

特徴

■ 高性能、低消費AVR® 8ビット マイクロ コントローラ

■ RISC構造

- 強力な128命令(多くは1周期で実行)
- 32個の1バイト長汎用レジスタ
- 完全なステディック動作
- 16MHz時、16MIPSに達する高速動作
- 2周期実行の乗算命令

■ データメモリと不揮発性プログラムメモリ

- 実装自己書き換え可能な8Kバイト(4K語)フラッシュメモリ内蔵(10,000回の書き換えが可能)
- 個別施錠ビットを持つ任意のブートコード領域
 - チップ内ブートプログラムによる実装書き換え
 - 真の書き込み中の読み出し動作
- 512バイトのEEPROM(100,000回の書き換えが可能)
- 512バイトの内蔵SRAM
- 64Kバイトまでの任意外部メモリ空間
- ソフトウェア保護用の設定可能な施錠機能

■ 内蔵周辺機能

- 独立した前置分周器、比較機能付き1つの8ビットタイマ/カウンタ
- 独立した前置分周器、比較、捕獲機能付き1つの16ビットタイマ/カウンタ
- 3つのPWM出力
- 設定可能な直列USART
- 主装置/従装置動作SPI直列インターフェース
- 設定可能な専用発振器付きウォッチドッグタイマ
- アナログ比較器

■ 特殊マイクロコントローラ機能

- 電源ONリセット回路と設定可能な低電圧検出器(BOD)
- 校正可能な内蔵RC発振器
- 外部及び内部の割り込み
- アイドル、パワーダウン、スタンバイの3つの低消費動作
- ヒューズでのAT90S4414/8515互換動作

■ I/Oと外周器

- 35ビットの設定可能なI/O
- 40ピンPDIP、44リードTQFP、PLCC、44パッドQFN/MLF

■ 動作電圧

- 2.7~5.5V (ATmega8515L)
- 4.5~5.5V (ATmega8515)

■ 動作速度

- 0~8MHz (ATmega8515L)
- 0~16MHz (ATmega8515)

本書は一般の方々の便宜のため有志により作成されたもので、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。



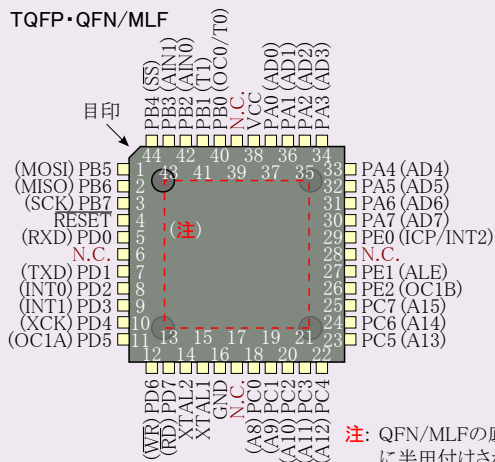
8ビット **AVR®**
マイクロコントローラ
実装書き換え可能な
8Kバイト
フラッシュメモリ内蔵

ATmega8515
ATmega8515L

ピン配置

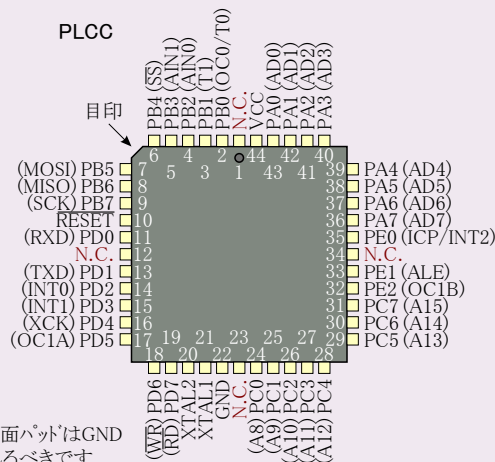
PDIP	
(OC0/T0) PB0	1
(T1) PB1	2
(AIN0) PB2	3
(AIN1) PB3	4
(SS) PB4	5
(MOSI) PB5	6
(MISO) PB6	7
(SCK) PB7	8
RESET	9
(RXD) PD0	10
(TXD) PD1	11
(INT0) PD2	12
(INT1) PD3	13
(XCK) PD4	14
(OC1A) PD5	15
(WR) PD6	16
(RD) PD7	17
XTAL2	18
XTAL1	19
GND	20
VCC	40
PA0 (AD0)	39
PA1 (AD1)	38
PA2 (AD2)	37
PA3 (AD3)	36
PA4 (AD4)	35
PA5 (AD5)	34
PA6 (AD6)	33
PA7 (AD7)	32
PE0 (ICP/INT2)	31
PE1 (ALE)	30
PE2 (OC1B)	29
PC7 (A15)	28
PC6 (A14)	27
PC5 (A13)	26
PC4 (A12)	25
PC3 (A11)	24
PC2 (A10)	23
PC1 (A9)	22
PC0 (A8)	21

TQFP・QFN/MLF



注: QFN/MLFの底面パッドはGNDに半田付けされるべきです。

PLCC



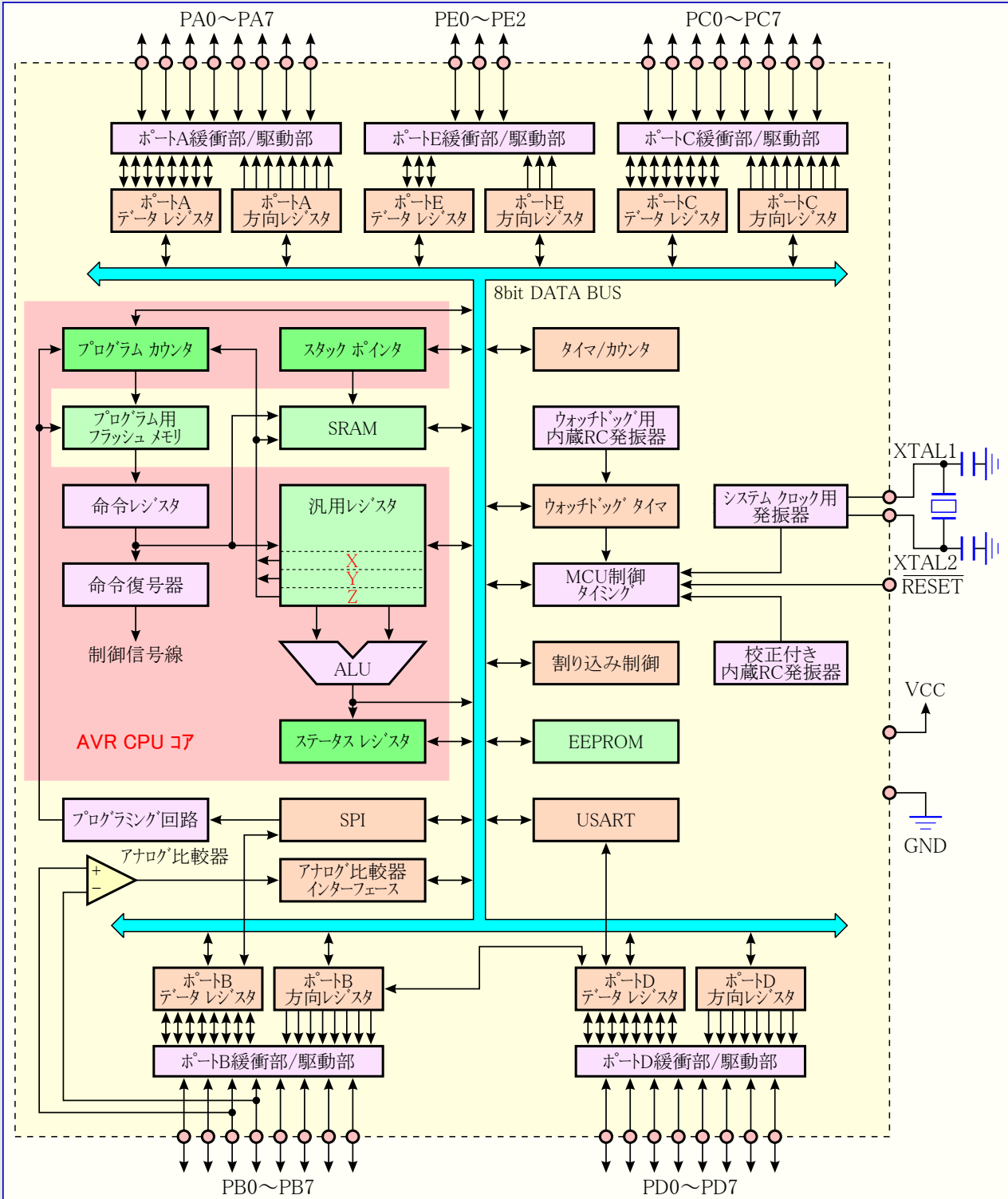
Rev. 2512K-01/10, 2512KJ7-11/21

概要

ATmega8515はAVR RISC構造の低消費CMOS 8ビットマイクロコントローラです。1周期で実行する強力な命令はMHzあたり1MIPSにも達し、実行速度対電力消費の最適化が容易に行えます。

構成図

図2. ATmega8515構成図



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

ATmega8515は次の特徴、書き込み中読める能力を持つ8Kバイトの実装書き換え可能なフラッシュメモリと512バイトのEEPROM、512バイトのSRAM、外部メモリインターフェース、35本の汎用入出力線、32個の汎用レジスタ、比較動作も含む柔軟性のある2つのタイマ/カウンタ、内部及び外部割り込み、設定変更可能な直列USART、設定変更可能な内部発振器付きウォッチドッグタイマ、SPI直列ポート、ソフトウェアで選択できる3つの低消費動作機能を提供します。アイドル動作では動作を停止しますが、SRAM、タイマ/カウンタ、SPIポート、割り込み機能は有効で、動作を継続します。パワーダウン動作ではレジスタの内容は保護されますが、発振器が停止するため、以降のハートウェアリセットか外部割り込みまで他の全機能を禁止(無効に)します。スタンバイ動作では水晶発振子/セラミック振動子用発振器が動作し、一方デバイスのその他は休止します。これは低消費電力と非常に速い起動の組み合わせを許します。

本デバイスはAtmelの高密度不揮発性メモリ技術を使って製造されています。内蔵の実装書き換え(ISP)可能なプログラム用フラッシュメモリは規定の不揮発性メモリ書き込み器、SPI直列インターフェース経由、AVRコア上ブートプログラムの実行によって再書き込みができます。ブートプログラムは応用領域フラッシュメモリ内の応用プログラムの読み込みにどのインターフェースでも使えます。ブート領域フラッシュメモリ内のプログラムは真の「書き込み中の読み出し可」動作により、応用領域フラッシュメモリ更新中も実行を継続します。モノリシックチップ上の実装書き換え可能なフラッシュメモリと、8ビットRISC型CPUの組み合わせによるAtmel ATmega8515は多くの組み込み制御の応用に対して高度な柔軟性と対費用効果をもたらす強力なマイクロコントローラです。

ATmega8515 AVRはCコンパイラ、マクロアセンブラ、デバッガ、シミュレータ、インサーキットエミュレータ、評価キットを含む完全なプログラム及びシステム開発ツールで支援されます。

お断り

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

ATmega8515とAT90S4414/8515の互換性

ATmega8515はAT90S4414/8515の全機能を提供します。加えて様々な新機能が追加されます。多くの場合でATmega8515は以前のAT90S4414/8515と互換です。けれども2つのマイクロコントローラ間にいくつかの非互換性が存在します。この問題を解決するために、S8515Cヒューズをプログラム(0)することによってAT90S4414/8515互換動作が選択できます。ATmega8515はAT90S4414/8515と100%ピン互換で、現状のプリント回路基板上のAT90S4414/8515を置換できます。けれどもヒューズビットの位置と電気的特性は2つのデバイス間で異なります。

AT90S4414/8515互換動作

S8515Cヒューズのプログラム(0)は次の機能を変更します。

- ウォッチドッグ計時時間変更に対する時間制限手順が禁止されます。詳細については31頁の「ウォッチドッグタイマ設定変更の時間制限手順」をご覧ください。
- USART受信レジスタの2重緩衝が禁止されます。詳細については89頁の「UARTとの互換性」をご覧ください。
- PORTE2、PORTE1は出力として設定され、PORTE0は入力として設定されます。

ピン概要

VCC

デジタル電源ピン。

GND

接地ピン。

PA7～PA0 (ポートA)

ポートAは(ビット毎に選択される)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートAの出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。PA0～PA7ピンが入力として使われ、外部的にLowへ引き込まれるとき、内蔵プルアップ抵抗が活性(有効)なら、吐き出し電流が流れます。リセット条件が有効になると、クロックが動いていなくてもポートAピンはHi-Zにされます。

ポートAは[40頁](#)で一覧されるATmega8515の様々な特殊機能も扱います。

PB7～PB0 (ポートB)

ポートBは(ビット毎に選択される)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートB出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力の時にプルアップ抵抗が活性(有効)なら、外部的にLowへ引き込まれたポートBピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動いていなくてもポートBピンはHi-Zにされます。

ポートBは[41頁](#)で一覧されるATmega8515の様々な特殊機能も扱います。

PC7～PC0 (ポートC)

ポートCは(ビット毎に選択される)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートCの出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力の時にプルアップ抵抗が活性(有効)なら、外部的にLowへ引き込まれたポートCピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動いていなくてもポートCピンはHi-Zにされます。

PD7～PD0 (ポートD)

ポートDは(ビット毎に選択される)内蔵プルアップ抵抗付き8ビット双方向入出力ポートです。ポートDの出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力の時にプルアップ抵抗が活性(有効)なら、外部的にLowへ引き込まれたポートDピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動いていなくてもポートDピンはHi-Zにされます。

ポートDは[44頁](#)で一覧されるATmega8515の様々な特殊機能も扱います。

PE2～PE0 (ポートE)

ポートEは(ビット毎に選択される)内蔵プルアップ抵抗付き3ビット双方向入出力ポートです。ポートEの出力緩衝部は高い吐き出しと吸い込み両方の能力の対称駆動特性です。入力の時にプルアップ抵抗が活性(有効)なら、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、クロックが動いていなくてもポートEピンはHi-Zにされます。

ポートEは[46頁](#)で一覧されるATmega8515の様々な特殊機能も扱います。

RESET

リセット入力。最小パルス幅より長いこのピンのLowレベルはクロックが動いていなくてもリセットを生成します。最小パルス幅は[27頁の表18](#)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。

XTAL1

発振器反転増幅器への入力と内部クロック操作回路への入力。

XTAL2

発振器反転増幅器からの出力。

資料

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

コード例について

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

AVR CPU コア

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。

構造概要

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

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

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

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

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

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

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

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

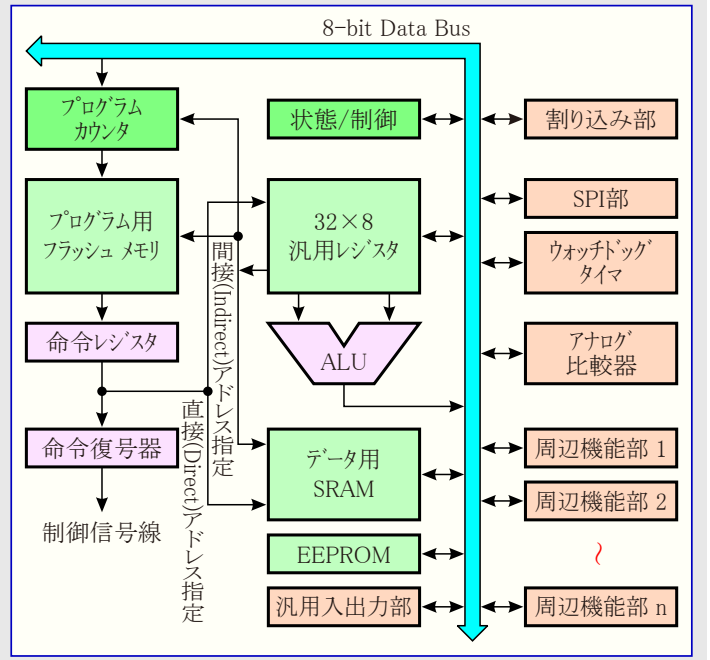
柔軟な割り込み部にはI/O空間の各制御レジスタとステータスレジスタ(SREG)の特別な全割り込み許可(I)ビットがあります。全ての割り込みは割り込みベクタ表に個別の割り込みベクタを持ちます。割り込みには割り込みベクタ表の位置に従う優先順があります。下位側割り込みベクタアドレスが高い優先順位です。

I/Oメモリ空間は制御レジスタ、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。

ALU (Arithmetic Logic Unit)

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

図3. AVR MCU構造



■ 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「[命令一式参考書](#)」で詳述したように、全てのALU操作後、更新されることに注目してください。これは多くの場合でそれ用の比較命令使用の必要をなくし、高速でより少ないコードに帰着します。

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

AVRのステータスレジスタ(SREG)は次のように定義されます。

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

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

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

■ ビット6 - T : ビット変数 (Bit Copy Storage)

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

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

ハーフキャリー(H)フラグはいくつかの算術操作でのハーフキャリーを示します。ハーフキャリーはBCD演算で有用です。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。

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

図4. AVR CPU 汎用レジスタ構成図

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

Xレジスタ, Yレジスタ, Zレジスタ

R26～R31レジスタには通常用途の使用にいくつかの追加機能があります。これらのレジスタはデータ空間の間接アドレス指定用の16ビットアドレスポインタです。3つのX,Y,Z間接アドレスレジスタは図5.で記載したように定義されます。

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

図5. 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
	15	YH (上位)	YL (下位)	0
Y レジスタ	7	R29 (\$1D)	0	7
	15	ZH (上位)	ZL (下位)	0
Z レジスタ	7	R31 (\$1F)	0	7

■ 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(誤注: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\$60以上を指示するように設定されなければなりません。スタック ポインタはPUSH命令でデータがスタックに格納されると-1され、サブルーチン呼び出しや割り込みで戻りアドレスがスタックに格納されると-2されます。スタック ポインタはPOP命令でデータがスタックから引き出されると+1され、サブルーチンからの復帰(RET)命令や割り込みからの復帰(RETI)命令でアドレスがスタックから引き出されると+2されます。

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

ビット	15	14	13	12	11	10	9	8	
\$3E (\$5E)	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$3D (\$5D)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

命令実行タイミング

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

図6.はハーバート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、機能対電源部についての好結果と対応するMHzあたり1MIPSを達成するための基本的なパイプラインの概念です。

図7.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 オペラントを使うALU操作が実行され、その結果が転送先レジスタへ書き戻されます。

図6. 命令の取得と実行の並列動作

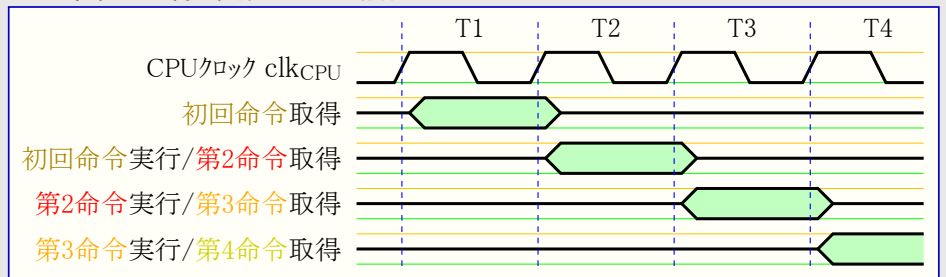
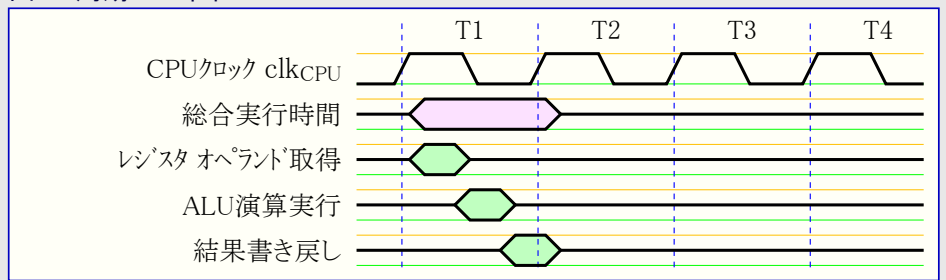


図7. 1周期ALU命令



リセットと割り込みの扱い

AVRは多くの異なる割り込み元を提供します。これらの割り込みと独立したリセット ベクタ各々はプログラム メモリ空間内に独立したプログラム ベクタを持ちます。全ての割り込みは割り込みを許可するために、ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない個別の許可ビットを割り当てられます。BLB02またはBLB12ブート施錠ビットがプログラム(0)されると、プログラム カウンタ値によっては割り込みが自動的に禁止されるかもしれません。この特質はソフトウェア保護を改善します。詳細については116頁の「メモリ フロギング」章をご覧ください。

既定でのプログラム メモリ空間の最下位アドレスはリセットと割り込みのベクタとして定義されます。ベクタの完全な一覧は32頁の「割り込み」で示されます。この一覧は各種割り込みの優先順位も決めます。下位側アドレスがより高い優先順位です。リセットが最高優先順位で次が外部割り込み要求0(INT0)です。割り込みベクタは一般割り込み制御レジスタ(GICR)の割り込みベクタ選択(IVSEL)ビットの設定(1)によってブートフラッシュ領域先頭へ移動できます。より多くの情報については32頁の「割り込み」を参照してください。リセット ベクタもBOOTRSTヒューズのプログラム(0)によりブートフラッシュ領域先頭へ移動できます。107頁の「ブート ロード支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

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

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

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

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

ステータスレジスタ(SREG)は割り込みルーチンへ移行時の保存も、復帰時の再設定も自動的に行われな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。

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。この増加は選択した休止形態からの起動時間に加えてです。

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

AVR ATmega8515のメモリ

この項はATmega8515の各種メモリを記述します。AVR構造にはプログラムメモリ空間とデータメモリ空間の2つの主なメモリ空間があります。加えてATmega8515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。

実装自己書き換え(ISP: In-System Program)可能なプログラム用フラッシュメモリ

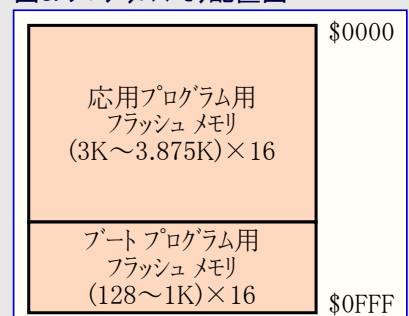
ATmega8515はプログラム保存用に実装書き換え可能な8Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、このフラッシュメモリは4K×16ビットとして構成されます。ソフトウェア保護のため、フラッシュプログラムメモリ空間はブートプログラム領域と応用プログラム領域の2つに分けられます。

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。ATmega8515のプログラムカウンタ(PC)は12ビット幅、従って4Kプログラムメモリ位置のアドレス指定です。ブートプログラム領域の操作と関係するソフトウェア保護用ブート施錠ビットは、107頁の「**ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング**」の詳細で記述されます。116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8頁の「**命令実行タイミング**」で示されます。

図8. プログラムメモリ配置図



データ用SRAM メモリ

図9.はATmega8515のデータ メモリ構成方法を示します。

下位608データ メモリ位置はレジスタ ファイル、I/Oメモリ、データ用内蔵SRAMに充てます。最初の96位置はレジスタ ファイルとI/Oメモリに充て、次の512位置はデータ用内蔵SRAMに充てます。

任意の外部データSRAMがATmega8515で使えます。このSRAMは64Kアドレス空間内の残りのアドレス領域を占有します。この領域は内蔵SRAMに続くアドレスから始まります。レジスタ ファイル、I/O領域、(訳注: 存在しない原文の拡張I/O領域を削除)、内蔵SRAMが最下位608バイトを占有するので、64K(65536)バイトの外部メモリ使用時、64928バイトの外部メモリが利用可能です。外部メモリ配置の利用法の詳細については14頁の「外部メモリ インターフェース」をご覧ください。

アドレスが内部データ メモリ領域を越えるSRAMメモリ空間をアクセスするとき、外部データSRAMは内部データ メモリ アクセスと同じ命令を使ってアクセスされます。内部データ メモリがアクセスされるとき、読みと書きのストロブ ピン(PD7とPD6)はそのアクセス周期中全てで無効です。外部SRAM操作はMCU制御レジスタ(MCUCR)の外部SRAM許可(SRE)ビットの設定(1)によって許可されます。

外部SRAMのアクセスは内蔵SRAMのアクセスに比べ1バイト毎に1追加クロック周期を要します。これは(外部SRAMへの)LD,ST,LDS,STS,LDD,STD,PUSH,POP命令が1追加クロック周期を要することを意味します。スタックが外部SRAMに配置されると、割り込みやサブルーチンの呼び出し/復帰時、2バイトのプログラム カウンタ(PC)が保存(プッシュ)/復帰(ポップ)され、そして内部メモリ アクセスでのパイプラインの優位性がなくなるため、2+1クロック周期の追加を要します。待ち状態付きで外部メモリ インターフェースが使われると、1バイトの外部アクセスは1,2,3待ち状態に対し、各々(内部SRAMアクセスに対して)2,3,4追加クロック周期要します。割り込みやサブルーチンの呼び出しと復帰は1,2,3待ち状態に対し、命令一式手引書で示される値よりも5,7,9クロック周期多く必要です。

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。

ATmega8515の32個の汎用レジスタ、64個のI/Oレジスタ、512バイトのデータ用内蔵SRAMは、これら全てのアドレス指定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7頁の「汎用レジスタ ファイル」で記述されます。

データ 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10.で記載されるように2clk_{CPU}周期で実行されます。

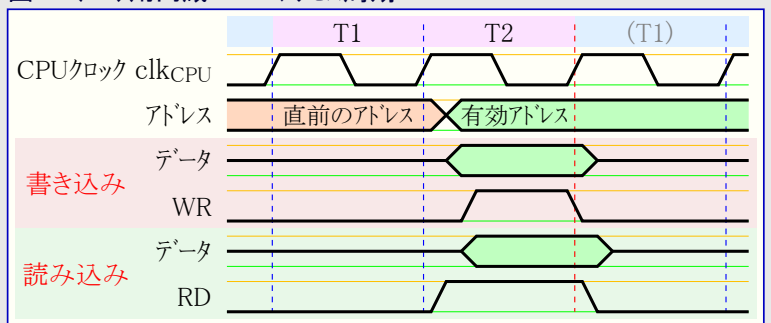
(訳注) 内蔵SRAMのアクセスを含む代表的な命令はT1,T2の2周期で実行され、T1で対象アドレスを取得/(算出)/確定し、T2で実際のアクセスが行われます。後続する(T1)は次の命令のT1です。

図9. データ空間とSRAMの配置

		アドレス
レジスタ ファイル (32×8)	R0～R31	\$0000～\$001F
I/O レジスタ (64×8)	\$00～\$3F	\$0020～\$005F
内蔵 SRAM (512×8)	\$0060 ～ \$025F	\$0060 ～ \$025F
外部 SRAM (0～63.4K×8)	\$0260 ～ \$FFFF	\$0260 ～ \$FFFF

注: 赤字はI/Oアドレス

図10. データ用内蔵SRAMアクセス周期



データ用EEPROMメモリ

ATmega8515は512バイトのデータEEPROMを含みます。それは単一バイトが読み書きできる分離したデータ空間として構成されます。EEPROMは最低100,000回の消去/書き込み回数の耐久性があります。CPUとEEPROM間のアクセスは以降の[EEPROMアドレスレジスタ](#)、[EEPROMデータレジスタ](#)、[EEPROM制御レジスタ](#)で詳細に記述されます。

[116ページの「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EEPROMプログラミングの詳細な記述を含みます。

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[12ページの表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[13ページの「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。

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

ビット	15	14	13	12	11	10	9	8	
\$1F (\$3F)	—	—	—	—	—	—	—	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	
\$1E (\$3E)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

■ ビット15~9 - Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

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

■ EEPROMデータレジスタ (EEPROM Data Register) EEDR

ビット	7	6	5	4	3	2	1	0	
\$1D (\$3D)	(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。

■ EEPROM制御レジスタ (EEPROM Control Register) EECR

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

■ ビット7~4 - Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

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

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

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

■ ビット1 – EEWWE : EEPROM書き込み許可 (EEPROM Write Enable)

EEPROM書き込み許可信号(EEMWE)はEEPROMへの書き込みスローブです。アドレスとデータが正確に設定されると、EEPROMへその値を書くためにEEWEビットは1を書かれなければなりません。論理1がEEWEへ書かれる前にEEPROM主書き込み許可(EEMWE)ビットは1を書かれなければならず、さもなければ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(EEMWE)ビットが0になるまで待ちます。
2. SPM制御レジスタ(SPMCR)のSPM操作許可(SPMEN)ビットが0になるまで待ちます。
3. 今回のEEPROMアドレスをEEPROMアドレスレジスタ(EEAR)に書きます。(任意、省略可)
4. 今回のEEPROMデータをEEPROMデータレジスタ(EEDR)に書きます。(任意、省略可)
5. EEPROM制御レジスタ(EECR)のEEPROM主書き込み許可(EEMWE)ビットに1を、EEPROM書き込み許可(EEMWE)ビットに0を同時に書きます。
6. EEMWEビット設定後4クロック周期内にEEPROM書き込み許可(EEMWE)ビットへ論理1を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMはプログラム(書き込み)ができません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを検査しなければなりません。2.はソフトウェアがフラッシュメモリをプログラム(書き込み)することをCPUに許すブートローダを含む場合だけ関係します。フラッシュメモリが決してCPUによって更新されないなら、2.は省略できます。ブートプログラミングについての詳細に関しては107頁の「ブートローダ支援 – 書き込み中読み出し可能な自己プログラミング」をご覧ください。

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

書き込みアクセス時間が経過されると、EEPROM書き込み許可(EEMWE)ビットはハードウェアによって解除(0)されます。使用者ソフトウェアは次バイトを書く前にこのビットをポーリングし、0を待つことができます。EEWE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ビットは論理1を書かれなければなりません。EEPROM読み出しアクセスは(その)1命令で行われ、要求したデータは直ちに利用できます。EEPROMが読まれるとき、次の命令が実行される前にCPUは4周期停止されます。

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

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

表1. EEPROM書き込み時間

項目	校正付き内蔵RC 発振器周期数 (注)	Typ
EEPROM書き込み(CPU)	8448	8.5ms

注: CKSELヒューズ設定と無関係に1MHzが使われます。

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

アセンブリ言語プログラム例

```
EEPROM_WR:  SBIC    EECR, EEWI          ;EEPROM書き込み完了ならばスキップ
            RJMP    EEPROM_WR         ;以前のEEPROM書き込み完了まで待機
;
            OUT     EEARH, R18         ;EEPROMアドレス上位バイト設定
            OUT     EEARL, R17         ;EEPROMアドレス下位バイト設定
            OUT     EEDR, R16         ;EEPROM書き込み値を設定
            SBI     EECR, EEMWE        ;EEPROM主書き込み許可ビット設定
            SBI     EECR, EEWI        ;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<<EEWE));          /* 以前のEEPROM書き込み完了まで待機 */
    EEAR = uiAddress;                  /* EEPROMアドレス設定 */
    EEDR = ucData;                    /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMWE);               /* EEPROM主書き込み許可 */
    EECR |= (1<<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, EEWI          ;EEPROM書き込み完了ならばスキップ
            RJMP    EEPROM_RD         ;以前のEEPROM書き込み完了まで待機
;
            OUT     EEARH, R18         ;EEPROMアドレス上位バイト設定
            OUT     EEARL, R17         ;EEPROMアドレス下位バイト設定
            SBI     EECR, EERE        ;EEPROM読み出し開始(読み込み許可ビット設定)
            IN      R16, EEDR          ;EEPROM読み出し値を取得
            RET                        ;呼び出し元へ復帰
```

C言語プログラム例

```
unsigned char EEPROM_read(unsigned int uiAddress)
{
    while(EECR & (1<<EEWE));          /* 以前のEEPROM書き込み完了まで待機 */
    EEAR = uiAddress;                  /* EEPROMアドレス設定 */
    EECR |= (1<<EERE);               /* EEPROM読み出し開始 */
    return EEDR;                      /* EEPROM読み出し値を取得,復帰 */
}
```

パワーダウン休止動作中のEEPROM書き込み

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

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。

I/O メモリ (レジスタ)

ATmega8515のI/O空間定義は156頁の「[レジスタ要約](#)」で示されます。

ATmega8515の全てのI/Oと周辺機能はI/O空間に配置されます。I/O領域は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IN命令とOUT命令によってアクセスされます。アドレス範囲\$00～\$1F内のI/OレジスタはSBI命令とCBI命令の使用で直接的にビットアクセス可能です。これらのレジスタではSBISとSBIC命令の使用によって単一ビット値が検査できます。より多くの詳細については「[命令要約](#)」章を参照してください。I/O指定命令INとOUTを使う時はI/Oアドレス\$00～\$3Fが使われなければなりません。LD命令とST命令を使い、データ空間としてI/Oレジスタをアクセスする時はこれらのアドレスに\$20が加算されなければなりません。

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I/Oレジスタ内の全ビットを操作し、設定(1)として読むどのフラグにも1を書き戻し、従ってフラグを解除(0)す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。

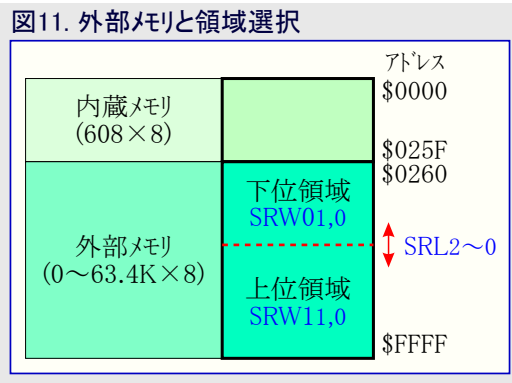
外部メモリ インターフェース

外部メモリ インターフェースが提供する全ての機能で、LCD表示器やA/D、D/Aのような周辺装置や、外部SRAMやフラッシュメモリのようなメモリデバイスへのインターフェースとしての動作に上手く仕立てられます。主な特徴を次に示します。

- ・ 4つの異なる待ち状態設定(含む待ち状態なし)
- ・ 異なる外部メモリ領域(領域容量設定可)に対する個別の待ち状態設定
- ・ アドレス上位バイトに割り当てるビット数が選択可能
- ・ 消費電流を最小とするため、データ線のバス保持機能(任意選択)

概要

外部メモリ(XMEM)が許可されると、内蔵SRAMを越えるアドレス空間は割り当てた外部メモリピンを使うことが利用可能になります(1頁の「[ピン配置](#)」、40頁の表26、43頁の表32、46頁の表38をご覧ください)。メモリ設定は図11.で示されます。



外部メモリ インターフェースの使用

このインターフェースは以下から成ります。

- ・ AD7～0 : 交互切り替えされた下位アドレスバスとデータバス
- ・ A15～8 : 上位アドレスバス(ビット数設定可能)
- ・ ALE : アドレスラッチ許可信号
- ・ $\overline{RD}$: 読み出しストロブ信号
- ・ $\overline{WR}$: 書き込みストロブ信号

外部メモリ(XMEM)インターフェースに関する制御ビットは3つのレジスタ、MCU制御レジスタ(MCUCR)、拡張MCU制御レジスタ(EMCUCR)、特殊I/O機能レジスタ(SFIOR)に配置されます。

XMEMインターフェースが許可されると、このインターフェースに割り当てたポートに対応するデータ方向レジスタ設定を無効にします。このポートの無効化についての詳細に関しては36頁の「[入出力ポート](#)」章の交換機能をご覧ください。XMEMインターフェースはアクセスが内部または外部どちらかを自動的に検知します。アクセスが外部の場合、XMEMインターフェースは図13.(本図は待ち状態なしの波形を示す)に従ったアドレス、データ、制御信号を出力します。ALEがHigh→Low(下降)になるとAD7～0上のアドレスが有効です。データ転送中、ALEはLowです。XMEMインターフェースが許可されると、内部アクセスでも同様にアドレス、データ、ALEポートの動きを引き起こしますが、内部アクセス中、RDとWRストロブは切り替わりません。外部メモリインターフェースが禁止されると、標準ピンとデータ方向設定が使われます。XMEMインターフェースが禁止される時に内蔵SRAM境界以上のアドレス空間が内蔵SRAMに割り当てられないことに注意してください。図12.はGがHighの時に透過(通過:トランスペアレント)となる8ビットのラッチ(一般的には74xx573または同等品)を使うAVRと外部SRAMの接続法を図解します。

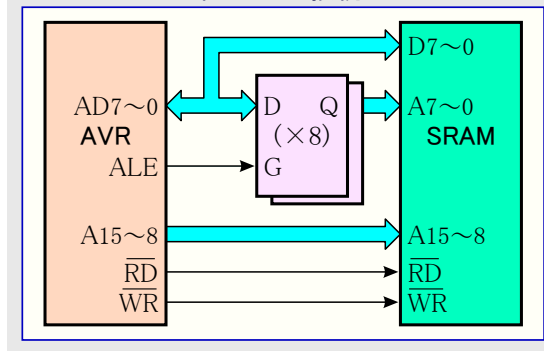
アドレス ラッチの必要性

XMEMインターフェースの高速動作のため、アドレス ラッチは4MHz/2.7V、8MHz/4V以上のシステム周波数に対し、注意して選択されなければなりません。これらの周波数以上の条件で動作するとき、代表的な旧デバイス74HCシリーズのラッチは不適切になります。外部メモリ インターフェースは74AHCシリーズのラッチに適応するよう設計されています。けれども主なタイミング パラメータに対応してさえいれば、多くのラッチが使えます。アドレス ラッチについての主なパラメータは次のとおりです。

- DからQへの伝播遅延時間 (t_{PD}) (訳注) 記号はAVR側ではなく、一般的なラッチ デバイス側での記号です。
- G下降端に対するデータ準備時間 (t_{SU})
- G下降後のデータ(アドレス)保持遅延時間 (t_H)

外部メモリ インターフェースはGがLowにされた後、 $t_H=5ns$ の最小アドレス保持時間を保証するよう設計されています(131～133頁、表98.～105.の t_{LAXX_LD} と t_{LAXX_ST} を参照してください)。DからQへの伝播遅延時間(t_{PD})は外部部品のアクセス時間必要条件を計算する時に考慮されなければなりません。G=Low(下降)前のデータ準備時間(t_{SU})はALE=Low(下降)前のアドレス有効時間(t_{AVLLC})－PCB配線遅延(容量性負荷に依存)を越えてはいけません。

図12. AVRと外部SRAMの接続



プルアップとバス保持機能

AD7～0ポートのプルアップ抵抗は対応するポートレジスタが1を書かれると、活性(有効)にできます。休止形態で消費電力を削減するために、休止形態移行前にポートレジスタへ0を書くことによってプルアップを禁止することが推奨されます。

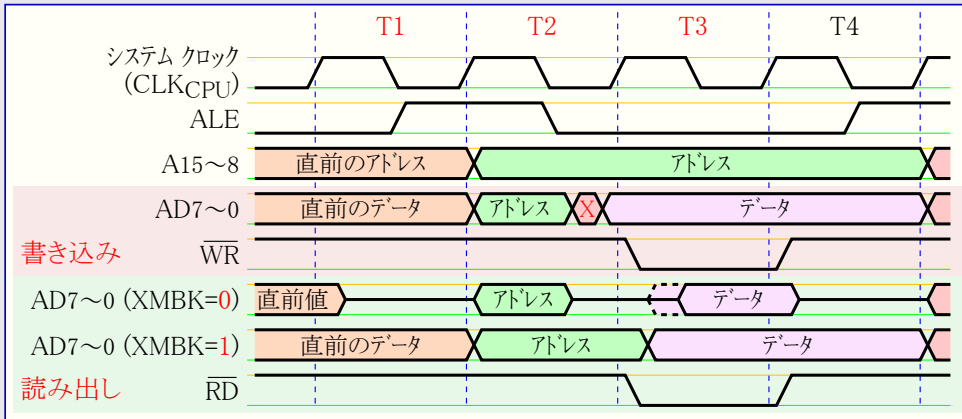
XMEMインターフェースはAD7～0線のバス保持機能も提供します。バス保持機能は18頁の「特殊I/O機能レジスタ(SFIOR)」で記述されるようにプログラムで許可と禁止ができます。許可されると、(本来)これらの線がXMEMインターフェースによりHi-Zとされる間中、バス保持機能はAD7～0バスで直前の値を保持します。

タイミング

外部メモリ デバイスは様々なタイミング必要条件を持ちます。これらの必要条件に合わせるため、ATmega8515のXMEMインターフェースは表3.に示される4つの異なる待ち状態を提供します。この待ち状態選択前に外部メモリ デバイスのタイミング特性を考慮することが大事です。重要なパラメータの多くはATmega8515の必要条件設定に関連する外部メモリのアクセス時間です。外部メモリのアクセス時間はチップ選択とアドレスを受け取ってから、そのアドレスのデータがバス上で実際に駆動されるまでの時間で定義されます。このアクセス時間はALEパルスがLowにされてから読み出し手順中にデータが安定にならなければならないまでの時間(131～132頁の表98.～105.の $t_{LLRL} + t_{RLRH} - t_{DVRH}$)を超えることはできません。各種待ち状態はソフトウェアで設定されます。付加機能として、外部メモリ空間を個別に待ち状態設定できる2つの領域に分割することが可能です。これは異なるタイミング必要条件の2つの異なるメモリ デバイスを同じXMEMインターフェースに接続することを可能にします。XMEMインターフェースの詳細タイミングについては131頁からの表98.～105.と図89.～92.を参照してください。

XMEMインターフェースは非同期で、次の図の波形が内部システム クロックに関連することに注意してください。内部と外部(XTAL1)のクロック間のズレは保証されません(デバイス、温度、供給電圧間で一様でない)。従ってXMEMインターフェースは同期動作用ではありません。

図13. 待ち状態なし 外部データメモリアクセス周期 (SRWn1=0, SRWn0=0) [T1~T3]



共通注意事項

SRWn1はSRW11(上位領域)またはSRW01(下位領域)、SRWn0はSRW10(上位領域)またはSRW00(下位領域)です。

[T1~Tn]は図での命令実行周期範囲です。

Xは値変更区間を示します。

最終周期のALEは次命令がRAM(内部または外部)をアクセスする場合だけ存在します。また、最後の周期のアドレスとデータの変更区間の有無も同様です。

図14. 1待ち状態 外部データメモリアクセス周期 (SRWn1=0, SRWn0=1) [T1~T4]

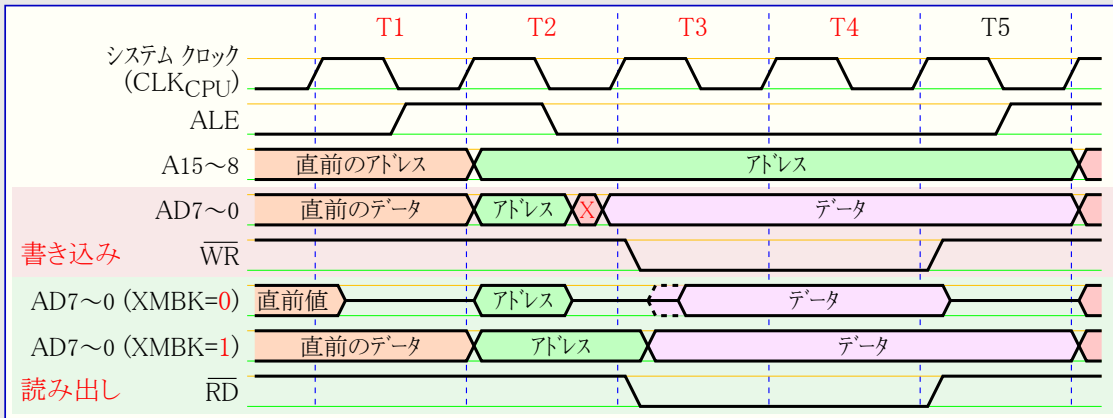


図15. 2待ち状態 外部データメモリアクセス周期 (SRWn1=1, SRWn0=0) [T1~T5]

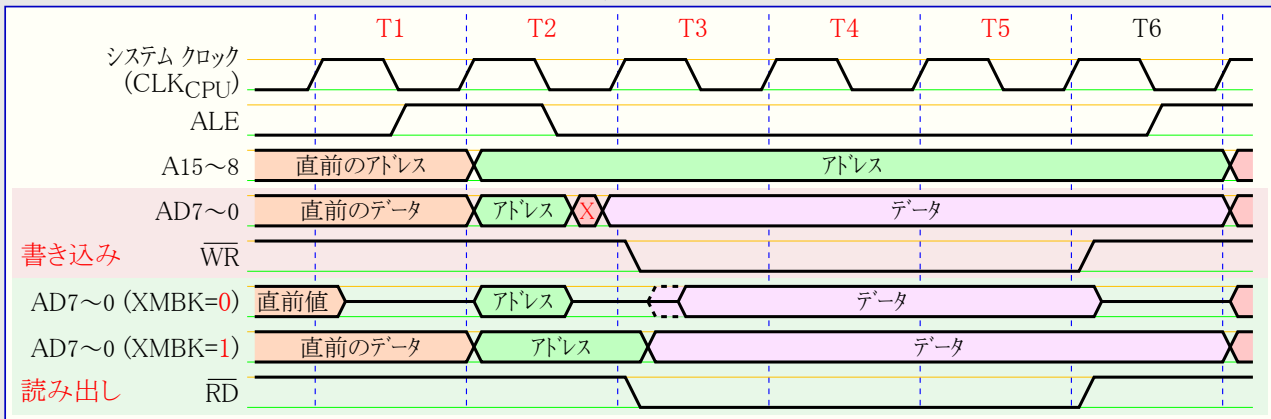
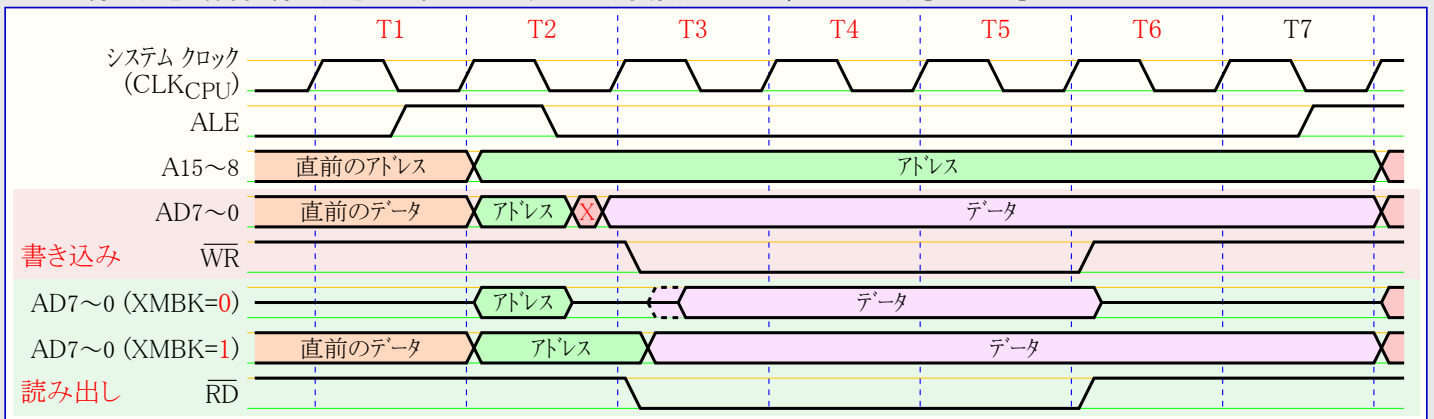


図16. 2待ち状態+保持1待ち状態 外部データメモリアクセス周期 (SRWn1=1, SRWn0=1) [T1~T6]



XMEMインターフェース用レジスタ

■ MCU制御レジスタ (MCU Control Register) MCUCR

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	MCUCR
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 - SRE : 外部メモリ許可 (External SRAM/XMEM Enable)

SREに1を書くことが外部メモリ インターフェースを許可します。A15~8, AD7~0, ALE, $\overline{RD}$, $\overline{WR}$ ピン機能は交換ピン機能として活性(有効)にされます。SREビットはそれぞれのデータ方向レジスタ内の何れのピン方向設定も無効にします。SREに0を書くことは外部メモリ インターフェースを禁止し、通常ピンとデータ方向設定が使われます。

■ ビット6 - SRW10 : 待ち状態選択ビット (Wait-state Select Bit)

詳細な記述については以降(EMCUCR記述)のSRWnビットの共通記述をご覧ください。

■ 拡張MCU制御レジスタ (Extended MCU Control Register) EMCUCR

ビット	7	6	5	4	3	2	1	0	
\$36 (\$56)	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	EMCUCR
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6~4 - SRL2~0 : 外部SRAM範囲選択 (Wait-state Sector Limit)

異なる外部メモリ領域に対して異なる待ち状態を指定することが可能です。外部メモリ アドレス空間は個別の待ち状態ビットを持つ2つの領域に分割できます。SRL2~0ビットは、これらの領域の分岐点を選びます。表2.と図11.をご覧ください。既定ではSRL2~0が0に設定され、外部メモリ アドレス空間全体が1つの領域として扱われます。SRAM(外部メモリ)アドレス空間全体が1つの領域として設定される場合、待ち状態はSRW11とSRW10ビットによって設定されます。

表2. 上位/下位領域選択

SRL2	SRL1	SRL0	下位領域	上位領域
0	0	0	なし	\$0260~\$FFFF
0	0	1	\$0260~\$1FFF	\$2000~\$FFFF
0	1	0	\$0260~\$3FFF	\$4000~\$FFFF
0	1	1	\$0260~\$5FFF	\$6000~\$FFFF
1	0	0	\$0260~\$7FFF	\$8000~\$FFFF
1	0	1	\$0260~\$9FFF	\$A000~\$FFFF
1	1	0	\$0260~\$BFFF	\$C000~\$FFFF
1	1	1	\$0260~\$DFFF	\$E000~\$FFFF

■ ビット1,MCUCRのビット6 - SRW11,0 : 上位領域待ち状態選択ビット1,0 (Wait-state Select Bits for Upper Sector)

SRW11とSRW10ビットは外部メモリ アドレス空間の上位領域に対する待ち状態数を制御します。表3.をご覧ください。

■ ビット3,2 - SRW01,0 : 下位領域待ち状態選択ビット1,0 (Wait-state Select Bits for Lower Sector)

SRW01とSRW00ビットは外部メモリ アドレス空間の下位領域に対する待ち状態数を制御します。表3.をご覧ください。

表3. 待ち状態選択

SRWn1	SRWn0	待ち状態
0	0	待ち状態なし
0	1	読み/書きスロープ中に1待ち状態挿入
1	0	読み/書きスロープ中に2待ち状態挿入
1	1	読み/書きスロープ中に2待ち状態、アドレス、データ出力保持中に1待ち状態挿入

注: nは0(下位領域)または1(上位領域)です。

外部メモリ インターフェースの待ち状態とタイミングのより多くの詳細については、図13.~16.のSRWビット設定がタイミングへ与える影響をご覧ください。

■ 特殊I/O機能レジスタ (Special Function I/O Register) SFIOR

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	–	XMBK	XMM2	XMM1	XMM0	PUD	–	PSR10	SFIOR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット6 – XMBK : バス保持許可 (External Memory Bus-keeper Enable)

XMBKに**1**を書くことがAD7～0線のバス保持機能を許可します。バス保持機能が許可されると、XMEMインターフェースがAD7～0線をHi-Zにしても、AD7～0はその線上で最後に駆動した値を保ちます。XMBKに**0**を書くことがバス保持機能を禁止します。XMBKは**SRE**の制限を受けませんので、XMEMインターフェースが禁止されても、バス保持機能はXMBKが**1**である限り未だ活性(有効)にされます。

■ ビット5～3 – XMM2～0 : 上位アドレス遮蔽 (External Memory High Mask)

外部メモリインターフェースが許可されると、既定では全てのポートCピンが上位アドレスバイトに使われます。外部メモリアクセスのために全64K (64928)バイトアドレス空間が必要とされない場合、**表4.**で記述されるように、いくつかまたは全てのポートCピンは標準ポートピン機能用に開放できます。**19頁の「外部メモリの全64Kバイト位置の使用」**で記述されるように、外部メモリの全64Kバイト位置のアクセスのためにXMMnビットの使用が可能です。

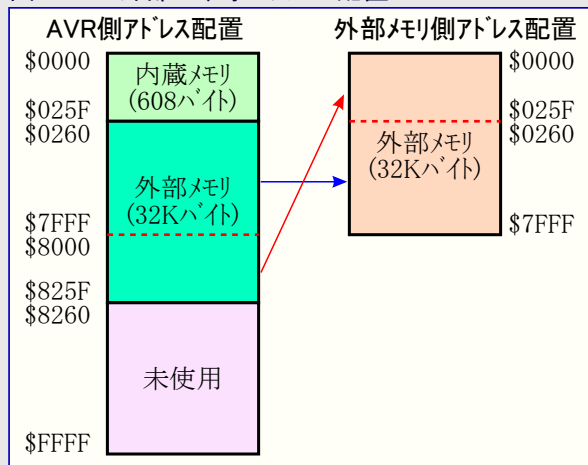
表4. 外部メモリ許可時に開放するポートCピン

XMM2	XMM1	XMM0	アドレス上位バイトビット数	開放ポートピン
0	0	0	8 (最大約63.4Kバイト)	なし
0	0	1	7	PORTC7
0	1	0	6	PORTC7～6
0	1	1	5	PORTC7～5
1	0	0	4	PORTC7～4
1	0	1	3	PORTC7～3
1	1	0	2	PORTC7～2
1	1	1	0 (上位バイトなし)	PORTC7～0

64Kバイト未満外部メモリの全域使用

外部メモリは**図11.**で示されるように内部メモリの後に配置されるため、データ空間の最初の608バイトをアドレス指定するとき、外部メモリは指定されません。外部メモリの最初の608バイト(外部アドレス\$0000～\$025F)がアクセスできないと思われるかもしれません。けれども64Kバイトより小さな外部メモリを接続するとき、例えば32Kバイトで、これらの位置は単純なアドレス\$8000～\$825F指定によって容易にアクセスされます。外部メモリアドレスA15ビットは外部メモリに接続されず、アドレス\$8000～\$825Fは外部メモリに対するアドレス\$0000～\$025Fとして見えます。\$825Fを越えるアドレス指定は、これが他の(下位側)アドレスによって既にアクセスされる外部メモリ位置のアドレス指定のため推奨されません。応用ソフトウェアにとって、この外部32Kバイトメモリは、\$0260～\$825Fの1つの直線的な32Kバイトアドレス空間として見えます。これは**図17.**で図解されます。

図17. 32K外部メモリ時のアドレス配置



外部メモリの全64Kバイト位置の使用

外部メモリは図11.で示されるように内部メモリの後に配置されるため、既定では外部メモリの約63.4K(64928)バイトだけが利用可能です(アドレス空間\$0000～\$025Fは内部メモリ用に予約されます)。けれども上位アドレスビットを0で遮蔽することにより、外部メモリ全体を利用することが可能です。これはXMMnビットを使い、アドレスの最上位側ビットをソフトウェアによって制御することで行えます。ポートCを\$00出力に設定し、上位側ビットを標準ポートピン動作作用に開放することにより、メモリインターフェースは\$0000～\$1FFFをアドレス指定します。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

```
. EQU    OFFSET=$2000                                ;切り替え点アドレス定義
;                                                    ;[ポートC初期化]
LDI      R16, $00                                     ;ポート出力値を取得
OUT      PORTC, R16                                  ;アドレス最上位側ビット値全0設定
LDI      R16, $FF                                     ;ポート方向全出力値を取得
OUT      DDRC, R16                                   ;アドレス上位バイト用ポート出力設定
;                                                    ;[$0000～$1FFFアクセス]
LDI      R16, (1<<XMM1) | (1<<XMM0)                  ;PORTC7～5開放値を取得
STS      SFIOR, R16                                  ;PORTC7～5開放(～$1FFF範囲設定)
LDI      R16, $AA                                     ;書き込み値を取得
STS      $0001+OFFSET, R16                           ;外部メモリの$0001番地に$AA書き込み
;                                                    ;[$2000～$FFFFアクセス]
LDI      R16, $00                                     ;最大外部メモリ(16ビット幅)値を取得
STS      SFIOR, R16                                  ;PORTC開放なし、(～$FFFF範囲設定)
LDI      R16, $55                                     ;書き込み値を取得
STS      $0001+OFFSET, R16                           ;外部メモリのOFFSET+$0001番地に$55書き込み
```

C言語プログラム例

```
#define OFFSET 0x2000                                /* 切り替え点アドレス定義 */
void XRAM_example(void)
{
    unsigned char *p = (unsigned char *) (OFFSET + 1); /* 書き込みアドレス(ポインタ)定義 */
    PORTC = 0x00;                                     /* アドレス最上位側ビット値全0設定 */
    DDRC = 0xFF;                                       /* アドレス上位バイト用ポート出力設定 */
    SFIOR = (1<<XMM1) | (1<<XMM0);                    /* PORTC7～5開放(～$1FFF範囲設定) */
    *p = 0xAA;                                         /* 外部メモリの$0001番地に$AA書き込み */
    SFIOR = 0x00;                                     /* PORTC開放なし、(～$FFFF範囲設定) */
    *p = 0x55;                                         /* 外部メモリの$2001番地に$55書き込み */
}
```

注: 4頁の「コード例について」をご覧ください。

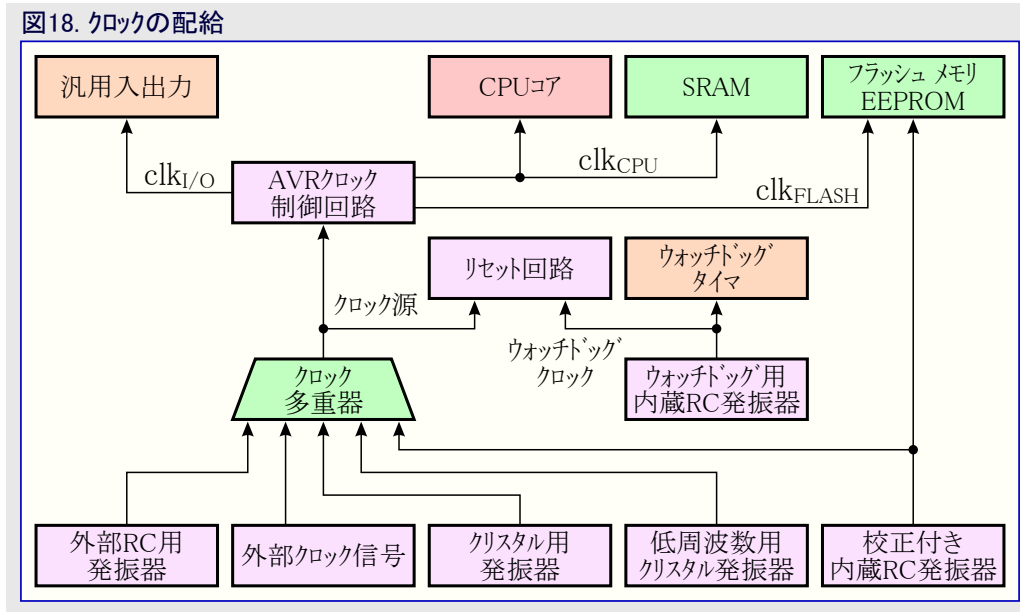
メモリの多くが遮蔽され隠されているとき、この選択手段の使用は注意して用いられなければなりません。

(訳補) 上の記述はXMMn設定によりバンク化動作となることを示しています。

システム クロックとクロック選択

クロック系統とその配給

図18.はAVR内の主要なクロック系統とその配給を示します。全てのクロックが与えられた時間有効である必要はありません。消費電力低減のため、24頁の「電力管理と休止形態」で記述される各種休止形態の使用によって、使われない部分のクロックを停止することができます。クロック系統は以下で詳細に示されます。



CPU クロック – clk_{CPU}

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

I/O クロック – clk_{I/O}

I/Oクロックはタイマ/カウンタ、SPI、USARTのようなI/O部の大部分で使われます。I/Oクロックは外部割り込み部でも使われますが、いくつかの外部割り込みは例えば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。

フラッシュ クロック – clk_{FLASH}

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

クロック元

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

各クロック選択に対する様々な選択は次項で得られます。CPUがパワーダウン動作から起動する時、選択したクロック元は命令実行開始前に安定な発振器動作を保証する起動時間に使われます。CPUがリセットから始まる時、これらは通常動作開始前に安定電圧へ達するのを電源に許す付加遅延です。ウォッチドッグ発振器は、この起動時間の実時間部のタイミングに使われます。各計時完了に使われるウォッチドッグ(WDT)発振器の周期数は表6.で示されます。134頁の「代表特性」で示されるようにウォッチドッグ発振器の周波数は電圧に依存します。

表5. クロック種別選択

クロック種別	CKSEL3~0
外部クリスタル/セラミック発振器	1111~1010
外部低周波数クリスタル発振器	1001
外部RC発振	1000~0101
校正付き内蔵RC発振器	0100~0001
外部クロック信号	0000

注: 1=非プログラム、0=プログラム

表6. WDT発振器の代表的計時完了値、周期数

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

既定のクロック元

このデバイスはCKSEL=0001, SUT=10で出荷されます。従って既定クロック元設定は最長起動時間の1MHz内蔵RC発振器です。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って、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。

クリスタル用発振器

XTAL1とXTAL2は図19.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。CKOPTヒューズは2つの異なる発振増幅器動作の1つを選択します。CKOPTヒューズがプログラム(0)されると、発振器出力は完全な供給電圧端振幅出力で発振します。この動作は雑音が多い環境やXTAL2出力が他のクロック緩衝器を駆動する時に適します。この動作は広い周波数範囲を持ちます。CKOPTヒューズが非プログラム(1)にされると、発振器はより小さな出力振幅になります。これはかなり消費電力を削減します。この動作は制限された周波数範囲を持ち、他のクロック緩衝器を駆動するのに使えません。

セラミック振動子での最高周波数はCKOPTヒューズが非プログラム(1)で8MHz、CKOPTヒューズがプログラム(0)で16MHzです。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7.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。

この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7.で示されるようにCKSEL3~1ヒューズによって選択されます。

CKSEL0ヒューズはSUT1,0ヒューズと共に表8.で示されるように起動時間を選択します。

図19. クリスタル発振子接続図

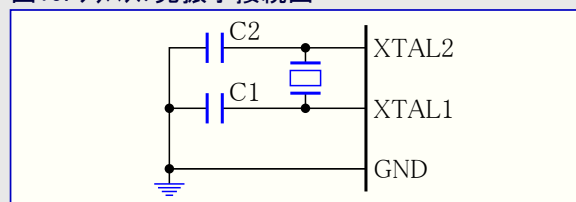


表7. クリスタル発振器動作

CKOPT	CKSEL3~1	周波数範囲	推奨C1,2容量
1	101 (注1)	0.4~0.9MHz	-
	110	0.9~3.0MHz	12~22pF
	111	3.0~8.0MHz	12~22pF
0	101~111	1.0~ MHz	12~22pF

注1: この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8. クリスタル発振子/セラミック振動子用起動遅延時間選択表

CKSEL0	SUT1,0	パワーダウン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)	4.1ms	外部セラミック振動子、高速上昇電源
	0 1	258 × CK (注1)	65ms	外部セラミック振動子、低速上昇電源
	1 0	1K × CK (注2)	-	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)	4.1ms	外部セラミック振動子、高速上昇電源
1	0 0	1K × CK (注2)	65ms	外部セラミック振動子、低速上昇電源
	0 1	16K × 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	4.1ms	外部クリスタル発振子、高速上昇電源
	1 1	16K × CK	65ms	外部クリスタル発振子、低速上昇電源

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

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

低周波数クリスタル用発振器

デバイスに対するクロック元として時計用32.768kHzクリスタルを使うには、CKSEL3~0ヒューズを'1001'に設定することによって低周波数クリスタル発振器が選択されなければなりません。クリスタルは図19.で示されるように接続されるべきです。CKOPTヒューズのプログラム(0)により、使用者はXTAL1とXTAL2の内部容量(コンデンサ)を許可でき、それによって外部コンデンサの必要がなくなります。内部容量は36pFの公称値です。

この発振器が選択されると、起動時間は表9.で示されるようにSUTヒューズによって決定されます。

表9. 低周波数クリスタル発振器用起動遅延時間選択表

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)	推奨使用法
0 0	1K × CK (注1)	4.1ms	高速上昇電源または低電圧検出(BOD)リセット許可
0 1	1K × CK (注1)	65ms	低速上昇電源
1 0	32K × CK	65ms	起動時周波数の安定重視
1 1			(予約)

注1: これらの選択は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。

外部RC発振器

タイミングに鈍感な応用に対しては図20.で示される外部RC設定が使えます。周波数は式 $f=1/(3RC)$ によって大まかに推測されます。Cは最低22pFであるべきです。CKOPTヒューズのプログラム(0)により、使用者はXTAL1とGND間の36pF内部容量を許可でき、それによって外部コンデンサの必要がなくなります。

この発振器は示された周波数範囲で各々最適化された4つの異なる種別で動作できます。この動作は表10.で示されるようにCKSEL3~0ヒューズによって選択されます。

この発振器が選択されると、起動時間は表11.で示されるようにSUTヒューズによって決定されます。

図20. 外部RC接続図

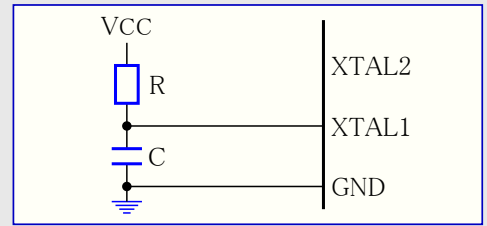


表10. 外部RC発振器動作

CKSEL3~0	周波数範囲 (MHz)
0 1 0 1	0.1~0.9
0 1 1 0	0.9~3.0
0 1 1 1	3.0~8.0
1 0 0 0	8.0~12.0

表11. 外部RC発振器用起動遅延時間選択表

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)	推奨使用法
0 0	18×CK	–	低電圧検出(BOD)リセット許可
0 1	18×CK	4.1ms	高速上昇電源
1 0	18×CK	65ms	低速上昇電源
1 1	6×CK (注1)	4.1ms	高速上昇電源または低電圧検出(BOD)リセット許可

