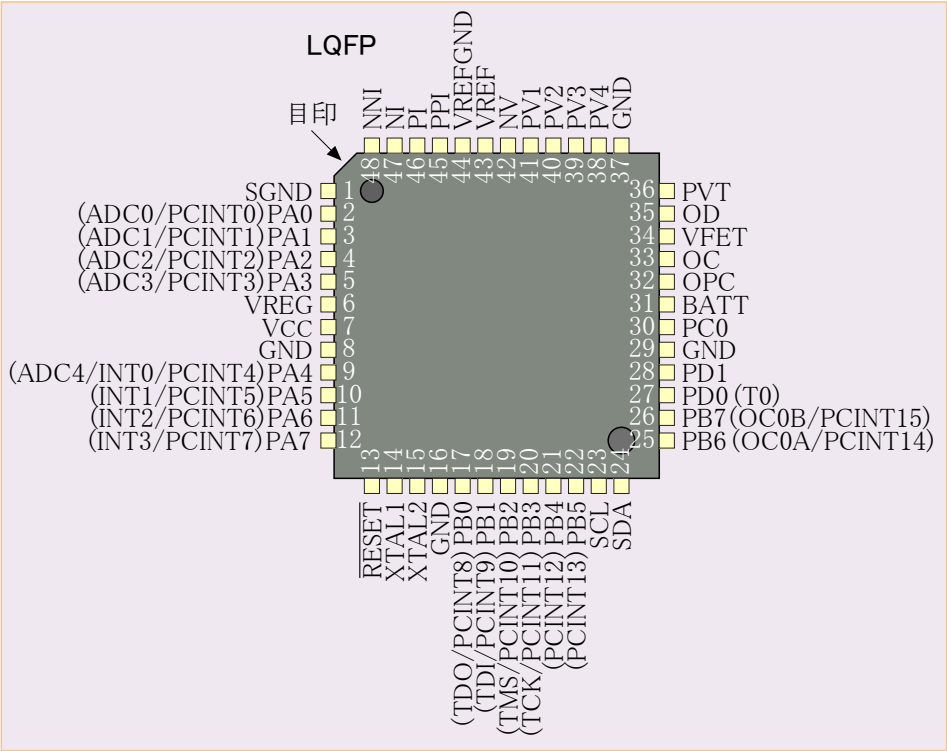


特徴

- 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
 - 完全なスタティック動作
 - 1MHz時、1MIPSに達する高速動作
- データ メモリと不揮発性プログラム メモリ
 - 実装自己書き換え可能な40Kバイト(20K語)フラッシュ メモリ内蔵 (10,000回書き換え可能)
 - 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
 - 512バイトのEEPROM (100,000回の書き換えが可能)
 - 2Kバイトの内蔵SRAM
 - ソフトウェア保護用の設定可能な施錠機能
- 内蔵デバッグ
 - 広範囲な内蔵デバッグ機能
 - JTAGインターフェースを通じて利用可能
- 電池管理機能
 - 直列2、3、4セル
 - 最低電圧保護
 - 過電流保護 (充電と放電)
 - 短絡保護 (放電)
 - セル平衡FETを統合(内蔵)
 - 充電/予備充電/放電FET駆動用高電圧出力
- 内蔵周辺機能
 - 独立した前置分周器、比較、PWM機能付き1つの8ビット タイマ/カウンタ
 - 独立した前置分周器、比較機能付き1つの16ビット タイマ/カウンタ
 - 外部8と内部2入力の12ビット 電圧A/D変換器
 - 電流測定用高分解能クーロン カウンタA/D変換器
 - SMBus用2線直列インターフェース(TWI)
 - 設定可能な起動タイマ
 - 設定可能なウォッチドッグ タイマ
- 特殊マイクロ コントローラ機能
 - 電源ONリセット回路
 - 電圧調整器内蔵
 - 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、パワーオフの5つの低消費動作
- 外周器 : 48ピンLQFP
- 動作電圧 : 4.0~25V
- 最大耐電圧(高電圧ピン) : 28V
- 温度範囲 : -30~85°C
- 動作周波数 : 1MHz

1. ピン配置



1.1. お断り

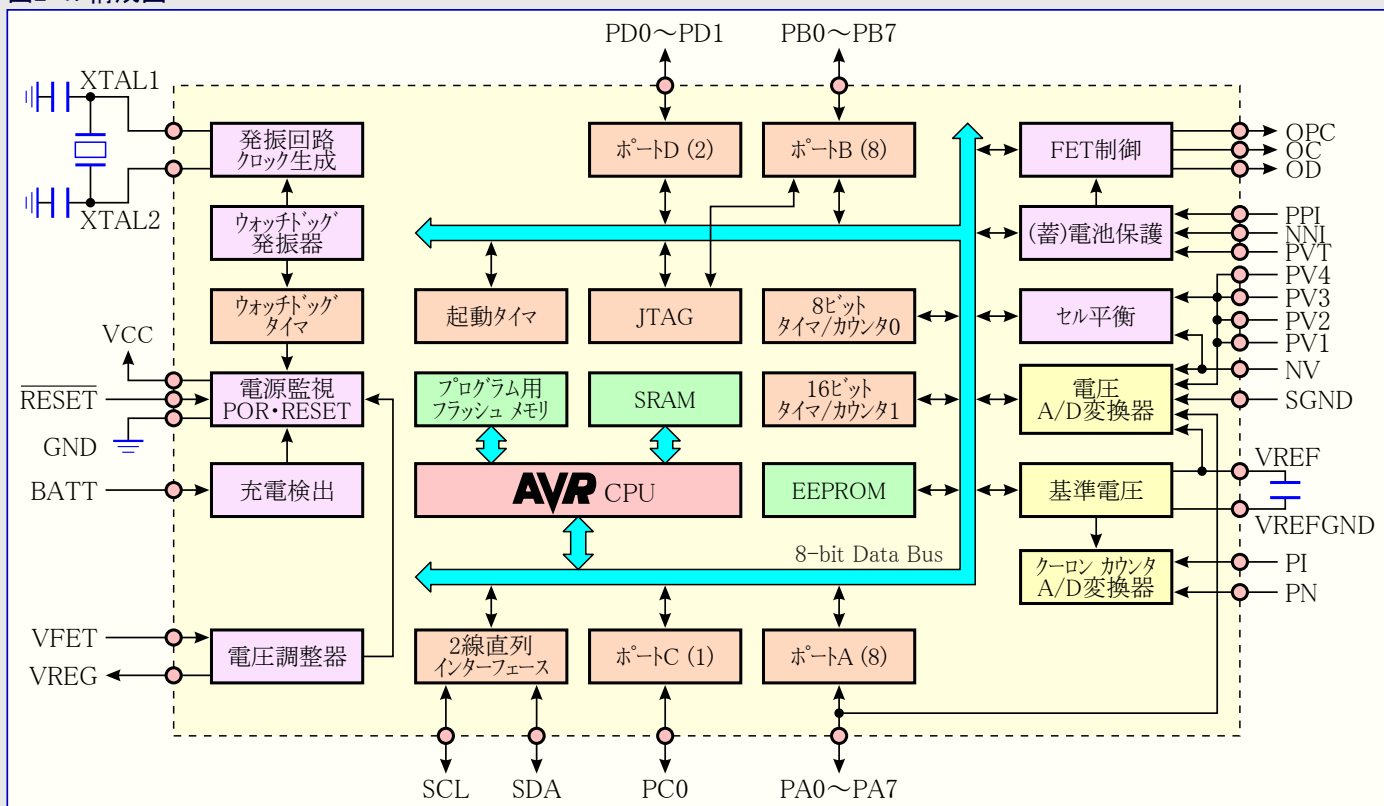
本データシート内で示された代表値はシミュレーションと同じ製法技術で製造された他のAVRマイクロ コントローラの特性に基づきます。最小と最大の値はデバイスの特性が記載された後に利用可能になります。

2. 概要

ATmega406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1MIPS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. 構成図



ATmega406は電圧調整器、専用の電池保護回路、統合(内蔵)されたセル平衡FET、高電圧アナログ前置回路、電池計量用の2つの内蔵基準電圧付きA/D変換器を含むMCUを提供します。

電圧調整器は4.0～25Vの広い電圧範囲で動作します。この電圧は集積論理回路とアナログ機能用へ公称3.3Vの電圧を定常的に供給するために安定化されます。

電池保護は不正状態を検知するために電池電圧と充放電電流を監視し、必要な時にそれらから電池を保護します。この不正状態は放電中の最低電圧と短絡、それと充放電中の過電流です。

統合(内蔵)されたセル平衡FETはソフトウェアで実装されるべきセル平衡算法を許します。

MCUは書き込み中読み出し可能な能力の40Kバイトの実装書き換え可能なフラッシュメモリと512バイトのEEPROM、2KバイトのSRAM、32個の汎用レジスタ、18本の汎用入出力線、11本の高電圧入出力線、内蔵デバッグ支援とプログラミング用のJTAG検査インターフェース、比較動作とPWMを含む柔軟性のある2つのタイマ/カウンタ、1つの起動タイマ、SMBus適合の2線直列インターフェース(TWI)、内部及び外部の割り込み、電圧と温度計測用の12ビット $\Delta\Sigma$ /D変換器、クーロン計測と瞬時電流計測用の高分解能 $\Delta\Sigma$ /D変換器、設定変更可能な内部発振器付きウォッチドッグタイマ、ソフトウェアで選択できる4つの低消費動作機能を提供します。

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

アイドル動作はCPUを止める一方で機能を継続するためにその他のチップ機能を許します。**パワーダウン動作**は電圧調整器、電池保護、電源電流検出、ウォッチドッグ タイマ、起動タイマを許す一方、以降のハードウェア リセットか割り込みまで他の全チップ機能を禁止します。**パワーセーブ動作**では起動タイマとクーロン カウンタ A/D変換器が走行を継続します。

本デバイスにはAtmelの高電圧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またはAVRコア上のブートプログラムの実行によって再書き込みができます。ブ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もどのインターフェースでも使えます。ブ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自己書き換え可能なフラッシュメモリ、セル平衡FET、電圧調整器と8ビットRISC型CPUの組み合わせによるAtmel ATmega406はリチウムイオン(Li-ion)電池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。

ATmega406 AVRはCコンパイラ、マクロアセンブラ、プログラムデバッガ/シミュレータ、内蔵デバッガ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. ピン説明

2.2.1. VFET

高電圧電源ピン。このピンは75頁の「[電圧調整器](#)」で記述される内蔵電圧調整器に対する供給に使われます。このピンに加えられる電圧値は最低電圧保護に関して電池保護回路によって監視されます。詳細については82頁の「[電池保護](#)」をご覧ください。

2.2.2. VCC

デジタル供給電圧(電源)。通常、VREGに接続されます。

2.2.3. VREG

内蔵電圧調整器出力。安定な調整器動作を保証するために外部デカップ(コンデンサ)を使ってください。詳細については75頁の「[電圧調整器](#)」をご覧ください。

2.2.4. VREF

外部デカップ用内部基準電圧。詳細については80頁の「[基準電圧と温度感知器](#)」をご覧ください。

2.2.5. VREFGND

内部基準電圧デカップ用接地。詳細については80頁の「[基準電圧と温度感知器](#)」をご覧ください。

2.2.6. GND

接地。

2.2.7. SGND

電圧A/D変換に関して基準として使われる信号接地ピン。詳細については76頁の「[電圧A/D変換器 \(10チャネル汎用12ビット \$\Delta\Sigma\$ A/D変換器\)](#)」をご覧ください。

2.2.8. PA7~PA0 (ポートA)

PA3~0は電圧A/D変換器へのアナログ入力として扱います。

ポートAは(ビット毎に選択される)内蔵プルアップ抵抗付き低電圧8ビット双方向入出力ポートも扱います。入力するとき、プルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートAピンに吐き出し電流が流れます。リセット条件が有効になると、クロックが動作していなくても、ポートAピンはHi-Zになります。

ポートAは44頁の「[ポートAの交換機能](#)」で一覧されるATmega406の様々な特殊機能も扱います。

2.2.9. PB7~PB0 (ポートB)

ポートBは(ビット毎に選択される)内蔵プルアップ抵抗付きの低電圧8ビット双方向入出力ポートです。入力するとき、プルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートBピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動作していなくても、ポートBピンはHi-Zになります。

ポートBは45頁の「[ポートBの交換機能](#)」で一覧されるATmega406の様々な特殊機能も扱います。

2.2.10. PC0 (ポートC)

ポートCは高電圧オープントレイン出力ポートです。

2.2.11. PD1,PD0 (ポートD)

ポートDは(ビット毎に選択される)内蔵プルアップ抵抗付きの低電圧2ビット双方向入出力ポートです。入力するとき、プルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートDピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動作していなくても、ポートDピンはHi-Zになります。

ポートDは46頁の「[ポートDの交換機能](#)」で一覧されるATmega406の様々な特殊機能も扱います。

2.2.12. SCL

SMBusクロック、オープントレイン双方向ピン。

2.2.13. SDA

SMBusデータ、オープントレイン双方向ピン。

2.2.14. OC,OD,OPC

外部の充電/放電/予備充電FETの駆動用高電圧出力。詳細については88頁の「[FET制御](#)」をご覧ください。

2.2.15. PPI,NNI

過電流と短絡検出に関して電池保護回路によって使われる、外部電流検出抵抗器からの濾波されない正(+)/負(-)入力です。詳細については82頁の「[電池保護](#)」をご覧ください。

2.2.16. PI, NI

電池パック内を流れる充放電電流を測定するためにクーロン カウンタA/D変換器によって使われる、濾波した正(+)/負(-)入力です。詳細については70頁の「[クーロン カウンタ \(容量/残量計専用 \$\Delta\Sigma\$ A/D変換器\)](#)」をご覧ください。

2.2.17. NV, PV1, PV2, PV3, PV4

NV, PV1, PV2, PV3, PV4は各セル電圧を測定するために電圧A/D変換器によって使われる、電池セル1,2,3,4用の入力です。

2.2.18. PVT

PVTはOD出力用のプルアップ電圧を定義します。

2.2.19. BATT

充電器接続時の検出用入力。このピンはOCとOPC出力用のプルアップ電圧も定義します。

2.2.20. RESET

リセット入力。クロックが動作してなくても、最小パルス幅より長い本ピンのLowレベルはリセットを生成します。最小パルス幅は149頁の表30-6.で与えられます。より短いパルスはリセットの生成が保証されません。

2.2.21. XTAL1

発振器反転増幅器への入力。

2.2.22. XTAL2

発振器反転増幅器からの出力。

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. コード例について

この資料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。

5. AVR CPU コア

5.1. 序説

本節は一般的な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。

5.2. 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、本項内で後述されます。

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

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

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構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

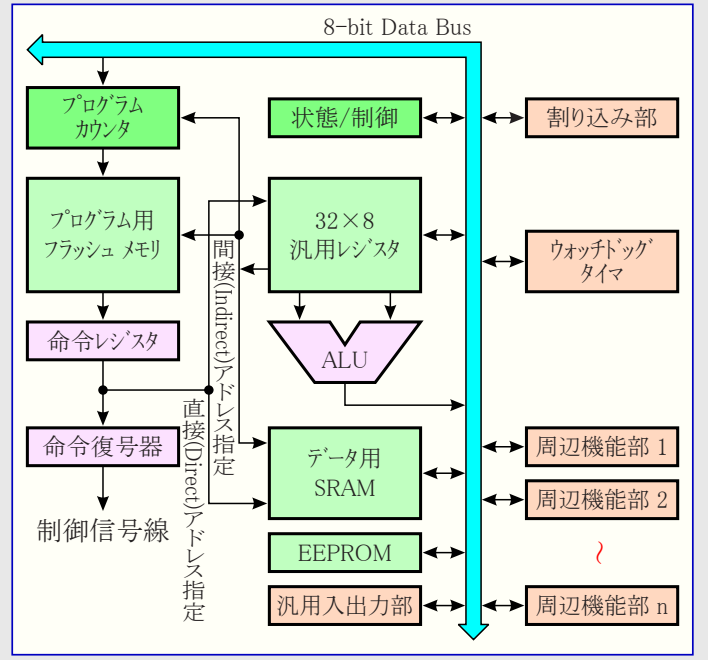
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(I)ビットがあります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタと他の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ATmega406にはST/STS/STDとLD/LDS/LDD命令だけが使えるSRAM内の\$60～\$FFに拡張I/O空間があります。

5.3. 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5-1. AVR MCU構造



5.4. AVR 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「[命令一式参考書](#)」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。

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

AVRのステータスレジスタ(SREG)は次のように定義されます。

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

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

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

● ビット6 – T: ビット変数 (Bit Copy Storage)

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

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

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

5.5. 汎用レジスタ ファイル

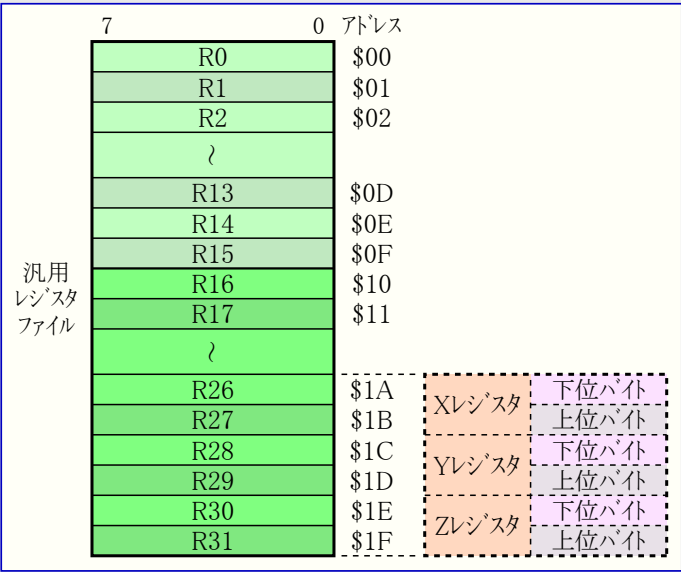
このレジスタ ファイルはAVRの増強したRISC命令群用に最適化されています。必要な効率と柔軟性を達成するために、次の入出力機構がレジスタ ファイルによって支援されます。

- ・ 1つの8ビット出力オペラントと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トと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トと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トと1つの16ビットの結果入力

図5-2.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5-2.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5-2. AVR CPU 汎用レジスタ構成図



5.5.1. 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5-3.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「命令一式参考書」をご覧ください)。

図5-3. X,Y,Zレジスタ構成図

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

5.6. スタック ポインタ

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

スタック ポインタはサブルーチンや割り込みのスタックが配置されるデータSRAMのスタック領域を指し示します。データSRAM内のスタック空間はサブルーチン呼び出しの実行や割り込みの許可の何れにも先立ってプログラムによって定義されなければなりません。スタック ポインタは\$100以上を指示するように設定されなければなりません。スタック ポインタはPUSH命令でデータがスタックに格納されると-1され、サブルーチン呼び出しや割り込みで戻りアドレスがスタックに格納されると-2されます。スタック ポインタはPOP命令でデータがスタックから引き出されると+1され、サブルーチンからの復帰(RET)命令や割り込みからの復帰(RETI)命令でアドレスがスタックから引き出されると+2されます。

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

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

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$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	

5.7. 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)用に選択したクロック元から直接的に生成したCPUクロック(clkCPU)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5-4.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1MIPSを達成するための基本的なパイプラインの概念です。

図5-5.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5-4. 命令の取得と実行の並列動作

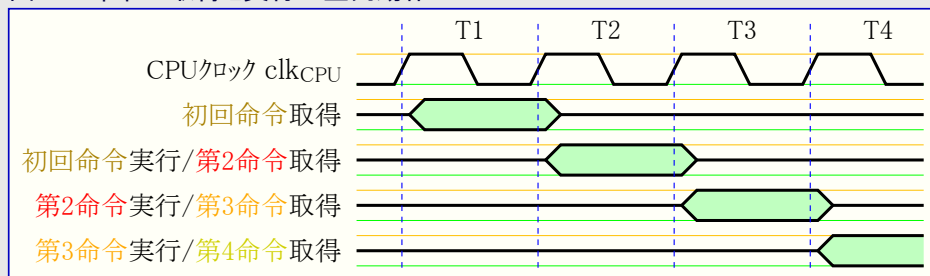
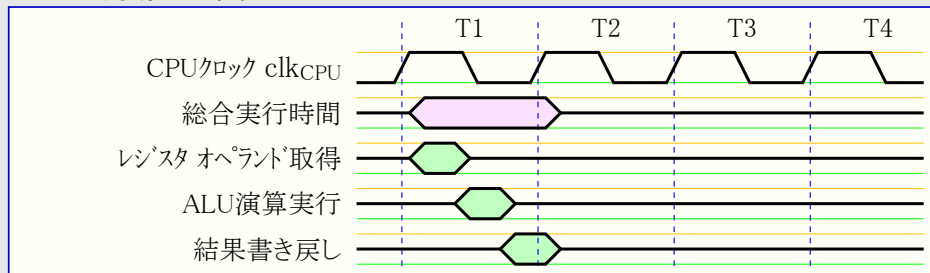


図5-5. 1周期ALU命令



5.8. 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ブ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値によっては割り込みが自動的に禁止されるかもしれません。この特質はソフトウェア保護を改善します。詳細については126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ベクタとして定義されます。ベクタの完全な一覧は33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す。割り込みベクタはMCU制御レジスタ(MCU_CR)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33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。116頁の「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」をご覧ください。

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

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

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

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

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

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

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

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

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

5.8.1. 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。この増加は選択した休止形態からの起動時間に加えてです。

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

6. AVR メモリ

この節はATmega406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ATmega406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且つ直線的なアドレスです。

6.1. 実装書き換え可能なプログラム用フラッシュメモリ

ATmega406はプログラム保存用に実装書き換え可能な40K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20K×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。ATmega406のプログラムカウンタ(PC)は15ビット幅、従って20K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、直列でのフラッシュメモリプログラミング(実装書き換え)の詳細な記述を含む116頁の「[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](#)」で詳細に記述されます。

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

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

図6-1. 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	\$4FFF

6.2. データ用SRAMメモリ

図6-2はATmega406のSRAMメモリ構成方法を示します。

ATmega406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。

ATmega406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8頁の「[汎用レジスタファイル](#)」で記述されます。

図6-2. 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

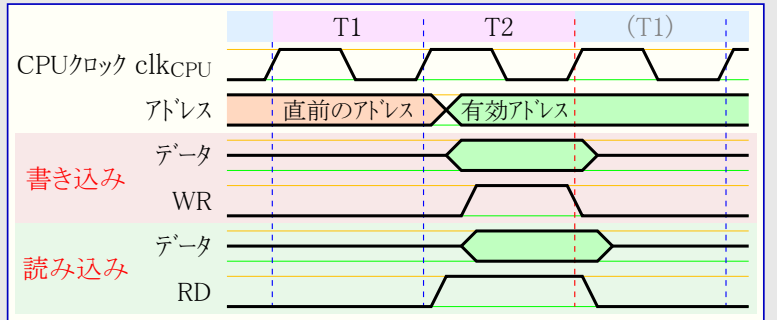
注: 赤字はI/Oアドレス

6.2.1. 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6-3で記載されるように2clkCPU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6-3. データ用内蔵SRAMアクセス周期



6.3. データ用EEPROMメモリ

ATmega406は512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への直列(JTAG)と並列のプログラミングの詳細記述については各々 [136頁](#)と[128頁](#)をご覧ください。

6.3.1. 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[表6-1](#)で与えられます。(書き込みは)自己タイミング機能ですが、使用者ソフトウェアは**次バイトが書ける時を検知**してください。使用者コードがEEPROMに書く命令を含む場合、いくつかの予防処置が取られなければなりません。

予期せぬEEPROM書き込みを防止するため、特別な書き込み手順に従わなければなりません。この詳細については「[EEPROM制御レジスタ](#)」の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。

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

ビット	15	14	13	12	11	10	9	8	
\$22 (\$42)	–	–	–	–	–	–	–	EEAR8	EEARH
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	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15～9 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

●ビット8～0 – EEAR8～0 : EEPROMアドレス (EEPROM Address)

EEPROMアドレスレジスタ(EEARHとEEARL)は512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0～511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。

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

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

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

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

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

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

●ビット7,6 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

EEPROMプログラミング種別ビット設定はEEPROMプログラミング許可(EEPE)書き込み時にどのプログラミング動作が起動されるかを定義します。1つの非分離操作(旧値消去と新値書き込み)、または2つの異なる操作として消去と書き込み操作を分離してデータをプログラムする(書く)ことが可能です。各動作に対するプログラミング時間は表6-1で示されます。EEPEが設定(1)されている間はEEPMMnへのどの書き込みも無視されます。リセット中、EEPMMnビットはEEPROMがプログラミング作業中を除いて'00'にリセットされます。

表6-1. EEPROMプログラミング種別

EEPMM1	EEPMM0	プログラミング時間	動作
0	0	3.4ms	1操作での消去と書き込み(非分離操作)
0	1	1.8ms	消去のみ
1	0	1.8ms	書き込みのみ
1	1	–	将来使用に予約

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

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

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

EEMPEビットはEEPROMプログラム許可(EEPE)ビットの1書き込みが有効か無効かどうかを決めます。EEMPEが設定(1)されると、4クロック周期内のEEPE設定(1)は選択したアドレスのEEPROMをプログラムします。EEMPEが0なら、EEPE設定(1)は無効です。EEMPEがソフトウェアによって設定(1)されてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。EEPROM書き込み手順については次のEEPE記述をご覧ください。

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

EEPROMプログラム許可信号(EEPE)はEEPROMへのプログラミング許可信号です。EEPEが(1)を書かれると、EEPROMはEEPMMnビット設定に従ってプログラムされます。論理1がEEPEへ書かれる前にEEPROM主プログラム許可(EEMPE)ビットは1を書かれなければならない、さもなければEEPROM書き込み(消去)は行われません。EEPROMを書くとき、次の手順に従うべきです(手順3と4の順番は重要ではありません)。

1. EEPROMプログラム許可(EEPE)ビットが0になるまで待ちます。
2. SPM制御/状態レジスタ(SPMCSR)の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主プログラム許可(EEMPE)ビットに1を、EEPROMプログラム許可(EEPE)ビットに0を同時に書きます。
6. EEMPEビット設定後4クロック周期内にEEPROMプログラム許可(EEPE)ビットへ論理1を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMはプログラム(書き込み)ができません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを確認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)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116ページの「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主プログラム許可が時間超過するため、書き込み周期失敗になります。EEPROMをアクセスする割り込みルーチンが他のEEPROMアクセスを中断し、EEARかEEDRが変更されると、中断したEEPROMアクセスを失敗させます。これらの問題を避けるため、全ての手順中、ステータスレジスタ(SREG)の全割り込み許可(I)ビットは解除(0)されていることが推奨されます。

書き込み(プログラミング)アクセス時間が経過されると、EEPROMプログラム許可(EEPE)ビットはハードウェアによって解除(0)されます。EEPEが設定(1)されてしまうと、次の命令が実行される前にCPUは2周期停止されます。

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

EEPROM読み込み許可信号(EERE)はEEPROMへの読み込みストロブです。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1を書かれなければなりません。EEPROM読み出しアクセスは(その)1命令で行われ、要求したデータは直ちに利用できます。EEPROMが読まれるとき、次の命令が実行される前にCPUは4周期停止されます。

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

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6-2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6-2. EEPROM書き込み時間

項目	校正付き内蔵RC発振器周期数	Typ
EEPROM書き込み(CPU)	26,368	3.3ms

6.3.A. 非分離バイトプログラミング

非分離バイトプログラミングの使用は最も簡単な動作です。EEPROMにバイトを書くとき、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEPmnビットが'00'ならば、(EEMPEが1を書かれる後の4周期内の)EEPEの1書き込みは消去/書き込み動作を起動します。消去と書き込みの両周期は1操作で行われ、総プログラミング時間は表6-1.で与えられます。EEPEビットは消去と書き込み動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

6.3.B. 分離バイトプログラミング

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

6.3.C. 消去

バイトを消去するにはアドレスがEEARに書かれなければなりません。EEPmnビットが'01'なら、(EEMPEが1を書かれた後の4周期内の)EEPEの1書き込みは消去動作だけを起動します(プログラミング時間は表6-1.で与えられます)。EEPEビットは消去動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

6.3.D. 書き込み

(特定)位置を書くため、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEPmnビットが'10'なら、(EEMPEが1を書かれる後の4周期内の)EEPEの1書き込みは書き込み動作だけを起動します(プログラミング時間は表6-1.で与えられます)。EEPEビットは書き込み動作が完了されるまで設定(1)に留まります。書かれるべき位置が書き込み前に消去されてしまっていなければ、元の保存したデータは失ったとみなされなければなりません。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

(訳注) 共通性のため本頁の内容を追加しました。

次のコード例はアセンブリ言語とC言語でのEEPROM消去、書き込み、または非分離書き込み関数を示します。本例は(例えば全割り込み禁止によって)割り込みが制御され、これらの関数実行中に割り込みが起きない前提です。本例はソフトウェア内にフラッシュブートローダが無い前提でもあります。そのようなコードが存在する場合、EEPROM書き込み関数は何れかが実行するSPM命令の完了も待たねばなりません。(訳注:共通性から次例は補足修正しています。)

アセンブリ言語プログラム例

```
EEPROM_WR:  SBIC    EECR, EEPE                ;EEPROMプログラミング完了ならばスキップ
             RJMP    EEPROM_WR              ;以前のEEPROMプログラミング完了まで待機
;
             LDI     R19, (0<<EEPM1) | (0<<EEPM0) ;プログラミング種別値取得(本例は非分離)
             OUT     EECR, R19               ;対応プログラミング種別設定
             OUT     EEARH, R18              ;EEPROMアドレス上位バイト設定
             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, EEMPE            ;EEPROM主プログラム許可ビット設定
             SBI     EECR, EEPE             ;EEPROMプログラミング開始(プログラム許可ビット設定)
             RET                               ;呼び出し元へ復帰
```

C言語プログラム例

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    while(EECR & (1<<EEPE));                /* 以前のEEPROMプログラミング完了まで待機 */
    EECR = (0<<EEPM1) | (0<<EEPM0);          /* 対応プログラミング種別設定 */
    EEAR = uiAddress;                          /* EEPROMアドレス設定 */
    EEDR = ucData;                             /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMPE);                       /* EEPROM主プログラム許可 */
    EECR |= (1<<EEPE);                       /* EEPROMプログラミング開始 */
}
```

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

アセンブリ言語プログラム例

