



8ビット AVR マイクロコントローラ

ATmega48/V, ATmega88/V, ATmega168/V

完全 データシート

## 序説

Atmel® ATmega48/V/88/V/168/VはAVR®強化RISC構造に基づく低電力CMOS 8ビットマイクロコントローラです。単一クロック周期での強力な命令の実行により、ATmega48/V/88/V/168/VはMHz当たり1 MIPS近くの単位処理量を達成します。これは消費電力対処理速度に対するデバイスの最適化をシステム設計者に許します。

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。

## 特徴

- 高性能、低消費Atmel® AVR® 8ビット マイクロ コントローラ
- 進化したRISC構造
  - 強力な129/131命令(多くは1周期実行)
  - 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
  - 20MHz時、20MIPS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4K/8K/16Kバイト(2K/4K/8K語)フラッシュメモリ内蔵
  - 256/512/512バイトのEEPROM
  - 512/1K/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°C, 100年/25°C
  - 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
- Atmel® QTouch®ライブラリ支援
  - 容量性接触の釦、滑動部、輪
  - QTouchとQMatrix®の採取
  - 64までの感知チャネル
- 内蔵周辺機能
  - 独立した前置分周器と比較機能付き2つの8ビットタイマ/カウンタ
  - 独立した前置分周器、比較、捕獲機能付き1つの16ビットタイマ/カウンタ
  - 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)
  - 6つのPWM出力
  - 6チャネル(PDIP, QFN/MLF28), 8チャネル(TQFP, QFN/MLF32)の10ビットA/D変換器
    - 温度測定
  - 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
  - バイト対応2線直列インターフェース(Philips I<sup>2</sup>C互換)
  - 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
  - ピン変化での割り込みと起動復帰
- 特殊マイクロ コントローラ機能
  - 電源ONリセット回路と設定可能な低電圧検出器(BOD)
  - 校正可能な内蔵RC発振器
  - 外部及び内部の割り込み
  - アイドル、A/D変換雑音低減、パワーセーブ、パワーダウン、スタンバイ、拡張スタンバイの6つの低消費動作
- I/Oと外圍器
  - 23ビットの設定可能なI/O
  - 28ピンPDIP、28ピンQFN/MLF、32ピンTQFP、32ピンQFN/MLF
- 動作温度
  - -40~85°C
- 動作電圧
  - 1.8~5.5V (ATmega48V/88V/168V)
  - 2.7~5.5V (ATmega48/88/168)
- 動作速度
  - 0~4MHz/1.8~5.5V, 0~10MHz/2.7~5.5V (ATmega48V/88V/168V)
  - 0~10MHz/2.7~5.5V, 0~20MHz/4.5~5.5V (ATmega48/88/168)
- 代表消費電力 (1MHz, 1.8V, 25°C)
  - 0.3mA (活動動作)
  - 0.1µA (パワーダウン動作)
  - 0.8µA (パワーセーブ動作, 32kHz RTCを含む)

序説	1	15.1. AVRのリセット	40
特徴	2	15.2. リセット元	40
1. 概要	5	15.3. 電源ONリセット	41
2. 製品形態要約	6	15.4. 外部リセット	41
3. 注文情報	7	15.5. 低電圧検出(BOD)リセット	41
4. 構成図	9	15.6. ウォッチドッグシステムリセット	41
5. ピン配置	10	15.7. 内部基準電圧	42
5.1. ピン接続図	10	15.8. ウォッチドッグタイマ	42
5.2. ピン説明	12	15.9. リセット関係レジスタ	44
6. 入出力多重化	13	16. INT – 割り込み	46
7. 資料	14	16.1. 割り込みベクタ	46
8. データ保持力	14	16.2. 割り込みベクタ移動関係レジスタ	51
9. コード例について	14	17. EXTINT – 外部割り込み	52
10. 容量性接触感知	14	17.1. ピン変化割り込みタイミング	52
10.1. QTouchライブラリ	14	17.2. 外部割り込み用レジスタ	52
11. AVR CPU コア	15	18. 入出力ポート	56
11.1. 概要	15	18.1. 概要	56
11.2. ALU (Arithmetic Logic Unit)	15	18.2. 標準デジタル入出力としてのポート	56
11.3. ステータスレジスタ	16	18.3. 交換ポート機能	59
11.4. 汎用レジスタファイル	17	18.4. I/Oポート用レジスタ	67
11.5. スタックポインタ	17	19. TC0 – 8ビットタイマ/カウンタ0 (PWM)	70
11.6. 命令実行タイミング	18	19.1. 特徴	70
11.7. リセットと割り込みの扱い	19	19.2. 概要	70
12. AVRのメモリ	21	19.3. タイマ/カウンタのクロック	71
12.1. 概要	21	19.4. 計数器部	71
12.2. 実装書き換え可能なプログラム用フラッシュメモリ	21	19.5. 比較出力部	72
12.3. データ用SRAMメモリ	21	19.6. 比較一致出力部	73
12.4. データ用EEPROMメモリ	22	19.7. 動作種別	74
12.5. I/Oメモリ (レジスタ)	22	19.8. タイマ/カウンタのタイミング	77
12.6. メモリ関係レジスタ	24	19.9. 8ビットタイマ/カウンタ0用レジスタ	78
13. システムクロックとクロック選択	28	20. TC1 – 16ビットタイマ/カウンタ1 (PWM)	83
13.1. クロックシステムとその配給	28	20.1. 概要	83
13.2. クロック元	29	20.2. 特徴	83
13.3. クリスタル用低電力発振器	30	20.3. 構成図	83
13.4. クリスタル用全振幅発振器	30	20.4. 定義	84
13.5. 低周波数クリスタル用発振器	31	20.5. 関係レジスタ	84
13.6. 校正付き内蔵RC発振器	32	20.6. 16ビットレジスタのアクセス	85
13.7. 128kHz内部発振器	32	20.7. タイマ/カウンタのクロック	87
13.8. 外部クロック信号	33	20.8. 計数器部	87
13.9. タイマ/カウンタ用発振器	33	20.9. 捕獲入力部	88
13.10. システムクロック出力緩衝部	33	20.10. 比較出力部	89
13.11. システムクロック前置分周器	33	20.11. 比較一致出力部	90
13.12. クロック関係レジスタ	34	20.12. 動作種別	91
14. PM – 電力管理と休止形態	35	20.13. タイマ/カウンタのタイミング	95
14.1. 概要	35	20.14. 16ビットタイマ/カウンタ1用レジスタ	96
14.2. 休止形態種別	35	21. タイマ/カウンタ0と1の前置分周器	102
14.3. アイドル動作	35	21.1. 内部クロック元	102
14.4. A/D変換雑音低減動作	35	21.2. 前置分周器リセット	102
14.5. パワーダウ動作	36	21.3. 外部クロック元	102
14.6. パワーセーブ動作	36	21.4. 同期系TC前置分周器制御用レジスタ	103
14.7. スタンバイ動作	36	22. TC2 – 8ビットタイマ/カウンタ2 (PWM,非同期動作)	104
14.8. 拡張スタンバイ動作	36	22.1. 特徴	104
14.9. 電力削減レジスタ	36	22.2. 概要	104
14.10. 消費電力の最小化	37	22.3. タイマ/カウンタのクロック	105
14.11. 電力管理用レジスタ	38	22.4. 計数器部	105
15. SCRST – システム制御とリセット	40	22.5. 比較出力部	106
		22.6. 比較一致出力部	107
		22.7. 動作種別	108

22.8.	タイマ/カウンタのタイミング	111	29.4.	ソフトウェア中断点	183
22.9.	タイマ/カウンタ2の非同期動作	112	29.5.	デバッグWIREの制限	183
22.10.	タイマ/カウンタ2の前置分周器	113	29.6.	デバッグWIRE用レジスタ	184
22.11.	8ビット タイマ/カウンタ2用レジスタ	114	30.	フラッシュの自己プログラミング - 48/V	185
23.	SPI - 直列周辺インターフェース	121	30.1.	概要	185
23.1.	特徴	121	30.2.	自己プログラミングでのフラッシュ アドレス指定	185
23.2.	概要	121	30.3.	自己プログラミング用レジスタ	189
23.3.	SSピンの機能	123	31.	BTLDR - ブート ローダ支援 (RWW自己プログラミング) - 88/V/168/V	190
23.4.	データ転送形式	124	31.1.	特徴	190
23.5.	SPI用レジスタ	125	31.2.	概要	190
24.	USART	127	31.3.	フラッシュ メモリの応用領域とブート ローダ領域	190
24.1.	特徴	127	31.4.	書き中に読みが可能な領域と不能な領域	190
24.2.	概要	127	31.5.	ブート ローダ施錠ビット	192
24.3.	構成図	127	31.6.	ブート ローダ プログラムへの移行	192
24.4.	クロック生成	128	31.7.	自己プログラミングでのフラッシュ アドレス指定	193
24.5.	フレーム形式	129	31.8.	フラッシュ メモリの自己プログラミング	193
24.6.	USARTの初期化	130	31.9.	ブート ローダ関係レジスタ	199
24.7.	データ送信 - USART送信部	131	32.	MEMPROG - メモリ プログラミング	201
24.8.	データ受信 - USART受信部	132	32.1.	プログラム メモリとデータ メモリ用施錠ビット	201
24.9.	非同期受信	135	32.2.	ヒューズ ビット	202
24.10.	複数プロセッサ通信動作	137	32.3.	識票バイト	203
24.11.	ボーレート設定例	138	32.4.	校正バイト	203
24.12.	USART用レジスタ	140	32.5.	ページ容量	203
25.	USARTSPI - USARTでのSPI動作	144	32.6.	並列プログラミング	204
25.1.	特徴	144	32.7.	並列プログラミング手順	205
25.2.	概要	144	32.8.	直列プログラミング	210
25.3.	クロック生成	144	32.9.	直列プログラミング手順	210
25.4.	SPIデータ形態とタイミング	144	33.	電気的特性	213
25.5.	フレーム形式	145	33.1.	絶対最大定格	213
25.6.	データ転送	146	33.2.	DC特性	213
25.7.	USARTでのMSPIMとSPIの比較	147	33.3.	速度勾配	214
25.8.	MSPIMでのUSART用レジスタ	147	33.4.	クロック特性	214
26.	TWI - 2線直列インターフェース	148	33.5.	システムとリセットの特性	215
26.1.	特徴	148	33.6.	SPI タイミング特性	216
26.2.	2線直列インターフェース バスの定義	148	33.7.	2線直列インターフェース特性	217
26.3.	データ転送とフレーム形式	149	33.8.	A/D変換器特性	218
26.4.	複数主装置バス システムの調停と同期	150	33.9.	並列プログラミング特性	219
26.5.	TWI部の概要	151	34.	代表特性	221
26.6.	TWIの使用法	153	34.1.	活動動作消費電流	221
26.7.	転送種別	155	34.2.	アイドル動作消費電流	223
26.8.	複数主装置システムでのバス競合と調停	165	34.3.	周辺機能部供給電流	225
26.9.	TWI用レジスタ	166	34.4.	パワーダウン動作消費電流	226
27.	AC - アナログ比較器	169	34.5.	パワーセーブ動作消費電流	226
27.1.	概要	169	34.6.	スタンバイ動作消費電流	227
27.2.	アナログ比較器入力選択	169	34.7.	ピンプルアップ	227
27.3.	アナログ比較器用レジスタ	169	34.8.	ピン駆動能力	229
28.	ADC - A/D変換器	172	34.9.	ピン 閾値とヒステリシス	231
28.1.	特徴	172	34.10.	低電圧検出器(BOD)閾値と アナログ比較器変位(オフセット)	233
28.2.	概要	172	34.11.	内部発振器周波数	235
28.3.	変換の開始	173	34.12.	周辺機能部消費電流	236
28.4.	前置分周と変換タイミング	174	34.13.	リセット消費電流とリセット パルス幅	238
28.5.	チャネル変更と基準電圧選択	176	35.	レジスタ要約	239
28.6.	雑音低減機能	177	36.	命令要約	242
28.7.	A/D変換の結果	178	37.	外圍器情報	244
28.8.	温度測定	178	38.	障害情報	245
28.9.	A/D変換用レジスタ	179	39.	データシート改訂履歴	247
29.	デバッグWIRE内蔵デバッグ機能	183			
29.1.	特徴	183			
29.2.	概要	183			
29.3.	物理インターフェース	183			

## 1. 概要

Atmel AVR<sup>®</sup>コアは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。

ATmega48/V/88/V/168/Vは書き込み中読み出し可能な能力を持つ4K/8K/16Kバイトの実装書き換え可能なフラッシュメモリと256/512/512バイトのEEPROM、512/1K/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な3つのタイマ/カウンタ、内部及び外部割り込み、設定変更可能な直列USART、バイト志向の2線直列インターフェース、SPI直列ポート、8(32ピン外圍器)、6(28ピン外圍器)チャンネルの10ビットA/D変換器、設定変更可能な内部発振器付きウォッチドッグタイマ、ソフトウェアで選べる6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。

AtmelはAVRマイクロコントローラに容量性接触釦、滑動器、輪の機能を組み込むためのQTouch<sup>®</sup>ライブラリを提供します。特許権を持つ充電転移信号採取は強力な感知を提供し、接触キーの完全な反発運動報告を含み、そしてキー事象の明白な検出のための隣接キー抑制(AKS<sup>®</sup>:Adjacent Key Suppression<sup>®</sup>)技術を含みます。簡単に使えるQTouch Suiteツールチェーンはあなた自身の接触応用に対して調査、開発、そしてデバッグを許します。

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コア上ポートプログラムの実行によって再書き込みができます。ポ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。ポ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」動作により、応用領域フラッシュメモリ更新中も実行を継続します。モ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 ATmega48/V/88/V/168/V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。

ATmega48/V/88/V/168/V AVRはCコンパイラ、マクロアセンブラ、デバッグ、シミュレータ、インサーキットエミュレータ、評価キットを含む専用のプログラム及びシステム開発ツールで支援されます。

## 2. 製品形態要約

表2-1. 製品形態要約とデバイス比較

機能項目	ATmega48/V	ATmega88/V	ATmega168/V
ピン数	28/32	28/32	28/32
フラッシュメモリ容量 (バイト)	4K	8K	16K
SRAM容量 (バイト)	512	1K	1K
EEPROM容量 (バイト)	256	512	512
割り込みベクタの大きさ (命令語数/ベクタ)	1	1	2
入出力ピン数	23	23	23
SPI	1	1	1
TWI (I <sup>2</sup> C)	1	1	1
USART	1	1	1
A/D変換器 (ADC)	10ビット 15ksps	10ビット 15ksps	10ビット 15ksps
ADCチャンネル数	8	8	8
アナログ比較器 (AC)	1	1	1
8ビットタイマ/カウンタ	2	2	2
16ビットタイマ/カウンタ	1	1	1
PWMチャンネル数	4(8ビット)+2(16ビット)	4(8ビット)+2(16ビット)	4(8ビット)+2(16ビット)

ATmega88/VとATmega168/Vは真の「書き込み中の読み出し可」自己プログラミング機構を支援します。これは独立したブートロータ領域で、SPM命令はその領域からだけ実行できます。ATmega48/Vでは「書き込み中の読み出し可」動作は支援されず、独立したブートロータ領域もありません。SPM命令はフラッシュメモリ全体で実行することができます。

### 3. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号 (注2)	外圍器	動作範囲	
ATmega48/V	10 (注3)	1.8~5.5V	ATmega48V-10PU	28P3	工業用 (-40°C~85°C)	
			ATmega48V-10AU	32A		
			ATmega48V-10AUR (注4)			
			ATmega48V-10MMU	28M1		
			ATmega48V-10MMUR (注4)			
			ATmega48V-10MMH (注5)			
			ATmega48V-10MMHR (注4,5)			
	ATmega48V-10MU	32M1-A				
	ATmega48V-10MUR (注4)	20 (注3)	2.7~5.5V	ATmega48-20PU		28P3
	ATmega48-20AU			32A		
	ATmega48-20AUR (注4)					
	ATmega48-20MMU			28M1		
	ATmega48-20MMUR (注4)					
	ATmega48-20MMH (注5)					
ATmega48-20MMHR (注4,5)						
ATmega48-20MU	32M1-A					
ATmega48-20MUR (注4)						
ATmega88/V	10 (注3)	1.8~5.5V	ATmega88V-10PU	28P3	工業用 (-40°C~85°C)	
			ATmega88V-10AU	32A		
			ATmega88V-10AUR (注4)			
			ATmega88V-10MU	32M1-A		
			ATmega88V-10MUR (注4)			
	20 (注3)	2.7~5.5V	ATmega88-20PU	32A		
			ATmega88-20AU	28P3		
			ATmega88-20AUR (注4)			
			ATmega88-20MU	32M1-A		
			ATmega88-20MUR (注4)			
ATmega168/V	10 (注3)	1.8~5.5V	ATmega168V-10PU	28P3	工業用 (-40°C~85°C)	
			ATmega168V-10AU	32A		
			ATmega168V-10AUR (注4)			
			ATmega168V-10MU	32M1-A		
			ATmega168V-10MUR (注4)			
	20 (注3)	2.7~5.5V	ATmega168-20PU	28P3		
			ATmega168-20AU	32A		
			ATmega168-20AUR (注4)			
			ATmega168-20MU	32M1-A		
			ATmega168-20MUR (注4)			

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: 速度と電源電圧の関係については「速度勾配」をご覧ください。

注4: テープとリール。

注5: ニッケル・パラジウム・金(NiPdAu)メッキ仕上げリード

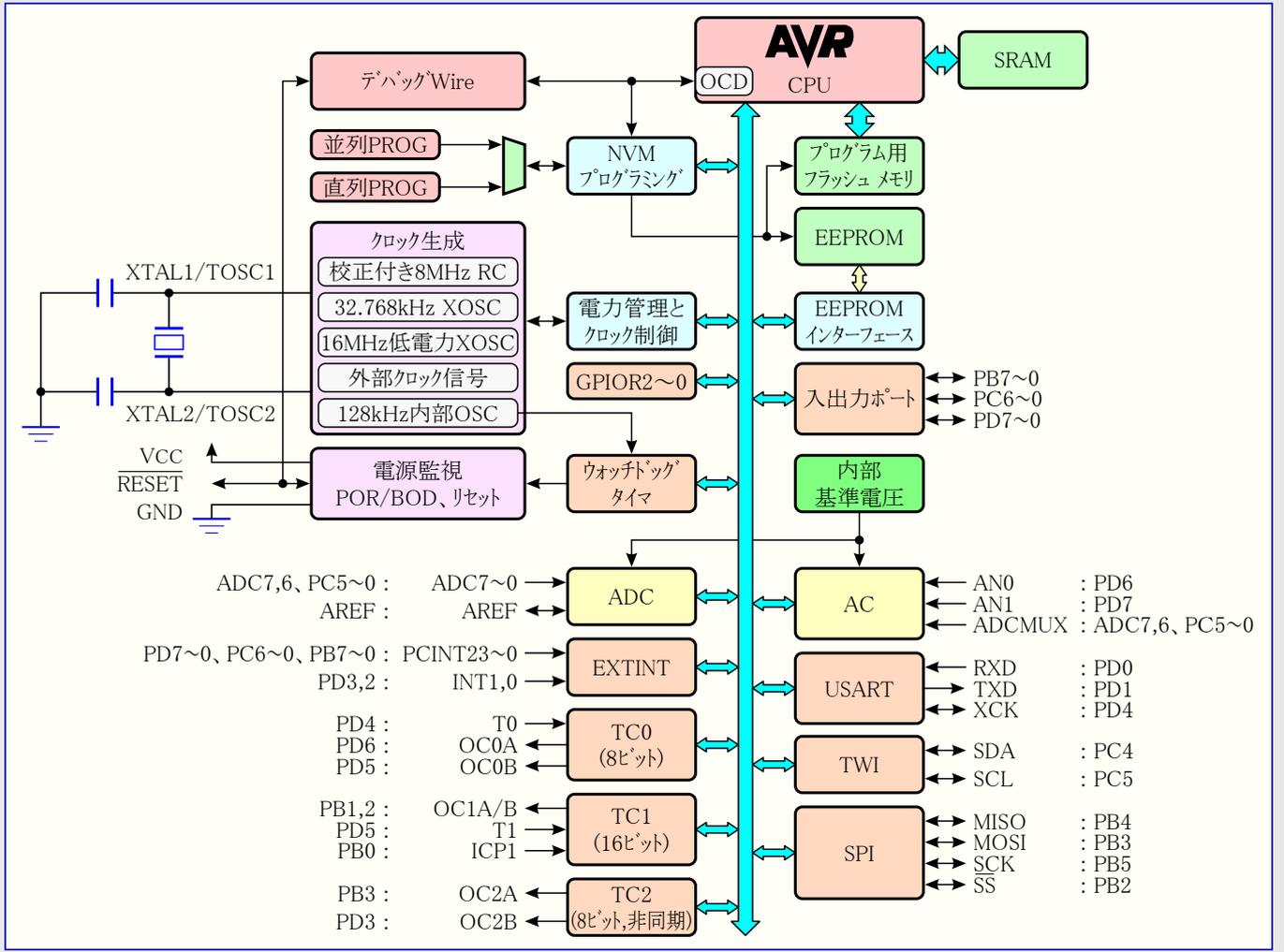
### 外圍器形式

28M1	28ピン 4×4×1mm 0.45mmピッチ 4方向平板リードなし/小リード枠外圍器 (QFN/MLF)
28P3	28ピン 300mil幅 プラスティック2列直線外圍器 (PDIP)
32A	32リード 1.0mm厚 プラスティック4方向平板外圍器 (TQFP)
32M1-A	32ピン 5×5×1mm 0.5mmピッチ 4方向平板リードなし/小リード枠外圍器 (QFN/MLF)

関連リンク [214頁の「速度勾配」](#)

## 4. 構成図

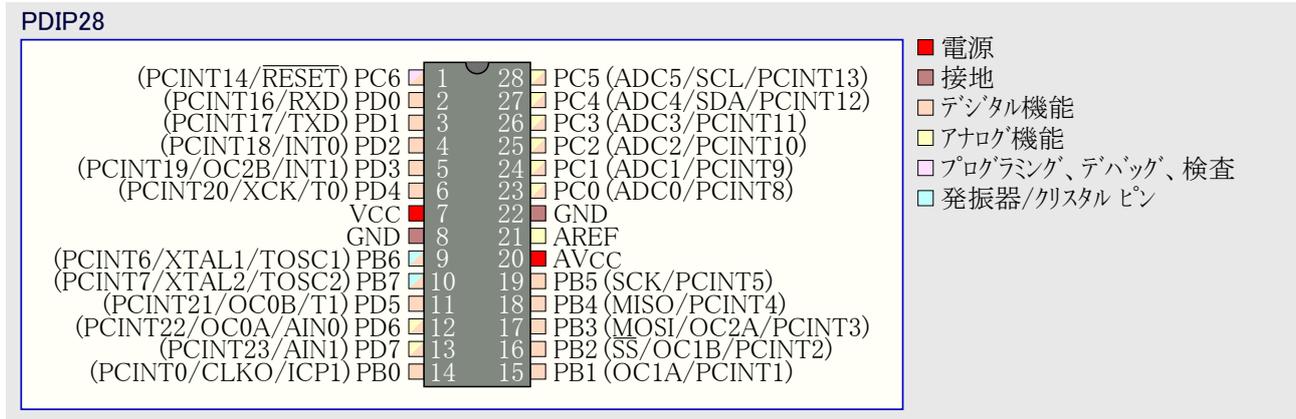
図4-1. 構成図



## 5. ピン配置

### 5.1. ピン接続図

#### 5.1.1. PDIP28

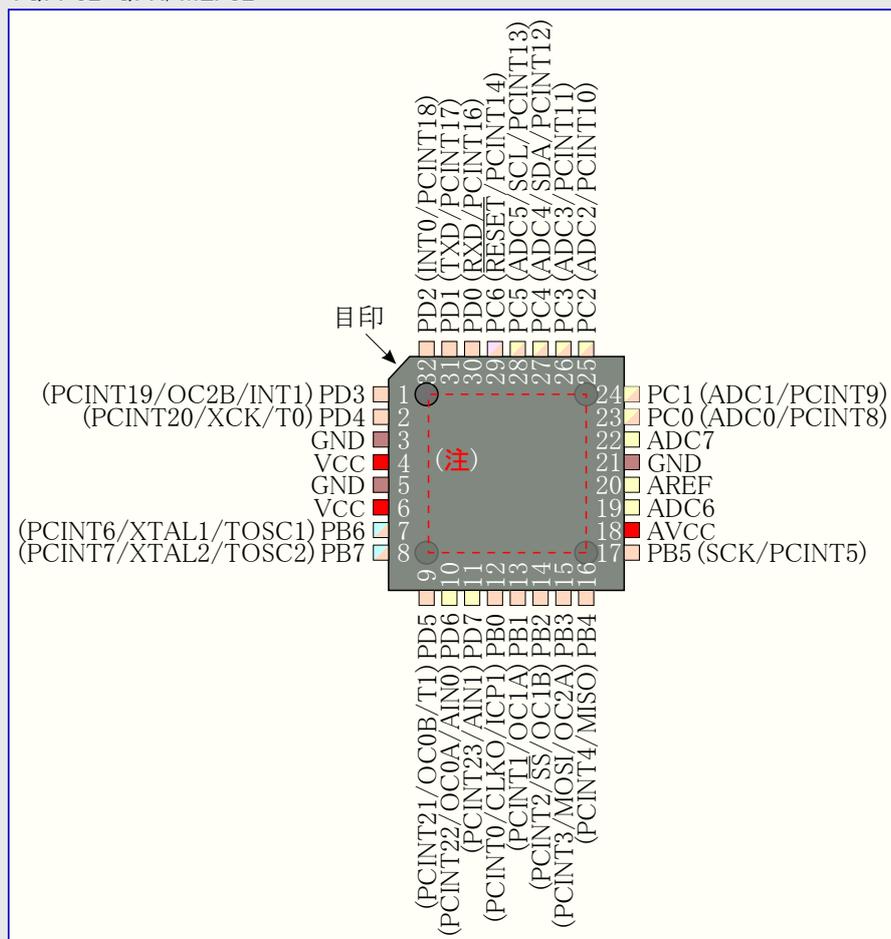


#### 5.1.2. QFN/MLF28



### 5.1.3. TQFP32とQFN/MLF32

TQFP32・QFN/MLF32



- 電源
- 接地
- デジタル機能
- アナログ機能
- プログラミング、デバッグ、検査
- 発振器/クリスタルピン

**注:** QFN/MLF外囲器底面中央の大パッドは良好な機構的安定を保障するため、GNDに半田付けされるべきです。

## 5.2. ピン概要

### 5.2.1. VCC

デジタル電源ピン。

### 5.2.2. GND

接地ピン。

### 5.2.3. PB7～PB0 (ポート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。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートBピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動作していても、ポートBピンはHi-Zになります。

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、非同期状態レジスタ(ASSR)の非同期動作(AS2)ビットが設定(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。

### 5.2.4. 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。入力の際にプルアップ抵抗が有効の場合、外部的にLowへ引き込まれたポートCピンには吐き出し電流が流れます。リセット条件が有効になると、クロックが動作していても、ポートCピンはHi-Zになります。

### 5.2.5. 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ヒューズが非プログラム(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。より短いパルスはリセットの生成が保証されません。

ポートCの各特殊機能は「ポートCの交換機能」で詳しく述べられます。

### 5.2.6. PD7～PD0 (ポートD)

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

### 5.2.7. 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。例えば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(6～4)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。

### 5.2.8. AREF

AREFはA/D変換器用アナログ基準(電圧)ピンです。

### 5.2.9. ADC7,6 (TQFP,QFN/MLF32のみ)

TQFPとQFN/MLF32外圍器での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ADCチャンネルとして扱われます。

## 6. 入出力多重化

各ピンは既定によって汎用入出力としてポートによって制御され、代わりに周辺機能の1つに割り当てることができます。

下表はポート入出力ピンに多重化される周辺機能信号を記述します。

表6-1. ポート機能多重化

ピン番号			ポートパッド	EXTINT	PCINT	ADC ・AC	OSC	T/C #0	T/C #1	USART	TWI	SPI
TQFP32/MLF32	MLF28	PDIP28										
1	1	5	PD3	INT1	PCINT19			OC2B				
2	2	6	PD4		PCINT20			T0		XCK		
4	3	7	VCC									
3	4	8	GND									
6	-	-	VCC									
5	-	-	GND									
7	5	9	PB6		PCINT6		XTAL1/TOSC1					
8	6	10	PB7		PCINT7		XTAL2/TOSC2					
9	7	11	PD5		PCINT21			OC0B	T1			
10	8	12	PD6		PCINT22	AIN0		OC0A				
11	9	13	PD7		PCINT23	AIN1						
12	10	14	PB0		PCINT0		CLKO	ICP1				
13	11	15	PB1		PCINT1			OC1A				
14	12	16	PB2		PCINT2			OC1B				$\overline{SS}$
15	13	17	PB3		PCINT3			OC2A				MOSI
16	14	18	PB4		PCINT4							MISO
17	15	19	PB5		PCINT5							SCK
18	16	20	AVCC									
19	-	-	ADC6			ADC6						
20	17	21	AREF									
21	18	22	GND									
22	-	-	ADC7			ADC7						
23	19	13	PC0		PCINT8	ADC0						
24	20	24	PC1		PCINT9	ADC1						
25	21	25	PC2		PCINT10	ADC2						
26	22	26	PC3		PCINT11	ADC3						
27	23	27	PC4		PCINT12	ADC4						SDA
28	24	28	PC5		PCINT13	ADC5						SCL
29	25	1	PC6/ $\overline{RESET}$		PCINT14							
30	26	2	PD0		PCINT16					RXD		
31	27	3	PD1		PCINT17					TXD		
32	28	4	PD2	INT0	PCINT18							

## 7. 資料

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

## 8. データ保持力

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

## 9. コード例について

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

これらのコード例はアセンブルまたはコンパイルに先立ってデバイス定義ファイルがインクルードされることが前提です。拡張I/O領域に配置したI/Oレジスタに対し、**IN**、**OUT**、**SBIS**、**SBIC**、**CBI**、**SBI**命令は拡張I/O領域へのアクセスを許す命令に置き換えられなければなりません。代表的には**SBRS**、**SBRC**、**SBR**、**CBR**命令と組み合わせた**LDS**、**STS**命令です。

## 10. 容量性接触感知

### 10.1. QTouchライブラリ

Atmel®のQTouch®ライブラリはAtmelの殆どのAVR®マイクロコントローラ上の接触感知インターフェース用の解決策を使うための単一物を提供します。QTouchライブラリはAtmel QTouchとAtmel QMatrix®採取法用の支援を含みます。

接触感知はQTouchライブラリをリンクすることによってどの応用にも容易に追加されます。これは接触チャネルと感知器を定義するために簡単なAPIの組を用いて行われ、そしてチャネル情報を取得して接触感知器の状態を決めるためにAPIを呼び出します。

QTouchライブラリは無料で以下の場所のAtmelのウェブサイトからダウンロードすることができます。

<http://www.atmel.com/technologies/touch/>

より多くの情報と実装の詳細についてはAtmelのウェブサイトからも入手可能なQTouchライブラリ使用者の手引きを参照してください。

## 11. AVR CPU コア

### 11.1. 概要

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

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

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

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

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

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

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

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

AVR構造に於けるメモリ空間は全て直線的な普通のメモリ配置です。

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

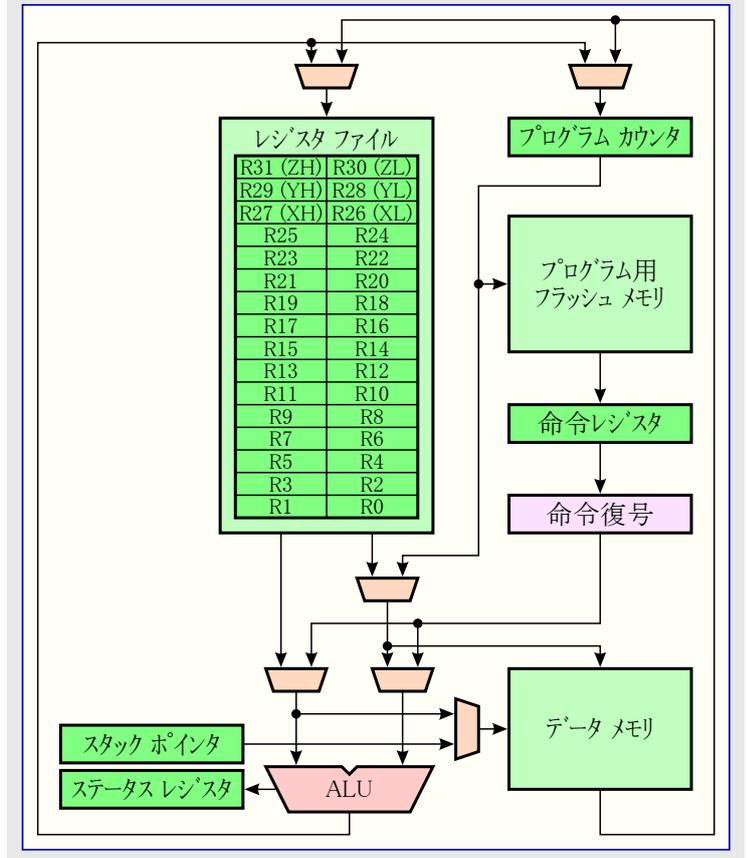
I/Oメモリ空間は制御レジスタ、SPI、他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリは直接またはレジスタファイルの次のデータ空間位置\$20~\$5Fとしてアクセスできます。加えて、本デバイスはST/STS/STDとLD/LDS/LDD命令だけが使えるSRAM内の\$60~\$FFに拡張I/O空間を持ちます。

### 11.2. ALU (Arithmetic Logic Unit)

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

関連リンク [242頁の「命令要約」](#)

図11-1. AVR基本構造の構成図



### 11.3. ステータス レジスタ

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

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SREG

変位 : \$5F (\$3F)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$3Fです。

ビット	7	6	5	4	3	2	1	0
	I	T	H	S	V	N	Z	C
アクセス種別	R/W							
リセット値	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こった後にハードウェアによって解除(0)され、後続の割り込みを許可するために、RETI命令によって設定(1)されます。Iビットは「**命令一式手引書**」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命令によってレジスタファイルのレジスタ内のビットに複写できます。

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

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

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

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

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

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

##### ● ビット2 – N : 負フラグ (Negative Flag)

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

##### ● ビット1 – Z : ゼロフラグ (Zero Flag)

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

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

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

## 11.4. 汎用レジスタ ファイル

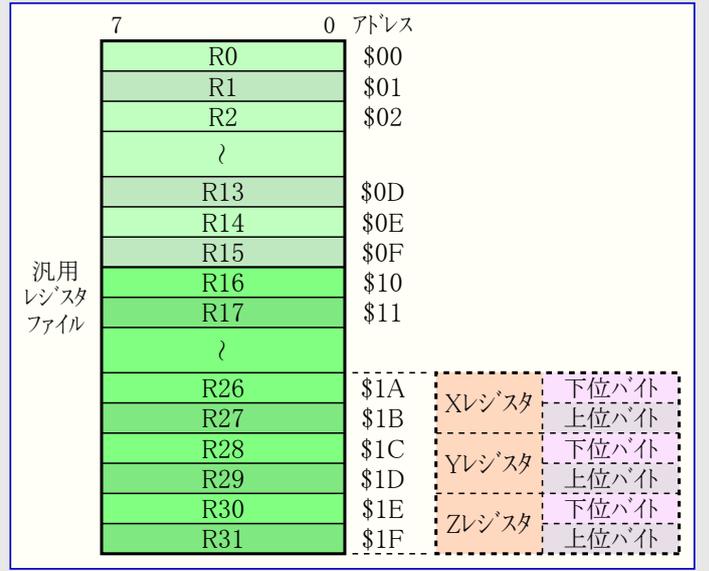
このレジスタ ファイルはAVRの増強したRISC命令群用に最適化されています。必要な効率と柔軟性を達成するために、次の入出力機構がレジスタ ファイルによって支援されます。

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

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位置へ直接配置することで、それらはデータ メモリ アドレスも割り当てられます。例え物理的にSRAM位置として実装されていないとしても、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11-2. AVR CPU 汎用レジスタ構成図



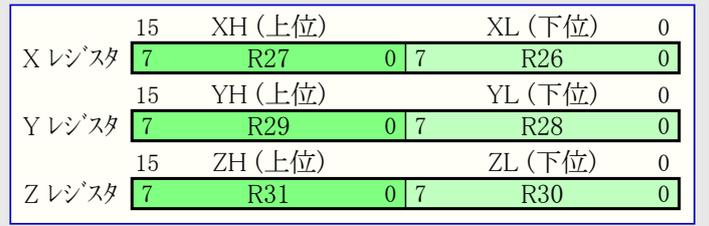
### 11.4.1. Xレジスタ, Yレジスタ, Zレジスタ

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

関連リンク [242頁の「命令要約」](#)

図11-3. X,Y,Zレジスタ構成図



## 11.5. スタック ポインタ

スタックは主に一時データの保存、局所変数の保存、割り込みとサブルーチン呼び出し後の戻りアドレスの保存に使われます。スタックは高位メモリから低位メモリへ伸長するように実行されます。スタック ポインタレジスタは常にこのスタックの先頭(訳注:次に使われるべき位置)を指し示します。スタック 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。スタックPUSH命令はスタックポインタを減らします。

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。初期スタック ポインタ値は内部SRAMの最終アドレスに等しく、スタック ポインタはSRAMの先頭以上に設定されなければなりません。スタック ポインタの詳細については表をご覧ください。

表11-1. スタック ポインタ命令

命令	スタック ポインタ	内容
PUSH	-1	データがスタック上に押し込まれます。
CALL,ICALL,RCALL	-2	サブルーチン呼び出しまたは割り込みでの戻りアドレスがスタック上に押し込まれます。
POP	+1	データがスタックから引き出されます。
RET,RETI	+2	サブルーチンまたは割り込みからの復帰での戻りアドレスがスタックから引き出されます。

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

### 11.5.1. SPH – スタックポインタ上位バイト (Stack Pointer High byte)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPH

変位 : \$5E (\$3E)

リセット : RAMENDの上位バイト (ATmega48/V=\$02, ATmega88/V/168/V=\$04)

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$3Eです。

ビット	7	6	5	4	3	2	1	0
	-					SP10~8, SPH2~0		
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0/1/1	1/0/0	0

#### ● ビット2~0 – SP10~8 : スタックポインタ上位バイト値 (Stack Pointer high byte value)

SPHとSPLはSPに組み合わせられます。SPH2~0はSP10~8を意味します。ATmega48/VでのSP10は無効です。

### 11.5.2. SPL – スタックポインタ下位バイト (Stack Pointer Low byte)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPL

変位 : \$5D (\$3D)

リセット : RAMENDの下位バイト (\$FF)

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$3Dです。

ビット	7	6	5	4	3	2	1	0
	SP7~0, SPL7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	1	1	1	1	1	1	1	1

#### ● ビット7~0 – SP7~0 : スタックポインタ下位バイト値 (Stack Pointer low byte value)

SPHとSPLはSPに組み合わせられます。SPL7~0はSP7~0を意味します。

## 11.6. 命令実行タイミング

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

右図はハーバード構造と高速アクセスレジスタファイルの概念によって可能とされる並列の命令取得と命令実行を示します。これは機能対費用、機能対クロック、機能対電源部に関する好結果と対応するMHzあたり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。

右図はレジスタファイルに対する内部タイミングの概念を示します。単一クロック周期で2つのレジスタオペランドを使うALU操作が実行され、その結果が転送先レジスタへ書き戻されます。

図11-4. 命令の取得と実行の並列動作

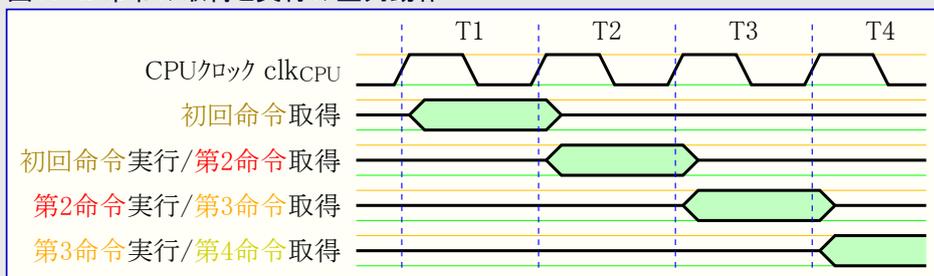
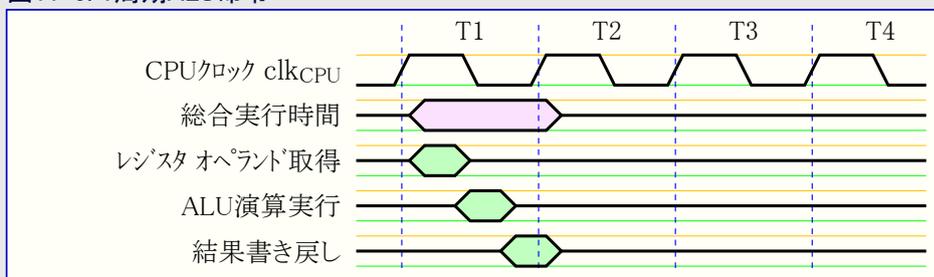