注1: この選択はデバイスの最高周波数付近で動作する時に使われるべきではありません。

校正付き内蔵RC発振器

校正された内蔵RC発振器は決められた1.0, 2.0, 4.0, 8.0MHzクロックを供給します。全ての周波数は5V, 25°Cでの公称値です。このクロックは表12.で示されるようにCKSELヒューズのプログラミングによってシステムクロックとして選択できます。選択したなら、外部部品なしで動作します。このクロック選択を使うとき、CKOPTヒューズは常に非プログラム(1)にされるべきです。リセット中、ハードウェアが発振校正レジスタ(OSCCAL)に校正値バイトを設定し、これによってRC発振器を自動的に校正します。5V, 25°Cで1.0MHz発振器周波数が選択され、この校正は公称周波数±3%以内の周波数を与えます。atmel.com/avrで利用可能な応用記述に記載された実行時校正法の使用で、与えられたもののVCCと温度でも±1%の精度を達成することができます。この発振器がチップ(システム)クロックとして使われるとき、ウォッチドッグ発振器は未だウォッチドッグタイマとリセット付加遅延タイマに使われます。予め設定された校正値のより多くの情報については117頁の「校正バイト」項をご覧ください。

この発振器が選択されると、起動時間は表13.で示されるようにSUTヒューズによって決定されます。XTAL1とXTAL2は未接続(NC)のままにされるべきです。

表12. 校正付き内蔵RC発振器動作

CKSEL3~0	公称周波数 (MHz)
0 0 0 1 (注1)	1.0
0 0 1 0	2.0
0 0 1 1	4.0
0 1 0 0	8.0

注1: デバイスはこの選択で出荷されます。

表13. 校正付き内蔵RC発振器用起動遅延時間選択表

SUT1,0	パワーダウンからの 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	–	低電圧検出リセット(BOD)許可
0 1	6×CK	4.1ms	高速上昇電源
1 0 (注1)	6×CK	65ms	低速上昇電源
1 1			(予約)

注1: デバイスはこの選択で出荷されます。

■ 発振校正レジスタ (Oscillator Calibration Register) OSCCAL

ビット	7	6	5	4	3	2	1	0	
\$04 (\$24)	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	デバイス固有の校正値								

■ ビット7~0 - CAL7~0 : 発振校正値 (Oscillator Calibration Value)

このアドレスへの校正バイト書き込みは発振器周波数の偏差処理を省くために内蔵発振器を調整します。リセット中、識別列上位バイト(アドレス\$00)に配置される1MHz校正値が発振校正レジスタ(OSCCAL)内へ自動的に設定されます。内蔵RC発振器が他の周波数で使われる場合、校正値は手動で設定されなければなりません。これは初めに書き込み器によって識別列を読み、そしてその後フラッシュメモリまたはEEPROMに校正値を保存することによって行えます。その後この値はソフトウェアによって読まれ、OSCCALに設定できます。OSCCALが0の時に最低利用可能周波数が選択されます。このレジスタへ0以外の値を書くことは内蔵発振器の周波数を増加します。このレジスタへの\$FF書き込みは最高使用可能周波数にします。校正付き発振器はフラッシュメモリとEEPROMのアクセス時間に使われます。フラッシュメモリまたはEEPROMが書かれる場合、公称周波数より上へ10%を越えて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。この発振器は1.0, 2.0, 4.0, 8.0MHzへの校正が意図されることに注意してください。表14.で示されるような他の値への調整は保証されません。

表14. 内蔵RC発振器周波数範囲

OSCCAL値	公称周波数に対する割合	
	最小	最大
\$00	50 %	100 %
\$7F	75 %	150 %
\$FF	100 %	200 %

外部クロック信号

外部クロック元からデバイスを駆動するためにXTAL1は図21.で示されるように駆動されるべきです。外部クロックでデバイスを走行するためにCKSELピンスは'0000'にプログラム(設定)されなければなりません。CKOPTピンスのプログラム(0)により、使用者はXTAL1とGND間の36pF内部容量を許可できます。

このクロック元が選択されると、起動時間は表15.で示されるようにSUTピンスによって決定されます。

図21. 外部クロック信号駆動接続図

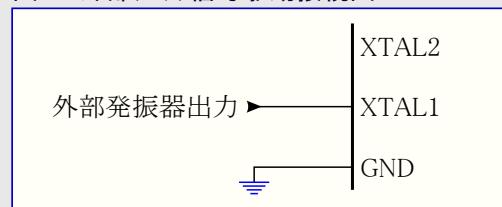


表15. 外部クロック信号駆動用起動遅延時間選択表

SUT1,0	パワーダウンからの 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	–	低電圧検出リセット(BOD)許可
0 1	6 × CK	4.1ms	高速上昇電源
1 0	6 × CK	65ms	低速上昇電源
1 1	(予約)		

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

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。

3つの休止形態の何れかへ移行するにはMCU制御レジスタ(MCUCR)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。MCU制御/状態レジスタ(MCUCSR)のSM2、MCU制御レジスタ(MCUCR)のSM1、拡張MCU制御レジスタ(EMCUCR)のSM0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、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16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。

20頁の図18はATmega8515の各種クロック系統とその配給を示します。この図は適切な休止形態を選択する助けになります。

■ MCU制御レジスタ (MCU Control Register) MCUCR

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	MCUCR
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5 – 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。

■ ビット4 – SM1 : 休止種別選択ビット1 (Sleep Mode Select Bit 1)

この種別選択ビットは表16で示される利用可能な3つの休止形態種別の1つを選びます。

■ MCU制御/状態レジスタ (MCU Control and Status Register) MCUCSR

ビット	7	6	5	4	3	2	1	0	
\$34 (\$54)	–	–	SM2	–	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R	R	R/W	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5 – SM2 : 休止種別選択ビット2 (Sleep Mode Select Bit 2)

この種別選択ビットは表16で示される利用可能な3つの休止形態種別の1つを選びます。

■ 拡張MCU制御レジスタ (Extended MCU Control Register) EMCUCR

ビット	7	6	5	4	3	2	1	0	
\$36 (\$56)	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	EMCUCR
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 – SM0 : 休止種別選択ビット0 (Sleep Mode Select Bit 0)

この種別選択ビットは表16で示される利用可能な3つの休止形態種別の1つを選びます。

表16. 休止形態種別選択

SM2	SM1	SM0	休止形態種別
0	0	0	アイドル動作
0	0	1	(予約)
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。

アイドル動作

休止種別選択(SM2~0)ビットが'000'を書かれるとき、**SLEEP**命令はMCUをアイドル動作へ移行させ、CPUを停止しますが、SPI、USART、アナログ比較器、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clk_{CPU}とclk_{FLASH}を停止する一方、他のクロックに走行を許します。

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

パワーダウン動作

休止種別選択(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、外部割り込みとウォッチドッグ機能は(許可されていれば)継続して動作します。外部リセット、ウォッチドッグリセット、低電圧検出(BOD)リセット、INT0またはINT1の外部レベル割り込み、INT2の外部割り込みだけが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49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20頁の「**クロック元**」で記述されるように、リセット遅延時間を定義するのと同じ**CKSEL**ヒューズによって定義されます。

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

表17. 各休止形態に於ける動作クロック範囲と復帰起動要因

休止種別	動作クロック範囲			動作発振器	復帰起動要因 (割り込み)		
	clk _{CPU}	clk _{FLASH}	clk _{I/O}	主クロック供給元	INT2,INT1,INT0	SPM/EEPROM操作可	その他I/O
アイドル			○	○	○	○	○
パワーダウン					○ (注2)		
スタンバイ(注1)				○	○ (注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択された場合です。

注2: INT2または、INT1とINT0のレベル割り込みだけです。

消費電力の最小化

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

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。その他の休止形態でのアナログ比較器は自動的に禁止されます。しかしアナログ比較器が入力として内部基準電圧を使う設定の場合、全休止形態でアナログ比較器は禁止されるべきです。さもなければ、内部基準電圧は休止形態と無関係に許可されます。アナログ比較器設定法の詳細については106頁の「[アナログ比較器](#)」を参照してください。

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BODエンブースによって許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。低電圧検出器(BOD)設定法の詳細については28頁の「[低電圧検出\(BOD\)](#)」を参照してください。

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で記述されたように禁止されると、内部基準電圧は禁止され、電力を消費しません。再び許可する場合、この出力が使われる前に使用者は基準電圧へ起動(安定時間)を与えなければなりません。基準電圧が休止形態でON保持される場合、この出力は直ちに使えます。起動時間の詳細については29頁の「[内部基準電圧](#)」を参照してください。

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30頁の「[ウォッチドッグ タイマ](#)」を参照してください。

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})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38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。

システム制御とリセット

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、きつとリセット処理ルーチンへの**RJMP**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、割り込みベクタがブート領域の場合やその逆も同様です。**図22**の回路構成図はリセット論理回路を示します。**表18**はリセット回路の電気的特性を定義します。

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

全てのリセット元が無効にされてしまった後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電源に許します。遅延タイマの遅延時間は**CKSELヒューズ**を通して使用者によって定義されます。この遅延時間についての各種選択は**20頁の「クロック元」**で示されます。

リセット元

ATmega8515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_{POT})**以下でMCUがリセットされます。
- ・ **外部リセット** RESETピンが**最小パルス幅**以上**Low**レベルに保たれると、MCUがリセットされます。
- ・ **ウォッチドッグ リセット** ウォッチドッグが許可され、ウォッチドッグ タイマが終了すると、MCUがリセットされます。
- ・ **低電圧リセット** 低電圧検出器(BOD)が許可され、供給電圧(VCC)が**低電圧検出電圧(V_{BOT})**以下でMCUがリセットされます。

図22. リセット回路構成

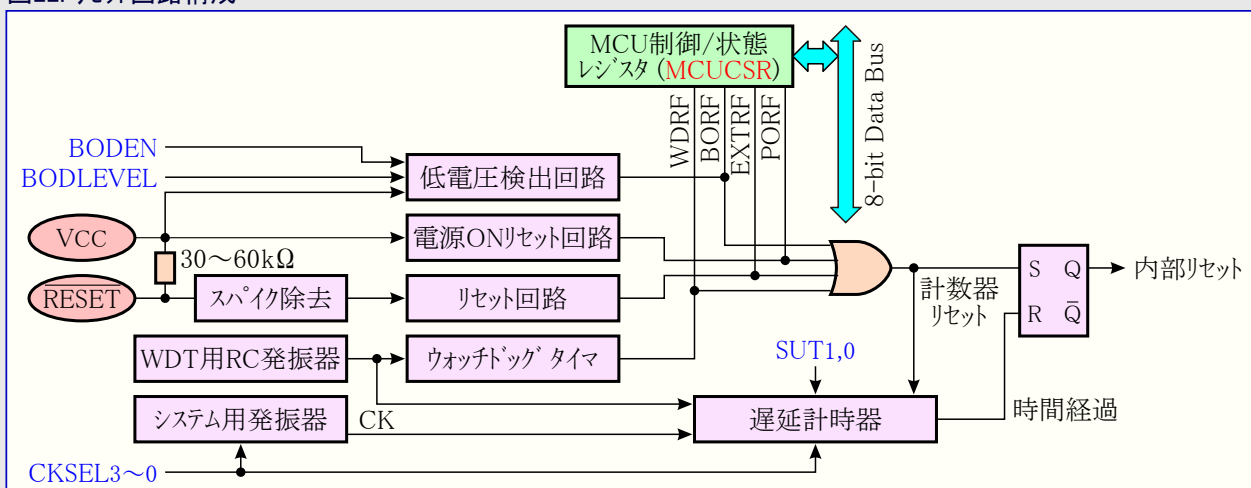


表18. リセット電気的特性

シンボル	項目		最小	代表	最大	単位
V _{POT}	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 _{RST}	RESETピン閾値電圧		0.1VCC		0.9VCC	
t _{RST}	リセット パルス幅		1.5			μs
V _{BOT}	低電圧検出 閾値電圧 (注2)	BODLEVEL=非プログラム(1)	2.5	2.7	3.2	V
		BODLEVEL=プログラム(0)	3.7	4.0	4.5	
t _{BOT}	最小低電圧検出時間	BODLEVEL=非プログラム(1)		2		μs
		BODLEVEL=プログラム(0)		2		
V _{HYST}	低電圧検出ヒステシス電圧			130		mV

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2: V_{BOT}はいくつかのデバイスについて公称最小動作電圧以下かもしれません。この状態のデバイスについて、そのデバイスは製造検査中にVCC=V_{BOT}へ落として検査されます。これはVCCが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へ落ちる前に低電圧(BOD)リセットが起きることを保証します。この検査はATmega8515LについてはBODLEVEL=1、ATmega8515についてはBODLEVEL=0を使って実行されます。ATmega8515に対してBODLEVEL=1は適用できません。

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18.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23. 内蔵電源ONリセット (RESETはVCCに接続)

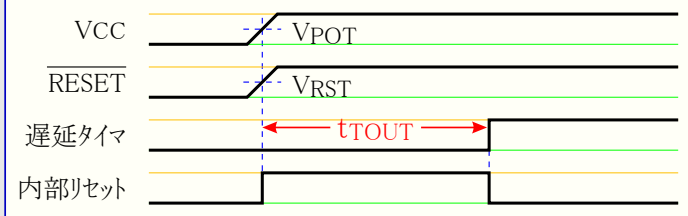
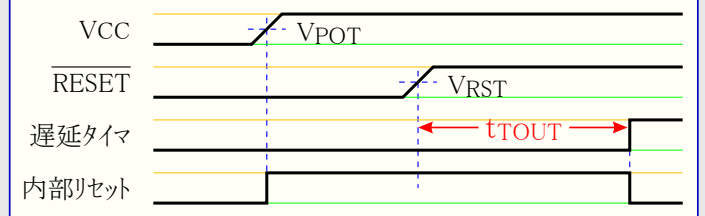


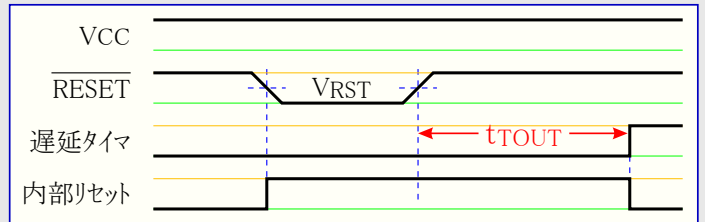
図24. 外部RESET信号による延長電源ONリセット



外部リセット

外部リセットはRESETピンのLowレベルによって生成されます。例えばクロックが動いていなくても、最小パルス幅(表18.参照)以上のリセットパルスはリセットを生成します。短すぎるパルスはリセットの生成が保証されません。印加された信号の上昇がリセット閾値電圧(V_{RST})に達すると(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。

図25. 動作中の外部リセット



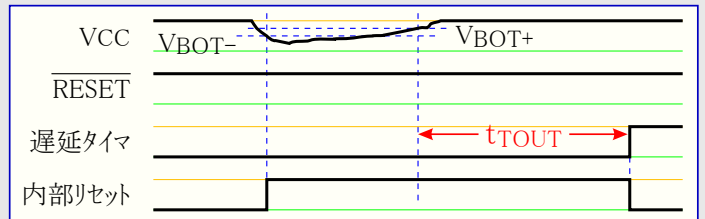
低電圧(ブラウンアウト)検出

ATmega8515には固定化された起動(検出)電圧と比較することで動作中のVCCを監視するチップ上の低電圧検出(BOD)回路があります。BODの起動電圧はBODLEVELヒューズにより、2.7V(非プログラム(1))または4.0V(プログラム(0))を選択できます。この起動電圧はスパイク対策BODを保証するためにヒステリシスを持ちます。検出電圧のヒステリシスは $V_{BOT+} = V_{BOT} + V_{HYST}/2$ 、 $V_{BOT-} = V_{BOT} - V_{HYST}/2$ と解釈すべきです。

BOD回路はBODENヒューズによって許可/禁止ができます。BODが許可(BODEN=プログラム(0))され、VCCが起動電圧以下の値に下降すると(図26.の V_{BOT-})、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(図26.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、MCUを始動します。

BOD回路は電圧が表18.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。

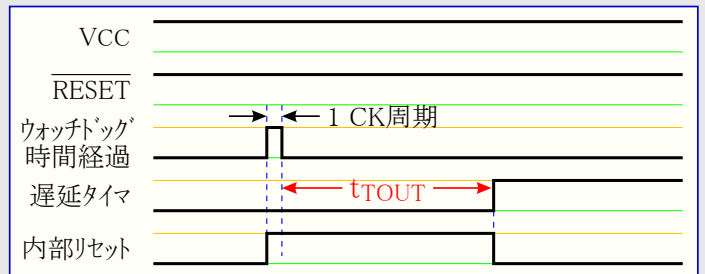
図26. 動作中の低電圧検出リセット



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30頁を参照してください。

図27. 動作中のウォッチドッグ リセット



■ MCU制御/状態レジスタ (MCU Control and Status Register) MCUCSR

MCU制御/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)	–	–	SM2	–	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R	R	R/W	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MCUCSR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、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。

内部基準電圧

ATmega8515は内部基準電圧が特徴です。この基準電圧は低電圧検出(BOD)に使われ、アナログ比較器の入力としても使えます。

基準電圧許可信号と起動時間

この基準電圧には使われるべき方法に影響を及ぼすかもしれない起動時間があります。この起動時間は表19.で与えられます。節電のために、この基準電圧は常にONではありません。この基準電圧は次の状態中ONです。

1. 低電圧検出リセット許可時 (BODENヒューズ=プログラム(0))
2. アナログ比較器基準電圧接続時 (アナログ比較器 制御/状態レジスタ(ACSR)の基準電圧入力選択(ACBG)=1)

従って低電圧検出(BOD)が許可されていないと、ACBGの設定(=1)後、使用者はアナログ比較器出力が使われる前に、基準電圧へ起動時間を与えなければなりません。パワーダウン動作での消費電力を減らすため、使用者はパワーダウン動作へ移行する前に基準電圧がOFFされるのを保証することで上の2つの状態を避けられます。

表19. 内部基準電圧特性

シンボル	項目	最小	代表	最大	単位
V _{BG}	基準電圧	1.15	1.23	1.35	V
t _{BG}	起動時間		40	70	μs
I _{BG}	消費電流		10		μA

ウォッチドッグ タイマ

ウォッチドッグ タイマは1MHzで動く独立したチップ上の発振器からクロック駆動されます。これはVCC=5Vでの代表値です。他のVCC電圧での代表値については「[特性データ](#)」をご覧ください。ウォッチドッグ タイマ前置分周器の制御により、ウォッチドッグ リセット間隔は表21.で示されるように調整できます。ウォッチドッグ リセット(WDR)命令はウォッチドッグ タイマをリセットします。ウォッチドッグ タイマはそれが禁止される時とチップ リセットが起こる時もリセットされます。8つの異なるクロック周期がこのリセット周期を決めるために選択できます。別のウォッチドッグ リセットなしにリセット周期が経過すると、ATmega8515はリセットしてリセットベクタから実行します。ウォッチドッグ リセットの詳細タイミングについては28頁を参照してください。

予期せぬウォッチドッグ 禁止や予期せぬ計時終了周期変更を防ぐため、3つの異なる安全レベルが表20.で示されるWDTONとS8515Cヒューズによって選択されます。安全レベル0はAT90S4414/8515での設定に相当します。どの安全レベルでもWDTの許可に制限はありません。詳細については31頁の「[ウォッチドッグ タイマ設定変更の時間制限手順](#)」を参照してください。

図28. ウォッチドッグ タイマ構成図

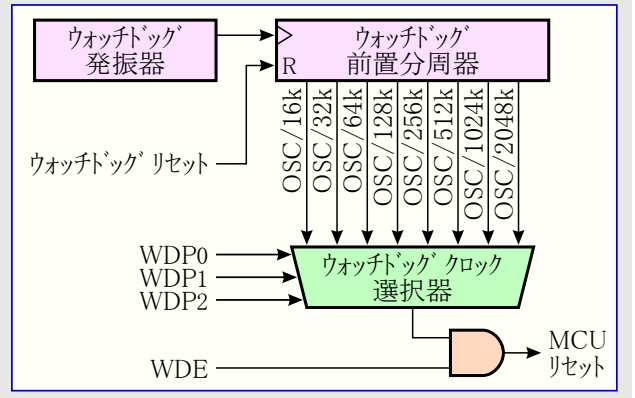


表20. S8515CとWDTONヒューズ設定によるウォッチドッグ機能設定

S8515C ヒューズ	WDTON ヒューズ	安全レベル	WDT初期状態	WDT禁止方法	計時時間変更方法
非プログラム(1)	非プログラム(1)	1	禁止	時間制限	時間制限
非プログラム(1)	プログラム(0)	2	許可	なし(常時許可)	時間制限
プログラム(0)	非プログラム(1)	0	禁止	時間制限	制限なし
プログラム(0)	プログラム(0)	2	許可	なし(常時許可)	時間制限

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

ビット	7	6	5	4	3	2	1	0	
\$21 (\$41)	—	—	—	WDCE	WDE	WDP2	WDP1	WDP0	WDTCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7～5 – Res : 予約 (Reserved)

これらのビットは予約されており、常に0として読まれます。

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

ウォッチドッグ許可(WDE)ビットが論理0を書かれるとき、このビットは設定(1)されなければなりません。さもなければ、ウォッチドッグは禁止されません。一旦1を書かれると、ハードウェアが4クロック周期後、このビットを解除(0)します。ウォッチドッグ禁止手順についてはWDEビットの記述を参照してください。安全レベル1と2では前置分周選択ビットを変更する時も、このビットが設定(1)されなければなりません。31頁の「[ウォッチドッグ タイマ設定変更の時間制限手順](#)」をご覧ください。

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

ウォッチドッグ許可(WDE)が論理1を書かれるとウォッチドッグ タイマが許可され、WDEが論理0を書かれるとウォッチドッグ タイマ機能が禁止されます。WDEはウォッチドッグ変更許可(WDCE)ビットが論理レベル1を持つ場合だけ解除(0)できます。許可されているウォッチドッグ タイマを禁止するには次の手順に従わなければなりません。

1. 同じ操作(命令)内で、ウォッチドッグ変更許可(WDCE)ビットとウォッチドッグ許可(WDE)ビットに論理1を書きます。例え禁止操作を始める前のWDEが1に設定されていても、論理1がWDEに書かれなければなりません。
2. 次からの4クロック周期内で、論理0をWDEに書きます。これがウォッチドッグを禁止します。

安全レベル2では上記の手順でもウォッチドッグ タイマを禁止することができません。31頁の「[ウォッチドッグ タイマ設定変更の時間制限手順](#)」をご覧ください。

■ ビット2~0 - WDP2~0 : ウォッチドッグ タイム前置分周選択 (Watchdog Timer Prescaler 2,1 and 0)

このWDP2~0ビットはウォッチドッグ タイムが許可される時のウォッチドッグ タイムの前置分周を決めます。異なる前置分周値と対応する計時完了周期は表21.に示されます。

表21. ウォッチドッグ前置分周選択

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

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

アセンブリ言語プログラム例

```

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

C言語プログラム例

```

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

ウォッチドッグ タイム設定変更の時間制限手順

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

安全レベル0

この動作種別はAT90S4414/8515にみられるウォッチドッグ操作と互換性があります。ウォッチドッグ タイムは初めに禁止されますが、どんな制限もなくウォッチドッグ許可(WDE)ビットに1を書くことによって許可できます。ウォッチドッグ計時完了周期は制限なしに何時でも変更できます。許可したウォッチドッグを禁止するには30頁のWDEビットの記述で示された手順に従わなければなりません。

安全レベル1

この動作種別ではウォッチドッグ タイムが初めに禁止されますが、どんな制限もなくウォッチドッグ許可(WDE)ビットに1を書くことにより許可できます。ウォッチドッグ計時完了周期を変更または許可したウォッチドッグ タイムを禁止する時に時間制限手順が必要とされます。許可したウォッチドッグ タイムを禁止や、ウォッチドッグ計時完了周期を変更するには次の手順に従わなければなりません。

1. 同じ操作(命令)でウォッチドッグ変更許可(WDCE)とWDEに論理1を書きます。WDEビットの直前の値に拘らず、論理1がWDEに書かれなければなりません。
2. 次からの4クロック周期内に同じ操作(命令)で欲したWDEとウォッチドッグ タイム前置分周選択(WDP2~0)ビットを書きますが、WDCEビットは解除(0)されてです。

安全レベル2

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

1. 同じ操作(命令)でWDCEとWDEに論理1を書きます。WDEビットが常に設定(1)されていても、WDEは時間制限手順を開始するために1を書かれなければなりません。
2. 次からの4クロック周期内に同じ操作(命令)で欲したWDP2~0ビットを書きますが、WDCEビットは解除(0)されてです。WDEビットに書かれた値は無関係です。

割り込み

本項はATmega8515によって実行される割り込み操作の詳細を記述します。AVR割り込み操作の一般説明については8頁の「リセットと割り込みの扱い」を参照してください。

ATmega8515の割り込みベクタ

表22. リセットと割り込みのベクタ

ベクタ番号	プログラム アドレス (注2)	発生元	備考
1	\$000 (注1)	リセット	電源ON, WDT, BOD等の各種リセット
2	\$001	INT0	外部割り込み要求0
3	\$002	INT1	外部割り込み要求1
4	\$003	タイマ/カウンタ1 CAPT	タイマ/カウンタ1捕獲発生
5	\$004	タイマ/カウンタ1 COMPA	タイマ/カウンタ1比較A一致
6	\$005	タイマ/カウンタ1 COMPB	タイマ/カウンタ1比較B一致
7	\$006	タイマ/カウンタ1 OVF1	タイマ/カウンタ1溢れ
8	\$007	タイマ/カウンタ0 OVF0	タイマ/カウンタ0溢れ
9	\$008	SPI STC	SPI 転送完了
10	\$009	USART RX	USART 受信完了
11	\$00A	USART UDRE	USART 送信緩衝部空き
12	\$00B	USART TX	USART 送信完了
13	\$00C	アナログ比較器 ANA_COMP	アナログ比較器出力遷移
14	\$00D	INT2	外部割り込み要求2
15	\$00E	タイマ/カウンタ0 COMP	タイマ/カウンタ0 比較一致
16	\$00F	EEPROM EE_RDY	EEPROM 操作可
17	\$010	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。107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: 一般割り込み制御レジスタ(GICR)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23.はBOOTRST(ヒューズ)とIVSEL(割り込みベクタ選択ビット)の様々な組み合わせに対するリセットと割り込みベクタの配置を示します。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードを置けます。これはリセットベクタが応用領域の一方、割り込みベクタがブート領域、またはその逆の場合でも同様です。

表23. リセットと割り込みベクタの配置

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

注: ブート領域先頭アドレスは115頁の表78.で示されます。

ATmega8515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		RJMP RESET	;各種リセット
\$0001		RJMP EXT_INT0	;外部割り込み要求0
\$0002		RJMP EXT_INT1	;外部割り込み要求1
\$0003		RJMP TIM1_CAPT	;タイマ/カウンタ1捕獲発生
\$0004		RJMP TIM1_COMPA	;タイマ/カウンタ1比較A一致
\$0005		RJMP TIM1_COMPB	;タイマ/カウンタ1比較B一致
\$0006		RJMP TIM1_OVF	;タイマ/カウンタ1溢れ
\$0007		RJMP TIM0_OVF	;タイマ/カウンタ0溢れ
\$0008		RJMP SPI_STC	;SPI転送完了
\$0009		RJMP USART_RXC	;USART受信完了
\$000A		RJMP USART_DRE	;USART送信緩衝部空
\$000B		RJMP USART_TXC	;USART送信完了
\$000C		RJMP ANA_COMP	;アナログ比較器出力遷移
\$000D		RJMP EXT_INT2	;外部割り込み要求2
\$000E		RJMP TIM0_COMP	;タイマ/カウンタ0 比較一致
\$000F		RJMP EE_RDY	;EEPROM操作可
\$0010		RJMP SPM_RDY	;SPM命令操作可
;			
\$0011	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$0012		OUT SPH, R16	;スタック ポインタ上位を初期化
\$0013		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0014		OUT SPL, R16	;スタック ポインタ下位を初期化
;			
;以下、I/O初期化など			

BOOTRSTヒューズが非プログラム(1)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立って一般割り込み制御レジスタ(GICR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (応用プログラム開始)
\$0001		OUT SPH, R16	;スタック ポインタ上位を初期化
\$0002		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0003		OUT SPL, R16	;スタック ポインタ下位を初期化
;			
;以下、I/O初期化など			
	. ORG	\$0C01	;ブート プログラム領域が2Kバイトの場合
\$0C01		RJMP EXT_INT0	;外部割り込み要求0
\$0C02		RJMP EXT_INT1	;外部割り込み要求1
;			
\$0C10		RJMP SPM_RDY	;SPM命令操作可

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

アドレス	ラベル	命令	注釈
		. ORG \$0001	; 割り込みベクタ先頭
\$0001		RJMP EXT_INT0	; 外部割り込み要求0
\$0002		RJMP EXT_INT1	; 外部割り込み要求1
		}	
\$0010		RJMP SPM_RDY	; SPM命令操作可
		}	; 以下、プログラムなど
		. ORG \$0C00	; ブートプログラム領域が2Kバイトの場合
\$0C00	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		OUT SPH, R16	; スタックポインタ上位を初期化
\$0C02		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$0C03		OUT SPL, R16	; スタックポインタ下位を初期化
		}	; 以下、I/O初期化など

BOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立って一般割り込み制御レジスタ(GICR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		. ORG \$0C00	; ブートプログラム領域が2Kバイトの場合
\$0C00		RJMP RESET	; 各種リセット (BOOTRSTヒューズ=0)
\$0C01		RJMP EXT_INT0	; 外部割り込み要求0
\$0C02		RJMP EXT_INT1	; 外部割り込み要求1
		}	
\$0C10		RJMP SPM_RDY	; SPM命令操作可
;			
\$0C11	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$0C12		OUT SPH, R16	; スタックポインタ上位を初期化
\$0C13		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$0C14		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

応用領域とブート領域間の割り込みベクタ移動

一般割り込み制御レジスタ(GICR)は割り込みベクタ表の配置を制御します。

■ 一般割り込み制御レジスタ (General Interrupt Control Register) GICR

ビット	7	6	5	4	3	2	1	0	
\$3B (\$5B)	INT1	INT0	INT2	—	—	—	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

IVSELビットが解除(0)されると、割り込みベクタはフラッシュ メモリの先頭に配置されます。このビットが設定(1)されると、割り込みベクタはフラッシュ メモリのブート ロード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107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」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 ロード領域に配置され、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 ロード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107頁の「ブート ロード支援 – 書き込み中読み出し可能な自己プログラミング」を参照してください。

■ 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, GICR          ; 現GICR値取得
              MOV     R17, R16          ; 現GICR値複写
              ORI     R16, (1<<IVCE)    ; IVCE論理1値を取得
              OUT     GICR, R16         ; IVCEに論理1書き込み
              ORI     R17, (1<<IVSEL)    ; IVSEL論理1値を取得
              OUT     GICR, 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

```

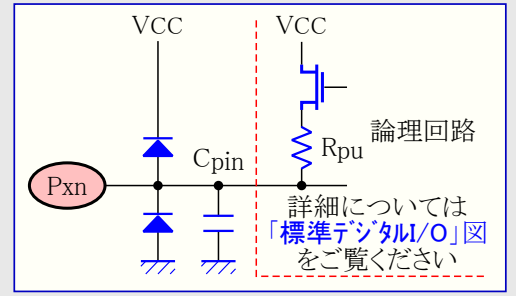
void Move_interrupts(void)
{
    uchar temp;                          /* 一時定数定義 */
    temp = GICR;                         /* 現GICR値取得 */
    GICR = temp | (1<<IVCE);             /* IVCEに論理1書き込み */
    GICR = temp | (1<<IVSEL);            /* ブート領域へ割り込みベクタを移動 */
}
    
```


入出力ポート

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時に真の読み-変更-書き(リード-モディファイライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方向をも無意識に変更することなく、1つのポートピンの方向を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。このピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピンは個別に選択可能な、供給電圧で抵抗値が変化しないプルアップ抵抗を持っています。全てのI/Oピンは図29で示されるようにVCCとGNDの両方に保護ダイオードを持っています。各値の完全な一覧については、128ページの「電気的特性」を参照してください。

図29. 入出力ピン等価回路



本項内の全てのレジスタとビットの参照は一般形で記されます。小文字の'*x*'はポート番号文字、小文字の'*n*'はビット番号を表します。けれどもプログラム内でレジスタやビット定義に使うとき、正確な形式(例えば、ここで一般に記されたPORTxnがポートBのビット3に対してはPORTB3)が使われなければなりません。物理的なI/Oレジスタとビット位置は47ページの「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。加えて特殊I/O機能レジスタ(SFIOR)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。

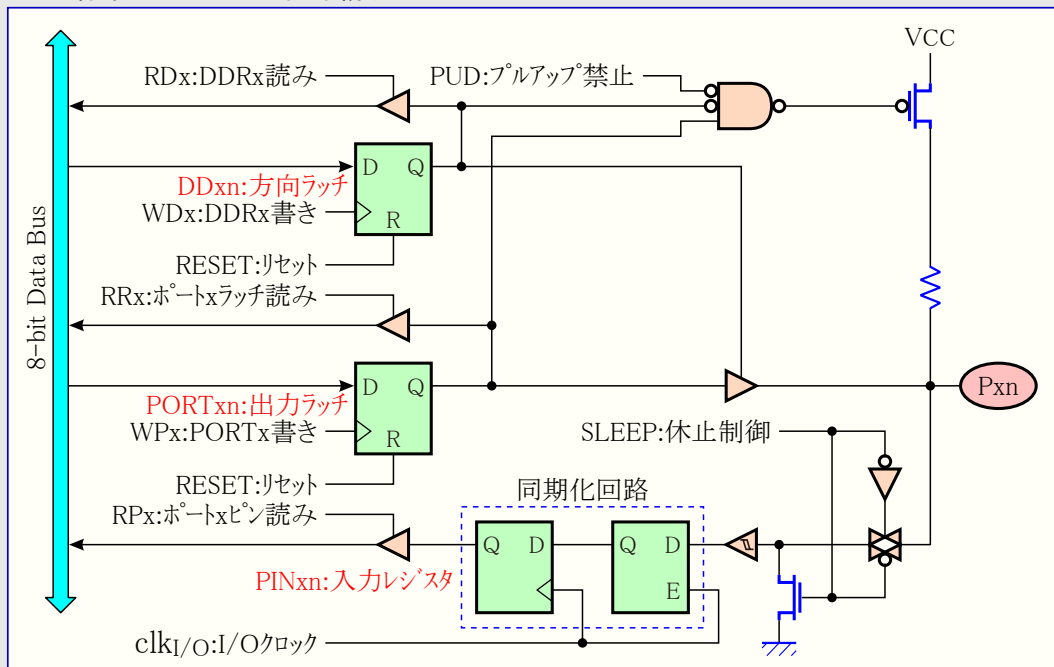
標準デジタルI/OとしてのI/Oポートの使用は次の「標準デジタル入出力としてのポート」で記述されます。多くのポートピンはデバイスの周辺機能用の交換機能と多重化されます。ポートピンとの各交換機能のインターフェース法は39ページの「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項を参照してください。

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

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30.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30. 標準デジタル入出力回路構成



注: 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。

ピンの設定

各ポートピンは3つのレジスタビット、**DDxn**、**PORTxn**、**PINxn**から成ります。47頁の「**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択します。DDxnが論理**1**を書かれるとPxnは出力ピンとして設定されます。DDxnが論理**0**を書かれるとPxnは入力ピンとして設定されます。

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

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

Hi-Z入力(DDxn=0, PORTxn=0)とHigh出力(DDxn=1, PORTxn=1)間の切り替え時、プルアップ許可入力(DDxn=0, PORTxn=1)またはLow出力(DDxn=1, PORTxn=0)のどちらかの中間状態が生じるに違いありません。通常、高インピーダンス環境は強力なHigh(吐き出し)駆動部とプルアップ間の違いに気付かないので、プルアップが許可された状態は十分受け入れられます。この事例でないなら、全ポートの全プルアップを禁止するのに**特殊I/O機能レジスタ(SFIOR)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24.はピン値に対する制御信号の一覧を示します。

表24. ポートピンの設定

DDxn	PORTxn	PUD (SFIOR)	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

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30.で示されるようにPINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックのエッジ付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。図31.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31.で)システムクロックの最初の上昇端の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32.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31. 外部供給ピン値読み込み時の同期化

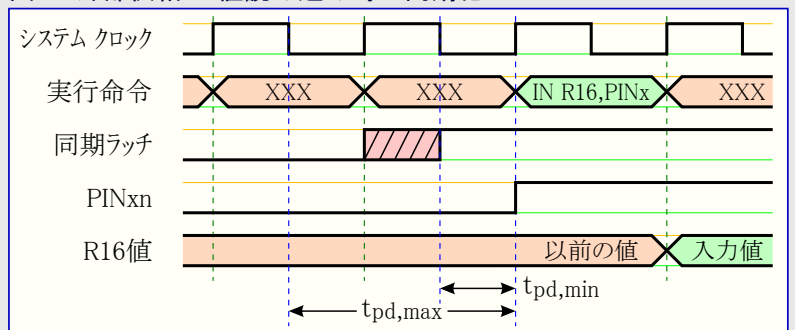
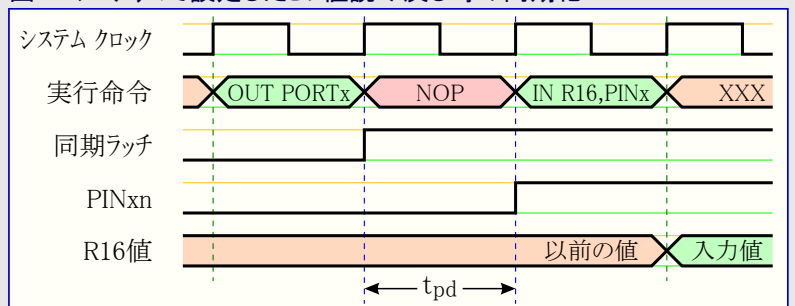


図32. プログラムで設定したピン値読み戻し時の同期化



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 */
_NOP(); /* 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。

デジタル入力許可と休止形態

図30.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**、**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39頁の「**交換ポート機能**」で記載されるように様々な他の交換機能によっても無視されます。

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

未接続ピン

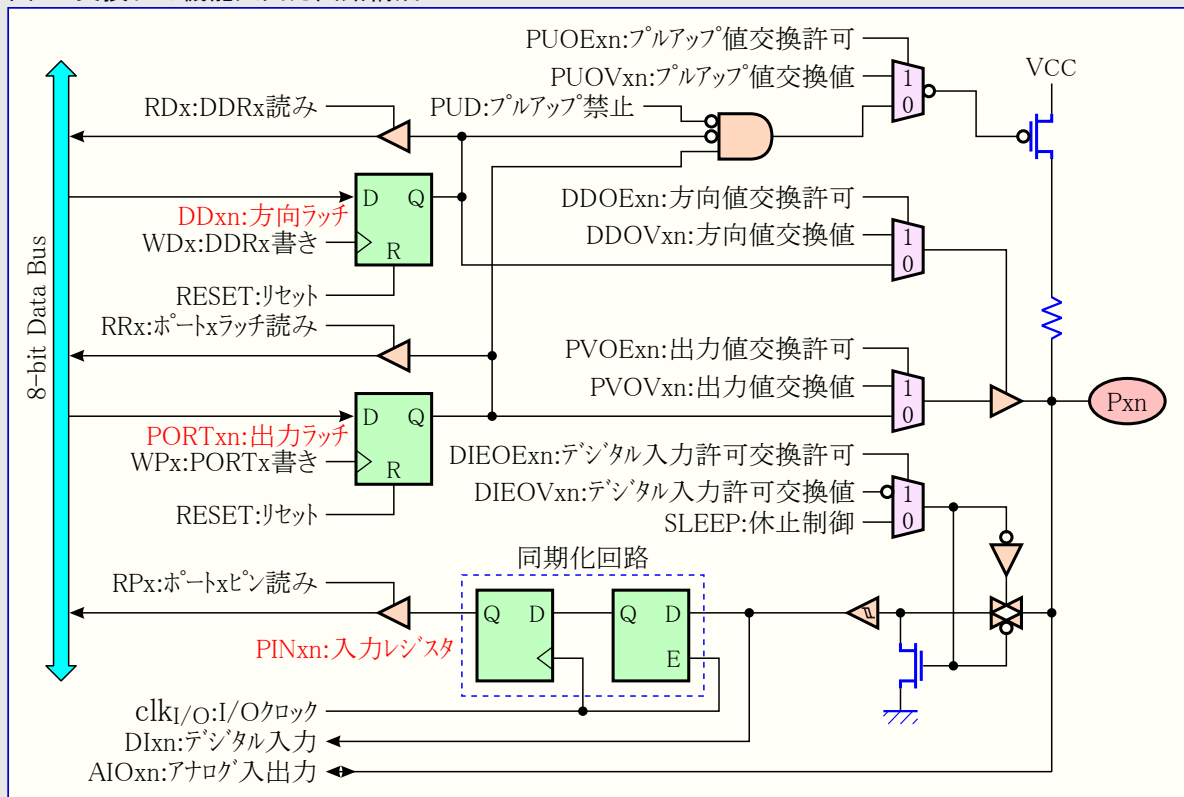
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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

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33.は単純化された図30.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVR マイクロ コントローラ系統の全ポートピンに適用できる一般的な記述として取り扱います。

図33. 交換ポート機能入出力回路構成



注: 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25.は重複(交換)信号の機能一覧を示します。図33.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25. 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。
DIEOE	デジタル入力許可交換許可	1で、デジタル入力許可はDIEOV信号で制御され、0の場合、MCUの状態(活動動作、休止形態)によって決定されます。
OIEOV	デジタル入力許可交換値	DIEOE=1時、MCUの状態(活動動作、休止形態)に関係なく、デジタル入力を許可(1)/禁止(0)します。
DI	デジタル入力	交換機能用デジタル入力です。この信号は図上でシュミットトリガ出力に接続されていますが、これは同期化前となります。この信号はクロックとしての使用を除き、各交換機能自身が同期化します。
AIO	アナログ入出力	交換機能用アナログ入出力です。この信号はピンに直接接続され、双方向使用ができます。

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

■ 特殊I/O機能レジスタ (Special Function I/O Register) SFIOR

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	–	XMBK	XMM2	XMM1	XMM0	PUD	–	PSR10	SFIOR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット2 – 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37頁の「[ピン設定](#)」をご覧ください。

ポートAの交換機能

ポートAには外部メモリ インターフェース用のアドレス下位8ビットとデータ信号線としての交換機能があります。

表26. ポートAピンの交換機能

ポートピン	交換機能
PA7	AD7 (外部メモリ インターフェース アドレス/データ ビット7)
PA6	AD6 (外部メモリ インターフェース アドレス/データ ビット6)
PA5	AD5 (外部メモリ インターフェース アドレス/データ ビット5)
PA4	AD4 (外部メモリ インターフェース アドレス/データ ビット4)
PA3	AD3 (外部メモリ インターフェース アドレス/データ ビット3)
PA2	AD2 (外部メモリ インターフェース アドレス/データ ビット2)
PA1	AD1 (外部メモリ インターフェース アドレス/データ ビット1)
PA0	AD0 (外部メモリ インターフェース アドレス/データ ビット0)

表27.と表28.はポートAの交換機能を39頁の図33.で示される交換信号に関連付けます。

表27. ポートA7～4の交換機能用交換信号

信号名	PA7/AD7	PA6/AD6	PA5/AD5	PA4/AD4
PUOE	SRE	SRE	SRE	SRE
PUOV	($\overline{\text{WR}} \text{ADA}$)・PORTA7	($\overline{\text{WR}} \text{ADA}$)・PORTA6	($\overline{\text{WR}} \text{ADA}$)・PORTA5	($\overline{\text{WR}} \text{ADA}$)・PORTA4
DDOE	SRE	SRE	SRE	SRE
DDOV	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$
PVOE	SRE	SRE	SRE	SRE
PVOV	A7・ADA D7出力・ $\overline{\text{WR}}$	A6・ADA D6出力・ $\overline{\text{WR}}$	A5・ADA D5出力・ $\overline{\text{WR}}$	A4・ADA D4出力・ $\overline{\text{WR}}$
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	D7入力	D6入力	D5入力	D4入力
AIO	–	–	–	–

表28. ポートA3～0の交換機能用交換信号

信号名	PA3/AD3	PA2/AD2	PA1/AD1	PA0/AD0
PUOE	SRE	SRE	SRE	SRE
PUOV	($\overline{\text{WR}} \text{ADA}$)・PORTA3	($\overline{\text{WR}} \text{ADA}$)・PORTA2	($\overline{\text{WR}} \text{ADA}$)・PORTA1	($\overline{\text{WR}} \text{ADA}$)・PORTA0
DDOE	SRE	SRE	SRE	SRE
DDOV	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$	$\overline{\text{WR}} \text{ADA}$
PVOE	SRE	SRE	SRE	SRE
PVOV	A3・ADA D3出力・ $\overline{\text{WR}}$	A2・ADA D2出力・ $\overline{\text{WR}}$	A1・ADA D1出力・ $\overline{\text{WR}}$	A0・ADA D0出力・ $\overline{\text{WR}}$
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	D3入力	D2入力	D1入力	D0入力
AIO	–	–	–	–

注: ADAはアドレス有効(ADdress Active)の略でアドレスが出力される時を表します。14頁の「[外部メモリ インターフェース](#)」をご覧ください。

ポートBの交換機能

ポートBピンの交換機能は表29.で示されます。

表29. ポートBピンの交換機能

ポートピン	交換機能
PB7	SCK (SPI 直列クロック 主装置側出力/従装置側入力)
PB6	MISO (SPI 主装置側データ入力/従装置側データ出力)
PB5	MOSI (SPI 主装置側データ出力/従装置側データ入力)
PB4	$\overline{SS}$ (SPI 従装置選択入力)
PB3	AIN1 (アナログ比較器反転入力)
PB2	AIN0 (アナログ比較器非反転入力)
PB1	T1 (タイマ/カウンタ1 外部クロック入力)
PB0	T0 (タイマ/カウンタ0 外部クロック入力) OC0 (タイマ/カウンタ0 比較一致出力)

交換ピンの設定は次のとおりです。

・ SCK – ポートB ビット7 : PB7

SCK : SPIチャネル用の主装置クロック出力、従装置クロック入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB7設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB7によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB7によって制御できます。

・ MISO – ポートB ビット6 : PB6

MISO : SPIチャネル用の主装置データ入力、従装置データ出力。SPIが主装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB6の設定に拘らず、入力として設定されます。SPIが従装置として許可されると、このピンのデータ方向はDDB6によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB6によって制御できます。

・ MOSI – ポートB ビット5 : PB5

MOSI : SPIチャネル用の主装置データ出力、従装置データ入力。SPIが従装置として許可されると、このピンはポート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、このピンのデータ方向はDDB5によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB5によって制御できます。

・ $\overline{SS}$ – ポートB ビット4 : PB4

$\overline{SS}$: SPI従装置選択入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB4の設定に拘らず、入力として設定されます。従装置としてこのピンがLowに駆動されると、SPI(機能)が活性化(有効に)されます。SPIが主装置として許可されると、このピンのデータ方向はDDB4によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB4によって制御できます。

・ AIN1 – ポートB ビット3 : PB3

AIN1 : アナログ比較器反転入力。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。

・ AIN0 – ポートB ビット2 : PB2

AIN0 : アナログ比較器非反転入力。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。

・ T1 – ポートB ビット1 : PB1

T1 : タイマ/カウンタ1の外部クロック入力ピンです。

・ T0/OC0 – ポートB ビット0 : PB0

T0 : タイマ/カウンタ0の外部クロック入力ピンです。

OC0 : タイマ/カウンタ0の比較一致出力。PB0ピンはタイマ/カウンタ0の比較一致用外部出力として扱えます。この機能を扱うため、PB0ピンは出力として設定(DDB0=1)されなければなりません。このOC0ピンはタイマ機能のPWM動作出力ピンでもあります。

表30.と表31.はポートBの交換機能を39頁の図33.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30. ポートB7～4の交換機能用交換信号

信号名	PB7/SCK	PB6/MISO	PB5/MOSI	PB4/SS
PUOE	SPE・MSTR	SPE・MSTR	SPE・MSTR	SPE・MSTR
PUOV	PORTB7・PUD	PORTB6・PUD	PORTB5・PUD	PORTB4・PUD
DDOE	SPE・MSTR	SPE・MSTR	SPE・MSTR	SPE・MSTR
DDOV	0	0	0	0
PVOE	SPE・MSTR	SPE・MSTR	SPE・MSTR	0
PVOV	SCK出力	SPI従装置出力	SPI主装置出力	0
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	SCK入力	SPI主装置入力	SPI従装置入力	SPI SS
AIO	—	—	—	—

表31. ポートB3～0の交換機能用交換信号

信号名	PB3/AIN1	PB2/AIN0	PB1/T1	PB0/T0/OC0
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	1	0	0	0
PVOE	0	0	0	OC0許可
PVOV	0	0	0	OC0
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	—	—	T1入力	T0入力
AIO	AIN1(反転)入力	AIN0(非反転)入力	—	—

ポートCの交換機能

ポートCの交換機能は表32.で示されます。

表32. ポートCピンの交換機能

ポートピン	交換機能
PC7	A15 (外部メモリ アドレス ビット15)
PC6	A14 (外部メモリ アドレス ビット14)
PC5	A13 (外部メモリ アドレス ビット13)
PC4	A12 (外部メモリ アドレス ビット12)
PC3	A11 (外部メモリ アドレス ビット11)
PC2	A10 (外部メモリ アドレス ビット10)
PC1	A9 (外部メモリ アドレス ビット9)
PC0	A8 (外部メモリ アドレス ビット8)

- ・ A15 – ポートC ビット7 : PC7 : 外部メモリ インターフェース アドレス15
- ・ A14 – ポートC ビット6 : PC6 : 外部メモリ インターフェース アドレス14
- ・ A13 – ポートC ビット5 : PC5 : 外部メモリ インターフェース アドレス13
- ・ A12 – ポートC ビット4 : PC4 : 外部メモリ インターフェース アドレス12
- ・ A11 – ポートC ビット3 : PC3 : 外部メモリ インターフェース アドレス11
- ・ A10 – ポートC ビット2 : PC2 : 外部メモリ インターフェース アドレス10
- ・ A9 – ポートC ビット1 : PC1 : 外部メモリ インターフェース アドレス9
- ・ A8 – ポートC ビット0 : PC0 : 外部メモリ インターフェース アドレス8

表33.と表34.はポートCの交換機能を39頁の図33.で示される交換信号に関連付けます。

表33. ポートC7～4の交換機能用交換信号

信号名	PC7/A15	PC6/A14	PC5/A13	PC4/A12
PUOE	SRE・(XMM<1)	SRE・(XMM<2)	SRE・(XMM<3)	SRE・(XMM<4)
PUOV	0	0	0	0
DDOE	SRE・(XMM<1)	SRE・(XMM<2)	SRE・(XMM<3)	SRE・(XMM<4)
DDOV	1	1	1	1
PVOE	SRE・(XMM<1)	SRE・(XMM<2)	SRE・(XMM<3)	SRE・(XMM<4)
PVOV	A15	A14	A13	A12
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	–	–	–	–
AIO	–	–	–	–

表34. ポートC3～0の交換機能用交換信号

信号名	PC3/A11	PC2/A10	PC1/A9	PC0/A8
PUOE	SRE・(XMM<5)	SRE・(XMM<6)	SRE・(XMM<7)	SRE・(XMM<7)
PUOV	0	0	0	0
DDOE	SRE・(XMM<5)	SRE・(XMM<6)	SRE・(XMM<7)	SRE・(XMM<7)
DDOV	1	1	1	1
PVOE	SRE・(XMM<5)	SRE・(XMM<6)	SRE・(XMM<7)	SRE・(XMM<7)
PVOV	A11	A10	A9	A8
DIOE	0	0	0	0
DIOV	0	0	0	0
DI	–	–	–	–
AIO	–	–	–	–

ポートDの交換機能

ポートDピンの交換機能は表35.で示されます。

表35. ポートDピンの交換機能

ポートピン	交換機能
PD7	$\overline{RD}$ (外部メモリ用読み出しスロープ信号出力)
PD6	$\overline{WR}$ (外部メモリ用書き込みスロープ信号出力)
PD5	OC1A (タイマ/カウンタ1 比較A一致出力)
PD4	XCK (USART 外部クロック入出力)
PD3	INT1 (外部割り込み1 入力)
PD2	INT0 (外部割り込み0 入力)
PD1	TXD (USART 送信データ出力)
PD0	RXD (USART 受信データ入力)

交換ピンの設定は次のとおりです。

- $\overline{RD}$ – ポートD ビット7 : PD7

$\overline{RD}$: 外部メモリ用読み出し制御スロープ信号出力ピンです。

- $\overline{WR}$ – ポートD ビット6 : PD6

$\overline{WR}$: 外部メモリ用書き込み制御スロープ信号出力ピンです。

- OC1A – ポートD ビット5 : PD5

OC1A : タイマ/カウンタ1の比較A一致出力。PD5ピンはタイマ/カウンタ1の比較A一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(DDD5=1)されなければなりません。このOC1Aピンはタイマ機能のPWM動作出力ピンでもあります。

- XCK – ポートD ビット4 : PD4

XCK : USARTの外部クロック入出力。ポートD方向レジスタ(DDRD)のDDD4は、このクロックが入力(DDD4=0)または出力(DDD4=1)のどちらかを制御します。XCKピンはUSARTが同期種別で動作する時だけ活動します(有効です)。

- INT1 – ポートD ビット3 : PD3

INT1 : 外部割り込み1入力。PD3ピンは外部割り込み元として扱えます。

- INT0 – ポートD ビット2 : PD2

INT0 : 外部割り込み0入力。PD2ピンは外部割り込み元として扱えます。

- TXD – ポートD ビット1 : PD1

TXD : 送信データ(USART用データ出力ピン)。USART送信部が許可されると、このピンはDDRDのDDD1の値に拘らず、出力として設定されます。

- RXD – ポートD ビット0 : PD0

RXD : 受信データ(USART用データ入力ピン)。USART受信部が許可されると、このピンはDDRDのDDD0の値に拘らず、入力として設定されます。USARTがこのピンを入力に強制するとき、プルアップは未だPORTD0ビットによって制御できます。

表36.と表37.はポートDの交換機能を39頁の図33.で示される交換信号に関連付けます。

表36. ポートD7～4の交換機能用交換信号

信号名	PD7/RD	PD6/ $\overline{\text{WR}}$	PD5/OC1A	PD4/XCK
PUOE	SRE	SRE	0	0
PUOV	0	0	0	0
DDOE	SRE	SRE	0	0
DDOV	1	1	0	0
PVOE	SRE	SRE	OC1A許可	XCK出力許可
PVOV	$\overline{\text{RD}}$	$\overline{\text{WR}}$	OC1A	XCK出力
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	–	–	–	XCK入力
AIO	–	–	–	–

表37. ポートD3～0の交換機能用交換信号

信号名	PD3/INT1	PD2/INT0	PD1/TXD	PD0/RXD
PUOE	0	0	TXEN	RXEN
PUOV	0	0	0	PORTD0・ $\overline{\text{PUD}}$
DDOE	0	0	TXEN	RXEN
DDOV	0	0	1	0
PVOE	0	0	TXEN	0
PVOV	0	0	TXD	0
DIEOE	INT1許可	INT0許可	0	0
DIEOV	1	1	0	0
DI	INT1入力	INT0入力	–	RXD
AIO	–	–	–	–

ポートEの交換機能

ポートEの交換機能は表38.で示されます。

表38. ポートEピンの交換機能

ポートピン	交換機能
PE2	OC1B (タイマ/カウンタ1 比較B一致出力)
PE1	ALE (外部メモリ用アドレス ラッチ許可信号出力)
PE0	ICP (タイマ/カウンタ1 捕獲起動入力) INT2 (外部割り込み2 入力)

交換ピンの設定は次のとおりです。

・ OC1B – ポートE ビット2 : PE2

OC1B : タイマ/カウンタ1の比較B一致出力。PE2ピンはタイマ/カウンタ1の比較B一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(DDE2=1)されなければなりません。このOC1Bピンはタイマ機能のPWM動作出力ピンでもあります。

・ ALE – ポートE ビット1 : PE1

ALE : 外部メモリ用アドレス ラッチ許可信号出力ピンです。

・ ICP/INT2 – ポートE ビット0 : PE0

ICP : タイマ/カウンタ1の捕獲起動入力。PE0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。

INT2 : 外部割り込み2入力。PE0ピンは外部割り込み元として扱えます。

表39.はポートEの交換機能を39頁の図33.で示される交換信号に関連付けます。

表39. ポートE2～0の交換機能用交換信号

信号名	PE2//OC1B	PE1/ALE	PE0/ICP/INT2
PUOE	0	SRE	0
PUOV	0	0	0
DDOE	0	SRE	0
DDOV	0	1	0
PVOE	OC1B許可	SRE	0
PVOV	OC1B	ALE	0
DIOE	0	0	INT2許可
DIOV	0	0	1
DI	–	–	ICP入力/INT2入力
AIO	–	–	–

I/Oポート用レジスタ

■ ポートA出カレジスタ (Port A Data Register) PORTA

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

■ ポートA方向レジスタ (Port A Data Direction Register) DDRA

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

■ ポートA入カレジスタ (Port A Input Address) PINA

ビット	7	6	5	4	3	2	1	0	
\$19 (\$39)	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
Read/Write	R	R	R	R	R	R	R	R	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

■ ポートB出カレジスタ (Port B Data Register) PORTB

ビット	7	6	5	4	3	2	1	0	
\$18 (\$38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートB方向レジスタ (Port B Data Direction Register) DDRB

ビット	7	6	5	4	3	2	1	0	
\$17 (\$37)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートB入カレジスタ (Port B Input Address) PINB

ビット	7	6	5	4	3	2	1	0	
\$16 (\$36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R	R	R	R	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

■ ポートC出カレジスタ (Port C Data Register) PORTC

ビット	7	6	5	4	3	2	1	0	
\$15 (\$35)	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	PORTC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートC方向レジスタ (Port C Data Direction Register) DDRC

ビット	7	6	5	4	3	2	1	0	
\$14 (\$34)	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートC入力レジスタ (Port C Input Address) PINC

ビット	7	6	5	4	3	2	1	0	
\$13 (\$33)	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	PINC
Read/Write	R	R	R	R	R	R	R	R	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

■ ポートD出レジスタ (Port D Data Register) PORTD

ビット	7	6	5	4	3	2	1	0	
\$12 (\$32)	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートD方向レジスタ (Port D Data Direction Register) DDRD

ビット	7	6	5	4	3	2	1	0	
\$11 (\$31)	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートD入力レジスタ (Port D Input Address) PIND

ビット	7	6	5	4	3	2	1	0	
\$10 (\$30)	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R	R	R	R	R	R	R	R	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

■ ポートE出レジスタ (Port E Data Register) PORTE

ビット	7	6	5	4	3	2	1	0	
\$07 (\$27)	—	—	—	—	—	PORTE2	PORTE1	PORTE0	PORTE
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ポートE方向レジスタ (Port E Data Direction Register) DDRE

ビット	7	6	5	4	3	2	1	0	
\$06 (\$26)	—	—	—	—	—	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	

■ ポートE入力レジスタ (Port E Input Address) PINE

ビット	7	6	5	4	3	2	1	0	
\$05 (\$25)	—	—	—	—	—	PINE2	PINE1	PINE0	PINE
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	不定	不定	不定	

外部割り込み

外部割り込みはINT0,INT1,INT2ピンによって起動されます。許可したなら、例えばINT0～2ピンが出力として設定されても、割り込みが起動することに注目してください。この特徴はソフトウェア割り込みを生成する方法を提供します。外部割り込みは上昇端または下降端(含む両端)、またはLowレベル(INT2はエッジ起動のみ)によって起動できます。これは**MCU制御レジスタ(MCUCR)**と**拡張MCU制御レジスタ(EMCUCR)**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(INT0/INT1のみ)、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT0とINT1の上昇端または下降端割り込みの認知は20頁の「**クロックシステムとその配給**」で記述されるI/Oクロックの存在を必要とすることに注意してください。INT0/INT1のLowレベル割り込みとINT2のエッジ割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイスを起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

レベル起動割り込みがパワーダウン動作からの起動に使われる場合、変化したレベルはMCUを起動するために一定時間保たね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5.0V,25℃で(公称)1μsです。ウォッチドッグ発振器周波数は134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20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ねばなりません。

■ MCU制御レジスタ (MCU Control Register) MCUCR

このMCU制御レジスタは一般的なMCU機能と割り込み判断制御用の制御ビットを含みます。

ビット	7	6	5	4	3	2	1	0	
\$35 (\$55)	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	MCUCR
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3,2 – ISC11,0 : 外部割り込み1条件制御 (Interrupt Sense Control 1 bit1 and 0)

外部割り込み1はステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み制御レジスタ(GICR)の**外部割り込み1許可(INT1)ビット**が設定(1)される場合のINT1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INT1ピンのエッジとレベルは表40.で定義されます。INT1ピンの値はエッジ検知前に採取されています。エッジまたは論理変化割り込みが選択される場合、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。Lowレベル割り込みが選択される場合、Lowレベルは割り込みを生成するために現在実行している命令の完了まで保たねばなりません。

表40. 外部割り込み1(INT1)割り込み条件

ISC11	ISC10	INT1ピン割り込み発生条件
0	0	Lowレベル
0	1	論理変化(両端)
1	0	下降端
1	1	上昇端

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

外部割り込み0はステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み制御レジスタ(GICR)の**外部割り込み0許可(INT0)ビット**が設定(1)される場合のINT0外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT0ピンのエッジとレベルは表41.で定義されます。INT0ピンの値はエッジ検知前に採取されています。エッジまたは論理変化割り込みが選択される場合、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。Lowレベル割り込みが選択される場合、Lowレベルは割り込みを生成するために現在実行している命令の完了まで保たねばなりません。

表41. 外部割り込み0(INT0)割り込み条件

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

■ 拡張MCU制御レジスタ (Extended MCU Control Register) EMCUCR

ビット	7	6	5	4	3	2	1	0	
\$36 (\$56)	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	EMCUCR
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0 – ISC2 : 外部割り込み2条件制御 (Interrupt Sense Control 2)

非同期外部割り込み2はステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み制御レジスタ(GICR)の**外部割り込み2許可(INT2)ビット**が設定(1)される場合のINT2外部ピンによって活性(有効)にされます。ISC2が0を書かれるとINT2ピンの下降端が割り込みを活性にします。ISC2が1を書かれるとINT2ピンの上昇端が割り込みを活性にします。INT2のエッジは非同期に記録されます。表42.で与えられた**最小パルス幅**より広いINT2のパルスは割り込みを生成します。より短いパルスは割り込みの発生が保証されません。ISC2ビット変更時、割り込みが起き得ます。従って最初にGICRの外部割り込み2許可(INT2)ビットを解除(0)することでINT2割り込みを禁止することが推奨されます。その後ISC2ビットは変更できます。最後にINT2割り込み要求フラグは割り込みが再許可される前に**一般割り込み要求フラグレジスタ(GIFR)の外部割り込み2要求(INTF2)フラグ**に論理1を書くことによって解除(0)されるべきです。

表42. 非同期外部割り込み特性

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

■ 一般割り込み制御レジスタ (General Interrupt Control Register) GICR

ビット	7	6	5	4	3	2	1	0	
\$3B (\$5B)	INT1	INT0	INT2	—	—	—	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R	R	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7 – INT1：外部割り込み1許可 (External Interrupt Request 1 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、INT1ビットが設定(1)されると、INT1外部ピン割り込みが許可されます。MCU制御レジスタ(MCUCR)の割り込み条件制御1のビット1と0(ISC11,0)はこの外部割り込みがINT1ピンの上昇端、下降端、両端、またはLowレベルのどれで活性(有効)にされるかを定義します。例えばINT1ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求1に対応する割り込みはINT1割り込みベクタから実行されます。

■ ビット6 – INT0：外部割り込み0許可 (External Interrupt Request 0 Enable)

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

■ ビット5 – INT2：外部割り込み2許可 (External Interrupt Request 2 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ビットが設定(1)されると、INT2外部ピン割り込みが許可されます。拡張MCU制御レジスタ(EMCUCR)の割り込み条件制御2(ISC2)ビットは、この外部割り込みがINT2ピンの上昇端または下降端のどちらで活性(有効)にされるかを定義します。例えばINT2ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求2に対する割り込みはINT2割り込みベクタから実行されます。

■ 一般割り込み要求フラグ レジスタ (General Interrupt Flag Register) GIFR

ビット	7	6	5	4	3	2	1	0	
\$3A (\$5A)	INTF1	INTF0	INTF2	—	—	—	—	—	GIFR
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 – INTF1：外部割り込み1要求フラグ (External Interrupt Flag1)

INT1ピン上のエッジまたは論理変化が割り込み要求を起動すると、INTF1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)ビットと一般割り込み制御レジスタ(GICR)の外部割り込み1許可(INT1)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み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。INT1がレベル割り込みとして設定されると、このフラグは常に解除(0)されます。

■ ビット6 – INTF0：外部割り込み0要求フラグ (External Interrupt Flag0)

INT0ピン上のエッジまたは論理変化が割り込み要求を起動すると、INTF0が設定(1)になります。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと一般割り込み制御レジスタ(GICR)の外部割り込み0許可(INT0)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは論理1を書くことによっても解除(0)できます。INT0がレベル割り込みとして設定されると、このフラグは常に解除(0)されます。

■ ビット5 – INTF2：外部割り込み2要求フラグ (External Interrupt Flag2)

INT2ピン上の出来事が割り込み要求を起動すると、INTF2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)ビットと一般割り込み制御レジスタ(GICR)の外部割り込み2許可(INT2)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割り込み禁止で或る種の休止形態へ移行すると、このピンの入力緩衝部が禁止されることに注意してください。これはINTF2フラグを設定(1)する内部信号の論理変化を引き起こすかもしれません。より多くの情報については38頁の「デジタル入力許可と休止形態」をご覧ください。