```
EEPROM_RD:  SBIC    EECR, EEPE                ;EEPROMプログラミング完了ならばスキップ
             RJMP    EEPROM_RD              ;以前のEEPROMプログラミング完了まで待機
;
             OUT     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<<EEPE));                /* 以前のEEPROMプログラミング完了まで待機 */
    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 */
}
```

6.4. I/O メモリ (レジスタ)

ATmega406のI/O空間定義は153頁の「レジスタ要約」で示されます。

ATmega406の全てのI/Oと周辺機能はI/O空間に配置されます。全てのI/O位置はI/O空間と32個の汎用作業レジスタ間のデータ転送を行うLD/LDS/LDD命令とST/STS/STD命令によってアクセスされます。アドレス範囲\$00～\$1F内のI/OレジスタはSBI命令とCBI命令の使用で直接的にビットアクセス可能です。これらのレジスタではSBISとSBIC命令の使用によって単一ビット値が検査できます。より多くの詳細については「命令要約」章を参照してください。I/O指定命令INとOUTを使う時はI/Oアドレス\$00～\$3Fが使われなければなりません。LD命令とST命令を使い、データ空間としてI/Oレジスタをアクセスする時はこれらのアドレスに\$20が加算されなければなりません。ATmega406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節で説明されます。

6.4.1. 汎用I/Oレジスタ

ATmega406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。

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

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

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

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

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

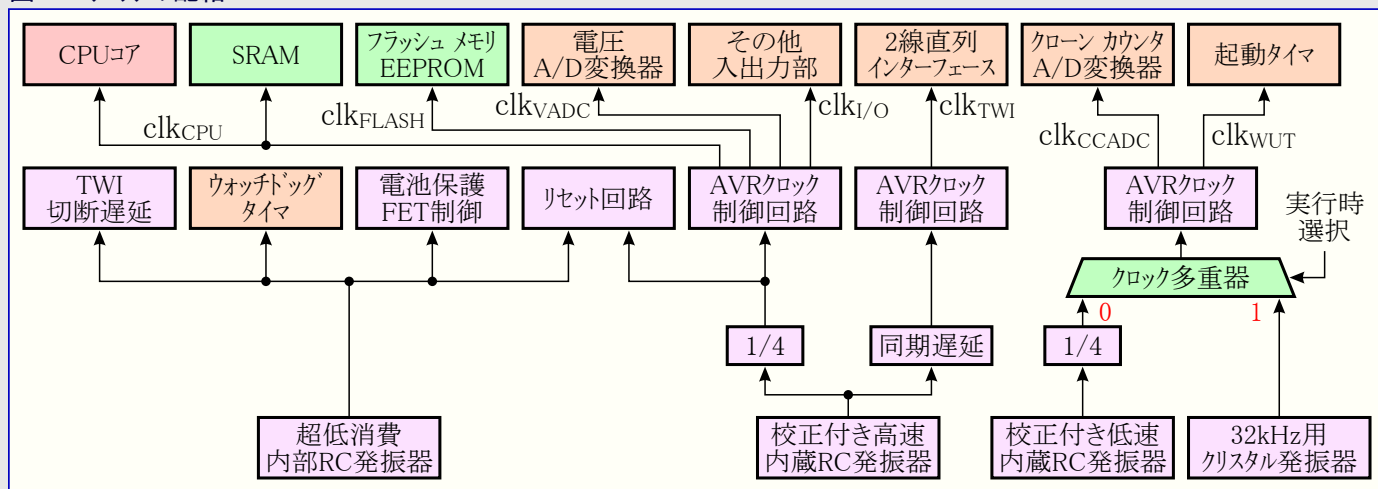
ビット	7	6	5	4	3	2	1	0	
\$1E (\$3E)	(MSB)							(LSB)	GPIOR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

7. システム クロックとクロック任意選択

7.1. クロック系統とその配給

図7-1.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、21頁の「電力管理と休止形態」で記述される各種休止形態の使用によって、使われない部分のクロックを停止することができます。クロック系統は以下で詳細に示されます。

図7-1. クロックの配給



7.1.1. CPU クロック – clkCPU

CPUクロックはAVRコアの動作と関係する系統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。

7.1.2. TWI クロック – clkTWI

2線直列インターフェース(TWI)部は専用のクロック範囲を用意されます。これはデータ転送速度仕様を達成するのにTWI部が4MHzクロックを必要とするためです。TWI通信が使われない時に、clkTWIクロックの停止によって、電力削減も許します。TWI部のアドレス認証はclkTWIが停止されると非同期に実行され、パワーオフ動作を除く全休止形態でTWIアドレス受信を可能とすることに注意してください。

7.1.3. I/O クロック – clkI/O

I/Oクロックは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のをこのような割り込みに許す、非同期論理回路によって検出されることに注意してください。

7.1.4. フラッシュ クロック – clkFLASH

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。

7.1.5. 電圧A/D変換クロック – clkVADC

電圧A/D変換器は専用のクロック範囲を用意されます。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。

7.1.6. クーロン カウンタA/D変換クロック – clkCCADC

クーロン カウンタA/D変換器は専用のクロック範囲を用意されます。継続電流測定についてパワーセーブのような低電力動作でのクーロン カウンタA/D変換器の動作を許します。

7.1.7. ウォッチドッグ タイマと電池保護

ウォッチドッグ タイマと電池保護は専用のクロック範囲を用意されます。これはパワーオフ動作を除く全ての動作種別での動作を許します。この目的専用の超低電力RC発振器を用いることによって超低電力動作を許します。

7.2. クロック元

本デバイスには以降のクロック元があります。クロックはAVR クロック発生器に入力され、適切な単位部へ配給されます。

7.3. 校正付き高速RC発振器

校正された高速RC発振器は2線直列インターフェース(TWI)を除く全単位部に対して、1.0MHzに分周される4.0MHzクロックを既定によって供給します。この周波数は25℃での公称値です。このクロックは外部部品なしで動作します。リセット中、ハードウェアが**高速RC発振器校正(FOSCCAL)レジスタ**に校正値バイトを設定し、これによって高速RC発振器を自動的に校正します。この校正は4MHz±3%の周波数を与えます。この発振器はFOSCCALレジスタの変更により、±1%の精度で3.7～4.0MHz範囲のどの周波数にも校正できます。予め設定された校正値のより多くの情報については127頁の「**校正バイト**」項をご覧ください。

この発振器が選択されると、起動時間は表7-1.で示されるようにSUTヒューズによって決定されます。

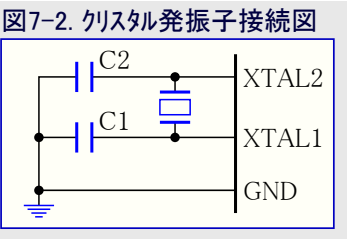
表7-1. 校正付き高速RC発振器用起動遅延時間		
SUT1,0	パワーダウン、パワーセーブからの起動遅延時間	リセットからの付加遅延時間
0 0	6×CK	14×CK
0 1	6×CK	14×CK+3.9ms
1 0	(注1) 6×CK	14×CK+62.5ms
1 1	(予約)	

注1: デバイスはこの選択で出荷されます。

7.4. 32kHzクリスタル用発振器

XTAL1とXTAL2は図7-2.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。この発振器は時計用の32kHzクリスタル発振子での使用に最適化されています。

C1とC2は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、浮遊容量の量、その環境の電磁雑音に依存します。コンデンサ選択法の情報と発振器動作のその他詳細については「**32kHzクリスタル発振器**」応用記述を参照してください。



7.5. 低速RC発振器

低速RC発振器は固定の131kHzクロックを供給します。このクロック元は32kHz発振器停止の場合に代替クロック元として使えます。クロック精度が受け入れ可能なシステムでだけ、実行時クロック元としても使えます。実行時クロック元としての使用時に良好な精度を提供するため、低速RC発振器には識票アドレス空間に格納された校正値バイトがあります。127頁の「**校正バイト**」項をご覧ください。実際の計時終了時間を得るため、応用ソフトウェアは32頁の表10-1.で得られる起動タイマ計時終了時間を合わせるのに、この校正値バイトを使わなければなりません。

7.6. 超低電力RC発振器

超低電力RC発振器(ULP発振器)は128kHzのクロックを供給します。これは周波数精度を犠牲にして、超低消費電力で動作します。

7.7. CPU,I/O,フラッシュ,電圧A/D変換のクロック

CPU、I/O、フラッシュ、電圧A/D変換器用のクロック元は校正付き高速RC発振器です。校正付き高速RC発振器は2線直列インターフェース(TWI)部に4MHzクロックを、その他の単位部に1MHzクロックを供給することに注意してください。

CPUが**パワーダウン**または**パワーセーブ**動作から起動するとき、命令実行を始める前に安定なクロックを保証するために、CPUクロック元が起動時間に使われます。これらはCPUがリセットから始まるとき、通常動作開始に先立って安定電圧へ達することを基準電圧に許す付加遅延です。超低電力RC発振器はこの起動時間の実時間部の計時用に使われます。起動時間は表7-2.で示されるようにSUTヒューズによって決められます。各計時終了時間に対して使われる超低電力RC発振器の周期数は表7-3.で示されます。

表7-2. 校正付き高速RC発振器用起動遅延時間		
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
0 0	6×CK	14×CK
0 1	6×CK	14×CK+3.9ms
1 0	6×CK	14×CK+62.5ms
1 1	(予約)	

表7-3. 超低電力RC発振器周期数	
代表計時終了時間	周期数
3.9ms	500
62.5ms	8000

7.8. クーロン カウンタA/D変換と起動タイマのクロック

クーロン カウンタA/D変換器と起動タイマのクロックは休止形態で低電力動作を許すためにCPUクロックと非同期に動作します。このクロック元は32kHzクリスタル用発振器か、または低速RC発振器(の4分周出力)のどちらかです。選択したクロックはAVRクロック制御部へ入力され、適切な単位部へ配給されます。

クーロン カウンタA/D変換器と起動タイマ用のクロック元は、**クロック制御/状態レジスタ(CCSR)の非同期クロック選択(ACS)ビット**によって選択されます。詳細については19頁の「**実行時クロック元選択**」をご覧ください。

7.9. ウォッチドッグ タイムと電池保護のクロック

ウォッチドッグ タイムと電池保護のクロックは超低電力RC発振器です。この発振器はウォッチドッグ タイムまたは電池保護のどちらかまたは両方が許可されている全ての動作で自動的に許可されます。リセット中も許可されます。

7.10. 実行時クロック元選択

クーロン カウンタA/D変換器と起動タイマ用のクロック元は32kHzクリスタル用発振器または低速RC発振器(の4分周出力)のどちらかとして、実行時に選択可能です。このクロック元はクロック制御/状態レジスタ(CCSR)の非同期クロック選択(ACS)ビットによって選択されます。

高いクロック精度を達成するためには32kHzクリスタル用発振器が推奨されるクロック元です。低速RC発振器は安価なシステム用のクロック元として、またはクリスタル クロック停止の場合の代替クロック元として用意されています。クリスタル クロックが正しい動作でないことをCPUが検知した場合、正確さを失うが未だ機能する代替の解決策として低速RC発振器に切り替えられます。

7.11. クロック関係レジスタ

7.11.1. FOSCCAL – 高速RC発振器校正レジスタ (Oscillator Calibration Register)

ビット (\$66)	7	6	5	4	3	2	1	0	
	FCAL7	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0	FOSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	デバイス固有の校正值								

●ビット7～0 – FCAL7～0 : 高速RC発振器校正值 (Fast RC Oscillator Calibration Value)

高速RC発振器校正レジスタは発振器周波数の偏差処理を省くための高速RC発振器の調整に使われます。リセット中、25℃で4MHzの発振器周波数を与える工場校正值が、この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±1%の精度で3.7～4.0MHz範囲のどの周波数にも校正できます。この範囲外は保証されません。

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がそれに応じて影響されることに注意してください。フラッシュ メモリまたはEEPROMが書かれる場合、4.4MHzより高く校正してはいけません。さもなければ、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。

FCAL7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)設定は低周波数側範囲になり、(1)設定は高周波数側範囲になります。この2つの周波数範囲は重複し、別の言葉では、FOSCCAL=\$7F設定はFOSCCAL=\$80設定より高い周波数になります。

FCAL6～0ビットは選択した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。FCAL6～0での1増加は3.7～4.0MHz周波数範囲で2%未満の周波数増加になります。

7.11.2. CCSR – クロック制御/状態レジスタ (Clock Control and Status Register)

ビット (\$C0)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	XOE	ACS	CCSR
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7～2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

●ビット1 – XOE : 32kHzクリスタル用発振器許可 (32kHz Crystal Oscillator Enable)

XOEビットは32kHzクリスタル用発振器がクロック元として選択されるのに先立って32kHzクリスタル用発振器を許可するのに使われます。32 kHzクリスタル用発振器は安定するのに概ね2秒を必要とし、これは使用者ソフトウェアで計時されなければなりません。ソフトウェアがXOEに0とACSに1を同時に書くことを試みると、ハードウェアによってXOEとACSの両方が解除(0)されます。従って32kHzクリスタル用発振器が禁止されている間中、それをクロック元として選択できません。

●ビット0 – ACS : 非同期クロック選択 (Asynchronous Clock Select)

ACSビットはクーロン カウンタA/D変換器と起動タイマ用の非同期クロック元を選択するのに使われます。このビットが解除(0)されると、低速RC発振器が選択されます。このビットが設定(1)されると、32kHzクリスタル用発振器が選択されます。

選択したクロック元と発振器許可条件は表7-4.で表されます。

表7-4. 非同期クロック元と発振器許可条件		
休止形態	32kHzクリスタル用発振器許可	低速RC発振器許可
パワーダウンまたはパワーオフ	0	0
その他の休止動作	XOE	ACS & (CADEN WUTEN)
活動動作	XOE	1

次はクーロン カウンタA/D変換器と起動タイマ用の非同期クロックとして低速RC発振器から32kHzクリスタル用発振器へ切り替えるための推奨方法です。

1. 32kHzクリスタル用発振器許可(XOE)ビットの設定(1)により、32kHzクリスタル用発振器を許可してください。
2. 起動タイマを許可し、計時終了時間2秒を選択して、起動タイマをリセットしてください(詳細については32頁の「起動タイマ」)。
3. 起動タイマの計時終了を待ってください。
4. XCEビットの設定(1)を保つと同時に非同期クロック選択(ACS)ビットを設定(1)することにより、32kHzクリスタル用発振器へ切り替えてください。
5. [任意] 32kHzクリスタル用発振器が正しく動作するのを保証するには、別の起動(監視)タイマの計時終了を待ってください。これは大事な意味を持つ、より長い計時終了時間で別のタイマ(/カウンタ)割り込みを許可し、起動タイマ計時終了が先に起こるのを検査することによって行なえます。

次はクーロン カウンタA/D変換器と起動タイマ用の非同期クロックとして32kHzクリスタル用発振器から低速RC発振器へ切り替えるための推奨方法です。

1. XCEビットの設定(1)を保つと同時にACSビットを解除(0)することにより、低速RC発振器へ切り替えてください。
2. ACSビットの解除(0)を保つと同時にXCEビットを解除(0)することにより、32kHzクリスタル用発振器を禁止してください。

8. 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。

5つの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(アイドル、A/D変換雑音低減、パワーダウン、パワーセーブ、パワーオフ)のどれかを選びます。一覧については表8-1をご覧ください。MCUが休止形態中に許可した割り込みが起これば、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。デバイスがパワーオフを除く休止形態から起動するとき、レジスタファイルとSRAMの内容は変えられません。休止形態中にリセットが起これば、MCUは起動し、リセットベクタから実行します。パワーオフ動作から復帰するとき、MCUはリセットします。

17頁の図7-1はATmega406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択する助けになります。

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

この休止形態制御レジスタは電力管理用の制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$33 (\$53)	–	–	–	–	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として読まれます。

● 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8-1で示される利用可能な5つの休止形態の1つを選択します。

表8-1. 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	パワーオフ動作 (注)
1	0	1	(予約)
1	1	0	(予約)
1	1	1	(予約)

注: この値が設定されてSEビットが論理1を書かれると、4周期後にSMCRは自動的に解除(0000)されます。本動作へ移行するには、SEの論理1書き込み後4周期以内に**SLEEP**命令を実行してください。

● 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。

8.1. 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、全周辺機能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。

8.2. A/D変換雑音低減動作

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、起動タイマ(WUT)、ウォッチドッグタイマ(WDT)、電圧A/D変換器(VADC)、クーロンカウンタA/D変換器(CCADC)、電池電流保護(CBP)、電池電圧保護(VBP)、定常電流での起動(WURC)、32kHzクリスタル用発振器または低速RC発振器、超低電力RC発振器、高速RC発振器の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。

これは電圧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。

8.3. パワーセーフ動作

休止種別選択(SM2〜0)ビットが'011'を書かれると、**SLEEP**命令はMCUをパワーセーフ動作へ移行させます。この動作では内蔵高速RC発振器が停止される一方、起動タイマ(WUT)、ウォッチドッグ タイマ(WDT)、クーロン カウンタA/D変換器(CCADC)、電池電流保護(CBP)、電池電圧保護(VBP)、定常電流での起動(WURC)、32kHzクリスタル用発振器または低速RC発振器、超低電力RC発振器は継続して動作します。

本動作は応用ソフトウェアが内蔵高速RC発振器で走行するCPU、フラッシュ メモリと周辺機能部のどれもの動作を必要としない時の既定動作です。

検出抵抗器を通る電流がクーロン カウンタA/D変換器で正確に測定できない程小さいなら、消費電力削減のために定常電流検出が許可されるべきです。起動タイマが時間を見失わないようにするので、電池の自己放電は計算できます。

レベルで起動した割り込みがパワーセーフ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37頁の「**外部割り込み**」を参照してください。

パワーセーフ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**」で定義されます。

8.4. パワーダウン動作

SM2〜0ビットが'010'を書かれると、**SLEEP**命令はMCUをパワーダウン動作へ移行させます。この動作では高速RC発振器、低速RC発振器、32kHzクリスタル用発振器が停止される一方、超低電力RC発振器、外部割り込み、電池保護、ウォッチドッグは(許可されていれば)継続して動作します。この休止形態は基本的に生成した全てのクロックを停止し、非同期部の動作だけを許します。

レベルで起動した割り込みがパワーダウン動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。より多くの詳細については37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。18頁の「**クロック元**」で定義されます。

8.5. パワーオフ動作

SM2〜0ビットが'110'を書かれると、**SLEEP**命令はCPUに対して、機能すべき電圧調整器部と充電検出回路を残したままのCPUへの電力停止を電圧調整器部に要求させます。意図する時にだけMCUがパワーオフ動作へ移行するのを保証するため、**SLEEP**命令はSM2〜0ビットが書かれた後の4クロック周期以内に実行されなければなりません。

パワーオフ休止動作へ移行する前にソフトウェアによって割り込みが禁止されるべきであることに注意してください。さもなければ制限時間内に実行されるべき**SLEEP**命令を妨げるかもしれません。

表8-2. 各休止形態に於ける活動単位部

単位部	活動動作	休止種別				
		アイドル	A/D変換雑音低減	パワーセーブ	パワーダウン	パワーオフ
高速RC発振器	○	○	○			
超低電力RC発振器	○	○	○	○	○	
低速RC発振器, 32kHzクリスタル用発振器	○	○	○	○		
CPU, フラッシュメモリ	○					
8/16ビットタイマ/カウンタ	○	○				
SMバス(TWI)	○	○	①	①	①	
電圧A/D変換器	○	○	○			
クローンカウンタA/D変換器, 起動タイマ	○	○	○	○		
外部割り込み, ウォッチドッグタイマ, 電池電圧保護	○	○	○	○	○	
電池電流保護 (注2)	○	○	○	○	○	
電圧調整器	○	○	○	○	○	○
充電検出						○

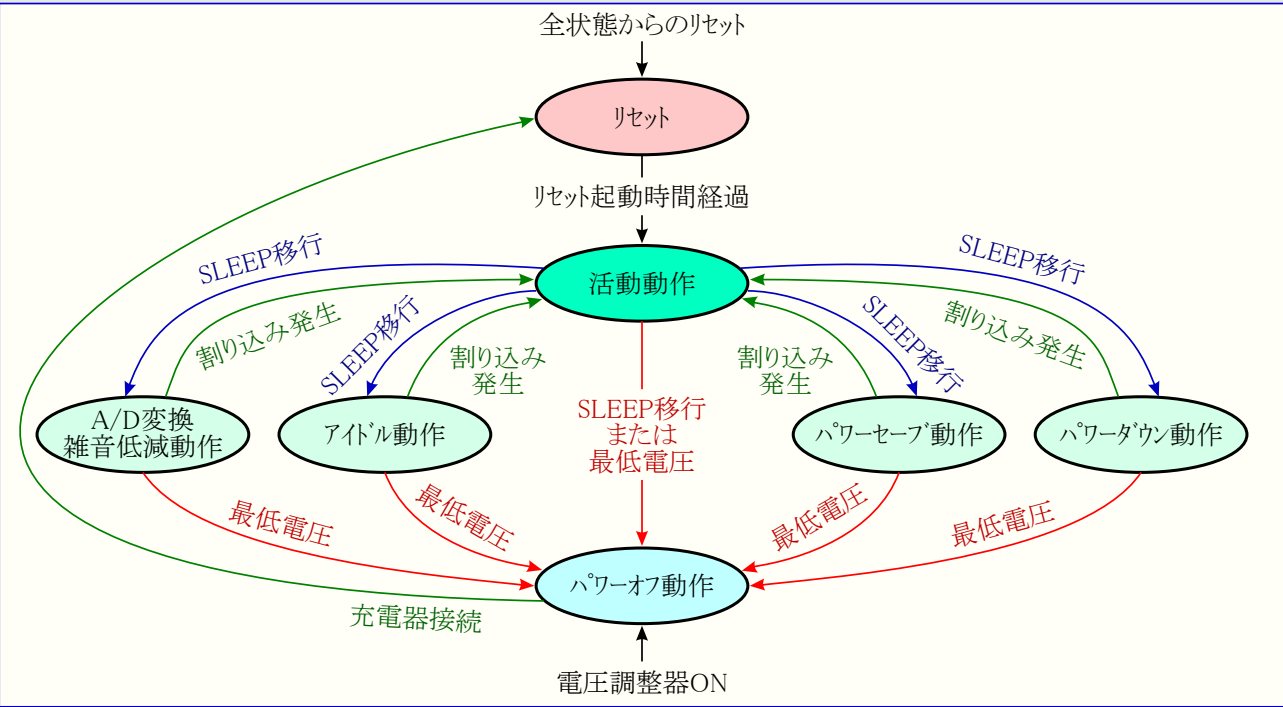
① : アドレス一致とバス接続/切断起動だけです。
(注2) : 放電FETがOFFのとき、短絡保護は消費電流削減のため、自動的に禁止されます。

表8-3. 各休止形態に於ける復帰起動要因 (割り込み)

休止種別	電圧調整器 電流起動	電池 保護	外部 割り込み	SMBus アドレス一致 接続/切断	ウォッチ ドッグ タイマ	起動 タイマ	SPM EEPROM 操作可	クローンカウンタ A/D 変換完了	電圧A/D 変換完了	その他 I/O	充電器 接続
アイドル	○	○	○	○	○	○	○	○	○	○	
A/D変換雑音低減	○	○	○	○	○	○	○	○	○		
パワーセーブ	○	○	○	○	○	○		○			
パワーダウン		○	○	○	○						
パワーオフ											○

休止形態状態遷移図は図8-1.で示されます。

図8-1. 休止形態状態遷移図



8.6. 電力削減

電力削減レジスタ(PRR0)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レジスタは書きません。クロックを停止している時に周辺機能によって使われていた資源は占有されたままなので、その周辺機能は殆どの場合、クロックを停止する前に禁止されるべきです。周辺機能部の起動は電力削減レジスタ(PRR0)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。

8.6.1. PRR0 – 電力削減レジスタ0 (Power Reduction Register 0)

ビット (\$64)	7	6	5	4	3	2	1	0	
	–	–	–	–	PRTWI	PRTIM1	PRTIM0	PRVADC	PRR0
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として読まれます。

- ビット3 – PRTWI : 2線直列インターフェース電力削減 (Power Reduction TWI)

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

- ビット2 – 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1 – PRTIM0 : タイマ/カウンタ0電力削減 (Power Reduction Timer/Counter0)

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

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

このビットへの1書き込みは電圧A/D変換器(VADC)を停止します。電圧A/D変換器は停止前に禁止されなければなりません。

8.7. 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ために選択されるべきです。必要とされない全ての機能は禁止されるべきです。特に次の機能部は最低可能消費電力の達成を試みるとき、特別な考慮を必要とするでしょう。

8.7.1. 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パワーオフ動作を除く全休止形態で許可されます。ウォッチドッグ タイマ消費電流はパワーダウン動作でだけ重要です。ウォッチドッグ タイマ設定法の詳細については28頁の「ウォッチドッグ タイマ」をご覧ください。

8.7.2. 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(clkI/O)と電圧A/D変換クロック(clkVADC)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のに必要とされ、その時は許可されます。どのピンが許可されるかの詳細については42頁の「デジタル入力許可と休止形態」をご覧ください。入力緩衝部が許可され、入力信号が浮いている状態のままか、またはアナログ信号電圧がVCC/2付近の場合、入力緩衝部は過大な電力を消費するでしょう。

アナログ入力ピンに対するデジタル入力緩衝部は常に禁止されるべきです。入力ピンでのVCC/2付近のアナログ信号入力は活動動作でも重要な電流を引き起こし得ます。デジタル入力緩衝部はデジタル入力禁止レジスタ(DIDR0)の書き込みによって禁止できます。詳細については79頁の「DIDR0 – デジタル入力禁止レジスタ」を参照してください。

8.7.3. 内蔵デバッグ機能

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

8.7.4. 電池保護

電池保護機能の1つが応用で必要とされないなら、その機能は禁止されるべきです。84頁「BPCR – 電池保護制御レジスタ」をご覧ください。放電FETがOFFに切り替えられるとき、消費電力最小化のために短絡検出回路は自動的に禁止されます。電池保護回路の消費電流はパワーダウン動作でだけ重要です。

8.7.5. 電圧A/D変換器

許可なら、電圧A/D変換器は休止形態と無関係に電力を消費します。節電するには使われない場合、パワーセーフまたはパワーダウンの休止形態へ移行する前に禁止されるべきです。電圧A/D変換器操作の詳細については76頁の「電圧A/D変換器 (10チャネル汎用12ビット $\Delta\Sigma$ A/D変換器)」をご覧ください。

8.7.6. クーロン カウンタA/D変換器

許可なら、クーロン カウンタA/D変換器は休止形態と無関係に電力を消費します。節電するには使われない場合、パワーダウン休止動作へ移行する前に禁止されるべきです。クーロン カウンタA/D変換器操作の詳細については70頁の「クーロン カウンタ (容量/残量計専用 $\Delta\Sigma$ A/D変換器)」をご覧ください。

9. システム制御とリセット

9.1. 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、割り込みベクタがファームウェア領域の場合やその逆も同様です。図9-1の回路構成図はリセット論理回路を示します。

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。

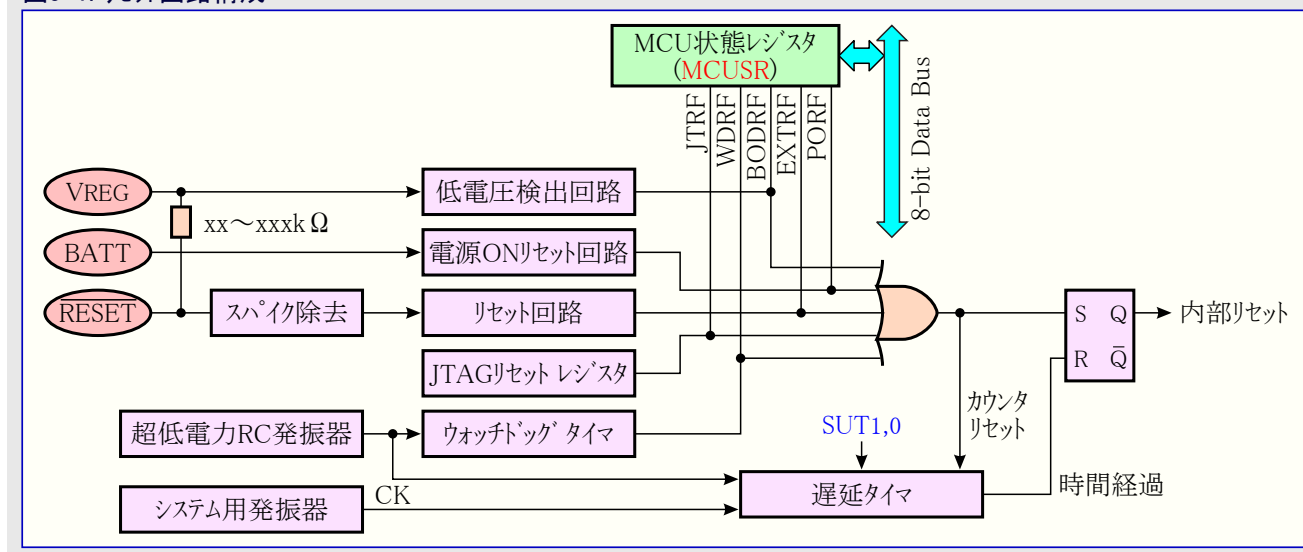
全てのリセット元が無効にされてしまった後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電圧調整器に許します。遅延タイマの遅延時間はSUTヒューズを通して使用者によって定義されます。この遅延時間についての各種選択は18頁の「クロック元」で示されます。

9.2. リセット元

ATmega406には次の各種リセット元があります。

- ・ **電源ONリセット** チップがパワーオフ動作で、充電器が接続されると、充電器検出部はリセットパルスを生成します。詳細については27頁の「電源ONリセットと充電器接続」をご覧ください。
- ・ **外部リセット** RESETピンに最小パルス幅より長くLowレベルが存在すると、MCUはリセットされます。詳細については27頁の「外部リセット」をご覧ください。
- ・ **ウォッチドッグリセット** ウォッチドッグシステムリセット動作が許可され、ウォッチドッグタイマ時間が終了すると、MCUはリセットされます。詳細については27頁の「ウォッチドッグリセット」をご覧ください。
- ・ **低電圧リセット** VREGが低電圧検出電圧(V_{BOT})以下のとき、MCUはリセットされます。詳細については27頁の「低電圧検出」をご覧ください。
- ・ **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はリセットされます。詳細については112頁の「JTAGインターフェースと内蔵デバッグ機能」をご覧ください。

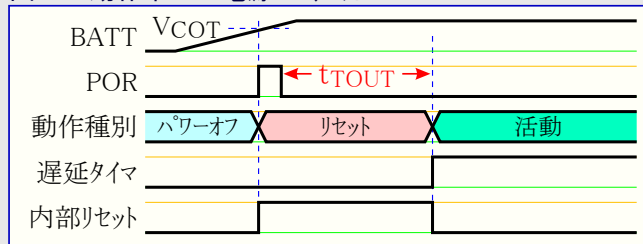
図9-1. リセット回路構成



9.2.1. 電源ONリセットと充電器接続

パワーオフ動作から開始できるためには充電器が検出されなければなりません。充電器を検出するためにはBATTピンの電圧が**充電器ON閾値電圧(VCOT)**以上に上がっていかねばなりません。これは電源ONリセット(POR)で発生し、チップはリセット動作へ移行します。遅延計数器が計時を終了すると、チップは活動動作へ移行します。149頁の表30-5は電源ONリセット特性を示します。

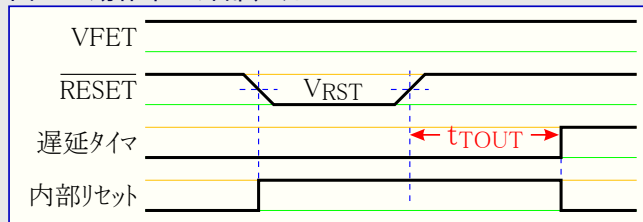
図9-2. 動作中での電源ONリセット



9.2.2. 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。例えばクロックが動いていなくても、最小パルス幅149頁の表30-6.参照)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の生成が保証されません。印加された信号の上昇が**リセット閾値電圧(VRST)**に達すると(遅延計数器を起動し)、遅延計数器は遅延時間(t_{TOUT})経過後にMCUを始動します。

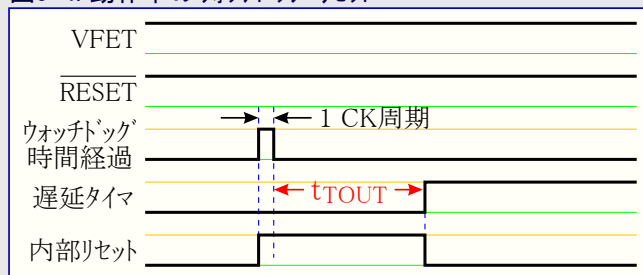
図9-3. 動作中の外部リセット



9.2.3. ウォッチドッグ リセット

ウォッチドッグ時間経過時、(内部的に)1CK周期幅の短いリセットパルスを生成します。このパルスの下降端で、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28頁を参照してください。

図9-4. 動作中のウォッチドッグ リセット

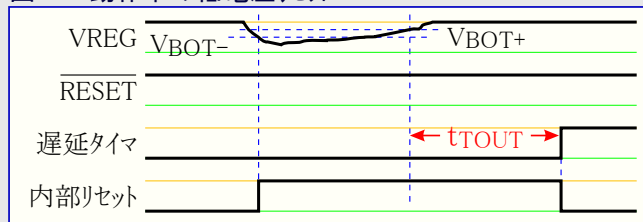


9.2.4. 低電圧(ブラウンアウト)検出リセット

ATmega406には固定化された起動(検出)電圧($V_{BOT}=2.7V$)と比較することで動作中のVREGを監視するチップ上の低電圧検出(BOD)回路があります。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。この検出電圧のヒステリシスは $V_{BOT+}=V_{BOT+} V_{HYST}/2$ と $V_{BOT-}=V_{BOT-} V_{HYST}/2$ と解釈されるべきです。

BOD許可時、VREGが起動(検出)電圧以下の値に下降すると(図9-5.の V_{BOT-})、低電圧リセットが直ちに有効とされます。VREGが起動電圧以上に上昇すると(図9-5.の V_{BOT+})、(遅延計数器が起動され、)遅延計数器は遅延時間(t_{TOUT})経過後にMCUを始動します。

図9-5. 動作中の低電圧リセット



9.3. ウォッチドッグ タイマ

ATmega406には強化されたウォッチドッグ タイマ(WDT)があります。主な機能を次に示します。

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

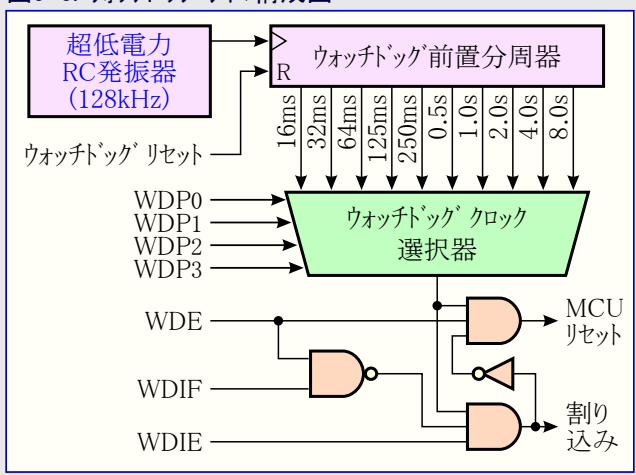
ウォッチドッグ タイマ(WDT)は128kHzで走行する超低電力発振器の周期で計時するタイマです。WDTは計数器が与えられた計時完了値に達した時に割り込みまたはシステム リセットを生じます。通常動作では計時完了値へ達する前に計数器を再始動するために、システムはウォッチドッグ リセット(WDR)命令を使う必要があります。システムが計数器を再始動しなければ、割り込みまたはシステム リセットが起こるでしょう。

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

ウォッチドッグ常時ON(WDTON)ヒューズのプログラム(0)はウォッチドッグ タイマをシステム リセット動作種別に強制します。このヒューズのプログラム(0)でシステム リセット動作(WDE)ビットと割り込み動作(WDIE)ビットは各々、'1'と'0'に固定されます。更にプログラム保護を保証するためにウォッチドッグ設定の変更は時間制限手順に従わなければなりません。システム リセット許可(WDE)の解除と計時完了時間設定の変更についての手順は次のとおりです。

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

図9-6. ウォッチドッグ タイマ構成図



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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 5頁の「コード例について」をご覧ください。

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

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 5頁の「コード例について」をご覧ください。

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

9.4. リセット関係用レジスタ

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

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

ビット	7	6	5	4	3	2	1	0	
\$34 (\$54)	–	–	–	JTRF	WDRF	BODRF	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として読めます。

● 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で選択された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 – BODRF : 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。

9.4.2. WDTCSR – ウォッチドッグ タイム制御レジスタ (Watchdog Timer Control Register)

ビット (\$60)	7	6	5	4	3	2	1	0	
	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	WDTCSR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	不定	0	0	0	

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

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

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

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

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

表9-1. ウォッチドッグ タイマ設定				
WDTON	WDE	WDIE	動作種別	計時完了での動作
1	0	0	停止	なし
1	0	1	割り込み	割り込み
1	1	0	システム リセット	リセット
1	1	1	割り込み及びシステム リセット	割り込み、その後システム リセット動作種別
0	x	x	システム リセット	リセット

(訳注)
左表のWDTONヒューズ値はプログラム=0, 非プログラム=1で表記しています。従って原書の表と逆論理になっています。

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

このビットはウォッチドッグ リセット許可(WDE)と前置分周器ビットの変更用の時間制限手順で使われます。WDEビットの解除(0)や前置分周器ビット変更のため、WDCEは設定(1)されなければなりません。一旦1を書かれると、4クロック周期後にハードウェアがWDCEを解除(0)します。

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

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

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

このWDP3~0ビットはウォッチドッグ タイマが走行する時のウォッチドッグ タイマの前置分周を決めます。各種前置分周値と対応する計時完了周期は表9-2.で示されます。

表9-2. ウォッチドッグ前置分周選択																
WDP3	0								1							
WDP2	0				1				0				1			
WDP1	0		1		0		1		0		1		0		1	
WDP0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
WDT発振周期数	2k	4k	8k	16k	32k	64k	128k	256k	512k	1024k	(予約)					
代表的計時完了周期 (VCC=3V)	16ms	32ms	64ms	0.125s	0.25s	0.5s	1.0s	2.0s	4.0s	8.0s						

10. 起動タイマ

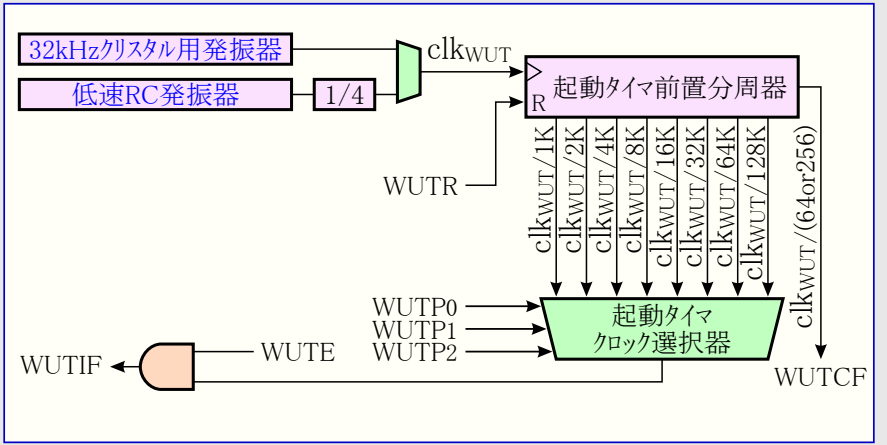
後続項はATmega406の起動タイマ(WUT)を説明します。

- 1つの起動タイマ割り込み
- 選択可能な8つの計時終了時間
- 独立した起動タイマ校正フラグ
- 独立したクロック元

10.1. 概要

起動タイマは低速RC発振器か32kHzクリスタル用発振器のどちらかからクロック駆動されます。詳細については19頁の「実行時クロック元選択」をご覧ください。起動タイマ前置分周器の制御によって31.25ms〜4sに調節できます。計時終了時間を決めるのに8つの異なるクロック周期が選択できます。

図10-1. 起動タイマ構成図



10.2. 起動タイマ用レジスタ

10.2.1. WUTCSR – 起動タイマ制御/状態レジスタ (Wake-up Timer Control and Status Register)

ビット (\$60)	7	6	5	4	3	2	1	0	
	WUTIF	WUTIE	WUTCF	WUTR	WUTE	WUTP2	WUTP1	WUTP0	WUTCSR
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 – WUTIF : 起動タイマ割り込み要求フラグ** (Wake-up Timer Interrupt Flag)
起動タイマで計時終了(溢れ)が起こると、WUTIFビットが設定(1)されます。対応する割り込み処理ベクタを実行すると、WUTIFはハードウェアによって解除(0)されます。代わりにWUTIFはこのフラグへの論理1書き込みによっても解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと起動タイマ割り込み許可(WUTIE)が設定(1)されていれば、起動タイマ割り込みが実行されます。
- **ビット6 – WUTIE : 起動タイマ割り込み許可** (Wake-up Timer Interrupt Enable)
このWUTIE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、起動タイマ割り込みが許可されます。起動タイマで終了(溢れ)が起こる、換言すると起動タイマ割り込み要求フラグ(WUTIF)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5 – WUTCF : 起動タイマ校正フラグ** (Wake-up Timer Calibration Flag)
このWUTCFビットは1.95ms(低速RC発振器=256、32kHzクリスタル用発振器=64クロック)毎に設定(1)されます。WUTCFはこのフラグへの論理1書き込みによって解除(0)されます。WUTCFは低速RC発振器または32kHzクリスタル用発振器に対する高速RC発振器校正に使えます。
- **ビット4 – WUTR : 起動タイマ リセット** (Wake-up Timer Reset)
WUTRビットが1を書かれると、起動タイマはリセットして0から計数を始めます。WUTRビットは常に0として読まれます。
- **ビット3 – WUTE : 起動タイマ許可** (Wake-up Timer Enable)
WUTEビットが設定(1)されると起動タイマが許可され、解除(0)されると起動タイマ機能が禁止されます。許可する時はWUTEと起動タイマリセット(WUTR)ビットが同時に設定(1)されることによって、起動タイマをリセットすることが推奨されます。
- **ビット2~0 – WUTP3~0 : 起動タイマ前置分周選択** (Wake-up Timer Prescaler 2,1 and 0)
このWUTP2~0ビットは起動タイマ許可時の起動タイマ前置分周を決めます。各種前置分周値と対応する計時完了周期は表10-1.で示されます。これらのビット変更時、起動タイマは常にリセットされるべきです。

表10-1. 起動タイマ前置分周器選択

WUTP2	0				1			
	0		1		0		1	
WUTP1	0		1		0		1	
WUTP0	0	1	0	1	0	1	0	1
低速RC発振器周期数	4k	8k	16k	32k	64k	128k	256k	512k
32kHzクリスタル用発振器周期数	1k	2k	4k	8k	16k	32k	64k	128k
計時終了周期代表値	31.25ms	62.5ms	125ms	250ms	0.5s	1.0s	2.0s	4.0s

11. 割り込み

本項はATmega406によって実行される割り込み操作の詳細を記述します。AVR割り込み操作の一般説明については9頁の「リセットと割り込みの扱い」を参照してください。

11.1. ATmega406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	外部ピン, 電源ON, WDT, BOD等の各種リセット
2	\$0002	BPINT	電池保護割り込み
3	\$0004	INT0	外部割り込み要求0
4	\$0006	INT1	外部割り込み要求1
5	\$0008	INT2	外部割り込み要求2
6	\$000A	INT3	外部割り込み要求3
7	\$000C	PCINT0 (PCI0)	ピン変化0群割り込み要求
8	\$000E	PCINT1 (PCI1)	ピン変化1群割り込み要求
9	\$0010	ウォッチドッグ WDT	ウォッチドッグ計時終了
10	\$0012	起動タイマ WAKE_UP	起動タイマ溢れ
11	\$0014	タイマ/カウンタ1 COMP1	タイマ/カウンタ1比較一致
12	\$0016	タイマ/カウンタ1 OVFI	タイマ/カウンタ1溢れ
13	\$0018	タイマ/カウンタ0 COMP0A	タイマ/カウンタ0比較A一致
14	\$001A	タイマ/カウンタ0 COMP0B	タイマ/カウンタ0比較B一致
15	\$001C	タイマ/カウンタ0 OVFI	タイマ/カウンタ0溢れ
16	\$001E	2線直列インターフェース TWI_BUS	2線直列インターフェースバス接続/切断
17	\$0020	2線直列インターフェース TWI	2線直列インターフェース状態変化
18	\$0022	電圧A/D変換器 VADC	電圧A/D変換完了
19	\$0024	CC-A/D変換器 CCADC	クーロン カウンタA/D変換器瞬間電流変換完了
20	\$0026	CC-A/D変換器 CCADC_REG_C	クーロン カウンタA/D変換器定常電流比較
21	\$0028	CC-A/D変換器 CCADC_ACC	クーロン カウンタA/D変換器累積電流変換完了
22	\$002A	EEPROM EE_RDY	EEPROM 操作可
23	\$002C	SPM命令 SPM_RDY	SPM命令操作可

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。116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。

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125頁の表27-7.で示されます。

ATmega406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 BPINT	;電池保護割り込み
\$0004		JMP EXT_INT0	;外部割り込み要求0
\$0006		JMP EXT_INT1	;外部割り込み要求1
\$0008		JMP EXT_INT2	;外部割り込み要求2
\$000A		JMP EXT_INT3	;外部割り込み要求3
\$000C		JMP PCINT0	;ピン変化0群割り込み要求
\$000E		JMP PCINT1	;ピン変化1群割り込み要求
\$0010		JMP WDT_OVF	;ウォッチドッグ計時終了
\$0012		JMP WAKE_UP	;起動タイマ溢れ
\$0014		JMP TIM1_COMP	;タイマ/カウンタ1比較一致
\$0016		JMP TIM1_OVF	;タイマ/カウンタ1溢れ
\$0018		JMP TIM0_COMPA	;タイマ/カウンタ0比較A一致
\$001A		JMP TIM0_COMPB	;タイマ/カウンタ0比較B一致
\$001C		JMP TIM0_OVF	;タイマ/カウンタ0溢れ
\$001E		JMP TWI_BUS	;2線直列インターフェースバス接続/切断
\$0020		JMP TWI	;2線直列インターフェース状態変化
\$0022		JMP VADC	;電圧A/D変換完了
\$0024		JMP CCADC	;クーロン カウンタA/D変換器瞬間電流変換完了
\$0026		JMP CCADC_REG_C	;クーロン カウンタA/D変換器定常電流比較
\$0028		JMP CCADC_ACC	;クーロン カウンタA/D変換器累積電流変換完了
\$002A		JMP EE_RDY	;EEPROM操作可
\$002C		JMP SPM_RDY	;SPM命令操作可
;			
\$002E	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$002F		OUT SPH, R16	;スタック ポインタ上位を初期化
\$0030		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0031		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2K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	\$4C02	;ブート プログラム領域が2Kバイトの場合
\$4C02		JMP BPINT	;電池保護割り込み
\$4C04		JMP EXT_INT0	;外部割り込み要求0
);			
\$4C2C		JMP SPM_RDY	;SPM命令操作可

BOOTRSTヒューズがプログラム(0)、ブート領域容量が2K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 BPINT	; 電池保護割り込み
\$0004		JMP EXT_INT0	; 外部割り込み要求0
		}	
\$002C		JMP SPM_RDY	; SPM命令操作可
		}	; 以下、プログラムなど
		. ORG \$4C00	; ブート プログラム領域が2Kバイトの場合
\$4C00	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$4C01		OUT SPH, R16	; スタック ポインタ上位を初期化
\$4C02		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$4C03		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2K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(MCU_CR)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 \$4C00	; ブート プログラム領域が2Kバイトの場合
\$4C00		JMP RESET	; 各種リセット (BOOTRSTヒューズ=0)
\$4C02		JMP BPINT	; 電池保護割り込み
\$4C04		JMP EXT_INT0	; 外部割り込み要求0
		}	
\$4C2C		JMP SPM_RDY	; SPM命令操作可
;			
\$4C2E	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$4C2F		OUT SPH, R16	; スタック ポインタ上位を初期化
\$4C30		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$4C31		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. 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。

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)
{
    uchr 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 */
}
```

11.3. 割り込みベクタ移動用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	–	–	PUD	–	–	IVSEL	IVCE	MCUCR
Read/Write	R/W	R	R	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116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116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。コード例をご覧ください。

12. 外部割り込み

12.1. 概要 (訳注: 共通性から本項は加筆されています。)

外部割り込みはINT0～3ピンまたはPCINT0～15ピンの何れかによって起動されます。許可したなら、例えばINT0～3またはPCINT0～15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。ピン変化割り込みの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はPCINT8～15に対応します。ピン変化割り込み許可レジスタn(PCMSK0, PCMSK1)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。PCINT0～15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パワーオフ動作を除いて、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。

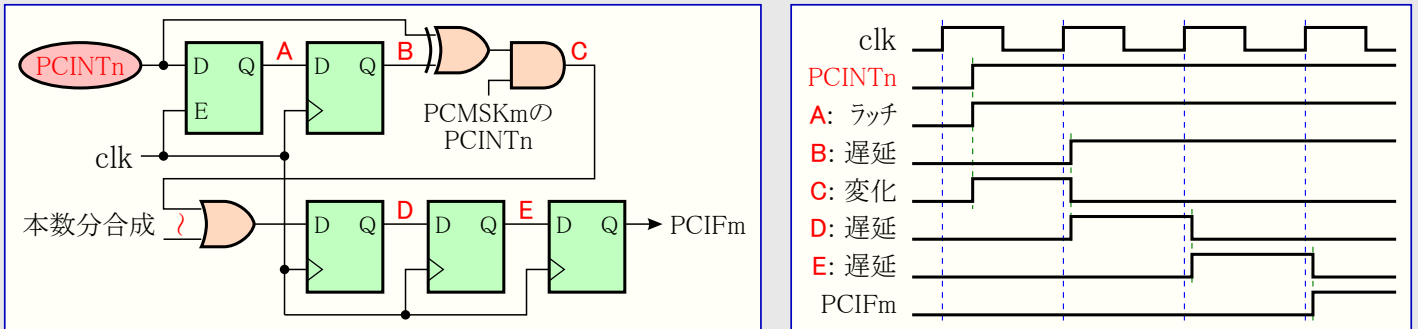
INT0～3割り込みは上昇端または下降端(含む両端)、またはLowレベルによって起動できます。これは外部割り込み制御レジスタA(EICRA)の詳述で示される設定です。INT0～3割り込みがレベル起動として設定、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT0～3の上昇端または下降端割り込みの認知は17頁の「クロックシステムとその配給」で記述されるI/Oクロックの存在を必要とすることに注意してください。INT0～3のLowレベル割り込みは非同期に検知されます。これはそれらの割り込みがパワーオフ動作を除いて、アイドル動作以外の休止形態からもデバイスを起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、変化したレベルはMCUを起動するために一定時間保たなければならないことに注意してください。これはMCUの雑音不安定性を減らします。変化したレベルは低速RC発振器クロックによって2度採取されます。低速RC発振器の周期は25℃で(公称)7.8μsです。この採取の間、入力が必要としたレベルであるか、または起動時間の最後まで保たれれば、MCUは起動します。起動時間は17頁の「システムクロックとクロック任意選択」で示されるようにSUTヒューズによって定義されます。このレベルが低速RC発振器クロックによって2度採取されるけれど、起動時間の最後に先立って消滅すると、MCUは今までもどおり起動しますが、割り込みが生成されません。この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保たれなければなりません。

12.2. ピン変化割り込みタイミング* (訳注: 共通性から本項追加)

ピン変化割り込みの例は図12-1.で示されます。

図12-1. ピン変化割り込みタイミング



12.3. 外部割り込み用レジスタ

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

ビット (\$69)	7	6	5	4	3	2	1	0	
	ISC31	ISC30	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	EICRA
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 – ISC31,0～ISC01,0 : 外部割り込み3～0条件制御 (External Interrupt3～0 Sense Control Bits)

外部割り込み3～0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)の対応する割り込み許可ビットが設定(1)される場合のINT3～0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INT3～0ピンのエッジとレベルは表12-1.で定義されます。INT3～0のエッジは非同期に記録されます。表12-2.で与えられた最小パルス幅より広いINT3～0の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択される場合、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。許可したなら、レベルで起動する割り込みはそのピンがLowを保持される限り割り込み要求を(連続的に)生成します。ISCnビット変更時、割り込みが起き得ます。従って最初にEIMSKの割り込み許可ビットを解除(0)することによってINTn割り込みを禁止することが推奨されます。その後、ISCnビットは変更できます。最後にINTn割り込み要求フラグは割り込み再許可前に外部割り込み要求フラグレジスタ(EIFR)の外部割り込みn要求(INTFn)フラグに論理1を書くことによって解除(0)されるべきです。

表12-1. 外部割り込み3～0 割り込み条件

ISCn1	ISCn0	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

注: n=3～0

ISCn1,0ビット変更時、EIMSKで割り込み許可ビットの解除(0)によって割り込みが禁止されなければなりません。さもなければビットが変更される時に割り込みが起き得ます。

表12-2. 非同期外部割り込み特性

シンボル	項目	最小	代表	最大	単位
t _{INT}	非同期外部割り込み最小パルス幅		50		ns

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

ビット \$1D (\$3D)	7	6	5	4	3	2	1	0	
	–	–	–	–	INT3	INT2	INT1	INT0	EIMSK
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として読まれます。

● ビット3～0 – INT3～INT0 : 外部割り込み3～0 許可 (External Interrupt Request 3～0 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、INT3～0ビットが1を書かれると、対応する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(EICRA)の割り込み条件制御n(ISCn1,ISCn0)ビットは、その外部割り込みが上昇端、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。例えばそのピンが出力として設定されても、そのピンの動きは割り込み要求を引き起こします。これはソフトウェア割り込みを生成する方法を提供します。

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

ビット \$1C (\$3C)	7	6	5	4	3	2	1	0	
	–	–	–	–	INTF3	INTF2	INTF1	INTF0	EIFR
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として読まれます。

● ビット3～0 – INTF3～INTF0 : 外部割り込み3～0要求フラグ (External Interrupt Flag3～0)

INT3～0ピン上のエッジまたは論理変化が割り込み要求を起動すると、INTF3～0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)の対応する割り込み許可(INT3～0)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。INT3～0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。INT3～0割り込み禁止で休止形態へ移行すると、これらのピンの入力緩衝部が禁止されることに注意してください。これはINTF3～0フラグを設定(1)する内部信号の論理変化を起こすかもしれません。より多くの情報については42頁の「デジタル入力許可と休止形態」をご覧ください。

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

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

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE0ビットが設定(1)されると、ピン変化割り込み0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PCIE0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。

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

ビット \$1B (\$3B)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	PCIF1	PCIF0	PCIFR
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 – 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)ビットとピン変化割り込み制御レジスタ(PCICR)のピン変化1群割り込み許可(PCIE1)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。

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

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

(訳注) PCICRとPCIFRのビット7～2は予約されており、常に0として読まれます。

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

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

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

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

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

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

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

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

13. 低電圧入出力ポート

13.1. 序説

全てのAVRの低電圧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。全ての低電圧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145頁の「電気的特性」を参照してください。

本項内の全てのレジスタとビットの参照は一般形で記されます。小文字の'*x*'はポート番号文字、小文字の'*n*'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記されたPORT x_n 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47頁の「低電圧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(PORT x)、方向レジスタ(DDR x)、入力レジスタ(PIN x)の各低電圧ポートに対して、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PIN x 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、全ポート内の全低電圧ピン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低電圧ポート」で記述されます。多くの低電圧ポートピンはデバイスの周辺機能用の交換機能と多重化されます。ポートピンとの各交換機能のインターフェース法は43頁の「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項目を参照してください。

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

13.2. 標準デジタル入出力としての低電圧ポート

この低電圧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P x_n 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. 低電圧入出力ピン等価回路

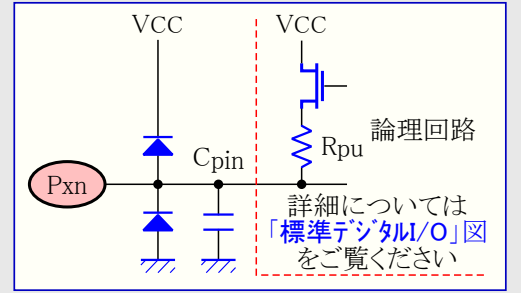
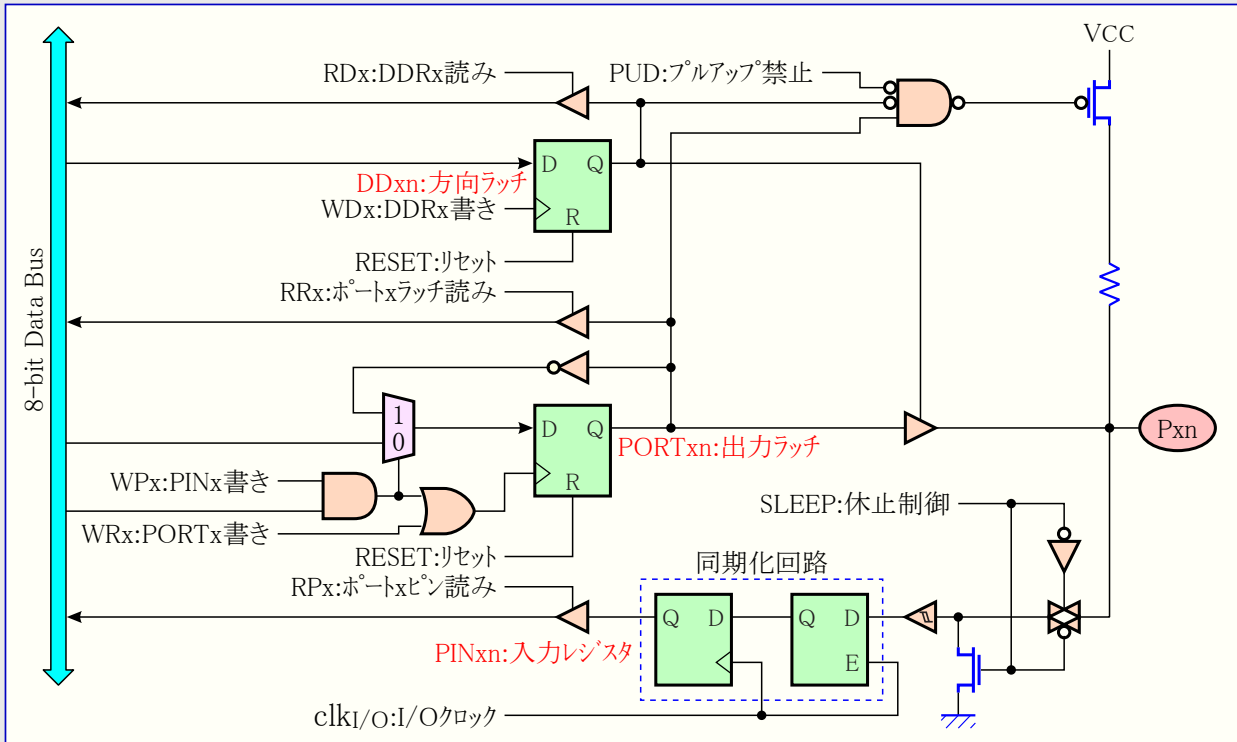


図13-2. 低電圧標準デジタル入出力回路構成



注: WR x , WP x , WD x , RR x , RP x , RD x は同一ポート内の全ピンで共通です。
clk_{I/O}, 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。47頁の「低電圧I/Oポート用レジスタ」で示されるようにDDxnビットはDDR_x I/Oアドレス、PORTxnビットはPORT_x I/Oアドレス、PINxnビットはPIN_x I/Oアドレスでアクセスされます。

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

そのピンが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。

表13-1. ポートピンの設定					
DDxn	PORTxn	PUD (MCUCR)	入出力	プルアップ抵抗	備考
0	0	X	入力	なし	高インピーダンス (Hi-Z)
0	1	0	入力	あり	Pxnに外部からLowを入力すると吐き出し電流が流れます。
0	1	1	入力	なし	高インピーダンス (Hi-Z)
1	0	X	出力	なし	Low (吸い込み)出力
1	1	X	出力	なし	High (吐き出し)出力

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エッジ付近で値を変える場合の未定義状態(メタ ステープル)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

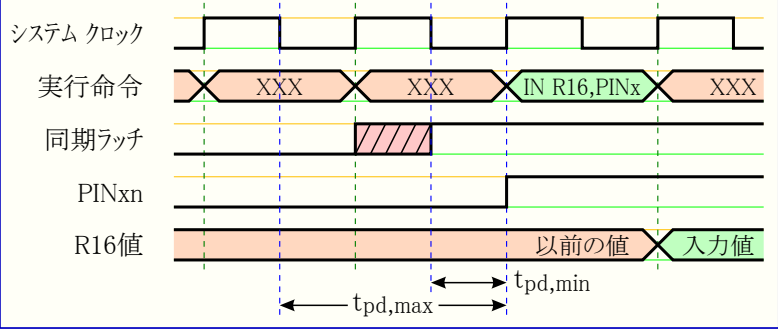
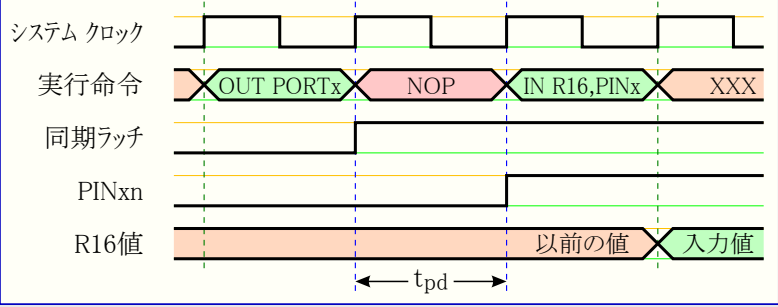


図13-4. プログラムで設定したピン値読み戻し時の同期化



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、**パワーダウン動作**、**パワーセーブ動作**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43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

多くの低電圧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系統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。

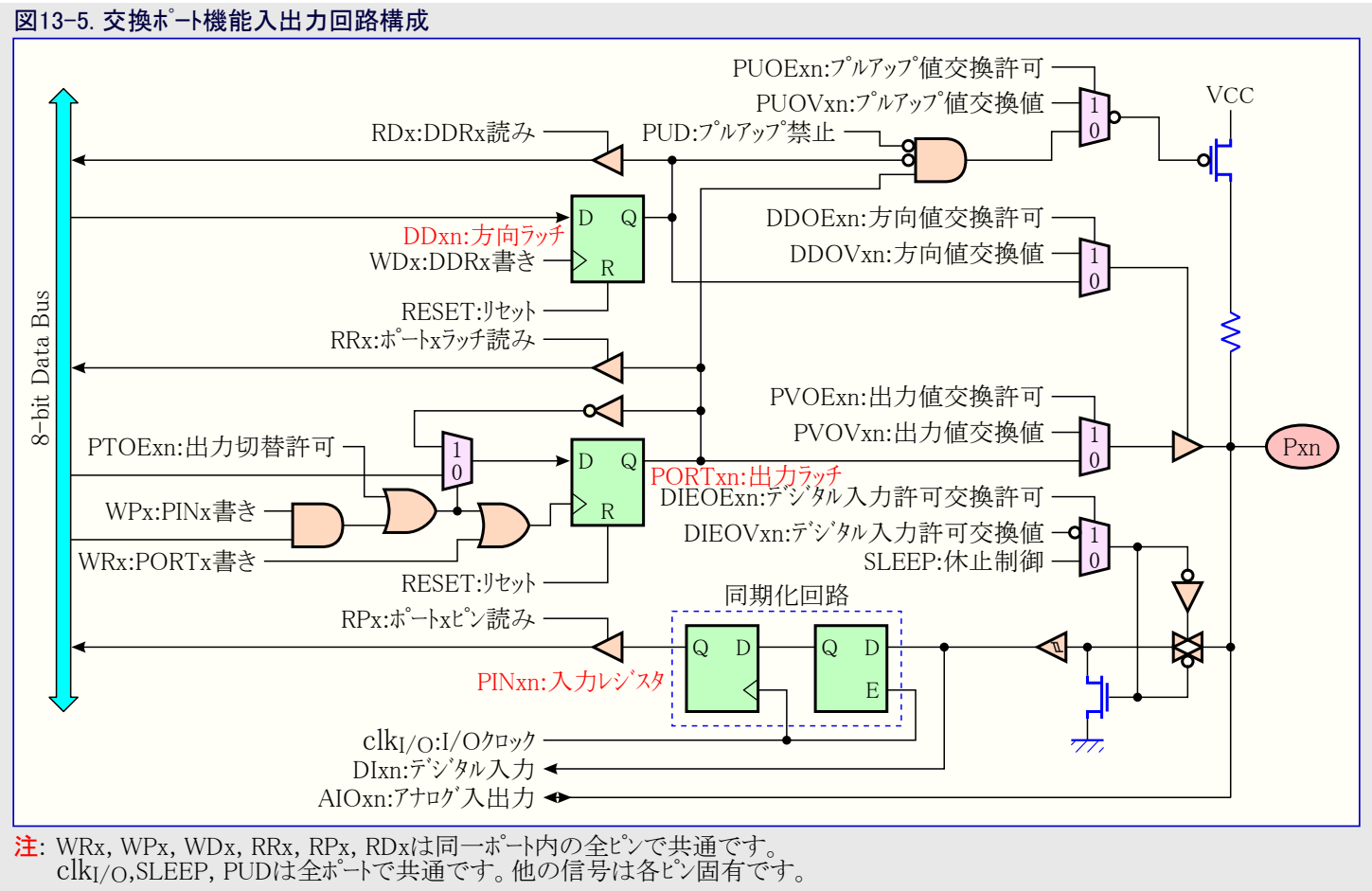


表13-2.は重複(交換)信号の機能一覧を示します。図13-5.で示すピンとポートは次表で示されません。重複(交換)信号は交換機能を持つ機能部で内部的に生成されます。

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。

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

13.3.1. ホートAの交換機能

ポートAには電圧A/D変換器への入力ピンとしての交換機能があります。

表13-3. ホートA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
PA7	INT3 (外部割り込み要求3入力) PCINT7 (ピン変化割り込み7入力)	PA3	ADC3 (電圧A/D変換器アナログ入力チャネル3) PCINT3 (ピン変化割り込み3入力)
PA6	INT2 (外部割り込み要求2入力) PCINT6 (ピン変化割り込み6入力)	PA2	ADC2 (電圧A/D変換器アナログ入力チャネル2) PCINT2 (ピン変化割り込み2入力)
PA5	INT1 (外部割り込み要求1入力) PCINT5 (ピン変化割り込み5入力)	PA1	ADC1 (電圧A/D変換器アナログ入力チャネル1) PCINT1 (ピン変化割り込み1入力)
PA4	INT0 (外部割り込み要求0入力) ADC4 (電圧A/D変換器アナログ入力チャネル4) PCINT4 (ピン変化割り込み4入力)	PA0	ADC0 (電圧A/D変換器アナログ入力チャネル0) 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。

・ INT3～0/ADC4/PCINT7～4 – ホートA ビット7～4 : PA7～4

INT3～0: 外部割り込み要求3～0入力。PA7～4ピンはMCUへの外部割り込み元として扱えます。

ADC4: 電圧A/D変換器アナログ入力チャネル4入力(PA4ピンのみ)。

PCINT7～4: ピン変化割り込み7～4入力。PA7～4ピンはMCUへの外部割り込み元として扱えます。

・ ADC3～0/PCINT3～0 – ホートA ビット3～0 : PA3～0

ADC3～0: 電圧A/D変換器アナログ入力チャネル3～0入力。

PCINT3～0: ピン変化割り込み3～0入力。PA3～0ピンはMCUへの外部割り込み元として扱えます。

表13-4.と表13-5.はポートAの交換機能を43頁の図13-5.で示される交換信号に関連付けます。

表13-4. ホートA7～4の交換機能用交換信号

信号名	PA7/INT3/PCINT7	PA6/INT2/PCINT6	PA5/INT1/PCINT5	PA4/INT0/ADC4/PCINT4
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	INT3許可	INT2許可	INT1許可	INT0許可
DIOV	INT3許可	INT2許可	INT1許可	INT0許可
DI	INT3/PCINT7入力	INT2/PCINT6入力	INT1/PCINT5入力	INT0/PCINT4入力
AIO	–	–	–	ADC4入力

表13-5. ホートA3～0の交換機能用交換信号

信号名	PA3/ADC3/PCINT3	PA2/ADC2/PCINT2	PA1/ADC1/PCINT1	PA0/ADC0/PCINT0
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	PCINT3入力	PCINT2入力	PCINT1入力	PCINT0入力
AIO	ADC3入力	ADC2入力	ADC1入力	ADC0入力

13.3.2. ポートBの交換機能

ポートBピンの交換機能は表13-6.で示されます。

表13-6. 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	OC0B (タイマ/カウンタ0 比較B一致/PWM出力) PCINT15 (ピン変化割り込み15入力)	PB3	TCK (JTAG クロック入力) PCINT11 (ピン変化割り込み11入力)
PB6	OC0A (タイマ/カウンタ0 比較A一致/PWM出力) PCINT14 (ピン変化割り込み14入力)	PB2	TMS (JTAG 検査種別選択入力) PCINT10 (ピン変化割り込み10入力)
PB5	PCINT13 (ピン変化割り込み13入力)	PB1	TDI (JTAG 検査データ入力) PCINT9 (ピン変化割り込み9入力)
PB4	PCINT12 (ピン変化割り込み12入力)	PB0	TDO (JTAG 検査データ出力) 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。

- OC0B/PCINT15 – ポートB ビット7 : PB7

OC0B: タイマ/カウンタ0の比較B一致出力。PB7ピンはタイマ/カウンタ0の比較B一致用外部出力として扱えます。この機能を扱うため、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OC0Bピンはタイマ機能のPWM動作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MCUへの外部割り込み元として扱えます。

- OC0A/PCINT14 – ポートB ビット6 : PB6

OC0A: タイマ/カウンタ0の比較A一致出力。PB6ピンはタイマ/カウンタ0の比較A一致用外部出力として扱えます。この機能を扱うため、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OC0Aピンはタイマ機能のPWM動作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MCUへの外部割り込み元として扱えます。

- PCINT13 – ポートB ビット5 : PB5

PCINT13: ピン変化割り込み13入力。PB5ピンはMCUへの外部割り込み元として扱えます。

- PCINT12 – ポートB ビット4 : PB4

PCINT12: ピン変化割り込み12入力。PB4ピンはMCUへの外部割り込み元として扱えます。

- TCK/PCINT11 – ポートB ビット3 : PB3

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。

PCINT11: ピン変化割り込み11入力。PB3ピンはMCUへの外部割り込み元として扱えます。

- TMS/PCINT10 – ポートB ビット2 : PB2

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。

PCINT10: ピン変化割り込み10入力。PB2ピンはMCUへの外部割り込み元として扱えます。

- TDI/PCINT9 – ポートB ビット1 : PB1

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。

PCINT9: ピン変化割り込み9入力。PB1ピンはMCUへの外部割り込み元として扱えます。

- TDO/PCINT8 – ポートB ビット0 : PB0

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。

PCINT8: ピン変化割り込み8入力。PB0ピンはMCUへの外部割り込み元として扱えます。

表13-7と表13-8はポートBの交換機能を43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B7～4の交換機能用交換信号

信号名	PB7/OC0B/PCINT15	PB6/OC0A/PCINT14	PB5/PCINT13	PB4/PCINT12
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	OC0B許可	OC0A許可	0	0
PVOV	OC0B	OC0A	0	0
PTOE	–	–	–	–
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	PCINT15入力	PCINT14入力	PCINT13入力	PCINT12入力
AIO	–	–	–	–

表13-8. ポートB3～0の交換機能用交換信号

信号名	PB3/TCK/PCINT11	PB2/TMS/PCINT10	PB1/TSI/PCINT9	PB0/TDO/PCINT8
PUOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
PUOV	1	1	1	0
DDOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
DDOV	0	0	0	IR移動+DR移動
PVOE	0	0	0	JTAGEN
PVOV	0	0	0	TDO
PTOE	–	–	–	–
DIEOE	JTAGEN	JTAGEN	JTAGEN	JTAGEN
DIEOV	0	0	0	0
DI	TCK/PCINT11入力	TMS/PCINT10入力	TDI/PCINT9入力	PCINT8入力
AIO	–	–	–	–

13.3.3. ポートDの交換機能

ポートDピンの交換機能は表13-9で示されます。

交換ピンの設定は次のとおりです。

- T0 – ポートDビット0 : PD0

T0: [タイマ/カウンタ0](#)の外部クロック入力ピンです。

表13-10はポートDの交換機能を43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Dピンの交換機能

ポートピン	交換機能
PD0	T0 (タイマ/カウンタ0 外部クロック入力)

表13-10. ポートD1～0の交換機能用交換信号

信号名	PD1	PD0/T0
PUOE	0	0
PUOV	0	0
DDOE	0	0
DDOV	0	0
PVOE	0	0
PVOV	0	0
PTOE	–	–
DIEOE	0	0
DIEOV	0	0
DI	–	T0入力
AIO	–	–

13.4. 低電圧I/Oポート用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	JTD	–	–	PUD	–	–	IVSEL	IVCE	MCUCR
Read/Write	R/W	R	R	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41頁の「[ピンの設定](#)」をご覧ください。

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

ビット	7	6	5	4	3	2	1	0	
\$02 (\$22)	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$01 (\$21)	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

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

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

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

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

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

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

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	–	–	–	–	–	–	PORTD1	PORTD0	PORTD
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$0A (\$2A)	–	–	–	–	–	–	DDD1	DDD0	DDRD
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$09 (\$29)	–	–	–	–	–	–	PIND1	PIND0	PIND
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	

14. 高電圧入出力ポート

全てのAVRの高電圧ポートは標準デジタルI/Oポートとして使われる時に真の読み-変更-書き(リード・モディファイ・ライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方角をも無意識に変更することなく、1つのポートピンの方角を変更できることを意味します。全ての高電圧I/Oピンは図14-1.で示される、GNDへの保護ツェナーダイオードを持ちます。各値の完全な一覧表については145頁の「電気的特性」を参照してください。

