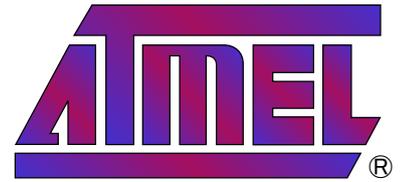


## 特徴

- 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
  - 1MHz当たり、1MIPSに達する高速動作
  - 2周期実行の乗算命令
- データメモリと揮発性プログラムメモリ
  - 実装書き換え可能な8/16Kバイト(4/8K語)フラッシュメモリ内蔵 (10,000回の書き換えが可能)
    - 耐久性: 10,000回の書き換え(書き込み/消去)
    - 施錠ビット保護
    - 個別施錠ビットを持つ任意の2Kバイトブートコード領域
    - チップ内ブートプログラムによる自己書き換え
    - 真の書き込み中の読み出し動作
  - 実装書き換え可能な512バイトのEEPROM
    - 4バイトのページ容量
  - 256/1024バイトの内蔵SRAM
- デバッグWIRE内蔵デバッグ機能
- 内蔵周辺機能
  - 拡張PSC2機能を持つ、1つの12ビット高速電力段制御器(PSC)
    - 柔軟な沈黙時間付き非重複反転PWM出力ピン
    - 可変PWMデューティ比と可変PWM周波数
    - 全PWMレジスタの同期更新
    - 分解能増強動作形態(16ビット)
    - A/D変換同期用の追加レジスタ
    - 捕獲入力
    - 4つの出力ピンと出力配列
  - 1つの12ビット高速電力段制御器(PSC)
    - 事象駆動力率改善(PFC)実装用、自動停止機能
    - 柔軟な沈黙時間付き非重複反転PWM出力ピン
    - 可変PWMデューティ比と可変PWM周波数
    - 全PWMレジスタの同期更新
    - 分解能増強動作形態(16ビット)
    - 捕獲入力
  - 1つの単純な汎用16ビットタイマ/カウンタ
  - 10ビット A/D変換器
    - 最大11のシングルエンド入力チャネルと、1つの完全な差動チャネル対
    - 設定可能な利得(差動チャネルでの×5, ×10, ×20, ×40)
    - 内部基準電圧
  - 1つの10ビット D/A変換器
  - 以下を持つ、3つのアナログ比較器
    - 比較電圧調整用抵抗列
    - 比較電圧調整用D/A変換器
  - 1つのSPI直列インターフェース
  - 3つの外部割り込み
  - 設定可能な専用発振器付きウォッチドッグタイマ
- 特殊マイクロコントローラ機能
  - アイドル、A/D変換雑音低減、パワーダウン、スタンバイの4つの低消費動作
  - 電源ONリセット回路と設定可能な低電圧検出器(BOD)
  - ビット設定可能なI/O空間内のフラグ列(4バイト)
  - SPIポート経由での実装書き込み
  - 校正可能な低電力内蔵RC発振器(8または1MHz、低変動)
  - 高速PWM(32,48,64MHz)とCPU(12,16MHz)用のチップ上PLL(供給元:RCまたはクリスタル)
  - 動的クロック切り替え
  - 温度感知器



8ビット Atmel  
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

AT90PWM81  
AT90PWM161

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. 7734Q-02/12, 7734QJ5-08/22

- 動作電圧 : 2.7~5.5V
- 温度範囲 : -40~105°Cと-40~125°C
- 動作速度
  - 5V : コア16MHz、64MHz PLL
  - 3.3V : コア12MHz、48MHz PLL

## 1. 製品形態

表1-1.に対して異なる製品形態が記述されます。

表1-1. AT90PWM81/161の形態

外圍器	SO20	QFN32	外圍器	SO20	QFN32
ピン数	20	32	A/D変換器入力数	8	11
フラッシュ メモリ容量 (注1)	8/16Kバイト	8/16Kバイト	A/D変換増幅器	1	1
EEPROM容量	512バイト	512バイト	温度感知器	1	1
RAM容量 (注2)	256/1024バイト	256/1024バイト	アナログ比較器	3	3
拡張機能付き12ビット電力段制御器(PSC)	1	1	D/A変換器	1	1
12ビット電力段制御器(PSC)	1	1	D/A変換増幅器	-	-
8ビットタイマ/カウンタ	-	-	UART/DALI	-	-
16ビットタイマ/カウンタ	1	1	SPI	1	1

注1: フラッシュ メモリ容量はAT90PWM81に対して8Kバイト、AT90PWM161に対して16Kバイトです。

注2: RAM容量はAT90PWM81に対して256バイト、AT90PWM161に対して1024(1K)バイトです。

## 2. ピン配置

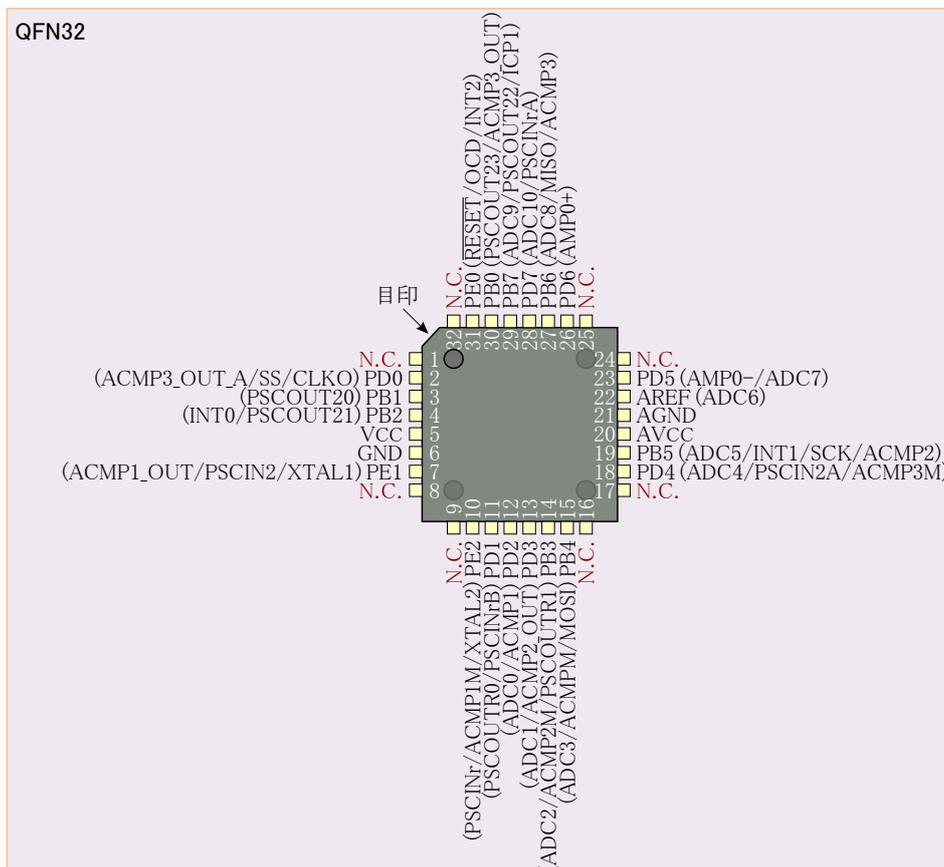
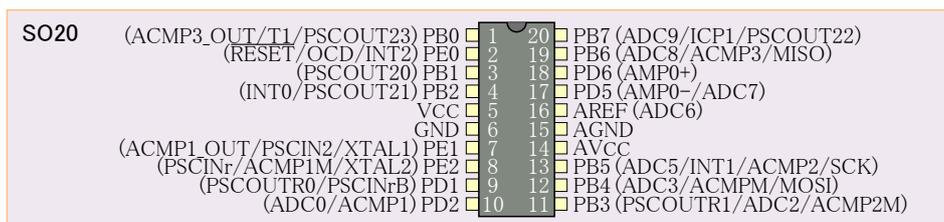


表2-1. 機能説明

表記	名称、機能、交換機能
GND	接地: 0V基準
AGND	アナログ接地: アナログ部用0V基準
VCC	電源:
AVCC	アナログ電源: これはアナログ部用の電源電圧です。通常の使用に関して、このピンは接続されなければなりません。
AREF	アナログ基準電圧: アナログ変換器用基準電圧。これはA/D変換器の基準電圧です。外部アナログ回路によって出力として使うことができます。
CLKO	システム クロック出力
RESET/OCD	リセット入力、チップ上デバッグ入出力
XTAL1	クリスタル用発振増幅器入力
XTAL2	クリスタル用発振増幅器出力
MISO	SPI 主装置入力/従装置出力
MOSI	SPI 主装置出力/従装置出力
SCK	SPI クロック
SS	SPI 従装置選択
INTn	外部割り込みn
Tn	タイマ/カウンタn クロック入力
PSCOUTxn	PSCx 出力n
PSCINx	PSCx デジタル入力
PSCOUT0n	縮小PSC 出力n
PSCINr	縮小PSC デジタル入力
ACMPn	アナログ比較器n 非反転入力
ACMPMn	アナログ比較器n 反転入力
ACMPM	アナログ比較器用反転入力
ACMPn_OUT	アナログ比較器n 出力
AMPn-	アナログ差動増幅器n 入力チャネル
AMPn+	アナログ差動増幅器n 入力チャネル
ADCn	アナログ変換器入力チャネルn

表2-2. ピン割り当て説明

ピン番号		ピン名	汎用	電力段制御器(PSC)	A/D変換器(ADC)	アナログ*
SO20	QFN32					
1	30	PB0	T1	PSCOUT23	-	ACMP3_OUT
2	31	PE0	RESET/OCD, INT2	-	-	-
N/A	2	PD0	CLKO, SS	-	-	ACMP3_OUT_A
3	3	PB1	-	PSCOUT20	-	-
4	4	PB2	INT0	PSCOUT21	-	-
5	5	VCC	電源	-	-	-
6	6	GND	接地	-	-	-
7	7	PE1	XTAL1	PSCIN2	-	ACMP1_OUT
8	10	PE2	XTAL2	PSCINr	-	ACMP1M
9	11	PD1	-	PSCOUTR0, PSCINrB	-	-
10	12	PD2	-	-	ADC0	ACMP1
N/A	13	PD3	-	-	ADC1	ACMP2_OUT
11	14	PB3	-	PSCOUTR1	ADC2	ACMP2M
12	15	PB4	MOSI	-	ADC3	ACMPM
N/A	18	PD4	-	PSCIN2A	ADC4	ACMP3M
13	19	PB5	INT1, SCK	-	ADC5	ACMP2
14	20	AVCC	アナログ電源	-	-	-
15	21	AGND	アナログ接地	-	-	-
16	22	AREF	AREF(アナログ基準電圧)	-	ADC6	-
17	23	PD5	-	-	ADC7	AMP0-
18	26	PD6	-	-	-	AMP+
19	27	PB6	MISO	-	ADC8	ACMP3
N/A	28	PD7	-	PSCINrA	ADC10	-
20	29	PB7	ICP1	PSCOUT22	ADC9	-

## 2.1. ピン説明

### 2.1.1. VCC

デジタル電源ピン。

### 2.1.2. GND

接地ピン。

### 2.1.3. PB7～PB0 (ポートB)

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

ポートBは46頁の表9-3.で一覧されるようにAT90PWM81/161の様々な特殊機能も扱います。

### 2.1.4. PD7～PD0 (ポートD)

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

ポートDは48頁の表9-6.で一覧されるようにAT90PWM81/161の様々な特殊機能も扱います。

### 2.1.5. PE2～PE0 (ポートE), RESET/XTAL1/XTAL2

ポートEは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの3ビットの双方向入出力ポートです。ポートE出力緩衝部は共に高い吐き出し/吸い込み能力の対称駆動特性です。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートEピンには吐き出し電流が流れます。リセット条件が有効になるとき、クロックが動作していても、ポートEピンはHi-Zになります。

RSTDISBLヒューズがプログラム(0)されると、PE0はI/Oピンとして使われます。PE0の電気的特性が他のピンのそれらと異なることに注意してください。

RSTDISBLヒューズが非プログラム(1)の場合、PE0はリセット入力として使われます。クロックが動作していても、最小パルス幅より長いこのピンのLowレベルはリセットを生成します。最小パルス幅は32頁の表7-1.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。

クロック選択ヒューズ設定に依存して、PE1は発振器反転増幅器への入力と内部クロック操作回路への入力として使えます。

クロック選択ヒューズ設定に依存して、PE2は発振器反転増幅器からの出力として使えます。

ポートEの様々な特殊機能は50頁の表9-9.と18頁の「クロック系統とその配給」で詳しく述べられます。

### 2.1.6. AVCC

AVCCはADC入力とA/D変換器用供給電圧(電源)ピンです。例えA/D変換が使われなくても、外部的にVCCへ接続されるべきです。A/D変換が使われる場合、低域通過濾波器を通してVCCに接続されるべきです。

### 3. AVR CPU コア

#### 3.1. 序説

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

#### 3.2. 構造概要

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

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

32個中の6つのレジスタは効率的なアドレス計算ができるデータ空間アドレス指定用の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。

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

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

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

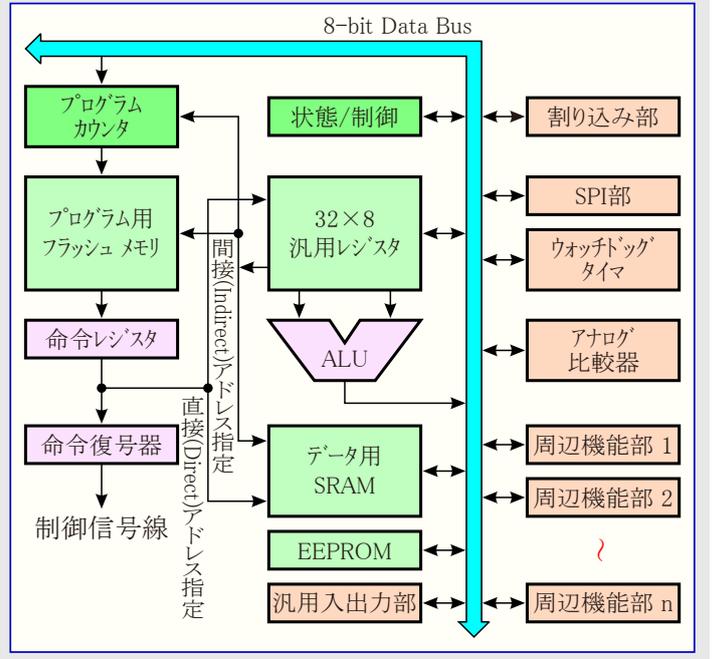
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(D)ビットがあります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタ、SPI、他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリは直接またはレジスタファイルの次のデータ空間位置\$20~\$5Fとしてアクセスできます。加えてAT90PWM81/161にはST/STS/STDとLD/LDS/LDD命令だけ使えるSRAM内の\$60~\$FFに拡張I/O空間があります。

#### 3.3. ALU (Arithmetic Logic Unit)

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

図3-1. AVR MCU構造



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

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「**命令要約**」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。

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

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

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

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

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

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

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

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

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

#### • ビット2 – N: 負フラグ (Negative Flag)

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

#### • ビット1 – Z: ゼロフラグ (Zero Flag)

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

#### • ビット0 – C: キャリーフラグ (Carry Flag)

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

### 3.5. 汎用レジスタ ファイル

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

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

図3-2.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3-2.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3-2. 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

#### 3.5.1. Xレジスタ, Yレジスタ, Zレジスタ

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

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

図3-3. 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)

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

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。

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	-	-	-	-	-	-	-	SP8	SPH
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	6	5	4	3	2	1	0	
\$3D (\$5D)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	

### 3.7. 命令実行タイミング

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

図3-4.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3-5.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3-4. 命令の取得と実行の並列動作

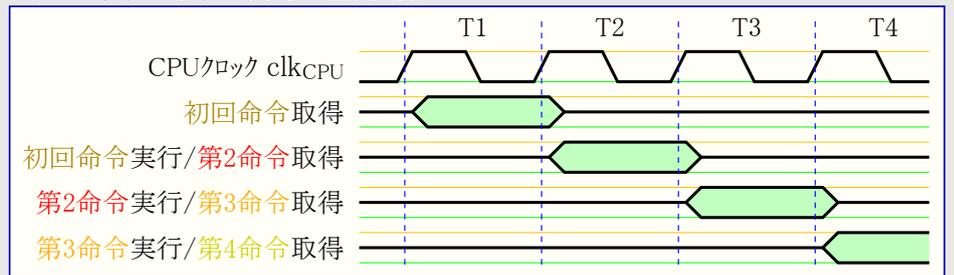
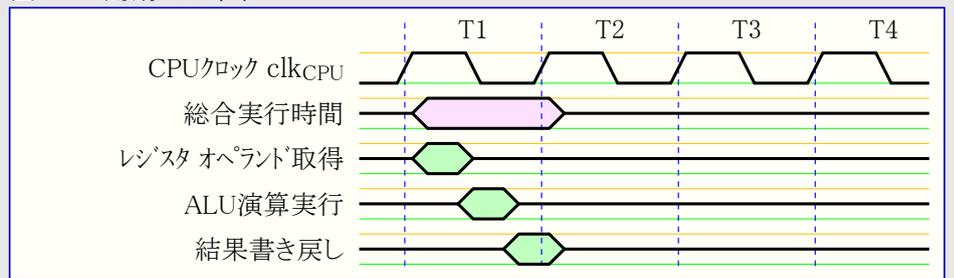


図3-5. 1周期ALU命令



### 3.8. リセットと割り込みの扱い

AVRは多くの異なる割り込み元を提供します。これらの割り込みと独立したリセットベクタ各々はプログラムメモリ空間内に独立したプログラムベクタを持ちます。全ての割り込みは割り込みを許可するために、ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない個別の許可ビットを割り当てられます。BLB02またはBLB12ブート施錠ビットがプログラム(0)されると、プログラムカウンタによっては割り込みが自動的に禁止されるかもしれません。この特質はソフトウェア保護を改善します。詳細については149頁の「メモリプログラミング」章をご覧ください。

既定でのプログラムメモリ空間の最下位アドレスはリセットと割り込みベクタとして定義されます。ベクタの完全な一覧は38頁の「割り込み」で示されます。この一覧は各種割り込みの優先順位も決めます。下位側アドレスがより高い優先順位です。リセットが最高優先順位で次が電力段階制御2捕獲発生(PSC2 CAPT)です。割り込みベクタはMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットの設定(1)によってブートフラッシュ領域先頭へ移動できます。より多くの情報については38頁の「割り込み」を参照してください。リセットベクタもBOOT RSTヒューズのプログラム(0)によってブートフラッシュ領域先頭へ移動できます。139頁の「ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

#### 3.8.1. 割り込みの動き

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、復帰時の回復も自動的に行われなことに注意してください。これはソフトウェアによって扱われなければなりません。

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

```
_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。

### 3.8.2. 割り込み応答時間

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

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

## 4. メモリ

この項はAT90PWM81/161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つの主要なメモリ空間があります。加えてAT90PWM81/161はデータ保存用EEPROMメモリが特徴です。3つのメモリ空間全ては一般的な直線的アドレスです。

### 4.1. 実装書き換え可能なプログラム用フラッシュメモリ

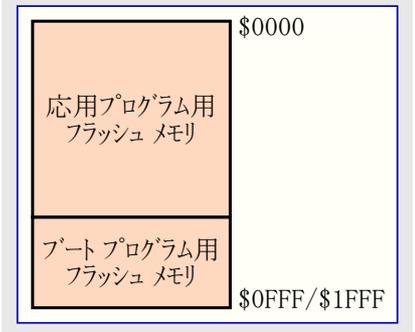
AT90PWM81/161はプログラム保存用に実装書き換え可能な8/16Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、このフラッシュメモリは4/8K×16ビットとして構成されます。ソフトウェア保護のため、フラッシュプログラムメモリ空間はブートプログラム領域と応用プログラム領域の2つに分けられます。

フラッシュメモリは最低10,000回の消去/書き込み回数の耐久性があります。AT90PWM81/161のプログラムカウンタ(PC)は12/13ビット幅、故に4/8Kプログラムメモリ位置のアドレス指定です。ブートプログラム領域の操作とソフトウェア保護用関連ブート施錠ビットは139頁の「[ブートロード支援 - 書き込み中読み出し可能な自己プログラミング](#)」の詳細で記述されます。149頁の「[メモリプログラミング](#)」はSPI(直列)または並列プログラミング動作でのフラッシュメモリプログラミングの詳細な記述を含みます。

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

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

図4-1. プログラムメモリ配置図



### 4.2. データ用SRAMメモリ

図4-2はAT90PWM81/161のSRAMメモリ構成方法を示します。

AT90PWM81/161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512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メモリに、その次の160位置は拡張I/Oメモリ、そして次の256/1024位置はデータ用内蔵SRAMに充てます。

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

直接アドレス指定はデータ空間全体に届きます。

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

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

AT90PWM81/161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、256バイトのデータ用内蔵SRAMはこれら全てのアドレス指定種別を通して全部アクセスできます。レジスタファイルは8頁の「[汎用レジスタファイル](#)」で記述されます。

#### 4.2.1. データ用SRAMメモリアクセスタイミング

この項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4-3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4-2. データメモリ配置図

		アドレス
レジスタファイル (32×8)	R0	\$0000
	R31	\$001F
I/Oレジスタ (64×8)	\$00	\$0020
	\$3F	\$005F
拡張I/Oレジスタ (160×8)	\$0060	\$0060
	\$00FF	\$00FF
内蔵SRAM (256/1024×8)	\$0100	\$0100
	\$01FF/\$04FF	\$01FF/\$04FF

注: 赤字はI/Oアドレス

図4-3. データ用内蔵SRAMアクセス周期



### 4.3. データ用EEPROMメモリ

AT90PWM81/161は512バイトのデータ用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/書き込み周回の耐久性があります。CPUとEEPROM間のアクセスは以降のEEPROMアドレスレジスタ、EEPROMデータレジスタ、EEPROM制御レジスタで詳細に記述されます。

SPI(直列)と並列でのEEPROMプログラミングの詳細については158頁の「直列プログラミング」と152頁の「並列プログラミング」をご覧ください。

#### 4.3.1. EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROMの書き込み(訳注:原文はアクセス)時間は13頁の表4-1.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。厳重に濾波した電源では電源投入/切断でVCCが緩やかに上昇または下降しそうです。これはデバイスが何周期かの時間、使われるクロック周波数に於いて最小として示されるより低い電圧で走行する原因になります。これらの状態で問題を避ける方法の詳細については14頁の「EEPROMデータ化けの防止」をご覧ください。

予期せぬEEPROM書き込みを防止するため、特別な書き込み手順に従わなければなりません。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。

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

ビット	15	14	13	12	11	10	9	8	
\$22 (\$42)	–	–	–	–	–	–	–	EEAR8	EEARH
Read/Write	R	R	R	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	不定	
ビット	7	6	5	4	3	2	1	0	
\$21 (\$41)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Read/Write	R/W								
初期値	不定								

- ビット15~9 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

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

EEPROMアドレスレジスタ(EEARHとEEARL)は512バイトEEPROM空間のEEPROMアドレスを指定します。EEPROMデータバイトは0~511間で直線的に配されます。EEARの初期値は不定です。EEPROMがアクセスされるであろう前に適切な値が書かれねばなりません。

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

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

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

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

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

ビット	7	6	5	4	3	2	1	0	
\$1F (\$3F)	NVMBSY	EEPAGE	EEPM1	EEPM0	EERIE	EEMWE	EEWE	EERE	EECR
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	0	0	不定	0	

- ビット7 – NVMBSY : 不揮発性メモリ多忙 (Non-volatile memory busy)

NVMBSYビットはNVMメモリ(フラッシュ、EEPROM、施錠ビット)がプログラミングで多忙なことを示す状態ビットです。一旦プログラミング操作が開始されると、このビットが設定(1)されて、プログラミング操作が完了されるまで設定(1)に留まります。

### •ビット6 – EEPAGE : EEPROMページ アクセス(複数バイト アクセス形態) (EEPROM Page access (multiple bytes access mode))

EEPAGEの1書き込みは複数バイト アクセス形態を許可します。それは多数のバイトが同時にEEPROMへプログラミングできることを意味します。EEPAGEビットが1を書かれると、EEPAGEビットはEEPROMプログラミング操作が完了されるまで設定(1)に留まります。代わりにこのビットはソフトウェアでEEPROM一時緩衝部が破棄された時にも解除(0)されます(EEPROMビット説明をご覧ください)。EEPROM書き込み許可(EWE)が1の間のどんなEEPAGE書き込みも無視されます。EEPROM一時ページへのデータ設定方法とEEPAGEビットの使い方の詳細については14頁の「1回の非分断操作での複数バイト書き込み」項をご覧ください。

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

EEPROMプログラミング種別ビット設定はEEPROM書き込み許可(EWE)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4-1.で示されます。EWEが設定(1)されている間はEEPROMへのどの書き込みも無視されます。リセット中、EEPROM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4-1. EEPROMプログラミング種別

EEPMM1	EEPMM0	プログラミング時間	動作
0	0	3.4ms	1操作での消去と書き込み(非分離操作)
0	1	1.8ms	消去のみ
1	0	1.8ms	書き込みのみ
1	1	-	EEPROM一時ページ緩衝部破棄

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

EEMWEビットはEEPROM書き込み許可(EWE)ビットの1書き込みが有効か無効かどちらかを決めます。EEMWEが設定(1)されると、4クロック周期内のEWE設定(1)は選んだアドレスのEEPROMを書きます。EEMWEが0なら、EWE設定(1)は無効です。EEMWEがソフトウェアによって設定(1)されてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。EEPROM書き込み手順については次のEWE記述をご覧ください。

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

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

1. EEPROM書き込み許可(EWE)ビットが0になるまで待ちます。
2. SPM制御/状態レジスタ(SPMCSR)のSPM操作許可(SPMEN)ビットが0になるまで待ちます。
3. 今回のEEPROMアドレスをEEPROMアドレスレジスタ(EEAR)に書きます。(任意、省略可)
4. 今回のEEPROMデータをEEPROMデータレジスタ(EEDR)に書きます。(任意、省略可)
5. EEPROM制御レジスタ(EECR)のEEPROM主書き込み許可(EEMWE)ビットに1を、EEPROM一時許可(EWE)ビットに0を同時に書きます。
6. EEMWEビット設定後4クロック周期内にEEPROM一時許可(EWE)ビットへ論理1を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMは書き込みができません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを検査しなければなりません。2.はソフトウェアがフラッシュメモリの書き込みをCPUに許すブートローダを含む場合にだけ関係します。フラッシュメモリが決してCPUによって更新されないなら、2.は省略できます。ブートプログラミングについての詳細に関しては139頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」をご覧ください。

**警告:** 手順5.と6.間の割り込みはEEPROM主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書き込み許可(EWE)ビットはハードウェアによって解除(0)されます。使用者ソフトウェアは次バイトを書く前にこのビットをポーリングし、0を待つことができます。EWEが設定(1)されてしまうと、次の命令が実行される前にCPUは2周期停止されます。

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

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

使用者は読み込み操作を始める前にEWE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4-2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4-2. 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)	26,368	3.3ms

#### 4.3.5. 1回の非分断操作での複数バイト書き込み

EEPROMへ複数バイトを書くことが可能です。プログラミング(消去/書き込み)を始める前に、書かれるべきデータがEEPROM一時ページ緩衝部内に格納されなければなりません。EEPROMページアクセス(EEPAGE)の1書き込みが格納操作を許可します。

EEPAGEビットが1を書かれると、EEPROM一時ページ緩衝部は格納準備が整います。データをEEPROM一時ページ緩衝部へ格納するには、EEPROMアドレス下位レジスタ(EEARL)とEEPROMデータレジスタ(EEDR)の各々にアドレスとデータが書かれなければなりません。EEDRが更新される時にデータが格納されることに注意してください。従って、データの前にアドレスが書かれなければなりません。EEPROM一時ページ緩衝部が満たされる、または書かれるべき全データが格納されるまで、この操作が繰り返されます。プログラミング操作実行に先立って格納されたバイト数はEEPROM一時ページ容量を超えてはなりません。プログラミング操作実行前にEEPROM一時ページ緩衝部の各バイトに対して複数回書けないことに注意してください。同じバイトが複数回書かれた場合、EEPROM一時ページの内容は書かれたデータ間でビット単位ANDになります(換言すると、\$AAと\$55が同じバイトに格納された場合、結果は\$00になります)。EEPROM一時緩衝部はプログラミング操作完了後に新しいデータの準備が整います。代わりに、EEPROMプログラミング種別(EPMn)ビットが'11'の場合に(EEPROM主書き込み許可(EEMWE)が書かれた後の4周期内に)EEPROM書き込み許可(EWE)を書くことによって、EEPROM一時緩衝部は破棄されて新しいデータの準備が整います。EEPROM一時緩衝部が破棄されると、EEPAGEビットが解除(0)されます。EEPROM一時緩衝部へのデータ格納は3 CPUクロック周期かかります。EEPAGEが設定(1)されている間にEEDRが書かれた場合、3周期の操作を保証するためにCPUは停止します。

各種のビットとレジスタは次のような順でアクセスされるべきです。

1. EEPROM制御レジスタ(EECR)でEEPAGEを書いてください(EEPROM一時緩衝部の格納が許可されます)。
  2. ページ内のアドレスバイトに必要とされるアドレスビットをEEARLへ書いてください。
  3. データをEEDRに書いてください。
  4. 緩衝部が満たされる、または全データが格納されるまで上の2.と3.を繰り返してください。
  5. 残りのアドレスビットをEEPROMアドレスレジスタ(EEARH;EEARL)に書いてください。
    - a. 実行されるべきプログラミング動作(EPMnビット)を選んでください。プログラミング動作を開始するために(EEMWEビットが書かれた後の4周期内に)EECRでEWEビットを書いてください。EEPROM一時ページ緩衝部はプログラミング動作完了後に自動消去します。
- または
- b. 異常状態が起きた場合、格納はソフトウェアによって終了されるべきです。EPM1,0に'11'書き、(EEMWEビットが書かれた後の4周期内に)EWEを書くことによって破棄してください。

#### 4.3.6. コード例

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

##### アセンブリ言語プログラム例

```
EEPROM_WR: SBIC   EECR, EEPE           ;EEPROMプログラミング完了ならばスキップ
            RJMP   EEPROM_WR         ;以前のEEPROMプログラミング完了まで待機
;
            LDI   R19, (0<<EEP1) | (0<<EEP0) ;プログラミング種別値取得(本例は非分離)
            OUT   EECR, R19          ;対応プログラミング種別設定
            OUT   EEARH, R18         ;EEPROMアドレス上位バイト設定
            OUT   EEARL, R17        ;EEPROMアドレス下位バイト設定
            OUT   EEDR, R16         ;EEPROM書き込み値を設定
            SBI   EECR, EEMWE       ;EEPROM主書き込み許可ビット設定
            SBI   EECR, EEWE        ;EEPROM書き込み開始(書き込み許可ビット設定)
            RET                       ;呼び出し元へ復帰
```

##### C言語プログラム例

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    while (EECR & (1<<EEPE));           /* 以前のEEPROMプログラミング完了まで待機 */
    EECR = (0<<EEP1) | (0<<EEP0);       /* 対応プログラミング種別設定 */
    EEAR = uiAddress;                  /* EEPROMアドレス設定 */
    EEDR = ucData;                     /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMWE);                 /* EEPROM主書き込み許可 */
    EECR |= (1<<EEWE);                  /* EEPROM書き込み開始 */
}
```

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

##### アセンブリ言語プログラム例

```
EEPROM_RD: SBIC   EECR, EEPE           ;EEPROMプログラミング完了ならばスキップ
            RJMP   EEPROM_RD         ;以前のEEPROMプログラミング完了まで待機
;
            OUT   EEARH, R18         ;EEPROMアドレス上位バイト設定
            OUT   EEARL, R17        ;EEPROMアドレス下位バイト設定
            SBI   EECR, EERE       ;EEPROM読み出し開始(読み込み許可ビット設定)
            IN    R16, EEDR          ;EEPROM読み出し値を取得
            RET                       ;呼び出し元へ復帰
```

##### C言語プログラム例

```
unsigned char EEPROM_read(unsigned int uiAddress)
{
    while (EECR & (1<<EEPE));           /* 以前のEEPROMプログラミング完了まで待機 */
    EEAR = uiAddress;                  /* EEPROMアドレス設定 */
    EECR |= (1<<EERE);                 /* EEPROM読み出し開始 */
    return EEDR;                       /* EEPROM読み出し値を取得,復帰 */
}
```

#### 4.3.7. EEPROMデータ化けの防止

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

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

EEPROMデータ化けは次の推奨設計によって容易に避けられます。

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

#### 4.4. ヒューズ ビット

AT90PWM81/161には3つのヒューズ バイトがあります。表4-3.~5.は全てのヒューズの概略機能とヒューズ バイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

表4-3. 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
PSC2RB	7	電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSC2RBA	6	OUT22と23用の電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRRB	5	縮小電力段制御器(PSCR)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRV	4	PSCOUTとPSCOUTRのリセット値	1 (非プログラム) High
PSCINRB	3	PSCとPSCRの入力 リセット時動作	1 (非プログラム) 標準ポート動作
BODLEVEL2	2	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注1)	1 (非プログラム)
BODLEVEL1	1		0 (プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: BODLEVELヒューズの復号については33頁の表7-2.をご覧ください。

表4-4. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
RSTDISBL(注1)	7	PE0がI/OピンかまたはRESETピンかを選択します。	1 (非プログラム) PE0はRESETピン
DWEN	6	デバッグ WIRE機能許可。	1 (非プログラム) デバッグ WIRE不許可
SPIEN (注2)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON (注3)	4	ウォッチドッグ タイマ常時有効。	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BOOTSZ1	2	ブート ロータ容量選択。(表20-7.参照) (注4)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOORST	0	リセット ベクタ(応用領域またはブート ロータ領域)選択	1 (非プログラム) 応用領域

注1: RSTDISBLヒューズの記述については50頁の「ポートEの交換機能」をご覧ください。

注2: SPIENヒューズは直列プログラミングでアクセスできません。

注3: 詳細については37頁の「WDTCSR - ウォッチドッグ タイマ制御レジスタ」をご覧ください。

注4: BOOTSZ1,0既定値は最大ブート ロータ容量になります。

表4-5. ヒューズ 下位バイト一覧

名称	ビット	意味	既定値
CKDIV8 (注4)	7	システム クロック 8分周選択。	0 (プログラム) 8分周
CKOUT (注3)	6	システム クロック出力許可。	1 (非プログラム) 不許可
SUT1	5	起動時間選択。 (注1)	1 (非プログラム)
SUT0	4		0 (プログラム)
CKSEL3	3	クロック種別選択。 (注2)	0 (プログラム)
CKSEL2	2		0 (プログラム)
CKSEL1	1		1 (非プログラム)
CKSEL0	0		0 (プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については20頁の表5-4.をご覧ください。

注2: CKSEL3~0の既定設定は8MHz校正付き内蔵RC発振器になります。詳細については19頁の表5-1.をご覧ください。

注3: CKOUTはポートD0に出力することをシステム クロックに許します。詳細については22頁の「クロック出力緩衝部」をご覧ください。

注4: 詳細については25頁の「システム クロック前置分周器」をご覧ください。

ヒューズ ビットの状態はチップ消去によって影響されません。施錠ビット1(LB1)がプログラム(0)されると、ヒューズ ビットが固定されることに注意してください。施錠ビットをプログラム(0)する前にヒューズ ビットをプログラミング(書き込み)してください。

#### 4.5. I/O メモリ (レジスタ)

AT90PWM81/161のI/O空間定義は181頁の「レジスタ要約」で示されます。

AT90PWM81/161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。AT90PWM81/161はINやOUT命令で予約した64位置で支援されるより多くの周辺機能部を持つ複合マイクロコントローラです。SRAM(データ空間)内\$60～\$FFの拡張I/O領域に対してはLD/LDS/LDDとST/STS/STD命令だけが使えます。

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

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

I/Oと周辺制御レジスタは以降の項で説明されます。

#### 4.6. 汎用I/Oレジスタ

AT90PWM81/161は4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。

##### 4.6.1. 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	
\$19 (\$39)	(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	

##### 4.6.2. 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	
\$1A (\$3A)	(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	

##### 4.6.3. 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	
\$1B (\$3B)	(MSB)							(LSB)	GPIOR2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

## 5. システム クロックとクロック選択

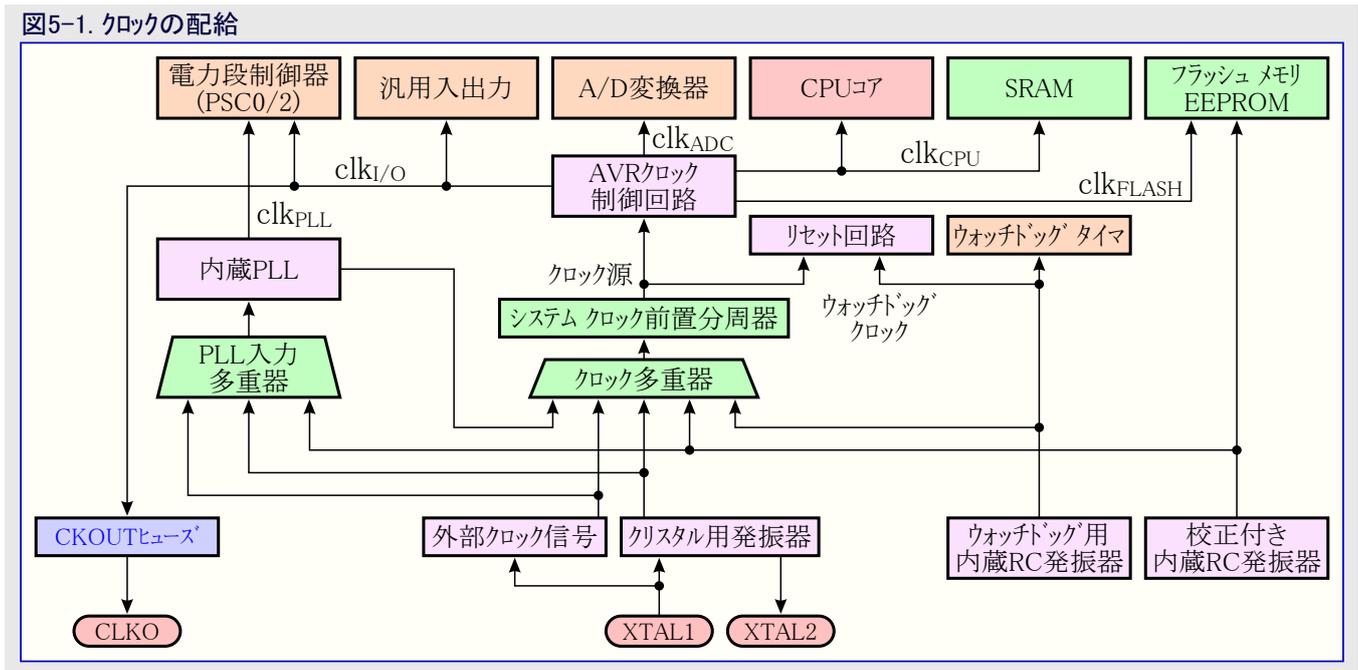
AT90PWM81/161は多数のクロック元を提供します。これらは内部と外部の2つの分野に分けることができます。

リセット後、**CKSEL**ヒューズが1つのクロック元を選びます。一旦デバイスが走行すると、ソフトウェア クロック切り替えが他のクロック元で利用可能です。

クロック切り替え管理のためにいくつかのハードウェア制御が提供されますが、或る特定手順が遵守されなければなりません。或る設定は使用者を不適切な構成設定でのデバイスプログラミングに導くかもしれません。

### 5.1. クロックシステムとその配給

図5-1はAVR内の主要なクロックシステムとその配給を示します。全てのクロックが与えられた時間活性でないかもしれません。消費電力低減のため、(29頁の「電力管理と休止形態」での)各種休止形態の使用、または(23頁の「動的クロック切り替え」での)動的クロック切り替えの機能を使うことにより、使われていない部分のクロックを停止することができます。クロックシステムは以下で詳述されます。



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

CPUクロックはAVRコアの動作と関係する系統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。

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

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

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

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

#### 5.1.4. PLLクロック - clk<sub>PLL</sub>

PLLクロックは64MHzから直接的にクロック駆動されるのを電力段制御器(PSC)に許します。16MHzクロックもCPU用に分周されます。

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

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

## 5.2. クロック元

このデバイスは下で示されるように、フラッシュヒューズビット(既定)、または(動的クロック切り替え回路)クロック選択レジスタ(CLKSELR)によって選択可能な以下のクロック元選択を持ちます。

表5-1. デバイス クロック任意選択(注1)、PLL供給元とPE1,PE2の機能

クロック種別	システム クロック	PLL入力 (注2)	CKSEL3~0 (注3) CSEL3~0 (注4)	PE1	PE2
外部クロック信号	外部クロック	RC発振器	0 0 0 0	CLKI	I/O
4分周PLL出力(内部RCによる16MHz駆動)	4分周PLL	RC発振器	0 0 0 1	I/O	I/O
校正付き8MHz内部RC発振器	RC発振器	RC発振器	0 0 1 0	I/O	I/O
128kHz内部RC発振器(ウォッチドッグ発振器)	WD発振器	(使用不可)	0 0 1 1	I/O	I/O
外部クリスタル/セラミック用発振器によって駆動される4分周PLL出力	4分周PLL	外部発振	0 1 0 0	XTAL1	XTAL2
外部クロック信号によって駆動される4分周PLL出力	4分周PLL	外部クロック	0 1 0 1	CLKI	I/O
校正付き1MHz内部RC発振器	RC発振器	(使用不可)	0 1 1 0	I/O	I/O
(予約)	-	-	0 1 1 1	-	-
外部クリスタル/セラミック用発振器(0.4~0.9MHz)	外部発振	外部発振	1 0 0 x	XTAL1	XTAL2
外部クリスタル/セラミック用発振器(0.9~3.0MHz)	外部発振	外部発振	1 0 1 x	XTAL1	XTAL2
外部クリスタル/セラミック用発振器(3.0~8.0MHz)	外部発振	外部発振	1 1 0 x	XTAL1	XTAL2
外部クリスタル/セラミック用発振器(8.0~16.0MHz)	外部発振	外部発振	1 1 1 x	XTAL1	XTAL2

注1: 全てのヒューズは1=非プログラム、0=プログラムです。

注2: PLLは公称8MHzクロック元によって駆動されなければなりません。

注3: フラッシュのヒューズビットです。

注4: クロック選択レジスタ(CLKSELR)のビットです。

注5: 外部発振: 外部クリスタル/セラミック用発振器です。

注6: RC発振器: 校正付き内蔵RC発振器(1MHzまたは8MHz)です。

注7: WD発振器: ウォッチドッグ用128kHz内部RC発振器です。

注8: 外部クロック: 外部クロック信号入力です。

各クロック選択に対する様々な選択は次項で得られます。

CPUがパワーダウンから起きる時か、または動的クロック切り替え回路によって新しいクロック元が許可される時に、選んだクロック元が始動時間に使われ、命令実行開始前に安定な発振器動作を保証します。

CPUがリセットから始まる時は通常動作開始前に安定電圧へ達することを電源に許す付加遅延があります。ウォッチドッグ発振器はこの起動時間の実時間部のタイミングに使われます。各計時完了に使われるウォッチドッグ(WDT)発振器の周期数は表5-2.で示されます。

表5-2. WDT発振器の代表計時完了値、周期数

代表的な計時完了値	周期数
4ms	512
64ms	8K (8192)

### 5.2.1. 既定のクロック元

本デバイスは常にCKSELヒューズによって定義されるクロック元とSUTヒューズによって定義される始動時間を用いてリセットから始動します。この構成設定はリセットに於いてクロック選択レジスタ(CLKSELR)にラッチされます。デバイスは常にCLKSELRレジスタ(クロック元選択(CSEL3~0)とクロック始動時間(CSUT1,0))によって定義されたクロック元を使って電源ONで始動します。

本デバイスはCKSEL=0010, SUT=10, CKDIV8=プログラム(0)で出荷されます。従って既定のクロック元設定は最長起動時間の(8.0MHzの)校正付き内蔵RC発振器で、8分周の初期システムクロックです。この既定設定は全ての使用者が実装または並列書き込み器を使い、それらを希望したクロック元設定にできることを保証します。この構成設定はISPツール使用時に注意しなければなりません。

### 5.2.2. 校正付き内蔵RC発振器

既定による内蔵RC発振器は概ね8.0MHzまたは1MHzのクロックを提供します。電圧と温度に依存しますが、このクロックは使用者によって非常に高精度な校正ができます。

8MHzと1MHz間の切り替えはMCUCRレジスタの校正付き8/1MHz RC発振器周波数選択(CKRC81)ビットによって行われます。より多くの詳細については27頁の「MCUCR - MCU制御レジスタ」をご覧ください。RC発振器はクロック元選択(CSEL)またはクロック始動時間(CSUT)の2つの構成設定によってアクセスすることができます。リセットに於いて、CKRC81ビットはCKSEL値と一致する値で初期化(1:CKSEL3~0=0110、0:他の全ての値)されます。

RC発振器はCKSEL3~0またはCSEL3~0の何れかの構成設定がシステムクロックまたはPLL供給元クロックとして使われている間で活性です。RC発振器は以下のCKSEL3~0またはCSEL3~0の場合に禁止されます。

- 0011 (128kHz内部RC発振器)
- 0100,0101 (外部クロック信号または外部クリスタル/セラミック用発振器によって駆動される4分周PLLシステム クロック)
- 1100,1101 (外部クリスタル/セラミック用発振器)

このデバイスはCKDIV8ヒューズがプログラム(0)で出荷されます。より多くの詳細については25頁の「システム クロック前置分周器」をご覧ください。このクロックは表5-1.で示されるようにCKSELヒューズまたはCSEL領域のプログラミング(書き込み)によってシステム クロックとして選べます。選んだなら、外部部品なしで動作します。リセット中、ハードウェアが発振校正(OSCCAL)レジスタに校正值バイトを設定し、これによってRC発振器を自動的に校正します。この校正の精度は162頁の表22-1.で工場校正として示されます。

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

この発振器がチップ(システム)クロックとして使われる時にウォッチドッグ発振器は未だウォッチドッグ タイムとリセット付加遅延タイムに使われます。予め設定された校正值のより多くの情報については151頁の「校正バイト」項をご覧ください。

この発振器が選ばれると、起動時間は表5-4.で示されるようにSUTヒューズによって決定されます。

表5-3. 校正付き内蔵RC発振器動作種別(注1,3)

CKSEL3~0 CSEL3~0	周波数範囲(注2) (MHz)
0 0 1 0	7.6~8.4
	0.95~1.05 (注4)

注1: デバイスはこの選択で出荷されます。

注2: この周波数範囲は暫定値です。実際の値はTBDです。

注3: この8MHz周波数がデバイス仕様(VCCに依存)を越える場合、内部周波数を8分周するためにCKDIV8ヒューズをプログラム(0)にできます。

注4: 8MHzと1MHzの切り替えはMCUCRのCKRC81ビットによって行われます。

表5-4. 校正付き内蔵RC発振器用起動遅延時間選択表 (SUT1,0:ヒューズ、CSUT1,0:CLKSELレジスタ内のビット)

SUT1,0 CSUT1,0	パワーダウンからの 起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
0 0	6×CK	14×CK (注2)	低電圧検出リセット(BOD)許可
0 1	6×CK	14×CK+4ms	高速上昇電源
1 0 (注1)	6×CK	14×CK+64ms	低速上昇電源
1 1	(予約)		

注1: デバイスはこの選択で出荷されます。

注2: RSTDISBLヒューズがプログラム(0)されると、プログラミング動作への移行可を保証するため、付加遅延時間は14×CK+4.1msに増やされます。

### 5.2.2.1. 工場でのRC発振器校正

このRC発振器は3V,25°Cに於いて8MHzの目標周波数に対して±1%の精度で校正されます。対応する値ASCAL(@amb)が識票列に格納され、リセットで自動的にOSCCALレジスタへ設定されます。このRC発振器は±5%の精度で105°Cまたは125°C(対製品版)で試験されます。

### 5.2.3. 128kHz内部発振器

128kHz内部発振器は128kHzのクロックを供給する低電力発振器です。この周波数は3V,25°Cでの公称値です。このクロックは表5-1.で示されるようにCKSELヒューズまたはクロック元選択(CSEL)領域のプログラミング(設定)によってシステム クロックとして選べます。

このクロック元が選ばれると、起動時間は表5-5.で示されるようにSUTヒューズまたはクロック始動時間(CSUT)領域によって決定されます。

表5-5. 128kHz内部発振器用起動遅延時間選択表 (SUT1,0:ヒューズ、CSUT1,0:CLKSELレジスタ内のビット)

SUT1,0 CSUT1,0	パワーダウンからの 起動遅延時間	リセットからの 付加遅延時間	推奨使用法
0 0	6×CK	14×CK	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4ms	高速上昇電源
1 0	6×CK	14×CK+64ms	低速上昇電源
1 1	(予約)		

## 5.2.4. クリスタル用発振器

XTAL1とXTAL2は図5-2.で示されるように、チップ上の発振器としての使用に設定できる反転増幅器の各々、入力と出力です。クリスタル発振子またはセラミック振動子のどちらでも使えます。

C1とC2はクリスタル発振子とセラミック振動子の両方について常に等しくすべきです。このコンデンサの最適値は使うクリスタル発振子やセラミック振動子、浮遊容量の量、その環境の電磁雑音に依存します。クリスタル発振子使用に対するコンデンサ選択について初期の指針のいくつかは表5-6.で与えられます。セラミック振動子については製造業者によって与えられたコンデンサ値が使われるべきです。

低電力発振器は示された周波数範囲で各々最適化された3つの異なる種別で動作できます。この動作種別は表5-6.で示されるようにCKSEL3~1ヒューズまたはクロック元選択(CSEL3~1)領域によって選ばれます。

CKSEL0ヒューズはSUT1,0ヒューズと、またはCSEL0はCSUT1,0領域と共に表5-7.で示されるように起動時間を選びます。

図5-2. クリスタル発振子接続図

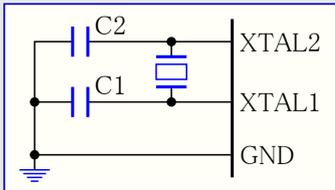


表5-6. クリスタル用発振器動作種別 (CKSEL3~0:ヒューズ、CSEL3~0:CLKSELレジスタ内のビット)

CKSEL3~1, CSEL3~1	周波数範囲	推奨C1,2容量
100 (注)	0.4~0.9MHz	-
101	0.9~3.0MHz	12~22pF
110	3.0~8.0MHz	12~22pF
111	8.0~16MHz	12~22pF

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

表5-7. クリスタル発振子/セラミック振動子用発振器起動遅延時間選択表

CKSEL0 CSEL0	SUT1,0 CSUT1,0	パワーダウンからの 起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
0	0 0	258×CK (注1)	14×CK+4ms	外部セラミック振動子、高速上昇電源
	0 1	258×CK (注1)	14×CK+64ms	外部セラミック振動子、低速上昇電源
	1 0	1K×CK (注2)	14×CK	外部セラミック振動子、低電圧検出(BOD)リセット許可
	1 1	1K×CK (注2)	14×CK+4ms	外部セラミック振動子、高速上昇電源
1	0 0	1K×CK (注2)	14×CK+64ms	外部セラミック振動子、低速上昇電源
	0 1	16K×CK	14×CK	外部クリスタル発振子、低電圧検出(BOD)リセット許可
	1 0	16K×CK	14×CK+4ms	外部クリスタル発振子、高速上昇電源
	1 1	16K×CK	14×CK+64ms	外部クリスタル発振子、低速上昇電源

注: CKSEL0とSUT1,0はヒューズビットで、CSEL0とCSUT1,0はクロック選択レジスタ(CLKSELR)内のビットです。

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

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

## 5.2.5. 外部クロック信号

この外部クロック元からデバイスを駆動するには、CLKIが図5-3.で示されるように駆動されるべきです。外部クロックでデバイスを走行するには、表5-1.で示されるようにCKSELヒューズまたはクロック元選択(CSEL)領域がプログラム(設定)されなければなりません。

このクロック元が選ばれると、起動時間は表5-8.で示されるようにSUTヒューズまたはクロック始動時間(CSUT)領域によって決定されます。

図5-3. 外部クロック信号駆動接続図

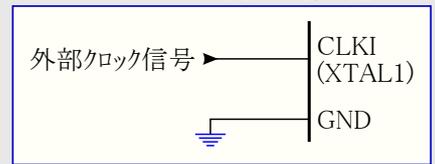


表5-8. 外部クロック信号駆動用起動遅延時間選択表 (SUT1,0:ヒューズ、CSUT1,0:CLKSELレジスタ内のビット)

SUT1,0 CSUT1,0	パワーダウンからの 起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
0 0	6×CK	14×CK	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4ms	高速上昇電源
1 0	6×CK	14×CK+64ms	低速上昇電源
1 1			(予約)

## 5.2.6. PLL

高い周波数と正確なPWM波形を生成するために、電力段階制御器(PSC)は高周波数クロック入力が必要です。このクロックはPLLによって生成されます。全てのPWM精度を維持するため、PLLの周波数係数はソフトウェアによって設定されなければなりません。

AT90PWM81/161の内部PLLは公称8MHzから通倍されたクロック周波数を生成します。この8MHzのPLL入力クロック元は3つの可能な供給元から選ぶことができます(図5-4をご覧ください。)

- 内部RC発振器
- クリスタル用発振器
- 外部クロック信号

内部PLLはPLL制御/状態レジスタ(PLLCSR)のPLL許可(PLLE)ビットが設定(1)される時だけ許可されます。PLLCSRのPLL固定化(PLOCK)ビットはPLLが固定化されている時に設定(1)されます。

ヒューズによってクロック元として選ばれると、PLL通倍係数は6の値で初期化され、3V電源に適合します。

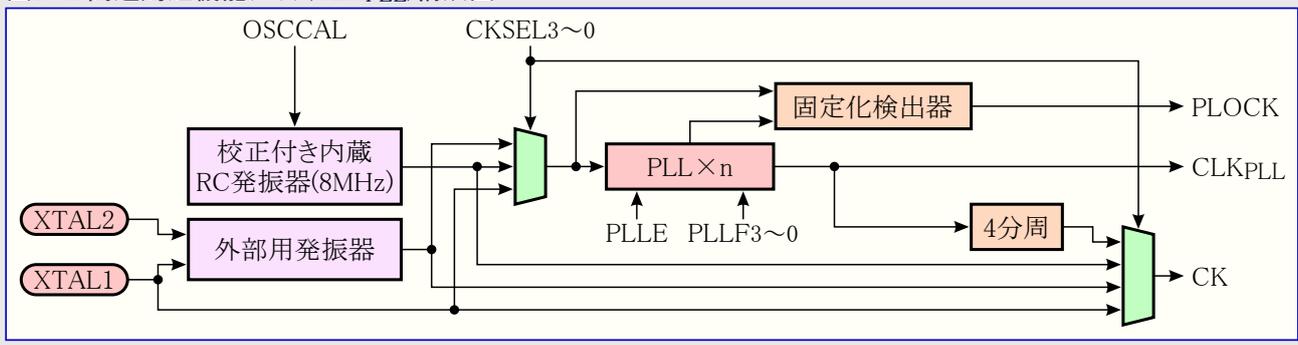
PLLの正しい固定化を保証するために、PLLは8MHz近くに留まらなければならない供給元発振器で固定化されます。

内部RC発振器とPLLはパワーダウンとスタンバイの休止形態でOFFへ切り替えられます。

表5-9. PLLシステム クロック用起動遅延時間選択表

CKSEL3~0 (クロック供給元)	SUT1,0	パワーダウンからの 起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)
0101 (外部クロック信号)	00	16K×CK	14×CK
	01	16K×CK	14×CK+4ms
	10	16K×CK	14×CK+4ms
	11	16K×CK	14×CK+64ms
0101 (外部クリスタル/セラミック発振器)	00	1K×CK	14×CK
	01	1K×CK	14×CK+4ms
	10	1K×CK	14×CK+64ms
	11	16K×CK	14×CK
0001 (校正付き内蔵RC発振器)	00	1K×CK	14×CK
	01	1K×CK	14×CK+4ms
	10	1K×CK	14×CK+64ms
	11	(予約)	

図5-4. 高速周辺機能クロック(CLKPLL)構成図



## 5.2.7. クロック出力緩衝部 (外部クロック出力)

デバイスはCLKOピンにシステムクロックを出力することができます。出力を許可するには、CKOUTヒューズまたはクロック選択レジスタ(CLKSEL R)のクロック出力(COUT)ビットがプログラム(設定)されなければなりません。この形態はチップのクロックがシステム上の他の回路を駆動する時用です。リセット中にクロックが出力されないのと、入出力ピンの標準動作が無効にされることに注意してください。クロックがCLKOに出力される時にどのクロック元も選ぶことができます。システムクロック前置分周器が使われると、分周されたシステムクロックが出力されます。

### 5.3. 動的クロック切り替え

#### 5.3.1. 特徴

AT90PWM81/161は進行中にデバイスのクロックのON/OFF切り替えを使用者に許す強力な動的クロック切り替えを提供します。組み込み整合回路がクロックに非同期での許可と禁止を許します。これは簡単且つ素早い実装のための効率的な電力管理の仕組みを許します。安全な応用では、動的クロック切り替え回路が外部クロック異常を継続的に監視するかもしれません。

AT90PWM81/161はクロックヒューズ代替用の1つのレジスタ(クロック選択レジスタ(CLKSELR))と動的クロック切り替え回路を制御するための1つのレジスタ(クロック制御/状態レジスタ(CLKCSR))を提供します。必要ならば、外部クロック元を監視するのにウォッチドッグが使われます。動的クロック切り替え回路の制御はソフトウェアによって管理されなければなりません。低位制御はCLKCSRレジスタを通してハードウェアによって実行されます。特徴は以下のとおりです。

- **安全な命令**：予期せぬ命令を避けるため、CLKCSRレジスタのビットを変更するには特別な手続きに従わなければなりません(28頁の「CLKCSR - クロック制御/状態レジスタ」をご覧ください)
- **排他的な活動**：活動は復号(命令表)によって制御されます。動的クロック切り替えの主な命令は次のとおりです。
  - 'クロック元禁止'
  - 'クロック元許可'
  - 'クロック有効性要求'
  - 'クロック元切り替え'
  - 'システムクロック元回復'
- **状態**：許可されたクロックの有効性の状態と、システムクロックを駆動するのに使われるクロック元の符号回復が提供されます。

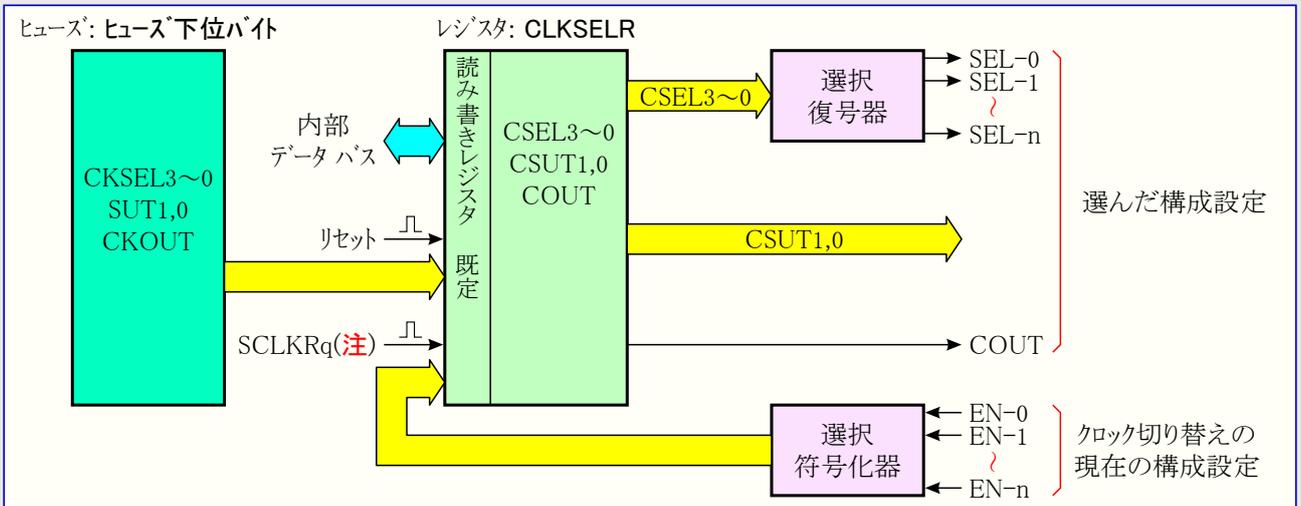
#### 5.3.2. ヒューズ代替

リセット中、ヒューズ下位バイトのビットがクロック選択レジスタ(CLKSELR)にラッチされます。このレジスタの内容はヒューズ下位バイトと同じ様に操作することができます。図5-5.で示されるようにCKSEL3~0、SUT1,0、CKOUTのヒューズが代替され、各々クロック元選択(CSEL3~0)、クロック始動時間(CSUT1,0)、クロック出力(COUT)によって置換されます。

#### 5.3.3. クロック元選択

利用可能なクロック元の符号は19頁の表5-1.で与えられます。

図5-5. ヒューズ代替とクロック元選択



注: SCLKRqはクロック制御/状態レジスタの命令

'クロック元許可/禁止'、'クロック有効性要求'または'クロック元切り替え'の命令が入力されると、CLKSELRレジスタによって提供される選んだ構成設定が各目的対象のクロック元に関してラッチされます。

'システムクロック元回復'命令はシステムクロックを駆動するのに使われたクロック元の回復を許します。そしてCLKSELRレジスタのCKSEL領域がその符号で更新されます。使われたSUTの情報やCKOUTの状態は全くありません。

選んだ構成設定がクロック元の段階でラッチされるため、与えられた時に多くのクロック元を許可することが可能です(例:システムクロック用に内部RC発振器+外部クリスタル用発振器)。使用者のソフトウェアにはこれの管理の責任があります。

'クロック有効性要求'命令はアドレス指定されたクロック元の動作指示を返します。この状態はCLKCSRレジスタのクロック準備可(CLKRDY)フラグに設定されます。

### 5.3.4. クロック元許可/禁止

'クロック元許可'命令は**クロック選択(CLKSELR)レジスタ(クロック元選択(CSEL3~0))**と**クロック始動時間(CSUT1,0)**の設定によって提供されるクロック元を選んで許可します。(CKSELとSUTのヒューズがそれを行うように)CSEL領域がクロック元を選び、CSUT領域が始動時間を選びます。クロック元が許可されたことを保証するため、'クロック元許可'命令の後に'クロック有効性要求'命令を実行するのが良いことです。

'クロック元禁止'命令はCLKSELRレジスタ(CSEL3~0のみ)の設定によって提供されるクロック元を禁止します。そのクロック元がシステムクロックを駆動するのに使われていたものなら、その命令は処理されません。

### 5.3.5. クロック有効性

'クロック有効性要求'命令は選んだクロック元(CSEL3~0)の発振計数を許可します。この計数はCSUT1,0によって提供されます。この計数完了時にそのクロックは準備可(**クロック準備可(CLKRDY)=1**)を宣言されます。このフラグは新しい計数まで無変化で留まります。CLKRDYフラグは計数開始時にリセットされます。この検査を実行するために、CSELとCSUTの領域は操作が動作中の間全てで変更すべきではありません。

以下のように2つの使用法が可能です。

#### 1. 切り替え前のクロック安定

一旦新しいクロック元が選ばれると、計数手順が動作します。使用者(コード)は切り替えを実行する前に**クロック制御/状態レジスタ(CLKCSR)**のCLKRDYフラグの設定(**1**)を待つべきです。

#### 2. 要求でのクロック有効

何時でも、使用者(コード)はクロック元の有効性を問うことができます。使用者(コード)はCLKCSRレジスタに適切な命令を書くことでそれを要求することができます。そしてクロック元の完全な状態を見ることができます。

### 5.3.6. クロック切り替え

システムクロックを駆動するために、使用者は現在のクロック元から以下の1つ(それらの1つが現在のクロック元です)に切り替えることができます。

1. 8.0/1.0MHz校正付き内蔵RC発振器
2. ウォッチドッグ用128kHz内部発振器
3. 外部クロック信号
4. 外部クリスタル発振子/セラミック振動子
5. 4分周PLL出力

クロック切り替えは命令の流れで実行されます。最初に、使用者(コード)は新しいクロック元が動作中であることを保証しなければなりません。その後に切り替え命令を入力することができます。最後に、使用者(コード)は直前のクロック元を停止することができます。一旦割り込みが禁止されている間にこの手順を動かす方がより良いでしょう。使用者(コード)にはこのクロック切り替え手順の責任があります。

これはこのような命令の手順を記述する“軽い”C言語コードです。

#### C言語プログラム例

```
void ClockSwiching (unsigned char clk-number, unsigned char sut) {

#define CLOCK-RECOVER      0x05
#define CLOCK-ENABLE      0x02
#define CLOCK-SWITCH      0x04
#define CLOCK-DISABLE     0x01

unsigned char previous-clk, temp;

// 全割り込み禁止
asm ("cli"); temp = SREG;
// "システムクロック元回復"命令
CLKCSR = 1 << CLKCCE;
CLKCSR = CLOCK-RECOVER;
previous-clk = CLKSELR & 0x0F;
// "クロック元許可"命令
CLKSELR = ((sut << 4) & 0x30) | (clk-number & 0x0F);
CLKCSR = 1 << CLKCCE;
CLKCSR = CLOCK-ENABLE;
// クロック有効性待ち
while ((CLKCSR & (1 << CLKRDY)) == 0);
// "クロック元切り替え"命令
CLKCSR = 1 << CLKCCE;
CLKCSR = CLOCK-SWITCH;
// 切り替え効果待ち
while (1) {
    CLKCSR = 1 << CLKCCE;
    CLKCSR = CLOCK-RECOVER;
    if ((CLKSELR & 0x0F) == (clk-number & 0x0F)) break;
}
// "クロック元禁止"命令
CLKSELR = previous-clk;
CLKCSR = 1 << CLKCCE;
CLKCSR = CLOCK-DISABLE;
// 全割り込み再許可
SREG = temp; asm ("sei");
}
```

#### 警告:

AT90PWM81/161では、外部クロック元の内の1つだけが与えられた時に許可することができ、両方が1つのピンを共用するために、外部クロック信号から外部用発振器への切り替えは不可能です。

また、システムクロックが4分周PLLの時にPLLの同期供給元を切り換えることは不可能です。これらの場合を識別するには19頁の表5-1をご覧ください。

校正付き8.0/1.0MHz内蔵RC発振器をアクセスするのが2つのアドレスなので、この2つの周波数間の変更はクロック切り替え機能によって許されません。この目的にはMCU制御レジスタ(MCUCR)の8/1MHz校正付きRC発振器周波数選択(CKRC81)ビットが使われなければなりません。

## 5.4. システム クロック前置分周器

### 5.4.1. 特徴

AT90PWM81/161のシステムクロックはクロック前置分周レジスタ(CLKPR)の設定によって分周することができます。この特徴(機能)は必要とされる処理能力が低い時の消費電力削減に使えます。これは全クロック種別で使え、CPUと全同期周辺機能のクロック周波数に影響を及ぼします。clkCPU、clkFLASH、clkI/O、clkADCは表5-10で示された係数によって分周されます。

## 5.4.2. 切り替え時間

前置分周器設定間を切り替えるとき、システムクロック前置分周器は中間(経過途中)の周波数が直前の設定に対応するクロック周波数または新規設定に対応するクロック周波数のどちらよりも高くなく、クロック系で不整合が起きないことを保証します。前置分周器として実行するリプル計数器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も速いかもしれません。従って例え(計数器値が)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。

## 5.5. クロック関係レジスタ

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

ビット (\$88)	7	6	5	4	3	2	1	0	
	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	デバイス固有の校正值								

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

発振校正レジスタは発振器周波数の偏差処理を省くための校正付き内蔵発振器の調整に使われます。チップのリセット中に工場で作られた値がこのレジスタへ自動的に書かれ、25°Cで8.0MHzの発振周波数を与えます。応用ソフトウェアは発振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7.6~8.4MHzの範囲で±1%内の精度でどの周波数にも校正することができます。

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

CAL7~0ビットは選ばれた範囲内で周波数を調整するのに使われます。\$00の設定はその範囲での最低周波数を与え、\$FFの設定がその範囲での最高周波数を与えます。CAL7~0の1増加は7.6~8.4MHzの周波数範囲で0,5%未満の周波数増加を与えます。

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

ビット (\$83)	7	6	5	4	3	2	1	0	
	CLKPCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	内容参照	内容参照	内容参照	内容参照	

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

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

#### • ビット6~4 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

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

これらのビットは選んだクロック元と内部システムクロック間の分周値を定義します。これらのビットは応用の必要条件に合わせた各種クロック周波数を実行時に書くことができます。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5-10で与えられます。

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

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

前置分周器設定変更時、手順が邪魔されないために、割り込みは禁止されなければなりません。

CKDIV8ヒューズがCLKPSビットの初期値を決めます。CKDIV8が非プログラム(1)にされると、CLKPSビットは'0000'にリセットされます。CKDIV8がプログラム(0)されると、CLKPSビットは起動時に8分周を与える'0011'にリセットされます。現在の動作条件でデバイスの最高周波数よりも高い周波数のクロック元を選んだ場合にこの機能が使われるべきです。CKDIV8ヒューズ設定に拘らず、どの値もCLKPSビットへ書けることに注意してください。応用ソフトウェアは現在の動作条件でデバイスの最高周波数よりも高い周波数のクロック元を選んだ場合、十分な分周値が選ばれることを保証しなければなりません。このデバイスはCKDIV8ヒューズがプログラム(0)で出荷されます。

表5-10. クロック前置分周器選択

CLKPS3	0								1								
CLKPS2	0				1				0				1				
CLKPS1	0		1		0		1		0		1		0		1		
CLKPS0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	
分周値(数)	1	2	4	8	16	32	64	128	256	(予約)							

### 5.5.3. PLLCSR – PLL制御/状態レジスタ (PLL Control and Status Register)

ビット (\$87)	7	6	5	4	3	2	1	0	PLLCSR
	-	-	PLL3	PLL2	PLL1	PLL0	PLLE	PLOCK	
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R	
初期値	0	0	0	1	0	0	1/0	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5~2 – PLL3~0 : PLL周波数係数 (PLL Factor)

PLLビットはPLLの通倍係数選択に使われます。

表5-11. PLL通倍係数

PLL3~0	n+2	PLL周波数 (MHz)
7~15		(予約)
6	8	64
5	7	56
4	6	48
3	5	40
2	4	32
0~1		(予約)

注: PLL3はデバッグ目的に使われます(配線されなければなりません)。

- ビット1 – PLLE : PLL許可 (PLL Enable)

PLLEが設定(1)されると、PLLが始動され、未だ始動されていないなら、PLL基準クロックとして内蔵RC発振器が始動されます。システムクロック元としてPLLが選ばれると、このビットに対する値は常に1です。

- ビット0 – PLOCK : PLL固定化検出 (PLL Lock Detector)

PLOCKビットが設定(1)されると、PLLが基準クロックに固定化され、電力段階制御器(PSC)に対するCLKPLLの許可は安全です。固定化に対する時間は22頁の表5-9.で指定されます。

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

ビット (\$35 (\$55))	7	6	5	4	3	2	1	0	MCUCR
	-	-	-	PUD	RSTDIS	CKRC81	IVSEL	IVCE	
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0/1(注)	0	0	

注: 値はCKSEL3~0ヒューズで初期化されます(CKSEL3~0=0110時は1、他の全ての場合は0)。

- ビット2 – CKRC81 : 8/1MHz校正付きRC発振器周波数選択 (Frequency Selection of the calibrated 8/1 MHz RC Oscillator)

MCUCRのCKRC81によって校正付きRC発振器の代表周波数が変更されます。

- CKRC81ビットが0を書かれると、RC発振器は8MHzです。
- CKRC81ビットが1を書かれると、RC発振器は1MHzです。

注: このビットはRC発振器が許可されている時にだけ変更することができます。

注: RC発振器がPLL供給元として使われている時にCKRC81は1を書かれてはなりません。

注: RC発振器が禁止された場合、このビットはハードウェアによって解除(0)されます。

### 5.5.5. CLKCSR – クロック制御/状態レジスタ (Clock Control & Status Register)

ビット (\$84)	7	6	5	4	3	2	1	0	CLKCSR
	CLKCCE	-	-	CLKRDY	CLKC3	CLKC2	CLKC1	CLKC0	
Read/Write	R/W	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 – CLKCCE : クロック制御変更許可 (Clock Control Change Enable)

CLKCSRのビットの変更を許可するにはCLKCCEビットが論理1を書かれなければなりません。CLKCCEビットはCLKCSRの他のビットが同時に0を書かれる時にだけ更新されます。CLKCCEはそれが書かれた4周期後、またはCLKCSRのビットが書かれた時にハードウェアによって解除(0)されます。この制限時間内のCLKCCEビットの再書き込みは制限時間の延長もCLKCCEビットの解除(0)もどちらも行いません。

#### • ビット6,5 – 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 – CLRDRDY : クロック準備可フラグ (Clock Ready Flag)

このフラグは'クロック有効性'論理回路の出力です。このフラグは一度'クロック有効性要求'命令が入力されると、リセットされます。

'クロック有効性'論理回路が(選んだ)クロックが動いて安定であることを通知する時にこれが設定(1)されます。要求からの遅延とフラグ設定は一定ではなく、クロック始動時間、クロック周波数、そして勿論クロックが生きているかに依存します。「クロック信号なし」と「クロック信号が未だ有効ではない」との違いは使用者が行わなければなりません。

#### • ビット3~0 – CLKC3~0 : クロック制御 (Clock Control Bits 3~0)

これらのビットは'クロック切り替え'部へ提供する命令を定義します。CLKCビットを変更するには特別な書き込み手順に従わなければなりません(「ビット7 – CLKCCE : クロック制御変更許可」をご覧ください)。

1. クロック制御変更許可(CLKCCE)ビットに1を、そしてCLKCSRの他の全てのビットに0を書いてください。
2. 4周期内に、CLKCSRレジスタに望む値を書くと同時にCLKCCEビットを解除(0)してください。

この手順を邪魔しないために、CLKCSRレジスタ設定時に割り込みは禁止されるべきです。

表5-12. クロック命令一覧

CLKC3~0	クロック命令	CLKC3~0	クロック命令
0 0 0 0	(命令なし)	0 1 0 0	クロック元切り替え
0 0 0 1	クロック元禁止	0 1 0 1	システム クロック元回復
0 0 1 0	クロック元許可	0 1 1 1	CKOUT命令
0 0 1 1	クロック有効性要求	1 x x x	(命令なし)

### 5.5.6. CLKSELR – クロック選択レジスタ (Clock Selection Register)

ビット (\$85)	7	6	5	4	3	2	1	0	CLKSELR
	-	COUT	CSUT1	CSUT0	CSEL3	CSEL2	CSEL1	CSEL0	
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	CKOUTヒューズ*	SUT1,0ヒューズ*					CKSEL3~0ヒューズ*	

#### • ビット7 – Res : 予約 (Reserved)

このビットは予約されており、常に0として読みます。

#### • ビット6 – COUT : クロック出力 (Clock Out)

COUTビットはCKOUTヒューズ\* ビットで初期化されます。COUTビットは'CKOUT'命令の場合にだけ使われます。使用については22頁の「5.2.7. クロック出力緩衝部」項を参照してください。'システム クロック元回復'命令の場合、COUTは影響を及ぼされません(この設定の回復なし)。

#### • ビット5,4 – CSUT1,0 : クロック始動時間 (Clock Start-up Time)

CSUTビットはSUTヒューズ\* ビットの値で初期化されます。'クロック元許可/禁止'命令の場合、CSUT領域はクロック始動時間の符号を提供します。クロック始動時間の符号については19頁の「5.2. クロック元」項の一部分を参照してください。'システム クロック元回復'命令の場合、CSUT領域は影響を及ぼされません(SUT符号の回復なし)。

#### • ビット3~0 – CSEL3~0 : クロック元選択 (Clock Source Select)

CSELビットはCKSELヒューズ\* ビットの値で初期化されます。'クロック元許可/禁止'、'クロック有効性要求'、または'クロック元切り替え'の命令の場合、CSEL領域はそのクロック元の符号を取り戻します。クロック元符号については19頁の表5-1と19頁の「5.2. クロック元」項の一部分を参照してください。'システム クロック元回復'命令の場合、CSEL領域は18頁の図5-1.で記述されるようにクロック制御部を駆動するのに使われたクロック元の符号を受け取ります。

## 6. 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。

### 6.1. 休止形態種別

18頁の図5-1はAT90PWM81/161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選択を助けます。表6-1は異なる休止形態とそれらの起動元を示します。

表6-1. 各休止動作に於ける動作クロック範囲と復帰起動要因

休止種別	動作クロック範囲					発振器 主クロック 供給元	復帰起動要因 (割り込み・リセット)					
	clk CPU	clk FLASH	clk IO	clk ADC	clk PLL		INT 0~2	電力段 制御器 (PSC)	SPM EEPROM 操作可	A/D 変換 完了	ウォッチ ドッグ	その他 I/O
アイドル			○	○	○	○	○	○	○	○	○	○
A/D変換雑音低減				○	○	○	②	○	○	○	○	
パワーダウン							②	○			○	
スタンバイ(注1)						○	②				○	

注1: クロック元として外部クリスタル発振子またはセラミック振動子が選ばれた場合です。

② INT0~2のレベル割り込みだけです。

4つの休止形態の何れかへ移行するには**休止形態制御レジスタ(SMCR)**の**休止許可(SE)ビット**が論理1を書かれ、**SLEEP**命令が実行されなければなりません。SMCRの**休止種別選択(SM2~0)ビット**は**SLEEP**命令によって活性(有効)にされる休止形態(アイドル、A/D変換雑音低減、パワーダウン、スタンバイ)のどれかを選びます。一覧については表6-2をご覧ください。

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

### 6.2. 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、SPI、アナログ比較器、A/D変換器、タイマ/カウンタ、ウォッチドッグ、割り込み機構の継続動作を許します。この休止形態は基本的にclkCPUとclkFLASHを停止する一方、他のクロックに走行を許します。

アイドル動作はMCUにタイマ溢れなどの内部割り込みだけでなく、外部で起動された割り込みからの起動も可能にします。アナログ比較器割り込みからの起動が必要とされないなら、**アナログ比較器n制御(ACnCON)レジスタのアナログ比較器n許可(ACnEN)ビット**を解除(0)することによってアナログ比較器を電源断にできます。これはアイドル動作での消費電力を削減します。A/D変換が許可されるなら、この動作に移行すると変換が自動的に始まります。

### 6.3. 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、CPUを停止しますが、A/D変換器、外部割り込み、(クロック元が外部(T0またはT1)ならば)タイマ/カウンタ、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkI/O、clkCPU、clkFLASHを停止する一方、他のクロックに走行を許します。

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

### 6.4. パワーダウン動作

SM2~0ビットが'010'を書かれると、**SLEEP**命令はMCUをパワーダウン動作へ移行させます。この動作では外部発振器が停止される一方で、外部割り込み、ウォッチドッグ機能は(許可ならば)継続して動作します。外部リセット、ウォッチドッグリセット、低電圧検出(BOD)リセット、INT0~2の外部レベル割り込みだけがMCUを起動できます。この休止形態は基本的に生成した全てのクロックを停止し、非同期部の動作だけを許します。

レベルで起動した割り込みがパワーダウン動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。詳細については52頁の「外部割り込み」を参照してください。

パワーダウン動作から起動するとき、起動条件が起きてから起動の効果が現れるまで遅延があります。これは停止されてしまっている後の再始動と安定になることをクロックに許します。この起動(遅延)時間は19頁の「クロック元」で記述されるように、リセット付加遅延時間を定義するのと同じCKSELヒューズによって定義されます。

## 6.5. スタンバイ動作

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

## 6.6. 電力削減 (レジスタ)

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

クロック停止中や停止後、クロック開始中の周辺機能の予測可能な全動作は保証されません。故に電力削減レジスタ(PRR)でクロックを停止前に周辺機能部を停止することが推奨されます。

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。

## 6.7. 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時、特別な考慮を必要とするでしょう。

### 6.7.1. A/D変換器 (ADC)

許可したなら、A/D変換器は全休止形態で許可されます。節電するため、休止形態の何れかへ移行する前にA/D変換器は禁止されるべきです。A/D変換器がOFFそして再びONに切り替えられると、次の(最初の)変換は延長された(初回)変換になります。A/D変換器操作の詳細については126頁の「**雑音低減機能**」を参照してください。

### 6.7.2. アナログ比較器

アイドル動作へ移行するとき、アナログ比較器は使われないなら、禁止されるべきです。A/D変換雑音削減動作へ移行するとき、アナログ比較器は禁止されるべきです。他の休止形態で、アナログ比較器は自動的に禁止されず、使わないなら、禁止されるべきです。とは言え、アナログ比較器が入力として内部基準電圧を使う設定の場合、全休止形態でアナログ比較器は禁止されるべきです。さもなければ内部基準電圧は休止形態と無関係に許可されます。アナログ比較器設定法の詳細については116頁の「**アナログ比較器**」を参照してください。

### 6.7.3. 低電圧検出器 (BOD)

低電圧検出器(BOD)が応用で必要とされないなら、この単位部はOFFにされるべきです。低電圧検出器が**BODLEVELヒューズ**によって許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。低電圧検出器(BOD)設定法の詳細については33頁の「**低電圧検出 (BOD)**」を参照してください。

### 6.7.4. 内部基準電圧

内部基準電圧は低電圧検出器(BOD)、アナログ比較器、A/D変換器によって必要とされる時に許可されます。これら単位部が上の項目で記述されたように禁止されると、内部基準電圧は禁止され、電力を消費しません。再び許可する場合、この出力が使われる前に使用者は基準電圧へ起動(安定時間)を与えなければなりません。基準電圧が休止形態でON保持される場合、この出力は直ちに使えます。起動時間の詳細については34頁の「**内部基準電圧**」を参照してください。

### 6.7.5. ウォッチドッグ タイマ

ウォッチドッグ タイマが応用で必要とされないなら、この単位部はOFFにされるべきです。ウォッチドッグ タイマが許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。ウォッチドッグ タイマ設定法の詳細については35頁の「**ウォッチドッグ タイマ**」を参照してください。

### 6.7.6. ポートピン

休止形態へ移行するとき、全てのポートピンは最小電力使用に設定されるべきです。最も重要なことはその時にピンが抵抗性負荷を駆動しないのを保証することです。I/Oクロック( $clk_{I/O}$ )とA/D変換クロック( $clk_{ADC}$ )の両方が停止される休止形態ではデバイスの入力緩衝部が禁止されます。これは必要とされない時に入力論理回路によって電力が消費されないことを保証します。いくつかの場合で入力論理回路は起動条件を検出するために必要とされ、その時は許可されます。どのピンが許可されるかの詳細については44頁の「**デジタル入力許可と休止形態**」を参照してください。入力緩衝部が許可され、入力信号が浮いている状態のままか、またはアナログ信号電圧がVCC/2付近の場合、入力緩衝部は過大な電力を消費するでしょう。

アナログ入力ピンに対するデジタル入力緩衝部は常に禁止されるべきです。入力ピンでのVCC/2付近のアナログ信号入力は活動動作でも重要な電流を引き起こし得ます。デジタル入力緩衝部はデジタル入力禁止レジスタ(DIDR0とDIDR1)の書き込みによって禁止できます。詳細については121頁と131頁の「**DIDR0 - デジタル入力禁止レジスタ0**」、「**DIDR1 - デジタル入力禁止レジスタ1**」を参照してください。

## 6.7.7. 内蔵デバッグ機能 (dW)

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

## 6.8. 電力管理用レジスタ

### 6.8.1. 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6-2.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6-2. 休止動作種別選択

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

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

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

ビット	7	6	5	4	3	2	1	0	
(\$86)	PRPSC2	-	PRPSCR	PRTIM1	-	PRSPI	-	PRADC	PRR
Read/Write	R/W	R	R/W	R/W	R	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7 – PRPSC2 : 電力段制御器(PSC)2電力削減 (Power Reduction PSC2)

このビットへの1書き込みはその部分へのクロック停止によって電力段制御器(PSC)2の消費を削減します。PSC2の再起動時、PSC2は正しい動作を保証するために再初期化されるべきです。

- ビット6,3,1 – Res : 予約 (Reserved)

これらのビットは予約されています。

- ビット5 – PRPSCR : 縮小電力段制御器(PSCR)電力削減 (Power Reduction PSC Reduced)

このビットへの1書き込みはその部分へのクロック停止によって縮小電力段制御器(PSCR)の消費を削減します。PSCRの再起動時、PSCR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4 – 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部の消費を削減します。タイマ/カウンタ1が許可されると、本ビット設定(=1)前のように動作は続きます。

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

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

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

このビットへの1書き込みはその部分へのクロック停止によってA/D変換器(ADC)の消費を削減します。A/D変換器は本機能使用前に禁止されなければなりません。



## 7. システム制御とリセット

### 7.1. システム制御概要

#### 7.1.1. AVRのリセット

リセット中、全てのI/Oレジスタはそれらの初期値に設定され、プログラムはリセットベクタから実行を開始します。リセットベクタに配置される命令は、きっとリセット処理ルーチンへのRJMP(相対分岐)命令でしょう。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが配置できます。これはリセットベクタが応用領域の一方、割り込みベクタがポート領域の場合やその逆も同様です。図7-1の回路構成図はリセット論理回路を示します。表7-1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た後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電源に許します。遅延タイマの遅延時間はSUTヒューズとCKSELヒューズを通して使用者によって定義されます。この遅延時間についての各種選択は19頁の「クロック元」で示されます。