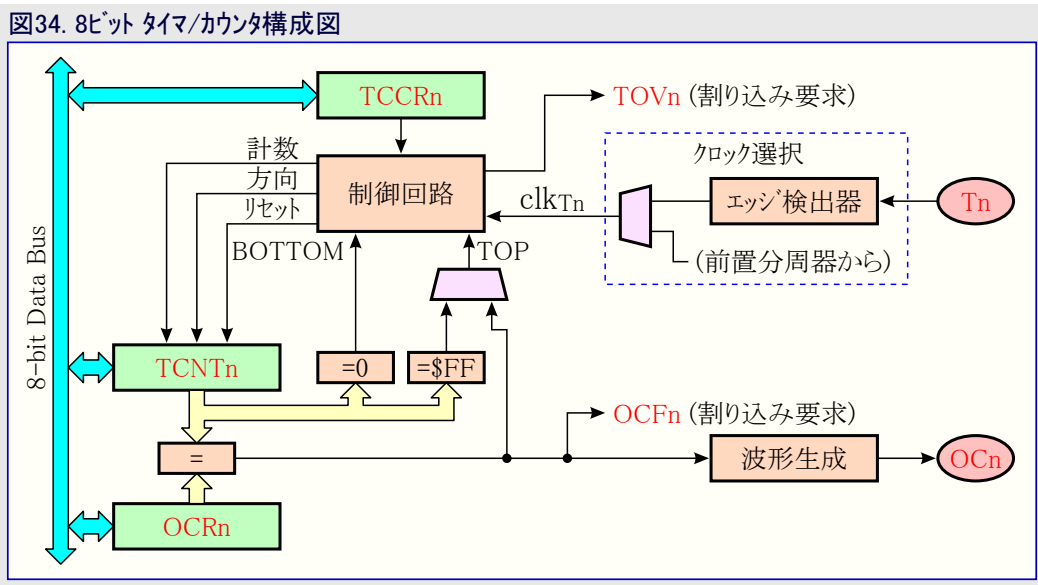
8ビット タイマ/カウンタ0 (PWM付き)

タイマ/カウンタ0は単一比較部付きの汎用8ビット タイマ/カウンタ部です。主な特徴は次のとおりです。

- ・ 単一比較部付き計数器
- ・ 比較一致でのタイマ/カウンタ解除 (自動再設定)
- ・ 不具合なしで正しい位相のパルス幅変調器 (PWM)
- ・ 周波数発生器
- ・ クロック用10ビット前置分周器
- ・ 溢れと比較一致割り込み (TOV0とOCF0)

概要

この8ビット タイマ/カウンタの簡易化した構成図は図34.で示されます。I/Oピンの実際の配置については1頁の「ピン配置」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注:原文は太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は59頁の「8ビット タイマ/カウンタ用レジスタ」で一覧されます。



関係レジスタ

タイマ/カウンタ(TCNT0)と比較レジスタ(OCR0)は8ビットのレジスタです。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKはこれらのレジスタが他のタイマ/カウンタと共用されるため、この図で示されません。

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

2重緩衝化した比較レジスタ(OCR0)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC0)ピンでPWMまたは可変周波数出力を作成するための波形生成器によって使えます。53頁の「比較出力部」をご覧ください。この比較一致発生は比較一致割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF0)も設定(1)します。

定義

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

表43.の定義は本資料を通して広範囲に渡って使われます。

表43. 用語定義

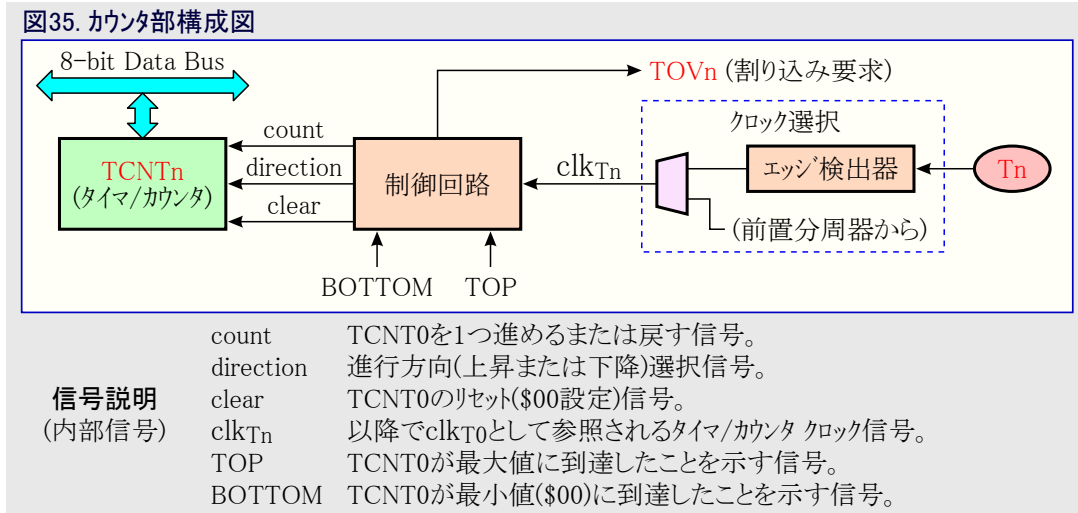
用語	意味
BOTTOM	タイマ/カウンタが\$00に到達した時。
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0値に到達した時。この指定(TOP)値は動作種別に依存します。

タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ0制御レジスタ(TCCR0)**に配置された**クロック選択(CS02~0)**ビットによって制御されるクロック選択論理回路で選択されます。クロック元と前置分周器の詳細については62頁の「**タイマ/カウンタ0とタイマ/カウンタ1の前置分周器**」をご覧ください。

計数器部

8ビット タイマ/カウンタの主な部分は設定可能な双方向カウンタ部です。図35.は、このカウンタとその周辺環境の構成図を示します。



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

計数順序(方法)は**タイマ/カウンタ制御レジスタ(TCCR0)**に配置された**波形生成種別(WGM01,0)**ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOC0比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に関しては55頁の「**動作種別**」をご覧ください。

タイマ/カウンタ溢れ(TOV0)フラグはWGM01,0ビットによって選択された動作種別に従って設定(1)されます。TOV0はCPU割り込み発生に使えます。

比較出力部

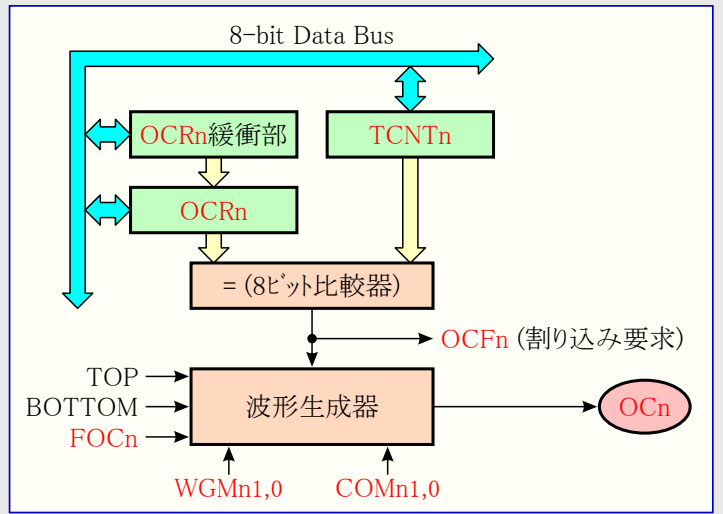
この8ビット比較器はTCNT0と比較レジスタ(OCR0)を継続的に比較します。TCNT0とOCR0が等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で**比較割り込み要求フラグ(OCF0)**を設定(1)します。許可(I=1, OCIE0=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF0は割り込みが実行されると自動的に解除(0)されます。代わりにOCF0はこのビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM01,0)ビット**と**比較出力選択(COM01,0)ビット**によって設定された動作種別に従った出力を生成するのにこの一致信号を使います。MAXとBOTTOM信号は動作種別(55頁の「動作種別」参照)のいくつかで両端値の特別な場合を扱うため、波形生成器によって使われます。

図36. は比較出力部の構成図を示します。

OCR0は**パルス幅変調(PWM)**のどれかを使う時に2重緩衝化されます。標準動作と**比較一致タイマ/カウンタ解除(CTC)動作**については2重緩衝動作が禁止されます。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR0レジスタ更新を同期化します。この同期化は奇数長、非対称PWMパルスの発生を防ぎ、それによって不具合なしの出力を作成します。

OCR0のアクセスは複雑なように思えますが決してそんなことはありません。2重緩衝動作が許可されるとCPUはOCR0緩衝部をアクセスし、禁止されるとOCR0レジスタを直接アクセスします。

図36. 比較出力部構成図



強制比較出力

非PWM波形生成動作での比較器の一致出力は、**強制変更(FOC0)ビット**に1を書くことによって強制(変更)できます。比較一致の強制は**比較割り込み要求フラグ(OCF0)**の設定(1)やタイマ/カウンタの再設定/解除を行いませんが、OC0ピンは実際の比較一致が起きた場合と同様に更新されます(**COM01,0ビット**設定がOC0ピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

TCNT0書き込みによる比較一致妨害

TCNT0への全てのCPU書き込みは、例えばタイマ/カウンタが停止されていても、次のタイマ/カウンタクロック周期で起こるどんな比較一致をも妨げます。この特質はタイマ/カウンタクロックが許可されている時に割り込みを起動することなく、TCNT0と同じ値に初期化されることをOCR0に許します。

比較一致部の使用

どの動作種別でのTCNT0書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、比較出力部を使う場合、TCNT0を変更する時に危険を伴います。TCNT0に書かれた値がOCR0値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。同様にタイマ/カウンタが下降計数のとき、BOTTOMに等しいTCNT0値を書いてはいけません。

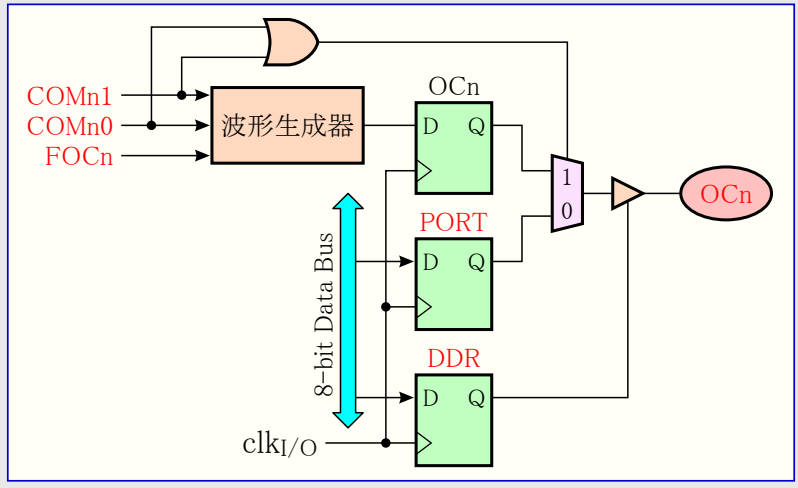
OC0の初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC0値を設定する一番簡単な方法は標準動作で**強制変更(FOC0)ストローブビット**を使うことです。波形生成動作種別間を変更する時であっても、OC0(内部)レジスタはその値を保ちます。

比較出力選択(COM01,0)ビットが比較値(OCR0)と共に2重緩衝されないことに気付いてください。COM01,0ビットの変更は直ちに有効となります。

比較一致出力部

比較出力選択(COM01,0)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力(OC0)状態の定義にCOM01,0ビットを使います。またCOM01,0ビットはOC0ピン出力元を制御します。図37.はCOM01,0ビット設定によって影響される論理回路の簡略化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM01,0ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。OC0の状態を参照するとき、その参照はOC0ピンでなく内部OC0レジスタに対してです。システムリセットが起ると、OC0レジスタは'0'に設定されます。

図37. 比較一致出力回路図



COM01,0ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力(OC0)によって無効にされます。けれどもOC0ピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。OC0ピンに対するポート方向レジスタのビット(DDR_OC0)はOC0値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は波形生成種別と無関係です。

比較出力ピン論理回路の設計は出力が許可される前のOC0状態の初期化を許します。いくつかのCOM01,0ビット設定が或る種の動作種別に対して予約されることに注意してください。59頁の「8ビットタイマ/カウンタ0用レジスタ」をご覧ください。

比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作でCOM01,0ビットを違うふうに使います。全ての動作種別に対してCOM01,0=00設定は次の比較一致で実行すべきOC0レジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については59頁の表45を参照してください。高速PWM動作については59頁の表46、位相基準PWMについては59頁の表47を参照してください。

COM01,0ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(FOC0)スローブビットを使うことによって直ちに効果を得ることを強制できます。

動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は**波形生成種別(WGM01,0)ビット**と**比較出力選択(COM01,0)ビット**の組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM01,0ビットは生成されるPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対するCOM01,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します(54頁の「**比較一致出力部**」をご覧ください)。

タイミング情報の詳細については58頁の「**タイマ/カウンタのタイミング**」を参照してください(訳注:本文中本位置の図番号省略)。

標準動作

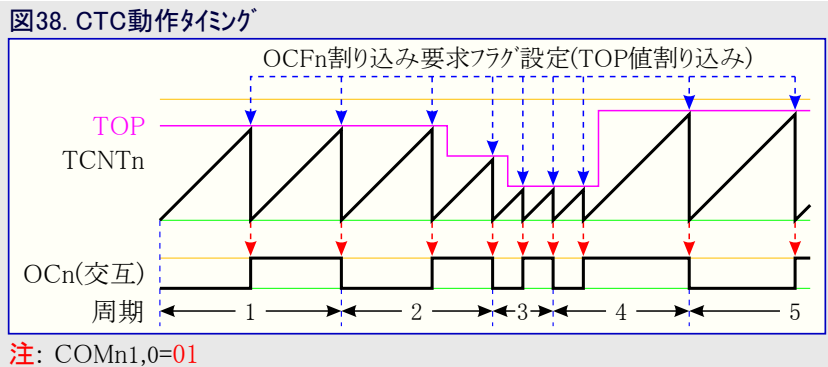
最も単純な動作種別が標準動作(WGM01,0=00)です。本動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作での**タイマ/カウンタ溢れ(TOV0)フラグ**はTCNT0が\$00になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV0フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのようになります。けれどもTOV0フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解能はソフトウェアによって増やせます。標準動作での考慮に特別な場合はなく、新しいカウンタ値は何時でも書けます。

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

比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM01,0=10)ではOCR0がカウンタ分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT0)値がOCR0と一致すると、カウンタは\$00に解除されます。OCR0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38.で示されます。カウンタ(TCNT0)値はTCNT0とOCR0間で比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT0)は解除(\$00)されます。



OCF0フラグを使うことにより、タイマ/カウンタ値がTOP値に到達する時毎に割り込みを生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。けれども前置分周なしまたは低い前置分周値でカウンタが走行している時にBOTTOMと近い値にTOPを変更することは、CTC動作が2重緩衝機能を持たないために注意して行わなければなりません。OCR0に書かれた新しい値がTCNT0の現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こるのに先立って最大値(\$FF)へ、そして次に\$00から始める計数をしなければならないでしょう。

CTC動作で波形出力を生成するため、OC0出力は**比較出力選択(COM01,0)ビット**を交互動作(=01)に設定することによって各比較一致での論理レベル交互切り替えに設定できます。OC0値はそのピンに対するデータ方向が出力(DDR_OC0=1)に設定されない限りポートピンで見えないでしょう。生成された波形はOCR0が0(\$00)に設定される時に $f_{OC0} = f_{clk_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

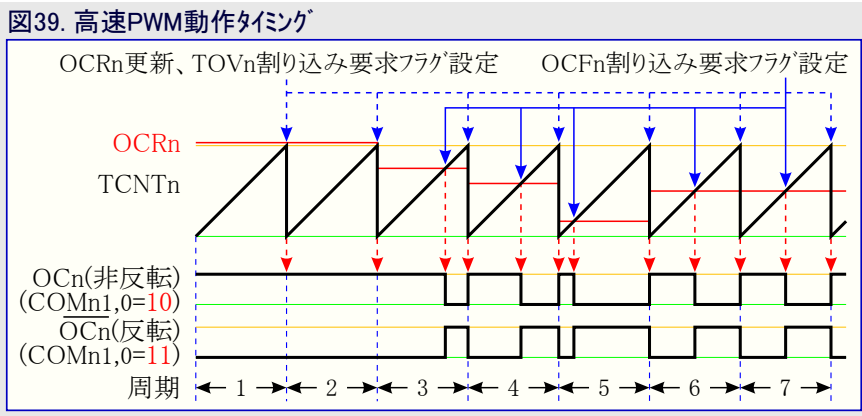
変数Nは前置分周数(1,8,64,256,1024)を表します。

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

高速PWM動作

高速パルス幅変調(PWM)動作(WGM01,0=11)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからMAXまで計数し、その後BOTTOMから再び始めます。非反転比較出力動作(COM01,0=10)での比較出力(OC0)はTCNT0とOCR0間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM01,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がMAX値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は図39.で示されます。TCNT0値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。赤細線はOCR0値を示し、TCNT0値との交点(接点)がTCNT0とOCR0間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF0)はOCR0=MAXを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。



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

高速PWM動作での比較部はOC0ピンでのPWM波形の生成を許します。COM01,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM01,0を'11'に設定することで生成できます(59頁の表46.をご覧ください)。実際のOC0値はポートピンに対するデータ方向(DDR_OC0)が出力として設定される場合だけ見えるでしょう。PWM波形はTCNT0とOCR0間の比較一致でOC0(内部)レジスタを設定(1)(または周期(0))と、カウンタが周期(\$00、MAXからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0レジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

$$f_{OCnPWM} = \frac{f_{clk_I/O}}{N \times 256}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

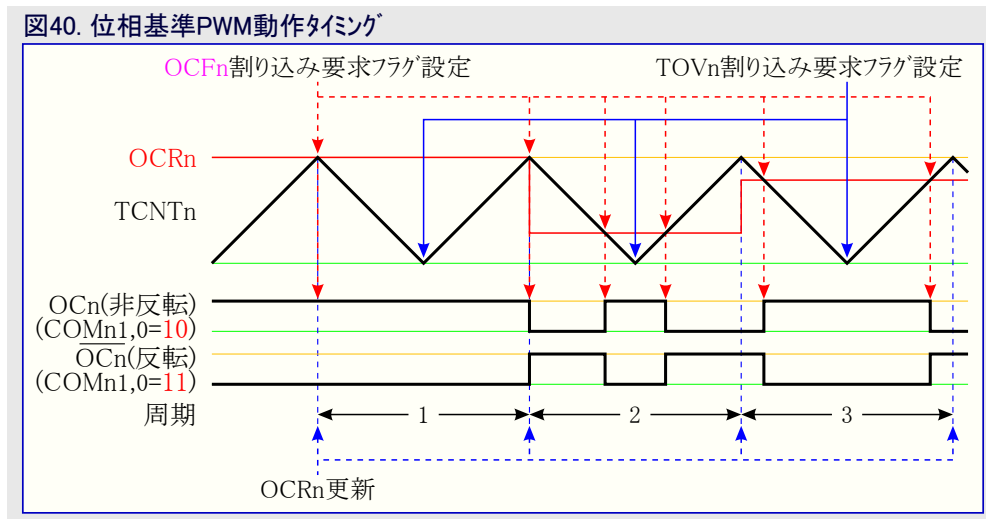
OCR0の両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合に当たります。OCR0がBOTTOM(\$00)と等しく設定されると、出力はMAX+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。OCR0がMAXに等しく設定されると、(COM01,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

(訳注:以下は本動作種別がTOP=MAX(\$FF)で固定のため基本的には不適切です。) 高速PWM動作で(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC0設定(COM01,0=01)によって達成できます。生成された波形はOCR0が0(\$00)に設定される時に $f_{OC0} = f_{clk_I/O} / 2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いてCTC動作でのOC0交互出力(COM01,0=01)と同じです。

位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM01,0=01)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とします。カウンタはBOTTOMからMAXへそして次にMAXからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM01,0=10)での比較出力(OC0)は上昇計数中のTCNT0とOCR0の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM01,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準PWM動作のPWM分解能は8ビットに固定されます。位相基準PWM動作でのカウンタはカウンタ値がMAXと一致するまで増加されます。カウンタはMAXに到達すると、計数方向を変えます。このTCNT0値は1タイマ/カウンタクロック周期間MAXと等しくなります。位相基準PWM動作のタイミング図は図40.で示されます。TCNT0値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。赤細線はOCR0値を示し、TCNT0値との交点(接点)がTCNT0とOCR0間の比較一致を示します(訳注:図補正に伴い本行若干変更)。



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

位相基準PWM動作での比較部はOC0ピンでのPWM波形の生成を許します。COM01,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM01,0ビットを'11'に設定することで生成できます(59頁の表47.をご覧ください)。実際のOC0値はそのポートピンに対するデータ方向(DDR_OC0)が出力として設定される場合だけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT0とOCR0間の比較一致でOC0(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT0とOCR0間の比較一致でOC0レジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

$$f_{OCnPCPWM} = \frac{f_{clkI/O}}{N \times 510}$$

変数Nは前置分周数(1,8,64,256,1024)を表します。

OCR0の両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合にあたります。非反転PWM動作ではOCR0がBOTTOM(\$00)に等しく設定されると出力は定常的なLow、MAXに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。

図40.の第2周期のその出発点に於いて、例えば比較一致がないとしても、OCnにはHighからLowへの遷移があります。この遷移点はBOTTOMを挟む対称を保証するためです。比較一致なしに遷移を生ずるのは2つの場合です。

- 図40.のようにOCR0はMAXからその値を変更します。OCR0値がMAXのとき、OCnピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)MAX(位置)でのOCn値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0値よりも高い値から数え始め、そしてその理屈のために比較一致、それ故上昇途中で起こされるであろうOCnの変更を逃します。(訳補: 従って上記同様、MAX位置で(直前がHならば)H→L遷移が生じます。)

タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従って以下の図でタイマ/カウンタクロック(clk_{T0})がクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時の情報を含みます。図41は基本的なタイマ/カウンタ動作についてのタイミング データを含みます。この図は位相基準 PWM動作以外の全ての動作種別でのMAX値近辺の計数の流れを示します。

図41. 前置分周なし(1/1)のタイマ/カウンタ タイミング

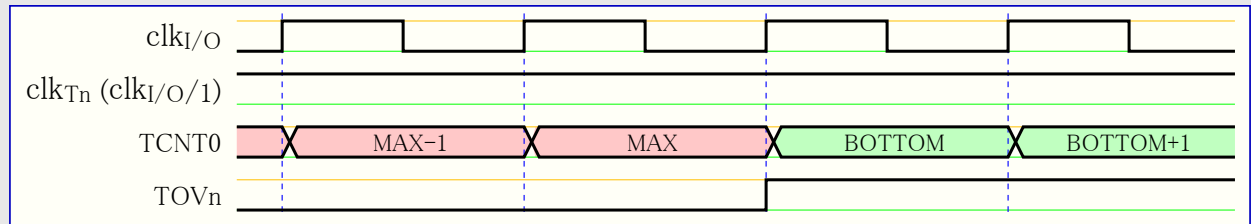


図42は同じタイミング データを示しますが、前置分周器が許可されています。

図42. 前置分周器($f_{\text{clk}_{I/O}/8}$)のタイマ/カウンタ タイミング

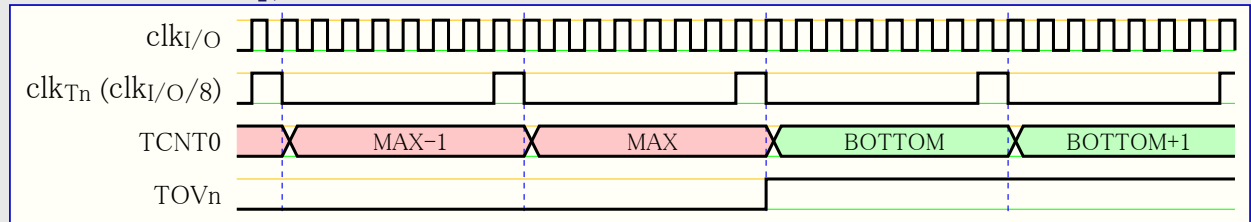


図43はCTC動作を除く全ての動作種別でのOCF0の設定を示します。

図43. 前置分周器($f_{\text{clk}_{I/O}/8}$)のタイマ/カウンタ、OCF0設定 タイミング

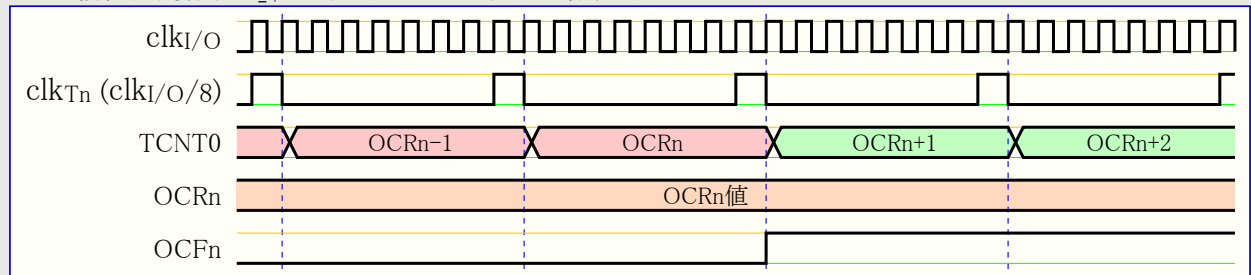
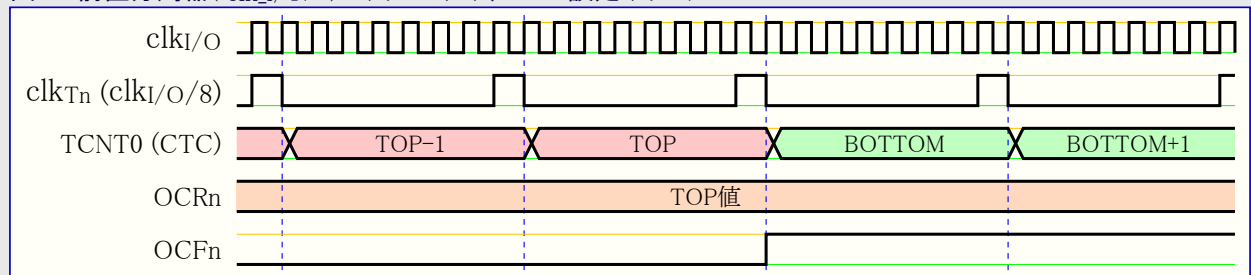


図44はCTC動作でのTCNT0の解除とOCF0の設定を示します。

図44. 前置分周器($f_{\text{clk}_{I/O}/8}$)のタイマ/カウンタ、OCF0設定 タイミング



8ビット タイマ/カウンタ0用レジスタ

■ タイマ/カウンタ0制御レジスタ (Timer/Counter0 Control Register) TCCR0

ビット	7	6	5	4	3	2	1	0	
\$33 (\$53)	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7 – FOC0 : OC0強制変更 (Force Output Compare 0)

FOC0ビットは**WGM01,0ビット**が非PWM動作を指示する時だけ有効です。けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR0が書かれる場合、このビットは**0**に設定されなければなりません。FOC0ビットに論理**1**を書くと、波形生成部で直ちに比較一致が強制されます。OC0出力は**COM01,0ビット**設定に従って変更されます。FOC0ビットがスロープとして実行されることに注意してください。従って強制した比較の効果を決めるのはCOM01,0ビットに存在する値です。

FOC0スロープは何れの割り込みの生成もTOPとしてOCR0を使う**比較一致タイマ解除(CTC)動作**でのタイマ/カウンタの解除(\$00)も行いません。

FOC0ビットは常に**0**として読まれます。

■ ビット3,6 – WGM01,0 : 波形生成種別 (Waveform Generation Mode bit 1 and 0)

これらのビットはカウンタの計数順序(方向)、最大(TOP)カウンタ値の供給元、使われるべき波形生成のどの形式かを制御します。タイマ/カウンタ部により支援される動作種別は標準動作、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。表44.と55頁の「**動作種別**」をご覧ください。

表44. 波形生成種別選択

番号	WGM01 (CTC0)	WGM00 (PWM0)	タイマ/カウンタ動作種別	TOP値	OCR0更新時	TOV0設定時
0	0	0	標準動作	\$FF	即時	MAX
1	0	1	位相基準PWM動作	\$FF	TOP	BOTTOM
2	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR0	即時	MAX
3	1	1	高速PWM動作	\$FF	BOTTOM	MAX

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

■ ビット5,4 – COM01,0 : 比較0出力選択 (Compare Match 0 Output Mode bit 1 and 0)

これらのビットはOC0比較出力ピンの動作を制御します。COM01,0ビットの1つまたは両方が**1**を書かれると、OC0出力はそのI/Oピンの通常ポート機能を無効にし、そのI/Oピンに接続されます。けれども出力駆動部を許可するため、OC0ピンに対応するポート方向レジスタ(DDR)のビットが設定(**1**)されなければならないことに注意してください。

OC0がピンに接続されるとき、COM01,0ビットの機能は**WGM01,0ビット**設定に依存します。

表45.はWGM01,0ビットが**標準動作**または**CTC動作**(つまりPWM以外)に設定される時のCOM01,0ビット機能を示します。

表46.はWGM01,0ビットが**高速PWM動作**に設定される時のCOM01,0ビットの機能を示します。

表47.はWGM01,0ビットが**位相基準PWM動作**に設定される時のCOM01,0ビットの機能を示します。

表45. 非PWM動作での比較出力選択

COM01	COM00	意味
0	0	標準ポート動作 (OC0切断)
0	1	比較一致でOC0ピントグル(交互)出力
1	0	比較一致でOC0ピン Lowレベル出力
1	1	比較一致でOC0ピン Highレベル出力

表46. 高速PWM動作での比較出力選択

COM01	COM00	意味
0	0	標準ポート動作 (OC0切断)
0	1	(予約)
1	0	上昇計数時の比較一致でLow、下降計数時の比較一致でHighをOC0ピンへ出力
1	1	上昇計数時の比較一致でHigh、下降計数時の比較一致でLowをOC0ピンへ出力

注: COM01が設定(**1**)され、OCR0がTOPと等しいときに特別な状態が起きます。この状態での比較一致は無視されますが、TOPでの設定(**1**)や解除(**0**)は行われます。より多くの詳細については57頁の「**位相基準PWM動作**」をご覧ください。

表46. 高速PWM動作での比較出力選択

COM01	COM00	意味
0	0	標準ポート動作 (OC0切断)
0	1	(予約)
1	0	比較一致でLow、BOTTOMでHighをOC0ピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC0ピンへ出力 (反転動作)

注: COM01が設定(**1**)され、OCR0がTOPと等しいときに特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMでの設定(**1**)や解除(**0**)は行われます。より多くの詳細については56頁の「**高速PWM動作**」をご覧ください。

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

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

表48. タイマ/カウンタ0入力クロック選択

CS02	CS01	CS00	意味
0	0	0	停止 (タイマ/カウンタ0動作停止)
0	0	1	clk _{I/O} (前置分周なし)
0	1	0	clk _{I/O} /8 (8分周)
0	1	1	clk _{I/O} /64 (64分周)
1	0	0	clk _{I/O} /256 (256分周)
1	0	1	clk _{I/O} /1024 (1024分周)
1	1	0	T0ピンの下降端 (外部クロック)
1	1	1	T0ピンの上昇端 (外部クロック)

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

■ タイマ/カウンタ0 (Timer/Counter0) TCNT0

ビット	7	6	5	4	3	2	1	0	
\$32 (\$52)	(MSB)							(LSB)	TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

このタイマ/カウンタレジスタは読み書き両方の操作について、タイマ/カウンタ部の8ビットカウンタに直接アクセスします。TCNT0への書き込みは次のタイマ/カウンタクロックでの比較一致を妨害(除去)します。カウンタが走行中にカウンタ(TCNT0)を変更することはTCNT0とOCR0間の比較一致消失の危険を誘発します。

■ タイマ/カウンタ0 比較レジスタ (Timer/Counter0 Output Compare Register) OCR0

ビット	7	6	5	4	3	2	1	0	
\$31 (\$51)	(MSB)							(LSB)	OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

この比較レジスタは継続的にカウンタ(TCNT0)値と比較される8ビットの値を含みます。一致は比較一致割り込みやOC0ピンでの波形出力を生成するのに使えます。

■ タイマ/カウンタ割り込み許可レジスタ (Timer/Counter Interrupt Mask Register) TIMSK

ビット	7	6	5	4	3	2	1	0	
\$39 (\$59)	TOIE1	OCIE1A	OCIE1B	–	TICIE1	–	TOIE0	OCIE0	TIMSK
Read/Write	R/W	R/W	R/W	R	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット1 – TOIE0 : タイマ/カウンタ0溢れ割り込み許可 (Timer/Counter0 Overflow Interrupt Enable)

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

■ ビット0 – OCIE0 : タイマ/カウンタ0比較割り込み許可 (Timer/Counter0 Output Compare Match Interrupt Enable)

OCIE0ビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較一致割り込みが許可されます。タイマ/カウンタ0で比較一致が起こる(換言すると、タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)で比較0割り込み要求フラグ(OCF0)が設定(1)されると、対応する割り込みが実行されます。

■ タイマ/カウンタ割り込み要求フラグレジスタ (Timer/Counter Interrupt Flag Register) TIFR

ビット	7	6	5	4	3	2	1	0	
\$38 (\$58)	TOV1	OCF1A	OCF1B	–	ICF1	–	TOV0	OCF0	TIFR
Read/Write	R/W	R/W	R/W	R	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット1 – TOV0 : タイマ/カウンタ0溢れ割り込み要求フラグ (Timer/Counter0 Overflow Flag)

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

■ ビット0 – OCF0 : タイマ/カウンタ0比較割り込み要求フラグ (Timer/Counter0, Output Compare Match Flag)

OCF0ビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行する時にOCF0はハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0は解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ割り込み許可レジスタ(TIMSK)のタイマ/カウンタ0比較一致割り込み許可(OCIE0)ビット、OCF0が設定(1)されると、タイマ/カウンタ0比較一致割り込みが実行されます。

タイマ/カウンタ0とタイマ/カウンタ1の前置分周器

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

内部クロック元

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

前置分周器リセット

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

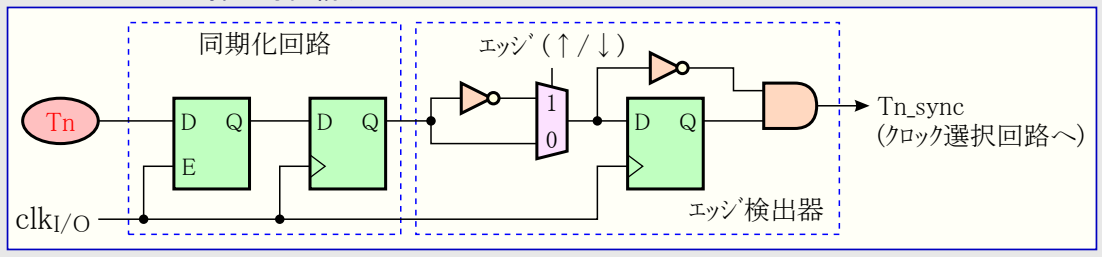
プログラム実行にタイマ/カウンタを同期することに対して前置分周器リセットを使うことが可能です。しかし、同じ前置分周器を共用する他のタイマ/カウンタも前置分周を使う場合、注意が必要とされなければなりません。前置分周器リセットはそれが接続される全タイマ/カウンタに関する前置分周器周期に影響を及ぼします。

外部クロック元

T0/T1ピンに印加された外部クロック元はタイマ/カウンタ クロック(f_{clk_T0}/f_{clk_T1})として使えます。このT0/T1ピンはピン同期化論理回路によって全てのシステム クロック周期に一度採取されます。この同期化(採取)された信号はその後にエッジ検出器を通して通過されます。図45. はT0/T1同期化とエッジ検出器論理回路の機能等価構成図を示します。レジスタは内部システム クロック($f_{clk_I/O}$)の上昇端でクロック駆動されます。ラッチは内部システム クロックのHigh区間で通過(Low区間で保持)です。

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

図45. T0/T1ピンの採取等価構成図



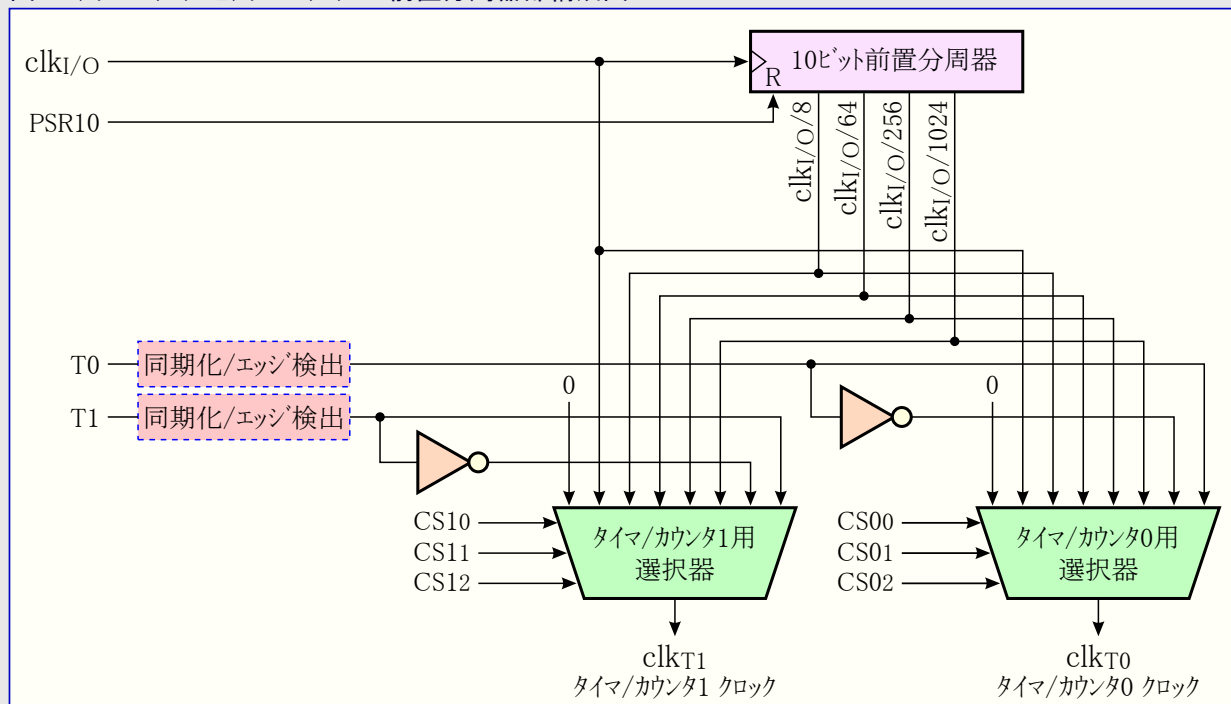
同期化とエッジ検出器論理回路はT0/T1ピンへ印加されたエッジから計数器が更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

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

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

外部クロック元は前置分周できません。

図46. タイマ/カウンタ0とタイマ/カウンタ1の前置分周器部構成図



注: 入力ピン(T0/T1)の同期化/エッジ検出論理回路は図45.で示されます。

■ 特殊I/O機能レジスタ (Special Function I/O Register) SFIOR

ビット	7	6	5	4	3	2	1	0	
\$30 (\$50)	—	XMBK	XMM2	XMM1	XMM0	PUD	—	PSR10	SFIOR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット0 – PSR10 : タイマ/カウンタ1,0 前置分周器リセット (Prescaler Reset Timer/Counter1 and 0)

このビットが1を書かれると、タイマ/カウンタ1とタイマ/カウンタ0の前置分周器がリセットされます。このビットはその操作実行後、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。タイマ/カウンタ1とタイマ/カウンタ0は同じ前置分周器を共用し、この前置分周器のリセットが両方のタイマ/カウンタに影響を及ぼすことに注意してください。このビットは常に0として読めます。

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)、波形生成、信号タイミング計測を許します。主な特徴は次に示されます。

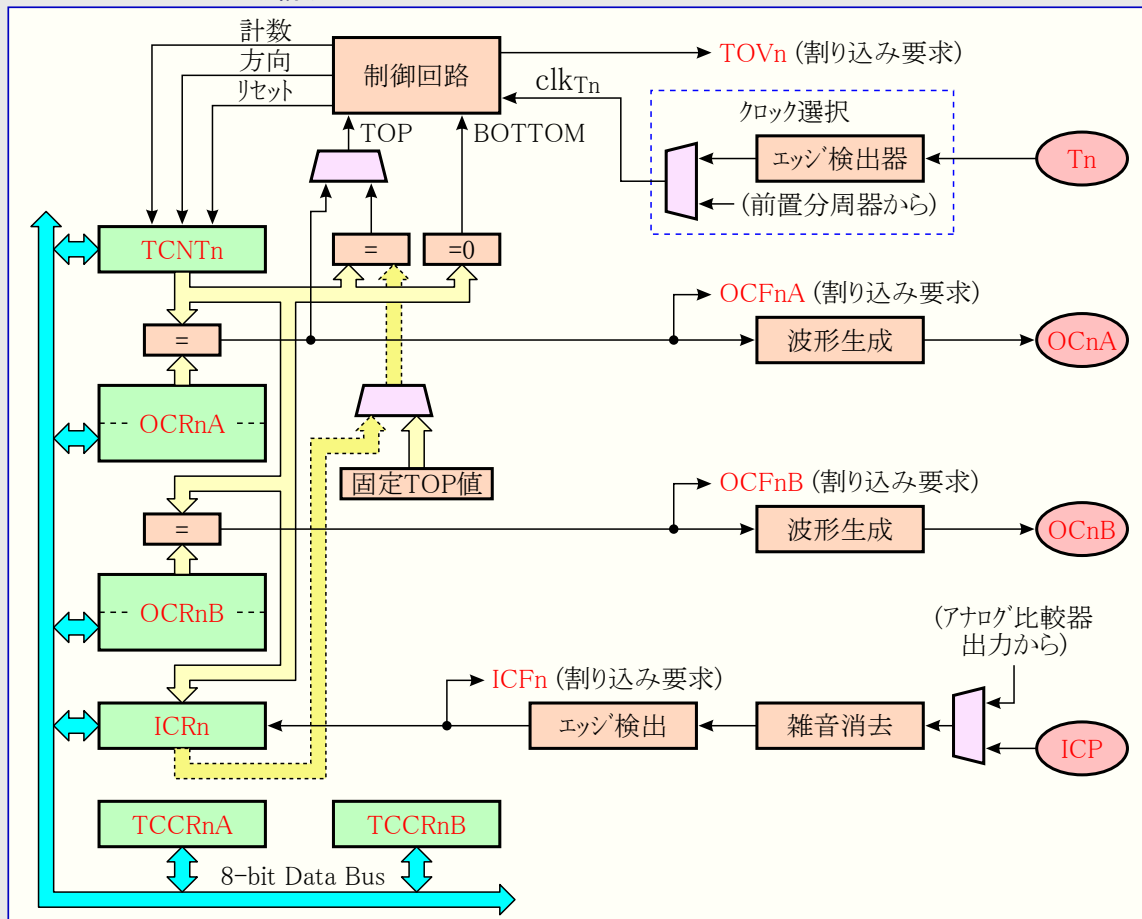
- ・ 真の16ビット設計 (換言すれば16ビットPWMの許容)
- ・ 2つの独立した比較出力部
- ・ 2重緩衝の比較レジスタ
- ・ 1つの捕獲入力部
- ・ 捕獲入力雑音消去器
- ・ 比較一致でのタイマ/カウンタ解除 (自動再設定)
- ・ 不具合なしで正しい位相のパルス幅変調器 (PWM)
- ・ 可変PWM周期
- ・ 周波数発生器
- ・ 外部事象計数器
- ・ 4つの独立した割り込み (TOV1,OCF1A,OCF1B,ICF1)

概要

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

この16ビット タイマ/カウンタの簡単化した構成図は図47.で示されます。実際のI/Oピンの配置については1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78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。

図47. 16ビット タイマ/カウンタ構成図



注: タイマ/カウンタ1のピンの配置と記述については1頁の「ピン配置」、41頁の表29、44頁の表35を参照してください。

関係レジスタ

タイマ/カウンタ(TCNT1)、比較レジスタ(OCR1A,OCR1B)、捕獲レジスタ(ICR1)は全て16ビットレジスタです。16ビットレジスタをアクセスするとき、特別な手順に従わなければなりません。これらの手順は66頁の「16ビットレジスタのアクセス」項で記述されます。タイマ/カウンタ1制御レジスタ(TCCR1A,TCCR1B)は8ビットレジスタで、CPUアクセス制限はありません。割り込み要求信号はタイマ/カウンタ割り込み要求レジスタ(TIFR)で全て見えます。全ての割り込みはタイマ/カウンタ割り込み許可レジスタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKはこれらのレジスタが他のタイマ/カウンタ部によって共用されるため、この図で示されません。

このタイマ/カウンタは内部的、前置分周器経由、またはT1ピンの外部クロック元によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加(または減少)するのに使うクロック元とエッジを制御します。クロック元が選択されないとき、タイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkT1)として参照されます。

2重緩衝化した比較レジスタ(OCR1A,OCR1B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC1A,OC1B)ピンでPWMまたは可変周波数出力を生成するための波形生成器によって使えます。70頁の「比較出力部」をご覧ください。この比較一致発生は比較出力割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF1A,OCF1B)も設定(1)します。

捕獲レジスタ(ICR1)は捕獲起動(ICP)ピン、またはアナログ比較器出力(106頁の「アナログ比較器」参照)のどちらかの外部(エッジで起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波部(雑音消去器)を含みます。

TOP値または最大タイマ/カウンタ値は、いくつかの動作種別で、OCR1A、ICR1、または一群の固定値のどれかによって定義できます。PWM動作でTOP値としてOCR1Aを使うと、OCR1AはPWM出力生成用に使いません。けれどもこの場合、TOP値は動作中に変更されるのをTOP値に許す2重緩衝化します。固定的なTOP値が必要とされる場合、ICR1が代わりに使い、PWM出力として使われるべきOCR1Aを開放します。

定義

次の定義は本資料を通して広範囲に使われます。

表49. 用語定義

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

互換性

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

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

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

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

次の制御ビットが16ビットタイマ/カウンタ制御レジスタ内に追加されます。

- ・FOC1AとFOC1BがTCCR1Aに追加。
- ・WGM13がTCCR1Bに追加。

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

16ビットレジスタのアクセス

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

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

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

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

アセンブリ言語プログラム例

```

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

```

C言語プログラム例

```

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

```

注: 4頁の「コード例について」をご覧ください。

アセンブリ言語コード例は R17:R16 レジスタ対に TCNT1 値を戻します。

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

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「コード例について」をご覧ください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「コード例について」をご覧ください。

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

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。

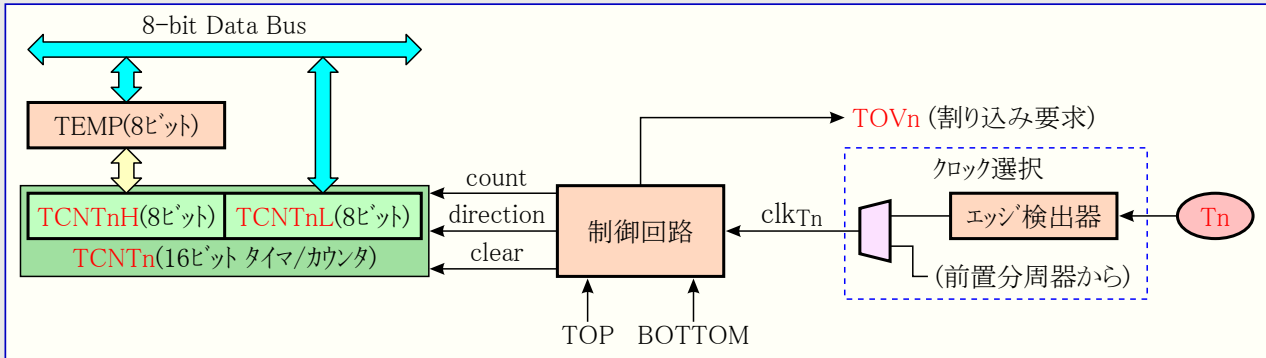
タイマ/カウンタのクロック

このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元は**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**クロック選択(CS12~0)**ビットによって制御されるクロック選択論理回路で選択されます。クロック元と前置分周器の詳細については62頁の「**タイマ/カウンタ0とタイマ/カウンタ1の前置分周器**」をご覧ください。

計数器部

16ビット タイマ/カウンタの主な部分は設定可能な16ビット双方向カウンタ部です。図48はこのカウンタとその周辺の構成図を示します。

図48. カウンタ部構成図



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

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

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

計数順序(方法)は**タイマ/カウンタ制御レジスタA(TCCR1A)**と**タイマ/カウンタ制御レジスタB(TCCR1B)**に配置された**波形生成種別(WGM13~0)**ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては73頁の「**動作種別**」をご覧ください。

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

捕獲入力部は図49.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。

[illegible]

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66頁の「16ビットレジスタのアクセス」を参照してください。

捕獲入力はICP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。

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

捕獲入力の使用

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

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

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

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

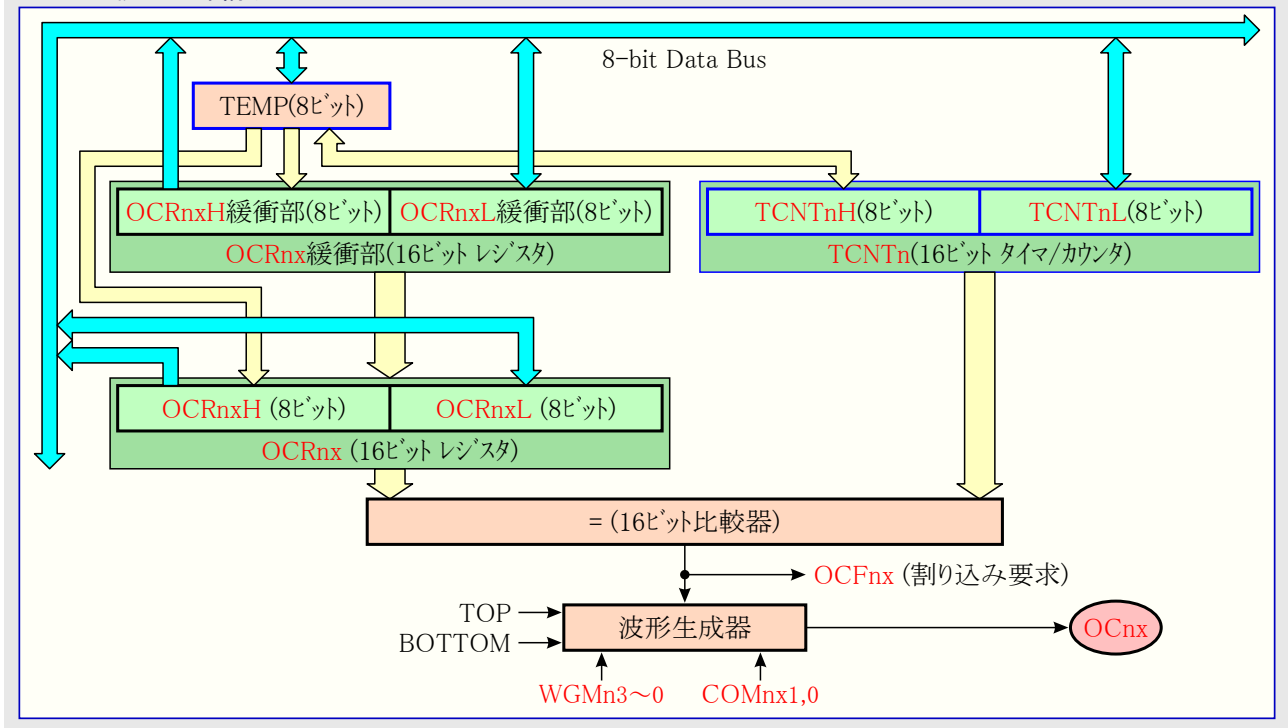
比較出力部

この16ビット比較器は**TCNT1**と**比較レジスタ(OCR1x)**を継続的に比較します。TCNT1とOCR1xが等しければ、比較器は一致を指示します。この一致は**次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF1x)**を設定(1)します。許可(I=1, **OCIE1x=1**)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF1xは割り込みが実行されると自動的に解除(0)されます。代わりにOCF1xはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**によって設定された動作種別に従った出力を生成するのにこの一致信号を使います。TOPとBOTTOM信号は動作種別(73頁の「動作種別」参照)のいくつかで両端値の特別な場合を扱うため、波形生成器によって使われます。

比較A出力部の特殊な特性はタイマ/カウンタのTOP値(換言するとカウンタの分解能)定義を許します。カウンタの分解能に加え、TOP値は波形生成器によって生成された波形の周期時間を定義します。

図50.は比較出力部の構成図を示します。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対してはn=1)、小文字の'x'は比較出力部(AまたはB)を表します。直接的な比較出力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。

図50. 比較出力部構成図



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

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

16ビットレジスタアクセス法のより多くの情報については66頁の「16ビットレジスタのアクセス」を参照してください。

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

強制比較出力

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

TCNT1書き込みによる比較一致妨害

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

比較一致部の使用

どの動作種別でのTCNT1書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、何れかの比較出力部を使う場合、TCNT1を変更する時に危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。可変TOP値のPWM動作でTOPに等しいTCNT1を書いてはいけません。(行った場合)TOPに対する比較一致は無視され、カウンタは\$FFFFへ(計数を)続けます。同様にカウンタが下降計数のとき、BOTTOMに等しいTCNT1値を書いてはいけません。

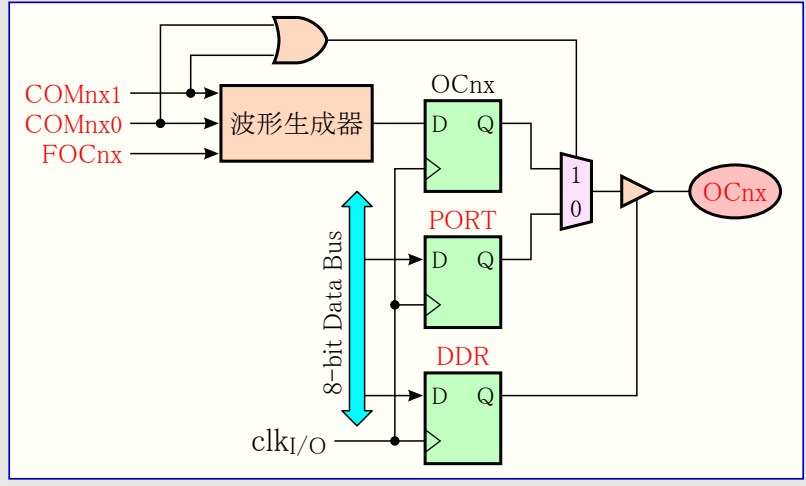
OC1xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)スローブビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

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

比較一致出力部

比較出力選択(COM1x1,0)ビットは2つの機能を持ちます。波形生成器は次の比較一致での比較出力(OC1x)状態の定義にCOM1x1,0ビットを使います。次にCOM1x1,0ビットはOC1xピン出力元を制御します。図51.はCOM1x1,0ビット設定によって影響される論理回路の簡略化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM1x1,0ビットによって影響を及ぼされる標準I/Oポート制御レジスタ(PORTとDDR)の部分だけが示されます。OC1xの状態を参照するとき、その参照はOC1xピンでなく内部OC1xレジスタに対してです。システムリセットが起ると、OC1xレジスタは0にリセットされます。

図51. 比較一致出力回路図



COM1x1,0ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力(OC1x)によって無効にされます。けれどもOC1xピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。OC1xピンに対するポート方向レジスタのビット(DDR_OC1x)はOC1x値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は一般的に波形生成種別と無関係ですが、いくつかの例外があります。詳細については表50、表51、表52を参照してください。

比較出力ピン論理回路の設計は出力が許可される前のOC1x状態の初期化を許します。いくつかのCOM1x1,0ビット設定が或る種の動作種別に対して予約されることに注意してください。78頁の「16ビットタイマ/カウンタ用レジスタ」をご覧ください。

COM1x1,0ビットは捕獲入力部での何の効果もありません。

比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作でCOM1x1,0ビットを違うふうに使います。全動作種別に対してCOM1x1,0=00設定は次の比較一致で実行すべきOC1xレジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については78頁の表50.を参照してください。高速PWM動作については78頁の表51、位相基準PWMと位相/周波数基準PWMについては78頁の表52.を参照してください。

COM1x1,0ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(FOC1x)スローブビットを使うことによって直ちに効果を得ることを強制できます。

動作種別

動作種別、換言するとタイマ/カウンタと比較出力ピンの動作は**波形生成種別(WGM13~0)ビット**と**比較出力選択(COM1x1,0)ビット**の組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM1x1,0ビットは生成されたPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対してのCOM1x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します。[72頁の「比較一致出力部」](#)をご覧ください。

タイミング情報の詳細については[77頁の「タイマ/カウンタのタイミング」](#)を参照してください。

標準動作

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

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

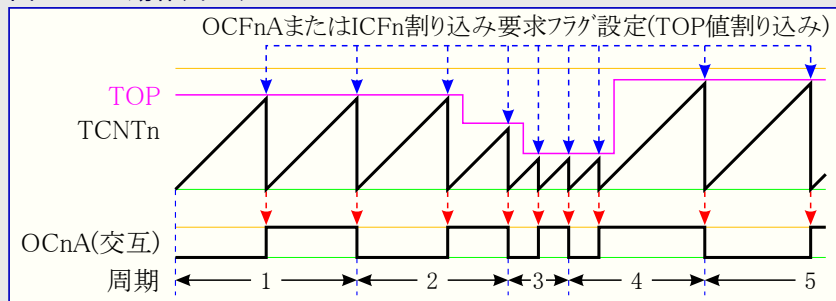
比較出力部は与えられた或る時間に割り込みを生成するのに使えます。標準動作で波形を生成するのに比較出力を使うのは、それが大変多くのCPU時間を占有するため推奨されません。

比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM13~0=0100または1100)ではOCR1AまたはICR1がカウンタの分解能を操作するのに使われます。CTC動作ではカウンタ(TCNT1)値がOCR1A(WGM13~0=4)またはICR1(WGM13~0=12)のどちらかと一致する時にカウンタが\$0000に解除されます。OCR1AまたはICR1はカウンタに対する**TOP値**、従って分解能も定義します。この動作種別はより大きい比較一致出力周波数の制御を許します。それは外部の出来事の計数操作も簡単にします。

CTC動作についてのタイミング図は[図52](#)で示されます。カウンタ(TCNT1)値はOCR1AまたはICR1のどちらかで比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT1)は解除(\$0000)されます。

図52. CTC動作タイミング



注: COMnA1,0=01

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

CTC動作で波形出力を生成するため、OC1A出力は**比較出力選択(COM1A1,0)ビット**を交互動作(=01)に設定することによって各比較一致での論理レベル交互切替に設定できます。OC1A値はそのピンに対するデータ方向が出力(DDR_OC1A=1)に設定されない限り、ポートピンで見えないでしょう。生成された波形はOCR1Aが0(\$0000)に設定される時に $f_{OC1A} = f_{clk_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

変数Nは前置分周数(1,8,64,256,1024)を表します。

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

高速PWM動作

高速パルス幅変調(PWM)動作(WGM13~0=0101,0110,0111,1110,1111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)はTCNT1とOCR1x間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM1x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準や位相/周波数基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

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

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

高速PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0101,0110,0111)、ICR1値(WGM13~0=1110)またはOCR1A値(WGM13~0=1111)のどれかと一致するまで増加されます。そしてカウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$0000)されます。高速PWM動作のタイミング図は図53.

