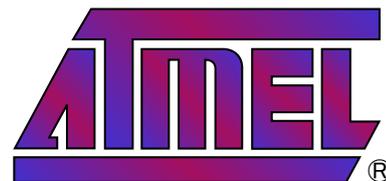


## 特徴

- 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129/131命令(多くは1周期で実行)
  - 32個の1バイト長汎用レジスタ
  - 完全なステティック動作
  - 4MHz時、4MIPS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8K/16Kバイト(4K/8K語)フラッシュメモリ
  - 256バイトのEEPROM
  - 512バイトの内蔵SRAM
  - 書き換え回数: 10,000/フラッシュ, 100,000/EEPROM
  - データ保持力: 20年/85°C, 100年/25°C
  - ソフトウェア保護用の設定可能な施錠機能
- 電池管理機能
  - 直列1または2セル
  - 過電流保護(充電と放電)
  - 短絡保護(放電)
  - 充放電Nch-FET駆動用高電圧出力
- 内蔵周辺機能
  - 独立した前置分周器と任意選択の捕獲、比較、CTC動作付きの2つの構成設定可能な8または16ビットタイマ/カウンタ
  - SPI - 直列周辺インターフェース
  - 外部4と内部1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
  - 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
  - デバッグWIRE内蔵デバッグシステム
  - SPIポート経由での実装書き換え
  - 電源ONリセット
  - 短絡回路監視インターフェース付き内蔵電圧調整器
  - 外部及び内部の割り込み
  - アイドル、A/D変換雑音低減、パワーセーブ、パワーオフの4つの低消費動作
- NDA(機密保持契約)下でのみ利用可能な追加の安全認証機能
- 外圍器
  - 36ピンLGA
  - 28ピンTSOP
- 動作電圧: 1.8~9V
- 最大耐電圧(高電圧ピン): 28V
- 温度範囲: -20~85°C
- 動作周波数: 1~4MHz



8ビット AVR®  
マイクロコントローラ  
実装書き換え可能な  
8/16Kバイト  
フラッシュメモリ内蔵

ATmega8HVA  
ATmega16HVA

暫定

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

Rev. 8024A-04/08, 8024AJ12-08/22

## 1. ピン配置

### 1.1. LGA

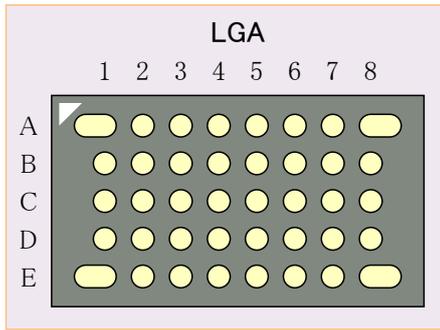
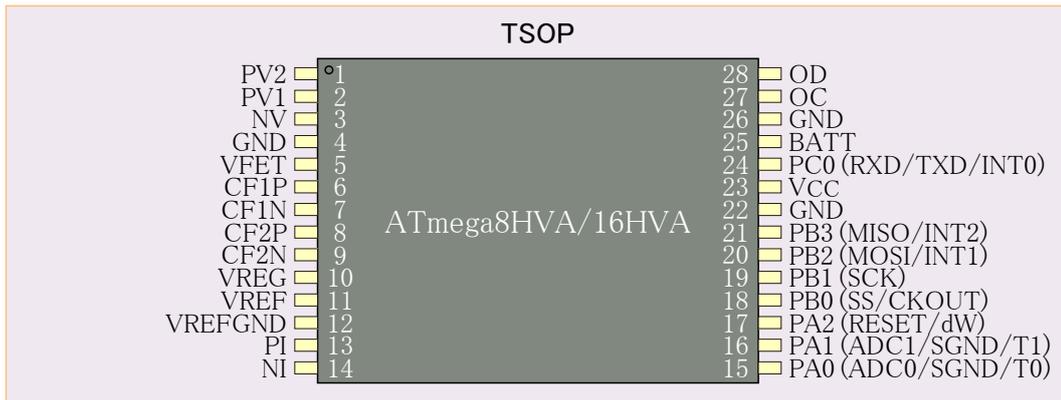


表1-1. LGA ピン配列 (ATmega8HVA/16HVA)

1	2	3	4	5	6	7	8	
<b>A</b>	DNC	PV2	PV1	NV	GND	OC	OD	DNC
<b>B</b>	CF2P	CF2N	VFET	CF1P	GND	PC0	DNC	GND
<b>C</b>	VREF	VREFGND	VREG	CF1N	VCC	GND	GND	BATT
<b>D</b>	PI	NI	GND	GND	GND	PB2	PB3	GND
<b>E</b>	DNC	DNC	PA1	PA0	PB1	PB0	RESET	DNC

### 1.2. TSOP



### 1.3. ピン説明

#### 1.3.1. VFET

内部電圧調整器に対する入力ピン。

#### 1.3.2. VCC

デジタル供給電圧(電源)。通常、VREGに接続されます。

#### 1.3.3. VREG

内部電圧調整器からの出力。

#### 1.3.4. CF1P/CF1N/CF2P/CF2N

CF1P/CF1N/CF2P/CF2Nは昇圧調整器のための外部昇圧コンデンサ接続用の接続ピンです。

#### 1.3.5. VREF

外部デカップ用内部基準電圧。

#### 1.3.6. VREFGND

内部基準電圧の雑音分離用接地。基板上のGNDまたはSGNDへ接続しないでください。

#### 1.3.7. GND

接地。

#### 1.3.8. PA1~PA0 (ポートA)

ポートAは(ビット毎に選ばれる)内蔵プルアップ抵抗付き低電圧2ビット双方向入出力ポートとして扱います。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ても、ポートAピンはHi-Zになります。

ポートAは45頁の「[ポートAの交換機能](#)」で一覧されるATmega8HVA/16HVAの様々な特殊機能も扱います。

#### 1.3.9. PB3~PB0 (ポートB)

ポートBは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの低電圧4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ても、ポートBピンはHi-Zになります。

ポートBは46頁の「[ポートBの交換機能](#)」で一覧されるATmega8HVA/16HVAの様々な特殊機能も扱います。

#### 1.3.10. PC0 (ポートC)

ポートCは40頁の「[ポートCの交換機能](#)」で一覧されるATmega8HVA/16HVAの様々な特殊機能も扱います。

#### 1.3.11. OC

充電FET駆動用高電圧出力。

#### 1.3.12. OD

放電FET駆動用高電圧出力。

#### 1.3.13. NI

NIは電流検出抵抗からの濾波された負入力です。

#### 1.3.14. PI

PIは電流検出抵抗からの濾波された正入力です。

#### 1.3.15. NV,PV1,PV2

NV,PV1,PV2は電池セル1と2用の入力です。

#### 1.3.16. BATT

充電器接続時の検出用入力。

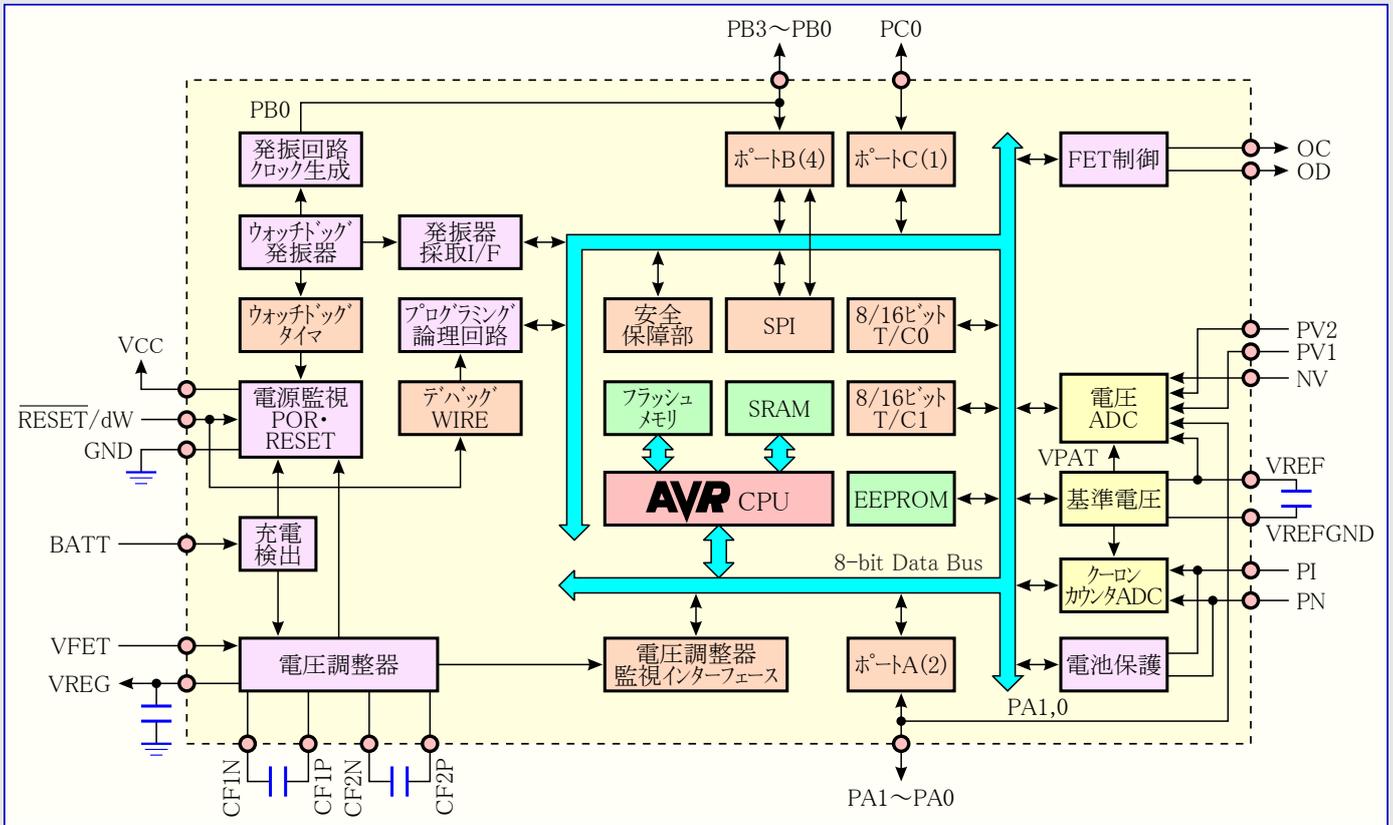
#### 1.3.17. RESET

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114頁の表29-6.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。このピンはデバッグWIRE通信ピンとしても使われます。

## 2. 概要

ATmega8HVA/16HVAは高い安全性/認証、正確な監視、低費用、高いセル エネルギー利用に集中した1または2セルのLi-ion応用に対する監視回路と保護回路です。デバイスは充放電中の自律的な電池保護だけでなく安全認証機能も含まれます。デバイスは1.84 $\mu$ V分解能の18ビットA/D変換器を使って、非常に正確な累積電流測定が可能です。これらの機能一式は高い安全、電池保護、正確な監視、高いシステム利用率、低費用に注目するどのシステムでもATmega8HVA/16HVAを中核部品にします。

図2-1. 構成図



結合された昇圧と直線型電圧調整器は1セルに対して1.8Vのような低い供給電圧でチップが動作できることを保証します。この電圧調整器は入力電圧が充分高い時に直線型動作へ自動的に切り換え、これによって常に最小消費電力を保証します。2セル応用に対しては直線型供給だけが許可されます。節電動作での極端に低い電力消費と結合した電圧調整器の能力は既存の方法と比べてセルエネルギー利用を大きく増強します。

チップは独立した予備充電FETを使うことなく、過放電した電池セルの予備充電を支援するAtmel特許の深下電圧回復(DUVR)動作を利用します。

ATmega8HVA/16HVAは各セルの電圧を個別に測定するのに使える12ビットA/D変換器を含みます。A/D変換器は組み込み温度感知器を使うチップ上温度、または専用のADC入力へ接続されたサーミスタを使う外部温度のどちらに於いても、温度を監視するのも使うことができます。ATmega8HVA/16HVAは高電圧に耐え、直列通信を支援するオープントレインI/Oピンを含みます。プログラミングはSPIプログラミングを支援する4つの汎用I/Oポートを使って実装環境で行うことができます。

MCUは書き込み中読み出し能力の8K/16Kバイトの実装書き換え可能なフラッシュメモリと256バイトのEEPROM、512バイトのSRAM、32個の汎用作業レジスタ、6本の汎用入出力線、チップ内デバッグ用のデバッグWIREと実装プログラミング用のSPI、捕獲と比較動作を含む柔軟性のある2つのタイマ/カウンタ、内部及び外部の割り込み、電圧と温度計測用の12ビット $\Delta$ ΣA/D変換器、クーロン計測と瞬時電流計測用の高分解能 $\Delta$ ΣA/D変換器、追加の安全認証機能、自律型電池保護部、起動能力付きの設定可能なウォッチドッグタイマ、ソフトウェアで選べる節電動作を含みます。

AVRコアは32の汎用作業レジスタと豊かな命令群を結合します。32のレジスタの全ては論理演算部(ALU)へ直結され、1つの単独命令で入出力される2つの独立したレジスタが1クロック周期で実行されるのを可能にします。その結果の構造はよりもっと効率的なコードと同時に現状のCISC型マイクロコントローラより10倍までの単位処理量を達成します。

本デバイスは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。チップ上のISPは規定の不揮発性メモリ書き込み器またはAVRコア上で走行するブートプログラムによって、SPI直列インターフェースを通して実装での再書き込みをプログラムメモリに許します。モ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、容量/残量計A/D変換器、専用電池保護回路、電圧調整器と8ビットRISC型CPUの結合によるATmega8HVA/16HVA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。

ATmega8HVA/16HVA AVRはCコンパイラ、マクロアセンブラ、プログラムデバッグ/シミュレータ、内蔵デバッグ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。

ATmega8HVA/16HVAはAVR構造に基いた低電力CMOS 8ビットマイクロコントローラです。リチウムイオン電池セルに対する安全認証、正確な監視、自律型保護を提供するAVRスマートバッテリー系列の一員です。

## 2.1. ATmega8HVAとATmega16HVA間の違い

ATmega8HVAとATmega16HVAはメモリ容量と割り込みベクタの大きさが異なるだけです。表2-1は2つのデバイスに対する異なる形態を要約します。

表2-1. 形態要約

デバイス名	フラッシュメモリ	割り込みベクタの大きさ
ATmega8HVA	8Kバイト	1命令語/ベクタ
ATmega16HVA	16Kバイト	2命令語/ベクタ

## 3. お断り

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

## 4. 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。

## 5. データ保持力

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

## 6. コード例について

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

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。

## 7. AVR CPU コア

### 7.1. 概要

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。

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

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

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

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

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

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

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

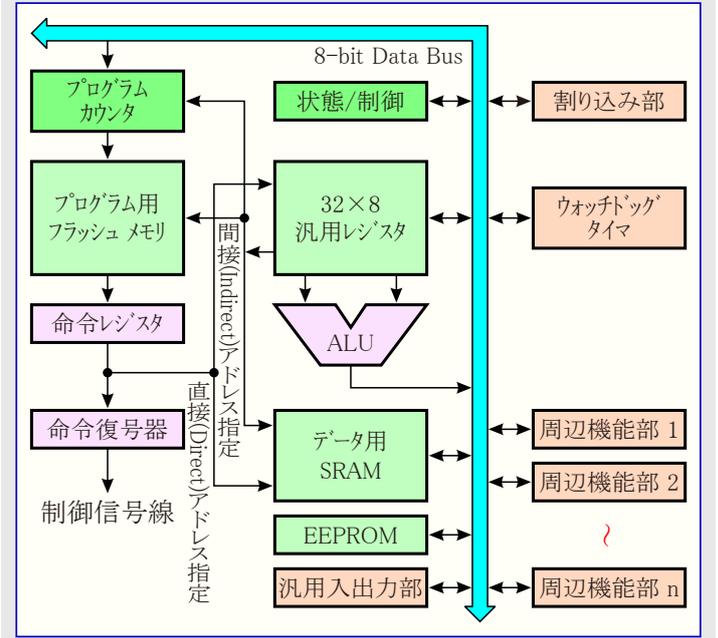
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(1ビット)があります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタと他の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ATmega8HVA/16HVA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。

### 7.2. 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7-1. AVR構造の構成図



## 7.3. ステータス レジスタ

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

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

### 7.3.1. 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								
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。そのとき、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され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、**RETI**命令によって設定(1)されます。Iビットは「命令一式手引書」で記述されるように**SEI**や**CLI**命令で応用(プログラム)によって設定(1)や解除(0)もできます。

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

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

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

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

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

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

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

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

#### • ビット2 – N: 負フラグ (Negative Flag)

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

#### • ビット1 – Z: ゼロフラグ (Zero Flag)

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

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

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

## 7.4. 汎用レジスタ ファイル

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

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

図7-2.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7-2.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7-2. AVR CPU 汎用レジスタ構成図

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

### 7.4.1. Xレジスタ, Yレジスタ, Zレジスタ

R26～R31レジスタには通常用途の使用にいくつかの追加機能があります。これらのレジスタはデータ空間の間接アドレス指定用の16ビット アドレス ポインタです。3つのX,Y,Z間接アドレスレジスタは図7-3.で記載したように定義されます。

種々のアドレス指定動作種別で、これらのアドレスレジスタは固定変位、自動増加(+1)、自動減少(-1)としての機能を持ちます(詳細については「命令一式手引書」をご覧ください)。

図7-3. X,Y,Zレジスタ構成図

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

## 7.5. 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。その場合、SPHレジスタは存在しません。

### 7.5.1. 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	

## 7.6. 命令実行タイミング

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

図7-4.はハード構造と高速アクセスレジスタファイルの概念によって可能とされる並列の命令取得と命令実行を示します。これは機能対費用、機能対クロック、機能対電源部に関する好結果と対応するMHzあたり1 MIPSを達成するための基本的なパイプラインの概念です。

図7-5.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7-4. 命令の取得と実行の並列動作

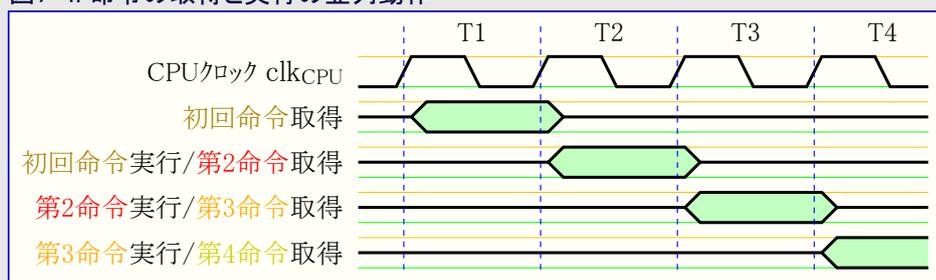
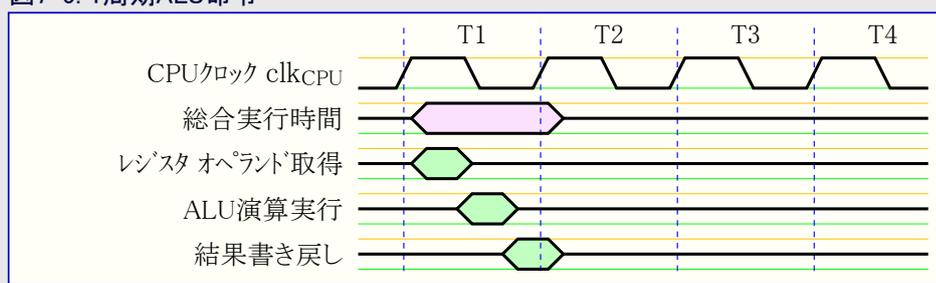


図7-5. 1周期ALU命令



## 7.7. リセットと割り込みの扱い

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

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34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。

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

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

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

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

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

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

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。

### 7.7.1. 割り込み応答時間

許可した全てのAVR割り込みに対する割り込み実行応答は最小4クロック周期です。4クロック周期後、実際の割り込み処理ルーチンに対するプログラムベクタアドレスが実行されます。この4クロック周期時間中にプログラムカウンタ(PC)がスタック上に保存(プッシュ)されます。このベクタは標準的に割り込み処理ルーチンへの無条件分岐で、この分岐は3(訳補:これはJMP命令=3でATmega16HVAの場合、ATmega8HVAの場合はRJMP命令=2)クロック周期要します。複数周期命令実行中に割り込みが起こると、その割り込みが扱われる前に、この命令が完了されます。MCUが休止形態の時に割り込みが起こると、割り込み実行応答時間は4クロック周期増やされます。この増加は選んだ休止形態からの起動時間に加えてです。

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

## 8. AVR メモリ

### 8.1. 概要

この節はATmega8HVA/16HVA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ATmega8HVA/16HVA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。

### 8.2. 実装書き換え可能なプログラム用フラッシュメモリ

ATmega8HVA/16HVAはプログラム保存用に実装書き換え可能な8K/16K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4K/8K×16ビットとして構成されます。

フラッシュメモリは最低10,000回の消去/書き込み回数の耐久性があります。ATmega8HVA/16HVAのプログラムカウンタ(PC)は12/13ビット幅、従って4K/8Kプログラムメモリ位置のアドレス指定です。98頁の「メモリプログラミング」はフラッシュ直列プログラミングの詳細な記述を含みます。

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

命令の取得と実行のタイミング図は9頁の「命令実行タイミング」で示されます。

図8-1. プログラムメモリ配置図



### 8.3. データ用SRAMメモリ

図8-2はATmega8HVA/16HVAのSRAMメモリ構成方法を示します。

ATmega8HVA/16HVA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768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512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動作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。

ATmega8HVA/16HVA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、512バイトのデータ用内蔵SRAMはこれら全てのアドレス指定動作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8-2. 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 (512×8)	\$0100 ～ \$02FF	\$0100 ～ \$02FF

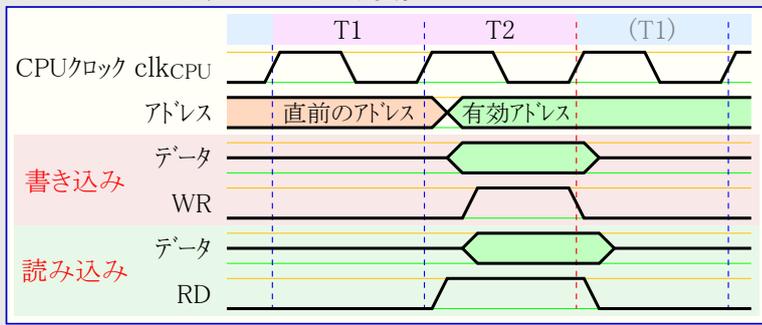
注: 赤字はI/Oアドレス

#### 8.3.1. データメモリアクセスタイミング

本節は内部メモリアクセスに対する一般的なアクセスタイミングの概念を記述します。データ用内蔵SRAMアクセスは図8-3.で記載されるように2 clkCPU周期で実行されます。

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

図8-3. データ用内蔵SRAMアクセス周期



## 8.4. データ用EEPROMメモリ

ATmega8HVA/16HVAは256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プログラミングの詳細記述については各々[101頁](#)と[104頁](#)をご覧ください。

### 8.4.1. EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROMの書き込み(訳注:原文はアクセス)時間は[表8-1](#)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。

#### 8.4.A. 非分離バイトプログラミング (訳注: 共通性のため8.4.A.~D.頁を追加しました。)

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8-1](#)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。

#### 8.4.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。

#### 8.4.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8-1](#)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。

#### 8.4.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8-1](#)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。

## 8.5. I/O メモリ (レジスタ)

ATmega8HVA/16HVAのI/O空間定義は[118頁](#)の「[レジスタ要約](#)」で示されます。

ATmega8HVA/16HVA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とき、これらのアドレスに\$20が加算されなければなりません。ATmega8HVA/16HVA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。

### 8.5.1. 汎用I/Oレジスタ

ATmega8HVA/16HVA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。

## 8.6. メモリ関係レジスタ

### 8.6.1. EEAR – EEPROMアドレスレジスタ (EEPROM Address Register)

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

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

EEPROMアドレスレジスタ(EEAR)は256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0~255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。

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

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

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

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

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

ビット	7	6	5	4	3	2	1	0	
\$1F (\$3F)	-	-	EEPМ1	EEPМ0	EERIE	EEMPE	EEPE	EERE	EECR
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	不定	0	0	不定	0	

#### • ビット7,6 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

#### • ビット5,4 – EEPМ1,0 : EEPROMプログラミング種別 (EEPROM Programing 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8-1.で示されます。EEPEが設定(1)されている間はEEPМnへのどの書き込みも無視されます。リセット中、EEPМn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8-1. EEPROMプログラミング種別 (fosc=4MHz)

EEPМ1	EEPМ0	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)は選んだアドレスのEEPROMをプログラムします。EEMPEが0なら、EEPE設定(1)は無効です。EEMPEがソフトウェアによって設定(1)されてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。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 – EEPEn : 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)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、さもないとEEPROM書き込み(消去)は行われません。EEPROMを書くとき、次の手順に従うべきです(手順2.と3.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. 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)
3. 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)
4. 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。
5. 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。

**警告:** 手順4.と5.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。

**警告:** EEPROM書き込み中の低電圧検出(BOD)リセットは進行中の操作結果を無効にします。

• **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への読み込みスローブです。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らない。EEPROM読み出しアクセスは(その)1命令で行われ、要求したデータは直ちに利用できます。EEPROMが読まれるとき、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8-2.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8-2. EEPROM書き込み時間