#### 7.1.2. リセット元

AT90PWM81/161には次の4つのリセット元があります。

- **電源ONリセット** …… 供給電圧が電源ONリセット閾値電圧(V<sub>POT</sub>)以下でMCUがリセットされます。
- **外部リセット** …… RESETピンが最小パルス幅以上Lowレベルに保たれると、MCUがリセットされます。外部リセットピンは次の2つの方法で禁止することができます。
  - RSTDISBLヒューズによる。この場合はSPIプログラミングが禁止されます。
  - MCU制御レジスタ(MCUCR)のリセットピン禁止(RSTDIS)ビットを使うソフトウェアによる。この場合は通電時にSPIプログラミングが未だ活性(有効)です。
- **ウォッチドッグリセット** …… ウォッチドッグシステムリセット動作が許可され、ウォッチドッグタイマが終了すると、MCUがリセットされます。
- **低電圧リセット** …… 低電圧検出が許可され、供給電圧(V<sub>CC</sub>)が低電圧検出電圧(V<sub>BOT</sub>)以下でMCUがリセットされます。

図7-1. リセット回路構成

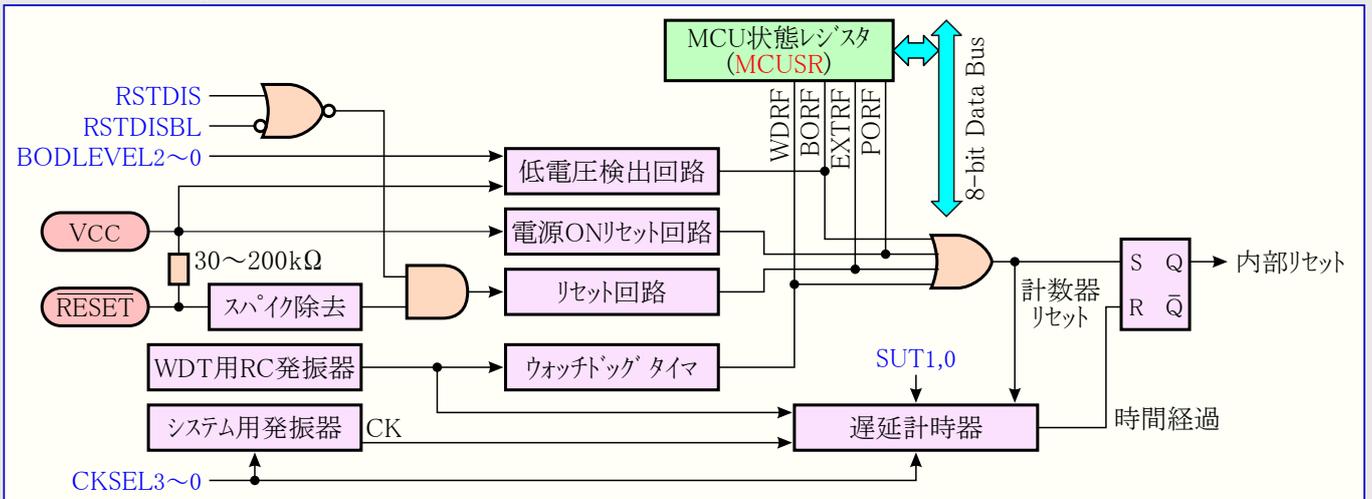


表7-1. リセット電気的特性

シンボル	項目	条件	最小	代表	最大	単位
V <sub>POT</sub>	上昇時電源ONリセット閾値電圧		-	1.4	2.3	V
	下降時電源ONリセット閾値電圧 (注1)		-	1.3	2.3	
V <sub>RST</sub>	RESETピン閾値電圧		0.2V <sub>CC</sub>		0.85V <sub>CC</sub>	
t <sub>RST</sub>	RESETピンでの最小パルス幅		400			ns

注1: 供給電圧がこの電圧以下にならないと、上昇時の電源ONリセットは動作しません。

注: 値は指針の意味だけで、実際の値はTBDです。

### 7.1.3. 電源ONリセット

電源ONリセット(POR)パルスはチップ上の検出回路によって生成されます。検出電圧は表7-1.で定義されます。POR信号はVCCが検出電圧以下の時は必ず活性(有効)にされます。POR回路は供給電圧異常検出は勿論、始動リセットの起動にも使えます。

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

図7-2. 内蔵電源ONリセット (RESETはVCCに接続)

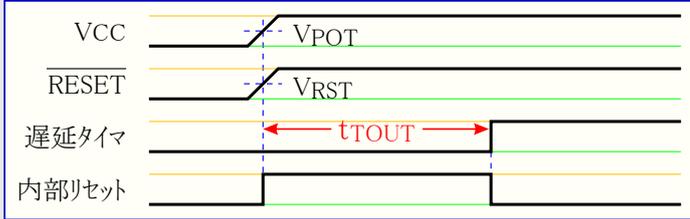
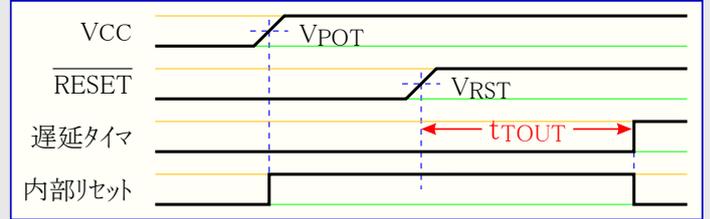


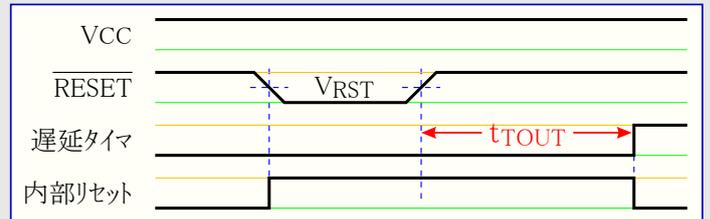
図7-3. 外部RESET信号による延長電源ONリセット



### 7.1.4. 外部リセット

外部リセットはRESETピンのLowレベルによって生成されます。クロックが動いていなくても、最小パルス幅(表7-1.参照)以上のリセットパルスはリセットを生成します。短すぎるパルスはリセット生成が保証されません。印加された信号の上昇がリセット閾値電圧( $V_{RST}$ )に達すると(遅延タイマを起動し)、遅延タイマは遅延時間( $t_{TOUT}$ )経過後にMCUを始動します。

図7-4. 動作中の外部リセット



### 7.1.5. 低電圧(ブラウンアウト)検出リセット

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

VCCが起動電圧以下に下がると(図7-5.の $V_{BOT-}$ )、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上がると(図7-5.の $V_{BOT+}$ )、(遅延タイマが起動され)遅延タイマは遅延時間( $t_{TOUT}$ )経過後にMCUを始動します。

BOD回路は電圧が表7-3.で与えられる $t_{BOD}$ 時間より長く起動電圧以下に留まる場合のみ、VCCでの低下を検出します。

図7-5. 動作中の低電圧検出リセット

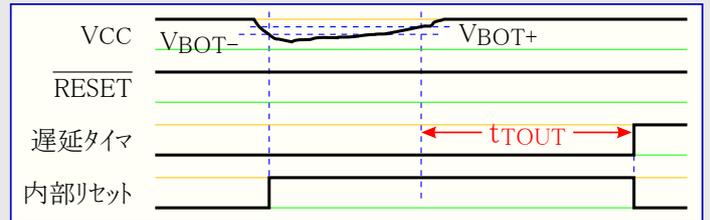


表7-3. 低電圧検出(BOD)特性

シンボル	項目	最小	代表	最大	単位
$V_{HYST}$	低電圧検出ヒステリシス電圧		70		mV
$t_{BOD}$	最小低電圧検出時間		2		$\mu$ s

表7-2. BODLEVELヒューズ ( $V_{BOT}$ ) 設定 (注1)

BODLEVEL2~0	000	001	010	011	100	101	110	111
最大					4.6			禁止、BODは許可されなければなりません。
代表	2.0	1.9	2.2	2.3	4.3	2.7	1.8	
最小					3.9			
単位	V							

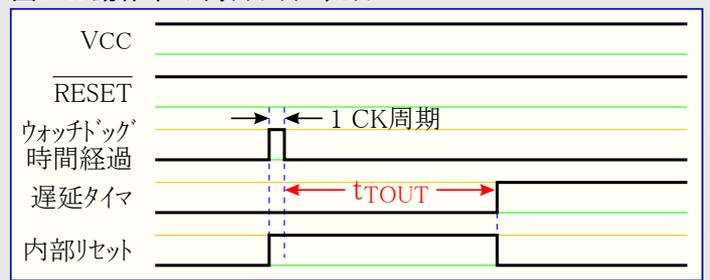
注: 値は指針の意味だけです。

注1: いくつかのデバイスで $V_{BOT}$ が公称最低動作電圧以下の可能性があります。この状態のデバイスについては製造検査中、 $VCC = V_{BOT}$ に落として検査されています。これはマイクロコントローラの正しい動作がもはや保証されない電圧になる前に、低電圧検出(BOD)リセットが起きることを保証します。この検査は高動作電圧に対してBODLEVEL = 010、低動作電圧に対してBODLEVEL = 101を使って実行されます。

### 7.1.6. ウォッチドッグリセット

ウォッチドッグ時間経過時、(内部的に)1CK周期幅の短いリセットパルスを生成します。このパルスの下降端で遅延タイマは遅延時間( $t_{TOUT}$ )の計時を始めます。ウォッチドッグタイマ操作の詳細については35頁を参照してください。

図7-6. 動作中のウォッチドッグリセット



## 7.2. システム制御用レジスタ

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

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

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

- **ビット7~4 – Res : 予約 (Reserved)**

これらのビットは予約されており、常に0として読みます。

- **ビット3 – WDRF : ウォッチドッグ リセット フラグ (Watchdog Reset Flag)**

このビットはウォッチドッグ リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

- **ビット2 – BORF : 低電圧リセット フラグ (Brown-Out Reset Flag)**

このビットは低電圧リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

- **ビット1 – EXTRF : 外部リセット フラグ (External Reset Flag)**

このビットは外部リセットが起こると設定(1)されます。このビットは電源ONリセットまたは、このフラグへの論理0書き込みによってリセット(0)されます。

- **ビット0 – PORF : 電源ONリセット フラグ (Power-on Reset Flag)**

このビットは電源ONリセットが起こると設定(1)されます。このビットはこのフラグへの論理0書き込みによってのみリセット(0)されます。

リセット条件の確認にリセット フラグを使うため、使用者はプログラム内で可能な限り早くMCUSRを読み、そして解除(0)すべきです。別のリセットが起こる前にこのレジスタが解除(0)されると、そのリセット元はリセット フラグを調べることによって得られます。

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	–	–	–	PUD	RSTDIS	CKRC81	IVSEL	IVCE	MCUCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0/1(注)	0	0	

注: 値はCKSEL3~0ヒューズで初期化されます(CKSEL3~0=0110時は1、他の全ての場合は0)。

- **ビット3 – RSTDIS : リセットピン禁止 (Reset Pin Disable)**

MCUCRのRSTDISによってリセット機能を禁止してこのピンを目的機能用にすることができます。

- RSTDISビットが0を書かれると、リセット信号は活性(有効)です。
- RSTDISビットが1を書かれると、リセット信号は無効です。

## 7.3. 内部基準電圧

AT90PWM81/161は内部バンドギャップ基準電圧が特徴です。このバンドギャップ基準電圧は低電圧検出(BOD)に使われ、アナログ比較器やA/D変換器に対するアナログ入力としても使うことができます。

D/A変換器やA/D変換器、それと比較器用の内部基準電圧はこのバンドギャップ電圧から配給されます。113頁の「チップ上基準電圧と温度感知器概要」をご覧ください。

Vref電圧はADMUXレジスタの基準電圧選択(REFS1,0)ビットによって構成設定されます(129頁の「ADMUX – A/D変換多重器選択レジスタ」をご覧ください)。

### 7.3.1. バンドギャップと内部基準電圧許可信号と起動時間

バンドギャップと内部基準電圧の特性は表7-4.で与えられます。節電のために、この基準電圧は常にONではありません。バンドギャップと内部基準電圧は次の状態中ONです。

1. 低電圧検出リセット許可時 (BODLEVELヒューズのプログラム(0)により)
2. アナログ比較器基準電圧接続時 (A/D変換多重器選択(ADMUX)レジスタの基準電圧選択(REFS1)=1)
3. アナログ比較器へのバンドギャップ基準電圧接続時
4. A/D変換部動作許可時 (A/D変換制御/状態レジスタ(ADCSRA)のA/D動作許可(ADEN)=1)

従って低電圧検出(BOD)が許可されていないと、A/D変換器、アナログ比較器、または内部基準電圧の許可後、使用者はアナログ比較器またはA/D変換器が使われる前に、基準電圧へ起動時間を与えなければなりません。パワーダウン動作での消費電力を減らすため、使用者はパワーダウン動作へ移行する前に基準電圧がOFFされるのを保証することで上の4つの状態を避けられます。

## 7.3.2. 内部基準電圧特性

表7-4. 内部基準電圧特性

シンボル	項目	条件	最小	代表	最大	単位
$V_{BG}$	基準電圧	-	-	1.1	-	V
$t_{BG}$	起動時間		-	40	-	$\mu$ s
$I_{BG}$	消費電流		-	15	-	$\mu$ A

注: これらの値は目安だけです。

## 7.4. ウォッチドッグ タイマ

AT90PWM81/161には強化したウォッチドッグ タイマ(WDT)があります。主な機能を次に示します。

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

ウォッチドッグ タイマ(WDT)はチップ上の独立した128kHz発振器の間隔で計時するタイマです。WDTはカウンタ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 リセットを与えます。通常動作では計時完了値へ達する前にカウンタを再始動するために、システムがウォッチドッグ リセット(WDR)命令を使う必要があります。システムがカウンタを再始動しなければ、割り込みまたはシステム リセットが起こるでしょう。

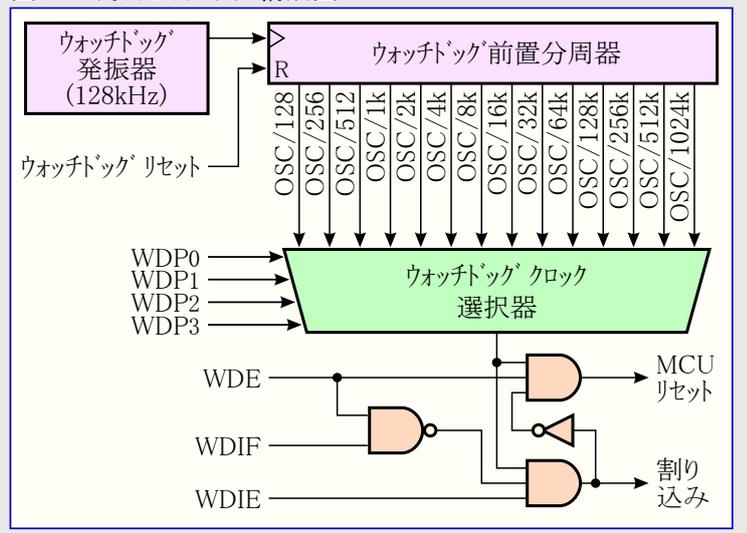
割り込み動作種別では、タイマ計時完了時にWDTが割り込みを生じます。この割り込みは休止形態からデバイスを起動するためや、一般的なシステム タイマとしても使えます。1つの例は或る動作に対して許された最大時間を制限することで、その動作が予測されたより長く走行する時に割り込みを生じます。システム リセット動作種別ではタイマ計時完了時にWDTがリセットを生じます。これは一般的にコード外走行の場合の中断を防止するのに使われます。3つ目の動作種別は先に割り込みを生じ、その後にシステム リセット動作種別に切り替えることで、他の2つの動作種別の組み合わせとなる、割り込み及びシステム リセット動作種別です。この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

ウォッチドッグ常時ON(WDTON)ヒューズのプログラム(0)はウォッチドッグ タイマをシステム リセット動作種別に強制します。このヒューズのプログラム(0)でシステム リセット動作(WDE)ビットと割り込み動作(WDIE)ビットは各々、'1'と'0'に固定されます。

更にプログラム保護を保証するためにウォッチドッグ設定の変更は時間制限手順に従わなければなりません。システム リセット許可(WDE)の解除と計時完了時間設定の変更についての手順は次のとおりです。

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

図7-7. ウォッチドッグ タイマ構成図



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

#### アセンブリ言語プログラム例

```

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

#### C言語プログラム例

```

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

**注:** このコード例はデバイス固有ヘッダ ファイルがインクルードされていると仮定します。

**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:** このコード例はデバイス固有ヘッダ ファイルがインクルードされていると仮定します。

**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。

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

ビット (\$82)	7	6	5	4	3	2	1	0	
	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	WDTCSR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	不定	0	0	0	

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

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

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

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

WDEが設定(1)されると、ウォッチドッグ タイマは割り込み及びシステムリセット動作種別になります。ウォッチドッグ タイマでの最初の計時完了がウォッチドッグ 割り込み要求(WDIF)フラグを設定(1)します。対応する割り込みベクタの実行はハードウェアによってWDIEとWDIFを自動的に解除(0)します。これは割り込みを使う間のウォッチドッグ リセット保護を維持するのに有用です。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7-5. 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=プログラム、1=非プログラムです。

(訳注)

左表のWDTONヒューズ値は願書と逆になっていますが、原書の値では本文中の説明と矛盾するため、本文説明に合せて統一修正しました。

### • 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7-6.で示されます。

表7-6. ウォッチドッグ 前置分周選択

WDP3	0				1				1						
	0		1		0		1		0		1				
WDP2	0		1		0		1		0		1				
WDP1	0	1	0	1	0	1	0	1	0	1	0	1			
WDP0	0	1	0	1	0	1	0	1	0	1	0	1			
WDT発振周期数	2k	4k	8k	16k	32k	64k	128k	256k	512k	1024k	1k	512	256	128	(予約)
代表的計時完了周期 (VCC=5V)	16ms	32ms	64ms	0.125s	0.25s	0.5s	1.0s	2.0s	4.0s	8.0s	8ms	4ms	2ms	1ms	

## 8. 割り込み

この項はAtmel AT90PWM81/161で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。

### 8.1. AT90PWM81/161の割り込みベクタ

表8-1. 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
	AT90PWM81	AT90PWM161		
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	電力段制御器2 PSC2_CAPT	電力段制御器2 捕獲発生/同期異常
3	\$0002	\$0004	電力段制御器2 PSC2_EC	電力段制御器2 周期終了(比較一致)
4	\$0003	\$0006	電力段制御器2 PSC2_EEC	電力段制御器2 拡張周期終了(比較一致)
5	\$0004	\$0008	縮小電力段制御器 PSCr_CAPT	縮小電力段制御器 捕獲発生/同期異常
6	\$0005	\$000A	縮小電力段制御器 PSCr_EC	縮小電力段制御器 周期終了(比較一致)
7	\$0006	\$000C	縮小電力段制御器 PSCr_EEC	縮小電力段制御器 拡張周期終了(比較一致)
8	\$0007	\$000E	アナログ比較器0 ANA_COMP0	アナログ比較器0出力遷移
9	\$0008	\$0010	アナログ比較器1 ANA_COMP1	アナログ比較器1出力遷移
10	\$0009	\$0012	アナログ比較器2 ANA_COMP2	アナログ比較器2出力遷移
11	\$000A	\$0014	INT0	外部割り込み要求0
12	\$000B	\$0016	タイマ/カウンタ1 CAPT1	タイマ/カウンタ1捕獲発生
13	\$000C	\$0018	タイマ/カウンタ1 OVF1	タイマ/カウンタ1溢れ
14	\$000D	\$001A	A/D変換器 ADC	A/D変換完了
15	\$000E	\$001C	INT1	外部割り込み要求1
16	\$000F	\$001E	SPI STC	SPI 転送完了
17	\$0010	\$0020	INT2	外部割り込み要求2
18	\$0011	\$0022	ウォッチドッグ WDT	ウォッチドッグ計時完了
19	\$0012	\$0024	EEPROM EE_RDY	EEPROM EE_RDY
20	\$0013	\$0026	SPM命令 SPM_RDY	SPM命令操作可

**注1:** BOOTRSTヒューズがプログラム(0)されると、デバイスはリセットでブート ローダ アドレスへ飛びます。139頁の「ブート ローダ支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

**注2:** MCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)されると、割り込みベクタはブートフラッシュ領域先頭(部)へ移動されます。そして各割り込みベクタのアドレスは、この表のアドレスがブートフラッシュ領域の先頭アドレスに加算されます。

表8-2.はBOOTRST(ヒューズ)とIVSEL(割り込みベクタ選択ビット)の様々な組み合わせに対するリセットと割り込みベクタの配置を示します。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが置けます。これはリセットベクタが応用領域の一方、割り込みベクタがブート領域、またはその逆の場合でも同様です。

表8-2. リセットと割り込みベクタの配置

BOOTRST	IVSEL	リセットベクタアドレス	割り込みベクタ先頭アドレス	
			AT90PWM81	AT90PWM161
非プログラム(1)	0	\$0000	\$0001	\$0002
	1	\$0000	ブート領域先頭アドレス+\$0001	ブート領域先頭アドレス+\$0002
プログラム(0)	0	ブート領域先頭アドレス	\$0001	\$0002
	1	ブート領域先頭アドレス	ブート領域先頭アドレス+\$0001	ブート領域先頭アドレス+\$0002

**注:** ブート領域先頭アドレスは148頁の表20-7.で示されます。

AT90PWM81/161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	PSC2_CAPT ; 電力段制御器2 捕獲発生/同期異常
\$0002/\$0004		RJMP/JMP	PSC2_EC ; 電力段制御器2 周期終了(比較一致)
\$0003/\$0006		RJMP/JMP	PSC2_EEC ; 電力段制御器2 拡張周期終了(比較一致)
\$0004/\$0008		RJMP/JMP	PSCr_CAPT ; 縮小電力段制御器 捕獲発生/同期異常
\$0005/\$000A		RJMP/JMP	PSCr_EC ; 縮小電力段制御器 周期終了(比較一致)
\$0006/\$000C		RJMP/JMP	PSCr_EEC ; 縮小電力段制御器 拡張周期終了(比較一致)
\$0007/\$000E		RJMP/JMP	ANA_COMP0 ; アナログ比較器0出力遷移
\$0008/\$0010		RJMP/JMP	ANA_COMP1 ; アナログ比較器1出力遷移
\$0009/\$0012		RJMP/JMP	ANA_COMP2 ; アナログ比較器2出力遷移
\$000A/\$0014		RJMP/JMP	EXT_INT0 ; 外部割り込み要求0
\$000B/\$0016		RJMP/JMP	TIM1_CAPT ; タイマ/カウンタ1捕獲発生
\$000C/\$0018		RJMP/JMP	TIM1_OVF ; タイマ/カウンタ1溢れ
\$000D/\$001A		RJMP/JMP	ADC ; ADC変換完了
\$000E/\$001C		RJMP/JMP	EXT_INT1 ; 外部割り込み要求1
\$000F/\$001E		RJMP/JMP	SPI_STC ; SPI転送完了
\$0010/\$0020		RJMP/JMP	EXT_INT2 ; 外部割り込み要求2
\$0011/\$0022		RJMP/JMP	WDT_OVF ; ウォッチドッグ計時完了
\$0012/\$0024		RJMP/JMP	EE_RDY ; EEPROM操作可
\$0013/\$0026		RJMP/JMP	SPM_RDY ; SPM命令操作可
;			
\$0014/\$0028	RESET:	LDI	R16, HIGH (RAMEND) ; RAM最終アドレス上位を取得
\$0015/\$0029		OUT	SPH, R16 ; スタック ポインタ上位を初期化
\$0016/\$002A		LDI	R16, LOW (RAMEND) ; RAM最終アドレス下位を取得
\$0017/\$002B		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

AT90PWM81/161でBOOTRSTヒューズが非プログラム(1)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000/\$0000	RESET:	LDI	R16, HIGH (RAMEND) ; 各種リセット (BOOTRSTヒューズ=1)
\$0001/\$0001		OUT	SPH, R16 ; スタック ポインタ上位を初期化
\$0002/\$0002		LDI	R16, LOW (RAMEND) ; RAM最終アドレス下位を取得
\$0003/\$0003		OUT	SPL, R16 ; スタック ポインタ下位を初期化
		}	; 以下、I/O初期化など
		. ORG	\$0C01/\$1C02 ; ブートプログラム領域が2Kバイトの場合
\$0C01/\$1C02		RJMP/JMP	PSC2_CAPT ; 電力段制御器2 捕獲発生/同期異常
\$0C02/\$1C04		RJMP/JMP	PSC2_EC ; 電力段制御器2 周期終了(比較一致)
		}	
\$0C13/\$1C26		RJMP/JMP	SPM_RDY ; SPM命令操作可

AT90PWM81/161でBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定される時の最も代表的且つ一般的なりセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令		注釈
		.ORG	\$0001/%0002	;割り込みベクタ先頭
\$0001/\$0002		RJMP/JMP	PSC2_CAPT	;電力段制御器2 捕獲発生/同期異常
\$0002/\$0004		RJMP/JMP	PSC2_EC	;電力段制御器2 周期終了(比較一致)
		}		
\$0013/\$0026		RJMP/JMP	SPM_RDY	;SPM命令操作可 ;以下、プログラムなど
		}		
		.ORG	\$0C00/\$1C00	;ブートプログラム領域が2Kバイトの場合
				;各種リセット (BOOTRSTヒューズ=0)
\$0C00/\$1C00	RESET:	LDI	R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (プログラム開始)
\$0C01/\$1C01		OUT	SPH, R16	;スタックポインタ上位を初期化
\$0C02/\$1C02		LDI	R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0C03/\$1C03		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

AT90PWM81/161でBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なりセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令		注釈
		.ORG	\$0C00/\$1C00	;ブートプログラム領域が2Kバイトの場合
\$0C00/\$1C00		RJMP/JMP	RESET	;各種リセット (BOOTRSTヒューズ=0)
\$0C01/\$1C02		RJMP/JMP	PSC2_CAPT	;電力段制御器2 捕獲発生/同期異常
\$0C02/\$1C04		RJMP/JMP	PSC2_EC	;電力段制御器2 周期終了(比較一致)
		}		
\$0C13/\$1C26		RJMP/JMP	SPM_RDY	;SPM命令操作可
		}		
\$0C14/\$1C27	RESET:	LDI	R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (プログラム開始)
\$0C15/\$1C28		OUT	SPH, R16	;スタックポインタ上位を初期化
\$0C16/\$1C29		LDI	R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0C17/\$1C2A		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

### 8.1.1. 応用領域とブート領域間の割り込みベクタ移動

MCU制御レジスタ(MCUCR)は割り込みベクタ表の配置を制御します。

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	-	-	-	PUD	RSTDIS	CKRC81	IVSEL	IVCE	MCUCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0/1(注)	0	0	

注: 値はCKSEL3~0ヒューズで初期化されます(CKSEL3~0=0110時は1、他の全ての場合は0)。

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

IVSELビットが解除(0)されると、割り込みベクタはフラッシュメモリの先頭に配置されます。このビットが設定(1)されると、割り込みベクタはフラッシュメモリのブートローダ領域の始まりへ移動されます。ブートフラッシュ領域先頭の実際のアドレスはBOOTSZヒューズによって決定されます。詳細については126頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」章を参照してください。割り込みベクタ表の予期せぬ変更を防ぐため、IVSELビットの変更は特別な書き込み手順に従わなければなりません。

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

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

注: 割り込みベクタがブートローダ領域に配置され、BLB02ブート施錠ビットがプログラム(0)されると、応用領域から実行する間中、割り込みが禁止されます。割り込みベクタが応用領域に配置され、BLB12ブート施錠ビットがプログラム(0)されると、ブートローダ領域から実行する間中、割り込みが禁止されます。ブート施錠ビットの詳細については139頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」を参照してください。

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

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

## アセンブリ言語プログラム例

```

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

```

## C言語プログラム例

```

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

```

## 9. 入出力ポート

### 9.1. 序説

全てのAVRのポートは標準デジタルI/Oポートとして使われるとき、真の読み-変更-書き(リード・モディファイライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方向をも無意識に変更することなく、1つのポートピンの方向を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。全てのポートピンは個別に選択可能な、供給電圧で抵抗値が変化しないプルアップ抵抗を持っています。全てのI/Oピンは図9-1.で示されるようにVCCとGNDの両方に保護ダイオードがあります。各値の完全な一覧については160頁の「電気的特性」を参照してください。

本章内の全てのレジスタとビットの参照は一般形で記されます。小文字の'x'はポート番号文字、小文字の'n'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記されたPORTXnがポートBのビット3に対してはPORTB3)が使われなければなりません。物理的なI/Oレジスタとビット位置は51頁の「I/Oポート用レジスタ」で一覧されます。

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

標準デジタルI/OとしてのI/Oポートの使用は次の「標準デジタル入出力としてのポート」で記述されます。多くのポートピンはデバイスの周辺機能用の交換機能と多重化されます。ポートピンとの各交換機能のインターフェース法は45頁の「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項目を参照してください。

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

### 9.2. 標準デジタル入出力としてのポート

このポートは任意の内部プルアップ付き双方向I/Oポートです。図9-2.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9-1. 入出力ピン等価回路

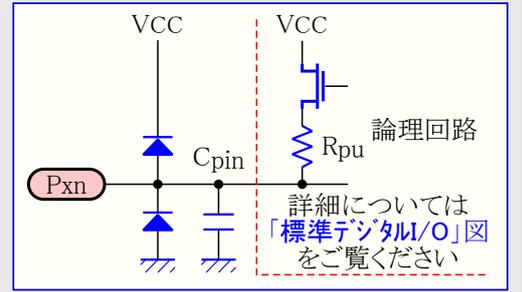
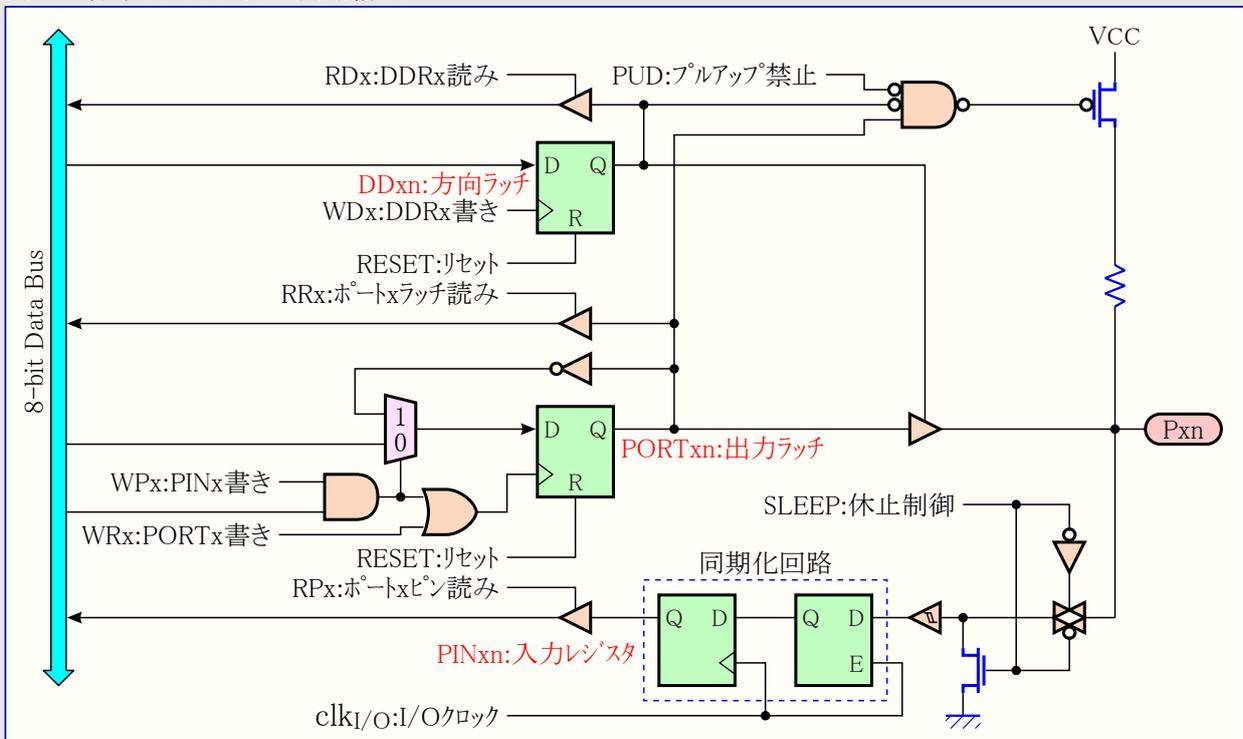


図9-2. 標準デジタル入出力回路構成



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

## 9.2.1. 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。51頁の「I/Oポート用レジスタ」で示されるように、DDxnビットはDDRx I/Oアドレス、PORTxnビットはPORTx I/Oアドレス、PINxnビットはPINx I/Oアドレスでアクセスされます。

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

そのピンが入力ピンとして設定される時にPORTxnが論理1を書かれると、プルアップ抵抗が活性(有効)にされます。プルアップ抵抗をOFFに切り替えるにはPORTxnが論理0を書かれるか、またはそのピンが出力ピンとして設定されなければなりません。

ポートピンは例えばクロックが動いていなくても、リセット条件が活性(有効)になるとHi-Zにされます。

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

## 9.2.2. 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。

## 9.2.3. 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9-1.はピン値に対する制御信号の一覧を示します。

表9-1. ポートピンの設定

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

## 9.2.4. 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9-2.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(エン)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9-3.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9-3.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9-4.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9-3. 外部供給ピン値読み込み時の同期化

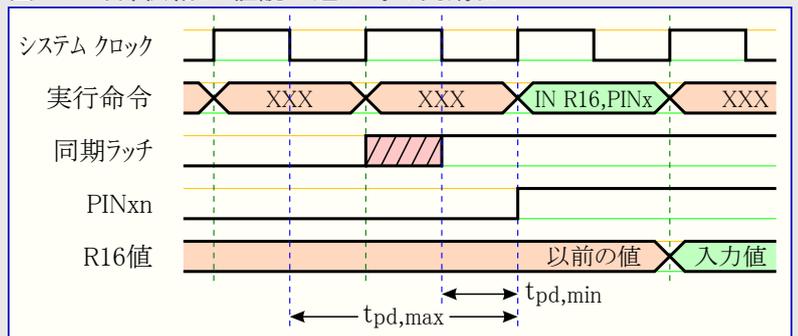
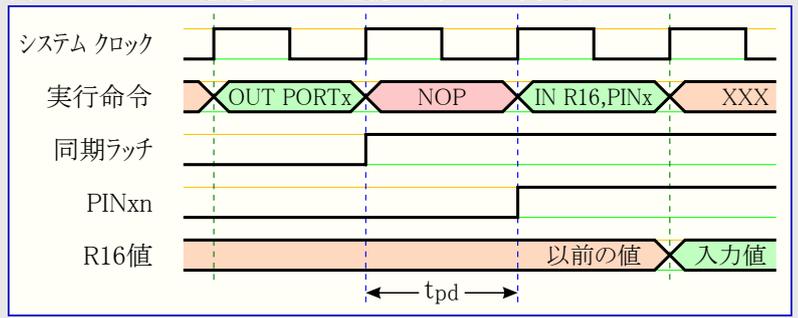


図9-4. プログラムで設定したピン値読み戻し時の同期化



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

#### アセンブリ言語プログラム例

```

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

```

#### C言語プログラム例

```

unsigned char i;
~
PORTB = (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0); /* */
DDRB = (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0); /* 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。

### 9.2.5. デジタル入力許可と休止形態

図9-2. で示されるようにデジタル入力信号はシュミットトリガの入力をGNDにクランプできます。この図でSLEEPと印された信号は入力信号のいくつかが開放のまま、またはVCC/2付近のアナログ信号電圧を持つ場合の高消費電力を避けるため、**パワーダウン動作**と**スタンバイ動作**でMCU休止制御器によって設定(1)されます。

SLEEPは外部割り込みピンとして許可されたポートピンに対しては無視されます。外部割り込み要求が許可されないなら、SLEEPは他のピンについてと同様に有効です。SLEEPは45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。

### 9.2.6. 未接続ピン (訳注: 共通性のため本項追加)

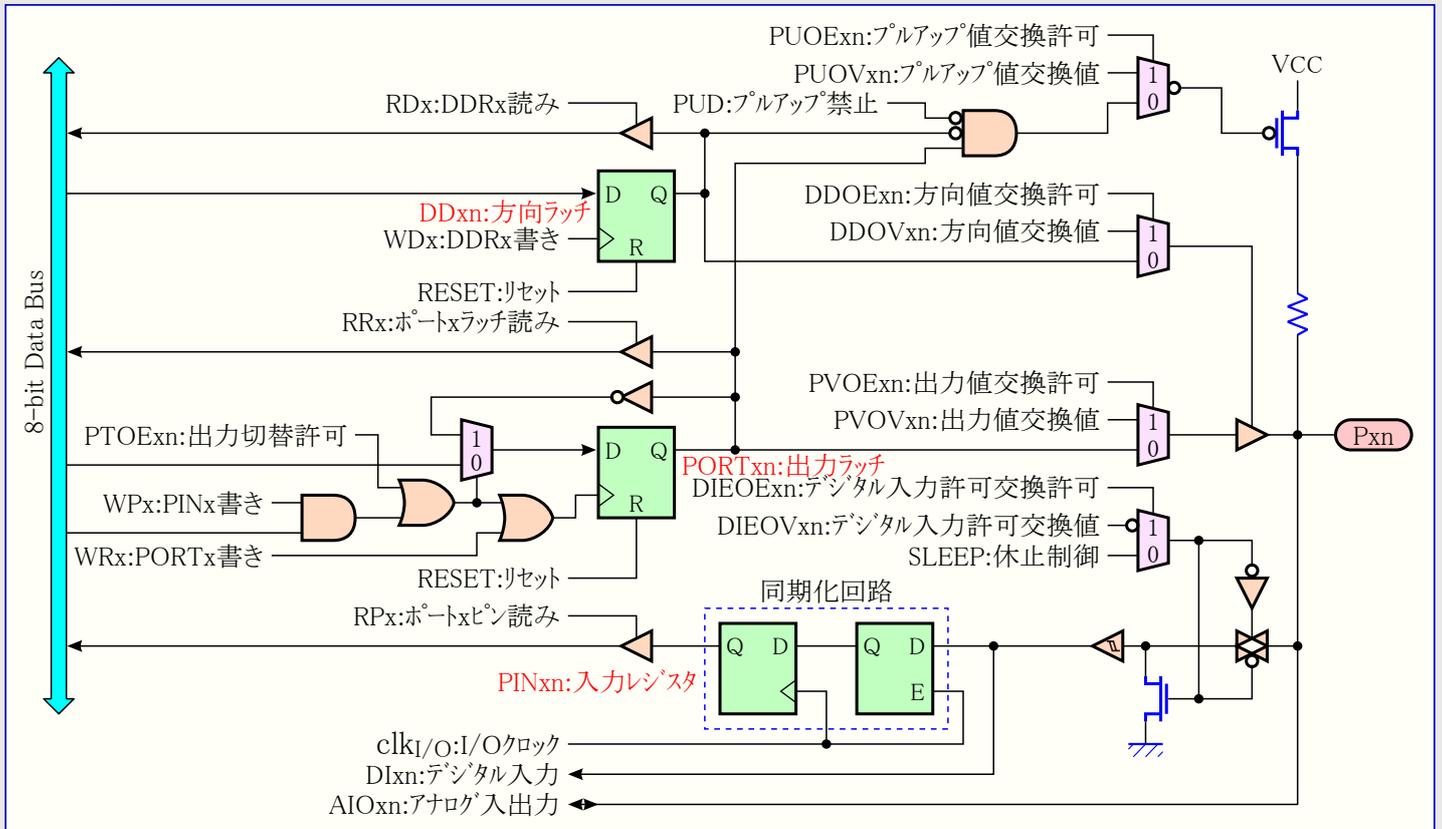
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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。

### 9.3. 交換ポート機能

多くのポートピンには標準デジタル入出力に加え交換機能があります。図9-5.は単純化された図9-2.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。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9-5. 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9-2.は重複(交換)信号の機能一覧を示します。図9-5.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9-2. 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9-5.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表9-2.の交換機能説明を参照してください。

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

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	-	-	-	PUD	RSTDIS	CKRC81	IVSEL	IVCE	MCUCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0/1(注)	0	0	

注: 値はCKSEL3~0ヒューズで初期化されます(CKSEL3~0=0110時は1、他の全ての場合は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43頁の「ピンの設定」をご覧ください。

### 9.3.2. ポートBの交換機能

ポートBピンの交換機能は表9-3.で示されます。

表9-3. ポートBピンの交換機能

ポートピン	交換機能	ポートピン	交換機能	
PB7	ICP1 (タイマ/カウンタ1 捕獲起動入力)	PB3	PSCOUTR1 (縮小電力段制御器 出力1)	
	PSCOUT22 (電力段制御器2 出力2)		ADC2 (A/D変換アナログ入力チャネル2)	
	ADC9 (A/D変換アナログ入力チャネル9)		ACMP2M (アナログ比較器2 反転入力)	
PB6	MISO (SPI 主側データ入力/従側データ出力)	PB2	INT0 (外部割り込み0入力)	
	ADC8 (A/D変換アナログ入力チャネル8)		PSCOUT21 (電力段制御器2 出力1)	
	ACMP3 (アナログ比較器3 非反転入力)			
PB5	INT1 (外部割り込み1入力)	PB1	PSCOUT20 (電力段制御器2 出力0)	
	SCK (SPI 直列クロック 主側出力/従側入力)			
	ADC5 (A/D変換アナログ入力チャネル5)			
PB4	ACMP2 (アナログ比較器2 非反転入力)	PB0	T1 (タイマ/カウンタ1外部クロック入力)	
	MOSI (SPI 主側データ出力/従側データ入力)		PSCOUT23 (電力段制御器2 出力3)	
	ADC3 (A/D変換アナログ入力チャネル3)		ACMP3_OUT (アナログ比較器3 出力)	
	ACMPM (アナログ比較器基準入力)			

交換ピンの設定は次のとおりです。

#### • ICP1/PSCOUT22/ADC9 – ポートB ビット7 : PB7

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

PSCOUT22 : 電力段制御器(PSC)2の出力2。

ADC9 : A/D変換チャネル9入力。

#### • MISO/ADC8/ACMP3 – ポートB ビット6 : PB6

MISO : SPI用主装置データ入力、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DDB6設定に拘らず入力として設定されます。SPIが従装置として許可されると、本ピンのデータ方向はDDB6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にプルアップは未だポートB出力レジスタ(PORTB)のPORTB6とPUDビットで制御できます。

ADC8 : A/D変換チャネル8入力。

ACMP3 : アナログ比較器3 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

#### • INT1/SCK/ADC5/ACMP2 – ポートB ビット5 : PB5

INT1 : 外部割り込み1入力。本ピンはMCUへの外部割り込み元として扱えます。

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

ADC5 : A/D変換チャネル5入力。

ACMP2 : アナログ比較器2 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

#### • MOSI/ADC3/ACMPM – ポートB ビット4 : PB4

MOSI : SPI用主装置データ出力、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DDB4設定に拘らず入力として設定されます。SPIが主装置として許可されると、本ピンのデータ方向はDDB4で制御されます。このピンがSPIによって入力を強制される時にプルアップは未だポートB出力レジスタ(PORTB)のPORTB4とPUDビットで制御できます。

ADC3 : A/D変換チャネル3入力。

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

• PSCOUTR1/ADC2/ACMP2M – ポートB ビット3 : PB3

PSCOUTR1 : 縮小電力段制御器(PSCR)の出力1。

ADC2 : A/D変換チャネル2入力。

ACMP2M : アナログ比較器2反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

• INT0/PSCOUT21 – ポートB ビット2 : PB2

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

PSCOUT21 : 電力段制御器(PSC)2の出力1。

• PSCOUT20 – ポートB ビット1 : PB1

PSCOUT20 : 電力段制御器(PSC)2の出力0。

• T1/PSCOUT23/ACMP3\_OUT – ポートB ビット0 : PB0

T1 : タイマ/カウンタ1の外部クロック入力。

PSCOUT23 : 電力段制御器(PSC)2の出力3。

ACMP3\_OUT : アナログ比較器3の出力。

表9-4.と表9-5.はポートBの交換機能を45頁の図9-5.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。

表9-4. ポートB7~4の交換機能用交換信号

信号名	PB7/ICP1/PSCOUT22/ ADC9	PB6/MISO/ADC8/ACMP3	PB5/INT1/SCK/ADC5/ ACMP2	PB4/MOSI/ADC3/ACMPM
PUOE	0	SPE・MSTR	SPE・MSTR	SPE・MSTR
PUOV	0	PORTB6・ $\overline{\text{PUD}}$	PORTB5・ $\overline{\text{PUD}}$	PORTB4・ $\overline{\text{PUD}}$
DDOE	PSC22許可	SPE・MSTR	SPE・MSTR	SPE・MSTR
DDOV	1	0	0	0
PVOE	PSC22許可	MSTR	MSTR	MSTR
PVOV	PSC22出力	MISO	SCK	MOSI
PTOE	-	-	-	-
DIOE	ADC9D	ADC8D	ADC5D INT1許可	ADC3D
DIOV	0	0	INT1許可	0
DI	ICP1	-	INT1	-
AIO	ADC9	ADC8/ACMP3	ADC5/ACMP2	ADC3/ACMPM

表9-5. ポートB3~0の交換機能用交換信号

信号名	PB3/PSCOUTR1/ADC2/ ACMP2M	PB2/INT0/PSCOUT21	PB1/PSCOUT20	PB0/PSCOUT23/ ACMP3_OUT
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	PSC01許可	PSC21許可	PSC20許可	PSC23許可 AC3EN
DDOV	1	1	1	1
PVOE	PSC01許可	PSC21許可	PSC20許可	PSC23許可 AC3EN
PVOV	PSCOUTR1	PSCOUT21	PSCOUT20	PSC23許可ならPSCOUT23 AC3ENならACMP3OUT さもなければ 0
PTOE	-	-	-	-
DIOE	ADC2D	INT0許可	-	-
DIOV	-	INT0許可	-	-
DI	-	INT0	-	-
AIO	ADC2/ACMP2M	-	-	-

### 9.3.3. ホートDの交換機能

ホートDピンの交換機能は表9-6.で示されます。

表9-6. ホートDピンの交換機能

ホートピン	交換機能	ホートピン	交換機能
PD7	PSCINrA (縮小電力段制御器 第1代替デジタル入力)	PD3	ADC1 (A/D変換アナログ入力チャンネル1)
	ADC10 (A/D変換アナログ入力チャンネル10)		ACMP2_OUT (アナログ比較器2 出力)
PD6	AMP0+ (アナログ差動増幅器0 非反転入力)	PD2	ADC0 (A/D変換アナログ入力チャンネル0)
PD5	ADC7 (A/D変換アナログ入力チャンネル7)	PD1	ACMP1 (アナログ比較器1 非反転入力)
	AMP0- (アナログ差動増幅器0 反転入力)		PSCOUTR0 (縮小電力段制御器 出力0)
PD4	PSCIN2A (電力段制御器2 代替デジタル入力)	PD0	PSCINrB (縮小電力段制御器 第2代替デジタル入力)
	ADC4 (A/D変換アナログ入力チャンネル4)		$\overline{SS}$ (SPI 従装置選択入力)
	ACMP3M (アナログ比較器3 反転入力)		ACMP3_OUT_A (アナログ比較器3 代替出力)
			CLKO (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。

- PSCINrA/ADC10 - ホートDビット7 : PD7

PSCINrA : 縮小電力段制御器の第1代替デジタル入力。

ADC10 : A/D変換チャンネル10入力。

- AMP0+ - ホートDビット6 : PD6

AMP0+ : A/D変換器用アナログ差動増幅器0非反転入力。

- ADC7/AMP0- - ホートDビット5 : PD5

ADC7 : A/D変換チャンネル7入力。

AMP0- : A/D変換器用アナログ差動増幅器0反転入力。

- PSCIN2A/ADC4/ACMP3M - ホートDビット4 : PD4

PSCIN2A : 電力段制御器2の代替デジタル入力。

ADC4 : A/D変換チャンネル4入力。

ACMP3M : アナログ比較器3反転入力。アナログ比較器機能を邪魔するデジタルホート機能は無効とするために、内部プルアップがOFFにされた入力としてホートピンを設定してください。

- ADC1/ACMP2\_OUT - ホートDビット3 : PD3

ADC1 : A/D変換チャンネル1入力。

ACMP2\_OUT : アナログ比較器2の出力。

- ADC0/ACMP1 - ホートDビット2 : PD2

ADC0 : A/D変換チャンネル0入力。

ACMP1 : アナログ比較器1非反転入力。アナログ比較器機能を邪魔するデジタルホート機能は無効とするために、内部プルアップがOFFにされた入力としてホートピンを設定してください。

- PSCOUTR0/PSCINrB - ホートDビット1 : PD1

PSCOUTR0 : 縮小電力段制御器の出力0。

PSCINrB : 縮小電力段制御器の第2代替デジタル入力。

- $\overline{SS}$ /ACMP3\_OUT\_A/CLKO - ホートDビット0 : PD0

$\overline{SS}$  : 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DDD0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、本ピンのデータ方向はDDD0によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だPORTD0によって制御できます。

ACMP3\_OUT\_A : アナログ比較器3の代替出力。

CLKO : システムクロック出力。分周したシステムクロックがこのピンに出力できます。分周したシステムクロックはCKOUTヒューズまたはCOUTビットがプログラム(0)されると、PORTD0とDDD0設定に拘らず、出力されます。これはリセット中に出力されません。

表9-7.と表9-8.はポートDの交換機能を45頁の図9-5.で示される交換信号に関連付けます。

表9-7. ポートD7~4の交換機能用交換信号

信号名	PD7/PSCINrA/ADC10	PD6/AMP0+	PD5/ADC7/AMP0-	PD4/PSCIN2A/ADC4/ACMP3M
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIEOE	ADC10D	AMP0+D	ADC7D	ADC4D
DIEOV	0	0	0	0
DI	PSCINrA	-	-	PSCIN2A
AIO	ADC10	AMP0+	ADC7/AMP0-	ADC4/ACMP3M

表9-8. ポートD3~0の交換機能用交換信号

信号名	PD3/ADC1/ACMP2_OUT	PD2/ADC0/ACMP1	PD1/PSCOUTR0/PSCINrB	PD0/ $\overline{SS}$ /ACMP3_OUT_A/CLKO
PUOE	0	0	0	SPE· $\overline{MSTR}$
PUOV	0	0	0	PORTD0· $\overline{PUD}$
DDOE	ACE2EN	0	PSC00許可	ACMP3D (SPE· $\overline{MSTR}$ )
DDOV	1	0	1	AC3EN
PVOE	AC2EN	0	PSC00許可	AC3EN
PVOV	ACMP2_OUT	0	PSCOUT00	ACMP3_OUT
PTOE	-	-	-	-
DIEOE	ADC1D	ADC0D	0	0
DIEOV	0	0	0	0
DI	-	-	PSCINrB	$\overline{SS}$
AIO	ADC1	ADC0/ACMP1	-	-

### 9.3.4. ポートEの交換機能

ポートEピンの交換機能は表9-9.で示されます。

表9-9. ポートEピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PE3	ADC6 (A/D変換アナログ入力チャネル6) AREF (アナログ基準電圧)	PE1	PSCIN2 (電力段制御器2 デジタル入力) ACMP1_OUT (アナログ比較器1 出力) XTAL1 (主クロック用発振増幅器入力)
PE2	ACMP1M (アナログ比較器1 反転入力) PSCINr (縮小電力段制御器 デジタル入力) XTAL2 (主クロック用発振増幅器出力)	PE0	INT2 (外部割り込み2入力) RESET (外部リセット入力) OCD (内蔵デバッグ機能用入出力)

交換ピンの設定は次のとおりです。

• ADC6/AREF – ポートE ビット3 : PE3

ADC6 : A/D変換チャネル6入力。

AREF : アナログ基準電圧。このピンの詳細については129頁の表17-3.をご覧ください。

このピンはデジタル出力ピンとしてだけ使うことができます。デジタル入力として読むことはできません。

• ACMP1M/PSCINr/XTAL2 – ポートE ビット2 : PE2

ACMP1M : アナログ比較器1反転入力。アナログ比較器機能を邪魔するデジタルポート機能を無効とするために、内部プルアップがOFFにされた入力としてポートピンを設定してください。

PSCINr : 縮小電力段制御器のデジタル入力。

XTAL2 : 主クロック用発振増幅器出力。クリスタル用低電力発振器用のクロックピンとして使われます。クロックピンとして使われると、このピンはI/Oピンとして使えません。

• PSCIN2/ACMP1\_OUT/XTAL1 – ポートE ビット1 : PE1

PSCIN2 : 電力段制御器2のデジタル入力。

ACMP1\_OUT : アナログ比較器1の出力。

XTAL1 : 主クロック用発振増幅器入力。校正付き内蔵RC発振器を除く全てのチップクロック元用に使われます。クロックピンとして使われると、このピンはI/Oピンとして使えません。

• INT2/RESET – ポートE ビット0 : PE0

INT2 : 外部割り込み2入力。このピンはMCUへの外部割り込み元として扱えます。

RESET : 外部リセット入力。RSTDISBLヒューズがプログラム(0)またはMCU制御レジスタ(MCU\_CR)のリセットピン禁止(RSTDIS)ビットが設定(1)されると、このピンは標準のI/Oピンとして機能し、デバイスにはリセット元として電源ONリセットと低電圧リセットに頼らなければなりません。RSTDISBLヒューズが非プログラム(1)にされ且つRSTDISビットが解除(0)されると、本ピンにリセット回路が接続され、本ピンはI/Oピンとして使えません。

PE0がリセットピンとして使われると、PORTE0, DDE0, PINE0は全て0を読みます。

表9-10.はポートEの交換機能を45頁の図9-5.で示される交換信号に関連付けます。

表9-10. ポートE3~0の交換機能用交換信号

信号名	PE3/ADC6/AREF	PE2/ACMP1M/PSCINr /XTAL2	PE1/PSCIN2/XTAL1	PE0/INT2/RESET/OCD
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	AC1EN	0
DDOV	0	0	1	0
PVOE	0	0	AC1EN	0
PVOV	0	0	ACMP1_OUT	0
PTOE	-	-	-	-
DIOE	0	ACMP1MD	0	INT2許可
DIOV	0	0	0	INT2許可
DI	-	-	PSCIN2	INT2
AIO	ADC6	ACMP1M	-	-

## 9.4. I/Oポート用レジスタ

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

ビット	7	6	5	4	3	2	1	0	
\$0D (\$2D)	–	–	–	–	–	DDE2	DDE1	DDE0	DDRE
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

ビット	7	6	5	4	3	2	1	0	
\$0C (\$2C)	–	–	–	–	–	PINE2	PINE1	PINE0	PINE
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	不定	不定	不定	

## 10. 外部割り込み

外部割り込みはINT2~0ピンによって起動されます。許可したなら、例えばINT2~0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、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT2~0の上昇端または下降端割り込みの認知は18頁の「**クロックシステムとその配給**」で記述されるI/Oクロックの存在を必要とすることに注意してください。I/Oクロックは**アイドル動作**を除く**全休止形態**で停止されます。

レベル起動割り込みが**パワーダウン動作**からの起動に使われる場合、変化したレベルはMCUを起動するために一定時間保たなければならないことに注意してください。これはMCUの雑音不安定性を減らします。変化したレベルはウォッチドッグ発振器クロックによって2度採取されます。ウォッチドッグ発振器の周期は3V、25°Cで(公称)7.8μsです。ウォッチドッグ発振器の周波数は168頁の「**代表特性**」で示されるように電圧依存です。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18頁の「**システムクロックとクロック選択**」で示されるように**SUTヒューズ**で定義されます。このレベルがウォッチドッグ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。

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

ビット (\$89)	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7~0 – 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10-1**で定義されます。INT2~0ピンの値は端検知前に採取されています。端または論理変化割り込みが選ばれる場合、1クロック周期より長く留まるパルスは割り込みを生成します。より短いパルスは割り込みの発生が保証されません。**システムクロック前置分周器**が許可される場合、CPUクロック周波数がXTAL周波数より低くできることを考慮してください。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。

表10-1. 外部割り込み2~0 割り込み条件

ISc <sub>n1</sub>	ISc <sub>n0</sub>	割り込み発生条件
0	0	INT <sub>n</sub> ピンのLowレベル
0	1	INT <sub>n</sub> ピンの論理変化(両端)
1	0	INT <sub>n</sub> ピンの下降端
1	1	INT <sub>n</sub> ピンの上昇端

注: n=2~0

ISc<sub>n1,0</sub>ビット変更時、割り込みはEIMSKで割り込み許可ビットの解除(0)によって禁止されなければなりません。さもなければビットが変更される時に割り込みが起き得ます。

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

ビット (\$21 (\$41))	7	6	5	4	3	2	1	0	
	–	–	–	–	–	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A(EICRA)**の割り込み条件制御n(ISc<sub>n1</sub>,ISc<sub>n0</sub>)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。

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

ビット (\$20 (\$40))	7	6	5	4	3	2	1	0	
	–	–	–	–	–	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2~0 – INTF2~INTF0 : 外部割り込み2~0要求フラグ (External Interrupt Flag2~0)

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、これらのフラグは常に解除(0)されます。

## 11. 縮小16ビット タイマ/カウンタ1

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

- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 1つの捕獲入力部
- 捕獲入力雑音除去器
- 外部事象計数器
- 2つの独立した割り込み (TOV1, ICF1)

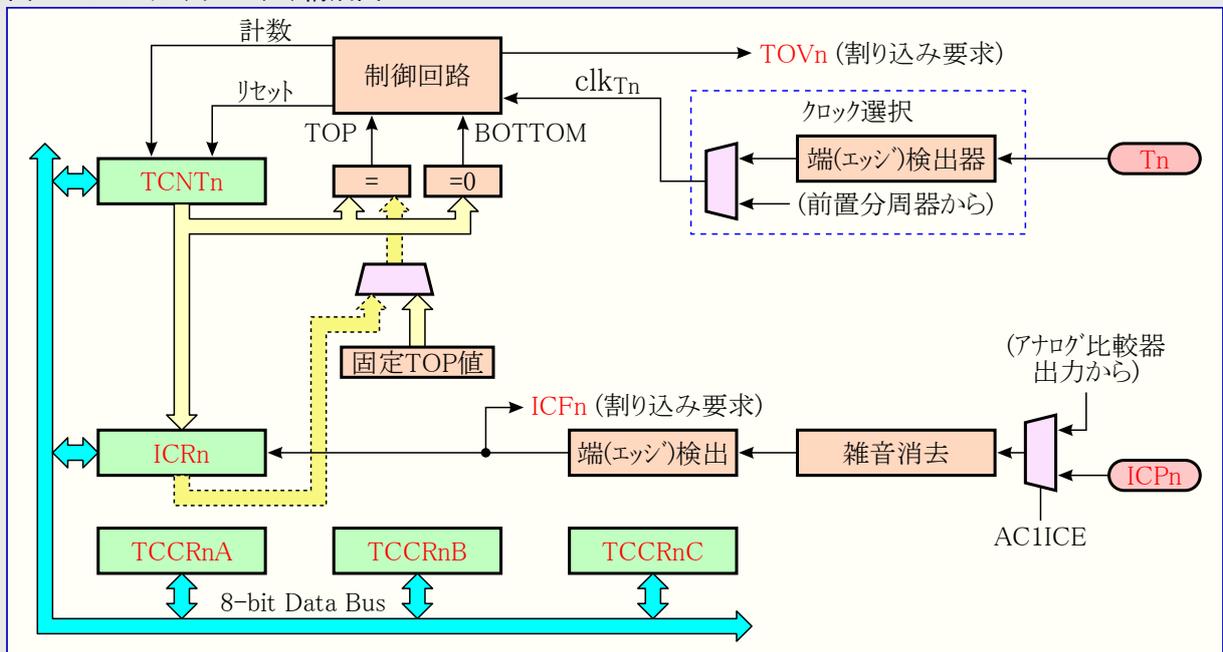
### 11.1. 概要

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

この16ビットタイマ/カウンタの単純化した構成図は図11-1.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59頁の「16ビットタイマ/カウンタ1用レジスタ」で示されます。

31頁の「PRR – 電力削減レジスタ」のPRTIM1ビットはタイマ/カウンタ1部を許可するために0を書かれなければなりません。

図11-1. 16ビットタイマ/カウンタ構成図



注: タイマ/カウンタ1のピンの配置と説明については3頁の表2-1と4頁の表2-2を参照してください。

#### 11.1.1. 関係レジスタ

タイマ/カウンタ(TCNT1)、捕獲レジスタ(ICR1)は全て16ビットレジスタです。16ビットレジスタをアクセスする時は特別な手順に従わなければなりません。これらの手順は54頁の「16ビットレジスタのアクセス」項で記述されます。タイマ/カウンタ1制御レジスタB(TCCRB)は8ビットレジスタで、CPUアクセスの制限はありません。割り込み要求信号はタイマ/カウンタ1割り込み要求レジスタ(TIFR1)で全て見えます。全ての割り込みはタイマ/カウンタ1割り込み許可レジスタ(TIMSK1)で個別に遮蔽(禁止)されます。TIFR1とTIMSK1は図11-1.で示されません。

このタイマ/カウンタは内部的またはT1ピンの外部クロック元によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加するのに使うクロック元と端(エッジ)を制御します。クロック元が選ばれない時にタイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック( $clk_{T1}$ )として参照されます。

捕獲レジスタ(ICR1)は捕獲起動(ICP1)ピンまたはアナログ比較器出力(116頁の「アナログ比較器」参照)のどちらかの外部(端で起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波器(雑音除去器)を含みます。

TOP値または最大タイマ/カウンタ値は、いくつかの動作種別で、ICR1または一群の固定値のどれかによって定義できます。

### 11.1.2. 定義

次の定義は本文書を通して広範囲に使われます。

表11-1. 用語定義

用語	意味
BOTTOM	カウンタが\$0000に到達した時。
MAX	カウンタが\$FFFF(65535)に到達した時。
TOP	カウンタがTOP値に到達した時(計数動作での最大値と等しくなった時)。TOP値はICR1値で指定できます。この指定は動作種別に依存します。

### 11.2. 16ビットレジスタのアクセス

TCNT1とICR1は8ビットバス経由でAVR CPUによってアクセスできる16ビットレジスタです。この16ビットレジスタは2回の読みまたは書き操作を使ってバイトアクセスされなければなりません。16ビットタイマ/カウンタは16ビットアクセスの上位バイトの一時保存用に1つの8ビットレジスタを持ちます。16ビットタイマ/カウンタ内の全ての16ビットレジスタ間で、この同じ一時レジスタが共用されます。下位バイトアクセスが16ビット読み書き動作を起動します。16ビットレジスタの下位バイトがCPUによって書かれると、一時レジスタに保存した上位バイトと書かれた下位バイトは同じクロック周期で両方が16ビットレジスタに複写されます。16ビットレジスタの下位バイトがCPUによって読まれると、16ビットレジスタの上位バイトは下位バイトが読まれるのと同じクロック周期で一時レジスタに複写されます。

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

次のコード例は割り込みが一時レジスタを更新しないことが前提の16ビットタイマ/カウンタレジスタのアクセス法を示します。ICR1レジスタのアクセスに対して同じ原理が直接的に使えます。C言語を使う時にコンパイラが16ビットアクセスを扱うことに注意してください。

#### アセンブリ言語プログラム例