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

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

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

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

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

高速PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます。78頁の表51をご覧ください。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$0000、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でのOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

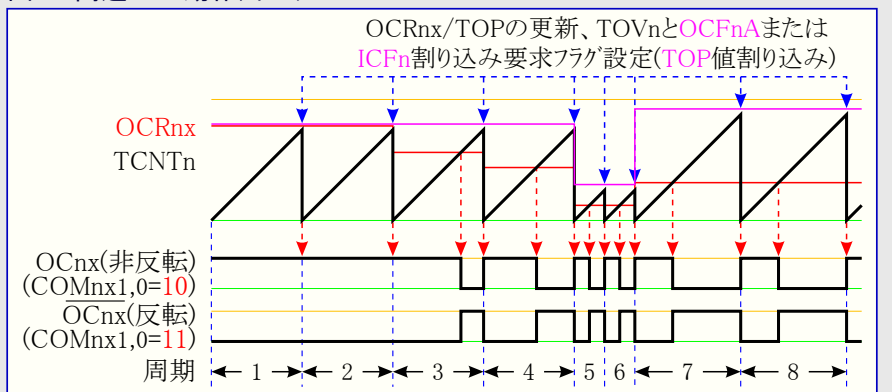
PWM出力周波数は次式によって計算できます。変数Nは前置分周数(1,8,64,256,1024)を表します。

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

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

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

図53. 高速PWM動作タイミング



位相基準PWM動作

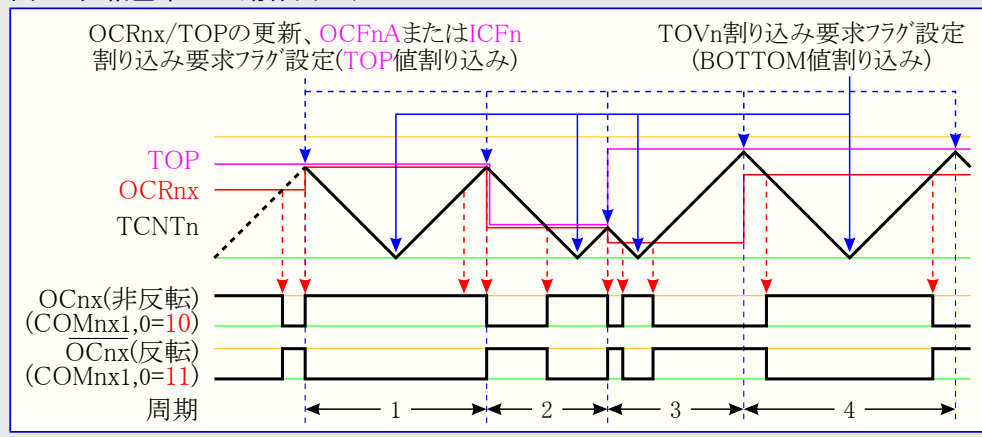
位相基準パルス幅変調(PWM)動作(WGM13~0=0001,0010,0011,1010,1011)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とした位相/周波数基準PWMと似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾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

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

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

位相基準PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0001,0010,0011), ICR1値(WGM13~0=1010)またはOCR1A値(WGM13~0=1011)のどれかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期TOPと等しくなります。位相基準PWM動作のタイミング図は図54.で示されます。本図はOCR1AかICR1がTOPを定義するのに使われる時の位相基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

図54. 位相基準PWM動作タイミング



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

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。固定TOP値を使う場合、どのOCR1xが書かれる時も、未使用ビットが0で隠(に置換)されることに注意してください。図54.で示される第3周期が図解するようにタイマ/カウンタが位相基準PWM動作で走行中にTOPを積極的に変更するのは、非対称出力で終わることが有り得ます。これに対する理由はOCR1xレジスタの更新時に見出せます。OCR1x更新はTOPで起きるので、PWM周期はTOPで始まりそして終わります。これは下降傾斜長が直前のTOP値によって決定され、一方上昇傾斜長は新しいTOP値で決定されることを意味します。これら2つの値(TOP)が違う時にその周期の2つの傾斜長は異なるでしょう。この長さの相違が出力での非対称な結果を生じます。

タイマ/カウンタが走行中にTOP値を変更する場合、位相基準PWM動作の代わりに位相/周波数基準PWM動作を使うことが推奨されます。一定のTOP値を使う時に2つの動作種別間に現実的な違いはありません。

位相基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます(78頁の表52.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

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

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

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

位相/周波数基準PWM動作

位相/周波数基準パルス幅変調(PWM)動作(WGM13~0=1000,1001)は高分解能で正しい位相と周波数のPWM波形生成選択を提供します。位相/周波数基準PWM動作は両傾斜(三角波)動作を基準とした**位相基準PWM**と似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

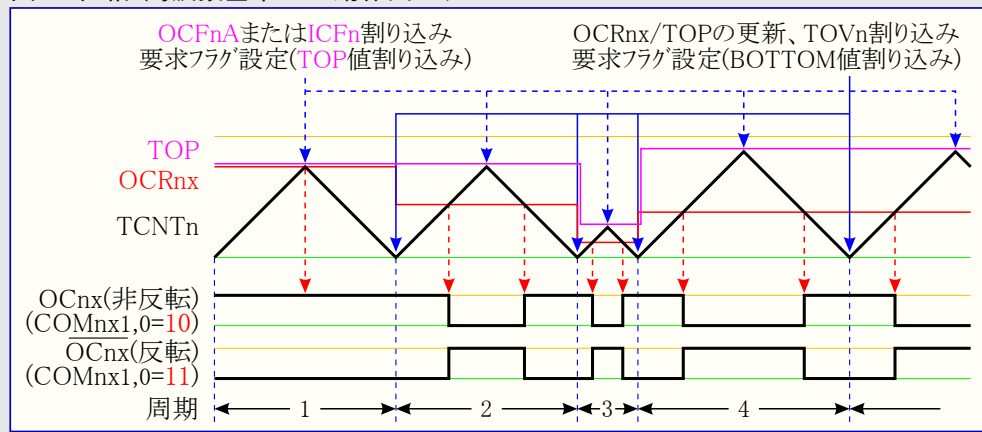
位相基準と位相/周波数基準PWM動作間の主な違いはOCR1xレジスタがOCR1x緩衝部によって更新される時(訳補:TOPとBOTTOM)です(図54と図55参照)。

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

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

位相/周波数基準PWM動作でのタイマ/カウンタはタイマ/カウンタ値がICR1値(WGM13~0=1000)かOCR1A値(WGM13~0=1001)のどちらかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期間、TOPと等しくなります。位相/周波数基準PWM動作のタイミング図は図55で示されます。この図はOCR1AかICR1がTOPを定義するのに使われる時の位相/周波数基準PWM動作を示します。TCNT1値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

図55. 位相/周波数基準PWM動作タイミング



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

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

図55.が示すように、生成された出力は位相基準PWM動作と異なり、全ての周期で対称です。OCR1xレジスタがBOTTOMで更新されるため、上昇と下降の傾斜長は常に等しくなります。これが対称出力パルス、従って正しい周波数を与えます。

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

位相/周波数基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x,0を'11'に設定することで生成できます(78頁の表52.をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。

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

$$f_{OCnxPFCPWM} = \frac{f_{clk_1/O}}{2 \times N \times TOP}$$

OCR1xの両端値は位相/周波数基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。TOP値定義にOCR1Aが使われ(WGM13~0=1001)、COM1A,0=01なら、OC1A出力はデューティ比50%で交互に変化します。

タイマ/カウンタのタイミング

このタイマ/カウンタは同期設計で、従ってタイマ/カウンタクロック(clk_{T1})が下図のクロック許可信号として示されます。この図は割り込みフラグが設定(1)される時、そして OCR1x レジスタが OCR1x 緩衝値で更新される時(2重緩衝を使う動作種別のみ)の情報を含みます。図56は OCF1x の設定についてのタイミング図を示します。

図56. 前置分周なし(1/1)のタイマ/カウンタ、 OCF1x 設定 タイミング

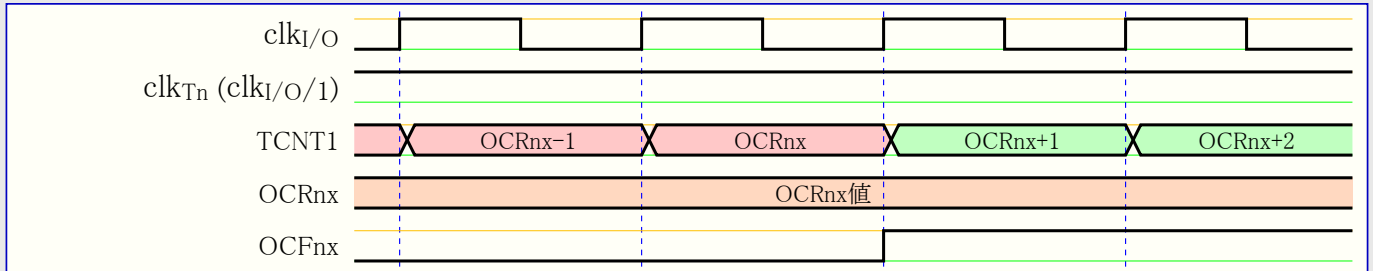


図57. は同じタイミング データを示しますが、前置分周器が許可されています。

図57. 前置分周器($f_{\text{clk}_{I/O}/8}$)のタイマ/カウンタ、 OCF1x 設定 タイミング

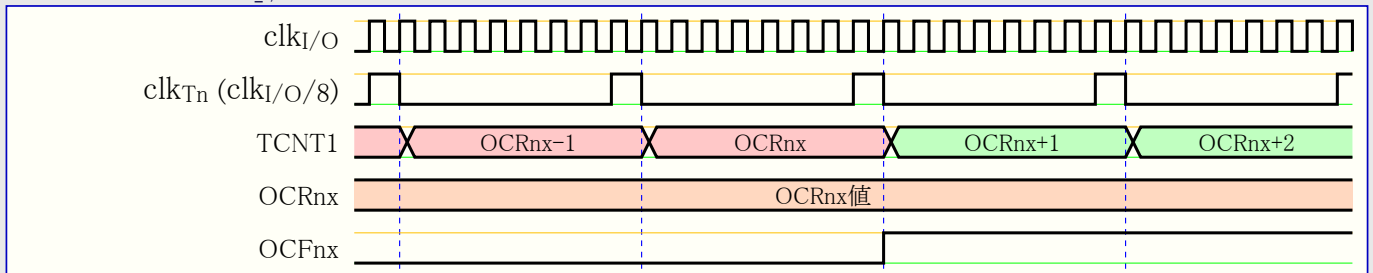


図58. は各動作種別でのTOP近辺の計数手順を示します。位相/周波数基準PWM動作使用時の OCR1x レジスタはBOTTOMで更新されます。タイミング図は同じになりますが、当然TOPはBOTTOMで、TOP-1はBOTTOM+1でなどのように置き換えられます。BOTTOMで TOV1 を設定(1)する動作種別についても、同様な名称変更が適用されます。

図58. 前置分周器なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング

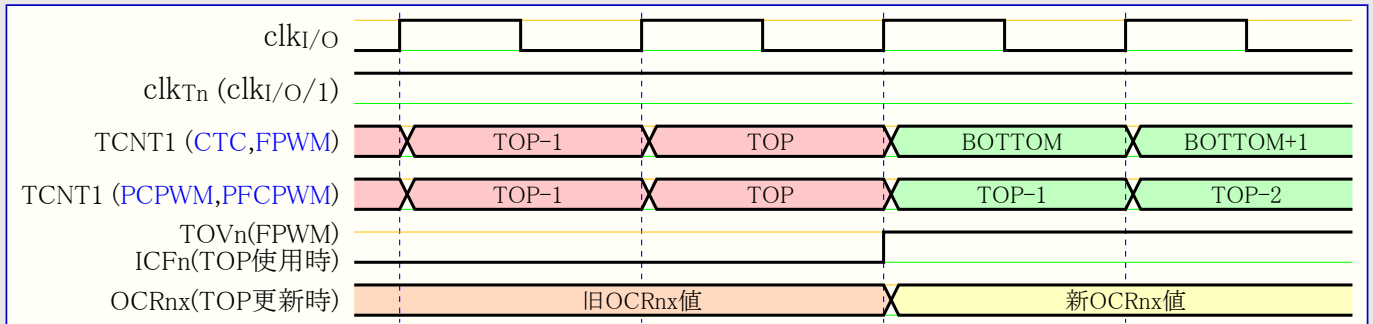
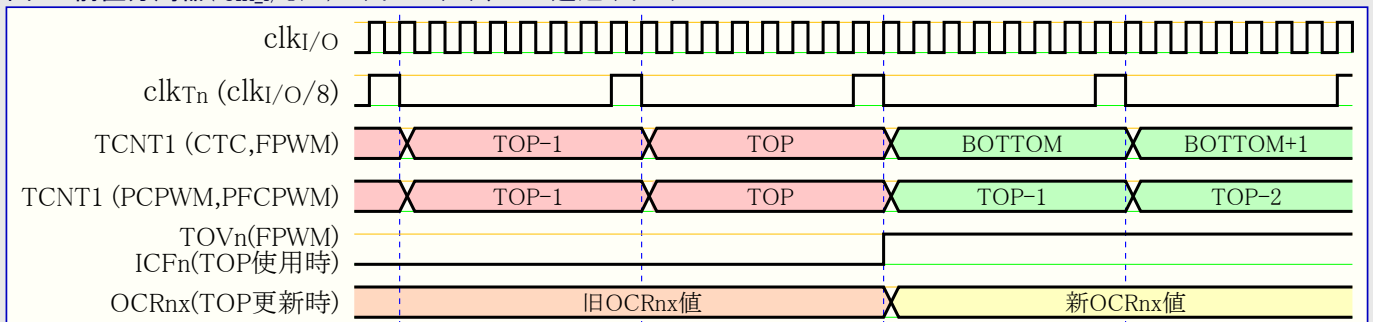


図59. は同じタイミング データを示しますが、前置分周器が許可されています。

図59. 前置分周器($f_{\text{clk}_{I/O}/8}$)のタイマ/カウンタ、TOP近辺 タイミング



16ビット タイマ/カウンタ1用レジスタ

■ タイマ/カウンタ1制御レジスタA (Timer/Counter1 Control Register A) TCCR1A

ビット	7	6	5	4	3	2	1	0	
\$2F (\$4F)	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7,6 – COM1A1,0 : 比較1A出力選択 (Compare Output Mode1A bit 1 and 0)

■ ビット5,4 – COM1B1,0 : 比較1B出力選択 (Compare Output Mode1B bit 1 and 0)

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

OC1AまたはOC1Bがピンに接続されるとき、COM1x1,0ビットの機能はWGM13~0ビット設定に依存します。表50.はWGM13~0ビットが標準動作またはCTC動作(つまり非PWM)に設定される時のCOM1x1,0ビット機能を示します。

表50. 非PWM動作での比較出力選択 (注: xはAまたはB)

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

表51.はWGM13~0ビットが高速PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表51. 高速PWM動作での比較出力選択 (注: xはAまたはB)

COM1x1	COM1x0	意味
0	0	標準ポート動作 (OC1x切断)
0	1	WGM13~0=1111 : 比較一致でOC1Aピン トグル(交互)出力、OC1Bは標準ポート動作(OC1B切断) WGM13~0上記以外 : 標準ポート動作 (OC1x切断)
1	0	比較一致でLow、BOTTOMでHighをOC1xピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC1xピンへ出力 (反転動作)

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。この状態での比較一致は無視されますが、BOTTOMでの設定(1)または解除(0)は実行されます。より多くの詳細については74頁の「高速PWM動作」をご覧ください。

表52.はWGM13~0ビットが位相基準または位相/周波数基準PWM動作に設定される時のCOM1x1,0ビット機能を示します。

表52. 位相基準または位相/周波数基準PWM動作での比較出力選択 (注: xはAまたはB, Xは0または1)

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

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。より多くの詳細については75頁の「位相基準PWM動作」をご覧ください。

■ビット3 – FOC1A : OC1A強制変更 (Force Output Compare 1A)

■ビット2 – FOC1B : OC1B強制変更 (Force Output Compare 1B)

FOC1A/FOC1BビットはWGM13~0ビットが非PWM動作を指示する時だけ有効です。けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR1Aが書かれる場合、これらのビットは0に設定されなければなりません。FOC1A/FOC1Bビットに論理1を書くと波形生成部で直ちに比較一致が強制されます。OC1x出力はCOM1x1,0ビット設定に従って変更されます。FOC1A/FOC1Bビットがスロープとして実行されることに注意してください。それによって強制された比較の効果を決めるのはCOM1x1N0ビットに存在する値です。

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

FOC1A/FOC1Bビットは常に0として読まれます。

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

タイマ/カウンタ制御レジスタB(TCCR1B)で得られるWGM13,2ビットと組み合わせたこれらのビットはカウンタの計数順序(方向)、最大計数(TOP)値供給元、使われるべき波形生成のどの形式かを制御します(表53.参照)。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と3形式のパルス幅変調(PWM)動作です。73頁の「動作種別」をご覧ください。

表53. 波形生成種別選択

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

注: CTC1とPWM11N0ビット定義名は旧名です。WGM12~0定義を使ってください。しかし、これらのビットの機能と位置は旧版のタイマ/カウンタと一致します。

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

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

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

このビットを(1)に設定することが捕獲起動入力雑音消去器を活性(有効)にします。雑音消去器が有効にされると、捕獲起動入力(ICP)ピンからの入力が増幅されます。この増幅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ICPピンの採取を必要とします。雑音消去器が許可されると、捕獲入力はそのによって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(ICP)ピンのどちらかのエッジを選択します。ICES1ビットが0を書かれると起動動作として下降(負)端が使われ、ICES1ビットが1を書かれると上昇(正)端が捕獲を起動します。

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

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

■ ビット5 – Res : 予約 (Reserved)

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

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

TCCR1AのWGM11,0ビット記述をご覧ください。

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

この3つのクロック選択ビットはタイマ/カウンタ(TCNT1)によって使われるべきクロック元を選択します。図56と図57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の制御を許します。

表54. タイマ/カウンタ1入力クロック選択

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

■ タイマ/カウンタ1 (Timer/Counter1) TCNT1H,TCNT1L (TCNT1)

ビット	15	14	13	12	11	10	9	8	
\$2D (\$4D)	(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	
\$2C (\$4C)								(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にアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。66ページの「16ビットレジスタのアクセス」をご覧ください。

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

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

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

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

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

ビット	15	14	13	12	11	10	9	8	
\$29 (\$49)	(MSB)								OCR1BH
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	
ビット	7	6	5	4	3	2	1	0	
\$28 (\$48)								(LSB)	OCR1BL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

この比較レジスタは容量16ビットです。CPUがこれらのレジスタへ書く時に上位と下位の両バイトが同時に書かれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。66頁の「16ビットレジスタのアクセス」をご覧ください。

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

ビット	15	14	13	12	11	10	9	8	
\$25 (\$45)	(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	
\$24 (\$44)								(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ICPピン(またはタイマ/カウンタ1については任意のアナログ比較器出力)で出来事が起こる毎にカウンタ(TCNT1)値で更新されます。この捕獲レジスタはタイマ/カウンタのTOP値を定義するのに使えます。

この捕獲レジスタは容量が16ビットです。CPUがこれらのレジスタをアクセスする時に上位と下位の両バイトが同時に読まれるのを保証するため、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の全ての16ビットレジスタによって共用されます。66頁の「16ビットレジスタのアクセス」をご覧ください。

■ タイマ/カウンタ割り込み許可レジスタ (Timer/Counter Interrupt Mask Register) TMSK

ビット	7	6	5	4	3	2	1	0	
\$39 (\$59)	TOIE1	OCIE1A	OCIE1B	–	TICIE1	–	TOIE0	OCIE0	TMSK
Read/Write	R/W	R/W	R/W	R	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

注: このレジスタは様々なタイマ/カウンタ用割り込み制御ビットを含みますが、タイマ/カウンタ1だけがこの項で記述されます。残りのビットは各々のタイマ/カウンタ項で記述されます。

■ ビット7 – TOIE1 : タイマ/カウンタ1溢れ割り込み許可 (Timer/Counter1 Overflow Interrupt Enable)

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

■ ビット6 – OCIE1A : タイマ/カウンタ1比較A割り込み許可 (Timer/Counter1 Output Compare A Match Interrupt Enable)

このビットが**1**を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(**1**)されると、タイマ/カウンタ1比較A一致割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置された**比較1A割り込み要求フラグ(OCF1A)**が設定(**1**)されると、対応する割り込みベクタ(32頁の「**割り込み**」参照)が実行されます。

■ ビット5 – OCIE1B : タイマ/カウンタ1比較B割り込み許可 (Timer/Counter1 Output Compare B Match Interrupt Enable)

このビットが**1**を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(**1**)されると、タイマ/カウンタ1比較B一致割り込みが許可されます。タイマ/カウンタ割り込み要求フラグレジスタ(TIFR)に配置された**比較1B割り込み要求フラグ(OCF1B)**が設定(**1**)されると、対応する割り込みベクタ(32頁の「**割り込み**」参照)が実行されます。

■ ビット3 – TICIE1 : タイマ/カウンタ1捕獲割り込み許可 (Timer/Counter1 Input Capture Interrupt Enable)

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

■ タイマ/カウンタ割り込み要求フラグレジスタ (Timer/Counter Interrupt Flag Register) TIFR

ビット	7	6	5	4	3	2	1	0	
\$38 (\$58)	TOV1	OCF1A	OCF1B	–	ICF1	–	TOV0	OCF0	TIFR
Read/Write	R/W	R/W	R/W	R	R/W	R	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

注: このレジスタは様々なタイマ/カウンタ用割り込み制御ビットを含みますが、タイマ/カウンタ1だけがこの項で記述されます。残りのビットは各々のタイマ/カウンタ項で記述されます。

■ ビット7 – TOV1 : タイマ/カウンタ1溢れ割り込み要求フラグ (Timer/Counter1 Overflow Flag)

このフラグの(**1**)設定はWGM13~0ビット設定に依存します。**標準**または**CTC動作**でのTOV1フラグはタイマ/カウンタ1溢れ時に設定(**1**)されます。他のWGM13~0ビット設定を使う時のTOV1フラグ動作については**79頁の表53**を参照してください。

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

■ ビット6 – OCF1A : タイマ/カウンタ1比較A割り込み要求フラグ (Timer/Counter1, Output Compare A Match Flag)

このフラグはカウンタ(TCNT1)値が**比較Aレジスタ(OCR1A)**と一致した後(次)のタイマ/カウンタクロック周期で設定(**1**)されます。

強制的な比較出力(FOC1A)ストローブがOCF1Aフラグを設定(**1**)しないことに注意してください。

比較A一致割り込みベクタが実行されると、OCF1Aは自動的に解除(**0**)されます。代わりにこのビット位置へ論理**1**を書くことによってOCF1Aは解除(**0**)できます。

■ ビット5 – OCF1B : タイマ/カウンタ1比較B割り込み要求フラグ (Timer/Counter1, Output Compare B Match Flag)

このフラグはカウンタ(TCNT1)値が**比較Bレジスタ(OCR1B)**と一致した後(次)のタイマ/カウンタクロック周期で設定(**1**)されます。

強制的な比較出力(FOC1B)ストローブがOCF1Bフラグを設定(**1**)しないことに注意してください。

比較B一致割り込みベクタが実行されると、OCF1Bは自動的に解除(**0**)されます。代わりにこのビット位置へ論理**1**を書くことによってOCF1Bは解除(**0**)できます。

■ ビット3 – 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)

ICPピンに捕獲の出来事が起こると、このフラグが設定(**1**)されます。捕獲レジスタ(ICR1)がWGM13~0によって**TOP値**として設定されると、ICF1フラグはカウンタがTOP値に到達する時に設定(**1**)されます。

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

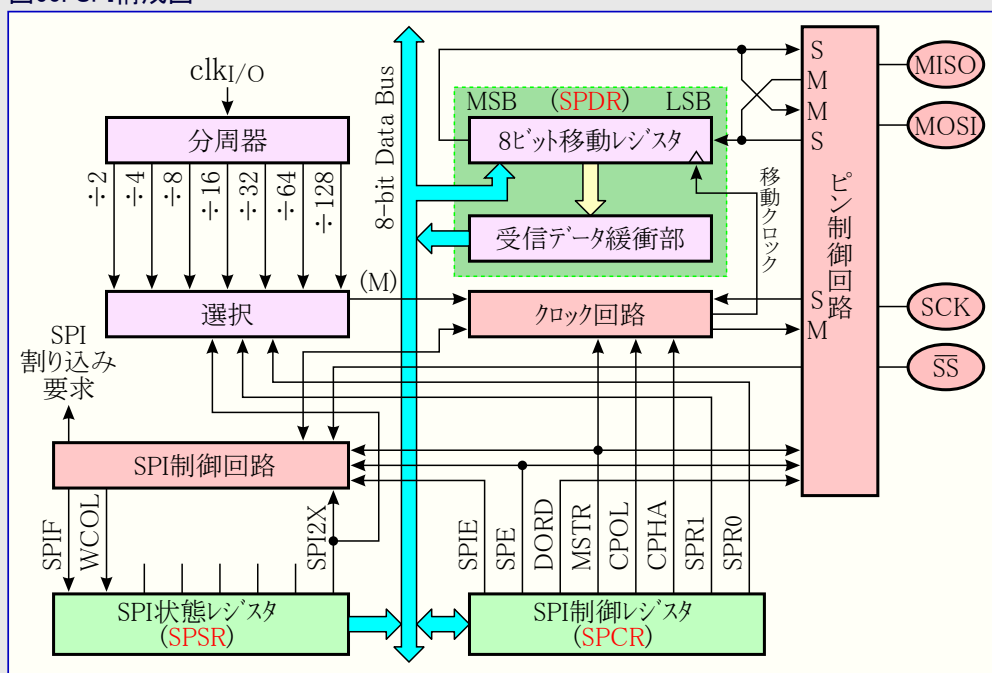
直列周辺インターフェース (SPI: Serial Peripheral Interface)

直列周辺インターフェースはATmega8515と様々なAVRデバイスや周辺デバイス間の高速同期データ転送を許します。ATmega8515のSPIは次の特徴を含みます。

- ・ 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

SPIでの主装置と従装置のCPU間相互接続は図61.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始めます。主装置と従装置は各々の移動レジスタで送るべきデータを用意し、主装置はデータを交換するのに必要なクロックパルスをSCK信号線に生成します。データは常にMOSI(Master Out Slave In)信号線を主装置から従装置へ、MISO(Master In Slave Out)信号線を従装置から主装置へ移動されます。各データパケット後、主装置はSS(従装置選択)ピンをHighへ引き上げることによって従装置と同期を取ります。

図60. SPI構成図



注: SPIピン配置については1頁の「ピン配置」と41頁の表29.を参照してください。

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クロック発生器を開始し、ハードウェアが従装置内へ8ビットを移動します。1バイトの移動後、SPIクロック発生器は停止し、SPI状態レジスタ(STATUS)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。しかし、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)から読まなければなりません。さもなければ最初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2CPU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55.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、39頁の「交換ポート機能」を参照してください。

図61. SPI 主装置/従装置の接続

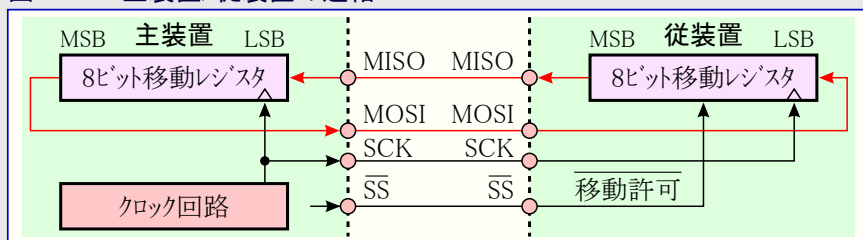


表55. SPIピン方向規定

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41頁の「ポートBの交換機能」を参照してください。

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

アセンブリ言語プログラム例

```
SPI_M_Init: LDI    R17, (1<<DD_MOSI) | (1<<DD_SCK)      ;MOSI, SCK=出力、他は入力値を取得
            OUT    DDR_SPI, R17                        ;MOSI, SCK=出力、他は入力に設定
            LDI    R17, (1<<SPE) | (1<<MSTR) | (1<<SPR0)  ;SPI許可、主装置、16分周値を取得
            OUT    SPCR, R17                          ;SPI許可、主装置、16分周に設定
            RET                                         ;呼び出し元へ復帰

SPI_M_Tx:   OUT    SPDR, R16                          ;データ(R16)送信開始
SPI_M_Tx_W: SBIS   SPSR, SPIF                        ;転送完了ならばスキップ
            RJMP   SPI_M_Tx_W                        ;転送完了まで待機

;

            RET                                         ;呼び出し元へ復帰
```

C言語プログラム例

```
void SPI_MasterInit(void)
{
    DDR_SPI = (1<<DD_MOSI) | (1<<DD_SCK);          /* MOSI, SCK=出力、他は入力に設定 */
    SPCR = (1<<SPE) | (1<<MSTR) | (1<<SPR0);        /* SPI許可、主装置、16分周に設定 */
}

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

注: 4頁の「コード例について」をご覧ください。

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

アセンブリ言語プログラム例

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

SPI_S_Rx:   SBIS   SPSR, SPIF                        ;受信(転送)完了ならばスキップ
            RJMP   SPI_S_Rx                        ;受信(転送)完了まで待機

;

            IN     R16, SPDR                        ;受信データを取得
            RET                                         ;呼び出し元へ復帰
```

C言語プログラム例

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

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

注: 4頁の「コード例について」をご覧ください。

SSピンの機能

従装置動作

SPIが従装置として設定されると、従装置選択(SS)ピンは常に入力です。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。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。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。

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、SSピンの方向は使用者が決められます。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SSピンを駆動するでしょう。

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。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SSピンがLowに駆動されると、SPIシステムは他の主装置が従装置として選択してデータ送信を始めると解釈します。バスの衝突を避けるためにSPIシステムは次の動作を行います。

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

従って割り込み駆動SPI送信が主装置動作で使われ、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。

■ SPI制御レジスタ (SPI Control Register) SPCR

ビット	7	6	5	4	3	2	1	0	
\$0D (\$2D)	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択します。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62と図63を参照してください。CPOL機能は右で要約されます。

表56. 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62と図63を参照してください。CPH A機能は右で要約されます。

表57. CPHA機能動作

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

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

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

表58. SCK速度選択 (f_{osc} =CPUクロック周波数)

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

■ SPI状態レジスタ (SPI Status Register) SPSR

ビット	7	6	5	4	3	2	1	0	
\$0E (\$2E)	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(1)ビット=1)されて、SPI制御レジスタ(SPCR)でSPI割り込み許可(SPIE)ビットが設定(1)されるなら、割り込みが生成されます。SPIが主装置動作の時にSSピンが入力でLowに駆動されるなら、これもこのSPIFフラグを同様に設定(1)します。対応する割り込み処理ベクタを実行する時にSPIFはハードウェアによって解除(0)されます。代わりにSPIFが設定(1)されたSPI状態レジスタ(SPSR)を始めに読み、その後にSPIデータレジスタ(SPCR)を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(SPCR)が書かれると、このWCOLビットが設定(1)されます。WCOLビット(とSPIFビット)はWCOLが設定(1)されたSPI状態レジスタ(SPSR)を始めに読み、その後にSPIデータレジスタ(SPCR)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として読まれます。

■ 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58.参照)。これは最小SCK周期が2CPUクロック周期であることを意味します。SPIが従装置として設定される時に、SPIは 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。

ATmega8515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126頁をご覧ください。

■ SPIデータレジスタ (SPI Data Register) SPDR

ビット	7	6	5	4	3	2	1	0	
\$0F (\$2F)	(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。

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62と図63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。これは以下で行われるように表56と表57を要約することによって明解にされます。

表59. 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62. SPIデータ転送形式 (CPHA=0)

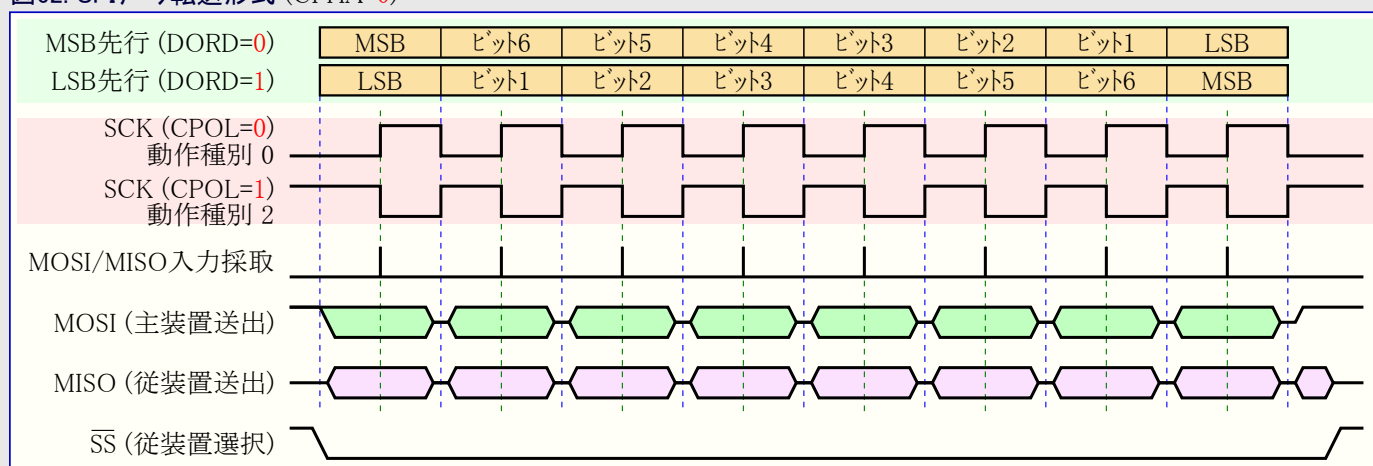
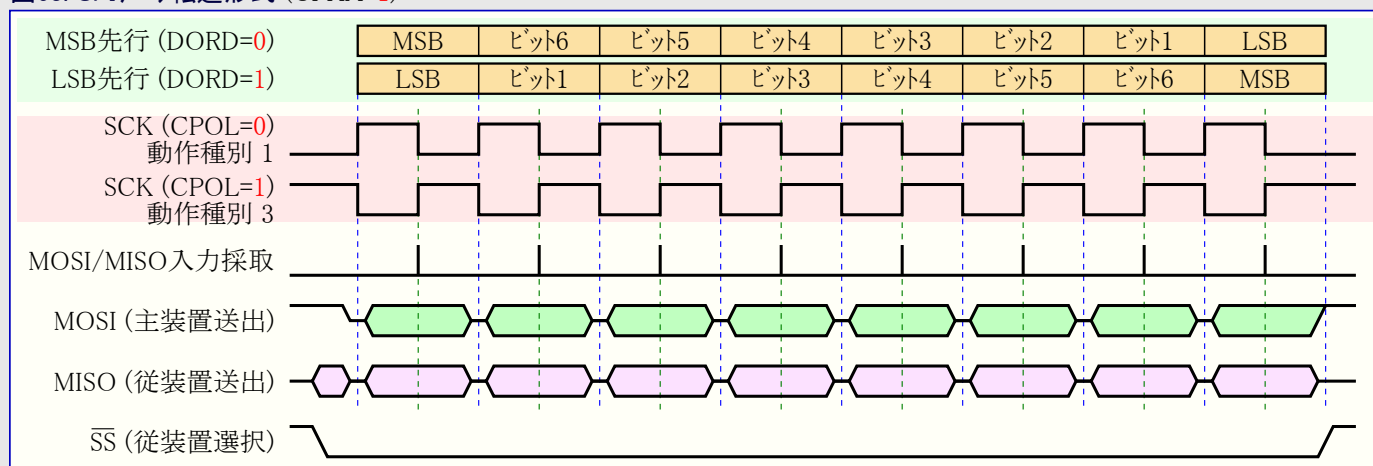


図63. SPIデータ転送形式 (CPHA=1)



USART

USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は高い柔軟性をもつ直列通信機能です。主な特徴を次に示します。

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

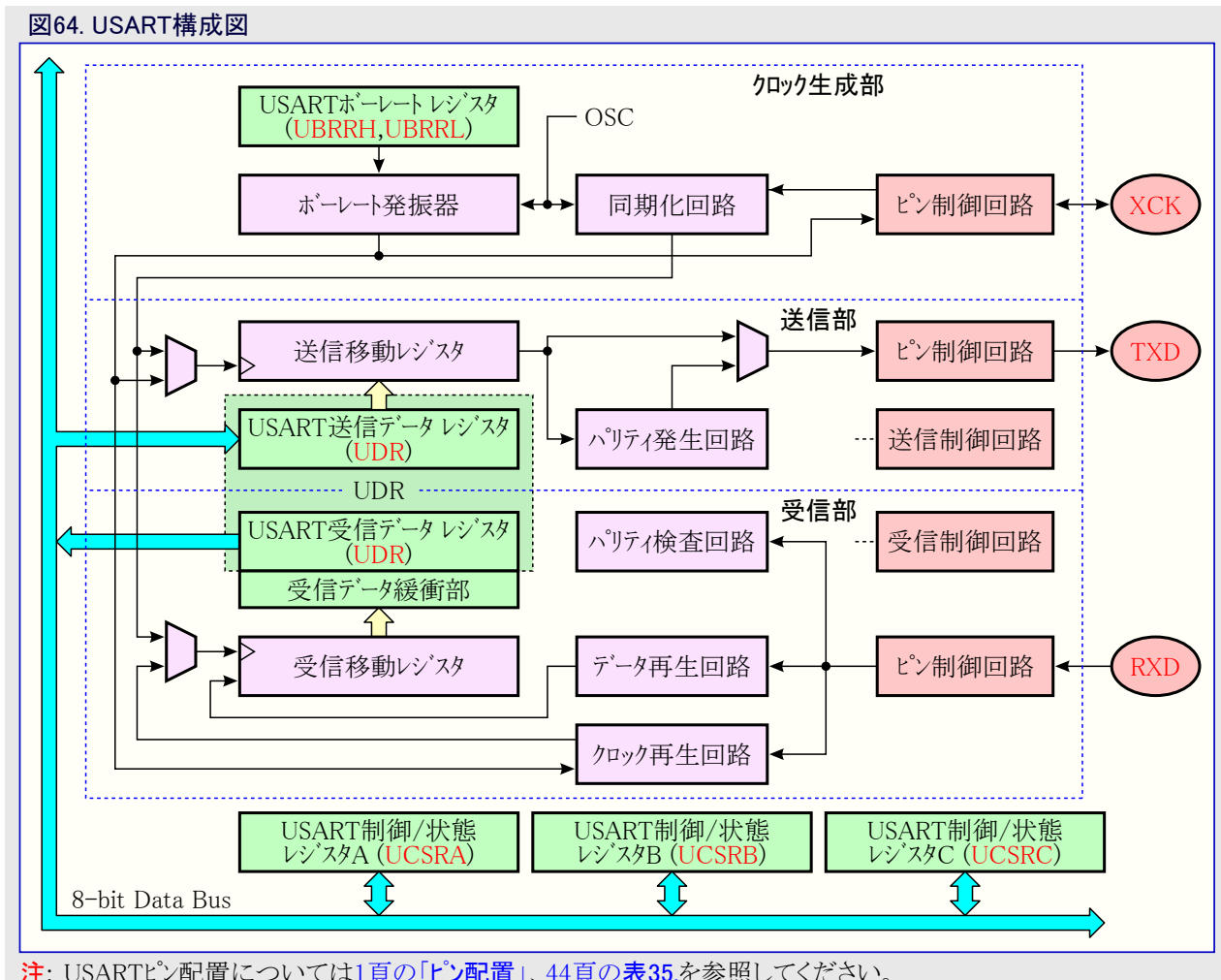
単一USART

ATmega8515には1つのUSARTがあります。USART機能については以下で記述されます。

AT90S4414/8515互換動作ではUSART受信レジスタの2重緩衝が禁止されることに注意してください。詳細については89頁の「UARTとの互換性」をご覧ください。

概要

USARTの簡便化した構成図は図64.で示されます。CPUがアクセス可能なレジスタとI/Oピンは赤文字(訳注:原文は太字)で示されます。



構成図内の破線はUSARTの3つの主要部分、(上から)クロック生成部、送信部、受信部を分けます。制御レジスタは全部によって共用されます。**クロック生成部**はボーレート発振器と同期従装置動作によって使われる外部クロック入力に対する同期化論理回路から成ります。転送クロック(XCK)ピンは同期転送動作だけで使われます。送信部は単一書き込み緩衝部(UDR)、直列移動レジスタ、パリティ発生器、異なる**直列フレーム形式**を扱うための制御論理回路から成ります。書き込み緩衝部はどんなフレーム間の遅れもなくデータの継続転送を許します。受信部はクロックとデータの再生部のため、USART部の最も複雑な部分です。再生部は非同期データ受信で使われます。再生部に加えて、受信部は2重の受信緩衝部(UDR)、移動レジスタ、パリティ検査器、制御論理回路を含みます。受信部は送信部と同じフレーム形式を支援し、**フレーミング異常**、**データ オーバーラン発生**、**パリティ誤り**を検知できます。

UARTとの互換性

USARTはAVRのUARTと次の項目に関して完全な互換性があります。

- 全てのUSART側レジスタでのビット位置
- ボーレート生成
- 送信操作
- 送信緩衝の動作
- 受信操作

けれども受信緩衝動作にはいくつかの特別な場合で互換性に影響を及ぼす2つの改良点があります。

- 第2受信緩衝部が追加されました。2つの緩衝レジスタは循環型FIFO緩衝部として動作します。従ってUSARTデータレジスタ(UDR)は到着データ毎に一度だけ読まれなければなりません。より重要なのは異常フラグ(FEとDOR)と第9データビット(RXB8)が受信緩衝部内のデータと共に緩衝されることの実事です。従って状態ビットは常にUSARTデータレジスタ(UDR)が読まれる前に読まれなければなりません。さもなければ緩衝部の状態が失われるため、異常情報も失われます。
- 受信部移動レジスタは第3緩衝段のように動けます。これは緩衝レジスタが一杯の場合、新規開始ビットが検出されるまで直列移動レジスタ(図64参照)内に留まるのを受信したデータに許すことによって行われます。従ってUSARTはデータオーバーラン(DOR)異常条件により耐えます。

次の制御ビットは名称変更されましたが、機能とレジスタ(ビット)位置は同じです。

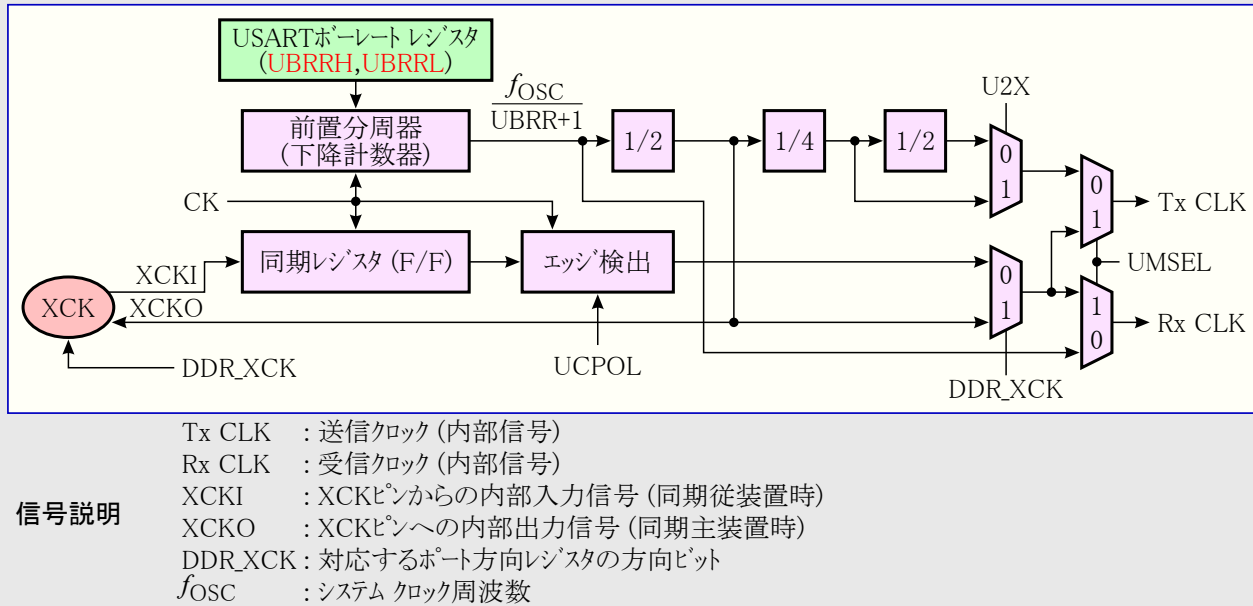
- 9ビット選択(CHR9)ビットはデータ長選択ビット2(UCSZ2)へ変更
- オーバーラン発生(OR)フラグはオーバーラン発生(DOR)フラグへ変更

クロック生成

クロック生成論理回路は送受信部用基準クロックを生成します。USARTは標準非同期、倍速非同期、同期主装置、同期従装置の4つのクロック動作種別を支援します。USART制御/状態レジスタ(UCSRC)のUSART動作種別選択(UMSEL)ビットは同期動作と非同期動作のどちらかを選びます。倍速動作(非同期動作のみ)はUSART制御/状態レジスタA(UCSRA)にある倍速許可(U2X)ビットによって制御されます。同期動作(UMSEL=1)を使うとき、XCKピンに対する方向制御ビット(DDR_XCK)はクロック元が内部(主装置動作)または外部(従装置動作)のどちらかを制御します。このXCKピンは同期動作を使う時だけ活性(有効)です。

図65.はクロック生成論理回路の構成図を示します。

図65. クロック生成部構成図



ボーレート発振器での内部クロック発生

内部クロック生成は非同期と同期主装置動作種別に対して使われます。本項の記述は図65.を参照してください。

USARTボーレート レジスタ(UBRR(UBRRH:UBRRL))と下降カウンタは設定変更可能な前置分周器またはボーレート発振器として機能するように接続されます。システム クロック(f_{OSC})で走行する下降カウンタは0への下降カウンタ時毎またはUBRRLレジスタが書かれる時にUBRR値で設定されます。1クロック(パルス)はカウンタが0に達する毎に生成されます。このクロックがボーレート発振器出力($=f_{OSC}/(UBRR+1)$)です。送信部は動作種別に依存してボーレート発振器出力を2,8,16分周します。ボーレート発振器出力は受信部クロックとデータ再生部によって直接使われます。しかし、再生部はUSART動作種別選択(UMSEL)、倍速許可(U2X)、DDR_XCKビットの状態によって設定される動作種別に依存して2,8,16段を使う順次処理回路を使います。

表60.は内部的に生成したクロック元を使う各動作種別に於けるボーレート(bps)とUBRR値の計算式を含みます。

表60. ボーレート レジスタ(UBRR)値計算式

動作種別	ボーレート計算式	UBRR値計算式
標準速非同期動作 (U2X=0)	$BAUD = \frac{f_{OSC}}{16 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{16 \times BAUD} - 1$
倍速非同期動作 (U2X=1)	$BAUD = \frac{f_{OSC}}{8 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{8 \times BAUD} - 1$
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRR+1)}$	$UBRR = \frac{f_{OSC}}{2 \times BAUD} - 1$

注: ボーレートは転送速度(ビット/1秒)で定義されます。

BAUD : ボーレート (bps)
 UBRR : UBRRHとUBRRLレジスタ値 (0~4095)
 f_{OSC} : システム発振器クロック周波数

いくつかのシステム クロック周波数に対するいくつかのUBRR値の例は表68.で得られます(104頁をご覧ください)。

倍速動作 (U2X)

転送速度はUSART制御/状態レジスタA(UCSRA)で倍速許可(U2X)ビットを設定(1)することによって倍にできます。このビットの設定は非同期動作に対してだけ有効です。同期動作を使うとき、このビットは0に設定してください。

このビットの設定(1)は事実上、非同期通信に対する転送速度を倍にするボーレート分周器の分周数を16から8に減じます。けれども受信部がデータ採取とクロック再生に対して半分の(16から8に減じた)採取数をこの場合だけ使い、従ってこの動作種別が使われる時はシステムクロックとボーレート設定の精度がより必要とされることに注意してください。送信部についての低下要因はありません。

外部クロック

外部クロックは同期従装置動作種別によって使われます。本項の記述での詳細については図65を参照してください。

XCKピンからの外部クロック入力是不確定レベル状態(メタステーブル)の機会を最少とするために同期化レジスタによって採取されます。同期化レジスタからの出力は送受信部で使い得るのに先立ってエッジ検出器を通過しなければなりません。この処理手順が2CPUクロック周期の遅延を持ち込み、このため最大外部XCKクロック周波数は次式によって制限されます。

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

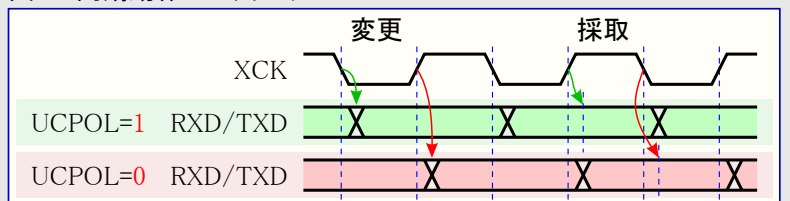
f_{OSC} がシステムクロック元の安定度に依存することに注意してください。従って周波数変動によるデータ消失の可能性を避けるため、いくらかの余裕分を追加することが推奨されます。

同期クロック動作

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

USART制御/状態レジスタC(UCSRC)のXCK極性(UCPOL)ビットはデータ採取とデータ変更に対してどちらのクロック端が使われるかを選びます。図66で示されるようにUCPOLが0のとき、データはXCKの上昇端で変更され、下降端で採取されます。UCPOLが設定(1)の場合、データはXCKの下降端で変更され、上昇端で採取されます。

図66. 同期動作XCKタイミング



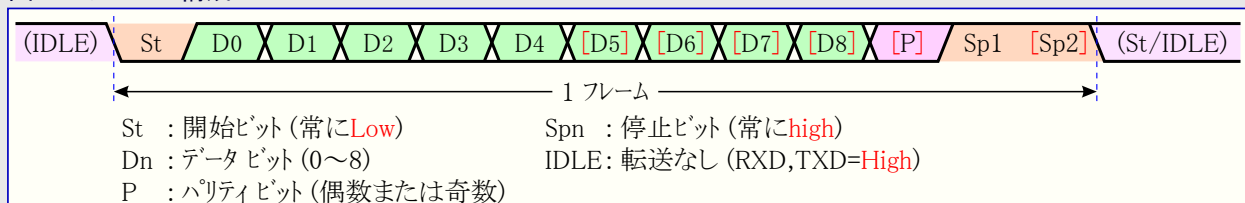
フレーム形式

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

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

フレームは最下位データビット(LSB)が次に続く開始ビットで始まります。その後次データビットが最後の最上位データビット(MSB)まで(最大合計9ビット続きます。許可したなら、パリティビットがデータビットの後、停止ビットの前に挿入されます。完全なフレームが送信されると、新規フレームによって直ちに後続されるか、または通信線をアイドル状態(high)に設定できます。図67は組み合わせ可能なフレーム形式を図解します。[]付きビットは任意選択です。

図67. 1フレームの構成



USARTによって使われるフレーム形式はUSART制御/状態レジスタBとC(UCSRB, UCSRC)でデータ長選択(UCSZ2~0)ビット、パリティ選択(UPM1,0)ビット、停止ビット選択(USBS)ビットによって設定されます。受信部と送信部は同じ設定を使います。これらのどのビットの設定変更も、送受信部両方に対して進行中の通信を不正とすることに注意してください。

データ長選択(UCSZ2~0)ビットはフレーム内のデータビット数を選びます。パリティ選択(UPM1,0)ビットはパリティビットの許可と種別(奇/偶)を設定します。1または2停止ビットのどちらかの選択は停止ビット選択(USBS)ビットによって行います。受信部は第2停止ビットを無視します。従ってフレーミング異常(FE)は最初の停止ビットが0(Low)の場合にだけ検出されます。

パリティビットの計算

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

偶数パリティビット = D0 Ex-OR D1 Ex-OR D2 Ex-OR ~ Ex-OR Dn-1
奇数パリティビット = D0 Ex-OR D1 Ex-OR D2 Ex-OR ~ Ex-OR Dn-1 Ex-OR 1 n : データビット長

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

USARTの初期化

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

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

次の簡単なUSART初期化コード例が示すアセンブリ言語とC言語の関数は機能的に同じです。本例は固定フレーム形式でボーリングを使う(割り込み不許可)非同期動作と仮定します。ボーレート(UBRR)値は関数の引数として与えられます。アセンブリ言語でのボーレート引数はR17:R16レジスタに格納されると仮定されます。関数が[USART制御/状態レジスタC\(UCSRC\)](#)へ書くとき、[ボーレートレジスタ上位\(UBRRH\)](#)とUCSRCによるI/O位置の共有のため、[レジスタ選択\(URSEL\)ビット](#)が設定(1)されなければなりません。

アセンブリ言語プログラム例

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

C言語プログラム例

```
#define FOSC 1843200 /* MCUクロック周波数 */
#define BAUD 9600    /* 目的USARTボーレート速度 */
#define MYUBRR FOSC/16/BAUD-1 /* 目的UBRR値 */

void main(void)
{
    ~~~~~
    USART_Init(MYUBRR); /* USART初期化 */
    ~~~~~
}

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

注: 4頁の「コード例について」をご覧ください。

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

データ送信 - USART送信部

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

5～8ビット データ フレーム送信

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

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「コード例について」をご覧ください。

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

9ビット データ フレーム送信

9ビット データが使われる場合(UCSZ2～0=111)、データの下位8ビットがUSARTデータレジスタ(UDR)に書かれる前に、第9ビットがUSART制御/状態レジスタB(UCSRB)の送信データビット8(TXB8)ビットに書かれなければなりません。次のコード例は9ビット データを扱う送信関数を示します。アセンブリ言語での送るべきデータはR17:R16レジスタに格納されると仮定されます。

アセンブリ言語プログラム例

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

C言語プログラム例

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

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

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

送信フラグと割り込み

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

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

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

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

UCSRBで**送信完了割り込み許可(TXCIE)ビット**が設定(1)され、(全割り込みが許可されていれば)TXCフラグが設定(1)になる時にUSART送信完了割り込みが実行されます。送信完了割り込みが使われるとき、割り込み処理ルーチンはTXCフラグを解除(0)しなくてもよく、これは割り込みが実行されるとき、自動的に行われます。

パリティ発生器

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

送信の禁止

送信部の禁止(UCSRBの**USART送信許可(TXEN)ビット**=0)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない時)まで有効になりません。禁止されると、送信部はもはやTXDピン(の標準ピン機能)を無効にしません。

データ受信 - USART受信部

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

5～8ビットデータフレーム受信

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

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「コード例について」をご覧ください。

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

9ビット データ フレーム受信

9ビット データが使われる場合(UCSZ2~0=111)、USARTデータレジスタ(UDR)から下位バイトを読む前に、第9ビットがUSART制御/状態レジスタB(UCSRB)の受信データビット8(RXB8)ビットから読まなければならない。この規則はフレーミング異常(FE)、オーバーラン発生(DOR)、パリティ誤り(PE)状態フラグにも適用します。USART制御/状態レジスタA(UCSRA)から状態情報を読み、その後にUDRからデータを読んでください。UDR I/O位置を読むことが受信FIFO緩衝部の状態を切り替え、その結果、FIFO内に保管されるRXB8,FE,DOR,PEビット全てが切り替わります。

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

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「コード例について」をご覧ください。

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

受信完了フラグと割り込み

USART受信部には受信部の状態を示す1つのフラグがあります。

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

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

受信異常フラグ

USART受信には3つの異常フラグ、フレーミング異常(FE)、データ オーバーラン発生(DOR)、パリティ誤り(PE)があります。全てが**USART制御/状態レジスタA(UCSRA)**を読むことによってアクセスできます。異常フラグに対する共通点は異常状態を示しているフレームと共に受信緩衝部に配置されることです。異常フラグが緩衝されるため、**USARTデータ レジスタ(UDR)** I/O位置を読むことが緩衝部読み出し位置を切り替えるので、UCSRAは受信緩衝部(UDR)の前に読まれなければなりません。異常フラグに対するその他の共通点はソフトウェアがこのフラグ位置へ書き込みを行うことによって変更できないことです。しかし、将来のUSART(機能)実装の上位互換性のため、UCSRAが書かれるとき、全てのフラグは**0**に設定されなければなりません。異常フラグはどれも割り込みを生成できません。

フレーミング異常(FE)フラグは受信緩衝部に格納された、次に読み込み可能なフレームの第1停止ビットの状態を示します。FEフラグは停止ビットが正しく(**High**として)読まれた時に**0**で、停止ビットが不正(**Low**)だった時にFEフラグは**1**です。このビットは同期外れ状態の検出、中断状態検出、規約での操作に使えます。受信部が最初(第1)以外の全停止ビットを無視するため、FEフラグは**USART制御/状態レジスタC(UCSRC)**の**停止ビット選択(USBS)ビット**設定によって影響を及ぼされません。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットを**0**に設定してください。

データ オーバーラン発生(DOR)フラグは受信緩衝部が一杯状態のためのデータ消失を示します。データ オーバーランは受信緩衝部(2フレーム)が一杯で、(次の)新規フレーム データが受信移動レジスタで待っており、(更に次の)新規開始ビットが検出される時に起きます。DORフラグが設定(**1**)なら、最後にUDRから読んだフレームと次にUDRから読むフレーム間で1つ以上の直列フレームが失われています。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットに**0**を書いてください。DORフラグは受信されたデータが移動レジスタから受信緩衝部へ正常に移動された時に解除(**0**)されます。

パリティ誤り(PE)フラグは受信緩衝部内の次フレームで受信時にパリティ異常があったことを示します。パリティ検査が許可されていない場合、PEフラグは常に**0**が読めます。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットを**0**に設定してください。より多くの詳細については92頁の「**パリティビットの計算**」と次の「**パリティ検査器**」をご覧ください。

パリティ検査器

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

PEフラグは受信緩衝部から読まれ得る次フレームで受信時にパリティ異常があり、その時点でパリティ検査が許可されていた(**UPM1=1**)場合に設定(**1**)されます。このビットはUSARTデータ レジスタ(UDR)が読まれるまで有効です。

受信の禁止

送信部と対照的に受信部の禁止は即時です。従って受信進行中のデータは失われます。禁止されると(換言すると、**USART制御/状態レジスタB(UCSRB)**の**USART受信許可(RXEN)ビット**が**0**に設定)、受信部はもはやRXDポート ピンの標準機能を無効にしません。受信FIFO緩衝部は受信部が禁止されると破棄されます。緩衝部内の残データは失われます。

受信緩衝部の破棄

受信FIFO緩衝部は受信が禁止されると破棄(換言すると、緩衝部は内容を空に)されます。未読データは失われます。例えば異常状態のため、通常動作中に緩衝部が破棄されなければならない場合、**USART制御/状態レジスタA(UCSRA)**の**受信完了(RXC)フラグ**が解除(**0**)されるまでUSARTデータ レジスタ(UDR) I/O位置を読んでください。次のコード例は受信緩衝部の破棄方法を示します。

アセンブリ言語プログラム例

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

C言語プログラム例

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

注: 4頁の「**コード例について**」をご覧ください。

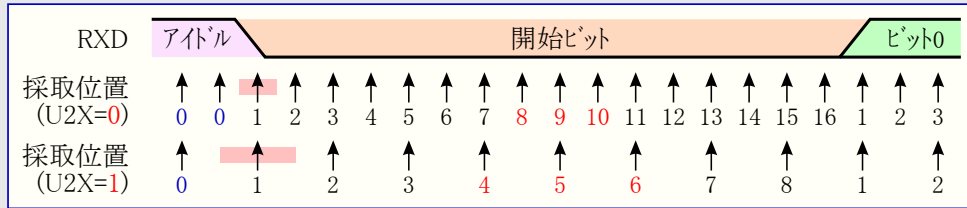
非同期受信

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

非同期クロック再生

クロック再生論理回路は内部クロックを到着直列フレームに同期化します。図68.は到着フレームの開始ビット採取手順を図解します。採取速度は標準動作でボーレートの16倍、倍速動作で8倍です。赤帯(訳注:原文は水平矢印)は採取処理のための同期変量を図示します。倍速動作(U2X=1)を使う時の広い変量時間に注意してください。採取番号0はRXD信号がアイドル(換言すると、通信の動きなし)の時にされる採取です。

図68. 開始ビットの採取

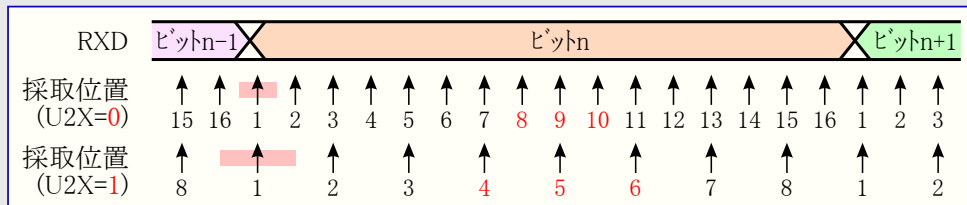


クロック再生論理回路がRXD信号線でHigh(アイドル)からLow(開始)の遷移を検出すると、開始ビット検出手順が開始されます。図で示されるように採取1は最初のLow採取を意味します。その後にクロック再生回路は有効な開始ビットが受信されるかを決定するために、標準動作に対して採取8,9,10、倍速動作に対して採取4,5,6(図の赤字(訳注:原文は箱枠内)で示された採取番号)を使います。これら3回で2回以上の採取が論理Highレベルの場合(多数決)、この開始ビットは尖頭雑音として捨てられ、受信部は次のHighからLowの遷移を探し始めます。けれども有効な開始ビットが検出されると、クロック再生論理回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

非同期データ再生

受信部クロックが開始ビットに同期化されるとデータ再生が始まります。データ再生部は標準動作で16段、倍速動作で8段の順列回路を使います。図69.はデータビットとパリティビットの採取を示します。各採取は再生部の各段に等しい番号を与えられます。

図69. データビットとパリティビットの採取



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

図70.は停止ビットの採取と次フレームの最も早い開始ビット開始の可能性を示します。

図70. 停止ビットの採取と次の開始ビットの採取



フレーム内の他のビットに対して行われるのと同じ多数決が停止ビットにも行われます。停止ビットが論理0値と記録されると、フレーミング異常(FE)フラグが設定(1)されます。

新規フレームの開始ビットを示すHighからLowへの遷移は多数決に使った最後のビット後に実現できます。標準動作での最初のLowレベル採取は図70.のA点で有り得ます。倍速動作での最初のLowレベルはB点に遅れます。C点は完全な長さの停止ビット(の終点)を示します。この早い開始ビット検出は受信部の動作範囲に影響します。

非同期での動作範囲

受信部の動作範囲は受信したビット速度と内部的に生成したボーレート間の不一致に依存します。送信部が速すぎるまたは遅すぎるビット速度でフレームを送出したり、内部的に発生した受信部のボーレートが類似した(表61.参照)基準周波数を持たない場合、受信部は開始ビットでフレームを同期できません。

次式は到着データ速度と内部受信部ボーレート間の比率計算に使えます。

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

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

表61. は許容できる最大受信部ボーレート誤差一覧です。標準速動作には、より高いボーレート変動許容能力があることに注目してください。

表61. 標準速と倍速での受信部ボーレート推奨最大許容誤差

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

注: Dはデータビット数とパリティビットの合計ビット数です。

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

受信部ボーレートの推奨最大許容誤差は最大総合許容誤差を送信部と受信部で等分割するという仮定の下で作られました。

受信部ボーレート誤差に対して2つの起こり得る原因があります。受信部のシステムクロック(XTAL)は供給電圧範囲と温度範囲に関して常に若干の不安定性があります。システムクロックを生成するのにクリスタル発振子を使う時は殆ど問題ありませんが、(セラミック)振動子でのシステムクロックは振動子偏差に依存して2%を越えて異なるかもしれません。2つ目の誤り原因はより制御可能です。ボーレート発振器は欲したボーレートを得るためにシステム周波数の正確な分周を常に行うことはできません。この場合、可能ならば受け入れ可能な低い誤差を与えるUBRR値が使えます。

複数プロセッサ通信動作

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

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

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

複数プロセッサ通信の使用法

主MCUとして動作するMCUは9ビットデータフレーム形式(**UCSZ=7**)を使えます。UCSRBの**送信第9(TXB8)ビット**はアドレスフレーム時に設定(1)、またはデータフレーム時に解除(0)されて送信されなければなりません。この場合、従MCUは9ビットデータフレーム形式の使用に設定されなければなりません。

複数プロセッサ通信動作でデータを交換するのに次の手順が使われるべきです。

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

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

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

UBRRH/UCSRCレジスタのアクセス

ボーレートレジスタ上位(UBRRH)はUSART制御/状態レジスタC(UCSRC)と同じI/O位置を共用しています。従ってこのI/O位置をアクセスする時にいくつかの特別な考慮が必要とされなければなりません。

UBRRH/UCSRCへの書き込み

このI/O位置の書き込みアクセスを行うとき、書かれる値の最上位ビット、USARTレジスタ選択(URSEL)ビットが2つのレジスタのどちらの1つが書かれるのかを制御します。書き込み操作中にURSELビットが0ならばUBRRH値が更新されます。URSELビットが1ならばUCSRC設定が更新されます。

次のコード例はこの2つのレジスタのアクセス法を示します。

アセンブリ言語プログラム例

```

    LDI    R16, 2                ;UBRRH値(2)を取得
    OUT    UBRRH, R16           ;UBRRHに2を設定

    LDI    R16, (1<<URSEL) | (1<<USBS) | (1<<UCSZ1) ;UCSRCのUSBS,UCSZ1のみ1設定
    OUT    UCSRC, R16           ;その他ビット=0

```

C言語プログラム例

```

UBRRH = 2; /* UBRRHに2を設定 */

UCSRC = (1<<URSEL) | (1<<USBS) | (1<<UCSZ1); /* UCSRCのUSBS,UCSZ1のみ1設定 */

```

注: 4頁の「コード例について」をご覧ください。

このコード例が示すように2つのレジスタの書き込みアクセスは相互にI/O位置共用の影響を受けません。

UBRRH/UCSRCからの読み込み

UBRRHまたはUCSRCに読み込みアクセスを行うのはより複雑な操作です。けれども殆どの応用ではこれらのレジスタのどれかを読む必要は稀です。

この読み込みアクセスは経過時間によって制御されます。このI/O位置を読むと、一旦UBRRHレジスタ内容を返します。このI/O位置が直前のシステムクロック周期で読まれたなら、現在のクロック周期でのレジスタ読み込みはUCSRC内容を返します。UCSRC読み込みの時間による手順が非分断操作であることに注意してください。従って読み込み操作中、割り込みは(例えば全割り込み禁止によって)制御されなければなりません。

次のコード例はUCSRCレジスタ内容の読み方を示します。

アセンブリ言語プログラム例

```

USART_R_UCSRC: IN    R16, UBRRH    ;初回読み込み(擬似)
                IN    R16, UCSRC    ;UCSRC内容取得
                RET                ;呼び出し元へ復帰

```

C言語プログラム例