項目	校正付き内蔵RC発振器周期数	代表
EEPROM書き込み(CPU)	13,600	3.4ms

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  EEPC, EEPE          ;EEPROMプログラミング完了ならばスキップ
            RJMP  EEPROM_WR        ;以前のEEPROMプログラミング完了まで待機
;
            LDI   R18, (0<<EEP1) | (0<<EEP0) ;プログラミング種別値取得(本例は非分離)
            OUT  EEPC, R18         ;対応プログラミング種別設定
            OUT  EEPC, R17        ;EEPROMアドレス設定
            OUT  EEDR, R16        ;EEPROM書き込み値を設定
            SBI  EEPC, EEMPE       ;EEPROM主プログラム許可ビット設定
            SBI  EEPC, 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(EEPC & (1<<EEPE));          /* 以前のEEPROMプログラミング完了まで待機 */
    EEPC = (0<<EEP1) | (0<<EEP0);     /* 対応プログラミング種別設定 */
    EEPC = uiAddress;                 /* EEPROMアドレス設定 */
    EEDR = ucData;                    /* EEPROM書き込み値を設定 */
    EEPC |= (1<<EEMPE);               /* EEPROM主プログラム許可 */
    EEPC |= (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  EEPC, 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  EEPC, R17        ;EEPROMアドレス設定
            SBI  EEPC, 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(EEPC & (1<<EEPE));          /* 以前のEEPROMプログラミング完了まで待機 */
    EEPC = uiAddress;                 /* EEPROMアドレス設定 */
    EEPC |= (1<<EERE);               /* EEPROM読み出し開始 */
    return EEDR;                      /* EEPROM読み出し値を取得,復帰 */
}
```

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

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

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

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

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

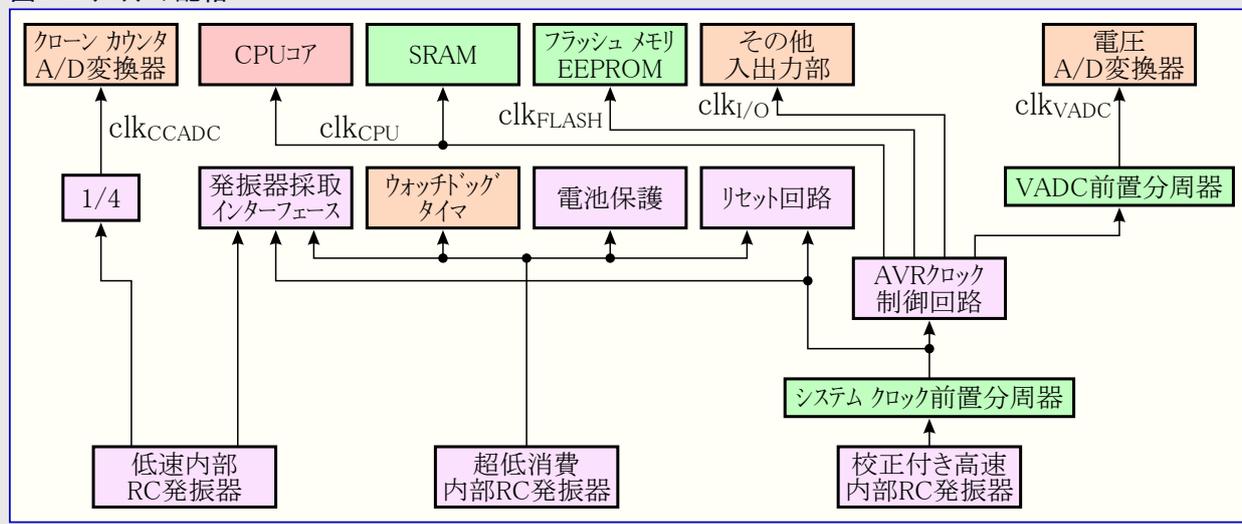
ビット	7	6	5	4	3	2	1	0	
\$1E (\$3E)	(MSB)							(LSB)	GPIOR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

## 9. システム クロックとクロック任意選択

### 9.1. クロック系統とその配給

図9-1.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、23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9-1. クロックの配給



#### 9.1.1. CPU クロック - $clk_{CPU}$

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

#### 9.1.2. I/O クロック - $clk_{I/O}$

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

#### 9.1.3. フラッシュ クロック - $clk_{FLASH}$

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

#### 9.1.4. 電圧A/D変換クロック - $clk_{VADC}$

電圧A/D変換器は専用のクロック範囲を提供されます。VADCクロックはVADC前置分周器によってシステムクロック前置分周器設定に比例して前置分周され、1MHz固定のVADCクロックを生じます。

#### 9.1.5. クーロンカウンタA/D変換クロック - $clk_{CCADC}$

クーロンカウンタA/D変換器は専用のクロック範囲を提供されます。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。

#### 9.1.6. 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提供されます。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。

## 9.2. クロック元

以降の項はデバイスで利用可能なクロック元を記述します。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。

## 9.3. 校正付き高速RC発振器

校正された高速RC発振器は既定で8.0MHzクロックをシステム クロック前置分周器へ供給します。この周波数は70°C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。正確な時間基準と走行時校正の使用により、この発振器は全温度範囲に渡って±1%の精度に校正することができます。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。70°Cで、この校正は8MHz±4%の周波数を与えます。この発振器はFOSCCALレジスタを変更することによって7.3~8.1MHz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95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。システムクロックの周波数が19頁の「システム クロック前置分周器」によって与えられることに注意してください。

チップの起動時間は前置分周された高速RC発振器周期(CK)数とULP発振器周期数として参照されます。起動時間は表9-1.で定義されるようにSUTヒューズによって選ばれます。

表9-1. 校正付き高速RC発振器用起動遅延時間

SUT2~0	パワーセーブからの起動遅延時間	リセットからの(注2)付加遅延時間
0 0 0	6×CK	14×CK+4ms
0 0 1	6×CK	14×CK+8ms
0 1 0	6×CK	14×CK+16ms
0 1 1	6×CK	14×CK+32ms
1 0 0	6×CK	14×CK+64ms
1 0 1	6×CK	14×CK+128ms
1 1 0	6×CK	14×CK+256ms
1 1 1	(注1) 6×CK	14×CK+512ms

注1: デバイスはこの選択で出荷されます。

注2: 実付加値、選択可能な4~512ms遅延は「超低電力RC発振器」の実際の周波数に依存します。次頁の表9-2.と111頁の「電気的特性」をご覧ください。

## 9.4. 低速RC発振器

低速RC発振器は131kHzクロック(代表値、詳細については111頁の「電気的特性」を参照)を供給します。このクロックは高速RC発振器の走行時校正用の基準タイミングとして、そして実際のULP発振器周波数を正確に測定するのに使うことができます。詳細については19頁の「OSI - 発振器採取インターフェース」を参照してください。低速RC発振器はクーロン カウンタADC用クロックも供給します。

実際の低速RC発振器周波数は工程変化と温度に依存します。111頁の「電気的特性」をご覧ください。タイミング基準として使う時に非常に良好な精度を提供するために低速RC発振器は識票アドレス空間に格納された予測バイトを持っています。詳細については95頁の「ソフトウェアからの識票列読み出し」項を参照してください。温度の関数としてのμsでの実際のクロック周期は次式によって与えられます。

$$\text{低速RC発振器周期} = \frac{\text{低速RC発振器周期} - \text{低速RC発振器温度予測語} - \frac{(T - T_{HOT})}{64}}{1024}$$

ここでTはケルビン温度でのダイ温度で、T<sub>HOT</sub>は識票列に格納された校正温度です。“低速RC発振器周期”パラメータはAtmel製造で測定した実際の低速RC発振器周期についての情報を保持します。このパラメータは識票アドレス空間から読むことができます。実際の低速RC発振器周波数は-10~+70°Cの温度範囲に渡って1%未満の誤差を得ることができます。ダイ温度は電圧ADCを使って得ることができます。詳細については73頁の「電圧A/D変換器(5チャンネル汎用12ビットΔΣA/D変換器)」項を参照してください。低速RC発振器周波数予測の例に関してはAVR351応用記述を参照してください。

## 9.5. 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(代表値)を供給します。この発振器はウォッチドッグ タイマと電池保護部用のクロックを提供します。実際のULP発振器周波数は工程変化と温度に依存します。111頁の「電気的特性」をご覧ください。この発振器は全ての動作種別で自動的に許可されます。これはリセットの間も許可されます。ULP発振器の実際のクロック周期を測定するには2つの代替方法があります。

1. ダイ温度の関数として正確なクロック周期を決めるには、応用によって必要とされるなら、発振器採取インターフェースが使われるべきです。詳細については19頁の「OSI - 発振器採取インターフェース」を参照してください。ULP RC発振器周波数計算の例に関してはAVR351応用記述を参照してください。
2. ダイ温度と無関係に実際のクロック周期に対する固定値を決める、例えば電池保護タイミングの最良設定を決めるには、識票アドレス空間に格納された“超低電力RC発振器周波数”を使ってください。詳細については95頁の「ソフトウェアからの識票列読み出し」項を参照してください。

## 9.6. 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。

## 9.7. ウォッチドッグ タイマ、電池保護、クーロン カウンタA/D変換のクロック

ウォッチドッグ タイマ、電池保護、クーロン カウンタA/D変換器(CC-ADC)のクロックは超低電力RC発振器です。この発振器は全ての動作種別で自動的に許可されます。リセット中も許可されます。

## 9.8. クロック起動手順

CPUがパワーセーブから起動するとき、命令実行を始める前の安定なクロックを保證する起動時間にCPUクロック元が使われます。CPUがリセットから始まる時、電圧調整器が通常動作を始める前に安定なレベルに達するのを可能にするのが付加遅延です。超低電力RC発振器は起動時間のこの実時間部のタイミングに使われます。起動時間は18頁の表9-1.で示されるようにSUTヒューズによって決められます。各時間経過に使われる超低電力RC発振器周期数は表9-2.で示されます。

表9-2. 超低電力RC発振器周期数

代表時間経過 (注)	周期数
4 ms	512
8 ms	1K
16 ms	2K
32 ms	4K
64 ms	8K
128 ms	16K
256 ms	32K
512 ms	64K

注: 実際の値は超低電力RC発振器の実際のクロック周期に依存します。詳細については18頁の「超低電力RC発振器」を参照してください。

## 9.9. クロック出力

2分周されたCPUクロックをPB0ピンに出力することができます。CPUはMCU制御レジスタ(MCUCR)のクロック出力許可(CKOE)ビットを設定(1)することによってクロック出力機能を許可にできます。このクロックはどの休止形態でも動作しません。

## 9.10. システム クロック前置分周器

ATmega8HVA/16HVAは校正付き高速RC発振器を前置分周するのに使われるシステム クロック前置分周器を持ちます。システム クロックはクロック前置分周レジスタ(CLKPR)の設定によって分周することができます、これは電力消費と処理能力の変化に対する必要条件としてシステム クロック周波数の増減を使用者に許します。このシステム クロックはCPUと全同期周辺機能のクロック周波数に影響を及ぼします。clkCPU、clkFLASH、clkI/Oは22頁の表9-3.で示された値によって分周されます。

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

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

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

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

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

## 9.11. 電圧A/D変換器(VADC)クロック前置分周器

VADCクロックはシステム クロック前置分周器設定に比例して自動的に前置分周されます。「システム クロック前置分周器」をご覧ください。クロック分周値選択(CLKPS1,0)ビットに依存して、VADCクロック(clkVADC)は22頁の表9-4.で示されるように1,2,4分周されます。

## 9.12. OSI – 発振器採取インターフェース

### 9.12.1. 特徴

- 走行時に選択可能な発振器入力(低速RCまたはULP RC発振器)
- 選んだ発振器の7ビット前置分周
- 前置分周したクロックの位相へのソフトウェア読み込みアクセス
- タイマ/カウンタ0用捕獲起動元

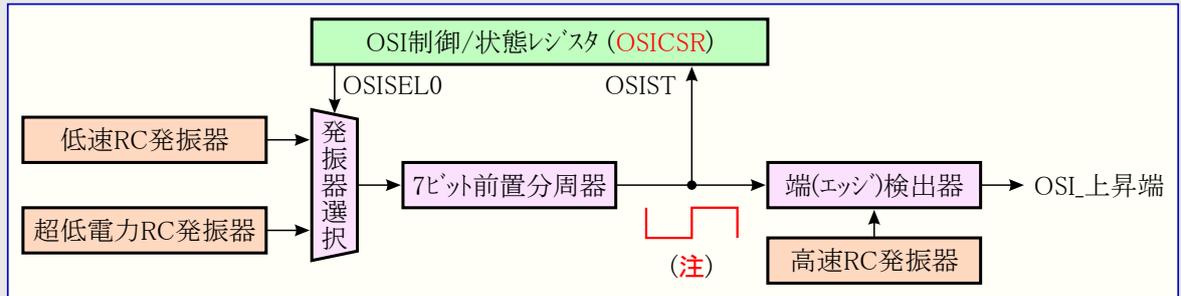
### 9.12.2. 概要

発振器採取インターフェース(OSI)はATmega8HVA/16HVAの低速RCと超低電力(ULP)RC発振器の採取を可能にします。OSIは高速RC発振器走行時に高い精度で校正するのに使うことができます。OSIはULP発振器周波数偏移補償用の正確な基準を提供することもできます。

前置分周された発振器位相はOSI制御/状態レジスタ(OSICSR)を通してCPUによって継続的に読むことができます。加えて、タイマ/カウンタ0の捕獲機能は前置分周したクロックの上昇端での起動に設定することができます。これは高速RC発振器に比例する発振器周波数の正確な測定を可能にします。

発振器採取インターフェースの簡単化した構成図は図9-2.で示されます。

図9-2. 発振器採取インターフェース構成図



注: 前置分周された低速RC/ULP発振器1周期は実際の低速RC/ULP発振器周期の128倍に対応します。

“OSI\_上昇端”信号は前置分周した低速RC/ULP発振器クロックの各上昇端でのパルスです。この信号はCPUによって直接アクセスできませんが、タイマ/カウンタ0の捕獲機能を起動するのに使うことができます。タイマ/カウンタ0の捕獲機能との組み合わせでのOSI使用は最小CPU計算での発振器周波数の正確な測定を容易にします。捕獲機能の許可方法の詳細に関しては50頁の「タイマ/カウンタ (T/C0,T/C1)」を参照してください。

### 9.12.3. 使用法

低速RC発振器は全温度範囲に渡って高く予想可能で正確なクロック元に相当し、高速RC発振器走行時校正用の優秀な基準を提供します。代表的に、温度範囲全体での非同期直列通信に対して正確な高速RC発振器周波数を提供するために実行時校正が必要とされます。クーロン カウンタA/D変換(CC-ADC)測定間の累積充電計算のために正確な時間基準も必要とされます。

$T_{HOT}$ (校正温度)での低速RC発振器周波数と低速RC発振器温度係数は識票列に格納されています。 $T_{HOT}$ の値も識票列に格納されています。これらの特性は与えられた温度で実際の低速RCクロック周期を高精度で計算するのに使うことができます。詳細に関しては18頁の「低速RC発振器」を参照してください。

1つ以上の前置分周された低速RCクロック周期のCPU周期数を測定することによって、実際の高速RC発振器クロック周期を測定することができます。そして高速RCクロック周期は高速RC発振器校正(FOSCCAL)レジスタへの書き込みによって調節することができます。校正後の新しい高速RCクロック周期は測定を繰り返すことによって検証され、必要ならば校正を繰り返すべきです。低速RCクロック周期の関数としての高速RCクロック周期は次式で与えられます。

$$T_{FastRC} = T_{SlowRC} \times \frac{128 \times n}{\text{前置分周された低速RCクロック}n\text{周期でのCPU周期数}}$$

ここでの $n$ は測定に使われた前置分周した低速RCの周期数です。より多くの前置分周された低速RC周期は測定誤差を減らしますが、校正に対する消費時間を増やします。低速RC発振器がOSI部によって許可された後の安定に非常に短い時間を必要とすることに注目してください。従って、校正方法は校正用の基準時間として最初と2番目の“OSI\_上昇端”間の時間を使えるかもしれません。

OSIの別な使用法はULP周波数の正確な測定です。 $T_{HOT}$ (校正温度)でのULP周波数とULP温度係数は識票列に格納されており、ULP周波数の直接計算を可能にします。けれども、ULP周波数は温度に対して低速RC発振器周波数より低い予測可能性で、従ってより高い精度の結果は低速RCとULP発振器間の比率を計算することによって得られます。これはULPと低速RC発振器の両方を採取してその結果を比較することによって行われます。この比率が既知のとき、実際のULP周波数は高精度で決めることができます。低速RCクロック周期の関数としてのULP RCクロック周期は次式によって与えられます。

$$T_{ULPRC} = T_{SlowRC} \times \frac{\text{前置分周されたULP RCクロック}n\text{周期でのCPU周期数}}{\text{前置分周された低速RCクロック}n\text{周期でのCPU周期数}}$$

ここでの $n$ は測定に使われた前置分周したULP RCと低速RCの周期数です。より多くの前置分周されたULP RCと低速RC周期は測定誤差を減らしますが、校正に対する消費時間を増やします。正確な結果を保証するためにこの操作の間にFOSCCALレジスタが一定値で維持されなければならないことに注意してください。

これらのクロック周期計算は直前の校正のためにダイ温度で意味ある変化がある時に再び実行されるべきです。ダイ温度は電圧A/D変換を使って得られます。詳細に関しては73頁の「電圧A/D変換器 (5チャンネル汎用12ビット $\Sigma$ A/D変換器)」項を参照してください。

## 9.13. クロック関係レジスタ

### 9.13.1. 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値								

#### • ビット7~0 – FCAL7~0 : 高速RC発振器校正値 (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、70°Cで8MHzの発振器周波数を与える工場校正値が、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7.3~8.1MHz範囲のどの周波数にも走行時校正することができます。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、8.1MHz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。

FCAL7~5ビットは発振器に対する操作範囲を決めます。これらのビットの'000'設定は最低周波数側範囲になり、'111'設定は最高周波数側範囲になります。周波数範囲は重なっています。例えばFOSCCAL=\$1F設定はFOSCCAL=\$20設定より高い周波数になります。

FCAL4~0ビットは選んだ範囲内の周波数調整に使われます。\$00設定はその範囲の最低周波数になり、\$1F設定はその範囲の最高周波数になります。FCAL4~0での1増加は7.3~8.1MHz周波数範囲で1.5%未満の周波数増加になります。正確な時間基準とで校正後に±1%の発振器精度が達成できます。周波数は温度で偏移し、故に精度を維持するために走行時校正が必要とされるでしょう。詳細に関しては19頁の「OSI – 発振器採取インターフェース」を参照してください。

識票列で得られる既定FOSCCAL値は区分の半分より低くなるように選ばれています(高速RC発振器の代表特性に関しては117頁の図30-1をご覧ください)。従って温度範囲全体に渡る周波数校正は既定区分と1つ下の区分で充分です。2つの区分間を移る時の大きな周波数変化を避けるために識票列にFOSC\_SEGMENT値が格納されています。これは既定区分の最低値よりも低い周波数を与える(1つ下の区分の)最初のFOSCCAL値で、高速RC発振器校正時に使われるべきです。

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

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

#### • ビット5 – CKOE : クロック出力許可 (Clock Output)

このビットが1を書かれると、2分周されたCPUクロックがPB0ピンに出力されます。

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

ビット (\$61)	7	6	5	4	3	2	1	0	
	CLKPCE	-	-	-	-	-	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	1	1	

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

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

#### • ビット1,0 – CLKPS1,0 : クロック分周値選択 (Clock Prescaler Select Bits 1~0)

これらのビットは選んだクロック元と内部システムクロック間の分周値を定義します。これらのビットは応用の必要条件に合わせた各種クロック周波数を実行時に書けます。分周値が使われると、分周器はMCUへの主クロックを分周し、全ての同期周辺機能の速度が減じられます。分周値は表9-3.で与えられます。システムクロック前置分周選択ビットへの書き込みがどんな実行中の電圧A/D(VADC)変換も中止することに注意してください。

表9-3. システム クロック前置分周器選択

CLKPS1	CLKPS0	クロック分周
0	0	(予約) (注)
0	1	2分周
1	0	4分周
1	1	8分周

注: 予約値はCLKPS1,0に書かれるべきではありません。

表9-4. 電圧A/D変換器クロック前置分周 (注)

CLKPS1	CLKPS0	VADCクロック分周
0	0	(予約)
0	1	4分周
1	0	2分周
1	1	1分周

注: 前置分周値変更時、VADC前置分周器はVADCクロック周波数を自動的に変更してどんな実行中の変換も中止します。

#### 9.13.4. OSICSR – 発振器採取インターフェース制御/状態レジスタ (Oscillator Sampling Interface Control and Statud Register)

ビット	7	6	5	4	3	2	1	0	
\$17 (\$37)	–	–	–	OSISEL0	–	–	OSIST	OSIEN	OSICSR
Read/Write	R	R	R	R/W	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~5,3,2 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

- ビット4 – OSISEL0 : 発振器採取インターフェース選択0 (Oscillator Sampling Interface Select 0)

表9-5. OSISEL0ビット内容

OSISEL0	発振器元
0	ULP発振器
1	低速RC発振器

- ビット1 – OSIST : 発振器採取インターフェース状態フラグ (Oscillator Sampling Interface Status)

このビットは前置分周されたクロックの位相を継続的に表示します。このビットは前置分周されたクロックの上昇端と下降端を測定するためにCPUによってポーリングすることができます。

- ビット0 – OSIEN : 発振器採取インターフェース許可 (Oscillator Sampling Interface Enable)

このビットの設定(1)が発振器採取インターフェースを許可します。このビットが解除(0)されると、発振器採取インターフェースは禁止されます。

注: OSICSRレジスタが書かれる度毎に前置分周器がリセットし、従ってその度毎に新しい発振器元が選ばれます。

## 10. 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。

### 10.1. 休止形態種別

17頁の図9-1はATmega8HVA/16HVAの各種クロック系統とその配給を示します。この図は適切な休止形態を選ぶ助けになります。各種休止形態とそれらの起動元は表10-1.で要約され、図10-1.が休止形態状態図を示します。

表10-1. 休止形態に対する復帰起動元

休止種別	調整器電流での起動	電池保護	外部割り込み	ウォッチドッグタイマ	EEPROM操作可	電圧調整器監視器	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	○	○	○	○	○	○	②	○	○	
A/D変換雑音低減	○	○	○	○	○	○	②	○		
パワーセーブ	○	○	○	○			②			
パワーオフ										①

①：充電器検出が活性(有効)であるには放電FETがOFFにされていなければなりません。

②：瞬時と累積の変換完了起動だけです。

休止形態の何れかへ移行するには**休止形態制御レジスタ(SMCR)**の**休止許可(SE)ビット**が論理1を書かれ、**SLEEP**命令が実行されなければなりません。SMCRの**休止種別選択(SM2~0)ビット**は**SLEEP**命令によって活性(有効)にされる休止形態のどれかを選びます。一覧については表10-3.をご覧ください。

MCUが休止形態中に許可した割り込みが起こると、MCUは起動します。その時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ると、MCUは起動し、リセットベクタから実行します。

図10-1. 休止形態状態遷移図

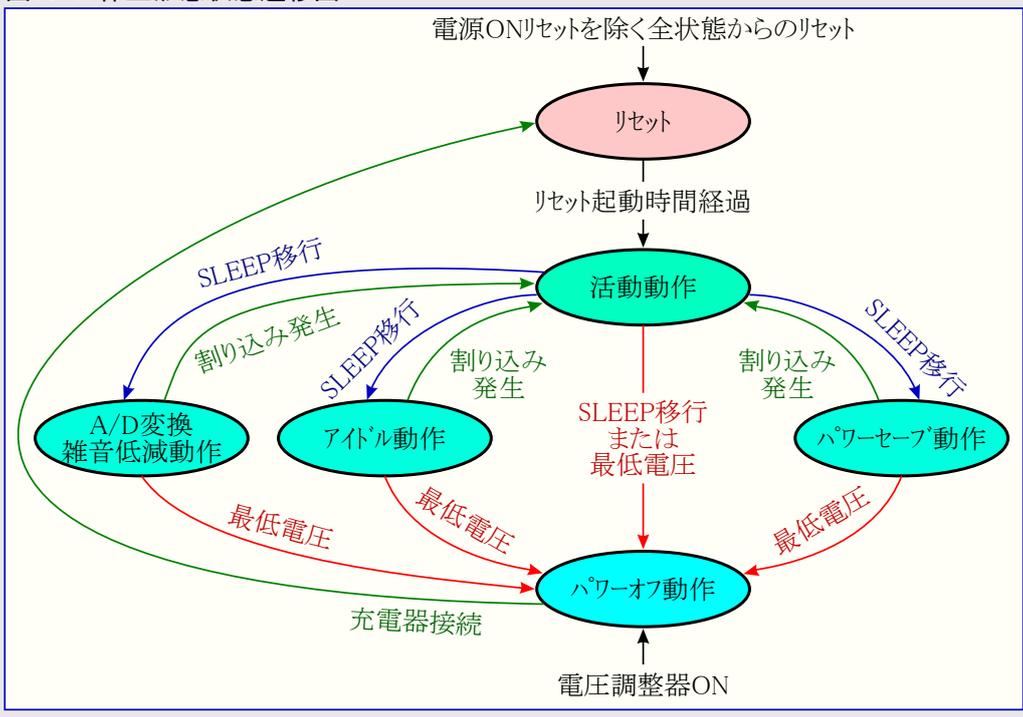


表10-2. 各休止形態に於ける活動単位部

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	パワーオフ
高速RC発振器	○	○	○	②	
超低電力RC発振器	○	○	○	○	
低速RC発振器	③	③	③	③	
CPU	○				
フラッシュメモリ	○				
タイマ/カウンタ	○	○			
直列周辺インターフェース(SPI)	○	○			
電圧A/D変換器(V-ADC)	○	○	○		
クーロンカウンタA/D変換器(CC-ADC)	○	④	④	④	
外部割り込み	○	○	○	○	
電池電流保護(CBP)	○	○	○	○	
ウォッチドッグタイマ	○	○	○	○	
電圧調整器	○	○	○	○	
充電器検出 (注1)	○	○	○	○	○
電圧調整器監視器	○	○	○		
発振器採取インターフェース(OSI)	○	○			

(注1) : 充電器検出が許可されるには放電FETがOFFされなければなりません。

- ② : 深下電圧回復(DUVR)動作許可の場合、パワーセーブ動作で高速RC発振器が動作します。また、充電FETと放電FETが許可された後、概ね125ms間も動作します。
- ③ : 低速RC発振器はクーロンカウンタA/D変換器(CC-ADC)が許可されている場合、またはこの発振器が発振器採取インターフェース(OSI)の入力として選ばれて且つ採取が許可されている時にだけ動作します。
- ④ : 瞬時と累積の変換完了起動だけです。

## 10.2. 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。

## 10.3. 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、電圧A/D変換器(VADC)、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)、超低電力(ULP)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。

## 10.4. パワーセーブ動作

休止種別選択(SM2~0)ビットが'011'を書かれると、SLEEP命令はMCUをパワーセーブ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、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)、超低電力(ULP)RC発振器、低速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ブ動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。詳細については36頁の「外部割り込み」を参照してください。

パワーセーブ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。

## 10.5. パワーオフ動作

休止種別選択(SM2~0)ビットが'110'を書かれると、SLEEP命令はCPUに対して電圧調整器を停止させ、充電器検出回路の動作だけを残したままにさせます。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。パワーオフ動作から復帰するとき、MCUはリセットします。

**注:** 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は禁止されるべきです。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。

## 10.6. 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(27頁の「電力削減レジスタ(PRR0)」をご覧ください)。周辺機能は現状で固定化され、I/Oレジスタは書けません。クロックを停止している時に周辺機能によって使われていた資源は占有されたままです。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。

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

### 10.7.1. 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セーブ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30頁の「ウォッチドッグ タイマ」をご覧ください。

### 10.7.2. 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( $clk_{I/O}$ )と電圧A/D変換クロック( $clk_{VADC}$ )の両方が停止される休止形態ではデバイスの入力緩衝部が禁止されます。これは必要とされない時に入力論理回路によって電力が消費されないことを保証します。いくつかの場合で入力論理回路は起動条件を検出するために必要とされ、その時は許可されます。どのピンが許可されるかの詳細については、43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VREG/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VREG/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76頁の「DIDR0 - デジタル入力禁止レジスタ」を参照してください。

### 10.7.3. 内蔵デバッグ機能

プログラム(0)されたDWENヒューズはクロック系のいくつかの部分に全休止形態での動作を許可します。これは休止の間の消費電力を増加するでしょう。従って、デバッグWIREが使われない時にDWENヒューズは禁止されるべきです。

### 10.7.4. 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。電池保護回路の消費電流はパワーセーブ動作でだけ重要です。両FETの禁止は消費電力最小化のために電池保護単位部を自動的に禁止します。電池保護が許可される時は必ず、バンドギャップ基準電圧が常に許可されるべきです。

### 10.7.5. 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。節電のため、使われないとき、パワーセーブ休止動作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73頁の「電圧A/D変換器 (5チャネル汎用12ビット $\Delta\Sigma$ A/D変換器)」をご覧ください。

### 10.7.6. 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、低速RC発振器の走行を維持します。節電のため、使われない時に禁止、または定常電流検出動作に設定されるべきです。クーロン カウンタA/D変換器操作の詳細については67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。

### 10.7.7. バンドギャップ基準電圧

許可なら、バンドギャップ基準電圧は休止形態と無関係に電力を消費します。節電のために電圧A/D変換器、クーロン カウンタA/D変換器、または電池保護に対する基準電圧として使われない時に禁止されるべきです。詳細については77頁の「基準電圧と温度感知器」をご覧ください。

### 10.7.8. FET駆動部

パワーセーブ動作での電力消費最小化のため、高速RC発振器が停止されるのを保証するためにFET駆動部の深下電圧回復(DUVR)動作は禁止されるべきです。

## 10.8. 電力管理用レジスタ

### 10.8.1. 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として読みます。

- 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10-3.で示される利用可能な4つの休止形態の1つを選びます。

- ビット0 – SE : 休止許可 (Sleep Enable)

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

表10-3. 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	(予約)
0	1	1	パワーセーブ動作
1	0	0	パワーオフ動作
1	0	1	(予約)
1	1	0	(予約)
1	1	1	(予約)

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

ビット	7	6	5	4	3	2	1	0	
(\$64)	-	-	PRVRM	-	PRSPI	PRTIM1	PRTIM0	PRVADC	PRR0
Read/Write	R	R	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,6,4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。将来のデバイスとの互換性のため、PRR0が書かれる時にこれらのビットは0が書かれなければなりません。

- ビット5 – PRVRM : 電圧調整監視器電力削減 (Power Reduction Voltage Regulator Monitor)

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電圧調整監視器を停止します。

- ビット3 – PRSPI : 直列周辺インターフェース電力削減 (Power Reduction SPI)

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直列周辺インターフェース(SPI)を停止します。SPIの再起動時、SPIは正しい動作を保証するために再初期化されるべきです。

- ビット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 : 電圧A/D変換器電力削減 (Power Reduction V-ADC)

このビットへの1書き込みは電圧A/D変換器(V-ADC)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。

注: 例えPRVADCビットが設定(1)されていても、V-ADC制御レジスタは更新することができます。

## 11. システム制御とリセット

### 11.1. 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。ATmega16HVAについてリセットベクタに配置される命令は、きつとリセット処理ルーチンへの**JMP**(絶対分岐)命令でしよう。ATmega8HVAについてリセットベクタに配置される命令は、きつとリセット処理ルーチンへの**RJMP**(相対分岐)命令でしよう。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、割り込みベクタがブート領域の場合やその逆も同様です。図11-1.の回路構成図はリセット論理回路を示します。114頁の表29-6.はリセット回路の電気的特性を定義します。

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

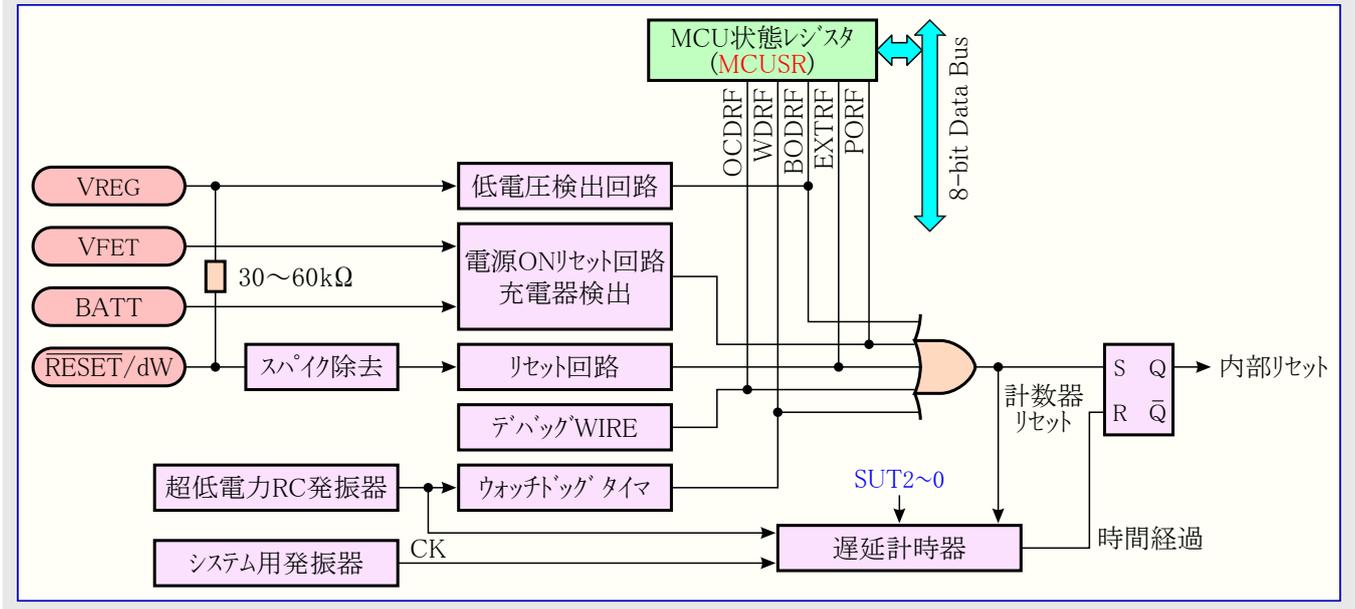
全てのリセット元が無効にされてしまった後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電圧調整器に許します。遅延タイマの遅延時間は**SUTヒューズ**を通して使用者によって定義されます。この遅延時間についての各種選択は18頁の「**クロック元**」で示されます。

### 11.2. リセット元

ATmega8HVA/16HVAには次の各種リセット元があります。

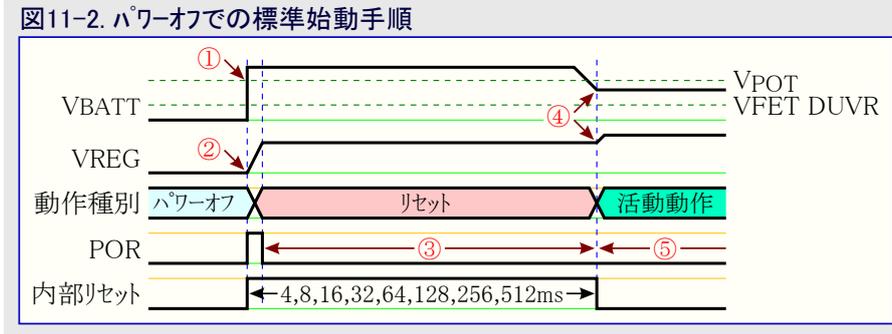
- **電源ONリセット** …… チップが**パワーオフ**動作で、充電器が接続されると、充電器検出部はリセットパルスを生成します。詳細については29頁の「**電源ONリセットと充電器接続**」をご覧ください。
- **外部リセット** …… **RESET**ピンに**最小パルス幅**より長く**Low**レベルが存在すると、MCUはリセットされます。詳細については29頁の「**外部リセット**」をご覧ください。
- **ウォッチドッグリセット** …… ウォッチドッグシステムリセット動作が許可され、ウォッチドッグタイマ時間が終了すると、MCUはリセットされます。詳細については29頁の「**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(VBOT)**以下のとき、MCUはリセットされます。詳細については30頁の「**低電圧検出**」をご覧ください。
- **デバッグWIREリセット** …… チップ内デバッグ動作でのデバッグWIREリセットはリセット命令を与える時にMCUをリセットします。

図11-1. リセット回路構成



## 11.2.1. 電源ONリセットと充電器接続

電圧調整器は充電器検出部が電圧調整器を許可するまで始動しません。これが起きる前のチップはパワーオフ動作で充電器検出部だけが許可されています。充電器検出部が電圧調整器を許可するためには、BATTピン電圧が電源ON閾値電圧(VPOT)を越えていなければなりません。BATTピン電圧がVPOTを越える時に電圧調整器が始動してチップがリセット動作へ移行します(図11-2.参照)。遅延カウンタが計時を終了すると、チップは活動動作へ移行します(図11-1.参照)。



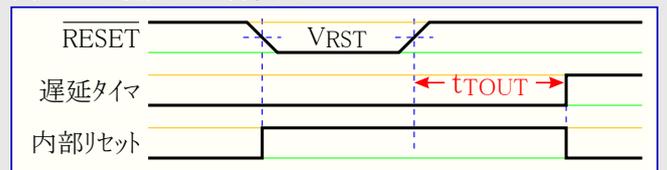
- ①: 充電器電圧がBATTピンを電源ON閾値電圧(VPOT)以上に引き上げます。
- ②: VBATTがVPOT以上に上ると、ATmega8HVA /16HVAは電圧調整器をONにし、VREGは上昇を開始します。PORリセットはVREG上昇と同時にHighとなり、チップの内部リセット状態を開始します。外部FETはOFFに初期化されます。
- ③: 内部リセットはPORリセットがLowになった後、 $t_{TOUT}$ で与えられた時間、Highを保持します。28頁の「システム制御とリセット」をご覧ください。チップがリセットの間にVREF校正レジスタがそれらの既定値でリセットされます。VREGとBOD基準は両方共VREF電圧を参照します。従って、リセットに於いてこれら全ての電圧水準が既定値になります。両FETはこの状態で相互補完に切り換えられます。
- ④: 内部リセットがLowになると直ぐに、チップは深下電圧回復(DUVR)動作での動作を開始します(DUVR動作の詳細に関しては91頁の「DUVR - 深下電圧回復動作」とAVR354応用記述をご覧ください)。DUVR動作では114頁の表29-5.で指定されたVFETレベルによって与えられるVFETピンでの電圧を得るために、FET駆動部が充電FETのゲート電圧を制御します。これはBATT電圧を減少させます。セル電圧がVFET\_DUVRレベルより低い間、DUVR動作がVFET電圧を調整するだけです。高いセル電圧に対して、DUVR動作は何の影響もありません。DUVR動作はチップが活動動作へ移行すると直ぐにソフトウェアによって禁止されるかもしれません。
- ⑤: 内部リセットがLowになると、ソフトウェアが始動して、VREF=1.100Vを得るための値をVREF校正レジスタに格納します。VREF電圧が変更されるので、VREG電圧とVFET\_DUVR電圧はVREFに比例して上昇するでしょう。

今やチップは通常動作ができますが、単一セル応用に対するDUVR動作でのEEPROM書き込みは避けるべきです。

## 11.2.2. 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(114頁の表29-6.参照)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(遅延カウンタを起動し)、遅延計数器は遅延時間( $t_{TOUT}$ )経過後にMCUを始動します。

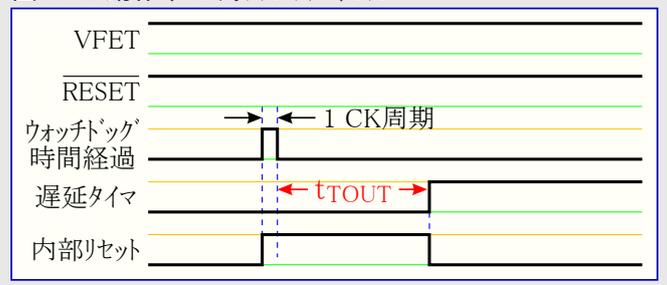
図11-3. 動作中の外部リセット



## 11.2.3. 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、遅延タイムは遅延時間( $t_{TOUT}$ )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30頁を参照してください。

図11-4. 動作中のウォッチドッグ リセット

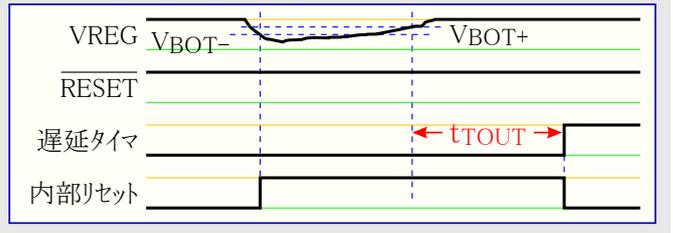


### 11.2.4. 低電圧(ブラウンアウト)検出リセット

ATmega8HVA/16HVAには固定化された起動(検出)電圧(VBOT)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 $VBOT+ = VBOT + VHYST / 2$ と $VBOT- = VBOT - VHYST / 2$ 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11-5のVBOT-)、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11-5のVBOT+)、(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11-5. 動作中の低電圧リセット



### 11.2.5. 電圧消失(Black-out)検出

追加安全機能として、VREGがVBL0T以下に低下した場合にチップは自動的にパワーオフ動作へ移行します。VBL0Tは常にBOD基準(VBOT)以下の適切な値です。

## 11.3. ウォッチドッグ タイマ

### 11.3.1. 特徴

- 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\*

### 11.3.2. 概要

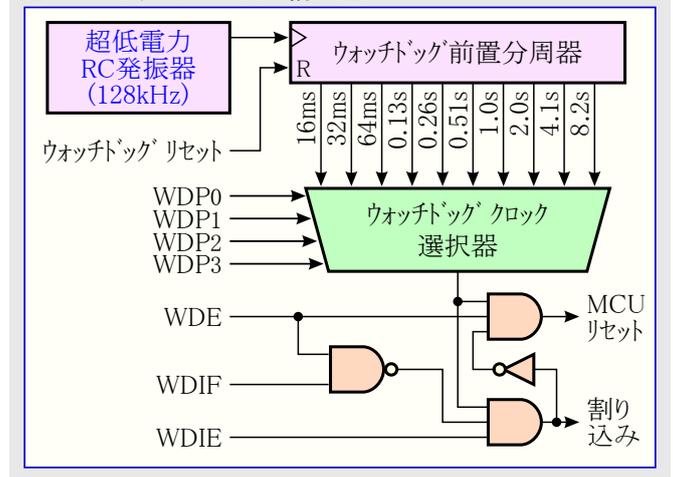
ATmega8HVA/16HVAには強化されたウォッチドッグタイマ(WDT)があります。ウォッチドッグタイマ(WDT)は超低電力発振器の周期で計時するタイマです。WDTは計数器が与えられた計時完了値に達した時に割り込みまたはシステムリセットを生じます。通常動作では計時完了値へ達する前に計数器を再始動するために、システムはウォッチドッグリセット(WDR)命令を使う必要があります。システムが計数器を再始動しなければ、割り込みまたはシステムリセットが起こるでしょう。

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与え、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11-6. 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なってしまう結果になり得るため、ウォッチドッグ タイマはWDPビットのどんな変更にも先立ってリセット(WDR命令)されるべきです。

## 11.4. リセット関係用レジスタ

### 11.4.1. 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)	–	–	–	OCDRF	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として読みます。

- ビット4 – OCDRF : チップ内デバッグ リセット フラグ (OCD Reset Flag)

このビットはデバッグWIREリセットが起きる場合に設定(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。

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

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

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

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

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

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

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

表11-1. ウォッチドッグ タイマ設定

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

注: WDTONヒューズは0でプログラム、1で非プログラムに設定です。

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

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

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

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

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

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

このWDP3~0ビットはウォッチドッグ タイマが走行する時のウォッチドッグ タイマの前置分周を決めます。各種前置分周値と対応する計時完了周期は表11-2.で示されます。

表11-2. ウォッチドッグ前置分周選択

WDP3	0				1						
	0		1		0		1				
WDP2	0		1		0		1				
WDP1	0	1	0	1	0	1	0	1			
WDP0	0	1	0	1	0	1	0	1			
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	0.13s	0.26s	0.51s	1.0s	2.0s	4.1s	8.2s	

注: 実際の計時完了周期は超低電力RC発振器の実際のクロック周期に依存します。詳細に関しては18頁の「超低電力RC発振器」を参照してください。

## 12. 割り込み

### 12.1. 概要

本章はATmega8HVA/16HVA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。

### 12.2. ATmega8HVA/16HVAの割り込みベクタ

表12-1. リセットと割り込みのベクタ

ベクタ番号	プログラム アドレス		発生元	備考
	ATmega8HVA	ATmega16HVA		
1	\$0000	\$0000	リセット	外部ピン, 電源ON, WDT, BOD等の各種リセット
2	\$0001	\$0002	BPINT	電池保護割り込み
3	\$0002	\$0004	VREGMON	電圧調整監視器割り込み
4	\$0003	\$0006	INT0	外部割り込み要求0
5	\$0004	\$0008	INT1	外部割り込み要求1
6	\$0005	\$000A	INT2	外部割り込み要求2
7	\$0006	\$000C	ウォッチドッグ WDT	ウォッチドッグ計時終了
8	\$0007	\$000E	タイマ/カウンタ1 IC1	タイマ/カウンタ1捕獲発生
9	\$0008	\$0010	タイマ/カウンタ1 COMP1	タイマ/カウンタ1比較A一致
10	\$0009	\$0012	タイマ/カウンタ1 COMPB1	タイマ/カウンタ1比較B一致
11	\$000A	\$0014	タイマ/カウンタ1 OV1	タイマ/カウンタ1溢れ
12	\$000B	\$0016	タイマ/カウンタ0 IC0	タイマ/カウンタ0捕獲発生
13	\$000C	\$0018	タイマ/カウンタ0 COMP0A	タイマ/カウンタ0比較A一致
14	\$000D	\$001A	タイマ/カウンタ0 COMP0B	タイマ/カウンタ0比較B一致
15	\$000E	\$001C	タイマ/カウンタ0 OV0	タイマ/カウンタ0溢れ
16	\$000F	\$001E	SPI STC	SPI 転送完了
17	\$0010	\$0020	電圧A/D変換器 VADC	電圧A/D変換完了
18	\$0011	\$0022	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
19	\$0012	\$0024	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
20	\$0013	\$0026	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
21	\$0014	\$0028	EEPROM EE_RDY	EEPROM 操作可

(訳注) 原書の表12-1.と表12-2.は表12-1.として纏めました。

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置けます。

ATmega8HVA/16HVA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/\$0000		RJMP/JMP	RESET ; 各種リセット
\$0001/\$0002		RJMP/JMP	BPINT ; 電池保護割り込み
\$0002/\$0004		RJMP/JMP	VREGMON_INT ; 電圧調整監視器割り込み
\$0003/\$0006		RJMP/JMP	EXT_INT0 ; 外部割り込み要求0
\$0004/\$0008		RJMP/JMP	EXT_INT1 ; 外部割り込み要求1
\$0005/\$000A		RJMP/JMP	EXT_INT2 ; 外部割り込み要求2
\$0006/\$000C		RJMP/JMP	WDT ; ウォッチドッグ計時終了
\$0007/\$000E		RJMP/JMP	TIM1_IC ; タイマ/カウンタ1捕獲発生
\$0008/\$0010		RJMP/JMP	TIM1_COMPA ; タイマ/カウンタ1比較A一致
\$0009/\$0012		RJMP/JMP	TIM1_COMPB ; タイマ/カウンタ1比較B一致
\$000A/\$0014		RJMP/JMP	TIM1_OVF ; タイマ/カウンタ1溢れ
\$000B/\$0016		RJMP/JMP	TIMO_IC ; タイマ/カウンタ0捕獲発生
\$000C/\$0018		RJMP/JMP	TIMO_COMPA ; タイマ/カウンタ0比較A一致
\$000D/\$001A		RJMP/JMP	TIMO_COMPB ; タイマ/カウンタ0比較B一致
\$000E/\$001C		RJMP/JMP	TIMO_OVF ; タイマ/カウンタ0溢れ
\$000F/\$001E		RJMP/JMP	SPI_STC ; 直列周辺インターフェース転送完了
\$0010/\$0020		RJMP/JMP	VADC ; 電圧A/D変換完了
\$0011/\$0022		RJMP/JMP	CCADC ; クーロン カウンタA/D変換器瞬間電流変換完了
\$0012/\$0024		RJMP/JMP	CCADC_REG_C ; クーロン カウンタA/D変換器定常電流比較
\$0013/\$0026		RJMP/JMP	CCADC_ACC ; クーロン カウンタA/D変換器累積電流変換完了
\$0014/\$0028		RJMP/JMP	EE_RDY ; EEPROM操作可
;			
\$0015/\$002A	RESET:	LDI	R16, HIGH (RAMEND) ; RAM最終アドレス上位を取得
\$0016/\$002B		OUT	SPH, R16 ; スタック ポインタ上位を初期化
\$0017/\$002C		LDI	R16, LOW (RAMEND) ; RAM最終アドレス下位を取得
\$0018/\$002D		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

(訳注) 原書のATmega8HVA/16HVA各々の設定例を纏めました。

## 13. 外部割り込み

### 13.1. 概要

外部割り込みはINT0～2ピンによって起動されます。許可したなら、例えばINT0～2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。外部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。外部割り込みがレベル起動として設定かつ許可されると、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。割り込みは非同期に検知されます。これはそれらの割り込みが**アイドル動作**以外の**休止形態**からもデバイスを起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

レベル起動割り込みが**パワーセーブ動作**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。変化したレベルは超低電力(ULP)RC発振器クロックによって2度採取されます。ULP発振器の周期は25°C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「**クロックシステムとその配給**」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ULP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。

### 13.2. 外部割り込み用レジスタ

#### 13.2.1. 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	
	–	–	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	EICRA
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)

これらのビットは予約されており、常に0として読みます。

- ビット5～0 – ISC21,0～ISC01,0 : 外部割り込み2～0条件制御 (External Interrupt2～0 Sense Control Bits)

外部割り込み2～0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INT2～0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INT2～0ピンの端(エッジ)とレベルは**表13-1**で定義されます。INT2～0の端は非同期に記録されます。**表29-2**で与えられた最小パルス幅より広いINT2～0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ばれる場合、Lowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。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(継続的に)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13-1. 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	上昇端
1	1	下降端

注: n=2～0

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ねばなりません。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。

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

ビット	7	6	5	4	3	2	1	0	
\$1D (\$3D)	-	-	-	-	-	INT2	INT1	INT0	EIMSK
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

- ビット2~0 – INT2~INT0 : 外部割り込み2~0許可 (External Interrupt Request 2~0 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、INT2~0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。

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

ビット	7	6	5	4	3	2	1	0	
\$1C (\$3C)	-	-	-	-	-	INTF2	INTF1	INTF0	EIFR
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~3 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

- ビット2~0 – INTF2~INTF0 : 外部割り込み2~0要求フラグ (External Interrupt Flag 2~0)

INT2~0ピン上の端(エッジ)または論理変化が割り込み要求を起動すると、INTF2~0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(INT2~0)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。INT2~0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。INT2~0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INTF2~0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43頁の「デジタル入力許可と休止形態」をご覧ください。

## 14. 高電圧入出力ポート

### 14.1. 概要

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。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111頁の「電気的特性」を参照してください。

本章内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使う時は正確な形式(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40頁の「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。

### 14.2. 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オープンドレイン出力ポートです。加えて、それらは汎用デジタル入力としても使えます。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

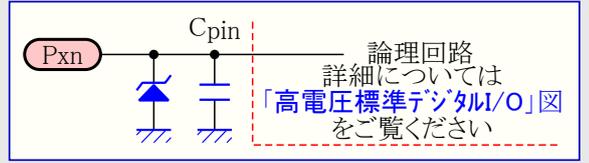
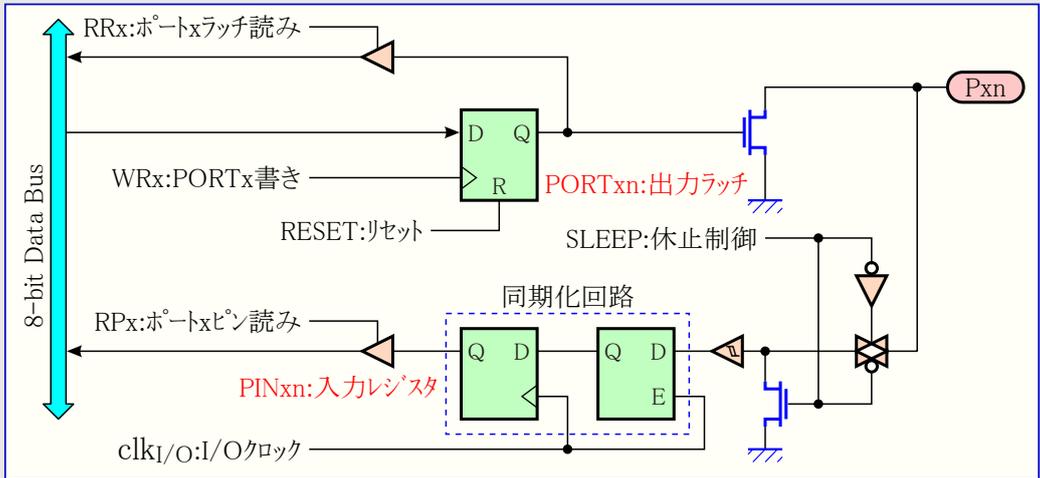


図14-2. 標準高電圧デジタル入出力



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

#### 14.2.1. ピンの設定

各ポートピンは2つのレジスタビット(PORTxnとPINxn)から成ります。「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、PINxnビットはPINx 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。

#### 14.2.2. ピンの読み方

ポートピンはPINxnレジスタのビットを通して読むことができます。図14-2.で示されるように、PINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。

## 14.3. 交換ポート機能

高電圧I/Oには標準デジタル入出力に加えて交換機能があります。図14-3は単純化された図14-2でのポートピン制御信号が交換機能により、どう重複できるかを示します。

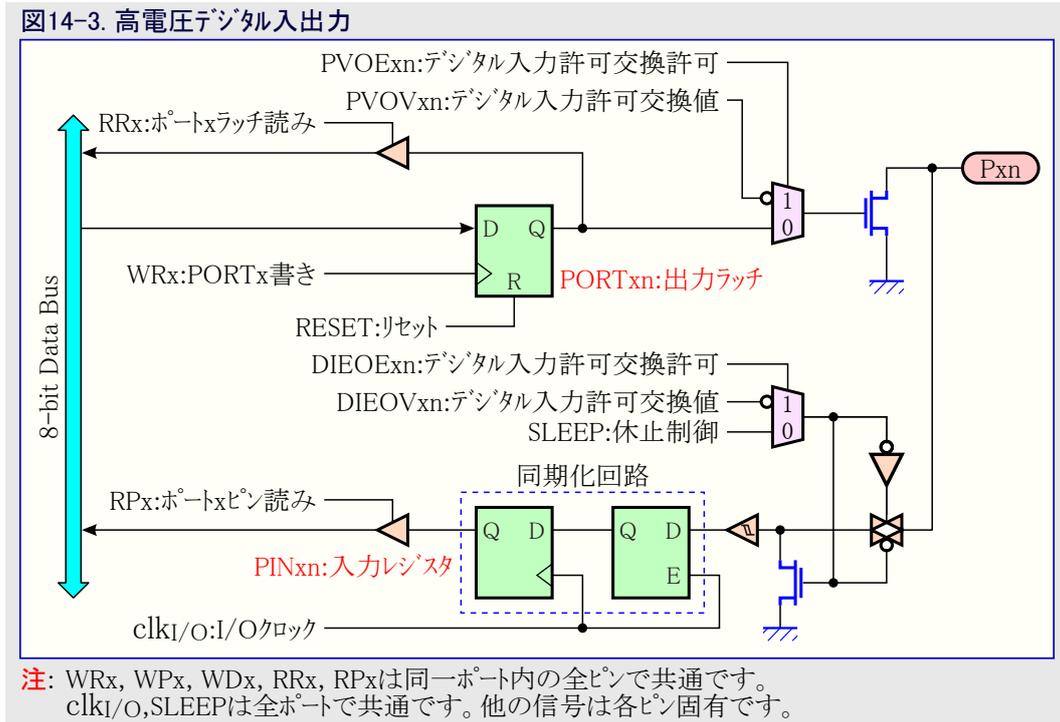


表14-1は重複(交換)信号の機能一覧を示します。図14-3.で示すピンとポートは次表で示されません。重複(交換)信号は交換機能を持つ機能部で内部的に生成されます。

表14-1. 交換機能用交換信号の一般定義

信号略名	信号名	意味
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。
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。

### 14.3.1. ポートCの交換機能

交換機能を持つポートCピンは表14-2.で示されます。

表14-2. ポートCピンの交換機能

ポートピン	交換機能
PC0	INT0 (外部割り込み要求0入力) ICP0 (タイマ/カウンタ0捕獲起動入力)

交換ピンの設定は次のとおりです。

- INT0 - ポートCビット0 : PC0

INT0 : 外部割り込み要求0入力。PC0ピンは外部割り込み元として扱えます。INT0は特殊機能が許可されているか否かに拘らず割り込みピンとして使えます。

表14-3.はポートCの交換機能を39頁の図14-3.で示される交換信号に関連付けます。

表14-3. ポートC0の交換機能用交換信号

信号名	PC0/INT0
PVOE	0
PVOV	-
DIEOE	INT0許可
DIEOV	1
DI	INT0入力

## 14.4. 高電圧I/Oポート用レジスタ

### 14.4.1. 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	

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

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	-	-	-	-	-	-	-	PINC0	PINC
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	不定	

## 15. 低電圧入出力ポート

### 15.1. 概要

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15-1.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111頁の「電気的特性」を参照してください。

本章内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使う時は正確な形式(例えば、ここで一般に記されたPORTxnがポートBのビット3に対してはPORTB3)が使われなければなりません。物理的なI/Oレジスタとビット位置は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(PORTx)、方向レジスタ(DDRx)、入力レジスタ(PINx)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PINxレジスタのビットへの論理1書き込みは、出力レジスタの対応ビット値を(1/0)反転する結果になります。加えてMCU制御レジスタ(MCUCR)のプルアップ禁止(PUD)ビットが設定(1)されると、全ポート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44頁の「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項を参照してください。

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

### 15.2. 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15-2.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15-1. 低電圧入出力ピン等価回路

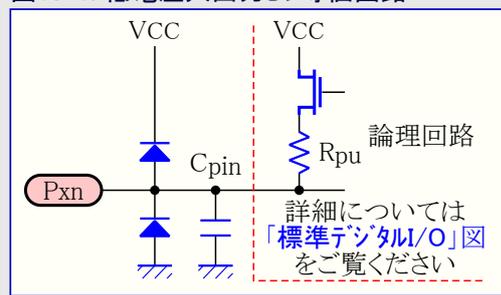
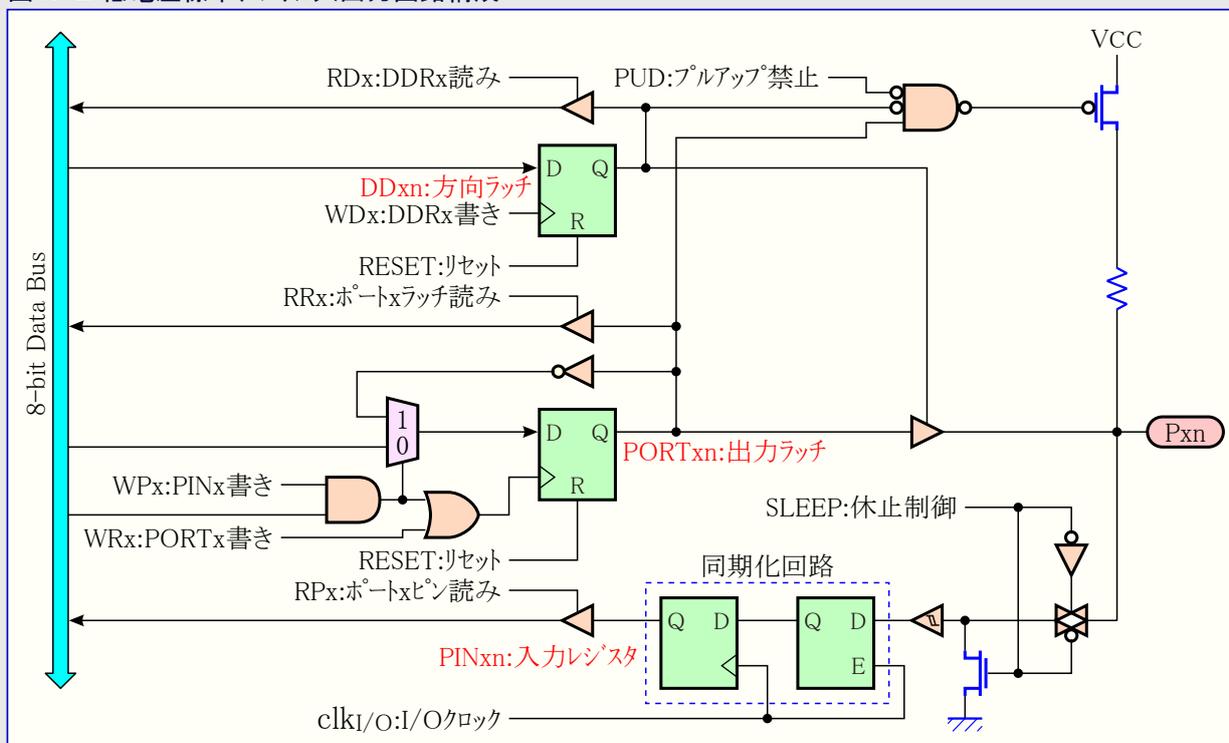


図15-2. 低電圧標準デジタル入出力回路構成



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

### 15.2.1. 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DDRx I/Oアドレス、PORTxnビットはPORTx I/Oアドレス、PINxnビットはPINx I/Oアドレスでアクセスされます。

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

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

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

### 15.2.2. 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。

### 15.2.3. 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15-1.はピン値に対する制御信号の一覧を示します。

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

### 15.2.4. 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15-2.で示されるようにPINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。図15-3.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15-3.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15-4.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15-3. 外部供給ピン値読み込み時の同期化

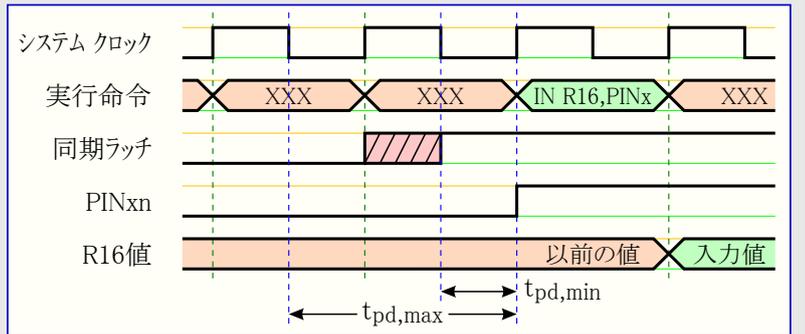
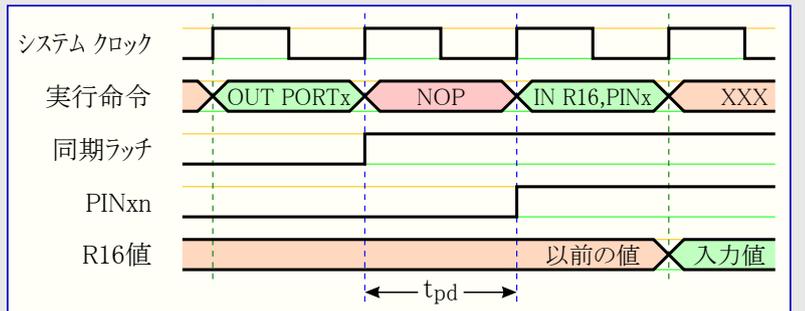


図15-4. プログラムで設定したピン値読み戻し時の同期化



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

## アセンブリ言語プログラム例

