

8Kバイト実装書き込み可能フラッシュ メモリ付き Atmel 8ビット マイクロ コントローラ

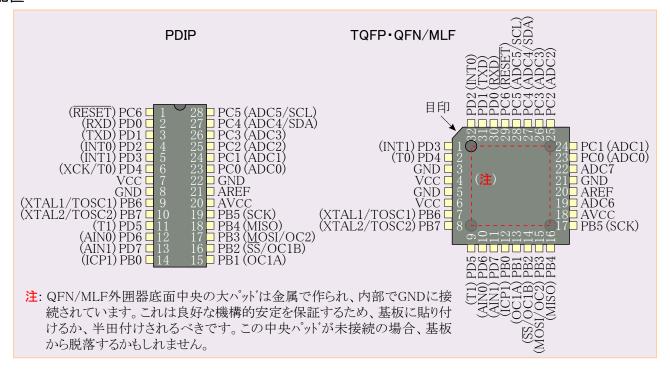
データシート

特徴

- 高性能、低消費AVR® 8ビット マイクロ コントローラ
- ・進化したRISC構造
 - 強力な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
 - 完全なスタティック動作
 - 16MHz時、16MIPSに達する高速動作
 - 2周期乗算命令
- 高耐久不揮発性メモリ部
 - 実装自己書き換え可能な8Kバイト(4K語)フラッシュ メモリ内蔵
 - 512バイトのEEPROM
 - 1Kバイトの内蔵SRAM
 - 書き換え回数: 10,000/フラッシュ, 100,000/EEPROM
 - データ保持力: 20年/85℃, 100年/25℃
 - 個別施錠ビットを持つ任意のブート コード領域
 - チップ 内ブート プログラムによる実装書き換え
 - 真の書き込み中の読み出し動作
 - ソフトウェア保護用の設定可能な施錠機能
- 内蔵周辺機能
 - ●独立した前置分周器付き2つの8ビットタイマ/カウンタ(1つは比較機能付き)
 - 独立した前置分周器、比較、捕獲機能付き1つの16ビットタイマ/カウンタ
 - 専用発振器と8ビット タイマ/カウンタによる実時間計数器(RTC)
 - 3つのPWM出力
 - 6チャネル(PDIP), 8チャネル(TQFP,QFN/MLF) 10ビット A/D変換器
 - バイト対応の2線直列インターフェース
 - 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
 - アイト・ル、A/D変換雑音低減、パワーセーブ、パワータ・ウン、スタンハイの5つの低消費動作
- I/Oと外囲器
 - 23ビットの設定可能なI/O
 - 28ピンPDIP、32リードTQFP、32パッドQFN/MLF
- 動作電圧
 - 2.7~5.5V (ATmega8L)
 - 4.5∼5.5V (ATmega8)
- 動作速度
 - 0~8MHz (ATmega8L)
 - 0~16MHz (ATmega8)
- 消費電流 (条件: 4MHz,3V,25℃)
 - 活動動作 ・・・・・・ 3.6mA
 - ア仆 ル動作 ・・・・・・ 1.0mA
 - n°ワータウン動作・・・・・ 0.5µA

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。

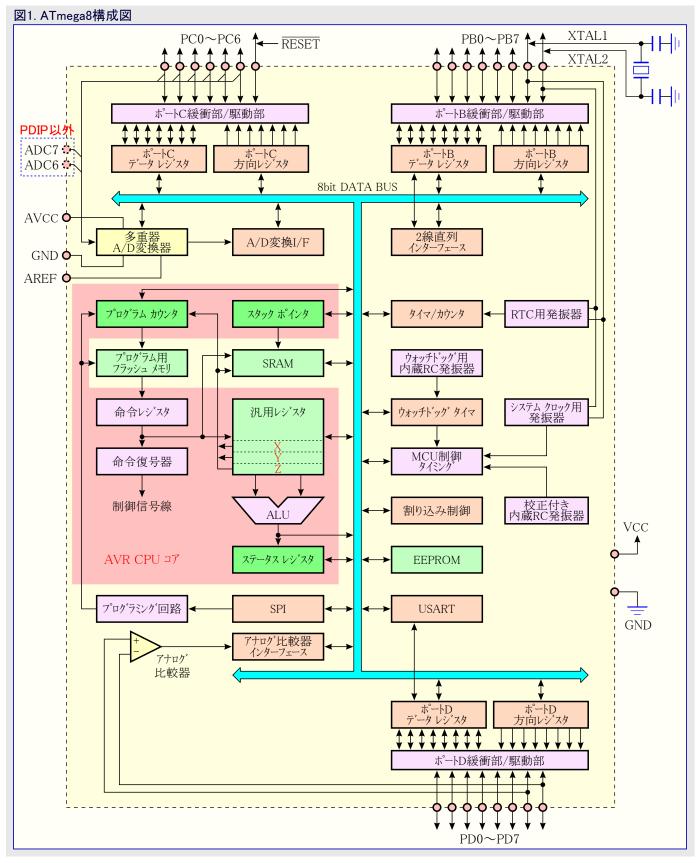
ピン配置





概要

Atmel® AVR® ATmega8はAVR RISC構造の低消費CMOS 8ビット マイクロコントローラです。1周期で実行する強力な命令はMHzあたり1 MIPSにも達し、実行速度対電力消費の最適化が容易に行えます。





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

ATmega8は書き込み中読み出し可能な能力を持つ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、1KバイトのSRAM、23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3つのタイマ/カウンタ、内部及び外部割り込み、設定可能な直列USART、バイト対応の2線直列インターフェース、6(PDIP),8(TQFP,QFN/MLF)チャネルの10ビットA/D変換器、設定可能な内部発振器付きウォッチドッグタイマ、SPI直列ポート、ソフトウェアで選べる5つの低消費動作機能を提供します。アイドル動作では動作を停止しますが、SRAM、タイマ/カウンタ、SPIポート、割り込み機能は有効で、動作を継続します。パワーダウン動作ではレジスタの内容は保護されますが、発振器が停止するため、以降のハードウェアリセットか外部割り込みまで他の全機能を禁止(無効に)します。パワーセーブ動作では非同期タイマ用発振器が動作を継続し、デバイスのその他が停止中であっても基準タイマの継続が許されます。A/D変換雑音低減動作ではA/D変換中の切り替え雑音を最小とするために非同期タイマとA/D変換器を除く周辺機能とCPUが停止します。スタンバイ動作ではクリスタル発振子/セラミック振動子用発振器が動作し、一方デバイスのその他は休止します。これは低消費電力と非常に速い起動の組み合わせを許します。

本デバイスはAtmelの高密度不揮発性メモリ技術を使って製造されています。内蔵の実装書き換え(ISP)可能なプログラム用フラッシュメモリは規定の不揮発性メモリ書き込み器、SPI直列インターフェース経由、AVRコア上のブートプログラムの実行によって再書き込みができます。ブートプログラムは応用領域フラッシュメモリ内の応用プログラムの読み込みに、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ソフトウェアは真の「書き込み中の読み出し可」動作によって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自己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 ATmega8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。

ATmega8はCコンパイラ、マクロアセンブラ、シミュレータ、評価キットを含む専用のプログラム及びシステム開発ツールで支援されます。

ピン概要

VCC

デジタル電源ピン。

GND

接地ピン。

PB7~PB0 (本一片B) XTAL1/XTAL2/TOSC1/TOSC2

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

クロック選択ヒューズ設定に依存し、PB6は発振器反転増幅器への入力や内部クロック操作回路の入力として使われます。

クロック選択ヒューズ設定に依存し、PB7は発振器反転増幅器からの出力として使われます。

校正付き内蔵RC発振器がチップ(システム)クロック元として使われる場合、非同期状態レシ、スタ(ASSR)の非同期動作(AS2)ピットが設定(1)されると、PB7,6は非同期タイマ/カウンタ2用のTOSC2,1ピンとして使われます。

ポートBの各特殊機能は36頁で詳しく述べられます。

PC5~PC0 (ポートC)

ポートCは(ビット毎に選ばれる)内蔵プルアップ抵抗付きの7ビットの双方向入出力ポートです。ポートC出力緩衝部は共に高い吐き出し/吸い込み能力の対称駆動特性です。入力のとき、プルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポート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き、クロックが動作していなくても、ポートCピンはHi-Zになります。

ポートCの各特殊機能は38頁で詳しく述べられます。

PC6/RESET

RSTDISBLヒューズがプログラム(0)されると、PC6はI/Oピンとして使われます。PC6の電気的特性がポートCの他のピンのそれらと異なることに注意してください。

RSTDISBLヒュース、が非プログラム(1)の場合、PC6はリセット入力として使われます。クロックが動作していなくても、最小パルス幅より長いこのピンのLowレベルはリセットを生成します。最小パルス幅は23頁の表15.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。

PD7~PD0 (ポートD)

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

ポートDは40頁で示されるATmega8の各特殊機能としても扱います。

(訳注) 原書に存在するRESET記述はPC6/RESET項目と重複するため省略しました。



AVCC

AVCCはADC7,6、ポートC(3~0)とA/D変換器用供給電圧(電源)ピンです。例えA/D変換が使われなくても、外部的にVCCへ接続されるべきです。A/D変換が使われる場合、VCCから低域通過濾波器を通して接続されるべきです。ポートC(5,4)がデジタル供給電圧(電源:VCC)を使うことに注意してください。

AREF

AREFはA/D変換器用アナログ基準(電圧)ピンです。

ADC7,6 (TQFP,QFN/MLFのみ)

TQFPとQFN/MLF外囲器でのADC7,ADC6はA/D変換器のアナログ入力として取り扱います。これらのピンはアナログ供給電源から電力供給され、10ビットA/Dチャネルとして扱われます。

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でのダウンロートで利用可能です。

データ保持力

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

コード例について

この文書はデバイスの様々な部分の使用法を手短に示す簡単なコート・例を含みます。これらのコート・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へッタ・ファイルがインクルート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タ・ファイル内にヒット定義を含めるとは限らず、またCでの割り込みの扱いがコンパイラに依存することに注意してください。より多くの詳細についてはCコンパイラの資料で確認してください。



AVR CPU 17

序説

ここでは一般的なAtmel® 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はメキモリ 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ト、構造を使います。プログラムメモリ内の命令は、単一段のハペ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つのオペラントがレジスタファイルからの出力で、1クロック周期内に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つのレシ、スタは効率的なアトンス計算ができるデータ空間アトンス指定用に3つの16ビット長間接アトンスポーインタ用レシ、スタとして使われます。これらアトンスポーインタの1つはプログラム用フラッシュメモリ内の定数用参照用アトンスポー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シ、スタはX,Y,Zレシ、スタで、本章内で後述されます。

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

プログラムの流れは条件/無条件分岐や呼び出し命令によって提供され、全アドレス空間を直接アドレス指定できます。AVR命令の多くは

16ビット語(ワード)形式です。全てのプログラム メモリのアドレスは(<mark>訳注</mark>: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、戻りアト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。全ての使用者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 ホーインタ(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つの異なるアトレス指定動作を通して容易にアクセスできます。

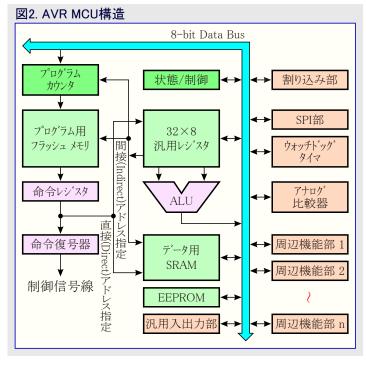
AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

柔軟な割り込み部にはI/O空間の各制御レシ、スタとステータスレシ、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へ、クタ表に個別の割り込みへ、クタを持ちます。割り込みには割り込みへ、クタ表の位置に従った優先順があります。下位側割り込みへ、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Atmel® AVR®のALUは32個の全汎用レシ、スタに直接接続され動作します。汎用レシ、スタ間または汎用レシ、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。





SREG - ステータス レシ スタ (Status Register)

ステータスレジスタは最も直前に実行した演算命令の結果についての情報を含みます。この情報は条件処理を行うためのプログラムの流れ変更に使えます。ステータスレジスタは「命令一式手引書」で詳述したように、全てのALU操作後に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 レシ、スタ(SREG)は次のように定義されます。

ピット	7	6	5	4	3	2	1	0	_
\$3F (\$5F)	I	Т	Н	S	V	N	Z	С	SREG
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	

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

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

● ビット6 - T: ビット変数 (Bit Copy Storage)

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

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

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

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

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

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

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

ビット2 - N: 負フラク (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 フラク (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。

ヒット0 − C : キャリー フラク (Carry Flag)

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



汎用レジスタファイル

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

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

図3.はCPU内の32個の汎用作業レジスタの構造を示します。

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

図3.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トレスも割り当てられます。例え物理的にSRAM位置として実装されていなくてもX,Y,Zレジスタ(ポインタ)がレジスタファイル内のどのレジスタの指示にも設定できるように、このメモリ構成は非常に柔軟なレジスタのアクセスを提供します。

図3. AV	図3. AVR CPU 汎用レシブスタ構成図								
	7 0	アトレス							
	R0	\$00							
	R1	\$01							
	R2	\$02							
	?								
	R13	\$0D							
	R14	\$0E							
्रात इस	R15	\$0F							
汎用 レジスタ	R16	\$10							
ファイル	R17	\$11							
. , (₹								
	R26	\$1A \$1B Xレジスタ 下位バイト 上位バイト							
	R27	\$1B ハック 上位バイト							
	R28	\$1C VISCAR 下位バイト							
	R29	 \$1D 上位バイト 							
	R30	\$1E Zレジスタ 下位バイト							
	R31	\$1F としょう 上位バイト							
	R31	\$1F Zレシスタ 上位バイト							

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トレス指定用の16ビットアドレスポーインタです。3つのX,Y,Z間接アトレスレジスタは図4.で記載したように定義されます。

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

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

SPH,SPL (SP) - スタック ポーインタ (Stack Pointer)

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(<mark>訳注</mark>: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。

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

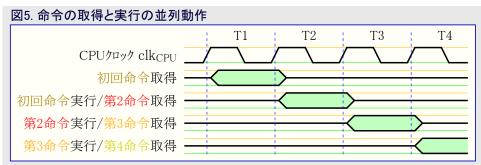


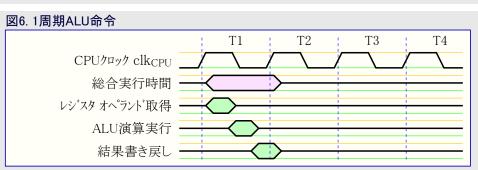
命令実行タイミング

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。Atmel® AVR® CPU はチップ(デバイス)用に選んだクロック元から直接的に生成したCPUクロック(clkCPU)によって 駆動されます。内部クロック分周は使われません。

図5.はハーハート・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、機能対電源部に対する好結果と対応するMHzあたり1 MIPSを達成するための基本的なパイプラインの概念です。

図6.はレシ、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。





リセットと割り込みの扱い

Atmel® AVR®は多くの異なる割り込み元を提供します。これらの割り込みと独立したリセット ベクタ各々はプログラム メモリ空間内に独立したプログラム ベクタを持ちます。全ての割り込みは割り込みを許可するために、ステータス レジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない個別の許可ビットを割り当てられます。BLB02またはBLB12 プート施錠ビットがプログラム(0)されると、プログラ ム カウンタ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144頁の「メモリ プログラミング」章をご覧ください。

既定でのプログラムメモリ空間の最下位アト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28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。下位側アト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の(INTO)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)ヒットの設定(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28頁の「割り込み」を参照してください。リセットベクタもBOOT RSTヒュース、のプログラム(0)によってブートフラッシュ領域先頭へ移動できます。135頁の「ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

割り込みが起こると全割り込み許可(I)ビットが解除(0)され、全ての割り込みは禁止されます。使用者ソフトウェアは多重割り込みを許可するために全割り込み許可(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(I)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へ、クタを指示し、ハードウェアが対応する割り込み要求フラグを解除(0)します。割り込み要求フラグは解除(0)されるべきフラグのビット位置へ論理1を書くことによっても解除(0)できます。対応する割り込み許可ビットが解除(0)されている間に割り込み条件が起こると、割り込み要求フラグが設定(1)され、割り込みが許可されるか、またはこのフラグがソフトウェアによって解除(0)されるまで記憶(保持)されます。同様に、全割り込み許可(I)ビットが解除(0)されている間に1つまたはより多くの割り込み条件が起こると、対応する割り込み要求フラグが設定(1)されて全割り込み許可(I)ビットが設定(1)されるまで記憶され、その(I=1)後で優先順に従って実行されます。

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

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

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



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

```
アセンブリ言語プログラム例
                                                    ;ステータスレジスタを保存
           IN
                 R16, SREG
                                                   ;EEPROM書き込み手順中割り込み禁止
           CLI
                 EECR, EEMWE
                                                   ;EEPROM主書き込み許可
           SBI
                 EECR, EEWE
                                                   ;EEPROM書き込み開始
           SBI
           OUT
                 SREG, R16
                                                   ;ステータスレジスタを復帰
C言語プログラム例
                                                   /* ステータス レシ スタ保存変数定義 */
           char cSREG;
                                                   /* ステータス レジ スタを保存 */
           cSREG = SREG;
           _CLI();
                                                   /* EEPROM書き込み手順中割り込み禁止 */
           EECR \mid = (1 << EEMWE);
                                                   /* EEPROM主書き込み許可*/
                                                   /* EEPROM書き込み開始 */
           EECR = (1 << EEWE);
                                                   /* ステータス レジ スタを復帰 */
           SREG = cSREG:
```

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。

アセンフ・リ言語フ゜ロク・ラム例	
SEI SLEEP	;全割り込み許可;休止形態移行(割り込み待ち)
C言語プログラム例	
_SEI(); _SLEEP();	/* 全割り込み許可 */ /* 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Atmel® 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 ペクタ 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。このペクタは標準的に割り込み処理ルーチンへの無条件分岐で、この分岐は2クロック周期(訳注:原文は3(JMP命令=3を想定、実際はRJMP命令=2))要します。複数周期命令実行中に割り込みが起こると、その割り込みが扱われる前に、この命令が完了されます。MCUが休止形態の時に割り込みが起こると、割り込み実行応答時間は4クロック周期増やされます。この増加は選んだ休止形態からの起動時間に加えてです。

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



AVR ATmega8のメモリ

本章はAtmel® AVR® ATmega8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ATmega8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。

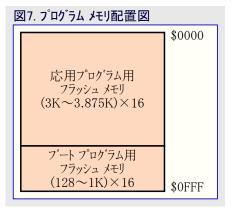
実装書き換え可能なプログラム用フラッシュメモリ

ATmega8はプログラム保存用に実装書き換え可能な8Kハイトのフラッシュ 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。ATmega8のプログラ ム カウンタ(PC)は12ビット幅、故に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135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。 144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 アトレス空間に配置できます。(LPM命令記述参照)

命令の取得と実行のタイミング図は9頁の「命令実行タイミング」で示されます。



データ用SRAM メモリ

図8.はATmega8のSRAMメモリ構成方法を示します。

下位1120データメモリ位置はレシブスタファイル、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1024位置はデータ用内蔵SRAMに充てます。

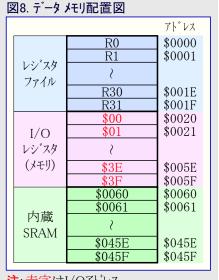
直接、間接、変位付き間接、事前減少付き間接、事後増加付き間接の5つの異なるアドレス 指定種別でデータ メモリ(空間)を網羅します。レシ、スタ ファイル内のレシ、スタ R26~R31は間接アト・レ ス指定ポインタ用レシ、スタが特徴です。

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レシ、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トレス指定動作を使うとき、(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。

ATmega8の32個の汎用レジスタ、64個のI/Oレジスタ、1024ハ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。レシ、スタファイルは8頁の「汎用レジ、スタ ファイル」で記述されます。

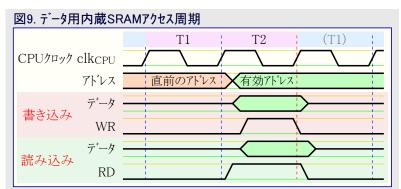


注:赤字はI/Oアドレス

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9.で記載 されるように2 clkcpu周期で実行されます。

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



データ用EEPROMメモリ

ATmega8は512小小のデータ用EEPROMを含みます。それは単一小小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。EEP ROM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アトレスレジスタ、EE PROMデータレジスタ、EEPROM制御レジスタで詳細に記述されます。

144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(<mark>訳注</mark>:原文はアクセス)時間は13頁の表1.で与えられます。(書き込みは)自己タイシンが機能ですが、使用者ソフトウェアは次ハーイが書ける時を検知してください。使用者コートが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。これはデバーイスが何周期かの時間で、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も低い電圧で走行する原因になります。これらの状態で問題を避ける方法の詳細については14頁の「EEPROMで、中人は100時止」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。

EEARH.EEARL (EEAR) - EEPROMプトレス レジスタ (EEPROM Address Register)

,	` ,	•		• `		0	•		
ピット	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	
ピット	7	6	5	4	3	2	1	0	_
\$1E (\$3E)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Read/Write	R/W								
初期値	不定								

ヒット15~9 - Res: 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

L*ット8~0 - EEAR8~0: EEPROMプトレス (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。

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

ピット	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。

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

ピット	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として読みます。

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

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



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

EEMWEL'ットはEEPROM書き込み許可(EEWE)ビットの1設定がEEPROM書き込みの原因となるかどうかを決めます。EEMWEが設定 (1)されるとき、EEWEの1設定は選ばれたアトレスのEEPROMにデータを書きます。EEMWEが0の場合、EEWEの1設定は無効です。EE MWE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ト・ウェアがこのビットを0に解除します。EEPROM書き込み手順については「書き込み許可(EEWE)ビット」の記述をご覧ください。

● ビット1 - EEWE: EEPROM書き込み許可 (EEPROM Write Enable)

EEPROM書き込み許可信号(EEWE)はEEPROMへの書き込みストローブです。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(EEWE)ビットがOになるまで待ちます。
- 2. SPM制御レシ、スタ(SPMCR)のSPM操作許可(SPMEN)ビットがOになるまで待ちます。
- 3. 今回のEEPROMアトレスをEEPROMアトレスレジスタ(EEAR)に書きます。(任意、省略可)
- 4. 今回のEEPROMデータをEEPROMデータレンブスタ(EEDR)に書きます。(任意、省略可)
- 5. EEPROM制御レジスタ(EECR)のEEPROM主書き込み許可(EEMWE)ビットに1を、EEPROM書き込み許可(EEWE)ビットに0を同時に書きます。
- 6. EEMWEL * ット設定後4クロック周期内にEEPROM書き込み許可(EEWE) と、ットへ論理1を書きます。

CPUがフラッシュ メモリ書き込み中、EEPROMはプログラム(書き込みが)できません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュ メモリのプログラミングが完了されていることを検査しなければなりません。2.はソフトウェアがフラッシュ メモリをプログラム(書き込みを)することをCPUに許すブートローダを含む場合だけ関係します。フラッシュ メモリが決してCPUによって更新されないなら、2.は省略できます。ブートプログラミングについての詳細に関しては135頁の「ブートローダ支援 - 書き込み中読み出し可能な自己プログラミング」をご覧ください。

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

書き込みアクセス時間が経過されると、EEPROM書き込み許可(EEWE)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への読み込みストローブです。EEARに正しいアドレスが設定されると、EEPROM読み出しを起動するためにEEREビットは論理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(EE AR)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)	代表					
EEPROM書き込み(CPU) 8448 8.5ms							
注: CKSELヒュース 設定と無限	関係に1MHzが使われ	ます。					

次のコート・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。そのようなコートが存在する場合、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:
                                                    ;EEPROM書き込み完了ならばスキップ
           SBIC
                  EECR, EEWE
           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アドレス下位バイ設定
           OUT
                  EEDR, R16
                                                    ;EEPROM書き込み値を設定
                  EECR, EEMWE
                                                    ;EEPROM主書き込み許可じット設定
           SBT
                                                    ;EEPROM書き込み開始(書き込み許可ビット設定)
           SBI
                  EECR, EEWE
           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 */
                                                    /* EEPROMアドレス設定 */
   EEAR = uiAddress;
   EEDR = ucData;
                                                    /* EEPROM書き込み値を設定 */
   EECR \mid = (1 << EEMWE);
                                                    /* EEPROM主書き込み許可*/
   EECR = (1 < \langle EEWE \rangle;
                                                    /* EEPROM書き込み開始 */
}
```

次のコート・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, EEWE
                                                  ;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
                                                  ;EEPROMアドレス下位バイト設定
           OUT
                 EEARL, R17
           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)
                                                  /* 以前のEEPROM書き込み完了まで待機 */
   while (EECR & (1<<EEWE));
                                                  /* EEPROMアドレス設定*/
   EEAR = uiAddress;
   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EPROM書き込みは正常に完了するが、その後発振器が止まらないことの注意)

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自身が命令を間違って実行し得ます。

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。

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



I/O メモリ (レジスタ)

ATmega8のI/O空間定義は206頁の「レジスタ要約」で示されます。

Atmel® AVR® ATmega8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、予約ビットはの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。 $CBI \geq SBI$ 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。 $CBI \geq SBI$ 命令は(I/Oアドレス)\$00~\$1Fのレジスタでのみ動作します。

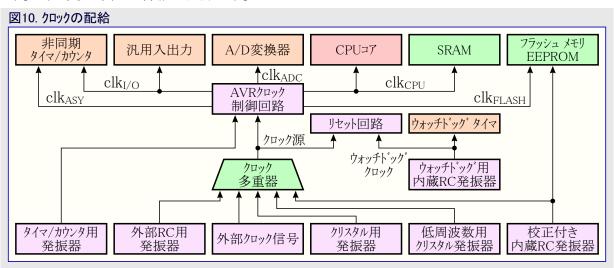
I/Oと周辺制御レジスタは以降の項で説明されます。



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10.はAtmel® 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、20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。クロック系統は図10.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。2線直列I/F(TWI)部のアドレス認証はclk_{I/O}が停止される時に非同期に実行され、全休止形態でTWIアドレス受信を可能とすることにも注意してください。

フラッシュ クロック - clkflash

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

非同期タイマ クロック - clk_{ASY}

非同期タイマクロックは外部32kHzクロック用クリスタルから非同期タイマ/カウンタの直接的なクロック駆動を許します。この専用クロックの範囲はディイスが休止形態の時でも、このタイマ/カウンタの実時間計数器としての使用を許します。非同期タイマ/カウンタはCPU主クロックと同じXTALピンを使いますが、発振器周波数の4倍より高いCPU主クロック周波数を必要とします。従って非同期動作はチップが内部発振器でクロック駆動されている間だけ利用可能です。

A/D変換クロック - clkaDC

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

クロック元

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

各クロック選択に対する様々な選択は次項で得られます。CPUがパワーダウンまたはパワーセーブから起動するとき、選んだクロック元は命令実行開始前に安定な発振器動作を保証するための起動時間に使われます。CPUがリセットから始まるとき、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3.で示されます。「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2. クロック元選択(注: 1=非プログラム、0=プログラム)

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



表3. 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11**.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つを選びます。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ヒュースが非プログラム(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ヒュース、が非プログラム(1)で8MHz、CKOPTヒュース、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。このコンテ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テンサ選択について初期の指針のいくつかは表4、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テ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4.で示されるようにCKSEL3~1ヒューズによって選ばれます。

CKSEL0ヒュース はSUT1,0ヒュース と共に表5.で示されるように起動時間を選びます。

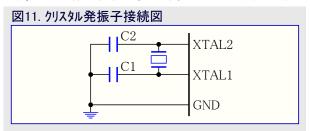


表4. クリスタル発振器動作

CKOPT	CKSEL3~1	周波数範囲	推奨C1,2容量
	101 (注1)	0.4~0.9MHz	_
1	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5. クリスタル発振子/セラミック振動子用起動遅延時間選択表

CKSEL0	SUT1,0	パワーダウン, パワーセーブ からの起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
	0 0	258×CK (注1)	4.1ms	外部セラミック振動子、高速上昇電源
0	0 1	258×CK (注1)	65ms	外部セラミック振動子、低速上昇電源
U	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
	0 0	1K×CK (注2)	65ms	外部セラミック振動子、低速上昇電源
1	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	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ばれなければなりません。クリスタルは図11.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。

この発振器が選ばれると、起動時間は表6.で示されるようにSUTヒューズによって決定されます。



表6. 低周波数クリスタル発振器用起動遅延時間選択表

SUT1,0	パワーダウン, パワーセーブ からの起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法					
0 0	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

タイングに鈍感な応用に対しては**図12**.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、それによって外部コンテ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7.で示されるようにCKSEL3~0ヒューズによって選ばれます。

この発振器が選ばれると、起動時間は表8.で示されるようにSUTヒュース、によって決定されます。

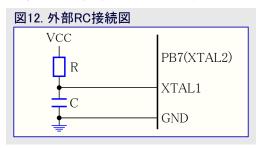


表7. 外部RC発振器動作							
CKSEL3~0	周波数範囲 (MHz)						
0 1 0 1	0.1~0.9						
0 1 1 0	0.9~3.0						
0 1 1 1	3.0~8.0						
1000	8.0~12.0						

表8. 外部RC発振器用起動遅延時間選択表

SUT1,0	パワーダウン, パワーセーブ からの起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
0 0	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
11	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℃での公称値です。このクロックは表9.で示されるようにCKSEL ヒューズのプログラミングによってシステム クロックとして選べます。選んだなら、外部部品なしで動作します。このクロック選択を使う時に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1MHz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℃で1.0MHz発振器周波数が選ばれ、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どの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

表9. 校正付き内蔵RC発振器動作

CKSEL3~0	公称周波数 (MHz)
0001(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。

て使われるとき、ウォッチト、ッグ発振器は未だウォッチト、ッグタイマとリセット付加遅延計時器に使われます。予め設定された校正値のより多くの情報については144頁の「校正が小」項をご覧ください。

この発振器が選ばれると、起動時間は**表10**.で示されるようにSUTヒューズによって決定されます。PB6(XTAL1/TOSC1)とPB7(XTAL2/TOSC2)は標準I/Oピンまたはタイマ/カウンタ用発振器ピンのどちらかとして使えます。

表10. 校正付き内蔵RC発振器用起動遅延時間選択表

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

注1: デバイスはこの選択で出荷されます。



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

ピット	7	6	5	4	3	2	1	0	_		
\$31 (\$51)	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)

このアトンスへの校正ハ・小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、識票列上位ハ・小(アトンス\$00)に配置される1MHz校正値が発振校正レシ、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票列を読み、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ばれます。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11.で示されるような他の値への調整は保証されません。

表11. 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13**.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ヒューズは'0000'にプログラム(設定)されなければなりません。CKOPTヒューズのプログラム(0)により、使用者はXTAL1とGND、XTAL2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ばれると、起動時間は**表12**.で示されるようにSUTヒューズによって決定されます。

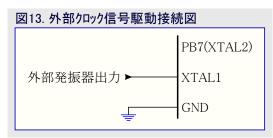


表12. 外部クロック信号駆動用起動遅延時間選択表

SUT1,0	パワーダウン, パワーセーブ からの起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法					
0 0	6×CK	-	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					
11	(予約)							

外部クロックを供給する時に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ことを保証することが必要とされます。

タイマ/カウンタ用発振器

AVR マイクロ コントローラのタイマ/カウンタ用発振器(TOSC1とTOSC2)ピンに対してクリスタル発振子は、このピン間に直接的に接続されます。 CKOPTヒュース、のプログラム(0)により、使用者はXTAL1(TOSC1)とXTAL2(TOSC2)の内部容量(コンデンサ)を許可にでき、それによって必要とした外部コンデンサを取り去れます。この発振器は時計用32.768kHzクリスタルでの使用に最適化されています。外部クロック元をTOSC1に供給することは推奨されません。

注: タイマ/カウンタ用発振器は低周波数クリスタル用発振器と同じ形式のクリスタル用発振器を使い、内部容量(C)は同じ36pFの公称値を持ちます。

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。

5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。MCUCRの休止種別選択(SM2~0)ビットはSLEEP命令によって活性(有効)にされる休止形態(アイ・ル、A/D変換雑音低減、パワータ・ウン、パワーセーブ、スタンハイ)のどれかを選びます。一覧については表13.をご覧ください。MCUが休止形態中に許可した割り込みが起こると、MCUは起動します。その時MCUは起動時間に加えて4周期停止され、割り込みルーチンを実行し、そしてSLEEP命令の次の命令から実行を再開します。デバイスが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は起動してリセット ベクタから実行します。

他のAVRの多くに存在する拡張スタンバイ動作はTOSCとXTAL入力が同じ物理ピンを共有するため、ATmega8で削除されてしまっていることに注意してください。

16頁の図10.はATmega8の各種クロック系統とその配給を示します。この図は適切な休止形態を選ぶ助けになります。

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

このMCU制御レジスタは電力管理用の制御ビットを含みます。

ピット	7	6	5	4	3	2	1	0	
\$35 (\$55)	SE	SM2	SM1	SM0	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 - 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)することが推奨されます。

• ビット6~4 - SM2~0: 休止種別選択 (Sleep Mode Select Bits 2, 1 and 0)

これらのビットは表13.で示される利用可能な5つの休止形態の1つを選びます。

表13. 休	表13. 休止形態種別選択									
SM2	SM1	SM0	休止形態種別							
0	0	0	アイドル動作							
0	0	1	A/D変換雑音低減動作							
0	1	0	パプーダウン動作							
0	1	1	パ゚ワーセーフ゛動作							
1	0	0	(予約)							
1	0	1	(予約)							
1	1	0	スタンバイ動作 (注)							
1	1	1	(予約)							

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

アイドル動作

休止種別選択($SM2\sim0$)ビットが'000'を書かれるとき、SLEEP命令はMCUをアイル動作へ移行させ、CPUを停止しますが、SPI、USAR T、アナログ比較器、A/D変換器、2線直列インターフェース、タイマ/カウンタ、ウォッチドック、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<math>ClkCPUとClkFLASHを停止する一方、他のクロックに走行を許します。

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

A/D変換雑音低減動作

SM2~0ビットが'001'を書かれるとき、SLEEP命令はMCUをA/D変換雑音低減動作へ移行させ、CPUを停止しますが、A/D変換器、外部割り込み、2線直列インターフェースのアドレス監視、タイマ/カウンタ2、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkɪ/O, clkCPU, clkFLASHを停止する一方、他のクロックに走行を許します。

これはA/D変換に対する雑音環境を改善し、より高い分解能の測定を可能にします。A/D変換器が許可されている場合、この動作に移行すると、変換が自動的に始まります。A/D変換完了割り込みからの他、外部リセット、ウォッチトッグ リセット、低電圧検出(BOD)リセット、2線直列インターフェースのアドレス一致割り込み、タイマ/カウンタ2の割り込み、SPM/EEPROM操作可割り込み、INTOまたはINT1の外部レベル割り込みだけが、A/D変換雑音低減動作からMCUを起動できます。



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、2線直列インターフェースのアドレス監視、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、2線直列インターフェースのアドレス一致割り込み、INT0またはINT1の外部レヘブル割り込みだけがMCUを起動できます。この休止形態は基本的に生成した全てのクロックを停止し、非同期部の動作だけを許します。

レベルで起動した割り込みがパワーダウン動作からの復帰に使われる場合、MCUを起動するためには変更したレベルが一定時間保持されなければならないことに注意してください。詳細については43頁の「外部割り込み」を参照してください。

ハプ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16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ヒュース、によって定義されます。

パワーセーブ動作

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

タイマ/カウンタ2が非同期にクロック駆動される、換言するとタイマ/カウンタ2非同期状態レジスタ(ASSR)の非同期クロック(AS2)ビットが設定(1)されると、タイマ/カウンタ2は休止中に走行(動作)します。ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、タイマ/カウンタ割り込み許可レジスタ(TIMSK)のタイマ/カウンタ2溢れ割り込み許可(TOIE2)ビットまたは比較2割り込み許可(OCIE2)ビットが設定(1)されるなら、デバイスは対応するどちらの割り込みからでも起動できます。

タイマ/カウンタ2が非同期にクロック駆動されないならば、パワーセーブ動作での起動復帰後のタイマ/カウンタ2のレジスタ内容はASSRのAS2=<mark>0</mark>でも不定とみなすべきなので、パワーダウン動作をパワーセーブ動作の代わりにすることが推奨されます。

この休止形態は基本的にclkASY以外の全てのクロックを停止し、非同期にクロック駆動されるならタイマ/カウンタ2を含め、非同期部の動作だけを許します。

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

表14. 各休止形態に於ける動作クロック範囲と復帰起動要因

			動作クロック範囲					動作発振器		復帰起動要因 (割り込み)					
休止種別	clk CPU	clk flash	clk 10	clk ADC	clk ASY	主クロック供給元	タイマ用 発振器	INT1 INT0	TWI アト・レス 一致	タイマ/ カウンタ2	SPM EEPROM 操作可	A/D 変換 完了	その他 I/O		
	アイドル			0	0	0	0	2	0	0	0	0	0	0	
A/D	変換雑音低減				0	0	0	2	3	0	0	0	0		
,	ハ [°] ワータ [゛] ウン								3	0					
)	ヽ゚ヷーセーブ゛					2		2	3	0	2				
	スタンバイ(注1)	·					0		3	0			·		

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

- ② タイマ/カウンタ2非同期状態レシ、スタ(ASSR)の非同期クロック(AS2)ビットが設定(1)された場合です。
- ③ 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ように選ばれるべきです。必要とされない全ての機能は禁止されるべきです。特に次の機能部は最低可能消費電力の達成を試みるとき、特別な考慮を必要とするでしょう。

A/D変換器 (ADC)

許可したなら、A/D変換器は全休止形態で許可されます。節電するために休止形態の何れかへ移行する前にA/D変換器は禁止されるべきです。A/D変換器がOFFそして再びONに切り替えられると、次の(最初の)変換は延長された(初回)変換になります。A/D変換器操作の詳細については127頁の「A/D変換器」を参照してください。

アナログ比較器

アイドル動作へ移行するとき、アナロゲ比較器は使われないなら、禁止されるべきです。A/D変換雑音削減動作へ移行するとき、アナロゲ比較器は禁止されるべきです。その他の休止形態でのアナロゲ比較器は自動的に禁止されます。しかしアナロゲ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ゲ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。アナロゲ比較器設定法の詳細については125頁の「アナロゲ比較器」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BODENヒュース、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24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、A/D変換器によって必要とされる時に許可されます。これら単位部が上の項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25頁の「内部基準電圧」を参照してください。

ウォッチト゛ック゛タイマ

ウォッチト、ック、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。ウォッチト、ック、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。ウォッチト、ック、タイマ 設定法の詳細については26頁の「ウォッチト、ック、タイマ」を参照してください。

ホ[°]ート ヒ[°]ン

休止形態へ移行するとき、全ポート ピンは最小電力使用に設定されるべきです。最も重要なことはその時にピンが抵抗性負荷を駆動しないのを保証することです。I/Oクロック(clk_{I/O})とA/D変換クロック(clk_{ADC})の両方が停止される休止形態ではデバイスの入力緩衝部が禁止されます。これは必要とされない時に入力論理回路によって電力が消費されないことを保証します。いくつかの場合で入力論理回路は起動条件を検出するために必要とされ、その時は許可されます。どのピンが許可されるかの詳細については34頁の「デン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レシ、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、ブクタから実行を開始します。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、ブクタは使われず、これらの位置に通常のプログラムコートが配置できます。これはリセット、ブクタが応用領域の一方、割り込み、ブクタがブート領域の場合やその逆も同様です。図14.の回路構成図はリセット論理回路を示します。表15.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(計時器)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16頁の「クロック元」で示されます。

リセット元

ATmega8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(VPOT)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。
- ウォッチト、ック、リセット・・・ ウォッチト、ック、が許可され、ウォッチト、ック、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(VBOT)以下でMCUがリセットされます。

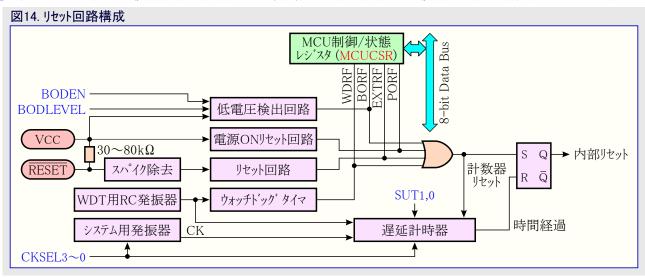


表15.1	カセット電気的特性	

シンホ゛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	注		1.4	2.3	
VPOT	下降時電源ONリセット閾値電	注圧 (<mark>注1</mark>)		1.3	2.3	V
V _{RST}	RESETt°ン閾値電圧		0.2VCC		0.9VCC	
t_{RST}	RESETピンでの最小パルス幅	1.5			μs	
V_{BOT}	低電圧検出 (注2)	BODLEVEL=非プログラム(1)	2.4	2.6	2.9	V
A BO I	閾値電圧 (元2)	BODLEVEL=プログラム(0)	3.7	4.0	4.5	V
tnom	是小任命已经出時期	BODLEVEL=非プログラム(1)		2		110
t_{BOT}	最小低電圧検出時間		2		μs	
V _{HYST}	低電圧検出ヒステリシス電圧			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ONJセットは動作しません。

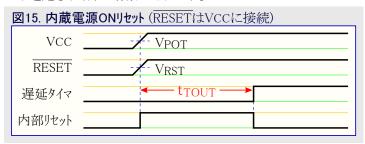
注2: VBOTはいくつかのデバイスについて公称最小動作電圧以下かもしれません。この状態のデバイスについて、そのデバイスは製造検査中にVCC=VBOTへ落として検査されます。これはVCCがマイクロコントローラの正しい動作がもはや保証されない電圧へ落ちる前に低電圧(BOD)リセットが起きることを保証します。この検査はATmega8についてはBODLEVEL=1、ATmega8についてはBODLEVEL=1、ATmega8についてはBODLEVEL=1、なるではBODLEVEL=1、ATmega8に対してBODLEVEL=1は適用できません。

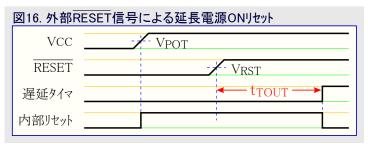


電源ONリセット

電源ONJ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15.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。

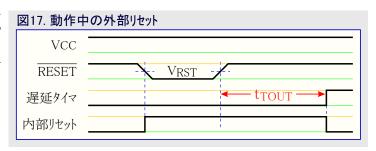
電源ONJセット回路はデバイスが電源投入でリセットされるのを保証します。電源ONJセット閾値電圧(VPOT)への到達はVCCの上昇後にデバイスがどのくらいリセットを保つかを決める遅延計数器(計時器)を起動します。VCCがこの検出電圧以下に低下すると、リセット信号はどんな遅延もなく再び有効にされます。





外部リセット

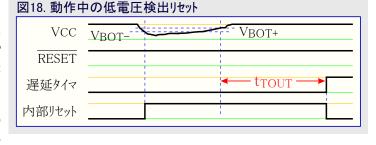
外部リセットはRESETtシのLowレベルによって生成されます。例えクロックが動いていなくても、最小パルス幅(表15.参照)以上のリセットパルスはリセットを生成します。短すぎるパルスはリセットの生成が保証されません。印加された信号の上昇がリセット閾値電圧(VRST)に達すると(遅延タイマが起動され)、遅延タイマは遅延時間(tTOUT)経過後にMCUを始動します。



低電圧(ブラウンアウト)検出

ATmega8には固定化された起動(検出)電圧と比較することによって動作中のVCCを監視するチップ。上の低電圧検出(BOD)回路があります。BODの起動電圧はBODLEVELヒュース。により、2.6V(非プログラム(1))または4.0V(プログラム(0))を選べます。この起動電圧はスパイク対策BODを保証するためにヒステリシスを持ちます。検出電圧のヒステリシスはVBOT+=VBOT+VHYST/2、VBOT-=VBOT-VHYST/2と解釈すべきです。

BOD回路はBODENtュースによって許可/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

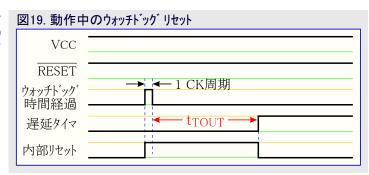


下降すると(図18.のVBOT-)、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(図18.のVBOT+)、(遅延タイマが起動され、)遅延タイマは遅延時間(tTOUT)経過後にMCUを始動します。

BOD回路は電圧が表15.で与えられるtBOD時間より長く起動電圧以下に留まる場合だけ、VCCでの低下を検出します。

ウォッチト゛ック゛リセット

ウォッチト、ック・時間経過時、(内部的に)1 CK周期幅の短いリセット パルスを生成します。本パルスの下降端で遅延タイマが遅延時間(t_{TOUT})の計時を始めます。ウォッチト、ック・タイマ操作の詳細については26頁を参照してください。



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

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。

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

ビット7~4 - Res: 予約 (Reserved)

これらのビットは予約されており、常に0として読みます。

ヒット3 - WDRF: ウォッチトック、リセット フラク (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 フラク (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 フラク (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 フラク (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

ATmega8は内部基準電圧が特徴です。この基準電圧は低電圧検出(BOD)に使われ、A/D変換やアナログ比較器の入力としても使えます。A/D変換器への2.56V基準電圧は内部基準電圧(1.30V)から生成されます。

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16.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。

- 低電圧検出リセット許可時(BODENヒューズ=プログラム(0))
- 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)
- A/D変換部動作許可時(A/D変換制御/状態レジスタA(ADCSRA)のA/D動作許可(ADEN)=1)

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

表16. 内部基準雷圧特性

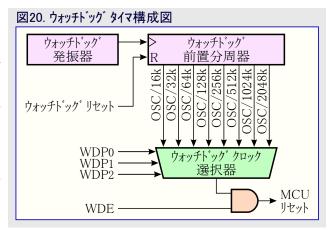
シンオ	゛ル	項目	最小	代表	最大	単位
$V_{\rm E}$	SG .	基準電圧	1.15	1.30	1.40	V
t_{B}	G	起動時間		40	70	μs
I_{B}	G	消費電流		10		μA



ウォッチト゛ック゛タイマ

ウォッチト、ック、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。ウォッチト、ック、タイマ前置分周器の制御により、ウォッチト、ック、リセット間隔は表17.で示されるように調整できます。ウォッチト、ック、リセット(WDR)命令はウォッチト、ック、タイマをリセットします。ウォッチト、ック、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。このリセット周期を決めるために8つの異なるクロック周期を選べます。別のウォッチト、ック、リセットなしにリセット周期が経過すると、ATmega8はリセットし、リセット、ブクタから実行します。ウォッチト、ック、リセットの詳細タイミングについては24頁を参照してください。

ウォッチト、ック、の予期せぬ禁止を防ぐため、ウォッチト、ック、が禁止されるとき、特別なOFFへの切り替え手順に従わなければなりません。詳細については「ウォッチト、ック、タイマ制御レシ、スタ(WDTCR)」を参照してください。



WDTCR - ウォッチドック タイマ制御レシ、スタ (Watchdog Timer Control Register)

ピット	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として読みます。

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

ウォッチト、ック、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、ウォッチト、ック、は禁止されません。一旦1を書かれると、ハートウェアが4クロック周期後、このビットを解除(0)します。ウォッチト、ック、禁止手順については「ウォッチト、ック、許可(WDE)」ビットを参照してください。安全基準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。27頁のコート例をご覧ください。

● ビット3 - WDE : ウォッチドック゛許可 (Watchdog Enable)

ウォッチト、ック、許可(WDE)が論理1を書かれるとウォッチト、ック、タイマが許可され、WDEが論理0を書かれるとウォッチト、ック、タイマ機能が禁止されます。WDEはウォッチト、ック、変更許可(WDCE)と、ットが論理レヘル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ト、ック、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、ウォッチト、ック、変更許可(WDCE)ヒ、ットとウォッチト、ック、許可(WDE)ヒ、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。

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

このWDP2~0ビットはウォッチドッグタイマが許可される時のウォッチドッグタイマの前置分周を決めます。各種前置分周値と対応する計時完了周期は表17. **b** 0 0 **b** 0

WDP2	WDP1	WDP0	WDT発振周期数	代表的な計	時完了周期
WDP2	WDP1	WDP0	WDI光派向别致	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

ウォッチト、ック、タイマ設定変更の時間制限手順

次のコート・例はウォッチト、ック(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。

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

```
アセンブリ言語プログラム例
                                                       ;ウォッチトック タイマ リセット
WDT OFF:
            WDR
                                                       ;現WDTCR値を取得
            IN
                   R16, WDTCR
            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
                   R16, (0<<WDE)
                                                       ;WDE論理0値を取得
            LDI
                   WDTCR, R16
                                                       ;ウォッチト・ック・禁止
            OUT
            RET
                                                       ;呼び出し元へ復帰
C言語プログラム例
void WDT_off(void)
                                                       /* ウォッチト・ック・タイマ リセット */
    _WDR();
    WDTCR = (1 << WDCE) | (1 << WDE);
                                                       /* WDCEとWDEに論理1書き込み */
                                                       /* ウォッチト・ック 禁止 */
    WDTCR = 0x00;
```

安全基準1 (WDTONヒュース 非プログラム(1))

この動作種別ではウォッチト、ック、タイマが初めに禁止されますが、どんな制限もなくウォッチト、ック、許可(WDE)ヒ、ットに1を書くことによって許可できます。ウォッチト、ック、計時完了周期を変更または許可したウォッチト、ック、タイマを禁止する時に時間制限手順が必要とされます。許可したウォッチト、ック、タイマを禁止や、ウォッチト、ック、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)でウォッチトックで変更許可(WDCE)とWDEに論理1を書きます。WDEビットの直前の値に拘らず、論理1がWDEに書かれなければなりません。
- 2. 次からの4クロック周期内に同じ操作(命令)で欲したWDEとウォッチドッグタイマ前置分周選択(WDP2~0)ビットを書きますが、WDCEビットは解除(0)されてです。

安全基準2 (WDTONヒューズ プログラム(0))

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

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

本章はATmega8によって実行される割り込み操作の詳細を記述します。AVR割り込み操作の一般説明については9頁の「リセットと割 り込みの扱い」を参照してください。

ATmega8の割り込みべかり

表18. 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 (<u>注2</u>)	発生元 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2 COMP2	タイマ/カウンタ2比較一致
5	\$004	タイマ/カウンタ2 OVF2	タイマ/カウンタ2溢れ
6	\$005	タイマ/カウンタ1 CAPT	タイマ/カウンタ1捕獲発生
7	\$006	タイマ/カウンタ1 COMPA	タイマ/カウンタ1比較A一致
8	\$007	タイマ/カウンタ1 COMPB	タイマ/カウンタ1比較B一致
9	\$008	タイマ/カウンタ1 OVF1	タイマ/カウンタ1溢れ
10	\$009	タイマ/カウンタ0 OVF0	タイマ/カウンタ0溢れ
11	\$00A	SPI STC	SPI 転送完了
12	\$00B	USART RX	USART 受信完了
13	\$00C	USART UDRE	USART 送信緩衝部空き
14	\$00D	USART TX	USART 送信完了
15	\$00E	A/D変換器 ADC	A/D変換完了
16	\$00F	EEPROM EE_RDY	EEPROM 操作可
17	\$010	アナロク・比較器 ANA_COMP	アナログ比較器出力遷移
18	\$011	2線直列インターフェース TWI	2線直列インターフェース状態変化
19	\$012	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フートローダアドレスへ飛びます。135頁の「プートローダ支援 - 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)の割り込みへ、クタ選択(IVSEL)ビットが設定(1)されると、割り込みへ、クタはフェ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19.はBOOTRST(ヒューズ)とIVSEL(割り込みベクタ 表19. リセットと割り込みベクタの配置 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トを置 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。

BOOTRST	IVSEL	リセット へ゛クタ アト゛レス	割り込みベクタ先頭アドレス
ゴヒつ°ロカ*ニ) (1)	0	\$0000	\$0001
非プログラム(1)	1	\$0000	ブート領域先頭アドレス+\$0001
プ [°] ロク [*] ラム(0)	0	ブート領域先頭アトレス	\$0001
7 17 74(0)	1	ブート領域先頭アトレス	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143頁の**表82**.で示されます。

ATmega8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。

アト・レス	ラヘ゛ル	命令		注釈
\$0000		R.JMP	RESET	;各種リセット
\$0001		RJMP	EXT_INTO	;外部割り込み要求0
\$0002		R.JMP	EXT_INT1	;外部割り込み要求1
\$0003		RJMP	TIM2_COMP	;タイマ/カウンタ2比較一致
\$0004		RJMP	TIM2_OVF	;タイマ/カウンタ2溢れ
\$0005		RJMP	TIM1_CAPT	;タイマ/カウンタ1捕獲発生
\$0006		RJMP	TIM1_COMPA	;タイマ/カウンタ1比較A一致
\$0007		RJMP	TIM1_COMPB	;タイマ/カウンタ1比較B一致
\$0008		RJMP	TIM1_OVF	;タイマ/カウンタ1溢れ
\$0009		RJMP	TIMO_OVF	;タイマ/カウンタ0溢れ
\$000A		RJMP	SPI_STC	;SPI転送完了
\$000B		RJMP	USART_RXC	;USART受信完了
\$000C		RJMP	USART_DRE	;USART送信緩衝部空
\$000D		RJMP	USART_TXC	;USART送信完了
\$000E		RJMP	ADC	;A/D変換完了
\$000F		RJMP	EE_RDY	;EEPROM操作可
\$0010		RJMP	ANA_COMP	;アナログ比較器出力遷移
\$0011		RJMP	TWI	;2線直列インターフェース状態変化
\$0012		RJMP	SPM_RDY	;SPM命令操作可
;			,	
\$0013	RESET:	LDI	R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$0014		OUT	SPH, R16	;スタック ポインタ上位を初期化
\$0015		LDI	R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0016		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。

アト・レス	ラヘ゛ル	命令		注釈
\$0000 ;		RJMP	RESET	;各種リセット (BOOTRSTヒューズ˙=1)
\$0001 \$0002 \$0003 \$0004	RESET:	LDI OUT LDI OUT	R16, HIGH (RAMEND) SPH, R16 R16, LOW (RAMEND) SPL, R16	; RAM最終アドレス上位を取得(応用プログラム開始) ; スタック ホプインタ上位を初期化 ; RAM最終アドレス下位を取得 ; スタック ホプ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 \$0C02		RJMP RJMP	EXT_INTO EXT_INT1	;外部割り込み要求0 ;外部割り込み要求1
\$0C12		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。

アト・レス	ラヘ゛ル	命令		注釈
		. ORG	\$0001	;割り込みベクタ先頭
\$0001		RJMP	EXT_INTO	;外部割り込み要求0
\$0002		RJMP	EXT_INT1	;外部割り込み要求1
\$0012		RJMP /	SPM_RDY	;SPM命令操作可 ;以下、プログラムなど
				72(17)77746
		. ORG	\$0C00	;ブート プログラム領域が <mark>2K</mark> バイトの場合
\$0C00 ;		RJMP	RESET	;各種リセット (BOOTRSTヒューズ=0)
\$0C01	RESET:	LDI	R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得(プログラム開始)
\$0C02		OUT	SPH, R16	;スタック ポインタ上位を初期化
\$0C03		LDI	R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0C04		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。

アト・レス	ラヘ゛ル	命令 . ORG	\$0C00	注釈 ; ブート プログラム領域が2Kバイトの場合
\$0C00 \$0C01		RJMP RJMP	RESET EXT_INTO	;各種リセット (BOOTRSTヒューズ=0) ;外部割り込み要求0
\$0C02 \$0C12		RJMP ≀ RJMP	EXT_INT1 SPM_RDY	;外部割り込み要求1 ;SPM命令操作可
\$0C13 \$0C14 \$0C15 \$0C16	RESET:	LDI OUT LDI OUT OUT	R16, HIGH (RAMEND) SPH, R16 R16, LOW (RAMEND) SPL, R16	;RAM最終アトレス上位を取得(プログラム開始) ;スタック ポインタ上位を初期化 ;RAM最終アトレス下位を取得 ;スタック ポプンタ下位を初期化 ;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シブスタ(GICR)は割り込みへブクタ表の配置を制御します。



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

ピット	7	6	5	4	3	2	1	0	_
\$3B (\$5B)	INT1	INT0	-	-	-	-	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R	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 ローダ領域の始まりへ移動されます。プート 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135頁の「**プート ローダ支援 - 書き込み中読み出し可能な自己プログラミング**」章を参照してください。割り込みベクタ表の予期せぬ変更を防ぐため、IVSELビットの変更は特別な書き込み手順に従わなければなりません。

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

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

注: 割り込みベクタがブート ローダ領域に配置され、BLB02ブート施錠ビットがプログラム(0)されると、応用領域から実行する間中、割り込みが禁止されます。割り込みベクタが応用領域に配置され、BLB12ブート施錠ビットがプログラム(0)されると、ブート ローダ領域から実行する間中、割り込みが禁止されます。ブート施錠ビットの詳細については135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ク**゙」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ビットは割り込みペクタ選択(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。以下のコート・例をご覧ください。

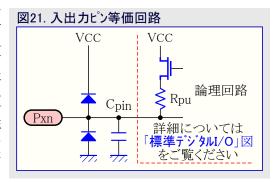
```
アセンブリ言語プログラム例
MOVE IVT:
                  R16, GICR
                                                     ;現GICR値取得
           ΤN
           MOV
                  R17, R16
                                                     ;現GICR値複写
                  R16, (1<<IVCE)
           ORI
                                                     ;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
                                                     ;ブート領域へ割り込みへづタを移動
           OUT
                  GICR, R17
           RET
                                                     ;呼び出し元へ復帰
C言語プログラム例
void Move_interrupts(void)
                                                     /*一時定数定義 */
   uchr temp;
    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へ、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(リート、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)ア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(シンケ)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充分な強さです。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21.で示されるようにVCCとGNDの両方に保護ゲイオート、を持っています。各値の完全な一覧については156頁の「電気的特性」を参照してください