```

unsigned char USART_RradUCSRC(void)
{
    unsigned char ucsrc; /* 戻り値変数定義 */
    ucsrc = UBRRH; /* 初回読み込み(擬似) */
    ucsrc = UCSRC; /* UCSRC内容取得 */
    return ucsrc; /* 呼び出し元へ復帰 */
}

```

注: 4頁の「コード例について」をご覧ください。

このアセンブリ言語例はR16にUCSRC値を返します。

UBRRH内容の読み込みは非分断操作ではなく、従って直前の命令がこのレジスタ位置をアクセスしない限り、通常のレジスタとして読むことができます。

USART用レジスタ

■ USARTデータレジスタ (USART I/O Data Register) UDR

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

同じI/Oアドレスを共用するUSART受信データ緩衝レジスタとUSART送信データ緩衝レジスタはUSARTデータレジスタまたはUDRとして引用しました。送信データ緩衝レジスタ(TXB)はUDRレジスタ位置に書かれるデータの転送先です。UDRレジスタ位置読み込みは受信データ緩衝レジスタ(RXB)の内容を返します。

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

送信緩衝部はUCSRAで送信データレジスタ空き(UDRE)フラグが設定(1)される時にだけ書けます。UDREフラグが設定(1)されない時にUDRへ書かれたデータはUSART送信部によって無視されます。送信部が許可されて送信緩衝部にデータが書かれると、送信部は送信移動レジスタが空きの時にそのデータを送信移動レジスタへ設定(移動)します。その後にデータはTXDピンで直列送信されます。

受信緩衝部は2段のFIFOから成ります。このFIFOは受信緩衝部がアクセスされる毎にその状態を切り替えます。この受信緩衝部の動きのため、この位置に読み-修正-書き(リード モデファイライト)命令(SBIとCBI)を使えません。ビット検査(SBICとSBIS)命令もFIFOの状態を換えるので、これらの命令を使う時は注意してください。

■ USART制御/状態レジスタ (USART Control and Status Register A) UCSRA

ビット	7	6	5	4	3	2	1	0	
\$0B (\$2B)	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
初期値	0	0	1	0	0	0	0	0	

■ ビット7 - RXC : USART受信完了フラグ (USART Receive Complete)

このフラグは受信緩衝部に未読データがある時に設定(1)され、受信緩衝部が空の(換言すると、どんな未読データも含まない)時に解除(0)されます。受信部が禁止されると、受信緩衝部が破棄され、その結果RXCフラグは0になります。RXCフラグは受信完了割り込みを発生するのに使えます(USART制御/状態レジスタB(UCSRB)の受信完了割り込み許可(RXCIE)ビットをご覧ください)。

■ ビット6 - TXC : USART送信完了フラグ (USART Transmit Complete)

このフラグは送信移動レジスタ内の完全なフレームが移動出力されてしまい、送信緩衝部(UDR)に新規データが現状存在しない時に設定(1)されます。TXCフラグは送信完了割り込みが実行されるとき、自動的に解除(0)されるか、またはこのビット位置に1を書くことによって解除(0)できます。TXCフラグは送信完了割り込みを発生できます(UCSRBの送信完了割り込み許可(TXCIE)ビットをご覧ください)。

■ ビット5 - UDRE : USART送信データレジスタ空きフラグ (USART Data Register Empty)

UDREフラグは送信緩衝部(UDR)が新規データを受け取る準備ができていかどうかを示します。UDREが1ならば緩衝部は空で、従って書かれる準備ができています。UDREフラグは送信緩衝部空き割り込みを発生できます(UCSRBの送信データレジスタ空き割り込み許可(UDRIE)ビットをご覧ください)。送信部が準備できているのを示すため、リセット後のUDREは設定(1)です。

■ ビット4 - FE : フレーミング異常フラグ (Framing Error)

受信緩衝部の次データが受信した時にフレーミング異常(換言すると、受信緩衝部で次データの最初の(第1)停止ビットがLow)だった場合、このビットが設定(1)されます。このフラグは受信緩衝部(UDR)が読まれるまで有効です。受信したデータの停止ビットがHighの時にFEフラグは0です。UCSRAに書くとき、常にこのビットを0に設定してください。

■ ビット3 - DOR : データ オーバーラン発生フラグ (Data OverRun)

このビットはオーバーラン状態が検出されると設定(1)されます。受信緩衝部(2フレーム分)が一杯で、新規フレームが受信移動レジスタ内で待機中に新規開始ビットが検出されるとデータ オーバーランが起こります。UCSRAに書くとき、常にこのビットを0に設定してください。

■ ビット2 - PE : パリティ誤りフラグ (Parity Error)

受信緩衝部の次データが受信した時にパリティ異常があり、その時点でパリティ検査が許可されていれば(UPM1=1)、このビットが設定(1)されます。このフラグは受信緩衝部(UDR)が読まれるまで有効です。UCSRAに書くとき、常にこのビットを0に設定してください。

■ ビット1 - U2X : 倍速許可 (Double the USART Transmission Speed)

このビットは非同期動作でだけ有効です。同期動作を使うとき、このビットに0を書いてください。

このビットに1を書くことはボーレート分周器の分周値を16から8に減らして事実上、非同期通信の転送速度を倍にします。

■ ビット0 - MPCM : 複数プロセッサ通信動作 (Multi-Processor Communication Mode)

このビットは複数プロセッサ通信動作を許可します。MPCMビットが1を書かれると、USART受信部で受信したアドレス情報を含まない全到着フレームは無視されます。送信部はMPCM設定に影響されません。より多くの詳細情報については99頁の「複数プロセッサ通信動作」をご覧ください。

■ USART制御/状態レジスタB (USART Control and Status Register B) UCSRB

ビット	7	6	5	4	3	2	1	0	
\$0A (\$2A)	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7 – RXCIE : 受信完了割り込み許可 (Receive Complete Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の受信完了(RXC)フラグでの割り込みを許可します。USART受信完了割り込みはRXCIEビットが1を書かれ、ステータスレジスタ(SREG)で全割り込み許可(I)ビットが1を書かれ、UCSRAでRXCフラグが設定(1)される場合にだけ生成されます。

■ ビット6 – TXCIE : 送信完了割り込み許可 (Transmit Complete Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の送信完了(TXC)フラグでの割り込みを許可します。USART送信完了割り込みはTXCIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでTXCフラグが設定(1)される場合にだけ生成されます。

■ ビット5 – UDRIE : 送信データレジスタ空き割り込み許可 (Transmit Data Register Empty Interrupt Enable)

このビットに1を書くことはUSART制御/状態レジスタA(UCSRA)の送信データレジスタ空き(UDRE)フラグでの割り込みを許可します。USART送信データレジスタ空き割り込みはUDRIEビットが1を書かれ、SREGで全割り込み許可(I)ビットが1を書かれ、UCSRAでUDREフラグが設定(1)される場合にだけ生成されます。

■ ビット4 – RXEN : 受信許可 (Receiver Enable)

このビットに1を書くことはUSART受信(部)を許可します。受信部は許可されるとRXDピンの標準ポート動作を無効にします。受信(部)を禁止することは受信緩衝部を破棄し、フレーミング異常(FE)、オーバラン(DOR)、パリティ誤り(PE)のフラグを無効にします。

■ ビット3 – TXEN : 送信許可 (Transmitter Enable)

このビットに1を書くことはUSART送信(部)を許可します。送信部は許可されるとTXDピンの標準ポート動作を無効にします。送信(部)の禁止(TXEN=0書き込み)は進行中と保留中の送信が完了される(換言すると、送信移動レジスタと送信緩衝レジスタが送信されるべきデータを含まない)まで有効になりません。禁止したとき、送信部はもはやTXDポート(の標準I/O機能)を無効にしません。

■ ビット2 – UCSZ2 : データビット長選択2 (Character Size)

USART制御/状態レジスタC(UCSRC)のUCSZ1,0ビットと組み合わせたUCSZ2ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

■ ビット1 – RXB8 : 受信データビット8 (Receive Data Bit 8)

RXB8は9ビットデータでの直列フレーム操作時に受信したフレームの第9データビット(ビット8)です。UDRから下位ビットを読む前に読んでください。

■ ビット0 – TXB8 : 送信データビット8 (Transmit Data Bit 8)

TXB8は9ビットデータでの直列フレーム操作時に送信されるべきデータの第9データビット(ビット8)です。UDRへ下位ビットを書く前に書いてください。

■ USART制御/状態レジスタC (USART Control and Status Register C) UCSRC

ビット	7	6	5	4	3	2	1	0	
\$20 (\$40)	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	1	0	0	0	0	1	1	0	

UCSRCレジスタはUBRRHレジスタと同じI/O位置を共用します。このレジスタのアクセス法を記述する項、100頁の「UBRRH/UCSRCレジスタのアクセス」をご覧ください。

■ ビット7 – URSEL : レジスタ選択 (Register Select)

このビットはUCSRCまたはUBRRHレジスタのどちらをアクセスするかを選択します。UCSRCを読むと1として読めます。UCSRCに書くとき、URSELは1でなければなりません。

■ ビット6 – UMSEL : USART動作選択 (USART Mode Select)

このビットは非同期と同期の動作種別のどちらかを選びます。

表63. USART動作選択

UMSEL	動作種別
0	非同期動作
1	同期動作

■ ビット5,4 – UPM1,0 : パリティ選択 (Parity Mode)

これらのビットはパリティの発生と検査の許可と種別を設定します。許可した場合、送信部は各フレーム内での送信したデータビットのパリティを自動的に生成して送出します。受信部は到着データからパリティ値を生成し、UPM0設定と比較します。不一致が検出されると、**USART制御/状態レジスタA(UCSRA)**で**パリティ誤り(PE)フラグ**が設定(1)されます。

表64. パリティ選択

UPM1	UPM0	パリティ動作
0	0	禁止
0	1	(予約)
1	0	偶数パリティ許可
1	1	奇数パリティ許可

■ ビット3 – USBS : 停止ビット選択 (Stop Bit Select)

このビットは送信部によって挿入される停止ビット数を選びます。受信部はこの設定を無視します(**訳補**:常に第1停止ビットだけが有効)。

表65. 停止ビット選択

USBS	停止ビット数
0	1ビット
1	2ビット

■ ビット2,1 – UCSZ1,0 : データビット長選択 (Character Size)

USART制御/状態レジスタB(UCSRB)の**UCSZ2ビット**と組み合わせたUCSZ1,0ビットは送受信部で使うフレームのデータビット数(Character size)を設定します。

表66. データビット長選択

UCSZ2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
データビット数	5ビット	6ビット	7ビット	8ビット	(予約)	(予約)	(予約)	9ビット

■ ビット0 – UCPOL : クロック極性選択 (Clock Polarity)

このビットは同期動作に対してだけ使われます。非同期動作が使われるとき、このビットに0を書いてください。UCPOLビットは同期クロック(XCK)、データ出力変更、データ入力採取間の関係を設定します。

表67. XCKクロック極性選択

UCPOL	送信データ変更 (TXDピン出力)	受信データ採取 (RXDピン入力)
0	XCKの上昇端	XCKの下降端
1	XCKの下降端	XCKの上昇端

■ USARTボーレートレジスタ (USART Baud Rate Register) UBRRH, UBRL (UBRR)

ビット	15	14	13	12	11	10	9	8	
\$20 (\$40)	URSEL	–	–	–	UBRR11	UBRR10	UBRR9	UBRR8	UBRRH
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	6	5	4	3	2	1	0	
\$09 (\$29)	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0	UBRL
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	

UBRRHレジスタはUCSRCレジスタと同じI/O位置を共用します。このレジスタのアクセス法を記述する項、100頁の「**UBRRH/UCSRCレジスタのアクセス**」をご覧ください。

■ ビット15 – URSEL : レジスタ選択 (Register Select)

このビットはUBRRHまたはUCSRCレジスタのどちらをアクセスするかを選択します。UBRRHを読むと0として読めます。UBRRHに書くとき、このビットは0でなければなりません。

■ ビット14~12 – Res : 予約 (Reserved Bits)

これらのビットは将来使用するために予約されています。将来のデバイスとの共通性のため、UBRRHが書かれるとき、これらのビットは0が書かれなければなりません。

■ ビット11~0 – UBRR11~0 : ボーレート分周値 (USART Baud Rate Register)

これはUSARTのボーレートを含む12ビットレジスタです。UBRRHがUSARTボーレートの上位4ビットを含み、UBRLが下位8ビットを含みます。ボーレートが変更されると、送受信部で進行中の転送は不正にされます。UBRL書き込みはボーレート前置分周器の更新を直ちに始めます。

ボーレート設定例

標準的なクリスタル発振子やセラミック振動子の周波数に対して最も一般的に使われる非同期動作のボーレートは表68.のUBRR設定を使うことによって生成できます。目的のボーレートに対して設定誤差1.5%(標準速:U2X=0)、1.0%(倍速:U2X=1)以上を赤字で示します(訳注:原書の本文は0.5%未満を太字)。より高い誤差率でも受け入れ可能ですが、特に長い直列フレーム(98頁の「非同期での動作範囲」参照)で誤差率が高いと、受信部は雑音耐性が低下します。誤差率は次式を使って計算されます。

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

表68. Xtal、ボーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

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

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

注: 最高速はUBRR=0、誤差=0.0%です。以降の周波数は次頁へ続く。

(訳注) 原書では頁割された表毎に表番号68.~71.となっていますが、共通性から纏めて表68.としました。
原書に対して数種の発振周波数を追加しました。

表68 (続き). Xtal、ホーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

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

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

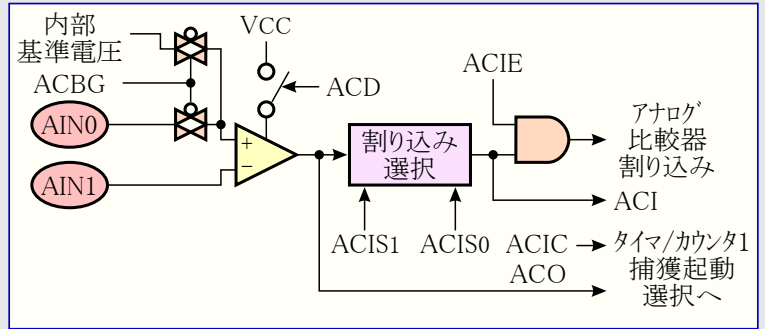
ホーレート (bps)	14.7456MHz				16MHz				18.4320MHz				20MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	767	0.0	1533	0.0	832	0.0	1666	0.0	959	0.0	1919	0.0	1041	0.0	2082	0.0
2400	383	0.0	767	0.0	416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

注: 最高速はUBRR=0、誤差=0.0%です。

アナログ比較器

アナログ比較器は非反転入力AIN0ピンと反転入力AIN1ピンの入力値を比較します。非反転AIN0ピンの電圧が反転AIN1ピンの電圧よりも高い時に**ACSRのアナログ比較器出力(ACO)ビット**が設定(1)されます。この比較器出力はタイマ/カウンタ1の捕獲機能を起動するように設定できます。加えて、この比較器はアナログ比較器専用の、独立した割り込みを起動できます。使用者は比較器出力の上昇端、下降端、またはその両方で割り込み起動を選べます。この比較器とその周辺論理回路の構成図は**図71**で示されます。

図71. アナログ比較器部構成図



注: アナログ比較器のピン配置については、1頁の「ピン配置」と41頁の表29を参照してください。

■ アナログ比較器 制御/状態レジスタ (Analog Comparator Control and Status Register) ACSR

ビット	7	6	5	4	3	2	1	0	
\$08 (\$28)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	不定	0	0	0	0	0	

■ ビット7 – ACD : アナログ比較器禁止 (Analog Comparator Disable)

このビットが論理1を書かれると、アナログ比較器への電力がOFFにされます。このビットはアナログ比較器をOFFにするために何時でも設定(1)できます。これは活動動作や**アイドル動作**で電力消費を削減します。ACDビットを変更する時にACSRの**アナログ比較器割り込み許可(ACIE)ビット**を解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、このビットが変更される時に割り込みが起り得ます。

■ ビット6 – ACBG : 基準電圧選択 (Analog Comparator Bandgap Select)

このビットが設定(1)されると、内部基準電圧(公称1.23V)がアナログ比較器への非反転入力に取って代わります。このビットが解除(0)されると、AIN0がアナログ比較器の非反転入力に印加されます。**29頁の「内部基準電圧」**をご覧ください。

■ ビット5 – ACO : アナログ比較器出力 (Analog Comparator Output)

アナログ比較器の出力は同期化され、その後に直接ACOへ接続されます。この同期化は1~2クロック周期の遅延をもたらします。

■ ビット4 – ACI : アナログ比較器割り込み要求フラグ (Analog Comparator Interrupt Flag)

このビットは比較器出力での出来事がACSRの**アナログ比較器割り込み条件(ACIS1,0)ビット**によって定義した割り込み方法で起動する時に設定(1)されます。ACSRのアナログ比較器割り込み許可(ACIE)ビットが設定(1)され、**ステータスレジスタ(SREG)の全割り込み許可(I)ビット**が設定(1)されていると、アナログ比較器割り込みルーチンが実行されます。対応する割り込み処理ベクタを実行すると、ACIはハードウェアによって解除(0)されます。代わりにこのフラグへ**論理1**を書くことによってもACIは解除(0)されます。

■ ビット3 – ACIE : アナログ比較器割り込み許可 (Analog Comparator Interrupt Enable)

ACIEビットが**論理1**を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みが活性(有効)にされます。**論理0**を書かれると、この割り込みは禁止されます。

■ ビット2 – ACIC : アナログ比較器捕獲起動許可 (Analog Comparator Input Capture Enable)

論理1を書かれると、このビットはアナログ比較器によって起動されるタイマ/カウンタ1の捕獲機能を許可します。この場合、比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの**雑音消去機能**と**エッジ選択機能**を利用させる**捕獲入力前置論理回路**へ直接的に接続されます。**論理0**を書かれると、アナログ比較器と捕獲機能間の接続は存在しません。比較器がタイマ/カウンタ1捕獲割り込みを起動するには、**タイマ/カウンタ割り込み許可レジスタ(TIMSK)の捕獲割り込み許可(TICIE1)ビット**が設定(1)されなければなりません。

■ ビット1,0 – ACIS1,0 : アナログ比較器割り込み条件 (Analog Comparator Interrupt Mode Select)

これらのビットは比較器のどの事象がアナログ比較器割り込みを起動するのかを決めます。各種設定は**表72**で示されます。

ACIS1,ACIS0ビットを変更する時にACSRのアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、これらのビットが変更される時に割り込みが起き得ます。

表72. アナログ比較器割り込み条件選択

ACIS1	ACIS0	割り込み発生条件
0	0	比較器出力の変移(トグル)
0	1	(予約)
1	0	比較器出力の下降端
1	1	比較器出力の上昇端

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

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。これは異なる保護レベルを選択する独特な柔軟性を使用者に与えます。

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-モディファイライト)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(118頁の表89.参照)。このページ構成は通常動作に影響を及ぼしません。

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73.参照)。各領域の容量は115頁の表78.と図73.で示されるようにBOOTSZヒューズによって設定されます。これら2つの領域は個別の施錠ビットの組を持つため、異なる保護レベルを持てます。

応用領域

応用領域は応用コードを格納するのに使われるフラッシュメモリの領域です。応用領域用保護レベルは応用ブート施錠ビット(ブート施錠ビット0)によって選択できます(109頁の表74.参照)。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。

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レベルはブートローダ施錠ビット(ブート施錠ビット1)によって選択できます(109頁の表75.参照)。

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108頁の図73.と115頁の表79.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。

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むと、そのソフトウェアは未知の状態へ行き着くかもしれません。これを避けるために割り込みは禁止、またはブートローダ領域へ移動のどちらかにされるべきです。ブートローダ領域は常にNRWW領域に配置されます。RWW領域が読み出しに対して妨げられている限り、SPM命令制御レジスタ(SPMCR)のRWW領域多忙(RWWSB)ビットが論理1として読めます。プログラミングが完了した後、RWW領域に配置したコードを読む前にRWWSBはソフトウェアによって解除(0)されなければなりません。RWWSBを解除(0)する方法の詳細については110頁の「SPM命令制御レジスタ(SPMCR)」をご覧ください。

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73. 書き込み中読み出し可能機能

プログラミング中にZポイントで指定される領域	プログラミング中に読める領域	CPU動作	RWW機能支援
RWW領域	NRWW領域	通常動作	あり
NRWW領域	なし	停止	なし

図72. RWW領域とNRWW領域の関係

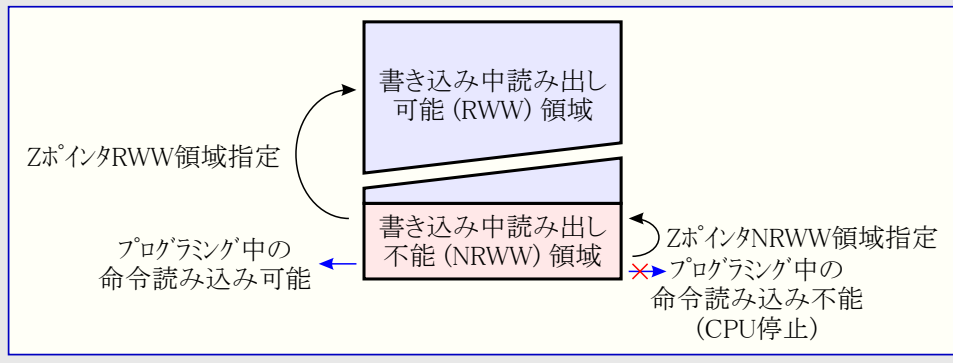
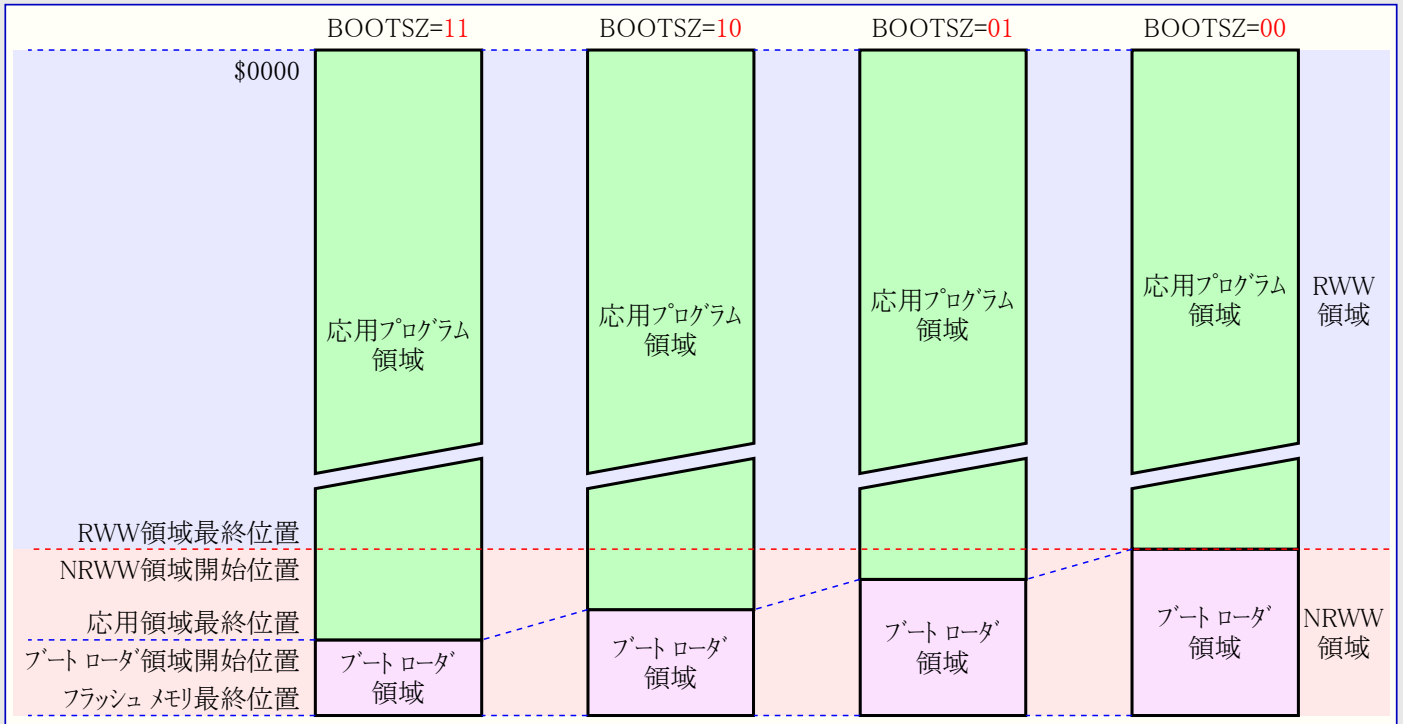


図73. 選択によるプログラム用フラッシュメモリの領域分割



注: 図内の各パラメータについては115頁の表78.で与えられます。

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。これは異なる保護レベルを選択する独特な柔軟性を使用者に与えます。

使用者は以下を選択できます。

- MCUによって更新するソフトウェアからフラッシュメモリ全体を保護
- MCUによって更新するソフトウェアからフラッシュメモリの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74と表75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74. 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75. 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。

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USARTやSPIインターフェース経由で受信した指令のような起点によって始められるかもしれません。代わりに、リセット後にリセットベクタがブートローダ領域開始アドレスを指示するようにブートリセット(BOOTRST)ヒューズをプログラム(0)できます。この場合、ブートローダがリセット後に開始されます。応用コードが設定された(書かれた)後、そのプログラム(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76. ブートリセットヒューズ (0=プログラム、1=非プログラム)

BOOTRST	リセット後実行開始アドレス (リセットベクタ)	
0	ブートローダリセット	ブートローダ開始アドレス(115頁の表78参照)
1	応用リセット	\$0000

■ SPM命令制御レジスタ (Store Program Memory Control Register) SPMCR

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	–	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	SPMCR
Read/Write	R/W	R	R	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

■ ビット7 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIEビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みはSPMCRのSPM操作許可(SPMEN)ビットが解除(0)されている限り実行されます。

■ ビット6 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)

RWW領域に自己プログラミング(ページ消去またはページ書き込み)操作が開始されると、RWWSBがハードウェアによって設定(1)されます。RWWSBビットが設定(1)されると、RWW領域はアクセスできません。自己プログラミングが完了された後にRWWSREビットが1を書かれると、このRWWSBビットは解除(0)されます。同様に、ページ設定操作が開始されると、RWWSBビットは自動的に解除(0)されます。

■ ビット5 – Res : 予約 (Reserved)

このビットは予約されており、常に0として読まれます。

■ ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域にプログラミング(ページ消去またはページ書き込み)すると、RWW領域は(RWWSBがハードウェアによって設定(1)され)読み出しに対して防がれます。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。

SPMCR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113頁の「ソフトウェアからのヒューズビットと施錠ビットの読み出し」をご覧ください。

■ ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の書き込み動作中にCPUは停止されます。

■ ビット1 – PGERS : ページ消去 (Page Erase)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の消去中にCPUは停止されます。

■ ビット0 – SPMEN : SPM操作許可 (Store Program Memory Enable)

このビットは次の4クロック周期間のSPM命令を許可します。このビットがRWWSRE、BLBSET、PGWRT、PGERSのどれかと共に1を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。SPMENだけが書かれると、続くSPM命令はZポインタによってアドレス指定したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SPMENビットはSPM命令の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ消去とページ書き込み中、SPMENビットはその動作が完了されるまで1に留まります。

下位5ビットに10001, 01001, 00101, 00011, 00001以外のどんな組み合わせを書いても無効です。

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

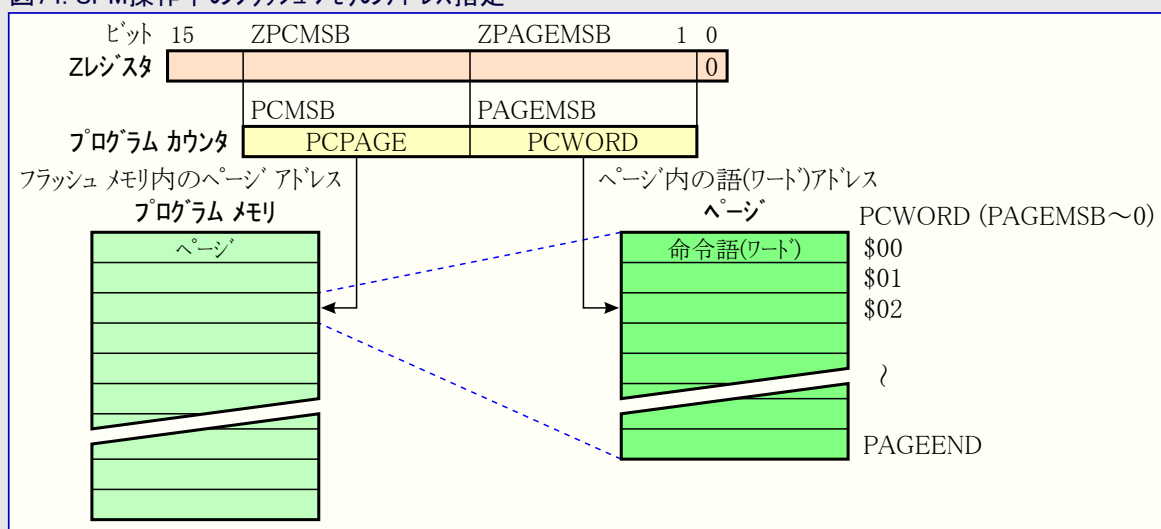
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

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(118頁の表89.参照)、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74.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74. 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115頁の表80.で一覧されます。

PCPAGEとPCWORDは118頁の表89.で一覧されます(訳注: 共通性のため本行追加)。

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使用者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114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。

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命令制御レジスタ(SPMCR)に'X0000011'を書き、SPMCR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0を書かれねばなりません。

- ・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。

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SPMCRに'00000001'を書き、SPMCR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SPMCR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。

注: 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。

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SPMCRに'X0000101'を書き、SPMCR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。

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、SPMCR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SPMCRをポーリングする代わりにこの割り込みが使えることを意味します。SPM操作可割り込みを使うとき、割り込みが読み出しに対して防がれる時にRWW領域をアクセスするのを避けるために、割り込みベクタはブートローダ領域(BLS)へ移動されるべきです。割り込み(ベクタ)の移動法は32頁の「割り込み」で記述されます。

ブートローダ領域更新中の考慮

ブート施錠ビット11(BLB11)が非プログラム(1)にされたままとすることによって使用者がブートローダ領域に更新を許す場合、特別な注意が祓われなければなりません。ブートローダ自身への予期せぬ書き込みはブートローダ全体を不正にし得て、更にソフトウェアの更新が不可能になるかもしれません。ブートローダ自体の変更が必要ないなら、内部ソフトウェアのどんな変更からもブートローダを保護するためにブート施錠ビット11(BLB11)をプログラム(0)することが推奨されます。

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。SPMCRのRWWSBはRWW領域が多忙である限り設定(1)されます。自己プログラミング中の割り込みベクタ表は32頁の「割り込み」で記述されるようにブートローダ領域(BLS)へ移動されるべきか、または割り込みが禁止されなければなりません。プログラミングが完了した後にRWW領域をアドレス指定する前に、使用者ソフトウェアはRWWSREの書き込みによってRWWSBを解除(0)しなければなりません。例については114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。

