SPI2X : SPI倍速許可** (Double SPI Speed Bit)

このビットが論理1を書かれると、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17-5参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定される時にSPIは $f_{OSC}$ (CPUクロック周波数)/4またはそれ以下の動作のみ保証されます。

ATmega325P/3250PのSPIインターフェースはフラッシュメモリやEEPROMの書き換え(読み書き)にも使われます。直列プログラミングと照合については183頁をご覧ください。

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

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。



## 18. USART (USART0)

### 18.1. 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
- データオーバーラン検出
- フレーミング異常検出
- 不正開始ビット検出とデジタル低域通過濾波器を含む雑音濾波器
- 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

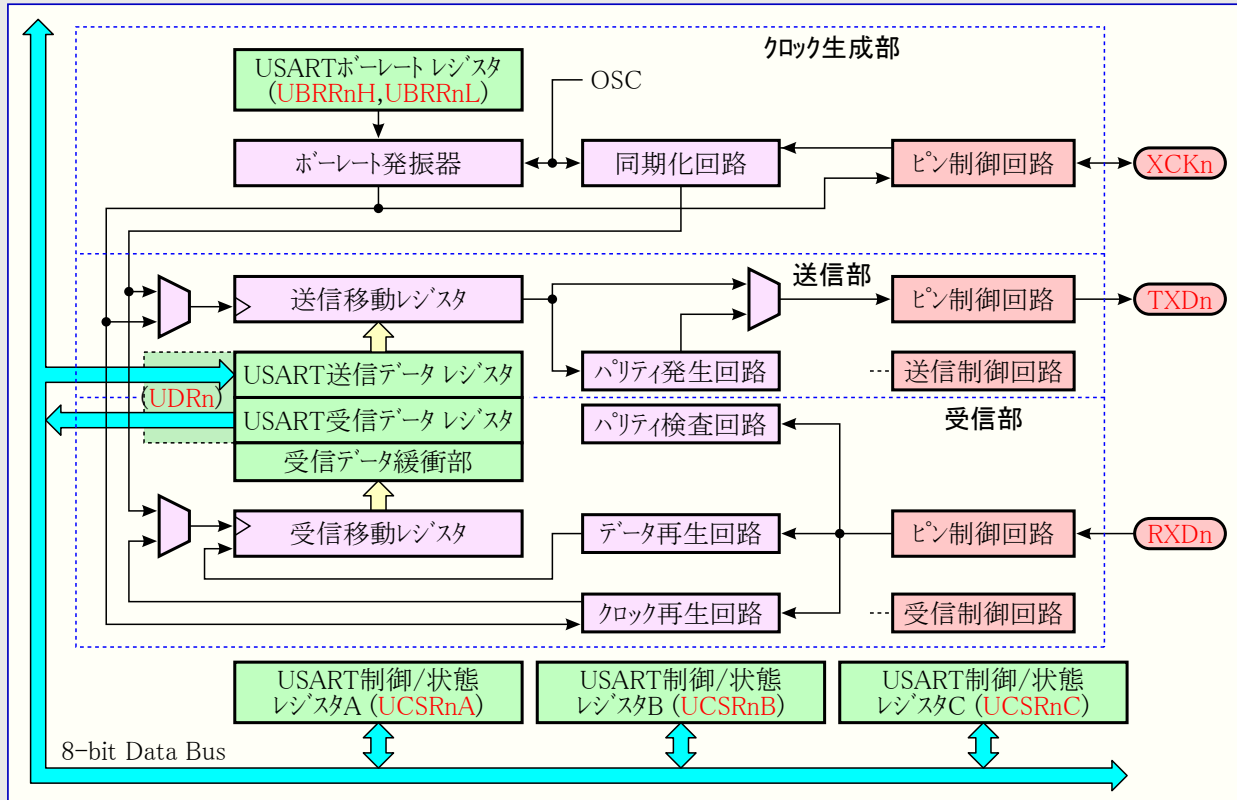
### 18.2. 概要

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の簡便化した構成図は図18-1.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:原文太字)で示されます。

26頁の「電力削減レジスタ(PRR)」でのUSART電力削減(PRUSART0)ビットはUSART部を許可するために0を書かれねばなりません。

図18-1. USART構成図



注: USARTピン配置については2頁の「ピン配置」、46頁の表13-8を参照してください。

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

### 18.2.1. UARTとの互換性

USARTはAVRのUARTと次の項目に関して完全な互換性があります。

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

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

- 第2受信緩衝部が追加されました。2つの緩衝レジスタは循環型FIFO緩衝部として動作します。従ってUSARTデータレジスタ(UDRn)は到着データ毎に一度だけ読まれなければなりません。より重要なのは異常フラグ(FEnとDORn)と第9データビット(RXB8n)が受信緩衝部内のデータと共に緩衝されることの事実です。従って状態ビットは常にUSARTデータレジスタ(UDRn)が読まれる前に読まれなければなりません。さもなければ緩衝部の状態が失われるため、異常情報も失われます。
- 受信部移動レジスタは第3緩衝段のように動けます。これは緩衝レジスタが一杯の場合、新規開始ビットが検出されるまで直列移動レジスタ(図18-1参照)内に留まるのを受信したデータに許すことによって行われます。従ってUSARTはデータオーバーラン(DORn)異常条件により耐えます。

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

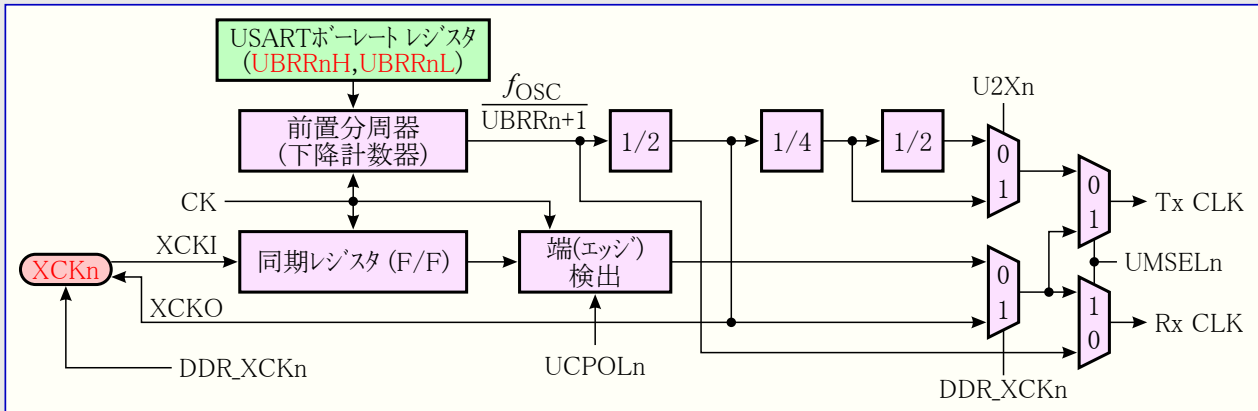
- 9ビット選択(CHR9)ビットはデータ長選択ビット2(UCSZn2)へ変更
- オーバーラン発生(OR)フラグはオーバーラン発生(DORn)フラグへ変更
- パリティ誤り(PE)フラグはパリティ誤り(UPEn)フラグへ変更 (訳注:原書で欠落のため追加)

## 18.3. 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生成します。USARTは標準非同期、倍速非同期、同期主装置、同期従装置の4つのクロック動作種別を支援します。USART制御/状態レジスタ(UCSRnC)のUSART動作種別選択(UMSELn)ビットは同期動作と非同期動作のどちらかを選びます。倍速動作(非同期動作のみ)はUSART制御/状態レジスタ(UCSRnA)にある倍速許可(U2Xn)ビットによって制御されます。同期動作(UMSELn=1)を使うとき、XCKnピンに対する方向制御ビット(DDR\_XCKn)はクロック元が内部(主装置動作)または外部(従装置動作)どちらかを制御します。このXCKnピンは同期動作を使う時だけ活性(有効)です。

図18-2. はクロック生成論理回路の構成図を示します。

図18-2. クロック生成部構成図



信号説明

- Tx CLK : 送信クロック (内部信号)
- Rx CLK : 受信クロック (内部信号)
- XCKI : XCKnピンからの内部入力信号 (同期従装置時)
- XCKO : XCKnピンへの内部出力信号 (同期主装置時)
- DDR\_XCKn : 対応するポート方向レジスタの方向ビット
- f<sub>OSC</sub> : システムクロック周波数

### 18.3.1. ボーレート発振器での内部クロック発生

内部クロック生成は非同期と同期主装置動作種別に対して使われます。本項の記述は図18-2.を参照してください。

USARTボーレートレジスタ(UBRRn(UBRRnH:UBRRnL))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振器として機能するように接続されます。システムクロック(f<sub>OSC</sub>)で走行する下降カウンタは0への下降計数時毎またはUBRRnLレジスタが書かれる時にUBRRn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。このクロックがボーレート発振器出力(=f<sub>OSC</sub>/(UBRRn+1))です。送信部は動作種別に依存してボーレート発振器出力を2,8,16分周します。ボーレート発振器出力は受信部クロックとデータ再生部によって直接使われます。しかし、再生部はUSART動作種別選択(UMSELn)、倍速許可(U2Xn)、DDR\_XCKnビットの状態によって設定される動作種別に依存して2,8,16段を使う順次処理回路を使います。

表18-1. は内部的に生成したクロック元を使う各動作種別に於けるボーレート(bps)とUBRRn値の計算式を含みます。

表18-1. ボーレートレジスタ(UBRRn)値計算式

動作種別	ボーレート計算式	UBRRn値計算式
標準速非同期動作 (U2Xn=0)	$BAUD = \frac{f_{OSC}}{16 \times (UBRRn + 1)}$	$UBRRn = \frac{f_{OSC}}{16 \times BAUD} - 1$
倍速非同期動作 (U2Xn=1)	$BAUD = \frac{f_{OSC}}{8 \times (UBRRn + 1)}$	$UBRRn = \frac{f_{OSC}}{8 \times BAUD} - 1$
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRRn + 1)}$	$UBRRn = \frac{f_{OSC}}{2 \times BAUD} - 1$

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)

UBRRn : UBRRnHとUBRRnLレジスタ値 (0~4095)

f<sub>OSC</sub> : システム発振器クロック周波数

いくつかのシステムクロック周波数に対するいくつかのUBRRn値の例は126~127頁の表18-9.で得られます。

### 18.3.2. 倍速動作 (U2Xn)

転送速度はUSART制御/状態レジスタA(UCSRnA)で倍速許可(U2Xn)ビットを設定(1)することによって倍にできます。このビットの設定は非同期動作に対してだけ有効です。同期動作を使うとき、このビットは0に設定してください。

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

### 18.3.3. 外部クロック

外部クロックは同期従装置動作種別によって使われます。本項の記述での詳細については図18-2を参照してください。

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

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

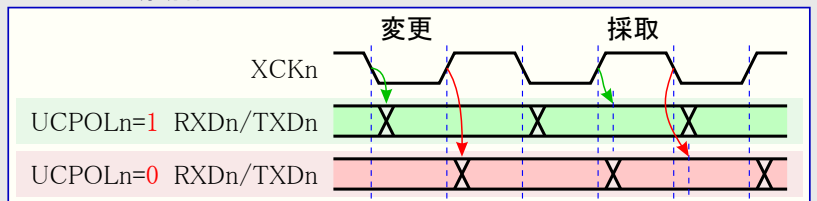
$f_{OSC}$ がシステムクロック元の安定度に依存することに注意してください。従って周波数変動によるデータ消失の可能性を避けるため、いくらかの余裕分を追加することが推奨されます。

### 18.3.4. 同期クロック動作

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

USART制御/状態レジスタC(UCSRnC)のXCKn極性(UCPOLn)ビットはデータ採取とデータ変更に対してどちらのクロック端が使われるのかを選びます。図18-3.で示されるようにUCPOLnが0のとき、データはXCKnの上昇端で変更され、下降端で採取されます。UCPOLnが設定(1)の場合、データはXCKnの下降端で変更され、上昇端で採取されます。

図18-3. 同期動作XCKnタイミング



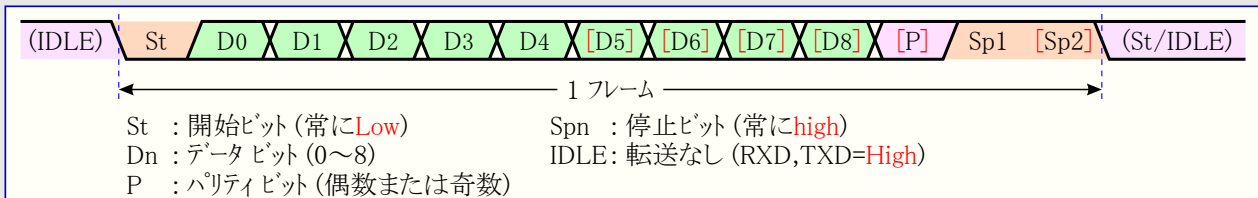
## 18.4. フレーム形式

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。

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

フレームは最下位データビット(LSB)が次に続く開始ビットで始まります。その後に次データビットが最後の最上位データビット(MSB)まで(最大)合計9ビット続きます。許可したなら、パリティビットがデータビットの後、停止ビットの前に挿入されます。完全なフレームが送信されると、新規フレームによって直ちに後続されるか、または通信線をアイドル状態(high)に設定できます。図18-4.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18-4. 1フレームの構成



USARTによって使われるフレーム形式はUSART制御/状態レジスタBとC(UCSRnB, UCSRnC)でデータ長選択(UCSZn2~0)ビット、パリティ選択(UPMn1,0)ビット、停止ビット選択(USBSn)ビットによって設定されます。受信部と送信部は同じ設定を使います。これらのどのビットの設定変更も、送受信部両方に対して進行中の通信を不正とすることに注意してください。

データ長選択(UCSZn2~0)ビットはフレーム内のデータビット数を選びます。パリティ選択(UPMn1,0)ビットはパリティビットの許可と種別(奇/偶)を設定します。1または2停止ビットのどちらかの選択は停止ビット選択(USBSn)ビットによって行います。受信部は第2停止ビットを無視します。従ってフレーミング異常(FEN)は最初の停止ビットが0(Low)の場合にだけ検出されます。

## 18.4.1. パリティビットの計算

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

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

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

## 18.5. 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操作に関して初期化を行うとき、[ステータスレジスタの全割り込み許可\(0\)ビット](#)は解除(0)される(そして全割り込みが禁止される)べきです。

ボーレートまたはフレーム形式の変更を伴う再初期化を行う前には、レジスタが変更される期間中に進行中の送信がないことを確実にしてください。[USART制御/状態レジスタA\(UCSRnA\)](#)の[送信完了\(TXCn\)フラグ](#)は送信部の全転送完了検査に使え、[受信完了\(RXCn\)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\(UDRn\)](#)が書かれる)前にTXCnフラグが解除(0)されなければならないことに注意してください。

次の簡単なUSART初期化コード例が示すアセンブリ言語とC言語の関数は機能的に同じです。この例は固定フレーム形式でボーリングを使う(割り込み不許可)非同期動作と仮定します。ボーレート(UBRRn)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    UBRRnH, R17                ;ボーレート設定(上位バイト)
              OUT    UBRRnL, R16                ;ボーレート設定(下位バイト)
              LDI     R16, (1<<USBSn) | (3<<UCSZn0) ;フレーム形式値を取得
              OUT    UCSRnC, R16                ;フレーム形式設定(8ビット,2停止ビット)
              LDI     R16, (1<<RXENn) | (1<<TXENn) ;送受信許可値を取得
              OUT    UCSRnB, R16                ;送受信許可
              RET                                ;呼び出し元へ復帰
```

### C言語プログラム例

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

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

void USART_Init(unsigned int baud)
{
    UBRRnH = (unsigned char) (baud>>8); /* ボーレート設定(上位バイト) */
    UBRRnL = (unsigned char) baud;       /* ボーレート設定(下位バイト) */
    UCSRnC = (1<<USBSn) | (3<<UCSZn0);  /* フレーム形式設定(8ビット,2停止ビット) */
    UCSRnB = (1<<RXENn) | (1<<TXENn);    /* 送受信許可 */
}
```

**注:** 6頁の「コード例について」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。



## 18.6. USARTのデータ送信

USART送信部はUSART制御/状態レジスタB(UCSRnB)で送信許可(TXENn)ビットを設定(1)することによって許可されます。送信部が許可されると、TXDn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、XCKn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。

### 18.6.1. 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(UDRn)I/O位置へ書くことによって送信緩衝部に設定できます。送信緩衝部内のデータは移動レジスタが新規フレームを送る準備が整った時に移動レジスタへ移されます。移動レジスタはアイドル状態(送信進行中以外)、または直前のフレームの最後の停止ビット送信後、直ちに新規データが設定されます。移動レジスタが新規データを設定されると、ポーレートレジスタ(UBRRnH:UBRRnL)と倍速許可(U2Xn)ビット、また動作種別によってはXCKnピンによって与えられる速度で1つの完全なフレームを転送します。8ビット未満のフレームを使うとき、UDRnに書かれた上位ビットは無視されます。

次のコード例はUSART制御/状態レジスタA(UCSRnA)の送信データレジスタ空き(UDREN)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    UCSRnA, UDREN      ;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     UDRn, 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 ( !(UCSRnA & (1<<UDREN)) ); /* 送信緩衝部空き待機 */
    UDRn = data;                       /* データ送信(送信開始) */
}
```

**注:** 6頁の「コード例について」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、UDRENの検査によって送信緩衝部が空になるのを単純に待ちます。送信緩衝部空き割り込みが使われる場合、その割り込み処理ルーチンがデータを緩衝部内に書きます。

### 18.6.2. 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(UCSZn2～0=111)、データの下位8ビットがUSARTデータレジスタ(UDRn)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(UCSRnB)の送信データビット8(TXB8n)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    UCSRnA, UDREN      ;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     UCSRnB, TXB8n      ;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     UCSRnB, TXB8n      ;第9ビットを1に設定
           OUT     UDRn, 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 ( !(UCSRnA & (1<<UDREN)) ); /* 送信緩衝部空き待機 */
    UCSRnB &= ~(1<<TXB8n);           /* TXB8nを0に仮設定 */
    if (data & 0x0100) UCSRnB |= (1<<TXB8n); /* 第9ビットをR17からTXB8nへ複写 */
    UDRn = 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。UCSRBnの内容が静的(換言すると、UCSRnBのTXB8n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。

6頁の「コード例について」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。

## 18.6.3. 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(UDREN)と送信完了(TXCn)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(UDREN)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るかどうかを示します。このビットは送信緩衝部が空の時に設定(1)され、送信緩衝部が移動レジスタに未だ移動されてしまっていない送信されるべきデータを含む時に解除(0)されます。将来のデバイスとの共通性のため、USART制御/状態レジスタA(UCSRnA)に書くとき、常にこのビットに0を書いてください。

USART制御/状態レジスタB(UCSRnB)でデータレジスタ空き割り込み許可(UDRIEN)ビットが1を書かれると、(全割り込みが許可されていれば)UDREN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。UDRENはUSARTデータレジスタ(UDRn)書き込みによって解除(0)されます。割り込み駆動データ送信が使われるとき、データレジスタ空き割り込みルーチンはUDRENを解除(0)するために新規データをUDRn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(TXCn)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。TXCnフラグは送信完了割り込みが実行されるとき、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。TXCn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。

UCSRnB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)TXCnフラグが設定(1)になる時にUSART送信完了割り込みが実行されます。送信完了割り込みが使われるとき、割り込み処理ルーチンはTXCnフラグを解除(0)しなくてもよく、これは割り込みが実行されるとき、自動的に行われます。

## 18.6.4. 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(UPMn1=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。

## 18.6.5. 送信の禁止

送信部の禁止(UCSRnBのUSART送信許可(TXEN)ビット=0)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない時)まで有効になりません。禁止されると、送信部はもはやTXDnピン(の標準ピン機能)を無効にしません。

## 18.7. USARTのデータ受信

USART受信部はUSART制御/状態レジスタB(UCSRnB)で受信許可(RXEN)ビットに1を書くことによって許可されます。受信部が許可されると、RXDnピンの標準ピン動作はUSARTによって無視され、受信部の直列入力としての機能を与えられます。何か直列受信が行われ得る前に一度はボーレート、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、XCKnピンのクロックは転送クロックとして使われます。

### 18.7.1. 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XCKnクロックで採取され、フレームの最初の停止ビットが受信されるまで受信移動レジスタ内へ移動されます。第2停止ビットは受信部によって無視されます。最初の停止ビットが受信されると(換言すると、受信移動レジスタに完全なフレームが存在すると)、この移動レジスタの内容は受信緩衝部内へ移されます。受信緩衝部はUSARTデータレジスタ(UDRn)I/O位置を読むことによって読めます。8ビット未満のフレームを使うとき、UDRn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(UCSRnA)の受信完了(RXCn)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    UCSRnA, RXCn      ;受信完了でスキップ
            RJMP    USART_Rx        ;受信完了待機
;
            IN      R16, UDRn        ;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 ( !(UCSRnA & (1<<RXCn)) ); /* 受信完了待機 */
    return UDRn;                      /* 受信データ取得 */
}
```

**注:** 6頁の「コード例について」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、RXCn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。

### 18.7.2. 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(UCSZn2~0=111)、USARTデータレジスタ(UDRn)から下位8ビッ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(UCSRnB)の受信データビット8(RXB8n)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(FEn)、オーバーラン発生(DORn)、パリティ誤り(UPEn)状態フラグにも適用されます。USART制御/状態レジスタA(UCSRnA)から状態情報を読み、その後にUDRnからデータを読んでください。UDRn I/O位置を読むことが受信FIFO緩衝部の状態を切り替え、その結果、FIFO内に保管されるRXB8n,FEn,DORn,UPEnビット全てが切り替わります。

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

#### アセンブリ言語プログラム例

```
USART_Rx:  SBIS      UCSRnA, RxCn          ;受信完了でスキップ
           RJMP     USART_Rx             ;受信完了待機
;
           IN       R18, UCSRnA          ;状態フラグ取得
           IN       R17, UCSRnB          ;受信第9ビット取得
           IN       R16, UDRn            ;受信データ取得
           ANDI     R18, (1<<FEn) | (1<<DORn) | (1<<UPEn) ;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                  ;RXB8nビットをビット0位置へ移動
           ANDI     R17, $01              ;RXB8n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;
    while ( !(UCSRnA & (1<<RxCn)) );      /* 一時変数定義 */
    status = UCSRnA;                       /* 受信完了待機 */
    resh = UCSRnB;                         /* 状態フラグ取得 */
    resl = UDRn;                           /* 受信第9ビット取得 */
    if ( status & ((1<<FEn) | (1<<DORn) | (1<<UPEn)) ) return -1; /* 受信データ取得 */
    resh = (resh>>1) & 0x01;               /* 受信異常で-1値設定/復帰 */
    return ((resh<<8) | resl);              /* RXB8nビットのみ有効最下位へ */
}
```

**注:** 6頁の「コード例について」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。

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

USART制御/状態レジスタB(UCSRnB)でUSART受信完了割り込み許可(RXCIEn)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)RxCnフラグが設定(1)されている限り、USART受信完了割り込みが実行されます。割り込み駆動データ受信が使われるとき、受信完了割り込みルーチンはRxCnフラグを解除(0)するためにUSARTデータレジスタ(UDRn)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。

## 18.7.4. 受信異常フラグ

USART受信にはフレーミング異常(FEn)、データ オーバーラン発生(DORn)、パリティ誤り(UPEn)の3つの異常フラグがあります。これら全てはUSART制御/状態レジスタA(UCSRnA)を読むことでアクセスできます。異常フラグに対する共通点は異常状態を示しているフレームと共に受信緩衝部に配置されることです。異常フラグが緩衝されるため、USARTデータ レジスタ(UDRn)I/O位置を読むことが緩衝部読み出し位置を切り替えるので、UCSRnAは受信緩衝部(UDRn)の前に読まれなければなりません。異常フラグに対するその他の共通点はソフトウェアがこのフラグ位置へ書き込みを行うことによって変更できないことです。しかし、将来のUSART(機能)実装の上位互換性のため、UCSRnAが書かれるとき、全てのフラグは0に設定されなければなりません。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(FEn)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。FEn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FEnフラグは1です。このビットは同期外れ状態の検出、中断状態検出、規約での操作に使えます。受信部が最初(第1)以外の全停止ビットを無視するため、FEnフラグはUSART制御/状態レジスタC(UCSRnC)の停止ビット選択(USBSn)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、UCSRnAに書くとき、常にこのビットを0に設定してください。

**データ オーバーラン発生(DORn)フラグ**は受信部緩衝部が一杯状態のためのデータ消失を示します。データ オーバーランは受信緩衝部(2フレーム)が一杯で、(次の)新規フレーム データが受信移動レジスタで待っており、(更に次の)新規開始ビットが検出される時に起きます。DORnフラグが設定(1)なら、最後にUDRnから読んだフレームと次にUDRnから読むフレーム間で1つ以上の直列フレームが失われています。将来のデバイスとの共通性のため、UCSRnAに書くとき、常にこのビットに0を書いてください。DORn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(UPEn)フラグ**は受信緩衝部内の次のフレームで受信時にパリティ異常があったことを示します。パリティ検査が許可されていない場合、UPEnフラグは常に0が読めます。将来のデバイスとの共通性のため、UCSRnAに書くとき、常にこのビットを0に設定してください。より多くの詳細については115頁の「パリティビットの計算」と次の「パリティ検査器」をご覧ください。

## 18.7.5. 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(UPMn1)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UPMn0ビットによって選ばれます。許可されると、パリティ検査器は到着フレーム内のデータビットのパリティを計算し、その結果と(受信)直列フレーム内のパリティビットを比較します。検査の結果は受信データ、停止ビットと共に受信緩衝部に格納されます。その後パリティ誤り(UPEn)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。

UPEnフラグは受信緩衝部から読まれ得る次フレームで受信時にパリティ異常があり、その時点でパリティ検査が許可されていた(UPMn1=1)場合に設定(1)されます。このビットはUSARTデータ レジスタ(UDRn)が読まれるまで有効です。

## 18.7.6. 受信の禁止

送信部と対照的に受信部の禁止は即時です。従って受信進行中のデータは失われます。禁止されると(換言すると、USART制御/状態レジスタB(UCSRnB)のUSART受信許可(RXENn)ビットが0に設定)、受信部はもはやRXDnポート ピンの標準機能を無効にしません。受信FIFO緩衝部は受信部が禁止されると破棄されます。緩衝部内の残データは失われます。

## 18.7.7. 受信緩衝部の破棄

受信FIFO緩衝部は受信が禁止されると破棄(換言すると、緩衝部は内容を空に)されます。未読データは失われます。例えば異常状態のため、通常動作中に緩衝部が破棄されなければならない場合、USART制御/状態レジスタA(UCSRnA)の受信完了(RXCn)フラグが解除(0)されるまでUSARTデータ レジスタ(UDRn)I/O位置を読んでください。次のコード例は受信緩衝部の破棄方法を示します。

### アセンブリ言語プログラム例

```
USART_Flush: SBIS    UCSRnA, RXCn    ;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, UDRn      ;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 ( UCSRnA & (1<<RXCn) ) dummy=UDRn; /* 未読データ読み捨て */
}
```

**注:** 6頁の「コード例について」をご覧ください。



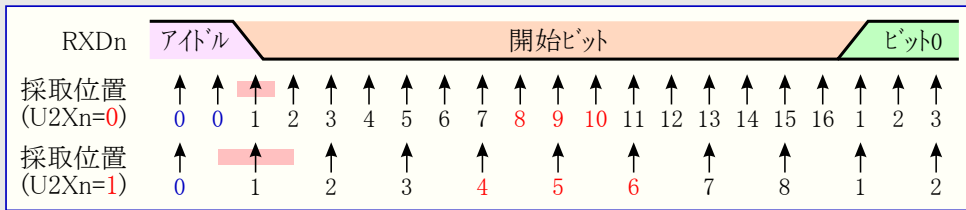
## 18.8. 非同期受信

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RXDnピンに到着する非同期直列フレームに同期化するために使われます。データ再生論理回路は到着ビット毎に低域通過の濾波と採取をし、それによって受信部の雑音耐性を改善します。非同期受信動作範囲は内部ボーレートクロック精度、到着フレーム速度、フレーム長のビット数に依存します。

### 18.8.1. 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18-5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(U2Xn=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RXDn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18-5. 開始ビットの採取

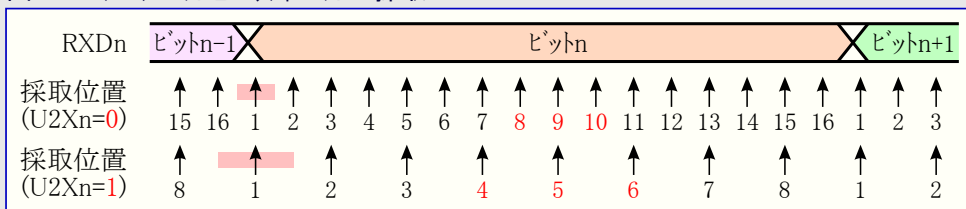


クロック再生論理回路がRXDn信号線でHigh(アイドル)からLow(開始)の遷移を検出すると、開始ビット検出手順が開始されます。図で示されるように採取1は最初のLow採取を意味します。その後にクロック再生回路は有効な開始ビットが受信されるかを決定するために、標準速動作に対して採取8,9,10、倍速動作に対して採取4,5,6(図の赤字(訳注:原文は箱枠内)で示された採取番号)を使います。これら3回で2回以上の採取が論理Highレベルの場合(多数決)、この開始ビットは尖頭雑音として捨てられ、受信部は次のHighからLowの遷移を探し始めます。けれども有効な開始ビットが検出されると、クロック再生論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

### 18.8.2. 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18-6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。

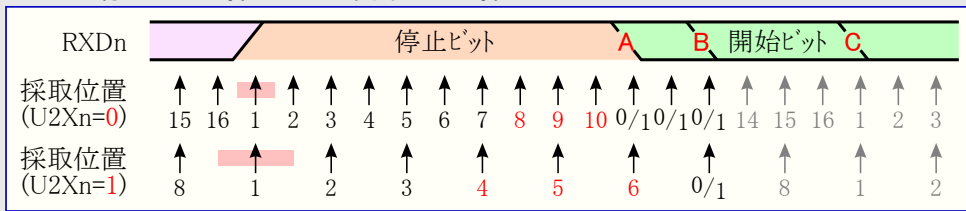
図18-6. データビットとパリティビットの採取



受信したビットの論理値の決定は受信したビット中央の3採取で論理値の多数決を取るによって行われます。この中央の3採取は図上の赤字(訳注:原文は箱枠内)の採取番号によって強調されます。多数決の手順は次のように行われます。2または全3採取がHighレベルならば受信したビットは論理1が記録されます。2または全3採取がLowレベルならば受信したビットは論理0が記録されます。この多数決手順はRXDn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18-7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18-7. 停止ビットの採取と次の開始ビットの採取



フレーム内の他のビットに対して行われるのと同じ多数決が停止ビットにも行われます。停止ビットが論理0値と記録されると、フレーミング異常(FEn)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。標準速動作での最初のLowレベル採取は図18-7のA点で有り得ます。倍速動作での最初のLowレベルはB点に遅れます。C点は完全な長さの停止ビット(の終点)を示します。この早い開始ビット検出は受信部の動作範囲に影響します。



## 18.8.3. 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ボーレート間の不一致に依存します。送信部が速すぎるまたは遅すぎるビット速度でフレームを送出したり、内部的に発生した受信部のボーレートが類似した(表18-2.参照)基準周波数を持たない場合、受信部は開始ビットでフレームを同期できません。

次式は到着データ速度と内部受信部ボーレート間の比率計算に使えます。

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

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> : は受信側ボーレートに対して許容できる最低受信ビット速度の比率です。  
 R<sub>fast</sub> : は受信側ボーレートに対して許容できる最高受信ビット速度の比率です。

表18-2.は許容できる最大受信部ボーレート誤差一覧です。標準速動作には、より高いボーレート変動許容力があることに注目してください。

表18-2. 標準速と倍速での受信部ボーレート推奨最大許容誤差

D	標準速動作 (U2Xn=0)				倍速動作 (U2Xn=1)			
	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)
5	93.20	106.67	-6.80~+6.67	±3.0	94.12	105.66	-5.88~+5.66	±2.5
6	94.12	105.79	-5.88~+5.79	±2.5	94.92	104.92	-5.08~+4.92	±2.0
7	94.81	105.11	-5.19~+5.11	±2.0	95.52	104.35	-4.48~+4.35	±1.5
8	95.36	104.58	-4.54~+4.58	±2.0	96.00	103.90	-4.00~+3.90	±1.5
9	95.81	104.14	-4.19~+4.14	±1.5	96.39	103.53	-3.61~+3.53	±1.5
10	96.17	103.78	-3.83~+3.78	±1.5	96.70	103.23	-3.30~+3.23	±1.0

注: Dはデータビット数とパリティビットの合計ビット数です。

(訳注) 原書は表18-1.に標準速、表18-2.に倍速を記載していますが、比較が容易なように表18-2.として纏めました。

受信部ボーレートの推奨最大許容誤差は最大総合許容誤差を送信部と受信部で等分割するという仮定の下で作られました。

受信部ボーレート誤差に対して2つの起こり得る原因があります。受信部のシステムクロック(XTAL)は供給電圧範囲と温度範囲に関して常に若干の不安定性があります。システムクロックを生成するのに水晶発振子を使う時は殆ど問題ありませんが、(セラミック)振動子でのシステムクロックは振動子偏差に依存して2%を越えて異なるかもしれません。2つ目の誤り原因はより制御可能です。ボーレート発振器は欲したボーレートを得るためにシステム周波数の正確な分周を常に行うことはできません。この場合、可能ならば受け入れ可能な低い誤差を与えるUBRRn値が使えます。

## 18.9. 複数プロセッサ通信動作

USART制御/状態レジスタA(UCSRnA)での複数プロセッサ通信動作(MPCMn)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MPCMn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(UCSRnB)の受信第9(RXB8n)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。

### 18.9.1. 複数プロセッサ通信の使用法

主MCUとして動作するMCUは9ビットデータフレーム形式(UCSZn=7)を使えます。UCSRnBの送信第9(TXB8n)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(UCSRnAの複数プロセッサ通信動作(MPCMn)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UCSRnAで受信完了(RXCn)フラグが設定(1)されます。
3. 各従MCUはUSARTデータレジスタ(UDRn)を読み、選ばれたかを判定します。選ばれた場合はUCSRnAのMPCMnビットを解除(0)し、そうでなければ(非選択の場合は)MPCMn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。MPCMn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MPCMn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(USBSn=1)に設定されなければなりません。

MPCMn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-モディファイライト)命令(SBIとCBI)を使ってはいけません。MPCMnビットは送信完了(TXCn)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) ATmega325P/3250PではUCSRnAのI/OアドレスがSBI,CBI命令適用範囲外のため、上記記述は不適切です。但し、命令の組み合わせによって同様処理を行う場合に対して、上記注意の本意は適切(有効)です。

## 18.10. USART用レジスタ

### 18.10.1. USARTデータレジスタ (USART I/O Data Register) UDRn

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	UDRn
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

同じI/Oアドレスを共用するUSART受信データ緩衝レジスタとUSART送信データ緩衝レジスタはUSARTデータレジスタまたはUDRnとして引用しました。送信データ緩衝レジスタ(TXB)はUDRnレジスタ位置に書かれるデータの転送先です。UDRnレジスタ位置読み込みは受信データ緩衝レジスタ(RXB)の内容を返します。

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

送信緩衝部はUCSRnAで**送信データレジスタ空き(UDREN)フラグ**が設定(1)される時にだけ書けます。UDRENフラグが設定(1)されない時にUDRnへ書かれたデータはUSART送信部によって無視されます。送信部が許可されて送信緩衝部にデータが書かれると、送信部は送信移動レジスタが空きの時にそのデータを送信移動レジスタへ設定(移動)します。その後データはTXDnピンで直列送信されます。

受信緩衝部は2段のFIFOから成ります。このFIFOは受信緩衝部がアクセスされる毎にその状態を切り替えます。この受信緩衝部の動きのため、この位置に読み-修正-書き(リード モデファイ ライト)命令(SBIとCBI)を使えません。ビット検査(SBICとSBIS)命令もFIFOの状態を換えるので、これらの命令を使う時は注意してください。(訳注:適用範囲外命令のため、この注意は不適切です。)

### 18.10.2. USART制御/状態レジスタA (USART Control and Status Register A) UCSRnA

ビット (\$C0)	7	6	5	4	3	2	1	0	
	RXCn	TXCn	UDREN	FEn	DORn	UPEn	U2Xn	MPCMn	UCSRnA
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 - RXCn : 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RXCnフラグは0になります。RXCnフラグは受信完了割り込みを発生するのに使えます(USART制御/状態レジスタB(UCSRnB)の**受信完了割り込み許可(RXCIEn)ビット**をご覧ください)。

#### • ビット6 - TXCn : 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(UDRn)に新規データが現存しない時に設定(1)されます。TXCnフラグは送信完了割り込みが実行されるとき、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。TXCnフラグは送信完了割り込みを発生できます(UCSRnBの**送信完了割り込み許可(TXCIEn)ビット**をご覧ください)。

#### • ビット5 - UDREN : USART送信データレジスタ空きフラグ (USART Data Register Empty)

UDRENフラグは送信緩衝部(UDRn)が新規データを受け取る準備ができているかどうかを示します。UDRENが1ならば緩衝部は空で、従って書かれる準備ができています。UDRENフラグは送信緩衝部空き割り込みを発生できます(UCSRnBの**送信データレジスタ空き割り込み許可(UDRIEn)ビット**をご覧ください)。送信部が準備できているのを示すため、リセット後のUDRENは設定(1)です。

#### • ビット4 - FEn : フレーミング異常フラグ (Framing Error)

受信緩衝部の次データが受信した時にフレーミング異常(換言すると、受信緩衝部で次データの最初の(第1)停止ビットがLow)だった場合、このビットが設定(1)されます。このフラグは受信緩衝部(UDRn)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FEnフラグは0です。UCSRnAに書くとき、常にこのビットを0に設定してください。

#### • ビット3 - DORn : データオーバラン発生フラグ (Data OverRun)

このビットはオーバラン状態が検出されると設定(1)されます。受信緩衝部(2フレーム分)が一杯で、新規フレームが受信移動レジスタ内で待機中に新規開始ビットが検出されるとデータオーバランが起こります。UCSRnAに書くとき、常にこのビットを0に設定してください。

#### • ビット2 - UPEn : パリティ誤りフラグ (USART Parity Error)

受信緩衝部の次データが受信した時にパリティ異常があり、その時点でパリティ検査が許可されていれば(UPMn1=1)、このビットが設定(1)されます。このフラグは受信緩衝部(UDRn)が読まれるまで有効です。UCSRnAに書くとき、常にこのビットを0に設定してください。

#### • ビット1 - U2Xn : 倍速許可 (Double the USART Transmission Speed)

このビットは非同期動作でだけ有効です。同期動作を使うとき、このビットに0を書いてください。

このビットに1を書くことはボーレート分周器の分周値を16から8に減らして事実上、非同期通信の転送速度を倍にします。

#### • ビット0 - MPCMn : 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。MPCMn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は無視されます。送信部はMPCMn設定に影響されません。より多くの詳細情報については122頁の「**複数プロセッサ通信動作**」をご覧ください。

### 18.10.3. USART制御/状態レジスタB (USART Control and Status Register B) UCSRnB

ビット (\$C1)	7	6	5	4	3	2	1	0	
	RXCIE <sub>n</sub>	TXCIE <sub>n</sub>	UDRIE <sub>n</sub>	RXEN <sub>n</sub>	TXEN <sub>n</sub>	UCSZ <sub>n2</sub>	RXB8 <sub>n</sub>	TXB8 <sub>n</sub>	UCSRnB
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<sub>n</sub> : 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(UCSRnA)の受信完了(RXC<sub>n</sub>)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<sub>n</sub>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、UCSRnAでRXC<sub>n</sub>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<sub>n</sub> : 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(UCSRnA)の送信完了(TXC<sub>n</sub>)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<sub>n</sub>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、UCSRnAでTXC<sub>n</sub>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<sub>n</sub> : 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(UCSRnA)の送信データレジスタ空き(UDRE<sub>n</sub>)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<sub>n</sub>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、UCSRnAでUDRE<sub>n</sub>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<sub>n</sub> : 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。受信部は許可されるとRXD<sub>n</sub>ピンの標準ポート動作を無効にします。受信の禁止は受信緩衝部を破棄し、フレーミング異常(FE<sub>n</sub>)、オーバラン(DOR<sub>n</sub>)、パリティ誤り(UPE<sub>n</sub>)のフラグを無効にします。

• **ビット3 – TXEN<sub>n</sub> : 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。送信部は許可されるとTXD<sub>n</sub>ピンの標準ポート動作を無効にします。送信の禁止(TXEN<sub>n</sub>=0書き込み)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない)まで有効になりません。禁止したとき、送信部はもはやTXD<sub>n</sub>ポートの標準I/O機能を無効にしません。

• **ビット2 – UCSZ<sub>n2</sub> : データビット長選択2 (Character Size)**

USART制御/状態レジスタC(UCSRnC)のUCSZ<sub>n1,0</sub>ビットと組み合わせたUCSZ<sub>n2</sub>ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

• **ビット1 – RXB8<sub>n</sub> : 受信データビット8 (Receive Data Bit 8)**

RXB8<sub>n</sub>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<sub>n</sub>から下位ビットを読む前に読んでください。

• **ビット0 – TXB8<sub>n</sub> : 送信データビット8 (Transmit Data Bit 8)**

TXB8<sub>n</sub>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<sub>n</sub>へ下位ビットを書く前に書いてください。

### 18.10.4. USART制御/状態レジスタC (USART Control and Status Register C) UCSRnC

ビット (\$C2)	7	6	5	4	3	2	1	0	
	–	UMSEL <sub>n</sub>	UPM <sub>n1</sub>	UPM <sub>n0</sub>	USBS <sub>n</sub>	UCSZ <sub>n1</sub>	UCSZ <sub>n0</sub>	UCPOL <sub>n</sub>	UCSRnC
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	1	1	0	

• **ビット7 – Res : 予約 (Reserved) (訳注: 共通性のため本項追加)**

このビットは将来使用するために予約されています。将来のデバイスとの共通性を保つため、UCSRnCが書かれるとき、このビットは0が書かれなければなりません。

• **ビット6 – UMSEL<sub>n</sub> : USART動作選択 (USART Mode Select)**

このビットは非同期と同期の動作種別のどちらかを選びます。

表18-4. USART動作選択

UMSEL <sub>n</sub>	動作種別
0	非同期動作
1	同期動作



## •ビット5,4 – UPMn1,0 : パリティ選択 (Parity Mode)

これらのビットはパリティの発生と検査の許可と種別を設定します。許可した場合、送信部は各フレーム内での送信したデータビットのパリティを自動的に生成して送出します。受信部は到着データからパリティ値を生成し、UPMn0設定と比較します。不一致が検出されると、USART制御/状態レジスタA(UCSRnA)でパリティ誤り(UPEn)フラグが設定(1)されます。

表18-5. パリティ選択

UPMn1	UPMn0	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 – USBn : 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(誤補:常に第1停止ビットだけが有効)。

表18-6. 停止ビット選択

USBn	停止ビット数
0	1ビット
1	2ビット

## •ビット2,1 – UCSZn1,0 : データビット長選択 (Character Size)

USART制御/状態レジスタB(UCSRnB)のUCSZn2ビットと組み合わせたUCSZn1,0ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

表18-7. データビット長選択

UCSZn2~0	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 – UCPOln : クロック極性選択 (Clock Polarity)

このビットは同期動作に対してだけ使われます。非同期動作が使われるとき、このビットに0を書いてください。UCPOlnビットは同期クロック(XCKn)、データ出力変更、データ入力採取間の関係を設定します。

表18-8. XCKクロック極性選択

UCPOln	送信データ変更 (TXDnピン出力)	受信データ採取 (RXDnピン入力)
0	XCKnの上昇端	XCKnの下降端
1	XCKnの下降端	XCKnの上昇端

## 18.10.5. USARTボーレートレジスタ (USART Baud Rate Register) UBRRnH, UBRRnL (UBRRn)

ビット (\$C5)	15	14	13	12	11	10	9	8	UBRRnH
	–	–	–	–	UBRR11	UBRR10	UBRR9	UBRR8	
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	UBRRnL
	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

## •ビット15~12 – Res : 予約 (Reserved Bits)

これらのビットは将来使用するために予約されています。将来のデバイスとの共通性のため、UBRRnHが書かれるとき、これらのビットは0が書かれなければなりません。

## •ビット11~0 – UBRR11~0 : ボーレート分周値 (USART Baud Rate Register)

USARTのボーレートを含む12ビットレジスタです。UBRRnHがUSARTボーレートの上位4ビットを含み、UBRRnLが下位8ビットを含みます。ボーレートが変更されると、送受信部で進行中の転送は不正にされます。UBRRnL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。



## 18.11. ホーレート設定例

標準的なクリスタル発振子やセラミック振動子の周波数に対して最も一般的に使われる非同期動作のホーレートは表18-9のUBRRn設定を使うことによって生成できます。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(121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18-9. Xtal、ホーレート対UBRRnH,UBRRnL設定 (UBRR=UBRRnH:UBRRnL)

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18-3,18-4,4,5となっていますが、共通性から纏めて表18-9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18-9 (続き). Xtal、ホーレート対UBRRnH,UBRRnL設定 (UBRR=UBRRnH:UBRRnL)

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。



## 19.3. 機能説明

### 19.3.1. 3線動作

USIの3線動作は直列周辺インターフェース(SPI)の動作種別0と1に準拠していますが、従装置選択(SS)ピン機能を持ちません。けれども、この特性(機能)は必要ならばソフトウェアで実現できます。この動作で使われるピン名はDI,DO,USCKです。

図19-2. 3線動作簡略構成図

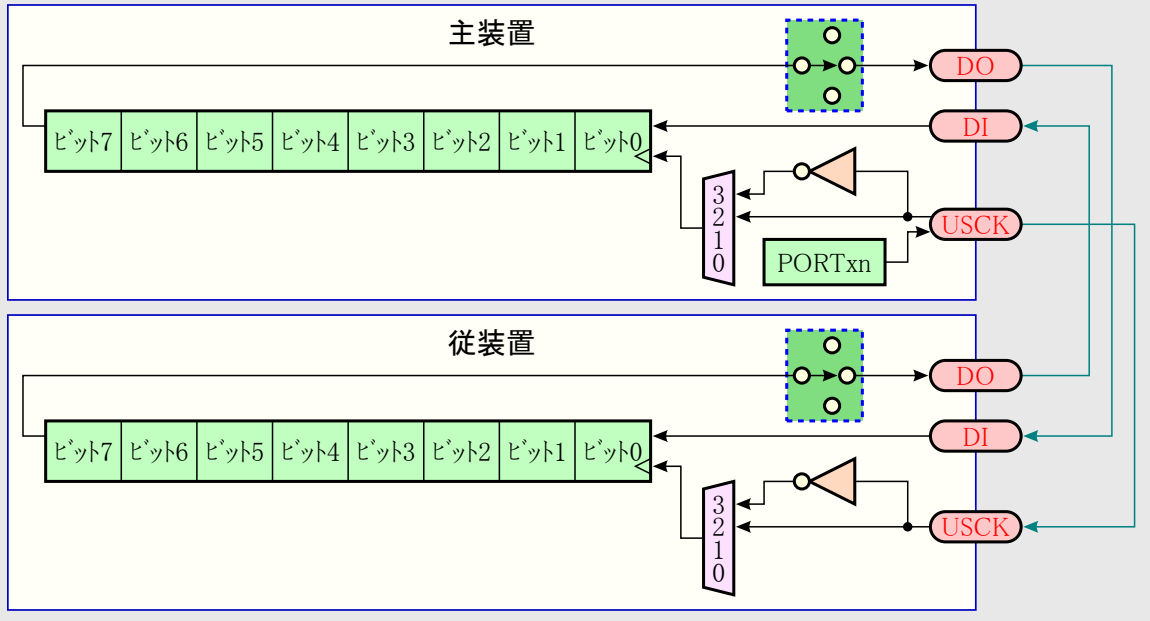
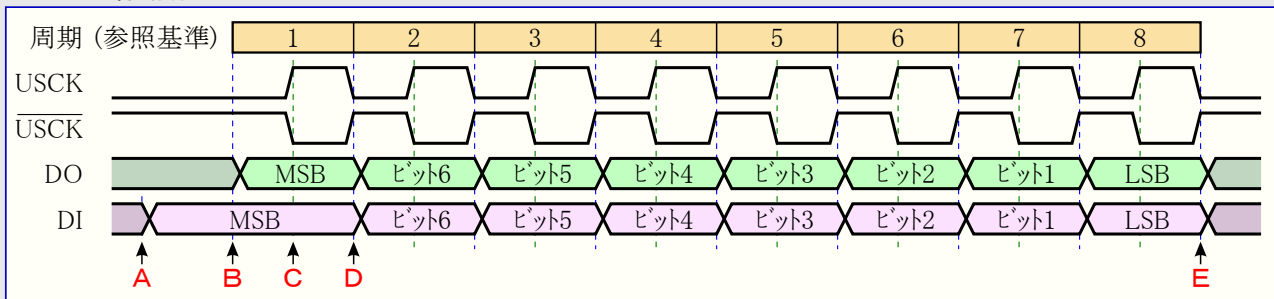


図19-2.は3線動作での2つのUSI部(一方は主装置、他方は従装置)動作を示します。このような方法で連結された2つの移動レジスタは8 USCKクロック後、各々のレジスタのデータが交換されます。この同じクロック(USCK)がUSIの4ビット計数器も増加(+1)します。従って計数器溢れ割り込み要求フラグ(USIOIF)は転送が完了される時を決めるのに使えます。このクロックはPORTレジスタ経由USCKピン1/0交互切り替え、またはUSICRのUSITCビットへの1書き込みによる主装置装置ソフトウェアによって生成されます。

図19-3. 3線動作タイミング



3線動作タイミングは図19-3.で示されます。図の最上部はUSCK周期参照基準です。これらの各周期に対して1ビットがUSI移動レジスタ(USIDR)に移動されます。USCKタイミングは両方の外部クロック動作について示されます。外部クロック動作0(USICS0=0)でのDIは上昇端で採取され、DOは下降端で変更(USIDRが1つ移動)されます。外部クロック動作1(USICS0=1)は外部クロック動作0に対して逆端を使用、換言すると下降端でデータを採取、上昇端で出力を変更します。USIクロック動作種別はSPIデータ動作種別0と1に対応します。

このタイミング図(図19-3.)を参照すると、バス転送は次の手順を含みます。

1. 主装置と従装置はデータ出力を設定し、使用規約に依存して出力駆動部を許可します(AとB)。この出力は直列データレジスタへの送信されるべきデータ書き込みによって設定します。出力の許可はポート方向レジスタの対応するビット設定によって行います。A点とB点はどんな特別な順番もありますが、両方共にデータが採取されるC点よりも最低1/2 USCK周期前でなければならないことに注意してください。これはデータ設定の必要条件を満足させるのを保証するために行わなければならないです。4ビット計数器は0にリセットします。
2. 主装置はUSCK線をソフトウェアで2度切り替えることによってクロックパルスを生成します(CとD)。主装置と従装置のデータ入力(DI)ピンのビット値は最初の端(エッジ)(C)でUSIによって採取され、データ出力は逆端(D)で変更されます。4ビット計数器は両端で計数します。
3. レジスタ(バイト)転送完了のために手順2.が8回繰り返されます。
4. 8クロックパルス(換言すると16クロック端)後、計数器が溢れて転送完了を示します。転送したバイトデータは新規転送が開始され得るのに先立って直ぐに処置されなければならないです。溢れ割り込みはプロセッサがアイドル動作に設定されているなら、プロセッサを起動復帰します。使用規約によって、従装置は直ぐに出力をHi-Z設定にできます。

### 19.3.2. SPI主装置操作例

次のコードはSPI主装置としてのUSI部使用法を実際に示します。

アセンブリ言語プログラム例			
SPIM:	STS	USIDR, R16	;送信データを設定
	LDI	R16, (1<<USIOIF)	;USIOIFビットのみ1値を取得
	STS	USISR, R16	;フラグ解除/計数器初期化
	LDI	R16, (1<<USIWM0)   (1<<USICS1)   (1<<USICLK)   (1<<USITC)	;3線動作クロック生成値を取得
;			
SPIM_LP:	STS	USICR, R16	;USCKクロック端発生
	LDS	R16, USISR	;USI状態フラグを取得
	SBRS	R16, USIOIF	;計数器溢れでスキップ
	RJMP	SPIM_LP	;計数器溢れまで継続
;			
	LDS	R16, USIDR	;受信データを取得
	RET		;呼び出し元へ復帰

このコードは9命令(+RET)のみ使用の容量最適化です。このコード例はDOとUSCKピンがDDREレジスタで出力として許可されると仮定します。この関数が呼び出される前にR16レジスタへ格納した値は従装置に転送され、転送が完了された時に従装置から受信したデータがR16レジスタに格納されて戻ります。

2行目と3行目の命令は計数器溢れ割り込み要求フラグ(USIOIF)を解除(0)し、USI 4ビット計数器値を解除(=0)します。4行目と6行目の命令は3線動作、上昇端移動レジスタクロック、USITCストロブ計数、USCK出力交互切り替えを設定します。この繰り返しは16回繰り返されます。

次のコードは最高速( $f_{USCK}=f_{CK}/4$ )でのSPI主装置としてのUSI部使用法を実際に示します。

アセンブリ言語プログラム例			
SPIM_F:	STS	USIDR, R16	;送信データを設定
	LDI	R16, (1<<USIWM0)   (0<<USICS0)   (1<<USITC)	;3線動作初期値を取得
	LDI	R17, (1<<USIWM0)   (0<<USICS0)   (1<<USITC)   (1<<USICLK)	;3線動作クロック生成値を取得
;			
	STS	USICR, R16	;MSB転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット6転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット5転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット4転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット3転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット2転送
	STS	USICR, R17	;
	STS	USICR, R16	;ビット1転送
	STS	USICR, R17	;
	STS	USICR, R16	;LSB転送
	STS	USICR, R17	;
;			
	LDS	R16, USIDR	;受信データを取得
	RET		;呼び出し元へ復帰



## 19.3.3. SPI従装置操作例

次のコードはSPI従装置としてのUSI部使用法を実際に示します。

### アセンブリ言語プログラム例

```
init:      LDI      R16, (1<<USIWM0) | (1<<USICS1)      ;3線動作,外部クロック値を取得
          STS      USICR, R16                          ;3線動作,外部クロック設定

SPIS:      STS      USIDR, R16                          ;送信データを設定
          LDI      R16, (1<<USIOIF)                    ;USIOIFビットのみ1値を取得
          STS      USISR, R16                          ;フラグ解除/計数器初期化

;
SPIS_LP:   LDS      R16, USISR                          ;USI状態フラグを取得
          SBRS     R16, USIOIF                          ;計数器溢れでスキップ
          RJMP     SPIS_LP                             ;計数器溢れまで継続

;
          LDS      R16, USIDR                          ;受信データを取得
          RET                                           ;呼び出し元へ復帰
```

このコードは9命令(+RET)のみ使用の容量最適化です。このコード例はDDREレジスタでDOピンが出力、USCKピンが入力として設定されると仮定します。この関数が呼び出される前にR16レジスタへ格納した値は主装置に転送され、転送が完了された時に主装置から受信したデータがR16レジスタに格納されて戻ります。

最初の2命令は初期化用だけで、一度だけ実行されるのを必要とすることに注意してください。これらの命令は3線動作と上昇端移動レジスタクロックを設定します。この繰り返しはUSI計数器溢れフラグが設定(1)されるまで繰り返されます。

## 19.3.4. 2線動作

USIの2線動作はI<sup>2</sup>C(TWI)バス規約に準拠していますが、出力のスレーブ制限と入力雑音濾波器がありません。この動作で使われるピン名はSCLとSDAです。

図19-4. 2線動作簡略構成図

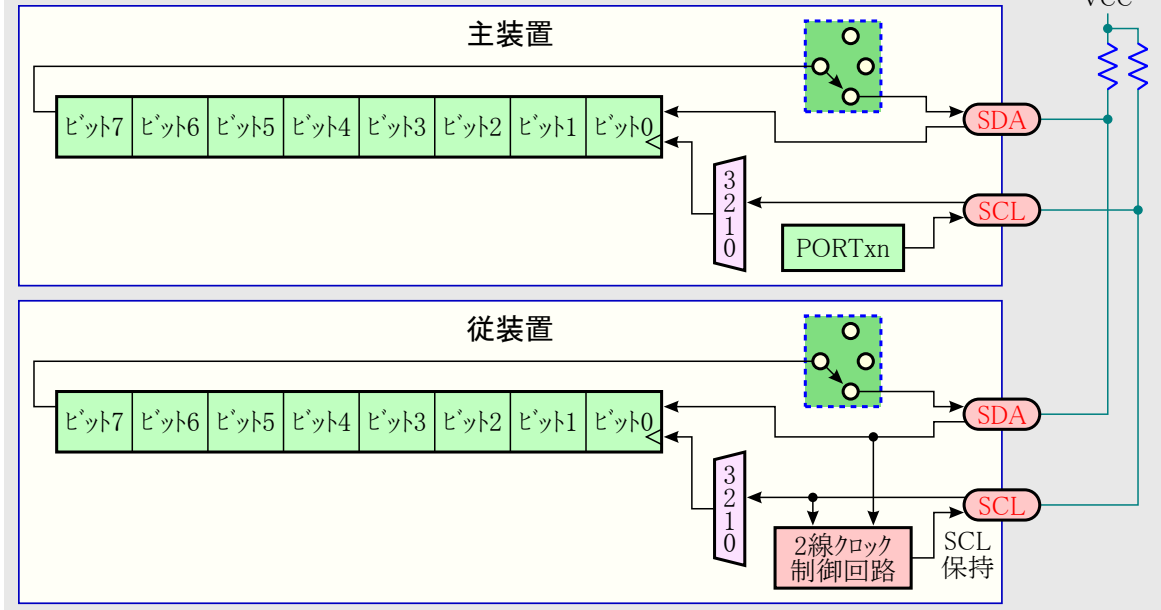


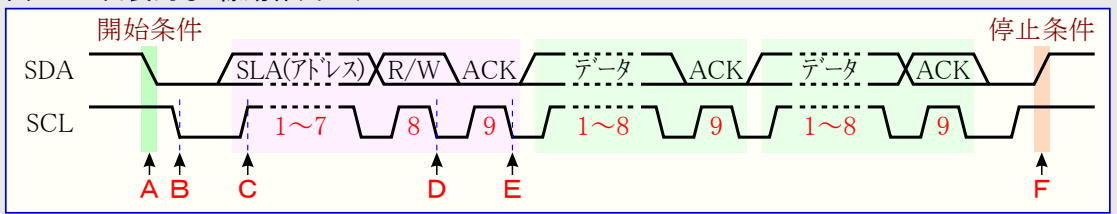
図19-4は2線動作での2つのUSI部（一方は主装置、他方は従装置）動作を示します。システム動作が使用通信構造に大きく依存するために示されるのは物理層のみです。この段階での主装置動作と従装置動作間の主な違いは常に主装置によって行われる直列クロック生成と、従装置だけがクロック制御部(回路)を使うことです。クロック生成はソフトウェアで実行されなければなりません、移動動作は両装置で自動的に行われます。データの移動に関してこの動作種別で下降端でのクロックだけを実際に使うことに注意してください。従装置はSCLクロックをLowに強制することにより、転送の開始と終了で待ち状態(ウェイト ステート)を挿入できます。これは(主装置が)上昇端を生成してしまった後に主装置はSCL線が実際に開放されたかを常に検査しなければならないことを意味します。