図11-5. 1周期ALU命令



## 11.7. リセットと割り込みの扱い

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

既定でのプログラムメモリ空間の最下位アドレスはリセットと割り込みベクタとして定義されます。下位側アドレスがより高い優先順位です。リセットが最高優先順位で次が外部割り込み要求0(INT0)です。割り込みベクタは**MCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビット**の設定(1)によって**ブートフラッシュ領域先頭**へ移動できます。リセットベクタも**BOOTRSTヒューズ**のプログラム(0)によってブートフラッシュ領域先頭へ移動できます。

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

根本的に2つの割り込み形式があります。

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

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

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

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

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

### アセンブリ言語プログラム例

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

### C言語プログラム例

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

注: 「コード例について」を参照してください。

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

### アセンブリ言語プログラム例

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

### C言語プログラム例

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

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

注: 「コード例について」を参照してください。

関連リンク [201頁の「MEMPROG - メモリプログラミング」](#)  
[190頁の「BTLDR - ブートローダ支援 \(RWW自己プログラミング\)」](#)

### 11.7.1. 割り込み応答時間

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

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

## 12. AVRのメモリ

### 12.1. 概要

本章は本デバイス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を持ちます。加えて本デバイスはデータ保存用EEPROMメモリが特徴です。全てのメモリ空間は一般的な直線的アドレスです。

### 12.2. 実装書き換え可能なプログラム用フラッシュメモリ

ATmega48/V/88/V/168/Vはプログラム保存用に実装書き換え可能な4/8/16Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、フラッシュメモリは2/4/8K×16ビットとして構成されます。ソフトウェア保護のため、フラッシュプログラムメモリ空間はATmega88/V/168/Vで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。ATmega48/V/88/V/168/Vのプログラムカウンタ(PC)は11/12/13ビット幅、故に2/4/8Kプログラムメモリ位置のアドレス指定です。ブートプログラム領域の操作と関係するソフトウェア保護用ブート施錠ビットは「自己プログラミング - ATmega48/V」と「BTLDR - 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 - ATmega88/V/168/V」で詳細に記述されます。SPIピンを使うフラッシュデータ直列書き込みの記述については「MEMPROG - メモリプログラミング」を参照してください。

定数表はプログラム用メモリ取得(LPM)命令を使って全てのプログラムメモリアドレス空間に配置することができます。

命令の取得と実行のタイミング図は「命令実行タイミング」で示されます。



関連リンク [190頁の「BTLDR - 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」](#)  
[201頁の「MEMPROG - メモリプログラミング」](#)  
[18頁の「命令実行タイミング」](#)

### 12.3. データ用SRAMメモリ

下図は本デバイスのSRAMメモリ構成方法を示します。

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

下位768/1280/1280データメモリ位置はレジスタファイル、I/Oメモリ、拡張I/Oメモリ、データ用内蔵SRAMに充てます。先頭の32位置はレジスタファイル、次の64位置は標準I/Oメモリ、その次の160位置は拡張I/Oメモリ、そして次の512/1024/1024位置はデータ用内蔵SRAMに充てます。

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。

- 直接
  - 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
  - 変位付き間接動作はYまたはZレジスタで与えられる基準アドレスからの63アドレス位置に届きます。
- 間接
  - レジスタファイル内のR26~R31レジスタは間接アドレス指定ポインタ用レジスタが特徴です。
- 事前減少付き間接
  - (使われる)X,Y,Zアドレスレジスタが減少(-1)されます。
- 事後増加付き間接
  - (使われる)X,Y,Zアドレスレジスタが増加(+1)されます。

本デバイスの32個の汎用レジスタ、64個のI/Oレジスタ、160個の拡張I/Oレジスタ、512/1K/1Kバイトのデータ用内蔵SRAMはこれら全てのアドレス指定種別を通して全部アクセスできます。

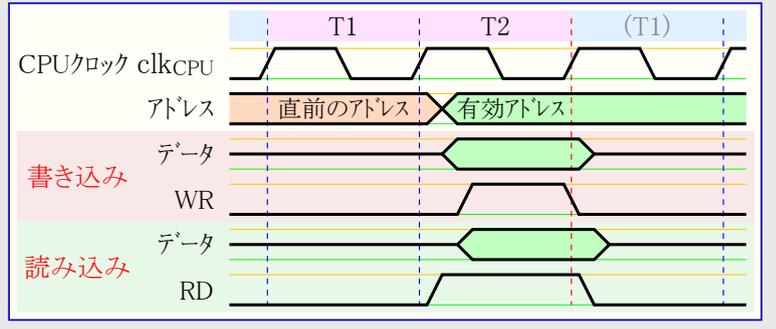


### 12.3.1. データ メモリ アクセス タイミング

データ用内蔵SRAMアクセスは右図で記載されるように2 clkCPU周期で実行されます。

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

図12-4. データ用内蔵SRAMアクセス周期



### 12.4. データ用EEPROMメモリ

ATmega48/V/88/V/168/Vは256/512/512ビットのデータ用EEPROMを含みます。それは単一ビットが読み書きできる分離したデータ空間として構成されます。EEPROMは最低100,000回の消去/書き込み回数の耐久性があります。CPUとEEPROM間のアクセスは以降のEEPROMアドレスレジスタ、EEPROMデータレジスタ、EEPROM制御レジスタで詳細に記述されます。

SPIまたは並列プログラミングでのEEPROMプログラミングの詳細な記述は関連リンクをご覧ください。

関連リンク [201頁の「MEMPROG - メモリ プログラミング」](#)

#### 12.4.1. EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROM書き込み(訳注:原文はアクセス)時間は表12-2.で与えられます。(書き込みは)自己タイミング機能ですが、使用者ソフトウェアは**次バ** **トが書ける時を検知**してください。使用者コードがEEPROMに書く命令を含む場合、いくつかの予防処置が取られねばなりません。厳重に濾波した電源では電源投入/切断でVCCが緩やかに上昇または下降しそうです。これはデバイスが何周期かの時間、使われるクロック周波数に於いて最小として示されるより低い電圧で走行する原因になります。これらの状態で問題を避ける方法の詳細については以下の「EEPROMデータ化けの防止」を参照してください。

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制御レジスタの説明と「**非分離ビット書き込み**」と「**分離ビット書き込み**」を参照してください(訳注:本行内容追加)。

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。

#### 12.4.2. 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。

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

### 12.5. I/Oメモリ (レジスタ)

本デバイスのI/O空間定義は「**レジスタ要約**」で示されます。

デバイスの全てのI/Oと周辺機能はI/O空間に配置されます。全てのI/O位置はI/O空間と32個の汎用作業レジスタ間のデータ転送を行うLD/LDS/LDD命令とST/STS/STD命令によってアクセスされます。アドレス範囲\$00~\$1F内のI/OレジスタはSBI命令とCBI命令の使用で直接的にビットアクセス可能です。これらのレジスタではSBISとSBIC命令の使用によって単一ビット値が検査できます。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。本デバイスはINやOUT命令で予約した64位置で支援されるより多くの周辺機能部を持つ複合マイクロコントローラです。SRAM(データ空間)内\$60~\$FFの拡張I/O領域に対してはLD/LDS/LDDとST/STS/STD命令だけが使えます。

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

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

I/Oと周辺制御レジスタは以降の項で説明されます。

関連リンク [201頁の「MEMPROG – メモリプログラミング」](#)  
[239頁の「レジスタ要約」](#)  
[242頁の「命令要約」](#)

### 12.5.1. 汎用I/Oレジスタ

本デバイスは3つの汎用I/Oレジスタを含みます。これらのレジスタはどの情報の格納にも使え、特に全体変数や状態フラグの格納に有用です。(I/O)アドレス範囲\$00～\$1Fの汎用I/OレジスタはSBI,CBI,SBIS,SBIC命令の使用で直接ビットアクセスが可能です。

(訳注) 参考のため、以下のEEPROMアクセス方法を追加しました。

#### 12.a.1. 非分離バイトプログラミング

非分離バイトプログラミングの使用は最も簡単な動作です。EEPROMにバイトを書くとき、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEPmnビットが'00'ならば、(EEMPEが1を書かれる後の4周期内の)EEPEの1書き込みは消去/書き込み動作を起動します。消去と書き込みの両周期は1操作で行われ、総プログラミング時間は表12-1.で与えられます。EEPEビットは消去と書き込み動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

#### 12.a.2. 分離バイトプログラミング

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

#### 12.a.3. 消去

バイトを消去するにはアドレスがEEARに書かれなければなりません。EEPmnビットが'01'なら、(EEMPEが1を書かれた後の4周期内の)EEPEの1書き込みは消去動作だけを起動します(プログラミング時間は表12-1.で与えられます)。EEPEビットは消去動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

#### 12.a.4. 書き込み

(特定)位置を書くため、使用者はEEARにアドレス、EEDRにデータを書かなければなりません。EEPmnビットが'10'なら、(EEMPEが1を書かれる後の4周期内の)EEPEの1書き込みは書き込み動作だけを起動します(プログラミング時間は表12-1.で与えられます)。EEPEビットは書き込み動作が完了されるまで設定(1)に留まります。書かれるべき位置が書き込み前に消去されていなければ、元の保存したデータは失ったとみなされなければなりません。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

## 12.6. メモリ関係レジスタ

### 12.6.1. EEARH – EEPROMアドレスレジスタ上位 (EEPROM Address Register High)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : EEARH

変位 : \$42 (\$22)

リセット : \$00, '0000000x' (ATmega48/V, ATmega88/V / 168/V)

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$22です。

ビット	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	(EEAR8)
アクセス種別	R	R	R	R	R	R	R	R/W
リセット値	0	0	0	0	0	0	0	不定

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

EEARLを参照してください。

ATmega48/Vに対してこれは未使用ビットで常に0を書かれなければなりません。

### 12.6.2. EEARL – EEPROMアドレスレジスタ下位 (EEPROM Address Register Low)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : EEARL

変位 : \$41 (\$21)

リセット : \$xx

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$21です。

ビット	7	6	5	4	3	2	1	0
	EEAR7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	不定	不定	不定	不定	不定	不定	不定	不定

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

EEPROMアドレスレジスタ(EEARHとEEARL)は256/512/512バイトのEEPROM空間内のEEPROMアドレスを指定します。EEPROMデータバイトは0～255/511/511間で直線的に配されます。EEARの初期値は不定です。EEPROMがアクセスされ得る前に適切な値が書かれなければなりません。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

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

名称 : EEDR

変位 : \$40 (\$20)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$20です。

ビット	7	6	5	4	3	2	1	0
	EEDR7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	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。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

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

名称 : EECR

変位 : \$3F (\$1F)

リセット : '00xx00x0'

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$1Fです。

ビット	7	6	5	4	3	2	1	0
	-	-	EEPROM1,0		EERIE	EEMPE	EEPE	EERE
アクセス種別	R	R	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	不定	不定	0	0	不定	0

### ● ビット5,4 – EEPROM1,0 : EEPROMプログラミング種別 (EEPROM Programming Mode Bits)

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

表12-1. EEPROMプログラミング種別

EEPROM1	EEPROM0	プログラミング時間	動作
0	0	3.4ms	1操作での消去と書き込み(非分離操作)
0	1	1.8ms	消去のみ
1	0	1.8ms	書き込みのみ
1	1	-	将来使用に予約

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

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

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

EEMPEビットはEEPROMプログラム許可(EEPE)ビットの'1'書き込みが有効か無効かどちらかを決めます。EEMPEが設定(1)されると、4クロック周期内のEEPE設定(1)は選んだアドレスのEEPROMをプログラムします。

EEMPEが0なら、EEPE設定(1)は無効です。EEMPEがソフトウェアによって'1'が書かれてしまうと、4クロック周期後にハードウェアがこのビットを0に解除します。EEPROM書き込み手順については次のEEPE記述をご覧ください。

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

EEPROMプログラム許可信号(EEPE)はEEPROMへの書き込み許可信号です。EEPEが(1)を書かれると、EEPROMはEEPROM1,0ビット設定に従ってプログラムされます。EEPEへ'1'が書かれる前にEEPROM主プログラム許可(EEMPE)ビットは'1'を書かれなければならない、さもなければEEPROM書き込み(消去)は行われません。EEPROMを書くとき、次の手順に従うべきです(手順3.と4.の順番は重要ではありません)。

1. EEPROMプログラム許可(EEPE)ビットが0になるまで待ちます。
2. SPM制御/状態レジスタ(SPMCSR)のSPM操作許可(SPMEN)ビットが0になるまで待ちます。
3. 今回のEEPROMアドレスをEEPROMアドレスレジスタ(EEAR)に書きます。(任意、省略可)
4. 今回のEEPROMデータをEEPROMデータレジスタ(EEDR)に書きます。(任意、省略可)
5. EEPROM制御レジスタ(EECR)のEEMPEビットに'1'、EEPEビットに'0'を同時に書きます。
6. EEMPEビット設定後4クロック周期内にEEPROMプログラム許可(EEPE)ビットへ'1'を書きます。

CPUがフラッシュメモリ書き込み中、EEPROMはプログラム(書き込み)ができません。ソフトウェアは新規EEPROM書き込みを始める前にフラッシュメモリのプログラミングが完了されていることを検査しなければなりません。2.はソフトウェアがフラッシュメモリをプログラム(書き込み)することをCPUに許すブートローダを含む場合だけ関係します。フラッシュメモリが決してCPUによって更新されないなら、2.は省略できます。



#### 警告:

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

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

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

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

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

EEPROMアクセスの時間には校正済み内蔵RC発振器が使われます。CPUからのEEPROMアクセスに対する代表的な書き込み時間については右表をご覧ください。

表12-2. EEPROM書き込み時間

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

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

```

アセンブリ言語プログラム例
EEPROM_WR:  SBIC  EECR, EEPE                ;EEPROMプログラミング完了ならばスキップ
              RJMP  EEPROM_WR            ;以前のEEPROMプログラミング完了まで待機
;
              LDI   R19, (0<<EEM1) | (0<<EEM0) ;プログラミング種別値取得(本例は非分離)
              OUT   EECR, R19             ;対応プログラミング種別設定
              OUT   EEARH, R18           ;EEPROMアドレス上位バイト設定
              OUT   EEARL, R17           ;EEPROMアドレス下位バイト設定
              OUT   EEDR, R16            ;EEPROM書き込み値を設定
              SBI   EECR, EEMPE          ;EEPROM主プログラム許可ビット設定
              SBI   EECR, EEPE          ;EEPROMプログラミング開始(プログラム許可ビット設定)
              RET                          ;呼び出し元へ復帰

C言語プログラム例
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    while (EECR & (1<<EEPE)); /* 以前のEEPROMプログラミング完了まで待機 */
    EECR = (0<<EEM1) | (0<<EEM0); /* 対応プログラミング種別設定 */
    EEAR = uiAddress; /* EEPROMアドレス設定 */
    EEDR = ucData; /* EEPROM書き込み値を設定 */
    EECR |= (1<<EEMPE); /* EEPROM主プログラム許可 */
    EECR |= (1<<EEPE); /* EEPROMプログラミング開始 */
}
    
```

注: 「コード例について」を参照してください。

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

```

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

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

注: 「コード例について」を参照してください。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : GPIOR2

変位 : \$4B (\$2B)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$2Bです。

ビット	7	6	5	4	3	2	1	0
	GPIOR27~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット7~0 – GPIOR27~0 : 汎用I/Oレジスタ2 (General Purpose I/O)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : GPIOR1

変位 : \$4A (\$2A)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$2Aです。

ビット	7	6	5	4	3	2	1	0
	GPIOR17~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット7~0 – GPIOR17~0 : 汎用I/Oレジスタ1 (General Purpose I/O)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : GPIOR0

変位 : \$3E (\$1E)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$1Eです。

ビット	7	6	5	4	3	2	1	0
	GPIOR07~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

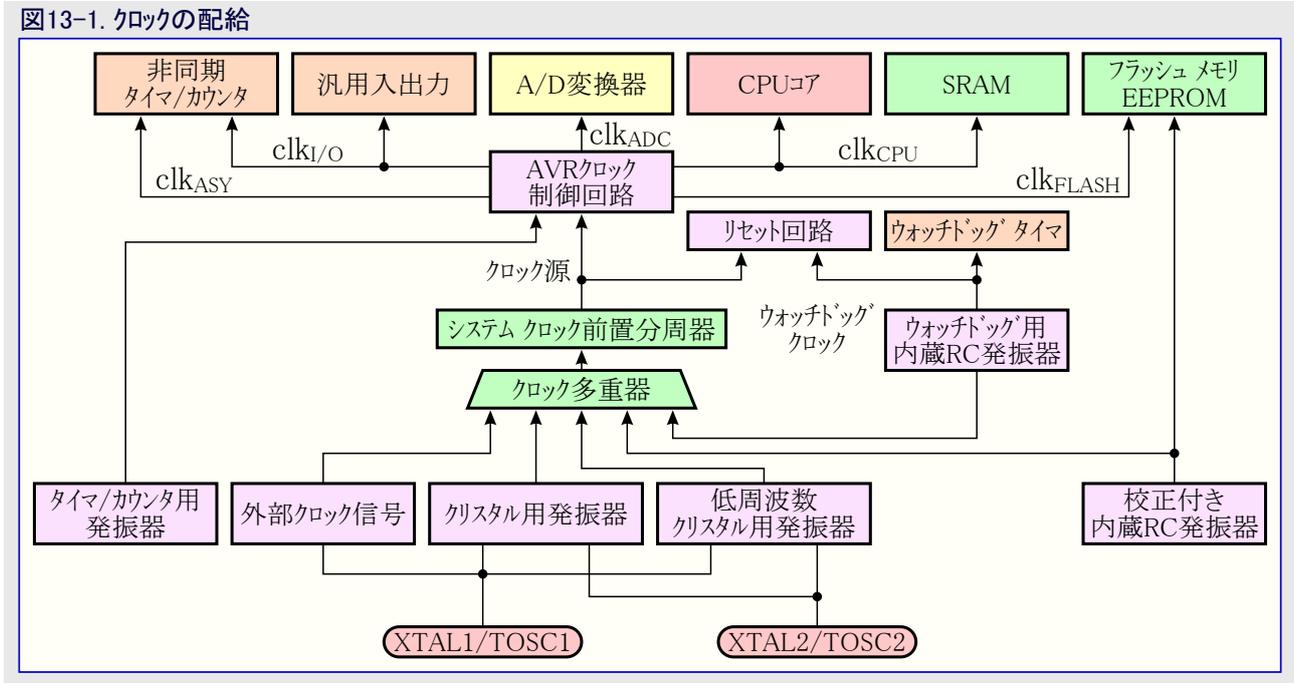
- ビット7~0 – GPIOR07~0 : 汎用I/Oレジスタ0 (General Purpose I/O)

## 13. システム クロックとクロック選択

### 13.1. クロックシステムとその配給

下図はデバイス内の主要なクロックシステムとそれらの配給を示します。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、各種**休止形態**を用いることによって、使われない部分のクロックを停止することができます。クロックシステムは以降の項で記述されます。

システム クロック周波数は**システム クロック前置分周器**から生成された周波数を参照します。AVRクロック制御部からの全てのクロック出力は同じ周波数で動きます。



#### 13.1.1. CPUクロック - $clk_{CPU}$

CPUクロックはAVRコアの動作と関係するシステム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。

#### 13.1.2. 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、2線直列インターフェース(TWI)単位部の開始条件検出は $clk_{I/O}$ が停止される時に非同期で実行されます。

**注:** パワーダウン動作から起き上がるのにレベル起動割り込みが使われる場合、そのレベル割り込みを起動するような完全な起動のために必要とされた割り込みはMCUに対して十分な長さを保持しなければなりません。始動時間の終了前にそのレベルが消滅すると、MCUは未だ起き上がりますが、割り込みが生成されません。始動時間は**SUT**と**CKSEL**のヒューズによって定義されます。

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

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

#### 13.1.4. 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クロック用結晶から直接的にクロック駆動されることを非同期タイマ/カウンタに許します。この専用クロック範囲はデバ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。

#### 13.1.5. A/D変換クロック - $clk_{ADC}$

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た雑音を低減するためにCPUとI/Oクロックの停止を許します。これはより正確なA/D変換結果を与えます。

## 13.2. クロック元

このデバイスには右で示されるようにフラッシュ ヒューズ ビットによって選択可能な後続のクロック元選択があります。選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。

表13-1. クロック元選択

クロック元	CKSEL3~0
外部クリスタル低電力発振器	1111~1000
外部クリスタル全振幅発振器	0111~0110
外部低周波数クリスタル発振器	0101~0100
128kHz内部(WDT)発振器	0011
校正付き内蔵RC発振器	0010
外部クロック信号	0000
(予約)	0001

注: 1=非プログラム、0=プログラム

### 13.2.1. 既定のクロック元

このデバイスは8.0MHzの校正付き内蔵RC発振器でCKDIV8ヒューズがプログラム(0)され、結果として1.0MHzのシステム クロックで出荷されます。起動時間は計時完了周期が許可され、最大に設定されます(CKSEL=0010, SUT=10, CKDIV8=プログラム(0))。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。

### 13.2.2. クロック起動手順

何れのクロック元も発振を開始するための十分なVCCと、それが安定であると考えられるのに先立って最低発振周期数が必要です。

十分なVCCを保証するために、その他全てのリセット元によってデバイスリセットが開放された後、デバイスは起動遅延時間( $t_{\text{TOU}}$ )の内部リセットを発生します。内部リセットに対する起動条件の記述については関連リンクをご覧ください。この遅延( $t_{\text{TOU}}$ )はウォッチドッグ発振器で計時され、遅延周期数はSUTとCKSELのヒューズにより設定されます。選択可能な遅延は右表で示されます。ウォッチドッグ発振器の周波数は電圧に依存します。

表13-2. WDT発振器の代表計時完了値、周期数

VCC=3.0V	VCC=5.0V	周期数
0ms	0ms	0
4.3ms	4.1ms	512
69ms	65ms	8K (8192)

遅延の主な目的は最小VCCを供給されるまでデバイスをリセットに保つことです。この遅延は実電圧を監視しませんので、VCC上昇時間より長い遅延を選ぶことが必要とされます。これが不可能な場合、内部または外部の低電圧検出回路(BOD)が使われるべきです。BOD回路がリセットを開放する前に十分なVCCを保証するでしょうから、起動遅延時間は禁止され得ます。低電圧検出回路(BOD)なしでの起動遅延時間の禁止は推奨されません。

この発振器はクロックが安定と見做されるのに先立つ最低クロック数の発振を必要とされます。内部リプル カウンタは発振器の出力クロックを監視し、与えられたクロック周期数間、内部リセットを活性に保ちます。このリセットはその後に開放され、デバイスが実行を開始します。推奨発振器起動時間はクロック種別に依存し、外部的に印加されたクロック用の6周期から、低周波数クリスタル用の32K周期まで変化します。

クロックについての起動手順は計時完了遅延とデバイスがリセットから起動するときの起動時間の両方を含みます。パワーダウンまたはパワーセーブから起動するとき、VCCは十分な電圧であると認識され、起動時間だけが含められます。

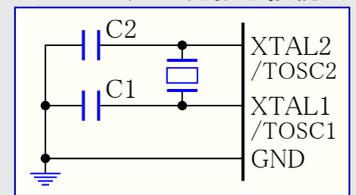
### 13.2.3. 発振器/振動子接続

XTAL1とXTAL2のピンは右図で示されるように、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。クリスタル発振器またはセラミック振動子のどちらでも使うことができます。

C1とC2はクリスタル発振器とセラミック振動子の両方について常に等しくすべきです。このコンデンサの最適値は使うクリスタル発振器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。クリスタル発振器使用に対するコンデンサ選択について初期の指針のいくつかは以降の表で与えられます。セラミック振動子については製造業者によって与えられたコンデンサ値が使われるべきです。

関連リンク [30頁の「クリスタル用低電力発振器」](#)  
[30頁の「クリスタル用全振幅発振器」](#)  
[31頁の「低周波数クリスタル用発振器」](#)

図13-2. クリスタル発振器接続図



### 13.3. クリスタル用低電力発振器

この発振器はXTAL2出力上の電圧振幅を減少した低電力発振器です。これは最低消費電力を与えますが、他のクロック入力を駆動する能力はなく、雑音が多い環境で、より雑音の影響を受け易くなります。これらの場合には「[クリスタル用全振幅発振器](#)」を参照してください。

クリスタル発振器使用に対するコンデンサ選択について、初期の指針のいくつかは下表で与えられます。クリスタル発振器は「[発振器/振動子接続](#)」で記載されるように接続されるべきです。

低電力発振器は示された周波数範囲で各々最適化された3つの異なる種別で動作できます。この動作種別は下表で示されるようにCKSEL3~1ヒューズによって選ばれます。

表13-3. 低電力クリスタル用発振器動作種別

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

注: これは各周波数範囲に対する推奨CKSEL設定です。

注: 周波数が仕様(VCC依存)超の場合、CKDIV8=0が可能です。分周後クロックが仕様内であることを保証しなければなりません。

注1: 本選択はクリスタル発振器ではなく、セラミック振動子でのみ使われるべきです。

CKSEL0ヒューズはSUT1,0ヒューズと共に下表で示されるように起動時間を選びます。

表13-4. クリスタル発振器/セラミック振動子用低電力発振器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 0	258×CK (注1)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
	0 1	258×CK (注1)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	1 0	1K×CK (注2)	14×CK	外部セラミック振動子、低電圧検出(BOD)リセット許可
	1 1	1K×CK (注2)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
1	0 0	1K×CK (注2)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	0 1	16K×CK	14×CK	外部クリスタル発振器、低電圧検出(BOD)リセット許可
	1 0	16K×CK	14×CK+4.1ms	外部クリスタル発振器、高速上昇電源
	1 1	16K×CK	14×CK+65ms	外部クリスタル発振器、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器用ではありません。

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器も使えます。

関連リンク [30ページの「クリスタル用全振幅発振器」](#)

### 13.4. クリスタル用全振幅発振器

この発振器はXTAL2出力上で供給電圧端振幅にする全振幅発振器です。これは雑音が多い環境や他のクロック入力を駆動するのに適します。消費電流は「[クリスタル用低電力発振器](#)」より多くなります。全振幅クリスタル発振器がVCC=2.7~5.5Vに対してのみ動作することに注意してください。

クリスタル発振器使用に対するコンデンサ選択について、初期の指針のいくつかは表13-5.で与えられます。クリスタル発振器は「[発振器/振動子接続](#)」で記載されるように接続されるべきです。

この動作種別は下表で示されるようにCKSEL3~1ヒューズによって選ばれます。

表13-5. 全振幅クリスタル用発振器動作種別

CKSEL3~1	周波数範囲	推奨C1,2容量
011	0.4~20MHz	12~22pF

注: 周波数が仕様(VCC依存)超の場合、CKDIV8=0が可能です。分周後クロックが仕様内であることを保証しなければなりません。

CKSEL0ヒューズはSUT1,0ヒューズと共に下表で示されるように起動時間を選びます。

表13-6. クリスタル発振子/セラミック振動子用全振幅発振器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 0	258×CK (注1)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
	0 1	258×CK (注1)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	1 0	1K×CK (注2)	14×CK	外部セラミック振動子、低電圧検出(BOD)リセット許可
	1 1	1K×CK (注2)	14×CK+4.1ms	外部セラミック振動子、高速上昇電源
1	0 0	1K×CK (注2)	14×CK+65ms	外部セラミック振動子、低速上昇電源
	0 1	16K×CK	14×CK	外部クリスタル発振子、低電圧検出(BOD)リセット許可
	1 0	16K×CK	14×CK+4.1ms	外部クリスタル発振子、高速上昇電源
	1 1	16K×CK	14×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の保証します。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。

関連リンク 30頁の「クリスタル用低電力発振器」

### 13.5. 低周波数クリスタル用発振器

低周波数クリスタル用発振器は時計用32.768kHzクリスタルでの使用に最適化されています。クリスタル選択時、負荷容量とクリスタルの等価直列抵抗(ESR)が考慮されなければなりません。両値はクリスタル販売業者によって指定されます。この発振器は非常に低い電力消費用に最適化されており、故にクリスタル選択時、推奨最大ESRを考慮してください。

低周波数クリスタル用発振器は各TOSCピンで内部負荷容量を提供します。

表13-7. 時計用32.768kHzクリスタル用推奨最大ESR

クリスタル負荷容量 (CL:pF)	最大ESR (kΩ) (注1)
6.5	75
9.0	65
12.5	30

注1: 最大ESRは特性を基にした代表値です。

表13-8. 低周波数クリスタル用発振器内部容量

32kHz発振器形式	容量 (pF)	
	XTAL1/TOSC1	XTAL2/TOSC2
システムクロック用発振器	18	8
タイマ/カウンタ用発振器	6	6

各TOSCピンで必要とする外部容量(C)は右式を使って計算することができます。

$$C_e + C_i = 2 \times C_L - C_s$$

C<sub>e</sub> : 図13-2.で記述されるように追加外部容量です。

C<sub>i</sub> : 上表でのピン容量です。

C<sub>L</sub> : クリスタル製造業者によって指定された32.768kHzクリスタル用の負荷容量です。

C<sub>s</sub> : 1つのTOSCピンに対する総浮遊容量です。

表13-8.で与えられたものより高い指定負荷容量(CL)のクリスタルは図13-2.で記述するように付加外部容量(コンデンサ)が必要です。

低周波数クリスタル用発振器はCKSEL0ヒューズを'0100'または'0101'に設定することによって選ばれなければなりません。

表13-9. 低周波数クリスタル発振器起動遅延時間選択表

CKSEL0	パワーダウン、パワーセーブからの起動遅延時間	推奨使用法
0 1 0 0 (注)	1K×CK	
0 1 0 1	32K×CK	始動での周波数安定

注: 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。

起動時間は次表で示されるようにSUTヒューズによって決定されます。

表13-10. 低周波数クリスタル発振器起動遅延時間選択表

SUT1,0	リセットからの付加遅延時間 (VCC=5.0V)	電力条件
0 0	14×CK	低電圧検出(BOD)リセット許可
0 1	14×CK+4.1ms	高速上昇電源
1 0	14×CK+65ms	低速上昇電源
1 1		(予約)

関連リンク 29頁の「発振子/振動子接続」

33頁の「タイマ/カウンタ用発振器」

### 13.6. 校正付き内蔵RC発振器

既定による校正された内蔵RC発振器は8.0MHzのクロックを供給します。電圧と温度に依存しますが、このクロックは使用者によって高精度な校正ができます。このデバイスはCKDIV8ヒューズがプログラム(0)で出荷されます。

このクロックは下表で示されるように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。リセット中、ハードウェアが発振校正(OSCCAL)レジスタに予めプログラムされた校正バイトを設定し、これによってRC発振器を自動的に校正します。

ソフトウェアからOSCCALレジスタを変更することによって、工場校正を使うよりも高い精度を得ることができます。

この発振器がチップ(システム)クロックとして使われる時に、ウォッチドッグ発振器は未だウォッチドッグタイムとリセット付加遅延計時器に使われません。

表13-11. 校正付き内蔵RC発振器動作種別

CKSEL3~0	周波数範囲 (MHz)
0 0 1 0	7.3~8.1

注: デバイスはこの選択で出荷されます。

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

この発振器が選ばれると、起動時間はSUTヒューズによって決定されます。

表13-12. 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	14×CK (注2)	低電圧検出リセット(BOD)許可
0 1	6×CK	14×CK+4.1ms	高速上昇電源
1 0 (注1)	6×CK	14×CK+65ms	低速上昇電源
1 1	(予約)		

注1: デバイスはこの選択で出荷されます。

- 関連リンク
- 214頁の「クロック特性」
  - 33頁の「システムクロック前置分周器」
  - 203頁の「校正バイト」
  - 34頁の「OSCCAL - 発振校正レジスタ」
  - 202頁の「ヒューズビット」

### 13.7. 128kHz内部発振器

128kHz内部発振器は128kHzのクロックを供給する低電力発振器です。この周波数は3V、25°Cでの公称値です。本クロックはCKSELヒューズを'0011'にプログラミング(設定)することによってシステムクロックとして選べます。

表13-13. 128kHz内部発振器動作種別

CKSEL3~0	公称周波数
0 0 1 1	128kHz

注: 128kHz発振器は非常に低い電力のクロック元で、高精度用に設計されていないことに注意してください。

このクロック元が選ばれると、起動時間はSUTヒューズによって決定されます。

表13-14. 128kHz内部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	推奨使用法
0 0	6×CK	14×CK (注1)	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4ms	高速上昇電源
1 0	6×CK	14×CK+64ms	低速上昇電源
1 1	(予約)		

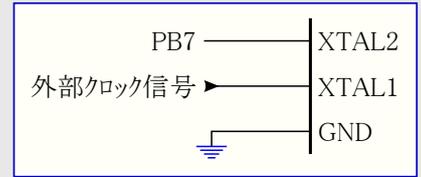
### 13.8. 外部クロック信号

外部クロック元からデバイスを駆動するには、図で示されるようにXTAL1が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はCKSELヒューズが'0000'にプログラム(設定)されなければなりません。

表13-15. 外部クロック信号動作種別

CKSEL3~0	周波数範囲
0 0 0 0	0~20MHz

図13-3. 外部クロック信号駆動接続図



このクロック元が選ばれると、起動時間はSUTヒューズによって決定されます。

表13-16. 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	14×CK	低電圧検出(BOD)リセット許可
0 1	6×CK	14×CK+4.1ms	高速上昇電源
1 0	6×CK	14×CK+65ms	低速上昇電源
1 1			(予約)

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

システムクロック前置分周器は安定な動作を保証しながら、内部クロック周波数の実行時変更の実現に使うことができます。

関連リンク [33ページの「システムクロック前置分周器」](#)

### 13.9. タイマ/カウンタ用発振器

本デバイスは低周波数クリスタル用発振器とタイマ/カウンタ用発振器に対して同じクリスタル用発振器を使います。この発振器とクリスタルの必要条件については「[低周波数クリスタル用発振器](#)」をご覧ください。

このデバイスではタイマ/カウンタ用発振器(TOSC1とTOSC2)ピンとXTAL1,XTAL2ピンを共用します。タイマ/カウンタ用発振器使用時、システムクロックはこの発振器周波数の4倍を必要とします。これとピン共用のため、タイマ/カウンタ用発振器はシステムクロック元として校正付き内蔵RC発振器選択時にだけ使うことができます。

TOSC1への外部クロック元印加は非同期状態レジスタの外部クロック許可(ASSR.EXCLK)ビットが'1'を書かれる場合に行うことができます。時計用32.768kHzクリスタルに代わる入力として外部クロックを選ぶ更なる記述については「[タイマ/カウンタの非同期動作](#)」の記述をご覧ください。

関連リンク [31ページの「低周波数クリスタル用発振器」](#)  
[119ページの「ASSR - 非同期状態レジスタ」](#)

### 13.10. クロック出力緩衝部 (外部クロック出力)

このデバイスはシステムクロックをCLKOピンに出力できます。本出力を許可するにはCKOUTヒューズがプログラム(0)されなければなりません。この動作はチップのクロックがシステム上の他の回路を駆動する時用です。このヒューズがプログラム(0)されると、I/Oピンの標準動作は無視され、このクロックはリセット中も出力されます。CLKOがクロック出力を扱うとき、校正付き内蔵RC発振器を含む何れのクロック元も選ばず、システムクロック前置分周器が使用されると、CKOUTヒューズがプログラム(0)された時の出力は分周したシステムクロックです。

### 13.11. システムクロック前置分周器

本デバイスはシステムクロック前置分周器を持ち、システムクロックはクロック前置分周レジスタ(CLKPR)を構成設定することによって分周できます。この特徴(機能)は必要とされる処理能力が低い時の消費電力削減に使えます。これは全クロック種別で使え、CPUと全同期周辺機能のクロック周波数に影響を及ぼします。clkCPU、clkFLASH、clkI/O、clkADCはCLKPR記述で示された値によって分周されます。

前置分周器設定間を切り替えるとき、システムクロック前置分周器は中間(経過途中)の周波数が直前の設定に対応するクロック周波数または新規設定に対応するクロック周波数のどちらよりも高くなく、クロック系で不具合が起きないことを保証します。前置分周器として実行するリップカウンタは分周されないクロック周波数で走行し、CPUのクロック周波数より速いかもしれません。従って例え(カウンタ値が)読めるとしても、前置分周器の状態を決めることはできず、1から他へのクロック分周値切り替えを行う正確な時間は必ずしも予測できません。クロック分周値選択(CLKPS3~0)ビット値が書かれる時から新規クロック周波数が活性(有効)になる前にT1+T2~T1+2×T2間かかります。この間で2つの有効なクロック端が生成されます。ここでのT1は直前のクロック周期、T2は新規前置分周器設定に対応する周期です。

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

1. クロック分周値変更許可(CLKPCE)ビットに'1'、CLKPR内の他の全ビットに'0'を書いてください。CLKPR=\$80
2. (次からの)4周期以内にCLKPCEへ'0'を書くと同時にCLKPS3~0へ望む値を書いてください。CLKPR=\$0n

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

関連リンク [32ページの「校正付き内蔵RC発振器」](#)  
[33ページの「外部クロック信号」](#)  
[34ページの「CLKPR - クロック前置分周レジスタ」](#)

## 13.12. クロック関係レジスタ

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

名称 : OSCCAL

変位 : \$66

リセット : デバイス固有の校正值

特質 : -

ビット	7	6	5	4	3	2	1	0
	CAL7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	デバイス固有の校正值							

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

発振校正レジスタは発振器周波数の偏差処理を省くための内蔵発振器の調整に使われます。チップのリセット中、「電気的特性」章の「クロック特性」項で指定されるように、工場校正周波数を与える予めプログラムされた値が本レジスタへ自動的に書かれます。応用ソフトウェアは発振器周波数を変更するために、このレジスタに書くことができます。この発振器は「電気的特性」章の「クロック特性」項で指定されるような周波数に校正できます。この範囲外への校正は推奨されません。

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

CAL7ビットは発振器に関する操作範囲を決めます。このビットの(0)設定は低周波数範囲になり、(1)設定は高周波数範囲になります。この2つの周波数範囲は重複し、別の言葉では、OSCCAL=\$7F設定はOSCCAL=\$80設定より高い周波数になります。

CAL6~0ビットは選んだ範囲内の周波数調整に使われます。\$00設定はその範囲の最低周波数になり、\$7F設定はその範囲の最高周波数になります。

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

名称 : CLKPR

変位 : \$61

リセット : ビット記述を参照

特質 : -

ビット	7	6	5	4	3	2	1	0
	CLKPCE	-	-	-	CLKPS3~0			
アクセス種別	R/W	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	1/0	1/0

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

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

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

これらのビットは選んだクロック元と内部システムクロック間の分周値を定義します。これらのビットは応用の必要条件に合わせた各種クロック周波数を実行時に書けます。分周値が使われると、分周器はMCUへの主クロックを分周し、全ての同期周辺機能の速度が減じられます。分周値は下表で与えられます。

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

表13-17. クロック前置分周器選択

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

## 14. 電力管理と休止形態 (Power Management and Sleep Modes)

### 14.1. 概要

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。本デバイスは応用に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。

### 14.2. 休止形態種別

下表は各種休止形態とそれらの起動元を示します。

表14-1. 各休止形態に於ける動作クロック範囲と復帰起動要因

休止種別	動作クロック範囲					動作発振器		復帰起動要因 (割り込み)						
	clkCPU	clkFLASH	clkI/O	clkADC	clkASY	主クロック供給元	タイマ用発振器	INTとPCINT	TWIアドレス一致	タイマ/カウンタ2	SPM EEPROM操作可	A/D変換完了	ウォッチドッグ	その他I/O
アイドル	○	○	○	○	○	○	②	○	○	○	○	○	○	○
A/D変換雑音低減				○	○	○	②	③	○	②	○	○	○	
パワーダウン								③	○				○	
パワーセーブ					○	○	②	③	○	○			○	
スタンバイ(注1)						○		③	○				○	
拡張スタンバイ(注1)					②	○	②	③	○	○			○	

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のみ推奨されます。

② 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)された場合です。

③ INT1とINT0についてはレベル割り込みだけです。

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

注: 「システムクロックとクロック選択」章の構成図は本デバイスの各種クロックシステムとそれらの配給に関する概要を提供します。この図は適切な休止形態の選ぶ助けになります。

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

関連リンク [28頁の「クロックシステムとその配給」](#)

### 14.3. アイドル動作

**休止種別選択(SM2~0)ビット**が'000'を書かれた時の**SLEEP**命令はMCUをアイドル動作へ移行させてCPUを停止しますが、SPI、USART、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、ウォッチドッグ、割り込み機構の継続動作を許します。この休止形態は基本的にclkCPUとclkFLASHを停止する一方、他のクロックに走行を許します。

アイドル動作はMCUにタイマ溢れや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許します。アナログ比較器割り込みからの起動が必要とされないなら、アナログ比較器制御/状態レジスタ(ACSR)の**アナログ比較器禁止(ACD)ビット**を設定(1)することによってアナログ比較器を電源断にできます。これはアイドル動作での消費電力を削減します。

関連リンク [170頁の「ACSR - アナログ比較器制御/状態レジスタ」](#)

### 14.4. A/D変換雑音低減動作

SM2~0ビットが'001'を書かれた時の**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(注)、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkI/O、clkCPU、clkFLASHを停止する一方、他のクロックに走行を許します。

これはA/D変換に対する雑音環境を改善し、より高い分解能の測定を可能にします。A/D変換器が許可されている場合、本動作に移行すると、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、以下のこれらの事象だけが、A/D変換雑音低減動作からMCUを起動することができます。

- 外部リセット
- ウォッチドッグ システムリセット
- ウォッチドッグ割り込み
- 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
- INTnの外部レベル割り込み
- ピン変化割り込み

注: タイマ/カウンタ2は非同期動作でだけ走行を維持します。

関連リンク [104頁の「TC2 - 8ビット タイマ/カウンタ2 \(PWM, 非同期動作付き\)」](#)

## 14.5. パワーダウン動作

SM2~0ビットが'010'を書かれるとき、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。

以下のこれらの事象の1つだけがMCUを起動することができます。

- 外部リセット
- ウォッチドッグ システム リセット
- ウォッチドッグ割り込み
- 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
- INTnの外部レベル割り込み
- ピン変化割り込み

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。

注: レベル起動割り込みがパワーダウン動作からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが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。起動時間はSUTとCKSELのヒューズで定義されます。

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

関連リンク [29頁の「クロック元」](#)  
[52頁の「EXTINT - 外部割り込み」](#)

## 14.6. 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が許可される場合、それらは休止中でも走行(動作)を維持します。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(I)され、タイマ/カウンタ2割り込み許可レジスタ(TIMSK2)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x割り込み許可(OCIE2x)ビットが設定(I)されるなら、デバイス是对応するどの割り込みからでも起動できます。

タイマ/カウンタ2が走行(動作)しないなら、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。

タイマ/カウンタ2はパワーセーブ動作で同期と非同期両方でクロック駆動できます。タイマ/カウンタ2が非同期クロックを使わない場合、休止中、タイマ/カウンタ用発振器は停止されます。タイマ/カウンタ2が同期クロックを使わない場合、休止中、そのクロック元は停止されます。例えばパワーセーブ動作で同期クロックが走行しても、このクロックはタイマ/カウンタ2に対してだけ利用可能です。

## 14.7. スタンバイ動作

外部クリスタル発振子/セラミック振動子系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(外部用)発振部が走行(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。

## 14.8. 拡張スタンバイ動作

外部クリスタル発振子/セラミック振動子系クロック種別が選ばれ、休止種別選択(SM2~0)ビットが'111'のとき、SLEEP命令はMCUを拡張スタンバイ動作へ移行させます。この動作は(外部用)発振部が走行(動作)を保たれる例外を除いてパワーセーブ動作と同じです。デバイスは拡張スタンバイ動作から6クロック周期で起動します。

## 14.9. 電力削減レジスタ

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

周辺機能部の停止は全体に亘る重要な消費電力の削減のために活動動作とアイドル動作で使えます。その他の休止形態ではクロックが予め停止されます。

関連リンク [38頁の「PRR - 電力削減レジスタ」](#)

## 14.10. 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。

### 14.10.1. A/D変換器 (ADC)

許可なら、A/D変換器は全休止形態で許可されます。電力を節約するため、休止形態の何れかへ移行する前にA/D変換器は禁止されるべきです。A/D変換器がOFFそして再びONに切り替えられると、次の(最初の)変換は延長された(初回)変換になります。

**関連リンク** [172頁の「ADC - A/D変換器」](#)

### 14.10.2. アナログ比較器

アイドル動作へ移行するとき、アナログ比較器は使われないなら、禁止されるべきです。A/D変換雑音削減動作へ移行するとき、アナログ比較器は禁止されるべきです。その他の休止形態でのアナログ比較器は自動的に禁止されます。しかしアナログ比較器が入力として内部基準電圧を使う設定の場合、全休止形態でアナログ比較器は禁止されるべきです。さもなければ内部基準電圧は休止形態と無関係に許可されます。

**関連リンク** [169頁の「AC - アナログ比較器」](#)

### 14.10.3. 低電圧検出器 (BOD)

低電圧検出器(BOD)が応用で必要とされないなら、この単位部はOFFにされるべきです。BODLEVELヒューズによってBOD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。

**関連リンク** [41頁の「低電圧検出\(BOD\)リセット」](#)

### 14.10.4. 内部基準電圧

内部基準電圧は低電圧検出器(BOD)、アナログ比較器、A/D変換器によって必要とされる時に許可されます。これら単位部が上の項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。

**関連リンク** [42頁の「内部基準電圧」](#)

### 14.10.5. ウォッチドッグ タイマ

ウォッチドッグ タイマが応用で必要とされないなら、この単位部はOFFにされるべきです。ウォッチドッグ タイマが許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。

**関連リンク** [42頁の「ウォッチドッグ タイマ」](#)

### 14.10.6. ポートピン

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

アナログ入力ピンに対するデジタル入力緩衝部は常に禁止されるべきです。入力ピンでのVCC/2付近のアナログ信号入力は活動動作でも重要な電流を引き起こし得ます。デジタル入力緩衝部はデジタル入力禁止レジスタ(A/D変換器用のDIDR0とアナログ比較器用のDIDR1)の書き込みによって禁止することができます。

**関連リンク** [58頁の「デジタル入力許可と休止形態」](#)  
[182頁の「DIDR0 - デジタル入力禁止レジスタ0」](#)  
[171頁の「DIDR1 - デジタル入力禁止レジスタ1」](#)

### 14.10.7. 内蔵デバッグ機能 (dW)

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

## 14.11. 電力管理用レジスタ

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

この休止形態制御レジスタは電力管理用の制御ビットを含みます。

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SMCR

変位 : \$53 (\$33)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$33です。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-		SM2~0		SE
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット3~1 – SM2~0 : 休止種別選択 (Sleep Mode Select Bit 2, 1 and 0)

SM2~0ビットは利用可能な6つの休止形態の1つを選びます。

表14-2. 休止形態種別選択

SM2~0	休止形態種別
0 0 0	アイドル動作
0 0 1	A/D変換雑音低減動作
0 1 0	パワーダウン動作
0 1 1	パワーセーブ動作
1 0 0	(予約)
1 0 1	(予約)
1 1 0	スタンバイ動作 (注)
1 1 1	拡張スタンバイ動作 (注)

注: (拡張)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の使用にだけ推奨されます。

#### ● ビット0 – SE : 休止許可 (Sleep Enable)

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

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

名称 : PRR

変位 : \$64

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	PRTWI	PRTIM2	PRTIM0	-	PRTIM1	PRSPI	PRUSART0	PRADC
アクセス種別	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7 – PRTWI : 2線直列インターフェース電力削減 (Power Reduction TWI)

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

#### ● ビット6 – PRTIM2 : タイマ/カウンタ2電力削減 (Power Reduction Timer/Counter2)

このビットへの論理1書き込みは同期動作(非同期動作許可(AS2)=0)でのタイマ/カウンタ2部を停止します。タイマ/カウンタ2が許可されると、停止前と同様に動作は継続します。

#### ● ビット5 – PRTIM0 : タイマ/カウンタ0電力削減 (Power Reduction Timer/Counter0)

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

#### ● ビット3 – PRTIM1 : タイマ/カウンタ1電力削減 (Power Reduction Timer/Counter1)

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

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

デバッグWIRE内蔵デバッグ機能を使うなら、このビットは**1**を書かれるべきではありません。このビットへの論理**1**書き込みはその単位部へのクロック停止によって**直列周辺インターフェース(SPI)**を停止します。SPIの再起動時、SPIは正しい動作を保証するために再初期化されるべきです。

- **ビット1 – PRUSART0 : USART0電力削減 (Power Reduction USART0)**

このビットへの論理**1**書き込みはその単位部へのクロック停止によって**USART**を停止します。USARTの再起動時、USARTは正しい動作を保証するために再初期化されるべきです。

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

このビットへの論理**1**書き込みは**A/D変換器(ADC)**を停止します。A/D変換器は停止前に**禁止**されなければなりません。A/D変換器停止時、**アナログ比較器**は**ADC入力切替器**を使えません。

## 15. SCRST – システム制御とリセット (System Control and Reset)

### 15.1. AVRのリセット

リセット中、全てのI/Oレジスタはそれらの初期値に設定され、プログラムはリセットベクタから実行を開始します。ATmega168/Vについてリセットベクタに配置される命令は、きっとリセット処理ルーチンへのJMP(絶対分岐)命令でしょう。ATmega48/V/88/Vについてリセットベクタに配置される命令は、きっとリセット処理ルーチンへのRJMP(相対分岐)命令でしょう。プログラムが決して割り込み元を許可しないなら、割り込みベクタは使われず、これらの位置に通常のプログラムコードが配置できます。これはリセットベクタが応用領域の一方、割り込みベクタがブート領域の場合やその逆も同様です(ATmega88/V/168/Vのみ)。次項の回路構成図はリセット論理回路を示します。

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

全てのリセット元が無効にされた後、遅延計数器(タイマ)が始動され、内部リセットを引き伸ばします。これは通常動作開始前に安定電圧へ達することを電源に許します。遅延計数器の遅延時間は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「システムクロックとクロック選択」で示されます。

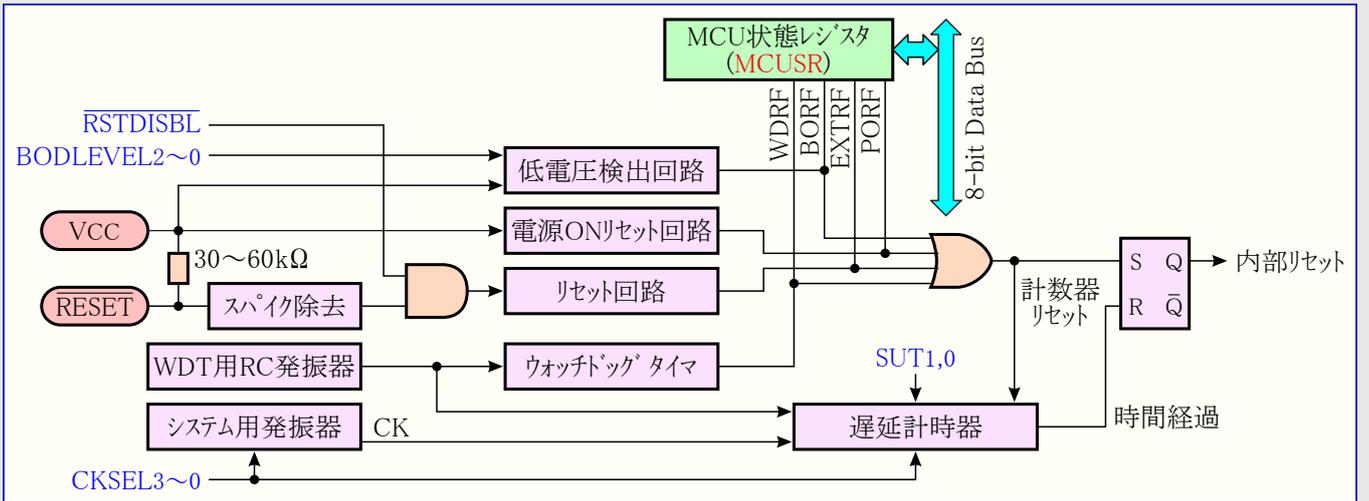
関連リンク [28頁の「システムクロックとクロック選択」](#)

### 15.2. リセット元

本デバイスには以下のリセット元があります。

- **電源ONリセット** …… 供給電圧が電源ONリセット閾値電圧(V<sub>POT</sub>)未満でMCUがリセットされます。
- **外部リセット** …… RESETピンが最小パルス幅以上Lowレベルに保たれると、MCUがリセットされます。
- **ウォッチドッグシステムリセット** …… ウォッチドッグシステムリセット動作が許可され、ウォッチドッグタイマが終了すると、MCUがリセットされます。
- **低電圧リセット** …… 低電圧検出が許可され、供給電圧(V<sub>CC</sub>)が低電圧検出電圧(V<sub>BOT</sub>)未満でMCUがリセットされます。

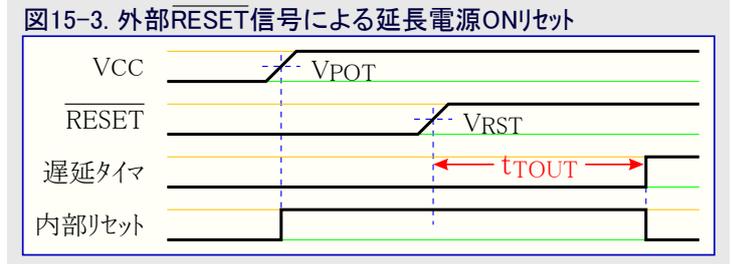
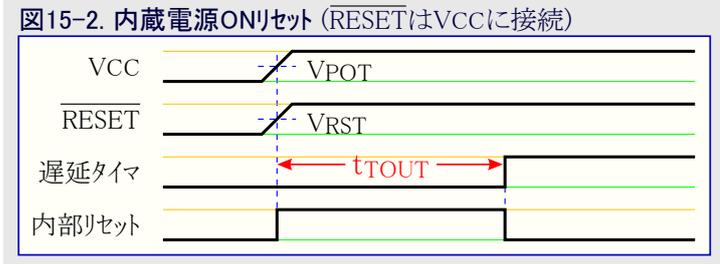
図15-1. リセット回路構成



### 15.3. 電源ONリセット

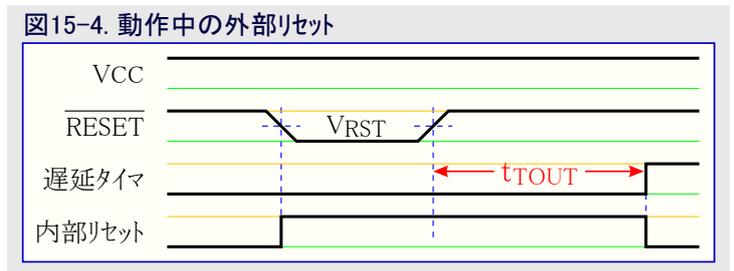
電源ONリセット(POR)パルスはチップ上の検出回路によって生成されます。POR信号はVCCが検出電圧以下の時は必ず活性(有効)にされます。POR回路は供給電圧異常検出は勿論、始動リセットの起動にも使えます。

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



### 15.4. 外部リセット

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

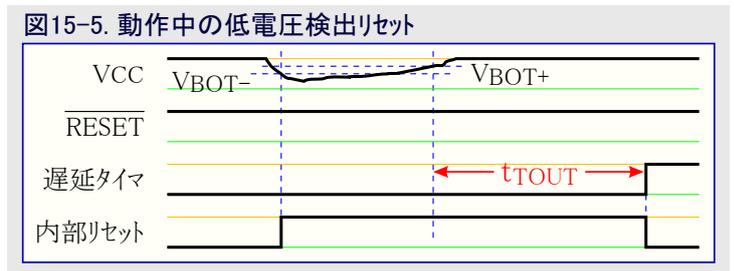


### 15.5. 低電圧(ブラウンアウト)検出リセット

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

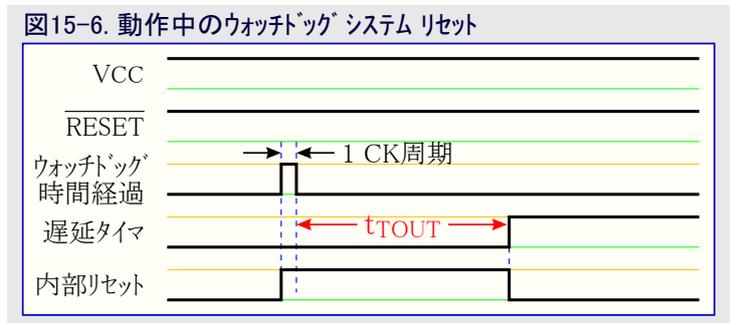
BODが許可され、VCCが起動電圧以下に下降すると(右図のV<sub>BOT-</sub>)、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(右図のV<sub>BOT+</sub>)、(遅延タイマが起動され)遅延タイマは遅延時間(t<sub>TOUT</sub>)経過後にMCUを始動します。

BOD回路は電圧がt<sub>BOD</sub>時間より長く起動電圧以下に留まる場合のみ、VCCでの低下を検出します。



### 15.6. ウォッチドッグ システムリセット

ウォッチドッグ時間経過時、(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は遅延時間(t<sub>TOUT</sub>)の計時を始めます。



## 15.7. 内部基準電圧

本デバイス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。

### 15.7.1. 基準電圧許可信号と起動時間

基準電圧には使われるべき方法に影響するかもしれない起動時間があります。節電のために、この基準電圧は常にONではありません。この基準電圧は次の状態中ONです。

1. 低電圧検出リセット許可時 (BODLEVELヒューズのプログラム(0)により)
2. アナログ比較器基準電圧接続時 (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)
3. A/D変換部動作許可時 (A/D変換制御/状態レジスタ(ADCSRA)のA/D動作許可(ADEN)=1)

従って低電圧検出(BOD)が許可されていないと、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。

## 15.8. ウォッチドッグ タイマ

応用でウォッチドッグ タイマが必要とされない場合は、この単位部をOFFにすべきです。ウォッチドッグ タイマが許可されたなら、全ての休止動作形態で許可され、故に常に電力を消費します。より深い休止動作形態で、これは総電流消費の重要な一因になるでしょう。

ウォッチドッグ タイマの構成設定方法の詳細については「ウォッチドッグ システム リセット」を参照してください。

### 15.8.1. 特徴

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

### 15.8.2. 概要

本デバイスは強化されたウォッチドッグ タイマ(WDT)を持ちます。このウォッチドッグ タイマ(WDT)はチップ上の独立した128kHz発振器の間隔で計時するタイマです。WDTは計数器が与えられた計時完了値に達した時に割り込みまたはシステム リセットを生じます。通常動作では計時完了値へ達する前に計数器を再始動するためにシステムはウォッチドッグ リセット(WDR)命令を使う必要があります。システムが計数器を再始動しなければ、割り込みまたはシステム リセットが起こるでしょう。

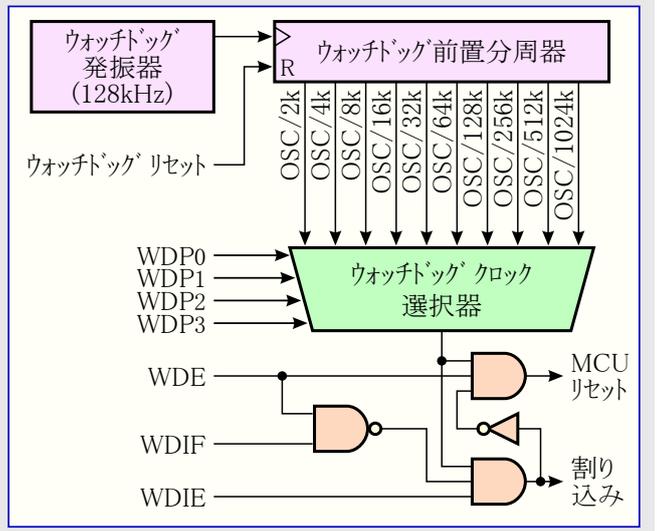
割り込み動作種別では、タイマ計時完了時にWDTが割り込みを生じます。この割り込みは休止形態からデバイスを起動するためや、一般的なシステムタイマとしても使えます。1つの例は或る動作に対して許された最大時間を制限することで、その動作が予測されたより長く走行する時に割り込みを生じます。システム リセット動作種別ではタイマ計時完了時にWDTがリセットを生じます。これは一般的にコード外走行の場合の中断を防止するのに使われます。3つ目の動作種別は先に割り込みを生じ、その後システム リセット動作種別に切り替えることで、他の2つの動作種別の組み合わせとなる、割り込み及びシステム リセット動作種別です。この動作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

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

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

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

図15-7. ウォッチドッグ タイマ構成図



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

#### アセンブリ言語プログラム例