```

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

```

#### C言語プログラム例

```

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

```

**注:** このコード例はデバイス固有ヘッダファイルがインクルードされていると仮定します。

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

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

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

次のコード例はTCNT1レジスタ内容の非分断読み込み法を示します。同じ原理を用いることによってICR1の読み込みも行えます。

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

**注:** このコード例はデバイス固有ヘッダファイルがインクルードされていると仮定します。

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

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

次のコード例はTCNT1レジスタ内容の非分断書き込み法を示します。同じ原理を用いることによってICR1の書き込みも行えます。

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

**注:** このコード例はデバイス固有ヘッダファイルがインクルードされていると仮定します。

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

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

### 11.2.1. 上位バイト一時レジスタの再使用

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

### 11.3. タイマ/カウンタのクロック元

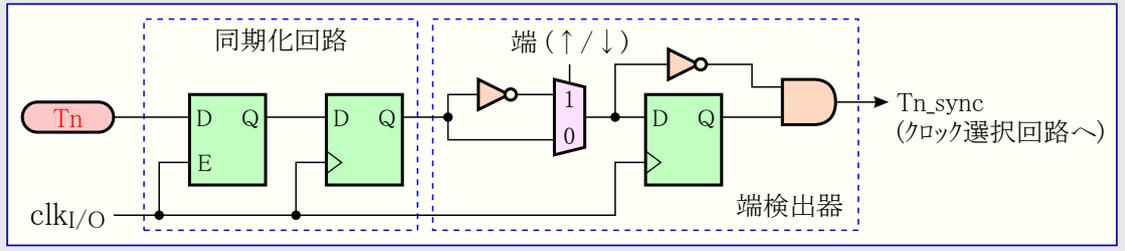
このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR1B)に配置されたクロック選択(CS12~0)ビットによって制御されるクロック選択論理回路で選ばれます。

#### 11.3.1. 外部クロック元

T1ピンに印加された外部クロック元はタイマ/カウンタクロック( $f_{clk\_T1}$ )として使えます。このT1ピンはピン同期化論理回路によって全てのシステムクロック周期に一度採取されます。この同期化(採取)された信号はその後に端(エッジ)検出器を通して通過されます。図11-2はT1同期化と端検出器論理回路の機能等価構成図を示します。レジスタは内部システムクロック( $f_{clk\_I/O}$ )の上昇端でクロック駆動されます。ラッチは内部システムクロックのHigh区間で通過(Low区間で保持)です。

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

図11-2. T1ピンの採取等価構成図



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

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

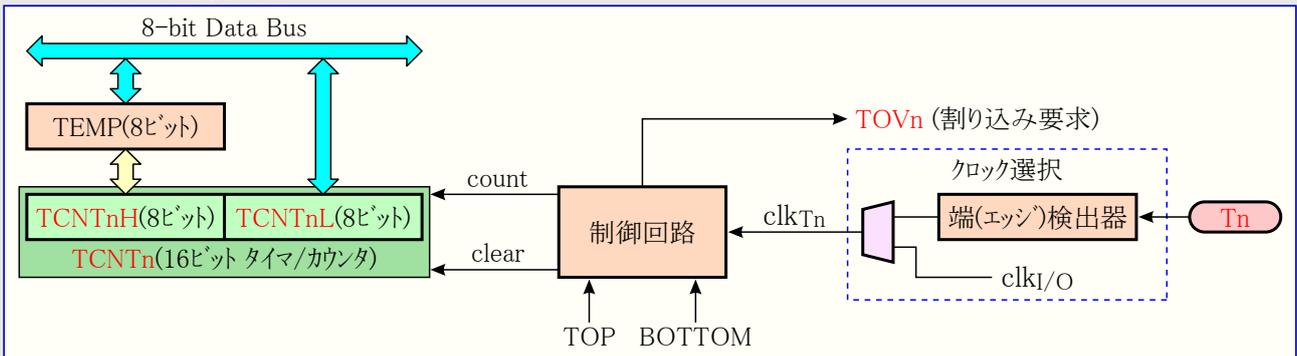
印加された外部クロックの各半周期は正しい採取を保証するために1システムクロック周期より長くなければなりません。この外部クロックは50%/50%デューティ比で与えられるものとして、システムクロック周波数の半分未満( $f_{EXTclk} < f_{clk\_I/O}/2$ )であることが保証されなければなりません。端検出器が採取を使うため、検出できる外部クロックの最大周波数は採取周波数の半分です(ナイキストの標本化定理)。然しながら、発振元(水晶発振子、セラミック振動子、コンデンサ)公差によって引き起こされたシステムクロック周波数やデューティ比の変動のため、外部クロック元の最大周波数は $f_{clk\_I/O}/2.5$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

外部クロック元は前置分周できません。

### 11.4. 計数器部

16ビットタイマ/カウンタの主な部分は設定可能な16ビット計数器部です。図11-3はこのカウンタとその周辺の構成図を示します。

図11-3. 計数器部構成図



count	TCNT1を1つ進める信号。
clear	TCNT1のリセット(\$0000設定)信号。
clkTn	以降で $clk_{T1}$ として参照されるタイマ/カウンタクロック信号。
TOP	TCNT1が最大値に到達したことを示す信号。
BOTTOM	TCNT1が最小値(\$0000)に到達したことを示す信号。
TEMP	一時レジスタ。

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

使った動作種別に依存して、カウンタは各タイマ/カウンタクロック( $\text{clk}_{T1}$ )でリセット(0)または増加(+1)されます。 $\text{clk}_{T1}$ はクロック選択(CS12~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS12~0=000)時にカウンタは停止されます。けれどもTCNT1値はタイマ/カウンタクロック( $\text{clk}_{T1}$ )が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先順位を持ちます)。

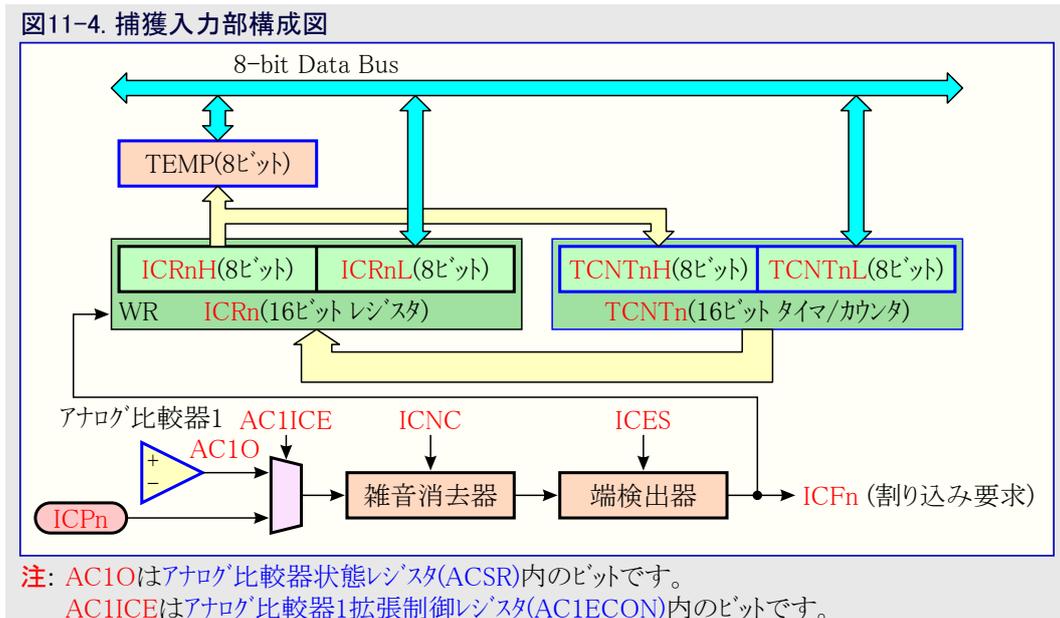
計数順序(方法)はタイマ/カウンタ制御レジスタB(TCCR1B)に配置された波形生成種別(WGM13)ビットの設定によって決定されます。

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

## 11.5. 捕獲入力部

タイマ/カウンタは外部の出来事を捕獲でき、発生時間を示す時間印(タイマ/カウンタ値)を与える捕獲入力部と合体します。出来事または複数の出来事を示す外部信号はICP1ピンまたは代わりにアナログ比較器部経由で印加できます。時間印はその後、周波数、デューティ比、印加された信号の他の特性の計算に使えます。代わりに時間印は出来事の記録作成にも使えます。

捕獲入力部は図11-4.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。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号を示します。



捕獲起動入力(ICP1)ピン若しくは代わりにアナログ比較器出力(AC10)で論理レベルの変化(出来事)が起き、その変化が端(エッジ)検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF1)はTCNT1値がICR1に複写されるのと同じシステムクロックで設定(1)されます。許可(I=1, ICIE1=1)ならば捕獲割り込み要求フラグは捕獲割り込みを発生します。ICF1は割り込みが実行されると自動的に解除(0)されます。代わりにこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。

捕獲レジスタ(ICR1)の16ビット値読み込みは、初めに下位バイト(ICR1L)、その後に上位バイト(ICR1H)を読むことによって行われます。下位バイトが読まれる時に上位バイトが上位バイト一時レジスタ(TEMP)に複写されます。CPUがICR1H I/O位置を読むと、この一時レジスタにアクセスします。

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

16ビットレジスタアクセス法のより多くの情報については54頁の「16ビットレジスタのアクセス」を参照してください。

### 11.5.1. 捕獲起動元

捕獲入力部用の主な起動元は捕獲起動入力(ICP1)ピンです。タイマ/カウンタ1は捕獲入力部用起動元としてアナログ比較器出力を代わりに使えます。アナログ比較器はアナログ比較器1拡張制御レジスタ(AC1ECON)のアナログ比較器1割り込み捕獲許可(AC1ICE)ビットの設定(1)によって起動元として選ばれます。起動元を変更することが捕獲を起動し得ることに気付いてください。従って捕獲割り込み要求フラグ(ICF1)は、その変更後に解除(0)されなければなりません。

捕獲起動入力(ICP1)ピンとアナログ比較器出力(AC10)の両入力は、T1ピン(56頁の図11-2参照)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します。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されないなら、雑音除去器と端検出器の入力が常に許可されることに注意してください。

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

### 11.5.2. 雑音除去器

雑音除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。雑音除去器の入力は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。

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

### 11.5.3. 捕獲入力の使用

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

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

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

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

## 11.6. 動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は**波形生成種別(WGM13)**ビットによって定義されます。

タイミング情報の詳細については59頁の「**タイマ/カウンタのタイミング図**」を参照してください。

### 11.6.1. 標準動作

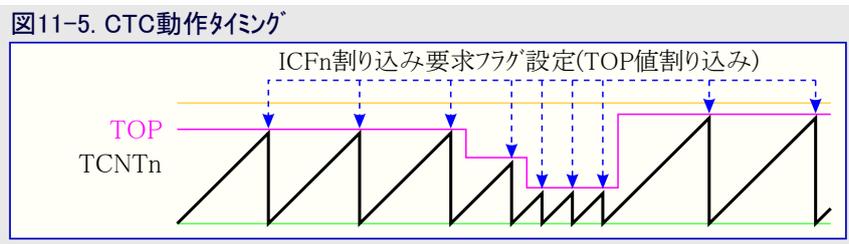
最も単純な動作種別が標準動作(WGM13=0)です。この動作種別での計数方向は常に上昇(+)で、カウンタの解除は実行されません。カウンタは16ビット最大値(MAX=\$FFFF)を通過すると単に範囲を超え、そして\$0000(BOTTOM)から再び始めます。通常動作での**タイマ/カウンタ溢れ(TOV1)フラグ**は**TCNT1**が\$0000になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV1フラグは設定(1)のみで解除(0)されないことを除いて第17ビットのようになります。けれどもTOV1フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わ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

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

### 11.6.2. 比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM13=1、以前の種別12)では**ICR1**がカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT1)値がICR1と一致する時にカウンタが\$0000に解除されます。ICR1はカウンタに対する**TOP値**、従って分解能も定義します。この動作種別はより大きい比較一致出力周波数の制御を許します。それは外部の出来事の計数操作も簡単にします。

CTC動作についての**タイミング図**は図11-5.で示されます。カウンタ(TCNT1)値はICR1のどちらかで比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT1)は解除(\$0000)されます。



ICF1を使うことによってカウンタ値がTOP値に到達する時毎に割り込みが生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。けれどもTOPをBOTTOMに近い値へ変更するのは、CTC動作が2重緩衝機能を持たないために注意して行わなければなりません。ICR1に書かれた新しい値がTCNT1の現在値よりも低い(小さい)場合、カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こせるのに先立って、最大値(\$FFFF)へそして次に\$0000から始める計数をしなければなりません。多くの場合でこの特性は好ましくありません。

標準動作と同じように、**タイマ/カウンタ溢れ割り込み要求(TOV1)フラグ**はカウンタがMAXから\$0000へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

## 11.7. タイマ/カウンタのタイミング図

このタイマ/カウンタは同期設計で、従ってタイマ/カウンタクロック( $clk_{T1}$ )が下図のクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時の情報を含みます。

図11-6.は各種動作形態でのTOP近辺の計数の流れを示します。

図11-6. タイマ/カウンタ タイミング図 (前置分周器なし)

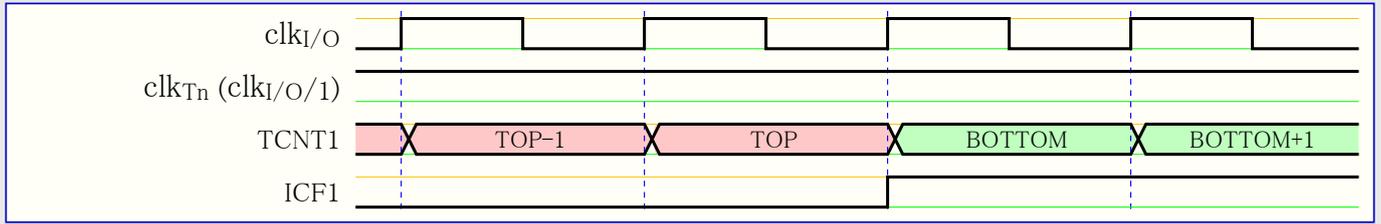
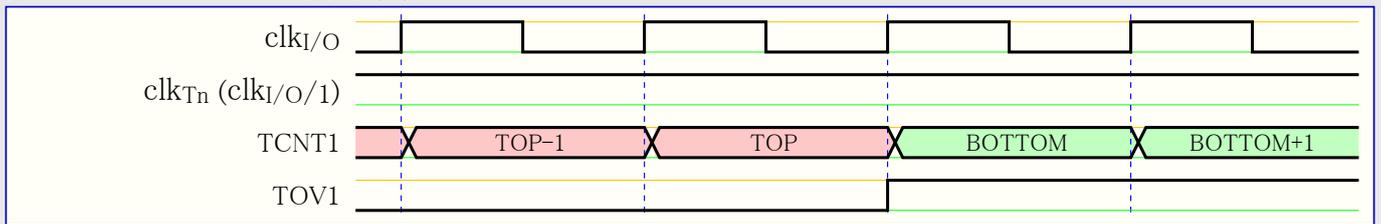


図11-7.は各種動作形態でのMAX近辺の計数の流れを示します。

図11-7. タイマ/カウンタ タイミング図 (前置分周器なし)



## 11.8. 16ビット タイマ/カウンタ1用レジスタ

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

ビット (\$8A)	7	6	5	4	3	2	1	0	TCCR1B
	ICNC1	ICES1	-	WGM13	-	CS12	CS11	CS10	
Read/Write	R/W	R/W	R	R/W	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット7 – ICNC1 : 捕獲起動入力1雑音消去許可 (Input Capture1 Noise Canceler)**

このビットを(1)に設定することが捕獲起動入力雑音消去器を活性(有効)にします。雑音消去器が有効にされると、捕獲起動入力(ICP1)ピンからの入力が濾波されます。この濾波器機能はそれが出力を更新するのに、連続4回等しく評価されたICP1ピンの採取を必要とします。雑音消去器が許可されると、捕獲入力はそれによって4発振器(システムクロック)周期遅らされます。

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

このビットは出来事での捕獲を起動するのに使われる捕獲起動入力(ICP1)ピンのどちらかの端(エッジ)を選びます。ICES1ビットが0を書かれると起動動作として下降(負)端が使われ、ICES1ビットが1を書かれると上昇(正)端が捕獲を起動します。

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

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

- **ビット5 – Res : 予約 (Reserved)**

このビットは将来の使用に対して予約されています。

- **ビット4 – WGM13 : 波形生成種別 (Waveform Generation Mode)**

動作形態定義については右表をご覧ください。

表11-2. 波形生成種別選択

番号	WGM13	タイマ/カウンタ動作種別	TOP値	TOV1設定時
0	0	標準動作	\$FFFF	MAX
12	1	比較一致タイマ/カウンタ解除(CTC)動作	ICR1	MAX

- **ビット3 – Res : 予約 (Reserved)**

このビットは将来の使用に対して予約されています。

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

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

タイマ/カウンタ1に対して外部ピン(クロック)動作が使われる場合、例えばT1ピンが出力として設定されても、T1ピンの遷移はカウンタをクロック駆動します。この特性はソフトウェアに計数制御を許します。

表11-3. タイマ/カウンタ1入力クロック選択

CS12	CS11	CS10	意味
0	0	0	停止 (タイマ/カウンタ1動作停止)
0	0	1	clkI/O (前置分周なし)
0	1	0	(予約)
0	1	1	(予約)
1	0	0	(予約)
1	0	1	(予約)
1	1	0	T1ピンの下降端 (外部クロック)
1	1	1	T1ピンの上昇端 (外部クロック)

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

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

この2つのタイマ/カウンタ I/O位置(TCNT1HとTCNT1Lを合わせたTCNT1)は、読み書き両方についてタイマ/カウンタ部の16ビットカウンタに直接アクセスします。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。54頁の「16ビットレジスタのアクセス」をご覧ください。

カウンタが走行中にカウンタ(TCNT1)を変更することはICR1とTCNT1間の比較一致消失の危険を誘発します。

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

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

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

この捕獲レジスタはICP1ピン(またはタイマ/カウンタ1については任意のアナログ比較器1出力)で出来事が起こる毎にカウンタ(TCNT1)値で更新されます。この捕獲レジスタはタイマ/カウンタのTOP値を定義するのに使えます。

この捕獲レジスタは容量が16ビットです。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読まれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。54頁の「16ビットレジスタのアクセス」をご覧ください。

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

ビット	7	6	5	4	3	2	1	0	
\$01 (\$21)	–	–	ICIE1	–	–	–	–	TOIE1	TIMSK1
Read/Write	R	R	R/W	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,6,4~1 – Res : 予約 (Reserved)

これらのビットは未使用ビットで、常に0として読みます。

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

このビットが1を書かれ、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)されると、タイマ/カウンタ1捕獲割り込みが許可されます。タイマ/カウンタ1割り込み要求フラグレジスタ(TIFR1)に配置された捕獲割り込み要求フラグ(ICF1)が設定(1)されると、対応する割り込みベクタ(38頁の表8-1.参照)が実行されます。

- ビット0 – TOIE1 : タイマ/カウンタ1溢れ割り込み許可 (Timer/Counter1 Overflow Interrupt Enable)

このビットが1を書かれ、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)されると、タイマ/カウンタ1溢れ割り込みが許可されます。タイマ/カウンタ1割り込み要求フラグレジスタ(TIFR1)に配置されたタイマ/カウンタ1溢れ割り込み要求フラグ(TOV1)が設定(1)されると、対応する割り込みベクタ(38頁の表8-1.参照)が実行されます。

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

ビット	7	6	5	4	3	2	1	0	
\$02 (\$22)	–	–	ICF1	–	–	–	–	TOV1	TIFR1
Read/Write	R	R	R/W	R	R	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,6,4~1 – Res : 予約 (Reserved)

これらのビットは未使用ビットで、常に0として読みます。

- ビット5 – ICF1 : タイマ/カウンタ1捕獲割り込み要求フラグ (Timer/Counter1, Input Capture Flag)

ICP1ピンに捕獲の事象が起こると、このフラグが設定(1)されます。捕獲レジスタ(ICR1)がWGM13によってTOP値として設定されると、ICF1フラグはカウンタがTOP値に到達する時に設定(1)されます。

捕獲割り込みベクタが実行されると、ICF1は自動的に解除(0)されます。代わりにこのビット位置へ論理1を書くことによってもICF1は解除(0)できます。

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

このフラグの設定(1)はWGM13ビット設定に依存します。

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

## 12. 電力段制御器 (PSCn)

この電力段制御器は高性能波形制御器です。

Atmel 90PWM81/161は1つのPSC2部を含みます。

### 12.1. 特徴

- PWM波形生成機能 (設定可能な2つの相互補完出力)
- 沈黙時間制御
- 12ビット分解能までの標準動作
- 分解能増強動作(12+4ビット)付きの周波数とパルス
- 64MHzまでの周波数
- 外部事象での条件付き波形 (ゼロ交差、電流感応...)
- チップ上の全電力段制御器(PSC)同期動作
- デジタル遅延レジスタ付きA/D変換器同期動作
- 入力無効化
- 過負荷保護機能
- 異常保護機能、全出力をLowレベルに強制する緊急入力
- 中央整列と端(エッジ)整列の同期動作
- ハードウェアによる高速緊急停止

### 12.2. 概要

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'は電力段制御器(PSC)番号の置き換えで、この場合は2です。けれどもプログラムでレジスタまたはビット定義に使う時は(例えば電力段制御器2(PSC2)同期/出力設定レジスタへのアクセスのPSOC2のように)正確な形式が使われなければなりません。
- 小文字の'x'は電力段制御器(PSC)部分の置き換えで、この場合はAまたはBです。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えば障害/再起動を設定する電力段正制御器2Aの入力制御レジスタに対するアクセスのPFRC2Aのように)。

電力段制御器(PSC)の目的は基板上の電力単位部を制御することです。これにはPSC2に4出力があります。

これらの出力は色々な方法で使えます。

- 単一電力トランジスタを駆動する'1出力' (DC/DC変換器、力率改善(PFC)、DC電動機...)
- 半ブリッジを駆動する'2出力' (調光、DC電動機...)
- 全ブリッジを駆動する'4出力' (調光、DC電動機...)

各PSCには生成波形の直接的な動きの提供を予定する目的の2つの入力があります。

- 電流感応安定化
- 0交差再起動
- 消磁再起動
- 障害入力

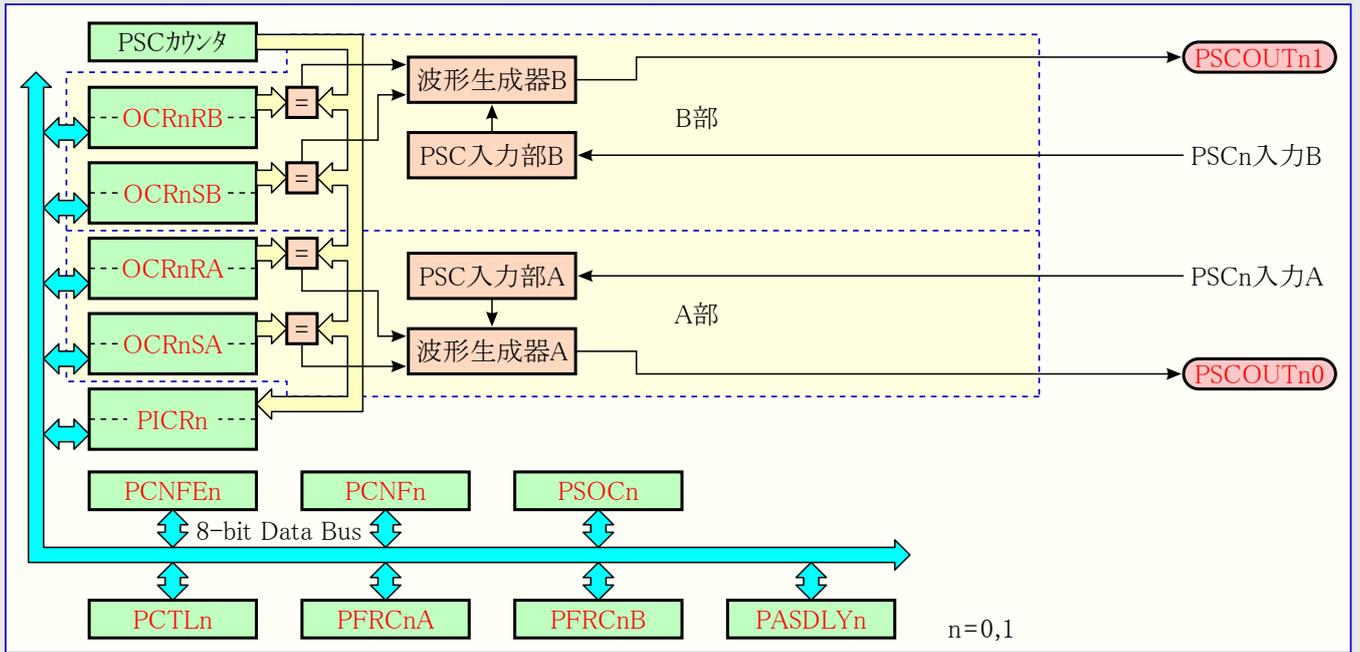
PSCは3つの半ブリッジ駆動設定を提供するための連結と同期ができます。この機能のため、非同期やブラシレスDC(BLDC)電動機のような応用に対して3相波形を生成することが可能です。(訳補:本記述は3つのPSC部を持つデバイスが対象です。)

31頁の「PRR – 電力削減レジスタ」でのPRPSC2ビットはPSC2部を許可するために0を書かれなければなりません。

(訳補) 以降で説明されるPSC動作は基本部分に関して一般的なAVRのタイマ/カウンタを使ったPWMと同様に考えられます。大きく異なるのは出力変移点を指定するための比較レジスタが4つあることです。これらのレジスタの基本的な使用方法は、2つを1組にして相互補完出力の一方を制御し、他の1組でもう一方の出力を制御します。各組内のレジスタの1つは出力のON遷移点を、他方はOFF遷移点を指定します。以降で記述される比較SA,RA,SB,RBの各レジスタが、これらに該当します。この1組の比較レジスタでの波形生成部が、各々A部、B部として記述されています。従って比較SAレジスタはA部セット(Set)点用比較レジスタ、比較RAレジスタはA部リセット(Reset)点用比較レジスタを意味します。

12.3. 電力段制御器(PSC)説明

図12-1. PSC0,1構成図 (訳注:本デバイスには存在しません。)

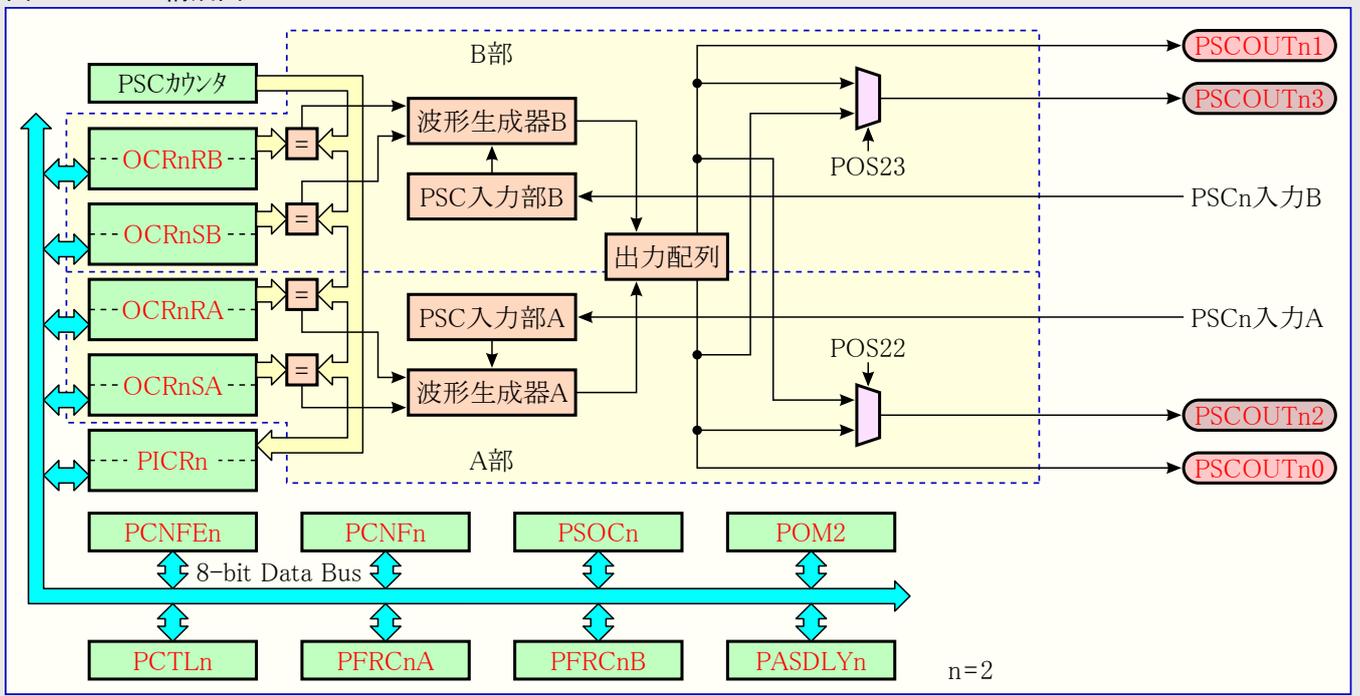


PSCの原理はカウンタ(PSCカウンタ)の使用を基にします。このカウンタは選んだ走行動作に対応するレジスタ内に格納された値からと値へ上昇計数と下降計数ができます。

PSCは2つの対象物のように見えます。PSCOUTn0出力を生成する1つの部分はA部と名付けられ、PSCOUTn1出力を生成する2つの部分はB部と名付けられます。

A部とB部の各々には選んだ入力を管理するためのPSC入力部があります。

図12-2. PSC2構成図



PSC2にはPSCOUT22とPSCOUT23の2つの補助出力があります。1つ目の選択器(POS22)によって、PSCOUT22はPSCOUT20またはPSCOUT21を二重化できます。2つ目の選択器(POS23)によって、PSCOUT23はPSCOUT20またはPSCOUT21を二重化できます。

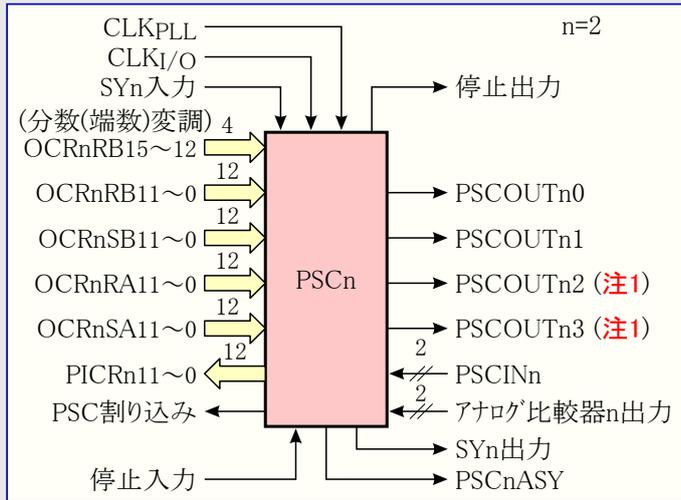
出力配列はPSC手順に対応した出力値設定の可能性を与える2×2種類の参照表です(77頁の「出力配列」をご覧ください)。

12.3.1. 出力極性

PSCの極性('High有効'または'Low有効')は設定可能です。以降の例のタイミング図は'High有効'極性で与えられます。

## 12.4. 信号説明

図12-3. PSCn外部接続構成図



注1: PSC2のみ利用可能。

### 12.4.1. 入力説明

表12-1. 内部入力

名称	説明	形式/ビット数
OCRnRB15~12	周波数分解能増強値 (小数分周器)	4ビットレジスタ
OCRnRB11~0	B部でのリセット信号比較値 (PSCOUTn1)	12ビットレジスタ
OCRnSB11~0	B部でのセット信号比較値 (PSCOUTn1)	12ビットレジスタ
OCRnRA11~0	A部でのリセット信号比較値 (PSCOUTn0)	12ビットレジスタ
OCRnSA11~0	A部でのセット信号比較値 (PSCOUTn0)	12ビットレジスタ
CLKI/O	I/Oクロックからのクロック入力	信号
CLKPLL	PLLからのクロック入力	信号
SYn入力	隣接PSCからの同期入力 (78頁の図12-41.参照)	信号
停止入力	同期動作に対する停止入力	信号

表12-2. PSC部外入力

名称	説明	形式/ビット数
PSCINn	障害/再起動機能に使われる入力0	信号
アナログ比較器1出力	障害/再起動機能に使われる入力1	信号
PSCINnA	障害/再起動機能に使われる入力2	信号
アナログ比較器2出力	障害/再起動機能に使われる入力3	信号

### 12.4.2. 出力説明

表12-3. PSC部出力

名称	説明	形式/ビット数
PSCOUTn0	PSCのA部からのPSCn出力0	信号
PSCOUTn1	PSCのB部からのPSCn出力1	信号
PSCOUTn2	PSCのA部またはB部からのPSCn出力2 (PSC2のみ)	信号
PSCOUTn3	PSCのA部またはB部からのPSCn出力3 (PSC2のみ)	信号

表12-4. 内部出力

名称	説明	形式/ビット数
PICRn11~0	PSCn捕獲入力レジスタ (再起動事象でのカウンタ値)	12ビットレジスタ
PSCn割り込み	PSC割り込み要求 (溢れ、障害、捕獲発生の3種)	信号
PSCnASY	A/D変換器(+増幅器)同期 (78頁の「アナログ同期化」参照)	信号
SYn出力	隣接PSCへの同期出力 (78頁の図12-41.参照)	信号
停止出力	同期動作に対する停止出力	信号

## 12.5. 機能説明

### 12.5.1. 波形周期

電力段制御器(PSC)によって生成した波形は2つの波形の順次連続として説明できます。

1つ目の波形はPSCのA部とPSCOUTn0出力に関係します。この波形部分は図12-4.での補助周期Aです。

2つ目の波形はPSCのB部とPSCOUTn1出力に関係します。この波形部分は図12-4.での補助周期Bです。

(周期の)全体波形は補助周期Bの最後で終了されます。これは波形Bの最後での意味です。

傾斜は波形生成器内に含んだPSCカウンタの出力を表します。中央動作は下降/上昇計数する1傾斜動作のようです。新規値群の更新が傾斜動作に拘らず、(PSC周期内の)最終傾斜の頂上(TOP)で行なわれることに注意してください。

図12-4. 1,2,4傾斜動作での周期表現図

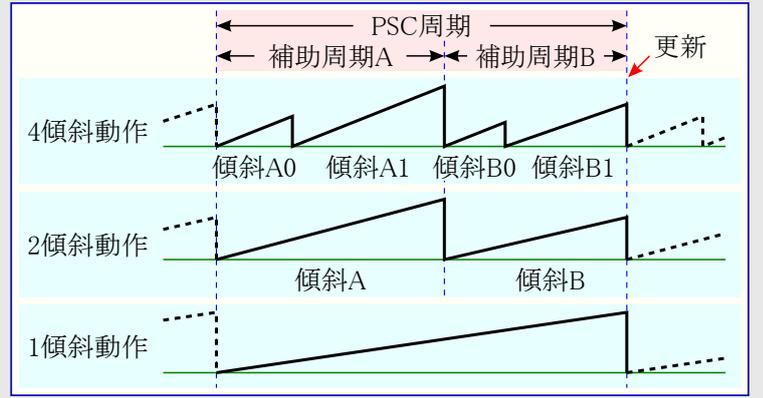
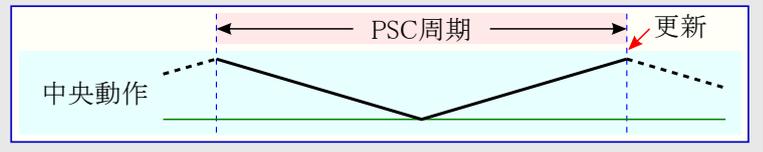


図12-5. 中央整列動作での周期表現図



### 12.5.2. 走行動作説明

出力信号の波形と長さは走行動作と時間値(DT0,OT0,DT1,OT1)によって決められます。次の4つの動作が可能です。

- 4傾斜動作
- 2傾斜動作
- 1傾斜動作
- 中央整列動作

#### 12.5.2.1. 4傾斜動作

4傾斜動作での周期内の各時間は各々自身の定義を持ちます。

PSCの入力クロックはCLKPSCによって与えられます。

PSCOUTn0とPSCOUTn1信号はON時間0(OT0)、沈黙時間0(DT0)、ON時間1(OT1)、沈黙時間1(DT1)の値によって定義されます。

$$\text{ON時間0} = \text{OCRnRA} \times 1/f_{\text{CLKPSC}}$$

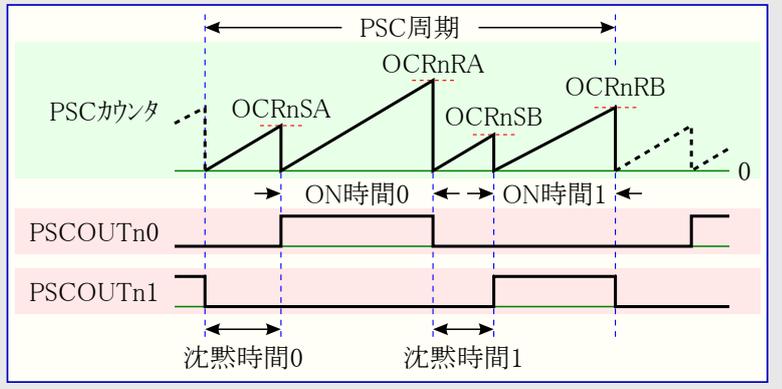
$$\text{ON時間1} = \text{OCRnRB} \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間0} = (\text{OCRnSA} + 2) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間1} = (\text{OCRnSB} + 2) \times 1/f_{\text{CLKPSC}}$$

注: 沈黙時間0と沈黙時間1の最小値は $2 \times 1/f_{\text{CLKPSC}}$ です。

図12-6. 4傾斜動作でのPSCn0とPSCn1基本波形



#### 12.5.2.2. 2傾斜動作

2傾斜動作での全体周期は瞬時区間時間を与えるON時間0でのPSCn0記述に対する1瞬時区間と、瞬時区間時間を与えるON時間1でのPSCn1記述に対する1瞬時区間の2つの瞬時区間に分けられます。

PSCOUTn0とPSCOUTn1信号はON時間0(OT0)、沈黙時間0(DT0)、ON時間1(OT1)、沈黙時間1(DT1)の値によって定義されます。

$$\text{ON時間0} = (\text{OCRnRA} - \text{OCRnSA}) \times 1/f_{\text{CLKPSC}}$$

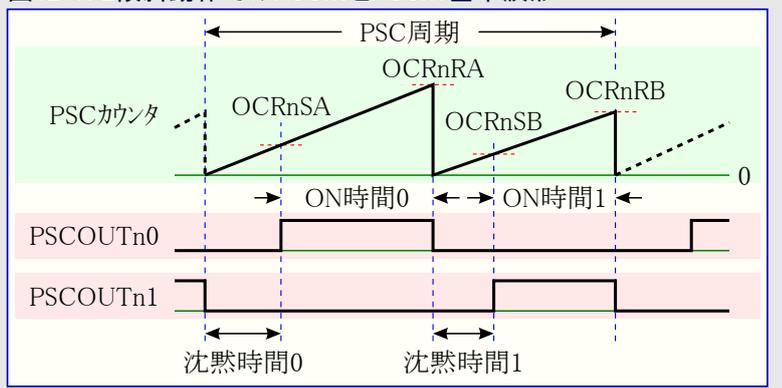
$$\text{ON時間1} = (\text{OCRnRB} - \text{OCRnSB}) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間0} = (\text{OCRnSA} + 1) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間1} = (\text{OCRnSB} + 1) \times 1/f_{\text{CLKPSC}}$$

注: 沈黙時間0と沈黙時間1の最小値は $1/f_{\text{CLKPSC}}$ です。

図12-7. 2傾斜動作でのPSCn0とPSCn1基本波形



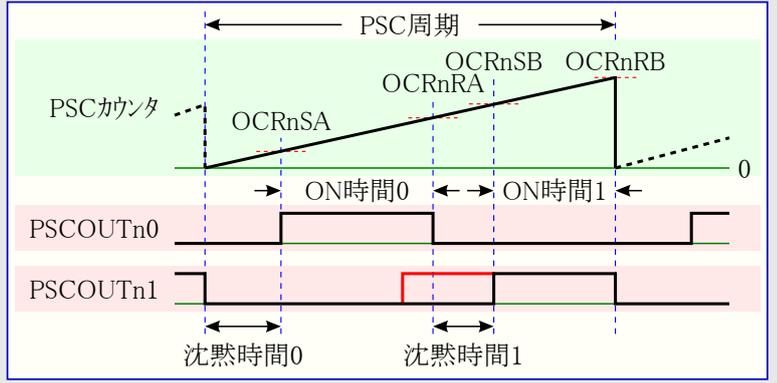
### 12.5.2.3. 1傾斜動作

1傾斜動作でのPSCOUTn0とPSCOUTn1出力はお互いに重複できません(訳補:沈黙時間1が負の場合、図の赤線)。

$$\begin{aligned} \text{ON時間0} &= (\text{OCRnRA} - \text{OCRnSA}) \times 1/f_{\text{CLKPSC}} \\ \text{ON時間1} &= (\text{OCRnRB} - \text{OCRnSB}) \times 1/f_{\text{CLKPSC}} \\ \text{沈黙時間0} &= (\text{OCRnSA} + 1) \times 1/f_{\text{CLKPSC}} \\ \text{沈黙時間1} &= (\text{OCRnSB} - \text{OCRnRA}) \times 1/f_{\text{CLKPSC}} \end{aligned}$$

注: 沈黙時間0の最小値は $1/f_{\text{CLKPSC}}$ です。

図12-8. 1傾斜動作でのPSCn0とPSCn1基本波形



### 12.5.2.4. 中央整列動作

中央整列傾斜動作でのPSCOUTn0とPSCOUTn1信号の中心は中央にされます。

$$\begin{aligned} \text{ON時間0} &= 2 \times \text{OCRnSA} \times 1/f_{\text{CLKPSC}} \\ \text{ON時間1} &= 2 \times (\text{OCRnRB} - \text{OCRnSB}) \times 1/f_{\text{CLKPSC}} \\ \text{沈黙時間} &= (\text{OCRnSB} - \text{OCRnSA}) \times 1/f_{\text{CLKPSC}} \\ \text{PSC周期} &= 2 \times (\text{OCRnRB} + 1) \times 1/f_{\text{CLKPSC}} \end{aligned}$$

注: PSC周期の最小値は $2 \times 1/f_{\text{CLKPSC}}$ です。

OCRnRAはPSC出力波形タイミング制御に使われません。けれどもA/D変換同期調節に有用です(78頁の「アナログ同期化」をご覧ください)。

図12-9. 中央整列動作でのPSCn0とPSCn1基本波形

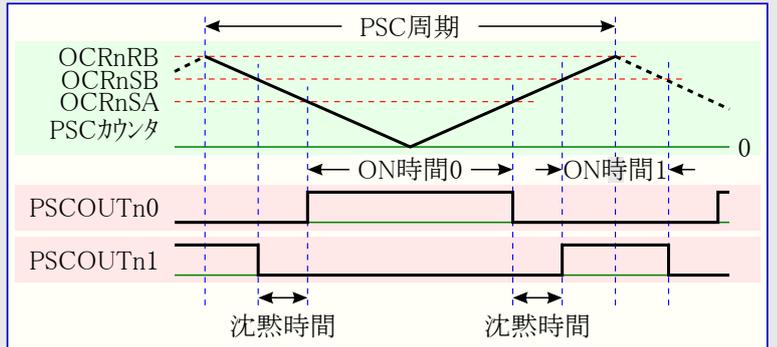
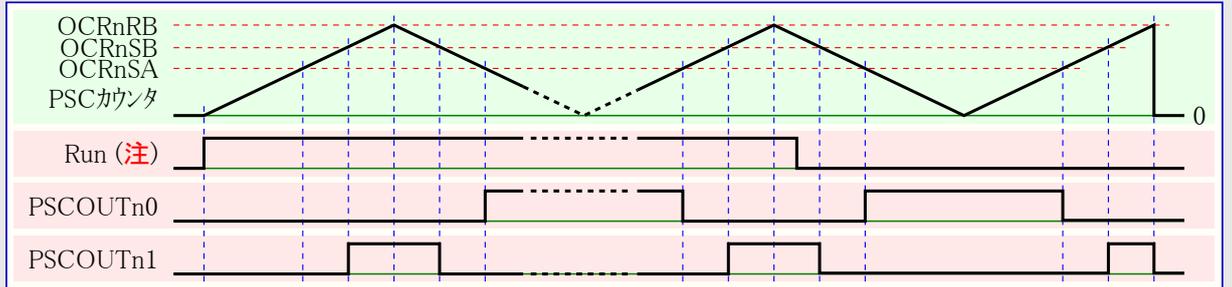


図12-10. 中央整列動作での走行と停止機構



注: 84頁の「PCTL2 - PSC2制御レジスタ」をご覧ください。

### 12.5.3. 50%波形設定

PSCOUTn0とPSCOUTn1が同じ特性を持つ時に50%動作(PCNFnレジスタのPFIFTYn=1)でPSCを設定することが可能です。PSCがこの動作の時にOCRnSBH, OCRnSBL, OCRnRBH, OCRnRBLレジスタをOCRnSAH, OCRnSAL, OCRnRAH, OCRnRALレジスタに複写します。故にOCRnSAH, OCRnSAL, OCRnRAH, OCRnRALレジスタを設定する必要はありません。

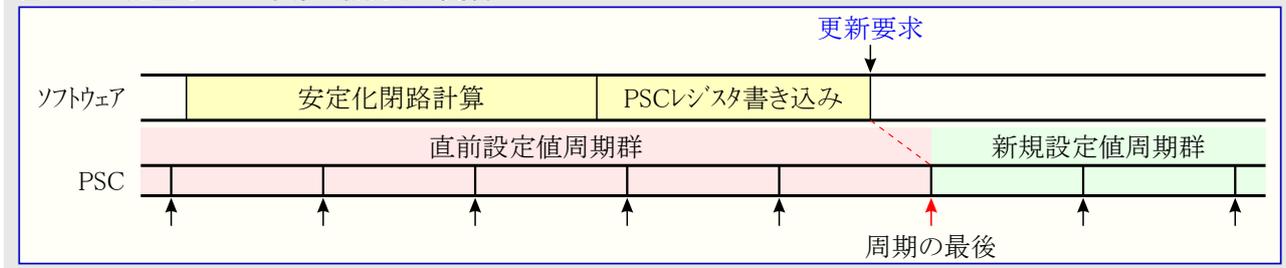
## 12.6. 値の更新

PSC波形レジスタ更新は以下の方法で行われます。

- PSC停止直後
- PSC走行時のPSC周期の最後
- 格納または自動格納の形態使用時に必要とされる条件に後続するPSC周期の最後

周期での非同時で矛盾する値を避けるため、各種値の1つの更新が必要な場合、PSCによって周期の終わりで同時に全ての値を更新することができます。新規値群はソフトウェアによって計算され、更新はソフトウェアによって開始されます。

図12-11. 完全なPSC周期の最後での更新



ソフトウェアは値更新を終える前に周期を停止し、新規PSC周期を再び開始できます。

### 12.6.1. 値の同期更新

新規タイミング値やPSC出力構成はPSC周期中に書けます。LOCK(PLOCKn)とAUTOLOCK(PALOCKn)構成ビットによって新規値群全体は以下の条件で考慮され(扱われ)ます。

- AUTOLOCK構成ビットが選択(=1)されると、PSC内部レジスタの更新は比較RBLレジスタ書き込みに後続するPSC周期の最後で行なわれます。AUTOLOCK構成ビットはこの最初のPSC周期の最後で考慮(処理)されます。
- LOCK構成ビットが設定(1)されると、PSC内部レジスタ更新はLOCKビットが0に開放された場合のPSC周期の最後で行なわれます。

LOCKとAUTOLOCKで同期更新されるレジスタはPSCn同期/出力構成(PSOCn)レジスタ、PSC2出力配列(POM2)レジスタ、PSCn比較SA(OCRnSAH/L)レジスタ、PSCn比較RA(OCRnRAH/L)レジスタ、PSCn比較SB(OCRnSBH/L)レジスタ、PSCn比較RB(OCRnRBH/L)レジスタです。

設定(1)されると、AUTOLOCK設定ビットはLOCK設定ビットに勝ります。

82頁の「PCNFn – PSCn構成レジスタ」をご覧ください。

## 12.7. 増強分解能

調光安定器応用は50/60Hzに下げる増強分解能が必要です。標準分解能を増強する方法は(小数分周器とも呼ばれる)分数(端数)変調に基きます。周回は16周期枠に集団化されます。周期は小数分周器の数値で与えられた順次手順によって変調されます。結果の出力周波数は枠内周波数の平均です。小数分周器( $d$ )はOCRnRB15~12によって与えられます。

PSC出力周期はPSCOUTn0のON時間+沈黙時間(OT0+DT0)とPSCOUTn1のON時間+沈黙時間(OT1+DT1)に直接匹敵します。これらの値は12ビット数です。周波数調節は専用計数器のような段階で行なえるだけです。段階幅は隣接する2つのPSC周波数間の周波数差として定義されます。

RB, RB+RA, SB, SB+SAで分数変調(FWM:Flank Width Modulation)を適用することが可能です。この選択は平衡分数変調動作形態(PBFMn1とPBFMn0)ビットで行われます。

傾斜形態と(PBFMn1,0によって定義される)増強分解能形態に従って周波数差は3つの異なる値を取り得ます。

$$\Delta f=0$$

$$\Delta f1=|f1-f2|= \left| \frac{f_{PSC}}{k} - \frac{f_{PSC}}{k+1} \right| = f_{PSC} \times \frac{1}{k(k+1)}$$

$$\Delta f2=|f1-f2|= \left| \frac{f_{PSC}}{k} - \frac{f_{PSC}}{k+2} \right| = f_{PSC} \times \frac{2}{k(k+2)}$$

$k$ はPSC周期内のCLKPSC周期数で、次式によって与えられます。

$$k = \frac{f_{PSC}}{f_{OP}}$$

$f_{OP}$ は出力動作周波数です。

例えば $f_{PLL}=64\text{MHz}$ 、最大動作周波数160kHzの標準動作での $k$ は400に相当します。結果の分解能は $\Delta f$ で、 $64\text{MHz}/400/401=400\text{Hz}$ になります。

増強動作での出力周波数は連続する16周期によって形成された枠の平均です。  
 $f_{b1}$ と $f_{b2}$ は隣接する2つの基本周波数です。

$$f_{AVERAGE} = \frac{16-d}{16} \times f_{b1} + \frac{d}{16} \times f_{b2}$$

周波数分解能は16で割られます。上の例での分解能は25Hzに等しくなります。

$$f_{AVERAGE} = \frac{16-d}{16} \times \frac{f_{PLL}}{k} + \frac{d}{16} \times \frac{f_{PLL}}{k+1}$$

傾斜形態と(PBFMn1,0によって定義される)増強分解能形態に従い、平均周波数偏差は以下のように3つの異なる値を取り得ます。

$$\begin{aligned} \Delta f(AVERAGE) &= 0 \\ \Delta f1(AVERAGE) &= f_{PSC} \times \frac{d}{16k(k+1)} \\ \Delta f2(AVERAGE) &= f_{PSC} \times \frac{d}{8k(k+2)} \end{aligned}$$

これらの値は表12-5.に対するように走行形態と増強分解能形態に従って適用されます。

1傾斜と2傾斜の形態では、定周波数を保つと同時にパルス幅だけに分数変調(FWM)の適用が可能なことに注意しなければなりません。

表12-5. 分数変調での周波数偏差

走行形態	PBFMn1,0			
	0 0	0 1	1 0	1 1
	RB	RB+RA	SB	SB+SA
4傾斜	$\Delta f1$	$\Delta f2$	$\Delta f1$	$\Delta f2$
2傾斜	$\Delta f1$	$\Delta f2$	0 (注)	0
1傾斜	$\Delta f1$	$\Delta f1$	0	0
中央整列	$\Delta f2$	$\Delta f2$	$\Delta f2$	$\Delta f2$

注: 変調はパルス幅です。

### 12.7.1. 周波数分配

周波数変調は16連続周期の枠内で2つの周波数切り替えによって行なわれます。これら2つの周波数は $f_{b1}$ と $f_{b2}$ で、 $f_{b1}$ は欲する周波数よりも高い最も近い基本周波数、 $f_{b2}$ は欲する周波数よりも低い最も近い基本周波数です。枠内の $f_{b1}$ の数は $16-d$ で、 $f_{b2}$ の数は $d$ です。 $f_{b1}$ と $f_{b2}$ 周波数は予め定義された様式に従って均等に配給されます。この様式は次表で与えられるように、または等価的に均等な配給を与える何れかの他の実装によって行なえます。

第15周期(表12-6.の番号での14)で割り込みを生成することができます。これはPSCn増強周期最後で割り込み許可(PEOPEN)ビットが設定(1)されている場合です。これは以下を許します。

- 新規増強周期開始でだけの変調変更
- ソフトウェアによる増強変調精度拡張

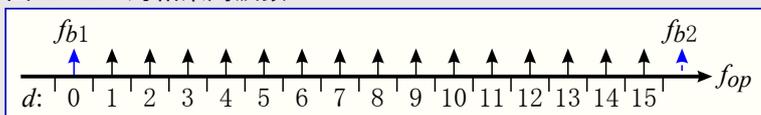
本表内の□の間、対応周期の $f_{b1}$ を $f_{b2}$ が満たします。従って各行について、 $f_{b2}$ 数が $f_{b1}$ に取って代わります。

表12-6. 変調された枠内の $f_{b2}$ 配給

小数分周器 (d)	枠内のPSC周期															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
1	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
2	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
3	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
4	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
5	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
6	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
7	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
8	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
9	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
10	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
11	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
12	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
13	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
14	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
15	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□

(訳注) 本図を含め全体的に $f_{b1}$ と $f_{b2}$ 間の周波数高低関係が統一されていません(矛盾します)。おそらく本図の高低関係が正しく、直前の説明の2周波数関係が逆転しているものと思われます。

図12-12. d対結果周波数



## 12.7.2. 動作種別

### 12.7.2.1. 標準動作

最も簡単な動作種別が標準動作です。図12-6をご覧ください。

PSCOUTn0の活性時間はON時間0(OT0)値で与えられます。PSCOUTn1の活性時間はON時間1(OT1)値で与えられます。沈黙時間0(DT0)と沈黙時間1(DT1)はPSCOUTn0とPSCOUTn1活性信号間の沈黙を調節します。

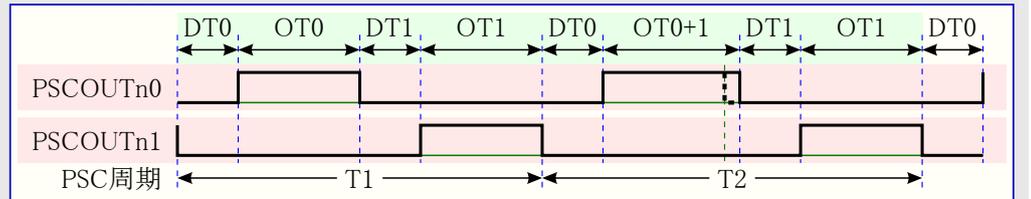
波形周波数は次式によって定義されます。

$$f_{PSCn} = \frac{1}{PSCn周期} = \frac{f_{CLKPSCn}}{(OT0 + OT1 + DT0 + DT1)}$$

### 12.7.2.2. 増強動作

増強動作は直前に記述された高分解能生成法を使います。図12-13はPBFMn1,0=00の例を与えます。

図12-13. 増強動作タイミング図



f<sub>b2</sub>を生成するために計数する補助段階は小数分周器に従って枠内で必要とする間、PSCn0信号に追加されます。表12-6をご覧ください。

波形周波数は次式で定義されます。

$$f_{1PSCn} = \frac{1}{T1周期} = \frac{f_{CLKPSCn}}{(OT0 + OT1 + DT0 + DT1)}$$

$$f_{2PSCn} = \frac{1}{T2周期} = \frac{f_{CLKPSCn}}{(OT0 + OT1 + DT0 + DT1 + 1)}$$

dは小数分周器因数です。

$$f_{AVERAGE} = \frac{d}{16} \times f_{1PSCn} + \frac{16-d}{16} \times f_{2PSCn}$$

分数変調(FWM)は83頁の表12-15で定義されるようにPSC出力波形内の各種位置で適用することができます。

## 12.8. 電力段制御器(PSC)入力

PSCのA部とB部の各々は1つのPSCn内部入力を考慮するための自身の系統を持ちます。A部とB部の各々はPSCn入力A/B制御(PFRCnA,PFRCnB)レジスタ(85頁記述参照)とPSCn構成レジスタ(82頁の「PCNF2 - PSC2構成レジスタ」項をご覧ください)によって構成設定することができます。

PSC入力A部は図12-14で示されます。

PSCn入力A制御レジスタ(85頁の「PSCn入力A制御レジスタ(PFRCnA)」をご覧ください)に従って、PSCn入力Aは再起動または異常入力として動くことができます。

A部とB部の各々は84頁の表12-18と表12-19で定義されるように最大4つの信号によって起動することができます。

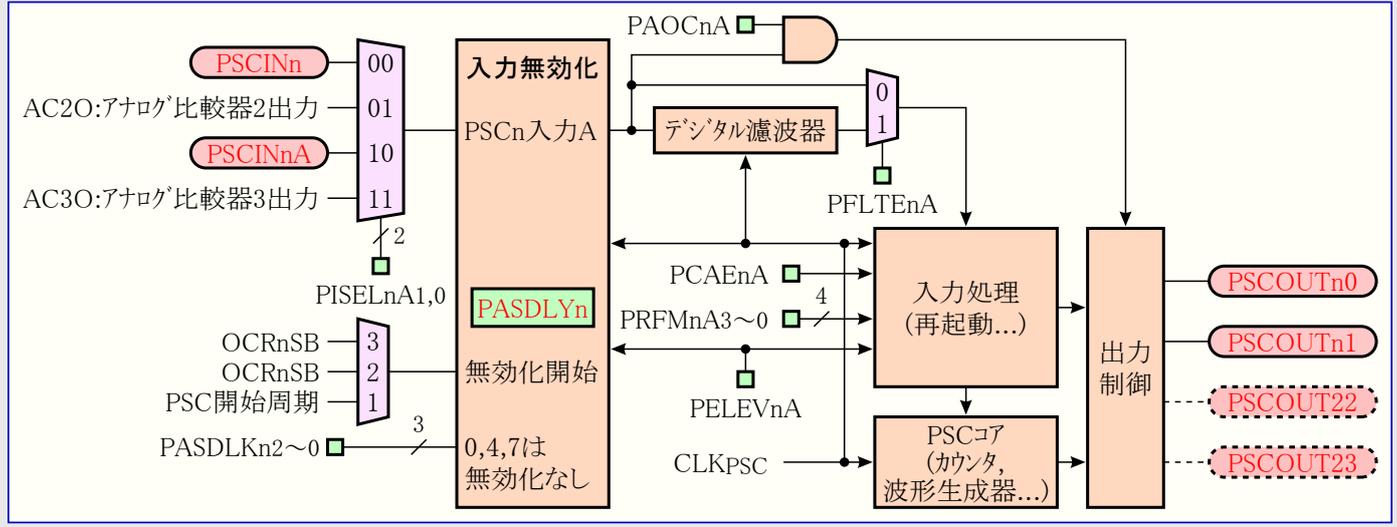
PSCのA部は或る区間の間にPSCn入力Aで表れるかもしれない望まない遷移の無効化を許す無効化部も持ちます。

無効化開始は83頁の表12-14でのように入力遅延選択(PASDLKn2~0)ビットによって定義されます。

無効化区間はアナログ同期遅延(PASDLYn)レジスタによって定義されます。対応するPASDLKn2~0ビットによって無効化が選ばれた場合、無効化開始から区間終了までに出現するかもしれない全ての遷移が無視されます。

無効化はレベル検知で、換言すると無効化窓内でパルスが開始され、窓後に未だ有効なレベルは有効な再起動事象を生成します。

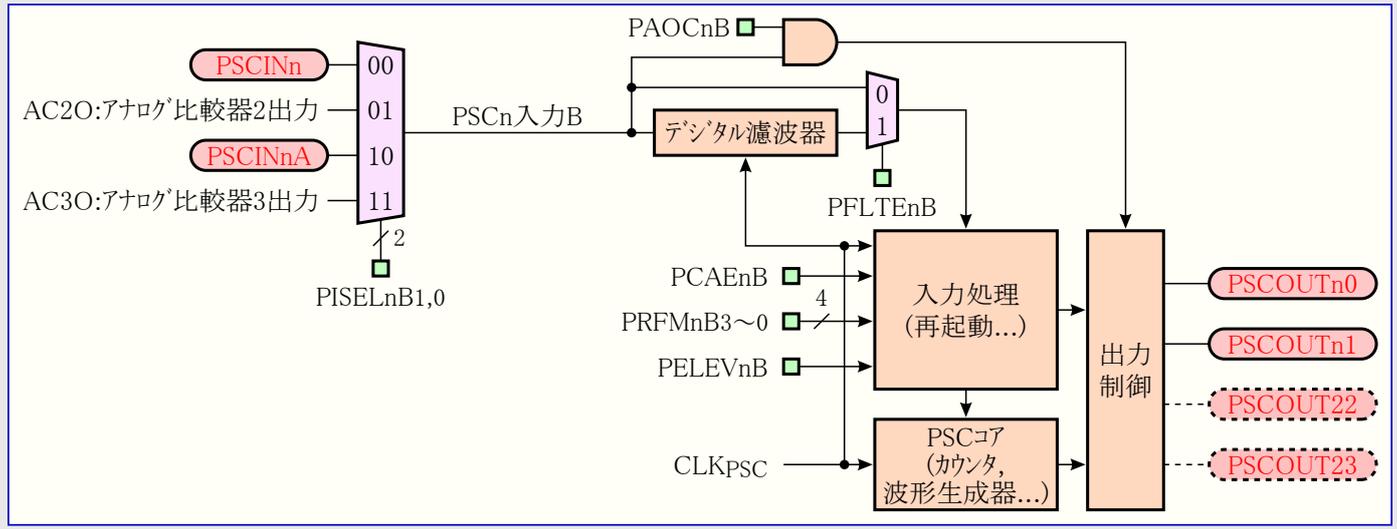
図12-14. PSC入力A部



PSC入力B部は図12-15.で示されます。

PSCn入力B制御レジスタ(85頁の「PFRCnB - PSCn入力B制御レジスタ」をご覧ください)に従って、PSCn入力Bは再起動または異常入力として動くことができます。

図12-15. PSC入力B部



### 12.8.1. PSC走行動作に対するPSC再起動の動き

中央整列動作での再起動入力は無効です。2傾斜動作または4傾斜動作での再起動入力AまたはBは、対応するAまたはB周期の終了と後続するBまたはA周期の開始を引き起こします。

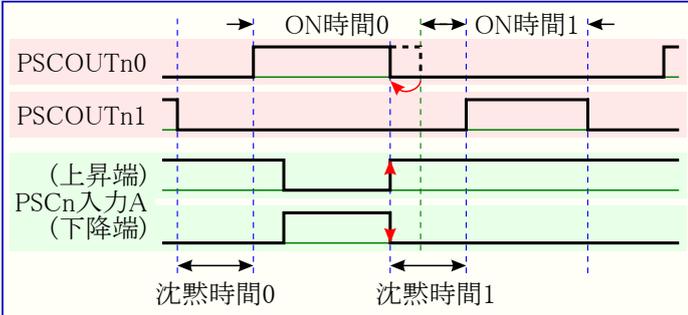
1傾斜動作での再起動入力AまたはBは現在のPSC計数を0にリセットします。

12.8.2. 外部事象でのPSCOUTn0再起動

PSCOUTn0出力はPSCn入力A(内部信号)の変更でのON時間0終了に先立ってリセットできます。PSCn入力Aは動作なし、レベルまたは端(エッジ)での動作に設定できます。PSCn入力Aの極性は感応制御部で設定可能です。PSCn入力AはPSCINn(外部)入力またはアナログ比較器nの出力にできます。

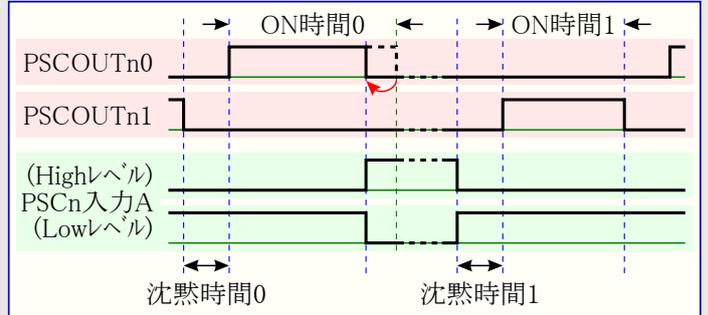
周期区間の減少のため、2つの出力周波数は一時的に増加します。

図12-16. PSCn入力Aで再起動されたPSCOUTn0 (端起動)



注: 本例は「2または4傾斜動作」での「入力動作8」で与えられます。詳細については図12-33をご覧ください。

図12-17. PSCn入力Aで再起動されたPSCOUTn0 (レベル動作)



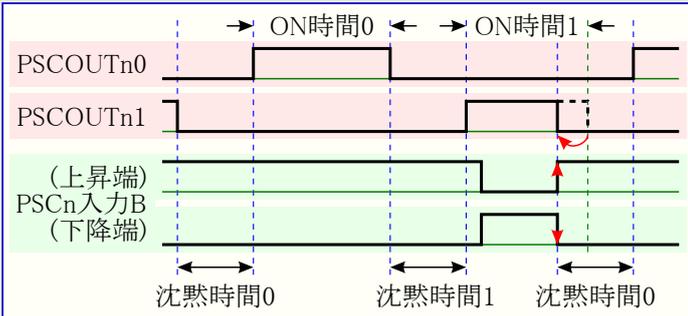
注: 本例は「2または4傾斜動作」での「入力動作1」で与えられます。詳細については図12-22をご覧ください。

12.8.3. 外部事象でのPSCOUTn1再起動

PSCOUTn1出力はPSCn入力B(内部信号)の変更でのON時間1終了に先立ってリセットできます。PSCn入力Bの極性は感応制御部で設定可能です。PSCn入力Bは動作なし、レベルまたは端(エッジ)での動作に設定できます。PSCn入力BはPSCINn(外部)入力またはアナログ比較器nの出力にできます。

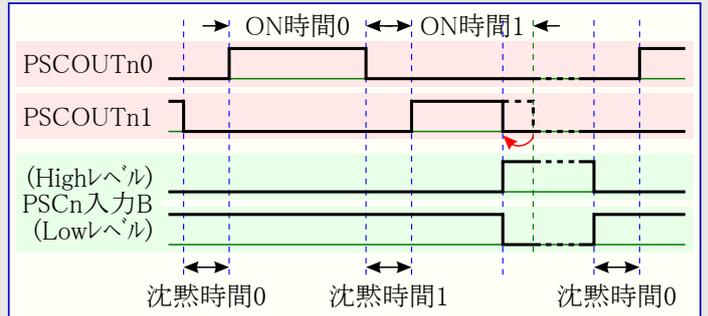
周期区間の減少のため、2つの出力周波数は一時的に増加します。

図12-18. PSCn入力Bで再起動されたPSCOUTn1 (端起動)



注: 本例は「2または4傾斜動作」での「入力動作8」で与えられます。詳細については図12-33をご覧ください。

図12-19. PSCn入力Bで再起動されたPSCOUTn1 (レベル動作)

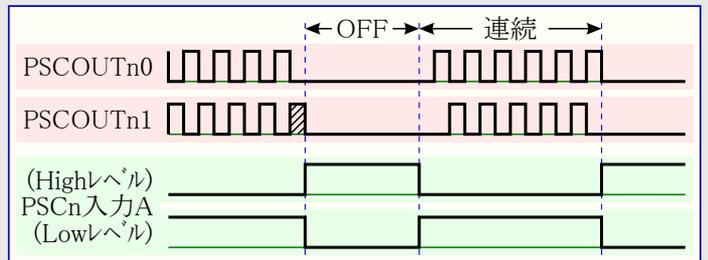


注: 本例は「2または4傾斜動作」での「入力動作1」で与えられます。詳細については図12-22をご覧ください。

12.8.3.1. 断続生成

注: レベル動作では入力動作3または4の使用によって断続生成にPSCを使うことが可能です(詳細については図12-26と図12-27をご覧ください)。

図12-20. レベル動作での断続生成



## 12.8.4. PSC入力設定

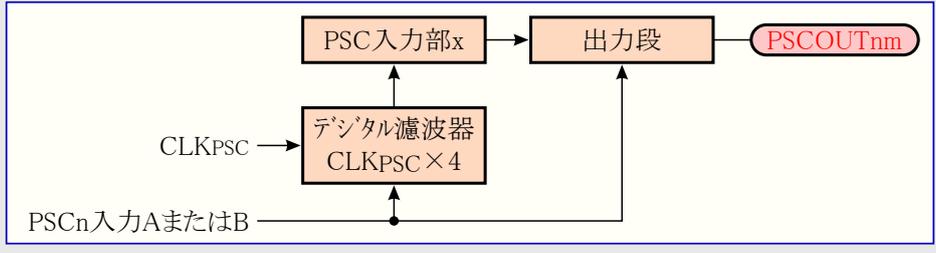
PSC入力設定は設定レジスタ内のビット設定によって行なわれます。

### 12.8.4.1. 濾波器許可

濾波器許可(PFLTE<sub>nx</sub>)ビットが設定(1)されると、4周期のデジタル濾波器が信号評価の前に挿入されます。本機能の禁止は雑音除去が大きすぎる遅延を与えることになる前置分周されたPSCクロックに対して主に必要とされます。

**重要:** デジタル濾波器が動作中の場合、出力を非活動にする(外部部品の緊急保護の)ために妨げられたPSCクロックでも、レベル感知は正しく動作します。同様に障害入力として使われる時にPSC<sub>n</sub>入力AやBはPSCOUT<sub>n</sub>0/1/2/3を動かすのにPSCを通さなければなりません。この経路はCLK<sub>PSC</sub>の走行を必要とします。故にPCTL<sub>n</sub>レジスタのPSC非同期出力制御(PAOC<sub>n</sub>A, PAOC<sub>n</sub>B)ビットによってPSCIN<sub>n</sub>入力は直接的にPSC出力を非活動にできます。この場合、入力はCLK<sub>PSC</sub>が走行すると直ぐに入力部によって通常として未だ考慮されます。

図12-21. PSC入力濾波器



### 12.8.4.2. 信号極性

有効端(エッジ動作)または有効レベル(レベル動作)が選べます。85頁の「PFRC<sub>n</sub>A - PSC<sub>n</sub>入力A制御レジスタ」項のPELEV<sub>nx</sub>ビット記述をご覧ください。

PELEV<sub>nx</sub>ビットが設定(1)なら、PSC<sub>n</sub>入力AまたはBの意味ある(有効)端が上昇(端動作)か、または活性(有効)レベルがHigh(レベル動作)で、逆(解除(0))については下降/Lowです。

- 2または4傾斜動作でのPSC<sub>n</sub>入力Aは沈黙時間0とON時間0の周期の間だけ考慮されます(PSC<sub>n</sub>入力Bに対しては各々沈黙時間1とON時間1)。
- 1傾斜動作でのPSC<sub>n</sub>入力AまたはPSC<sub>n</sub>入力Bは傾斜全体で動作します。

### 12.8.4.3. 入力動作の操作

4つの設定ビット(PRFM<sub>nx3~0</sub>)で、PSC入力の動作を定義することが可能です。これらの動作は表12-7.で一覧にされます。

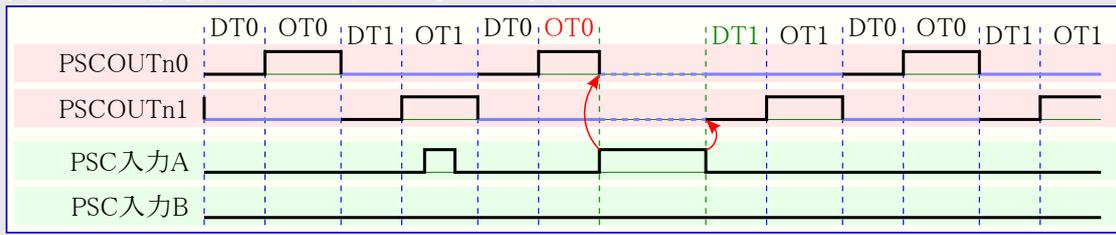
表12-7. PSC入力動作の動き (注: n=0,1,2, x=A,B)

番号	PRFM <sub>nx3~0</sub>	説明
0	0 0 0 0	PSC入力はPSC出力での動作なし。
1	0 0 0 1	73頁の「PSC入力動作1: 信号停止、反対側沈黙時間へ飛び、待機」をご覧ください。
2	0 0 1 0	73頁の「PSC入力動作2: 信号停止、反対側沈黙時間実行、待機」をご覧ください。
3	0 0 1 1	74頁の「PSC入力動作3: 信号停止、障害有効間反対側沈黙時間実行」をご覧ください。
4	0 1 0 0	74頁の「PSC入力動作4: タイミング変更なしで出力非活動」をご覧ください。
5	0 1 0 1	75頁の「PSC入力動作5: 信号停止、沈黙時間挿入」をご覧ください。
6	0 1 1 0	75頁の「PSC入力動作6: 信号停止、反対側沈黙時間へ飛び、待機」をご覧ください。
7	0 1 1 1	75頁の「PSC入力動作7: PSC停止、ソフトウェア操作待機」をご覧ください。
8	1 0 0 0	75頁の「PSC入力動作8: 端再起動PSC」をご覧ください。
9	1 0 0 1	76頁の「PSC入力動作9: 周波数固定端再起動PSC」をご覧ください。
10	1 0 1 0	(予約): 使用不可
11	1 0 1 1	
12	1 1 0 0	
13	1 1 0 1	76頁の「PSC入力動作14: 周波数固定端再起動PSC、出力非活動」をご覧ください。
14	1 1 1 0	
15	1 1 1 1	

注: 後続の全ての例は上昇端またはHighレベルの活性(有効)入力与えられます。

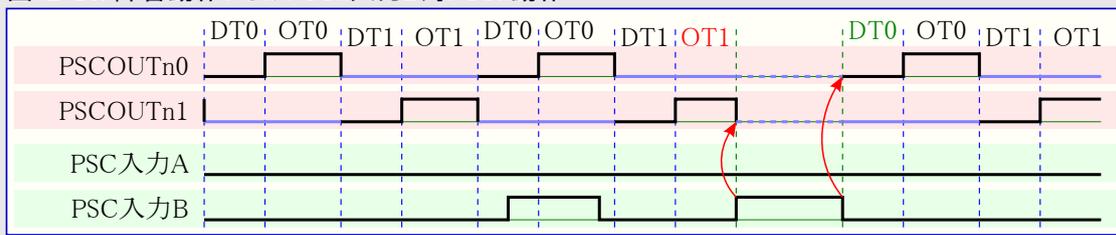
12.9. PSC入力動作1：信号停止、反対側沈黙時間へ飛び、待機

図12-22. 障害動作1でのPSCn入力A対PSCn動作



PSC入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSC入力A事象が起こると、PSCはPSCOUTn0を開放し、PSC入力Aの非活性(無効)状態に対して待ち、そしてDT1とOT1へ飛んで実行します。

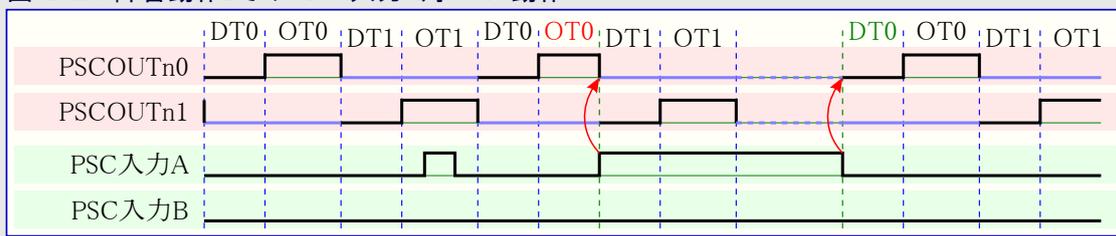
図12-23. 障害動作1でのPSCn入力B対PSCn動作



PSC入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSC入力B事象が起こると、PSCはPSCOUTn1を開放し、PSC入力Bの非活性(無効)状態に対して待ち、そしてDT0とOT0へ飛んで実行します。

12.10. PSC入力動作2：信号停止、反対側パルス沈黙時間実行、待機

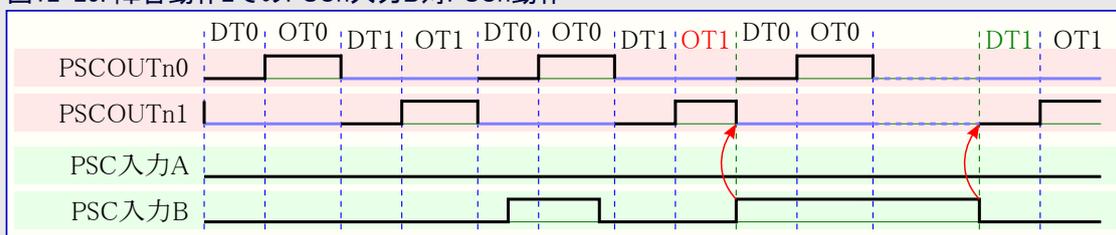
図12-24. 障害動作2でのPSCn入力A対PSCn動作



PSC入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSC入力A事象が起こると、PSCはPSCOUTn0を開放し、DT1とOT1へ飛んで実行し、そしてPSC入力Aの非活性(無効)状態を待ちます。

PSC入力AがDT1またはOT1中に開放されても、DT1とOT1の(PSC)半周期は常に全体が実行されます。

図12-25. 障害動作2でのPSCn入力B対PSCn動作

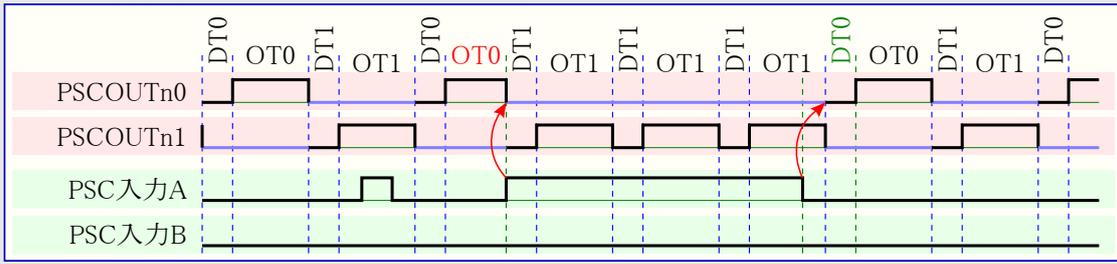


PSC入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSC入力B事象が起こると、PSCはPSCOUTn1を開放し、DT0とOT0へ飛んで実行し、そしてPSC入力Bの非活性(無効)状態を待ちます。

PSC入力BがDT0またはOT0中に開放されても、DT0とOT0の(PSC)半周期は常に全体が実行されます。

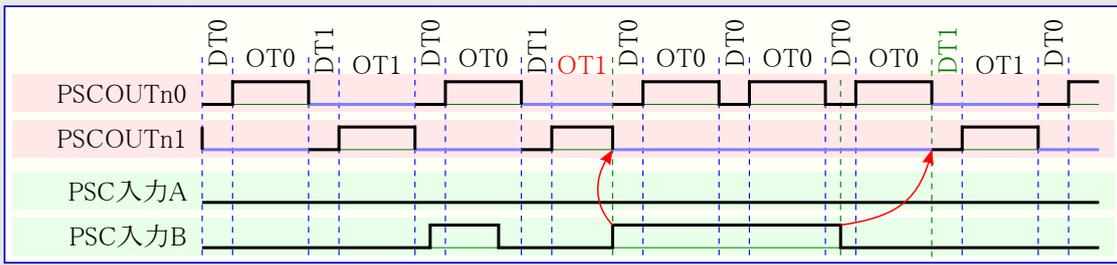
### 12.11. PSC入力動作3：信号停止、障害有効間反対側パルス沈黙時間実行

図12-26. 障害動作3でのPSCn入力A対PSCn動作



PSC入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSC入力A事象が起こると、PSCはPSCOUTn0を開放し、PSC入力Aが活性(有効)状態である間、DT1とOT1へ飛んで実行します。PSC入力AがDT1またはOT1中に開放されても、DT1とOT1の(PSC)半周期は常に全体が実行されます。

図12-27. 障害動作3でのPSCn入力B対PSCn動作



PSC入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSC入力B事象が起こると、PSCはPSCOUTn1を開放し、PSC入力Bが活性(有効)状態である間、DT0とOT0へ飛んで実行します。PSC入力BがDT0またはOT0中に開放されても、DT0とOT0の(PSC)半周期は常に全体が実行されます。

### 12.12. PSC入力動作4：タイミング変更なしで出力非活動

図12-28. 障害動作4でのPSCn入力AまたはPSCn入力B対PSCn動作

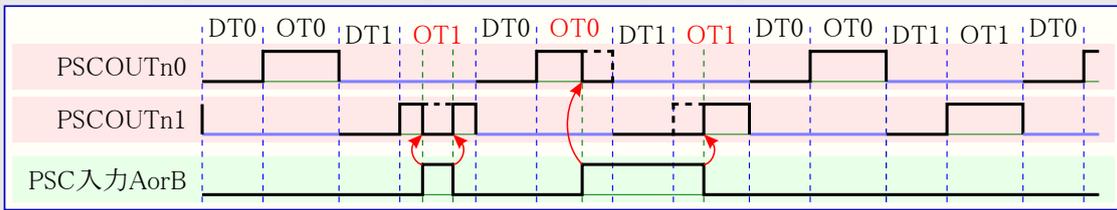
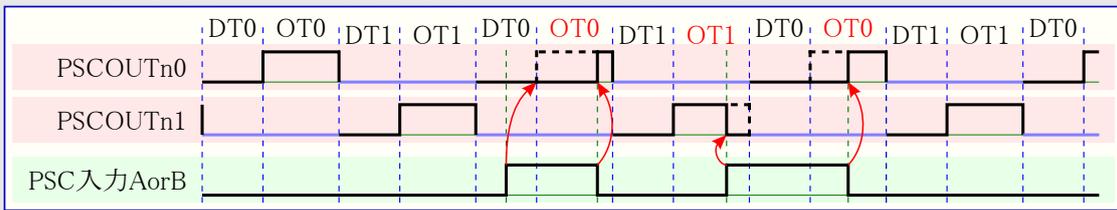
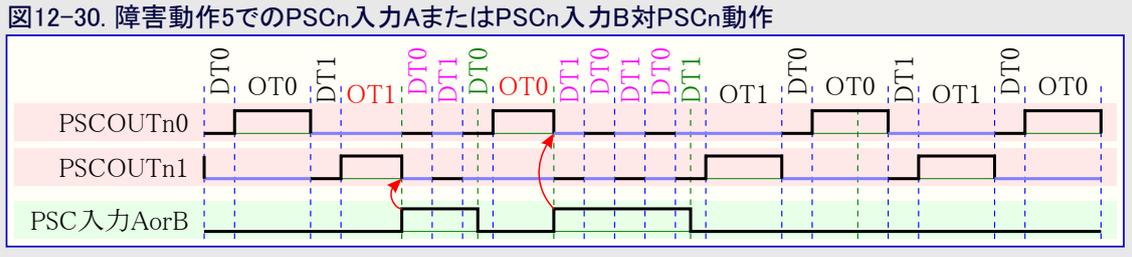


図12-29. 障害動作4でのPSCn入力AまたはPSCn入力B対PSCn動作



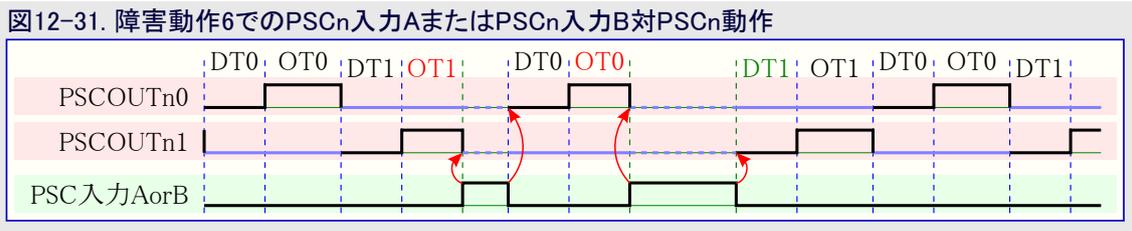
PSC入力A及びPSC入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

12.13. PSC入力動作5：信号停止、沈黙時間挿入



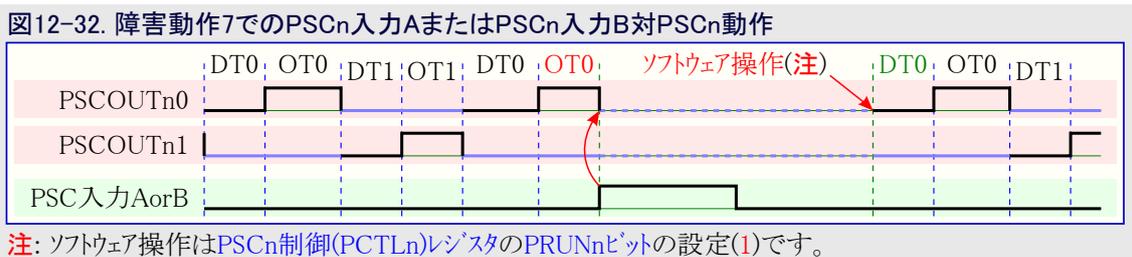
障害動作5使用でのPSC入力A及びPSC入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

12.14. PSC入力動作6：信号停止、反対側沈黙時間へ飛び、待機



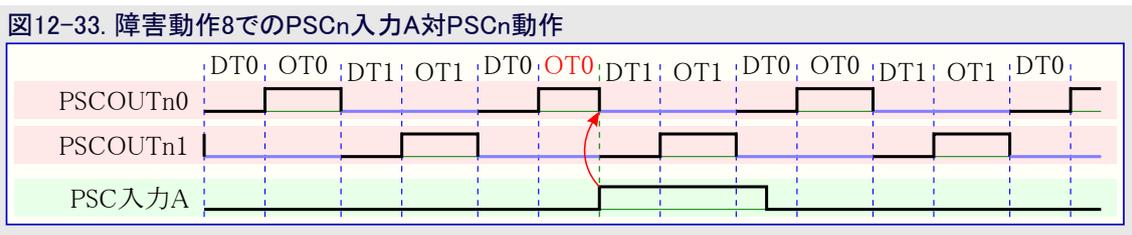
障害動作6使用でのPSC入力A及びPSC入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

12.15. PSC入力動作7：PSC停止、ソフトウェア操作待機

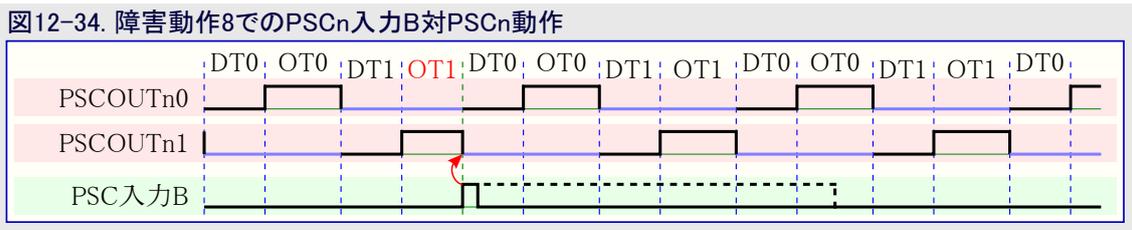


障害動作7使用でのPSC入力A及びPSC入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

12.16. PSC入力動作8：端再起動PSC



出力周波数は再起動入力の意味ある(有効)端の発生によって変調されます。



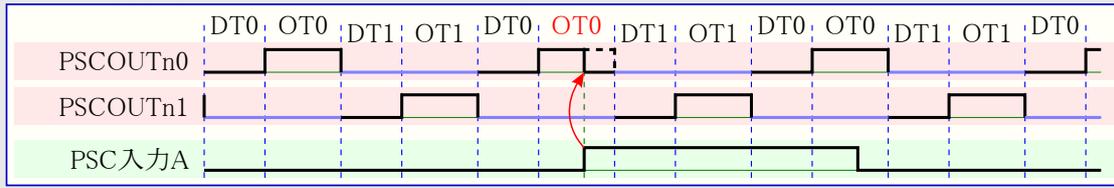
出力周波数は再起動入力の意味ある(有効)端の発生によって変調されます。

再起動事象は対応するON時間間に起きる場合にだけ考慮さ(扱)われます。

注: 1傾斜動作での入力Aの再起動事象は傾斜全体をリセットします。故にPSCは逆側沈黙時間(DT1)に飛び(移行)しません。

## 12.17. PSC入力動作9：周波数固定端再起動PSC

図12-35. 障害動作9でのPSCn入力A対PSCn動作

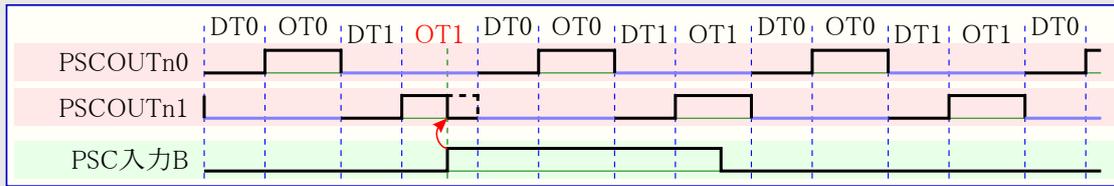


出力周波数は再起動入力の意味ある(有効)端の発生によって変調されません。

再起動入力の意味ある(有効)端が起きる時の出力だけが非活動(無効レベル)にされます。

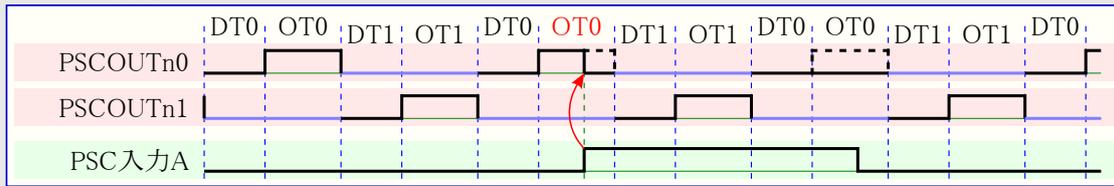
**注:** 本動作でのPSC出力は例え再起動/障害入力が活性(有効)でも、次の傾斜間、活動(通常動作)になります。再起動/障害入力の意味ある(有効)端だけが考慮さ(扱)われます。

図12-36. 障害動作9でのPSCn入力B対PSCn動作



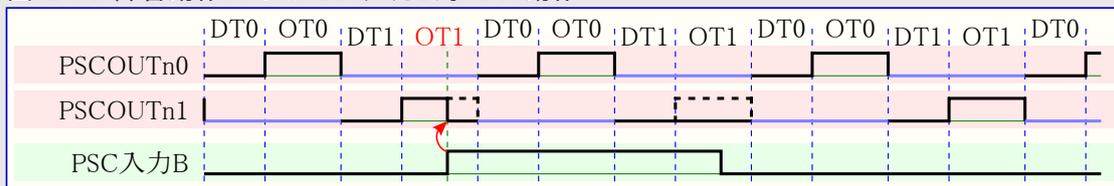
## 12.18. PSC入力動作14：周波数固定端再起動PSC、出力非活動

図12-37. 障害動作14でのPSCn入力A対PSCn動作



出力周波数は再起動入力の意味ある(有効)端の発生によって変調されません。

図12-38. 障害動作14でのPSCn入力B対PSCn動作



出力は再起動入力が活性(有効)である間、非活動(無効レベル)にされます。

PSCの出力は非活性(無効レベル)状態に設定され、対応する傾斜(カウンタ)が中断停止されます。出力は再起動/障害入力が活性(有効)である間、非活性状態に留まります。このPSCは一定周波数で走行します。

## 12.18.1. 走行動作による利用可能な入力動作

いくつかの入力動作はいくつかの走行動作で矛盾があります。故に下表は走行動作によって有効な入力動作を与えます。

表12-8. 走行動作による利用可能な入力動作

入力動作番号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1傾斜動作	可	不可	不可	可	不可	不可	可	可	可	使用 不可				可	使用 不可
2傾斜動作	可	可	可	可	可	可	可	可	可					可	
4傾斜動作	可	可	可	可	可	可	可	可	可					可	
中央整列動作	不可	不可	不可	可	不可	不可	可	不可	不可					不可	

## 12.18.2. 事象捕獲

本PSCはPSC入力にて再起動事象または障害事象が起きる時に時間(PSCカウンタ)値を捕獲できます。この値はソフトウェアによってPSCn捕獲レジスタ(PICRnH, PICRnL)で読めます。

## 12.18.3. 捕獲入力部の使用

PSCn捕獲レジスタ(PICRn)でのPSCカウンタ捕獲の起動には2つの方法があります。(訳注: 共通性から3行追加)

- PSC入力信号によって起動されるハードウェア捕獲入力
- 捕獲ソフトウェア起動(PCSTn)ビットによって起動されるソフトウェア捕獲入力

捕獲入力部使用時の主な要求は到着事象を扱うための十分な能力をプロセッサに割り当てることです。2つの事象間の時間は危険です。プロセッサが次の事象発生前にPICRnレジスタ内の捕獲した値を読めなかったなら、PICRnは新しい値で書き換えられるでしょう。この場合、捕獲結果は不正にされるでしょう。

捕獲入力割り込み使用時、PICRnレジスタ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。捕獲入力割り込みが相対的に高い優先順であっても、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かかる最大クロック周期数に依存します。

## 12.19. PSC2出力

### 12.19.1. 出力配列

PSC2には4傾斜動作で各傾斜に対してPSCOUT20とPSCOUT21の2進値の値設定を許す出力配列があります。

PSCOUT2m(m=0,1,2,3)は全ての対応する傾斜の間、表12-9. で与えられた値を取ります。この出力配列によってPSCOUT20とPSCOUT21の組み合わせの全種類を生成することが可能です。

出力配列使用時、PSCn出力極性(POPn)ビットは出力上の動きを持ちません(無効)。

表12-9. 出力配列 対 傾斜番号

基準PSC出力	傾斜0	傾斜1	傾斜2	傾斜3
PSCOUT20	POMV2A0	POMV2A1	POMV2A2	POMV2A3
PSCOUT21	POMV2B0	POMV2B1	POMV2B2	POMV2B3

### 12.19.2. PSCOUT22とPSCOUT23の選択器

PSC2にはPSCOUT22とPSCOUT23の2つの補助的な出力があります。

PSOC2レジスタのPOS22とPOS23ビットに従って、PSCOUT22とPSCOUT23はPSCOUT20とPSCOUT21を二重化します。

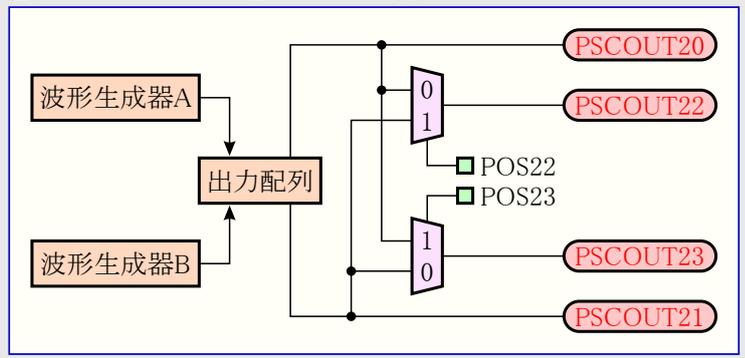
PSOC2レジスタのPOS22ビットが解除(0)なら、PSCOUT22はPSCOUT20を二重化します。

PSOC2レジスタのPOS22ビットが設定(1)なら、PSCOUT22はPSCOUT21を二重化します。

PSOC2レジスタのPOS23ビットが解除(0)なら、PSCOUT23はPSCOUT21を二重化します。

PSOC2レジスタのPOS23ビットが設定(1)なら、PSCOUT23はPSCOUT20を二重化します。

図12-39. PSCOUT22とPSCOUT23出力



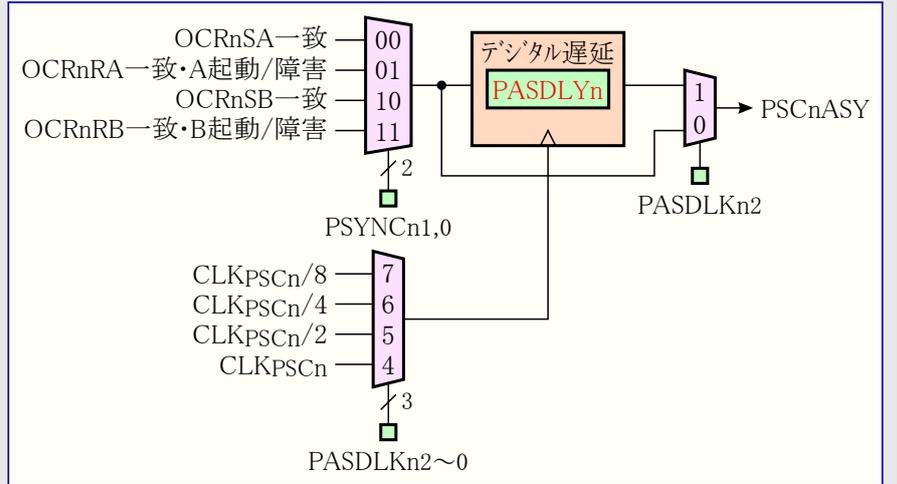
## 12.20. アナログ同期化

PSCは採取&保持またはA/D変換開始と同期するための信号を生成します。同期化は測定にとって必須です。

この信号は80頁の表12-11.と表12-12.で定義されるようにPSCOUTn0またはPSCOUTn1出力の全ての上昇端または下降端から選択できます。

この信号はアナログ同期遅延(PASDLYn)レジスタによって定義されるデジタル遅延によって移動することができます。移動クロックは「ビット7~5 - PASDLKn2~0 : アナログ同期出力遅延/入力無効化選択」で説明されるように、CLKpScまたはCLKpSc/4のどちらかで行えます。

図12-40. アナログ同期化



## 12.21. 割り込みの扱い

各PSCは1つの機能について専用化でき、各PSCにはそれ自身の割り込みシステムがあります。

割り込み元一覧:

- カウンタ再設定(=0、ON時間1(OT1)の最後)
- 増強した周期の最後
- PSC入力事象(設定された事象の有効な端またはレベルの開始)
- PSC相互同期化異常

## 12.22. PSCの同期化

**注:** AT90PWM81/161ではこの機能が無関係で、PRUN2とPARUN2は0に固定されます。

2つまたは3つのPSCが共に同期化できます。この場合2つの波形整列が可能です。

- 波形は主PSCと従PSCが全てで同じPSC周期(本来の使用法)なら、波形は中央整列動作で中央に整列されます。
- 波形は1、2、または4傾斜動作での端(エッジ)で整列されます。

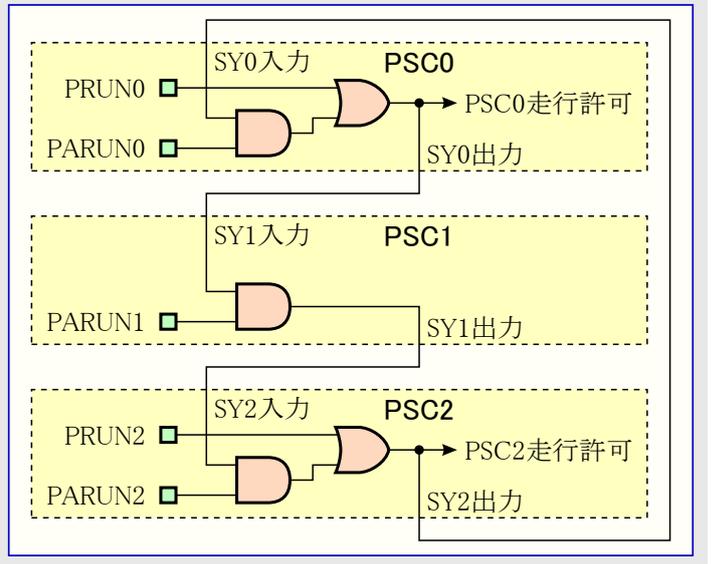
PSCnが設定(1)のPARUNnビットを持つなら、PSCn-1と同時に開始できます。

PRUNnとPARUNnビットはPCTLnレジスタに配置されます(84頁の「PCTL2 - PSC2制御レジスタ」をご覧ください)。

**注:** 同時に3つのPSCのPARUNnビットを設定(1)しないでください。

本機能によって例えば、1つの主PSC動作(PARUNm=0, PRUNm=0)と2つの従PSC動作(PARUNn=1, PRUNn=0)を構成できます。この主PSCは全PSCを同じ瞬間に開始できます(PRUNm=1)。

図12-41. PSC走行同期化