このクロックが(4ビット)計数器も増加(+1)するため、計数器溢れ(USIOIF)は転送が完了されたことを示すのに使えます。このクロックは主装置によるPORTEレジスタ経由のSCLピン交互切り替えによって生成されます。

データ方向は物理層によって与えられません。データの流れを制御するためにTWIバスで使ったような或る種の規約が実装されなければなりません。

このタイミング図(図19-5.)を参照すると、バス転送は次の手順を含みます。

図19-5. 代表的な2線動作タイミング



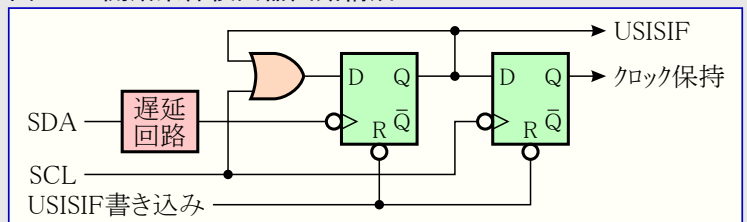
1. 開始条件は主装置によるSCL線High中のSDA線Low強制によって生成されます(A)。SDAは移動レジスタのビット7への0書き込みか、またはPORTレジスタで対応ビットを0に設定のどちらかによって強制実行できます。出力が許可されるにはデータ方向(DDR)レジスタのビットが1に設定されなければならないことに注意してください。従装置の開始条件検出器論理回路(図19-6.参照)が開始条件を検出してUSISIFフラグを設定(1)します。このフラグは必要ならば割り込みを発生できます。
2. 加えて開始条件検出器は主装置がSCL線で上昇端を強制してしまった後にSCL線をLowに保持します(B)。これはアドレスを受信するために移動レジスタを設定する前に休止形態から起動復帰または他の処理を完了することを従装置に許します。これ(保持解除)は(4ビット)計数器をリセット(=0)して開始条件検出フラグ(USISIF)を解除(0)することによって行われます。
3. 主装置は転送されるべき最初のビットを設定してSCL線を開放します(C)。従装置はSCLクロックの上昇端でデータを採取して移動レジスタへ移動します。
4. 従装置アドレスとデータ方向(R/W)を含む8ビットが転送された後、従装置の(4ビット)計数器が溢れてSCL線がLowを強制されます(D)。従装置は主装置の1つがアドレス指定してしまわない場合、SCL線を開放して新規開始条件を待ちます。
5. 従装置がアドレス指定されると、再びSCL線をLowに保持する前の応答(ACK)周期の間中、SDA線をLowに保持します(換言するとD点でSCLを開放する前に(4ビット)計数器レジスタは14に設定されなければなりません)。R/Wビットに依存して主装置が従装置は出力を許可します。このビットが設定(1)の場合、主装置読み込み動作が進行中です(換言すると従装置がSDA線を駆動します)。従装置は応答後(E)にSCL線をLowに保持することができます。
6. 主装置によって停止条件(F)か新規開始条件(再送開始条件)が与えられるまで全て同方向で複数バイトが直ぐに転送できます。従装置はより多くのデータを受信できない場合、最後に受信したバイトデータに回答しません。主装置が読み込み動作を行うとき、最後のバイトが転送された後の応答ビットをLowに強制することによってこの動作を終了しなければなりません。

### 19.3.5. 開始条件検出器

開始条件検出器は図19-6.で示されます。SDA線はSCL線の有効な採取を保証するために(50~300nsの範囲で)遅延されます。開始条件検出器は2線動作でだけ許可されます。

開始条件検出器は非同期に動作し、従ってパワーダウン休止動作からプロセッサを起動できます。けれども使う規約がSCL保持時間で制限を持つかもしれません。従ってこの場合にこの特徴(機能)を使う時にCKSELヒューズで設定する発振器起動時間(17頁の「クロックシステムとその配給」参照)も考慮内に取り入れられなければなりません。より多くの詳細については133頁のUSISIFフラグ記述を参照してください。

図19-6. 開始条件検出器回路構成



### 19.3.6. クロック速度の考察

SCLとUSCKに対する最大周波数は $f_{CK}/4$ です。これは2線と3線の両動作での最大データ転送と受信速度でもあります。2線従装置動作では従装置が更にデータを受信する準備ができるまで、2線クロック制御部がSCLを保持します。これは2線動作で実データ速度を減らすかもしれません。

## 19.4. USIでの代替使用

### 19.4.1. 半二重非同期データ転送

USI部を直列通信に使わないなら、柔軟な設計のため、代替処理を行うように設定できます。

3線動作で移動レジスタを使うことによってソフトウェアだけよりも簡潔で高性能のUART実装が可能です。

### 19.4.2. 4ビット計数器

4ビット計数器は溢れ割り込みとの自立型計数器として使えます。この計数器が外部的にクロック駆動される場合、両クロック端が増加(+1)を発生することに注意してください。

### 19.4.3. 12ビット計数器

USIの4ビット計数器とタイマ/カウンタ0の組み合わせは12ビット計数器としての使用を許します。

### 19.4.4. 端(エッジ)起動外部割り込み

4ビット計数器を最大値(\$F)に設定することによって追加外部割り込みとして機能できます。溢れフラグと割り込み許可ビットはこの外部割り込みに対して使われます。この特徴(機能)はUSIC1ビット(=1)によって選ばれます。

### 19.4.5. ソフトウェア割り込み

4ビット計数器の溢れ割り込みはクロックストロブによって起動するソフトウェア割り込みとして使えます。

## 19.5. USI用 レジスタ

### 19.5.1. USIデータレジスタ (USI Data Register) USIDR

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

USIは直列(移動)レジスタの緩衝をしません、換言するとデータレジスタ(USIDR)をアクセスする時に直列レジスタは直接的にアクセスされます。レジスタが書かれるのと同じ周期で直列クロックが起こると、レジスタは書いた値を含み、移動は実行されません。(左)移動動作はUSIC1,0ビット設定に従って実行されます。この移動動作は外部クロック端、**タイマ/カウンタ0比較一致**により、またはUSICLKストローブビットを使うソフトウェアによって直接的に制御できます。2または3線動作種別が選ばれない(USIWM1,0=00)時でも、外部データ入力(DI/SDA)と外部クロック入力(USCK/SCL)の両方が未だ移動レジスタによって使われ得ることに注意してください。

動作種別に従って使う出力ピン(DOまたはSDA)は出力ラッチを経由してデータレジスタの最上位ビット(ビット7)に接続されます。この出力ラッチは外部クロック元が選ばれる(USICS1=1)時に直列クロック周期の前半中が透過(トランスパレント)で、内部クロック元が使われる(USICS1=0)時に定常的な透過です。ラッチが透過である限り、新規MSBが書かれると、出力は直ちに更新されます。このラッチは違う(逆の)クロック端でデータ入力が採取され、データ出力が更新されるのを保証します。

移動レジスタからの出力データを許可するために、ピンに対応するデータ方向レジスタ(DDxn)が1に設定されなければならないことに注意してください。

### 19.5.2. USI状態レジスタ (USI Status Register) USISR

ビット (\$B9)	7	6	5	4	3	2	1	0	
	USISIF	USIOIF	USIPF	USIDC	USICNT3	USICNT2	USICNT1	USICNT0	USISR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この状態レジスタは割り込み要求フラグ、状態フラグ、計数値を含みます。

#### • ビット7 – USISIF : 開始条件検出割り込み要求フラグ (Start Condition Interrupt Flag)

2線動作が選ばれた時に**開始条件**が検出されると、USISIFフラグが設定(1)されます。出力禁止動作または3線動作が選ばれ、(USICS1,0=11且つUSICLK=0)または(USICS1,0=10且つUSICLK=0)の時にUSCKピンのどの端(エッジ)もがこのフラグを設定(1)します。

**ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUSI制御レジスタ(USICR)で開始条件検出許可(USISIE)ビット**が設定(1)されている間にこのフラグが設定(1)されると、割り込みが生成されます。このフラグはUSISIFビットへの論理1書き込みによってのみ解除(0)されます。このビットの解除(0)は2線動作でのSCLの**開始条件保持**を解除(開放)します。

開始条件検出割り込みは全ての**休止形態**からプロセッサを起動します。

#### • ビット6 – USIOIF : 計数器溢れ割り込み要求フラグ (Counter Overflow Interrupt Flag)

このフラグは4ビット計数器が溢れると(換言すると15から0への遷移で)設定(1)されます。SREGの全割り込み許可(I)ビットとUSICRで**計数器溢れ割り込み許可(USIOIE)ビット**が設定(1)されている間にこのフラグが設定(1)されると、割り込みが生成されます。本フラグはUSIOIFビットへの論理1書き込みによってのみ解除(0)されます。本ビットの解除(0)は2線動作でのSCLの**計数器溢れ保持**を解除(開放)します。

計数器溢れ割り込みは**アイドル休止動作**からプロセッサを起動します。

#### • ビット5 – USIPF : 停止条件検出フラグ (Stop Condition Flag)

2線動作が選ばれた時に**停止条件**が検出されると、USIPFフラグが設定(1)されます。このフラグはこのビットへの1書き込みによって解除(0)されます。これが割り込み要求フラグでないことに注意してください。この合図は2線バス主装置調停の実装時に有用です。

#### • ビット4 – USIDC : 出力データ衝突フラグ (Data Output Collision Flag)

移動レジスタのビット7(MSB)が物理ピン値と異なる時にこのビットは論理1です。このフラグは2線動作が使われる時にだけ有効です。この合図は2線バス主装置調停の実装時に有用です。

#### • ビット3~0 – USICNT3~0 : 計数器値 (Counter Value)

これらのビットは現在の4ビット計数器値を反映します。この4ビット計数器値はCPUによって直接的に読み書きできます。

この4ビット計数器は外部クロック端検出器、タイマ/カウンタ0比較一致、USICLKまたはUSITCストローブビットを使うソフトウェアのどれかによって生成した各クロックに対して1つ増加します。このクロック元はUSICS1,0ビット設定に依存します。外部クロック動作に対してUSITCストローブビットへの書き込みによって生成されることをクロックに許す、特別な特徴が付加されます。この特徴は外部クロック元(USICS1=1)設定中のUSICLKへの1書き込みによって許可されます。

出力禁止動作(USIWM1,0=00)が選ばれている時でも、外部クロック入力(USCK/SCL)が未だ計数器によって使われることに注意してください。

### 19.5.3. USI制御レジスタ (USI Control Register) USICR

ビット (\$B8)	7	6	5	4	3	2	1	0	
	USISIE	USIOIE	USIWM1	USIWM0	USICS1	USICS0	USICLK	USITC	USICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	W	W	
初期値	0	0	0	0	0	0	0	0	

この制御レジスタは割り込み許可制御、動作種別設定、クロック選択設定、クロック スロープを含みます。

#### • ビット7 – USISIE : 開始条件検出割り込み許可 (Start Condition Interrupt Enable)

このビットを1に設定することが開始条件検出割り込みを許可します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUSISIEビットが設定(1)される時に保留割り込みなら、これは直ちに実行されます。より多くの詳細については133頁の開始条件検出割り込み要求フラグ(USISIF)記述を参照してください。

#### • ビット6 – USIOIE : 計数器溢れ割り込み許可 (Counter Overflow Interrupt Enable)

このビットを1に設定することが計数器溢れ割り込みを許可します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとUSIOIEビットが設定(1)される時に保留割り込みなら、これは直ちに実行されます。より多くの詳細については133頁の計数器溢れ割り込み要求フラグ(USIOIF)記述を参照してください。

#### ビット5,4 – USIWM1,0 : 動作種別選択 (Wire Mode)

これらのビットは使われるべき動作種別(2または3線動作)を設定します。基本的に出力の機能だけがこれらのビットによって影響を及ぼされます。データとクロック入力を選んだ動作種別によって影響を及ぼされず、常に同じ機能を持ちます。従って出力が禁止される時でも移動レジスタと計数器は外部的にクロック駆動でき、入力データが採取されます。USI操作とUSIWM1,0間の関連は表19-1.で要約されます。

表19-1. USI操作とUSIWM1,0の関係

USIWM1	USIWM0	説明
0	0	出力、クロック保持、開始条件検出器は禁止。ポートピンは標準として動作。
0	1	3線動作。DO,DI,USCKピンを使用。 データ出力(DO)ピンはこの動作でPORTレジスタの対応ビットを無視します。けれども対応するDDRビットは未だデータ方向を制御します。ポートピンが入力として設定(DDxn=0)されると、そのピンのプルアップはPORTビットによって制御されます。 データ入力(DI)と直列クロック(USCK)ピンは標準ポート操作に影響を及ぼしません。主装置として動作する時のクロックパルスはデータ方向が出力に設定されている(DDxn=1)間のPORTレジスタの交互切り替えによってソフトウェアで生成されます。USICRのUSITCビットがこの目的に使えます。
1	0	2線動作。SDA(DI)とSCL(USCK)ピンを使用。(注1) 直列データ(SDA)と直列クロック(SCL)ピンは双方向でオープンドレイン出力駆動を使います。この出力駆動部はDDRレジスタでSDAとSCLに対応するビットの設定(=1)によって許可されます。 SDAピンに対して出力駆動部が許可されると、出力駆動部は移動レジスタの出力またはPORTレジスタの対応ビットが0ならばSDA線をLowに強制します。さもないと、SDA線は駆動されません(換言すると開放されます)。SCLピン出力駆動部が許可されると、SCL線はPORTレジスタの対応ビットが0なら、または開始条件検出器によってLowを強制されます。さもないとSCL線は駆動されません。 SCL線は出力が許可され、開始条件検出器が開始条件を検出すると、Lowに保持されます。開始条件検出フラグ(USISIF)の解除(0)がその線を開放します。SDAとSCLピン入力はこの動作の許可によって影響を及ぼされません。SDAとSCLピンのプルアップは2線動作で禁止されます。
1	1	2線動作。SDAとSCLピンを使用。 SCL線は計数器溢れが起こる時にもLowへ保持され、計数器溢れフラグ(USIOIF)が解除(0)されるまでLowに保持されることを除いて、上で記述される2線動作と同じ動作です。

注1: DIとUSCKピンは動作種別間での混乱を避けるため、各々直列データ(SDA)と直列クロック(SCL)に改称されます。



## • ビット3,2 – USICS1,0 : クロック選択 (Clock Source Select)

これらのビットは移動レジスタと計数器に対するクロック元を設定します。外部クロック元(USCK/SCL)を使う時にデータ出力ラッチはデータ入力(DI/SDA)の採取の逆端で出力が変更されるのを保証します。タイマ/カウンタ0比較一致またはソフトウェア ストローブが選ばれると、出力ラッチは透過で、従って出力は直ちに變更されます。USICS1,0ビットの解除(=00)がソフトウェア ストローブ選択を許可します。この選択を使う時のUSICLKビットへの1書き込みは移動レジスタと計数器の両方をクロック駆動します。外部クロック元(USICS1=1)に対してUSICLKビットはもはやストローブとして使われませんが、外部クロックとUSITCストローブ ビットによるソフトウェア クロック間を選びます。

表19-2.は移動レジスタと4ビット計数器に対して使われるクロック元とUSICS1,0、USICLK設定間の関連を示します。

表19-2. USICS1,0とUSICLK設定との関係

USICS1	USICS0	USICLK	移動レジスタ クロック元	4ビット計数器クロック元
0	0	0	クロックなし	クロックなし
		1	ソフトウェア クロック ストローブ (USICLK)	ソフトウェア クロック ストローブ (USICLK)
	1	X	タイマ/カウンタ0 比較一致	タイマ/カウンタ0 比較一致
1	0	0	外部クロック上昇端	外部クロック両端
	1		外部クロック下降端	
	0	1	外部クロック上昇端	ソフトウェア クロック ストローブ (USITC)
	1		外部クロック下降端	

## • ビット1 – USICLK : クロック ストローブ (Clock Strobe)

ソフトウェア ストローブ選択が選ばれるように行うことによってUSICS1,0ビットがゼロ(=00)に設定されるなら、このビット位置への1書き込みは移動レジスタを1段移動して計数器を1つ増加します。このクロック ストローブが実行される時、直ちに、換言すると同じ命令周期内で出力が替わります。移動レジスタ内に移動した値は直前の命令周期で採取されます。このビットは0として読みます。

外部クロック元が選ばれる(USICS1=1)と、USICLK機能はクロック ストローブからクロック選択レジスタに変更されます。この場合のUSICLKビットの設定(=1)は4ビット計数器に対するクロック元としてのUSITCストローブを選びます(表19-2.参照)。

## • ビット0 – USITC : クロック値切り替え (Toggle Clock Port Pin)

このビット位置に1を書くことはUSCK/SCL値を0から1、または1から0のどちらかへ切り替えます。この切り替えはデータ方向レジスタでの設定と無関係ですが、PORT値がピンで見られるべきなら、DDE4が出力として設定(=1)されなければなりません。この特徴は主装置を実現する時の容易なクロック生成を許します。このビットは0として読みます。

外部クロック元が選ばれ(USICS1=1)、USICLKビットが1に設定されると、USITCストローブ ビットへの書き込みは直接的に4ビット計数器をクロック駆動します。これは主装置として動作する時に転送が行われる時の早い検出を許します。



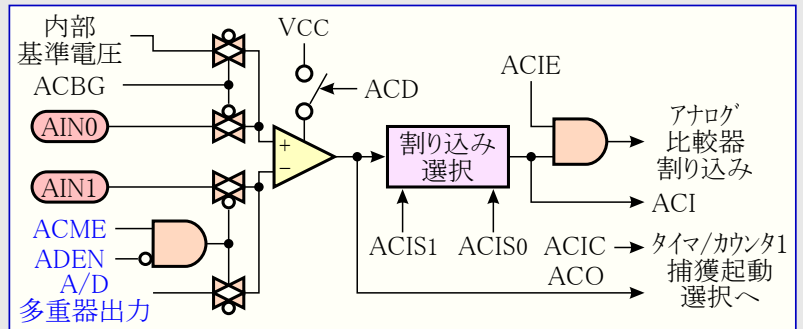
## 20. アナログ比較器

### 20.1. 概要

アナログ比較器は非反転入力AIN0ピンと反転入力AIN1ピンの入力値を比較します。非反転AIN0ピンの電圧が反転AIN1ピンの電圧よりも高い時に**ACSRのアナログ比較器出力(ACO)ビット**が設定(1)されます。この比較器出力はタイマ/カウンタ1の捕獲機能を起動するように設定できます。加えて、この比較器はアナログ比較器専用の独立した割り込みを起動できます。使用者は比較器出力の上昇端、下降端、またはその両方で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20-1.で示されます。

ADC入力(A/D多重器出力)の使用を可能とするには、26頁の「電力削減レジスタ(PRR)」のPRADCBビットが論理0を書かれることで(電力削減を)禁止されなければなりません。

図20-1. アナログ比較器部構成図



注: A/D多重器出力については表20-1.をご覧ください。  
アナログ比較器ピン配置については2頁の「ピン配置」と46頁の表13-8.を参照してください。

### 20.2. アナログ比較器入力選択

アナログ比較器への反転入力を取り替えるのにADC7~0のどれかを選ぶことができます。A/D変換の多重器がこの入力選択に使われ、従ってこの機能を利用するためにA/D変換部がOFF(動作禁止)にされなければなりません。ADCSRBの**アナログ比較器多重器許可(ACME)ビット**が設定(1)され、A/D変換部がOFF(ADCSRAのADENビットが0)にされていれば、表20-1.で示されるように**ADMUXのチャネル選択(MUX2~0)ビット**がアナログ比較器への反転入力を取り替えるための入力ピンを選びます。ACMEが解除(0)、またはADENが設定(1)されると、AIN1がアナログ比較器への反転入力に印加されます。

表20-1. アナログ比較器反転入力選択

ACME	ADEN	MUX2~0	アナログ比較器反転入力	備考
0	x	x x x	AIN1	
1	1	x x x	AIN1	
	0	0 0 0	ADC0	
		0 0 1	ADC1	
		0 1 0	ADC2	
		0 1 1	ADC3	
		1 0 0	ADC4	
		1 0 1	ADC5	
		1 1 0	ADC6	
		1 1 1	ADC7	

### 20.3. アナログ比較器用レジスタ

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

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

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

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

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

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	0	0	0	0	0	

### • ビット7 – ACD : アナログ比較器禁止 (Analog Comparator Disable)

このビットが論理1を書かれると、アナログ比較器への電力がOFFにされます。このビットはアナログ比較器をOFFにするために何時でも設定(1)できます。これは活動動作やアイドル動作で電力消費を削減します。ACDビットを変更する時にACSR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こり得ます。

### • ビット6 – ACBG : 基準電圧選択 (Analog Comparator Bandgap Select)

このビットが設定(1)されると、内部基準電圧(公称1.1V)がアナログ比較器への非反転入力に置き換わります。本ビットが解除(0)されると、AIN0がアナログ比較器の非反転入力に印加されます。内部基準電圧がアナログ比較器への入力として使われるとき、電圧の安定に一定時間を必要とします。安定(待機を)しない場合、不正な比較になるかもしれません。29ページの「[内部基準電圧](#)」をご覧ください。

### • ビット5 – ACO : アナログ比較器出力 (Analog Comparator Output)

アナログ比較器の出力は同期化され、その後に直接ACOへ接続されます。この同期化は1～2クロック周期の遅延をもたらします。

### • ビット4 – ACI : アナログ比較器割り込み要求フラグ (Analog Comparator Interrupt Flag)

このビットは比較器出力での出来事がACSRの[アナログ比較器割り込み条件\(ACIS1,0\)ビット](#)によって定義した割り込み方法で起動する時に設定(1)されます。ACSRのアナログ比較器割り込み許可(ACIE)ビットが設定(1)され、[ステータスレジスタ\(SREG\)の全割り込み許可\(I\)ビット](#)が設定(1)されていると、アナログ比較器割り込みルーチンが実行されます。対応する割り込み処理ベクタを実行すると、ACIはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもACIは解除(0)されます。

### • ビット3 – ACIE : アナログ比較器割り込み許可 (Analog Comparator Interrupt Enable)

ACIEビットが論理1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みが活性(有効)にされます。論理0を書かれると、この割り込みは禁止されます。

### • ビット2 – ACIC : アナログ比較器捕獲起動許可 (Analog Comparator Input Capture Enable)

論理1を書かれると、このビットはアナログ比較器によって起動されるタイマ/カウンタ1の捕獲機能を許可します。この場合、比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの[雑音消去機能](#)と[端\(エッジ\)選択機能](#)を利用させる[捕獲入力前置論理回路](#)へ直接的に接続されます。論理0を書かれると、アナログ比較器と捕獲機能間の接続は存在しません。比較器がタイマ/カウンタ1捕獲割り込みを起動するには、[タイマ/カウンタ1割り込み許可レジスタ\(TIMSK1\)の捕獲割り込み許可\(ICIE1\)ビット](#)が設定(1)されなければなりません。

### • ビット1,0 – ACIS1,0 : アナログ比較器割り込み条件 (Analog Comparator Interrupt Mode Select)

これらのビットは比較器のどの事象がアナログ比較器割り込みを起動するかを決めます。各種設定は表20-2.で示されます。

ACIS1,ACIS0ビットを変更する時にACSRのアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、これらのビットが変更される時に割り込みが起き得ます。

表20-2. アナログ比較器割り込み条件選択

ACIS1	ACIS0	割り込み発生条件
0	0	比較器出力の変移(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

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

ビット	7	6	5	4	3	2	1	0	
(\$7F)	–	–	–	–	–	–	AIN1D	AIN0D	DIDR1
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット1,0 – AIN1D,AIN0D : AIN1,AIN0 デジタル入力禁止 (AIN1,AIN0 Digital Input Disable)

このビットが論理1を書かれると、AIN1/0ピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、対応するポート入力レジスタのビット(PINx)は常に0として読みます。AIN1/0ピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされない時にデジタル入力緩衝部での消費電力を削減するため、このビットは論理1を書かれるべきです。

## 21. A/D変換器

### 21.1. 特徴

- 10ビット分解能
- 積分非直線性誤差0.5 LSB
- 絶対精度 $\pm 2$  LSB
- 変換時間13~260 $\mu$ s (50kHz~1MHz変換クロック)
- 最大分解能で76.9kSPS(採取/s)まで
- 8チャンネルのシングルエンド入力多重器内蔵
- 7チャンネルの差動入力チャンネル (訳注: 共通性から追加)
- A/D変換結果読み出しに対する任意の左揃え
- 0~VCC A/D変換入力電圧範囲
- 選択可能な1.1V A/D変換基準電圧
- 連続と単独の変換動作
- 割り込み元の自動起動によるA/D変換開始
- A/D変換完了割り込み
- 休止形態雑音低減機能

### 21.2. 概要

ATmega325P/3250Pは10ビット逐次比較A/D変換器が特徴です。このA/D変換器はポートFのピンから構成された8つのシングルエンド電圧入力を許す8チャンネルアナログ多重器に接続されます。このシングルエンド電圧入力は0V(GND)が基準です。

デバイスは16通りの差動電圧入力も支援します。7つの差動アナログ入力は共通反転端子(ADC1)を共用し、一方他のADC入力は非反転入力端子として選べます。(訳注: 共通性から直前の2行追加)

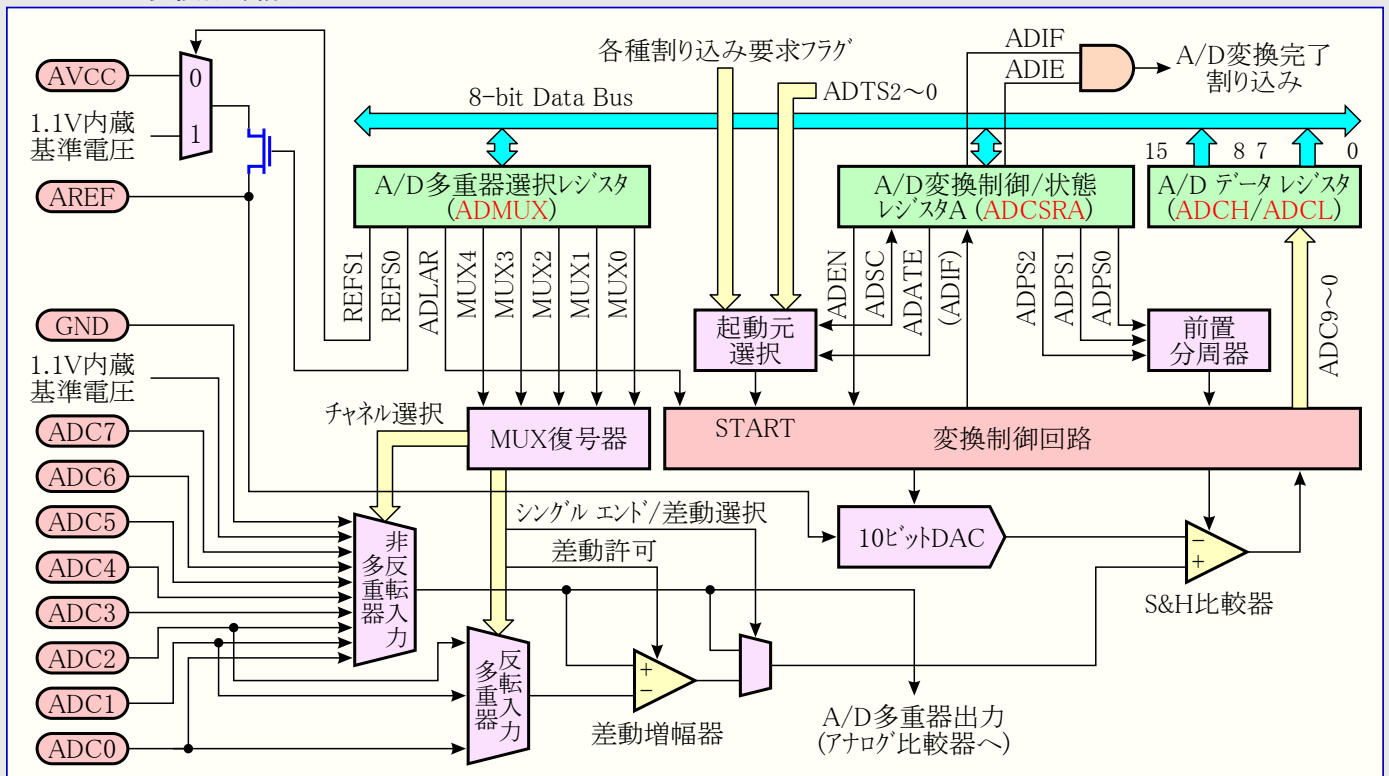
このA/D変換器はA/D変換器への入力電圧が変換中に一定の値で保持されることを保証する採取&保持(S/H)回路を含みます。A/D変換部の構成図は図21-1.で示されます。

A/D変換部には分離されたアナログ電源供給ピン(AVCC)があります。AVCCはVCCから $\pm 0.3$ Vよりも多く違ってはなりません。このピンの接続方法は143頁の「雑音低減技術」項をご覧ください。

公称1.1Vの内蔵基準電圧またはAVCCがチップ上で提供されます。この基準電圧は雑音特性向上のため、コンデンサによってAREFピンで外部的にデカップ(雑音分離)できます。

26頁の「電力削減レジスタ(PRR)」でのPRADCBITはA/D変換部を許可するために0を書かれなければなりません。

図21-1. A/D変換器部構成図



## 21.3. 操作

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AREFピンの電圧-1 LSBを表します。A/D多重器選択レジスタ(ADMUX)の基準電圧選択(REFS1,0)ビットへの書き込みにより、任意でAVCCまたは内部1.1V基準電圧がAREFピンに接続できます。従ってこの内部基準電圧は雑音耐性を改善するためにAREFピンで外部コンデンサによってデカップ(雑音結合減少)ができます。

アナログ入力チャネルはADMUXのチャネル選択(MUX4~0)ビットへの書き込みによって選ばれます。GNDと固定基準電圧(1.1V内蔵基準電圧(VBG))だけでなく、どのADC入力ピン(ADC7~0)もがA/D変換器のシングルエンド入力として選べます。A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットの設定(1)によって(動作が)許可されます。基準電圧と入力チャネルの選択は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変換部は電力を消費しないので、節電をする休止形態へ移行する前にA/D変換部をOFFに切り替えることが推奨されます。

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

この結果が左揃え補正され、8ビットを越える精度が必要とされない場合はADCHを読むことで足ります。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。

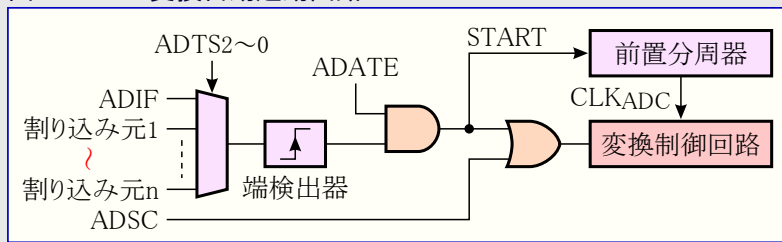
A/D変換部には変換完了時に起動できる自身の割り込みがあります。A/DデータレジスタへのA/D変換器アクセスがADCLとADCHの読み込み間で禁止されている場合、例えその変換結果が失われても割り込みは起動します。

## 21.4. 変換の開始

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、変換が完了されると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ばれると、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(ADCSRB)のA/D変換起動元選択(ADTS2~0)ビットの設定によって選ばれます(起動元の一覧についてはADTSビットの記述をご覧ください)。選んだ起動信号上に上昇端が起きると、A/D変換用前置分周器がリセットし、変換が開始されます。これは一定間隔での変換開始の方法を提供します。変換完了時、起動信号が未だ設定(1)されている場合、新規の変換は開始されません。変換中にこの起動信号上で別の上昇端が起きると、その端(エッジ)は無視されます。指定した割り込みが禁止またはステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)でも、割り込み要求フラグが設定(1)されることに注意してください。従って割り込みを起こ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21-2. A/D変換自動起動回路



起動元としてA/D変換完了割り込み要求フラグ(ADIF)を使うことは、A/D変換器に実行中の変換が完了されると直ぐに新規変換を開始させます。そのため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ないます。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)されるかどうかに関らず、連続的な変換を実行します。

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

変換はA/D変換雑音低減機能の使用によっても開始され得ます。この機能はアイドル休止動作とA/D変換雑音低減休止動作中に変換を許可します。詳細については143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)



## 21.5. 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。

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)される限り走行を保ち、ADENが0の時は継続的にリセットします。

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

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

内部基準電圧がA/D変換器への入力として使われるとき、電圧の安定に一定時間を必要とします。安定(待機を)しない場合は初回変換後の最初の読み込み値は不正になるかもしれません。

実際の採取&保持(保持開始点)は通常変換の開始後1.5変換クロック周期、初回変換の開始後13.5変換クロック周期で行われます。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。

自動起動が使われると、前置分周器は起動要因発生時にリセットされます。これは起動要因から変換開始までの一定の遅延を保証します。この動作での採取&保持は起動要因となる信号の上昇後、2変換クロック周期で採取が行われます。同期化論理回路(端(エッジ)検出器)に対して、追加の3 CPUクロック周期が費やされます。A/D変換完了以外の要因からの自動起動を伴う差動動作を使うとき、各変換は25変換クロックを必要とします。これはA/D変換器が毎変換後、禁止そして再許可されなければならないからです。

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。変換時間の概要については表21-1をご覧ください。

図21-3. A/D変換前置分周器部構成

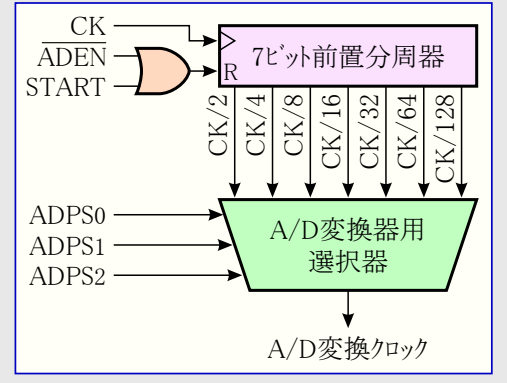


図21-4. 初回変換タイミング (単独変換動作)

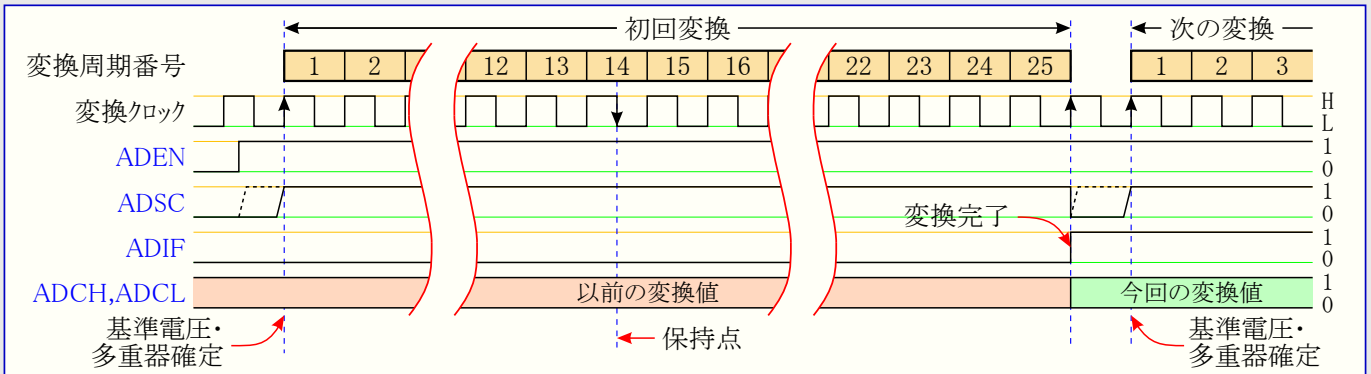


図21-5. 通常変換タイミング (単独変換動作)

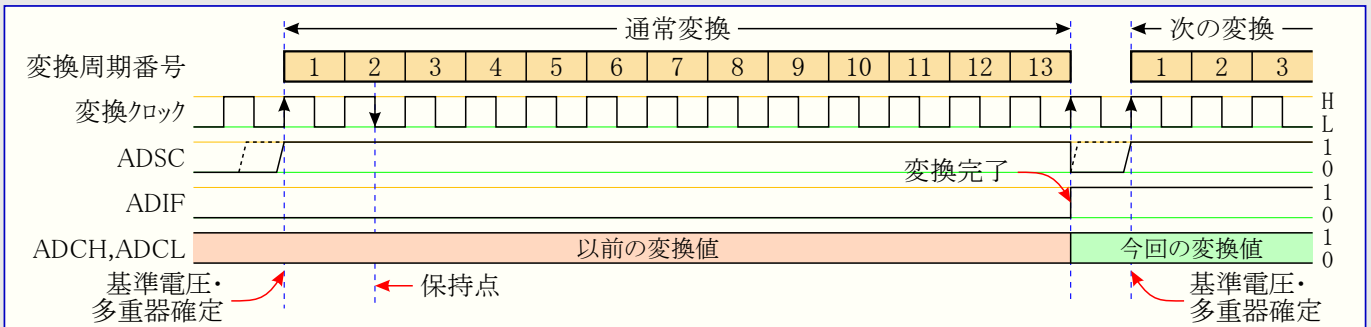




図21-6. 通常変換タイミング\* (自動起動変換動作)

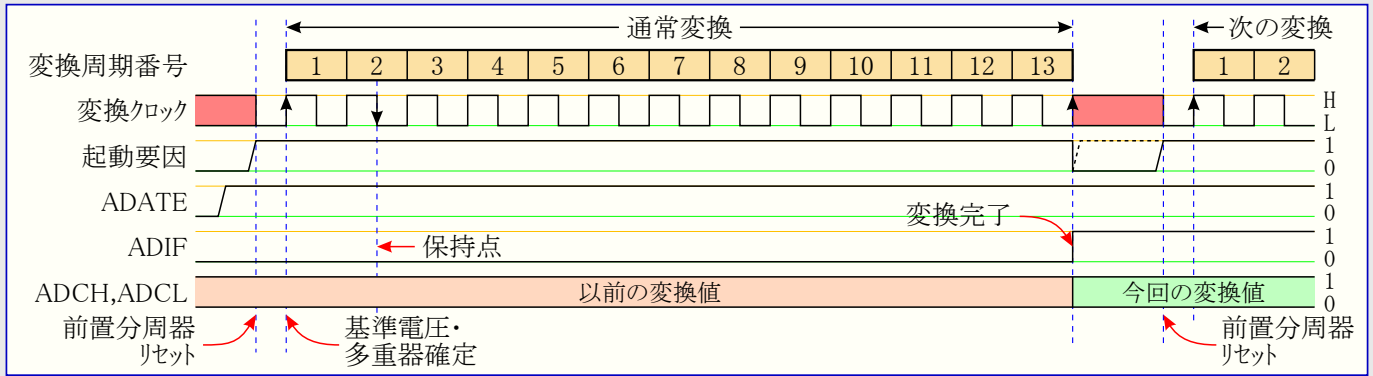


図21-7. 連続変換動作タイミング

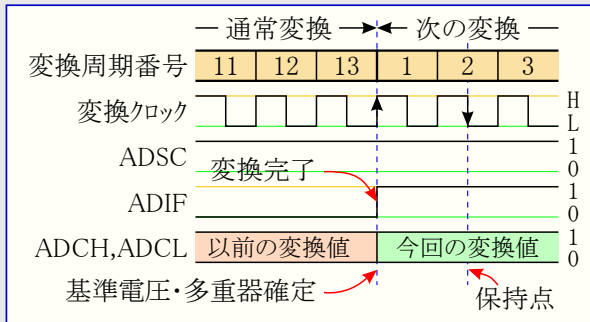


表21-1. A/D変換時間

変換種別	保持点	変換時間
初回変換	13.5	25
シングルエントリ入力通常変換	1.5	13
自動起動変換	1.5 (2)	13.5

注: 変換時間を除く各値は変換開始からの変換クロック数です。

## 21.6. チャネル変更と基準電圧選択

A/D多重器選択レジスタ(ADMUX)のチャネル選択(MUX4~0)ビットと基準電圧選択(REFS1,0)ビットはCPUが乱順にアクセスするための一時レジスタを通して単独緩衝されます。これはチャネルと基準電圧の選択が変換中の安全なところでだけ行うのを保証します。チャネルと基準電圧の選択は変換が開始されるまで継続的に更新されます。一旦変換が始まると、A/D変換器に対して十分な採取/変換時間を保証するためにチャネルと基準電圧の選択は固定されます。継続的な更新は変換完了(ADCSRAのADIF=1)前の最後の変換クロック周期で再開します。ADCSRAの変換開始(ADSC)ビットが書かれた後の次の変換クロックの上昇端で変換が始まることに注意してください。従って使用者はADSC書き込み後、1変換クロック周期(経過)まで新しいチャネルまたは基準電圧選択値をADMUXに書かないことが推奨されます。

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

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

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

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

JTAGインターフェースが許可されると、PORTF7~4のADC入力チャネル機能が無効にされます。48頁の表13-11を参照してください。

(訳注:共通性のため上記行追加)

### 21.6.1. A/D入力チャネル

チャネル選択を変更する時に使用者は正しいチャネルが選ばれることを保証するために次の指針を守るべきです。

単独変換動作では常に変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれませんが。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が。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。

### 21.6.2. A/D変換基準電圧

このA/D変換用の基準電圧(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 エント入力チャネルは\$3FFで打ち切る符号に帰着します。VREFはAVCC、内部1.1V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してA/D変換部に接続されます。内部1.1V基準電圧は内蔵基準(バンドギャップ)電圧(VBG)から内部増幅器を通して生成されます。どちらの場合でも外部AREFピンは直接的にA/D変換部へ接続され、AREFピンとGND間にコンデンサを接続することにより、基準電圧は雑音耐性をより高められます。VREF(電圧)は高入力インピーダンス電圧計とAREFピンで測定することもできます。VREFは高インピーダンス出力で、容量性負荷のみがシステム内で接続されるべきであることに注意してください。

使用者がAREFピンに接続された固定電圧源にするなら、この外部電圧がその他の内部基準電圧と短絡してしまうため、使用者はこの応用内で他の基準電圧選択を使ってはなりません。外部電圧がAREFピンに印加されないなら、使用者は基準電圧選択としてAVCCと内部1.1V基準電圧間の切り替えができます。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。

## 21.7. 雑音低減機能

このA/D変換部はCPUコアと他の周辺I/Oが誘導した雑音を削減するために**休止形態**中の変換を可能にする雑音低減機能が特徴です。この機能は**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れなければ、A/D変換完了割り込みはCPUを起動してA/D変換完了割り込みルーチンを実行します。A/D変換完了前に他の割り込みがCPUを起動すると、その割り込みが実行され、A/D変換完了割り込み要求はA/D変換完了時に生成されます。CPUは新規**SLEEP**命令が実行されるまで活動動作に留まります。

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

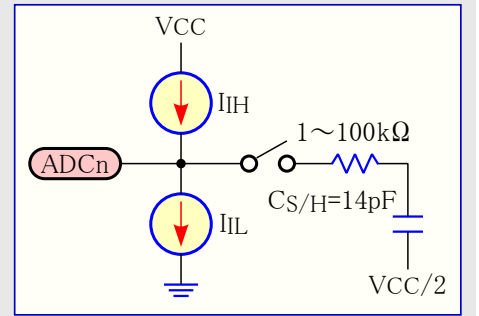
### 21.7.1. アナログ入力回路

シングル エンド入力チャネルのアナログ回路は**図21-8**で図示されます。ADCnに印加したアナログ(信号)源はそのチャネルがADC入力として選ばれているかどうかにかかわらず、ピン容量とそのピンの漏れ電流に左右されます。そのチャネルが選ばれると、(アナログ信号)源は直列抵抗(入力経路の合成抵抗)を通してS/Hコンデンサを駆動しなければなりません。

A/D変換部は概ね10kΩ若しくはそれ以下の出力インピーダンスのアナログ信号用に最適化されています。このような(アナログ信号)源が使われるなら、採取時間は無視してもよいでしょう。より高いインピーダンスの(アナログ信号)源が使われる場合、採取時間は広範囲に変化し得るS/Hコンデンサを充電するために(アナログ信号)源がどれくらいの時間を必要とするかに依存します。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低インピーダンス(アナログ信号)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21-8. アナログ入力回路

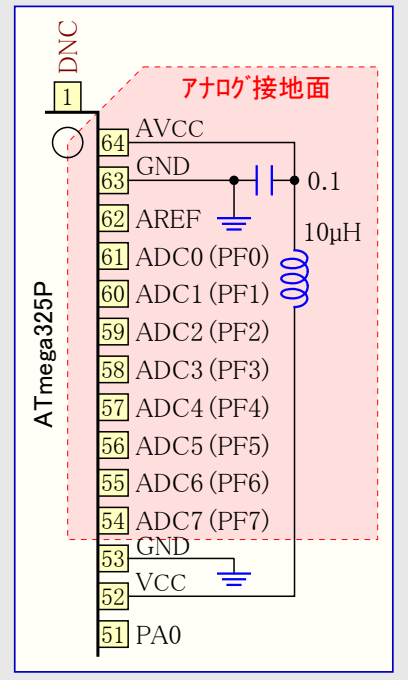


### 21.7.2. アナログ雑音低減技術

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が発生します。精密な変換精度が必要な場合、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。

1. アナログ信号経路を可能な限り最短にしてください。アナログ信号線がアナログGND面上を走ることに注意し、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。
2. デバイスのAVCCピンは**図21-9**で示されるようにLC濾波器を経由してデジタル供給電圧(VCC)に接続されるべきです。
3. CPUからの誘導雑音を低減するために**A/D変換の雑音低減機能**を使ってください。
4. どれかのADCポートピンがデジタル出力として使われる場合、これらが変換進行中に切り替わらないことが重要です。

図21-9. A/D変換部電源接続



### 21.7.3. A/D変換の精度定義

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

#### • 変位(オフセット)誤差 – 図21-10.

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

#### • 利得誤差 – 図21-11.

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

#### • 積分非直線性誤差 (INL) – 図21-12.

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

#### • 微分非直線性誤差 (DNL) – 図21-13.

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

#### • 量子化誤差

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

#### • 絶対精度

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

図21-10. 変位(オフセット)誤差

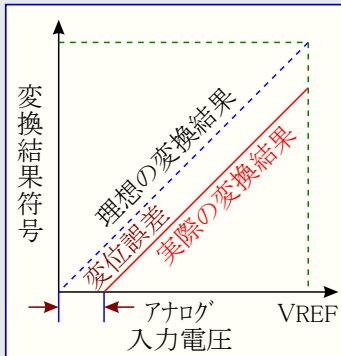


図21-11. 利得誤差

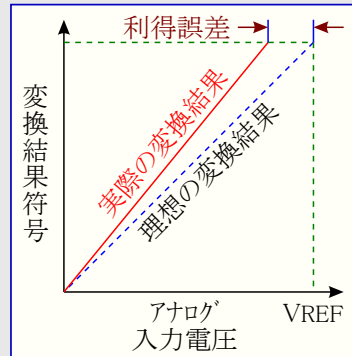


図21-12. 積分非直線性誤差

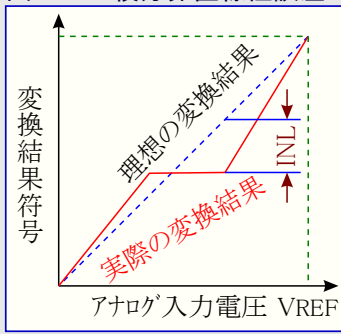
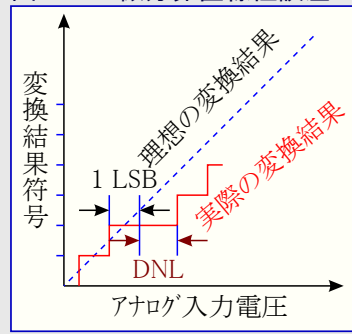


図21-13. 微分非直線性誤差



### 21.8. A/D変換の結果

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

シングルエンド入力変換での結果は右式で示されます。VINは選んだ入力ピンの電圧で、VREFは選んだ基準電圧です(145頁の表21-3と表21-4をご覧ください)。\$000はアナログGNDを表し、\$3FFは選んだ基準電圧-1 LSBを表します。

$$ADC = \frac{VIN \times 1024}{VREF}$$

差動チャネルが使われると、その結果は右式で示されます。VPOSは非反転入力ピンの電圧、VNEGは反転入力ピンの電圧、VREFは選んだ基準電圧です。この結果は-512(\$200)~+511(\$1FF)の2の補数形式で示されます。使用者が結果の符号を素早く検査したいなら、結果のMSB(ADCHのADC9)を読むことで充分なことに注目してください。このビットが1ならばその結果は-、0ならばその結果は+です。図21-14は差動入力範囲の符号化を示します。

$$ADC = \frac{(VPOS - VNEG) \times 512}{VREF}$$

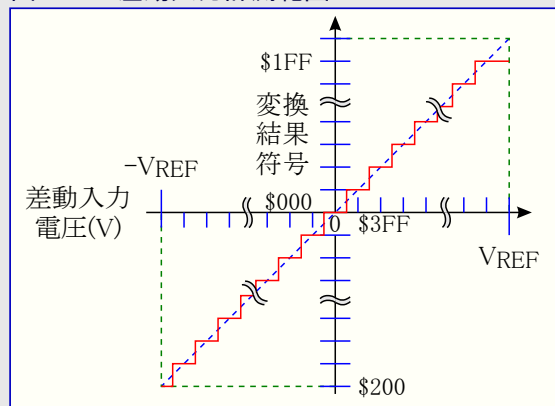
表21-2は差動入力チャネルの組(ADCn-ADCm)が基準電圧(VREF)で選ばれた場合の出力符号の結果を示します。

表21-2. 入力電圧と出力符号の関係

VADCn	読み出し符号	対応する10進値
VADCm+VREF	\$1FF	511
VADCm+(511/512)VREF	\$1FF	511
VADCm+(510/512)VREF	\$1FE	510
}	}	}
VADCm+(1/512)VREF	\$001	1
VADCm	\$000	0
VADCm-(1/512)VREF	\$3FF	-1
}	}	}
VADCm-(511/512)VREF	\$201	-511
VADCm-VREF	\$200	-512

例: ADMUX=\$ED(ADC3とADC2差動入力、VREF=1.1V、左揃え)で、ADC3が300mV、ADC2が500mVの場合、A/D変換値は $512 \times (300 - 500) \div 1100 = -93 = \$3A3$ 、従って、ADCH=\$D8, ADCL=\$C0、ADLAR=0なら、ADCH=\$03, ADCL=\$A3。

図21-14. 差動入力計測範囲



## 21.9. A/D変換用レジスタ

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

ビット (\$7C)	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7,6 – REFS1,0 : 基準電圧選択 (Reference Select Bits 1, 0)

これらのビットは表21-3.で示されるようにA/D変換器の基準電圧を選びます。これらのビットが変換中に変更されると、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。AREFピンに外部基準電圧が印加される場合、内部の基準電圧が使われてはなりません。

表21-3. A/D変換部の基準電圧選択

REFS1	REFS0	基準電圧
0	0	AREFピンの外部基準電圧 (AVCCと内部1.1V基準電圧は切り離されます。)
0	1	AVCC (内部1.1V基準電圧は切り離されますが、AREFにデカップ用コンデンサが接続できます。)
1	0	(予約)
1	1	内部1.1V基準電圧 (AVCCは切り離されますが、AREFにデカップ用コンデンサが接続できます。)

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

ADLARビットはA/Dデータレジスタ内の変換結果の配置に影響を及ぼします。結果を左揃えにするにはADLARに1を書いてください。さもなければ結果は右揃えです。ADLARビットの変更はどんな進行中の変換にも拘らず、直ちにA/Dデータレジスタの内容に影響を及ぼします。このビットの完全な記述については147頁の「A/Dデータレジスタ」をご覧ください。

#### • ビット4~0 – MUX4~0 : A/Dチャネル選択 (Analog Channel Select Bits 4~0)

これらビットの値はA/D変換器にどのアナログ入力の組み合わせが接続されるかを選びます。詳細については表21-4.をご覧ください。これらのビットが変換中に変更される場合、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。

表21-4. アナログ入力チャネル選択

MUX4~0	シングル エント 入力	差動入力		MUX4~0	シングル エント 入力	差動入力	
		非反転入力	反転入力			非反転入力	反転入力
00000	ADC0			10000		ADC0	ADC1
00001	ADC1			10001		ADC1	ADC1
00010	ADC2			10010		ADC2	ADC1
00011	ADC3			10011		ADC3	ADC1
00100	ADC4			10100		ADC4	ADC1
00101	ADC5			10101		ADC5	ADC1
00110	ADC6			10110		ADC6	ADC1
00111	ADC7			10111		ADC7	ADC1
01000	(利用不可)	(利用不可)	(利用不可)	11000		ADC0	ADC2
01001				11001		ADC1	ADC2
01010				11010		ADC2	ADC2
01011				11011		ADC3	ADC2
01100				11100		ADC4	ADC2
01101				11101		ADC5	ADC2
01110				11110	1.1V(V <sub>BG</sub> )		
01111				11111	0V(GND)		



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

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されます。変換が進行中にA/D変換部をOFFにすることはその変換を(途中)終了します。

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

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

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

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

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

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

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

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

このビットが**1**を書かれ、SREGの全割り込み許可(I)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21-5. A/D変換クロック選択 (CK=システム クロック)

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

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

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

#### • ビット7,5~3 – Res : 予約 (Reserved Bits)

これらのビットは将来の使用に対して予約されています。将来のデバイスとの共通性のため、ADCSRが書かれるとき、これらのビットは**0**が書かれなければなりません。

#### • ビット2~0 – ADTS2~0 : A/D変換自動起動要因選択 (ADC Auto Trigger Source)

A/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットが**1**を書かれると、これらのビットの値はどの起動元がA/D変換を起動するかを選びます。ADATEが解除(**0**)されると、ADTS2~0設定は無効です。変換は選んだ割り込みフラグの上昇端によって起動されます。解除(**0**)されている起動元から設定(**1**)されている起動元への切り替えが、起動信号上に上昇端を生成することに注意してください。ADCSRAのA/D許可(ADEN)ビットが設定(**1**)されているなら、これが変換を開始させます。連続変換動作(ADTS2~0=0)への切り替えは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。(訳補:起動元一覧表は次頁参照)

表21-6. A/D変換自動起動元選択

ADTS2	ADTS1	ADTS0	起動元
0	0	0	連続変換動作
0	0	1	アナログ比較器
0	1	0	外部割り込み要求0
0	1	1	タイマ/カウンタ0比較一致
1	0	0	タイマ/カウンタ0溢れ
1	0	1	タイマ/カウンタ1比較B一致
1	1	0	タイマ/カウンタ1溢れ
1	1	1	タイマ/カウンタ1捕獲要求

## 21.9.4. A/Dデータレジスタ (ADC Data Register) ADCH,ADCL

<b>ADLAR=0時</b>								
ビット (\$79)	15	14	13	12	11	10	9	8
	—	—	—	—	—	—	ADC9	ADC8
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
Read/Write	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0
<b>ADLAR=1時</b>								
	15	14	13	12	11	10	9	8
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2
	7	6	5	4	3	2	1	0
	ADC1	ADC0	—	—	—	—	—	—

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。差動チャネルが使われると、結果は2の補数形式で表されます(訳補:本行追加)。

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(差動入力チャネルについては符号+7ビット)を越える精度が必要とされないなら、ADCHを読むことで用が足ります。さもないければADCLが先に、その後ADCHが読まれなければなりません。

A/D多重器選択レジスタ(ADMUX)の左揃え選択(ADLAR)ビットとA/Dチャネル選択(MUX4~0)ビットは本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)ならば結果は左揃えにされます。ADLARが解除(0:既定)ならば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144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。

## 21.9.5. デジタル入力禁止レジスタ0 (Digital Input Disable Register 0) DIDR0

ビット (\$7E)	7	6	5	4	3	2	1	0
	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7~0 – ADC7D~ADC0D : ADC7~0 デジタル入力禁止 (ADC7~0 Digital Input Disable)

このビットが論理1を書かれると、対応するADCnピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、対応するポート入力レジスタのビット(PINx)は常に0として読みます。アナログ信号がADCnピンに印加され、そのピンからのデジタル入力が必要とされない時にデジタル入力緩衝部での消費電力を削減するため、そのビットは論理1を書かれるべきです。

## 22. JTAGインターフェースと内蔵デバッグ機能

### 22.1. 特徴

- JTAGインターフェース (IEEE標準1149.1準拠)
- JTAG規格に従った境界走査(Boundary-Scan)能力
- デバッグは各部のアクセスが可能
  - 全ての内蔵周辺機能
  - 内部及び外部RAM
  - 内蔵レジスタ ファイル
  - プログラム カウンタ
  - EEPROM及びフラッシュ メモリ
- 中断(BREAK)によって支援される広範囲な内蔵デバッグ機能
  - AVRのBREAK命令
  - プログラムの流れ変更での停止
  - 1命令実行(シングル ステップ)停止
  - プログラム メモリ上の単一アドレスまたはアドレス範囲による中断点(ブレークポイント)
  - データ メモリ上の単一アドレスまたはアドレス範囲による中断点
- JTAGインターフェースを介してのフラッシュ メモリ、EEPROM、ヒューズ ビット、施錠ビットの読み書き(プログラミング)
- AVR Studioによる内蔵デバッグ機能の支援