```

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

#### C言語プログラム例

```

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

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

次のコード例はウォッチドッグ タイマの計時完了値を変更する方法を示します。

#### アセンブリ言語プログラム例

```

WDT_PR3:    CLI                ;全割り込み禁止
            WDR                ;ウォッチドッグ タイマ リセット
            LDS   R16, WDTCSR  ;現WDTCSR値を取得(他ビット保護用)
            ORI   R16, (1<<WDCE) | (1<<WDE) ;WDCEとWDE論理1値を設定
            STS   WDTCSR, R16  ;WDCEとWDEに論理1書き込み
            LDI   R16, (1<<WDE) | (1<<WDP2) | (1<<WDP0) ;WDE=1,計時間隔=0.5s値を取得
            STS   WDTCSR, R16  ;0.5s監視間隔リセット動作開始
            SEI                ;全割り込み許可
            RET                ;呼び出し元へ復帰
    
```

#### C言語プログラム例

```

void WDT_off(void)
{
    __disable_interrupt();          /* 全割り込み禁止 */
    __watchdog_reset();            /* ウォッチドッグ タイマ リセット */
    WDTCSR |= (1<<WDCE) | (1<<WDE); /* WDCEとWDEに論理1書き込み */
    WDTCSR = (1<<WDE) | (1<<WDP2) | (1<<WDP0); /* 0.5s監視間隔リセット動作開始 */
    __enable_interrupt();         /* 全割り込み許可 */
}
    
```

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

## 15.9. リセット関係レジスタ

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

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : MCUSR

変位 : \$54 (\$34)

リセット : \$0x (以下の説明を参照)

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$34です。

ビット	7	6	5	4	3	2	1	0
	–	–	–	–	WDRF	BORF	EXTRF	PORF
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	内容参照	内容参照	内容参照	内容参照

#### ● ビット3 – WDRF : ウォッチドッグ システム リセット フラグ (Watchdog System Reset Flag)

このビットはウォッチドッグ システム 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、それへの0書き込みによってリセット(0)されます。

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

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

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

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

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

このビットは電源ONリセットが起こると設定(1)されます。このビットはそれへの0書き込みによってのみリセット(0)されます。

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

名称 : WDTCSR

変位 : \$60

リセット : '0000x000'

特質 : –

ビット	7	6	5	4	3	2	1	0
	WDIF	WDIE	WDP3	WDCE	WDE	WDP2~0		
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	不定	0	0	0

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

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

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

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

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

表15-1. ウォッチドッグ タイマ設定

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

注: WDTONヒューズは0でプログラム、1で非プログラムに設定です。

● ビット5 - WDP3 : ウォッチドッグ タイマ前置分周選択ビット3 (Watchdog Timer Prescaler 3)

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

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

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

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

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

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

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

表15-2. ウォッチドッグ前置分周選択

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

## 16. INT – 割り込み (Interrupts)

この章は本デバイスの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「リセットと割り込みの扱い」を参照してください。

ATmega48/V/88/V/168/Vの割り込みベクタは次の違いを除いてほぼ同じです。

- 各割り込みベクタはATmega168/Vで2命令語、ATmega48/VとATmega88/Vで1命令語を占有します。
- ATmega48/Vは独立したブートローダ領域を持ちません。ATmega88/VとATmega168/Vでは、リセットベクタがBOOTRSTヒューズにより、割り込みベクタ開始アドレスがMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットによって影響を及ぼされます。

関連リンク [19頁の「リセットと割り込みの扱い」](#)

### 16.1. 割り込みベクタ

表16-1. リセットと割り込みのベクタ

ベクタ 番号	プログラム アドレス			発生元	備考
	ATmega48/V	ATmega88/V (注2)	ATmega168/V (注2)		
1	\$0000	\$0000 (注1)	\$0000 (注1)	RESET	電源ON, WDT, BOD等の各種リセット
2	\$0001	\$0001	\$0002	INT0	外部割り込み要求0
3	\$0002	\$0002	\$0004	INT1	外部割り込み要求1
4	\$0003	\$0003	\$0006	PCINT0 (PCI0)	ピン変化0群割り込み要求
5	\$0004	\$0004	\$0008	PCINT1 (PCI1)	ピン変化1群割り込み要求
6	\$0005	\$0005	\$000A	PCINT2 (PCI2)	ピン変化2群割り込み要求
7	\$0006	\$0006	\$000C	WDT	ウォッチドッグ計時完了
8	\$0007	\$0007	\$000E	TIMER2_COMPA	タイマ/カウンタ2比較A一致
9	\$0008	\$0008	\$0010	TIMER2_COMPB	タイマ/カウンタ2比較B一致
10	\$0009	\$0009	\$0012	TIMER2_OVF	タイマ/カウンタ2溢れ
11	\$000A	\$000A	\$0014	TIMER1_CAPT	タイマ/カウンタ1捕獲発生
12	\$000B	\$000B	\$0016	TIMER1_COMPA	タイマ/カウンタ1比較A一致
13	\$000C	\$000C	\$0018	TIMER1_COMPB	タイマ/カウンタ1比較B一致
14	\$000D	\$000D	\$001A	TIMER1_OVF	タイマ/カウンタ1溢れ
15	\$000E	\$000E	\$001C	TIMER0_COMPA	タイマ/カウンタ0比較A一致
16	\$000F	\$000F	\$001E	TIMER0_COMPB	タイマ/カウンタ0比較B一致
17	\$0010	\$0010	\$0020	TIMER0_OVF	タイマ/カウンタ0溢れ
18	\$0011	\$0011	\$0022	SPI_STC	SPI 転送完了
19	\$0012	\$0012	\$0024	USART_RX	USART 受信完了
20	\$0013	\$0013	\$0026	USART_UDRE	USART 送信緩衝部空き
21	\$0014	\$0014	\$0028	USART_TX	USART 送信完了
22	\$0015	\$0015	\$002A	ADC	A/D変換完了
23	\$0016	\$0016	\$002C	EE_RDY	EEPROM 操作可
24	\$0017	\$0017	\$002E	ANALOG_COMP	アナログ比較器出力遷移
25	\$0018	\$0018	\$0030	TWI	2線直列インターフェース状態変化
26	\$0019	\$0019	\$0032	SPM_READY	SPM命令操作可

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

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

(訳注) 原書の表16-1.,表16-2.,表16-4.は表16-1.として纏めました。

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

表16-3. リセットと割り込みベクタの配置

BOOTRST	IVSEL	リセットベクタアドレス	割り込みベクタ先頭アドレス	
			ATmega88/V	ATmega168/V
非プログラム(1)	0	\$0000	\$0001	\$0002
	1	\$0000	ブート領域先頭アドレス+\$0001	ブート領域先頭アドレス+\$0002
プログラム(0)	0	ブート領域先頭アドレス	\$0001	\$0002
	1	ブート領域先頭アドレス	ブート領域先頭アドレス+\$0001	ブート領域先頭アドレス+\$0002

注: ブート領域先頭アドレスは表31-7.で示されます。

(訳注) 原書の表16-3.と表16-5.は表16-3.として纏めました。  
本頁上記はATmega88/VとATmega168/Vにだけ適用されます。

ATmega48/V/88/Vでの最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000		RJMP RESET	;各種リセット
\$0001		RJMP EXT_INT0	;外部割り込み要求0
\$0002		RJMP EXT_INT1	;外部割り込み要求1
\$0003		RJMP PCINT0	;ピン変化0群割り込み要求
\$0004		RJMP PCINT1	;ピン変化1群割り込み要求
\$0005		RJMP PCINT2	;ピン変化2群割り込み要求
\$0006		RJMP WDT_OVF	;ウォッチドッグ計時完了
\$0007		RJMP TIM2_COMPA	;タイマ/カウンタ2比較A一致
\$0008		RJMP TIM2_COMPB	;タイマ/カウンタ2比較B一致
\$0009		RJMP TIM2_OVF	;タイマ/カウンタ2溢れ
\$000A		RJMP TIM1_CAPT	;タイマ/カウンタ1捕獲発生
\$000B		RJMP TIM1_COMPA	;タイマ/カウンタ1比較A一致
\$000C		RJMP TIM1_COMPB	;タイマ/カウンタ1比較B一致
\$000D		RJMP TIM1_OVF	;タイマ/カウンタ1溢れ
\$000E		RJMP TIM0_COMPA	;タイマ/カウンタ0比較A一致
\$000F		RJMP TIM0_COMPB	;タイマ/カウンタ0比較B一致
\$0010		RJMP TIM0_OVF	;タイマ/カウンタ0溢れ
\$0011		RJMP SPI_STC	;SPI転送完了
\$0012		RJMP USART_RXC	;USART受信完了
\$0013		RJMP USART_UDRE	;USART送信緩衝部空
\$0014		RJMP USART_TXC	;USART送信完了
\$0015		RJMP ADC	;ADC変換完了
\$0016		RJMP EE_RDY	;EEPROM操作可
\$0017		RJMP ANA_COMP	;アナログ比較器出力遷移
\$0018		RJMP TWI	;2線直列インターフェース状態変化
\$0019		RJMP SPM_RDY	;SPM命令操作可
		;	
\$001A	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$001B		OUT SPH, R16	;スタックポインタ上位を初期化
\$001C		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$001D		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

ATmega88/VでBOOTRSTヒューズが非プログラム(1)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なりセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
			;各種リセット (BOOTRSTヒューズ=1)
\$0000	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (応用プログラム開始)
\$0001		OUT SPH, R16	;スタックポインタ上位を初期化
\$0002		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0003		OUT SPL, R16	;スタックポインタ下位を初期化
		}	;以下、I/O初期化など
	.ORG	\$0C01	;ブートプログラム領域が2Kバイトの場合
\$0C01		RJMP EXT_INT0	;外部割り込み要求0
\$0C02		RJMP EXT_INT1	;外部割り込み要求1
		}	
\$0C19		RJMP SPM_RDY	;SPM命令操作可

ATmega88/VでBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定される時の最も代表的且つ一般的なりセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		.ORG \$0001	;割り込みベクタ先頭
\$0001		RJMP EXT_INT0	;外部割り込み要求0
\$0002		RJMP EXT_INT1	;外部割り込み要求1
		}	
\$0019		RJMP SPM_RDY	;SPM命令操作可
		}	;以下、プログラムなど
	.ORG	\$0C00	;ブートプログラム領域が2Kバイトの場合
			;各種リセット (BOOTRSTヒューズ=0)
\$0C00	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (プログラム開始)
\$0C01		OUT SPH, R16	;スタックポインタ上位を初期化
\$0C02		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0C03		OUT SPL, R16	;スタックポインタ下位を初期化
		}	;以下、I/O初期化など

ATmega88/VでBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なりセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		.ORG \$0C00	;ブートプログラム領域が2Kバイトの場合
\$0C00		RJMP RESET	;各種リセット (BOOTRSTヒューズ=0)
\$0C01		RJMP EXT_INT0	;外部割り込み要求0
\$0C02		RJMP EXT_INT1	;外部割り込み要求1
		}	
\$0C19		RJMP SPM_RDY	;SPM命令操作可
;			
\$0C1A	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (プログラム開始)
\$0C1B		OUT SPH, R16	;スタックポインタ上位を初期化
\$0C1C		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0C1D		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

ATmega168/Vでの最も代表的且つ一般的なリセットと割り込みのベクタ アドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000		JMP RESET	; 各種リセット
\$0002		JMP EXT_INT0	; 外部割り込み要求0
\$0004		JMP EXT_INT1	; 外部割り込み要求1
\$0006		JMP PCINT0	; ピン変化0群割り込み要求
\$0008		JMP PCINT1	; ピン変化1群割り込み要求
\$000A		JMP PCINT2	; ピン変化2群割り込み要求
\$000C		JMP WDT_OVF	; ウォッチドッグ計時完了
\$000E		JMP TIM2_COMPA	; タイマ/カウンタ2比較A一致
\$0010		JMP TIM2_COMPB	; タイマ/カウンタ2比較B一致
\$0012		JMP TIM2_OVF	; タイマ/カウンタ2溢れ
\$0014		JMP TIM1_CAPT	; タイマ/カウンタ1捕獲発生
\$0016		JMP TIM1_COMPA	; タイマ/カウンタ1比較A一致
\$0018		JMP TIM1_COMPB	; タイマ/カウンタ1比較B一致
\$001A		JMP TIM1_OVF	; タイマ/カウンタ1溢れ
\$001C		JMP TIM0_COMPA	; タイマ/カウンタ0比較A一致
\$001E		JMP TIM0_COMPB	; タイマ/カウンタ0比較B一致
\$0020		JMP TIM0_OVF	; タイマ/カウンタ0溢れ
\$0022		JMP SPI_STC	; SPI転送完了
\$0024		JMP USART_RXC	; USART 受信完了
\$0026		JMP USART_UDRE	; USART 送信緩衝部空
\$0028		JMP USART_TXC	; USART 送信完了
\$002A		JMP ADC	; ADC変換完了
\$002C		JMP EE_RDY	; EEPROM操作可
\$002E		JMP ANA_COMP	; アナログ比較器出力遷移
\$0030		JMP TWI	; 2線直列インターフェース状態変化
\$0032		JMP SPM_RDY	; SPM命令操作可
;			
\$0034	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得
\$0035		OUT SPH, R16	; スタックポインタ上位を初期化
\$0036		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$0037		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

ATmega168/VでBOOTRSTヒューズが非プログラム(1)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
\$0000	RESET:	LDI R16, HIGH (RAMEND)	; 各種リセット (BOOTRSTヒューズ=1)
\$0001		OUT SPH, R16	; RAM最終アドレス上位を取得 (応用プログラム開始)
\$0002		LDI R16, LOW (RAMEND)	; スタックポインタ上位を初期化
\$0003		OUT SPL, R16	; RAM最終アドレス下位を取得
		}	; スタックポインタ下位を初期化
		}	; 以下、I/O初期化など
	. ORG	\$1C02/\$3C02	; ブートプログラム領域が2Kバイトの場合
\$1C02/\$3C02		JMP EXT_INT0	; 外部割り込み要求0
\$1C04/\$3C04		JMP EXT_INT1	; 外部割り込み要求1
		}	
\$1C32/\$3C32		JMP SPM_RDY	; SPM命令操作可

ATmega168/VでBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定される時の最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		.ORG \$0002	; 割り込みベクタ先頭
\$0002		JMP EXT_INT0	; 外部割り込み要求0
\$0004		JMP EXT_INT1	; 外部割り込み要求1
		}	
\$0032		JMP SPM_RDY	; SPM命令操作可 ; 以下、プログラムなど
		}	
		.ORG \$1C00/\$3C00	; ブートプログラム領域が2Kバイトの場合
			; 各種リセット (BOOTRSTヒューズ=0)
\$1C00/\$3C00	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$1C01/\$3C01		OUT SPH, R16	; スタックポインタ上位を初期化
\$1C02/\$3C02		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$1C03/\$3C03		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

ATmega168/VでBOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立ってMCU制御レジスタ(MCUCR)の割り込みベクタ選択(IVSEL)ビットが設定(1)される時の最も代表的且つ一般的なリセットと割り込みのベクタアドレス用設定を次に示します。

アドレス	ラベル	命令	注釈
		.ORG \$1C00/\$3C00	; ブートプログラム領域が2Kバイトの場合
\$1C00/\$3C00		JMP RESET	; 各種リセット (BOOTRSTヒューズ=0)
\$1C02/\$3C02		JMP EXT_INT0	; 外部割り込み要求0
\$1C04/\$3C04		JMP EXT_INT1	; 外部割り込み要求1
		}	
\$1C32/\$3C32		JMP SPM_RDY	; SPM命令操作可
		}	
\$1C34/\$3C34	RESET:	LDI R16, HIGH (RAMEND)	; RAM最終アドレス上位を取得 (プログラム開始)
\$1C35/\$3C35		OUT SPH, R16	; スタックポインタ上位を初期化
\$1C36/\$3C36		LDI R16, LOW (RAMEND)	; RAM最終アドレス下位を取得
\$1C37/\$3C37		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

## 16.2. 割り込みベクタ移動関係レジスタ

### 16.2.1. 応用領域とブート領域間の割り込みベクタ移動

MCU制御レジスタ(MCUCR)は割り込みベクタ表の配置を制御します。(ATmega88/VとATmega168/Vに対して)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : MCUCR

変位 : \$55 (\$35)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$35です。

ビット	7	6	5	4	3	2	1	0
	-	-	-	PUD	-	-	IVSEL	IVCE
アクセス種別	R	R	R	R/W	R	R	R/W	R/W
リセット値	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されます。ブートフラッシュ領域先頭の実際のアドレスはBOOTSZレジスタによって決定されます。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を書いてください。
2. 4周期内に、望む値をIVSELに書き、同時にIVCEへ0を書いてください。

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を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。

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

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

##### アセンブリ言語プログラム例

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

##### C言語プログラム例

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

## 17. EXTINT – 外部割り込み (External Interrupts)

外部割り込みはINTピンまたはPCINTピンの何れかによって起動されます。許可したなら、例えINTピンまたはPCINT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。

**PCI2ピン変化割り込み**は許可したPCINT16~23の何れかが切り替わると起動します。**PCI1ピン変化割り込み**は許可したPCINT8~14の何れかが切り替わると起動します。**PCI0ピン変化割り込み**は許可したPCINT0~7の何れかが切り替わると起動します。ピン変化割り込み許可レジスタn(PCMSK0,PCMSK1,PCMSK2)は、どのピンがピン変化割り込み要因となるかを制御します。PCINTnでのピン変化割り込みは非同期に検知されます。これはそれらの割り込みがアイドル動作以外の休止形態からもデバイス起動するのに使えることを意味します。

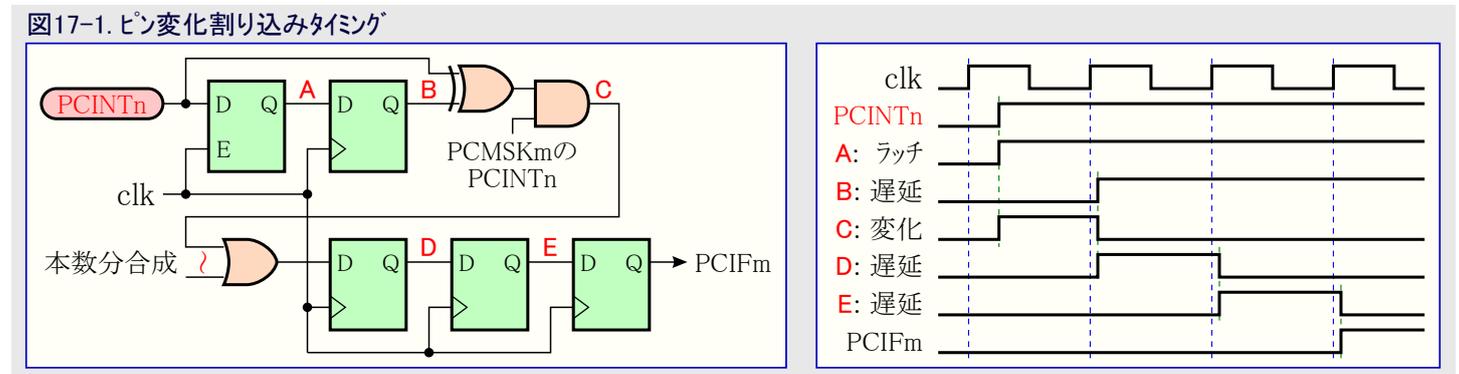
INT割り込みは上昇端または下降端(含む両端)、またはLowレベルによって起動できます。これは**外部割り込み制御レジスタA(EICRA)**の詳述で示されるように構成設定されます。INT割り込みがレベル起動として設定、且つ許可されると、そのピンがLowに保持される限り、割り込みは(継続的に)起動します。INTの上昇端や下降端割り込みの認知はI/Oクロックの存在を必要とすることに注意してください。INTのLowレベル割り込みは非同期に検知されます。これはこの割り込みが**アイドル動作**以外の**休止形態**からもデバイス起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

**注:** レベル起動割り込みが**パワーダウン動作**からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが起動時間の最後に先立って消滅すると、MCUは今まで通り起動しますが、割り込みが生成されません。起動時間は**SUT**と**CKSEL**のヒューズで定義されます。

関連リンク 28頁の「システムクロックとクロック選択」

### 17.1. ピン変化割り込みタイミング

ピン変化割り込みの例は下図で示されます。



## 17.2. 外部割り込み用レジスタ

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

外部割り込み制御レジスタAは割り込み条件制御ビットを含みます。

名称 : EICRA  
 変位 : \$69  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	ISC11,0		ISC01,0	
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	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)

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**外部割り込み許可レジスタ(EIMSK)の外部割り込み1許可(INT1)ビット**が設定(1)される場合のINT1外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT1ピンの端(エッジ)とレベルは右表で定義されます。INT1ピン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パルスは割り込みを生成します。より短いパルスは割り込みの発生が保証されません。Lowレベル割り込みが選ばれる場合、そのLowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表17-1. 外部割り込み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(両端)。
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)**

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)ビットと外部割り込み許可レジスタ(EIMSK)の外部割り込み0許可(INT0)ビットが設定(1)される場合のINT0外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT0ピンの端(エッジ)とレベルは右表で定義されます。INT0ピン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パルスは割り込みを生成します。より短いパルスは割り込みの発生が保証されません。Lowレベル割り込みが選ばれる場合、そのLowレベルは割り込みを生成するために現在実行している命令の完了まで保たれなければなりません。

表17-2. 外部割り込み0(INT0)割り込み条件

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。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : EIMSK

変位 : \$3D (\$1D)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$1Dです。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	INT1	INT0
アクセス種別	R	R	R	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

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

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : EIFR

変位 : \$3C (\$1C)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$1Cです。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	INTF1	INTF0
アクセス種別	R	R	R	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● **ビット1 – INTF1 : 外部割り込み1要求フラグ (External Interrupt Flag 1)**

INT1ピン上の端(エッジ)または論理変化が割り込み要求を起動すると、INTF1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)ビットと外部割り込み許可レジスタ(EIMSK)の外部割り込み1許可(INT1)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは1を書くことによっても解除(0)できます。INT1がレベル割り込みとして設定されると、このフラグは常に解除(0)されます。

● **ビット0 – INTF0 : 外部割り込み0要求フラグ (External Interrupt Flag 0)**

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

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

名称 : PCICR  
 変位 : \$68  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	PCIE2	PCIE1	PCIE0
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

### ● ビット2 – PCIE2 : ピン変化2群割り込み許可 (Pin Change Interrupt Enable 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、PCIE2ビットが設定(1)されると、ピン変化2群割り込みが許可されます。許可したPCINT16～23ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みはPCI2割り込みベクタから実行されます。PCINT16～23ピンはピン変化割り込み許可レジスタ2(PCMSK2)によって個別に許可されます。

### ● ビット1 – PCIE1 : ピン変化1群割り込み許可 (Pin Change Interrupt Enable 1)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE1ビットが設定(1)されると、ピン変化1群割り込みが許可されます。許可したPCINT8～14ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みはPCI1割り込みベクタから実行されます。PCINT8～14ピンはピン変化割り込み許可レジスタ1(PCMSK1)によって個別に許可されます。

### ● ビット0 – PCIE0 : ピン変化0群割り込み許可 (Pin Change Interrupt Enable 0)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、PCIE0ビットが設定(1)されると、ピン変化0群割り込みが許可されます。許可したPCINT0～7ピンの何れかの変化が割り込みを起こします。このピン変化割り込み要求に対応する割り込みはPCI0割り込みベクタから実行されます。PCINT0～7ピンはピン変化割り込み許可レジスタ0(PCMSK0)によって個別に許可されます。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PCIFR  
 変位 : \$3B (\$1B)  
 リセット : \$00  
 特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$1Bです。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	PCIF2	PCIF1	PCIF0
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

### ● ビット2 – PCIF2 : ピン変化2群割り込み要求フラグ (Pin Change Interrupt Flag 2)

PCINT16～23ピンの何れかの論理変化が割り込み要求を起動すると、PCIF2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)ビットとピン変化割り込み制御レジスタ(PCICR)のピン変化2群割り込み許可(PCIE2)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは1を書くことによっても解除(0)できます。

### ● ビット1 – PCIF1 : ピン変化1群割り込み要求フラグ (Pin Change Interrupt Flag 1)

PCINT8～14ピンの何れかの論理変化が割り込み要求を起動すると、PCIF1が設定(1)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビットとピン変化割り込み制御レジスタ(PCICR)のピン変化1群割り込み許可(PCIE1)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込み処理ルーチンが開始されると解除(0)されます。代わりにこのフラグは1を書くことによっても解除(0)できます。

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

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

### 17.2.6. PCMSK2 – ピン変化割り込み許可レジスタ2 (Pin Change Mask Register 2)

名称 : PCMSK2  
 変位 : \$6D  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – PCINT23~PCINT16 : ピン変化割り込み23~16許可 (Pin Change Enable Mask 23~16)

各PCINT16~23ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT16~23とピン変化割り込み制御レジスタ(PCICR)のPCIE2が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT16~23が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

### 17.2.7. PCMSK1 – ピン変化割り込み許可レジスタ1 (Pin Change Mask Register 1)

名称 : PCMSK1  
 変位 : \$6C  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
アクセス種別	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット6~0 – PCINT14~PCINT8 : ピン変化割り込み14~8許可 (Pin Change Enable Mask 14~8)

各PCINT8~14ビットは対応するI/Oピンでピン変化割り込みが許可されるかどうかを選びます。PCINT8~14とピン変化割り込み制御レジスタ(PCICR)のPCIE1が設定(1)なら、対応するI/Oピンのピン変化割り込みが許可されます。PCINT8~14が解除(0)されると、対応するI/Oピンのピン変化割り込みは禁止されます。

### 17.2.8. PCMSK0 – ピン変化割り込み許可レジスタ0 (Pin Change Mask Register 0)

名称 : PCMSK0  
 変位 : \$6B  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

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

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

## 18. 入出力ポート

### 18.1. 概要

全てのAVRのポートは標準デジタルI/Oポートとして使われる時に真の読み-変更-書き(リード-モディファイ-ライト)を機能的に持ちます。これはSBIとCBI命令で他のどのピンの方角をも無意識に変更することなく、1つのポートピンの方角を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。このピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピンに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があります。全てのI/Oピンは右図で示されるようにVCCとGNDの両方に保護ダイオードを持ちます。

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

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

標準デジタルI/OとしてのI/Oポートの使用は次項で記述されます。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本章内の「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を及ぼされません。

### 18.2. 標準デジタル入出力としてのポート

このポートは任意の内部プルアップ付き双方向I/Oポートです。下図はここで属にPxnと呼ばれるI/Oポートピンの1つの機能説明を示します。

図18-1. 入出力ピン等価回路

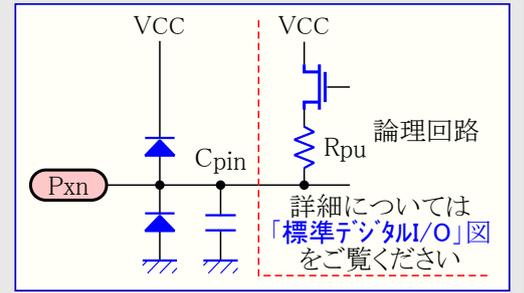
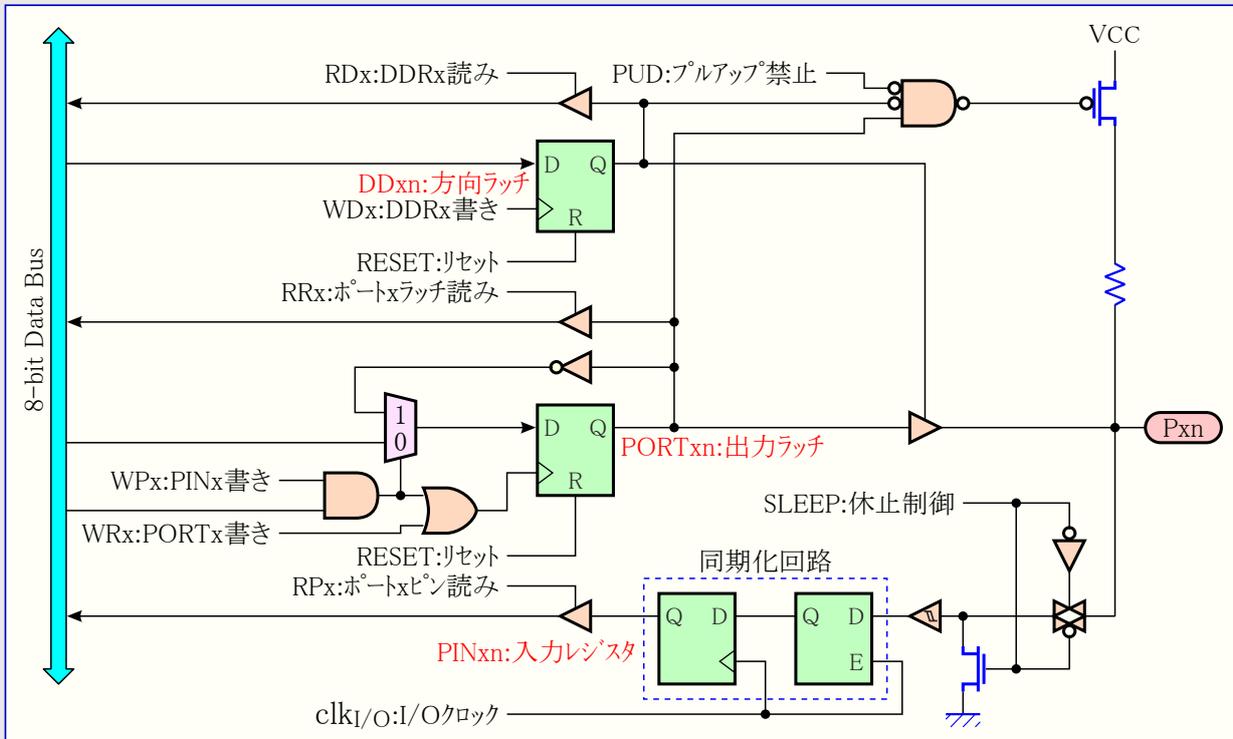


図18-2. 標準デジタル入出力回路構成



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

### 18.2.1. ピンの設定

各ポートピンは3つのレジスタビット、DDxn、PORTxn、PINxnから成ります。「I/Oポート用レジスタ」で示されるようにDDxnビットはDDRx I/Oアドレス、PORTxnビットはPORTx I/Oアドレス、PINxnビットはPINx I/Oアドレスでアクセスされます。

DDRxレジスタ内のDDxnビットはそのピンの方向を選びます。DDxnが1を書かれるとPxnは出力ピンとして設定されます。DDxnが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1を書かれると、そのポートピンはHigh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0を書かれると、そのポートピンはLowに駆動されます。

### 18.2.2. ピンの出力交互切り替え

PINxnへの1書き込みはDDRxnの値に拘らず、PORTxnの値を反転切り替えます。SBI命令はポート内の1ビットの反転切り替えに使うことができます。

### 18.2.3. 入出力間の切り替え

Hi-Z入力(DDxn=0, PORTxn=0)とHigh出力(DDxn=1, PORTxn=1)間の切り替え時、プルアップ許可入力(DDxn=0, PORTxn=1)またはLow出力(DDxn=1, PORTxn=0)のどちらかの中間状態が生じるに違いありません。通常、高インピーダンス環境は強力なHigh(吐き出し)駆動部とプルアップ間の違いに気付かないので、プルアップが許可された状態は十分受け入れられます。この事例でないなら、全ポートの全プルアップを禁止するために、MCU制御レジスタ(MCUCR)のプルアップ禁止(PUD)ビットが設定(1)できます。

プルアップ入力とLow出力間の切り替えは同じ問題を発生します。使用者は中間状態としてHi-Z入力(DDxn=0, PORTxn=0)またはHigh出力(DDxn=1, PORTxn=1)のどちらかを使わなければなりません。

下表はピン値に対する制御信号の一覧を示します。

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

### 18.2.4. ピン値の読み込み

DDxn方向ビットの設定に関係なく、ポートピンはPINxnレジスタビットを通して読めます。「標準デジタル入出力としてのポート」で示されるようにPINxnレジスタビットと先行するラッチは同期化回路を構成します。これは物理ピンが内部クロックの端(エッジ)付近で値を変える場合の未定義状態(メタステーブル)を避けるために必要とされますが、それは遅延も持ち込みます。右図は外部的に加えられたピン値を読む時の同期化タイミング図を示します。伝播遅延の最小と最大は各々 $t_{pd,min}$ と $t_{pd,max}$ で示されます。

(右図で)システムクロックの最初の下降端直後から始まるクロック周期を考察してください。このラッチはクロックがLowの時に閉じ、クロックがHighの時に同期ラッチ信号の斜線部分で示されるように通過(トランスペアレント)となります。この信号値はシステムクロックがLowになる時に保持(ラッチ)されます。それが続くクロックの上昇端でPINxnレジスタに取り込まれます。2つの矢印 $t_{pd,min}$ と $t_{pd,max}$ によって示されるように、ピン上の単一信号遷移は出現時点に依存して0.5~1.5システムクロック周期遅らされます。

ソフトウェアが指定したピン値を読み戻す時は、右図で示されるようにNOP命令が挿入されなければなりません。OUT命令はシステムクロックの上昇端で同期ラッチを設定します。この場合、同期化回路を通過する遅延時間( $t_{pd}$ )は1システムクロック周期です。

図18-3. 外部供給ピン値読み込み時の同期化

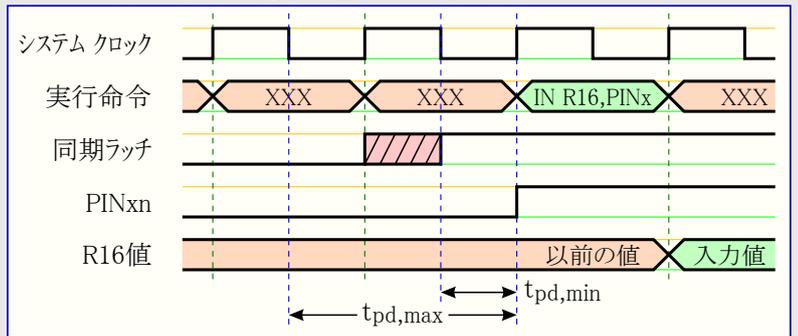
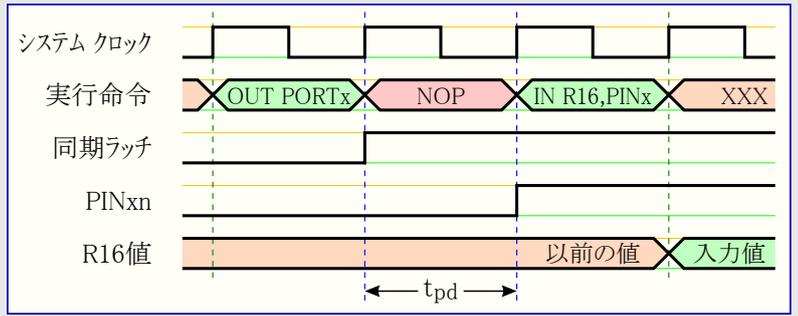


図18-4. プログラムで設定したピン値読み戻し時の同期化



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

#### アセンブリ言語プログラム例