本項内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記されたPORTxnがポートCのビット3に対してはPORTC3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「高電圧出力ポート用レジスタ」で一覧されます。

出力レジスタ(PORTx)の各高電圧ポートに対して1つI/Oメモリアドレス位置が割り当てられます。出力レジスタは読み書きされます。

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高電圧ポート」で記述されます。

14.1. 標準デジタル入出力としての高電圧ポート

この高電圧ポートは耐高電圧のオープンドレイン出力ポートです。図14-2.はここで属にPxnと呼ばれるI/Oポートピンの1つの機能説明を示します。

図14-1. 高電圧入出力ピン等価回路

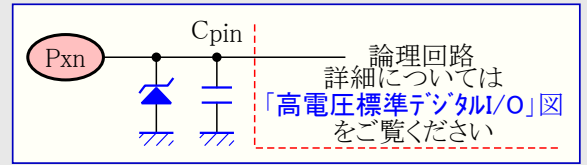
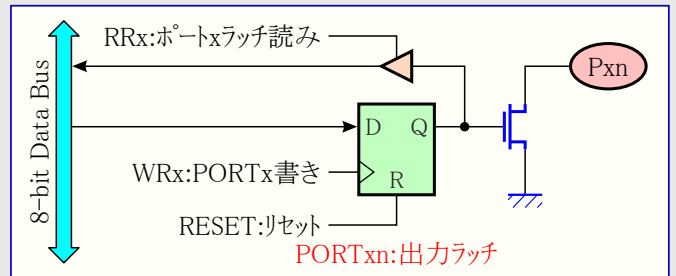


図14-2. 高電圧標準デジタル入出力回路構成



注: WRxとRRxは同一ポート内の全ピンで共通です。

14.2. ピンの設定

各ポートピンは1つのレジスタビット(PORTxn)を持ちます。「高電圧出力ポート用レジスタ」で示されるように、このPORTxnビットはPORTx I/Oアドレスでアクセスされます。PORTxnが論理1を書かれると、そのポートピンはLow(0)に駆動されます。PORTxnが論理0を書かれると、そのポートピンはHi-Zにされます。ポートピンは例えクロックが動いていなくても、リセット条件が活性(有効)になるとHi-Zにされます。

14.3. 高電圧出力ポート用レジスタ

14.3.1. PORTC - ポートC出力レジスタ (Port C Data Register)

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	—	—	—	—	—	—	—	PORTC0	PORTC
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

15. 8ビット タイマ/カウンタ0 (PWM付き)

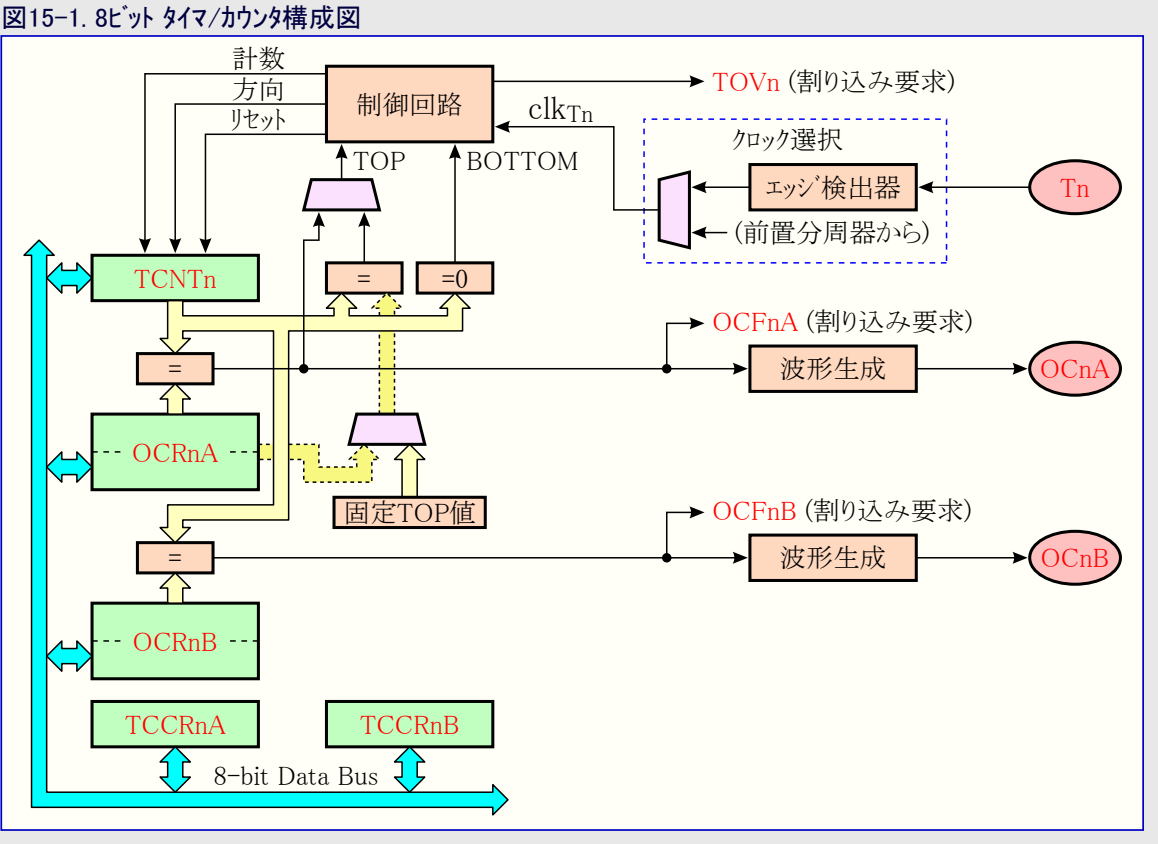
タイマ/カウンタ0は2つの独立した比較出力部とPWM支援付きの汎用8ビット タイマ/カウンタ部です。それは正確なプログラム実行タイミング(事象管理)、波形生成を許します。主な特徴は次のとおりです。

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

15.1. 概要

この8ビット タイマ/カウンタの簡単化した構成図は図15-1.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58頁の「8ビット タイマ/カウンタ0用レジスタ」で一覧されます。

24頁の「PRR0 – 電力削減レジスタ0」のPRTIM0ビットはタイマ/カウンタ0部を許可するために0を書かれなければなりません。



15.1.1. 定義

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

表15-1.の定義は本資料を通して広範囲に渡って使われます。

表15-1. 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。

15.1.2. 関係レジスタ

タイマ/カウンタ(TCNT0)と比較レジスタ(OCR0AとOCR0B)は8ビットのレジスタです。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR0)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK0)で個別に遮蔽(禁止)されます。TIFR0とTIMSK0はこの図で示されません。

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

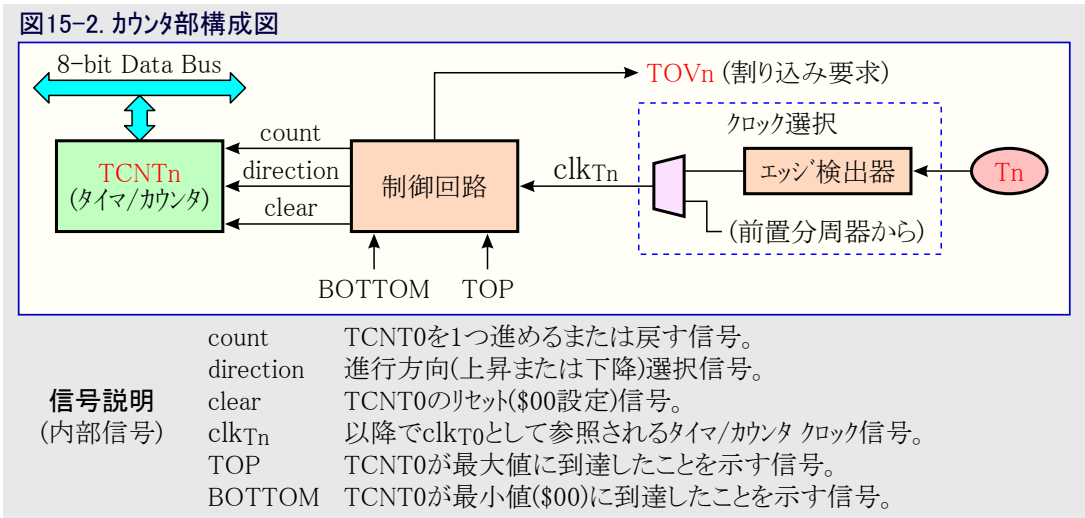
2重緩衝化した比較レジスタ(OCR0AとOCR0B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC0AとOC0B)ピンでPWMまたは可変周波数出力を作成するための波形生成器によって使えます。詳細については52頁の「比較出力部」をご覧ください。この比較一致発生は比較一致割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF0AとOCF0B)も設定(1)します。

15.2. タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR0B)に配置されたクロック選択(CS02~0)ビットによって制御されるクロック選択論理回路で選択されます。クロック元と前置分周器の詳細については68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。

15.3. 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15-2.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存して、カウンタは各タイマ/カウンタクロック(clkT0)で解除(\$00)、増加(+1)、または減少(-1)されます。clkT0はクロック選択(CS02~0)ビットによって選択された内部または外部のクロック元から生成できます。クロック元が選択されない(CS02~0=000)とき、タイマ/カウンタは停止されます。けれどもTCNT0値はタイマ/カウンタクロック(clkT0)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先順位を持ちます)。

計数順序(方法)はタイマ/カウンタ0制御レジスタA(TCCR0A)に配置された波形生成種別(WGM01,0)ビットとタイマ/カウンタ0制御レジスタB(TCCR0B)に配置された波形生成種別(WGM02)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOC0A/OC0B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54頁の「動作種別」をご覧ください。

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

15.4. 比較出力部

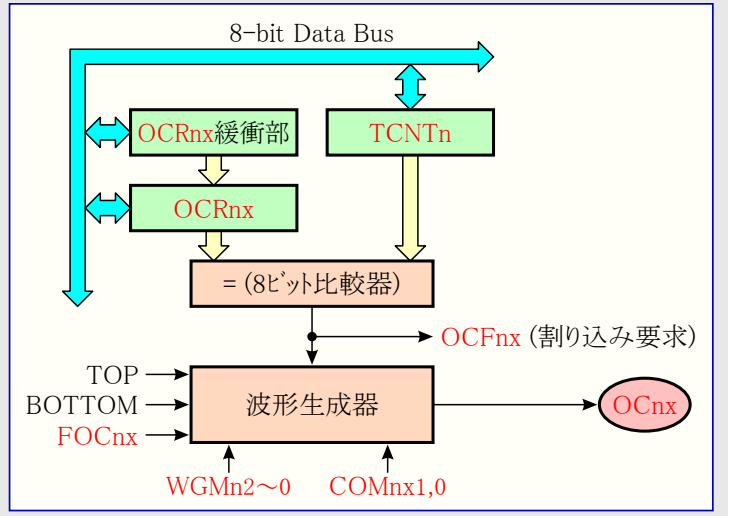
この8ビット比較器はTCNT0と比較レジスタ(OCR0AとOCR0B)を継続的に比較します。TCNT0がOCR0AまたはOCR0Bと等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF0AまたはOCF0B)を設定(1)します。対応する割り込みが許可(I=1, OCIE0AまたはOCIE0B=1)されているならば、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットによって設定された動作種別に従った出力を生成するのにこの一致信号を使います。MAXとBOTTOM信号は動作種別(54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15-3.は比較出力部の構成図を示します。

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

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

図15-3. 比較出力部構成図



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

15.4.1. 強制比較出力

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

15.4.2. 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OCR0xに許します。

15.4.3. 比較一致部の使用

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

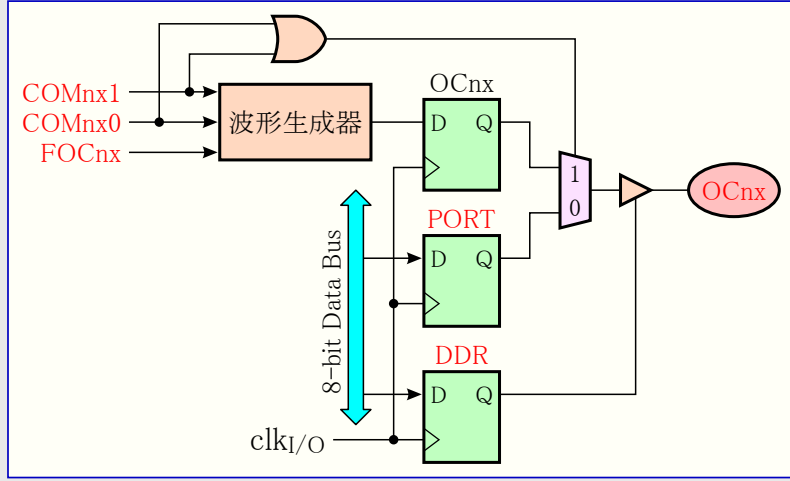
OC0xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC0x値を設定する一番簡単な方法は標準動作で強制変更(FOC0x)スローブビットを使うことです。波形生成動作種別間を変更する時でも、OC0x(内部)レジスタはその値を保ちます。

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

15.5. 比較一致出力部

比較出力選択($COM0x1,0$)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力($OC0x$)状態の定義に $COM0x1,0$ ビットを使います。また $COM0x1,0$ ビットは $OC0x$ ピン出力元を制御します。図15-4は $COM0x1,0$ ビット設定によって影響を及ぼされる論理回路の簡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。 $COM0x1,0$ ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。 $OC0x$ の状態を参照するとき、その参照は $OC0x$ ピンでなく内部 $OC0x$ レジスタに対してです。システムリセットが起ると、 $OC0x$ レジスタは'0'にリセットされます。

図15-4. 比較一致出力回路図



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

比較出力ピン論理回路の設計は出力が許可される前の $OC0x$ 状態の初期化を許します。いくつかの $COM0x1,0$ ビット設定が或る種の動作種別に対して予約されることに注意してください。58頁の「8ビット タイマ/カウンタ用レジスタ」をご覧ください。

15.5.1. 比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作で $COM0x1,0$ ビットを違うふうに使います。全ての動作種別に対して $COM0x1,0=00$ 設定は次の比較一致で実行すべき $OC0x$ レジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については58頁の表15-2と表15-5を参照してください。高速PWM動作については58頁の表15-3と表15-6、位相基準PWMについては58頁の表15-4と表15-7を参照してください。

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

15.6. 動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は波形生成種別(WGM02~0)ビットと比較出力選択(COM0x1,0)ビットの組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM0x1,0ビットは生成されるPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対するCOM0x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します(53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57頁の「タイマ/カウンタ0のタイミング」を参照してください。

15.6.1. 標準動作

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

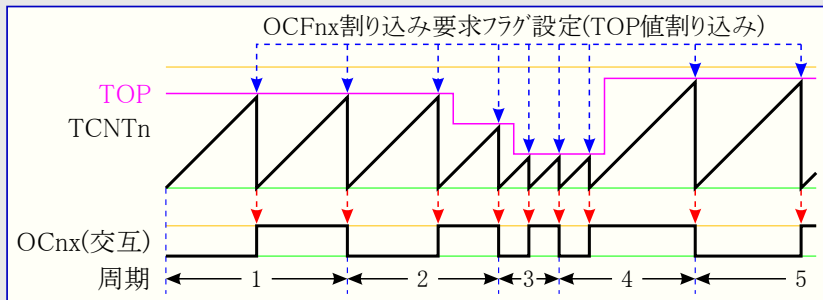
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

15.6.2. 比較一致タイマ/カウンタ解除(CTC)動作

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

CTC動作についてのタイミング図は図15-5.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15-5. CTC動作タイミング



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

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

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

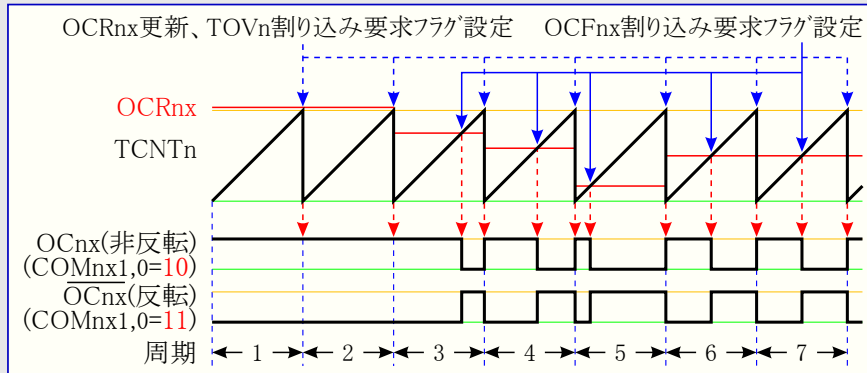
標準動作と同じように、タイマ/カウンタ溢れ(TOV0)フラグはカウンタがMAXから\$00へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

15.6.3. 高速PWM動作

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

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がTOP値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は図15-6.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。赤細線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF0x)はOCR0x=TOPを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。

図15-6. 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TOPに到達する時毎に設定(1)されます。割り込みが許可されるなら、その割り込み処理ルーチンは比較値を更新するのに使えます。

高速PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0を'11'に設定することで生成できます。WGM02ビットが設定(1)ならば、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(58頁の表15-3と表15-6をご覧ください)。実際のOC0x値はポートピンに対するデータ方向(DDR_OC0x)が出力として設定される場合だけ見えるでしょう。PWM波形はTCNT0とOCR0x間の比較一致で、OC0x(内部)レジスタを設定(1)(または周期(0))と、カウンタが解除(\$00,TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0xレジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

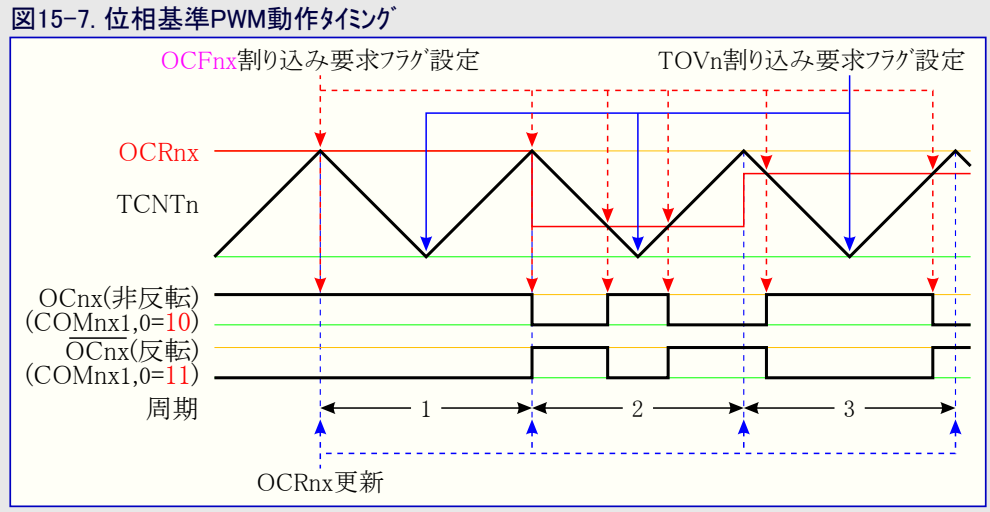
OCR0xの両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合にあたります。OCR0xがBOTTOM(\$00)と等しく設定されると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。OCR0xがTOPに等しく設定されると、(COM0x1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

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

15.6.4. 位相基準PWM動作

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

位相基準PWM動作でのカウンタはカウンタ値がTOPと一致するまで増加されます。カウンタはTOPに達すると計数方向を変更します。このTCNT0値は1タイマ/カウンタクロック周期間TOPと等しくなります。位相基準PWM動作のタイミング図は図15-7.で示されます。TCNT0値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。



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

位相基準PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM0x1,0ビットを'11'に設定することで生成できます。WGM02ビットが設定(1)なら、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(58頁の表15-4.と表15-7.をご覧ください)。実際のOC0x値はそのポートピンに対するデータ方向(DDR_OC0x)が出力として設定される場合だけ見えてでしょう。PWM波形はカウンタが増加する時のTCNT0とOCR0x間の比較一致でOC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT0とOCR0x間の比較一致でOC0xレジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

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

図15-7.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15-7.のようにOCR0xはTOPからその値を変更します。OCR0x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0x値よりも高い値から数え始め、そしてその理屈のために比較一致、それ故上昇途中で起こされるであろうOCnxの変更を逃します。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)

15.7. タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従って以下の図でタイマ/カウンタクロック(clk_{T0})がクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時の情報を含みます。図15-8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15-8. 前置分周なし(1/1)のタイマ/カウンタ タイミング

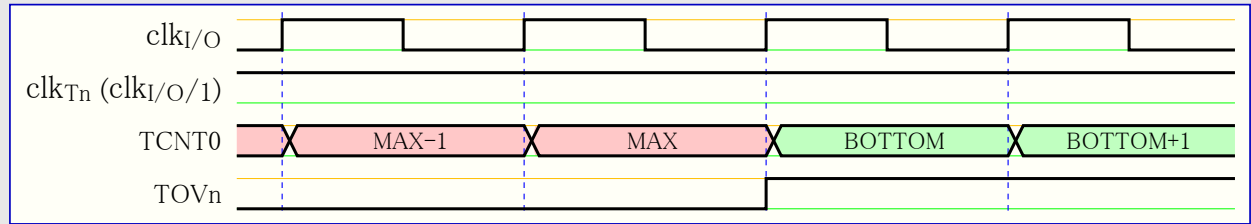


図15-9. は同じタイミング データを示しますが、前置分周器が許可されています。

図15-9. 前置分周器($f_{\text{clk}_{I/O}}/8$)のタイマ/カウンタ タイミング

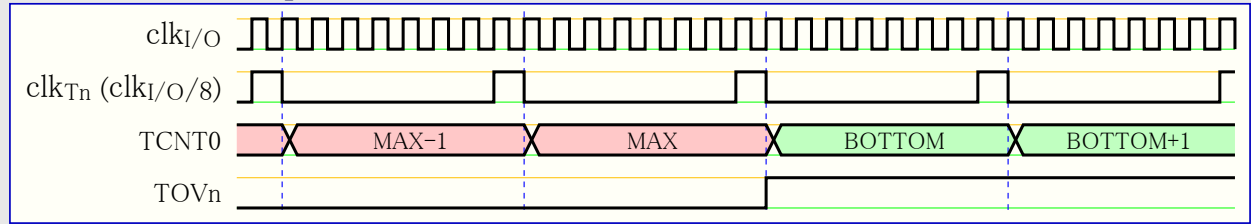


図15-10. はCTC動作とOCR0AがTOPのPWM動作を除く全動作種別でのOCF0Aと全動作種別でのOCF0Bの設定を示します。

図15-10. 前置分周器($f_{\text{clk}_{I/O}}/8$)のタイマ/カウンタ、OCF0x設定 タイミング

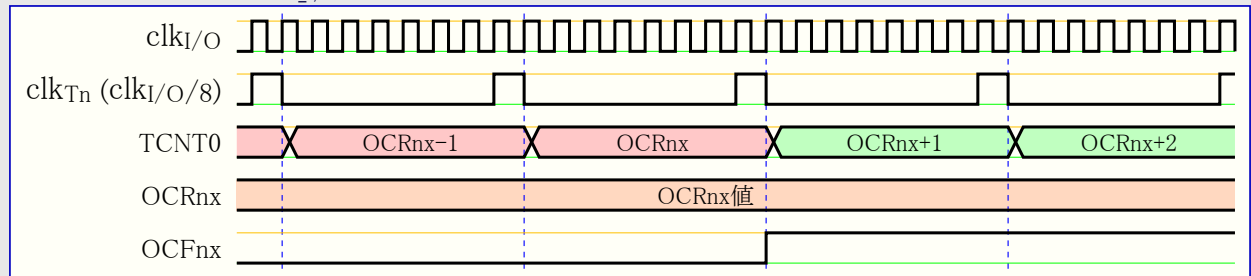
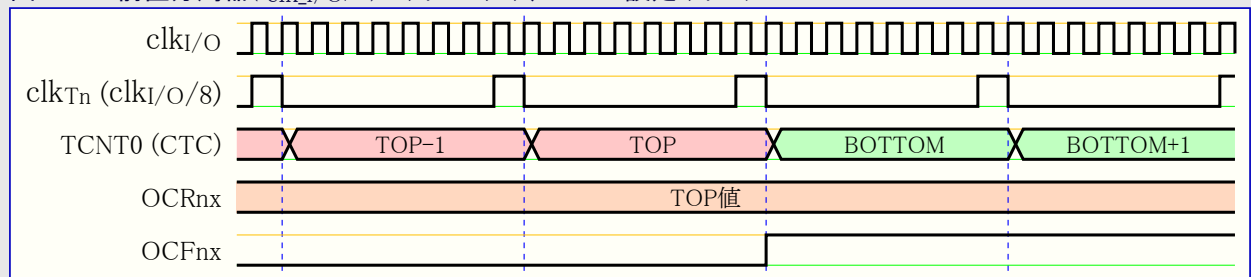


図15-11. はOCR0AがTOPの高速PWM動作と、CTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図15-11. 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



15.8. 8ビット タイマ/カウンタ0用レジスタ

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

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

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

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

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

表15-2.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0A1,0ビット機能を示します。

表15-3.はWGM02~0ビットが高速PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表15-4.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表15-2. 非PWM動作比較A出力選択

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

表15-4. 位相基準PWM動作比較A出力選択 (共通注意参照)

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

表15-3. 高速PWM動作比較A出力選択 (共通注意参照)

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

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

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

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

表15-5.はWGM02~0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM0B1,0ビット機能を示します。

表15-6.はWGM02~0ビットが高速PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表15-7.はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0B1,0ビットの機能を示します。

表15-5. 非PWM動作比較B出力選択

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

表15-7. 位相基準PWM動作比較B出力選択 (共通注意参照)

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

表15-6. 高速PWM動作比較B出力選択 (共通注意参照)

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

共通注意: COM0x1が設定(1)され、対応するOCR0xがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMまたはTOPでの設定(1)または解除(0)は行われます。より多くの詳細については55頁の「高速PWM動作」または56頁の「位相基準PWM動作」をご覧ください。(表15-3,4,6,7.各々での注を纏めました。)

●ビット3,2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

タイマ/カウンタ制御レジスタB(TCCR0B)で得られるWGM02ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元、使われるべき波形生成のどの形式かを制御します(表15-8参照)。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。54頁の「動作種別」をご覧ください。

表15-8. 波形生成種別選択							
番号	WGM02	WGM01	WGM00	タイマ/カウンタ動作種別	TOP値	OCR0x更新時	TOV0設定時
0	0	0	0	標準動作	\$FF	即時	MAX
1	0	0	1	8ビット位相基準PWM動作	\$FF	TOP	BOTTOM
2	0	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR0A	即時	MAX
3	0	1	1	8ビット高速PWM動作	\$FF	BOTTOM	MAX
4	1	0	0	(予約)	-	-	-
5	1	0	1	位相基準PWM動作	OCR0A	TOP	BOTTOM
6	1	1	0	(予約)	-	-	-
7	1	1	1	高速PWM動作	OCR0A	BOTTOM	TOP

注: MAX=\$FF、BOTTOM=\$00です。

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

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

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

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

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

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

FOC0Aビットは常に0として読まれます。

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

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

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

FOC0Bスロープは何れの割り込みの生成も行いません。

FOC0Bビットは常に0として読まれます。

●ビット5,4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

58頁の「TCCR0A – タイマ/カウンタ制御レジスタA」のWGM01,0ビット記述をご覧ください。

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

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

表15-9. タイマ/カウンタ0入力クロック選択			
CS02	CS01	CS00	意味
0	0	0	停止 (タイマ/カウンタ0動作停止)
0	0	1	clkI/O (前置分周なし)
0	1	0	clkI/O/8 (8分周)
0	1	1	clkI/O/64 (64分周)
1	0	0	clkI/O/256 (256分周)
1	0	1	clkI/O/1024 (1024分周)
1	1	0	T0ピンの下降端 (外部クロック)
1	1	1	T0ピンの上昇端 (外部クロック)

外部ピン(クロック)動作がタイマ/カウンタ0に対して使われる場合、例えT0ピンが出力として設定されても、T0ピンの遷移はカウンタをクロック駆動します。この特徴がソフトウェアに計数制御を許します。

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

ビット	7	6	5	4	3	2	1	0	
\$26 (\$46)	(MSB)							(LSB)	TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

ビット	7	6	5	4	3	2	1	0	
\$27 (\$47)	(MSB)							(LSB)	OCR0A
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

15.8.5. OCR0B – タイマ/カウンタ0 比較Bレジスタ (Timer/Counter0 Output Compare B Register)

ビット	7	6	5	4	3	2	1	0	
\$28 (\$48)	(MSB)							(LSB)	OCR0B
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

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

- ビット7～3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読めます。

- ビット2 – OCIE0B : タイマ/カウンタ0比較B割り込み許可 (Timer/Counter0 Output Compare Match B Interrupt Enable)

OCIE0Bビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較B一致割り込みが許可されます。タイマ/カウンタ0で比較B一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)で比較B割り込み要求フラグ(OCF0B)が設定(1)されると、対応する割り込みが実行されます。

- ビット1 – OCIE0A : タイマ/カウンタ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A一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)で比較A割り込み要求フラグ(OCF0A)が設定(1)されると、対応する割り込みが実行されます。

- ビット0 – TOIE0 : タイマ/カウンタ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。

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

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

- ビット7～3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読めます。

- ビット2 – OCF0B : タイマ/カウンタ0比較B割り込み要求フラグ (Timer/Counter0, Output Compare B Match Flag)

OCF0Bビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0B)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF0Bはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0Bは解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ0割り込み許可レジスタ(TIMSK0)のタイマ/カウンタ0比較B一致割り込み許可(OCIE0B)ビット、OCF0Bが設定(1)されると、タイマ/カウンタ0比較B一致割り込みが実行されます。

- ビット1 – OCF0A : タイマ/カウンタ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A一致割り込み許可(OCIE0A)ビット、OCF0Aが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが実行されます。

- ビット0 – TOV0 : タイマ/カウンタ0溢れ割り込み要求フラグ (Timer/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。

これらフラグの設定はWGM02～0ビット設定に依存します。59ページの波形生成種別ビット記述の表15-8を参照してください。

16. 16ビット タイマ/カウンタ1

この16ビット タイマ/カウンタ部は正確なプログラム実行タイミング(事象管理)を許します。主な特徴は次のとおりです。

- 1つの比較出力部
- 比較一致でのタイマ/カウンタ解除(自動再設定)
- 2つの独立した割り込み (TOV1,OCF1A)

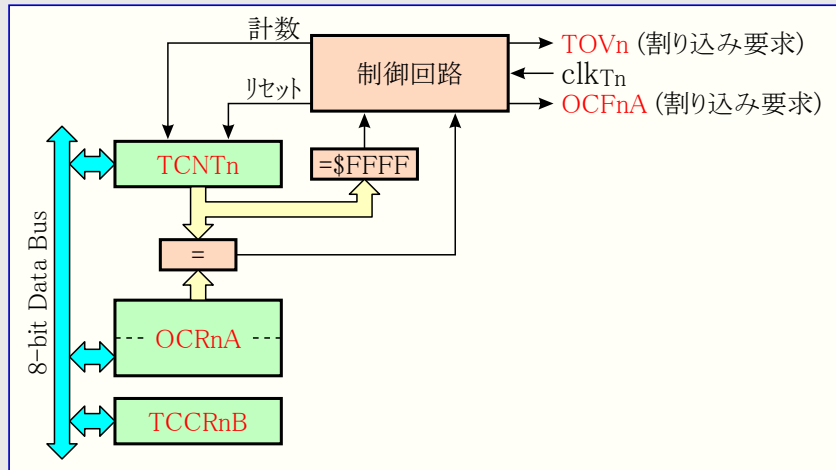
16.1. 概要

本資料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)。デバイス仕様のI/Oレジスタとビット位置は66頁の「16ビット タイマ/カウンタ1用レジスタ」で示されます。

この16ビット タイマ/カウンタの単純化した構成図は図16-1.で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注: 原文太字)で示されます。

24頁の「PRR0 - 電力削減レジスタ0」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16-1. 16ビット タイマ/カウンタ構成図



16.1.1. 関係レジスタ

タイマ/カウンタ(TCNT1)と比較レジスタ(OCR1A)は両方とも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63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(TCCR1B)は8ビットレジスタで、CPUアクセスの制限はありません。割り込み要求信号はタイマ/カウンタ1割り込み要求レジスタ(TIFR1)で全て見えます。両割り込みはタイマ/カウンタ1割り込み許可レジスタ(TIMSK1)で個別に遮蔽(禁止)されます。TIFR1とTIMSK1はこの図で示されません。

このタイマ/カウンタは内部的に前置分周器経由によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加するのに使うクロック元を制御します。クロック元が選択されないと、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。

比較レジスタ(OCR1A)はタイマ/カウンタ値と常に比較されます。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)を設定(1)します。

16.2. 16ビットレジスタのアクセス

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

アセンブリ言語プログラム例

```

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

C言語プログラム例

```

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

注: 5頁の「コード例について」をご覧ください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

16ビットレジスタアクセスが非分断操作である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のどの読み込みも行えます。

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 5頁の「コード例について」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のどの書き込みも行えます。

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 5頁の「コード例について」をご覧ください。

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

16.2.1. 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。

16.3. タイマ/カウンタのクロック

このタイマ/カウンタは内部クロック元によってクロック駆動されます。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)ビットで制御されるクロック選択論理回路によって選択されます。クロック元と前置分周器の詳細については68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。

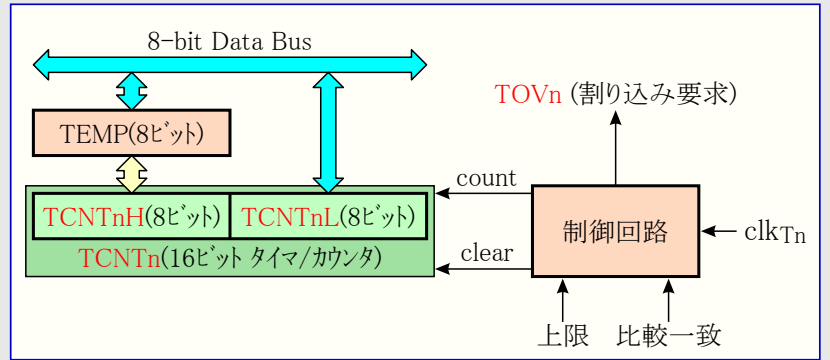
16.4. 計数器部

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

この16ビット カウンタはカウンタの上位8ビットを含むカウンタ上位(TCNT1H)と下位8ビットを含むカウンタ下位(TCNT1L)の2つの8ビット I/Oメモリ位置に配置されます。TCNT1HレジスタはCPUによって間接的なアクセスのみできます。CPUがTCNT1H I/O位置をアクセスするとき、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 バス経由に於いて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)または増加(+1)されます。clkT1はクロック選択(CS12~0)ビットによって選択された内部クロック元から生成されます。クロック元が選択されない(CS12~0=000)時にカウンタは停止されます。けれどもTCNT1値はタイマ/カウンタ クロック(clkT1)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先権を持ちます)。

図16-2. カウンタ部構成図



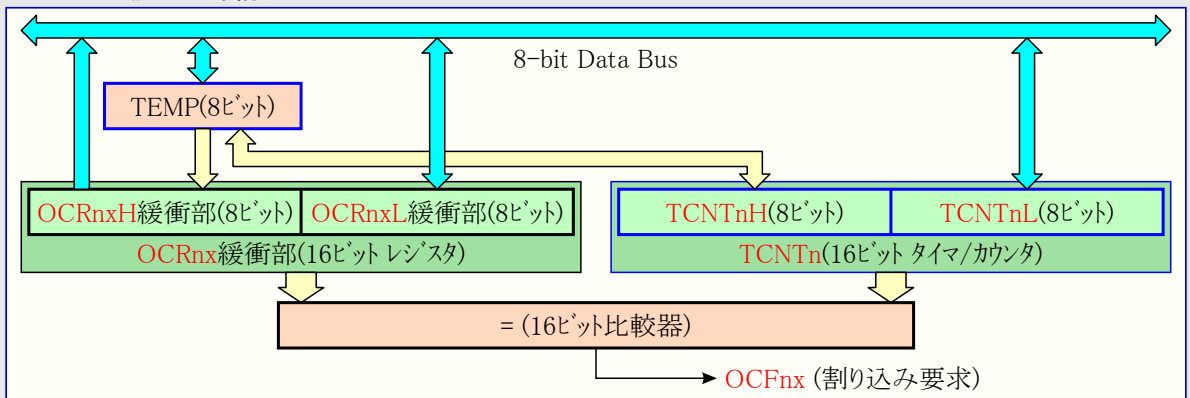
信号説明 (内部信号)
count TCNT1を1つ進めるまたは戻す信号。
clear TCNT1のリセット(\$0000設定)信号。
clkTn 以降でclkT1として参照されるタイマ/カウンタ クロック信号。

16.5. 比較出力部

この16ビット比較器はTCNT1と比較レジスタ(OCR1A)を継続的に比較します。TCNT1とOCR1A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(OCF1A)を設定(1)します。許可(I=1, OCIE1A=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。OCF1A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OCF1AはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。

図16-3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)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16-3. 比較出力部構成図



16.5.1. 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、TCNT1と同じ値に初期化されることをOCR1xに許します。

16.5.2. 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、どれかの比較出力部を使う場合にTCNT1を変更する時は危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は(発生せずに)失われます。

16.6. 16ビット タイマ/カウンタ1用レジスタ

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

ビット (\$81)	7	6	5	4	3	2	1	0	
	–	–	–	–	CTC1	CS12	CS11	CS10	TCCR1B
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として読めます。
- ビット3 – CTC1 : 一致解除許可 (Clear Timer/Counter1 on Compare Match)
CTC1制御ビットが設定(1)されると、タイマ/カウンタ1(TCNT1)は比較一致後のCPUクロック周期で\$0000にリセットされます。
- ビット2～0 – CS12～0 : クロック選択1 (Clock Select1, bit 2,1 and 0)
この3つのクロック選択ビットはタイマ/カウンタ(TCNT1)によって使われるべきクロック元を選択します。

表16-1. タイマ/カウンタ1入力クロック選択							
CS12	CS11	CS10	意味	CS12	CS11	CS10	意味
0	0	0	停止 (タイマ/カウンタ1動作停止)	1	0	0	clk _{I/O} /64 (前置分周器64分周)
0	0	1	clk _{I/O} (前置分周器なし)	1	0	1	clk _{I/O} /128 (前置分周器128分周)
0	1	0	clk _{I/O} /8 (前置分周器8分周)	1	1	0	clk _{I/O} /256 (前置分周器256分周)
0	1	1	clk _{I/O} /32 (前置分周器32分周)	1	1	1	clk _{I/O} /1024 (前置分周器1024分周)

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

ビット (\$85)	15	14	13	12	11	10	9	8	
	(MSB)								TCNT1H
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$84)	7	6	5	4	3	2	1	0	
								(LSB)	TCNT1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この2つのタイマ/カウンタ I/O位置(TCNT1HとTCNT1Lを合わせたTCNT1)は、読み書き両方についてタイマ/カウンタ部の16ビット カウンタに直接アクセスします。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビット レジスタによって共用されます。63 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。

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

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

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を生成するのに使えます (誤注:共通性から本行追加)。

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。63 頁の「16ビット レジスタのアクセス」をご覧ください。

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

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

- ビット7～2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

- 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(33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(33頁の「割り込み」参照)が実行されます。

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

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

- ビット7～2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

- 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。

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)

TOV1フラグはタイマ/カウンタ1溢れ時に設定(1)されます。

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TOV1は解除(0)できます。

17. 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。

17.1. 内部クロック元

タイマ/カウンタはシステム クロック($CSn2 \sim 0 = 001$ 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タップの1つがクロック元として使えます。タイマ/カウンタ0設定については60頁の表15-9を、タイマ/カウンタ1設定については66頁の表16-1をご覧ください。前置分周したクロックは $f_{clk_I/O}/8$, $f_{clk_I/O}/32$, $f_{clk_I/O}/64$, $f_{clk_I/O}/128$, $f_{clk_I/O}/256$, $f_{clk_I/O}/1024$ の何れかの周波数を持ちます。

17.2. 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($CS02 \sim 0 = 5 \sim 2$ または $CS12 \sim 0 = 7 \sim 2$)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, 32, 64, 128, 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。

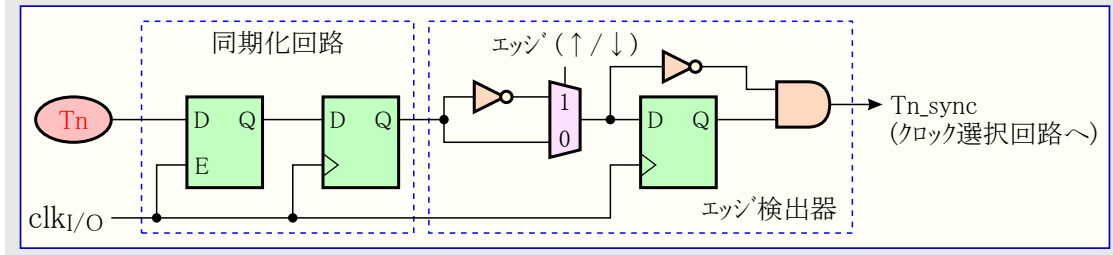
プログラム実行にタイマ/カウンタを同期することに対して前置分周器リセットを使うことが可能です。しかし、同じ前置分周器を共有する他のタイマ/カウンタも前置分周を使う場合、注意が必要とされなければなりません。前置分周器リセットはそれが接続される全タイマ/カウンタについての前置分周器周期に影響を及ぼします。

17.3. 外部クロック元

T0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})として使えます。このT0ピンはピン同期化論理回路によって全てのシステム クロック周期に一度採取されます。この同期化(採取)された信号はその後エッジ検出器を通して通過されます。図17-1はT0同期化とエッジ検出器論理回路の機能等価構成図を示します。レジスタは内部システム クロック($f_{clk_I/O}$)の上昇端でクロック駆動されます。ラッチは内部システム クロックのHigh区間で通過(Low区間で保持)です。

エッジ検出器は上昇端($CS02 \sim 0 = 111$)または下降端($CS02 \sim 0 = 110$)の検出毎に1つの clk_{T0} パルスを生成します。

図17-1. T0ピンの採取等価構成図



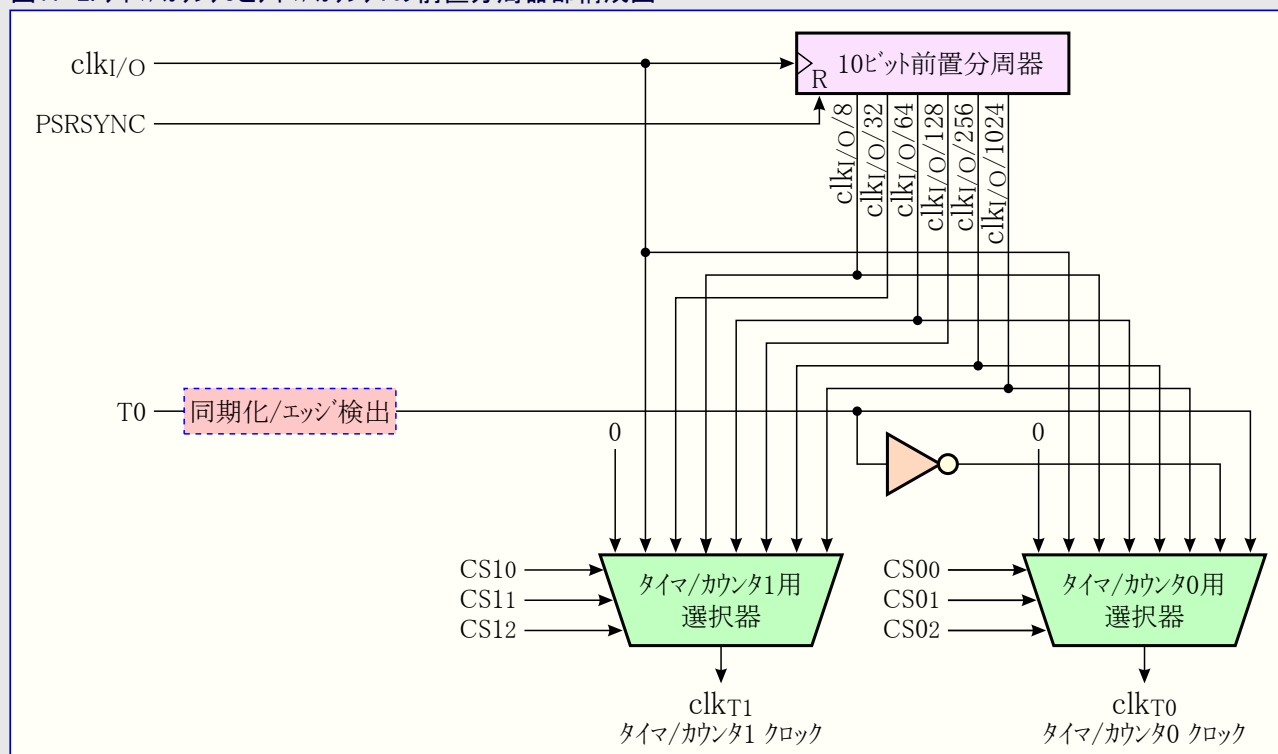
同期化とエッジ検出器論理回路はT0ピンへ印加されたエッジから系隙が更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

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

印加された外部クロックの各半周期は正しい採取を保証するために1システム クロック周期より長くなければなりません。この外部クロックは50%/50%デューティ比で与えられるものとして、システム クロック周波数の半分未満($f_{EXTclk} < f_{clk_I/O}/2$)であることが保証されなければなりません。エッジ検出器が採取を使うため、検出できる外部クロックの最大周波数は採取周波数の半分です(ナイquist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17-2. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器部構成図



注: 入力ピン(T0)の同期化/エッジ検出論理回路は図17-1.で示されます。

17.4. タイマ/カウンタ前置分周器用レジスタ

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

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

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

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

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

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

18. クーロン カウンタ (容量/残量計専用 $\Delta\Sigma$ A/D変換器)

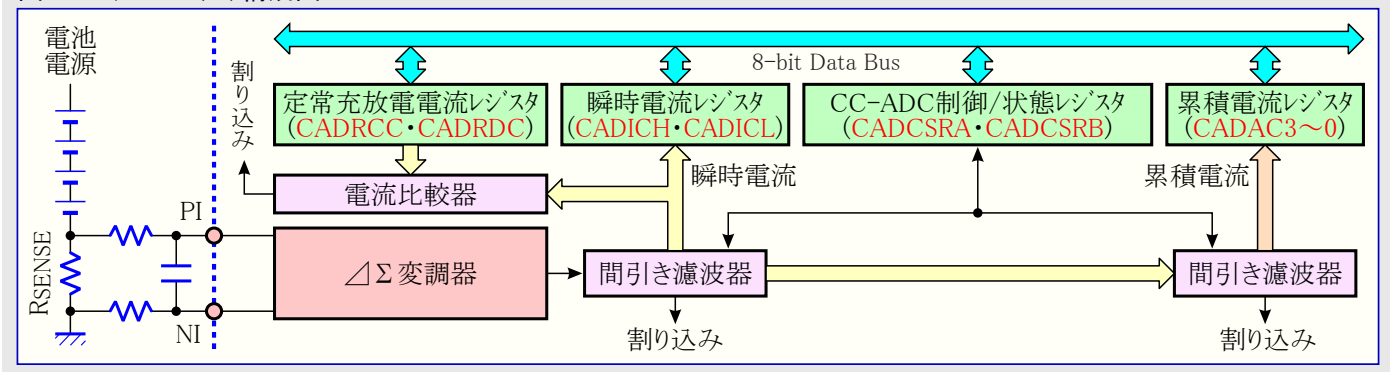
18.1. 特徴

- ・ 採取式クーロン カウンタ
- ・ クーロン カウンタ用に最適化された低電力 $\Delta\Sigma$ A/D変換器
- ・ 変換時間3.9msの瞬時電流出力
- ・ 設定可能な変換時間(125/250/500/1000ms)の累積電流出力
- ・ 5m Ω に於いて ± 30 Aを越える計測を許す ± 0.15 Vよりも大きな入力電圧範囲
- ・ 瞬時電流出力について53.7 μ V(5m Ω で10.7mA)に対応する(符号を含む)13ビット分解能
- ・ 累積電流出力について1.68 μ V(5m Ω で0.335mA)に対応する(符号を含む)18ビット分解能
- ・ A/D変換器に関して10 μ V未満の入力変位(オフセット)
- ・ 瞬時電流変換完了での割り込み
- ・ 累積電流変換完了での割り込み
- ・ 設定変更可能な比較値と採取間隔(250/500/1000/2000ms)での定常電流割り込み

ATmega406は外部検出抵抗器(RSENSE)を通して流れる充放電電流を採取するための、クーロン カウンタ用に最適化された専用の $\Delta\Sigma$ A/D変換器(CC-ADC)が特徴です。瞬時電流と累積電流の2つの異なる出力値が提供されます。瞬時電流出力はより低い分解能への犠牲による短い変換時間です。累積電流出力はクーロン カウンタ用の高精度電流計測を提供します。

クーロン カウンタの採取は高精度と柔軟な(解決)方法を提供します。精度は変換時間と容易に取引できます。これは定常電流検出も提供します。これは小さな充放電電流が流れる時の「パワーセーブ動作」での超低電力動作を許します。

図18-1. クーロン カウンタ構成図

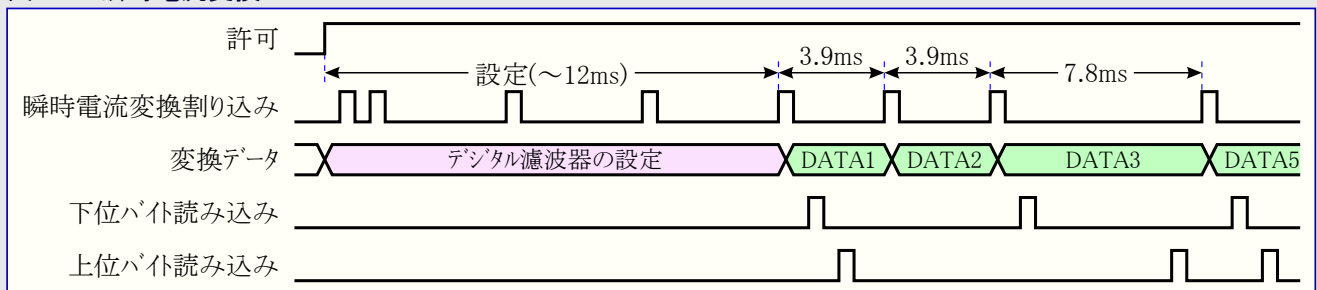


18.2. 操作

許可時、クーロン カウンタA/D変換器(CC-ADC)は外部検出抵抗器(RSENSE)上の電圧を継続的に計測します。

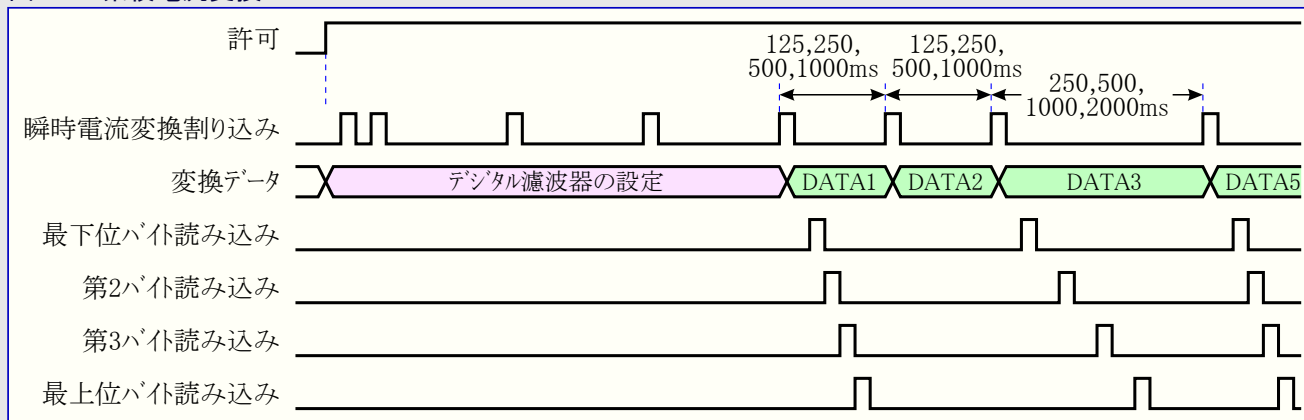
瞬時電流用の変換時間は入力信号直後での出力を可能にする3.9msに固定されています。各瞬時電流変換後、その割り込みが許可されていれば割り込みが生成されます。変換からのデータは割り込みが与えられるのと同時に瞬時電流(CADICH,CADICL)レジスタで更新されます。変換データの消失を避けるため、対応する割り込みが与えられた後、3.9ms時間窓内に上下バイト両方が読まれなければならない。下位バイトレジスタが読まれると、瞬時電流レジスタ更新と割り込みは上位バイトが読まれるまで停止されます。図18-2は制限時間内にDATA3読み込みが完了されないためにDATA4が失われる瞬時電流変換構成図を示します。