### 22.2. 概要

AVRのIEEE標準1149.1準拠JTAGインターフェースは以下について使えます。

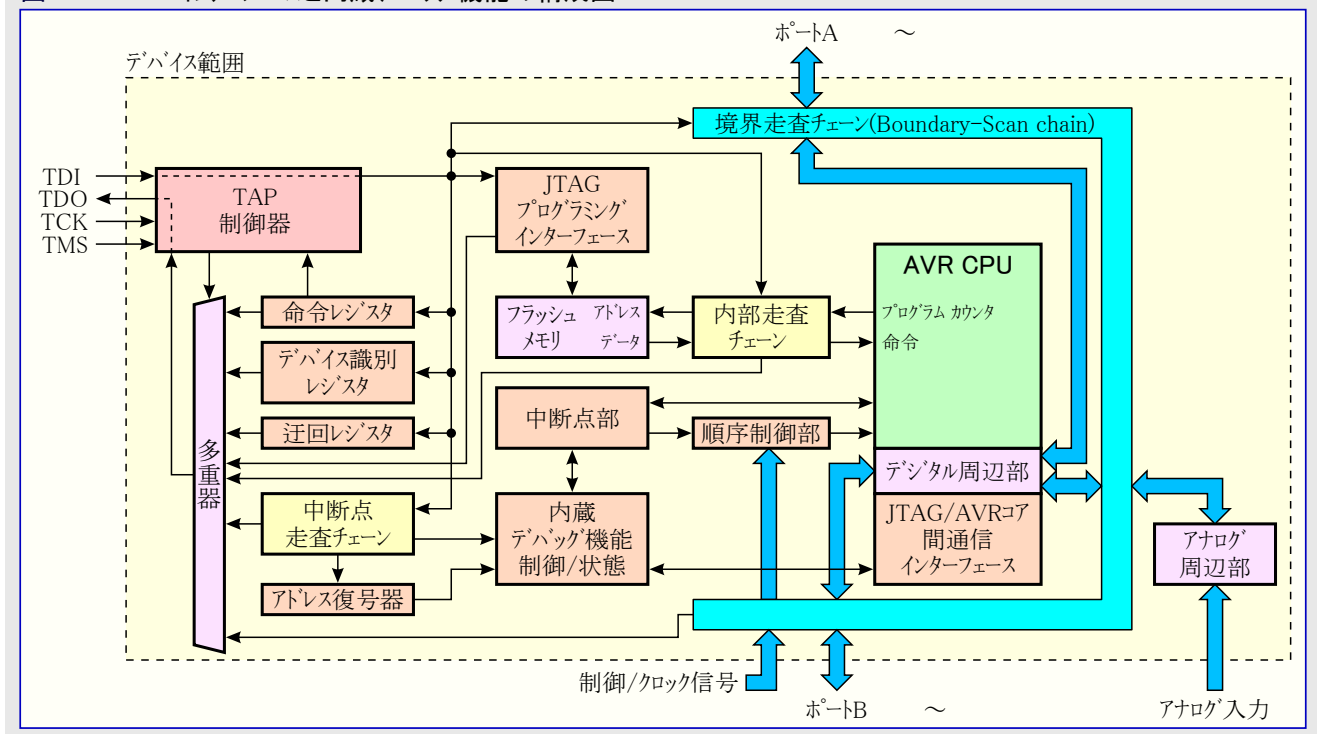
- JTAG境界走査(Boundary-Scan)能力による基板などの試験
- 不揮発性メモリ、ヒューズ ビット、施錠ビットの読み書き(プログラミング)
- 内蔵デバッグ機能による実装デバッグ

簡単な記述は次節で与えられます。JTAGインターフェース経由のプログラミングと境界走査(Boundary-Scan)チェーンに関する詳細な記述は、各々186頁の「JTAGインターフェース経由のプログラミング」と152頁の「IEEE 1149.1(JTAG)境界走査(Boundary-Scan)」で得られます。内蔵デバッグ機能の支援は独自JTAG命令で考慮されており、Atmelと選ばれた業者のみに配布されます。

図22-1はJTAGインターフェースと内蔵デバッグ機能の構成図を示します。TAP(検査入出力ポート)制御器はTMSとTCKによって制御される順次回路です。TAP制御器はTDI入力とTDO出力間の走査チェーン(移動レジスタ接続)として、JTAG命令レジスタまたは各種データレジスタの1つのどれかを選びます。JTAG命令レジスタはデータレジスタの動きを制御するJTAG命令を保持します。

デバイス識別(Identification)レジスタ、迂回(Bypass)レジスタ、境界走査チェーン(Boundary-Scan chain)の各データレジスタは基板段階の試験に使われます。(実際には現実と仮定の各種データレジスタで構成する)JTAGプログラミング インターフェースはJTAGインターフェース経由での直列プログラミングに使われます。内部走査チェーン(Scan chain)と中断点(ブレークポイント)走査チェーンは内部デバッグ機能だけで使われます。

図22-1. JTAGインターフェースと内蔵デバッグ機能の構成図



## 22.3. 検査入出力ポート (TAP:Test Access Port)

JTAGインターフェースはAVRの4つのピンを通してアクセスされます。JTAG用語ではこれらのピンが検査入出力ポート(TAP)を構成します。

- **TMS** : 検査種別選択。このピンはTAP制御器順次回路を通しての指示に使われます。
- **TCK** : 検査クロック。JTAG操作はTCKに同期します。
- **TDI** : 検査データ入力。命令レジスタまたはデータレジスタ(走査チェーン)内に移動されるべき直列入力データです。
- **TDO** : 検査データ出力。命令レジスタまたはデータレジスタからの直列出力データです。

IEEE標準1149.1では任意TAP信号の検査リセット(TRST)も記載されていますが、この信号は提供されません。

JTAGENヒューズが非プログラム(1)の時にこれら4つのTAPピンは標準ポートピンで、TAP制御器はリセットです。プログラム(0)され、MCU制御レジスタ(MCUCR)のJTAGインターフェース禁止(JTD)ビットが解除(0)されると、TAP信号入力は内部的にHighへ引かれ、JTAGは境界走査(Boundary-Scan)とプログラミングに関して許可されます。このデバイスはこのヒューズがプログラム(0)されて出荷されます。

内蔵デバッグ機能ではJTAGインターフェースピンに加え、外部リセット元が検知できるため、RESETピンがデバッグによって監視されます。応用でリセット信号線にオープンコレクタ(トレイン)だけが使われるとすれば、デバッグは全システムをリセットするためにRESETピンをLowにすることもできます。

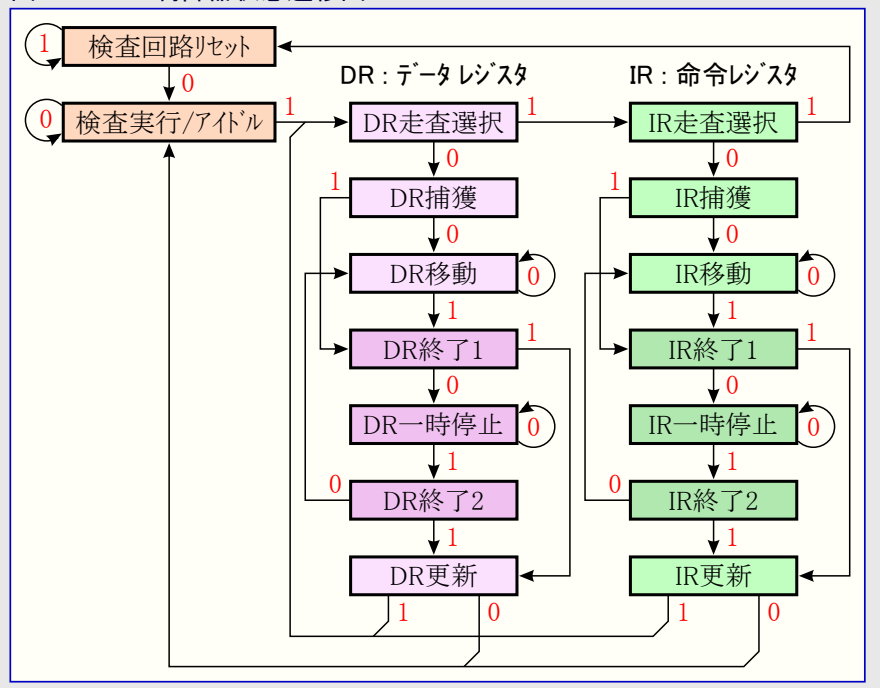
## 22.4. TAP制御器

TAP制御器は境界走査(Boundary-Scan)回路、JTAGプログラミング回路、内蔵デバッグ機能の働きを制御する16段の無限順次回路です。図22-2.に描かれた状態遷移はTCKクロックの上昇時の(各状態遷移付近で示される)TMS上に存在する信号に依存します。電源投入リセット後の初期状態は検査回路リセットです。

この文書内の定義として、全ての移動レジスタに関して入出力ともLSBが最初に移動されます。

検査実行/アイドル状態が現在の状態と仮定した、JTAGインターフェースを使うための典型的な手順を次に示します。

図22-2. TAP制御器状態遷移図



- 命令レジスタ(IR)移動状態へ移行するためにTCKの上昇でTMSへ順次1,1,0,0を与えます。この状態中、TCKの上昇端でTDI入力からJTAG命令レジスタ内に4ビットのJTAG命令を移動します。TMS入力はIR移動状態に留まるために3 LSBの入力中、Lowに保持されなければなりません。命令のMSBはTMS入力のHigh設定によってこの状態を抜ける時に移動入力されます。命令がTDIピンから移動されている間、捕獲されたIRの状態(\$01)がTDOピンへ移動出力されます。JTAG命令はTDIとTDO間に通ず(接続する)特定のデータレジスタを選び、選んだデータレジスタ周辺回路を制御します。
- 検査実行/アイドル状態へ再度移行するためにTMSへ順次1,1,0を与えます。この(JTAG)命令はIR更新状態通過で移動レジスタから並列出力にラッチされます。IR終了1、IR一時停止、IR終了2の各状態は順次回路の誘導操作のためだけに使われます。
- データレジスタ(DR)移動状態へ移行するためにTCKの上昇端でTMSへ順次1,0,0を与えます。この状態中、TCKの上昇端でTDI入力から(JTAG命令レジスタ内の現在のJTAG命令で)選んだデータレジスタに値を移動入力します。DR移動状態に留まるためにMSBを除く全ビットの入力中、TMS入力はLowに保持されなければなりません。データのMSBはTMS入力のHigh設定によってこの状態を抜ける時に移動入力されます。データレジスタがTDIピンから移動入力される間、DR捕獲状態(1,0)でデータレジスタに捕獲した並列入力がTDOピンへ移動出力されます。
- 検査実行/アイドル状態へ再移行するためにTMSへ順次1,1,0を与えます。選んだデータレジスタがラッチした並列出力を持つ場合、DR更新状態でラッチを行います。DR終了1、DR一時停止、DR終了2の各状態は順次回路の誘導操作のためだけに使われます。

状態遷移図で示されるように検査実行/アイドル状態はJTAG命令の選択とデータレジスタの使用間で移行される必要はなく、また、いくつかのJTAG命令は検査実行/アイドル状態で実行されるべき必然的な機能を選ぶかもしれず、それはアイドル状態として不適当になります。

**注:** TAP制御器の初期状態に関係なく、5 TCKクロック周期に対してTMSをHighに保持することにより、常に検査回路リセット状態へ移行することができます。

JTAG仕様の詳細情報については151頁の「参考文献」に記載された文献を参照してください。



## 22.5. 境界走査チェーン(Boundary-Scan Chain)の使用

境界走査能力の完全な説明は152頁の「IEEE 1149.1(JTAG)境界走査(Boundary-Scan)」章で与えられます。

## 22.6. 内蔵デバッグ機能の使用

図22-1.で示されるように内蔵デバッグに関するハードウェア支援は主に次の部分から成ります。

- 内部AVR CPUコアと内部周辺機能部間インターフェースの走査チェーン(Scan chain)
- 中断点(ブレークポイント)部
- CPUとJTAGシステム間の通信インターフェース

デバッグの実行で必要とされる全ての読み込みまたは変更/書き込みは内部AVR CPU走査チェーン(Scan chain)経由のAVR命令によって行われます。CPUはCPUとJTAGシステム間の通信インターフェース部分が配置されるI/Oメモリ位置に結果を送ります。

中断点部はプログラムの流れ変更での一時停止(Break)、1命令実行(Single Step Break)、2つのプログラムメモリ用中断点、2つの複合中断点を実行します。合わせて4つ中断点は次のどれかとして設定ができます。

- 4つのプログラムメモリ単一中断点
- 3つのプログラムメモリ単一中断点 + 1つのデータメモリ単一中断点
- 2つのプログラムメモリ単一中断点 + 2つのデータメモリ単一中断点
- 2つのプログラムメモリ単一中断点 + 1つのプログラムメモリ範囲中断点(中断点と遮蔽)
- 2つのプログラムメモリ単一中断点 + 1つのデータメモリ範囲中断点(中断点と遮蔽)

けれどもAVR Studioのようなデバッグは最終使用者への柔軟性をより少なくする内部目的にこれら資源の1つまたはそれ以上を使うかもしれません。

内蔵デバッグ特殊JTAG命令の一覧は「内蔵デバッグ特殊JTAG命令」で与えられます。

JTAG検査入出力ポート(TAP)を許可するにはJTAGENヒューズがプログラム(0)されなければなりません。加えて内蔵デバッグ機能で作業するには更にOCDENヒューズがプログラム(0)され、施錠ビットが施錠なしに設定されなければなりません。何れかの施錠ビットが設定(0)されると、保護機能で内蔵デバッグ機能が禁止されます。そうでなければ内蔵デバッグ機能が保護されたデバイスへの裏口を提供することになっていでしょう(訳補:保護の意味がなくなるの意)。

AVR Studioは使用者にチップ内蔵デバッグ能力(機能)、AVRインサーキットエミュレータ、または(AVR Studio)組み込みAVR命令一式シミュレータでAVRデバイスのプログラム実行を完全に制御することを可能にします。AVR Studio®はAtmel AVRアセンブラでアセンブルしたアセンブリ言語プログラムと他業者製コンパイラでコンパイルしたC言語プログラムのソースレベル実行を支援します。

AVR StudioはMicrosoft Windows® 98/2000、Windows NT®, Windows XP®, Windows Vista®下で走行します。

AVR Studioの完全な記述についてはAVR Studio使用者の手引きを参照してください。要点だけが本文書で示されます。

全ての必要な実行指令はAVR Studio内のソースレベルと逆アセンブリレベルの両方で利用できます。使用者はプログラムの実行、1命令実行(Single Step)、内側(関数内)実行、外側(関数外)実行、命令行へのカーソル配置と命令行が(カーソル行へ)到達されるまでの実行、一時停止、実行対象のリセットができます。加えて使用者は無制限数のコードの中断点(BREAK命令使用)と、2つまでのデータメモリの中断点か、代わりに遮蔽による範囲指定(複合)中断点を持てます。

## 22.7. 内蔵デバッグ特殊JTAG命令

内蔵デバッグの支援は独自JTAG命令で考慮され、Atmelと選ばれた業者だけに配布されます。参考のため、命令コードの一覧が示されます。

- 独自命令0 : \$8 (内蔵デバッグ機能アクセス用独自JTAG命令)
- 独自命令1 : \$9 (内蔵デバッグ機能アクセス用独自JTAG命令)
- 独自命令2 : \$A (内蔵デバッグ機能アクセス用独自JTAG命令)
- 独自命令3 : \$B (内蔵デバッグ機能アクセス用独自JTAG命令)

## 22.8. JTAGプログラミング能力の使用

JTAG経由AVRデバイスプログラミングは4ピンのJTAGポート、TCK,TMS,TDI,TDO経由で実行されます。これらはJTAGプログラミング実行のために(電源ピンに加えて)制御/監視される必要があるピンだけです。外部的な12V印加は必要とされません。JTAG検査入出力ポート(TAP)を許可するにはJTAGENヒューズがプログラム(0)され、MCU制御レジスタ(MCUCR)のJTAGインターフェース禁止(JTD)ビットが解除(0)されなければなりません。

JTAGプログラミング能力は以下を支援します。

- フラッシュメモリのプログラミングと照合
- EEPROMのプログラミングと照合
- ヒューズビットのプログラミングと照合
- 施錠ビットのプログラミングと照合

施錠ビット保護は丁度並列プログラミング動作でのようです。施錠ビットLB1またはLB2がプログラム(0)されると、先にチップ消去を行わない限り、OCDENヒューズはプログラム(0)できません。これは保護されたデバイスの内容を読み出す裏口がないことを保証する保護機能です。

JTAGインターフェースを通ずプログラミングとプログラミング特殊JTAG命令の記述は186頁の「JTAGインターフェース経由のプログラミング」項で与えられます。



## 22.9. 参考文献

一般的な境界走査(Boundary-Scan)についての詳細は、次の文献を参照してください。

- IEEE : IEEE Std 1149.1-1990. IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE, 1993
- Colin Maunder : The Board Designers Guide to Testable Logic Circuits, Addison Wesley, 1992

## 22.10. 内蔵デバッグ用レジスタ

### 22.10.1. 内蔵デバッグ レジスタ (On-chip Debug Register) OCDR

ビット	7	6	5	4	3	2	1	0	
\$31 (\$51)	IDRD/MSB							LSB	OCDR
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	

OCDRはマイクロ コントローラ内の実行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。CPUはこの位置へ書くことによってデバッグへバイトを転送できます。このレジスタが書かれてしまっているのをデバッグに示すため、同時に内部フラグIDRD(I/Oデバッグ レジスタ更新)が設定(1)されます。CPUがOCDRを読むとき、LSB 7ビットがOCDRからで、一方MSBはIDRDビットです。デバッグはこの情報を読む時にIDRDビットを解除(0)します。

いくつかのAVRデバイスに於いてこのレジスタは標準I/O位置で共用されます。この場合、OCDRは内蔵デバッグ許可(OCDEN)ヒューズがプログラム(0)された場合にだけアクセスでき、デバッグがOCDRへアクセスできます。他の全ての場合、標準I/O位置がアクセスされます。

このレジスタの使用法のより多くの情報はデバッグの資料を参照してください。

## 23. IEEE 1149.1 (JTAG) 境界走査(Boundary-Scan)

### 23.1. 特徴

- JTAGインターフェース (IEEE標準1149.1準拠)
- JTAG規格に従った境界走査(Boundary-Scan)能力
- チップ外接線を持つアナログ回路は勿論、全てのポート機能の完全な走査(Scan)
- 任意IDCODE命令の支援
- AVRリセット用公開AVR\_RESET命令の追加

### 23.2. 概要

境界走査チェーン(Boundary-Scan chain)にはデジタルI/Oピンは勿論、チップ外接線を持つアナログ回路のアナログとデジタル間境界の論理値を駆動、監視する能力があります。システム上ではJTAG能力を持つ全てのICがTDIとTDOの接続により長い移動レジスタの形態で接続されます。外部制御器がこれらの出力ピンで値を駆動するようにデバイスを設定し、他のデバイスから受け取った入力値を監視します。この制御器は予測される結果と受信した値を比較します。この方法で境界走査(Boundary-Scan)は4つのTAP信号だけを使うことにより、基板上の部品全般と内部結線の検査機構を提供します。

IEEE 1149.1で定義された4つの必須JTAG命令、IDCODE, BYPASS, SAMPLE/PRELOAD, EXTESTだけでなく、AVR特殊公開JTAG命令のAVR\_RESETも基板検査に使えます。既定JTAG命令がIDCODEなので、データレジスタ経路の初期走査はデバイスの識別符号(ID-code)を示します。検査動作中にリセット(状態)のAVRデバイスがあることは好都合かもしれませんが、リセットでない場合、デバイスの入力走査動作によって決定されるかもし、検査動作を終了する時に内部ソフトウェアが未確定状態になってしまうかもしれません。リセットへの移行で、どのポートピンの出力も直ちにHi-Z状態に移行し、HIGHZ命令が不要になります。必要とされるなら、デバイスを通する走査チェーンを可能な限り最短とするのにBYPASS命令が発行できます。外部RESETピンをLowに引き込むか、またはリセットデータレジスタの適切な設定によるAVR\_RESET命令の発行でデバイスをリセット状態に設定できます。

EXTEST命令は外部ピンの採取と出力ピンの値設定に使われます。EXTEST命令がJTAG命令レジスタ(IR)に設定されると、直ちに出力ラッチからの値がピンで出力駆動されます。このため、最初にEXTEST命令を発行する時に基板の損傷を避けるため、走査チェーン内に初期値を設定するのにSAMPLE/PRELOAD命令も使われるべきです。SAMPLE/PRELOAD命令は通常動作中デバイスの外部ピン的高速監視にも使えます。

JTAG検査入出力ポート(TAP)を許可するにはJTAGENヒューズがプログラム(0)され、MCU制御レジスタ(MCUCR)のJTAGインターフェース禁止(JTD)ビットが解除(0)されなければなりません。

境界走査(Boundary-Scan)にJTAGインターフェースを使う場合、内部チップ周波数よりも高いJTAG TCKクロック周波数の使用が可能です。チップのクロックは走行に必要とされません。

### 23.3. データレジスタ

境界走査(Boundary-Scan)に関連するデータレジスタ(DR)を次に示します。

- 迂回(Bypass)レジスタ • デバイス識別(Device Identification)レジスタ • リセット(Reset)レジスタ • 境界走査チェーン(Boundary-Scan chain)

#### 23.3.1. 迂回 (Bypass) レジスタ

迂回レジスタは1段の移動レジスタから成ります。TDIとTDO間の経路として迂回レジスタが選ばれると、このレジスタはTAP制御器がデータレジスタ(DR)捕獲状態を抜ける時に0へリセットされます。迂回レジスタは他のデバイスが検査される時にシステム上の走査チェーン(Scan chain)を短くするのに使えます。

#### 23.3.2. デバイス識別 (Device Identification) レジスタ

図23-1. はデバイス識別レジスタの構造を示します。

図23-1. デバイス識別レジスタの形式

ビット	31	28	27	12	11	1	0
デバイス識別	版番号				部品番号		製造者識別
ビット数	4				16		1

**版番号** 版番号は部品(デバイス)の改訂を示す4ビットです。JTAG版番号はデバイスの改訂版に従います。改訂Aは\$00, 改訂Bは\$01, 以下同様です。

**部品番号** 部品番号は部品(名)を示す16ビットです。ATmega325P/3250PのJTAG部品番号は174頁の表25-6.で示されます。

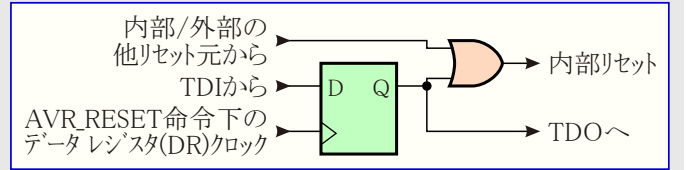
**製造者識別** 製造者識別は製造業者を示す11ビットです。AtmelのJTAG製造者識別は174頁の表25-6.で示されます。

## 23.3.3. リセット (Reset) レジスタ

リセットレジスタはデバイスのリセットに使われる検査データレジスタです。リセットによってAVRのポートピンがHi-Z状態となるので、未実装の任意JTAG命令HIGHZ機能の代わりにできます。

リセットレジスタ内の値1は外部RESETピンをLowに引っ張るのと同じです。デバイスはリセットレジスタ内の値が1である限り、リセットです。リセットレジスタの開放後、デバイスはクロック選択用ヒューズビットの設定に従ってリセット起動遅延時間(18頁の「クロック元」参照)、リセットに留まります。図23-2で示されるようにこのデータレジスタからの出力はラッチされず、直ちにリセットが行われます。

図23-2. リセットレジスタ



## 23.3.4. 境界走査チェーン(Boundary-Scan Chain)

境界走査チェーンにはデジタルI/Oピンは勿論、チップ外の接続を持つアナログ回路のアナログとデジタル間境界の論理値上の設定と監視能力があります。

完全な記述については154頁の「境界走査チェーン(Boundary-Scan chain)」をご覧ください。

## 23.4. 境界走査(Boundary-Scan)用JTAG命令

命令レジスタ(IR)は16種までの命令を支援する4ビット幅です。以下に示した一覧は境界走査(Boundary-Scan)操作に有用なJTAG命令です。全ポートピンをHi-Z状態にするための任意HIGHZ命令は実装されませんが、AVR\_RESET命令を使うことによって全出力がHi-Z状態に設定できることに注目してください。

このデータシート内の定義として、全ての移動レジスタに関して入出力ともLSBが最初に移動されます。

各命令の命令コードは命令名下の16進形式で示されます。本文は各命令についてTDIとTDO間の経路として選ばれるデータレジスタを記述します。(訳注:配置構成変更により、内容を一部変更)

### 23.4.1. EXTEST - \$0

AVRデバイスに外部回路的検査を行うためにデータレジスタ(DR)として境界走査チェーン(Boundary-Scan chain)を選ぶための必須JTAG命令です。ポートピンは方向、プルアップ、出力値、入力値は走査チェーン(Scan chain)内で全てアクセス可能です。チップ外接線を持つアナログ回路についてはアナログとデジタル論理間のインターフェースが走査チェーン内になります。境界走査チェーンのラッチした出力値はJTAG命令レジスタ(IR)にEXTEST命令が設定されると直ぐに駆動されます。本命令が有効な状態を次に示します。

- DR捕獲：外部ピン上の値が境界走査チェーン内に採取されます。
- DR移動：内部走査チェーンがTCKクロック入力によって移されます。
- DR更新：走査チェーンからの値が出力ピンに印加(出力)されます。

### 23.4.2. IDCODE - \$1

データレジスタ(DR)として32ビットのデバイス識別(ID)レジスタを選ぶ任意JTAG命令です。デバイス識別レジスタは版番号、デバイス番号と、JEDECによって決められた製造者符号から成ります。これは電源投入後の既定命令です。本命令が有効な状態を次に示します。

- DR捕獲：デバイス識別レジスタ内の値が境界走査チェーン内に採取されます。
- DR移動：内部走査チェーン(デバイス識別レジスタ値)がTCKクロック入力によって移されます。

### 23.4.3. SAMPLE\_PRELOAD - \$2

システム動作に影響を与えずに入出力ピンの状態採取、出力ラッチの事前設定を行うための必須JTAG命令です。けれども出力ラッチはピンに接続されません。データレジスタ(DR)として境界走査チェーン(Boundary-Scan chain)が選ばれます。本命令が有効な状態を次に示します。

- DR捕獲：外部ピン上の値が境界走査チェーン内に採取されます。
- DR移動：境界走査チェーンがTCKクロック入力によって移されます。
- DR更新：境界走査チェーンからの値が出力ラッチに印加(設定)されます。けれども出力ラッチはピンに接続されません。

### 23.4.4. AVR\_RESET - \$C

AVRデバイスを強制的にリセット状態、またはJTAGリセット元から開放するためのAVR特殊公開JTAG命令です。この命令で検査入出力ポート(TAP)制御器はリセットされません。データレジスタ(DR)として1ビットのリセットレジスタが選ばれます。リセットチェーン(レジスタ)が論理1である限り、リセットが活性(有効)なことに注意してください。このチェーンからの出力はラッチされません。本命令が有効な状態を次に示します。

- DR移動：走査チェーン(リセットレジスタ)がTCKクロック入力によって移されます。

### 23.4.5. BYPASS - \$F

データレジスタ(DR)に迂回(Bypass)レジスタを選ぶ必須JTAG命令です。本命令が有効な状態を次に示します。

- DR捕獲：迂回レジスタに論理0を設定します。
- DR移動：TDIとTDO間の迂回レジスタ(セル)が移されます。

## 23.5. 境界走査チェーン(Boundary-Scan Chain)

境界走査チェーンにはデジタルI/Oピンは勿論、チップ外の接続を持つアナログ回路のアナログとデジタル間境界の論理値の設定と監視の能力があります。

### 23.5.1. デジタル ポートピンの走査

図23-3.はプルアップ機能付き双方向ポートピン用境界走査(Boundary-Scan)セルを示します。このセルはプルアップ許可(PUE<sub>xn</sub>)用標準境界走査(Boundary-Scan)セルと3つの信号、出力(方向)制御(OC<sub>xn</sub>)、出力データ(OD<sub>xn</sub>)、入力データ(ID<sub>xn</sub>)の組み合わせの双方向ピン用セルの2段の移動レジスタだけから成ります。ポートとピンの添え字は続く記述で使われません。

境界走査(Boundary-Scan)論理回路はデータシート内の図に含まれません。図23-4.は39頁の「入出力ポート」章で記載される単純なデジタルポートピンを示します。図23-3.からの境界走査セルの詳細が図23-4.上の破線内に置き換わります。

交換ポート機能が存在しない場合、入力データ(ID)はPIN<sub>xn</sub>レジスタ値(しかしIDは同期化回路を持たない)、出力データ(OD)はPORT<sub>xn</sub>レジスタ、出力制御(OC)はDD<sub>xn</sub>方向レジスタ、プルアップ許可(PUE)は論理合成値(PUD AND DD<sub>xn</sub> AND PORT<sub>xn</sub>)に各々対応します。

デジタル交換ポート(ピン)機能は走査チェーン(Scan chain)が実際のピン値を読むために図23-4.内の破線外側に接続されます。アナログ機能について、それらは外部ピンからアナログ回路へ直結で、走査チェーンはアナログ回路とデジタル論理回路間のインターフェースに挿入されます。

JTAG IRがEXTTESTまたはSAMPLE\_PRELOADを内包すると、クロックは例えばCKOUTヒューズがプログラム(0)されていてもポートピンに送出されません。例えばクロックが出力でも、JTAG IRがSAMPLE\_PRELOADを内包するとき、クロックは境界走査によって採取されません。

図23-3. プルアップ機能付き双方向ポート用境界走査(Boundary-Scan)セル構成図

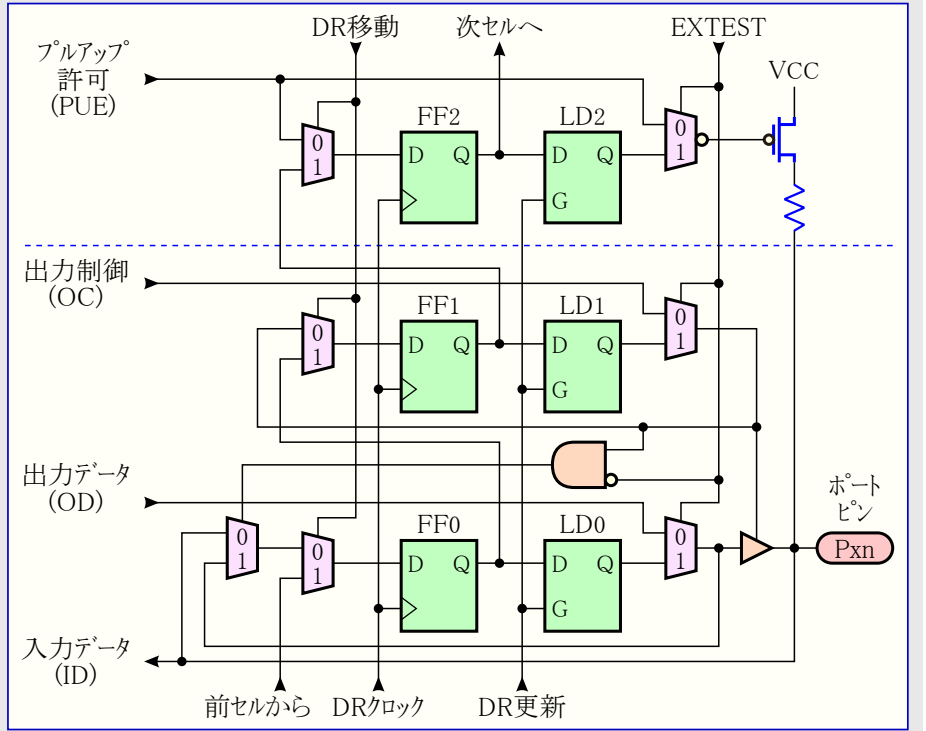
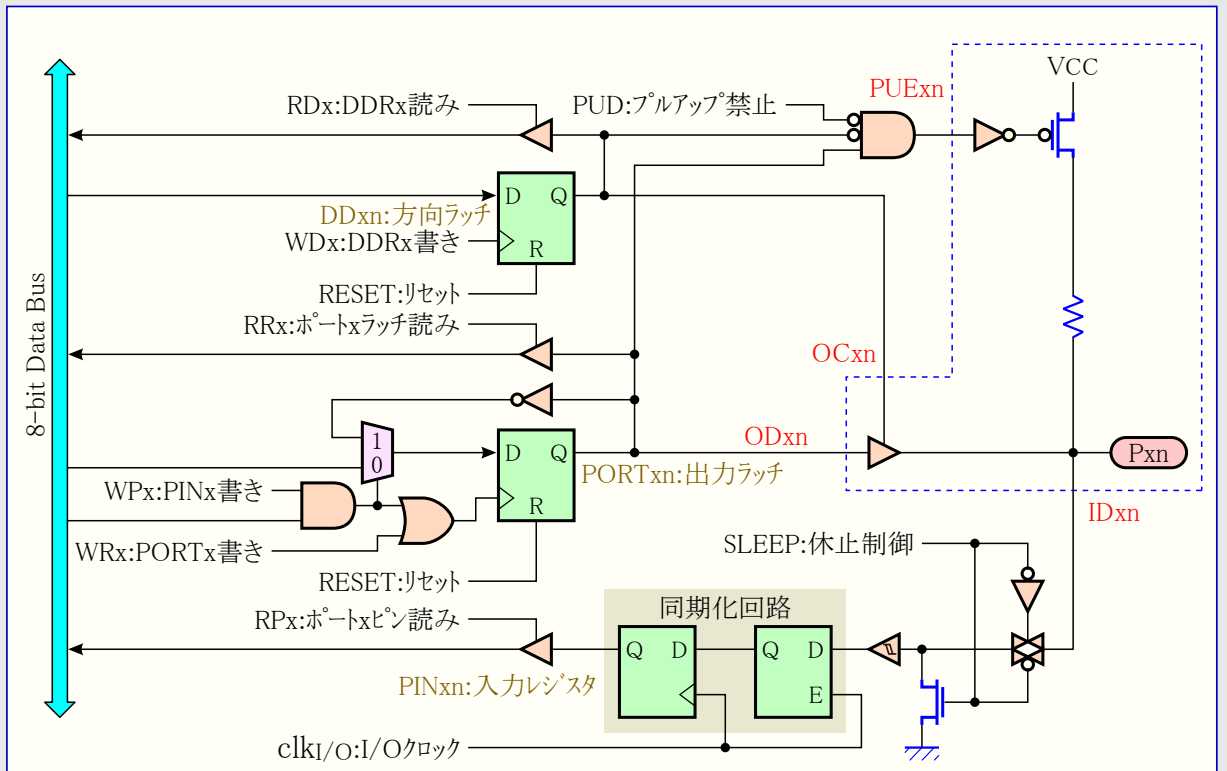


図23-4. 標準ポートピン構成図

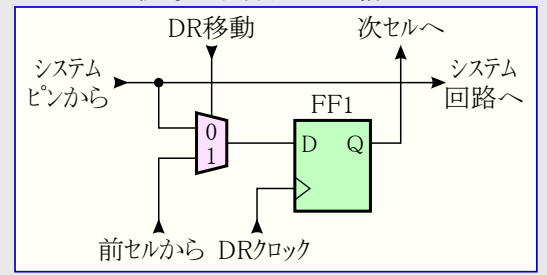


注: 詳細については境界走査(Boundary-Scan)をご覧ください。

## 23.5.2. RESETピンの走査

RESETピンは標準リセット動作について5V負論理(Low有効)、高電圧並列プログラミングについて12V正論理(High有効)を受け入れます。図23-5.で示される監視専用セルが5Vリセット信号(RSTT)と12Vリセット信号(RSTHV)の両方に挿入されます。

図23-5. 監視専用境界走査セル構成図



## 23.5.3. クロックピンの走査

AVRデバイスにはヒューズによって選択可能な多くのクロック種別があります。これらは校正付き内蔵RC発振器、外部RC発振器、外部クロック信号、(高周波数)水晶発振子、低周波数水晶発振子、セラミック振動子です。

図23-6.は走査チェーン内で支援される各発振器と外部接続を示します。許可信号は標準境界走査セルで支援され、一方発振器/クロック出力には監視専用セルが付随されます。主クロックに加え、タイマ用発振器も同じ方法で走査されます。校正付き内蔵RC発振器からの出力はこの発振器が外部接続を持たないので走査されません。

図23-6. クロック機能と発振器用境界走査セル構成図

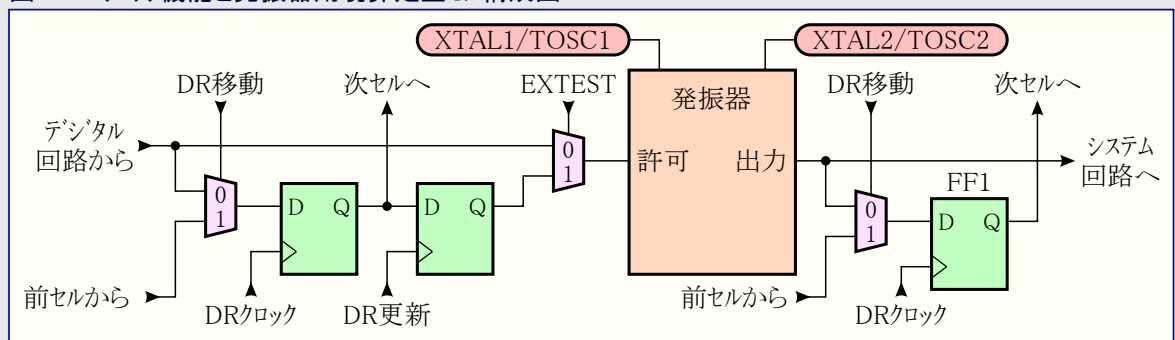


表23-1.はタイマ用32kHz発振器は勿論、外部クロックピンXTAL1、XTAL1/XTAL2接続の発振器に対する走査レジスタの一覧です。

表23-1. 発振器用走査信号

許可信号	走査クロック信号線	クロック種別	未使用時値
EXTCLKEN	EXTCLK(XTAL1)	外部クロック信号	0
OSCON	OSCCK	外部水晶発振子 外部セラミック振動子	1
OSC32EN	OSC32CK	低周波数外部水晶	1

- 注: 1. 主クロックとして同時に1つよりも多くのクロック元を許可してはいけません。
2. 発振器出力の走査は内部発振器とJTAGのTCKクロック間の周波数偏差のため、予期せぬ結果を与えます。可能なら、外部クロックでの走査が望まれます。
3. クロック設定はヒューズによってプログラミングされます。ヒューズは実行動作時に変更されないため、クロック設定は与えられた応用について考慮して決定されます。使用者は最終システムで使われるのと同じクロック選択で走査することを推奨されます。システム論理回路が休止形態でクロック選択(元)を禁止できるため、許可信号は走査チェーン内で支援され、それ(その禁止)によって(クロックが)供給されない場合、発振器ピンを走査経路から切り離します。



### 23.5.4. アナログ比較器の走査

境界走査に関する比較器関連の信号は図23-7.で示されます。図23-8.の境界走査セルがこれら信号の各々に付随します。この信号は表23-2.で記述されます。

この比較器は全てのアナログ入力デジタルポートピンとしても兼用されるため、純粋な接続試験用に使われる必要がありません。

図23-7. アナログ比較器部構成図

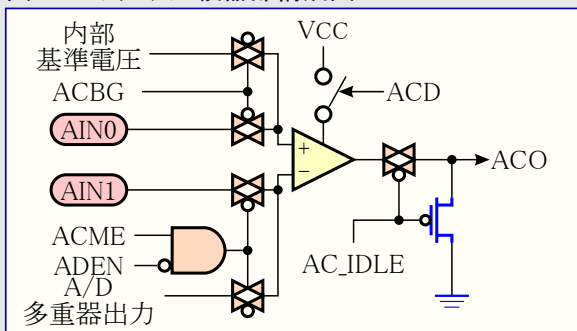


図23-8. 比較器、A/D変換用標準境界走査セル構成図

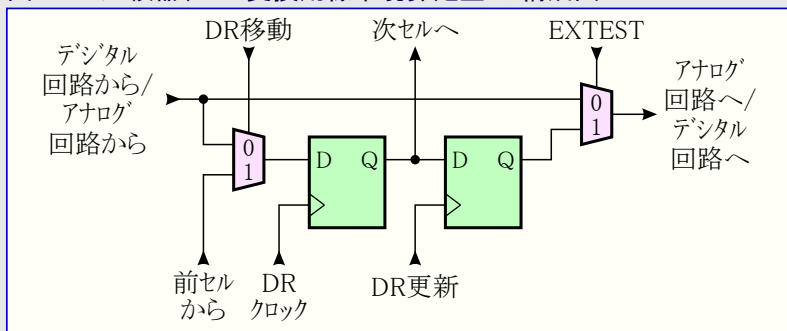


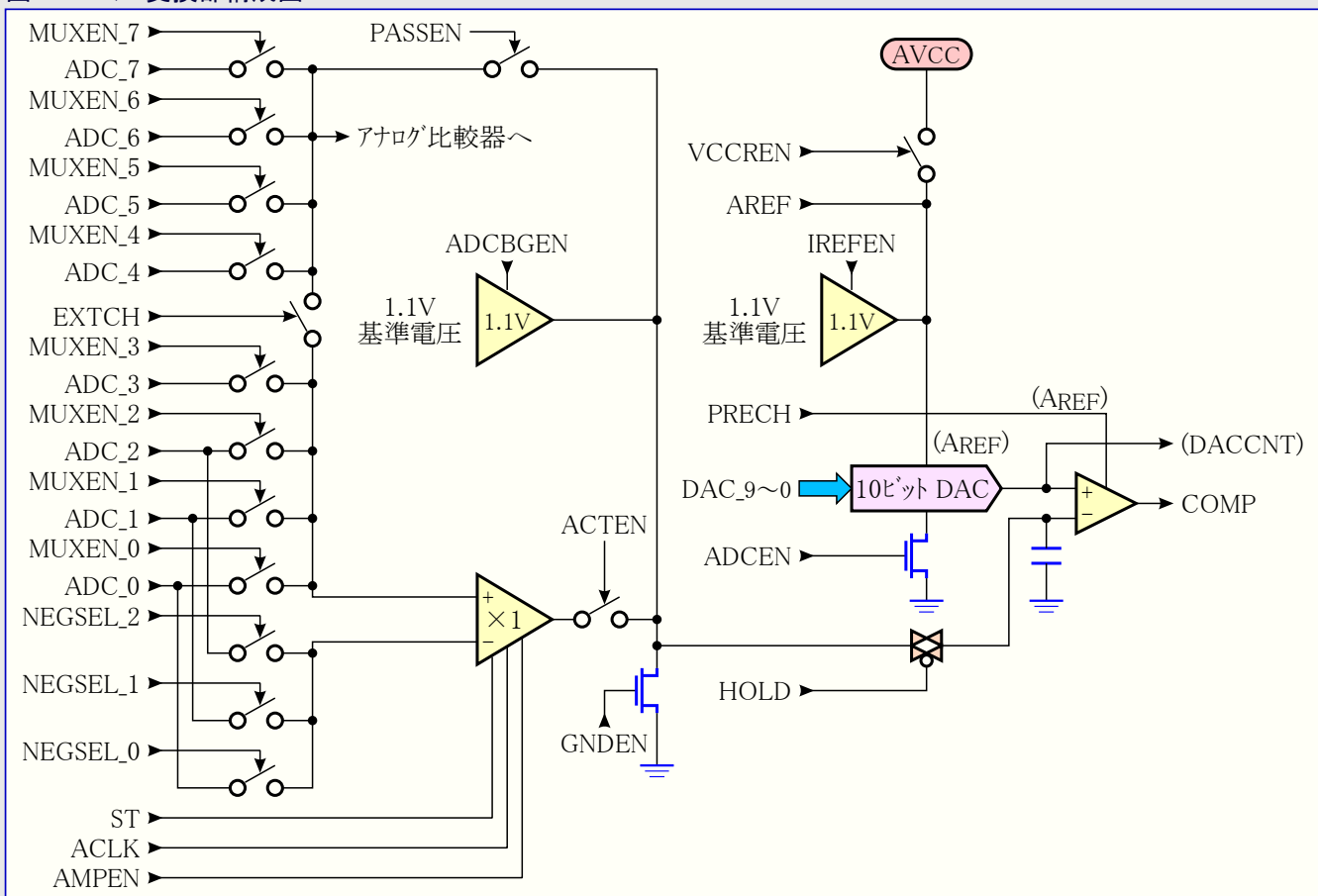
表23-2. アナログ比較器用境界走査信号

信号名	比較器側での方向	意味	未使用時推奨入力値	推奨入力時出力値
AC_IDLE	入力	真の時にアナログ比較器OFF	1	CPU実行に依存
ACO	出力	アナログ比較器出力	CPU実行時入力	0
ACME	入力	真の時にA/D多重器出力使用	0	CPU実行に依存
ACBG	入力	内蔵基準電圧許可	0	CPU実行に依存

### 23.5.5. A/D変換器の走査

図23-9.はA/D変換部と関連する制御と監視信号の構成図を示します。図23-8.の境界走査セルがこれら信号の各々に付随されます。このA/D変換器は全アナログ入力デジタルポートピンとしても兼用されるため、純粋な接続試験に使われる必要がありません。

図23-9. A/D変換部構成図



各信号の概要は表23-3.で示されます。

表23-3. A/D変換用境界走査信号

信号名	A/D変換器側での方向	意味	未使用時推奨入力値	推奨入力または未使用時出力値
COMP	出力	比較器出力	0	0
ACLK	入力	差動段のスイッチ キャパシタのクロック	0	0
ACTEN	入力	差動段⇒比較器接続許可	0	0
ADCBGEN	入力	比較器反転入力へ内部1.1V基準電圧接続	0	0
ADCEN	入力	A/D変換器電源ON信号	0	0
AMPEN	入力	差動段電源ON信号	0	0
DAC_9	入力	D/A変換器デジタル入力ビット9	1	1
DAC_8	入力	D/A変換器デジタル入力ビット8	0	0
DAC_7	入力	D/A変換器デジタル入力ビット7	0	0
DAC_6	入力	D/A変換器デジタル入力ビット6	0	0
DAC_5	入力	D/A変換器デジタル入力ビット5	0	0
DAC_4	入力	D/A変換器デジタル入力ビット4	0	0
DAC_3	入力	D/A変換器デジタル入力ビット3	0	0
DAC_2	入力	D/A変換器デジタル入力ビット2	0	0
DAC_1	入力	D/A変換器デジタル入力ビット1	0	0
DAC_0	入力	D/A変換器デジタル入力ビット0	0	0
EXTCH	入力	ADC3～0の差動段迂回接続	1	1
GNDEN	入力	比較器反転入力GND接続	0	0
HOLD	入力	S&H信号。0=採取、1=保持。差動段使用の場合ACLK=1の時に切り替えなければなりません。	1	1
IREFEN	入力	D/A変換器へのAREFとして内部1.1V基準電圧許可	0	0
MUXEN_7	入力	ADC7 アナログ入力チャネル接続	0	0
MUXEN_6	入力	ADC6 アナログ入力チャネル接続	0	0
MUXEN_5	入力	ADC5 アナログ入力チャネル接続	0	0
MUXEN_4	入力	ADC4 アナログ入力チャネル接続	0	0
MUXEN_3	入力	ADC3 アナログ入力チャネル接続	0	0
MUXEN_2	入力	ADC2 アナログ入力チャネル接続	0	0
MUXEN_1	入力	ADC1 アナログ入力チャネル接続	0	0
MUXEN_0	入力	ADC0 アナログ入力チャネル接続	1	1
NEGSEL_2	入力	ADC2⇒反転差動入力接続	0	0
NEGSEL_1	入力	ADC1⇒反転差動入力接続	0	0
NEGSEL_0	入力	ADC0⇒反転差動入力接続	0	0
PASSEN	入力	差動段迂回路許可	1	1
PRECH	入力	比較器出力ラッチのプリチャージ (負論理)	1	1
ST	入力	AMPEN=1後の最初の2ACLK周期間1ならば、差動段出力がより早く安定	0	0
VCCREN	入力	D/A変換器へのAREFとしてAVCC許可	0	0

注: 図23-9.内の不正なスイッチ設定は信号衝突の原因になり、デバイスを損傷するかもしれません。それらは多くの入力を図23-9.で出力比較器の反転入力のS&Hへ選びます。1つのADCピン、内部基準電圧、GNDのどれか1つの経路だけが選ばれることを確認してください。

走査中にA/D変換器が使われないなら、表23-3の推奨入力値が使われるべきです。使用者は走査中に差動段を使わないことが推奨されます。スイッチ キャパシタを基にした差動段は走査チェーン内で使われる時に達成が難しい高速な操作とタイミング精度が必要です。従って差動段の操作に関する詳細は提供されません。

AVRのA/D変換部はデジタル論理回路内に実装した逐次比較方式と図23-9で示したアナログ回路が基です。境界走査使用時、問題は印加したアナログ電圧が或る範囲内で計測されるのを常に保証することです。これは逐次比較動作を行わずして容易に実行でき、DAC\_9~0デジタル(信号)線に下限値を加え、比較器からの出力がLow(0)であることを確認し、その後DAC\_9~0デジタル(信号)線に上限値を加え、比較器からの出力がHigh(1)であることを検証します。

このA/D変換器は全てのアナログ入力デジタルポートピンとしても兼用されるため、純粋な接続試験に使われる必要がありません。

A/D変換を使う時に次のことに留意してください。

- ADCチャネルで使うポートピンは信号の衝突を避けるため、プルアップを禁止した入力に設定されなければなりません。
- 通常動作ではA/D変換部許可時に(10回の比較で構成される)擬似変換が実行されます。使用者はA/D変換許可後、A/D変換信号の制御/監視をする前に最低200ns待機するか、または最初の変換結果を使う前に擬似変換の実行を推奨されます。
- DAC値はHOLD信号がLow(0:採取)の時に中間値(\$200)で安定していなければなりません。

例として、電源電圧5VでAREFが外部的にVCCへ接続される時のADCチャネル3で1.5V±5%の入力信号を検証する手順を考察してください。

$$\begin{aligned} \text{上限: } & 1024 \times 1.5V \times 1.05 \div 5V = 323 = \$143 \\ \text{下限: } & 1024 \times 1.5V \times 0.95 \div 5V = 291 = \$123 \end{aligned}$$

表23-4の手順内で他の値が与えられる以外は表23-3の推奨値が使われます。(表23-4.には)走査チェーンのDAC値とポートピン値だけが示されます。「動作」欄は続く行内の値で境界走査チェーン レジスタを満たす前にどのJTAG命令を使うべきかを記述します。表の同一列上のデータを走査入力する時に走査出力のデータで検証が行われるべきです。

表23-4. A/D変換使用手順

手順番号	動作	ADGEN	DAC_9~0	MUXEN_7~0	HOLD	PRECH	PF3データ	PF3制御	PF3プルアップ許可
1	SAMPLE_PRELOAD	1	\$200	\$08	1	1	0	0	0
2	EXTEST	1	\$200	\$08	0	1	0	0	0
3		1	\$200	\$08	1	1	0	0	0
4		1	\$123	\$08	1	1	0	0	0
5		1	\$123	\$08	1	0	0	0	0
6	0となるべきCOMP走査出力の検証	1	\$200	\$08	1	1	0	0	0
7		1	\$200	\$08	0	1	0	0	0
8		1	\$200	\$08	1	1	0	0	0
9		1	\$143	\$08	1	1	0	0	0
10		1	\$143	\$08	1	0	0	0	0
11	1となるべきCOMP走査出力の検証	1	\$200	\$08	1	1	0	0	0

この手法の使用ではHOLD信号の束縛タイミングがTCKクロック周波数を押さえます。手順の第5段階でのHOLDのHigh(1)に保持で、TCKクロック周波数は最大保持時間(t<sub>hold\_max</sub>)で割られた走査ビット数の最低5倍でなければなりません。

## 23.6. ATmega325P/3250Pの境界走査(Boundary-Scan)順

表23-5と表23-6は境界走査チェーンがデータ経路として選ばれる時のTDIとTDO間の走査順を示します。ビット0が最下位ビットで走査の最初に入出力されます。走査順は可能な限りピン配置順に従います。このためにポートAのビットは他のポートと違うビット順で走査されます。この規則の例外はアナログ回路が接続される物理ピンに関係なく走査チェーンの最上位を構成するアナログ回路用走査チェーンです。図23-3上で、Pxn出力はFF0、Pxn制御はFF1、Pxnプルアップ許可はFF2に各々対応します。ポートFのビット4,5,6,7はJTAG許可時にTAPピンを構成するので走査チェーンには存在しません。

表23-5. ATmega325P(64ピン)境界走査(Boundary-Scan)順

ビット	信号名	基本単位	ビット	信号名	基本単位	ビット	信号名	基本単位
197	AC_IDLE	アナログ比較器	144	PE4.Control	ポートE	91	PD0.Control	ポートD
196	ACO		143	PE4.Pullup_Enable		90	PD0.Pullup_Enable	
195	ACME		142	PE5.Data		89	PD1.Data	
194	ACBG	(注)	141	PE5.Control		88	PD1.Control	
193	COMP		140	PE5.Pullup_Enable		87	PD1.Pullup_Enable	
192	ACLK		139	PE6.Data		86	PD2.Data	
191	ACTEN		138	PE6.Control		85	PD2.Control	
190	PRIVATE_SIGNAL1		137	PE6.Pullup_Enable		84	PD2.Pullup_Enable	
189	ADCBGEN		136	PE7.Data		83	PD3.Data	
188	ADCEN		135	PE7.Control		82	PD3.Control	
187	AMPEN		134	PE7.Pullup_Enable		81	PD3.Pullup_Enable	
186	DAC_9		133	PB0.Data		80	PD4.Data	
185	DAC_8		132	PB0.Control		79	PD4.Control	
184	DAC_7		131	PB0.Pullup_Enable		78	PD4.Pullup_Enable	
183	DAC_6		130	PB1.Data		77	PD5.Data	
182	DAC_5		129	PB1.Control		76	PD5.Control	
181	DAC_4		128	PB1.Pullup_Enable		75	PD5.Pullup_Enable	
180	DAC_3		127	PB2.Data		74	PD6.Data	
179	DAC_2		126	PB2.Control		73	PD6.Control	
178	DAC_1		125	PB2.Pullup_Enable		72	PD6.Pullup_Enable	
177	DAC_0		124	PB3.Data		71	PD7.Data	
176	EXTCH		123	PB3.Control		70	PD7.Control	
175	GNDEN		122	PB3.Pullup_Enable		69	PD7.Pullup_Enable	
174	HOLD		121	PB4.Data	ポートG	68	PG0.Data	ポートG
173	IREFEN		120	PB4.Control		67	PG0.Control	
172	MUXEN_7		119	PB4.Pullup_Enable		66	PG0.Pullup_Enable	
171	MUXEN_6		118	PB5.Data		65	PG1.Data	
170	MUXEN_5		117	PB5.Control		64	PG1.Control	ポートC
169	MUXEN_4		116	PB5.Pullup_Enable		63	PG1.Pullup_Enable	
168	MUXEN_3		115	PB6.Data		62	PC0.Data	
167	MUXEN_2		114	PB6.Control		61	PC0.Control	
166	MUXEN_1		113	PB6.Pullup_Enable		60	PC0.Pullup_Enable	
165	MUXEN_0		112	PB7.Data		59	PC1.Data	
164	NEGSEL_2		111	PB7.Control		58	PC1.Control	
163	NEGSEL_1		110	PB7.Pullup_Enable		57	PC1.Pullup_Enable	
162	NEGSEL_0		109	PG3.Data	ポートG	56	PC2.Data	
161	PASSEN		108	PG3.Control		55	PC2.Control	
160	PRECH		107	PG3.Pullup_Enable		54	PC2.Pullup_Enable	
159	ST		106	PG4.Data		53	PC3.Data	
158	VCCREN	ポートE	105	PG4.Control		52	PC3.Control	
157	PE0.Data		104	PG4.Pullup_Enable		51	PC3.Pullup_Enable	
156	PE0.Control		103	PG5.Data	(監視専用セル)	50	PC4.Data	
155	PE0.Pullup_Enable		102	RSTT	リセット回路	49	PC4.Control	
154	PE1.Data		101	RSTHV	(監視専用セル)	48	PC4.Pullup_Enable	
153	PE1.Control		100	EXTCLKEN	主クロック/発振器用許可信号	47	PC5.Data	
152	PE1.Pullup_Enable		99	OSCON		46	PC5.Control	
151	PE2.Data		98	RCOSCEN		45	PC5.Pullup_Enable	
150	PE2.Control		97	OSC32EN	主クロック入力/発振器(監視専用セル)	44	PC6.Data	
149	PE2.Pullup_Enable		96	EXTCLK(XTAL1)		43	PC6.Control	
148	PE3.Data		95	OSCK		42	PC6.Pullup_Enable	
147	PE3.Control		94	RCCK	ポートD	41	PC7.Data	
146	PE3.Pullup_Enable		93	OSC32CK		40	PC7.Control	
145	PE4.Data		92	PD0.Data		39	PC7.Pullup_Enable	

注: PRIVATE\_SIGNAL1は常に0として走査されるべきです。

次頁へ続く

表23-5 (続き). ATmega325P(64ピン)境界走査(Boundary-Scan)順

ビット	信号名	基本単位	ビット	信号名	基本単位	ビット	信号名	基本単位
38	PG2.Data	ポートG	25	PA4.Control	ポートA	12	PA0.Pullup_Enable	ポートA
37	PG2.Control		24	PA4.Pullup_Enable		11	PF3.Data	ポートF
36	PG2.Pullup_Enable		23	PA3.Data		10	PF3.Control	
35	PA7.Data	ポートA	22	PA3.Control		9	PF3.Pullup_Enable	
34	PA7.Control		21	PA3.Pullup_Enable		8	PF2.Data	
33	PA7.Pullup_Enable		20	PA2.Data		7	PF2.Control	
32	PA6.Data		19	PA2.Control		6	PF2.Pullup_Enable	
31	PA6.Control		18	PA2.Pullup_Enable		5	PF1.Data	
30	PA6.Pullup_Enable		17	PA1.Data		4	PF1.Control	
29	PA5.Data		16	PA1.Control		3	PF1.Pullup_Enable	
28	PA5.Control		15	PA1.Pullup_Enable		2	PF0.Data	
27	PA5.Pullup_Enable		14	PA0.Data		1	PF0.Control	
26	PA4.Data		13	PA0.Control		0	PF0.Pullup_Enable	

表23-6. ATmega3250P(100ピン)境界走査(Boundary-Scan)順

ビット	信号名	基本単位	ビット	信号名	基本単位	ビット	信号名	基本単位
242	AC_IDLE	アナログ 比較器	199	PE1.Data	ポートE	156	PB5.Control	ポートB
241	ACO		198	PE1.Control		155	PB5.Pullup_Enable	
240	ACME		197	PE1.Pullup_Enable		154	PB6.Data	
239	ACBG	(注)  A/D変換	196	PE2.Data		153	PB6.Control	
238	COMP		195	PE2.Control		152	PB6.Pullup_Enable	ポートG
237	ACLK		194	PE2.Pullup_Enable		151	PB7.Data	
236	ACTEN		193	PE3.Data		150	PB7.Control	
235	PRIVATE_SIGNAL1		192	PE3.Control		149	PB7.Pullup_Enable	
234	ADCBGEN		191	PE3.Pullup_Enable		148	PG3.Data	ポートG
233	ADCEN		190	PE4.Data		147	PG3.Control	
232	AMPEN		189	PE4.Control		146	PG3.Pullup_Enable	
231	DAC_9		188	PE4.Pullup_Enable		145	PG4.Data	
230	DAC_8		187	PE5.Data		144	PG4.Control	
229	DAC_7		186	PE5.Control		143	PG4.Pullup_Enable	
228	DAC_6		185	PE5.Pullup_Enable		142	PG5.Data	(監視専用セル)
227	DAC_5		184	PE6.Data	ポートJ	141	RSTT	リセット回路 (監視専用セル)
226	DAC_4		183	PE6.Control		140	RSTHV	(監視専用セル)
225	DAC_3		182	PE6.Pullup_Enable		139	EXTCLKEN	主クロック/発振 器用許可信号
224	DAC_2		181	PE7.Data		138	OSCON	
223	DAC_1		180	PE7.Control		137	RCOSCEN	
222	DAC_0		179	PE7.Pullup_Enable		136	OSC32EN	主クロック用 入力/発振器 (監視専用セル)
221	EXTCH		178	PJ0.Data		135	EXTCLK(XTAL1)	
220	GNDEN		177	PJ0.Control		134	OSCCK	
219	HOLD		176	PJ0.Pullup_Enable		133	RCCK	
218	IREFEN		175	PJ1.Data	ポートB	132	OSC32CK	ポートJ
217	MUXEN_7		174	PJ1.Control		131	PJ2.Data	
216	MUXEN_6		173	PJ1.Pullup_Enable		130	PJ2.Control	
215	MUXEN_5		172	PB0.Data		129	PJ2.Pullup_Enable	
214	MUXEN_4		171	PB0.Control		128	PJ3.Data	
213	MUXEN_3		170	PB0.Pullup_Enable		127	PJ3.Control	
212	MUXEN_2		169	PB1.Data		126	PJ3.Pullup_Enable	
211	MUXEN_1		168	PB1.Control		125	PJ4.Data	
210	MUXEN_0		167	PB1.Pullup_Enable		124	PJ4.Control	
209	NEGSEL_2		166	PB2.Data		123	PJ4.Pullup_Enable	
208	NEGSEL_1		165	PB2.Control		122	PJ5.Data	
207	NEGSEL_0		164	PB2.Pullup_Enable		121	PJ5.Control	
206	PASSEN	ポートE	163	PB3.Data		120	PJ5.Pullup_Enable	
205	PRECH		162	PB3.Control		119	PJ6.Data	ポートD
204	ST		161	PB3.Pullup_Enable		118	PJ6.Control	
203	VCCREN		160	PB4.Data		117	PJ6.Pullup_Enable	
202	PE0.Data		159	PB4.Control		116	PD0.Data	
201	PE0.Control		158	PB4.Pullup_Enable		115	PD0.Control	
200	PE0.Pullup_Enable		157	PB5.Data		114	PD0.Pullup_Enable	