本項内の全てのレシ、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シ、スタやビット定義に使う時は正確な形式(例えば、ここで一般に記されたPORTxnがポートBのビット3に対してはPORT B3)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42頁の「I/Oポート用レジスタ」で一覧されます。

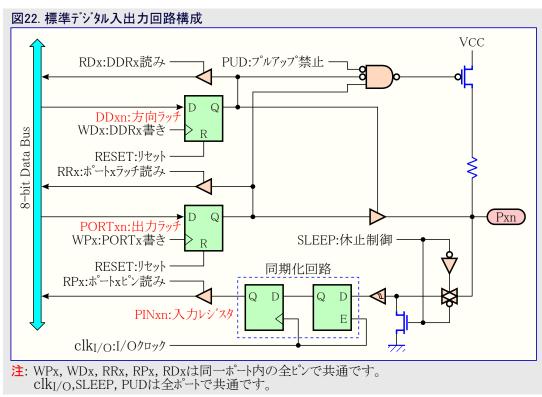
各々1つの出力レジスタ(PORTx)、方向レジスタ(DDRx)、入力レジスタ(PINx)の各ポートに対して、3つI/Oメモリアトレス位置が割り当てられます。入力レジスタのI/O位置は読むだけで、一方出力レジスタと方向レジスタは読み書き(両方)です。加えて特殊I/O機能レジスタ(SFIOR)のプルアップ禁止(PUD)ビットが設定(1)されると、全ポートで全ビットに対してプルアップ機能を禁止します。

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35頁の「交換ポート機能」で記述されます。交換機能の完全な記述については個別機能部項を参照してください。

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

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22.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。



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。42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、PINxビットはPINx I/Oアドレスでアクセスされます。

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

そのピンが入力ピンとして設定される時にPORTxnが論理1を書かれると、プルアップ抵抗が活性(有効)にされます。プルアップ抵抗をOFF に切り替えるにはPORTxnが論理0を書かれるか、またはそのピンが出力ピンとして設定されなければなりません。ポート ピンは例え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機能レン、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20.はピン値に対する制御信号の一覧を示します。

表20. ポート ピンの設定

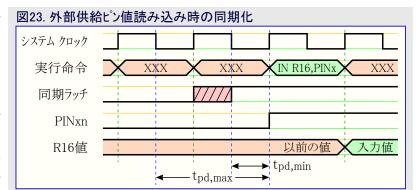
DDxn	PORTxn	PUD (SFIOR)	入出力	プルアップ抵抗	備考
0	0	X	入力	なし	高インピーダンス (Hi-Z)
0	1	0	入力	あり	Pxnに外部からLowを入力すると吐き出し電流が流れます。
0	1	1	入力	なし	高インピーダンス (Hi-Z)
1	0	X	出力	なし	Low (吸い込み)出力
1	1	X	出力	なし	High (吐き出し)出力

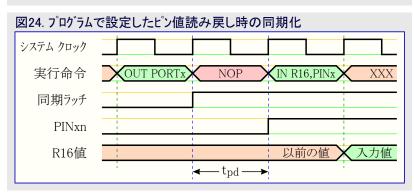
ピン値の読み込み

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22.で示されるようにPINxnレジスタ ビットを先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタ ステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。図23.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tpd,minとtpd,maxで示されます。

(図23.で)システム クロックの最初の下降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(トランスペプレント)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tpd,minとtpd,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24.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(tpd)は1システムクロック周期です。







次のコート・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
                   R16, (1<<PB7) | (1<<PB6) | (1<<PB1) | (1<<PB0)
            LDI
                                                                 ;プルアップとHigh値を取得
                   R17, (1<<DDB3) | (1<<DDB2) | (1<<DDB1) | (1<<DDB0)
                                                                 ;出力ビット値を取得
            LDI
                                                                 ;プルアップとHigh値を設定
            OUT
            OUT
                   DDRB, R17
                                                                 ;入出力方向を設定
            NOP
                                                                 ;同期化遅延対処
                                                                 ;ピン値読み戻し
            IN
                   R16, PINB
C言語プログラム例
unsigned char i;
                                                                 /* */
    PORTB = (1 << PB7) | (1 << PB6) | (1 << PB1) | (1 << PB0);
                                                                 /* プルアップとHigh値を設定 */
    DDRB = (1 << DDB3) | (1 << DDB2) | (1 << DDB1) | (1 << DDB0);
                                                                 /* 入出力方向を設定 */
    NOP();
                                                                 /* 同期化遅延対処 */
    i = PINB;
                                                                 /* 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22.で示されるようにデジタル入力信号はシュミットトリカの入力を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、パワータウン動作、パワーセーブ動作、スタンバイ動作でMCU休止制御器によって設定(1)されます。

SLEEPは外部割り込みピンとして許可されたポート ピンに対しては無視されます。外部割り込み要求が許可されないなら、SLEEPは他のピンについてと同様に有効です。SLEEPは35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 ピンには標準デッタル入出力に加えて交換機能があります。図25.は単純化された図22.でのポート ピン制御信号が交換機能によってどう重複され得るかを示します。この重複信号は全てのポート ピンに存在する訳ではありませんが、この図はAVR マイクロコントローラ系統の全ポート ピンに適用できる一般的な記述として取り扱います。

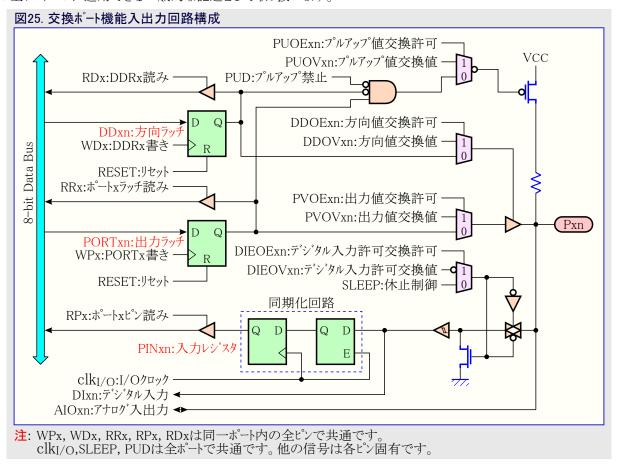


表21.は重複(交換)信号の機能一覧を示します。図25.で示すピンとポートは次表で示されません。重複(交換)信号は交換機能を持つ機能部で内部的に生成されます。

表21. 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レシブスタ値に関係なく、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レシブスタ値に関係なく、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リガ出力に接続されていますが、 これは同期化前となります。この信号はクロックとしての使用を除き、各交換機能自身が同 期化します。					
AIO	アナログ入出力	交換機能用アナログ入出力です。この信号はピンに直接接続され、双方向使用ができます。					

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



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

ピット	7	6	5	4	3	2	1	0	
\$30 (\$50)	_	-	-	-	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ビット2 - 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33頁の「ピンの設定」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22.で示されます。

表22. ポートBピンの交換機能

ホ°ート ヒ°ン	交換機能
PB7	XTAL2 (システム クロック用発振増幅器出力) TOSC2 (タイマ用発振増幅器出力)
PB6	XTAL1 (システム クロック用発振増幅器入力または外部クロック信号入力) TOSC1 (タイマ用発振増幅器入力)
PB5	SCK (SPI 直列クロック 主装置側出力/従装置側入力)
PB4	MISO (SPI 主装置側データ入力/従装置側データ出力)
PB3	MOSI (SPI 主装置側データ出力/従装置側データ入力) OC2 (タイマ/カウンタ2 比較一致出力)
PB2	SS
PB1	OC1A (タイマ/カウンタ1 比較A一致出力)
PB0	ICP1 (タイマ/カウンタ1 捕獲起動入力)

交換ピンの設定は次のとおりです。

XTAL2/TOSC2 - ポートB ビット7 : PB7

XTAL2: チップ(システム) クロック発振器ピン2。 クリスタル発振器または低周波数クリスタル発振器用クロックピンとして使われます。 クロック ピンとして使われると、このピンはI/Oピンとして使えません。

TOSC2: タイマ発振器ピン2。 チップ クロック元として校正付き内蔵RC発振器が選ばれ、非同期タイマが非同期状態レジスタ(ASSR)の正しい設定により許可される場合だけ使われます。 タイマ/カウンタ2の非同期クロック動作を許可するためにASSRの非同期許可(AS 2)ビットが設定(1)されると、PB7ピンはポートから切り離され、発振用反転増幅器の出力になります。 この動作では、このピンにクリスタル発振器が接続され、このピンはI/Oピンとして使えません。

PB7がクロック ピンとして使われると、PORTB7, DDB7, PINB7は全て0を読みます。

• XTAL1/TOSC1 - ポートB ビット6 : PB6

XTAL1: チップ(システム) クロック発振器ピン1。 クリスタル発振器または低周波数クリスタル発振器用クロックピンとして使われます。 クロック ピンとして使われると、このピンはI/Oピンとして使えません。

TOSC1: タイマ発振器ピン1。チップ゚ クロック元として校正付き内蔵RC発振器が選ばれ、非同期タイマが非同期状態レシ、スタ(ASSR)の正しい設定により許可される場合だけ使われます。タイマ/カウンタ2の非同期クロック動作を許可するためにASSRの非同期許可(AS2)ビットが設定(1)されると、PB6ピンはポートから切り離され、発振用反転増幅器の入力になります。この動作では、このピンにクリスタル発振器が接続され、このピンはI/Oピンとして使えません。

PB6がクロック ピンとして使われると、PORTB6, DDB6, PINB6は全て0を読みます。

• SCK - ポートB ビット5 : PB5

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 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で制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB5で制御できます。

• MISO - ポートB ビット4 : PB4

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 DDB4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DDB4で制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB4で制御できます。

MOSI/OC2 - ポートB ビット3 : PB3

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方向レシ、スタ(DDRB)の DDB3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DDB3で制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジ、スタ(PORTB)のPORTB3で制御できます。

OC2: タイマ/カウンタ2の比較一致出力。PB3ピンはタイマ/カウンタ2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、PB3ピンは出力として設定(DDB3=1)されなければなりません。このOC2ピンはタイマ機能のPWM動作用出力ピンでもあります。

• SS/OC1B - ポートB ビット2 : PB2

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DDB2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DDB2によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB2によって制御できます。

OC1B: タイマ/カウンタ1の比較B一致出力。PB2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、PB2 ピンは出力として設定(DDB2=1)されなければなりません。このOC1Bピンはタイマ機能のPWM動作用出力ピンでもあります。

• OC1A - ポートB ビット1 : PB1

OC1A: タイマ/カウンタ1の比較A一致出力。PB1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、PB1ピンは出力として設定(DDB1=1)されなければなりません。このOC1Aピンはタイマ機能のPWM動作用出力ピンでもあります。

• ICP1 - ポートB ビット0 : PB0

ICP1: タイマ/カウンタ1の捕獲起動入力。PB0ピンはタイマ/カウンタ1用捕獲起動入力ピンとして動作できます。

表23.と表24.は、ポートBの交換機能を35頁の図25.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23. ポートB7~4の交換機能用交換信号

₹20. ホ TD7	10人民族能用人族后与			
信号名	PB7/XTAL2/TOSC2	PB6/XTAL1/TOSC1	PB5/SCK	PB4/MISO
PUOE	EXT•(INTRC+AS2)	INTRC+AS2	SPE•MSTR	SPE•MSTR
PUOV	0	0	PORTB5•PUD	PORTB4•PUD
DDOE	EXT • (INTRC+AS2)	INTRC+AS2	SPE•MSTR	SPE•MSTR
DDOV	0	0	0	0
PVOE	0	0	SPE•MSTR	SPE•MSTR
PVOV	0	0	SCK出力	SPI従装置出力
DIEOE	EXT•(INTRC+AS2)	INTRC+AS2	0	0
DIEOV	0	0	0	0
DI	-	_	SCK入力	SPI主装置入力
AIO	発振増幅器出力	発振増幅器/クロック入力	-	-

注: 1. INTRCは校正付き内蔵RC発振器が(CKSELヒューズによって)選ばれることを意味します。

2. EXTは外部RC発振器または外部クロックが(CKSELヒューズによって)選ばれることを意味します。

表24. ポートB3~0の交換機能用交換信号

信号名	PB3/MOSI/OC2	PB2/SS/OC1B	PB1/OC1A	PB0/ICP1
PUOE	SPE•MSTR	SPE•MSTR	0	0
PUOV	PORTB3•PUD	PORTB2•PUD	0	0
DDOE	SPE•MSTR	SPE•MSTR	0	0
DDOV	0	0	0	0
PVOE	SPE·MSTR+OC2許可	OC1B許可	OC1A許可	0
PVOV	SPI主装置出力+OC2	OC1B	OC1A	0
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	SPI従装置入力	SPI SS	1	ICP1入力
AIO	_	_	_	_



ポートCの交換機能

ポートCピンの交換機能は表25.で示されます。

表25. ポートCピンの交換機能

ポート ピン	交換機能
PC6	RESET (リセット ピン)
PC5	ADC5 (A/D変換チャネル5入力) SCL (2線直列バス クロック入出力)
PC4	ADC4 (A/D変換チャネル4入力) SDA (2線直列バス データ入出力)
PC3	ADC3 (A/D変換チャネル3入力)
PC2	ADC2 (A/D変換チャネル2入力)
PC1	ADC1 (A/D変換チャネル1入力)
PC0	ADC0 (A/D変換チャネル0入力)

交換ピンの設定は次のとおりです。

• RESET - ポートC ビット6 : PC6

RESET: リセット ピン。RSTDISBLヒューズがプログラム(0)されると、このピンは通常のI/Oピンとして機能し、デバイスはリセット元として電源ONリセットと低電圧リセットに頼らなければなりません。RSTDISBLヒューズが非プログラム(1)にされると、このピンにリセット回路が接続され、このピンはI/Oピンとして使えません。

PC6がリセット ピンとして使われると、PORTC6, DDC6, PINC6は全て0を読みます。

SCL/ADC5 - ポートC ビット5 : PC5

SCL: 2線直列インターフェース クロック。2線直列インターフェースを許可するために2線直列インターフェース制御レシ、スタ(TWCR)の2線直列インターフェース動作許可(TWEN)ビットが設定(1)されると、PC5は(通常の)ボートから切り離されて、2線直列インターフェース用直列クロック入出力ピッとなります。この動作では入力信号上の50nsより短いスパーク(瞬間雑音)を消去するためにスパーイク除去器があり、スリューレート(上昇/下降速度)制限付きオープントレイン駆動部によって駆動されます。

ADC5: PC5はA/D変換チャネル5入力としても使えます。A/D変換チャネル5入力がデジタル電源を使うことに注意してください。

• SDA/ADC4 - ポートC ビット4: PC4

SDA: 2線直列インターフェースデータ。2線直列インターフェースを許可するために2線直列インターフェース制御レシ、スタ(TWCR)の2線直列インターフェース動作許可(TWEN)ビットが設定(1)されると、PC4は(通常の)ポートから切り離されて、2線直列インターフェース用直列データ入出力ピンになります。この動作では入力信号上の50nsより短いスパイク(瞬間雑音)を消去するためにスパイク除去器があり、スリューレート(上昇/下降速度)制限付きオープントレイン駆動部によって駆動されます。

ADC4: PC4はA/D変換チャネル4入力としても使えます。A/D変換チャネル4入力がデジタル電源を使うことに注意してください。

• ADC3 - ポートC ビット3 : PC3

ADC3: PC3はA/D変換チャネル3入力としても使えます。A/D変換チャネル3入力がアナログ電源を使うことに注意してください。

• ADC2 - ポートC ビット2 : PC2

ADC2: PC2はA/D変換チャネル2入力としても使えます。A/D変換チャネル2入力がアナログ電源を使うことに注意してください。

• ADC1 - ポートC ビット1 : PC1

ADC1: PC1はA/D変換チャネル1入力としても使えます。A/D変換チャネル1入力がアナログ電源を使うことに注意してください。

ADC0 - ホートC ビット0: PC0

ADCO: PC0はA/D変換チャネル0入力としても使えます。A/D変換チャネル0入力がアナログ電源を使うことに注意してください。

表26.と表27.はポートCの交換機能を35頁の図25.で示される交換信号に関連付けます。

表26. ポートC6~4の交換機能用交換信号

信号名	PC6/RESET	PC5/SCL/ADC5	PC4/SDA/ADC4
PUOE	RSTDISBL	TWEN	TWEN
PUOV	0	PORTC5•PUD	PORTC4•PUD
DDOE	RSTDISBL	TWEN	TWEN
DDOV	0	SCL出力	SDA出力
PVOE	0	TWEN	TWEN
PVOV	0	0	0
DIEOE	RSTDISBL	0	0
DIEOV	0	0	0
DI	-	_	_
AIO	リセット入力	ADC5入力/SCL入力	ADC4入力/SDA入力

注: 許可されると、2線直列インターフェースはPC4とPC5ピン出力のスリューレート制御を許可します。これは図で示されていません。 加えて、ポート図で示されるAIO出力と2線直列インターフェース(TWI)部間にスパイク除去器が接続されます。

表27. ポートC3~0の交換機能用交換信号

信号名	PC3/ADC3	PC2/ADC2	PC1/ADC1	PC0/ADC0	
PUOE	0	0	0	0	
PUOV	0	0	0	0	
DDOE	0	0	0	0	
DDOV	0	0	0	0	
PVOE	0	0	0	0	
PVOV	0	0	0	0	
DIEOE	0	0	0	0	
DIEOV	0	0	0	0	
DI	-	_	-	_	
AIO	ADC3入力	ADC2入力	ADC1入力	ADC0入力	



ポートDの交換機能

ポートDピンの交換機能は表28.で示されます。

表28 ポートDピンの交換機能

ホ°−ト ピン	交換機能
PD7	AIN1 (アナロケ・比較器反転入力)
PD6	AINO (アナログ・比較器非反転入力)
PD5	T1 (タイマ/カウンタ1 外部クロック入力)
PD4	XCK (USART 外部クロック入出力) T0 (タイマ/カウンタ0 外部クロック入力)
PD3	INT1 (外部割り込み1 入力)
PD2	INTO (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。

AIN1 – ホ[°]ートD ビット7 : PD7

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 - ホートD ビット6: PD6

AINO: 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 - ホートD ビット5: PD5

T1: タイマ/カウンタ1の外部クロック入力ピンです。

• XCK/T0 - ポートD ビット4 : PD4

XCK: USARTの外部クロック入出力ピンです。 **TO**: タイマ/カウンタ0の外部クロック入力ピンです。

• 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

INTO: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、このピンはポートD方向レシブスタ(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DDD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29.と表30.はポートDの交換機能を35頁の図25.で示される交換信号に関連付けます。

表20	#°-kD	1~1m	 办场 继 能	用交換信号	
AX ZJ.	ハードレル	40)	X 1X 1DD FILE	用义役后与	

信号名	PD7/AIN1	PD6/AIN0	PD5/T1	PD4/XCK/T0
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	UMSEL
PVOV	0	0	0	XCK出力
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	_	_	T1入力	XCK入力/T0入力
AIO	AIN1入力	AIN0入力	-	-

表30. 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•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	_	_	_	_



I/Oポート用レジスタ

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

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

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

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

PINB - ホートB入力レジスタ (Port B Input Address)

ピット	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								

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

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

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

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

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

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

PORTD - ポートD出力レジスタ (Port D Data Register)

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

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

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

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

ピット	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								



外部割り込み

外部割り込みはINT0とINT1ピンによって起動されます。許可したなら、例えINT0,1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レヘブルによって起動できます。これはMCU制御レジスタ(MCUCR)の詳述で示される設定です。外部割り込みが(Low)レヘブル起動として設定、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INT0とINT1の上昇端または下降端割り込みの認知は16頁の「クロック系統とその配給」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レヘブル割り込みは非同期に検知されます。これはそれらの(Lowレヘブル)割り込みがアイブル動作以外の休止形態からもデブブイスを起動するのに使えることを意味します。I/Oクロックはアイブル動作を除く全ス休止形態で停止されます。

レヘ、ル起動割り込みがハ。ワーダウン動作からの起動に使われる場合、変化したレヘ、ルはMCUを起動するために一定時間保たれなければならないことに注意してください。これはMCUの雑音不安定性を減らします。変化したレヘ、ルはウォッチト、ッグ発振器クロックによって2度採取されます。ウォッチト、ッグ発振器の周期は5.0V、25℃で(公称)1μsです。ウォッチト、ッグ発振器の周波数は163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ヘ、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16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。このレヘ、ルがウォッチト、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。この必要とされるレヘ、ルはレヘ、ル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保たれなければなりません。

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

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。

ピット	7	6	5	4	3	2	1	0	
\$35 (\$55)	SE	SM2	SM1	SM0	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外部ピッとよって活性的(有効)にされます。割り込みを活性にする外部INT1ピッの端(エッジ)とレヘルは表31.で定義されます。INT1ピッの値は端検出に先立って採取されています。端または論理変化割り込みが選ばれる場合、1クロック周期よりも長く留まるハールスは割り込みを生成します。より短いハールスは割り込みの発生が保証されません。Lowレヘール割り込みが選ばれる場合、そのLowレヘールは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表31. 外	部割り込	込み1(INT1)割り込み条件
ISC11	ISC10	割り込み発生条件
0	0	INT1ピンがLowレベルで発生。
0	1	INT1ピンの論理変化(端)。
1	0	INT1ピンの下降端で発生。
1	1	INT1ピンの上昇端で発生。

● 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(INTO)ビットが設定(1)される場合のINTO外部ピッとよって活性的(有効)にされます。割り込みを活性にする外部INTOピッの端(エッジ)とレヘルは表32.で定義されます。INTOピック値は端検出に先立って採取されています。端または論理変化割り込みが選ばれる場合、1クロック周期よりも長く留まるハッルスは割り込みを生成します。より短いハッルスは割り込みの発生が保証されません。Lowレヘッル割り込みが選ばれる場合、そのLowレヘッルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表32. 外	部割り込	<u>込み0(INT0)割り込み条件</u>
ISC01	ISC00	割り込み発生条件
0	0	INT0ピンがLowレベルで発生。
0	1	INT0ピンの論理変化(両端)。
1	0	INT0ピンの下降端で発生。
1	1	INT0ピンの上昇端で発生。

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

ピット	7	6	5	4	3	2	1	0	
\$3B (\$5B)	INT1	INT0	-	-	-	-	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R	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,ISC10)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レヘルのどれで活性(有効)にされるかを定義します。例え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割り込みへブタから実行されます。

● ビット6 - INTO: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,ISC00)はこの外部割り込みがINT0ピンの上昇端、下降端、両端、またはLowレベルのどれで活性(有効)にされるかを定義します。例え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。外部割り込み要求の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。

GIFR - 一般割り込み要求フラク゛レジスタ (General Interrupt Flag Register)

ピット	7	6	5	4	3	2	1	0	
\$3A (\$5A)	INTF1	INTF0	-	-	-	-	-	-	GIFR
Read/Write	R/W	R/W	R	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ピン上の出来事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)ピットが設定(1)なら、MCUは対応する割り込みへ、クタへ飛びます。このフラク゚は割り込みルーチンが実行されると解除(0)されます。代わりにこのフラク゛は論理1を書くことによっても解除(0)できます。INT1がレヘ・ル割り込みとして設定されると、このフラク゛は常に解除(0)されます。

● ビット6 - INTFO:外部割り込み0要求フラグ(External Interrupt Flag0)

INTOピン上の出来事が割り込み要求を起動すると、INTFO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(INTO)ビットが設定(1)なら、MCUは対応する割り込みへ、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。代わりにこのフラク゛は論理1を書くことによっても解除(0)できます。INTOがレヘ・ル割り込みとして設定されると、このフラク゛は常に解除(0)されます。



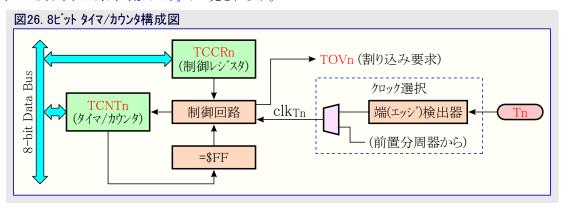
8ビット タイマ/カウンタ0

タイマ/カウンタのは汎用単一チャネルの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。

- 単一チャネル計数器
- 周波数発生器
- 外部事象計数器
- クロック用10ビット前置分周器

概要

この8ビット タイマ/カウンタの簡単化した構成図は**図26**.で示されます。I/Oピンの実際の配置については2頁の「**ピン配置**」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは<mark>赤文字(訳注</mark>: 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47頁の「8ビット タイマ/カウンタ0用レジスタ」で一覧されます。



関係レジスタ

このタイマ/カウンタ(TCNT0)は8ビットのレシ、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シ、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シ、スタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKはこれらのレシ、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TOピンの外部クロック元によってクロック駆動されます。クロック選択論理部はタイマ/カウンタが値を増加するのに使うクロック元と端(エッシ)を制御します。クロック元が選ばれないと、このタイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkTo)として参照されます。

定義

本文書でのレシ、スタとヒ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シ、スタまたはヒット定義に使う時は正確な形式が使われなければなりません(例えばタイマ/カウンタ0のカウンタ値のアクセスに対してのTCNT0のように)。

表33.の定義は本データシートを通して広範囲に渡って使われます。

表33. 用語知	表33. 用語定義						
用語	意味						
BOTTOM	計数器が\$00に到達した時。						
MAX	計数器が\$FF(255)に到達した時。						

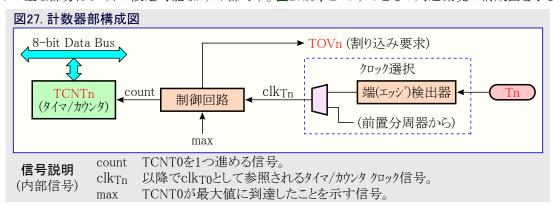


タイマ/カウンタのクロック

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ばれます。クロック元と前置分周器の詳細については48頁の「タイマ/カウンタ0とタイマ/カウンタ1の前置分周器」をご覧ください。

計数器部

8ビット タイマ/カウンタの主な部分は8ビットの設定可能なカウンタ部です。図27.は、このカウンタとその周辺環境の構成図を示します。



このカウンタはタイマ/カウンタ クロック(clkTo)毎に増加(+1)されます。clkToはクロック選択(CS02~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS02~0=000)時にタイマ/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(clkTo)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)。

計数器部の動作

計数方向は常に上昇(増加)で、カウンタの解除はありません。このカウンタは8ビット最大値(MAX=\$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0溢れ(TOV0)フラグはTCNT0が0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0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。新しいカウンタ値は何時でも書けます。

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(clkT0)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ケが設定(1)される時の情報を含みます。図28.は基本的なタイマ/カウンタ動作についてのタイミング データを含みます。この図はMAX値付近の一連の計数を示します。

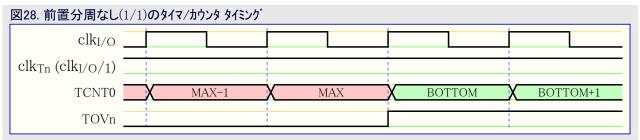
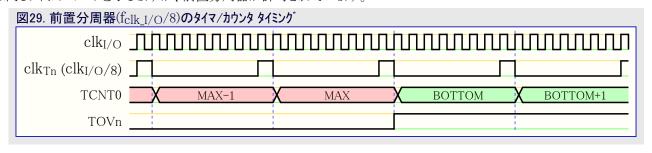


図29.は同じタイミングデータを示しますが、前置分周器が許可されています。





8ビット タイマ/カウンタ0 用レシ スタ

TCCRO - タイマ/カウンタ0制御レジスタ (Timer/CounterO Control Register)

ピット	7	6	5	4	3	2	1	0	_
\$33 (\$53)	_	-	-	-	-	CS02	CS01	CS00	TCCR0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

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

この3ビットのクロック選択ビットはタイマ/カウンタ0(TCNT0)によって使わ 表34. タイマ/カウンタ0入力クロック選択 れるクロック元を選びます。

タイマ/カウンタ0に外部ピン動作が使われる場合、例えT0ピンが出力 として設定されてもTO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。

l	CS02	CS01	CS00	クロック元 (clk _{T0})
	0	0	0	停止(タイマ/カウンタ0動作停止)
I	0	0	1	clk _{I/O} (前置分周なし)
I	0	1	0	clk _{I/O} /8 (8分周)
I	0	1	1	clk _{I/O} /64 (64分周)
I	1	0	0	clk _{I/O} /256 (256分周)
I	1	0	1	clk _{I/O} /1024 (1024分周)
I	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)

TCNT0 - タイマ/カウンタ0 (Timer/Counter0)

ピット	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/カウンタレジブタはタイマ/カウンタ部の8ビットカウンタに読み書き操作両方に対して直接アクセスを許します。

TIMSK - タイマ/カウンタ割り込み許可レシ、スタ (Timer/Counter Interrupt Mask Register)

					•	-	•	-		
	ピット	7	6	5	4	3	2	1	0	
	\$39 (\$59)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	TIMSK
R	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0 - TOIE0 : タイマ/カウンタ0溢れ割り込み許可 (Timer/Counter0 Overflow Interrupt Enable)

ステータス レシ、スタ(SREG)の全割り込み許可(I)ビットが設定(1)され、このTOIE0ビットが1を書かれると、タイマ/カウンタの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・レシ、スタ(TIFR)でタイマ/カウンタ0溢れ割り込み要求フラグ・ (TOV0)が設定(1)されると、対応する割り込みが実行されます。

TIFR - タイマ/カウンタ割り込み要求フラク゛レシ゛スタ (Timer/Counter Interrupt Flag Register)

ピット	7	6	5	4	3	2	1	0	
\$38 (\$58)	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	TIFR
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0 - TOV0 : タイマ/カウンタ0溢れ割り込み要求フラグ(Timer/Counter0 Overflow Interrupt Flag)

タイマ/カウンタ0溢れが起こると、このTOV0ビットが設定(1)されます。TOV0は対応する割り込み処理ベクタを実行するとき、ハードウェアによっ て解除(0)されます。代わりにTOV0はこのフラク・に論理1を書くことによっても解除(0)されます。ステータス レシ 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)ビットとTOV0が設定(1)されれば、タイマ/カ ウンタ0溢れ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ていますが、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(fclk_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fclk I/O/8, fclk I/O/64, fclk I/O/256, fclk I/O/1024の何れかの周波数です。

前置分周器リセット

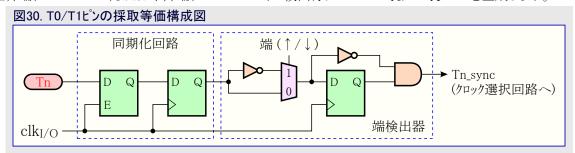
この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ0とタイマ/カウンタ1によって共用されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが使われる状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステムクロック周期数はNが前置分周値(8,64,256,1024)とすると、1~N+1システムクロック周期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

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/カウンタについての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(fclk_T0/fclk_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端(エッジ)検出器を通して通過されます。図 30.はT0/T1同期化と端検出器論理回路の機能等価構成図を示します。レジスタは内部システム クロック(fclk_I/O)の上昇端でクロック駆動されます。 ラッチは内部システム クロックのHigh区間で通過(Low区間で保持)です。

端検出器は上昇端(CSn2~0=111)または下降端(CSn2~0=110)の検出毎に1つのclkT0/clkT1パルスを生成します。



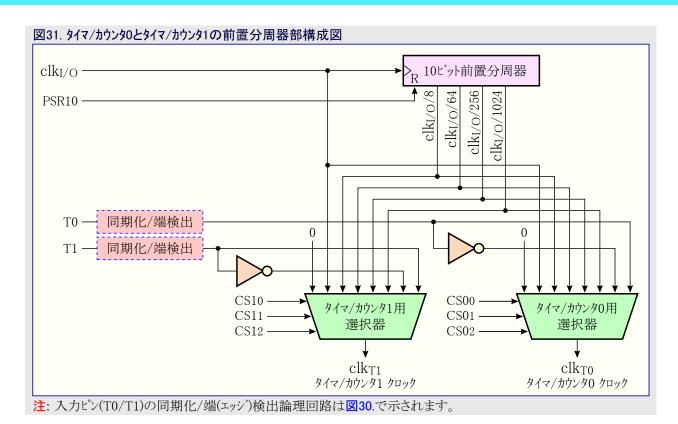
同期化と端検出器論理回路はT0/T1ピンへ印加された端から計数器が更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

クロック入力の許可と禁止はT0/T1が最低1システム クロック周期に対して安定してしまっている時に行われなければならず、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。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(fextelk fclk_I/O/2)であることが保証されなければなりません。端検出器が採取を使うため、検出できる外部クロックの最大周波数は採取周波数の半分です(ナイキストの標本化定理)。けれども発振元(クリスタル発振子、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fclk 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。





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

ピット	7	6	5	4	3	2	1	0	•
\$30 (\$50)	_	_	_	-	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット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として読みます。

(訳補) SFIORのビット4は元のADHSMが削除されました。この結果、次の項目が必要と思われます。

● ビット4 - Res: 予約 (Reserved Bit)

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



16ビット タイマ/カウンタ1

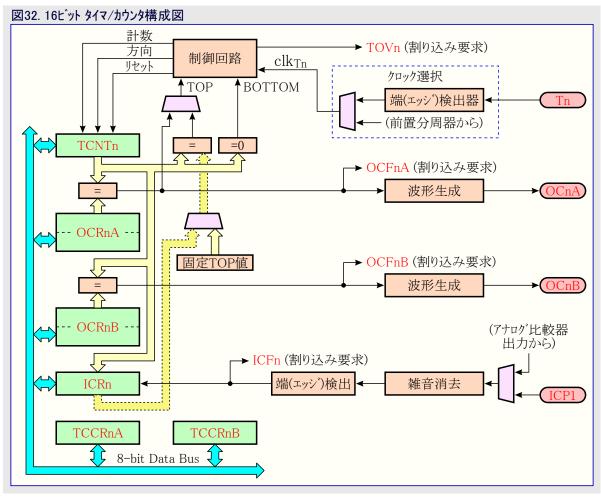
この16ビット タイマ/カウンタ部は正確なプログラム実行タイミング(事象管理)、波形生成、信号タイミング計測を許します。主な特徴は次に示されます。

- 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

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

この16ビット タイマ/カウンタの簡単化した構成図は**図32**.で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは<mark>赤文字</mark> (<mark>訳注</mark>: 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64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。



注: タイマ/カウンタ1のピンの配置と記述については2頁の「ピン配置」、36頁の表22、40頁の表28.を参照してください。



関係レジスタ

タイマ/カウンタ(TCNT1)、比較レシ、スタ(OCR1A,OCR1B)、捕獲レシ、スタ(ICR1)は全て16ビットレシ、スタです。16ビットレシ、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52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制御レシ、スタ(TCCR1A,TCCR1B)は8ビットレシ、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端(エッシ)を制御します。クロック元が選ばれないと、タイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkT1)として参照されます。

2重緩衝化した比較レシ、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。56頁の「**比較出力部**」をご覧ください。この比較一致発生は比較出力割り込み要求の発生に使える比較一致割り込み要求フラグ(OCF1A,OCF1B)も設定(1)します。

捕獲レジスタ(ICR1)は捕獲起動(ICP1)ピンまたはアナログ比較器出力(125頁の「**アナログ比較器**」参照)のどちらかの外部(端で起動された) 事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波部(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出力生成用に使えません。 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が代わりに使え、PWM出力として使われるべきOCR1Aを開放します。

定義

次の定義は本文書を通して広範囲に使われます。

表35. 用語定	2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。
ТОР	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シ、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レシ、スタアトンスの位置。
- タイマ/カウンタ割り込み関係レシ、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レシ、スタ内のビット位置。
- ・割り込みべりタ。

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。



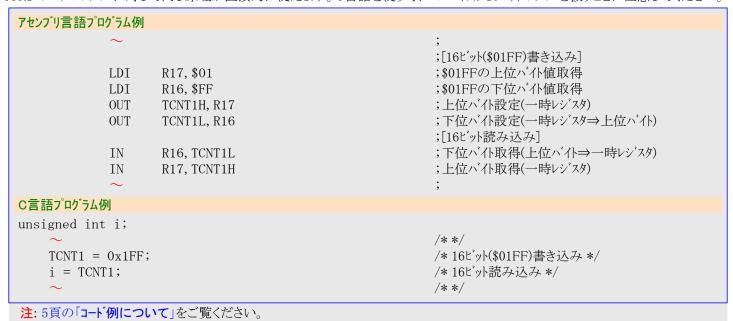
16ビット レジスタのアクセス

TCNT1,OCR1A,OCR1B,ICR1は8ビット バス経由でAVR CPUによってアクセスできる16ビット レジスタです。この16ビット レジスタは2回の読みまたは書き操作を使ってバイト アクセスされなければなりません。16ビット タイマ/カウンタは16ビット アクセスの上位バイトの一時保存用に1つの8 ビット レジスタを持ちます。16ビット タイマ/カウンタ内の全ての16ビット レジスタ間で、この同じ一時レジスタが共用されます。下位バイト アクセスが16ビット読み書き動作を起動します。16ビット レジスタの下位バイトがCPUによって書かれると、一時レジスタに保存した上位バイトと書かれた下位バイトは同じクロック周期で両方が16ピット 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。

次のコート・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, ICR1レジスタのアクセスに対して同じ原理が直接的に使えます。C言語を使う時にコンパ゚イラが16ビット アクセスを扱うことに注意してください。



アセンブリ言語コート 例はR17:R16レシ 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。



次のコート・例はTCNT1レシ、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:
                                                  ;現全割り込み許可フラグ(I)を保存
          IN
                 R18, SREG
          CLI
                                                  ;全割り込み禁止
                                                  ;TCNT1下位バイト取得(上位バイト⇒一時レジスタ)
                 R16, TCNT1L
          ΙN
                 R17, TCNT1H
                                                  ;TCNT1上位バイト取得(一時レジスタ)
          ΤN
          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 レシブスター時保存変数定義 */
   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 */
                                                  /* 全割り込み許可フラグ(I)を復帰 */
   SREG = sreg;
                                                  /* TCNT1値で呼び出し元へ復帰 */
   return i;
```

注: 5頁の「**コート'例について**」をご覧ください。

アセンフリ言語コート 例はR17:R16レジスタ対にTCNT1値を戻します。

次のコート・例はTCNT1レシ、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
                                                  ;TCNT1上位バイ設定(一時レジスタ)
           OUT
                 TCNT1H, R17
           OUT
                                                  ;TCNT1下位バイト設定(一時レジスタ⇒上位バイト)
                 TCNT1L, R16
           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)
                                                  /* ステータス レジスタ一時保存変数定義 */
   unsigned char sreg;
   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 */
                                                  /* 全割り込み許可フラグ(I)を復帰 */
   SREG = sreg;
```

注: 5頁の「コート・例について」をご覧ください。

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

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。

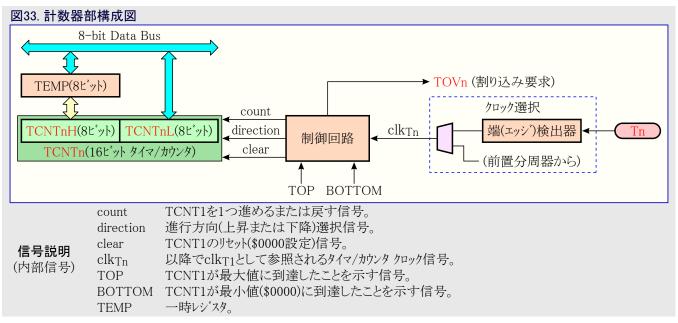


タイマ/カウンタのクロック

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/カウンタ制御レシ、スタB(TCCR1B)に配置されたクロック選択(CS12~0)ビットによって制御されるクロック選択論理回路で選ばれます。クロック元と前置分周器の詳細については48頁の「タイマ/カウンタ0とタイマ/カウンタ1の前置分周器」をご覧ください。

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33.はこのカウンタとその周辺の構成図を示します。



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

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

計数順序(方法)はタイマ/カウンタ制御レシ、スタA(TCCR1A)とタイマ/カウンタ制御レシ、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59頁の「動作種別」をご覧ください。

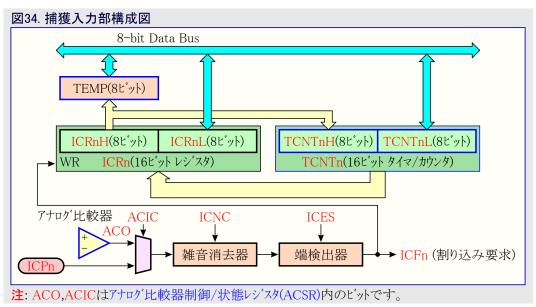
タイマ/カウンタ溢れ(TOV1)フラグはWGM13~0ビットによって選ばれた動作種別に従って設定(1)されます。TOV1はCPU割り込み発生に使えます。



捕獲入力部

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

捕獲入力部は**図34**.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(<mark>訳注</mark>:原文は灰色背景)で示されます。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号を示します。



捕獲起動入力(ICP1)ピン若しくは代わりにアナログ比較器出力(ACO)で論理レベルの変化(出来事)が起き、その変化が端(エッジ)検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF1)はTCNT1値がICR1に複写されるのと同じシステム クロックで設定(1)されます。許可(I=1, TICIE1=1)ならば捕獲割り込み要求フラグは捕獲割り込みを発生します。ICF1は割り込みが実行されると自動的に解除(0)されます。代わりにこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。

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

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

16ビット レジスタ アクセス法のより多くの情報については52頁の「16ビット レジスタのアクセス」を参照してください。

捕獲起動元

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

捕獲起動入力(ICP1)ピンとアナログ比較器出力(ACO)の両入力は、T1ピン(48頁の**図30**.参照)についてと同じ技法を使って採取されます。端検出器も全く同じです。けれども雑音消去が許可されると、付加論理回路が端検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されないなら、雑音消去器と端検出器の入力は常に許可されることに注意してください。

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

雑音消去器

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

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



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充分なプロセッサ能力を当てがうことです。2つの出来事間の時間が際どいとします。次の出来事が起こる前に捕獲した捕獲レジスタ(ICR1)の値をプロセッサが読めなかった場合、ICR1は新しい値で上書きされます。この場合、捕獲の結果は不正にされます。

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

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

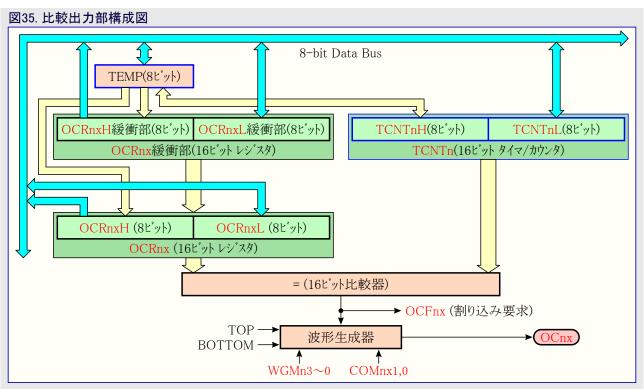
外部信号のデューティ比測定は各捕獲後に起動端が変更されることを必要とします。検出端の変更はICR1が読まれてしまった後に可能な限り早く行われなければなりません。端(エッシ)の変更後、捕獲割り込み要求フラケ(ICF1)はソフトウェア(I/Oビット位置への論理1書き込み)によって解除(0)されなければなりません(訳補:端変更によってICF1が設定(1)されることを想定)。周波数のみの測定について、(割り込み処理が使われる場合)ICF1の解除(0)は必要とされません。

比較出力部

この16ビット比較器はTCNT1と比較レシ、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(59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35.は比較出力部の構成図を示します。レシブ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。





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は上位パイトー時レジ、スタ(TEMP)経由で読まれません。けれども他の16ピット レジ、スタをアクセスする時のように下位パイトを先に読むのは良い習慣です。OCR1x書き込みは16ピ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。上位パイト(OCR1xH)が先に書かれなければなりません。上位パイト(OCR1xH)が先に書かれなければなりません。上位パイト(OCR1xH)が先に書かれなければなりません。上位パイト(OCR1xH)が方に優野であると、一時レジ、スタ内の)上位パイトは(下位パイト書き込みと)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52頁の「16ビット レジスタのアクセス」を参照してください。

(<mark>訳注</mark>) 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ないかに拘らず、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。

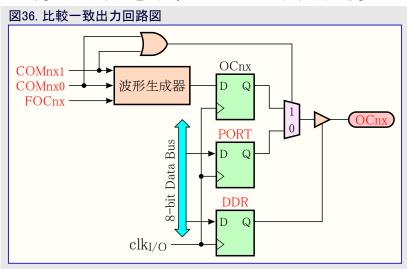
OC1xの初期設定はポート ピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)ストローブ ビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

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ピットを使います。次にCOM1x1,0ピットはOC1xピン出力元を制御します。図36.はCOM1x1,0ピット設定によって影響される論理回路の簡単化した図を示します。図のI/Oレジスタ、I/Oピット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM1x1,0ピ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 リセットが起こると、OC1xレジスタは0にリセットされます。



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36、表37、表38.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。64頁の「16ビット タイマ/カウンタ1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。全ての動作種別に対してCOM1x1,0=00設定は次の比較一致で実行すべきOC1xレジスタの動きがないことを波形生成器へ告げます。非PWM動作での比較出力動作については64頁の表36.を参照してください。高速PWM動作については64頁の表37.、位相基準PWMと位相/周波数基準PWMについては64頁の表38.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)ストロープ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。58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63頁の「タイマ/カウンタ1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1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。

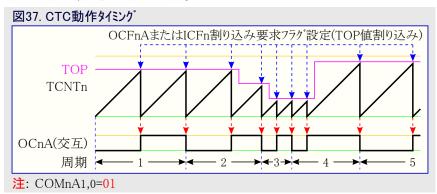
捕獲入力部は標準動作での使用が容易です。けれども外部の事象間の最大間隔がタイマ/カウンタの分解能(16ビット長)を越えてはならないことに気付いてください。事象間の間隔が長すぎる場合、捕獲部に対して分解能を拡張するために、タイマ/カウンタ溢れ割り込みまたは前置分周器が使われなければなりません。

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37**.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。



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値はそのビッ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、ポートビッ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fOC1A=fclk_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。

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

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

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

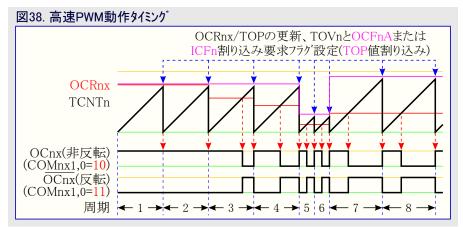
高速PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13 \sim 0=0101,0110,0111)、ICR1 値(WGM13 \sim 0=1110)またはOCR1A値(WGM13 \sim 0=1111)のどれかと一致するまで増加されます。そしてカウンタは(一致の)次のタイマ/カウンタ クロック周期で解除(\$0000)されます。高速PWM動作のタイミング図は図38.