図18-2. 瞬時電流変換



累積電流出力は**クーロン カウンタA/D変換器制御/状態レジスタ(CADCSRA)**の**累積電流変換時間選択(CADAS1,0)**ビットによって選択した設定可能な変換時間での高精度、高分解能出力です。変換された値は1変換時間中に平均電流計測の流れを累積します。CC-ADCはその**割り込みが許可**されていれば、新規累積電流変換が完了してしまう毎に割り込みを生成します。変換からのデータは割り込みが与えられるのと同時に**累積電流(CADAC3~0)レジスタ**で更新されます。変換データの消失を避けるため、選択した変換時間内に全バイトが読まれなければなりません。下位バイトレジスタが読まれると、累積電流レジスタ更新と割り込みは最上位バイトが読まれるまで停止されます。**図18-3**は制限時間内にDATA3読み込みが完了されないためにDATA4が失われる累積電流変換例を示します。

図18-3. 累積電流変換



CC-ADCが変換中、CPUは**休止形態**へ移行して累積電流変換からの割り込みを待つことができます。クーロン カウンタに対して新しい累積電流値を追加後にCPUは再び休止形態に戻れます。これはCPUの作業負荷を減らし、電力消費を減らす低電力動作に多くの時間を費やすことを許します。CC-ADCは瞬時電流変換の結果が**設定可能な閾値**よりも大きい場合に割り込みを発生できます。これは定常電流状態の検出を可能にします。この機能は**パワーダウン動作**と**パワーオフ動作**を除く全休止形態と活動動作で利用できます。これはCC-ADCが設定可能な電流採取間隔の定常電流検出動作移行に設定し得る**パワーセーブ動作**での超低電力動作を許します。**クーロン カウンタA/D変換器制御/状態レジスタ(CADCSRA)**の**電流採取許可(CADSE)**ビットの設定(1)により、CADCSRAの**電流採取間隔選択(CADS1,0)**ビットによって指定された時間間隔でOFFに切り替えられる前に、クーロン カウンタは(1つの)瞬時電流変換を繰り返し行ないます。これは殆どの時間でクーロン カウンタOFFを保つ一方での定常電流検出動作を許します。

クーロン カウンタはパワーダウン動作で停止されます。この動作では電池の自己放電特性と時間測定が充放電の流れを見積もるのに使われるべきです。パワーダウン動作からの起動時、CC-ADCは継続する動作を自動的に再開します。

CC-ADCはCADCSRAの**クーロン カウンタA/D変換器許可(CADEN)**ビットの設定(1)によって許可されます。バントキャップ基準電圧が別に許可されなければならないことに注意してください。**81頁の「BGCCR - 基準電圧校正レジスタ」**をご覧ください。

CC-ADCはCADENが解除(0)されると電力を消費しません。従ってクーロン カウンタまたは定常電流検出機能が使われない時には必ずCC-ADCをOFFに切り替えることが推奨されます。CC-ADCはパワーダウンとパワーオフ動作で自動的に禁止されます。

CADENビットの設定(1)か、またはCADENが既に設定(1)でパワーダウン動作を抜けるかのどちらかによってCC-ADCが許可された後、最初の4変換は有用なデータを含まず、無視されるべきです。これはCADSEビットの解除(0)後にも適用します。

CC-ADCに対する代表的なシステム内変位電圧は0~100μVの範囲です。この変位誤差を補償するにはCC-ADCオフセット値がEEPROM内に格納され、クーロン計数に対して結果の値が加算されるのに先立って各累積電流変換値から減算されるべきです。このCC-ADC変位値は代表的な温度に於いてRSENSEを通して流れる電流0でのCC-ADC変換を実行することによって得られます。

電池が使われない、または電流値が長時間非常に低く留まるとき、クーロン計数に対してCC-ADCを使う代わりに充電の流れを概算することが推奨されます。この充電電流概算は電池システムの待機電流と電池の自己放電率を基にされるべきです。

18.3. クーロン カウンタA/D変換器用レジスタ

18.3.1. CADCSRA – CC-ADC 制御/状態レジスタA (CC-ADC Control and Status Register A)

ビット (\$E4)	7	6	5	4	3	2	1	0	
	CADEN	–	CADUB	CADAS1	CADAS0	CADSI1	CADSI0	CADSE	CADCSRA
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 – CADEN : CC-ADC 許可 (CC-ADC Enable)

CADENビットが解除(0)されると、クーロン カウンタA/D変換器(CC-ADC)が禁止され、実行中のどの変換も終了されます。CADENビットが設定(1)されると、CC-ADCは外部検出抵抗器(RSENSE)上の降下電圧を継続的に計測します。パワーオフ動作でCC-ADCは常に禁止されます。ハントキャップ基準電圧が別に許可されなければならないことに注意してください。81頁の「BGCCR – 基準電圧校正レジスタ」をご覧ください。

● ビット6 – Res : 予約 (Reserved)

このビットは予約されており、常に0として読めます。

● ビット5 – CADUB : CC-ADC 更新中フラグ (CC-ADC Update Busy)

CC-ADCはCPUとは別のクロック範囲で動作します。新しい値がCC-ADC制御/状態レジスタA(CADCSRA)、CC-ADC定常充電電流レジスタ(CADRCC)、またはCC-ADC定常放電電流レジスタ(CADRDC)へ書かれる毎にその値はCC-ADCクロック範囲に同期化されなければなりません。この同期化の間中、後続する書き込みは阻止されます。レジスタの1つの同期化は他の全ての更新を阻止します。これらの何れかのレジスタが同期化実行されている間中、このCADUBビットは1として読め、どのレジスタも同期化実行されていない時に0として読めます。

● ビット4,3 – CADAS1,0 : CC-ADC 累積電流変換時間選択 (CC-ADC Accumulate Current Select)

このCADASビットは表18-1.で示される累積電流出力に対する変換時間を選びます。

表18-1. CC-ADC累積電流変換時間

CADAS1,0	累積電流変換時間
0 0	125ms
0 1	250ms
1 0	500ms
1 1	1s

● ビット2,1 – CADSI1,0 : CC-ADC 定常電流検出間隔選択 (CC-ADC Current Sampling Interval)

このCADSIビットは表18-2.で示される定常電流検出に対する電流採取間隔を選びます。この電流採取間隔は定常電流検出許可(CADSE)ビットが設定(1)の場合だけ使われます。

表18-2. CC-ADC定常電流採取間隔

CADSI1,0	定常電流採取間隔
0 0	250ms (+採取時間)
0 1	500ms (+採取時間)
1 0	1s (+採取時間)
1 1	2s (+採取時間)

注: ・ 実際の値は18頁の「低速RC発振器」の実周波数に依存します。145頁の「電氣的特性」をご覧ください。
・ 採取時間 ~12ms。

● ビット0 – CADSE : CC-ADC 定常電流検出許可 (CC-ADC Current Sampling Enable)

CADSEビットが1を書かれると、実行中のCC-ADC変換は停止され、CC-ADCは定常電流検出動作へ移行します。

18.3.2. CADCSRB – CC-ADC 制御/状態レジスタB (CC-ADC Control and Status Register B)

ビット (\$E5)	7	6	5	4	3	2	1	0	
	–	CADACIE	CADRCIE	CADICIE	–	CADACIF	CADRCIF	CADICIF	CADCSRB
Read/Write	R	R/W	R/W	R/W	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7,3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読めます。

● ビット6 – CADACIE : CC-ADC 累積電流割り込み許可 (CC-ADC Accumulate Current Interrupt Enable)

CADACIE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)なら、CC-ADC累積電流割り込みが許可されます。

● **ビット5 – CADRCIE : CC-ADC 定常電流割り込み許可** (CC-ADC Regular Current Interrupt Enable)

CADRCIEビットが設定(1)され、SREGの全割り込み許可(I)ビットが設定(1)なら、CC-ADC定常電流割り込みが許可されます。

● **ビット4 – CADICIE : CC-ADC 瞬時電流割り込み許可** (CC-ADC Instantaneous Current Interrupt Enable)

CADICIE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)なら、CC-ADC瞬時電流割り込みが許可されます。

● **ビット2 – CADACIF : CC-ADC 累積電流割り込み要求フラグ** (CC-ADC Accumulate Current Interrupt Flag)

このCADACIFビットは累積電流変換が完了された後に設定(1)されます。累積電流割り込み許可(CADACIE)ビットとステータスレジスタの全割り込み許可(I)ビットが設定(1)なら、CC-ADC累積電流割り込みが実行されます。CADACIFは対応する割り込み処理ベクタを実行するとき、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CADACIFは解除(0)されます。

● **ビット1 – CADRCIF : CC-ADC 定常電流割り込み要求フラグ** (CC-ADC Regular Current Interrupt Flag)

このCADRCIFビットは最後のCC-ADC変換結果の絶対値が、CC-ADC定常充電電流レジスタ(CADRCC)とCC-ADC定常放電電流レジスタ(CADRDC)によって設定される比較値と等しいかまたは大きい時に設定(1)されます。正数は定常充電電流値と比較され、負数は定常放電電流値と比較されます。定常電流割り込み許可(CADRCIE)ビットとステータスレジスタの全割り込み許可(I)ビットが設定(1)なら、CC-ADC定常電流割り込みが実行されます。CADRCIFは対応する割り込み処理ベクタを実行するとき、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CADRCIFは解除(0)されます。

● **ビット0 – CADICIF : CC-ADC 瞬時電流割り込み要求フラグ** (CC-ADC Instantaneous Current Interrupt Flag)

このCADICIFビットは瞬時電流変換が完了された後に設定(1)されます。瞬時電流割り込み許可(CADICIE)ビットとステータスレジスタの全割り込み許可(I)ビットが設定(1)なら、CC-ADC瞬時電流割り込みが実行されます。CADICIFは対応する割り込み処理ベクタを実行するとき、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CADICIFは解除(0)されます。

18.3.3. CADICH, CADICL (CADIC) – CC-ADC 瞬時電流レジスタ (CC-ADC Instantaneous Current Register)

ビット (\$E9)	15	14	13	12	11	10	9	8	
	(CADIC15)								CADICH
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$E8)	7	6	5	4	3	2	1	0	
								(CADIC0)	CADICL
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

CC-ADC瞬時電流変換が完了すると、その結果がこれら2つのレジスタで得られます。CADIC15~0は16ビットへ符号拡張した2の補数形式で変換結果を表します。

CADICLが読まれると、CC-ADC瞬時電流レジスタはCADICHが読まれるまで更新されません。CADICL⇒CADICH手順でのレジスタ読み込みは矛盾のない値が読まれるのを保証します。

18.3.4. CADAC3~0 (CADAC) – CC-ADC 累積電流レジスタ (CC-ADC Accumulate Current Register)

ビット (\$E3)	31	30	29	28	27	26	25	24	
	(CADAC31)								CADAC3
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$E2)	23	22	21	20	19	18	17	16	
									CADAC2
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$E1)	15	14	13	12	11	10	9	8	
									CADAC1
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$E0)	7	6	5	4	3	2	1	0	
								(CADAC0)	CADAC0
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

CADAC3, CADAC2, CADAC1, CADAC0レジスタは32ビットへ符号拡張した2の補数形式で計測する累積電流を含みます。

CADAC0が読まれると、CC-ADC累積電流レジスタはCADAC3が読まれるまで更新されません。CADAC0⇒CADAC1⇒CADAC2⇒CADAC3手順でのレジスタ読み込みは矛盾のない値が読まれるのを保証します。

18.3.5. CADRCC – CC-ADC 定常充電電流レジスタ (CC-ADC Regular Chage Current Register)

ビット (\$E6)	15	14	13	12	11	10	9	8	
	(CADRCC7)							(CADRCC0)	CADRCC
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CC-ADC定常充電電流レジスタは定常充電電流検出についての閾値を決めます。CC-ADC瞬時電流変換の結果が、この定常充電電流値と等しいか、またはより大きい正数の時に**CC-ADC定常電流割り込みフラグ(CADRCIF)**が設定(1)されます。

このレジスタの値は2の補数形式で指定され、定常充電電流値の下位8ビットを定義します。定常充電電流値の上位側ビットは常に0です。定常充電電流に対する設定可能な範囲は表18-3.で与えられます。

このCC-ADC定常充電電流レジスタは**CC-ADC変換完了割り込み要求フラグ(CADICIF)**の設定(1)には影響を及ぼしません。

表18-3. 定常充電電流値用設定可能範囲

項目	最小	最大	1段階	単位
電圧	0	13700	53.7	μV
電流	RSENSE=5mΩ	0	2740	10.7
	RSENSE=7mΩ	0	1957	7.7

18.3.6. CADRDC – CC-ADC 定常放電電流レジスタ (CC-ADC Regular Discharge Current Register)

ビット (\$E7)	15	14	13	12	11	10	9	8	
	(CADRDC7)							(CADRDC0)	CADRDC
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CC-ADC定常放電電流レジスタは定常放電電流検出についての閾値を決めます。CC-ADC瞬時電流変換の結果が、この定常放電電流値と等しいか、またはより大きい負数の時に**CC-ADC定常電流割り込みフラグ(CADRCIF)**が設定(1)されます。

このレジスタの値は2の補数形式で指定され、定常放電電流値の下位8ビットを定義します。定常放電電流値の上位側ビットは常に0です。定常放電電流に対する設定可能な範囲は表18-4.で与えられます。

このCC-ADC定常放電電流レジスタは**CC-ADC変換完了割り込み要求フラグ(CADICIF)**の設定(1)には影響を及ぼしません。

表18-4. 定常放電電流値用設定可能範囲

項目	最小	最大	1段階	単位
電圧	0	13700	53.7	μV
電流	RSENSE=5mΩ	0	2740	10.7
	RSENSE=7mΩ	0	1957	7.7

19. 電圧調整器

19.1. 特徴

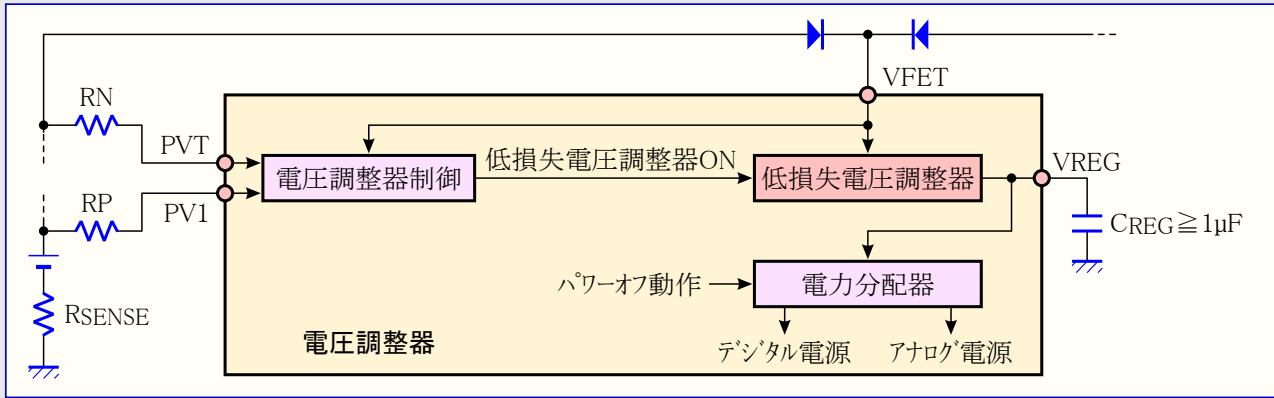
- ・ 直線的調節
- ・ 動作電圧範囲4.0～25V
- ・ 3.3Vの固定出力電圧

ATmega406はVFET端子によって電力供給されます。VFET端子の動作電圧範囲は4.0～25Vです。内蔵電圧調整器はこの電圧を内部の論理回路、入出力線とアナログ回路用の供給電圧である3.3Vに下げて調節します。

電圧調整器の安定動作用に1 μ Fまたはより大きなデカップ用外部コンデンサが必要とされます。より大きなコンデンサはより大きな負荷電流と起動時間の増加を許します。

電圧調整器の構成図は図19-1.で示されます。

図19-1. 電圧調整器構成図



19.2. 動作

この電圧調整器はパワーオフ動作を含む全休止形態で動作します。この動作での電圧調整器は電圧調整器自体と充電検出部だけに動作を許して全周辺部に対する供給をOFFすることによってATmega406の消費電力を自動的に減らします。

電圧高正規はVFET端子からの調節開始を許す前に安定動作状態になることを自動的に保証します。VFETピンの電圧が電圧調整器ON閾値電圧(VROT)以下なら、低損失電圧調整器はOFFにされます。

電圧調整器の電源投入は電池パックにスマート バッテリ制御器が組み込まれていて、充電器が存在しない場合の電池側、または過放電が起こってしまった(0V充電)場合の充電器側のどちらかで行われます。

充電器有りで電源投入する時にVFETとPVTピン間の電圧は充電閾値電圧(VCHT)以上でなければなりません。

充電器無しで電源投入する時に電池セル1の電圧(VPV1)は電池セル1閾値電圧(VPV1T)以上でなければなりません。

電圧調整器電源投入後、チップはパワーオフ休止動作(最低消費電力)に移行します。充電器が検出されるまでチップはこの動作に留まります。充電器検出の詳細については27頁の「[電源ON/セットと充電器接続](#)」をご覧ください。

149頁の表30-4.は低損失電圧調整器電源投入に対する特性を示します。

20. 電圧A/D変換器 (10チャンネル汎用12ビット $\Delta\Sigma$ A/D変換器)

20.1. 特徴

- ・ 12ビット分解能
- ・ ± 1 LSB精度
- ・ 519 μ s変換時間
- ・ 電池セル電圧測定用の4つの差動入力チャンネル
- ・ 6チャンネルのシングル エンド入力チャンネル
- ・ $0 \sim 0.9 \times V_{REF}$ の入力電圧範囲
- ・ VREGと電池セル電圧の $\times 0.2$ 前置分圧
- ・ 電圧A/D変換完了割り込み

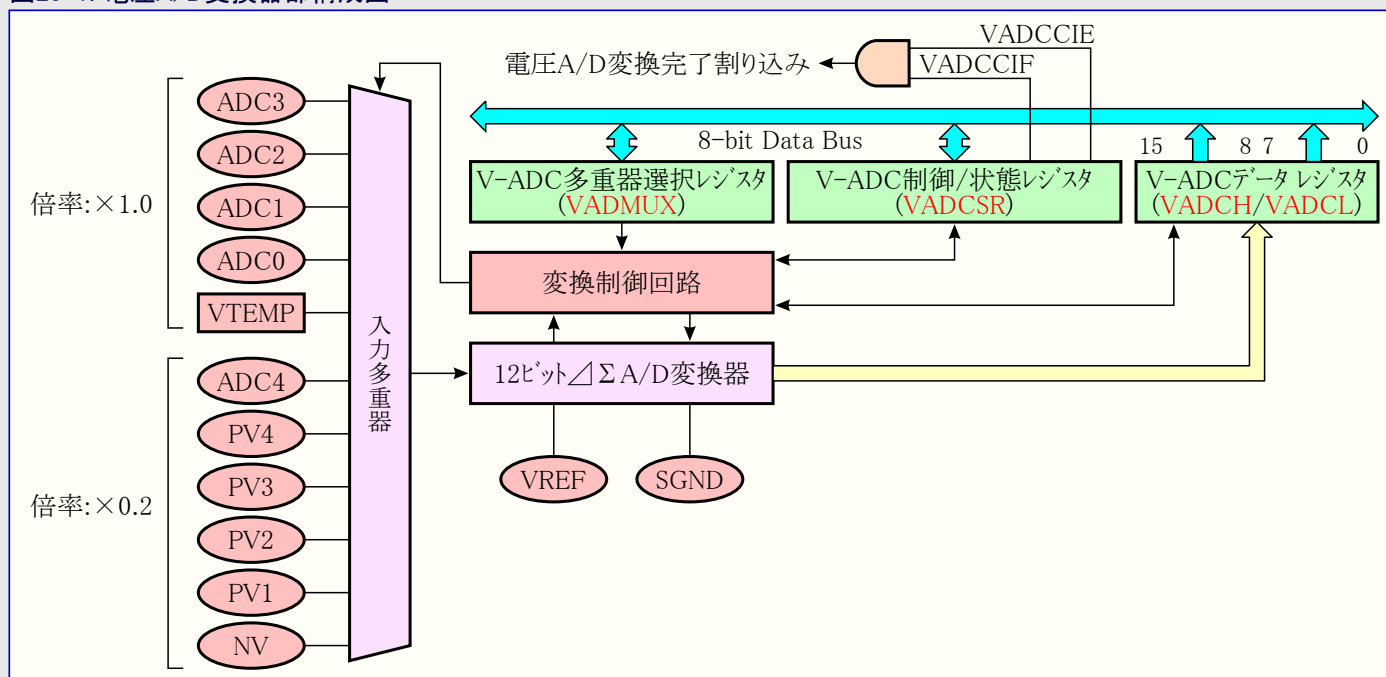
ATmega406は12ビット $\Delta\Sigma$ A/D変換器が特徴です。自動変位相殺技術が入力変位電圧を0.5mV未満に減少します。

この電圧A/D変換器(V-ADC)は入力多重器を通して異なる10の信号源に接続されます。電池セル電圧測定用に4つの差動チャンネルがあります。これらのチャンネルはV-ADCの全計測範囲に対応するため、0.2倍に縮小されます。加えてSGNDを基準とした6つのシングル エンドチャンネルがあります。ポートAのADC入力ピン測定用の5チャンネルと、1つのチャンネルは内部温度感知器(VPTAT)測定用です。ADC3 $\sim$ 0は縮尺されず、1.1Vに応じた全計測範囲表現を意味します。ADC4は0.2倍に縮小され、5.5Vに応じた全計測範囲表現を意味します。PA4ピンが外部サーミスタ補充配備に使われる時に、このADC4入力はこのピンの電圧測定に使えます。143頁の図29-1をご覧ください。

電池セル電圧測定に関して $\pm 0.25\%$ よりも良い総絶対精度を得るため、アナログ前置回路に個別セル電圧利得用校正レジスタが提供されます。工場校正値は識票列に格納されています(123頁の「ソフトウェアからの識票列読み出し」をご覧ください)。セル電圧のV-ADC変換はアナログ前置回路での利得誤差を正すため、ソフトウェアによって対応する校正値で尺度補正されなければなりません。

24頁の「PRR0 - 電力削減レジスタ0」のPRVADCB $\times$ ビットは電圧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。

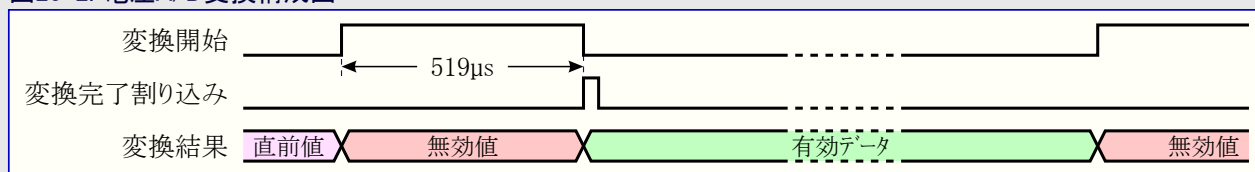
図20-1. 電圧A/D変換器部構成図



20.2. 操作

電圧A/D変換器(V-ADC)の変換を許可するにはV-ADC制御/状態レジスタ(VADCSR)のV-ADC許可(VADEN)ビットが設定(1)されなければなりません。このビットが解除(0)されると、V-ADCはOFFにされ、実行中のどの変換も終了されます。V-ADCはパワーセーブ動作、パワーダウン動作、パワーオフ動作で自動的に停止されます。バンドギャップ基準電圧が別に許可/禁止されなければならないことに注意してください。81頁の「BGCCR – 基準電圧校正レジスタ」をご覧ください。

図20-2. 電圧A/D変換構成図



V-ADC変換を実行するにはV-ADC多重器選択レジスタ(VADMUX)のVADMUX3~0ビットへの書き込みによって、アナログ入力チャンネルが先に選択されなければなりません。VADCSRのV-ADC変換開始(VADSC)ビットへ論理1が書かれると、選択したチャンネルの変換が始まります。VADSC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き、ハードウェアによって解除(0)されます。変換進行中の間に違うデータチャンネルが選択されると、そのチャンネル変更を実行するのに先立ってV-ADCは現在の変換を(正常に)完了します。変換完了時、VADCSRのV-ADC変換完了割り込み要求フラグ(VADCCIF)が設定(1)されます。1回の12ビット変換は開始(VADSC)ビットが設定(1)されてから割り込みフラグ(VADCCIF)が設定(1)されるまでの完了に519µsかかります。正しいデータが読まれるのを保証するため、上位と下位の両データレジスタ(VADCH/VADCL)は新しい変換を開始する前に読まれるべきです。

20.3. 電圧A/D変換器用レジスタ

20.3.1. VADMUX – V-ADC 多重器選択レジスタ (V-ADC Multiplexer Selection Register)

ビット (\$7C)	7	6	5	4	3	2	1	0	
	–	–	–	–	VADMUX3	VADMUX2	VADMUX1	VADMUX0	VADMUX
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として読めます。
- ビット3～0 – VADMUX3～0 : V-ADC チャンネル選択ビット (V-ADC Channel Selection Bits)
このVADMUX3～0ビットはV-ADCチャンネル選択を決定します。表20-1をご覧ください。

表20-1. VADMUXチャンネル選択							
VADMUX3～0	選択チャンネル	倍率	備考	VADMUX3～0	選択チャンネル	倍率	備考
0 0 0 0				1 0 0 0	ADC1	×1.0	
0 0 0 1	PV1-NV	×0.2	電池セル1	1 0 0 1	ADC2	×1.0	
0 0 1 0	PV2-PV1	×0.2	電池セル2	1 0 1 0	ADC3	×1.0	
0 0 1 1	PV3-PV2	×0.2	電池セル3	1 0 1 1			
0 1 0 0	PV4-PV3	×0.2	電池セル4	1 1 0 0			
0 1 0 1	ADC4	×0.2		1 1 0 1			
0 1 1 0	VTEMP	×1.0	温度感知器	1 1 1 0			
0 1 1 1	ADC0	×1.0		1 1 1 1			

20.3.2. VADCSR – V-ADC 制御/状態レジスタ (V-ADC Control and Status Register)

ビット (\$7A)	7	6	5	4	3	2	1	0	
	–	–	–	–	VADEN	VADSC	VADCCIF	VADCCIE	VADCSR
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として読めます。
- ビット3 – VADEN : V-ADC 許可 (V-ADC Enable)
このビットへの1書き込みが電圧A/D変換器(V-ADC)を許可します。0書き込みによってV-ADCはOFFに切り替えられます。変換進行中のV-ADCのOFFへの切り替えはその変換を(強制)終了します。バンドギャップ基準電圧が別に許可されなければならないことに注意してください。81頁の「BGCCR – 基準電圧校正レジスタ」をご覧ください。
- ビット2 – VADSC : V-ADC 変換開始 (V-ADC Start Conversion)
このビットへの1書き込みが選択したチャンネルでの新規変換を開始します。
VADSCは変換が完了しない限り1として読めます。変換が完了すると、それは0に戻ります。このビットへの0書き込みは無効です。V-ADC許可(VADEN)ビットが0を書かれると、VADSCは自動的に解除(0)されます。
- ビット1 – VADCCIF : V-ADC 変換完了割り込み要求フラグ (V-ADC Conversion Complete Interrupt Flag)
V-ADC変換が完了してデータレジスタ(VADCH/VADCL)が更新されると、このビットが設定(1)されます。(その時に)V-ADC変換完了割り込み許可(VADCCIE)ビットとステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、V-ADC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VADCCIF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の論理1書き込みによってもVADCCIFは解除(0)されます。VADCSR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。
- ビット0 – VADCCIE : VADC 変換完了割り込み許可 (V-ADC Conversion Complete Interrupt Enable)
このビットが1を書かれ、且つSREGの全割り込み許可(I)ビットが設定(1)されていると、V-ADC変換完了割り込みが活性化(許可)されます。

20.3.3. VADCH,VADCL (VADCD) – V-ADC データレジスタ (V-ADC Data Register)

ビット (\$79)	15	14	13	12	11	10	9	8	
	–	–	–	–	VADC11	VADC10	VADC9	VADC8	VADCH
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	
	VADC7	VADC6	VADC5	VADC4	VADC3	VADC2	VADC1	VADC0	VADCL
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

V-ADC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。正しいデータが読まれるのを保証するため、データレジスタは新規変換が開始されるのに先立って読まれなければなりません。

● VADC11~0 : V-ADC 変換結果 (V-ADC Conversion Result)

これらのビットは変換での結果を表します。

電池セル電圧測定での最大精度を得るために利得と変位(オフセット)補償が必要とされます。工場校正値がデバイスの識票列内に格納されています。詳細については123頁の「ソフトウェアからの識票列読み出し」項をご覧ください。セル電圧は次式によってmVで与えられます。

$$\text{セルn電圧(mV)} = \frac{\text{セルnA/D変換結果} \times \text{利得校正値}}{\text{TBD}} - \text{セルn変位校正値}$$

Vtemp変換実行時、この結果は識票列に格納された工場校正値によって補正されなければなりません。詳細については123頁の「ソフトウェアからの識票列読み出し」項をご覧ください。絶対温度は次式によってケルビン(K)で与えられます。

$$\text{絶対温度:T(K)} = \frac{\text{VtempのA/D変換結果} \times \text{PTAT校正値}}{\text{TBD}}$$

20.3.4. DIDR0 – デジタル入力禁止レジスタ0 (Digital Input Disable Register 0)

ビット (\$7E)	7	6	5	4	3	2	1	0	
	–	–	–	–	VADC3D	VADC2D	VADC1D	VADC0D	DIDR0
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7,4 – Res : 予約 (Reserved Bits)

これら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、DIDR0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3~0 – VADC3D~VADC0D : V-ADC ADC3~0ピン デジタル入力禁止 (V-ADC3~0 Digital Input Disable)

このビットが論理1を書かれると、対応するV-ADC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。

21. 基準電圧と温度感知器

21.1. 特徴

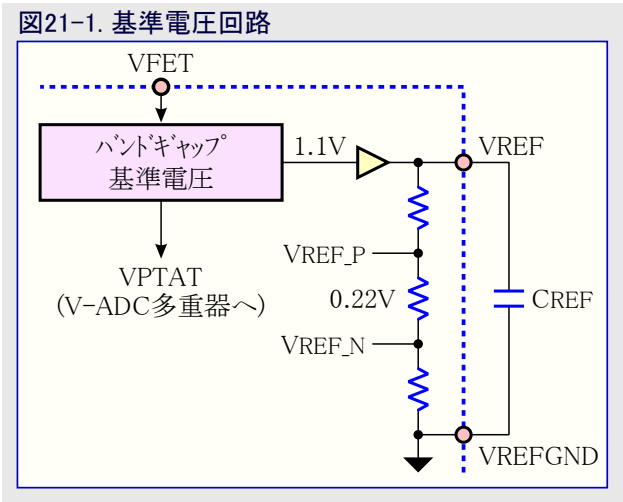
- ・ 1.100Vの精密基準電圧
- ・ 校正後±0.1%精度 (校正段階2mV)
- ・ 校正後温度変動80ppm/°C未満
- ・ 電圧調整器用代替低電力基準電圧
- ・ 温度感知器内蔵
- ・ 基準電圧と内蔵発振器の両方で実行時温度変動補償可能
- ・ 最適雑音性能用外部デカップ(端子)
- ・ 低消費電力

低電力バンドギャップ基準電圧は1.100Vの正確なチップ上の基準電圧(VREF)をATmega406に供給します。この基準電圧はチップ上の電圧調整器、電圧A/D変換器(V-ADC)、クーロンカウンタA/D変換器(CC-ADC)用の基準として使われます。A/D変換器への基準電圧は最小消費電力での優秀な雑音性能を可能とするために緩衝器と外部デカップコンデンサを使います。CC-ADCへの基準電圧(VREF_P/VREF_N)は電流検出力ピンで必要とする全尺度に合わせるために尺度補正されます。この形態はV-ADCとCC-ADC両方の同時動作も可能にします。

ATmega406は工場校正後に温度超低変動を保証するための2段階の校正法が特徴です。最初の段階は85°Cで、そして2つ目の段階は室温で実行されます。既定によってAtmelの工場校正は85°Cで実行され、その結果はフラッシュメモリ内に格納されます。利用者はそれらの試験の手順で2つ目の校正段階を容易に実装できます。これは正確な入力電圧と安定な室温を必要とします。この校正後の温度変動は設計と特性付けによって0~60°Cでの80ppm/°Cと0~85°Cでの100ppm/°C未満が保証されます。基準電圧校正レジスタ(BGCCR)はソフトウェアで温度補償を実装する実行時切り替えもできます。従って温度範囲内のどの温度にも対する超高精度は追加校正手順の費用で達成できます。

低電力低精度の基準電圧源が存在します。この基準電圧源はバンドギャップ基準電圧が禁止される時に必ず電圧調整器用の基準として選択されます。この基準電圧源はV-ADCとCC-ADCに対して利用できません。

ATmega406にはタイ温度監視用の内蔵温度感知器があります。絶対温度比例電圧(VPTAT : Voltage Proportional-To-Absolute-Temperature)は基準電圧回路で生成され、V-ADC入力の多重器に接続されます。この温度感知器は基準電圧と内蔵発振器の両方での実行時温度変動補償に使えます。ケルビン温度での絶対温度を得るために測定したVPTAT電圧は識票列内に格納されたVPTAT工場校正值で尺度補正されなければなりません。詳細については123頁の「ソフトウェアからの識票列読み出し」をご覧ください。



21.2. 基準電圧校正レジスタへの書き込み

校正レジスタが変更されると、電圧調整器出力と低電圧検出器(BOD)の両方に影響を及ぼします。BODは新しい検出値に素早く対応し、一方電圧調整器は外部デカップ容量(コンデンサ)に依存してもっとゆっくり電圧を調整します。校正が行われる時にBODリセットが起きるのを避けるため、バンドギャップ基準電圧校正值(BGCC5~0)と基準電圧温度校正值(BGCR7~0)ビット値を各段階間の保持時間と共に1段階ずつ変更することが推奨されます。

この保持時間は電圧調整器の外部デカップ容量(コンデンサ)に依存します。詳細については表21-1をご覧ください。

表21-1. CREGに依存する保持時間

電圧調整器容量(CREG)	BGCCR用保持時間	BGCR7用保持時間
1μF	1.2μs	3.0μs
2μF	2.4μs	6.0μs
3μF	3.6μs	9.0μs
4μF	4.8μs	12.0μs
5μF	6.0μs	15.0μs
6μF	7.2μs	18.0μs
7μF	8.4μs	21.0μs
8μF	9.6μs	24.0μs
9μF	10.8μs	27.0μs
10μF	12.0μs	30.0μs

21.3. 基準電圧と温度感知器用レジスタ

21.3.1. BGCCR – 基準電圧校正レジスタ (Bandgap Calibration C Register)

ビット (\$D0)	7	6	5	4	3	2	1	0	
	BGEN	–	BGCC5	BGCC4	BGCC3	BGCC2	BGCC1	BGCC0	BGCCR
Read/Write	R (R/W)	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	工場校正値						

● ビット7 – BGEN : バンドギャップ基準電圧許可 (Bandgap Enable)

本ビットはATmega406の改訂Eから利用できません。完全な記述は改訂Aの本資料で得られます。

(訳補) 以下が改訂Aのデータシート該当内容です。レジスタ構成図の()内も同様。

BGENビットの1設定がバンドギャップ基準電圧を許可します。このビットは電圧A/D変換器(V-ADC)またはクーロンカウンタA/D変換器(CC-ADC)が許可される前に設定(1)されなければならず、どちらかのA/D変換器が許可されている間中、設定(1)に留まらなければなりません。

バンドギャップ基準電圧はこのBGENビットが設定(1)されると、電圧調整器用の基準電圧として使われます。低精度基準電圧はBGENビットが解除(0)されている時に使われます。

● ビット6 – Res : 予約 (Reserved Bit)

このビットは将来の使用に対して予約されています。

● ビット5~0 – BGCC5~0 : バンドギャップ基準電圧校正値 (BG Calibration of PTAT Current)

これらのビットはバンドギャップ基準電圧の公称値調節に使われます。これらの値は2進値です。最小VREFは000000、最大VREFは111111です。1段階は概ね2mVです。

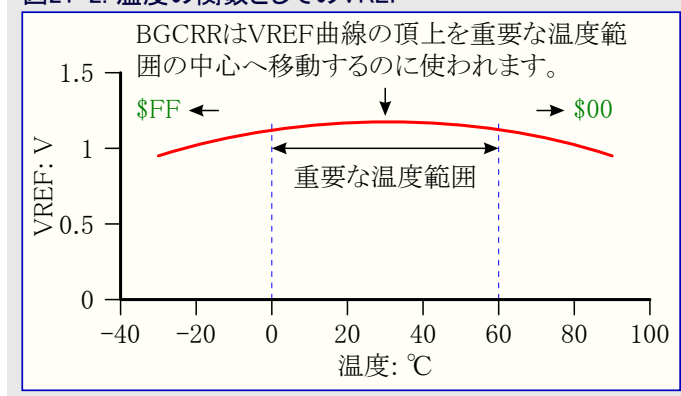
21.3.2. BGCR – 基準電圧温度係数校正レジスタ (Bandgap Calibration R Register)

ビット (\$D1)	7	6	5	4	3	2	1	0	
	BGCR7	BGCR6	BGCR5	BGCR4	BGCR3	BGCR2	BGCR1	BGCR0	BGCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	工場校正値								

● ビット7~0 – BGCR7~0 : 基準電圧温度係数校正値 (BG Calibration of Resistor ladder)

これらのビットはバンドギャップ基準電圧の温度係数補正に使われます。図21-2は温度の関数としてのVREFを図解します。VREFは低温で正の温度係数、高温で負の温度係数を持ちます。製法変化によって、VREF曲線の頂上はより高いまたはより低い温度に位置されるかもしれません。重要な温度範囲での温度変動を最小とするため、BGCRは重要な温度範囲の中心に対して曲線の頂上を補正するのに使われます。BGCRビットは9つの可能な設定(\$00,\$01,\$03,\$07,\$0F,\$1F,\$3F,\$7F,\$FF)となる符号化された温度です。値\$00はVREF曲線の頂上を最高可能温度へ移し、値\$FFはVREF曲線の頂上を最低可能温度へ移します。

図21-2. 温度の関数としてのVREF



22. 電池保護

22.1. 特徴

- ・ 最低電圧保護
- ・ 充電過電流保護
- ・ 放電過電流保護
- ・ 短絡保護
- ・ 設定と施錠が可能な検出値と再動作時間
- ・ CPUから独立した自律動作

VFETピンの電圧が設定可能な最低電圧検出値以下に下がると、OC(C-FET)、OPC(PC-FET)とOD(D-FET)は禁止され、チップは電力消費を最小化に減らすため、**パワーオフ動作**に設定されます。

電流/電池保護回路(CBP)は充放電電流を監視して、過電流または短絡状態が検出されると、C-FET、PC-FETとD-FETを禁止します。これらには設定可能な3つの異なる検出値、**放電過電流検出値**、**充電過電流検出値**、**短絡検出値**があります。PIとNI入力ピンの外部濾波器は短絡検出に対して大きすぎる遅延を引き起こします。従って独立したPPIとNNI入力ピンが電流/電池保護に使われます。これらには電流/電池保護稼動(状態)にするための設定可能な2つの異なる遅延、**短絡再動作時間**と**過電流再動作時間**があります。電流/電池保護が稼動状態にされてしまった後、応用ソフトウェアはFETを再度許可しなければなりません。電池保護ハードウェアはソフトウェアが放電FETを再度許可し得る前に1秒の防止時間を提供します。これは応用ソフトウェアがあまりに早く意図せずに放電FETを再度許可する場合の安全を提供します。

保護の発動はCPUへの割り込みも起こします。この**電池保護割り込み**はCPUによって個別に許可と禁止ができます。

様々な電池保護種別の効果は表22-1.で与えられます。

表22-1. 電池保護種別の効果 (訳注:過電流/短絡状態脱出は原書で本表以外にその記述がありません。)						
電池保護種別	割り込み要求	C-FET	D-FET	PC-FET	電池セル平衡FET	MCU動作
最低電圧検出	警告と脱出でのCPUリセット	禁止	禁止	禁止	禁止	パワーオフ動作
放電過電流保護	(保護状態)突入と脱出	禁止	禁止	禁止	動作継続	動作継続
充電過電流保護	(保護状態)突入と脱出	禁止	禁止	禁止	動作継続	動作継続
短絡保護	(保護状態)突入と脱出	禁止	禁止	禁止	動作継続	動作継続

消費電力を減少するために放電過電流保護と短絡保護はD-FETが禁止されるとき、自動的に非活動(禁止)にされます。充電過電流保護はC-FETとPC-FETの両方が禁止される時に禁止されます。けれどもC-FETかPC-FETのどちらかがPWMによって制御されていると、決して自動的に禁止されないことに注意してください。

22.2. 最低電圧保護

最低電圧保護は電池セルが設定可能な最低電圧検出値よりも深く放電されないことを保証します。VFETピンの電圧が設定可能な遅延時間よりも長い時間、この最低電圧検出値以下なら、C-FET、PC-FETとD-FETは自動的にOFFされ、チップはパワーオフ動作へ移行します。**最低電圧早期警告割り込み要求フラグ(DUVIF)**はパワーオフ移行前250msで設定(1)されます。これは電力がOFFされる前に必要な動作を行なう機会をCPUに与えます。

デバイスは充電器が接続されるまでパワーオフ動作に留まります。充電器が検出されると、通常の電源投入手順が開始され、チップは既定状態に初期化します。

最低電圧遅延時間と最低電圧検出値は**電池保護最低電圧レジスタ(BPDUV)**で設定できます。規定値レジスタ群は初期設定後に施錠でき、次のハードウェアリセットまで更なるどんな更新も禁止します。

レジスタ記述については84頁の「**電池保護用レジスタ**」を参照してください。

22.3. 放電過電流保護

電流/電池保護(CBP)はPPIとNNI入力ピンの分路抵抗電圧を採取することでセル電流を監視します。差動演算増幅器は適切な利得でこの電圧を増幅します。演算増幅器からの出力はアナログ比較器によって設定可能な内蔵基準電圧と正確に比較されます。分路抵抗電圧が**過電流保護再動作時間**よりも長い時間、**放電過電流検出値**以上なら、チップは放電過電流保護を稼動(状態)にします。**内蔵超低電力RC発振器(ULP)**によってクロック駆動された採取システムが過電流と短絡保護に使われます。これは確かなクロック元と変位(オフセット)除去と低消費電力を保証します。

放電過電流保護が稼動(状態)にされると、外部C-FET、PC-FETとD-FETは禁止され、電流保護計時器が起動されます。この計時器はFETが最低1秒間禁止されるのを保証します。その後、応用ソフトウェアは通常動作を再度許可するのに**FET制御/状態レジスタ(FCSR)**の**放電FET許可(DFE)ビット**と**充電FET許可(CFE)ビット**を設定(1)しなければなりません。電池の負荷が未だ大きすぎる間にD-FETが再度許可されると、放電過電流保護が再び稼動(状態)にされるでしょう。

22.4. 充電過電流保護

PPIとNNIPの電圧が過電流保護再動作時間よりも長い時間、**充電過電流検出値**以上なら、チップは充電過電流保護を稼動(状態)にします。

充電過電流保護が稼動(状態)にされると、外部C-FET、PC-FETとD-FETは禁止され、電流保護計時器が起動されます。この計時器はFETが最低1秒間禁止されるのを保証します。その後、応用ソフトウェアは通常動作を再度許可するのにFET制御/状態レジスタ(FCSR)の放電FET許可(DFE)ビットと充電FET許可(CFE)ビットを設定(1)しなければなりません。C-FETが再度許可され、且つ充電器が高すぎる電流供給を継続すると、充電過電流保護が再び稼動(状態)にされるでしょう。

22.5. 短絡保護

高電流検出の第2段階は非常に大きな放電電流へのより速い応答時間を可能とするために提供されます。**短絡再動作時間**よりも長い時間、**短絡検出値**よりも大きな放電電流が存在すると、短絡保護が稼動(状態)にされます。

短絡保護が稼動(状態)にされると、外部C-FET、PC-FETとD-FETは禁止され、電流保護計時器が起動されます。この計時器はFETが最低1秒間禁止されるのを保証します。その後、応用ソフトウェアは通常動作を再度許可するのにFET制御/状態レジスタ(FCSR)の放電FET許可(DFE)ビットと充電FET許可(CFE)ビットを設定(1)しなければなりません。短絡を引き起こした条件が取り除かれる前にD-FETが再許可されると、短絡保護が再び稼動(状態)にされるでしょう。

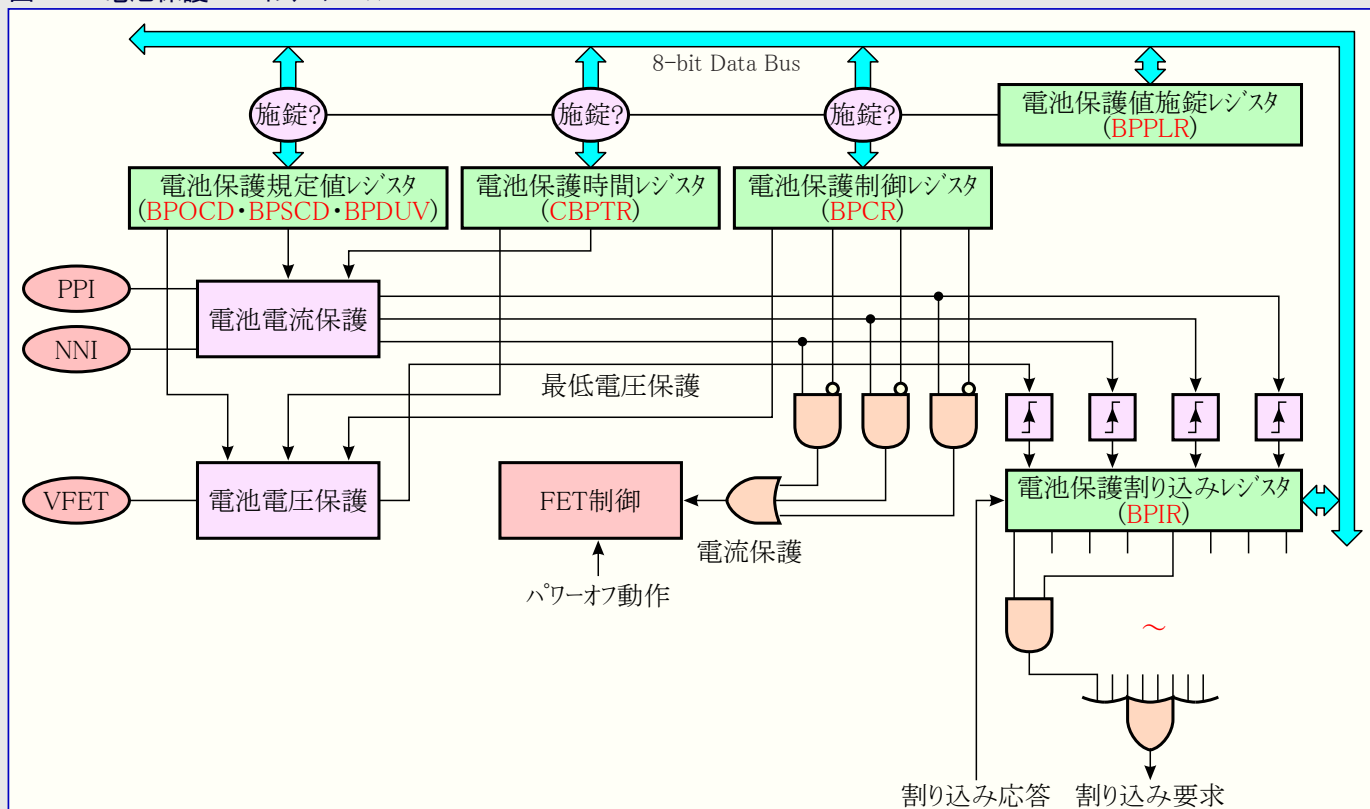
過電流と短絡保護規定値は各種電池形式に適合するように設定可能です。この各規定値はI/Oレジスタへの書き込みによって設定します。規定値レジスタ群は初期設定後に施錠でき、次のハードウェアリセットまで更新どんな更新も禁止します。

レジスタ記述については84頁の「**電池保護用レジスタ**」を参照してください。

22.6. 電池保護CPUインターフェース

電池保護CPUインターフェースは図22-1.で描かれます。

図22-1. 電池保護CPUインターフェース



各保護には**割り込み要求フラグ**があります。各フラグはCPUによって読み、そして解除(0)でき、各フラグには個別の割り込み許可があります。許可した全フラグはCPUへの1つの電池保護割り込み要求に結合されます。この(結合した)割り込みは**パワーオフ動作**を除くどの**動作種別**からもCPUを起動できます。割り込み要求フラグはCPUからのそれらのビット位置への論理1書き込みによって解除(0)されます。フラグまたは状態ビットのどれも、チップがパワーオフ動作に移行されてしまったことを示さないことに注意してください。これはこの動作種別でCPUが電力停止にされるためです。けれども動作を再開する時に**CPUのリセットフラグ**を監視することによって、CPUはパワーオフ状態から来たことを検知することができます。

22.7. 電池保護用レジスタ

電池保護部はCPUと異なるクロック範囲で動作します。

電池保護部はCPUと別のクロック範囲で動作します。新規値が電池保護制御レジスタ(BPCR)、電流/電池保護時間レジスタ(CBPTR)、電池過電流保護検出値レジスタ(BPOCD)、電池短絡保護検出値レジスタ(BPSCD)、または電池最低電圧レジスタ(BPDUV)へ書かれる毎にその値は電池保護クロック範囲に同期化されなければなりません。この同期化の間、後続するこれらのレジスタへの書き込みは行なわれません。従ってこれらのレジスタの1つへの書き込み後、同じレジスタは次の8CPUクロック期間内に再書き込みされるべきではありません。各レジスタは他のレジスタと独立して同期化されることに注意してください。

22.7.1. BPPLR – 電池保護値施錠レジスタ (Battery Protection Parameter Lock Register)

ビット (\$F8)	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	BPPLR	BPPL	BPPLR
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7～2 – Res : 予約 (Reserved Bits)

これらのビットは将来の使用に対して予約されており、常に0として読まれます。

●ビット1 – BPPLR : 電池保護規定値施錠許可 (Battery Protection Parameter Lock Enable)

●ビット0 – BPPL : 電池保護規定値施錠 (Battery Protection Parameter Lock)

電池保護規定値レジスタ群内で設定する電池保護規定値群と電池保護制御レジスタ(BPCR)内で設定する禁止機能は、更なるどんなソフトウェア更新からも施錠できます。一旦施錠されると、これらのレジスタは次のハードウェアリセットまでアクセスできません。これはソフトウェア暴走による意図せぬ変更からこれらのレジスタを保護する安全な方法を提供します。リセット直後にソフトウェアがこれらのレジスタを設定し、その後の更なるどんな更新からも保護することが推奨されます。

これらのレジスタを施錠するには以下の手順に従わなければなりません。

1. 同一操作で電池保護規定値施錠許可(BPPLR)と電池保護規定値施錠(BPPL)に論理1を書いてください。
2. 次からの4クロック周期以内に同一操作でBPPLRに論理0、BPPLに論理1を書いてください。