```

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

#### C言語プログラム例

```

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

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

### 18.2.5. デジタル入力許可と休止形態

図18-2.で示されるようにデジタル入力信号はシュmittトリガの入力をGNDにクランプできます。この図でSLEEPと印された信号は入力信号のいくつかが開放のまま、またはVCC/2付近のアナログ信号電圧を持つ場合の高消費電力を避けるため、**パワーダウン動作**、**パワーセーフ動作**、**スタンバイ動作**、**拡張スタンバイ動作**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は本章内の「**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。

### 18.2.6. 未接続ピン

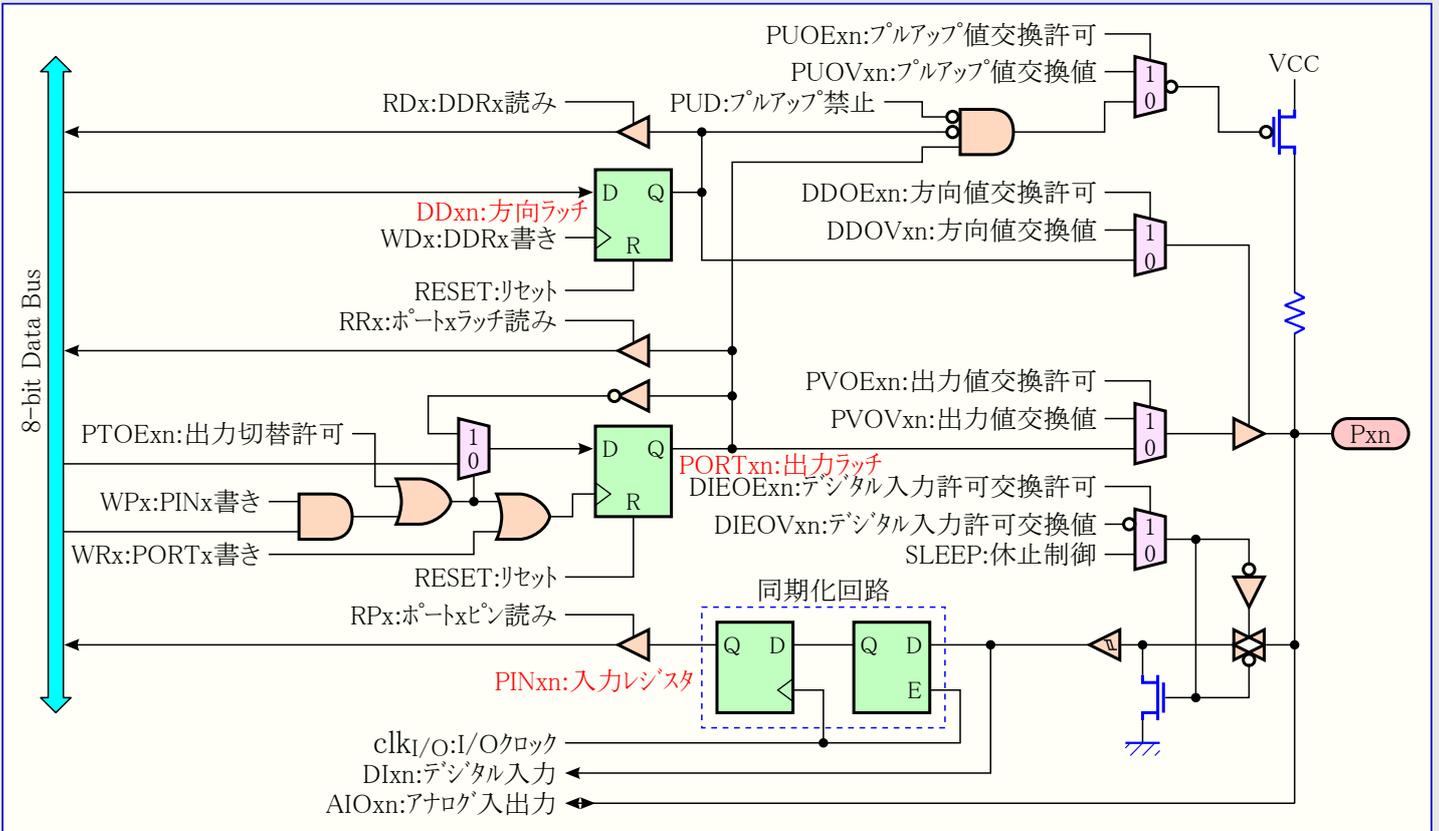
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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

### 18.3. 交換ポート機能

多くのポートピンには標準デジタル入出力に加え交換機能があります。下図は単純化された図18-2.でのポートピン制御信号が交換機能によってどう重複できるかを示します。この重複信号は全てのポートピンに存在する訳ではありませんが、この図はAVR マイクロ コントローラシステムの全ポートピンに適用できる一般的な記述として取り扱います。

図18-5. 交換ポート機能入出力回路構成



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

下表は重複(交換)信号の機能一覧を示します。前の図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18-2. 交換機能用交換信号の一般定義

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

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

### 18.3.1. ポートBの交換機能

ポートBピンの交換機能は下表で示されます。

表18-3. ポートBピンの交換機能

ポートピン	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) PCINT7 (ピン変化割り込み7入力)
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) PCINT6 (ピン変化割り込み6入力)
PB5	SCK (SPI直列クロック 主装置側出力/従装置側入力) PCINT5 (ピン変化割り込み5入力)
PB4	MISO (SPI主装置側データ入力/従装置側データ出力) PCINT4 (ピン変化割り込み4入力)
PB3	MOSI (SPI主装置側データ出力/従装置側データ入力) OC2A (タイマ/カウンタ2 比較A一致出力) PCINT3 (ピン変化割り込み3入力)
PB2	$\overline{SS}$ (SPI従装置選択入力) OC1B (タイマ/カウンタ1 比較B一致出力) PCINT2 (ピン変化割り込み2入力)
PB1	OC1A (タイマ/カウンタ1 比較A一致出力) PCINT1 (ピン変化割り込み1入力)
PB0	ICP1 (タイマ/カウンタ1 捕獲起動入力) CLKO (システムクロック出力) PCINT0 (ピン変化割り込み0入力)

交換ピンの設定は次のとおりです。

#### • XTAL2/TOSC2/PCINT7 - 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。クリスタル発振器使用のタイマ/カウンタ2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、外部クロック信号許可(EXCLK)ビットが解除(0)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。
- PCINT7 : ピン変化割り込み7入力。PB7ピンは外部割り込み元としても扱えます。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/PCINT6 - 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。クリスタル発振器使用のタイマ/カウンタ2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、外部クロック信号許可(EXCLK)ビットが解除(0)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。
- PCINT6 : ピン変化割り込み6入力。PB6ピンは外部割り込み元としても扱えます。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/PCINT5 - 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、本ピンはポートB方向レジスタ(DDRB)のDDB5設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB5によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB5によって制御できます。
- PCINT5 : ピン変化割り込み5入力。PB5ピンは外部割り込み元としても扱えます。

- MISO/PCINT4 – 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、本ピンはポート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によって制御されます。このピンがSPIによって入力を強制されるとき、プルアップは未だポートB出力レジスタ(PORTB)のPORTB4によって制御できます。
- PCINT4 : ピン変化割り込み4入力。PB4ピンは外部割り込み元としても扱えます。

- MOSI/OC2A/PCINT3 – 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、本ピンはポートB方向レジスタ(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PORTB3によって制御できます。
- OC2A : タイマ/カウンタ2の比較A一致出力。PB3ピンはタイマ/カウンタ2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、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OC2Aピンはタイマ機能のPWM動作用出力ピンでもありません。
- PCINT3 : ピン変化割り込み3入力。PB3ピンは外部割り込み元としても扱えます。

- $\overline{SS}$ /OC1B/PCINT2 – ポートB ビット2 : PB2

- $\overline{SS}$  : SPI従装置選択入力。SPIが従装置として許可されると、本ピンはポートB方向レジスタ(DDRB)の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せん。
- PCINT2 : ピン変化割り込み2入力。PB2ピンは外部割り込み元としても扱えます。

- OC1A/PCINT1 – 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せん。
- PCINT1 : ピン変化割り込み1入力。PB1ピンは外部割り込み元としても扱えます。

- ICP1/CLKO/PCINT0 – 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。
- CLKO : システム クロック出力。分周したシステム クロックがPB0ピンに出力できます。分周したシステム クロックはCKOUTヒューズがプログラム(0)されると、PORTB0とDDB0設定に拘らず、出力されます。これはリセット中にも出力されます。
- PCINT0 : ピン変化割り込み0入力。PB0ピンは外部割り込み元としても扱えます。

下表はポートBの交換機能を図18-5.で示される交換信号に関連付けます。SPI主装置入力とSPI従装置出力がMISOを構成し、一方MOSIはSPI主装置出力とSPI従装置入力に分けられます。

表18-4. ポートB7~4の交換機能用交換信号

信号名	PB7/XTAL2/ TOSC2/PCINT7	PB6/XTAL1/ TOSC1/PCINT6	PB5/SCK/PCINT5	PB4/MISO/PCINT4
PUOE	$\overline{\text{INTRC}} \cdot \overline{\text{EXTCK}} + \text{AS2}$	$\overline{\text{INTRC}} + \text{AS2}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \text{MSTR}$
PUOV	0	0	$\text{PORTB5} \cdot \text{PUD}$	$\text{PORTB4} \cdot \text{PUD}$
DDOE	$\overline{\text{INTRC}} \cdot \overline{\text{EXTCK}} + \text{AS2}$	$\overline{\text{INTRC}} + \text{AS2}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \text{MSTR}$
DDOV	0	0	0	0
PVOE	0	0	$\text{SPE} \cdot \text{MSTR}$	$\text{SPE} \cdot \text{MSTR}$
PVOV	0	0	SCK出力	SPI従装置出力
PTOE	-	-	-	-
DIEOE	$\overline{\text{INTRC}} \cdot \overline{\text{EXTCK}} + \text{AS2} + \text{PCINT7} \cdot \text{PCIE0}$	$\overline{\text{INTRC}} + \text{AS2} + \text{PCINT6} \cdot \text{PCIE0}$	$\text{PCINT5} \cdot \text{PCIE0}$	$\text{PCINT4} \cdot \text{PCIE0}$
DIEOV	$(\overline{\text{INTRC}} + \overline{\text{EXTCK}}) \cdot \text{AS2}$	$\overline{\text{INTRC}} + \text{AS2}$	1	1
DI	PCINT7入力	PCINT6入力	SCK/PCINT5入力	SPI主装置/PCINT4入力
AIO	発振増幅器出力	発振増幅器入力/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. EXTCKは外部クロック信号が(CKSELヒューズによって)選ばれることを意味します。

表18-5. ポートB3~0の交換機能用交換信号

信号名	PB3/MOSI/OC2A/PCINT3	PB2/ $\overline{\text{SS}}$ /OC1B/PCINT2	PB1/OC1A/PCINT1	PB0/ICP1/CLKO/PCINT0
PUOE	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	0	0
PUOV	$\text{PORTB3} \cdot \text{PUD}$	$\text{PORTB2} \cdot \text{PUD}$	0	0
DDOE	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	0	0
DDOV	0	0	0	0
PVOE	$\text{SPE} \cdot \overline{\text{MSTR}} + \text{OC2A}$ 許可	OC1B許可	OC1A許可	0
PVOV	SPI主装置出力+OC2A	OC1B	OC1A	0
PTOE	-	-	-	-
DIEOE	$\text{PCINT3} \cdot \text{PCIE0}$	$\text{PCINT2} \cdot \text{PCIE0}$	$\text{PCINT1} \cdot \text{PCIE0}$	$\text{PCINT0} \cdot \text{PCIE0}$
DIEOV	1	1	1	1
DI	SPI従装置/PCINT3入力	SPI $\overline{\text{SS}}$ /PCINT2入力	PCINT1入力	ICP1/PCINT0入力
AIO	-	-	-	-

### 18.3.2. ポートCの交換機能

ポートCピンの交換機能は下表で示されます。

表18-6. ポートCピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PC6	RESET (リセットピン) PCINT14 (ピン変化割り込み14入力)	PC3	ADC3 (A/D変換チャネル3入力) PCINT11 (ピン変化割り込み11入力)
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) PCINT13 (ピン変化割り込み13入力)	PC2	ADC2 (A/D変換チャネル2入力) PCINT10 (ピン変化割り込み10入力)
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) PCINT12 (ピン変化割り込み12入力)	PC1	ADC1 (A/D変換チャネル1入力) PCINT9 (ピン変化割り込み9入力)
		PC0	ADC0 (A/D変換チャネル0入力) PCINT8 (ピン変化割り込み8入力)

交換ピンの設定は次のとおりです。

#### • RESET/PCINT14 – 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、このピンは標準の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。

- **PCINT14** : ピン変化割り込み14入力。PC6ピンは外部割り込み元としても扱えます。

#### • SCL/ADC5/PCINT13 – 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ジスタ(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(標準の)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クロック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)を消去するためのスパイク除去器があり、スルーレート(上昇/下降速度)制限付きオープンド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。

- **PCINT13** : ピン変化割り込み13入力。PC5ピンは外部割り込み元としても扱えます。

#### • SDA/ADC4/PCINT12 – 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ジスタ(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(標準の)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)を消去するためのスパイク除去器があり、スルーレート(上昇/下降速度)制限付きオープンド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。

- **PCINT12** : ピン変化割り込み12入力。PC4ピンは外部割り込み元としても扱えます。

#### • ADC3/PCINT11 – 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。

- **PCINT11** : ピン変化割り込み11入力。PC3ピンは外部割り込み元としても扱えます。

#### • ADC2/PCINT10 – 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。

- **PCINT10** : ピン変化割り込み10入力。PC2ピンは外部割り込み元としても扱えます。

#### • ADC1/PCINT9 – 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。

- **PCINT9** : ピン変化割り込み9入力。PC1ピンは外部割り込み元としても扱えます。

#### • ADC0/PCINT8 – ポートCビット0 : PC0

- **ADC0** : 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。

- **PCINT8** : ピン変化割り込み8入力。PC0ピンは外部割り込み元としても扱えます。

下表はポートCの交換機能を図18-5.で示される交換信号に関連付けます。

表18-7. ポートC6~4の交換機能用交換信号

信号名	PC6/RESET/PCINT14	PC5/SCL/ADC5/PCINT13	PC4/SDA/ADC4/PCINT12
PUOE	RSTDISBL	TWEN	TWEN
PUOV	1	PORTC5•PUD	PORTC4•PUD
DDOE	RSTDISBL	TWEN	TWEN
DDOV	0	SCL出力	SDA出力
PVOE	0	TWEN	TWEN
PVOV	0	0	0
PTOE	-	-	-
DIEOE	RSTDISBL+PCINT14•PCIE1	ADC5D+PCINT13•PCIE1	ADC4D+PCINT12•PCIE1
DIEOV	RSTDISBL	PCINT13•PCIE1	PCINT12•PCIE1
DI	PCINT14入力	PCINT13入力	PCINT12入力
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ピン出力のスレーブ制御を許可します。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18-8. ポートC3~0の交換機能用交換信号

信号名	PC3/ADC3/PCINT11	PC2/ADC2/PCINT10	PC1/ADC1/PCINT9	PC0/ADC0/PCINT8
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	-	-	-	-
DIEOE	ADC3D+PCINT11•PCIE1	ADC2D+PCINT10•PCIE1	ADC1D+PCINT9•PCIE1	ADC0D+PCINT8•PCIE1
DIEOV	PCINT11•PCIE1	PCINT10•PCIE1	PCINT9•PCIE1	PCINT8•PCIE1
DI	PCINT11入力	PCINT10入力	PCINT9入力	PCINT8入力
AIO	ADC3入力	ADC2入力	ADC1入力	ADC0入力

### 18.3.3. ホートDの交換機能

ホートDピンの交換機能は下表で示されます。

表18-9. ホートDピンの交換機能

ホートピン	交換機能	ホートピン	交換機能
PD7	AIN1 (アナログ比較器反転入力) PCINT23 (ピン変化割り込み23入力)	PD3	INT1 (外部割り込み1 入力) OC2B (タイマ/カウンタ2 比較B一致出力) PCINT19 (ピン変化割り込み19入力)
PD6	AIN0 (アナログ比較器非反転入力) OC0A (タイマ/カウンタ0 比較A一致出力) PCINT22 (ピン変化割り込み22入力)	PD2	INT0 (外部割り込み0 入力) PCINT18 (ピン変化割り込み18入力)
PD5	T1 (タイマ/カウンタ1 外部クロック入力) OC0B (タイマ/カウンタ0 比較B一致出力) PCINT21 (ピン変化割り込み21入力)	PD1	TXD (USART 送信データ出力) PCINT17 (ピン変化割り込み17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) PCINT20 (ピン変化割り込み20入力)	PD0	RXD (USART 受信データ入力) PCINT16 (ピン変化割り込み16入力)

交換ピンの設定は次のとおりです。

#### • AIN1/PCINT23 – 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 ホ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ホートピンを設定してください。
- PCINT23 : ピン変化割り込み23入力。PD7ピンは外部割り込み元としても扱えます。

#### • AIN0/OC0A/PCINT22 – ホートD ビット6 : PD6

- AIN0 : アナログ比較器非反転入力。アナログ比較器機能を邪魔するデジタル ホ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ホートピンを設定してください。
- OC0A : タイマ/カウンタ0比較A一致出力。PD6ピンはタイマ/カウンタ0の比較A一致用外部出力として扱えます。この機能を扱うため、PD6ピンは出力として設定(DDD6=1)されなければなりません。このOC0Aピンはタイマ機能のPWM動作用出力ピンでもあります。
- PCINT22 : ピン変化割り込み22入力。PD6ピンは外部割り込み元としても扱えます。

#### • T1/OC0B/PCINT21 – 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。
- OC0B : タイマ/カウンタ0比較B一致出力。PD5ピンはタイマ/カウンタ0の比較B一致用外部出力として扱えます。この機能を扱うため、PD5ピンは出力として設定(DDD5=1)されなければなりません。このOC0Bピンはタイマ機能のPWM動作用出力ピンでもあります。
- PCINT21 : ピン変化割り込み21入力。PD5ピンは外部割り込み元としても扱えます。

#### • XCK/T0/PCINT20 – ホートD ビット4 : PD4

- XCK : USARTの外部クロック入出力ピンです。
- T0 : タイマ/カウンタ0の外部クロック入力ピンです。
- PCINT20 : ピン変化割り込み20入力。PD4ピンは外部割り込み元としても扱えます。

#### • INT1/OC2B/PCINT19 – ホート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。
- OC2B : タイマ/カウンタ2比較B一致出力。PD3ピンはタイマ/カウンタ2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、PD3ピンは出力として設定(DDD3=1)されなければなりません。このOC2Bピンはタイマ機能のPWM動作用出力ピンでもあります。
- PCINT19 : ピン変化割り込み19入力。PD3ピンは外部割り込み元としても扱えます。

#### • INT0/PCINT18 – ホートD ビット2 : PD2

- INT0 : 外部割り込み0入力。PD2ピンは外部割り込み元として扱えます。
- PCINT18 : ピン変化割り込み18入力。PD2ピンは外部割り込み元としても扱えます。

#### • TXD/PCINT17 – ホート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、このピンはホートD方向レジスタ(DDRD)のDDD1の値に拘らず出力として設定されます。
- PCINT17 : ピン変化割り込み17入力。PD1ピンは外部割り込み元としても扱えます。

• RXD/PCINT16 – ホートD ビット0 : PD0

- RXD : 受信データ(USART用データ入力ピン)。USART受信部が許可されると、このピンはDDRDのDDD0の値に拘らず入力として設定されます。USARTがこのピンを入力に強制するとき、プルアップは未だPORTD0ビットによって制御できます。
- PCINT16 : ピン変化割り込み16入力。PD0ピンは外部割り込み元としても扱えます。

下表はホートDの交換機能を図18-5.で示される交換信号に関連付けます。

表18-10. ホートD7~4の交換機能用交換信号

信号名	PD7/AIN1/PCINT23	PD6/AIN0/OC0A/PCINT22	PD5/T1/OC0B/PCINT21	PD4/XCK/T0/PCINT20
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	OC0A許可	OC0B許可	UMSEL
PVOV	0	OC0A	OC0B	XCK出力
PTOE	-	-	-	-
DIOE	PCINT23・PCIE2	PCINT22・PCIE2	PCINT21・PCIE2	PCINT20・PCIE2
DIOV	0	0	0	0
DI	PCINT23入力	PCINT22入力	T1/PCINT21入力	XCK入力/T0/PCINT20入力
AIO	AIN1入力	AIN0入力	-	-

表18-11. ホートD3~0の交換機能用交換信号

信号名	PD3/INT1/OC2B/PCINT19	PD2/INT0/PCINT18	PD1/TXD/PCINT17	PD0/RXD/PCINT16
PUOE	0	0	TXEN	RXEN
PUOV	0	0	0	PORTD0・PUD
DDOE	0	0	TXEN	RXEN
DDOV	0	0	1	0
PVOE	OC2B許可	0	TXEN	0
PVOV	OC2B	0	TXD	0
PTOE	-	-	-	-
DIOE	INT1許可+PCINT19・PCIE2	INT0許可+PCINT18・PCIE2	PCINT17・PCIE2	PCINT16・PCIE2
DIOV	1	1	1	1
DI	INT1/PCINT19入力	INT0/PCINT18入力	PCINT17入力	RXD/PCINT16入力
AIO	-	-	-	-

## 18.4. I/Oポート用レジスタ

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : MCUCR

変位 : \$55 (\$35)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$35です。

ビット	7	6	5	4	3	2	1	0
	–	–	–	PUD	–	–	IVSEL	IVCE
アクセス種別	R	R	R	R/W	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

このビットが1を書かれると、例えばDDxnとPORTxnレジスタがプルアップを許可(DDxn=0, PORTxn=1)に設定されていても、I/Oポートのプルアップは禁止されます。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PORTB

変位 : \$25 (\$05)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$05です。

ビット	7	6	5	4	3	2	1	0
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7～0 – PORTB7～0 : ポートB出力 (Port B Data)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : DDRB

変位 : \$24 (\$04)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$04です。

ビット	7	6	5	4	3	2	1	0
	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7～0 – DDB7～0 : ポートBデータ方向 (Port B Data Direction)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PINB

変位 : \$23 (\$03)

リセット : 不定

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$03です。

ビット	7	6	5	4	3	2	1	0
	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0
アクセス種別	R/W							
リセット値	不定							

#### ● ビット7～0 – PINB7～0 : ポートB入力 (Port B Input Pins)

PINxレジスタへの書き込みはI/Oに対する交互切り替え機能を提供します。「[ピンの出力交互切り替え](#)」をご覧ください。

#### 18.4.5. PORTC – ポートC出力レジスタ (Port C Data Register)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PORTC

変位 : \$28 (\$08)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$08です。

ビット	7	6	5	4	3	2	1	0
	–	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
アクセス種別	R	R/W						
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7～0 – PORTC6～0 : ポートC出力 (Port C Data)

#### 18.4.6. DDRC – ポートC方向レジスタ (Port C Data Direction Register)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : DDRC

変位 : \$27 (\$07)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$07です。

ビット	7	6	5	4	3	2	1	0
	–	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
アクセス種別	R	R/W						
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7～0 – DDC6～0 : ポートCデータ方向 (Port C Data Direction)

#### 18.4.7. PINC – ポートC入力レジスタ (Port C Input Pins Address)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PINC

変位 : \$26 (\$06)

リセット : 不定

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$06です。

ビット	7	6	5	4	3	2	1	0
	–	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
アクセス種別	R	R/W						
リセット値	0	不定						

#### ● ビット7～0 – PINC6～0 : ポートC入力 (Port C Input Pins)

PINxレジスタへの書き込みはI/Oに対する交互切り替え機能を提供します。「[ピンの出力交互切り替え](#)」をご覧ください。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PORTD

変位 : \$2B (\$0B)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$0Bです。

ビット	7	6	5	4	3	2	1	0
	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

● ビット7～0 – PORTD7～0 : ホートD出力 (Port D Data)

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : DDRD

変位 : \$2A (\$0A)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$0Aです。

ビット	7	6	5	4	3	2	1	0
	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
アクセス種別	R/W							
リセット値	0	0	0	0	0	0	0	0

● ビット7～0 – DDD7～0 : ホートDデータ方向 (Port D Data Direction)

#### 18.4.10. PIND – ホートD入力レジスタ (Port D Input Pins Address)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : PIND

変位 : \$29 (\$09)

リセット : 不定

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$09です。

ビット	7	6	5	4	3	2	1	0
	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0
アクセス種別	R/W							
リセット値	不定							

● ビット7～0 – PIND7～0 : ホートD入力 (Port D Input Pins)

PINxレジスタへの書き込みはI/Oに対する交互切り替え機能を提供します。「[ピンの出力交互切り替え](#)」をご覧ください。

## 19. TC0 – 8ビット タイマ/カウンタ0 (PWM付き)

### 19.1. 特徴

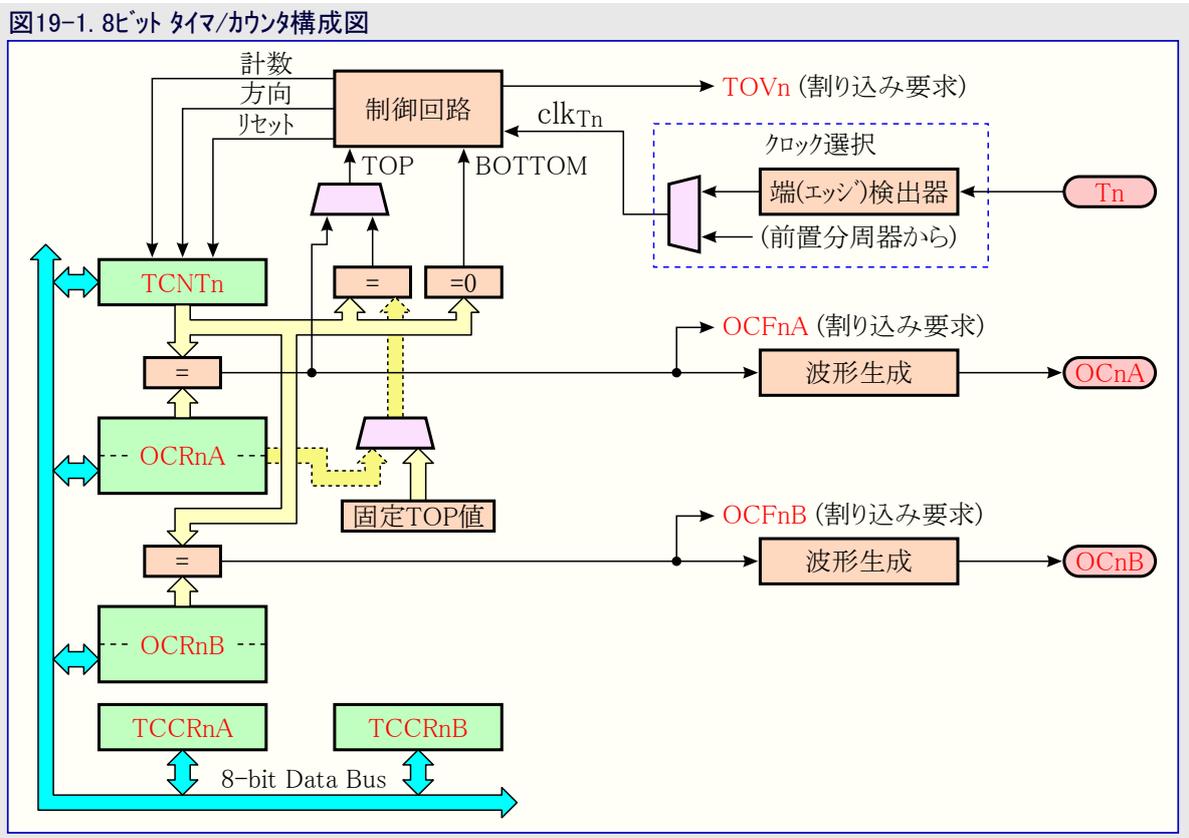
- 2つの独立した比較出力部
- 2重緩衝の比較レジスタ
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 可変PWM周期
- 周波数発生器
- 3つの独立した割り込み (TOV0, OCF0A, OCF0B)

### 19.2. 概要

タイマ/カウンタ0(TC0)は2つの独立した比較出力部とPWM支援付きの汎用8ビット タイマ/カウンタ部です。それは正確なプログラム実行タイミング (事象管理)、波形生成を許します。

この8ビット タイマ/カウンタの単純化した構成図は下で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注:原文は太字)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「8ビット タイマ/カウンタ0用レジスタ」で一覧されます。I/Oピンの実際の配置については「ピン配置」を参照してください。

TC0は電力削減レジスタ(PRR)のPRTIM0ビットが0に書かれる時に許可されます。



#### 19.2.1. 定義

本章でのレジスタとビット参照の多くは以下のように一般形で書かれます。

- $n=0$ はタイマ/カウンタ番号を表します。
- $x=A, B$ は比較出力部AまたはBを表します。

けれども、プログラムでレジスタまたはビット定義に使う時は正確な形式、即ち、タイマ/カウンタ0のカウンタ値アクセスに対してTCNT0が使われなければなりません。

右の定義は本章全体に渡って使われます。

表19-1. 用語定義

用語	意味
BOTTOM	タイマ/カウンタが\$00に到達した時。
MAX	タイマ/カウンタが\$FF(255)に到達した時。
TOP	タイマ/カウンタが指定された固定値(\$FF)またはOCR0A値に到達した時。この指定(TOP)値は動作種別に依存します。

## 19.2.2. 関係レジスタ

タイマ/カウンタ0(TCNT0)レジスタと比較レジスタ(OCR0x)は8ビットレジスタです。割り込み要求信号はタイマ/カウンタ0割り込み要求レジスタ(TIFR0)で全て見えます。全ての割り込みはタイマ/カウンタ0割り込み許可レジスタ(TIMSK0)で個別に遮蔽(禁止)されます。TIFR0とTIMSK0はこの図で示されません。

このTCは内部的、前置分周器経由、またはT0ピンの外部クロック元によってクロック駆動されます。クロック選択論理部は値を増加(または減少)するためにタイマ/カウンタによってどのクロック元と端(エッジ)が使われるかを制御します。クロック元が選ばれないと、TCは動きません。クロック選択論理部からの出力はタイマ/カウンタクロック(clkT0)として参照されます。

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

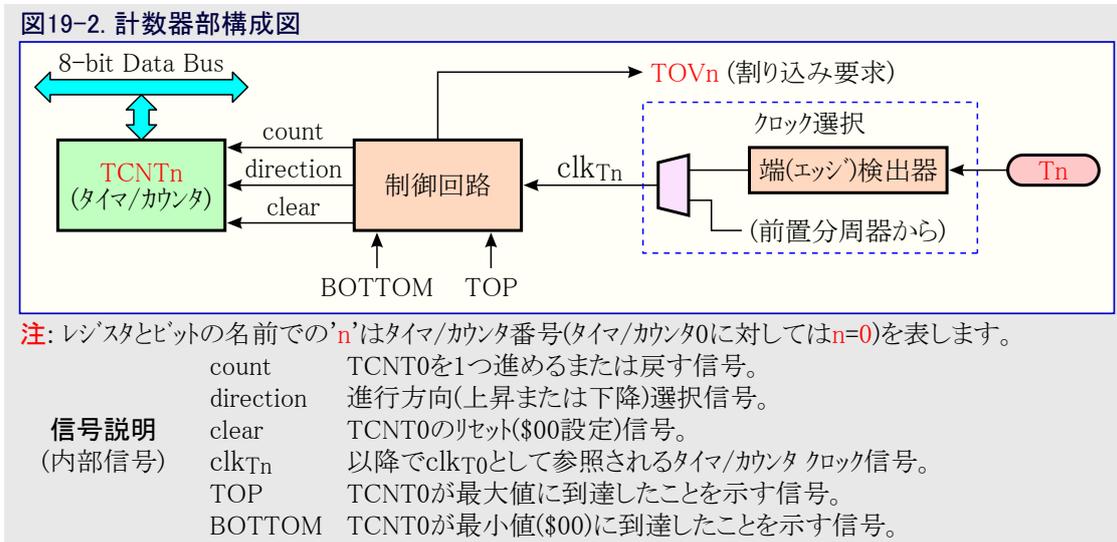
## 19.3. タイマ/カウンタのクロック

このTCは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR0B)のクロック選択(CS02~0)ビット書き込みによって選ばれます。

関連リンク 102頁の「タイマ/カウンタ0と1の前置分周器」

## 19.4. 計数器部

8ビットタイマ/カウンタの主な部分は設定可能な双方向計数器部です。下はこの計数器部とその周辺環境の構成図を示します。



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

計数順序(方法)はタイマ/カウンタ0制御レジスタA(TCCR0A)に配置された波形生成種別(WGM01,0)ビットとタイマ/カウンタ0制御レジスタB(TCCR0B)に配置された波形生成種別(WGM02)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法とOC0A/OC0B比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に関しては「動作種別」をご覧ください。

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

## 19.5. 比較出力部

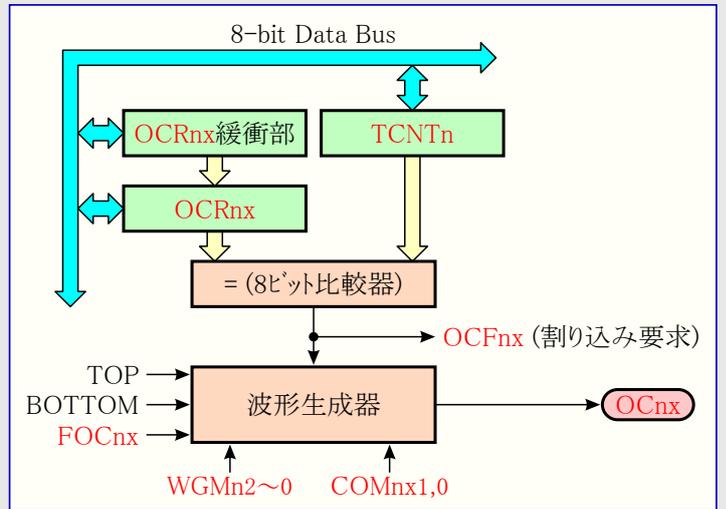
この8ビット比較器はTCNT0と比較レジスタ(OCR0AとOCR0B)を継続的に比較します。TCNT0がOCR0AまたはOCR0Bと等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF0AまたはOCF0B)を設定(1)します。対応する割り込みが許可(I=1, OCIE0AまたはOCIE0B=1)されているならば、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM2~0)ビットと比較出力選択(COM0x1,0)ビットによって設定された動作種別に従った出力を生成するのにこの一致信号を使います。MAXとBOTTOM信号は動作種別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。

OCR0xはパルス幅変調(PWM)のどれかを使う時に2重緩衝されます。2重緩衝動作が許可されると、CPUはOCR0x緩衝部をアクセスします。2重緩衝動作は計数の流れのTOPまたはBOTTOMのどちらかに対してOCR0xレジスタの更新を同期化します。この同期化は奇数長、非対称PWMパルスの発生を防ぎ、それによって不具合なしの出力を作成します。

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、OCR0xレジスタを直接アクセスします。

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

図19-3. 比較出力部構成図



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)、「x'は比較出力部(AまたはB)を表します。

### 19.5.1. 強制比較出力

非PWM波形生成動作での比較器の一致出力は、強制変更(TCCR0B.FOC0x)ビットに1を書くことによって強制(変更)できます。比較一致の強制は比較割り込み要求フラグ(OCF0x)の設定(1)やタイマ/カウンタの再設定/解除を行いませんが、OC0xピンは実際の比較一致が起きた場合と同様に更新されます(TCCR0A.COM0x1,0ビットがOC0xピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

### 19.5.2. TCNT0書き込みによる比較一致妨害

TCNT0への全てのCPU書き込みは、例えタイマ/カウンタが停止されていても、次のタイマ/カウンタクロック周期で起こるどんな比較一致をも妨げます。この特質はタイマ/カウンタクロックが許可されている時に割り込みを起動することなく、TCNT0と同じ値に初期化されることをOCR0xに許します。

### 19.5.3. 比較一致部の使用

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

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

比較出力選択(TCCR0A.COM0x1,0)ビットが比較値(OCR0x)と共に2重緩衝されないことに気付いてください。TCCR0A.COM0x1,0ビットの変更は直ちに有効となります。

## 19.6. 比較一致出力部

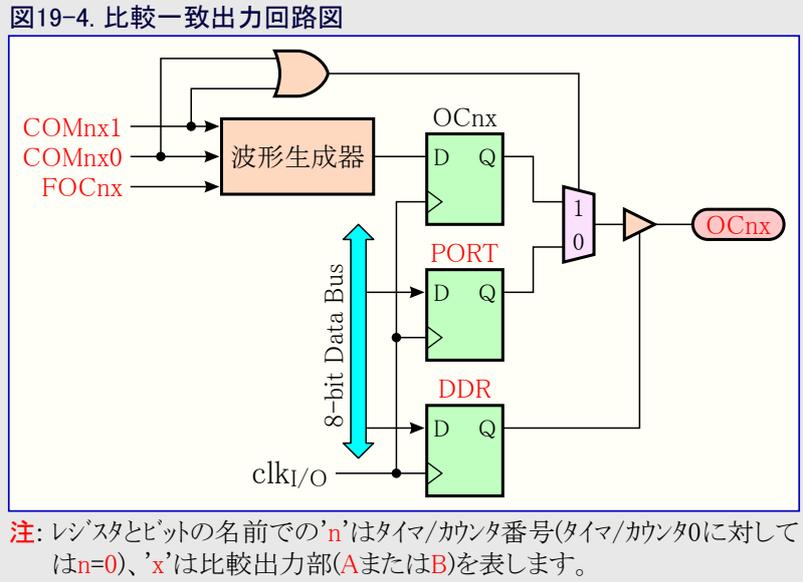
タイマ/カウンタ0制御レジスタAの比較出力選択(TCCR0A.COM0x1,0)ビットは以下の2つの機能を持ちます。

- 波形生成器は次の比較一致での比較出力(OC0x)状態の定義にCOM0xビットを使います。
- COM0xビットはOC0xピン出力元を制御します。

下図はCOM0xビットによって影響を及ぼされる論理回路の簡略化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM0x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の部分、即ちPORTとDDRだけが示されます。

システムリセットでOC0xレジスタは'0'にリセットされます。

注: 'OC0xの状態'はOC0xピンでなく、常に内部OC0xレジスタを参照します。



COM0x1,0ビットのどちらかが設定(1)されると、標準I/Oポート機能は波形生成器からの比較出力(OC0x)によって無効にされます。けれどもOC0xピンの方向(入出力)はポートピンに対するポート方向レジスタ(DDR)によって未だ制御されます。ポート方向レジスタでOC0xピンに対するビット(DDR\_OC0x)は、OC0x値がピンで見えるのに先立って出力として設定されなければなりません。このポートの交換機能は波形生成種別と無関係です。

比較出力ピン論理回路の設計は出力が許可される前のOC0x状態の初期化を許します。いくつかのTCCR0A.COM0x1,0ビット設定は或る種の動作種別に対して予約されています。

### 19.6.1. 比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作でTCCR0A.COM0x1,0ビットを違うふうに使います。全ての動作種別に対してTCCR0A.COM0x1,0=00設定は次の比較一致で実行すべきOC0xレジスタの動きがないことを波形生成器へ告げます。比較出力選択の記述も参照してください。

TCCR0A.COM0x1,0ビットの状態変更はこのビットが書かれた後の最初の比較一致で有効になります。非PWM動作について、この動作は強制変更(TCCR0B.FOC0x)スローフ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。

## 19.7. 動作種別

動作種別はタイマ/カウンタと比較出力ピンの動きを決めます。これはタイマ/カウンタ制御レジスタA(TCCR0A)とタイマ/カウンタ制御レジスタB(TCCR0B)の波形生成種別(TCCR0B.WGM02とTCCR0A.WGM01,0)ビットとTCCR0Aの比較出力選択(COM0x1,0)ビットの組み合わせによって定義されます。比較出力選択ビットは計数順序(動作)に影響を及ぼしませんが、一方波形生成種別ビットは影響を及ぼします。COM0x1,0ビットは生成されるPWM出力が反転されるべきか、されないべきか(反転または非反転PWM)どちらかを制御します。非PWM動作に対するCOM0x1,0ビットは比較一致で出力が解除(0)、設定(1)、1/0交互のどれにされるべきかを制御します(前の「比較一致出力部」項をご覧ください)。

タイミング情報の詳細については以降の「タイマ/カウンタ0のタイミング」項を参照してください。

関連リンク [73頁の「比較一致出力部」](#)  
[77頁の「タイマ/カウンタ0のタイミング」](#)

### 19.7.1. 標準動作

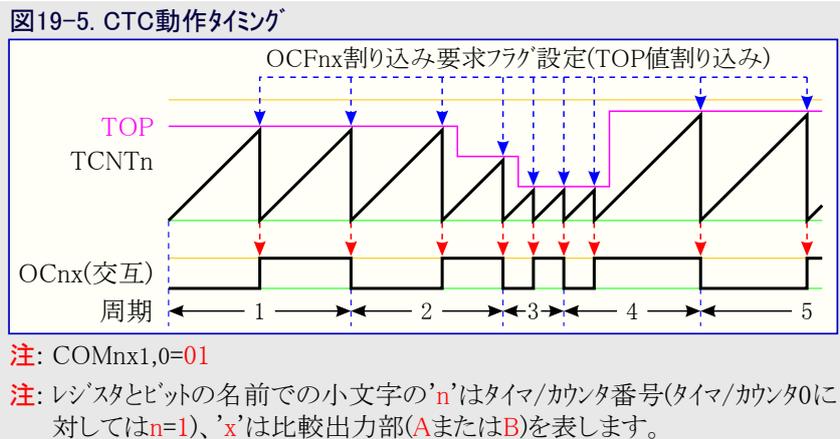
最も単純な動作種別が標準動作(WGM02~0=000)です。この動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV0)フラグはTCNT0が\$00になると同じタイマ/カウンタクロック周期で設定(1)されます。この場合、TOV0フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのように動きます。けれどもTOV0フラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解能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

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

### 19.7.2. 比較一致タイマ/カウンタ解除(CTC)動作

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

CTC動作についてのタイミング図は下で示されます。カウンタ(TCNT0)値はTCNT0とOCR0A間で比較一致が起こるまで増加し、そしてその後カウンタ(TCNT0)は解除(\$00)されます。



OCF0Aフラグの設定(1)により、タイマ/カウンタ値がTOP値に達する時毎に割り込みを生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。

注: 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間にBOTTOMと近い値にTOPを変更することは、CTC動作が2重緩衝を提供しないために注意して行わなければなりません。OCR0Aに書かれた新しい値がTCNT0の現在値よりも低い(小さい)場合、タイマ/カウンタはその回の比較一致を失います。その後、カウンタは比較一致が起こるのに先立って、最大値(\$FF)へ計数して\$00で循環を始めます。

CTC動作で波形出力を生成するため、OC0A出力はタイマ/カウンタ制御レジスタAの比較出力選択(TCCR0A.COM0A1,0)ビットを交互動作(=01)へ書くことによって各比較一致での論理レベル交互切り替えに設定できます。OC0A値はそのピンに対するデータ方向が出力に設定される場合にだけポートピンで見えます。生成された波形はOCR0Aが\$00に設定される時に $f_{OC0A} = f_{clk\_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

Nは前置分周数(1,8,64,256,1024)を表します。

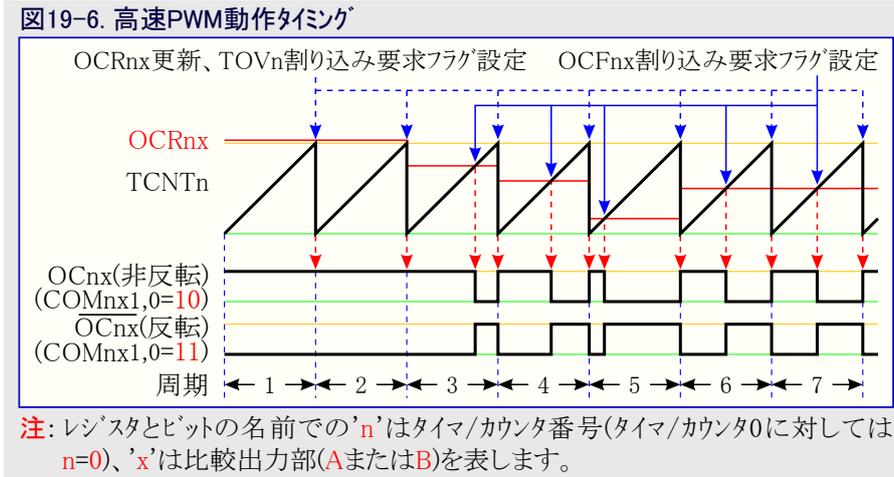
標準動作と同じように、タイマ/カウンタ溢れ(TOV0)フラグはカウンタがMAXから\$00へ計数すると同じタイマ/カウンタクロック周期で設定(1)されます。

### 19.7.3. 高速PWM動作

高速パルス幅変調(PWM)動作(WGM02~0=011または111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。TOPはWGM02~0=011時に\$FFとして定義されます。TOPはWGM02~0=111時にOCR0Aとして定義されます。

非反転比較出力動作(COM0x1,0=10)での比較出力(OC0x)はTCNT0とOCR0x間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM0x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。この単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がTOP値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は下で示されます。TCNT0値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。この図は非反転と反転のPWM出力を含みます。赤細線はOCR0x値を示し、TCNT0値との交点(接点)がTCNT0とOCR0x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF0x)はOCR0x=TOPを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。



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

高速PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。タイマ/カウンタ0制御レジスタのTCCR0A.COM0A1,0ビットの'10'書き込みは非反転PWM出力を生成、TCCR0A.COM0A1,0ビットの'11'書き込みは反転PWM出力を生成します。タイマ/カウンタ0制御レジスタB(TCCR0B)のWGM02ビットが設定(1)なら、TCCR0AのCOM0A1,0ビットの'01'書き込みは比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません。実際のOC0x値はポートピンに対するデータ方向が出力として設定される場合にだけ見えます。PWM波形はTCNT0とOCR0x間の比較一致で、OC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$00, TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC0xレジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

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

Nは前置分周数(1,8,64,256,1024)を表します。

OCR0xの両端値は高速PWM動作のPWM波形出力に対する特別な場合を表します。OCR0xがBOTTOM(\$00)と等しく書かれると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。OCR0x=TOP書き込みは(COM0x1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

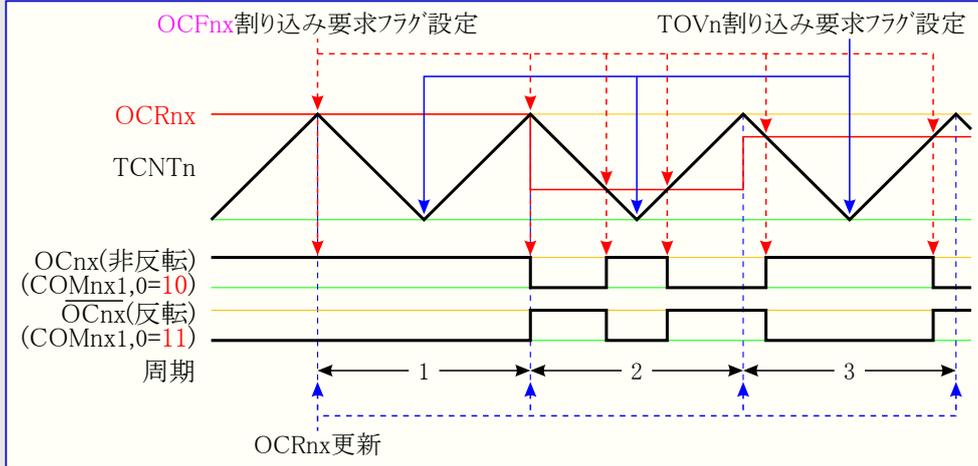
(訳補:WGM02~0=111の場合については、)デューティ比50%での周波数の波形出力は高速PWM動作で比較一致毎に論理反転するOC0A(COM0A1,0=01)選択によって達成できます。生成された波形はOCR0Aが\$00に設定される時に $f_{OC0x} = f_{clk\_I/O} / 2$ の最大周波数です。この特性は高速PWM動作で比較出力部の2重緩衝が許可されることを除き、CTC動作でのOC0A交互出力(COM0A1,0=01)と同じです。

## 19.7.4. 位相基準PWM動作

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

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

図19-7. 位相基準PWM動作タイミング



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)、『x'は比較出力部(AまたはB)を表します。

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