### 12.22.1. 自動走行での障害事象

主/従機構全体のため、障害事象(入力動作7)はPSCnからPSCn-1へとPSCn-1からPSCnへ伝達されます。

走行信号を後続PSCへ伝達するPSCは走行信号が非活性の時にそのPSCを停止します。

3つのPSC間の「PSC走行信号デュージーチェーン」を構築するPSC同期化概念に従い、デュージーチェーンに沿って送信されるPRUNnビットを通してPSCを「停止」できるのは障害動作(PSC入力動作7)だけです。

直前のPSCから走行信号を受けるPSCは、(許可されていれば、)直前のPSCへ障害信号を送ります。故に従PSCはそれらが障害事象を設定且つ許可の時に障害事象を伝達します。

## 12.23. PSCクロック元

PSCは**増強分解能**での高い周波数を生成できなければなりません。

各PSCは2つのクロック入力を持ちます。

- PLLからのCLKPLL
- CLK<sub>I/O</sub>

PSC<sub>n</sub>構成(PCNF<sub>n</sub>)レジスタのPCLKSEL<sub>n</sub>ビットはクロック元選択に使われます。

PSC<sub>n</sub>制御(PCTL<sub>n</sub>)レジスタのPPRE<sub>n1,0</sub>ビットはクロックの分周係数選択に使われます。

図12-42. クロック選択

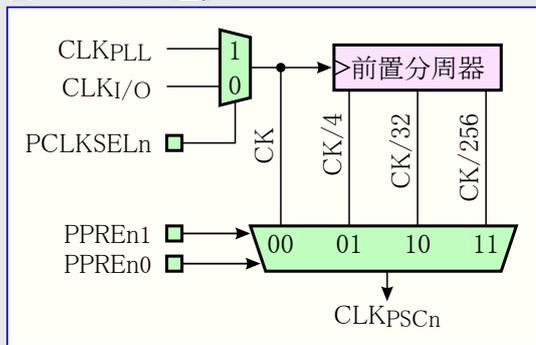


表12-10. クロック元と前置分周の選択

PCLKSEL <sub>n</sub>	PPRE <sub>n1</sub>	PPRE <sub>n0</sub>	CLKPSC <sub>n</sub> 出力
0	0	0	CLK <sub>I/O</sub>
0	0	1	CLK <sub>I/O</sub> /4
0	1	0	CLK <sub>I/O</sub> /32
0	1	1	CLK <sub>I/O</sub> /256
1	0	0	CLKPLL
1	0	1	CLKPLL/4
1	1	0	CLKPLL/32
1	1	1	CLKPLL/256

## 12.24. 割り込み

本項はAT90PWM81/161で行なわれる割り込み動作の仕様を記述します。

### 12.24.1. 割り込みベクタの一覧

各PSCは3つの割り込みベクタを提供します。

- PSC<sub>n</sub> EC (End of Cycle : 周期終了) : 許可で且つOCR<sub>n</sub>RBとの一致で発生
- PSC<sub>n</sub> EEC (End of Enhanced Cycle : 増強周期終了) : 許可で且つ第15増強周期でのOCR<sub>n</sub>RBとの一致で発生
- PSC<sub>n</sub> CAPT (Capture Event : 捕獲事象) : 許可で且つ、同期異常またはPSCカウンタの再起動/捕獲で発生

87頁の「PIM2 - PSC2割り込み許可レジスタ」をご覧ください。

## 12.25. 電力段制御器(PSC)用レジスタ

レジスタはPSC0について説明されます。これらはPSC1について同じです。PSC2に対してのみ異なるレジスタは記述されます。

(訳注) 特記事項を除いて本節内のnはPSC番号を表し、2です。暗背景のビットはAT90PWM81/161で利用不可または無効です。

### 12.25.1. PSOC2 – PSC2同期・出力構成レジスタ (PSC 2 Synchro and Output Configuration)

ビット (\$6E)	7	6	5	4	3	2	1	0	PSOC2
	POS23	POS22	PSYNC21	PSYNC20	POEN2D	POEN2B	POEN2C	POEN2A	
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 – POS23 : PSCOUT23選択 (PSCOUT23 Selection) (PSC2のみ)**

このビットが解除(0)されると、PSCOUT23出力波形は波形生成器Bによって生成されます。

このビットが設定(1)されると、PSCOUT23出力波形は波形生成器Aによって生成されます。

- **ビット6 – POS22 : PSCOUT22選択 (PSCOUT22 Selection) (PSC2のみ)**

このビットが解除(0)されると、PSCOUT22出力波形は波形生成器Aによって生成されます。

このビットが設定(1)されると、PSCOUT22出力波形は波形生成器Bによって生成されます。

- **ビット5,4 – PSYNCn,0 : A/D変換器用同期出力選択 (Synchronization Out for ADC Selection)**

同期化用にA/D変換器へ送る信号を生成する信号元と極性を選びます。

表12-11. 1.2.4傾斜動作での同期元

PSYNCn1	PSYNCn0	説明
0	0	PSCOUTn0の先行端(OCRnSAとの比較一致)で信号送出。
0	1	PSCOUTn0の後行端(OCRnRAとの比較一致またはA部での障害/再起動)で信号送出。
1	0	PSCOUTn1の先行端(OCRnSBとの比較一致)で信号送出。
1	1	PSCOUTn1の後行端(OCRnRBとの比較一致またはB部での障害/再起動)で信号送出。

表12-12. 中央整列動作での同期元

PSYNCn1	PSYNCn0	説明
0	0	(PSC下降計数中の)OCRnRAとの比較一致で信号送出。OCRnRAの最小値は1でなければなりません。
0	1	(PSC上昇計数中の)OCRnRAとの比較一致で信号送出。OCRnRAの最小値は1でなければなりません。
1	0	同期化信号なし。
1	1	

- **ビット3 – POEN2D : PSCOUT23出力許可 (PSCOUT23 Output Enable) (PSC2のみ)**

このビットが解除(0)されると、PSCOUT23に影響を及ぼす第2 I/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUT23に影響を及ぼす第2 I/OピンはPSC波形生成器Bへ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

- **ビット2 – POENnB : PSCOUTn1/B部出力許可 (PSCn OUT Part B Output Enable)**

このビットが解除(0)されると、PSCOUTn1に影響を及ぼすI/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUTn1に影響を及ぼすI/OピンはPSC波形生成器Bへ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

- **ビット1 – POEN2C : PSCOUT22出力許可 (PSCOUT22 Output Enable) (PSC2のみ)**

このビットが解除(0)されると、PSCOUT22に影響を及ぼす第2 I/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUT22に影響を及ぼす第2 I/OピンはPSC波形生成器Aへ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

- **ビット0 – POENnA : PSCOUTn0/A部出力許可 (PSCn OUT Part A Output Enable)**

このビットが解除(0)されると、PSCOUTn0に影響を及ぼすI/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUTn0に影響を及ぼすI/OピンはPSC波形生成器Aへ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

## 12.25.2. OCRnSAH,OCRnSAL (OCRnSA) – PSCn比較SAレジスタ (Output Compare SA Register) (注: n=2のみ)

ビット (\$65)	15	14	13	12	11	10	9	8	
	-				OCRnSA11~8				OCRnSAH
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$64)	7	6	5	4	3	2	1	0	
	OCRnSA7~0								OCRnSAL
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

## 12.25.3. OCRnRAH,OCRnRAL (OCRnRA) – PSCn比較RAレジスタ (Output Compare RA Register) (注: n=2のみ)

ビット (\$2F (\$4F))	15	14	13	12	11	10	9	8	
	-				OCRnRA11~8				OCRnRAH
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$2E (\$4E))	7	6	5	4	3	2	1	0	
	OCRnRA7~0								OCRnRAL
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

## 12.25.4. OCRnSBH,OCRnSBL (OCRnSB) – PSCn比較SBレジスタ (Output Compare SB Register) (注: n=2のみ)

ビット (\$27 (\$47))	15	14	13	12	11	10	9	8	
	-				OCRnSB11~8				OCRnSBH
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$26 (\$46))	7	6	5	4	3	2	1	0	
	OCRnSB7~0								OCRnSBL
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

## 12.25.5. OCRnRBH,OCRnRBL (OCRnRB) – PSCn比較RBレジスタ (Output Compare RB Register) (注: n=2のみ)

ビット (\$29 (\$49))	15	14	13	12	11	10	9	8	
	OCRnRB15~12				OCRnRB11~8				OCRnRBH
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$28 (\$48))	7	6	5	4	3	2	1	0	
	OCRnRB7~0								OCRnRBL
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

PSCn比較RA, RB, SA, SBレジスタは継続的にPSCnカウンタ値と比較される12ビット値を含みます。一致は比較一致割り込みや、関係ピンでの波形出力生成に使えます。

比較RBレジスタは分数(端数)変調(小数分周器)に使われる4ビット値も含みます。

比較レジスタは16ビットと12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、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16ビットまたは12ビットの全レジスタで共用されます。

### 12.25.6. PCNF2 – PSC2構成レジスタ (PSC 2 Configuration Register)

ビット	7	6	5	4	3	2	1	0	
\$15 (\$35)	PFIFTY2	PALOCK2	PLOCK2	PMODE21	PMODE20	POP2	PCLKSEL2	POME2	PCNF2
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	

PSCn構成レジスタはPSCの走行動作構成に使われます。

- **ビット7 – PFIFTYn : PSCn 50%動作 (PSC n Fifty)**

このビットの1書き込みはOCRnRBとOCRnSBだけが使われる50%動作にPSCを設定します。OCRnRBの更新中に、これら(の値)はOCRnRAとOCRnSAに複製されます。この機能は50%波形実行に有用です。

- **ビット6 – PALOCKn : PSCn 自動格納 (PSC n Autolock)**

このビットが設定(1)されると、比較RA,SA,SBレジスタ、出力配列(POM2)レジスタ、PSCn同期/出力構成(PSOCn)レジスタはPSC周期を妨げることなく書けます。PSC内部レジスタの更新は比較Rレジスタが書かれてしまう時に、そのPSC周期の最後で行なわれます。

設定(1)時、本ビットはLOCKnビット(ビット5)に優先します。

- **ビット5 – PLOCKn : PSCn 格納 (PSC n Lock)**

このビットが設定(1)されると、比較RA,RB,SA,SBレジスタ、出力配列(POM2)レジスタ、PSCn同期/出力構成(PSOCn)レジスタはPSC周期を妨げることなく書けます。PSC内部レジスタの更新はLOCKビットが0に解除される時に行なわれます。

- **ビット4,3 – PMODEn1,0 : PSCn 走行動作種別 (PSC n Mode)**

PSCの動作を選びます。

表12-13. PSCn 走行動作種別選択

PMODEn1	PMODEn0	走行動作種別
0	0	1傾斜動作
0	1	2傾斜動作
1	0	4傾斜動作
1	1	中央整列動作

- **ビット2 – POPn : PSCn 出力極性 (PSC n Output Polarity)**

このビットが解除(0)されている場合、PSC出力はLow有効です。

このビットが設定(1)されている場合、PSC出力はHigh有効です。

- **ビット1 – PCLKSELn : PSCn 入力クロック選択 (PSC n Input Clock Select)**

このビットは高速クロックまたは低速クロックのどちらかを選びます。

高速クロック(CLKPLL)を選ぶにはこのビットを設定(1)してください。

低速クロック(CLKI/O)を選ぶにはこのビットを解除(0)してください。

- **ビット0 – POME2 : PSC2 出力配列許可 (PSC 2 Output Matrix Enable) (PSC2のみ)**

PSC2出力で出力配列機能を許可するにはこのビットを設定(1)してください。

出力配列が使われると、PSC2出力極性(POP2)は出力での動作を持ちません(無効です)。

## 12.25.7. PCNFE2 – PSC2拡張構成レジスタ (PSC 2 Extended Configuration Register)

ビット (\$70)	7	6	5	4	3	2	1	0	
	PASDLK22	PASDLK21	PASDLK20	PBFM21	PELEV2A1	PELEV2A0	PESEL2A1	PESEL2B1	PCNFE2
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	

PSCn拡張構成レジスタはPSCの走行動作構成に使われます。

- ビット7~5 – PASDLKn2~0 : アナログ同期出力遅延/入力無効化選択 (Analog Synchronization Output Delay or Input Blanking select)  
アナログ信号同期遅延または入力無効化の形態を定義します。

表12-14. アナログ信号同期または入力無効化の形態選択

PASDLKn2	PASDLKn1	PASDLKn0	説明
0	0	0	アナログ信号同期遅延なし、入力無効化なし
0	0	1	アナログ信号同期遅延なし、PSCクロックを用いてPSC周期の最後で始まる入力無効化
0	1	0	アナログ信号同期遅延なし、PSCクロックを用いてOCRnSA事象で始まる入力無効化
0	1	1	アナログ信号同期遅延なし、PSCクロックを用いてOCRnSB事象で始まる入力無効化
1	0	0	PSCクロックでのアナログ信号同期遅延、入力無効化なし
1	0	1	PSCクロック/2でのアナログ信号同期遅延、入力無効化なし
1	1	0	PSCクロック/4でのアナログ信号同期遅延、入力無効化なし
1	1	1	PSCクロック/8でのアナログ信号同期遅延、入力無効化なし

- ビット4 – PBFMn1 : PSCn平衡分数(端数)変調(小数分周器)動作種別ビット1 (PSC n Balance Flank Width Modulation bit 1)  
PSCn制御(PCTLn)レジスタのPBFMn0ビットと共に平衡分数変調を定義します。

表12-15. 分数変調動作形態選択

PBFMn1	PBFMn0	説明
0	0	RB(ON時間1のみ)で分数変調動作
0	1	RB+RA(ON時間0とON時間1)で分数変調動作
1	0	SB(沈黙時間1のみ)で分数変調動作 (注)
1	1	SB+SA(沈黙時間0と沈黙時間1)で分数変調動作

注: 1傾斜動作ではSAやSA+SBの変更もON時間に影響を及ぼします。1傾斜動作でのPSCOUT出力の基本波形をご覧ください。

- ビット3 – PELEVnA1 : PSCn A部入力端(エッジ)/レベル選択 (PSC n Edge Level Selector of Input part A)  
PELEVnA0と共にPSCのA部の有効な端またはレベルを定義します。

表12-16. PSCの端とレベルの入力選択

PELEVnA1	PELEVnA0	説明
0	0	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
0	1	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
1	0	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
1	1	(予約)

- ビット2 – PELEVnB1 : PSCn B部入力端(エッジ)/レベル選択 (PSC n Edge Level Selector of Input part B)  
PELEVnB0と共にPSCのB部の有効な端またはレベルを定義します。

表12-17. PSCの端とレベルの入力選択

PELEVnB1	PELEVnB0	説明
0	0	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
0	1	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
1	0	選んだ入力の上昇端またはHighレベルが再起動または障害機能に対する重要な事象を生成します。
1	1	(予約)

- ビット1 – PISELnA1 : PSCn A部入力選択 (PSC n Input Select for part A)

PISELnA0と共にPSCのA部の有効な信号を定義します。

表12-18. PSCの再起動/障害の入力選択

PISELnA1	PISELnA0	説明
0	0	PSCINn
0	1	アナログ比較器2出力
1	0	PSCINnA
1	1	アナログ比較器3出力

- ビット0 – PISELnB1 : PSCn B部入力選択 (PSC n Input Select for part B)

PISELnB0と共にPSCのB部の有効な信号を定義します。

表12-19. PSCの再起動/障害の入力選択

PISELnB1	PISELnB0	説明
0	0	PSCINn
0	1	アナログ比較器2出力
1	0	PSCINnA
1	1	アナログ比較器3出力

### 12.25.8. PASDLYn – PSCnアナログ同期遅延レジスタ (Analog Synchronization Delay Register) (注: n=2のみ)

ビット (\$71)	7	6	5	4	3	2	1	0	PASDLYn
Read/Write	W	W	W	W	W	W	W	W	W
初期値	0	0	0	0	0	0	0	0	0

アナログ同期遅延レジスタは以下で使われる8ビットの遅延を格納します。

- 入力信号無効化。69頁の「PSC入力」項をご覧ください。
- PSCOUTnx端とPSCnASY信号の移動。78頁の「アナログ同期化」項をご覧ください。

83頁の「ビット7~5 – PASDLKn2~0 : アナログ同期出力遅延/入力無効化選択」と80頁の「ビット5,4 – PSYNcn1,0 : A/D変換器用同期出力選択」もご覧ください。

### 12.25.9. PCTL2 – PSC2制御レジスタ (PSC 2 Control Register)

ビット (\$16 (\$36))	7	6	5	4	3	2	1	0	PCTL2
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	0

- ビット7,6 – PPREn1,0 : PSCn 前置分周選択 (PSC n Prescaler Select)

この2ビットはPSCn入力クロック分周係数を選びます。全ての生成波形がこの係数によって変更されます。

表12-20. PSCn 前置分周選択

PPREn1	PPREn0	前置分周
0	0	なし(1/1)
0	1	4分周
1	0	32分周
1	1	256分周

- ビット5 – PBFMn0 : PSCn平衡分数(端数)変調(小数分周器)動作種別ビット0 (PSC n Balance Flank Width Modulation bit 0)

PSCn拡張構成(PCNFEn)レジスタのPBFMn1ビットと共に平衡分数変調を定義します。83頁の表12-15をご覧ください。

- ビット4 – PAOCnB : PSCn 非同期出力制御B (PSC n Asynchronous Output Control B)

このビットが設定(1)されると、B部へ選ばれた障害入力はPSCOUTn1出力を直接的に操作できます。72頁の「PSC入力設定」をご覧ください。

- ビット3 – PAOCnA : PSCn 非同期出力制御A (PSC n Asynchronous Output Control A)

このビットが設定(1)されると、A部へ選ばれた障害入力はPSCOUTn0出力を直接的に操作できます。72頁の「PSC入力設定」をご覧ください。

• **ビット2 – PARUNn : PSCn 自動走行許可 (PSC n Autorun)**

このビットが設定(1)されると、PSCnはPSCn-1と共に始まります。これは次のどちらかの場合にPSCnが始動することを意味します。

- PCTLn-1のPRUNn-1が設定(1)される時
- PCTLn-1のPARUNn-1が設定(1)され、且つPCTLn-2のPRUNn-2が設定(1)される時

• **ビット1 – PCCYCn : PSCn 周期完了許可 (PSC n Complete Cycle)**

このビットが設定(1)されると、PSCnはPRUNnの解除(0)によって要求した停止操作に先立って波形周期全体を完了します。従属動作(PARUNn=1)での本ビットは関連がありません。

• **ビット0 – PRUNn : PSCn 走行許可 (PSC n Run)**

このビットの1書き込みはPSCnを始動します。78頁の「PSCの同期化」をご覧ください。

設定(1)時、本ビットはPARUNnビット(ビット2)に優先します。

**12.25.10. PFRCnA – PSCn入力A制御レジスタ (PSC n Input A Control Register) (注: n=2のみ)**

ビット (\$66)	7	6	5	4	3	2	1	0	
	PCAE <sub>nA</sub>	PISEL <sub>nA0</sub>	PELEV <sub>nA0</sub>	PFLTE <sub>nA</sub>	PRFM <sub>nA3</sub>	PRFM <sub>nA2</sub>	PRFM <sub>nA1</sub>	PRFM <sub>nA0</sub>	PFRC <sub>nA</sub>
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0	0

**12.25.11. PFRCnB – PSCn入力B制御レジスタ (PSC n Input B Control Register) (注: n=2のみ)**

ビット (\$67)	7	6	5	4	3	2	1	0	
	PCAE <sub>nB</sub>	PISEL <sub>nB0</sub>	PELEV <sub>nB0</sub>	PFLTE <sub>nB</sub>	PRFM <sub>nB3</sub>	PRFM <sub>nB2</sub>	PRFM <sub>nB1</sub>	PRFM <sub>nB0</sub>	PFRC <sub>nB</sub>
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0	0

入力制御レジスタはPSCの2つ(A部とB部)の再起動/障害部構成に使われます。この2つの単位部は同一(構造)で、従って同じ方法で設定されます。(訳補:以下のxはAまたはBです。)

• **ビット7 – PCAEnx : PSCn x部入力捕獲許可 (PSC n Capture Enable Input Part x)**

このビットの1書き込みはx部に対する入力として選んだ入力で外部事象が起こる時の捕獲機能を許可します(次のPISELnx0ビットをご覧ください)。

• **ビット6 – PISELnx0 : PSCn x部入力選択 (PSC n Input Select for Part x)**

PSCn拡張構成(PCNFEn)レジスタのPISELnx1ビットと共にPSCn A部の活性(有効)信号を定義します。84頁の表12-18と表12-19をご覧ください。

• **ビット5 – PELEVnx0 : PSCn 非同期出力制御x (PSC n Asynchronous Output Control x)**

PSCn拡張構成(PCNFEn)レジスタのPELEVnx1ビットと共にPSCn A部の活性(有効)な端(エッジ)とレベル信号を定義します。83頁の表12-16と表12-17をご覧ください。

• **ビット4 – PFLTE<sub>nA</sub> : PSCn x部入力濾波器許可 (PSC n Filter Enable on Input Part x)**

このビットの設定(1)は捕獲入力雑音消去器を活性(有効)にします。雑音消去器が活動(状態)にされると、再起動ピンからの入力は濾波されます。この濾波機能はその出力を変更するのに再起動ピンの4連続同一値採取を必要とします。従って捕獲入力は雑音消去器が許可される時に4発振器周期遅らされます。

• **ビット3~0 – PRFMnx3~0 : PSCn 障害/再起動 x部動作選択 (PSC n Fault Mode)**

これらの4ビットは障害または再起動機能の動作種別を定義します。(より多くの説明については72頁の表12-17をご覧ください。)

表12-21. 障害/再起動動作種別とPSC入力動作 (端(エッジ)/レベル選択)

番号	PRFMnx3~0	説明
0	0 0 0 0	動作なし、PSC入力は無効。
1	0 0 0 1	PSC入力動作1：信号停止、反対側パルス沈黙時間へ飛び、待機
2	0 0 1 0	PSC入力動作2：信号停止、反対側パルス沈黙時間実行、待機
3	0 0 1 1	PSC入力動作3：信号停止、障害有効間反対側沈黙時間実行
4	0 1 0 0	PSC入力動作4：タイミング変更なしで出力非活動
5	0 1 0 1	PSC入力動作5：信号停止、沈黙時間挿入
6	0 1 1 0	PSC入力動作6：信号停止、反対側沈黙時間へ飛び、待機
7	0 1 1 1	PSC入力動作7：PSC停止、ソフトウェア操作待機
8	1 0 0 0	PSC入力動作8：端再起動PSC
9	1 0 0 1	PSC入力動作9：周波数固定端再起動PSC
10	1 0 1 0	(予約)：使用不可
11	1 0 1 1	
12	1 1 0 0	
13	1 1 0 1	
14	1 1 1 0	PSC入力動作14：周波数固定端再起動PSC、出力非活動
15	1 1 1 1	(予約)：使用不可

12.25.12. PICRnH, PICRnL (PICRn) – PSCn捕獲レジスタ (PSC n Input Capture Register) (注: n=2のみ)

ビット (\$6D)	15	14	13	12	11	10	9	8	
	PCSTn	-	-	-	PICRn11~8				PICRnH
Read/Write	R/W	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$6C)	7	6	5	4	3	2	1	0	
	PICRn7~0								PICRnL
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

• ビット7 – PCSTn 捕獲ソフトウェア起動 (PSC n Capture Software Trig bit)

PSCカウンタの捕獲を起こすにはこのビットを設定(1)してください。読み込み時、本ビットが設定(1)されていれば捕獲操作はPCSTnの設定(1)によって起動されたことを意味し、さもなければ捕獲操作はPSC入力によって起動されたことを意味します。

捕獲入力は捕獲機能が許可される(PFRCnxレジスタのPCAEnxビットが設定(1)される)場合の、許可した入力ピン(または任意選択のアナログ比較器n出力)での事象発生毎にPSCnカウンタ値で更新されます。

捕獲レジスタは12ビットの容量です。CPUがこれらのレジスタをアクセスする時に上位と下位のバイトが同時に読まれるのを保証するために、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16ビットまたは12ビットの全レジスタで共有されます。

12.26. 電力段制御器(PSC)2用特定レジスタ

12.26.1. POM2 – PSC2出力配列レジスタ (PSC 2 Output Matrix)

ビット (\$6F)	7	6	5	4	3	2	1	0	
	POMV2B3	POMV2B2	POMV2B1	POMV2B0	POMV2A3	POMV2A2	POMV2A1	POMV2A0	POM2
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	

• ビット7 – POMV2B3 : PSC2 出力配列 出力B傾斜3 (Output Matrix Output B Ramp 3)

このビットは傾斜3(第4傾斜)間のPSCOUT21(と/またはPSCOUT23)の状態を与えます。

• ビット6 – POMV2B2 : PSC2 出力配列 出力B傾斜2 (Output Matrix Output B Ramp 2)

このビットは傾斜2(第3傾斜)間のPSCOUT21(と/またはPSCOUT23)の状態を与えます。

• ビット5 – POMV2B1 : PSC2 出力配列 出力B傾斜1 (Output Matrix Output B Ramp 1)

このビットは傾斜1(第2傾斜)間のPSCOUT21(と/またはPSCOUT23)の状態を与えます。

- **ビット4 – POMV2B0 : PSC2 出力配列 出力B傾斜0 (Output Matrix Output B Ramp 0)**  
このビットは傾斜0(第1傾斜)間のPSCOUT21(と/またはPSCOUT23)の状態を与えます。
- **ビット3 – POMV2A3 : PSC2 出力配列 出力A傾斜3 (Output Matrix Output A Ramp 3)**  
このビットは傾斜3(第4傾斜)間のPSCOUT20(と/またはPSCOUT22)の状態を与えます。
- **ビット2 – POMV2A2 : PSC2 出力配列 出力A傾斜2 (Output Matrix Output A Ramp 2)**  
このビットは傾斜2(第3傾斜)間のPSCOUT20(と/またはPSCOUT22)の状態を与えます。
- **ビット1 – POMV2A1 : PSC2 出力配列 出力A傾斜1 (Output Matrix Output A Ramp 1)**  
このビットは傾斜1(第2傾斜)間のPSCOUT20(と/またはPSCOUT22)の状態を与えます。
- **ビット0 – POMV2A0 : PSC2 出力配列 出力A傾斜0 (Output Matrix Output A Ramp 0)**  
このビットは傾斜0(第1傾斜)間のPSCOUT20(と/またはPSCOUT22)の状態を与えます。

(訳補) 出力配列については77頁の「PSC2出力」をご覧ください。

## 12.27. 電力段制御器(PSC)割り込み用レジスタ

### 12.27.1. PIMn – PSCn割り込み許可レジスタ (PSC n Interrupt Mask Register) (注: n=2のみ)

ビット	7	6	5	4	3	2	1	0	
\$13 (\$33)	–	–	PSEIEn	PEVEnB	PEVEnA	–	PEOPEEn	PEOPEEn	PIMn
Read/Write	R	R	R/W	R/W	R/W	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- **ビット5 – PSEIEn : PSCn 同期異常割り込み許可 (PSC n Synchro Error Interrupt Enable)**  
このビットが設定(1)されると、同期異常割り込み要求(PSEIn)ビットが(設定(1)ならば)割り込みを生成します。
- **ビット4 – PEVEnB : PSCn B部外部事象割り込み許可 (PSC n External Event B Interrupt Enable)**  
このビットが設定(1)されると、B部の障害/再起動部からの捕獲を発生できる外部事象は割り込みも生成します。
- **ビット3 – PEVEnA : PSCn A部外部事象割り込み許可 (PSC n External Event A Interrupt Enable)**  
このビットが設定(1)されると、A部の障害/再起動部からの捕獲を発生できる外部事象は割り込みも生成します。
- **ビット1 – PEOPEEn : PSCn 増強周期終了割り込み許可 (PSC n End Of Enhanced Cycle Interrupt Enable)**  
このビットが設定(1)されると、PSCが第15PSC周期の最後に達する時に割り込みが生成されます。これは割り込みルーチンでPSC値を更新して次のPSC周期の最後に新しい値の新しい増強周期を開始することを許します。
- **ビット0 – PEOPEEn : PSCn 周期終了割り込み許可 (PSC n End Of Cycle Interrupt Enable)**  
このビットが設定(1)されると、PSCが周期全体(PSC周期)の最後に達する時に割り込みが生成されます。

### 12.27.2. PIFRn – PSCn割り込み要求フラグ レジスタ (PSC n Interrupt Flag Register) (注: n=2のみ)

ビット	7	6	5	4	3	2	1	0	
\$14 (\$34)	POACnB	POACnA	PSEIn	PEVnB	PEVnA	PRNn1	PRNn0	PEOPn	PIFRn
Read/Write	R	R	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 – POACnB : PSCn B部出力活動 (PSC n Output B Activity)**  
このビットはPSCOUTn1出力が0⇒1または1⇒0に変化する毎にハードウェアによって設定(1)されます。これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。この機能は固定化した外部入力信号のためにPSC出力が変化しないことを検知するのに有用です。
- **ビット6 – POACnA : PSCn A部出力活動 (PSC n Output A Activity)**  
このビットはPSCOUTn0出力が0⇒1または1⇒0に変化する毎にハードウェアによって設定(1)されます。これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。この機能は固定化した外部入力信号のためにPSC出力が変化しないことを検知するのに有用です。
- **ビット5 – PSEIn : PSCn 同期異常割り込み要求フラグ (PSC n Synchro Error Interrupt)**  
このビットは自動走行(PARUNn=1)に設定したPSCnの更新(またはPSC周期の最後)が走行入力信号を発生したPSCn-1と同時に起きない時にハードウェアによって設定(1)されます。(PSC0に対するPSCn-1はPSC2)  
これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。  
この機能はPSCが主PSCと同じ速度、同じ位相で走行しないことを検知するのに有用です。

•ビット4 – PEVnB : PSCn B部外部事象割り込み要求フラグ (PSC n External Event B Interrupt)

このビットはB部の障害/再起動部からの捕獲または再起動を発生できる外部事象が起こる時に設定(1)されます。これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEVEnBビット=0)、読めます。

•ビット3 – PEVnA : PSCn A部外部事象割り込み要求フラグ (PSC n External Event A Interrupt)

このビットはA部の障害/再起動部からの捕獲または再起動を発生できる外部事象が起こる時に設定(1)されます。これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEVEnAビット=0)、読めます。

•ビット2,1 – PRNn1,0 : PSCn 傾斜番号 (PSC n Ramp Number)

最後のA部外部事象割り込み要求(PEVnA)またはB部外部事象割り込み要求(PEVnB)発生時の傾斜番号の記録です。

表12-22. PEVnx発生時傾斜番号

PRNn1,0	00	01	10	11
傾斜番号	0	1	2	3

•ビット0 – PEOPn : PSCn 周期終了割り込み要求フラグ (PSC n End Of Cycle Interrupt)

このビットは周期全体(PSC周期)を達成した時に設定(1)されます。これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

12.27.3. リセット中の電力段制御器(PSC)出力の動き

外部部品の安全のため、リセット中のPSC出力状態はPSCRV,PSCRRB,PSC2RBA,PSC2RBヒューズによってプログラム(設定)できます。これらのヒューズは以下のように拡張ヒューズ バイトに配置されます。

表12-23. 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
PSC2RB	7	電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSC2RBA	6	OUT22と23用の電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRRB	5	縮小電力段制御器(PSCR)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRV	4	PSCOUTとPSCOUTRのリセット値	1 (非プログラム) High
PSCINRB	3	PSCとPSCRの入力 リセット時動作	1 (非プログラム) 標準ポート動作
BODLEVEL2	2	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注1)	1 (非プログラム)
BODLEVEL1	1		0 (プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: BODLEVELヒューズの復号については33頁の表7-2をご覧ください。

PSCRVはPSC0RBとPSC2RBヒューズによって選んだPSC出力でHighかLowのどちらの状態を強制するのかわ与えます。PSCRVヒューズが1に等しい(非プログラム)なら、選んだPSC出力はHigh状態に強制されます。

PSCRRBヒューズが1に等しい(非プログラム)なら、PSCOUTR0とPSCOUTR1は標準ポート動作を保ちます。PSCRRBヒューズが0に等しい(プログラム)なら、PSCOUTR0とPSCOUTR1はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUTR0とPSCOUTR1はPSC0同期/出力構成(PSOC0)レジスタが書かれるまで、この強制された状態を保ちます。

PSC2RBヒューズが1に等しい(非プログラム)なら、PSCOUT20とPSCOUT21は標準ポート動作を保ちます。PSC2RBヒューズが0に等しい(プログラム)なら、PSCOUT20とPSCOUT21はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUT20とPSCOUT21はPSC2同期/出力構成(PSOC2)レジスタが書かれるまで、この強制された状態を保ちます。

PSC2RBAヒューズが1に等しい(非プログラム)なら、PSCOUT22とPSCOUT23は標準ポート動作を保ちます。PSC2RBAヒューズが0に等しい(プログラム)なら、PSCOUT22とPSCOUT23はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUT22とPSCOUT23はPSC2同期/出力構成(PSOC2)レジスタが書かれるまで、この強制された状態を保ちます。

12.27.4. リセット中の電力段制御器(PSC)入力の動き

リセット下の電力消費の理由のために、リセット中のPSCとPSCRの入力の状態はPSCINRBヒューズによってプログラム(設定)できます。

PSCINRBヒューズが1に等しい(非プログラム)なら、PSCとPSCRの入力は標準ポート動作を保ちます。PSCINRBヒューズが0に等しい(プログラム)なら、リセットが活性の間、PSCとPSCRの入力プルアップが強制されます。影響を及ぼされるピンはPSCIN2,PSCINr,PSCIN2A,PSCINrAです。PDIでのどの衝突も防ぐために、このヒューズはPSCINrBで無効です。

## 13. 縮小電力段制御器 (PSCR)

この縮小電力段制御器は高性能波形制御器です。

### 13.1. 特徴

- PWM波形生成機能 (設定可能な2つの相互補完出力)
- 沈黙時間制御
- 12ビット分解能までの標準動作
- 最大16ビットの増強分解能
- 64MHzまでの周波数
- 外部事象での条件付き波形 (ゼロ交差、電流感応...)
- A/D変換器同期動作
- 過負荷保護機能
- 異常保護機能、全出力Hi-Zまたは非活性状態(ヒューズ設定可能)に強制する緊急入力
- ハードウェアによる高速緊急停止

### 13.2. 概要

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'または'r'は電力段制御器(PSC)番号の置き換えで、この場合は0です。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えば電力段制御器0(PSC0)同期/出力設定レジスタに対するアクセスのPSOC0のように)。
- 小文字の'x'は縮小電力段制御器(PSCR)部分の置き換えで、この場合はAまたはBです。けれどもプログラムでレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えば障害/再起動を設定する電力段正制御器0Aの入力制御レジスタに対するアクセスのPFRC0Aのように)。

電力段制御器(PSC)の目的は基板上の電力単位部を制御することです。これは2つの出力を持ちます。

これらの出力は色々な方法で使えます。

- 単一電力トランジスタを駆動する'1出力' (例えば、DC/DC変換器、力率改善(PFC)応用)
- 半ブリッジを駆動する'2出力' (例えば、調光応用)

PSCRには生成波形の直接的な動きの提供を予定する目的の2つの入力があります。

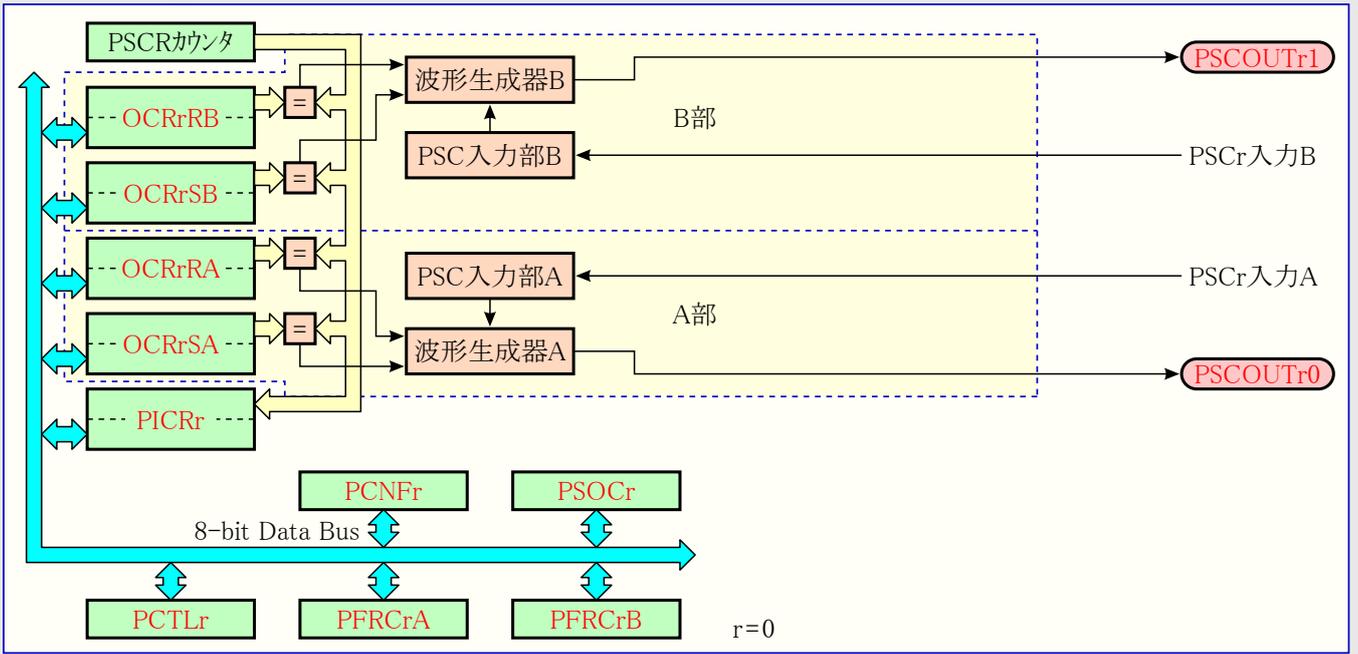
- 電流感応安定化
- 0交差再起動
- 消磁再起動
- 障害入力

31頁の「PRR – 電力削減レジスタ」でのPRPSCRビットはPSCR部を許可するために0を書かれなければなりません。

(訳補) 以降で説明されるPSC動作は基本部分に関して一般的なAVRのタイマ/カウンタを使ったPWMと同様に考えられます。大きく異なるのは出力変移点を指定するための比較レジスタが4つあることです。これらのレジスタの基本的な使用方法は、2つを1組にして相互補完出力の一方を制御し、他の1組でもう一方の出力を制御します。各組内のレジスタの1つは出力のON遷移点を、他方はOFF遷移点を指定します。以降で記述される比較SA,RA,SB,RBの各レジスタが、これらに該当します。この1組の比較レジスタでの波形生成部が、各々A部、B部として記述されています。従って比較SAレジスタはA部セット(Set)点用比較レジスタ、比較RAレジスタはA部リセット(Reset)点用比較レジスタを意味します。

### 13.3. 縮小電力段制御器(PSCR)説明

図13-1. PSCR構成図

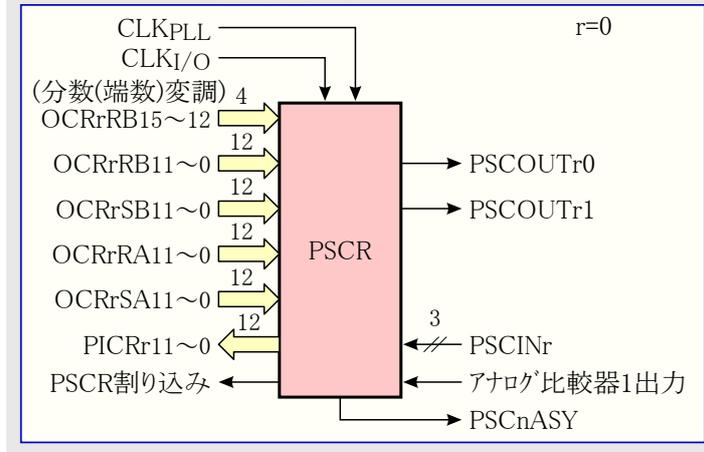


#### 13.3.1. 出力極性

PSCRの極性('High有効'または'Low有効')は設定可能です。以降の例のタイミング図は'High有効'極性で与られます。

### 13.4. 信号説明

図13-2. PSCR外部接続構成図



#### 13.4.1. 入力説明

表13-1. 内部入力

名称	説明	形式/ビット数
OCRrRB15~12	周波数分解能増強値 (小数分周器)	4ビットレジスタ
OCRrRB11~0	B部でのリセット信号比較値 (PSCOUTr1)	12ビットレジスタ
OCRrSB11~0	B部でのセット信号比較値 (PSCOUTr1)	12ビットレジスタ
OCRrRA11~0	A部でのリセット信号比較値 (PSCOUTr0)	12ビットレジスタ
OCRrSA11~0	A部でのセット信号比較値 (PSCOUTr0)	12ビットレジスタ
CLKI/O	I/Oクロックからのクロック入力	信号
CLKPLL	PLLからのクロック入力	信号

表13-2. PSCR部外入力

名称	説明	形式/ビット数
PSCINr	障害/再起動機能に使われる入力0	信号
アナログ比較器1出力	障害/再起動機能に使われる入力1	信号
PSCINrA	障害/再起動機能に使われる入力2	信号
PSCINrB	障害/再起動機能に使われる入力3	信号

13.4.2. 出力説明

表13-3. PSCR部出力

名称	説明	形式/ビット数
PSCOUTr0	PSCRのA部からのPSCr出力0	信号
PSCOUTr1	PSCRのB部からのPSCr出力1	信号

表13-4. 内部出力

名称	説明	形式/ビット数
PICRr11~0	PSCR捕獲入力レジスタ (再起動事象でのカウンタ値)	12ビットレジスタ
PSCR割り込み	PSCR割り込み要求 (溢れ、障害、捕獲発生)の3種	信号
PSCRASY	A/D変換器(+増幅器)同期 (100頁の「アナログ同期化」参照)	信号

13.5. 機能説明

13.5.1. 波形周期

縮小電力段制御器(PSCR)によって生成した波形は2つの波形の順次連続として説明できます。

1つ目の波形はPSCRのA部とPSCOUTr0出力に関係します。この波形部分は図13-3.での補助周期Aです。

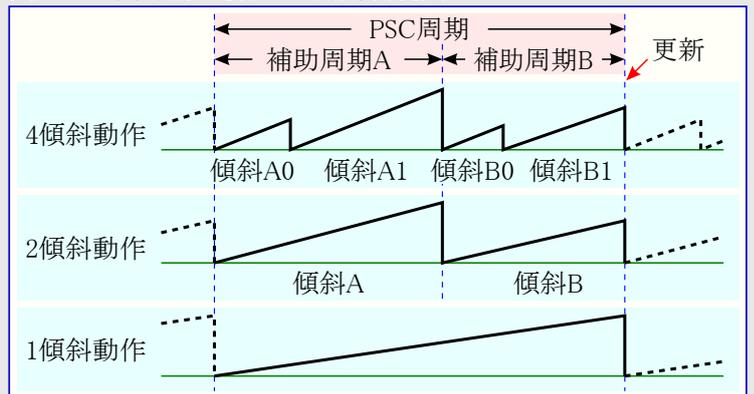
2つ目の波形はPSCRのB部とPSCOUTr1出力に関係します。この波形部分は図13-3.での補助周期Bです。

(周期の)全体波形は補助周期Bの最後で終了されます。これは波形Bの最後での意味です。

傾斜は波形生成器内に含まれるPSCRカウンタの出力を表します。

新規値群の更新が傾斜動作に拘らず、(PSCR周期内の)最終傾斜の頂上(TOP)で行なわれることに注意してください。

図13-3. 1,2,4傾斜動作での周期表現図



13.5.2. 走行動作説明

出力信号の波形と長さは走行動作と時間値(DT0,OT0,DT1,OT1)によって決められます。次の4つの動作が可能です。

- 4傾斜動作
- 2傾斜動作
- 1傾斜動作

PSCOUTr0の活性時間は信号の波形と長さはON時間0(OT0)値によって与えられます。PSCOUTr1の活性時間は信号の波形と長さはON時間1(OT1)値によって与えられます。それら両方は12ビット値です。沈黙時間0(DT0)と沈黙時間1(DT1)によってPSCOUTr0とPSCOUTr1の活性信号間の沈黙時間を調整します。

この波形の周波数は次式によって定義されます。

$$f_{PSCn} = \frac{1}{PSCR周期} = \frac{f_{CLK\_PSCn}}{(OT0+OT1+DT0+DT1)}$$

### 13.5.2.1. 4傾斜動作

4傾斜動作での周期内の各時間は各々自身の定義を持ちます。

PSCの入力クロックはCLKPSCによって与えられます。

PSCOUTr0とPSCOUTr1信号はON時間0(OT0)、沈黙時間0(DT0)、ON時間1(OT1)、沈黙時間1(DT1)の値によって定義されます。

$$\text{ON時間0} = \text{OCrRrRA} \times 1/f_{\text{CLKPSC}}$$

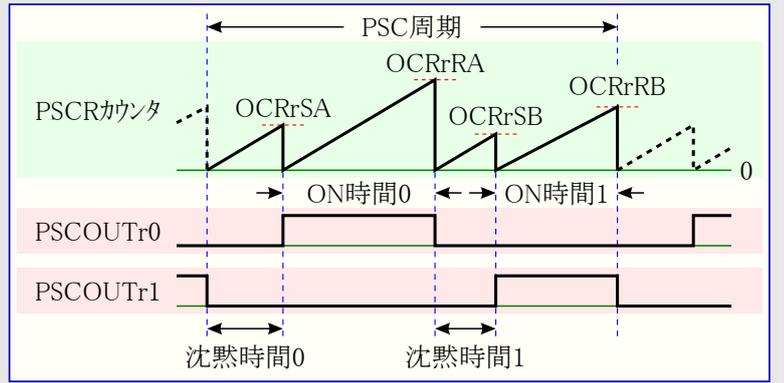
$$\text{ON時間1} = \text{OCrRrRB} \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間0} = (\text{OCrRrSA} + 2) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間1} = (\text{OCrRrSB} + 2) \times 1/f_{\text{CLKPSC}}$$

**注:** 沈黙時間0と沈黙時間1の最小値は $2 \times 1/f_{\text{CLKPSC}}$ です。

図13-4. 4傾斜動作でのPSCr0とPSCr1基本波形



### 13.5.2.2. 2傾斜動作

2傾斜動作での全体周期は瞬時区間時間を与えるON時間0でのPSCr0記述に対する1瞬時区間と、瞬時区間時間を与えるON時間1でのPSCr1記述に対する1瞬時区間の2つの瞬時区間に分けられます。

PSCOUTr0とPSCOUTr1信号はON時間0(OT0)、沈黙時間0(DT0)、ON時間1(OT1)、沈黙時間1(DT1)の値によって定義されます。

$$\text{ON時間0} = (\text{OCrRrRA} - \text{OCrRrSA}) \times 1/f_{\text{CLKPSC}}$$

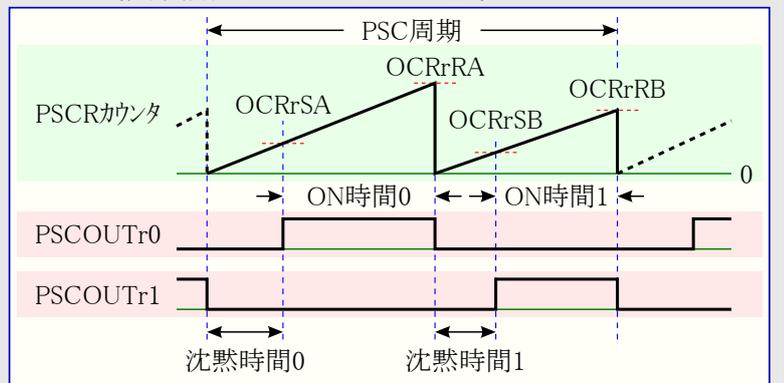
$$\text{ON時間1} = (\text{OCrRrRB} - \text{OCrRrSB}) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間0} = (\text{OCrRrSA} + 1) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間1} = (\text{OCrRrSB} + 1) \times 1/f_{\text{CLKPSC}}$$

**注:** 沈黙時間0と沈黙時間1の最小値は $1/f_{\text{CLKPSC}}$ です。

図13-5. 2傾斜動作でのPSCr0とPSCr1基本波形



### 13.5.2.3. 1傾斜動作

1傾斜動作でのPSCOUTr0とPSCOUTr1出力はお互いに重複できます(訳補:沈黙時間1が負の場合、図の赤線)。

$$\text{ON時間0} = (\text{OCrRrRA} - \text{OCrRrSA}) \times 1/f_{\text{CLKPSC}}$$

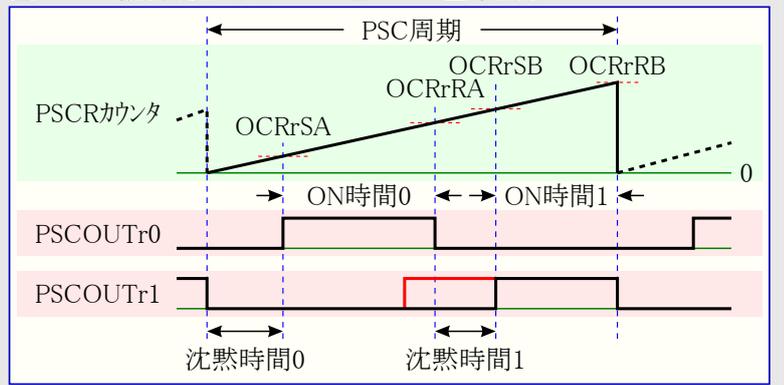
$$\text{ON時間1} = (\text{OCrRrRB} - \text{OCrRrSB}) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間0} = (\text{OCrRrSA} + 1) \times 1/f_{\text{CLKPSC}}$$

$$\text{沈黙時間1} = (\text{OCrRrSB} - \text{OCrRrRA}) \times 1/f_{\text{CLKPSC}}$$

**注:** 沈黙時間0の最小値は $1/f_{\text{CLKPSC}}$ です。

図13-6. 1傾斜動作でのPSCr0とPSCr1基本波形



### 13.5.3. 50%波形設定

PSCOUTr0とPSCOUTr1が同じ特性を持つ時に50%動作(PCNFrレジスタのPFIFTYr=1)でPSCRを設定することが可能です。PSCRがこの動作の時にOCrRrSBH,OCrRrSBL,OCrRrRBH,OCrRrRBLレジスタをOCrRrSAH,OCrRrSAL,OCrRrRAH,OCrRrRALレジスタに複写します。故にOCrRrSAH,OCrRrSAL,OCrRrRAH,OCrRrRALレジスタを設定する必要はありません。

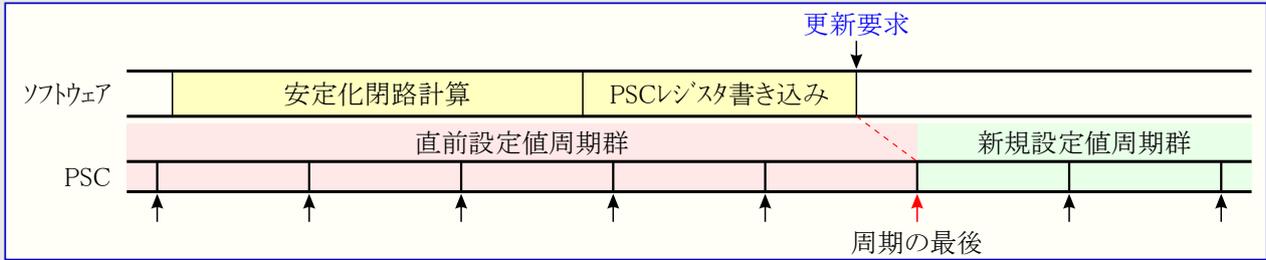
### 13.6. 値の更新

PSC波形レジスタ更新は以下の方法で行われます。

- PSC停止直後
- PSC走行時のPSC周期の最後
- 格納または自動格納の形態使用時に必要とされる条件に後続するPSC周期の最後

周期での非同時で矛盾する値を避けるため、各種値の1つの更新が必要な場合、PSCによって周期の終わりで同時に全ての値が更新されます。新規値群はソフトウェアによって計算され、更新はソフトウェアによって開始されます。

図13-7. 完全なPSCR周期の最後での更新



ソフトウェアは値更新を終える前に周期を停止し、新規PSCR周期を再び開始できます。

#### 13.6.1. 値の同期更新

新規タイミング値やPSCR出力構成はPSCR周期中に書けます。LOCK(PLOCKn)とAUTOLOCK(PALOCKn)構成ビットによって新規値群全体は以下の条件で考慮され(扱われ)ます。

- AUTOLOCK構成ビットが選択(=1)されると、PSCR内部レジスタの更新は比較Rレジスタ書き込みに後続するPSCR周期の最後で行なわれます。AUTOLOCK構成ビットはこの最初のPSCR周期の最後で考慮(処理)されます。
- LOCK構成ビットが設定(1)されると、PSCR内部レジスタの更新はLOCKビットが0に開放された場合のPSCR周期の最後で行なわれず。

LOCKとAUTOLOCKで同期更新されるレジスタはPSCR同期/出力構成(PSCrレジスタ、PSCR比較SA(OCRrSAH/L)レジスタ、PSCR比較RA(OCRrRAH/L)レジスタ、PSCR比較SB(OCRrSBH/L)レジスタ、PSCR比較RB(OCRrRBH/L)レジスタ)です。PSCrレジスタのPSC A部入力選択ビット1(PISELrA1)とPSC B部入力選択ビット1(PISELrB1)はPSC A部入力選択ビット0(PISELrA0)とPSC B部入力選択ビット0(PISELrB0)のように振舞うために直ちに更新されます。

設定(1)されると、AUTOLOCK設定ビットはLOCK設定ビットに勝ります。

103頁の「PCNF0 - PSC0構成レジスタ」をご覧ください。

### 13.7. 増強分解能

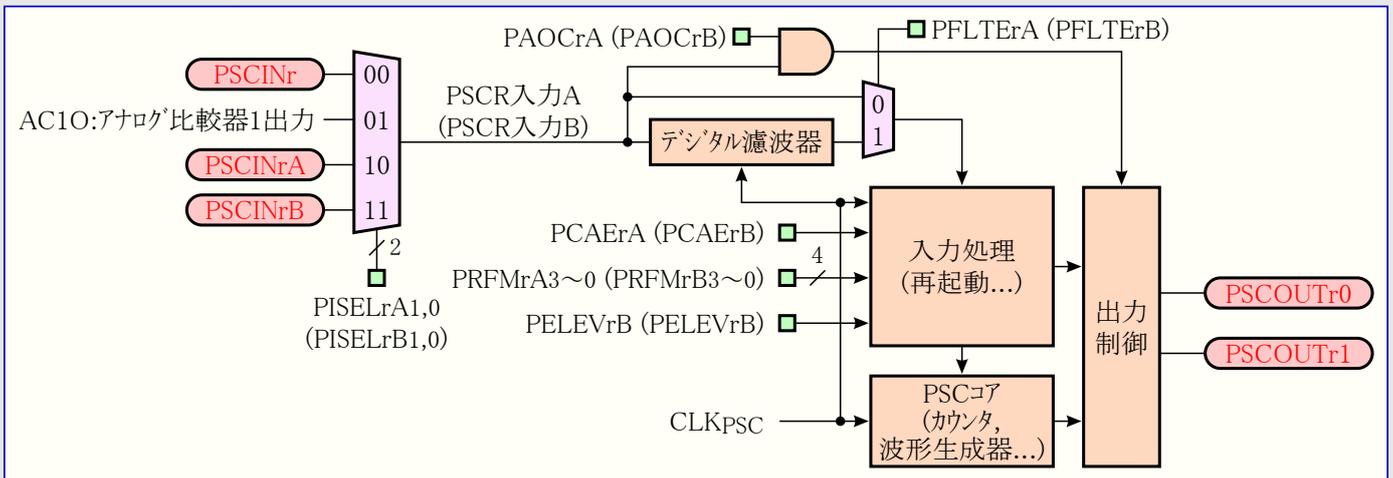
PSCRはPSCと同じ分解能増強を含みます。この機能の記述については67頁の「増強分解能」項をご覧ください。

### 13.8. 縮小電力段制御器(PSCR)入力

PSCRのA部とB部の各々は1つのPSCn内部入力を考慮するための自身の系統を持ちます。PSCR入力A/B制御(PFRCrA,PFRCrB)レジスタ(104頁をご覧ください)に従って、PSCRIN0/1M入力は再起動または異常入力として動くことができます。

このA系またはB系はこのPSCR入力A/B制御(PFRCrA/B)レジスタによっても構成設定されます。

図13-8. PSCR入力部



### 13.8.1. PSCR走行動作に対するPSCR再起動の動き

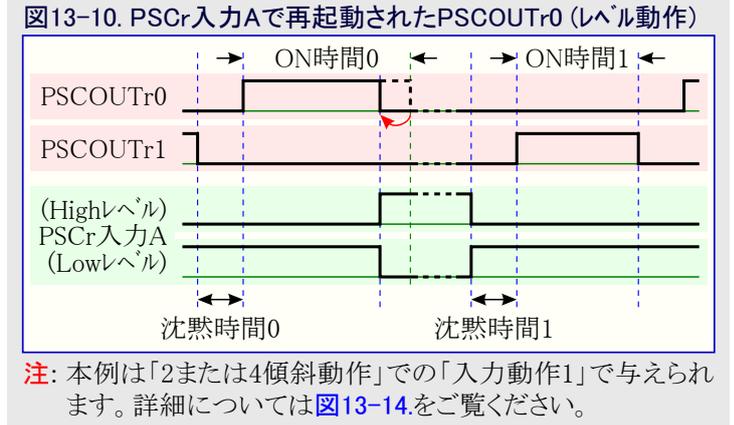
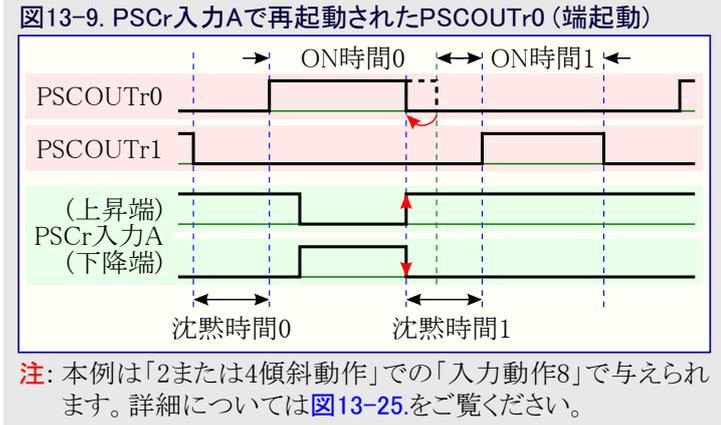
2傾斜動作または4傾斜動作での再起動入力AまたはBは、対応するAまたはB周期の終了と後続するBまたはA周期の開始を引き起こします。

1傾斜動作での再起動入力AまたはBは現在のPSCR計数を0にリセットします。

### 13.8.2. 外部事象でのPSCOUTr0再起動

PSCOUTr0出力はPSCR入力A(内部信号)の変更でのON時間0終了に先立ってリセットできます。PSCR入力Aは動作なし、レベルまたは端(エッジ)での動作に設定できます。PSCR入力Aの極性は感応制御部で設定可能です。PSCR入力AはPSCINr(外部)入力またはアナログ比較器1の出力にできます。

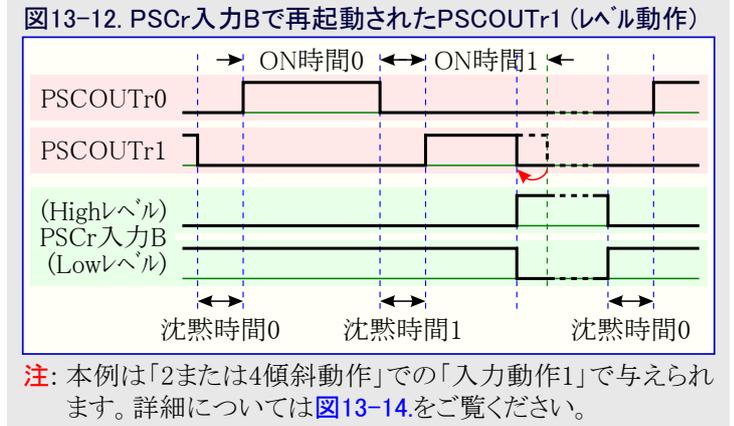
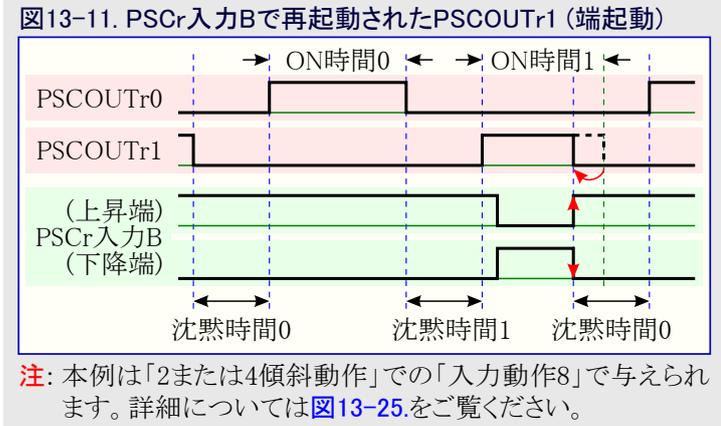
周期区間の減少のため、2つの出力周波数は一時的に増加します。



### 13.8.3. 外部事象でのPSCOUTr1再起動

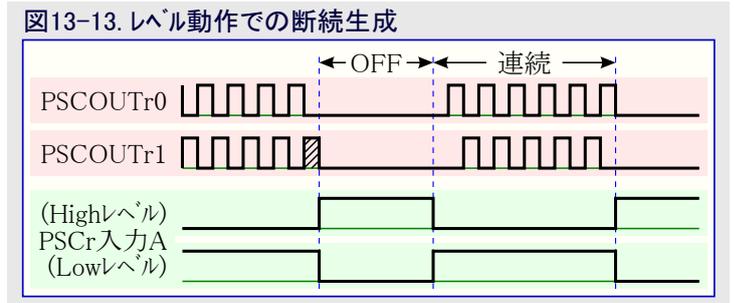
PSCOUTr1出力はPSCR入力B(内部信号)の変更でのON時間1終了に先立ってリセットできます。PSCR入力Bの極性は感応制御部で設定可能です。PSCR入力Bは動作なし、レベルまたは端(エッジ)での動作に設定できます。PSCR入力BはPSCINr(外部)入力またはアナログ比較器1の出力にできます。

周期区間の減少のため、2つの出力周波数は一時的に増加します。



#### 13.8.3.1. 断続生成

注: レベル動作では入力動作3または4の使用によって断続生成にPSCRを使うことが可能です(詳細については図13-18と図13-19をご覧ください)。



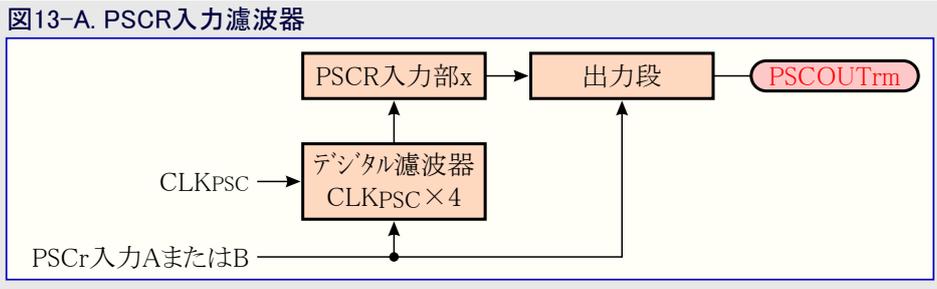
13.8.4. PSC入力設定

PSCR入力設定は設定レジスタ内のビット設定によって行なわれます。

13.8.4.1. 濾波器許可

濾波器許可(PFLTE0x)ビットが設定(1)されると、4周期のデジタル濾波器が信号評価の前に挿入されます。本機能の禁止は雑音除去が大きすぎる遅延を与えることになる前置分周されたPSCRクロックに対して主に必要とされます。

**重要:** デジタル濾波器が動作中の場合、出力を非活動にする(外部部品の緊急保護のため)に妨げられたPSCRクロックでも、レベル感知は正しく動作します。同様に障害入力として使われる時にPSCR入力AやBはPSCOUTr0/1を動かすのにPSCRを通さなければなりません。この経路はCLKPSCの走行を必要とします。故にPCTLrレジスタのPSCR非同期出力制御(PAOCrA,PAOCrB)ビットによってPSCINr入力は直接的にPSC出力を非活動にできます。この場合、入力はCLKPSCが走行すると直ぐに入力部によって通常として未だ考慮されます。



13.8.4.2. 信号極性

有効端(エッジ動作)または有効レベル(レベル動作)が選べます。104頁の「PFRC0A - PSCR入力A制御レジスタ」項のPELEV0xビット記述をご覧ください。

PELEV0xビットが設定(1)なら、PSCR入力AまたはBの意味ある(有効)端が上昇(端動作)か、または活性(有効)レベルがHigh(レベル動作)で、逆(解除(0))については下降/Lowです。

- 2または4傾斜動作でのPSCR入力Aは沈黙時間0とON時間0の周期の間だけ考慮されます(PSCR入力Bに対しては各々沈黙時間1とON時間1)。
- 1傾斜動作でのPSCR入力AまたはPSCR入力Bは傾斜全体で動作します。

13.8.4.3. 入力動作の操作

4つの設定ビット(PRFM0x3~0)で、PSCR入力の動作を定義することが可能です。これらの動作は表13-5.で一覧にされます。

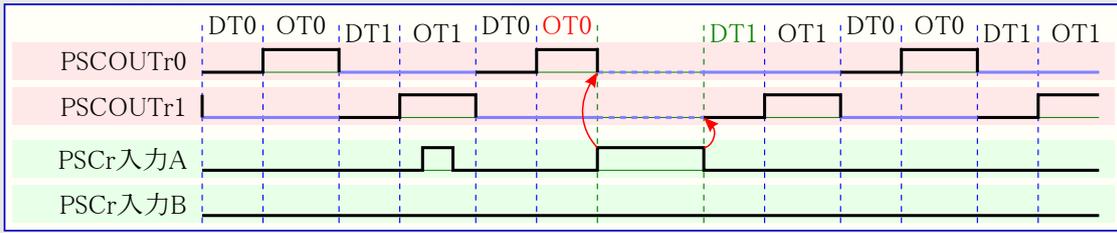
表13-5. PSCR入力動作の動き (注: x=A,B)

番号	PRFM0x3~0	説明
0	0 0 0 0	PSCR入力はPSCR出力での動作なし。
1	0 0 0 1	96頁の「PSCR入力動作1: 信号停止、反対側沈黙時間へ飛び、待機」をご覧ください。
2	0 0 1 0	96頁の「PSCR入力動作2: 信号停止、反対側沈黙時間実行、待機」をご覧ください。
3	0 0 1 1	97頁の「PSCR入力動作3: 信号停止、障害有効間反対側沈黙時間実行」をご覧ください。
4	0 1 0 0	97頁の「PSCR入力動作4: タイミング変更なしで出力非活動」をご覧ください。
5	0 1 0 1	98頁の「PSCR入力動作5: 信号停止、沈黙時間挿入」をご覧ください。
6	0 1 1 0	98頁の「PSCR入力動作6: 信号停止、反対側沈黙時間へ飛び、待機」をご覧ください。
7	0 1 1 1	98頁の「PSCR入力動作7: PSCR停止、ソフトウェア操作待機」をご覧ください。
8	1 0 0 0	98頁の「PSCR入力動作8: 端再起動PSCR」をご覧ください。
9	1 0 0 1	99頁の「PSCR入力動作9: 周波数固定端再起動PSCR」をご覧ください。
10	1 0 1 0	(予約): 使用不可
11	1 0 1 1	
12	1 1 0 0	
13	1 1 0 1	(予約): 使用不可
14	1 1 1 0	
15	1 1 1 1	

注: 後続の全ての例は上昇端またはHighレベルの活性(有効)入力で与えられます。

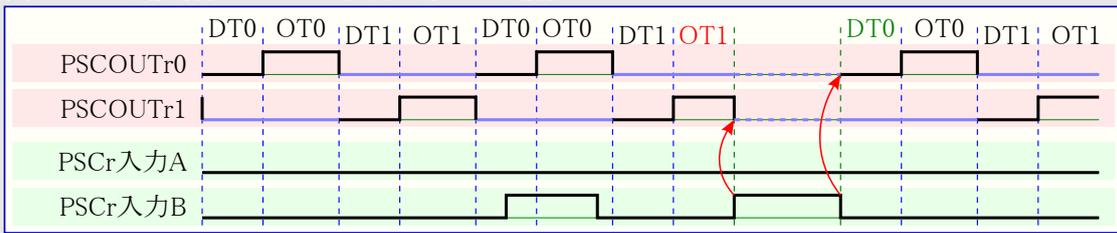
### 13.9. PSCr入力動作1 : 信号停止、反対側沈黙時間へ飛び、待機

図13-14. 障害動作1でのPSCr入力A対PSCr動作



PSCr入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSCr入力A事象が起こると、PSCrはPSCOUTr0を開放し、PSCr入力Aの非活性(無効)状態に対して待ち、そしてDT1とOT1へ飛んで実行します。

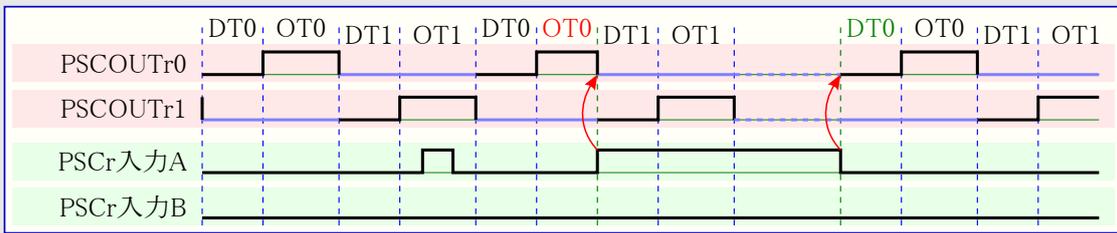
図13-15. 障害動作1でのPSCr入力B対PSCr動作



PSCr入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSCr入力B事象が起こると、PSCrはPSCOUTr1を開放し、PSCr入力Bの非活性(無効)状態に対して待ち、そしてDT0とOT0へ飛んで実行します。

### 13.10. PSCr入力動作2 : 信号停止、反対側沈黙時間実行、待機

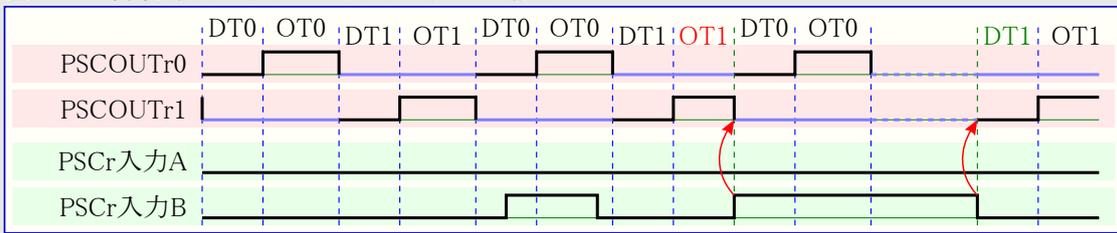
図13-16. 障害動作2でのPSCr入力A対PSCr動作



PSCr入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSCr入力A事象が起こると、PSCrはPSCOUTr0を開放し、DT1とOT1へ飛んで実行し、そしてPSCr入力Aの非活性(無効)状態を待ちます。

PSCr入力AがDT1またはOT1中に開放されても、DT1とOT1の(PSCr)半周期は常に全体が実行されます。

図13-17. 障害動作2でのPSCr入力B対PSCr動作

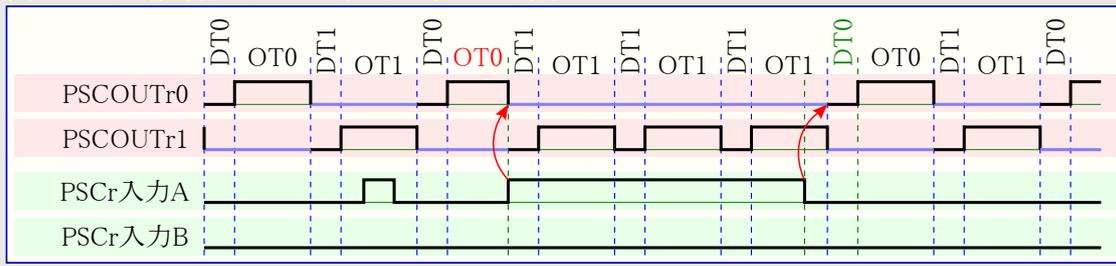


PSCr入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSCr入力B事象が起こると、PSCrはPSCOUTr1を開放し、DT0とOT0へ飛んで実行し、そしてPSCr入力Bの非活性(無効)状態を待ちます。

PSCr入力BがDT0またはOT0中に開放されても、DT0とOT0の(PSCr)半周期は常に全体が実行されます。

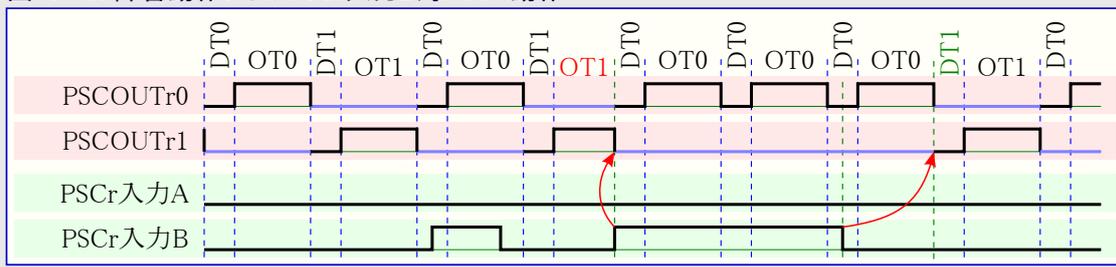
13.11. PSCR入力動作3 : 信号停止、障害有効間反対側パルス沈黙時間実行

図13-18. 障害動作3でのPSCR入力A対PSCR動作



PSCR入力Aは沈黙時間0(DT0)とON時間0(OT0)間だけ考慮され(扱われ)ます。沈黙時間1(DT1)とON時間1(OT1)間は無効です。PSCR入力A事象が起こると、PSCRはPSCOUTr0を開放し、PSCR入力Aが活性(有効)状態である間、DT1とOT1へ飛んで実行します。PSCR入力AがDT1またはOT1中に開放されても、DT1とOT1の(PSCR)半周期は常に全体が実行されます。

図13-19. 障害動作3でのPSCR入力B対PSCR動作



PSCR入力Bは沈黙時間1(DT1)とON時間1(OT1)間だけ考慮され(扱われ)ます。沈黙時間0(DT0)とON時間0(OT0)間は無効です。PSCR入力B事象が起こると、PSCRはPSCOUTr1を開放し、PSCR入力Bが活性(有効)状態である間、DT0とOT0へ飛んで実行します。PSCR入力BがDT0またはOT0中に開放されても、DT0とOT0の(PSCR)半周期は常に全体が実行されます。

13.12. PSCR入力動作4 : タイミング変更なしで出力非活動

図13-20. 障害動作4でのPSCR入力AまたはPSCR入力B対PSCR動作

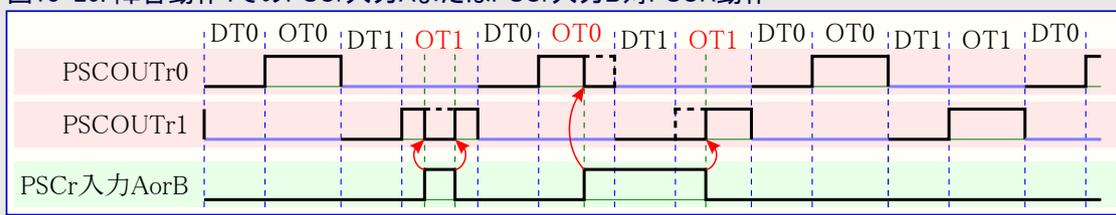
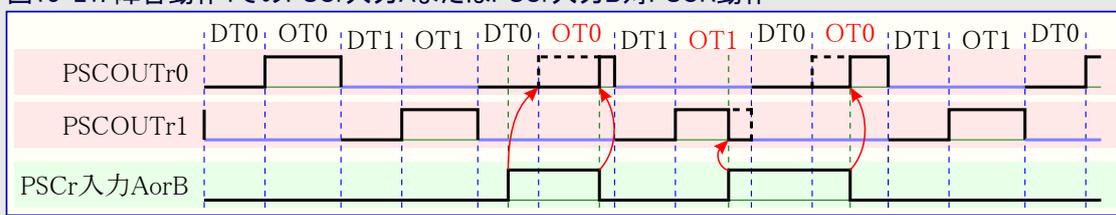


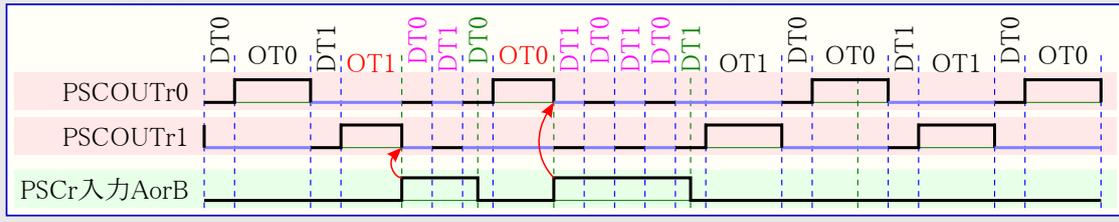
図13-21. 障害動作4でのPSCR入力AまたはPSCR入力B対PSCR動作



PSCR入力A及びPSCR入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

### 13.13. PSCR入力動作5 : 信号停止、沈黙時間挿入

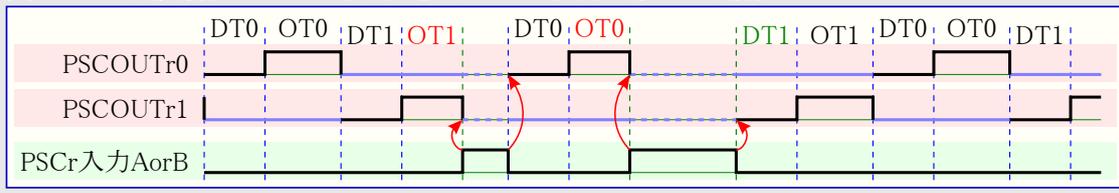
図13-22. 障害動作5でのPSCR入力AまたはPSCR入力B対PSCR動作



障害動作5使用でのPSCR入力A及びPSCR入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

### 13.14. PSCR入力動作6 : 信号停止、反対側沈黙時間へ飛び、待機

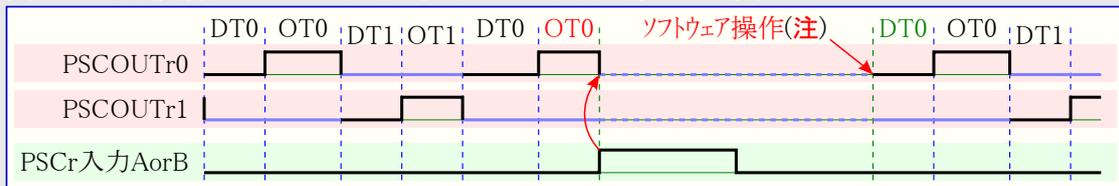
図13-23. 障害動作6でのPSCR入力AまたはPSCR入力B対PSCR動作



障害動作6使用でのPSCR入力A及びPSCR入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

### 13.15. PSCR入力動作7 : PSCR停止、ソフトウェア操作待機

図13-24. 障害動作7でのPSCR入力AまたはPSCR入力B対PSCR動作

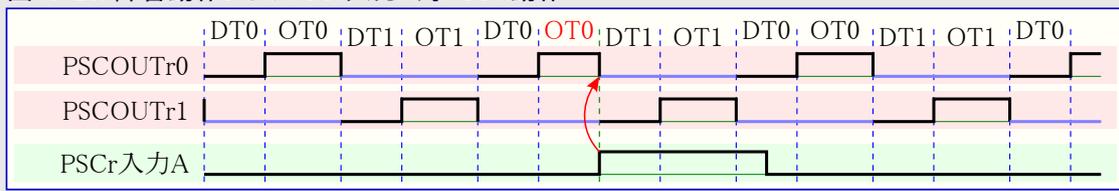


注: ソフトウェア操作はPSCR制御(PCTL0)レジスタのPRUNnビットの設定(1)です。

障害動作7使用でのPSCR入力A及びPSCR入力Bは沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

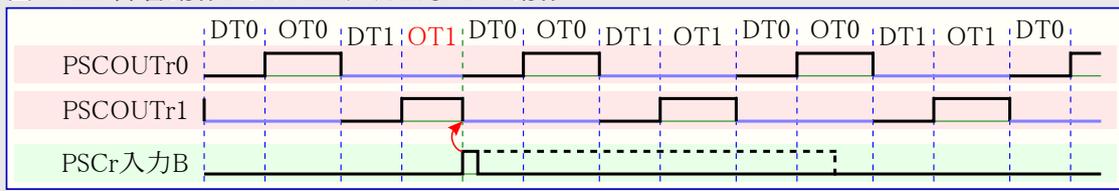
### 13.16. PSCR入力動作8 : 端再起動PSCR

図13-25. 障害動作8でのPSCR入力A対PSCR動作



出力周波数は再起動入力の意味ある(有効)端の発生によって変調されます。

図13-26. 障害動作8でのPSCR入力B対PSCR動作



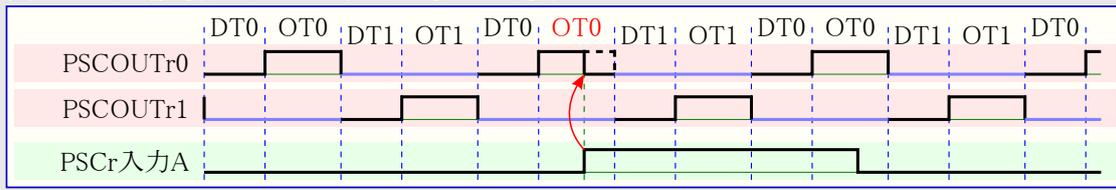
出力周波数は再起動入力の意味ある(有効)端の発生によって変調されます。

再起動事象は対応するON時間間に起きる場合にだけ考慮さ(扱)われます。

注: 1傾斜動作での入力Aの再起動事象は傾斜全体をリセットします。故にPSCRは逆側沈黙時間(DT1)に飛び(移行)しません。

13.17. PSCR入力動作9：周波数固定端再起動PSCR

図13-27. 障害動作9でのPSCR入力A対PSCR動作

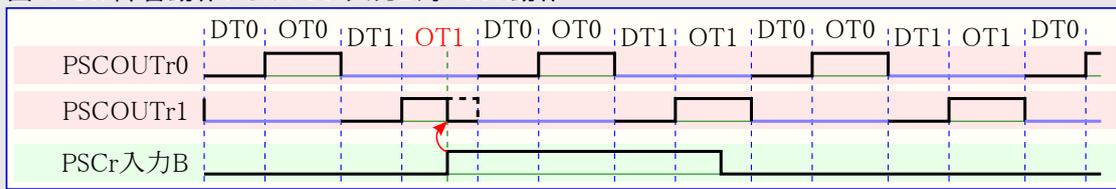


出力周波数は再起動入力の意味ある(有効)端の発生によって変調されません。

再起動入力の意味ある(有効)端が起きる時の出力だけが非活動(無効レベル)にされます。

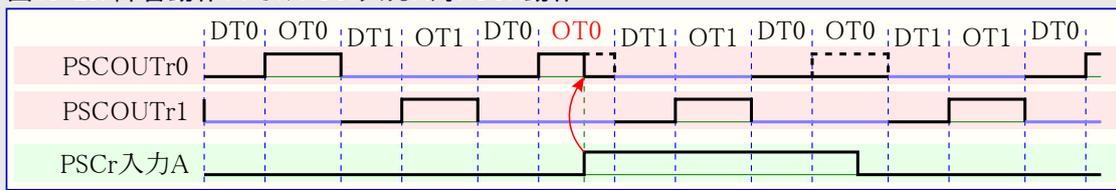
**注:** 本動作でのPSCR出力は例え再起動/障害入力が活性(有効)でも、次の傾斜間、活動(通常動作)になります。再起動/障害入力の意味ある(有効)端だけが考慮(扱)われます。

図13-28. 障害動作9でのPSCR入力B対PSCR動作



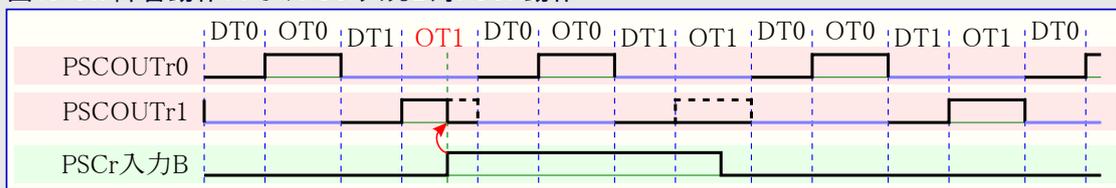
13.18. PSCR入力動作14：周波数固定端再起動PSCR、出力非活動

図13-29. 障害動作14でのPSCR入力A対PSCR動作



出力周波数は再起動入力の意味ある(有効)端の発生によって変調されません。

図13-30. 障害動作14でのPSCR入力B対PSCR動作



出力は再起動入力が活性(有効)である間、非活動(無効レベル)にされます。

PSCRの出力は非活性(無効レベル)状態に設定され、対応する傾斜(カウンタ)が中断停止されます。出力は再起動/障害入力が活性(有効)である間、非活性状態に留まります。このPSCRは一定周波数で走行します。

### 13.18.1. 走行動作による利用可能な入力動作

いくつかの入力動作はいくつかの走行動作で矛盾があります。故に表13-6は走行動作によって有効な入力動作を与えます。

表13-6. 走行動作による利用可能な入力動作

入力動作番号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
1傾斜動作	可	不可	不可	可	不可	不可	可	可	可	使用不可					可	使用不可	
2傾斜動作	可	可	可	可	可	可	可	可	可						可		可
4傾斜動作	可	可	可	可	可	可	可	可	可						可		

### 13.18.2. 事象捕獲

本PSCRはPSCR入力で再起動事象または障害事象が起きる時に時間(PSCRカウンタ)値を捕獲できます。この値はソフトウェアによってPSCR捕獲レジスタ(PICRrH,PICRrL)で読めます。

### 13.18.3. 捕獲入力部の使用

PSCR捕獲レジスタ(PICRr)でのPSCRカウンタ捕獲の起動には2つの方法があります。(訳注: 共通性から3行追加)

- PSCR入力信号によって起動されるハードウェア捕獲入力
- 捕獲ソフトウェア起動(PCSTr)ビットによって起動されるソフトウェア捕獲入力