注: PRIVATE\_SIGNAL1は常に0として走査されるべきです。

次頁へ続く



表23-6 (続き). ATmega3250P(100ピン)境界走査(Boundary-Scan)順

ビット	信号名	基本単位	ビット	信号名	基本単位	ビット	信号名	基本単位
113	PD1.Data	ポートD	75	PC3.Pullup_Enable	ポートC	37	PA4.Control	ポートA
112	PD1.Control		74	PC4.Data		36	PA4.Pullup_Enable	
111	PD1.Pullup_Enable		73	PC4.Control		35	PA3.Data	
110	PD2.Data		72	PC4.Pullup_Enable		34	PA3.Control	
109	PD2.Control		71	PC5.Data		33	PA3.Pullup_Enable	
108	PD2.Pullup_Enable		70	PC5.Control		32	PA2.Data	
107	PD3.Data		69	PC5.Pullup_Enable	ポートH	31	PA2.Control	
106	PD3.Control		68	PH0.Data		30	PA2.Pullup_Enable	
105	PD3.Pullup_Enable		67	PH0.Control		29	PA1.Data	
104	PD4.Data		66	PH0.Pullup_Enable		28	PA1.Control	
103	PD4.Control		65	PH1.Data		27	PA1.Pullup_Enable	
102	PD4.Pullup_Enable		64	PH1.Control		26	PA0.Data	
101	PD5.Data		63	PH1.Pullup_Enable		25	PA0.Control	ポートH
100	PD5.Control		62	PH2.Data		24	PA0.Pullup_Enable	
99	PD5.Pullup_Enable		61	PH2.Control		23	PH4.Data	
98	PD6.Data		60	PH2.Pullup_Enable		22	PH4.Control	
97	PD6.Control		59	PH3.Data		21	PH4.Pullup_Enable	
96	PD6.Pullup_Enable		58	PH3.Control		20	PH5.Data	
95	PD7.Data	ポートG	57	PH3.Pullup_Enable		19	PH5.Control	
94	PD7.Control		56	PC6.Data	ポートC	18	PH5.Pullup_Enable	
93	PD7.Pullup_Enable		55	PC6.Control		17	PH6.Data	
92	PG0.Data		54	PC6.Pullup_Enable		16	PH6.Control	
91	PG0.Control		53	PC7.Data	ポートG	15	PH6.Pullup_Enable	
90	PG0.Pullup_Enable		52	PC7.Control		14	PH7.Data	ポートF
89	PG1.Data	ポートC	51	PC7.Pullup_Enable	ポートA	13	PH7.Control	
88	PG1.Control		50	PG2.Data		12	PH7.Pullup_Enable	
87	PG1.Pullup_Enable		49	PG2.Control	ポートA	11	PF3.Data	
86	PC0.Data		48	PG2.Pullup_Enable		10	PF3.Control	
85	PC0.Control		47	PA7.Data		9	PF3.Pullup_Enable	
84	PC0.Pullup_Enable		46	PA7.Control		8	PF2.Data	
83	PC1.Data		45	PA7.Pullup_Enable		7	PF2.Control	
82	PC1.Control		44	PA6.Data		6	PF2.Pullup_Enable	
81	PC1.Pullup_Enable		43	PA6.Control		5	PF1.Data	
80	PC2.Data		42	PA6.Pullup_Enable		4	PF1.Control	
79	PC2.Control		41	PA5.Data		3	PF1.Pullup_Enable	
78	PC2.Pullup_Enable		40	PA5.Control		2	PF0.Data	
77	PC3.Data		39	PA5.Pullup_Enable		1	PF0.Control	
76	PC3.Control		38	PA4.Data		0	PF0.Pullup_Enable	

## 23.7. 境界走査記述言語(Boundary-Scan Description Language)ファイル

境界走査記述言語(BSDL)ファイルは境界走査が可能なデバイスを自動検査生成ソフトウェアによって使われる標準形式で記述します。境界走査データレジスタ内のビットの順番と機能がこの記述内に含まれます。ATmega325P/3250P用のBSDLファイルは入手可能です。

## 23.8. JTAG用レジスタ

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

MCU制御レジスタは一般MCU機能制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	BODS	BODSE	PUD	—	—	IVSEL	IVCE	MCUCR
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 - JTD : JTAGインターフェース禁止 (JTAG interface disable)

このビットが0の時にJTAGENヒューズがプログラム(0)されていれば、JTAGインターフェースが許可されます。このビットが1の場合、JTAGインターフェースは禁止されます。JTAGインターフェースの予期せぬ許可や禁止を避けるためにこのビットを変更する時は時間制限手順に従わなければなりません。この値を変更するために応用ソフトウェアは4周期以内に希望する値をこのビットへ2回書き込まなければなりません。内蔵デバッグシステム使用時、このビットが切り替えられてはならないことに注意してください。

JTAGインターフェースが他のJTAG回路に接続されないままにされる場合、このJTDビットは1に設定されるべきです。この理由はJTAGインターフェースのTDOピンの静止時電流を避けるためです。

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

MCU状態レジスタはMCUリセットを起こしたリセット元の情報を提供します。

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

#### • ビット4 - JTRF : JTAG リセット フラグ (JTAG Reset Flag)

このビットはJTAG命令AVR\_RESETによって選んだJTAG リセット レジスタ内の論理1でリセットが起こされると設定(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。

## 24. ブート ロード 支援 – 書き込み中読み出し可能な自己プログラミング

### 24.1. 特徴

- 書き込み中読める(Read-While-Write)自己プログラミング
- 柔軟性のあるブート ロード メモリ容量
- 高い安全性 (柔軟な保護用の独立したブート施錠ビット)
- リセット ベクタ選択用の独立したヒューズ
- 最適化されたページ容量 (注1)
- 効率的なコード手法
- 効率的な読み-変更-書き(リード モデファイライト)支援

**注1:** ページはプログラミング中に使われる多数のバイトから成るフラッシュ メモリの区画です(175頁の表25-11.参照)。このページ構成は通常動作に影響を及ぼしません。

### 24.2. 概要

ブート ロード 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 ロード プログラムを使うMCUによって制御される柔軟な応用ソフトウェア更新を可能にします。ブート ロード 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 ロード 領域内のプログラム コードはブート ロード 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 ロード 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 ロード メモリの容量はヒューズで設定可能で、ブート ロード 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ぶ独特な柔軟性を使用者に与えます。

### 24.3. フラッシュ メモリの応用領域とブート ロード 領域

フラッシュ メモリは応用領域とブート ロード 領域の2つの主な領域で構成されます(図24-2.参照)。各領域の容量は170頁の表24-6.と図24-2.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。

#### 24.3.1. 応用領域

応用領域は応用コードを格納するのに使われるフラッシュ メモリの領域です。応用領域用保護レベルは応用ブート施錠ビット(ブート施錠ビット0)によって選べます(165頁の表24-2.参照)。応用領域から実行される時にSPM命令が禁止されるので、応用領域はどんなブート ロード コードも決して格納し得ません。

#### 24.3.2. ブート ロード 領域 (BLS)

応用領域が応用コード格納用に使われるのに対して、SPM命令は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 ロード 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。ブート ロード 領域用保護レベルはブート ロード 施錠ビット(ブート施錠ビット1)によって選べます(165頁の表24-3.参照)。

### 24.4. 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

どちらのアドレスがプログラミングされるかによって、CPUが書き込み中の読み出しを支援するか、ブート ロード ソフトウェアが更新中にCPUが停止されるかのどちらです。上で記述されるようなBOOTSZヒューズによって設定可能な2つの領域に加え、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領域間の境界は164頁の図24-2.と170頁の表24-7.で与えられます。この2つの領域間の主な違いを次に示します。

- RWW領域側に配置されたページを消去または書くとき、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とき、その全ての動作中にCPUは停止されます。

ブート ロード ソフトウェア動作中、使用者ソフトウェアがRWW領域側に配置されるどのコードも決して読めないことに注意してください。「書き込み中読み出し可能領域」という記述はプログラミング(消去または書き込み)される領域としての引用で、ブート ロード ソフトウェアが更新中に実際に読まれる領域ではありません。

**訳補:** 上の記述はNRWW領域からRWW領域をプログラミングするという前提で、消去または書き込みを行う側ではなく、行われる側でこの名称が定義されていることを意味します。即ち、NRWW領域からRWW領域をプログラミングすると、NRWW領域のプログラムは通常通り動作する(即ち読める)ので、プログラミングされる側はRWW領域と名付けられ、この逆ではCPUが停止する(即ち読めない)ので、NRWW領域と名付けられているという意味です。

#### 24.4.1. RWW – 書き込み中読み出し可能領域

ブートローダソフトウェア更新がRWW領域側のページをプログラミングする場合、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領域に配置されるコードだけです。プログラミング実行中、そのソフトウェアはRWW領域が決して読まれないことを保証しなければなりません。使用者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領域側に配置されるコードを読むと、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ブートローダ領域へ移動のどちらかにされるべきです。ブートローダ領域は常にNRWW領域に配置されます。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)ビットは論理1として読みます。プログラミングが完了した後、RWW領域に配置し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。RWWSBを解除(0)する方法の詳細については171頁の「SPM命令制御/状態レジスタ(SPMCSR)」をご覧ください。

#### 24.4.2. NRWW – 書き込み中読み出し不能領域

NRWW領域に配置したコードはブートローダソフトウェアがRWW領域内のページを更新する時に読めます。ブートローダコードがNRWW領域を更新するとき、全てのページ消去またはページ書き込み動作中にCPUが停止されます。

表24-1. 書き込み中読み出し可能機能

プログラミング中にZポイントで指定される領域	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

図24-1. RWW領域とNRWW領域の関係

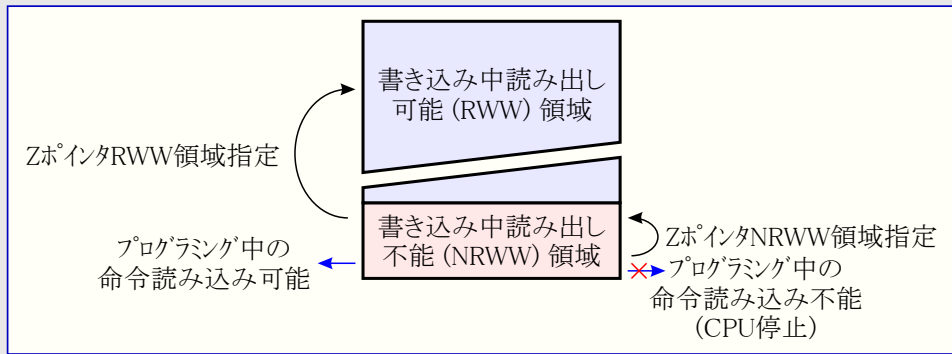
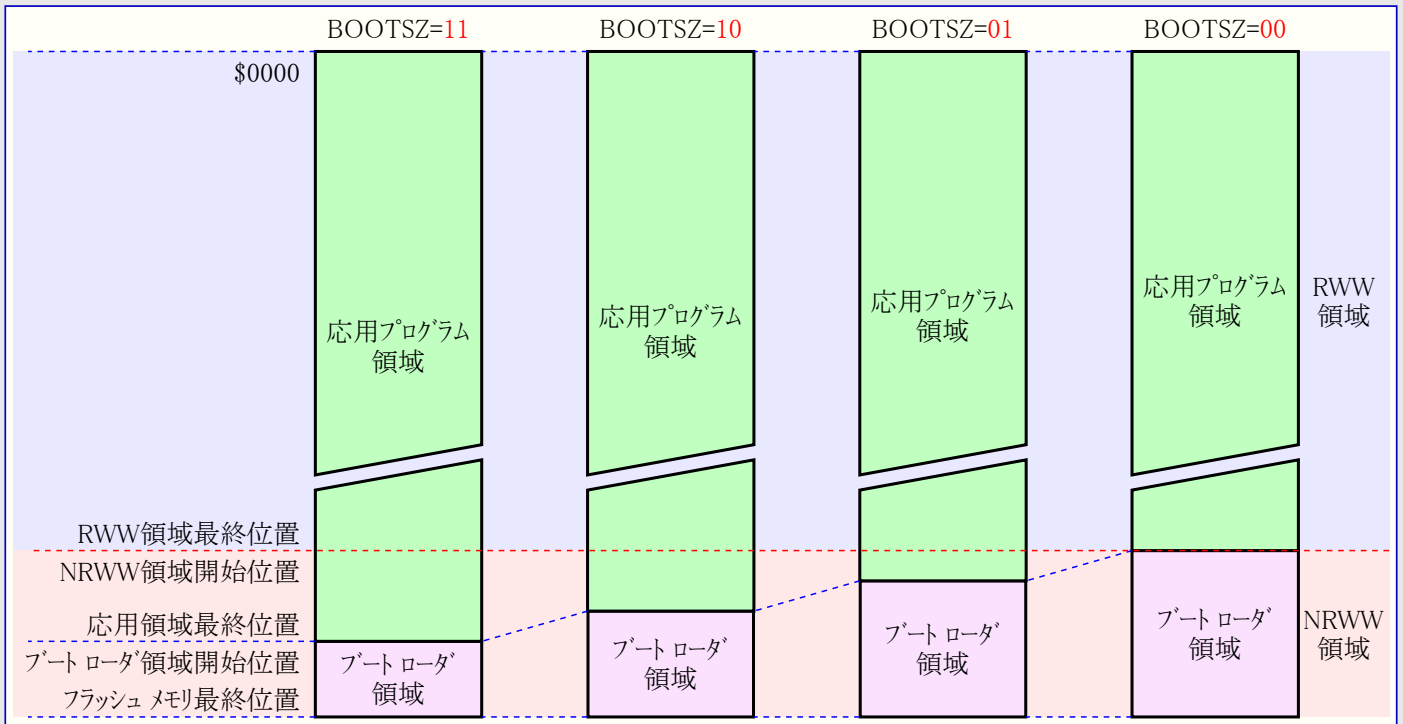


図24-2. 選択によるプログラム用フラッシュメモリの領域分割



注: 図内の各パラメータについては170頁の表24-6. で与えられます。

## 24.5. ブート ロード 施錠ビット

ブート ロード 能力が必要とされないなら、フラッシュ メモリ全体が応用コード用に利用可能です。ブート ロード は個別に設定可能な独立した2組のブート 施錠ビットを持ちます。これは異なる保護レベルを選ぶ独特な柔軟性を使用者に与えます。

使用者は以下を選べます。

- MCUによって更新するソフトウェアからフラッシュ メモリ全体を保護
- MCUによって更新するソフトウェアからフラッシュ メモリのブート ロード 領域だけを保護
- MCUによって更新するソフトウェアからフラッシュ メモリの応用領域だけを保護
- フラッシュ メモリ全体で更新するソフトウェアを許可

より多くの詳細については表24-2と表24-3をご覧ください。ブート 施錠ビットはソフトウェアと直列または並列のプログラミング動作で設定(0)できますが、これらのビットはチップ 消去指令によってのみ解除(1)できます。一般書き込み禁止(LB動作種別2)はSPM命令によるフラッシュ メモリのプログラミングを制御しません。同様に、試みられたなら、一般読み書き禁止(LB動作種別3)はLPM命令とSPM命令による読み込みも書き込みも制御しません。(訳補: 一般LBはLPM/SPM命令に関して無関係の意)

表24-2. 応用領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB0 動作種別	BLB02	BLB01	保護種別
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブート ロード 領域でのLPM命令による応用領域からの読み込みが許されません。(注)
4	0	1	ブート ロード 領域でのLPM命令による応用領域からの読み込みが許されません。(注)

注: BLB02=0で、割り込みベクタがブート ロード 領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

表24-3. ブート ロード 領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB1 動作種別	BLB12	BLB11	保護種別
1	1	1	LPM, SPM命令がブート ロード 領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブート ロード 領域に書くことを許されません。
3	0	0	SPM命令によるブート ロード 領域への書き込みと、応用領域でのLPM命令によるブート ロード 領域からの読み込みが許されません。(注)
4	0	1	応用領域でのLPM命令によるブート ロード 領域からの読み込みが許されません。(注)

注: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブート ロード 領域での実行時に割り込みが禁止されます。

## 24.6. ブート ロード プログラムへの移行

ブート ロード 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)によって行います。これはUSARTやSPIインターフェース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。代わりに、リセット後にリセット ベクタがブート ロード 領域開始アドレスを指示するようにブート リセット(BOOTRST)ヒューズをプログラム(0)できます。この場合、ブート ロード がリセット後に開始されます。応用コードが設定された(書かれた)後、そのプログラム(ブート ロード)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。これは一旦ブート リセット 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 ロード リセットを指示し、このヒューズが直列プログラミング、並列プログラミングまたはJTAG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。

表24-4. ブート リセット ヒューズ (0=プログラム、1=非プログラム)

BOOTRST	リセット後実行開始アドレス (リセット ベクタ)	
0	ブート ロード リセット	ブート ロード 開始アドレス(170頁の表24-6.参照)
1	応用リセット	\$0000



## 24.7. 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

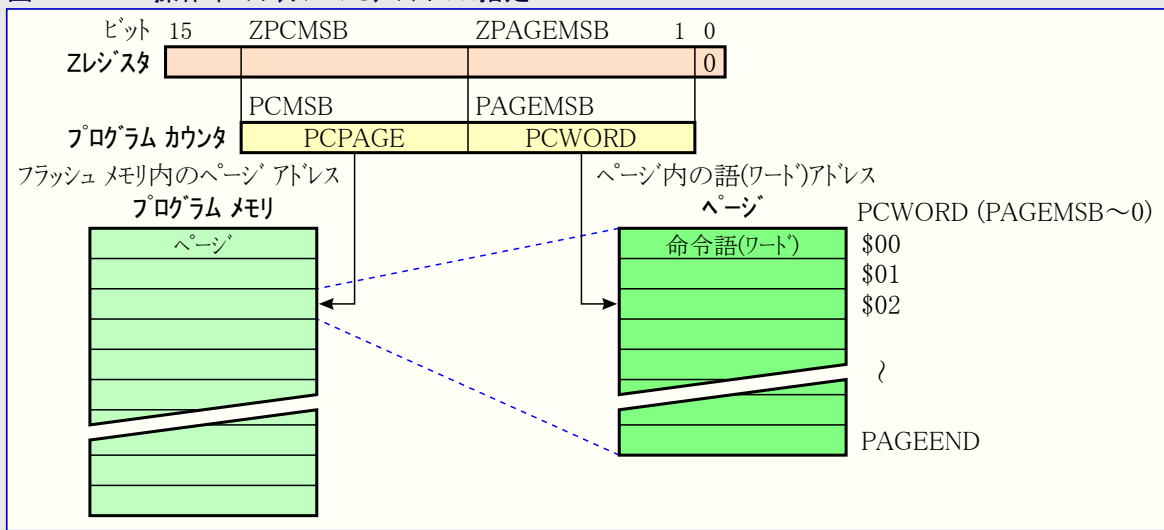
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

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(175頁の表25-11.参照)、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つように扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは図24-3.で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってブートローダソフトウェアはページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。一旦プログラミング操作が開始されると、このアドレスはラッチされ、Zポインタは他の操作に使えます。

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。LPM命令もアドレスを格納するのにZポインタを使います。この命令はフラッシュメモリのバイト単位をアドレス指定するので、Zポインタの最下位ビット(Z0)も使われます。

図24-3. SPM操作中のフラッシュメモリのアドレス指定



注: 図内で使った各変数は170頁の表24-8.で一覧されます。

PCPAGEとPCWORDは175頁の表25-11.で一覧されます。

## 24.8. フラッシュメモリの自己プログラミング

プログラムメモリはページ単位形式で更新されます。ページ一時緩衝部へ格納したデータでページを書く前にそのページが消去されなければなりません。ページ一時緩衝部はSPM命令使用時毎の1語(ワード)で満たされ、この緩衝部はページ消去命令前、またはページ消去とページ書き込み操作間のどちらかで満たすことができます。

### 手段1 (ページ消去前の一時緩衝部格納)

- ページ一時緩衝部を満たしてください。
- ページ消去を実行してください。
- ページ書き込みを実行してください。

### 手段2 (ページ消去後の一時緩衝部格納)

- ページ消去を実行してください。
- ページ一時緩衝部を満たしてください。
- ページ書き込みを実行してください。

ページの一部の変更だけが必要な場合、消去前にページの残す部分は(例えばページ一時緩衝部に)保存されなければならない、その後に変更して書かれます。手段1.を使う場合、初めにページを読んで必要な変更を行い、その後に変更したデータを書き戻すことをユーザーソフトウェアに許す効率的な読み-修正-書き(リット モデファイ ライト)機能をブートローダが提供します。手段2.が使われる場合、ページが既に消去されているため、格納中の旧データを読むことができません。ページ一時緩衝部は乱順でアクセスできます。ページ消去とページ書き込み操作の両方で使われるページアドレスは同じページをアドレス指定することが非常に重要です。アセンブリ言語でのコード例については169頁の「アセンブリ言語による簡単なブートローダ例」をご覧ください。

## 24.8.1. SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタにアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'X0000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGEに書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- RWW領域のページ消去 : ページ消去中、NRWW領域は読めます。
- NRWW領域のページ消去 : ページ消去中、CPUは停止されます。

## 24.8.2. ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタにアドレス、R1:R0にデータを設定してSPMCSRに'00000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORDの内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRの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ています。一時緩衝部を消去せずに各アドレスへ複数回書くことができないことに注意してください。

SPMページ設定操作の途中でEEPROMが書かれると、設定した全データが失われます。

## 24.8.3. ページ書き込みの実行

ページ書き込みを行うにはZポインタにアドレスを設定してSPMCSRに'X0000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGEに書かれなければなりません。この操作中に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領域のページ書き込み : ページ書き込み中、NRWW領域は読めます。
- NRWW領域のページ書き込み : ページ書き込み中、CPUは停止されます。

## 24.8.4. SPM操作可割り込みの使用法

SPM操作可割り込みが許可されると、SPMCSRのSPMENビットが解除(0)されている時にSPM操作可割り込みが継続的に発生します。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うとき、割り込みが読み出しに対して防がれる時に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(BLS)へ移動されるべきです。割り込み(ベクタ)の移動法は32頁の「割り込み」で記述されます。

## 24.8.5. ブートローダ領域更新中の考慮

ブート施錠ビット11(BLB11)が非プログラム(1)にされたままとすることによって使用者がブートローダ領域に更新を許す場合、特別な注意が祓われなければなりません。ブートローダ自身への予期せぬ書き込みはブートローダ全体を不正にし得て、更にソフトウェアの更新が不可能になるかもしれません。ブートローダ自体の変更が必要ないなら、内部ソフトウェアのどんな変更からもブートローダを保護するためにブート施錠ビット11(BLB11)をプログラム(0)することが推奨されます。

## 24.8.6. 自己プログラミング中のRWW領域読み込みの防止

自己プログラミング中(ページ消去もページ書き込みも)、RWW領域は読み出しに対して常に防がれます。使用者ソフトウェアそれ自身が自己プログラミング操作中にこの領域がアドレス指定されるのを防止しなければなりません。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の割り込みベクタ表は32頁の「割り込み」で記述されるようにブートローダ領域(BLS)へ移動されるべきか、または割り込みが禁止されなければなりません。プログラミングが完了した後に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の書き込みによってRWWSBを解除(0)しなければなりません。例については169頁の「アセンブリ言語による簡単なブートローダ例」をご覧ください。

## 24.8.7. SPM命令によるブートローダ施錠ビットと一般施錠ビットの設定

ブートローダ施錠ビットと一般施錠ビットを解除(0)するには希望したデータをR0に設定してSPMCSRに'X0001001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	LB2	LB1

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表24-2と表24-3をご覧ください。

R0のビット5～0が解除(0)される場合、SPMCSRでSPMENとブート施錠ビット設定(BLBSET)ビットが設定(1)された後4クロック周期内にSPM命令が実行されると、対応する施錠ビットがプログラム(0)されます。この操作中、Zポインタは関係ありませんが、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は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、この操作中に全てのフラッシュメモリは読むことができます。

#### 24.8.8. SPM命令での書き込み時のEEPROM書き込みによる妨害

EEPROM書き込み動作がフラッシュメモリへの全ソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(EECR)のEEPROM書き込み許可(EWE)ビットを検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前にこのビットが解除(0)されているのを確認することが推奨されます。

#### 24.8.9. ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビットの両方を読むことが可能です。施錠ビットを読むにはZポインタに\$0001を設定してSPMCSRのSPMENとブート施錠ビット設定(BLBSET)ビットを設定(1)してください。SPMENとBLBSETビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SPMENとBLBSETビットは施錠ビット読み出しの完了で、または3 CPU周期内にLPM命令が実行されないか、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SPMENとBLBSETビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	BLB12	BLB11	BLB02	BLB01	LB2	LB1

ヒューズ下位ビットを読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポインタに\$0000を設定してSPMCSRのSPMENとBLBSETビットを設定(1)してください。SPMENとBLBSETビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ下位ビット(FLB)の値が転送先レジスタに格納されます。ヒューズ下位ビットの配置と詳細な記述については173頁の表25-5を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様に、ヒューズ上位ビットを読むにはZポインタに\$0003を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるようにヒューズ上位ビット(FHB)の値が転送先レジスタに格納されます。ヒューズ上位ビットの配置と詳細な記述については173頁の表25-4を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

拡張ヒューズビットを読む時はZポインタに\$0002を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるように拡張ヒューズビット(EFB)の値が転送先レジスタに格納されます。拡張ヒューズビットの配置と詳細な記述については173頁の表25-3を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	EFB2	EFB1	EFB0

プログラム(0)されたヒューズと施錠ビットは0として読みます。非プログラム(1)にされたヒューズと施錠ビットは1として読みます。

#### 24.8.10. フラッシュメモリデータ化けの防止

低VCCの期間中、CPUとフラッシュメモリの正しい動作に対して供給電圧が低すぎるためにフラッシュメモリのプログラムが不正にされ得ます。これらの問題はフラッシュメモリを使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

フラッシュメモリのプログラム化けは電圧が低すぎる時の2つの状態によって起こされます。1つ目としてフラッシュメモリへの通常の書き込み手順は正しく動作するための最低電圧が必要です。2つ目として供給電圧が低すぎると、CPU自身が命令を間違えて実行し得ます。

フラッシュメモリ化けは次の推奨設計によって容易に避けられます(1つは必須)。

- 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。
- 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。

#### 24.8.11. SPM命令使用時のフラッシュメモリ用プログラミング(書き込み)時間

校正された内蔵RC発振器がフラッシュメモリアクセス時間に使われます。表24-5はCPUからのフラッシュメモリアクセスに対する代表的なプログラミング時間を示します。

表24-5. SPM命令によるフラッシュメモリのプログラミング時間

項目	最小	最大
SPM命令によるフラッシュ書き込み (ページ消去、ページ書き込み、施錠ビット書き込み)	3.7ms	4.5ms

注: 最小と最大の時間は(項目の)個別操作毎に対してです。

## 24.8.12. アセンブリ言語による簡単なブートローダ例

このルーチンはRAMからフラッシュメモリへ1ページのデータを書きます。RAM内の最初のデータ位置はYレジスタによって指示され、フラッシュメモリ内の最初のデータ位置はZレジスタによって指示されます。異常処理は含まれません。このルーチン(少なくともSPMJサブルーチン)はブートローダ領域側に配置されなければなりません。NRWW領域側のコードだけが自己プログラミング(ページ消去とページ書き込み)中に読めます。使用レジスタはR0,R1,TMP,CNTL,CNTH,SPMCで、レジスタの保存と復帰はこのルーチン内に含まれず、使用レジスタはコード量を犠牲にすれば最適化できます。割り込み表がブートローダ領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更になります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令	注釈
	.EQU PGSZB = PAGESIZE*2 .ORG SMALLBOOTSTART	; PGSZBはページ内のバイト数です。(PAGESIZEは語(ワート)数) ;
WRPG:	LDI SPMC, (1<<PGERS)+(1<<SPMEN) CALL SPMJ	; [ページ消去] ; ページ消去SPMCSR値を取得 ; ページ消去 ; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) CALL SPMJ	; RWW領域読み出し許可SPMCSR値を取得 ; RWW領域読み出し許可
WLP:	LDI CNTL, LOW(PGSZB) LDI CNTH, HIGH(PGSZB) LD R0, Y+ LD R1, Y+ LDI SPMC, (1<<SPMEN) CALL SPMJ ADIW ZH:ZL, 2 SBIW CNTH:CNTH, 2 BRNE WLP	; [RAMからフラッシュページ一時緩衝部へ転送] ; バイト計数器を初期化 ; (削除) ; RAM上の下位データを取得(ポインタ進行) ; RAM上の上位データを取得(ポインタ進行) ; ページ一時緩衝部書き込みSPMCSR値を取得 ; 対応語(ワート)データをページ一時緩衝部に設定 ; ページ一時緩衝部ポインタ進行 ; 計数器を減数 (SUBI) ; 指定バイト数分継続 ; [ページ書き込み] ; ページ一時緩衝部先頭にポインタを復帰 ; (削除)
	SUBI ZL, LOW(PGSZB) SBCI ZH, HIGH(PGSZB) LDI SPMC, (1<<PGWRT)+(1<<SPMEN) CALL SPMJ	; フラッシュ書き込みSPMCSR値を取得 ; フラッシュメモリページ書き込み ; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) CALL SPMJ	; RWW領域読み出し許可SPMCSR値を取得 ; RWW領域読み出し許可 ; [読み戻し照合(任意)]
	LDI CNTL, LOW(PGSZB) LDI CNTH, HIGH(PGSZB) SUBI YL, LOW(PGSZB) SBCI YH, HIGH(PGSZB)	; バイト計数器を初期化 ; (削除) ; RAMデータ先頭にポインタを復帰
RLP:	LPM R0, Z+ LD R1, Y+ CPSE R0, R1 JMP ERROR	; フラッシュメモリから1バイト取得(ポインタ進行) ; RAMから1バイトデータを取得(ポインタ進行) ; 値一致でスキップ ; 不一致で異常処理へ
;	SBIW CNTH:CNTH, 1 BRNE RLP	; 計数器を減数 (SUBI) ; 指定バイト数分継続
:		; [RWW領域へ復帰]
RTN:	IN TMP, SPMCSR SBRS TMP, RWWSB RET	; SPM命令制御/状態レジスタ値を取得 ; RWW領域多忙でスキップ ; 準備可で呼び出し元へ復帰
;		; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) CALL SPMJ RJMP RTN	; RWW領域読み出し許可SPMCSR値を取得 ; RWW領域読み出し許可 ; RWW領域準備可まで待機へ ; [SPM命令実行サブルーチン]
SPMJ:	IN TMP, SPMCSR SBRC TMP, SPMEN RJMP SPMJ	; SPM命令制御/状態レジスタ値を取得 ; 操作可能(直前のSPM完了)でスキップ ; 操作可まで待機
;		
WAIT:	IN TMP, SREG CLI SBIC EECR, EEWEIF RJMP WAIT	; ステータスレジスタ値を保存 ; 全割り込み禁止 ; EEPROM書き込み中以外でスキップ ; EEPROM書き込み完了まで待機
;		
	OUT SPMCSR, SPMC SPM OUT SREG, TMP RET	; SPM動作指定 ; 対応SPM動作実行 ; ステータスレジスタ値を復帰 ; 呼び出し元へ復帰



### 24.8.13. ATmega325P/3250P用ブートローダのパラメータ

自己プログラミングの記述で使ったパラメータは表24-6.~8.で与えられます。

表24-6. 応用領域とブートローダ領域の分割設定

BOOTSZ1	BOOTSZ0	ブートローダ領域		アドレス範囲		
		容量 (語)	ページ数	応用プログラム領域	ブートローダプログラム領域	ブートリセットアドレス
1	1	256	4	\$0000~\$3EFF	\$3F00~\$3FFF	\$3F00
1	0	512	8	\$0000~\$3DFF	\$3E00~\$3FFF	\$3E00
0	1	1024	16	\$0000~\$3BFF	\$3C00~\$3FFF	\$3C00
0	0	2048	32	\$0000~\$37FF	\$3800~\$3FFF	\$3800

注: 各種BOOTSZヒューズ設定は図24-2.で示されます。

表24-7. RWW領域とNRWW領域の範囲

領域	ページ数	アドレス範囲
書き込み中読み出し可能(RWW)領域	224	\$0000~\$37FF
書き込み中読み出し不能(NRWW)領域	32	\$3800~\$3FFF

注: これら2つの領域についての詳細に関しては164頁の「RWW - 書き込み中読み出し可能領域」と「NRWW - 書き込み中読み出し不能領域」をご覧ください。

表24-8. 図24-3.で使った各変数説明とZポイントの配置

変数名	対応値		意味
	PC	Zポイント (注)	
PCMSB	PC13		プログラムカウンタの最上位ビット。(プログラムカウンタは14ビット、PC13~0)
PAGEMSB	PC5		1ページ内に使われる語(ワード)アドレスの最上位ビット。 (ページ内の64語(ワード)には6ビット PC5~0が必要)
ZPCMSB		Z14	PCMSBに配置される(対応する)Zレジスタ内のビット。 (Z0が使われないため、ZPCMSB=PCMSB+1)
ZPAGEMSB		Z6	PAGEMSBに配置される(対応する)Zレジスタ内のビット。 (Z0が使われないため、ZPAGEMSB=PAGEMSB+1)
PCPAGE	PC13~6	Z14~7	プログラムカウンタ ページ アドレス: ページ消去とページ書き込み用のページ選択
PCWORD	PC5~0	Z6~1	プログラムカウンタ 語(ワード)アドレス: 一時緩衝部格納用語(ワード)選択 (ページ書き込み操作中は0でなければなりません。)

注: Z0: 全てのSPM命令に対して0であるべきで、LPM命令に対するバイト選択です。

自己プログラミング中のZポイントの使用についての詳細に関しては166頁の「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」をご覧ください。



## 24.9. ブート ロード用レジスタ

### 24.9.1. SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register) SPMCSR

このレジスタはブート ロード動作を制御するため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	—	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	SPMCSR
Read/Write	R/W	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7 – 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SPMCSRの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)

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)操作が開始されると、RWWSBがハードウェアによって設定(1)されます。RWWSBビットが設定(1)されると、RWW領域はアクセスできません。自己プログラミングが完了された後にRWWSREビットが1を書かれると、このRWWSBビットは解除(0)されます。同様に、ページ設定操作が開始されると、RWWSBビットは自動的に解除(0)されます。

#### • ビット5 – Res : 予約 (Reserved)

このビットは予約されており、常に0として読みます。

#### • ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域にプログラミング(ページ消去またはページ書き込み)すると、RWW領域は(RWWSBがハードウェアによって設定(1)され)読み出しに對して防がれます。RWW領域を再許可するために使用者ソフトウェアはプログラミングが完了(SPMENが解除(0))されるまで待たねばなりません。その後、RWWSREビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令がRWW領域を再許可します。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(SPMEN=1)の間、RWW領域は再許可できません。フラッシュメモリが設定されている間にRWWSREビットが書かれると、フラッシュメモリ設定操作は失敗し、設定したデータが失われます。

#### • ビット3 – BLBSET : ブート施錠ビット設定 (Boot Lock Bits Set)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令はR0のデータに従ったブート周期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。BLBSETビットは周期ビットの設定完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。

SPMCSRでBLBSETとSPMENが設定(1)された後の3クロック周期内のLPM命令は(ZポインタのZ0によって)ヒューズビットまたは周期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168頁の「ソフトウェアからのヒューズビットと施錠ビットの読み出し」をご覧ください。

#### • ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。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を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。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RWWSRE、BLBSET、PGWRT、PGERSのどれかと共に1を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。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したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SPMENビットはSPM命令の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ消去とページ書き込み中、SPMENビットはその動作が完了されるまで1に留まります。

下位5ビットに10001, 01001, 00101, 00011, 00001以外のどんな組み合わせを書いても無効です。

## 25. メモリプログラミング

### 25.1. プログラム メモリとデータ メモリ用施錠ビット

ATmega325P/3250Pは非プログラム(1)のままか、表25-2で一覧される付加機能を得るためにプログラム(0)できる6つの施錠ビットを提供します。この施錠ビットはチップ消去指令でのみ1に消去できます。

表25-1. 施錠ビット バイトの内容

名称	ビット番号	意味	既定値 (注)
—	7		1 (非プログラム)
—	6		1 (非プログラム)
BLB12	5	ブート ロータ領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB11	4		1 (非プログラム)
BLB02	3	応用領域に対する保護用ブート施錠ビット	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	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表25-2. 施錠ビットの保護種別

メモリ施錠ビット			保護種別
LB 種別	LB2	LB1	直列、並列、またはJTAG経由プログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズ ビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズ ビットとブート施錠ビットが固定されます。(注1)
LB0 種別	BLB02	BLB01	フラッシュ メモリの応用プログラム領域に対する保護
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブート ロータ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
4	0	1	ブート ロータ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
LB1 種別	BLB12	BLB11	フラッシュ メモリのブート ロータ プログラム領域に対する保護
1	1	1	LPM, SPM命令がブート ロータ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブート ロータ領域に書くことを許されません。
3	0	0	SPM命令によるブート ロータ領域への書き込みと、応用領域でのLPM命令によるブート ロータ領域からの読み込みが許されません。(注3)
4	0	1	応用領域でのLPM命令によるブート ロータ領域からの読み込みが許されません。(注3)

注: 0はプログラム、1は非プログラムを意味します。

注1: 施錠ビットを書く前にヒューズ ビットとブート施錠ビットを書いてください。

注2: BLB02=0で、割り込みベクタがブート ロータ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

注3: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブート ロータ領域での実行時に割り込みが禁止されます。

## 25.2. ヒューズ ビット

ATmega325P/3250Pには3つのヒューズ バイトがあります。表25-3.~5.は全てのヒューズの概略機能とヒューズ バイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

表25-3. 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
—	7		1 (非プログラム)
—	6		1 (非プログラム)
—	5		1 (非プログラム)
—	4		1 (非プログラム)
—	3		1 (非プログラム)
BODLEVEL1	2	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注1)	1 (非プログラム)
BODLEVEL0	1		1 (非プログラム)
RSTDISBL	0	外部リセット禁止(PG5として使用) (注2)	1 (非プログラム) 外部リセット有効

注1: BODLEVELヒューズの符号化については196頁の表26-6.をご覧ください。

注2: PG5は入力専用で、常にプルアップが有効です。50頁の「ポートGの交換機能」をご覧ください。

表25-4. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
OCDEN (注4)	7	内蔵デバッグ機能(OCD)許可。	1 (非プログラム) OCD禁止
JTAGEN (注5)	6	JTAGインターフェース許可。	0 (プログラム) JTAG許可
SPIEN (注1)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON (注3)	4	ウォッチドッグ タイマ常時有効。	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BOOTSZ1	2	ブート ロータ容量選択。(表24-6.参照) (注2)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOOTRST	0	リセット ベクタ(応用領域またはブート ロータ領域)選択。	1 (非プログラム) 応用領域

注1: SPIENヒューズは直列プログラミングでアクセスできません。

注2: BOOTSZ1,0既定値は最大ブート ロータ容量になります。170頁の表24-6.をご覧ください。

注3: 詳細については30頁の「ウォッチドッグ タイマ制御レジスタ(WDTCSR)」をご覧ください。

注4: 施錠ビットやJTAGENヒューズの設定に拘らず、OCDENヒューズがプログラム(0)された製品の出荷は決してありません。プログラム(0)されたOCDENヒューズは全休止形態でクロック系の一部を許可します。これは消費電力の増加になってしまいます。

注5: JTAGインターフェースが未接続のままにされる場合、JTAGENヒューズは可能ならば禁止されるべきです。これはJTAGインターフェースのTDOピンの静止時電流を避けるためです。

表25-5. ヒューズ 下位バイト一覧

名称	ビット	意味	既定値
CKDIV8 (注4)	7	システム クロック 8分周選択。	0 (プログラム) 8分周
CKOUT (注3)	6	システム クロック出力許可。	1 (非プログラム) 不許可
SUT1	5	起動時間選択。 (注1)	1 (非プログラム)
SUT0	4		0 (プログラム)
CKSEL3	3	クロック種別選択。 (注2)	0 (プログラム)
CKSEL2	2		0 (プログラム)
CKSEL1	1		1 (非プログラム)
CKSEL0	0		0 (プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については19頁の表8-9.をご覧ください。

注2: CKSEL3~0の既定設定は8MHz校正付き内蔵RC発振器になります。詳細については18頁の表8-1.をご覧ください。

注3: CKOUTはポートE7に出力することをシステム クロックに許します。詳細については20頁の「クロック出力緩衝部」をご覧ください。

注4: 詳細については20頁の「システム クロック前置分周器」をご覧ください。

ヒューズ ビットの状態はチップ消去により影響されません。施錠ビット1(LB1)がプログラム(0)されると、ヒューズ ビットが固定されることに注意してください。施錠ビットをプログラム(0)する前にヒューズ ビットをプログラミング(書き込み)してください。

### 25.2.1. ヒューズ'のラッチ

ヒューズ'値はデバイスがプログラミング動作へ移行する時にラッチされ、ヒューズ'値の変更はデバイスがプログラミング動作を去るまで無効です。これは一旦プログラム(0)されると直ぐに効果があるEESAVEヒューズ'には適用されません。ヒューズ'は通常動作での電源投入でもラッチされません。

### 25.3. 識票バイト

全てのAtmelマイクロ コントローラはデバイスを識別する3バイトの識票符号を持ちます。この符号は直列と並列の両プログラミング動作で、またデバイスが施錠されていても読めます。この3バイトは分離された空間に存在します。ATmega325P/3250P用の識票バイトは表25-6.で与えられます。

表25-6. デバイスとJTAGの識別番号(ID)

部品番号	識票バイト アドレス			JTAG	
	\$0000	\$0001	\$0002	部品番号	製造者ID
ATmega325P	\$1E	\$95	\$0D	\$950D	\$01F
ATmega3250P	\$1E	\$95	\$0E	\$950E	\$01F

### 25.4. 校正バイト

ATmega325P/3250Pは内蔵RC発振器用に1バイトの校正値を持っています。このバイトは識票アドレス空間でアドレス\$0000の上位バイトにあります。リセット中、校正付き内蔵RC発振器の正しい周波数を保証するために、このバイトが発振校正(OSCCAL)レジスタへ自動的に書かれます。

## 25.5. 並列プログラミング

この項はATmega325P/3250Pのプログラム用フラッシュメモリ、データ用EEPROM、メモリの**施錠ビット**、**ヒューズビット**の並列プログラミングと照合の方法を記述します。特記事項を除いて、パルス幅は最低250nsと仮定されます。

### 25.5.1. 信号名

本章でATmega325P/3250Pのいくつかのピンは並列プログラミング中の機能を表す信号名で参照されます。**図25-1**と**表25-7**をご覧ください。後続の表で示されないピンはピン名によって参照されます。

XA0とXA1ピンはXTAL1ピンが正パルスを与えられる時に実行される動作を決めます。このビット符号化は**表25-9**で示されます。

WRまたはOEパルス送出处、設定された指令が決める動作が実行されます。各種指令は**表25-10**で示されます。

図25-1. 並列プログラミング構成図

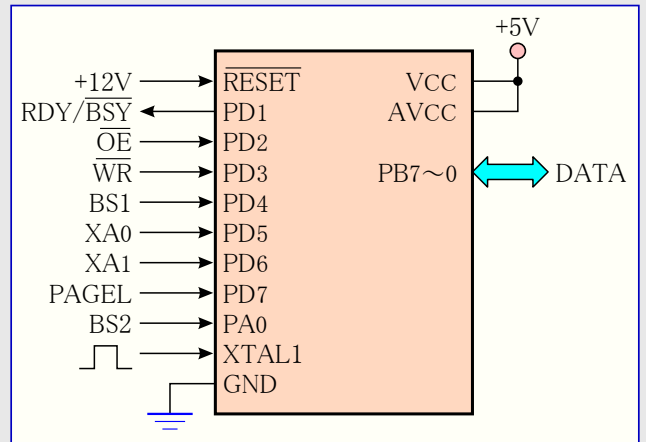


表25-7. 信号名とピン名の関係

信号名	ピン名	入出力	機能
RDY/BSY	PD1	出力	0(Low) : 多忙(プログラミング中)      1(High) : 準備可(指令受付可)
OE	PD2	入力	出力許可(負論理)
WR	PD3	入力	書き込みパルス(負論理)
BS1	PD4	入力	上位/下位バイト選択1 (0:下位, 1:上位)      (一般用)
XA0	PD5	入力	XTAL動作ビット0
XA1	PD6	入力	XTAL動作ビット1
PAGEL	PD7	入力	プログラムメモリとEEPROMデータをページ一時緩衝部に設定
BS2	PA0	入力	上位/下位バイト選択2 (0:下位, 1:上位)      (ヒューズビット用)
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

表25-8. プログラミング動作移行時のピン値

ピン名	シンボル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

表25-9. XA0とXA1の符号化(機能)

XA1	XA0	XTAL1パルス時の動作
0	0	フラッシュまたはEEPROMのアドレス設定 (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)

表25-10. 指令バイトのビット符号化

指令バイト	指令の機能
\$80 (1000 0000)	チップ消去
\$40 (0100 0000)	ヒューズビット書き込み
\$20 (0010 0000)	施錠ビット書き込み
\$10 (0001 0000)	フラッシュメモリ書き込み
\$11 (0001 0001)	EEPROM書き込み
\$08 (0000 1000)	識票、校正バイト読み出し
\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$02 (0000 0010)	フラッシュメモリ読み出し
\$03 (0000 0011)	EEPROM読み出し

表25-11. フラッシュメモリのページ数とページの語数

ATmega	全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
325P 3250P	16K語 (32Kバイト)	64語 (128バイト)	PC5~0	256	PC13~6	13

表25-12. EEPROMメモリのページ数とページの語数

ATmega	全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
325P 3250P	1Kバイト	4バイト	EEA1~0	256	EEA9~2	9



## 25.6. 並列プログラミング手順

### 25.6.1. 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、 $\overline{\text{RESET}}$ ピンを0V、175頁の表25-8.で一覧されるProg\_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5～5.5Vを印加します。
- ③ 20 $\mu$ s以内にVCCが最低1.8Vに達することを保証してください。
- ④ 20～60 $\mu$ s待ち、 $\overline{\text{RESET}}$ に11.5～12.5Vを印加します。
- ⑤ Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu$ s、Prog\_enableピンを無変化に保ちます。
- ⑥ 如何なる並列プログラミング指令を与えるのにも先立って少なくとも300 $\mu$ s間待ちます。
- ⑦ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

VCCの上昇時間が上で示した必要条件を完全に満たせない場合、次の代替手順が使えます。

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、175頁の表25-8.で一覧されるProg\_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5～5.5Vを印加します。
- ③ VCCを監視し、0.9～1.1Vに達したら直ぐ、 $\overline{\text{RESET}}$ に11.5～12.5Vを印加します。
- ④ Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu$ s、Prog\_enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立ってVCCが実際に4.5～5.5Vに達するまで待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

### 25.6.2. 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。この考慮は識票バイト読み出しにも適用されます。

### 25.6.3. チップ消去

チップ消去はフラッシュメモリ、EEPROM(注1)、施錠ビットを消去します。施錠ビットはプログラムメモリが完全に消去されてしまうまでリセット(消去)されません。ヒューズビットは変更されません。チップ消去はフラッシュメモリやEEPROMが再書き込みされる前に実行されなければなりません。

**注1:** EESAVEヒューズがプログラム(0)されていると、EEPROMはチップ消去中、保護されます。

「チップ消去」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$80(1000 0000)に設定します。これはチップ消去指令です。
- ④ XTAL1に正パルスを与えます。これはチップ消去指令を設定します。
- ⑤  $\overline{\text{WR}}$ に負パルスを与えます。これはチップ消去を開始します。RDY/ $\overline{\text{BSY}}$ がLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/ $\overline{\text{BSY}}$ がHigh(1)になるまで待ちます。

## 25.6.4. フラッシュ メモリ書き込み (図25-3.タイミングを参照)

フラッシュ メモリはページで構成されます(175頁の表25-11.参照)。フラッシュ メモリへ書く時にプログラム データはページ緩衝部にラッチされます。これは同時に書かれることをプログラム データの1ページに許します。次の手順は完全なフラッシュ メモリの書き込み方法を記述します。

### A. 「フラッシュ メモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュ メモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュ メモリ書き込み指令を設定します。

### B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選びます。
- ③ DATAにアドレス下位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス下位バイトを設定します。

### C. データ下位バイト設定

- ① XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ② DATAにデータ下位バイト(\$00～\$FF)を設定します。
- ③ XTAL1に正パルスを与えます。これはデータ下位バイトを設定します。

### D. データ上位バイト設定

- ① BS1をHigh(1)に設定します。これは上位バイトを選びます。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはデータ上位バイトを設定します。

### E. 語(ワード)データをページ一時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイトを選びます。
- ② PAGELに正パルスを与えます。これは語データをページ一時緩衝部にラッチ(設定)します。

### F. 緩衝部全体が満たされるか、または(必要な)全てのデータが緩衝部内に設定されるまで、B～Eを繰り返す

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、上位ビットがフラッシュ メモリ内のページをアドレス指定します。これは図25-2.で図示されます。ページ内の語アドレスに8ビット未満が必要とされる場合(ページ容量<256)、アドレス下位バイトの最上位(側)ビットがページ書き込み実行時のページ アドレスに使われることに注意してください。

### G. アドレス上位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選びます。
- ③ DATAにアドレス上位バイト(\$00～\$3F)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイトを設定します。

### 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/BSYがLow(0)になります。
- ③ RDY/BSYがHigh(1)になるまで待ちます。

### I. フラッシュ メモリ全部が書かれるか、または(必要な)全データが書かれてしまうまで、B～Hを繰り返す

### 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 XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図25-2. ページで構成されたフラッシュ メモリのアドレス指定

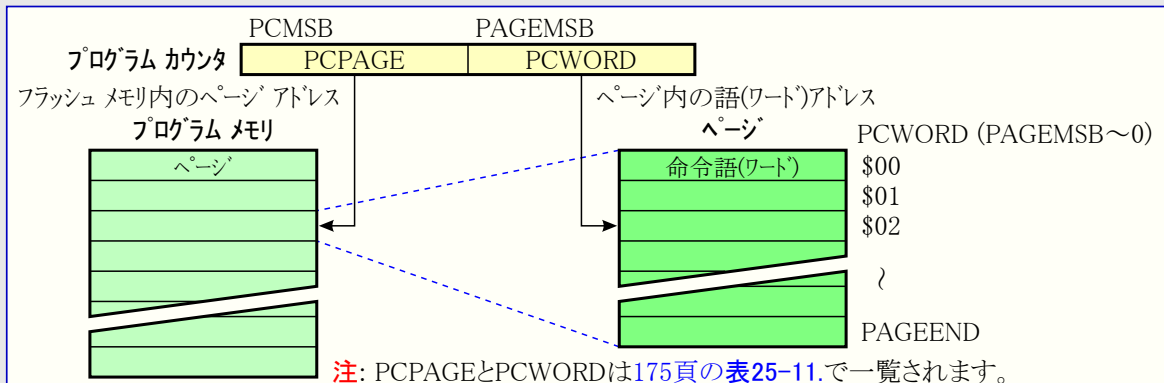
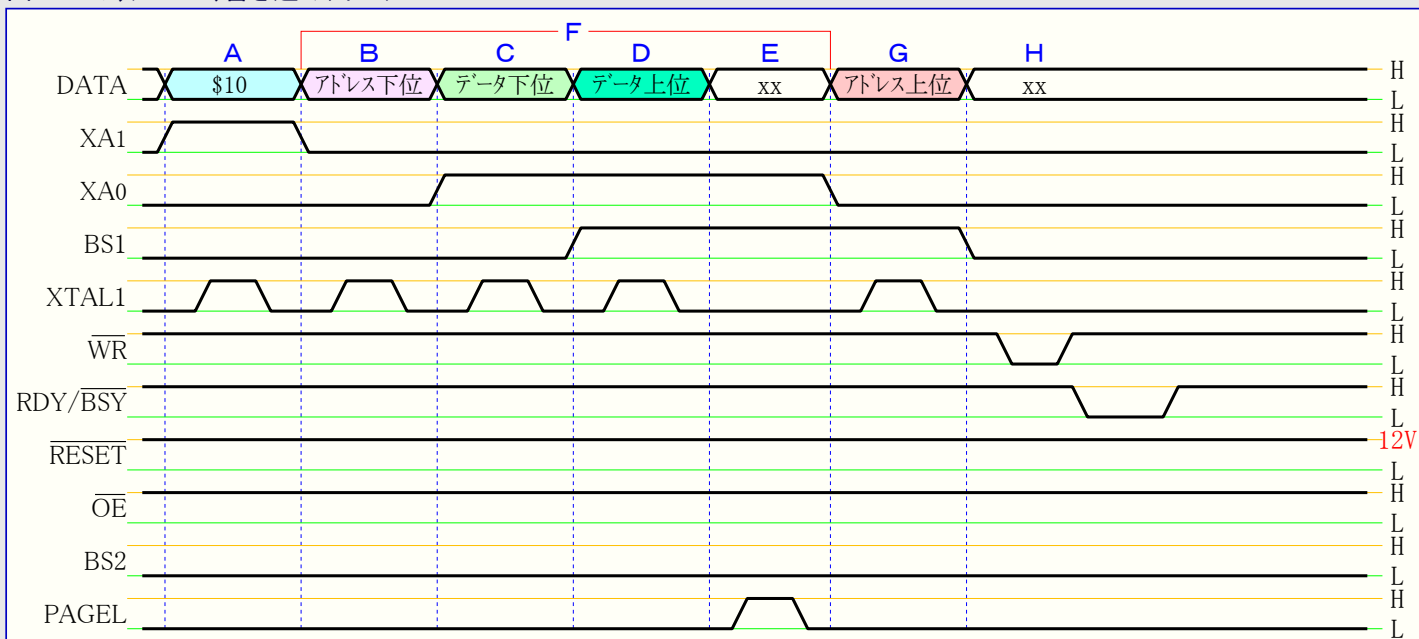


図25-3. フラッシュ メリ書き込みタイミング



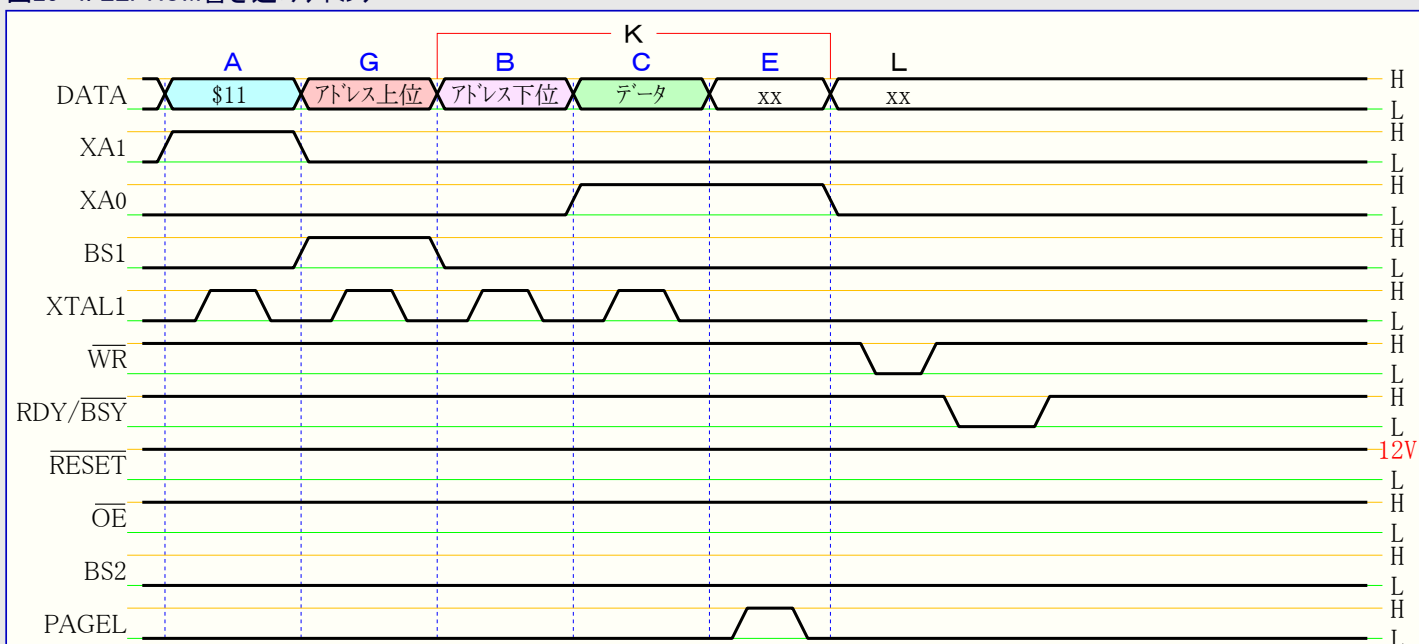
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

### 25.6.5. EEPROM書き込み

EEPROMはページで構成されます(175頁の表25-12参照)。EEPROMを書く時にデータはページ緩衝部にラッチされます。これは同時に書かれることをデータの1ページに許します。データ用EEPROMメモリの書き込み方法は次のとおりです。(指令、アドレス、データ設定の詳細については177頁の「フラッシュ メリの書き込み」を参照。図25-4. タイミング参照。)

1. EEPROM書き込み指令\$11(0001 0001)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00～\$03)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00～\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. データバイト(\$00～\$FF)を設定します。(「フラッシュ メリ書き込み」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 WRに負パルスを与えます。これはEEPROMページ書き込みを開始します。RDY/BSYがLow(0)になります。
  - ③ 次のページを書く前に、RDY/BSYがHigh(1)になるまで待ちます。

図25-4. EEPROM書き込みタイミング



## 25.6.6. フラッシュ メリ読み出し

フラッシュ メリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については177頁の「フラッシュ メリの書き込み」を参照)

1. フラッシュ メリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$3F)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。フラッシュ メリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュ メリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. OEをHigh(1)に設定します。DATAはHi-Zになります。