```

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

```

## C言語プログラム例

```

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

```

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

### 15.2.5. デジタル入力許可と休止形態

図15-2.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44頁の「**交換ポート機能**」で記載されるように様々な他の交換機能によっても無視されません。

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

### 15.2.6. 未接続ピン

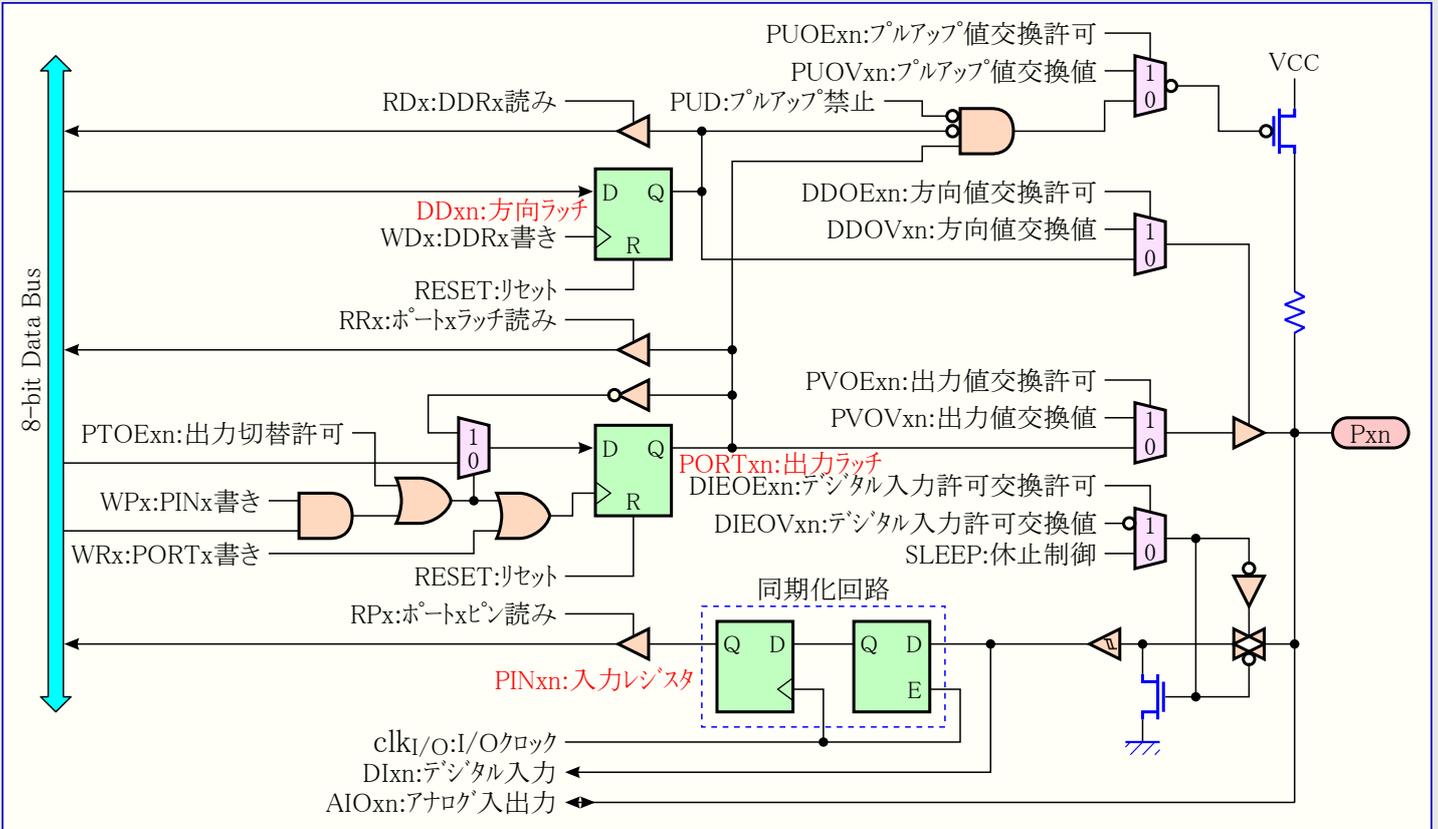
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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。

### 15.3. 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15-5.は単純化された図15-2.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVRマイクロコントローラシステムの全ポートピンに適用できる一般的な記述として取り扱います。

図15-5. 交換ポート機能入出力回路構成



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

表15-2.は重複(交換)信号の機能一覧を示します。図15-5.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15-2. 交換機能用交換信号の一般定義

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

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

## 15.3.1. ポートAの交換機能

交換機能を持つポートAピンが表15-3.で示されます。

表15-3. ポート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
PA1	ADC1 (電圧A/D変換器アナログ入力チャネル1) SGND (電圧A/D変換器用信号GND) T1 (タイマ/カウンタ1外部クロック入力)	PA0	ADC0 (電圧A/D変換器アナログ入力チャネル0) SGND (電圧A/D変換器用信号GND) T0 (タイマ/カウンタ0外部クロック入力)

交換ピンの設定は次のとおりです。

- ADC1/SGND/T1 - ポートAビット1 : PA1  
 ADC1 : 電圧A/D変換器アナログ入力チャネル1入力。  
 SGND : 電圧A/D変換器用信号GND。  
 T1 : タイマ/カウンタ1用外部クロック入力。
- ADC0/SGND/T0 - ポートAビット0 : PA0  
 ADC0 : 電圧A/D変換器アナログ入力チャネル0入力。  
 SGND : 電圧A/D変換器用信号GND。  
 T0 : タイマ/カウンタ0用外部クロック入力。

表15-4.はポートAの交換機能を44頁の図15-5.で示される交換信号に関連付けます。

表15-4. ポートA1~0の交換機能用交換信号

信号名	PA1/ADC1/SGND/T1	PA0/ADC0/SGND/T0
PUOE	0	0
PUOV	0	0
DDOE	0	0
DDOV	0	0
PVOE	0	0
PVOV	0	0
PTOE	-	-
DIEOE	PA1DID	PA0DID
DIEOV	0	0
DI	T1入力	T0入力
AIO	ADC1/SGND入力	ADC0/SGND入力

### 15.3.2. ポートBの交換機能

交換機能を持つポートBピンは表15-5.で示されます。

ポートピン	交換機能
PB3	INT2 (外部割り込み要求2入力) MISO (SPI 主装置側データ入力/従装置側データ出力)
PB2	INT1 (外部割り込み要求1入力) MOSI (SPI 主装置側データ出力/従装置側データ入力)
PB1	SCK (SPI 直列クロック 主装置側出力/従装置側入力)
PB0	$\overline{SS}$ (SPI 従装置選択入力) CKOUT (2分周システム クロック出力)

交換ピンの設定は次のとおりです。

• MISO/INT2 – ポートB ビット3 : PB3

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

INT2 : 外部割り込み要求2入力。SPI動作で操作されていない時、このピンは外部割り込み元として扱えます。

• MOSI/INT1 – ポートB ビット2 : PB2

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

INT1 : 外部割り込み要求1入力。SPI動作で操作されていない時、このピンは外部割り込み元として扱えます。

• SCK – ポートB ビット1 : PB1

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

•  $\overline{SS}$  – ポートB ビット0 : PB0

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

CKOUT : 2分周システム クロック出力。SPI動作で操作されていないとき、このピンは2分周システム(CPU)クロック出力として扱えます。19頁の「クロック出力」をご覧ください。

表15-6.はポートBの交換機能を44頁の図15-5.で示される交換信号に関連付けます。SPI主装置入力とSPI従装置出力がMISOを構成し、一方MOSIはSPI主装置出力とSPI従装置入力に分けられます。

信号名	PB3/MISO	PB2/MOSI	PB1/SCK	PB0/ $\overline{SS}$ /CKOUT
PUOE	SPE・MSTR	SPE・ $\overline{MSTR}$	SPE・ $\overline{MSTR}$	SPE・ $\overline{MSTR}$ ・CKOE
PUOV	PORTB3・ $\overline{PUD}$	PORTB2・ $\overline{PUD}$	PORTB1・ $\overline{PUD}$	PORTB0・ $\overline{PUD}$
DDOE	SPE・MSTR	SPE・MSTR	SPE・MSTR	SPE・MSTR・CKOE
DDOV	0	0	0	CKOE
PVOE	SPE・ $\overline{MSTR}$	SPE・MSTR	SPE・MSTR	CKOE
PVOV	SPI従装置出力	SPI主装置出力	SCK出力	0
PTOE	-	-	-	-
DIOE	INT2許可	INT1許可	-	CKOE
DIOV	INT2許可	INT1許可	-	0
DI	SPI主装置/INT2入力	SPI従装置/INT1入力	SCK入力	$\overline{SS}$ 入力
AIO	-	-	-	-

## 15.4. 低電圧I/Oポート用レジスタ

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

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

• ビット4 – PUD : プルアップ禁止 (Pull-up Disable)

このビットが1を書かれると、例えば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42頁の「[ピンの設定](#)」をご覧ください。

### 15.4.2. 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)	–	–	–	–	–	–	PORTA1	PORTA0	PORTA
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### 15.4.3. 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)	–	–	–	–	–	–	DDA1	DDA0	DDRA
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### 15.4.4. 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)	–	–	–	–	–	–	PINA1	PINA0	PINA
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	

### 15.4.5. 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)	–	–	–	–	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
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	

### 15.4.6. 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)	–	–	–	–	DDB3	DDB2	DDB1	DDB0	DDRB
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	

### 15.4.7. 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)	–	–	–	–	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	不定	不定	不定	不定	

## 16. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器

### 16.1. 概要

タイマ/カウンタ1とタイマ/カウンタ0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。

#### 16.1.1. 内部クロック元

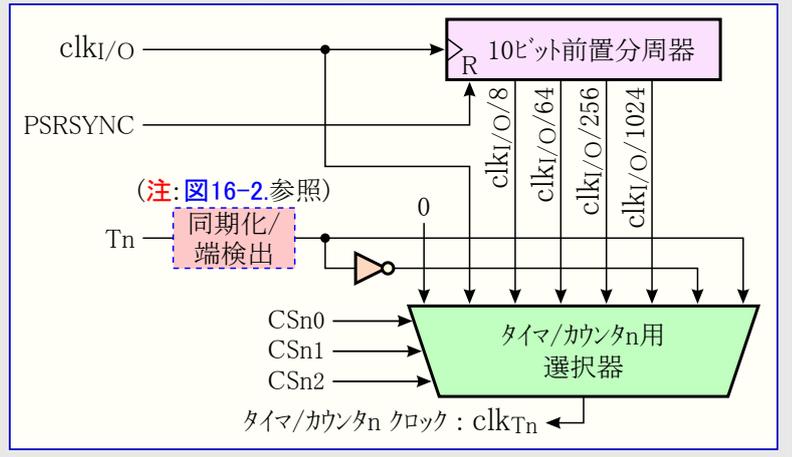
タイマ/カウンタはシステムクロック(CSn2~0=001設定)によって直接的にクロック駆動できます。これはシステムクロック周波数( $f_{clk\_I/O}$ )と等しいタイマ/カウンタ最大クロック周波数での最高速動作を提供します。選択で前置分周器からの4つのタップの1つがクロック元として使えます。この前置分周したクロックは $f_{clk\_I/O}/8$ ,  $f_{clk\_I/O}/64$ ,  $f_{clk\_I/O}/256$ ,  $f_{clk\_I/O}/1024$ の何れかの周波数です。

#### 16.1.2. 前置分周器リセット

この前置分周器は自由走行です(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します)。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て、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステムクロック周期数はNが前置分周値(8, 64, 256, 1024)とすると、1~N+1システムクロック周期になり得ます。

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

図16-1. タイマ/カウンタの前置分周器部構成図

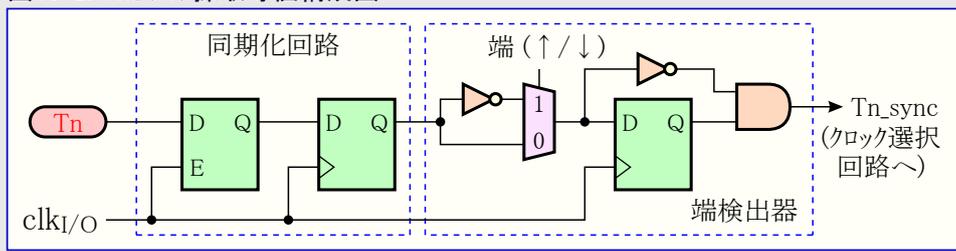


### 16.2. 外部クロック元

Tnピンに印加した外部クロック元はタイマ/カウンタクロック( $f_{clk\_Tn}$ )として使えます。このTnピンはピン同期化論理回路によって全システムクロック周期に1回採取されます。この同期化(採取)された信号はその後に端(エッジ)検出器を通して通過されます。図16-2はTn同期化と端検出器論理回路の機能等価構成図を示します。レジスタは内部システムクロック( $f_{clk\_I/O}$ )の上昇端でクロック駆動されます。ラッチは内部システムクロックのHigh区間で通過(Low区間で保持)です。

端検出器は上昇端(CSn2~0=111)または下降端(CSn2~0=110)の検出毎に1つの $clk_{Tn}$ パルスを生成します。詳細については次頁の表16-1をご覧ください。

図16-2. Tnピンの採取等価構成図



同期化と端検出器論理回路はTnピンへ印加した端からカウンタが更新されるまでに2.5~3.5システムクロック周期の遅延をもたらします。クロック入力の許可と禁止はTn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らない。この外部クロックは50%/50%デューティ比で与えられるとし、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らない。端検出器が採取を使うため、検出できる外部クロックの最大周波数は採取周波数の半分です(ナイストの標準化定理)。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。

## 16.3. タイマ/カウンタ前置分周器関係レジスタ

### 16.3.1. TCCRnB – タイマ/カウンタ制御レジスタB (Timer/Counter n Control Register B)

ビット	7	6	5	4	3	2	1	0	
\$25(\$45), \$81	–	–	–	–	–	CSn2	CSn1	CSn0	TCCRnB
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2~0 – CSn2~0 : クロック選択n (Clock Select n, bit 2,1 and 0)

このクロック選択ビット2~0はタイマ/カウンタ(TCNTn)の前置分周供給元を定義します。

表16-1. タイマ/カウンタ入力クロック選択

CSn2	CSn1	CSn0	意味
0	0	0	停止 (タイマ/カウンタ動作停止)
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	Tnピンの下降端 (外部クロック)
1	1	1	Tnピンの上昇端 (外部クロック)

外部ピン(クロック)動作がタイマ/カウンタに対して使われる場合、例えばTnピンが出力として設定されても、Tnピンの遷移は計数器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。

### 16.3.2. 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。

## 17. タイマ/カウンタ (T/C0,T/C1)

### 17.1. 特徴

- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 捕獲入力部
- 4つの独立した割り込み (TOVn,OCFnA,OCFnB,ICFn)
- 2つの独立した比較部での8ビット動作
- 1つの独立した比較部での16ビット動作

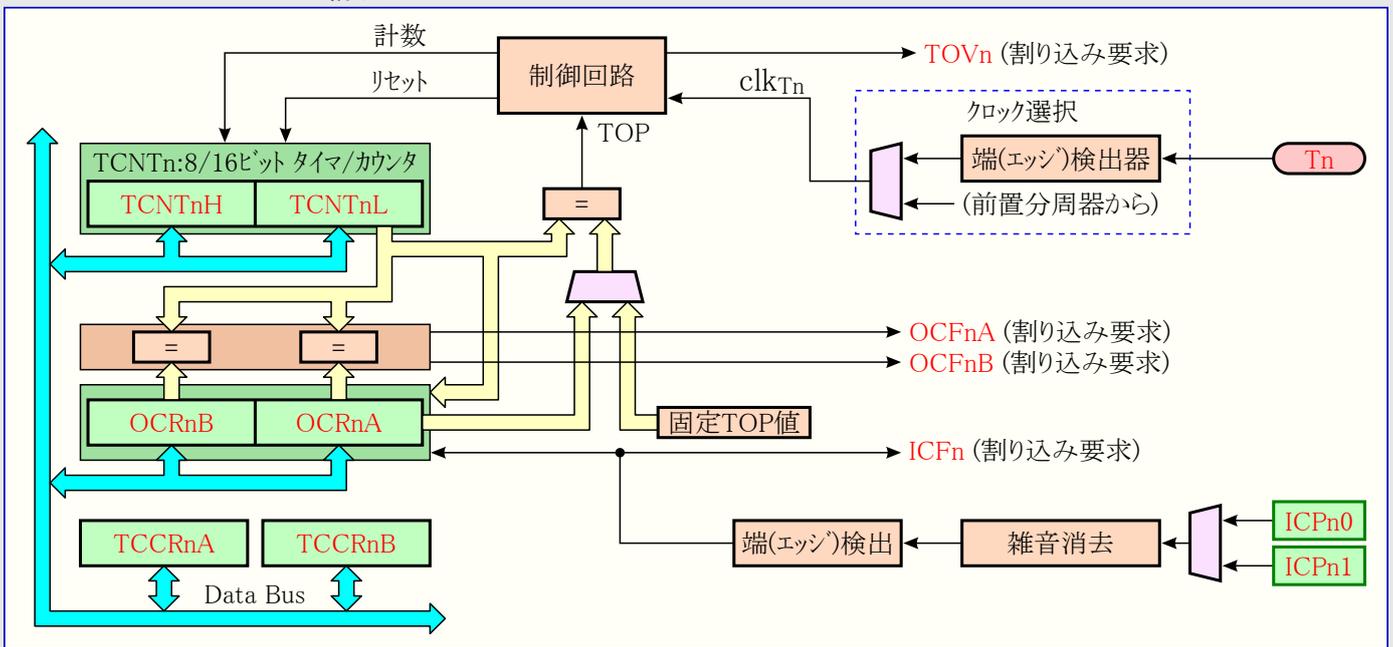
### 17.2. 概要

タイマ/カウンタnは2/1つの比較出力部と捕獲入力を持つ汎用8/16ビットタイマ/カウンタ部です。

ATmega8HVA/16HVAはタイマ/カウンタ0とタイマ/カウンタ1の2つのタイマ/カウンタを持ちます。両タイマ/カウンタに関する機能が以下で記述されます。118頁の「レジスタ要約」で示されるようにタイマ/カウンタ0とタイマ/カウンタ1は異なるタイマ/カウンタレジスタを持ちます。

タイマ/カウンタnの一般動作は8/16ビット動作で記述されます。この8/16ビットタイマ/カウンタの簡略化された構成図は図17-1.で示されます。実際のI/Oピンの配置については2頁の「ピン配置」を参照してください。CPUがアクセス可能なI/OビットとI/Oピンを含むI/Oレジスタは赤文字(訳注: 原文太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は58頁の「タイマ/カウンタ用レジスタ」で示されます。

図17-1. 8/16ビットタイマ/カウンタ構成図



#### 17.2.1. 関係レジスタ

タイマ/カウンタ下位バイト(TCNTnL)レジスタ、比較レジスタ(OCRnA,OCRnB)は8ビットレジスタです。(図17-1.で省略された)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(TIFRn)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(TIMSKn)で個別に遮蔽(禁止)されます。TIFRnとTIMSKnはこの図で示されません。

16ビット動作のタイマ/カウンタには更に1つの8ビットレジスタ、タイマ/カウンタ上位バイト(TCNTnH)レジスタがあります。尚、2つの比較レジスタのOCRnAとOCRnBは1つの16ビット比較レジスタに組み合わされるので、16ビット動作では1つの比較部だけです。OCRnAは語の下位バイトを含み、OCRnBは語の上位バイトを含みます。16ビットレジスタをアクセスするとき、56頁の「16ビット動作でのレジスタアクセス」項で記述される特別な手順に従わなければなりません。

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

## 17.2.2. 定義

本章でのレジスタとビット参照の多くは一般形で書かれます。小文字の'n'は単位部番号、例えばタイマ/カウンタ番号で置き換えます。小文字のxは部、例えばOCRnXとICPnxはOCRnB/AとICP1/0xと記述します。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TCNT0Lのように)。

表17-1.の定義は本文書を通して広範囲に渡ってます。

表17-1. 用語定義

用語	意味
BOTTOM	タイマ/カウンタが\$00/\$0000に到達した時。
MAX	タイマ/カウンタが8ビット動作で\$FF(255)または16ビット動作で\$FFFF(65536)に到達した時。
TOP	タイマ/カウンタが計数内の最大値と等しくなるTOPに到達した時。TOP値は固定値(\$FF/\$FFFF)またはOCRnAレジスタに格納した値を指定できます。

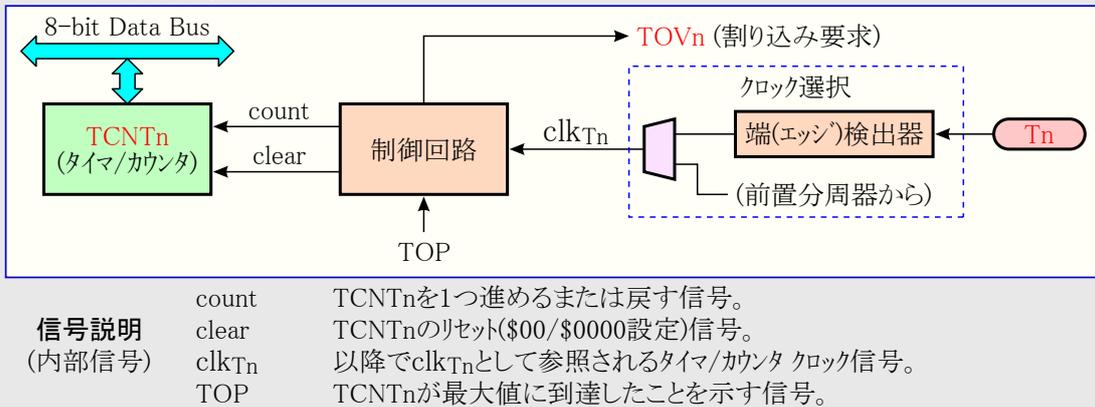
## 17.3. タイマ/カウンタのクロック

このタイマ/カウンタは内部、前置分周器経由、または外部クロック元によってクロック駆動できます。クロック選択論理回路はタイマ/カウンタ制御レジスタB(TCCRNb)に配置されたクロック選択(CSn2~0)ビットによって制御され、タイマ/カウンタが進行に使うクロック元と端(エッジ)を制御します。クロック選択論理回路からの出力はタイマ/カウンタクロック(clkTn)として参照されます。クロック元と前置分周器の詳細については48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。

## 17.4. 計数器部

8ビットタイマ/カウンタの主な部分は設定可能なカウンタ部です。図17-2.は、このカウンタとその周辺環境の構成図を示します。

図17-2. 計数器部構成図



カウンタはTOP値を通過するまで各タイマ/カウンタクロック(clkTn)で増加(+1)され、そしてBOTTOMから再び計数します。計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCRNb)に配置された波形生成種別(WGMn0)ビットの設定によって決定されます。計数順序についてより多くの詳細に関しては55頁の「タイマ/カウンタのタイミング」をご覧ください。clkTnはクロック選択(CSn2~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CSn2~0=000)と、タイマ/カウンタは停止されます。けれどもTCNTn値はタイマ/カウンタクロック(clkTn)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)。タイマ/カウンタ溢れ(TOVn)フラグはカウンタが最大値に到達する時に設定(1)され、それはCPU割り込み発生に使えます。

## 17.5. 動作種別

動作種別はタイマ/カウンタ制御レジスタA(TCCRNb)のタイマ/カウンタ幅(TCWn)、捕獲入力許可(ICENn)、波形生成種別(WGMn0)ビットによって定義されます。表17-2.は各種動作種別を示します。

表17-2. 動作種別

番号	ICENn	TCWn	WGMn0	タイマ/カウンタ動作種別	TOP値	OCRnx更新時	TOVn設定時
0	0	0	0	標準8ビット動作	\$FF	即時	MAX(\$FF)
1	0	0	1	8ビット比較一致T/C解除(CTC)動作	OCRnA	即時	MAX(\$FF)
2	0	1	0	16ビット動作	\$FFFF	即時	MAX(\$FFFF)
3	0	1	1	16ビット比較一致T/C解除(CTC)動作	OCRnB, OCRnA	即時	MAX(\$FFFF)
4	1	0	0	8ビット捕獲入力動作	\$FF	-	MAX(\$FF)
5	1	1	0	16ビット捕獲入力動作	\$FFFF	-	MAX(\$FFFF)

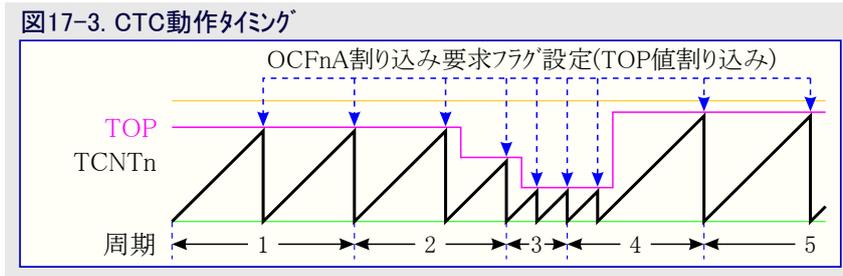
### 17.5.1. 標準8ビット動作

標準8ビット動作(表17-2.参照)ではタイマ/カウンタ(TCNTnL)が8ビット最大値(MAX=\$FF)を通過するまで増加し、そしてBOTTOM(\$00)から再度始めます。溢れ(TOVn)フラグはTCNTnLが0になるのと同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOVn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TOVn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。

### 17.5.2. 8ビット比較一致タイマ/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(表17-2.参照)ではOCRnAレジスタがカウンタの分解能を操作するために使われます。CTC動作ではカウンタ(TCNTn)値がOCRnAと一致すると、カウンタが\$00に解除されます。OCRnA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17-3.で示されます。カウンタ(TCNTn)値はTCNTnとOCRnA間で比較一致が起きるまで増加し、そしてその後カウンタ(TCNTn)は解除(\$00)されます。



OCFnA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と近い値へのTOP値変更は注意して行わなければなりません。OCRnAに書かれた新しい値がTCNTnの現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後カウンタは比較一致が起こるのに先立って最大値(\$FFF)へ、そして次に\$00から始める計数をしななければならないでしょう。標準動作と同じように、タイマ/カウンタ溢れ(TOVn)フラグはカウンタがMAXから\$00へ計数する同じタイマ/カウンタクロック周期で設定(1)されます。

### 17.5.3. 16ビット動作

16ビット動作(表17-2.参照)ではカウンタ(TCNTnH/L)が16ビット最大値(MAX=\$FFFF)を通過するまで増加し、そしてBOTTOM(\$0000)から再度始めます。溢れ(TOVn)フラグはTCNTnH/Lが0になるのと同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOVnフラグは設定(1)のみで解除(0)されないことを除いて第17ビットのようになります。けれどもTOVn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。

### 17.5.4. 16ビット比較一致タイマ/カウンタ解除(CTC)動作

16ビット比較一致解除動作ではOCRnB/Aレジスタがカウンタの分解能を操作するために使われます。ビット設定については表17-2.をご覧ください。CTC動作ではカウンタ(TCNTn)値がOCRnB/Aと一致すると、カウンタが\$0000に解除され、ここでのOCRnBは8 MSBを表し、OCRnAは8 LSBを表します。OCRnB/A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。

OCFnA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と近い値へのTOP値変更は注意して行わなければなりません。OCRnB/Aに書かれた新しい値がTCNTnの現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後カウンタは比較一致が起こるのに先立って最大値(\$FFFF)へそして次に\$0000から始める計数をしななければならないでしょう。16ビット動作と同じように、タイマ/カウンタ溢れ(TOVn)フラグはカウンタがMAXから\$0000へ計数する同じタイマ/カウンタクロック周期で設定(1)されます。

### 17.5.5. 8ビット捕獲入力動作

タイマ/カウンタは8ビット捕獲入力動作にも使え、ビット設定については表17-2.をご覧ください。完全な記述に関しては次ページの「捕獲入力部」項をご覧ください。

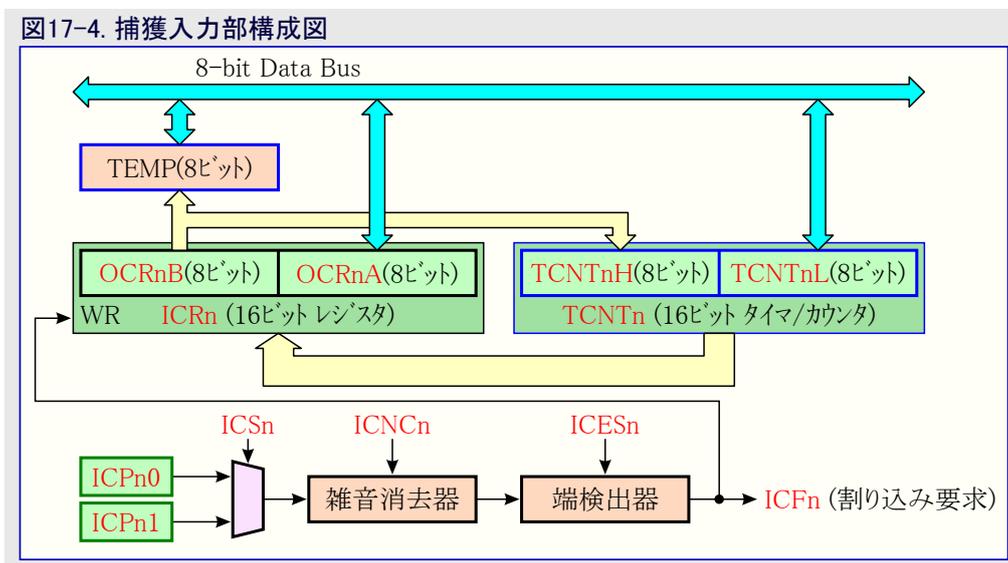
### 17.5.6. 16ビット捕獲入力動作

タイマ/カウンタは16ビット捕獲入力動作にも使え、ビット設定については表17-2.をご覧ください。完全な記述に関しては次ページの「捕獲入力部」項をご覧ください。

## 17.6. 捕獲入力部

タイマ/カウンタは外部の出来事を捕獲でき、発生時間を示す時間印(タイマ/カウンタ値)を与える捕獲入力部と合体します。外部信号は出来事または複数の出来事を示します。タイマ/カウンタ0に対する事象はPC0ピン(ICP01)または代わりに発振器採取インターフェースでの“OSI\_上昇端”信号(ICP00)経由で印加できます。タイマ/カウンタ1に対する事象は電池保護割り込み(ICP10)または代わりに電圧調整器割り込み(ICP11)によって印加できます。時間印はその後、周波数、デューティ比、印加された信号の他の特徴の計算に使えます。代わりに時間印は出来事の記録作成にも使えます。

捕獲入力部は図17-4.で示される構成図によって図解されます。直接的な捕獲入力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。



比較レジスタ(OCRnA)は8ビット捕獲入力レジスタ(ICRn)としても使われる二重目的レジスタです。16ビット捕獲入力動作での比較Bレジスタ(OCRnB)は捕獲入力レジスタ(ICRn)の上位バイトを扱います。8ビット捕獲入力動作でのOCRnBは標準比較レジスタとして使われるために空いていますが、16ビット捕獲入力動作での比較部は空いている比較レジスタがないので使えません。本章では捕獲入力レジスタはICRnと呼ばれますが、これは比較レジスタへの参照です。16ビットレジスタアクセス法のより多くの情報に関しては56頁の「16ビット動作でのレジスタアクセス」を参照してください。

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

### 17.6.1. 捕獲起動元

捕獲入力部に対する既定の起動元はタイマ/カウンタ0でのI/OポートPC0とタイマ/カウンタ1での電池保護割り込みです。タイマ/カウンタ0は発振器採取インターフェースの“OSI\_上昇端”信号、タイマ/カウンタ1は電圧調整器割り込みを代わりに使えます。“OSI\_上昇端”信号はタイマ/カウンタ0制御レジスタA(TCCR0A)で、電圧調整器割り込みはタイマ/カウンタ1制御レジスタA(TCCR1A)で捕獲入力選択(ICSn)ビットを各々ICS0=0とICS1=1に設定することによって起動元として選べます。起動元を変更することが捕獲を起動し得ることに気付いてください。従って捕獲割り込み要求フラグ(ICFn)は、その変更後に解除(0)されなければなりません。

捕獲起動入力はTnピン(48頁の図16-2.参照)についてと同じ技法を使って採取されます。端検出器も全く同じです。けれども雑音除去が許可されると、付加論理回路が端検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタ0での捕獲入力PC0ピンのポートを制御することによってソフトウェアで起動できます。

### 17.6.2. 雑音消去器

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

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

### 17.6.3. 捕獲入力の使用

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

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

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

表17-3. タイマ/カウンタ0捕獲起動元(ICS)

ICS0	起動元
0	ICP00: OSI単位部からの"OSI_上昇端" (注)
1	ICP01: ポートPC0

注: 詳細に関しては19頁の「OSI - 発振器採取インターフェース」をご覧ください。

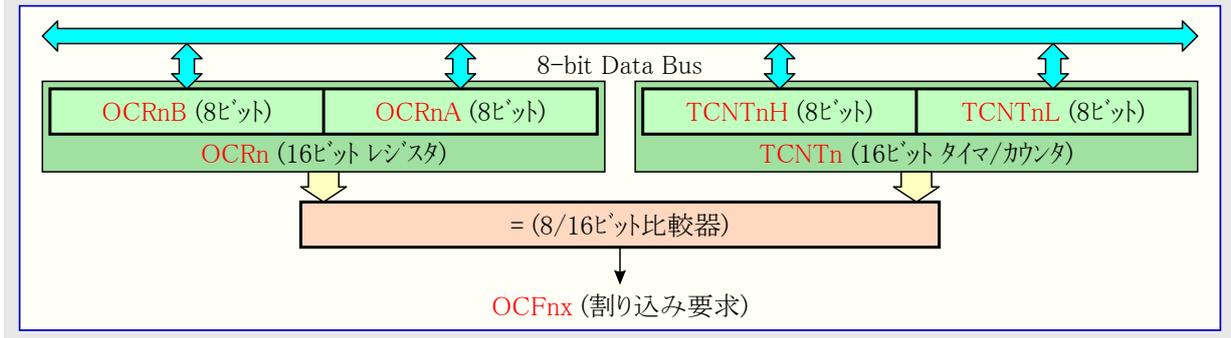
表17-4. タイマ/カウンタ1捕獲起動元(ICS)

ICS1	起動元
0	ICP10: 電池保護割り込み
1	ICP11: 電圧調整器割り込み

### 17.7. 比較部

この比較器はTCNTnと比較レジスタ(OCRnAとOCRnB)を継続的に比較し、タイマ/カウンタが比較レジスタと等しい時は必ず、比較器が一致を指示します。この一致は**次のタイマ/カウンタクロック周期**で比較割り込み要求フラグ(OCFnx)を設定(1)します。8ビット動作での一致は比較一致フラグ、OCFnAまたはOCFnBのどちらをも設定(1)し得ます。16ビット動作での一致は1つの比較部だけのため、比較一致フラグ(OCFnA)だけを設定(1)し得ます。対応する割り込みが許可(I=1, OCIEnx=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。OCFnx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OCFnx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17-5は比較部の構成図を示します。

図17-5. 比較部構成図



#### 17.7.1. TCNTn書き込みによる比較一致妨害

TCNTnH/L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、TCNTnと同じ値に初期化されることをOCRnB/Aに許します。

#### 17.7.2. 比較一致部の使用

どの動作種別でのTCNTnH/L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ないかに拘らず、何れかの比較出力部を使う場合にTCNTnH/Lを変更する時は危険を伴います。TCNTnH/Lに書かれた値がOCRnB/A値と同じ場合、比較一致は失われます(一致が発生しません)。

## 17.8. タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従って以下の図でタイマ/カウンタクロック( $clk_{Tn}$ )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17-6.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MAX値近辺の計数の流れを示します。

図17-6. 前置分周なし(1/1)のタイマ/カウンタ タイミング

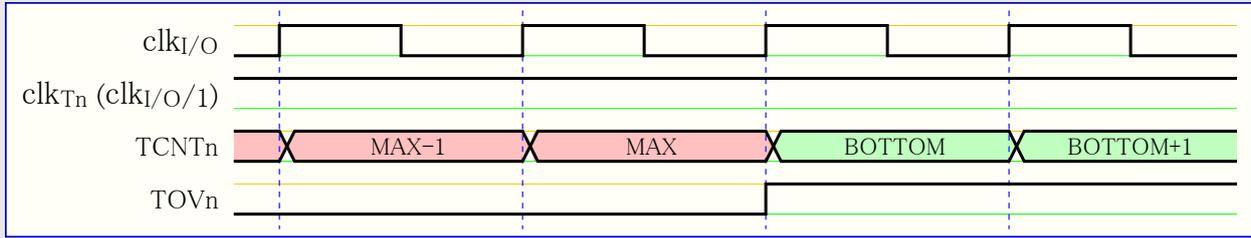


図17-7.は同じタイミング データを示しますが、前置分周器が許可されています。

図17-7. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ タイミング

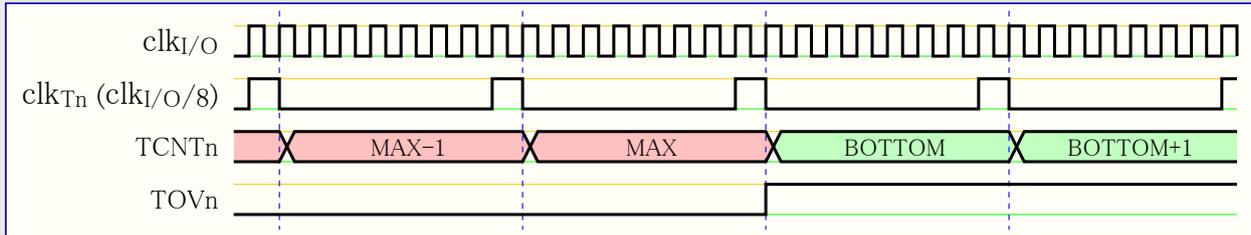


図17-8.は標準動作での $OCFnA$ と $OCFnB$ の設定(0⇒1)を示します。

図17-8. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、 $OCFn_x$ 設定 タイミング

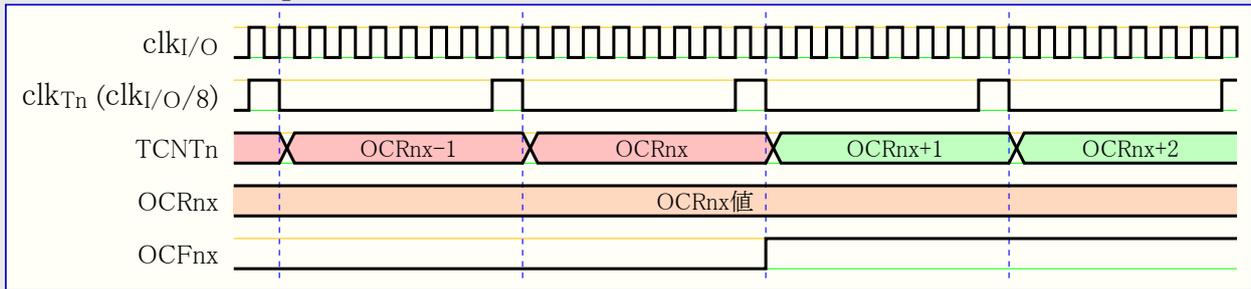
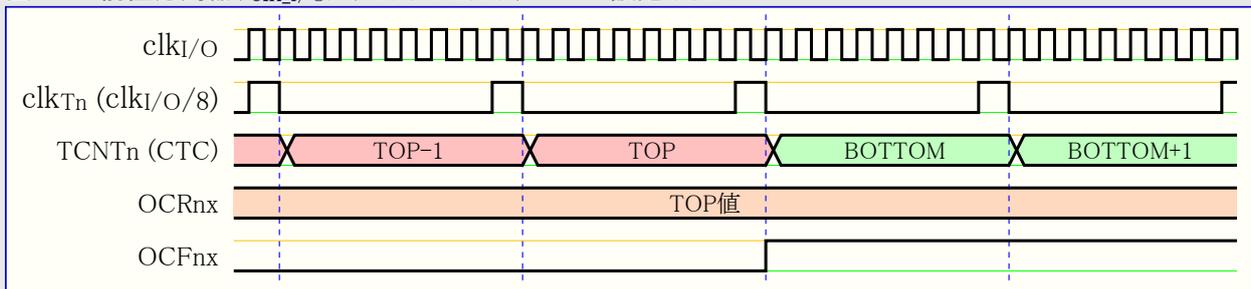


図17-9.はCTC動作での $TCNTn$ の解除と $OCFnA$ の設定(0⇒1)を示します。

図17-9. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、 $OCFnA$ 設定 タイミング



## 17.9. 16ビット動作でのレジスタ アクセス

16ビット動作(TCCRnAのタイマ/カウンタ幅(TCWn)ビットが1に設定)でのTCNTnH/LとOCRnB/A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。全ての16ビットレジスタ間でこの同じ一時レジスタが共用されます。下位バイトアクセスが16ビット読み書き動作を起動します。16ビットレジスタの下位バイトがCPUによって書かれると、一時レジスタに保存した上位バイトと書かれた下位バイトは同じクロック周期で16ビットレジスタに両方が複写されます。16ビットレジスタの下位バイトがCPUによって読まれると、16ビットレジスタの上位バイトは下位バイトが読まれるのと同じクロック周期で一時レジスタに複写されます。

一時レジスタ使用に1つの例外があります。比較動作での16ビット比較レジスタ(OCRnA, OCRnB)は、比較レジスタがCPUアクセスによってだけ変更される固定値を含むので、一時レジスタなしに読まれます。けれどもOCRnAとOCRnBによって形成された捕獲入力動作でのICRnレジスタは一時レジスタとでアクセスされなければなりません。

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。OCRnB/A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。

### アセンブリ言語プログラム例