捕獲入力部使用時の主な要求は到着事象を扱うための十分な能力をプロセッサに割り当てることです。2つの事象間の時間は危険です。プロセッサが次の事象発生前にPICRrレジスタ内の捕獲した値を読めなかったなら、PICRrは新しい値で上書きされるでしょう。この場合、捕獲結果は不正にされるでしょう。

捕獲入力割り込み使用時、PICRrレジスタ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。捕獲入力割り込みが相対的に高い優先順であっても、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かかる最大クロック周期数に依存します。

### 13.19. アナログ同期化

PSCRは採取&保持またはA/D変換開始と同期するための信号を生成します。同期化は測定にとって必須です。この信号はPSCOUTn0またはPSCOUTn1出力の全ての上昇端または下降端から選べます。

### 13.20. 割り込みの扱い

各PSCRは1つの機能について専用化でき、各PSCRにはそれ自身の割り込み系統(ベクタ等)があります。

割り込み元一覧:

- カウンタ再設定(=0、ON時間1(OT1)の最後)
- PSCR入力事象(設定された事象の有効な端(エッジ)またはレベルの開始)

### 13.21. PSCRクロック元

PSCRは増強分解能での高い周波数を生成できなければなりません。

各PSCRは2つのクロック入力を持ちます。

- PLLからのCLKPLL
- CLKI/O

PSCR構成(PCNFr)レジスタのPCLKSELrビットはクロック元選択に使われます。

PSCR制御(PCTLr)レジスタのPPREr1,0ビットはクロックの分周係数選択に使われます。

図13-31. クロック選択

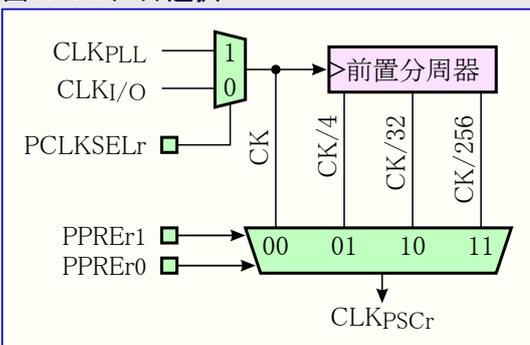


表13-7. クロック元と前置分周の選択

PCLKSELr	PPREr1	PPREr0	CLKpSCR出力
0	0	0	CLKI/O
0	0	1	CLKI/O/4
0	1	0	CLKI/O/32
0	1	1	CLKI/O/256
1	0	0	CLKPLL
1	0	1	CLKPLL/4
1	1	0	CLKPLL/32
1	1	1	CLKPLL/256

### 13.22. 割り込み

本項はAT90PWM81/161で行なわれる割り込み動作の仕様を記述します。

#### 13.22.1. 割り込みベクタの一覧

PSCRは3つの割り込みベクタを提供します。

- PSC0 EC (End of Cycle : 周期終了) : 許可で且つOCRrRBとの一致で発生
- PSC0 EEC (End of Enhanced Cycle : 増強周期終了) : 許可で且つ第15増強周期でのOCRrRBとの一致で発生
- PSC0 CAPT (Capture Event : 捕獲事象) : 許可で且つ、同期異常またはPSCRカウンタの再起動/捕獲で発生

105頁の「PIM0 – PSC0割り込み許可レジスタ」と106頁の「PIFR0 – PSC0割り込み要求フラグ レジスタ」をご覧ください。

### 13.23. 縮小電力段制御器(PSCR)用レジスタ

#### 13.23.1. PSOC0 – PSCR同期・出力構成レジスタ (PSCR Synchro and Output Configuration)

ビット (\$6A)	7	6	5	4	3	2	1	0	PSOC0
	PISEL0A1	PISEL0B1	PSYNC01	PSYNC00	–	POEN0B	–	POEN0A	
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 – PISEL0A1 : PSCR A部入力選択 (PSCR Input Select for part A)

PISEL0A0と共にPSCRのA部の有効な信号を定義します。

表13-8. PSCRの再起動/障害の入力選択

PISEL0A1	PISEL0A0	説明
0	0	PSCINr
0	1	アナログ比較器1出力
1	0	PSCINrA
1	1	PSCINrB

- ビット6 – PISEL0B1 : PSCR B部入力選択 (PSCR Input Select for part B)

PISEL0B0と共にPSCRのB部の有効な信号を定義します。

表13-9. PSCRの再起動/障害の入力選択

PISEL0B1	PISEL0B0	説明
0	0	PSCINr
0	1	アナログ比較器1出力
1	0	PSCINrA
1	1	PSCINrB

- ビット5,4 – PSYNC01,0 : A/D変換器用同期出力選択 (Synchronization Out for ADC Selection)

同期化用にA/D変換器へ送る信号を生成する信号元と極性を選びます。

表13-10. 1,2,4傾斜動作での同期元

PSYNC01	PSYNC00	説明
0	0	PSCOUTR0の先行端(OCR0SAとの比較一致)で信号送出。
0	1	PSCOUTR0の後行端(OCR0RAとの比較一致またはA部での障害/再起動)で信号送出。
1	0	PSCOUTR1の先行端(OCR0SBとの比較一致)で信号送出。
1	1	PSCOUTR1の後行端(OCR0RBとの比較一致またはB部での障害/再起動)で信号送出。

- ビット3 – Res : 予約 (Reserved Bit)

このビットは予約されています。

- ビット2 – POEN0B : PSCOUTR1/B部出力許可 (PSCR OUT Part B Output Enable)

このビットが解除(0)されると、PSCOUTR1に影響を及ぼすI/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUTR1に影響を及ぼすI/OピンはPSCR波形生成器Bへ接続され、PSCR操作に従って設定(1)と解除(0)を行ないます。

•ビット1 – Res : 予約 (Reserved Bit)

このビットは予約されています。

•ビット0 – POEN0A : PSCOUTR0/A部出力許可 (PSCR OUT Part A Output Enable)

このビットが解除(0)されると、PSCOUTR0に影響を及ぼすI/Oピンは標準ポートとして動作します。

このビットが設定(1)されると、PSCOUTR0に影響を及ぼすI/OピンはPSCR波形生成器Aへ接続され、PSCR操作に従って設定(1)と解除(0)を行いません。

13.23.2. OCR0SAH,OCR0SAL (OCR0SA) – PSCR比較SAレジスタ (Output Compare SA Register)

ビット (\$61)	15	14	13	12	11	10	9	8	OCR0SAH
	-				OCR0SA11~8				
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット (\$60)	7	6	5	4	3	2	1	0	OCR0SAL
	OCR0SA7~0								
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

13.23.3. OCR0RAH,OCR0RAL (OCR0RA) – PSCR比較RAレジスタ (Output Compare RA Register)

ビット \$2B (\$4B)	15	14	13	12	11	10	9	8	OCR0RAH
	-				OCR0RA11~8				
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット \$2A (\$4A)	7	6	5	4	3	2	1	0	OCR0RAL
	OCR0RA7~0								
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

13.23.4. OCR0SBH,OCR0SBL (OCR0SB) – PSCR比較SBレジスタ (Output Compare SB Register)

ビット \$23 (\$43)	15	14	13	12	11	10	9	8	OCR0SBH
	-				OCR0SB11~8				
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット \$22 (\$42)	7	6	5	4	3	2	1	0	OCR0SBL
	OCR0SB7~0								
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

13.23.5. OCR0RBH,OCR0RBL (OCR0RB) – PSCR比較RBレジスタ (Output Compare RB Register)

ビット \$25 (\$45)	15	14	13	12	11	10	9	8	OCR0RBH
	OCR0RB15~12				OCR0RB11~8				
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	
ビット \$24 (\$44)	7	6	5	4	3	2	1	0	OCR0RBL
	OCR0RB7~0								
Read/Write	W	W	W	W	W	W	W	W	
初期値	0	0	0	0	0	0	0	0	

PSCR比較RA, RB, SA, SBレジスタは継続的にPSCRカウンタ値と比較される12ビット値を含みます。一致は比較一致割り込みや、関係ピンでの波形出力生成に使えます。

比較RBレジスタは分数(端数)変調(小数分周器)に使われる4ビット値も含みます。

比較レジスタは12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、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16ビットまたは12ビットの全レジスタで共用されません。

## 13.23.6. PCNF0 – PSCR構成レジスタ (PSCR Configuration Register)

ビット	7	6	5	4	3	2	1	0	
\$11 (\$31)	PFIFTY0	PALOCK0	PLOCK0	PMODE01	PMODE00	POP0	PCLKSEL0	–	PCNF0
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 – PFIFTY0 : PSCR 50%動作 (PSCR Fifty)

このビットの1書き込みはOCR0RBとOCR0SBだけが使われる50%動作にPSCRを設定します。OCR0RBの更新中に、これら(の値)はOCR0RAとOCR0SAに複写されます。この機能は50%波形実行に有用です。

### • ビット6 – PALOCK0 : PSCR 自動格納 (PSCR Autolock)

このビットが設定(1)されると、比較RA,SA,SBレジスタ、PSCR同期/出力構成(PSOC0)レジスタはPSCR周期を妨げることなく書けます。PSCR内部レジスタの更新は比較Rレジスタが書かれてしまう時に、そのPSCR周期の最後で行なわれます。

設定(1)時、本ビットはLOCK0ビット(ビット5)に優先します。

### • ビット5 – PLOCK0 : PSCR 格納 (PSCR Lock)

このビットが設定(1)されると、比較RA,RB,SA,SBレジスタ、PSCR同期/出力構成(PSOC0)レジスタはPSCR周期を妨げることなく書けます。PSCR内部レジスタの更新はLOCKビットが0に解除される時に行なわれます。

### • ビット4,3 – PMODE01,0 : PSCR 走行動作種別 (PSCR Mode)

PSCRの動作を選びます。

表13-11. PSCR走行動作種別選択

PMODEn1	PMODEn0	走行動作種別
0	0	1傾斜動作
0	1	2傾斜動作
1	0	4傾斜動作
1	1	(予約)

### • ビット2 – POP0 : PSCR 出力極性 (PSCR Output Polarity)

このビットが解除(0)されている場合、PSCR出力はLow有効です。

このビットが設定(1)されている場合、PSCR出力はHigh有効です。

### • ビット1 – PCLKSEL0 : PSCR 入力クロック選択 (PSCR Input Clock Select)

このビットは高速クロックまたは低速クロックのどちらかを選びます。

高速クロック(CLKPLL)を選ぶにはこのビットを設定(1)してください。

低速クロック(CLKI/O)を選ぶにはこのビットを解除(0)してください。

### • ビット0 – Res : 予約 (Reserved Bit)

このビットは予約されています。

## 13.23.7. PSCR制御レジスタ (PSCR Control Register) PCTL0

ビット	7	6	5	4	3	2	1	0	
\$12 (\$32)	PPRE01	PPRE00	PBFM01	PAOC0B	PAOC0A	PBFM00	PCCYC0	PRUN0	PCTL0
Read/Write	R/W	R/W							
初期値	0	0	0	0	0	0	0	0	

### • ビット7,6 – PPRE01,0 : PSCR 前置分周選択 (PSCR Prescaler Select)

この2ビットはPSCR入力クロック分周係数を選びます。全ての生成波形がこの係数によって変更されます。

表13-12. PSCR前置分周選択

PPRE01	PPRE00	前置分周
0	0	なし(1/1)
0	1	4分周
1	0	32分周
1	1	256分周

- ビット5 – PBFM01 : PSCR平衡分数(端数)変調(小数分周器)動作種別ビット1 (PSCR Balance Flank Width Modulation bit 1)

PBFMn0ビットと共に平衡分数変調を定義します。

表13-13. 分数変調動作形態選択

PBFM01	PBFM00	説明
0	0	RB(ON時間1のみ)で分数変調動作
0	1	RB+RA(ON時間0とON時間1)で分数変調動作
1	0	SB(沈黙時間1のみ)で分数変調動作 (注)
1	1	SB+SA(沈黙時間0と沈黙時間1)で分数変調動作

注: 1傾斜動作ではSAやSA+SBの変更もON時間に影響を及ぼします。1傾斜動作でのPSCOUT出力の基本波形をご覧ください。

- ビット4 – PAOC0B : PSCR 非同期出力制御B (PSCR Asynchronous Output Control B)

このビットが設定(1)されると、B部へ選ばれた障害入力PSCOUTR1出力を直接的に操作できます。95頁の「PSCR入力設定」をご覧ください。

- ビット3 – PAOC0A : PSCR 非同期出力制御A (PSCR Asynchronous Output Control A)

このビットが設定(1)されると、A部へ選ばれた障害入力PSCOUTR0出力を直接的に操作できます。95頁の「PSCR入力設定」をご覧ください。

- ビット2 – PBFM00 : PSCR平衡分数(端数)変調(小数分周器)動作種別ビット0 (PSCR Balance Flank Width Modulation bit 0)

PBFMn1ビットと共に平衡分数変調を定義します。

- ビット1 – PCCYC0 : PSCR 周期完了許可 (PSCR Complete Cycle)

このビットが設定(1)されると、PSCRはPRUN0の解除(0)によって要求した停止操作に先立って波形周期全体を完了します。

- ビット0 – PRUN0 : PSCR 走行許可 (PSCR Run)

このビットの1書き込みはPSCRを始動します。

### 13.23.8. PFRC0A – PSCR入力A制御レジスタ (PSCR Input A Control Register)

ビット (\$66)	7	6	5	4	3	2	1	0	
	PCAE0A	PISEL0A0	PELEV0A	PFLTE0A	PRFM0A3	PRFM0A2	PRFM0A1	PRFM0A0	PFRC0A
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### 13.23.9. PFRC0B – PSCR入力B制御レジスタ (PSCR Input B Control Register)

ビット (\$67)	7	6	5	4	3	2	1	0	
	PCAE0B	PISEL0B0	PELEV0B	PFLTE0B	PRFM0B3	PRFM0B2	PRFM0B1	PRFM0B0	PFRC0B
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	

入力制御レジスタはPSCRの2つ(A部とB部)の再起動/障害部構成に使われます。この2つの単位部は同一(構造)で、従って同じ方法で設定されます。(訳補: 以下のxはAまたはBです。)

- ビット7 – PCAE0x : PSCR x部入力捕獲許可 (PSCR Capture Enable Input Part x)

このビットの1書き込みはx部に対する入力として選んだ入力外部事象が起こる時の捕獲機能を許可します(次のPISEL0x0ビットをご覧ください)。

- ビット6 – PISEL0x0 : PSCR x部入力選択 (PSCR Input Select for Part x)

PSCR同期・出力構成(PSOC0)レジスタのPISEL0x1ビットと共にPSCR A部の活性(有効)信号を定義します。101頁の表13-8と表13-19をご覧ください。

- ビット5 – PELEV0x : PSCR 非同期出力制御x (PSCR Asynchronous Output Control x)

このビットが解除(0)されると、選んだ入力の下降かLowレベルが障害または再起動機能に対して意味ある事象を生成します。このビットが設定(1)されると、選んだ入力の上昇かHighレベルが障害または再起動機能に対して意味ある事象を生成します。

- ビット4 – PFLTE0x : PSCR x部入力濾波器許可 (PSCR Filter Enable on Input Part x)

このビットの設定(1)は捕獲入力雑音消去器を活性(有効)にします。雑音消去器が活動(状態)にされると、再起動ピンからの入力は濾波されます。この濾波機能はその出力を変更するのに再起動ピンの4連続同一値採取を必要とします。従って捕獲入力は雑音消去器が許可される時に4発振器周期遅らされます。

•ビット3~0 – PRFM0x3~0 : PSCR 障害/再起動 x部動作選択 (PSCR Fault Mode)

これらの4ビットは障害または再起動機能の動作種別を定義します。(より多くの説明については95頁の表13-5をご覧ください。)

表13-14. 障害/再起動動作種別とPSC入力動作 (端(エッジ)/レベル選択)

番号	PRFM0x3~0	説明
0	0 0 0 0	動作なし、PSCR入力は無効。
1	0 0 0 1	PSCR入力動作1 : 信号停止、反対側沈黙時間へ飛び、待機
2	0 0 1 0	PSCR入力動作2 : 信号停止、反対側沈黙時間実行、待機
3	0 0 1 1	PSCR入力動作3 : 信号停止、障害有効間反対側沈黙時間実行
4	0 1 0 0	PSCR入力動作4 : タイミング変更なしで出力非活動
5	0 1 0 1	PSCR入力動作5 : 信号停止、沈黙時間挿入
6	0 1 1 0	PSCR入力動作6 : 信号停止、反対側沈黙時間へ飛び、待機
7	0 1 1 1	PSCR入力動作7 : PSCR停止、ソフトウェア操作待機
8	1 0 0 0	PSCR入力動作8 : 端再起動PSCR
9	1 0 0 1	PSCR入力動作9 : 周波数固定端再起動PSCR
10	1 0 1 0	(予約) : 使用不可
11	1 0 1 1	
12	1 1 0 0	
13	1 1 0 1	
14	1 1 1 0	PSCR入力動作14 : 周波数固定端再起動PSCR、出力非活動
15	1 1 1 1	(予約) : 使用不可

### 13.23.10. PICR0H, PICR0L (PICR0) – PSCR捕獲レジスタ (PSCR Input Capture Register)

ビット (\$69)	15	14	13	12	11	10	9	8	
	PCST0	-	-	-	PICR011~8				PICR0H
Read/Write	R/W	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット (\$68)	7	6	5	4	3	2	1	0	
	PICR07~0								PICR0L
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

•ビット7 – PCST0 捕獲ソフトウェア起動 (PSCR Capture Software Trig bit)

PSCRカウンタの捕獲を起こすにはこのビットを設定(1)してください。読み込み時、本ビットが設定(1)されていれば捕獲操作はPCST0の設定(1)によって起動されたことを意味し、さもなければ捕獲操作はPSC入力によって起動されたことを意味します。

捕獲入力は捕獲機能が許可される(PFRC0xレジスタのPCAE0xビットが設定(1)される)場合の、許可した入力ピン(または任意選択のアナログ比較器1出力)での事象発生毎にPSCRカウンタ値で更新されます。

捕獲レジスタは12ビットの容量です。CPUがこれらのレジスタをアクセスする時に上位と下位のバイトが同時に読まれるのを保証するために、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16ビットまたは12ビットの全レジスタで共有されます。

### 13.23.11. PIM0 – PSCR割り込み許可レジスタ (PSCR Interrupt Mask Register)

ビット \$0F (\$2F)	7	6	5	4	3	2	1	0	
	-	-	-	PEVE0B	PEVE0A	-	PEOEPE0	PEOPE0	PIM0
Read/Write	R	R	R	R/W	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

•ビット7~5 – Res : 予約 (Reserved Bit)

これらのビットは予約されています。

•ビット4 – PEVE0B : PSCR B部外部事象割り込み許可 (PSCR External Event B Interrupt Enable)

このビットが設定(1)されると、B部の障害/再起動部からの捕獲を発生できる外部事象は割り込みも生成します。

•ビット3 – PEVE0A : PSCR A部外部事象割り込み許可 (PSCR External Event A Interrupt Enable)

このビットが設定(1)されると、A部の障害/再起動部からの捕獲を発生できる外部事象は割り込みも生成します。



• **ビット2 – Res : 予約 (Reserved Bit)**

このビットは予約されています。

• **ビット1 – PEOPE0 : PSCR 増強周期終了割り込み許可 (PSCR End Of Enhanced Cycle Interrupt Enable)**

このビットが設定(1)されると、PSCRが第15PSCR周期の最後に達する時に割り込みが生成されます。これは割り込みルーチンでPSCR値を更新して次のPSCR周期の最後で新しい値の新しい増強周期を開始することを許します。

• **ビット0 – PEOPE0 : PSCR 周期終了割り込み許可 (PSCR End Of Cycle Interrupt Enable)**

このビットが設定(1)されると、PSCRが周期全体(PSCR周期)の最後に達する時に割り込みが生成されます。

**13.23.12. PIFR0 – PSCR割り込み要求フラグ レジスタ (PSCR Interrupt Flag Register)**

ビット	7	6	5	4	3	2	1	0	
\$10 (\$30)	POAC0B	POAC0A	–	PEV0B	PEV0A	PRN01	PRN00	PEOP0	PIFR0
Read/Write	R	R	R	R/W	R/W	R	R	R/W	
初期値	0	0	0	0	0	0	0	0	

• **ビット7 – POAC0B : PSCR B部出力活動 (PSCR Output B Activity)**

このビットはPSCOUTR1出力が0⇒1または1⇒0に変化する毎にハードウェアによって設定(1)されます。

これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

この機能は固定化した外部入力信号のためにPSCR出力が変化しないことを検知するのに有用です。

• **ビット6 – POAC0A : PSCR A部出力活動 (PSCR Output A Activity)**

このビットはPSCOUTR0出力が0⇒1または1⇒0に変化する毎にハードウェアによって設定(1)されます。

これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

この機能は固定化した外部入力信号のためにPSCR出力が変化しないことを検知するのに有用です。

• **ビット5 – Res : 予約 (Reserved Bit)**

このビットは予約されています。

• **ビット4 – PEV0B : PSCR B部外部事象割り込み要求フラグ (PSCR External Event B Interrupt)**

このビットはB部の障害/再起動部からの捕獲または再起動を発生できる外部事象が起こる時に設定(1)されます。

これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

このビットは例え対応する割り込みが許可されてなくても(PEVE0Bビット=0)、読めます。

• **ビット3 – PEV0A : PSCR A部外部事象割り込み要求フラグ (PSCR External Event A Interrupt)**

このビットはA部の障害/再起動部からの捕獲または再起動を発生できる外部事象が起こる時に設定(1)されます。

これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

このビットは例え対応する割り込みが許可されてなくても(PEVE0Aビット=0)、読めます。

• **ビット2,1 – PRN01,0 : PSCR 傾斜番号 (PSCR Ramp Number)**

最後のA部外部事象割り込み要求(PEV0A)またはB部外部事象割り込み要求(PEV0B)発生時の傾斜番号の記録です。

表13-15. PEV0x発生時傾斜番号

PRN01,0	00	01	10	11
傾斜番号	0	1	2	3

• **ビット0 – PEOPE0 : PSCR 周期終了割り込み要求フラグ (PSCR End Of Cycle Interrupt)**

このビットは周期全体(PSCR周期)を達成した時に設定(1)されます。

これは本位置に1を書くことでソフトウェアによって解除(0)されなければなりません。

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

### 14.1. 特徴

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

### 14.2. 概要

直列周辺インターフェースはAT90PWM81/161と様々なAVRデバイスや周辺装置間的高速同期データ転送を許します。

31頁の「PRR - 電力削減レジスタ」のP<sub>RSPI</sub>ビットはSPI部を許可するために0を書かれなければなりません。

SPIでの主装置と従装置のCPU間相互連結は図14-2で示されます。このシステムは2つの移動レジスタと主装置クロック発生器から成ります。SPI主装置は希望した従装置のSS(従装置選択)ピンをLowへ引き込む時に一群の通信を開始します。主装置と従装置は各々の移動レジスタに送出すべきデータを用意し、主装置はデータを交換する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へ引き上げるることによって従装置と同期を取ります。

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

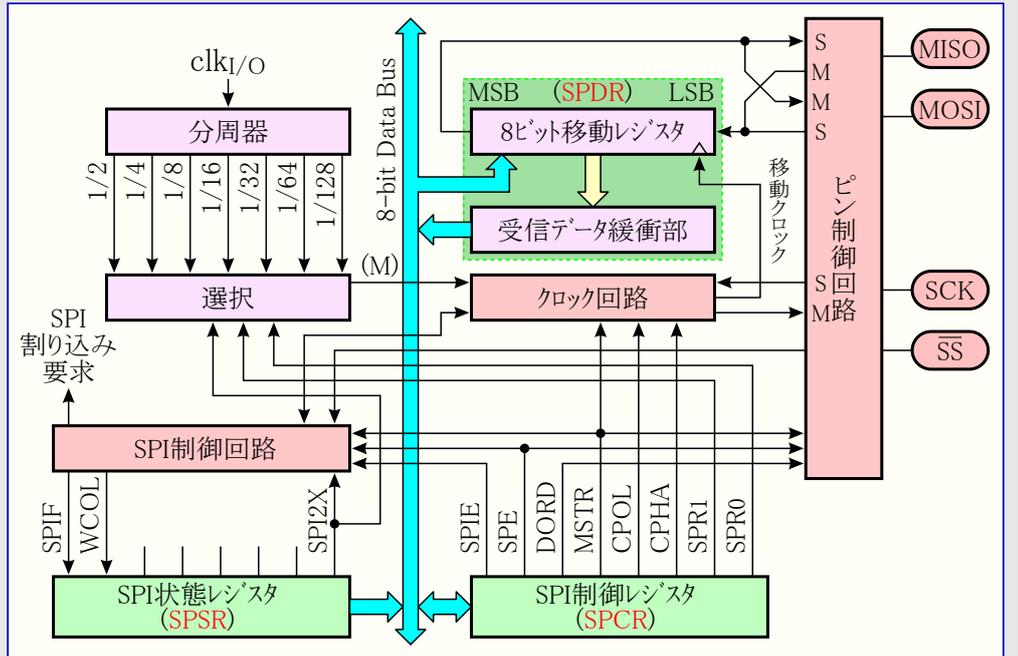
従装置として設定されると、SPIインターフェースはSSピンがHighに駆動される限り、MISOをHi-Zにした休止状態に留まります。この状態でプログラムはSPIデータレジスタ(SPDR)の内容を更新できますが、そのデータはSSピンがLowに駆動されるまでSCKピンでの到着クロックパルスによって移動出力されません。1バイトが完全に移動されてしまうと転送完了フラグ(SPIF)が設定(1)されます。SPCRでSPI割り込み許可(SPIE)ビットが設定(1)されていれば割り込みが要求されます。従装置は受信データを読む前にSPDR内へ送られるべき次のデータの配置を続けられます。最後の到着バイトはその後の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

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

SPI従装置動作では制御論理回路がSCKピンの到着信号を採取します。このクロック信号の正しい採取を保証するため、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14-1に従って無視されます。自動的なポート無視のより多くの詳細については45頁の「交換ポート機能」を参照してください。

図14-1. SPI構成図



注: SPIピン配置については2頁の「ピン配置」と46頁の表9-3を参照してください。

図14-2. SPI 主装置/従装置の連結

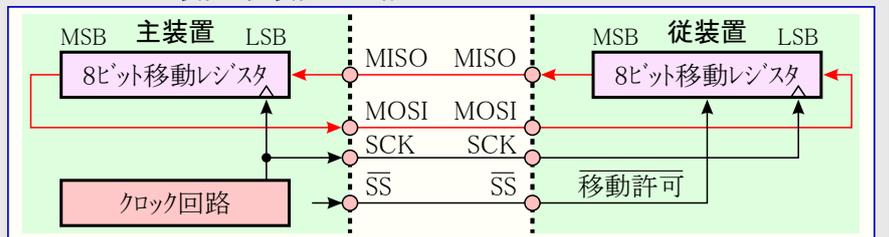


表14-1. SPIピン方向規定

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

注: 使用者定義SPIピンの方向定義方法の詳細記述については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PB2ピンに配置されるなら、DD\_MOSIはDDB2、DDR\_SPIはDDRBに置き換ええます。

#### アセンブリ言語プログラム例

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

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

#### C言語プログラム例

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

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

**注:** このコード例はデバイス固有ヘッダファイルがインクルードされていると仮定します。

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:** このコード例はデバイス固有ヘッダファイルがインクルードされていると仮定します。

### 14.3. $\overline{SS}$ ピンの機能

#### 14.3.1. 従装置動作

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

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

#### 14.3.2. 主装置動作

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

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

$\overline{SS}$ が入力として設定されると、SPI主装置動作を保証するため、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。

### 14.4. データ転送形式

直列データに関してはSPI制御レジスタ(SPCR)のSCK位相(CPHA)とSCK極性(CPOL)制御ビットによって決定されるSCK位相と極性で4つの組み合わせがあります。このSPIデータ転送形式は図14-3と図14-4.で示されます。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。これは下表で行われるように表14-3.と表14-4.を要約することによって明解にされます。

表14-2. CPOL,CPHA機能動作

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

図14-3. SPIデータ転送形式 (CPHA=0)

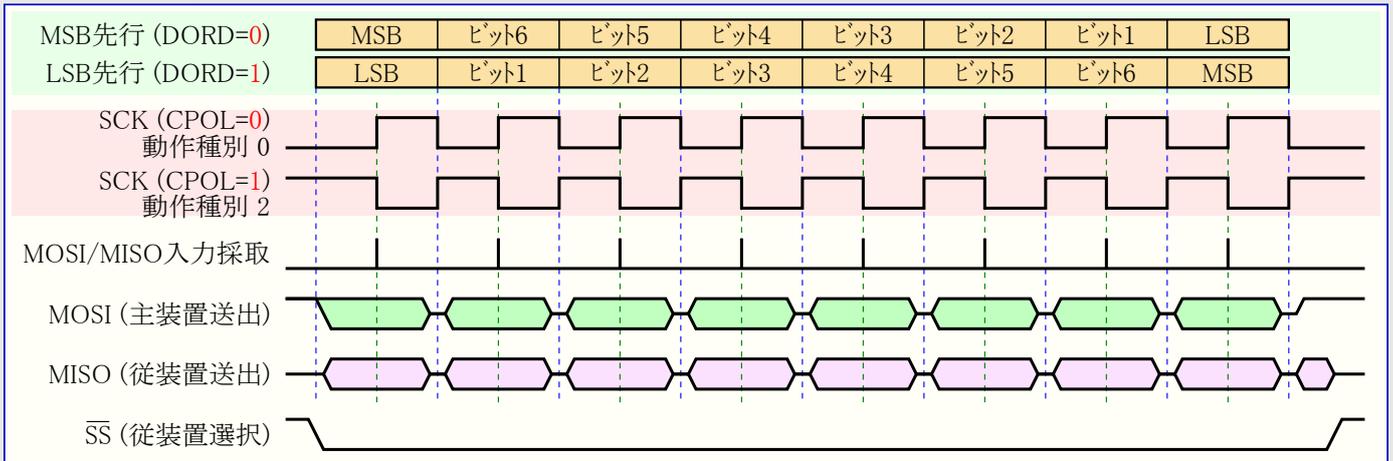
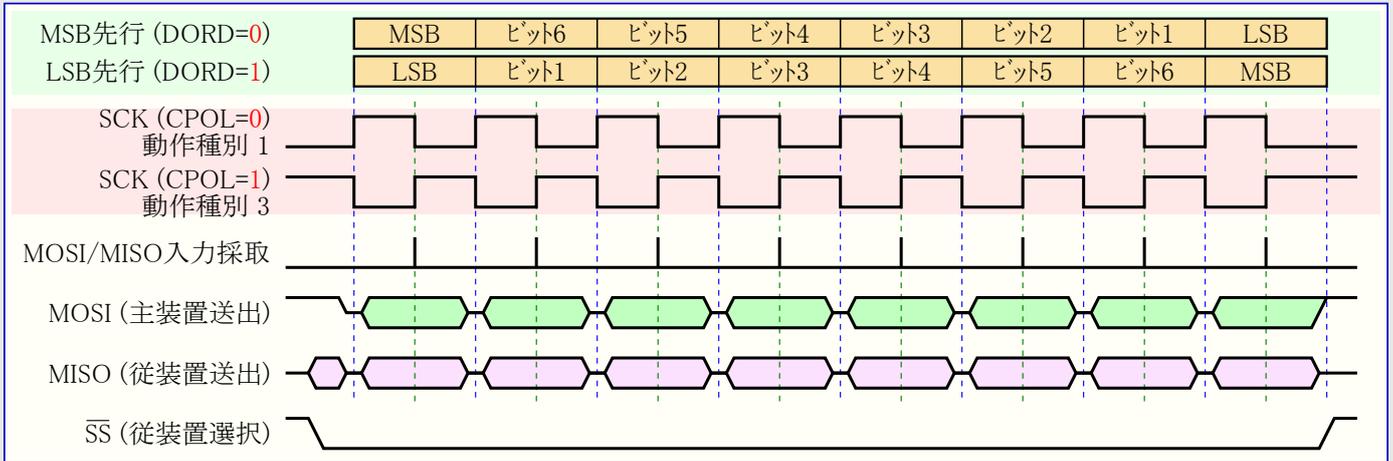


図14-4. SPIデータ転送形式 (CPHA=1)



## 14.5. SPI用レジスタ

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

ビット	7	6	5	4	3	2	1	0	
\$17 (\$37)	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14-3と図14-4を参照してください。CPOL機能は右で要約されます。

表14-3. CPOL機能動作

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

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

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

表14-4. CPHA機能動作

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

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

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

表14-5. 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	

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

ビット	7	6	5	4	3	2	1	0	
\$18 (\$38)	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14-5参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定される時に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。

AT90PWM81/161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158頁をご覧ください。

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

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

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

## 15. 電圧基準と温度感知器

### 15.1. 特徴

- 2.56Vの正確な基準電圧
- 内部温度感知器
- 電圧基準とチップ上の発振器の両方での温度変動の走行時補償の可能性
- 低消費電力

### 15.2. チップ上の電圧基準と温度感知器の概要

低電力バンドギャップ電圧基準はAT90PWM81/161に正確な1.100Vのチップ上バンドギャップ電圧(VBG)を提供します。

そしてSW1がOFFでSW2とSW3がONの時にバンドギャップ電圧は乗算され、2.56Vの内部基準電圧(VREF)を生成します。この基準電圧はA/D変換器とD/A変換器の基準として使われ、**図15-1**で示されるように最小の消費電力で優秀な雑音性能を許すために、(SW0がONの時に)外部デカップ(雑音分離)コンデンサでの緩衝を用いることができます。

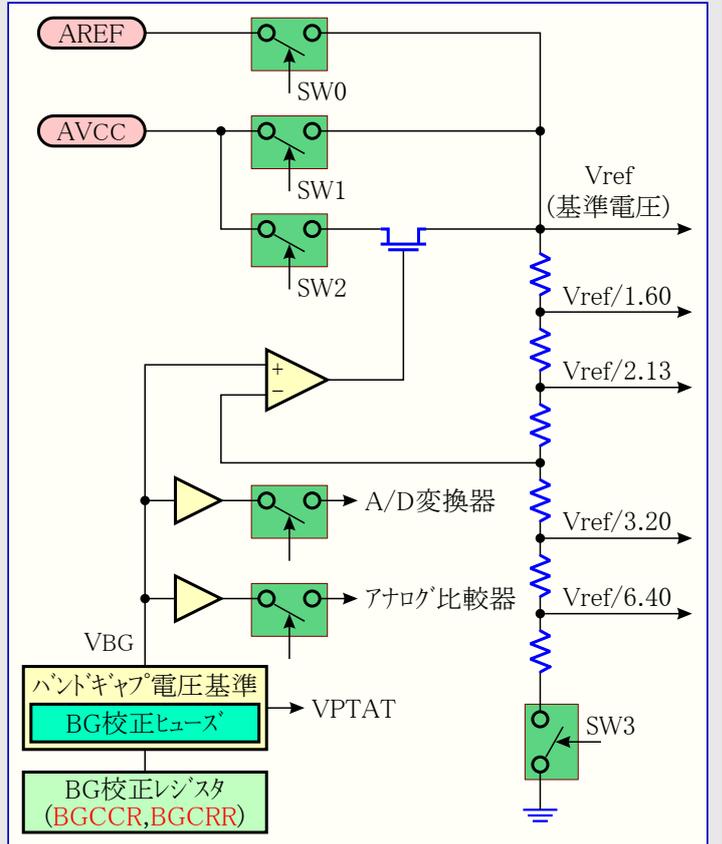
全てのアナログ部分(A/D変換器、D/A変換器、アナログ比較器)に関する基準電圧の選択はADMUXレジスタの**基準電圧選択(REFS1,0)**ビットを使って行われます。**129頁の「ADMUX - A/D変換多重器選択レジスタ」**をご覧ください。

バンドギャップと内部基準電圧を使う条件については、**34頁の「バンドギャップと内部基準電圧許可信号と始動時間」**をご覧ください。

AT90PWM81/161はダイ温度を監視するためにチップ上の温度感知器を持ちます。基準電圧回路で絶対温度に比例した電圧(VPTAT: Voltage Proportional-To-Absolute-Temperature)が生成され、緩衝した後でA/D変換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うことができます。華氏温度での絶対温度を得るため、測定したVtemp電圧は識票列内に格納されたVtemp工場校正値で尺度補正されなければなりません。詳細については**114頁の「温度測定」**項をご覧ください。

VBGとVtempはADMUXで正しいADCチャネルを選ぶことによって統合したA/D変換器で測定することができます(**129頁の「ADMUX - A/D変換多重器選択レジスタ」**をご覧ください)。

図15-1. 基準電圧回路



注: SW0~3を制御するのにREFS1,0が使われます。

### 15.3. 電圧基準電圧と温度感知器用レジスタ

#### 15.3.1. BGCCR - 電圧基準校正レジスタ (Bandgap Calibration C Register)

ビット (\$81)	7	6	5	4	3	2	1	0	BGCCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	1	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予約されています。

- ビット3~0 - BGCC3~0 : バンドギャップ基準電圧校正値 (BG Calibration of PTAT Current)

これらのビットはバンドギャップ基準電圧の公称値調節に使われます。これらの値は2進値で、故にVBGの最小値はBGCC3~0が0000の時に達し、BGCC3~0が1111の時に最大値です。1段階は概ね5mVです。

BGCCビットの更新は低電圧検出(BOD)基準に影響を及ぼします。BODは新しい検出基準に素早く反応します。

### 15.3.2. BGCRR – 電圧基準温度係数校正レジスタ (Bandgap Calibration Resistor Register)

ビット (\$80)	7	6	5	4	3	2	1	0	BGCRR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	1	0	0	0	

• ビット7~4 – Res : 予約 (Reserved Bits)

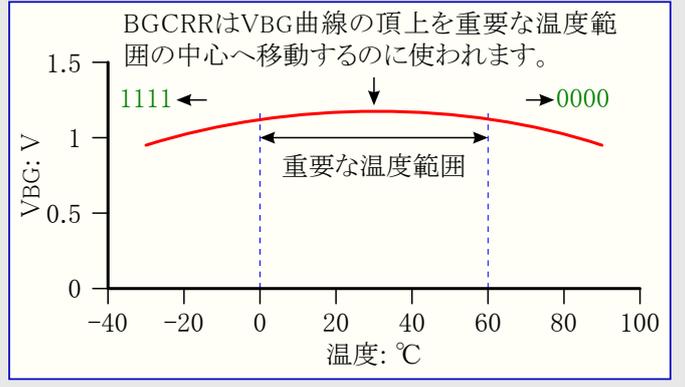
これらのビットは将来の使用に予約されています。

• ビット3~0 – BGCRR3~0 : 基準電圧温度係数校正値 (BG Calibration of Resistor ladder)

これらのビットはバンドギャップ基準電圧の温度係数補正に使われます。図15-2は温度の関数としてのVBGを図解します。VBG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、VBG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、BGCRR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。BGCRRビットは5つの可能な設定(0000,0001,0011,0111,1111)となる符号化された温度です。値0000はVBG曲線の頂上を最高可能温度へ移し、値1111はVREF曲線の頂上を最低可能温度へ移します。

図15-2. 温度の関数としてのVREF



### 15.4. 温度測定

温度測定は図15-3.で示されるように、シングル エント ADC12チャンネルと対にされたチップ上の温度感知器に基きます。

図15-3. 温度感知器回路



ADMUXレジスタでのA/Dチャンネル選択(MUX3~0)ビットの'1100'書き込みによるADC12チャンネル選択が温度感知を許可します(129頁の「AD MUX – A/D変換多重器選択レジスタ」をご覧ください)。温度感知器測定に関して推奨されるA/D変換基準源は内部2.56V基準電圧です。温度感知器が許可されると、A/D変換器は温度感知器上の電圧を測定するために単独変換動作で使うことができます。増幅器は最大A/D変換クロック速度でA/D変換器の採取/保持コンデンサの充電を許します。測定した電圧は表15-1.で記述されるように温度に対して直線的な関連を持ちます。基準電圧が2.56Vと等しい時に変換結果は温度に対して概ね1 LSB/°C(または2.5mV/°C)の相関を持ち、温度測定の代表的な精度は変位(オフセット)校正後で±10°Cです。

表15-1. 温度対感知器出力電圧 (代表値)

温度(°C)	-40°C	25°C	105°C	125°C
電圧(mV)	600	762	-	1012
A/D変換値	240	305	-	405

表15-1.に記載した値は代表値です。然しながら製法変化のため、温度感知器出力電圧は或るチップと別のチップで変化します。より正確な結果達成を可能とするために温度測定は応用ソフトウェアで校正することができます。

温度感知器を使うと、(華氏での)温度は次のように計算されます。

$T = A \times T_{ptat} + B$  ここで、

- A : 利得修正乗数 (定数1または符号なし固定小数点数)
- B : 変位(オフセット)修正項 (2の補数符号付きバイト)
- $T_{ptat}$  : 内部2.56V基準電圧で温度感知電圧測定時のA/D変換結果
- T : 華氏での温度 (°K=°C+273)

例:

A=\$80(=1.00)とB=8、そしてA/D変換結果が\$15E(350)なら、これは $T = 1.00 \times 350 + 8 = 358^\circ\text{K}(+85^\circ\text{C})$ の測定温度を与えます。

### 15.4.1. 製造時校正

識票列で利用可能な校正値を使うこともできます(146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)。

校正値は概ね+25°Cの室温での検査中に測定された値から決められます。校正測定は内部Vref(1.1V)形態でのA/D変換器でVCC=3Vで行われます。

摂氏での温度は次式を利用することで計算することができます。

$$T = ((([(ADCH \ll 8 | ADCL] - (273 + 25 - TSOFFSET)) \times TSGAIN] \div 128) + 25$$

ここで、

- ADCHとADCLはA/D変換データレジスタです。
- TSGAINは温度感知器利得(識票列のアドレス\$0007で前もって格納された1/128単位での符号なし固定小数点8ビット温度感知器係数)。146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。
- TSOFFSETは温度感知器変位(オフセット)修正項(識票列のアドレス\$0005で前もって格納された符号付き2の補数7ビット温度感知器変位(オフセット)読み出し)。

## 16. アナログ比較器

アナログ比較器はACMPn非反転ピンとACMPMまたはACMPMx反転ピンの入力値を比較します。

### 16.1. 特徴

- 3つのアナログ比較器
- 高速アナログ比較器
- $\pm 25\text{mV}$ 、 $\pm 10\text{mV}$ または0のヒステリシス
- 4つの基準レベル
- 構成設定可能な割り込みの生成

### 16.2. 概要

AT90PWM81/161は3つの高速アナログ比較が特徴です。

各比較器には非反転入力専用入力があり、各比較器の反転入力は次のように設定できます。

- A/D変換多重器選択(ADMUX)レジスタの基準電圧選択(REFS1,0)ビットで選んだVrefによって定義された4つの内部基準電圧内の固定値
- 内蔵D/A変換器で生成した値
- 外部ACMPMxアナログ入力

ACMPn非反転ピンの電圧がアナログ比較器n制御(ACnCON)レジスタの反転入力選択(ACnM2~0)ビットによって選んだ反転入力電圧よりも高い場合にアナログ比較器n出力(ACnO)が設定(1)されます。

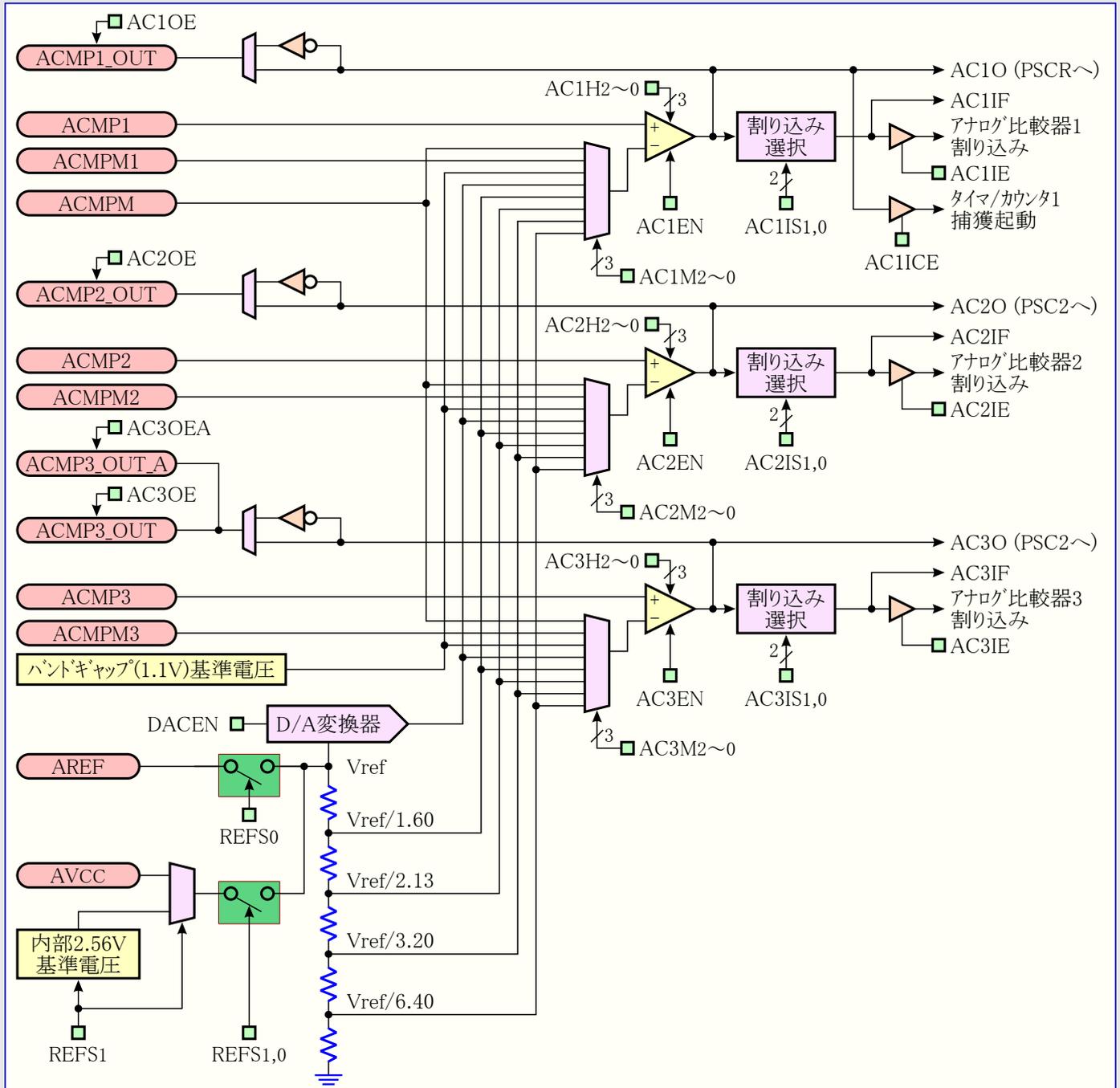
各比較器はアナログ比較器専用の独立した割り込みを起動できます。加えて、使用者は比較器出力の上昇端、下降端、切り替わり(両端)での割り込み起動を選べます。

この割り込み要求フラグはA/D変換器やD/A変換器の同期に使うこともできます。

更に比較器1の比較機出力はタイマ/カウンタ1の捕獲機能の起動に設定できます。

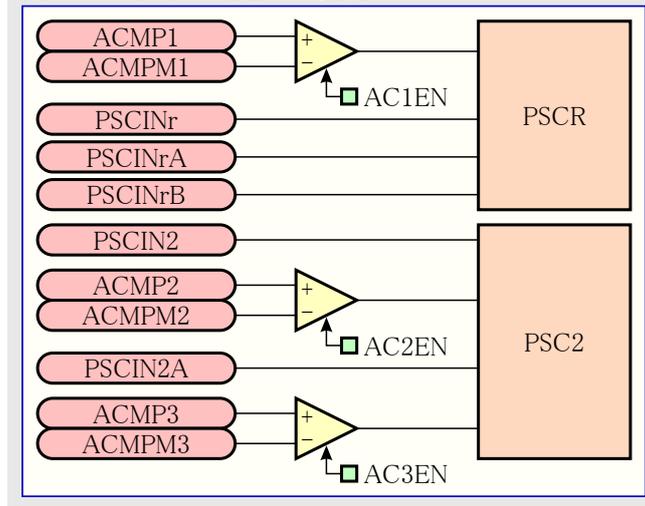
3つの比較器とそれらの周辺回路の構成図は図16-1.で示されます。

図16-1. アナログ比較器構成図



注: ・アナログ比較器ピン配置については2頁の「ピン配置」をご覧ください。  
 ・Vref電圧は129頁の「基準電圧選択」の表17-3.で定義されます。

図16-2. アナログ比較器PSC接続



### 16.3. アナログ比較器とA/D変換器間の共用ピン

多数のアナログ比較器入力ピンもA/D変換器入力として使うことができ、故に比較電圧を測定することが可能です。けれども、比較器入力がA/D変換器入力として選ばれると、A/D変換の採取段階中に尖頭雑音が発生します。これは比較器出力で望まない遷移を引き起こすかもしれません。従って入力の1つで電圧を測定する前に比較器出力を無効にすることが安全なソフトウェアの習慣です。

### 16.4. アナログ比較器レジスタ説明

各アナログ比較器にはそれ自身の制御レジスタがあります。

専用レジスタは3つのアナログ比較の出力とフラグを引き渡すために設計されています。

(訳注) 原書のAC0CON, AC1CON, AC2CONの個別記述は共通のACnCONとして纏めました。

#### 16.4.1. AC1CON – アナログ比較器1制御レジスタ (Analog Comparator 1 Control Register)

ビット (\$7D)	7	6	5	4	3	2	1	0	
	AC1EN	AC1IE	AC1IS1	AC1IS0	–	AC1M2	AC1M1	AC1M0	AC1CON
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### 16.4.2. AC2CON – アナログ比較器2制御レジスタ (Analog Comparator 2 Control Register)

ビット (\$7E)	7	6	5	4	3	2	1	0	
	AC2EN	AC2IE	AC2IS1	AC2IS0	–	AC2M2	AC2M1	AC2M0	AC2CON
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### 16.4.3. AC3CON – アナログ比較器3制御レジスタ (Analog Comparator 3 Control Register)

ビット (\$7F)	7	6	5	4	3	2	1	0	
	AC3EN	AC3IE	AC3IS1	AC3IS0	AC3OEA	AC3M2	AC3M1	AC3M0	AC3CON
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 – ACnEN : アナログ比較器n 動作許可 (Analog Comparator n Enable)

アナログ比較器nを許可するには、このビットを設定(1)してください。アナログ比較器nを禁止するには、このビットを解除(0)してください。

• ビット6 – ACnIE : アナログ比較器n 割り込み許可 (Analog Comparator n Interrupt Enable)

アナログ比較器n割り込みを許可するには、このビットを設定(1)してください。アナログ比較器n割り込みを禁止するには、このビットを解除(0)してください。

•ビット5,4 – ACnIS1,0 : アナログ比較器n 割り込み条件選択 (Analog Comparator n Interrupt Select bits)

これら2ビットは割り込み起動事象を決めます。各種設定は表16-1.で示されます。

表16-1. アナログ比較器割り込み条件選択 (注: n=0,1,2)

ACnIS1	ACnIS0	割り込み発生条件
0	0	比較器n出力の変移(トグル)
0	1	(予約)
1	0	比較器n出力の下降端
1	1	比較器n出力の上昇端

•ビット3 – AC3OEA : アナログ比較器3代替出力許可 (Analog Comparator 3 Alternate Output Enable) (アナログ比較器3のみ)

アナログ比較器3の代替出力ピンを許可するにはこのビットを設定(1)してください。アナログ比較器3の代替出力ピンを禁止するにはこのビットを解除(0)してください。

•ビット2~0 – ACnM2~0 : アナログ比較器n 反転入力選択 (Analog Comparator n Multiplexer register)

これら3ビットはアナログ比較器nの反転入力の入力を決めます。各種設定は表16-2.で示されます。

表16-2. アナログ比較器n 反転入力選択 (注: n=0,1,2)

ACnM2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
反転入力	Vref/6.40	Vref/3.20	Vref/2.13	Vref/1.60	バンドギャップ	D/A変換器	ACMPMnピン	ACMPMピン

注: Vref電圧は129頁の「基準電圧選択」の表17-3.で定義されます。

#### 16.4.4. ACnECON – アナログ比較器n拡張制御レジスタ (Analog Comparator n Extended Control Register)

ビット	7	6	5	4	3	2	1	0	
(\$7A,\$7B,\$7C)	-	-	ACnOI	ACnOE	AC1ICE	ACnH2	ACnH1	ACnH0	ACnECON
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います。

•ビット5 – ACnOI : アナログ比較器n 出力反転 (Analog Comparator n Output Invert)

アナログ比較器nの出力を反転するには、このビットを設定(1)してください。アナログ比較器nの出力を維持するには、このビットを解除(0)してください。

•ビット4 – ACnOE : アナログ比較器n 出力許可 (Analog Comparator n Output Enable)

アナログ比較器nの出力ピンを許可するには、このビットを設定(1)してください。アナログ比較器nの出力ピンを禁止するには、このビットを解除(0)してください。

•ビット3 – AC1ICE : アナログ比較器1捕獲起動許可 (Analog Comparator 1 Input Capture Enable)

アナログ比較器1の事象でタイマ/カウンタ1の捕獲機能を許可するにはこのビットを設定(1)してください。この場合、比較器1出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音除去機能と端(エッジ)選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。比較器がタイマ/カウンタ1捕獲割り込みを起動するには、タイマ/カウンタ1割り込み許可レジスタ(TIMSK1)の捕獲割り込み許可(ICIE1)ビットが設定(1)されなければなりません。

(59頁の「TCCR1B – タイマ/カウンタ1制御レジスタ」の)ICES1ビットが1設定の場合はAC1Oの上昇端がタイマ/カウンタ1の捕獲起動事象、ICES1が0設定の場合はそれが下降端、でのどちらかが考慮され(扱われ)ます。

この機能を禁止するにはこのビットを解除(0)してください。この場合、アナログ比較器1と捕獲機能間の接続は存在しません。

•ビット2~0 – ACnH2~0 : アナログ比較器n ヒステリシス選択 (Analog Comparator n Hysteresis register)

これら3ビットはアナログ比較器nのヒステリシス値を決めます。各種設定は表16-7.で示されます。

表16-7. アナログ比較器n ヒステリシス選択

ACnH2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
説明	なし	+10mV	-10mV	±10mV	(予約)	+25mV	-25mV	±25mV

### 16.4.5. ACSR – アナログ比較器状態レジスタ (Analog Comparator Status Register)

ビット	7	6	5	4	3	2	1	0	
\$00 (\$20)	AC3IF	AC2IF	AC1IF	–	AC3O	AC2O	AC1O	–	ACSR
Read/Write	R/W	R/W	R/W	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	

• **ビット7 – AC3IF : アナログ比較器3 割り込み要求フラグ (Analog Comparator 3 Interrupt Flag)**

このビットはアナログ比較器3出力が**アナログ比較器3制御レジスタ(AC3CON)**の**割り込み条件選択(AC3IS1,0)**ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC3CONレジスタの**割り込み許可(AC3IE)**ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期に使うことができます。

• **ビット6 – AC2IF : アナログ比較器2 割り込み要求フラグ (Analog Comparator 2 Interrupt Flag)**

このビットはアナログ比較器2出力が**アナログ比較器2制御レジスタ(AC2CON)**の**割り込み条件選択(AC2IS1,0)**ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC2CONレジスタの**割り込み許可(AC2IE)**ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期に使うことができます。

• **ビット5 – AC1IF : アナログ比較器1 割り込み要求フラグ (Analog Comparator 1 Interrupt Flag)**

このビットはアナログ比較器1出力が**アナログ比較器1制御レジスタ(AC1CON)**の**割り込み条件選択(AC1IS1,0)**ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC1CONレジスタの**割り込み許可(AC1IE)**ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期に使うことができます。

• **ビット4 – Res : 予約 (Reserved Bits)**

このビットは予約されています。

• **ビット3 – AC3O : アナログ比較器3 出力 (Analog Comparator 3 Output)**

AC4Oビットはアナログ比較器4の直接的な出力です。

比較器出力が**1**の時に設定(1)されます。

比較器出力が**0**の時に解除(0)されます。

• **ビット2 – AC2O : アナログ比較器2 出力 (Analog Comparator 2 Output)**

AC2Oビットはアナログ比較器2の直接的な出力です。

比較器出力が**1**の時に設定(1)されます。

比較器出力が**0**の時に解除(0)されます。

• **ビット1 – AC1O : アナログ比較器1 出力 (Analog Comparator 1 Output)**

AC1Oビットはアナログ比較器1の直接的な出力です。

比較器出力が**1**の時に設定(1)されます。

比較器出力が**0**の時に解除(0)されます。

• **ビット0 – Res : 予約 (Reserved Bits)**

このビットは予約されています。

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

ビット	7	6	5	4	3	2	1	0	
(\$77)	ADC8D ACMP3D	ADC7D AMP0-D	ADC5D ACMP2D	ADC4D ACMP3MD	ADC3D ACMPMD	ADC2D ACMP2MD	ADC1D	ADC0D ACMP1D	DIDR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7,5,0 – ACMPnD : ACMPn デジタル入力禁止 (ACMPn Digital Input Disable)
- ビット4,2 – ACMPnMD : ACMPnM デジタル入力禁止 (ACMPnM Digital Input Disable)
- ビット3 – ACMPMD : ACMPM デジタル入力禁止 (ACMPM Digital Input Disable)

このビットが論理1を書かれると、対応するアナログ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)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。これらのピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされないとき、このビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

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

ビット	7	6	5	4	3	2	1	0	
(\$78)	–	–	–	–	ACMP1MD	AMP0+D	ADC10D	ADC9D	DIDR1
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 – ACMP1MD : ACMP1M デジタル入力禁止 (ACMP1M Digital Input Disable)

このビットが論理1を書かれると、対応するアナログ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)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。これらのピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされないとき、このビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

## 17. A/D変換器

### 17.1. 特徴

- 10ビット分解能
- 積分非直線性誤差0.5 LSB
- 絶対精度±2 LSB
- 変換時間8~250μs
- 最大分解能で120kSPS(採取/s)まで
- 8チャンネルのシングルエンド入力多重器内蔵
- 5%精度で5,10,20,40倍に設定可能な1つの差動入力チャネル
- A/D変換結果読み出しに対する任意の左揃え
- 0~VCC A/D変換入力電圧範囲
- 選択可能な2.56V A/D変換基準電圧
- 連続と単独の変換動作
- 割り込み元の自動起動によるA/D変換開始
- A/D変換完了割り込み
- 休止形態雑音低減機能
- 温度感知器

AT90PWM81/161は10ビット逐次比較A/D変換器が特徴です。このA/D変換器は8のシングルエンド電圧入力を許す11チャンネルアナログ多重器に接続されます。このシングルエンド電圧入力は0V(GND)が基準です。

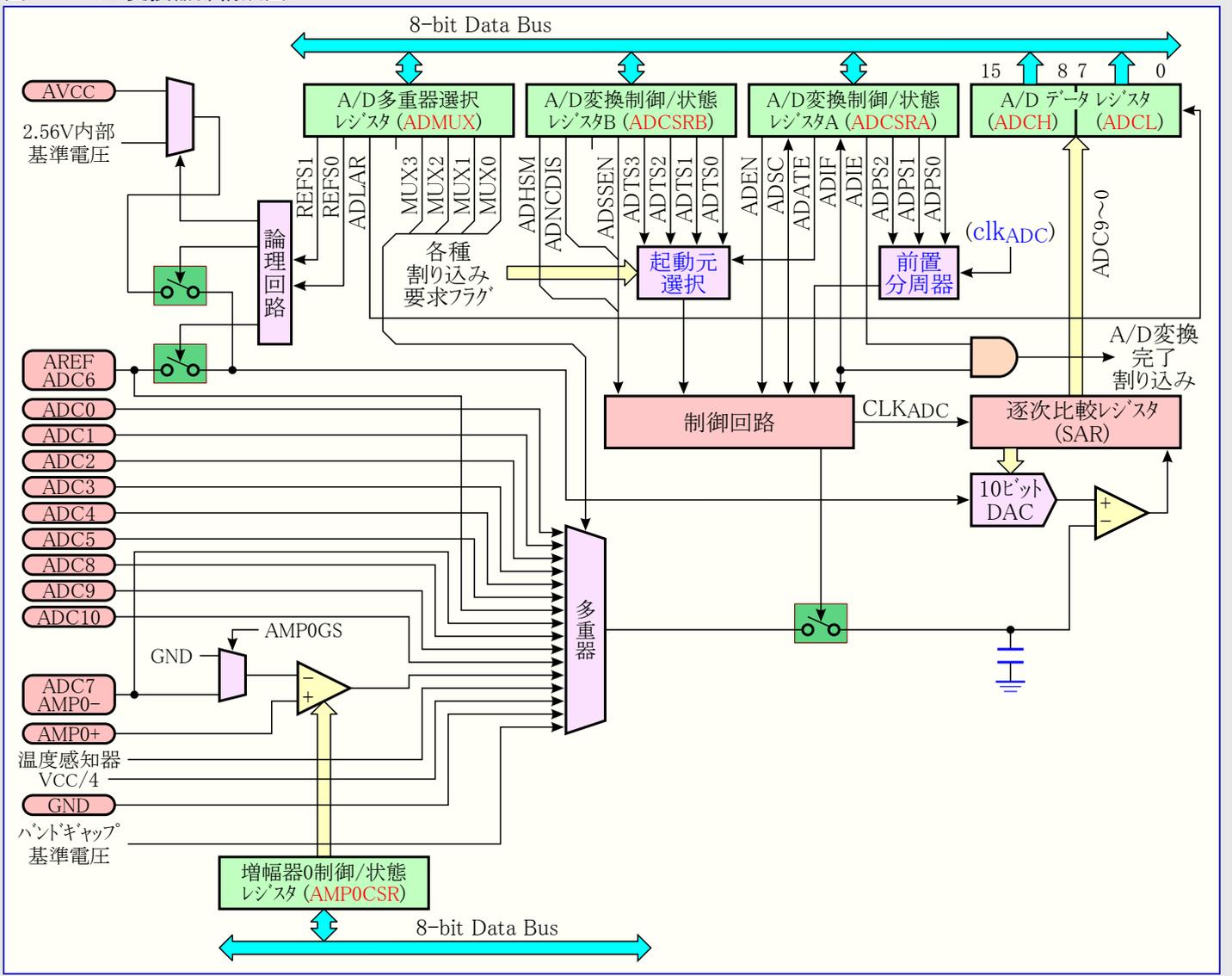
デバイスはA/D変換前に差動入力電圧で14dB(×5)、20dB(×10)、26dB(×20)、32dB(×40)の増幅段を提供する設定変更可能な利得段が装備された2つの差動電圧入力の組み合わせも支援します。増幅したチャネルでは8ビット分解能が期待できます。

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

A/D変換部には分離されたアナログ電源供給ピン(AVCC)があります。AVCCはVCCから±0.3Vよりも多く違ってはなりません。このピンの接続方法は127頁の「雑音低減技術」項をご覧ください。

公称2.56Vの内蔵基準電圧またはAVCCがチップ上で提供されます。この基準電圧は雑音特性向上のため、コンデンサによってAREFピンで外部的にデカップ(雑音分離)できます。

図17-1. A/D変換器部構成図



## 17.2. 操作

A/D変換部は逐次比較を通してアナログ入力電圧を10ビットのデジタル値に変換します。最小値はGNDを表し、最大値はAREFピンの電圧を表します。A/D多重器選択レジスタ(ADMUX)の基準電圧選択(REFS1,0)ビットへの書き込みにより、任意でAVCCまたは内部2.56V基準電圧がAREFピンに接続できます。従ってこの内部基準電圧は雑音耐性を改善するためにAREFピンで外部コンデンサによってデカップ(雑音結合減少)ができます。

アナログ入力チャネルはADMUXのチャネル選択(MUX3~0)ビットへの書き込みによって選ばれます。GNDとハントギャップ固定基準電圧だけでなく、どのADC入力ピン(ADC10~0)もがA/D変換器のシングルエンド入力として選べます。

A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットの設定(1)によって(動作が)許可されます。基準電圧はA/D変換部が許可されるかされないかがどうでも、ADMUXレジスタのREFS1とREFS0ビットによって設定されます。ADENが解除(0)されているとA/D変換部は電力を消費しないので、節電をする休止形態へ移行する前にA/D変換部をOFFに切り替えることが推奨されます。

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

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

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

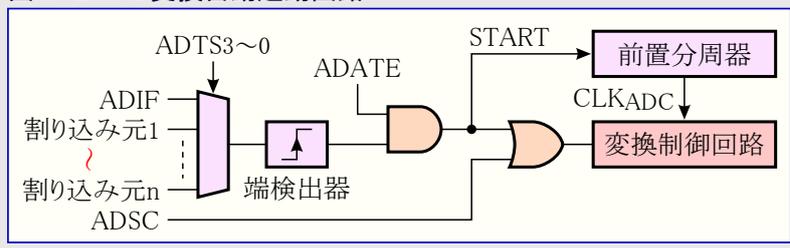
## 17.3. 変換の開始

単独変換はADCSRAで変換開始(ADSC)ビットに論理1を書くことによって開始されます。このビットは変換が進行中である限り、1に留まり、変換が完了されるとハードウェアによって解除(0)されます。変換が進行中に違う入力チャネルが選ばれると、A/D変換部はそのチャネル変更を実行する前に現在の変換を済ませます。

代わりに、変換は様々な起動元によって自動的に起動できます。自動起動はA/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットの設定(1)によって許可されます。起動元はA/D変換制御/状態レジスタB(ADCSRB)のA/D変換起動元選択(ADTS3~0)ビットの設定によって選ばれます(起動元の一覧についてはADTSビットの記述をご覧ください)。選んだ起動信号上に上昇端が起きると、A/D変換用前置分周器がリセットし、変換が開始されます。これは一定間隔での変換開始の方法を提供します。変換完了時、起動信号が未だ設定(1)されている場合、新規の変換は開始されません。変換中にこの起動信号上で別の上昇端が起きると、その端(エッジ)は無視されます。指定した割り込みが禁止またはステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)でも、割り込み要求フラグが設定(1)されることに注意してください。従って割り込みを起こさずに変換が起動できます。けれども次の割り込み要因で新規変換を起動するために、割り込み要求フラグは解除(0)されなければなりません。

電力段階制御器(PSC)の同期信号からの起動はこれらと異なり、フラグがありません。この場合、新規変換は各々の起動信号で開始されます。けれども、ADCSRBのPSC同期信号での単発動作許可(ADSSEN)ビットの設定(1)によって単発動作形態を活性(有効)にすることができます。この場合の同期信号はADCHレジスタが読まれるまで防がれます。

図17-2. A/D変換自動起動回路



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

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

変換はA/D変換雑音低減機能の使用によっても開始され得ます。この機能はアイドル休止動作とA/D変換雑音低減休止動作中に変換を許可します。詳細については126頁の「雑音低減機能」をご覧ください。(訳注:共通性から2行追加)

## 17.4. 前置分周と変換タイミング

既定での逐次比較回路は最大分解能を得るのに50kHz～2MHzの入力クロック周波数を必要とします。10ビットよりも低い分解能が必要とされるなら、A/D変換器への入力クロック周波数はより高い採取速度を得るために2MHzよりも高くできます。

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

ADCSRAのA/D変換開始(ADSC)ビットの設定(1)によってシングルエンド入力の変換を起動すると、その変換は直後の変換クロックの上昇端で始まります。差分入力変換タイミングの詳細については125頁の「チャンネル変更と基準電圧選択」をご覧ください。

通常変換は15.5変換クロック周期で行われます。A/D変換部がONされる(ADCSRAのADEN=1)後の最初の変換はアナログ回路を初期化するために25変換クロック周期で行われます。

実際の採取&保持(保持開始点)は通常変換の開始後3.5変換クロック周期、初回変換の開始後13.5変換クロック周期で行われます。変換が完了すると、結果がA/Dデータレジスタ(ADCH,ADCL)に書かれ、ADCSRAのA/D変換完了割り込み要求フラグ(ADIF)が設定(1)されます。単独変換動作(ADATE=0)では同時にADCSRAのA/D変換開始(ADSC)ビットが解除(0)されます。その後にソフトウェアは再びADSCを設定(1)でき、新規変換は変換クロックの最初の上昇端で開始されます。

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

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

図17-3. A/D変換前置分周器部構成

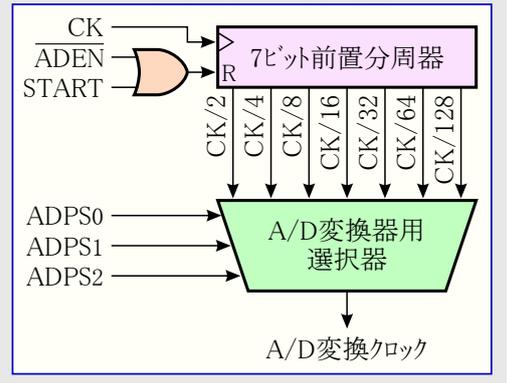


図17-4. 初回変換タイミング (単独変換動作)

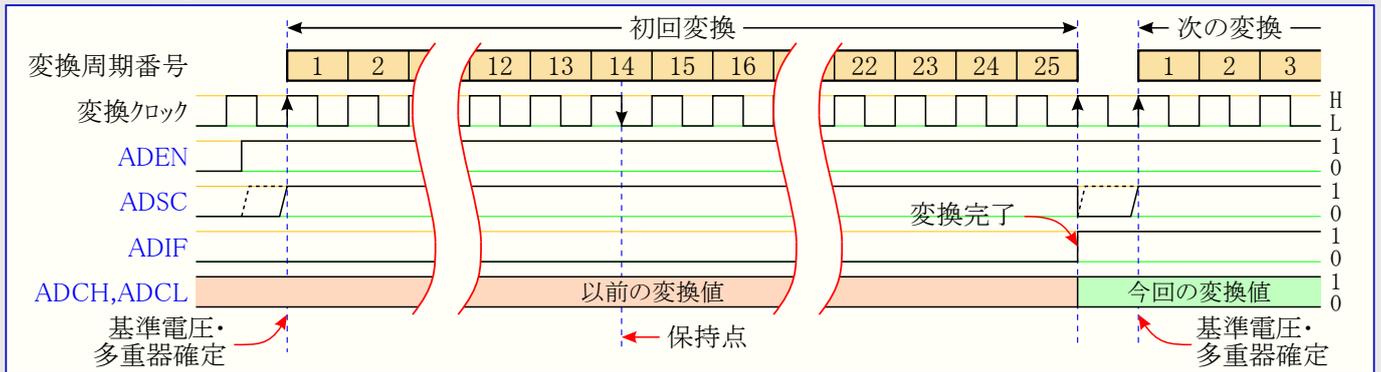
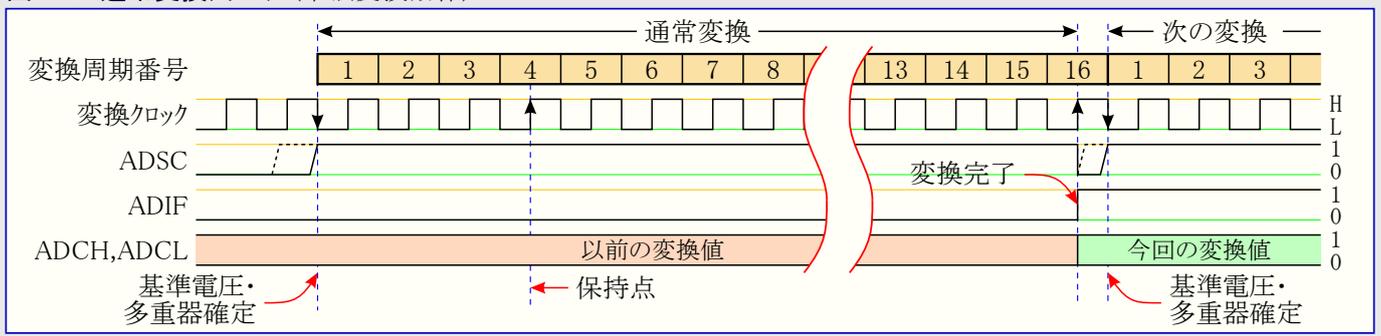


図17-5. 通常変換タイミング (単独変換動作)



(訳注) 図17-4.と図17-5.は本文に合わせて補正しています。

図17-6. 通常変換タイミング\* (自動起動変換動作)

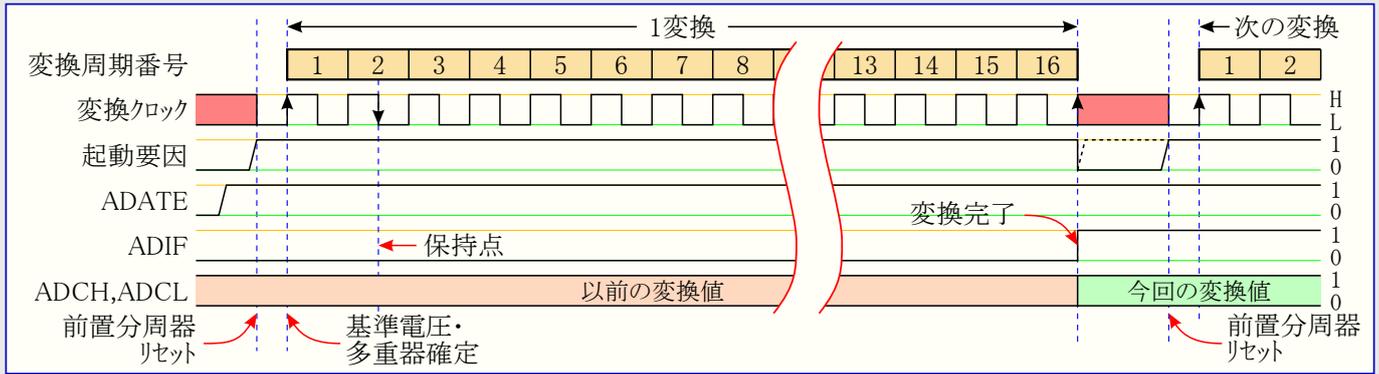


図17-7. 連続変換動作タイミング

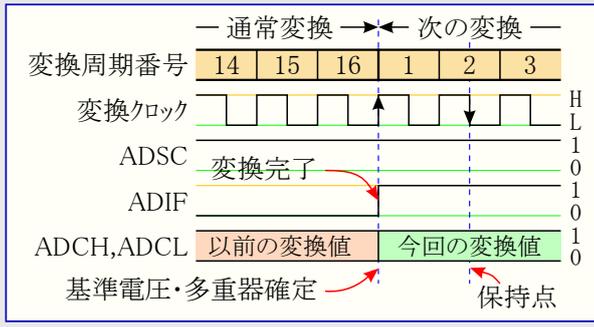


表17-1. A/D変換時間

変換種別	保持点	変換時間
初回変換	13.5	25
シングルエンド入力通常変換	3.5	15.5
自動起動変換	4	16

## 17.5. チャネル変更と基準電圧選択

A/D多重器選択レジスタ(ADMUX)のチャネル選択(MUX3~0)ビットと基準電圧選択(REFS1,0)ビットはCPUが乱順にアクセスするための一時レジスタを通して単独緩衝されます。これはチャネルと基準電圧の選択が変換中の安全なところでだけ行うのを保証します。チャネルと基準電圧の選択は変換が開始されるまで継続的に更新されます。一旦変換が始まると、A/D変換器に対して十分な採取/変換時間を保証するためにチャネルと基準電圧の選択は固定されます。継続的な更新は変換完了(ADCSRAのADIF=1)前の最後の変換クロック周期で再開します。ADCSRAの変換開始(ADSC)ビットが書かれた後の次の変換クロックの上昇端で変換が始まることに注意してください。従って使用者はADSC書き込み後、1変換クロック周期(経過)まで新しいチャネルまたは基準電圧選択値をADMUXに書かないことが推奨されます。

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

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

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

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

### 17.5.1. A/D入力チャネル

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

- 単独変換動作では常に変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法はチャネル選択を変更する前に変換が完了するまで待つことです。
- 連続変換動作では常に最初の変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法は最初の変換が完了するまで待ち、その後にチャネル選択を変更することです。既に次の変換が自動的に開始されているので、次の結果は直前のチャネル選択を反映します。それに続く変換は新しいチャネル選択を反映します。
- 連続変換動作では増幅器が増幅チャネル変換の最後でADSCビットを解除(0)するので、各変換の最後でADSCビットがソフトウェアによって再び設定(1)される場合を除き、連続変換動作の使用は不可能です。