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

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値はタイング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線は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)フラケ 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割り込み要求フラケ 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OCR1 x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時に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)レジスタはその後に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いきます。ICR1を使うことにより、OC1AでのPWM出力を生成するためにOCR1Aが自由に使えます。けれども基準PWM周波数が(TOP値を変更することによって)動的に変更される場合、OCR1Aが2重緩衝機能のため、TOPとしてOCR1Aを使うことは明らかに良い選択です。

高速PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます。64頁の表37.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。 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あたります。OCR

1xがBOTTOM(\$0000)に等しく設定されると、出力はTOP+1 タイマ/カクンタ クロック周期毎の狭いスパイク(パルス)になるでしょう。TOPに等しいOCR1x設定は(COM1x1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

高速PWM動作で(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC1A設定(COM1A1,0=01)によって達成できます。これはTOP値を定義するのにOCR1Aが使われる(WGM13~0=1111)の場合にだけ適用されます。生成された波形はOCR1Aが0 (\$0000)に設定される時に $f_{\text{OC1A}}=f_{\text{clk_I/O}}/2$ 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。



位相基準PWM動作

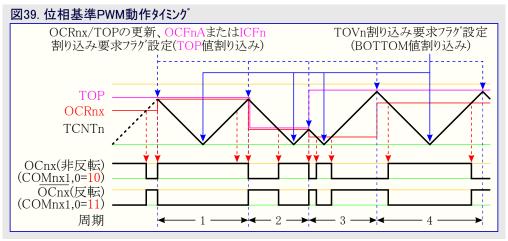
位相基準パルス幅変調(PWM)動作(WGM13~0=0001,0010,0011,1010,1011)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作を基準とした位相/周波数基準PWMと似ています。カウンタはBOTTOM(\$0000)からTOPへ、そしてその後にTOPからBOTTOMへを繰り返し計数します。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(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。

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

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

ク周期間、TOPと等しくなります。位相基準PWM動作のタイシンが図は図39.で示されます。この図はOCR1AかICR1がTOPを定義するために使われる時の位相基準PWM動作を示します。TCNT1値はタイシンが図で両傾斜動作(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。



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、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Oで隠(に置換)されることに注意してください。図39.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値を使うとき、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OC1xt°ンでのPWM波形の生成を許します。COM1x1,0t°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(64頁の表38.をご覧ください)。実際のOC1x値はそのポート t°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。

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

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あたります。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(COM1x1,0=10)での比較出力(OC1x)は上昇計数中のTCNT1とOCR1x間の比較一致で解除(0)され、下降計数中の比較一致で設定(1)されます。反転出力動作(COM1x1,0=11)での動作は逆にされます。両傾斜(三角波)動作は単一傾斜(鋸波)動作よりも低い最大動作周波数になります。けれども両傾斜(三角波)PWM動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

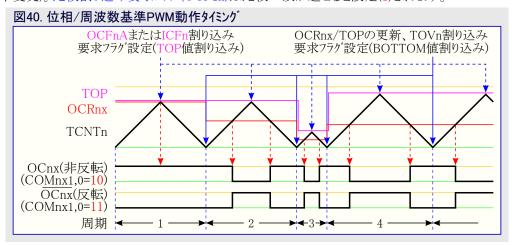
位相基準と位相/周波数基準PWM動作間の主な違いはOCR1xレジスタがOCR1x緩衝部によって更新される時(<mark>訳補</mark>:TOPとBOTTO M)です(図39.と図40.参照)。

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。

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

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40**.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。



タイマ/カウンタ溢れ(TOV1)フラグはOCR1xレシ、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、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40.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いきます。ICR1を使うことにより、OC1AでのPWM出力を生成するためにOCR1Aが自由に使えます。けれども基準PWM周波数が(TOP値を変更することによって)動的に変更される場合、OCR1Aが2重緩衝機能のため、TOPとしてOCR1Aを使うことは明らかに良い選択です。

位相/周波数基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットを'10'に設定することは非反転 PWM出力を作成し、反転PWM出力はCOM1x1,0を'11'に設定することで生成できます(64頁の表38.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OCR1 x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。

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

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あたります。非反転PWM動作では OCR1xがBOTTOMに等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに 対する出力は逆の論理値になります。

TOP値定義にOCR1Aが使われ(WGM13~0=1001)、COM1A1,0=01なら、OC1A出力はデューティ比50%で交互に変化します。



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(clkT1)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クが設定(1)される時、そしてOCR1xレシ、スタがOCR1x緩衝値で更新される時(2重緩衝部を使う動作種別のみ)の情報を含みます。図41.はOCF1xの設定についてのタイングの図を示します。

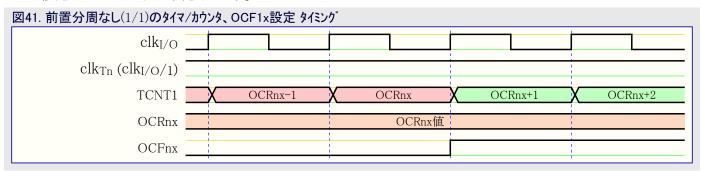


図42.は同じタイミングデータを示しますが、前置分周器が許可されています。



図43.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。

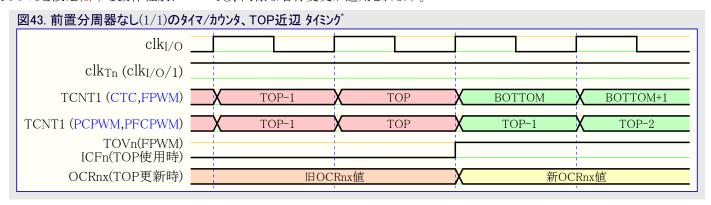
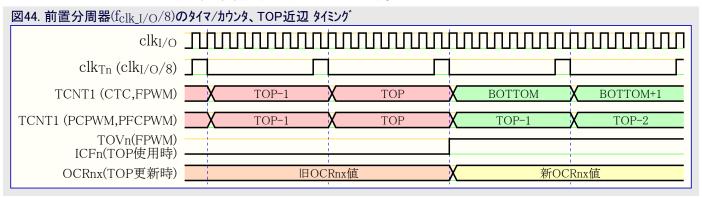


図44.は同じタイミングデータを示しますが、前置分周器が許可されています。





16ビット タイマ/カウンタ1 用レジスタ

TCCR1A - タイマ/カウンタ1制御レシ、スタA (Timer/Counter1 Control Register A)

ピット	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	

- ビットア,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36**.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36. 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37.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37. 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OC1xt°ンへ出力(非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC1xt°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、BO TTOM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60頁の「高速PWM動作」をご覧ください。

表38.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38. 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OC1xtシへ出力
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61頁の「位相基準P WM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ビットがストロープとして実行されることに注意してください。それによって強制された比較の効果を決めるのはCOM1x1,0ビットに存在する値です。

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

FOC1A/FOC1Bビットは常に0として読みます。

● 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39.参照)。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。59頁の「動作種別」をご覧ください。

主って	波形牛成種別選択	
オマ ひり.	次 Ⅲ~ 〒 Ⅳ. ★里 川 1共 4八	

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



TCCR1B - タイマ/カウンタ1制御レシ、スタB (Timer/Counter1 Control Register B)

ピット	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 (ICP1)ピンからの入力が濾波されます。この濾波機能はそれが出力を更新するのに、連続4回等しく評価されたICP1ピンの採取を必要とします。雑音消去器が許可されると、捕獲入力はそれによって4発振器(システム クロック)周期遅らされます。

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

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

捕獲がICES1設定に従って起動されると、カウンタ値が捕獲レシ、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ク (ICF1)も設定(1)し、そしてこれは、この割り込みが許可されていれば捕獲入力割り込みを起こすのに使えます。

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

ビット5 - Res: 予約 (Reserved)

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

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

TCCR1AのWGM11,0ビット記述をご覧ください。

べきクロック元を選びます。 図41.と図42.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

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 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40. タイ	表40. 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)							

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

ピット	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	
ピット	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、スタをアクセスする時に上位と下位の両バイトが同時に読み書きされるのを保証するため、このアクセスは8ビット上位バイトー時レジ、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。



OCR1AH,OCR1AL (OCR1A) - タイマ/カウンタ1 比較Aレシ、スタ (Timer/Counter1 Output Compare Register A)

ピット	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	
ピット	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	

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

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

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

この比較レシ、スタは容量が16ビットです。CPUがこれらのレシ、スタへ書く時に上位と下位の両バイが同時に書かれるのを保証するため、このアクセスは8ビット上位バイトー時レシ、スタ(TEMP)を使って実行されます。この一時レシ、スタは他の全ての16ビット レシ、スタのようとで覧ください。 52頁の「16ビット レジ、スタのアクセス」をご覧ください。

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

ピット	15	14	13	12	11	10	9	8	
\$27 (\$47)	(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	
ピット	7	6	5	4	3	2	1	0	_
\$26 (\$46)								(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	

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

この捕獲レシ、スタは容量が16ビットです。CPUがこれらのレシ、スタをアクセスする時に上位と下位の両バイが同時に読まれるのを保証するため、このアクセスは8ビット上位バイトー時レシ、スタ(TEMP)を使って実行されます。この一時レシ、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。52頁の「16ビットレジ、スタのアクセス」をご覧ください。



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

ピット	7	6	5	4	3	2	1	0	
\$39 (\$59)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	TIMSK
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	

注: このレシ、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5 - 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(28頁の「割り込み」参照)が実行されます。

● ビット4 - 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(28頁の「割り込み」参照)が実行されます。

● ビット3 - 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(28頁の「割り込み」参照)が実行されます。

● ビット2 - 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(28頁の「割り込み」参照)が実行されます。

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

ピット	7	6	5	4	3	2	1	0	_
\$38 (\$58)	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	TIFR
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	

注: このレシ、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5 - ICF1: タイマ/カウンタ1捕獲割り込み要求フラグ(Timer/Conter1, Input Capture Flag)

ICP1ピンに捕獲の事象が起こると、このフラケが設定(1)されます。捕獲レシ、スタ(ICR1)が $WGM13\sim0$ によってTOP値として設定されると、ICF1フラケルカウンタがTOP値に到達する時に設定(1)されます。

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

● ビット4 - OCF1A : タイマ/カウンタ1比較A割り込み要求フラグ(Timer/Conter1, 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($\frac{0}{0}$)されます。代わりにこのビット位置へ論理 $\frac{1}{0}$ を書くことによってもOCF1Aは解除($\frac{0}{0}$)できます。

● ビット3 - OCF1B: タイマ/カウンタ1比較B割り込み要求フラグ(Timer/Conter1, Output Compare B Match Flag)

このフラグはカウンタ(TCNT1)値が比較Bレシ、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 $^{\prime}$ 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 $^{\prime}$ ット位置へ論理1を書くことによってもOCF1Bは解除 (0) できます。

● ビット2 - 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65頁の表39.を参照してください。

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



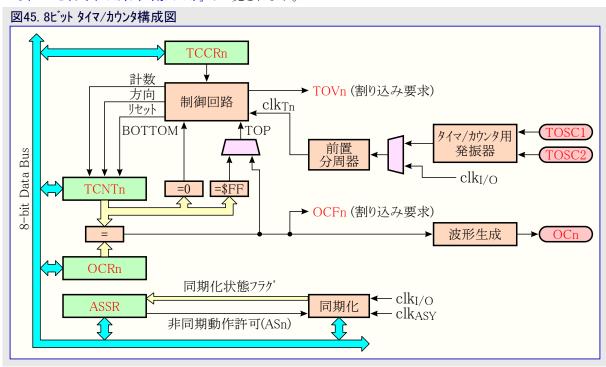
8ビット タイマ/カウンタ2 (PWM, 非同期動作付き)

タイマ/カウンタ2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 (TOV2とOCF2)
- I/O(システム)クロックに依存しない時計用外部32kHzクリスタルからのクロック駆動可能

概要

この8ビット タイマ/カウンタの簡単化した構成図は**図45**.で示されます。I/Oピンの実際の配置については2頁の「**ピン配置**」を参照してください。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(**訳注**:原文は太字)で示されます。デバイス仕様のI/Oレジスタとビット位置は77頁の「8ビット タイマ/カウンタ2用レジスタ」で一覧されます。



関係レジスタ

タイマ/カウンタ(TCNT2)と比較レシ、スタ(OCR2)は8ビットのレシ、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シ、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シ、スタ(TIMSK)で個別に遮蔽(禁止)されます。TIFRとTIMSKはこれらのレシ、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、または本章後ろで詳述されるようにTOSC1/2ピンから非同期にクロック駆動されます。非同期動作は非同期状態レジスタ(ASSR)によって制御されます。クロック選択論理部はタイマ/カウンタが値を増加(または減少)するのに使うク ロック元を制御します。クロック元が選ばれないとこのタイマ/カウンタは動きません。クロック選択論理部からの出力はタイマ/カウンタ クロック(clkT2)として参照されます。

2重緩衝化した比較レジスタ(OCR2)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(OC2)ピンでPWMまたは可変周波数出力を生成するための波形生成器によって使えます。詳細については71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(OCF2)も設定(1)します。

定義

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

表41.の定義は本文書を通して広範囲に渡って使われます。

表41. 用語句	表41. 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。								
ТОР	タイマ/カウンタが指定された固定値(\$FF) またはOCR2値に到達した時。この指 定(TOP)値は動作種別に依存します。								

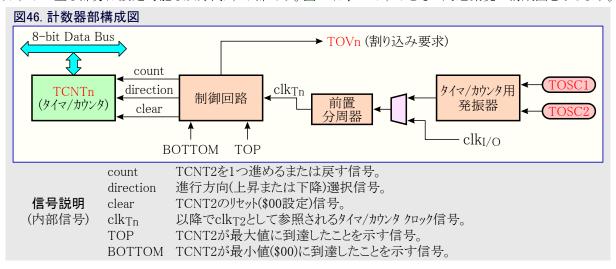


タイマ/カウンタのクロック

このタイマ/カウンタは内部同期または外部非同期クロック元によってクロック駆動できます。既定のクロック元(clk_T2)はMCUクロック(clk_I/0)と同じです。非同期状態レジスタ(ASSR)の非同期動作許可(AS2)ビットが論理1を書かれると、クロック元はTOSC1とTOSC2に接続されたタイマ/カウンタ用発振器から取得されます。非同期動作の詳細については80頁の「ASSR - 非同期状態レジスタ」をご覧ください。クロック元と前置分周器の詳細については82頁の「タイマ/カウンタ2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46.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(clkT2)で解除(\$00)、増加(+1)、または減少(-1)されます。clkT2はクロック選択 (CS22~0)ビットによって選ばれた内部または外部のクロック元から生成できます。クロック元が選ばれない(CS22~0=000)時にタイマ/カウンタは停止されます。けれどもTCNT2値はタイマ/カウンタ クロック(clkT2)が存在するしないに拘らず、CPUによってアクセスできます。CPU書き込みは全てのカウンタ解除や計数動作を無視します(上位優先権を持ちます)。

計数順序(方法)はタイマ/カウンタ制御レジスタ(TCCR2)に配置された波形生成種別(WGM21,0)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOC2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73頁の「動作種別」をご覧ください。

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



比較出力部

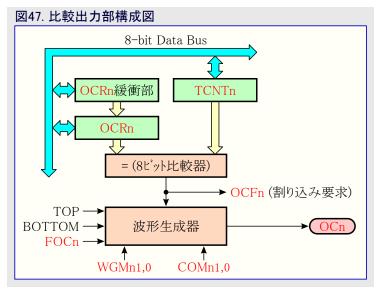
この8ビット比較器はTCNT2と比較レジスタ(OCR2)を継続的に比較します。TCNT2とOCR2が等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタ クロック周期で比較割り込み要求フラグ(OCF2)を設定(1)します。許可(I=1, OCIE2=1)なら、この比較割り込み要求フラグは比較割り込みを発生します。OCF2は割り込みが実行されると自動的に解除(0)されます。代わりにOCF2はこのビット位

置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM21,0)ビットと比較出力選択(COM21,0)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(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。

図47.は比較出力部の構成図を示します。

OCR2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)動作に対しては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OCR2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。

OCR2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OCR2緩衝部をアクセスし、禁止されるとOCR2レジスタを直接アクセスします。



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(FOC2)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(OCF2)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、OC2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(COM21,0ビット設定がOC2 ピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

TCNT2書き込みによる比較一致妨害

TCNT2への全てのCPU書き込みは、例えタイマ/カウンタが停止されていても、次のタイマ/カウンタ クロック周期で起こるどんな比較一致をも妨げます。この特質はタイマ/カウンタ クロックが許可されている時に割り込みを起動することなく、TCNT2と同じ値に初期化されることをOCR2に許します。

比較一致部の使用

どの動作種別でのTCNT2書き込みでも1タイマ/カウンタ クロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるかないかに拘らず、比較出力部を使う場合、TCNT2を変更する時に危険を伴います。TCNT2に書かれた値がOCR2値と等しい場合、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TCNT2値を書いてはいけません。

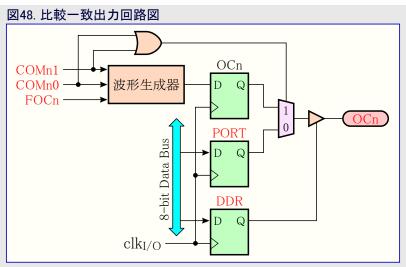
OC2の初期設定はポート ピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC2値を設定する一番簡単な方法は標準動作で強制変更(FOC2)ストロープ ビットを使うことです。波形生成動作種別間を変更する時であっても、OC2(内部)レジスタはその値を保ちます。

比較出力選択(COM21,0)ビットが比較値(OCR2)と共に2重緩衝されないことに気付いてください。COM21,0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(COM21,0)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(OC2)状態定義にCOM21,0ビットを使います。またCOM21,0ビットはOC2ピン出力元を制御します。図48.はCOM21,0ビット設定によって影響を及ぼされる論理回路の簡単化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM21,0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。OC2の状態を参照するとき、その参照はOC2ピンでなく内部OC2レジスタに対してです。



COM21,0ピ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(OC2)によって無効にされます。けれどもOC2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。OC2ピンに対するポート方向レジスタのビット(DDR_OC2)はOC2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OC2状態の初期化を許します。いくつかのCOM21,0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。77頁の「8ビット タイマ/カウンタ2用レジスタ」をご覧ください。

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COM21,0ビットを違うふうに使います。全動作種別に対してCOM21,0=00設定は次の比較一致で実行すべきOC2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77頁の表43.を参照してください。高速PWM動作については77頁の表44、位相基準PWMについては77頁の表45.を参照してください。

COM21,0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(FOC2)ストロープ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(WGM21,0)ビットと比較出力選択(COM21,0)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。COM21,0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COM21,0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76頁の「タイマ/カウンタ2のタイミング」を参照してください。

標準動作

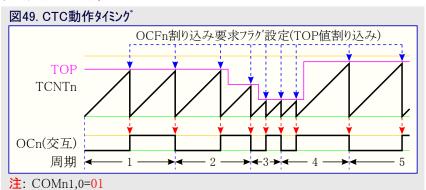
最も単純な動作種別が標準動作(WGM21,0=00)です。この動作種別での計数方向は常に上昇(+)でカウンタ解除は実行されません。カウンタは8ビット最大値(MAX=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV2)フラグはTCNT2が\$00になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV2フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのようになります。けれどもTOV2フラグを自動的に解除(0)するタイマ/カウンタ2溢れ割り込みと組み合わせたタイマ/カウンタの分解能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(WGM21,0=10)ではOCR2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(TC NT2)値がOCR2と一致すると、カウンタは\$00に解除されます。OCR2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49**.で示されます。カウンタ(TCNT2)値はTCNT2とOCR2間で比較一致が起こるまで増加し、そしてその後にカウンタ(TCNT2)は解除(\$00)されます。



OCF27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BOTTO M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。OCR2に書かれた新しい値がTCNT2の現在値よりも低い(小さい)場合、タイマ/カウンタは(その回の)比較一致を失います。その後のカウンタは比較一致が起こるのに先立って、最大値(\$FF)へそして次に\$00から始める計数をしなければならないでしょう。

CTC動作で波形出力を生成するため、OC2出力は比較出力選択(COM21,0)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。OC2値はそのピンに対するデータ方向が出力(DDR_OC2=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OCR2が0(\$00)に設定される時にfoC2=fclk_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。

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

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

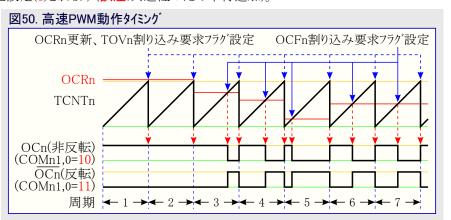
標準動作と同じように、タイマ/カウンタ溢れ(TOV2)フラグはカウンタがMAXから\$00へ計数するのと同じタイマ/カウンタ クロック周期で設定(1)されます。



高速PWM動作

高速 n° ルス幅変調(PWM)動作(WGM21,0=11)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と違い異なります。カウンタはBOTTOMからMAXまで計数し、その後にBOTTOMから再び始めます。非反転比較出力動作(COM21,0=10)での比較出力(OC2)は、TCNT2とOCR2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(COM21,0=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50.で示されます。TCNT2値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。本図は非反転と反転のPWM出力を含みます。赤細線はOCR2値を示し、TCNT2値との交点(接点)がTCNT2とOCR2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(OCF2)はOCR2=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(TOV2)フラグ はカウンタがMAXに到達する時毎に設定(1)されます。割り込みが許可されるなら、その割り込み処理ルーチンは比較値を更新するのに使えます。

高速PWM動作での比較部はOC2ピンでのPWM波形の生成を許します。COM21,0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COM21,0を'11'に設定することで生成できます(77頁の表44.をご覧ください)。実際のOC2値はポート ピンに対するデータ方向(DDR_OC2)が出力として設定される場合にだけ見えるでしょう。PWM波形はTCNT2とOCR2間の比較一致でOC2(内部)レシ、スタを設定(1)(または解除(0))と、カウンタが解除(\$00、MAXからBOTTOMへ変更)されるタイマ/カウンタ クロック周期でOC2レシ、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。

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

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

OCR2の両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合にあたります。OCR2がBOTTOM(\$00)と等しく設定されると、出力はMAX+1 タイマ/カケンタ クロック周期毎の狭いスパーイク(パールス)になるでしょう。OCR2がMAXに等しく設定されると、(COM21,0 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。

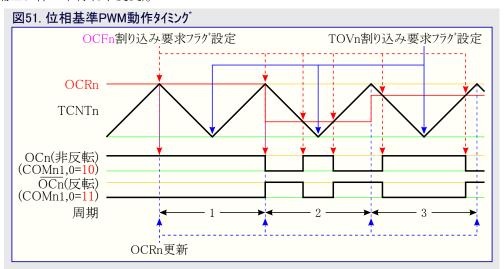
(<mark>訳注</mark>:以下は本動作種別がTOP=MAX(\$FF)で固定のため基本的に不適切です。) 高速PWM動作で(デューティ比50%)周波数の波形 出力は比較一致毎に論理反転するOC2設定(COM21,0=01)によって達成できます。生成された波形はOCR2が0(\$00)に設定される時に $f_{OC2}=f_{Clk_I/O}/2$ の最大周波数でしょう。この特性は高速PWM動作で比較出力部の2重緩衝機能が許可されることを除いて、CTC動作でのOC2交互出力(COM21,0=01)と同じです。



位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM21,0=01)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作に基きます。かシタは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(COM21,0=10)での比較出力(OC2)は上昇計数中のTCNT2とOCR2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(COM21,0=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です。けれども両傾斜(三角波)動作の対称特性のため、これらの動作種別は電動機制御の応用に好まれます。

位相基準PWM動作のPWM分解能は8ビットに固定されます。位相基準PWM動作でのカウンタはカウンタ値がMAXと一致するまで増加されます。カウンタはMAXに達すると計数方向を変更します。このTCNT2値は1タイマ/カウンタ クロック周期間MAXと等しくなります。位相基準PWM動作のタイミング図は図51.で示されます。TCNT2値はタイミング図で両傾斜動作(三角波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。細い赤線はOCR2値を示し、TCNT2値との交点(接点)がTCNT2とOCR2間の比較一致を示します(訳注:図補正に伴い本行若干変更)。



タイマ/カウンタ溢れ(TOV2)フラグはタイマ/カウンタがBOTTOMに到達する時毎に設定(1)されます。この割り込み要求フラグはカウンタがBOTT OM値に到達する毎に割り込みを発生するのに使えます。

位相基準PWM動作での比較部はOC2ピンでのPWM波形の生成を許します。COM21,0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COM21,0ピットを'11'に設定することで生成できます(77頁の表45.をご覧ください)。実際のOC2値はそのポート ピンに対するデータ方向(DDR_OC2)が出力として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT2とOCR2間の比較一致でOC2(内部)レシ、スタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT2とOCR2間の比較一致でOC2レシ、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。

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

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

OCR2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OCR2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51.の第2周期のその出発点に於いて、例え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51.でのようにOCR2はMAXからその値を変更します。OCR2値がMAXのとき、OCnt°ン値は下降計数での比較一致の結果と同じです(<mark>訳補: L→H、</mark>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OCR2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

以下の図は同期動作でのタイマ/カウンタを示し、従ってタイマ/カウンタ クロック(clkT2)が計数許可信号として示されます。非同期動作では clkI/Oがタイマ/カウンタ用発振器(TOSC)クロックによって置換されるべきです。この図は割り込みフラケが設定(1)される時の情報を含みます。図52.は基本的なタイマ/カウンタ動作についてのタイシング・データを示します。この図は位相基準PWM動作以外の全ての動作種別での MAX値近辺の計数の流れを示します。

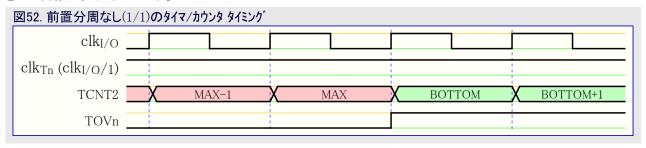


図53.は同じタイミング・データを示しますが、前置分周器が許可されています。

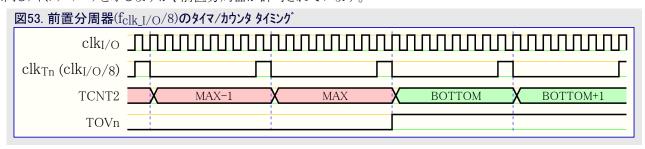


図54.はCTC動作を除く全ての動作種別でのOCF2の設定を示します。

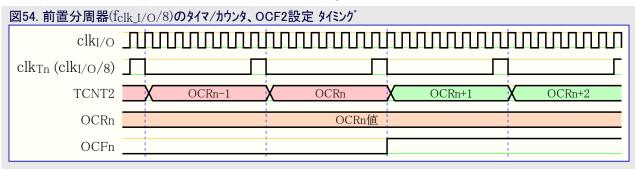
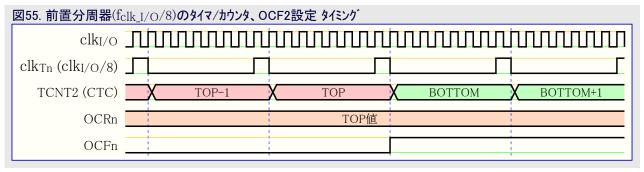


図55.はCTC動作でのTCNT2の解除とOCF2の設定を示します。





8ビット タイマ/カウンタ2 用レシ スタ

TCCR2 - タイマ/カウンタ2制御レシ、スタ (Timer/Counter2 Control Register)

ピット	7	6	5	4	3	2	1	0	_
\$25 (\$45)	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	TCCR2
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 - FOC2: OC2強制変更 (Force Output Compare 2)

FOC2ビットはWGM21,0ビットが非PWM動作を指示する時だけ有効です。けれども将来のデバイスとの共通性を保証するため、PWM動作で扱う時にTCCR2が書かれる場合、このビットはのに設定されなければなりません。FOC2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。OC2出力はCOM21,0ビット設定に従って変更されます。FOC2ビットがストローブとして実行されることに注意してください。従って強制した比較の効果を決めるのはCOM21,0ビットに存在する値です。

FOC2ストローブは何れの割り込みの生成もTOPとしてOCR2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。

FOC2ビットは常に0として読みます。

● ビット3,6 - WGM21,0:波形生成種別 (Waveform Generation Mode bit 1 and 0)

これらのビットはカウンタの計数順序(方向)、最大(TOP)カウンタ値の供給元、使われるべき波形生成のどの形式かを制御します。タイマ/カウンタ部によって支援される動作種別は標準動作、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。表42.と73頁の「動作種別」をご覧ください。

表42.	波形生成	【種別選 】	5			
番号	WGM21 (CTC2)	WGM20 (PWM2)	タイマ/カウンタ動作種別	TOP値	OCR2 更新時	TOV2 設定時
0	0	0	標準動作	\$FF	即値	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	OCR2	即値	MAX
3	1	1	高速PWM動作	\$FF	ВОТТОМ	MAX

注: CTC2とPWM2ビット定義名は旧名です。WGM21,0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 - COM21,0: 比較2出力選択 (Compare Match 2 Output Mode bit 1 and 0)

これらのビットはOC2比較出力ピンの動作を制御します。COM21,0ビットの1つまたは両方が1を書かれると、OC2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、OC2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。

OC2がピンに接続されるとき、COM21,0ピットの機能はWGM21,0ピット設定に依存します。

表43.はWGM21,0ビットが標準動作またはCTC動作(つまりPWM以外)に設定される時のCOM21,0ビット機能を示します。

表44.はWGM21,0ビットが高速PWM動作に設定される時のCOM2 1,0ビットの機能を示します。

表45.はWGM21,0ビットが位相基準PWM動作に設定される時のC OM21,0ビットの機能を示します。

表43.	非PWM動作	での比較出力選択

COM21	COM20	意味
0	0	標準ポート動作 (OC2切断)
0	1	比較一致でOC2ピン トグル(交互)出力
1	0	比較一致でOC2ピン Lowレベル出力
1	1	比較一致でOC2ピン Highレヘブル出力

表44. 高速PWM動作での比較出力選択

COM21	COM20	意味
0	0	標準ポート動作(OC2切断)
0	1	(予約)
1	0	比較一致でLow、BOTTOMでHighを OC2ピンへ出力(非反転動作)
1	1	比較一致でHigh、BOTTOMでLowを OC2ピンへ出力(反転動作)

注: COM21が設定(1)され、OCR2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74頁の「高速PWM動作」をご覧ください。

表45. 位相基準PWM動作での比較出力選択

COM21	COM20	意味
0	0	標準ポート動作 (OC2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OC2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OC2ピンへ出力

注: COM21が設定(1)され、OCR2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75頁の「位相基準PWM動作」をご覧ください。

● ビット2~0 - CS22~0: クロック選択2 (Clock Select2, bit 2,1 and 0)

この3つのクロック選択ビットはタイマ/カウンタ(TCNT2)によって使われるクロック元を選びます。表46.をご覧ください。

表46. タイマ/カウンタ2入力クロック選択

CS22	CS21	CS20	意味
0	0	0	停止(タイマ/カウンタ2動作停止)
0	0	1	clk _{T2S} (前置分周なし)
0	1	0	clk _{T2S} /8 (8分周)
0	1	1	clk _{T2S} /32 (32分周)
1	0	0	clk _{T2S} /64 (64分周)
1	0	1	clk _{T2S} /128 (128分周)
1	1	0	clk _{T2S} /256 (256分周)
1	1	1	clk _{T2S} /1024 (1024分周)

TCNT2 - タイマ/カウンタ2 (Timer/Counter2)

ピット	7	6	5	4	3	2	1	0	_
\$24 (\$44)	(MSB)							(LSB)	TCNT2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

このタイマ/カウンタ レジスタは読み書き両方の操作について、タイマ/カウンタ部の8ビット カウンタに直接アクセスします。TCNT2への書き込みは次のタイマ/カウンタ クロックでの比較一致を妨害(除去)します。カウンタが走行中にカウンタ(TCNT2)を変更することは、TCNT2とOCR2間の比較一致消失の危険を誘発します。

OCR2 - タイマ/カウンタ2 比較レジスタ (Timer/Counter2 Output Compare Register)

ピット	7	6	5	4	3	2	1	0	
\$23 (\$43)	(MSB)							(LSB)	OCR2
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	

この比較レシ、スタは継続的にカウンタ(TCNT2)値と比較される8ビットの値を含みます。一致は比較一致割り込みやOC2ピンでの波形出力を生成するのに使えます。



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

ピット	7	6	5	4	3	2	1	0	_
\$39 (\$59)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	TIMSK
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 - OCIE2 : タイマ/カウンタ2比較割り込み許可 (Timer/Counter2 Output Compare Match Interrupt Enable)

OCIE2ビットが1を書かれ、ステータス レジ、スタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ2比較一致割り込みが許可されます。タイマ/カウンタ2で比較一致が起こる(換言すると、タイマ/カウンタ割り込み要求フラグ・レジ、スタ(TIFR)で比較2割り込み要求フラグ(OCF2)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6 - TOIE2: タイマ/カウンタ2溢れ割り込み許可 (Timer/Counter2 Overflow Interrupt Enable)

TOIE2ビットが1を書かれ、ステータス レジ スタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ2溢れ割り込みが許可されます。タイマ/カウンタ2溢れが起こる(換言すると、タイマ/カウンタ割り込み要求フラグ レジ スタ(TIFR)でタイマ/カウンタ2溢れ割り込み要求(TOV2)フラケが設定(1)される)と、対応する割り込みが実行されます。

TIFR - タイマ/カウンタ割り込み要求フラク レシ スタ (Timer/Counter Interrupt Flag Register)

ピット	7	6	5	4	3	2	1	0	_
\$38 (\$58)	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	TIFR
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 - OCF2 : タイマ/カウンタ2比較割り込み要求フラグ(Timer/Conter2, Output Compare Match Flag)

OCF2ビットはタイマ/カウンタ(TCNT2)と比較レジスタ(OCR2)間で比較一致が起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF2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OCF2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2比較一致割り込み許可(OCIE2)ビット、OCF2が設定(1)されると、タイマ/カウンタ2比較一致割り込みが実行されます。

● ビット6 - TOV2 : タイマ/カウンタ2溢れ割り込み要求フラグ(Timer/Counter2 Overflow Flag)

TOV2ビットはタイマ/カウンク(TCNT2)溢れが起こる時に設定(1)されます。対応する割り込み処理べクタを実行すると、TOV2はハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもTOV2は解除(0)されます。ステータス レジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンク割り込み許可レジスタ(TIMSK)のタイマ/カウンク2溢れ割り込み許可(TOIE2)ビット、TOV2が設定(1)されると、タイマ/カウンク2溢れ割り込みが実行されます。PWM動作ではタイマ/カウンク2が\$00で計数方向を変える時にこのビットが設定(1)されます。



タイマ/カウンタ2の非同期動作

ASSR - タイマ/カウンタ2非同期状態レジスタ (Timer/Counter2 Asynchronous Status Register)

ピット	7	6	5	4	3	2	1	0	
\$22 (\$42)	_	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	ASSR
Read/Write	R	R	R	R	R/W	R	R	R	
初期値	0	0	0	0	0	0	0	0	

● ビット3 - AS2: タイマ/カウンタ2非同期動作許可 (Asynchronous Timer/Counter2)

AS2が0を書かれると、タイマ/カウンタ2はI/Oクロック($clk_{I/O}$)からクロック駆動されます。AS2が1を書かれると、タイマ/カウンタ2はタイマ発振器 (TOSC1,TOSC2)ピンに接続されたクリスタル発振器からクロック駆動されます。AS2の値が変更されると、タイマ/カウンタ2(TCNT2)、比較2レジスタ(OCR2)、タイマ/カウンタ2制御レジスタ(TCCR2)の内容は不正にされるかもしれません。

● ビット2 - TCN2UB: タイマ/カウンタ2更新中フラク (Timer/Counter2 Update Busy)

タイマ/カウンタ2が非同期に動き、タイマ/カウンタ2(TCNT2)が書かれると、このビットが設定(1)になります。TCNT2が一時保存レジスタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理0はTCNT2が新しい値で更新される用意ができたことを示します。

● ビット1 - OCR2UB: 比較2レジスタ更新中フラグ (Output Compare Register2 Update Busy)

タイマ/カウンタ2が非同期に動き、比較2レシ、スタ(OCR2)が書かれると、このビットが設定(1)になります。OCR2が一時保存レシ、スタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理のは比較2レシ、スタ(OCR2)が新しい値で更新される用意ができたことを示します。

● ビット0 - TCR2UB: タイマ/カウンタ2制御レジスタ更新中フラグ(Timer/Counter2 Control Register Update Busy)

タイマ/カウンタ2が非同期に動き、タイマ/カウンタ2制御レシ、スタ(TCCR2)が書かれると、このビットが設定(1)になります。TCCR2が一時保存レシ、スタから更新されてしまうと、このビットはハードウェアによって解除(0)されます。このビットの論理のはタイマ/カウンタ2制御レシ、スタ(TCCR2)が新しい値で更新される用意ができたことを示します。

更新中フラグが設定(1)中に3つのタイマ/カウンタ2 レジスタのどれかに書き込みが実行されると、更新された値は不正にされ、予期せぬ割り込みを起こす原因になるかもしれません。

TCNT2、OCR2、TCCR2読み込みについての機構は異なります。TCNT2を読む時は実際のタイマ/カウンタ値が読まれ、OCR2またはTCCR2を読む時は一時保存レジスタの値が読まれます。



タイマ/カウンタ2 非同期動作

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

- 警告: タイマ/カウンタ2の同期/非同期クロック駆動間を切り替えると、タイマ/カウンタ2(TCNT2)、タイマ/カウンタ2比較レシ、スタ(OCR2)、タイマ/カウンタ2制御レシ、スタ(TCCR2)が不正となるかもしれません。クロック元を切り替える安全な手順を次に示します。
 - 1. タイマ/カウンタ割り込み許可レジスタ(TIMSK)のOCIE2とTOIE2の解除(0)により、タイマ/カウンタ2割り込みを禁止します。
 - 2. 非同期状態レジスタ(ASSR)の非同期動作許可(AS2)設定によってクロック元を適切に選びます。
 - 3. TCNT2, OCR2, TCCR2に新しい値を書きます。
 - 4. 非同期動作へ切り替えるには、TCN2UB, OCR2UB, TCR2UBについて(=0まで)待機します。
 - 5. タイマ/カウンタ割り込み要求フラグ レシ、スタ(TIFR)のOCF2とTOV2フラグを解除(0)します。
 - 6. 必要とされるなら、割り込みを許可します。
- 発振器は時計用32.768kHzクリスタルで使うために最適化されています。TOSC1ピンに外部クロックを印加することは不正なタイマ/カウンタ2動作に終わるかもしれません。CPU主クロック周波数は、この発振器周波数の4倍よりも高くなければなりません。
- TCNT2, OCR2, TCCR2レジスタの1つに書くとき、その値は一時レジスタへ転送され、TOSC1で2つの上昇端後、(実レジスタ)に設定されます。一時レジスタの内容がそれらの転送先へ転送されてしまう前に新しい値を書くべきではありません。記載された3つのレジスタの各々がそれら個別の一時レジスタを持ちます。それは、例えばTCNT2書き込みがOCR2書き込みの実行を妨げないことを意味します。転送先レジスタへの転送が起きたことを検知するために非同期状態レジスタ(ASSR)は実装されました。
- TCNT2, OCR2, TCCR2に書いた後でパワーセーブ動作へ移行するとき、デバイスを起動するのにタイマ/カウンタ2が使われる場合、使用者は書かれたレジスタが更新されてしまうまで待たなければなりません。さもなければMCUは変更が有効になる前に休止形態へ移行するでしょう。タイマ/カウンタ2比較一致割り込みがデバイスを起動するのに使われる場合、TCNT2またはOCR2書き込み中の比較機能が禁止されるため、これは非常に重要です。書き込み周期が完了されず、OCR2UBがOに戻る前にMCUが休止形態へ移行すると、デバイスは決して比較一致割り込みを受け取らず、そしてMCUは起動しないでしょう。
- パワーセーブ動作からデバイスを起動するのにタイマ/カウンタ2が使われる場合、使用者がこれら動作へ(復帰後)再移行することを欲するなら、予防処置を講じなければなりません。割り込み論理回路はリセットするために1 TOSC1周期必要です。起動と休止形態再移行間の時間が1 TOSC周期未満の場合に割り込みが起きず、デバイスは起動に失敗するでしょう。 パワーセーブ動作再移行前の時間が充分であるかどうか迷うなら、1 TOSC1周期が経過されることを保証するために次の手順が使えます。
 - 1. TCNT2, OCR2, TCCR2に値を書きます。
 - 2. 非同期状態レジスタ(ASSR)の対応する更新中フラグがOに戻るまで待ちます。
 - 3. パワーセーブ動作へ移行します。
- 非同期動作が選ばれると、タイマ/カウンタ2用32.768kHz発振器はパワーダウン動作とスタンバイ動作を除いて常に動作します。電源投入リセット、パワーダウンまたはスタンバイ動作から起動後、この発振器が安定するのに1秒程度かかるかもしれないことを使用者は承知すべきです。電源投入、パワーダウンまたはスタンバイ動作から起動後、タイマ/カウンタ2を使う前に少なくとも1秒待機することが推奨されます。この発振器の使用またはクロック信号がTOSCピンに印加されるかのどちらかに拘らず、起動時の不安的なクロック信号のため、パワーダウンまたはスタンバイ動作からの起動復帰後、タイマ/カウンタ2の全レジスタの内容が失われたとみなされなければなりません。
- タイマ/カウンタ2が非同期でクロック駆動される時のパワーセーブ動作からの起動の説明。割り込み条件が合致すると、タイマ/カウンタ クロック の次の周期で起動処理が開始され、プロセッサがカウンタ値を読めるのに先立ってタイマ/カウンタは常に最低1、進行されます。起動後 MCUは4周期停止され、割り込み処理ルーチンを実行し、そしてSLEEP命令の次の命令から実行を再開します。
- パワーセーブ動作から起動直後のTCNT2の読み込みは不正な結果を得るかもしれません。TCNT2が非同期TOSCクロックでクロック駆動されるため、TCNT2読み込みは内部I/Oクロック領域に同期したレジスタを通して行われなければなりません。同期化はTOSCクロックの全上昇端で行われます。パワーセーブ動作から起動し、I/Oクロック(clk_{I/O})が再び活性(有効)になるとき、TCNT2はTOSCクロックの次の上昇端まで以前(休止形態移行前)の値を読むでしょう。パワーセーブ動作から起動後のTOSCクロックの位相は起動時間に依存するため本質的に特定できません。従ってTCNT2読み込みに対する推奨手順は次のとおりです。
 - 1. OCR2またはTCCR2のどちらかに何か値を書きます。
 - 2. 非同期状態レジスタ(ASSR)の対応する更新中フラグが解除(0)されるまで待ちます。
 - **3**. TCNT2を読みます。
- 非同期動作の間中、非同期タイマ用割り込み要求フラグの同期化は3プロセッサ周期+1タイマ周期かかります。従ってプロセッサが割り込み要求フラグ設定の原因となったタイマ値を読めるのに先立って、このタイマは最低1、進行されます。比較出力ピンはタイマクロックで変更され、プロセッサクロックに同期されません。

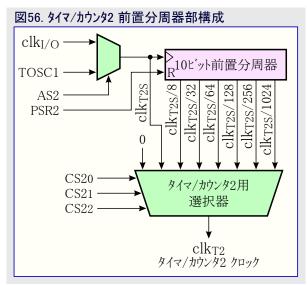


タイマ/カウンタ2の前置分周器

タイマ/カウンタ2用クロック元の名前はclkT2Sです。既定でのclkT2Sは主システム I/O クロック(clkI/O)に接続されます。タイマ/カウンタ2非同期状態レジ、スタ(ASSR)の非同期クロック(AS2)と、ットの設定(1)により、タイマ/カウンタ2はTOSC1 と、ンから非同期にクロック駆動されます。これは実時間計数器(RTC)としてのタイマ/カウンタ2 使用を可能にします。AS2が設定(1)されると、TOSC1とTOSC2と、ンはポートBから切り離されます。クリスタル発振子はタイマ/カウンタ2用の独立したクロック元として扱うため、TOSC1とTOSC2と、ン間に接続できます。この発振器は32.768kHzクリスタル発振子で使うために最適化されています。TOSC1に外部クロック元を印加することは推奨されません。

gイマ/カウンg2に対して可能な前置分周済み選択はclk $_{T2S}/8$, clk $_{T2S}/32$, clk $_{T2S}/64$, clk $_{T2S}/128$, clk $_{T2S}/256$, clk $_{T2S}/1024$ です。加えて0(停止)は 勿論clk $_{T2S}$ も選択可能です。

特殊I/O機能レジ、スタ(SFIOR)のタイマ/カウンタ2 前置分周器リセット(PSR2)ビットの設定(1)は前置分周器をリセットします。これは予測可能な前置分周器での操作を使用者に許します。



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

ピット	7	6	5	4	3	2	1	0	
\$30 (\$50)	_	-	-	-	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

ビット1 - PSR2:タイマ/カウンタ2 前置分周器リセット (Prescaler Reset Timer/Counter2)

このビットが1を書かれると、タイマ/カウンタ2の前置分周器はリセットします。動作が実行された後、このビットはハードウェアによって解除(0)されます。このビットへの0書き込みは無効です。タイマ/カウンタ2が内部CPUクロックによってクロック駆動される場合、このビットは常に0として読みます。タイマ/カウンタ2が非同期動作で動いている時にこのビットが書かれると、このビットは前置分周器がリセットされてしまうまで1に留まります。



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ATmega8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。ATmega8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間相互連結は図58.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ハルスを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。各データハプル外後、主装置はSS(従

図57. SPI構成図 (MISO) Bus clk_{I/O} M MSB (SPDR) LSB M Data] (MOSI) 分周器 8ビット移動レジスタ 1/32 8-bit 1/2 移 動クロック 制 受信データ緩衝部 御 S路 (M) SCK) 選択 クロック回路 SPI 割り込み SS 要求 SPI制御回路 SPI2X CPOL DOR MST] SPI状態レシ SPI制御レシブスタ (SPCR)

す。各データ パケット後、主装置はSS(従 注: SPIピン配置については2頁の「ピン配置」と36頁の表22.を参照してください。

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。

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)から読まれなければなりません。さもなければ始めのバイトは失われます。

SPI従装置動作では制御論理回路がSCKピンの到着信号を採取します。このクロック信号の正しい採取を保証するため、LowとHighの最小周期は以下であるべきです。

- Low周期 : 2 CPUクロック周期より長い - High周期: 2 CPUクロック周期より長い

SPIが許可されると、MOSI, MISO, SCK, SSピンの データ方向は**表47**.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、35頁の 「<mark>交換ポート機能</mark>」を参照してください。

図58. SPI 主装置/従装置の連結 MSB 主装置 従装置 LSB MSB LSB MISO MISO 8ピット移動レジスタ 8ビット移動レジスタ MOSI MOSI. SCK SCK SS SS 移動許可 クロック回路

ピン方同規定	
主装置時の方向規定	従装置時の方向規定
ポートB方向レジスタ(DDRB)の指定	入力
入力	ポートB方向レジスタ(DDRB)の指定
ポートB方向レジスタ(DDRB)の指定	入力
ポートB方向レジスタ(DDRB)の指定	入力
	主装置時の方向規定 ポートB方向レシ、スタ(DDRB)の指定 入力 ポートB方向レシ、スタ(DDRB)の指定

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36頁の「ポートB の交換機能」を参照してください。



次のコート例は主装置として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SPIt゚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
                                                       ;MOSI, SCK=出力、他は入力値を取得
SPI_M_Init: LDI
                   R17, (1<<DD_MOSI) | (1<<DD_SCK)
            OUT
                   DDR_SPI, R17
                                                       ;MOSI, SCK=出力、他は入力に設定
                   R17, (1<<SPE) | (1<<MSTR) | (1<<SPR0)
                                                       ;SPI許可、主装置、16分周値を取得
            LDI
            OUT
                   SPCR, R17
                                                       ;SPI許可、主装置、16分周に設定
            RET
                                                       ;呼び出し元へ復帰
                                                       ; データ(R16) 送信開始
SPI_M_Tx:
           OUT
                  SPDR, R16
                                                       ;転送完了ならばスキップ
SPI_M_Tx_W: SBIS
                  SPSR, SPIF
                                                      ;転送完了まで待機
            R.JMP
                  SPI_M_Tx_W
            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) \mid (1 << MSTR) \mid (1 << SPR0);
                                                      /* SPI許可、主装置、16分周に設定 */
}
void SPI_MasterTransmit(char cData)
                                                      /* データ送信開始 */
    SPDR = cData;
    while(!(SPSR & (1<<SPIF)));
                                                      /* 転送完了まで待機 */
```

注: 5頁の「コート・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
                                                   ;MISO出力、他は入力値を取得
                 R17, (1<<DD_MISO)
           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
                                                   ;呼び出し元へ復帰
           RET
SPI_S_Rx:
                 SPSR, SPIF
                                                   ;受信(転送)完了ならばスキップ
           SBIS
                                                   ;受信(転送)完了まで待機
           R.JMP
                 SPI_S_Rx
                                                   ;受信データを取得
           IN
                 R16, SPDR
           RET
                                                   ;呼び出し元へ復帰
C言語プログラム例
void SPI_SlaveInit(void)
   DDR SPI = (1 << DD MISO);
                                                   /* MISO出力、他は入力に設定 */
   SPCR = (1 << SPE);
                                                   /* SPI許可設定 */
}
char SPI_SlaveReceive(void)
   while(!(SPSR & (1<<SPIF)));
                                                   /* 受信(転送)完了まで待機 */
   return SPDR;
                                                   /* 受信データと共に復帰 */
注: 5頁の「コード例について」をご覧ください。
```



SSL°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んでデータ送信を始めると解釈します。バスの衝突を避けるためにSPIシステムは次の動作を行います。

- 1. SPCRで主/従選択(MSTR)ピットが解除(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、且つステータスレシ、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。

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

ピット	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びます。 \overline{SS} が入力として設定され、 \overline{MSTR} が設定(1)の間に Lowへ駆動されると、 \overline{MSTR} が解除(0)され、 \overline{SPI} 状態レジスタ(SPSR)で \overline{SPI} 割り込み要求フラケ(SPIF)が設定(1)になります。その後に使用者は \overline{SPI} 主装置動作を再び許可するために \overline{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、7イドル時にSCKはHighです。CPOLが0を書かれると、7イドル時にSCKはLowです。例については図59.と図60.を参照してください。CPOL機能は右で要約されます。

表48. CI	表48. 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59.と図60.を参照してください。CPH A機能は右で要約されます。

表49. CF	表49. CPHA機能動作									
CPHA	SCK先行端	SCK後行端								
0	入力採取	出力設定								
1	出力設定	入力採取								

● ビット1.0 - SPR1.0: SPIクロック選択 (SPI Clock Rate Select 1 and 0)

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

表50. SCK速度選択(fosc=CPUクロック周波数)

SPR1	0)			1		
SPR0	()		1	()	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	

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

ピット	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フラケが設定(1)されます。全割り込みが許可(ステータスレジスタ(SREG)の全割り込み許可(I)ビット=1)されて、SPI制御レジスタ(SPCR)でSPI割り込み許可(SPIE)ビットが設定(1)されるなら、割り込みが生成されます。SPIが主装置動作の時にSSEピンが入力でLowに駆動されるなら、これもこのSPIFフラケを同様に設定(1)します。対応する割り込み処理へクタを実行する時にSPIFはハート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(SPDR)をアクセスすることによってもSPIFフラケは解除(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データレンブスタ(SPDR)が書かれると、このWCOLビットが設定(1)されます。WCOLビット(とSPIFビット)はWCOLが設定(1)されたSPI状態レシブスタ(SPSR)を始めに読み、その後にSPIデータレシブスタ(SPDR)をアクセスすることによって解除(0)されます。

● ビット5~1 - Res: 予約 (Reserved Bit)

これらのビットは予約されており、常に0として読みます。

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

このビットが論理1を書かれると、SPIが主装置動作の時にSCK速度(SCK周波数)が倍にされます(表50.参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定される時に、SPIはfosc(CPUクロック周波数)/4またはそれ以下での動作のみ保証されます。

ATmega8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154頁をご覧ください。

SPDR - SPIデータ レジスタ (SPI Data Register)

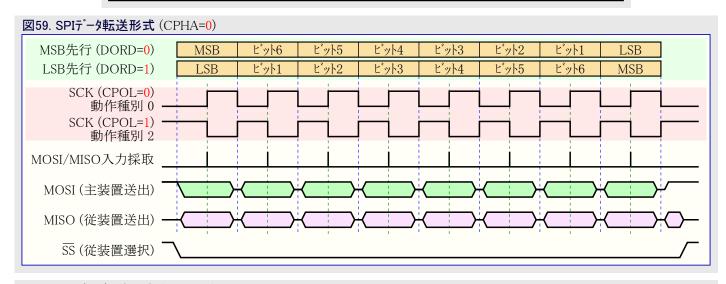
ピット	7	6	5	4	3	2	1	0	_
\$0F (\$2F)	(MSB)							(LSB)	SPDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	不定	不定	不定	不定	不定	不定	不定	不定	

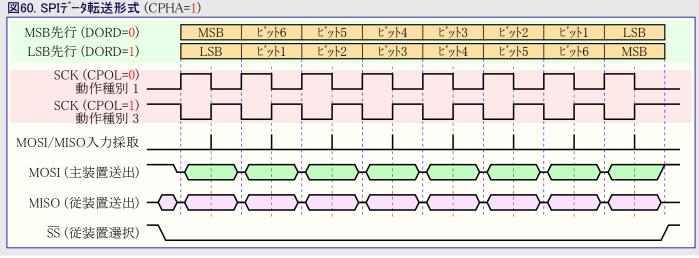
SPIデータレジスタはSPI移動レジスタとレジスタファイル(汎用レジスタ)間のデータ転送に使われる読み書き可能なレジスタです。このレジスタへの書き込みがデータ送信を開始します。このレジスタの読み込みは移動レジスタの受信緩衝部読み出しを引き起こします。

データ転送形式

直列データに関してはSPI制御レシ、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59.と図60.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充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48.と表49.を要約することによって明解にされます。

表51. CPOL,CPHA機能動作 SPI動作種別番号 CPOL CPHA SCK先行端 SCK後行端 0 入力採取/上昇端 出力設定/下降端 0 1 0 出力設定/上昇端 入力採取/下降端 2 0 入力採取/下降端 出力設定/上昇端 3 出力設定/下降端 入力採取/上昇端





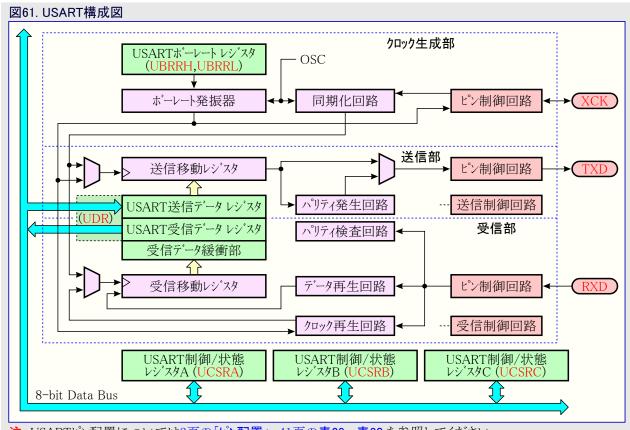
USART

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
- ハート・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除去器
- 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

概要

USARTの簡単化した構成図は図61.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。



注: USARTピン配置については2頁の「ピン配置」、41頁の表29.、表30.を参照してください。

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



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事実です。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。
- 受信部移動レシ、スタは第3緩衝段のように動けます。これは緩衝レシ、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シ、スタ(図61.参照)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 オーハーラン(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。

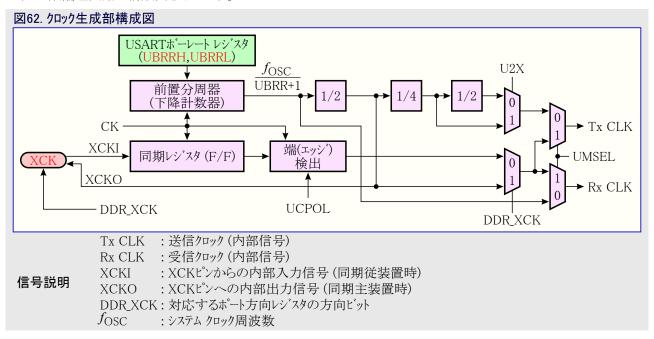
- 9ビット選択(CHR9)ビットはデータ長選択ビット2(UCSZ2)へ変更
- オーハーラン発生(OR)フラク はオーハーラン発生(DOR)フラク へ変更



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C(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62.はクロック生成論理回路の構成図を示します。



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62.を参照してください。

USARTボーレート レジスタ(UBRR(UBRRH:UBRRL))と下降計数器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(fosc)で走行する下降計数器は0への下降計数時毎または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)は計数器が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(=fosc/(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52.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52. 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レシブスタ値(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60.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62.を参照してください。

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

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

fosc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)が変更される端と反対の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/状態レシ、スタC(UCSRC)のXCK極性(UCPO L)ピ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63.で示されるようにUCPOLがのとき、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。



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 tinh Find
- 奇数または偶数パリティビット、またはなし
- 1 または 2 停止ビット

フレームは最下位データ ビット(LSB)が次に続く開始ビットで始まります。その後に次データ ビットが最後の最上位データ ビット(MSB)まで(最大)合計9ビット続きます。許可したなら、パリティ ビットがデータ ビットの後、停止ビットの前に挿入されます。完全なフレームが送信されると、新規フレームによって直ちに後続されるか、または通信線をアイドル状態(high)に設定できます。図64.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。



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、レシブ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初期化コート・例が示すアセンフ・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ポーリングを使う (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レシブ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 レジブスタ上位(UBRRH)と UCSRCによるI/O位置の共有のため、レシ、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)
                   UBRRL, R16
                                                        ;ボーレート設定(下位バイト)
            OUT
                   R16, (1<<URSEL) | (1<<USBS) | (3<<UCSZ0)
            LDI
                                                        ;フレーム形式値を取得
                   UCSRC, R16
                                                        ;フレーム形式設定(8ビット,2停止ビット)
            OUT
                   R16, (1<<RXEN) | (1<<TXEN)
            LDT
                                                        ;送受信許可値を取得
            OUT
                   UCSRB, R16
                                                        ;送受信許可
                                                        ;呼び出し元へ復帰
            RET
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值 */
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) */
    UBRRL = (unsigned char) baud;
                                                        /* ボーレート設定(下位バイト) */
    UCSRC = (1 \le URSEL) | (1 \le USBS) | (3 \le UCSZO) :
                                                        /* 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: 5頁の「コート・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。



USARTのデータ送信

USART送信部はUSART制御/状態レシ、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デル状態(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ビ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。

次のコート・例はUSART制御/状態レシ、スタA(UCSRA)の送信データレシ、スタ空き(UDRE)フラケ、のホーリンケ。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レシ、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: 5頁の「コート・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内に書きます。