```

~
LDI    R17, $01
LDI    R16, $FF
OUT    TCNTnH, R17
OUT    TCNTnL, R16

IN     R16, TCNTnL
IN     R17, TCNTnH
~

```

```

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

```

### C言語プログラム例

```

unsigned int i;
~
TCNTnH = 0x01;
TCNTnL = 0xFF;

i = TCNTnL;
i |= ((unsigned int)TCNTnH<<8);
~

```

```

/**/
/* [16ビット($01FF)書き込み] */
/* 上位バイト設定(一時レジスタ) */
/* 下位バイト設定(一時レジスタ⇒上位バイト) */
/* [16ビット読み込み] */
/* 下位バイト取得(上位バイト⇒一時レジスタ) */
/* 上位バイト取得(一時レジスタ)合成 */
/**/

```

**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TCNTnH/L値を戻します。

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

次のコード例はTCNTnH/L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、OCRnB/Aのどんな読み込みも行えます。

## アセンブリ言語プログラム例

```
RD_TCNT0:   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, TCNTnL        ; TCNTn下位バイト取得(上位バイト⇒一時レジスタ)
            IN      R17, TCNTnH        ; TCNTn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_TCNT0(void)
{
    unsigned char sreg;                /* ステータスレジスタ一時保存変数定義 */
    unsigned int i;                    /* TCNTn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 = TCNTnL;                         /* 下位バイト取得(上位バイト⇒一時レジスタ) */
    i |= ((unsigned int)TCNTnH<<8);    /* 上位バイト取得(一時レジスタ)合成 */
    SREG = sreg;                       /* 全割り込み許可フラグ(I)を復帰 */
    return i;                          /* TCNTn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TCNTnH/L値を戻します。

次のコード例はTCNTnH/L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、OCRnB/Aのどんな書き込みも行えます。