電池保護規定値レジスタ群は、電池保護制御レジスタ(BPCR)、電流/電池保護時間レジスタ(CBPTR)、電池過電流保護検出値レジスタ(BPOCD)、電池短絡保護検出値レジスタ(BPSCD)、電池最低電圧レジスタ(BPDUV)です。

22.7.2. BPCR – 電池保護制御レジスタ (Battery Protection Control Register)

ビット (\$F7)	7	6	5	4	3	2	1	0	
	–	–	–	–	DUVD	SCD	DCD	CCD	BPCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7～4 – Res : 予約 (Reserved Bits)

これらのビットは将来の使用に対して予約されており、常に0として読まれます。

●ビット3 – DUVD : 最低電圧保護禁止 (Deep Under-voltage Protection Disable)

このDUVDビットが設定(1)されると、最低電圧保護が禁止されます。最低電圧検出が禁止され、どの最低電圧状態も無視されます。

●ビット2 – SCD : 短絡保護禁止 (Short Circuit Protection Disable)

このSCDビットが設定(1)されると、短絡保護が禁止されます。短絡検出が禁止され、どんな短絡状態も無視されます。

●ビット1 – DCD : 放電過電流保護禁止 (Discharge Over-current Protection Disable)

このDCDビットが設定(1)されると、放電過電流保護が禁止されます。放電過電流検出が禁止され、どんな放電過電流状態も無視されます。

●ビット0 – CCD : 充電過電流保護禁止 (Charge Over-current Protection Disable)

このCCDビットが設定(1)されると、充電過電流保護が禁止されます。充電過電流検出が禁止され、どんな充電過電流状態も無視されます。

22.7.3. CBPTR – 電流/電池保護時間レジスタ (Current Battery Protection Timing Register)

ビット (\$F6)	7	6	5	4	3	2	1	0	
	SCPT3	SCPT2	SCPT1	SCPT0	OCPT3	OCPT2	OCPT1	OCPT0	CBPTR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7～4 – SCPT3～0 : 短絡保護確認時間 (Short-circuit Protection Timing)

これらのビットは短絡保護の(確認用)遅延を制御します。表22-2.をご覧ください。

表22-2. SCPT3～0に対応する短絡遅延時間 (短絡保護再動作時間)

SCPT3～0	Typ	SCPT3～0	Typ	SCPT3～0	Typ	SCPT3～0	Typ
0 0 0 0	61μs	0 1 0 0	305μs	1 0 0 0	610μs	1 1 0 0	1098μs
0 0 0 1	122μs	0 1 0 1	366μs	1 0 0 1	732μs	1 1 0 1	1220μs
0 0 1 0	183μs	0 1 1 0	427μs	1 0 1 0	854μs	1 1 1 0	1342μs
0 0 1 1	244μs	0 1 1 1	488μs	1 0 1 1	976μs	1 1 1 1	1464μs

●ビット3～0 – OCPT3～0 : 過電流保護確認時間 (Over-current Protection Timing)

これらのビットは充放電電流保護の(確認用)遅延を制御します。表22-3.をご覧ください。同じ設定が過電流保護の両形式に適用されることに注意してください。

表22-3. OCPT3～0に対応する過電流遅延時間 (過電流保護再動作時間)

OCPT3～0	Typ	OCPT3～0	Typ	OCPT3～0	Typ	OCPT3～0	Typ
0 0 0 0	1ms	0 1 0 0	8ms	1 0 0 0	16ms	1 1 0 0	24ms
0 0 0 1	2ms	0 1 0 1	10ms	1 0 0 1	18ms	1 1 0 1	26ms
0 0 1 0	4ms	0 1 1 0	12ms	1 0 1 0	20ms	1 1 1 0	28ms
0 0 1 1	6ms	0 1 1 1	14ms	1 0 1 1	22ms	1 1 1 1	30ms

22.7.4. BPOCD – 電池保護過電流検出値レジスタ (Battery Protection Over-current Level Register)

ビット (\$F5)	7	6	5	4	3	2	1	0	
	OCDL3	OCDL2	OCDL1	OCDL0	CCDL3	CCDL2	CCDL1	CCDL0	BPOCD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

●ビット7～4 – OCDL3～0 : 放電過電流検出値 (Discharge Over-current Detection Level)

これらのビットは表22-4.で定義される、放電過電流検出用のRSENSE電圧値を設定します。

表22-4. DCDL3～0に対応する放電過電流検出用RSENSE電圧

DCDL3～0	Typ	DCDL3～0	Typ	DCDL3～0	Typ	DCDL3～0	Typ
0 0 0 0	0.050V	0 1 0 0	0.070V	1 0 0 0	0.110V	1 1 0 0	0.160V
0 0 0 1	0.055V	0 1 0 1	0.080V	1 0 0 1	0.120V	1 1 0 1	0.180V
0 0 1 0	0.060V	0 1 1 0	0.090V	1 0 1 0	0.130V	1 1 1 0	0.200V
0 0 1 1	0.065V	0 1 1 1	0.100V	1 0 1 1	0.140V	1 1 1 1	0.220V

●ビット3～0 – CCDL3～0 : 充電過電流検出値 (Charge Over-current Detection Level)

これらのビットは表22-5.で定義される、充電過電流検出用のRSENSE電圧値を設定します。

表22-5. CCDL3～0に対応する充電過電流検出用RSENSE電圧

CCDL3～0	Typ	CCDL3～0	Typ	CCDL3～0	Typ	CCDL3～0	Typ
0 0 0 0	0.050V	0 1 0 0	0.070V	1 0 0 0	0.110V	1 1 0 0	0.160V
0 0 0 1	0.055V	0 1 0 1	0.080V	1 0 0 1	0.120V	1 1 0 1	0.180V
0 0 1 0	0.060V	0 1 1 0	0.090V	1 0 1 0	0.130V	1 1 1 0	0.200V
0 0 1 1	0.065V	0 1 1 1	0.100V	1 0 1 1	0.140V	1 1 1 1	0.220V

22.7.5. BPSCD – 電池保護短絡検出値レジスタ (Battery Protection Short-circuit Detection Level Register)

ビット (\$F4)	7	6	5	4	3	2	1	0	
	–	–	–	–	SCDL3	SCDL2	SCDL1	SCDL0	BPSCD
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7～4 – Res : 予約 (Reserved Bits)

これらのビットは将来の使用に対して予約されており、常に0として読まれます。

● ビット3～0 – SCDL3～0 : 短絡検出値 (Short-circuit Detection Level)

これらのビットは表22-6で定義される、放電方向での短絡検出用のRSENSE電圧値を設定します。

表22-6. SCDL3～0に対応する短絡検出用RSENSE電圧

SCDL3～0	Typ	SCDL3～0	Typ	SCDL3～0	Typ	SCDL3～0	Typ
0 0 0 0	0.100V	0 1 0 0	0.140V	1 0 0 0	0.220V	1 1 0 0	0.320V
0 0 0 1	0.110V	0 1 0 1	0.160V	1 0 0 1	0.240V	1 1 0 1	0.360V
0 0 1 0	0.120V	0 1 1 0	0.180V	1 0 1 0	0.260V	1 1 1 0	0.400V
0 0 1 1	0.130V	0 1 1 1	0.200V	1 0 1 1	0.280V	1 1 1 1	0.440V

22.7.6. BPDUV – 電池保護最低電圧レジスタ (Battery Protection Deep Under Voltage Register)

ビット (\$F3)	7	6	5	4	3	2	1	0	
	–	–	DUVT1	DUVT0	DUDL3	DUDL2	DUDL1	DUDL0	BPDUV
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7,6 – 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おり、常に0として読まれます。

● ビット3 – DUVT1,0 : 最低電圧確認時間 (Deep Under-voltage Timing)

これらのビットは最低電圧保護(確認用)遅延を設定します。

表22-7. DUVT1,0に対応する最低電圧遅延

DUVT1,0	Typ
0 0	750ms
0 1	1000ms
1 0	1250ms
1 1	1500ms

● ビット3～0 – DUDL3～0 : 最低電圧検出値 (Deep Under Voltage Detection Level)

これらのビットは最低電圧検出値を設定します。

表22-8. DUDL3～0に対応する最低電圧検出値

DUDL3～0	Typ	DUDL3～0	Typ	DUDL3～0	Typ	DUDL3～0	Typ
0 0 0 0	4.71V	0 1 0 0	5.97V	1 0 0 0	7.23V	1 1 0 0	8.49V
0 0 0 1	5.03V	0 1 0 1	6.29V	1 0 0 1	7.54V	1 1 0 1	8.80V
0 0 1 0	5.34V	0 1 1 0	6.60V	1 0 1 0	7.86V	1 1 1 0	9.11V
0 0 1 1	5.66V	0 1 1 1	6.91V	1 0 1 1	8.17V	1 1 1 1	9.43V

22.7.7. BPIR – 電池保護割り込みレジスタ (Battery Protection Interrupt Register)

ビット (\$F2)	7	6	5	4	3	2	1	0	
	DUVIF	COCIF	DOCIF	SCIF	COCIE	DOCIE	DUVIE	SCIE	BPIR
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 – DUVIF : 最低電圧早期警告割り込み要求フラグ** (Deep Under-voltage Early Warning Interrupt Flag)

VFETピンの電圧が**最低電圧検出値**よりも低く、**最低電圧遅延**の残りが250msの場合にだけ、DUVIFが設定(1)になります。このフラグはそこへ論理1を書くことによって解除(0)されなければなりません。

- **ビット6 – COCIF : 充電過電流保護割り込み要求フラグ** (Charge Over-current Protection Activated Interrupt Flag)

充電過電流保護が働くとCOCIFが設定(1)になります。このフラグはそこへ論理1を書くことによって解除(0)されなければなりません。

- **ビット5 – DOCIF : 放電過電流保護割り込み要求フラグ** (Discharge Over-current Protection Activated Interrupt Flag)

放電過電流保護が働くとDOCIFが設定(1)になります。このフラグはそこへ論理1を書くことによって解除(0)されなければなりません。

- **ビット4 – SCIF : 短絡保護割り込み要求フラグ** (Short-circuit Protection Activated Interrupt Flag)

短絡保護が働くとSCIFが設定(1)になります。このフラグはそこへ論理1を書くことによって解除(0)されなければなりません。

- **ビット3 – DUVIE : 最低電圧早期警告割り込み許可** (Deep Under-voltage Early Warning Interrupt Enable)

このDUVIEビットは最低電圧早期警告割り込み(要求フラグ)によって起こされる割り込みを許可します。

- **ビット2 – COCIE : 充電過電流保護割り込み許可** (Charge Over-current Protection Activated Interrupt Enable)

このCOCIEビットは充電過電流保護割り込み(要求フラグ)によって起こされる割り込みを許可します。

- **ビット1 – DOCIE : 放電過電流保護割り込み許可** (Discharge Over-current Protection Activated Interrupt Enable)

このDOCIEビットは放電過電流保護割り込み(要求フラグ)によって起こされる割り込みを許可します。

- **ビット0 – SCIE : 短絡保護割り込み許可** (Short-circuit Protection Activated Interrupt Enable)

このSCIEビットは短絡保護割り込み(要求フラグ)によって起こされる割り込みを許可します。

電池保護割り込み要求フラグの1つが設定(1)され、対応する割り込み許可ビットと**ステータスレジスタ(SREG)の全割り込み許可(I)ビット**が設定(1)なら、MCUは電池保護割り込みベクタへ飛びます。応用ソフトウェアは割り込み原因を決定するために電池保護割り込みレジスタを読まなければなりません。この割り込み要求フラグは割り込み処理ルーチンが実行される時に解除(0)されず、これらは論理1を書くことによって解除(0)されなければなりません。

23. FET制御

電池保護回路からのFET禁止制御信号に加え、CPUはFET制御/状態レジスタ(FCSR)への書き込みによって充電FET(C-FET)、放電FET(D-FET)、またはその両方を禁止できます。CPUには電池保護回路によって禁止されているFETの許可が決して許されないことに注意してください。FET制御は図23-1.で示されます。

8ビットのタイマ/カウンタ0からのPWM出力(OC0B)はCFET、予備充電FET(PC-FET)、またはその両方の直接駆動に設定できます。これは電池セルの充電制御に対して有用となり得ます。このPWMはタイマ/カウンタ制御レジスタA(TCCR0A)とタイマ/カウンタ制御レジスタB(TCCR0B)の比較B出力選択(COM0B1,0)と波形生成種別(WGM02~0)ビットによって設定されます。OC0Bピンが出力として設定される必要がないことに注意してください。これは、このPWM出力がOC0Bピンの占有なしにC-FETやPC-FETの駆動に使い得ることを意味します。

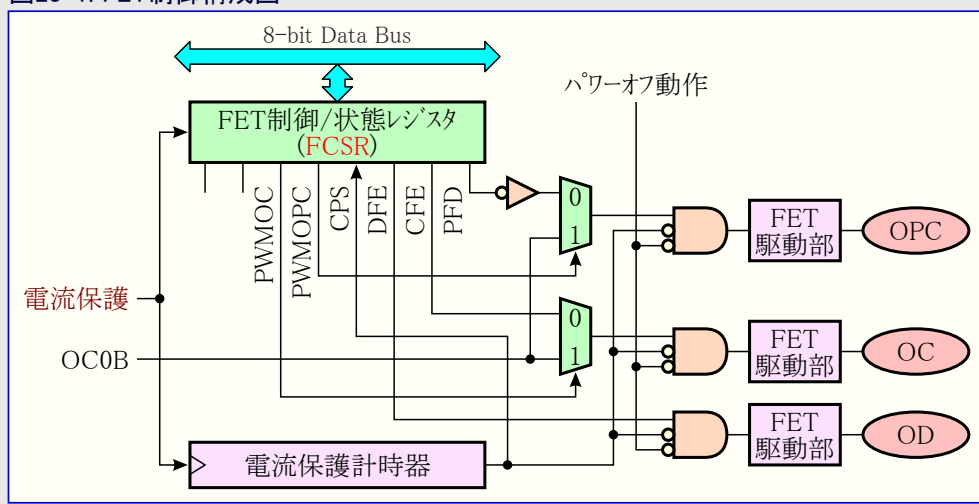
C-FETが禁止でD-FETが許可される場合、放電電流はC-FETの本体ドレインダイオードを通して流れるでしょう。またその逆も同様です。この状態での発熱問題を避けるため、ソフトウェアは充電電流が流れる時にD-FETが禁止されず、放電電流が流れる時にC-FETが禁止されないことを保証しなければなりません。

電池が徹底的に放電されてしまうと、充電器が接続される時に大きな波動電流になるかもしれません。電流制限抵抗を通して電池を先に予備充電することが推奨されます。この目的に対してATmega406は予備充電FET(PC-FET)制御出力を提供します。この出力は既定で許可されます。

ATmega406がパワーオフ動作へ移行されてしまうと、全FET制御出力が禁止されます。充電器が接続されると、CPUは起動します。パワーオフ動作から起動すると、C-FETとD-FET制御出力が禁止に留まる一方、PC-FETは既定で許可されます。セル電圧が標準充電を許すのに十分な上昇をCPUが検知する時にPC-FET制御出力を禁止してC-FETとD-FET制御出力を許可すべきです。

電流/電池保護が稼動(状態)にされてしまっていた場合、ソフトウェアが外部FETを再度許可し得る前に電流保護計時器が1秒の防止時間を保証します。

図23-1. FET制御構成図



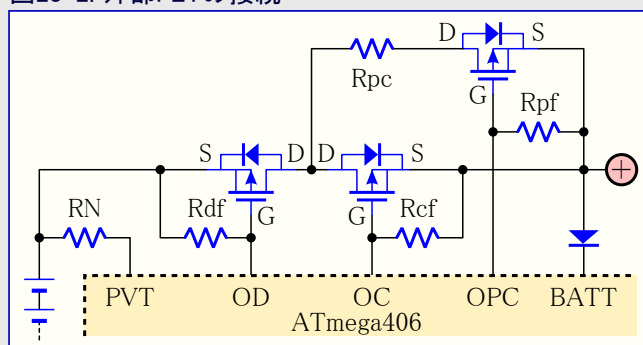
23.1. FET駆動部

OD, OC, OPCへの外部FET接続は図23-2.で示されます。

FETをONにする時に出力はFETの発熱を避けるために素早くゲート(G)をLowに引き込みます。FETが完全にONされると、出力は消費電流を減らすために動作種別を変更します。ON時のFETに対するゲート(G)/ソース(S)間電圧(V_{GS_ON})は $13V \pm 15\%$ に制限されます。

外部FETを禁止する時にFET駆動部出力はほぼ0のFETゲート(G)/ソース(S)間電圧を作ってゲート(G)電圧を素早くソース(S)ピン電位へ押し上げます。これがFETを禁止し、そしてFET駆動部出力は消費電流を減らすために動作種別をHi-Zに切り替えます。外部抵抗はFETが再び許可されるまでゲート(G)/ソース(S)間電圧を0に保ち、(許可で)そのゲート(G)は上記のようにLowへ引かれます。

図23-2. 外部FETの接続



23.2. FET制御用レジスタ

FET制御器はCPUと異なるクロック範囲で動作します。新しい値がFET制御/状態レジスタ(FCSR)へ書かれる毎に、その値はFET制御器クロック範囲に同期化されなければなりません。この同期化の間、後続するこのレジスタへの書き込みは行なわれません。従ってこのレジスタへの書き込み後、3 超低電力発振器周期+3 CPUクロック周期の保護時間が必要とされます。

電池保護割り込み(BPINT)を処理する時にソフトウェアはFET制御/状態レジスタ(FCSR)を読むことだけが推奨されます。

23.2.1. FCSR – FET制御/状態レジスタ (FET Control and Status Register)

ビット (\$F0)	7	6	5	4	3	2	1	0	
	–	–	PWMOC	PWMOPC	CPS	DFE	CFE	PFD	FCSR
Read/Write	R	R	R/W	R/W	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7,6 – 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おり、常に0として読まれます。

● ビット5 – PWMOC : 充電FET PWM駆動 (Pulse Width Modulation of OC output)

PWMOCビットが解除(0)されると、充電FET許可(CFE)ビットと電池保護回路がOC出力(ピン)を制御します。このビットが設定(1)されると、OC出力(ピン)は8ビット タイマ/カウンタ0からのPWM出力(OC0B)と電池保護回路からの反転電流保護信号の論理ANDです。

● ビット4 – PWMOPC : 予備充電FET PWM駆動 (Pulse Width Modulation of OPC output)

PWMOPCビットが解除(0)されると、予備充電FET禁止(PFD)ビットと電池保護回路がOPC出力(ピン)を制御します。このビットが設定(1)されると、OPC出力(ピン)は8ビット タイマ/カウンタ0からのPWM出力(OC0B)と電池保護回路からの反転電流保護信号の論理ANDです。

● ビット3 – CPS : 電流保護状態フラグ (Current Protection Status)

このCPSビットは電流保護の状態を示します。このビットは電流保護計時器動作時に設定(1)され、防止時間経過時に解除(0)されます。

● ビット2 – DFE : 放電FET許可 (Discharge FET Enable)

このDFEビットが解除(0)されると、放電FETは電池保護回路の状態に拘らず禁止されます。このビットが設定(1)されると、放電FETの状態は電池保護回路によって決められます。電流保護信号(図23-1.)が設定(1)されると、このビットは解除(0)されます。

● ビット1 – CFE : 充電FET許可 (Charge FET Enable)

このCFEビットが解除(0)されると、充電FETは電池保護回路の状態に拘らず禁止されます。このビットが設定(1)されると、充電FETの状態は電池保護回路によって決められます。電流保護信号(図23-1.)が設定(1)されると、このビットは解除(0)されます。

● ビット0 – PFD : 予備充電FET禁止 (Precharge FET Disable)

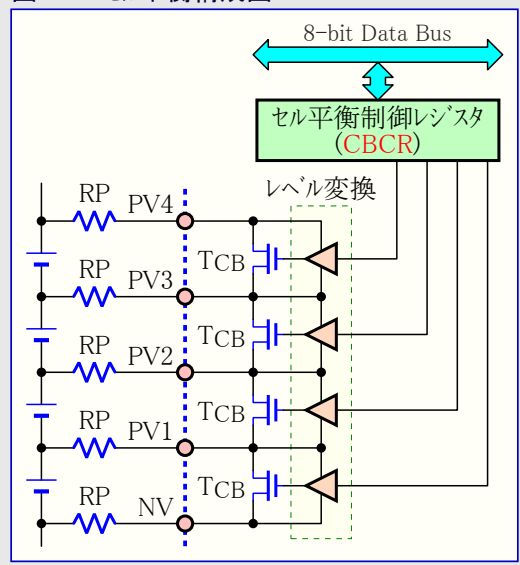
このPFDビットは予備充電FETの完全な制御を提供します。PFDビットが解除(0)されると、予備充電FETは許可されます。PFDビットが設定(1)されると、予備充電FETは禁止されます。電流保護信号(図23-1.)が設定(1)されると、このビットは解除(0)されます。

24. 電池セル平衡

ATmega406は電池のセル平衡用FETを内蔵します。本チップは直列電池セルの各々に対して1つのセル平衡FETを提供します。これらのFETはソフトウェアで実装されるセル平衡手法を可能にする応用ソフトウェアによって直接制御されます。これらのFETは個別電池セルと並列に接続されます。セル平衡は図24-1.で描かれます。本図は4セル構成を示します。セル平衡FETはパワーオフ動作で禁止されます。

セル平衡FET(TCB)を通る代表的な電流は2mAです。セル平衡FETはセル平衡制御レジスタ(CBCR)によって制御されます。隣接したFETは同時に許可できません。2つの隣接FETの許可を試みると、両方が禁止されます。

図24-1. セル平衡構成図



24.1. 電池セル平衡用レジスタ

24.1.1. CBCR – セル平衡制御レジスタ (Cell Balancing Control Register)

ビット (\$F1)	7	6	5	4	3	2	1	0	
	–	–	–	–	CBE4	CBE3	CBE2	CBE1	CBCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7～4 – Res : 予約 (Reserved Bits)

これらのビットは将来の使用に対して予約されており、常に0として読まれます。

● ビット3 – CBE4 : セル平衡4許可 (Cell Balancing Enable 4)

このビットが設定(1)されると、PV3とPV4端子間の統合したセル平衡FETが許可されます。このビットが解除(0)されると、セル平衡FETは禁止されます。これらのセル平衡FETはパワーオフ動作で常に禁止されます。CBE3が設定(1)されると、CBE4は設定(1)できません。

● ビット2 – CBE3 : セル平衡3許可 (Cell Balancing Enable 3)

このビットが設定(1)されると、PV2とPV3端子間の統合したセル平衡FETが許可されます。このビットが解除(0)されると、セル平衡FETは禁止されます。これらのセル平衡FETはパワーオフ動作で常に禁止されます。CBE2またはCBE4が設定(1)されると、CBE3は設定(1)できません。

● ビット1 – CBE2 : セル平衡2許可 (Cell Balancing Enable 2)

このビットが設定(1)されると、PV1とPV2端子間の統合したセル平衡FETが許可されます。このビットが解除(0)されると、セル平衡FETは禁止されます。これらのセル平衡FETはパワーオフ動作で常に禁止されます。CBE1またはCBE3が設定(1)されると、CBE2は設定(1)できません。

● ビット0 – CBE1 : セル平衡1許可 (Cell Balancing Enable 1)

このビットが設定(1)されると、NVとPV1端子間の統合したセル平衡FETが許可されます。このビットが解除(0)されると、セル平衡FETは禁止されます。これらのセル平衡FETはパワーオフ動作で常に禁止されます。CBE2が設定(1)されると、CBE1は設定(1)できません。

25. 2線直列インターフェース (TWI:Two-wire Serial Interface, I²C)

25.1. 特徴

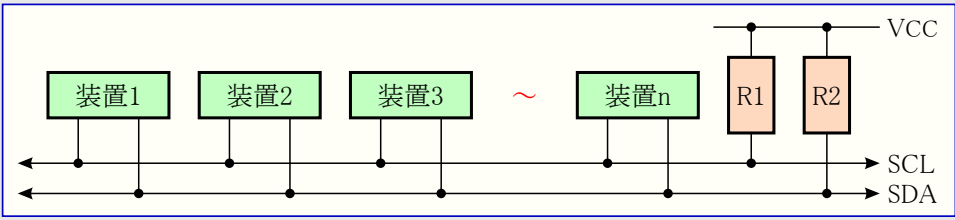
- ・ 2本のバス信号線のみ必要な、単純ながら強力な柔軟な通信インターフェース
- ・ 主装置動作と従装置動作の両方を支援
- ・ 送信装置または受信装置として動作可能
- ・ 7ビットのアドレス空間が128までの異なる従装置アドレスを許容
- ・ 複数主装置の調停支援
- ・ 4MHzクロック動作で100kHzまでのデータ転送速度を達成
- ・ 上昇/下降(スレーブ)制限された出力駆動回路
- ・ バス信号線のスパイクを排除する雑音消去回路
- ・ 一斉呼び出しを含む完全に設定変更可能な従装置アドレスの支援
- ・ AVRが休止形態の時のアドレス認証(一致)起動

24ページの「PRR0 – 電力削減レジスタ」のPRTWIビットは2線直列インターフェース部を許可するために0を書かれなければなりません。

25.2. 2線直列インターフェース バスの定義

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

図25-1. 2線直列(TWI)バス構成



25.2.1. TWI用語定義

次の定義は本項で度々使われます。

表25-1. TWI用語定義

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

25.2.2. 電気的な相互接続

図25-1で描かれたように両方のバス信号線はプルアップ抵抗を通して正供給電圧に接続されます。全てのTWI準拠装置のバス駆動部はオープンドレインかオープン コレクタです。これはインターフェースの動作のために重要なワイアードAND機能を実現します。TWIバス信号線のLowレベルは1つまたはより多くのTWI装置の0出力時に生成されます。Highレベルは全TWI装置がHi-Z出力時の出力で、プルアップ抵抗に信号線をHighへ引き上げさせます。どんなバス動作を許すのにも、TWIバスに接続した全てのAVRデバイスが電力供給されなければならないことに注意してください。

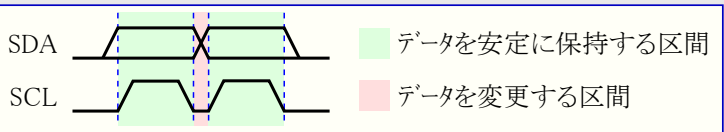
このバスに接続できる装置数は7ビットの従装置アドレス空間と400pFのバス容量制限によってのみ制限されます。TWIの電気的特性の詳細仕様は148ページの「2線直列インターフェース特性」で与えられます。

25.3. データ転送とフレーム形式

25.3.1. ビット転送

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

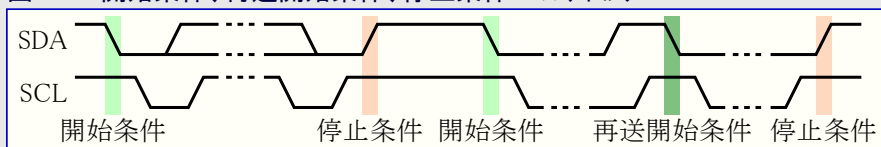
図25-2. データの有効性



25.3.2. 開始条件と停止条件

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

図25-3. 開始条件、再送開始条件、停止条件 バス タイミング



25.3.3. アドレス パケット形式

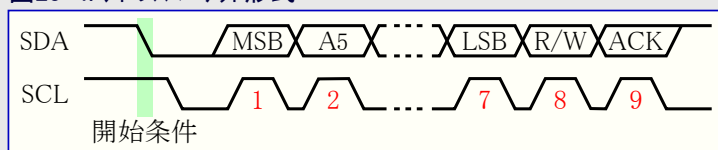
TWIバスに送信した全てのアドレス パケットは7ビットのアドレス ビット、1ビットの方向(Read/Write)制御ビット、1ビットの応答ビットから成る9ビットです。方向(R/W)ビットが設定(1)されると読み出し操作が実行され、さもなければ書き込み操作が実行されるべきです。従装置がアドレス指定されたことを認証すると、9番目のSCL(ACK)周期でSDAを**Low**へ引くことによって確認応答すべきです。アドレス指定された従装置が忙しいまたはその他の理由で主装置の要求を扱えない場合、**確認応答(ACK)**クロック周期でSDA信号線を**High**のままにすべきです。主装置はその後に**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

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

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

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

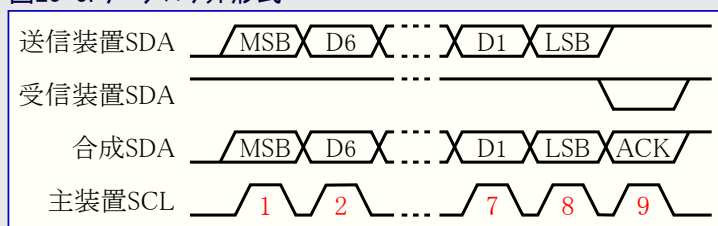
図25-4. アドレス パケット形式



25.3.4. データ パケット形式

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

図25-5. データ パケット形式

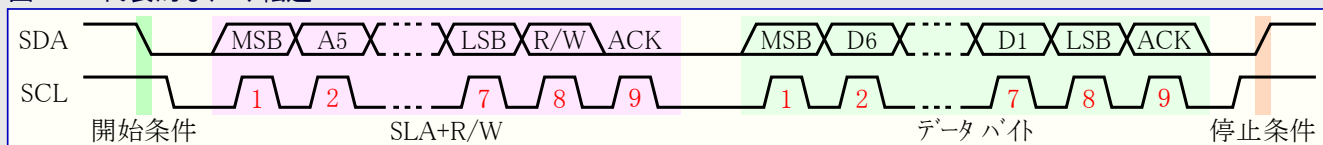


25.3.5. 転送内でのアドレス パケットとデータ パケットの組み合わせ

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

図25-6.は代表的なデータ転送を示します。応用ソフトウェアによって実装されたソフトウェア規約に依存して、様々なデータがSLA+R/Wと停止条件間に送信できることに注意してください。

図25-6. 代表的なデータ転送



25.4. 複数主装置バス システムの調停と同期

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

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

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

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

図25-7. 複数主装置間でのSCL同期化

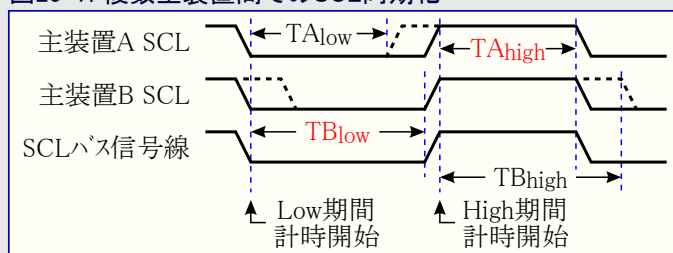
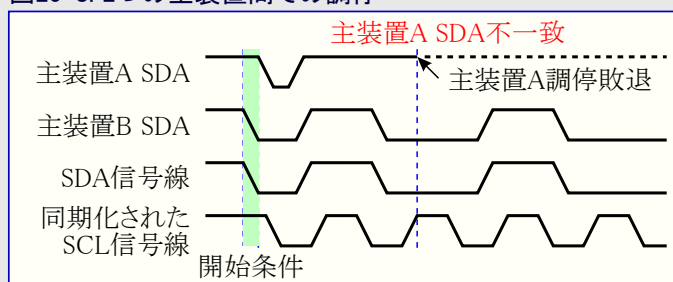


図25-8. 2つの主装置間での調停



調停が次の状態間で許されないことに注意してください。

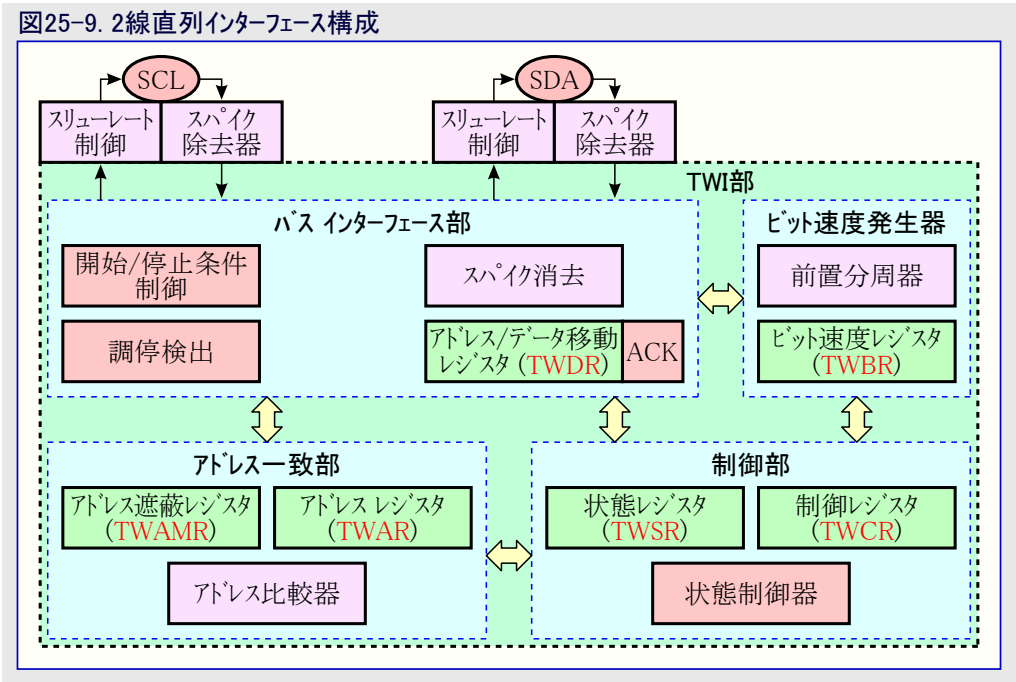
- 再送開始条件とデータ ビット間
- 停止条件とデータ ビット間
- 再送開始条件と停止条件間

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

(訳補) 同じデータ パケット数とは、或る主装置が最後まで調停を継続し、他の主装置がパケットを残している場合を想定しています。

25.5. TWI部の概要

図25-9.で示されるようにTWI部は様々な部分から成ります。赤文字で示された(訳注:原文は太線で描かれた)全てのレジスタはAVRデータバスを通してアクセス可能です。



25.5.1. SCLとSDAピン

これらのピンはAVR TWIをMCUシステムのその他とインターフェースします。出力駆動部はTWI仕様に適合させるためのスローレート(上昇/下降)制限器を含みます。入力段は50nsよりも短いスパイクを除去するスパイク除去部を含みます。

25.5.2. ビット速度発生器

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

$$\text{SCL周波数} = \frac{\text{CPUクロック周波数}}{16 + 2 \times (\text{TWBR}) \times \text{前置分周値}}$$

TWBR

: TWI ビット速度レジスタ値

前置分周値

: TWI状態レジスタ内TWPSで指定(97頁の表25-2.参照)

- 注:
- ・ プルアップ抵抗値はSCL周波数とバス信号線の容量性負荷に応じて選択されるべきです。プルアップ抵抗の値については148頁の表30-3.をご覧ください。
 - ・ TWIクロックは4MHzです。18頁の「校正付き高速RC発振器」をご覧ください。

25.5.3. バス インターフェース部

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

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

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

25.5.4. アドレス一致部

アドレス一致部は受信したアドレスバイトがTWI アドレスレジスタ(TWAR)の7ビットアドレスと一致するかを検査します。TWARで一斉呼び出し検出許可(TWGCE)ビットが1を書かれると、全ての到着アドレスビットは一斉呼び出しアドレスに対しても比較されます。アドレス一致で制御部は通知され、正しい処置を行うことを許します。TWIはTWI制御レジスタ(TWCR)の設定によって、そのアドレスへの応答をするかもしれないし、しないかもしれません。アドレス一致部はAVR MCUが主装置によってアドレス指定された場合にMCUを起動できる休止形態の1つの時でも、アドレスを比較できます。

25.5.5. 制御部

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

TWI割り込み要求フラグ(TWINT)は次の場合に設定(1)されます。

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

25.6. TWI用レジスタ

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

ビット (\$B8)	7	6	5	4	3	2	1	0	
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7～0 – TWBR7～0 : TWIビット速度選択 (TWI Bit Rate Register)

TWBRはビット速度発生器用の分周値を選びます。ビット速度発生器は主装置動作でのSCLクロック周波数を生成する周波数分周器です。ビット速度の計算については94頁の「[ビット速度発生器](#)」をご覧ください。

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

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

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

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

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

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

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

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

TWEAビットに0を書くことによって一時的かつ仮想的に装置を2線直列バスから切り離すことができます。アドレス認証はその後に再びTWEAビットへ1を書くことによって再開できます。

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

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

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

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

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

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

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

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

● ビット1 – Res : 予約 (Reserved Bit)

このビットは予約されており、常に0として読まれます。

●ビット0 – TWIE : TWI割り込み許可 (TWI Interrupt Enable)

このビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、TWI割り込み要求フラグ(TWINT)が1である限り、TWI割り込み要求が活性に(発生)されます。

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

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

●ビット7~3 – TWS7~3 : TWI状態 (TWI Status)

これら5ビットはTWI論理回路と2線直列バスの状態を反映します。各種状態符号は、103頁の表25-3.~109頁の表25-6.で記述されます。TWSRから読む値が5ビットの状態符号と2ビットの前置分周値の両方を含むことに注意してください。応用設計者は状態ビットを検査する時に前置分周器ビットを0で隠すべきです。これは前置分周器設定に関係なく状態検査を行います。この手法は特記事項を除いてこのデータシート内で使われます。

●ビット2 – Res : 予約 (Reserved Bit)

このビットは予約されており、常に0として読まれます。

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

これらのビットは読み書きでき、ビット速度の前置分周器を制御します。

ビット速度を計算するには94頁の「ビット速度発生器」をご覧ください。TWPS1,0の値はこの式で使われます。

表25-2. TWIビット速度前置分周器選択

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

25.6.4. TWDR – TWIデータレジスタ (TWI Data Register)

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

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

●ビット7~0 – TWD7~0 : TWIデータ (TWI Data)

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

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

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

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

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

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

これら7ビットはTWI部の従装置アドレスを構成します。

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

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

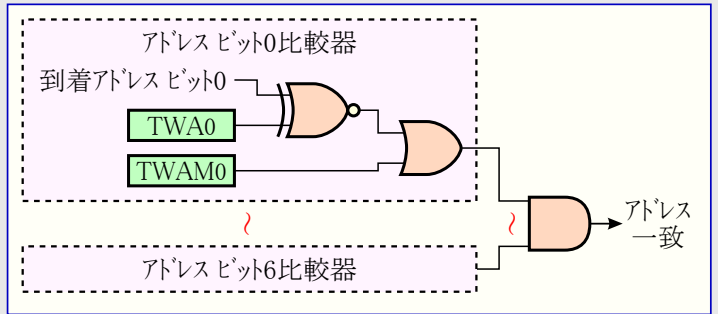
25.6.6. TWAMR – TWI(従装置)アドレス遮蔽レジスタ (TWI (Slave) Address Mask Register)

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

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

TWAMRは7ビットの従装置アドレス遮蔽値を格納できます。TWAMR内の各ビットはTWI(従装置)アドレスレジスタ(TWAR)内の対応するアドレスビットを遮蔽(禁止)します。遮蔽ビットが1に設定されると、その後のアドレス一致論理回路は到着アドレスビットとTWAR内の対応ビット間の比較を無視します。図25-10.はアドレス一致論理回路を詳細に示します。

図25-10. TWIアドレス一致論理 構成図



● ビット0 – Res : 予約 (Reserved Bit)

このビットは使われず、常に0として読まれます。

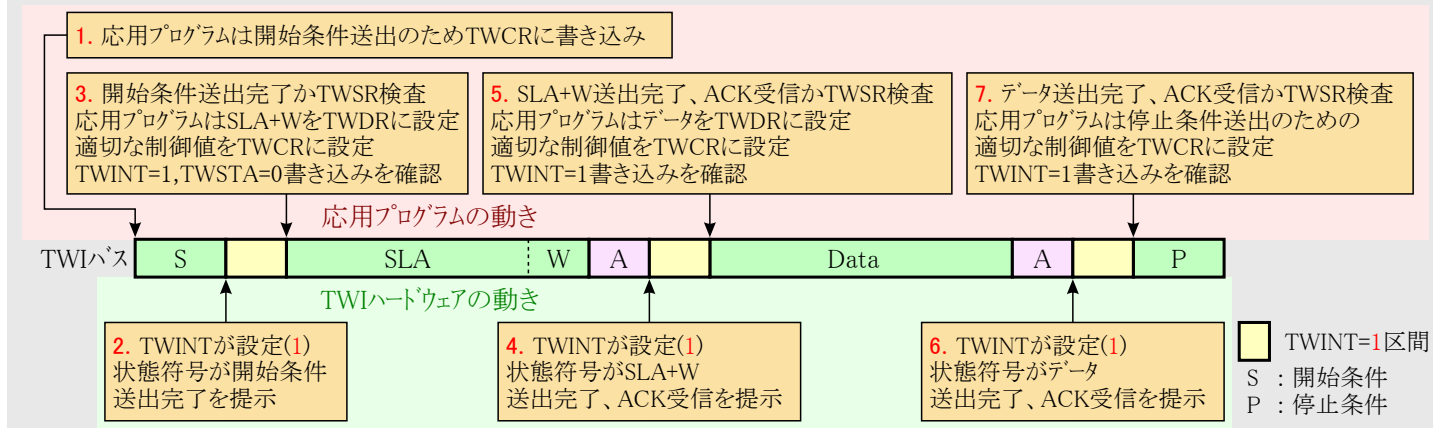
25.7. TWIの使用法

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

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

図25-11はアプリケーションソフトウェアがTWIハードウェアにどうインターフェースできるかの簡単な例です。この例では主装置が単一データパルソを従装置に送信しようします。この内容はかなり大雑把ですので、より詳細な説明が本項の後に続きます。希望した動きを実現する簡単なコード例も示されます。

図25-11. 代表的な送信でのアプリケーションプログラムとTWIのインターフェース



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

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

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

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

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

注: 5頁の「コード例について」をご覧ください。

25.8. 転送種別

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

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

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

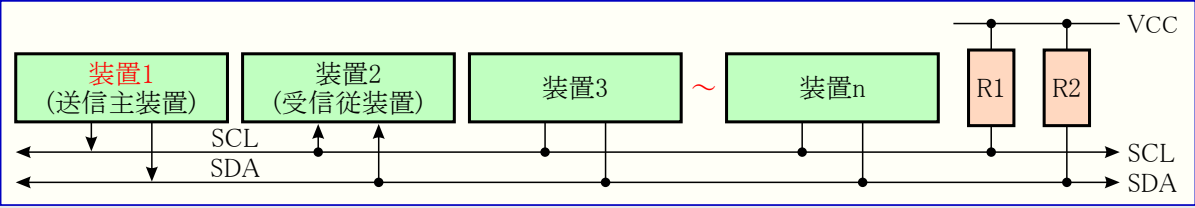
図25-13. ~19.内の楕円(訳注:原文は円)はTWI制御レジスタ(TWCR)のTWI割り込み要求フラグ(TWINT)が設定(1)されたことを示すのに使われます。この楕円内の番号は前置分周選択ビットが0で遮蔽されたTWI状態レジスタ(TWSR)に保持した状態符号を表します。これら位置での動きはTWI転送の継続または完了が応用(ソフトウェア)によって行われなければなりません。TWI転送はソフトウェアによってTWINTフラグが解除(0)されるまで一時停止されます。

TWI割り込み要求フラグ(TWINT)が設定(1)される時のTWI状態レジスタ(TWSR)の状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対する必要なソフトウェア動作や後続の直列転送の詳細は表25-3. ~6.で与えられます。これらの表に於いて前置分周選択ビットが0で遮蔽されていることに注意してください。

25.8.1. 送信主装置動作

送信主装置動作では何バイトかのデータが受信従装置へ送信されます(図25-12.参照)。主装置動作へ移行するには**開始条件**が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。**SLA+W**が送信されると送信主装置(MT)へ移行し、**SLA+R**が送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は**前置分周選択ビット**が0か、または0で遮蔽されることが前提です。

図25-12. 送信主装置動作でのデータ転送



開始条件はTWCRに次の値を書くことによって送出されます。

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

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。TWSTAは**開始条件**を送出するために1を書かれねばならず、TWINTはTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに**開始条件**を生成します。**開始条件**が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、TWSRの状態符号が\$08(表25-3.参照)になります。送信主装置へ移行するには**SLA+W**が送信されなければなりません。これはTWDRに**SLA+W**を書くことによって行います。その後、転送を継続するためにTWINTビットは1の書き込みによって解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

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

SLA+Wが送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$18,\$20,\$38です。これら状態符号の各々に対する適切な動作は表25-3.で詳述されます。

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

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

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

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

再送開始条件はTWCRに次の値を書くことによって生成されます。

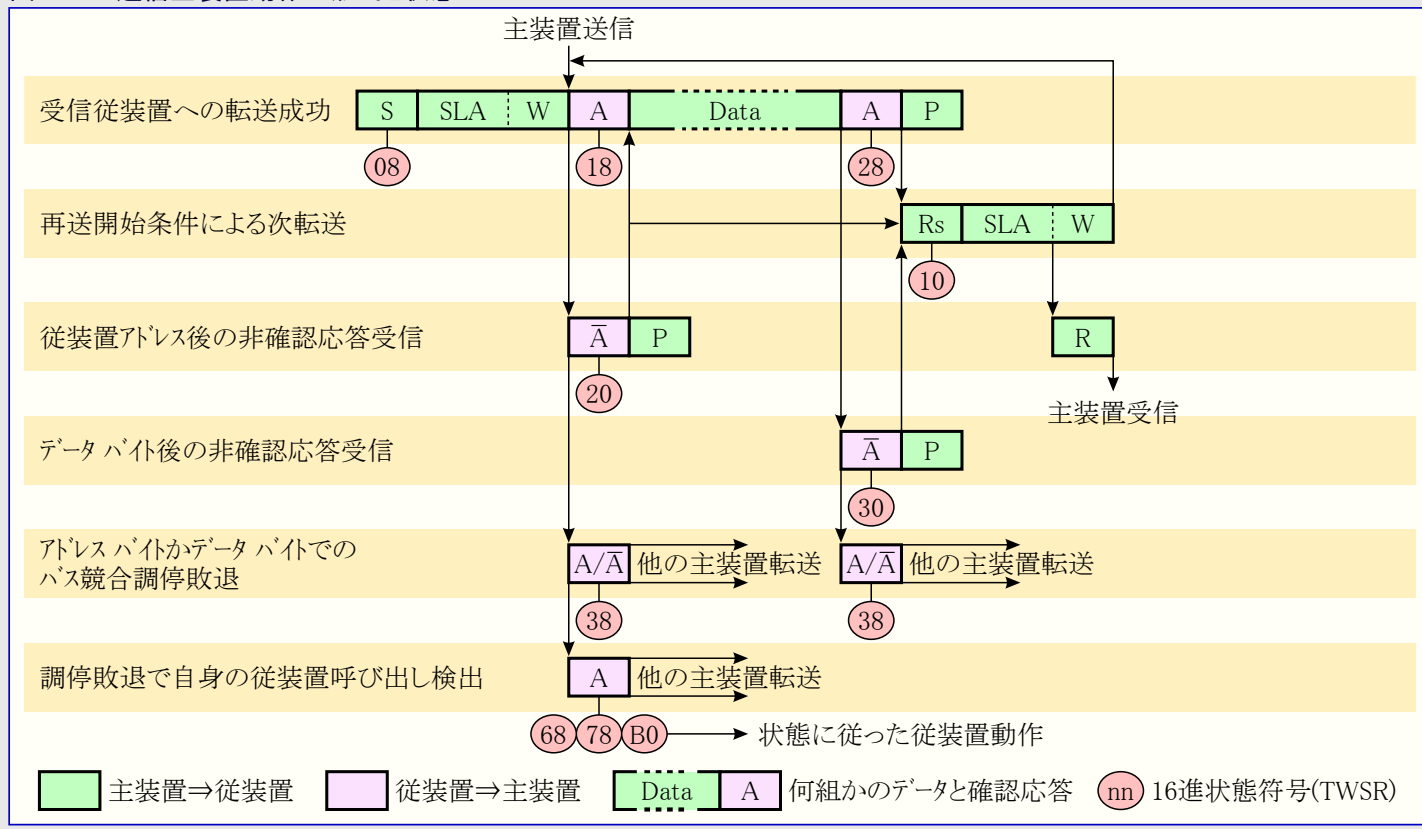
ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	—	TWIE
設定値	1	X	1	0	X	1	0	X

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

表25-3. 送信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

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

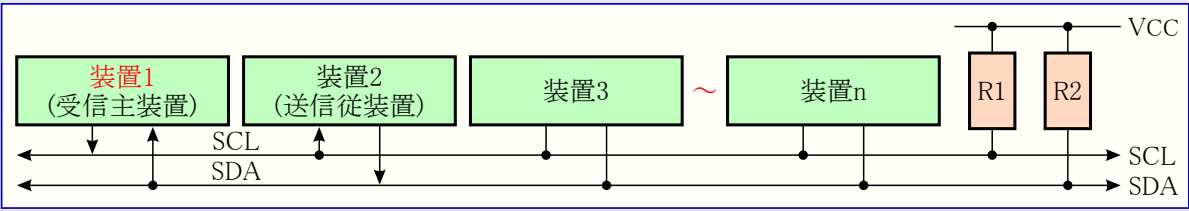
図25-13. 送信主装置動作の形式と状態



25.8.2. 受信主装置動作

受信主装置動作では何バイトかのデータが送信従装置から受信されます(図25-14.参照)。主装置動作へ移行するには**開始条件**が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。**SLA+W**が送信されると送信主装置(MT)へ移行し、**SLA+R**が送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は**前置分周選択ビット**が**0**か、または**0**で遮蔽されることが前提です。

図25-14. 受信主装置動作でのデータ転送



開始条件はTWCRに次の値を書くことによって送出されます。

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

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。**TWSTA**は**開始条件**を送出するために1を書かれねばならず、**TWINT**はTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに**開始条件**を生成します。**開始条件**が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、**TWSR**の状態符号が\$08(表25-4.参照)になります。受信主装置へ移行するには**SLA+R**が送信されなければなりません。これは**TWDR**に**SLA+R**を書くことによって行います。その後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

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

SLA+Rが送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$38,\$40,\$48です。これら状態符号の各々に対する適切な動作は表25-4.で詳述されます。

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

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

再送開始条件はTWCRに次の値を書くことによって生成されます。

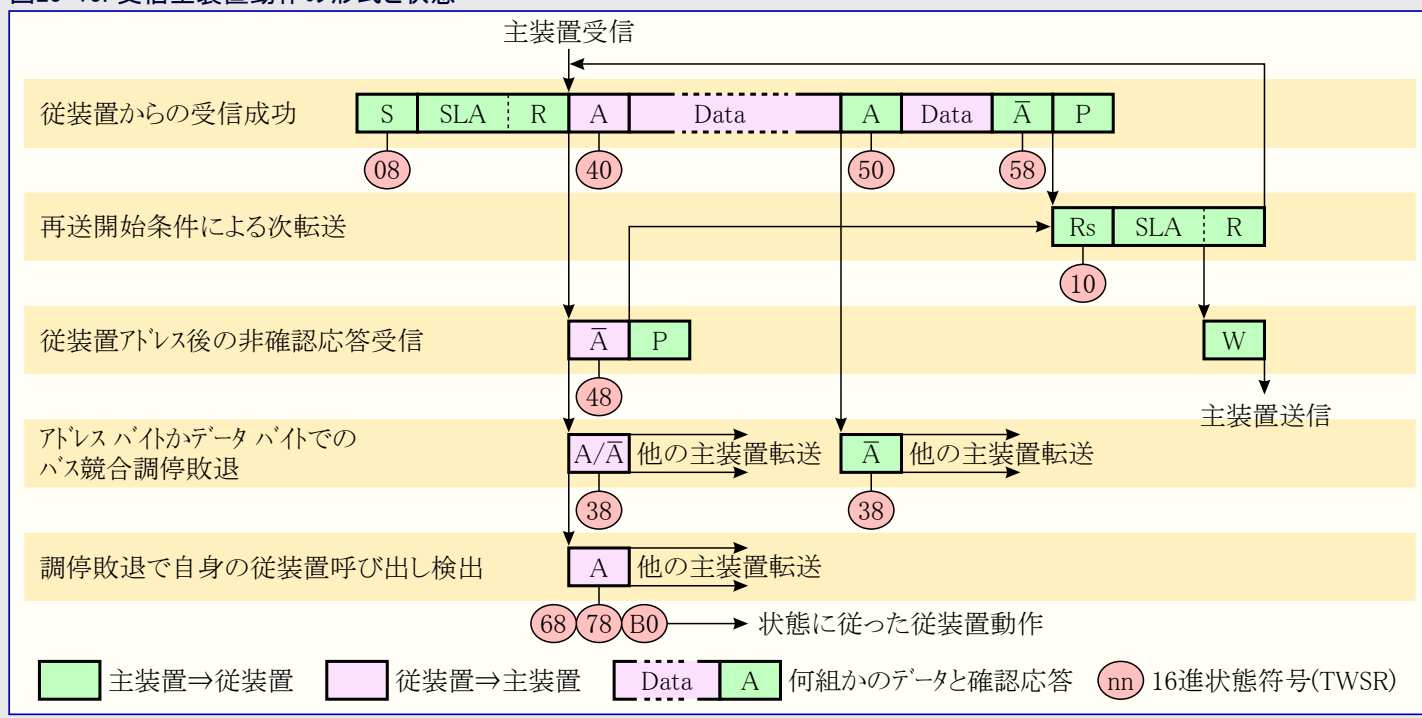
ビット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