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)、データの下位バイトが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
                                                      ;送信緩衝部空き待機
            RJMP
                   USART_Tx
                  UCSRB, TXB8
                                                      ;第9ビットを0に仮設定
            CBI
            SBRC
                   R17, 0
                                                      ;送信すべき第9ビットが<mark>0</mark>でスキップ
                   UCSRB, TXB8
            SBI
                                                       ; 第9ビットを1に設定
                                                       ;データ送信(送信開始)
            OUT
                   UDR, R16
                                                       ;呼び出し元へ復帰
            RET
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 &= ^{\sim}(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/状態レシ、スタB(UCSRB)でデータレシ、スタ空き割り込み許可(UDRIE)ヒットが1を書かれると、(全割り込みが許可されているなら)UDREフラケが設定(1)されている限り、USARTデータレシ、スタ空き割り込みが実行されます。UDREはUSARTデータレシ、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ず、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シ、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も解除(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(ハーフデュープレックス)通信インターフェースで有用です。

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USAR T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。

USARTのデータ受信

USART受信部はUSART制御/状態レシ、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ビ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ビ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り無視されます。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。

次のコート・例はUSART制御/状態レシ、スタA(UCSRA)の受信完了(RXC)フラケのポーリンケを基にした簡単な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
                 USART Rx
           R TMP
                                                  ;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: 5頁の「コート'例について」をご覧ください。
```

注: 5貝のバートがについて」をご見くたさい。

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)から下位ハイトを読む前に、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)ビットから読まれなければなりません。この規則はフレーミング異常(FE)、オーハーラン発生(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。

次のコート 例は9tiット データと状態ti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:
                 UCSRA, RXC
                                                       ;受信完了でスキップ
           SBIS
                                                       ;受信完了待機
           R.JMP
                 USART_Rx
           IN
                  R18, UCSRA
                                                       ;状態フラグ取得
                  R17, UCSRB
           IN
                                                       ;受信第9ビット取得
                  R16, UDR
                                                       ;受信データ取得
           IN
                  R18, (1<<FE) | (1<<DOR) | (1<<PE)
                                                       ;受信異常検査
           ANDI
           BREQ
                  USART Rx V
                                                       ;異常なしで分岐
                  R17, -1
                                                       ;異常で-1値設定
           LDI
                  R16, -1
           LDI
                  R17
                                                       ;RXB8ビットをビット0位置へ移動
USART_Rx_V: LSR
           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, resh, 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 */
   resh = UCSRB;
                                                       /* 受信第9ビット取得 */
   res1 = UDR;
                                                       /* 受信データ取得 */
   if (status & ((1<<FE) | (1<<PE)) ) return -1; /* 受信異常で-1値設定/復帰*/
   resh = (resh >> 1) \& 0x01;
                                                       /* RXB8ビットのみ有効最下位へ */
                                                       /* 結果9ビット データ取得/復帰 */
   return ((resh<<8) | resl);
```

注: 5頁の「コート・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レシ、スタをレシ、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できるため、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/状態レシ、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フラケが設定(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フラケを解除(0)するためにUSARTデータレシ、スタ(UDR)から受信したデータを読まなければならず、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、フレーシング異常(FE)、データ オーハーラン発生(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 オーハーラン発生(DOR)フラグは受信部緩衝部が一杯状態のためのデータ消失を示します。データ オーハ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)フラク・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フラグは常にのが読めます。将来のデバイスとの共通性のため、UCSRAに書くとき、常にこのビットをの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ビットによって選ばれます。許可されると、パリティ検査器は到着フレーム内のデータ ビットのパリティを計算し、その結果と(受信)直列フレーム内のパリティ ビットを比較します。検査の結果は受信データ、停止ビットと共に受信緩衝部に格納されます。その後パリティ誤り(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フラク・は受信緩衝部から読まれ得る次フレームで受信時にハッリティ異常があり、その時点でハッ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データレシ、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O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。次のコート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
                                                   ;未読データありでスキップ。
USART Flush: SBIS
                  UCSRA, RXC
            RET
                                                   ;未読データなしで復帰
                                                   ;データ受信
            IN
                  R16, UDR
            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)
                                                   /*一時変数定義 */
   unsigned char dummy;
   while ( UCSRA & (1<<RXC) ) dummy=UDR;
                                                   /* 未読データ読み捨て */
注: 5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65.は到着フレームの開始ビット採取手順を図解します。採取速度は標準速動作でボーレートの16倍、倍速動作で8倍です。赤帯(<mark>訳注</mark>: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)の時に行われる採取です。



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めるために、標準速動作に対して採取8,9,10、倍速動作に対して採取4,5,6(図の赤数字(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められます。データ再生部は標準速動作で16段、倍速動作で8段の順列回路を使います。図66.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。



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ことによって行われます。この中央の3採取は図上の赤文字(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($p-n^2$ 、p)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67.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。



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。標準速動作での最初のLowレベル採取は図67.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53.参照)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。

 $R_{\text{slow}} = \frac{(D+1)\times S}{S-1+D\times S+S_F}$ $R_{\text{fast}} = \frac{(D+2)\times S}{(D+1)\times S+S_M}$

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53.は許容できる最大受信部ボーレート誤差一覧です。標準速動作には、より高いボーレート変動許容力があることに注目してください。

表53. 標準速と倍速での受信部ボーレート推奨最大許容誤差

 表66. 你不是CITE CO文品的样子!TEXXXIII I WE												
D		標	準速動作 (U2X= <mark>0</mark>)		倍速動作 (U2X=1)							
ט	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)	Rslow(%)	Rfast(%)	総合許容誤差(%)	推奨許容誤差(%)				
5	93.20	106.67	-6.80~+6.67	±3.0	94.12	105.66	-5.88~+5.66	±2.5				
6	94.12	105.79	-5.88~+5.79	±2.5	94.92	104.92	-5.08~+4.92	±2.0				
7	94.81	105.11	-5.19~+5.11	±2.0	95.52	104.35	-4.48~+4.35	±1.5				
8	95.36	104.58	-4.54~+4.58	±2.0	96.00	103.90	-4.00~+3.90	±1.5				
9	95.81	104.14	-4.19~+4.14	±1.5	96.39	103.53	-3.61~+3.53	±1.5				
10	96.17	103.78	-3.83~+3.78	±1.5	96.70	103.23	-3.30~+3.23	±1.0				

注: Dはデータビット数とパリティビットの合計ビット数です。

(訳注) 原書は表53.に標準速、表54.に倍速を記載していますが、比較が容易なように表53.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元で作られました。

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がアトレス指定されるかを検出するため、最初にアト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がアト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は他のアト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、選ばれたかを判定します。選ばれた場合は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. アト・レス指定されたMCUは新規アト・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(フルデュープレックス)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(リード モデファイ 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レシ 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。

次のコート・例はこの2つのレシ、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
                                                            ;UBRRHに2を設定
               OUT
                      UBRRH, R16
               LDI
                      R16, (1<<URSEL) | (1<<USBS) | (1<<UCSZ1)
                                                            ;UCSRCのUSBS,UCSZ1のみ1設定
               OUT
                                                            ;その他ビット=0
                      UCSRC, R16
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: 5頁の「**コート 例について**」をご覧ください。