## アセンブリ言語プログラム例

```
WR_TCNT0:   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     TCNTnH, R17        ; TCNTn上位バイト設定(一時レジスタ)
            OUT     TCNTnL, R16        ; TCNTn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_TCNT0(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;                    /* TCNTn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 */
    TCNTnH = (i>>8);                  /* 上位バイト設定(一時レジスタ) */
    TCNTnL = (unsigned char)i;        /* 下位バイト設定(一時レジスタ⇒上位バイト) */
    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TCNTnH/Lへ書かれるべき値を含む必要があります。

### 17.9.1. 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。

## 17.10. タイマ/カウンタn用レジスタ

### 17.10.1. TCCRnA – タイマ/カウンタ制御レジスタA (Timer/Counter n Control Register A)

ビット	7	6	5	4	3	2	1	0	
\$24(\$44):(\$80)	TCWn	ICENn	ICNCn	ICESn	ICSn	–	–	WGMn0	TCCRnA
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット7 – TCWn : タイマ/カウンタ幅 (Timer/Counter n Width)**

このビットが**1**を書かれると、16ビット動作が選ばれます。タイマ/カウンタの幅が16ビットに設定され、比較レジスタのOCRnAとOCRnBが1つの16ビット比較レジスタを形成するために組み合わせられます。16ビットレジスタのTCNTnH/LとOCRnB/Aは8ビットデータバス経由でAVR CPUによってアクセスされるので、特別な手順に従わなければなりません。これらの手順は56頁の「16ビット動作でのレジスタアクセス」項で記述されます。

- **ビット6 – ICENn : 捕獲入力動作許可 (Input Capture Mnode Enable)**

このビットが**1**を書かれると、捕獲入力動作が許可されます。

- **ビット5 – ICNCn : 捕獲起動入力雑音消去許可 (Input Capture Noise Canceler)**

このビットを**1**に設定することは捕獲起動入力雑音消去器を活性(有効)にします。雑音消去器が有効にされると、捕獲起動元からの入力が濾波されます。この濾波器機能はそれが出力を更新することに対して連続4回等しく評価された捕獲起動元の採取を必要とします。雑音消去器が許可されると、捕獲入力はこれによって4発振器(システムクロック)周期遅らされます。

- **ビット4 – ICESn : 捕獲起動入力端選択 (Input Capture Edge Select)**

このビットは出来事での捕獲を起動するために使われる捕獲起動元のどちらかの端(エッジ)を選びます。ICESnビットが**0**を書かれると起動動作として下降(負)端が使われ、ICESnビットが**1**を書かれると上昇(正)端が捕獲を起動します。捕獲がICESn設定に従って起動されると、カウンタ値が捕獲レジスタ(OCRnB/A)に複写されます。この出来事は捕獲入力割り込み要求フラグ(ICFn)も設定**1**し、そしてこれはこの割り込みが許可されていれば捕獲入力割り込みを起こすのに使えます。

- **ビット3 – ICSn : 捕獲起動入力選択 (Input Capture Select)**

論理**1**を書かれると、このビットはタイマ/カウンタの捕獲機能に対する起動元として代替捕獲起動元を選びます。タイマ/カウンタ捕獲割り込みを起動するには、タイマ/カウンタ割り込み許可レジスタ(TIMSKn)の捕獲割り込み許可(ICIEEn)ビットが設定**1**されなければなりません。54頁の表17-3と表17-4をご覧ください。

- **ビット2,1 – Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

- **ビット0 – WGMn0 : 波形生成種別 (Waveform Generation Mode)**

このビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元を制御します。51頁の表17-2をご覧ください。タイマ/カウンタ部によって支援される動作種別は標準動作(計数器)と比較一致タイマ/カウンタ解除(CTC)動作です。51頁の「動作種別」をご覧ください。

## 17.10.2. TCNTnL – タイマ/カウンタn下位バイト (Timer/Counter n Register Low Byte)

ビット	7	6	5	4	3	2	1	0	
\$26(\$46):(\$84)	(MSB)							(LSB)	TCNTnL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

タイマ/カウンタレジスタ下位バイト(TCNTnL)は読み書き両方についてタイマ/カウンタ部の8ビットカウンタに直接アクセスします。TCNTnLレジスタ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(TCNTnL)を変更することは、OCRnxとTCNTnL間の比較一致消失の危険を誘発します。16ビット動作でのTCNTnLレジスタは16ビットタイマ/カウンタレジスタの下位部を含みません。

## 17.10.3. TCNTnH – タイマ/カウンタn上位バイト (Timer/Counter n Register High Byte)

ビット	7(15)	6(14)	5(13)	4(12)	3(11)	2(10)	1(9)	0(8)	
\$27(\$47):(\$85)	(MSB)							(LSB)	TCNTnH
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	

16ビット動作が選ばれる(TCCRNのTCWn=1)と、タイマ/カウンタn上位レジスタ(TCNTnH)はタイマ/カウンタn下位レジスタ(TCNTnL)と組み合わせられ、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に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。56頁の「16ビット動作でのレジスタアクセス」をご覧ください。8ビット動作でのこのレジスタは読み書き両方に対してアクセス可能ですが、カウンタによって更新されません。

## 17.10.4. OCRnA – タイマ/カウンタn 比較Aレジスタ (Timer/Counter n Output Compare A Register)

ビット	7	6	5	4	3	2	1	0	
\$28(\$48):(\$88)	(MSB)							(LSB)	OCRnA
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Aレジスタは継続的にカウンタ(TCNTnL)値と比較される8ビットの値を含みます。一致は比較一致割り込みを生成するために使えます。

16ビット動作でのOCRnAレジスタは16ビット比較レジスタの下位バイトを含みます。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。56頁の「16ビット動作でのレジスタアクセス」をご覧ください。

OCRnAが捕獲動作で書き込み不能なことに注意してください。

## 17.10.5. OCRnB – タイマ/カウンタn 比較Bレジスタ (Timer/Counter n Output Compare B Register)

ビット	7(15)	6(14)	5(13)	4(12)	3(11)	2(10)	1(9)	0(8)	
\$29(\$49):(\$89)	(MSB)							(LSB)	OCRnB
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Bレジスタは継続的にカウンタ(8ビット動作でのTCNTnLまたは16ビット動作でのTCNTnH)値と比較される8ビットの値を含みます。一致は比較一致割り込みを生成するために使えます。

16ビット動作でのOCRnBレジスタは16ビット比較レジスタの上位バイトを含みます。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。56頁の「16ビット動作でのレジスタアクセス」をご覧ください。

OCRnBが捕獲動作で書き込み不能なことに注意してください。

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

ビット (\$6E):(\$6F)	7	6	5	4	3	2	1	0	TIMSKn
	-	-	-	-	ICIE <sub>n</sub>	OCIE <sub>nB</sub>	OCIE <sub>nA</sub>	TICIE <sub>n</sub>	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット3 – ICIE<sub>n</sub> : タイマ/カウンタn捕獲割り込み許可 (Timer/Counter n Input Capture Interrupt Enable)**

このビットが**1**を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(**1**)されると、タイマ/カウンタn捕獲割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFRn)で**タイマ/カウンタn捕獲割り込み要求(ICFn)フラグ**が設定(**1**)されると、対応する割り込み(34頁の「割り込み」参照)が実行されます。

• **ビット2 – OCIE<sub>nB</sub> : タイマ/カウンタn比較B割り込み許可 (Timer/Counter n Output Compare Match B Interrupt Enable)**

OCIE<sub>nB</sub>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n比較B一致割り込みが許可されます。タイマ/カウンタnで比較B一致が起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFRn)で**タイマ/カウンタn比較B割り込み要求フラグ(OCFnB)**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 – OCIE<sub>nA</sub> : タイマ/カウンタn比較A割り込み許可 (Timer/Counter n Output Compare Match A Interrupt Enable)**

OCIE<sub>nA</sub>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n比較A一致割り込みが許可されます。タイマ/カウンタnで比較A一致が起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFRn)で**タイマ/カウンタn比較A割り込み要求フラグ(OCFnA)**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 – TOIE<sub>n</sub> : タイマ/カウンタn溢れ割り込み許可 (Timer/Counter n Overflow Interrupt Enable)**

TOIE<sub>n</sub>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n溢れ割り込みが許可されます。タイマ/カウンタn溢れが起こる、換言するとタイマ/カウンタ割り込み要求フラグレジスタ(TIFRn)で**タイマ/カウンタn溢れ割り込み要求(TOVn)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。

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

ビット \$15(\$35):\$16(\$36)	7	6	5	4	3	2	1	0	TIFRn
	-	-	-	-	ICFn	OCFnB	OCFnA	TOVn	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット3 – ICF<sub>n</sub> : タイマ/カウンタn捕獲割り込み要求フラグ (Timer/Counter n Input Capture Flag)**

**タイマ/カウンタ制御レジスタA(TCCRnA)のICEN<sub>n</sub>, ICES<sub>n</sub>, ICS<sub>n</sub>ビットの設定**に対応した捕獲事象が起こると、このフラグが設定(**1**)されます。捕獲割り込みベクタが実行されると、ICFnは自動的に解除(**0**)されます。代わりにこのビット位置へ論理**1**を書くことによってもICFnは解除(**0**)できます。

• **ビット2 – OCF<sub>nB</sub> : タイマ/カウンタn比較B割り込み要求フラグ (Timer/Counter n Output Compare B Match Flag)**

OCFnBビットは比較一致が**タイマ/カウンタ(TCNTn)と比較レジスタ(OCRnB)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、OCFnB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OCFnBは解除(**0**)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSKn)の**タイマ/カウンタn比較B一致割り込み許可(OCIE<sub>nB</sub>)ビット**、OCFnBが設定(**1**)されると、タイマ/カウンタn比較B一致割り込みが実行されます。

OCFnBは比較B(OCRnB)レジスタが16ビット比較/捕獲動作で16ビットの比較レジスタまたは捕獲入力レジスタの上位バイトとして使われる時は設定(**1**)されません。

• **ビット1 – OCF<sub>nA</sub> : タイマ/カウンタn比較A割り込み要求フラグ (Timer/Counter n Output Compare A Match Flag)**

OCFnAビットは比較一致が**タイマ/カウンタ(TCNTn)と比較レジスタ(OCRnA)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、OCFnA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OCFnAは解除(**0**)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSKn)の**タイマ/カウンタn比較A一致割り込み許可(OCIE<sub>nA</sub>)ビット**、OCFnAが設定(**1**)されると、タイマ/カウンタn比較A一致割り込みが実行されます。

OCFnAは**16ビット動作**でタイマ/カウンタとOCRnB/A内のデータ間に比較一致が起こる時にも設定(**1**)されます。OCFnAは捕獲入力レジスタとしてOCFnAが使われる時の捕獲入力では設定(**1**)されません。

• **ビット0 – TOV<sub>n</sub> : タイマ/カウンタn溢れ割り込み要求フラグ (Timer/Counter n Overflow Flag)**

TOVnビットは**タイマ/カウンタ(TCNTn)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、TOVn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TOVnは解除(**0**)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSKn)の**タイマ/カウンタn溢れ割り込み許可(TOIE<sub>n</sub>)ビット**、TOVnが設定(**1**)されると、タイマ/カウンタn溢れ割り込みが実行されます。

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

### 18.1. 特徴

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

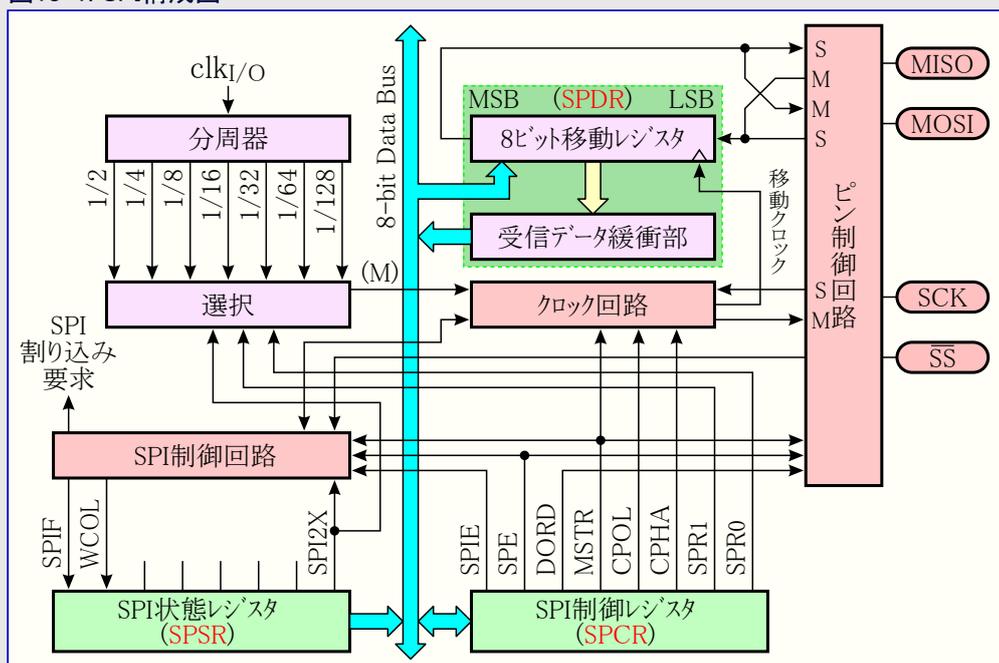
### 18.2. 概要

直列周辺インターフェースはATmega8HVA/16HVAと様々なAVRデバイスや周辺デバイス間の高速同期データ転送を許します。直列周辺インターフェースの簡略構成図は図18-1.で示されます。

27頁の「PRR0 - 電力削減レジスタ」のPRSPIビットはSPI部を許可するために0を書かれなければなりません。

SPIでの主装置と従装置のCPU間相互接続は図18-2.で示されます。このシステムは2つの移動レジスタと主装置ク

図18-1. SPI構成図



注: SPIピン配置については2頁の「ピン配置」と46頁の表15-5.を参照してください。

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

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

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

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

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

SPIが許可されると、MOSI, MISO, SCK, SSピンのデータ方向は表18-1.に従って無視されます。自動的なポート無視のより多くの詳細については44頁の「交換ポート機能」を参照してください。

図18-2. SPI 主装置/従装置の連結

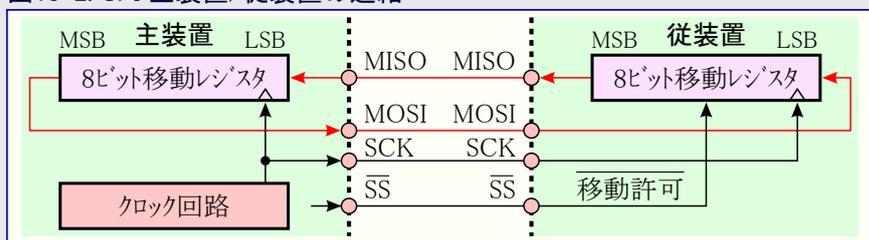


表18-1. SPIピン方向規定

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

注: 使用者定義SPIピンの方向定義方法の詳細記述については46頁の「ポートBの交換機能」を参照してください。

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

#### アセンブリ言語プログラム例

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

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

#### C言語プログラム例

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

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

注: 5頁の「コード例について」をご覧ください。

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

#### アセンブリ言語プログラム例

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

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

#### C言語プログラム例

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

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

注: 5頁の「コード例について」をご覧ください。

## 18.3. $\overline{SS}$ ピンの機能

### 18.3.1. 従装置動作

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

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

### 18.3.2. 主装置動作

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

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

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

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

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

## 18.4. データ転送形式

直列データに関してはSPI制御レジスタ(SPCR)のSCK位相(CPHA)とSCK極性(CPOL)制御ビットによって決定されるSCK位相と極性で4つの組み合わせがあります。このSPIデータ転送形式は図18-3と図18-4で示されます。データビットは安定のためデータ信号に対して十分な時間を保証するSCK信号の反対端で移動出力と(入力)ラッチが行われます。これは表18-2で行われるように表18-3と表18-4を要約することによって明解にされます。

表18-2. CPOL,CPHA機能動作

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

図18-3. SPIデータ転送形式 (CPHA=0)

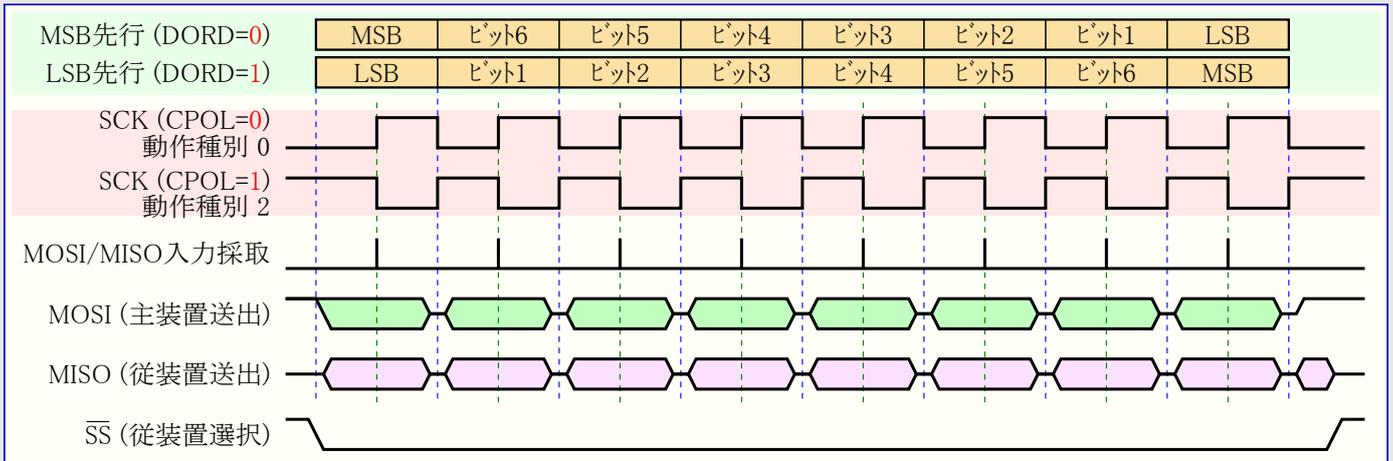
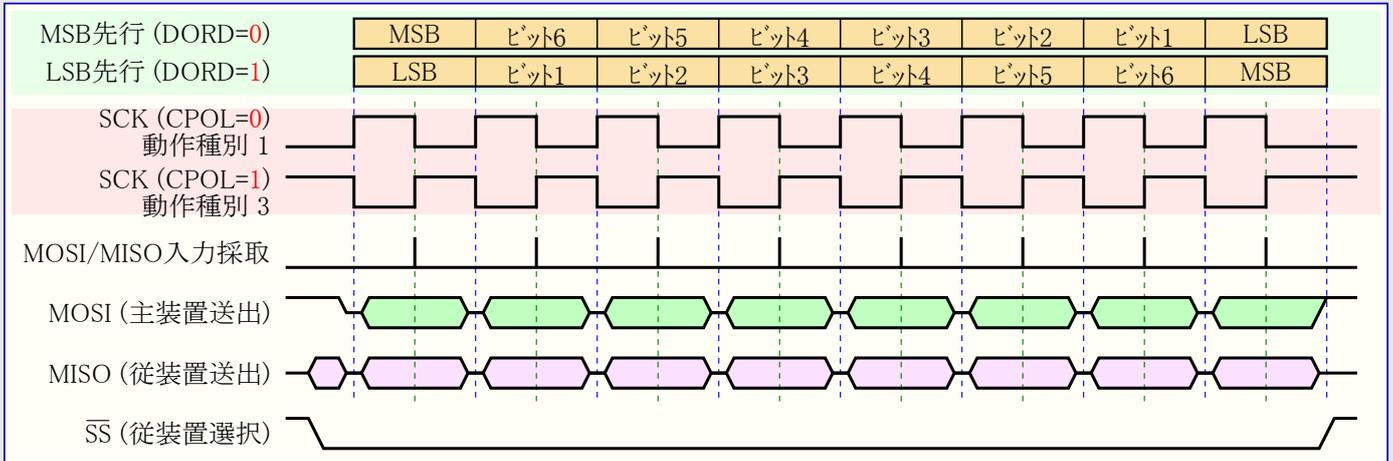


図18-4. SPIデータ転送形式 (CPHA=1)



## 18.5. SPI用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$2C (\$4C)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7 – SPIE : SPI割り込み許可 (SPI Interrupt Enable)**

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されて、SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)されると、このビットがSPI割り込みを実行させます。

• **ビット6 – SPE : SPI許可 (SPI Enable)**

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

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

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

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

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

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

このビットが1を書かれると、アイドル時にSCKはHighです。CPOLが0を書かれると、アイドル時にSCKはLowです。例については図18-3と図18-4を参照してください。CPOL機能は右で要約されます。

表18-3. CPOL機能動作

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

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

このSCK位相選択(CPHA)ビットの設定はデータがSCKの先行(先)端または後行(後)端で採取/(設定)されるかを決めます。例については図18-3と図18-4を参照してください。CPHA機能は右で要約されます。

表18-4. CPHA機能動作

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

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

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

表18-5. SCK速度選択 (fosc=CPUクロック周波数)

SPR1	0	0	1	1
SPR0	0	1	0	1
SPI2X	1	0	1	0
SCK周波数	fosc/2	fosc/4	fosc/8	fosc/16
	fosc/32	fosc/64	fosc/128	

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

ビット	7	6	5	4	3	2	1	0	
\$2D (\$4D)	SPIF	WCOL	-	-	-	-	-	SPI2X	SPSR
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7 – SPIF : SPI割り込み要求フラグ (SPI Interrupt Flag)**

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

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

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

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

これらのビットは予約されており、常に0として読みます。

- **ビット0 – SPI2X : SPI倍速許可** (Double SPI Speed Bit)

このビットが論理1を書かれると、SPIが主装置動作の時にSCK速度(SCK周波数)が倍にされます(表18-5参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定される時にSPIは $f_{OSC}$ (CPUクロック周波数)/4またはそれ以下の動作のみ保証されます。

ATmega8HVA/16HVAのSPIインターフェースはフラッシュメモリやEEPROMの書き換え(読み書き)にも使われます。直列プログラミングと照合については101頁をご覧ください。

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

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

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

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

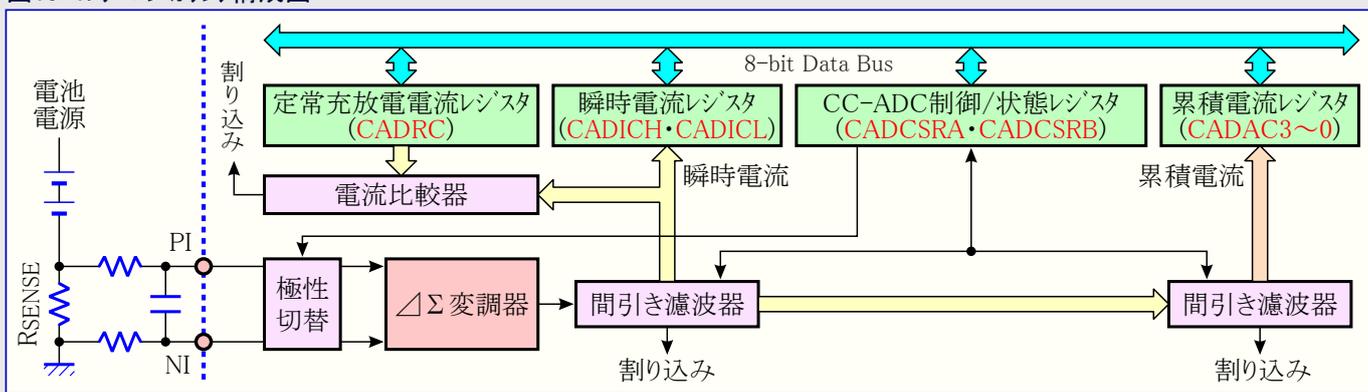
### 19.1. 特徴

- 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
  - 13ビット分解能(符号を含む)
  - 瞬時電流変換完了での割り込み
- 累積電流出力
  - 設定可能な変換時間: 125/250/500/1000ms
  - 18ビット分解能(符号を含む)
  - 累積電流変換完了での割り込み
- 定常電流検出動作
  - 設定変更可能な採取間隔: 250/500/1000/2000ms
- $\pm 100\text{mV}$ の入力電圧範囲
  - $10\text{m}\Omega$ で $\pm 10\text{A}$ 計測可能
- 入力極性切り換えによる変位消去

### 19.2. 概要

ATmega8HVA/16HVAは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。外部検出抵抗器(RSENSE)両端電圧を採取することによって電池セルの入出力電流の流れを追跡するのにCC-ADCが使われます。

図19-1. クーロンカウンタ構成図



標準動作では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。

CC-ADCは特別な定常電流検出動作も提供します。これは小さな充放電電流が流れる時に「パワーセーブ」動作での超低電力を可能にします。

変位消去に関して走行時に入力信号の極性を切り換えることができます。この機能の正しい使用が内部CC-ADC変位を取り去ります。AVR352応用記述をご覧ください。

### 19.3. 標準動作

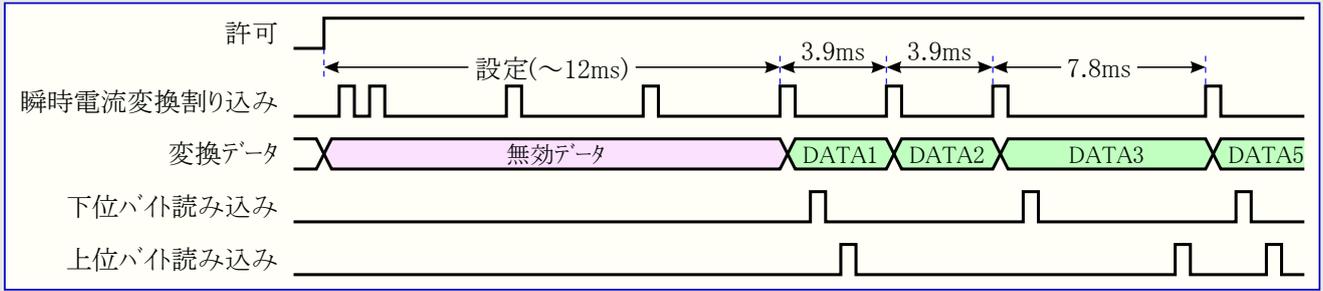
許可時、クーロンカウンタA/D変換器(CC-ADC)は外部検出抵抗器(RSENSE)上の電圧を継続的に計測します。標準変換動作での走行では2つのデータ変換出力が提供されます。

- 瞬時変換結果
- 累積変換結果

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(代表値)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(CADIC<sub>H</sub>, CADIC<sub>L</sub>)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りません。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。

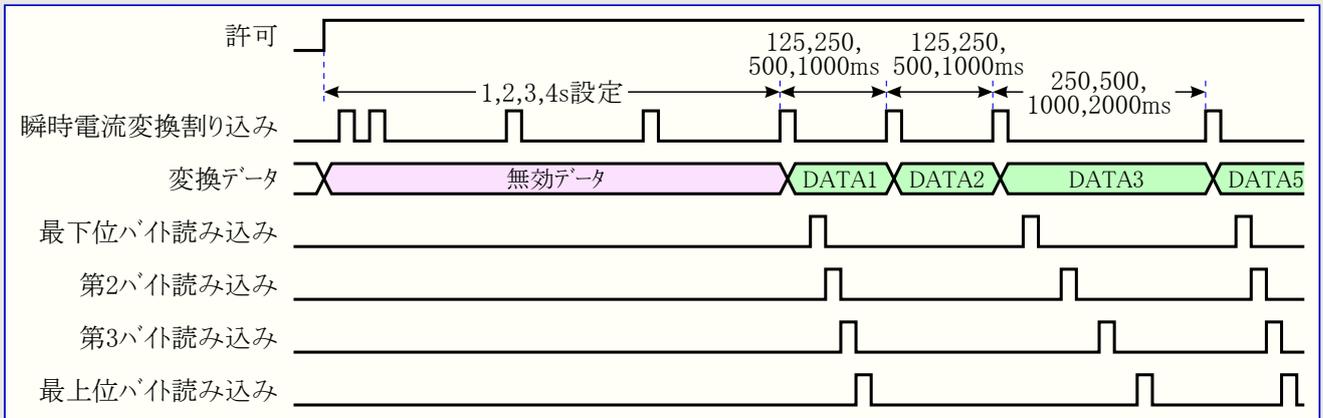
図19-2は制限時間内にDATA3読み込みが完了されないためにDATA4が失われる瞬時電流変換構成図を示します。

図19-2. 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A(CADC<sub>CSRA</sub>)の累積電流変換時間選択(CAD<sub>AS1,0</sub>)ビットによって選んだ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され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(CAD<sub>AC3~0</sub>)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、選んだ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19-3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19-3. 累積電流変換



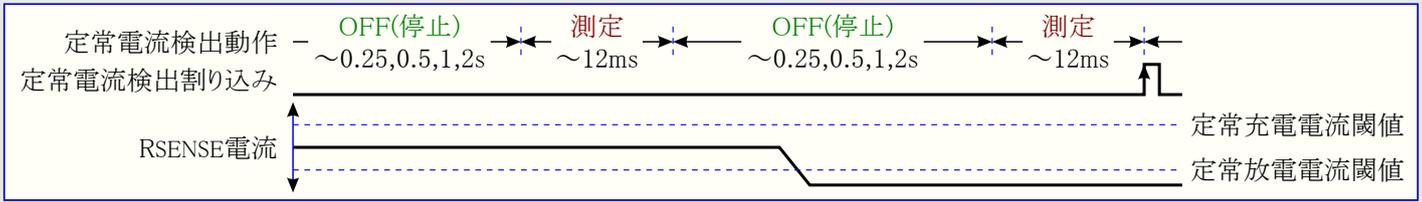
## 19.4. 定常電流検出動作

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)によって特別な定常電流検出動作へ移行します。

この動作ではCC-ADCが定常採取間隔で1回の瞬時電流変換を行う一方、累積電流レジスタは自動的に禁止されます。この採取間隔はCADCSRAの電流採取間隔選択(CADSI,0)ビットへの書き込みによって制御されます。

変換が完了される度毎にその結果は定常電流(CADRC)レジスタで指定した定常充放電閾値と比較されます。割り込みが許可され、電圧が指定された限度以上/以下なら、定常電流検出割り込みが発行されます。図19-4は定常電流検出動作を図解します。

図19-4. 定常電流検出動作(CADSE=1)



定常電流検出は独立した割り込みを持ち、定常電流割り込み許可(CADRCIE)ビットの設定(1)によってこの割り込みが許可されます。この定常電流検出割り込みが休止形態からのCPU起動に使えないことに注意してください。休止形態で定常電流検出機能を使えるようにするには、瞬時電流割り込み許可(CADICIE)ビットの設定(1)によって瞬時電流割り込みが起動元として許可されるべきです。すると、デバイスは単独瞬時電流測定後に休止から起動復帰するでしょう。定常電流検出が起きたかどうかを調べるのに定常電流割り込み要求フラグ(CADRCIF)を読むべきです。

## 19.5. 極性切り換えによる変位(オフセット)消去

CC-ADCは内部変位消去用の極性切り換えを提供します。選んだ時間間隔での採取入力信号の極性切り換えによってCC-ADCの内部電圧変位は出力で打ち消されます。この機能は時間に渡る変位誤差累積からCC-ADCを防ぎます。

## 19.6. 構成設定と使用法

CC-ADCが変換中の間、CPUは休止形態へ移行して割り込みを待つことができます。クーロン計数に対して変換データを追加後にCPUは再び休止へ戻ることができます。これはCPUの作業負荷を減らして低電力動作で過ごすより多くの時間を可能にし、電力消費を低減します。

CC-ADCを使うにはバンドギャップ基準電圧が許可されなければなりません。77頁の「基準電圧と温度感知器」をご覧ください。

CC-ADCはCC-ADC許可(CADEN)ビットが解除(0)されている時に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はパワーオフ動作で自動的に禁止されます。

CADENビットの設定(1)によってCC-ADCが許可された後の最初の3回の瞬時変換は有用なデータを含まず、これは無視されるべきです。これはCADSEビットの解除(0)後、または極性(CADPOL)ビットやCADSEビットの変更後にも適用します。

変換時間と採取間隔は低速RC発振器によって制御され、その実際の周波数に依存します。正確なクーロン計数結果を得るために実際の変換時間が計算されなければなりません。詳細に関しては18頁の「低速RC発振器」を参照してください。

## 19.7. クーロン カウンタA/D変換器用レジスタ

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

ビット (\$E4)	7	6	5	4	3	2	1	0	CADCSRA
	CADEN	CADPOL	CADUB	CADAS1	CADAS0	CADSI1	CADSI0	CADSE	
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7 – 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。パワーオフ動作ではCC-ADCが常に禁止されます。バンドギャップ基準電圧が許可されなければならないことに注意してください。77頁の「基準電圧と温度感知器」をご覧ください。

#### • ビット6 – CADPOL : CC-ADC 極性 (CC-ADC Polarity)

CADPOLビットは $\Delta\Sigma$ 変調器での入力採取極性の変更に使われます。このビットへの1書き込みでの極性は負です。このビットへの0書き込みでの極性は正です。

#### • 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(CADCSRA)、CC-ADC定常電流レジスタ(CADRC)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として読み、どのレジスタも同期化実行されていない時に0として読みます。

#### • 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19-1.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19-1. CC-ADC累積電流変換時間

CADAS1,0	累積電流変換時間	CC-ADCクロック周期数
0 0	125ms	4096
0 1	250ms	8192
1 0	500ms	16384
1 1	1s	32768

**注:** 実際の値は低速RC発振器の実際の周波数に依存します。18頁の「低速RC発振器」をご覧ください。

#### • 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19-2.で示されるように、定常電流検出に対する電流採取間隔を選びます。

表19-2. CC-ADC定常電流採取間隔

CADSI1,0	定常電流採取間隔	CC-ADCクロック周期数
0 0	250ms (+採取時間)	8192 (+採取時間)
0 1	500ms (+採取時間)	16384 (+採取時間)
1 0	1s (+採取時間)	32768 (+採取時間)
1 1	2s (+採取時間)	65536 (+採取時間)

**注:** • 実際の値は低速RC発振器の実際の周波数に依存します。18頁の「低速RC発振器」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。

## 19.7.2. 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として読みます。

- **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も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定常電流レジスタ(CADRC)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。

- **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もCADICIFは解除(0)されます。

## 19.7.3. 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2の補数形式で変換結果を表します。CADIC12~0は(符号を含む)13ビットA/D変換結果で、一方CADIC15~13は符号拡張ビットです。

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。変換完了時、次の変換が完了する前に両レジスタが読まれなければならない、さもなければデータが失われるでしょう。

#### 19.7.4. 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2の補数形式で計測する累積電流を含みます。CADAC17~0は(符号を含む)18ビットA/D変換結果で、一方CADAC31~18は符号拡張ビットです。

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。変換完了時、次の変換が完了する前に4つのレジスタ全てが読まれなければならない、さもなければデータが失われるでしょう。

#### 19.7.5. CADRC – CC-ADC 定常電流レジスタ (CC-ADC Regular Current Register)

ビット (\$E6)	15	14	13	12	11	10	9	8	
	(CADRC7)							(CADRC0)	CADRC
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、またはより大きい絶対値の時に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定常電流値の下位8ビットを定義します。定常電流値の上位側ビットは常に0です。定常電流に対する設定可能な範囲は表19-3.で与えられます。

このCC-ADC定常電流レジスタは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19-3. 定常電流値用設定可能範囲

項目	最小	最大	1段階	単位
電圧	0	6848	26.9	μV
電流	RSENSE=1mΩ	0	6848	mA
	RSENSE=5mΩ	0	1370	
	RSENSE=10mΩ	0	685	

## 20. 電圧A/D変換器 (5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
- 519 $\mu$ s変換時間/1MHz clkVADC
- 電池セル電圧測定用の2つの差動入力チャネル
- 3つのシングルエンド入力チャネル
- 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

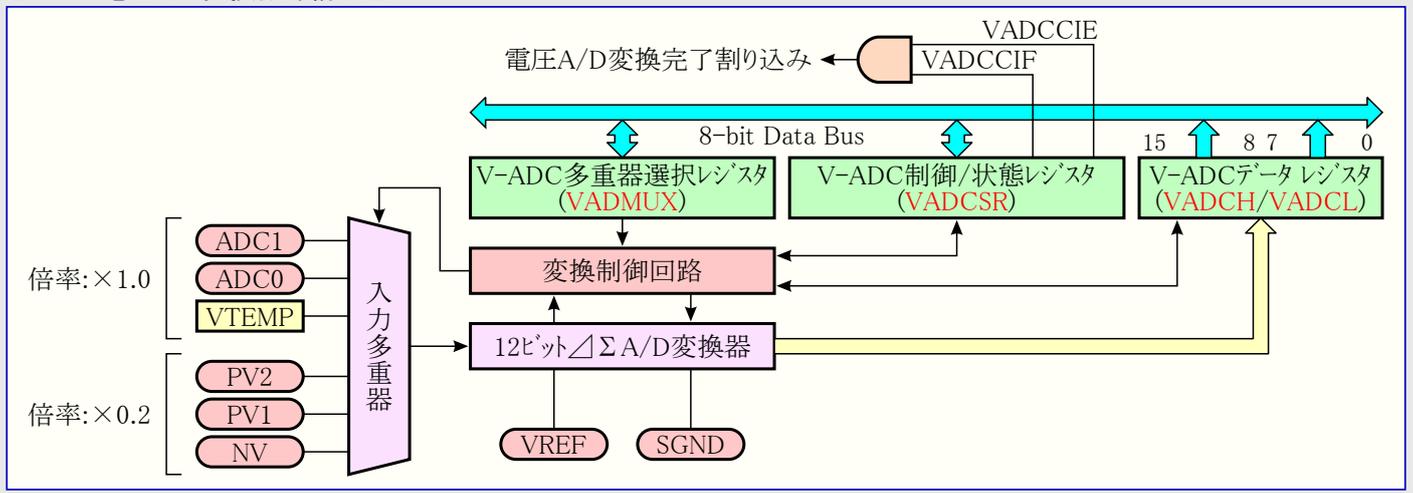
### 20.2. 概要

ATmega8HVA/16HVAは12ビット $\Delta\Sigma$ A/D変換器が特徴です。

この電圧A/D変換器(V-ADC)は入力多重器を通して異なる5つの信号源に接続されます。電池セル電圧測定用に2つの差動チャネルがあります。これらのチャネルは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3つのシングルエンドチャネルがあります。1つのチャネルは内部温度感知器(VPTAT)測定用で、2つのチャネルはADC0とADC1での電圧測定用です。

V-ADCが使われないなら、PRR0のPRVADCFビットに1を書くことによって電力消費を最小にできます。PRVADCFビット使用法の詳細に関しては27頁の「PRR0 - 電力削減レジスタ」をご覧ください。

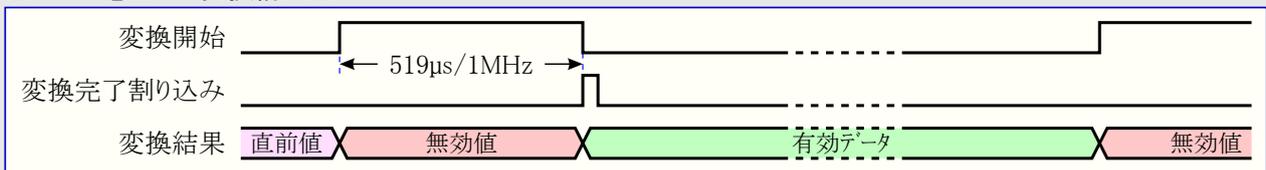
図20-1. 電圧A/D変換器部構成図



### 20.3. 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が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」と「パワーオフ動作」で自動的に停止されます。バンドキ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。78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、アナログ入力チャネルが先に選ば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、選んだチャ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。変換進行中の時のV-ADCデータレジスタ(VADCH/VADCL)は無効です。V-ADC変換中にシステムクロック前置分周器が変更された場合、変換は中止されます。変換進行中の間に違うデータチャネルが選ばれると、そのチャ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519 $\mu$ s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。

### 20.3.1. V-ADC動作PA1,PA0構成設定

ADC0またはADC1のシングルエンドチャンネルの1つがV-ADCのアナログ入力として使われる時に、PA0またはPA1のどちらかが信号接地(SGND)として使われます。入力チャンネルとしてADC0/1が選ばれると、PA1/0が自動的にSGNDへ切り替えられます。

SGNDとしてのPA1及びPA0の使用は109頁の「動作回路」で示されるサーミスタ形態に有効です。RT1とRT2の両サーミスタは共通の分圧抵抗(R1)を通して各々PA0とPA1に接続されます。

PA0とPA1の両方はADC入力として使われる時に非常に高い入力インピーダンスを持ち、それは109頁の「動作回路」で示される形態で2つのサーミスタ接続を可能にします。けれども、高精度が必要とされた場合に入力インピーダンスは制限され、PA0とPA1間に1つのサーミスタだけが接続されるべきです。2つのサーミスタが接続された場合、その形態は次のとおりです。

- RT1測定時、PA1が入力チャンネルとして使われるべきで、PA0は自動的にSGNDへ切り替えられます。
- RT2測定時、PA0が入力チャンネルとして使われるべきで、PA1は自動的にSGNDへ切り替えられます。

## 20.4. 電圧A/D変換器用レジスタ

### 20.4.1. 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として読みます。

- 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
0 0 0 0	(予約)	-	
0 0 0 1	PV1-NV	×0.2	電池セル1
0 0 1 0	PV2-PV1	×0.2	電池セル2
0 0 1 1	(予約)	-	
0 1 0 0	VTEMP	×1.0	温度感知器 (注)
0 1 0 1	ADC0	×1.0	
0 1 1 0	ADC1	×1.0	
0111~1111	(予約)	-	

注: 識票列に格納された校正値使用時に最高精度を得るためにVTEMPは活動動作で測定されなければなりません。

### 20.4.2. 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として読みます。

- 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。78頁の「基準電圧校正レジスタ(BGCCR)」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書き込みが選んだ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として読みます。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 : V-ADC 変換完了割り込み許可 (V-ADC Conversion Complete Interrupt Enable)

本ビットが1を書かれて、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。

### 20.4.3. VADCH, VADCL – 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、上位/下位両方の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るべきです。

#### • 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95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。

$$\text{セルn電圧(mV)} = \frac{(\text{セルnA/D変換結果} - \text{セルn変位}) \times \text{セルn利得校正値}}{16384}$$

ADCnの電圧は次式によって与えられます。

$$\text{ADCn(mV)} = \frac{1}{10} \times \frac{(\text{セルnA/D変換結果} - \text{ADCn変位}) \times \text{ADCn利得校正値}}{16384} \quad (\text{訳注}) \text{ 頁下参照}$$

Vtemp変換実行時、この結果は識票列内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95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。

$$\text{絶対温度:T(K)} = \frac{\text{VtempのA/D変換結果} \times \text{PTAT校正値}}{16384}$$

### 20.4.4. 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	
	-	-	-	-	-	-	PA1DID	PA0DID	DIDR0
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1,0 – PANDI : V-ADC PAピン デジタル入力禁止 (PA1DID, PA0DID Digital Input Disable)

このビットが論理1を書かれると、対応するポートA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PA1/0ピンに印加されて、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。

(訳注) ADCn電圧計算式は1/10になっていますが、AVR353応用記述では10倍になっており、矛盾します。これはどちらかが誤っているものと思われる。

## 21. 基準電圧と温度感知器

### 21.1. 特徴

- 1.100Vの精密基準電圧
- 内部温度感知器
- 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

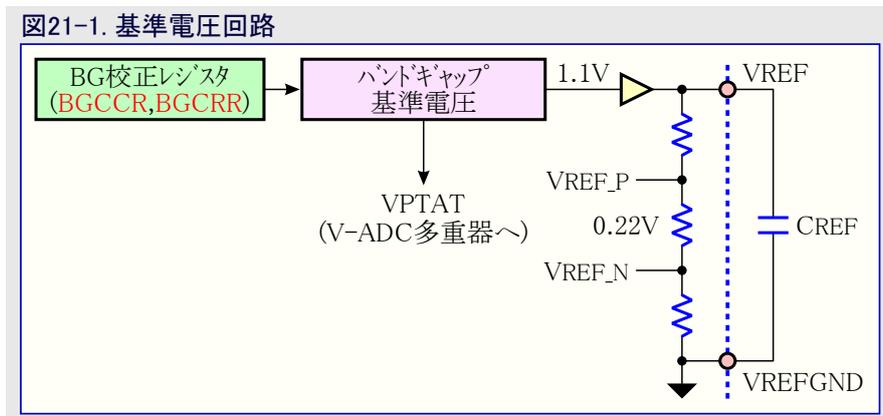
### 21.2. 概要

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ATmega8HVA/16HVA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)は電流検出入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。

ATmega8HVA/16HVA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 $T_{HOT}$ °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 $T_{HOT}$ °Cで実行され、その結果は識票列内に格納されます。その $T_{HOT}$ の値は識票列で得られます。詳細に関しては95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。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-10~70°Cでの90ppm/°C未満が保証されます。基準電圧校正レジスタ(BGCCRR)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。

ATmega8HVA/16HVAにはダイ温度監視用のチップ内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、測定したVPTAT電圧は識票列内に格納されたVPTAT工場校正値で尺度補正されなければなりません。詳細に関しては95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。

図21-1. 基準電圧回路



## 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	
	BGD	-	BGCC5	BGCC4	BGCC3	BGCC2	BGCC1	BGCC0	BGCCR