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

表25-4. 受信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

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

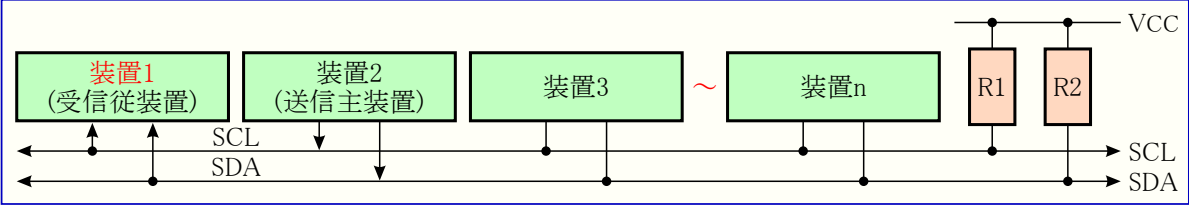
図25-15. 受信主装置動作の形式と状態



25.8.3. 受信従装置動作

受信従装置動作では何バイトかのデータが送信主装置から受信されます(図25-16参照)。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。

図25-16. 受信従装置動作でのデータ転送



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

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

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

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

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

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可なら、一斉呼び出しアドレス)とそれに続くデータ方向ビットによってアドレス指定されるまで待機します。方向ビットが0(W)ならばTWIは受信従装置で動作し、さもなく(1(R)ならば送信従装置へ移行されます。自身の従装置アドレスとWビットが受信されてしまった後にTWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表25-5で詳述されます。受信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$68,\$78参照)

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

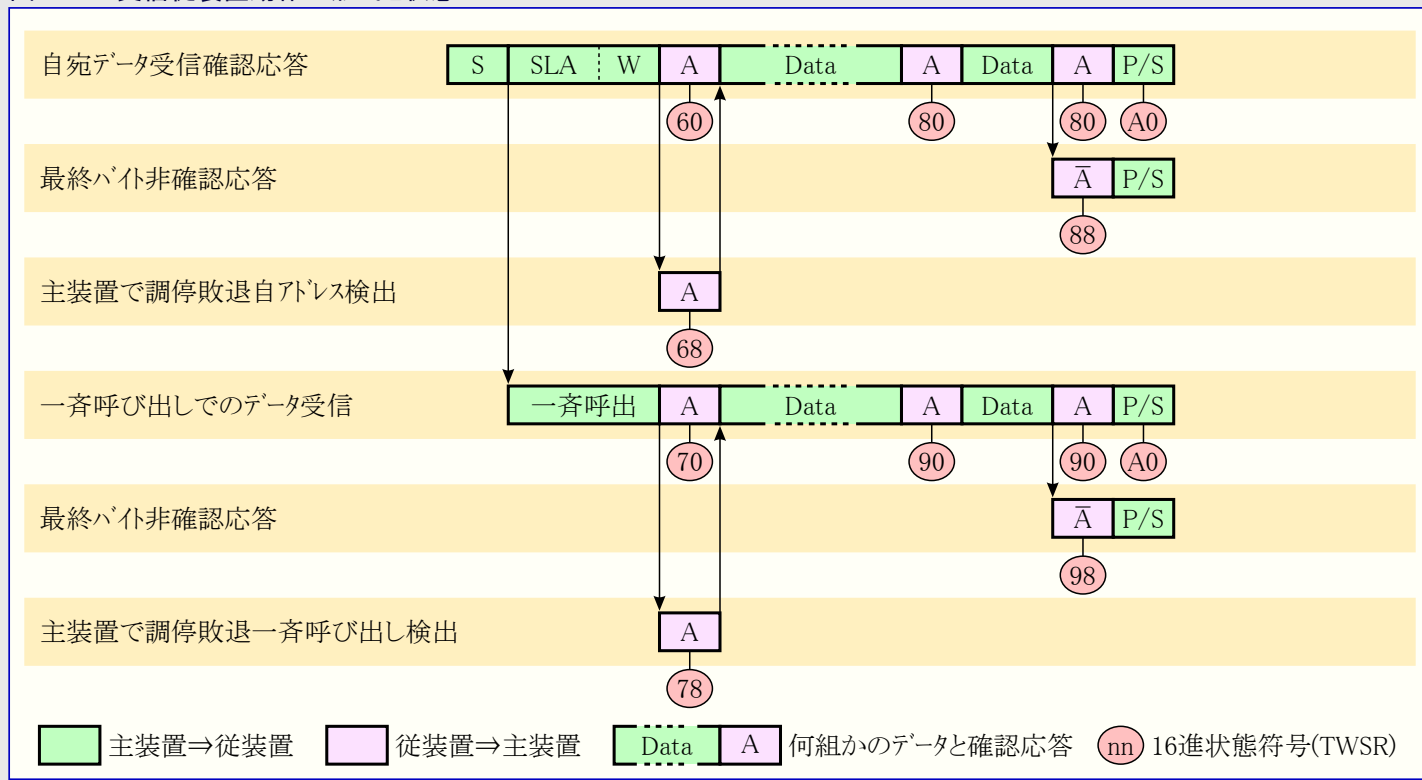
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが(1書き込みによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないこと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

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

表25-5. 受信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

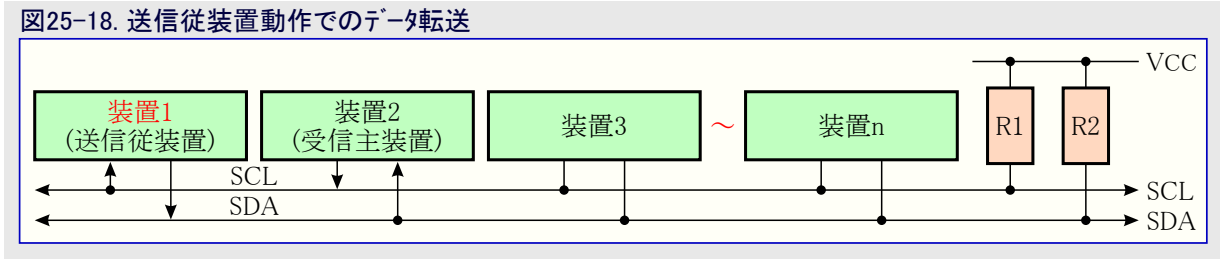
状態符号 (TWSR)	直前の動作と バスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$60	自宛SLA+W受信 ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$68	主装置のSLA+R/Wで 調停敗退/自宛SLA+W 受信/ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$70	一斉呼び出し受信 ACK応答	なし	X	0	1	0	データ受信、NACK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

図25-17. 受信従装置動作の形式と状態



25.8.4. 送信従装置動作

送信従装置動作では何バイトかのデータが送信主装置へ送信されます(図25-18参照)。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。



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

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

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

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

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

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可ならば一斉呼び出しアドレス)とそれに続くデータ方向ビットによってアドレス指定されるまで待機します。方向ビットが1(R)ならばTWIは送信従装置で動作し、さもなくば0(W)ならば受信従装置へ移行されます。自身の従装置アドレスとRビットが受信されてしまった後、TWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表25-6で詳述されます。送信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$B0参照)

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

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

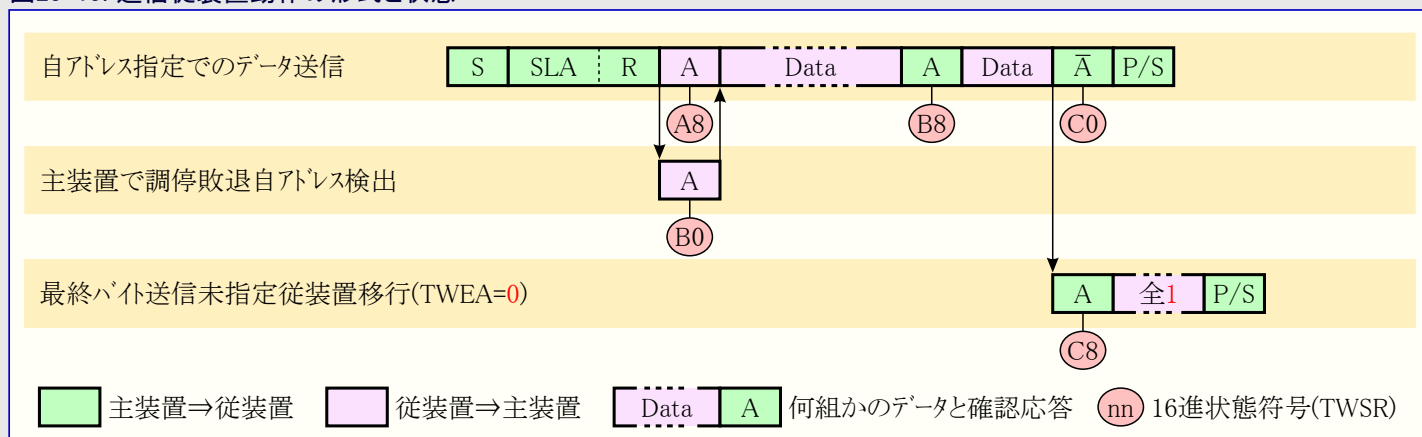
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが1書き込みによって解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないことに気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

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

表25-6. 送信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

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

図25-19. 送信従装置動作の形式と状態



25.8.5. その他の状態

定義したTWI状態に従わない2つの状態符号があります。表25-7をご覧ください。

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

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

表25-7. その他の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

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

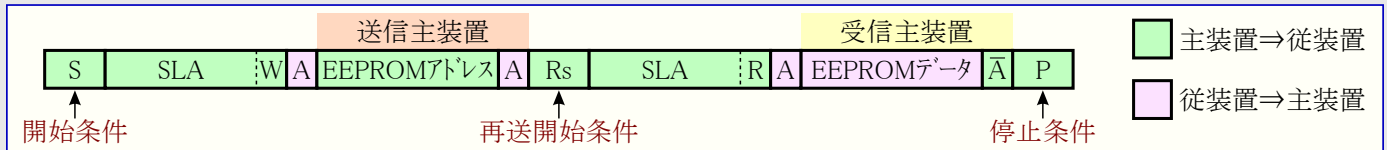
25.8.6. 各種TWI動作種別の組み合わせ

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

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

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

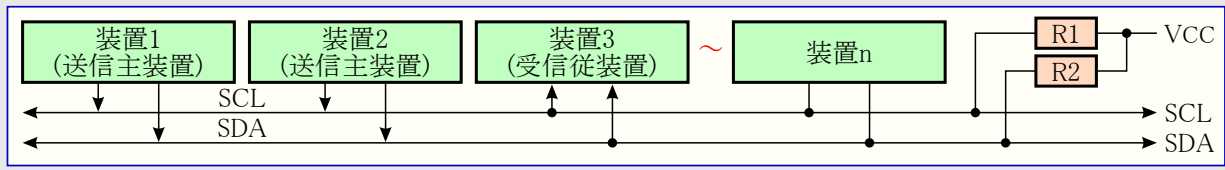
図25-20. 直列EEPROMアクセスでの各種TWI動作種別の組み合わせ



25.9. 複数主装置システムでのバス競合と調停

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

図25-21. バスの競合調停例

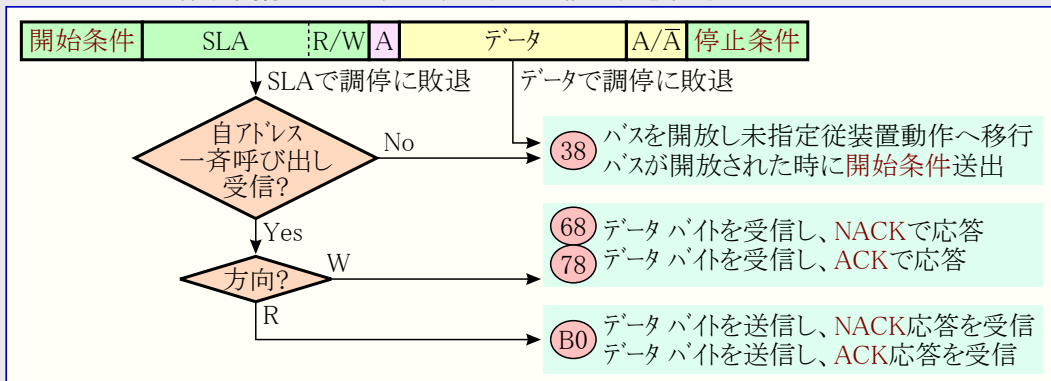


以下で示されるように様々な異なる状況が調停中に起こるかもしれません。

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

これは図25-22で要約されます。利用可能な状態符号は楕円(訳注:原文は円)で与えられます。

図25-22. バスの競合調停によって発生する利用可能な状態符号



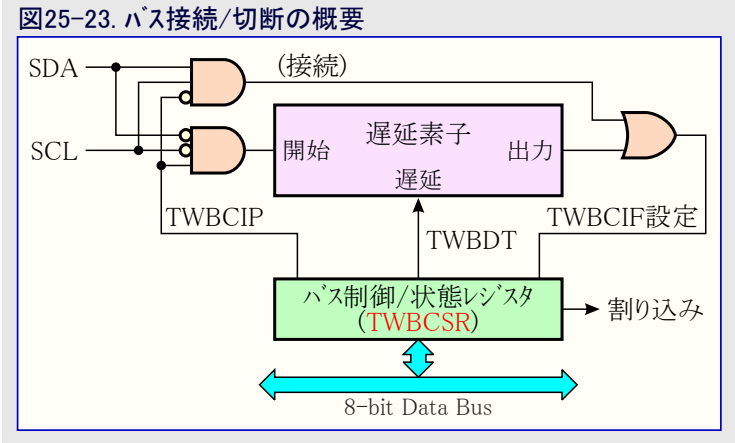
25.10. 2線直列インターフェース用バス接続/切断

バス接続/切断部は2線直列インターフェース(TWI)への追加です。設定ビットを基にTWIバスの接続と切断のどちらでも割り込みが生成できます。

図25-23.はSDAとSCLが各々TWIのデータとクロックであるバス接続/切断論理を描きます。

TWIバスが接続されると、SDAとSCLは同時に両方共Highになります。接続/切断割り込み極性(TWBCIP)ビットが解除(0)されているなら、割り込みが許可されていれば割り込みが実行されます。一旦バスが接続されたなら、TWBCIPビットが設定(1)されるべきです。これはバスが切断される時の検出を可能にし、SDAとSCLの両方がHigh(換言するとバスアイドル状態)毎の繰り返し割り込みを防ぎます。

TWIバスが切断されると、SDAとSCLは同時に両方共Lowになります。TWBCIPビットが設定(1)されている場合、両信号線が設定可能な時間の区間に対してLowに留まり、且つ割り込みが許可されていたなら、割り込みが実行されます。この無効時間の追加により、通常バス通信の最中に両信号線がLowになることによって引き起こされる不要な割り込みが防止されます。



25.10.1. TWBCSR – TWI バス制御/状態レジスタ (TWI Bus Control and Status Register)

ビット (\$BE)	7	6	5	4	3	2	1	0	
	TWBCIF	TWBCIE	–	–	–	TWBDT1	TWBDT0	TWBCIP	TWBCSR
Read/Write	R/W	R/W	R	R	R	R/W	R/W	R/W	
初期値	不定	0	0	0	0	0	0	0	

●ビット7 – TWBCIF : TWI バス接続/切断割り込み要求フラグ (TWI Bus Connect/Disconnect Interrupt Flag)

接続/切断割り込み極性(TWBCIP)ビットを基に、TWBCIFビットはTWIバスが接続または切断される時に設定(1)されます。TWBCIFは対応する割り込み処理ベクタを実行するとき、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TWBCIF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、バス接続/切断割り込み許可フラグ(TWBCIE)とTWBCIFが設定(1)される時にTWIバス接続/切断割り込みが実行されます。リセット中にSDAとSCLの両方がHighの場合、TWBCIFはリセット後に設定(1)されます。さもなければTWBCIFはリセット後に解除(0)されます。

●ビット6 – TWBCIE : TWI バス接続/切断割り込み許可 (TWI Bus Connect/Disconnect Interrupt Enable)

TWBCIEビットとSREGの全割り込み許可(I)ビットが設定(1)されると、TWIバス接続/切断割り込みが許可されます。TWIバス接続/切断が起こる、換言するとバス接続/切断割り込み要求フラグ(TWBCIF)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5~3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

●ビット2,1 – TWBDT1,0 : TWI バス切断認知時間 (TWI Bus Disconnect Time-out Period)

TWBDT1,0ビットはTWI データ(SDA)とクロック(SCL)の両信号線が、TWIバス切断割り込みを生成する前に、どれくらい長くLowでなければならないかを決めます。各設定値と対応する計時終了時間は表25-8.で示されます。

表25-8. TWIバス切断認知時間

TWBDT1,0	計時終了時間
0 0	250ms
0 1	500ms
1 0	1000ms
1 1	2000ms

●ビット0 – TWBCIP : TWI バス接続/切断割り込み極性 (TWI Bus Connect/Disconnect Interrupt Polarity)

TWBCIPビットはTWIバス接続/切断割り込みフラグ(TWBCIF)がバス接続またはバス切断で設定(1)されるべきかを決めます。TWBCIPが解除(0)されると、TWBCIFはバス接続で設定(1)されます。TWBCIPが設定(1)されると、TWBCIFはバス切断で設定(1)されます。

26. JTAGインターフェースと内蔵デバッグ機能

26.1. 特徴

- ・ JTAGインターフェース (IEEE標準1149.1準拠)
- ・ 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によって支援された内蔵デバッグ機能

26.2. 概要

AVRのIEEE標準1149.1適合JTAGインターフェースは以下について使えます。

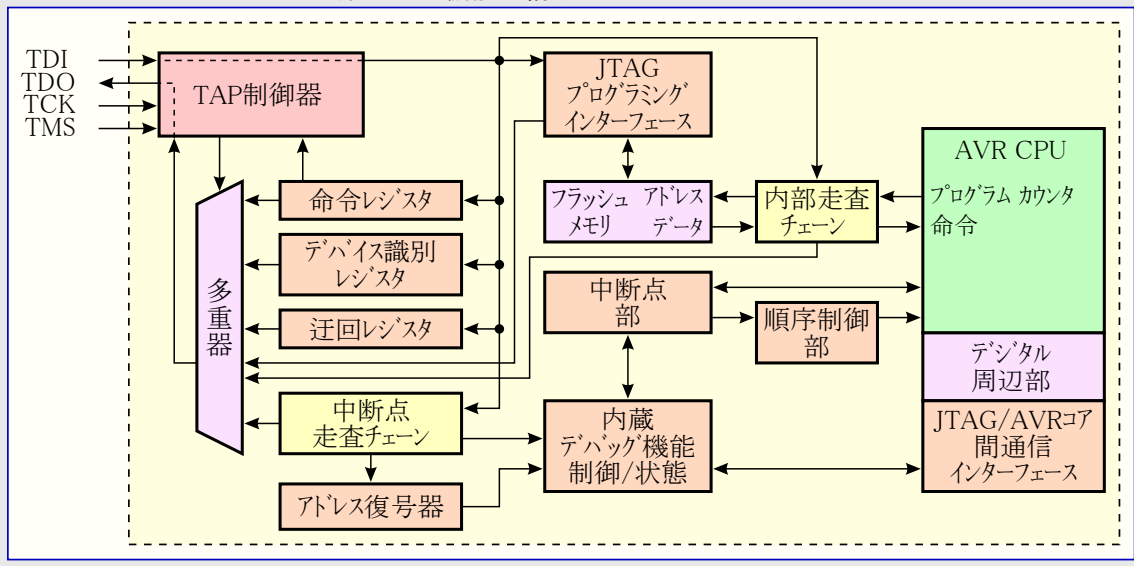
- ・ 不揮発性メモリ、ヒューズ ビット、施錠ビットの読み書き(プログラミング)
- ・ 内蔵デバッグ機能による実装デバッグ

簡単な記述は次項で与えられます。JTAGインターフェース経由のプログラミングについての詳細な記述は136頁の「[JTAGインターフェース経由のプログラミング](#)」項で得られます。内蔵デバッグ機能支援は独自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せん。

図26-1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。

(実際には現実と仮定の各種データレジスタから成る)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26-1. JTAGインターフェースと内蔵デバッグ機能の構成図



26.3. 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。

JTAGエンハンスが非プログラム(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。

26.4. TAP制御器

TAP制御器は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段無限順次回路です。

図26-2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。

この資料内の定義として、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。

- 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3LSB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択し、選択したデータレジスタ周辺回路を制御します。

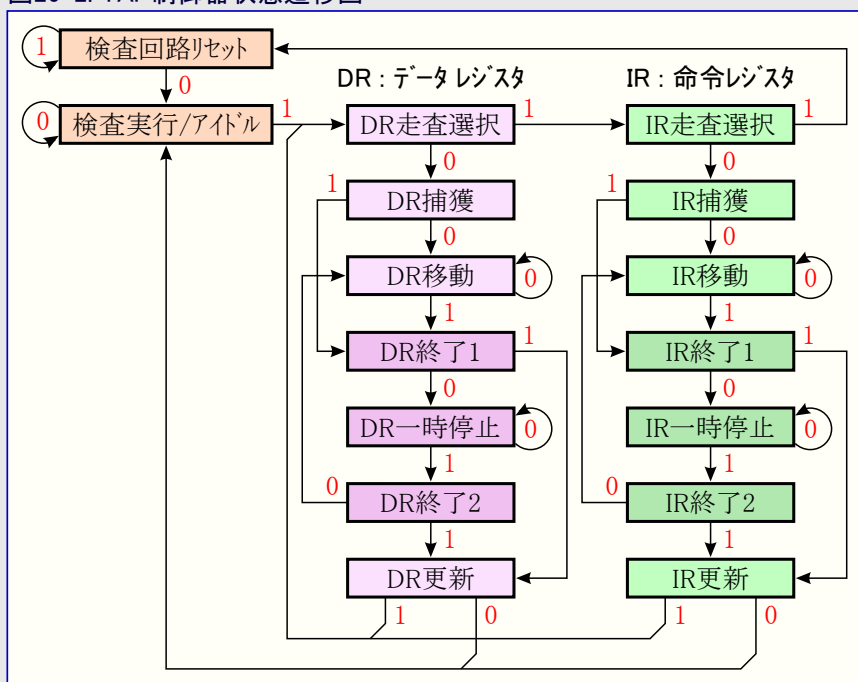
- 検査実行/アイドル状態へ再度移行するためにTMSへ順次1,1,0を与えます。この(JTAG)命令はIR更新状態通過で移動レジスタから並列出力にラッチされます。IR終了1、IR一時停止、IR終了2の各状態は順次回路の誘導操作のためだけに使われます。
- データレジスタ(DR)移動状態へ移行するためにTCKの上昇端でTMSへ順次1,0,0を与えます。この状態中、TCKの上昇端でTDI入力から(JTAG命令レジスタ内の現在のJTAG命令で)選択したデータレジスタに値を移動入力します。DR移動状態に留まるためにMSBを除く全ビットの入力中、TMS入力はLowに保持されなければなりません。データのMSB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出力が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。選択した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択するかもしれない、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115頁の「参考文献」に記載された文献を参照してください。

図26-2. TAP制御器状態遷移図



26.5. 内蔵デバッグ機能の使用

図26-1.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ユーザー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。要点だけが本資料で示されます。

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。

26.6. 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。

26.6.1. 独自命令0 : \$8 (内蔵デバッグ機能アクセス用独自JTAG命令)

26.6.2. 独自命令1 : \$9 (内蔵デバッグ機能アクセス用独自JTAG命令)

26.6.3. 独自命令2 : \$A (内蔵デバッグ機能アクセス用独自JTAG命令)

26.6.4. 独自命令3 : \$B (内蔵デバッグ機能アクセス用独自JTAG命令)

26.7. 内蔵デバッグに関連するI/Oメモリ内のレジスタ

26.7.1. OCDR – 内蔵デバッグ レジスタ (On-chip Debug Register)

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。

(訳注) 原書本位置のMCUCR記述は整合性のため以下の26.9.項内に記述しています。

26.8. 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存在し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すプログラミングとプログラミング特殊JTAG命令の記述は136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。

参考文献 (訳注:整合性のため本項追加)

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

26.9. JTAGインターフェースに関連するI/Oメモリ内のレジスタ (訳注:整合性のため本節を追加)

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

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	–	–	PUD	–	–	IVSEL	IVCE	MCUCR
Read/Write	R/W	R	R	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、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。

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

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	BODRF	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によって選択した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。

27. ブート ロータ 支援 – 書き込み中読み出し可能な自己プログラミング

ブート ロータ 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 ロータ 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 ロータ 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 ロータ 領域内のプログラム コードはブート ロータ 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 ロータ 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 ロータ メモリの容量はヒューズで設定可能で、ブート ロータ は個別に設定可能な2組の独立したブート 施錠ビットを持ちます。これは異なる保護レベルを選択する独特な柔軟性を使用者に与えます。

27.1. 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 ロータ 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(128頁の「ページ容量」参照)。このページ構成は通常動作に影響を及ぼしません。

27.2. フラッシュ メモリの応用領域とブート ロータ 領域

フラッシュ メモリは応用領域とブート ロータ 領域の2つの主な領域で構成されます(図27-2.参照)。各領域の容量は125頁の表27-7.と図27-2.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。

27.2.1. 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)によって選択できます(118頁の表27-2.参照)。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 ロータ コードも決して格納し得ません。

27.2.2. ブート ロータ 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 ロータ 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 ロータ 領域用保護レベルはブート ロータ 施錠ビット(ブート 施錠ビット1)によって選択できます(118頁の表27-3.参照)。

27.3. 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 ロータ 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117頁の図27-2.と125頁の表27-8.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 ロータ 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 ロータ 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。

27.3.1. 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)ビットが論理1として読めます。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118頁の「SPMCSR – SPM命令制御/状態レジスタ」をご覧ください。

27.3.2. 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27-1. 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27-1. RWW領域とNRWW領域の関係

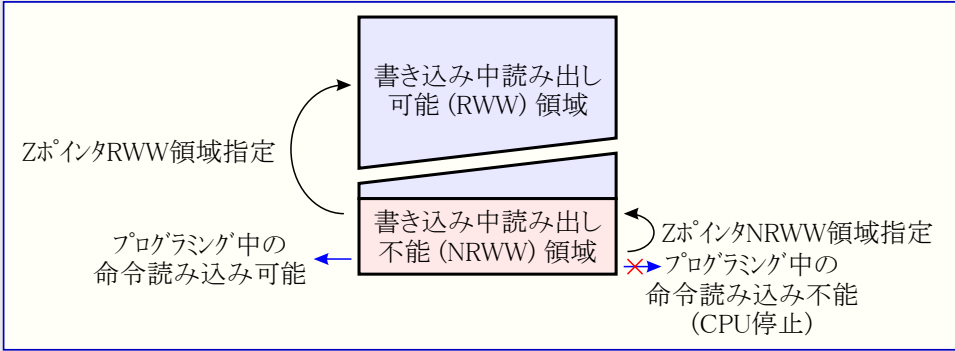
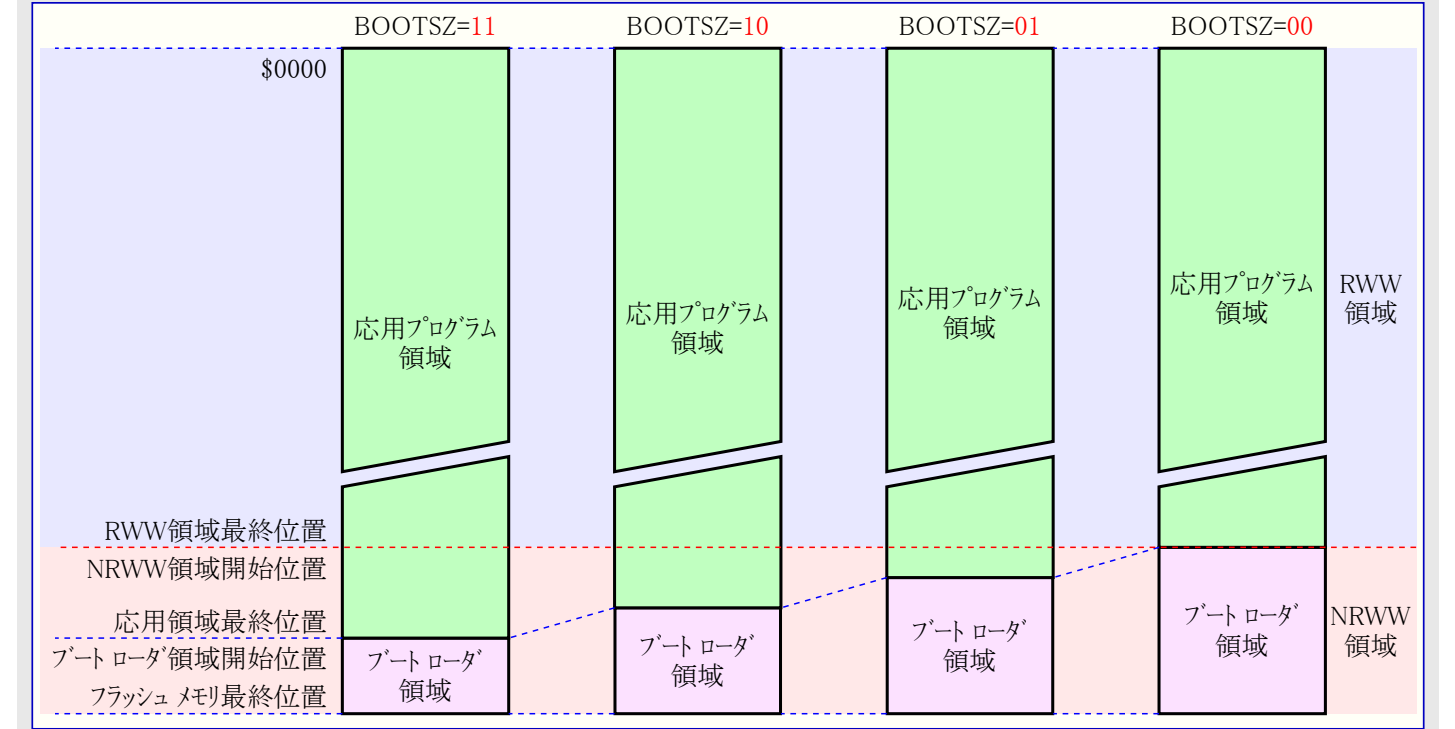


図27-2. 選択によるプログラム用フラッシュメモリの領域分割



注: 図内の各パラメータについては125頁の表27-7.で与えられます。

27.4. フォート ローダ 施錠ビット

フォート ローダ 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。フォート ローダは個別に設定可能な独立した2組のフォート 施錠ビットを持ちます。これは異なる保護レベルを選択する独特な柔軟性を使用者に与えます。

使用者は以下を選択できます。

- 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 メモリのフォート ローダ 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27-2と表27-3をご覧ください。フォート 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27-2. 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、フォート ローダ 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	フォート ローダ 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、割り込みベクタがフォート ローダ 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27-3. フォート ローダ 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命令がフォート ローダ 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命令はフォート ローダ 領域に書くことを許されません。
3	0	0	SPM命令によるフォート ローダ 領域への書き込みと、応用領域でのLPM命令によるフォート ローダ 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命令によるフォート ローダ 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、フォート ローダ 領域での実行時に割り込みが禁止されます。

27.5. フォート ローダ プログラムへの移行

フォート ローダ への移行は応用プログラムから分岐(Jump)または呼び出し(Call)によって行います。これはTWI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 ベクタがフォート ローダ 領域開始アドレスを指示するようにフォート リセット(BOOTRST)ヒューズをプログラム(0)できます。この場合、フォート ローダ 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(フォート ローダ)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。これは一旦フォート リセット ヒューズがプログラム(0)されると、リセット ベクタは常にフォート ローダ リセットを指示し、このヒューズが直列(JTAG)プログラミングまたは並列プログラミング インターフェースを通してのみ変更できることを意味します。

表27-4. フォート 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	フォート ローダ リセット	フォート ローダ 開始アドレス(125頁の表27-7.参照)
1	応用リセット	\$0000

27.5.1. SPMCSR – SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register)

このレジスタはブートローダー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	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	SPMCSR
Read/Write	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIEビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みは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 – SIGRD : 識票列読み出し (Signature Row Read)

このビットがSPMENと同時に1を書かれると、次の3クロック周期内のLPM命令は識票列からバイトを転送先レジスタに読みます。詳細については123頁の「ソフトウェアからの識票列読み出し」をご覧ください。

SIGRDとSPMENが設定(1)された後の4クロック周期内のSPM命令は無効です。この操作は将来の使用に対して予約されており、使われるべきではありません。

● 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)読み出しに対して防がれます。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施錠ビットを設定します。R1のデータとZポインタのアドレスは無視されます。BLBSETビットは施錠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施錠ビットのどちらかを転送先レジスタに読みます。詳細については122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6ビットに100001, 010001, 001001, 000101, 000011, 000001以外のどんな組み合わせを書いても無効です。

27.6. 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

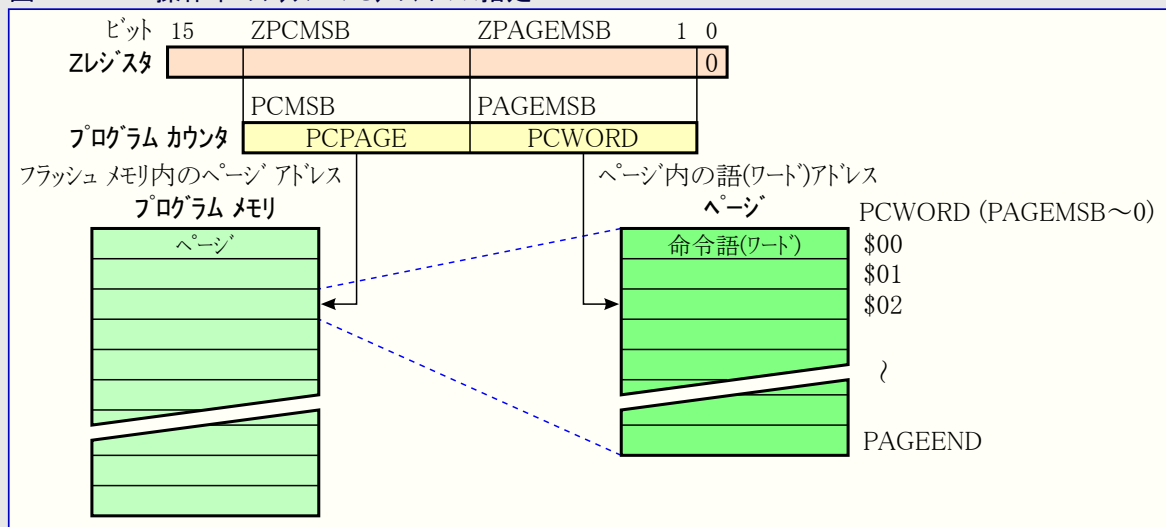
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

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(128頁の「ページ容量」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27-3.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 ロータ 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 ロータ 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27-3. 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125頁の表27-9.で一覧されます。
PCPAGEとPCWORDは128頁の「ページ容量」で一覧されます。(訳注: 共通性から追加)

27.7. 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。

ページの一部分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、その後に変更され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(リド モデファイ ライト)機能をブート ロータが提供します。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124頁の「アセンブリ言語による簡単なブート ロータ例」をご覧ください。

27.7.1. 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ポイント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ポイントのPCPAGEに書かれなければなりません。この操作中、Zポイント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。

27.7.2. 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ポイントにアドレス、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ポイントの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。

27.7.3. ページ書き込みの実行

ページ書き込みを行うにはZポイント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ポイントの)PCPAGEに書かれなければなりません。Zポイントの他のビットは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。

27.7.4. 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33頁の「割り込み」で記述されます。

27.7.5. 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。

27.7.6. 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33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124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。

27.7.7. SPM命令によるブートローダ施錠ビットの設定

ブートローダ施錠ビットを解除(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。アクセス可能な施錠ビットはMCUによるどのソフトウェア更新からも応用とブートローダ領域を保護できるブート施錠ビットだけです。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	1	1

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27-2と表27-3をご覧ください。

R0のビット5～2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、対応するブート施錠ビットがプログラム(0)されます。この操作中、Zポイントは関係ありませんが、将来との共通性のために(施錠ビット読み出しに使われるのと同じ)0001でZポイント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、R0のビット7,6,1,0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。

27.7.8. 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プログラム許可(EEPE)ビットを検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前にこのビットが解除(0)されているのを確認することが推奨されます。

27.7.9. 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AVR命令一式説明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127頁の表28-4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127頁の表28-3を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	—	—	—	—	—	—	FHB1	FHB0

プログラム(0)されたヒューズと施錠ビットは0として読めます。非プログラム(1)にされたヒューズと施錠ビットは1として読めます。

27.7.10. ソフトウェアからの識票列読み出し

ソフトウェアから識票列を読むには表27-5.で与えられる識票バイトアドレスをZポイントに設定してSPMCSRのSPMENと識票列読み出し(SIGRD)ビットを設定(1)してください。SPMENとSIGRDビットがSPMCSRに設定された後3CPU周期内にLPM命令が実行されると、識票バイト値が転送先レジスタに格納されます。SPMENとSIGRDビットは識票バイト読み出しの完了で、または3CPU周期内にLPM命令が実行されない場合、自動的に解除(0)されます。SPMENとSIGRDビットが解除(0)されると、LPMはAVR命令一式説明で記述されるように動作します。

表27-5. 識票列アドレス一覧

識票バイト	Zポイント アドレス	識票バイト	Zポイント アドレス
デバイス識票バイト 1 (製造者識別)	\$0000	V-ADC電池セル1利得校正値下位 (注9)	\$0010
デバイス識票バイト 2 (フラッシュ容量)	\$0002	V-ADC電池セル1利得校正値上位 (注9)	\$0011
デバイス識票バイト 3 (デバイス種別)	\$0004	V-ADC電池セル2利得校正値下位 (注9)	\$0012
高速RC発振器校正バイト (注1)	\$0001	V-ADC電池セル2利得校正値上位 (注9)	\$0013
(予約)	\$0003	V-ADC電池セル3利得校正値下位 (注9)	\$0014
低速RC発振器周波数 (注2)	\$0005	V-ADC電池セル3利得校正値上位 (注9)	\$0015
低速RC発振器計時完了時間下位 (注3)	\$0006	V-ADC電池セル4利得校正値下位 (注9)	\$0016
低速RC発振器計時完了時間上位 (注3)	\$0007	V-ADC電池セル4利得校正値上位 (注9)	\$0017
低速RC発振器温度変化予測値下位 (注7)	\$000C	V-ADC ADC0利得校正値下位 (注10)	\$0018
低速RC発振器温度変化予測値上位 (注7)	\$000D	V-ADC ADC0利得校正値上位 (注10)	\$0019
超低電力RC発振器周波数 (注5)	\$0008	V-ADC電池セル1オフセット値 (注12)	\$001C
超低電力RC発振器計時完了時間下位 (注6)	\$000A	V-ADC電池セル2オフセット値 (注12)	\$001D
超低電力RC発振器計時完了時間上位 (注6)	\$000B	V-ADC電池セル3オフセット値 (注12)	\$001E
バンドギャップPTAT電流校正バイト (注4)	\$0009	V-ADC電池セル4オフセット値 (注12)	\$001F
V-ADC列セル1校正値下位 (注8)	\$000E	VPTAT校正校正値下位 (注11)	\$001A
V-ADC列セル1校正値上位 (注8)	\$000F	VPTAT校正校正値上位 (注11)	\$001B

注1: リセット後の既定高速RC発振器校正(FOSCCAL)レジスタ値。

注2: kHzでの低速RC発振器周波数。

注3: μsでの低速RC発振器最速計時完了時間。

注4: BGCCR=\$0F時にVREFで1.1Vを与えるBGCCRについて得られた校正値。

注5: kHzでの超低電力RC発振器周波数。

注6: μsでの超低電力RC発振器最速計時完了時間。

注7: 低速RC発振器温度変化予測値。多くのロットに渡って測定。未実装。

注8: VREF校正の第2段階で使う校正値。この段階は25℃でお客様によって実行されます。格納された値はセル1が85℃で4096mVの時のVADCH/L値です。

注9: V-ADCセル1〜4入力での利得誤差に対する補償に使う校正値。mVでのセルn=VADCH/L×本値÷16384。

注10: V-ADCのADC0入力での利得誤差に対する補償に使う校正値。0.1mVでのVADC0=VADCH/L×本値÷16384。

注11: VTEMP変換からケルビンでの絶対温度計算に使う校正値。Kでの温度=VADCH/L×本値÷16384。

注12: V-ADCセル1〜4入力でのオフセットに対する補償に使う校正値。未実装。

注: 他の全てのアドレスは将来の使用に対して予約されています。

27.7.11. 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

高速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27-6.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27-6. 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

27.7.12. アセンブリ言語による簡単なブート ロード例

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 ロード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 ロード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	; PGSZBはページ内のバイト数です。(PAGESIZEはワード数)
	. ORG SMALLBOOTSTART	;
WRPG:	LDI SPMC, (1<<PGERS) + (1<<SPMEN)	; [ページ消去]
	CALL SPMJ	; ページ消去SPMCSR値を取得
		; ページ消去
	LDI SPMC, (1<<RWWSRE) + (1<<SPMEN)	; [RWW領域読み出し再許可]
	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
	LDI CNTL, LOW (PGSZB)	; [RAMからフラッシュ ページ一時緩衝部へ転送]
	LDI CNTH, HIGH (PGSZB)	; バイト計数器を初期化
	LD R0, Y+	; (削除)
	LD R1, Y+	; RAM上の下位データを取得(ポインタ進行)
	LDI SPMC, (1<<SPMEN)	; RAM上の上位データを取得(ポインタ進行)
	CALL SPMJ	; ページ一時緩衝部書き込みSPMCSR値を取得
	ADIW ZH: ZL, 2	; 対応語(ワード)データをページ一時緩衝部に設定
	SBIW CNTH: CNTL, 2	; ページ一時緩衝部ポインタ進行
	BRNE WLP	; 計数器を減数 (SUBI)
		; 指定バイト数分継続
		; [ページ書き込み]
	SUBI ZL, LOW (PGSZB)	; ページ一時緩衝部先頭にポインタを復帰
	SBCI ZH, HIGH (PGSZB)	; (削除)
	LDI SPMC, (1<<PGWRT) + (1<<SPMEN)	; フラッシュ書き込みSPMCSR値を取得
	CALL SPMJ	; 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)	; RWW領域読み出し許可SPMCSR値を取得
	CALL SPMJ	; RWW領域読み出し許可
		; [読み戻し照合 (任意)]
	LDI CNTL, LOW (PGSZB)	; バイト計数器を初期化
	LDI CNTH, HIGH (PGSZB)	; (削除)
	SUBI YL, LOW (PGSZB)	; RAMデータ先頭にポインタを復帰
	SBCI YH, HIGH (PGSZB)	;
RLP:	LPM R0, Z+	; フラッシュ メモリから1バイト取得(ポインタ進行)
	LD R1, Y+	; RAMから1バイト データを取得(ポインタ進行)
	CPSE R0, R1	; 値一致でスキップ
	JMP ERROR	; 不一致で異常処理へ
;		
	SBIW CNTH: CNTL, 1	; 計数器を減数 (SUBI)
	BRNE RLP	; 指定バイト数分継続
		; [RWW領域へ復帰]
RTN:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRS TMP, RWWSB	; RWW領域多忙でスキップ
	RET	; 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)	; RWW領域読み出し許可SPMCSR値を取得
	CALL SPMJ	; RWW領域読み出し許可
	RJMP RTN	; 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	; SPM命令制御/状態レジスタ値を取得
	SBRC TMP, SPMEN	; 操作可能(直前のSPM完了)でスキップ
	RJMP SPMJ	; 操作可まで待機
;		
	IN TMP, SREG	; ステータス レジスタ値を保存
	CLI	; 全割り込み禁止
WAIT:	SBIC EECR, EEPE	; EEPROMプログラミング中以外でスキップ
	RJMP WAIT	; EEPROMプログラミング完了まで待機
;		
	OUT SPMCSR, SPMC	; SPM動作指定
	SPM	; 対応SPM動作実行
	OUT SREG, TMP	; ステータス レジスタ値を復帰
	RET	; 呼び出し元へ復帰

27.7.13. ATmega406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27-7.～9.で与えられます。

表27-7. 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(ワード)	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～\$4EFF	\$4F00～\$4FFF	\$4F00
1	0	512	8	\$0000～\$4DFF	\$4E00～\$4FFF	\$4E00
0	1	1024	16	\$0000～\$4BFF	\$4C00～\$4FFF	\$4C00
0	0	2048	32	\$0000～\$47FF	\$4800～\$4FFF	\$4800

注: 各種BOOTSZヒューズ設定は図27-2.で示されます。

表27-8. 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	288	\$0000～\$47FF
書き込み中読み出し不能(NRWW)領域	32	\$4800～\$4FFF

注: これら2つの領域についての詳細に関しては117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27-9. 図27-3.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	PC14		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15ビット、PC14～0)
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		Z15	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	PC14～6	Z15～7	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は120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。

28. メモリプログラミング

28.1. プログラムメモリとデータメモリ用施錠ビット

ATmega406は非プログラム(1)のままか、表28-2.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28-1. 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	ブートローダ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28-2. 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(JTAG)または並列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、ブートローダ領域での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	ブートローダ領域での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ダプログラム領域に対する保護
1	1	1	LPM, SPM命令がブートローダ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命令はブートローダ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ダ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ダ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ダ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ダ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、ブートローダ領域での実行時に割り込みが禁止されます。

28.2. ヒューズ ビット

ATmega406には2つのヒューズ ビットがあります。表28-3.~4.は全てのヒューズの概略機能とヒューズ ビット内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

28.2.1. 上位ビット

表28-3. ヒューズ 上位ビット一覧

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)
—	2		1 (非プログラム)
OCDEN (注1)	1	内蔵デバッグ機能(OCD)許可。	1 (非プログラム) OCD禁止
JTAGEN	0	JTAGインターフェース許可。	0 (プログラム) JTAG許可

注1: 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。これは消費電力を増加するかもしれません。

28.2.2. 下位ビット

表28-4. ヒューズ 下位ビット一覧

名称	ビット	意味	既定値
WDTON (注3)	7	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	6	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BOOTSZ1	5	ブート ロード容量選択。(表27-7.参照) (注2)	0 (プログラム)
BOOTSZ0	4		0 (プログラム)
BOOTRST	3	リセット ベクタ(応用領域またはブート ロード領域)選択。	1 (非プログラム) 応用領域
SUT1	2	起動時間選択。 (注1)	1 (非プログラム)
SUT0	1		0 (プログラム)
CKSEL (注4)	0		1 (非プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については18頁の表7-2.をご覧ください。
注2: BOOTSZ1,0既定値は最大ブート ロード容量になります。125頁の表27-7.をご覧ください。
注3: 詳細については31頁の「WDTCSR – ウォッチドッグ タイマ制御レジスタ」をご覧ください。
注4: 非プログラム(1)時、内蔵RC発振器が使われます。このビットのプログラム(0)は検査目的だけで、応用で使われるべきではありません。

ヒューズ ビットの状態はチップ消去によって影響されません。施錠ビット1(LB1)がプログラム(0)されると、ヒューズ ビットが固定されることに注意してください。施錠ビットをプログラム(0)する前にヒューズ ビットをプログラミング(書き込み)してください。

28.2.3. 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す。

28.3. 識票ビット

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ビッ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ビットは分離された空間に存在します。ATmega406の識票ビットを右に示します。

- ① \$000 : \$1E 製造業者Atmelを示します。
② \$001 : \$95 フラッシュ メモリ容量40Kバイトを示します。
③ \$002 : \$07 ②値\$95と合せ、ATmega406を示します。

28.4. 校正ビット

ATmega406は高速RC発振器、低速RC発振器、内部基準電圧、内部温度基準と各電池セル差動入力用の校正値を持ちます。これらのビットは識票アドレス空間内の上位ビットに属しています。リセットの間に、高速RC発振器用の校正値が自動的に対応する校正レジスタ(FOSCCAL)へ書かれます。その他の校正値は応用ソフトウェアによって扱われるべきです。詳細については123頁の「ソフトウェアからの識票列読み出し」をご覧ください。

28.5. ページ容量

表28-5. フラッシュメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
20Kワード (40Kバイト)	64ワード	PC5~0	320	PC14~6	14

表28-6. EEPROMメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
512バイト	4バイト	EEA1~0	128	EEA8~2	8

28.6. 並列プログラミング

本項はATmega406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。

28.6.1. 効率的なプログラミングへの考慮

設定した指令とアドレスはプログラミング中、維持されます。効率的なプログラミングを行うために次を考慮すべきです。

- 複数のメモリ領域を読み書きする時に指令設定は一度だけ必要です。
- チップ消去後のフラッシュメモリと(**EESAVEヒューズ**がプログラム(0)されている場合を除き)EEPROM全体の内容は\$FFなので、値が\$FFのデータ書き込みを飛ばします。
- アドレス上位バイトはフラッシュメモリで新規256語(ワード)枠、EEPROMで新規256バイト枠の読み書き前に一度だけ必要です。この考慮は**識別バイト**読み出しにも適用されます。

28.6.2. 信号名

本項でATmega406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28-1**と**表28-7**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定します。このビット符号化は**表28-9**で示されます。

WRまたはOEパルス送出時、設定された指令が決める動作が実行されます。各種指令は**表28-10**で示されます。**135**頁の**表28-11**は並列プログラミング特性を示します。

図28-1. 並列プログラミング構成図

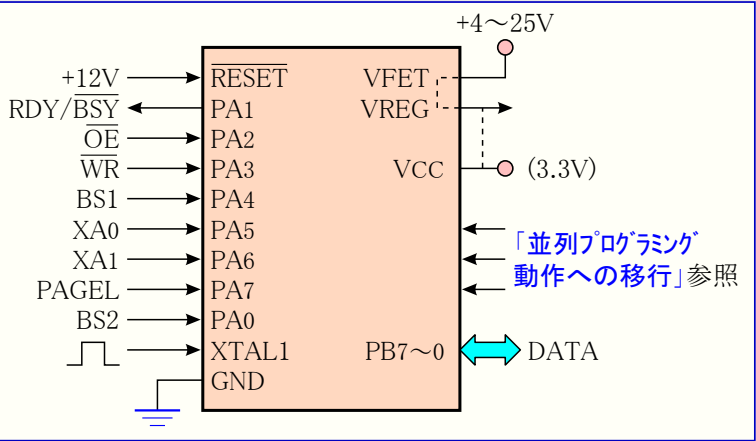


表28-7. 信号名とピン名の関係

信号名	ピン名	入出力	機能
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)
RDY/BSY	PA1	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	PA2	入力	出力許可(負論理)
WR	PA3	入力	書き込みパルス(負論理)
BS1	PA4	入力	上位/下位バイト選択1 (0:下位, 1:上位) (一般用)
XA0	PA5	入力	XTAL動作ビット0
XA1	PA6	入力	XTAL動作ビット1
PAGEL	PA7	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
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

表28-8. 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28-9. 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28-10. 指令バイトのビット符号化