このコート'例が示すように2つのレシ'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シ、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シ、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
                                                       ;初回読み込み(擬似)
                    R16, UBRRH
              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内容取得 */
   ucsrc = UCSRC;
                                                       /* 呼び出し元へ復帰 */
   return ucsrc;
```

注: 5頁の「**コート** 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

UDR - USARTデータレシ スタ (USART I/O Data Register)

ピット	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								
初期値	0	0	0	0	0	0	0	0	

同じI/Oアトレスを共用するUSART受信データ緩衝レシ、スタとUSART送信データ緩衝レシ、スタはUSARTデータレジ、スタまたはUDRとして引用しました。送信データ緩衝レシ、スタ(TXB)はUDRレシ、スタ位置に書かれるデータの転送先です。UDRレシ、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(リート・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。

UCSRA - USART制御/状態レジスタA (USART Control and Status Register A)

ピット	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る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)と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-タ オーバーラン発生フラク (Data OverRun)

このビットはオーハ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 オーハ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。



UCSRB - USART制御/状態レジスタB (USART Control and Status Register B)

ピット	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)、オーバーラン(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/状態レシ、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。

UCSRC - USART制御/状態レジスタC (USART Control and Status Register C)

ビット	7	6	5	4	3	2	1	0	
\$20 (\$40) UR	RSEL U	MSEL L	PM1 (JPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write R	/W 1	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レシ、スタはUBRRHレシ、スタと同じI/O位置を共用します。このレシ、スタのアクセス法を記述する項目、100頁の「UBRRH/UCSRCレシ、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びます。UCSRCを読むと1として読みます。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55. USART動	表55. 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、UPMo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56. 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57. 停止ビット選択										
停止ビット数										
1ピット										
2ピット										

● 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58. データ ビット長選択

UCSZ2∼0	0 0 0	0 0 1	0 1 0	0 1 1	100	101	1 1 0	111
データビット数	5ピット	6ビット	7ピット	8t [*] yh	(予約)	(予約)	(予約)	9Ľ'yŀ

● 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時にこのピットに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59. XCI	表59. 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												

UBRRH, UBRRL (UBRR) – USARTボーレートレジスタ (USART Baud Rate Register)

ピット	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	
ピット	7	6	5	4	3	2	1	0	
\$09 (\$29)	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0	UBRRL
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びます。UBRRHを読むとOとして読みます。UBRRHに書くとき、このビットはOでなければなりません。

ヒット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<mark>0</mark>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、UBRRL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。UBRRL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60**.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(<mark>訳注</mark>: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。

誤差率(%) = (UBRR設定ボーレート(最近似値) - 1) ×100(%)

表60. Xtal、ホーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ボーレート		1M	lHz			1.843	2MHz			2M	lHz			2.457	6MHz	
(bps)	U2	X= <mark>0</mark>	U2	2X=1	U2	!X= <mark>0</mark>	U2	!X=1	U2	X= <mark>0</mark>	U2	!X=1	U2	2X= <mark>0</mark>	U2	2X=1
(phs)	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	62	500	1:	25k	11	5.2k	23	0.4k	1:	25k	2	50k	15	3.6k	30	7.2k

ホ゛ーレート		3.276	8MHz			3.686	4MHz			4M	Hz			4.608	3MHz	
/- \	U2	!X= <mark>0</mark>	U2	X=1	U2	X= <mark>0</mark>	U2	X=1	U2	X= <mark>0</mark>	U2	!X=1	U2	!X= <mark>0</mark>	U2	X=1
(bps)	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	-	_	1	-	_	1	_	-	-	_	_	_	_	_	0	-42.4
最高速	20	4.8k	40	9.6k	23	0.4k	46	0.8k	2	50k	50	00k	28	88k	5	76k

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。

(<mark>訳注</mark>) 原書では頁割された表毎に表番号60.~63.となっていますが、共通性から纏めて表60.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60 (続き). Xtal、ボーレート対UBRRH,UBRRL設定 (UBRR=UBRRH:UBRRL)

ボーレート		4.915	2MHz			6.144	IMHz			7.372	8MHz			8M	lHz	
	U2	X= <mark>0</mark>	U2	X=1	U2	!X= <mark>0</mark>	U2	X=1	U2	!X= <mark>0</mark>	U2	X=1	U2	X= <mark>0</mark>	U2	X=1
(bps)	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	30	7.2k	61	4.4k	38	84k	76	58k	46	0.8k	92	1.6k	500k 1M			.M

ホーレート		9.216	MHz			9.830	4MHz			10N	ИНz			11.059	2MHz	
(bps)	U2	X= <mark>0</mark>	U2	!X=1	U2	!X= <mark>0</mark>	U2	X=1	U2	!X= <mark>0</mark>	U2	!X=1	U2	!X= <mark>0</mark>	U2	2X=1
(phs)	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	5′	76k	1.1	.52M	61	4.4k	1.22	288M	625k		1.25M		69	1.2k	1.3824M	

ホーレート	14.7456MHz				16MHz				18.4320MHz				20MHz			
(bps)	U2X=0		U2X=1		U2X= <mark>0</mark>		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。



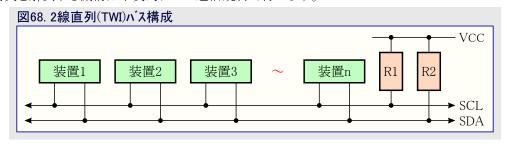
2線直列インターフェース (TWI:Two-wire Serial Interface, I²C)

特徴

- 2本のバス信号線のみ必要な、単純ながら強力で柔軟な通信インターフェース
- 主装置動作と従装置動作の両方を支援
- 送信装置または受信装置として動作可能
- 7ビットのアドレス空間が128までの異なる従装置アドレスを許容
- 複数主装置の調停支援
- 400kHzまでのデータ転送速度
- 上昇/下降(スリューレート)制限された出力駆動回路
- バス信号線のスパイクを排除する雑音消去回路
- 一斉呼び出しを含む完全に設定変更可能な従装置アドレスの支援
- AVRが休止形態の時のアドレス認証(一致)起動

2線直列インターフェース バスの定義

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



TWI用語定義

次の定義は本章で度々使われます。

表64. TWI用語定義					
用語	意味				
主装置	送信の開始と終了を行う装置。主装置はSCLクロックも生成します。				
従装置	主装置によって指定された装置。				
送信装置	バス上にデータを送り出す装置。				
受信装置	バスからデータを読み込む装置。				

電気的な相互接続

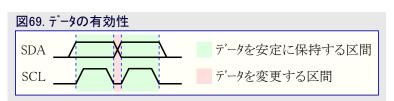
図68.で描かれたように両方のバス信号線はプルアップ抵抗を通して正供給電圧に接続されます。全てのTWI準拠装置のバス駆動部はオープントレインかオープンコレクタです。これはインターフェースの動作のために重要なワイアート、AND機能を実現します。TWIバス信号線のLowレベルは1つまたはより多くのTWI装置のO出力時に生成されます。Highレベルは全TWI装置がHi-Z出力時の出力で、プルアップ抵抗に信号線をHighへ引き上げさせます。どんなバス動作を許すのにも、TWIバスに接続した全てのAVRデ゙バイスが電力供給されなければならないことに注意してください。

このバスに接続できる装置数はアビットの従装置アドレス空間と400pFのバス容量制限によってのみ制限されます。TWIの電気的特性の詳細仕様は158頁の「**2線直列インターフェース特性**」で与えられます。そこで与えられる2組の異なる仕様は、1つがバス速度100kHz以下に関するもので、もう1つはバス速度400kHzまでに関して有効です。

データ転送とフレーム形式

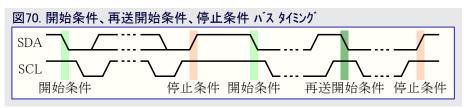
ビット転送

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



開始条件と停止条件

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



アトレスパケット形式

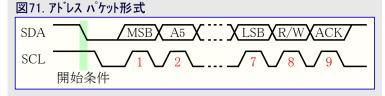
TWIハスに送信した全てのアドレス パケットはアビットのアドレス ビット、1ビットの方向(Read/Write)制御ビット、1ビットの応答ビットから成る9ビットです。方向(R/W)ビットが設定(1)されると読み出し操作が実行され、さもなければ書き込み操作が実行されるべきです。従装置がアドレス指定されたことを認証すると、9番目のSCL(ACK)周期でSDAをLowへ引くことによって確認応答すべきです。アドレス指定された従装置が忙しいまたはその他の理由で主装置の要求を扱えない場合、確認応答(ACK)クロック周期でSDA信号線をHighのままにすべきです。主装置はその後に停止条件または新規転送を始め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

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

一斉呼び出しが起こされると、全従装置は確認応答(ACK)周期でSDA信号線をLowにすることによって応答すべきです。一斉呼び出しは主装置がシステム内のそれぞれの従装置に同じ通信内容を送信したい時に使われます。一斉呼び出しアドレスに続きW(方向が書き込み)ビットがハ、スに送信されると、一斉呼び出しに応答する設定の全ての従装置は、ACK周期でSDA信号線をLowに引き込みま

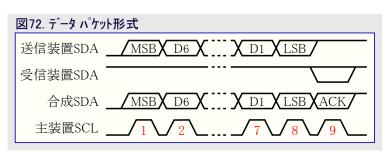
す。そして後続のデータ パケットは一斉呼び出しに確認応答した全従装置によって受信されます。一斉呼び出しアドレスに続くR(方向が読み出し)ビットの送信は、従装置それぞれが異なるデータの送信を始めた場合の衝突の原因となるので意味がないことに注意してください。

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



データ パケット形式

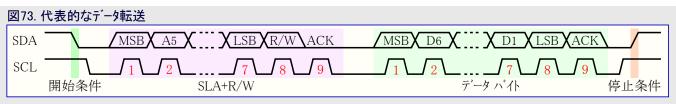
TWIN、スに送信した全てのデータハットットは1ハ、仆のデータと1ビットの応答ビットから成る9ビットです。データ転送中、主装置はクロックと開始条件、停止条件を生成し、一方受信装置は受信に応答する責任があります。確認応答(ACK)は受信装置が9番目のSCL周期中にSDA信号線をLowに引き込むことによって示されます。受信装置がSDA信号線をHighのままにするとNACKを示します。受信装置が最終ハ・仆を受信したとき、または何らかの理由でこれ以上のハ・イトを受信ができないとき、最終ハ・仆後にNACKを送ることによって送信装置へ通知すべきです。データハ・イトの最上位(MSB)ビットが最初に送信されま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



転送内でのアドレス パケットとデータ パケットの組み合わせ

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

図73.は代表的なデータ転送を示します。応用ソフトウェアによって実装されたソフトウェア規約に依存して、様々なデータがSLA+R/Wと停止条件間に送信できることに注意してください。



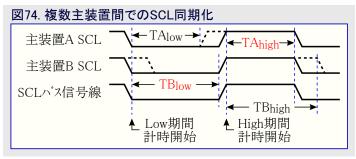
複数主装置バスシステムの調停と同期

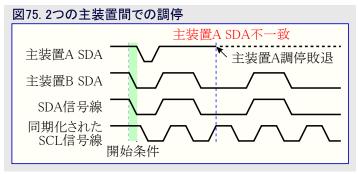
TWI規約は多数主装置のバス システムを許します。例え2つ以上の主装置が同時に送信を始めても、送信が通常のように続行することを保証するために特別な手段が講じられます。複数主装置のシステムでは2つの問題が起こります。

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

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

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





調停が次の状態間で許されないことに注意してください。

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

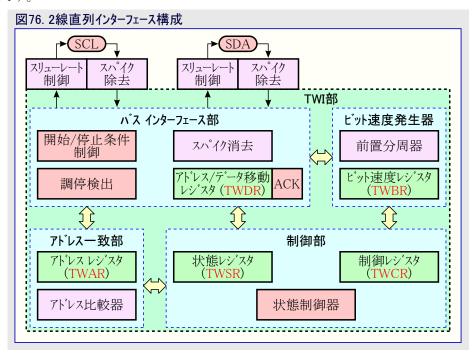
これらの違法な調停状態を決して起こさないように保証するのは使用者ソフトウェアの責任です。これは複数主装置システムでの全てのデータ転送は同じ構成、SLA+R/Wとデータ パケットを使わなければならないことを意味します。言葉を変えると、全ての送信は同じデータ パケット数を含まなければならず、さもなければ調停の結果は不定にされます。

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



TWI部の概要

図76.で示されるようにTWI部は様々な部分から成ります。赤文字で示された(訳注:原文は太線で描かれた)全てのレジスタはAVRデータ バスを通してアクセス可能です。



SCLとSDAピン

これらのピンはAVR TWIをMCUシステムのその他とインターフェースします。出力駆動部はTWI仕様に適合させるためのスリューレート(上昇/下降)制限器を含みます。入力段は50nsよりも短いスパーイクを除去するスパーイク消去部を含みます。「入出力ポート」章で説明したようにAVR パット・の内部プルアップはSCLとSDAピンに対応するポートのビットを設定(=1)することによって許可できることに注目してください。内部プルアップはいくつかのシステムで外部抵抗の必要をなくせます。

ビット速度発生器

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

注: プハアップ抵抗値はSCL周波数とバス信号線の容量性負荷に応じて選ばれるべきです。プハアップ抵抗の値については158頁の表 101.をご覧ください。

バス インターフェース部

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

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

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



アドレス一致部

アトンス一致部は受信したアトンス ハイトがTWI アトンス レジスタ(TWAR)の7じット アトンスと一致するかを検査します。TWARで一斉呼び出し 検出許可(TWGCE)ピットが1を書かれると、全ての到着アトンス ピットは一斉呼び出しアトンスに対しても比較されます。アトンス一致で制御 部は通知され、正しい処置を行うことを許します。TWIはTWI制御レジスタ(TWCR)の設定によって、そのアトンスへの応答をするかもしれ ないし、しないかもしれません。アトンス一致部はAVR MCUが主装置によってアトンス指定された場合にMCUを起動できる休止形態の1 つの時でも、アトンスを比較できます。TWIがハプワータウン動作でのアトンス一致でCPUを起動中に他の割り込み(例えばINT0)が起こると、 TWIは動作を停止し、アイトル状態で復帰します。これが何らかの問題の原因なら、パワータウン動作へ移行する時にTWIアトンス一致だ けが割り込みを許可されることを保証してください。

制御部

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

TWI割り込み要求フラグ(TWINT)は次の場合に設定(1)されます。

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



TWI用レジスタ

TWBR - TWI ビット速度レジスタ (TWI Bit Rate Register)

ピット	7	6	5	4	3	2	1	0	_
\$00 (\$20)	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
Read/Write	R/W								
初期値	0	0	0	0	0	0	0	0	

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

TWBRはビット速度発生器用の分周値を選びます。ビット速度発生器は主装置動作でのSCLクロック周波数を生成する周波数分周器です。ビット速度の計算については109頁の「ビット速度発生器」をご覧ください。

TWCR - TWI制御レジスタ (TWI Control Register)

ピット	7	6	5	4	3	2	1	0	
\$36 (\$56)	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
初期値	0	0	0	0	0	0	0	0	

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

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

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

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

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

- 装置が自分用の従装置アドレスを受信した場合。
- TWIアドレス レジスタ(TWAR)の一斉呼び出し検出許可(TWGCE)ビットが設定(1)されている時に一斉呼び出しを受信した場合。
- 主受信装置または従受信装置動作でデータ バイトを受信した場合。

TWEAビットに0を書くことによって一時的かつ仮想的に装置を2線直列バスから切り離すことができます。7ドレス認証はその後に再び TWEAビットへ1を書くことによって再開できます。

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

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

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

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

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

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

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

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

● ビット1 - Res: 予約 (Reserved Bit)

このビットは予約されており、常に0として読みます。



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

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

TWSR - TWI 状態レジスタ (TWI Status Register)

ピット	7	6	5	4	3	2	1	0	
\$01 (\$21)	TWS7	TWS6	TWS5	TWS4	TWS3	-	TWPS1	TWPS0	TWSR
Read/Write	R	R	R	R	R	R	R/W	R/W	
初期値	1	1	1	1	1	0	0	0	

● ビット7~3 - TWS7~3: TWI状態 (TWI Status)

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

● ビット2 - Res: 予約 (Reserved Bit)

このビットは予約されており、常に0として読みます。

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

これらのビットは読み書きでき、ビット速度の前置分周器を制御します。

ビット速度を計算するには109頁の「**ビット速度発生器**」をご覧ください。TWPS1,0の値はこの式で使われます。

表65. TWIL'	小速度	前置分	周器選	択
TWPS1	0	0	1	1
TWPS0	0	1	0	1
分周値	1	4	16	64

TWDR - TWIデータレジスタ (TWI Data Register)

ピット	7	6	5	4	3	2	1	0	
\$03 (\$23)	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0	TWDR
Read/Write	R/W								
初期値	1	1	1	1	1	1	1	1	

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

L*ット7~0 - TWD7~0: TWIデータ (TWI Data)

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

TWAR - TWI(従装置)アトレス レジスタ (TWI (Slave) Address Register)

ピット	7	6	5	4	3	2	1	0	
\$02 (\$22)	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	TWAR
Read/Write	R/W								
初期値	1	1	1	1	1	1	1	0	

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

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

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

これら7ビットはTWI部の従装置アドレスを構成します。

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

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

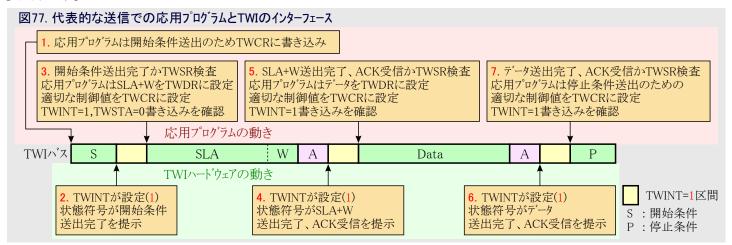


TWIの使用法

AVR TWIは、小志向で割り込みが基本です。割り込みは、小の受信や開始条件の送出のような全ての、、ステータスレジスタ(SREG)のTWIは割り込みが基本のため、応用ソフトウェアはTWI、小転送中に他の操作を続行するために開放されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共にTWI制御レジスタ(TWCR)のTWI割り込み許可(TWIE)ビットは、TWCRのTWI割り込み要求フラケ(TWINT)の設定(1)が割り込み要求を発生すべきかどうか決めることを応用(ソフトウェア)に許します。TWIEビットが解除(0)されると、応用(ソフトウェア)はTWI、スの動きを検知するためにTWINTフラグをポーリングしなければなりません。

TWINTフラケが設定(1)されると、TWIは動作を終え、応用(ソフトウェア)の応答を待ちます。この場合、TWI状態レジスタ(TWSR)はTWIハ、スの現在の状態を示す値を含みます。そして応用ソフトウェアはTWCRとTWDRの操作により、TWIが次のTWIハ、ス周期で何を行うべきかを決定できます。

図77.は応用(ソフトウェア)がTWIハードウェアにどうインターフェースできるかの簡単な例です。この例では主装置が単一データ バイトを従装置に送信しようとします。この内容はかなり大雑把ですので、より詳細な説明が本項の後に続きます。希望した動きを実現する簡単なコード例も示されます。



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



この例は簡単とはいえ、全てのTWI送信に関係した原理を示しています。これらは次のように要約できます。

- TWIが動作を終了して応用(ソフトウェア)の反応を予想する時にTWINTフラグが設定(1)されます。SCL信号線はTWINTが解除(0)されるまでLowに引き込まれます。
- TWINTフラグが設定(1)されると、使用者は次のTWIバス周期に関連した値で、(必要な)全てのTWIレジスタを更新しなければなりません。例で示されるようにTWDRは次のTWIバス周期で送信されるべき値を設定されなければなりません。
- (必要な)全てのTWIレジスタを更新し、その他保留中の応用ソフトウェアの処理が完了されてしまった後にTWCRが書かれます。TWCR 書き込み時、TWINTビットが設定(1)されるべきです。TWINTへの1書き込みはこのフラグを解除(0)します。TWCR設定によってどの動作が指定されても、TWIはその(TWINT=0)後に実行を始めます。

次にアセンブリ言語とC言語の実装例が与えられます。以下のコードは例えばインクルードファイルの使用により、様々な定義が作成されてしまっている前提であることに注意してください。

	アセ	ンブリ言語プログラム例	C言語プログラム例	注釈
1.	LDI OUT	R16, (1< <twint) (1<<twsta)<br="" =""> (1<<twen) TWCR, R16</twen) </twint)>	TWCR = (1< <twint) (1<<twsta)<br="" =""> (1<<twen);< th=""><th>;開始条件送出</th></twen);<></twint)>	;開始条件送出
2.	WAIT1: IN SBRS RJMP	R16, TWCR R16, TWINT WAIT1	while (!(TWCR & (1< <twint)));< th=""><th>;TWINT=1まで待機 ;(開始条件送出完了待機)</th></twint)));<>	;TWINT=1まで待機 ;(開始条件送出完了待機)
0	IN ANDI CPI BRNE	R16, TWSR R16, \$F8 R16, START ERROR	if ((TWSR & 0xF8) != START) ERROR();	;TWI状態レジスタ値検査 ;前置分周選択ビットの遮蔽 ;STARTと異なる状態符号で ;異常処理へ
3.	LDI OUT LDI OUT	R16, SLA_W TWDR, R16 R16, (1< <twint) (1<<twen)<br="" ="">TWCR, R16</twint)>	TWDR = SLA_W; TWCR = (1< <twint) (1<<twen);<="" td="" =""><td>;TWDRにSLA+W設定 ;アドレス送信開始のため ;TWCRのTWINTを解除(0)</td></twint)>	;TWDRにSLA+W設定 ;アドレス送信開始のため ;TWCRのTWINTを解除(0)
4.	WAIT2: IN SBRS RJMP	R16, TWCR R16, TWINT WAIT2	while (!(TWCR & (1< <twint)));< th=""><th>;TWINT=<mark>1</mark>まで待機 ;(SLA+W送出完了と ;ACK/NACK受信完了待機)</th></twint)));<>	;TWINT= <mark>1</mark> まで待機 ;(SLA+W送出完了と ;ACK/NACK受信完了待機)
5.	IN ANDI CPI BRNE	R16, TWSR R16, \$F8 R16, MT_SLA_ACK ERROR	if ((TWSR & 0xF8) != MT_SLA_ACK) ERROR();	;TWI状態レジスタ値検査 ;前置分周選択ビットの遮蔽 ;MT_SLA_ACKと違う状態符号で ;異常処理へ
5.	LDI OUT LDI OUT	R16, DATA TWDR, R16 R16, (1< <twint) (1<<twen)<br="" ="">TWCR, R16</twint)>	TWDR = DATA; TWCR = (1< <twint) (1<<twen);<="" td="" =""><td>;TWDRにデータ設定 ;データ送信開始のため ;TWCRのTWINTを解除(0)</td></twint)>	;TWDRにデータ設定 ;データ送信開始のため ;TWCRのTWINTを解除(0)
6.	WAIT3: IN SBRS RJMP	R16, TWCR R16, TWINT WAIT3	while (!(TWCR & (1< <twint)));< td=""><td>;TWINT=1まで待機 ;(データ送出完了と ;ACK/NACK受信完了待機)</td></twint)));<>	;TWINT=1まで待機 ;(データ送出完了と ;ACK/NACK受信完了待機)
7.	IN ANDI CPI BRNE	R16, TWSR R16, \$F8 R16, MT_DATA_ACK ERROR	if ((TWSR & 0xF8) != MT_DATA_ACK) ERROR();	;TWI状態レジスタ値検査 ;前置分周選択ビットの遮蔽 ;MT_DATA_ACKと違う状態符号で ;異常処理へ
	LDI OUT	R16, (1< <twint) (1<<twsto)<br="" =""> (1<<twen) TWCR, R16</twen) </twint)>	TWCR = (1< <twint) (1<<twsto)<br="" =""> (1<<twen);< td=""><td>;停止条件送出</td></twen);<></twint)>	;停止条件送出



転送種別

TWIは4つの主な動作種別の1つで動けます。これらは送信主装置(MT)、受信主装置(MR)、送信従装置(ST)、受信従装置(SR)と名付けられます。これら種別の多くは同じ応用に使えます。例えば、TWI方式のEEPROM内にデータを書くのにTWIはMT動作を、EEPR OMからデータを読み戻すのにMR動作を使えます。システム内に他の主装置が存在する場合、それらのいくつかがTWIにデータを送信するかもしれず、するとSR動作が使われるでしょう。どの動作種別が適正かを決めるのは応用ソフトウェアです。

次項はこれら動作種別の各々を記述します。起こり得る状態符号は各動作種別のデータ伝送詳細図に沿って示されます。これらの図は次の略号を含みます。

- S 開始(START)条件
- Rs 再送開始(REPEATED START)条件
- R 読み出し指定ビット (SDA=High)
- W 書き込み指定ビット(SDA=Low)
- A 確認応答(ACK)ビット(SDA=Low)
- A 非確認応答(NACK)ビット(SDA=High)
- Data 8ビット データ バイト
- P 停止(STOP)条件
- SLA 従装置アドレス

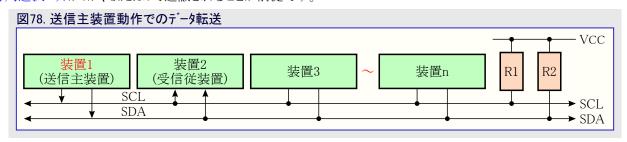
図79.~85.内の楕円(訳注:原文は円)はTWI制御レシ、スタ(TWCR)のTWI割り込み要求フラグ(TWINT)が設定(1)されたことを示すために使われます。この楕円内の番号は前置分周選択ビットが0で遮蔽されたTWI状態レシ、スタ(TWSR)に保持した状態符号を表します。これら位置での動きはTWI転送の継続または完了が応用(ソフトウェア)によって行われなければなりません。TWI転送はソフトウェアによってTWINTフラグが解除(0)されるまで一時停止されます。

TWI割り込み要求フラグ(TWINT)が設定(1)される時のTWI状態レジスタ(TWSR)の状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対する必要なソフトウェア動作や後続の直列転送の詳細は表66.~69.で与えられます。これらの表に於いて前置分周選択ビットが0で遮蔽されていることに注意してください。



送信主装置動作

送信主装置動作では何バイかのデータが受信従装置へ送信されます(図78.参照)。主装置動作へ移行するには開始条件が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。SLA+Wが送信されると送信主装置(MT)へ移行し、SLA+Rが送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は前置分周選択ビットがのか、または0で遮蔽されることが前提です。



開始条件はTWCRに次の値を書くことによって送出されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。TWSTAは開始条件を送出するために1を書かれねばならず、TWINTはTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに開始条件を生成します。開始条件が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、TWSRの状態符号が\$08(表66.参照)になります。送信主装置へ移行するにはSLA+Wが送信されなければなりません。これはTWDRにSLA+Wを書くことによって行います。その後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

SLA+Wが送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$18,\$20,\$38です。これら状態符号の各々に対する適切な動作は表66.で詳述されます。

SLA+Wが正常に送信されてしまうと、データ パケットが送信されるべきです。これはTWDRにデータ バイトを書くことによって行われます。 TWDRはTWINTが1の時にだけ書かれなければなりません。さもなければ、そのアクセスは破棄され、TWCRで上書き発生(TWWC)フラ グが設定(1)されます。 TWDR更新後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これは TWCRに次の値を書くことによって成し遂げられます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

最後のバイトが送られてしまうまでこの手順が繰り返され、この転送は<mark>停止条件</mark>または<mark>再送開始条件</mark>を生成することによって終了されます。 停止条件はTWCRに次の値を書くことによって生成されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	1	X	1	0	X

再送開始条件はTWCRに次の値を書くことによって生成されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

再送開始条件(状態符号\$10)後、2線直列インターフェースは停止条件を送出せずに再び同じ従装置または新しい従装置にアクセスできます。再送開始条件は主装置がハブスの制御を失わずに送信主装置、受信主装置間の切り替えを可能にします(訳注:原文では従装置も含まれていますが、基本動作に対して不適切なため削除しました)。

表66. 送信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

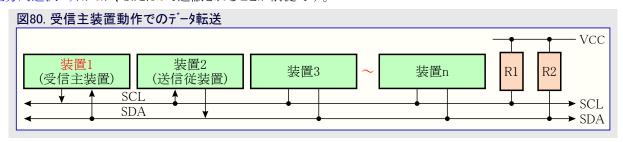
业业体口	ませる私 ルし		ソフトウ	ェアの対応	-		
状態符号 (TWSR)	直前の動作と バスの状態	TWDR操作		TWCF	R設定		TWCR設定によるハート・ウェア動作
(1 Woll)	ハハの水路	TWDN採作	TWSTA	TWSTO	TWINT	TWEA	
\$08	開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
φ10	丹及開始未件及信	SLA+R設定	0	0	1	X	SLA+R送信、受信主装置動作へ移行
		データ設定	0	0	1	X	データ送信、ACKかNACK受信
\$18	SLA+W送信		1	0	1	X	再送開始条件送信
φ10	ACK受信	なし	0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0
		データ設定	0	0	1	X	データ送信、ACKかNACK受信
\$20	SLA+W送信		1	0	1	X	再送開始条件送信
Φ Δ0	NACK受信	なし	0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0
		データ設定	0	0	1	X	データ送信、ACKかNACK受信
\$28	データバイ送信		1	0	1	X	再送開始条件送信
φΔΟ	ACK受信	なし	0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0
		データ設定	0	0	1	X	データ送信、ACKかNACK受信
\$30	データバイ送信		1	0	1	X	再送開始条件送信
φου	NACK受信	なし	0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0
\$38	SLA+W, データ バイトで	なし	0	0	1	X	バス開放、未指定従装置動作へ移行
φοσ	バス競合調停敗退	/s C	1	0	1	X	バス開放時に開始条件送信

図79. 送信主装置動作の形式と状態 主装置送信 W Р 受信従装置への転送成功 S SLA Α Α Data 28 18 (08) 再送開始条件による次転送 Rs SLA W 10 従装置アドレス後の非確認応答受信 Р R 主装置受信 データ バイト後の非確認応答受信 Р (30) アドレスバイトかデータバイトでの 他の主装置転送 A/A 他の主装置転送 バス競合調停敗退 38 A 他の主装置転送 調停敗退で自身の従装置呼び出し検出 (68)(78)(B0) → 状態に従った従装置動作 主装置⇒従装置 従装置⇒主装置 A 何組かのデータと確認応答 (nn) 16進状態符号(TWSR) Data



受信主装置動作

受信主装置動作では何バイかのデータが送信従装置から受信されます(図80.参照)。主装置動作へ移行するには開始条件が送出されなければなりません。それに続くアドレスパケットの形式が送信主装置または受信主装置のどちらへ移行すべきかを決めます。SLA+Wが送信されると送信主装置(MT)へ移行し、SLA+Rが送信されると受信主装置(MR)へ移行します。本項で言及する全ての状態符号は前置分周選択ビットがのか、または0で遮蔽されることが前提です。



開始条件はTWCRに次の値を書くことによって送出されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	1	0	X	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために設定(1)されなければなりません。TWSTAは開始条件を送出するために1を書かれねばならず、TWINTはTWINTフラグを解除(0)するために1を書かれなければなりません。そしてTWIは2線直列バスを検査し、バスが開放になると直ぐに開始条件を生成します。開始条件が送出されてしまった後、TWINTフラグがハードウェアによって設定(1)され、TWSRの状態符号が\$08(表67.参照)になります。受信主装置へ移行するにはSLA+Rが送信されなければなりません。これはTWDRにSLA+Rを書くことによって行います。その後、転送を継続するためにTWINTビットは(1の書き込みによって)解除(0)されるべきです。これはTWCRに次の値を書くことによって成し遂げられます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	0	X	1	0	X

SLA+Rが送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$38,\$40,\$48です。これら状態符号の各々に対する適切な動作は表67.で詳述されます。

ハート・ウェアによってTWINTフラグが設定(1)されると、受信したデータがTWDRから読めます。この手順は最後のバイが受信されてしまうまで繰り返されます。最後のバイが受信されてしまった後、受信主装置は最後に受信したデータバイ後のNACK送信によって送信従装置へ通知すべきです。この転送は停止条件または再送開始条件を生成することによって終了されます。停止条件はTWCRに次の値を書くことによって生成されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	1	X	0	1	X	1	0	X

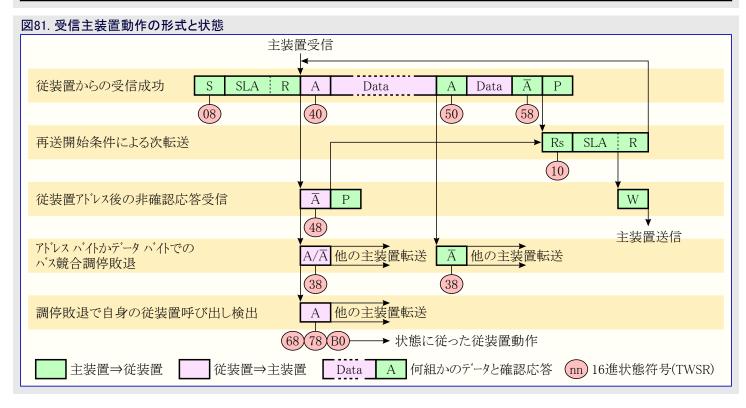
再送開始条件はTWCRに次の値を書くことによって生成されます。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	_	TWIE
設定値	1	X	1	0	X	1	0	X

再送開始条件(状態符号\$10)後、2線直列インターフェースは停止条件を送出せずに再び同じ従装置または新しい従装置にアクセスできます。再送開始条件は主装置がバスの制御を失わずに送信主装置、受信主装置間の切り替えを可能にします(訳注:原文では従装置も含まれていますが、基本動作に対して不適切なため削除しました)。

表67. 受信主装置動作の状態符号 (注: TWSRの前置分周選択 ビットは0の前提)

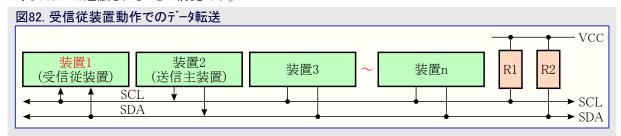
	本芸の私作し		ソフトウェアの対応				
状態符号 (TWSR)	直前の動作と バスの状態	TWDR操作		TWCF	R設定		TWCR設定によるハードウェア動作
(1 77511)	ババの代題	TWDN採作	TWSTA	TWSTO	TWINT	TWEA	
\$08	開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
φ10	丹 及開始未件及信	SLA+W設定	0	0	1	X	SLA+W送信、送信主装置動作へ移行
\$38	SLA+Rで調停敗退	なし	0	0	1	X	バス開放、未指定従装置動作へ移行
φυσ	またはNACK受信	/ ₄ U	1	0	1	X	バス開放時に開始条件送信
\$40	SLA+R送信	なし	0	0	1	0	データ受信、NACK応答
φ40	ACK受信	/ よし	0	0	1	1	データ受信、ACK応答
	CL A LD)光层		1	0	1	X	再送開始条件送信
\$48	SLA+R送信 NACK受信	なし	0	1	1	X	停止条件送信、TWSTO=0
	IVIONXIII		1	1	1	X	停止条件→開始条件送信、TWSTO=0
\$50	データバイ受信	データ取得	0	0	1	0	データ受信、NACK応答
φυσ	ACK応答	/ プリスイサ	0	0	1	1	データ受信、ACK応答
	ニッカバハ巫伝		1	0	1	X	再送開始条件送信
\$58	データ バイト受信 NACK応答	データ取得	0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0





受信従装置動作

受信従装置動作では何バイかのデータが送信主装置から受信されます(図82.参照)。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。



受信従装置動作を始めるにはTWARとTWCRが次のように初期化されなければなりません。

ピット	7	6	5	4	3	2	1	0
TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
設定値			装置自	身の従装置	アトレス			1/0

上位7ビットは主装置によってアドレス指定される時に2線直列インターフェースが応答するアドレスです。最下位(TWGCE)ビットが設定(1)されるなら、TWIは一斉呼び出し(\$00)に応答し、さもなければ一斉呼び出しアドレスを無視します。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	0	1	0	0	0	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために1を書かれなければなりません。TWEAは装置自身の従装置アドレスまたは一斉呼び出しアドレスの確認応答(ACK)を許可するために1を書かれなければなりません。TWSTAとTWSTOは0を書かれなければなりません。

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アトンス(または許可なら、一斉呼び出しアトンス)とそれに続くデータ方向ビットによってアトンス指定されるまで待機します。方向ビットが0(W)ならばTWIは受信従装置で動作し、さもなくば(1(R)ならば)送信従装置へ移行されます。自身の従装置アトンスとWビットが受信されてしまった後にTWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表68.で詳述されます。受信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$68,\$78参照)

転送中にTWEAビットがリセット(0)されると、TWIは次に受信したデータ バイト後のSDAに非確認応答(NACK)(SDA=High)を返します。これは従装置がこれ以上受信できないことを示すのに使えます。TWEAが0の間中、TWIは自身の従装置アドレスに応答しませんが、2線直列バスは未だ監視され、アドレス認証はTWEAの設定(1)によって何時でも再開できます。これはTWEAビットがTWIを2線直列バスから一時的に隔離するのに使えることを意味します。

アイ・ル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAt'ットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラケが(1書き込みによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないことに気付いてください。

これらの(アイドル動作を除く)休止形態から起動すると、2線直列インターフェース データ レジスタ(TWDR)がバスで渡す最後のバイトを反映しないことに注意してください。



表68. 受信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

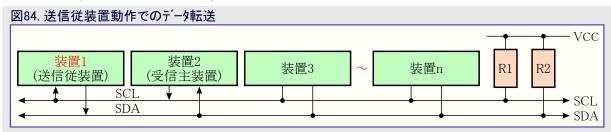
状態符号	直前の動作と			ェアの応答			
(TWSR)	した しょう	TWDR操作		TWCF	設定		TWCR設定によるハードウェア動作
(TWSR)	ハスの仏態	IWDR採TF	TWSTA	TWSTO	TWINT	TWEA	
\$60	自宛SLA+W受信	なし	X	0	1	0	データ受信、NACK応答
φυυ	ACK応答	/ ₄ C	X	0	1	1	データ受信、ACK応答
\$68	主装置のSLA+R/Wで	なし	X	0	1	0	データ受信、NACK応答
\$UO	調停敗退/自宛SLA+ W受信/ACK応答	/ ₄ C	X	0	1	1	データ受信、ACK応答
\$70	一斉呼び出し受信	なし	X	0	1	0	データ受信、NACK応答
\$70	ACK応答	なし	X	0	1	1	データ受信、ACK応答
¢70	主装置のSLA+R/Wで	<i>4</i> >1	X	0	1	0	データ受信、NACK応答
\$78	主装置のSLA+R/Wで調停敗退/一斉呼び出し受信/ACK応答	なし	X	0	1	1	データ受信、ACK応答
¢00	自宛データバイ受信	ニール形相	X	0	1	0	データ受信、NACK応答
\$80	ACK応答	データ取得	X	0	1	1	データ受信、ACK応答
			0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
\$88	自宛データ バイト受信 NACK応答	データ取得	1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$90	一斉呼び出しのデータ	データ取得	X	0	1	0	データ受信、NACK応答
\$90	バイ受信/ACK応答	ノーク取行	X	0	1	1	データ受信、ACK応答
			0	0	1	0	未指定従装置動作へ移行、応答禁止
	一斉呼び出しのデータ		0	0	1	1	未指定従装置動作へ移行、応答対応
\$98	バイ受信 NACK応答	データ取得	1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
			0	0	1	0	未指定従装置動作へ移行、応答禁止
	自指定中の		0	0	1	1	未指定従装置動作へ移行、応答対応
\$A0	停止条件または	なし	1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
	再送開始条件検出		1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信





送信従装置動作

送信従装置動作では何バイトかのデータが送信主装置へ送信されます(**図84**.参照)。本項で言及する全ての状態符号は前置分周選択 ビットが<mark>0</mark>か、または0で遮蔽されることが前提です。



送信従装置動作を始めるにはTWARとTWCRが次のように初期化されなければなりません。

ピット	7	6	5	4	3	2	1	0
TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
設定値			装置自	身の従装置	アトレス			1/0

上位7ビットは主装置によってアドレス指定される時に2線直列インターフェースが応答するアドレスです。最下位(TWGCE)ビットが設定(1)されるなら、TWIは一斉呼び出し(\$00)に応答し、さもなければ一斉呼び出しアドレスを無視します。

ピット	7	6	5	4	3	2	1	0
TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
設定値	0	1	0	0	0	1	0	X

TWENは2線直列インターフェース(TWI)を許可するために1を書かれなければなりません。TWEAは装置自身の従装置アドレスまたは一斉呼び出しアドレスの確認応答(ACK)を許可するために1を書かれなければなりません。TWSTAとTWSTOは0を書かれなければなりません。

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可ならば一斉呼び出しアドレス)とそれに続くデータ方向 ビットによってアドレス指定されるまで待機します。方向ビットが1(R)ならばTWIは送信従装置で動作し、さもなくば(0(W)ならば)受信従装置へ移行されます。自身の従装置アドレスとRビットが受信されてしまった後、TWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。この状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対して行うべき適切な動作は表69.で詳述されます。送信従装置動作はTWIが主装置動作の間で調停に敗れた場合にも移行されるかもしれません。(状態符号\$B0参照)

転送中にTWEAビットが0を書かれると、TWIは転送の最後のバイトを送信します。受信主装置が最終バイト後にACKまたはNACKのどちらを送信するかによって状態\$C0か\$C8へ移行します。TWIはアドレス指定されていない従装置動作に切り替えられ、主装置が転送を続ける場合、その主装置を無視します。従って受信主装置は直列データとして全て1を受信します。従装置が最後のバイトを送信(TWEAが0で主装置からのNACKを予測)したとしても、主装置が(ACK送信によって)追加データ バイトを要求すると状態\$C8へ移行します。

TWEAが0の間中、TWIは自身の従装置アドレスに応答しませんが、2線直列バスは未だ監視され、アドレス認証はTWEAの設定(1)によって何時でも再開できます。これはTWEAビットがTWIを2線直列バスから一時的に隔離するのに使えることを意味します。

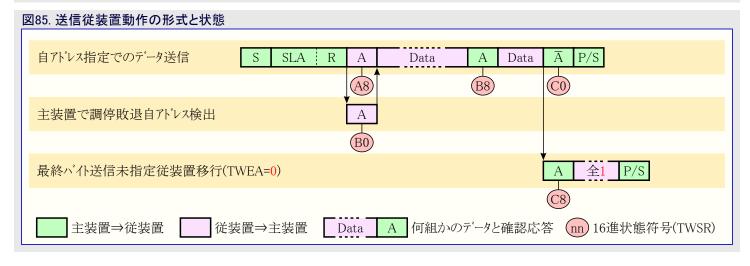
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラケが(1書き込みによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれないことに気付いてください。

これらの(アイドル動作を除く)休止形態から起動すると、2線直列インターフェース データ レジスタ(TWDR)がバスで渡す最後のバイトを反映しないことに注意してください。



表69. 送信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

小台: 外 口	古並の私作し		ソフトウ	ェアの応答	F		
状態符号 (TWSR)	直前の動作と バスの状態	TWDR操作		TWCF	R設定		TWCR設定によるハードウェア動作
(1 77511)	バスの大窓	IWDN採用	TWSTA	TWSTO	TWINT	TWEA	
\$A8	自宛SLA+R受信	データ設定	X	0	1	0	最終データバイル送信、NACK受信予定
ΨΛΟ	ACK応答	/ / 放足	X	0	1	1	データバ小送信、ACK受信予定
\$B0	主装置のSLA+R/Wで 調停時退/自家SLA+	データ設定	X	0	1	0	最終データバイ送信、NACK受信予定
φυσ	調停敗退/自宛SLA+ R受信/ACK応答	/ / 放足	X	0	1	1	データバイ送信、ACK受信予定
\$B8	データバイと信	データ設定	X	0	1	0	最終データバイル送信、NACK受信予定
фДО	ACK受信	/ / 放足	X	0	1	1	データバイ送信、ACK受信予定
			0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
\$C0	データ バイト送信 NACK受信	なし	1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
			0	0	1	0	未指定従装置動作へ移行、応答禁止
	最終データバイ送信		0	0	1	1	未指定従装置動作へ移行、応答対応
\$C8	(TWEA=0) ACK受信	なし	1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信



その他の状態

定義したTWI状態に従わない2つの状態符号があります。表70.をご覧ください。

状態\$F8はTWI割り込み要求フラグ(TWINT)が設定(1)されないので適切な情報が利用できないことを示します。これは他の状態間でTWIが直列転送に関係しない時に起きます。

状態\$00は2線直列バス転送中にバス異常が起きたことを示します。バス異常はフレーム形式の不正な位置で開始(START)条件または停止(STOP)条件が起きる時に発生します。このような不正位置の例はアドレス バイト、データ バイト、確認応答(ACK)ビットの直列転送中です。バス異常が起きるとTWINTが設定(1)されます。バス異常から回復するには、停止(STOP)条件生成許可(TWSTO)ビットが設定(1)され、TWINTが論理1書き込みによって解除(0)されなければなりません。これはTWIをアドレス指定されていない従装置動作にさせ、TW STOビットを解除(0)させます(TWCRの他のビットは影響されません)。SDAとSCL信号進は開放され、停止条件は送出されません。

表70. その他の状態符号 (注: TWSRの前置分周選択 ビットは0の前提)

业业和	古台の私 佐し		ソフトウ	ェアの応答	F				
状態符号 (TWSR)	直前の動作と がえの状態	TWDR操作		TWCF	R設定		TWCR設定によるハート・ウェア動作		
(1 WSIC)	ハスの状態	TWDR採作	TWSTA	TWSTO	TWINT	TWEA			
\$F8	適切な状態情報なし TWINT= <mark>0</mark>	なし	-	-	-	-	待機または現在の転送続行		
\$00	不正な開始条件/停 止条件でのバス異常	なし	0	1	1	X	停止条件を送出せずにバスを開放 TWSTO= <mark>0</mark>		



各種TWI動作種別の組み合わせ

いくつかの場合で望んだ動作を満たすために各々のTWI動作種別は組み合わされなければなりません。例えば直列EEPROMからのデータ読み出しを考えてください。一般的にこのような転送は次の段階を含みます。

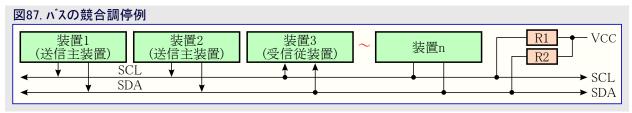
- 1. 転送が開始されなければなりません。
- 2. EEPROMは読み出すべき場所を指示されなければなりません。
- 3. 読み出しが実行されなければなりません。
- 4. 転送が終了されなければなりません。

データが主装置から従装置へとその逆の両方向へ転送されることに注意してください。主装置はどの場所を読みたいかを従装置に指示しなければならず、送信主装置動作の使用を必要とします。その後にデータを従装置から読まねばならず、受信主装置動作の使用を意味します。その後にデータを従装置から読まねばならず、受信主装置動作の使用を意味します。従って転送方向が切り替えられなければなりません。主装置はこれら全ての段階中にバスの制御を保持しなければならず、この手順は排他的(非分断)操作として行われるべきです。複数主装置システムでこの原則に違反すると、他の主装置が②と③段階間でEEPROM内のデータ ポーインタを変更するかもしれず、(元の)主装置は不正なデータ位置を読むでしょう。このような転送方向の切り替えはアトンス バーイの送信とデータの受信間で再送開始条件を送出することによって成し遂げられます。再送開始条件後も主装置はバスの占有権を保持します。次の図はこの転送の流れを示します。



複数主装置システムでのバス競合と調停

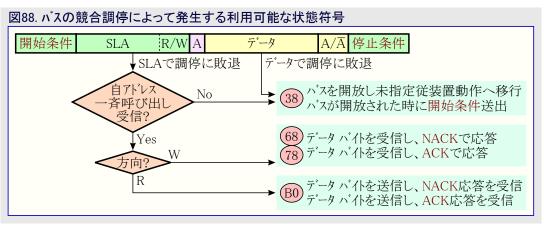