## 25.6.7. 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177頁の「フラッシュ メリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$03)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。EEPROMのバイト データが直ぐにDATAで読めます。
5. OEをHigh(1)に設定します。DATAはHi-Zになります。

## 25.6.8. 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。)

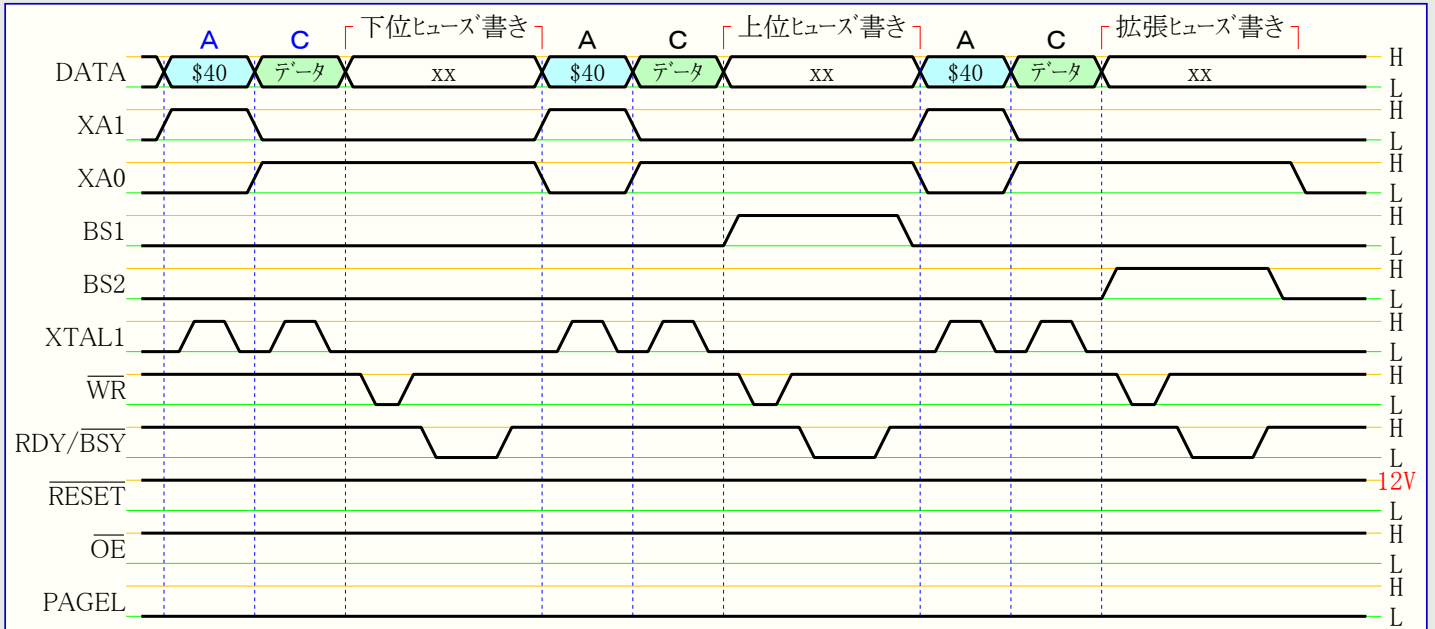
各ヒューズ バイトの書き込み方法は次のとおりです。(指令とデータ設定の詳細については177頁の「フラッシュ メリの書き込み」を参照)

1. ヒューズ ビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュ メリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4. WRに負パルスを与え、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びます。

表A. ヒューズ バイト対応BS1,BS2設定

ヒューズ バイト	BS1	BS2
拡張バイト	Low(0)	High(1)
上位バイト	High(1)	Low(0)
下位バイト	Low(0)	Low(0)

図25-5. ヒューズ書き込みタイミング



### 25.6.9. 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177頁の「フラッシュメモリの書き込み」を参照)

1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. データ下位バイト(としてデータ)を設定します。0=プログラム,1=無変化です。LB保護種別3が設定(LB1とLB2がプログラム(0))されると、どの外部的なプログラミング動作種別によってもブート施錠ビットはプログラミングできません。(「フラッシュメモリの書き込み」のCを参照)
3.  $\overline{WR}$ に負パルスを与え、RDY/BSYがHighになるまで待ちます。

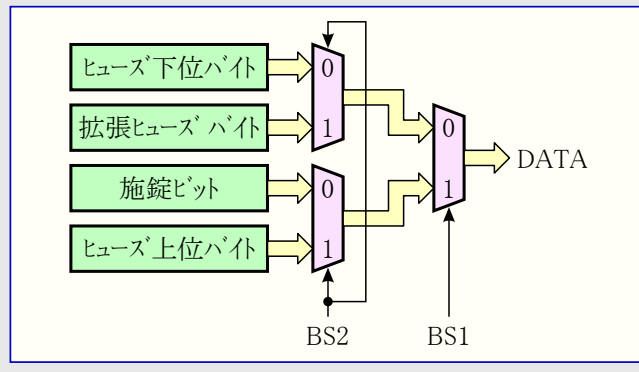
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

### 25.6.10. 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177頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズビットと施錠ビットの読み出し指令\$04(0000 0100)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. BS1とBS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ下位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
3. BS1とBS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ上位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
4. BS1をLow(0)、BS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。拡張ヒューズビットの状態が直ぐにDATAで読めます。(0=プログラム)
5. BS1をhigh(1)、BS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。施錠ビットの状態が直ぐにDATAで読めます。(0=プログラム)
6.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

図25-6. ヒューズ・施錠ビット読み出し中のBS1, BS2との関係



### 25.6.11. 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177頁の「フラッシュメモリの書き込み」を参照)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。選んだ識票バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

### 25.6.12. 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177頁の「フラッシュメモリの書き込み」を参照)

1. 校正バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイトに\$00を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。校正バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。



## 25.6.13. 並列プログラミング特性

図25-7. 並列プログラミング タイミング (一般的な必要条件)

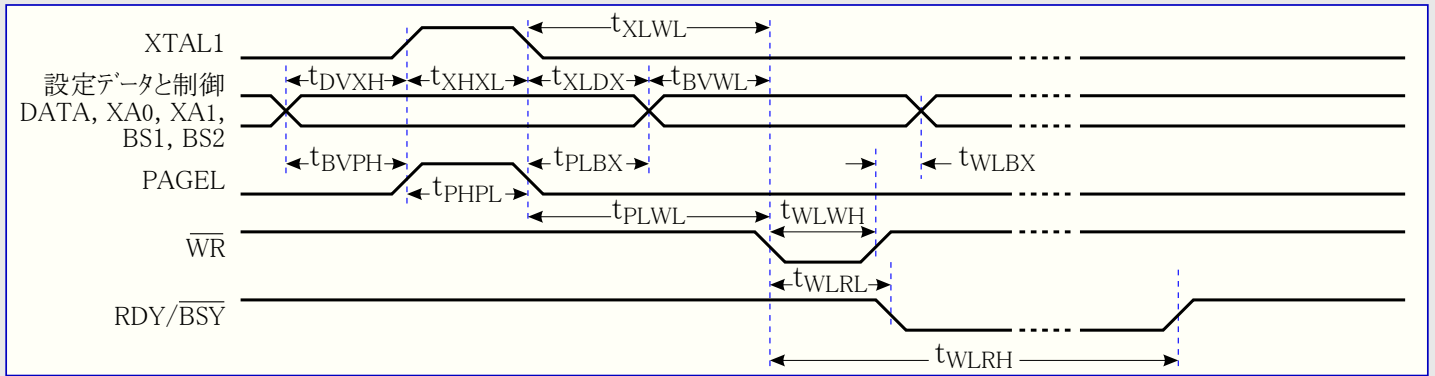
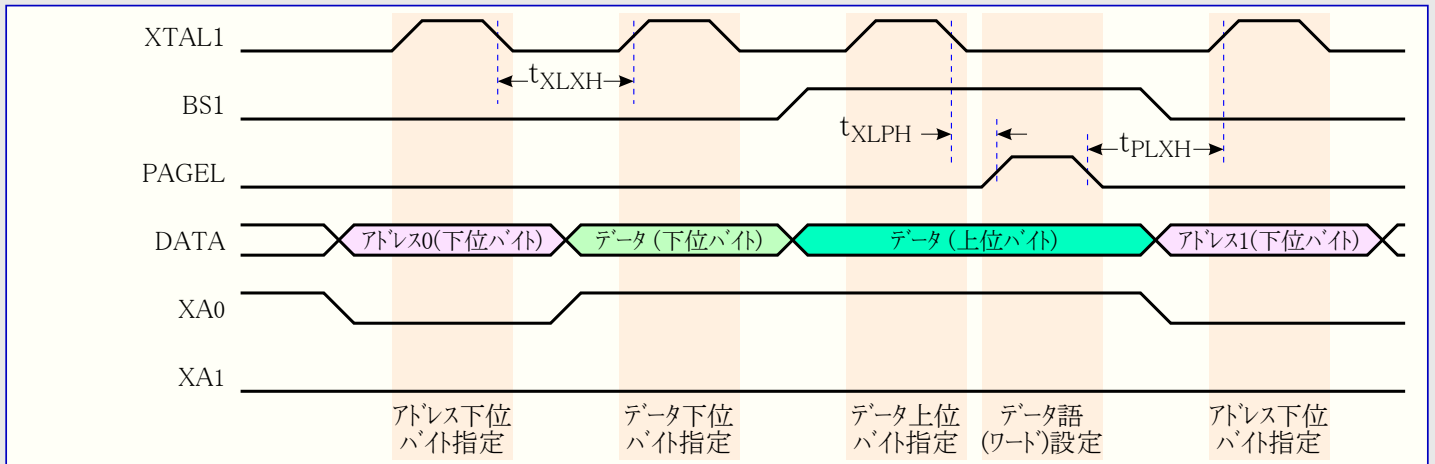
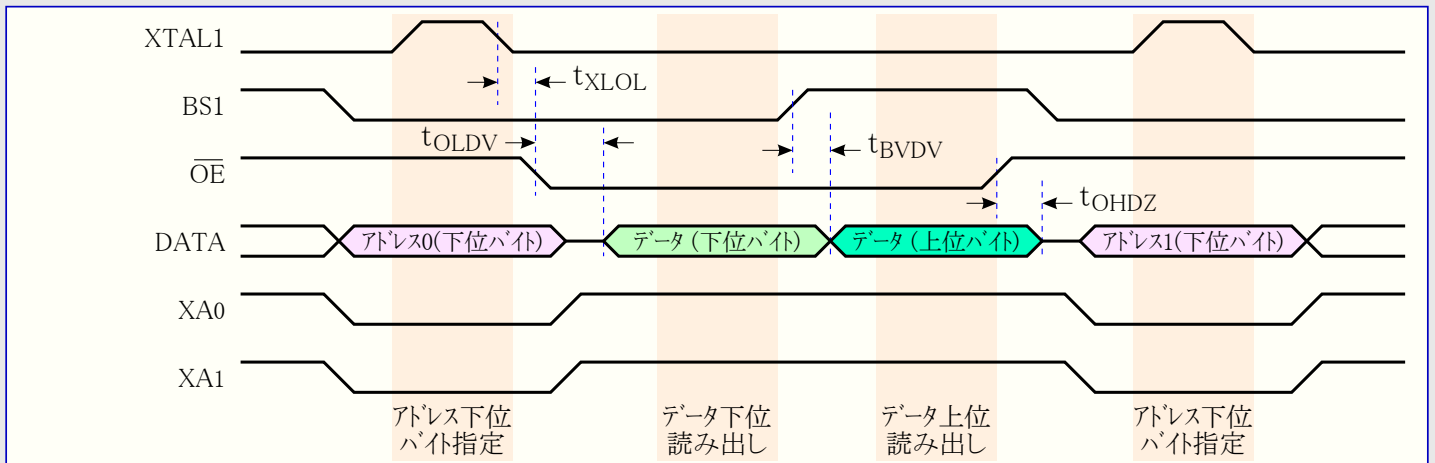


図25-8. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図25-7.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25-9. 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25-7.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25-13. 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
VPP	プログラミング許可電圧	11.5		12.5	V
IPP	プログラミング許可電流			250	μA
tDVXH	XTAL1 ↑ に対するデータと制御の準備時間	67			ns
tXLXH	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
tXHXL	XTAL1 Highパルス幅	150			
tXLDX	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
tXLWL	XTAL1パルス ↓ 後のWR ↓ 待機時間	0			
tXLPH	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
tPLXH	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			
tBVPH	PAGELパルス ↑ に対するBS1準備時間	67			
tPHPL	PAGEL Highパルス幅	150			
tPLBX	PAGELパルス ↓ 後のBS1保持時間	67			
tWLBX	WRパルス ↓ 後のBS1,BS2保持時間	67			
tPLWL	PAGELパルス ↓ 後のWRパルス ↓ 待機時間	67			
tBVWL	WRパルス ↓ に対するBS1準備時間	67			
tWLWH	WR Lowパルス幅	150			
tWLRL	WRパルス ↓ 後のRDY/BSY ↓ 遅延時間	0		1	μs
tWLRH	書き込み時間 (WR ↓ からRDY/BSY ↑) (注1)	3.7		4.5	ms
tWLRH_CE	チップ消去時間 (WR ↓ からRDY/BSY ↑) (注2)	7.5		9	
tXLCL	XTAL1パルス ↓ 後のOE ↓ 待機時間	0			ns
tBVDV	BS1有効からのDATA遅延時間	0		250	
tOLDV	OE ↓ 後のDATA出力遅延時間			250	
tOHDZ	OE ↑ 後のDATA Hi-Z遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

## 25.7. 直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され得るのに先立って**プログラミング許可命令**が初めに実行されることを必要とします。**注意**、表25-14.でSPIプログラミング用のピン配置が一覧されます。全てのデバイスが内部SPIインターフェースに対する専用SPIピンを使うとは限りません。

### 25.7.1. 直列プログラミング用ピン配置

表25-14. 直列プログラミング用ピン配置

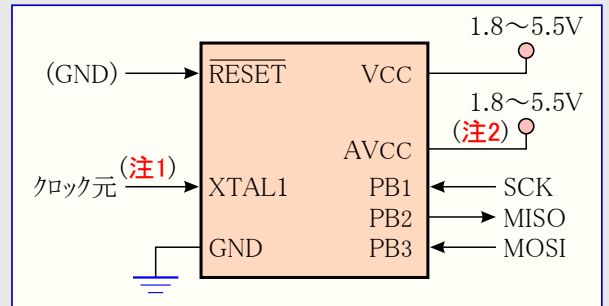
信号名	ピン名	入出力	機能
SCK	PB1	入力	直列クロック
MOSI	PB2	入力	直列データ入力
MISO	PB3	出力	直列データ出力

EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(直列プログラミングのみ)、**チップ消去命令**を初めに実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全てのメモリ位置の内容を\$FFにします。

CKSELヒューズによる有効なクロックが与えられなければなりません。直列クロック(SCK)入力のLowとHighの区間の最小値は次のように定義されます。

$f_{CK} < 12\text{MHz}$ : Low区間 > 2 CPUクロック周期	$f_{CK} < 12\text{MHz}$ : High区間 > 2 CPUクロック周期
$f_{CK} \geq 12\text{MHz}$ : Low区間 > 3 CPUクロック周期	$f_{CK} \geq 12\text{MHz}$ : High区間 > 3 CPUクロック周期

図25-10. 直列プログラミング構成図



**注1:** デバイスが内蔵発振器で動作する場合、XTAL1ピンにクロック元を接続する必要はありません。

**注2:**  $VCC - 0.3V < AVCC < VCC + 0.3V$  ですが、AVCCは常に1.8~5.5V内にすべきです。

## 25.7. 直列プログラミング手順

ATmega325P/3250Pに直列データを書く時にデータはSCKの上昇端で行われ、ATmega325P/3250Pから読む時にデータはSCKの下降端で行われます。タイミングの詳細については図25-11.をご覧ください。

直列プログラミング動作でのATmega325P/3250Pのプログラミングと照合は次手順が推奨されます(表25-16.の命令一式参照)。

### 1. 電源投入手順:

RESETとSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加します。いくつかのシステムで電源投入中、SCKがLowに保持されるのを書き込み器が保証できません。この場合、SCKがLow(0)に設定されてしまった後、RESETは最低2 CPUクロック周期幅の正パルスを与えられなければなりません。

### 2. 最低20ms待ち、MOSIピンにプログラミング許可命令を送ることによって直列プログラミングを許可してください。

3. 通信が同期を外していると、直列プログラミング命令は動作しません。同期していると、プログラミング許可命令の第3バイト送出時に第2バイト(\$53)が戻ります。この戻りが正しいかどうかによらず、命令の4バイト全てが送信されなければなりません。\$53が戻らない場合、RESETに正パルスを与え、新規プログラミング許可命令を行ってください。

4. フラッシュメモリは1ページ単位で書かれます。ページ容量は175頁の表25-11.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6+1ビットとデータを供給することによって1バイトずつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。プログラムメモリのページはアドレスの上位8ビットを含むページ書き込み命令の設定によって(フラッシュメモリに)格納されます。ポーリング(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低tWD\_FLASH(表25-15.参照)待たなければなりません。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。

### 5. EEPROMはページ単位かバイト単位のどちらかでプログラミングできます。

**バイト単位:** EEPROMは適切なEEPROM書き込み命令と共にアドレスとデータを供給することによって1バイト単位で書かれます。EEPROMのメモリ位置は新規データが書かれるのに先立って始めて自動的に消去されます。ポーリング(BSY/RDY)が使われない場合、次のバイトを行う前に最低tWD\_EEPROM(表25-15.参照)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下位2ビットとデータを供給することによって1バイトずつ設定されます。EEPROMページはアドレスの上位8ビットを含むEEPROMページ書き込み命令によって(EEPROMに)格納されます。EEPROMページアクセス使用時、EEPROMページ設定命令で設定したバイト位置だけが変更されます。残りの位置は無変化で留まります。ポーリング(BSY/RDY)が使われない場合、次ページ(表25-12.参照)を行う前に最低tWD\_EEPROM(表25-15.参照)待たなければなりません。チップ消去されたデバイスでの\$FFデータは書かれる必要がありません。

6. どのメモリ位置も選んだアドレスの内容を直列出力MISOに読み戻す読み出し命令を使うことによって照合できます。

7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。

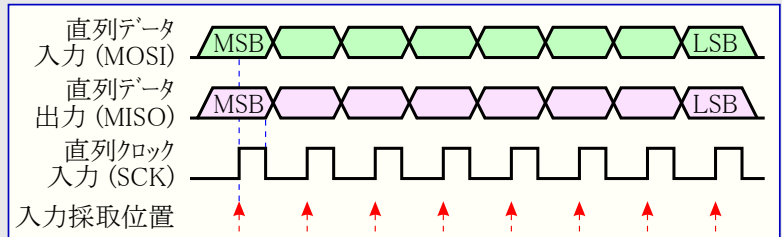
### 8. 電源OFF手順 (必要とされるならば)

- RESETをHigh(1)に設定します。
- VCC電源をOFFにします。

表25-15. ヒューズ、フラッシュ、EEPROMの  
次位置書き込み前の待機時間

シンボル	最低待機時間	備考
tWD_FUSE	4.5ms	ヒューズ書き込み
tWD_FLASH	4.5ms	フラッシュメモリ書き込み
tWD_EEPROM	9.0ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

図25-11. 直列プログラミングバイト通信波形



### 25.8.1. 直列プログラミング命令一式

表25-16.と次ページの図25-12.は命令一式を記述します。

表25-16. 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	
識別バイト読み出し	\$30	\$00	アドレス	識別バイト	
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 (注2)					
フラッシュ ページ書き込み	\$4C	アドレス上位	アドレス下位	\$00	
EEPROMバイト書き込み	\$C0	アドレス上位	アドレス下位	バイト	
EEPROMページ書き込み	\$C2	アドレス上位	アドレス下位	\$00	注: ページ指示以外のアドレスビットは0。
施錠ビット書き込み	\$AC	\$E0	\$00	施錠ビット値	
ヒューズ下位書き込み	\$AC	\$A0	\$00	ヒューズ下位	
ヒューズ上位書き込み	\$AC	\$A8	\$00	ヒューズ上位	
拡張ヒューズ書き込み	\$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: 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: ・施錠ビットとヒューズ値はプログラムが0、非プログラムが1です。将来との互換性のため、未使用のヒューズと施錠ビットは非プログラム(1)にすべきです。

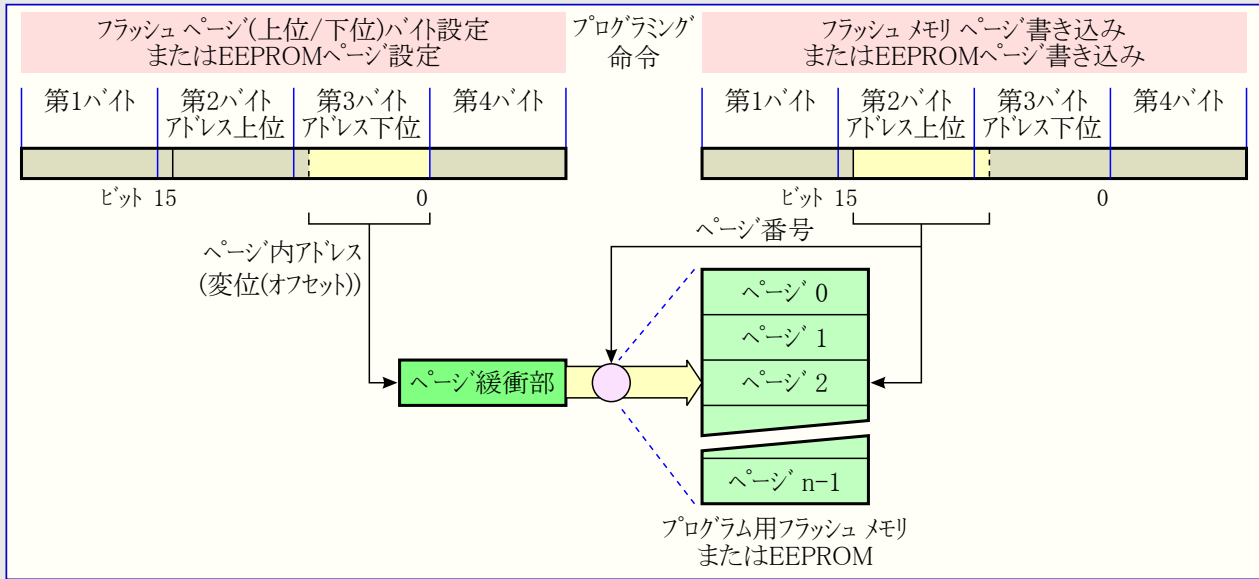
- ・ヒューズ、施錠ビット、識別バイト、校正バイト、ページ容量については対応項目を参照してください。
- ・プログラミングと書き込み器に関する応用記述については <http://www.atmel.com/avr> をご覧ください。
- ・第4バイトの赤背景はホスト読み込み(デバイス出力)を示します。

多忙/準備可検査バイト データ出力のLSBが1なら、プログラミング操作が未だ保留(動作中)です。次の命令が実行される前に本ビットが0に戻るまで待ってください。

同じページ内で、下位バイト データは上位バイト データに先行して格納されなければなりません。

データがページ緩衝部に格納された後にEEPROMページをプログラムしてください。図25-12.をご覧ください。

図25-12. 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。ATmega325P/3250P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	00LL LLLL	ATmega325P/3250P : L=PC5~0
EEPROMページ内バイト設定	0000 0000	0000 00LL	ATmega325P/3250P : L=EEA1~0
フラッシュメモリ読み出し	00HH HHHH	LLLL LLLL	ATmega325P/3250P : H=PC13~8, L=PC7~0
EEPROM読み出し	0000 00HH	LLLL LLLL	ATmega325P/3250P : H=EEA9~8, L=EEA7~0
フラッシュページ書き込み	00HH HHHH	LL00 0000	ATmega325P/3250P : H=PC13~8, L=PC7~6
EEPROMバイト書き込み	0000 00HH	LLLL LLLL	ATmega325P/3250P : H=EEA9~8, L=EEA7~0
EEPROMページ書き込み	0000 00HH	LLLL LL00	ATmega325P/3250P : H=EEA9~8, L=EEA7~2

## 25.8.2. 直列プログラミング特性

SPI部の特性については196頁の「SPIタイミング特性」を参照してください。



## 25.9. JTAGインターフェース経由のプログラミング

JTAGインターフェースを通すプログラミングは4つのJTAG仕様ピン、TCK,TMS,TDI,TDOの制御が必要です。RESETピンとクロックピンの制御は必要とされません。

JTAGインターフェースが使われ得るにはJTAGENヒューズがプログラム(0)されなければなりません。このデバイスは既定でこのヒューズがプログラム(0)されて出荷されます。更にMCU制御レジスタ(MCUCR)のJTAGインターフェース禁止(JTD)ビットが解除(0)されなければなりません。JTDビットが設定(1)の場合、代わりに外部リセットを強制的なLowにできます。その後、2チップ(CPU)クロック後にJTDビットが解除(0)され、JTAGピンはプログラミングに利用できます。これはJTAGインターフェース経由実装書き込みを許すにも拘らず、一方、実行動作で標準ポートピンとしてJTAGピンを使う手段を提供します。境界走査(Boundary-Scan)や内蔵デバッグ機能にJTAGピンを使う時にこの手法が使えないことに注意してください。これらの場合のJTAGピンはその目的専用にされなければなりません。

プログラミング中、TCK入力クロックの周波数はチップの最高周波数よりも低くなければなりません。システムクロック前置分周器はTCKクロック入力を十分に低い周波数へ分周するのには使えません。

この文書内の定義では全ての移動レジスタについて入出力ともLSBが最初に移動されます。

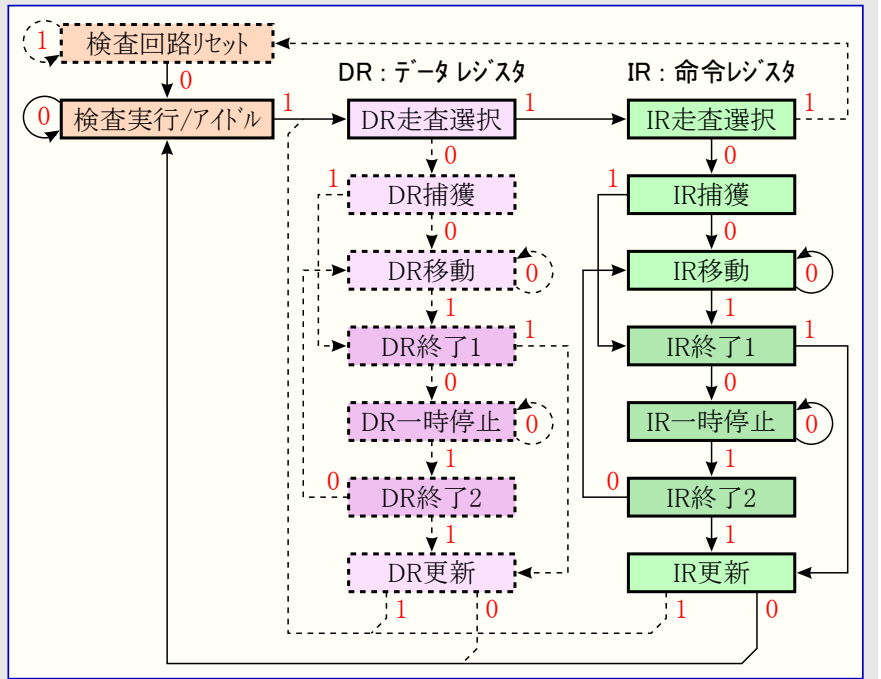
### 25.9.1. プログラミング特殊JTAG命令

命令レジスタ(IR)は4ビット長で、16種類までの命令を支援します。以下に示される一覧はプログラミングに有用なJTAG命令です。

各命令の命令コードは命令名後の16進形式で示されます。本文は各命令についてTDIとTDO間の経路として選ばれるデータレジスタを記述します。

検査入出力ポート(TAP)制御器の検査実行/アイドル状態は内部クロックの発生に使われます。また、JTAG手順間のアイドル状態としても使えます。命令を切り替えるための順次回路の順番は図25-13.で示されます。

図25-13. 命令語変更手順時順次回路状態遷移図



### 25.9.2. AVR\_RESET (\$C)

AVRデバイスをリセット状態へ、またはリセット状態から開放するためのAVR固有公開JTAG命令です。この命令によって検査入出力ポート(TAP)制御器はリセットされません。1ビットのリセットレジスタがデータレジスタ(DR)として選ばれます。リセットチェーンに論理1がある限り、リセットが有効であることに注意してください。このチェーンからの出力はラッチされません。本命令が有効な状態を次に示します。

- DR移動 : リセットレジスタがTCKクロック入力によって移されます。

### 25.9.3. PROG\_ENABLE (\$4)

JTAGポート経由のプログラミングを許可するためのAVR固有公開JTAG命令です。16ビット長のプログラミング許可レジスタがデータレジスタ(DR)として選ばれます。本命令が有効な状態を次に示します。

- DR移動 : プログラミング許可識別子がデータレジスタに移動入力されます。
- DR更新 : プログラミング許可識別子が正規値と比較され、識別子が有効なら、プログラミング動作に移行されます。

### 25.9.4. PROG\_COMMANDS (\$5)

JTAGポート経由のプログラミング命令移行用のAVR固有公開JTAG命令です。15ビット長のプログラミング命令レジスタがデータレジスタ(DR)として選ばれます。本命令が有効な状態を次に示します。

- DR捕獲 : 直前の命令の結果がデータレジスタに設定されます。
- DR移動 : データレジスタがTCK入力により移され、直前の命令の結果を移動出力し、新規命令を移動入力します。
- DR更新 : プログラミング命令がフラッシュメモリ入力に適用されます。
- 検査実行/アイドル : 1つのクロック周期が生成され、適用された命令を実行します(常に必要とされる訳でなく、以降の表25-17.をご覧ください)。

## 25.9.5. PROG\_PAGELOAD (\$6)

JTAGポート経由でフラッシュメモリのページデータを直接設定するためのAVR固有公開JTAG命令です。8ビットフラッシュバイトデータレジスタがデータレジスタ(DR)として選ばれます。これは物理的にプログラミング命令レジスタの下位8ビットです。活性(有効)状態は以下です。

- DR移動 : フラッシュバイトデータレジスタがTCKクロック入力によって移動されます。
- DR更新 : フラッシュバイトデータレジスタの内容が一時レジスタに複写されます。11 TCK周期で開始される書き込み(設定)手順が一時レジスタの内容をフラッシュページ緩衝部に設定します。AVRはPROG\_PAGELOAD命令移行後出会った最初のDR更新に対して下位バイトで始め、新規DR更新状態毎に上位/下位バイト書き込み間を自動的に切り替えます。プログラムカウンタは最初の書き込みバイトを除き、下位バイト書き込み前に予め増加(+1)されます。これは最初のデータがPROG\_COMMANDSによって設定したアドレスに書かれ、ページ緩衝部の最終位置設定がプログラムカウンタを次ページへ増加しないことを保証します。

## 25.9.6. PROG\_PAGEREAD (\$7)

JTAGポート経由でフラッシュメモリ内容を直接捕獲するためのAVR固有公開JTAG命令です。8ビットフラッシュバイトデータレジスタがデータレジスタ(DR)として選ばれます。これは物理的にプログラミング命令レジスタの下位8ビットです。活性(有効)状態は以下です。

- DR捕獲 : 選んだフラッシュメモリバイトの内容がフラッシュバイトデータレジスタに捕獲されます。AVRはPROG\_PAGEREAD命令移行後に出会った最初のDR捕獲に対して下位バイトで始め、新規DR捕獲状態毎に上位/下位バイト読み込み間を自動的に切り替えます。プログラムカウンタは最初の読み込みバイトを含め、各上位バイト読み込み後に増加(+1)されます。これは最初のデータがPROG\_COMMANDSによって設定した先頭アドレスから捕獲され、ページの最終位置読み込みがプログラムカウンタを次ページへ増加(進行)することを保証します。
- DR移動 : フラッシュバイトデータレジスタがTCKクロック入力によって移動されます。

## 25.9.7. データレジスタ

データレジスタ(DR)は186頁の「プログラミング特殊JTAG命令」項で記載されたJTAG命令レジスタ(IR)によって選ばれます。プログラミング操作に関連するデータレジスタを次に示します。

- リセット(Reset)レジスタ
- プログラミング許可(Programming Enable)レジスタ
- フラッシュバイトデータ(Flash Data Byte)レジスタ
- プログラミング命令(Programming Command)レジスタ

## 25.9.8. リセット (Reset) レジスタ

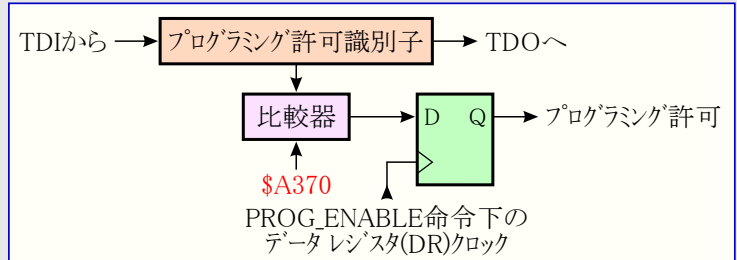
リセットレジスタはプログラミング中、デバイスをリセットするのに使われる検査データレジスタです。プログラミング動作への移行に先立ってデバイスをリセットするのに必要とされます。

リセットレジスタ内の値1は外部RESETをLowに引き込むことに相当します。リセットレジスタ内の値が1である限り、デバイスはリセットされます。リセットレジスタの開放後、デバイスはクロック種別ヒューズ設定に従ってリセット起動遅延時間(18頁の「クロック元」参照)分リセットを維持します。153頁の図23-2.で示されるようにこのデータレジスタからの出力はラッチされず、それ故リセットが直ちに起こります。

## 25.9.9. プログラミング許可 (Programming Enable) レジスタ

プログラミング許可レジスタは16ビット長のレジスタです。このレジスタの内容はプログラミング許可識別子\$A370(1010 0011 0111 0000)と比較されます。このレジスタの内容がプログラミング許可識別子と一致すると、JTAGポート経由のプログラミングが許可されます。このレジスタは電源ONリセットで0にリセットされ、プログラミング動作を抜ける時には常にリセットされるべきです。

図25-14. プログラミング許可 レジスタ



## 25.9.10. プログラミング命令 (Programming Command) レジスタ

プログラミング命令レジスタは15ビット長のレジスタです。このレジスタはプログラミング命令を連続的に移動入力し、直前のプログラミング命令の結果を連続的に移動出力するのに使われます。JTAGプログラミング命令一式は表25-17.で示されます。プログラミング命令を移動入力する時の状態順は図25-16.で図解されます。

図25-15. プログラミング命令 レジスタ

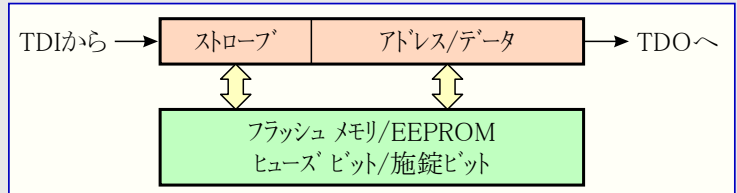


表25-17. JTAGプログラマ命令一式 (手順)

命令			TDI入力	TDO出力	備考
チップ消去	1 ①	チップ消去	010 0011 1000 0000	xxx xxxx xxxx xxxx	
			011 0001 1000 0000	xxx xxxx xxxx xxxx	
			011 0011 1000 0000	xxx xxxx xxxx xxxx	
			011 0011 1000 0000	xxx xxxx xxxx xxxx	
フラッシュメモリ書き込み	1 ②	チップ消去完了検査	011 0011 1000 0000	xxx xxSx xxxx xxxx	(注2)
	2 ①	フラッシュ書き込み移行	010 0011 0001 0000	xxx xxxx xxxx xxxx	
	2 ②	アドレス上位バイト設定	000 0111 HHHH HHHH	xxx xxxx xxxx xxxx	(注10)
	2 ③	アドレス下位バイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	2 ④	データ下位バイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	
	2 ⑤	データ上位バイト設定	001 0111 WWWW WWWW	xxx xxxx xxxx xxxx	
	2 ⑥	データラッチ	011 0111 0000 0000	xxx xxxx xxxx xxxx	(注1)
			111 0111 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
	2 ⑦	ページ書き込み	011 0111 0000 0000	xxx xxxx xxxx xxxx	(注1)
			011 0101 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
	2 ⑧	ページ書き込み完了検査	011 0111 0000 0000	xxx xxSx xxxx xxxx	(注2)
フラッシュ読み出し	3 ①	フラッシュ読み出し移行	010 0011 0000 0010	xxx xxxx xxxx xxxx	
	3 ②	アドレス上位バイト設定	000 0111 HHHH HHHH	xxx xxxx xxxx xxxx	(注10)
	3 ③	アドレス下位バイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	3 ④	データ下位/上位バイト取得	011 0010 0000 0000	xxx xxxx RRRR RRRR	下位バイト
EEPROM書き込み	4 ①	EEPROM書き込み移行	011 0110 0000 0000	xxx xxxx RRRR RRRR	上位バイト
			011 0111 0000 0000	xxx xxxx RRRR RRRR	
			011 0111 0000 0000	xxx xxxx RRRR RRRR	
	4 ②	EEPROM書き込み移行	010 0011 0001 0001	xxx xxxx xxxx xxxx	
	4 ③	アドレス上位バイト設定	000 0111 HHHH HHHH	xxx xxxx xxxx xxxx	(注10)
	4 ④	アドレス下位バイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	4 ⑤	データバイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	
	4 ⑥	データラッチ	011 0111 0000 0000	xxx xxxx xxxx xxxx	(注1)
			111 0111 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
	4 ⑦	EEPROM書き込み	011 0011 0000 0000	xxx xxxx xxxx xxxx	(注1)
			011 0001 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx xxxx xxxx	
	4 ⑧	EEPROM書き込み完了検査	011 0011 0000 0000	xxx xxSx xxxx xxxx	(注2)
EEPROM読み出し	5 ①	EEPROM読み出し移行	010 0011 0000 0011	xxx xxxx xxxx xxxx	
	5 ②	アドレス上位バイト設定	000 0111 HHHH HHHH	xxx xxxx xxxx xxxx	(注10)
	5 ③	アドレス下位バイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	5 ④	EEPROM読み出し	011 0011 LLLL LLLL	xxx xxxx RRRR RRRR	
ヒューズビット書き込み	6 ①	ヒューズ書き込み移行	010 0011 0100 0000	xxx xxxx xxxx xxxx	
	6 ②	データ下位バイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	(注3) ビット配置は173頁の表25-3.参照
	6 ③	拡張ヒューズ書き込み	011 1011 0000 0000	xxx xxxx xxxx xxxx	(注1)
			011 1001 0000 0000	xxx xxxx xxxx xxxx	
			011 1011 0000 0000	xxx xxxx xxxx xxxx	
			011 1011 0000 0000	xxx xxxx xxxx xxxx	
	6 ④	拡張ヒューズ書き込み完了検査	011 1011 0000 0000	xxx xxSx xxxx xxxx	(注2)
	6 ⑤	データ下位バイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	(注3) ビット配置は173頁の表25-4.参照
	6 ⑥	上位側ヒューズ書き込み	011 0111 0000 0000	xxx xxxx xxxx xxxx	(注1)
			011 0101 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
			011 0111 0000 0000	xxx xxxx xxxx xxxx	
	6 ⑦	上位ヒューズ書き込み完了検査	011 0111 0000 0000	xxx xxSx xxxx xxxx	(注2)
	6 ⑧	データ下位バイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	(注3) ビット配置は173頁の表25-5.参照
	6 ⑨	下位側ヒューズ書き込み	011 0011 0000 0000	xxx xxxx xxxx xxxx	(注1)
			011 0001 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx xxxx xxxx	
	6 ⑩	下位ヒューズ書き込み完了検査	011 0011 0000 0000	xxx xxSx xxxx xxxx	(注2)

次頁へ続く

表25-17 (続き). JTAGプログラミング命令一式

命令		TDI入力	TDO出力	備考
施錠ビット書き込み	7 ① 施錠ビット書き込み移行	010 0011 0010 0000	xxx xxxx xxxx xxxx	
	7 ② データ下位バイト設定	001 0011 11WW WWWW	xxx xxxx xxxx xxxx	(注4) ビット配置は172頁の表25-1.参照
	7 ③ 施錠ビット書き込み	011 0011 0000 0000	xxx xxxx xxxx xxxx	(注1)
		011 0001 0000 0000	xxx xxxx xxxx xxxx	
		011 0011 0000 0000	xxx xxxx xxxx xxxx	
		011 0011 0000 0000	xxx xxxx xxxx xxxx	
	7 ④ 施錠ビット書き込み完了検査	011 0011 0000 0000	xxx xxSx xxxx xxxx	(注2)
ヒューズ/施錠ビット読み出し	8 ① ヒューズ/施錠ビット読み出し移行	010 0011 0000 0100	xxx xxxx xxxx xxxx	
	8 ② 拡張ヒューズ読み出し	011 1010 0000 0000	xxx xxxx xxxx xxxx	(注5) ビット配置は173頁の表25-3.参照
		011 1011 0000 0000	xxx xxxx RRRR RRRR	
	8 ③ 上位側ヒューズ読み出し	011 1110 0000 0000	xxx xxxx xxxx xxxx	(注5) ビット配置は173頁の表25-4.参照
		011 1111 0000 0000	xxx xxxx RRRR RRRR	
	8 ④ 下位側ヒューズ読み出し	011 0010 0000 0000	xxx xxxx xxxx xxxx	(注5) ビット配置は173頁の表25-5.参照
		011 0011 0000 0000	xxx xxxx RRRR RRRR	
	8 ⑤ 施錠ビット読み出し	011 0110 0000 0000	xxx xxxx xxxx xxxx	(注5) ビット配置は172頁の表25-1.参照
		011 0111 0000 0000	xxx xxxx xxRR RRRR	
	8 ⑥ ヒューズ/施錠ビット読み出し	011 1010 0000 0000	xxx xxxx xxxx xxxx	(注5)
		011 1110 0000 0000	xxx xxxx RRRR RRRR	
		011 0010 0000 0000	xxx xxxx RRRR RRRR	
		011 0110 0000 0000	xxx xxxx RRRR RRRR	
		011 0111 0000 0000	xxx xxxx RRRR RRRR	
識票バイト	9 ① 識票バイト読み出し移行	010 0011 0000 1000	xxx xxxx xxxx xxxx	
	9 ② アドレスバイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	9 ③ 識票バイト読み出し	011 0010 0000 0000	xxx xxxx xxxx xxxx	
		011 0011 0000 0000	xxx xxxx RRRR RRRR	
校正バイト	10 ① 校正バイト読み出し移行	010 0011 0000 1000	xxx xxxx xxxx xxxx	
	10 ② アドレスバイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	10 ③ 校正バイト読み出し	011 0110 0000 0000	xxx xxxx xxxx xxxx	
		011 0111 0000 0000	xxx xxxx RRRR RRRR	
	11 ① 無操作設定	010 0011 0000 0000	xxx xxxx xxxx xxxx	
		011 0011 0000 0000	xxx xxxx xxxx xxxx	

H = アドレス上位バイトのビット

L = アドレス下位バイトのビット

S = 動作完了フラグ

R = 読み出しデータ (MCU出力)

W = 書き込みデータ (MCU入力)

x = 0か1 (無効/無意味)

注1 : この命令は直前の命令手順によって上位7ビットが正しく設定されている(通常)の場合、必要ではありません。

注2 : Sが1になるまで繰り返します。(待機)

注3 : 設定値0でプログラム、1で非プログラムです。

注4 : 設定値0でプログラム、1では施錠ビットが変化しません。

注5 : 読み出し値0でプログラム、1で非プログラムです。

(訳注) 原書での注6～9は表内備考欄に直接記載しました。

注10 : PCMSB(表25-11.参照)とEEAMSB(表25-12.参照)を越えるアドレスビットは無効です。

注11 : 全てのTDIとTDO手順は2進数で示されています。



### 25.9.11. フラッシュ バイト データ (Flash Data Byte) レジスタ

フラッシュ バイト データ レジスタはページ書き込み実行前のフラッシュ メモリ ページ緩衝部全体設定、またはフラッシュ メモリ内容の読み出しや照合の効率的な方法を提供します。順次回路はフラッシュ メモリへの制御信号を設定し、フラッシュ メモリからのスロープ信号を判断し、従って移動入出力されるためのデータ語だけがが必要です。

実際のフラッシュ バイト データ レジスタは8ビットの走査チェーンと8ビットの一時レジスタから成ります。ページ設定中、**DR更新**状態は走査チェーンによる内容を一時レジスタへ複製し、11 TCK周期内で一時レジスタの内容をフラッシュ メモリ ページ緩衝部へ設定する書き込み(設定)手順を始めます。AVRは**PROG\_PAG ELOAD**命令移行後に出会った最初の**DR更新**に対して下位バイトで始め、新規**DR更新**毎に上位/下位バイト書き込み間を自動的に切り替えます。プログラム カウンタは最初の書き込みバイトを除き、下位バイト書き込み前に予め増加(+1)されます。これは**PROG\_COMMANDS**によって設定したアドレスに最初のデータが書かれ、ページ緩衝部の最終位置設定がプログラム カウンタを次ページに増加しないことを保証します。

ページ読み込み中、選んだフラッシュ メモリのバイトの内容は**DR捕獲**状態中にフラッシュ バイト データ レジスタ内に捕獲されます。AVRは**PROG\_PAGEREAD**命令移行後に出会った最初の**DR捕獲**に対して下位バイトで始め、新規**DR捕獲**毎に上位/下位バイト読み込み間を自動的に切り替えます。プログラム カウンタは最初の読み込みバイトを含め、各上位バイト読み込み後に増加(+1)されます。これは最初のデータが**PROG\_COMMANDS**によって設定した先頭アドレスから捕獲され、ページの最終位置読み込みがプログラム カウンタを次ページに増加(進行)することを保証します。

図25-16. データ語変更/読み出し手順時順次回路状態遷移図

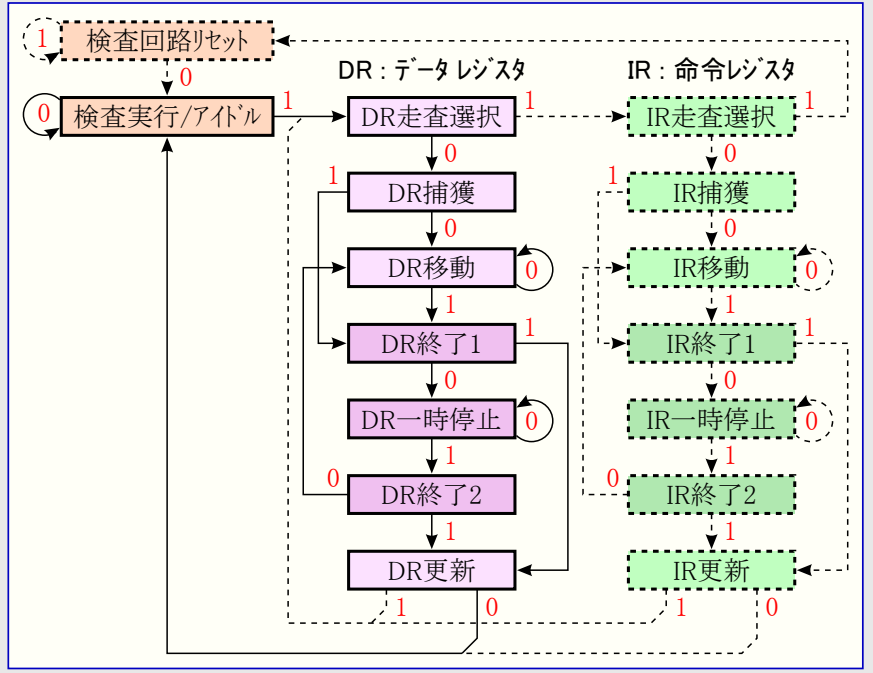
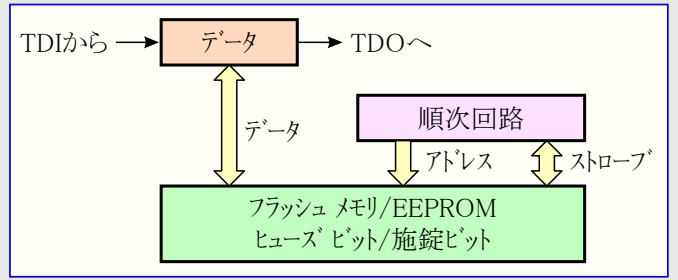


図25-17. フラッシュ バイト データ レジスタ



フラッシュ バイト データ レジスタを制御する順次回路はTCKによってクロック駆動されます。通常動作中内でフラッシュ メモリの各バイトに対して8ビットが移動され、TAP制御器を通して誘導するために必要としたこのクロック周期はフラッシュ バイト データ レジスタに対し、それが使用者にとって明白に操作を完了するための充分なクロック パルス数で順次回路を自動的に送ります。けれどもページ設定中に各**DR更新**状態間で更に数ビットが移動される場合、各**DR更新**状態間の最小11 TCK周期を保証するためのいくつかのTCK周期に対し、TAP制御器は検査実行/アイドル状態に留まるべきです。

### 25.9.12. プログラミング手法

1①、1②形式のような以下の全ての参照は表25-17を参照してください。

### 25.9.13. プログラミング動作への移行

1. JTAG命令**AVR\_RESET**を入力し、リセット レジスタに**1**を移動します。
2. **PROG\_ENABLE**命令を入力し、プログラミング許可レジスタに**\$A370(1010 0011 0111 0000)**を移動します。

### 25.9.14. プログラミング動作からの抜け出し

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. 無操作命令(11①参照)により、全てのプログラミング命令を禁止します。
3. **PROG\_ENABLE**命令を入力し、プログラミング許可レジスタに**\$0000(0000 0000 0000 0000)**を移動します。
4. JTAG命令**AVR\_RESET**を入力し、リセット レジスタに**0**を移動します。

### 25.9.15. チップ消去の実行

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. チップ消去命令(1①参照)を使い、チップ消去を開始します。
3. チップ消去完了検査(1②参照)を使い、完了までポーリングするか、**tWLRH\_CE**(182頁の表25-13.参照)時間待ちます。



## 25.9.16. フラッシュ メモリの書き込み

フラッシュ メモリへ書き込む前にチップ消去が実行されなければなりません。190頁の「チップ消去の実行」をご覧ください。

1. JTAG 命令 **PROG\_COMMANDS** を入力します。
2. フラッシュ書き込み移行命令(2①参照)を使い、フラッシュ メモリ書き込みを許可します。
3. アドレス上位設定命令(2②参照)を使い、書き込みアドレス上位バイトを設定します。
4. アドレス下位設定命令(2③参照)を使い、書き込みアドレス下位バイトを設定します。
5. データ設定命令(2④, 2⑤, 2⑥参照)を使い、書き込みデータを設定します。
6. ページ内の全語(ワード)数分4.と5.を繰り返します。
7. ページ書き込み命令(2⑦参照)を使い、ページをフラッシュ メモリに書き込みます。
8. ページ書き込み完了検査(2⑧参照)を使い、完了までポーリングするか、**tWLRH\_FLASH**(182頁の表25-13.参照)時間待ちます。
9. 全データ書き込みまで3.~8.を繰り返します。

より効率的なデータ転送は**PROG\_PAGELOAD** 命令を使うことで達せられます。

1. JTAG 命令 **PROG\_COMMANDS** を入力します。
2. フラッシュ書き込み移行命令(2①参照)を使い、フラッシュ メモリ書き込みを許可します。
3. アドレス設定命令(2②, 2③参照)を使い、ページ アドレスを設定します。ページ内アドレスには**PCWORD**(175頁の表25-11.参照)が使われ、これらのビットは0として書かれなければなりません。
4. JTAG 命令 **PROG\_PAGELOAD** を入力します。
5. 先頭命令のLSBから始まって最終命令のMSBで終わるページ内にバイト単位で全命令語を移動入力することによってページ全体を設定します。フラッシュ バイト データレジスタの内容をフラッシュ ページ位置に複写し、各新規語(ワード)前にプログラム カウンタを自動増加するの**にDR更新**を使います。
6. JTAG 命令 **PROG\_COMMANDS** を入力します。
7. ページ書き込み命令(2⑦参照)を使い、ページをフラッシュ メモリに書き込みます。
8. ページ書き込み完了検査(2⑧参照)を使い、完了までポーリングするか、**tWLRH\_FLASH**(182頁の表25-13.参照)時間待ちます。
9. 全データ書き込みまで3.~8.を繰り返します。

## 25.9.17. フラッシュ メモリの読み出し

1. JTAG 命令 **PROG\_COMMANDS** を入力します。
2. フラッシュ読み出し移行命令(3①参照)を使い、フラッシュ メモリ読み出しを許可します。
3. アドレス設定命令(3②, 3③参照)を使い、読み出しアドレスを設定します。
4. データ読み出し命令(3④参照)を使い、データを読み出します。
5. 全データ読み出しまで3.~4.を繰り返します。

より効率的なデータ転送は**PROG\_PAGEREAD** 命令を使うことで達せられます。

1. JTAG 命令 **PROG\_COMMANDS** を入力します。
2. フラッシュ読み出し移行命令(3①参照)を使い、フラッシュ メモリ読み出しを許可します。
3. アドレス設定命令(3②, 3③参照)を使い、ページ アドレスを設定します。ページ内アドレスには**PCWORD**(175頁の表25-11.参照)が使われ、これらのビットは0として書かれなければなりません。
4. JTAG 命令 **PROG\_PAGEREAD** を入力します。
5. 先頭命令のLSBから始まって最終命令のMSBで終わるページ(またはフラッシュ)内で全命令語を移動出力することで、ページ(またはフラッシュ)全体を読みます。**DR捕獲**状態はフラッシュ メモリからのデータ捕獲と各語(ワード)が読まれた後にプログラム カウンタも自動増加します。**DR移動**に先立って**DR捕獲**が起きることに注意してください。従って移動出力される先頭バイトは有効なデータを含みます。
6. JTAG 命令 **PROG\_COMMANDS** を入力します。
7. 全データ読み出しまで3.~6.を繰り返します。

## 25.9.18. EEPROMの書き込み

EEPROMへ書き込む前にチップ消去が実行されなければなりません。190頁の「チップ消去の実行」をご覧ください。

1. JTAG 命令 **PROG\_COMMANDS** を入力します。
2. EEPROM書き込み移行命令(4①参照)を使い、EEPROMメモリ書き込みを許可します。
3. アドレス上位設定命令(4②参照)を使い、書き込みアドレス上位バイトを設定します。
4. アドレス下位設定命令(4③参照)を使い、書き込みアドレス下位バイトを設定します。
5. データ設定命令(4④, 4⑤参照)を使い、書き込みデータを設定します。
6. ページ内の全バイト数分4.と5.を繰り返します。
7. EEPROM書き込み命令(4⑥参照)を使い、データをEEPROMメモリに書き込みます。
8. EEPROM書き込み完了検査(4⑦参照)を使い、完了までポーリングするか、**tWLRH**(182頁の表25-13.参照)時間待ちます。
9. 全データ書き込みまで3.~8.を繰り返します。

**PROG\_PAGELOAD** 命令がEEPROM書き込み時に使えないことに注意してください。

### 25.9.19. EEPROMの読み出し

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. EEPROM読み出し移行命令(5①参照)を使い、EEPROMメモリ読み出しを許可します。
3. アドレス設定命令(5②, 5③参照)を使い、読み出しアドレスを設定します。
4. データ読み出し命令(5④参照)を使い、データを読み出します。
5. 全データが読み出されてしまうまで3.~4.を繰り返します。

**PROG\_PAGEREAD**命令がEEPROM読み出し時に使えないことに注意してください。

### 25.9.20. ヒューズビットの書き込み (訳注:本項は修正してあります。)

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. ヒューズ書き込み移行命令(6①参照)を使い、ヒューズ書き込みを許可します。
3. 拡張データ設定命令(6②参照)を使い、拡張ヒューズ値を設定します。(0=プログラム、1=非プログラム)
4. 拡張ヒューズ書き込み命令(6③参照)を使い、拡張ヒューズを書き込みます。
5. ヒューズ書き込み完了検査(6④参照)を使い、完了までポーリングするか、tWLRH(182頁の表25-13.参照)時間待ちます。
6. 上位データ設定命令(6⑤参照)を使い、上位側ヒューズ値を設定します。(0=プログラム、1=非プログラム)
7. 上位ヒューズ書き込み命令(6⑥参照)を使い、上位側ヒューズを書き込みます。
8. ヒューズ書き込み完了検査(6⑦参照)を使い、完了までポーリングするか、tWLRH(182頁の表25-13.参照)時間待ちます。
9. 下位データ設定命令(6⑧参照)を使い、下位側ヒューズ値を設定します。(0=プログラム、1=非プログラム)
10. 下位ヒューズ書き込み命令(6⑨参照)を使い、下位側ヒューズを書き込みます。
11. ヒューズ書き込み完了検査(6⑩参照)を使い、完了までポーリングするか、tWLRH(182頁の表25-13.参照)時間待ちます。

### 25.9.21. 施錠ビットの書き込み

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. 施錠ビット書き込み移行命令(7①参照)を使い、施錠ビット書き込みを許可します。
3. データ設定命令(7②参照)を使い、施錠ビット値を設定します。(0=プログラム、1=無変化)
4. 施錠ビット書き込み命令(7③参照)を使い、施錠ビットに書き込みます。
5. 施錠ビット書き込み完了検査(7④参照)を使い、完了までポーリングするか、tWLRH(182頁の表25-13.参照)時間待ちます。

### 25.9.22. ヒューズ/施錠ビットの読み出し (訳注:本項は修正してあります。)

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. ヒューズ/施錠ビット読み出し移行命令(8①参照)を使い、ヒューズ/施錠ビット読み出しを許可します。
3. 全てのヒューズと施錠ビットを読むにはヒューズ/施錠ビット読み出し命令(8⑥参照)を使います。  
拡張ヒューズビットのみを読み出す場合は拡張ヒューズ読み出し命令(8②参照)を使います。  
上位側ヒューズビットだけを読むには上位ヒューズ読み出し命令(8③参照)を使います。  
下位側ヒューズビットだけを読むには下位ヒューズ読み出し命令(8④参照)を使います。  
施錠ビットだけを読むには施錠ビット読み出し命令(8⑤参照)を使います。

### 25.9.23. 識票バイトの読み出し

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. 識票読み出し移行命令(9①参照)を使い、識票読み出しを許可します。
3. アドレス設定命令(9②参照)を使い、読み出しアドレス\$00を設定します。
4. データ読み出し命令(9③参照)を使い、識票バイトを読み出します。
5. 第2、第3バイトを読むためにアドレスを\$01,\$02として各々3.~4.を繰り返します。

### 25.9.24. 校正バイトの読み出し

1. JTAG命令**PROG\_COMMANDS**を入力します。
2. 校正バイト読み出し移行命令(10①参照)を使い、校正バイト読み出しを許可します。
3. アドレス設定命令(10②参照)を使い、読み出しアドレス\$00を設定します。
4. データ読み出し命令(10③参照)を使い、校正バイトを読み出します。

## 26. 電気的特性

### 26.1. 絶対最大定格 (警告)

動作温度	-55℃ ~ +125℃
保存温度	-65℃ ~ +150℃
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。