位相基準PWM動作での比較部はOC0xピンでのPWM波形の生成を許します。COM0x1,0ビットの'10'書き込みは非反転PWM出力を生成します。反転PWM出力はCOM0x1,0ビットの'11'書き込みによって生成できます。タイマ/カウンタ0制御レジスタB(TCCR0B)のWGM02ビットが設定(1)なら、タイマ/カウンタ0制御レジスタA(TCCR0A)のCOM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません。実際のOC0x値はそのポートピンに対するデータ方向が出力として設定される場合にだけ見えます。PWM波形はカウンタが増加する時のTCNT0とOCR0x間の比較一致でOC0x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT0とOCR0x間の比較一致でOC0xレジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は以下によって計算できます。

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

Nは前置分周数(1,8,64,256,1024)を表します。

OCR0xの両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合を表します。非反転PWM動作ではOCR0xレジスタにBOTTOM(\$00)が書かれると出力は定常的なLowです。OCR0xにTOPが書かれると定常的なHighです。反転PWMに対する出力は逆の論理値になります。

上のタイミング図の第2周期のその出発点に於いて、例えば比較一致がないとしても、OCnxにはHighからLowへの遷移があります。この遷移はBOTTOMを挟む対称の保証の要求を満たします。比較一致なしに遷移を生ずるのは2つの場合です。

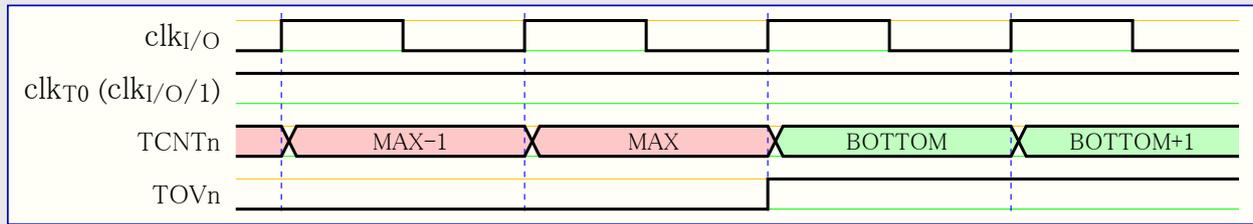
- タイミング図のようにOCR0xはTOPからその値を変更します。OCR0x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0x値よりも高い値から数え始め、そしてその理屈のために比較一致、従ってOCnxは上昇途中で起こされるであろう変更を起こしません。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)

## 19.8. タイマ/カウンタのタイミング

タイマ/カウンタは同期設計で、従って以降の図でタイマ/カウンタクロック( $clk_{T0}$ )がクロック許可信号として示されます。与えられた $TCn$ の実体が非同期動作を支援する場合、 $clk_{I/O}$ はTC発振部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グが設定(1)される時の情報を含みます。下の最初の図は位相基準PWM動作以外の全ての動作種別でのMAX値近辺の基本的なタイマ/カウンタ動作に関するタイミング データを図解します。

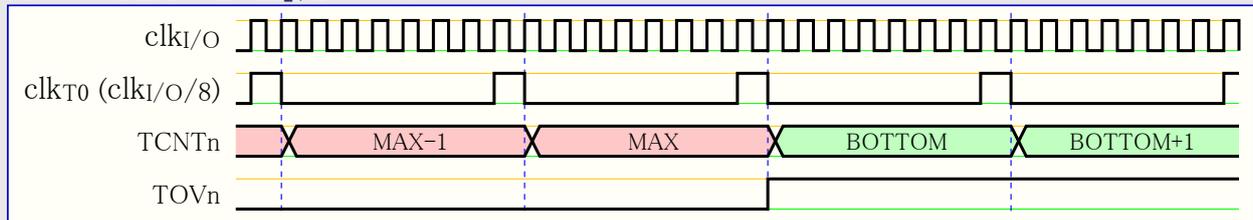
図19-8. 前置分周なし(1/1)のタイマ/カウンタ タイミング



注: レジスタとビットの名前での 'n' はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)を表します。

次図は同じタイミング データを示しますが、前置分周器が許可されています。

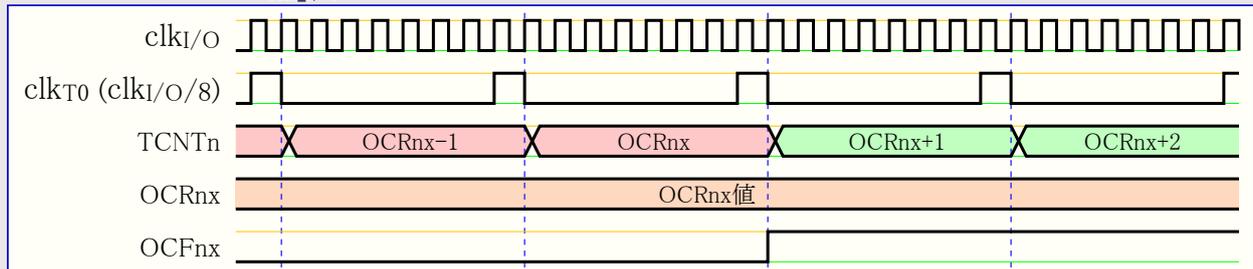
図19-9. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ タイミング



注: レジスタとビットの名前での 'n' はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)を表します。

次図はCTC動作とOCR0AがTOPのPWM動作を除く全動作種別でのOCF0Aと全動作種別でのOCF0Bの設定を示します。

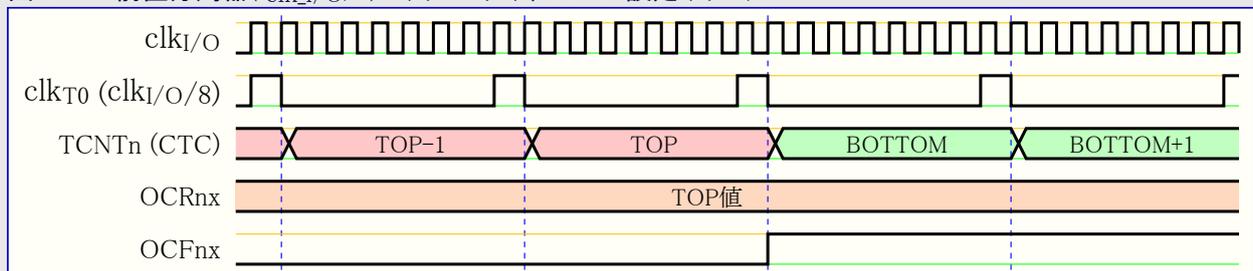
図19-10. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF0x設定 タイミング



注: レジスタとビットの名前での 'n' はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)、'x' は比較出力部(AまたはB)を表します。

次図はOCR0AがTOPの高速PWM動作と、CTC動作でのTCNT0の解除とOCF0Aの設定を示します。

図19-11. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF0A設定 タイミング



注: レジスタとビットの名前での 'n' はタイマ/カウンタ番号(タイマ/カウンタ0に対してはn=0)、'x' は比較出力部(AまたはB)を表します。

## 19.9. 8ビット タイマ/カウンタ0用レジスタ

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TCCR0A

変位 : \$44 (\$24)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$24です。

ビット	7	6	5	4	3	2	1	0
	COM0A1,0		COM0B1,0		–	–	WGM01,0	
アクセス種別	R/W	R/W	R/W	R/W	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ●ビット7,6 – COM0A1,0 : 比較A出力選択 (Compare Output Mode for Channel A)

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

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

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

表19-2. 非PWM動作比較A出力選択

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

下表はWGM02～0ビットが高速PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表19-3. 高速PWM動作比較A出力選択

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

注: COM0A1が設定(1)され、OCR0A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、設定(1)または解除(0)はBOTTOMで行われます。詳細については「高速PWM動作」を参照してください。

下表はWGM02～0ビットが位相基準PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表19-4. 位相基準PWM動作比較A出力選択

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

注: COM0A1が設定(1)され、OCR0A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、設定(1)または解除(0)はTOPで行われます。詳細については「位相基準PWM動作」を参照してください。

#### ●ビット5,4 – COM0B1,0 : 比較B出力選択 (Compare Output Mode for Channel B)

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

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

下表はWGM02～0ビットが**標準動作**または**CTC動作**(つまりPWM以外)に設定される時のCOM0B1,0ビット機能を示します。

表19-5. 非PWM動作比較B出力選択

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

下表はWGM02～0ビットが**高速PWM動作**に設定される時のCOM0B1,0ビットの機能を示します。

表19-6. 高速PWM動作比較B出力選択

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

注: COM0B1が設定(1)され、OCR0B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、設定(1)または解除(0)はBOTTOMで行われます。詳細については「**高速PWM動作**」を参照してください。

下表はWGM02～0ビットが**位相基準PWM動作**に設定される時のCOM0B1,0ビットの機能を示します。

表19-7. 位相基準PWM動作比較B出力選択

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

注: COM0B1が設定(1)され、OCR0B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、設定(1)または解除(0)はTOPで行われます。詳細については「**位相基準PWM動作**」を参照してください。

● **ビット1,0 - WGM01,0 : 波形生成種別 (Waveform Generation Mode)**

タイマ/カウンタ制御レジスタ**B(TCCR0B)**で得られる**WGM02ビット**と組み合わせたこれらのビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元、使われるべき波形生成のどの形式かを制御します。タイマ/カウンタ部によって支援される動作種別は標準動作(計数器)、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。「**動作種別**」をご覧ください。

表19-8. 波形生成種別選択

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

注: MAX=\$FF、BOTTOM=\$00です。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TCCR0B

変位 : \$45 (\$25)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$25です。

ビット	7	6	5	4	3	2	1	0
	FOC0A	FOC0B	–	–	WGM02	CS02~0		
アクセス種別	R/W	R/W	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

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

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

FOC0Aビットは常に0として読みます。

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

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

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

FOC0Bスロープは何れの割り込みの生成も行いません。

FOC0Bビットは常に0として読みます。

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

タイマ/カウンタ0制御レジスタA(TCCR0A)を参照してください。

### ● ビット2～0 – CS02～0 : クロック選択 (Clock Select)

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

表19-9. クロック選択ビット説明

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

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

### 19.9.3. TCNT0 – タイマ/カウンタ計数值レジスタ (TC0 Counter Value Register)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TCNT0

変位 : \$46 (\$26)

リセット : \$00

特質 : I/O特定命令でデータ空間としてI/Oレジスタをアクセスする時の変位アドレスは\$26です。

ビット	7	6	5	4	3	2	1	0
	TCNT07~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – TCNT07~0 : タイマ/カウンタ計数值 (TC0 Counter Value)

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

### 19.9.4. OCR0A – タイマ/カウンタ比較Aレジスタ (TC0 Output Compare Register 0 A)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : OCR0A

変位 : \$47 (\$27)

リセット : \$00

特質 : I/O特定命令でデータ空間としてI/Oレジスタをアクセスする時の変位アドレスは\$27です。

ビット	7	6	5	4	3	2	1	0
	OCR0A7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – OCR0A7~0 : タイマ/カウンタ比較A値 (Output Compare 0 A)

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

### 19.9.5. OCR0B – タイマ/カウンタ比較Bレジスタ (TC0 Output Compare Register 0 B)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : OCR0B

変位 : \$48 (\$28)

リセット : \$00

特質 : I/O特定命令でデータ空間としてI/Oレジスタをアクセスする時の変位アドレスは\$28です。

ビット	7	6	5	4	3	2	1	0
	OCR0B7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – OCR0B7~0 : タイマ/カウンタ比較B値 (Output Compare 0 B)

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

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

名称 : TIMSK0  
 変位 : \$6E  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

● **ビット1 – OCIE0A : タイマ/カウンタ0比較A割り込み許可 (Timer/Counter0 Output Compare A Match Interrupt Enable)**

OCIE0Aビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが許可されます。タイマ/カウンタ0で比較A一致が起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)で比較A割り込み要求フラグ(OCF0A)が設定(1)されると、対応する割り込みが実行されます。

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

TOIE0ビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ0溢れ割り込みが許可されます。タイマ/カウンタ0溢れが起こる、換言するとタイマ/カウンタ0割り込み要求フラグレジスタ(TIFR0)でタイマ/カウンタ0溢れ割り込み要求(TOV0)フラグが設定(1)されると、対応する割り込みが実行されます。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TIFR0  
 変位 : \$35 (\$15)  
 リセット : \$00  
 特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$15です。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCF0B	OCF0A	TOV0
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

● **ビット1 – OCF0A : タイマ/カウンタ0比較A割り込み要求フラグ (Timer/Counter0, Output Compare A Match Flag)**

OCF0Aビットは比較一致がタイマ/カウンタ(TCNT0)と比較レジスタ(OCR0A)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF0Aはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもOCF0Aは解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ0割り込み許可レジスタ(TIMSK0)のタイマ/カウンタ0比較A一致割り込み許可(OCIE0A)ビット、OCF0Aが設定(1)されると、タイマ/カウンタ0比較A一致割り込みが実行されます。

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

TOV0ビットはタイマ/カウンタ(TCNT0)溢れが起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、TOV0はハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもTOV0は解除(0)されます。ステータスレジスタ(SREG)の全割り込み許可(I)ビット、タイマ/カウンタ0割り込み許可レジスタ(TIMSK0)のタイマ/カウンタ0溢れ割り込み許可(TOIE0)ビット、TOV0が設定(1)されると、タイマ/カウンタ0溢れ割り込みが実行されます。位相基準PWM動作ではタイマ/カウンタ0が\$00で計数方向を変える時にこのビットが設定(1)されます。

これらフラグの設定はWGM02～0ビット設定に依存します。波形生成種別ビット記述の表19-8を参照してください。



## 20.4. 定義

本章でのレジスタとビット参照の多くは以下のように一般形で書かれます。

- $n$ はタイマ/カウンタ番号を表します。
- $x=A,B$ は比較出力部AまたはBを表します。

けれども、プログラムでレジスタまたはビット定義に使う時は正確な形式、即ち、タイマ/カウンタ1のカウンタ値アクセスに対してTCNT1が使われなければなりません。

以下の定義は本章全体に渡って広範囲に使われます。

表20-1. 用語定義

用語	意味
BOTTOM	カウンタが\$0000に到達した時。
MAX	カウンタが\$FFFF(65535)に到達した時。
TOP	カウンタがTOP値に到達した時(計数動作での最大値と等しくなった時)。TOP値は固定値(\$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。

## 20.5. 関係レジスタ

タイマ/カウンタ(TCNT1)、比較レジスタ(OCR1A,OCR1B)、捕獲レジスタ(ICR1)は全て16ビットレジスタです。16ビットレジスタをアクセスするとき、特別な手順に従わなければなりません。これらの手順は「16ビットレジスタのアクセス」項で記述されます。タイマ/カウンタ1制御レジスタ(TCCR1A,TCCR1B,TCCR1C)は8ビットレジスタで、CPUアクセスの制限はありません。割り込み要求信号はタイマ/カウンタ1割り込み要求レジスタ(TIFR1)で全て見えます。全ての割り込みはタイマ/カウンタ1割り込み許可レジスタ(TIMSK1)で個別に遮蔽(禁止)されます。TIFR1とTIMSK1は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(エッジ)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( $clk_{T1}$ )として参照されます。

2重緩衝化した比較レジスタ(OCR1A,OCR1B)はタイマ/カウンタ値と常に比較されます。この比較結果は比較出力(OC1A,OC1B)ピンでPWMまたは可変周波数出力を生成するための波形生成器によって使えます。「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のどちらかの外部(端で起動された)事象でタイマ/カウンタ値を捕獲(複写)できます。捕獲入力部は尖頭雑音を捕らえる機会を軽減するためにデジタル濾波器(雑音除去器)を含みます。

TOP値または最大タイマ/カウンタ値は、いくつかの動作種別で、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。

## 20.6. 16ビットレジスタのアクセス

TCNT1, OCR1A, OCR1B, ICRIは8ビットバス経由で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的に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TEMP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、現在TEMPに保存された上位バイトと書かれつつある下位バイトは同じクロック周期で両方が16ビットレジスタに複写されます。16ビットレジスタの下位バイトがCPUによって読まれると、16ビットレジスタの上位バイトは下位バイトが読まれるのと同じクロック周期でTEMPレジスタに複写され、その後に(上位バイトが)読まなければならない。

**注:** 16ビット書き込みを実行するには下位バイトに先立って上位バイトが書かれなければならない。16ビット読み込みについては上位バイトの前に下位バイトが読まれなければならない。

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ビットアクセス

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

**注:** 「コード例について」を参照してください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

### 非分断読み込み

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らない。

次のコード例はTCNT1レジスタ内容の非分断読み込み実行法を示します。OCR1A,OCR1B,ICR1レジスタは同じ原理を使うことによって読むことができます。

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

注: 「コード例について」を参照してください。

アセンブリ言語コード例はR17:R16レジスタ対にTCNT1値を戻します。

#### 非分断読み込み

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

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

注: 「コード例について」を参照してください。

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

関連リンク [14頁の「コード例について」](#)

### 20.6.1. 上位バイト一時レジスタの再使用

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

## 20.7. タイマ/カウンタのクロック

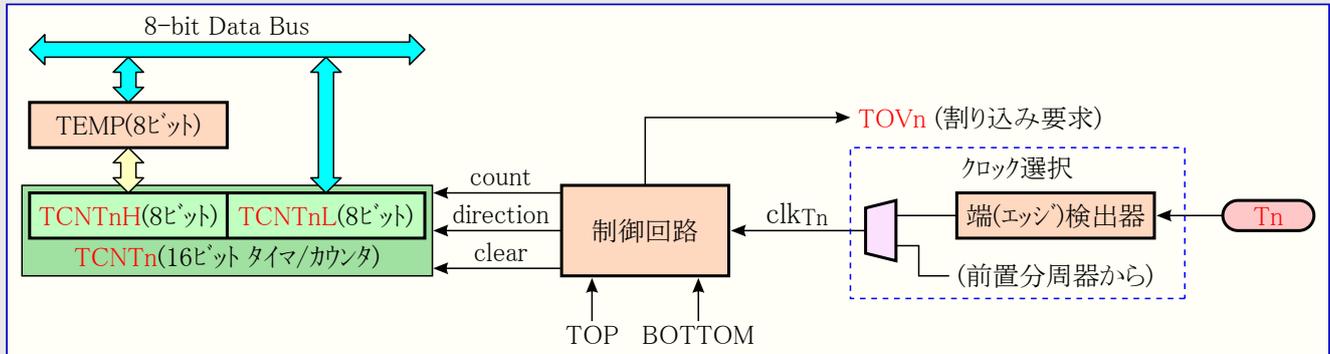
このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR1B)のクロック選択(CS12~0)ビットによって制御されるクロック選択論理回路で選ばれます。

関連リンク 102頁の「タイマ/カウンタ0と1の前置分周器」

## 20.8. 計数器部

16ビットタイマ/カウンタの主な部分は構成図で示されるように、設定可能な16ビット双方向カウンタ部です。

図20-2. 計数器部構成図



注: レジスタとビットの名前での 'n' は装置番号(タイマ/カウンタ1についてはn=1)を表します。

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

この16ビット カウンタはカウンタの上位8ビットを含むカウンタ上位(TCNT1H)と下位8ビットを含むカウンタ下位(TCNT1L)の2つの8ビット I/Oメモリ位置に配置されます。TCNT1HレジスタはCPUによる間接的なアクセスのみできます。CPUがTCNT1H I/O位置をアクセスするとき、CPUは上位バイト一時レジスタ(TEMP)をアクセスします。この一時レジスタはTCNT1Lが読まれる時にTCNT1H値で更新され、TCNT1Lが書かれる時にTCNT1Hは一時レジスタ値で更新されます。これは8ビット データ バス経由で1クロック周期内での16ビット カウンタ値全体の読み書きをCPUに許します。

注: カウンタが計数中の間にTCNT1レジスタを書く時が予測不能な結果を生じる特別な場合があります。これらの特別な場合はそれらが重要となる項で記述されます。

選んだ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クロックはタイマ/カウンタ制御レジスタB(TCCR1B)のクロック選択(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(即ち、上位優先権を持ちます)。

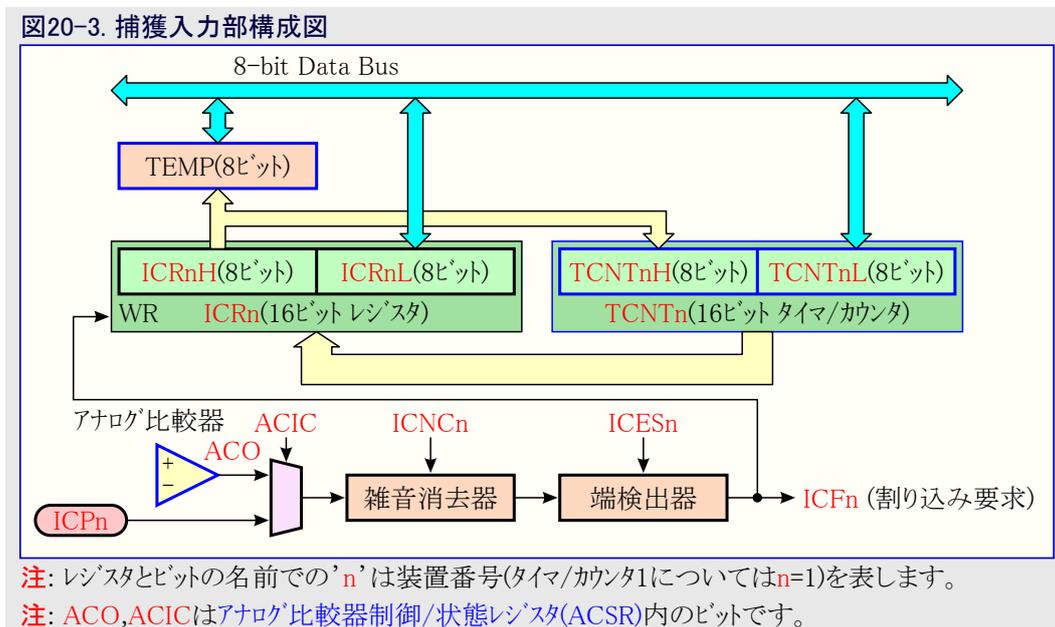
計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCR1A)とTCCR1Bの波形生成種別(TCCR1B.WGM13,2とTCCR1A.WGM11,0)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては「動作種別」をご覧ください。

TC1のタイマ/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。

## 20.9. 捕獲入力部

タイマ/カウンタ1は外部の出来事を捕獲でき、発生時間を示す時間印(タイマ/カウンタ値)を与える捕獲入力部と合体します。出来事または複数の出来事を示す外部信号はICP1ピンまたは代わりにアナログ比較器部経由で印加できます。時間印はその後、周波数、デューティ比、印加された信号の他の特性の計算に使えます。代わりに時間印は出来事の記録作成にも使えます。

捕獲入力部は下の構成図によって図解されます。直接的な捕獲入力部の部分でない構成図の要素は青枠(訳注:原文は灰色背景)で示されます。レジスタとビット名での小文字の'n'はタイマ/カウンタ番号を示します。



捕獲起動入力(ICP1)ピン若しくは代わりにアナログ比較器出力(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)検出器の設定を迫認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(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周期で設定(1)されます。許可(I=1, ICIE1=1)ならば捕獲割り込み要求フラグは捕獲割り込みを発生します。ICF1は割り込みが実行されると自動的に解除(0)されます。代わりにこのI/Oビット位置に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)、その後上位バイト(ICR1H)を読むことによって行われます。ICR1Lから下位バイトが読まれる時に上位バイトが上位バイト一時レジスタ(TEMP)に複写されます。CPUがICR1H I/O位置を読むと、この一時レジスタにアクセスします。

ICR1はカウンタのTOP値定義にICR1を利用する波形生成種別を使う時にだけ書けます。これらの場合、TOP値がICR1に書かれ得る前に波形生成種別(WGM13~0)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ビットレジスタのアクセス」もご覧ください。

### 20.9.1. 捕獲起動元

捕獲入力部用の主な起動元は捕獲起動入力(ICP1)ピンです。タイマ/カウンタ1は捕獲入力部用起動元としてアナログ比較器出力を代わりに使えます。アナログ比較器はアナログ比較器制御/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、T1ピンについて同じ技法を使って採取されます。端検出器も全く同じです。けれども雑音除去が許可されると、付加論理回路が端検出器の前に挿入され、そして遅延を4システムクロック周期増やします。タイマ/カウンタがTOP値定義にICR1を使う波形生成種別に設定されない限り、雑音除去器と端検出器の入力が常に許可されます。

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。

関連リンク 102頁の「タイマ/カウンタ0と1の前置分周器」

### 20.9.2. 雑音除去器

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

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

### 20.9.3. 捕獲入力の使用

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

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

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

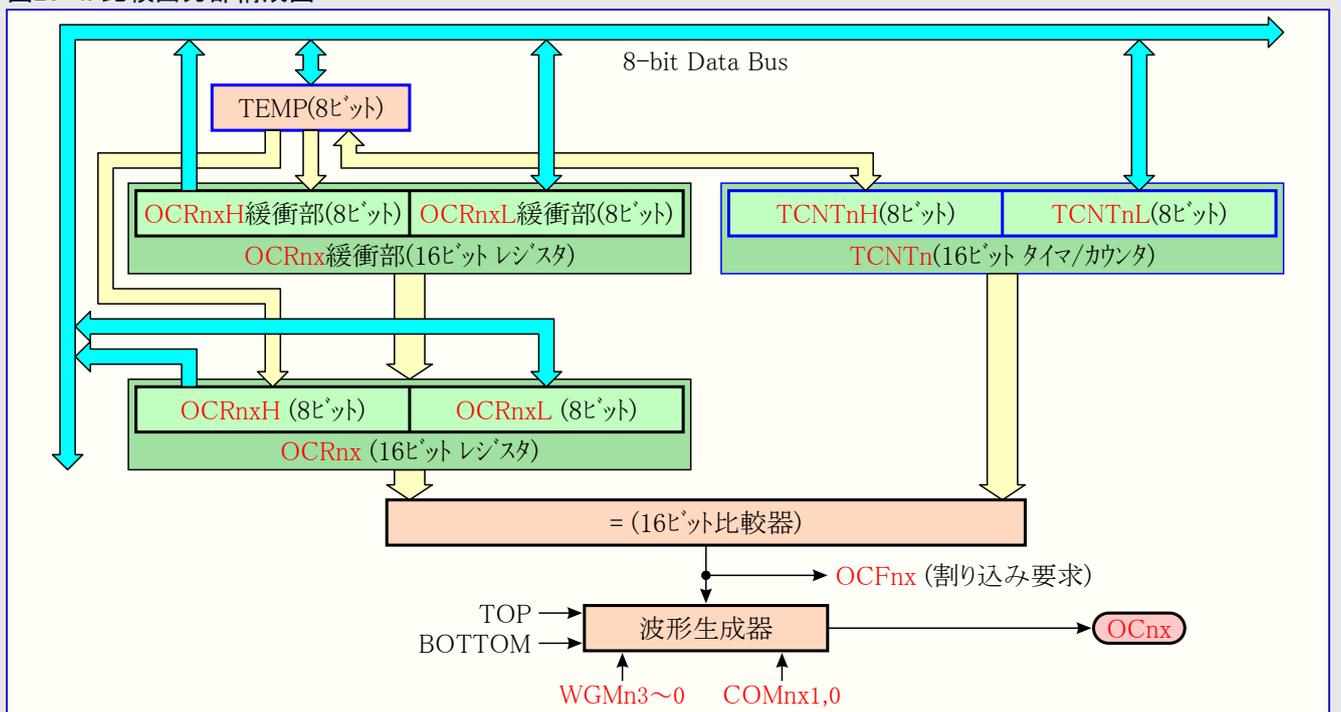
外部信号のデューティ比測定は各捕獲後に起動端が変更されることを必要とします。検出端の変更はICR1が読まれてしまった後に可能な限り早く行われなければなりません。端(エッジ)の変更後、**捕獲割り込み要求フラグ(ICF1)**はソフトウェア(I/Oビット位置への論理1書き込み)によって解除(0)されなければなりません(訳補:端変更によってICF1が設定(1)されることを想定)。周波数のみの測定について(割り込み処理が使われる場合)、ICF1の解除(0)は必要とされません。

### 20.10. 比較出力部

この16ビット比較器は**TCNT1**と**比較レジスタ(OCR1x)**を継続的に比較します。TCNT1とOCR1xが等しければ、比較器は一致を指示します。この一致は**次のタイマ/カウンタクロック周期**で**比較割り込み要求フラグ(TIFR1.OCF1x)**を設定(1)します。許可(I=1, TIMSK1.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の信号は動作種別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。「**動作種別**」をご覧ください。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。

下は比較出力部の構成図を示します。構成図で比較出力部の直接的部分でない要素は青枠(訳注:原文灰色背景)で示されます。

図20-4. 比較出力部構成図



注: レジスタとビットの名前での'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)、「x」は比較出力部(AまたはB)を表します。

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。

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ます。2重緩衝動作が禁止されると直接OCR1xレジスタをアクセスします。

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

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

(訳注) ここでは比較1xレジスタ全体をOCR1x、OCR1xを構成する緩衝部部分をOCR1x緩衝部、実際の比較に使用されるレジスタ本体部分をOCR1x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。

### 20.10.1. 強制比較出力

非PWM波形生成動作での比較器の一致出力は強制変更(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)。

### 20.10.2. TCNT1書き込みによる比較一致妨害

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

### 20.10.3. 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。

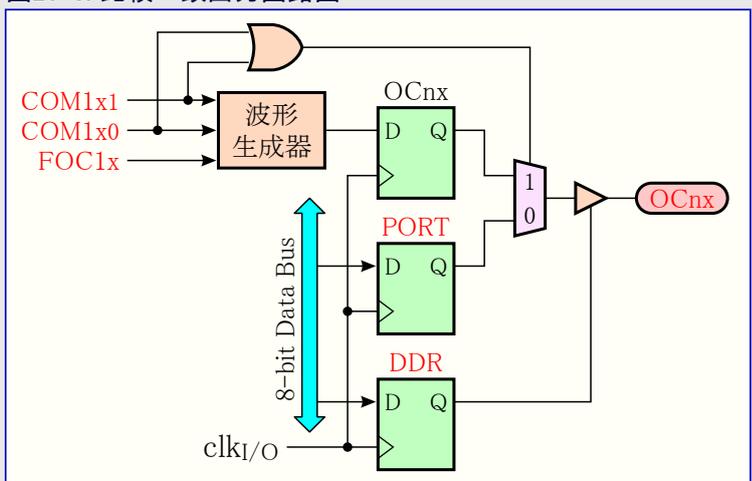
OC1xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)スローブビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

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。

## 20.11. 比較一致出力部

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

図20-5. 比較一致出力回路図



注: レジスタとビットの名前での'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)、『x』は比較出力部(AまたはB)を表します。

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。

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。

COM1x1,0ビットは捕獲入力部での何の効果もありません。

### 20.11.1. 比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作で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。比較出力選択の記述も参照してください。

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

## 20.12. 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(TCCR1A.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。TCCR1A.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)のどちらかを制御します。非PWM動作に対するTCCR1A.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。

関連リンク 90頁の「比較一致出力部」  
95頁の「タイマ/カウンタのタイミング」

### 20.12.1. 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BOTTOM=\$0000から再び始めます。通常動作でのタイマ/カウンタ溢れ(TIFR1.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、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。

### 20.12.2. 比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(動作形態4, 12, 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、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が右で示されます。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、その後にTCNT1は解除(\$0000)されます。

実際のCTC動作形態に依存して、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。

注: 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間にBOTTOMと近い値にTOPを変更するのは、CTC動作が2重緩衝を提供し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、カウンタは比較一致が起こるのに先立って、最大値(\$FFFF)へ計数して\$0000から循環を始めます。

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)を使うことでしよう。

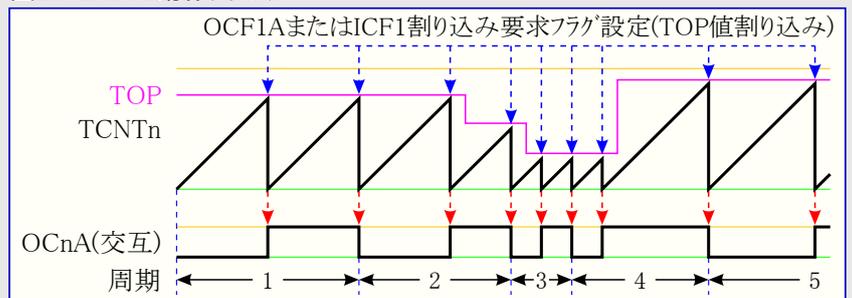
CTC動作で波形出力を生成するため、OC1A出力は比較出力選択(COM1A1,0)ビットを交互動作(=01)に設定することによって各比較一致での論理レベル交互切替に設定できます。OC1A値はそのピンに対するデータ方向が出力(DDR\_OC1A=1)に設定されない限り、ポートピンで見えないでしょう。生成された波形はOCR1Aが0(\$0000)に設定される時に $f_{OC1A} = f_{clk\_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

注: レジスタとビットの名前での小文字の'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)を表します。  
• 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1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。

図20-6. CTC動作タイミング



注: COM1A1,0=01

注: レジスタとビットの名前での小文字の'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)を表します。

### 20.12.3. 高速PWM動作

高速パルス幅変調(PWM)動作(動作形態5,6,7,14,15、WGM13~0=0101,0110,0111,1110,1111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。非反転比較出力動作(COM1x1,0=10)での比較出力(OC1x)はTCNT1とOCR1x間の比較一致で解除(0)され、BOTTOMで設定(1)されます。反転出力動作(COM1x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準や位相/周波数基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

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

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

高速PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0101,0110,0111)、ICR1値(WGM13~0=1110)またはOCR1A値(WGM13~0=1111)のどれかと一致するまで増加されます。そしてカウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$0000)されます。TOPを定義するのにOCR1AかICR1を使う高速PWM動作のタイミング図は下で示されます。TCNT1値はタイミング図で単一傾斜動作(鋸波)を表す折れ線グラフとして示されます。本図は非反転と反転のPWM出力を含みます。細い赤線はOCR1x値を示し、TCNT1値との交点(接点)がTCNT1とOCR1x間の比較一致を示します(訳注:図補正に伴い本行若干変更)。比較割り込み要求フラグ(OCF1x)は比較一致が起こると設定(1)されます。

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

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

ICR1がTOP値を定義するのに使われるとき、ICR1を更新する手順はOCR1Aの更新と異なります。ICR1は2重緩衝されません。これは前置分周なしまたは低い前置分周値でカウンタが走行している時にICR1が小さな値に変更される場合、書かれた新しいICR1値がTCNT1の現在値よりも小さくなる危険を意味します。結果として、カウンタは(その回の)TOP値での比較一致を失います。その後のカウンタは比較一致を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位置が書かれる時に書かれた値は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なります。ICR1を使うことにより、OC1AでのPWM出力を生成するためにOCR1Aが自由に使えます。けれども基準PWM周波数が(TOP値を変更することによって)動的に変更される場合、OCR1Aが2重緩衝機能のため、TOPとしてOCR1Aを使うことは明らかに良い選択です。

高速PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットの'10'書き込みは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0の'11'書き込みによって生成できます。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。

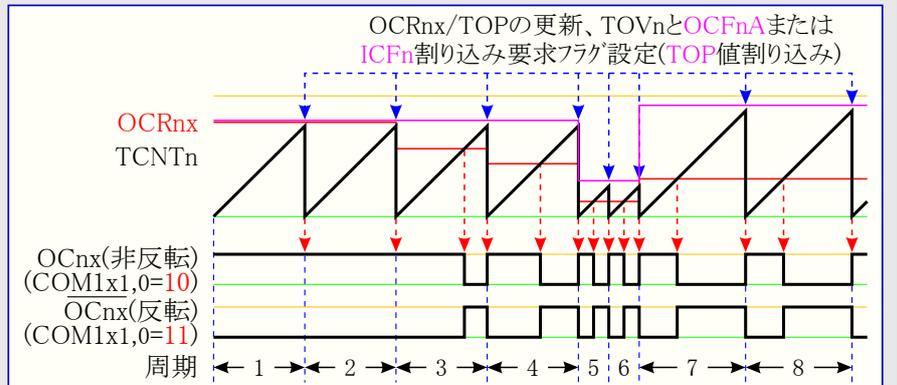
PWM出力周波数は次式によって計算できます。

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

注: レジスタとビットの名前での'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)、'x'は比較出力部(AまたはB)を表します。  
• 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当たります。OCR1xがBOTTOM(\$0000)に等しく設定されると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭いスパイク(パルス)になるでしょう。TOPに等しいOCR1x設定は(COM1x1,0ビットによって制御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。

図20-7. 高速PWM動作タイミング



注: レジスタとビットの名前での'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)、'x'は比較出力部(AまたはB)を表します。

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

#### 20.12.4. 位相基準PWM動作

位相基準パルス幅変調(PWM)動作(動作形態1,2,3,10,11、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TOPからBOTTOMへを繰り返して計数します。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。

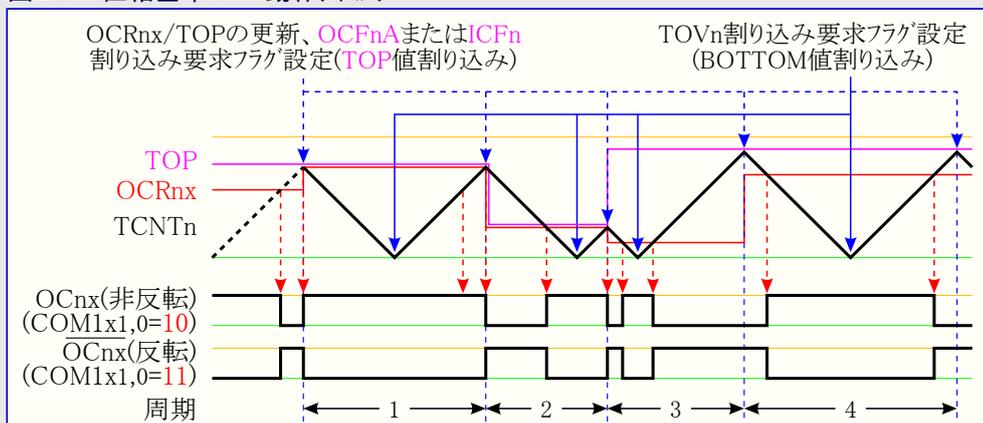
位相基準PWM動作のPWM分解能は8,9,10ビット固定、またはOCR1AかICR1のどちらかによって定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は右式を使うことによって計算できます。

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

位相基準PWM動作でのカウンタはカウンタ値が固定値\$00FF,\$01FF,\$03FF(WGM13~0=0001,0010,0011)、ICR1値(WGM13~0=1010)またはOCR1A値(WGM13~0=1011)のどれかと一致するまで増加されます。カウンタはTOPに到達したその時に計数方向を変更します。このTCNT1値は1タイマ/カウンタクロック周期TOPと等しくなります。TOPを定義するのにOCR1AかICR1を使う位相基準PWM動作のタイミング図は下で示されます。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)フラグはカウンタがBOTTOMに到達する毎に設定(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時に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。

図20-8. 位相基準PWM動作タイミング



注: レジスタとビットの名前での'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)、『x'は比較出力部(AまたはB)を表します。

TOP値を変更するとき、プログラムは新TOP値が全ての比較レジスタ値と等しいか大きいことを保証しなければなりません。TOP値が何れかの比較レジスタよりも小さな場合、TCNT1とそのOCR1x間で比較一致は決して起きません。固定TOP値を使う場合、どのOCR1xが書かれる時も、未使用ビットが0で隠(に置換)されることに注意してください。タイミング図の第3周期によって図示されるように、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)が違うとき、その周期の2つの傾斜長は異なるでしょう。この長さの相違が出力での非対称な結果を生じます。

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

位相基準PWM動作での比較部はOC1xピンでのPWM波形の生成を許します。COM1x1,0ビットの'10'書き込みは非反転PWM出力を作成し、反転PWM出力はCOM1x1,0の'11'書き込みで生成できます。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

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

注: レジスタとビットの名前での'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)、『x'は比較出力部(AまたはB)を表します。

• Nは前置分周数(1,8,64,256,1024)を表します

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

## 20.12.5. 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(動作形態8,9、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(訳補:TOPとBOTTOM)です(図20-8と下のタイミング図をご覧ください)。

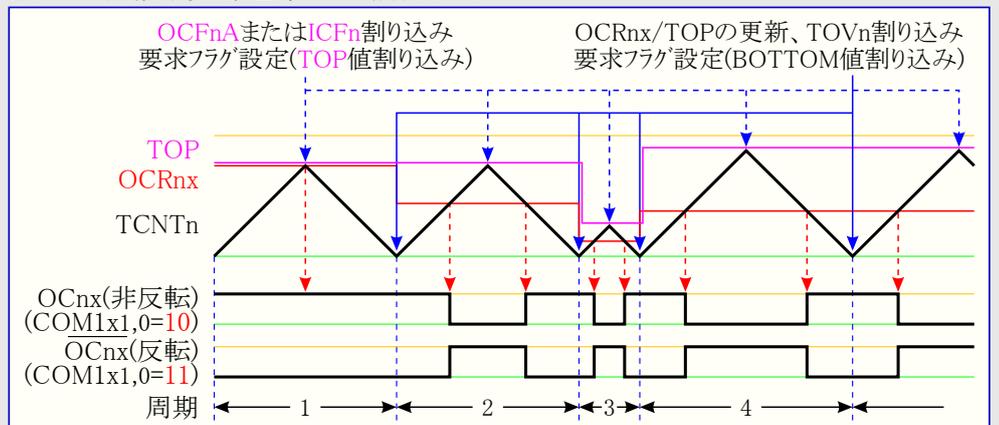
位相/周波数基準PWM動作のPWM分解能はOCR1AかICR1のどちらかで定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は右式を使うことによって計算できます。

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

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、TOPと等価です。位相/周波数基準PWM動作のタイミング図は下で示されます。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レジスタが(BOTTOMに於いて)2重緩衝値で更新されるのと同じタイマ/カウンタクロック周期で設定(1)されます。OCR1AかICR1のどちらかがTOP値を定義するのに使われると、OCF1AまたはICF1A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値またはBOTTOM値に到達する毎に割り込みを発生するのに使えます。

図20-9. 位相/周波数基準PWM動作タイミング\*



注: レジスタとビットの名前での'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)、'x'は比較出力部(AまたはB)を表します。

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。

上のタイミング図で示されるように、生成される出力は位相基準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(TCCRIA.COM1x1,0の記述をご覧ください)。実際のOC1x値はそのポートピンに対するデータ方向が出力(DDR\_OC1x=1)として設定される場合にだけ見えるでしょう。PWM波形はカウンタが増加する時のTCNT1とOCR1x間の比較一致でOC1x(内部)レジスタを設定(1)(または解除(0))と、カウンタが減少する時のTCNT1とOCR1x間の比較一致でOC1xレジスタを解除(0)(または設定(1))することによって生成されます。位相/周波数基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

$$f_{OCnxPFCPWM} = \frac{f_{clk\_1/O}}{2 \times N \times \text{TOP}}$$

注: レジスタとビットの名前での'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)、'x'は比較出力部(AまたはB)を表します。

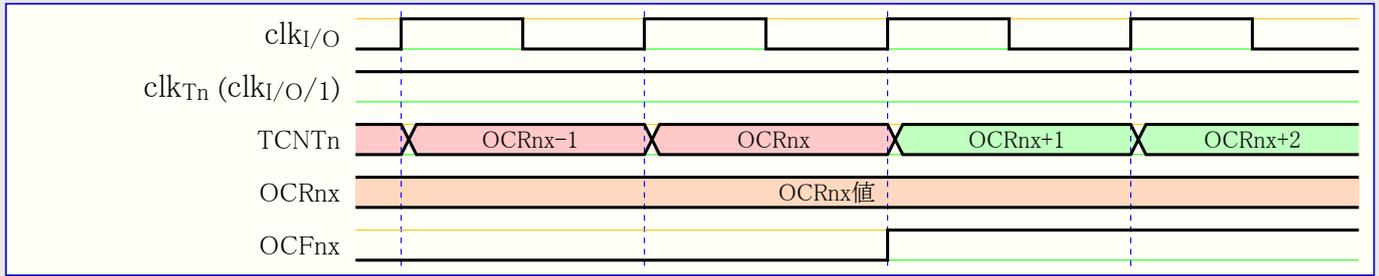
• Nは前置分周数(1,8,64,256,1024)を表します

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

## 20.13. タイマ/カウンタのタイミング

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

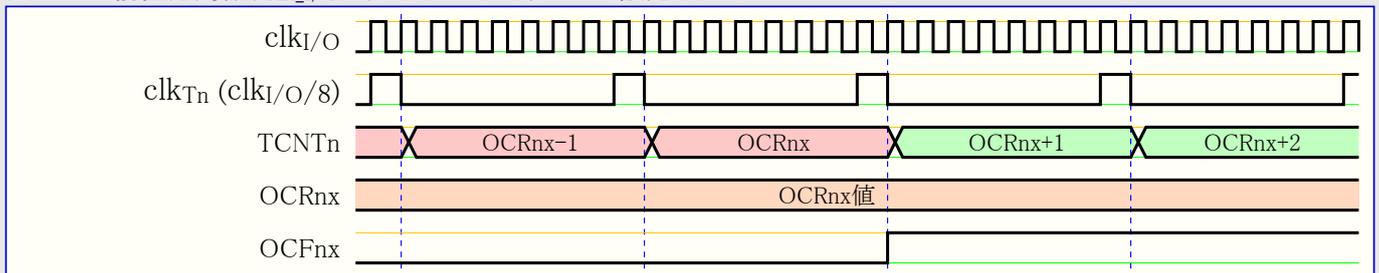
図20-10. 前置分周なし(1/1)のタイマ/カウンタ、 $OCF1x$ 設定 タイミング



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対しては $n=1$ )、'x'は比較出力部(AまたはB)を表します。

次図は同じタイミング データを示しますが、前置分周器が許可されています。

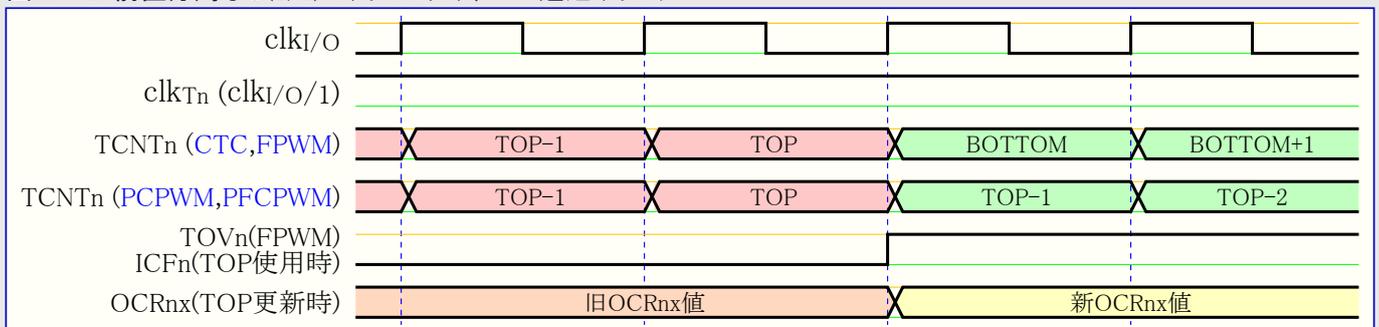
図20-11. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、 $OCF1x$ 設定 タイミング



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対しては $n=1$ )、'x'は比較出力部(AまたはB)を表します。