複数の主装置が同じバスに接続されると、それらの1つまたはそれ以上によって同時に送信が開始されるかもしれません。TWIは主装置の1つが転送を続けることを許され、手順内でデータが失われないような方法でこのような状態が扱われることを標準で保証します。2つの主装置が受信従装置へデータを送信することを試みる場合の調停状況の例は以下で図示されます。



以下で示されるように様々な異なる状況が調停中に起こるかもしれません。

- 複数の主装置が同じ従装置に全く同じ通信を実行する場合。この場合、主/従装置のどれもがバスの衝突について知りません。
- 複数の主装置が異なるデータまたは方向ビット(R/W)で同じ従装置をアクセスする場合。この場合、R/Wビットまたはデータ ビットのどちらかで調停が起きます。他の主装置がSDAに0を出力する間に1を出力しようとする主装置が調停に敗れます。 敗れた主装置は応用ソフトウェアの処置によって未指定従装置動作に切り替えるか、またはバスが開放になるまで待って新規開始条件を送出します。
- 複数の主装置が異なる従装置をアクセスする場合。この場合、SLAビット内で調停が起きます。他の主装置がSDAに0を出力する間に1を出力しようとする主装置が調停に敗れます。SLA内で調停に敗れた主装置は勝った主装置によってアドレス指定されるかを検査するために従装置動作へ切り替えます。アドレス指定されると、R/Wビットの値によって受信従装置(SR)動作または送信従装置(ST)動作へ切り替えます。アドレス指定されないなら、応用ソフトウェアの処置によって未指定従装置動作に切り替えるか、またはバスが開放になるまで待って新規開始条件を送出します。

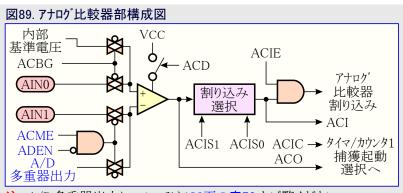
これは**図88**.で要約されます。利用可能な状態符号は楕円(<mark>訳注</mark>:原文は円)で与えられます。





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)ピットが設定(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89.で示されます。



注: A/D多重器出力については126頁の表72.をご覧ください。 アナログ比較器ピン配置については2頁の「ピン配置」と40頁の表28.を 参照してください。

ACSR - アナログ比較器 制御/状態レジスタ (Analog Comparator Control and Status Register)

ピット	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デル動作で電力消費を削減します。ACDビットを変更する時にACSRのアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、このビットが変更される時に割り込みが起こり得ます。

● ビット6 - ACBG: 基準電圧選択 (Analog Comparator Bandgap Select)

このビットが設定(1)されると、内部基準電圧(公称1.30V)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。25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ハーチンが実行されます。対応する割り込み処理ベクタを実行すると、ACIはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもACIは解除(0)されます。

● ビット3 - ACIE: アナログ比較器割り込み許可 (Analog Comparator Interrupt Enable)

ACIEビットが論理1を書かれ、ステータス レジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みが活性(有効)にされます。論理0を書かれると、この割り込みは禁止されます。

● ビット2 - ACIC: アナログ比較器捕獲起動許可 (Analog Comparator Input Capture Enable)

論理1を書かれると、このビットはアナログ比較器によって起動されるタイマ/カウンタ1の捕獲機能を許可します。この場合、比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音消去機能と端(エッジ)選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。論理0を書かれると、アナログ比較器と捕獲機能間の接続は存在しません。比較器がタイマ/カウンタ1捕獲割り込みを起動するには、タイマ/カウンタ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71.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71. アナログ と	比較器割り込∂	4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



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

ピット	7	6	5	4	3	2	1	0	
\$30 (\$50)	_	-	-	-	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット3 - ACME: アナログ比較器多重器許可 (Analog Comparator Multiplexer Enable)

このビットが論理1を書かれ、A/D変換部がOFF(ADCSRAのADENビットが0)にされると、A/D変換の多重器がアナログ比較器への反転入力を選びます。このビットが論理0を書かれると、AIN1がアナログ比較器の反転入力に印加されます。このビットの詳細な記述については次の「アナログ比較器入力選択」をご覧ください。

アナログ比較器入力選択

アナログ比較器への反転入力を取り替えるのにADC7~0のどれかを選ぶことができます。A/D変換の多重器がこの入力選択に使われ、従ってこの機能を利用するためにA/D変換部がOFF(動作禁止)にされなければなりません。SFIORのアナログ比較器多重器許可(ACME)と、かが設定(1)され、A/D変換部がOFF(ADCSRAのADENと、かが0)にされていれば、表72.で示されるようにADMUXのチャネル選択(MUX2~0)と、かアナログ比較器への反転入力を取り替えるための入力と、を選びます。ACMEが解除(0)、またはADENが設定(1)されると、AIN1がアナログ比較器への反転入力に印加されます。

表72. アナログ変換器反転入力選択

ACME	ADEN	MUX2~0	アナログ比較器反転入力	備考
0	X	XXX	AIN1	
	1	XXX	AINI	
		0 0 0	ADC0	
		0 0 1	ADC1	
		0 1 0	ADC2	
1	0	0 1 1	ADC3	
	U	100	ADC4	
		1 0 1	ADC5	
		1 1 0	ADC6	TQFP, QFN/MLF
		111	ADC7	外囲器のみ



A/D変換器

A/D変換部の主な特徴を次に示します。

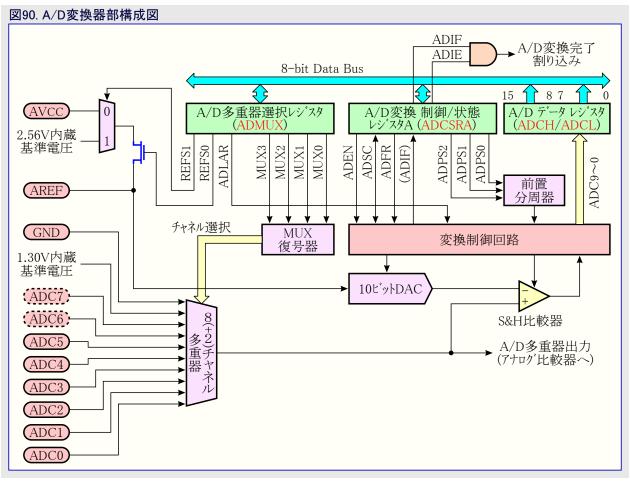
- 10ビット分解能
- 積分非直線性誤差0.5 LSB
- 絶対精度±2 LSB
- 変換時間13~260us
- 76.9kSPS(採取/s)まで(最大分解能で15kSPSまで)
- 6チャネルの多重器内蔵 (PDIP外囲器のみ)
- 8チャネルの多重器内蔵 (TQFP, QFN/MLF外囲器のみ)
- A/D変換結果読み出しに対する任意の左揃え
- 0~VCC A/D変換入力電圧範囲
- 選択可能な2.56V A/D変換基準電圧
- 連続と単独の変換動作
- A/D変換完了割り込み
- 休止形態雑音低減機能

ATmega8は10ビット逐次比較A/D変換器が特徴です。このA/D変換器はポートCのピンから構成された8(6)つのシングル エンド電圧入力を許す8チャネル アナログ多重器に接続されます。このシングル エンド電圧入力は0V(GND)が基準です。

このA/D変換器はA/D変換器への入力電圧が変換中に一定の値で保持されることを保証する採取&保持(S/H)回路を含みます。 A/D変換部の構成図は図90.で示されます。

A/D変換部には分離されたアナログ電源供給ピン(AVCC)があります。AVCCはVCCから±0.3Vよりも多く違ってはなりません。このピンの接続方法は131頁の「**雑音低減技術**」項をご覧ください。

公称2.56Vの内蔵基準電圧またはAVCCがチップ上で提供されます。この基準電圧は雑音特性向上のため、コンデンサによってAREFt[®]ンで外部的にデカップ(雑音分離)できます。





操作

A/D変換部は逐次比較を通してアナログ入力電圧を10ビットのデジタル値に変換します。最小値はGNDを表し、最大値はAREFピンの電圧-1 LSBを表します。A/D多重器選択レジスタ(ADMUX)の基準電圧選択(REFS1,0)ビットへの書き込みにより、任意でAVCCまたは内部 2.56V基準電圧がAREFピンに接続できます。この内部基準電圧は雑音耐性を改善するためにAREFピンで外部コンデンサによってデカップ(雑音結合減少)のようにできます。

アナログ入力チャネルはADMUXのチャネル選択(MUX3~0)ビットへの書き込みによって選ばれます。GNDや公称1.30V内蔵基準電圧(VBG) だけでなくA/D変換入力ピンの何れもがA/D変換器へのシングルエンド入力として選べます。

A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットの設定(1)によって(動作が)許可されます。基準電圧と入力チャネルの選択はADENが設定(1)されるまで実施しません。ADENが解除(0)されているとA/D変換部は電力を消費しないので、節電をする休止形態へ移行する前にA/D変換部をOFFに切り替えることが推奨されます。

A/D変換部はA/Dデータレシ、スタ(ADCH,ADCL)で示される10ビットの結果を生成します。既定では、この結果は右揃え(16ビットのビットの 側10ビット)で示されますが、ADMUXで左揃え選択(ADLAR)ビットを設定(1)することにより、任意で左揃え(16ビットのビット15側10ビット)で示せます。

この結果が左揃え補正され、8ビットを越える精度が必要とされない場合はADCHを読むことで足ります。さもなければデータレジスタの内容が同じ変換に属すこと(からの結果)を保証するため、ADCLが初めに、次にADCHが読まれなければなりません。一度ADCLが読まれると、A/D変換器からのA/Dデータレジスタ(ADCH,ADCL)アクセスが阻止されます。これはADCLが読まれてしまい、ADCHが読まれる前に変換が完了すると、どちらのレジスタ(ADCH,ADCL)も更新されず、その変換からの結果が失われることを意味します。ADCHが読まれると、ADCH,ADCLへのA/D変換器アクセスが再び許可されます。

A/D変換部には変換完了時に起動できる自身の割り込みがあります。A/DデータレジスタへのA/D変換器アクセスがADCLとADCHの読み込み間で禁止されている場合、例えその変換結果が失われても割り込みは起動します。

変換の開始

単独変換はADCSRAで変換開始(ADSC)ビットに論理1を書くことによって開始されます。このビットは変換が進行中である限り、1に留まり、変換が完了されるとハードウェアによって解除(0)されます。変換が進行中に違う入力チャネルが選ばれると、A/D変換部はそのチャネル変更を実行する前に現在の変換を済ませます。

連続変換動作でのA/D変換部は継続的に採取/変換とA/Dデータレシ、スタを更新します。連続変換動作はADCSRAでA/D動作選択 (ADFR)ビットに1を書くことによって選ばれます。最初の変換はADCSRAでADSCビットに論理1を書くことによって始めなければなりません。この動作でのA/D変換器はA/D変換完了割り込み要求フラグ(ADIF)が解除(0)されるかどうかに拘らず、連続的な変換を実行します。

変換はA/D変換雑音低減機能の使用によっても開始され得ます。この機能はアイドル休止動作とA/D変換雑音低減休止動作中に変換を許可します。詳細については130頁の「**雑音低減機能**」をご覧ください。(<mark>訳注</mark>:共通性から2行追加)

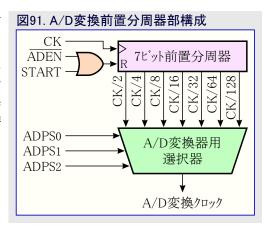


前置分周と変換タイミング

既定での逐次比較回路は最大分解能を得るのに50~200kHzの入力クロック周波数を必要とします。10ビットよりも低い分解能が必要とされるなら、A/D変換器への入力クロック周波数はより高い採取速度を得るために200kHzよりも高くできます。

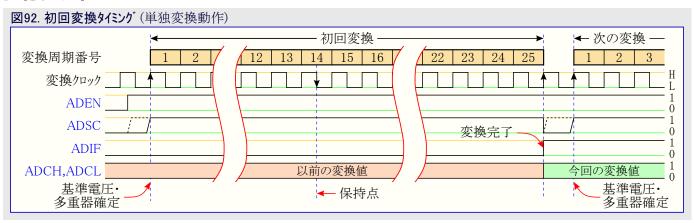
A/D変換部は100kHz以上のどんなCPUクロックからも受け入れ可能なA/D変換クロック周波数を生成する前置分周器を含みます。この前置分周はA/D変換制御/状態レジスタA(ADCSRA)のA/Dクロック選択(ADPS2~0)ビットによって設定されます。前置分周器はADCSRAでA/D許可(ADEN)ビットの設定(1)によってA/D変換部がONにされた瞬間から計数を始めます。前置分周器はADENビットが設定(1)される限り走行を保ち、ADENが0の時は継続的にリセットします。

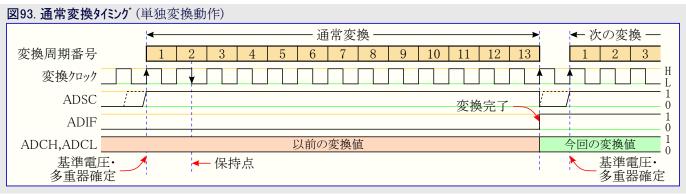
ADCSRAでA/D変換開始(ADSC)ビットの設定(1)によって単独変換を起動すると、その変換は直後の変換クロックの上昇端で始まります。通常の変換は13変換クロック周期で行われます。A/D変換部がONされる(ADCSRAのADEN=1)後の最初の変換はアナログ回路を初期化するために25変換クロック周期で行われます。



実際の採取&保持(保持開始点)は通常変換の開始後1.5変換クロック周期、初回変換の開始後13.5変換クロック周期で行われます。変換が完了すると、結果がA/Dデータレジスタ(ADCH,ADCL)に書かれ、ADCSRAのA/D変換完了割り込み要求フラグ(ADIF)が設定(1)されます。単独変換動作(ADFR=0)では同時にADCSRAのA/D変換開始(ADSC)ビットが解除(0)されます。その後にソフトウェアは再びADSCを設定(1)でき、新規変換は変換クロックの最初の上昇端で開始されます。

連続変換動作(ADFR=1)では変換完了後直ちに新規変換が開始され、一方ADSCは1に留まります。変換時間の概要については表73.をご覧ください。





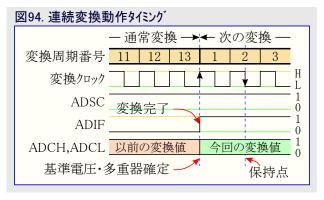


表 /3. A/D変	哭吁间			
変換種別	保持点	変換終了	総変換時間	変換時間(µs)
連続(初回)	13.5	25	25	125~500
単独(初回)	13.5	25	26	130~520
連続(通常)	1.5	13	13	65~260
単独(通常)	1.5	13	14	70~280

注: 変換時間を除く各値は変換開始からの変換クロック数です。

チャネル変更と基準電圧選択

A/D多重器選択レシ、スタ(ADMUX)のチャネル選択(MUX3~0)ピットと基準電圧選択(REFS1,0)ピットはCPUが乱順にアクセスするための一時レジスタを通して単独緩衝されます。これはチャネルと基準電圧の選択が変換中の安全なところでだけ行うのを保証します。チャネルと基準電圧の選択は変換が開始されるまで継続的に更新されます。一旦変換が始まると、A/D変換器に対して充分な採取/変換時間を保証するためにチャネルと基準電圧の選択は固定されます。継続的な更新は変換完了(ADCSRAのADIF=1)前の最後の変換クロック周期で再開します。ADCSRAの変換開始(ADSC)ピットが書かれた後の次の変換クロックの上昇端で変換が始まることに注意してください。従って使用者はADSC書き込み後、1変換クロック周期(経過)まで新しいチャネルまたは基準電圧選択値をADMUXに書かないことを推奨されます。

ADCSRAのA/D許可(ADEN)と連続/単独変換動作選択(ADFR)の両方が1を書かれると、何時でも割り込みが起き得ます。この期間でADMUXが変更されると、使用者は次の変換が旧設定または新設定どちらが基準にされるかを知ることができません。ADMUXは次の方法で安全に更新できます。

- ADENまたはADFRが解除(0)されているとき。
- 変換開始後、最低1変換クロック周期経過後の変換中。
- 変換後から、変換起動元として使った割り込みフラケが解除(0)される直前まで。(<mark>訳注</mark>:自動起動機能用記述、ATmega8はなし) これら条件の1つでADMUXを更新すると、新設定は次のA/D変換に影響を及ぼします。

A/D入力チャネル

チャネル選択を変更する時に使用者は正しいチャネルが選ばれることを保証するために次の指針を守るべきです。

単独変換動作では常に変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法はチャネル選択を変更する前に変換が完了するまで待つことです。

連続変換動作では常に最初の変換を始める前にチャネルを選んでください。チャネル選択はADSCへの1書き込み後、1変換クロック周期で変更されるかもしれません。とは言え、最も簡単な方法は最初の変換が完了するまで待ち、その後にチャネル選択を変更することです。既に次の変換が自動的に開始されているので、次の結果は直前のチャネル選択を反映します。それに続く変換は新しいチャネル選択を反映します。

A/D変換基準電圧

このA/D変換用の基準電圧(VREF)はA/D変換に対する変換範囲を示します。VREFを越えるシングルェンド入力チャネルは\$3FFで打ち切る符号に帰着します。VREFはAVCC、内部2.56V基準電圧、外部AREFt'ンのどれかとして選べます。

AVCCは受動型スイッチを通してA/D変換部に接続されます。内部2.56V基準電圧は内蔵基準(バンドギャップ)電圧(VBG)から内部増幅器を通して生成されます。どちらの場合でも外部AREFピンは直接的にA/D変換部へ接続され、AREFピンとGND間にコンデンサを接続することにより、基準電圧は雑音耐性をより高められます。VREF(電圧)は高入力インピーダンス電圧計とAREFピンで測定することもできます。VREFは高インピーダンス出力で、容量性負荷のみがシステム内で接続されるべきであることに注意してださい。

使用者がAREFt°ンに接続された固定電圧源にするなら、この外部電圧がその他の内部基準電圧と短絡してしまうため、使用者はこの応用内で他の基準電圧選択を使ってはなりません。外部電圧がAREFt°ンに印加されないなら、使用者は基準電圧選択としてAVCCと内部2.56V基準電圧間の切り替えができます。基準電圧源切り替え後の最初のA/D変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

雑音低減機能

このA/D変換部はCPUコアと他の周辺I/Oが誘導した雑音を削減するために休止形態中の変換を可能にする雑音低減機能が特徴です。この機能はA/D変換雑音低減動作とアイドル動作で使えます。この機能を使うには次の手順が使われるべきです。

- 1. A/D変換部が許可(ADEN=1)され、変換中でない(ADSC=0)ことを確認してください。単独変換動作が選択(ADFR=0)され、且つ A/D変換完了割り込みが許可(ADIE=1)されていなければなりません。
- 2. A/D変換雑音低減(またはアイドル)動作に移行してください。一旦CPUが停止されてしまうと、A/D変換部は変換を始めます。
- 3. A/D変換完了前に他の割り込みが起こらなければ、A/D変換完了割り込みはCPUを起動してA/D変換完了割り込みルーチンを実行します。A/D変換完了前に他の割り込みがCPUを起動すると、その割り込みが実行され、A/D変換完了割り込み要求はA/D変換完了時に生成されます。CPUは新規SLEEP命令が実行されるまで活動動作に留まります。

アイドル動作とA/D変換雑音低減動作を除く他の休止形態へ移行する時にA/D変換部が自動的にOFFへ切り替えられないことに注意してください。使用者は余分な消費電力を避けるため、このような休止形態へ移行する前にADENへ0を書くことが推奨されます。

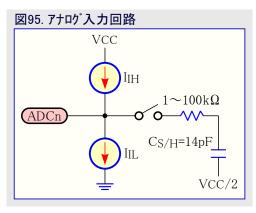


アナログ入力回路

シングル エンド、入力チャネルのアナログ回路は**図95**.で図示されます。ADCnに印加したアナログ(信号)源はそのチャネルがADC入力として選ばれているかどうかに拘らず、ピン容量とそのピンの漏れ電流に左右されます。そのチャネルが選ばれると、(アナログ信号)源は直列抵抗(入力経路の合成抵抗)を通してS/Hコンデンサを駆動しなければなりません。

A/D変換部は概ね10kΩ若しくはそれ以下の出力インピーダンスのアナログ信号用に最適化されています。このような(アナログ信号)源が使われるなら、採取時間は無視してもよいでしょう。より高いインピーダンスの(アナログ信号)源が使われる場合、採取時間は広範囲に変化し得るS/Hコンデンサを充電するために(アナログ信号)源がどれくらいの時間を必要とするかに依存します。必要とされるS/Hコンデンサへの充放電を最小とするため、使用者は緩やかに変化する低インピーダンス(アナログ信号)源だけを使うことが推奨されます。

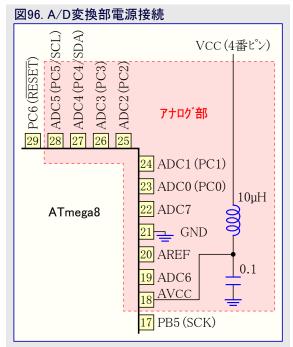
特定できない信号の渦からの歪を避けるために、どのチャネルに対してもナイキスト周波数(fADC/2)よりも高い信号成分が存在すべきではありません。使用者はADC入力として信号を印加する前に低域通過濾波器(ローパプスフィルタ)で高い周波数成分を取り除くことが推奨されます。



アナログ雑音低減技術

デバイス内外のデジタル回路がアナログ測定の精度に影響を及ぼすかもしれないEMIを発生します。精密な変換精度が必要な場合、次の技法を適用することによって雑音レベルを低減できます。

- 1. アナログ信号経路を可能な限り最短にしてください。アナログ信号線がGND面上を走ることに注意し、高速切り替えデジタル信号線から充分離すことを守ってください。
- 2. デバイスのAVCCピンは図96.で示されるようにLC濾波器を経由してデジタル供給電圧(VCC)に接続されるべきです。
- 3. CPUからの誘導雑音を低減するためにA/D変換の雑音低減機能を使ってください。
- 4. どれかのADC0~3 ポート ピンがデジタル出力として使われる場合、これらが変換進行中に切り替わらないことが重要です。けれども 2線直列インターフェース(ADC4とADC5)の使用はADC4とADC5の変換にのみ影響し、他のADCチャネルには影響しません。



(訳注)

本図は原図に対し修正されています。

A/D変換の精度定義

シングル エント、入力電圧のnt゙ットA/D変換はGNDとVREF間を 2^n で直線的に変換します。最低値符号は0として読み、最高値符号は 2^n -1として読みます。以下の各種パラメータは理想状態からの偏差を表します。

• 変位(オフセット)誤差 - 図97.

最初の遷移点(\$000から\$001)で理想遷移点(差0.5 LSB)と比べた偏差です。理想値は0 LSBです。

• 利得誤差 - 図98.

変位誤差補正後の最後の遷移点(\$3FEから\$3FF)で理想遷 移点(最大差1.5 LSB以下)と比べた偏差です。理想値は0 LSBです。

• 積分非直線性誤差 (INL) - 図99.

変位誤差と利得誤差補正後の全ての遷移点で理想遷移点と比べた最大偏差です。理想値は0LSBです。

• 微分非直線性誤差 (DNL) - 図100.

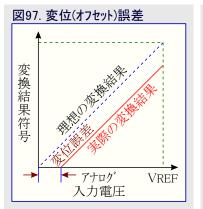
実際の符号の幅(隣接する2つの遷移点間)で理想符号幅(1 LSB)と比べた最大偏差です。理想値は0 LSBです。

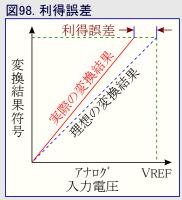
• 量子化誤差

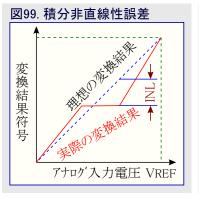
有限数の符号で入力電圧を量子化するため、1 LSB幅となる入力電圧範囲は同じ値の符号になります。この値は常に±0.5 LSBです。

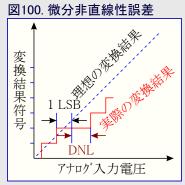
絶対精度

補正しない全ての遷移点で理想遷移点と比べた最大偏差です。これは、変位誤差、利得誤差、差動誤差、非直線誤差の影響の合成です。理想値は±0.5 LSBです。









A/D変換の結果

変換完了(ADIF=1)後、変換結果はA/Dデータレジスタ(ADCH, ADCL)で得られます。 シングルエント・入力変換での結果は次式で示されます。

$$ADC = \frac{V_{IN} \times 1024}{V_{REF}}$$

VINは選んだ入力ピンの電圧で、VREFは選んだ基準電圧です(133頁の表74.と表75.をご覧ください)。 \$000はGNDを表し、\$3FFは選んだ基準電圧-1 LSBを表します。

A/D変換用レジスタ

ADMUX - A/D多重器選択レジスタ (ADC Multiplexer Select Register)

ピット	7	6	5	4	3	2	1	0	
\$07 (\$27)	REFS1	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7.6 - REFS1.0: 基準電圧選択 (Reference Select Bits 1, 0)

これらのビットは表74.で示されるようにA/D変換器の基準電圧を選びます。これらのビットが変換中に変更されると、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。AREFピンに外部基準電圧が印加される場合、内部の基準電圧が使われてはなりません。

表74. A/D変換部の基準電圧選択

REFS1	REFS0	基準電圧
0	0	AREFt°ンの外部基準電圧(AVCCと内部2.56V基準電圧は切り離されます。)
0	1	AVCC (内部2.56V基準電圧は切り離されますが、AREFにデカップ用コンデンサが接続できます。)
1	0	(予約)
1	1	内部2.56V基準電圧(AVCCは切り離されますが、AREFにデカップ用コンデンサが接続できます。)

● ビット5 - ADLAR: 左揃え選択 (ADC Left Adjust Result)

ADLARビットはA/Dデータ レジスタ内の変換結果の配置に影響を及ぼします。結果を左揃えにするにはADLARに1を書いてください。さもなければ結果は右揃えです。ADLARビットの変更はどんな進行中の変換にも拘らず、直ちにA/Dデータ レジスタの内容に影響を及ぼします。このビットの完全な記述については134頁の「A/Dデータ レジスタ」をご覧ください。

● ビット3~0 - MUX3~0: A/Dチャネル選択 (Analog Channel Select Bits 3~0)

これらのビットの値はA/D変換器に接続されるアナログ入力を選びます。詳細については表75.をご覧ください。これらのビットが変換中に変更される場合、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。

表75.アナログ入力チャネル選択

201017717777	1 1 1 1 2 2 37	`									
MUX3~0	0000	0001	0010	0011	0100	0101	0110	0111	1000~1101	1110	1111
アナログ入力チャネル	ADC0	ADC1	ADC2	ADC3	ADC4	ADC5	(ADC6)	(ADC7)	(予約)	1.30V	0V
備考	PC0	PC1	PC2	PC3	PC4	PC5	ADC6	ADC7		VBG	GND

(訳注) PDIP外囲器は6チャネルのため、ADC6,7はありません。

ADCSRA - A/D制御/状態レジスタA (ADC Control and Status Register A)

ピット	7	6	5	4	3	2	1	0	
\$06 (\$26)	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
初期値	0	0	0	0	0	0	0	0	

● ビット7 - ADEN: A/D許可 (ADC Enable)

このビットに1を書くことがA/D変換部(動作)を許可します。0を書くことによってA/D変換部は(電源が)OFFされます。変換が進行中に A/D変換部をOFFにすることはその変換を(途中)終了します。

● ビット6 - ADSC: A/D変換開始 (ADC Start Conversion)

単独変換動作で各変換を始めるにはこのビットへ1を書いてください。連続変換動作で最初の変換を始めるにはこのビットへ1を書いてください。A/D変換部が許可される(ADEN=1)と同時にADSCが書かれるか、またはA/D変換部が許可されてしまった後にADSCが書かれた後の初回変換は、通常の13に代わって25変換クロック周期で行います。この初回変換はA/D変換部の初期化を実行します。

ADSCは変換が進行中である限り1として読みます。変換が完了すると0に戻ります。このビットへの0書き込みは無効です。

● ビット5 - ADFR:連続/単独変換動作選択(ADC Free Running Select)

このビットが設定(1)されるとA/D変換器は連続変換動作で動作します。この動作種別では連続的にA/D変換の採取/変換とA/Dデータ レジスタを更新します。このビットを解除(0)することが連続変換動作を終了します(単独変換動作になります)。

● ビット4 - ADIF: A/D変換完了割り込み要求フラグ(ADC Interrupt Flag)

A/D変換が完了し、A/Dデータレジスタが更新されると、このフラケが設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビットと A/D変換完了割り込み許可(ADIE)ビットが設定(1)されていれば、A/D変換完了割り込みが実行されます。対応する割り込み処理へクタを実行する時にADIFはハートウェアによって解除(0)されます。代わりにこのフラケに論理1を書くことによってもADIFは解除(0)されます。 ADCSRAで読み-変更-書き(リート・モディファイライト)を行うと、保留中の割り込みが禁止され得ることに注意してください。これはSBI,CBI 命令が使われる場合にも適用されます。

● ビット3 - ADIE: A/D変換完了割り込み許可(ADC Interrupt Enable)

このビットが1を書かれ、SREGの全割り込み許可(I)ビットが設定(1)されていると、A/D変換完了割り込みが活性に(許可)されます。この ビットが解除(0)されると、この割り込みは禁止されます(訳注:共通性のため本行追加)。

● ビット2~0 - ADPS2~0: A/D変換クロック選択 (ADC Prescaler Select Bits)

これらのビットはXTAL(システム)周波数とA/D変換部への入力クロック間の分周値を決めます。

表76. A/D変換クロッ	ク選択 (CK=シ	ステム クロック)						
ADPS2	0	0	0	0	1	1	1	1
ADPS1	0	0	1	1	0	0	1	1
ADPS0	0	1	0	1	0	1	0	1
A/D変換クロック	CK/2	CK/2	CK/4	CK/8	CK/16	CK/32	CK/64	CK/128

ADCH,ADCL - A/Dデータレジスタ (ADC Data Register)

	ADLAR=0	時							
ピット	15	14	13	12	11	10	9	8	_
\$05 (\$25)	_	-	-	-	-	-	ADC9	ADC8	ADCH
Read/Write	R	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
ピット	7	6	5	4	3	2	1	0	
\$04 (\$24)	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
Read/Write	R	D	D	D	D	D	D	D	
read/ write	Л	R	R	R	R	R	R	R	
初期値	0	0	0	0	0	0	0	0	
	0	0				= -			
		0				= -			
	0 ADLAR=1	0 時	0	0	0	0	0	0] ADCH
	0 ADLAR=1	0 時 14	0 13	0 12	0 11	0	0 9	0 8] ADCH
	0 ADLAR=1 15 ADC9	0 時 14 ADC8	0 13 ADC7	0 12 ADC6	0 11 ADC5	0 10 ADC4	0 9	0 8 ADC2] ADCH

A/D変換が完了すると、その結果がこれら2つのレジスタで得られます。

ADCLが読まれると、A/DデータレジスタはADCHが読まれるまで更新されません。従ってこの結果が左揃えで、且つ8ビットを越える精度が必要とされないなら、ADCHを読むことで用が足ります。さもなければADCLが初めに、その後にADCHが読まれなければなりません。

A/D多重器選択レシブスタ(ADMUX)の左揃え選択(ADLAR)ビット(<mark>訳注</mark>:原文では「とA/Dチャネル選択(MUX3~0)ビット」となっていますが、ATmega8は全チャネル同一形態の(差動入力がない)ため、この部分は不適切です)は、このレジブスタから結果を読む方法に影響を及ぼします。ADLARが設定(1)されると結果は左揃えにされます。ADLARが解除(0)されていると(既定)、結果は右揃えにされます。

● ADC9~0: A/D変換結果 (ADC Conversion result)

これらのビットは132頁の「A/D変換の結果」で詳述されるように変換での結果を表します。



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ト、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トを書き(プログラム)、コートを読み、またはプログラムメモリからコートを読むのに、利用可能なデータインターフェースと関連する規約のどれもが使えます。ブートローダ領域内のプログラムコードはブートローダメモリを含むフラッシュメモリ全体を書く能力を持ちます。従ってブートローダは自身をも変更でき、この機能がそれ以上必要とされないなら、そのコードから自身を消去することもできます。ブートローダ、メモリの容量はヒューズで設定可能で、ブートローダは個別に設定可能な2組の独立したブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

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)
- 効率的なコート・手法
- 効率的な読み-変更-書き(リート、モデファイライト)支援

注1: ページ はプログラミング 中に使われる多数のバイトから成るフラッシュ メモリの区画です(145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102.参照)。各領域の容量は143頁の表82.と図102.で示されるようにBOOTSZヒューズによって設定されます。これら2つの領域は個別の施錠ビットの組を持つため、異なる保護基準を持てます。

応用領域

応用領域は応用コートを格納するのに使われるフラッシュ メモリの領域です。応用領域用保護基準は応用ブート施錠ビット(ブート施錠ビット0) によって選べます(137頁の表78.参照)。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 コート・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ト・格納用に使われるのに対して、SPM命令はBLSから実行する時にだけプログラミングを始められるので、ブートローダソフトウェアはBLSに格納されなければなりません。SPM命令はBLS自身を含む全てのフラッシュメモリをアクセスできます。ブートローダ領域用保護基準はブートローダ・施錠ビット(ブート施錠ビット1)によって選べます(137頁の表79.参照)。

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

どちらのアト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ヒュース、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136頁の図102と143頁の表83.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ヘーン・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ト・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

ブートロータ・ソフトウェア更新がRWW領域側のペーシをプログラミングする場合、フラッシュメモリからコートを読むことが可能ですが、NRWW領域に配置されるコートだけです。プログラミング実行中、そのソフトウェアはRWW領域が決して読まれないことを保証しなければなりません。使用者ソフトウェアがプログラミング中に(例えば、CALL,JMP,LPM系命令または割り込みによって)RWW領域側に配置されるコートを読もうとすると、そのソフトウェアは未知の状態へ行き着くかもしれません。これを避けるために割り込みは禁止、またはブートロータ、領域へ移動のどちらかにされるべきです。ブートロータ、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命令制御レシ、スタ(SPMCR)のRWW領域多忙(RWWSB)ビットは論理してで読みます。プログラミングが完了した後、RWW領域に配置したコートを読む前に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138頁の「SPMCR - SPM命令制御レジスタ」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タ゛ソフトウェアがRWW領域内のヘ゜ーシ゛を更新する時に読めます。 ブートロータ゛コードがNRWW領域を更新するとき、全てのヘ゜ーシ゛消去またはヘ゜ーシ゛書き込み動作中にCPUが停止されます。

表77. 書き込み中読み出し可能機能

プログラミング中にZポインタで指定される領域	プログラミング中に読める領域	CPU動作	RWW機能支援
RWW領域	NRWW領域	通常動作	あり
NRWW領域	なし	停止	なし

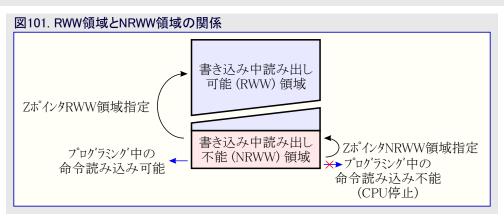
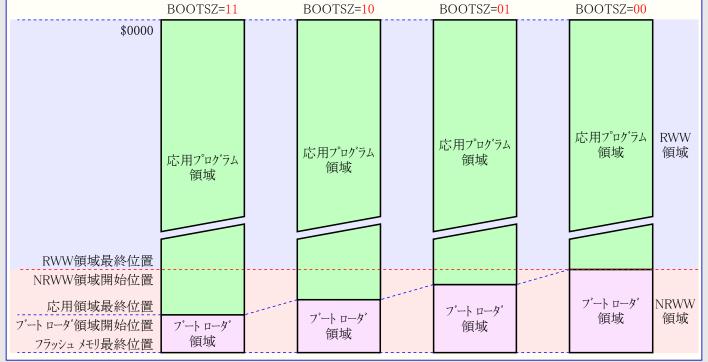


図102. 選択によるプログラム用フラッシュ メモリの領域分割 BOOTSZ=11



注: 図内の各パラメータについては143頁の表82.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ト・用に利用可能です。ブート ローダは個別に設定可能な独立した2組のブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

使用者は以下を選べます。

- MCUによって更新するソフトウェアからフラッシュ メモリ全体を保護
- MCUによって更新するソフトウェアからフラッシュ メモリのフート 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78.と表79.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78. 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、ブートローが領域での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79. 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タ、領域への書き込みと、応用領域でのLPM命令ロータ、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。

ブートローダブログラムへの移行

ブートロータ、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、グタがブートロータ、領域開始アトンスを指示するようにブートリセット(BOOTRST)ヒュース、をプログラム(0)できます。この場合、ブートロータがリセット後に開始されます。応用コートが設定された(書かれた)後、このプログラム(ブートロータ)は応用コートの実行を始めることができます。このヒュースが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スがプログラム(0)されると、リセット、グクタは常にブートロータ、リセットを指示し、このヒュース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80. ブート リセット ヒュース (()=プログラム、1=非プログラム)

BOOTRST		リセット後実行開始アトレス (リセット ヘ・クタ)
0	ブート ロータ゛リセット	ブートローダ 開始アドレス(143頁の 表82 .参照)
1	応用リセット	\$0000

SPMCR - SPM命令制御レジスタ (Store Program Memory Control Register)

このレシ、スタはフートローダ、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。

ピット	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)

SPMIEL'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)ヒットが設定(1)されていれば、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(ヘーン)消去またはヘーン・書き込み)操作が開始されると、RWWSBがハート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、ヘーン・設定操作が開始されると、RWWSBビットは自動的に解除(0)されます。

ビット5 - Res: 予約 (Reserved)

このビットは予約されており、常に0として読みます。

● ビット4 - RWWSRE: RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域にプロケラミンケ(ペーシ)消去またはペーシ。書き込み)すると、RWW領域は(RWWSBがハート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ケラミンケが完了(SPMENが解除(0))されるまで待たねばなりません。その後、RWWSREt、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。ペーシ消去またはページ。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RWWSREt、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(ページ・一時緩衝部はRWW領域が再許可される時に解除されます)。

● 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141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。RIとRO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トレス指定したヘーン・一時緩衝部へR1:R0の値を格納します。Zレシ、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。ヘーン・消去とヘーン・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。



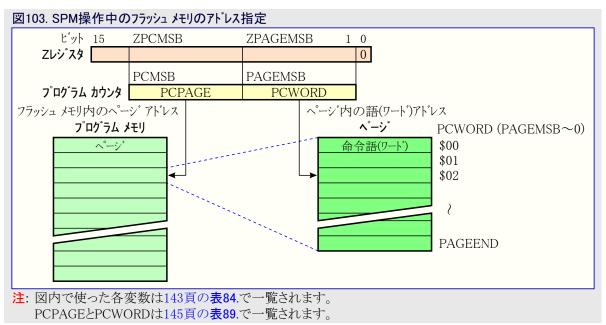
自己プログラミング中のフラッシュメモリのアドレス指定

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。

ピット	15	14	13	12	11	10	9	8
ZH (R31)	Z15	Z14	Z13	Z12	Z11	Z10	Z9	Z8
ピット	7	6	5	4	3	2	1	0
ZL (R30)	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

フラッシュ メモリがヘージで構成されるため(145頁の表89.参照)、プログラム カウンタ(アト・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、ヘージ内の語(ワード)をアト・レス指定し、一方上位側ビットはそのヘージをアト・レス指定します。これは図103.で示されます。ヘージ消去とヘージ書き込み操作が個別にアト・レス指定されることに注意してください。従ってブートロータ・ソフトウェアはヘージ消去とヘージ書き込み操作の両方で同じヘージをアト・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、このアト・レスはラッチされ、ス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。



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 ^゚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(ペーシ)消去前の一時緩衝部格納)

- ページー時緩衝部を満たしてください。
- ぺ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)

- ぺージ消去を実行してください。
- ページー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。

へージの一部の変更だけが必要な場合、消去前にヘージの残す部分は(例えばヘージー・時緩衝部に)保存されなければならず、その後に改めて書かれます。手段1.を使う場合、初めにヘ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ー修正ー書き(リート・モデファイライト)機能をブートローダが提供します。手段2.が使われる場合、ヘ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。ヘージー・時緩衝部は乱順でアクセスできます。ヘージ消去とヘージ書き込み操作の両方で使われるヘージ、アトレスは同じヘージをアト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ト・例については142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

へ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無視されます。

- RWW領域のページ消去 : ページ消去中、NRWW領域は読めます。
- NRWW領域のページ消去:ページ消去中、CPUは停止されます。
- 注: 時間手順内で割り込みが起きた場合に4周期アクセスが保証できません。非分断操作を保証するためにSPMCRへ書く前に割り込みを禁止しなければなりません。

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

へージ書き込みを行うにはZポインタにアトレスを設定して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。ヘージ アトレスは(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領域のヘージ書き込み : ヘージ書き込み中、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SPMENt'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、ブ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(ベブクタ)の移動法は28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ト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28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領域をアトレス指定する前に、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142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。

R0 1 1 BLB12 BLB3		
1 1 BEB12 BEB	1 BLB02 BL	02 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 ロータ施錠ビットの各種設定法については**表78**.と**表79**.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とプ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、対応するプ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スと施錠 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(EEWE) ヒットを検査し、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ス、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ス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とブート施錠ビット設定(BLBSET)ビットを設定(1)してください。SPMENとBLBSETビットが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シ、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。

ピット	7	6	5	4	3	2	1	0
Rd	_	ı	BLB12	BLB11	BLB02	BLB01	LB2	LB1

ヒュース、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。ヒュース、下位ビットを読み出すにはZポインタに \$0000を設定してSPM CR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ス、下位ビット(FLB)の値が転送先レシ、スタに格納されます。ヒュース、下位ビットの配置と詳細な記述については145頁の表88.を参照してください。

ピット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様に、ヒュース・上位ビットを読むにはZポーインタに \$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ス・上位ビット(FHB)の値が転送先レシ、スタに格納されます。ヒュース・上位ビットの配置と詳細な記述については145頁の表87.を参照してください。

ピット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

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

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自身が命令を間違って実行し得ます。 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81.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。

非 Ω1	SPM命令によるフラッシュ	メエリのプログラミング時間
สฐดเ.	OFINION THICAGO / バソノエ	グエリひょう ロン フミンソ 山井1日1

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

注: 最小と最大の時間は(項目の)個別操作毎に対してです。



アセンブリ言語による簡単なブートローダ例

このルーチンはRAMからフラッシュメモリへ1ペーシ、のデータを書きます。RAM内の最初のデータ位置はYレシ、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レシ、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領域側のコート、だけが自己プログラミング(ペーシ、消去とペーシ、書き込み)中に読めます。使用レシ、スタはRO,R1,TMP,CNTL,CNTH,SPMCで、レシ、スタの保存と復帰はこのルーチン内に含まれず、使用レシ、スタはコート・量を犠牲にすれば最適化できます。割り込み表がブートローダ、領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイル以下の場合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)。

ラヘ゛ル	命令		注釈
	. EQU . ORG	PGSZB = PAGESIZE*2 SMALLBOOTSTART	;PGSZBはページ内のバイト数です。(PAGESIZEは語(ワード)数) ;
WRPG:	LDI RCALL	SPMC, (1< <pgers)+(1<<spmen) spmj<="" td=""><td>; [^゚ーシ´消去] ; ^゚ーシ´消去SPMCR値を取得 ; ^゚ーシ´消去 ; [RWW領域読み出し再許可]</td></pgers)+(1<<spmen)>	; [^゚ーシ´消去] ; ^゚ーシ´消去SPMCR値を取得 ; ^゚ーシ´消去 ; [RWW領域読み出し再許可]
	LDI RCALL	SPMC, (1< <rwwsre)+(1<<spmen) spmj<="" td=""><td>;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 ヘーン 一時緩衝部へ転送]</td></rwwsre)+(1<<spmen)>	;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 ヘーン 一時緩衝部へ転送]
WLP:	LDI LDI LD LD	CNTL, LOW (PGSZB) CNTH, HIGH (PGSZB) RO, Y+ R1, Y+	; バイト計数器を初期化 ; <mark>(削除)</mark> ; 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)
	LDI RCALL ADIW SBIW BRNE	SPMC, (1< <spmen) cnth:cntl,2="" spmj="" td="" wlp<="" zh:zl,2=""><td>; 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</td></spmen)>	; 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
	SUBI SBCI LDI RCALL	ZL, LOW (PGSZB) ZH, HIGH (PGSZB) SPMC, (1< <pgwrt)+(1<<spmen) spmj<="" td=""><td>;フラッシュメモリヘ゜ーシ゛書き込み</td></pgwrt)+(1<<spmen)>	;フラッシュメモリヘ゜ーシ゛書き込み
	LDI RCALL	SPMC, (1< <rwwsre)+(1<<spmen) SPMJ</rwwsre)+(1<<spmen) 	;RWW領域読み出し許可
	LDI LDI SUBI SBCI	CNTL, LOW (PGSZB) CNTH, HIGH (PGSZB) YL, LOW (PGSZB) YH, HIGH (PGSZB)	; [読み戻し照合 (任意)] ; ハ 小計数器を初期化 ; <mark>(削除)</mark> ; RAMデータ先頭にホペンタを復帰 ;
RLP:	LPM LD CPSE RJMP	RO, Z+ R1, Y+ RO, R1 ERROR	; フラッシュ メモリから1バ・仆取得(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 BRNE	CNTH:CNTL, 1 RLP	;計数器を減数 <mark>(SUBI)</mark> ;指定バ仆数分継続 ;[RWW領域へ復帰]
RTN:	IN SBRS RET	TMP, SPMCR TMP, RWWSB	;SPM命令制御レシブスタ値を取得 ;RWW領域多忙でスキップ ;準備可で呼び出し元へ復帰
;	LDI RCALL RJMP	SPMC, (1< <rwwsre)+(1<<spmen) SPMJ RTN</rwwsre)+(1<<spmen) 	;[RWW領域読み出し再許可] ;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 SBRC RJMP	TMP, SPMCR TMP, SPMEN SPMJ	,[SFM即 元 美1] リノルーリン] ; SPM命 令制御レシ スタ値を取得 ; 操作可能(直前のSPM完了)でスキップ ; 操作可まで待機
WAIT:	IN CLI SBIC RJMP	TMP, SREG EECR, EEWE WAIT	; ステータス レジ [・] スタ値を保存 ; 全割り込み禁止 ; EEPROM書き込み中以外でスキップ。 ; EEPROM書き込み完了まで待機
;	OUT SPM OUT RET	SPMCR, SPMC SREG, TMP	; 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



ATmega8用ブートローターのハプラメータ

自己プログラミングの記述で使ったパラメータは表82.~84.で与えられます。

表82. 応用領域とブートローダ領域の分割設定

	X-1 TO THE VIEW TO						
ı	BOOTSZ1	BOOTSZ0	ブートローダ領域		アドレス範囲		
	B001321		容量 (語)	ページ数	応用プログラム領域	ブートローダプログラム領域	ブート リセット アドレス
	1	1	128	4	\$0000~\$0F7F	\$0F80∼\$0FFF	\$0F80
	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ヒュース 設定は図102.で示されます。

表83. RWW領域とNRWW領域の範囲

領域	ページ数	アト・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	

これら2つの領域についての詳細に関しては136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84 図103で使った各変数説明とフポインタの配置

衣64. 図103. C 使りに音を数説明C2小123の配直				
変数名	対応値		意味	
多数 石	PC	Zポインタ (注)	忠怀	
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~゚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。 (Zo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。 (Zo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 ヘーシ゛アト・レス:ヘーシ゛消去とヘーシ゛書き込み用のヘーシ゛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O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ポインタの使用についての詳細に関しては139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 プログラミンク゛

プログラム メモリとデータ メモリ用施錠ビット

ATmega8は非プログラム(1)のままか、表86.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85. 施錠ビット バイの内容

名称	ピット番号	意味	既定値(注)
_	7		1 (非プログラム)
_	6		1 (非プログラム)
BLB12	5	フ゛ートロータ゛領域に対する保護用フ゛ート施錠ビット	1 (非プログラム)
BLB11	4	/ 一トローク映域に対する体護用/一ト地鉄にタト	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	心用限域に対する休護用/『心臓にが	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	フラッシュとEEPROMPモリ(CX) りる一放休護用地爽にりト	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86. 施錠ビットの保護種別

	メモリ施錠ビット	ני <i>ת</i>	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。 ヒュース ' ビットが固定されます。 (<mark>注1</mark>)
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。ヒュース・ヒットとブート施錠ビットが固定されます。(<mark>注1</mark>)
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命令がブートロータ、領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブート ローダ領域に書くことを許されません。
3	0	0	SPM命令によるプ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タ、領域からの読み込みが許されません。(<mark>注3</mark>)

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。

識票バイ

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。ATmega8の識票バイトを右に示します。

\$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:\$07 ②値\$93と合せ、ATmega8を示します。

校正バイ

ATmega8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。リセット中に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19頁の「OSCCAL - 発振校正レジスタ」をご覧ください。