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 – BGD : ハントギャップ基準電圧禁止 (Bandgap Disable)

BGDビットの1設定がハントギャップ基準電圧を禁止します。このビットは電圧A/D変換器(V-ADC)、クーロンカウンタA/D変換器(CC-ADC)、または電池保護が許可される前に設定(1)されなければならない、これらの単位部のどれかが許可されている間中、解除(0)に留まらなければなりません。

#### • 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 : ハントギャップ基準電圧校正值 (BG Calibration of PTAT Current)

これらのビットはハント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。

BGCCビットの更新は電圧調整器出力電圧と低電圧検出(BOD)基準の両方に影響を及ぼします。BODは新しい検出基準へ素早く達し、一方電圧調整器は外部蓄電コンデンサのため、もっとゆっくりと出力電圧を調整します。従って、値が或る段階量よりも多く増された場合、BOD基準が調整器出力電圧以上に上昇するかも知れず、そして誤ったBODリセットが起きるでしょう。BGCCビットは1段階毎での更新が推奨されます。各段階間で新しい値に達するのを電圧調整器で可能とするには、BGCC値の各更新間に20 $\mu$ sの遅延を追加すべきです。

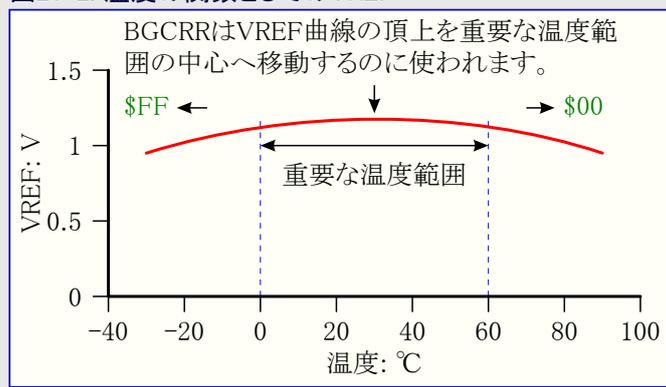
### 21.3.2. BGCR – 基準電圧温度係数校正レジスタ (Bandgap Calibration R Register)

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								
初期値	0	0	0	0	1	1	1	1	

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

これらのビットはハント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. 電圧調整器

### 22.1. 特徴

- 3.3Vの固定出力電圧
- VFET電圧に依存した昇圧または直線的調節の自動選択
- 2セル応用に対して固定直線的調節が選択可能
- VREGをLowに引くことなくVFETでの大きな電圧低下を許す電池パック短絡動作

### 22.2. 概要

電圧調整器は昇圧と直線を結合した電圧調整器です。これは大きな範囲の入力電圧に対して効率的に使われることを電圧調整器に許します。

外部コンデンサとの組み込みチャージポンプは1.8~9.0V範囲の入力電圧に対して一定出力電圧を保つ直線調整器と結合されます。

図22-1は昇圧/直線結合動作に対する外部部品との電圧調整器構成図を示します。図22-2は1セル動作に対するVFET入力電圧の関数としてのVREG安定化電圧を示します。VFETが充分高い時に調整器は自動的に直線動作へ切り替えます。VFETが或るレベル以下に低下した時に調整器は自動的に昇圧調整に戻ります。初期化間の各種リセット元と停止も示されます。

図22-3は2セル応用を意図した直線動作だけに対する外部部品との電圧調整器構成図を示します。直線動作だけでの入力電圧は3.6~9.0Vです。この場合では外部コンデンサの必要がなく、CF1NはGNDに接続されるべきです。図22-4はこの動作を図解します。

電池パック短絡の場合、調整器の入力電圧が素早く低下するでしょう。それが最低動作電圧以下に低下した場合、電圧調整器は内部または外部の回路にもはや供給できません。けれども、出力電圧はこれに付随して下へ引かれず、CREG外部コンデンサの容量と総電流消費によって与えられる時間と同じ期間、コンデンサが回路に(電力を)供給できます。BODリセットを避けるためにVREGはBOD閾値以上に留らなければなりません。VREGが3.3Vに等しく、BOD基準が2.9Vの時に電池パック短絡が起きた場合、チップは次式によって与えられた時間の間、動作を継続することができます。

$$t = \frac{c \Delta v}{I_{AVG}} = \frac{C_{REG} \times 0.4V}{I_{AVG}}$$

ここで $I_{AVG}$ はCREGからの平均引き込み電流を表します。CREG=2.2μFと $I_{AVG}$ =100μAに対するこの時間は8.8msに等しくなります。電圧調整監視器は短絡回路が起きた場合を検知して最小 $I_{AVG}$ をソフトウェアで可能にします。

深く過放電したセルを充電する時にFET駆動部は深下電圧回復(DUVR)動作で操作されます。90頁の「FET駆動部」をご覧ください。この動作ではVFETピンでの適切な動作電圧を保証するために適合する電圧降下が充電FETを介して展開されます。これはセルが安全なセル電圧に達する前の急速充電動作での充電器設定なしに0V充電の間のチップの標準動作を保証します。

図22-1. 昇圧/直線結合動作の電圧調整器構成図

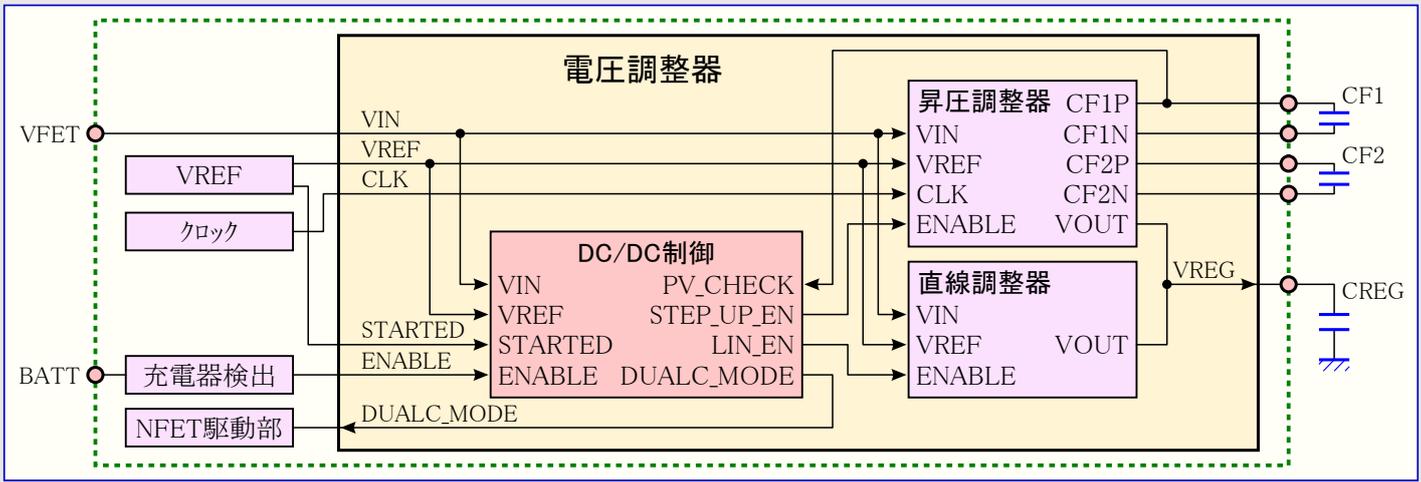


図22-2. 1セル動作に対する入力電圧昇降の関数としての電圧調整器とリセット信号

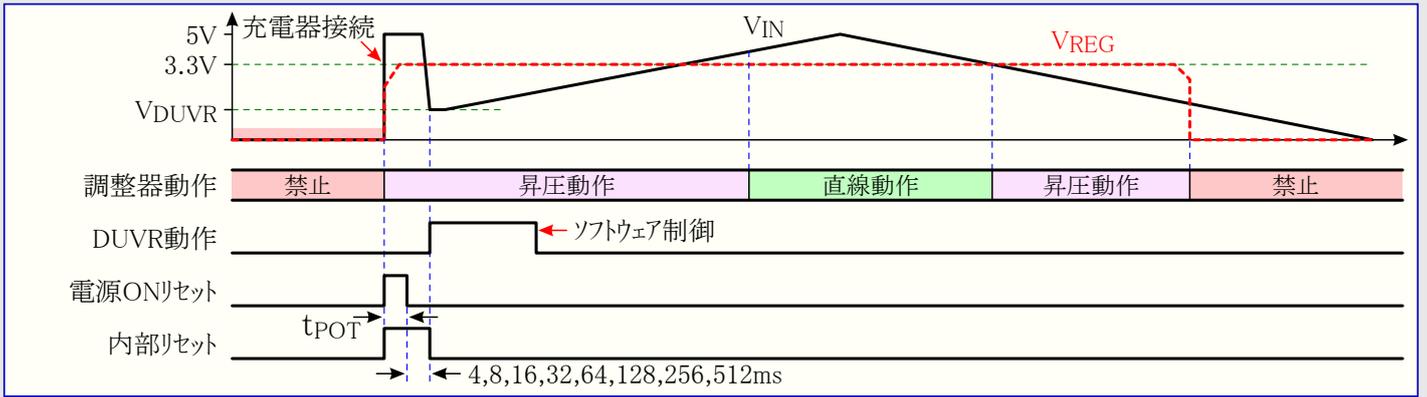


図22-3. 直線動作だけの電圧調整器構成図

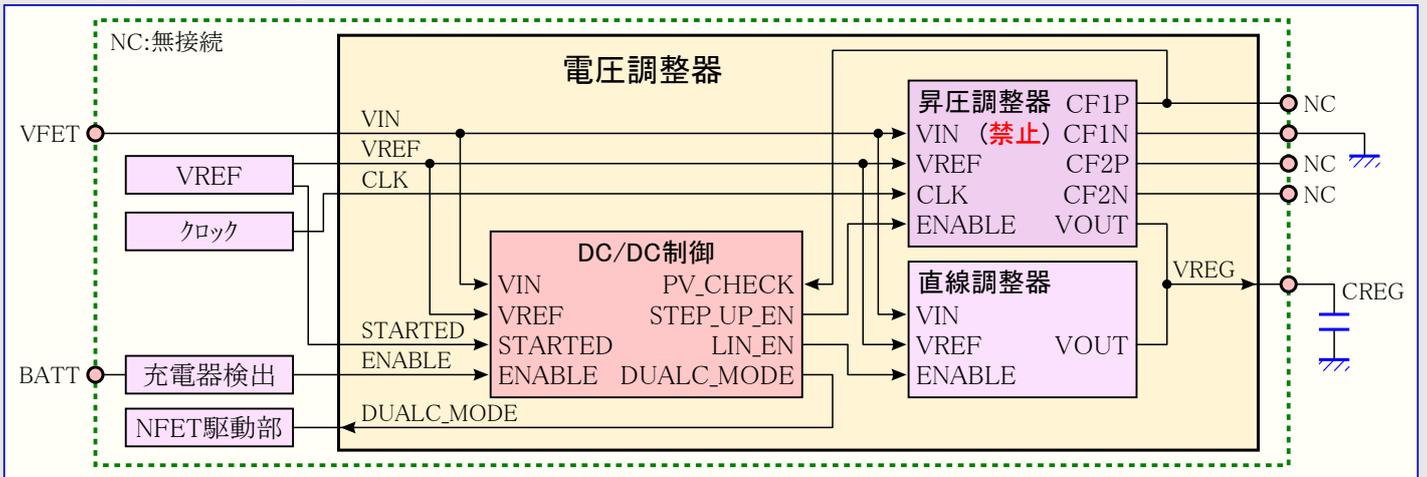
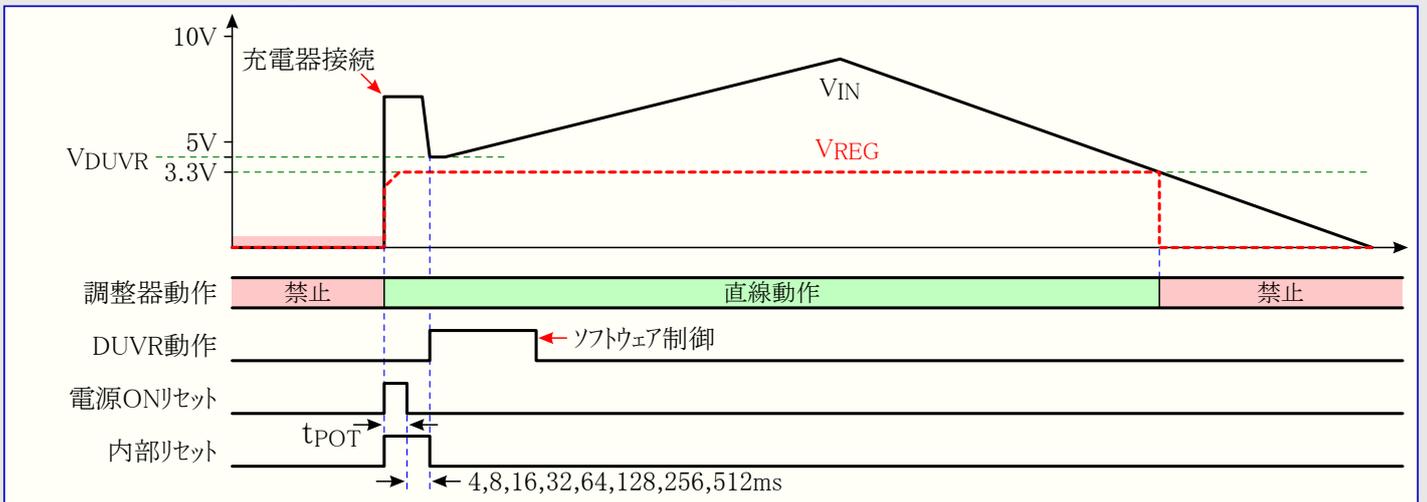


図22-4. 2セル動作に対する入力電圧昇降の関数としての電圧調整器とリセット信号



## 22.3. 電圧調整監視器

この単位部は電圧調整器の動作状態を監視します。VFETでの電圧が調整器短絡回路レベル(RSCL、111頁の「電気的特性」をご覧ください)以下に低下した場合、電圧調整器は電池パック短絡動作に移行します。この動作では電圧調整器出力での急速な低下を避けるためにVFETがVREGから切断されます。電圧調整器がこの動作へ移行すると、チップは外部蓄電コンデンサ(CREG)から全面給電されます。これは例えばVFET電圧が電圧調整器動作に低すぎても、低電圧検出(BOD)リセットへの移行なしに或る時間の動作をチップに許します。

電圧調整器動作状況レジスタ(ROCR)の電圧調整器警告割り込み許可(ROCWIE)ビットが設定(1)されているなら、調整器が電池パック短絡動作へ移行する時に割り込みが発行されます。これは電力消費低減を行う動作を許し、従ってチップ給電に使われ得るCREGの時間を延長します。代表的な短絡回路状態では高電流消費の結果としてVFETが低下し、電池保護部がFETを禁止すると直ぐに回復します。従って、CREGはハードウェアまたはソフトウェアのどちらかによってFETが禁止されるまでBODリセットへの移行なしでチップが動作を維持できるような容量であるべきです。電圧供給器が電池パック短絡動作へ移行する時の最小電力消費のため、チップは電圧調整器警告割り込み(ROCWIF)検知後に可能な限り早くパワーセーブ休止動作へ移行すべきです。ウォッチドッグタイマは安全と考察される時間後にCPUを起動するように構成設定されるべきです。強化型ウォッチドッグタイマの使用に関してはAVR132応用記述をご覧ください。そしてソフトウェアは電圧調整器動作状態(ROCS)フラグを調べるべきです。ROCSフラグが解除(0)なら、標準動作を再開できるかもしれません。

## 22.4. 電圧調整器用レジスタ

### 22.4.1. ROCR – 電圧調整器動作状況レジスタ (Regulator Operating Condition Register)

ビット (\$C8)	7	6	5	4	3	2	1	0	ROCR
	ROCS	–	–	–	–	–	ROCWIF	ROCWIE	
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 – ROCS : 電圧調整器動作状態フラグ (ROC Status)**

このフラグは電圧調整器が電池パック短絡動作で作動している時に設定(1)され、さもなくば解除(0)されます。

- **ビット6~2 – Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

- **ビット1 – ROCWIF : 電圧調整器警告割り込み要求フラグ (ROC Warning Interrupt Flag)**

ROCWIFフラグは電圧調整器の電池パック短絡動作移行時の電圧調整器警告反応時間内で設定(1)されます。このフラグは論理1書き込み、または対応する割り込み処理ベクタ実行でハードウェアによって解除(0)されます。

- **ビット0 – ROCWIE : 電圧調整器警告割り込み許可 (ROC Warning Interrupt Enable)**

ROCWIEビットは電圧調整器警告割り込み要求フラグによって引き起こされる割り込みを許可(=1)します。

## 23. 電池保護

### 23.1. 特徴

- 短絡回路保護
- 放電過電流保護
- 充電過電流保護
- 放電高電流保護
- 充電高電流保護
- 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

### 23.2. 概要

電流/電池保護回路(CBP)は充放電電流を監視し、高電流、過電流、または短絡状態が検出された場合にC-FETとD-FETを禁止します。これらには設定可能な5つの異なる検出値、**放電高電流検出値**、**充電高電流検出値**、**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。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3つの異なる遅延、**高電流反応時間**、**過電流反応時間**、**短絡反応時間**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23-1.で与えられます。

消費電力を減少するために放電高電流保護、放電過電流保護、短絡保護はD-FETが禁止されると、自動的に非活動(禁止)にされます。充電高電流保護と充電過電流保護はC-FETが禁止される時に禁止されます。けれども、チップが深下電圧回復(DUVR)動作で動いている時に充電高電流保護と充電過電流保護が決して自動的に禁止されないことに注意してください。

電流/電池保護回路(CBP)はPI/NI入力ピンでの分路抵抗電圧を採取してセル電流を監視します。差動増幅器が適切な利得でこの電圧を増幅します。この差動増幅器からの出力はアナログ比較器による設定可能なチップ上基準電圧と正確に比較されます。分路抵抗電圧が対応する保護反応時間よりも長い時間の間、検出値以上なら、チップは電流保護を活性(有効)にします。超低電力(ULP)発振器によってクロック駆動される採取システムが高電流、過電流、短絡の保護に使われます。これは確かなクロック元、変位(オフセット)除去、低消費電力を保証します。

表23-1. 電池保護種別の効果

電池保護種別	割り込み要求	C-FET	D-FET	MCU動作
短絡保護	移行時	禁止	禁止	動作継続
放電過電流保護	移行時	禁止	禁止	動作継続
充電過電流保護	移行時	禁止	禁止	動作継続
放電高電流保護	移行時	禁止	禁止	動作継続
充電高電流保護	移行時	禁止	禁止	動作継続

### 23.3. 短絡保護

短絡保護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。PI/NIピンでの電圧が**短絡反応時間**よりも長い時間、**短絡検出値**以上なら、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、外部のC-FETとD-FETは禁止され、電流保護計時器が始動されます。この計時器はC-FETとD-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。

### 23.4. 放電過電流保護

PI/NIピンでの電圧が過電流保護反応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。

放電過電流保護が稼動(状態)にされると、外部のC-FETとD-FETは禁止され、電流保護計時器が始動されます。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。

### 23.5. 充電過電流保護

PI/NIピンでの電圧が**過電流保護反応時間**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、外部のC-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。

## 23.6. 放電高電流保護

PI/NIピンでの電圧が高電流保護反応時間よりも長い時間、放電高電流検出値以上なら、チップは放電高電流保護を稼動(状態)にします。

放電高電流保護が稼動(状態)にされると、外部のC-FETとD-FETは禁止され、電流保護計時器が始動されます。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、放電高電流保護が再び稼動(状態)にされるでしょう。

## 23.7. 充電高電流保護

PI/NIピンでの電圧が高電流保護反応時間よりも長い時間、充電高電流検出値以上なら、チップは充電高電流保護を稼動(状態)にします。

充電高電流保護が稼動(状態)にされると、外部のC-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、充電高電流保護が再び稼動(状態)にされるでしょう。

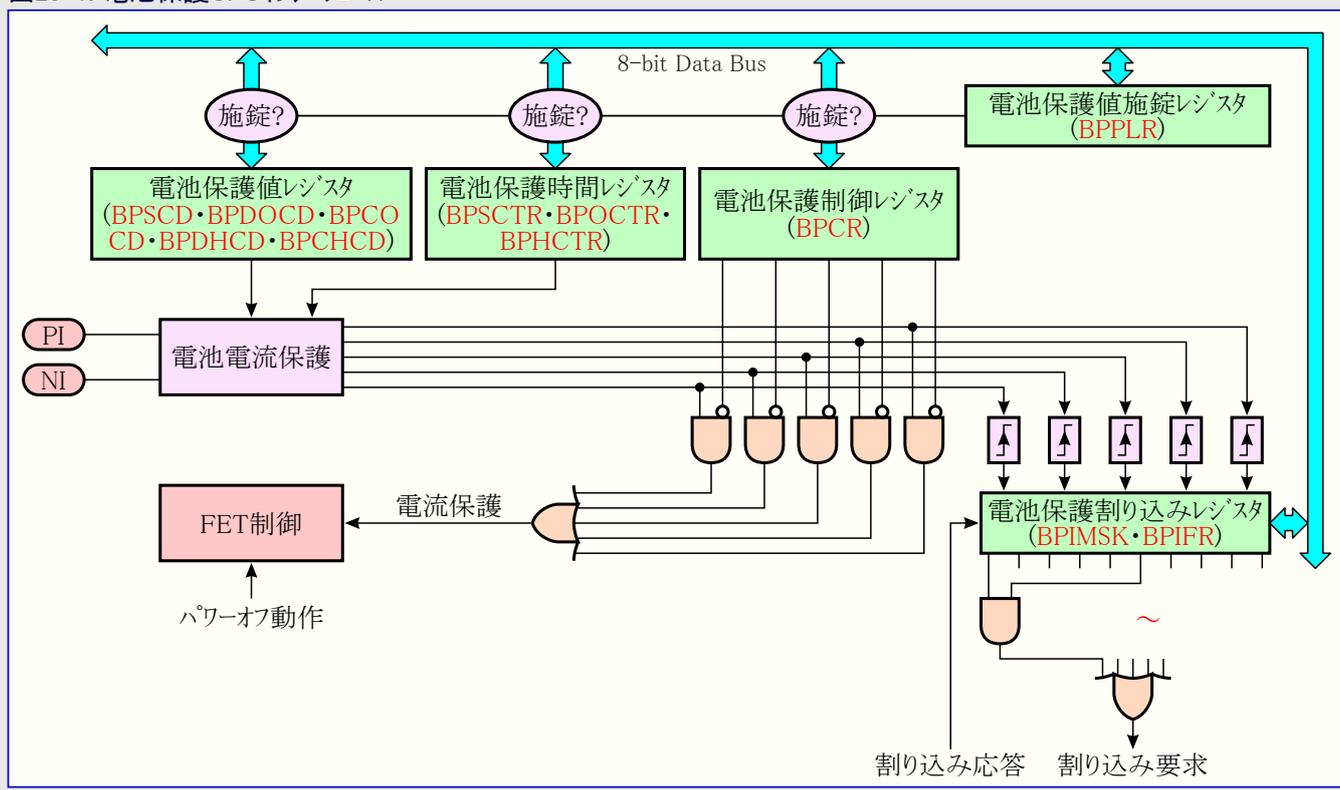
高電流、過電流、短絡の保護の値は各種電池型式を調節するように設定可能です。これらの値はI/Oレジスタへの書き込みによって設定されます。この値レジスタ群は初期設定後に施錠でき、次のハードウェアリセットまで更なるどんな更新も禁止します。

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。

## 23.8. 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23-1.で図解されます。

図23-1. 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。

## 23.9. 電池保護用レジスタ

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

ビット (\$FE)	7	6	5	4	3	2	1	0	
	-	-	-	-	-	-	BPPL	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として読みます。

- **ビット1 – BPPL** : 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)

電池保護制御レジスタ(BPCR)、電池短絡保護時間レジスタ(BPSCTR)、電池過電流保護時間レジスタ(BPOCTR)、電池高電流保護時間レジスタ(BPHCTR)、電池短絡保護検出値レジスタ(BPSCD)、電池放電過電流保護検出値レジスタ(BPDOCD)、電池充電過電流保護検出値レジスタ(BPCOCD)、電池放電高電流保護検出値レジスタ(BPDHCD)、電池充電高電流保護検出値レジスタ(BPCHCD)の電池保護レジスタ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(BPPL)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BPPLに論理0、BPPLに論理1を書いてください。

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

ビット (\$FD)	7	6	5	4	3	2	1	0	
	-	-	-	SCD	DOCD	COCD	DHCD	CHCD	BPCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	1	0	0	0	0	0	

- **ビット7,6 – Res : 予約 (Reserved Bits)**

これらのビットは予約されており、常に0として読みます。

- **ビット5 – Res : 予約 (Reserved Bits)**

このビットは予約されており、常に1として読みます。

- **ビット4 – 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3 – DOCD : 放電過電流保護禁止 (Discharge Over-current Protection Disable)**

このDOCD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2 – COCD : 充電過電流保護禁止 (Charge Over-current Protection Disable)**

このCOCD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。

- **ビット1 – DHCD : 放電高電流保護禁止 (Discharge High-current Protection Disable)**

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

- **ビット0 – CHCD : 充電高電流保護禁止 (Charge High-current Protection Disable)**

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

**注:** クロック領域間での値同期化のため、BPCRレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPCRレジスタ書き込みも無視されます。

## 23.9.3. BPSCTR – 電池短絡保護時間レジスタ (Battery Protection Short-circuit Timing Register)

ビット (\$FA)	7	6	5	4	3	2	1	0	BPSCTR
	-	SCPT6	SCPT5	SCPT4	SCPT3	SCPT2	SCPT1	SCPT0	
Read/Write	R	R/W							
初期値	0	0	0	1	0	0	0	0	

### • ビット7 – Res : 予約 (Reserved Bits)

このビットは予約されており、常に0として読みます。

### • ビット6~0 – SCPT6~0 : 短絡保護反応時間 (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23-2.で示されるように短絡(確認)時間は62.5 $\mu$ sの段階量で設定することができます。

表23-2. SCPT6~0に対応する短絡遅延時間 (短絡保護反応時間)

SCPT6~0	代表
\$00	(15.5~70.5 $\mu$ s) + Td
\$01	(15.5~70.5 $\mu$ s) + Td
\$02	(78.0~133.0 $\mu$ s) + Td
\$03	(140.5~195.5 $\mu$ s) + Td
}	}
\$7E	(7.83~7.88ms) + Td
\$7F	(7.89~7.95ms) + Td

注: • 実際の値は18頁の「超低電力RC発振器」の実際の周波数に依存します。111頁の「電気的特性」をご覧ください。

- SCPT6~0の初期値は\$10(1ms)です。
- 追加遅延(Td)は保護回路初期化のため、放電FET許可後に予測され得ます。公称超低電力発振器周波数でのこの遅延は最大86 $\mu$ sです。
- クロック領域間での値同期化のため、BPSCTRレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPSCTRレジスタ書き込みも無視されます。

## 23.9.4. BPOCTR – 電池過電流保護時間レジスタ (Battery Protection Over-current Timing Register)

ビット (\$FB)	7	6	5	4	3	2	1	0	BPOCTR
	-	-	OCPT5	OCPT4	OCPT3	OCPT2	OCPT1	OCPT0	
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	1	0	

### • ビット7,6 – Res : 予約 (Reserved Bits)

これらのビットは予約されており、常に0として読みます。

### • ビット5~0 – OCPT5~0 : 過電流保護反応時間 (Over-current Protection Timing)

これらのビットは過電流保護の(確認用)遅延を制御します。表23-3.で示されるように過電流(確認)時間は0.5msの段階量で設定することができます。

表23-3. OCPT5~0に対応する過電流遅延時間 (過電流保護反応時間)

OCPT5~0	代表
\$00	(0.0~0.5ms) + Td
\$01	(0.0~0.5ms) + Td
\$02	(0.5~1.0ms) + Td
\$03	(1.0~1.5ms) + Td
}	}
\$3E	(30.5~31.0ms) + Td
\$3F	(31.0~31.5ms) + Td

注: • 実際の値は18頁の「超低電力RC発振器」の実際の周波数に依存します。111頁の「電気的特性」をご覧ください。

- OCPT5~0の初期値は\$02です。
- 追加遅延(Td)は対応するFET許可後と予測され得ます。これは保護回路の初期化に関連付けられます。放電過電流保護に対して、これは放電FET許可時に適用します。充電過電流保護に対して、これは充電FET許可時にきょうします。公称超低電力発振器周波数でのこの遅延は最大0.1msです。
- クロック領域間での値同期化のため、BPOCTRレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPOCTRレジスタ書き込みも無視されます。

### 23.9.5. BPHCTR – 電池高電流保護時間レジスタ (Battery Protection High-current Timing Register)

ビット (\$FC)	7	6	5	4	3	2	1	0	
	-	-	HCPT5	HCPT4	HCPT3	HCPT2	HCPT1	HCPT0	BPHCTR
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)

これらのビットは予約されており、常に0として読みます。

• ビット5~0 – HCPT5~0 : 高電流保護反応時間 (High-current Protection Timing)

これらのビットは高電流保護の(確認用)遅延を制御します。表23-4.で示されるように高電流(確認)時間は2msの段階量で設定することができます。

表23-4. HCPT5~0に対応する高電流遅延時間 (高電流保護反応時間)

HCPT5~0	代表
\$00	(0.0~2ms) + Td
\$01	(0.0~2ms) + Td
\$02	(2~4ms) + Td
\$03	(4~6ms) + Td
}	}
\$3E	(122~124ms) + Td
\$3F	(124~126ms) + Td

注: • 実際の値は18頁の「超低電力RC発振器」の実際の周波数に依存します。111頁の「電気的特性」をご覧ください。

- HCPT5~0の初期値は\$01です。
- 追加遅延(Td)は対応するFET許可後と予測され得ます。これは保護回路の初期化に関連付けられます。放電高電流保護に対して、これは放電FET許可時に適用します。充電高電流保護に対して、これは充電FET許可時にてきょうします。公称超低電力発振器周波数でのこの遅延は最大0.2msです。
- クロック領域間での値同期化のため、BPHCTRレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPHCTRレジスタ書き込みも無視されます。

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

ビット (\$F5)	7	6	5	4	3	2	1	0	
	SCDL7	SCDL6	SCDL5	SCDL4	SCDL3	SCDL2	SCDL1	SCDL0	BPSCD
Read/Write	R/W								
初期値	1	1	1	1	0	0	1	1	

• ビット7~0 – SCDL7~0 : 短絡検出値 (Short-circuit Detection Level)

これらのビットは表23-5.で定義されるように放電方向での短絡検出用のRSENSE電圧値を設定します。

注: クロック領域間での値同期化のため、BPSCDレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPSCDレジスタ書き込みも無視されます。

### 23.9.7. BPDOCD – 電池保護放電過電流検出値レジスタ (Battery Protection Discharge Over-current Level Register)

ビット (\$F6)	7	6	5	4	3	2	1	0	
	DOCDL7	DOCDL6	DOCDL5	DOCDL4	DOCDL3	DOCDL2	DOCDL1	DOCDL0	BPDOCD
Read/Write	R/W								
初期値	1	1	1	1	0	0	1	1	

• ビット7~0 – DOCDL7~0 : 放電過電流検出値 (Discharge Over-current Detection Level)

これらのビットは表23-5.で定義されるように放電過電流検出用のRSENSE電圧値を設定します。

注: クロック領域間での値同期化のため、BPDOCDレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPDOCDレジスタ書き込みも無視されます。

### 23.9.8. BPCOCD – 電池保護充電過電流検出値レジスタ (Battery Protection Charge Over-current Level Register)

ビット (\$F7)	7	6	5	4	3	2	1	0	
	COCDL7	COCDL6	COCDL5	COCDL4	COCDL3	COCDL2	COCDL1	COCDL0	BPCOCD
Read/Write	R/W								
初期値	1	1	1	1	0	0	1	1	

• ビット7~0 – COCDL7~0 : 充電過電流検出値 (Charge Over-current Detection Level)

これらのビットは表23-5.で定義されるように充電過電流検出用のRSENSE電圧値を設定します。

注: クロック領域間での値同期化のため、BPCOCDレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPCOCDレジスタ書き込みも無視されます。

## 23.9.9. BPDHCD – 電池保護放電高電流検出値レジスタ (Battery Protection Discharge High-current Level Register)

ビット (\$F8)	7	6	5	4	3	2	1	0	BPDHCD
	DHCDL7	DHCDL6	DHCDL5	DHCDL4	DHCDL3	DHCDL2	DHCDL1	DHCDL0	
Read/Write	R/W								
初期値	1	1	1	1	0	0	1	1	

•ビット7~0 – DHCDL7~0 : 放電高電流検出値 (Discharge High-current Detection Level)

これらのビットは表23-5.で定義されるように放電高電流検出用のRSENSE電圧値を設定します。

**注:** クロック領域間での値同期化のため、BPDHCDレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPDHCDレジスタ書き込みも無視されます。

## 23.9.10. BPCHCD – 電池保護充電高電流検出値レジスタ (Battery Protection Charge High-current Level Register)

ビット (\$F9)	7	6	5	4	3	2	1	0	BPCHCD
	CHCDL7	CHCDL6	CHCDL5	CHCDL4	CHCDL3	CHCDL2	CHCDL1	CHCDL0	
Read/Write	R/W								
初期値	1	1	1	1	0	0	1	1	

•ビット7~0 – CHCDL7~0 : 充電高電流検出値 (Charge High-current Detection Level)

これらのビットは表23-5.で定義されるように充電高電流検出用のRSENSE電圧値を設定します。

**注:** クロック領域間での値同期化のため、BPCHCDレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのBPCHCDレジスタ書き込みも無視されます。

表23-5. 全電流検出値用の対応するRSENSE電流での~DL7~0 (RSENSE=10mΩ, VREF=1.100±0.005V, TA=-10~70°C)

~DL7~0	最小	代表	最大	~DL7~0	最小	代表	最大	~DL7~0	最小	代表	最大
\$F3		2.0A		\$FC		6.5A		\$9E		12.0A	
\$F4		2.5A		\$FD		7.0A		\$7C		13.0A	
\$F5		3.0A		\$FE		7.5A		\$7D		14.0A	
\$F6		3.5A		\$DD		8.0A		\$7E		15.0A	
\$F7		4.0A		\$DE		8.5A		\$7F		16.0A	
\$F8		4.5A		\$DF		9.0A		\$5C		17.0A	
\$F9		5.0A		\$BD		9.5A		\$5D		18.0A	
\$FA		5.5A		\$BE		10.0A		\$5E		19.0A	
\$FB		6.0A		\$9D		11.0A		その他		(予約)	

### 23.9.11. BPIMSK – 電池保護割り込み許可レジスタ (Battery Protection Interrupt Mask Register)

ビット (\$F2)	7	6	5	4	3	2	1	0	
	-	-	-	SCIE	DOCIE	COCIE	DHCIE	CHCIE	BPIMSK
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット7~5 – Res : 予約 (Reserved Bits)

これらのビットは予約されており、常に0として読みます。

• ビット4 – 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。

• ビット3 – 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 – DHCIE : 放電高電流保護割り込み許可 (Discharge High-current Protection Activated Interrupt Enable)

このDHCIEビットは放電高電流保護割り込み(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 – CHCIE : 充電高電流保護割り込み許可 (Charge High-current Protection Activated Interrupt Enable)

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

### 23.9.12. BPIFR – 電池保護割り込み要求フラグ レジスタ (Battery Protection Interrupt Flag Register)

ビット (\$F3)	7	6	5	4	3	2	1	0	
	-	-	-	SCIF	DOCIF	COCIF	DHCIF	CHCIF	BPIFR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

• ビット7~5 – Res : 予約 (Reserved Bits)

これらのビットは予約されており、常に0として読みます。

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

一旦短絡保護違反が検知されるとSCIFが設定(1)になります。本フラグはそこへの論理1書き込みによって解除(0)されます。

• ビット3 – DOCIF : 放電過電流保護割り込み要求フラグ (Discharge Over-current Protection Activated Interrupt)

一旦放電過電流保護違反が検知されるとDOCIEが設定(1)になります。本フラグはそこへの論理1書き込みによって解除(0)されます。

• ビット2 – COCIF : 充電過電流保護割り込み要求フラグ (Charge Over-current Protection Activated Interrupt)

一旦充電過電流保護違反が検知されるとCOCIEが設定(1)になります。本フラグはそこへの論理1書き込みによって解除(0)されます。

• ビット1 – DHCIF : 放電高電流保護割り込み要求フラグ (Discharge High-current Protection Activated Interrupt)

一旦放電高電流保護違反が検知されるとDHCIEが設定(1)になります。本フラグはそこへの論理1書き込みによって解除(0)されます。

• ビット0 – CHCIF : 充電高電流保護割り込み要求フラグ (Charge High-current Protection Activated Interrupt)

一旦充電高電流保護違反が検知されるとCHCIEが設定(1)になります。本フラグはそこへの論理1書き込みによって解除(0)されます。

## 24. FET制御

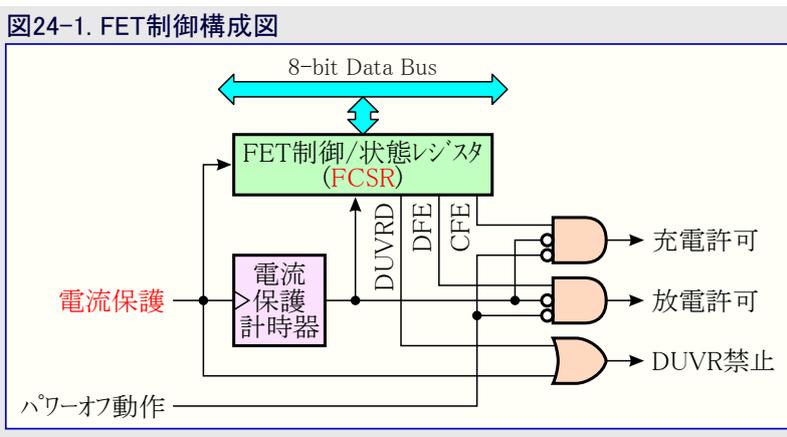
### 24.1. 概要

FET制御は充電FETと放電FETの許可と禁止に使われます。通常、FETはFET制御/状態レジスタ(FCSR)へのソフトウェア書き込みによって許可と禁止が行われます。けれども、高すぎる充放電電流から電池セルを保護するために自律型電池保護回路がソフトウェア設定を無効にする場合があります。CPUは電池保護回路によって禁止されているFETの許可を決して許されないことに注意してください。FET制御は図24-1.で示されます。

電流保護が電池保護回路によって稼動(有効に)されている場合、充電FETと放電FETの両方がハードウェアによって禁止されます。保護が解消されると、ソフトウェアが外部FETを再び許可し得る前に電流保護計時器が1秒の保持時間を保証します。

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。

深く過放電したセルを充電する場合、FET駆動部は深下電圧回復(DUVR)動作で動かなければなりません。セル電圧が受け入れ可能なレベルに上昇すると、FCSRの深下電圧回復禁止(DUVRD)ビットの設定(1)によってソフトウェアで禁止されるべきです。電流保護が活性(有効)の間にC-FETが開放にされるのを避けるため、この状態ではDUVR動作がハードウェアによって自動的に禁止されます。



#### 24.1.1. リセット中のFET禁止

リセットの間、両FETは直ちに禁止され、チップはDUVR動作から抜け出します。リセットが電池保護値設定と無関係にFETの即時禁止を引き起こすのに注意することが重要です。低電圧検出(BOD)リセットが短絡状態の結果として起きるかもしれません。選んだ電池保護時間、実際の電流消費、CREG容量に依存して、FETの禁止を引き起こす電池保護遅延時間が経過する前にBODリセットが起きるかもしれません。

## 24.2. FET駆動部

### 24.2.1. 特徴

- High側のNch-FETに対する適切なゲート駆動生成用のチャージポンプ
- 深く過放電した電池充電中に0Vから標準動作を可能にする深下電圧回復(DUVR)動作

### 24.2.2. 概要

ATmega8HVA/16HVAはFET駆動部を含みます。FET駆動部は1または2セルのLi-ion電池パックでのHigh側の切り換えとして使うNch-FET駆動用に設計されています。FET駆動部の構成図は図24-2.で示されます。

深く過放電したセルの充電時、FET駆動部は深下電圧回復(DUVR)動作で動きます。DUVR動作ではFET駆動部がVFETピンでの電圧を1セル応用で代表的に2.0V、2セル応用で代表的に4.0Vに調整します。これは必要とするVFET動作電圧とセル電圧間の電圧差と等しいドレイン、ソース間電圧で充電FETを操作することによって行われます。セル電圧の増加に連れ、ドレイン、ソース間電圧は充電FETが完全にONとなるまで減少します。詳細に関しては114頁の表29-5.をご覧ください。

標準動作(DUVRD=1)ではVFET供給電圧よりも十分に高くOC/ODを押し上げることによって充電FETと放電FETをONにします。充電FETと放電FETをOFFにするにはOC/ODが素早くLowに引きます。詳細に関しては図24-3.と114頁の表29-5.をご覧ください。

図24-2. FET駆動部構成図

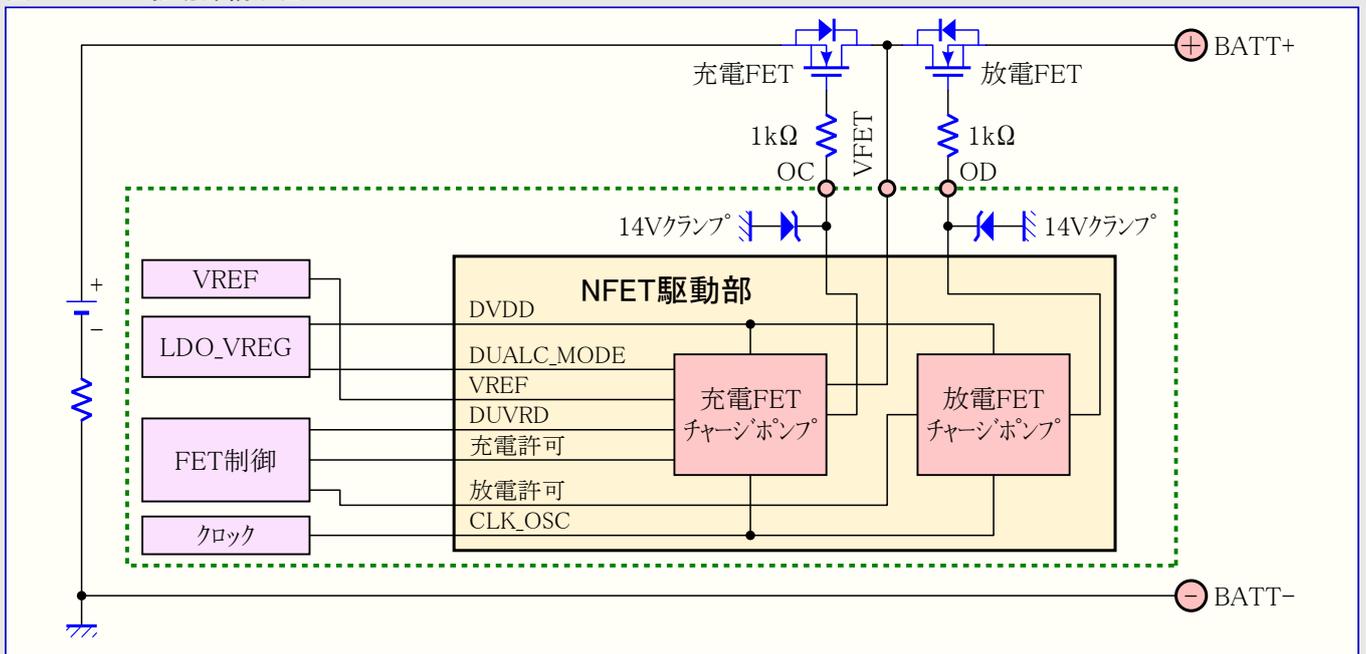
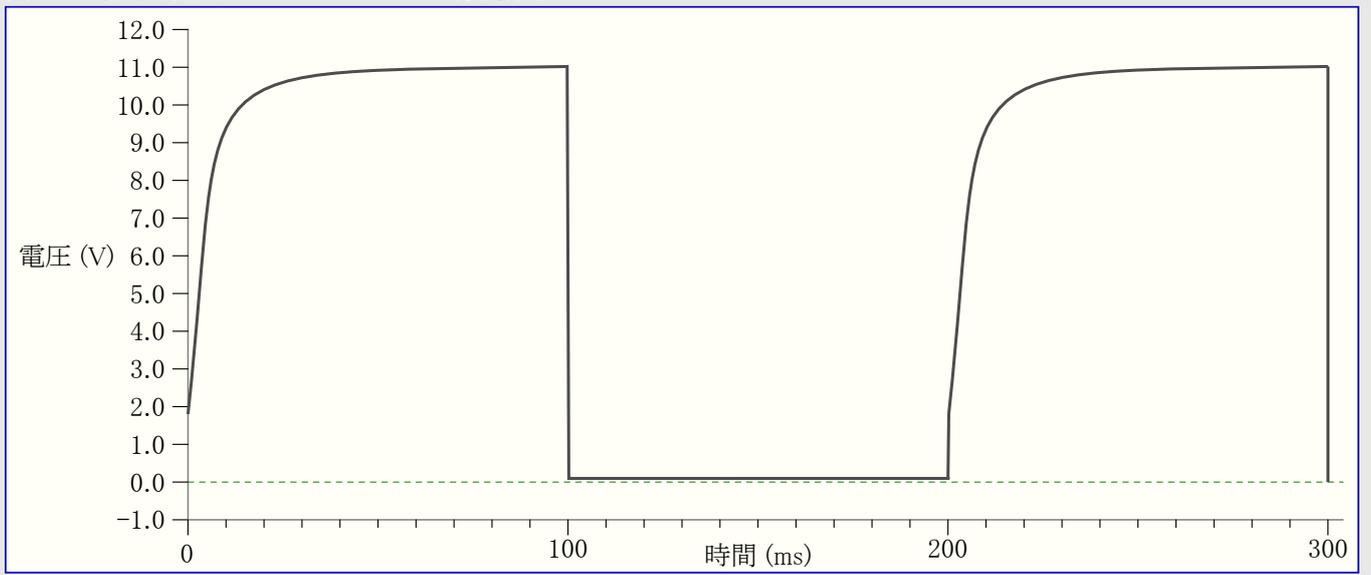


図24-3. 標準動作中のNFETのON/OFF切り換え



## 24.3. DUVR – 深下電圧回復動作

DUVR動作の目的は最低動作電圧以下でセルを充電する間にVFET電圧が最低動作電圧以上であるように充電FETを制御することです。これはセルがチップの最低動作電圧以下に放電されてしまった時に有用です。DUVR動作ではセル電圧とVFET端子間に適切な電圧降下を提供するために充電FETが部分的にONされます。セル電圧が増加するため、充電FETの電圧降下は充電FETが完全にONとなるまで徐々に減少します。これは高いセル電圧に関して充電FETの通常許可(CFE=1)と等価なことを意味します。

ATmega8HVA/16HVAはセルが深い低電圧状態から回復されたことをソフトウェアが検知するまでDUVR動作で動くべきです。セルが深い低電圧状態から回復される時にソフトウェアは最初にCFE=1を設定すべきです。これはセル電圧が最低動作電圧以上なので今や安全です。その後、ソフトウェアはDUVRD=1設定によってDUVR動作を禁止すべきです。

セル電圧が最低動作電圧以上になる前にDUVRDとCFEの両ビットを設定(1)した場合、VFETが低下してチップは低電圧検出(BOD)リセットに移行し、そして充電と放電の両FETをOFFにします。FETのOFF切り換えはVFET電圧を再上昇させ、故にチップはDUVR=0とCFE=0(既定値)でBODリセットから再始動します。これを避けるためにソフトウェアはCFE=1設定の前にV-ADC測定によってセル電圧を常に調べなければなりません。

リセット後、DUVR動作は既定で許可されています。けれども、チップがリセット状態である間、DUVR動作は禁止されています。これは充電過電流保護が動作するまで充電FETがONされないことを保証する安全機能です。これは充電器が接続された時から選んだ起動時間が経過するまで、DUVR動作が禁止されていることを意味します。この期間の間、VFET電圧はDUVR動作での一般的なVFETレベルよりも高くなるでしょう。

DUVR動作についてのより多くの詳細に関してはAVR354応用記述を参照してください。

## 24.4. FET制御用レジスタ

### 24.4.1. 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	
	-	-	-	-	DURVD	CPS	DFE	CFE	FCSR
Read/Write	R	R	R	R	R/W	R	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として読みます。

#### • ビット3 – DURVD : 深下電圧回復動作禁止 (Deep Under-voltage Recovery Disabled)

DURVDが解除(0)されると、FET駆動部は深下電圧回復動作での動きを強制されます。詳細に関しては、「DUVR – 深下電圧回復動作」をご覧ください。電流保護またはまたは内部リセットの最中にFET駆動部がC-FETのON切り換えの試みを避けるため、これらの場合にDURVDビットはハードウェアによって1にされます。このビットが設定(1)されると、FET駆動部のDUVR動作は禁止されます。

#### • ビット2 – 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が活性(有効)の時に設定(1)され、さもなければ解除(0)されます。

#### • ビット1 – 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が許可されます。このビットは電流保護が活性(有効)化される時に電流/電池保護(CBP)回路によって自動的に解除(0)されます。このビットが解除(0)されると、電池保護制御レジスタ(BPCR)の設定に拘らず、短絡回路、放電高電流、放電過電流の保護が禁止されます。

#### • ビット0 – 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が許可されます。このビットは電流保護が活性(有効)化される時に電流/電池保護(CBP)回路によって自動的に解除(0)されます。このビットが解除(0)され、且つDURVDビットが設定(1)されていると、電池保護制御レジスタ(BPCR)の設定に拘らず、充電高電流と充電過電流の保護が禁止されます。DURVDビットが解除(0)されると、CFEビットの状態に拘らず、DUVR動作によって充電FETは許可されます。

**注:** クロック領域間での値同期化のため、FCSRレジスタが書かれる毎に3超低電力(ULP)発振器周期+3 CPUクロック周期の保護時間が必要です。この時間中のどのFCSRレジスタ書き込みも無視されます。

## 25. デバッグWIRE 内蔵デバッグ システム

### 25.1. 特徴

- 完全なプログラムの流れ制御
- RESETピンを除くデジタルとアナログ両方でのチップ全機能のエミュレート
- 実時間(リアルタイム)動作
- シンボリックデバッグ支援 (アセンブリ及びC言語または他の高位言語)
- 無制限数のプログラム中断点(ブレークポイント:ソフトウェア中断点使用)
- 邪魔しない動作
- 実デバイスと同じ電気的特性
- 自動設定システム
- 高速動作
- 不揮発性メモリのプログラミング

### 25.2. 概要

デバッグWIRE内蔵デバッグシステムはCPUでのAVR命令実行、プログラムの流れ制御、各種不揮発性メモリのプログラミングのための1本線の双方向インターフェースを使います。

### 25.3. 物理インターフェース

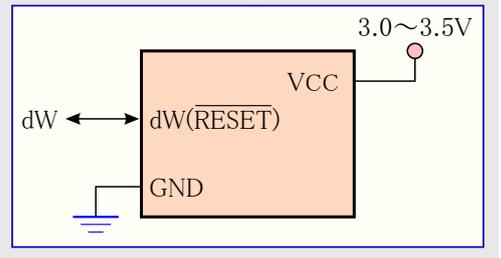
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

図25-1.はエミュレータと許可したデバッグWIREでの対象MCUとの接続の図を示します。システムクロックはデバッグWIREによって影響を及ぼされず、常にOSCSELヒューズで選んだクロック元です。

デバッグWIREが使われるシステムの設計時、正しい動作のために次の注意点が厳守されなければなりません。

- dW/(RESET)線のプルアップ抵抗は10kΩよりも小さくしてはなりません。この抵抗はデバッグWIRE機能の必要条件ではありません。
- RESETピンのVCCへの直接的な接続では動作しません。
- RESETピンに挿入したコンデンサはデバッグWIRE使用時、切断されなければなりません。
- 全ての外部リセット元は切断されなければなりません。

図25-1. デバッグWIRE構成図



### 25.4. ソフトウェア中断点 (ブレークポイント)

デバッグWIREはAVRのBREAK命令によってプログラムメモリの中断点を支援します。AVR Studio®での中断点設定はプログラムメモリにBREAK命令を挿入します。BREAK命令で置換した(元の)命令は保存されます。プログラム実行が継続される時、プログラムメモリから継続される前に保存した命令が実行されます。一時停止(ブレーク)はプログラムにBREAK命令を置くことによって手動で挿入できます。

フラッシュメモリは中断点の変更される度毎に再書き換えされなければなりません。これはデバッグWIREインターフェースを通してAVR Studioによって自動的に操作されます。従って中断点の使用はフラッシュメモリのデータ保持力を低下させます。デバッグ目的に使ったデバイスは最終顧客へ出荷すべきではありません。

### 25.5. デバッグWIREの制限

デバッグWIRE通信(dW)ピンは物理的に外部リセット(RESET)と同じピンに配置されます。従ってデバッグWIREが許可されると、外部リセット元が支援されません。

プログラム(0)にしたDWENヒューズは全休止形態でクロック系のいくつかの部分の走行を許可します。これは休止間中の消費電力を増加します。従ってDWENヒューズはデバッグWIREが使われない場合、禁止されるべきです。

### 25.6. デバッグWIRE用レジスタ

次項はデバッグWIREで使うレジスタを記述します。

#### 25.6.1. DWDR - デバッグWIRE データレジスタ (debugWIRE Data Register)

ビット	7	6	5	4	3	2	1	0	
\$31 (\$51)	(MSB)							(LSB)	DWDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

DWDRはMCU内で走行するプログラムからデバッグへの通信チャネルを提供します。このレジスタはデバッグWIREでだけアクセス可能で、従って通常動作で一般目的レジスタとして使えません。

## 26. フラッシュメモリの自己プログラミング

### 26.1. 概要

本デバイスはMCU自身によるプログラムコードのダウンロードとアップロード用の自己プログラミング機構を提供します。自己プログラミングはフラッシュメモリ内にコードを書き(プログラム)、コードを読み、またはプログラムメモリからコードを読むのに、利用可能なデータインターフェースと関連する規約のどれもが使えます。

プログラムメモリはページ単位形式で更新されます。ページ一時緩衝部へ格納したデータでページを書く前にそのページは消去されなければなりません。ページ一時緩衝部はSPM命令使用時毎の1語(ワード)で満たされ、この緩衝部はページ消去命令前、またはページ消去とページ書き込み操作間のどちらかで満たすことができます。

#### 手段1 (ページ消去前の一時緩衝部格納)

- ページ一時緩衝部を満たしてください。
- ページ消去を実行してください。
- ページ書き込みを実行してください。

#### 手段2 (ページ消去後の一時緩衝部格納)

- ページ消去を実行してください。
- ページ一時緩衝部を満たしてください。
- ページ書き込みを実行してください。

ページの一部の変更だけが必要な場合、消去前にページの残す部分は(例えばページ一時緩衝部に)保存されなければならない、その後改めて書かれます。手段1.を使う場合、初めにページを読んで必要な変更を行い、その後に変更したデータを書き戻すことをユーザーソフトウェアに許す効率的な読み-修正-書き(リード モデファイライト)機能をデバイスが提供します。手段2.が使われる場合、ページが既に消去されているため、格納中の旧データを読むことができません。ページ一時緩衝部は乱順でアクセスできます。ページ消去とページ書き込み操作の両方で使われるページアドレスは同じページをアドレス指定することが非常に重要です。

#### 26.1.1. SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタにアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'00000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGEに書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- ページ消去中、CPUは停止されます。

#### 26.1.2. ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタにアドレス、R1:R0にデータを設定してSPMCSRに'00000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORDの内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのCTPBビット(=1)書き込みによって自動消去します。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。

#### 26.1.3. ページ書き込みの実行

ページ書き込みを行うにはZポインタにアドレスを設定してSPMCSRに'00000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGEに書かれなければなりません。この操作の間、Zポインタの他のビットは0を書かれなければなりません。

- ページ書き込み中、CPUは停止されます。

## 26.2. 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

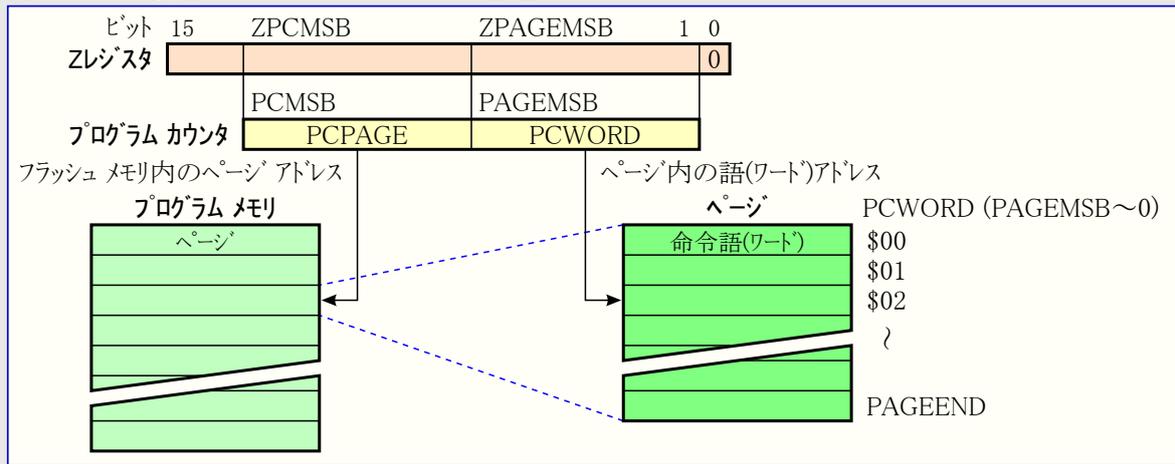
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

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(100頁の表27-6.参照)、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つように扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは次頁の図26-1.で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってソフトウェアはページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。

LPM命令はアドレスを格納するのにZポインタを使います。この命令はフラッシュメモリのバイト単位をアドレス指定するので、Zポインタの最下位ビット(Z0)も使われます。

図26-1. SPM操作中のフラッシュメモリのアドレス指定



注: 図内で使った各変数は100頁の表27-6.で一覧されます。

### 26.2.1. SPM命令での書き込み時のEEPROM書き込みによる妨害

EEPROM書き込み動作がフラッシュメモリへの全ソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(ECCR)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。

### 26.2.2. ソフトウェアからの施錠ビット設定

施錠ビットを設定するには希望するデータをR0に書いてください。R0のビット1,0が解除(0)された場合、SPMCSRでヒューズ/施錠ビット読み込み(RFLB)とSPMENビットが設定(1)された後4クロック周期内にSPM命令が実行されるなら、対応する施錠ビットがプログラム(0)されます。この操作中のZポインタは関係ありませんが、将来との共通性のため、(施錠ビット読み出しに使われるのと同じ)\$0001でZポインタを設定することが推奨されます。将来との共通性のため、施錠ビット書き込み時、R0のビット7~2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。