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SPMCRに'X0001001'を書き、SPMCR書き込み後4クロック周期内にSPM命令を実行してください。アクセス可能な施錠ビットはMCUによるどのソフトウェア更新からも応用とブートローダ領域を保護できるブート施錠ビットだけです。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	1	1

フラッシュメモリのアクセスに影響を及ぼすブートローダ施錠ビットの各種設定法については表74と表75をご覧ください。

R0のビット5～2が解除(0)される場合、SPMCR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。この操作中のZポインタは関係ありませんが、将来との共通性のために(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。

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てのソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(EECR)のEEPROM書き込み許可(EWE)ビットを検査し、SPM命令制御レジスタ(SPMCR)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。

ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビットの両方を読むことが可能です。施錠ビットを読むにはZポインタに\$0001を設定してSPMCRのSPMENとフ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SPMCR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場合、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SPMENとBLBSETビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	BLB12	BLB11	BLB02	BLB01	LB2	LB1

ヒューズ下位ビットを読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポインタに\$0000を設定してSPMCRのSPMENとBLBSETビットを設定(1)してください。SPMENとBLBSETビットがSPMCR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117頁の表84を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様に、ヒューズ上位ビットを読むにはZポインタに\$0003を設定してください。SPMENとBLBSETビットがSPMCR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117頁の表83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

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

フラッシュメモリデータ化けの防止

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

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

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

- そのシステムでブートローダ更新が必要ない場合、どんなブートローダソフトウェア更新をも防ぐためにブートローダ施錠ビットをプログラム(0)してください。
- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、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、SPMCR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。

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77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77. 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

アセンブリ言語による簡単なブート ロータ例

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 ロータ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 ロータ領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更になります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令	注釈
	. EQU PGSZB = PAGESIZE*2 . ORG SMALLBOOTSTART	; PGSZBはページ内のバイト数です。(PAGESIZEはワード数) ;
WRPG:	LDI SPMC, (1<<PGERS)+(1<<SPMEN) RCALL SPMJ	; [ページ消去] ; ページ消去SPMCR値を取得 ; ページ消去 ; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) RCALL SPMJ	; RWW領域読み出し許可SPMCR値を取得 ; 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]
WLP:	LDI CNTL, LOW (PGSZB) LDI CNTH, HIGH (PGSZB) LD R0, Y+ LD R1, Y+ LDI SPMC, (1<<SPMEN) RCALL SPMJ ADIW ZH:ZL, 2 SBIW CNTH:CNTH, 2 BRNE WLP	; バイト計数器を初期化 ; (削除) ; RAM上の下位データを取得(ポインタ進行) ; RAM上の上位データを取得(ポインタ進行) ; ページ一時緩衝部書き込みSPMCR値を取得 ; 対応語(ワード)データをページ一時緩衝部に設定 ; ページ一時緩衝部ポインタ進行 ; 計数器を減数 (SUBI) ; 指定バイト数分継続 ; [ページ書き込み] ; ページ一時緩衝部先頭にポインタを復帰 ; (削除)
	SUBI ZL, LOW (PGSZB) SBCI ZH, HIGH (PGSZB) LDI SPMC, (1<<PGWRT)+(1<<SPMEN) RCALL SPMJ	; フラッシュ書き込みSPMCR値を取得 ; フラッシュ メモリ ページ書き込み ; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) RCALL SPMJ	; RWW領域読み出し許可SPMCR値を取得 ; RWW領域読み出し許可 ; [読み戻し照合 (任意)]
	LDI CNTL, LOW (PGSZB) LDI CNTH, HIGH (PGSZB) SUBI YL, LOW (PGSZB) SBCI YH, HIGH (PGSZB)	; バイト計数器を初期化 ; (削除) ; RAMデータ先頭にポインタを復帰
RLP:	LPM R0, Z+ LD R1, Y+ CPSE R0, R1 RJMP ERROR	; フラッシュ メモリから1バイト取得(ポインタ進行) ; RAMから1バイト データを取得(ポインタ進行) ; 値一致でスキップ ; 不一致で異常処理へ
;		
	SBIW CNTH:CNTH, 1 BRNE RLP	; 計数器を減数 (SUBI) ; 指定バイト数分継続 ; [RWW領域へ復帰]
RTN:	IN TMP, SPMCR SBRS TMP, RWWSB RET	; SPM命令制御レジスタ値を取得 ; RWW領域多忙でスキップ ; 準備可で呼び出し元へ復帰 ; [RWW領域読み出し再許可]
;		
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN) RCALL SPMJ RJMP RTN	; RWW領域読み出し許可SPMCR値を取得 ; RWW領域読み出し許可 ; RWW領域準備可まで待機へ ; [SPM命令実行サブルーチン]
SPMJ:	IN TMP, SPMCR SBRC TMP, SPMEN RJMP SPMJ	; SPM命令制御レジスタ値を取得 ; 操作可能(直前のSPM完了)でスキップ ; 操作可まで待機
;		
	IN TMP, SREG CLI SBIC EECR, EEWAIT RJMP WAIT	; ステータスレジスタ値を保存 ; 全割り込み禁止 ; EEPROM書き込み中以外でスキップ ; EEPROM書き込み完了まで待機
WAIT:		
;		
	OUT SPMC, SPMC SPM OUT SREG, TMP RET	; SPM動作指定 ; 対応SPM動作実行 ; ステータスレジスタ値を復帰 ; 呼び出し元へ復帰

ATmega8515用ブートローダのパラメータ

自己プログラミングの記述で使ったパラメータは表78.～80.で与えられます。

表78. 応用領域とブートローダ領域の分割設定

BOOTSZ1	BOOTSZ0	ブートローダ領域		アドレス範囲		
		容量(ワード)	ページ数	応用プログラム領域	ブートローダプログラム領域	ブートリセットアドレス
1	1	128	4	\$0000～\$0F7F	\$0F80～\$0FFF	\$0F80
1	0	256	8	\$0000～\$0EFF	\$0F00～\$0FFF	\$0F00
0	1	512	16	\$0000～\$0DFF	\$0E00～\$0FFF	\$0E00
0	0	1024	32	\$0000～\$0BFF	\$0C00～\$0FFF	\$0C00

注: 各種BOOTSZヒューズ設定は図73.で示されます。

表79. RWW領域とNRWW領域の範囲

領域	ページ数	アドレス範囲
書き込み中読み出し可能(RWW)領域	96	\$0000～\$0BFF
書き込み中読み出し不能(NRWW)領域	32	\$0C00～\$0FFF

注: これら2つの領域についての詳細に関しては、108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80. 図74.で使った各変数説明とZポイントの配置

変数名	対応値		意味
	PC	Zポイント(注)	
PCMSB	PC11		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)
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	Z12～6	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	Z5～1	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: 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命令に対するバイト選択です。

自己プログラミング中のZポイントの使用についての詳細に関しては111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。

メモリプログラミング

プログラムメモリとデータメモリ用施錠ビット

ATmega8515は非プログラム(1)のままか、表82.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81. 施錠ビットバイトの内容

名称	ビット番号	意味	既定値 (注)
—	7		1 (非プログラム)
—	6		1 (非プログラム)
BLB12	5	ブートローダ領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB11	4		1 (非プログラム)
BLB02	3	応用領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB01	2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表82. 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	直列または並列プログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズビットとブート施錠ビットが固定されます。(注1)
LB0 種別	BLB02	BLB01	フラッシュメモリの応用プログラム領域に対する保護
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
4	0	1	ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
LB1 種別	BLB12	BLB11	フラッシュメモリのブートローダプログラム領域に対する保護
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)
4	0	1	応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)

注: 0はプログラム、1は非プログラムを意味します。

注1: 施錠ビットを書く前にヒューズビットとブート施錠ビットを書いてください。

注2: BLB02=0で、割り込みベクタがブートローダ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

注3: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

ヒューズ ビット

ATmega8515には2つのヒューズ バイトがあります。表83.と表84.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83. ヒューズ上位バイト一覧

名称	ビット	意味	既定値
S8515C (注1)	7	AT90S4414/8515互換動作。	1 (非プログラム) 非互換動作
WDTON	6	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
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
CKOPT (注3)	4	クロック発振器任意選択機能。	1 (非プログラム)
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 ロード容量選択。(表78.参照) (注4)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOOTRST	0	リセット ベクタ(応用領域またはブート ロード領域)選択。	1 (非プログラム) 応用領域

注1: 詳細については3頁の「AT90S4414/8515互換動作」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: CKOPTヒューズの機能はCKSELヒューズ設定に依存します。詳細は20頁の「クロック元」をご覧ください。

注4: BOOTSZ1,0既定値は最大ブート ロード容量になります。115頁の表78.をご覧ください。

表84. ヒューズ下位バイト一覧

名称	ビット	意味	既定値
BODLEVEL	7	低電圧検出器(BOD)検出電圧選択。	1 (非プログラム)
BODEN	6	低電圧検出器(BOD)許可。	1 (非プログラム) BOD禁止
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		0 (プログラム)
CKSEL0	0		1 (非プログラム)

注1: SUT1,0の既定値は最大起動時間になります。詳細については22頁の表13.をご覧ください。

注2: CKSEL3~0の既定設定は1MHz校正付き内蔵RC発振器になります。詳細については20頁の表5.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。

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。

識票バイト

全てのAtmelマイクロ コントローラはデバイスを識別する3バイトの識票符号を持ちます。この符号は直列と並列の両プログラミング動作で、またデバイスが施錠されていても読めます。この3バイトは分離された空間に存在します。ATmega8515の識票バイトを右に示します。

- ① \$000 : \$1E 製造業者Atmelを示します。
- ② \$001 : \$93 フラッシュ メモリ容量8Kバイトを示します。
- ③ \$002 : \$06 ②値\$93と合せ、ATmega8515を示します。

校正バイト

ATmega8515は内蔵RC発振器用に4つの異なる校正値を備えています。これらのバイトは1,2,4,8MHzの各々に対して識票列のアドレス\$000,\$001,\$002,\$003の上位バイトにあります。リセット中、1MHz値がOSCCALレジスタへ自動的に設定されます。他の周波数が使われる場合、その校正値は手動で設定されなければなりません。詳細については23頁の「発振校正レジスタ(OSCCAL)」をご覧ください。

並列プログラミング

この項はATmega8515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。

信号名

本項でATmega8515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75**と**表85**をご覧ください。後続の表で示されないピンはピン名によって参照されます。

XA0とXA1ピンはXTAL1ピンが正パルスを与えられる時に実行される動作を決めます。このビット符号化は**表87**で示されます。

WRまたはOEパルス送付時、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88**で示されます。

図75. 並列プログラミング構成図

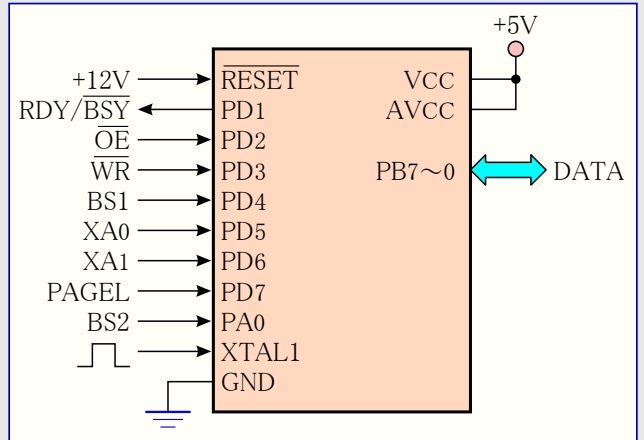


表85. 信号名とピン名の関係

信号名	ピン名	入出力	機能
RDY/BSY	PD1	出力	0(Low) : 多忙(プログラミング中) 1(High) : 準備可(指令受付可)
OE	PD2	入力	出力許可(負論理)
WR	PD3	入力	書き込みパルス(負論理)
BS1	PD4	入力	上位/下位バイト選択1 (0:下位, 1:上位) (一般用)
XA0	PD5	入力	XTAL動作ビット0
XA1	PD6	入力	XTAL動作ビット1
PAGEL	PD7	入力	プログラムメモリとEEPROMデータをページ一時緩衝部に設定
BS2	PA0	入力	上位/下位バイト選択2 (0:下位, 1:上位) (ヒューズビット用)
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

表86. プログラミング動作移行時のピン値

ピン名	シンボル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

表87. 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88. 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)	識別バイト、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89. フラッシュメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
4Kワード 8Kバイト	32ワード	PC4~0	128	PC11~5	11

表90. EEPROMメモリのページ数とページの語数

全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
512バイト	4バイト	EEA1~0	128	EEA8~2	8

並列プログラミング手順

並列プログラミング動作への移行

次に示す方法がデバイスを並列プログラミング動作にします。

- ① VCCとGND間に4.5～5.5Vを印加し、最低100μs待ちます。
- ② RESETをLow(0)にし、XTAL1を少なくとも6回、交互に(High/Low)切り替えます。
- ③ 118頁の表86.で一覧されるProg_enableピンを全てLow(0)に設定し、最低100ns待ちます。
- ④ RESETに11.5～12.5Vを印加します。+12VがRESETに印加されてしまった後100ns以内のProg_enableピンのどんな動きも、デバイスのプログラミング動作への移行を失敗させます。

注意: 外部クリスタルや外部RC発振器設定が選択されていると、条件付きのXTAL1パルスを印加できないため、同じことが当てはまるかもしれません。このような場合、次の方法に従うべきです。

- ① 118頁の表86.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を印加すると同時にRESETへ11.5～12.5Vを印加します。
- ③ 100ns待ちます。
- ④ クロック元として外部クロックが選択される(CKSEL3～0=0000)ことを保証するためにヒューズを再プログラミング(再書き込み)します。施錠ビットがプログラム(0)されている場合、ヒューズを変更するのに先立ってチップ消去指令が実行されなければなりません。
- ⑤ デバイスの電源を落とすか、RESETピンをLow(0)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。
- ⑥ 上で記述したように元の方法でプログラミング動作へ移行します。

効率的なプログラミングへの考慮

設定した指令とアドレスはプログラミング中、維持されます。効率的なプログラミングを行うために次を考慮すべきです。

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

チップ消去

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

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

「チップ消去」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$80(1000 0000)に設定します。これはチップ消去指令です。
- ④ XTAL1に正パルスを与えます。これはチップ消去指令を設定します。
- ⑤ WRに負パルスを与えます。これはチップ消去を開始します。RDY/BSYがLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/BSYがHigh(1)になるまで待ちます。

フラッシュメモリ書き込み (図77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(118頁の表89参照)。フラッシュメモリへ書く時にプログラムデータはページ緩衝部にラッチされます。これは同時に書かれることをプログラムデータの1ページに許します。次の手順は完全なフラッシュメモリの書き込み方法を記述します。

A. 「フラッシュメモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュメモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュメモリ書き込み指令を設定します。

B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選択します。
- ③ DATAにアドレス下位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス下位バイトを設定します。

C. データ下位バイト設定

- ① XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ② DATAにデータ下位バイト(\$00～\$FF)を設定します。
- ③ XTAL1に正パルスを与えます。これはデータ下位バイトを設定します。

D. データ上位バイト設定

- ① BS1をHigh(1)に設定します。これは上位バイトを選択します。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはデータ上位バイトを設定します。

E. 語(ワード)データをページ一時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイトを選択します。
- ② PAGELに正パルスを与えます。これは語データをページ一時緩衝部にラッチ(設定)します。

F. 緩衝部全体が満たされるか、または(必要な)全てのデータが緩衝部内に設定されるまで、B～Eを繰り返す

アドレス内の下位ビットがページ内の語(ワード)位置を指示する一方、上位ビットがフラッシュメモリ内のページをアドレス指定します。これは図76で図示されます。ページ内の語アドレスに8ビット未満が必要とされる場合(ページ容量<256)、アドレス下位バイトの最上位(側)ビットがページ書き込み実行時のページアドレスに使われることに注意してください。

G. アドレス上位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選択します。
- ③ DATAにアドレス上位バイト(\$00～\$0F)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイトを設定します。

H. ページ書き込み

- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/BSYがLow(0)になります。
- ③ RDY/BSYがHigh(1)になるまで待ちます。

I. フラッシュメモリ全部が書かれるか、または(必要な)全データが書かれてしまうまで、B～Hを繰り返す

J. ページ書き込み終了

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図76. ページで構成されたフラッシュメモリのアドレス指定

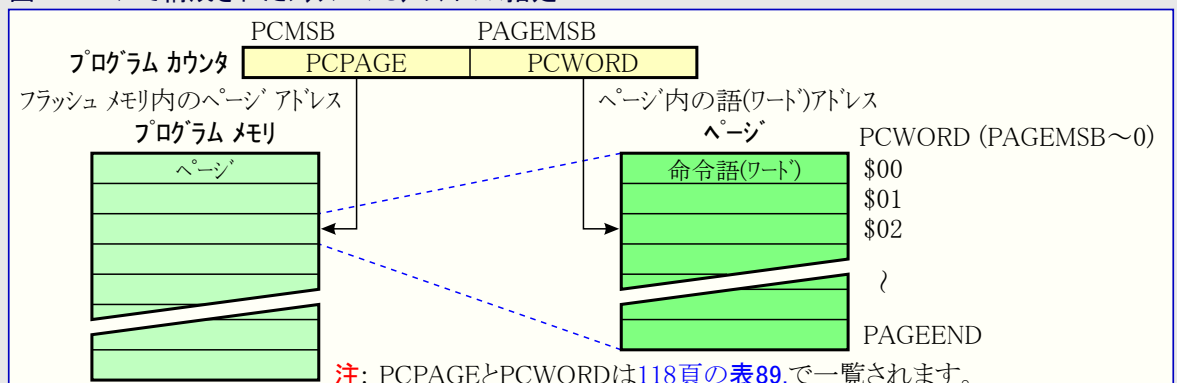
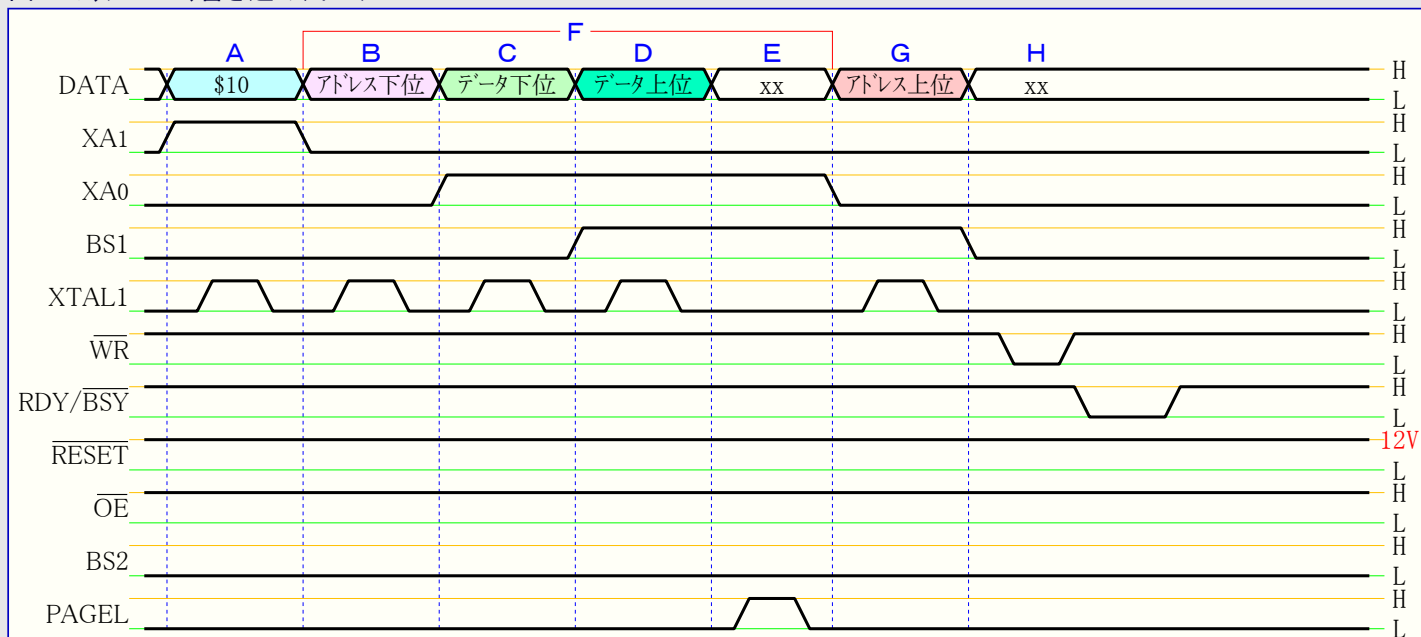


図77. フラッシュメモリ書き込みタイミング



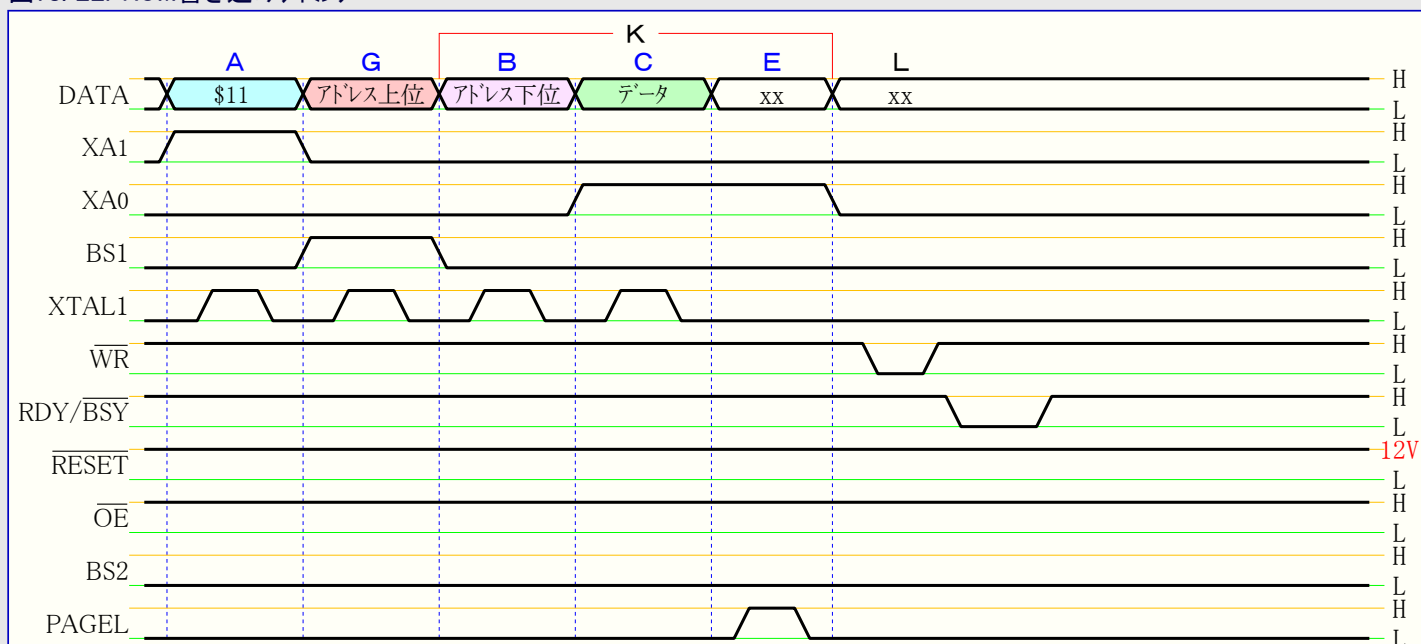
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

EEPROM書き込み

EEPROMはページで構成されます(118頁の表90参照)。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120頁の「フラッシュメモリの書き込み」を参照。図78. 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78. EEPROM書き込みタイミング



フラッシュメモリ読み出し

フラッシュメモリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については120頁の「フラッシュメモリの書き込み」を参照)

1. フラッシュメモリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$0F)を設定します。(「フラッシュメモリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。フラッシュメモリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュメモリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. OEをHigh(1)に設定します。DATAはHi-Zになります。

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120頁の「フラッシュメモリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$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. 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。

ヒューズ下位ビット書き込み

ヒューズ下位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120頁の「フラッシュメモリの書き込み」を参照)

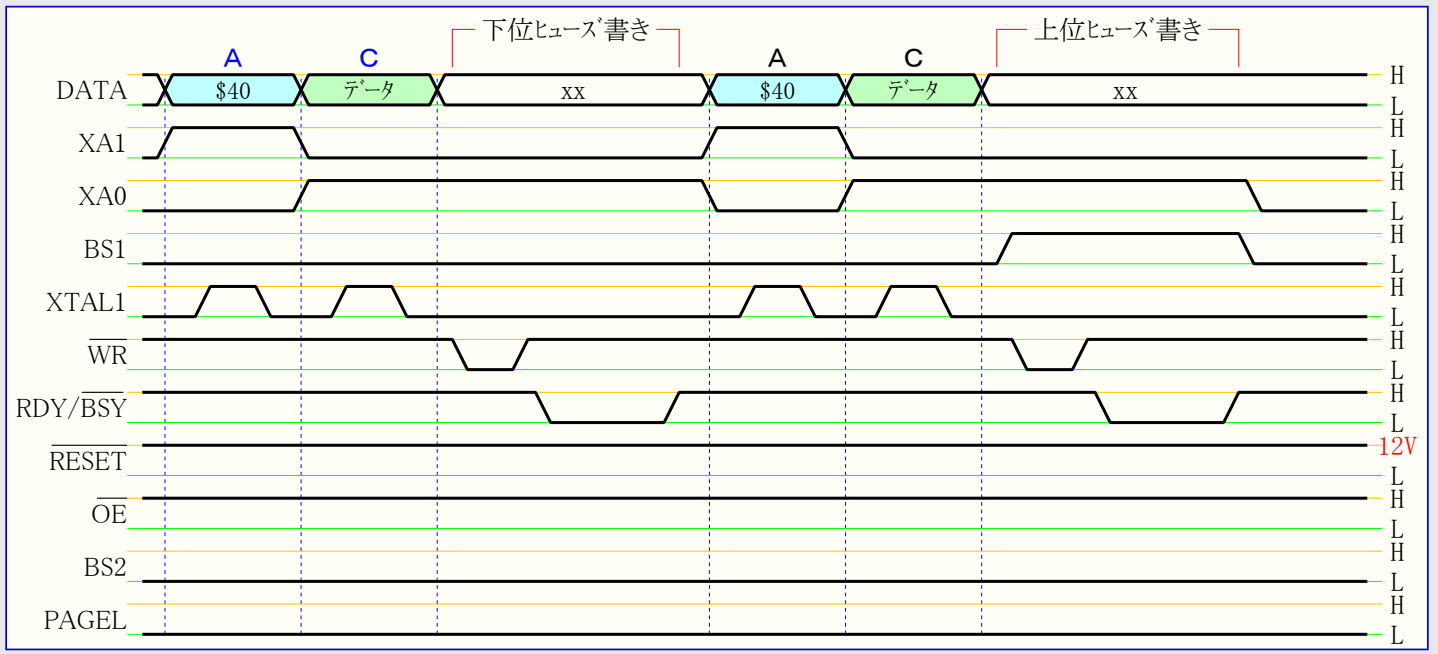
1. ヒューズビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュメモリ書き込み」のCを参照)
3. BS1とBS2をLow(0)に設定します。
4. WRに負パルスを与え、RDY/BSYがHighになるまで待ちます。

ヒューズ上位ビット書き込み

ヒューズ上位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120頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. データ下位バイト(としてデータ)を設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュメモリ書き込み」のCを参照)
3. BS1をHigh(1)にし、BS2をLow(0)に設定します。
4. WRに負パルスを与え、RDY/BSYがHighになるまで待ちます。
5. BS1をLow(0)に設定します。これはデータ下位バイトを選択します。

図79. ヒューズ書き込みタイミング



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120頁の「フラッシュメモリの書き込み」を参照)

1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュメモリ書き込み」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。(「フラッシュメモリ書き込み」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。

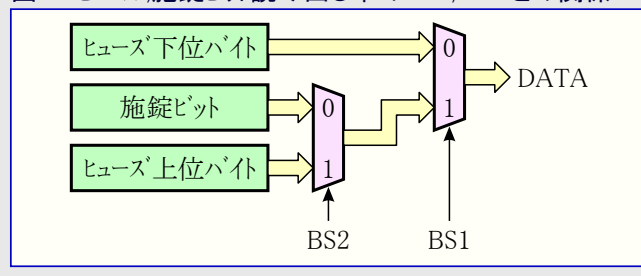
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

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120頁の「フラッシュメモリの書き込み」を参照)

1. ヒューズビットと施錠ビットの読み出し指令\$04(0000 0100)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. BS1とBS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ下位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
3. BS1とBS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ上位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
4. BS1をhigh(1)、BS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。施錠ビットの状態が直ぐにDATAで読めます。(0=プログラム)
5. $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

図80. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



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120頁の「フラッシュメモリの書き込み」を参照)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリ書き込み」のBを参照)
3. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。選択した識票バイトが直ぐにDATAで読めます。
4. $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

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120頁の「フラッシュメモリの書き込み」を参照)

1. 校正バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス下位バイトに(\$00~\$03)を設定します。(「フラッシュメモリ書き込み」のBを参照) (訳注:本行は整合性のため修正しました)
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。

並列プログラミング特性

図81. 並列プログラミング タイミング (一般的な必要条件)

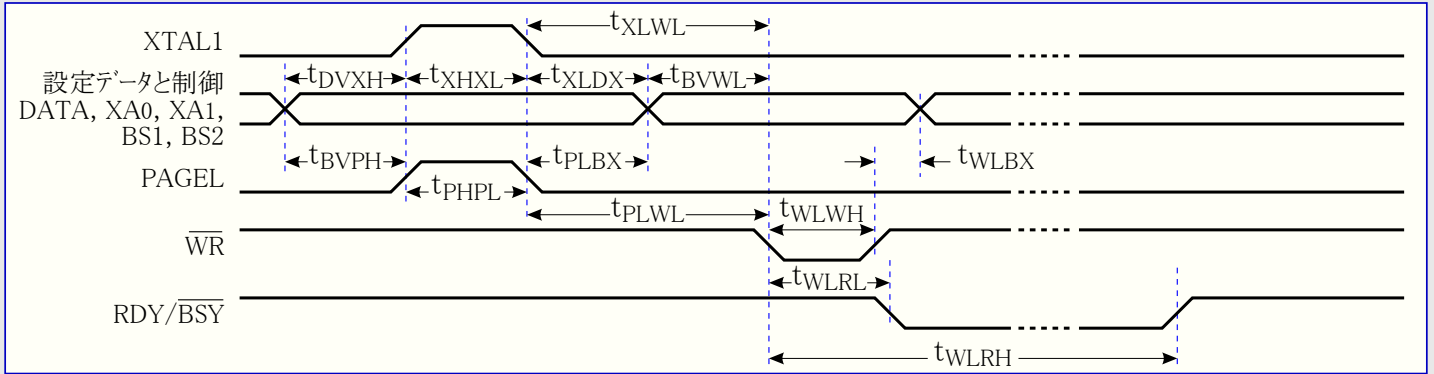
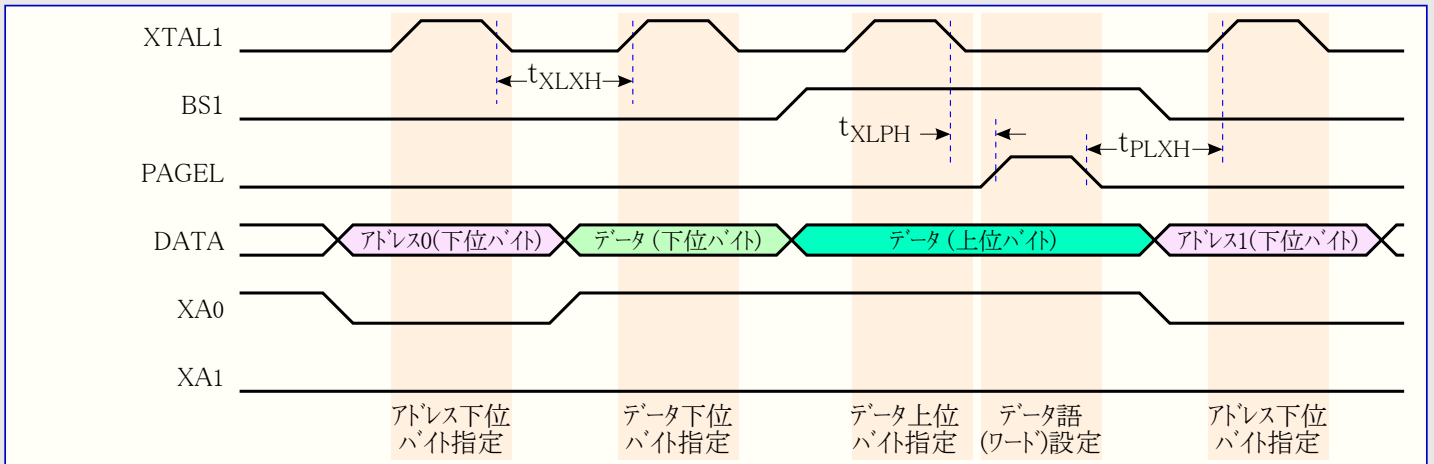
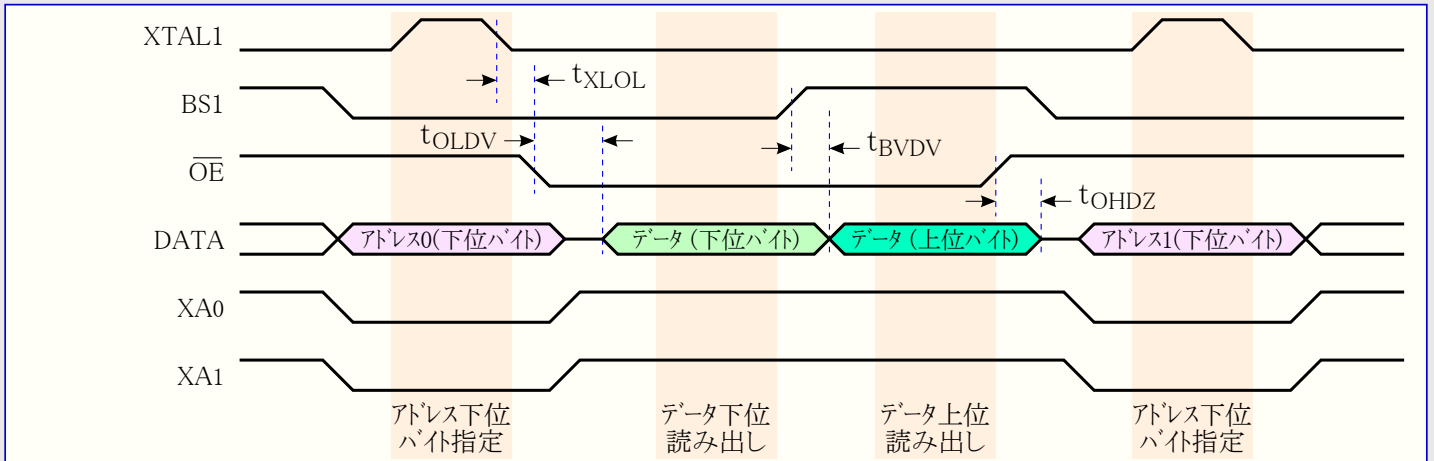


図82. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図81.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHLX} 、 t_{XLDX})は設定操作にも適用されます。

図83. 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81.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHLX} 、 t_{XLDX})は読み出し操作にも適用されます。

表91. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
VPP	プログラミング許可電圧	11.5		12.5	V
IPP	プログラミング許可電流			250	μA
tDVXH	XTAL1 ↑ に対するデータと制御の準備時間	67			ns
tXLXH	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
tXHXL	XTAL1 Highパルス幅	150			
tXLDX	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
tXLWL	XTAL1パルス ↓ 後のWR ↓ 待機時間	0			
tXLPH	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
tPLXH	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			
tBVPH	PAGELパルス ↑ に対するBS1準備時間	67			
tPHPL	PAGEL Highパルス幅	150			