指令バイト	指令の機能	指令バイト	指令の機能
\$80 (1000 0000)	チップ消去	\$08 (0000 1000)	識別バイト、校正バイト読み出し
\$40 (0100 0000)	ヒューズビット書き込み	\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$20 (0010 0000)	施錠ビット書き込み	\$02 (0000 0010)	フラッシュメモリ読み出し
\$10 (0001 0000)	フラッシュメモリ書き込み	\$03 (0000 0011)	EEPROM読み出し
\$11 (0001 0001)	EEPROM書き込み		

28.6.3. 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並列プログラミング動作にします。

- ① 27頁の「電源ONリセットと充電器接続」項で説明されるようにチップが開始されることを確実にしてください。
- ② RESETをLow(0)にし、XTAL1を少なくとも6回、交互に(High/Low)切り替えます。
- ③ 128頁の表28-8. で一覧されるProg_enableピンを全てLow(0)に設定し、最低100ns待ちます。
- ④ RESETに11.5～12.5Vを印加します。+12VがRESETに印加されてしまった後100ns以内のProg_enableピンのどんな動きも、デバイスのプログラミング動作への移行を失敗させます。
- ⑤ 新規指令送出前に少なくとも50μs間待ちます。

28.6.4. 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 WRに負パルスを与えます。これはチップ消去を開始します。RDY/BSYがLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/BSYがHigh(1)になるまで待ちます。

28.6.5. フラッシュ メモリ書き込み (次頁の図28-3.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(128頁の表28-5.参照)。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)を選択します。
- ③ DATAにアドレス下位バイト(\$00~\$FF)を設定します。
- ④ 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択します。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00~\$FF)を設定します。
- ④ 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択します。
- 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28-2.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)を選択します。
- ③ DATAにアドレス上位バイト(\$00~\$4F)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28-2. ページで構成されたフラッシュ メモリのアドレス指定

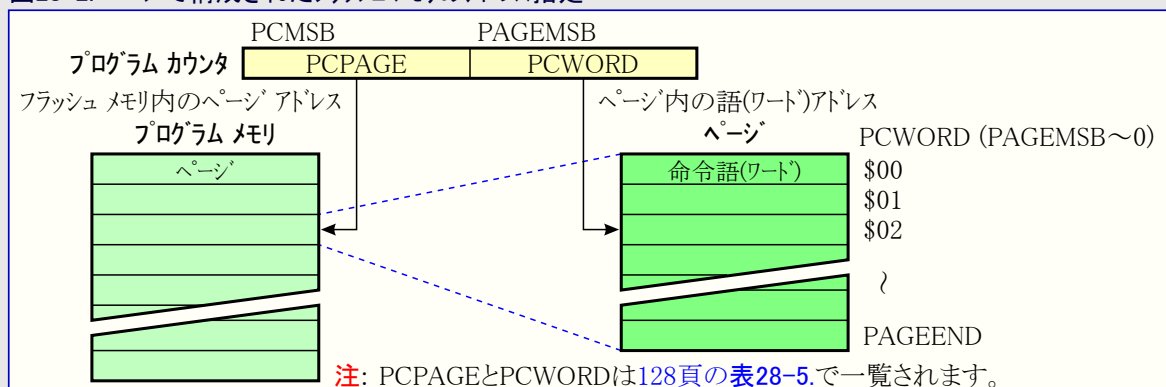
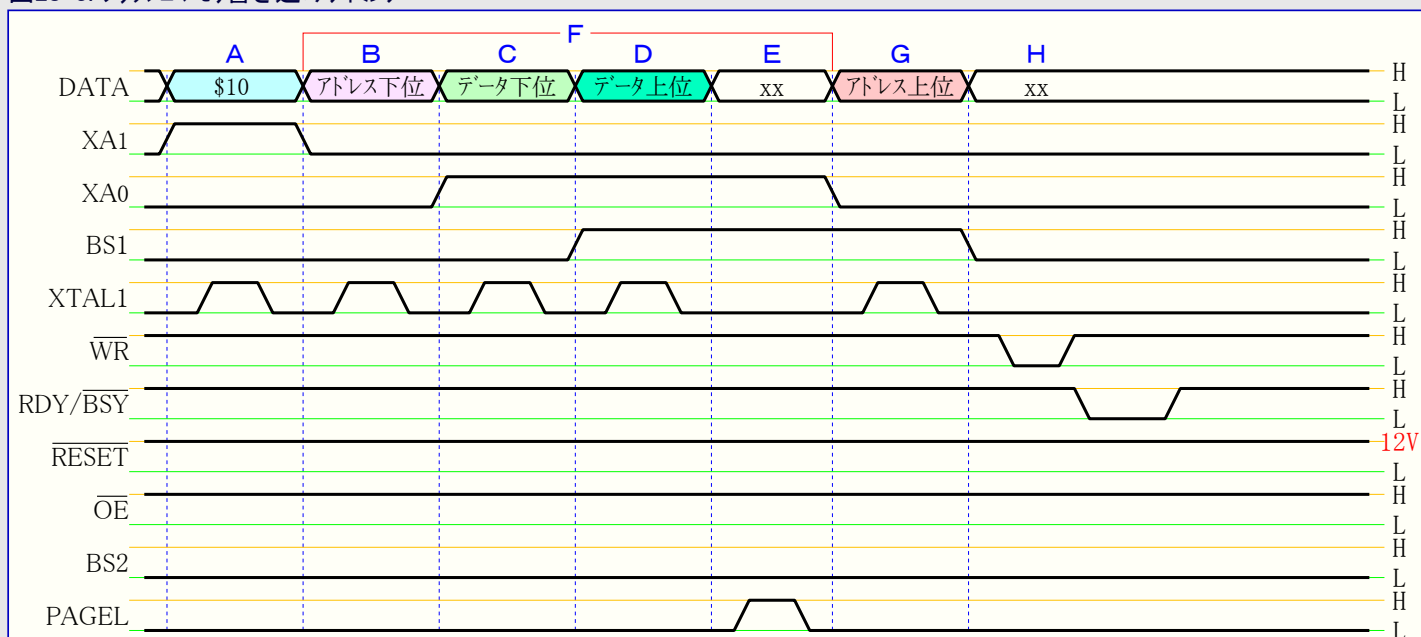


図28-3. フラッシュ メリ書き込みタイミング



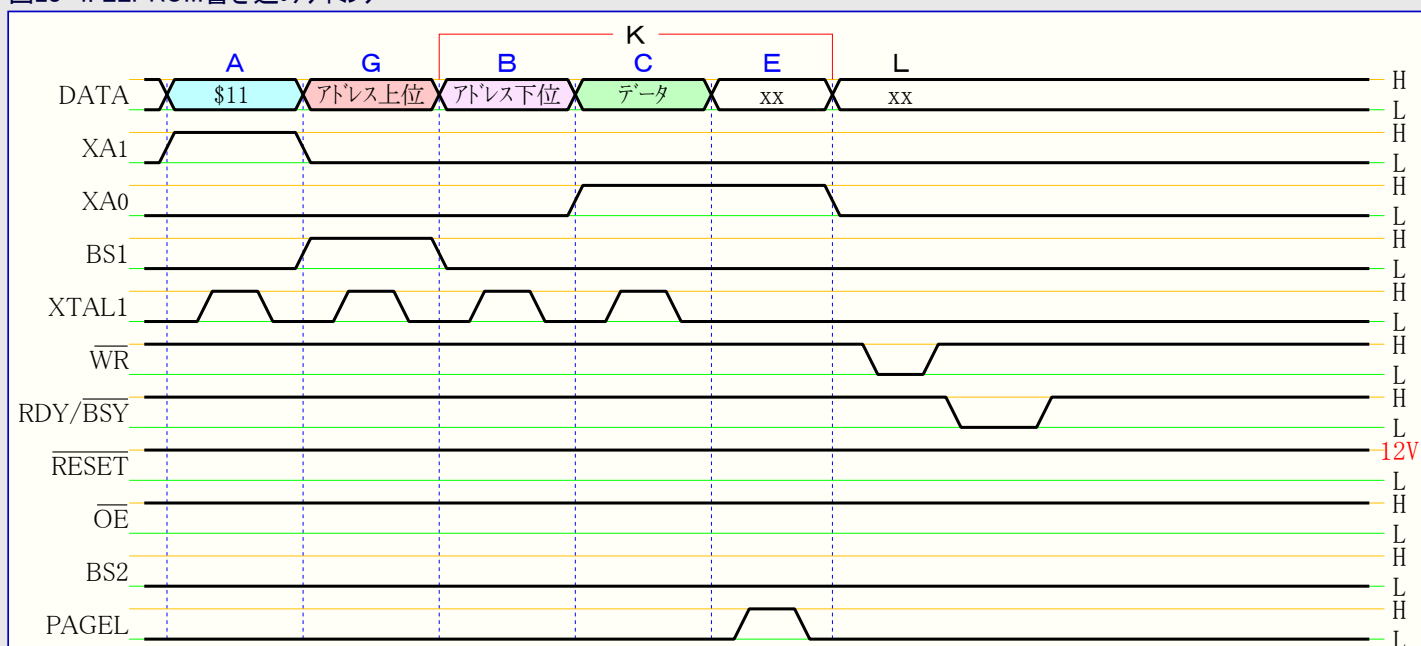
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

28.6.6. EEPROM書き込み

EEPROMはページで構成されます(128頁の表28-6参照)。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130頁の「フラッシュ メリの書き込み」を参照。図28-4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～\$01)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 $\overline{WR}$ に負パルスを与えます。これはEEPROMページ書き込みを開始します。RDY/ $\overline{BSY}$ がLow(0)になります。
 - ③ 次のページを書く前に、RDY/ $\overline{BSY}$ がHigh(1)になるまで待ちます。

図28-4. EEPROM書き込みタイミング



28.6.7. フラッシュ メモリ読み出し

フラッシュメモリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については130頁の「フラッシュメモリの書き込み」を参照)

1. フラッシュ メリ読み出し指令\$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~\$4F)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。

28.6.8. 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130頁の「フラッシュメモリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00～\$01)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。

28.6.9. ヒューズビット書き込み (訳注:原書での上位/下位ヒューズ項を1つに纏めました。)

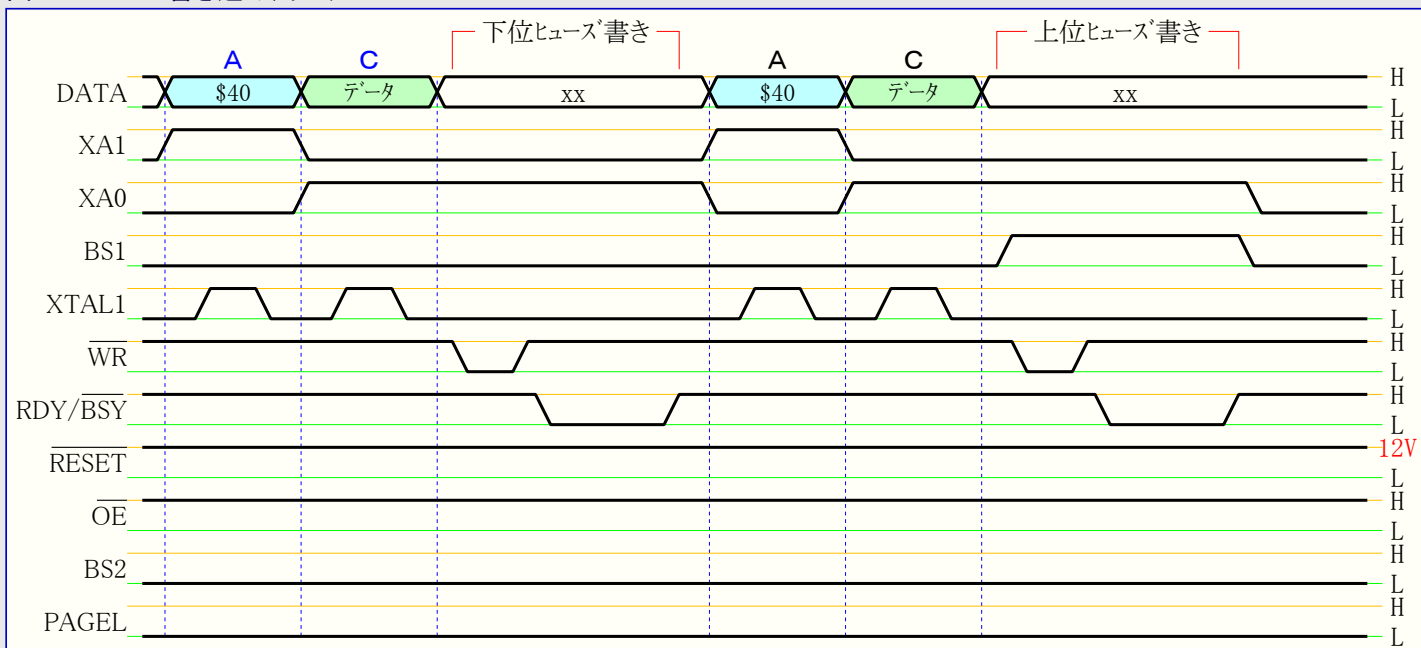
各ヒューズ・バットの書き込み方法は次のとおりです。(指令とデータ設定の詳細については130頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズ・ビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュ メリ書き込み」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. $\overline{WR}$ に負パルスを与え、RDY/ $\overline{BSY}$ がHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選択します。

表A. ヒューズ`バイト対応BS1,BS2設定

ヒューズ バイト	BS1	BS2
上位バイト	High(1)	Low(0)
下位バイト	Low(0)	Low(0)

図28-5. ヒューズ書き込みタイミング



28.6.10. 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130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。

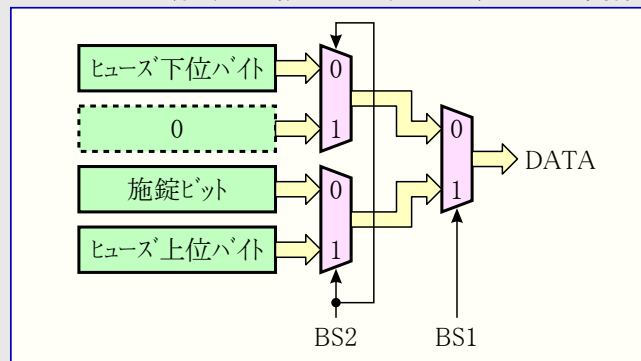
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

28.6.11. 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130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)
5. $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

図28-6. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



28.6.12. 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130頁の「フラッシュメモリの書き込み」を参照)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。選択した識票バイトが直ぐにDATAで読めます。
4. $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

28.6.13. 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130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。

(訳注) 上記校正バイト読み出し法は校正バイトが複数のため不適切です。2.で任意のアドレス指定を行なうことが予測されます。

28.6.14. 並列プログラミング特性

図28-7. 並列プログラミング タイミング (一般的な必要条件)

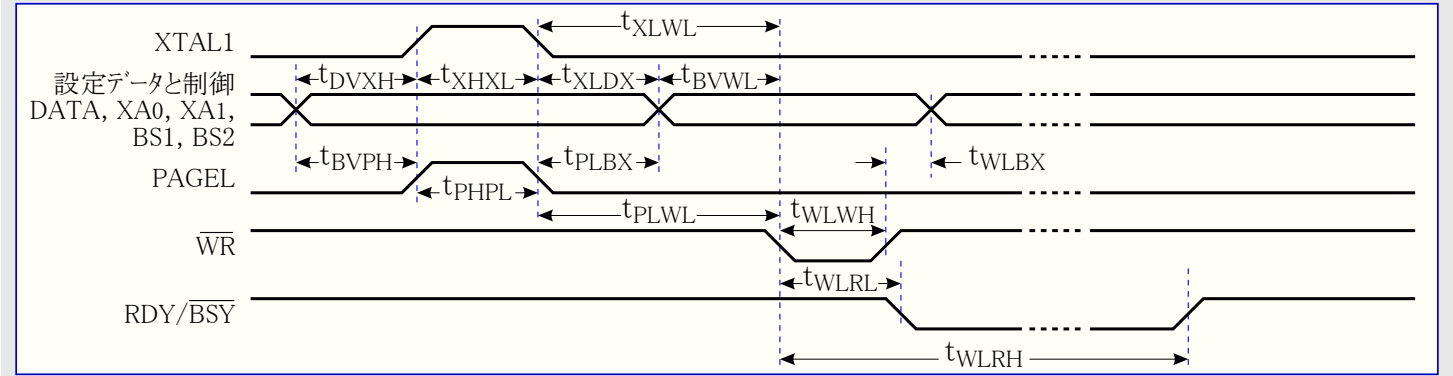
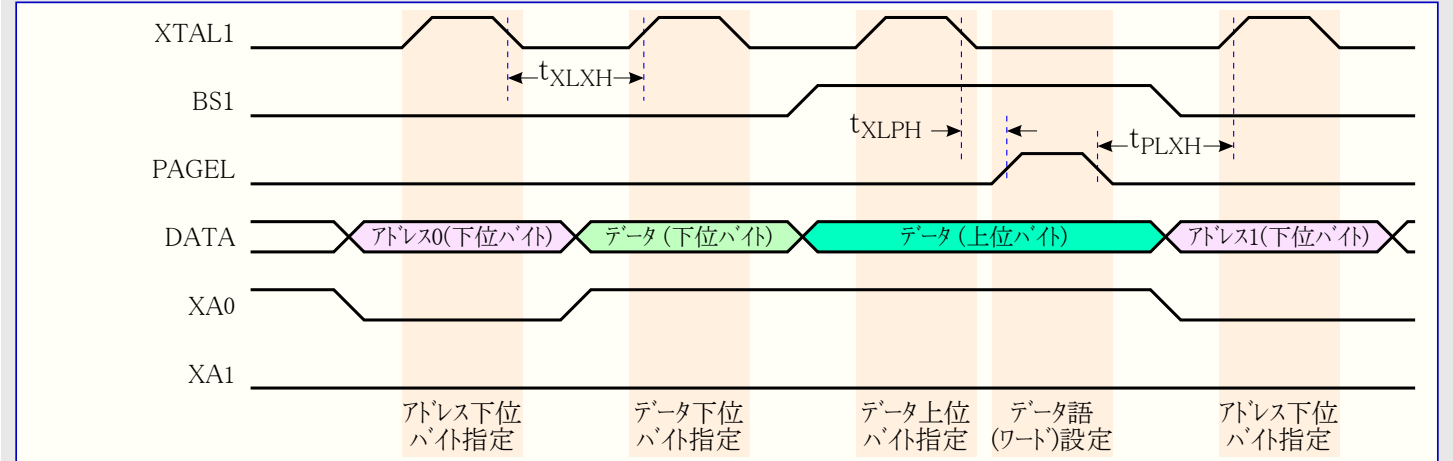
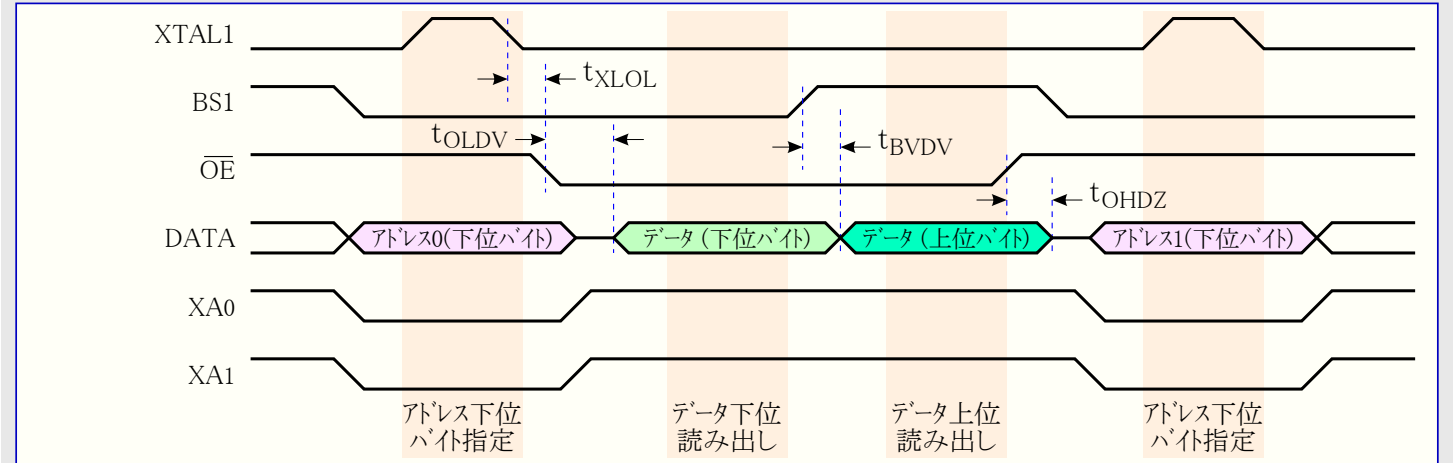


図28-8. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図28-7.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28-9. 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28-7.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28-11. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
V _{PP}	プログラミング許可電圧	11.5		12.5	V
I _{PP}	プログラミング許可電流			250	μA
t _{DVXH}	XTAL1 ↑ に対するデータと制御の準備時間	67			
t _{XLXH}	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
t _{XHXL}	XTAL1 Highパルス幅	150			
t _{XLDX}	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
t _{XLWL}	XTAL1パルス ↓ 後の $\overline{WR}$ ↓ 待機時間	0			
t _{XLPH}	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
t _{PLXH}	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			ns
t _{BVPH}	PAGELパルス ↑ に対するBS1準備時間	67			
t _{PHPL}	PAGEL Highパルス幅	150			
t _{PLBX}	PAGELパルス ↓ 後のBS1保持時間	67			
t _{WL BX}	$\overline{WR}$ パルス ↓ 後のBS1,BS2保持時間	67			
t _{PLWL}	PAGELパルス ↓ 後の $\overline{WR}$ パルス ↓ 待機時間	67			
t _{BVWL}	$\overline{WR}$ パルス ↓ に対するBS1準備時間	67			
t _{WLWH}	$\overline{WR}$ Lowパルス幅	150			
t _{WLRL}	$\overline{WR}$ パルス ↓ 後のRDY/ $\overline{BSY}$ ↓ 遅延時間	0		1	μs
t _{WLRH}	書き込み時間 ($\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注1)	3.7		4.5	ms
t _{WLRH_CE}	チップ消去時間 ($\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注2)	7.5		9	
t _{XLCL}	XTAL1パルス ↓ 後の $\overline{OE}$ ↓ 待機時間	0			
t _{BVDV}	BS1有効からのDATA遅延時間	0		250	ns
t _{OLDV}	$\overline{OE}$ ↓ 後のDATA出力遅延時間			250	
t _{OHDZ}	$\overline{OE}$ ↑ 後のDATA Hi-Z遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

28.8. 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(MCU_CR)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(訳注:共通性から本行追加)。

この資料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。

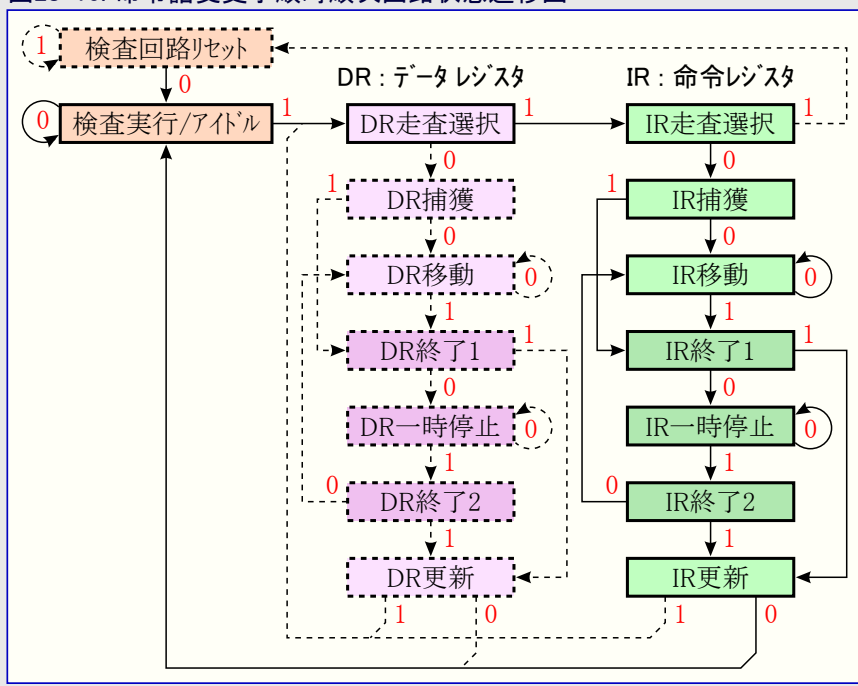
28.8.1. プログラミング特殊JTAG命令

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間の経路として選択さ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28-10.で示されます。

図28-10. 命令語変更手順時順次回路状態遷移図



28.8.2. 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)として選択されます。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。

28.8.3. 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)として選択されます。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。

28.8.4. 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)として選択されます。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28-12.をご覧ください)。

28.8.5. 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)として選択されます。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。11TCK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。

28.8.6. 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)として選択されます。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 : 選択した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。

28.8.7. データレジスタ

データレジスタ(DR)は136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)によって選択されます。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
- プログラミング命令(Programming Command)レジスタ
- フラッシュバイトデータ(Flash Data Byte)レジスタ

28.8.8. リセット (Reset) レジスタ

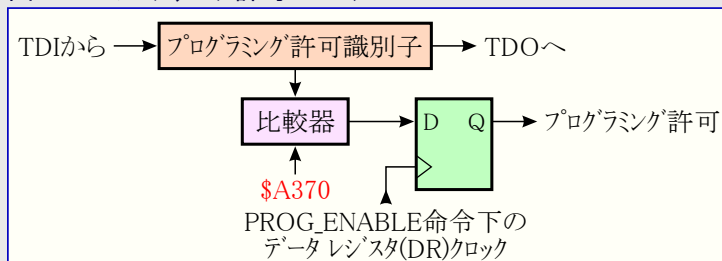
リセットレジスタはプログラミング中、デバイスをリセットするのに使われる検査データレジスタです。プログラミング動作への移行に先立ってデバイスをリセットするのに必要とされます。

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、デバイスは起動時間選択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。26頁の図9-1.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。

28.8.9. 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28-11. プログラミング許可 レジスタ



28.8.10. 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28-12.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28-13.で図解されます。

図28-12. プログラミング命令 レジスタ

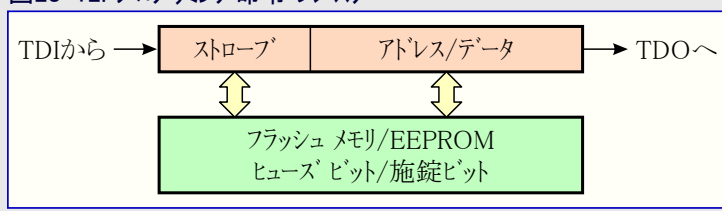


表28-12. JTAGプログラミング命令一式 (手順)

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	
	1 ②	チップ消去完了検査	011 0011 1000 0000	xxx xxSx xxxx xxxx	(注2)
フラッシュメモリ書き込み	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9)
	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	
	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9)
	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 xxxx xxxx	
			011 0110 0000 0000	xxx xxxx RRRR RRRR	下位バイト
			011 0111 0000 0000	xxx xxxx RRRR RRRR	上位バイト
EEPROM書き込み	4 ①	EEPROM書き込み移行	010 0011 0001 0001	xxx xxxx xxxx xxxx	
	4 ②	アドレス上位バイト設定	000 0111 HHHH HHHH	xxx xxxx xxxx xxxx	(注9)
	4 ③	アドレス下位バイト設定	000 0011 LLLL LLLL	xxx xxxx xxxx xxxx	
	4 ④	データバイト設定	001 0011 WWWW WWWW	xxx xxxx xxxx xxxx	
	4 ⑤	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 ⑥	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	
	4 ⑦	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9)
	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 xxxx xxxx	
			011 0010 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx RRRR RRRR	

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)

次頁へ続く

表28-12 (続き). JTAGプログラミング命令一式

命令		TDI入力	TDO出力	備考	
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 (注6)	001 0011 WWW WWW	xxx xxxx xxxx xxxx	(注3)
	6 ③	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 ④	上位ヒューズ書き込み完了検査	011 0111 0000 0000	xxx xxS xxxx xxxx	(注2)
	6 ⑤	データ下位バイト設定 (注7)	001 0011 WWW WWW	xxx xxxx xxxx xxxx	(注3)
	6 ⑥	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 ⑦	下位ヒューズ書き込み完了検査	011 0011 0000 0000	xxx xxS xxxx xxxx	(注2)
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 (注8)	001 0011 11WW WWW	xxx xxxx xxxx xxxx	(注4)
	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 xxS 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	上位側ヒューズ読み出し (注6)	011 1110 0000 0000	xxx xxxx xxxx xxxx	(注5)
			011 1111 0000 0000	xxx xxxx RRRR RRRR	
	8 ③	下位側ヒューズ読み出し (注7)	011 0010 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx RRRR RRRR	
	8 ④	施錠ビット読み出し (注8)	011 0110 0000 0000	xxx xxxx xxxx xxxx	(注5)
			011 0111 0000 0000	xxx xxxx xxRR RRRR	
	8 ⑤	ヒューズ/施錠ビット読み出し	011 1110 0000 0000	xxx xxxx xxxx xxxx	上位ヒューズ 下位ヒューズ 施錠ビット (注5)
			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	
校正バイト	10 ①	校正バイト読み出し移行	011 0011 0000 0000	xxx xxxx RRRR RRRR	
			011 0011 0000 0000	xxx xxxx RRRR RRRR	
	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	
	11 ①	無操作設定	011 0111 0000 0000	xxx xxxx RRRR RRRR	
			010 0011 0000 0000	xxx xxxx xxxx xxxx	
			011 0011 0000 0000	xxx xxxx xxxx xxxx	

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。

注6: ヒューズ上位バイトのビット配置については127頁の表28-3.で一覧されます。

注7: ヒューズ下位バイトのビット配置については127頁の表28-4.で一覧されます。

注8: 施錠ビットバイトのビット配置については126頁の表28-1.で一覧されます。

注9: PCMSB(表28-5.参照)とEEAMSB(表28-6.参照)を越えるアドレスビットは無効です。

28.8.11. 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 メモリからのストローブ信号を判断し、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写し、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_PAGELoad**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、選択したフラッシュ 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28-13. データ語変更/読み出し手順時順次回路状態遷移図

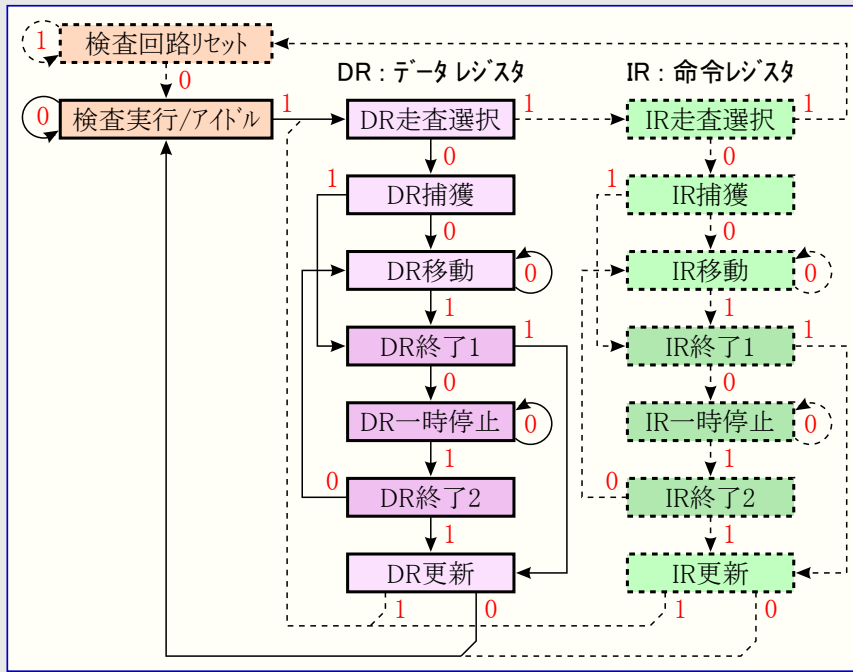
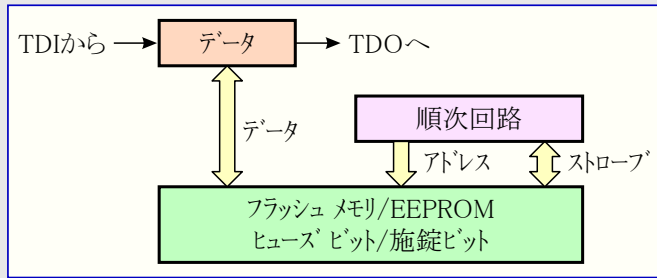


図28-14. フラッシュ バイト データ レジスタ



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十分なクロック 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。

28.8.12. 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28-12を参照してください。

28.8.13. 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。

28.8.14. 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(1①参照)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。

28.8.15. 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**(135頁の表28-11.参照)時間待ちます。

28.8.16. 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。140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**(135頁の表28-11.参照)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**(128頁の表28-5.参照)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**(135頁の表28-11.参照)時間待ち機します。
9. 全データ書き込みまで3.~8.を繰り返します。

28.8.17. 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**(128頁の表28-5.参照)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。

28.8.18. 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。140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**(135頁の表28-11.参照)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。

28.8.19. 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. 全データ読み出しまで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。

28.8.20. ヒューズビットの書き込み

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. 上位データ設定命令(6②参照)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)
4. 上位ヒューズ書き込み命令(6③参照)を使い、上位側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**(135頁の表28-11.参照)時間待ちます。
6. 下位データ設定命令(6⑤参照)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)
7. 下位ヒューズ書き込み命令(6⑥参照)を使い、下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**(135頁の表28-11.参照)時間待ちます。

28.8.21. 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**(135頁の表28-11.参照)時間待ちます。

28.8.22. ヒューズ/施錠ビットの読み出し

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(8⑤参照)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(8②参照)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(8③参照)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(8④参照)を使います。

28.8.23. 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。

28.8.24. 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。

(訳注) 上記校正バイト読み出し法は校正バイトが複数のため不適切です。3.で任意のアドレス指定を行なうことが予測されます。

29. 動作回路

図29-1. 動作回路構成図

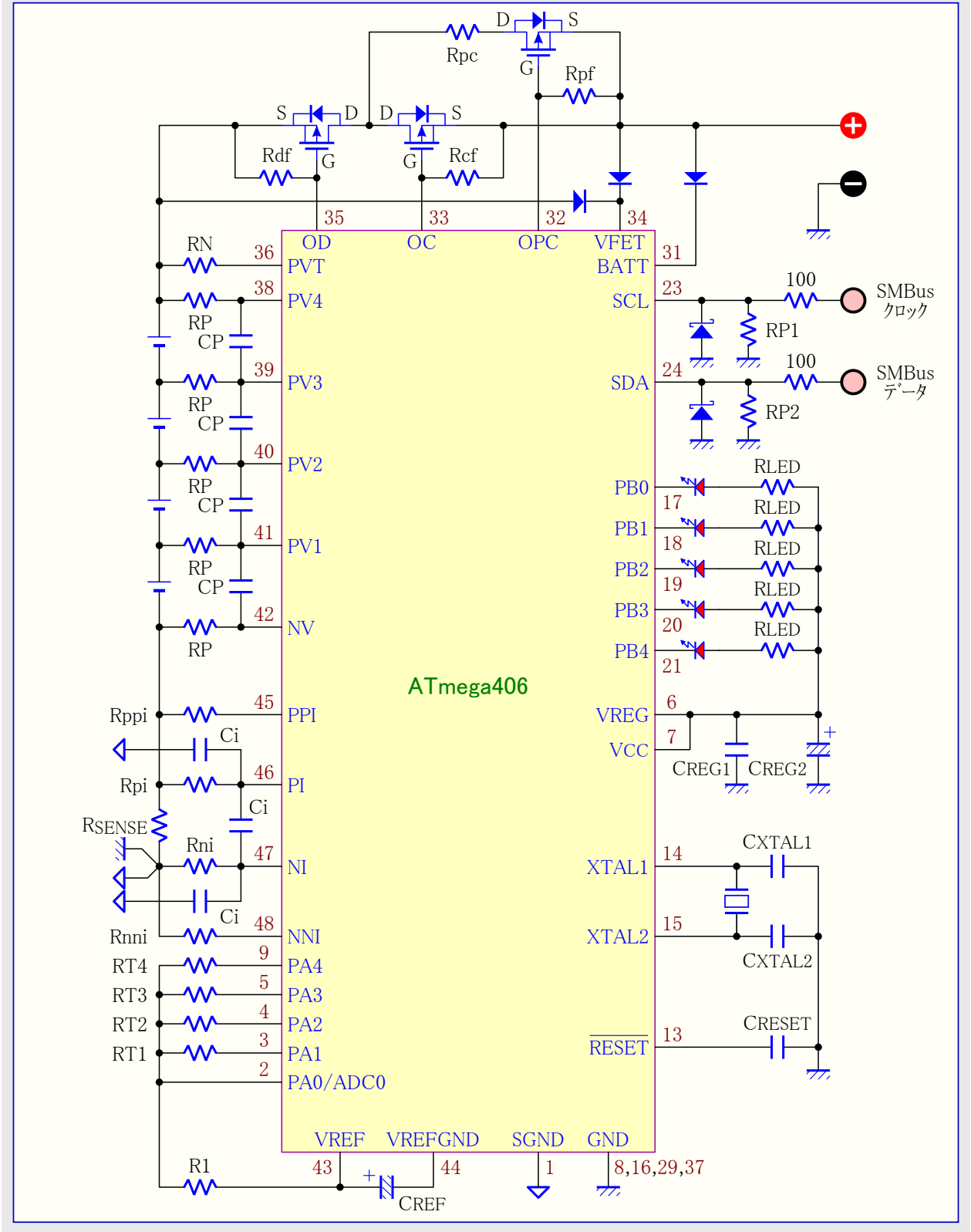


表29-1. 外部部品推奨値

シンボル	用途	項目	最小	代表	最大	単位
R1	サーミスタ用プルアップ抵抗器	R		10		k Ω
RT1,RT2,RT3,RT4	NTCサーミスタ	R/25°C		10		
		β 定数	3000		4000	K
RS	V-ADC入力として使う時の供給元インピーダンス	R	0	3	7	k Ω
		RSのために最悪利得誤差	0	1	2	%
Rnni,Rppi	電流保護低域通過濾波抵抗器	R		1		k Ω
Rni,Rpi	電流監視低域通過濾波抵抗器	R	10	100	500	Ω
Ci	電流監視低域通過濾波コンデンサ	C	0.01	0.1	0.4	μ F
(Rpi-Rnl) $\times$ Ci	電流監視低域通過濾波器時定数	τ		10	20	μ s
RSENSE (注1)	電流監視抵抗器	R		5		m Ω
RP	セル入力低域通過濾波抵抗器	R	10	500	1000	Ω
CP	セル入力低域通過濾波コンデンサ	C	0.01	0.1	0.5	μ F
RP $\times$ CP	セル入力低域通過濾波器時定数	τ	6.5	25	100	μ s
RN	プルアップ抵抗器	R	0	10	TBD	Ω
Rcf,Rdf,Rpf	プルアップ抵抗器	R		1		M Ω
Rpc (注2)	予備充電抵抗器	R		1		k Ω
CREF	VREFデカップ コンデンサ	C	1	1	22	
CREG1	VREG充電格納	C		0.1		μ F
CREG2				1		
RP1,RP2	TWIプルダウン抵抗器	R		2.2		M Ω
CRESET	リセット積分器コンデンサ	C		0.1		μ F

注1: 電流監視抵抗器は応用に関する電流の流れに対して調節されるべきです。

注2: 予備充電抵抗器は応用に関する予備充電電流の流れに対して調節されるべきです。

30. 電気的特性

30.1. 絶対最大定格 (警告)

動作温度	-30℃ ~ +85℃
保存温度	-65℃ ~ +150℃
PA0~7,PB0~7,PD0~1,VCC,PI,PPI, NI,NNI,XTAL1,XTAL2ピン許容電圧	-0.5V ~ VREG+0.5V
SCL,SDA,NV,PV1,RESETピン許容電圧	-0.5V ~ +6.0V
PVT,VFETピン許容電圧	-0.5V ~ +35.0V
PC0,OPC,OC,BATTピン許容電圧	-0.5V ~ VFET+0.5V
OD,PV2~4ピン許容電圧	-0.5V ~ PVT+0.5V
最大動作電圧	25.0V

(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。

30.2. DC特性

TA=-30℃～85℃, VCC=3.3V							
項目		条件	最小	代表	最大	単位	
消費電流	消費電流	1MHz活動動作		1.2		mA	
		1MHzアイドル動作		270			
		A/D変換雑音削減動作		220			
		パワーセーブ動作		35		μA	
		パワーダウン動作		20			
		パワーオフ動作		1.5			
電圧調整器 (注1)	調整出力電圧 (注2)	IOUT=5mA	3.25	3.3	3.35	V	
	温度変動 (注2)	IOUT=5mA, TA=0～60℃		±5	±15		
		IOUT=5mA, TA=-30～85℃		±20	±70		
	負荷変動	IOUT=0.1～5mA		±20	±60	mV	
	入力変動	VFET=4～25mA, IOUT=5mA		±2	±10		
基準電圧 (VREF)	基準電圧			1.1		V	
	基準電圧精度	校正後の校正温度にて		±0.1	±0.2	%	
	温度変化 (注3)	TA=0～60℃		80		ppm/℃	
電圧 A/D変換器	基準電圧			1.100		V	
	変換時間			519		μs	
	有効分解能			12		ビット	
	非縮尺(1.0倍)入力1LSB電位差			269		μV	
	縮尺(0.2倍)入力1LSB電位差			1.34		mV	
	積分非直線性誤差			±1	±3	LSB	
	入力電圧範囲	ADC0,ADC1,ADC2,ADC3,VTEMP	0		1		
		ADC4	0		5		
		電池セル1	2		5		V
		電池セル2, PV1≥2V	0		5		
		電池セル3, PV1≥2V	0		5		
		電池セル4, PV1≥2V	0		5		
	オフセット誤差			1.6		mV	
	電池セル入力利得誤差 (注5)	TA=0～60℃		±0.5	±1	%	
クーロン カウンタ A/D変換器	基準電圧			±220		mA	
	変換時間と分解能	53.7μV分解能		3.9		ms	
		1.68μV分解能	125		1000		
	積分非直線性誤差	1000ms変換時間			±4	LSB	
	オフセット (注6)	未補償		±50	±200	μV	
	オフセット変動 (注4)	TA=0～60℃		±1	±15		
	利得誤差				±1	%	

次ページへ続く

項目		条件	最小	代表	最大	単位
温度感知器	絶対温度対VPTAT電圧係数			0.6		mV/K
	絶対精度 (注3)				±4	K
	ON時ゲート(G)/ソース(S)間電圧差		11		15	V
FET駆動部	ON⇒OFF時OC,OD上昇時間	10⇒90%,CL=0.01μF		10	50	μs
	OFF⇒ON時OC,OD下降時間	VGS=0⇒-5V,CL=0.01μF			100	
	ON⇒OFF時OPC上昇時間	10⇒90%,CL=0.001μF		100	500	
	OFF⇒ON時OPC下降時間	VGS=0⇒-5V,CL=0.001μF		100	500	
低速RC発振器	周波数	TA=-30～85℃		165		kHz
	温度変動			5		%
超低電力RC発振器	周波数	TA=-30～85℃		124		kHz
	温度変動			8		%

注1: 電圧調整器能力は1μF平滑コンデンサが基準です。

注2: VREF第2段階温度校正後。既定によって第1段階校正はAtmelの工場試験に於いて85℃で実行されます。第2段階校正は室温での標準試験手順で容易に実装できます。

注3: この値は製造に於いて検査されません。

注4: ソフトウェアでのシステム オフセット補償後の値です。

注5: ソフトウェア利得誤差補償後の値です。

注6: この値はシステム レベルで測定され、ソフトウェア オフセット補償用にEEPROM内に格納されるべきです。

30.3. 汎用入出力線特性

30.3.1. 低電圧ポート

表30-1. TA=-30℃～85℃, VCC=3.3V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V _{IL}	Lowレベル入力電圧		-0.5		0.3VCC (注2)	V
V _{IH}	Highレベル入力電圧		0.6VCC (注3)		VCC+0.5	
V _{OL}	Lowレベル出力電圧 (注4)	IOL=5mA			0.5	
V _{OH}	Highレベル出力電圧 (注5)	IOH=-2mA	2.3			
I _{IL}	L側入力漏れ電流	ピン=L(絶対値)			1	μA
I _{IH}	H側入力漏れ電流	ピン=H(絶対値)			1	

注1: PC0を除いて全てに適用できます。

注2: 最大値はピンがLowとして読むことが保証される最高電圧を意味します。

注3: 最小値はピンがHighとして読むことが保証される最低電圧を意味します。

注4: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3.3Vで5mA)よりも多くの吸い込み電流を流すことができますが、次の条件が厳守されなければなりません。

1. 全ポートのIOLの合計が20mA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。検査条件で示されるよりも大きな吸い込み電流を流すことは保証されません。

注5: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3.3Vで2mA)よりも多くの吐き出し電流を流すことができますが、次の条件が厳守されなければなりません。

1. 全ポートのIOHの合計が2mA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。検査条件で示される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。

30.3.2. 高電圧ポート

表30-2. TA=-30℃～85℃, VCC=3.3V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V _{OL}	Lowレベル出力電圧 (注1)				0.5	V
t _r	上昇時間 (注3)				300	ns
t _{of}	V _{IHmin} ⇒V _{ILmax} 上昇時間 (注4)	Cb<400pF (注5)			200	

注1: 検査なしで特性付けされたパラメータ。

注2: Cbは1本のバス線のpFでの容量です。

30.4. 2線直列インターフェース特性

表30-3は2線直列バスに接続される装置についての必要条件を記述します。ATmega406の2線直列インターフェースは記載された条件下でこれらの必要条件を越えるかまたは合致します。

図30-1. 2線直列バス タイミング

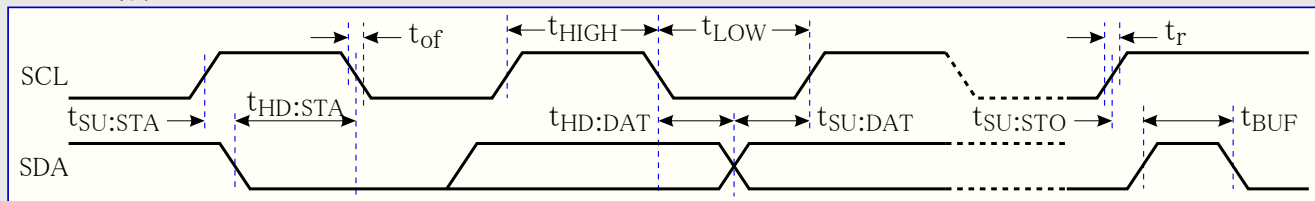


表30-3. 2線直列インターフェース必要条件

シンボル	項目	条件	最小	最大	単位
V_{IL}	Lowレベル入力電圧		-0.5	0.8	V
V_{IH}	Highレベル入力電圧		2.1	5.5	
V_{OL} ①	Lowレベル出力電圧	$I_{OL}=350\mu A$	0	0.4	
t_r ①	出力上昇時間($V_{ILmin} \rightarrow V_{IHmax}$)			300	
t_{of} ①	出力下降時間($V_{IHmin} \rightarrow V_{ILmax}$)	$C_b < 400pF$ ②		250	ns
t_{SP} ①	入力パルス最小幅(尖頭消去濾波)		0	50	
I_i	入力電流(ピン単位)	$0.1V_{BUS} < V_i < 0.9V_{BUS}$	-5	5	μA
C_i ①	ピン入力容量			10	pF
f_{SCL}	SCLクロック周波数 ③ ④	$f_{CK} > \max(16f_{SCL}, 450kHz)$	0	100	kHz
R_p	プルアップ抵抗値	$f_{SCL} \leq 100kHz$	$\frac{(V_{CC}-0.4V)}{350\mu A}$	$\frac{(V_{CC}-0.4V)}{100\mu A}$	Ω
$t_{HD:STA}$	(再送)開始条件保持時間	$f_{SCL} \leq 100kHz$	4.0		μs
t_{LOW}	SCLクロックLowレベル時間	$f_{SCL} \leq 100kHz$	4.7		
t_{HIGH}	SCLクロックHighレベル時間	$f_{SCL} \leq 100kHz$	4.0		
$t_{SU:STA}$	再送開始条件準備時間	$f_{SCL} \leq 100kHz$	4.7		ns
$t_{HD:DAT}$	データ保持時間	$f_{SCL} \leq 100kHz$	0.3	3.45	
$t_{SU:DAT}$	データ準備時間	$f_{SCL} \leq 100kHz$	250		
$t_{SU:STO}$	停止条件準備時間	$f_{SCL} \leq 100kHz$	4.0		μs
t_{BUF}	停止条件→開始条件間バス開放時間	$f_{SCL} \leq 100kHz$	4.7		

① ATmega406で、この項目は特性が記載されていますが、100%検査はされていません。

② C_b は1つのバス信号線の容量(pF)です。

③ f_{CK} はCPU(システム)クロック周波数です。

④ この必要条件はATmega406の全ての2線直列インターフェース動作に適用します。2線直列バスに接続した他の装置は一般的な f_{SCL} 必要条件に従うことだけを必要とします。

30.5. リセット特性

表30-4. 低損失電圧調整器電源投入用特性

	シンボル	項目	最小	代表	最大	単位
充電器有り	V _{ROT}	電圧調整器電源ON閾値電圧	3.0		4.0	V
	V _{CHT}	充電電圧閾値		1.0		
充電器無し	V _{ROT}	電圧調整器電源ON閾値電圧	3.0		4.0	
	V _{PV1T}	電池セル1での閾値電圧		2.0		

注: ・ 電源ONリセットは充電器が接続され、電圧調整器が安定動作条件の時に起こります。
・ 値は特性付けが基準です。

表30-5. 電源ONリセット特性

シンボル	項目	条件	最小	代表	最大	単位
V _{COT}	充電器ON閾値電圧	電圧調整器作動	6	7	8	V

注: 内蔵電圧調整器がONでなければなりません。

表30-6. 外部リセット特性

シンボル	項目	条件	最小	代表	最大	単位
V _{RST}	RESETピン閾値電圧	V _{REG} =3.3V	0.66		2.8	V
t _{RST}	RESETピン最小リセットパルス幅			900		ns

注: 内蔵電圧調整器がONでなければなりません。

30.6. 周辺機能部供給電流

表30-7.は電力削減レジスタ(PRR0)で制御する全てのI/O部に関して活動動作消費電流(I_{CC})とアイドル動作消費電流(I_{CC})に対して追加消費電流を示します。詳細については24頁の「PRR0 – 電力削減レジスタ」をご覧ください。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対する追加消費電流の計算に使えます。

表30-7. 各部追加消費電流

PRR内ビット	1MHz,3.3Vでの消費電流 (μA)	活動動作での比率 (%)	アイドル動作での比率 (%)
PRTWI	68.0	5.6	25.2
PRTIM1	4.5	0.4	1.7
PRTIM0	6.0	0.5	2.2
PRVADC	5.0	4.2	1.9

例1: VCC=3.3V, f=1MHzでタイマ/カウンタ1、電圧A/D変換器、電池保護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30-7.のアイドル動作列からタイマ/カウンタ1が1.7%、電圧A/D変換器が1.9%、TWI部が25.2%追加する必要があります。145頁の「DC特性」を読み、VCC=3V, f=1MHzでのアイドル動作消費電流が約0.27mAである事を得ます。タイマ/カウンタ1、電圧A/D変換器、TWI許可のアイドル動作での総消費電流を得ます。

総消費電流=0.27mA×(1+0.017+0.019+0.252)≒0.35mA

31. 代表特性 – 暫定

以下の図は少数のマイクロ コントローラだけで検査されています。これらの図は製造中に検査されず、説明目的用に追加されています。

31.1. ピンプルアップ

図31-1. I/Oピンプルアップ抵抗電流 対 入力電圧 ($V_{CC}=3.3V$)