### 26.2. DC特性

TA=-40℃~85℃, VCC=1.8V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V <sub>IL</sub>	Lowレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	-0.5		0.2VCC (注1)	V
		VCC=2.4~5.5V	-0.5		0.3VCC (注1)	
V <sub>IL1</sub>	Lowレベル入力電圧 (XTAL1)	VCC=1.8~5.5V	-0.5		0.1VCC (注1)	
V <sub>IH</sub>	Highレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	0.7VCC (注2)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注2)		VCC+0.5	
V <sub>IH1</sub>	Highレベル入力電圧 (XTAL1)	VCC=1.8~2.4V	0.8VCC (注2)		VCC+0.5	
		VCC=2.4~5.5V	0.7VCC (注2)		VCC+0.5	
V <sub>IH2</sub>	Highレベル入力電圧 (RESET)	VCC=1.8~5.5V	0.85VCC (注2)		VCC+0.5	
V <sub>OL</sub>	Lレベル出力電圧 (注3)	ポートB以外 IOL=10mA, VCC=5V			0.9	
		IOL=5mA, VCC=3V			0.6	
V <sub>OL1</sub>		ポートB IOL=20mA, VCC=5V			0.9	
		IOL=10mA, VCC=3V			0.6	
V <sub>OH</sub>	Hレベル出力電圧 (注4)	ポートB以外 IOH=-10mA, VCC=5V	4.2			μA
		IOH=-5mA, VCC=3V	2.3			
V <sub>OH1</sub>		ポートB IOH=-20mA, VCC=5V	4.2			
		IOH=-10mA, VCC=3V	2.3			
I <sub>IL</sub>	I/OピンLowレベル入力漏れ電流	VCC=5.5V			1	
I <sub>IH</sub>	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			1	
R <sub>RST</sub>	RESETピンプルアップ抵抗		20		100	kΩ
R <sub>PU</sub>	I/Oピンプルアップ抵抗		20		100	
I <sub>CC</sub>	活動動作消費電流 (電力削減レジスタ(PRR)全ビット=1)	VCC=2V, 1MHz		0.5	0.75	mA
		VCC=3V, 4MHz		2.6	3.5	
		VCC=5V, 8MHz		9	12	
	アイドル動作消費電流 (電力削減レジスタ(PRR)全ビット=1)	VCC=2V, 1MHz		0.14	0.25	
		VCC=3V, 4MHz		0.75	1.5	
		VCC=5V, 8MHz		2.9	5	
	パワーセーブ動作消費電流	32kHz VCC=1.8V		0.75		μA
		TOSC許可 VCC=3V		1		
	パワーダウン動作消費電流 (注5)	VCC=3V, WDT有効		6.7	15	
		VCC=3V, WDT禁止		0.2	2	
V <sub>ACIO</sub>	アナログ比較器入力変位(オフセット)電圧	VCC<3.6V, Vin<0.5V		<15	60 (注6)	mV
		VCC>3.6V, Vin<0.5V		<15	500 (注6)	
I <sub>ACLK</sub>	アナログ比較器入力漏れ電流	VCC=5V, Vin=VCC/2	-50		50	nA
t <sub>ACPD</sub>	アナログ比較器伝播遅延時間	VCC=2.7V		750		ns
		VCC=4.0V		500		

(注1)~(注5)は次頁を参照してください。

**注1:** Lowレベルの認識が保証される最高電圧です。

**注2:** Highレベルの認識が保証される最低電圧です。

**注3:** 各I/Oポートは安定状態(非過渡時)に於いて検査条件(ポートBはVCC=3Vで10mA, VCC=5Vで20mA、他のポートはVCC=3Vで5mA, VCC=5Vで10mA)よりも多くの吸い込み電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全ポートのIOLの合計が400mAを超えるべきではありません。
2. ポートA7~0、C7~4、G2のIOLの合計が100mAを超えるべきではありません。
3. ポートB7~0、E7~0、G5~3のIOLの合計が100mAを超えるべきではありません。
4. ポートC3~0、D7~0、G1~0のIOLの合計が100mAを超えるべきではありません。
5. ポートF7~0のIOLの合計が100mAを超えるべきではありません。

IOLが検査条件を超える場合、VOLも仕様書での値を超えます。表の検査条件よりも大きな吸い込み電流を流すことは保証されません。

**注4:** 各I/Oポートは安定状態(非過渡時)に於いて検査条件(ポートBはVCC=3Vで10mA, VCC=5Vで20mA、他のポートはVCC=3Vで5mA, VCC=5Vで10mA)よりも多くの吐き出し電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全ポートのIOHの合計が400mAを超えるべきではありません。
2. ポートA7~0、C7~4、G2のIOHの合計が100mAを超えるべきではありません。
3. ポートB7~0、E7~0、G5~3のIOHの合計が100mAを超えるべきではありません。
4. ポートC3~0、D7~0、G1~0のIOHの合計が100mAを超えるべきではありません。
5. ポートF7~0のIOHの合計が100mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件よりも大きな吐き出し電流を流すことは保証されません。

**注5:** 代表値は25°Cでの値です。

**注6:** これらの値は特性付けに基づきます。従って、製品での上限は保証できません。

## 26.3. 速度勾配

図26-1. ATmega325PV/3250PVの最高周波数対VCC

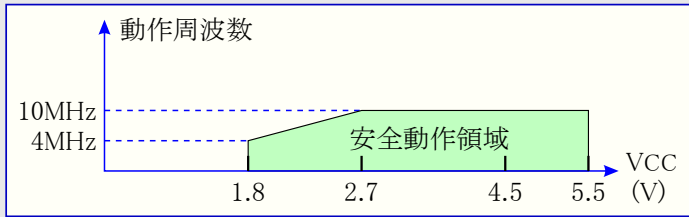
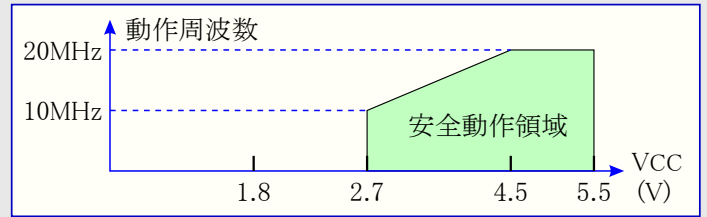


図26-2. ATmega325P/3250Pの最高周波数対VCC



## 26.4. クロック特性

表26-1. 校正付き内蔵RC発振器の校正精度

校正種別	周波数	VCC	温度	校正精度
工場校正	8.0MHz	3V	25°C	±10%
使用者校正	7.3~8.1MHz	1.8~5.5V(注1), 2.7~5.5V(注2)	-40~85°C	±1%

**注1:** ATmega325PV/3250PVに対する電圧範囲

**注2:** ATmega325P/3250Pに対する電圧範囲

図26-3. 外部クロック駆動波形

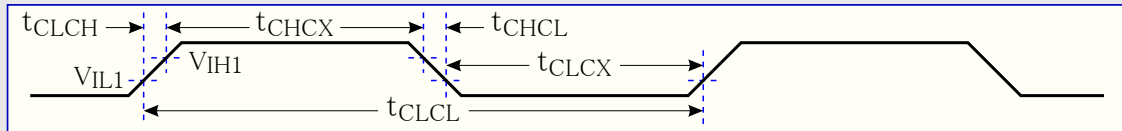


表26-2. 外部クロック特性

シンボル	項目	VCC=1.8~5.5V		VCC=2.7~5.5V		VCC=4.5~5.5V		単位
		最小	最大	最小	最大	最小	最大	
1/t <sub>CLCL</sub>	クロック周波数	0	4	0	10	0	20	MHz
t <sub>CLCL</sub>	クロック周期	250		100		50		ns
t <sub>CHCX</sub>	Highレベル時間	100		40		20		
t <sub>CLCX</sub>	Lowレベル時間	100		40		20		
t <sub>CLCH</sub>	上昇時間		2.0		1.6		0.5	μs
t <sub>CHCL</sub>	下降時間		2.0		1.6		0.5	
Δt <sub>CLCL</sub>	隣接クロック周期間の変化率		2		2		2	%

注: 詳細については19頁の「外部クロック信号」を参照してください。

## 26.5. システムとリセットの特性

表26-3. リセット、低電圧検出(BOD)、内部基準電圧の特性

シンボル	項目	条件	最小	代表	最大	単位
V <sub>RST</sub>	RESETピン閾値電圧		0.2VCC		0.85VCC	V
t <sub>RST</sub>	リセットパルス幅			800		ns
V <sub>HYST</sub>	低電圧検出ヒステリシス電圧			50		mV
t <sub>BOD</sub>	最小低電圧検出時間			2		μs
V <sub>BG</sub>	基準電圧	VCC=2.7V TA=25°C	1.0	1.1	1.2	V
t <sub>BG</sub>	起動時間			40	70	μs
I <sub>BG</sub>	消費電流			15		μA

### 26.5.1. 標準電源ONリセット

この電源ONリセットの実装はATmega325P/3250Pの初期版で存在します。下表はこの電源ONリセットの特性を記述し、以下のデバイスに対してだけ有効です。

- ATmega325P : 改訂Bとそれ以前
- ATmega3250P : 改訂Bとそれ以前

表26-4. 標準電源ONリセット特性 (TA=-40°C~85°C)

シンボル	項目	最小	代表	最大	単位
V <sub>POT</sub>	上昇時電源ONリセット閾値電圧	0.7	1.0	1.4	V
	下降時電源ONリセット閾値電圧 (注1)	0.05	0.9	1.3	
S <sub>RON</sub>	電源ON電圧傾斜率	0.01		4.5	V/ms

注1: 供給電圧がこの電圧以下にならないと、上昇時の電源ONリセットは動作しません。

### 26.5.2. 強化電源ONリセット

この電源ONリセットの実装はATmega325P/3250Pの新版で存在します。下表はこの電源ONリセットの特性を記述し、以下のデバイスに対してだけ有効です。

- ATmega325P : 改訂Cとそれ以降
- ATmega3250P : 改訂Cとそれ以降

表26-5. 強化電源ONリセット特性 (TA=-40°C~85°C)

シンボル	項目	最小	代表	最大	単位
V <sub>POT</sub>	上昇時電源ONリセット閾値電圧	1.1	1.4	1.6	V
	下降時電源ONリセット閾値電圧 (注1)	0.6	1.3	1.6	
S <sub>RON</sub>	電源ON電圧傾斜率	0.01		10	V/ms

注: 値は指針だけです。実際の値はTBDです。

注1: 供給電圧がこの電圧以下にならないと、上昇時の電源ONリセットは動作しません。



### 26.5.3. 低電圧検出 (BOD)

表26-6. BODLEVELヒューズ<sup>®</sup> (VBOT) 設定 (注1)

BODLEVEL1,0	最小	代表	最大	単位
1 1	低電圧検出(BOD)リセット禁止			
1 0	1.7	1.8	2.0	V
0 1	2.5	2.7	2.9	
0 0	4.1	4.3	4.5	

注1: いくつかのデバイスでVBOTが公称最低動作電圧以下の可能性があります。この状態のデバイスについては、製造検査中、VCC=VBOTに落として検査されます。これはマイクロコントローラの正しい動作がもはや保証されない電圧になる前に、低電圧検出(BOD)リセットが起きることを保証します。この検査はATmega325P/3250PについてBODLEVEL=01を、ATmega325PV/3250PVについてBODLEVEL=10を使って実行されます。

### 26.6. SPIタイミング特性

図26-4. SPI タイミング必要条件 (主装置動作)

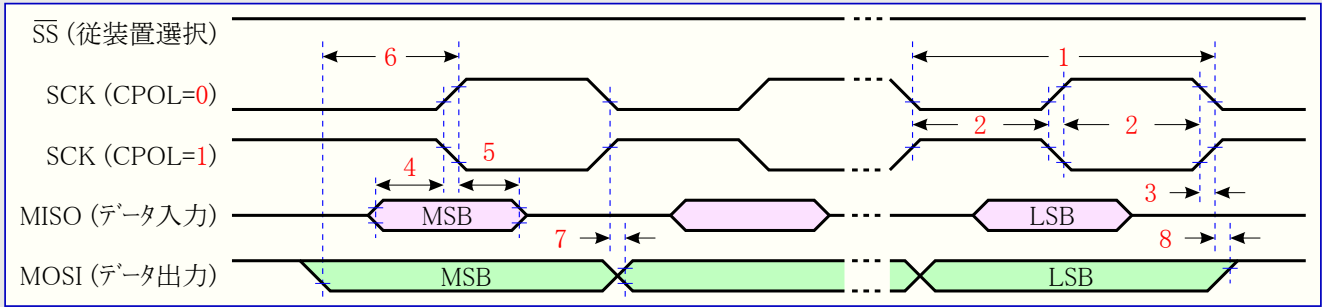


図26-5. SPI タイミング必要条件 (従装置動作)

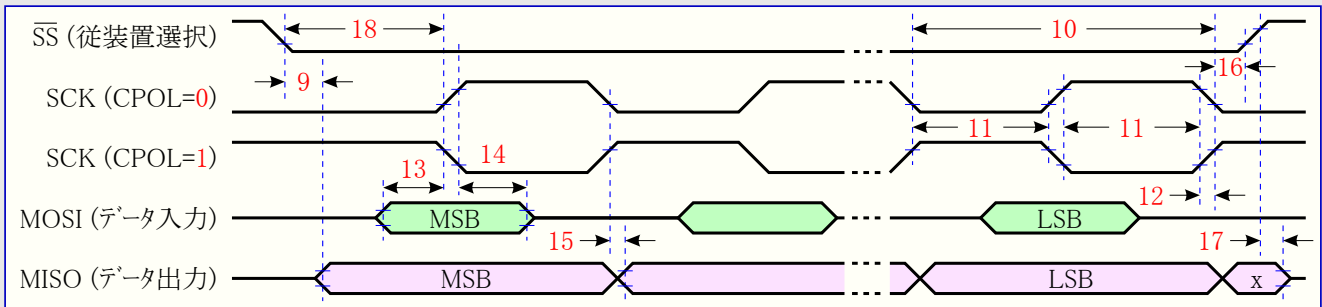


表26-7. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表17-5.参照		ns
2	SCK High/Low期間	主装置		50%デューティ比		
3	SCK上昇/下降時間	主装置		3.6		
4	入力データ 準備時間	主装置		10		
5	入力データ 保持時間	主装置		10		
6	出力からSCK変移時間	主装置		0.5tSCK		
7	SCKからの出力遅延時間	主装置		10		
8	SCKからのHigh出力時間	主装置		10		
9	SS↓からの出力遅延時間	従装置		15		μs
10	SCK周期	従装置	4tCK			
11	SCK High/Low期間 (注)	従装置	2tCK			
12	SCK上昇/下降時間	従装置		1.6		
13	入力データ 準備時間	従装置	10			
14	入力データ 保持時間	従装置	tCK			
15	SCKからの出力遅延時間	従装置		15		ns
16	SCKからのSS↑遅延時間	従装置	20			
17	SS↑からの出力Hi-Z遅延時間	従装置		10		
18	SS↓からのSCK遅延時間	従装置	20tCK			

注: SPIプログラミングでの最小SCK High/Low期間は、2tCLCL(tCK < 12MHz)、3tCLCL(tCK ≥ 12MHz)です。

## 26.7. A/D変換器特性 (暫定)

表26-8. A/D変換特性

シンボル	項目	条件	最小	代表	最大	単位
	分解能	シングル エント' 入力変換		10		ビット
		差動入力変換		8		
	絶対精度 (積分非直線性誤差、 微分非直線性誤差、 量子化誤差、利得誤差、 変位(オフセット)誤差を含む)	シングル エント' 入力変換 VCC=4V VREF=4V	変換クロック=200kHz	2	2.5	LSB
			変換クロック=1MHz	2.5		
			変換クロック=200kHz 雑音低減動作	2		
			変換クロック=1MHz 雑音低減動作	2.5		
	積分非直線性誤差	シングル エント' 入力変換 VCC=4V、VREF=4V 変換クロック=200kHz		1.25		
	微分非直線性誤差			0.25		
	利得誤差			2		
	変位(オフセット)(ゼロ)誤差			1.25		
	変換時間	連続変換動作	13		260	μs
	変換クロック周波数		0.05		1	MHz
AVCC	アナログ供給電圧		VCC-0.3		VCC+0.3	V
VREF	基準電圧	シングル エント' 入力変換	1.0		AVCC	
		差動入力変換	1.0		AVCC-0.5	
VIN	入力電圧	シングル エント' 入力変換	GND		VREF	
		差動入力変換	チャンネル ピン入力	GND	AVCC	
			入力差電圧(注1)	-0.85VREF	VREF	
	入力周波数帯域	シングル エント' 入力変換		38.5		kHz
		差動入力変換		4		
VINT	内蔵基準電圧		1.0	1.1	1.2	V
RREF	基準電圧入力インピーダンス			32		kΩ
RAIN	アナログ入力インピーダンス			100		MΩ

注1: 差動チャンネル間の差電圧(訳補: 最低入力電圧がチャンネルピン入力電圧で規定されることに注意してください。)

## 27. 代表特性

以下の図は代表的な特性を示します。これらの図は製造中に検査されていません。全ての消費電流測定は全I/Oピンを入力として設定した内部プルアップ許可で行われています。電源幅振幅の方形波発振器がクロック源として使われています。

活動動作とアイドル動作の消費電流測定は電力削減レジスタ(PRR)で全ビットが設定(1)で行なわれ、従って関係周辺機能部はOFFにされます。アナログ比較器もこれらの測定中に禁止されます。詳細については26頁の「電力削減レジスタ(PRR)」をご覧ください。

パワーダウン動作での消費電力はクロック選択と無関係です。

消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。支配的な要素は動作電圧と動作周波数です。

容量性負荷のピンの引き込み電流は(1つのピンに対して)  $CL(\text{負荷容量}) \times VCC(\text{動作電圧}) \times f(I/O\text{ピンの平均切り替え周波数})$  として推測できます。

デバイスは検査範囲よりも高い周波数特性を示します。デバイスは注文番号が示す周波数よりも高い周波数での機能特性を保証されません。

ウォッチドッグ タイマ許可のパワーダウン動作での消費電流とウォッチドッグ タイマ禁止のパワーダウン動作での消費電流間の違いは、ウォッチドッグ タイマによって引き込んだ(消費した)差電流を表します。

### 27.1. 活動動作消費電流

図27-1. 活動動作消費電流 対 周波数 (100kHz~1MHz)

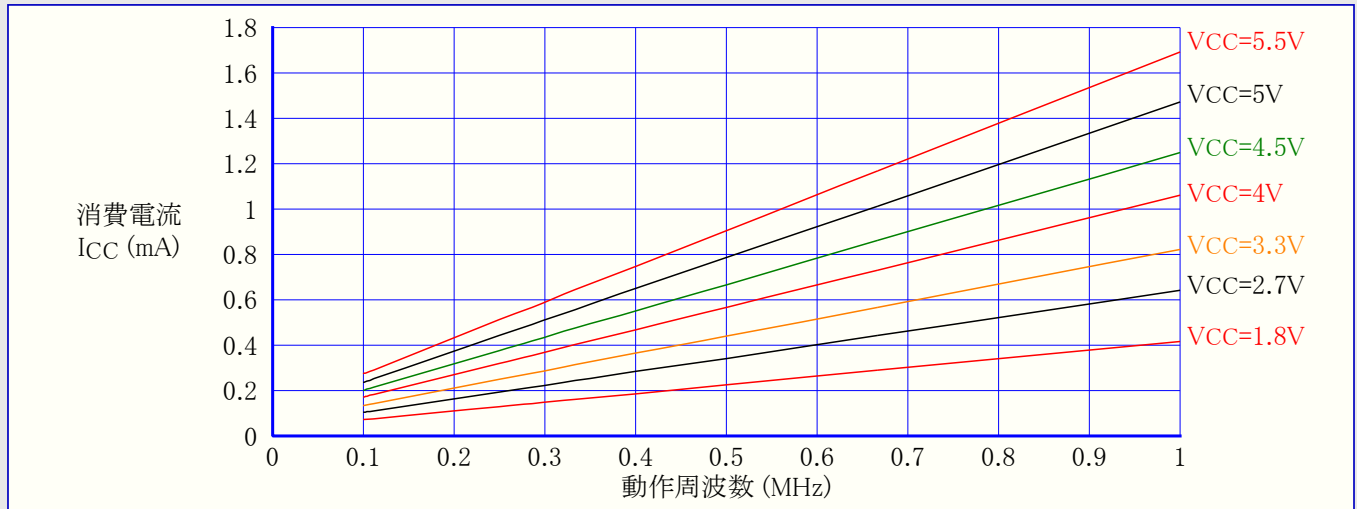


図27-2. 活動動作消費電流 対 周波数 (1MHz~20MHz)

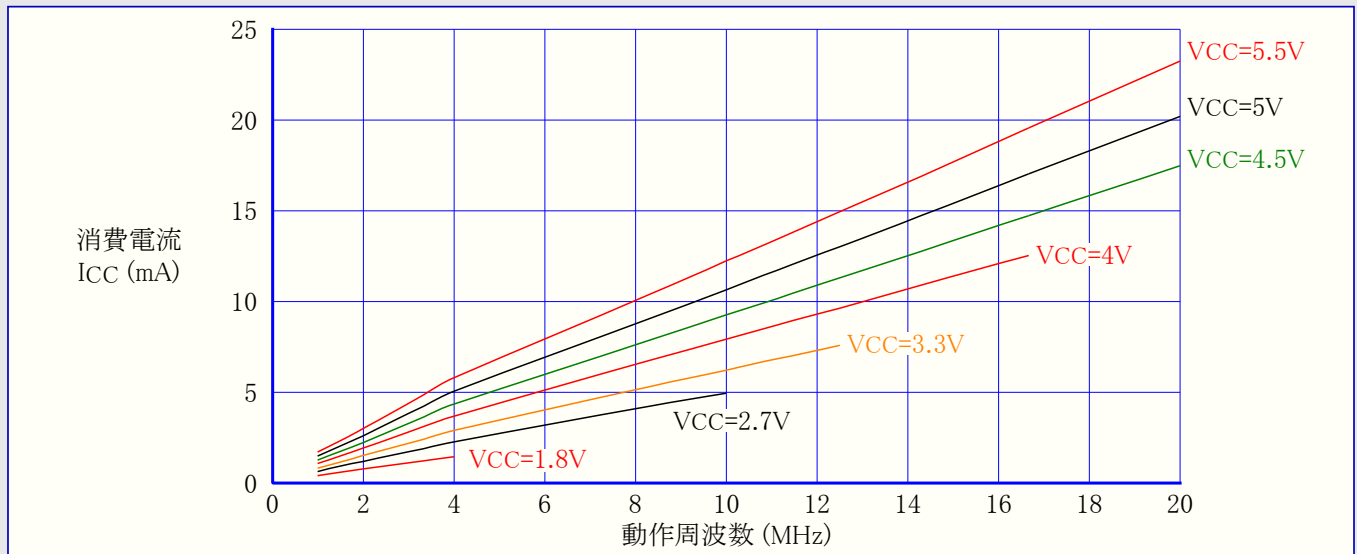


図27-3. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

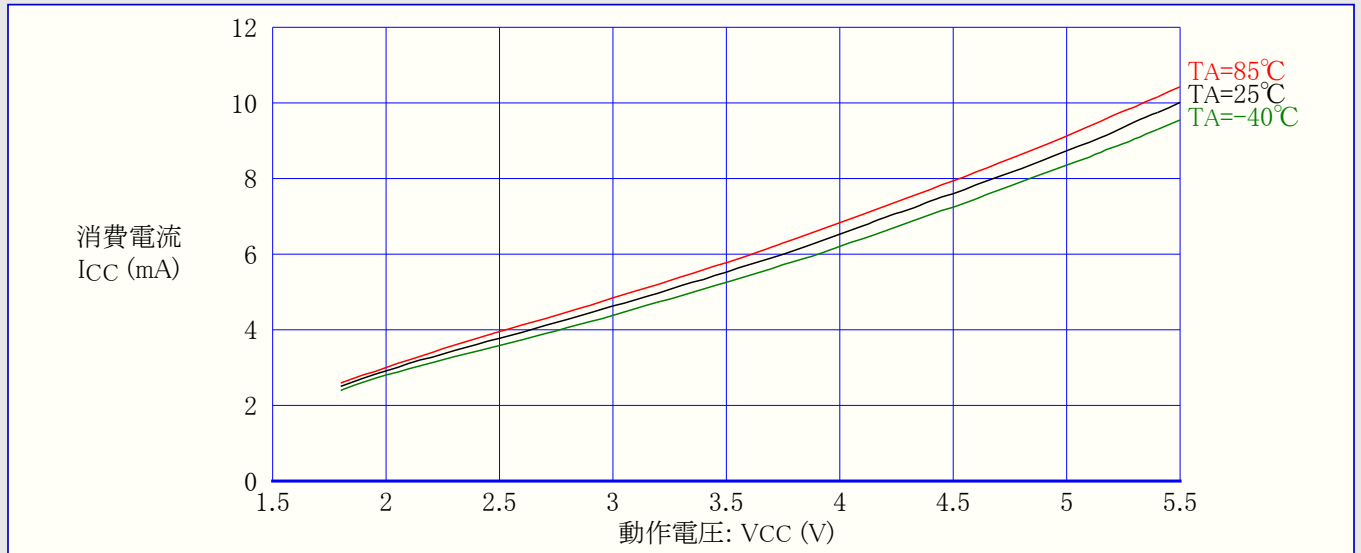
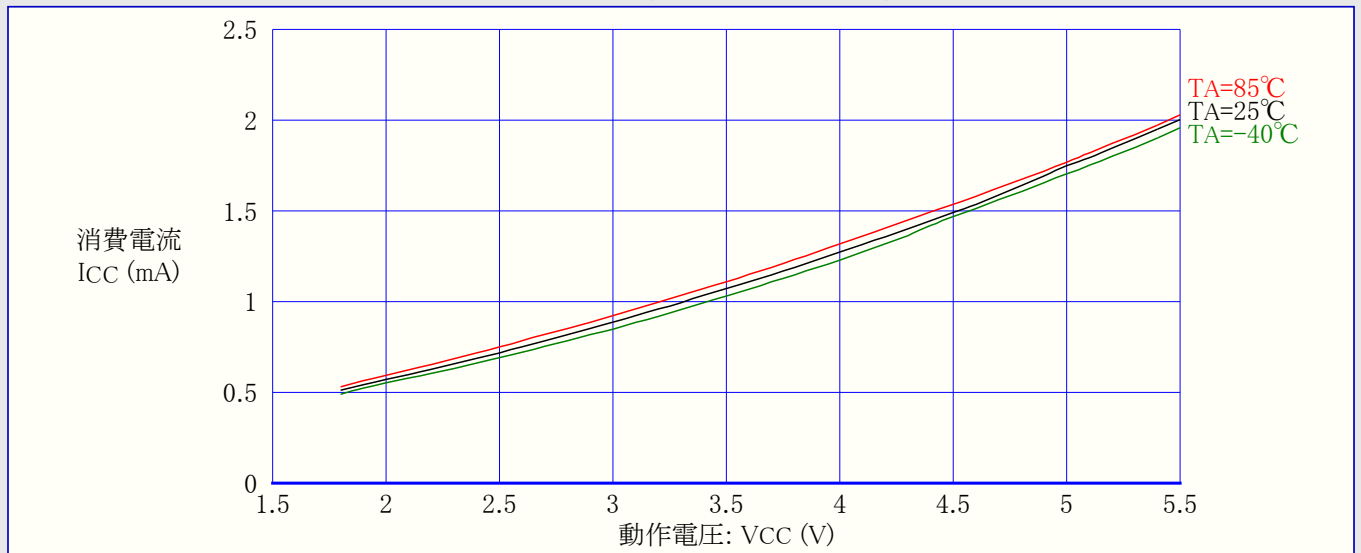


図27-4. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)



## 27.2. アイドル動作消費電流

図27-5. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

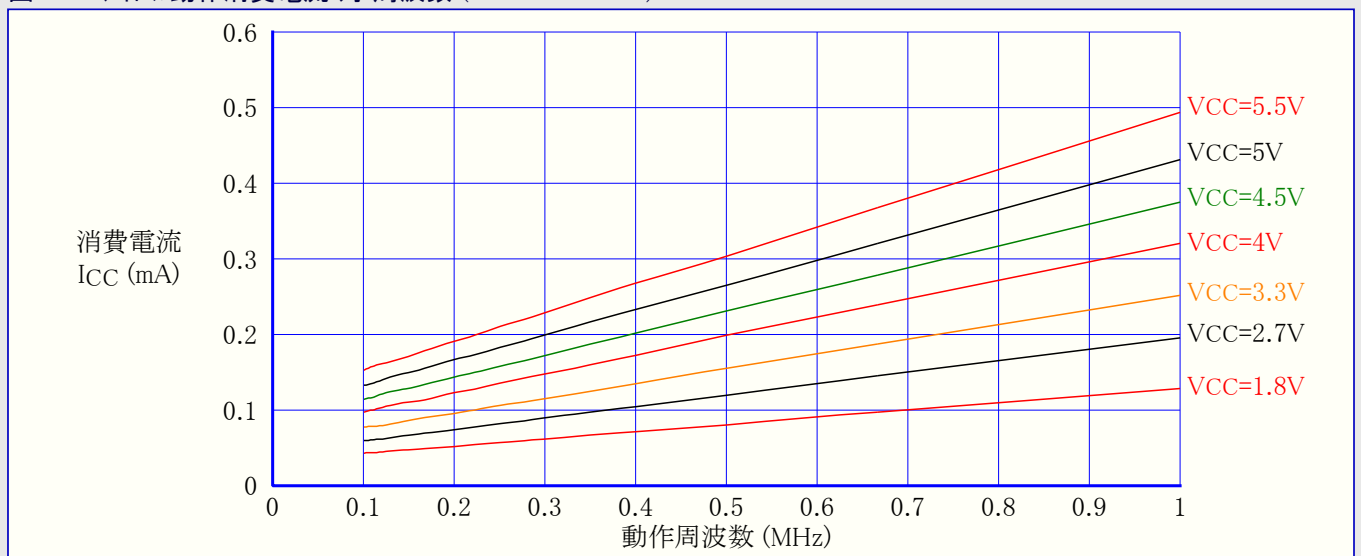


図27-6. アイドル動作消費電流 対 周波数 (1MHz~20MHz)

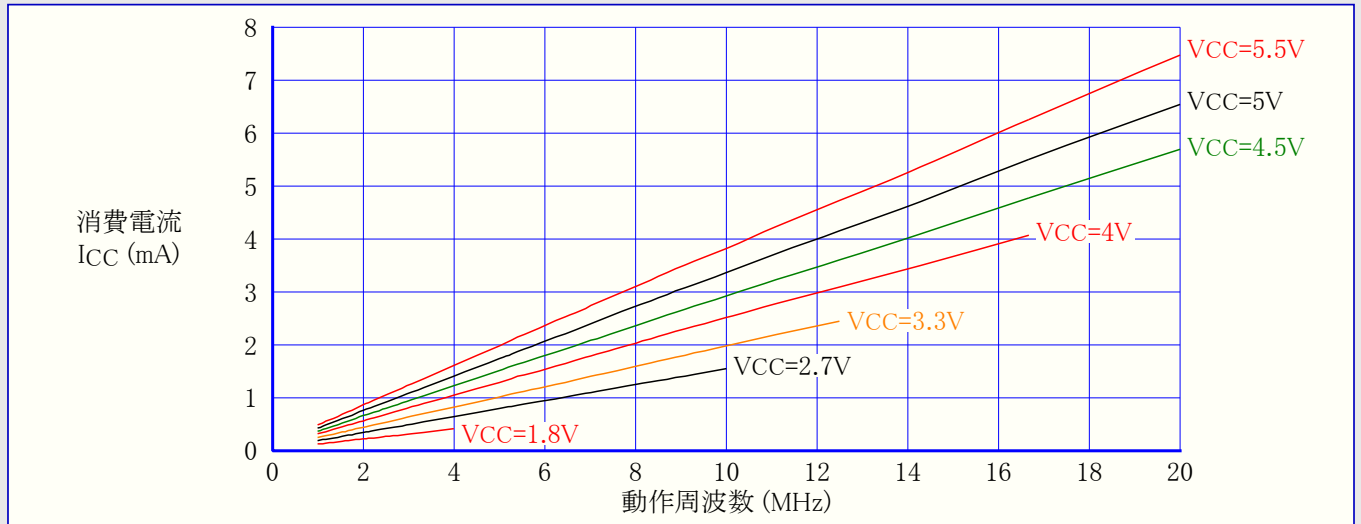


図27-7. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

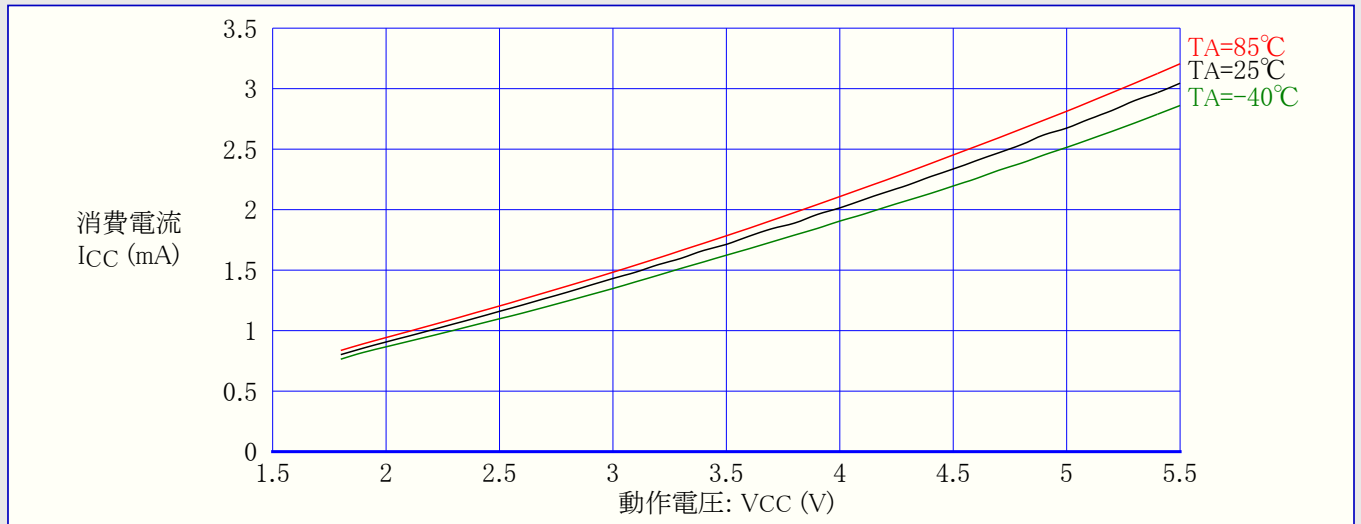
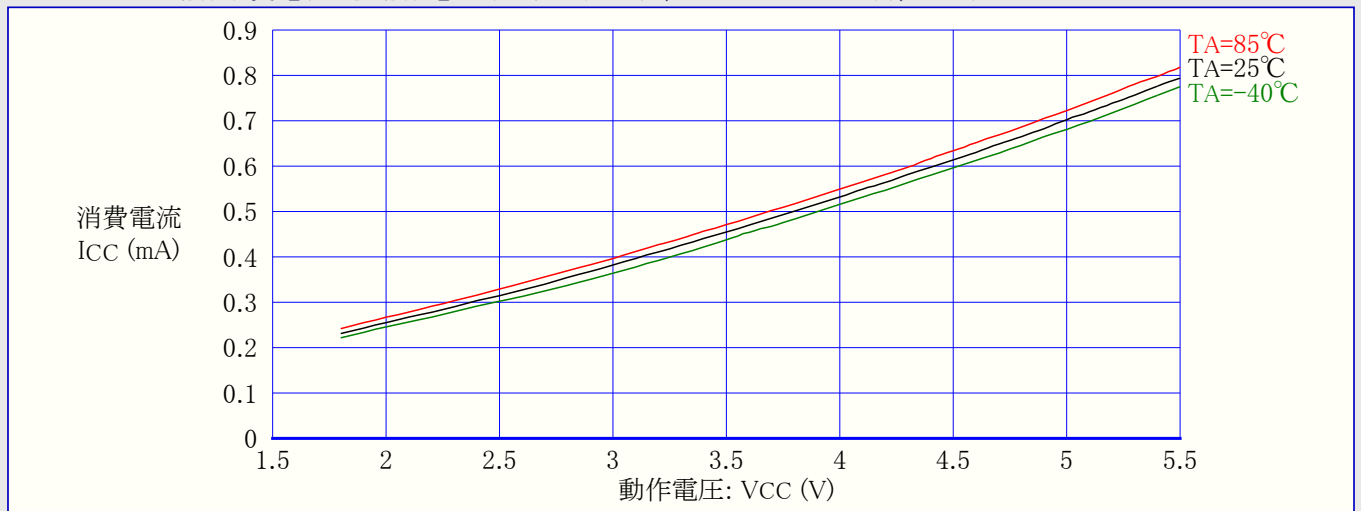


図27-8. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)





## 27.3. 周辺機能部供給電流

以下の表と式は活動動作とアイドル動作で個別周辺機能部に対する追加消費電流の計算に使えます。周辺機能部の許可や禁止は電力削減レジスタによって制御されます。詳細については26頁の「電力削減レジスタ(PRR)」をご覧ください。

表27-1. 各部追加消費電流 (絶対値:  $\mu\text{A}$ )

PRR内ビット	1MHz, 2V	4MHz, 3V	8MHz, 5V
PRADC	19	102	395
PRUSART0	9	57	240
PRSPI	11	68	276
PRTIM1	5	30	116

表27-2. 各部追加消費電流 (相対値: %)

PRR内ビット	活動動作(図27-1., 図27-2.)	アイドル動作(図27-5., 図27-6.)
PRADC	4.2	9.0
PRUSART0	2.3	4.9
PRSPI	2.7	5.9
PRTIM1	1.2	2.6

表27-1.で一覧される以外のVCCと周波数設定については表27-2.からの数値を元に代表的な消費電流を計算できます。

例1: VCC=3V,  $f=1\text{MHz}$ でUSART0, SPI, タイマ/カウンタ1が許可されたアイドル動作での予測される消費電流を計算します。表27-2.のアイドル動作列からUSART0が4.9%、SPIが5.9%、タイマ/カウンタ1が2.6%追加する必要があります。図27-5.を読み、VCC=3V,  $f=1\text{MHz}$ でのアイドル動作消費電流が約0.21mAである事を得ます。USART0, SPI, タイマ/カウンタ1許可のアイドル動作での総消費電流を得ます。

$$\text{総消費電流} = 0.21\text{mA} \times (1 + 0.049 + 0.059 + 0.026) \approx 0.24\text{mA}$$

## 27.4. パワーダウン動作消費電流

図27-9. 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)

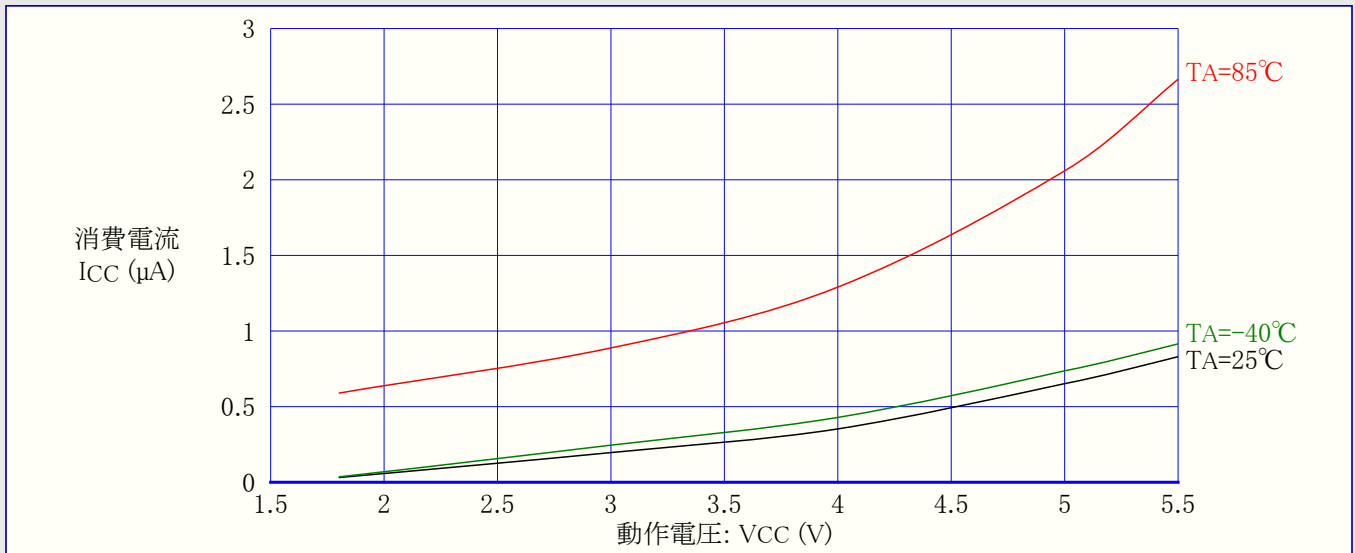
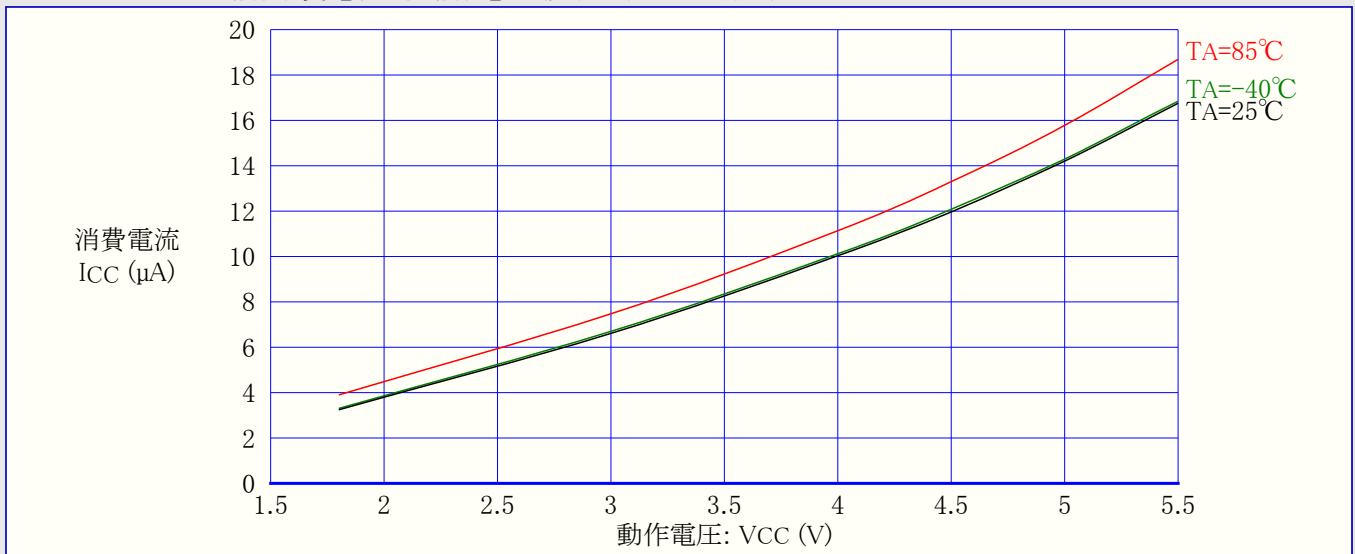
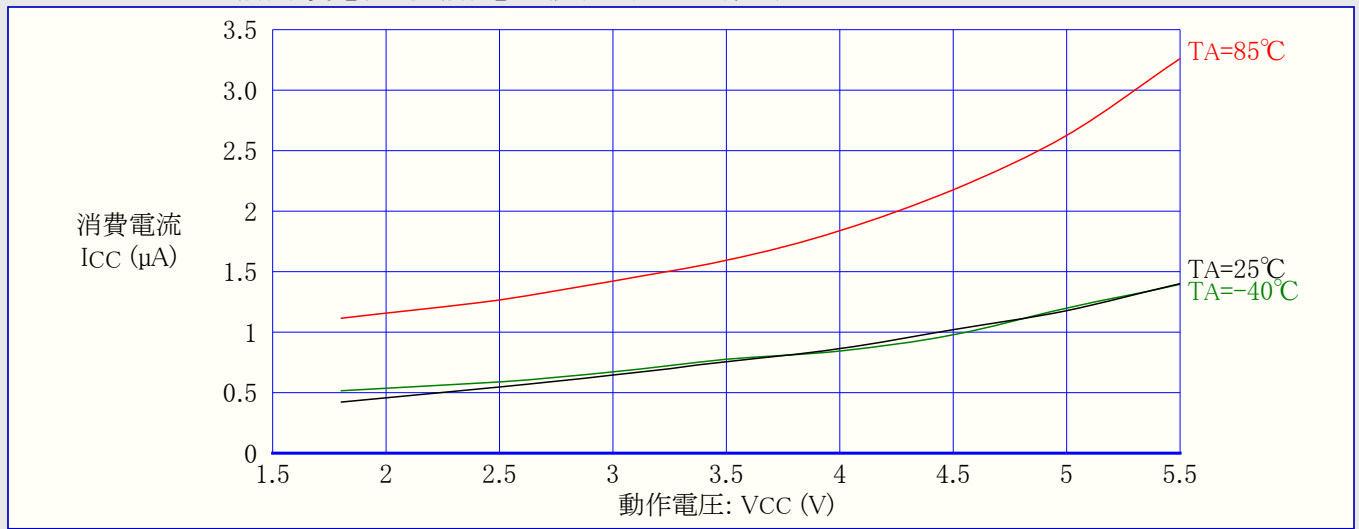


図27-10. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



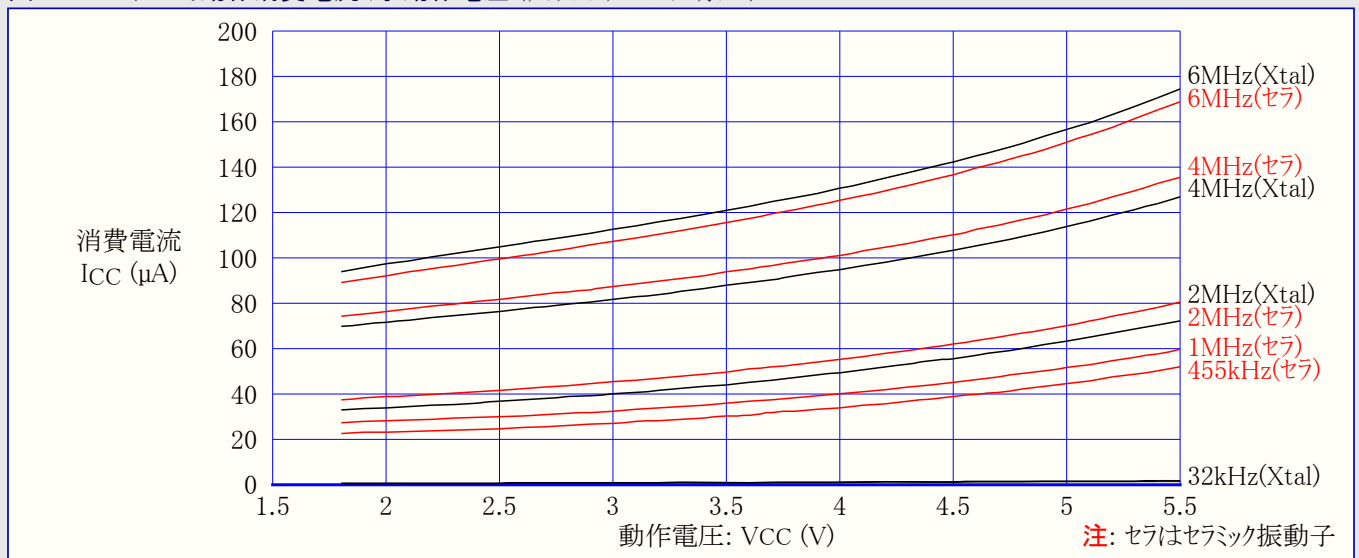
## 27.5. パワーセーブ動作消費電流

図27-11. パワーセーブ動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)



## 27.6. スタンバイ動作消費電流

図27-12. スタンバイ動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)



## 27.7. ピンプルアップ

図27-13. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

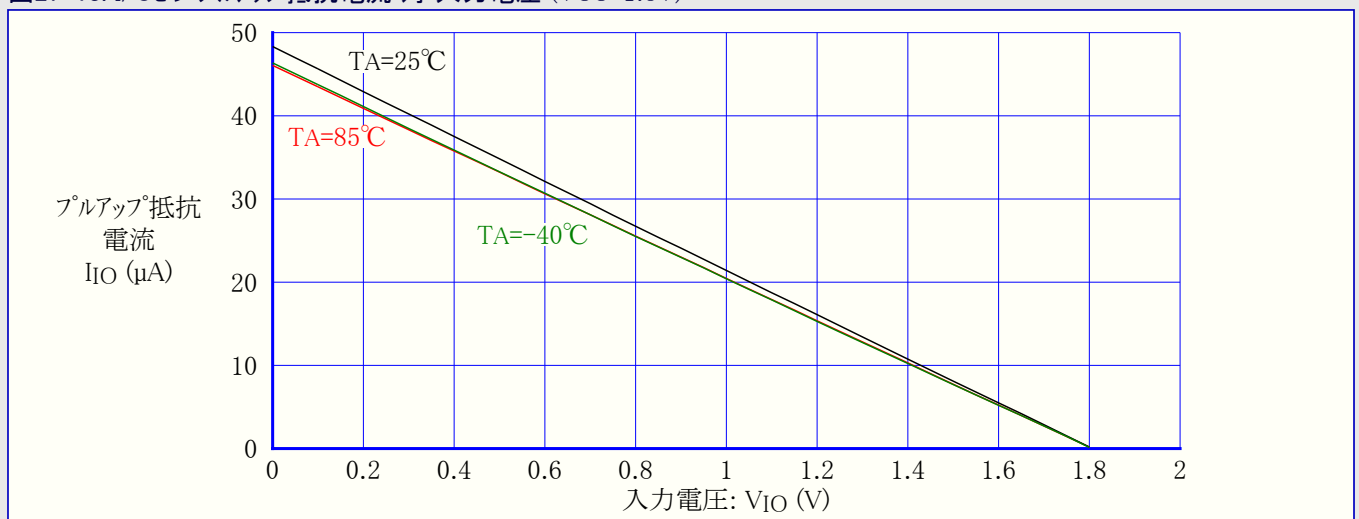


図27-14. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

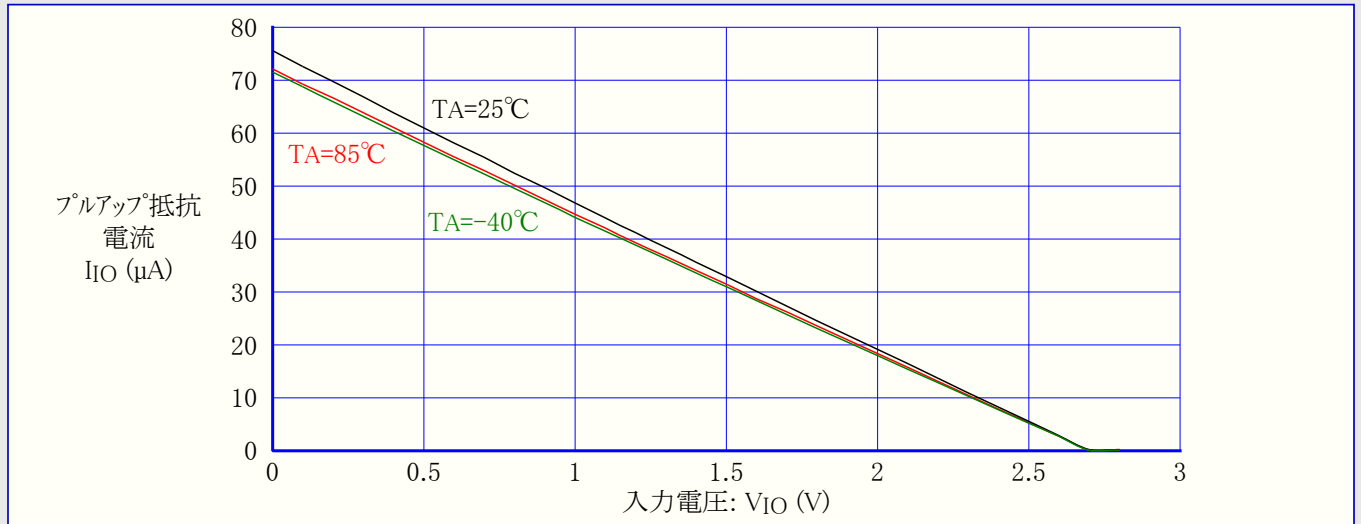


図27-15. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

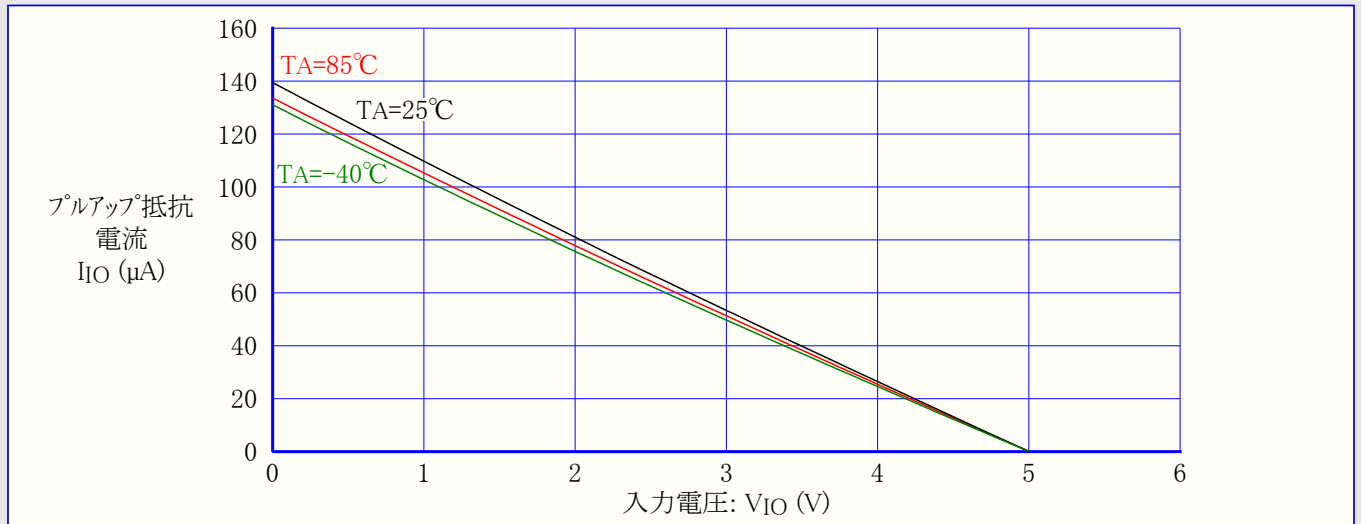


図27-16. RESETプルアップ抵抗電流 対 入力電圧 (VCC=1.8V)

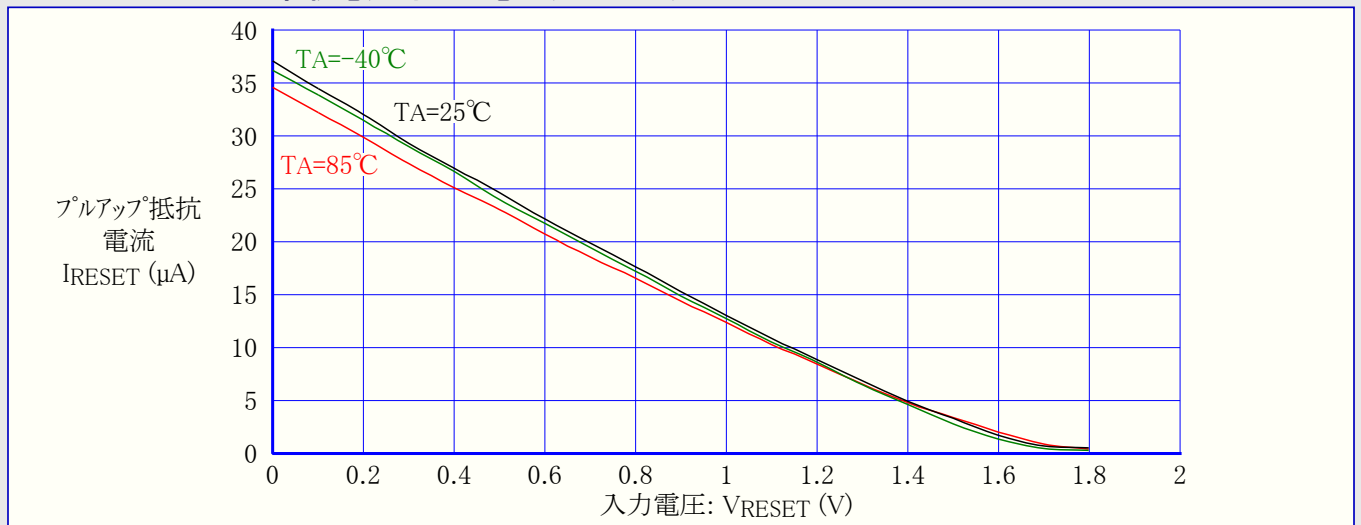


図27-17. RESETフルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

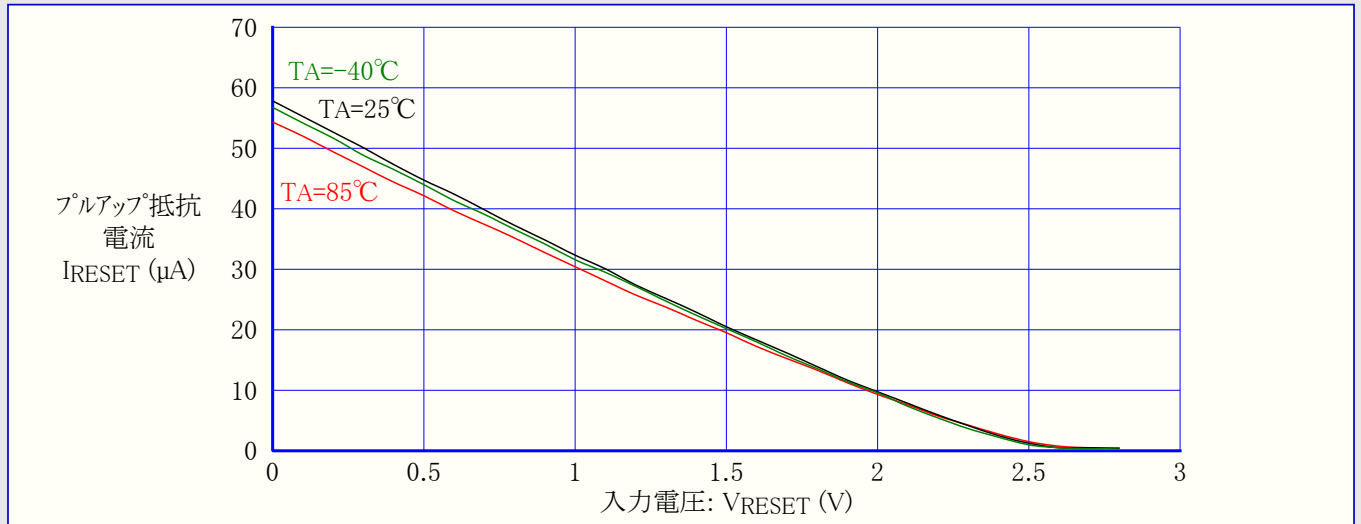
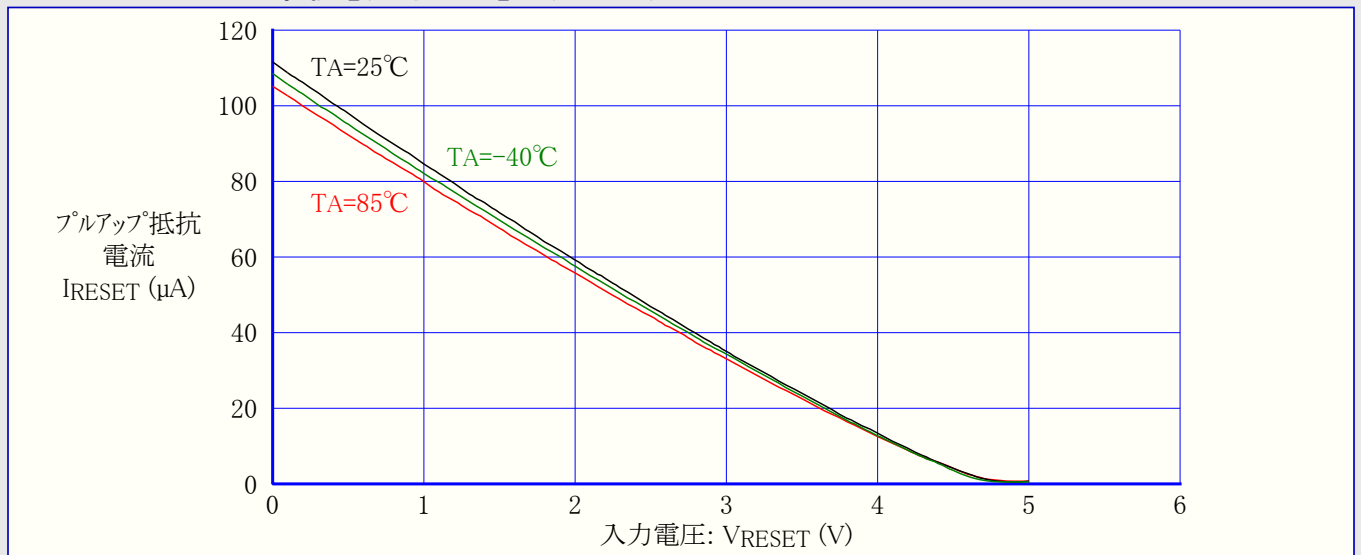