次図は各動作種別での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。

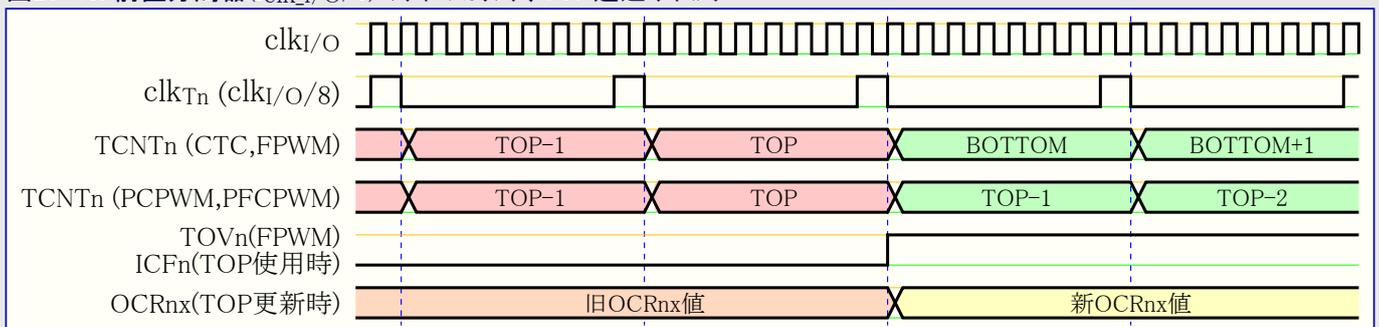
図20-12. 前置分周なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対しては $n=1$ )、'x'は比較出力部(AまたはB)を表します。

次図は同じタイミング データを示しますが、前置分周器が許可されています。

図20-13. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、TOP近辺 タイミング



注: レジスタとビットの名前での'n'はタイマ/カウンタ番号(タイマ/カウンタ1に対しては $n=1$ )、'x'は比較出力部(AまたはB)を表します。

## 20.14. 16ビット タイマ/カウンタ1用レジスタ

### 20.14.1. TCCR1A – タイマ/カウンタ1制御レジスタA (Timer/Counter 1 Control Register A)

名称 : TCCR1A  
 変位 : \$80  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	COM1A1,0		COM1B1,0		-	-	WGM11,0	
アクセス種別	R/W	R/W	R/W	R/W	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット7,6 – COM1A1,0 : 比較A出力選択 (Compare Output Mode for Channel A)
- ビット5,4 – COM1B1,0 : 比較B出力選択 (Compare Output Mode for Channel B)

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時、COM1x1,0ビットの機能はWGM13~0ビット設定に依存します。下表は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20-2. 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

下表は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20-3. 高速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=111X : 比較一致でOC1Aピントグル(交互)出力、OC1Bは標準ポート動作(OC1B切断) WGM13~0上記以外 : 標準ポート動作 (OC1x切断)
1	0	比較一致でLow、BOTTOMでHighをOC1xピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC1xピンへ出力 (反転動作)

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

下表は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20-4. 位相基準または位相/周波数基準PWM動作での比較出力選択 (注: xはAまたはB、Xは0または1)

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

注: COM1x1が設定(1)され、OCR1xがTOPと等しい時に特別な状態が起きます。詳細については「位相基準PWM動作」を参照してください。

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

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。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。「動作種別」をご覧ください。

表20-5. 波形生成種別選択

番号	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	タイマ/カウンタ動作種別	TOP値	OCR1x 更新時	TOV1 設定時
0	0	0	0	0	標準動作	\$FFFF	即値	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	即値	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	即値	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。

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

名称 : TCCR1B  
 変位 : \$81  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	ICNC1	ICES1	-	WGM13,2		CS12~0		
アクセス種別	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

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

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

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

ICR1がTOP値として使われると(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4,3 – WGM13,2 : 波形生成種別 (Waveform Generation Mode)

タイマ/カウンタ制御レジスタA(TCCR1A)を参照してください。

● ビット2~0 - CS12~0 : クロック選択 (Clock Select)

この3つのクロック選択ビットはタイマ/カウンタ(TCNT1)によって使われるべきクロック元を選びます。図20-10と図20-11を参照してください。

表20-6. タイマ/カウンタn入力クロック選択

CS12	CS11	CS10	意味
0	0	0	停止 (タイマ/カウンタ1動作停止)
0	0	1	clkI/O (前置分周なし)
0	1	0	clkI/O/8 (8分周)
0	1	1	clkI/O/64 (64分周)
1	0	0	clkI/O/256 (256分周)
1	0	1	clkI/O/1024 (1024分周)
1	1	0	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)

20.14.3. TCCR1C - タイマ/カウンタ1制御レジスタC (Timer/Counter 1 Control Register C)

名称 : TCCR1C

変位 : \$82

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	FOC1A	FOC1B	-	-	-	-	-	-
アクセス種別	R/W	R/W	R	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

● ビット7 - FOC1A : OC1A強制変更 (Force Output Compare for Channel A)

● ビット6 - FOC1B : OC1B強制変更 (Force Output Compare for Channel B)

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。FOC1A/FOC1Bビットに論理1を書くとき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。

20.14.4. TCNT1L - タイマ/カウンタ1計数值下位バイト (Timer/Counter 1 Counter Value Low byte)

名称 : TCNT1L

変位 : \$84

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	TCNT17~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 - TCNT17~0 : タイマ/カウンタ1計数值下位バイト (Timer/Counter Counter Value Low byte)

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

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。

## 20.14.5. TCNT1H – タイマ/カウンタ1計数值上位バイト (Timer/Counter 1 Counter Value High byte)

名称 : TCNT1H  
 変位 : \$85  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TCNT115~8							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – TCNT115~8 : タイマ/カウンタ1計数值上位バイト (Timer/Counter Counter Value High byte)

タイマ/カウンタ1下位バイト(TCNT1L)を参照してください。

## 20.14.6. OCR1AL – タイマ/カウンタ1比較Aレジスタ下位バイト (Output Compare Register 1 A Low byte)

名称 : OCR1AL  
 変位 : \$88  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR1A7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – OCR1A7~0 : タイマ/カウンタ1比較A下位バイト (Output Compare A Low byte)

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。

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

## 20.14.7. OCR1AH – タイマ/カウンタ1比較Aレジスタ上位バイト (Output Compare Register 1 A High byte)

名称 : OCR1AH  
 変位 : \$89  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR1A15~8							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – OCR1A15~8 : タイマ/カウンタ1比較A上位バイト (Output Compare A High byte)

上のタイマ/カウンタ1比較Aレジスタ下位バイト(OCR1AL)を参照してください。

## 20.14.8. OCR1BL – タイマ/カウンタ1比較Bレジスタ下位バイト (Output Compare Register 1 B Low byte)

名称 : OCR1BL  
 変位 : \$8A  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR1B7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – OCR1B7~0 : タイマ/カウンタ1比較B下位バイト (Output Compare B Low byte)

上のタイマ/カウンタ1比較Aレジスタ下位バイト(OCR1AL)を参照してください。

### 20.14.9. OCR1BH – タイマ/カウンタ1比較レジスタ上位バイト (Output Compare Register 1 B High byte)

名称 : OCR1BH  
 変位 : \$8B  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR1B15~8							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – OCR1B15~8 : タイマ/カウンタ1比較B上位バイト (Output Compare B High byte)

タイマ/カウンタ1比較Aレジスタ下位バイト(OCR1AL)を参照してください。

### 20.14.10. ICR1L – タイマ/カウンタ1捕獲レジスタ下位バイト (Input Capture Register 1 Low byte)

名称 : ICR1L  
 変位 : \$86  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	ICR17~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – ICR17~0 : タイマ/カウンタ1捕獲下位バイト (Input Capture Low byte)

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

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

### 20.14.11. ICR1H – タイマ/カウンタ1捕獲レジスタ上位バイト (Input Capture Register 1 High byte)

名称 : ICR1H  
 変位 : \$87  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	ICR115~8							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – ICR115~8 : タイマ/カウンタ1捕獲A上位バイト (Input Capture High byte)

上のタイマ/カウンタ1捕獲レジスタ下位バイト(ICR1L)を参照してください。

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

名称 : TIMSK1  
 変位 : \$6F  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1
アクセス種別	R	R	R/W	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

このビットが1が書かれて、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1割り込み要求フラグレジスタ(TIFR1)に配置された捕獲割り込み要求フラグ(ICF1)が設定(1)されると、対応する割り込みベクタが実行されます。

● **ビット2 – OCIE1B : タイマ/カウンタ1比較B割り込み許可** (Timer/Counter 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1割り込み要求フラグ レジスタ(TIFR1)に配置された比較B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が実行されます。

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

このビットが**1**を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(**1**)されると、タイマ/カウンタ1比較A一致割り込みが許可されます。タイマ/カウンタ1割り込み要求フラグ レジスタ(TIFR1)に配置された比較A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が実行されます。

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

このビットが**1**を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(**1**)されると、タイマ/カウンタ1溢れ割り込みが許可されます。タイマ/カウンタ1割り込み要求フラグ レジスタ(TIFR1)に配置されたタイマ/カウンタ1溢れ割り込み要求フラグ(TOV1)が設定(**1**)されると、対応する割り込みベクタが実行されます。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TIFR1

変位 : \$36 (\$16)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは各々\$16です。

ビット	7	6	5	4	3	2	1	0
	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1
アクセス種別	R	R	R/W	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● **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/Counter, Input Capture Flag)

ICP1ピンに捕獲の事象が起こると、このフラグが設定(**1**)されます。捕獲レジスタ(ICR1)がWGM13～0によってTOP値として設定されると、ICF1フラグはカウンタがTOP値に到達する時に設定(**1**)されます。

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

● **ビット2 – OCF1B : タイマ/カウンタ1比較B割り込み要求フラグ** (Timer/Counter, Output Compare B Match Flag)

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

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

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

● **ビット1 – OCF1A : タイマ/カウンタ1比較A割り込み要求フラグ** (Timer/Counter, 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)ストロープがOCF1Aフラグを設定(**1**)しないことに注意してください。

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

● **ビット0 – TOV1 : タイマ/カウンタ1溢れ割り込み要求フラグ** (Timer/Counter 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n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波形生成種別ビット記述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。

## 21. タイマ/カウンタ0と1の前置分周器

8ビット タイマ/カウンタ0(TC0)と16ビット タイマ/カウンタ1(TC1)は同じ前置分周器部を共有しますが、タイマ/カウンタは異なる前置分周器設定ができます。以下の記述はTC0とTC1に適用されます。

関連リンク [70頁の「TC0 - 8ビット タイマ/カウンタ \(PWM付き\)」](#)  
[83頁の「TC1 - 16ビット タイマ/カウンタ \(PWM付き\)」](#)

### 21.1. 内部クロック元

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

### 21.2. 前置分周器リセット

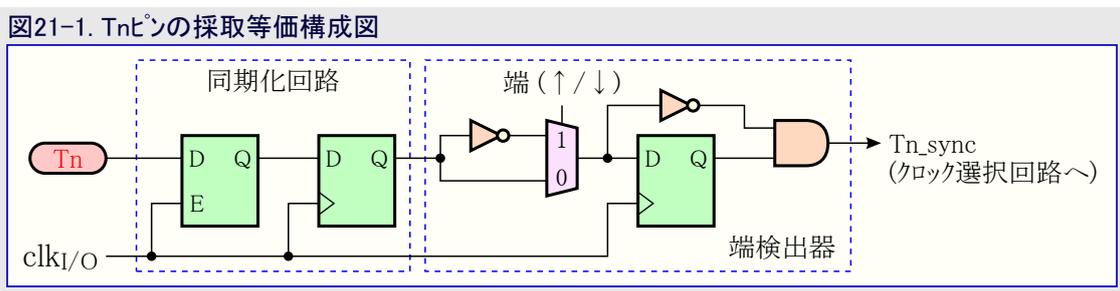
この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ0とタイマ/カウンタ1によって共有されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが使われる状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステム クロック周期数は、Nが前置分周値(8, 64, 256, 1024)とすると、 $1 \sim N+1$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

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

### 21.3. 外部クロック元

Tnピンに印加された外部クロック元はタイマ/カウンタ クロック( $f_{clk\_Tn}$ )として使えます。このTnピンはピン同期化論理回路によって全てのシステム 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。下のTn同期化と端検出器論理回路の構成図もご覧ください。レジスタは内部システム クロック( $f_{clk\_I/O}$ )の上昇端でクロック駆動されます。ラッチは内部システム クロックのHigh区間で通過(Low区間で保持)です。

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



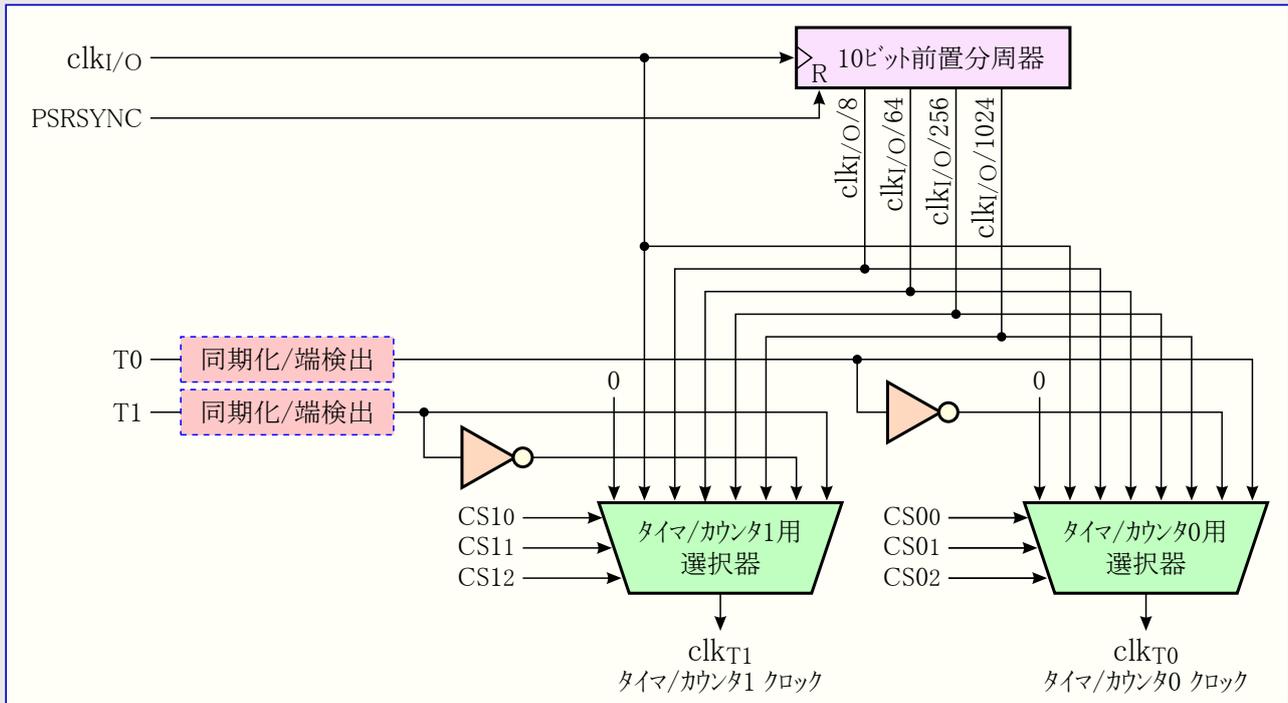
同期化と端検出器論理回路はTnピンへ印加された端から計数が更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

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

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

外部クロック元は前置分周できません。

図21-2. タイマ/カウンタ0と1の前置分周器部構成図



注: 入力ピン(T0/T1)の同期化/端(エッジ)検出論理回路は前の構成図で示されます。

## 21.4. 同期系タイマ/カウンタ前置分周器制御関係レジスタ

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : GTCCR

変位 : \$43 (\$23)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$23です。

ビット	7	6	5	4	3	2	1	0
	TSM	-	-	-	-	-	PSRASY	PSRSYNC
アクセス種別	R/W	R	R	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

#### ● ビット1 – PSRASY : 非同期系タイマ/カウンタ前置分周器リセット (Prescaler Reset Timer/Counter2)

このビットが1の時にタイマ/カウンタ2の前置分周器はリセットします。通常、このビットはハードウェアによって直ちに解除(0)されます。タイマ/カウンタ2が非同期動作の時にこのビットが(1)を書かれると、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。TSMビットが設定(1)される場合、このビットはハードウェアによって解除(0)されません。

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

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

## 22. TC2 – 8ビット タイマ/カウンタ2 (PWM, 非同期動作付き)

### 22.1. 特徴

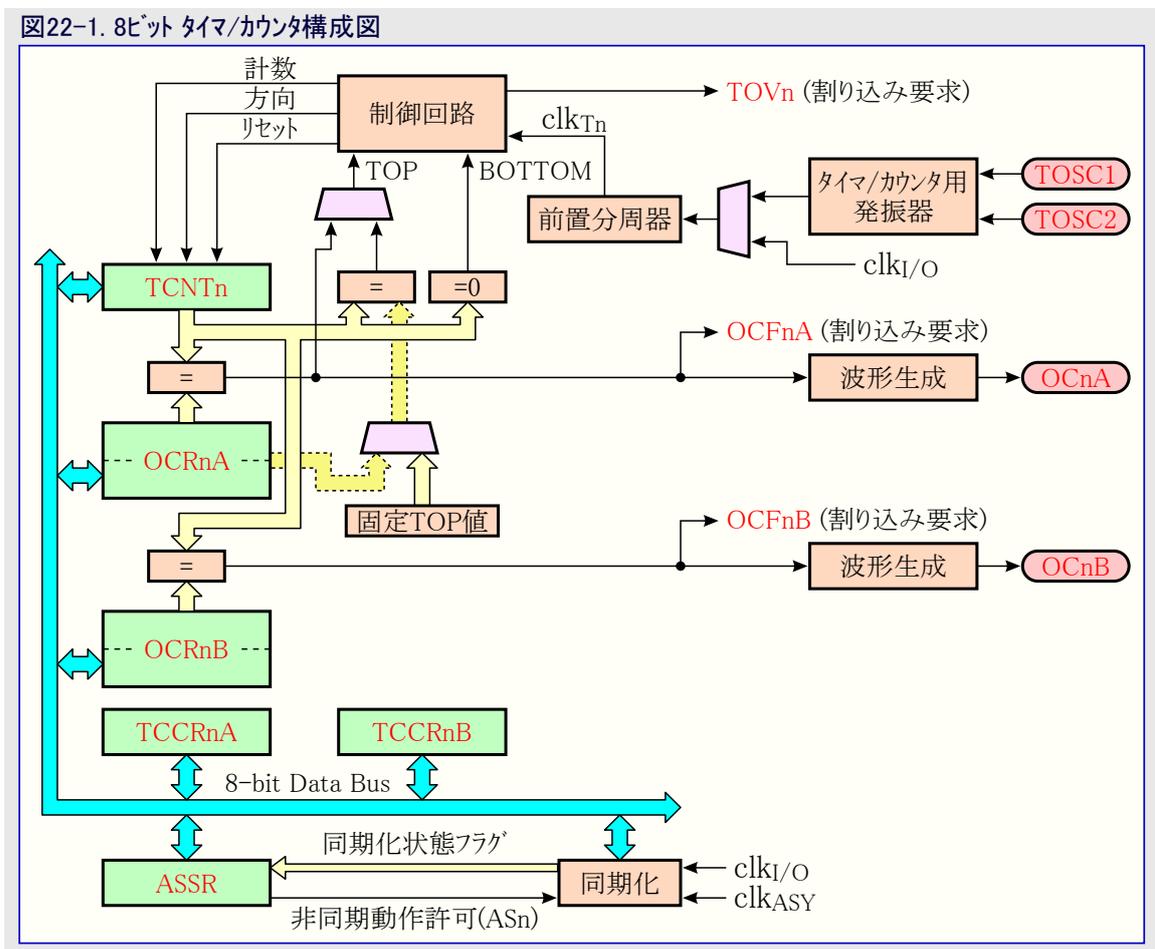
- 2つの比較部付き計数器
- 比較一致でのタイマ/カウンタ解除 (自動再設定)
- 不具合なしで正しい位相のパルス幅変調器 (PWM)
- 周波数発生器
- クロック用10ビット前置分周器
- 溢れと比較一致割り込み (TOV2, OCF2A, OCF2B)
- I/O(システム)クロックに依存しない時計用外部32kHz水晶からのクロック駆動可能

### 22.2. 概要

タイマ/カウンタ2は2つの独立した比較出力部とPWM支援付きの汎用8ビット タイマ/カウンタ部です。

この8ビット タイマ/カウンタの単純化した構成図は下で示されます。CPUがアクセス可能な(I/OビットとI/Oピンを含む)I/Oレジスタは赤文字(訳注:原文は太字)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以降の「8ビット タイマ/カウンタ2用レジスタ」で一覧されます。I/Oピンの実際の配置については「ピン配置」を参照してください。

TC2は電力削減レジスタ(PRR)のPRTIM2ビットが0を書かれる時に許可されます。



関連リンク [10頁の「ピン配置」](#)

#### 22.2.1. 定義

本章でのレジスタとビット参照の多くは以下のように一般形で書かれます。

- $n$ はタイマ/カウンタ番号を表します。
- $x=A, B$ は比較出力部AまたはBを表します。

けれども、プログラムでレジスタまたはビット定義に使う時は正確な形式、即ち、タイマ/カウンタ2のカウンタ値アクセスに対してTCNT2が使われなければなりません。

右の定義は本章全体に渡って使われます。

表22-1. 用語定義

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

## 22.2.2. 関係レジスタ

タイマ/カウンタ(TCNT2)と比較レジスタ(OCR2AとOCR2B)は8ビットのレジスタです。割り込み要求信号は**タイマ/カウンタ2割り込み要求レジスタ(TIFR2)**で全て見えます。全ての割り込みは**タイマ/カウンタ2割り込み許可レジスタ(TIMSK2)**で個別に遮蔽(禁止)されます。TIFR2とTIMSK2はこの図で示されません。

このタイマ/カウンタは内部的、前置分周器経由、または本章内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(OCR2AとOCR2B)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(OC2AとOC2B)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「**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(OCF2AとOCF2B)も設定(1)します。

## 22.3. タイマ/カウンタのクロック

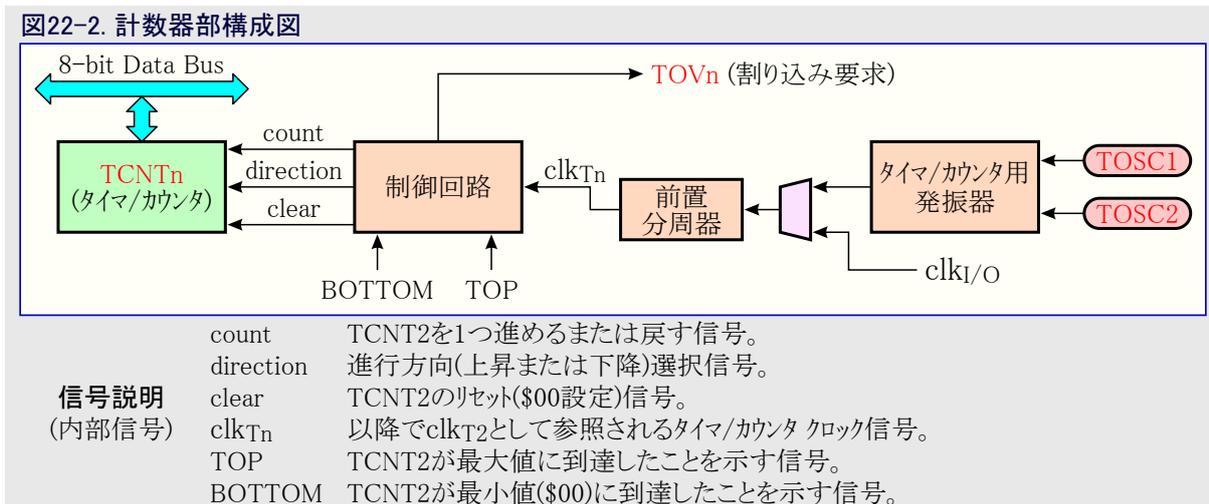
このタイマ/カウンタは内部同期または外部非同期クロック元によってクロック駆動できます。既定のクロック元(clkT2)はMCUクロック(clkI/O)と同じです。

非同期状態レジスタ(ASSR)の**非同期動作許可(AS2)ビット**が1を書かれると、クロック元はTOSC1とTOSC2に繋がったタイマ/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「**ASSR - 非同期状態レジスタ**」の記述をご覧ください。クロック元と前置分周器の詳細については「**タイマ/カウンタ2の前置分周器**」をご覧ください。

## 22.4. 計数器部

8ビットタイマ/カウンタの主な部分は設定可能な双方向計数器部です。下は、この計数器とその周辺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2制御レジスタA(TCCR2A)**に配置された**波形生成種別(WGM21,0)ビット**と**タイマ/カウンタ2制御レジスタB(TCCR2B)**に配置された**波形生成種別(WGM22)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OC2A/OC2B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「**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WGM22~0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。

## 22.5. 比較出力部

この8ビット比較器はTCNT2と比較レジスタ(OCR2AとOCR2B)を継続的に比較します。TCNT2がOCR2AまたはOCR2Bと等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCF2AまたはOCF2B)を設定(1)します。対応する割り込みが許可(I=1, OCIE2AまたはOCIE2B=1)されているなら、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に論理1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM22~0)ビットと比較出力選択(COM2x1,0)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の信号は動作種別(「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。

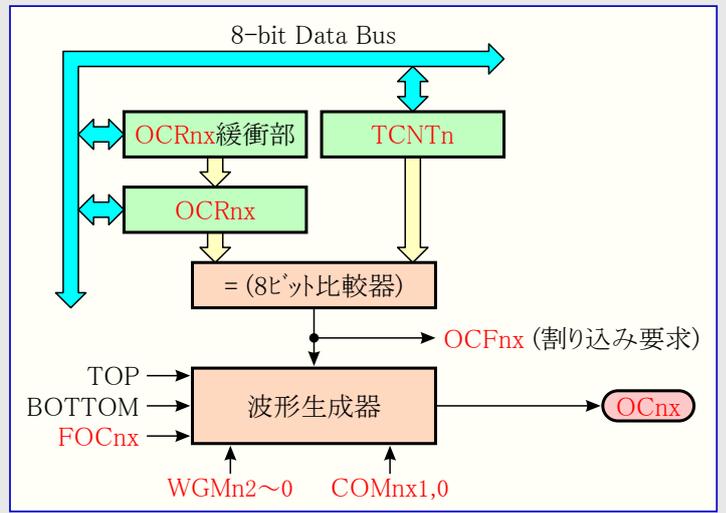
右図は比較出力部の構成図を示します。

OCR2xはパルス幅変調(PWM)の何れかを使う時に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OCR2x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。

OCR2x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OCR2x緩衝部をアクセスし、禁止されるとOCR2xレジスタを直接アクセスします。

関連リンク 108頁の「動作種別」

図22-3. 比較出力部構成図



(訳注) ここでは比較nxレジスタ全体をOCR2x、OCR2xを構成する緩衝部部分をOCR2x緩衝部、実際の比較に使用されるレジスタ本体部分をOCR2x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。

### 22.5.1. 強制比較出力

非PWM波形生成動作での比較器の一致出力は強制変更(FOC2x)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(OCF2x)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、OC2x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(COM2x1,0ビット設定がOC2xピンの設定(1)、解除(0)、1/0交互のどれかを定義)。

### 22.5.2. 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、例えばタイマ/カウンタが停止されているにも、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OCR2xに許します。

### 22.5.3. 比較一致部の使用

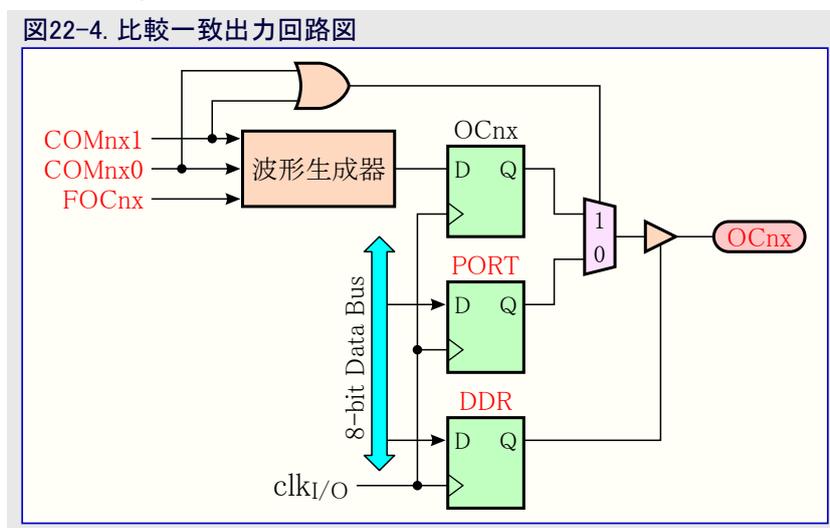
どの動作種別でのTCNT2書き込みでも1タイマ/カウンタクロック周期間、全ての比較一致を妨げるため、タイマ/カウンタが走行中であるか否かに拘らず、比較出力部を使う場合、TCNT2を変更する時に危険を伴います。TCNT2に書かれた値がOCR2x値と同じ場合に比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。同様にタイマ/カウンタが下降計数のとき、BOTTOMに等しいTCNT2値を書き込んではいけません。

OC2x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。OC2x値を設定する一番簡単な方法は標準動作で強制変更(FOC2x)スローブビットを使うことです。波形生成動作種別間を変更する時でも、OC2x(内部)レジスタはその値を保ちます。

比較出力選択(COM2x1,0)ビットが比較値(OCR2x)と共に2重緩衝部されないことに気付いてください。COM2x1,0ビットの変更は直ちに有効となります。

## 22.6. 比較一致出力部

比較出力選択(COM2x1,0)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(OC2x)状態の定義にCOM2x1,0ビットを使います。またCOM2x1,0ビットはOC2xピン出力元を制御します。下図はCOM2x1,0ビット設定によって影響を及ぼされる論理回路の簡略化した図を示します。図のI/Oレジスタ、I/Oビット、I/Oピンは赤文字(訳注:原文は太字)で示されます。COM2x1,0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。OC2xの状態を参照するとき、その参照はOC2xピンでなく内部OC2xレジスタに対してです。



COM2x1,0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(OC2x)によって無効にされます。けれどもOC2x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。OC2xピンに対するポート方向レジスタのビット(DDR\_OC2x)はOC2x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OC2x状態の初期化を許します。いくつかのCOM2x1,0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。「[8ビットタイマ/カウンタ2用レジスタ](#)」をご覧ください。

[関連リンク](#) 108頁の「[動作種別](#)」

### 22.6.1. 比較一致出力選択と波形生成

波形生成器は標準、CTC、PWM動作でCOM2x1,0ビットを違うふうに使います。全ての動作種別に対してCOM2x1,0=00設定は次の比較一致で実行すべきOC2xレジスタの動きがないことを波形生成器へ告げます。[比較出力選択の記述](#)も参照してください。

COM2x1,0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\(FOC2x\)ストローブビット](#)を使うことによって直ちに効果を得ることを強制できます。

## 22.7. 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(WGM22~0)ビットと比較出力選択(COM2x1,0)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。COM2x1,0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COM2x1,0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(「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「タイマ/カウンタ2のタイミング」を参照してください。

### 22.7.1. 標準動作

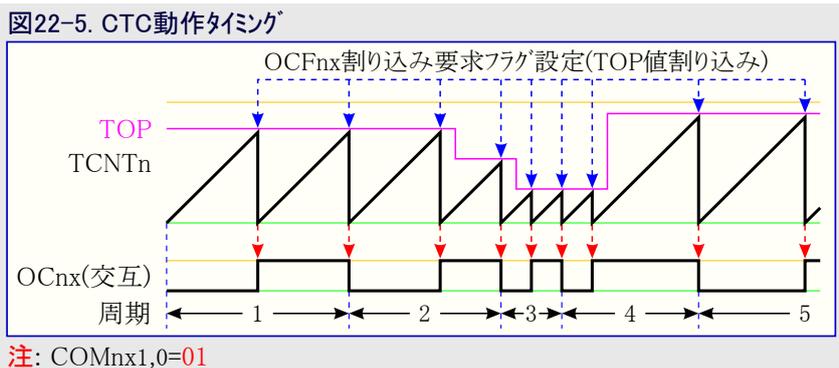
最も単純な動作種別が標準動作(WGM22~0=000)です。この動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV2)フラグはTCNT2が\$00になる時と同じタイマ/カウンタクロック周期で設定(1)されます。この場合のTOV2フラグは設定(1)のみで解除(0)されないことを除いて第9ビットのようになります。けれどもTOV2フラグを自動的に解除(0)するタイマ/カウンタ2溢れ割り込みと組み合わせたタイマ/カウンタの分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。

### 22.7.2. 比較一致タイマ/カウンタ解除(CTC)動作

比較一致タイマ/カウンタ解除(CTC)動作(WGM22~0=010)ではOCR2A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(TCNT2)値がOCR2Aと一致すると、カウンタは\$00に解除されます。OCR2A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下で示されます。カウンタ(TCNT2)値はTCNT2とOCR2A間で比較一致が起こるまで増加し、そしてその後カウンタ(TCNT2)は解除(\$00)されます。



OCF2Aフラグを使うことにより、タイマ/カウンタ値がTOP値に達する時毎に割り込みが生成できます。割り込みが許可されるなら、割り込み処理ルーチンはTOP値を更新するのに使えます。けれども前置分周なしまたは低い前置分周値でカウンタが走行している時にBOTTOMと近い値にTOPを変更することは、CTC動作が2重緩衝機能を持たないために注意して行わなければなりません。OCR2A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、OC2A出力は比較出力選択(COM2A1,0)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。OC2A値はそのピンに対するデータ方向が出力(DDR\_OC2A=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OCR2Aが0(\$00)に設定される時に $f_{OC2A} = f_{clk\_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

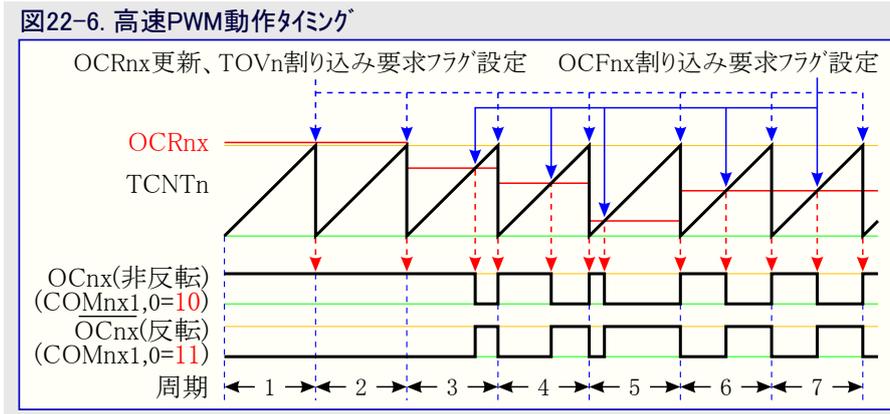
変数Nは前置分周数(1,8,32,64,128,256,1024)を表します。

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。

### 22.7.3. 高速PWM動作

高速パルス幅変調(PWM)動作(WGM22~0=011または111)は高周波数PWM波形生成選択を提供します。高速PWMはそれが単一傾斜(鋸波)動作であることによって他のPWM動作と異なります。カウンタはBOTTOMからTOPまで計数し、その後BOTTOMから再び始めます。TOPはWGM22~0=011時に\$FF、WGM22~0=111時にOCR2Aとして定義されます。非反転比較出力動作(COM2x1,0=10)での比較出力(OC2x)はTCNT2とOCR2x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(COM2x1,0=11)の出力は比較一致で設定(1)され、BOTTOMで解除(0)されます。この単一傾斜動作のため、高速PWM動作の動作周波数は両傾斜(三角波)動作を使う位相基準PWM動作よりも2倍高くできます。この高い周波数は電力調節、整流、D/A変換に対して高速PWM動作を都合よく適合させます。高い周波数は物理的に小さな外部部品(コイルやコンデンサ)を許し、従ってシステム総費用を削減します。

高速PWM動作でのタイマ/カウンタはタイマ/カウンタ値がTOP値と一致するまで増加されます。そしてタイマ/カウンタは(一致の)次のタイマ/カウンタクロック周期で解除(\$00)されます。高速PWM動作のタイミング図は下図で描かれます。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。この図は非反転と反転のPWM出力を含みます。赤細線はOCR2x値を示し、TCNT2値との交点(接点)がTCNT2とOCR2x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(OCF2x)はOCR2x=TOPを除いて比較一致が起こると設定(1)されます(訳注:共通性のため本行追加)。



タイマ/カウンタ溢れ(TOV2)フラグはカウンタがTOPに到達する時毎に設定(1)されます。割り込みが許可されるなら、その割り込み処理ルーチンは比較値を更新するために使えます。

高速PWM動作での比較部はOC2xピンでのPWM波形の生成を許します。COM2x1,0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COM2x1,0を'11'に設定することで生成できます。WGM22ビットが設定(1)なら、COM2A1,0ビットの'01'設定は比較一致での交互反転をOC2Aピンに許します。この任意選択はOC2Bピンに対して利用できません(訳注:前2行修正追加)。実際のOC2x値はポートピンに対するデータ方向が出力として設定される場合にだけ見えます。PWM波形はTCNT2とOCR2x間の比較一致でOC2x(内部)レジスタを設定(1)(または解除(0))と、カウンタが解除(\$00、TOPからBOTTOMへ変更)されるタイマ/カウンタクロック周期でOC2xレジスタを解除(0)または設定(1)することによって生成されます。

PWM出力周波数は次式によって計算できます。

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

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

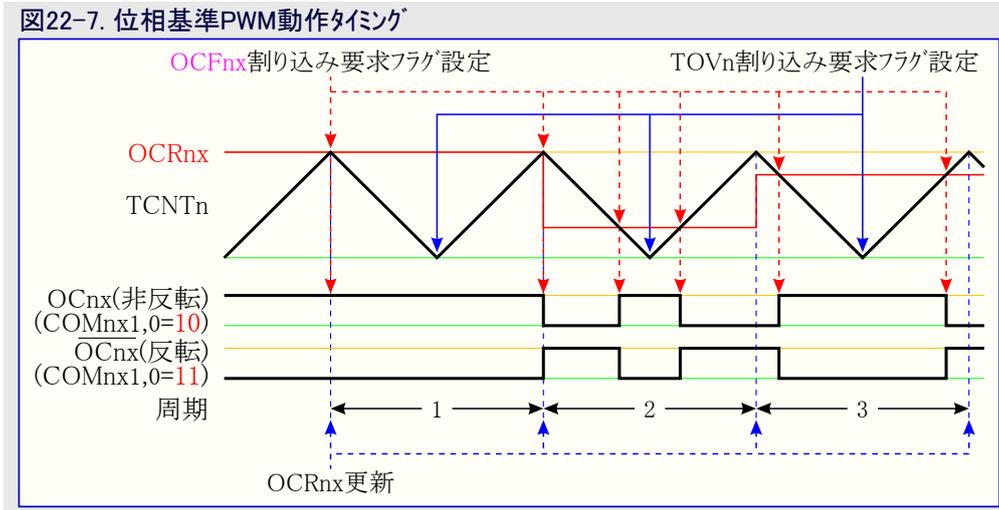
OCR2xの両端値は高速PWM動作でPWM波形出力を生成する時の特別な場合にあたりま。OCR2xがBOTTOM(\$00)と等しく設定されると、出力はTOP+1 タイマ/カウンタクロック周期毎の狭い尖頭(パルス)になるでしょう。OCR2xがTOPに等しく設定されると、(COM2x1,0ビットによって設定される出力極性に依存して)定常的なLowまたはHigh出力に終わるでしょう。

(訳補:WGM22~0=111の場合については、)高速PWM動作での(デューティ比50%)周波数の波形出力は比較一致毎に論理反転するOC2A設定(COM2A1,0=01)によって達成できます。生成された波形はOCR2Aが0(\$00)に設定される時に $f_{OC2x} = f_{clk_{I/O}} / 2$ 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OC2A交互出力(COM2A1,0=01)と同じです。

## 22.7.4. 位相基準PWM動作

位相基準パルス幅変調(PWM)動作(WGM22~0=001または101)は高分解能で正しい位相のPWM波形生成選択を提供します。位相基準PWM動作は両傾斜(三角波)動作に基づきます。カウンタはBOTTOMからTOPへそして次にTOPからBOTTOMへを繰り返し計数します。TOPはWGM22~0=001時に\$FF、WGM22~0=101時にOCR2Aとして定義されます。非反転比較出力動作(COM2x1,0=10)での比較出力(OC2x)は上昇計数中のTCNT2とOCR2x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(COM2x1,0=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動作でのカウンタはカウンタ値がTOPと一致するまで増加されます。カウンタはTOPに達すると計数方向を替えます。このTCNT2値は1タイマ/カウンタクロック周期間TOPと等しくなります。位相基準PWM動作のタイミング図は図19-7.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OCR2x値を示し、TCNT2値との交点(接点)がTCNT2とOCR2x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BOTTOM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OC2xピンでのPWM波形の生成を許します。COM2x1,0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COM2x1,0ビットを'11'に設定することで生成できます。WGM02ビットが設定(1)なら、COM0A1,0ビットの'01'設定は比較一致での交互反転をOC0Aピンに許します。この任意選択はOC0Bピンに対して利用できません(訳注:前2行修正追加)。実際のOC2x値はそのポートピンに対するデータ方向(DDR\_OC2x)が出力として設定される場合にだけ見えます。PWM波形はカウンタが増加する時のTCNT2とOCR2x間の比較一致でOC2x(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TCNT2とOCR2x間の比較一致でOC2xレジスタを解除(0)(または設定(1))によって生成されます。位相基準PWMを使う時の出力に対するPWM周波数は次式によって計算できます。

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

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

OCR2xの両端値は位相基準PWM動作でPWM波形出力を生成する時の特別な場合に当たります。非反転PWM動作ではOCR2xがBOTTOM(\$00)に等しく設定されると出力は定常的なLow、TOPに等しく設定されると定常的なHighになるでしょう。反転PWMに対する出力は逆の論理値になります。

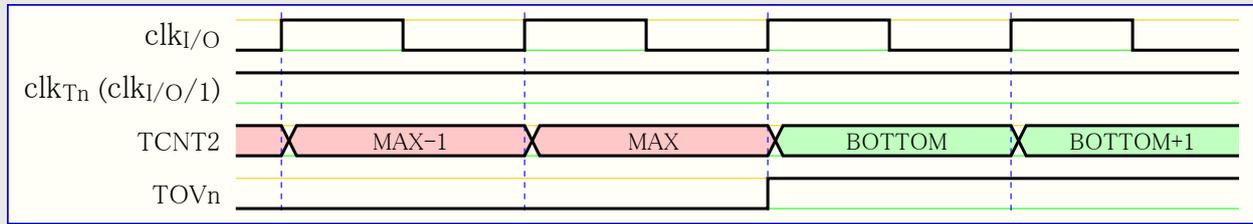
上図の第2周期のその出発点に於いて、例えば比較一致がないとしても、OCnxにはHighからLowへの遷移があります。この遷移点はBOTTOMを挟む対称を保証するためです。比較一致なしに遷移を生ずるのは2つの場合です。

- 前図で示されるようにOCR2xはTOPからその値を変更します。OCR2x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR2x値よりも高い値から数え始め、そしてその理屈のために比較一致、それ故上昇途中で起こされるであろうOCnxの変更を逃します。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)

## 22.8. タイマ/カウンタのタイミング

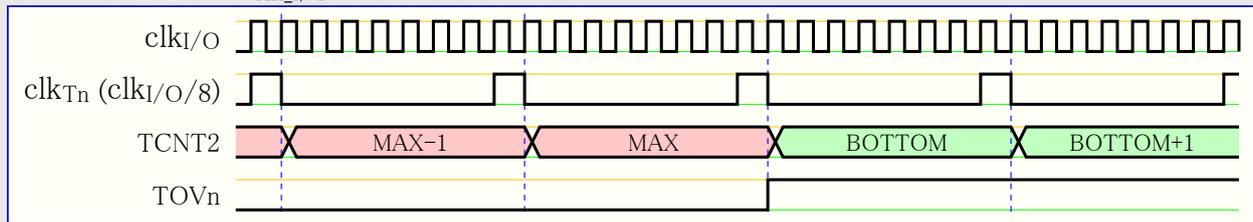
以下の図は同期動作でのタイマ/カウンタを示し、従ってタイマ/カウンタクロック( $clk_{Tn}$ )が計数許可信号として示されます。非同期動作では $clk_{I/O}$ がタイマ/カウンタ用発振器(TOSC)クロックによって置換されるべきです。この図は割り込みフラグが設定(1)される時の情報を含みます。下図は基本的なタイマ/カウンタ動作についてのタイミング データを示します。この図は位相基準PWM動作以外の全ての動作種別でのMAX値近辺の計数の流れを示します。

図22-8. 前置分周なし(1/1)のタイマ/カウンタ タイミング



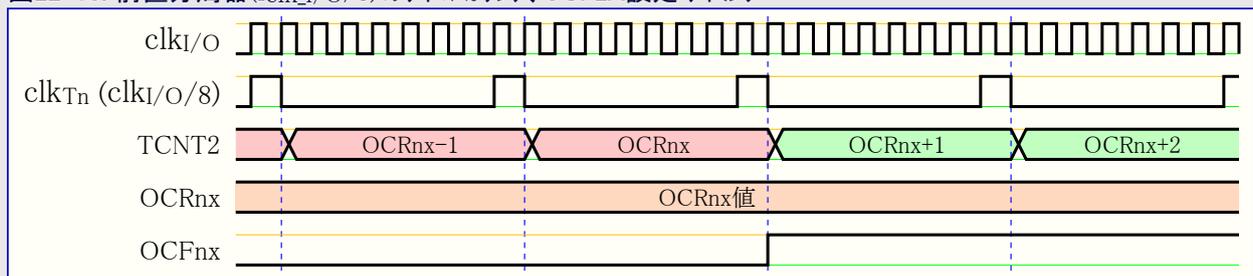
下図は同じタイミング データを示しますが、前置分周器が許可されています。

図22-9. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ タイミング



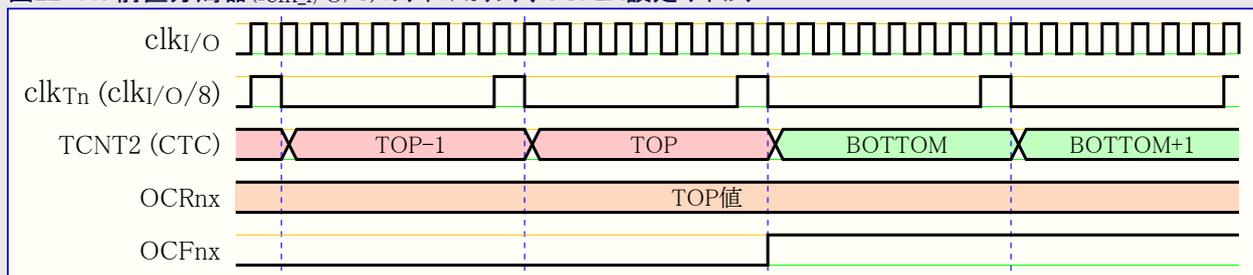
下図はCTC動作を除く全ての動作種別でのOCF2Aの設定を示します。

図22-10. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF2A設定 タイミング



下図はCTC動作でのTCNT2の解除とOCF2Aの設定を示します。

図22-11. 前置分周器( $f_{clk_{I/O}/8}$ )のタイマ/カウンタ、OCF2A設定 タイミング



## 22.9. タイマ/カウンタ2の非同期動作

TC2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。

- 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比較レジスタ(OCR2x)**、**タイマ/カウンタ2制御レジスタ(TCCR2x)**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2割り込み許可レジスタ(TIMSK2)**のOCIE2xとTOIE2の解除(0)により、TC2割り込みを禁止してください。
  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んでください。
  3. TCNT2,OCR2x,TCCR2xに新しい値を書いてください。
  4. 非同期動作へ切り替えるには、**TCN2UB**、**OCR2xUB**、**TCR2xUB**について(=0まで)待機してください。
  5. **タイマ/カウンタ2割り込み要求フラグ レジスタ(TIFR2)**のOCF2xとTOV2のフラグを解除(0)してください。
  6. 必要とされるなら、割り込みを許可してください。
- CPU主クロック周波数はタイマ発振器周波数の4倍よりも高くなければなりません。
- TCNT2,OCR2x,TCCR2x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の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OCR2x書き込みの実行を妨げないことを意味します。非同期状態レジスタ(ASSR)は転送先レジスタへの転送が起きたことを示します。
- TCNT2,OCR2x,TCCR2xに書いた後で**パワーセーブ**または**A/D変換雑音低減**動作へ移行するとき、デバイスを起動するのにTC2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と、TCNT2またはOCR2x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、OCR2xUBが0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**または**A/D変換雑音低減**動作からデバイスを起動するのにTC2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。1 TOSC1周期以内に休止形態へ再移行する場合、割り込みが直ちに起こり、デバイスは再び起動復帰するでしょう。その結果は複数の割り込みと最初の割り込みから1 TOSC1周期以内の起動復帰です。**パワーセーブ**または**A/D変換雑音動作**再移行前の時間が充分かどうか迷うなら、1 TOSC1周期が経過されることを保証するのに次の手順が使えます。
  1. TCNT2,OCR2x,TCCR2xに値を書いてください。
  2. 非同期状態レジスタ(ASSR)の対応する更新中フラグが0に戻るまで待ってください。
  3. **パワーセーブ**または**A/D変換雑音低減**動作へ移行してください。
- **非同期動作が選ばれる**と、TC2用32.768kHz発振器は**パワーダウン**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、TC2を使う前に少なくとも1秒待つ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、起動時の不安定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、TC2の全レジスタの内容が失われたと見做されなければなりません。
- TC2が非同期でクロック駆動される時の**パワーセーブ**または**A/D変換雑音低減**動作から起動の説明。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。起動後にMCUは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。
- **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<sub>I/O</sub>)が再び活性(有効)になると、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. OCR2xまたはTCCR2xのどれ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。