## 17.5.2. A/D変換基準電圧

このA/D変換用の基準電圧(VREF)はA/D変換に対する変換範囲を示します。VREFを越えるシングル エント 入力チャネルは\$3FFで打ち切る符号に帰着します。VREFはAVCC、内部2.56V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してA/D変換部に接続されます。内部2.56V基準電圧は内蔵基準(バンドギャップ)電圧(VBG)から内部増幅器を通して生成されます。外部AREFピンがA/D変換部に接続されている場合、AREFピンとGND間にコンデンサを接続することにより、基準電圧は雑音耐性をより高められます。VREF(電圧)は高入力インピーダンス電圧計とAREFピンで測定することもできます。VREFは高インピーダンス出力で、容量性負荷のみがシステム内で接続されるべきであることに注意してください。

使用者は基準電圧選択としてAVCC、AREF、内部2.56V基準電圧間の切り替えができます。基準電圧源切り替え後の最初のA/D変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

## 17.6. 雑音低減機能

このA/D変換部はCPUコアと他の周辺I/Oが誘導した雑音を削減するために休止形態中の変換を可能にする雑音低減機能が特徴です。この機能はA/D変換雑音低減動作とアイドル動作で使えます。この機能を使うには次の手順が使われるべきです。

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

別の可能な手順は自動起動変換で可能です。

1. A/D変換雑音消去器禁止(ADNCDIS)ビットが設定(1)されていることを確認してください。
2. A/D変換自動起動許可(ADATE)ビットが設定(1)されていることを確認してください。
3. A/D変換雑音低減(またはアイドル)動作に移行してください。A/D変換部は次の起動事象で変換を始めます。
4. A/D変換完了前に他の割り込みが起こらなければ、A/D変換完了割り込みはCPUを起動してA/D変換完了割り込みルーチンを実行します。A/D変換完了前に他の割り込みがCPUを起動すると、その割り込みが実行され、A/D変換完了割り込み要求はA/D変換完了時に生成されます。CPUは新規SLEEP命令が実行されるまで活動動作に留まります。

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

このような休止形態でA/D変換が許可され、使用者が差動変換の実行を望む場合、有効な結果を得るのに延長した(初回)変換を指示するため、使用者は休止形態から起動後にA/D変換部をOFF→ON(ADEN=0→1)に切り替えることが推奨されます。

### 17.6.1. アナログ入力回路

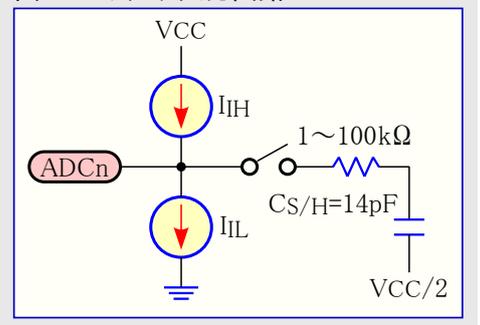
シングル エント 入力チャネルのアナログ回路は図17-8.で図示されます。ADCnに印加したアナログ(信号)源はそのチャネルがADC入力として選ばれているかどうかにかかわらず、ピン容量とそのピンの漏れ電流に左右されます。そのチャネルが選ばれると、(アナログ)信号源は直列抵抗(入力経路の合成抵抗)を通してS/Hコンデンサを駆動しなければなりません。

A/D変換部は概ね5kΩ若しくはそれ以下の出力インピーダンスのアナログ信号用に最適化されています。このような(アナログ)信号源が使われるなら、採取時間は無視してもよいでしょう。より高いインピーダンスの(アナログ)信号源が使われる場合、採取時間は広範囲に変化し得るS/Hコンデンサを充電するために(アナログ)信号源がどれくらいの時間を必要とするかに依存します。必要とされるS/Hコンデンサへの充放電を最小とするため、使用者は緩やかに変化する低インピーダンス(アナログ)信号源だけを使うことが推奨されます。

差動増幅チャネルが使われる場合、この入力回路は多少違って見えるので、数100kΩまたはそれ以下の供給元インピーダンスが推奨されます。

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

図17-8. アナログ入力回路

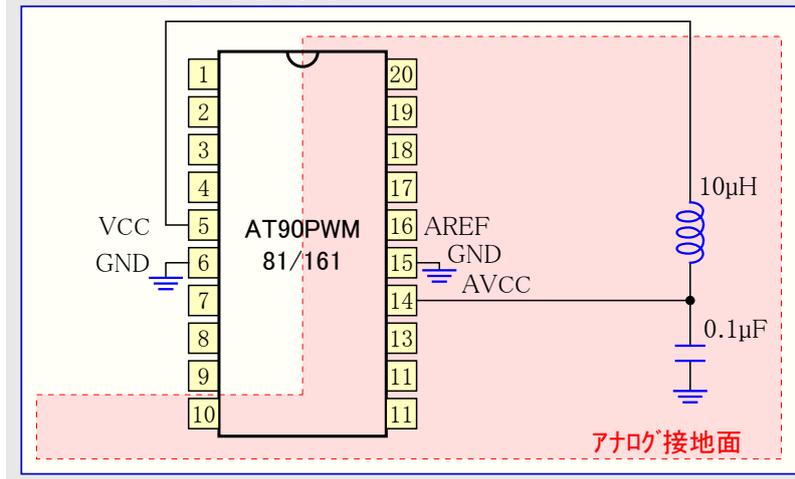


### 17.6.2. アナログ雑音低減技術

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

1. アナログ信号経路を可能な限り最短にしてください。アナログ信号線がアナログGND面上を走ることに注意し、高速切り替えデジタル信号線から充分離すことを守ってください。
2. デバイスのAVCCピンは図17-9.で示されるようにLC濾波器を経由してデジタル供給電圧(VCC)に接続されるべきです。
3. CPUからの誘導雑音を低減するために、A/D変換の雑音低減機能を使ってください。
4. どれかのADCポートピンがデジタル出力として使われる場合、これらが変換進行中に切り替わらないことが重要です。

図17-9. A/D変換部電源接続



### 17.6.3. 変位(オフセット)補償の仕組み

利得段には差動測定の変位(オフセット)をできるだけ無効にする組み込み変位補正回路があります。アナログ経路内の残留変位は両入力未接続でAMP0ISEビットを使う両差動入力短絡によって直接的に計測できます(134頁の「AMP0CSR - 増幅器0制御/状態レジスタ」をご覧ください)。その後この残留変位は計測結果からソフトウェアで減算できます。この変位補正に基いたソフトウェア手法の使用はどのチャンネルの変位も1 LSB以下に減少できます。

### 17.6.4. A/D変換の精度定義

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

• **変位(オフセット)誤差** - 図17-10.

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

• **利得誤差** - 図17-11.

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

• **積分非直線性誤差 (INL)** - 図17-12.

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

• **微分非直線性誤差 (DNL)** - 図17-13.

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

• **量子化誤差**

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

• **絶対精度**

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

図17-10. 変位(オフセット)誤差

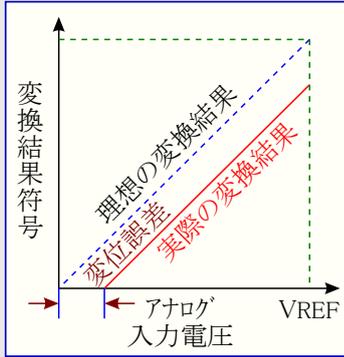


図17-11. 利得誤差

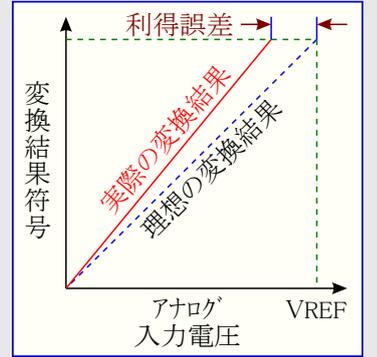


図17-12. 積分非直線性誤差

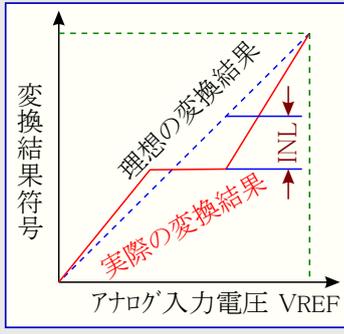
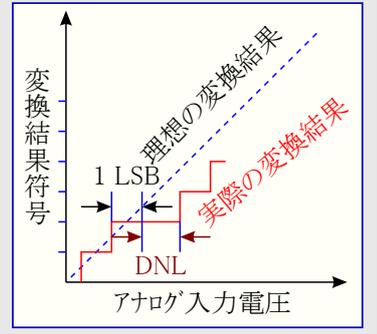


図17-13. 微分非直線性誤差



### 17.7. A/D変換の結果

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

シングルエンド入力変換の結果は右式で示されます。VINは選んだ入力ピンの電圧、VREFは選んだ基準電圧です(129頁の表17-3と表17-4をご覧ください)。\$000はアナログGNDを表し、\$3FFは選んだ基準電圧を表します。

$$ADC = \frac{VIN \times 1023}{VREF}$$

差動チャンネルが使われると、その結果は右式で示されます。VPOSは非反転入力ピンの電圧、VNEGは反転入力ピンの電圧、GAINは選んだ倍率、VREFは選んだ基準電圧です。この結果は-512(\$200)~+511(\$1FF)の2の補数形式で示されます。使用者が結果の符号を素早く検査したいなら、結果のMSB(ADCHのADC9)を読むことで充分なことに注目してください。このビットが1ならばその結果は-、0ならばその結果は+です。図17-14.は差動入力範囲の符号化を示します。

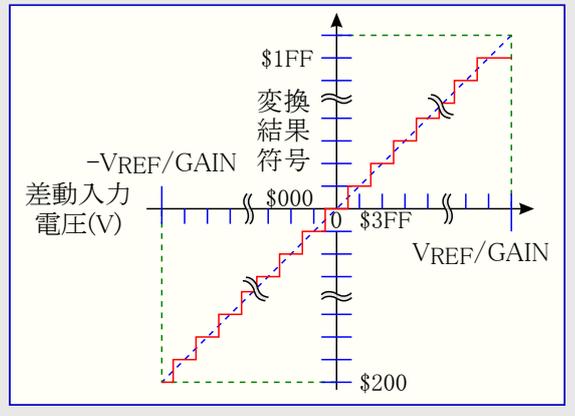
$$ADC = \frac{(VPOS - VNEG) \times GAIN \times 512}{VREF}$$

表17-2.は差動入力チャンネルの組(ADCn-ADCm)が基準電圧(VREF)で選ばれた場合の出力符号の結果を示します。

表17-2. 入力電圧と出力符号の関係

VADCn	読み出し符号	対応する10進値
VADCm + VREF/GAIN	\$1FF	511
VADCm + (511/512)VREF/GAIN	\$1FF	511
VADCm + (510/512)VREF/GAIN	\$1FE	510
}	}	}
VADCm + (1/512)VREF/GAIN	\$001	1
VADCm	\$000	0
VADCm - (1/512)VREF/GAIN	\$3FF	-1
}	}	}
VADCm - (511/512)VREF/GAIN	\$201	-511
VADCm - VREF/GAIN	\$200	-512

図17-14. 差動入力計測範囲



例1: ADMUX=\$EB(AMP0+とAMP0-の差動10倍入力、VREF=2.56V、左揃え)で、AMP0+が300mV、AMP0-が500mVの場合、A/D変換値=  $512 \times 10 \times (300 - 500) \div 2560 = -400 = \$270$ 、従ってADCH=\$9C、ADCL=\$00、ADLAR=0ならADCH:L=\$0270

(訳注) 原書本位置の例2は本デバイスで設定不可能な例のため削除しました。

## 17.8. A/D変換用レジスタ説明

AT90PWM81/161のA/D変換器は3つの異なるレジスタを通して制御されます。ADCSRAとADCSRBのレジスタはA/D変換器制御/状態レジスタで、ADMUXは変換すべきチャネルと基準電圧源(Vref)の選択を許します。

変換結果は各々上位側ビットと下位側ビットを含むADCHとADCLレジスタに格納されます。

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

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	REFS1	REFS0	ADLAR	–	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7,6 – REFS1,0 : 基準電圧選択 (ADC Vref Selection Bits)

これら2ビットはA/D変換器用の基準電圧を決めます。各種設定は表17-3.で示されます。

表17-3. A/D変換部の基準電圧選択

REFS1	REFS0	基準電圧	PE3/AREFピン
0	0	外部基準電圧(Vref)	外部基準電圧
0	1	AVCC	–
1	0	内部2.56V基準電圧	内部基準電圧のテックアップ(雑音分離)用外部コンデンサ接続
1	1	内部2.56V基準電圧	PE3ピンは入出力ピンとして開放

これらのビットが変換中に変更されると、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。

内部Vref(AVCCまたは内部2.56V基準電圧)が選ばれる場合、これを必要とするアナログ機能が(活動に)設定されると直ぐにONへ切り替わります。

#### • ビット5 – ADLAR : A/D変換 左揃え選択 (ADC Left Adjust Result)

A/D変換結果を左揃えにするにはこのビットを設定(1)してください。

A/D変換結果を右揃えにするにはこのビットを解除(0)してください。

ADLARビットはA/Dデータレジスタ内の変換結果の配置に影響を及ぼします。ADLARビットの変更はどんな進行中の変換にも拘らず、直ちにA/Dデータレジスタの内容に影響を及ぼします。このビットの完全な記述については131頁の「A/Dデータレジスタ」をご覧ください。

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

これら4ビットはどのアナログ入力A/D変換器入力に接続されるかを決めます。各種設定は表17-4.をご覧ください。

表17-4. アナログ入力チャネル選択

MUX3~0	A/D変換器入力	MUX3~0	A/D変換器入力
0 0 0 0	ADC0 (PD2)	1 0 0 0	ADC8 (PB6)
0 0 0 1	ADC1 (PD3)	1 0 0 1	ADC9 (PB7)
0 0 1 0	ADC2 (PB3)	1 0 1 0	ADC10 (PD7)
0 0 1 1	ADC3 (PB4)	1 0 1 1	AMP0 (–はPD5/AMP0–,+はPD6/AMP0+)
0 1 0 0	ADC4 (PD4)	1 1 0 0	温度感知器 (Vtemp)
0 1 0 1	ADC5 (PB5)	1 1 0 1	VCC/4
0 1 1 0	ADC6 (AREF)	1 1 1 0	バンドキャップ電圧(VBG)
0 1 1 1	ADC7 (PD5)	1 1 1 1	0V(GND)

これらのビットが変換中に変更される場合、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。

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

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

#### • ビット7 – ADEN : A/D許可 (ADC Enable)

A/D変換部(動作)を許可するにはこのビットを設定(1)してください。

A/D変換部(動作)を禁止するにはこのビットを解除(0)してください。

変換走行中のこのビットの解除(0)は変換の最後で実施します。

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

単独変換動作での変換の開始、または連続変換動作での最初の変換を始めるには、このビットを設定(1)してください。変換完了時、ハードウェアによって解除(0)されます。このビットへの0書き込みは無効です。初回変換はA/D変換部の初期化を実行します。

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

A/D変換器の自動起動を許可するにはこのビットを設定(1)してください。単独変換動作へ戻すにはこのビットを解除(0)してください。

自動起動動作での起動元はA/D変換制御/状態レジスタB(ADCSRB)のA/D変換起動要因選択(ADTS3~0)ビットで選ばれます。表17-6をご覧ください。

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

変換が完了すると直ぐにハードウェアによって設定(1)され、データレジスタ(ADCH,ADCL)が変換結果で更新されます。対応する割り込み処理ベクタを実行する時にハードウェアによって解除(0)されます。代わりにこのフラグに論理1を書くことによってもADIFが解除(0)できます。

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

A/D変換完了割り込みを活性化(許可)するにはこのビットを設定(1)してください。A/D変換完了割り込みを禁止するにはこのビットを解除(0)してください。

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

これら3ビットはシステムクロック周波数とA/D変換器の入力クロック間の分周係数を決めます。各種設定は表17-5.で示されます。

表17-5. A/D変換 前置分周器選択

ADPS2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
分周数	2	2	4	8	16	32	64	128

17.8.3. ADCSRB – A/D変換 制御/状態レジスタB (ADC Control and Status Register B)

ビット (\$7B)	7	6	5	4	3	2	1	0	
	ADHSM	ADNCDIS	-	ADSSEN	ADTS3	ADTS2	ADTS1	ADTS0	ADCSRB
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 – ADHSM : 高速動作 (ADC High Speed Mode)

このビットの1書き込みがA/D変換器高速動作を許可します。200kHzよりも高いA/D変換クロック周波数で変換をしたい場合、このビットを設定(1)してください。

•ビット6 – ADNCDIS : A/D変換雑音消去器禁止 (ADC Noise Canceller Disable)

アイドルまたはA/D変換雑音低減動作移行時に自動的なA/D変換の開始を禁止するには、このビットを設定(1)してください。アイドルまたはA/D変換雑音低減動作移行時に自動的なA/D変換の開始を許可するには、このビットを解除(0)してください。誤ったA/D変換再開を防ぐため、A/D変換部が動作中または自動起動の場合、アイドルまたはA/D変換雑音低減動作移行前にADNCDISが設定(1)されなければなりません。

•ビット5 – Res : 予約 (Reserved Bits)

このビットは予約されています。

•ビット4 – ADSSEN : PSC同期信号でのA/D変換単発許可 (ADC Single Shot Enable on PSC's synchronization signals)

PSC同期信号(PSCRASYとPSC2ASY)での自動起動時に単発動作を強化するには、このビットを設定(1)してください。この場合、単独変換が実行され、ADCHレジスタ読み込みまでPSCRASYとPSC2ASYは防がれます。PSCRASYとPSC2ASYの自動起動での連続的な変換を許可するには、このビットを解除(0)してください。

•ビット3~0 – ADTS3~0 : A/D変換自動起動要因選択 (ADC Auto Trigger Source)

これらのビットはA/D変換器が自動起動動作で動く場合にだけ必要です。これはA/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットが設定(1)される場合の意味です。

表17-6.に従ってこれら4ビットは変換開始起動を生成する割り込み事象を選びます。変換の開始は割り込みが許可であろうとなかろうと、選んだ割り込み要求フラグの上昇端によって生成されます。電力段制御器(PSC)nアナログ同期(PSCnASY)事象の場合、フラグがありません。従って、ADSSENがリセット(0)されているなら、この変換は直前の変換が完了され、この起動事象が現れる毎に開始します。

表17-6. A/D変換自動起動元選択

ADTS3~0	起動元	ADTS3~0	起動元
0 0 0 0	連続変換動作	1 0 0 0	アナログ比較器3
0 0 0 1	アナログ比較器1	1 0 0 1	(予約)
0 0 1 0	外部割り込み要求0	1 0 1 0	(予約)
0 0 1 1	タイマ/カウンタ1溢れ	1 0 1 1	(予約)
0 1 0 0	タイマ/カウンタ1捕獲要求	1 1 0 0	(予約)
0 1 0 1	縮小電力段制御器(PSCR)アナログ同期事象	1 1 0 1	(予約)
0 1 1 0	電力段制御器2(PSC2)アナログ同期事象	1 1 1 0	(予約)
0 1 1 1	アナログ比較器2	1 1 1 1	(予約)

### 17.8.4. ADCH,ADCL - A/Dデータレジスタ (ADC Data Register)

A/D変換が完了すると、その変換結果はこれら2つの結果データレジスタに格納されます。

ADCLが読まれると、2つのA/D変換結果データレジスタはADCHも読まれてしまうまで更新されません。

従って10ビット形態ではADCHに先立ってADCLレジスタが最初に読まれなければなりません。

それにも関わらず、8ビット精度だけで容易に作業するため、A/D多重器選択レジスタ(ADMUX)の左揃え選択(ADLAR)ビットによって結果を左揃えにすることが可能です。このようにして変換結果を得るにはADCHを読むだけで充分です。

ADLAR=0時									
ビット	15	14	13	12	11	10	9	8	
\$2D (\$4D)	-	-	-	-	-	-	ADC9	ADC8	ADCH
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$2C (\$4C)	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ADLAR=1時									
ビット	15	14	13	12	11	10	9	8	
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
ビット	7	6	5	4	3	2	1	0	
	ADC1	ADC0	-	-	-	-	-	-	ADCL

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

ビット	7	6	5	4	3	2	1	0	
(\$77)	ADC8D ACMP3D	ADC7D AMP0-D	ADC5D ACMP2D	ADC4D ACMP3MD	ADC3D ACMPMD	ADC2D ACMP2MD	ADC1D	ADC0D ACMP1D	DIDR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7~0 - ADC8~0D,AMP0-D : ADC7~0 デジタル入力禁止 (ADC8~0,AMP0- Digital Input Disable)

このビットが論理1を書かれると、対応するADCnピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。ADCnピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされない時にこのビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

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

ビット	7	6	5	4	3	2	1	0	
(\$78)	-	-	-	-	ACMP1MD	AMP0+D	ADC10D	ADC9D	DIDR1
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2~0 - ADC10~9D,AMP0+D : ADC10,9,AMP0+ デジタル入力禁止 (ADC10~9, AMP0+ Digital Input Disable)

このビットが論理1を書かれると、対応するADCn(アナログ)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)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。ADCnピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされない時にこのビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

## 17.9. 増幅器

AT90PWM81/161は5,10,20,40倍の利得段付きの2つの作動増幅チャネルが特徴です。結果が10ビットで与えられるにも拘らず、増幅器は8ビット分解能を与える量(精度)にされてしまっています。

増幅器の反転入力に内部的にアナログGNDに切り換えることができます。けれども、増幅器の特性は差動入力で詳述されます。

この増幅器がスイッチ キャパシタ増幅器のため、本文書で増幅器同期クロックと呼ばれる同期信号によってクロック駆動される必要があります。増幅器は同期信号の下降端で入力値を採取します。これは同期と同じ区間でのアナログ信号測定を許します。増幅器に対する最大周波数は250kHzです。

結果精度を保証するため、この増幅器入力には最低4増幅器同期クロック周期間、採取点で静定入力値を持つ必要があります。

増幅変換は電力段制御器(PSC)事象(80頁の表12-11.「1,2,4傾斜動作での同期元」と表12-12.「中央整列動作での同期元」参照)、または8 A/D変換クロックに等しい内部クロックCKADCに同期化され得ます。この同期化がA/D変換クロックの8分周で行なわれる場合、この同期化は採取&保持がCKADC2の特定位相で起きるといような方法でA/D変換器インターフェースによって自動的に行われます。CKADC2がLowの時に使用者によって開始した変換(即ち全ての単独変換と連続変換の最初)はシングル エンド入力変換(前置分周した次のクロック周期から13変換クロック周期)と同じ変換時間になります。CKADC2がHighの時に使用者によって開始した変換は同期化機構のため14変換クロック周期になります。

増幅器を使うための一般的な方法は増幅器n制御/状態レジスタ(AMPnCSR)の起動元選択(AMPnTS1,0)ビット経由で同期化クロックを選ぶことです。その後、増幅器はONに切り替えでき、増幅は各同期事象で行なわれます。この増幅はA/D変換器と無関係に行われます。

増幅チャネルで増幅したA/D変換を始めるには129頁の表17-4.で詳述されるようにA/D変換多重器選択(ADMUX)レジスタを設定しなければなりません。

A/D変換開始はA/D変換制御/状態レジスタA(ADCSRA)でA/D変換開始(ADSC)ビットを設定(1)することによって行なわれます。

その変換が達成されないまでの間、別チャネルでの変換を開始することは不可能です。

AT90PWM81/161での変換は最小変換時間を保証するために増幅器特性の利点を利用します。

変換がADSCビットによって要求されると直ぐにA/D変換が開始されます。増幅器同期化動作を良く理解するため、タイミング図の例が図17-15.で示されます。

図17-16.で示されるように増幅器出力がA/D変換器の採取段階の間に変更される場合、実行中の変換は中止され、増幅器の出力が安定すると直ぐに改めて開始されます。

唯一行う注意はADCclk/4よりも低い起動信号(PSC)周波数を保証することです。

増幅チャネルでの自動起動変換も可能です。この場合の変換はA/D変換制御/状態レジスタB(ADCSRB)の自動起動要因選択(ADTS3~0)ビットで選んだ最後の自動起動事象に後続する最初の増幅器クロック事象で開始されます。自動起動変換での連続変換動作は各変換後にソフトウェアによってADCSRAのADSCビットが設定(1)される場合を除いて不可能です。

図17-15. 増幅器同期化タイミング構成図 (アナログ入力信号での変更との)

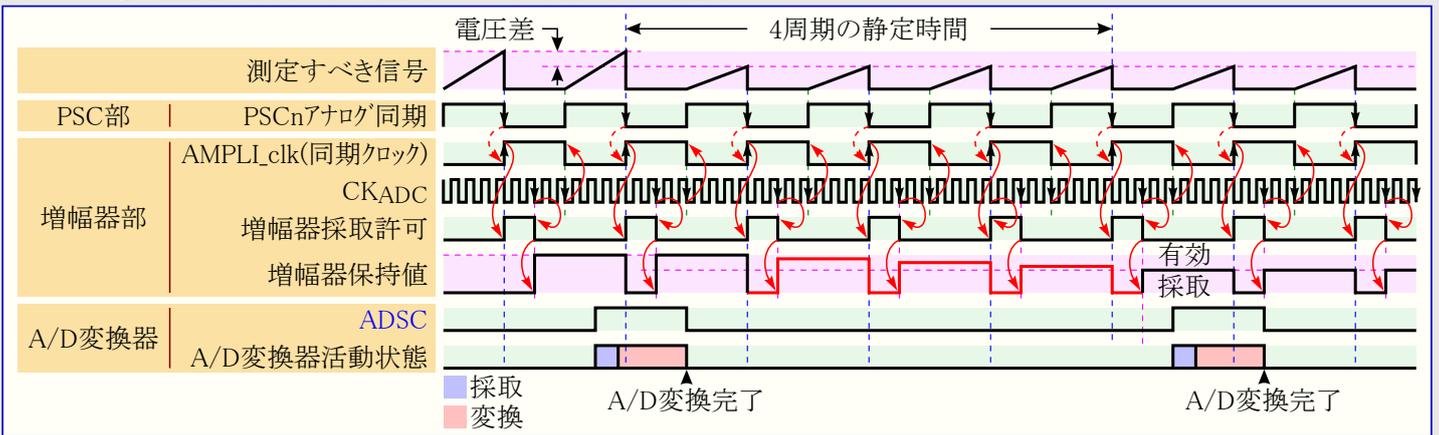
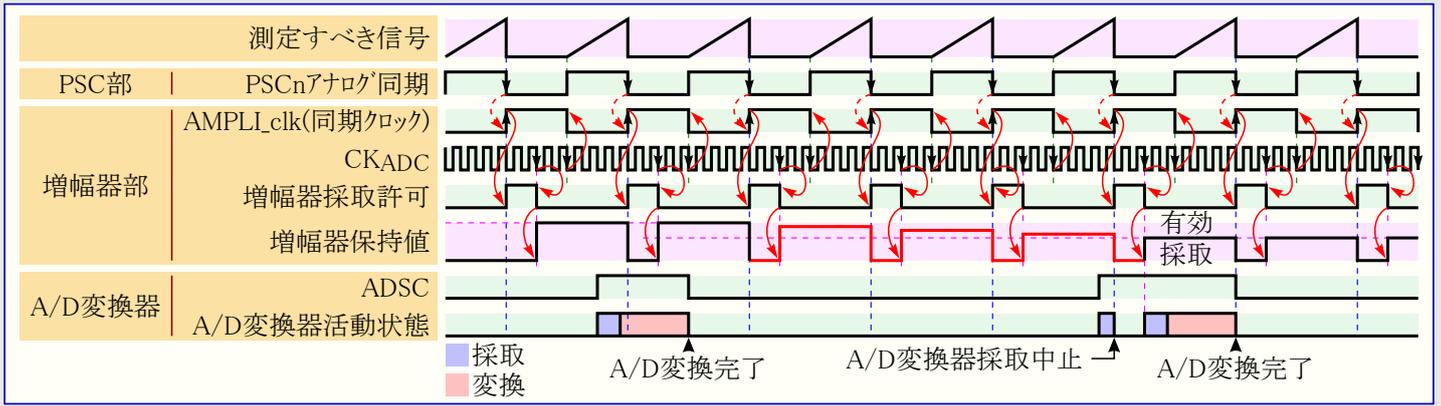


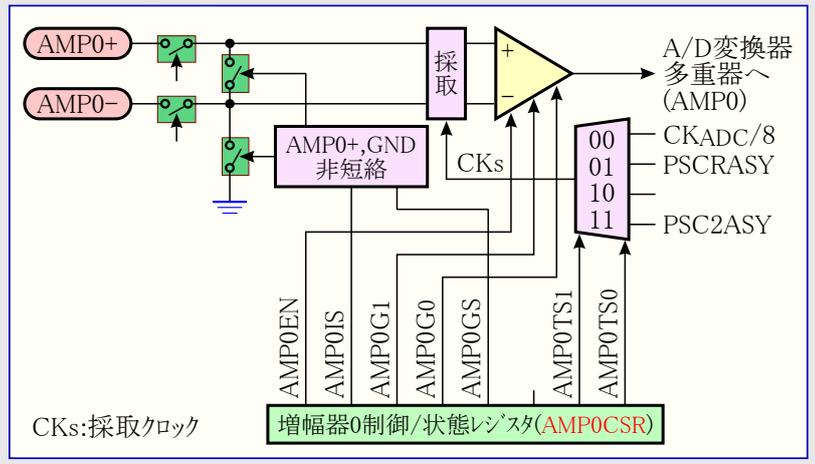
図17-16. 増幅器同期化タイミング構成図 (増幅器クロック切り替えによる増幅器出力変更時のADSC設定)



増幅器の構成図は図17-17.で示されます。

増幅器0 AMP0-のGND選択(AMP0GS)ビットが設定(1)された場合、AMP0-入力は開放され、PD5/AMP0-ピンは他の使用のために開放されます。同時に増幅器の反転入力は内部的なGNDです。

図17-17. 増幅器構成図



## 17.10. 増幅器制御レジスタ

増幅器の形態は1つの専用レジスタ、AMP0CSR経由で制御されます。そして変換の開始はA/D変換器制御/状態レジスタ経由で行なわれます。

変換結果は各々上位側ビットと下位側ビットを含むADCHとADCLレジスタに格納されます。

### 17.10.1. AMP0CSR – 増幅器0制御/状態レジスタ (Amplifier 0 Control and Status register)

ビット (\$79)	7	6	5	4	3	2	1	0	AMP0CSR
	AMP0EN	AMP0IS	AMP0G1	AMP0G0	AMP0GS	–	AMP0TS1	AMP0TS0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7 – AMP0EN : 増幅器0 許可 (Amplifier 0 Enable)

増幅器0を許可するにはこのビットを設定(1)してください。  
増幅器0を禁止するにはこのビットを解除(0)してください。  
変換走行中のこのビットの解除(0)は変換の最後で実施します。

- ビット6 – AMP0IS : 増幅器0 入力切替(短絡) (Amplifier 0 Input Shunt)

増幅器0入力(間)を短絡するにはこのビットを設定(1)してください。増幅器0 AMP0のGND選択(AMP0GS)ビットが設定(1)されている場合、入力間短絡の間、GNDスイッチは開放されます。  
増幅器0を通常使用するにはこのビットを解除(0)してください。

- ビット5,4 – AMP0G1,0 : 増幅器0 利得選択 (Amplifier 0 Gain Selection Bits)

これら2ビットは増幅器0の利得を決めます。  
各種設定は表17-7.で示されます。

表17-7. 増幅器0 利得選択

AMP0G1	AMP0G0	利得
0	0	利得5 (×5)
0	1	利得10 (×10)
1	0	利得20 (×20)
1	1	利得40 (×40)

正確な結果を保証するために利得値が変更されてしまった後、最低4増幅器同期クロック周期間、増幅器入力は静定入力値である必要があります。

- ビット3 – AMP0GS : 増幅器0 AMP0のGND選択 (Amplifier 0 Ground Select of AMP0)

このビットは増幅器0の反転入力を選びます。  
増幅器0反転入力をGNDにするには、このビットを設定(1)してください。  
通常の増幅器0差動入力を使うには、このビットを解除(0)してください。

- ビット1,0 – AMP0TS1,0 : 増幅器0 起動元選択 (Amplifier 0 Trigger Source Selection Bits)

表17-8.に従ってこれら2ビットは増幅器0に対する起動生成事象を選びます。この起動元は増幅チャネルでの変換を始めるために必要です。

表17-8. 増幅器0 自動起動元選択

AMP0TS1	AMP0TS0	同期方法
0	0	A/D変換クロック/8で自動同期
0	1	縮小電力制御器(PSCR)のアナログ同期(PSCRASY)信号で同期化起動
1	0	(予約)
1	1	電力制御器2(PSC2)のアナログ同期(PSC2ASY)信号で同期化起動

## 18. D/A変換器

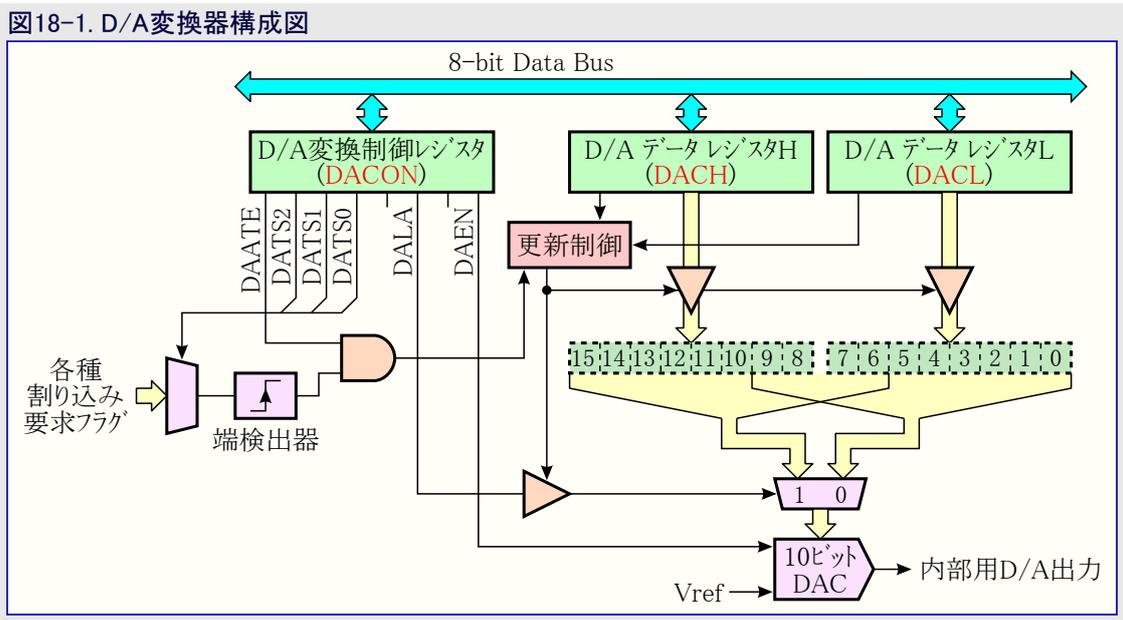
### 18.1. 特徴

- 10ビット分解能
- 8ビット直線精度
- 100mV～(AVCC-100mV)間±0.5 LSB精度
- 出力電圧(VOUT)=D/A変換器値×基準電圧(Vref)/1023
- アナログ比較器反転入力に接続可能

AT90PWM81/161は10ビットD/A変換器が特徴です。このD/A変換器はアナログ比較器に対して使うことができます。

D/A変換器には分離されたアナログ電源供給ピン(AVCC)があります。AVCCはVCCから±0.3Vよりも多く違ってはなりません。このピンの接続方法は127頁の「雑音低減技術」項をご覧ください。

基準電圧はA/D変換器で使われるのと同じです。129頁の「ADMUX - A/D多重器選択レジスタ」をご覧ください。公称2.56Vの内部基準電圧またはAVCCがチップ上で提供されます。この基準電圧は雑音特性向上のため、コンデンサによってAREFピンで外部的にデカップ(雑音分離)できます。



### 18.2. 操作

D/A変換器はD/Aデータレジスタ値に比例したアナログ信号を生成します。

正確な採取周波数制御を持っているので、各種起動事象を通してD/A変換器入力値を更新することが可能です。

### 18.3. 変換の開始

D/A変換器はD/A制御(DACON)レジスタによって構成設定されます。DACONレジスタのD/A許可(DAEN)ビットが設定(1)されると直ぐにD/A変換器はDACONレジスタ設定に従ってD/Aデータ(DACH,DACL)レジスタに存在する値を変換します。

代わりに変換器は様々な供給元によって自動的に起動できます。自動起動はDACONのD/A変換自動起動許可(DAATE)ビットの設定(1)によって許可されます。起動元はDACONのD/A変換起動元選択(DATS2~0)ビット設定によって選ばれます(起動元一覧についてはDATSビット記述をご覧ください)。選んだ起動信号で上昇端が起ると、D/A変換器はDACONレジスタ設定に従い、DACHとDACLレジスタに存在する値を変換します。これは固定間隔での変換開始方法を提供します。変換完了時に未だ起動信号が設定(1)されていると、新しい変換は開始されません。変換中に別の起動信号で上昇端が起る場合、その端(エッジ)は無効です。特定の割り込みが禁止、またはステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)されていても、割り込み要求フラグが設定(1)されることに注意してください。従って変換は割り込みなしで起動できます。とは言い、割り込み要求フラグは次の割り込み事象で新しい(次の)変換を起動するために解除(0)されなければなりません。

### 18.3.1. D/A変換器基準電圧

A/D変換器用の基準電圧(VREF)がD/A変換器に対する変換範囲を示します。VREFはAVCC、内部2.56V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してD/A変換部に接続されます。内部2.56V基準電圧は内蔵基準(バンドギャップ)電圧(VBG)から内部増幅器を通して生成されます。外部AREFピンがD/A変換部へ接続された場合、AREFピンとGND間にコンデンサを接続することにより、基準電圧は雑音耐性をより高められます。VREF(電圧)は高入力インピーダンス電圧計とAREFピンで測定することもできます。VREFは高インピーダンス出力で、容量性負荷のみがシステム内で接続されるべきであることに注意してください。

使用者は基準電圧選択としてAREF、AVCC、内部2.56V基準電圧間の切り替えができます。基準電圧源切り替え後の最初のD/A変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

### 18.4. D/A変換器用レジスタ説明

D/A変換器は3つの専用レジスタ経由で制御されます。

D/A変換制御(DACON)レジスタがD/A変換器構成設定に使われます。

D/Aデータ(DACHとDACL)レジスタが変換すべき値の設定に使われます。

#### 18.4.1. DACON – D/A変換制御レジスタ (Digital to Analog Conversion Control Register)

ビット (\$76)	7	6	5	4	3	2	1	0	
	DAATE	DATS2	DATS1	DATS0	-	DALA	-	DAEN	DACON
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

- ビット7 – DAATE : D/A変換自動起動許可 (DAC Auto Trigger Enable) (有用でなく、互換用に存在)

DACONレジスタのDATS2~0ビットで選んだ起動信号の上昇端でD/A変換器入力値を更新するにはこのビットを設定(1)してください。値がDACHレジスタへ書かれる時にD/A変換器入力を自動的に更新するにはこれを解除(0)してください。

- ビット6~4 – DATS2~0 : D/A変換自動起動要因選択 (DAC Trigger Selection Bits) (有用でなく、互換用に存在)

これらのビットはD/A変換器が自動起動動作で作動する場合だけ必要です。これはDAATEビットが設定(1)される場合の意味です。

表18-1.に従ってこれら3ビットはD/A変換器入力値の更新を生成する割り込み事象を選びます。更新は割り込みが許可であろうとなかろうと、選んだ割り込み要求フラグの上昇端によって生成されます。

表18-1. D/A変換自動起動元選択

DATS2	DATS1	DATS0	起動元
0	0	0	アナログ比較器0
0	0	1	アナログ比較器1
0	1	0	外部割り込み要求0
0	1	1	(予約)
1	0	0	(予約)
1	0	1	(予約)
1	1	0	タイマ/カウンタ1溢れ
1	1	1	タイマ/カウンタ1捕獲要求

- ビット2 – DALA : D/A変換 左揃え選択 (Digital to Analog Left Adjust)

D/A変換器入力データを左揃えにするにはこのビットを設定(1)してください。

D/A変換器入力データを右揃えにするにはこのビットを解除(0)してください。

DALAビットはD/A変換データレジスタの形態に影響を及ぼします。このビットの変更は次(から)のDACH書き込みでのD/A変換器出力に影響を及ぼします。

- ビット5 – Res : 予約 (Reserved Bits)

このビットは予約されています。

- ビット0 – DAEN : D/A変換 許可 (Digital to Analog Enable)

D/A変換器を許可するにはこのビットを設定(1)してください。

D/A変換器を禁止するにはこれを解除(0)してください。

## 18.4.2. DACH, DACL – D/Aデータレジスタ (Digital to Analog Converter Input Register)

DACHとDACLレジスタはアナログ電圧に変換されるべき値を含みます。

DACLレジスタ書き込みはDACHも書かれてしまうまで入力値の更新を禁止します。従ってD/A変換レジスタに10ビット値を書く一般的な方法はまず第1にDACL、次にDACHを書くことです。

8ビット精度だけで容易に作業するために入力値を左揃えにすることが可能です。このようにしてD/A変換値を更新するにはDACHを書くことで充分です。

DALA=0時									
ビット	15	14	13	12	11	10	9	8	
\$39 (\$59)	-	-	-	-	-	-	DAC9	DAC8	DACH
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$38 (\$58)	DAC7	DAC6	DAC5	DAC4	DAC3	DAC2	DAC1	DAC0	DACL
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	
DALA=1時									
ビット	15	14	13	12	11	10	9	8	
	DAC9	DAC8	DAC7	DAC6	DAC5	DAC4	DAC3	DAC2	DACH
ビット	7	6	5	4	3	2	1	0	
	DAC1	DAC0	-	-	-	-	-	-	DACL

10ビットD/A変換器で動作するには2つのレジスタが更新されなければなりません。途中の値を避けるため、アナログ信号に変換される実際のD/A変換入力値は届くことができないレジスタ内に緩衝します。標準動作でのこの隠れたレジスタの更新はDACHレジスタが書かれる時に行なわれます。

D/A変換自動起動許可(DAATE)ビットが設定(1)される場合、D/A変換入力値は起動元選択(DATS2~0)ビットを通して選んだ起動事象で更新されます。

不正なD/A変換入力値を避けるため、更新はDACLとDACHレジスタの各々を書いてしまった後でだけ行なえます。DACH値書き込みだけによる8ビット形態で動作することが可能です。この場合(訳補:結果的に)、更新は各起動事象で行なわれます。

DAATEビットが解除(0)される場合、このD/A変換器は自動更新動作です。DACHレジスタ書き込みがDACHとDACLレジスタ値でD/A変換入力値を自動的に更新します。

これはDAATEビット設定がどうであれ、DACHレジスタも更新されてしまうまで、DACLレジスタの変更がD/A変換器出力に対して無効であることを意味します。従って10ビットで動作するにはDACHに先立ってDACLが最初に書かれなければなりません。8ビット形態での動作ではDACH書き込みがD/A変換器の更新を許します。

## 19. デバッグWIRE 内蔵デバッグ システム

### 19.1. 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گرامング

### 19.2. 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گرام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。

### 19.3. 物理インターフェース

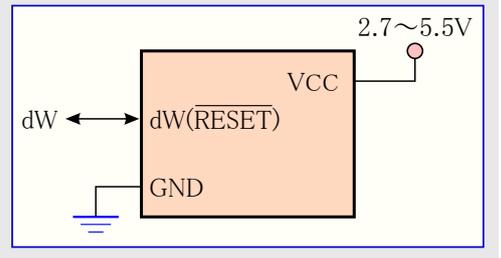
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

図19-1.はエミュレータと許可したデバッグWIREでの対象MCUとの接続の図を示します。システムクロックはデバッグWIREによって影響を及ぼされず、常にCKSELヒューズで選んだクロック元です。

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

- dW/(RESET)線のプルアップ抵抗は10kΩよりも小さくしてはなりません。プルアップ抵抗はデバッグWIRE機能に必要ありません。
- 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19-1. デバッグWIRE構成図



### 19.4. 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。

### 19.5. デバッグWIREの制限

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

デバッグWIREシステムは全速度、換言するとCPUのプログラムが走行する時に全I/O機能を正確エミュレートします。CPUが停止される時にデバッグ(AVR Studio)経由でいくつかのI/Oレジスタをアクセスする間、注意が疎視されなければなりません。この制限の詳細説明についてはデバッグWIRE資料をご覧ください。

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。

### 19.6. デバッグWIREに関連するI/Oメモリ内のレジスタ

次項はデバッグWIREで使うレジスタを記述します。

#### 19.6.1. 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。

## 20. ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング

本デバイスでのブートローダ支援はMCU自身によるプログラムコードのダウンロードとアップロード用の真の書き込み中の読み出しが可能な自己プログラミング機構を提供します。この特徴はフラッシュメモリに常駐するブートローダプログラムを使うMCUによって制御される柔軟な応用ソフトウェア更新を可能にします。ブートローダプログラムはフラッシュメモリ内にコードを書き(プログラム)、コードを読み、またはプログラムメモリからコードを読むのに、利用可能なデータインターフェースと関連する規約のどれもが使えます。ブートローダ領域内のプログラムコードはブートローダメモリを含むフラッシュメモリ全体を書く能力を持ちます。従ってブートローダは自身をも変更でき、この機能がそれ以上必要とされないなら、そのコードから自身を消去することもできます。ブートローダメモリの容量はヒューズで設定可能で、ブートローダは個別に設定可能な2組の独立したブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

### 20.1. 特徴

- 書き込み中読める(Read-While-Write)自己プログラミング
- 柔軟性のあるブートローダメモリ容量
- 高い安全性 (柔軟な保護用の独立したブート施錠ビット)
- リセットベクタ選択用の独立したヒューズ
- 最適化されたページ容量 (注1)
- 効率的なコード手法
- 効率的な読み-変更-書き(リード-モデファイライト)支援

**注1:** ページはプログラミング中に使われる多数のバイトから成るフラッシュメモリの区画です(153頁の表21-12参照)。このページ構成は通常動作に影響を及ぼしません。

### 20.2. フラッシュメモリの応用領域とブートローダ領域

フラッシュメモリは応用領域とブートローダ領域の2つの主な領域で構成されます(図20-2参照)。各領域の容量は148頁の表20-7と図20-2で示されるようにBOOTSZヒューズによって設定されます。これら2つの領域は個別の施錠ビットの組を持つため、異なる保護基準を持てます。

#### 20.2.1. 応用領域

応用領域は応用コードを格納するのに使われるフラッシュメモリの領域です。応用領域用保護基準は応用ブート施錠ビット(ブート施錠ビット0)によって選べます(141頁の表20-2参照)。応用領域から実行される時にSPM命令が禁止されるので、応用領域はどんなブートローダコードも決して格納し得ません。

#### 20.2.2. ブートローダ領域 (BLS)

応用領域が応用コード格納用に使われるのに対して、SPM命令はBLSから実行する時にだけプログラミングを始められるので、ブートローダソフトウェアはBLSに格納されなければなりません。SPM命令はBLS自身を含む全てのフラッシュメモリをアクセスできます。ブートローダ領域用保護基準はブートローダ施錠ビット(ブート施錠ビット1)によって選べます(141頁の表20-3参照)。

### 20.3. フラッシュメモリの書き込み中に読み出し可能な領域と不能な領域

どちらのアドレスがプログラミングされるかによって、CPUが書き込み中の読み出しを支援するか、ブートローダソフトウェアが更新中にCPUが停止されるかのどちらです。上で記述されるようなBOOTSZヒューズによって設定可能な2つの領域に加え、フラッシュメモリは書き込み中読み出し可能な(RWW)領域と書き込み中読み出し不能な(NRWW)領域の2つの固定領域にも分けられます。RWWとNRWW領域間の境界は140頁の図20-2と148頁の表20-9で与えられます。この2つの領域間の主な違いを次に示します。

- RWW領域側に配置されたページを消去または書くとき、NRWW領域はその動作中に読むことができます。
- NRWW領域側に配置されたページを消去または書くとき、その全ての動作中にCPUは停止されます。

ブートローダソフトウェア動作中、使用者ソフトウェアがRWW領域側に配置されるどのコードも決して読めないことに注意してください。「書き込み中読み出し可能領域」という記述はプログラミング(消去または書き込み)される領域としての引用で、ブートローダソフトウェアが更新中に実際に読まれる領域ではありません。

**訳補:** 上の記述はNRWW領域からRWW領域をプログラミングするという前提で、消去または書き込みを行う側ではなく、行われる側でこの名称が定義されていることを意味します。即ち、NRWW領域からRWW領域をプログラミングすると、NRWW領域のプログラムは通常通り動作する(即ち読める)ので、プログラミングされる側はRWW領域と名付けられ、この逆ではCPUが停止する(即ち読めない)ので、NRWW領域と名付けられているという意味です。

### 20.3.1. RWW – 書き込み中読み出し可能領域

ブートローダソフトウェア更新がRWW領域側のページをプログラミングする場合、フラッシュメモリからコードを読むことが可能ですが、NRWW領域に配置されるコードだけです。プログラミング実行中、そのソフトウェアはRWW領域が決して読まれないことを保証しなければなりません。使用者ソフトウェアがプログラミング中に(例えば、RCALL,RJMP,LPM系命令または割り込みによって)RWW領域側に配置されるコードを読もうとすると、そのソフトウェアは未知の状態へ行き着くかもしれません。これを避けるために割り込みは禁止またはブートローダ領域へ移動のどちらかにされるべきです。ブートローダ領域は常にNRWW領域に配置されます。RWW領域が読み出しに対して妨げられている限り、SPM命令制御/状態レジスタ(SPMCSR)のRWW領域多忙(RWWSB)ビットは論理1として読みます。プログラミングが完了した後、RWW領域に配置したコードを読む前にRWWSBはソフトウェアによって解除(0)されなければなりません。RWWSBを解除(0)する方法の詳細については142頁の「SPMCSR – SPM命令制御/状態レジスタ」をご覧ください。

### 20.3.2. NRWW – 書き込み中読み出し不能領域

NRWW領域に配置したコードはブートローダソフトウェアがRWW領域内のページを更新する時に読めます。ブートローダコードがNRWW領域を更新するとき、全てのページ消去またはページ書き込み動作中にCPUが停止されます。

表20-1. 書き込み中読み出し可能機能

プログラミング中にZポインタで指定される領域	プログラミング中に読める領域	CPU動作	RWW機能支援
RWW領域	NRWW領域	通常動作	あり
NRWW領域	なし	停止	なし

図20-1. RWW領域とNRWW領域の関係

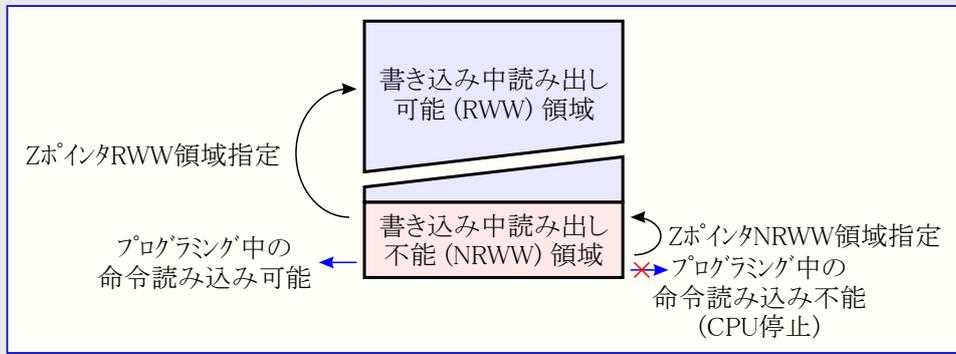
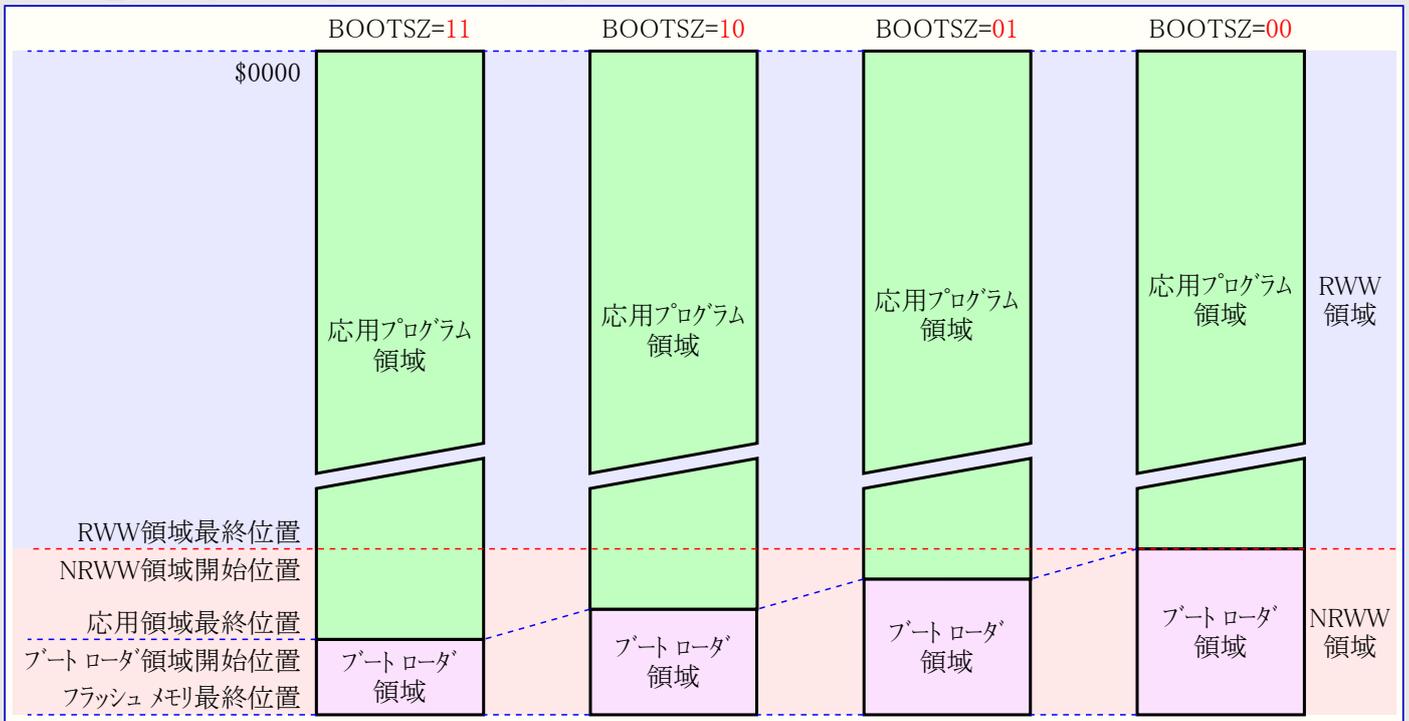


図20-2. 選択によるプログラム用フラッシュメモリの領域分割



注: 図内の各パラメータについては148頁の表20-7.で与えられます。

## 20.4. ブートローダ施錠ビット

ブートローダ能力が必要とされないなら、フラッシュメモリ全体が応用コード用に利用可能です。ブートローダは個別に設定可能な独立した2組のブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

使用者は以下を選べます。

- MCUによって更新するソフトウェアからフラッシュメモリ全体を保護
- MCUによって更新するソフトウェアからフラッシュメモリのブートローダ領域だけを保護
- MCUによって更新するソフトウェアからフラッシュメモリの応用領域だけを保護
- フラッシュメモリ全体で更新するソフトウェアを許可

より多くの詳細については表20-2と表20-3をご覧ください。ブート施錠ビットはソフトウェアと直列または並列のプログラミング動作で設定(0)できますが、これらのビットはチップ消去指令によってのみ解除(1)できます。一般書き込み禁止(LB動作種別2)はSPM命令によるフラッシュメモリのプログラミングを制御しません。同様に、試みられたなら、一般読み書き禁止(LB動作種別3)はLPM命令とSPM命令による読み込みも書き込みも制御しません。(訳補: 一般LBはLPM/SPM命令に関して無関係の意)

表20-2. 応用領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB0 動作種別	BLB02	BLB01	保護種別
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注)
4	0	1	ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注)

注: BLB02=0で、割り込みベクタがブートローダ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

表20-3. ブートローダ領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB1 動作種別	BLB12	BLB11	保護種別
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注)
4	0	1	応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注)

注: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

## 20.5. ブートローダプログラムへの移行

ブートローダへの移行は応用プログラムから分岐(Jump)または呼び出し(Call)によって行います。これはSPIインターフェース経由で受信した指令のような起点によって始められるかもしれませんが、代わりに、リセット後にリセットベクタがブートローダ領域開始アドレスを指示するようにブートリセット(BOTRST)ヒューズをプログラム(0)できます。この場合、ブートローダはリセット後に開始されます。応用コードが設定された(書かれた)後、そのプログラム(ブートローダ)は応用コードの実行を始めることができます。このヒューズがMCU自身によって変更できないことに注意してください。これは一旦ブートリセットヒューズがプログラム(0)されると、リセットベクタは常にブートローダリセットを指示し、このヒューズが直列プログラミングまたは並列プログラミングインターフェースを通してのみ変更できることを意味します。

表20-4. ブートリセットヒューズ (0=プログラム、1=非プログラム)

BOTRST	リセット後実行開始アドレス (リセットベクタ)	
0	ブートローダリセット	ブートローダ開始アドレス(148頁の表20-7.参照)
1	応用リセット	\$0000

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

このレジスタはブートローダ動作を制御するために必要とされる制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$37 (\$57)	SPMIE	RWWSB	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	SPMCSR
Read/Write	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

### • ビット7 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIEビットが**1**を書かれ、ステータスレジスタ(SREG)の全割り込み許可(0)ビットが設定(1)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みはSPMCSRのSPM操作許可(SPMEN)ビットが解除(0)されている限り実行されます。

### • ビット6 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)

RWW領域に自己プログラミング(ページ消去またはページ書き込み)操作が開始されると、RWWSBがハードウェアによって設定(1)されます。RWWSBビットが設定(1)されると、RWW領域はアクセスできません。自己プログラミングが完了された後にRWWSREビットが**1**を書かれると、このRWWSBビットは解除(0)されます。同様に、ページ設定操作が開始されると、RWWSBビットは自動的に解除(0)されます。

### • ビット5 – SIGRD : 識票列読み出し (Signature Row Read)

このビットがSPMENと同時に**1**を書かれると、次の3クロック周期内のLPM命令は識票列からバイトを転送先レジスタに読みます。詳細については146頁の「ソフトウェアからの識票列読み出し」をご覧ください。

SIGRDとSPMENが設定(1)された後の4クロック周期内のSPM命令は無効です。この操作は将来の使用に対して予約されており、使われるべきではありません。

### • ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域にプログラミング(ページ消去またはページ書き込み)すると、RWW領域は(RWWSBがハードウェアによって設定(1)され)読み出しに対して防がれます。RWW領域を再許可するために使用者ソフトウェアはプログラミングが完了(SPMENが解除(0)されるまで待たねばなりません。その後、RWWSREビットがSPMENと同時に**1**を書かれると、次の4クロック周期内のSPM命令がRWW領域を再許可します。ページ消去またはページ書き込みでフラッシュメモリが多忙(SPMEN=1)の間、RWW領域は再許可できません。フラッシュメモリが設定されている間にRWWSREビットが書かれると、フラッシュメモリ設定操作は失敗し、設定したデータが失われます。

### • ビット3 – BLBSET : ブート施錠ビット設定 (Boot Lock Bits Set)

このビットがSPMENと同時に**1**を書かれると、次の4クロック周期内のSPM命令はR0のデータに従ったブート施錠ビットを設定します。R1のデータとZポインタのアドレスは無視されます。BLBSETビットは施錠ビットの設定完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。

SPMCSRでBLBSETとSPMENが設定(1)された後の3クロック周期内のLPM命令は(ZポインタのZ0によって)ヒューズビットまたは施錠ビットのどちらかを転送先レジスタに読みます。詳細については145頁の「ソフトウェアからのヒューズビットと施錠ビットの読み出し」をご覧ください。

### • ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に**1**を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の書き込み動作中にCPUは停止されます。

### • ビット1 – PGERs : ページ消去 (Page Erase)

このビットがSPMENと同時に**1**を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERsビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の消去中にCPUは停止されます。

### • ビット0 – SPMEN : SPM操作許可 (Store Program Memory Enable)

このビットは次の4クロック周期間のSPM命令を許可します。このビットがRWWSRE, BLBSET, PGWRT, PGERsのどれかと共に**1**を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。SPMENだけが書かれると、続くSPM命令はZポインタによってアドレス指定したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SPMENビットはSPM命令の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ消去とページ書き込み中、SPMENビットはその動作が完了されるまで**1**に留まります。

下位6ビットに100001, 010001, 001001, 000101, 000011, 000001以外のどんな組み合わせを書いても無効です。

## 20.6. 自己プログラミング中のフラッシュメモリのアドレス指定

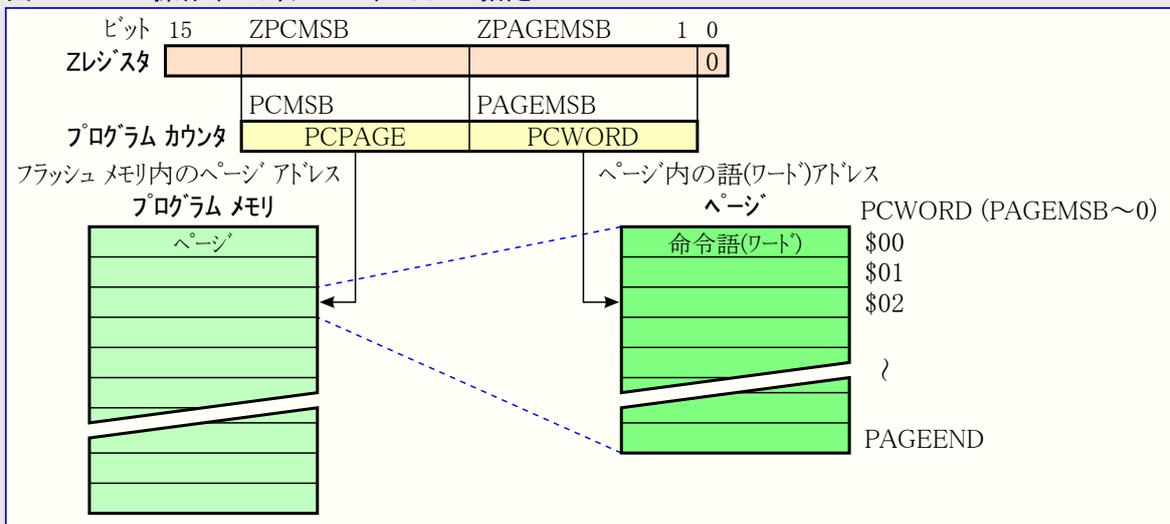
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

ビット	15	14	13	12	11	10	9	8
ZH (R31)	Z15	Z14	Z13	Z12	Z11	Z10	Z9	Z8
ビット	7	6	5	4	3	2	1	0
ZL (R30)	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

フラッシュメモリがページで構成されるため(153頁の表21-12参照)、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つように扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは図20-3.で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってブートローダソフトウェアはページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。一旦プログラミング操作が開始されると、このアドレスはラッチされ、Zポインタは他の操作に使えます。

Zポインタを使わないSPM操作はブートローダ施錠ビット設定だけです。この操作でZレジスタの内容は無視され、無効です。LPM命令もアドレスを格納するのにZポインタを使います。この命令はフラッシュメモリのバイト単位をアドレス指定するので、Zポインタの最下位ビット(Z0)も使われます。

図20-3. SPM操作中のフラッシュメモリのアドレス指定



注: 図内で使った各変数は148頁の表20-10.で一覧されます。  
PCPAGEとPCWORDは153頁の表21-12.で一覧されます。(訳注: 共通性から追加)

## 20.7. フラッシュメモリの自己プログラミング

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

### 手段1 (ページ消去前の一時緩衝部格納)

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

### 手段2 (ページ消去後の一時緩衝部格納)

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

ページの一部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-モディファイ-ライト)機能をブートローダが提供します。手段2.が使われる場合、ページが既に消去されているため、格納中の旧データを読むことができません。ページ一時緩衝部は乱順でアクセスできます。ページ消去とページ書き込み操作の両方で使われるページアドレスは同じページをアドレス指定することが非常に重要です。アセンブリ言語でのコード例については147頁の「アセンブリ言語による簡単なブートローダ例」をご覧ください。

### 20.7.1. SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタにアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'X0000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGEに書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- RWW領域のページ消去 : ページ消去中、NRWW領域は読めます。
- NRWW領域のページ消去 : ページ消去中、CPUは停止されます。

### 20.7.2. ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタにアドレス、R1:R0にデータを設定してSPMCSRに'00000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORDの内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのRWWSREビット書き込みによって自動的に消去されます。システムリセット後も消去されています。一時緩衝部を消去せずに各アドレスへ複数回書くことができないことに注意してください。

SPMページ設定操作の途中でEEPROMが書かれると、設定した全データが失われます。

### 20.7.3. ページ書き込みの実行

ページ書き込みを行うにはZポインタにアドレスを設定してSPMCSRに'X0000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGEに書かれなければなりません。Zポインタの他のビットは0を書かれなければなりません。

- RWW領域のページ書き込み : ページ書き込み中、NRWW領域は読めます。
- NRWW領域のページ書き込み : ページ書き込み中、CPUは停止されます。

### 20.7.4. SPM操作可割り込みの使用法

SPM操作可割り込みが許可されると、SPMCSRのSPMENビットが解除(0)されている時にSPM操作可割り込みが継続的に発生します。これはソフトウェアでSPMCSRをポーリングする代わりにこの割り込みが使えることを意味します。SPM操作可割り込み使用時、割り込みが読み出しに対して防がれる時にRWW領域をアクセスするのを避けるために、割り込みベクタはブートローダ領域(BLS)へ移動されるべきです。割り込み(ベクタ)の移動法は40頁の「応用領域とブート領域間の割り込みベクタ移動」で記述されます。

### 20.7.5. ブートローダ領域更新中の考慮

ブート施錠ビット11(BLB11)が非プログラム(1)にされたままとすることによって使用者がブートローダ領域に更新を許す場合、特別な注意が祓われなければなりません。ブートローダ自身への予期せぬ書き込みはブートローダ全体を不正に使得、更にソフトウェアの更新が不可能になるかもしれません。ブートローダ自体の変更が必要ないなら、内部ソフトウェアのどんな変更からもブートローダを保護するためにブート施錠ビット11(BLB11)をプログラム(0)することが推奨されます。

### 20.7.6. 自己プログラミング中のRWW領域読み込みの防止

自己プログラミング中(ページ消去もページ書き込みも)、RWW領域は読み出しに対して常に防がれます。使用者ソフトウェアそれ自身が自己プログラミング操作中にこの領域がアドレス指定されるのを防止しなければなりません。SPMCSRのRWWSBはRWW領域が多忙である限り設定(1)されます。自己プログラミング中の割り込みベクタ表は40頁の「応用領域とブート領域間の割り込みベクタ移動」で記述されるようにブートローダ領域(BLS)へ移動されるべきか、または割り込みが禁止されなければなりません。プログラミングが完了した後にRWW領域をアドレス指定する前に、使用者ソフトウェアはRWWSREの書き込みによってRWWSBを解除(0)しなければなりません。例については147頁の「アセンブリ言語による簡単なブートローダ例」をご覧ください。

### 20.7.7. SPM命令によるブートローダ施錠ビットの設定

ブートローダ施錠ビットを解除(0)するには希望したデータをR0に設定してSPMCSRに'X0001001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。アクセス可能な施錠ビットはMCUによるソフトウェア更新からも応用とブートローダ領域を保護できるブート施錠ビットだけです。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	1	1

フラッシュメモリのアクセスに影響を及ぼすブートローダ施錠ビットの各種設定法については表20-2と表20-3をご覧ください。

R0のビット5~2が解除(0)される場合、SPMCSRでSPMENとブート施錠ビット設定(BLBSET)ビットが設定(1)された後4クロック周期内にSPM命令が実行されると、対応するブート施錠ビットがプログラム(0)されます。この操作中、Zポインタは関係ありませんが、将来との共通性のために、(施錠ビット読み出しに使われるのと同じ)\$0001でZポイン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7,6,1,0は1に設定することも推奨されます。施錠ビットをプログラミングするとき、この操作中に全てのフラッシュメモリは読むことができます。

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

EEPROM書き込み動作がフラッシュメモリへの全ソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(ECCR)のEEPROM書き込み許可(EWE)ビットを検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前にこのビットが解除(0)されているのを確認することが推奨されます。

### 20.7.9. ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビットの両方を読むことが可能です。施錠ビットを読むにはZポイントに\$0001を設定してSPMCSRのSPMENとブート施錠ビット設定(BLBSET)ビットを設定(1)してください。SPMENとBLBSETビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SPMENとBLBSETビットは施錠ビット読み出しの完了で、または3 CPU周期内にLPM命令が実行されないか、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SPMENとBLBSETビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	BLB12	BLB11	BLB02	BLB01	LB2	LB1

ヒューズ下位ビットを読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポイントに\$0000を設定してSPMCSRのSPMENとBLBSETビットを設定(1)してください。SPMENとBLBSETビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ下位ビット(FLB)の値が転送先レジスタに格納されます。ヒューズ下位ビットの配置と詳細な記述については151頁の表21-6を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様に、ヒューズ上位ビットを読むにはZポイントに\$0003を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるようにヒューズ上位ビット(FHB)の値が転送先レジスタに格納されます。ヒューズ上位ビットの配置と詳細な記述については150頁の表21-5を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

拡張ヒューズビットを読む時はZポイントに\$0002を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるように拡張ヒューズビット(EFB)の値が転送先レジスタに格納されます。拡張ヒューズビットの配置と詳細な記述については150頁の表21-4を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	EFB7	EFB6	EFB5	EFB4	-	EFB2	EFB1	EFB0

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