図27-18. RESETフルアップ抵抗電流 対 入力電圧 (VCC=5V)



## 27.8. ピン駆動能力

図27-19. I/Oピン(ポートA,C,D,E,F,G) 吸い込み電流 対 出力電圧 (VCC=1.8V)

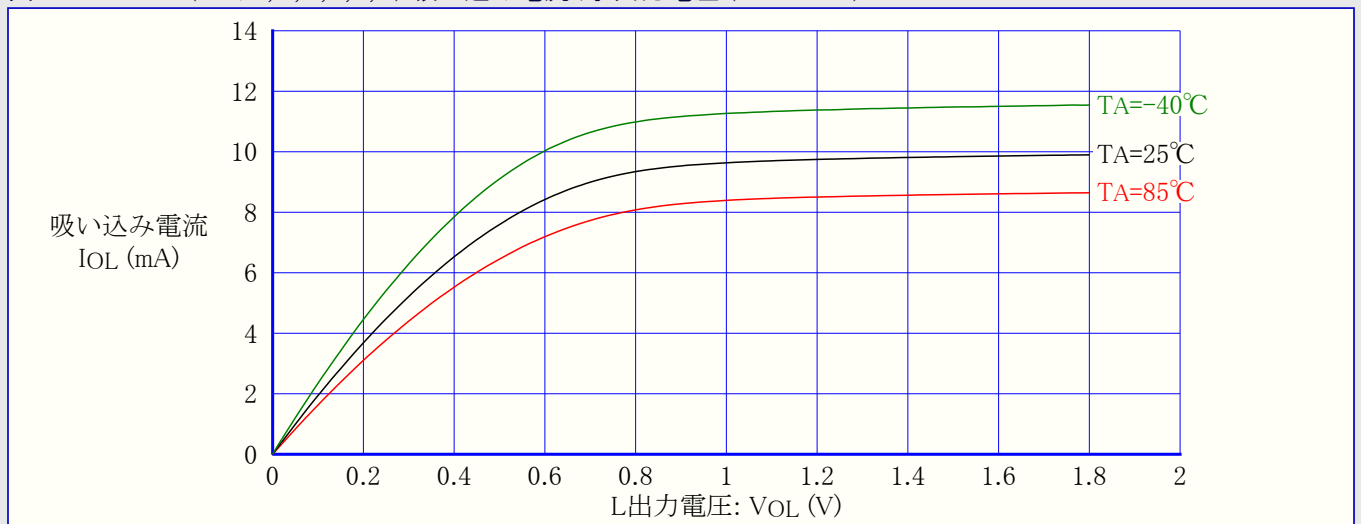


図27-20. I/Oピン(ポートB) 吸い込み電流 対 出力電圧 (VCC=2.7V)

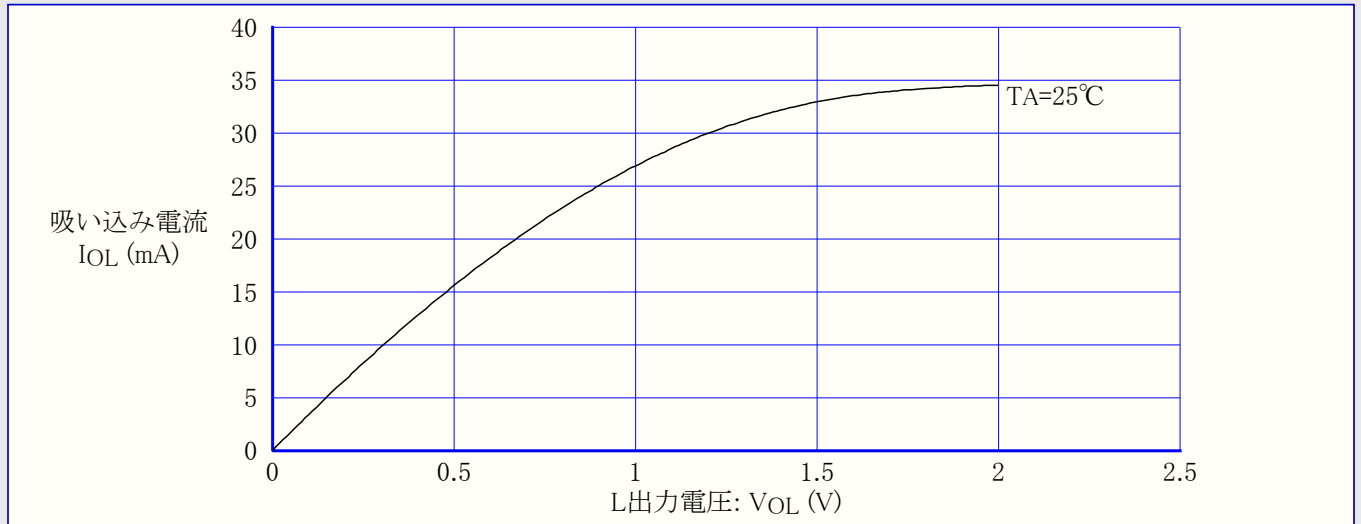


図27-21. I/Oピン(ポートB) 吸い込み電流 対 出力電圧 (VCC=5V)

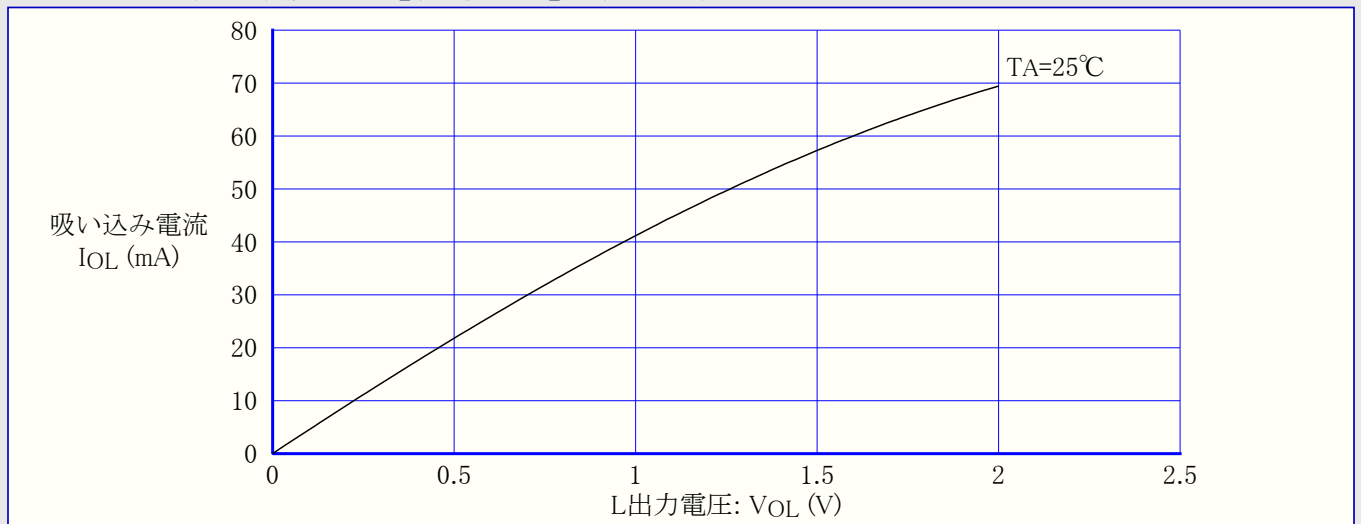


図27-22. I/Oピン(ポートA,C,D,E,F,G,H,J) 吸い込み電流 対 出力電圧 (VCC=1.8V)

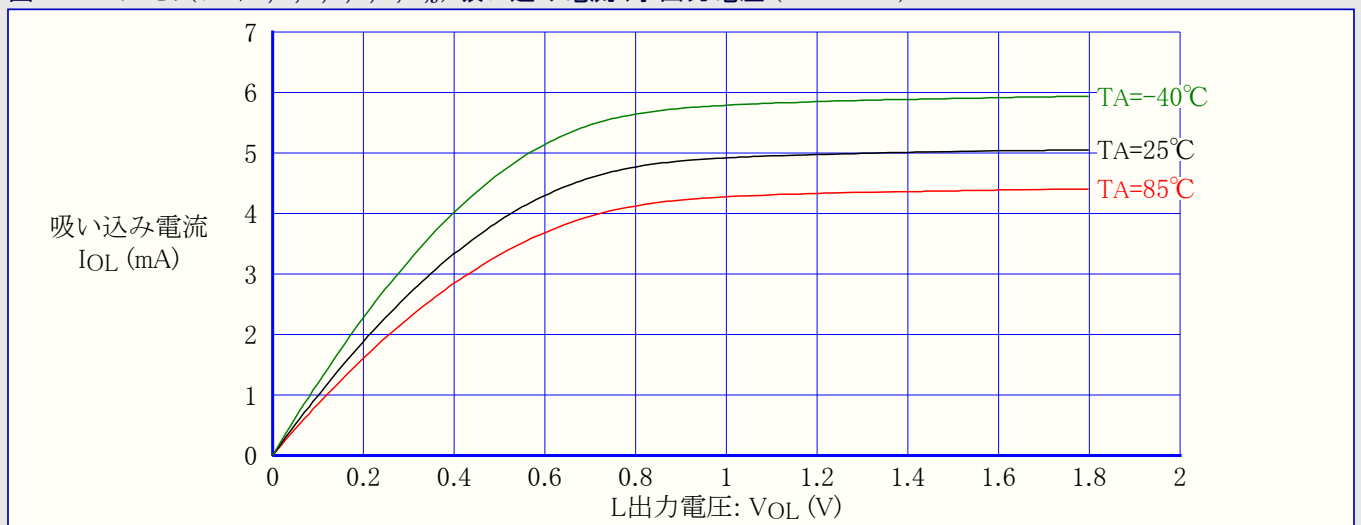




図27-23. I/Oピン(ポートA,C,D,E,F,G,H,J) 吸い込み電流 対 出力電圧 (VCC=2.7V)

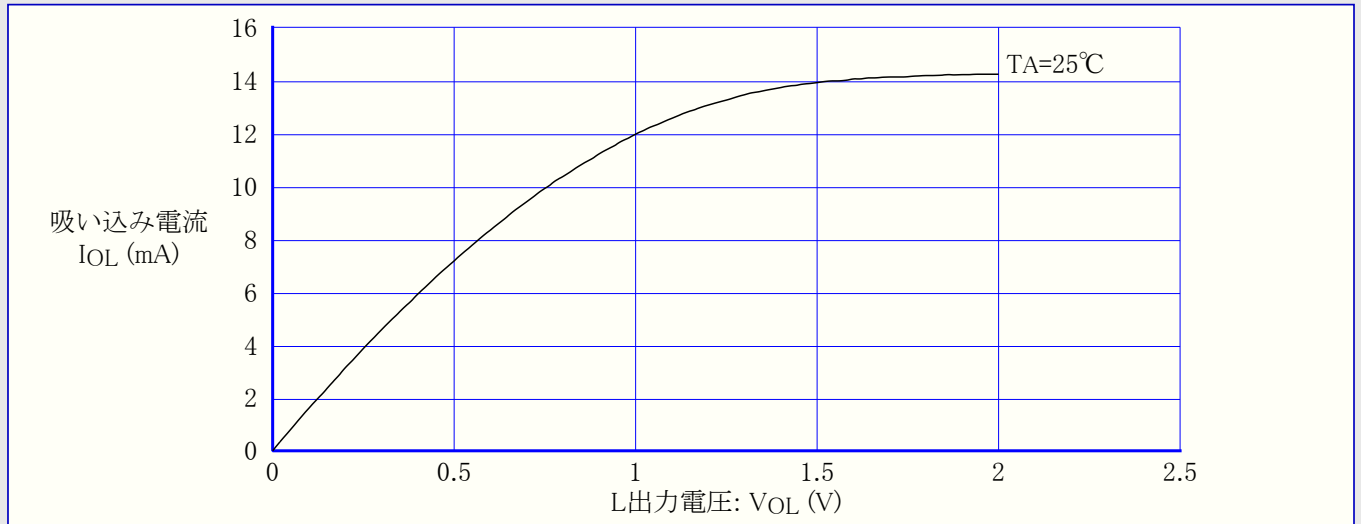


図27-24. I/Oピン(ポートA,C,D,E,F,G,H,J) 吸い込み電流 対 出力電圧 (VCC=5V)

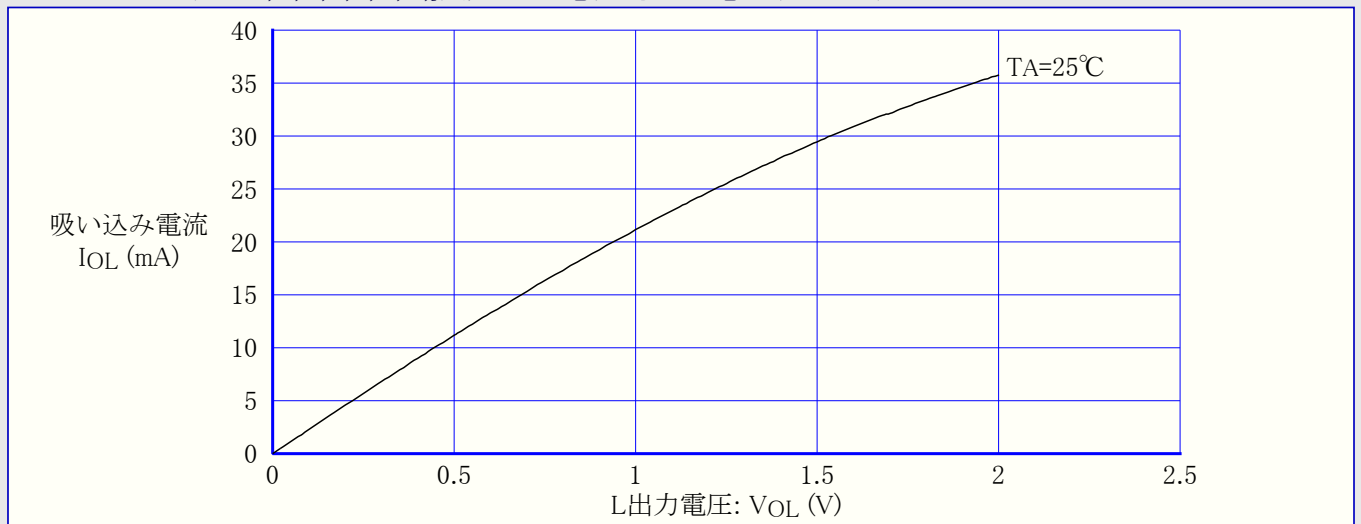


図27-25. I/Oピン(ポートB) 吐き出し電流 対 出力電圧 (VCC=1.8V)

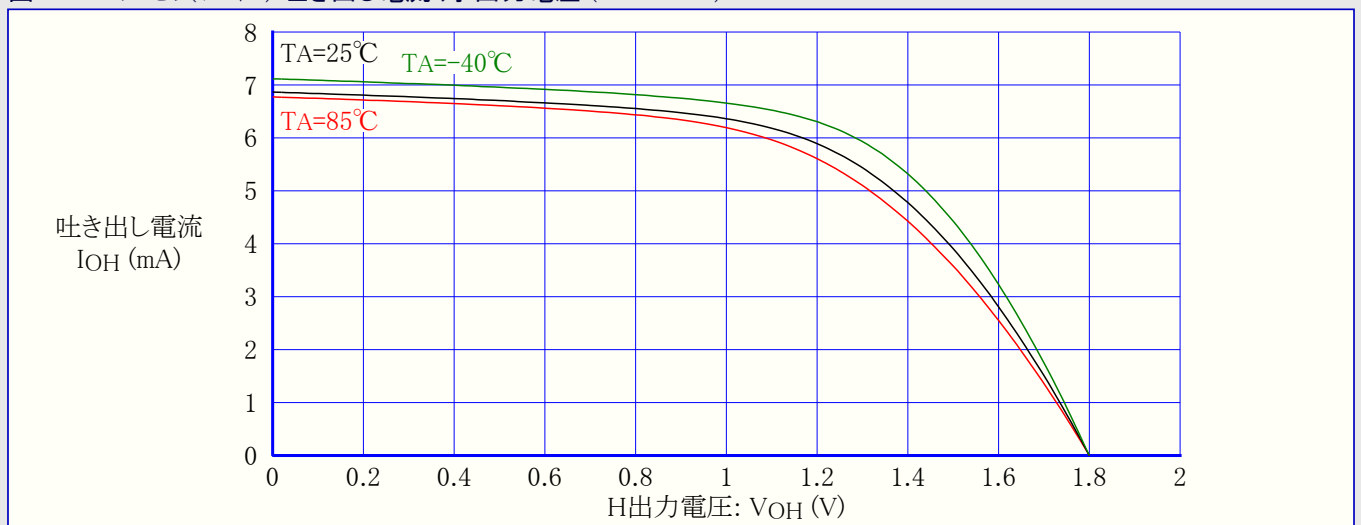


図27-26. I/Oピン(ポートB) 吐き出し電流 対 出力電圧 (VCC=2.7V)

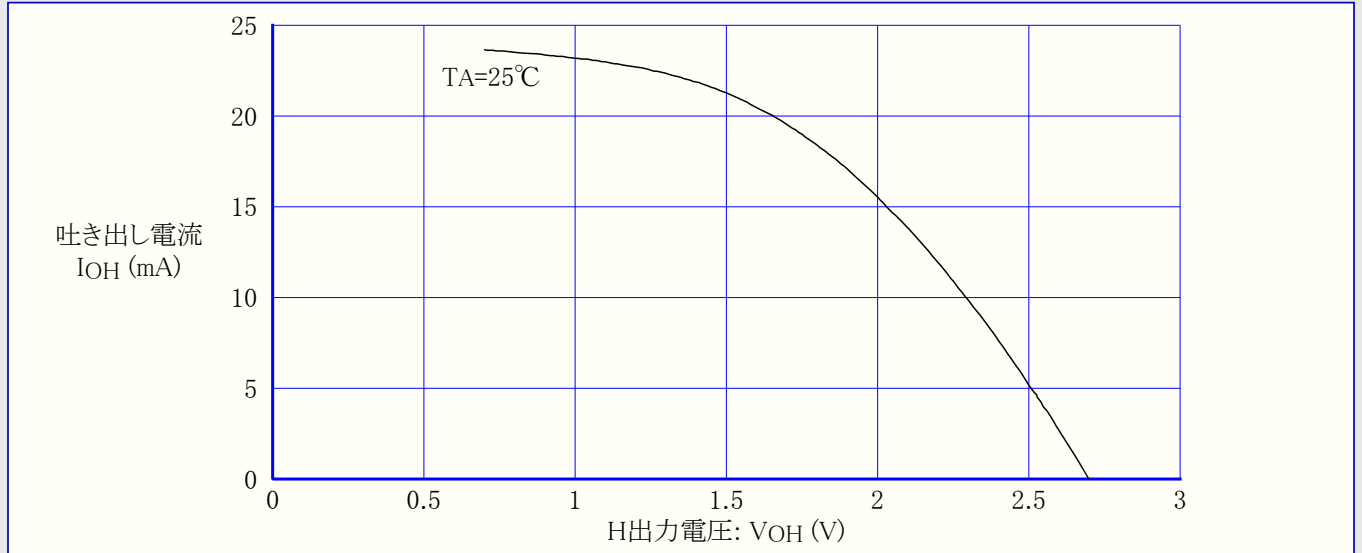


図27-27. I/Oピン(ポートB) 吐き出し電流 対 出力電圧 (VCC=5V)

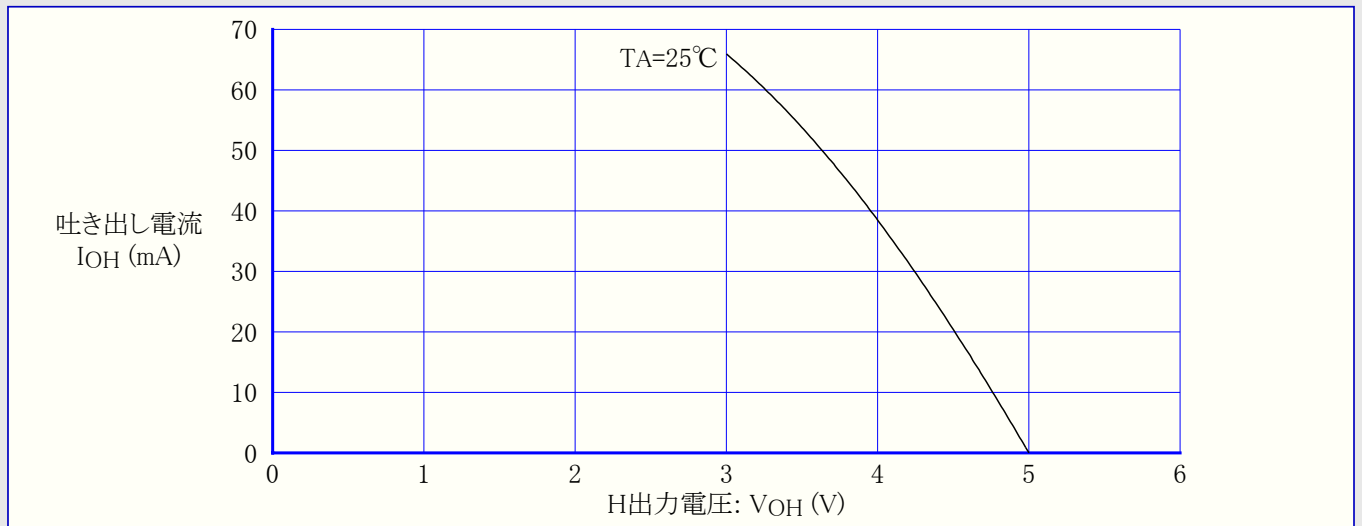


図27-28. I/Oピン(ポートA,C,D,E,F,G,H,J) 吐き出し電流 対 出力電圧 (VCC=1.8V)

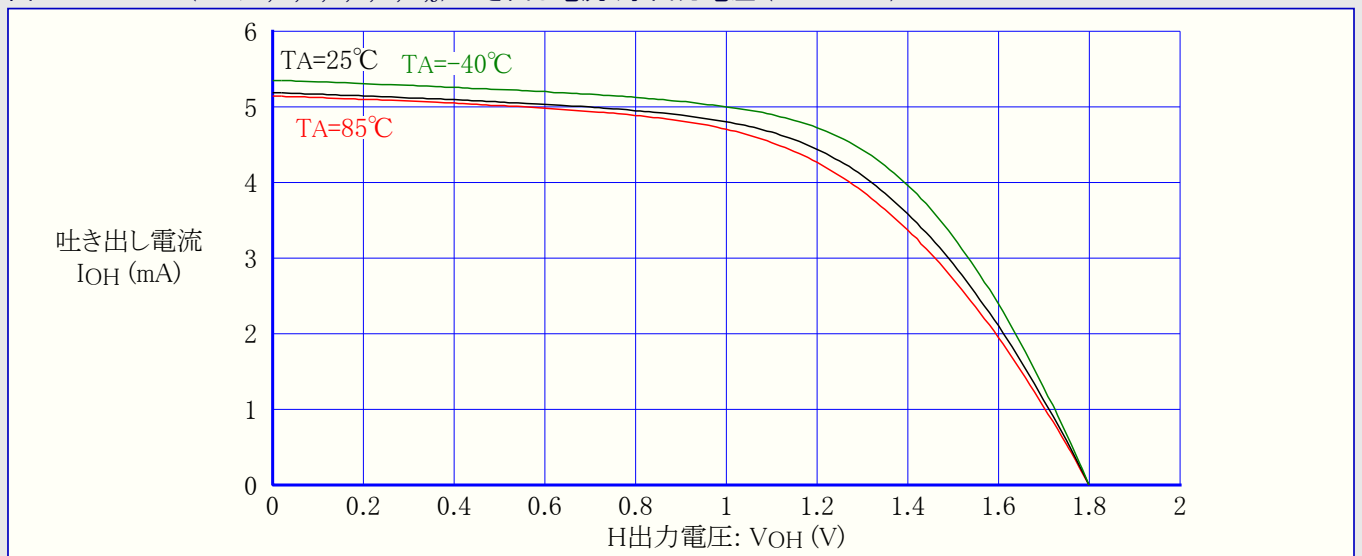


図27-29. I/Oピン(ポートA,C,D,E,F,G,H,J) 吐き出し電流 対 出力電圧 (VCC=2.7V)

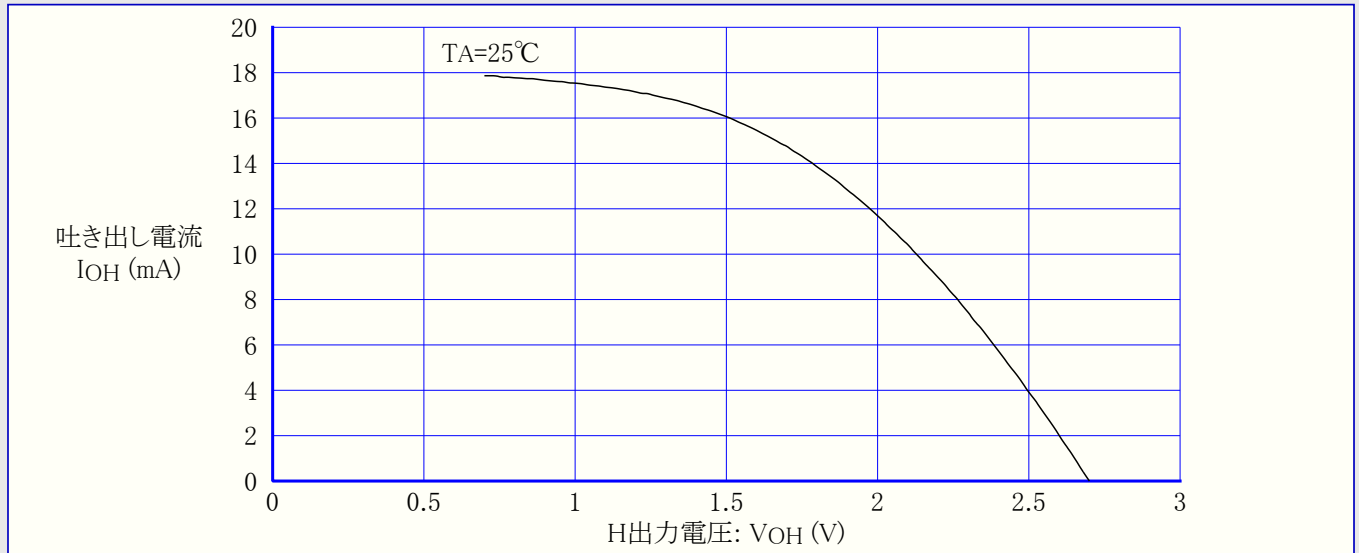
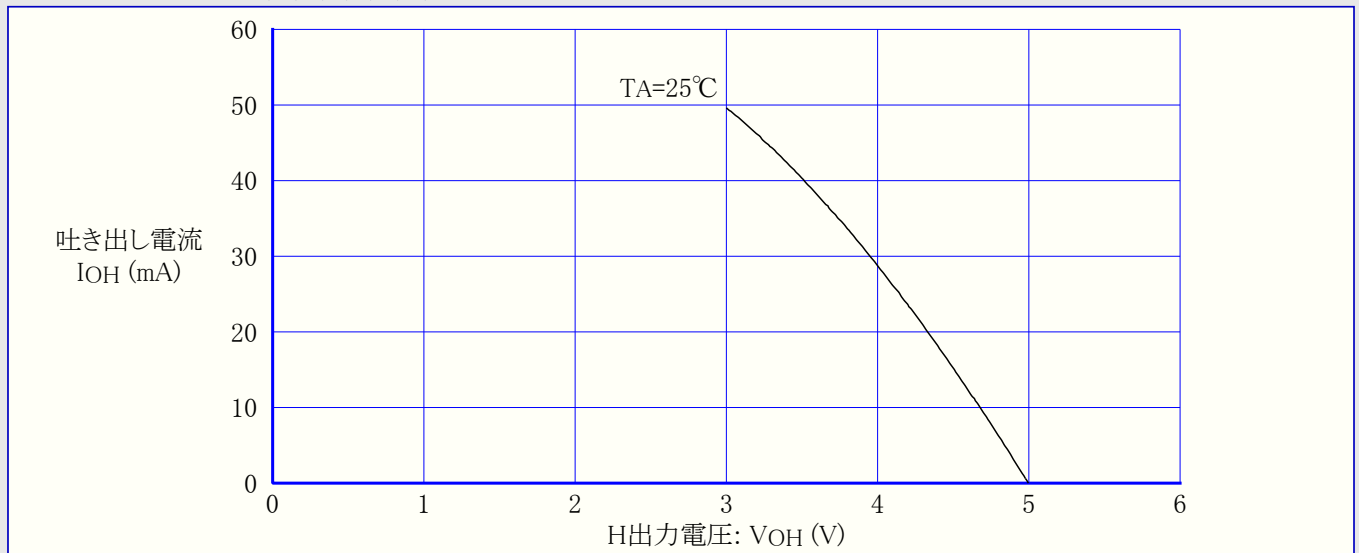


図27-30. I/Oピン(ポートA,C,D,E,F,G,H,J) 吐き出し電流 対 出力電圧 (VCC=5V)



## 27.9. ピン 閾値とヒステリシス

図27-31. I/Oピン(ポートB)入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

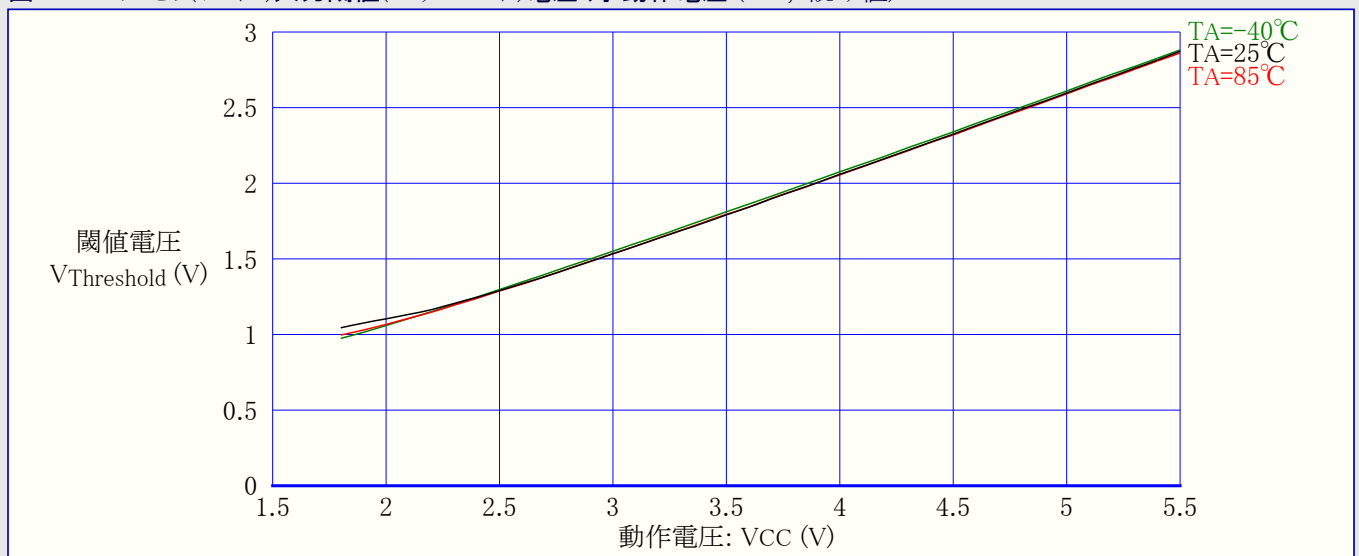


図27-32. I/Oピン(ポートB)入力閾値(スレッショルド)電圧 対 動作電圧 (VIL,0読み値)

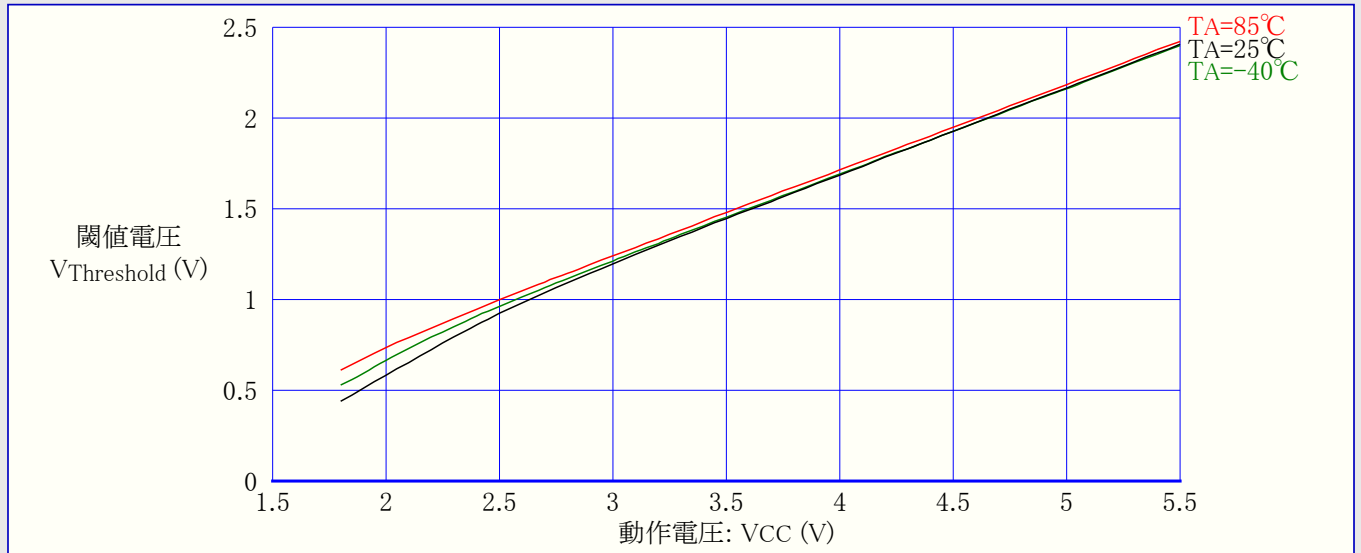


図27-33. I/Oピン(ポートB)入力ヒステリシス電圧 対 動作電圧

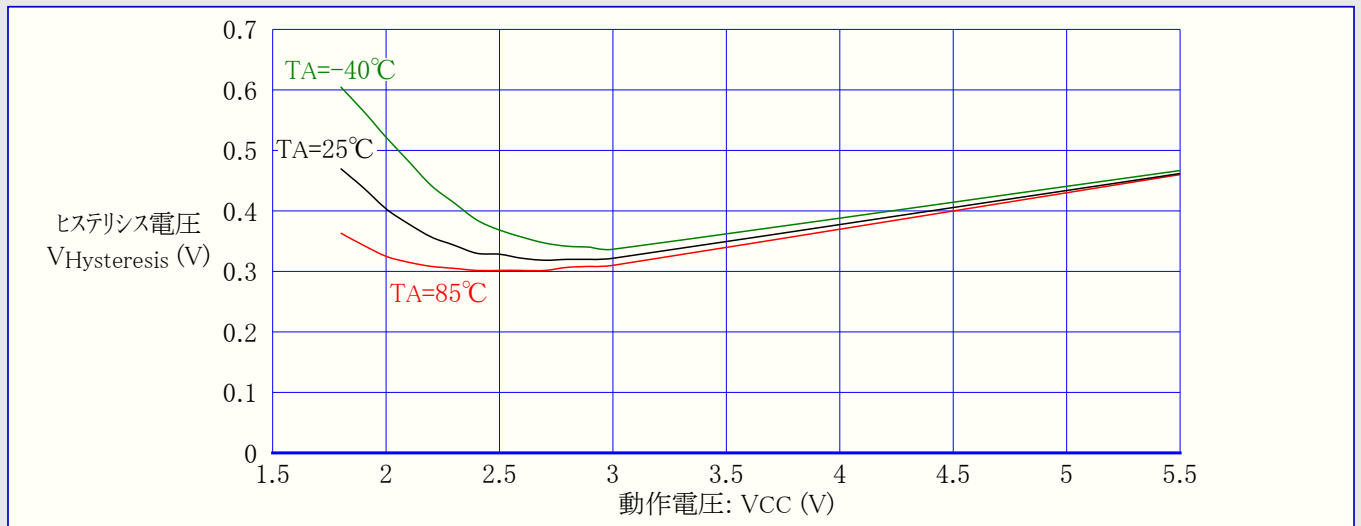


図27-34. I/Oピン(ポートA,C,D,E,F,G,H,J)入力閾値(スレッショルド)電圧 対 動作電圧 (VIH,1読み値)

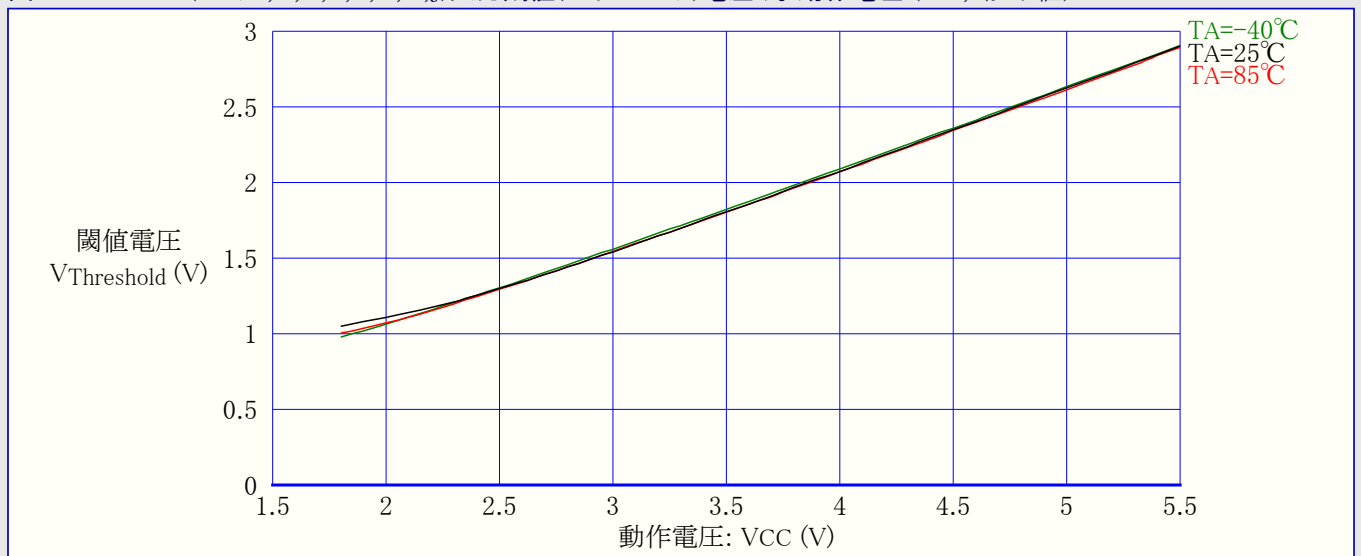


図27-35. I/Oピン(ポートA,C,D,E,F,G,H,J)入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

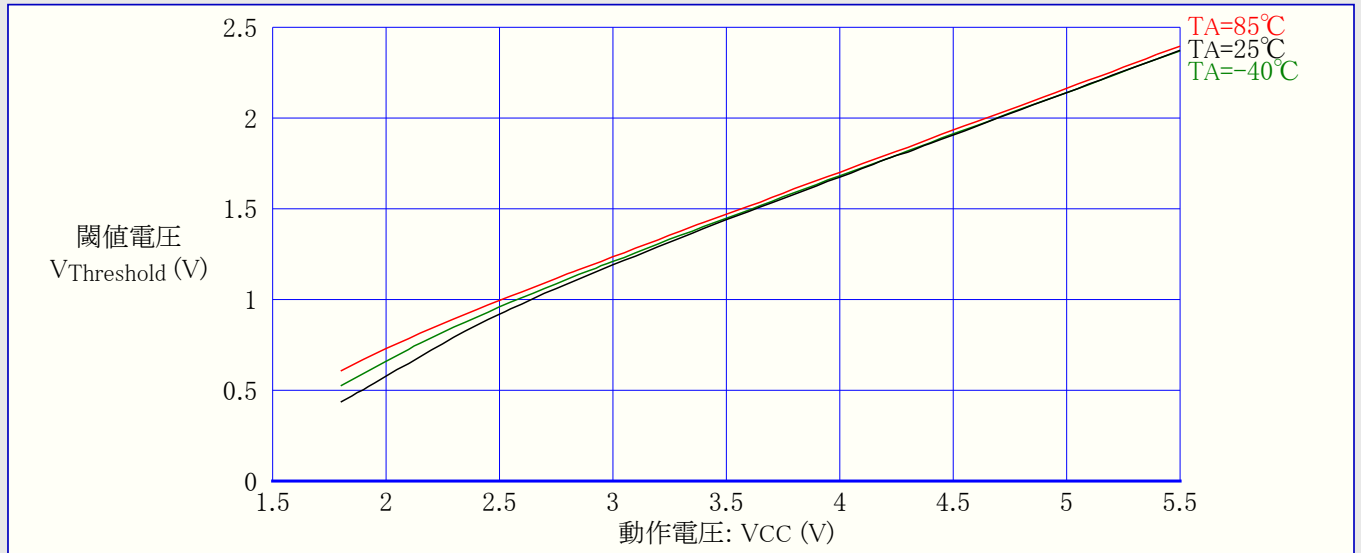


図27-36. I/Oピン(ポートA,C,D,E,F,G,H,J)入力ヒステリシス電圧 対 動作電圧

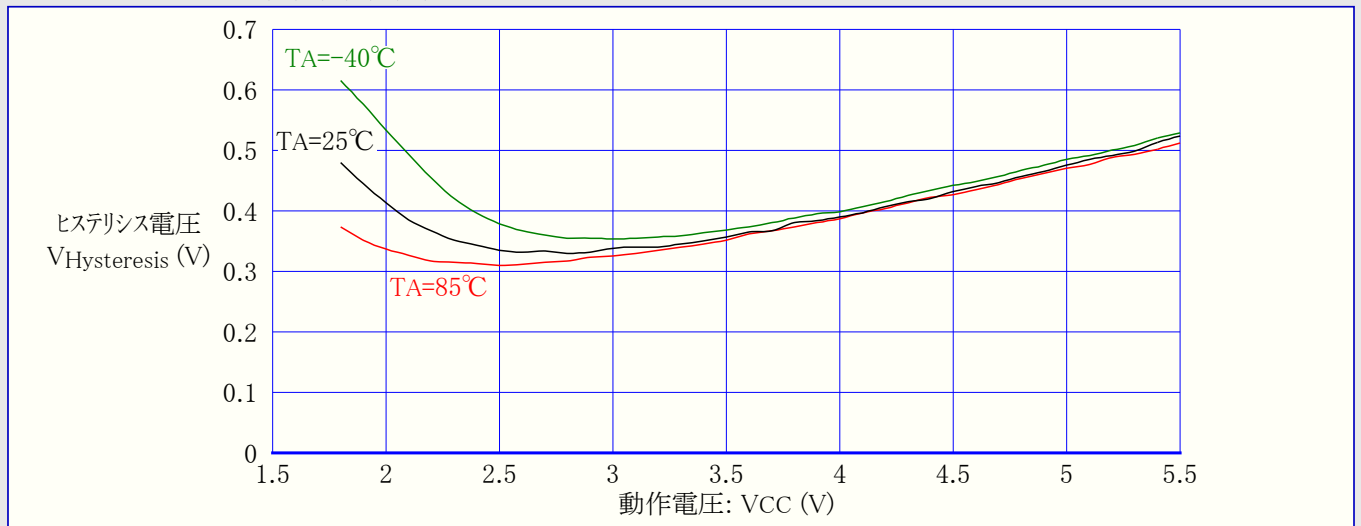


図27-37. RESET入力閾値(スレッショルド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

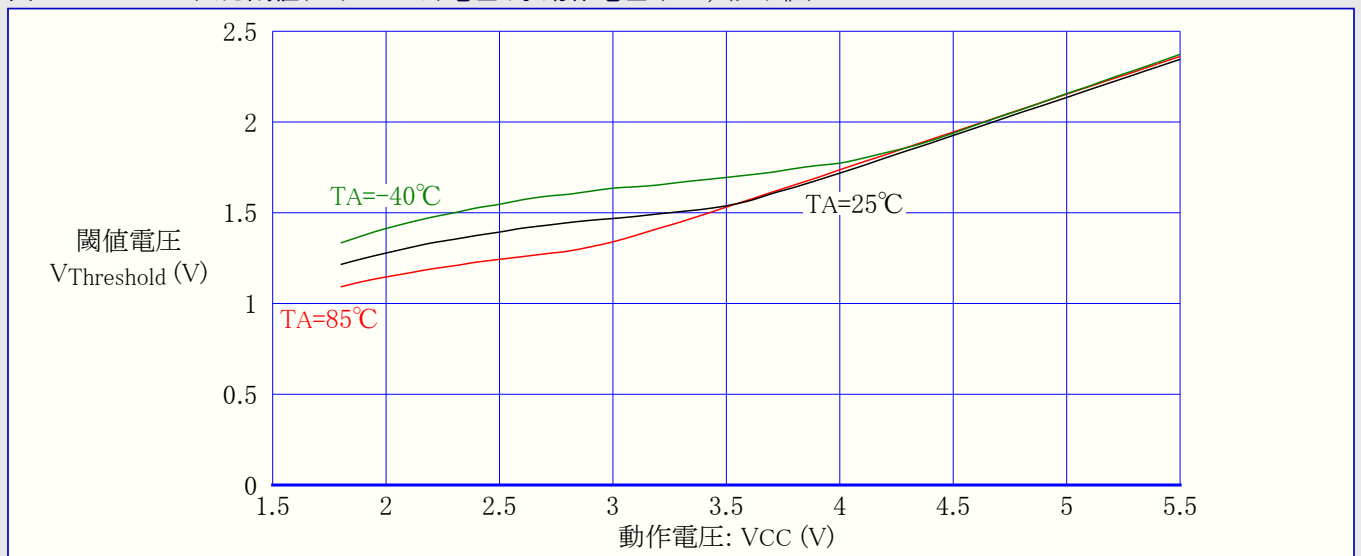




図27-38. RESET入力閾値(スレッショルド)電圧 対 動作電圧 (VIL,0読み値)

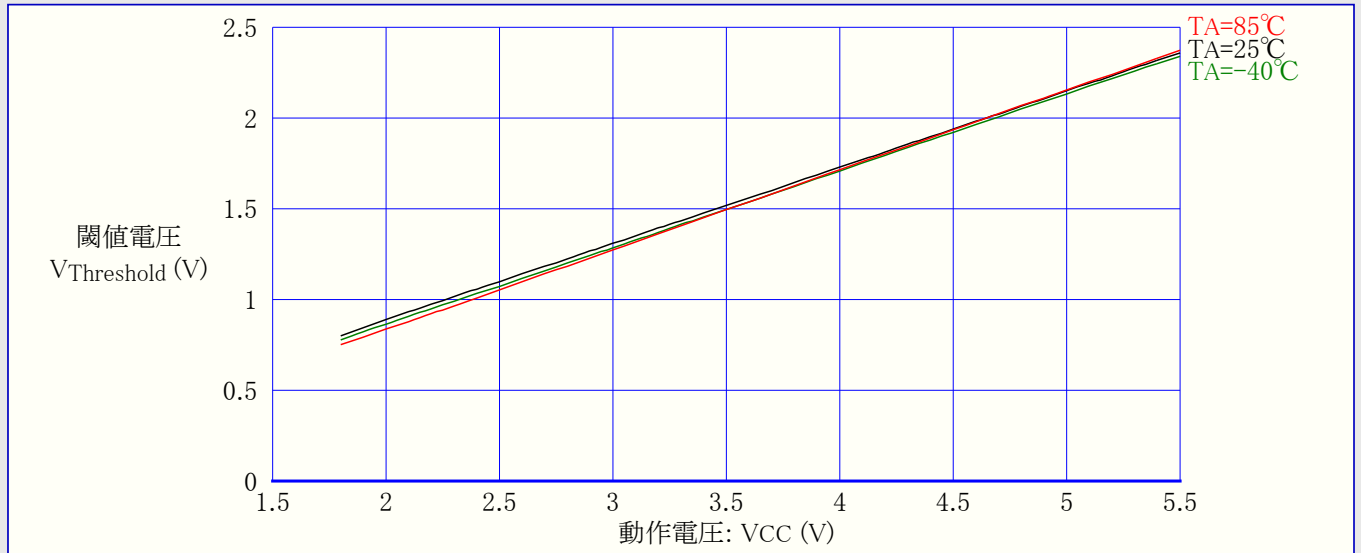
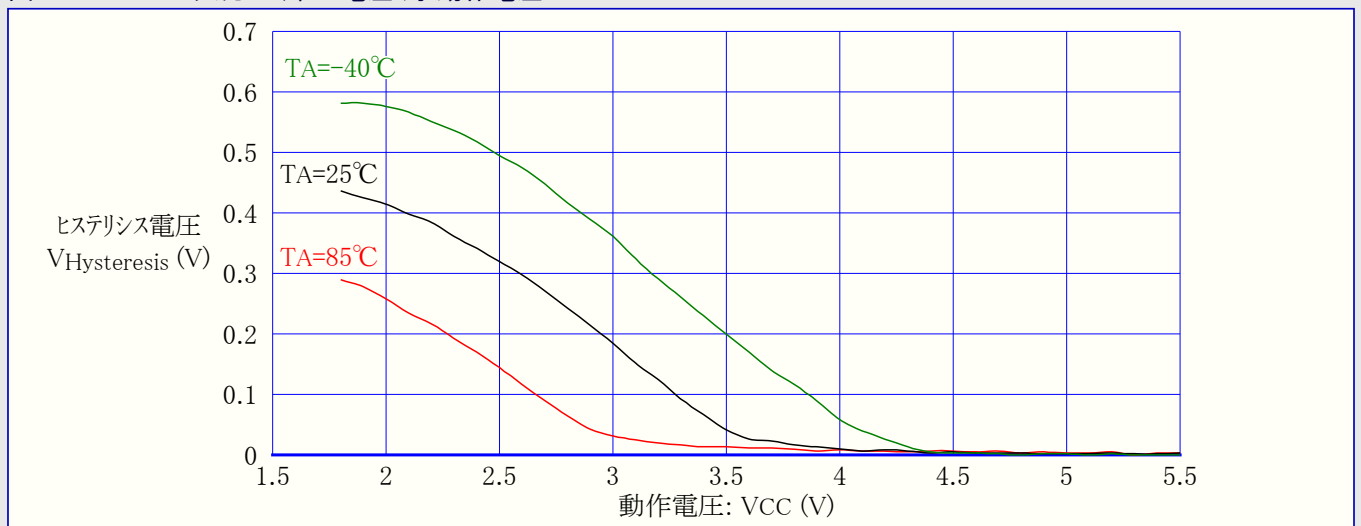