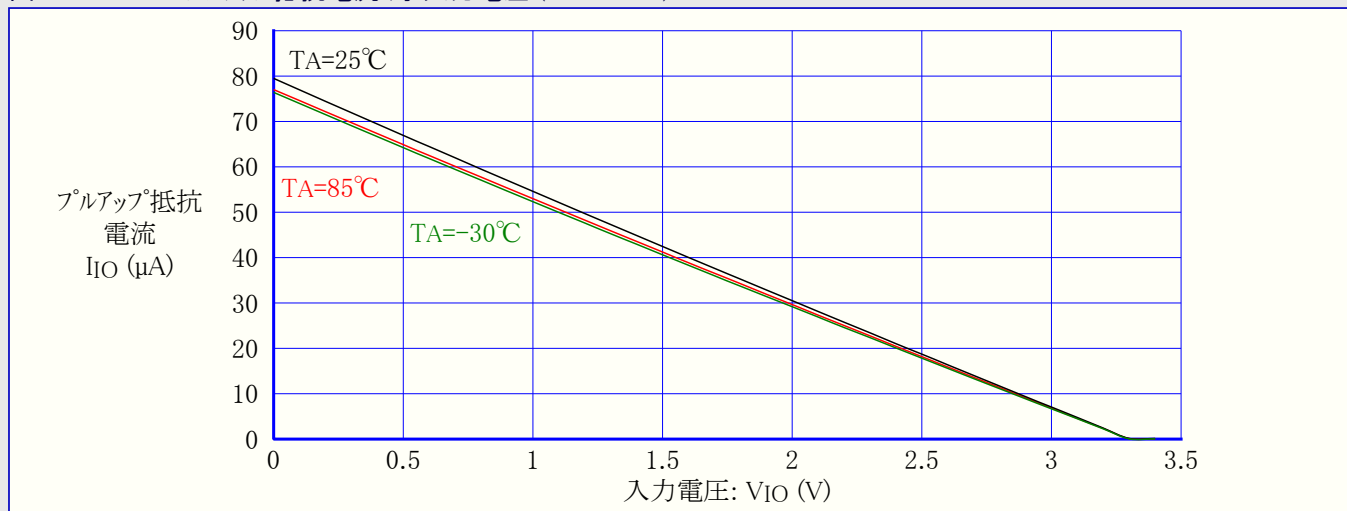
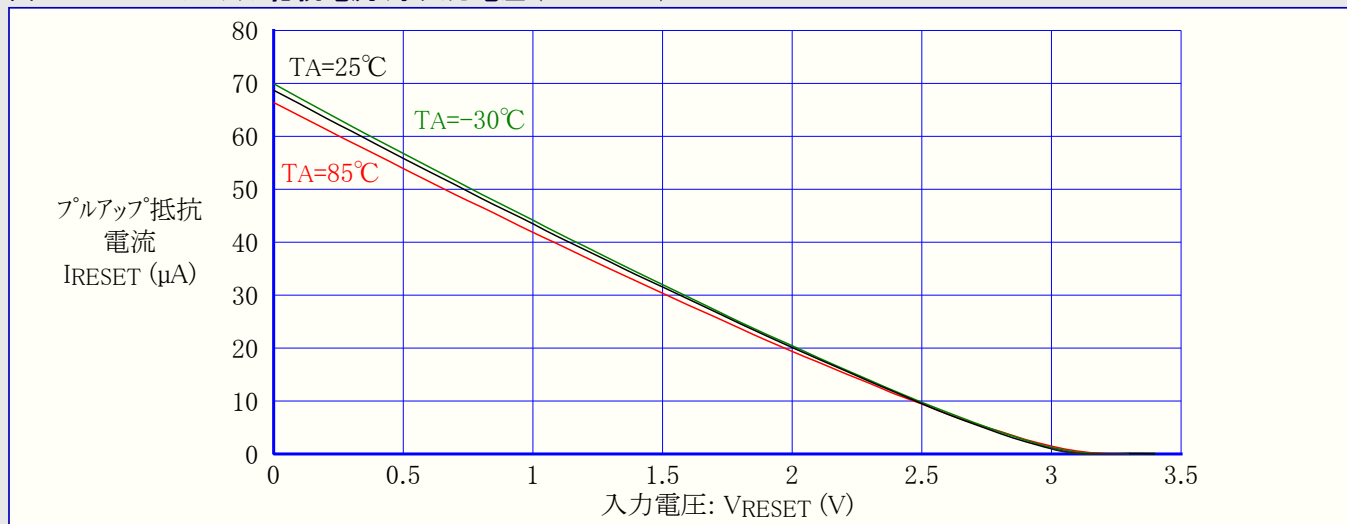


図31-2. RESETプルアップ抵抗電流 対 入力電圧 ($V_{CC}=3.3V$)



31.2. ピン駆動能力

図31-3. I/Oピン出力電圧 対 吸い込み電流 ($V_{CC}=3.3V$)

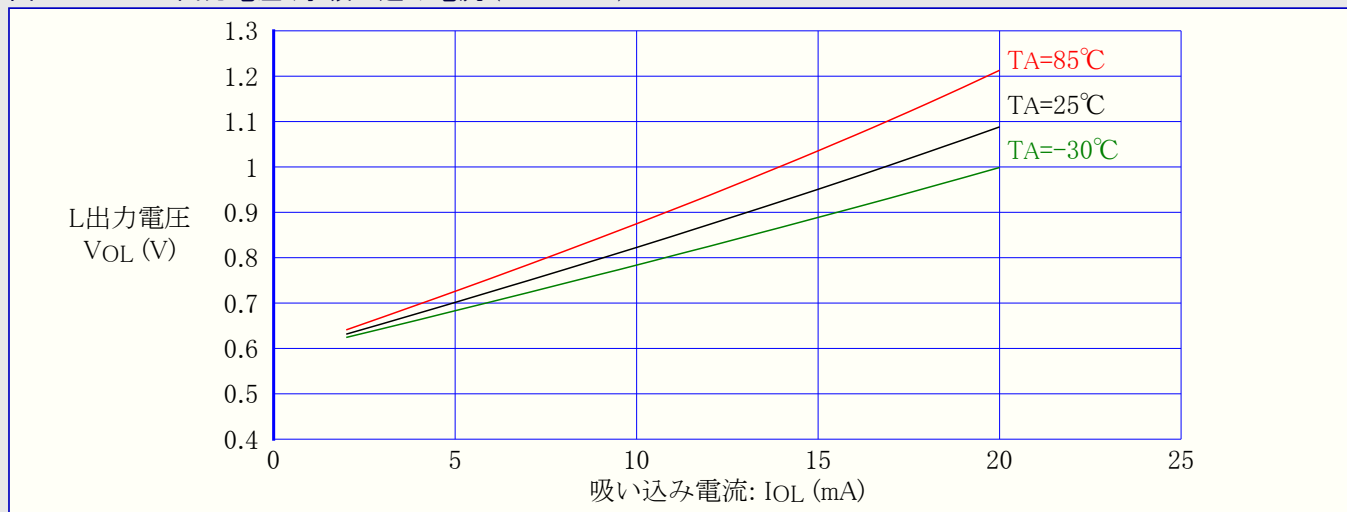
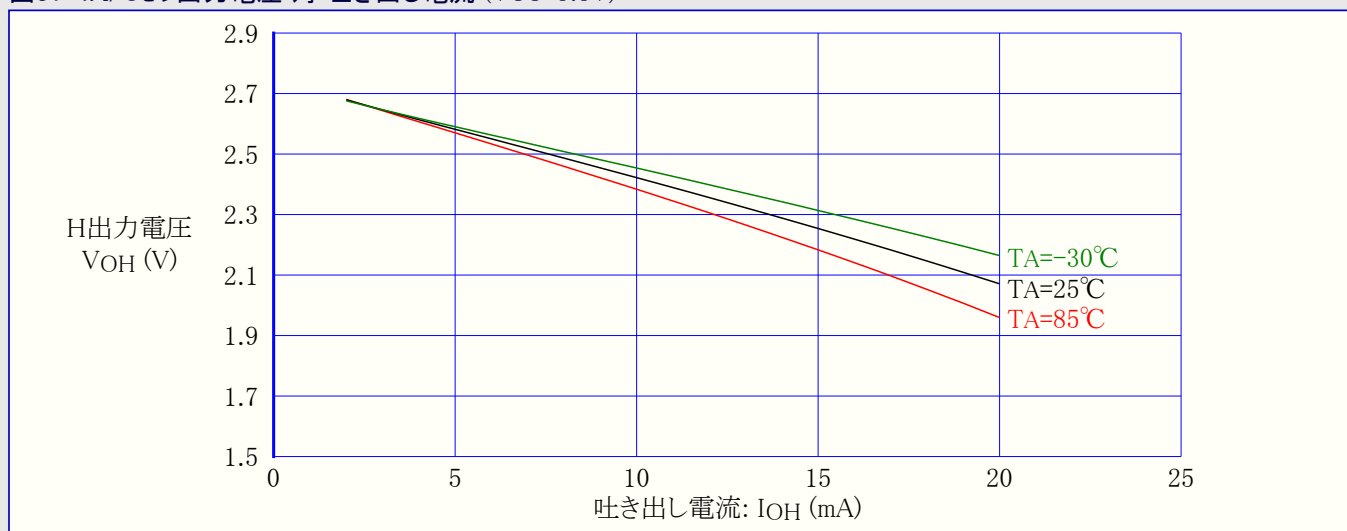


図31-4. I/Oピン出力電圧 対 吐き出し電流 (VCC=3.3V)



31.3. 内部発振器周波数

図31-5. ウォッチバック発振器周波数 対 動作温度 (VCC=3.3V)

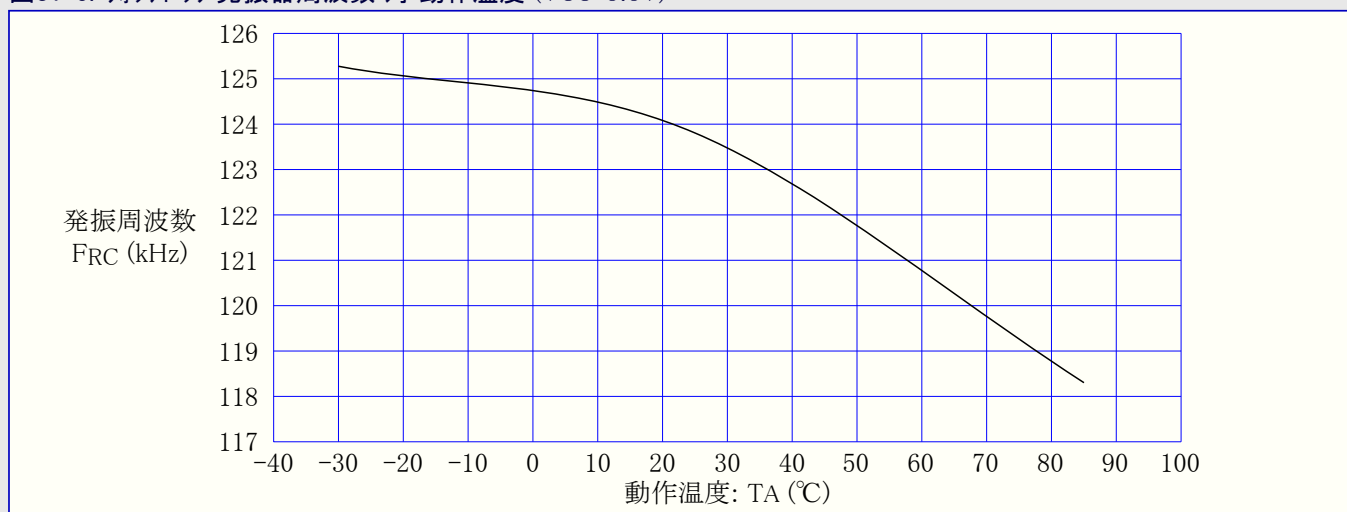


図31-6. 校正済み1MHz内蔵RC発振器周波数 対 動作温度 (VCC=3.3V)



図31-7. 校正付き1MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値 (VCC=3.3V)

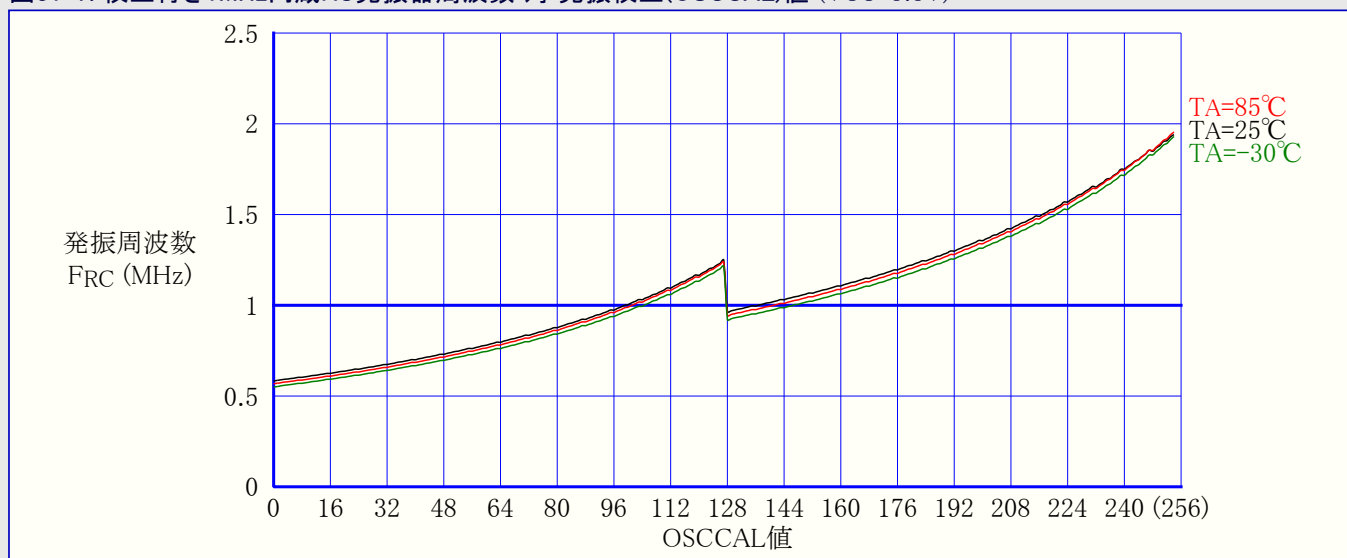
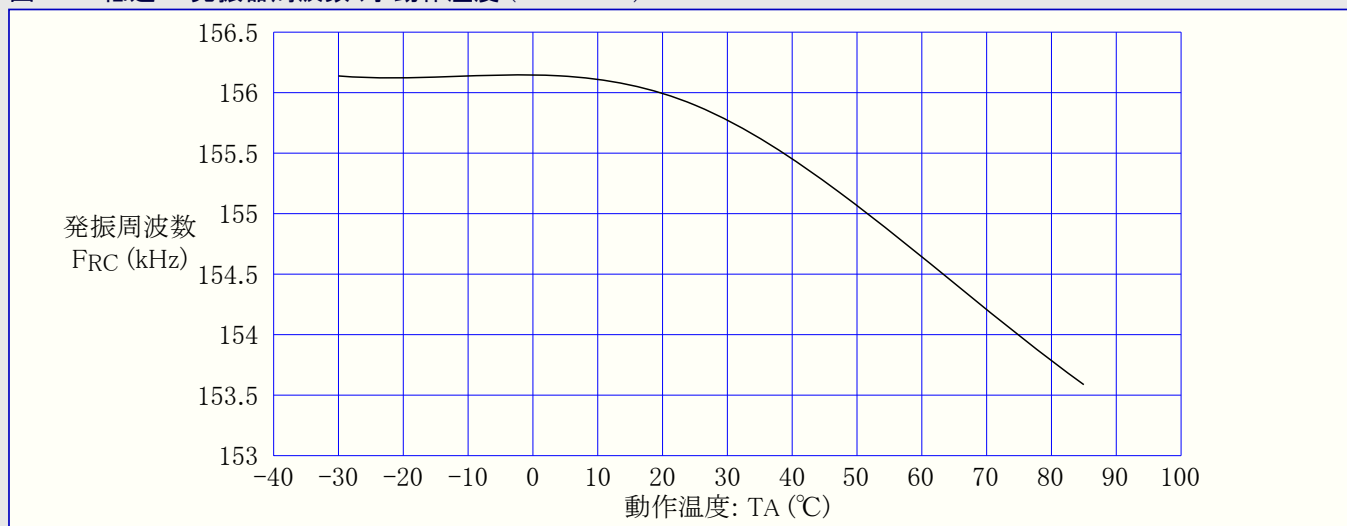


図31-8. 低速RC発振器周波数 対 動作温度 (VCC=3.3V)



32. レジスタ要約

拡張1/レジスタ領域 (1/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
(\$FF)	予約									
(\$FE)	予約									
(\$FD)	予約									
(\$FC)	予約									
(\$FB)	予約									
(\$FA)	予約									
(\$F9)	予約									
(\$F8)	BPPLR	-	-	-	-	-	-	BPPLR	BPPL	84
(\$F7)	BPCR	-	-	-	-	DUVD	SCD	DCD	CCD	84
(\$F6)	CBPTR	短絡保護確認時間(SCPT3~0)				過電流保護確認時間(OCPT3~0)				85
(\$F5)	BPOCD	放電過電流検出値(DCDL3~0)				充電過電流検出値(CCDL3~0)				85
(\$F4)	BPSCD	-	-	-	-	短絡検出値(SCDL3~0)				86
(\$F3)	BPDUV	-	-	DUVT1	DUVT0	最低電圧検出値(DUDL3~0)				86
(\$F2)	BPIR	DUDIF	COCIF	DOCIF	SCIF	DUDIE	COCIE	DOCIE	SCIE	87
(\$F1)	CBPCR	-	-	-	-	CBE4	CBE3	CBE2	CBE1	90
(\$F0)	FCSR	-	-	PWMOC	PWMOPC	CPS	DFE	CFE	PFD	89
(\$EF)	予約									
(\$EE)	予約									
(\$ED)	予約									
(\$EC)	予約									
(\$EB)	予約									
(\$EA)	予約									
(\$E9)	CADICH	CC-ADC 瞬時電流レジスタ上位 (CADIC15~8)								73
(\$E8)	CADICL	CC-ADC 瞬時電流レジスタ下位 (CADIC7~0)								
(\$E7)	CADRDC	CC-ADC 定常放電電流レジスタ (CADRDC7~0)								74
(\$E6)	CADRCC	CC-ADC 定常充電電流レジスタ (CADRCC7~0)								
(\$E5)	CADCSR	-	CADACIE	CADRDCIE	CADICIE	-	CADACIF	CADRDCIF	CADICIF	72
(\$E4)	CADCSRA	CADEN	-	CADUB	CADAS1	CADAS0	CADSI1	CADSI0	CADSE	72
(\$E3)	CADAC3	CC-ADC 累積電流レジスタ 最上位バイト (CADAC31~24)								73
(\$E2)	CADAC2	CC-ADC 累積電流レジスタ 第3バイト (CADAC23~16)								
(\$E1)	CADAC1	CC-ADC 累積電流レジスタ 第2バイト (CADAC15~8)								
(\$E0)	CADAC0	CC-ADC 累積電流レジスタ 最下位バイト (CADAC7~0)								
(\$DF)	予約									
(\$DE)	予約									
(\$DD)	予約									
(\$DC)	予約									
(\$DB)	予約									
(\$DA)	予約									
(\$D9)	予約									
(\$D8)	予約									
(\$D7)	予約									
(\$D6)	予約									
(\$D5)	予約									
(\$D4)	予約									
(\$D3)	予約									
(\$D2)	予約									
(\$D1)	BGCCR	基準電圧温度係数校正レジスタ (BGCR7~0)								81
(\$D0)	BGCCR	(BGEN)	-	基準電圧校正値 (BGCC5~0)						81
(\$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)	予約									
(\$C5)	予約									
(\$C4)	予約									
(\$C3)	予約									
(\$C2)	予約									
(\$C1)	予約									
(\$C0)	CCSR	-	-	-	-	-	-	XOE	ACS	19

(訳注) (BGEN)ビットは改訂デバイスから利用不可(削除)になりました。

拡張I/Oレジスタ領域 (2/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
(\$BF)	予約									
(\$BE)	TWBCSR	TWBCIF	TWBCIE	-	-	-	TWBDT1	TWBDT0	TWBCIP	111
(\$BD)	TWAMR	2線直列インターフェース:TWI(従装置)アドレス遮蔽レジスタ (TWAM6~0)							-	98
(\$BC)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	96
(\$BB)	TWDR	2線直列インターフェース:TWI データレジスタ								97
(\$BA)	TWAR	2線直列インターフェース:TWI(従装置)アドレス値 (TWA6~0)							TWGCE	97
(\$B9)	TWSR	2線直列インターフェース:TWI状態 (TWS7~3)					-	TWPS1	TWPS0	97
(\$B8)	TWBR	2線直列インターフェース:TWI ビット速度レジスタ								96
(\$B7)	予約									
(\$B6)	予約									
(\$B5)	予約									
(\$B4)	予約									
(\$B3)	予約									
(\$B2)	予約									
(\$B1)	予約									
(\$B0)	予約									
(\$AF)	予約									
(\$AE)	予約									
(\$AD)	予約									
(\$AC)	予約									
(\$AB)	予約									
(\$AA)	予約									
(\$A9)	予約									
(\$A8)	予約									
(\$A7)	予約									
(\$A6)	予約									
(\$A5)	予約									
(\$A4)	予約									
(\$A3)	予約									
(\$A2)	予約									
(\$A1)	予約									
(\$A0)	予約									
(\$9F)	予約									
(\$9E)	予約									
(\$9D)	予約									
(\$9C)	予約									
(\$9B)	予約									
(\$9A)	予約									
(\$99)	予約									
(\$98)	予約									
(\$97)	予約									
(\$96)	予約									
(\$95)	予約									
(\$94)	予約									
(\$93)	予約									
(\$92)	予約									
(\$91)	予約									
(\$90)	予約									
(\$8F)	予約									
(\$8E)	予約									
(\$8D)	予約									
(\$8C)	予約									
(\$8B)	予約									
(\$8A)	予約									
(\$89)	OCR1AH	タイマ/カウンタ1 比較Aレジスタ上位バイト								66
(\$88)	OCR1AL	タイマ/カウンタ1 比較Aレジスタ下位バイト								
(\$87)	予約									
(\$86)	予約									
(\$85)	TCNT1H	タイマ/カウンタ1 上位バイト								66
(\$84)	TCNT1L	タイマ/カウンタ1 下位バイト								
(\$83)	予約									
(\$82)	予約									
(\$81)	TCCR1B	-	-	-	-	CTC1	CS12	CS11	CS10	66
(\$80)	予約									

拡張I/Oレジスタ領域 (3/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	頁
(\$7F)	予約									
(\$7E)	DIDR0	-	-	-	-	VADC3D	VADC2D	VADC1D	VADC0D	79
(\$7D)	予約									
(\$7C)	VADMUX	-	-	-	-	VADMUX3	VADMUX2	VADMUX1	VADMUX0	78
(\$7B)	予約									
(\$7A)	VADCSR	-	-	-	-	VADEN	VADSC	ADCCIF	ADCCIE	78
(\$79)	VADCH	-	-	-	-	V-ADC データレジスタ 上位 (VADC11~8)				79
(\$78)	VADCL	V-ADC データレジスタ 下位 (VADC7~0)								
(\$77)	予約									
(\$76)	予約									
(\$75)	予約									
(\$74)	予約									
(\$73)	予約									
(\$72)	予約									
(\$71)	予約									
(\$70)	予約									
(\$6F)	TIMSK1	-	-	-	-	-	-	OCIE1A	TOIE1	67
(\$6E)	TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0	61
(\$6D)	予約									
(\$6C)	PCMSK1	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	39
(\$6B)	PCMSK0	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	39
(\$6A)	予約									
(\$69)	EICRA	ISC31	ISC30	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	38
(\$68)	PCICR	-	-	-	-	-	-	PCIE1	PCIE0	39
(\$67)	予約									
(\$66)	FOSCCAL	内蔵高速RC発振器 発振校正値レジスタ								19
(\$65)	予約									
(\$64)	PRR0	-	-	-	-	PRTWI	PRTIM1	PRTIM0	PRVADC	24
(\$63)	予約									
(\$62)	WUTCSR	WUTIF	WUTIE	WUTCF	WUTR	WUTE	WUTP2	WUTP1	WUTP0	32
(\$61)	予約									
(\$60)	WDTCR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	31

注意: ・ 将来のデバイスとの共通性のため、アクセスされる場合の予約ビットは0を書かれるべきです。予約したI/Oメモリ アドレスは決して書かれるべきではありません。

- ・ アドレス範囲\$00~\$1F内のI/OレジスタはSBIとCBI命令を使う直接ビット アクセスが可能です。これらのレジスタではSBISとSBIC命令を使うことによって単一ビット値が検査できます。
- ・ いくつかの状態ビットはそれらへ論理1を書くことによって解除(0)されます。他の多くのAVRと異なり、CBIとSBI命令は指定ビットだけ操作し、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。ATmega406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	7
\$3E (\$5E)	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	8
\$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	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	119
\$36 (\$56)	予約									
\$35 (\$55)	MCUCR	JTD	-	-	PUD	-	-	IVSEL	IVCE	115,47,36
\$34 (\$54)	MCUSR	-	-	-	JTRF	WDRF	BODRF	EXTRF	PORF	115,30
\$33 (\$53)	SMCR	-	-	-	-	SM2	SM1	SM0	SE	21
\$32 (\$52)	予約									
\$31 (\$51)	OCDR	IDRD/ OCDR7	OCDR6	OCDR5	OCDR4	OCDR3	OCDR2	OCDR1	OCDR0	114
\$30 (\$50)	予約									
\$2F (\$4F)	予約									
\$2E (\$4E)	予約									
\$2D (\$4D)	予約									
\$2C (\$4C)	予約									
\$2B (\$4B)	GPIOR2	汎用I/Oレジスタ2								16
\$2A (\$4A)	GPIOR1	汎用I/Oレジスタ1								16
\$29 (\$49)	予約									
\$28 (\$48)	OCR0B	タイマ/カウンタ0 比較レジスタ								60
\$27 (\$47)	OCR0A	タイマ/カウンタ0 比較Aレジスタ								60
\$26 (\$46)	TCNT0	タイマ/カウンタ0								60
\$25 (\$45)	TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	59
\$24 (\$44)	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	58
\$23 (\$43)	GTCCR	TSM	-	-	-	-	-	-	PSRSYNC	69
\$22 (\$42)	EEARH	-	-	-	-	-	-	-	EEAR8	12
\$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								12
\$1F (\$3F)	EECR	-	-	EEPM1	EEPM0	EERIE	EEMPE	EEPE	EERE	12
\$1E (\$3E)	GPIOR0	汎用I/Oレジスタ0								16
\$1D (\$3D)	EMSK	-	-	-	-	INT3	INT2	INT1	INT0	38
\$1C (\$3C)	EIFR	-	-	-	-	INTF3	INTF2	INTF1	INTF0	38
\$1B (\$3B)	PCIFR	-	-	-	-	-	-	PCIF1	PCIF0	39
\$1A (\$3A)	予約									
\$19 (\$39)	予約									
\$18 (\$38)	予約									
\$17 (\$37)	予約									
\$16 (\$36)	TIFR1	-	-	-	-	-	-	OCF1A	TOV1	67
\$15 (\$35)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0	61
\$14 (\$34)	予約									
\$13 (\$33)	予約									
\$12 (\$32)	予約									
\$11 (\$31)	予約									
\$10 (\$30)	予約									
\$0F (\$2F)	予約									
\$0E (\$2E)	予約									
\$0D (\$2D)	予約									
\$0C (\$2C)	予約									
\$0B (\$2B)	PORTD	-	-	-	-	-	-	PORTD1	PORTD0	48
\$0A (\$2A)	DDRD	-	-	-	-	-	-	DDD1	DDD0	48
\$09 (\$29)	PIND	-	-	-	-	-	-	PIND1	PIND0	48
\$08 (\$28)	PORTC	-	-	-	-	-	-	-	PORTC0	48
\$07 (\$27)	予約									
\$06 (\$26)	予約									
\$05 (\$25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	47
\$04 (\$24)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	47
\$03 (\$23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	47
\$02 (\$22)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	47
\$01 (\$21)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	47
\$00 (\$20)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	47

(訳注) 原書本位置の注意は前頁に移動しました。

33. 命令要約

ニーモニック	オペラント	意味	動作	フラグ	クロック
算術、論理演算命令					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,1	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,0,0,0,1,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		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		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, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	$Rd - Rr$	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	$Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	$Rd - K$	I,T,H,S,V,N,Z,C	1
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ*	$Rr(b)=0$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBRS	Rr,b	汎用レジスタのビットが設定(1)でスキップ*	$Rr(b)=1$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ*	$P(b)=0$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBIS	P,b	I/Oレジスタのビットが設定(1)でスキップ*	$P(b)=1$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
BRBS	s,k	ステータスフラグが設定(1)で分岐	$SREG(s)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRBC	s,k	ステータスフラグが解除(0)で分岐	$SREG(s)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BREQ	k	一致で分岐	$Z=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRNE	k	不一致で分岐	$Z=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCS	k	キャリーフラグが設定(1)で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCC	k	キャリーフラグが解除(0)で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRSH	k	符号なしの $\geq$ で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしの $<$ で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	$N=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	$N=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの $\geq$ で分岐	$(N \text{ EOR } V)=0$ 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X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	ニフル(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)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,1,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
MCU制御命令					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグ機能専用	I,T,H,S,V,N,Z,C	N/A

34. 注文情報

速度(MHz)	電源電圧	注文コード	外囲器	動作範囲
1	4.0~25.0V	ATmega406-1AAU (注2)	48AA	工業用 (-30℃~85℃)

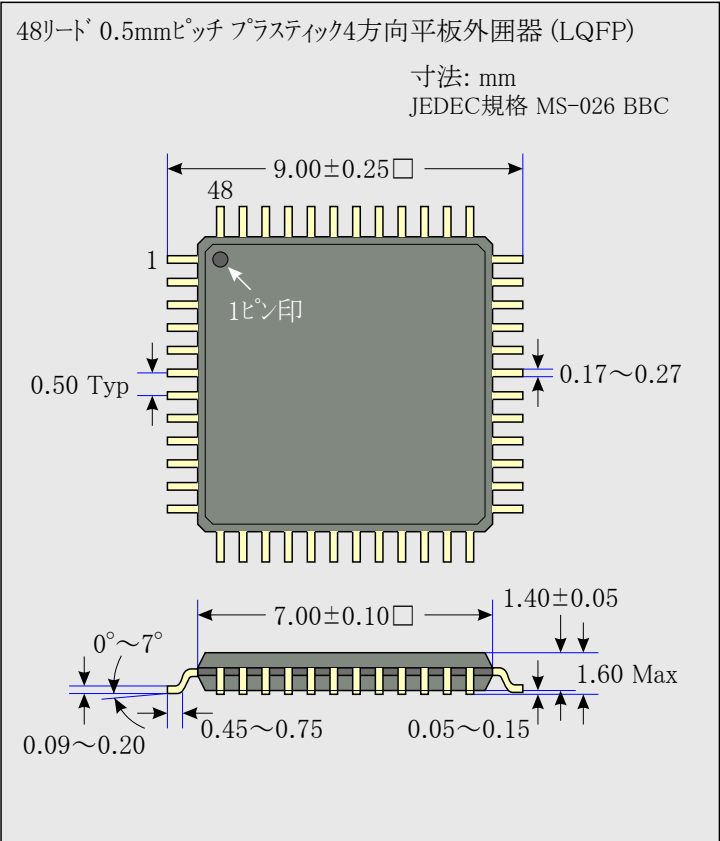
注: このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

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。

外囲器形式	
48AA	48リード 7×7×1.44mm 0.5mmピッチ 薄型プラスチック4方向平板外囲器 (LQFP)

35. 外囲器情報

35.1. 48AA



36. 障害情報

この章の改訂番号はATmega406デバイスの改訂版を参照してください。

- ・ 電圧A/D変換器が0℃以下で機能しない
- ・ 電圧A/D変換器の同相入力定常偏差
- ・ 基準電圧のスパイク
- ・ 電圧調整器部始動手順
- ・ VREFがMCUの状態によって影響される
- ・ 応用コードからのEEPROM読み出しが施錠ビット種別3で働かない

D/E
D/E/F
D/E/F
D
D
D

1. 低温での電圧A/D変換器誤り (D/E)

電圧A/D変換器は0℃以下で機能しません。電圧A/D変換器は0℃以下で非常に大きな誤差を持ち、使えません。

対策/対処

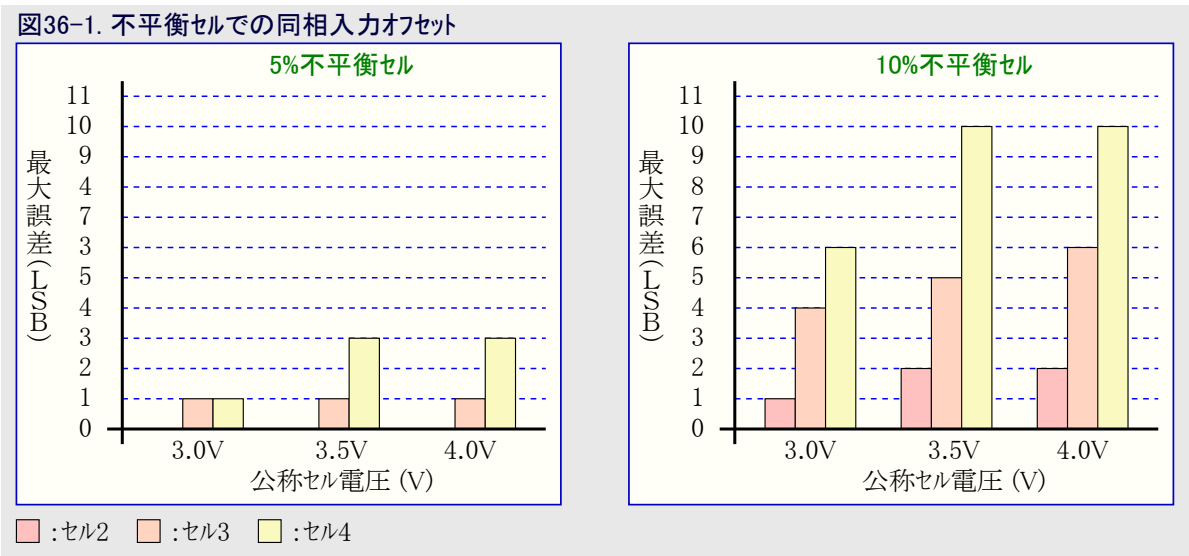
ありません。

2. 電圧A/D変換器の同相入力定常偏差 (D/E/F)

電池セル変換は同相入力(CM)水準に依存する偏り誤差を持ちます。これは電池セルの誤差が下位セルの電圧に依存することを意味します。CM定常偏差はAtmelの製造でセル平衡時に校正されます。セルが不平衡になると、CMに依存する定常偏差が再び出現します。

- (1) セル1はそれ自身のCMレベルを定義し、CMに依存する定常偏差によって決して影響を及ぼされません。
- (2) セル2に対するCMレベルはセル1電圧がセル2電圧と離れている場合に変わります。
- (3) セル3に対するCMレベルはセル1電圧やセル2電圧がセル3電圧と離れている場合に変わります。最悪状態誤差はセル1とセル2が平衡している一方でセル3電圧がセル1とセル2の電圧と離れている場合です。
- (4) セル4に対するCMレベルはセル1電圧、セル2電圧やセル3電圧がセル4電圧と離れている場合に変わります。最悪状態誤差はセル1とセル2とセル3が平衡している一方でセル4電圧がセル1、セル2、セル3の電圧と離れている場合です。

図36-1.は5%と10%の不平衡セルでのセル2、セル3、セル4の誤差を示します。



対策/対処

内蔵セル平衡FETを使うことによって不平衡セルになるのを避けてください。

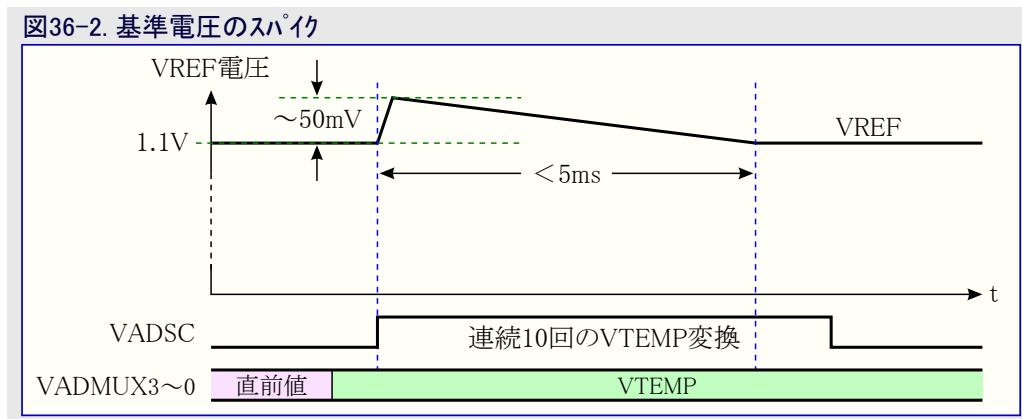
3. 基準電圧のスパイク (D/E/F)

基準電圧(VREF)は電圧A/D変換器(V-ADC)で温度測定が開始される毎にスパイクが出ます。

対策/対処

正確な温度測定は10回の温度A/D変換を行うことによって、その直後に得られます。最初の9回の結果は不正確でしょうが、10回目の変換は正しいでしょう。

図36-2.は連続10回の温度変換を行う時の基準電圧上のスパイクを図解します(外部デカップ コンデンサ=1μF)。



V-ADCが温度測定を行っている間にクーロン カンタA/D変換器(CC-ADC)が電流累積を行う場合、瞬時と累積の両方の変換結果が影響を及ぼされます。VREF上のスパイクは1回の累積電流変換結果(CADAC3~0レジスタ)または2回の瞬時電流変換結果(CADICH/Lレジスタ)で見えるでしょう。

4. 電圧調整器部始動手順 (D)

ATmega406電源投入時、電圧調整器部の正しい始動を保証するのにいくつかの予防処置が必要です。

対策/対処

電圧調整器部の正しい始動を保証するのに以下の3段階が必要とされます。

1. VFETピンに0.1μFよりも大きな容量を接続しないでください。これは供給電圧接続時にVFETピンの高速上昇時間を保証するためです。
2. 組み立て中、常に最初にセル1、次にセル2と、最上位セルがPVTに接続されるまで同様に接続してください。セル電圧が約2Vまたはより大きければ、電圧調整器部は標準的に正しくパワーオフ動作で始動するでしょう(VREGは概ね2.8V)。
3. 手順2.で記載されるように電池セルが組み立てられてしまった後、チップを初期化するために充電器がBATT+端子に接続されなければなりません。本データシートで27頁の「電源ON/リセットと充電器接続」項をご覧ください。

セルの組み立て中に電圧調整器部がパワーオフ動作で始動すると、充電器がBATTピンで7~8Vを越える電圧にする時にチップが初期化を始めるでしょう。

電圧調整器部が正しく始動しない場合、正しい始動と初期化を保証するために、充電器には1つの追加必要条件があります。この場合の充電器はVFETピン電圧がPVTピン電圧以上で最低3Vに素早く増加し、しかもBATTピン電圧が7~8Vを越えるのを保証しなければなりません。これがチップを直接的に始動、初期化するでしょう。

5. VREFがMCUの状態によって影響される (D)

VREFピンでの基準電圧は次のデバイス条件に依存します。

- ・ 過充電保護や過放電保護が活性(許可)で短絡保護が非活性(禁止)。これは全ての電流保護が禁止される状態と比較してVREF電圧を1mV(代表値)増加するでしょう。
- ・ 短絡保護が活性(許可)の場合。電池保護制御レジスタ(BPCR)の短絡保護禁止(SCD)が0(既定)で、且つFET制御/状態レジスタ(FCSR)の放電FET許可(DFE)が設定(1)の時に、短絡測定が活動(実行)。これは短絡測定が非活性(非動作)の状態と比較してVREF電圧を8mV(代表値)増加するでしょう。
- ・ 内部VTEMP電圧のV-ADC変換。これは短絡測定が非活性(非動作)の状態と比較してVREF電圧を15mV(代表値)増加するでしょう。

対策/対処

最高精度を保証するにはチップが実際の電池保護設定で、且つ放電FETが許可された後、VREFで1.100Vを得るために基準電圧校正レジスタ(BGCCR)のバントギャップ基準電圧校正値(BGCC5~0)を設定してください。

6. 応用コードからのEEPROM読み出しが施錠ビット種別3で働かない (D)

メモリ施錠ビットのLB2とLB1が動作種別3にプログラム(設定)されると、EEPROM読み出しが応用コードから動作しません。

対策/対処

応用コードがEEPROMからの読み出しを必要とする時は施錠ビット保護動作種別3を設定しないでください。

37. データシート改訂履歴

この章内の参照頁番号はこの資料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資料の改訂番号を参照してください。

37.1. 2548A – 2005年1月

1. 初版

37.2. 2548B – 2005年4月

1. 誤植更新、'PSRASY'ビット削除、CS12:0をCS1[2:0]に改名
2. BGCCRレジスタでBGENビット削除。バンドギャップ基準電圧はATmega406改訂Eで常に許可
3. 3頁の図2-1., 17頁の図7-1., 83頁の図22-1., 94頁の図25-9., 128頁の図28-1.を更新
4. 23頁の表8-2.と表8-3., 149頁の表30-4., 123頁の表27-5.を更新
5. 44頁の「13.3.2. ポートAの交換機能」項と82頁の「22. 電池保護」項記述を更新
6. 38頁の「EIFR – 外部割り込み要求フラグ レジスタ」と59頁の「TCCR0B – タイマ/カウンタ0制御レジスタB」のレジスタ更新
7. 70頁の「18.1. 特徴」項と「18.2. 操作」項、76頁の「20.1. 特徴」項、81頁の「20.2. 基準電圧と温度感知器用レジスタ」項を更新
8. 145頁の「30. 電気的特性」項を更新
9. 160頁の「36. 障害情報」項を更新

37.3. 2548C – 2005年5月

1. 160頁の「36. 障害情報」項を更新

37.4. 2548D – 2005年6月

1. 160頁の「36. 障害情報」項を更新

37.5. 2548E – 2006年7月

1. 4頁の「ピン説明」を更新
2. 21頁の「A/D変換雑音低減動作」を更新
3. 22頁の「パワーセーフ動作」を更新
4. 22頁の「パワーダウン動作」を更新
5. 22頁の「パワーオフ動作」を更新
6. 24頁の「電力削減(レジスタ)」を更新
7. 25頁に「電圧A/D変換器」と「クーロン カウンタA/D変換器」を追加
8. 26頁の「リセット元」を更新
9. 27頁の「電源ONリセットと充電器接続」を更新
10. 27頁の「外部リセット」を更新
11. 27頁の「低電圧(ブラウアウト)検出リセット」でVREGによってVCCを置換
12. 43頁からの「交換ポート機能」を更新
13. 68頁の「内部クロック」を更新
14. 68頁の「外部クロック」を更新
15. 70頁の「クーロン カウンタ (容量/残量計専用 $\Delta\Sigma$ A/D変換器)」で特徴を更新
16. 70頁の「18. クーロン カウンタ (容量/残量計専用 $\Delta\Sigma$ A/D変換器)」で操作を更新
17. 75頁の「電圧調整器」で特徴を更新
18. 75頁の「電圧調整器」で動作を更新
19. 79頁の「VADCH, VADCL – V-ADCデータ レジスタ」でビット記述を更新
20. 80頁の「基準電圧校正レジスタへの書き込み」を更新
21. 89頁の「FET制御用レジスタ」内で文章を更新
22. 115頁に「MCUCR – MCU制御レジスタ」を追加
23. 143頁の「動作回路」を更新
24. 145頁の「電気的特性」を更新
25. 150頁から「代表特性 – 暫定」を追加
26. 153頁からの「レジスタ要約」を更新
27. 160頁の「障害情報」を更新
28. 31頁の表9-2., 123頁の表27-5.を更新

29. 23頁の図8-1., 27頁の図9-5., 68頁の図17-2., 70頁の図18-2., 71頁の図18-3., 75頁の図19-1., 143頁の図29-1.を更新

30. レジスタ アドレス更新

37.5. 2548F – 2013年3月

1. 4頁で「PPI,NNI」と「PI,NI」の表題を更新(交換)

2. 123頁の表27-5.で注10を更新

3. 146頁の30.2.項を更新

目次

特徴	1	12.3. 外部割り込み用レジスタ	38
1. ピン配置	2	13. 低電圧入出力ポート	40
1.1. お断り	3	13.1. 序説	40
2. 概要	3	13.2. 標準デジタル入出力としての低電圧ポート	40
2.1. 構成図	3	13.3. 交換ポート機能	43
2.2. ピン説明	4	13.4. 低電圧I/Oポート用レジスタ	47
3. 資料	5	14. 高電圧入出力ポート	49
4. コード例について	5	14.1. 標準デジタル入出力としての高電圧ポート	49
5. AVR CPU コア	6	14.2. ピンの設定	49
5.1. 序説	6	14.3. 高電圧I/Oポート用レジスタ	49
5.2. 構造概要	6	15. 8ビット タイマ/カウンタ0 (PWM)	50
5.3. ALU (Arithmetic Logic Unit)	6	15.1. 概要	50
5.4. ステータス レジスタ	7	15.2. タイマ/カウンタのクロック	51
5.5. 汎用レジスタ ファイル	8	15.3. 計数器部	51
5.6. スタック ポインタ	8	15.4. 比較出力部	52
5.7. 命令実行タイミング	9	15.5. 比較一致出力部	53
5.8. リセットと割り込みの扱い	9	15.6. 動作種別	54
6. AVR メモリ	11	15.7. タイマ/カウンタのタイミング	57
6.1. 実装書き換え可能なプログラム用フラッシュ メモリ	11	15.8. 8ビット タイマ/カウンタ0用レジスタ	58
6.2. データ用SRAMメモリ	11	16. 16ビット タイマ/カウンタ1	62
6.3. データ用EEPROMメモリ	12	16.1. 概要	62
6.4. I/Oメモリ (レジスタ)	16	16.2. 16ビット レジスタのアクセス	63
7. システム クロックとクロック任意選択	17	16.3. タイマ/カウンタのクロック	64
7.1. クロックシステムとその配給	17	16.4. 計数器部	65
7.2. クロック元	18	16.5. 比較出力部	65
7.3. 校正付き高速RC発振器	18	16.6. 16ビット タイマ/カウンタ1用レジスタ	66
7.4. 32kHzクリスタル用発振器	18	17. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器	68
7.5. 低速RC発振器	18	17.1. 内部クロック	68
7.6. 超低電力RC発振器	18	17.2. 前置分周器リセット	68
7.7. CPU,I/O,フラッシュ,電圧A/D変換のクロック	18	17.3. 外部クロック	68
7.8. クーロン カウンタA/D変換と起動タイマのクロック	18	17.4. タイマ/カウンタ前置分周器用レジスタ	69
7.9. ウォッチドッグ タイマと電池保護のクロック	19	18. クーロン カウンタ (容/残量計専用 $\Delta\Sigma$ A/D変換器)	70
7.10. 実行時クロック元選択	19	18.1. 特徴	70
7.11. クロック関係レジスタ	19	18.2. 操作	70
8. 電力管理と休止形態	21	18.3. クーロン カウンタA/D変換器用レジスタ	72
8.1. アイドル動作	21	19. 電圧調整器	75
8.2. A/D変換雑音低減動作	21	19.1. 特徴	75
8.3. パワーセーブ動作	22	19.2. 動作	75
8.4. パワーダウン動作	22	20. 電圧ADC (10チャネル汎用12ビット $\Delta\Sigma$ A/D変換器)	76
8.5. パワーオフ動作	22	20.1. 特徴	76
8.6. 電力削減(電力削減レジスタ)	24	20.2. 操作	77
8.7. 消費電力の最小化	25	20.3. 電圧A/D変換器用レジスタ	78
9. システム制御とリセット	26	21. 基準電圧と温度感知器	80
9.1. AVRのリセット	26	21.1. 特徴	80
9.2. リセット元	26	21.2. 基準電圧校正レジスタへの書き込み	80
9.3. ウォッチドッグ タイマ	28	21.3. 基準電圧と温度感知器用レジスタ	81
9.4. リセット関係用レジスタ	30	22. 電池保護	82
10. 起動タイマ	32	22.1. 特徴	82
10.1. 概要	32	22.2. 最低電圧保護	82
10.1. 起動タイマ用レジスタ	32	22.3. 放電過電流保護	82
11. 割り込み	33	22.4. 充電過電流保護	83
11.1. ATmega406の割り込みベクタ	33	22.5. 短絡保護	83
11.2. 応用/ポート領域間の割り込みベクタ移動	36	22.6. 電池保護CPUインターフェース	83
11.3. 割り込みベクタ移動用レジスタ	36	22.7. 電池保護用レジスタ	84
12. 外部割り込み	37	23. FET制御	88
12.1. 概要	37	23.1. FET駆動部	88
12.2. ピン変化割り込みタイミング	37	23.2. FET制御用レジスタ	89

24.	電池セル平衡	90
24.1.	電池セル平衡用レジスタ	90
25.	2線直列インターフェース (TWI)	91
25.1.	特徴	91
25.2.	2線直列インターフェース バスの定義	91
25.3.	データ転送とフレーム形式	91
25.4.	複数主装置バス システムの調停と同期	93
25.5.	TWI部の概要	94
25.6.	TWI用レジスタ	96
25.7.	TWIの使用法	99
25.8.	転送種別	101
25.9.	複数主装置システムでのバス競合と調停	110
25.10.	2線直列インターフェース用バス接続/切断	111
26.	JTAGインターフェースと内蔵デバッグ機能	112
26.1.	特徴	112
26.2.	概要	112
26.3.	検査入出力ポート(TAP:Test Access Port)	113
26.4.	TAP制御器	113
26.5.	内蔵デバッグ機能の使用	114
26.6.	内蔵デバッグ特殊JTAG命令	114
26.7.	内蔵デバッグに関連するI/Oメモリ内のレジスタ	114
26.8.	JTAGプログラミング能力の使用	115
26.9.	JTAG-I/Fに関連するI/Oメモリ内のレジスタ	115
27.	ブート ロード支援 – RWW 自己プログラミング	116
27.1.	特徴	116
27.2.	フラッシュ メモリの応用領域とブート ロード領域	116
27.3.	書き中に読みが可能な領域と不能な領域	116
27.4.	ブート ロード施錠ビット	118
27.5.	ブート ロード プログラムへの移行	118
27.6.	自己プログラミングでのフラッシュ アドレス指定	120
27.7.	フラッシュ メモリの自己プログラミング	120
28.	メモリ プログラミング	126
28.1.	プログラム メモリとデータ メモリ用施錠ビット	126
28.2.	ヒューズ ビット	127
28.3.	識票バイト	127
28.4.	校正バイト	127
28.5.	ページ容量	128
28.6.	並列プログラミング	128
28.7.	並列プログラミング手順	129
28.8.	JTAGインターフェース経由プログラミング	136
29.	動作回路	143
30.	電気的特性	145
30.1.	絶対最大定格	145
30.2.	DC特性	145
30.3.	汎用入出力線特性	147
30.4.	2線直列インターフェース(TWI)特性	148
30.5.	リセット特性	149
30.6.	周辺機能部供給電流	149
31.	代表特性 (暫定)	150
31.1.	ピン プルアップ	150
31.2.	ピン駆動能力	150
31.3.	内部発振器周波数	151
32.	レジスタ要約	153
33.	命令要約	157
34.	注文情報	159
35.	外圍器情報	159
36.	障害情報	160
37.	データシート改訂履歴	162



Enabling Unlimited Possibilities®

Atmel Corporation

1600 Technology Drive
San Jose, CA 95110
USA
TEL (+1)(408) 441-0311
FAX (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited

Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
TEL (+852) 2245-6100
FAX (+852) 2722-1369

Atmel Munich GmbH

Business Campus
Parking 4
D-85748 Garching b. Munich
GERMANY
TEL (+49) 89-31970-0
FAX (+49) 89-3194621

Atmel Japan G.K.

141-0032 東京都品川区
大崎1-6-4
新大崎勸業ビル 16F
アトメル ジャパン合同会社
TEL (+81)(3)-6417-0300
FAX (+81)(3)-6417-0370

© 2013 Atmel Corporation. 不許複製 / 改訂:2548F-AVR-03/2013

Atmel®, Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®, 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。

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、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© HERO 2021.

本データシートはAtmelのATmega406英語版データシート(改訂2548F-03/2013)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。

原書に対して若干構成が異なるため、一部の節/項番号が異なります。