### 20.7.10. ソフトウェアからの識票列読み出し

ソフトウェアから識票列を読むには表20-5.で与えられる識票バイトアドレスをZポイントに設定してSPMCSRのSPMENと識票列読み出し(SIGRD)ビットを設定(1)してください。SPMENとSIGRDビットがSPMCSRに設定された後、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周期で自動解除(0)し、SPMCSRはこれらの周期間の更なる書き込みに対して施錠されます。SPMENとSIGRD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。

表20-5. 識票列アドレス指定 (データ内'/'はAT90PWM81/AT90PWM161)

アドレス	データ	識票バイト	アドレス	データ	識票バイト
\$0001	\$xx	8MHz RC発振器OSCCAL校正バイト	\$0000	\$1E	デバイス識票バイト 1 (製造者識別)
\$0003	\$xx	(予約)	\$0002	\$93/\$94	デバイス識票バイト 2 (フラッシュ容量)
\$0005	\$xx	温度感知器変位(オフセット): TSOFFSET	\$0004	\$88/\$8B	デバイス識票バイト 3 (デバイス種別)
\$0007	\$xx	温度感知器利得: TSGAIN (注1)	\$0006	\$xx	(予約)
\$000F	\$xx	種類でのロット番号(ASCII)第1バイト(最上位)	\$000E	\$xx	種類でのロット番号(ASCII)第2バイト
\$0011	\$xx	種類でのロット番号(ASCII)第3バイト	\$0010	\$xx	種類でのロット番号(ASCII)第4バイト
\$0013	\$xx	種類でのロット番号(ASCII)第5バイト	\$0012	\$xx	種類でのロット番号(ASCII)第5バイト
\$003D	\$xx	室温最終検査Vref上位バイト (注3)	\$003C	\$xx	室温最終検査Vref下位バイト (注2)
\$003F	\$xx	高温最終検査Vref上位バイト (注5)	\$003E	\$xx	高温最終検査Vref上位バイト (注4)

注1: TSGAINの代表値は\$80=128です。

注2: 注3をご覧ください。

注3: 室温最終検査Vref上位バイトと下位バイトは次のとおりです。  
Vrefに対する代表値=2.56Vで、上位バイト=\$0A、下位バイト=\$00です。  
これは室温最終検査Vref=\$0A00=2560=Vref×1000を意味します。

注4: 値の形式の詳細は注3をご覧ください。

注5: 値の形式の詳細は注3をご覧ください。

### 20.7.11. フラッシュメモリデータ化けの防止

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

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

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

- そのシステムでファームウェア更新が必要ない場合、どんなファームウェア更新をも防ぐためにファームウェア施錠ビットをプログラム(0)してください。
- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアを「パワーダウン休止動作」に保ってください。これはCPUが命令の復号と実行を試みるのを防ぎ、SPMCSR従ってフラッシュメモリを予期せぬ書き込みから効果的に保護します。

### 20.7.12. SPM命令使用時のフラッシュメモリ用プログラミング(書き込み)時間

校正された内蔵RC発振器がフラッシュメモリアクセス時間に使われます。表20-6.はCPUからのフラッシュメモリアクセスに対する代表的なプログラミング時間を示します。

表20-6. SPM命令によるフラッシュメモリのプログラミング時間

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

## 20.7.13. アセンブリ言語による簡単なブートローダ例

このルーチンはRAMからフラッシュメモリへ1ページのデータを書きます。RAM内の最初のデータ位置はYレジスタによって指示され、フラッシュメモリ内の最初のデータ位置はZレジスタによって指示されます。異常処理は含まれません。このルーチン(少なくともSPMJサブルーチン)はブートローダ領域側に配置されなければなりません。NRWW領域側のコードだけが自己プログラミング(ページ消去とページ書き込み)中に読めます。使用レジスタはR0,R1,TMP,CNTL,CNTH,SPMCで、レジスタの保存と復帰はこのルーチン内に含まれず、使用レジスタはコード量を犠牲にすれば最適化できます。割り込み表がブートローダ領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更になります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令	注釈
	.EQU PGSZB = PAGESIZE*2	; PGSZBはページ内のバイト数です。(PAGESIZEは語(ワード)数)
	.ORG SMALLBOOTSTART	;
WRPG:	LDI SPMC, (1<<PGERS)+(1<<SPMEN)	; [ページ消去]
	RCALL SPMJ	; ページ消去SPMCSR値を取得
		; ページ消去
		; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWSRE)+(1<<SPMEN)	; RWW領域読み出し許可SPMCSR値を取得
	RCALL SPMJ	; RWW領域読み出し許可
		; [RAMからフラッシュページ一時緩衝部へ転送]
	LDI CNTL, LOW(PGSZB)	; バイト計数器を初期化
	LDI CNTH, HIGH(PGSZB)	; (削除)
WLP:	LD R0, Y+	; RAM上の下位データを取得(ポインタ進行)
	LD R1, Y+	; RAM上の上位データを取得(ポインタ進行)
	LDI SPMC, (1<<SPMEN)	; ページ一時緩衝部書き込みSPMCSR値を取得
	RCALL SPMJ	; 対応語(ワード)データをページ一時緩衝部に設定
	ADIW ZH:ZL, 2	; ページ一時緩衝部ポインタ進行
	SBIW CNTH:CNTH, 2	; 計数器を減数 (SUBI)
	BRNE WLP	; 指定バイト数分継続
		; [ページ書き込み]
		; ページ一時緩衝部先頭にポインタを復帰
	SUBI ZL, LOW(PGSZB)	; (削除)
	SBCI ZH, HIGH(PGSZB)	; (削除)
	LDI SPMC, (1<<PGWRT)+(1<<SPMEN)	; フラッシュ書き込みSPMCSR値を取得
	RCALL SPMJ	; フラッシュメモリページ書き込み
		; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWSRE)+(1<<SPMEN)	; RWW領域読み出し許可SPMCSR値を取得
	RCALL SPMJ	; RWW領域読み出し許可
		; [読み戻し照合(任意)]
	LDI CNTL, LOW(PGSZB)	; バイト計数器を初期化
	LDI CNTH, HIGH(PGSZB)	; (削除)
	SUBI YL, LOW(PGSZB)	; RAMデータ先頭にポインタを復帰
	SBCI YH, HIGH(PGSZB)	;
RLP:	LPM R0, Z+	; フラッシュメモリから1バイト取得(ポインタ進行)
	LD R1, Y+	; RAMから1バイトデータを取得(ポインタ進行)
	CPSE R0, R1	; 値一致でスキップ
	RJMP ERROR	; 不一致で異常処理へ
		;
	SBIW CNTH:CNTH, 1	; 計数器を減数 (SUBI)
	BRNE RLP	; 指定バイト数分継続
		; [RWW領域へ復帰]
RTN:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRS TMP, RWWSB	; RWW領域多忙でスキップ
	RET	; 準備可で呼び出し元へ復帰
		; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWSRE)+(1<<SPMEN)	; RWW領域読み出し許可SPMCSR値を取得
	RCALL SPMJ	; RWW領域読み出し許可
	RJMP RTN	; RWW領域準備可まで待機へ
		; [SPM命令実行サブルーチン]
SPMJ:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRC TMP, SPMEN	; 操作可能(直前のSPM完了)でスキップ
	RJMP SPMJ	; 操作可まで待機
		;
	IN TMP, SREG	; ステータスレジスタ値を保存
	CLI	; 全割り込み禁止
WAIT:	SBIC EECR, EEWEIF	; EEPROM書き込み中以外でスキップ
	RJMP WAIT	; EEPROM書き込み完了まで待機
		;
	OUT SPMC, SPMC	; SPM動作指定
	SPM	; 対応SPM動作実行
	OUT SREG, SREG	; ステータスレジスタ値を復帰
	RET	; 呼び出し元へ復帰

## 20.7.14. ブートローダのパラメータ

自己プログラミングの記述で使ったパラメータは表20-7～9.で与えられます。

表20-7. 応用領域とブートローダ領域の分割設定 (上段:AT90PWM81, 下段:AT90PWM161)

BOOTSZ1	BOOTSZ0	ブートローダ領域		アドレス範囲		
		容量 (語)	ページ数	応用プログラム領域	ブートローダプログラム領域	ブートリセットアドレス
1	1	128	4	\$0000～\$0F7F	\$0F80～\$0FFF	\$0F80
		256		\$0000～\$1EFF	\$1F00～\$1FFF	\$1F00
1	0	256	8	\$0000～\$0EFF	\$0F00～\$0FFF	\$0F00
		512		\$0000～\$1DFF	\$1E00～\$1FFF	\$1E00
0	1	512	16	\$0000～\$0DFF	\$0E00～\$0FFF	\$0E00
		1024		\$0000～\$1BFF	\$1C00～\$1FFF	\$1C00
0	0	1024	32	\$0000～\$0BFF	\$0C00～\$0FFF	\$0C00
		2048		\$0000～\$17FF	\$1800～\$1FFF	\$1800

注: 各種BOOTSZヒューズ設定は図20-2.で示されます。

(訳注) 原書の表20-7.(AT90PWM81)と表20-8.(AT90PWM161)は表20-7.として纏めました。

表20-9. RWW領域とNRWW領域の範囲 (上段:AT90PWM81, 下段:AT90PWM161)

領域	ページ数	アドレス範囲
書き込み中読み出し可能(RWW)領域	96	\$0000～\$0BFF \$0000～\$1BFF
書き込み中読み出し不能(NRWW)領域	32	\$0C00～\$0FFF \$1C00～\$1FFF

これら2つの領域についての詳細に関しては140頁の「RWW - 書き込み中読み出し可能領域」と「NRWW - 書き込み中読み出し不能領域」をご覧ください。

表20-10. 図20-3.で使用した各変数説明とZポイントの配置 (上段:AT90PWM81, 下段:AT90PWM161)

変数名	対応値		意味
	PC	Zポイント (注)	
PCMSB	PC11 PC12		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ビット、PC11～0) (プログラムカウンタは13ビット、PC12～0)
PAGEMSB	PC4		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32語(ワード)には5ビット PC4～0が必要)
ZPCMSB		Z12	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		Z5	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	PC11～5 PC12～6	Z12～6 Z13～7	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	PC4～0 PC5～0	Z5～1 Z6～1	プログラムカウンタ 語(ワード)アドレス: 一時緩衝部格納用語(ワード)選択 (ページ書き込み操作中は0でなければなりません。)

注: Z15～13/14: 常に無視されます。

Z0: 全てのSPM命令に対して0であるべきで、LPM命令に対するビット選択です。

自己プログラミング中のZポイントの使用については143頁の「自己プログラミング中のフラッシュメモリのアドレス指定」をご覧ください。

## 21. メモリプログラミング

### 21.1. プログラムメモリとデータメモリ用施錠ビット

AT90PWM81/161は非プログラム(1)のままか、表21-2.で一覧される付加機能を得るためにプログラム(0)できる6つの施錠ビットを提供します。この施錠ビットはチップ消去指令でのみ1に消去できます。

表21-1. 施錠ビットハイトの内容

名称	ビット番号	意味	既定値(注)
-	7		1 (非プログラム)
-	6		1 (非プログラム)
BLB12	5	ブートローダ領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB11	4		1 (非プログラム)
BLB02	3	応用領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB01	2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表21-2. 施錠ビットの保護種別

メモリ施錠ビット			保護種別
LB 種別	LB2	LB1	直列、並列、またはデバッグWIRE経由でプログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズビットとブート施錠ビットが固定されます。(注1)
LB0 種別	BLB02	BLB01	フラッシュメモリの応用プログラム領域に対する保護
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
4	0	1	ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
LB1 種別	BLB12	BLB11	フラッシュメモリのブートローダプログラム領域に対する保護
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)
4	0	1	応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)

注: 0はプログラム、1は非プログラムを意味します。

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

注2: BLB02=0で、割り込みベクタがブートローダ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

注3: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

(訳注) 原書の表21-2.と表21-3.は表21-2.として纏めました。

## 21.2. ヒューズ ビット

AT90PWM81/161には3つのヒューズ バイトがあります。表21-4.~6.は全てのヒューズの概略機能とヒューズ バイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

表21-4. 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
PSC2RB	7	電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSC2RBA	6	OUT22と23用の電力段制御器2(PSC2)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRRB	5	縮小電力段制御器(PSCR)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRV	4	PSCOUTとPSCOUTRのリセット値	1 (非プログラム) High
PSCINRB	3	PSCとPSCRの入力 リセット時動作	1 (非プログラム) 標準ポート動作
BODLEVEL2	2	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注1)	1 (非プログラム)
BODLEVEL1	1		0 (プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: BODLEVELヒューズの復号については33頁の表7-2.をご覧ください。

### 21.2.1. リセット中の電力段制御器(PSC)出力の動き

外部部品の安全のため、リセット中のPSC出力状態はPSCRV,PSCRRB,PSC2RBA,PSC2RBヒューズによってプログラム(設定)できます。これらのヒューズは以下のように拡張ヒューズ バイトに配置されます(表21-4.をご覧ください)。

PSCRVはPSC0RBとPSC2RBヒューズによって選んだPSC出力でHighかLowのどちらの状態を強制するのかを与えます。PSCRVヒューズが1に等しい(非プログラム)なら、選んだPSC出力はHigh状態に強制されます。

PSCRRBヒューズが1に等しい(非プログラム)なら、PSCOUTR0とPSCOUTR1は標準ポート動作を保ちます。PSCRRBヒューズが0に等しい(プログラム)なら、PSCOUTR0とPSCOUTR1はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUTR0とPSCOUTR1はPSC0同期/出力構成(PSOC0)レジスタが書かれるまで、この強制された状態を保ちます。

PSC2RBヒューズが1に等しい(非プログラム)なら、PSCOUT20とPSCOUT21は標準ポート動作を保ちます。PSC2RBヒューズが0に等しい(プログラム)なら、PSCOUT20とPSCOUT21はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUT20とPSCOUT21はPSC2同期/出力構成(PSOC2)レジスタが書かれるまで、この強制された状態を保ちます。

PSC2RBAヒューズが1に等しい(非プログラム)なら、PSCOUT22とPSCOUT23は標準ポート動作を保ちます。PSC2RBAヒューズが0に等しい(プログラム)なら、PSCOUT22とPSCOUT23はリセットでPSCRVヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSCOUT22とPSCOUT23はPSC2同期/出力構成(PSOC2)レジスタが書かれるまで、この強制された状態を保ちます。

### 21.2.2. リセット中の電力段制御器(PSC)入力の動き

リセット下の電力消費の理由のために、リセット中のPSCとPSCRの入力の状態はPSCINRBヒューズによってプログラム(設定)できます。

PSCINRBヒューズが1に等しい(非プログラム)なら、PSCとPSCRの入力は標準ポート動作を保ちます。PSCINRBヒューズが0に等しい(プログラム)なら、リセットが活性の間、PSCとPSCRの入力プルアップが強制されます。影響を及ぼされるピンはPSCIN2,PSCINr,PSCIN2A,PSCINrAです。PDIでのどの衝突も防ぐために、このヒューズはPSCINrBで無効です。

表21-5. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
RSTDISBL(注1)	7	PE0がI/OピンかまたはRESETピンかを選びます。	1 (非プログラム) PE0はRESETピン
DWEN	6	デバッグWIRE機能許可。	1 (非プログラム) デバッグWIRE不許可
SPIEN (注2)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON (注3)	4	ウォッチドッグ タイマ常時有効。	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BOOTSZ1	2	ブートローダ容量選択。(表20-7.参照) (注4)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOOTRST	0	リセット領域(応用領域またはブートローダ領域)選択	1 (非プログラム) 応用領域

注1: RSTDISBLヒューズの記述については50頁の「ポートEの交換機能」をご覧ください。

注2: SPIENヒューズは直列プログラミングでアクセスできません。

注3: 詳細については37頁の「WDTCSR - ウォッチドッグ タイマ制御レジスタ」をご覧ください。

注4: BOOTSZ1,0既定値は最大ブートローダ容量になります。

表21-6. ヒューズ下位バイト一覧

名称	ビット	意味	既定値
CKDIV8 (注4)	7	システムクロック 8分周選択。	0 (プログラム) 8分周
CKOUT (注3)	6	システムクロック出力許可。	1 (非プログラム) 不許可
SUT1	5	起動時間選択。	1 (非プログラム)
SUT0	4		0 (プログラム)
CKSEL3	3	クロック種別選択。	0 (プログラム)
CKSEL2	2		0 (プログラム)
CKSEL1	1		1 (非プログラム)
CKSEL0	0		0 (プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については20頁の表5-4をご覧ください。

注2: CKSEL3~0の既定設定は8MHz校正付き内蔵RC発振器になります。詳細については19頁の表5-1をご覧ください。

注3: CKOUTはポートD0に出力することをシステムクロックに許します。詳細については22頁の「クロック出力緩衝部」をご覧ください。

注4: 詳細については25頁の「システムクロック前置分周器」をご覧ください。

ヒューズビットの状態はチップ消去によって影響されません。施錠ビット1(LB1)がプログラム(0)されると、ヒューズビットが固定されることに注意してください。施錠ビットをプログラム(0)する前にヒューズビットをプログラミング(書き込み)してください。

### 21.2.3. 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。

## 21.3. 識票バイト

全てのAtmelマイクロコントローラはデバイスを識別する3バイトの識票符号を持ちます。この符号は直列と並列の両プログラミング動作で、またデバイスが施錠されていても読めます。この3バイトは分離された空間、識票列に存在します。

AT90PWM81/161の識票バイトは以下です。

表21-7. デバイスの識別番号(ID)

部品番号	識票バイトアドレス		
	\$0000	\$0002	\$0004
AT90PWM81	\$1E	\$93	\$88
AT90PWM161	\$1E	\$94	\$8B

- ① \$0000 : \$1E 製造業者Atmelを示します。
- ② \$0002 : \$93 フラッシュメモリ容量8Kバイトを示します。  
\$94 フラッシュメモリ容量16Kバイトを示します。
- ③ \$0004 : \$88 ②値\$93と合せ、AT90PWM81を示します。  
\$8B ②値\$94と合せ、AT90PWM161を示します。

## 21.4. 校正バイト

AT90PWM81/161は内蔵RC発振器用に1バイトの校正値を持ちます。このバイトは識票アドレス空間でアドレス\$0003のバイトにあります。リセット中、校正付き内蔵RC発振器の正しい周波数を保証するために、このバイトが発振校正(OSCCAL)レジスタへ自動的に書かれます。

## 21.5. 並列プログラミング

この項はAT90PWM81/161のプログラム用フラッシュメモリ、データ用EEPROM、メモリの**施錠ビット**、**ヒューズビット**の並列プログラミングと照合の方法を記述します。特記事項を除いて、パルス幅は最低250nsと仮定されます。

### 21.5.1. 信号名

本章でAT90PWM81/161のいくつかのピンは並列プログラミング中の機能を表す信号名で参照されます。**図21-1**と**表21-8**をご覧ください。**表21-8**で示されないピンはピン名によって参照されます。

XA0とXA1ピンはXTAL1ピンが正パルスを与えられる時に実行される動作を決めます。このビット符号化は**表21-10**で示されます。

WRまたはOEパルス送出時、設定された指令が決める動作が実行されます。各種指令は**表21-11**で示されます。

図21-1. 並列プログラミング構成図

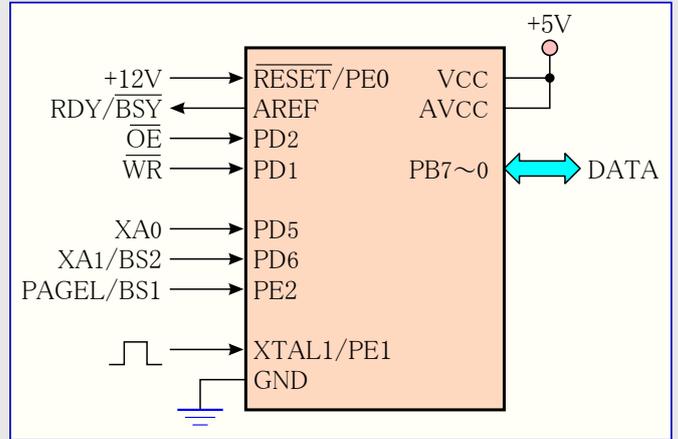


表21-8. 信号名とピン名の関係

信号名	ピン名	入出力	機能
RDY/BSY	AREF	出力	0(Low) : 多忙(プログラミング中)      1(High) : 準備可(指令受付可)
OE	PD2	入力	出力許可(負論理)
WR	PD1	入力	書き込みパルス(負論理)
XA0	PD5	入力	XTAL動作ビット0
XA1/BS2	PD6	入力	XTAL動作ビット1 上位/下位バイト選択2 (0:下位, 1:上位)      (ヒューズビット用)
PAGEL/BS1	PE2	入力	プログラムメモリとEEPROMデータをページ一時緩衝部に設定 上位/下位バイト選択1 (0:下位, 1:上位)      (一般用)
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

表21-9. プログラム動作移行時のピン値

ピン名	シンボル	値
XA1/BS2	Prog_enable[3]	0
XA0	Prog_enable[2]	0
OE	Prog_enable[1]	0
WR	Prog_enable[0]	0

表21-10. XA0とXA1の符号化(機能)

XA1	XA0	XTAL1パルス時の動作
0	0	フラッシュまたはEEPROMのアドレス設定 (上位/下位はBS1で指定)
0	1	データ設定 (フラッシュの上位/下位はBS1で指定)
1	0	指令設定
1	1	アイドル (動作なし)

表21-11. 指令バイトのビット符号化

指令バイト	指令の機能
\$80 (1000 0000)	チップ消去
\$40 (0100 0000)	ヒューズビット書き込み
\$20 (0010 0000)	施錠ビット書き込み
\$10 (0001 0000)	フラッシュメモリ書き込み
\$11 (0001 0001)	EEPROM書き込み
\$08 (0000 1000)	識票バイト、校正バイト読み出し
\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$02 (0000 0010)	フラッシュメモリ読み出し
\$03 (0000 0011)	EEPROM読み出し

表21-12. 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
AT90PWM81	4K語 (8Kバイト)	32語	PC4~0	128	PC11~5	11
AT90PWM161	8K語 (16Kバイト)	64語	PC5~0	128	PC12~6	12

表21-13. 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
AT90PWM81,AT90PWM161	512バイト	4バイト	EEA1,0	128	EEA8~2	8

(訳注) 原書本位置の表21-14.は共通性のため「直列プログラミング」項へ移動しました。

## 21.6. 並列プログラミング手順

### 21.6.1. 並列プログラミング動作への移行

次に示す方法がデバイスを(高電圧)並列プログラミング動作にします。

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、表21-9.で一覧されるProg.enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5~5.5Vを印加します。それから20 $\mu$ s以内にVCCが最低1.8Vに達することを保証してください。
- ③ 20~60 $\mu$ s待ち、 $\overline{\text{RESET}}$ に11.5~12.5Vを印加します。
- ④ Prog.enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu$ s、Prog.enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立って少なくとも300 $\mu$ s間待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持つてくることによってプログラミング動作を抜けます。

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

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、表21-9.で一覧されるProg.enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5~5.5Vを印加します。
- ③ VCCを監視し、0.9~1.1Vに達したら直ぐ、 $\overline{\text{REEST}}$ に11.5~12.5Vを印加します。
- ④ Prog.enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu$ s、Prog.enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立ってVCCが実際に4.5~5.5Vに達するまで待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持つてくることによってプログラミング動作を抜けます。

### 21.6.2. 効率的なプログラミングへの考慮

設定した指令とアドレスはプログラミング中、維持されます。効率的なプログラミングを行うために次を考慮すべきです。

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

### 21.6.3. チップ消去

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

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

「チップ消去」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$80(1000 0000)に設定します。これはチップ消去指令です。
- ④ XTAL1に正パルスを与えます。これはチップ消去指令を設定します。
- ⑤  $\overline{\text{WR}}$ に負パルスを与えます。これはチップ消去を開始します。RDY/BSYがLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/BSYがHigh(1)になるまで待ちます。

#### 21.6.4. フラッシュ メモリ書き込み (次頁の図21-3. タイミングを参照)

フラッシュ メモリはページで構成されます(153頁の表21-12. 参照)。フラッシュ メモリへ書く時にプログラム データはページ緩衝部にラッチされます。これは同時に書かれることをプログラム データの1ページに許します。次の手順は完全なフラッシュ メモリの書き込み方法を記述します。

##### A. 「フラッシュ メモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュ メモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュ メモリ書き込み指令を設定します。

##### B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選びます。
- ③ DATAにアドレス下位バイト(\$00~\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス下位バイトを設定します。

##### C. データ下位バイト設定

- ① XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ② DATAにデータ下位バイト(\$00~\$FF)を設定します。
- ③ XTAL1に正パルスを与えます。これはデータ下位バイトを設定します。

##### D. データ上位バイト設定

- ① BS1をHigh(1)に設定します。これは上位バイトを選びます。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00~\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはデータ上位バイトを設定します。

##### E. 語(ワード)データをページ一時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイトを選びます。
- ② PAGELに正パルスを与えます。これは語データをページ一時緩衝部にラッチ(設定)します。

##### F. 緩衝部全体が満たされるか、または(必要な)全てのデータが緩衝部内に設定されるまで、B~Eを繰り返す

アドレス内の下位ビットがページ内の語位置を指示する一方、上位ビットがフラッシュ メモリ内のページをアドレス指定します。これは図21-2. で図示されます。ページ内の語アドレスに8ビット未満が必要とされる場合(ページ容量<256)、アドレス下位バイトの最上位(側)ビットがページ書き込み実行時のページ アドレスに使われることに注意してください。

##### G. アドレス上位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選びます。
- ③ DATAにアドレス上位バイト(\$00~\$0F/\$1F)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイトを設定します。

##### H. ページ書き込み

- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/ $\overline{BSY}$ がLow(0)になります。
- ③ RDY/ $\overline{BSY}$ がHigh(1)になるまで待ちます。

##### I. フラッシュ メモリ全部が書かれるか、または(必要な)全データが書かれてしまうまで、B~Hを繰り返す

##### J. ページ書き込み終了

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図21-2. ページで構成されたフラッシュ メモリのアドレス指定

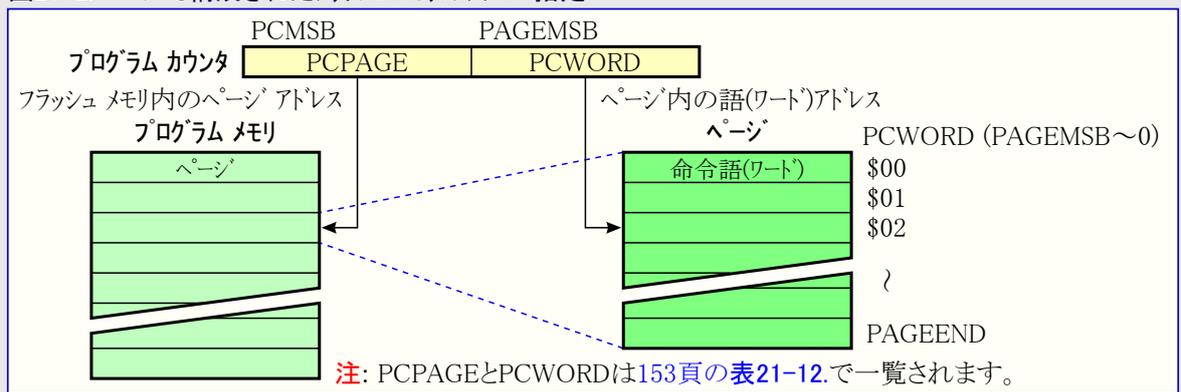
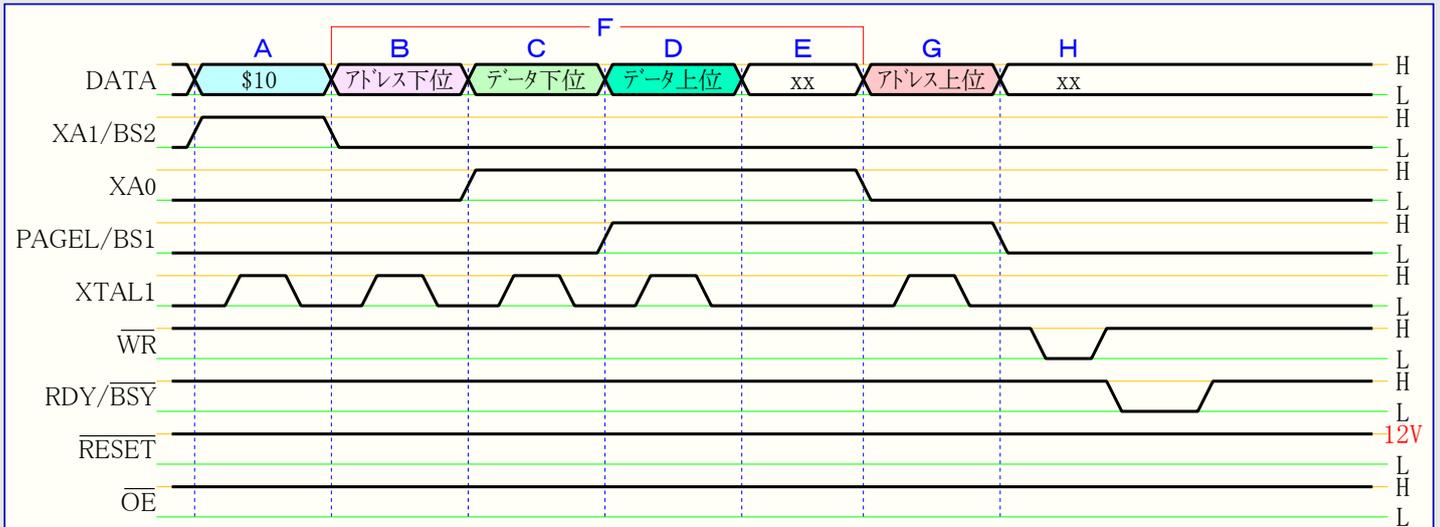


図21-3. フラッシュメモリ書き込みタイミング



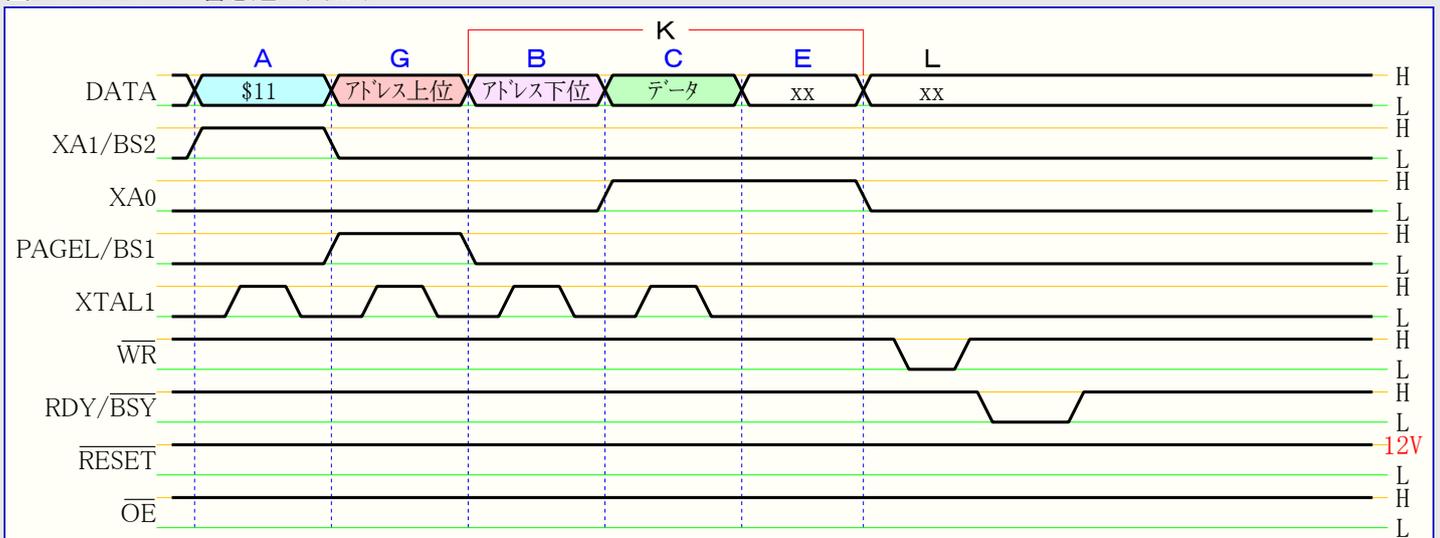
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

### 21.6.5. EEPROM書き込み

EEPROMはページで構成されます(153頁の表21-13参照)。EEPROMを書く時にデータはページ緩衝部にラッチされます。これは同時に書かれることをデータの1ページに許します。データ用EEPROMメモリの書き込み方法は次のとおりです。(指令、アドレス、データ設定の詳細については154頁の「フラッシュメモリの書き込み」を参照。図21-4. タイミング参照。)

1. EEPROM書き込み指令\$11(0001 0001)を設定します。(「フラッシュメモリ書き込み」のAを参照)
  2. アドレス上位バイト(\$00～\$01)を設定します。(「フラッシュメモリ書き込み」のGを参照)
  3. アドレス下位バイト(\$00～\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
  4. データバイト(\$00～\$FF)を設定します。(「フラッシュメモリ書き込み」のCを参照)
  5. データをラッチします(PAGELに正パルスを与えます)。(「フラッシュメモリ書き込み」のEを参照)
- K. 緩衝部全体が満たされるまで3～5を繰り返します。
- L. EEPROMページ書き込み
- ① BS1をLow(0)に設定します。
  - ② WRに負パルスを与えます。これはEEPROMページ書き込みを開始します。RDY/BSYがLow(0)になります。
  - ③ 次のページを書く前に、RDY/BSYがHigh(1)になるまで待ちます。

図21-4. EEPROM書き込みタイミング



### 21.6.6. フラッシュ メリ読み出し

フラッシュ メリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については154頁の「フラッシュ メリの書き込み」参照。)

1. フラッシュ メリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$0F/\$1F)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。フラッシュ メリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュ メリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. OEをHigh(1)に設定します。DATAはHi-Zになります。

### 21.6.7. EEPROM読み出し

データ用EEPROMの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については154頁の「フラッシュ メリの書き込み」参照。)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$01)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。EEPROMのバイト データが直ぐにDATAで読めます。
5. OEをHigh(1)に設定します。DATAはHi-Zになります。

### 21.6.8~10. ヒューズ ビット書き込み (訳注:原書での拡張/上位/下位ヒューズ項を1つに纏めました。)

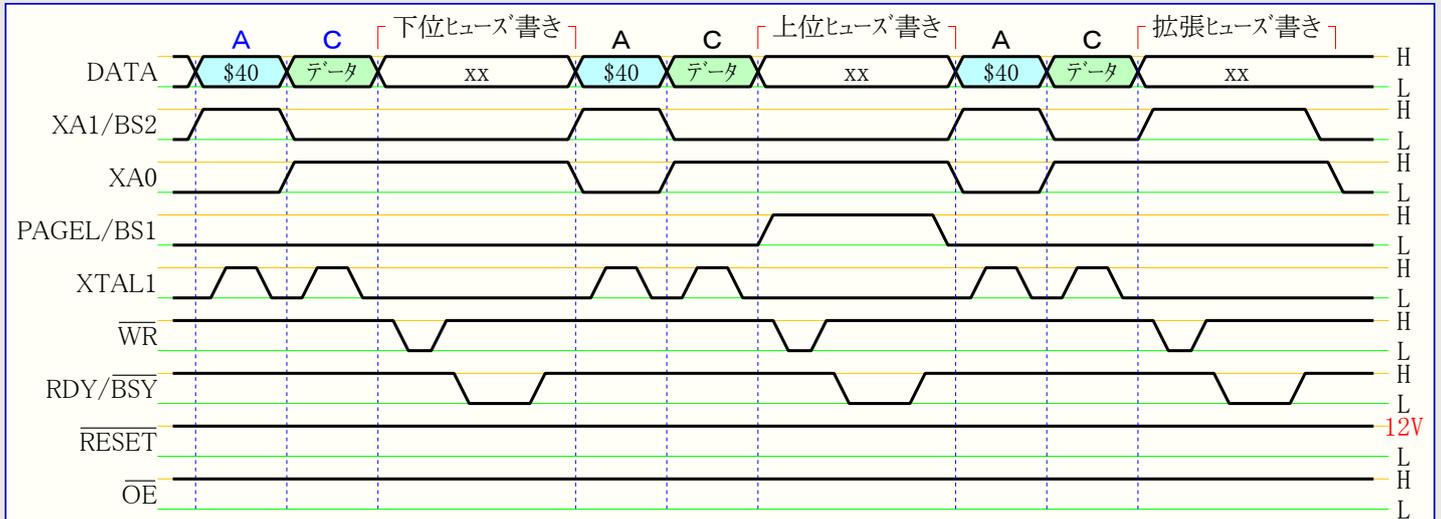
各ヒューズ バイトの書き込み方法は次のとおりです。(指令とデータ設定の詳細については154頁の「フラッシュ メリの書き込み」参照。)

1. ヒューズ ビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュ メリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4. WRに負パルスを与え、RDY/BSYがHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選びます。

表A. ヒューズ バイト対応BS1,BS2設定

ヒューズ バイト	BS1	BS2
拡張バイト	Low(0)	High(1)
上位バイト	High(1)	Low(0)
下位バイト	Low(0)	Low(0)

図21-5. ヒューズ書き込みタイミング



### 21.6.11. 施錠ビット書き込み

施錠ビットの書き込み方法は次のとおりです。(指令とデータ設定の詳細については154頁の「フラッシュメモリの書き込み」参照。)

1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. データ下位バイト(としてデータ)を設定します。0=プログラム,1=無変化です。LB保護種別3が設定(LB1とLB2がプログラム(0)されると、どの外部的なプログラミング動作種別によってもポート施錠ビットはプログラミングできません。(「フラッシュメモリの書き込み」のCを参照)
3.  $\overline{WR}$ に負パルスを与え、RDY/BSYがHighになるまで待ちます。

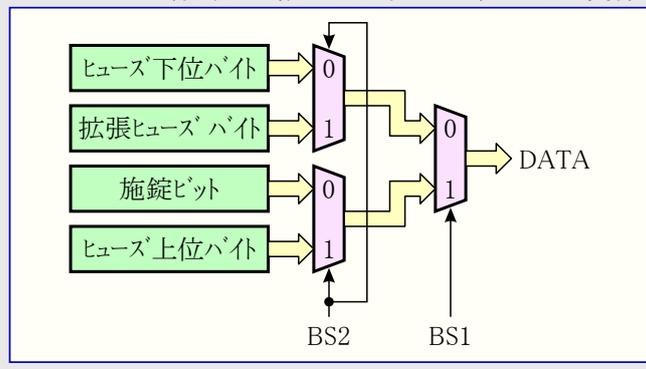
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

### 21.6.12. ヒューズビットと施錠ビットの読み出し

ヒューズビットと施錠ビットの読み出し方法は次のとおりです。(指令設定の詳細については154頁の「フラッシュメモリの書き込み」参照。)

1. ヒューズビットと施錠ビットの読み出し指令\$04(0000 0100)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. BS1とBS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ下位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
3. BS1とBS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ上位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
4. BS1をLow(0)、BS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。拡張ヒューズビットの状態が直ぐにDATAで読めます。(0=プログラム)
5. BS1をhigh(1)、BS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。施錠ビットの状態が直ぐにDATAで読めます。(0=プログラム)
6.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

図21-6. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



### 21.6.13. 識票バイト読み出し

識票バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については154頁の「フラッシュメモリの書き込み」参照。)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。選んだ識票バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

### 21.6.14. 校正バイト読み出し

校正バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については154頁の「フラッシュメモリの書き込み」参照。)

1. 校正バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイトに\$00を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。校正バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

## 21.7. 直列プログラミング

フラッシュメモリとEEPROMの両方は $\overline{\text{RESET}}$ がGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。 $\overline{\text{RESET}}$ がLowに設定された後、書き込み/消去操作が実行され得るのに先立って**プログラミング許可命令**が初めに実行されることを必要とします。**注意**、表21-14.でSPIプログラミング用のピン配置が一覧されます。全てのデバイスが内部SPIインターフェースに対する専用SPIピンを使うとは限りません。

### 21.7.1. 直列プログラミング用ピン配置

表21-14. 直列プログラミング用ピン配置

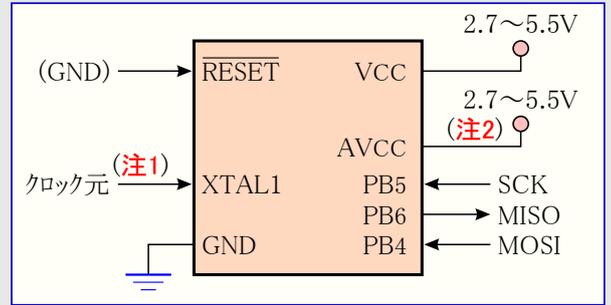
信号名	ピン名	入出力	機能
MOSI	PB4	入力	直列データ入力
MISO	PB6	出力	直列データ出力
SCK	PB5	入力	直列クロック

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(直列プログラミングのみ)、**チップ消去命令**を初めに実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全てのメモリ位置の内容を\$FFにします。

CKSELヒューズによる有効なクロックが与えられなければなりません。直列クロック(SCK)入力のLowとHighの区間の最小値は次のように定義されます。

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

図21-7. 直列プログラミング構成図



**注1:** デバイスが内蔵発振器で動作する場合、XTAL1ピンにクロック元を接続する必要はありません。

**注2:**  $V_{\text{CC}} - 0.3\text{V} < AV_{\text{CC}} < V_{\text{CC}} + 0.3\text{V}$ ですが、AVCCは常に2.7~5.5V内にすべきです。

## 21.8. 直列プログラミング手順

AT90PWM81/161へ直列データを書く時にはデータがSCKの上昇端で行われ、AT90PWM81/161から読む時にはデータがSCKの下降端で行われます。タイミングの詳細については図21-8.をご覧ください。

直列プログラミング動作でのAT90PWM81/161のプログラミングと照合は次手順が推奨されます(4バイト命令形式は表21-16.参照)。

### 1. 電源投入手順:

$\overline{\text{RESET}}$ とSCKがLow(0)に設定されている間にVCCとGND間へ電源を印加します。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、 $\overline{\text{RESET}}$ 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バイト送出時に第2バイト(\$53)が戻ります。この戻りが正しいかどうかによらず、命令の4バイト全てが送信されなければなりません。\$53が戻らない場合、 $\overline{\text{RESET}}$ に正パルスを与え、新規プログラミング許可命令を行ってください。

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153頁の表21-12.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5/6+1ビットとデータを供給することによって1バイトづつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。プログラムメモリのページはアドレスの上位7ビットを含む**ページ書き込み命令**の設定によって(フラッシュメモリに)格納されます。ポーリング(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低tWD\_FLASH(表21-15.参照)待たなければなりません。フラッシュ書き込み操作完了前の(ポーリング以外の)直列プログラミングインターフェースでのアクセスは不正な書き込み結果になり得ます。

5. EEPROMはページ単位かバイト単位のどちらかでプログラミングできます。(訳注:本項は表21-16.に合せて差し替えてあります。)

**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21-15.参照)待たなければなりません。

**ページ単位:** EEPROMの1ページは**EEPROMページ設定命令**と共にアドレスの下位2ビットとデータを供給することによって1バイトづつ設定されます。EEPROMページはアドレスの上位7ビットを含む**EEPROMページ書き込み命令**によって(EEPROMに)格納されます。EEPROMページアクセス使用時、EEPROMページ設定命令で設定したバイト位置だけが変更されます。残りの位置は無変化で留まります。ポーリング(BSY/RDY)が使われない場合、次ページ(表21-13.参照)を行う前に最低tWD\_EEPROM(表21-15.参照)待たなければなりません。チップ消去されたデバイスでの\$FFデータは書かれる必要がありません。

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

7. プログラミング作業終了時、 $\overline{\text{RESET}}$ 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)

- $\overline{\text{RESET}}$ をHigh(1)に設定します。
- VCC電源をOFFにします。

**(訳注)**

原書本位置の「フラッシュデータホーリング」と「EEPROMデータホーリング」は書き込み手順を修正したために不要となりましたので他のデータシート同様に削除しました。

表21-15. 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	3.6ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

**(訳注)**

共通性からtWD\_FUSE項目を追加

図21-8. 直列プログラミングバイト通信波形

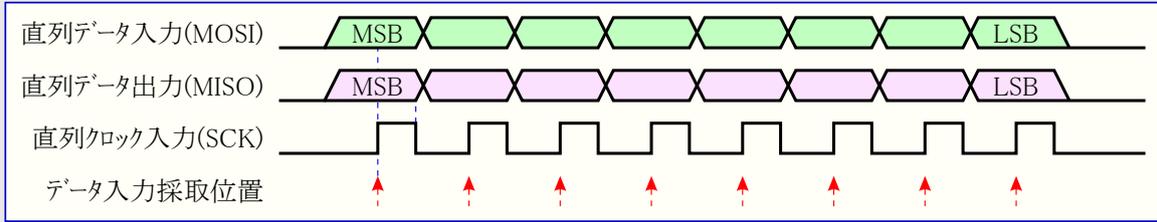


表21-16. 直列プログラミング命令一式

命令	命令形式				動作
	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	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	RESET=Low後のプログラミングを許可します。
チップ消去	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	フラッシュメモリ,EEPROM,施錠ビットを消去します。
フラッシュメモリ読み出し	0010 P000	0000 HHHH	LLLL LLLL	RRRR RRRR	アドレス:LのP(H/L)バイトを読み出します。
フラッシュページ設定	0100 P000	00xx xxxx	xxxL LLLL	WWW WWW	緩衝部アドレスLのP(H/L)バイトに書き込みます。
フラッシュページ書き込み	0100 1100	0000 HHHH	LLLx xxxx	xxxx xxxx	アドレス:H:Lのページに書き込みます。
EEPROM読み出し	1010 0000	000x xxxH	LLLL LLLL	RRRR RRRR	アドレス:H:Lのバイトを読み出します。
EEPROMバイト書き込み	1100 0000	000x xxxH	LLLL LLLL	WWW WWW	アドレス:H:Lのバイトに書き込みます。
EEPROMページ設定	1100 0001	0000 0000	0000 00LL	RRRR RRRR	緩衝部アドレスLのバイトに書き込みます。
EEPROMページ書き込み	1100 0010	00xx xxxH	LLLL LL00	xxxx xxxx	アドレス:H:Lのページに書き込みます。
施錠ビット読み出し	0101 1000	0000 0000	xxxx xxxx	xxRR RRRR	施錠ビットを読み出します。
施錠ビット書き込み	1010 1100	111x xxxx	xxxx xxxx	11WW WWW	(149頁の表21-1.参照)に書き込みます。
ヒューズ下位読み出し	0101 0000	0000 0000	xxxx xxxx	RRRR RRRR	ヒューズ下位ビットを読み出します。
ヒューズ下位書き込み	1010 1100	1010 0000	xxxx xxxx	WWW WWW	(151頁の表21-6.参照)に書き込みます。
ヒューズ上位読み出し	0101 1000	0000 1000	xxxx xxxx	RRRR RRRR	ヒューズ上位ビットを読み出します。
ヒューズ上位書き込み	1010 1100	1010 1000	xxxx xxxx	WWW WWW	(150頁の表21-5.参照)に書き込みます。
拡張ヒューズ読み出し	0101 0000	0000 1000	xxxx xxxx	RRRR xRRR	拡張ヒューズビットを読み出します。
拡張ヒューズ書き込み	1010 1100	1010 0100	xxxx xxxx	WWW xWWW	(150頁の表21-4.参照)に書き込みます。
識票バイト読み出し	0011 0000	000x xxxx	xxxx xxLL	RRRR RRRR	アドレスLの識票バイトを読み出します。
校正バイト読み出し	0011 1000	000x xxxx	0000 0000	RRRR RRRR	校正バイトを読み出します。
多忙/準備可検査	1111 0000	0000 0000	xxxx xxxx	xxxx xxxR	R=1で多忙、他の操作前に0まで待機します。

H = アドレス上位バイトのビット      L = アドレス下位バイトのビット      P = 0=下位バイト, 1=上位バイト  
 R = 読み出しデータ (MCU出力)      W = 書き込みデータ (MCU入力)      x = 0または1 (無意味/不定)

**21.8.1. 直列プログラミング特性**

SPI部の特性については164頁の「SPIタイミング特性」を参照してください。



## 22. 電気的特性

(重要注意) 本製品についての電気的特性は未だ終了されてしまっておりません。ここで一覧した全ての値を非契約の暫定とみなしてください。

### 22.1. 絶対最大定格 (警告)

動作温度	-40°C ~ +105/+125°C
保存温度	-65°C ~ +150°C
RESETを除くピン許容電圧	-1.0V ~ VCC+0.5V
RESETピン許容電圧	-1.0V ~ +13.0V
最大動作電圧	6.0V
入出力ピン出力電流	40.0mA
消費電流	200.0mA

(警告)

絶対最大定格を超える負担はデバイスに定常的な損傷を与えます。絶対最大定格は負担の定格を示すためだけのもので、この値または、この仕様書の動作特性で示された値を超える条件で動作することを示すものではありません。長時間の最大定格での使用はデバイスの信頼性を損なう場合があります。

### 22.2. DC特性

TA=-40°C~105°C, VCC=2.7V~5.5V (特記事項を除く)

シンボル	項目		条件	最小	代表	最大	単位
V <sub>IL</sub>	Lowレベル入力電圧 (一般/XTALポート)		ポートB,DとI/O使用の XTAL1,XTAL2	-0.5		0.2VCC (注1)	V
V <sub>IL1</sub>	Lowレベル入力電圧 (XTAL1)		外部クロック信号選択	-0.5		0.1VCC (注1)	
V <sub>IL2</sub>	Lowレベル入力電圧 (RESET)		外部リセット入力	-0.5		0.2VCC (注1)	
V <sub>IL3</sub>	Lowレベル入力電圧 ( $\overline{\text{RESET}}$ )		I/Oとして使用	-0.5		0.2VCC (注1)	
V <sub>IH</sub>	Highレベル入力電圧 (一般/XTALポート)		ポートB,DとI/O使用の XTAL1,XTAL2	0.6VCC (注2)		VCC+0.5	
V <sub>IH1</sub>	Highレベル入力電圧 (XTAL1)		外部クロック信号選択	0.7VCC (注2)		VCC+0.5	
V <sub>IH2</sub>	Highレベル入力電圧 (RESET)		外部リセット入力	0.9VCC (注2)		VCC+0.5	
V <sub>IH3</sub>	Highレベル入力電圧 ( $\overline{\text{RESET}}$ )		I/Oとして使用	0.8VCC (注2)		VCC+0.5	
V <sub>OL</sub>	Lowレベル出力電圧 (注3)	I/Oとしての RESET以外	IOL=10mA, VCC=5V			0.6	
			IOL=5mA, VCC=3V			0.5	
V <sub>OL3</sub>		$\overline{\text{RESET}}$ =I/O	IOL=2.1mA, VCC=5V			0.7	
			IOL=0.8mA, VCC=3V			0.5	
V <sub>OH</sub>	Highレベル出力電圧 (注4)	I/Oとしての RESET以外	IOH=-10mA, VCC=5V	4.3			
			IOH=-5mA, VCC=3V	2.5			
V <sub>OH3</sub>		$\overline{\text{RESET}}$ =I/O	IOH=-0.6mA, VCC=5V	3.8			
			IOH=-0.4mA, VCC=3V	2.2			
I <sub>IL</sub>	I/OピンLowレベル入力漏れ電流		VCC=5.5V			1	μA
I <sub>IH</sub>	I/OピンHighレベル入力漏れ電流		確実なH/L範囲			1	
RRST	RESETピンプルアップ抵抗			30		200	kΩ
RPU	I/Oピンプルアップ抵抗			20		50	
I <sub>CC</sub>	活動動作消費電流	PRR=\$FF	VCC=3V, 8MHz(RC)		3.5	5	mA
			VCC=5V, 16MHz(外部)		10.5	15	
	アイドル動作消費電流		VCC=3V, 8MHz(RC)		1.5	2	
			VCC=5V, 16MHz(外部)		4.5	7	
	パワーダウン動作消費電流 (注5)	WDT有効	VCC=3V	TA=25°C		7	
				TA=105°C			30
			VCC=5V	TA=25°C		10	
				TA=105°C			50
WDT禁止		VCC=3V	TA=25°C		0.5		
			TA=105°C			25	
VCC=5V	TA=25°C		1				
	TA=105°C			40			
V <sub>REF</sub>	内部基準電圧 (注7)		TA=25°C	2.46	2.56	2.66	V
アナログ比較器同相入力範囲			0.1		VCC-0.1		

次頁へ続く

(続き) TA=-40°C~105°C, VCC=2.7V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V <sub>ACIO</sub>	アナログ比較器 入力変位(オフセット) 電圧	0.1V<V <sub>in</sub> <VCC-0.1V	入力変位電圧	±1.5	±10	mV
			±10mVヒステリシスで	±10	±20	
			±50mVヒステリシスで	±25	±60	
I <sub>ACLK</sub>	アナログ比較器入力漏れ電流	VCC=5V, V <sub>in</sub> =VCC/2	-50		50	nA
t <sub>ACPD</sub>	アナログ比較器伝播遅延時間	VCC=2.7V, VCC=5V		50 (注6)		ns

TA=-40°C~125°C, VCC=2.7V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位	
V <sub>IL</sub>	Lowレベル入力電圧 (一般/XTALポート)	ポートB,DとI/O使用の XTAL1,XTAL2	-0.5		0.2VCC (注1)	V	
V <sub>IL1</sub>	Lowレベル入力電圧 (XTAL1)	外部クロック信号選択	-0.5		0.1VCC (注1)		
V <sub>IL2</sub>	Lowレベル入力電圧 (RESET)	外部リセット入力	-0.5		0.2VCC (注1)		
V <sub>IL3</sub>	Lowレベル入力電圧 (RESET)	I/Oとして使用	-0.5		0.2VCC (注1)		
V <sub>IH</sub>	Highレベル入力電圧 (一般/XTALポート)	ポートB,DとI/O使用の XTAL1,XTAL2	0.6VCC (注2)		VCC+0.5		
V <sub>IH1</sub>	Highレベル入力電圧 (XTAL1)	外部クロック信号選択	0.7VCC (注2)		VCC+0.5		
V <sub>IH2</sub>	Highレベル入力電圧 (RESET)	外部リセット入力	0.9VCC (注2)		VCC+0.5		
V <sub>IH3</sub>	Highレベル入力電圧 (RESET)	I/Oとして使用	0.8VCC (注2)		VCC+0.5		
V <sub>OL</sub>	Lowレベル出力電圧 (注3)	I/Oとしての RESET以外	IOL=10mA, VCC=5V		0.6		
			IOL=5mA, VCC=3V		0.5		
V <sub>OL3</sub>		RESET=I/O	IOL=2.1mA, VCC=5V		0.7		
			IOL=0.8mA, VCC=3V		0.5		
V <sub>OH</sub>	Highレベル出力電圧 (注4)	I/Oとしての RESET以外	IOH=-10mA, VCC=5V	4.3			
				IOH=-5mA, VCC=3V	2.5		
V <sub>OH3</sub>		RESET=I/O	IOH=-0.6mA, VCC=5V	3.8			
			IOH=-0.4mA, VCC=3V	2.2			
I <sub>IL</sub>	I/OピンLowレベル入力漏れ電流	VCC=5.5V			1	μA	
I <sub>IH</sub>	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			1		
RRST	RESETピンプルアップ抵抗		30		200	kΩ	
RPU	I/Oピンプルアップ抵抗		20		50		
I <sub>CC</sub>	活動動作消費電流	PRR=\$FF	VCC=3V, 8MHz(RC)	3.5	5	mA	
			VCC=5V, 16MHz(外部)	10.5	15		
	アイドル動作消費電流		VCC=3V, 8MHz(RC)	1.5	2		
			VCC=5V, 16MHz(外部)	4.5	7		
	パワーダウン動作消費電流 (注5)	WDT有効	VCC=3V	TA=25°C	7		μA
				TA=125°C		70	
			VCC=5V	TA=25°C	10		
				TA=125°C		110	
WDT禁止		VCC=3V	TA=25°C	0.5			
			TA=125°C		35		
	VCC=5V	TA=25°C	1				
TA=125°C			55				
V <sub>REF</sub>	内部基準電圧 (注7)	TA=25°C	2.46	2.56	2.66	V	
	アナログ比較器同相入力範囲		0.1		VCC-0.1		
V <sub>ACIO</sub>	アナログ比較器 入力変位(オフセット) 電圧	0.1V<V <sub>in</sub> <VCC-0.1V	入力変位電圧	±1.5	±10	mV	
			±10mVヒステリシスで	±10	±20		
			±50mVヒステリシスで	±25	±60		
I <sub>ACLK</sub>	アナログ比較器入力漏れ電流	VCC=5V, V <sub>in</sub> =VCC/2	-50		50	nA	
t <sub>ACPD</sub>	アナログ比較器伝播遅延時間	VCC=2.7V, VCC=5V		50 (注6)		ns	

(注1)~(注7)は次頁を参照してください。



**注1:** Lowレベルの認識が保証される最高電圧です。

**注2:** Highレベルの認識が保証される最低電圧です。

**注3:** 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3Vで5mA, VCC=5Vで10mA)よりも多くの吸い込み電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全IOLの合計が400mAを超えるべきではありません。
2. ポートB7,6、D3~0、E0のIOLの合計が100mAを超えるべきではありません。
3. ポートB1,0、D4、E2,1のIOLの合計が100mAを超えるべきではありません。

IOLが検査条件を超える場合、VOLも仕様書での値を超えます。表の検査条件よりも大きな吸い込み電流を流すことは保証されません。

**注4:** 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3Vで5mA, VCC=5Vで10mA)よりも多くの吐き出し電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全IOHの合計が400mAを超えるべきではありません。
2. ポートB7,6、D3~0、E0のIOHの合計が150mAを超えるべきではありません。
3. ポートB1,0、D4、E2,1のIOHの合計が150mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件よりも大きな吐き出し電流を流すことは保証されません。

**注5:** パワーダウン動作に対する最小VCCは2.5Vです。

**注6:** 100mV過駆動状態での内部比較器の伝播遅延。

**注7:** 精度:  $\pm 8\%$  ( $-40\sim 125^{\circ}\text{C}$ )

## 22.3. クロック特性

### 22.3.1. 校正付き内蔵RC発振器の精度

表22-1. 校正付き内蔵RC発振器の校正精度

校正種別	周波数	VCC	温度	校正精度
工場校正	8.0MHz	3V	25°C	$\pm 1\%$
		2.7~5.5V	-40~105/125°C	$\pm 6\%$
使用者校正	7.6~8.4MHz	2.7~5.5V	-40~105/125°C	$\pm 5\%$

### 22.3.2. ウォッチドッグ発振器精度

表22-2. ウォッチドッグ発振器の精度

周波数	精度
128kHz	$\pm 40\%$

### 22.3.3. 外部クロック信号駆動

図23-1. 外部クロック駆動波形

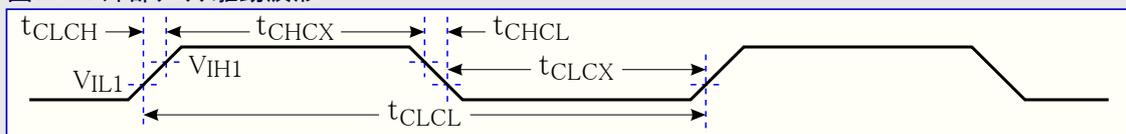


表22-3. 外部クロック特性

シンボル	項目	VCC=2.7~5.5V		VCC=4.5~5.5V		単位
		最小	最大	最小	最大	
$1/t_{CLCL}$	クロック周波数	0	12	0	16	MHz
$t_{CLCL}$	クロック周期	83		62		ns
$t_{CHCX}$	Highレベル時間	30		20		
$t_{CLCX}$	Lowレベル時間	30		20		
$t_{CLCH}$	上昇時間		1.6		0.5	$\mu\text{s}$
$t_{CHCL}$	下降時間		1.6		0.5	
$\Delta t_{CLCL}$	隣接クロック周期間の変化率		2		2	%

**注:** 詳細については21頁の「外部クロック信号」を参照してください。

## 22.4. 最高速と動作電圧

最高周波数は動作電圧に依存します。図22-2.で示されるように最高周波数はVCCが2.7～4.5V間に含まれる時に12MHzで、VCCが4.5～5.5V間に含まれる時に16MHzです。



## 22.5. PLL特性

表22-4. PLL特性 VCC=2.7V～5.5V (特記事項を除く)

シンボル	項目	最小	代表	最大	単位
PLLIF	入力周波数 (注1)		8		MHz
PLLF	PLL係数(倍率)	4		8 (注2)	
PLLLT	固定化時間			64	μs

**注1:** 外部クロック信号または外部発振器接続時、PLL入力周波数は回路の駆動部分の駆動に対応する周波数で出力を提供するように選ばなければなりません。

**注2:** VCCが4.5V以下の時は最大PLLFが6です。

## 22.6. SPIタイミング特性

図22-3. SPI タイミング必要条件 (主装置動作)

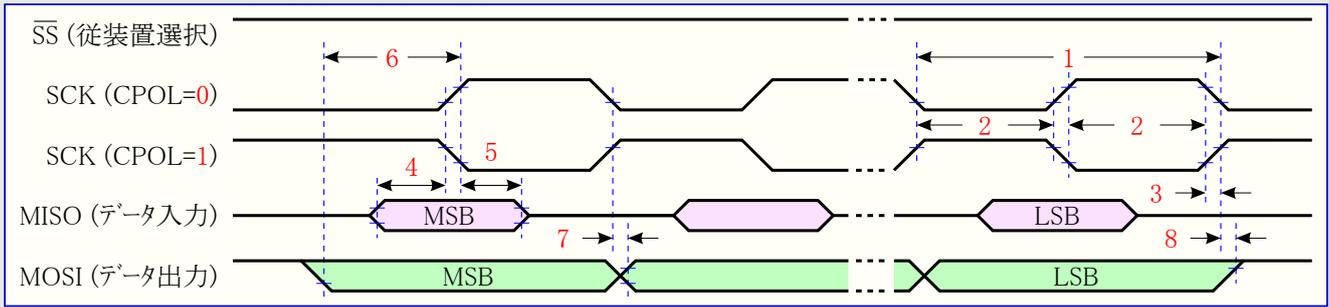


図22-4. SPI タイミング必要条件 (従装置動作)

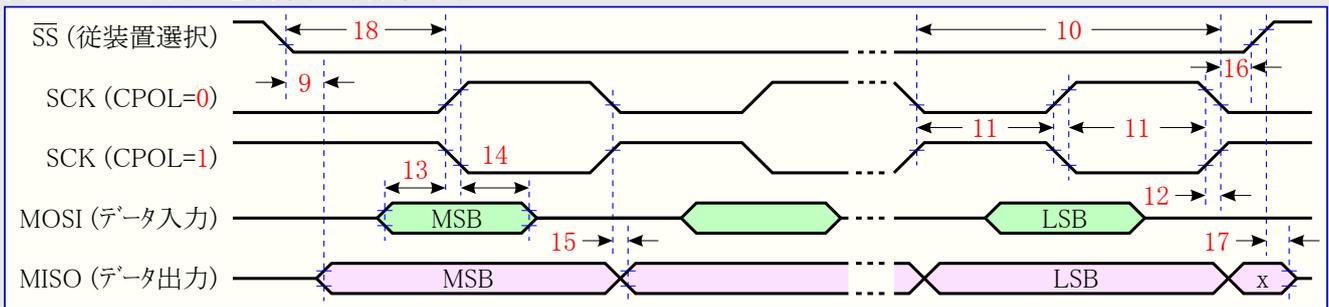


表22-5. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表14-5.参照		ns
2	SCK High/Low期間	主装置		50%デューティ比		
3	SCK上昇/下降時間	主装置		3.6		
4	入力データ 準備時間	主装置		10		
5	入力データ 保持時間	主装置		10		
6	出力からSCK変移時間	主装置		$0.5t_{SCK}$		
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	$\overline{SS}$ ↓からの出力遅延時間	従装置		15		μs
10	SCK周期	従装置	$4t_{CK}$			
11	SCK High/Low期間 (注)	従装置	$2t_{CK}$			ns
12	SCK上昇/下降時間	従装置			16	
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	$t_{CK}$			
15	SCKからの出力遅延時間	従装置		15		
16	SCKからの $\overline{SS}$ ↑遅延時間	従装置	20			
17	$\overline{SS}$ ↑からの出力Hi-Z遅延時間	従装置		10		
18	$\overline{SS}$ ↓からのSCK遅延時間	従装置	$2t_{CK}$			

注: SPIプログラミングでの最小SCK High/Low期間は、 $2t_{CLCL}$  ( $t_{CK} < 12\text{MHz}$ )、 $3t_{CLCL}$  ( $t_{CK} \geq 12\text{MHz}$ )です。

22.7. A/D変換器特性

表22-6. A/D変換特性 (TA=-40~+105°C, VCC=2.7~5.5V) (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位	
シングル エンド 入力 変換	分解能			10		ビット	
	絶対精度	VCC=4V, VREF=4V, 変換クロック=1MHz		2	4	LSB	
		VCC=2.7V, VREF=2.56V, 変換クロック=2MHz		2.2	4		
	積分非直線性 誤差 (INL)	VCC=4V, VREF=4V, 変換クロック=1MHz		0.5	1		
		VCC=4V, VREF=4V, 変換クロック=2MHz		0.7	1.5		
		VCC=2.7V, VREF=2.56V, 変換クロック=2MHz		1.0	2.5		
	微分非直線性 誤差 (DNL)	VCC=4V, VREF=4V, 変換クロック=1MHz		0.2	0.5		
		VCC=4V, VREF=4V, 変換クロック=2MHz		0.6	1		