図27-39. RESET入力ヒステリシス電圧 対 動作電圧



## 27.10. 内部発振器周波数

図27-40. ウォッチドッグ発振器周波数 対 動作電圧

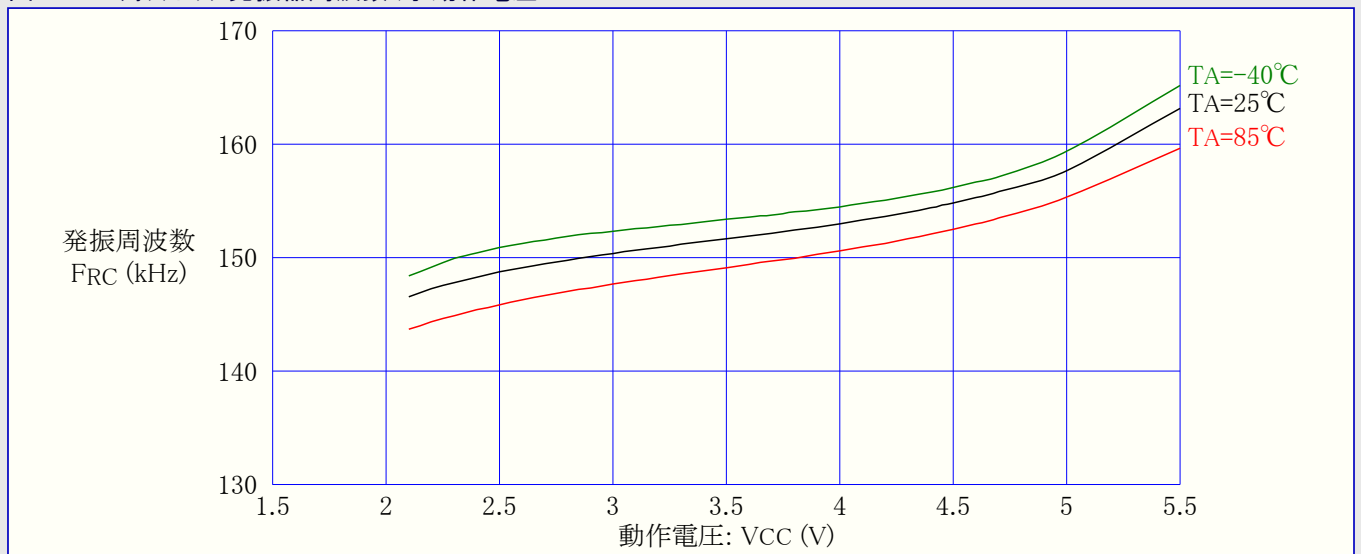


図27-41. ウォッチドッグ発振器周波数 対 動作温度

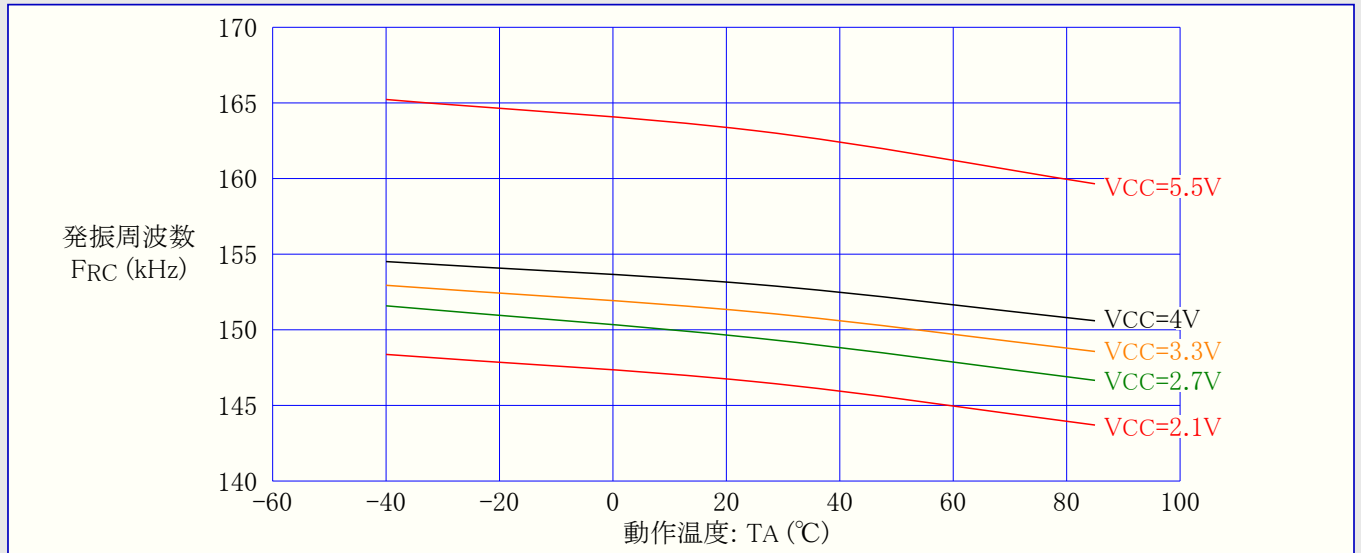


図27-42. 校正済み8MHz内蔵RC発振器周波数 対 動作電圧

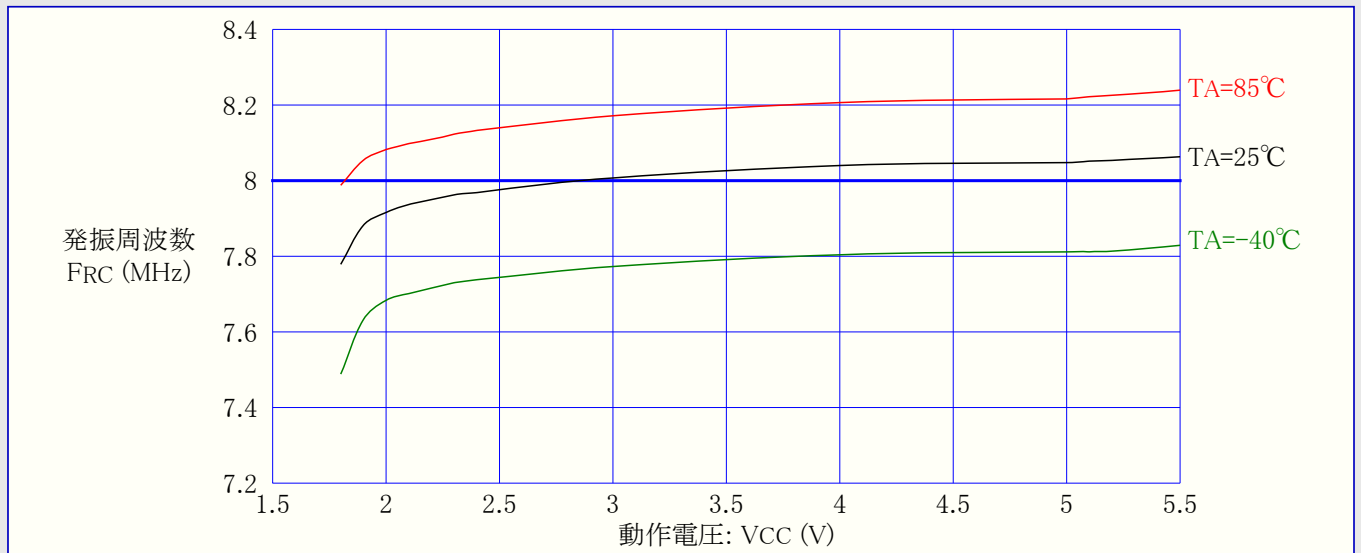


図27-43. 校正済み8MHz内蔵RC発振器周波数 対 動作温度

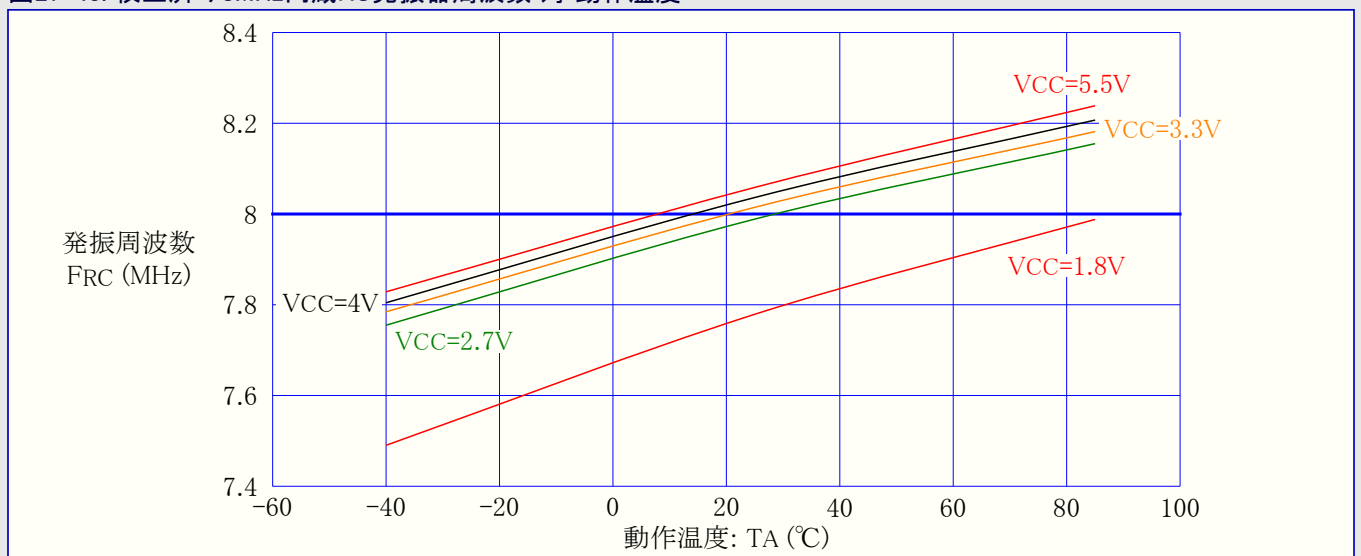
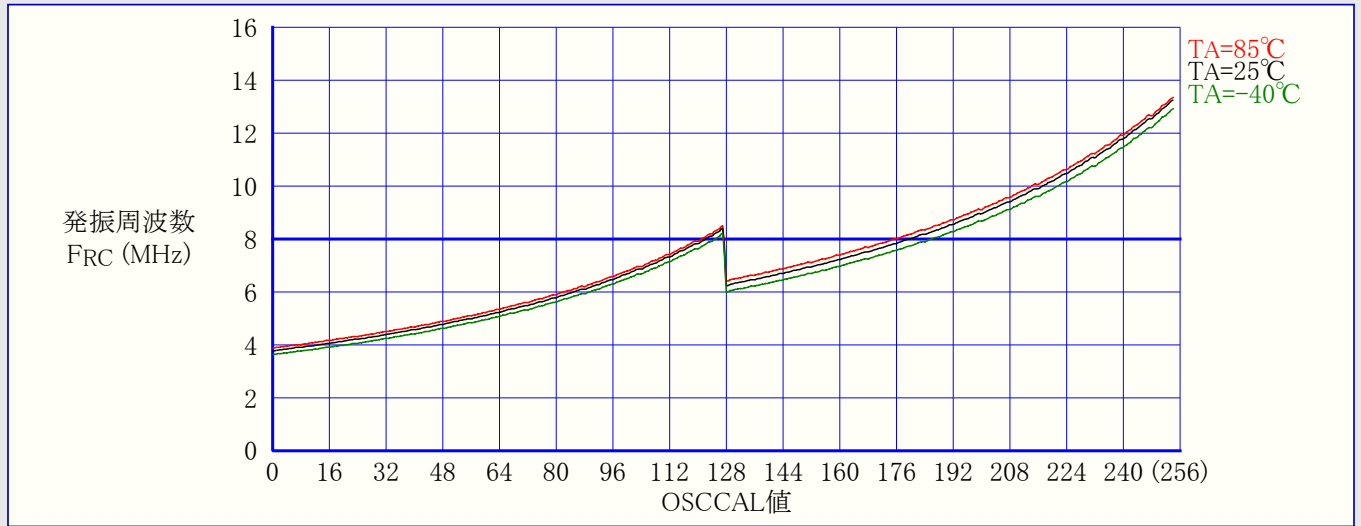


図27-44. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値



## 27.11. 周辺機能部消費電流

図27-45. 低電圧検出器(BOD)消費電流 対 動作電圧

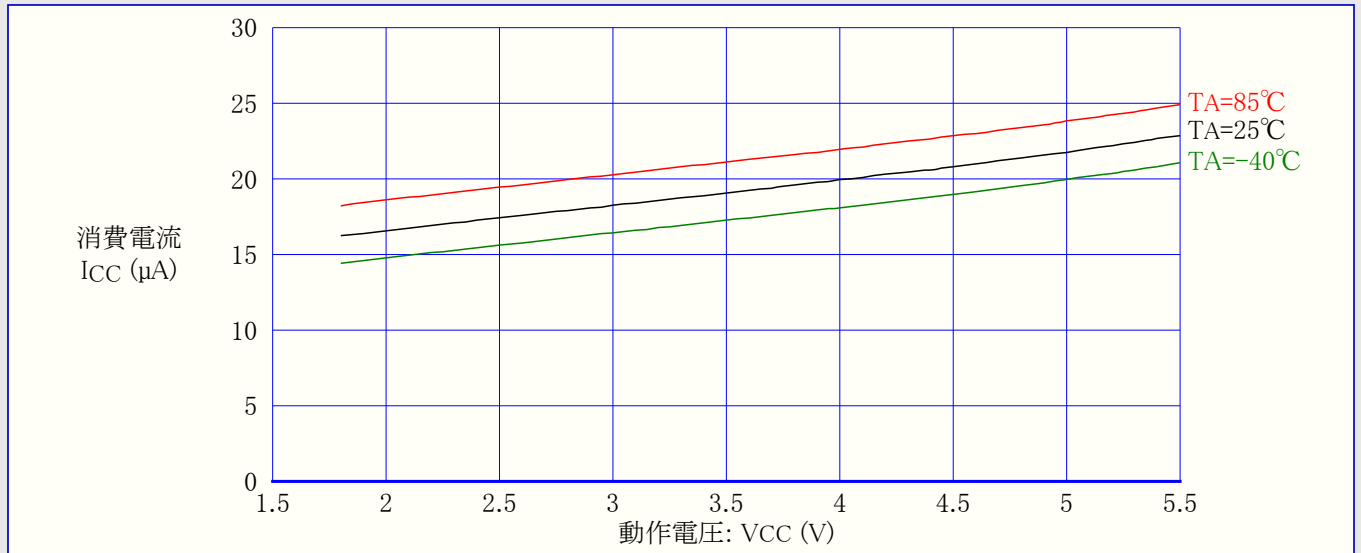


図27-46. A/D変換器消費電流 対 動作電圧 (AREF=AVCC)

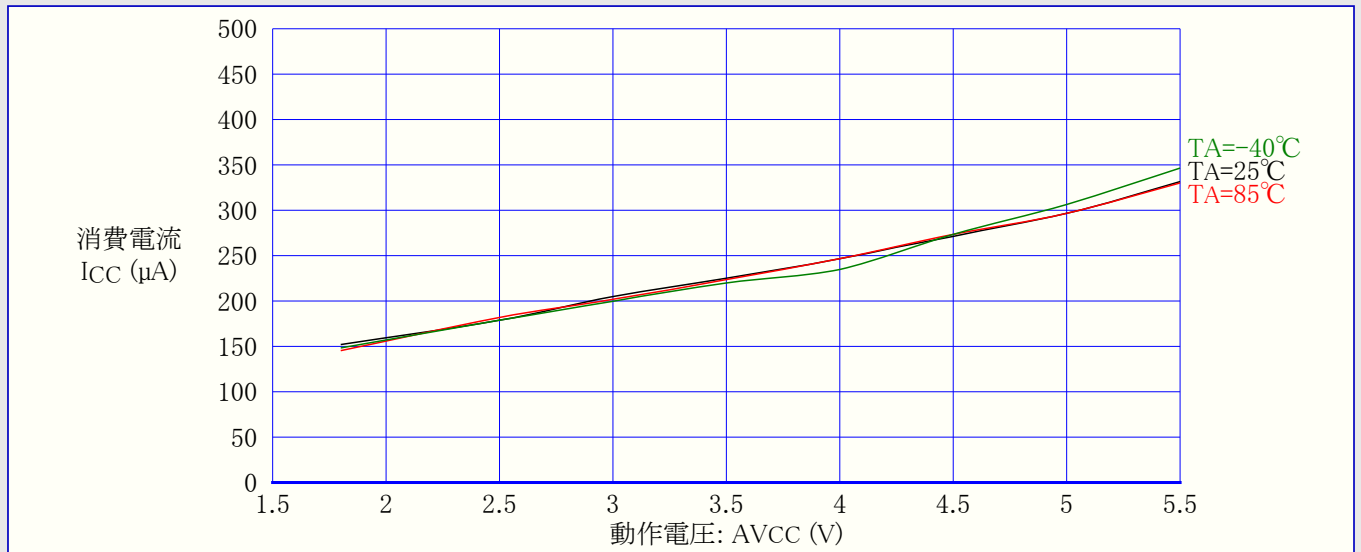


図27-47. 外部基準電圧(AREF)電流 対 動作電圧

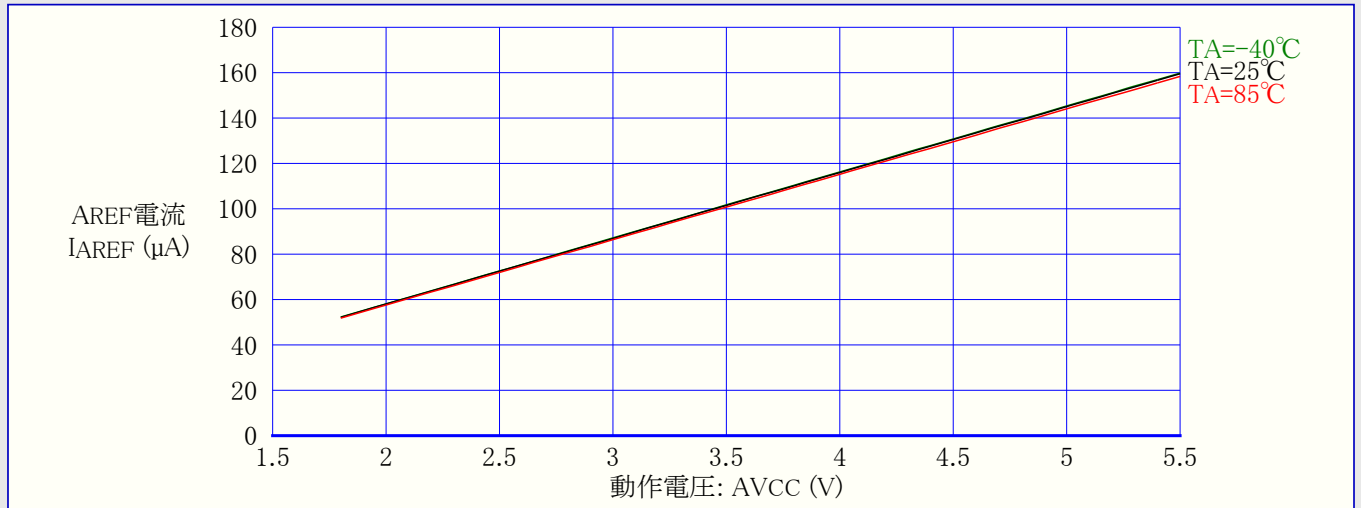


図27-48. ウォッチドッグ タイマ消費電流 対 動作電圧

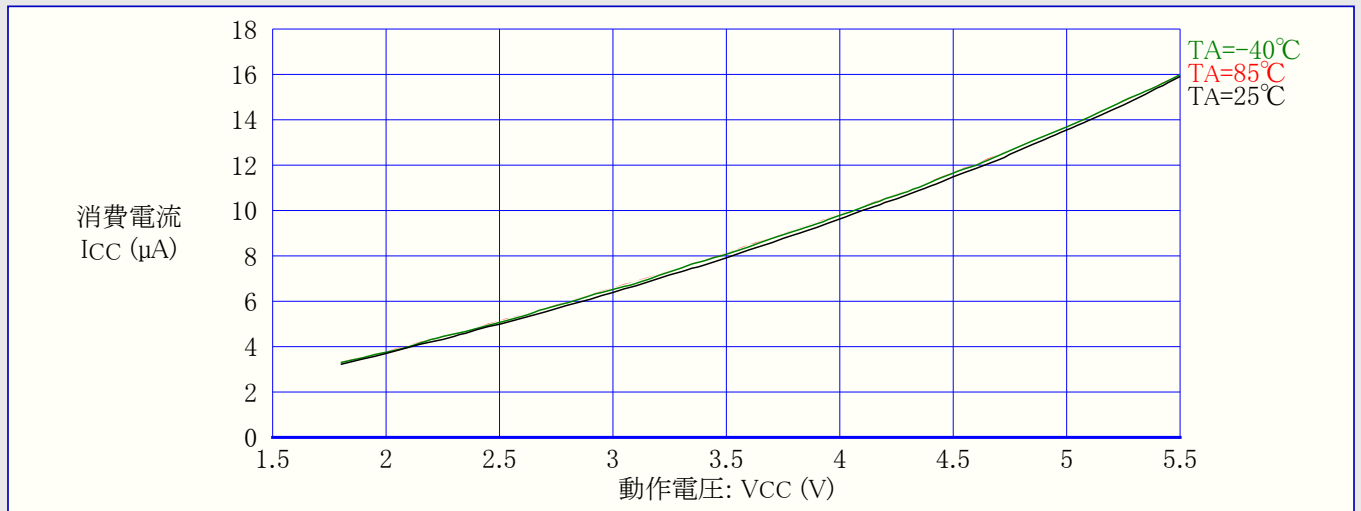


図27-49. アナログ比較器消費電流 対 動作電圧

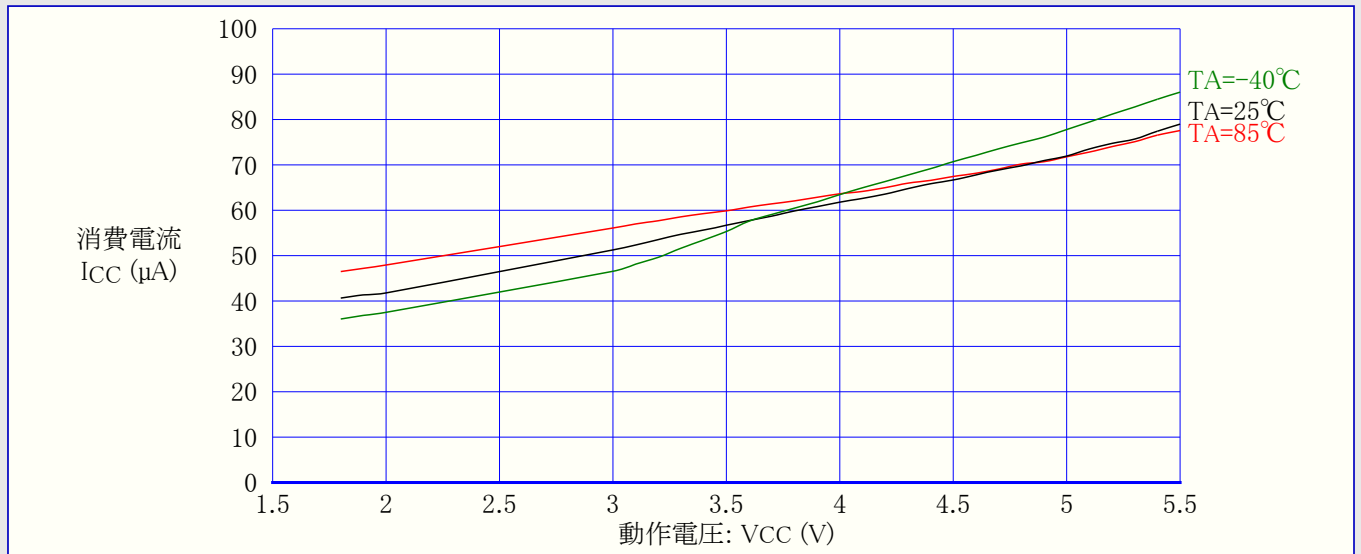
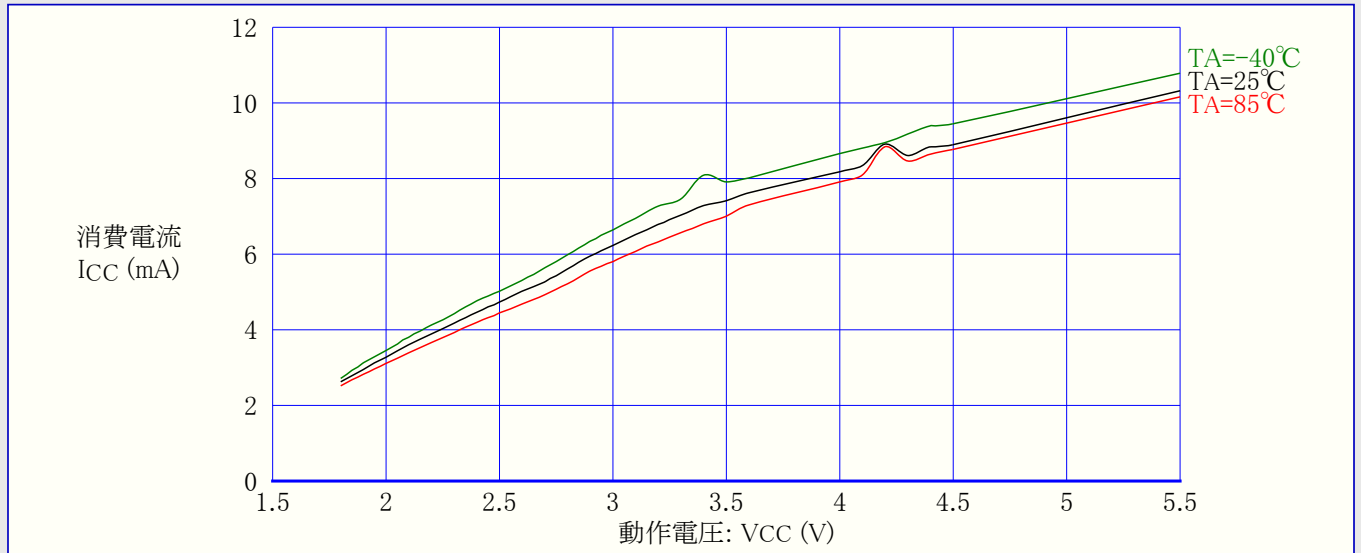


図27-50. プログラミング電流 対 動作電圧



## 27.12. リセット消費電流とリセット パルス幅

図27-51. リセット消費(供給)電流 対 周波数 (100kHz~1MHz)

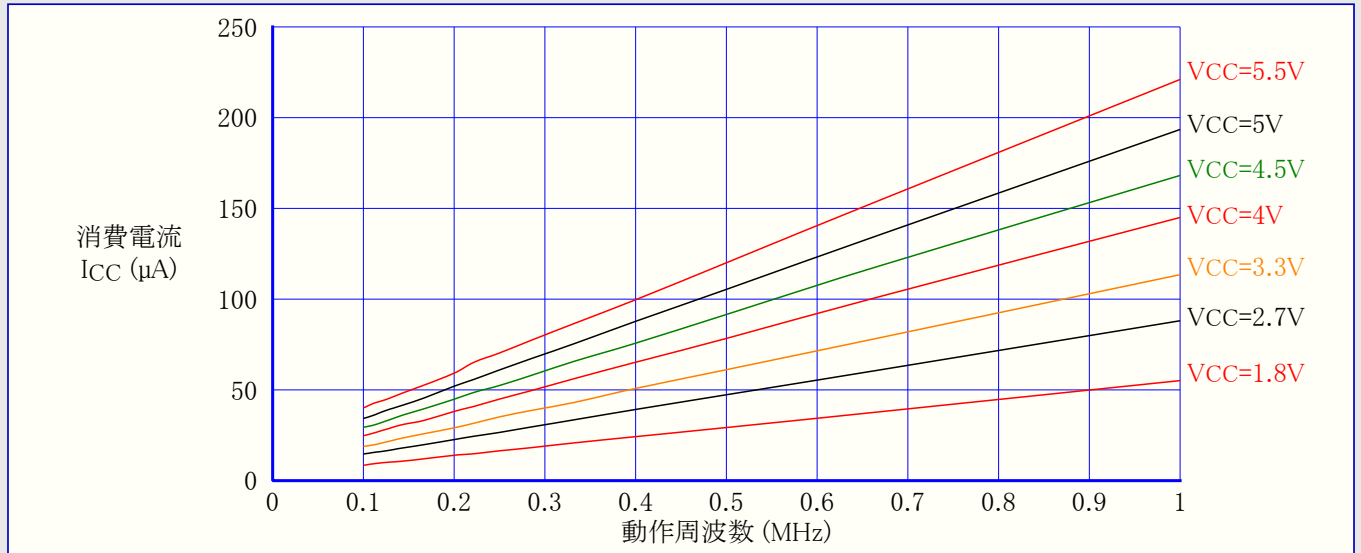


図27-52. リセット消費(供給)電流 対 周波数 (1MHz~20MHz)

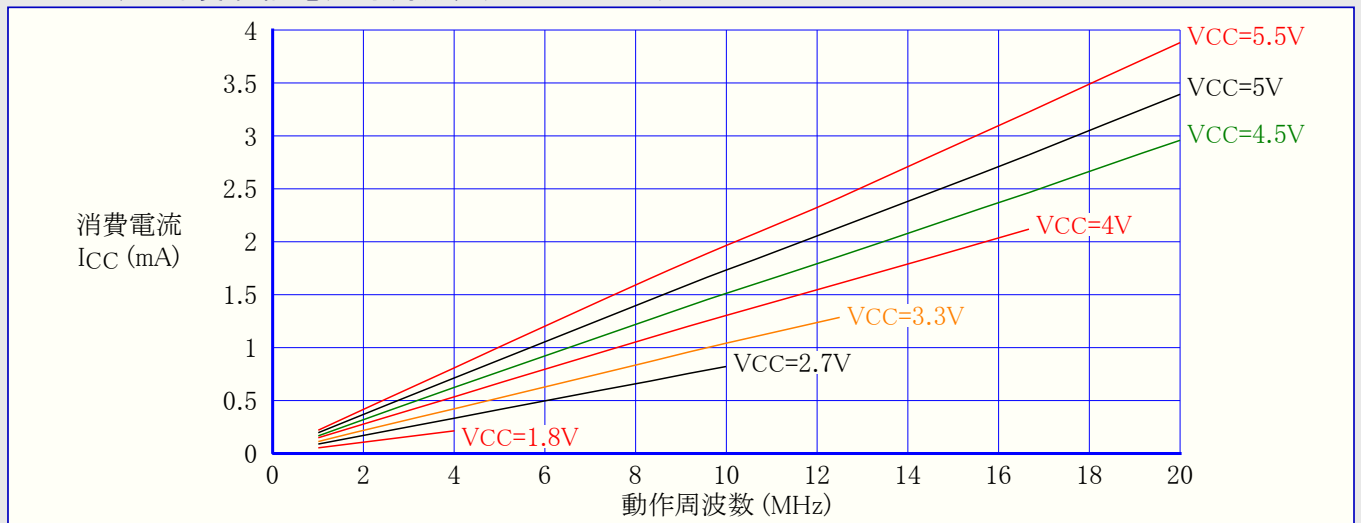
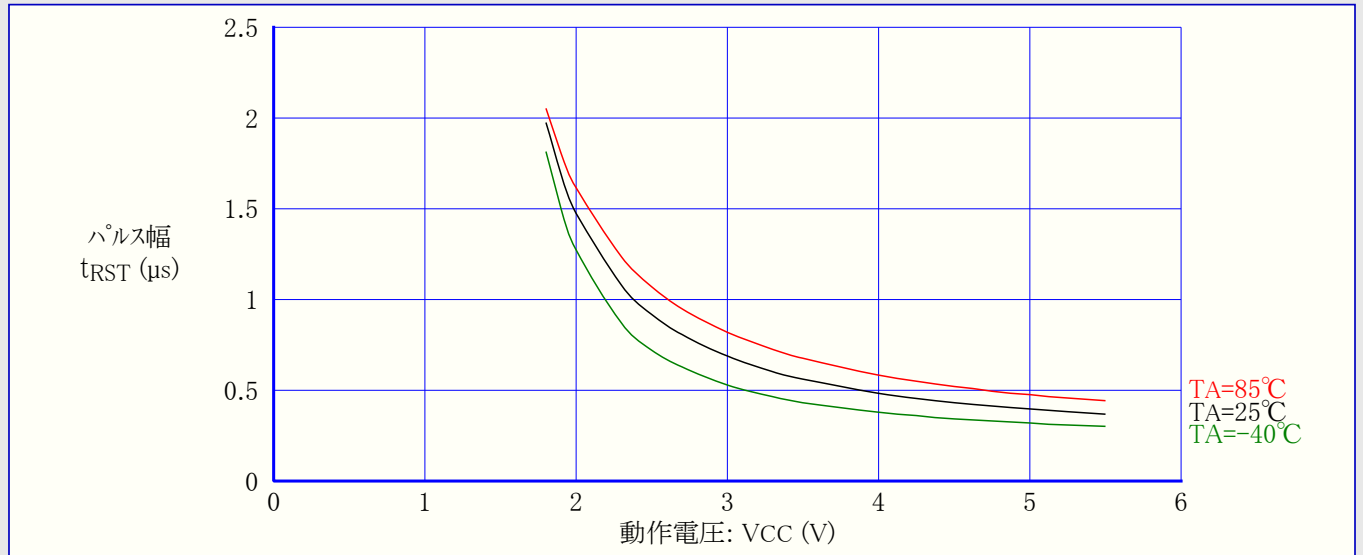




図27-53. リセット パルス幅 対 動作電圧



## 28. レジスタ要約

拡張I/Oレジスタ領域 (1/2)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
～\$FF) (\$E0	予約									
(\$DF)	予約									
(\$DE)	予約									
(\$DD)	PORTJ	－	PORTJ6	PORTJ5	PORTJ4	PORTJ3	PORTJ2	PORTJ1	PORTJ0	58
(\$DC)	DDRJ	－	DDJ6	DDJ5	DDJ4	DDJ3	DDJ2	DDJ1	DDJ0	58
(\$DB)	PINJ	－	PINJ6	PINJ5	PINJ4	PINJ3	PINJ2	PINJ1	PINJ0	58
(\$DA)	PORTH	PORTH7	PORTH6	PORTH5	PORTH4	PORTH3	PORTH2	PORTH1	PORTH0	57
(\$D9)	DDRH	DDH7	DDH6	DDH5	DDH4	DDH3	DDH2	DDH1	DDH0	57
(\$D8)	PINH	PINH7	PINH6	PINH5	PINH4	PINH3	PINH2	PINH1	PINH0	58
(\$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)	UDR0	USART データレジスタ								123
(\$C5)	UBRR0H	－	－	－	－	USART ホールレートレジスタ上位 (UBRR11～8)				123
(\$C4)	UBRR0L	USART ホールレートレジスタ下位 (UBRR7～0)								
(\$C3)	予約									
(\$C2)	UCSR0C	－	UMSEL0	UPM01	UPM00	USBS0	UCSZ01	UCSZ00	UCPOL0	124
(\$C1)	UCSR0B	RXCIE0	TXCIE0	UDRIE0	RXEN0	TXEN0	UCSZ02	RXB80	TXB80	124
(\$C0)	UCSR0A	RXC0	TXC0	UDRE0	FE0	DOR0	UPE0	U2X0	MPCM0	123
(\$BF)	予約									
(\$BE)	予約									
(\$BD)	予約									
(\$BC)	予約									
(\$BB)	予約									
(\$BA)	USIDR	USI データレジスタ								134
(\$B9)	USISR	USISIF	USIOIF	USIPF	USIDC	USICNT3	USICNT2	USICNT1	USICNT0	133
(\$B8)	USICR	USISIE	USIOIE	USIWM1	USIWM0	USICS1	USICS0	USICLK	USITC	133
(\$B7)	予約									
(\$B6)	ASSR	－	－	－	EXCLK	AS2	TCN2UB	OCR2UB	TCR2UB	103
(\$B5)	予約									
(\$B4)	予約									
(\$B3)	OCR2A	タイマ/カウンタ2 比較Aレジスタ								102
(\$B2)	TCNT2	タイマ/カウンタ2								102
(\$B1)	予約									
(\$B0)	TCCR2A	FOC2A	WGM20	COM2A1	COM2A0	WGM21	CS22	CS21	CS20	101
～\$AF) (\$90	予約									

注: 暗背景のレジスタとビットはATmega3250Pにだけ存在します。

## 拡張I/Oレジスタ領域 (2/2)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
(\$8F)	予約									
(\$8E)	予約									
(\$8D)	予約									
(\$8C)	予約									
(\$8B)	OCR1BH			タイマ/カウンタ1 比較Bレジスタ 上位バイト						89
(\$8A)	OCR1BL			タイマ/カウンタ1 比較Bレジスタ 下位バイト						
(\$89)	OCR1AH			タイマ/カウンタ1 比較Aレジスタ 上位バイト						89
(\$88)	OCR1AL			タイマ/カウンタ1 比較Aレジスタ 下位バイト						
(\$87)	ICR1H			タイマ/カウンタ1 捕獲レジスタ 上位バイト						89
(\$86)	ICR1L			タイマ/カウンタ1 捕獲レジスタ 下位バイト						
(\$85)	TCNT1H			タイマ/カウンタ1 上位バイト						88
(\$84)	TCNT1L			タイマ/カウンタ1 下位バイト						
(\$83)	予約									
(\$82)	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-	88
(\$81)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	87
(\$80)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10	86
(\$7F)	DIDR1	-	-	-	-	-	-	AIN1D	AIN0D	137
(\$7E)	DIDR0	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	147
(\$7D)	予約									
(\$7C)	ADMUX	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	145
(\$7B)	ADCSRB	-	ACME	-	-	-	ADTS2	ADTS1	ADTS0	136,146
(\$7A)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	146
(\$79)	ADCH	A/Dデータレジスタ上位バイト (ADC9〜8またはADC9〜2)								147
(\$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	-	PCINT30	PCINT29	PCINT28	PCINT27	PCINT26	PCINT25	PCINT24	38
(\$72)	予約									
(\$71)	予約									
(\$70)	TIMSK2	-	-	-	-	-	-	OCIE2A	TOIE2	104
(\$6F)	TIMSK1	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1	90
(\$6E)	TIMSK0	-	-	-	-	-	-	OCIE0A	TOIE0	71
(\$6D)	PCMSK2	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	38
(\$6C)	PCMSK1	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	38
(\$6B)	PCMSK0	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	38
(\$6A)	予約									
(\$69)	EICRA	-	-	-	-	-	-	ISC01	ISC00	36
(\$68)	予約									
(\$67)	予約									
(\$66)	OSCCAL	内蔵RC発振器 発振校正値レジスタ								21
(\$65)	予約									
(\$64)	PRR	-	-	-	-	PRTIM1	PRSPI	PRUSART0	PRADC	26
(\$63)	予約									
(\$62)	予約									
(\$61)	CLKPR	CLKPCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	21
(\$60)	WDTCSR	-	-	-	WDCE	WDE	WDP2	WDP1	WDP0	30

注: 暗背景のレジスタとビットはATmega3250Pでだけ利用可能です。

- 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操作し、故にこのような状態フラグを含むレジスタで使えます。CBIとSBI命令は\$00~\$1Fのレジスタだけで動作します。
  - I/O指定命令INとOUTを使う時はI/Oアドレス\$00~\$3Fが使われなければなりません。LDとST命令を使ってデータ空間としてI/Oレジスタをアドレス指定する時はこれらのアドレスに\$20が加算されなければなりません。ATmega325P/3250PはINとOUT命令で予約した64位置で支援できるよりも多くの周辺部(機能)の複合マイクロコントローラです。SRAM(データ空間)内の拡張I/O空間はST/STS/STDとLD/LDS/LDD命令だけが使えます。

## 標準I/Oレジスタ領域

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
\$3F (\$5F)	SREG	I	T	H	S	V	N	Z	C	8
\$3E (\$5E)	SPH	—	—	—	—	SP11	SP10	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	—	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	171
\$36 (\$56)	予約									
\$35 (\$55)	MCUCR	JTD	BODS	BODSE	PUD	—	—	IVSEL	IVCE	162,25,55,35
\$34 (\$54)	MCUSR	—	—	—	JTRF	WDRF	BORF	EXTRF	PORF	162,30
\$33 (\$53)	SMCR	—	—	—	—	SM2	SM1	SM0	SE	25
\$32 (\$52)	予約									
\$31 (\$51)	OCDR	IDRD/ OCDR7	OCDR6	OCDR5	OCDR4	OCDR3	OCDR2	OCDR1	OCDR0	151
\$30 (\$50)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	137
\$2F (\$4F)	予約									
\$2E (\$4E)	SPDR	SPI データレジスタ								110
\$2D (\$4D)	SPSR	SPIF	WCOL	—	—	—	—	—	SPI2X	109
\$2C (\$4C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	109
\$2B (\$4B)	GPIOR2	汎用I/Oレジスタ2								16
\$2A (\$4A)	GPIOR1	汎用I/Oレジスタ1								16
\$29 (\$49)	予約									
\$28 (\$48)	予約									
\$27 (\$47)	OCR0A	タイマ/カウンタ0 比較Aレジスタ								70
\$26 (\$46)	TCNT0	タイマ/カウンタ0								70
\$25 (\$45)	予約									
\$24 (\$44)	TCCR0A	FOC0A	WGM00	COM0A1	COM0A0	WGM01	CS02	CS01	CS00	69
\$23 (\$43)	GTCCR	TSM	—	—	—	—	—	PSR2	PSR10	103,71
\$22 (\$42)	EEARH	—	—	—	—	—	—	EEAR9	EEAR8	14
\$21 (\$41)	EEARL	EEPROMアドレスレジスタ下位バイト (EEAR7~0)								
\$20 (\$40)	EEDR	EEPROMデータレジスタ								14
\$1F (\$3F)	EECR	—	—	—	—	EERIE	EEMWE	EEWE	EERE	14
\$1E (\$3E)	GPIOR0	汎用I/Oレジスタ0								16
\$1D (\$3D)	EMSK	(PCIE3)	(PCIE2)	PCIE1	PCIE0	—	—	—	INT0	37
\$1C (\$3C)	EIFR	(PCIF3)	(PCIF2)	PCIF1	PCIF0	—	—	—	INTF0	37
\$1B (\$3B)	予約									
\$1A (\$3A)	予約									
\$19 (\$39)	予約									
\$18 (\$38)	予約									
\$17 (\$37)	TIFR2	—	—	—	—	—	—	OCF2A	TOV2	104
\$16 (\$36)	TIFR1	—	—	ICF1	—	—	OCF1B	OCF1A	TOV1	90
\$15 (\$35)	TIFR0	—	—	—	—	—	—	OCF0A	TOV0	71
\$14 (\$34)	PORTG	—	—	—	PORTG4	PORTG3	PORTG2	PORTG1	PORTG0	57
\$13 (\$33)	DDRG	—	—	—	DDG4	DDG3	DDG2	DDG1	DDG0	57
\$12 (\$32)	PING	—	—	PING5	PING4	PING3	PING2	PING1	PING0	57
\$11 (\$31)	PORTF	PORTF7	PORTF6	PORTF5	PORTF4	PORTF3	PORTF2	PORTF1	PORTF0	57
\$10 (\$30)	DDRF	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0	57
\$0F (\$2F)	PINF	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0	57
\$0E (\$2E)	PORTE	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	56
\$0D (\$2D)	DDRE	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	56
\$0C (\$2C)	PINE	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0	56
\$0B (\$2B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	56
\$0A (\$2A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	56
\$09 (\$29)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	56
\$08 (\$28)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	55
\$07 (\$27)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	56
\$06 (\$26)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	56
\$05 (\$25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	55
\$04 (\$24)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	55
\$03 (\$23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	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	55

注: ( ) 付きのビットはATmega3250Pでのみ利用できます。

(訳注) 原書本位置の注意は前頁に移動しました。

## 29. 命令要約

ニーモニック	オペランド	意味	動作	フラグ	クロック
算術、論理演算命令					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,C	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
MUL	Rd,Rr	符号なし間の乗算	$R1:R0 \leftarrow Rd \times Rr$ (U×U)	I,T,H,S,V,N,Z,C	2
MULS	Rd,Rr	符号付き間の乗算	$R1:R0 \leftarrow Rd \times Rr$ (S×S)	I,T,H,S,V,N,Z,C	2
MULSU	Rd,Rr	符号付きと符号なしの乗算	$R1:R0 \leftarrow Rd \times Rr$ (S×U)	I,T,H,S,V,N,Z,C	2
FMUL	Rd,Rr	符号なし間の固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (U×U)	I,T,H,S,V,N,Z,C	2
FMULS	Rd,Rr	符号付き間の固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (S×S)	I,T,H,S,V,N,Z,C	2
FMULSU	Rd,Rr	符号付きと符号なしの固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (S×U)	I,T,H,S,V,N,Z,C	2
分岐命令					
RJMP	k	相対無条件分岐	$PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	2
IJMP		Zレジスタ間接無条件分岐	$PC \leftarrow Z$	I,T,H,S,V,N,Z,C	2
JMP	k	絶対無条件分岐	$PC \leftarrow k$	I,T,H,S,V,N,Z,C	3
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
CALL	k	絶対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow k$	I,T,H,S,V,N,Z,C	4
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
SBRB	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	$(N \text{ EOR } V)=0$ 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	$(N \text{ EOR } V)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	$H=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	$H=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	$T=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	$T=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	$V=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	$V=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	$I=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	$I=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ  
b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)



ニーモニック	オペランド	意味	動作	フラグ	クロック
データ移動命令					
MOV	Rd,Rr	汎用レジスタ間の複写	$Rd \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
MOVW	Rd,Rr	汎用レジスタ対間の複写	$Rd+1:Rd \leftarrow Rr+1:Rr$	I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	$Rd \leftarrow K$	I,T,H,S,V,N,Z,C	1
LD	Rd,X	Xレジスタ間接での取得	$Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,X+	事後増加付きXレジスタ間接での取得	$Rd \leftarrow (X), X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-X	事前減少付きXレジスタ間接での取得	$X \leftarrow X - 1, Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y	Yレジスタ間接での取得	$Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上 (任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上 (事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
ビット関係命令					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	1,1,1,1,1,1,1,1	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0,0	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,1	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)	$T \leftarrow 0$	0,0,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
MCU制御命令					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグ機能専用	I,T,H,S,V,N,Z,C	N/A

### 30. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号 (注2)	外囲器	動作範囲
ATmega325P	10 (注3)	1.8~5.5V	ATmega325PV-10AU	64A	工業用 (-40°C~85°C)
			ATmega325PV-10MU	64M1	
	20 (注3)	2.7~5.5V	ATmega325P-20AU	64A	
			ATmega325P-20MU	64M1	
ATmega3250P	10 (注3)	1.8~5.5V	ATmega3250PV-10AU	100A	工業用 (-40°C~85°C)
	20 (注3)	2.7~5.5V	ATmega3250P-20AU		

**注:** このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

**注2:** 有害物質使用制限に関する欧州指令(RoHS指令)適合の鉛フリー製品。またハロゲン化合物フリーで完全に安全です。

**注3:** 速度と電源電圧の関係については194頁の図26-1.と図26-2.をご覧ください。

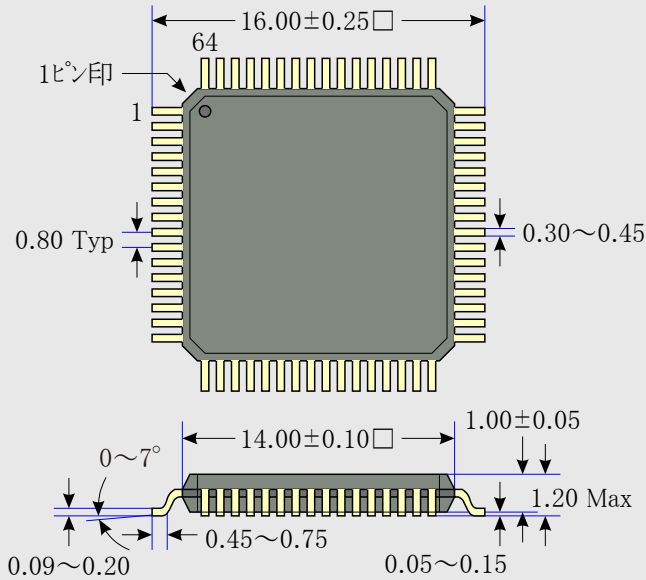
外囲器形式	
64A	64リット 14×14×1.0mm厚 薄型プラスチック4方向平板外囲器 (TQFP)
64M1	64パッド 9×9×1mm 0.5mmピッチ 4方向平板リットなし/小リット枠外囲器 (QFN/MLF)
100A	100リット 14×14×1mm 0.5mmピッチ 薄型プラスチック4方向平板外囲器 (TQFP)

## 31. 外圍器情報

### 31.1. 64A

64リード 0.8mmピッチ プラスティック4方向平板外圍器 (TQFP)

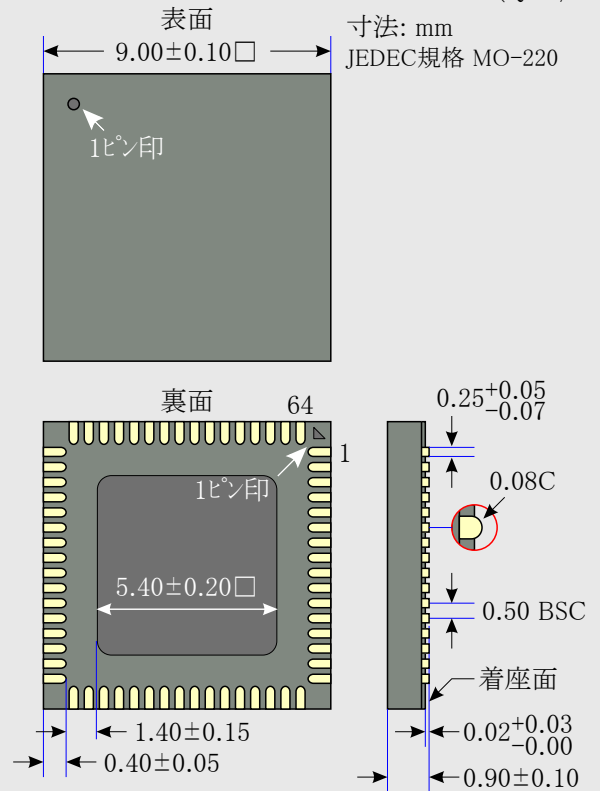
寸法: mm  
JEDEC規格 MS-026 AEB



### 31.2. 64M1

64パッド 0.5mmピッチ 4方向平板リードなし/小リード枠外圍器 (QFN/MLF)

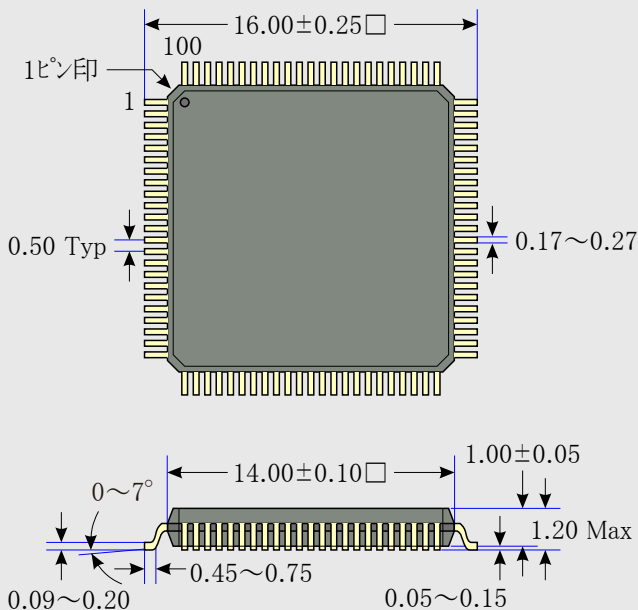
寸法: mm  
JEDEC規格 MO-220



### 31.3. 100A

100リード 0.5mmピッチ プラスティック4方向平板外圍器 (TQFP)

寸法: mm  
JEDEC規格 MS-026 AED



## 32. 障害情報

この章の改訂番号はATmega325P/3250P(V)デバイスの改訂版を参照してください。

- 非同期タイマでタイマレジスタ書き込み時に割り込みが失われる可能性 325P-A/B/C, 3250P-A/B/C
- BOD禁止(設定)使用がチップリセットを行う 325P-A, 3250P-A

### 1. 非同期タイマ/カウンタでタイマレジスタ書き込み時に割り込みが失われる可能性 (325P-A/B/C, 3250P-A/B/C)

非同期タイマ/カウンタレジスタ(TCNTx)が\$00の時に同期タイマ/カウンタクロックのタイマ/カウンタレジスタが書かれる場合に割り込みが失われるでしょう。

#### 対策/対処

非同期タイマ/カウンタ制御レジスタ(TCCR<sub>x</sub>)、非同期タイマ/カウンタ(TCNT<sub>x</sub>)レジスタまたは非同期タイマ/カウンタ比較レジスタ(OCR<sub>x</sub>)を書く前に、常に非同期タイマ/カウンタレジスタが\$FFまたは\$00のどちらの値でもないことを調べてください。

### 2. BOD禁止(設定)使用がチップリセットを行う (325P-A, 3250P-A)

BOD禁止任意選択許可でのBOD(動作)OFFでデバイスが休止へ移行すると、起動でBODリセットが生成されてチップがリセットします。

#### 対策/対処

BOD禁止を使わないでください。

### 33. データシート改訂履歴

この章内の参照頁番号はこの文書が参照されていることに注意してください。この章内の改訂番号は文書の改訂番号を参照してください。

#### 33.1. 8023A – 2006年12月

1. 初版

#### 33.2. 8023B – 2007年8月

1. 1頁の「特徴」を更新
2. 174頁の表24-6.でデバイスとJTAG IDを更新
3. 195頁の「システムとリセットの特性」を更新
4. 198頁の「代表特性」を更新

#### 33.3. 8023C – 2007年8月

1. 224頁の「障害情報」を更新

#### 33.4. 8023D – 2007年8月

1. 6頁に「データ保持力」を追加
2. 174頁の「識別バイト」でデバイスとJTAGのIDを更新

#### 33.5. 8023E – 2008年6月

1. 9頁の「スタック ポインタ」の記述を更新
2. 18頁の「低周波数クリスタル用発振器」の記述を更新
3. 36頁の「外部割り込み」の「概要」項に注を追加
4. 55頁の「ポートA出力レジスタ(PORTA)」のアドレスを更新
5. 99頁の「タイマ/カウンタ2の非同期動作」で第5公示を更新
6. 195頁の「システムとリセットの特性」を更新
7. 224頁の「障害情報」に改訂Cに対する障害情報を追加

#### 33.6. 8023F – 2009年7月

1. 224頁の「障害情報」を更新
2. Atmelの新住所で最終頁を更新



## 目次

特徴	1	11.1. 概要	32
1. ピン配置	2	11.2. ATmega325P/3250Pの割り込みベクタ	32
1.1. お断り	3	11.3. ベクタ移動用レジスタ	35
2. 概要	3	12. 外部割り込み	36
2.1. 構成図	3	12.1. 概要	36
2.2. ATmega325PとATmega3250P間の違い	4	12.2. ピン変化割り込みタイミング	36
2.3. ピン説明	4	12.3. 外部割り込み用レジスタ	36
3. 資料	6	13. 入出力ポート	39
4. データ保持力	6	13.1. 概要	39
5. コード例について	6	13.2. 標準デジタル入出力としてのポート	39
6. AVR CPU コア	7	13.3. 交換ポート機能	42
6.1. 概要	7	13.4. I/Oポート用レジスタ	55
6.2. 構造概要	7	14. 8ビット タイマ/カウンタ0 (PWM)	59
6.3. ALU (Arithmetic Logic Unit)	7	14.1. 特徴	59
6.4. ステータス レジスタ	8	14.2. 概要	59
6.5. 汎用レジスタ ファイル	9	14.3. タイマ/カウンタのクロック	60
6.6. スタック ポインタ	9	14.4. 計数器部	60
6.7. 命令実行タイミング	10	14.5. 比較出力部	61
6.8. リセットと割り込みの扱い	10	14.6. 比較一致出力部	62
7. AVR メモリ	12	14.7. 動作種別	63
7.1. 概要	12	14.8. タイマ/カウンタのタイミング	66
7.2. 実装書き換え可能なプログラム用フラッシュ メモリ	12	14.9. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器	67
7.3. データ用SRAMメモリ	12	14.10. 8ビット タイマ/カウンタ0用レジスタ	69
7.4. データ用EEPROMメモリ	13	15. 16ビット タイマ/カウンタ1	72
7.5. I/Oメモリ (レジスタ)	13	15.1. 特徴	72
7.6. メモリ関係レジスタ	14	15.2. 概要	72
8. システム クロックとクロック選択	17	15.3. 16ビット レジスタのアクセス	74
8.1. クロック系統とその配給	17	15.4. タイマ/カウンタのクロック	76
8.2. クロック元	18	15.5. 計数器部	76
8.3. 既定のクロック元	18	15.6. 捕獲入力部	77
8.4. クリスタル用発振器	18	15.7. 比較出力部	78
8.5. 低周波数クリスタル用発振器	19	15.8. 比較一致出力部	80
8.6. 校正付き内蔵RC発振器	19	15.9. 動作種別	81
8.7. 外部クロック信号	20	15.10. タイマ/カウンタのタイミング	85
8.8. システム クロック出力緩衝部	20	15.11. 16ビット タイマ/カウンタ1用レジスタ	86
8.9. タイマ/カウンタ用発振器	20	16. 8ビット タイマ/カウンタ2 (PWM,非同期動作)	91
8.10. システム クロック前置分周器	20	16.1. 特徴	91
8.11. クロック関係用レジスタ	21	16.2. 概要	91
9. 電力管理と休止形態	22	16.3. タイマ/カウンタのクロック	92
9.1. 休止形態種別	22	16.4. 計数器部	92
9.2. 低電圧検出器(BOD)禁止	22	16.5. 比較出力部	93
9.3. アイドル動作	22	16.6. 比較一致出力部	94
9.4. A/D変換雑音低減動作	23	16.7. 動作種別	95
9.5. パワーダウン動作	23	16.8. タイマ/カウンタのタイミング	98
9.6. パワーセーブ動作	23	16.9. タイマ/カウンタ2の非同期動作	99
9.7. スタンバイ動作	23	16.10. タイマ/カウンタ2の前置分周器	100
9.8. 電力削減(電力削減レジスタ)	23	16.11. 8ビット タイマ/カウンタ2用レジスタ	101
9.9. 消費電力の最小化	24	17. SPI (直列周辺インターフェース)	105
9.10. 電力管理用レジスタ	25	17.1. 特徴	105
10. システム制御とリセット	27	17.2. 概要	105
10.1. AVRのリセット	27	17.3. SSピンの機能	107
10.2. リセット元	27	17.4. データ転送形式	108
10.3. 内部基準電圧	29	17.5. SPI用レジスタ	109
10.4. ウォッチドッグ タイマ	29	18. USART	111
10.5. リセット関係用レジスタ	30	18.1. 特徴	111
11. 割り込み	32	18.2. 概要	111
		18.3. クロック生成	113

18.4.	フレーム形式	114	25.	メモリプログラミング	172
18.5.	USARTの初期化	115	25.1.	プログラムメモリとデータメモリ用施錠ビット	172
18.6.	USARTのデータ送信	116	25.2.	ヒューズビット	173
18.7.	USARTのデータ受信	117	25.3.	識票バイト	174
18.8.	非同期受信	120	25.4.	校正バイト	174
18.9.	複数プロセッサ通信動作	122	25.5.	並列プログラミング	175
18.10.	USART用レジスタ	123	25.6.	並列プログラミング手順	176
18.11.	ボーレート設定例	126	25.7.	直列プログラミング	183
19.	多用途直列インターフェース (USI)	128	25.8.	直列プログラミング手順	183
19.1.	特徴	128	25.9.	JTAGインターフェース経由プログラミング	186
19.2.	概要	128	26.	電気的特性	193
19.3.	機能説明	129	26.1.	絶対最大定格	193
19.4.	USIでの代替使用	132	26.2.	DC特性	193
19.5.	USI用レジスタ	133	26.3.	速度勾配	194
20.	アナログ比較器	136	26.4.	クロック特性	194
20.1.	概要	136	26.5.	システムとリセットの特性	195
20.2.	アナログ比較器入力選択	136	26.6.	SPI タイミング特性	196
20.3.	アナログ比較器用レジスタ	136	26.7.	A/D変換器特性 (暫定)	197
21.	A/D変換器	138	27.	代表特性	198
21.1.	特徴	138	27.1.	活動動作消費電流	198
21.2.	概要	138	27.2.	アイドル動作消費電流	199
21.3.	操作	139	27.3.	周辺機能部供給電流	201
21.4.	変換の開始	139	27.4.	パワーダウン動作消費電流	201
21.5.	前置分周と変換タイミング	140	27.5.	パワーセーフ動作消費電流	202
21.6.	チャネル変更と基準電圧選択	142	27.6.	スタンバイ動作消費電流	202
21.7.	雑音低減機能	143	27.7.	ピンプルアップ	202
21.8.	A/D変換の結果	144	27.8.	ピン駆動能力	204
21.9.	A/D変換用レジスタ	145	27.9.	ピン 閾値とヒステリシス	208
22.	JTAGインターフェースと内蔵デバッグ機能	148	27.10.	内部発振器周波数	211
22.1.	特徴	148	27.11.	周辺機能部消費電流	213
22.2.	概要	148	27.12.	リセット消費電流とリセットパルス幅	215
22.3.	検査入出力ポート (TAP:Test Access Port)	149	28.	レジスタ要約	217
22.4.	TAP制御器	149	29.	命令要約	220
22.5.	境界走査チェーン(Boundary-Scan Chain)の使用	150	30.	注文情報	222
22.6.	内蔵デバッグ機能の使用	150	31.	外圍器情報	223
22.7.	内蔵デバッグ特殊JTAG命令	150	32.	障害情報	224
22.8.	JTAGプログラミング能力の使用	150	33.	データシート改訂履歴	225
22.9.	参考文献	151			
22.10.	内蔵デバッグ用レジスタ	151			
23.	IEEE1149.1(JTAG)境界走査	152			
23.1.	特徴	152			
23.2.	概要	152			
23.3.	データレジスタ	152			
23.4.	境界走査(Boundary-Scan)用JTAG命令	153			
23.5.	境界走査チェーン(Boundary-Scan Chain)	154			
23.6.	ATmega325P/3250Pの境界走査順	159			
23.7.	境界走査記述言語(BSDL)ファイル	161			
23.8.	JTAG用レジスタ	162			
24.	ブートローダ支援 - RWW 自己プログラミング	163			
24.1.	特徴	163			
24.2.	概要	163			
24.3.	フラッシュメモリの応用領域とブートローダ領域	163			
24.4.	書き中に読みが可能な領域と不能な領域	163			
24.5.	ブートローダ施錠ビット	165			
24.6.	ブートローダプログラムへの移行	165			
24.7.	自己プログラミングでのフラッシュアドレス指定	166			
24.8.	フラッシュメモリの自己プログラミング	166			
24.9.	ブートローダ用レジスタ	171			



## 本社

### Atmel Corporation

2325 Orchard Parkway  
San Jose, CA 95131  
USA  
TEL 1(408) 441-0311  
FAX 1(408) 487-2600

## 国外営業拠点

### Atmel Asia

Unit 1-5 & 16, 19/F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
Hong Kong  
TEL (852) 2245-6100  
FAX (852) 2722-1369

### Atmel Europe

Le Krebs  
8, Rue Jean-Pierre Timbaud  
BP 309  
78054 Saint-Quentin-en-  
Yvelines Cedex  
France  
TEL (33) 1-30-60-70-00  
FAX (33) 1-30-60-71-11

### Atmel Japan

104-0033 東京都中央区  
新川1-24-8  
東熱新川ビル 9F  
アトメル ジャパン株式会社  
TEL (81) 03-3523-3551  
FAX (81) 03-3523-7581

## 製品窓口

### ウェブサイト

[www.atmel.com](http://www.atmel.com)

### 技術支援

[avr@atmel.com](mailto:avr@atmel.com)

### 販売窓口

[www.atmel.com/contacts](http://www.atmel.com/contacts)

### 文献請求

[www.atmel.com/literature](http://www.atmel.com/literature)

**お断り:** 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイト位置する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえば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、または利益の損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使われるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© Atmel Corporation 2009. 不許複製 Atmel®、ロゴとそれらの組み合わせ、AVR®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

## © HERO 2023.

本データシートはAtmelのATmega325P/3250P英語版データシート(改訂8023F-07/09)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力は、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には( )内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。。

若干の構成変更により、一部の節、項番号が異なります。図表番号が原書に対して修正されています。