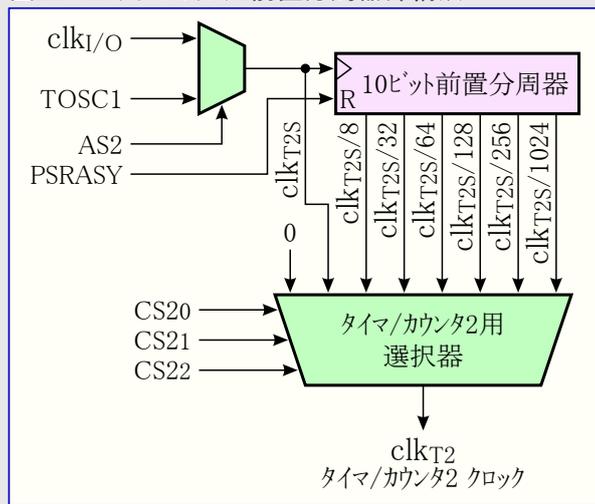
## 22.10. タイマ/カウンタ2の前置分周器

TC2用クロック元の名前は $clk_{T2S}$ です。既定での $clk_{T2S}$ は主システムI/Oクロック( $clk_{I/O}$ )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)の非同期クロック許可(AS2)ビットの1書き込みにより、TC2はTOSC1ピ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TC2の使用を可能にします。AS2が設定(1)されると、TOSC1とTOSC2のピンは主クロック発振器や標準ポートから切り離されます。クリスタル発振子はTC2用の独立したクロック元として扱うTOSC1とTOSC2ピ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に外部クロック信号を印加する場合、ASSRの外部クロック許可(EXCLK)ビットが設定(1)されなければなりません(訳注:矛盾回避のため本行置換)。

TC2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。更に0(停止)は勿論 $clk_{T2S}$ も選択可能です。

この前置分周器は一般タイマ/カウンタ制御レジスタ(GTCCR)の非同期系タイマ/カウンタ前置分周器リセット(PSRASY)ビットへの1書き込みによってリセットします。これは明確にした前置分周器での操作を使用者に許します。

図22-12. タイマ/カウンタ2前置分周器部構成



## 22.11. 8ビット タイマ/カウンタ2用レジスタ

### 22.11.1. TCCR2A – タイマ/カウンタ2制御レジスタA (Timer/Counter2 Control Register A)

名称 : TCCR2A  
 変位 : \$B0  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	COM2A1,0		COM2B1,0		-	-	WGM21,0	
アクセス種別	R/W	R/W	R/W	R/W	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7,6 – COM2A1,0 : 比較A出力選択 (Compare Output Mode for Channel A)

これらのビットはOC2A比較出力ピンの動作を制御します。COM2A1,0ビットの1つまたは両方が1を書かれると、OC2A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、OC2A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。

OC2Aがピンに接続されるとき、COM2A1,0ビットの機能はWGM22~0ビット設定に依存します。

下表はWGM22~0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COM2A1,0ビットの機能を示します。

表22-2. 非PWM動作比較A出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A切断)
0	1	比較一致でOC2Aピン トグル(交互)出力
1	0	比較一致でOC2Aピン Lowレベル出力
1	1	比較一致でOC2Aピン Highレベル出力

下表はWGM22~0ビットが高速PWM動作に設定される時のCOM2A1,0ビットの機能を示します。

表22-3. 高速PWM動作比較A出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A切断)
0	1	WGM22=0 : 標準ポート動作 (OC2A切断) WGM22=1 : 比較一致でOC2Aピン トグル(交互)出力
1	0	比較一致でLow、BOTTOMでHighをOC2Aピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC2Aピンへ出力 (反転動作)

注: COM2A1が設定(1)され、OCR2A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、設定(1)または解除(0)はBOTTOMで行われます。詳細については「高速PWM動作」をご覧ください。

下表はWGM02~0ビットが位相基準PWM動作に設定される時のCOM0A1,0ビットの機能を示します。

表22-4. 位相基準PWM動作比較A出力選択

COM2A1	COM2A0	意味
0	0	標準ポート動作 (OC2A切断)
0	1	WGM22=0 : 標準ポート動作 (OC2A切断) WGM22=1 : 比較一致でOC2Aピン トグル(交互)出力
1	0	上昇計数時の比較一致でLow、下降計数時の比較一致でHighをOC2A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OC2Aピンへ出力

注: COM2A1が設定(1)され、OCR2A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、設定(1)または解除(0)はTOPで行われます。詳細については「位相基準PWM動作」をご覧ください。

#### ● ビット5,4 – COM2B1,0 : 比較B出力選択 (Compare Output Mode for Channel B)

これらのビットはOC2B比較出力ピンの動作を制御します。COM2B1,0ビットの1つまたは両方が1を書かれると、OC2B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、OC2B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。

OC2Bがピンに接続されるとき、COM2B1,0ビットの機能はWGM22~0ビット設定に依存します。

下表はWGM22~0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COM2B1,0ビット機能を示します。

表22-5. 非PWM動作比較B出力選択

COM2B1	COM2B0	意味
0	0	標準ポート動作 (OC2B切断)
0	1	比較一致でOC2Bピン(交互)出力
1	0	比較一致でOC2Bピン Lowレベル出力
1	1	比較一致でOC2Bピン Highレベル出力

下表はWGM22~0ビットが**高速PWM動作**に設定される時のCOM2B1,0ビットの機能を示します。

表22-6. 高速PWM動作比較B出力選択

COM2B1	COM2B0	意味
0	0	標準ポート動作 (OC2B切断)
0	1	(予約)
1	0	比較一致でLow、BOTTOMでHighをOC2Bピンへ出力 (非反転動作)
1	1	比較一致でHigh、BOTTOMでLowをOC2Bピンへ出力 (反転動作)

注: COM2B1が設定(1)され、OCR2B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、設定(1)または解除(0)はBOTTOMで行われます。詳細については「**高速PWM動作**」を参照してください。

下表はWGM22~0ビットが**位相基準PWM動作**に設定される時のCOM2B1,0ビットの機能を示します。

表22-7. 位相基準PWM動作比較B出力選択

COM2B1	COM2B0	意味
0	0	標準ポート動作 (OC2B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OC2B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OC2Bピンへ出力

注: COM2B1が設定(1)され、OCR2B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、設定(1)または解除(0)はTOPで行われます。詳細については「**位相基準PWM動作**」を参照してください。

● **ビット1,0 - WGM21,0 : 波形生成種別 (Waveform Generation Mode)**

タイマ/カウンタ制御レジスタ**(TCCR2B)**で得られる**WGM22ビット**と組み合わせたこれらのビットはカウンタの計数順序(方向)、最大カウンタ(TOP)値の供給元、使われるべき波形生成のどの形式かを制御します。タイマ/カウンタ部によって支援される動作種別は標準動作(カウンタ)、比較一致タイマ/カウンタ解除(CTC)動作と2形式のパルス幅変調(PWM)動作です。「**動作種別**」をご覧ください。

表22-8. 波形生成種別選択

番号	WGM22	WGM21	WGM20	タイマ/カウンタ動作種別	TOP値	OCR2x更新時	TOV2設定時
0	0	0	0	標準動作	\$FF	即時	MAX
1	0	0	1	8ビット位相基準PWM動作	\$FF	TOP	BOTTOM
2	0	1	0	比較一致タイマ/カウンタ解除(CTC)動作	OCR2A	即時	MAX
3	0	1	1	8ビット高速PWM動作	\$FF	BOTTOM	MAX
4	1	0	0	(予約)	-	-	-
5	1	0	1	位相基準PWM動作	OCR2A	TOP	BOTTOM
6	1	1	0	(予約)	-	-	-
7	1	1	1	高速PWM動作	OCR2A	BOTTOM	TOP

注: MAX=\$FF、BOTTOM=\$00です。

## 22.11.2. TCCR2B – タイマ/カウンタ2制御レジスタB (Timer/Counter2 Control Register B)

名称 : TCCR2B

変位 : \$B1

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	FOC2A	FOC2B	-	-	WGM22	CS22~0		
アクセス種別	R/W	R/W	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

### ● ビット7 – FOC2A : OC2A強制変更 (Force Output Compare A)

FOC2AビットはWGM22~0ビットが非PWM動作を指示する時だけ有効です。

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TCCR2Bが書かれる場合、このビットは0に設定されなければなりません。FOC2A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。OC2A出力はCOM2A1,0ビット設定に従って変更されます。FOC2A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定めるのはCOM2A1,0ビットに存在する値です。

FOC2Aストローブは何れの割り込みの生成もTOPとしてOCR2A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。

FOC2Aビットは常に0として読みます。

### ● ビット6 – FOC2B : OC2B強制変更 (Force Output Compare B)

FOC2BビットはWGM22~0ビットが非PWM動作を指示する時だけ有効です。

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TCCR2Bが書かれる場合、このビットは0に設定されなければなりません。FOC2B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。OC2B出力はCOM2B1,0ビット設定に従って変更されます。FOC2B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定めるのはCOM2B1,0ビットに存在する値です。

FOC2Bストローブは何れの割り込みの生成も行いません。

FOC2Bビットは常に0として読みます。

### ● ビット3 – WGM22 : 波形生成種別 (Waveform Generation Mode bit 2)

タイマ/カウンタ2制御レジスタA(TCCR2A)を参照してください。

### ● ビット2~0 – CS22~0 : クロック選択 (Clock Select)

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22-9. 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 <sub>T2S</sub> (前置分周なし)
0	1	0	clk <sub>T2S</sub> /8 (8分周)
0	1	1	clk <sub>T2S</sub> /32 (32分周)
1	0	0	clk <sub>T2S</sub> /64 (64分周)
1	0	1	clk <sub>T2S</sub> /128 (128分周)
1	1	0	clk <sub>T2S</sub> /256 (256分周)
1	1	1	clk <sub>T2S</sub> /1024 (1024分周)

### 22.11.3. TCNT2 – タイマ/カウンタ2計数值レジスタ (TC2 Counter Value Register)

名称 : TCNT2  
変位 : \$B2  
リセット : \$00  
特質 : -

ビット	7	6	5	4	3	2	1	0
	TCNT27~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – TCNT27~0 : タイマ/カウンタ2計数值 (TC2 Counter Value)

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OCR2x間の比較一致消失の危険を誘発します。

### 22.11.4. OCR2A – タイマ/カウンタ2比較Aレジスタ (Output Compare Register 2 A)

名称 : OCR2A  
変位 : \$B3  
リセット : \$00  
特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR2A7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – OCR2A7~0 : タイマ/カウンタ2比較A値 (Output Compare 2 A)

この比較レジ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OC2Aピンでの波形出力を生成するのに使えます。

### 22.11.5. OCR2B – タイマ/カウンタ2比較Bレジスタ (Output Compare Register 2 B)

名称 : OCR2B  
変位 : \$B4  
リセット : \$00  
特質 : -

ビット	7	6	5	4	3	2	1	0
	OCR2B7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – OCR2B7~0 : タイマ/カウンタ2比較B値 (Output Compare 2 B)

この比較レジ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OC2Bピンでの波形出力を生成するのに使えます。

## 22.11.6. TIMSK2 – タイマ/カウンタ2割り込み許可レジスタ (Timer/Counter 2 Interrupt Mask Register)

名称 : TIMSK2  
 変位 : \$70  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCIE2B	OCIE2A	TOIE2
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

### ● ビット2 – OCIE2B : タイマ/カウンタ2比較B割り込み許可 (Timer/Counter2 Output Compare Match B Interrupt Enable)

OCIE2B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B一致割り込みが許可されます。タイマ/カウンタ2で比較B一致が起こる、換言するとタイマ/カウンタ2割り込み要求フラグレジスタ(TIFR2)で比較B割り込み要求フラグ(OCF2B)が設定(1)されると、対応する割り込みが実行されます。

### ● ビット1 – OCIE2A : タイマ/カウンタ2比較A割り込み許可 (Timer/Counter2 Output Compare Match A Interrupt Enable)

OCIE2A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A一致割り込みが許可されます。タイマ/カウンタ2で比較A一致が起こる、換言するとタイマ/カウンタ2割り込み要求フラグレジスタ(TIFR2)で比較A割り込み要求フラグ(OCF2A)が設定(1)されると、対応する割り込みが実行されます。

### ● ビット0 – 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2割り込み要求フラグレジスタ(TIFR2)でタイマ/カウンタ2溢れ割り込み要求(TOV2)フラグが設定(1)されると、対応する割り込みが実行されます。

## 22.11.7. TIFR2 – タイマ/カウンタ2割り込み要求フラグレジスタ (Timer/Counter 2 Interrupt Flag Register)

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : TIFR2  
 変位 : \$37 (\$17)  
 リセット : \$00  
 特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$17です。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCF2B	OCF2A	TOV2
アクセス種別	R	R	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

### ● ビット2 – OCF2B : タイマ/カウンタ2比較B割り込み要求フラグ (Timer/Counter2, Output Compare B Match Flag)

OCF2Bビットは比較一致がタイマ/カウンタ(TCNT2)と比較レジスタ(OCR2B)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF2B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OCF2B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2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ2比較B一致割り込み許可(OCIE2B)ビット、OCF2Bが設定(1)されると、タイマ/カウンタ2比較B一致割り込みが実行されます。

### ● ビット1 – OCF2A : タイマ/カウンタ2比較A割り込み要求フラグ (Timer/Counter2, Output Compare A Match Flag)

OCF2Aビットは比較一致がタイマ/カウンタ(TCNT2)と比較レジスタ(OCR2A)間で起こる時に設定(1)されます。対応する割り込み処理ベクタを実行すると、OCF2A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OCF2A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2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ2比較A一致割り込み許可(OCIE2A)ビット、OCF2Aが設定(1)されると、タイマ/カウンタ2比較A一致割り込みが実行されます。

### ● ビット0 – 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/カウンタ(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ベクタを実行すると、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/カウンタ2割り込み許可レジスタ(TIMSK2)のタイマ/カウンタ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/カウンタ2溢れ割り込みが実行されます。位相基準PWM動作ではタイマ/カウンタ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。

これらフラグの設定はWGM22～0ビット設定に依存します。波形生成種別ビット記述の表22-8を参照してください。

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

名称 : ASSR

変位 : \$B6

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	EXCLK	AS2	TCN2UB	OCR2AUB	OCR2BUB	TCR2AUB	TCR2BUB
アクセス種別	R	R/W	R/W	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

### ● ビット6 – EXCLK : 外部クロック信号許可 (Enable External Clock Input)

EXCLKが1を書かれ、非同期クロックが選ばれると、外部クロック入力緩衝部が許可され、32kHzクリスタルの代わりに外部クロックがタイマ発振器1(TOSC1)ピンへ入力できます。EXCLKへの書き込みは非同期動作が選ばれる前に行うべきです。クリスタル発振器はこのビットが0の時だけに走行(動作)することに注意してください。

### ● ビット5 – 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)**、**比較レジスタ(OCR2A,OCR2B)**、**タイマ/カウンタ2制御レジスタ(TCCR2A,TCCR2B)**の内容は不正にされるかもしれません。

### ● ビット4 – TCN2UB : タイマ/カウンタ2更新中フラグ (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3 – OCR2AUB : タイマ/カウンタ2比較Aレジスタ更新中フラグ (Output Compare A Register2 Update Busy)

タイマ/カウンタ2が非同期に動き、比較Aレジスタ(OCR2A)が書かれると、このビットが設定(1)になります。OCR2A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OCR2A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2 – OCR2BUB : タイマ/カウンタ2比較Bレジスタ更新中フラグ (Output Compare B Register2 Update Busy)

タイマ/カウンタ2が非同期に動き、比較Bレジスタ(OCR2B)が書かれると、このビットが設定(1)になります。OCR2B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OCR2B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 – TCR2AUB : タイマ/カウンタ2制御レジスタA更新中フラグ (Timer/Counter2 Control Register A 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制御レジスタA(TCCR2A)が書かれると、このビットが設定(1)になります。TCCR2A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TCCR2A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 – TCR2BUB : タイマ/カウンタ2制御レジスタB更新中フラグ (Timer/Counter2 Control Register B 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制御レジスタB(TCCR2B)が書かれると、このビットが設定(1)になります。TCCR2B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TCCR2B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。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : GTCCR

変位 : \$43 (\$23)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$23です。

ビット	7	6	5	4	3	2	1	0
	TSM	-	-	-	-	-	PSRASY	PSRSYNC
アクセス種別	R/W	R	R	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

### ● ビット1 – PSRASY : 非同期系タイマ/カウンタ前置分周器リセット (Prescaler Reset Timer/Counter2)

このビットが1の時にタイマ/カウンタ2の前置分周器はリセットします。通常、このビットはハードウェアによって直ちに解除(0)されます。タイマ/カウンタ2が非同期動作の時にこのビットが(1)を書かれると、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。TSMビットが設定(1)される場合、このビットはハードウェアによって解除(0)されません。

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

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

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

### 23.1. 特徴

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

### 23.2. 概要

直列周辺インターフェースはデバイスと種変機能部間や様々なAVRデバイス間の高速同期データ転送を許します。

USARTはSPI主装置動作でも使えます。「USARTでのSPI動作」章を参照してください。

SPI部を許可するために電力削減レジスタの直列周辺インターフェース電力削減(PRR.PRSP)ビットは0を書かれなければなりません。

SPIでの主装置と従装置のCPU間相互接続は下図で示されます。このシステムは2つの移動レジスタと主装置クロック発生器から成ります。SPI主装置は希望した従装置のSS(従装置選択)ピンをLowへ引き込む時に一群の通信を開始します。主装置と従装置は各々の移動レジスタに送出すべきデータを用意し、主装置はデータを交換するのに必要なクロックパルスを生じます。データは常にMOSI(Master Out Slave In)信号線を主装置から従装置へ、MISO(Master In Slave Out)信号線を従装置から主装置へ移動されます。各データパケット後、主装置はSS(従装置選択)ピンをHighへ引き上げることによって従装置と同期を取ります。

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

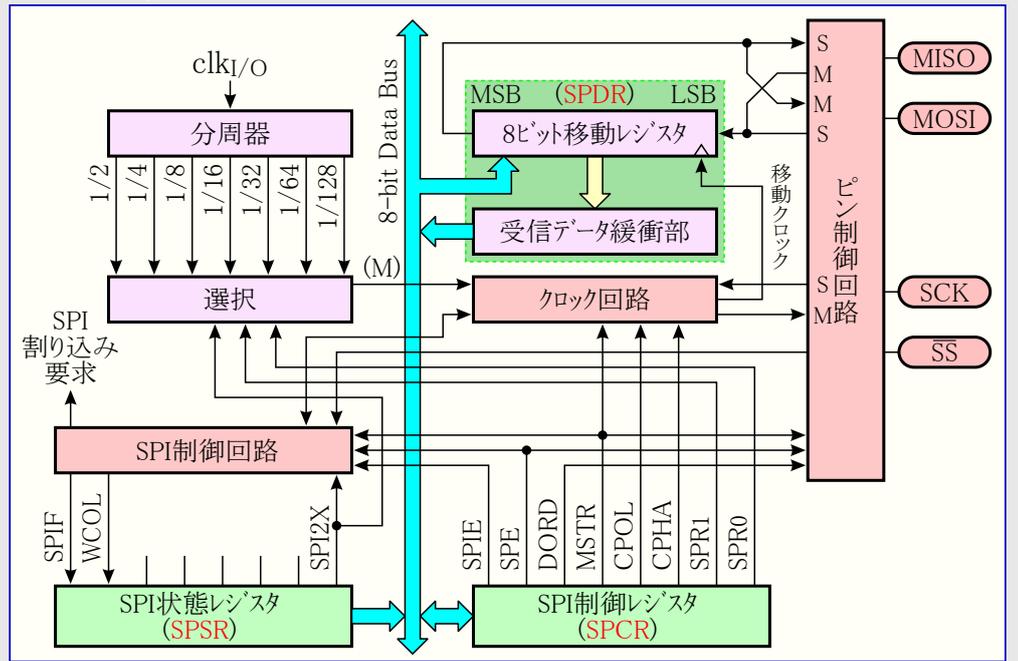
従装置として設定されると、SPIインターフェースはSSピンがHighに駆動される限り、MISOをHi-Zにした休止状態に留まります。この状態でプログラムはSPIデータレジスタ(SPDR)の内容を更新できますが、そのデータはSSピンがLowに駆動されるまでSCKピンでの到着クロックパルスによって移動出力されません。1バイトが完全に移動されてしまうと転送完了フラグ(SPIF)が設定(1)されます。SPCRでSPI割り込み許可(SPIE)ビットが設定(1)されていれば割り込みが要求されます。従装置は受信データを読む前にSPDR内へ送られるべき次のデータの配置を続けられます。最後の到着バイトはその後の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

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

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

SPIが許可されると、MOSI, MISO, SCK, SSピンのデータ方向は右表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「入出力ポート」記述を参照してください。

図23-1. SPI構成図



注: SPIピン配置については「ピン配置」記述と「入出力ポート」記述を参照してください。

図23-2. SPI 主装置/従装置の接続

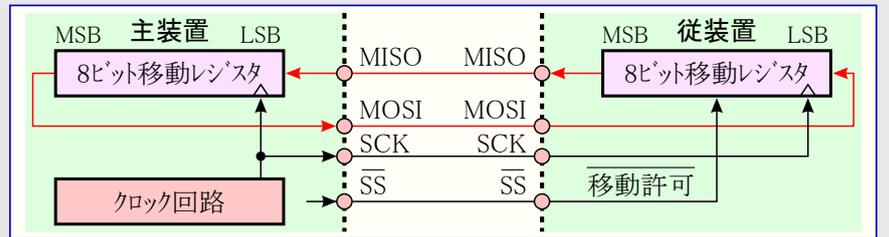


表23-1. SPIピン方向規定

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

注: SPIピンの方向定義方法については「入出力ポート」記述をご覧ください。

次のコード例は主装置としてSPIを初期化する方法と簡単な送信を実行する方法を示します。例でのDDR\_SPIはSPIピンを制御する実際のポート方向レジスタに置き換えられなければなりません。DD\_MOSI, DD\_MISO, DD\_SCKはこれらのピンに対する実際のポート方向ビットに置き換えられなければなりません。例えばMOSIがPB3ピンに配置されるなら、DD\_MOSIはDDB3、DDR\_SPIはDDRBに置き換えま

#### アセンブリ言語プログラム例

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

SPI_M_Tx:   OUT    SPDR, R16                          ;データ(R16)送信開始
SPI_M_Tx_W: IN     R17, SPSR                          ;SPI状態レジスタ値取得
            SBRS   R17, SPIF                          ;転送完了ならばスキップ
            RJMP  SPI_M_Tx_W                          ;転送完了まで待機

;

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

#### C言語プログラム例

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

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

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

#### アセンブリ言語プログラム例

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

SPI_S_Rx:   IN     R16, SPSR                        ;SPI状態レジスタ値取得
            SBRS   R16, SPIF                       ;受信(転送)完了ならばスキップ
            RJMP  SPI_S_Rx                         ;受信(転送)完了まで待機

;

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

#### C言語プログラム例

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

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

関連リンク [14頁の「コード例について」](#)

## 23.3. $\overline{SS}$ ピンの機能

### 23.3.1. 従装置動作

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

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

### 23.3.2. 主装置動作

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

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

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

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

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

## 23.4. データ転送形式

直列データに関してはSPI制御レジスタ(SPCR)のSCK位相(CPHA)とSCK極性(CPOL)制御ビットによって決定されるSCK位相と極性で4つの組み合わせがあります。データビットは安定のためデータ信号に対して十分な時間を保証するSCK信号の反対端で移動出力と(入力)ラッチが行われます。下表はCPOLとCPHAの設定を要約します。

表23-2. CPOL,CPHA機能動作

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

SPIデータ転送形式は以下の図で示されます。

図23-3. SPIデータ転送形式 (CPHA=0)

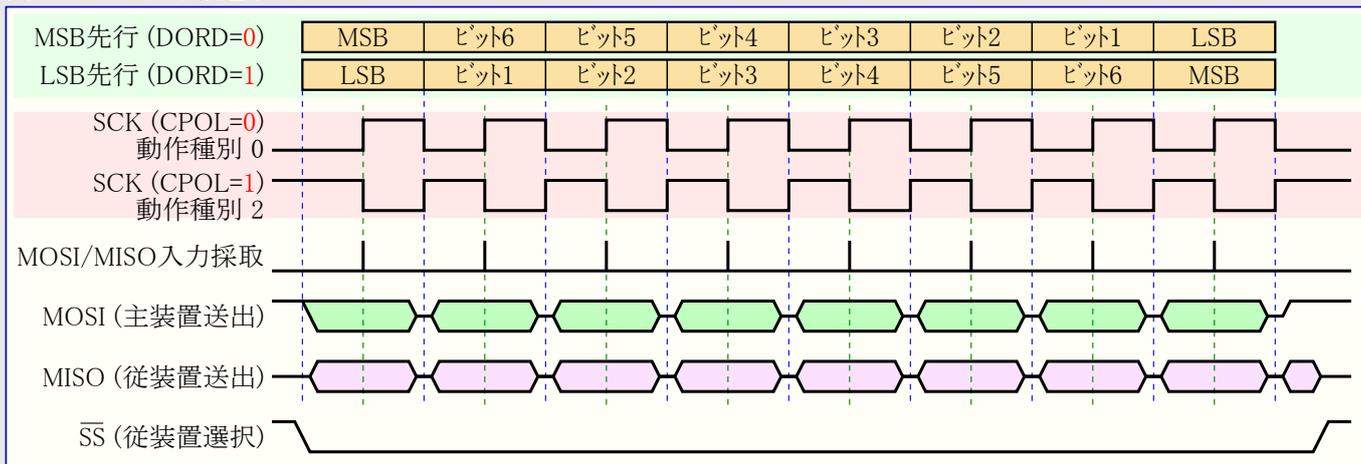
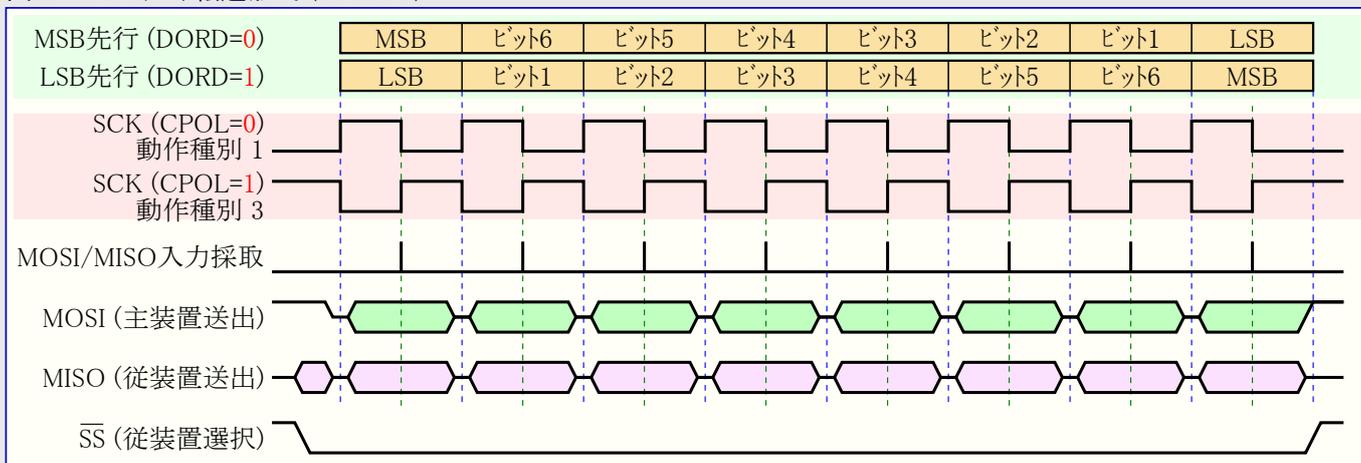


図23-4. SPIデータ転送形式 (CPHA=1)



## 23.5. SPI用レジスタ

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPCR

変位 : \$4C (\$2C)

リセット : \$00

特質 : I/O特定命令でデータ空間としてI/Oレジスタをアクセスする時の変位アドレスは\$2Cです。

ビット	7	6	5	4	3	2	1	0
	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1,0	
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	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(0)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 : データ順序選択 (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。SSが入力として設定され、MSTRが設定(1)の間にLowへ駆動されると、MSTRが解除(0)され、SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)になります。その後には使用者はSPI主装置動作を再び許可するためにMSTRを設定(1)しなければなりません。

#### ● ビット3 – CPOL : SCK極性選択 (Clock Polarity)

このビットが1を書かれると、アイドル時にSCKはHighです。CPOLが0を書かれると、アイドル時にSCKはLowです。例については図23-3と図23-4を参照してください。CPOL機能は右で要約されます。

表23-3. CPOL機能動作

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

#### ● ビット2 – CPHA : SCK位相選択 (Clock Phase)

このSCK位相選択(CPHA)ビットの設定はデータがSCKn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23-3と図23-4を参照してください。CPHA機能は右で要約されます。

表23-4. CPHA機能動作

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

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

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

表23-5. SCKn速度選択 (f<sub>OSC</sub>=CPUクロック周波数)

SPR1	0	0	1	1
SPR0	0	1	0	1
SPI2X	1	0	1	0
SCK周波数	f <sub>OSC</sub> /2	f <sub>OSC</sub> /4	f <sub>OSC</sub> /8	f <sub>OSC</sub> /16
	f <sub>OSC</sub> /32	f <sub>OSC</sub> /64	f <sub>OSC</sub> /128	

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPSR

変位 : \$4D (\$2D)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタをアクセスする時の変位アドレスは\$2Dです。

ビット	7	6	5	4	3	2	1	0
	SPIF	WCOL	-	-	-	-	-	SPI2X
アクセス種別	R	R	R	R	R	R	R	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

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

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

#### ● ビット0 – SPI2X : SPIIn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23-5参照)。これは最小SCK周期が2 CPUクロック周期であることを意味します。SPIが従装置として設定される時にSPIはf<sub>OSC</sub>(CPUクロック周波数)/4またはそれ以下の動作のみ保証されます。

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。プログラミングと照合については直列プログラミングをご覧ください。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPDR

変位 : \$4E (\$2E)

リセット : \$xx

特質 : I/O特定命令でI/Oレジスタをアクセスする時の変位アドレスは\$2Eです。

ビット	7	6	5	4	3	2	1	0
	SPID7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	不定	不定	不定	不定	不定	不定	不定	不定

#### ● ビット7~0 – SPID7~0 : SPIデータ (SPI Data)

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用さ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。

## 24. USART (Universal Synchronous Asynchronous Receiver Transceiver)

### 24.1. 特徴

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

### 24.2. 概要

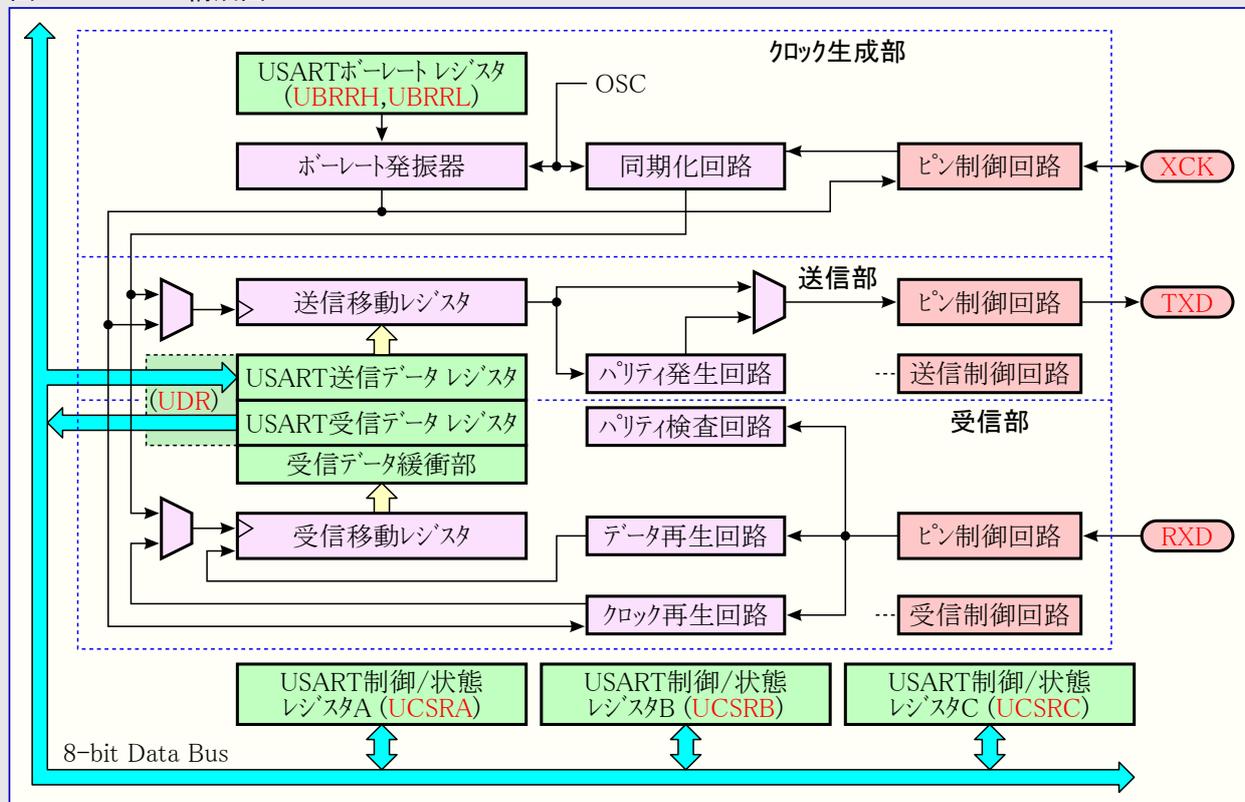
USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は高い柔軟性をもつ直列通信機能です。

USARTは主装置SPI動作での使用もできます。電力削減レジスタ(PRR)のUSART電力削減(PRUSART0)ビットはUSARTnを許可するために0を書かれなければなりません。PRUSART0はPRRに在ります。

### 24.3. 構成図

構成図に於いて、CPUがアクセス可能なレジスタとI/Oピンは赤文字(訳注:原文は太字)で示されます。構成図内の破線はUSARTの3つの主要部分、(上から)クロック生成部、送信部、受信部を分けます。制御レジスタは全部によって共用されます。クロック生成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(XCKn)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(UDRn)、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しに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(UDRn)、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。

図24-1. USART構成図



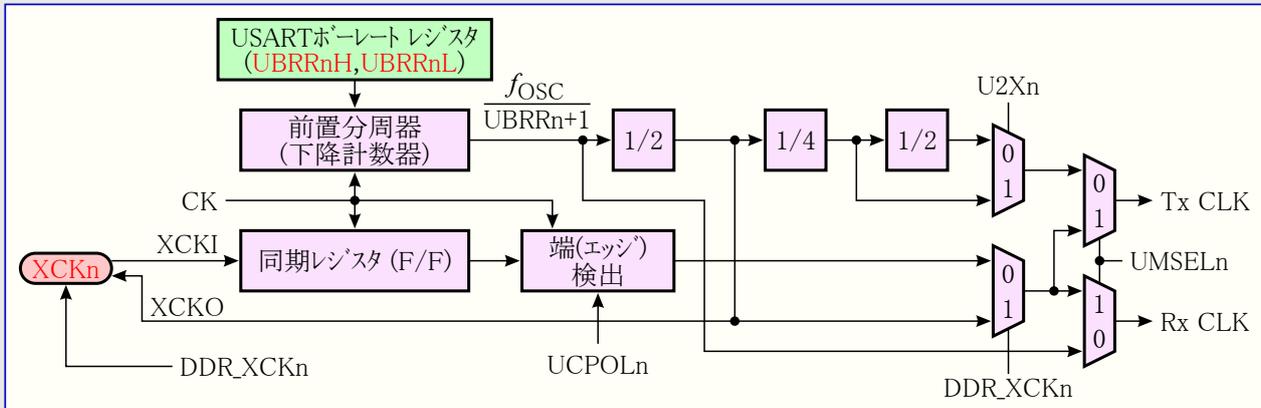
注: USARTピン配置については「ピン配置」記述と「入出力ポート」記述を参照してください。

## 24.4. 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(UCSRnC)のUSART動作種別選択(UMSEL0)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(UCSRnA)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(UMSEL0=1)を使うとき、XCKnピンに対する方向制御ビット(DDR\_XCKn)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XCKnピンは同期動作を使う時だけ活性(有効)です。

下はクロック生成論理回路の構成図を示します。

図24-2. クロック生成部構成図



### 信号説明

- Tx CLK : 送信クロック (内部信号)
- Rx CLK : 受信クロック (内部信号)
- XCKI : XCKnピンからの内部入力信号 (同期従装置時)
- XCKO : XCKnピンへの内部出力信号 (同期主装置時)
- DDR\_XCKn : 対応するポート方向レジスタの方向ビット
- f<sub>OSC</sub> : システムクロック周波数

### 24.4.1. ホ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。本項の記述は前項のクロック生成部構成図を参照してください。

USARTホーレートレジスタ(UBRRn(UBRRnH:UBRRnL))と下降カウンタは設定可能な前置分周器またはホーレート発振器として機能するように接続されます。システムクロック(f<sub>OSC</sub>)で走行する下降カウンタは0への下降計数時毎またはUBRRnLレジスタが書かれる時にUBRRn値で設定されます。1クロックはカウンタが0に達する毎に生成されます。このクロックがホーレート発振器出力(=f<sub>OSC</sub>/(UBRRn+1))です。送信部は動作種別に依存してホーレート発振器出力を2,8,16分周します。ホ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(UMSEL0)、倍速許可(U2X)、DDR\_XCKn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。

下表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ホーレート(bps)とUBRRn値の計算式を含みます。

表24-1. ホーレートレジスタ(UBRRn)値計算式

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

注: ホーレートは転送速度(ビット/1秒)で定義されます。

- BAUD : ホーレート (bps)
- UBRRn : UBRRnHとUBRRnLレジスタ値 (0~4095)
- f<sub>OSC</sub> : 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UBRRn値の例は「ホーレート設定例」で得られます。

## 24.4.2. 倍速動作 (U2X)

転送速度はUSART制御/状態レジスタA(UCSRnA)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ポーレート分周器の分周数を16から8に減らします。けれども、この場合、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)採取数しか使わず、従ってこの動作種別が使われる時にシステムクロックとポーレート設定の精度がより必要とされます。

送信部についての低下要因はありません。

## 24.4.3. 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。本項の記述は前項のクロック生成部構成図を参照します。

XCKnピンからの外部クロック入力是不確定レベル状態(メタステーブル)の機会を最少とするために同期化レジスタによって採取されます。同期化レジスタからの出力は送受信部で使い得るのに先立って端(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XCKnクロック周波数は次式によって制限されます。

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

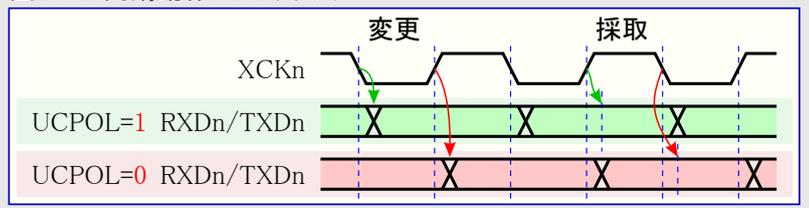
$f_{OSC}$ の値がシステムクロック元の安定度に依存します。従って周波数変動によるデータ消失の可能性を避けるため、いくらかの余裕分を追加することが推奨されます。

## 24.4.4. 同期クロック動作

同期動作が使われる(UMSEL=1)とき、XCKn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(TXDn)が変更される端と反対のXCKnクロック端でデータ入力(RXDn)が採取されることです。

USART制御/状態レジスタC(UCSRnC)のXCKn極性(UCPOL)ビットはデータ採取とデータ変更に対してどちらのクロック端が使われるのかを選びます。右のタイミング構成図で示されるようにUCPOLが0の時にデータはXCKn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XCKnの下降端で変更され、上昇端で採取されます。

図24-3. 同期動作XCKnタイミング



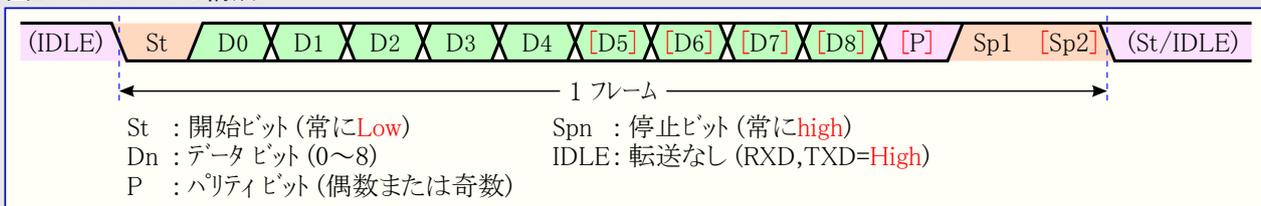
## 24.5. フレーム形式

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

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

フレームは最初に最下位データビット(LSB)、その後以最上位データビット(MSB)で終わる次データビットの、(合計で5~9ビットの)データビットが後続する開始ビットで始まります。許可したなら、パリティビットがデータビットの後、1つまたは2つの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。下図は組み合わせ可能なフレーム形式を図解します。[ ]付きビットは任意選択です。

図24-4. 1フレームの構成



USARTによって使われるフレーム形式は以下によって設定されます。

- フレーム内のデータビット数を選ぶUSART制御/状態レジスタBとC(UCSRnB,UCSRnC)のデータ長選択(UCSZ2,UCSZ1,0)ビット
- パリティを許可とパリティビットの種別(奇/偶)を設定するパリティ選択(UPM1,0)ビット
- 停止ビット数(1または2)を選ぶ停止ビット選択(USBS)ビット (受信部は第2停止ビットを無視します。)

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。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。

### 24.5.1. 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。

$$\begin{aligned} \text{偶数パリティビット} &= D_0 \text{ Ex-OR } D_1 \text{ Ex-OR } D_2 \text{ Ex-OR } \sim \text{Ex-OR } D_{n-1} \\ \text{奇数パリティビット} &= D_0 \text{ Ex-OR } D_1 \text{ Ex-OR } D_2 \text{ Ex-OR } \sim \text{Ex-OR } D_{n-1} \text{ Ex-OR } 1 \end{aligned} \quad n: \text{データビット長}$$

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。

### 24.6. 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\(0\)ビット](#)は解除(0)される(そして全割り込みが禁止される)べきです。

ボーレートまたはフレーム形式の変更を伴う再初期化を行う前には、レジスタが変更される期間中に進行中の送信がないことを確実にしてください。[USART制御/状態レジスタA\(UCSRnA\)](#)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\(UDRn\)](#)が書かれる)前にTXCフラグが解除(0)されなければなりません。

次の簡単なUSART初期化コード例が示すアセンブリ言語とC言語の関数は機能的に同じです。この例は固定フレーム形式でボーリングを使う(割り込み不許可)非同期動作と仮定します。ボーレート(UBRRn)値は関数の引数として与えられます。アセンブリ言語でのボーレート引数はR17:R16レジスタに格納されると仮定されます。

#### アセンブリ言語プログラム例

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

#### C言語プログラム例

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

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

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

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。

**関連リンク** [14頁の「コード例について」](#)

## 24.7. データ送信 – USART送信部

USART送信部はUSART制御/状態レジスタB(UCSRnB)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、TXDn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。何かの送信を行う前に一度はポーレート、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、XCKn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。

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

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

#### アセンブリ言語プログラム例