ヒュース゛ヒ゛ット

ATmega8には2つのヒュース、バイトがあります。表87.と表88.は全てのヒュース、の概略機能とヒュース、バイト内でどのように配置されるかを示します。ヒュース、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87. ヒューズ上位バイト一覧

名称	ピット	意味	既定値
RSTDISBL(注4)	7	PC6がI/OピンかまたはRESETピンかを選びます。	1 (非プログラム) PC6はRESETピン
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WDTCRで許可
SPIEN (注1)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
CKOPT (注2)	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	プートロール 公長 湿和 (主00 会昭) (注0)	<mark>0</mark> (プ [°] ログ [*] ラム)
BOOTSZ0	1	ブートローダ 容量選択。(表82 .参照) (注3)	<mark>0</mark> (プ [°] ログ [*] ラム)
BOOTRST	0	リセット ベクタ(応用領域またはブート ローダ領域)選択。	1 (非プログラム) 応用領域

注1: SPIENヒュース は直列プログラミングでアクセスできません。

注2: CKOPTヒュースでの機能はCKSELヒュース、設定に依存します。詳細については16頁の「クロック元」をご覧ください。

注3: BOOTSZ1,0の既定値は最大ブートローダ容量になります。143頁の表82.をご覧ください。

注4: RSTDISBLヒューズをプログラム(0)すると、ヒューズ変更または更なるプログラミングを実行するのに並列プログラミングが使われなければなりません。

表88. ヒューズ下位バイト一覧

名称	ピット	意味	既定值
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	也到时间进 <u>价。</u> (注1)	<mark>0</mark> (プ [°] ロク [*] ラム)
CKSEL3	3		<mark>0</mark> (プ [°] ロク [*] ラム)
CKSEL2	2	クロック種別選択。 (<u>注</u> 2)	<mark>0</mark> (プ [°] ロク [*] ラム)
CKSEL1	1	クロック作品が、 (注2)	<mark>0</mark> (プ [°] ロク [°] 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18頁の表10.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16頁の表2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。

ヒュース、のラッチ

ヒュース^{*}値はデ[゙]ハ^{*}イスがプ[°]ログ[†]ラミング^{*}動作へ移行する時にラッチされ、ヒュース^{*}値の変更はデ[゙]ハ^{*}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 ヒュース^{*}には適用されません。ヒュース^{*}は通常動作での電源投入でもラッチされます。

ページ容量

表89. フラッシュ メモリのヘージ 数とヘージ の語数

200.7777-7-07-0	· 20 ·	** ## >><			
全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
4K語 (8Kバイト)	32語	PC4∼0	128	PC11~5	11

表90. EEPROMメモリのヘ゜ーシ゛数とヘ゜ーシ゛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

この章はATmega8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、ヒュース・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

本章でATmega8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104.と表91.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93.で示されます。

 $\overline{\text{WR}}$ または $\overline{\text{OE}}$ パルス送出時、設定された指令が決める動作が実行されます。各種指令は $\mathbf{\overline{k}94}$.で示されます。

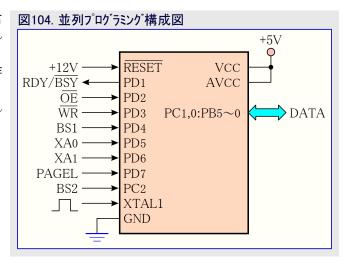


表91. 信号名とピン名の関係

信号名	ピン名	入出力			
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)		
ŌĒ	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データをヘーシー 一時緩衝部に設定		
BS2	PC2	入力	上位/下位バイ・選択2 (0:下位, 1:上位) (ヒュース・ビット用)		
DATA	PC1,0:PB5∼0	入出力	双方向データ バス (OE=Low時出力)		

表92. プログラミング動作移行時のピン値

ピン名	シンホ゛ル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

表93. 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94. 指令バイのビット符号化

指令バイト	指令の機能
\$80 (1000 0000)	チップ消去
\$40 (0100 0000)	ヒューズビット書き込み
\$20 (0010 0000)	施錠ビット書き込み
\$10 (0001 0000)	フラッシュメモリ書き込み
\$11 (0001 0001)	EEPROM書き込み
\$08 (0000 1000)	識票バイト、校正バイト読み出し
\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$02 (0000 0010)	フラッシュ メモリ読み出し
\$03 (0000 0011)	EEPROM読み出し



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。
- ③ 146頁の表92.で一覧されるProg_enablet°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_enablet°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: RESET ピンがRSTDISBLヒューズのプログラム(0)によって禁止されていると、上記で提唱された方法に従うことができなくなってしまいます。外部クリスタルや外部RC発振器設定が選ばれていると、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。

- ① 146頁の表92.で一覧されるProg_enablet°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~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ばれ(CKSEL3~0=0000)、RESET ピンが活性(有効)にされる(RSTDISBL=非プログラム(1))ことを保証するためにヒュースで再プログラミング(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、ヒュースで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、RESETt゚ンをLow(0)に持ってくることによってプログラミング動作を抜けます。
- **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ヒュース゛がプログラム(0)されている場合を除き)EEPROM全体の内容は\$FFなので、値が\$FFの データ書き込みを飛ばします。
- アドレス上位バイトはフラッシュ メモリで新規256語(ワード)枠、EEPROMで新規256バイト枠の読み書き前に一度だけ必要です。この考慮は識票バイト読み出しにも適用されます。

チップ消去

チップ消去はフラッシュ メモリ、EEPROM(<mark>注1</mark>)、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106.タイミングを参照)

フラッシュ メモリはヘーシ で構成されます(145頁の表89.参照)。フラッシュ メモリへ書く時にプログラム データはヘージ 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ヘージ に許します。 次の手順は完全なフラッシュ メモリの書き込み方法を記述します。

A. 「フラッシュ メモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュ メモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュ メモリ書き込み指令を設定します。

B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選びます。
- ③ DATAにアトレス下位ハイト(\$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にデータ下位ハイト(\$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。これは上位バイを選びます。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位ハイト(\$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(ワート)データをヘージー時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイを選びます。
- ② PAGELに正パルスを与えます。これは語データをページー時緩衝部にラッチ(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を繰り返し

アト・レス内の下位ビットがヘーシ・内の語(ワート)位置を指示する一方、上位ビットがフラッシュ メモリ内のヘージをアト・レス指定します。これは**図105**.で図示されます。 ヘージ・内の語アト・レスに8ビット未満が必要とされる場合(ヘージ・容量く256)、アト・レス下位ハーイの最上位(側)ビットがヘージ書き込み実行時のヘージ・アト・レスに使われることに注意してください。

G. アドレス上位バイト設定

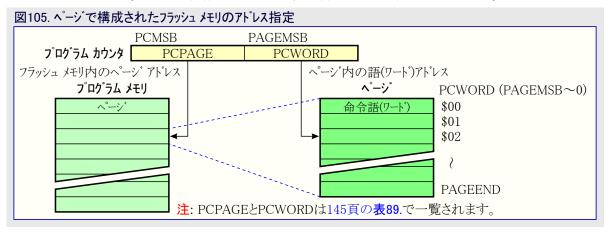
- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選びます。
- ③ DATAにアトレス上位ハイト(\$00~\$0F)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイを設定します。

H. ページ書き込み

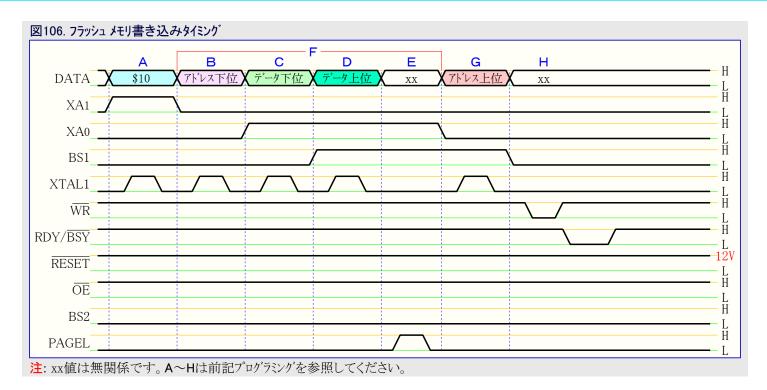
- ① BS1をLow(0)に設定します。
- ② WRに負 $^{\circ}$ ルスを与えます。これはデータの $^{\circ}$ ーシ'全体の書き込みを開始します。RDY/ $\overline{\text{BSY}}$ がLow(0)になります。
- ③ RDY/BSYがHigh(1)になるまで待ちます。
- I. フラッシュ メモリ全部が書かれるか、または(必要な)全データが書かれてしまうまで、B~Hを繰り返し

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に正パパスを与えます。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。



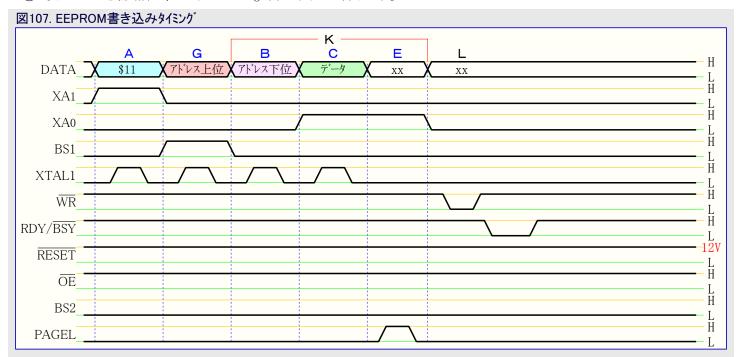




EEPROM書き込み

EEPROMはヘーシで構成されます(145頁の表90.参照)。EEPROMを書く時にデータはヘーシ、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ヘーシに許します。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、アトレス、データ設定の詳細については148頁の「フラッシュ メモリの書き込み」を参照。図107.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. アトンス上位ハイト(\$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. アトンス下位バイト(\$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 次のヘージを書く前に、RDY/BSYがHigh(1)になるまで待ちます。





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148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. アドレス上位ハイト(\$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. アドレス下位ハーイ(\$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(ワート)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(ワート)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148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. アト・レス上位ハ・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<u>(</u>\$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

ヒュース、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148頁の「フラッシュ メモリの書き込み」を参照)

- 1. ヒュース 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

ヒュース、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148頁の「フラッシュ メモリの書き込み」を参照)

- 1. ヒュース^{*} 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. データ下位ハイト(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を選びます。

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148頁の「フラッシュ メモリの書き込み」を参照)

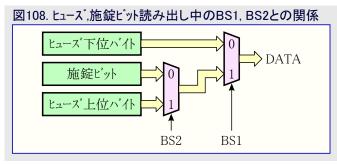
- 1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュ メモリ書き込み」のAを参照)
- 2. データ下位ハーイト(としてデータ)を設定します。0=プログラム,1=無変化です。(「フラッシュメモリ書き込み」のCを参照)
- 3. WRに負パルスを与え、RDY/BSYがHighになるまで待ちます。

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

ヒュース、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148頁の「フラッシュ メモリの書き込み」を参照)

- 1. ヒュース` 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)、OEをLow(0)に設定します。ヒュース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)、OEをLow(0)に設定します。ヒュース、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)、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. OEをHigh(1)に設定します。DATAはHi-Zになります。





識票バイ読み出し

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148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. アドレス下位ハイト(\$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)、○EをLow(0)に設定します。選んだ識票バイが直ぐにDATAで読めます。
- 4. OEをHigh(1)に設定します。DATAはHi-Zになります。

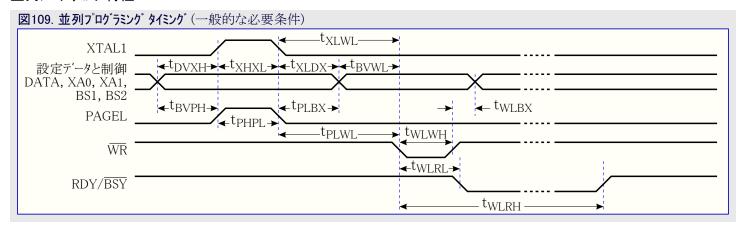
校正バイ読み出し

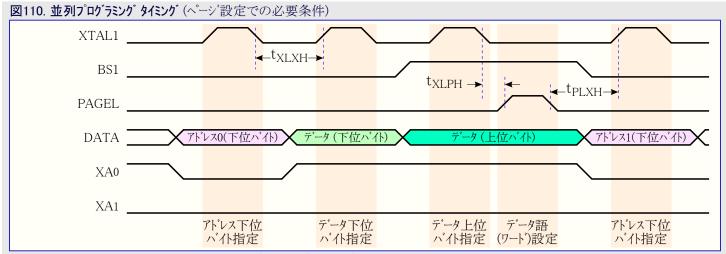
校正バイの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については148頁の「フラッシュ メモリの書き込み」を参照)

- 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. アドレス下位ハーイトに(\$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)
- 3. BS1をHigh(1)、OEをLow(0)に設定します。校正バイが直ぐにDATAで読めます。
- 4. OEをHigh(1)に設定します。DATAはHi-Zになります。

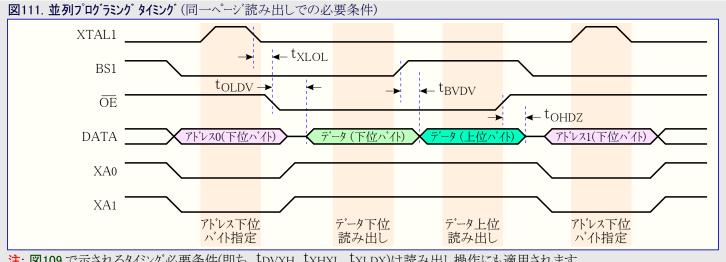


並列プログラミング特性





注: 図109.で示されるタイシンが必要条件(即ち、tDVXH、tXHXL、tXLDX)は設定操作にも適用されます。



注: 図109.で示されるタイシケンが要条件(即ち、^tDVXH、^tXHXL、^tXLDX)は読み出し操作にも適用されます。

表95. 並列プログラミング特性 (VCC=5V±10%)

シンホ゛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
t _{DVXH}	XTAL1↑に対するデータと制御の準備時間	67			
txLxH	XTAL1↓から次XTAL1↑までの待機時間	200			
txHXL	XTAL1 Highパルス幅	150			
t _{XLDX}	XTAL1パルス↓後のデータと制御の保持時間	67			
txLWL	XTAL1パハンス↓後のWR↓待機時間	0			
t _{XLPH}	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			ns
t _{PHPL}	PAGEL Highパルス幅	150			
t _{PLBX}	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			
t _{PLWL}	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	
twlrh_ce	チップ。消去時間 (WR↓からRDY/BSY↑) (注2)	7.5		9	ms
txLCL	XTAL1パルス↓後のOE↓待機時間	0			
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	ns
toHDZ	 OE↑後のDATA Hi-Z遅延時間			250	

注1: フラッシュ メモリ、EEPROM、ヒュース゛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ハ、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96.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。

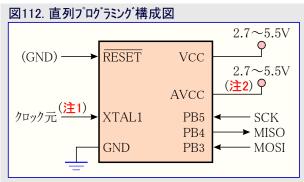
直列プログラミング用ピン配置

主の6	古列-	°ロカ゛ニこ	いが田	ピン配置
বছ গ্ৰত.	日 クリュ	/ Hソ フミ	ノソ HI	

信号名	ピン名	入出力	機能
MOSI	PB3	入力	直列データ入力
MISO	PB4	出力	直列データ出力
SCK	PB5	入力	直列クロック

EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(直列プログラミングのみ)、チップ消去命令を初めに実行する必要がありません。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。



注1: デバイスが内蔵発振器で動作する場合、XTAL1 ピンにクロック元を接続する必要はありません。

注2: VCC-0.3V〈AVCC〈VCC+0.3Vですが、AVCCは常に2.7~5.5V内にすべきです。

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

直列プログラミング手順

ATmega8に直列データを書く時にデータはSCKの上昇端で行われ、ATmega8から読む時にデータはSCKの下降端で行われます。タイシグの詳細については図113.をご覧ください。

直列プログラジング動作でのATmega8のプログラジングと照合は次の手順が推奨されます(4小小命令形式は表98.をご覧ください)。

1. 電源投入手順:

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

- 2. 最低20ms待ち、MOSIピンにプログラミング許可命令を送ることによって直列プログラミングを許可してください。
- 3. 通信が同期を外していると、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パイト送出時に第2 パイト(\$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パ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ペーシ 単位で書かれます。ページ 容量は145頁の表89.で得られます。このメモリペーシ はペーシ 設定命令と共にアドレスの下位5+1ビットとデータを供給することによって1パイトづつ設定されます。ページ が正しく設定されるのを保証するため、与えられたアドレスにデータ上位パイトが適用される前にデータ下位パイトが設定されなければなりません。(プログラム メモリ)のページ はアドレスの上位7ビットを含むページ 書き込み命令の設定によって(フラッシュ メモリに)格納されます。ポーリングが使われない場合、使用者は次のページを行う前に最低twp FLASH(表97.参照)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ポーリング(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。
- 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供給することによって1バイト単位で書かれます。EEPROMのメモリ 位置は新規データが書かれるのに先立て始めで自動的に消去されます。ポーリングが使われない場合、次のバイトを行う前に最低 twp_EEPROM(表97.参照)待たなければなりません。チップ消去されたデ゙バイスでの\$FFデータは書かれる必要がありません。
- 6. どのメモリ位置も選んだアドレスの内容を直列出力MISOに読み戻す読み出し命令を使うことによって照合できます。
- 7. プログラミング作業終了時、RESETは通常動作を開始するため、High(1)に設定できます。
- 8. 電源OFF手順(必要とされるならば)
 - ・ RESET をHigh(1)に設定します。
 - VCC電源をOFFにします。



フラッシュ メモリの データ ホーリング

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<mark>\$FF</mark>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twp 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97**.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<mark>\$FF</mark> 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 ${}^{
m FF}$ に対して使えず、次バイト書き込み前に最低t ${}^{
m two}$ _EEPROM待たなければなりません。 ${}^{
m two}$ _ EEPROM値については**表97**.をご 覧ください。

表97. 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 ヒューズ書き込み 4.5ms two fuse 4.5ms フラッシュメモリ書き込み twd flash twd_eeprom 9.0ms EEPROM書き込み チップ消去 9.0ms two erase

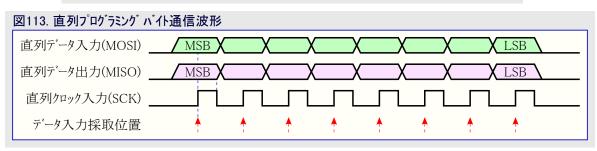


表98. 直列プログラミング命令一式

命令		命令	形式	動作			
中中	第1バイト	第2バイト	第3バイト	第4バイト	到15		
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	i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 НННН	LLLL LLLL	RRRR RRRR	アトレスH:LのP(H/L)バイトを	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)/	、・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	
フラッシュヘージ書き込み	0100 1100	0000 НННН	LLLx xxxx	XXXX XXXX	アドレスH:Lのヘ゜ーシ゛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	(144頁の表85.参照)	に書き込みます。	
ヒュース・下位読み出し	0101 0000	0000 0000	XXXX XXXX	RRRR RRRR	ヒュース・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	(145頁の 表88 .参照)	に書き込みます。	
ヒュース・上位読み出し	0101 1000	0000 1000	XXXX XXXX	RRRR RRRR	ヒュース・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	(145頁の 表87 .参照)	に書き込みます。	
識票バイト読み出し	0011 0000	00xx xxxx	xxxx xxLL	RRRR RRRR	アドレスLの識票バイトを読み出します。		
校正バ小読み出し	0011 1000	00xx xxxx	0000 00LL	RRRR RRRR	アドレスLの校正バイトを読み	み出します。	

H=アドレス上位バイトのビット

 $L = アト i \nu$ ス下位 γ i かん $P = 0 = \Gamma$ 位 γ i かん γ i を γ i γ

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)

直列プログラミング特性

SPI部の特性については159頁の「SPIタイミング特性」をご覧ください。



電気的特性 - (TA=-40°C~85°C)

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

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
RESETt°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 ····· 40.0 mA
消費電流 ······ 200.0 mA

(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°C~85°C, VCC=2.7V~5.5V (特記事項を除く)

シンホ゛ル	項目		条件	最小	代表	最大	単位
VIL	Lowレヘ・ル入力電圧 (一般/XTALポート)		ポートB,C,DとI/O使用のXTAL1,XTAL2	-0.5		0.2VCC (注1)	
VIL1	Lowレヘブル入力電圧 (XTAL1)		外部クロック信号選択	-0.5		0.1VCC (注1)	
VIL2	Lowレベル入力電圧 (RESI	ET)	外部リセット入力	-0.5		0.2VCC (注1)	
V _{IL} 3	Lowレベル入力電圧 (RESI	ET)	I/Oとして使用	-0.5		0.2VCC (注1)	
VIH	Highレベル入力電圧 (一般/XTALポート)		ポートB,C,DとI/O使用 のXTAL1,XTAL2	0.6VCC (注2)		VCC+0.5	T 7
V _I H ₁	Highレベル入力電圧 (XTA	AL1)	外部クロック信号選択	0.8VCC (注2)		VCC+0.5	V
V _{IH2}	Highレベル入力電圧 (RES	ET)	外部リセット入力	0.9VCC (注2)		VCC+0.5	
V _{IH3}	Highレベル入力電圧 (RES	ET)	I/Oとして使用	0.8VCC (注2)		VCC+0.5	
Vol	Lレベル出力電圧	(注3)	IOL=20mA, VCC=5V			0.9	
VOL	(ポートB,C,D)		IOL=10mA, VCC=3V			0.6	
Vон	Hレヘ・ル出力電圧 (注4)		IOH=-20mA, VCC=5V	4.2			
	(# -\B,C,D)		IOH=-10mA, VCC=3V	2.2			
IIL	I/OビンLowレベル入力漏れ電流		VCC=5.5V (絶対値)			1	μA
IIH	I/OビンHighレヘブル入力漏れ	1電流	確実なH/L範囲			1	pa 1
RRST	RESETピン プルアップ抵抗			30		80	kΩ
Rpu	I/Oピン プルアップ抵抗			20		50	1100
	活動動作消費電流		VCC=3V, 4MHz		3	5	
	旧数数件 们负电师	ATmega8	VCC=5V, 8MHz		11	15	mA
ICC	アイル動作消費電流	ATmega8L			1	2	1111 1
100		ATmega8	VCC=5V, 8MHz		4.5	7	
	パワーダウン動作消費電流	(注5)	VCC=3V, WDT許可		<22	28	μA
			VCC=3V, WDT禁止		<1	3	μΛ
	アナログ比較器入力変位(オフセット)電圧		VCC=5V, Vin=VCC/2			40	mV
IACLK	アナログ比較器入力漏れ電	流	v CC -5 v , v III – v CC / 2	-50		50	nA
tacpp	アナログ比較器伝播遅延時	調	VCC=2.7V		750		ns
ACPD	/ / - / 四秋仙山田	l H1	VCC=5.0V		500		113

注1: Low いかい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300mAを超えるべきではありません。
 - 2. ポートC0~C5のIOLの合計が100mAを超えるべきではありません。
 - 3. ポートB0~B7、C6、D0~D7とXTAL2のIOLの合計が200mA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300mAを超えるべきではありません。
 - 2. ポートC0~C5のIOHの合計が100mAを超えるべきではありません。
 - 3. ポートB0~B7、C6、D0~D7とXTAL2のIOHの合計が200mA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

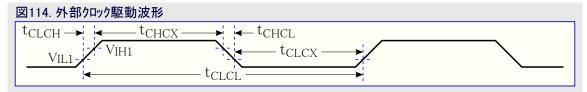


表99. 外部クロック特性

シンホ゛ル		VCC=2.	7 ~ 5.5V	VCC=4.5∼5.5V		単位	
シンホ ル	坦	最小	最大	最小	最大	単位	
1/tclcl	クロック周波数	0	8	0	16	MHz	
tclcl	クロック周期	125		62.5			
tchcx	Highレヘール時間	50		25		ns	
tclcx	Lowレヘブル時間	50		25			
tclch	上昇時間		1.6		0.5	110	
tchcl	下降時間		1.6		0.5	μs	
⊿tclcl	隣接クロック周期間の変化率		2		2	%	

注: 詳細については19頁の「外部クロック信号」を参照してください。

表100. 代表的な外部RC発振周波数

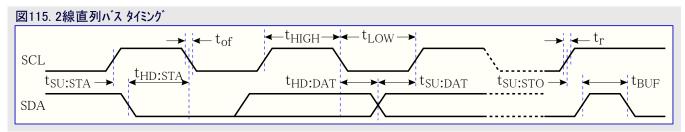
24111111	The second of the second property and the second se								
周波数 f	抵抗 R(kΩ)	容量 C (pF)							
650kHz	33	22							
2.0MHz	10	22							

注: Rの範囲は3~100k Ω 、Cは最小20pFとすべきです。表で与えられたCの値は \mathfrak{l}° ン容量を含みます。これは外囲器形状や基板配置で変化します。



2線直列インターフェース特性

表101.は2線直列バスに接続した装置に対する必要条件を記述します。ATmega8の2線直列インターフェースは記載条件下で、これらの必要条件を越えるか、または合致します。



耒1	101	9線店	石川イング	カーフェー	ス必要多	と仕
4X I	IUI.	乙形以旦	クリインフ	ァーノエー	へとい女ス	K

シンホ゛ル	項目	条件	最小	最大	単位
VIL	Lowレベル入力電圧		-0.5	0.3VCC	
VIH	Highレヘブル入力電圧		0.7VCC	VCC+0.5	V
Vhys 1	シュミットトリカ、入力ヒステリシス電圧		2 0.05 VCC		V
Vol 1	Lowレヘブル出力電圧	IOL=3mA	0	0.4	
t_r 1	出力上昇時間(V _{ILmin} →V _{IHmax})		2 20+0.1Cb 3	300	
tof 1	出力下降時間(V _{IHmin} →V _{ILmax})	10pF <cb<400pf 3<="" th=""><th>2 20+0.1Cb 3</th><th>250</th><th>ns</th></cb<400pf>	2 20+0.1Cb 3	250	ns
tsp 1	入力パルス最小幅(尖頭消去濾波)		0	2 50	
Ii	入力電流(ピン単位)	0.1VCC <vi<0.9vcc< th=""><th>-10</th><th>10</th><th>μA</th></vi<0.9vcc<>	-10	10	μA
Ci ①	ピン入力容量			10	pF
fSCL	SCLクロック周波数 ④⑤	f_{CK} max(16 f_{SCL} ,250kHz)	0	400	kHz
Rp	 プルアップ・抵抗値	$f_{\rm SCL} \leq 100 \text{kHz}$	(VCC-0.4V)/3mA	1000ns/Cb	Ω
Кр		$f_{\rm SCL} > 100 {\rm kHz}$	(VCC-0.4V)/3mA	300ns/Cb	22
tup cra	(再送)開始条件保持時間	$f_{\rm SCL} \leq 100 \text{kHz}$	4.0		
tHD:STA	(丹达)開始未件保持時间	$f_{\rm SCL} > 100 {\rm kHz}$	0.6		
t_{LOW}	SCLクロックLowレヘブル時間	$f_{\text{SCL}} \leq 100 \text{kHz}$ 6	4.7		
CLOW	SCL/ログ/LOWV・、/ル時 [日]	$f_{\rm SCL} > 100 \text{kHz}$	1.3		
t _{HIGH}	 SCLクロックHighレベル時間	f _{SCL} ≤100kHz	4.0		μs
чнісн	SCL/Py/Tigity Trugia	$f_{\rm SCL} > 100 {\rm kHz}$	0.6		μδ
tsu:sta	 再送開始条件準備時間	$f_{\rm SCL} \leq 100 \text{kHz}$	4.7		
ν50:51A	开 及两州未任平州时间	$f_{\rm SCL} > 100 {\rm kHz}$	0.6		
t _{HD:DAT}	プータ保持時間	f _{SCL} ≤100kHz	0	3.45	
THD:DAT	7 7 体1分时间	$f_{\rm SCL} > 100 {\rm kHz}$	0	0.9	
tsu:DAT	」 データ準備時間	f _{SCL} ≤100kHz	250		ne
tsu:DAT	/ / 1 \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ 	$f_{\rm SCL} > 100 {\rm kHz}$	100		ns
t _{SU:STO}	停止条件準備時間	f _{SCL} ≤100kHz	4.0		
450:510	11.4.11.4.11.11.11.11.11.11.11.11.11.11.	$f_{\rm SCL} > 100 {\rm kHz}$	0.6		μs
t _{BUF}	 停止条件→開始条件間バス開放時間	f _{SCL} ≤100kHz	4.7		μδ
rBOH		$f_{\rm SCL}$ $>$ 100kHz	1.3		

- ① ATmega8で、この項目は特性が記載されていますが、100%検査はされていません。
- ② $f_{SCL} > 100 kHz$ についてのみ必要とされます。
- ③ Cbは1つのバス信号線の容量(pF)です。
- **4** f_{CK} はCPU(システム)クロック周波数です。
- ⑤ この必要条件はATmega8の全ての2線直列インターフェース動作に適用します。2線直列バスに接続した他の装置は一般的なfscl 必要条件に従うことだけを必要とします。
- ⑥ ATmega8の2線直列インターフェースによって生成した実際のLow区間は $(1/f_{SCL}-2/f_{CK})$ で、従ってLow時間の必要条件に対して $f_{SCL}=100$ kHzで厳密に満たされるには、 f_{CK} が(概ね)6MHz以上でなければなりません。
- ⑦ ATmega8の2線直列インターフェースによって生成した実際のLow区間は(1/f_{SCL}-2/f_{CK})で、従ってf_{CK}=8MHz時、厳密にはf_{SCL}> 308kHzでLow時間の必要条件が満たされません。それにも拘らず、バスに接続されたATmega8装置は相応なt_{LOW}許容余地のある他の装置だけでなく他のATmega8装置と最高速(400kHz)で通信できるでしょう。



SPIタイミング特性

図116. SPI タイミング必要条件(主装置動作) SS (従装置選択) SCK (CPOL=0) SCK (CPOL=1) MISO (データ入力) -MOSI (データ出力) **MSB**

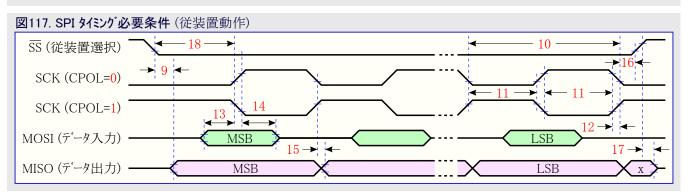


表102. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表50.参照		
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		ng
10	SCK周期	従装置	4tcK			ns
11	SCK High/Low期間 (注)	従装置	2tck			
12	SCK上昇/下降時間	従装置			1600	
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、2tCLCL(tCK<12MHz)、3tCLCL(tCK≥12MHz)です。



A/D変換器特性

表103. A/D変換特性

シンホ゛ル	項目		条件	最小	代表	最大	単位
	分解能				10		ピット
	絶対精度 (積分非直線性誤差、		変換クロック=200kHz		1.75		
	微分非直線性誤差、 量子化誤差、利得誤差、 変位(オフセット)誤差を含む)	VCC=4V	変換クロック=1MHz		3		LSB
	積分非直線性誤差	VREF=4V			0.75		Lob
	微分非直線性誤差		変換クロック=200kHz		0.5		
	利得誤差		<u>多</u> (契/19/7-200KF1Z		1		
	変位(オフセット)(セ゚ロ)誤差				1		
	変換時間 (注3)	連続変換動	作	13		260	μs
	変換クロック周波数			50		1000	kHz
AVCC	アナログ供給電圧			VCC-0.3(注1)		VCC+0.3(注2)	
Vref	基準電圧			2.0		AVCC	V
VIN	入力電圧			GND		VREF	
	入力周波数帯域				38.5		kHz
VINT	内蔵2.56V基準電圧			2.3	2.56	2.9	V
RREF	基準電圧入力インピーダンス				32		kΩ
RAIN	アナログ入力インピーダンス			55	100		ΜΩ

注: これらの値は目安です。 **注1**: AVCCの最小値は2.7Vです。 **注2**: AVCCの最大値は5.5Vです。

注3: 最大変換時間は1/50kHz×25=0.5msです。



電気的特性 - (TA=-40°C~105°C)

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

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
RESETt°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 ・・・・・・・・・・・ 40.0 mA
消費電流 ······ 200.0 mA

(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℃~105℃, VCC=2.7V~5.5V (特記事項を除く)

	$C \sim 105 \text{C}, \text{ VCC} = 2.7 \text{V} \sim 5$.01 (小 町 中		□ .1.	/v. ±	日上	꽈 /ㅗ
シンホ゛ル			条件	最小	代表	最大	単位
VIL	Lowレベル入力電圧		XTAL1ピンを除く	-0.5		0.2VCC (注1)	
VIL1	Lowレベル入力電圧 (XTAL1)		外部クロック信号選択	-0.5		0.1VCC (注1)	
VIH	Highレベル入力電圧		XTAL1,RESETt°ンを除く	0.6VCC (注2)		VCC+0.5	
V _I H ₁	Highレベル入力電圧 (XTA	L1)	外部クロック信号選択	0.8VCC (注2)		VCC+0.5	
V _I H ₂	Highレベル入力電圧 (RES	ET)		0.9VCC (注2)		VCC+0.5	V
Vol	Lレベル出力電圧	(注3)	IOL=20mA, VCC=5V			0.9	
VOL	(ポートB,C,D)	(/土3)	IOL=10mA, VCC=3V			0.6	
Voh	Hレベル出力電圧	(注4)	IOH=-20mA, VCC=5V	4.2			
VOH	(赤°-トB,C,D)		IOH=-10mA, VCC=3V	2.2			
IIL	I/OビンLowレベル入力漏れ電流		VCC=5.5V (絶対値)			3	11 Λ
IIH	I/OビンHighレベル入力漏オ	1電流	確実なH/L範囲			3	μA
RRST	RESETピン プルアップ抵抗			30		80	kΩ
Rpu	I/Oピン プルアップ抵抗			20		50	K 12
	活動動作消費電流	ATmega8L	VCC=3V, 4MHz			6	
	伯别别作仍复电机	ATmega8	VCC=5V, 8MHz			15	A
ICC	アイドル動作消費電流	ATmega8L	VCC=3V, 4MHz			3	mA
ICC	74179111日有电机	ATmega8	VCC=5V, 8MHz			8	
	、°D. 心心, 新龙沙弗泰达	() 上()	VCC=3V, WDT許可			35	A
	パワーダウン動作消費電流 (注5)		VCC=3V, WDT禁止			66	μA
VACIO	アナログ比較器入力変位(オフセット)電圧		VCC=5V, Vin=VCC/2			20	mV
IACLK	アナログ比較器入力漏れ電	流	VCC-5V, VIII-VCC/2	-50		50	nA
t a capa	アナログ比較器伝播遅延時	: EE	VCC=2.7V		750		ne
ACPD	/ / 四 儿牧命仏猫迷些时	^[HJ	VCC=5.0V		500		ns

注1: Lowレヘールの認識が保証される最高電圧です。

注2: Highレベルの認識が保証される最低電圧です。

注3: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=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。

PDIP 1. 全ポートのIOLの合計が400mAを超えるべきではありません。

2. ポートC0~C5のIOLの合計が200mAを超えるべきではありません。

3. ポートB0~B7、C6、D0~D7とXTAL2のIOLの合計が100mAを超えるべきではありません。

TQFP/MLF 1. 全ポートのIOLの合計が400mAを超えるべきではありません。

2. ポートC0~C5のIOLの合計が200mAを超えるべきではありません。

3. ポートC6、D0~D4のIOLの合計が300mAを超えるべきではありません。

4. ポートB0~B7、D5~D7のIOLの合計が300mA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)~(注5)は次頁を参照してください。



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。

PDIP 1. 全ポートのIOHの合計が400mAを超えるべきではありません。

2. ポートCo~C5のIOHの合計が100mAを超えるべきではありません。

3. ポートB0~B7、C6、D0~D7とXTAL2のIOHの合計が100mAを超えるべきではありません。

TQFP/MLF 1. 全ポートのIOHの合計が400mAを超えるべきではありません。

2. ポートCo~C5のIOHの合計が200mAを超えるべきではありません。

3. ポートC6、D0~D4のIOHの合計が300mAを超えるべきではありません。

4. ポートB0~B7、D5~D7のIOHの合計が300mA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。



代表特性 - (TA=-40°C~85°C)

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。

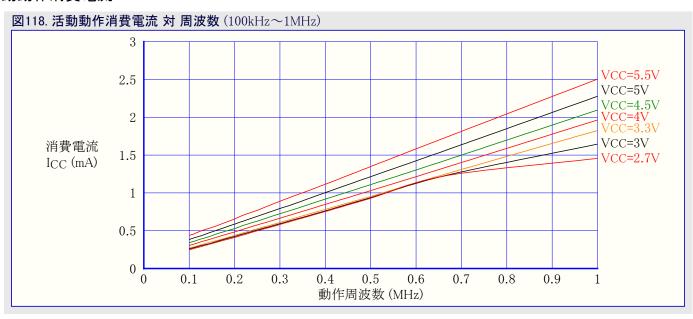
消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。 支配的な要素は動作電圧と動作周波数です。

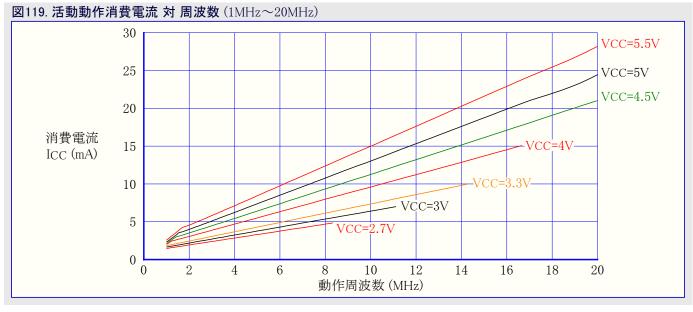
容量性負荷の \mathfrak{l}° ンの引き込み電流は(1つの \mathfrak{l}° ンに対して) C_L (負荷容量)× V_{CC} (動作電圧)×f($I/O\mathfrak{l}^{\circ}$ ンの平均切り替え周波数) として推測できます。

ディーイスは検査範囲よりも高い周波数特性を示します。ディー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。

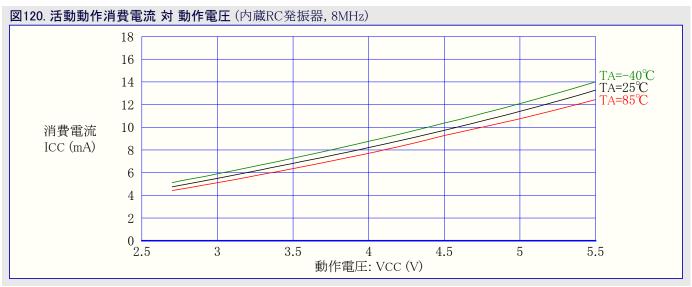
ウォッチト、ック、タイマ許可のハ°ワータ、ウン動作での消費電流とウォッチト、ック、タイマ禁止のハ°ワータ、ウン動作での消費電流間の違いは、ウォッチト、ック、タイマによって引き込んだ(消費した)差電流を表します。

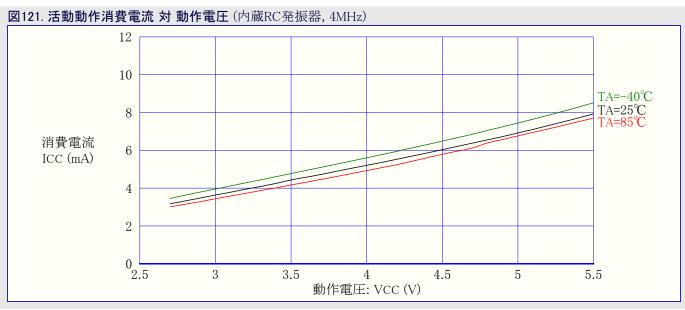
活動動作消費電流





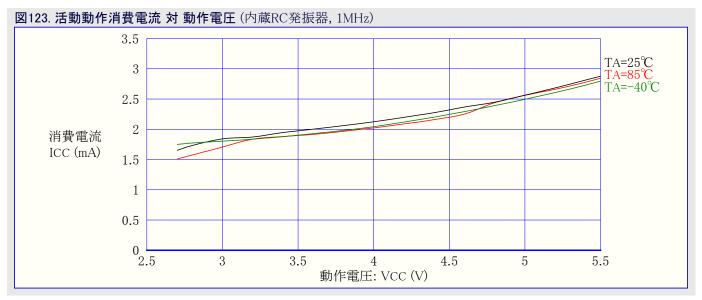






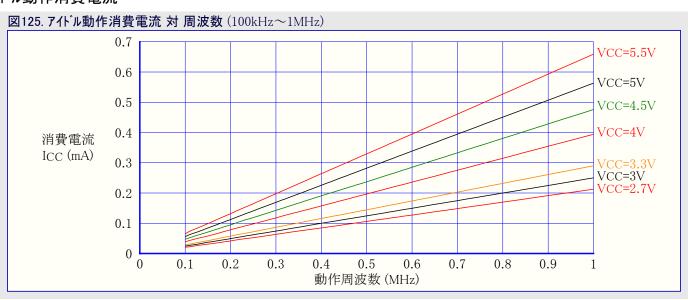




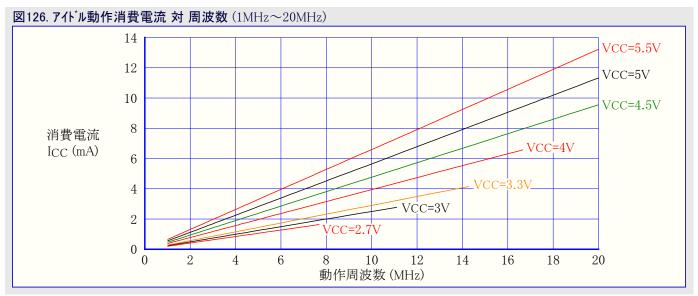




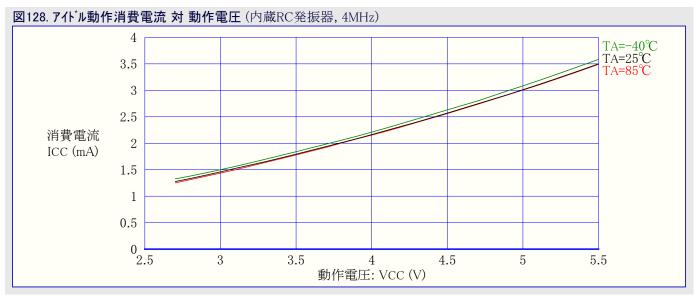
アイドル動作消費電流



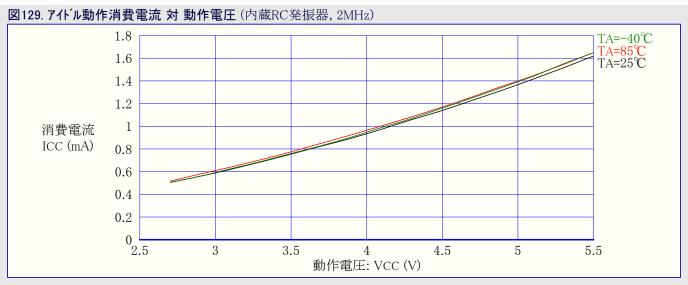


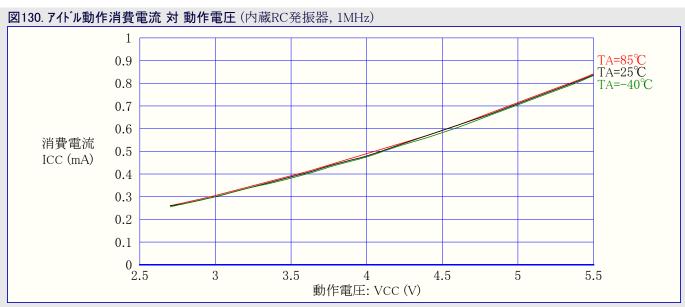








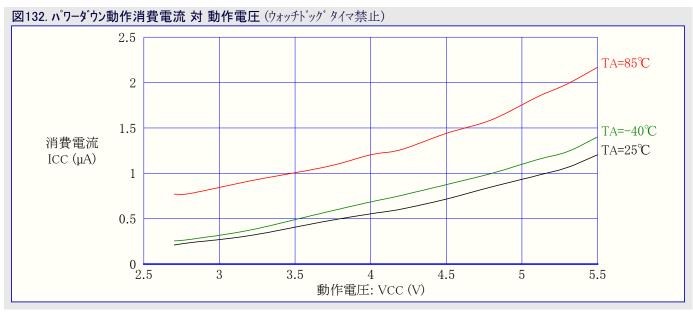






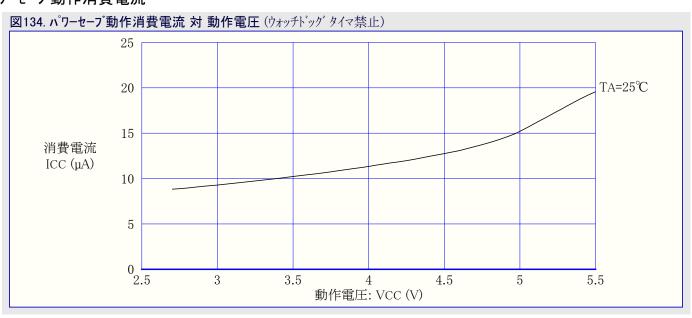


パワーダウン動作消費電流



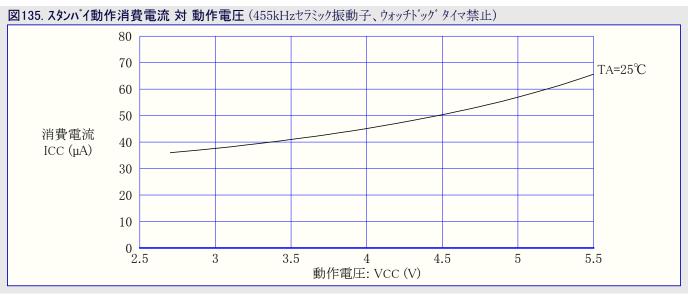


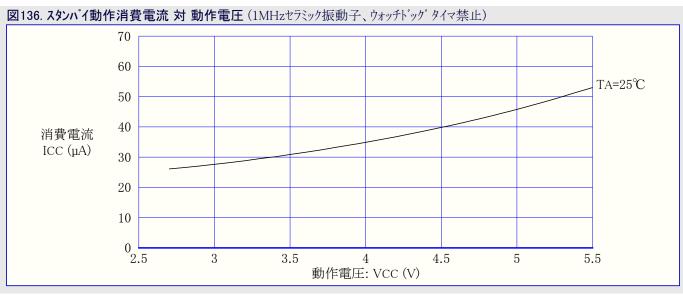
パワーセーブ動作消費電流

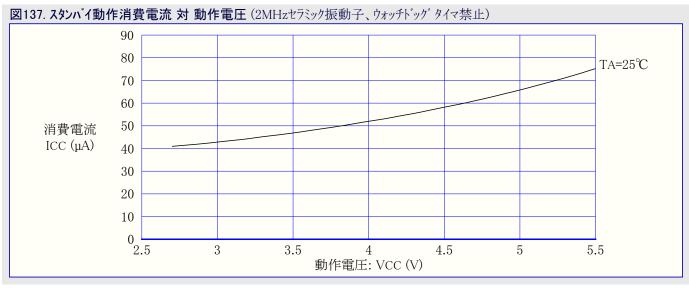




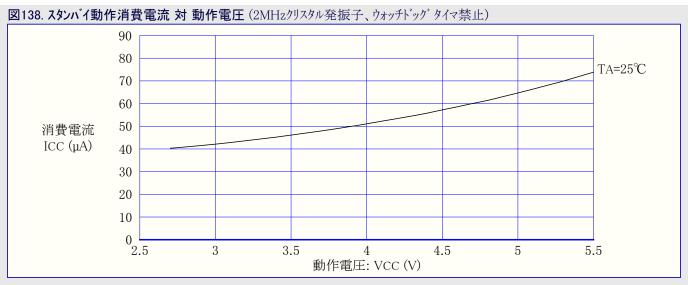
スタンバイ動作消費電流

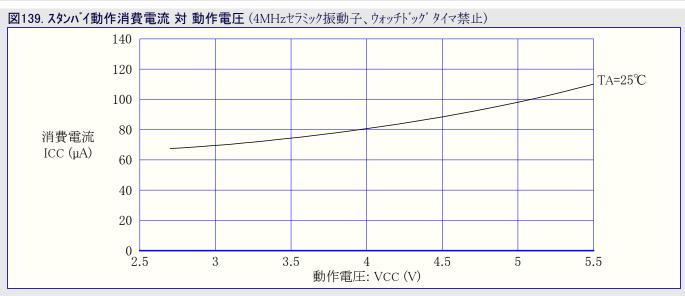


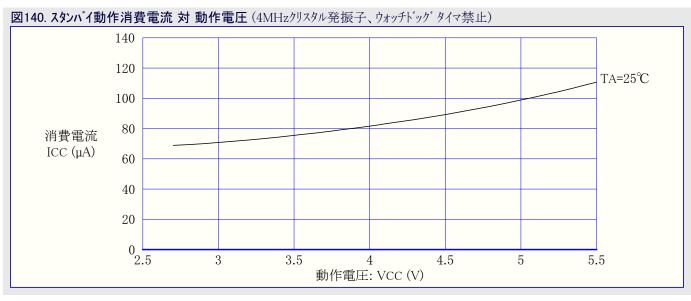




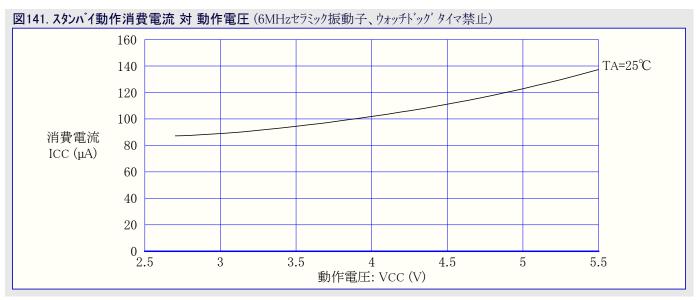


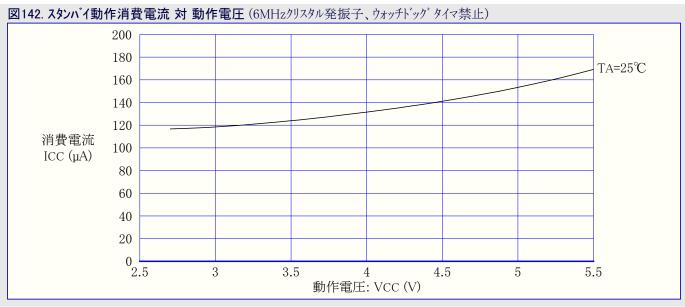




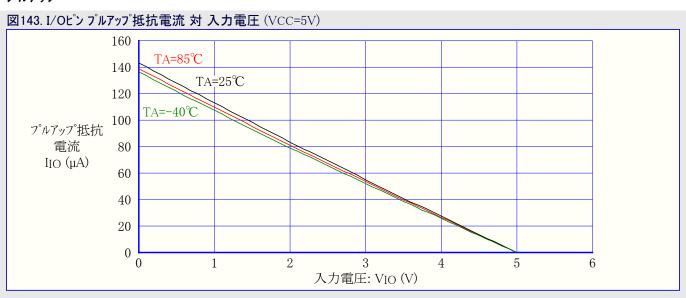




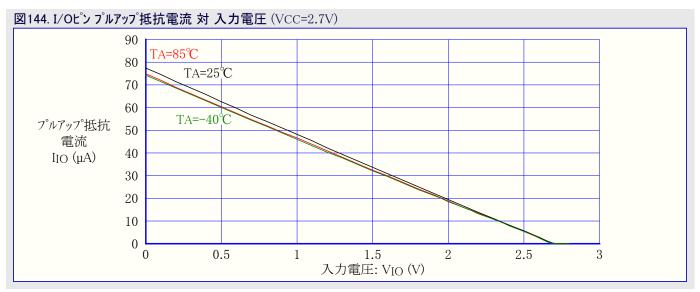


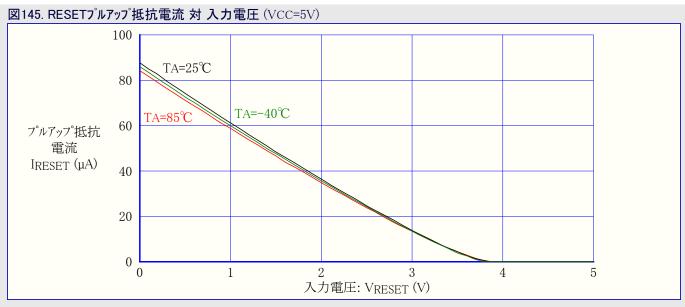


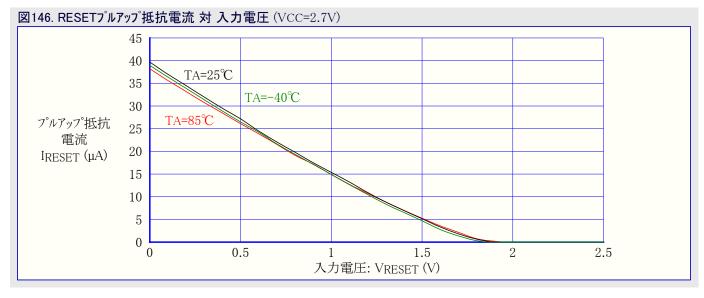
ピン プルアップ





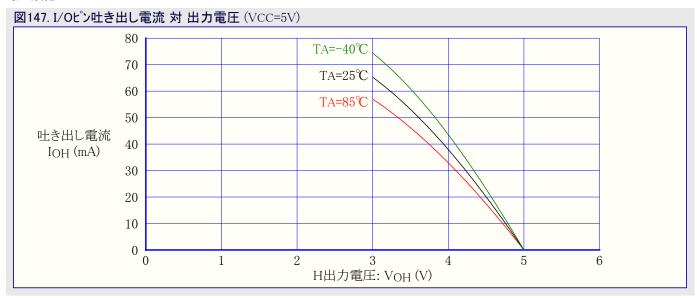


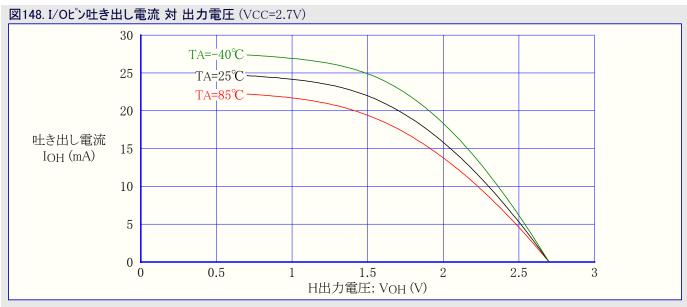


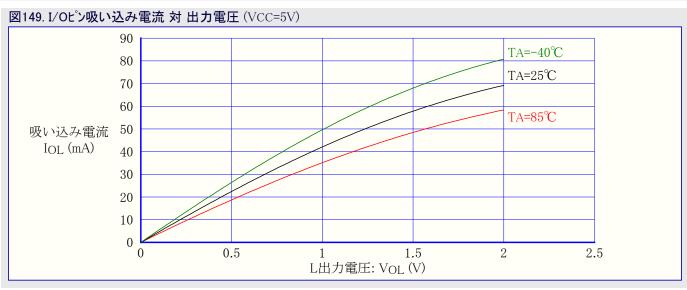




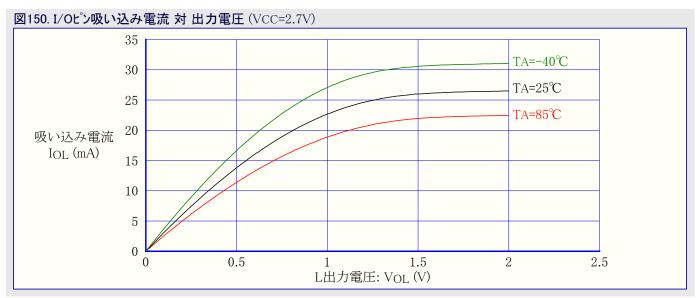
ピン駆動能力

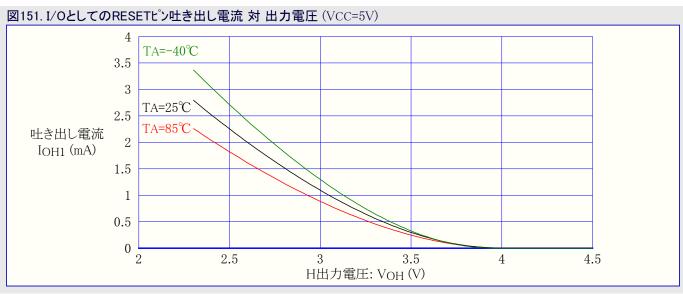


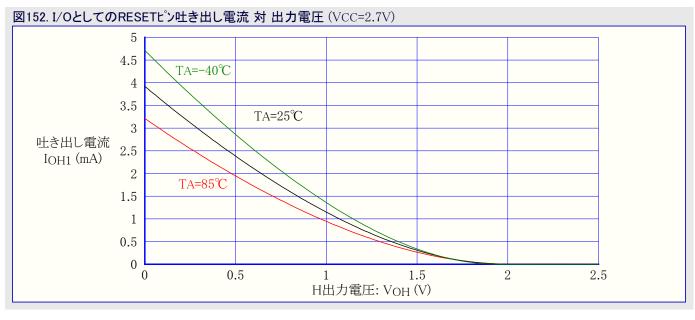




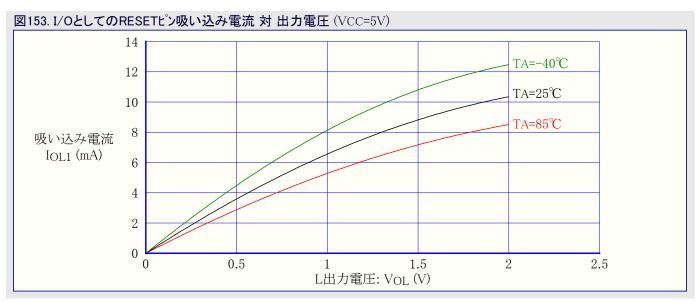


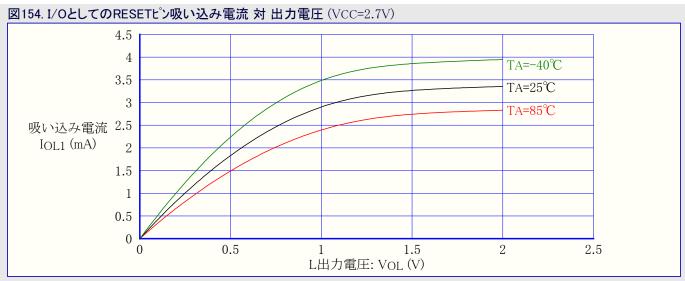




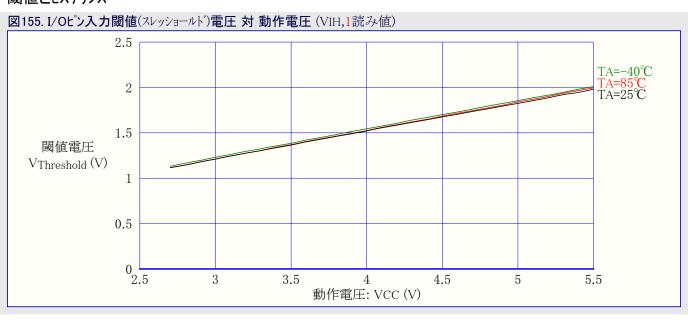




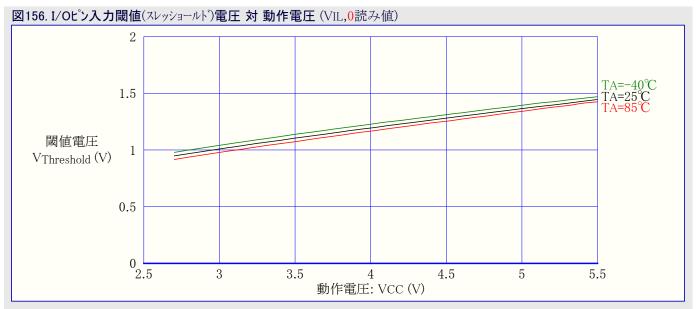


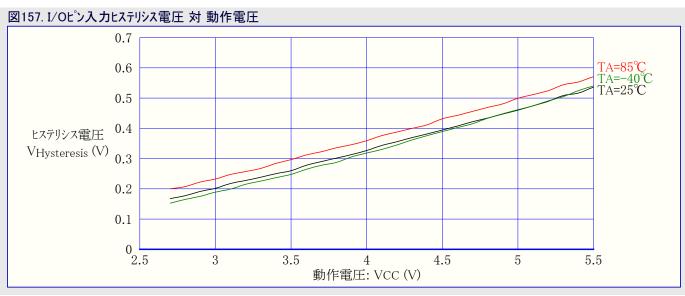


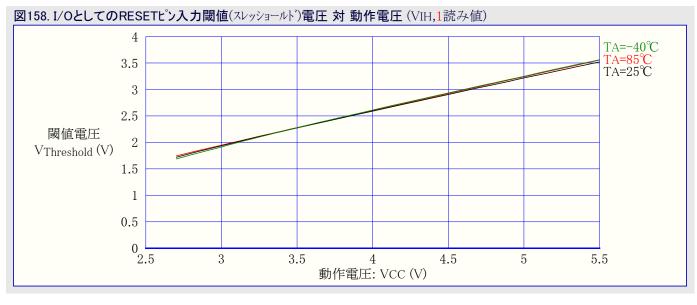
ピン 閾値とヒステリシス



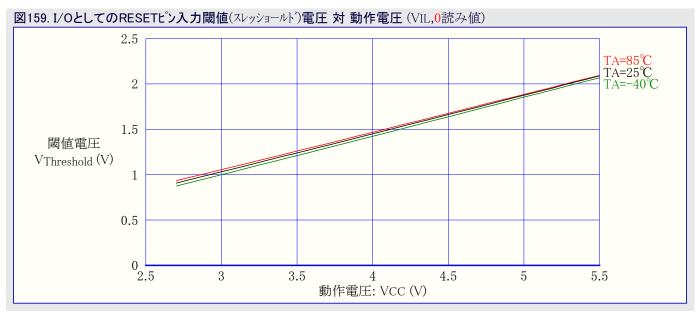


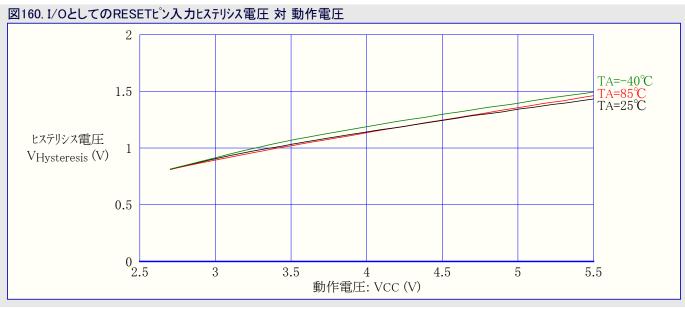


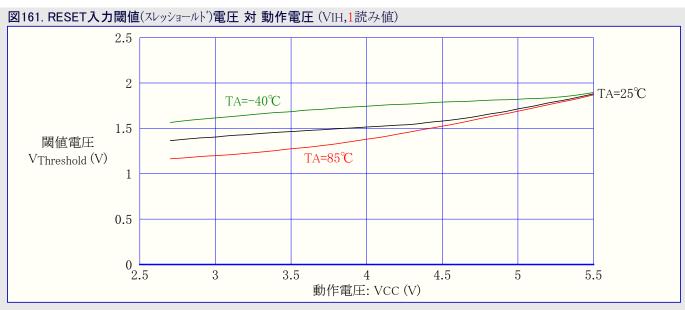






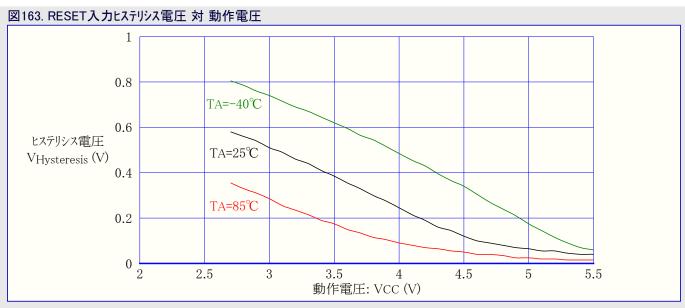




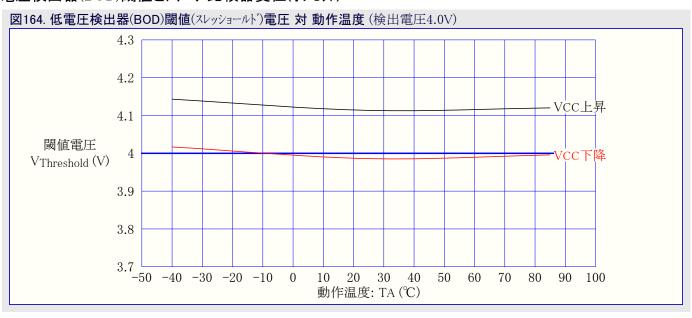




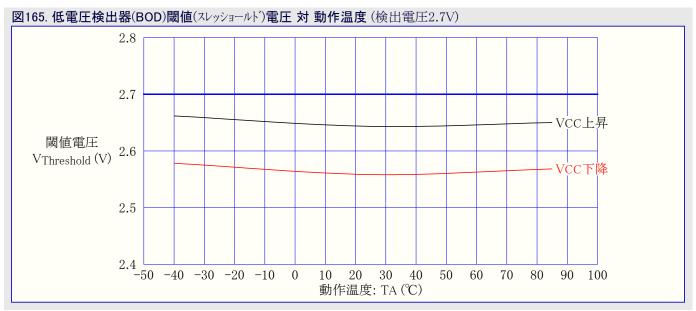


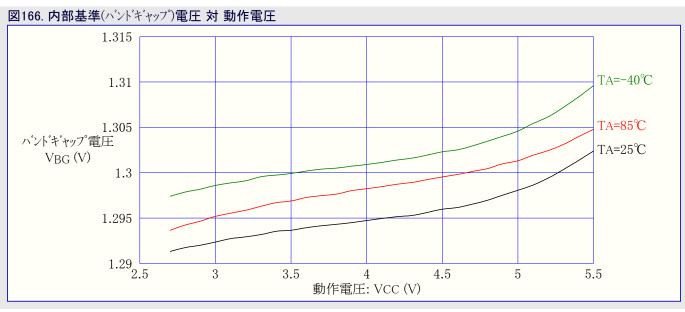


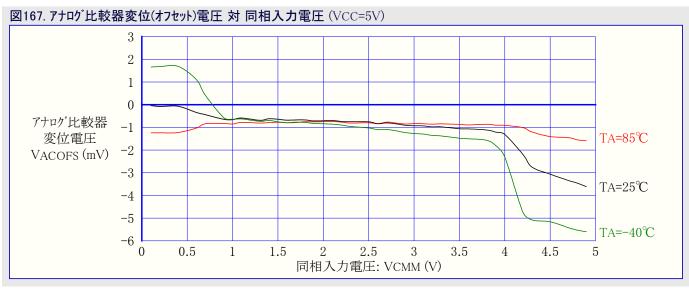
低電圧検出器(BOD)閾値とアナログ比較器変位(オフセット)



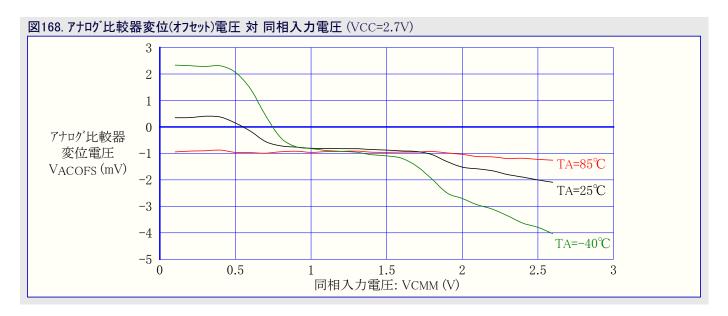




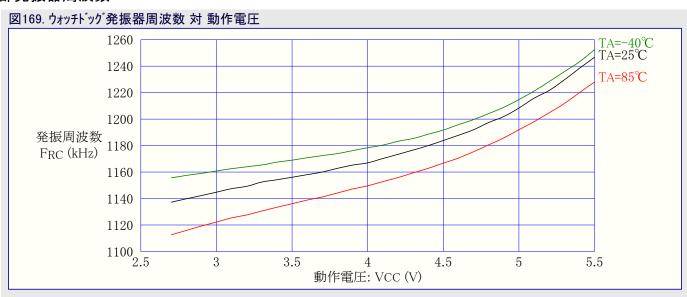


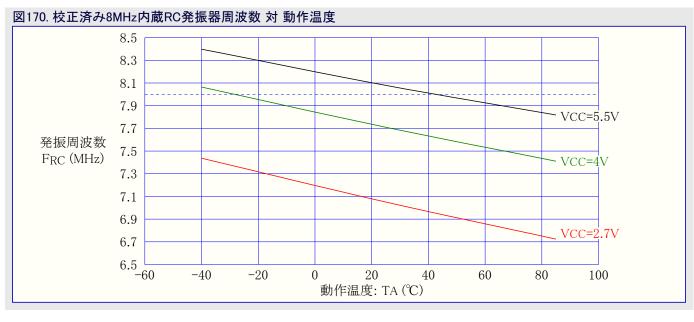




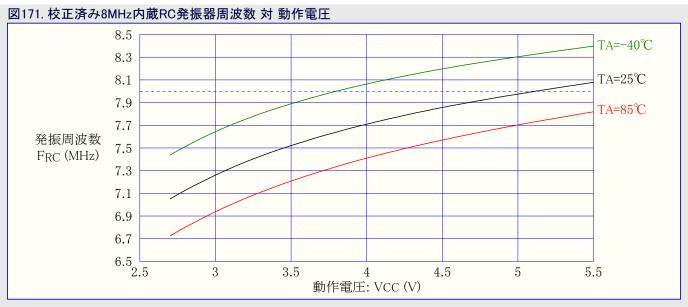


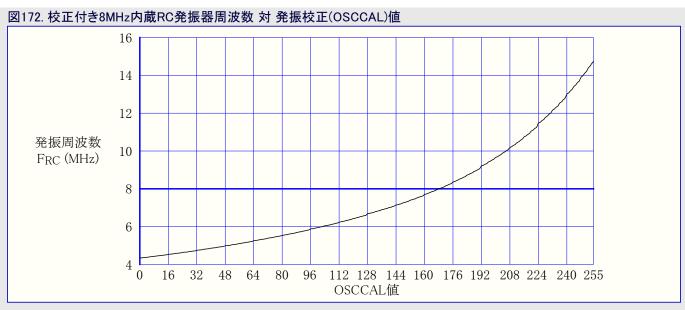
内部発振器周波数

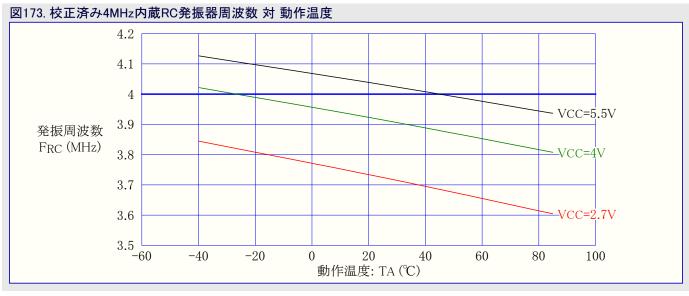




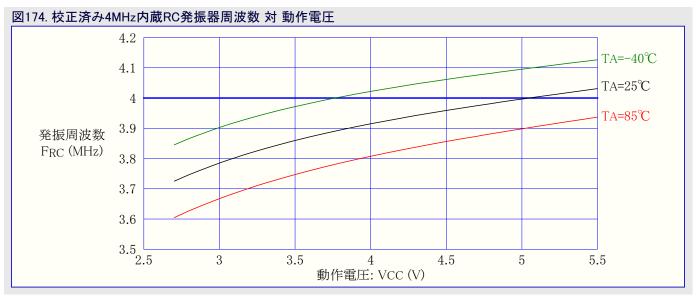


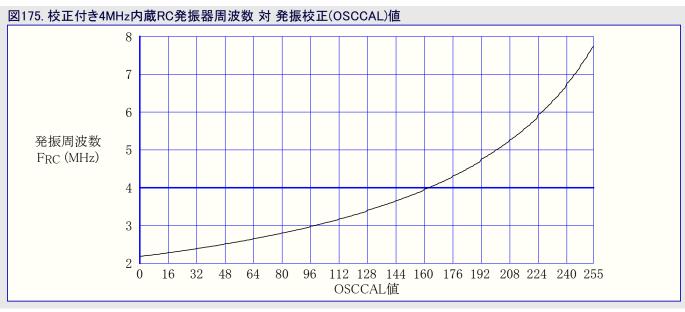


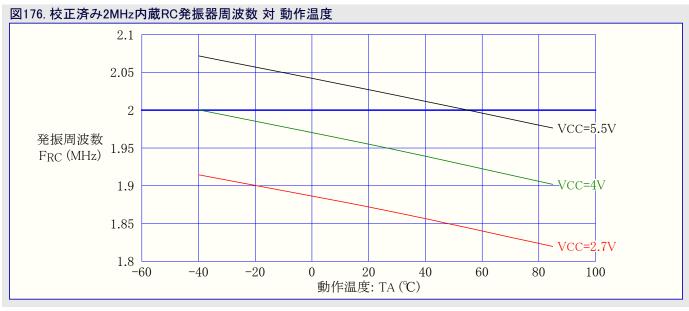




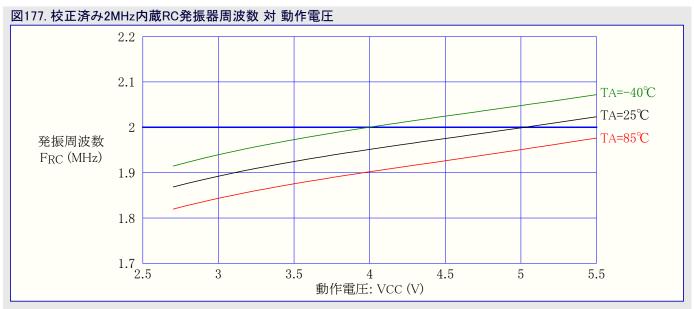


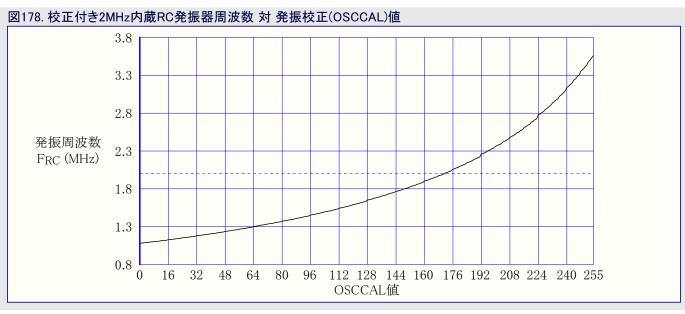


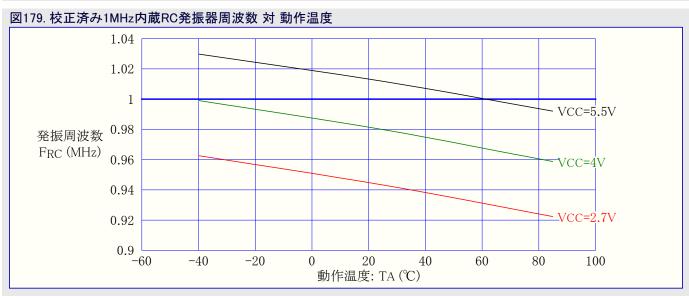




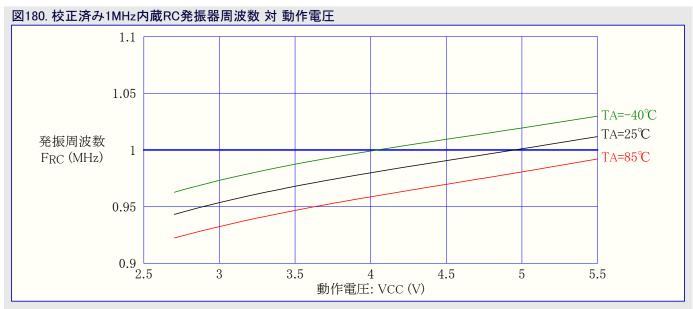


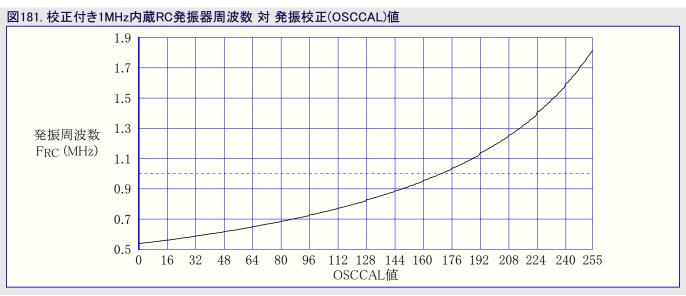




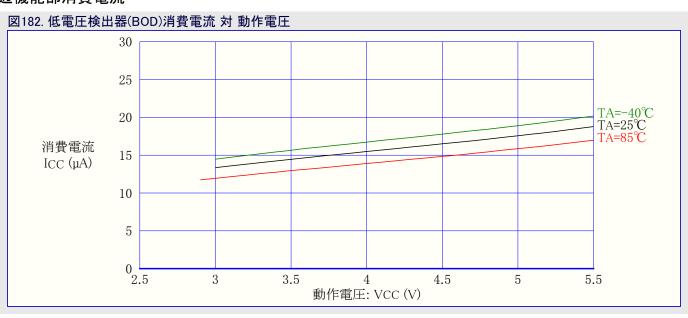




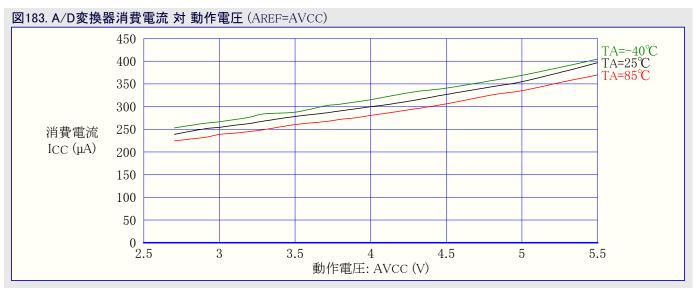


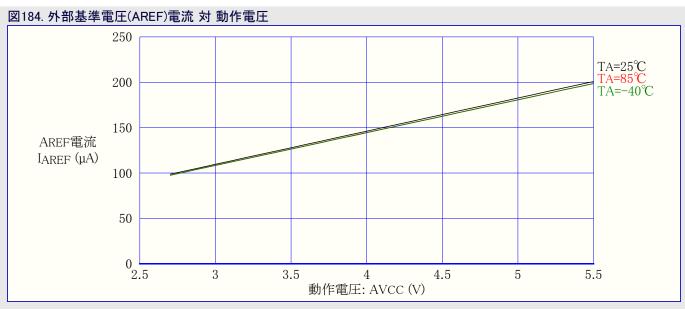


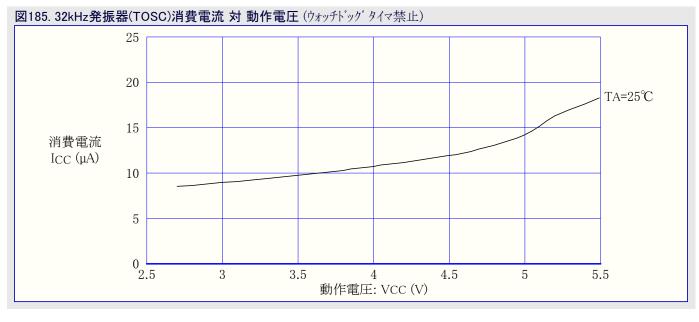
周辺機能部消費電流



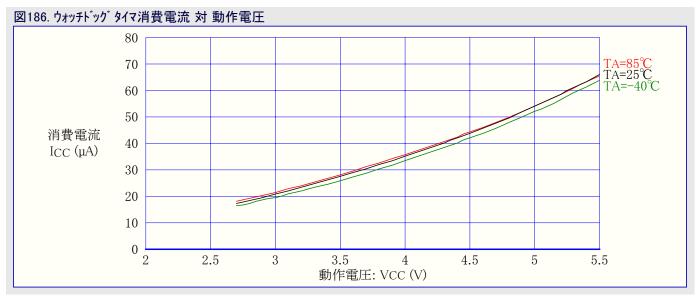


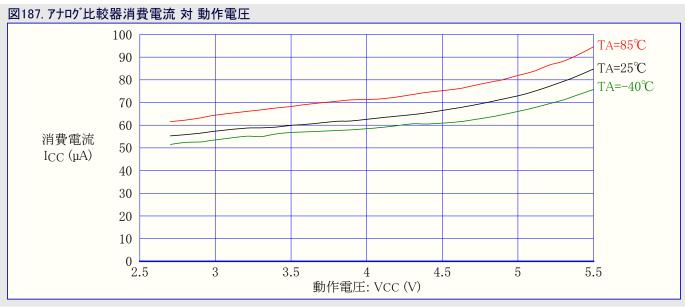


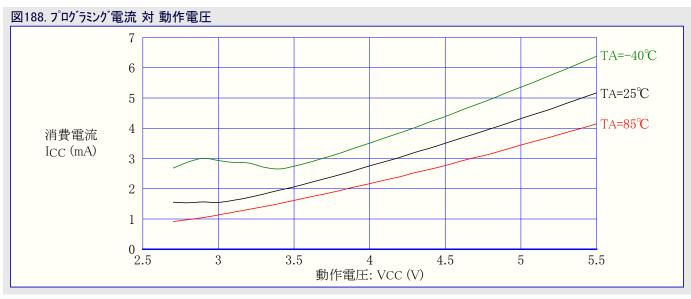






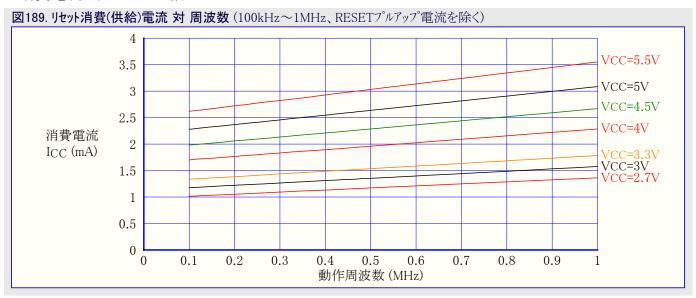


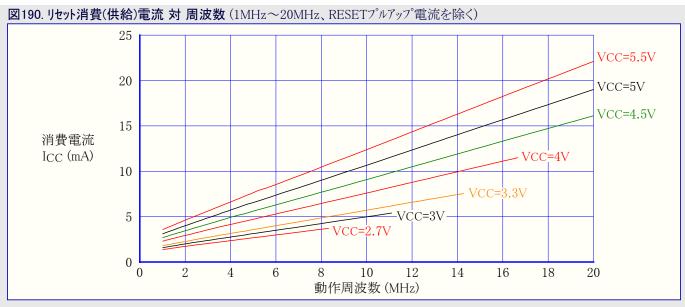


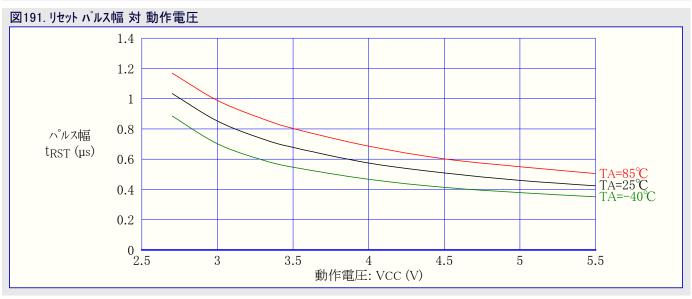




リセット消費電流とリセットパルス幅









代表特性 - (TA=-40°C~105°C)

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。

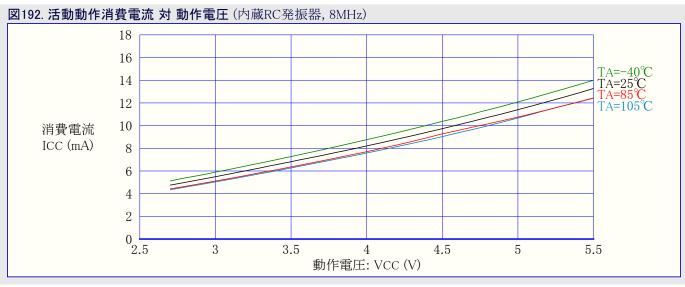
消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。 支配的な要素は動作電圧と動作周波数です。

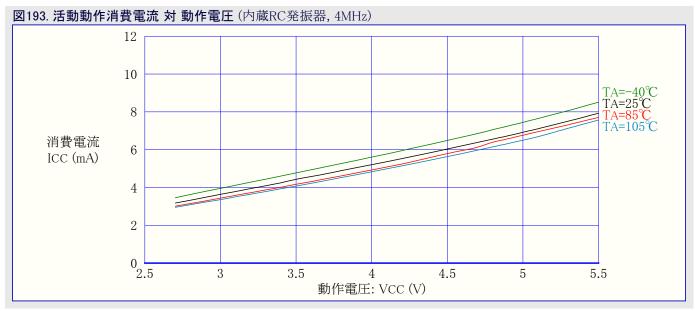
容量性負荷の \mathfrak{l}° ンの引き込み電流は(1つの \mathfrak{l}° ンに対して) C_L (負荷容量)× V_{CC} (動作電圧)×f($I/O\mathfrak{l}^{\circ}$ ンの平均切り替え周波数) として推測できます。

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。

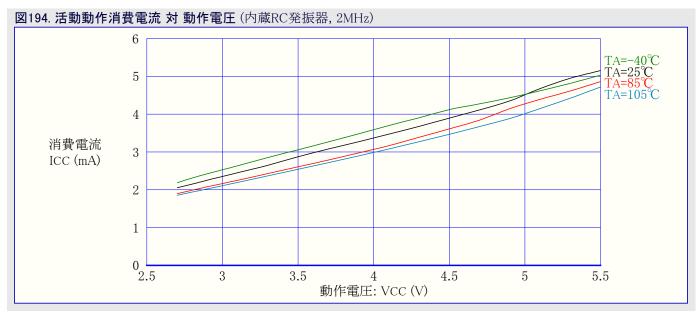
ウォッチト、ック、タイマ許可のハ。ワーダ・ウン動作での消費電流とウォッチト、ック、タイマ禁止のハ。ワーダ・ウン動作での消費電流間の違いは、ウォッチト、ック、タイマによって引き込んだ(消費した)差電流を表します。

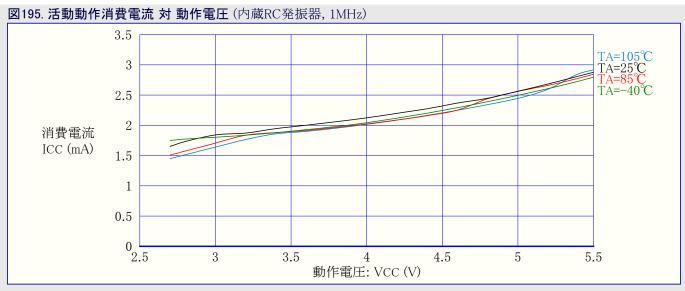
活動動作消費電流



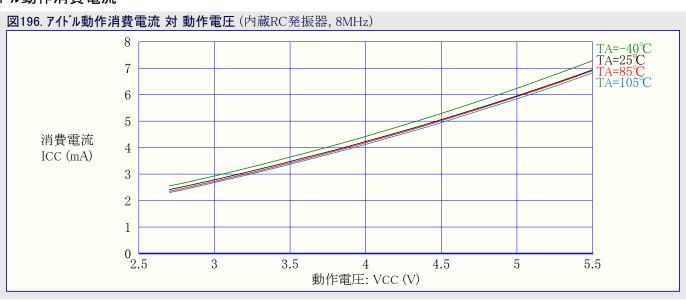




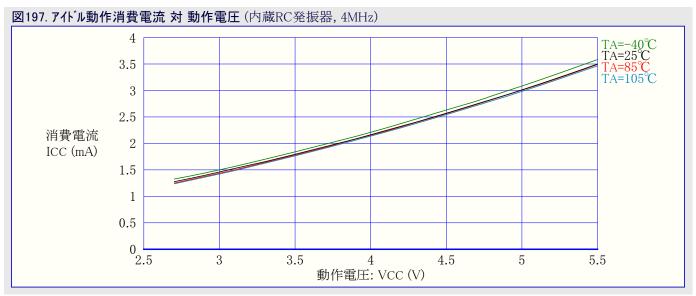


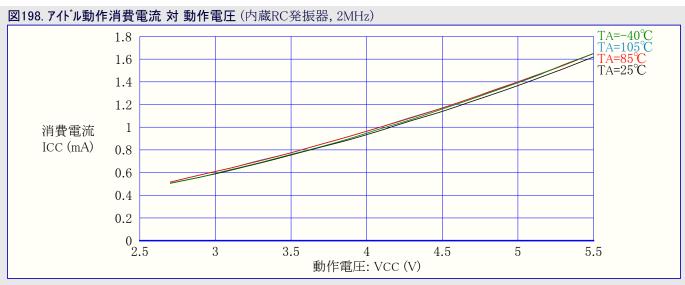


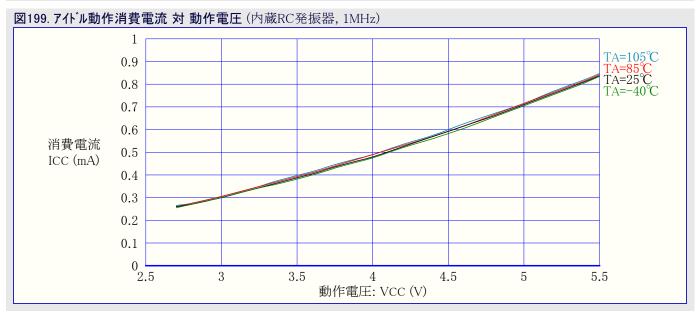
アイドル動作消費電流





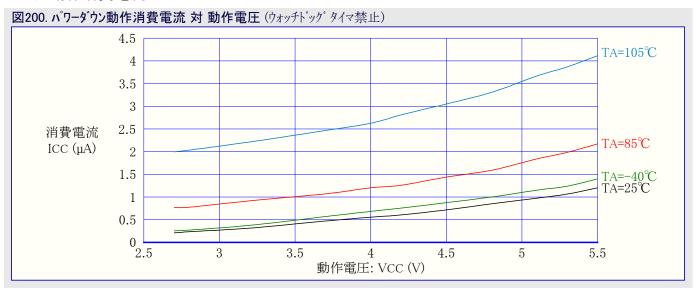


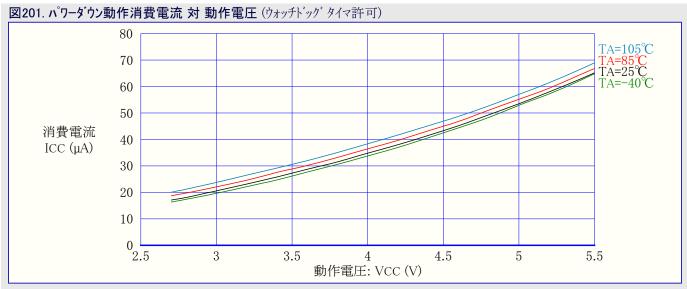




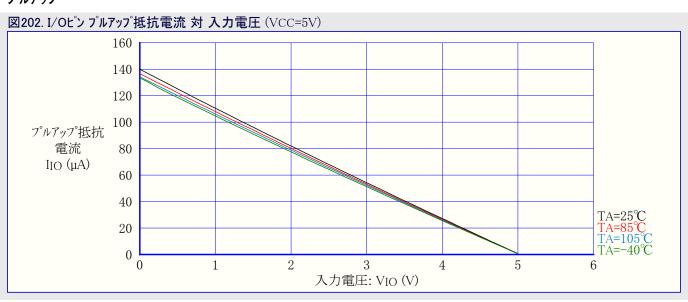


パワーダウン動作消費電流

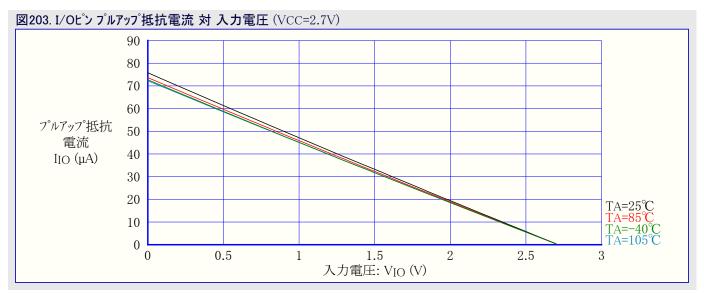


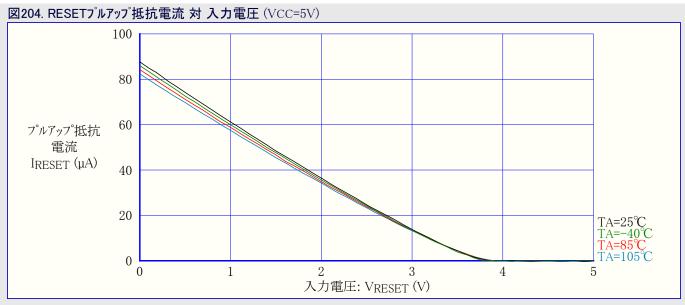


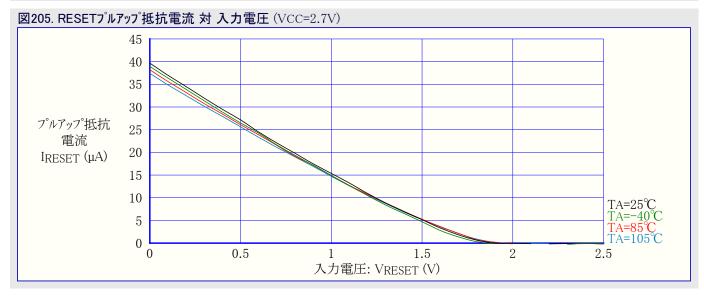
ピン プルアップ





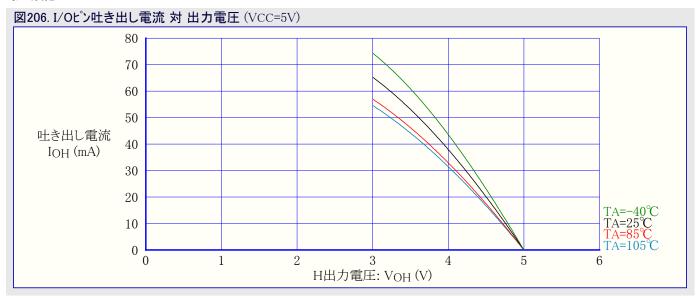


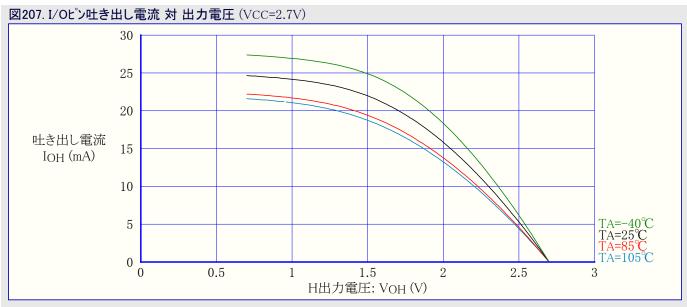


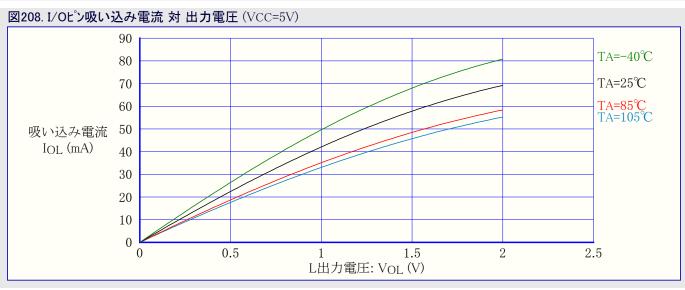


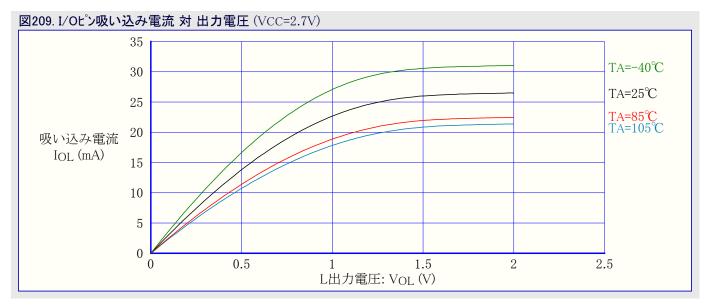


ピン駆動能力

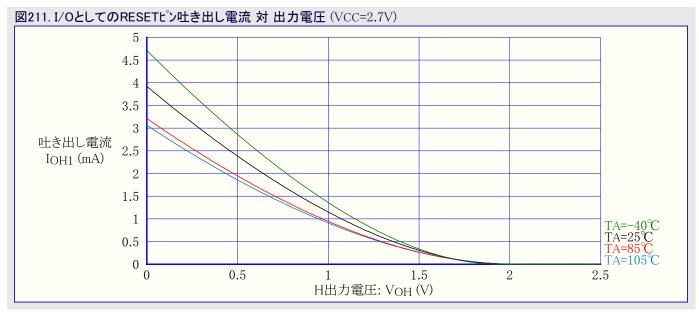




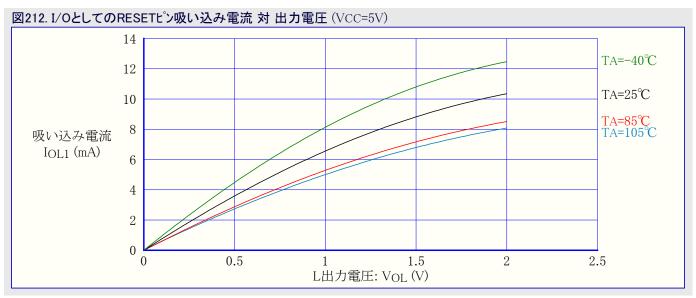


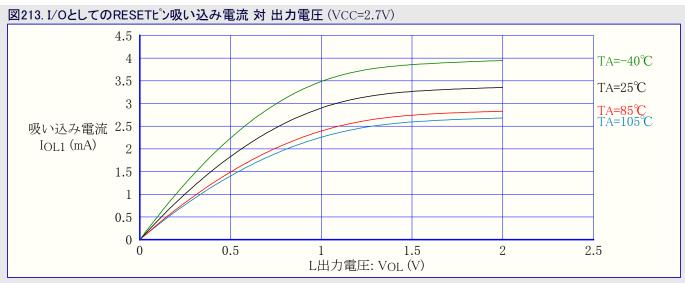




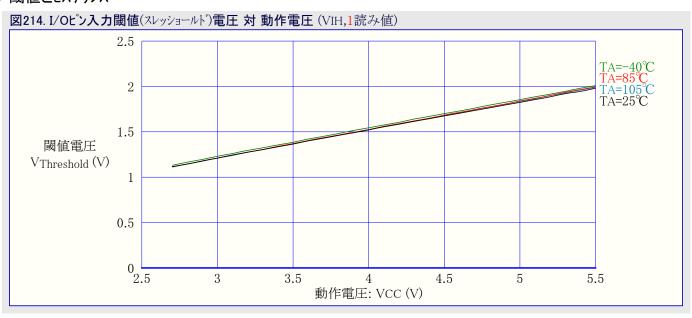




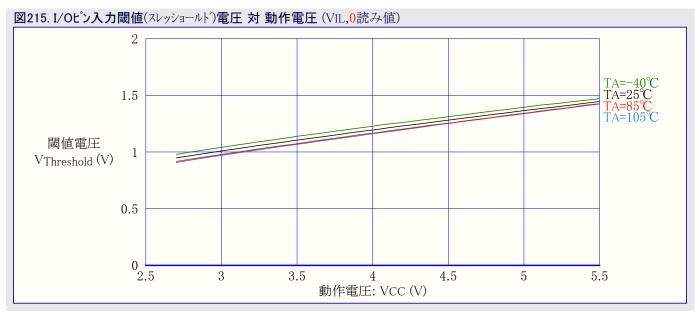


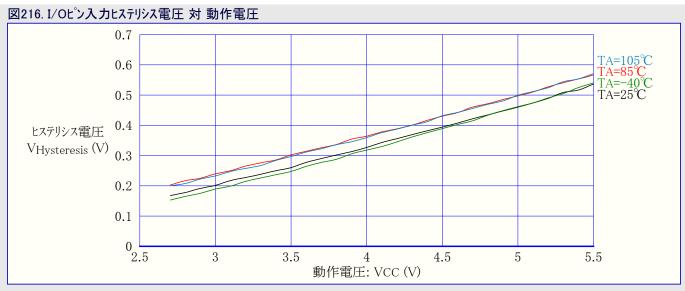


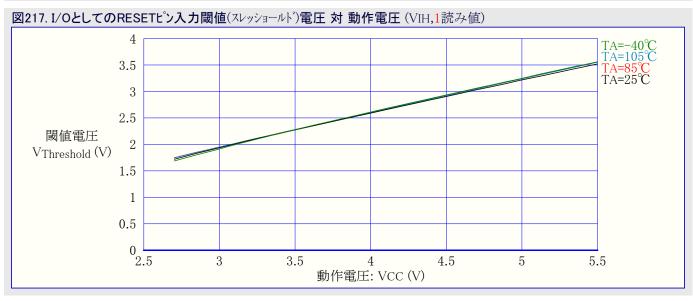
ピン 閾値とヒステリシス



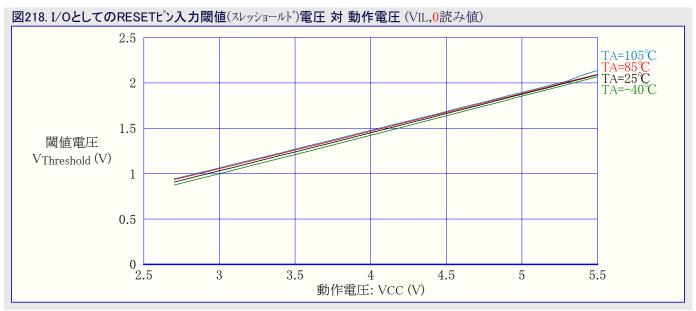


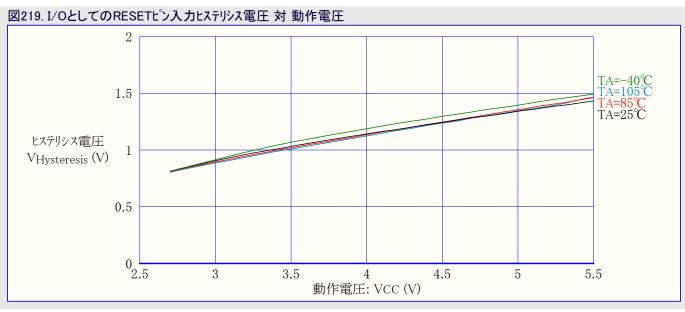


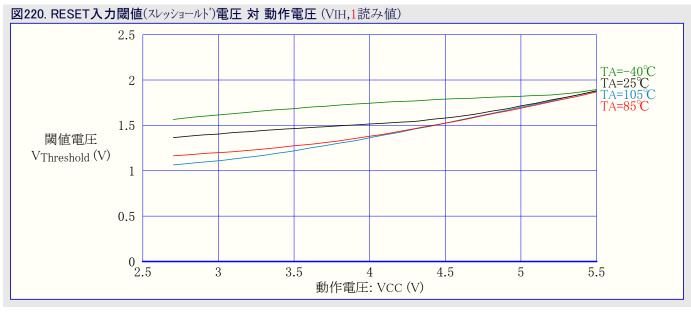




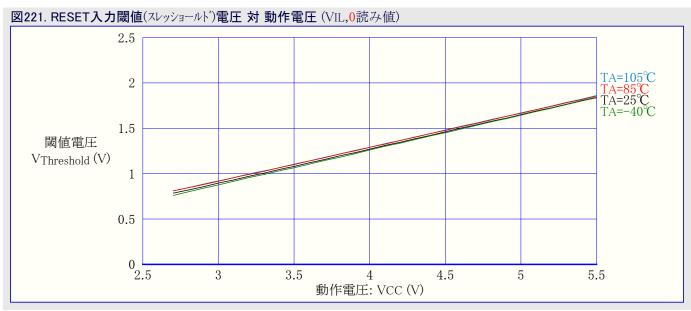


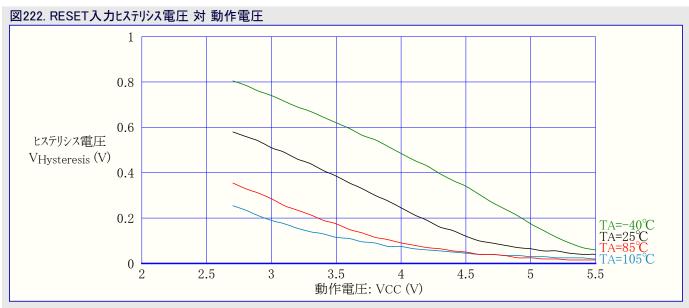




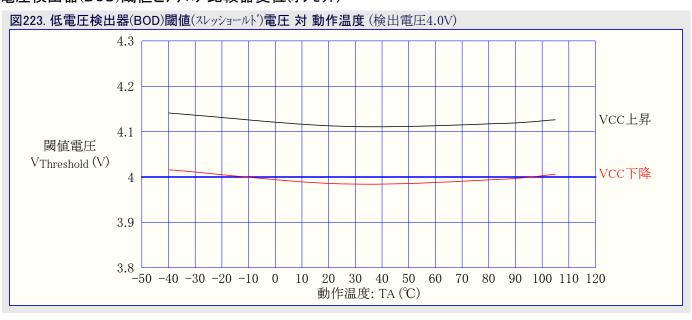




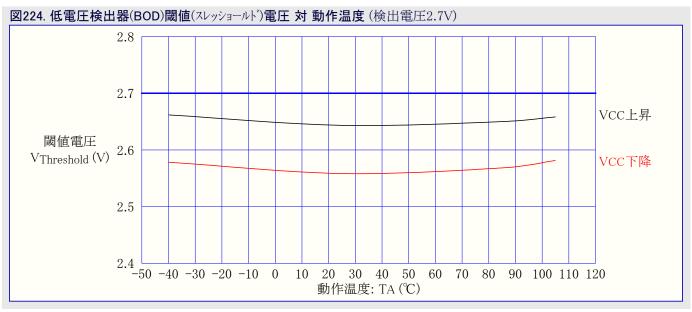


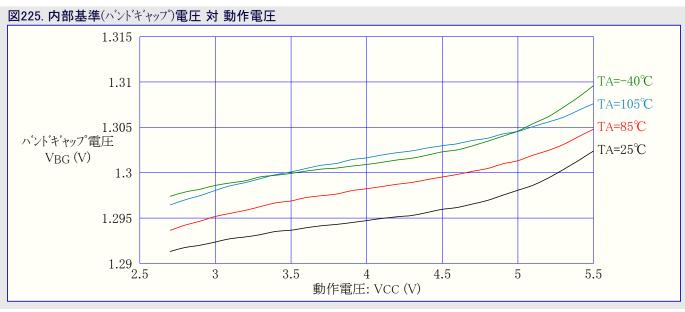


低電圧検出器(BOD)閾値とアナログ比較器変位(オフセット)



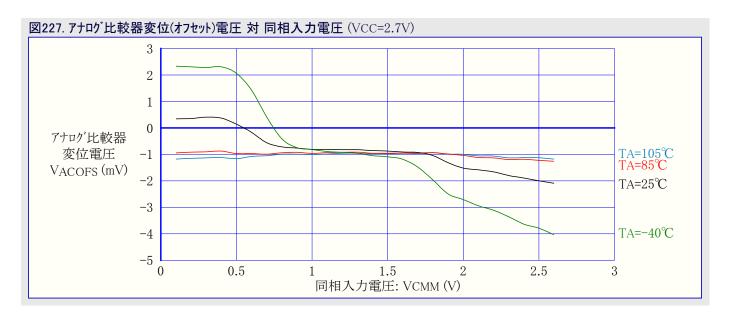




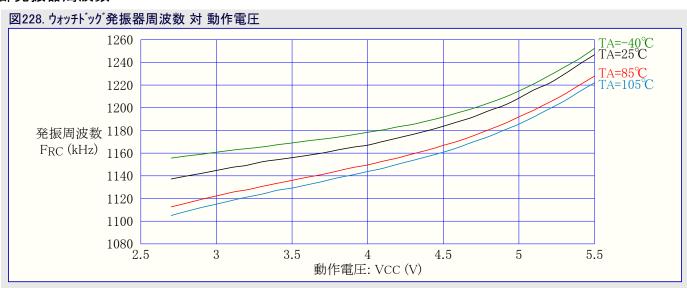


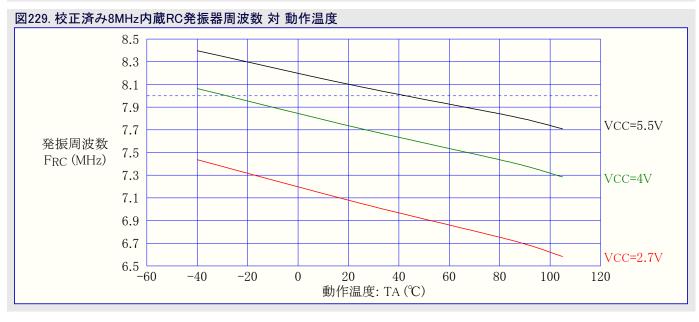




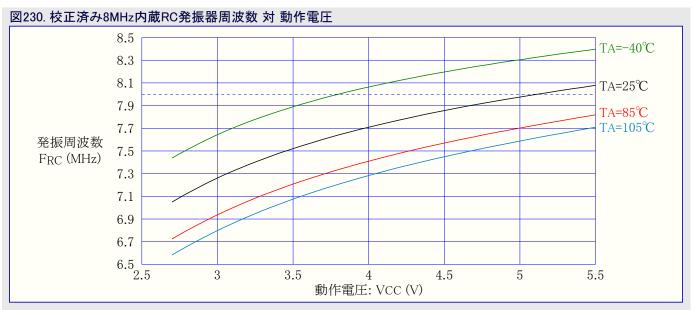


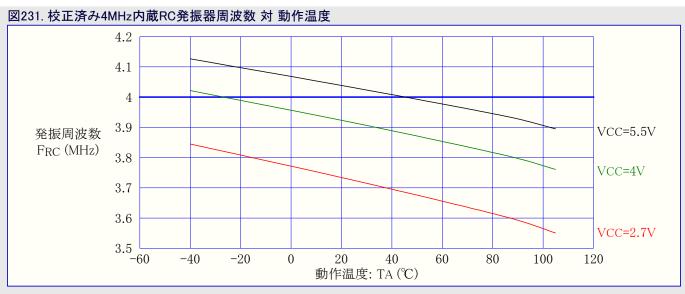
内部発振器周波数

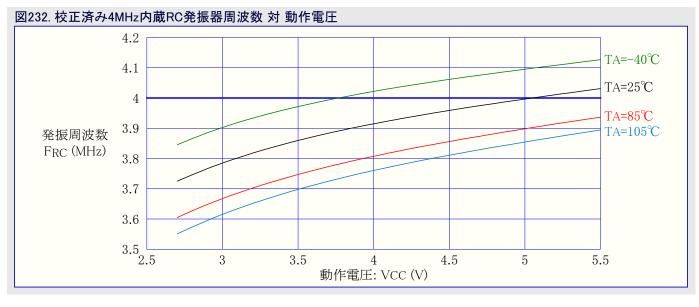




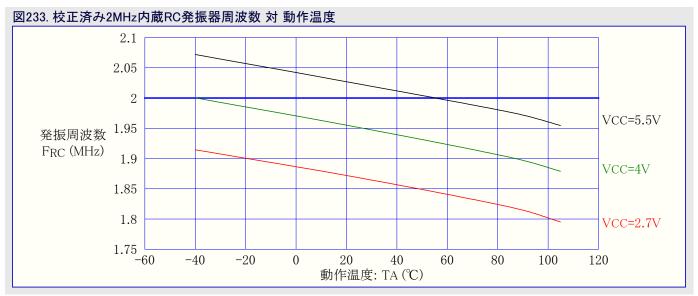


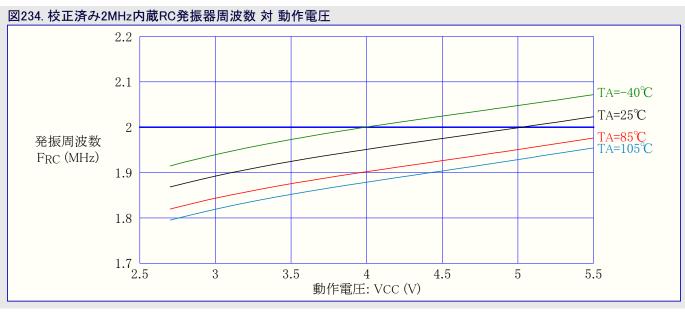


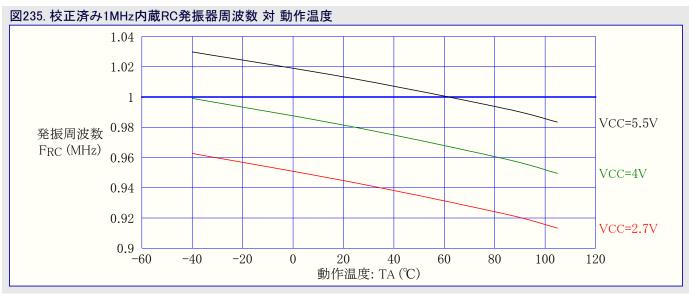




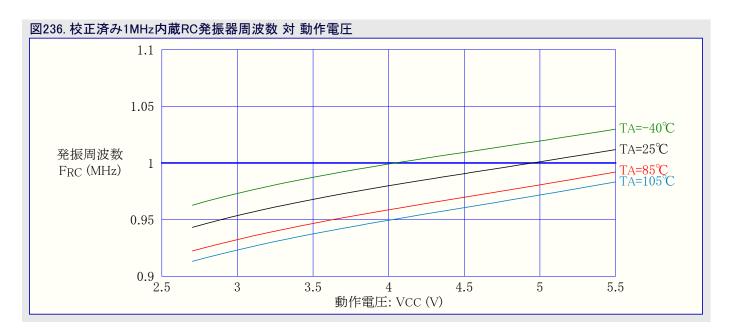




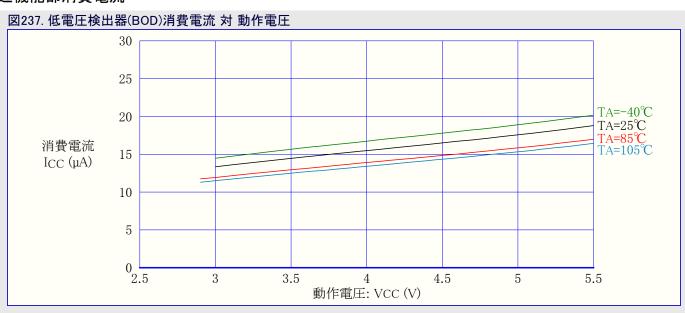


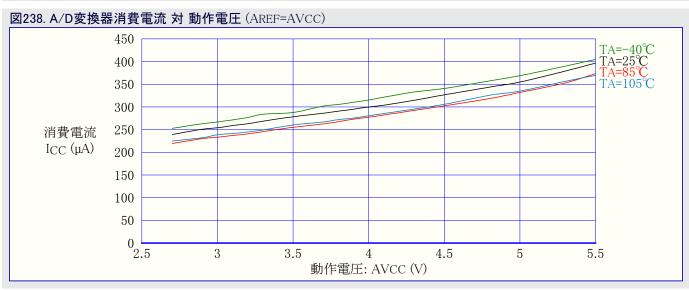




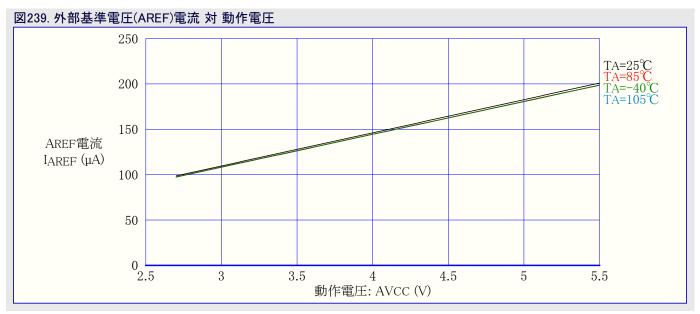


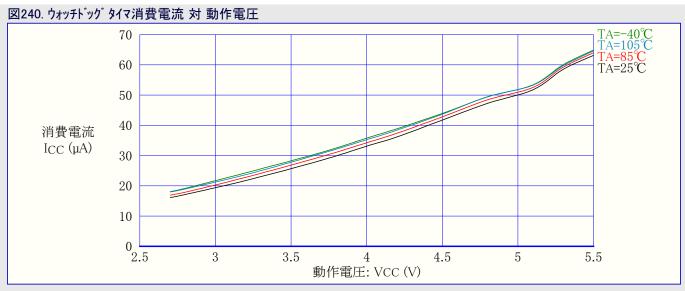
周辺機能部消費電流

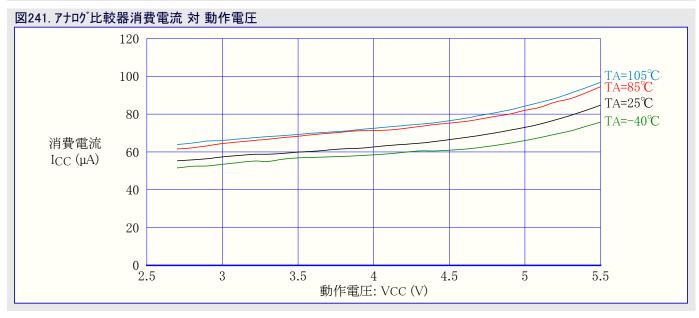




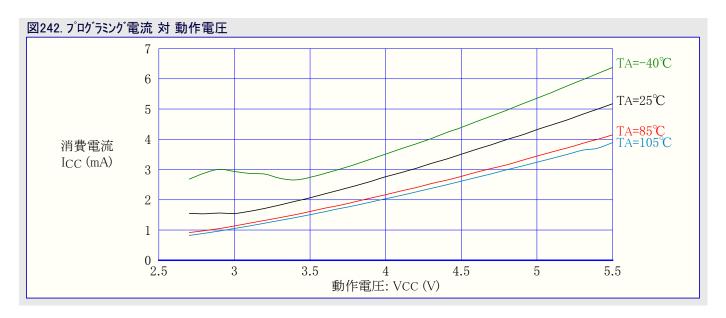




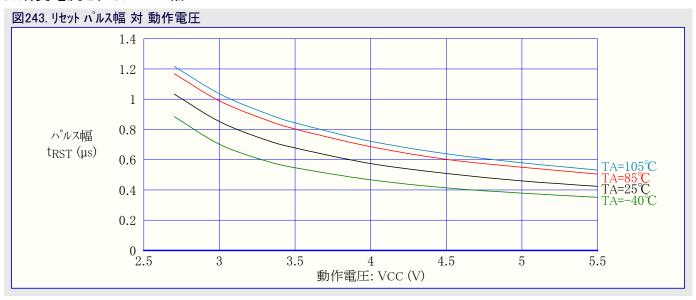








リセット消費電流とリセットパルス幅





レジスタ要約

アト・レス	レジスタ略称	ヒ゛ット7	ビット6	ヒ゛ット5	じット4	ヒ゛ット3	ピット2	ピット1	ピット0	頁
\$3F (\$5F)	SREG	I	Т	Н	S	V	N	Z	С	7
\$3E (\$5E)	SPH	_	-	-	-	-	SP10	SP9	SP8	8
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	
\$3C (\$5C)	予約									
\$3B (\$5B)	GICR	INT1	INTO	-	-	-	-	IVSEL	IVCE	44,31
\$3A (\$5A)	GIFR	INTF1	INTF0	- TIOUE1	- OCIE14	- OCUEAR	- TOIR1	-		44
\$39 (\$59) \$38 (\$58)	TIMSK TIFR	OCIE2 OCF2	TOIE2 TOV2	TICIE1 ICF1	OCIE1A	OCIE1B OCF1B	TOIE1 TOV1	_	TOIE0 TOV0	79,68,47
\$37 (\$57)	SPMCR	SPMIE	RWWSB	ICF1	OCF1A RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	79,68,47 138
\$36 (\$56)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	- T GERS	TWIE	111