		VCC=2.7V, VREF=2.56V, 変換クロック=2MHz		1.0	2.5		
	利得誤差	VCC=4V, VREF=4V, 変換クロック=1MHz		0.0	-6.0		
VCC=2.7V, VREF=2.56V, 変換クロック=2MHz			0.0	-6.0			
変位(オフセット) (ゼロ)誤差	VCC=4V, VREF=4V, 変換クロック=1MHz		-1.0	2.0			
	VCC=2.7V, VREF=2.56V, 変換クロック=2MHz		1.0	4.0			
VIN	入力電圧		GND		VREF	V	
	入力周波数帯域			38.5		kHz	
差動 入力 変換	分解能	×5,10利得		8		ビット	
		×20,40利得		8			
	絶対精度	VCC=5V, VREF=4V	×5,10利得, 変換クロック=1MHz		1.2	2.0	LSB
			×20,40利得, 変換クロック=2MHz		1.5	3.0	
	×5,10利得, 変換クロック=1MHz			0.5	1.0		
	×20,40利得, 変換クロック=2MHz			0.8	2.0		
	×5,10利得, 変換クロック=1MHz			0.3	0.8		
	×20,40利得, 変換クロック=2MHz			0.5	1.0		
利得誤差	変換クロック=1MHz		-2.0	+2.0			
変位(オフセット)誤差	変換クロック=1MHz		-1.0	+1.0			
VIN	入力電圧		-VREF/利得		+VREF/利得	V	
	入力周波数帯域			4		kHz	
共通	変換時間	単独変換動作	8		260	μs	
	変換クロック周波数		50		2000	kHz	
	AVCC	アナログ供給電圧	VCC-0.3		VCC+0.3	V	
	VREF	基準電圧	2.56		AVCC-0.6		
	RREF	基準電圧入力インピーダンス		30		kΩ	
	RAIN	アナログ入力インピーダンス		23			
	CAIN	アナログ入力容量		10		pF	
	IHSM	高速動作増加消費電流	高速動作形態, 単独変換動作			380	μA

22.8. D/A変換器特性

表22-7. D/A変換特性 (TA=-40~+105°C, VCC=2.7~5.5V) (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
	分解能			10		ビット
	絶対精度	VCC=4V, VREF=4V		2.5	5	LSB
	積分非直線性誤差(INL)			0.8	1.5	
	微分非直線性誤差(DNL)			0.2	0.5	
	利得誤差			-5.0	0.0	
	変位(オフセット)(ゼロ)誤差			0.0	2.0	
VREF	基準電圧		2.56		AVCC	V



## 22.9. 並列プログラミング特性

図22-5. 並列プログラミング タイミング (一般的な必要条件)

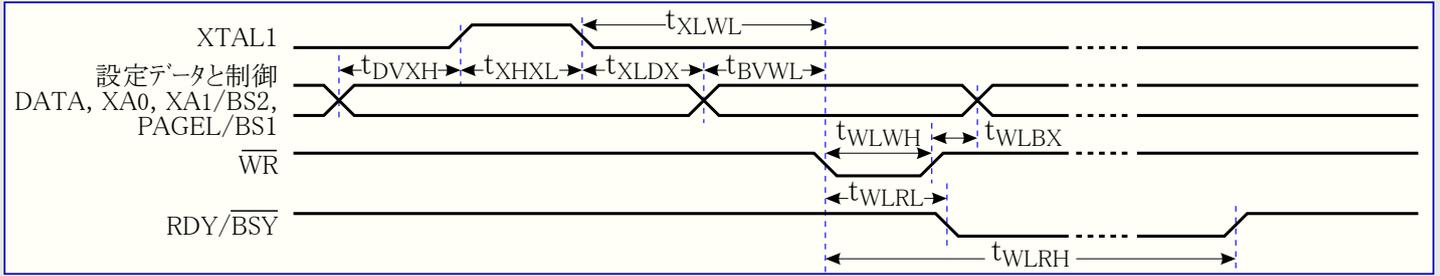
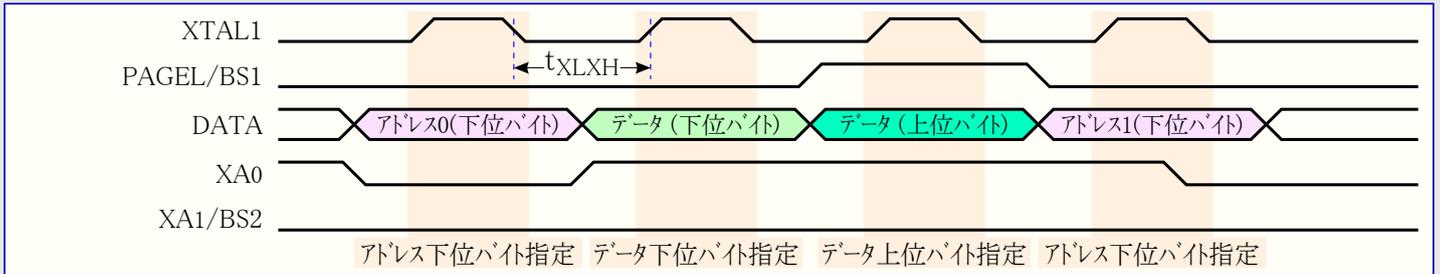
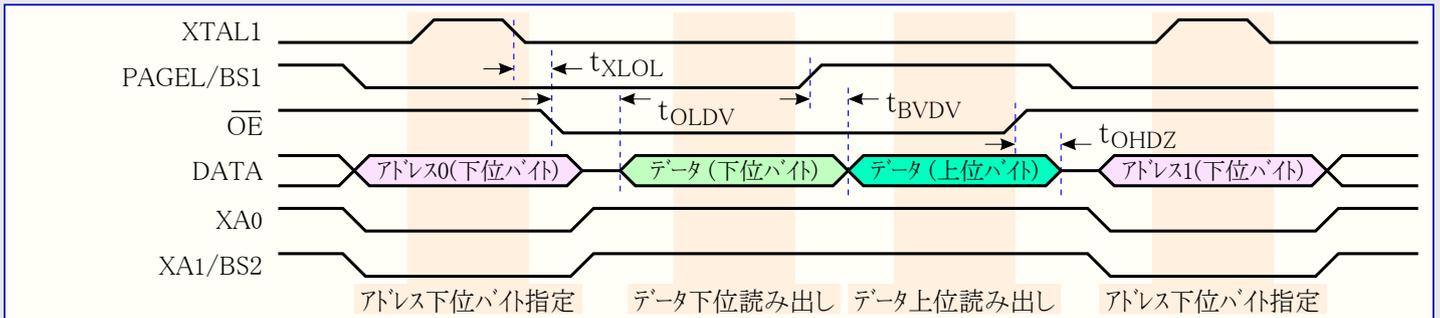


図22-6. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図22-5.で示されるタイミング必要条件(即ち、 $t_{DVXH}$ 、 $t_{XHXL}$ 、 $t_{XLDX}$ )は設定操作にも適用されます。

図22-7. 並列プログラミング タイミング (同一ページ読み出しでの必要条件)



注: 図22-5.で示されるタイミング必要条件(即ち、 $t_{DVXH}$ 、 $t_{XHXL}$ 、 $t_{XLDX}$ )は読み出し操作にも適用されます。

表22-8. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
VPP	プログラミング許可電圧	11.5		12.5	V
IPP	プログラミング許可電流			250	μA
tDVXH	XTAL1 ↑ に対するデータと制御の準備時間	67			ns
tXLXH	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
tXHXL	XTAL1 Highパルス幅	150			
tXLDX	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
tXLWL	XTAL1パルス ↓ 後のWR ↓ 待機時間	0			
tXLPH	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
tPLXH	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			
tBVPH	PAGELパルス ↑ に対するBS1準備時間	67			
tPHPL	PAGEL Highパルス幅	150			
tPLBX	PAGELパルス ↓ 後のBS1保持時間	67			
tWLBX	WRパルス ↓ 後のBS1,BS2保持時間	67			
tPLWL	PAGELパルス ↓ 後のWRパルス ↓ 待機時間	67			
tBVWL	WRパルス ↓ に対するBS1準備時間	67			
tWLWH	WR Lowパルス幅	150			
tWLRH	書き込み時間 (WR ↓ からRDY/BSY ↑) (注1)	3.7		4.5	
tWLRH_CE	チップ消去時間 (WR ↓ からRDY/BSY ↑) (注2)	7.5		9	ms
tXLCL	XTAL1パルス ↓ 後のOE ↓ 待機時間	0			ns
tBVDV	BS1有効からのDATA遅延時間	0		250	
tOLDV	OE ↓ 後のDATA出力遅延時間			250	
tOHDZ	OE ↑ 後のDATAフローティング遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

## 23. 代表特性

以下の図は代表的な特性を示します。これらの図は製造中に検査されていません。全ての消費電流測定は全I/Oピンを入力として設定した内部プルアップ許可で行われています。電源幅振幅の方形波発振器がクロック源として使われています。

活動動作とアイドル動作の消費電流測定は電力削減レジスタ(PRR)で全ビットが設定(1)で行なわれ、従って関係周辺機能部はOFFにされます。アナログ比較器もこれらの測定中に禁止されます。171頁の表23-1と表23-2はPRRによって制御される全周辺機能部についてICCに対する追加消費電流を示します。詳細については31頁の「PRR - 電力削減レジスタ」をご覧ください。

パワーダウン動作での消費電力はクロック選択と無関係です。

消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。支配的な要素は動作電圧と動作周波数です。

容量性負荷のピンの引き込み電流は(1つのピンに対して)  $CL(\text{負荷容量}) \times VCC(\text{動作電圧}) \times f(\text{I/Oピンの平均切り替え周波数})$  として推測できます。

デバイス検査範囲よりも高い周波数特性を示します。デバイスは注文番号が示す周波数よりも高い周波数での機能特性を保証されません。

ウォッチドッグ タイマ許可のパワーダウン動作での消費電流とウォッチドッグ タイマ禁止のパワーダウン動作での消費電流間の違いは、ウォッチドッグ タイマによって引き込んだ(消費した)差電流を表します。

### 23.1. 活動動作消費電流

図23-1. 活動動作消費電流 対 周波数 (100kHz~1MHz)

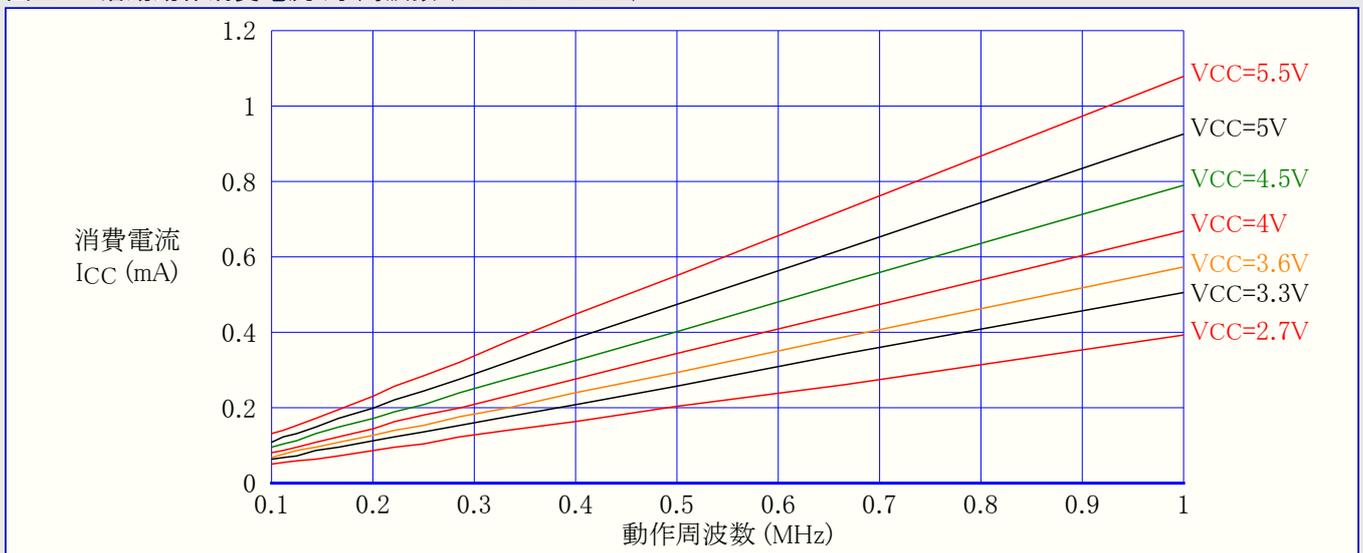
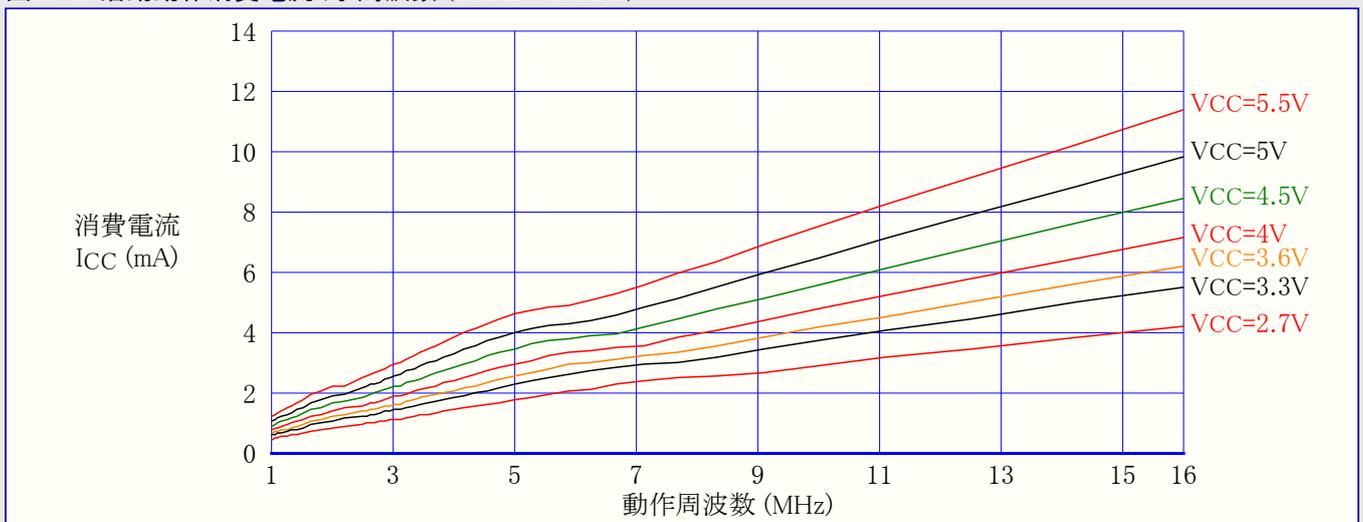


図23-2. 活動動作消費電流 対 周波数 (1MHz~16MHz)



(訳注) 原書の「代表特性」章は先祖帰りの可能性があるため、本書は23.2.1と23.2.2項とその関係文を残しました。

図23-3. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

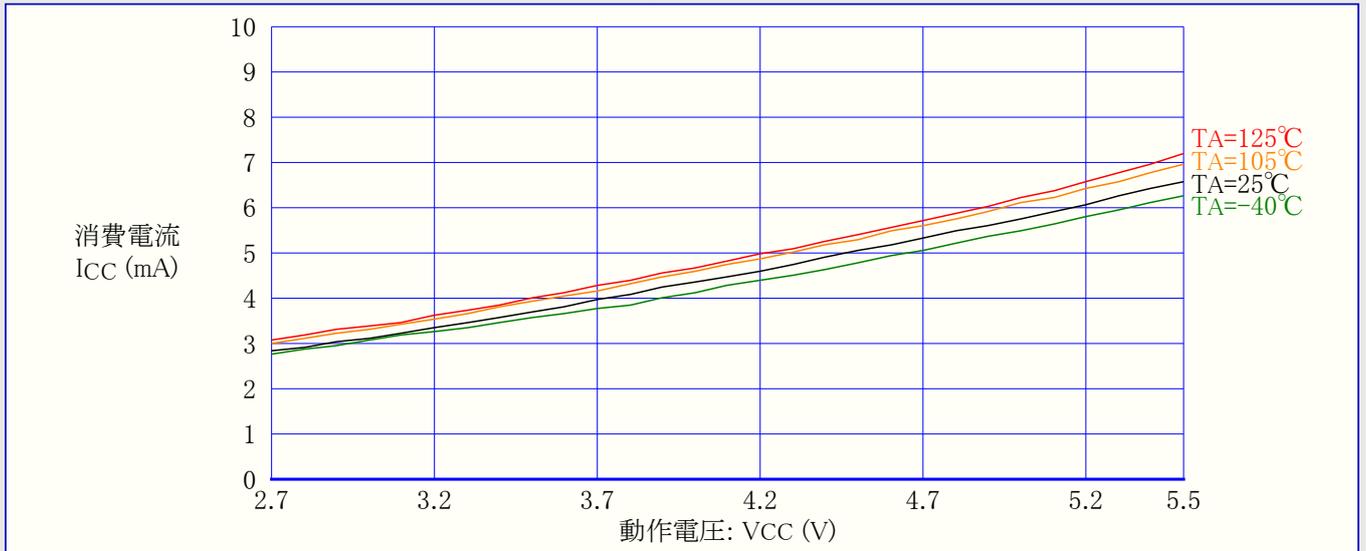
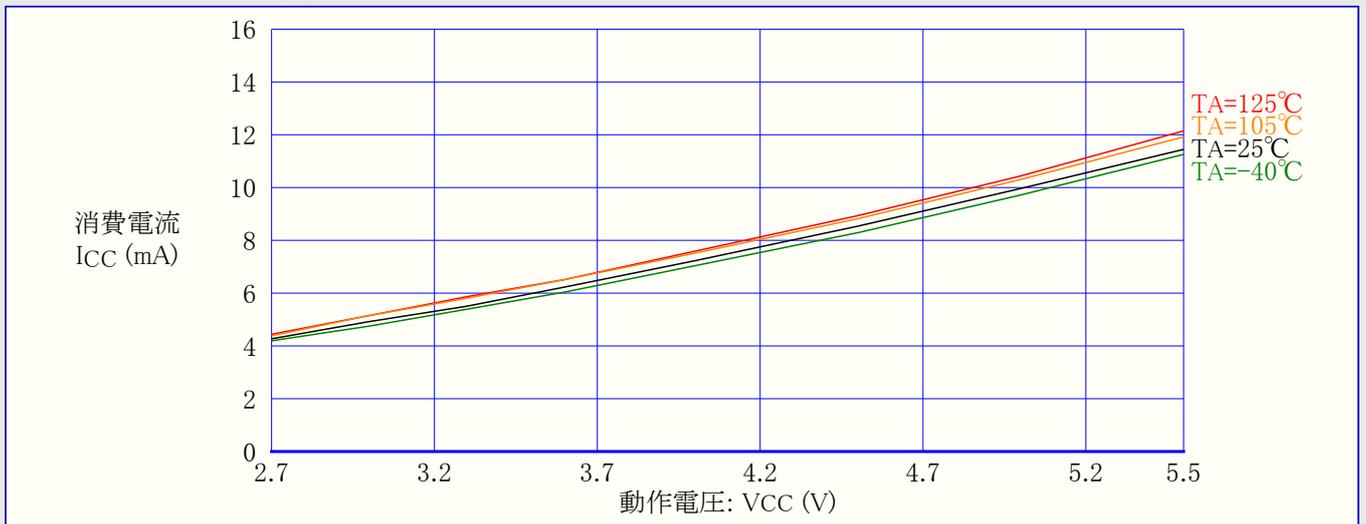


図23-4. 活動動作消費電流 対 動作電圧 (外部クロック信号,16MHz)



### 23.2. アイドル動作消費電流

図23-5. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

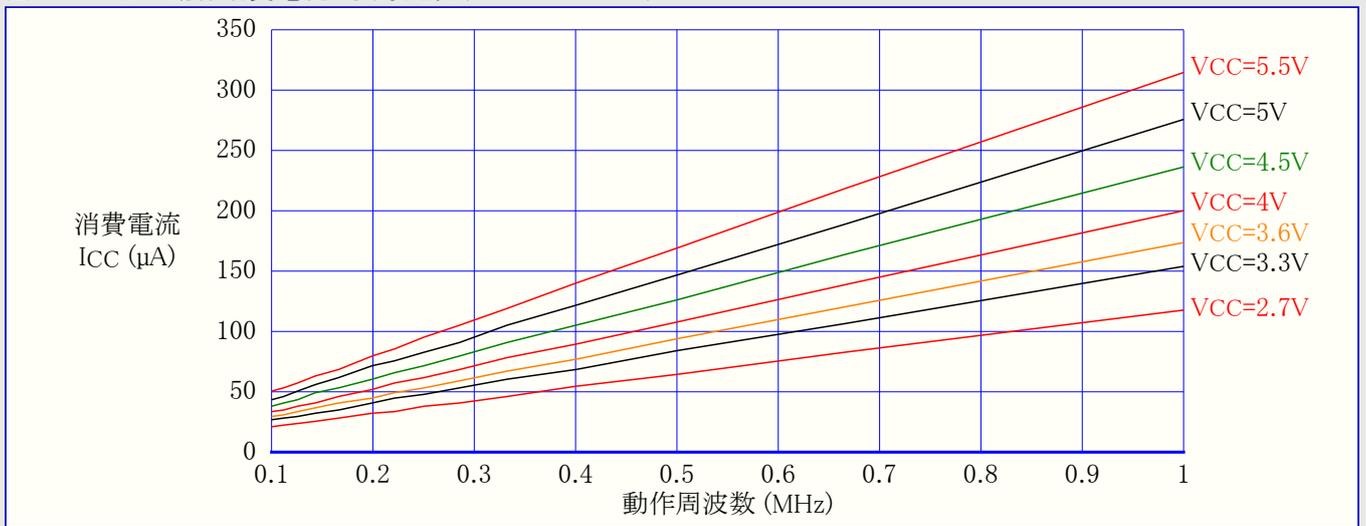


図23-6. アイドル動作消費電流 対 周波数 (1MHz~16MHz)

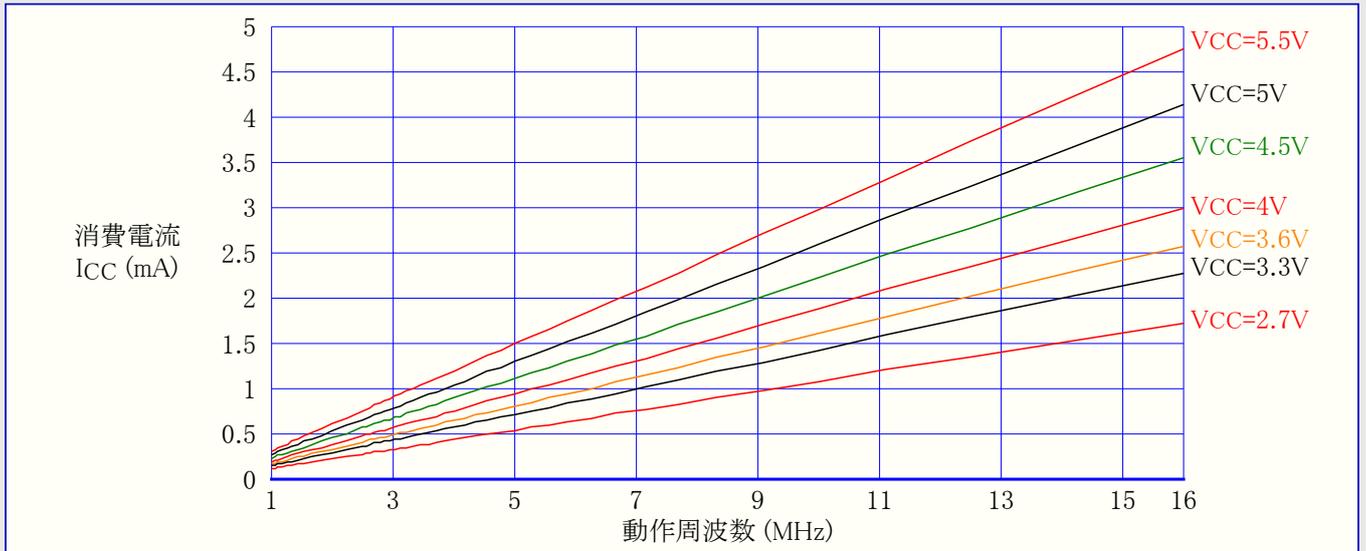


図23-7. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

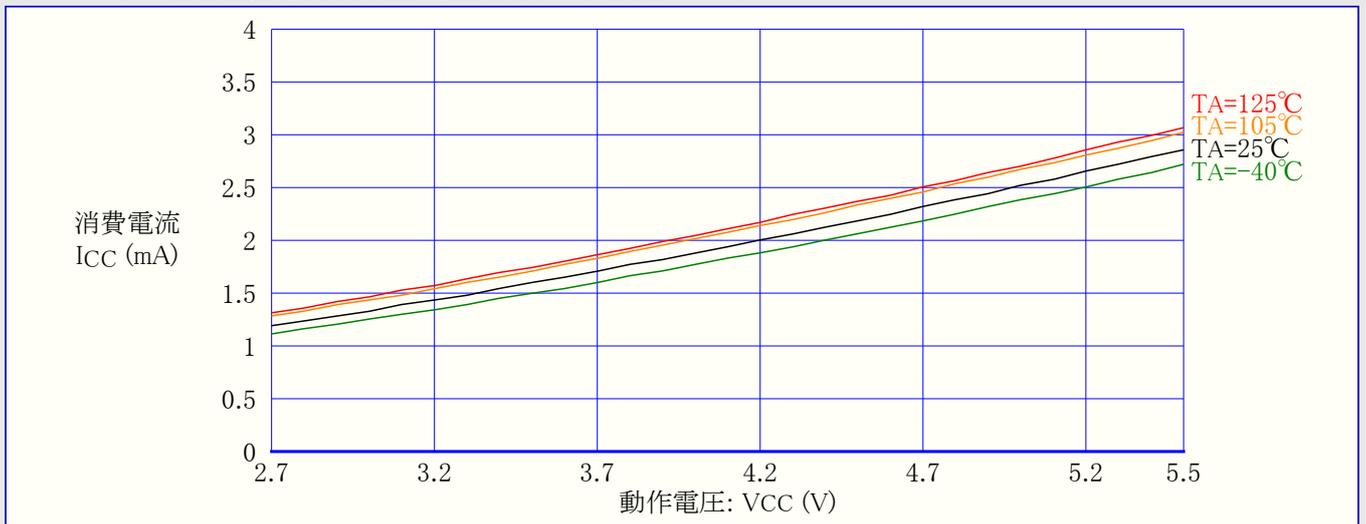
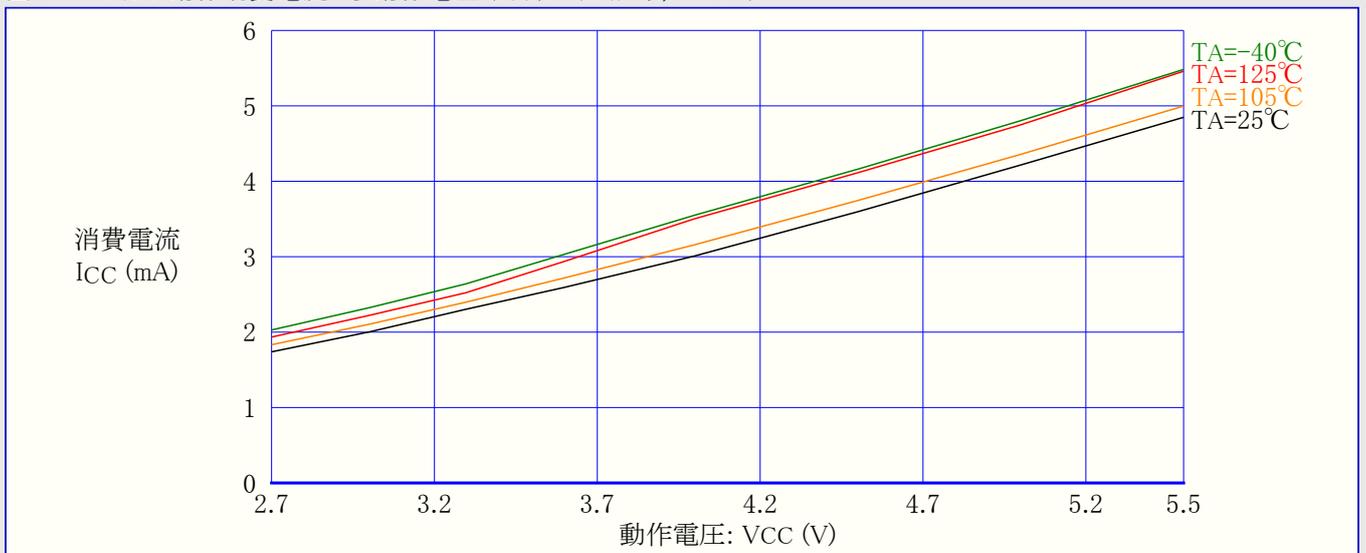


図23-8. アイドル動作消費電流 対 動作電圧 (外部クロック信号,16MHz)



## 23.2.1. 電力削減レジスタの使い方

以下の表と式は活動動作とアイドル動作で個別周辺機能部に対する追加消費電流の計算に使えます。周辺機能部の許可や禁止は電力削減レジスタによって制御されます。詳細については31頁の「PRR - 電力削減レジスタ」をご覧ください。

表23-1. 各部追加消費電流 (絶対値:  $\mu\text{A}$ )

PRR内ビット	8MHz,3V	16MHz,5V
PRPSC2	350	1300
PRPSCR	350	1300
PRTIM1	300	1150
PRSPI	250	900
PRADC	350	1300

表23-2. 各部追加消費電流 (相対値: %)

PRR内ビット	活動動作(図23-1,図23-2)	アイドル動作(図23-5,図23-6)
PRPSC2	10	25
PRPSCR	10	25
PRTIM1	8.5	22
PRSPI	5.3	14
PRADC	10.5	25

表23-1.で一覧される以外のVCCと周波数設定については表23-2.からの数値を元に代表的な消費電流を計算できます。

例1: VCC=3V,  $f=1\text{MHz}$ でタイマ/カウンタ1,SPIが許可されたアイドル動作での予測される消費電流を計算します。表23-2.のアイドル動作列からタイマ/カウンタ1が22%, SPIが14%追加する必要があります。図23-5.を読み、VCC=3V,  $f=1\text{MHz}$ でのアイドル動作消費電流が約0.17mAであることを得ます。タイマ/カウンタ1,SPI許可のアイドル動作での総消費電流を得ます。

$$\text{総消費電流} = 0.17\text{mA} \times (1 + 0.22 + 0.14) \approx 0.23\text{mA}$$

例2: 例1と同じ条件ですが、代わりに活動動作です。表23-2.の活動動作列からタイマ/カウンタ1が8.5%, SPIが5.3%追加する必要があります。図23-1.を読み、VCC=3V,  $f=1\text{MHz}$ での活動動作消費電流が約0.6mAであることを得ます。タイマ/カウンタ1,SPI許可の活動動作での総消費電流を得ます。

$$\text{総消費電流} = 0.6\text{mA} \times (1 + 0.085 + 0.053) \approx 0.68\text{mA}$$

例3: VCC=3.6V,  $f=10\text{MHz}$ で全周辺機能部許可の活動動作での予測消費電流を計算します。周辺機能部を除く活動動作消費電流が約7.0mAであることを得ます(図23-2.より)。そして表23-2.の活動動作列からの数値を使い、総消費電流を得ます。

$$\text{総消費電流} = 7.0\text{mA} \times (1 + 0.1 + 0.1 + 0.085 + 0.053 + 0.105) \approx 10.1\text{mA}$$

## 23.2.2. クロック前置分周での消費電力予測

図23-1., 図23-2., 図23-5., 図23-6.の曲線は外部クロック元で測定され、与えられた周波数でのマイクロコントローラコアの消費電力を提供します。

マイクロコントローラの総消費電力は18頁の図5-1.で示されるクロック系統図から生じ、故にそれは以下の項目の合計です。

- 発振器
- (使われるなら)PLL
- クロック前置分周器
- コア
- 周辺機能
- 入出力

周波数を減少するためのクロック前置分周器(CLKPR)使用時、発振器と前置分周器の消費電力が活動とアイドルの消費電力に対する支配的要素になり得ます。

発振器の代表的な消費電力は表25-3.で与えられます。

表23-3. クロックシステムの消費電力

単位部	周波数 (MHz)	代表的な消費電流 ( $\mu\text{A}$ )	
		VCC=3V	VCC=5V
ウォッチドッグ発振器	0.128	5	5
クリスタル用発振器	8	125	125
	16	250	250
RC発振器	8	350	350
PLL	64/16 (注1)	700	700
	48/12 (注2)	580	580
クロック前置分周器	N (注3)	$N \times 30$	$N \times 50$

注1: PLLはPSC用に64MHz、CPU用に16MHzを生成します。

注2: PLLはPSC用に48MHz、CPU用に12MHzを生成します。

注3: Nはクロック前置分周器の入力に於ける(MHzでの)周波数です。

例1: VCC=3Vでクロック前置分周レジスタ(CLKPR)を使って更に16分周される16MHzを生成するためのPLLに供給する8MHzのクリスタル用発振器でのアイドル動作の予測消費電流を計算します。従ってシステムクロックは1MHzです。周辺機能と入出力は活性ではありません。

表23-3を読み、クリスタル用発振器電流が0.125mA、PLL電流が0.7mA、そして前置分周器は16MHzでクロック駆動されるものとして $16 \times 30 = 0.48\text{mA}$ が必要であることを知ります。

図23-5を読み、VCC=3V、 $f=1\text{MHz}$ でのコアのアイドル動作消費電流が約0.17mAであることを得ます。

クリスタル用発振器とPLLとでのアイドル動作での総消費電流を得ます。

$$\text{総消費電流} = 0.125 + 0.7 + 0.48 + 0.17 = 1.475\text{mA}$$

例2: VCC=5Vでクロック前置分周レジスタ(CLKPR)を使って更に8分周される4MHzのクリスタル用発振器での活動動作の予測消費電流を計算します。従ってシステムクロックは0.5MHzです。周辺機能と入出力は活性ではありません。

表23-3を読み、クリスタル用発振器電流が0.06mA、PLL電流が0.7mA、そして前置分周器は4MHzでクロック駆動されるものとして $4 \times 50 = 0.2\text{mA}$ が必要であることを知ります。

図23-1を読み、VCC=5V、 $f=0.5\text{MHz}$ でのコアの活動動作消費電流が約0.6mAであることを得ます。

4MHzクリスタルとでの活動動作での総消費電流を得ます。

$$\text{総消費電流} = 0.06 + 0.2 + 0.6 = 0.86\text{mA}$$

### 23.3. パワーダウン動作消費電流

図23-9. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)

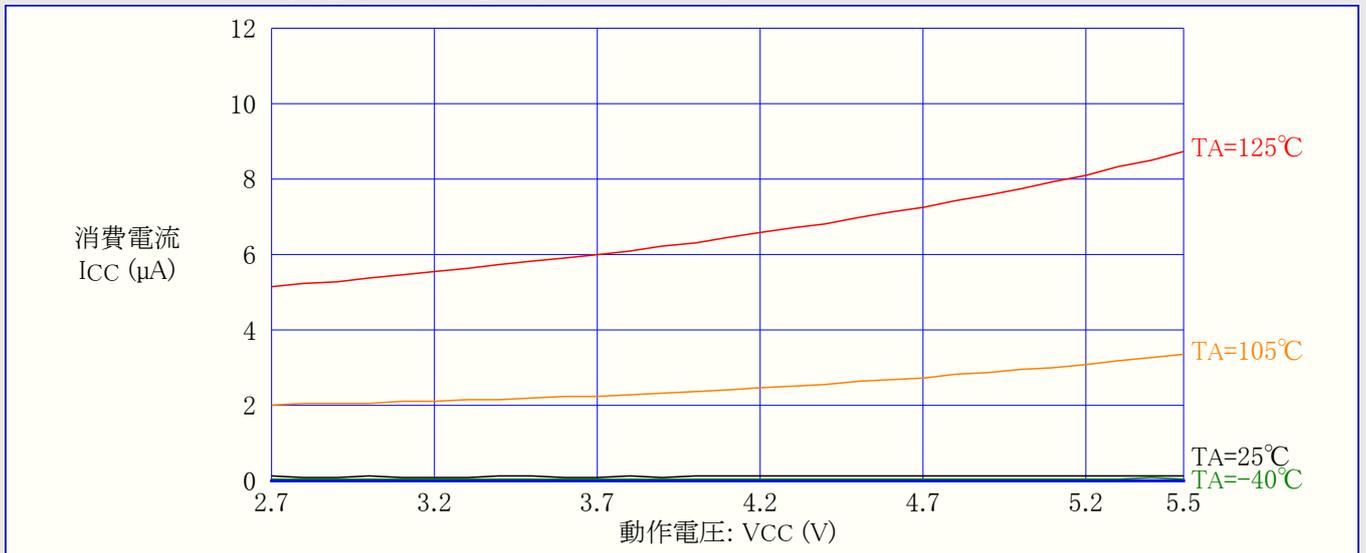
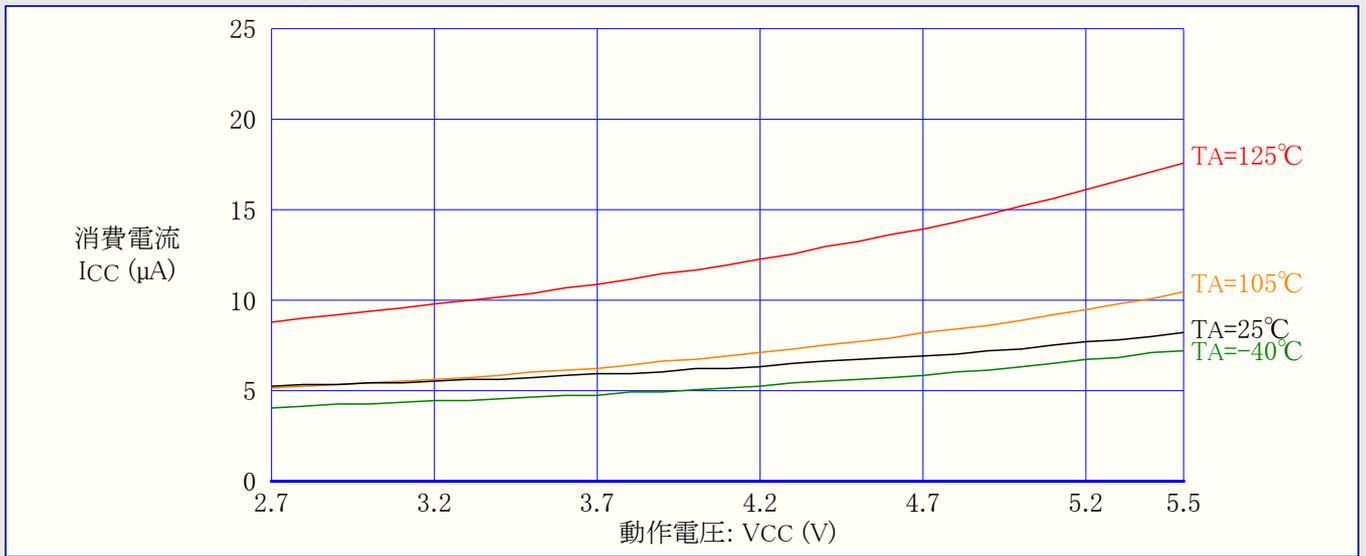


図23-10. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



23.4. ピンプルアップ

図23-11. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

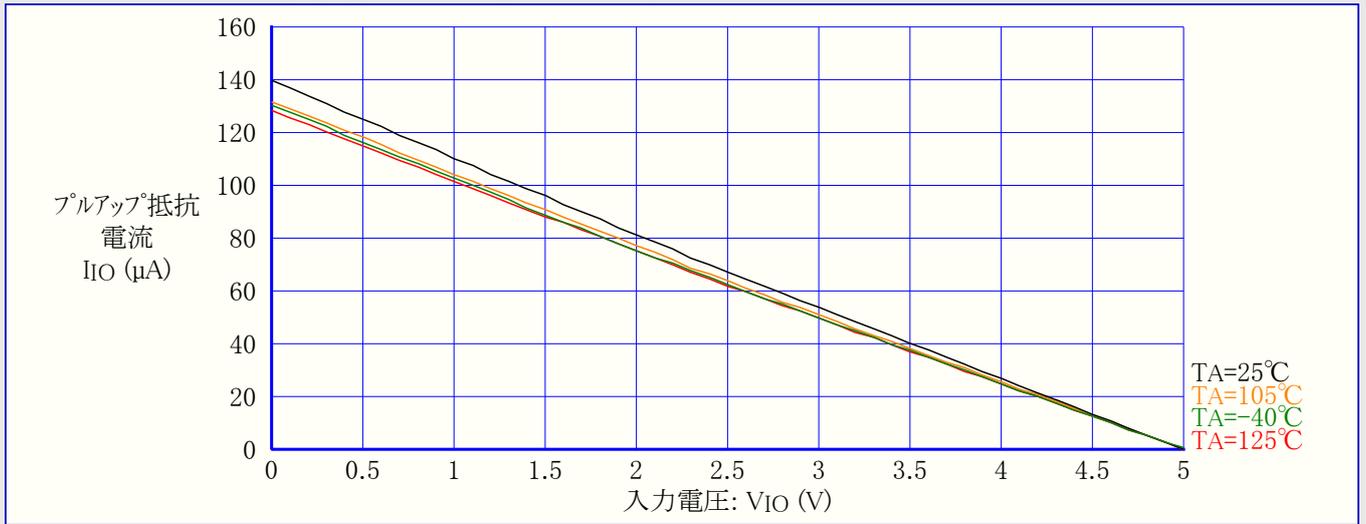


図23-12. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

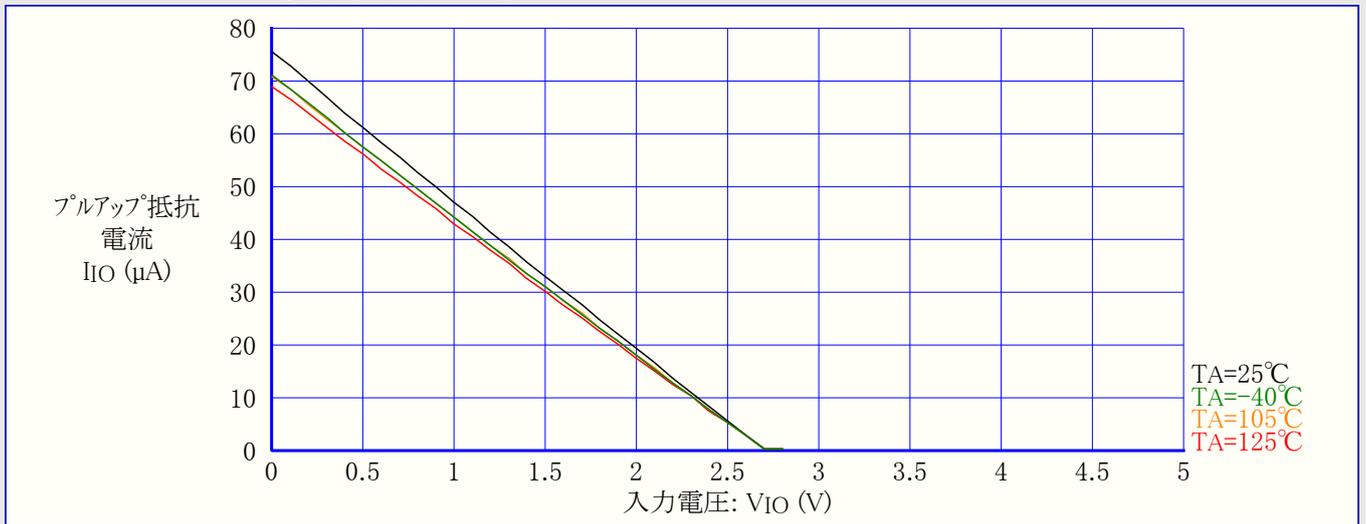


図23-13. PE1,PE2 I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

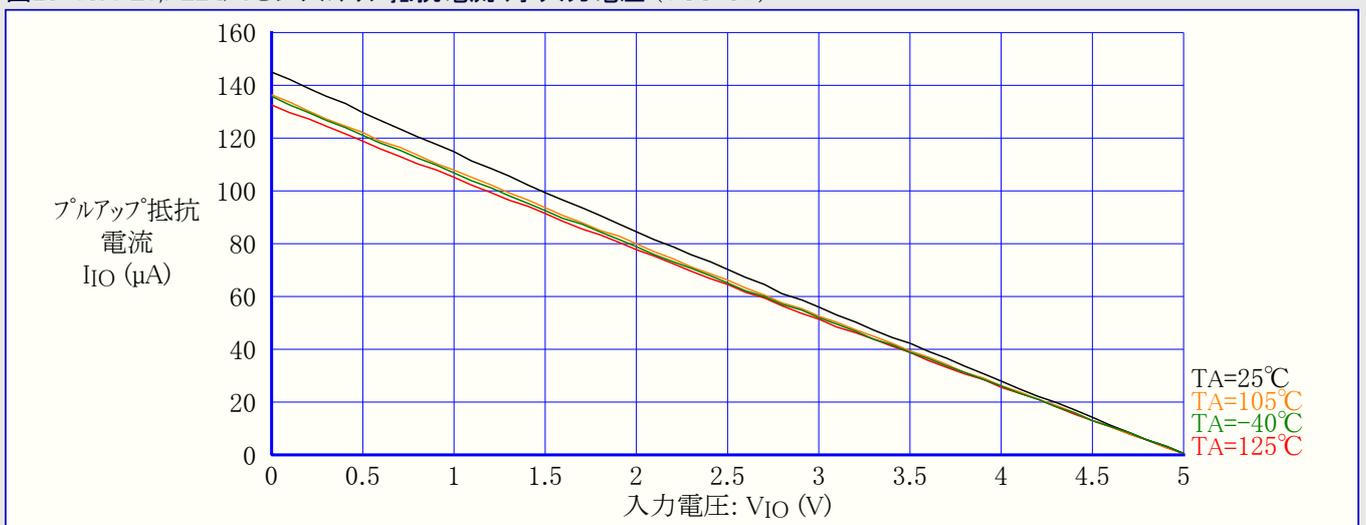


図23-14. PE1,PE2 I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

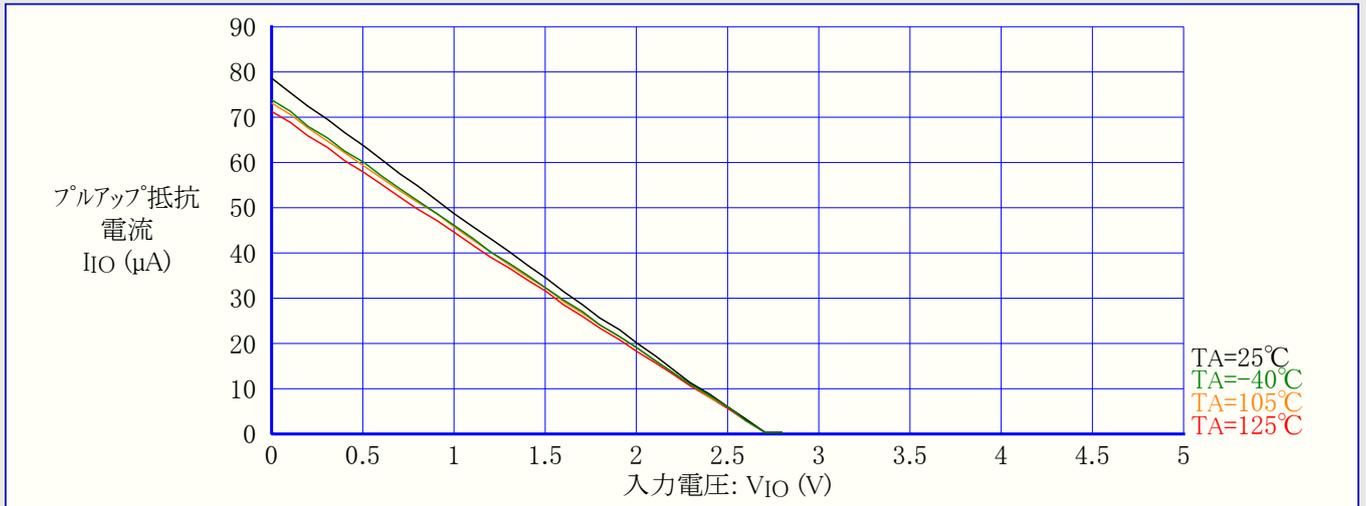


図23-15. RESETピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

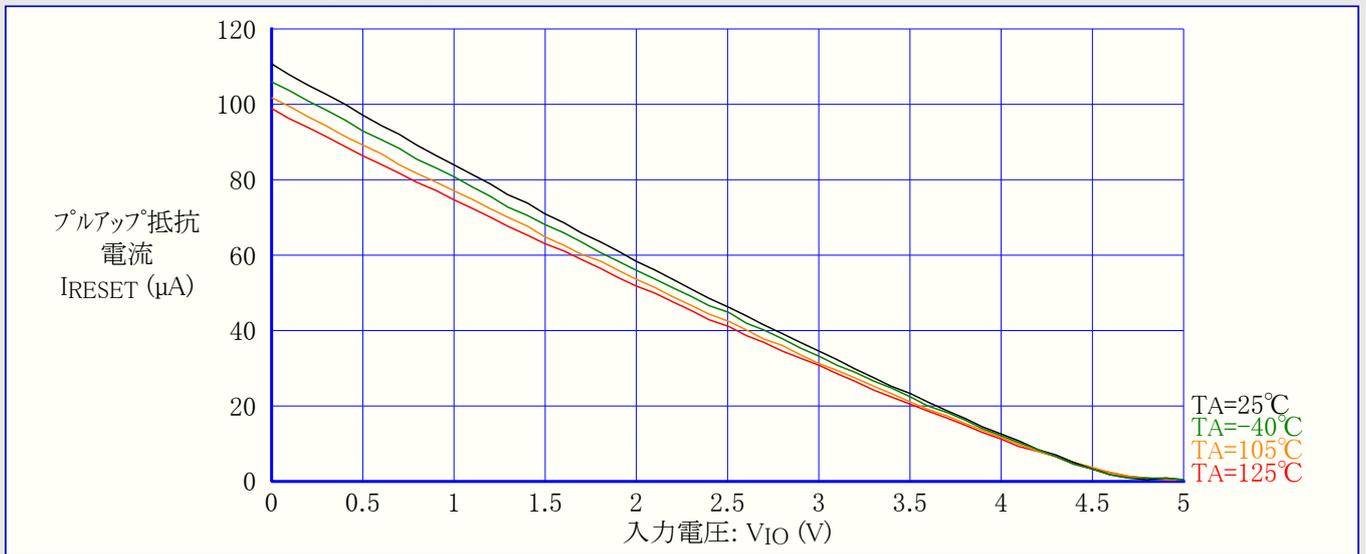
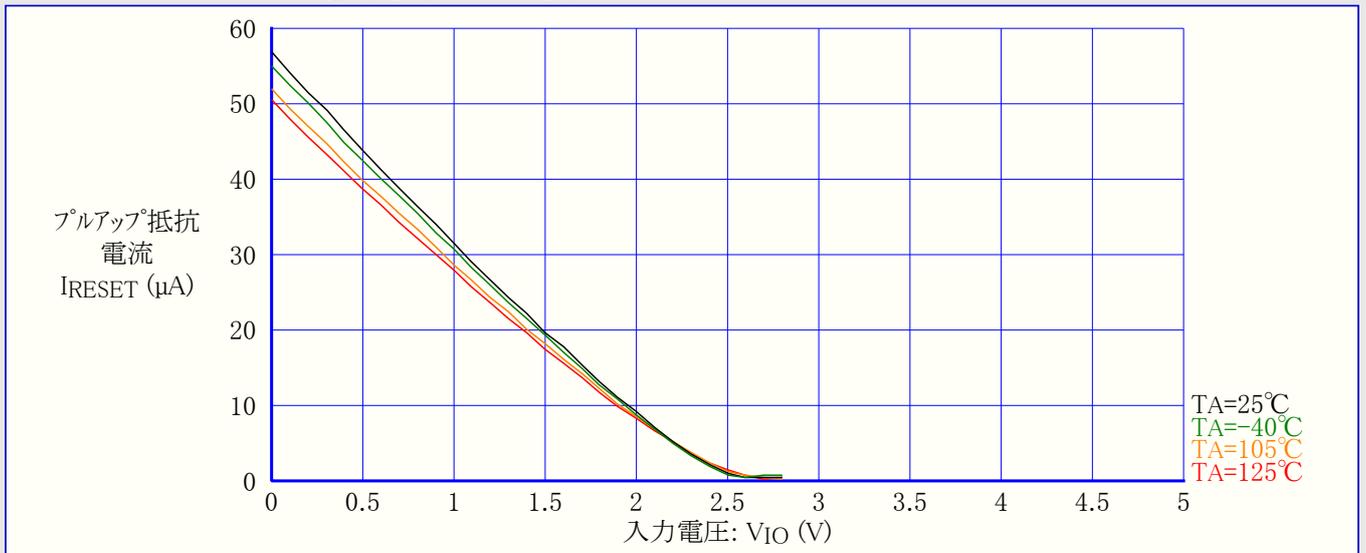


図23-16. RESETピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)



23.5. ピンHigh出力電圧

図23-17. I/Oピン吐き出し電流 対 出力電圧 (VCC=5V)

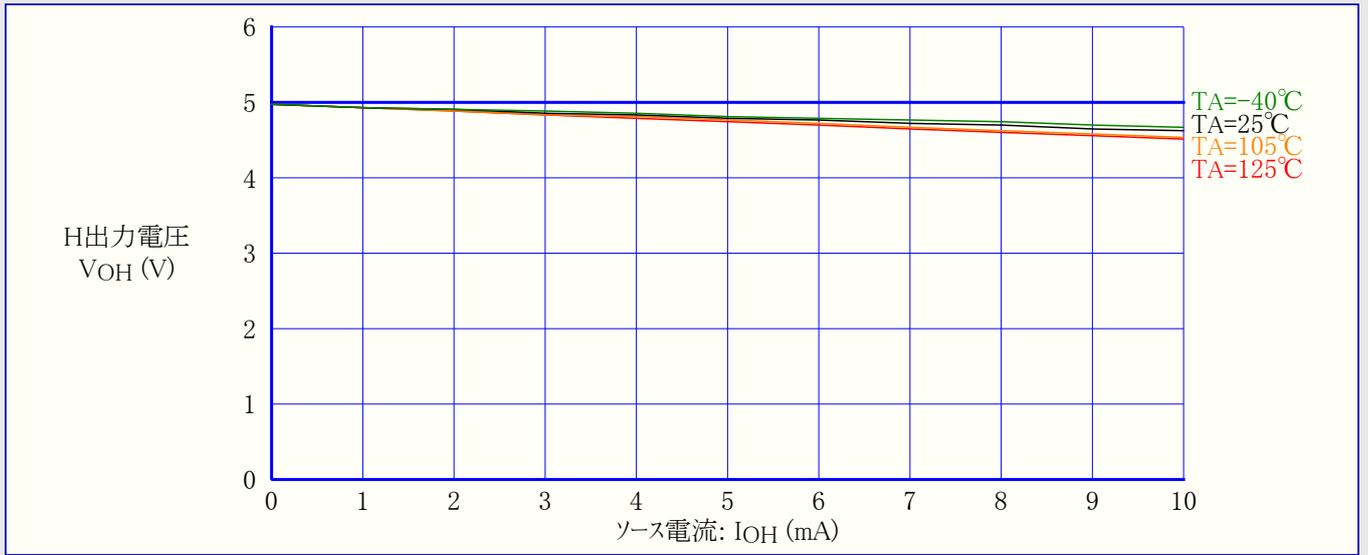
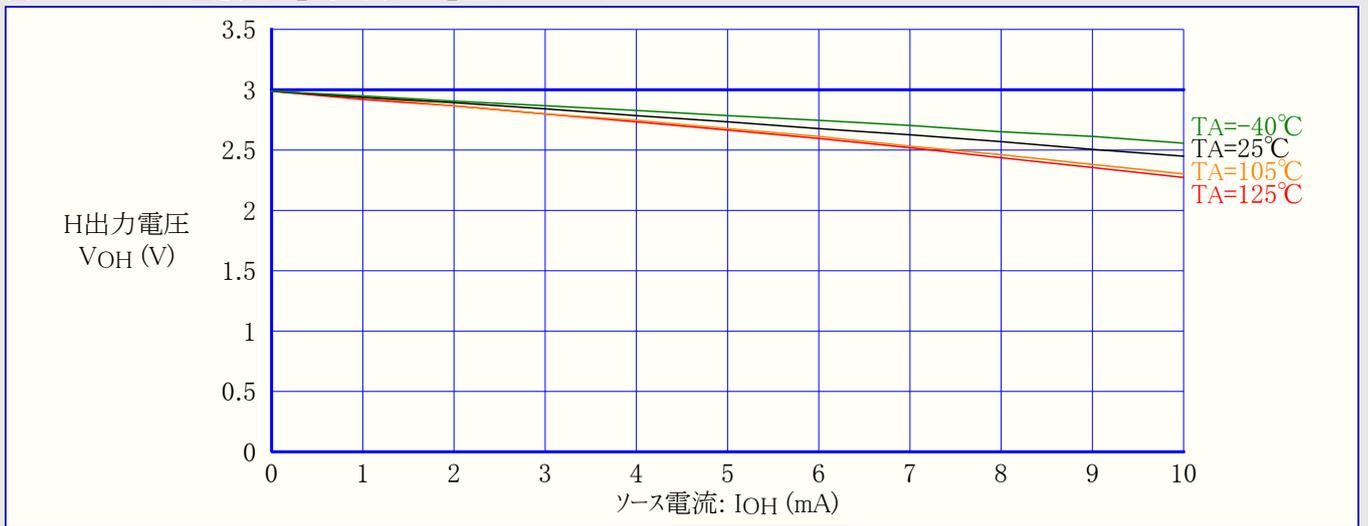


図23-18. I/Oピン吐き出し電流 対 出力電圧 (VCC=3V)



23.6. ピンLow出力電圧

図23-19. I/Oピン吸い込み電流 対 出力電圧 (VCC=5V)

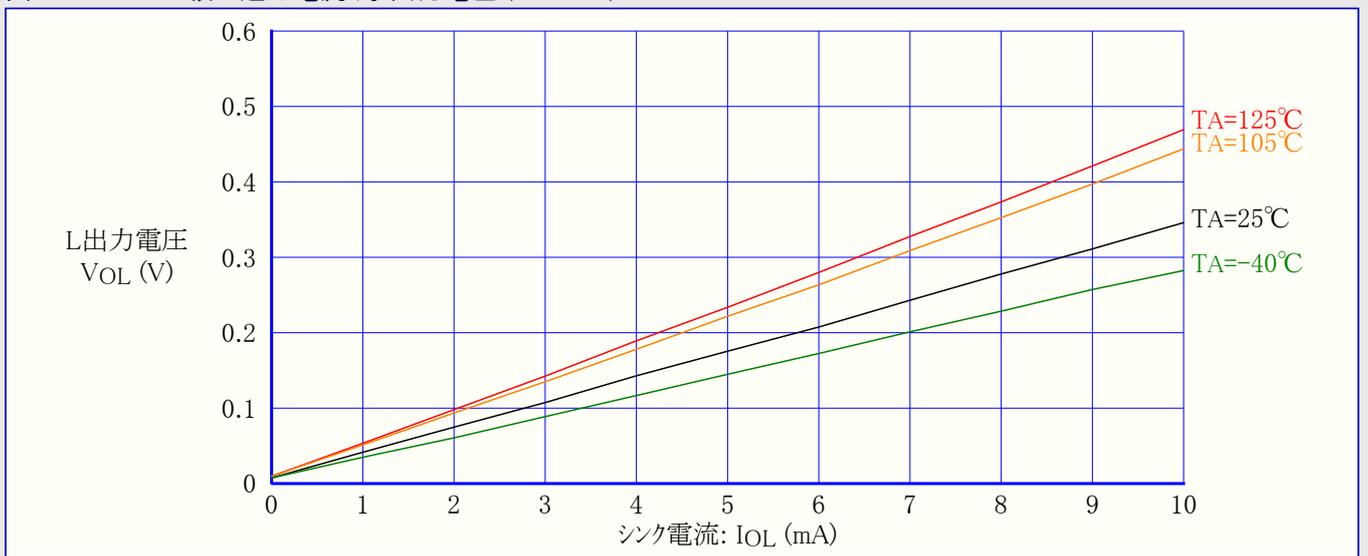
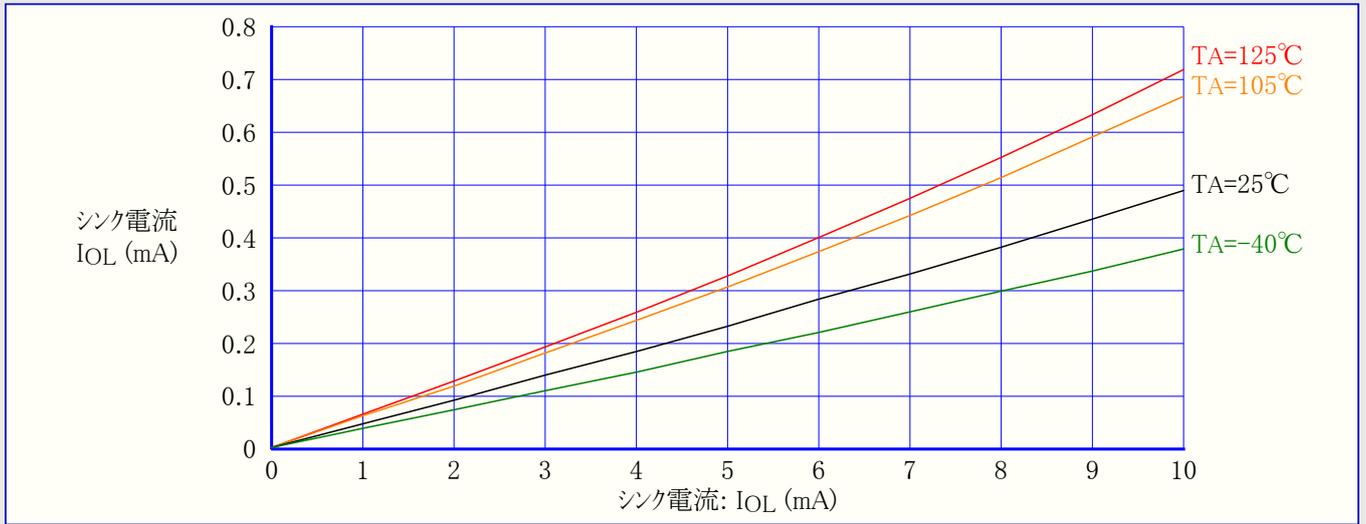


図23-20. I/Oピン吸い込み電流 対 出力電圧 (VCC=3V)



### 23.7. ピン閾値

図23-21. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIL,0読み値)

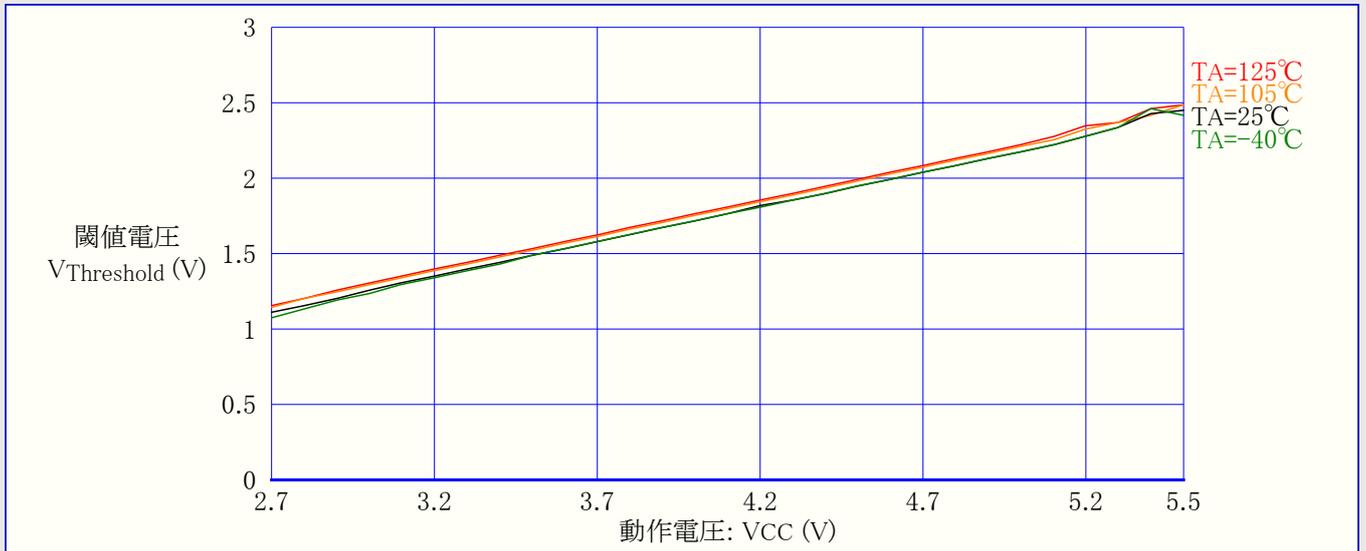
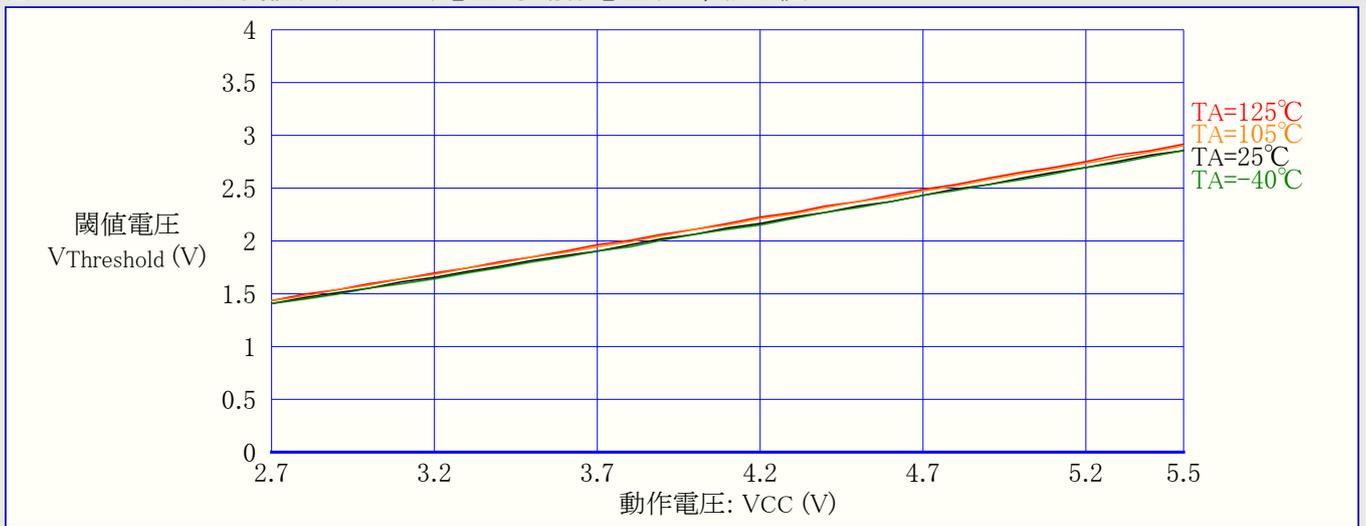


図23-22. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIH,1読み値)



23.8. 低電圧検出器(BOD)閾値

図23-23. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧4.3V)

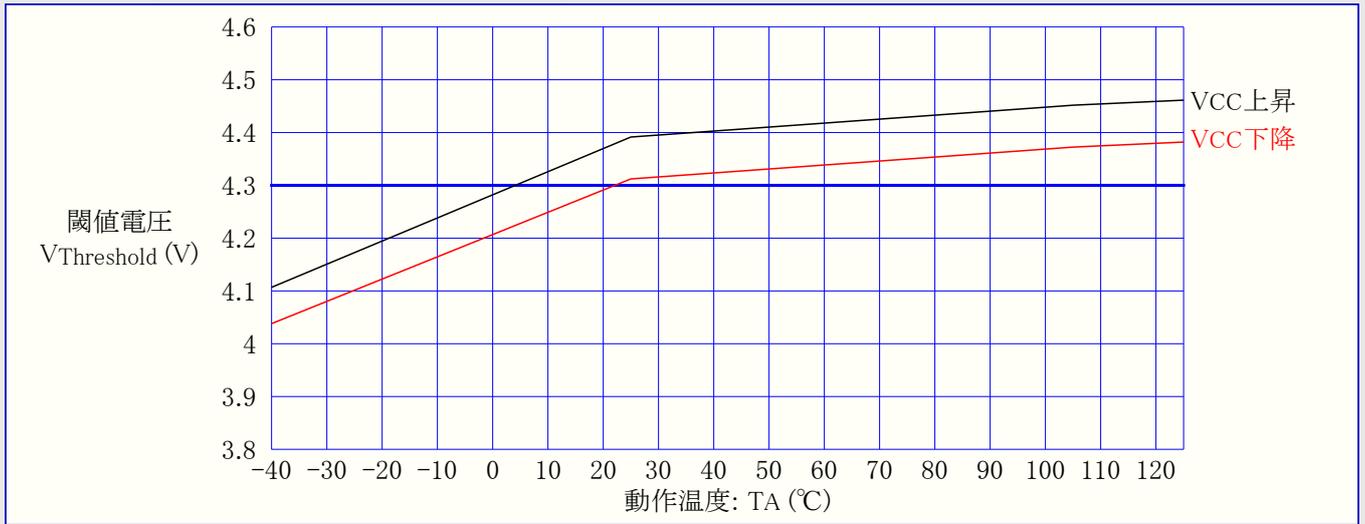
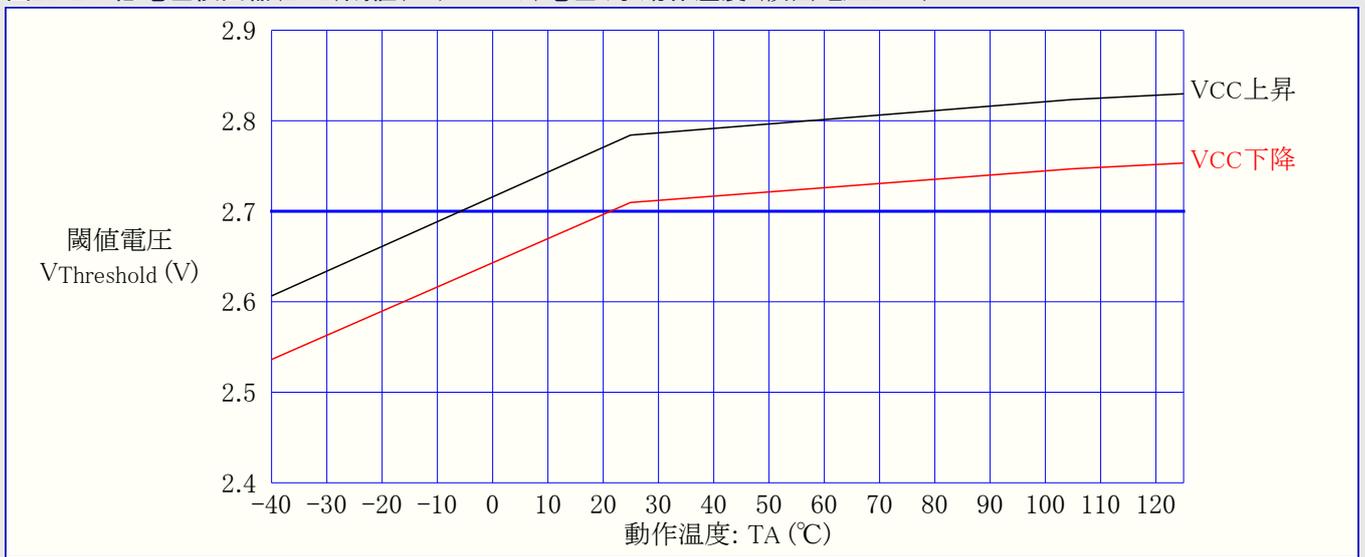


図23-24. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)



23.9. アナログ基準電圧

図23-25. 内部2.56V基準電圧 対 動作電圧

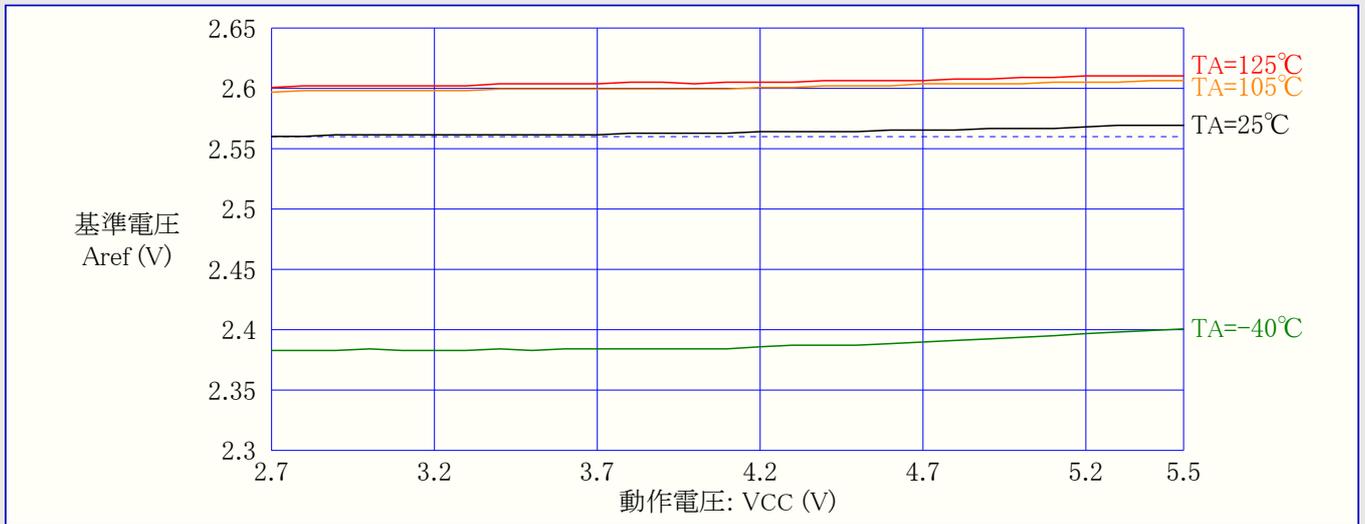
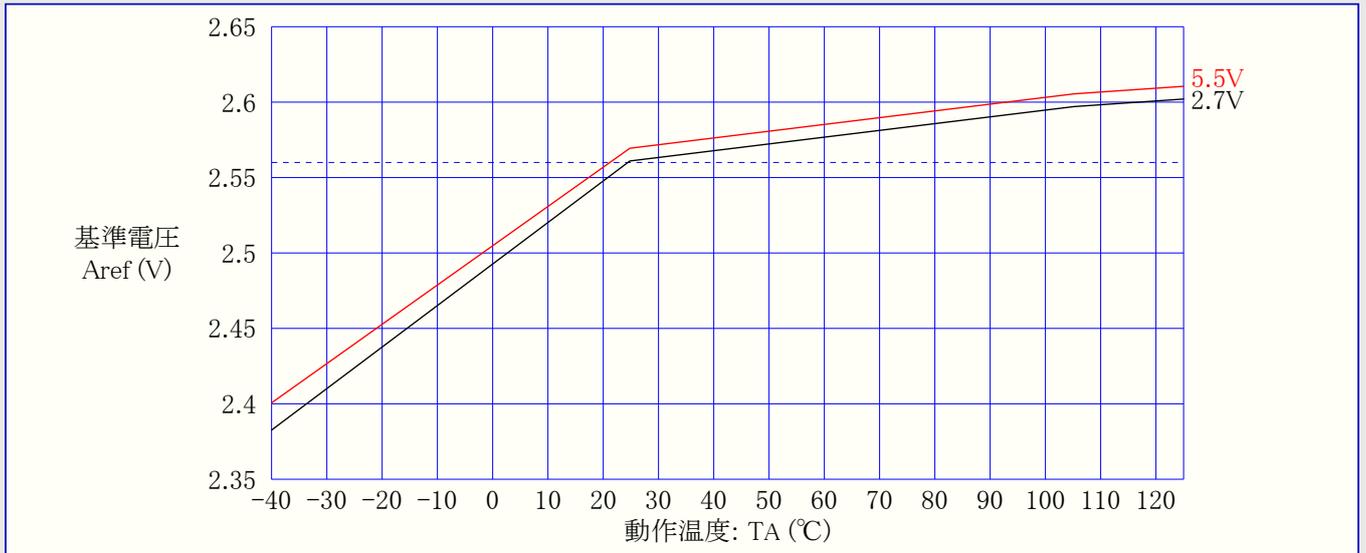


図23-26. 内部2.56V基準電圧 対 動作温度



### 23.10. 内部発振器周波数

図23-27. ウォッチドッグ発振器周波数 対 動作電圧

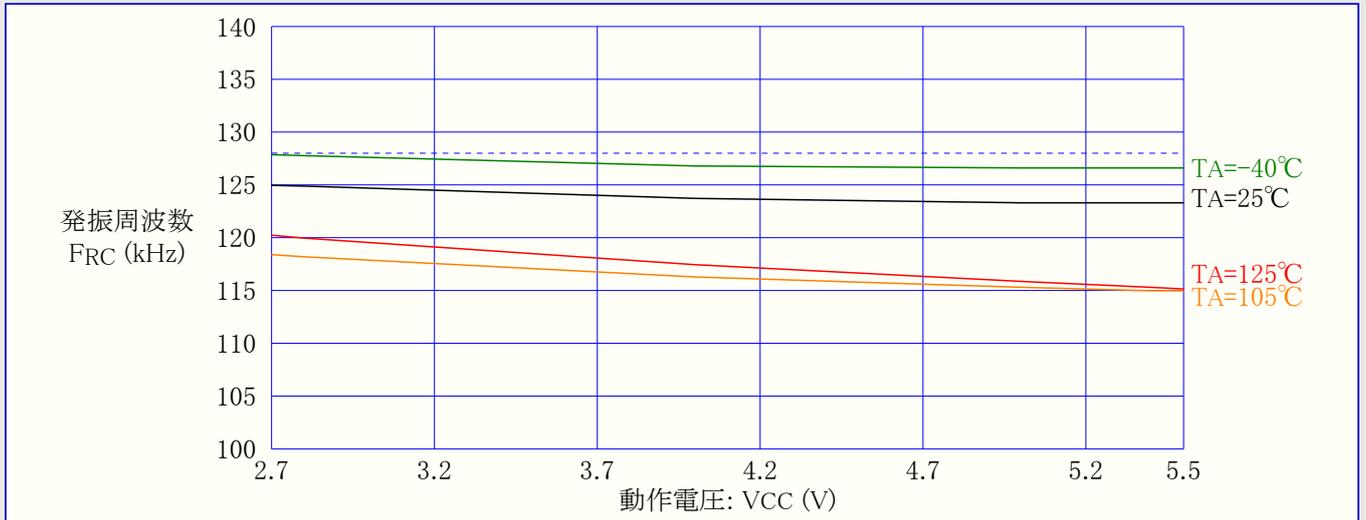


図23-28. 校正済み8MHz内蔵RC発振器周波数 対 動作温度

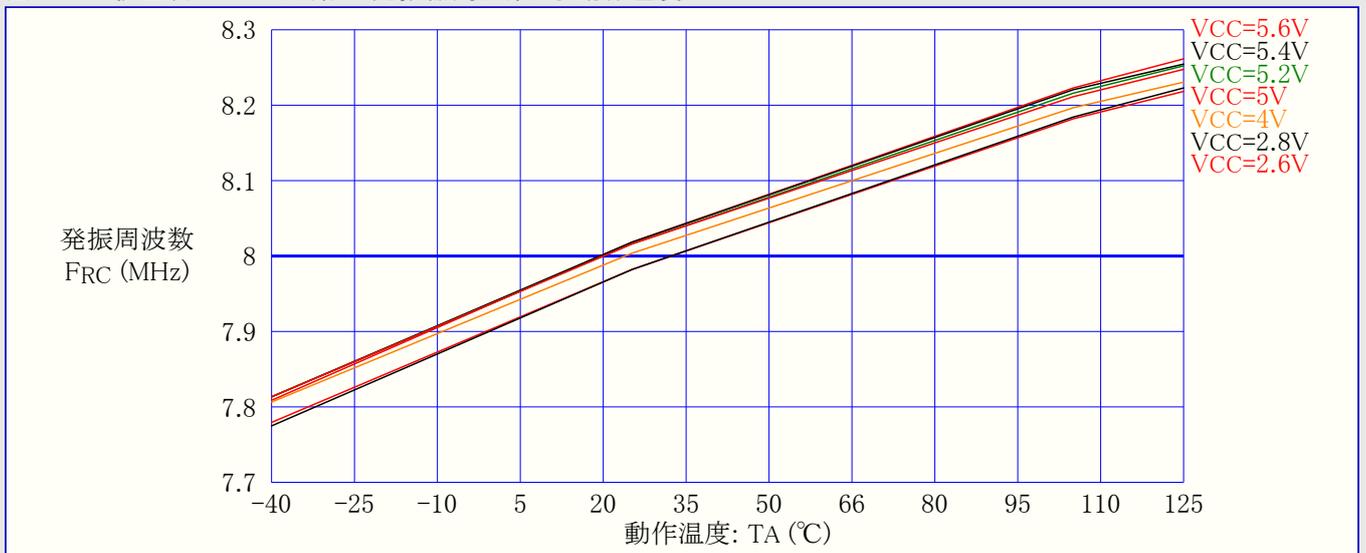


図23-29. 校正済み8MHz内蔵RC発振器周波数 対 動作電圧

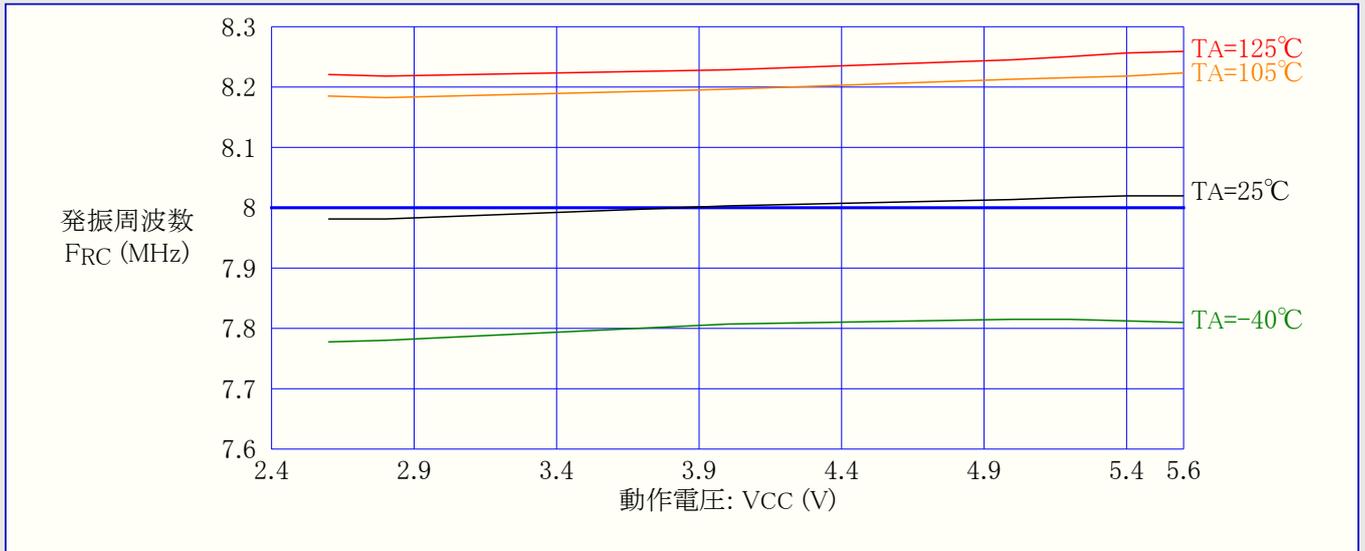
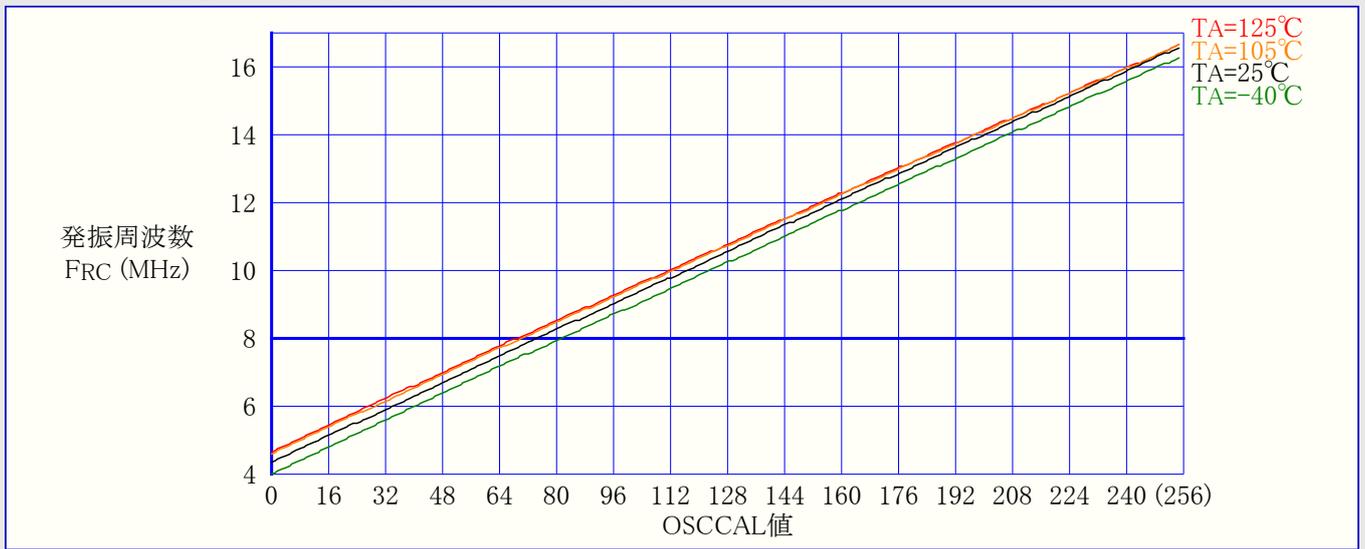


図23-30. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値 (VCC=3V, 250ns間隔10000回採取)



### 23.11. リセット消費電流

図23-31. リセット消費(供給)電流 対 周波数 (100kHz~1MHz、RESETプルアップ電流を除く)

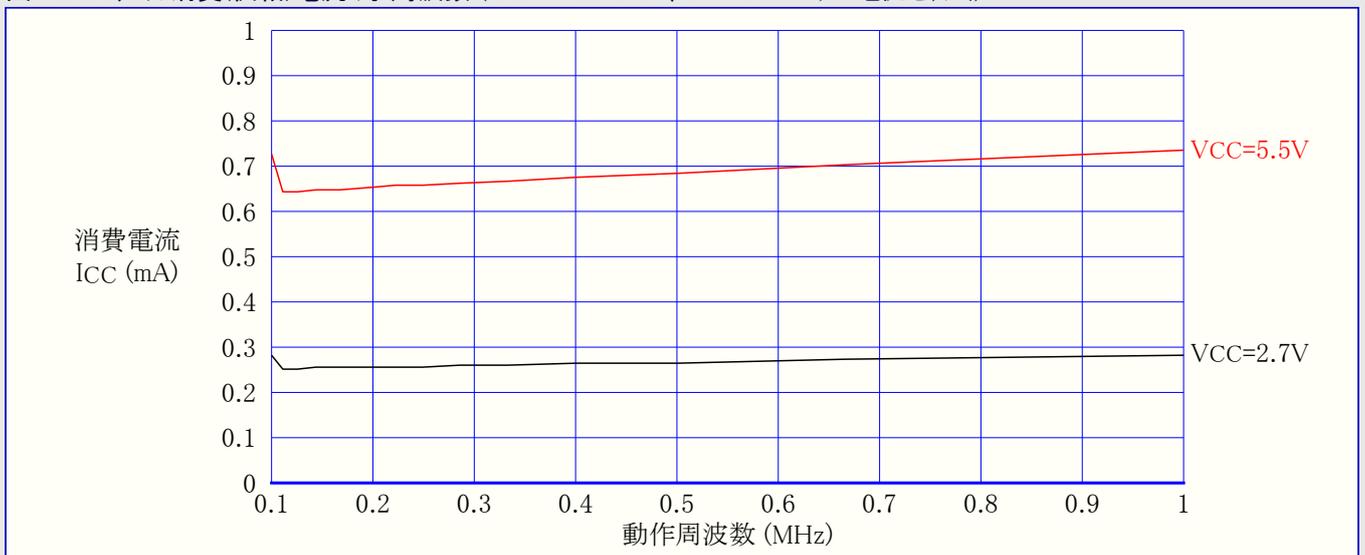
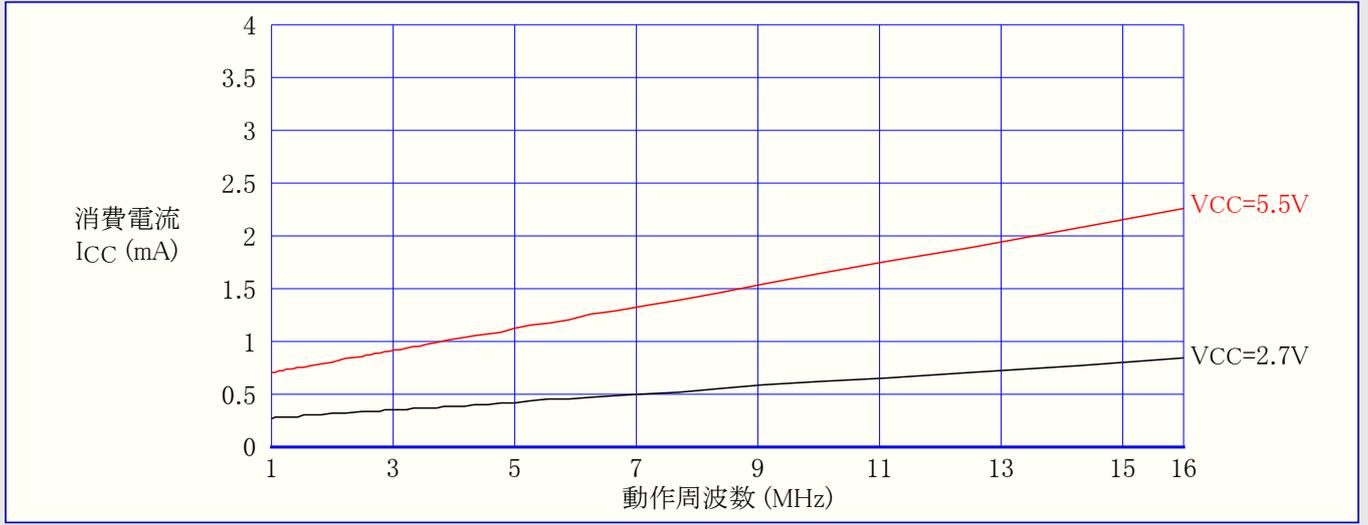


図23-32. リセット消費(供給)電流 対 周波数 (1MHz~16MHz、RESETプルアップ電流を除く)



24. レジスタ要約

拡張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
(\$FF)~(\$90)	予約									
(\$8F)	予約									
(\$8E)	予約									
(\$8D)	ICR1H	タイマ/カウンタ1 捕獲レジスタ上位バイト								60
(\$8C)	ICR1L	タイマ/カウンタ1 捕獲レジスタ下位バイト								
(\$8B)	予約									
(\$8A)	TCCR1B	ICNC1	ICES1	-	WGM13	-	CS12	CS11	CS10	59
(\$89)	EICRA	-	-	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	52
(\$88)	OSCCAL	内蔵RC発振器 発振校正値レジスタ								26
(\$87)	PLLCSR	-	-	PLLF3	PLLF2	PLLF1	PLLF0	PLLE	PLOCK	27
(\$86)	PRR	PRPSC2	-	PRPSCR	PRTIM1	-	PRSPI	-	PRADC	31
(\$85)	CLKSELR	-	COU	CSUT1	CSUT0	CSEL3	CSEL2	CSEL1	CSEL0	28
(\$84)	CLKCSR	CLKCCE	-	-	CLKRDY	CLKC3	CLKC2	CLKC1	CLKC0	28
(\$83)	CLKPR	CLKPCE	-	-	-	CLKPS3	CLKPS2	CLKPS1	CLKPS0	26
(\$82)	WDTCR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	37
(\$81)	BGCCR	-	-	-	-	BGCC3	BGCC2	BGCC1	BGCC0	113
(\$80)	BGCRR	-	-	-	-	BGCR3	BGCR2	BGCR1	BGCR0	114
(\$7F)	AC3CON	AC3EN	AC3IE	AC3IS1	AC3IS0	AC3OEA	AC3M2	AC3M1	AC3M0	118
(\$7E)	AC2CON	AC2EN	AC2IE	AC2IS1	AC2IS0	-	AC2M2	AC2M1	AC2M0	118
(\$7D)	AC1CON	AC1EN	AC1IE	AC1IS1	AC1IS0	-	AC1M2	AC1M1	AC1M0	118
(\$7C)	AC3ECON	-	-	AC3OIS1	AC3OE	-	AC3H2	AC3H1	AC3H0	119
(\$7B)	AC2ECON	-	-	AC2OIS1	AC2OE	-	AC2H2	AC2H1	AC2H0	119
(\$7A)	AC1ECON	-	-	AC1OIS1	AC1OE	AC1ICE	AC1H2	AC1H1	AC1H0	119
(\$79)	AMP0CSR	AMP0EN	AMP0IS	AMP0G1	AMP0G0	AMP0GS	-	AMP0TS1	AMP0TS0	134
(\$78)	DIDR1	-	-	-	-	ACMP1MD	AMP0+D	ADC10D	ADC9D	121,131
(\$77)	DIDR0	ADC8D ACMP3D	ADC7D AMP0-D	ADC5D ACMP2D	ADC4D ACMP3MD	ADC3D ACMPMD	ADC2D ACMP2MD	ADC1D	ADC0D ACMP1D	121,131
(\$76)	DACON	DAATE	DATS2	DATS1	DATS0	-	DALA	-	DAEN	136
(\$75)	予約	(アナログ検査レジスタ用)								
(\$74)	予約									
(\$73)	予約									
(\$72)	予約									
(\$71)	PASDLY2	アナログ同期遅延レジスタ								84
(\$70)	PCNFE2	PASDLK22	PASDLK21	PASDLK20	PBFM21	PELEV2A1	PELEV2B1	PISEL2A1	PISEL2B1	83
(\$6F)	POM2	POMV2B3	POMV2B2	POMV2B1	POMV2B0	POMV2A3	POMV2A2	POMV2A1	POMV2A0	86
(\$6E)	PSOC2	POS23	POS22	PSYNC21	PSYNC20	POEN2D	POEN2B	POEN2C	POEN2A	80
(\$6D)	PICR2H	PCST2	-	-	-	電力段制御器2 捕獲レジスタ 上位バイト				86
(\$6C)	PICR2L	電力段制御器2 捕獲レジスタ 下位バイト								
(\$6B)	予約									
(\$6A)	PSOC0	PISEL0A1	PISEL0B1	PSYNC01	PSYNC00	-	POEN0B	-	POEN0A	101
(\$69)	PICR0H	PCST0	-	-	-	電力段制御器0 捕獲レジスタ 上位バイト				105
(\$68)	PICR0L	電力段制御器0 捕獲レジスタ 下位バイト								
(\$67)	PFRC2B	PCAE2B	PISEL2B0	PELEV2B0	PFLTE2B	PRFM2B3	PRFM2B2	PRFM2B1	PRFM2B0	104
(\$66)	PFRC2A	PCAE2A	PISEL2A0	PELEV2A0	PFLTE2A	PRFM2A3	PRFM2A2	PRFM2A1	PRFM2A0	104
(\$65)	OCR2SAH	-	-	-	-	電力段制御器2 比較SAレジスタ 上位バイト				81
(\$64)	OCR2SAL	電力段制御器2 比較SAレジスタ 下位バイト								
(\$63)	PFRC0B	PCAE0B	PISEL0B0	PELEV0B0	PFLTE0B	PRFM0B3	PRFM0B2	PRFM0B1	PRFM0B0	101
(\$62)	PFRC0A	PCAE0A	PISEL0A0	PELEV0A0	PFLTE0A	PRFM0A3	PRFM0A2	PRFM0A1	PRFM0A0	101
(\$61)	OCR0SAH	-	-	-	-	縮小電力段制御器 比較SAレジスタ 上位バイト				102
(\$60)	OCR0SAL	縮小電力段制御器 比較SAレジスタ 下位バイト								

注意: • 将来のデバイスとの共通性のため、アクセスされる場合の予約ビットは0を書かれるべきです。予約したI/Oメモリアドレスは決して書かれるべきではありません。

- アドレス範囲\$00~\$1F内のI/OレジスタはSBIとCBI命令を使う直接ビットアクセスが可能です。これらのレジスタではSBISとSBIC命令を使うことによって単一ビット値が検査できます。
- いくつかの状態ビットはそれらへ論理1を書くことによって解除(0)されます。他の多くのAVRと異なり、CBIとSBI命令は指定ビットだけ操作し、故にこのような状態フラグを含むレジスタで使えます。CBIとSBI命令は\$00~\$1Fのレジスタだけで動作します。
- I/O指定命令INとOUTを使う時はI/Oアドレス\$00~\$3Fが使われなければなりません。LDとST命令を使ってデータ空間としてI/Oレジスタをアドレス指定する時はこれらのアドレスに\$20が加算されなければなりません。AT90PWM81/161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	-	-	-	-	-	-	-	SP8	8
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	
\$3C (\$5C)	予約									
\$3B (\$5B)	TCNT1H	タイマ/カウンタ1 上位バイト								60
\$3A (\$5A)	TCNT1L	タイマ/カウンタ1 下位バイト								
\$39 (\$59)	DACH	D/Aデータレジスタ上位バイト (DAC9~8またはDAC9~2)								137
\$38 (\$58)	DACL	D/Aデータレジスタ下位バイト (DAC7~0またはDAC1,0)								
\$37 (\$57)	SPMCSR	SPMIE	RWWSB	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	142
\$36 (\$56)	SPDR	SPI データレジスタ								112
\$35 (\$55)	MCUCR	-	-	-	PUD	RSTDIS	CKRC81	IVSEL	IVCE	46,34,27,41
\$34 (\$54)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	34
\$33 (\$53)	SMCR	-	-	-	-	SM2	SM1	SM0	SE	31
\$32 (\$52)	MSMCR	モータ停止動作制御レジスタ								
\$31 (\$51)	DWDR	モータ(デバッグWIRE) データレジスタ								125
\$30 (\$50)	予約									
\$2F (\$4F)	OCR2RAH	-	-	-	-	電力段制御器2 比較RAレジスタ 上位バイト				81
\$2E (\$4E)	OCR2RAL	電力段制御器2 比較RAレジスタ 下位バイト								
\$2D (\$4D)	ADCH	A/Dデータレジスタ上位バイト (ADC9~8またはADC9~2)								131
\$2C (\$4C)	ADCL	A/Dデータレジスタ下位バイト (ADC7~0またはADC1,0)								
\$2B (\$4B)	OCR0RAH	-	-	-	-	縮小電力段制御器 比較RAレジスタ 上位バイト				102
\$2A (\$4A)	OCR0RAL	縮小電力段制御器 比較RAレジスタ 下位バイト								
\$29 (\$49)	OCR2RBH	-	-	-	-	電力段制御器2 比較RBレジスタ 上位バイト				81
\$28 (\$48)	OCR2RBL	電力段制御器2 比較RBレジスタ 下位バイト								
\$27 (\$47)	OCR2SBH	-	-	-	-	電力段制御器2 比較SBレジスタ 上位バイト				81
\$26 (\$46)	OCR2SBL	電力段制御器2 比較SBレジスタ 下位バイト								
\$25 (\$45)	OCR0RBH	-	-	-	-	縮小電力段制御器 比較RBレジスタ 上位バイト				102
\$24 (\$44)	OCR0RBL	縮小電力段制御器 比較RBレジスタ 下位バイト								
\$23 (\$43)	OCR0SBH	-	-	-	-	縮小電力段制御器 比較SBレジスタ 上位バイト				102
\$22 (\$42)	OCR0SBL	縮小電力段制御器 比較SBレジスタ 下位バイト								
\$21 (\$41)	EIMSK	-	-	-	-	-	INT2	INT1	INT0	52
\$20 (\$40)	EIFR	-	-	-	-	-	INTF2	INTF1	INTF0	52
\$1F (\$3F)	EEARH	-	-	-	-	-	-	-	EEAR8	12
\$1E (\$3E)	EEARL	EEPROMアドレスレジスタ下位バイト (EEAR7~0)								
\$1D (\$3D)	EEDR	EEPROMデータレジスタ								12
\$1C (\$3C)	EEDR	NVMSY	EEDR	EEDR	EEDR	EEDR	EEDR	EEDR	EEDR	12
\$1B (\$3B)	GPIOR2	汎用I/Oレジスタ2								17
\$1A (\$3A)	GPIOR1	汎用I/Oレジスタ1								17
\$19 (\$39)	GPIOR0	汎用I/Oレジスタ0								17
\$18 (\$38)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	111
\$17 (\$37)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	111
\$16 (\$36)	PCTL2	PPRE21	PPRE20	PBFM20	PAOC2B	PAOC2A	PARUN2	PCCYC2	PRUN2	84
\$15 (\$35)	PCNF2	PFIFTY2	PALOCK2	PLOCK2	PMODE21	PMODE20	POP2	PCLKSEL2	POME2	82
\$14 (\$34)	PIFR2	POAC2B	POAC2A	PSEI2	PEV2B	PEV2A	PRN21	PRN20	PEOP2	87
\$13 (\$33)	PIM2	-	-	PSEIE2	PEVE2B	PEVE2A	-	PEOEPE2	PEOPE2	87
\$12 (\$32)	PCTL0	PPRE01	PPRE00	PBFM00	PAOC0B	PAOC0A	PARUN0	PCCYC0	PRUN0	103
\$11 (\$31)	PCNF0	PFIFTY0	PALOCK0	PLOCK0	PMODE01	PMODE00	POP0	PCLKSEL0	-	103
\$10 (\$30)	PIFR0	POAC0B	POAC0A	-	PEV0B	PEV0A	PRN01	PRN00	PEOP0	106
\$0F (\$2F)	PIM0	-	-	-	PEVE0B	PEVE0A	-	PEOEPE0	PEOPE0	105
\$0E (\$2E)	PORTE	-	-	-	-	-	PORTE2	PORTE1	PORTE0	51
\$0D (\$2D)	DDRE	-	-	-	-	-	DDE2	DDE1	DDE0	51
\$0C (\$2C)	PINE	-	-	-	-	-	PINE2	PINE1	PINE0	51
\$0B (\$2B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	51
\$0A (\$2A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	51
\$09 (\$29)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	51
\$08 (\$28)	ADMUX	REFS1	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	129
\$07 (\$27)	ADCSRB	ADHSM	ADNCDIS	-	ADSSEN	ADTS3	ADTS2	ADTS1	ADTS0	130
\$06 (\$26)	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	129
\$05 (\$25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	51
\$04 (\$24)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	51
\$03 (\$23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	51
\$02 (\$22)	TIFR1	-	-	ICF1	-	-	-	-	TOV1	61
\$01 (\$21)	TIMSK1	-	-	ICIE1	-	-	-	-	TOIE1	61
\$00 (\$20)	ACSR	AC3IF	AC2IF	AC1IF	-	AC3O	AC2O	AC1O	-	120

(訳注) 原書本位置の注意は前頁に移動しました。

25. 命令要約

ニーモニック	オペラント*	意味	動作	フラグ	クロック
<b>算術、論理演算命令</b>					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,I	1
NEG	Rd	2の補数	$Rd \leftarrow \$00 - Rd$	I,T,H,S,V,N,Z,C	1
SBR	Rd,K	汎用レジスタの(複数)ビット設定(1)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
CBR	Rd,K	汎用レジスタの(複数)ビット解除(0)	$Rd \leftarrow Rd \text{ AND } (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レジスタの増加(+1)	$Rd \leftarrow Rd + 1$	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	$Rd \leftarrow Rd - 1$	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レジスタのゼロとマイナス検査	$Rd \leftarrow Rd \text{ AND } Rd$	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レジスタの全0設定(=\$00)	$Rd \leftarrow Rd \text{ EOR } Rd$	I,T,H,0,0,0,I,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	$Rd \leftarrow \$FF$	I,T,H,S,V,N,Z,C	1
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 EOR V)=0なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLT	k	符号付きの $<$ で分岐	(N EOR V)=1なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	H=1なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	H=0なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	T=1なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	T=0なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	V=1なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	V=0なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	I=1なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	I=0なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ  
 b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)

注: これらの命令はAT90PWM161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
LD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上(任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上(事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
<b>ビット関係命令</b>					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	I,T,H,S,V,1,I,I,I	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLC		キャリーフラグを解除(0)	$C \leftarrow 0$	I,T,H,S,V,N,Z,0	1
SEN		負フラグを設定(1)	$N \leftarrow 1$	I,T,H,S,V,1,Z,C	1
CLN		負フラグを解除(0)	$N \leftarrow 0$	I,T,H,S,V,0,Z,C	1
SEZ		ゼロフラグを設定(1)	$Z \leftarrow 1$	I,T,H,S,V,N,1,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,0,C	1
SEI		全割り込み許可	$I \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLI		全割り込み禁止	$I \leftarrow 0$	0,T,H,S,V,N,Z,C	1
SES		符号フラグを設定(1)	$S \leftarrow 1$	I,T,H,1,V,N,Z,C	1
CLS		符号フラグを解除(0)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	$V \leftarrow 1$	I,T,H,S,1,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	$V \leftarrow 0$	I,T,H,S,0,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,1,H,S,V,N,Z,C	1
CLT		一時フラグを解除(0)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
<b>MCU制御命令</b>					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグ機能専用	I,T,H,S,V,N,Z,C	N/A

## 26. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号	外囲器	動作範囲
AT90PWM81	16	2.7~5.5V	AT90PWM81-16MN	QFN32 (注1)	範囲拡張版 (-40°C~105°C)
			AT90PWM81-16SN	SO20	
			AT90PWM81-16MF	QFN32 (注2)	範囲拡張版 (-40°C~125°C)
			AT90PWM81-16SF	SO20	
AT90PWM161	16	2.7~5.5V	AT90PWM161-16MN	QFN32 (注3)	範囲拡張版 (-40°C~105°C)
			AT90PWM161-16SN	SO20	
			AT90PWM161-16MF	QFN32 (注4)	範囲拡張版 (-40°C~125°C)
			AT90PWM161-16SF	SO20	

注: 全外囲器が鉛フリー、完全にLHF。

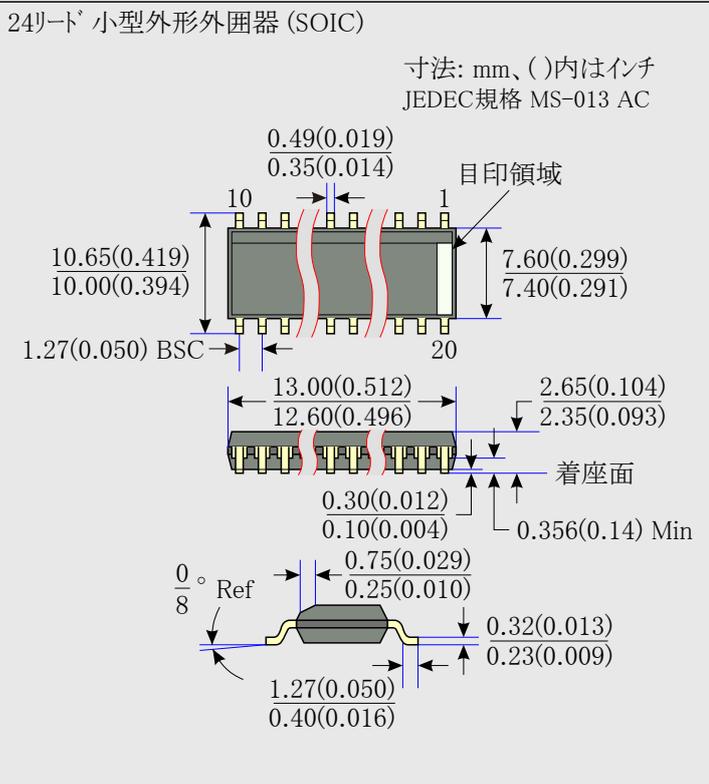
- このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。
- 部品番号はスティック(SO)またはトレイ(QFN)での出荷用です。これらのデバイスはテープとリールでも出荷できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

- この外囲器上の刻印はPWM81-MNです。
- この外囲器上の刻印はPWM81-MFです。
- この外囲器上の刻印はPWM161-MNです。
- この外囲器上の刻印はPWM161-MFです。

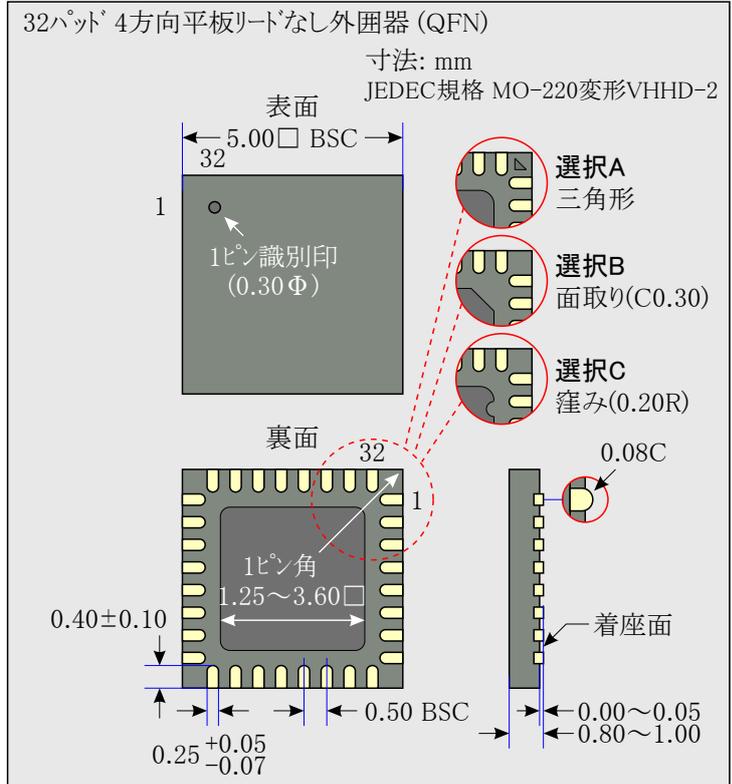
## 27. 外囲器情報

外囲器形式	
SO20	20リード、300mil本体幅、プラスチック翼状小型外形外囲器 (SOIC)
QFN32	32パッド、5×5mm本体、0.50mmピッチ、4方向平板リードなし外囲器 (QFN)

### 27.1. SQ24



### 27.2. QFN32



## 28. 障害情報

AT90PWM81改訂A 求めに応じて提供可能

• クロック切り替え禁止	81-B/C/D/E, 161-A/B
• クロック切り替えでのクリスタル用発振器制御	81-B/C/D/E, 161-A/B
• 低電圧検出(BOD)リセット禁止ヒューズ*	81-B/C/D/E, 161-A/B
• パワーダウン休止動作での過剰な消費	81-B/C/D, 161-A
• CPUクロックが切り替えられた場合のフラッシュメモリとEEPROMのプログラミング失敗	81-B/C/D
• A/D変換増幅器測定が不安定	81-B/C/D
• PSC2同期変換でA/D変換測定が異常な値を報告	81-B/C/D
• リセットでの電力段制御器(PSC)出力	81-B/C
• PSCRRBヒューズ*	161-A/B

### 1. クロック切り替えの許可と禁止 (81-B/C/D/E, 161-A/B)

“クロック元許可”または“クロック元禁止”の命令後、その命令は次のクロック制御/状態レジスタ(CLKCSR)のアクセスまで未だ活性です。CLKSELRが新しい値を書かれた場合、対応するクロックが意図せずに許可または禁止されます。

#### 対策/対処

許可または禁止の命令後、1<<CLKCCEの値でCLKCSRを書いてください。

### 2. クロック切り替えでのクリスタル用発振器制御 (81-B/C/D/E, 161-A/B)

クリスタル用発振器が活性で、クロック選択レジスタ(CLKSELR)が別のクロック元(例えばRCまたはWD)選択用の新しい値で書かれると、クリスタル用発振器の利得は正しくなくなります。

#### 対策/対処

“クロック元許可”と“クロック元切り替え”の命令後、活性なクリスタル用発振器に対応する値でCLKSELRを書き戻してください。

### 3. 低電圧検出(BOD)リセット禁止ヒューズ\* (81-B/C/D/E, 161-A/B)

BOD活性を保つことが強く推奨されます。実際に電源が低電圧になる時にRC発振器が活性なら、それは固まるかもしれません。

#### 対策/対処

BODの禁止が必須なら、リセット中のクロック元としてRC発振器を設定せず、電源が最低POR電圧(2.6V)以下の時にRC発振器が決して活性でないことを保証してください。

### 4. パワーダウン休止動作での過剰な消費 (81-B/C/D, 161-A)

パワーダウン動作に於いて、最大500μAまでの追加の消費電力が発生し得ます。

#### 対策/対処

対策はありません。

### 5. CPUクロックが切り替えられた場合のフラッシュメモリとEEPROMのプログラミング失敗 (81-B/C/D)

応用でクロック切り替えが使われる場合、メモリプログラミングはシステムクロックとして内蔵RC発振器が選ばれている時だけ可能です。

応用が内蔵RC発振器と異なるクロック元でメモリプログラミングを必要とする場合、そのクロック元へ切り替えてはなりません。

#### 対策/対処

– フラッシュメモリとEEPROMのプログラミング時に内蔵RC発振器を使ってください。

または

– クロック切り替えを使わないでください。

### 6. A/D変換増幅器測定が不安定 (81-B/C/D)

シングルエンドチャンネルから増幅器チャンネルへの切り換え時、A/D変換上位雑音が出現し得ます。

#### 対策/対処

シングルエンドから増幅器チャンネルへの切り換え後、最初のA/D変換を破棄してください。

### 7. PSC2同期変換でA/D変換測定が異常な値を報告 (81-B/C/D)

同期形態でA/D変換器を使う時に、予期せぬ追加のシングルエンド変換が不正に再始動し得ます。これは変換完了と起動事象が同時に起きた時に発生し得ます。

#### 対策/対処

対策はありません。

#### 8. リセットでの電力段制御器(PSC)出力 (81-B/C)

リセットに於いて、PSC出力はPSCヒューズ構成設定(拡張ヒューズ 81-Bのビット4)と違う値で設定されるかもしれません。

##### 対策/対処

プログラムからPSC出力の状態を初期化してください。

#### 9. PSCRRBヒューズ (161-A/B)

並列プログラミング動作でこのヒューズ ビットがプログラム(0)されると、並列とISPでの更なるプログラミングはデバイスを正しくプログラミングしません。

##### 対策/対処

ISP動作を使ってPSCRRBヒューズをプログラム(0)してください。

## 29. 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。

### 29.1. 7734改訂A

1. 初版

### 29.2. 7734改訂B

1. 互換性の利用のためにGPIO3消去
2. PB7とPD7のピン配置を入れ替え
3. CKSEL値再定義
4. クロック切り替えとクロック監視を追加
5. PSCrOUTの名前をPSCOUTRに変更
6. PSC同期でのA/D変換自動起動を改良
7. 並列プログラミングを20ピン用に更新
8. ヒューズ更新：互換性とリセットレベルに対する潜在的な衝突

### 29.3. 7734改訂C

1. ピン配置変更要求
2. 段落落ちと番号付けで多くを改善
3. 19頁：デバイスクロック任意選択
4. 113頁：BGDビット消去
5. 181,182頁：レジスタアドレス変更

### 29.4. 7734改訂D

1. ピン名AGND
2. 縮小電力段制御器(PSCR)増強分解能支援(応用要求)

### 29.5. 7734改訂E

1. 3Vでの速度が12MHz
2. ピン説明説追加(PE要求)
3. 表5-1. : 128kHz RC発振器に対するPEの機能はI/O
4. 20頁でのRC発振器の詳細
5. 25頁のクロック切り替えでの新しい警告
6. 27頁のCKRC81の詳細
7. 29頁でのパワーダウんでPSC起動元が利用不可
8. DIDR0,DIDR1の誤植
9. 自動変換でのA/D変換採取/保持時間
10. リセット中の正確なPSC入力動き
11. 既存周辺機能でPRR例を用いて更新
12. 並列プログラミング入力ピン
13. I/Oヒステリシス曲線

### 29.6. 7734改訂F

1. クロック章で全ての“パワーセーブ”を削除
2. 19頁で「校正付き内蔵RC発振器」を更新
3. 22頁の表5-9.を更新：PLLに対するSUT
4. 29頁のアイドル動作節を更新
5. 70頁で「PSC入力A部」と「PSC入力B部」の図を更新
6. 76頁で「障害動作14でのPSCn入力B対PSCn動作」の図を更新
7. 83頁で「表12-16. PSCの端とレベルの入力選択」と「表12-17. PSCの端とレベルの入力選択」を更新
8. 電力段制御器(PSC)章を整理(PSC0とPSC1のレジスタはもうありません。)

9. 縮小電力段制御器(PSCR)のレジスタとビットをrから0に改名
10. 121頁で「DIDR0 – デジタル入力禁止レジスタ0」と「DIDR1 – デジタル入力禁止レジスタ1」項を更新
11. 166頁で並列プログラミングの図、図22-5、図22-6、図22-7を更新
12. 21.7.14.項の後の「並列プログラミング特性」を消去、今や「並列プログラミング特性」は166頁

### 29.7. 7734改訂G

1. AREFピンとしてのPE3でピン配置定義更新：2頁の「ピン配置」、4頁の「ピン割り当て」表、5頁の「PE2～PE0(ポートE)/RESET/XTAL1/XTAL2」項、50頁の「ポートEの交換機能」
2. CKSEL 0111,1100,1101に関して、19頁の表5-1を更新
3. 122頁「A/D変換器部構成図」の図を更新
4. 129頁の表17-3を更新：もはや障害ではなく警告

### 29.8. 7734改訂H

1. 2頁で製品形態の表1-1を更新
2. 20頁で「工場でのRC発振器校正」項を追加
3. 34頁の「内部基準電圧」項を更新
4. 113頁の「チップ上の電圧基準と温度感知器の概要」項を更新
5. 114頁の「温度測定」項を更新
6. 117頁の「アナログ比較器構成図」を更新
7. 146頁の「ソフトウェアからの識票列読み出し」項を更新
8. 162頁の「校正付き内蔵RC発振器の精度」項を更新
9. 171頁に「クロック前置分周での消費電力予測」を追加
10. 186頁の「障害情報」章を更新

### 29.9. 7734改訂I

1. PE3入出力機能削除(AREFとADCの機能のみ)
2. 22頁の「自動再設定動作でのウォットアップ許可」と27頁の表5-12を削除
3. 20頁で「5.2.2.1. 工場でのRC発振器校正」項を更新
4. 24頁の5.3.7.項を削除
5. 113頁の「15.3. ハンドキャップ校正手順」項を削除
6. 113頁の「温度校正」を更新
7. 115頁の「15.4.3. 温度感知器2点校正」項を削除
8. 146頁の識票列アドレス指定を更新
9. DC特性を更新
10. 「障害情報」を更新

### 29.10. 7734改訂J

1. 2頁の表1-1：QFN32：32ピン
2. 4頁の表2-2：SO24とQFN20を削除
3. 20頁の5.2.2.1.項：RC発振器は125°Cで監視
4. 33頁の表7-2：BOD許可は必須
5. 98頁：AT90PWM2/3の注釈を削除
6. 162頁の表22-1：±5%での使用者校正
7. 165頁のA/D変換器特性を更新
8. 165頁にD/A変換器特性を追加

### 29.11. 7734改訂K

1. 115頁の15.4.1.項：高温検査中の温度感知器読み込みについての最後の文章を削除
2. 115頁の15.4.1.項：T式を新しいTSGAINで変更
3. 146頁の表20-5：識票列アドレス指定表を正しいアドレスで更新

## 29.12. 7734改訂L

1. 改訂E障害情報更新

## 29.13. 7734改訂M

1. 122頁の図17-1. : REFS2ビット削除
2. 168～180頁 : 代表特性を更新

## 29.14. 7734改訂N

1. 33頁 : BOD基準更新
2. 161～163頁 : 標準、アイドル、パワーダウンのICC、Vref、ウォッチドッグ特性を更新

## 29.15. 7734改訂O

1. 161,162頁 : アナログ比較器特性更新
2. 162頁 : 表22-1.(校正付き内蔵RC発振器精度)に新規行を追加
3. 165頁の表22-6. : A/D変換機特性更新

## 29.16. 7734改訂P

1. 160頁の「電気的特性」と168頁の「代表特性」を更新

## 29.17. 7734改訂Q

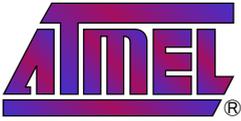
1. 全般的な更新と無数の小さな修正
2. 新しいリンクを追加
3. AT90PWM161(16Kフラッシュ、1K SRAM)の追加とそれによるデータシート更新
4. 多数のリンク先修正
5. 表2-2.の表題を「ピン割り当て説明」に修正
6. 11頁の「実装書き換え可能なプログラム用フラッシュメモリ」の1と2節で更にAT90PWM161のデータを含めるように変更
7. 16頁で2ヶ所の文字「表113」を「表20-7」で置換
8. 20頁で「表24-1」を「表22-2」に変更
9. 27頁の「ビット2 - CKRC81」の最初の注を「RC発振器許可時にだけ変更可」に修正
10. 30頁の「A/D変換器」項で「リンク先移動」を「雑音低減機能」で置換
11. 33頁の表7-2.内の値を修正
12. 38頁の表8-1.を更新
13. 45頁の表9-2.内で文字「図」をリンク「図9-5」で置換
14. 45頁の「交換ポート機能」項で表9-2.直下でリンク「表9-2」を追加
15. 46頁の「ビット4 - PUD」で最後の参照文を正しく追加/修正
16. 46頁の「MISO/ADC8/ACMP3 - PB6」で「DDB0」を「DDB6」、「PORTB0」を「PORTB6」に修正
17. 46頁の「INT1/SCK/ADC5/ACMP2 - PB5」で「DDD4」を「DDB5」、「PORT」を「PORTB5」に修正
18. 46頁で「MOSI/ADC3/ACMPM - PB4」で「DDB1」を「DDB4」、「PORTB1」を「PORTB4」に修正
19. 47頁の表9-4.と表9-5.、49頁の表9-7.と表9-8.、50頁の表9-10.の落丁情報を追加
20. 53頁の「関係レジスタ」項で「図」をリンク「図11-1.」で置換
21. 61頁の「ビット5 - ICIE1」で「XXX」文字を「表80-1」に置換
22. 62頁の「特徴」で「異常保護機能～」項目を以下、全出力をLowレベルに強制する緊急入力」に修正
23. 72頁の「入力動作の操作」で「これらの動作は表12-7.で一覧にされます。」を追加
24. 78頁の「割り込みの扱い」の先頭文で「(ヘクタ...)」文字を削除
25. 85頁での「PFRcNB」の「ビット3～0 - PRFMnx3～0」で「PSC機能記述」を「表12-7」で置換
26. 90頁の図13-2.で文字「アナログ」の綴り誤りを修正
27. 95頁の「入力動作の操作」内の文を更新
28. 95頁の表13-5.内の1行目で文字/リンク「13.9」を削除
29. 101頁の表13-8.の「説明」列の文を修正
30. 101頁の表13-9.の「説明」列の文を修正
31. 105頁での「PFRcOB」の「ビット3～0 - PRFM0x3～0」で「PSC機能記述」を「表13-5」で置換

32. 107頁の表14-1.下の注で上線付き文字を標準文字に変更
33. 117頁の図16-1.下の注の1行目で参照先を修正
34. 118頁の図16-2.を修正
35. 119頁の「ビット2~0 - ACnM2~0」で参照の「表16-4」を「表16-6」に修正 (訳注:本書では無効)
36. 121頁、131頁、181頁の「DIDR0」で「AMP3D」を「ACMP3D」で置換
37. 124頁の「前置分周と変換タイミング」で「(four XXX to be cnfirmed)」文を削除
38. 128頁の「A/D変換の結果」で文字「表82」を「表17-2」に置換
39. 132頁の図17-15.と133頁の図17-16.に対して図表題と図の位置を修正
40. 135頁の「特徴」で第5と6項目を削除
41. 135頁の「特徴」で第5項目として新しい項目を追加 (訳補:実際には第5項目の修正)
42. 150頁の表21-5.で「表113」の参照を「表20-7」に修正
43. 151頁の表21-7.内の「TBD」を「\$8B」に変更
44. 151頁の表21-7.の次の表記の第3項目をAT90PWM161を含めて拡張
45. 152頁の「信号名」項で「以降の表」を「表21-8」の参照に置換
46. 157頁「校正バリエーション」の一覧下の文字「2」を削除
47. 159頁の表21-16.で「Table XXX」を「表21-6」に置換
48. 160頁の「DC特性」の表で多数の「@25°C」を削除
49. 161頁の「DC特性」の表で105°Cを記述する行を削除
50. 162頁の表22-2.で項目名の「校正精度」を「精度」に変更
51. 163頁の表22-4.で注1の文「CPUコア,PSC...」を削除
52. 183頁の25.章の「分岐命令」にJMPとCALLの命令を追加
53. 185頁の「注文情報」を更新
54. 186頁の「障害情報」の一覧にAT90PWM161改訂Aを追加
55. 186頁の「障害情報」にAT90PWM161改訂Aの障害情報を追加
56. 186頁の「障害情報」にAT90PWM161改訂Bの障害情報を追加
57. Atmelジャパンの住所を変更

## 目次

特徴	1	11.8. 16ビット タイマ/カウンタ1用レジスタ	59
1. 製品形態	2	12. 電力段制御器 (PSCn)	62
2. ピン配置	2	12.1. 特徴	62
2.1. ピン説明	5	12.2. 概要	62
3. AVR CPU コア	6	12.3. 電力段制御器(PSC)説明	63
3.1. 序説	6	12.4. 信号説明	64
3.2. 構造概要	6	12.5. 機能説明	65
3.3. ALU (Arithmetic Logic Unit)	6	12.6. 値の更新	67
3.4. ステータス レジスタ	7	12.7. 増強分解能	67
3.5. 汎用レジスタ ファイル	8	12.8. 電力段制御器(PSC)入力	69
3.6. スタック ポインタ	8	12.9. PSC入力動作1: 信号停止、 反対側沈黙時間へ飛び、待機	73
3.7. 命令実行タイミング	9	12.10. PSC入力動作2: 信号停止、 反対側沈黙時間実行、待機	73
3.8. リセットと割り込みの扱い	9	12.11. PSC入力動作3: 信号停止、 障害有効間反対側沈黙時間実行	74
4. メモリ	11	12.12. PSC入力動作4: タイミング変更なしで 出力非活動	74
4.1. 実装書き換え可能なプログラム用フラッシュ メモリ	11	12.13. PSC入力動作5: 信号停止、 沈黙時間挿入	75
4.2. データ用SRAMメモリ	11	12.14. PSC入力動作6: 信号停止、 反対側沈黙時間へ飛び、待機	75
4.3. データ用EEPROMメモリ	12	12.15. PSC入力動作7: PSC停止、 ソフトウェア操作待機	75
4.4. ヒューズ ビット	14	12.16. PSC入力動作8: 端再起動PSC	75
4.5. I/Oメモリ (レジスタ)	17	12.17. PSC入力動作9: 周波数固定 端再起動PSC	76
4.6. 汎用I/Oレジスタ	17	12.18. PSC入力動作14: 周波数固定 端再起動PSC、出力非活動	76
5. システム クロックとクロック選択	18	12.19. PSC2出力	77
5.1. クロック系統とその配給	18	12.20. アナログ同期化	78
5.2. クロック元	19	12.21. 割り込みの扱い	78
5.3. 動的クロック切り替え	23	12.22. PSCの同期化	78
5.4. システム クロック前置分周器	25	12.23. PSCクロック元	79
5.5. クロック関係レジスタ	26	12.24. 割り込み	79
6. 電力管理と休止形態	29	12.25. 電力段制御器(PSC)用レジスタ	80
6.1. 休止形態種別	29	12.26. PSC2用特定レジスタ	86
6.2. アイドル動作	29	12.27. 電力段制御器(PSC)割り込み用レジスタ	87
6.3. A/D変換雑音低減動作	29	13. 縮小電力段制御器 (PSCR)	89
6.4. パワーダウン動作	29	13.1. 特徴	89
6.5. スタンバイ動作	30	13.2. 概要	89
6.6. 電力削減(電力削減レジスタ)	30	13.3. 縮小電力段制御器(PSCR)説明	90
6.7. 消費電力の最小化	30	13.4. 信号説明	90
6.8. 電力管理用レジスタ	31	13.5. 機能説明	91
7. システム制御とリセット	32	13.6. 値の更新	93
7.1. システム制御概要	32	13.7. 増強分解能	93
7.2. システム制御用レジスタ	34	13.8. 縮小電力段制御器(PSCR)入力	93
7.3. 内部基準電圧	34	13.9. PSCR入力動作1: 信号停止、 反対側沈黙時間へ飛び、待機	96
7.4. ウォッチドッグ タイマ	35	13.10. PSCR入力動作2: 信号停止、 反対側沈黙時間実行、待機	96
8. 割り込み	38	13.11. PSCR入力動作3: 信号停止、 障害有効間反対側沈黙時間実行	97
8.1. AT90PWM81/161の割り込みベクタ	38	13.12. PSCR入力動作4: タイミング変更なしで 出力非活動	97
9. 入出力ポート	42	13.13. PSCR入力動作5: 信号停止、 沈黙時間挿入	98
9.1. 序説	42	13.14. PSCR入力動作6: 信号停止、 反対側沈黙時間へ飛び、待機	98
9.2. 標準デジタル入出力としてのポート	42	13.15. PSCR入力動作7: PSCR停止、 ソフトウェア操作待機	98
9.3. 交換ポート機能	45		
9.4. I/Oポート用レジスタ	51		
10. 外部割り込み	52		
11. 縮小16ビット タイマ/カウンタ1	53		
11.1. 概要	53		
11.2. 16ビット レジスタのアクセス	54		
11.3. タイマ/カウンタのクロック	56		
11.4. 計数器部	56		
11.5. 捕獲入力部	57		
11.6. 動作種別	58		
11.7. タイマ/カウンタのタイミング	59		

13.16.	PSCR入力動作8: 端再起動PSCR	98	21.2.	ヒューズビット	150
13.17.	PSCR入力動作9: 周波数固定 端再起動PSCR	99	21.3.	識票バイト	151
13.18.	PSCR入力動作14: 周波数固定 端再起動PSCR、出力非活動	99	21.4.	校正値バイト	151
13.19.	アナログ同期化	100	21.5.	並列プログラミング	152
13.20.	割り込みの扱い	100	21.6.	並列プログラミング手順	153
13.21.	PSCクロック元	100	21.7.	直列プログラミング	158
13.22.	割り込み	101	21.8.	直列プログラミング手順	158
13.23.	縮小電力段制御器(PSCR)用レジスタ	101	22.	電気的特性	160
14.	SPI (直列周辺インターフェース)	107	22.1.	絶対最大定格	160
14.1.	特徴	107	22.2.	DC特性	160
14.2.	概要	107	22.3.	クロック特性	162
14.3.	SSピンの機能	109	22.4.	最高速と動作電圧	163
14.4.	データ転送形式	110	22.5.	PLL特性	163
14.5.	SPI用レジスタ	111	22.6.	SPI タイミング特性	164
15.	電圧基準と温度感知器	113	22.7.	A/D変換器特性	165
15.1.	特徴	113	22.8.	D/A変換器特性	165
15.2.	チップ上の電圧基準と温度感知器の概要	113	22.9.	並列プログラミング特性	166
15.3.	電圧基準電圧と温度感知器用レジスタ	113	23.	代表特性	168
15.4.	温度測定	114	23.1.	活動動作消費電流	168
16.	アナログ比較器	116	23.2.	アイドル動作消費電流	169
16.1.	特徴	116	23.3.	パワーダウン動作消費電流	171
16.2.	概要	116	23.4.	ピンプルアップ	171
16.3.	アナログ比較器とA/D変換器間の共用ピン	118	23.5.	ピンHigh出力電圧	173
16.4.	アナログ比較器用レジスタ説明	118	23.6.	ピンLow出力電圧	174
17.	A/D変換器	122	23.7.	ピン閾値	175
17.1.	特徴	122	23.8.	低電圧検出器(BOD)閾値	175
17.2.	操作	123	23.9.	アナログ基準電圧	176
17.3.	変換の開始	123	23.10.	内部発振器周波数	177
17.4.	前置分周と変換タイミング	124	23.11.	リセット消費電流	178
17.5.	チャンネル変更と基準電圧選択	125	24.	レジスタ要約	181
17.6.	雑音低減機能	126	25.	命令要約	183
17.7.	A/D変換の結果	128	26.	注文情報	185
17.8.	A/D変換用レジスタ説明	129	27.	外圍器情報	185
17.9.	増幅器	132	28.	障害情報	186
17.10.	増幅器制御レジスタ	134	29.	データシート改訂履歴	188
18.	D/A変換器	135			
18.1.	特徴	135			
18.2.	操作	135			
18.3.	変換の開始	135			
18.4.	D/A変換用レジスタ説明	136			
19.	デバッグWIRE内蔵デバッグ機能	138			
19.1.	特徴	138			
19.2.	概要	138			
19.3.	物理インターフェース	138			
19.4.	ソフトウェア中断点	138			
19.5.	デバッグWIREの制限	138			
19.6.	デバッグWIRE関連のI/Oメモリ内レジスタ	138			
20.	ブートローダ支援 - RWW 自己プログラミング	139			
20.1.	特徴	139			
20.2.	フラッシュメモリの応用領域とブートローダ領域	139			
20.3.	書き中に読みが可能な領域と不能な領域	139			
20.4.	ブートローダ施錠ビット	141			
20.5.	ブートローダプログラムへの移行	141			
20.6.	自己プログラミングでのフラッシュアドレス指定	143			
20.7.	フラッシュメモリの自己プログラミング	143			
21.	メモリプログラミング	149			
21.1.	プログラムメモリとデータメモリ用施錠ビット	149			



#### *Atmel Corporation*

2325 Orchard Parkway  
San Jose, CA 95131  
USA  
TEL (+1)(408) 441-0311  
FAX (+1)(408) 487-2600  
[www.atmel.com](http://www.atmel.com)

#### *Atmel Asia Limited*

Unit 01-5 & 16, 19F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
HONG KONG  
TEL (+852) 2245-6100  
FAX (+852) 2722-1369

#### *Atmel Munich GmbH*

Business Campus  
Parking 4  
D-85748 Garching b. Munich  
GERMANY  
TEL (+49) 89-31970-0  
FAX (+49) 89-3194621

#### *Atmel Japan*

141-0032 東京都品川区  
大崎1-6-4  
新大崎勸業ビル 16F  
アトメル ジャパン合同会社  
TEL (+81)(3)-6417-0300  
FAX (+81)(3)-6417-0370

© 2012 Atmel Corporation. 不許複製

Atmel®、Atmelロゴとそれらの組み合わせ、それとAVR®、AVR®ロゴ、AVR Studio®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

**お断り:** 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイトに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、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© HERO 2022.

本データシートはAtmelの英語版データシート AT90PWM81/161(Rev.7734Q-02/12)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。必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。

原書に対して若干構成が異なるため、一部の節/項、図表番号が異なります。