ビット	7	6	5	4	3	2	1	0
R0	1	1	1	1	1	1	LB2	LB1

フラッシュメモリのアクセスに影響を及ぼす施錠ビットの各種設定法については表27-1.と表27-2.をご覧ください。

### 26.2.3. 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SPM操作許可(SPMEN)とフラッシュ/施錠ビット読み込み(RFLB)ビットを設定(1)してください。SPMENとRFLBビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SPMENとRFLBビットは4 CPU周期内にSPM命令が実行されない場合、SPMCSR書き込み後、6周期で自動的に解除(0)します。SPMCSRは自動的に解除(0)されるまで、更なる書き込みに対して施錠されます。LPM命令はSPMCSR書き込み後、3 CPU周期内に実行されなければなりません。SPMENとRFLBビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	-	LB2	LB1

ヒューズ下位ビットを読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポインタに\$0000を設定してSPMCSRのSPMENとRFLBビットを設定(1)してください。SPMENとRFLBビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ下位ビット(FLB)の値が転送先レジスタに格納されます。ヒューズ下位ビットの配置と詳細な記述については99頁の表27-4.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

プログラム(0)されたヒューズと施錠ビットは0として読みます。非プログラム(1)にされたヒューズと施錠ビットは1として読みます。

## 26.2.4. フラッシュ メモリ データ化けの防止

低VCCの期間中、CPUとフラッシュ メモリの正しい動作に対して供給電圧が低すぎるためにフラッシュ メモリのプログラムが不正にされ得ます。これらの問題はフラッシュ メモリを使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

フラッシュ メモリのプログラム化けは電圧が低すぎる時の2つの状態によって起こされます。1つ目としてフラッシュ メモリへの通常の書き込み手順は正しく動作するための最低電圧が必要です。2つ目として供給電圧が低すぎると、CPU自身が命令を間違えて実行し得ます。

フラッシュ メモリ化けは次の推奨設計によって容易に避けられます(1つは必須)。

- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部**低電圧検出器(BOD)**を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアを**パワーセーブ休止動作**に保ってください。これはCPUが命令の復号と実行を試みるのを防ぎ、SPMCSR 従ってフラッシュ メモリを予期せぬ書き込みから効果的に保護します。

## 26.2.5. 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26-1.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、3 CPU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SPMCSR書き込み後、6周期で自動的に解除(0)し、そしてそれはそれらの周期の間、更なる書き込みに対して施錠されます。LPM命令はSPMCSR書き込み後、3 CPU周期内に実行されなければなりません。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26-1. 識票列アドレス一覧

Zポイントアドレス	識票バイト	Zポイントアドレス	識票バイト
\$0001	高速RC発振器校正バイト (注1)	\$0000	デバイス識票バイト 1 (製造者識別)
\$0003	高速RC発振器校正区分 (注2)	\$0002	デバイス識票バイト 2 (フラッシュ容量)
\$0005	(予約)	\$0004	デバイス識票バイト 3 (デバイス種別)
\$0007	低速RC発振器周期上位 (注3)	\$0006	低速RC発振器周期下位 (注3)
\$0009	低速RC発振器温度変化予測値上位 (注4)	\$0008	低速RC発振器温度変化予測値下位 (注4)
\$000B	低速RC発振器周波数 (注5)	\$000A	超低電力RC発振器周波数 (注6)
\$000D	(予約)	\$000C	(予約)
\$000F	BGCCR校正值@25°C	\$000E	(予約)
\$0011	BGCRR校正值@25°C	\$0010	(予約)
\$0013	BGCCR校正值@ <i>T<sub>HOT</sub></i> °C (注7)	\$0012	(予約)
\$0015	V-ADC列セル1校正值上位 (注8)	\$0014	V-ADC列セル1校正值下位 (注8)
\$0017	V-ADC列ADC0校正值上位 (注8)	\$0016	V-ADC列ADC0校正值下位 (注8)
\$0019	VPTAT校正值上位 (注15)	\$0018	VPTAT校正值下位 (注15)
\$001B	V-ADC電池セル1利得校正值上位 (注9)	\$001A	V-ADC電池セル1利得校正值下位 (注9)
\$001D	V-ADC電池セル2利得校正值上位 (注10)	\$001C	V-ADC電池セル2利得校正值下位 (注10)
\$001F	V-ADC電池セル2変位(オフセット)値 (注11)	\$001E	V-ADC電池セル1変位(オフセット)値 (注11)
\$0021	V-ADC ADC0利得校正值上位 (注12)	\$0020	V-ADC ADC0利得校正值下位 (注12)
\$0023	V-ADC ADC1利得校正值上位 (注13)	\$0022	V-ADC ADC1利得校正值下位 (注13)
\$0025	V-ADC ADC1変位(オフセット)値 (注14)	\$0024	V-ADC ADC0変位(オフセット)値 (注14)
\$0027~\$002F	(予約)	\$0026~\$002E	(予約)
\$0031	(予約)	\$0030	<i>T<sub>HOT</sub></i> (注16)

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: 高速RC発振器校正時に或る1区分から次への遷移を円滑にするのに使われるFOSCCAL設定。

注3: 発振器採取インターフェースを使ったμsでの8前置分周低速RC周期(@*T<sub>HOT</sub>*°C)。

注4: 特性付けされた低速RC発振器周波数温度偏移予測値。

注5: kHzでの低速RC発振器周波数(@*T<sub>HOT</sub>*°C)。

注6: kHzでの超低電力RC発振器周波数(@*T<sub>HOT</sub>*°C)。

注7: VREF校正(BGCRR=\$0F)の第1段階に於いてBGCCRで得られる校正值(@*T<sub>HOT</sub>*°C)。

注8: VREF校正の第2段階で使う校正語値。

注9: V-ADCのセル1入力での利得誤差に対する補償に使う校正語値。

注10: V-ADCのセル2入力での利得誤差に対する補償に使う校正語値。

注11~16: 次頁へ

- 注11:** V-ADCセル1,2入力での変位(オフセット)に対する補償に使う校正バイト値。  
**注12:** V-ADCのADC0入力での利得誤差に対する補償に使う校正語値。  
**注13:** V-ADCのADC1入力での利得誤差に対する補償に使う校正語値。  
**注14:** V-ADCのADC0,1入力での変位(オフセット)に対する補償に使う校正バイト値。  
**注15:** VTEMP変換からケルビンでの絶対温度計算に使う校正值。  
**注16:** 工場校正で使われた°Cでの $T_{HOT}$ 温度。  
**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。

### 26.2.6. 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26-2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26-2. SPM命令によるフラッシュメモリのプログラミング時間 ( $f_{osc}=8\text{MHz}$ )

項目	最小	最大
SPM命令によるフラッシュ書き込み (ページ消去、ページ書き込み、施錠ビット書き込み)	3.7ms	4.5ms

**注:** 最小と最大の時間は(項目の)個別操作毎に対してです。

表26-3. 図26-1.で使った各変数説明とZポイントの配置 (ATmega8HVA/ATmega16HVA)

変数名	対応値		意味
	PC	Zポイント(注)	
PCMSB	PC11/12		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12/13ビット、PC11/12~0)
PAGEMSB	PC5/5		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64/64語(ワード)には6/6ビット PC5/5~0が必要)
ZPCMSB		Z12/13	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		Z6/6	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	PC11/12~6/6	Z12/13~7/7	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	PC5/5~0	Z6/6~1	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。)

(訳注) 原書の表26-3.(ATmega8HVA)と表26-4.(ATmega16HVA)は表26-3.として纏めました。

## 26.3. 自己プログラミング用レジスタ

### 26.3.1. SPMCSR – SPM命令制御/状態レジスタ (Store Program Memory Control and Status Register)

このレジスタはプログラムメモリ操作を制御するために必要とする制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$37 (\$57)	–	–	SIGRD	CTPB	RFLB	PGWRT	PGERS	SPMEN	SPMCSR
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います。

将来のデバイスとの互換性のため、SPMCSRが書かれる時にこれらのビットは0を書かれなければなりません。

- **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95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 – CTPB : ページ一時緩衝部消去 (Clear Temporary Page Buffer)**

ページ一時緩衝部を満たしている間にCTPBビットが1を書かれると、ページ一時緩衝部は消去され、データは失われます。

- **ビット3 – RFLB : ヒューズ/施錠ビット読み込み (Read Fuse and Lock Bits)**

SPMCSRでRFLBとSPMENが設定(1)された後の3クロック周期内のLPM命令は(ZポイントのZ0に依存して)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94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ポイントの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動解除(0)します。ページ全体の書き込み動作中、CPUは停止されます。

- **ビット1 – PGERS : ページ消去 (Page Erase)**

このビットが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ポイントの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動解除(0)します。ページ全体の消去中、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SPM命令を許可します。このビットがCTPB,RFLB,PGWRT,PGERSのどれかと共に1を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。SPMENだけが書かれると、続くSPM命令はZポイント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、自動解除(0)します。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. メモリプログラミング

### 27.1. プログラムメモリとデータメモリ用施錠ビット

ATmega8HVA/16HVAは非プログラム(1)のままか、表27-2.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2つの施錠ビットを提供します。この施錠ビットはチップ消去指令でのみ1に消去できます。

表27-1. 施錠ビットバイトの内容

名称	ビット番号	意味	既定値 (注)
-	7		1 (非プログラム)
-	6		1 (非プログラム)
-	5		1 (非プログラム)
-	4		1 (非プログラム)
-	3		1 (非プログラム)
-	2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表27-2. 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	低電圧及び高電圧直列またはデバッグWIRE経由プログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズビットが固定されます。(注1)

注: 0はプログラム、1は非プログラムを意味します。

注1: 施錠ビットを書く前にヒューズビットを書いてください。デバッグWIREは全面的に禁止されます。

## 27.2. ヒューズ ビット

ATmega8HVA/16HVAには2つのヒューズ バイトがあります。表27-3.~4.は全てのヒューズの概略機能とヒューズ バイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

### 27.2.1. ヒューズ 上位バイト

表27-3. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
-	7~2	-	1 (非プログラム)
OSCSEL1	1	発振器選択	0 (プログラム)
OSCSEL0	0		1 (非プログラム)

**注:** 既定のOSCSEL1,0設定は変更されるべきではありません。OSCSEL1,0='00'は検査目的に予約されています。他の値は将来の使用の対して予約されています。

### 27.2.2. ヒューズ 下位バイト

表27-4. ヒューズ 下位バイト一覧

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
SPIEN (注2)	5	低電圧直列プログラミング許可。	0 (プログラム) 低電圧直列プログラミング許可
DWEN	4	デバッグWIRE機能許可。	1 (非プログラム) デバッグWIRE不許可
SELFPRGEN	3	自己プログラミング機能許可。	1 (非プログラム) 自己プログラミング不許可
SUT2	2	起動時間選択。 (注1)	1 (非プログラム)
SUT1	1		1 (非プログラム)
SUT0	0		1 (非プログラム)

**注1:** 起動時間についての詳細に関しては18頁の表9-1.をご覧ください。

**注2:** SPIENヒューズは低電圧直列プログラミングでアクセスできません。

**注3:** 詳細については33頁の「WDTCSR - ウォッチドッグ タイム制御レジスタ」をご覧ください。

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。

### 27.2.3. 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せん。

### 27.3. 識票バイト

全てのAtmelマイクロコントローラはデバイスを確認する3バイトの識票符号を持ちます。この符号は直列と高電圧プログラミング動作の両方で、またデバイスが施錠されていても読めます。この3バイトは分離された空間に存在します。ATmega8HVA/16HVA用の識票バイトは表27-5.で与えられます。

表27-5. デバイスの識別番号(ID)

部品番号	識票バイトアドレス		
	\$0000	\$0001	\$0002
ATmega8HVA	\$1E	\$93	\$10
ATmega16HVA	\$1E	\$94	\$0C

### 27.4. 校正バイト

ATmega8HVA/16HVAは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。これらのバイトは識票アドレス空間に属しています。詳細については95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。

### 27.5. ページ容量

表27-6. フラッシュメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
ATmega8HVA	4K語 (8Kバイト)	64語	PC5~0	64	PC11~6	11
ATmega16HVA	8K語 (16Kバイト)	64語	PC5~0	128	PC12~6	12

表27-7. EEPROMメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
ATmega8HVA/16HVA	256バイト	4バイト	EEA1~0	64	EEA7~2	7

## 27.6. 低電圧直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され得るのに先立ってプログラミング許可命令が初めに実行されることを必要とします。**注意**、表27-8.でSPIプログラミング用のピン配置が一覧されます。全てのデバイスが内部SPIインターフェースに対する専用SPIピンを使うとは限りません。

表27-8. 低電圧直列プログラミング用ピン配置

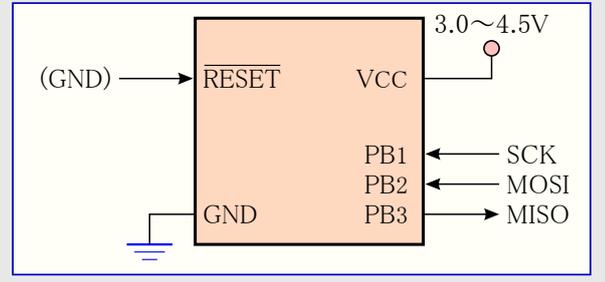
信号名	ピン名	入出力	機能
MISO	PB3	出力	直列データ出力
MOSI	PB2	入力	直列データ入力
SCK	PB1	入力	直列クロック

EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(低電圧直列プログラミングのみ)、チップ消去命令を先に実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全メモリ位置の内容を\$FFにします。

OSCSSELヒューズに依存して有効なクロックが存在しなければなりません。直列クロック(SCK)入力のLowとHighの区間の最小値は次のように定義されます。

$f_{CK} < 12\text{MHz}$ : Low区間	> 2 CPUクロック周期	$f_{CK} < 12\text{MHz}$ : High区間	> 2 CPUクロック周期
$f_{CK} \geq 12\text{MHz}$ : Low区間	> 3 CPUクロック周期	$f_{CK} \geq 12\text{MHz}$ : High区間	> 3 CPUクロック周期

図27-1. 低電圧直列プログラミング構成図



### 27.6.1. 低電圧直列プログラミング手順

ATmega8HVA/16HVAに直列データを書く時にデータはSCKの上昇端で行われ、ATmega8HVA/16HVAから読む時にデータはSCKの下降端で行われます。タイミングの詳細については116頁の「低電圧直列プログラミング特性」をご覧ください。

低電圧直列プログラミング動作でのATmega8HVA/16HVAのプログラミングと照合は次の手順が推奨されます(表27-10.の4バイト命令形式をご覧ください)。

#### 1. 電源投入手順:

RESETとSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加します。いくつかのシステムで電源投入中、SCKがLowに保持されるのを書き込み器が保証できません。この場合、SCKがLow(0)に設定されてしまった後、RESETは最低2 CPUクロック周期幅の正パルスを与えられなければなりません。

#### 2. 最低20ms待ち、MOSIピンにプログラミング許可命令を送ることによって直列プログラミングを許可してください。

#### 3. 通信が同期を外している、直列プログラミング命令は動作しません。同期していると、プログラミング許可命令の第3バイト送付時に第2バイト(\$53)が戻ります。この戻りが正しいかどうかによらず、命令の4バイト全てが送信されなければなりません。\$53が戻らない場合、RESETに正パルスを与え、新規プログラミング許可命令を行ってください。

#### 4. フラッシュメモリは1ページ単位で書かれます。ページ容量は100頁の表27-6.で得られます。このメモリページはページ設定命令と共にアドレスの下位5+1ビットとデータを供給することによって1バイトずつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。プログラムメモリのページはアドレスの上位6/7ビットを含むページ書き込み命令の設定によって(フラッシュメモリに)格納されます。ポーリング(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低tWD\_FLASH(表27-9.参照)待たなければなりません。フラッシュ書き込み操作完了前の(ポーリング以外の)直列プログラミングインターフェースでのアクセスは不正な書き込み結果になり得ます。

#### 5. EEPROMはページ単位かバイト単位のどちらかでプログラミングできます。

**バイト単位:** EEPROMは適切なEEPROM書き込み命令と共にアドレスとデータを供給することによって1バイト単位で書かれます。EEPROMのメモリ位置は新規データが書かれるのに先立って始めて自動的に消去されます。ポーリング(BSY/RDY)が使われない場合、次のバイトを行う前に最低tWD\_EEPROM(表27-9.参照)待たなければなりません。

**ページ単位:** EEPROMの1ページはEEPROMページ設定命令と共にアドレスの下位2ビットとデータを供給することによって1バイトずつ設定されます。EEPROMページはアドレスの上位6ビットを含むEEPROMページ書き込み命令によって(EEPROMに)格納されます。EEPROMページアクセス使用時、EEPROMページ設定命令で設定したバイト位置だけが変更されます。残りの位置は無変化で留まります。ポーリング(BSY/RDY)が使われない場合、次のページ(表27-7.参照)を行う前に最低tWD\_EEPROM(表27-9.参照)待たなければなりません。チップ消去されたデバイスでの\$FFデータは書かれる必要がありません。

#### 6. どのメモリ位置も選んだアドレスの内容を直列出力MISOに読み戻す読み出し命令を使うことによって照合できます。

#### 7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。

#### 8. 電源OFF手順 (必要とされるならば)

- RESETをHigh(1)に設定します。
- VCC電源をOFFにします。

表27-9. ヒューズ、フラッシュ、EEPROM次位置書き込み前の待機時間

シンボル	最低待機時間	備考
tWD_FUSE	4.5ms	ヒューズ書き込み
tWD_FLASH	4.5ms	フラッシュメモリ書き込み
tWD_EEPROM	4.0ms	EEPROM書き込み
tWD_ERASE	4.0ms	チップ消去

### 27.6.2. 低電圧直列プログラミング命令一式

表27-10.と次頁の図27-2.は命令一式を記述します。

表27-10. 低電圧直列プログラミング命令一式

命令	命令形式				備考
	第1バイト	第2バイト	第3バイト	第4バイト	
プログラミング許可	\$AC	\$53	\$00	\$00	
チップ消去	\$AC	\$80	\$00	\$00	
多忙/準備可検査	\$F0	\$00	\$00	状態値	最下位ビットが多忙フラグ。
<b>設定系命令</b>					
拡張アドレス設定 (注1)	\$4D	\$00	拡張アドレス	\$00	
フラッシュ ページ内上位バイト設定	\$48	アドレス上位	アドレス下位	上位バイト	
フラッシュ ページ内下位バイト設定	\$40	アドレス上位	アドレス下位	下位バイト	
EEPROMページ内バイト設定	\$C1	\$00	ページ内位置	バイト	注: ページ内指示以外のビットは0。
<b>読み出し命令</b>					
フラッシュメモリ上位バイト読み出し	\$28	アドレス上位	アドレス下位	上位バイト	
フラッシュメモリ下位バイト読み出し	\$20	アドレス上位	アドレス下位	下位バイト	
EEPROM読み出し	\$A0	アドレス上位	アドレス下位	バイト	
施錠ビット読み出し	\$58	\$00	\$00	施錠ビット値	
識票バイト読み出し	\$30	\$00	アドレス	識票バイト	
ヒューズ下位読み出し	\$50	\$00	\$00	ヒューズ下位	
ヒューズ上位読み出し	\$58	\$08	\$00	ヒューズ上位	
拡張ヒューズ読み出し	\$50	\$08	\$00	拡張ヒューズ	拡張ヒューズは存在しません。
校正バイト読み出し	\$38	\$00	\$00	校正バイト	
<b>書き込み命令 (注2)</b>					
フラッシュ ページ書き込み	\$4C	アドレス上位	アドレス下位	\$00	
EEPROMバイト書き込み	\$C0	アドレス上位	アドレス下位	バイト	
EEPROMページ書き込み	\$C2	アドレス上位	アドレス下位	\$00	注: ページ指示以外のアドレスビットは0。
施錠ビット書き込み	\$AC	\$E0	\$00	施錠ビット値	
ヒューズ下位書き込み	\$AC	\$A0	\$00	ヒューズ下位	
ヒューズ上位書き込み	\$AC	\$A8	\$00	ヒューズ上位	
拡張ヒューズ書き込み	\$AC	\$A4	\$00	拡張ヒューズ	拡張ヒューズは存在しません。

注1: 全命令が全デバイスで利用可能な訳ではありません。

注2: プログラム用メモリにアクセスする命令は語(ワード)アドレスを使います。このアドレスはページ範囲内で乱順にできます。

注: 施錠ビットとヒューズ値はプログラムが0、非プログラムが1です。将来との互換性のため、未使用のヒューズと施錠ビットは非プログラム(1)にすべきです。

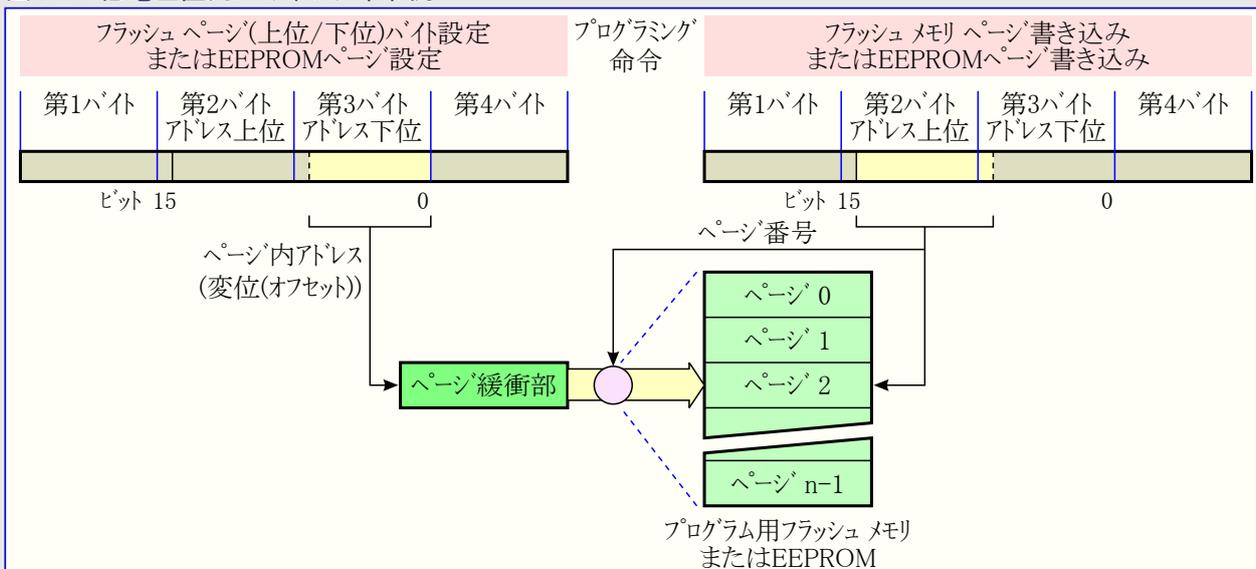
- ヒューズ、施錠ビット、識票バイト、校正バイト、ページ容量については対応項目を参照してください。
- プログラミングと書き込み器に関する応用記述については <http://www.atmel.com/avr> をご覧ください。
- 第4バイトの赤背景はホスト読み込み(デバイス出力)を示します。

多忙/準備可検査バイト データ出力のLSBが1なら、プログラミング操作が未だ保留(動作中)です。次の命令が実行される前に本ビットが0に戻るまで待ってください。

同じページ内で、下位バイト データは上位バイト データに先行して格納されなければなりません。

データがページ緩衝部に格納された後にEEPROMページをプログラムしてください。次頁の図27-2.をご覧ください。

図27-2. 低電圧直列プログラミング命令例



(訳補) フラッシュメモリ、EEPROM、ページ緩衝部内のアドレス(位置)指定に使われるビットはメモリ容量とページ構成に依存します。ATmega8HVA/16HVAでのこれらの指定方法は次表で要約されます。

表27-A. アドレス(第2,3バイト)指定法

命令	第2バイト	第3バイト	備考
拡張アドレス設定			該当命令なし
フラッシュページ内バイト設定	0000 0000	00LL LLLL	ATmega8HVA/16HVA : L=PC5~0
EEPROMページ内バイト設定	0000 0000	0000 00LL	ATmega8HVA/16HVA : L=EEA1~0
フラッシュメモリ読み出し	0000 HHHH 000H HHHH	LLLL LLLL	ATmega8HVA : H=PC11~8, L=PC7~0 ATmega16HVA : H=PC12~8, L=PC7~0
EEPROM読み出し	0000 0000	LLLL LLLL	ATmega8HVA/16HVA : L=EEA7~0
フラッシュページ書き込み	0000 HHHH 000H HHHH	LL00 0000	ATmega8HVA : H=PC11~8, L=PC7~6 ATmega16HVA : H=PC12~8, L=PC7~6
EEPROMバイト書き込み	0000 0000	LLLL LLLL	ATmega8HVA/16HVA : L=EEA7~0
EEPROMページ書き込み	0000 0000	LLLL LL00	ATmega8HVA/16HVA : L=EEA7~2

## 27.7. 高電圧直列プログラミング

本項はATmega8HVA/16HVAでのプログラム用フラッシュメモリ、データ用EEPROM、**施錠ビット**、**ヒューズビット**のプログラミングと照合の方法を記述します。

表27-11. 高電圧直列プログラミング用ピン配置

信号名	ピン名	入出力	機能
SDO	PB1	出力	直列データ出力
SDI	PB2	入力	直列データ入力
SII	PB3	入力	直列命令入力
SCI	PC0	入力	直列クロック入力 (最小=2/fck周期)

図27-3. 高電圧直列プログラミング構成図

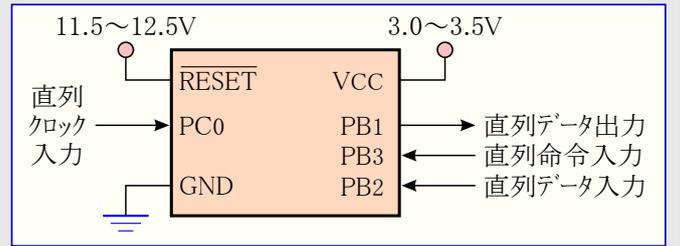


表27-12. プログラミング動作移行に使うピン値

ピン(信号)名	シンボル	値
PB0	Prog_enable[0]	0
PB1	Prog_enable[1]	0
PB2	Prog_enable[2]	0
PB3	Prog_enable[3]	0

## 27.8. 高電圧直列プログラミング手順

高電圧直列プログラミング動作でのATmega8HVA/16HVAのプログラミングと照合は、次の手順が推奨されます(表27-14.の命令形式をご覧ください)。

### 27.8.1. 高電圧直列プログラミング動作への移行

次に示す方法がデバイスを高電圧直列プログラミング動作にします。

- VCCとRESETピンを0V、表27-12.で一覧されるProg\_enableピンを全てLow(0)に設定します。
- VCCとGND間に3.0~3.5Vを印加します。20μs以内にVCCが最低1.8Vに達することを保証してください。
- RESETに11.5~12.5Vを印加し、20~60μs待ちます。
- Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加された後、最低tHVRST間、Prog\_enableピンを無変化に保ちます。
- Prog\_enable[1]/SDOピンでの駆動衝突を避けるため、Prog\_enable[1]/SDOピンを開放します。
- SDI/SIIで如何なる直列命令を与えるのにも先立って少なくとも300μs間待ちます。

VCCの上昇時間が上で示した必要条件を完全に満たせない場合、次の代替手順が使えます。

- VCCとRESETピンを0V、表27-12.で一覧されるProg\_enableピンを全てLow(0)に設定します。
- VCCとGND間に3.0~3.5Vを印加します。
- VCCを監視し、0.9~1.1Vに達したら直ぐ、RESETにVHVRST~12.5Vを印加します。
- Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加された後、最低tHVRST間、Prog\_enableピンを無変化に保ちます。
- Prog\_enable[1]/SDOピンでの駆動衝突を避けるため、Prog\_enable[1]/SDOピンを開放します。
- SDI/SIIでどの直列命令を与えるのにも先立って、VCCが実際に3.0~3.5Vに達するまで待ちます。

表27-13. 高電圧リセット特性

供給電圧 (VCC)	RESETピン高電圧閾値電圧 (VHVRST)	プログラミング許可識別ラッチに対する最小高電圧時間 (tHVRST)
3.0V	11.5V	10μs
3.5V	11.5V	10μs

## 27.8.2. 効率的なプログラミングへの考慮

設定した命令とアドレスはプログラミング中、デバイス内で維持されます。効率的なプログラミングを行うために次が考慮されるべきです。

- 複数のメモリ領域を読み書きする時に命令設定は一度だけ必要です。
- チップ消去後のフラッシュメモリと(EESAVEヒューズがプログラム(0)されている場合を除き)EEPROM全体の内容は\$FFなので、値が\$FFのデータ書き込みを飛ばします。
- アドレス上位バイトはフラッシュメモリで新規256語(ワード)枠、EEPROMで新規256バイト枠の読み書き前に一度だけ必要です。この考慮は識票バイト読み出しにも適用されます。

## 27.8.3. チップ消去

チップ消去はフラッシュメモリ、EEPROM(注1)、施錠ビットを消去します。施錠ビットはプログラムメモリが完全に消去されてしまうまでリセット(消去)されません。ヒューズビットは変更されません。チップ消去はフラッシュメモリやEEPROMが再書き込みされる前に実行されなければなりません。

**注1:** EESAVEヒューズがプログラム(0)されていると、EEPROMはチップ消去中、保護されます。

1. チップ消去命令を設定します(表27-14をご覧ください)。
2. チップ消去終了に対してSDOがHighになるまで待ちます。
3. 無操作命令を設定します。

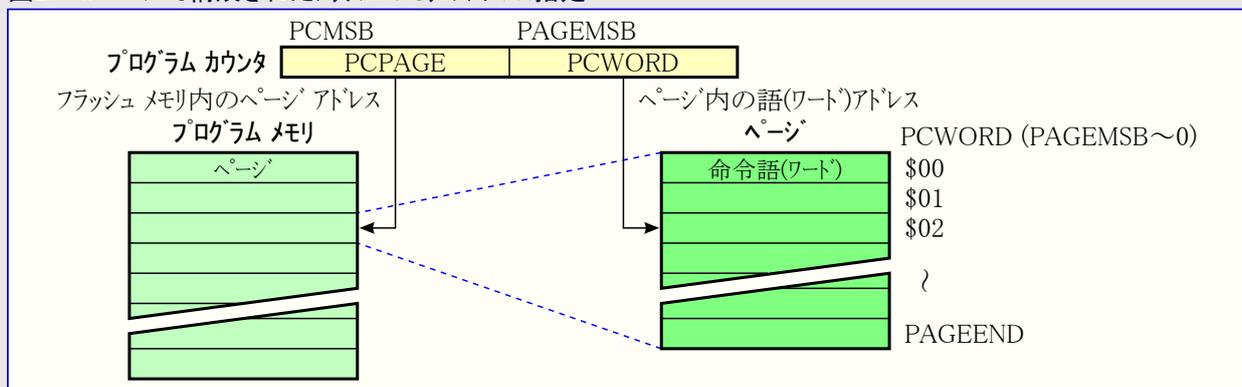
## 27.8.4. フラッシュメモリ書き込み

フラッシュメモリはページで構成されます(100頁の表27-6参照)。フラッシュメモリを書く時にプログラムデータはページ緩衝部にラッチされます。これは同時に書かれることをプログラムデータの1ページに許します。次の手順は完全なフラッシュメモリの書き込み方法を記述します。

1. フラッシュ書き込み移行命令を設定します(表27-14をご覧ください)。
2. フラッシュメモリページ緩衝部を設定します。
3. フラッシュメモリ上位アドレス設定とフラッシュページ書き込み命令を設定します。第3命令バイト後、ページ書き込み終了に対してSDOがhighになるまで待ちます。
4. フラッシュメモリ全体または全データが書かれてしまうまで②～③を繰り返します。
5. 無操作命令の設定によってフラッシュページ書き込みを終えます。

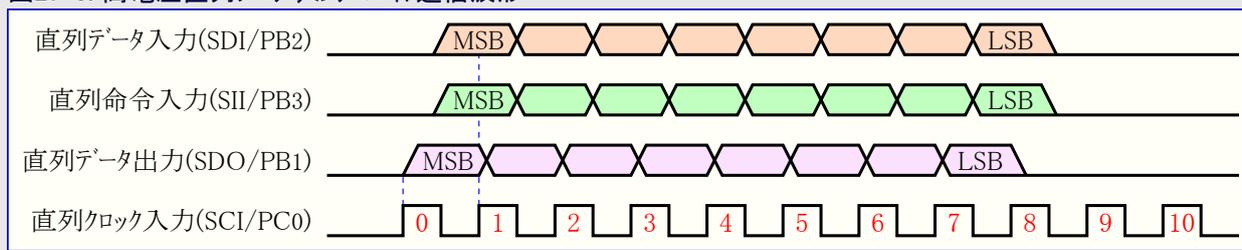
ATmega8HVA/16HVAとの直列データ読み書き時にデータは直列クロックの上昇端でクロック駆動(取得/出力変更)されます。詳細については図27-5と116頁の「高電圧直列プログラミング特性」をご覧ください。

図27-4. ページで構成されたフラッシュメモリのアドレス指定



**注:** PCPAGEとPCWORDは100頁の表27-6で一覧されます。

図27-5. 高電圧直列プログラミングバイト通信波形



### 27.8.5. EEPROM書き込み

EEPROMはページで構成されます(100頁の表27-7参照)。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。(表27-14参照)

1. EEPROM書き込み移行命令を設定します。
2. EEPROMページ緩衝部を設定します。
3. EEPROMページ書き込み命令を設定します。第2命令バイト後、ページ書き込み終了に対してSDOがhighになるまで待ちます。
4. EEPROM全体または全データが書かれてしまうまで②～③を繰り返します。
5. 無操作命令の設定によりEEPROMページ書き込みを終えます。

### 27.8.6. フラッシュメモリ読み出し

フラッシュメモリの読み出し方法は次のとおりです。(表27-14参照)

1. フラッシュ読み出し移行命令を設定します。
2. フラッシュの上下バイトを読み出します。選んだアドレスの内容はSDO直列出力で利用可能です。

### 27.8.7. EEPROM読み出し

EEPROMの読み出し方法は次のとおりです。(表27-14参照)

1. EEPROM読み出し移行命令を設定します。
2. EEPROMのバイトを読み出します。選んだアドレスの内容はSDO直列出力で利用可能です。

### 27.8.8. ヒューズと施錠ビットの読み出し/書き込み

ヒューズ上位/下位と施錠ビットの読み出し/書き込み方法は表27-14.で示されます。

### 27.8.9. 識票バイトと校正バイトの読み出し

識票バイトと校正バイトの読み出し方法は表27-14.で示されます。

### 27.8.10. 電源OFF手順

デバイスの電源断、またはRESETピンを0Vに引き込むことによってプログラミング動作を抜けます。

表27-14. 高電圧直列プログラミング命令一式

命令	命令形式						備考
	第1バイト	第2バイト	第3バイト	第4バイト	第5バイト	第6バイト	
チップ消去	SDI	0 1000 0000 00	0 0000 0000 00	0 0000 0000 00			第3バイト後SDO=Highまで待機。
	SII	0 0100 1100 00	0 0110 0100 00	0 0110 1100 00			
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx			
フラッシュメモリ書き込み移行	SDI	0 0001 0000 00					フラッシュメモリ書き込み処理移行。
	SII	0 0100 1100 00					
	SDO	x xxxx xxxx xx					
フラッシュページ緩衝部設定	SDI	0 AAAA AAAA 00	0 LLLL LLLL 00	0 HHHH HHHH 00	0 0000 0000 00	0 0000 0000 00	必要数分第1～5バイト繰り返し。 A=下位アドレス(注1) H=上位データ L=下位データ
	SII	0 0000 1100 00	0 0010 1100 00	0 0011 1100 00	0 0111 1101 00	0 0111 1100 00	
	SDO	x xxxx xxxx xx	x xxxx xxxx xx				
フラッシュメモリ上位アドレス設定ページ書き込み	SDI	0 0000 UUUU 00	0 0000 0000 00	0 0000 0000 00	第3バイト後SDO=Highまで待機。		必要数分第1～5バイト繰り返し。 A=下位アドレス(注1) H=上位データ L=下位データ
	SII	0 0001 1100 00	0 0110 0100 00	0 0110 1100 00	第3バイト後SDO=Highまで待機。		
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx	第3バイト後SDO=Highまで待機。		
フラッシュメモリ読み出し移行	SDI	0 0000 0010 00					フラッシュメモリ読み出し処理移行。
	SII	0 0100 1100 00					
	SDO	x xxxx xxxx xx					
フラッシュメモリ上下バイト読み出し	SDI	0 AAAA AAAA 00	0 0000 UUUU 00	0 0000 0000 00	0 0000 0000 00	0 0000 0000 00	L LLLL LLLx xx x xxxx xxxx xx H HHHH HHHx xx
	SII	0 0000 1100 00	0 0001 1100 00	0 0110 1000 00	0 0110 1100 00	0 0111 1000 00	
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx	L LLLL LLLx xx	x xxxx xxxx xx	
EEPROM書き込み移行	SDI	0 0001 0001 00		EEPROM書き込み処理移行	第1,3～6バイトを繰り返し。新規256バイトページ時は第2バイトも繰り返し。 A=下位アドレス(注1), H=上位データ, L=下位データ		
	SII	0 0100 1100 00					
	SDO	x xxxx xxxx xx					
EEPROMページ緩衝部設定	SDI	0 AAAA AAAA 00	0 LLLL LLLL 00	0 0000 0000 00	0 0000 0000 00		必要数分第1～4バイト繰り返し。(注2) A=下位アドレス, L=データ
	SII	0 0000 1100 00	0 0010 1100 00	0 0110 1101 00	0 0110 1100 00		
	SDO	x xxxx xxxx xx					
EEPROMページ書き込み	SDI	0 0000 0000 00	0 0000 0000 00				第2バイト後SDO=Highまで待機。
	SII	0 0110 0100 00	0 0110 1100 00				
	SDO	x xxxx xxxx xx	x xxxx xxxx xx				
EEPROMバイト書き込み	SDI	0 AAAA AAAA 00	0 LLLL LLLL 00	0 0000 0000 00	0 0000 0000 00	0 0000 0000 00	
	SII	0 0000 1100 00	0 0010 1100 00	0 0110 1101 00	0 0110 0100 00	0 0110 1100 00	
	SDO	x xxxx xxxx xx	x xxxx xxxx xx				
EEPROM読み出し移行	SDI	0 0000 0011 00		EEPROM読み出し処理移行	第5バイト後SDO=Highまで待機。 必要数分第1～5バイト繰り返し。(注2)(注3) A=下位アドレス, L=データ		
	SII	0 0100 1100 00					
	SDO	x xxxx xxxx xx					
EEPROMバイト読み出し	SDI	0 AAAA AAAA 00	0 0000 0000 00	0 0000 0000 00	0 0000 0000 00		必要数分第1～4バイト繰り返し。
	SII	0 0000 1100 00	0 0001 1100 00	0 0110 1000 00	0 0110 1100 00		
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx	L LLLL LLLx xx		
ヒューズ下位書き込み	SDI	0 0100 0000 00	0 7654 3210 00	0 0000 0000 00	0 0000 0000 00		第4バイト後SDO=Highまで待機。 7～0はビット位置で論理0でプログラム。 (99頁の表27-4参照)
	SII	0 0100 1100 00	0 0010 1100 00	0 0110 0100 00	0 0110 1100 00		
	SDO	x xxxx xxxx xx					
ヒューズ上位書き込み	SDI	0 0100 0000 00	0 1111 1110 00	0 0000 0000 00	0 0000 0000 00		第4バイト後SDO=Highまで待機。 1～0はビット位置で論理0でプログラム。 (99頁の表27-3参照)
	SII	0 0100 1100 00	0 0010 1100 00	0 0111 0100 00	0 0111 1100 00		
	SDO	x xxxx xxxx xx					
施錠ビット書き込み	SDI	0 0010 0000 00	0 0000 0010 00	0 0000 0000 00	0 0000 0000 00		第4バイト後SDO=Highまで待機。 1～0はビット位置で論理0でプログラム。 (98頁の表27-1参照)
	SII	0 0100 1100 00	0 0010 1100 00	0 0110 0100 00	0 0110 1100 00		
	SDO	x xxxx xxxx xx					
ヒューズ下位読み出し	SDI	0 0000 0100 00	0 0000 0000 00	0 0000 0000 00			7～0はビット位置で論理0でプログラム。 (99頁の表27-4参照)
	SII	0 0100 1100 00	0 0110 1000 00	0 0110 1100 00			
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	7 6543 210x xx			
ヒューズ上位読み出し	SDI	0 0000 0100 00	0 0000 0000 00	0 0000 0000 00			1～0はビット位置で論理0でプログラム。 (99頁の表27-3参照)
	SII	0 0100 1100 00	0 0111 1010 00	0 0111 1110 00			
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx x10x xx			
施錠ビット読み出し	SDI	0 0000 0100 00	0 0000 0000 00	0 0000 0000 00			1～0はビット位置で論理0でプログラム。 (98頁の表27-1参照)
	SII	0 0100 1100 00	0 0111 1000 00	0 0111 1100 00			
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx x10x xx			

次頁へ続く



表27-14 (続き). 高電圧直列プログラミング命令一式

命令	命令形式						備考
	第1バイト	第2バイト	第3バイト	第4バイト	第5バイト	第6バイト	
識票バイト読み出し	SDI	0 0000 1000 00	0 00AA AAAA 00	0 0000 0000 00	0 0000 0000 00	A=アドレス 7~0はビット位置。	
	SII	0 0100 1100 00	0 0000 1100 00	0 0110 1000 00	0 0110 1100 00		
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx	7 6543 210x xx		
校正バイト読み出し	SDI	0 0000 1000 00	0 0000 0000 00	0 0000 0000 00	0 0000 0000 00	7~0はビット位置。	
	SII	0 0100 1100 00	0 0000 1100 00	0 0111 1000 00	0 0111 1100 00		
	SDO	x xxxx xxxx xx	x xxxx xxxx xx	x xxxx xxxx xx	7 6543 210x xx		
無操作	SDI	0 0000 0000 00				アイドル状態に復帰。	
	SII	0 0100 1100 00					
	SDO	x xxxx xxxx xx					

**注1:** 256語(ワード)よりも少ないページ容量に関する上位余剰ビットはページアドレス部です。

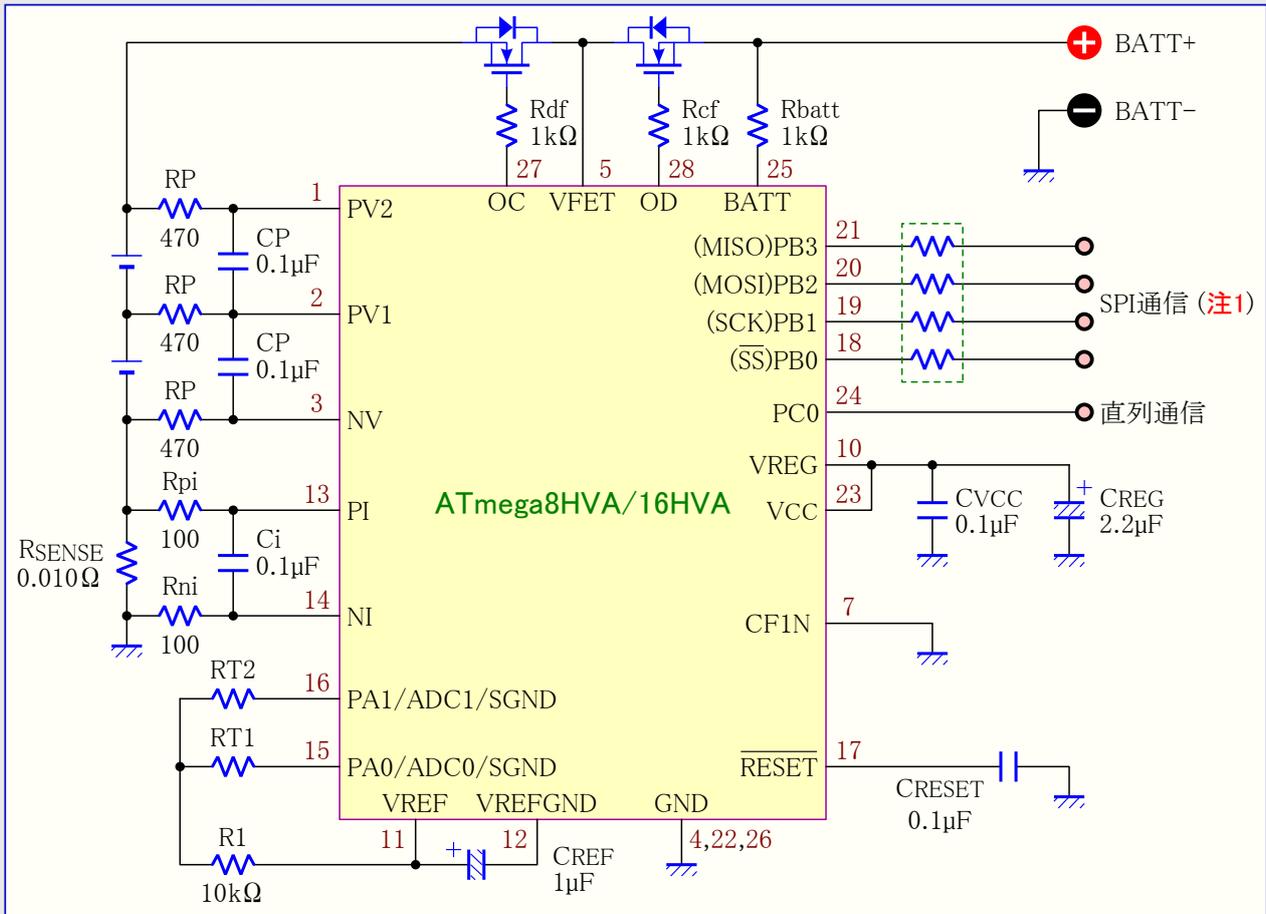
**注2:** 256バイトよりも少ないページ容量に関する上位余剰ビットはページアドレス部です。

**注3:** EEPROMはページ単位で書かれます。しかし、ページ内に設定されたバイトだけが実際にEEPROMへ書かれます。複数バイトが同じページに書かれるべきなら、ページ単位EEPROMアクセスはより効果的です。EEPROMの自動消去はSPI(低電圧)直列プログラミングだけで、高電圧直列プログラミングで利用できないことに注意してください。

**注4:** 上位(ページ)アドレスの有効ビットはATmega8HVA/16HVAで各々異なります(100頁の「ページ容量」と103頁の図27-2.参照)。

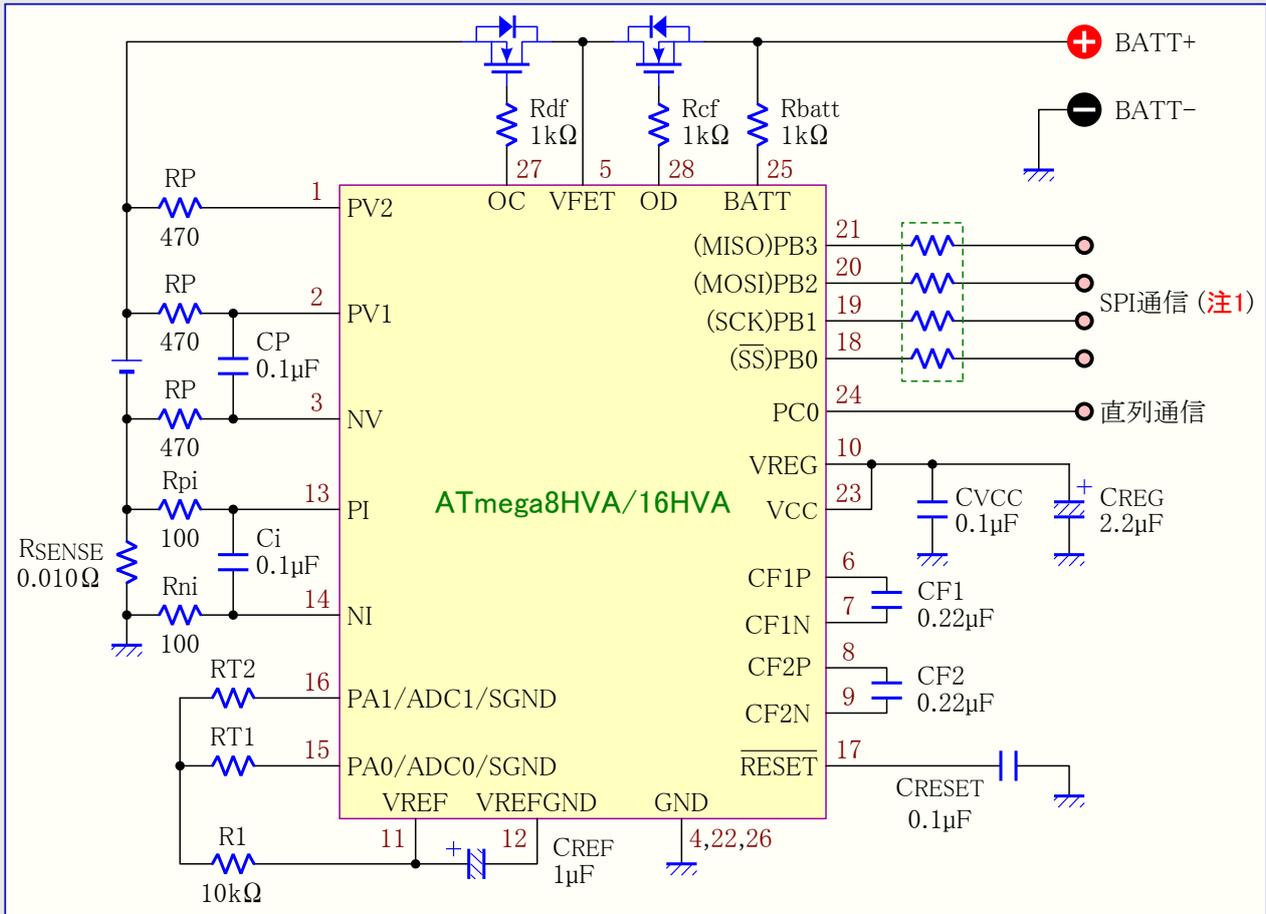
## 28. 動作回路

図28-1. 2セル動作回路構成図



- 注1:** SPI線の直列抵抗は実装書き込みとチップ内デバッグ支援に必要です。直列抵抗の値は応用に依存しています。10kΩの値は正しいプログラミングとデバッグを保証しますが、これはSPIインターフェースの通常動作に影響を及ぼさないことを最終使用者によって決定されなければなりません。
- 注2:** RT2サーミスタ電圧測定時、PA1はSGNDに接続されるべきです。RT1サーミスタ電圧測定時、PA0はSGNDに接続されるべきです。
- 注3:** CF1P,CF2N,CF2PをGNDへ接続することが推奨されます。

図28-2. 1セル動作回路構成図



- 注1:** SPI線の直列抵抗は実装書き込みとチップ内デバッグ支援に必要です。直列抵抗の値は応用に依存しています。10kΩの値は正しいプログラミングとデバッグを保証しますが、これはSPIインターフェースの通常動作に影響を及ぼさないことを最終使用者によって決定されなければなりません。
- 注2:** RT2サーミスタ電圧測定時、PA1はSGNDに接続されるべきです。RT1サーミスタ電圧測定時、PA0はSGNDに接続されるべきです。

表28-1. 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	8	10	12	kΩ
RT1,RT2	NTCサーミスタ	R/25°C	8	10	12	
		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	kΩ
		RSのために最悪利得誤差	0	1	2	
CREF	VREF雑音分離(デカップ)コンデンサ	C	1	2.2	22	μF
CREG	VREG蓄電	C	1.1 (注)	1	22	
CVCC	VCC雑音分離(デカップ)コンデンサ	C		0.1		
CF1,CF2	チャージポンプ	C		0.22		
Rcf,Rdf	-	R		1		kΩ
Rbatt	-	R		1		
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	μF
RP×CP	セル入力低域通過濾波器時定数	τ	6.5	25	100	μs
Rpi,Rni	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	μF
(Rpi-Rni)×Ci	電流監視低域通過濾波器時定数	τ		10	20	μs
RSENSE	クーロンカウンタ電流検出抵抗器	R		10		mΩ

注: これは電圧調整器の安定動作を保証するために必要とする絶対最低容量です。

## 29. 電気的特性

### 29.1. 絶対最大定格 (警告)

動作温度	-20°C ~ +85°C
保存温度	-65°C ~ +150°C
PA0~1,PI,NIB <sup>レ</sup> 許容電圧	-0.5V ~ VREG+0.5V
PB0~3 <sup>レ</sup> 許容電圧	-0.5V ~ VCC+0.5V
PC0 <sup>レ</sup> 許容電圧	-0.5V ~ +6.0V
VFET <sup>レ</sup> 許容電圧	-0.5V ~ +18.0V
OC,OD,BATT,RESET <sup>レ</sup> 許容電圧	-0.5V ~ +13.0V