tPLBX	PAGELパルス ↓ 後のBS1保持時間	67			
tWLBX	WRパルス ↓ 後のBS1,BS2保持時間	67			
tPLWL	PAGELパルス ↓ 後のWRパルス ↓ 待機時間	67			
tBVWL	WRパルス ↓ に対するBS1準備時間	67			
tWLWH	WR Lowパルス幅	150			
tWLRL	WRパルス ↓ 後のRDY/BSY ↓ 遅延時間	0		1	μs
tWLRH	書き込み時間 (WR ↓ からRDY/BSY ↑) (注1)	3.7		4.5	ms
tWLRH_CE	チップ消去時間 (WR ↓ からRDY/BSY ↑) (注2)	7.5		9	
tXLCL	XTAL1パルス ↓ 後のOE ↓ 待機時間	0			ns
tBVDV	BS1有効からのDATA遅延時間	0		250	
tOLDV	OE ↓ 後のDATA出力遅延時間			250	
tOHDZ	OE ↑ 後のDATA Hi-Z遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され得るのに先立って**プログラミング許可命令**が初めに実行されることを必要とします。**注意**、**表92**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。

直列プログラミング用ピン配置

表92. 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	PB5	入力	直列データ入力
MISO	PB6	出力	直列データ出力
SCK	PB7	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。

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

直列プログラミング手順

ATmega8515に直列データを書くとき、データはSCKの上昇端で行われ、ATmega8815から読むとき、データはSCKの下降端で行われます。タイミングの詳細については**図85**をご覧ください。

直列プログラミング動作でのATmega8515のプログラミングと照合は次の手順が推奨されます(4バイト命令形式は**表94**をご覧ください)。

1. 電源投入手順:

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

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

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

4. フラッシュメモリは1ページ単位で書かれます。ページ容量は**118頁の表89**で得られます。このメモリページは**ページ設定命令**と共にアドレスの下位5+1ビットとデータを提供することによって1バイトずつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。(プログラムメモリ)のページはアドレスの上位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。**ホーリング**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93**参照)待たなければなりません。

注: 何れの(フラッシュ、EEPROM、施錠ビット、ヒューズ)書き込み操作が完了される前にホーリング(読み)以外の命令が加えられると、不正な書き込みに終わるかもしれません(**訳注:**共通性のため原書の該当行に代わり本行追加)。

5. EEPROMは適切な**EEPROM書き込み命令**と共にアドレスとデータを提供することによって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。**ホーリング**が使われない場合、次のバイトを行う前に最低tWD_EEPROM(**表93**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択したアドレスの内容を直列出力MISOに読み戻す**読み出し命令**を使うことによって照合できます。

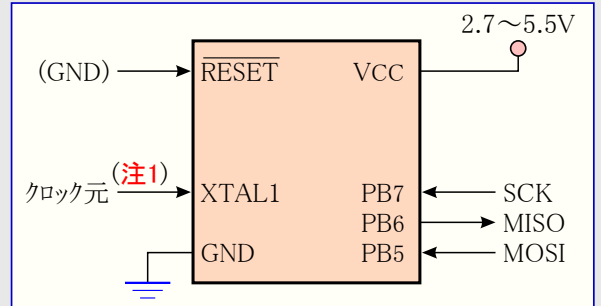
7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。

8. 電源OFF手順 (必要とされるならば)

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

(**訳注**) 原書では「直列プログラミング」項と「直列プログラミング用ピン配置」項先頭部の内容が重複しています。(削除忘れ)

図84. 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。

フラッシュメモリのデータポーリング

フラッシュメモリにページが書かれつつある時の書かれているページ内のアドレス位置読み出しは値\$FFを与えます。デバイスが新規ページに対する準備ができると、書いた値が正しく読めます。これは次ページが書ける時を決めるのに使われます。ページ全体が同時に書かれ、ページ内のどのアドレスでもポーリングに使えることに留意してください。フラッシュメモリのデータポーリングは値\$FFに対して行えないので、この値を書く時に使用者は次ページを書く前に最低tWD_FLASH待たなければなりません。全ての場所が\$FFのチップ消去されたデバイスは\$FFデータのアドレスの書き込みを飛ばせます。tWD_FLASH値については表93をご覧ください。

EEPROMのデータポーリング

新規バイト(書き込み命令)が書かれてEEPROMに書かれつつある時の書かれているアドレス位置読み出しは値\$FFを与えます。デバイスが新規バイトに対する準備ができると、書いた値が正しく読めます。これは次バイトが書ける時を決めるのに使われます。これは値\$FFに対して行えませんが、使用者は次のことを覚えておくべきです。全ての場所が\$FFのチップ消去されたデバイスは\$FFデータのアドレスの書き込みを飛ばせます。これはデバイスをチップ消去せずにEEPROMが書き換えられる場合には適用しません。この場合、データポーリングは値\$FFに対して使えず、次バイト書き込み前に最低tWD_EEPROM待たなければなりません。tWD_EEPROM値については表93をご覧ください。

表93. 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	9.0ms	EEPROM書き込み
tWD_ERASE	9.0ms	チップ消去

図85. 直列プログラミングバイト通信波形

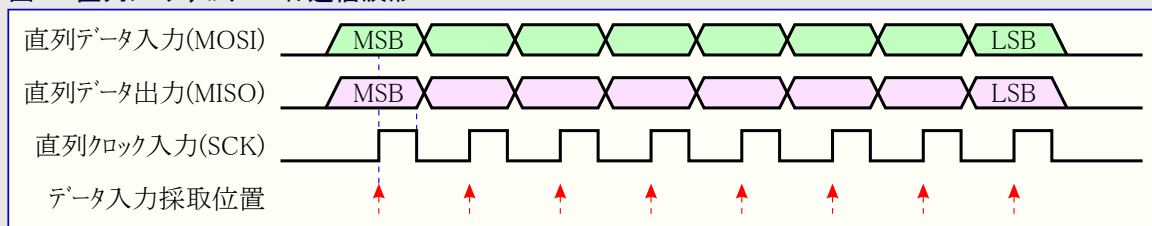


表94. 直列プログラミング命令一式

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H:LのP(H/L)バイトを読み出します。
ページ一時緩衝部設定	0100 P000	0000 xxxx	xxxL LLLL	WWWW WWWW	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	LLxL 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	00xx xxxH	LLLL LLLL	RRRR RRRR	アドレスH:Lのバイトを読み出します。
EEPROM書き込み	1100 0000	00xx xxxH	LLLL LLLL	WWWW WWWW	アドレスH:Lのバイトに書き込みます。
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 WWWW	(116頁の表81.参照)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	WWWW WWWW	(117頁の表84.参照)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	WWWW WWWW	(117頁の表83.参照)に書き込みます。
識別バイト読み出し	0011 0000	00xx xxxx	xxxx xxLL	RRRR RRRR	アドレスLの識別バイトを読み出します。
校正バイト読み出し	0011 1000	00xx xxxx	0000 00LL	RRRR RRRR	アドレスLの校正バイトを読み出します。

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)

(訳注) 校正値読み出しの第3バイトのLLは原書では00ですが、仕様からLLと予測されます(未確認)。

直列プログラミング特性

SPI部の特性については130頁の「SPIタイミング特性」をご覧ください。

電気的特性

絶対最大定格 (警告)

動作温度	-55℃ ~ +125℃
保存温度	-65℃ ~ +150℃
RESETを除くピン許容電圧	-0.5V ~ VCC+0.5V
RESETピン許容電圧	-0.5V ~ +13.0V
最大動作電圧	6.0V
入出力ピン出力電流	40.0mA
消費電流	200.0mA

(警告)

絶対最大定格を超える負担はデバイスに定常的な損傷を与えます。絶対最大定格は負担の定格を示すためだけのもので、この値または、この仕様書の動作特性で示された値を超える条件で動作することを示すものではありません。長時間の最大定格での使用はデバイスの信頼性を損なう場合があります。

DC特性

TA=-40℃~85℃, 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 _{IL}	Lowレベル入力電圧	XTAL1, RESETを除く	-0.5		0.2VCC (注1)	V
V _{IL1}	Lowレベル入力電圧	XTAL1	-0.5		0.1VCC (注1)	
V _{IL2}	Lowレベル入力電圧	RESET	-0.5		0.2VCC (注1)	
V _{IH}	Highレベル入力電圧	XTAL1, RESETを除く	0.6VCC (注2)		VCC+0.5	
V _{IH1}	Highレベル入力電圧	XTAL1	0.8VCC (注2)		VCC+0.5	
V _{IH2}	Highレベル入力電圧	RESET	0.9VCC (注2)		VCC+0.5	
V _{OL}	Lレベル出力電圧 (ポートA,B,C,D) (注3)	IOL=20mA, VCC=5V			0.7	
		IOL=10mA, VCC=3V			0.5	
V _{OH}	Hレベル出力電圧 (ポートA,B,C,D) (注4)	IOH=-20mA, VCC=5V	4.2			μA
		IOH=-10mA, VCC=3V	2.2			
I _{IL}	I/OピンLowレベル入力漏れ電流	VCC=5.5V (絶対値)			1	μA
I _{IH}	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			1	
R _{RST}	RESETピンプルアップ抵抗		30		60	kΩ
R _{PU}	I/Oピンプルアップ抵抗		20		50	
I _{CC}	活動動作消費電流	ATmega8515L VCC=3V, 4MHz			4	mA
		ATmega8515 VCC=5V, 8MHz			12	
	アイドル動作消費電流	ATmega8515L VCC=3V, 4MHz			1.5	
		ATmega8515 VCC=5V, 8MHz			5.5	
	パワーダウン動作消費電流 (注5)	VCC=3V, WDT許可			<13	μA
		VCC=3V, WDT禁止			<2	
V _{ACIO}	アナログ比較器入力オフセット電圧	VCC=5V, Vin=VCC/2			40	mV
I _{ACLK}	アナログ比較器入力漏れ電流		-50		50	nA
t _{ACPD}	アナログ比較器伝播遅延時間	VCC=2.7V		750		ns
		VCC=4.0V		500		

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)~(注5)は次頁を参照してください。

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=5Vで20mA、VCC=3Vで10mA)よりも多くの吸い込み電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全ポートのIOLの合計が200mAを超えるべきではありません。
2. ポートB0～B7、D0～D7とXTAL2のIOLの合計が100mAを超えるべきではありません。
3. ポートA0～A7、C0～C7、E0～E2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=5Vで20mA、VCC=3Vで10mA)よりも多くの吐き出し電流を流すことができますが、次の条件を厳守しなければなりません。

1. 全ポートのIOHの合計が200mAを超えるべきではありません。
2. ポートB0～B7、D0～D7とXTAL2のIOHの合計が100mAを超えるべきではありません。
3. ポートA0～A7、C0～C7、E0～E2のIOHの合計が100mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件よりも大きな吐き出し電流を流すことは保証されません。

注5: パワーダウン動作に対する最小電源電圧(VCC)は2.5Vです。

外部クロック特性

図86. 外部クロック駆動波形

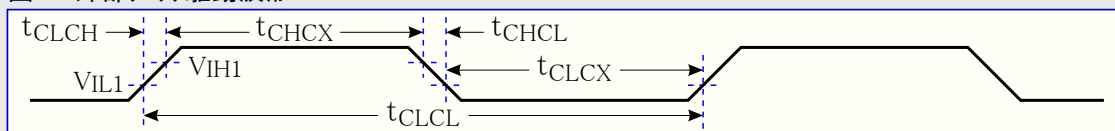


表95. 外部クロック特性

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/tCLCL	クロック周波数	0	8	0	16	MHz
tCLCL	クロック周期	125		62.5		ns
tCHCX	Highレベル時間	50		25		
tCLCX	Lowレベル時間	50		25		
tCLCH	上昇時間		1.6		0.5	μs
tCHCL	下降時間		1.6		0.5	
ΔtCLCL	隣接クロック周期間の変化率		2		2	%

注: 詳細については23頁の「外部クロック信号」を参照してください。

表96. 代表的な外部RC発振周波数

周波数 f	抵抗 R (kΩ)	容量 C (pF)
650kHz	33	22
2.0MHz	10	22

注: Rの範囲は3～100kΩ、Cは最小20pFとすべきです。表で与えられたCの値はピン容量を含みます。これは外圍器形状や基板配置で変化します。

SPIタイミング特性

図87. SPI タイミング必要条件 (主装置動作)

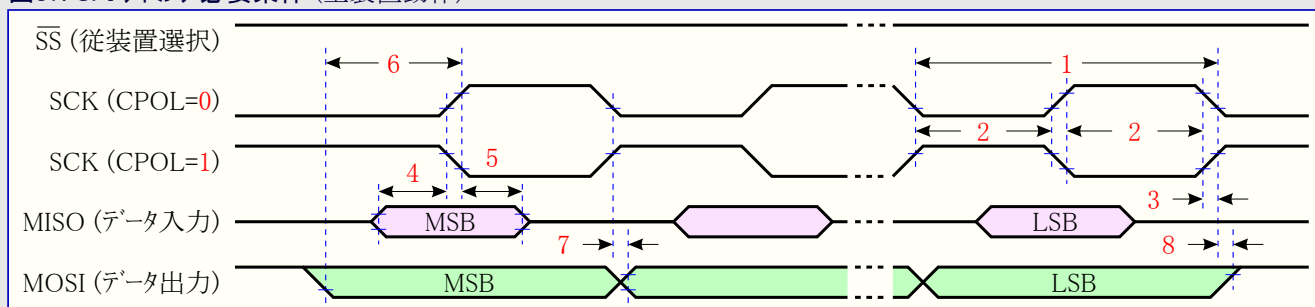


図88. SPI タイミング必要条件 (従装置動作)

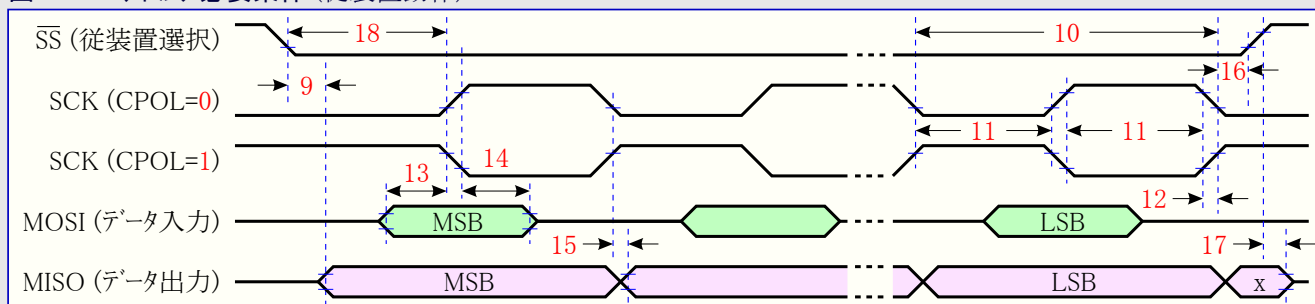


表97. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表58.参照		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	SS ↓からの出力遅延時間	従装置		15		
10	SCK周期	従装置	4t _{CK}			μs
11	SCK High/Low期間 (注)	従装置	2t _{CK}			
12	SCK上昇/下降時間	従装置			1.6	
13	入力データ 準備時間	従装置	10			
14	入力データ 保持時間	従装置	10			
15	SCKからの出力遅延時間	従装置		15		
16	SCKからのSS ↑ 遅延時間	従装置	20			
17	SS ↑からの出力Hi-Z遅延時間	従装置		10		
18	SS ↓からのSCK遅延時間	従装置	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} < 12MHz)、3t_{CLCL}(t_{CK} ≥ 12MHz)です。

外部メモリ タイミング特性

表98. 外部データメモリ特性 (VCC=4.5V~5.5V、待ちなし)

	シンボル	項目	8MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	16		MHz
1	tLHLL	ALE Highパルス幅	115		1.0tCLCL-10			ns
2	tAVLL	ALE ↓ 前 下位アドレス準備時間	57.5		0.5tCLCL-5		1	
3A	tLLAX,ST	ライト時 ALE ↓ 後 下位アドレス保持時間	5		5			
3B	tLLAX,LD	リード時 ALE ↓ 後 下位アドレス保持時間	5		5			
4	tAVLLC	ALE ↓ 前 上位アドレス準備時間	57.5		0.5tCLCL-5		1	
5	tAVRL	リード時 RD ↓ 前 下位アドレス有効時間	115		1.0tCLCL-10			
6	tAVWL	ライト時 WR ↓ 前 下位アドレス有効時間	115		1.0tCLCL-10			
7	tLLWL	ALE ↓ 後 WR ↓ 遅延時間	47.5	67.5	0.5tCLCL-15	0.5tCLCL+5	2	
8	tLLRL	ALE ↓ 後 RD ↓ 遅延時間	47.5	67.5	0.5tCLCL-15	0.5tCLCL+5	2	
9	tDVRH	RD ↑ 前 データ準備時間	40		40			
10	tRLDV	RD ↓ 後 データ出力遅延時間		75		1.0tCLCL-50		
11	tRHDX	RD ↑ 後 データ保持時間	0		0			
12	tRLRH	RD Lowパルス幅	115		1.0tCLCL-10			
13	tOVWL	WR ↓ 前 データ準備時間	42.5		0.5tCLCL-20		1	
14	tWHDX	WR ↑ 後 データ保持時間	115		1.0tCLCL-10			
15	tDVWH	WR ↑ 前 データ有効時間	125		1.0tCLCL			
16	tWLWH	WR Lowパルス幅	115		1.0tCLCL-10			

注1: 一般式の定数はデューティサイクル=50%(XTAL1の外部クロックのHigh時間は半周期)と仮定した値です。

注2: 一般式の定数はデューティサイクル=50%(XTAL1の外部クロックのLow時間は半周期)と仮定した値です。

表99. 外部データメモリ特性 (VCC=4.5V~5.5V、SRWn1=0, SRWn0=1 (1待ち周期))

	シンボル	項目	8MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	16		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		200		2.0tCLCL-50		ns
12	tRLRH	RD Lowパルス幅	240		2.0tCLCL-10			
15	tDVWH	WR ↑ 前 データ有効時間	250		2.0tCLCL			
16	tWLWH	WR Lowパルス幅	240		2.0tCLCL-10			

表100. 外部データメモリ特性 (VCC=4.5V~5.5V、SRWn1=1, SRWn0=0 (2待ち周期))

	シンボル	項目	8MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	16		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		325		3.0tCLCL-50		ns
12	tRLRH	RD Lowパルス幅	365		3.0tCLCL-10			
15	tDVWH	WR ↑ 前 データ有効時間	375		3.0tCLCL			
16	tWLWH	WR Lowパルス幅	365		3.0tCLCL-10			

表101. 外部データメモリ特性 (VCC=4.5V~5.5V、SRWn1=1, SRWn0=1 (2&1待ち周期))

	シンボル	項目	8MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	16		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		325		3.0tCLCL-50		ns
12	tRLRH	RD Lowパルス幅	365		3.0tCLCL-10			
14	tWHDX	WR ↑ 後 データ保持時間	240		2.0tCLCL-10			
15	tDVWH	WR ↑ 前 データ有効時間	375		3.0tCLCL			
16	tWLWH	WR Lowパルス幅	365		3.0tCLCL-10			

表102. 外部データメモリ特性 (VCC=2.7V~5.5V、待ちなし)

	シンボル	項目	4MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	8		MHz
1	tLHLL	ALE Highパルス幅	235		1.0tCLCL-15			ns
2	tAVLL	ALE ↓ 前 下位アドレス準備時間	115		0.5tCLCL-10		1	
3A	tLLAX,ST	ライト時 ALE ↓ 後 下位アドレス保持時間	5		5			
3B	tLLAX,LD	リード時 ALE ↓ 後 下位アドレス保持時間	5		5			
4	tAVLLC	ALE ↓ 前 上位アドレス準備時間	115		0.5tCLCL-10		1	
5	tAVRL	リード時 RD ↓ 前 下位アドレス有効時間	235		1.0tCLCL-15			
6	tAVWL	ライト時 WR ↓ 前 下位アドレス有効時間	235		1.0tCLCL-15			
7	tLLWL	ALE ↓ 後 WR ↓ 遅延時間	115	130	0.5tCLCL-10	0.5tCLCL+5	2	
8	tLLRL	ALE ↓ 後 RD ↓ 遅延時間	115	130	0.5tCLCL-10	0.5tCLCL+5	2	
9	tDVRH	RD ↑ 前 データ準備時間	45		45			
10	tRLDV	RD ↓ 後 データ出力遅延時間		190		1.0tCLCL-60		
11	tRHDX	RD ↑ 後 データ保持時間	0		0			
12	tRLRH	RD Lowパルス幅	235		1.0tCLCL-15			
13	tOVWL	WR ↓ 前 データ準備時間	105		0.5tCLCL-20		1	
14	tWHDX	WR ↑ 後 データ保持時間	235		1.0tCLCL-15			
15	tDVWH	WR ↑ 前 データ有効時間	250		1.0tCLCL			
16	tWLWH	WR Lowパルス幅	235		1.0tCLCL-15			

注1: 一般式の定数はデューティサイクル=50%(XTAL1の外部クロックのHigh時間は半周期)と仮定した値です。

注2: 一般式の定数はデューティサイクル=50%(XTAL1の外部クロックのLow時間は半周期)と仮定した値です。

表103. 外部データメモリ特性 (VCC=2.7V~5.5V、SRWn1=0, SRWn0=1 (1待ち周期))

	シンボル	項目	4MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	8		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		440		2.0tCLCL-60		ns
12	tRLRH	RD Lowパルス幅	485		2.0tCLCL-15			
15	tDVWH	WR ↑ 前 データ有効時間	500		2.0tCLCL			
16	tWLWH	WR Lowパルス幅	485		2.0tCLCL-15			

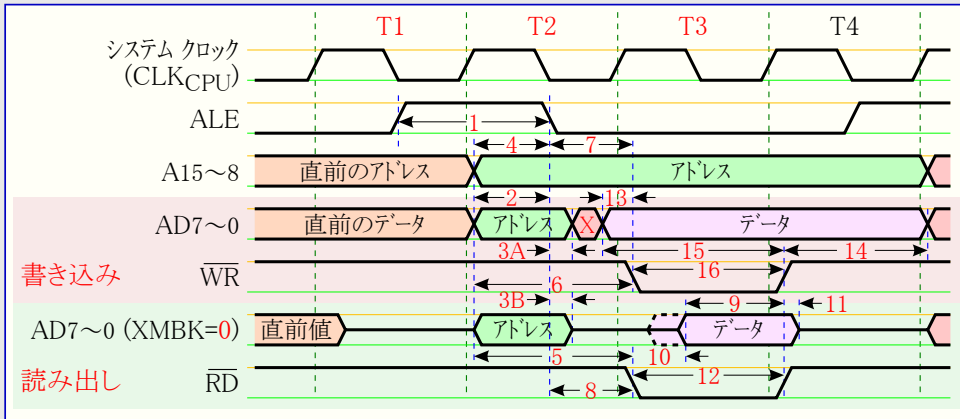
表104. 外部データメモリ特性 (VCC=2.7V~5.5V、SRWn1=1, SRWn0=0 (2待ち周期))

	シンボル	項目	4MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	8		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		690		3.0tCLCL-60		ns
12	tRLRH	RD Lowパルス幅	735		3.0tCLCL-15			
15	tDVWH	WR ↑ 前 データ有効時間	750		3.0tCLCL			
16	tWLWH	WR Lowパルス幅	735		3.0tCLCL-15			

表105. 外部データメモリ特性 (VCC=2.7V~5.5V、SRWn1=1, SRWn0=1 (2&1待ち周期))

	シンボル	項目	4MHz時		一般式			単位
			最小	最大	最小	最大	注	
0	1/tCLCL	発振器周波数			0.0	8		MHz
10	tRLDV	RD ↓ 後 データ出力遅延時間		690		3.0tCLCL-60		ns
12	tRLRH	RD Lowパルス幅	735		3.0tCLCL-15			
14	tWHDX	WR ↑ 後 データ保持時間	485		2.0tCLCL-15			
15	tDVWH	WR ↑ 前 データ有効時間	750		3.0tCLCL			
16	tWLWH	WR Lowパルス幅	735		3.0tCLCL-15			

図89. 待ち状態なし外部データ メモリ アクセス周期 (SRWn1=0, SRWn0=0) [T1~T3]



共通注意事項

SRWn1はSRW11(上位領域)またはSRW01(下位領域)、SRWn0はSRW10(上位領域)またはSRW00(下位領域)です。

[T1~Tn]は図での命令実行周期範囲です。

Xは値変更区間を示します。

最終周期のALEは次命令がRAM(内部または外部)をアクセスする場合だけ存在します。また、最後の周期のアドレスとデータの変更区間の有無も同様です。

図90. 1待ち状態 外部データ メモリ アクセス周期 (SRWn1=0, SRWn0=1) [T1~T4]

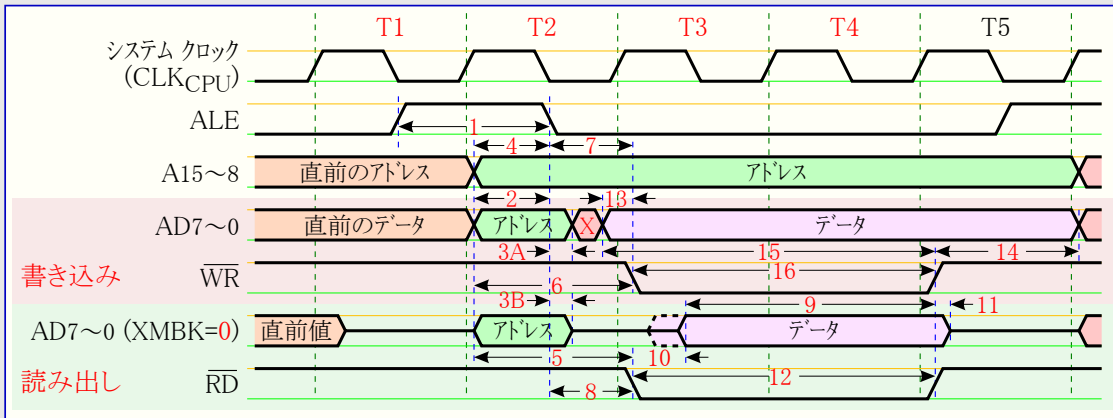


図91. 2待ち状態 外部データ メモリ アクセス周期 (SRWn1=1, SRWn0=0) [T1~T5]

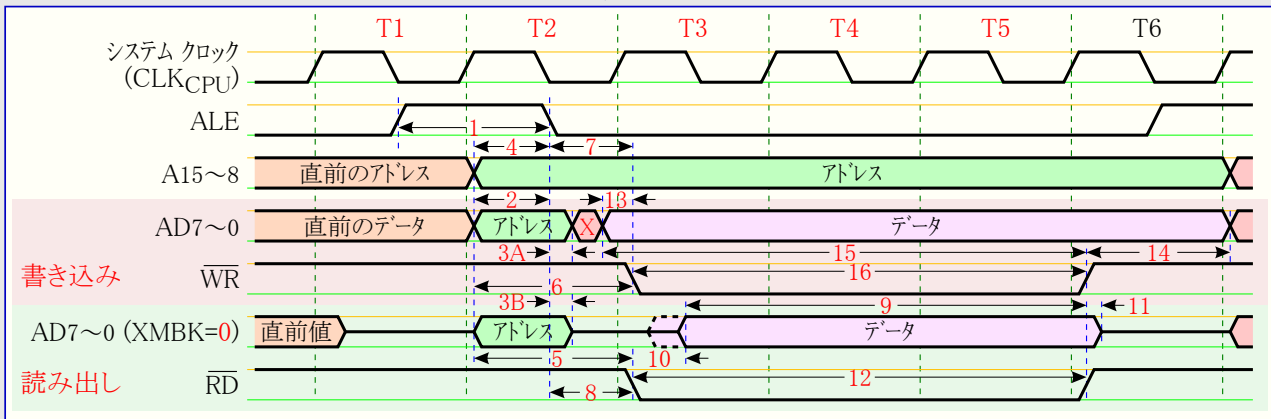
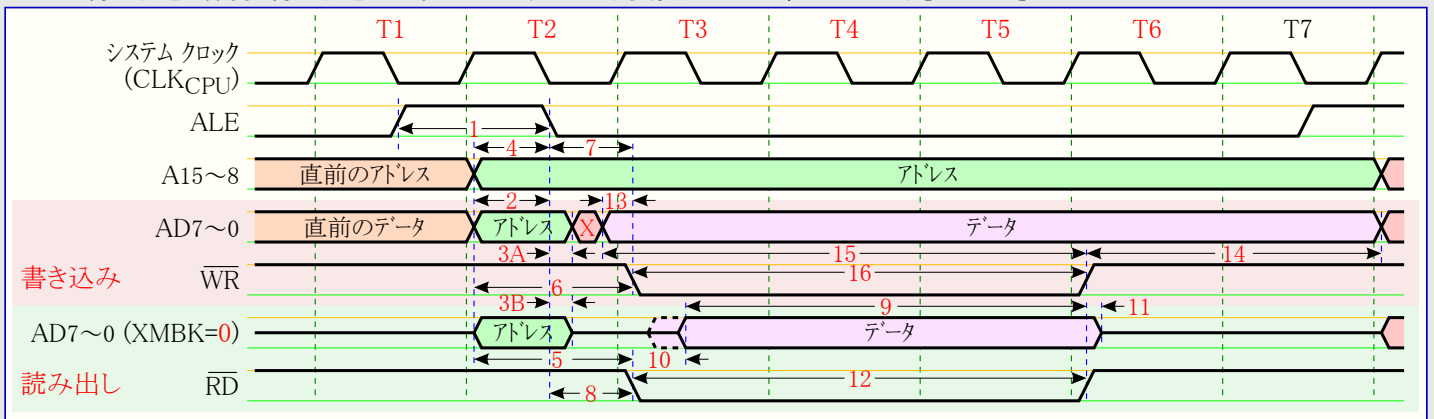


図92. 2待ち状態+保持1待ち状態 外部データ メモリ アクセス周期 (SRWn1=1, SRWn0=1) [T1~T6]



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。

パワーダウン動作での消費電力はクロック選択と無関係です。

消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。支配的な要素は動作電圧と動作周波数です。

容量性負荷のピンの引き込み電流は(1つのピンに対して) $CL(\text{負荷容量}) \times VCC(\text{動作電圧}) \times f(I/O\text{ピンの平均切り替え周波数})$ として推測できます。

デバイスは検査範囲よりも高い周波数特性を示します。デバイスは注文番号が示す周波数よりも高い周波数での機能特性を保証されません。

ウォッチドッグ タイマ許可のパワーダウン動作での消費電流とウォッチドッグ タイマ禁止のパワーダウン動作での消費電流間の違いは、ウォッチドッグ タイマによって引き込んだ(消費した)差電流を表します。

活動動作消費電流

図93. 活動動作消費電流 対 周波数 (100kHz～1MHz)

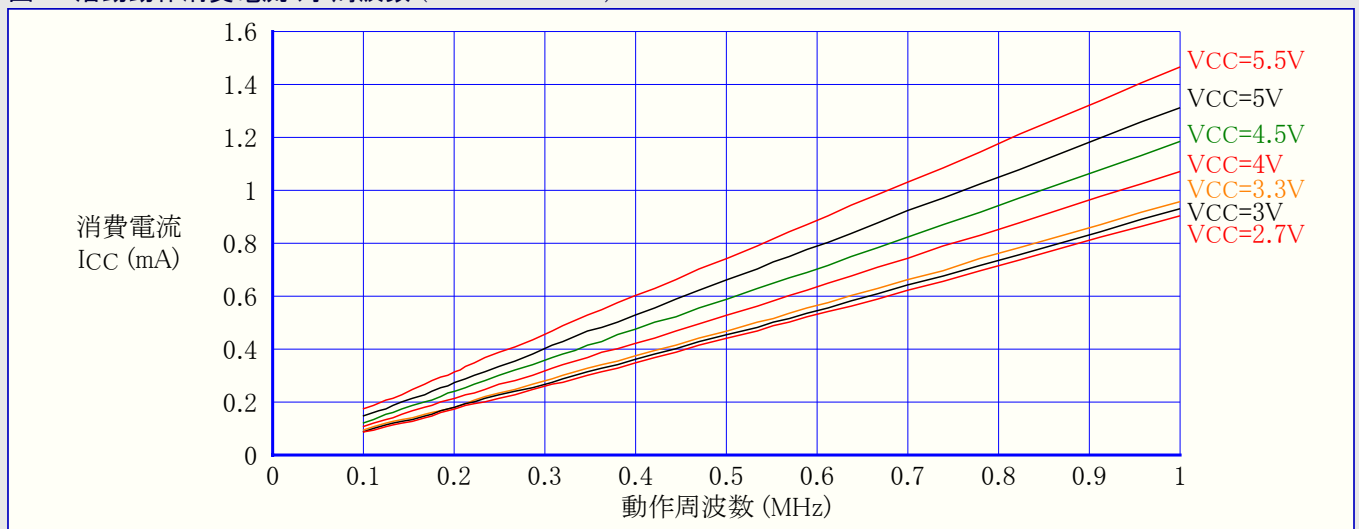


図94. 活動動作消費電流 対 周波数 (1MHz～20MHz)

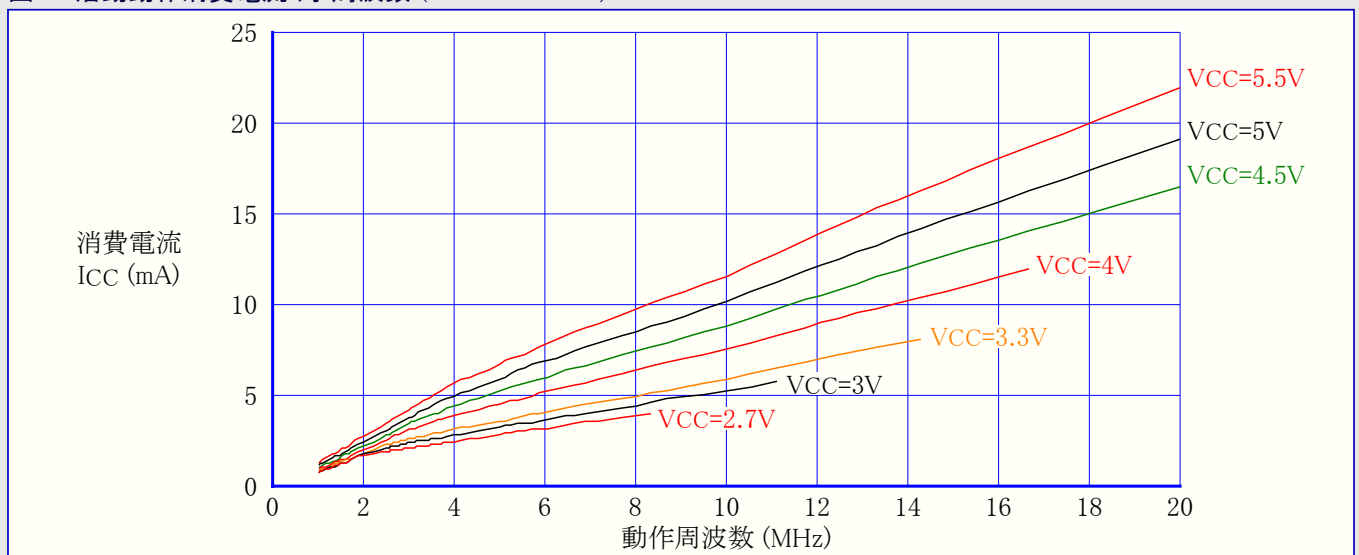


図95. 活動動作消費電流 対 動作電圧 (内蔵RC発振器, 8MHz)

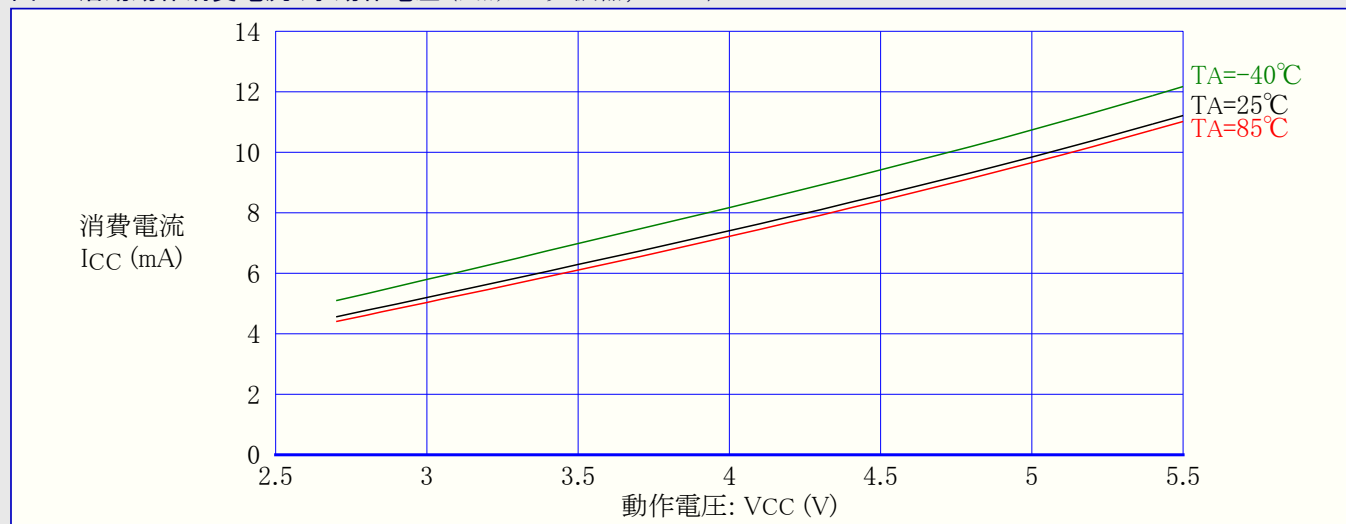


図96. 活動動作消費電流 対 動作電圧 (内蔵RC発振器, 4MHz)

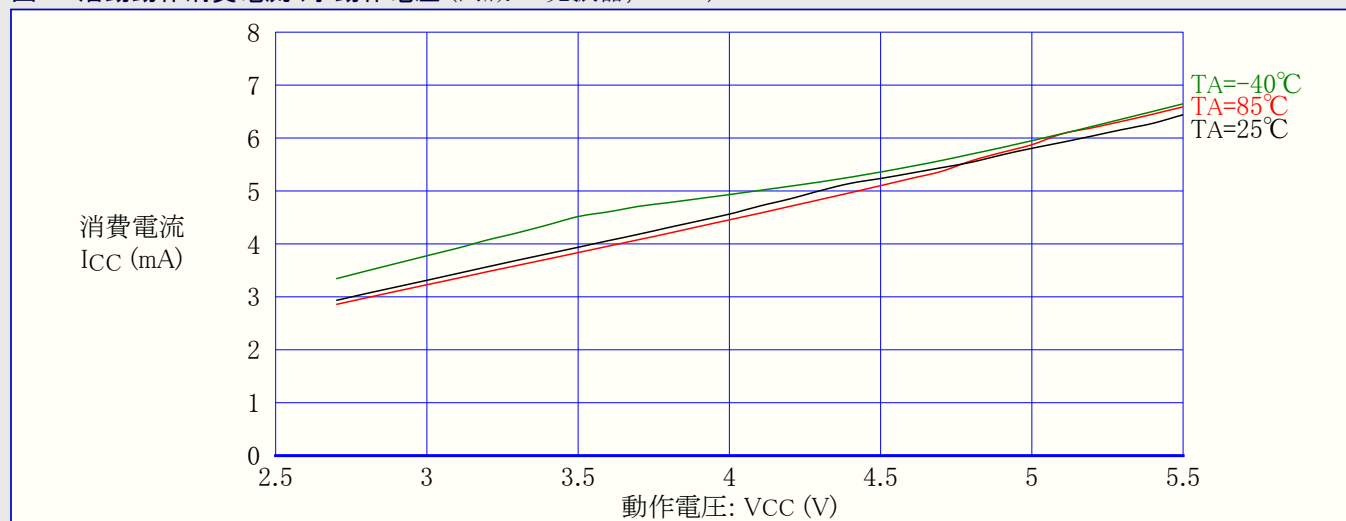


図97. 活動動作消費電流 対 動作電圧 (内蔵RC発振器, 2MHz)

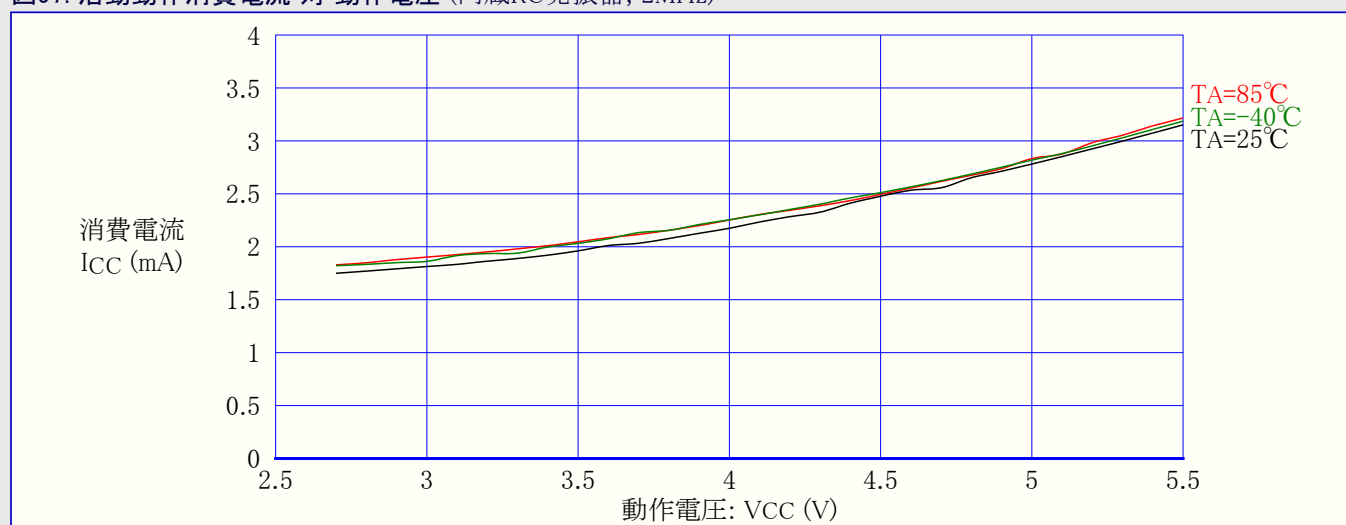


図98. 活動動作消費電流 対 動作電圧 (内蔵RC発振器, 1MHz)

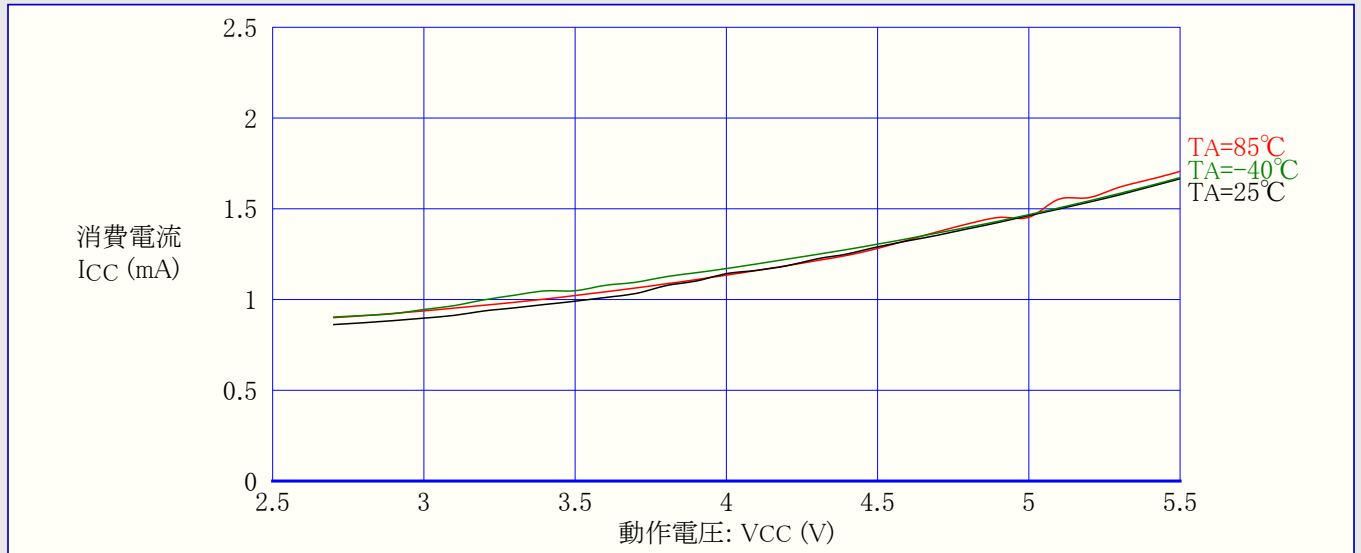
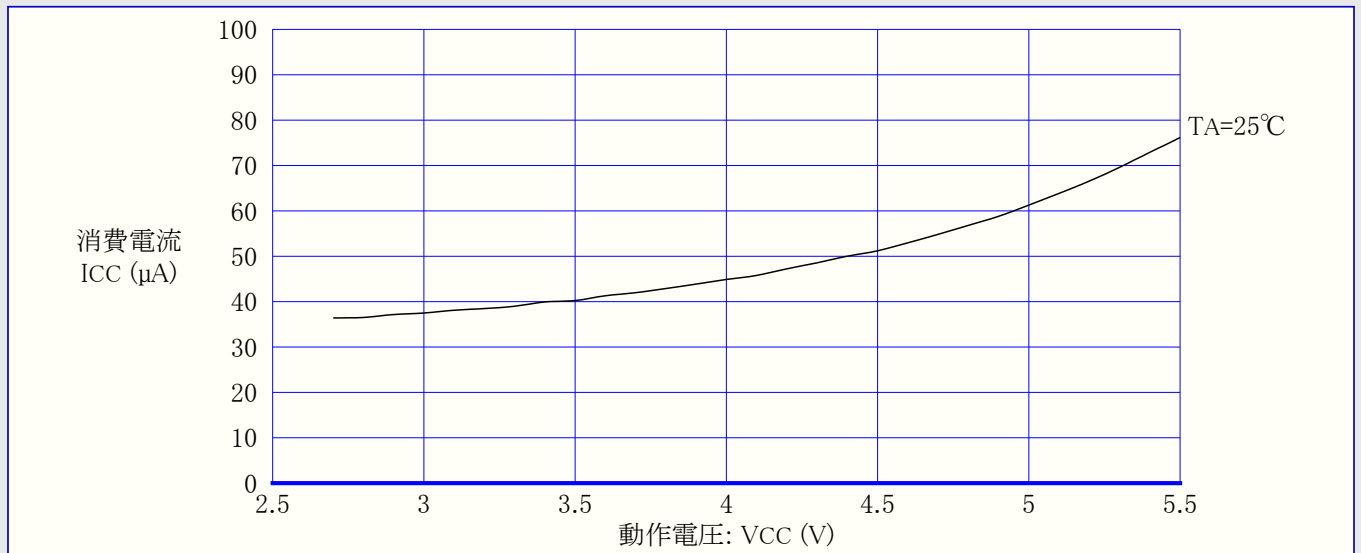


図99. 活動動作消費電流 対 動作電圧 (32kHz外部発振器)



アイドル動作消費電流

図100. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

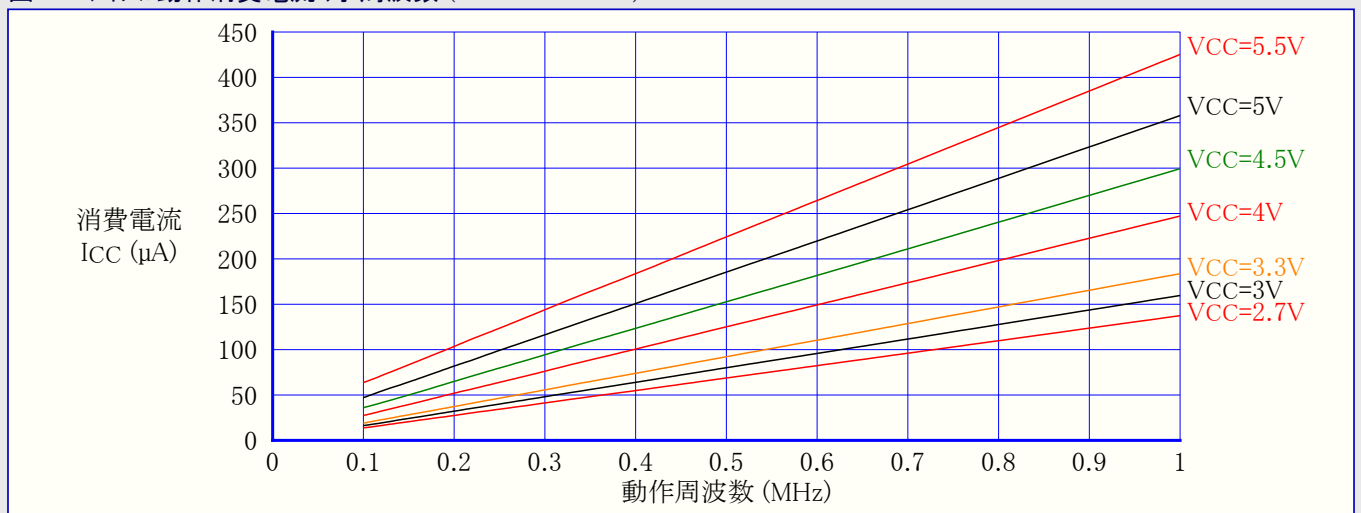


図101. アイドル動作消費電流 対 周波数 (1MHz~20MHz)

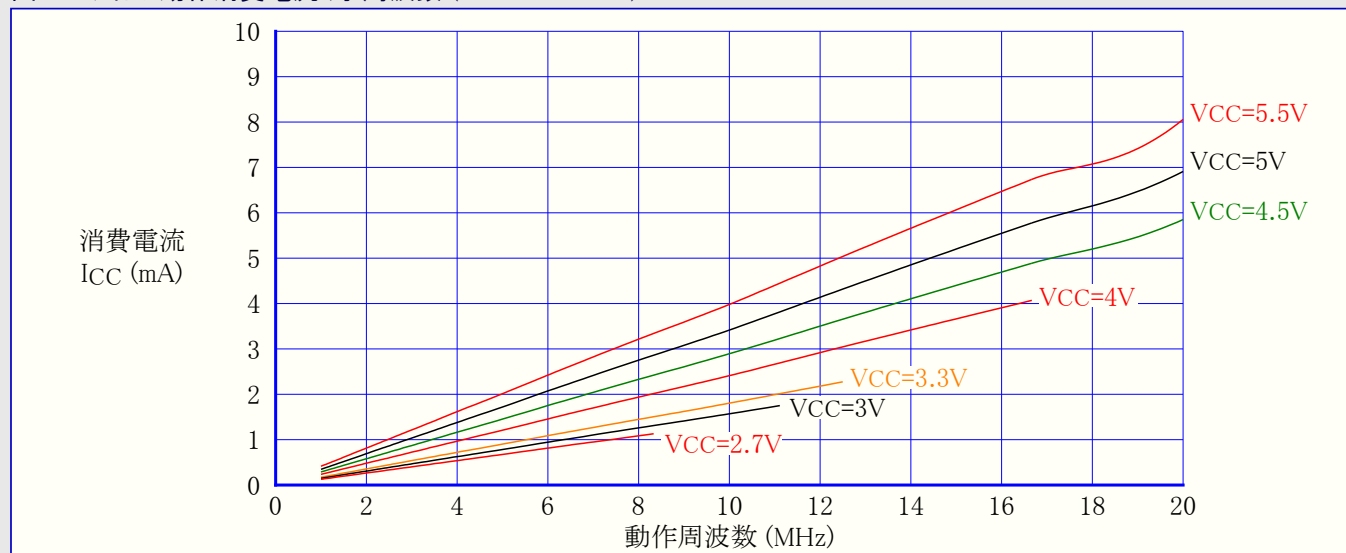


図102. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器, 8MHz)

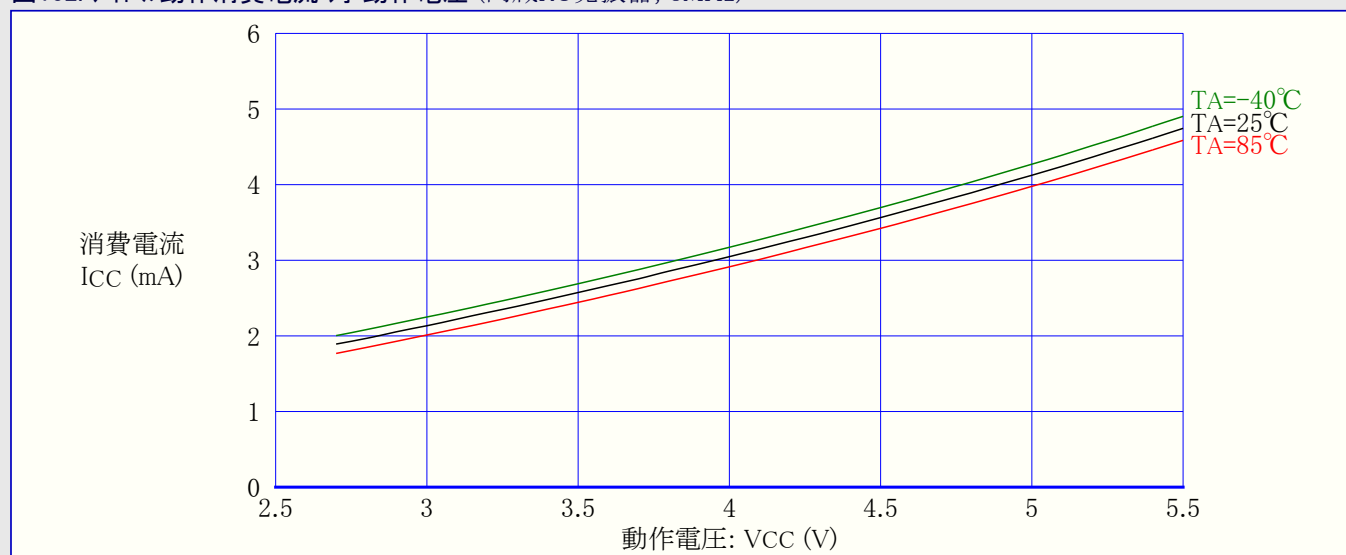


図103. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器, 4MHz)

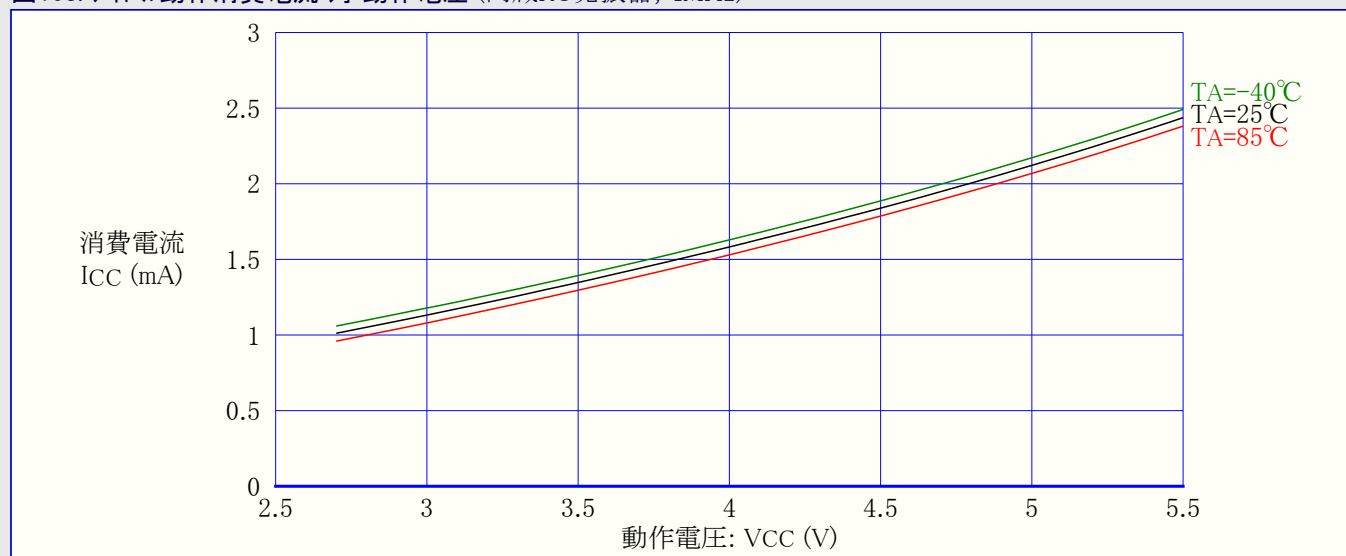


図104. アイトル動作消費電流 対 動作電圧 (内蔵RC発振器, 2MHz)

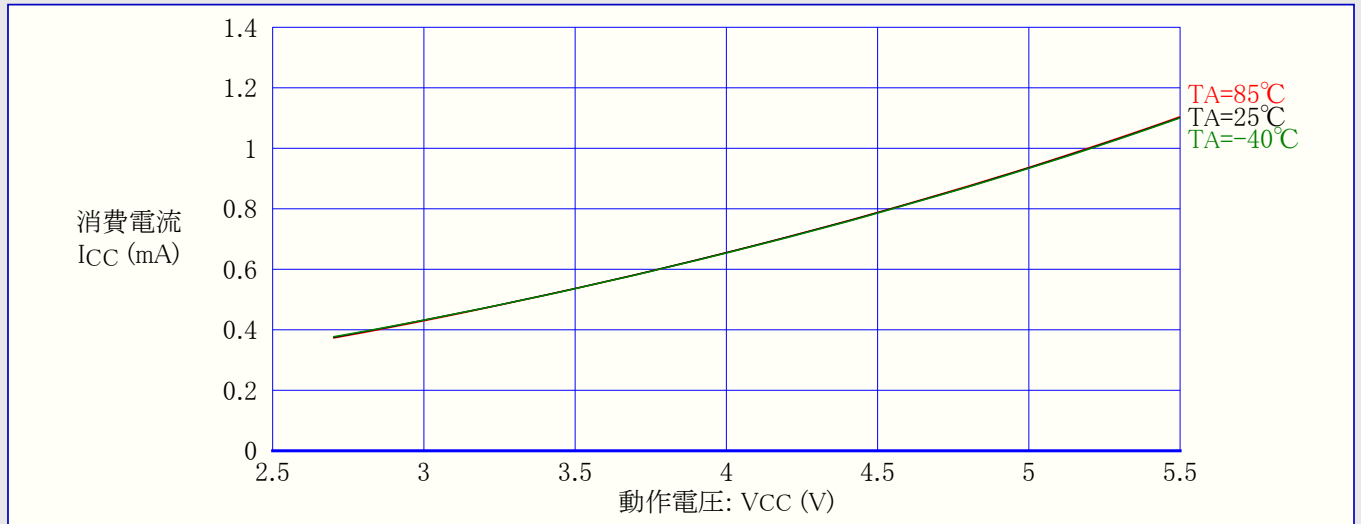


図105. アイトル動作消費電流 対 動作電圧 (内蔵RC発振器, 1MHz)

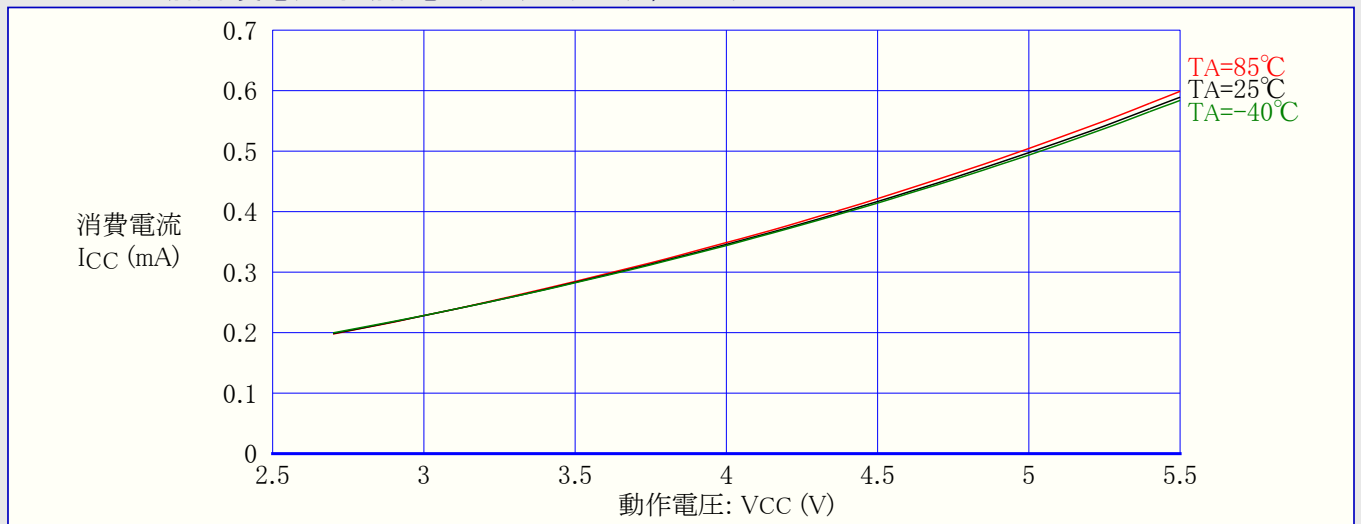
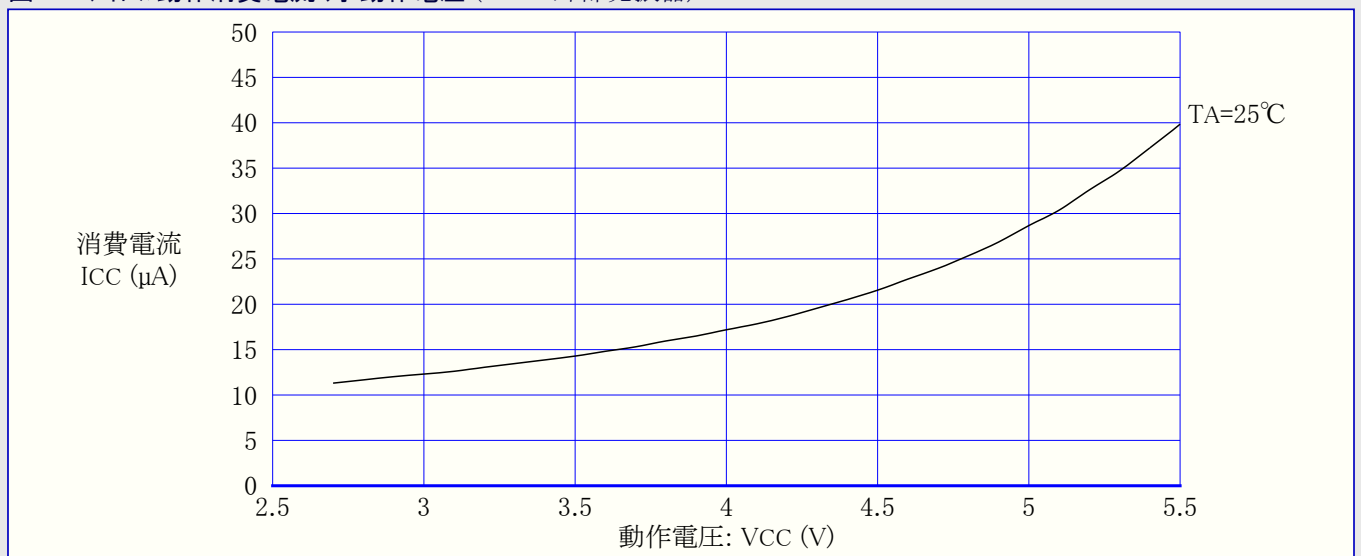


図106. アイトル動作消費電流 対 動作電圧 (32kHz外部発振器)



パワーダウン動作消費電流

図107. 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)

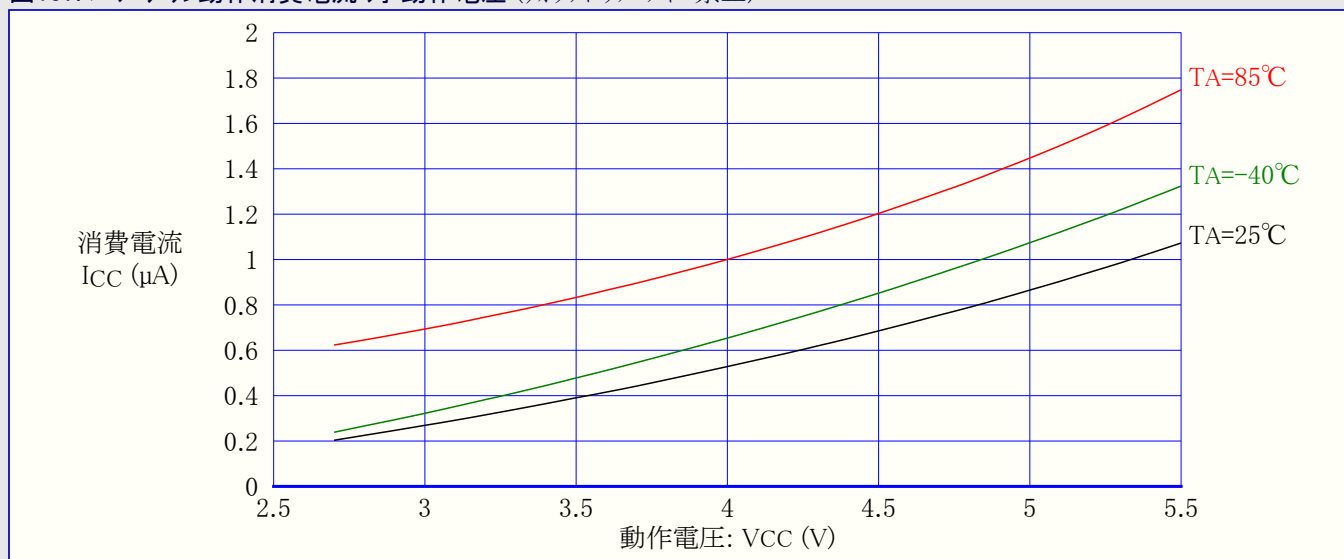
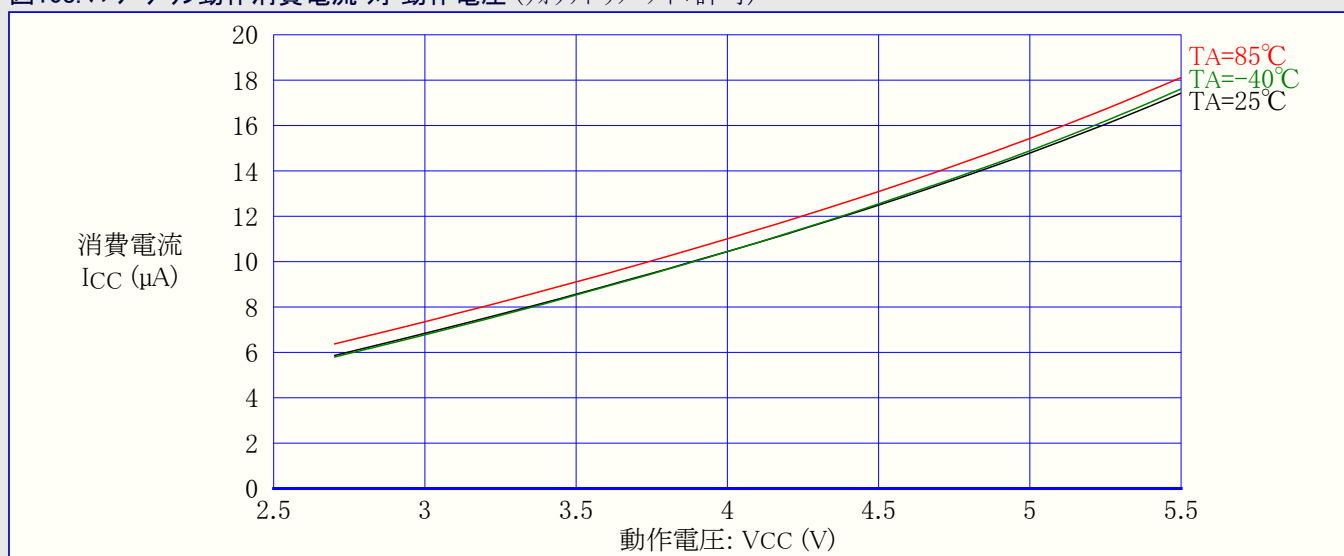


図108. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



スタンバイ動作消費電流

図109. スタンバイ動作消費電流 対 動作電圧 (455kHzセラミック振動子、ウォッチドッグ タイマ禁止)

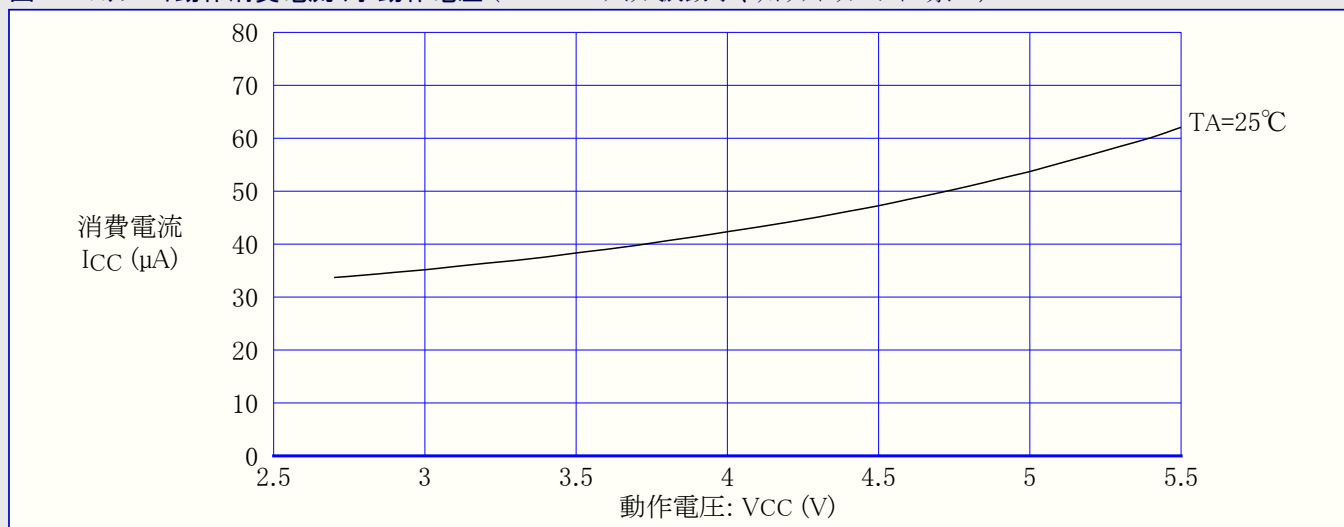


図110. スタンバイ動作消費電流 対 動作電圧 (1MHzセラミック振動子、ウォッチドッグ タイマ禁止)

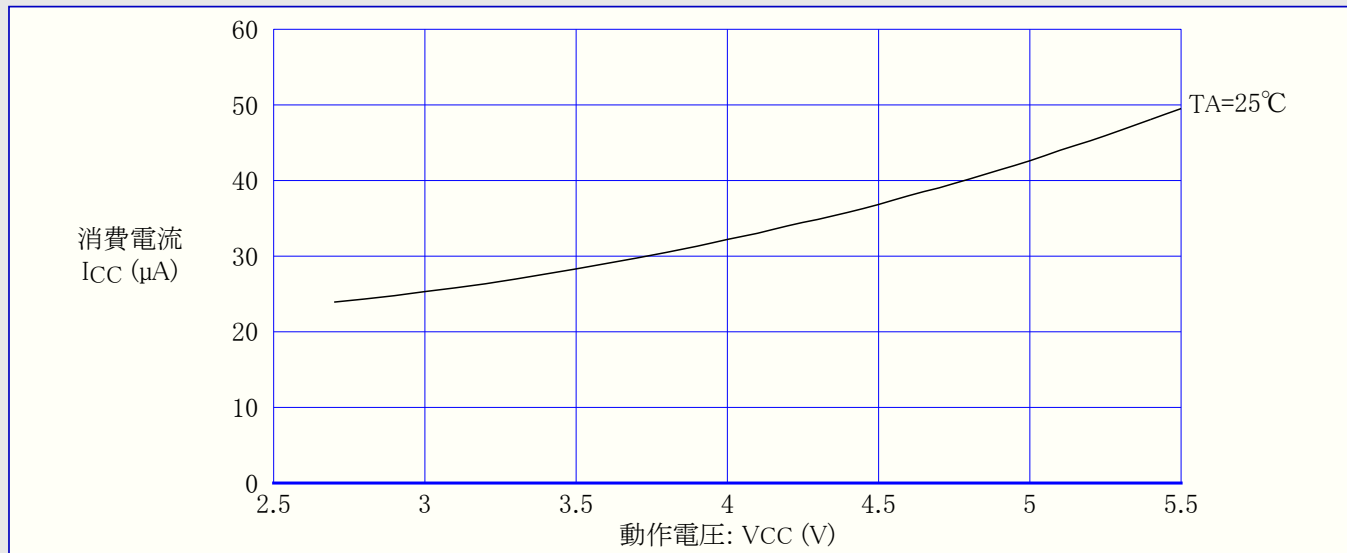


図111. スタンバイ動作消費電流 対 動作電圧 (2MHzセラミック振動子、ウォッチドッグ タイマ禁止)

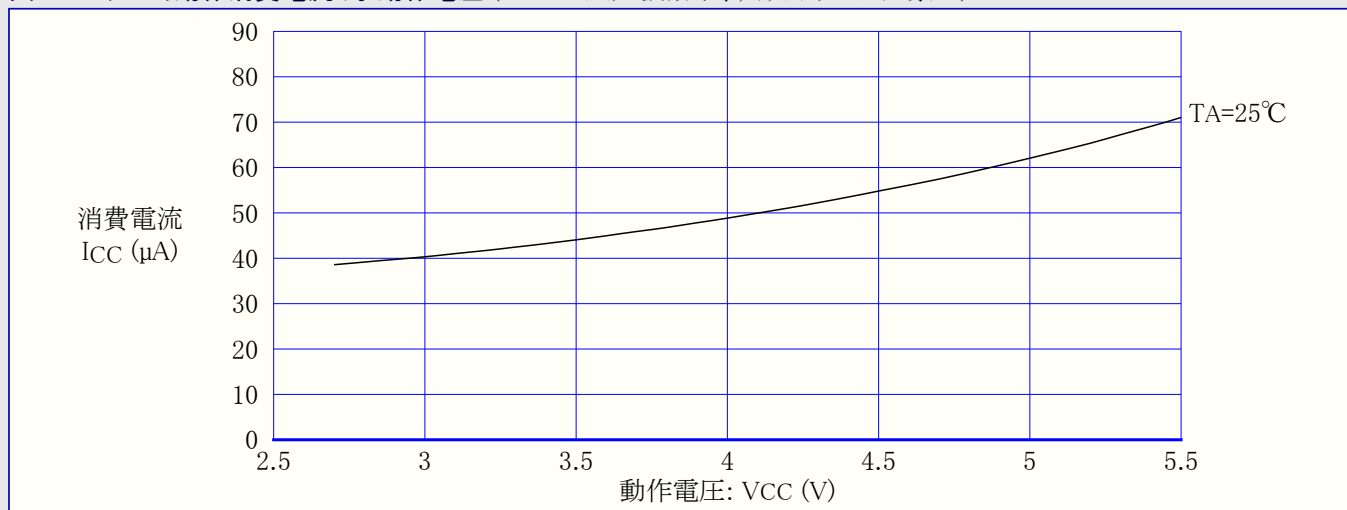


図112. スタンバイ動作消費電流 対 動作電圧 (2MHzクリスタル発振子、ウォッチドッグ タイマ禁止)

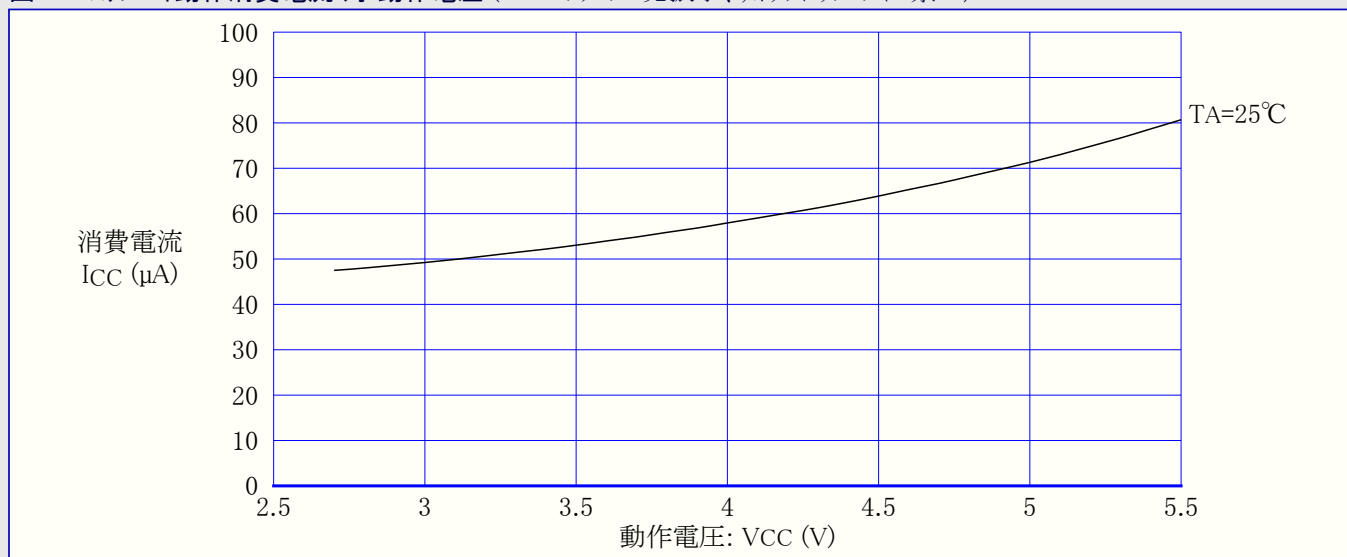


図113. スタンバイ動作消費電流 対 動作電圧 (4MHzセラミック振動子、ウォッチドッグ タイマ禁止)

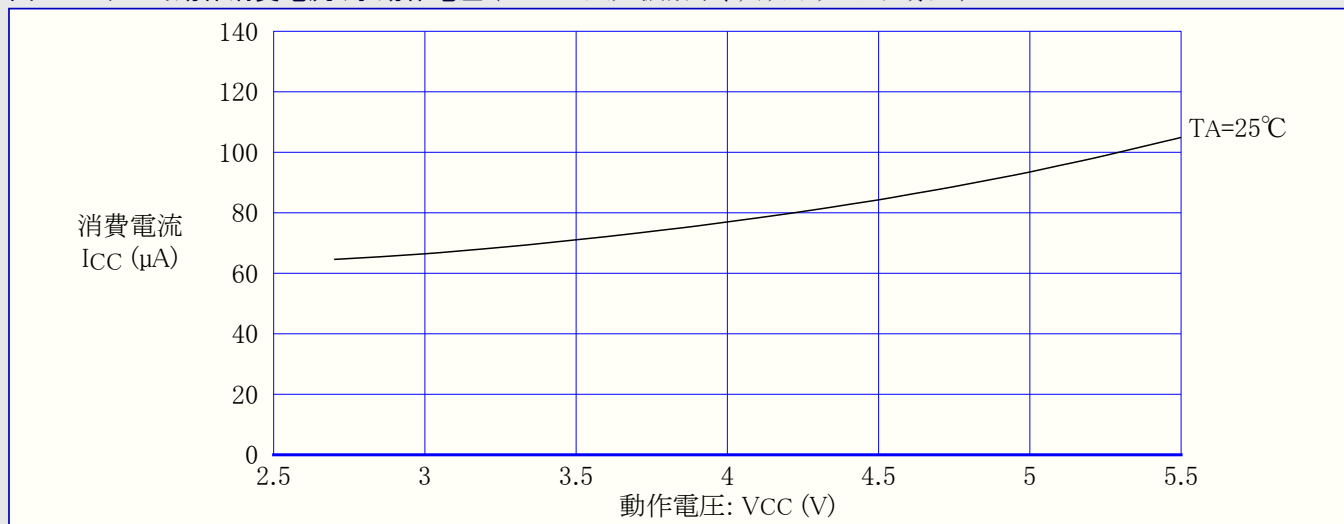


図114. スタンバイ動作消費電流 対 動作電圧 (4MHzクリスタル発振子、ウォッチドッグ タイマ禁止)

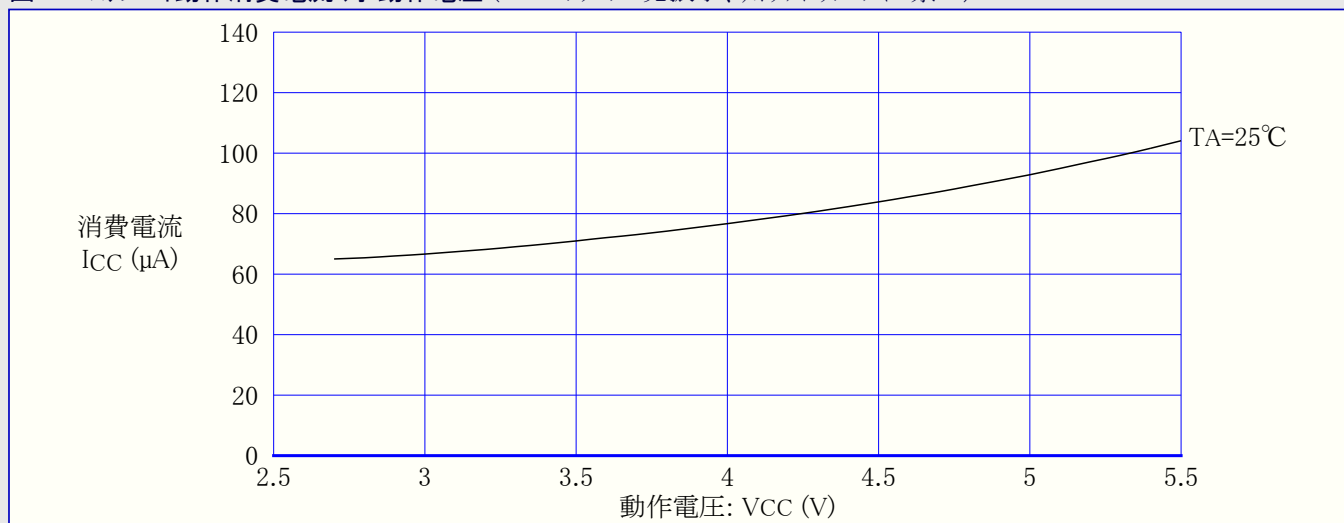


図115. スタンバイ動作消費電流 対 動作電圧 (6MHzセラミック振動子、ウォッチドッグ タイマ禁止)

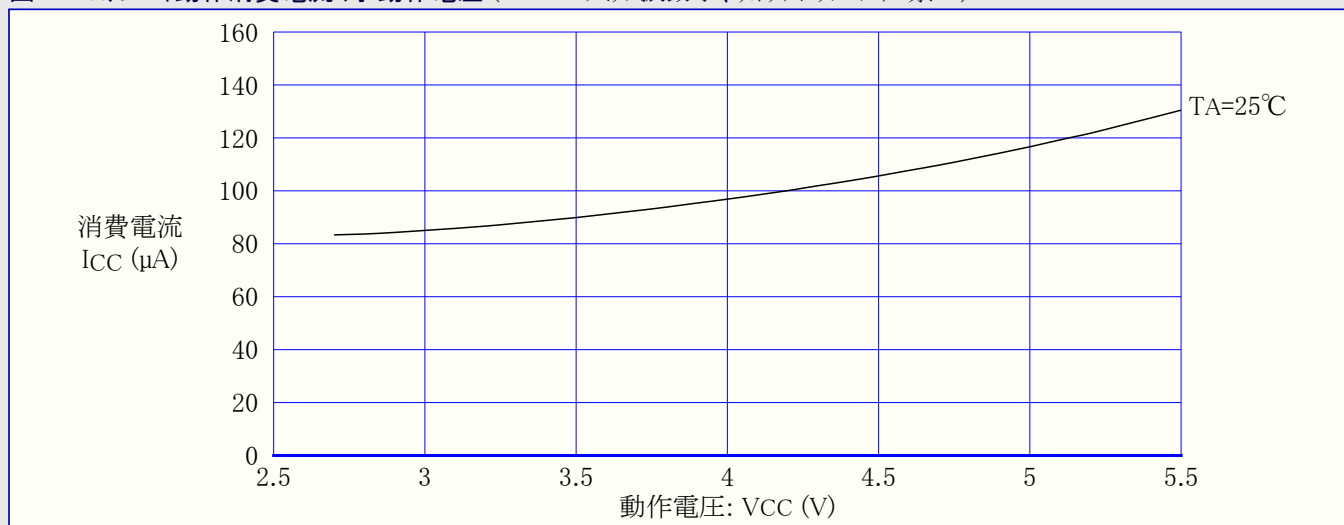
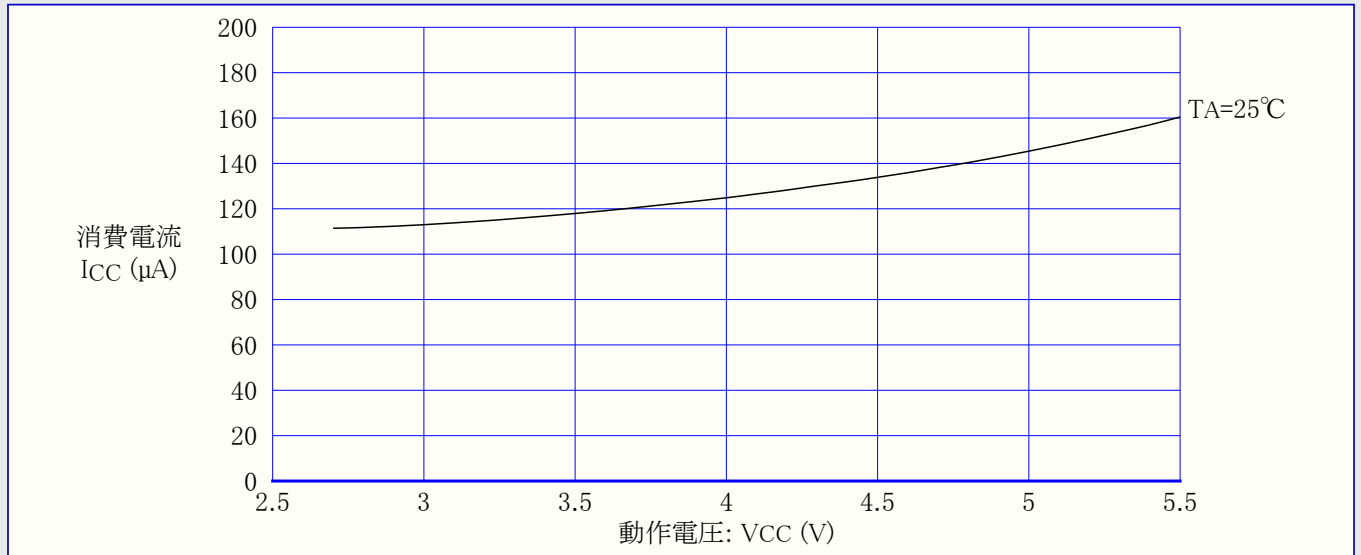


図116. スタンバイ動作消費電流 対 動作電圧 (6MHzクリスタル発振子、ウォッチドッグ タイマ禁止)



ピンプルアップ

図117. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

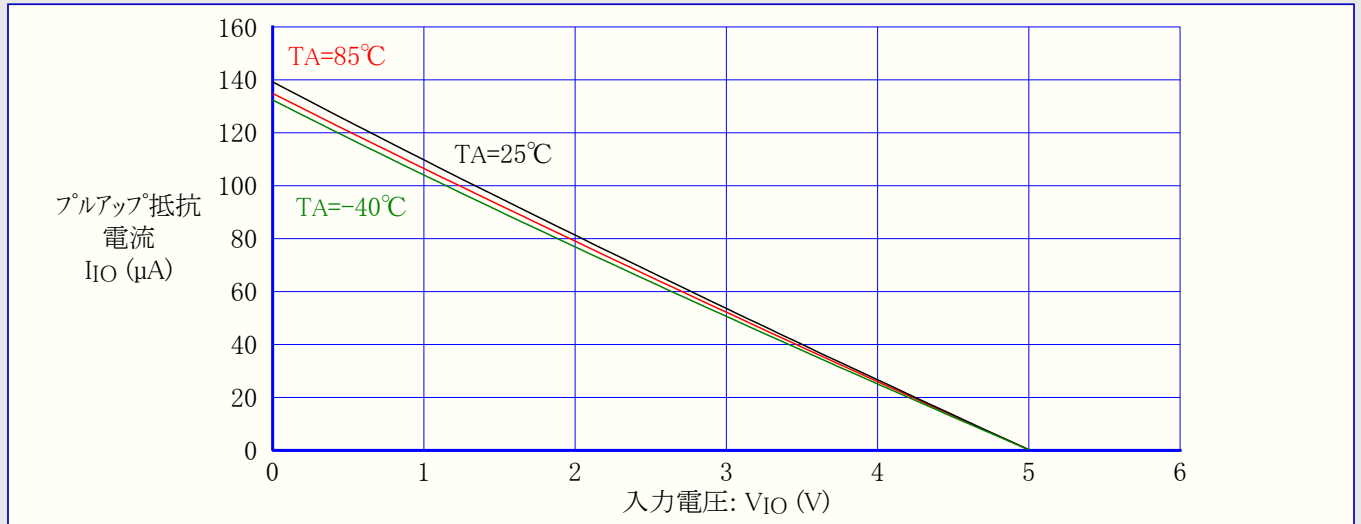


図118. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

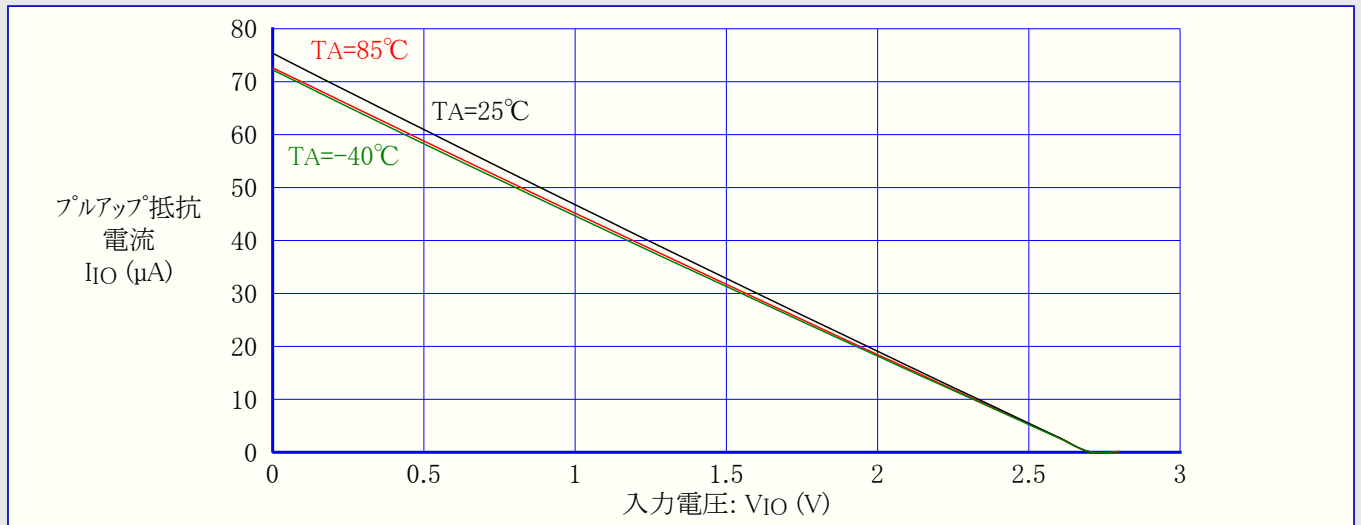


図119. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)

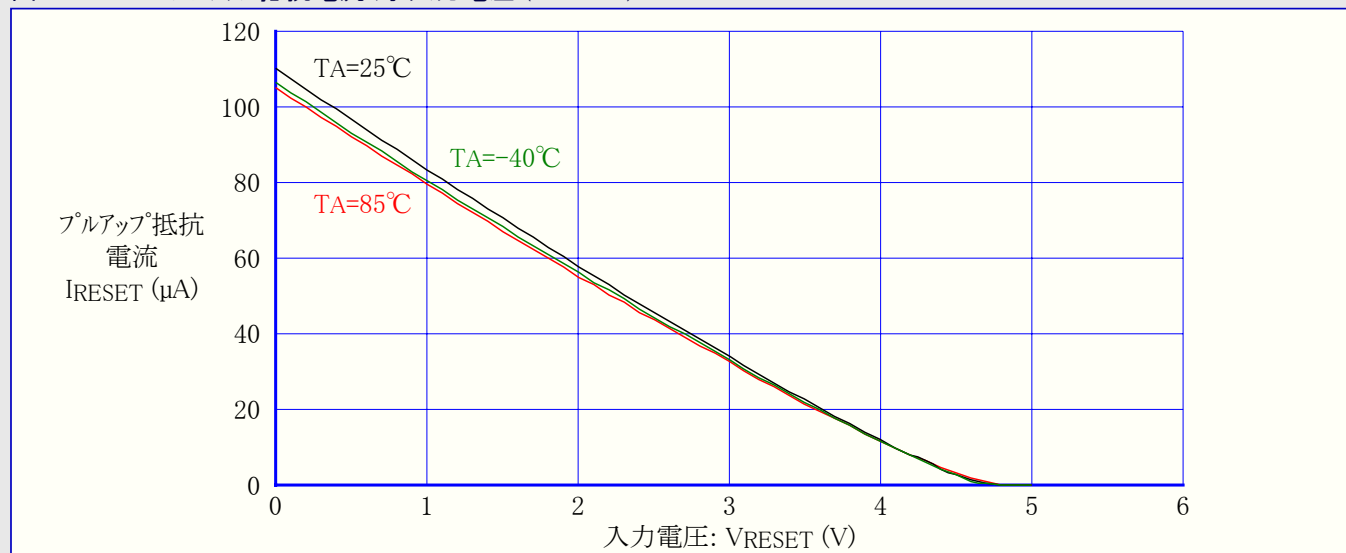
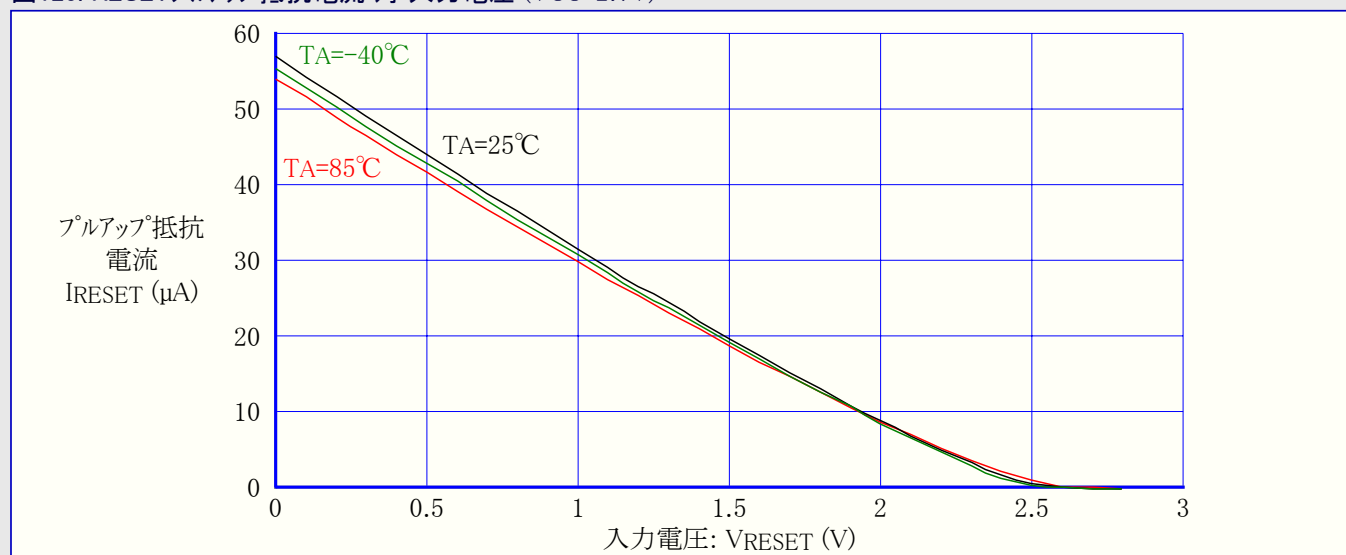


図120. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)



ピン駆動能力

図121. I/Oピン 吐き出し電流 対 出力電圧 (VCC=5V)

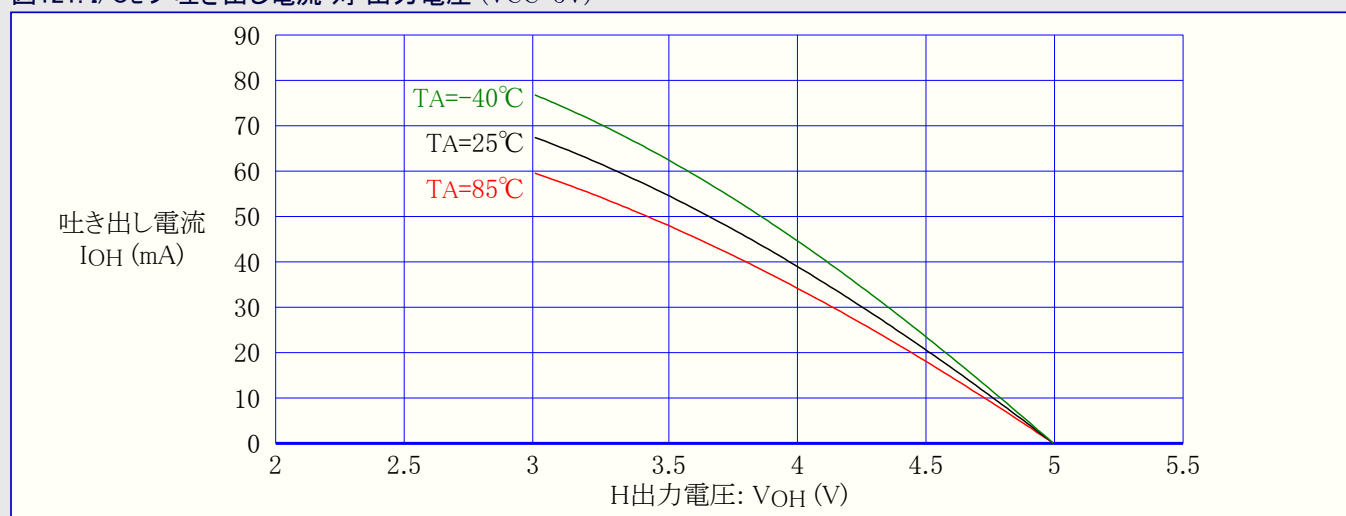


図122. I/Oピン 吐き出し電流 対 出力電圧 (VCC=2.7V)

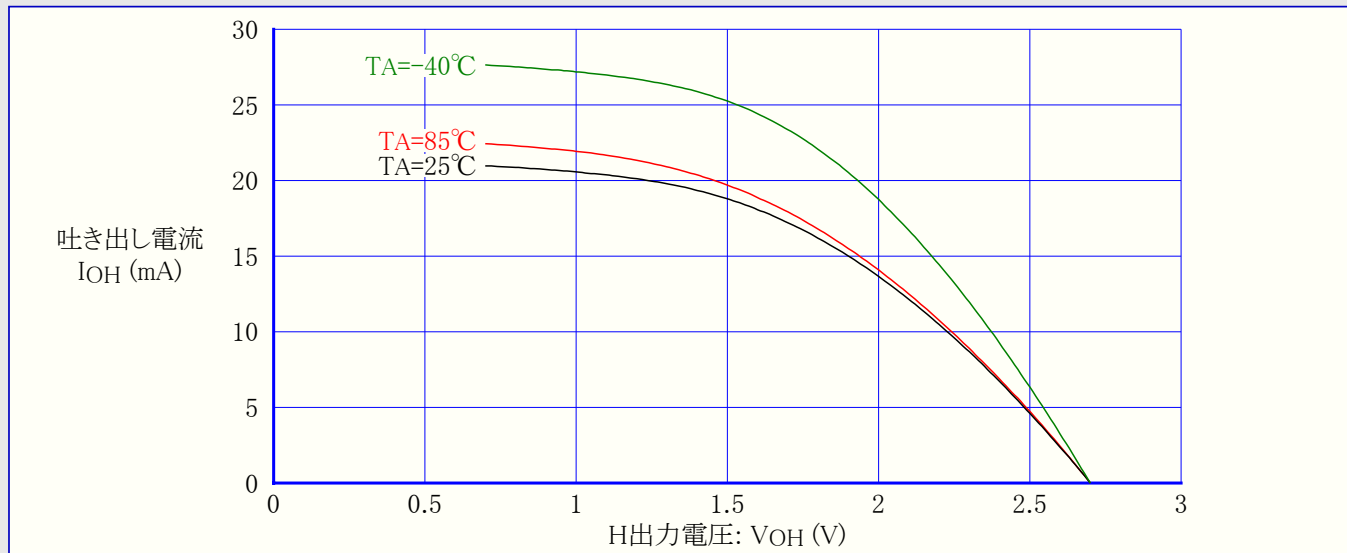


図123. I/Oピン 吸い込み電流 対 出力電圧 (VCC=5V)

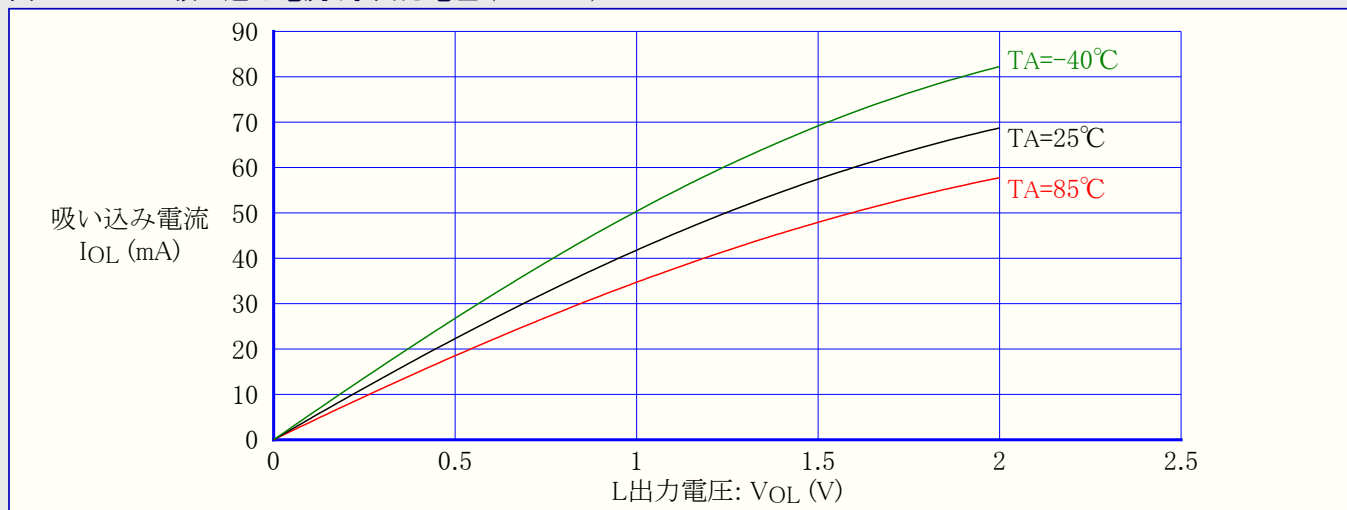
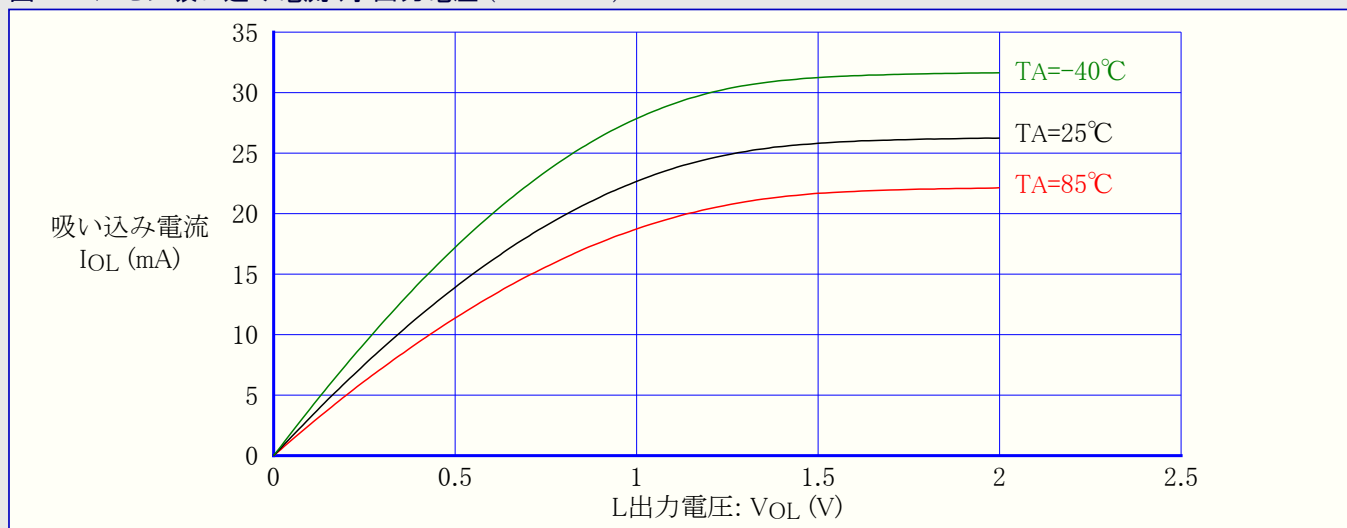


図124. I/Oピン 吸い込み電流 対 出力電圧 (VCC=2.7V)



ピン 閾値とヒステリシス

図125. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IH} , 1読み値)

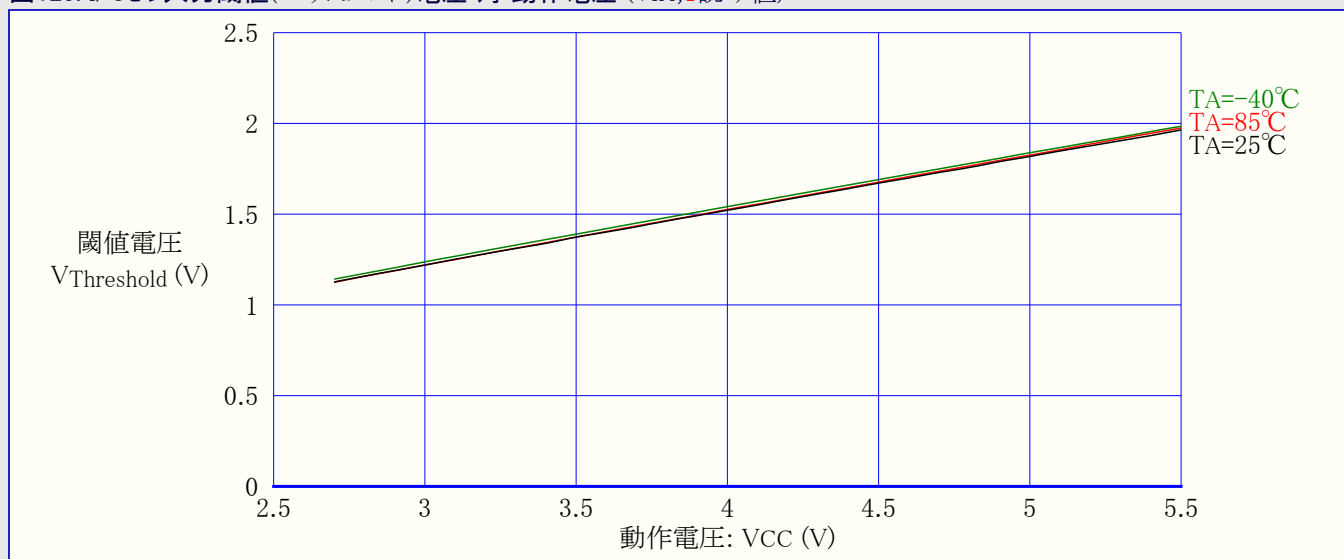


図126. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IL} , 0読み値)

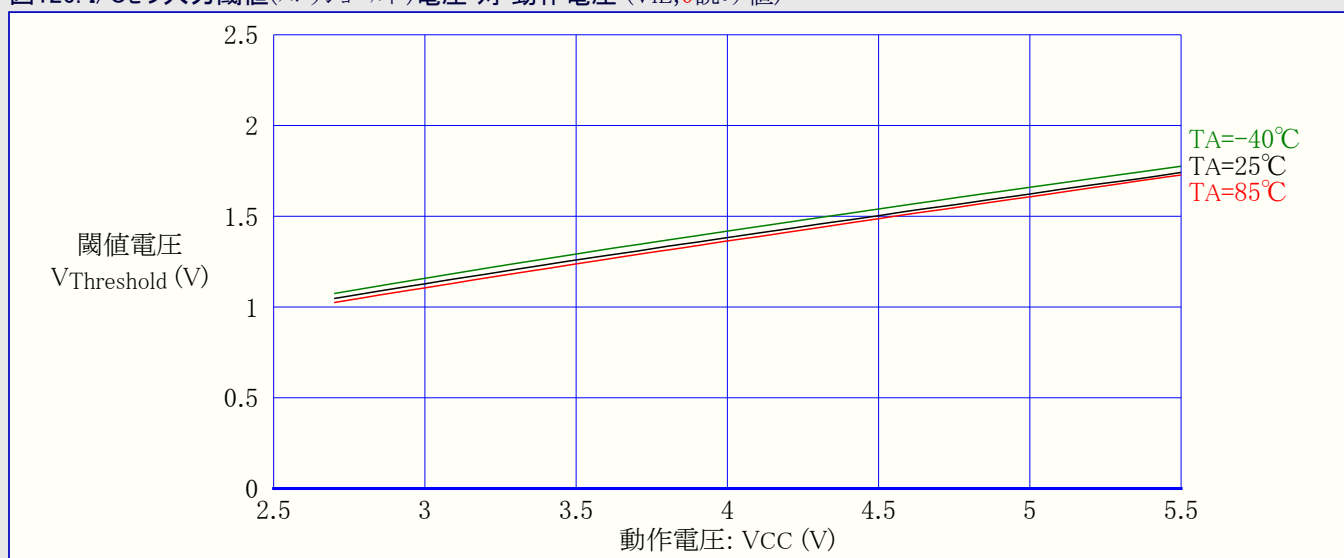


図127. I/Oピン入力ヒステリシス電圧 対 動作電圧

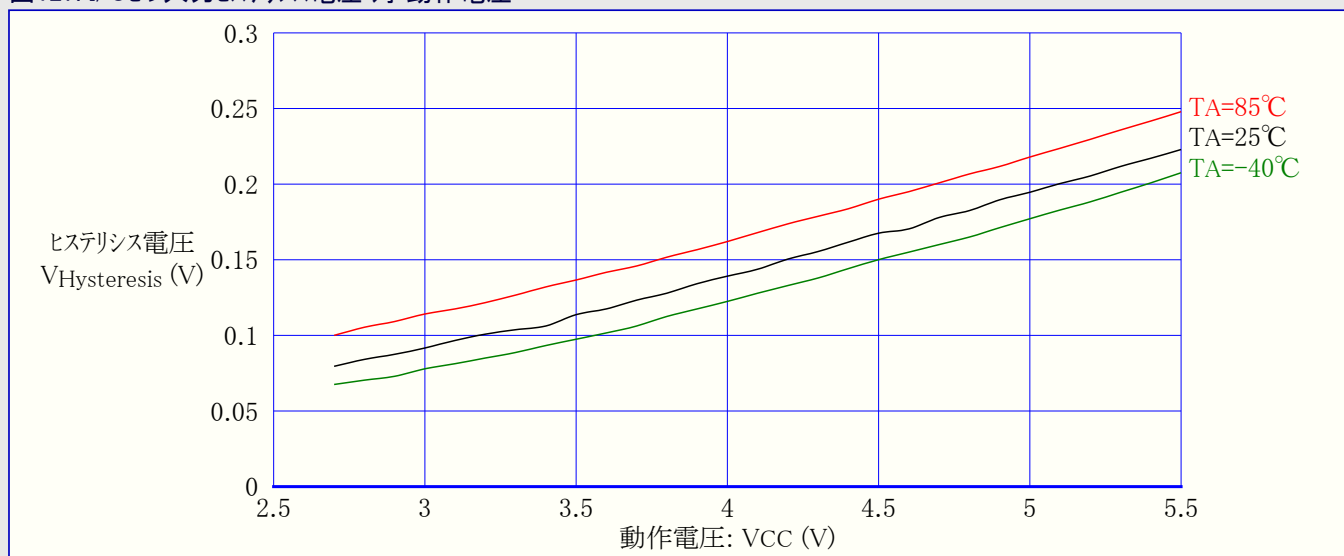


図128. RESET入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IH} , I読み値)

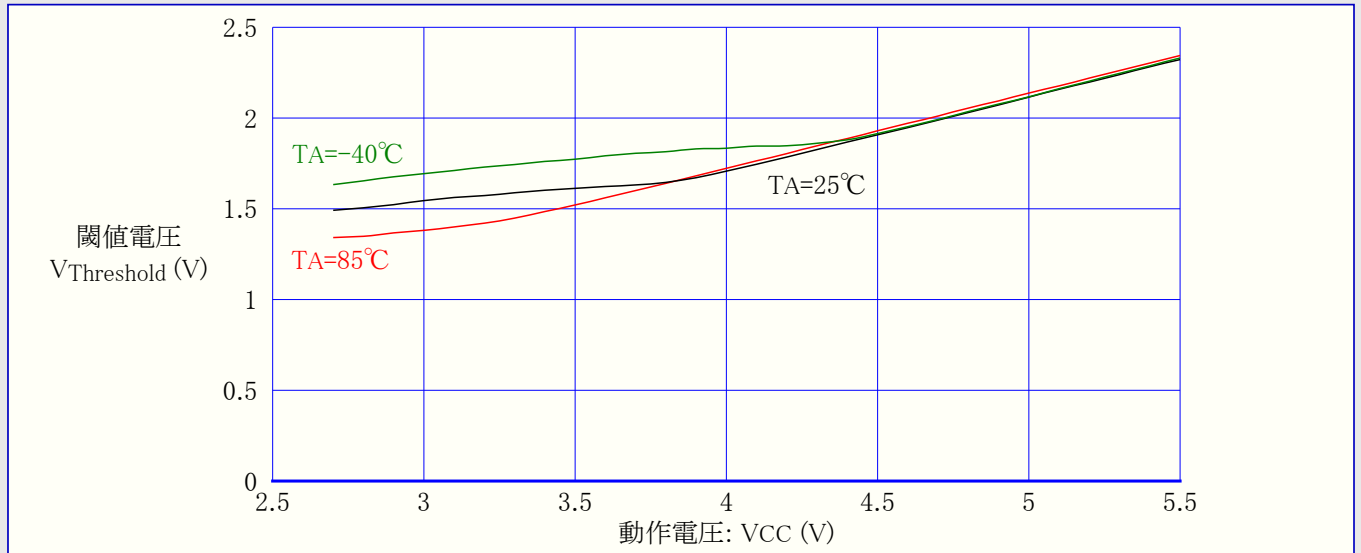


図129. RESET入力閾値(スレッショルド)電圧 対 動作電圧 (V_{IL} , 0読み値)

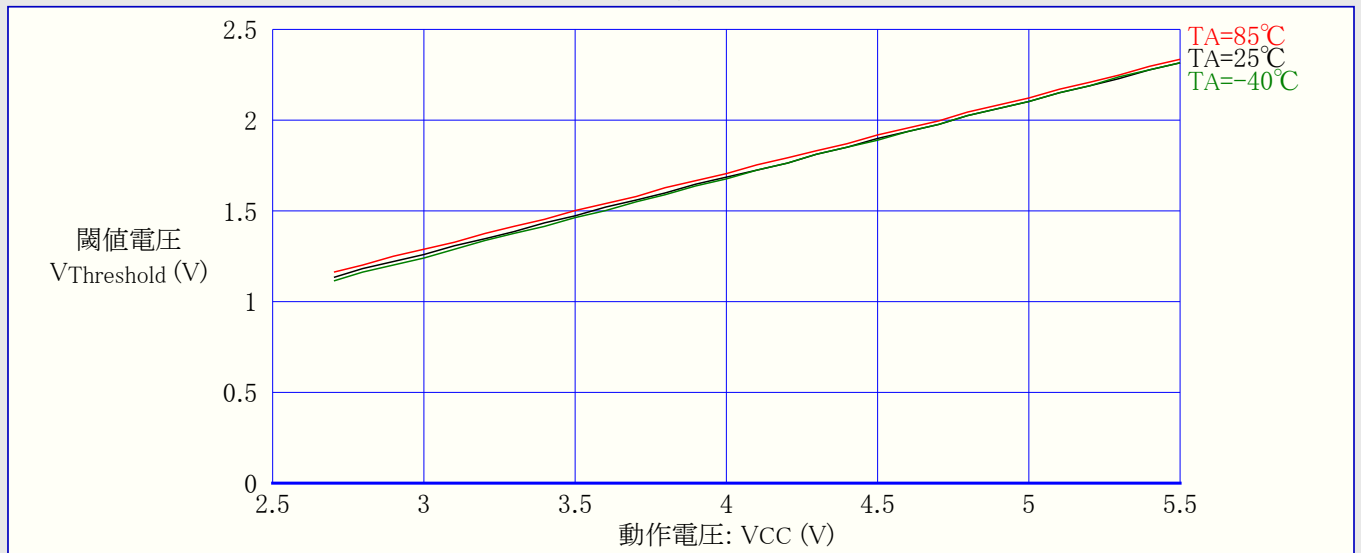
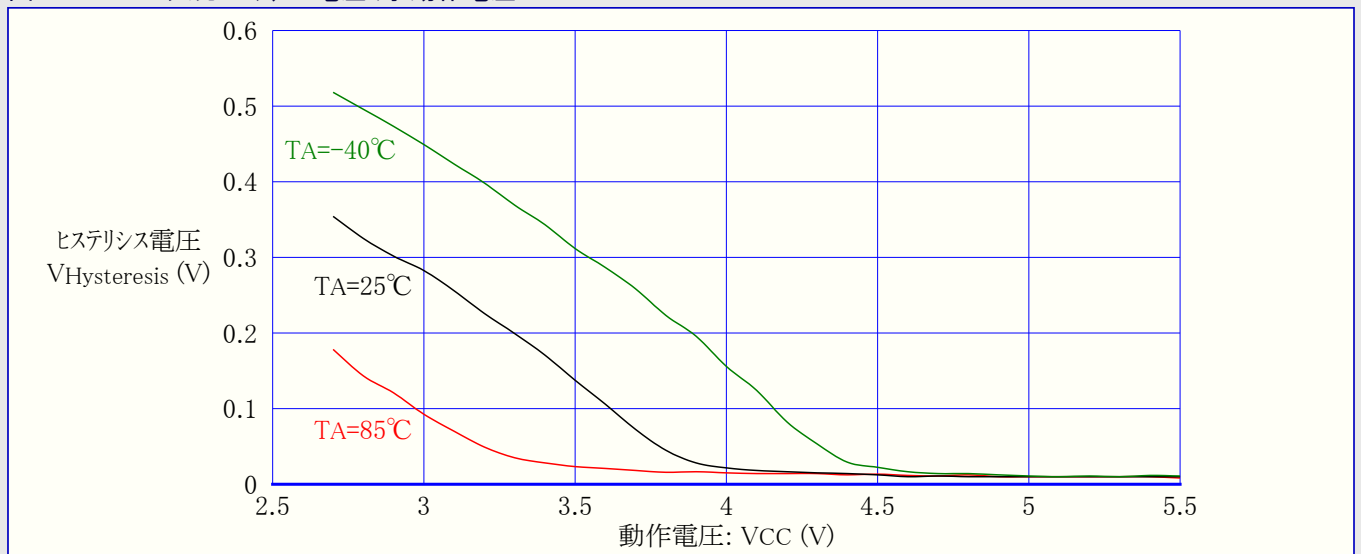


図130. RESET入力ヒステリシス電圧 対 動作電圧



低電圧検出器(BOD)閾値とアナログ比較器オフセット

図131. 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4.0V)

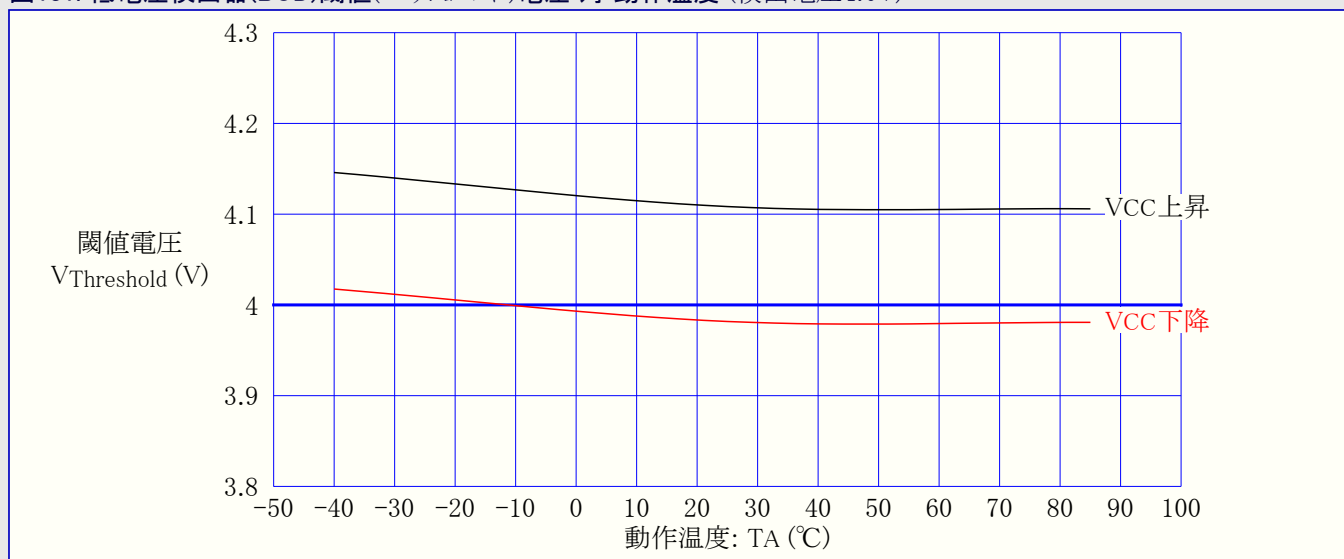


図132. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)

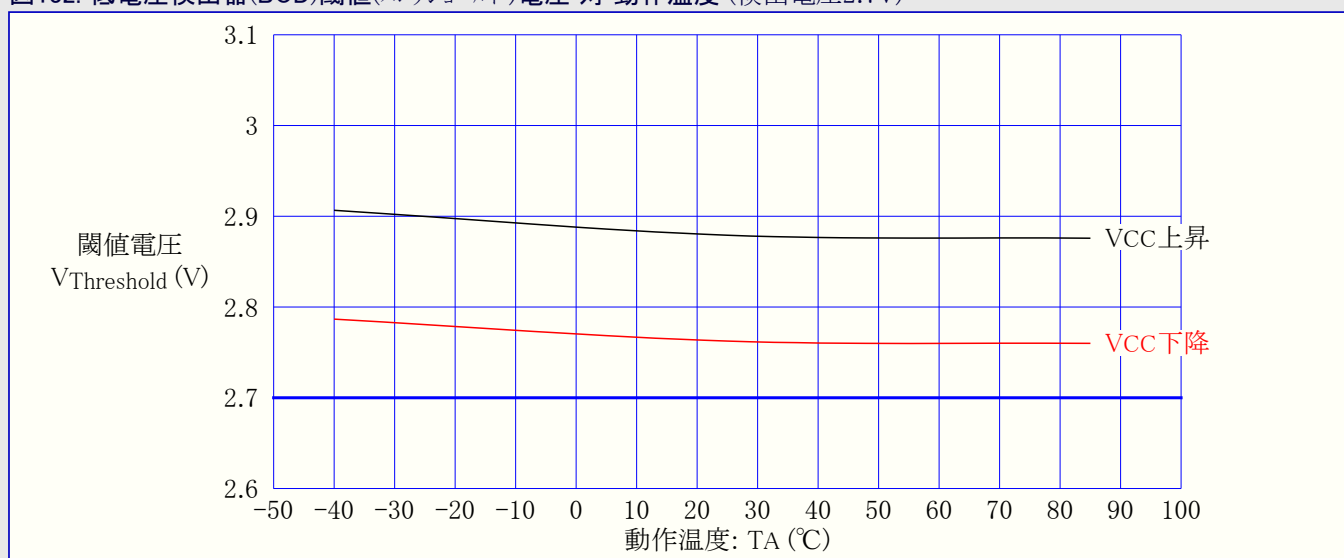


図133. 内部基準(バンドギャップ)電圧 対 動作電圧

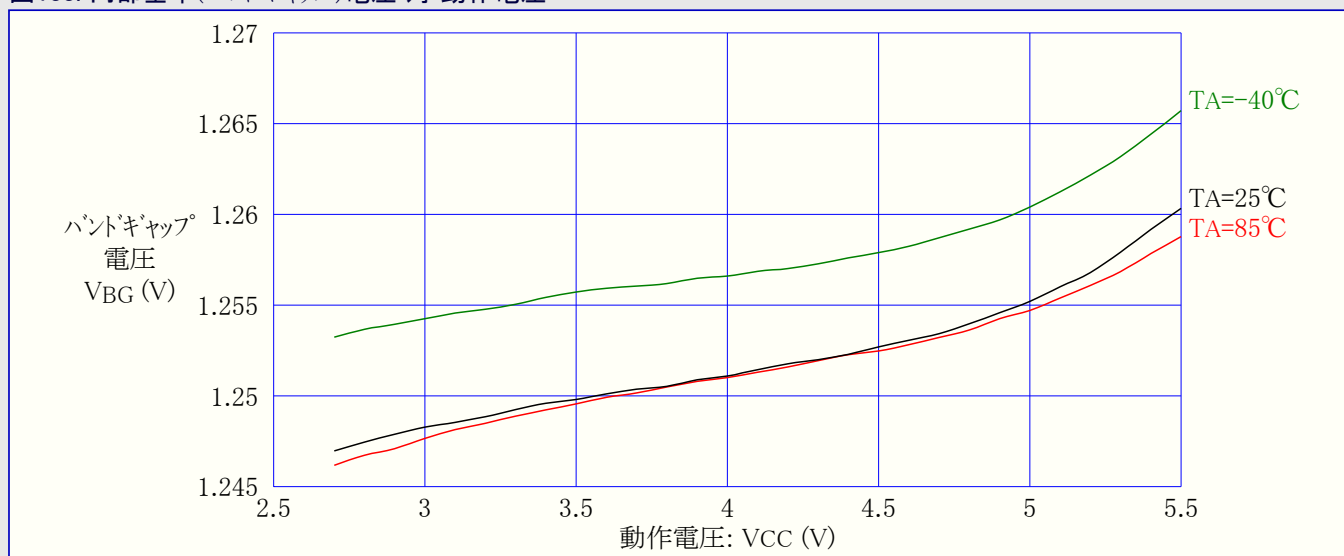


図134. アナログ比較器オフセット電圧 対 同相入力電圧 (VCC=5V)

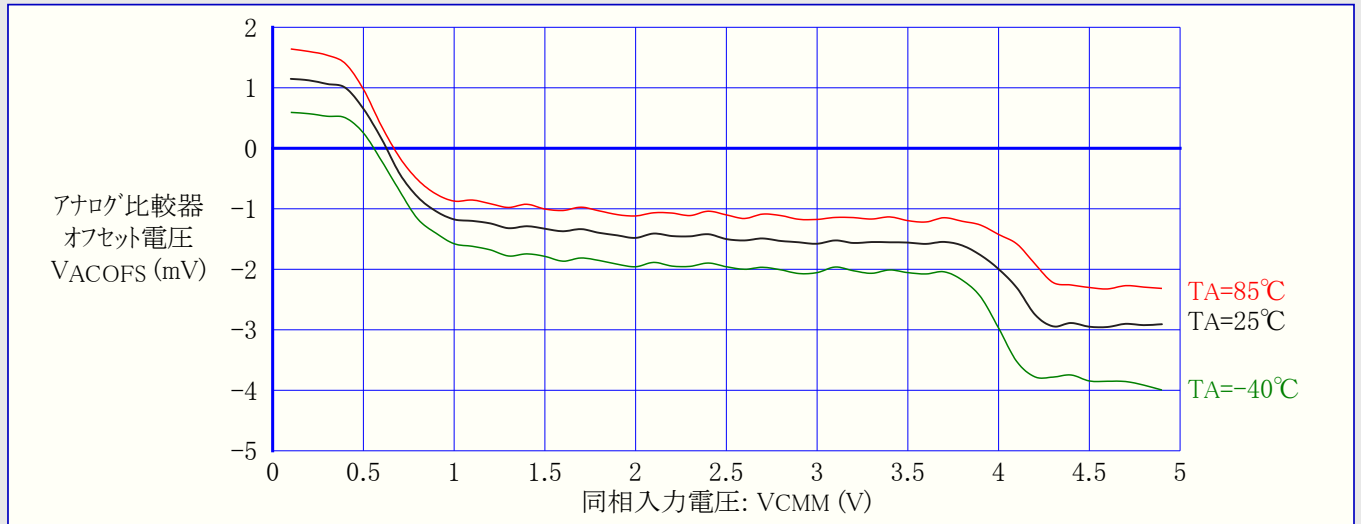
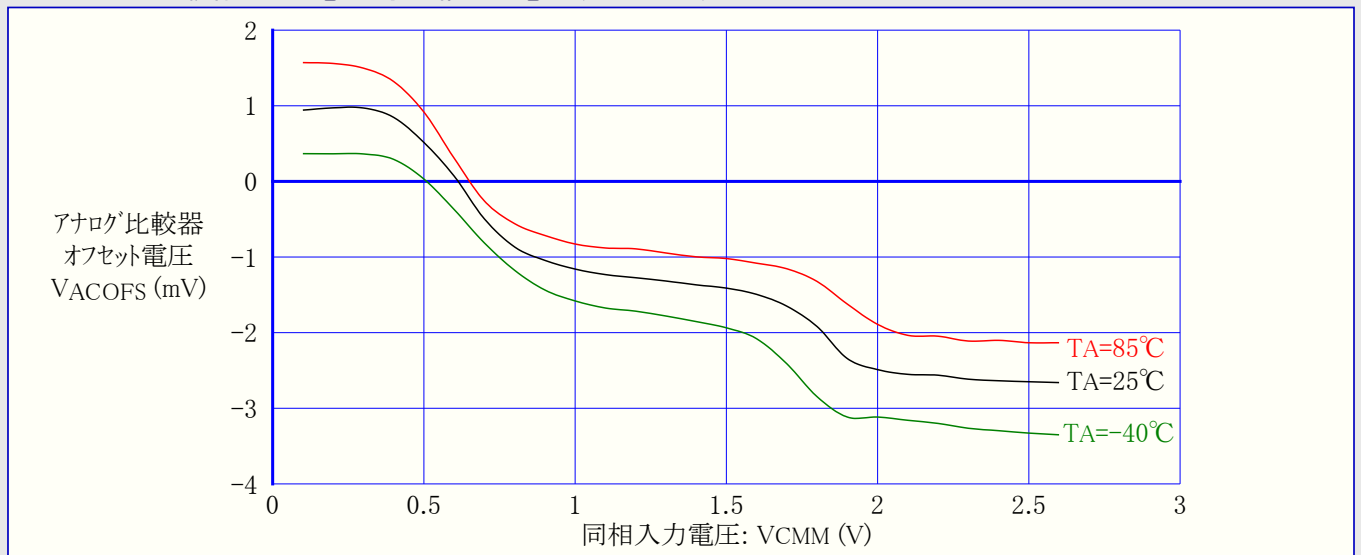