\$35 (\$55)	MCUCR	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00	20,43
\$34 (\$54)	MCUCSR	-	-	-	-	WDRF BORF		EXTRF	PORF	25
\$33 (\$53)	TCCR0	-	_	-	-	-	CS02	CS01	CS00	47
\$32 (\$52)	TCNT0		-		タイマ/:	カウンタ0				47
\$31 (\$51)	OSCCAL			内蔵	RC発振器 矛	を振校正値レ	シスタ			19
\$30 (\$50)	SFIOR	_	-	_	-	ACME	PUD	PSR2	PSR10	126,36,82,49
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	64
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	66
\$2D (\$4D)	TCNT1H				タイマ/カウンタ					66
\$2C (\$4C)	TCNT1L				タイマ/カウンタ		1. * 22			
\$2B (\$4B)	OCR1AH				カウンタ1 比較					67
\$2A (\$4A)	OCR1AL		-		カウンタ1 比較		立バイト			
\$29 (\$49) \$28 (\$48)	OCR1BH OCR1BL		-		カウンタ1 比較 カウンタ1 比較		<u>立バイト </u>			67
\$27 (\$47)	ICR1H		-	カイマ/	カウンタ1 元戦 カウンタ1 捕獲	ロンジスタートは				
\$26 (\$46)	ICR11		+		(カウンタ1 捕獲				-	67
\$25 (\$45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	77
\$24 (\$44)	TCNT2	1002	WGMZ0	COME		カウンタ2	0022	0521	0520	78
\$23 (\$43)	OCR2		-	-		比較レジスタ	,	-		78
\$22 (\$42)	ASSR	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	80
\$21 (\$41)	WDTCR	-	-	-	WDCE	WDE	WDP2	WDP1	WDP0	26
¢20 (¢40)	UBRRH	URSEL	_	_	_		ボーレート レシブ	タ上位 (UBF		103
\$20 (\$40)	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	102
\$1F (\$3F)	EEARH	-	-	-	-	-	-	-	EEAR8	12
\$1E (\$3E)	EEARL		1	EEPROM	トレス レシ スタ		EEAR7~0)			
\$1D (\$3D)	EEDR			<u> </u>	EEPROM	ータ レシブスタ	DD) GUD			12
\$1C (\$3C)	EECR	-	-	_	_	EERIE	EEMWE	EEWE	EERE	12
\$1B (\$3B)	予約									
\$1A (\$3A) \$19 (\$39)	<u>予約</u> 予約									
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	42
\$17 (\$37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	42
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	42
\$15 (\$35)	PORTC	-	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	42
\$14 (\$34)	DDRC	-	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	42
\$13 (\$33)	PINC	-	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	42
\$12 (\$32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	42
\$11 (\$31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	42
\$10 (\$30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	42
\$0F (\$2F)	SPDR	05-5	111001		SPIデー	タレジスタ			aprari	86
\$0E (\$2E)	SPSR	SPIF	WCOL	- D.C.D.D.	-	- CDC:	- CPLLA	- CDD	SPI2X	86
\$0D (\$2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	СРНА	SPR1	SPR0	85
\$0C (\$2C)	UDR	DVC	TVC	LIDDE		ータレジスタ	DE	LIOV	MDCM	101
\$0B (\$2B) \$0A (\$2A)	UCSRA UCSRB	RXC RXCIE	TXC TXCIE	UDRE UDRIE	FE RXEN	DOR TXEN	PE UCSZ2	U2X RXB8	MPCM TXB8	101 102
\$0A (\$2A) \$09 (\$29)	UBRRL	KACIE	IACIE		KXEN -レート レジスタ			KAD8	IAD8	102
\$09 (\$29)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	125
\$07 (\$27)	ADMUX	REFS1	REFS0	ADLAR	ACI -	MUX3	MUX2	MUX1	MUX0	133
\$06 (\$26)	ADCSRA	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	133
\$05 (\$25)	ADCH	TIDEN			タ 上位バイト				1101 00	
\$04 (\$24)	ADCL				メタ 下位バイ					134
\$03 (\$23)	TWDR		1		直列インターフェ					112
\$02 (\$22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	112
\$01 (\$21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	_	TWPS1	TWPS0	112
\$00 (\$20)	TWBR			2線直	列インターフェー	ースビット速度	レシ、スタ			111
÷. • HDDD	HとUCSRCの	アカヤフ七汁	カ学を加えてい	VTH FLICAL	ロエュミコンポナッチ	立四1 ナノゼ	411			

- 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 \sim 1 F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	オペラント゛	意味	動作	フラク゛	クロック
ADD	DID		国演算命令	LTHOUNG C	1
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 ADIW	Rd,Rr Rd,K6	キャリーを含めた汎用レジスタ間の加算 即値の語(ワード)長加算	$ \begin{array}{l} Rd \leftarrow Rd + Rr + C \\ RdH:RdL \leftarrow RdH:RdL + K6 \end{array} $	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	1 2
SUB	Rd,Rr	汎用レジスタ間の減算	Rd ← Rd − Rr	I.T.H.S.V.N.Z.C	1
SUBI	Rd,Ki	汎用レジスタから即値の減算	Rd ← 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 ← 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 ← 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 ← Rd 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 ← Rd AND K	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レシブスタ間の論理和(OR)	Rd ← Rd OR Rr	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レシブスタと即値の論理和(OR)	Rd ← Rd OR K	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レシブスタ間の排他的論理和(Ex-OR)	Rd ← Rd EOR Rr	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	Rd ← \$FF - Rd	$I,T,H,S,0,N,Z,\Omega$	1
NEG	Rd	2の補数	Rd ← \$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 ← Rd 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(<mark>0</mark>)	$Rd \leftarrow Rd \ AND \ (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レシブスタの増加(+1)	Rd ← Rd + 1	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	Rd ← Rd - 1	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レシブスタのセゴロとマイナス検査	Rd ← Rd AND Rd	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レシブスタの全 <mark>0</mark> 設定(=\$00)	Rd ← Rd EOR Rd	I,T,H,0,0,0,1,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	Rd ← \$FF	I,T,H,S,V,N,Z,C	1
MUL	Rd,Rr	符号なし間の乗算	$R1:R0 \leftarrow Rd \times Rr$ $(U \times U)$	I,T,H,S,V,N,Z,C	2
MULS	Rd,Rr	符号付き間の乗算	$R1:R0 \leftarrow Rd \times Rr$ $(S \times S)$	I,T,H,S,V,N,Z,C	2
MULSU	Rd,Rr	符号付きと符号なしの乗算	$R1:R0 \leftarrow Rd \times Rr$ (S×U)	I,T,H,S,V,N,Z,C	2
FMUL	Rd,Rr	符号なし間の固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) << 1 \qquad (U \times U)$	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	$R1:R0 \leftarrow (Rd \times Rr) << 1 \qquad (S \times S)$	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	$R1:R0 \leftarrow (Rd \times Rr) << 1 \qquad (S \times U)$	I,T,H,S,V,N,Z,C	2
			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 ← 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レシブ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 ← STACK	I,T,H,S,V,N,Z,C	4
RETI		割り込みからの復帰	PC ← STACK	1 ,T,H,S,V,N,Z,C	4
CPSE	Rd,Rr	汎用レシ、スタ間比較、一致でスキップ	Rd=Rrなら、PC ← PC + 2or3	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	Rd - Rr	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	Rd - Rr - C	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	Rd - K	I,T,H,S,V,N,Z,C	1 /0 0
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ	$Rr(b)=0$ \$\frac{1}{2}\$, $PC \leftarrow PC + 2$ or $RC + 2$	I,T,H,S,V,N,Z,C	1/2,3
SBRS	Rr,b	汎用レジスタのビットが設定(1)でスキップ	$Rr(b)=1/26$, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ。	P(b)=0'\$\hat{c}, PC ← PC + 2\text{or3}	I,T,H,S,V,N,Z,C	
SBIS BRBS	P,b	I/Oレジスタのビットが設定(1)でスキップ ステータス フラグが設定(1)で分岐	P(b)=1なら, PC ← PC + 2or3 SREG(s)=1なら, PC ← PC + K + 1	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	
BRBC	s,k s,k	ステータス フラグが解除(0)で分岐	SREG(s)=1なら, PC ← PC + K + 1 SREG(s)=0なら, PC ← PC + K + 1	I, T, H, S, V, N, Z, C	,
BREQ	k	一致で分岐	Z=1なら, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	$\frac{1/2}{1/2}$
BRNE	k	不一致で分岐	Z=0/\$6, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	$\frac{1/2}{1/2}$
BRCS	k	キャリー フラグが設定(1)で分岐	C=1736, PC ← 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	$C=0$ 7 c 5, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	$\frac{1/2}{1/2}$
BRSH	k	符号なしの≧で分岐	C=07\$6, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしのくで分岐	C=1なら, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	N=1&6, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	N=0\$6, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの≧で分岐	(N EOR V)=0なら, PC ← PC + K + 1	I,T,H,S,V,N,Z,C	
BRLT	k	符号付きのくで分岐	(N EOR V)=1なら, PC ← 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 ← 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 ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時7ラグが設定(<u>1</u>)で分岐	T=1なら, PC ← 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=07\$6, PC ← 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 ← PC + K + 1	I,T,H,S,V,N,Z,C	,
BRVC	k	2の補数溢れフラグが解除(0)で分岐	V=0/\$\$, PC ← 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 ← 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 ← PC + K + 1	I,T,H,S,V,N,Z,C	1/2
VG V · G	のビュルマ米		: 汎用レジスタ(R0~R31) X, Y, Z: X, Y, Zレ		

 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: ピット(0~7)
 k: アドレス定数(7,12,16ビット)
 q: 符号なし6ピット定数(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	オペラント゛	意味	動作	フラク゛	クロック
MON	DID	ず一外		ITHOMN 7.0	-1
MOV MOVW	Rd,Rr Rd,Rr	汎用レジスタ間の複写 汎用レジスタ対間の複写	Rd ← Rr Rd+1:Rd ← Rr+1:Rr	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	Rd ← 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 LD	Rd,-X Rd,Y	事前減少付きXレジスタ間接での取得 Yレジスタ間接での取得	$X \leftarrow X - 1$, $Rd \leftarrow (X)$ $Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	2 2
LD	Rd, Y Rd, Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$	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	
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 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 LD	Rd,Z+ Rd,-Z	事後増加付きZレジスタ間接での取得 事前減少付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	2
LDD	Rd,-Z Rd,Z+q	変位付きZレジスタ間接での取得	$Z \leftarrow Z - 1$, $Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$	I, T, H, S, V, N, Z, C	2 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レシブスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	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 Y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 ST	Y,Rr Y+,Rr	¥レシン メタ間接での設定 事後増加付きYレジスタ間接での設定	$ (Y) \leftarrow Rr $ $ (Y) \leftarrow Rr, Y \leftarrow Y + 1 $	I,T,H,S,V,N,Z,C	2 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 2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow \operatorname{Rr}, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST STD	−Z,Rr Z+q,Rr	事前減少付きZレシ、スタ間接での設定 変位付きZレシ、スタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$ $(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	2 2
STS	k,Rr	家位的さなグーク間接での設定 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(任意のレシ、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 IN	Rd,P	プログラム領域へZレジスタ間接での設定 I/Oレジスタからの入力	(Z) ← R1:R0 Rd ← P	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	<u>-</u>
OUT	P,Rr	I/OVシスタがらい人力 I/OVシスタへの出力	P ← 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 ← Rr	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタ〜復帰	Rd ← STACK	I,T,H,S,V,N,Z,C	2
CDI	D.I	Li(Oliviahのは、Lith中(1)	係命令 [VO(D1) c 1	ITHCMN 7 C	0
SBI CBI	P,b P,b	I/Oレシブスタのビット設定(1) I/Oレシブスタのビット解除(0)	$ \frac{I/O(P,b) \leftarrow 1}{I/O(P,b) \leftarrow 0} $	I,T,H,S,V,N,Z,C I,T,H,S,V,N,Z,C	2 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,\emptyset,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 SWAP	Rd Rd	算術的右ビット移動 ニブル(4ビット)上位/下位交換	$Rd(n) \leftarrow Rd(n+1), n=0 \sim 6$ $Rd(7 \sim 4) \Leftrightarrow Rd(3 \sim 0)$	I,T,H,S,V,N,Z,C 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, T, H, \$, Y, N, Z, C	1
BCLR	S	ステータス レシ スタのヒ 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 CLC		キャリー フラグを設定(1) キャリー フラグを解除(0)	C ← 1 C ← 0	I,T,H,S,V,N,Z, ① I,T,H,S,V,N,Z, ①	1
SEN		(1) (1) (2) (2) (2) (2) (3) (4) (5) (6) (7) (7) (7) (7) (7) (7) (7) (7) (7) (7	$N \leftarrow 1$	I,T,H,S,V,N,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		セ゛ロ フラケ を設定(1)	Z ← 1	I,T,H,S,V,N,I,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,Ø,C	1
SEI CLI		全割り込み許可 全割り込み禁止	$ \begin{array}{c} I \leftarrow 1 \\ I \leftarrow 0 \end{array} $	1,T,H,S,V,N,Z,C 0,T,H,S,V,N,Z,C	1
SES		年割り込み禁止 符号フラグを設定(1)	1 ← 0 S ← 1	I,T,H,\$,V,N,Z,C	1
CLS		符号フラグを解除(<mark>0</mark>)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	V ← 1	I,T,H,S,¥,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	V ← 0	I,T,H,S,O,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLT SEH		一時フラグを解除(0) ハーフキャリー フラグを設定(1)	T ← 0 H ← 1	I, 0 ,H,S,V,N,Z,C 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		ウォッチト゛ック゛ 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	注文符号	(注2)	外囲器	動作範囲
		ATmega8L-8AU		32A	
		ATmega8L-8AUR	(注3)	32A	
8	2.7~5.5V	ATmega8L-8PU		28P3	
		ATmega8L-8MU		32M1-A	
		ATmega8L-8MUR	(注3)	JZIVII A	工業用
		ATmega8-16AU		32A	(-40°C∼85°C)
		ATmega8-16AUR	(注3)	32A	
16	4.5∼5.5V	ATmega8-16PU		28P3	
		ATmega8-16MU		32M1-A	
		ATmega8-16MUR	(注3)	JZIVII A	
		ATmega8L-8AN		32A	
		ATmega8L-8ANR	(注3)	32A	
8	2.7~5.5V	ATmega8L-8PN		28P3	
		ATmega8L-8MN		32M1-A	
		ATmega8L-8MNR	(注3)	52WII A	工業用 (注4)
		ATmega8-16AN		32A	(-40°C∼105°C)
		ATmega8-16ANR	(注3)	32A	
16	4.5∼5.5V	ATmega8-16PN		28P3	
		ATmega8-16MN		32M1-A	
		ATmega8-16MNR	(注3)	JZWIT ⁻ A	

注: このデバイスはウェハー(チップ単体)形状でも供給できます。最低数量と詳細な注文情報については最寄のAtmel営業所へお問い合わせください。

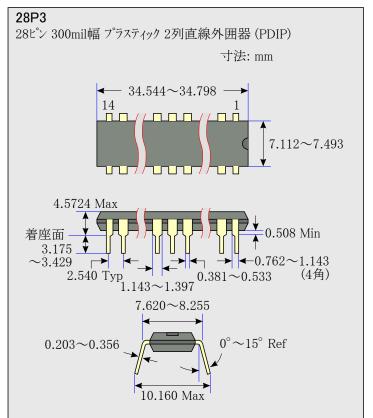
注2: 有害物質使用制限に関する欧州指令(RoHS指令)適合の鉛フリー製品。またハロゲン化合物フリーで完全に安全です。

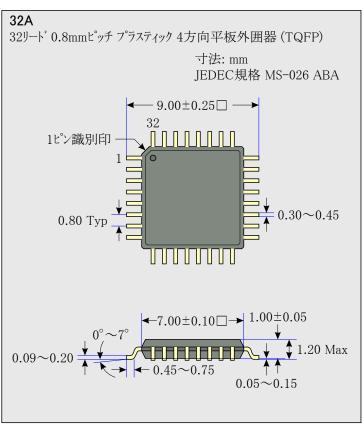
注3: テープとリール。

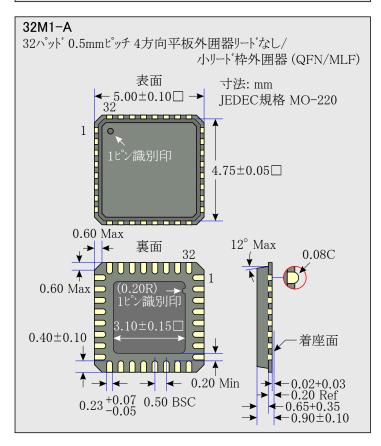
注4: 105℃の特性仕様をご覧ください。

外囲器形式							
32A	32リート、1.0mm厚 プラスティック 4方向平板外囲器 (TQFP)						
28P3	28ピン 300mil幅 プラスティック 2列直線外囲器 (PDIP)						
32M1-A	32パッド5×5×1mm 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)						

外囲器情報









データシート改訂履歴

以下は本データシートの変更記録です。全ての頁番号は本データシートが基準です。

改訂2486B-2001年12月から2486C-2002年3月への変更

1. 2線直列インターフェース(TWI)章の更新

ハプワーダウン動作でのTWIの使用と低TWRR値での主装置としてのTWI使用に関する詳細が追加されました。

109頁の「ビット速度生成部」の最後に注意事項が追加されました。

110頁の「アトーレス比較一致部」の最後に説明が追加されました。

2. 発振校正値(OSCCAL)の説明更新

2,4,8MHz発振選択用の発振校正値バイトの利用法が説明されていませんでした。これを次の項目に追加しました。

19頁の「OSCCAL - 発振校正値レジスタ」と145頁の「校正バイト」の記載が改善されました。

3. いくつかの試験条件と特性値の追加

次の表や頁内でいくつかのTBDを削除しました。

16頁の**表3**.、23頁の**表15**.、25頁の**表16**.、26頁の**表17**.、と156頁の**"DC特性** TA=−40~85℃, VCC= 2.7~5.5V (特記事項を除く)"、157頁の**表98**.、159頁の**表101**.。

4. プログラミングの図の更新

AVCCがプログラミング中に接続しなければならないことを反映し、146頁の図104.と154頁の図112.が更新されました。

5. RESETピン禁止または外部発振器選択時の並列プログラミングへの移行法の説明追加

147頁の「プログラミング・動作への移行」の項に注意事項が追加されました。

改訂2486C-2002年3月から2486D-2002年3月への変更

1. 起動待機時間代表値の更新

次の表が更新されました。

17頁の表5.「クリスタル発振器選択時の起動待機時間」、18頁の表6.「低周波数クリスタル発振器選択時の起動待機時間」、18頁の表8.「外部RC発振器選択時の起動待機時間」、19頁の表12.「外部クロック信号選択時の起動待機時間」。

2. 163頁の「代表特性」の追加

改訂2486D-2002年3月から2486E-2002年6月への変更

1. いくつかの検査制限と特性データの更新

次の表が更新されました。

23頁の表15.「Jセット特性」、25頁の表16.「内蔵基準電圧特性」、156頁の「DC特性」、160頁の表103.「A/D変換特性」。

2. 外部クロック周波数関係の変更

19頁の「外部クロック信号」の最後に説明が追加されました。

157頁の表99.「外部クロック特性」に周期間変更値が追加されました。

3. TWI節での更新

112頁の表65.「TWIL'ット速度前置分周選択」とTWIL'ット速度前置分周器の使用に関してが増加しました。

改訂2486E-2002年6月から2486F-2002年7月への変更

- 1. 34頁の「デジタル入力許可と休止形態」内を変更
- 2. 37頁の「MOSI/OC2 ポートB ビット3」にOC2説明追加
- 3. 次の表を更新

87頁の表51.「CPOL,CPHA機能動作」、103頁の表59.「UCPOL選択」、126頁の表72.「アナログ比較器反転入力選択」、129頁の表73.「A/D変換時間」、133頁の表75.「A/D入力チャネル選択」、143頁の表84.「図103.で使った各変数説明とZポインタの配置」。

- 4. 151頁の「校正バイの読み出し」を変更
- 5. 相互参照を正しく修正

改訂2486F-2002年7月から2486G-2002年9月への変更

1. 160頁の表103.「A/D変換器特性」の更新

改訂2486G-2002年9月から2486H-2002年9月への変更

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2486H-2002年9月から2486I-2002年9月への変更

1. 215頁で改訂D,E,Fの障害情報を追加



改訂2486I-2002年9月から2486J-2003年2月への変更

- 1. 16頁の「非同期タイマ クロックーclk_{ASY}」の記述を改善
- 2. 存在しない「多目的発振器」、「32kHzクリスタル発振器」応用記述の参照を削除
- 3. 60頁の図38.でOCn波形を修正
- 4.「タイマ/カウンタ1」で各種の微小修正
- 5.「2線直列インターフェース(TWI)」で各種の微小修正
- 6. 140頁の「ページー時緩衝部の設定 (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
- 7. ADHSMを完全に削除
- 8. 14頁に「パワーダウン休止動作中のEEPROM書き込み」項目を追加
- 9. 4頁のXTAL1とXTAL2記述が既に「ポートB/XTAL1/XTAL2/TOSC1/TOSC2」の一部として記述されているために削除
- 10. 159頁「SPIタイミング特性」下の表を改善、155頁「直列プログラミング特性」下の表を削除
- 11. 38頁の「ポートCの交換機能」でPC6を修正
- **12**. 36頁の「ポートBの交換機能」でPB6とPB7を修正
- **13**. 104頁の「**ホーレート設定例**」で230.4Mbpsを230.4kbpsに修正
- 14. 75頁の「位相基準PWM動作」でタイマ/カウンタ2に対するPWMの対称性についての情報を追加
- 15. 109頁の図76.でアクセス可能なレジスタの周囲に太線を追加
- 16. 140頁「ペ-ジ書き込みの実行」の未使用Zポインタ ビットに対する「無効」を「0書き込み」に変更
- **17**. 145頁の表87.でRSTDISBLヒュース、についての注意を追加
- 18. 210頁の「外囲器情報」を更新

改訂2486J-2003年2月から2486K-2003年8月への変更

- 1. 23頁の表15.でVBOT値を更新
- 2. 160頁の「A/D変換器特性」を更新
- 3. 163頁の「代表特性」を更新
- 4. 215頁の「障害情報」を更新

改訂2486K-2003年8月から2486L-2003年10月への変更

- 1. 暫定とTBDを削除
- 2. ピン名ICPをICP1に改名
- 3. CALLとJMP命令を削除
- **4**. 23頁の表15.のt_{RST}、25頁の表16.のV_{BG}、157頁の表100、159頁の表102を更新
- 5. 18頁の表10.直前文章置換、19頁「タイマ/カウンタ用発振器」のXTAL1/2とCKOPT関係文追加
- 6. 27頁「ウォッチト、ック、タイマ設定変更の時間制限手順」のコート、例を更新
- 7. 36頁「特殊I/O機能レジスタ」のビット4 ADHSMを削除
- 8. 139頁の図103.に注追加
- 9. 154頁「**直列プログラミング手順**」の第4項目を更新
- 10. 155頁表97.にtwp FUSEを追加、表98.の校正バイ読み出し第3バイを更新
- **11**. 156頁の「**絶対最大特性**」と「**DC特性**」を更新

改訂2486L-2003年10月から2486M-2003年12月への変更

1. 18頁「校正付き内蔵RC発振器」を更新

改訂2486M-2003年12月から2486N-2004年9月への変更

- 1. 2頁の「ピン配置」でQFN/MLF外囲器に注意を追加
- 2. 25頁の「内部基準電圧特性」表を更新
- 3. 156頁の「DC特性」を更新
- 4. ADC4とADC5は10ビット精度を支援します。これを反映するように文書が更新されました。
 - 127頁の「A/D変換器」で特徴を更新
 - 160頁の「A/D変換器特性」を更新
- 5. 18頁の「外部RC発振器」から「外部RC発振器応用記述」への参照を削除

改訂2486N-2004年9月から2486O-2004年10月への変更

1. 「アナロケ GND」を削除し、「GND」に置換



- 2. 18頁の表7.、23頁の表15.、157頁の表100.を更新
- 3. 18頁の「校正付き内蔵RC発振器」を1MHz既定値で更新
- 4. 146頁の表89.と表90.を145頁の新規項目「ページ容量」へ移動
- **5**. 138頁の「SPMCR SPM命令制御レジスタ」でビット4に対する記述を更新
- 6. 209頁の「注文情報」を更新

改訂2486O-2004年10月から2486P-2006年2月への変更

- 1. 5頁の「資料」を追加
- 2. 19頁の「外部クロック信号」を更新
- 3. 83頁の「**直列周辺インターフェース (SPI)**」を更新
- 4. 92頁の「USART初期化」でコート 例を更新
- 5. 109頁の「**ビット速度発生器**」で注を更新
- 6. 155頁の表98.を更新
- 7. 160頁の表103.で注を更新
- 8. 215頁の「障害情報」を更新

改訂2486P-2006年2月から2486Q-2006年10月への変更

- 1. 19頁の「タイマ/カウンタ用発振器」を更新
- 2. 60頁の「高速PWM動作」を更新
- **3**. 92頁の「**USART初期化**」でコート 例を更新
- 4. 64頁の表37.、65頁の表39.、77頁の表42.,表44.、155頁の表98.を更新
- 5. 215頁の「障害情報」を更新

改訂2486Q-2006年10月から2486R-2007年7月への変更

- 1. 1頁の「内蔵周辺機能」で誤植を修正
- 2. 25頁の表16.を更新
- 3. 133頁の表75.を更新
- 4. 141頁の表81.に(注)文章を追加
- 5. 156頁「DC特性」の注項で冗長部削除と誤植更新

改訂2486R-2007年7月から2486S-2007年8月への変更

- 1. 1頁の「特徴」を更新
- 2. 5頁に「データ保持力」を追加
- 3.85頁の「従装置動作」を更新
- 4. 215頁の「障害情報」を更新

改訂2486S-2007年8月から2486T-2008年5月への変更

- 1. 155頁の表98.を更新
- 2. 209頁の「注文情報」を更新
 - 一般用注文符号を削除
 - ・鉛フリーでない外囲器選択を削除

改訂2486T-2008年5月から2486U-2008年8月への変更

1. ICC代表値で156頁の「DC特性」を更新

改訂2486U-2008年8月から2486V-2009年5月への変更

- 1. 215頁の「障害情報」を更新
- 2. Atmelの新住所で最終頁を更新

改訂2486V-2009年5月から2486W-2010年2月への変更

1. VINT最大値(2.9V)で160頁の「A/D変換器特性」を更新

改訂2486W-2010年2月から2486X-2010年6月への変更

1. 新VOL最大値(0.9Vと0.6V)で156頁の「**DC特性**」を更新



改訂2486X-2010年6月から2486Y-2010年10月への変更

- 1. 140頁の「SPM命令によるページ消去の実行」に注を追加
- 2. 159頁の表102.内の最大上昇/下降時間を1.6nsから1600nsに修正
- 3. 多数のリンクを更新/修正、いくつかの新しいリンクを追加
- 4. 新規格に従って最終頁を更新

改訂2486Y-2010年10月から2486Z-2011年2月への変更

- 1. 新しいAtmelの品質様式指針に従ってデータシートを更新
- 2. 209頁の「注文情報」を更新、"テープとリール"品に関する注文情報を追加

改訂2486Z-2011年2月から2486AA-2013年2月への変更

- 1. Atmelの新品質形式指針に従ってデータシートを更新
- 2. 内容から"チップ。上デバッグ"への参照を削除
- **3**. 161頁に「**電気的特性 -** (TA=-40℃~105℃)」を追加
- **4**. 188頁に「**代表特性** (TA=-40℃~105℃)」を追加
- 5. 209頁の「注文情報」を更新



障害情報

この章の改訂番号はATmega8デバイスの改訂版を参照してください。

改訂D~I.M

● 初回アナログ比較器変換が遅らされるかもしれない	D ∼ I,M
● 非同期タイマでタイマ レジスタ書き込み時に割り込みが失われる可能性	D ∼ I,M
● 32kHz発振器がタイマ/カウンタ2の非同期クロックに使われるとき、CKOPTがXTALn/TOSCnピンに内部容量を許可しない	D ∼ I,M
● 識票が直列プログラミング動作で消去されるかもしれない	D ∼ I,M
● EEREL ットを設定(1)するためのSTまたはSTS使用によるEEPROM読み込みが予期せぬ割り込み要求を起動する	D ∼ I,M

1. 初回アナログ比較器変換が遅らされるかもしれない(D~I.M)

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かかるでしょう。 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。

2. 非同期タイマ/カウンタでタイマ/カウンタ レジスタ書き込み時に割り込みが失われる可能性(D~I,M)

非同期タイマ/カウンタ(TCNTx)が\$00の時に非同期タイマ/カウンタのクロックに同期化されるタイマ/カウンタのレシ、スタが書かれると、その割り込みは失われます。

対策/対処

非同期タイマ/カウンタ制御レジ、スタ(TCCRx)、非同期タイマ/カウンタ(TCNTx)レジ、スタ、または非同期タイマ/カウンタ比較レジ、スタ(OCRx)を書く前に、タイマ/カウンタ(TCNTx)レジ、スタが\$00と\$FFのどちらの値でもないことを常に調べてください。

3. 32kHz発振器がタイマ/カウンタ2の非同期クロックに使われるとき、CKOPTがXTALn/TOSCnピンに内部容量を許可しない(D~I,M)

内蔵RC発振器が主クロック元として使われるとき、XTAL1/TOSC1とXTAL2/TOSC2間への32kHz発振器接続により、タイマ/カウンタ2を非同期で走行できます。けれども、内蔵RC発振器が主クロック元として選ばれると、CKOPTヒューズはXTAL1/TOSC1とXTAL2/TOSC2の内部容量を制御しません。XTAL1/TOSC1とXTAL2/TOSC2に容量(コンデンサ)が接続されていない以上、この発振器の安定な動作は保証されません。

対策/対処

XTAL1/TOSC1とXTAL2/TOSC2に20~36pFの範囲で外部コンデンサを使ってください。これは改訂Gで、内蔵RC発振器が主クロック元として選ばれる時も、CKOPTヒュースが内部容量を制御するように修正されます。改訂GではCKOPT=0(プログラム)がXTAL1とXTAL2の内部容量を許可します。旧版と改訂G間の互換性を望むお客様にあってはCKOPTが非プログラム(1)にされるのを保証しなければなりません。

4. 識票が直列プログラミング動作で消去されるかもしれない(D~I,M)

識票バイトがチップ消去命令完了前に読まれた場合、識票はデバイス識別子と校正バイトの消滅を伴い消去されるかもしれません。デ バイスが内蔵RC発振器で動いているなら、これは特に重要です。

対策/対処

次の命令を適用する前にチップ消去命令が既に終えているのを確認してください。

5. EEREL ットを設定(1)するためのSTまたはSTS使用によるEEPROM読み込みが予期せぬ割り込み要求を起動する(D~I,M)

EEPROM制御レジスタ(EECR)のEEPROM読み込み許可(EERE)ビットを設定(1)するためのSTまたはSTS命令使用によるEEPROM読み込みが予期せぬEEPROM割り込み要求を起動します。

対策/対処

EECR内のEEREを設定(1)するのにOUTまたはSBIを常に使ってください。

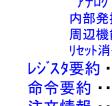


目次

タイマ/カウンタ0とタイマ/カウンタ1の前置分周器 ・・ 48
16ビット タイマ/カウンタ1 ・・・・・・・・・ 50
概要
16ビット レシ スタのアクセス ・・・・・・・・ 52
タイマ/カウンタのクロック ····· 54
計数器部 ······· 54
捕獲入力部 · · · · · · · 55
比較出力部 56
比較一致出力部 ・・・・・・・・ 58
動作種別 ・・・・・・・・ 59 タイマ/カウンタのタイミング ・・・・・・・・・ 63
91 マ/カウンタロタイミング
8ビット タイマ/カウンタ2 (PWM,非同期動作) ・・・・ 69 概要 ・・・・・・・・・・・ 69
(城安
計数器部 ・・・・・・・・・・ 70
比較出力部 ・・・・・・・・ 71
比較一致出力部 ・・・・・・・・・・・・・・・ 72
動作種別
タイマ/カウンタのタイミング・・・・・・・・・ 76
8ビット タイマ/カウンタ2用レジスタ ・・・・・・・・ 77
タイマ/カウンタ2の非同期動作 ・・・・・・・・ 80
タイマ/カウンタ2の前置分周器 ・・・・・・・ 82
SPI (直列周辺インターフェース) ・・・・・・・ 83
SSL ² ンの機能 ······ 85
データ転送形式 ・・・・・・・・・・・ 87
USART 88
概要 88
クロック生成・・・・・・・90
フレーム形式 ・・・・・・・・・・・・・・・・・・・・・・・91
USARTの初期化 ・・・・・・・・・・ 92 USARTのデータ送信 ・・・・・・・・・ 93
USARTのデータ送信・・・・・・・・・・・・・・・・・・93 USARTのデータ受信・・・・・・・・・・・・・94
非同期受信 ・・・・・・・・・ 97
複数プロセッサ通信動作・・・・・・・・・・・・99
UBRRH/UCSRCレジスタのアクセス ・・・・・・ 100
USART用レジスタ・・・・・・・・ 101
ボーレート設定例 ・・・・・・・・・・・・・・・・ 104
2線直列インターフェース (TWI) ・・・・・・・・ 106
特徴 ・・・・・・・・・・・・・ 106
2線直列インターフェース バスの定義 ・・・・・・・ 106
データ転送とフレーム形式・・・・・・・・ 106
複数主装置バスシステムの調停と同期・・・・・・108
TWI部の概要 · · · · · · · · · · · · · · · · · 109
TWI用レジスタ・・・・・・・・・・・111
TWIの使用法 ・・・・・・・・・・ 113 転送種別 ・・・・・・・・・・ 115
複数主装置システムでのバス競合と調停・・・・・ 124
アナログ :比較器 ・・・・・・・・・・・・・・ 125
7,747,比較器入力選択 ・・・・・・・・・・ 126
A/D変換器 ····································
特徴 ······ 127
111 124
操作 ······ 128
操作 ・・・・・・・・・ 128 変換の開始 ・・・・・・・・・・ 128
操作 ・・・・・・・・・・・ 128 変換の開始 ・・・・・・・・・・ 128 前置分周と変換タイミング ・・・・・・・・・・・・ 129 チャネル変更と基準電圧選択 ・・・・・・・・ 130
操作 ・・・・・・・・・・ 128 変換の開始 ・・・・・・・・・ 128 前置分周と変換タイミング・・・・・・・・・・ 129



A/D変換用レジスタ ・・・・・・・・・・・・・ 13	3
ブートローダ支援 - RWW 自己プログラミング・・・ 13	5
特徴 ······ 13	5
フラッシュ メモリの応用領域とブート ローダ領域 ・・・ 13	
書き中に読みが可能な領域と不能な領域・・13	
ブートローダ施錠ビット・・・・・・・13	
ブートローダブログラムへの移行・・・・・・・・13	
自己プログラミングでのフラッシュ アドレス指定・・・・・ 13	9
7ラッシュ メモリの自己プログラミング・・・・・・ 13 メモリ プログラミング・・・・・・・・・・・・・・・・・ 14	9
メモリ ノログ フミング ・・・・・・・・・・・・・・・・ 14	4
プログラム メモリとデータ メモリ用施錠ビット・・・・・・ 14 識票バイト・・・・・・・・・・・・・・・・・・ 14	
識票バイト ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	
*X	
へ 一ジ 容量 ・・・・・・・・・・・ 14	
並列プログラミング・・・・・・・・・・・・・・・・・ 14	
並列プログラミング手順 ・・・・・・ 14	
直列プログラミング・・・・・・15	
直列プログラミング手順・・・・・・・・・ 15	
雷気的特性 - (TA=40°C~85°C) • • • • • • 156	
絶対最大定格 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	
DC特性 ························· 15	6
外部クロック特性 ・・・・・・・・・ 15	7
2線直列インターフ ェース特性 ・・・・・・・・・ 15	8
3 1 7 1 2 7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
A/D変換器特性 •••••• 16	
電気的特性 $- (TA=40^{\circ}C \sim 105^{\circ}C) \cdots 16$	
絶対最大定格 · · · · · · · · · · · · · · · · · 1 6	
	1
DC特性 ········ 16	
代表特性 - (TA=40℃~85℃) · · · · · · · · 165	3
代表特性 - (TA=40°C~85°C) · · · · · · · · · · · 165 活動動作消費電流 · · · · · · · · · 16	3
代表特性 - (TA=40°C~85°C) ・・・・・・・ 165 活動動作消費電流 ・・・・・・・・ 165 アイトル動作消費電流 ・・・・・・・・ 16	3 3 5
代表特性 - (TA=40℃~85℃) 165 活動動作消費電流 16 7イドル動作消費電流 16 ハプワーダウン動作消費電流 16	3 3 5 8
代表特性 - (TA=40℃~85℃) 165 活動動作消費電流 16 7イドル動作消費電流 16 ハプワーダウン動作消費電流 16	3 3 5 8
代表特性 - (TA=40℃~85℃) 165 活動動作消費電流 16 7イドル動作消費電流 16 ハプワーダウン動作消費電流 16	3 3 5 8 9
代表特性 - (TA=40°C~85°C)	3 5 8 8 9
代表特性 - (TA=40°C~85°C)	3 5 8 9 1 3
代表特性 - (TA=40°C~85°C)	3 5 8 8 9 1 3 5
代表特性 - (TA=40°C~85°C)	3 5 8 8 9 1 3 5
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 アイトル動作消費電流 - 16 パワーをずうン動作消費電流 - 16 パワーセーブ動作消費電流 - 16 スタンハイ動作消費電流 - 16 とシュプルアップ - 17 ピン駆動能力 - 17 ピン駆動能力 - 17 ピン 閾値とヒステリシス - 17 低電圧検出器(BOD)閾値と - 7ナログ・比較器変位(オフセット)	3 3 5 8 9 1 3 5 8 0
代表特性 - (TA=40℃~85℃) - 165 活動動作消費電流 - 16 アイトル動作消費電流 - 16 ハプワータ・ウン動作消費電流 - 16 ハプワーセーフ・動作消費電流 - 16 スタンハ・イ動作消費電流 - 16 とシュアルアップ - 17 ヒン駆動能力 - 17 ヒン駆動能力 - 17 低電圧検出器(BOD)閾値と アナログ・比較器変位(オフセット) - 17 内部発振器周波数 - 18 周辺機能部消費電流 - 18	3 3 5 8 8 9 1 3 5 8 0 4
代表特性 - (TA=40℃~85℃) - 165 活動動作消費電流 1674トル動作消費電流 1674トル動作消費電流 16 パワーセーブ動作消費電流 16 スタンハイ動作消費電流 16 ピンプルアップ 17 ピン駆動能力 17 ピン駆動能力 17 ピン 閾値とヒステリシス 17 低電圧検出器(BOD)閾値と 177中グ・比較器変位(オフセット) 内部発振器周波数 18周辺機能部消費電流 18リセット消費電流とリセット パルス幅 18	$ \begin{array}{c} 3 \\ 3 \\ 5 \\ 8 \\ 9 \\ 1 \\ 3 \\ 5 \\ 8 \\ 0 \\ 4 \\ 7 \end{array} $
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 アイトル動作消費電流 - 16 パワータ・ウン動作消費電流 - 16 パワーセーブ・動作消費電流 - 16 スタンハ・イ動作消費電流 - 16 と゚ンプルアップ - 17 ピン駆動能力 - 17 ピン駆動能力 - 17 ピン関値とヒステリシス - 17 低電圧検出器(BOD)閾値と - 17 低電圧検出器(BOD)閾値と - 17 内部発振器周波数 - 18 周辺機能部消費電流 - 18 リセット消費電流とリセット パルス幅 - 18 イ・素性性 - (TA-40°C~105°C) - 1850	3 3 5 8 8 9 1 3 5 8 0 4 7 8
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 アイトル動作消費電流 - 16 パワータ・ウン動作消費電流 - 16 パワーセーブ・動作消費電流 - 16 スタンハ・イ動作消費電流 - 16 と゚ンプルアップ - 17 ピン駆動能力 - 17 ピン駆動能力 - 17 ピン関値とヒステリシス - 17 低電圧検出器(BOD)閾値と - 17 低電圧検出器(BOD)閾値と - 17 内部発振器周波数 - 18 周辺機能部消費電流 - 18 リセット消費電流とリセット パルス幅 - 18 イ・素性性 - (TA-40°C~105°C) - 1850	$3 \\ 3 \\ 5 \\ 8 \\ 8 \\ 9 \\ 1 \\ 3 \\ 5 \\ 8 \\ 0 \\ 4 \\ 7 \\ 8 \\ 8$
代表特性 - (TA=40℃~85℃) - 165 活動動作消費電流 - 16 7イトル動作消費電流 - 16 16 16 16 16 16 16 16 16 16 16 16 16	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9
代表特性 - (TA=40℃~85℃) - 165 活動動作消費電流 - 16 7イトル動作消費電流 - 16 16 16 16 16 16 16 16 16 16 16 16 16	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 パワーセーブ動作消費電流 - 16 パワーセーブ動作消費電流 - 16 スタンハ・イ動作消費電流 - 16 と゚ンプルアップ - 17 と゚ン駆動能力 - 17 と゚ン駆動能力 - 17 低電圧検出器(BOD)閾値と アナログ・比較器変位(オフセット) 内部発振器周波数 - 18 周辺機能部消費電流 - 18 リセット消費電流とリセット パルス幅 - 18 リセット消費電流 - 18 大き特性 - (TA=40°C~105°C) - 185 活動動作消費電流 - 18 アイトル動作消費電流 - 19 と゚ンプルアップ - 19 ピンプルアップ - 19	335889135804788911
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 パワーセーブ動作消費電流 - 16 パワーセーブ動作消費電流 - 16 スタンハ・イ動作消費電流 - 16 と゚ンプルアップ - 17 と゚ン駆動能力 - 17 と゚ン駆動能力 - 17 低電圧検出器(BOD)閾値と アナログ・比較器変位(オフセット) 内部発振器周波数 - 18 周辺機能部消費電流 - 18 リセット消費電流とリセット パルス幅 - 18 リセット消費電流 - 18 大き特性 - (TA=40°C~105°C) - 185 活動動作消費電流 - 18 アイトル動作消費電流 - 19 と゚ンプルアップ - 19 ピンプルアップ - 19	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 アイトル動作消費電流 - 16 パワーを「ウン動作消費電流 - 16 パワーセーブ動作消費電流 - 16 スタンパイ動作消費電流 - 16 と゚ンプルアップ - 17 ピン駆動能力 - 17 ピン駆動能力 - 17 ピン駆動能力 - 17 佐電圧検出器(BOD)閾値と - 7ナログ・比較器変位(オフセット) 内部発振器周波数 - 18 周辺機能部消費電流 - 18 リセット消費電流 - 18 リセット消費電流 - 18 イ表特性 - (TA=40°C~105°C) - 185 活動動作消費電流 - 18 アイト・ル動作消費電流 - 18 アイト・ル動作消費電流 - 18 アイト・ル動作消費電流 - 19 ピンアルアップ - 19 ピン駆動能力 - 19	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 5
代表特性 - (TA=40°C~85°C) 16: 活動動作消費電流 16 アイトル動作消費電流 16 ハプワーセーブ動作消費電流 16 ハプワーセーブ動作消費電流 16 スタンハイ動作消費電流 16 ピップルアップ 17 ピッ駆動能力 17 ピッ駆動能力 17 世の 閾値とヒステリシス 17 低電圧検出器(BOD)閾値と アナログ・比較器変位(オフセット) 内部発振器周波数 18 周辺機能部消費電流 18 リセット消費電流とリセット ハプルス幅 18 代表特性 - (TA=40°C~105°C) 18: 活動動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 19 ピップルアップ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8
代表特性 - (TA=40°C~85°C) 16: 活動動作消費電流 16 アイトル動作消費電流 16 ハプワーセーブ動作消費電流 16 ハプワーセーブ動作消費電流 16 スタンハイ動作消費電流 16 ピップルアップ 17 ピッ駆動能力 17 ピッ駆動能力 17 世の 閾値とヒステリシス 17 低電圧検出器(BOD)閾値と アナログ・比較器変位(オフセット) 内部発振器周波数 18 周辺機能部消費電流 18 リセット消費電流とリセット ハプルス幅 18 代表特性 - (TA=40°C~105°C) 18: 活動動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 18 アイトル動作消費電流 19 ピップルアップ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 16 16 16 16 16 16 17 17 17 18 18 18 19 19 19 19 19 19 19 19 19 19 19 19 19	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 16 16 16 16 16 17 17 18 17 18 18 18 19 19 19 19 19 18 18 18 19 19 19 19 19 19 19 19 19 19 19 19 19	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8 0 3 5
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 16 16 16 16 16 17 17 18 17 18 18 18 19 19 19 19 19 18 18 18 19 19 19 19 19 19 19 19 19 19 19 19 19	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8 0 3 5
代表特性 - (TA=40°C~85°C) - 165 活動動作消費電流 - 16 74トル動作消費電流 - 16 16 16 16 16 16 17 17 18 17 18 18 18 19 19 19 19 19 18 18 18 19 19 19 19 19 19 19 19 19 19 19 19 19	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8 0 3 5 6
代表特性 - (TA=40°C~85°C) - 16: 活動動作消費電流 16 アイドル動作消費電流 16 パワータウン動作消費電流 16 パワーセーブ動作消費電流 16 スタンバイ動作消費電流 16 ピンプルアップ 17 ピン駆動能力 17 ピン駆動能力 17 ピン駆動能力 17 低電圧検出器(BOD)閾値と アナログ比較器変位(オフセット) 内部発振器周波数 18 周辺機能部消費電流 18 リセット消費電流 18 リセット消費電流 18 アイドル動作消費電流 18 アイドル動作消費電流 18 アイドル動作消費電流 18 アイドル動作消費電流 18 アイドル動作消費電流 19 ピンプルアップ 19 ピンプルアップ 19 ピンプルアップ 19 ピンプルアップ 19 ピンプルアップ 19 ピン駆動能力 19 ピン 閾値とヒステリシス 19 低電圧検出器(BOD)閾値と アナログ比較器変位(オフセット) 内部発振器周波数 20 周辺機能部消費電流 20 リセット消費電流とリセット パルス幅 20 レジスタ要約 200	3 3 5 8 8 9 1 3 5 8 0 4 7 8 8 9 1 1 3 5 8 0 3 5 6 7



データシート改訂履歴				•	 -				•		 2	211
障害情報 •••••	 •	•	 •	•	 •	•	 •	•	•	•	 2	215





Enabling Unlimited Possibilities®

Atmel Corporation

1600 Technology Drive San Jose, CA 95110 USA

TEL (+1)(408) 441-0311 FAX (+1)(408) 487-2600

www.atmel.com

Atmel Asia Limited

Unit 01-5 & 16, 19F BEA Tower, Millennium City 5 418 Kwun Tong Road Kwun Tong, Kowloon HONG KONG

TEL (+852) 2245-6100 FAX (+852) 2722-1369

Atmel Munich GmbH

Business Campus Parking 4 D-85748 Garching b. Munich GERMANY TEL (+49) 89-31970-0 FAX (+49) 89-3194621

Atmel Japan G.K.

141-0032 東京都品川区 大崎1-6-4 新大崎勧業ピル 16F アトメル ジャパン合同会社 TEL (+81)(3)-6417-0300 FAX (+81)(3)-6417-0370

© 2013 Atmel Corporation. 不許複製 / 改訂:2486AA-AVR-02/2013

Atmel®、Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

お断り:本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイに位置する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえるtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmellこ責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© HERO 2022.

本データシートはAtmelのATmega8英語版データシート(Rev.2486AA-02/2013)の翻訳日本語版です。日本語では不自然となる重複する 形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意訳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。