NV,PV1,PV2 <sup>レ</sup> 許容電圧	-0.5V ~ VFET+1.0V
VREG,VCC最大動作電圧	4.5V
VFET最大動作電圧	9.0V

(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だけのもので、この値または、この仕様書の動作特性で示された値を超える条件で動作することを示すものではありません。長時間の最大定格での使用はデバイスの信頼性を損なう場合があります。

### 29.2. DC特性

表29-1. 電気的特性 (TA=-10°C~70°C,別記指定を除く)

項目		条件		最小	代表	最大	単位		
消費電流	活動動作	4V ≤ VFET ≤ 8.4V PRR全ビット=1	4.0MHz		2.5		mA		
			1.0MHz		800				
	アイドル動作	4V ≤ VFET ≤ 8.4V PRR全ビット=1	4.0MHz		550			μA	
			1.0MHz		270				
	A/D変換雑音低減動作	4V ≤ VFET ≤ 8.4V,VADC許可 PRVADCを除くPRR全ビット=1	4.0MHz		580				μA
			1.0MHz		380				
パワーセーブ動作	WDTのみ許可,DUVR禁止,VFET=4V				25		nA		
	WDT,CC-ADC,OC,OD,電池保護許可,DUVR禁止	VFET=8.4V			110				
		VFET=3V			240				
パワーオフ動作	VFET ≤ 6V				10	1000	nA		
	6V ≤ VFET ≤ 8.4V				2	10	μA		
電圧調整器 (注2)	動作電圧	昇圧/直接結合動作		1.8		9	V		
		直線動作のみ		3.6		9			
	調整出力電圧(VREG) (注3)	VFET=1.8V,Iload=2mA		2.9		3.4			
		VFET=2.0V,Iload=5mA		3.0		3.4			
		VFET=2.4V,Iload=8.5mA		3.1		3.4			
		VFET=3.0V,Iload=10mA		3.1		3.4			
		VFET=3.8V,Iload=10mA		3.1		3.4			
		VFET=5.5V,Iload=10mA		3.1		3.4			
	VFET昇圧/直線切替電圧 (注5)	直線⇒昇圧,Iload=2mA				3.5			
		昇圧⇒直線,Iload=2mA				3.6			
	短絡検出電圧(RSCL)	昇圧/直接結合動作				1.7			
直線動作のみ				3.5					
昇圧動作リップル電圧 (注5)	VFET=3.0V,CREG=2.2μF, ESR=0.1Ω	IOUT=1mA			5		mV		
		IOUT=10mA			30				
基準電圧 (VREF)	基準電圧				1.100		V		
	基準電圧精度		校正後の校正温度にて		±0.1	±0.2	%		
	温度変動 (注3,5)					90	ppm/°C		

次頁へ続く

表29-1 (続き). 電気的特性 (TA=-10°C~70°C,別記指定を除く)

項目		条件	最小	代表	最大	単位	
電圧 A/D変換器	変換時間	clkVADC=1MHz		519		μs	
	有効分解能			12		ビット	
	非縮尺(1.0倍)入力1 LSB電位差			263		μV	
	縮尺(0.2倍)入力1 LSB電位差			1.43		mV	
	積分非直線性誤差			1	3	LSB	
	入力電圧範囲	ADC0,ADC1,VTEMP		0		1	V
		セル1 (注9)		1.5		5	
		セル2,PV1≥1.5V (注9)		0		5	
	変位(オフセット)誤差 (注8)			6		LSB	
	ADC0/1入力利得誤差 (注3,7)	0.1V<VADC<0.9V			±0.5	%	
電池セル入力利得誤差 (注3,7)	VCELL=4V			±0.5			
クーロン カウンタ A/D変換器	基準電圧			±110		mV	
	変換時間と分解能 (注5)	26.9μV分解能		3.9		ms	
		0.84μV分解能		1000			
	積分非直線性誤差 (注5)				4	%	
	変位(オフセット) (注5,6)			2.5	±15		
利得誤差 (注3)	-100mV<VPI-NI<100mV		±0.1	±1	LSB		
温度感知器	絶対温度対VPTAT電圧係数			0.67		mV/K	
	絶対精度 (注4)	活動動作での測定		±2	±5	K	
低速 RC発振器 (注10)	周波数		91	131	171	kHz	
	温度変動 (注5)			1.5			
	予測誤差 (注5)				1		
超低電力RC 発振器(注10)	周波数		89	128	167	kHz	
	温度変動 (注5)			6			

注1: 本データシートに含まれる全てのDC特性は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付けとシミュレーションに基づいています。これらの値は設計目標を表す暫定値で、実際のシリコンの特性付け後に更新されます。

注2: 電圧供給器の能力は0.22μFチャージホンプ コンデンサと2.2μF平滑コンデンサに基づいています。

注3: 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THOT°Cで実行されます。THOTの値は識票列に格納されています。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4: 測定したVPTAT電圧は絶対温度を得るためにVPTAT校正レジスタ内に格納された校正値で縮尺補正されなければなりません。指定値はAtmel工場校正後の目標精度を表します。精度は良く知られた温度でシステム校正測定を行うことによって更に改善することができます。

注5: この値は製造に於いて検査されません。

注6: 極性切り換え(CADPOL)機能を使ったソフトウェアでのシステム変位(オフセット)補償後の値です。

注7: 識票列に格納された変位(オフセット)と利得の校正値を使ったVADC列データの尺度補正後の値です。

注8: 識票列に格納された各チャネルに対する実際の変位は、この変位誤差を取り去るのに使うことができます。

注9: PV1が1.5V以下の時にセル入力を測定する必要がある場合、Atmelはこの範囲で正確な測定をより少なくして容易にするデータを提供できます。

注10: Atmelの工場での測定された実際の値が識票列に格納されています。

### 29.3. 外部割り込み特性

表29-2. 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 <sub>INT</sub>	非同期外部割り込み最小パルス幅		50		ns

## 29.4. 汎用入出力線特性

表29-3. (TA=-10°C~70°C, VCC=3.3V)

シンボル	項目	条件	最小	代表	最大	単位
VIL	Lowレベル入力電圧(RESETを除く)		-0.5		0.3VCC (注2)	V
VIL1	RESETピンLowレベル入力電圧				0.3VCC (注2)	
VIH	Highレベル入力電圧(RESETを除く)		0.6VCC (注3)		VCC+0.5	
VIH1	RESETピンHighレベル入力電圧		0.9VCC (注3)		VCC+0.5	
VOL	Lowレベル出力電圧	IOL=5mA			0.5	
VOH	Highレベル出力電圧	IOH=-2mA	2.3			
IIL	L側入力漏れ電流	ピン=L(絶対値)			1	μA
IIH	H側入力漏れ電流	ピン=H(絶対値)			1	
RRST	リセットプルアップ抵抗		30		60	kΩ
RPU	I/Oピンプルアップ抵抗		20		50	

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。

表29-4. PC0特性 (TA=-10°C~70°C, 特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
VIL	Lowレベル入力電圧		-0.5		0.8 (注1)	V
VIH	Highレベル入力電圧		2.1 (注2)		5.5	
VOL	Lowレベル出力電圧	IOL=350μA	0		0.4	
t <sub>r</sub>	上昇時間 (注3)				300	ns
t <sub>of</sub>	VIHmin⇒VILmax 下降時間 (注3)	Cb<400pF (注4)			250	
t <sub>sp</sub>	入力濾波器スパイク消去 (注3)		0		50	
I <sub>i</sub>	入力電流 (注3)	0.1VBUS<Vi<0.9VBUS	-5		5	
C <sub>i</sub>	入力容量 (注3)				10	pF

注1: 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2: 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3: この値は製造で検査されていません。

注4: Cbは1本のバス線のpFでの容量です。

## 29.5. FET駆動部特性

表29-5. FET駆動部出力特性 (TA=-10°C~70°C, 特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
	VFET DC電圧 (注2)	1セルDUVR動作, VREF=1.100V	1.9	2.0	2.1	V
		2セルDUVR動作, VREF=1.100V	3.8	4.0	4.2	
	VFETリップル電圧 (注2)	1セルDUVR動作		±0.1		
		2セルDUVR動作		±0.1		
	OC, ODクランプ電圧			14.0		
	OC, OD ON電圧	標準動作	VFET+2.5	VFET+4	VFET+6.5	
	OC, OD OFF電圧	標準動作		0.0	0.1	
	OC, OC 0⇒90%上昇時間 (注2,3)			1	2	
	OC, OC 100⇒10%下降時間 (注2,3)			5	10	μs

注1: 本データシートに含まれる全てのDC特性は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付けとシミュレーションに基づいています。これらの値は設計目標を表す暫定値で、実際のシリコンの特性付け後に更新されます。

注2: これらの数値は1つの外部Nch-FETのTPCS8210の使用が前提です。他のFETが使われた場合、数値は何かが変わるかもしれませんがありません。OCとODでの等価容量性負荷はおよそ1200pFです。上昇及び下降の尺度は概ね容量性負荷に比例します。

注3: 製造で検査されません。

## 29.6. 電源ONとリセットの特性

表29-6. リセット特性

シンボル	項目	条件	最小	代表	最大	単位
VPOT	電源ONリセット閾値電圧 (注1)	VFET=8.4V	2.75	3.65	4.1	V
		VFET=4.2V	2.75	3.5	3.95	
tRST	RESETピン最小パルス幅			900		ns
VBOT	低電圧検出(BOD)電圧			2.9		V
VHYST	低電圧検出ヒステリシス電圧			100		mV
VBLOT	電源OFF閾値電圧			2.4		V

注1: 電池パック+端子での電圧はチップが許可された時にVPOTよりも僅かに高くなるでしょう。これは50~110μAでのBATTピンの内部プルダウン電流と、電池パック+端子とBATTピン間に接続されたRBATT抵抗のためです。RBATT=1kΩは0.05~0.11Vの電圧低下を生じます。

## 29.7. SPIタイミング特性

図29-1. SPI タイミング必要条件 (主装置動作)

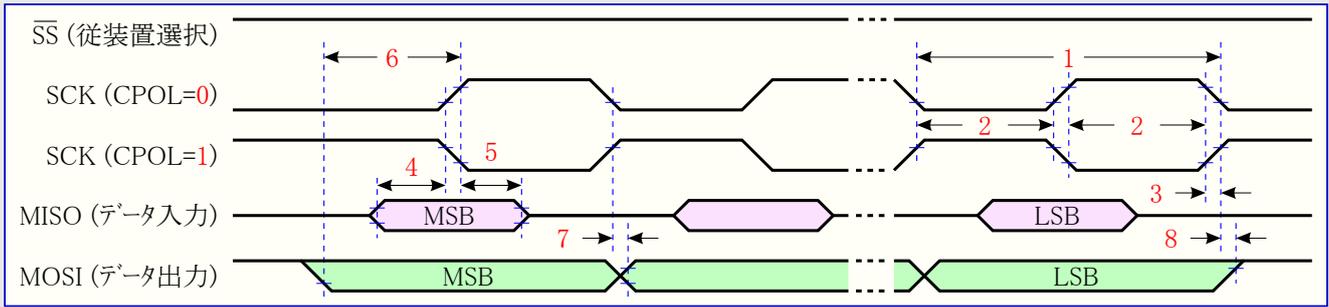


図29-2. SPI タイミング必要条件 (従装置動作)

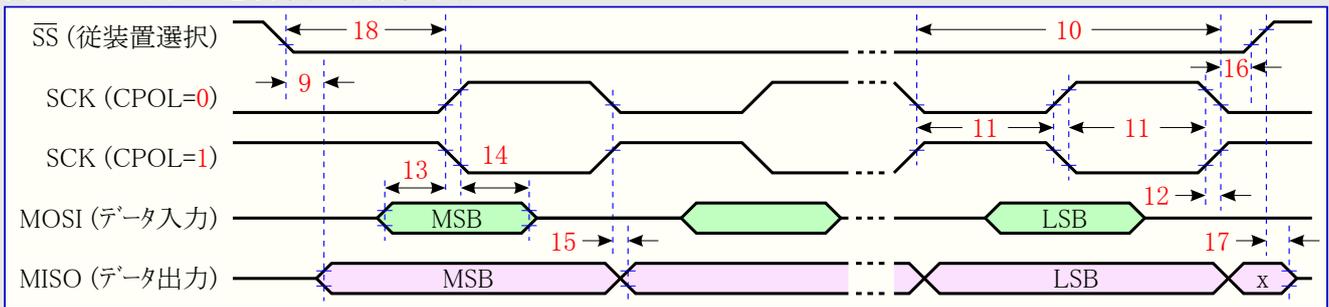


表29-7. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表18-5.参照		ns
2	SCK High/Low期間	主装置		50%デューティ比		
3	SCK上昇/下降時間	主装置		3.6		
4	入力データ 準備時間	主装置		10		
5	入力データ 保持時間	主装置		10		
6	出力からSCK変移時間	主装置		0.5t <sub>SCK</sub>		
7	SCKからの出力遅延時間	主装置		10		
8	SCKからのHigh出力時間	主装置		10		
9	SS ↓からの出力遅延時間	従装置		15		μs
10	SCK周期	従装置	4t <sub>CK</sub> +40			
11	SCK High/Low期間 (注)	従装置	2t <sub>CK</sub> +20			ns
12	SCK上昇/下降時間	従装置		1.6		
13	入力データ 準備時間	従装置	10			
14	入力データ 保持時間	従装置	t <sub>CK</sub>			
15	SCKからの出力遅延時間	従装置		15		
16	SCKからのSS ↑遅延時間	従装置	20			
17	SS ↑からの出力Hi-Z遅延時間	従装置		10		
18	SS ↓からのSCK遅延時間	従装置	20			

注: SPIプログラミングの必要条件に関しては次頁の「低電圧直列プログラミング特性」を参照してください。

## 29.8. フロタラミング特性

### 29.8.1. 低電圧直列フロタラミング特性

図29-3. 低電圧直列フロタラミング タイミング

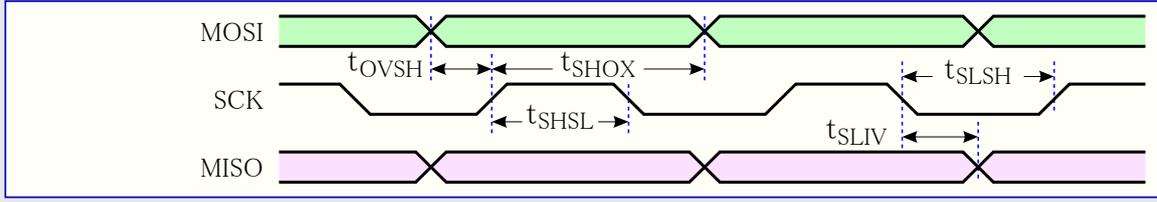


図29-4. 低電圧直列フロタラミング バイト通信波形

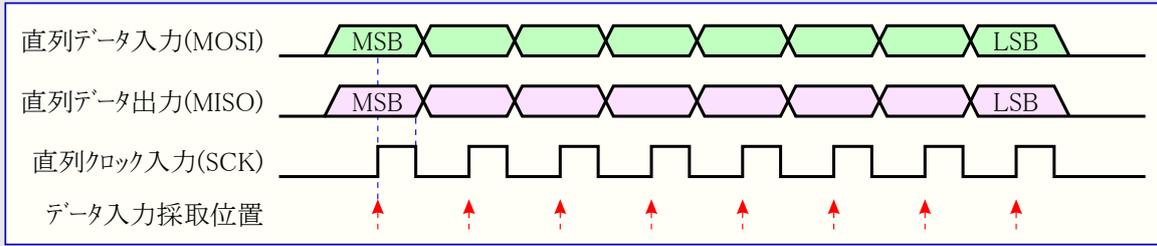


表29-8. 低電圧直列フロタラミング特性 (特記条件を除いて、 $T_A = -10^{\circ}\text{C} \sim 70^{\circ}\text{C}$ ,  $V_{CC} = 3.0 \sim 5.5\text{V}$ )

シンボル	項目	最小	代表	最大	単位
$1/t_{CLCL}$	発振器周波数	0		4	MHz
$t_{CLCL}$	発振器周期	250			ns
$t_{SHSL}$	SCKパルスHレベル幅 (注1)	$2.2t_{CLCL}$			
$t_{SLSH}$	SCKパルスLレベル幅 (注1)	$2.2t_{CLCL}$			
$t_{OVSH}$	SCK $\uparrow$ に対するMOSI準備時間	$t_{CLCL}$			
$t_{SHOX}$	SCK $\uparrow$ に対するMOSI保持時間	$2t_{CLCL}$			
$t_{SLIV}$	SCK $\downarrow$ に対するMISO出力遅延時間		15		

注1:  $f_{CK} < 12\text{MHz}$ 時 $2.2t_{CLCL}$ 、 $f_{CK} \geq 12\text{MHz}$ 時 $3t_{CLCL}$ 。

### 29.8.2. 高電圧直列フロタラミング特性

図29-5. 高電圧直列フロタラミング タイミング

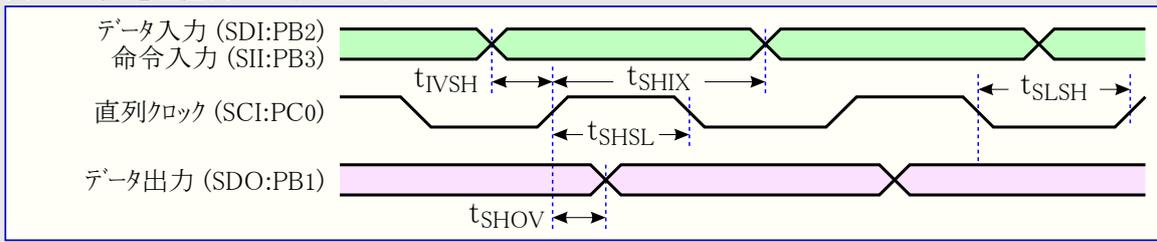


表29-9. 高電圧直列フロタラミング特性 (特記条件を除いて、 $T_A = 25^{\circ}\text{C} \pm 10\%$ ,  $V_{CC} = 3.3\text{V} \pm 10\%$ )

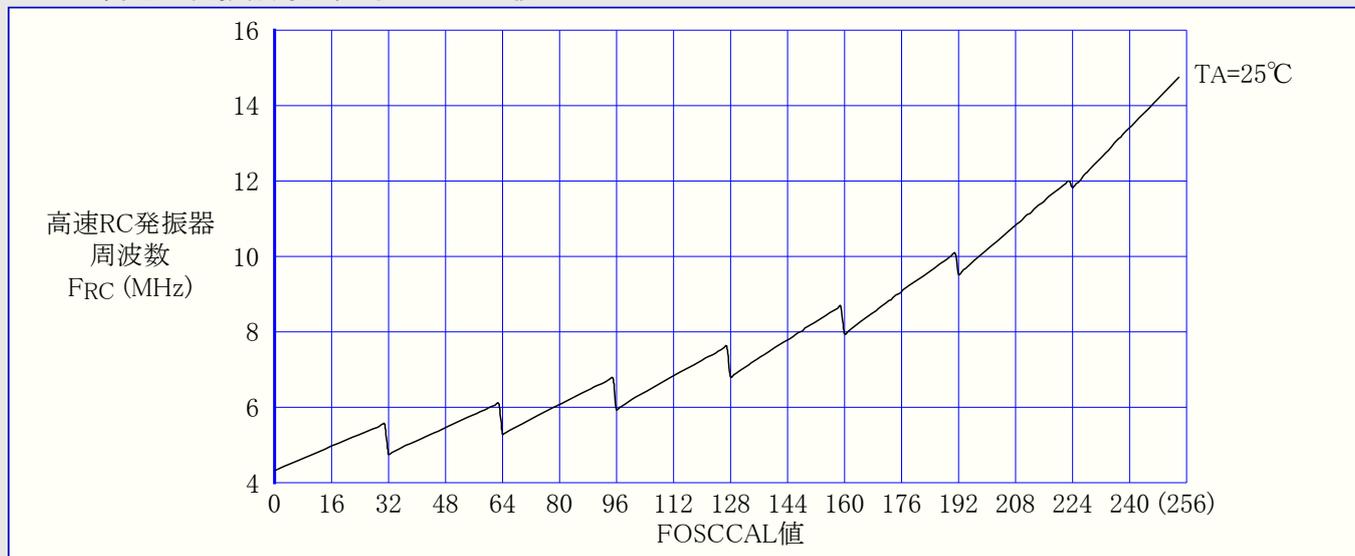
シンボル	項目	最小	代表	最大	単位
$t_{SHSL}$	SCIパルスHレベル幅	$1/f_{ck}$			ns
$t_{SLSH}$	SCIパルスLレベル幅	$1/f_{ck}$			
$t_{IVSH}$	SCI $\uparrow$ に対するSDI,SII準備時間	50			
$t_{SHIX}$	SCI $\uparrow$ に対するSDI,SII保持時間	50			
$t_{SHOV}$	SCI $\uparrow$ に対するSDO出力遅延時間		16		
$t_{WLWH\_PFB}$	ヒューズ書き込み第3バイト後待機時間		2.5		ms

## 30. 代表特性 - 暫定

本データシートに含まれる全ての代表特性は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付けとシミュレーションに基づいています。これらの図は暫定で、実際のシリコンの特性付け後に更新されます

これらの図は製造中に検査されておらず、図解目的用だけに付加されています。

図30-1. 高速RC発振器周波数 対 FOSCCAL値



### 31. 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)	BPPLR	-	-	-	-	-	-	BPPL	BPPL	84		
(\$FD)	BPCR	-	-	-	SCD	DOCD	COCD	DHCD	CHCD	84		
(\$FC)	BPHCTR	-	-	高電流保護反応時間値(HCPT5~0)						86		
(\$FB)	BPOCTR	-	-	過電流保護反応時間値(OCPT5~0)						85		
(\$FA)	BPSCTR	-	短絡保護反応時間値(SCPT6~0)						85			
(\$F9)	BPCHCD	充電高電流検出値(CHCDL7~0)						87				
(\$F8)	BPDHCD	放電高電流検出値(DHCDL7~0)						87				
(\$F7)	BPCOCD	充電過電流検出値(COCDL7~0)						86				
(\$F6)	BPDOCD	放電過電流検出値(DOCDL7~0)						86				
(\$F5)	BPSCD	短絡検出値(SCDL7~0)						86				
(\$F4)	予約											
(\$F3)	BPIFR	-	-	-	SCIF	DOCIF	COCIF	DHCIF	CHCIF	88		
(\$F2)	BPIMSK	-	-	-	SCIE	DOCIE	COCIE	DHCIE	CHCIE	88		
(\$F1)	予約											
(\$F0)	FCSR	-	-	-	-	DUVRD	CPS	DFE	CFE	91		
(\$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)										
(\$E8)	CADICL	CC-ADC 瞬時電流レジスタ下位 (CADIC7~0)									71	
(\$E7)	予約											
(\$E6)	CADRC	CC-ADC 定常電流レジスタ (CADRC7~0)									72	
(\$E5)	CADCSRB	-	CADACIE	CADRCIE	CADICIE	-	CADACIF	CADRCIF	CADICIF	71		
(\$E4)	CADCSRA	CADEN	CADPOL	CADUB	CADAS1	CADAS0	CADS1	CADS10	CADSE	70		
(\$E3)	CADAC3	CC-ADC 累積電流レジスタ 最上位バイト (CADAC31~24)										
(\$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)									72	
(\$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)	BGCRR	基準電圧温度係数校正値 (BGC7~0)									78	
(\$D0)	BGCCR	BGD	-	基準電圧校正値 (BGCC5~0)								78
(\$CF)	予約											
(\$CE)	予約											
(\$CD)	予約											
(\$CC)	予約											
(\$CB)	予約											
(\$CA)	予約											
(\$C9)	予約											
(\$C8)	ROCR	ROCS	-	-	-	-	-	ROCWIF	ROCWIE	81		
(\$C7)	予約											
(\$C6)	予約											
(\$C5)	予約											
(\$C4)	予約											
(\$C3)	予約											
(\$C2)	予約											
(\$C1)	予約											
(\$C0)	予約											

## 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)	予約									
(\$BD)	予約									
(\$BC)	予約									
(\$BB)	予約									
(\$BA)	予約									
(\$B9)	予約									
(\$B8)	予約									
(\$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)	OCR1B									59
(\$88)	OCR1A									59
(\$87)	予約									
(\$86)	予約									
(\$85)	TCNT1H									59
(\$84)	TCNT1L									59
(\$83)	予約									
(\$82)	予約									
(\$81)	TCCR1B	-	-	-	-	-	CS12	CS11	CS10	49
(\$80)	TCCR1A	TCW1	ICEN1	ICNC1	ICES1	ICS1	-	-	WGM10	58

## 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	-	-	-	-	-	-	-	PA0DID	76
(\$7D)	予約									
(\$7C)	VADMUX	-	-	-	-	VADMUX3	VADMUX2	VADMUX1	VADMUX0	75
(\$7B)	予約									
(\$7A)	VADCSR	-	-	-	-	VADEN	VADSC	ADCCIF	ADCCIE	75
(\$79)	VADCH	-	-	-	-	V-ADC データレジスタ 上位 (VADC11~8)				76
(\$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	-	-	-	-	ICIE1	OCIE1B	OCIE1A	TOIE1	60
(\$6E)	TIMSK0	-	-	-	-	ICIE0	OCIE0B	OCIE0A	TOIE0	60
(\$6D)	予約									
(\$6C)	予約									
(\$6B)	予約									
(\$6A)	予約									
(\$69)	EICRA	-	-	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	36
(\$68)	予約									
(\$67)	予約									
(\$66)	FOSCCAL	内蔵高速RC発振器 発振校正値レジスタ								21
(\$65)	予約									
(\$64)	PRR0	-	-	PRVRM	-	PRSPI	PRTIM1	PRTIM0	PRVADC	27
(\$63)	予約									
(\$62)	予約									
(\$61)	CLKPR	CLKPCE	-	-	-	-	-	CLKPS1	CLKPS0	21
(\$60)	WDTCSR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	33

- 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。CBIとSBI命令は指定ビットだけを操作し、故にこのような状態フラグを含むレジスタで使えます。CBIとSBI命令は\$00~\$1Fのレジスタだけで動作します。
  - I/O指定命令INとOUTを使う時はI/Oアドレス\$00~\$3Fが使われなければなりません。LDとST命令を使ってデータ空間としてI/Oレジスタをアドレス指定する時はこれらのアドレスに\$20が加算されなければなりません。ATmega8HVA/16HVA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	-	-	SIGRD	CTPB	RFLB	PGWRT	PGERS	SPMEN	97
\$36 (\$56)	予約									
\$35 (\$55)	MCUCR	-	-	CKOE	PUD	-	-	-	-	21,47
\$34 (\$54)	MCUSR	-	-	-	OCDRF	WDRF	BODRF	EXTRF	PORF	32
\$33 (\$53)	SMCR	-	-	-	-	SM2	SM1	SM0	SE	27
\$32 (\$52)	予約									
\$31 (\$51)	DWDR	デバッグWIRE データレジスタ								92
\$30 (\$50)	予約									
\$2F (\$4F)	予約									
\$2E (\$4E)	SPDR	SPI データレジスタ								66
\$2D (\$4D)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	65
\$2C (\$4C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	65
\$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)	OCR0B	タイマ/カウンタ0 比較Bレジスタ								59
\$28 (\$48)	OCR0A	タイマ/カウンタ0 比較Aレジスタ								59
\$27 (\$47)	TCNT0H	タイマ/カウンタ0 上位バイト								59
\$26 (\$46)	TCNT0L	タイマ/カウンタ0 下位バイト								59
\$25 (\$45)	TCCR0B	-	-	-	-	-	CS02	CS01	CS00	49
\$24 (\$44)	TCCR0A	TCW0	ICEN0	ICNC0	ICES0	ICS0	-	-	WGM00	58
\$23 (\$43)	GTCCR	TSM	-	-	-	-	-	-	PSRSYNC	49
\$22 (\$42)	予約									
\$21 (\$41)	EEAR	EEPROMアドレスレジスタ (EEAR7~0)								13
\$20 (\$40)	EEDR	EEPROMデータレジスタ								13
\$1F (\$3F)	EECR	-	-	EPM1	EPM0	EERIE	EEMPE	EEPE	EERE	13
\$1E (\$3E)	GPIOR0	汎用I/Oレジスタ0								16
\$1D (\$3D)	EIMSK	-	-	-	-	-	INT2	INT1	INT0	37
\$1C (\$3C)	EIFR	-	-	-	-	-	INTF2	INTF1	INTF0	37
\$1B (\$3B)	予約									
\$1A (\$3A)	予約									
\$19 (\$39)	予約									
\$18 (\$38)	予約									
\$17 (\$37)	OSICSR	-	-	-	OSISEL0	-	-	OSIST	OSIEN	22
\$16 (\$36)	TIFR1	-	-	-	-	ICF1	OCF1B	OCF1A	TOV1	60
\$15 (\$35)	TIFR0	-	-	-	-	ICF0	OCF0B	OCF0A	TOV0	60
\$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)	予約									
\$0A (\$2A)	予約									
\$09 (\$29)	予約									
\$08 (\$28)	PORTC	-	-	-	-	-	-	-	PORTC0	40
\$07 (\$27)	予約									
\$06 (\$26)	PINC	-	-	-	-	-	-	-	PINC0	40
\$05 (\$25)	PORTB	-	-	-	-	PORTB3	PORTB2	PORTB1	PORTB0	47
\$04 (\$24)	DDRB	-	-	-	-	DDB3	DDB2	DDB1	DDB0	47
\$03 (\$23)	PINB	-	-	-	-	PINB3	PINB2	PINB1	PINB0	47
\$02 (\$22)	PORTA	-	-	-	-	-	-	PORTA1	PORTA0	47
\$01 (\$21)	DDRA	-	-	-	-	-	-	DDA1	DDA0	47
\$00 (\$20)	PINA	-	-	-	-	-	-	PINA1	PINA0	47

(訳注) 原書本位置の注意は前頁に移動しました。



### 31. 命令要約

ニーモニック	オペラント*	意味	動作	フラグ†	クロック
<b>算術、論理演算命令</b>					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,I	1
NEG	Rd	2の補数	$Rd \leftarrow \$00 - Rd$	I,T,H,S,V,N,Z,C	1
SBR	Rd,K	汎用レジスタの(複数)ビット設定(1)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
CBR	Rd,K	汎用レジスタの(複数)ビット解除(0)	$Rd \leftarrow Rd \text{ AND } (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レジスタの増加(+1)	$Rd \leftarrow Rd + 1$	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	$Rd \leftarrow Rd - 1$	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レジスタのゼロとマイナス検査	$Rd \leftarrow Rd \text{ AND } Rd$	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レジスタの全0設定(=\$00)	$Rd \leftarrow Rd \text{ EOR } Rd$	I,T,H,0,0,0,I,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	$Rd \leftarrow \$FF$	I,T,H,S,V,N,Z,C	1
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
<b>分岐命令</b>					
RJMP	k	相対無条件分岐	$PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	2
IJMP		Zレジスタ間接無条件分岐	$PC \leftarrow Z$	I,T,H,S,V,N,Z,C	2
JMP (注)	k	絶対無条件分岐	$PC \leftarrow k$	I,T,H,S,V,N,Z,C	3
RCALL	k	相対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	3
ICALL		Zレジスタ間接サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow Z$	I,T,H,S,V,N,Z,C	3
CALL (注)	k	絶対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow k$	I,T,H,S,V,N,Z,C	4
RET		サブルーチンからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
RETI		割り込みからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
CPSE	Rd,Rr	汎用レジスタ間比較、一致でスキップ*	$Rd=Rr$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	$Rd - Rr$	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	$Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	$Rd - K$	I,T,H,S,V,N,Z,C	1
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ*	$Rr(b)=0$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
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, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ*	$P(b)=0$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBIS	P,b	I/Oレジスタのビットが設定(1)でスキップ*	$P(b)=1$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
BRBS	s,k	ステータスフラグが設定(1)で分岐	$SREG(s)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRBC	s,k	ステータスフラグが解除(0)で分岐	$SREG(s)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BREQ	k	一致で分岐	$Z=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRNE	k	不一致で分岐	$Z=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCS	k	キャリーフラグが設定(1)で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCC	k	キャリーフラグが解除(0)で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRSH	k	符号なしの $\geq$ で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしの $<$ で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	$N=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	$N=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの $\geq$ で分岐	$(N \text{ EOR } V)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLT	k	符号付きの $<$ で分岐	$(N \text{ EOR } V)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	$H=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	$H=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	$T=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	$T=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	$V=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	$V=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	$I=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	$I=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ  
 b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)  
 注: これらの命令はATmega16HVAでだけ利用できます。

ニーモニック	オペランド	意味	動作	フラグ	クロック
<b>データ移動命令</b>					
MOV	Rd,Rr	汎用レジスタ間の複写	$Rd \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
MOVW	Rd,Rr	汎用レジスタ対間の複写	$Rd+1:Rd \leftarrow Rr+1:Rr$	I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	$Rd \leftarrow K$	I,T,H,S,V,N,Z,C	1
LD	Rd,X	Xレジスタ間接での取得	$Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,X+	事後増加付きXレジスタ間接での取得	$Rd \leftarrow (X), X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-X	事前減少付きXレジスタ間接での取得	$X \leftarrow X - 1, Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y	Yレジスタ間接での取得	$Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
LDD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上 (任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上 (事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
<b>ビット関係命令</b>					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	I,T,H,S,V,1,I,I	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLC		キャリーフラグを解除(0)	$C \leftarrow 0$	I,T,H,S,V,N,Z,0	1
SEN		負フラグを設定(1)	$N \leftarrow 1$	I,T,H,S,V,1,Z,C	1
CLN		負フラグを解除(0)	$N \leftarrow 0$	I,T,H,S,V,0,Z,C	1
SEZ		ゼロフラグを設定(1)	$Z \leftarrow 1$	I,T,H,S,V,N,1,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,0,C	1
SEI		全割り込み許可	$I \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLI		全割り込み禁止	$I \leftarrow 0$	0,T,H,S,V,N,Z,C	1
SES		符号フラグを設定(1)	$S \leftarrow 1$	I,T,H,1,V,N,Z,C	1
CLS		符号フラグを解除(0)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	$V \leftarrow 1$	I,T,H,S,1,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	$V \leftarrow 0$	I,T,H,S,0,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,1,H,S,V,N,Z,C	1
CLT		一時フラグを解除(0)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,1,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
<b>MCU制御命令</b>					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグWIRE機能専用	I,T,H,S,V,N,Z,C	N/A

### 33. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号 (注1)	外囲器	動作範囲
ATmega8HVA	1~4	1.8~9.0V	ATmega8HVA-4CKU	36CK1	-20°C~85°C
			ATmega8HVA-4TU	28T	
ATmega16HVA			ATmega16HVA-4CKU	36CK1	
			ATmega16HVA-4TU	28T	

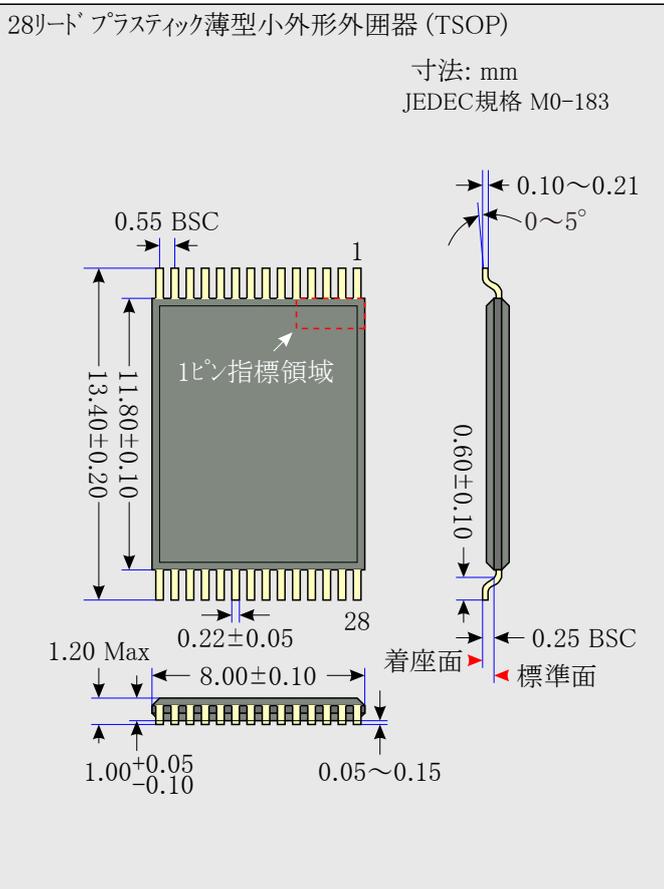
注1: 有害物質使用制限に関する欧州指令(RoHS指令)適合の鉛フリー製品。またハロゲン化合物フリーで完全に安全です。

#### 外囲器形式

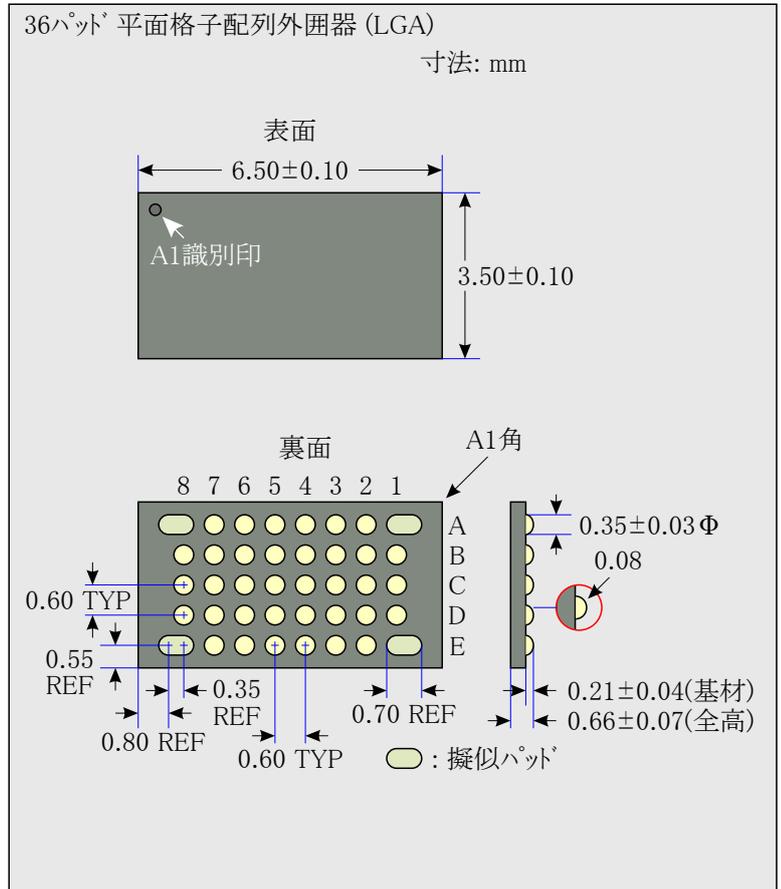
36CK1	36パッド 6.50×3.50×0.85mm 0.60mmピッチ 平面格子配列外囲器 (LGA)
28T	28リード 8×13.4mm プラスチック薄型小外形外囲器1型 (TSOP)

### 34. 外囲器情報

#### 34.1. 28T



#### 34.2. 36CK1



## 35. 障害情報

この章の改訂番号はATmega8HVA/16HVAデバイスの改訂版を参照してください。

ATmega8HVA 改訂A : 既知の障害はありません。

ATmega16HVA 改訂A : 既知の障害はありません。

## 36. データシート改訂履歴

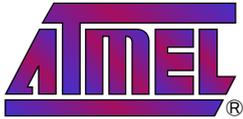
この章内の参照頁番号はこの文書が参照されていることに注意してください。この章内の改訂番号は文書の改訂番号を参照してください。

### 36.1. 8024A-2008年4月 1. 初版

## 目次

特徴	1	12.2. ATmega8HVA/16HVAの割り込みベクタ	34
1. ピン配置	2	13. 外部割り込み	36
1.1. LGA	2	13.1. 概要	36
1.2. TSOP	2	13.2. 外部割り込み用レジスタ	36
1.3. ピン説明	3	14. 高電圧入出力ポート	38
2. 概要	4	14.1. 概要	38
2.1. ATmega8HVAとATmega16HVA間の違い	5	14.2. 標準デジタル入出力としての高電圧ポート	38
3. お断り	5	14.3. 交換ポート機能	39
4. 資料	5	14.4. 高電圧I/Oポート用レジスタ	40
5. データ保持力	5	15. 低電圧入出力ポート	41
6. コード例について	5	15.1. 概要	41
7. AVR CPU コア	6	15.2. 標準デジタル入出力としての低電圧ポート	41
7.1. 概要	6	15.3. 交換ポート機能	44
7.2. ALU (Arithmetic Logic Unit)	6	15.4. 低電圧I/Oポート用レジスタ	47
7.3. ステータスレジスタ	7	16. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器	48
7.4. 汎用レジスタファイル	8	16.1. 概要	48
7.5. スタックポインタ	8	16.2. 外部クロック	48
7.6. 命令実行タイミング	9	16.3. タイマ/カウンタ前置分周器用レジスタ	49
7.7. リセットと割り込みの扱い	9	17. タイマ/カウンタ (T/C0,T/C1)	50
8. AVR メモリ	11	17.1. 特徴	50
8.1. 概要	11	17.2. 概要	50
8.2. 実装書き換え可能なプログラム用フラッシュメモリ	11	17.3. タイマ/カウンタのクロック	51
8.3. データ用SRAMメモリ	11	17.4. 計数器部	51
8.4. データ用EEPROMメモリ	12	17.5. 動作種別	51
8.5. I/Oメモリ (レジスタ)	12	17.6. 捕獲入力部	53
8.6. メモリ関係レジスタ	13	17.7. 比較部	54
9. システムクロックとクロック任意選択	17	17.8. タイマ/カウンタのタイミング	55
9.1. クロック系統とその配給	17	17.9. 16ビット動作でのレジスタアクセス	56
9.2. クロック元	18	17.10. タイマ/カウンタn用レジスタ	58
9.3. 校正付き高速RC発振器	18	18. SPI (直列周辺インターフェース)	61
9.4. 低速RC発振器	18	18.1. 特徴	61
9.5. 超低電力RC発振器	18	18.2. 概要	61
9.6. CPU,I/O,フラッシュ,電圧A/D変換のクロック	18	18.3. SSピンの機能	63
9.7. ウォッチドッグタイマ,電池保護,CC-ADCのクロック	18	18.4. データ転送形式	64
9.8. クロック起動手順	19	18.5. SPI用レジスタ	65
9.9. クロック出力	19	19. クーロンカウンタ (容/残量計専用 $\Delta\Sigma$ A/D変換器)	67
9.10. システムクロック前置分周器	19	19.1. 特徴	67
9.11. 電圧A/D変換器クロック前置分周器	19	19.2. 概要	67
9.12. OSI - 発振器採取インターフェース	19	19.3. 標準動作	68
9.13. クロック関係レジスタ	21	19.4. 定常電流検出動作	69
10. 電力管理と休止形態	23	19.5. 極性切り換えによる変位(オフセット)消去	69
10.1. 休止形態種別	23	19.6. 構成設定と使用法	69
10.2. アイドル動作	24	19.7. クーロンカウンタA/D変換器用レジスタ	70
10.3. A/D変換雑音低減動作	24	20. 電圧ADC (5チャンネル汎用12ビット $\Delta\Sigma$ A/D変換器)	73
10.4. パワーセーブ動作	24	20.1. 特徴	73
10.5. パワーオフ動作	25	20.2. 概要	73
10.6. 電力削減(電力削減レジスタ)	25	20.3. 操作	73
10.7. 消費電力の最小化	26	20.4. 電圧A/D変換器用レジスタ	75
10.8. 電力管理用レジスタ	27	21. 基準電圧と温度感知器	77
11. システム制御とリセット	28	21.1. 特徴	77
11.1. AVRのリセット	28	21.2. 概要	77
11.2. リセット元	28	21.3. 基準電圧と温度感知器用レジスタ	78
11.3. ウォッチドッグタイマ	30	22. 電圧調整器	79
11.4. リセット関係用レジスタ	32	22.1. 特徴	79
12. 割り込み	34	22.2. 概要	79
12.1. 概要	34	22.3. 電圧調整監視器	81

22.4.	電圧調整器用レジスタ	81
23.	電池保護	82
23.1.	特徴	82
23.2.	概要	82
23.3.	短絡保護	82
23.4.	放電過電流保護	82
23.5.	充電過電流保護	82
23.6.	放電高電流保護	83
23.7.	充電高電流保護	83
23.8.	電池保護CPUインターフェース	83
23.9.	電池保護用レジスタ	84
24.	FET制御	89
24.1.	概要	89
24.2.	FET駆動部	90
24.3.	DUVR-深下電圧回復動作	91
24.4.	FET制御用レジスタ	91
25.	デバッグWIRE内蔵デバッグ機能	92
25.1.	特徴	92
25.2.	概要	92
25.3.	物理インターフェース	92
25.4.	ソフトウェア中断点	92
25.5.	デバッグWIREの制限	92
25.6.	デバッグWIRE用レジスタ	92
26.	フラッシュの自己プログラミング	93
26.1.	概要	93
26.2.	自己プログラミングでのフラッシュアドレス指定	93
26.3.	自己プログラミング用レジスタ	97
27.	メモリプログラミング	98
27.1.	プログラムメモリとデータメモリ用施錠ビット	98
27.2.	ヒューズビット	99
27.3.	識票バイト	100
27.4.	校正バイト	100
27.5.	ページ容量	100
27.6.	低電圧直列プログラミング	101
27.7.	高電圧直列プログラミング	104
27.8.	高電圧直列プログラミング手順	104
28.	動作回路	109
29.	電気的特性	111
29.1.	絶対最大定格	111
29.2.	DC特性	111
29.3.	外部割り込み特性	112
29.4.	汎用入出力線特性	113
29.5.	FET駆動部特性	114
29.6.	電源ONとリセットの特性	114
29.7.	SPIタイミング特性	115
29.8.	プログラミング特性	116
30.	代表特性 (暫定)	117
31.	レジスタ要約	118
32.	命令要約	122
33.	注文情報	124
34.	外圍器情報	124
35.	障害情報	125
36.	データシート改訂履歴	126



## 本社

### *Atmel Corporation*

2325 Orchard Parkway  
San Jose, CA 95131  
USA

TEL 1(408) 441-0311  
FAX 1(408) 487-2600

## 国外営業拠点

### *Atmel Asia*

Unit 1-5 & 16, 19/F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
Hong Kong  
TEL (852) 2245-6100  
FAX (852) 2722-1369

### *Atmel Europe*

Le Krebs  
8, Rue Jean-Pierre Timbaud  
BP 309  
78054 Saint-Quentin-en-  
Yvelines Cedex  
France  
TEL (33) 1-30-60-70-00  
FAX (33) 1-30-60-71-11

### *Atmel Japan*

104-0033 東京都中央区  
新川1-24-8  
東熱新川ビル 9F  
アトメル ジャパン株式会社  
TEL (81) 03-3523-3551  
FAX (81) 03-3523-7581

## 製品窓口

### ウェブサイト

[www.atmel.com](http://www.atmel.com)

### 技術支援

[avr@atmel.com](mailto:avr@atmel.com)

### 販売窓口

[www.atmel.com/contacts](http://www.atmel.com/contacts)

### 文献請求

[www.atmel.com/literature](http://www.atmel.com/literature)

**お断り:** 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイトに位置する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえAtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益の損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© Atmel Corporation 2008. 不許複製 Atmel®、ロコとそれらの組み合わせ、AVR®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

## © HERO 2022.

本データシートはAtmelのATmega8HVA/16HVA英語版データシート(Rev.8024A-04/08)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力は、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。

原書に於ける図表番号の不具合を修正したため、原書に対して一部の図表番号が異なります。