図135. アナログ比較器オフセット電圧 対 同相入力電圧 (VCC=2.7V)



内部発振器周波数

図136. ウォッチドッグ発振器周波数 対 動作温度

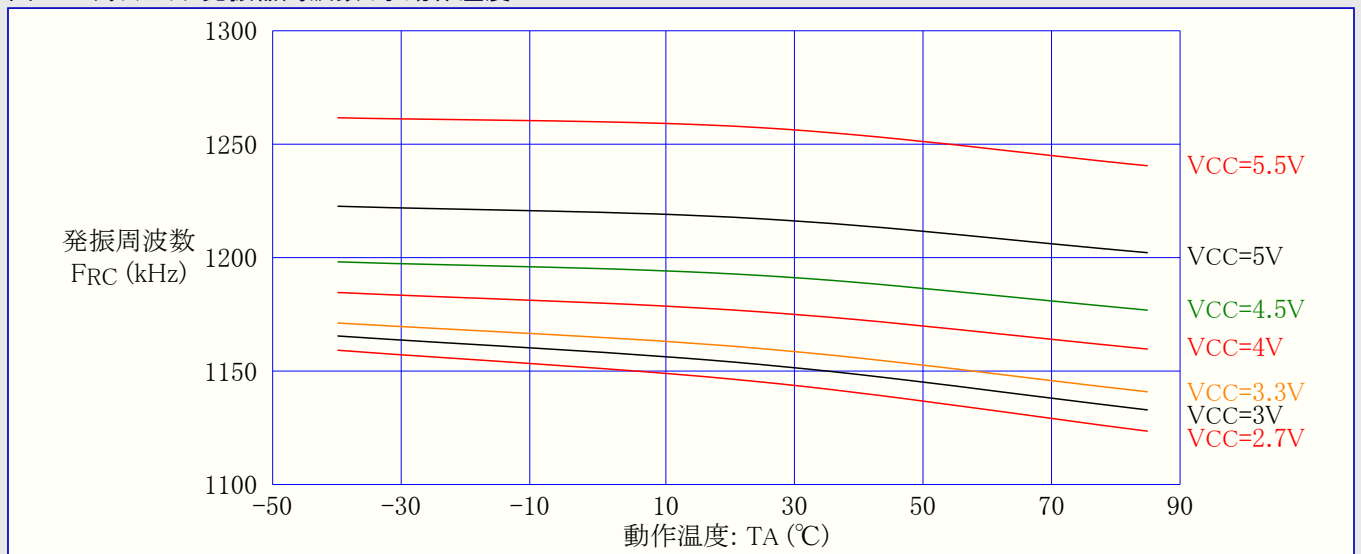


図137. ウォッチドッグ発振器周波数 対 動作電圧

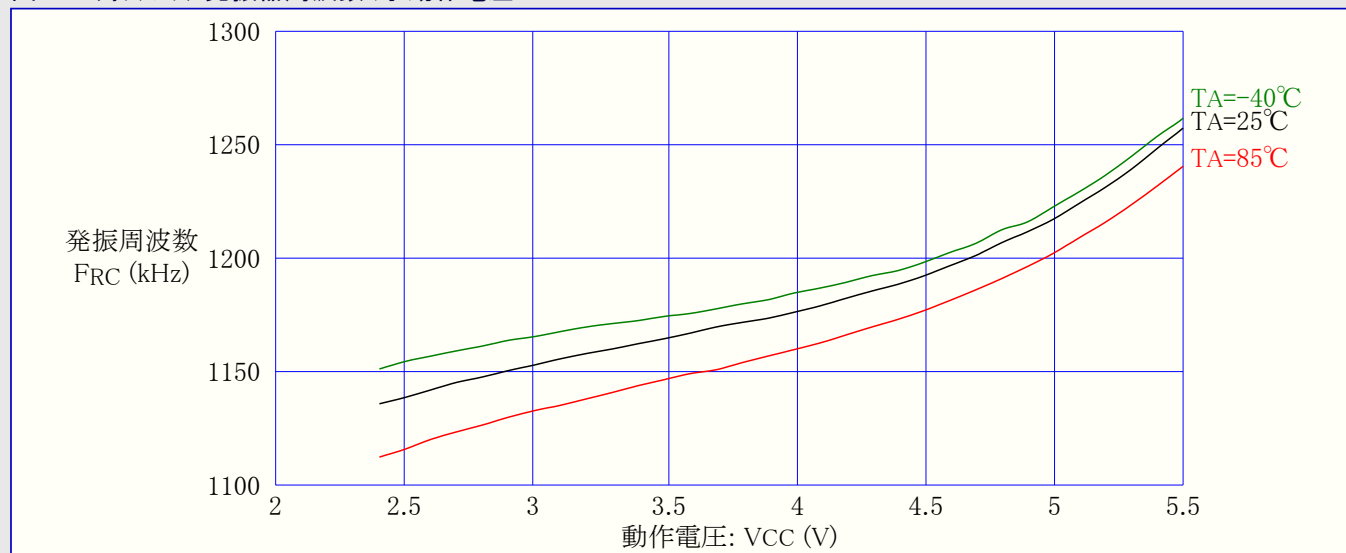


図138. 校正済み8MHz内蔵RC発振器周波数 対 動作温度

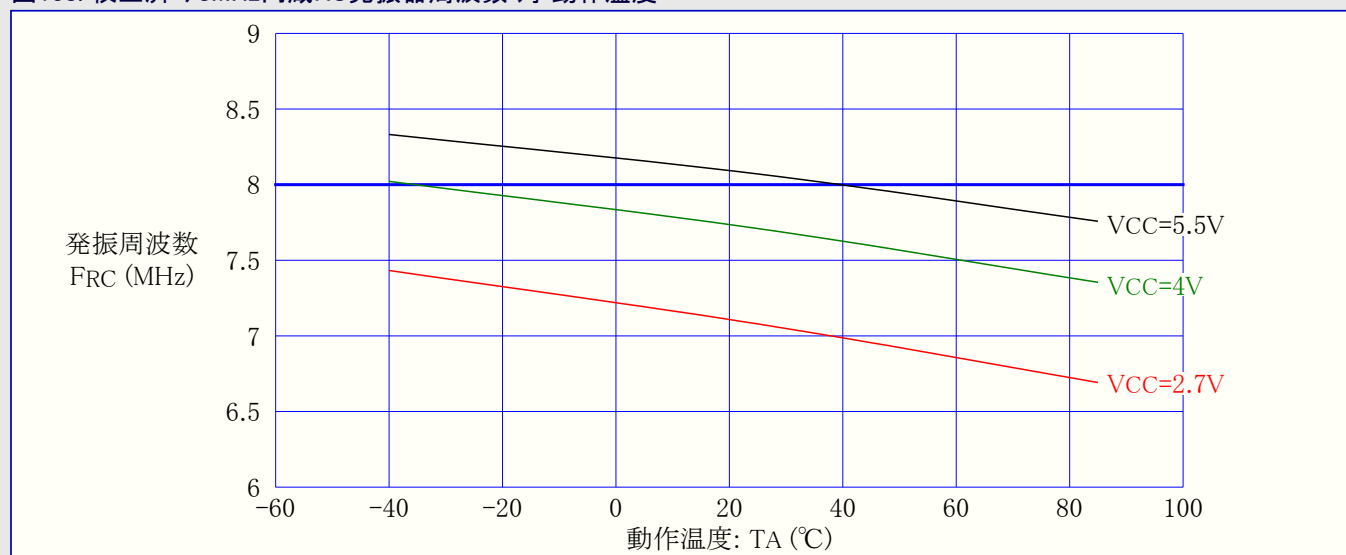


図139. 校正済み8MHz内蔵RC発振器周波数 対 動作電圧

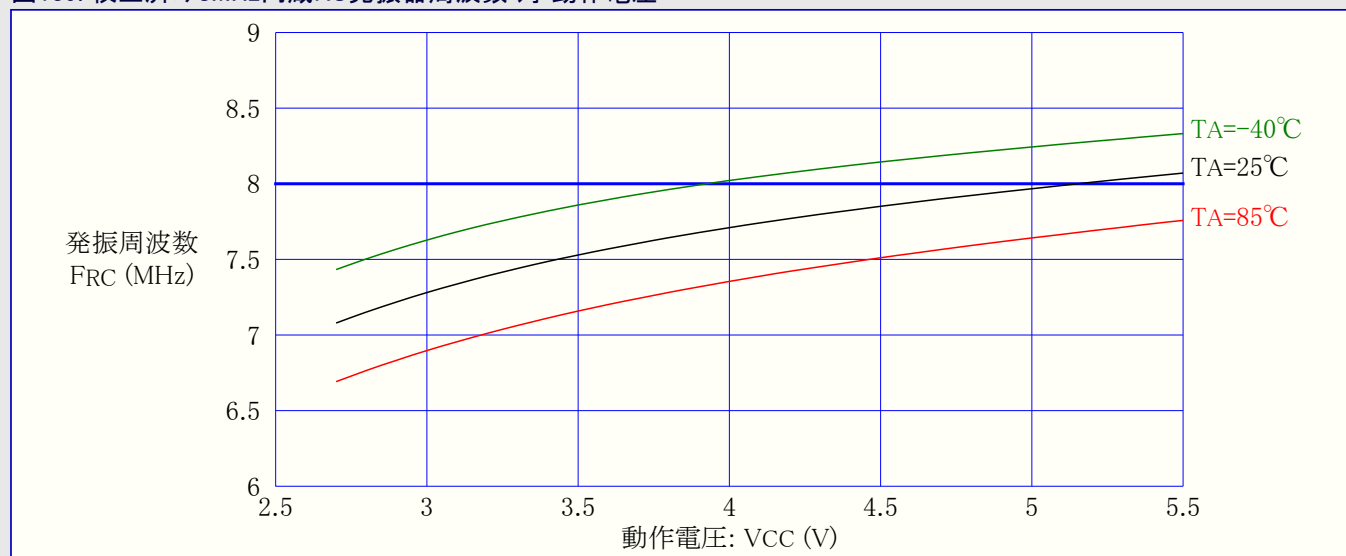


図140. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値

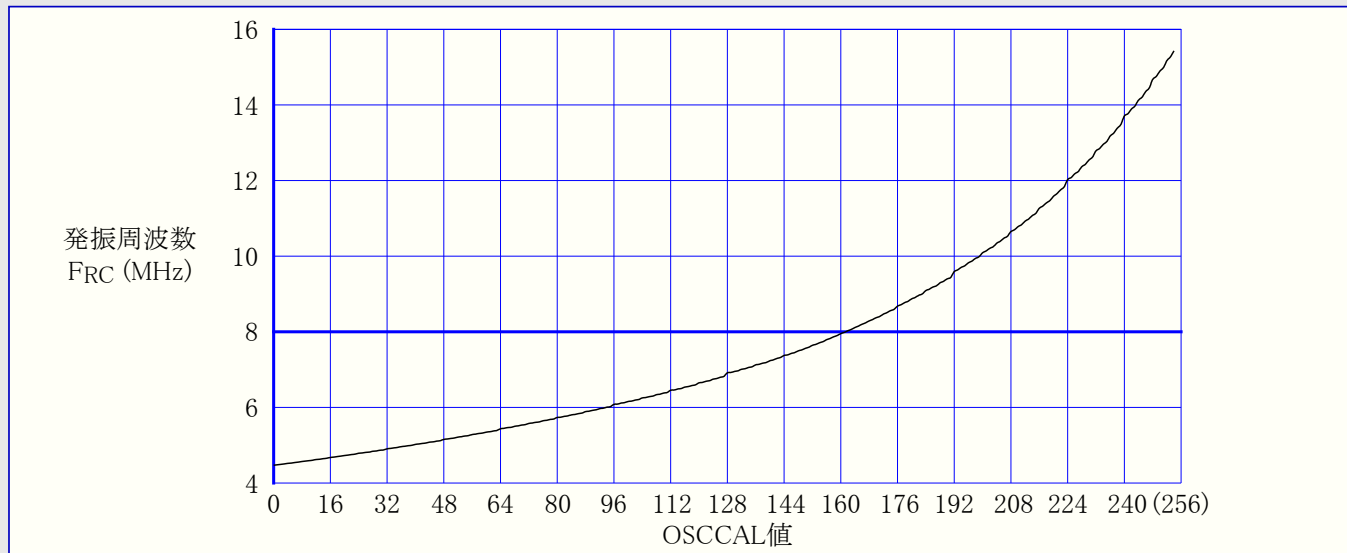


図141. 校正済み4MHz内蔵RC発振器周波数 対 動作温度

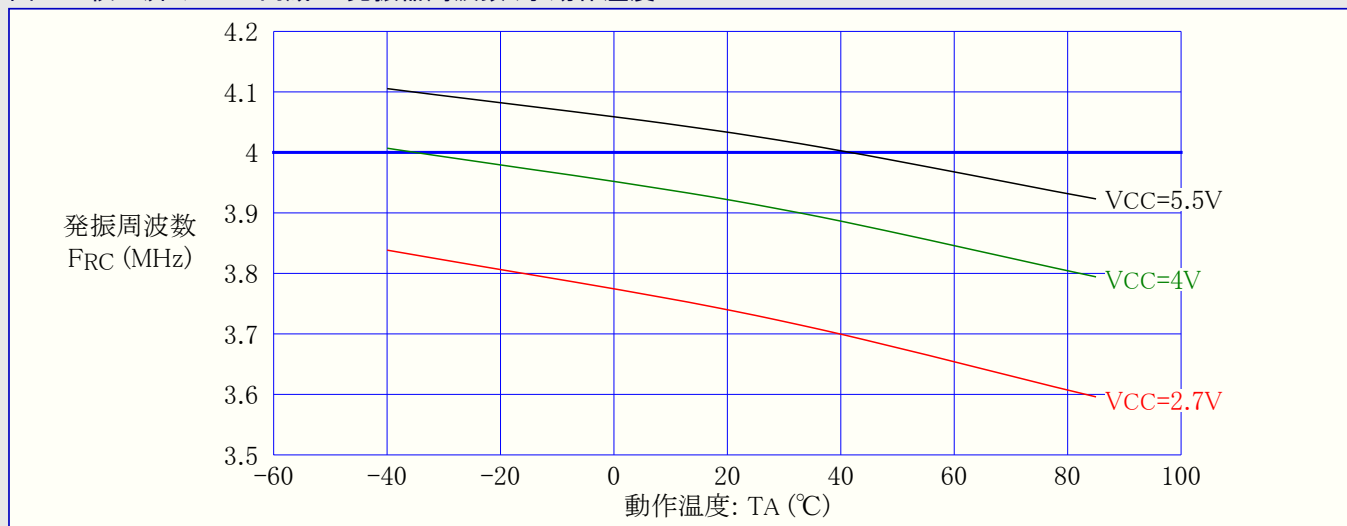


図142. 校正済み4MHz内蔵RC発振器周波数 対 動作電圧

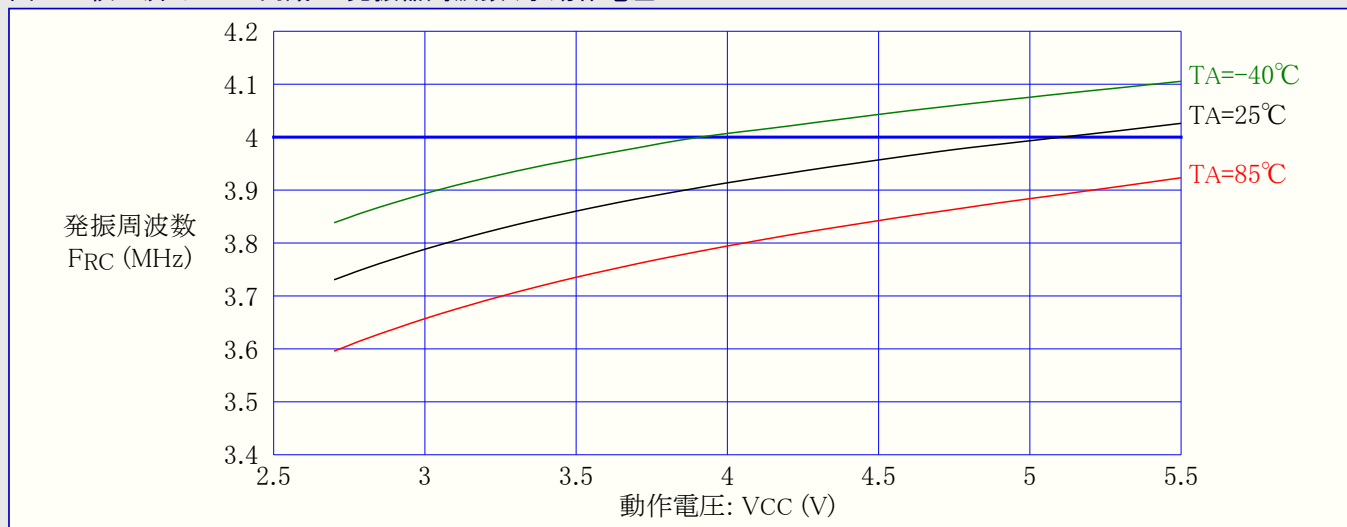


図143. 校正付き4MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値

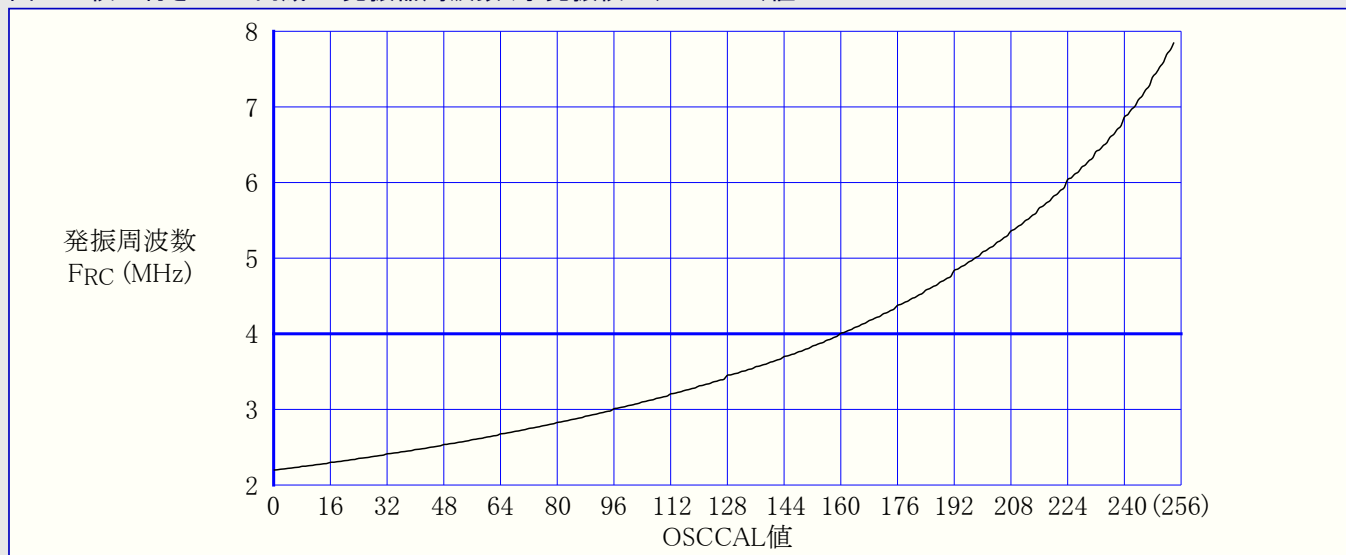


図144. 校正済み2MHz内蔵RC発振器周波数 対 動作温度

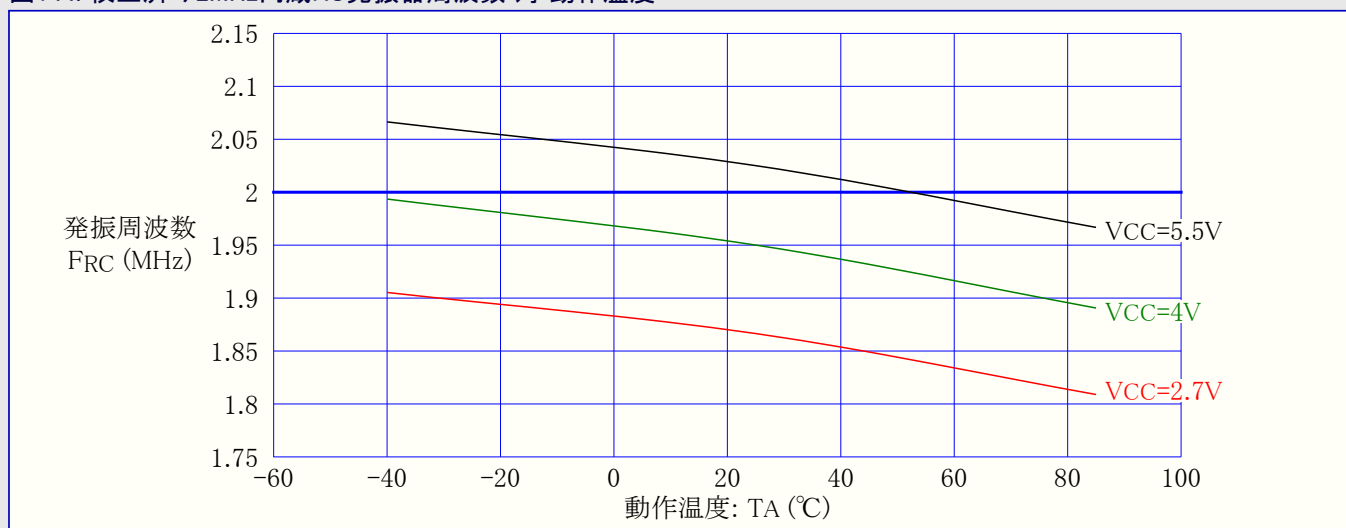


図145. 校正済み2MHz内蔵RC発振器周波数 対 動作電圧

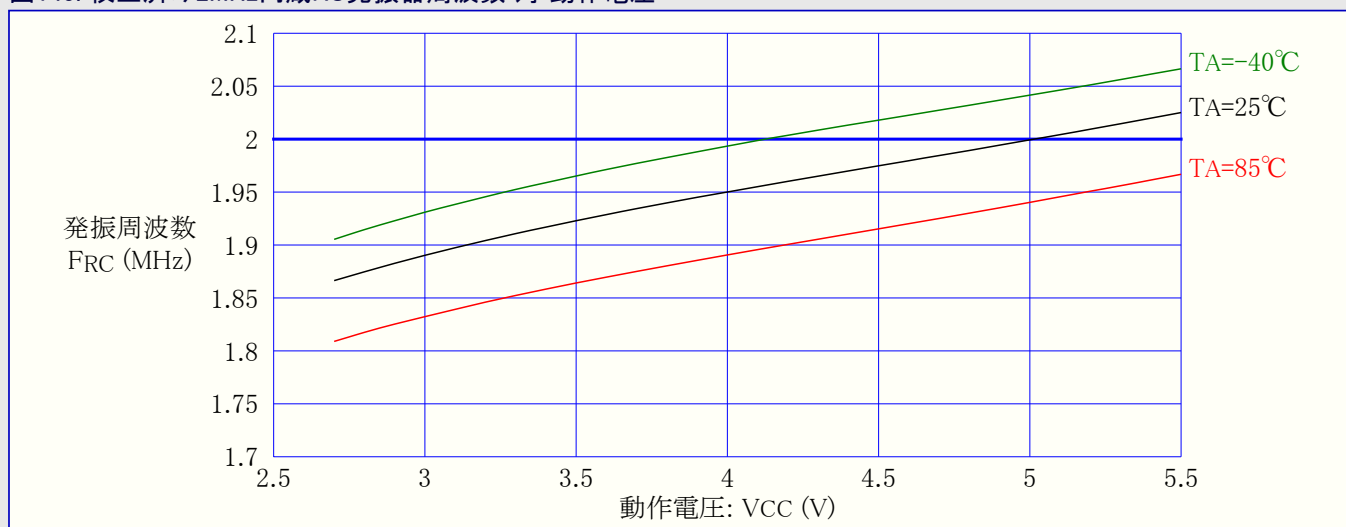


図146. 校正付き2MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値

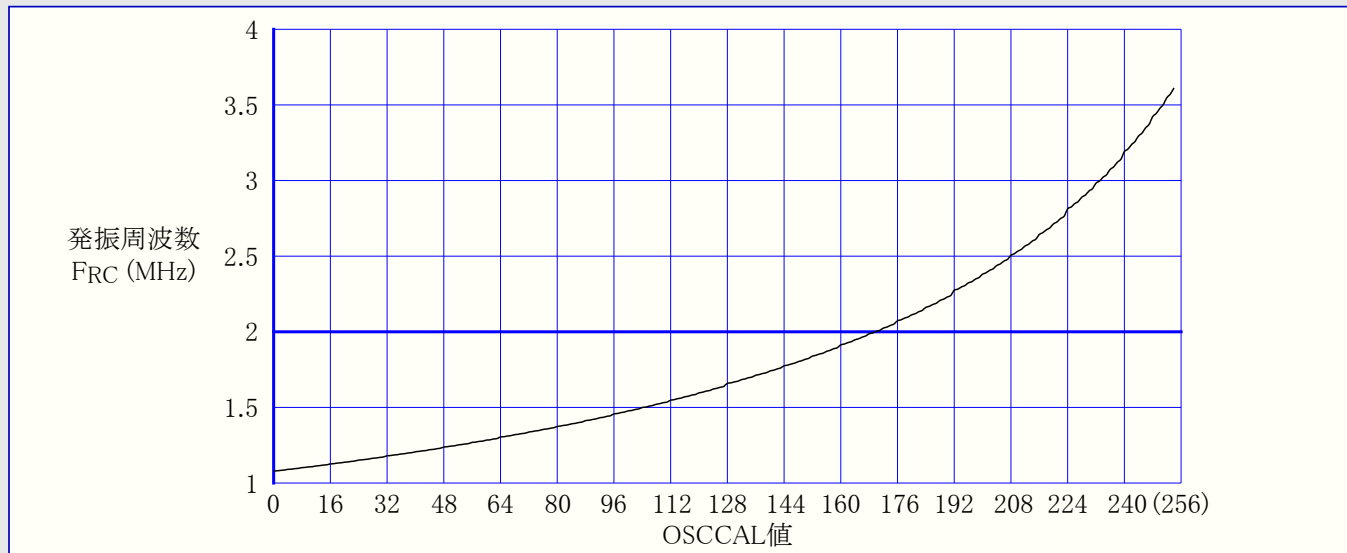


図147. 校正済み1MHz内蔵RC発振器周波数 対 動作温度

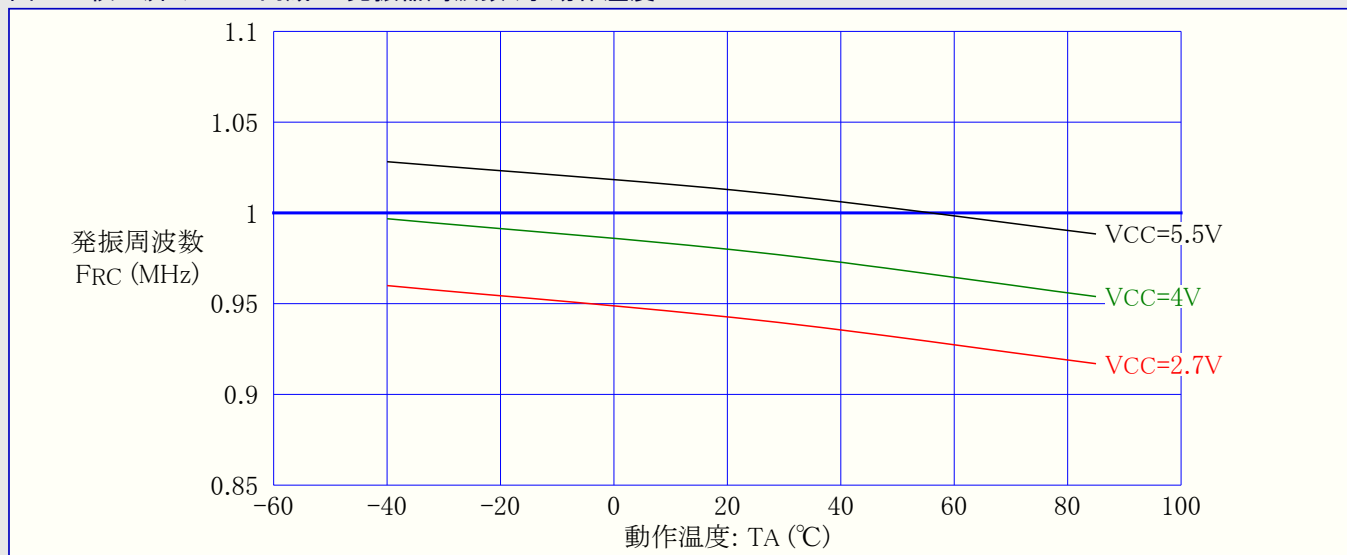


図148. 校正済み1MHz内蔵RC発振器周波数 対 動作電圧

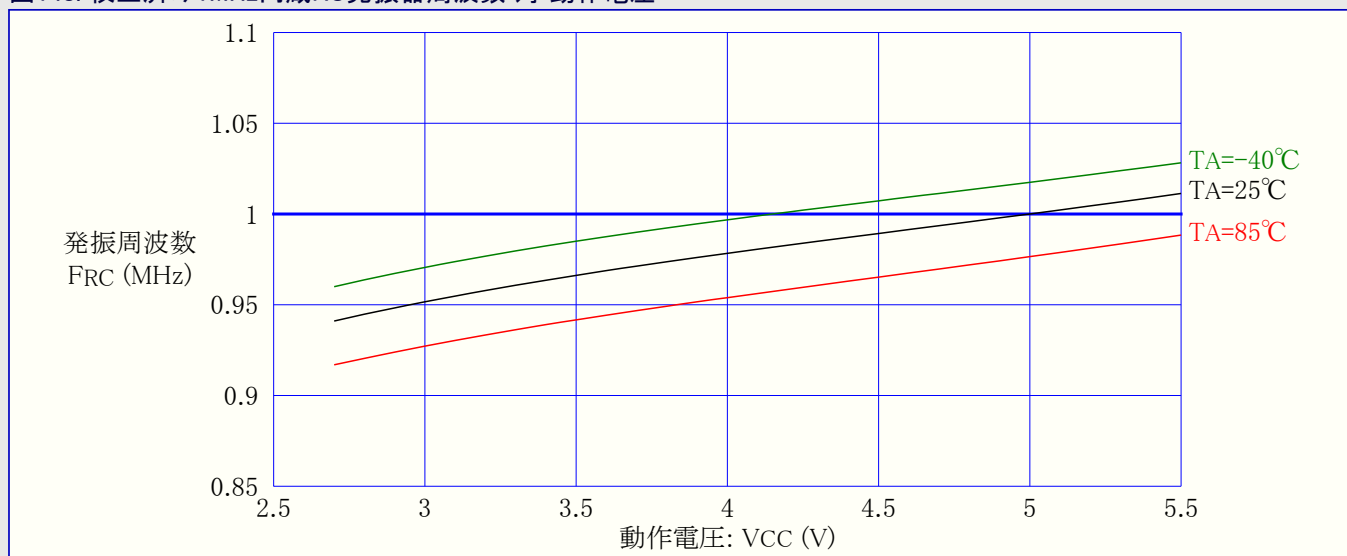
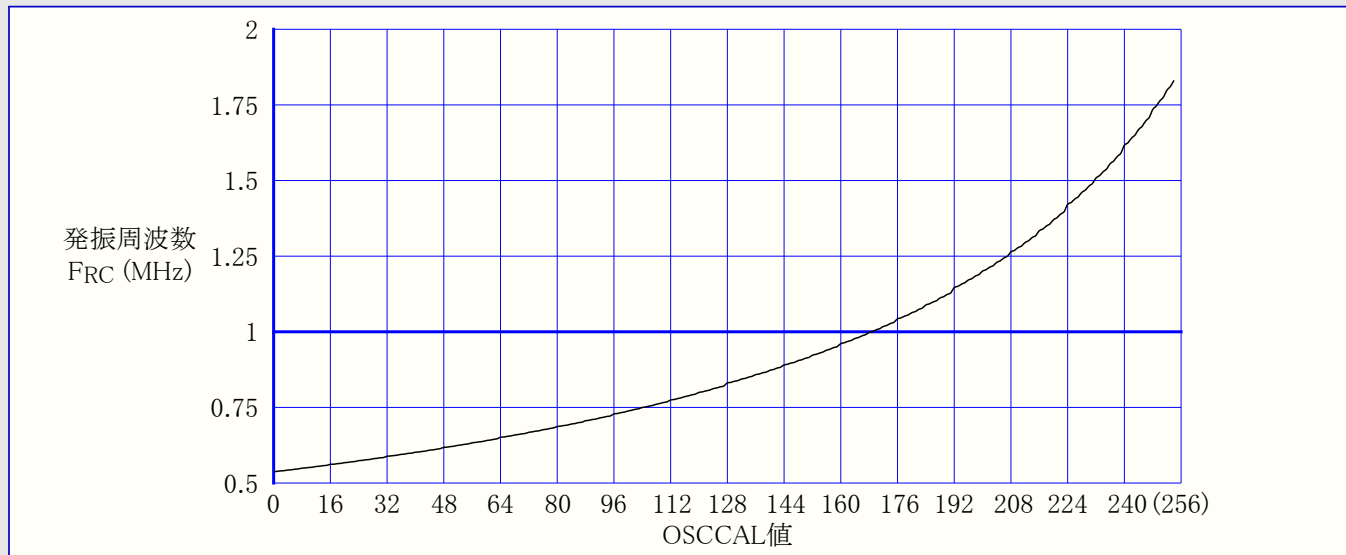


図149. 校正付き1MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値



周辺機能部消費電流

図150. 7ナング比較器消費電流 対 動作電圧

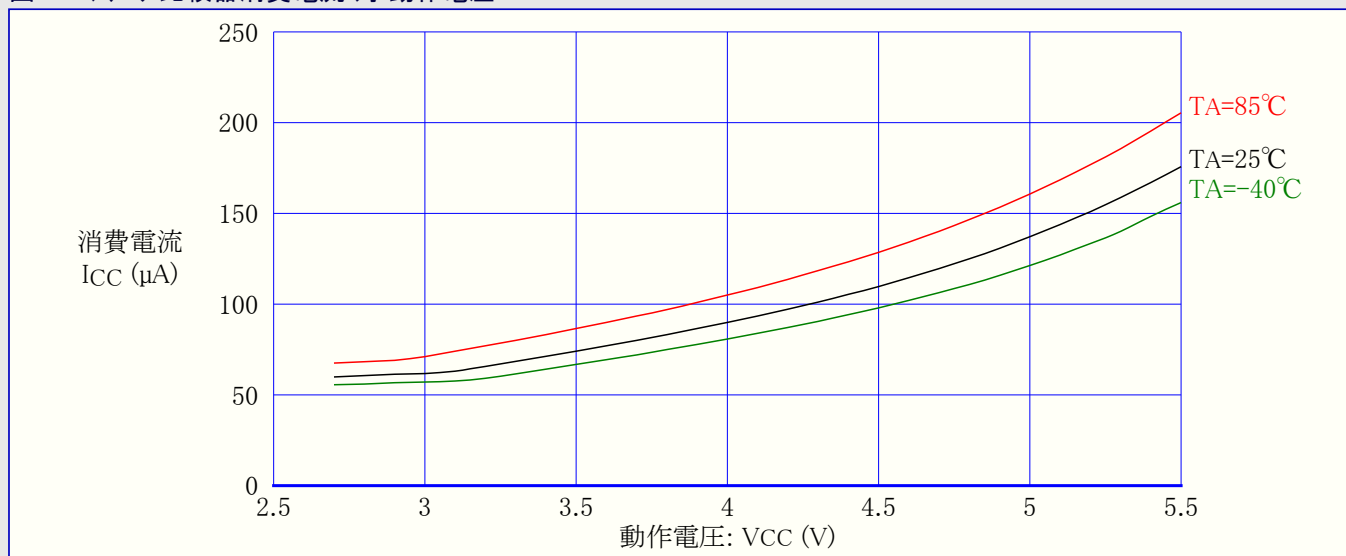


図151. 低電圧検出器(BOD)消費電流 対 動作電圧

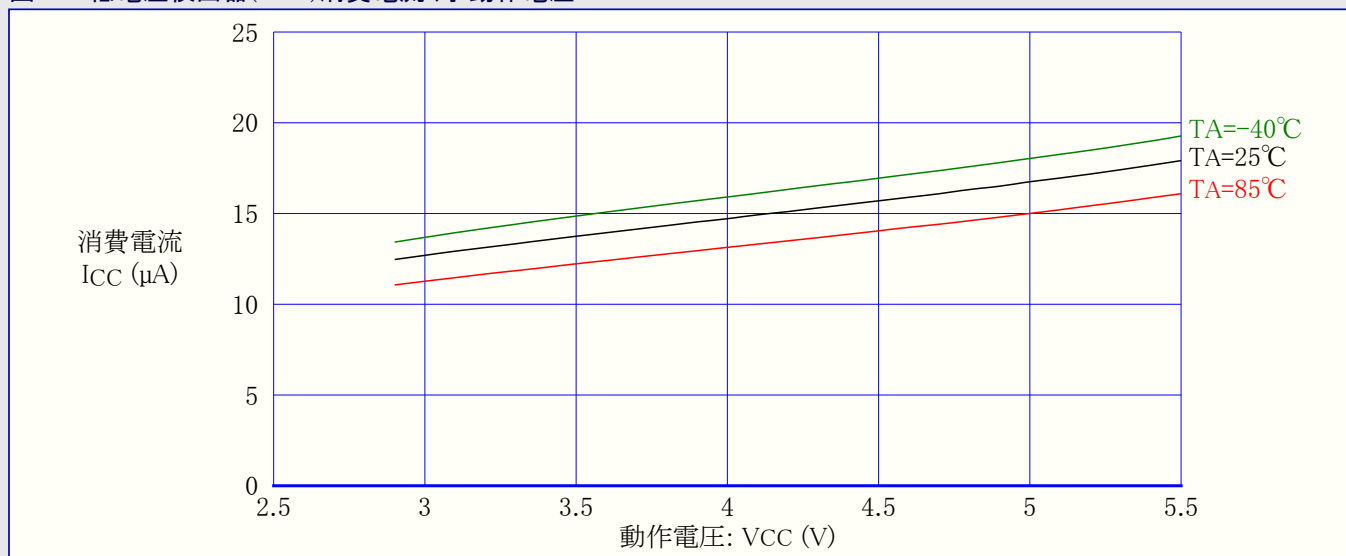
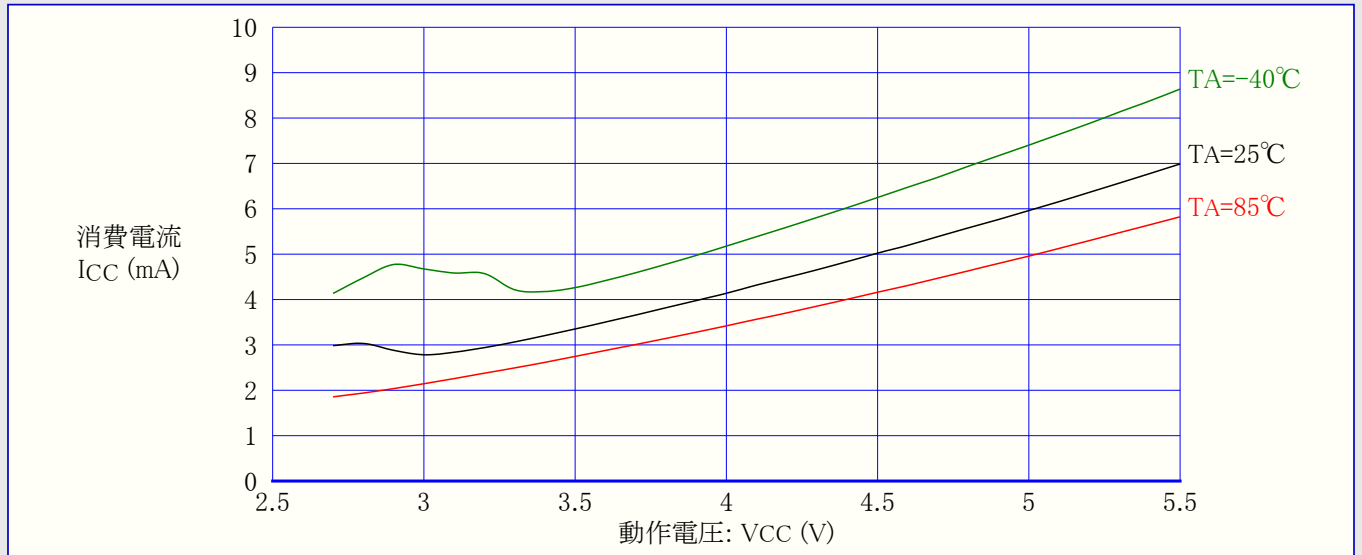


図152. プログラミング電流 対 動作電圧



リセット消費電流とリセットパルス幅

図153. 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)

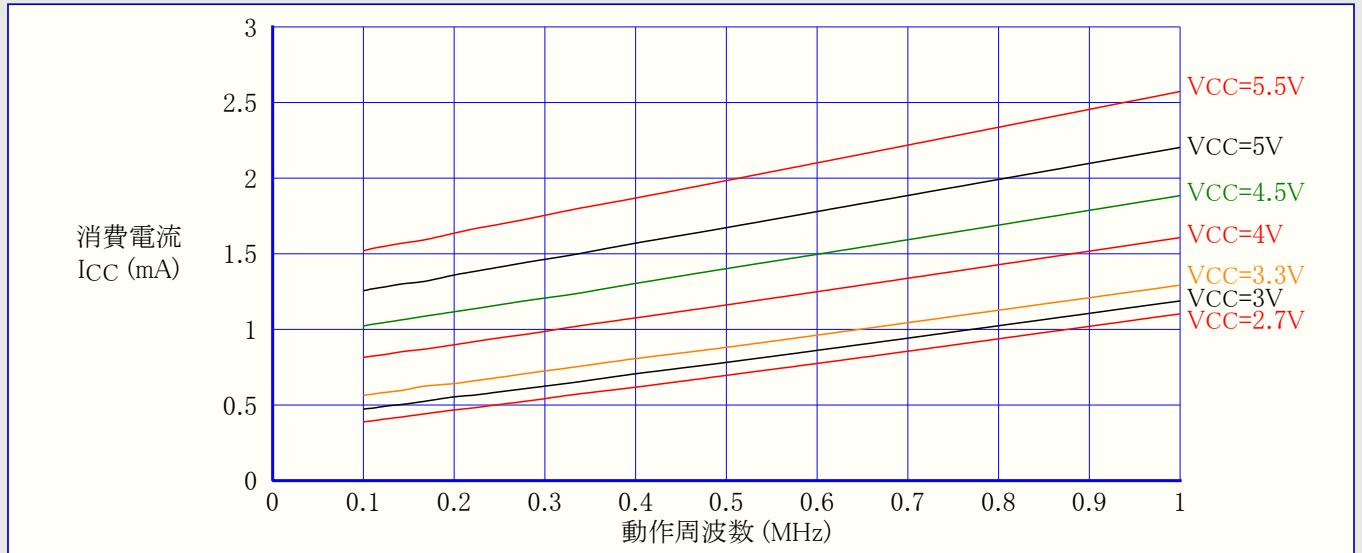


図154. リセット消費(供給)電流 対 周波数 (1MHz～20MHz、RESETプルアップ電流を除く)

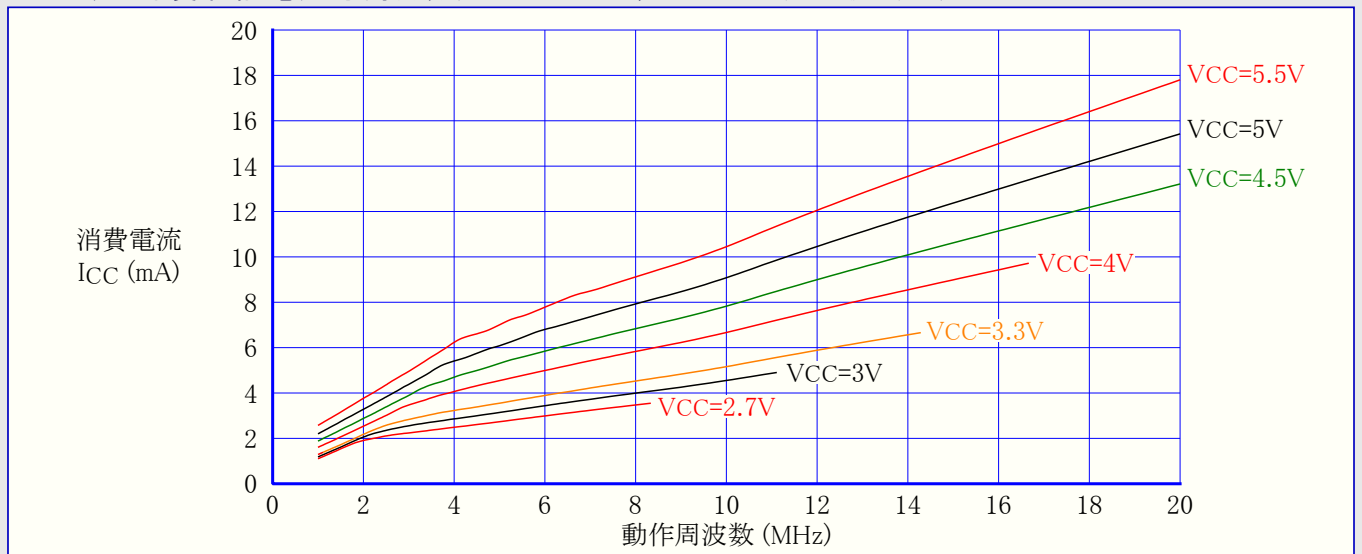
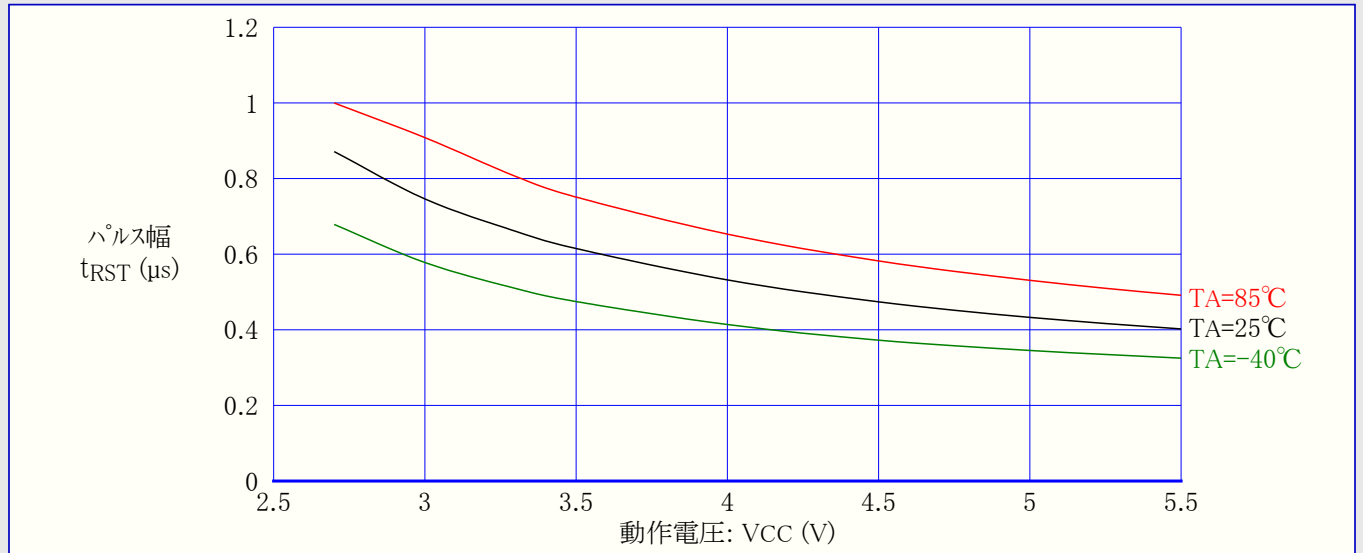


図155. リセット パルス幅 対 動作電圧



レジスタ要約

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	6
\$3E (\$5E)	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	7
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	
\$3C (\$5C)	予約									
\$3B (\$5B)	GICR	INT1	INT0	INT2	-	-	-	IVSEL	IVCE	50,35
\$3A (\$5A)	GIFR	INTF1	INTF0	INTF2	-	-	-	-	-	50
\$39 (\$59)	TIMSK	TOIE1	OCIE1A	OCIE1B	-	TICIE1	-	TOIE0	OCIE0	82,61
\$38 (\$58)	TIFR	TOV1	OCF1A	OCF1B	-	ICF1	-	TOV0	OCF0	82,61
\$37 (\$57)	SPMCR	SPMIE	RWWSB	-	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	110
\$36 (\$56)	EMUCUCR	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	24,17,49
\$35 (\$55)	MCUCR	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	17,24,49
\$34 (\$54)	MCUCSR	-	-	SM2	-	WDRF	BORF	EXTRF	PORF	24,29
\$33 (\$53)	TCCR0	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	59
\$32 (\$52)	TCNT0	タイマ/カウンタ0								60
\$31 (\$51)	OCR0	タイマ/カウンタ0 比較レジスタ								60
\$30 (\$50)	SFIOR	-	XMBK	XMM2	XMM1	XMM0	PUD	-	PSR10	18,40,63
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	78
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	80
\$2D (\$4D)	TCNT1H	タイマ/カウンタ1 上位バイト								80
\$2C (\$4C)	TCNT1L	タイマ/カウンタ1 下位バイト								
\$2B (\$4B)	OCR1AH	タイマ/カウンタ1 比較Aレジスタ 上位バイト								
\$2A (\$4A)	OCR1AL	タイマ/カウンタ1 比較Aレジスタ 下位バイト								81
\$29 (\$49)	OCR1BH	タイマ/カウンタ1 比較Bレジスタ 上位バイト								81
\$28 (\$48)	OCR1BL	タイマ/カウンタ1 比較Bレジスタ 下位バイト								
\$27 (\$47)	予約									
\$26 (\$46)	予約									81
\$25 (\$45)	ICR1H	タイマ/カウンタ1 捕獲レジスタ 上位バイト								
\$24 (\$44)	ICR1L	タイマ/カウンタ1 捕獲レジスタ 下位バイト								
\$23 (\$43)	予約									30
\$22 (\$42)	予約									
\$21 (\$41)	WDTCSR	-	-	-	WDCE	WDE	WDP2	WDP1	WDP0	
\$20 (\$40)	UBRRH	URSEL	-	-	-	USARTボーレートレジスタ 上位 (UBRR11~8)				103
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	102
\$1F (\$3F)	EEARH	-	-	-	-	-	-	-	EEAR8	11
\$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								
\$1C (\$3C)	EECR	-	-	-	-	EERIE	EEMWE	EWE	EERE	11
\$1B (\$3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	47
\$1A (\$3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	47
\$19 (\$39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	47
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	47
\$17 (\$37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	47
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	47
\$15 (\$35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	47
\$14 (\$34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	47
\$13 (\$33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	48
\$12 (\$32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	48
\$11 (\$31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	48
\$10 (\$30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	48
\$0F (\$2F)	SPDR	SPI データ レジスタ								86
\$0E (\$2E)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	86
\$0D (\$2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	85
\$0C (\$2C)	UDR	USART データ レジスタ								101
\$0B (\$2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	101
\$0A (\$2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	102
\$09 (\$29)	UBRRL	USARTボーレートレジスタ 下位バイト (UBRR7~0)								103
\$08 (\$28)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	106
\$07 (\$27)	PORTE	-	-	-	-	-	PORTE2	PORTE1	PORTE0	48
\$06 (\$26)	DDRE	-	-	-	-	-	DDE2	DDE1	DDE0	48
\$05 (\$25)	PINE	-	-	-	-	-	PINE2	PINE1	PINE0	48
\$04 (\$24)	OSCCAL	内蔵RC発振器 発振校正値レジスタ								23
\$03 (\$23)	予約									
\$02 (\$22)	予約									
\$01 (\$21)	予約									
\$00 (\$20)	予約									

注:・UBRRHとUCSRCのアクセス方法の詳細については「USART」記述を参照してください。

- 将来のデバイスとの共通性のため、予約ビットへ書く場合は0を書くべきです。予約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。
- いくつかの状態ビットはそれらに論理1を書くことによって解除(0)されます。CBI,SBI命令はそのI/Oレジスタ内の全ビットを操作し、設定(1)として読んだどのフラグにも1を書き戻します。従ってフラグを解除(0)します。CBI,SBI命令は\$00~\$1FのI/Oレジスタだけで動作します。

命令要約

ニーモニック	オペランド	意味	動作	フラグ	クロック
算術、論理演算命令					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,C	1
NEG	Rd	2の補数	$Rd \leftarrow \$00 - Rd$	I,T,H,S,V,N,Z,C	1
SBR	Rd,K	汎用レジスタの(複数)ビット設定(1)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
CBR	Rd,K	汎用レジスタの(複数)ビット解除(0)	$Rd \leftarrow Rd \text{ AND } (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レジスタの増加(+1)	$Rd \leftarrow Rd + 1$	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	$Rd \leftarrow Rd - 1$	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レジスタのゼロとマイナス検査	$Rd \leftarrow Rd \text{ AND } Rd$	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レジスタの全0設定(=\$00)	$Rd \leftarrow Rd \text{ EOR } Rd$	I,T,H,0,0,0,I,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	$Rd \leftarrow \$FF$	I,T,H,S,V,N,Z,C	1
MUL	Rd,Rr	符号なし間の乗算	$R1:R0 \leftarrow Rd \times Rr \quad (U \times U)$	I,T,H,S,V,N,Z,C	2
MULS	Rd,Rr	符号付き間の乗算	$R1:R0 \leftarrow Rd \times Rr \quad (S \times S)$	I,T,H,S,V,N,Z,C	2
MULSU	Rd,Rr	符号付きと符号なしの乗算	$R1:R0 \leftarrow Rd \times Rr \quad (S \times U)$	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	$R1:R0 \leftarrow (Rd \times Rr) \ll 1 \quad (U \times U)$	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	$R1:R0 \leftarrow (Rd \times Rr) \ll 1 \quad (S \times S)$	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	$R1:R0 \leftarrow (Rd \times Rr) \ll 1 \quad (S \times U)$	I,T,H,S,V,N,Z,C	2
分岐命令					
RJMP	k	相対無条件分岐	$PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	2
IJMP		Zレジスタ間接無条件分岐	$PC \leftarrow Z$	I,T,H,S,V,N,Z,C	2
RCALL	k	相対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	3
ICALL		Zレジスタ間接サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow Z$	I,T,H,S,V,N,Z,C	3
RET		サブルーチンからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
RETI		割り込みからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
CPSE	Rd,Rr	汎用レジスタ間比較、一致でスキップ	$Rd=Rr$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	$Rd - Rr$	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	$Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	$Rd - K$	I,T,H,S,V,N,Z,C	1
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ	$Rr(b)=0$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBR	Rr,b	汎用レジスタのビットが設定(1)でスキップ	$Rr(b)=1$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ	$P(b)=0$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
SBSI	P,b	I/Oレジスタのビットが設定(1)でスキップ	$P(b)=1$ なら, $PC \leftarrow PC + 2 \text{ or } 3$	I,T,H,S,V,N,Z,C	1/2,3
BRBS	s,k	ステータスフラグが設定(1)で分岐	$SREG(s)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRBC	s,k	ステータスフラグが解除(0)で分岐	$SREG(s)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BREQ	k	一致で分岐	$Z=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRNE	k	不一致で分岐	$Z=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCS	k	キャリーフラグが設定(1)で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCC	k	キャリーフラグが解除(0)で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRSH	k	符号なしの $\geq$ で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしの $<$ で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	$N=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	$N=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの $\geq$ で分岐	$(N \text{ EOR } V)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLT	k	符号付きの $<$ で分岐	$(N \text{ EOR } V)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	$H=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	$H=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	$T=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	$T=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	$V=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	$V=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	$I=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	$I=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ
b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)

ニーモニック	オペランド	意味	動作	フラグ	クロック
データ移動命令					
MOV	Rd,Rr	汎用レジスタ間の複写	$Rd \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
MOVW	Rd,Rr	汎用レジスタ対間の複写	$Rd+1:Rd \leftarrow Rr+1:Rr$	I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	$Rd \leftarrow K$	I,T,H,S,V,N,Z,C	1
LD	Rd,X	Xレジスタ間接での取得	$Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,X+	事後増加付きXレジスタ間接での取得	$Rd \leftarrow (X), X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-X	事前減少付きXレジスタ間接での取得	$X \leftarrow X - 1, Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y	Yレジスタ間接での取得	$Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上 (任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上 (事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
ビット関係命令					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	1,1,1,1,1,1,1,1	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,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)	$T \leftarrow 0$	0,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
MCU制御命令					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1

注文情報

速度(MHz)	電源電圧	注文コード*	外囲器	動作範囲
8	2.7~5.5V	ATmega8515L-8AC	44A	一般用 (0°C~70°C)
		ATmega8515L-8PC	40P6	
		ATmega8515L-8JC	44J	
		ATmega8515L-8MC	44M1	
		ATmega8515L-8AI	44A	工業用 (-40°C~85°C)
		ATmega8515L-8AU (注2)	44A	
		ATmega8515L-8PI	40P6	
		ATmega8515L-8PU (注2)	40P6	
		ATmega8515L-8JI	44J	
		ATmega8515L-8JU (注2)	44J	
		ATmega8515L-8MI	44M1	
		ATmega8515L-8MU (注2)	44M1	
16	4.5~5.5V	ATmega8515-16AC	44A	一般用 (0°C~70°C)
		ATmega8515-16PC	40P6	
		ATmega8515-16J	44J	
		ATmega8515-16MC	44M1	
		ATmega8515-16AI	44A	工業用 (-40°C~85°C)
		ATmega8515-16AU (注2)	44A	
		ATmega8515-16PI	40P6	
		ATmega8515-16PU (注2)	40P6	
		ATmega8515-16JI	44J	
		ATmega8515-16JU (注2)	44J	
		ATmega8515-16MI	44M1	
		ATmega8515-16MU (注2)	44M1	

注: このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

注2: 有害物質使用制限に関する欧州指令(RoHS指令)適合の鉛フリー製品。またハロゲン化合物フリーで完全に安全です。

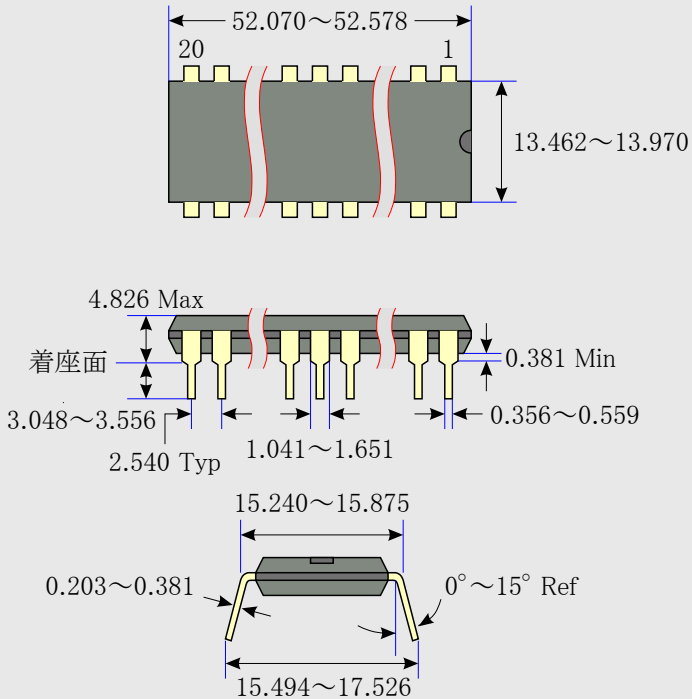
外囲器形式	
44A	44リード 1.0mm厚 プラスティック4方向平板外囲器 (TQFP)
40P6	40ピン 600mil幅 プラスティック2列直線外囲器 (PDIP)
44J	44リード プラスティック Jリード チップ搬送体 (PLCC)
44M1	44パッド 7×7×1mm 0.5mmピッチ 4方向平板リードなし/小リード枠外囲器 (QFN/MLF)

外周器情報

40P6

40ピン 600mil幅 プラスチック2列直線外周器 (PDIP)

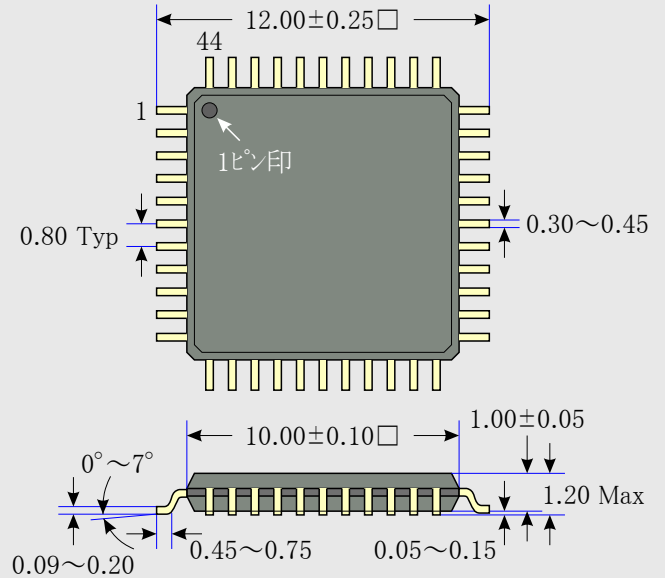
寸法: mm
JEDEC規格 MS-011 AC



44A

44リード 0.8mmピッチ プラスチック4方向平板外周器 (TQFP)

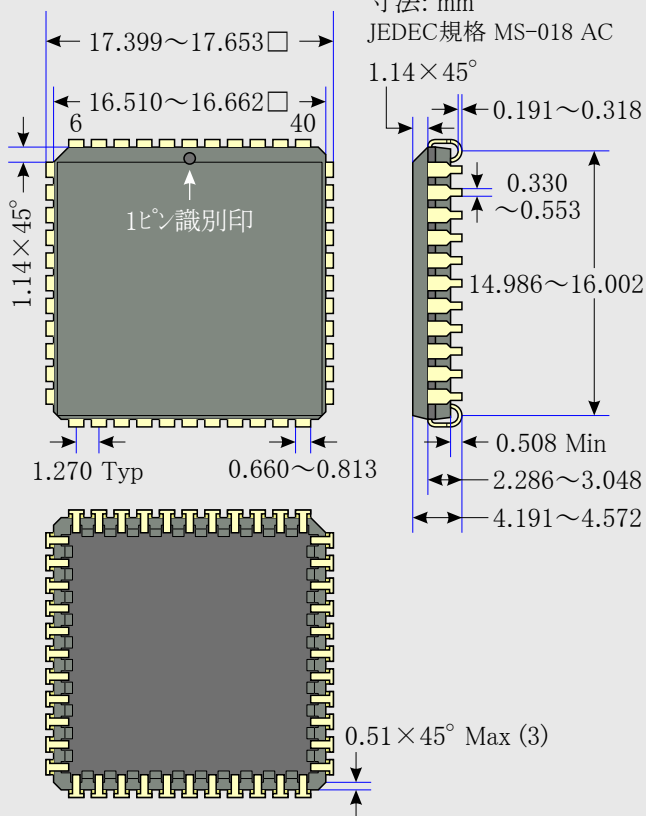
寸法: mm
JEDEC規格 MS-026 ACB



44J

44リード プラスチック Jリード チップ搬送体 (PLCC)

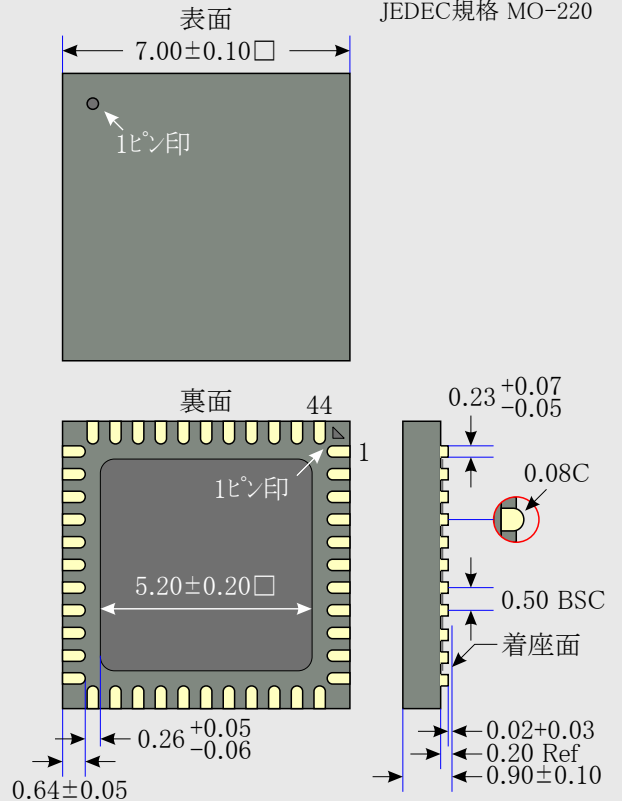
寸法: mm
JEDEC規格 MS-018 AC



44M1

44パッド 0.5mmピッチ 4方向平板リードなし/小リード枠外周器 (QFN/MLF)

寸法: mm
JEDEC規格 MO-220



障害情報

この章の改訂番号はATmega8515デバイスの改訂版を参照してください。

改訂C/D

- ・ 初回アナログ比較器変換が遅らされるかもしれない

C/D

1. 初回アナログ比較器変換が遅らされるかもしれない (C/D)

デバイスが低速上昇VCCで給電される場合、最初のアナログ比較器変換はいくつかのデバイスで予想よりも長くなるでしょう。

対策/対処

デバイスが給電またはリセットされた時は最初の変換前にアナログ比較器を禁止し、そして許可してください。

データシート改訂履歴

この章内の参照頁番号はこの資料が参照されていることに注意してください。この章内の改訂番号は資料の改訂番号を参照してください。

改訂2512A – 2002年4月

1. 初版

改訂2512B – 2002年9月

1. フラッシュメモリの書き換え可能回数(寿命)を1,000から10,000に変更

改訂2512C – 2002年10月

1. 18頁の「64Kバイト未満外部メモリの全域使用法」章の追加
2. 全TBDの削除
3. 2,4,8MHz発振校正値についての説明追加
4. 23頁の「外部クロック信号」に周波数変動を追加
5. 27頁の表18.にV_{BOT}についての注意追加
6. 38頁の「未接続ピン」についてを追加
7. タイマ/カウンタ1について、78頁の表51.と表52.を修正
8. 119頁の「プログラミング動作への移行」と「チップ消去」、121頁の図77.と図78.を更新
9. 128頁の「DC特性」、129頁「外部クロック」の表95.と表96.、130頁「SPIタイミング特性」の表97.を更新
10. 162頁の「障害情報」を追加

改訂2512D – 2003年2月

1. 13頁に「パワーダウン休止動作中のEEPROM書き込み」項を追加
2. 57頁の「位相基準PWM動作」の記述を改善
3. 74頁の図53.でOCn波形を修正
4. 112頁の「ページ一時緩衝部の設定 (ページ設定)」にSPMページ設定中のEEPROM書き込みについての注意を追加
5. 127頁の表93.を更新
6. 160頁の「外圍器情報」を更新

改訂2512E – 2003年9月

1. 「暫定」を削除
2. 27頁の表18.、128頁「電気的特性」の「絶対最大定格」と「DC特性」を更新
3. 135頁からの「代表特性」を更新

改訂2512F – 2003年12月

1. 22頁の「校正付きRC発振器」を更新

改訂2512G – 2005年3月

1. MLF外圍器を4方向平板リードなし/小リード枠外圍器QFN/MLFに代替変更
2. 128頁の「電気的特性」を更新
3. 159頁の「注文情報」を更新

改訂2512H – 2006年4月

1. 4頁で「資料」を追加
2. 75頁の「位相基準PWM動作」参照表番号を更新
3. 82頁の「タイマ/カウンタ割り込み許可レジスタ (TIMSK)」を更新
4. 83頁の「直列周辺インターフェース (SPI)」を更新
5. 117頁で「校正値バイト」の破棄項を削除
6. 22頁の表10.、78頁の表52.、127頁の表94.、129頁の表96.を更新

改訂2512I – 2006年8月

1. 159頁の「注文情報」に注文コードを追加

改訂2512J – 2006年8月

1. 全タイマ/カウンタの高速PWM動作についてTOP/BOTTOM記述を更新
2. 162頁の「障害情報」を更新

改訂2512K – 2010年1月

1. 27頁の表18.でリセット特性を更新

目次

特徴	1	動作種別	55
ピン配置	1	タイマ/カウンタのタイミング	58
概要	2	8ビット タイマ/カウンタ0用レジスタ	59
構成図	2	タイマ/カウンタ0と1の前置分周器	62
お断り	2	16ビット タイマ/カウンタ1	64
ATmega8515とAT90S4414/8515の互換性	3	概要	64
ピン説明	4	16ビットレジスタのアクセス	66
資料	4	タイマ/カウンタのクロック	68
コード例について	4	計数器部	68
AVR CPU コア	5	捕獲入力部	69
序説	5	比較出力部	70
構造概要	5	比較一致出力部	72
ALU (Arithmetic Logic Unit)	5	動作種別	73
ステータスレジスタ	6	タイマ/カウンタのタイミング	77
汎用レジスタファイル	7	16ビット タイマ/カウンタ1用レジスタ	78
スタックポインタ	7	SPI (直列周辺インターフェース)	83
命令実行タイミング	8	SSピンの機能	85
リセットと割り込みの扱い	8	データ転送形式	87
AVR ATmega8515のメモリ	9	USART	88
実装書き換え可能なプログラム用フラッシュメモリ	9	単一USART	88
データ用SRAMメモリ	10	概要	88
データ用EEPROMメモリ	11	クロック生成	90
I/Oメモリ (レジスタ)	14	フレーム形式	91
外部メモリ インターフェース	14	USARTの初期化	92
システムクロックとクロック選択	20	データ送信 - USART送信部	93
クロック系統とその配給	20	データ受信 - USART受信部	94
クロック元	20	非同期受信	97
既定のクロック元	20	複数プロセッサ通信動作	99
クリスタル用発振器	21	UBRRH/UCSRCLレジスタのアクセス	100
低周波数クリスタル用発振器	21	USART用レジスタ	101
外部RC発振器	22	ボーレート設定例	104
校正付き内蔵RC発振器	22	アナログ比較器	106
外部クロック信号	23	ブートローダ支援 - RWW自己プログラミング	107
電力管理と休止形態	24	特徴	107
アイドル動作	25	フラッシュメモリの応用領域とブートローダ領域	107
パワーダウン動作	25	書き中に読みが可能な領域と不可能な領域	107
スタンバイ動作	25	ブートローダ施錠ビット	109
消費電力の最小化	26	ブートローダプログラムへの移行	109
システム制御とリセット	27	自己プログラミングでのフラッシュアドレス指定	111
内部基準電圧	29	フラッシュメモリの自己プログラミング	111
ウォッチドッグタイマ	30	メモリプログラミング	116
ウォッチドッグタイマ設定変更の時間制限手順	31	プログラムメモリとデータメモリ用施錠ビット	116
割り込み	32	ヒューズビット	117
ATmega8515の割り込みベクタ	32	識別バイト	117
入出力ポート	36	校正バイト	117
序説	36	並列プログラミング	118
標準デジタル入出力としてのポート	36	並列プログラミング手順	119
交換ポート機能	39	直列プログラミング	126
I/Oポート用レジスタ	47	直列プログラミング手順	126
外部割り込み	49	電気的特性	128
8ビットタイマ/カウンタ0 (PWM)	51	絶対最大定格	128
概要	51	DC特性	128
タイマ/カウンタのクロック	52	外部クロック特性	129
計数器部	52	SPIタイミング特性	130
比較出力部	53	データ用外部メモリタイミング特性	131
比較一致出力部	54	代表特性	134
		活動動作消費電流	134

アイドル動作消費電流	136
パワーダウン動作消費電流	139
スタンバイ動作消費電流	139
ピン プルアップ	142
ピン駆動能力	143
ピン 閾値とヒステリシス	145
低電圧検出器(BOD)閾値と アナログ 比較器オフセット	147
内部発振器周波数	148
周辺機能部消費電流	153
リセット消費電流とリセット パルス幅	154
レジスタ要約	156
命令要約	157
注文情報	159
外圍器情報	160
障害情報	161
データシート改訂履歴	162



本社

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131
USA
TEL 1(408) 441-0311
FAX 1(408) 487-2600

国外営業拠点

Atmel Asia

Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
TEL (852) 2245-6100
FAX (852) 2722-1369

Atmel Europe

Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
TEL (33) 1-30-60-70-00
FAX (33) 1-30-60-71-11

Atmel Japan

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

製品窓口

ウェブサイト

www.atmel.com

技術支援

avr@atmel.com

販売窓口

www.atmel.com/contacts

文献請求

www.atmel.com/literature

お断り: 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイト位置する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえばAtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益の損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© Atmel Corporation 2010. 不許複製 Atmel®、ロゴとそれらの組み合わせ、その他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

© HERO 2021.

本データシートはAtmelのATmega8515英語版データシート(改訂2512K-01/10)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力は、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。