```
USART_Tx:  SBIS   UCSRnA, 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    UDRn, 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 ( !(UCSRnA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UDRn = data;                          /* データ送信(送信開始) */
}
```

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

関連リンク [14ページの「コード例について」](#)

### 24.7.2. 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(UDRn)に書かれるのに先立って第9ビットがUSART制御/状態レジスタB(UCSRnB)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。

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。

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

#### C言語プログラム例

```
void USART_Transmit(unsigned int data)
{
    while ( !(UCSRnA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UCSRnB &= ~(1<<TXB8);                /* TXB8を0に仮設定 */
    if (data & 0x0100) UCSRnB |= (1<<TXB8); /* 第9ビットをR17からTXB8へ複写 */
    UDRn = 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。UCSRnBの内容が静的(換言すると、UCSRnB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。

関連リンク [14ページの「コード例について」](#)

### 24.7.3. 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ない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(UCSRnA)に書くとき、常にこのビットに0を書いてください。

USART制御/状態レジスタB(UCSRnB)でデータレジスタ空き割り込み許可(UDRIE)ビットが1を書かれると、(全割り込みが許可されていれば)UDREフラグが設定(1)されている限り、USARTデータレジスタ空き割り込みが実行されます。UDREはUSARTデータレジスタ(UDRn)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UDRnに書くか、データレジスタ空き割り込みを禁止するかのどちらかを行わなければならない、さもなければ一旦割り込みルーチンを終了しても新しい割り込みが(継続的に)起こります。

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

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

### 24.7.4. 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。

### 24.7.5. 送信の禁止

USART制御/状態レジスタB(UCSRnB)のUSART送信許可(TXEN)ビットへの0書き込み時、送信部の禁止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部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TXDnピンの(標準ピン機能)を無効にしません。

## 24.8. データ受信 - USART受信部

USART受信部はUSART制御/状態レジスタB(UCSRnB)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、RXDnピンの標準ピン動作はUSARTによって無視され、受信部の直列入力としての機能を与えられます。何か直列受信が行われ得る前に一度はポーレート、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、XCKnピンのクロックは転送クロックとして使われます。

### 24.8.1. 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ポーレートまたはXCKnクロックで採取され、フレームの最初の停止ビットが受信されるまで受信移動レジスタ内へ移動されます。第2停止ビットは受信部によって無視されます。最初の停止ビットが受信されると(換言すると、受信移動レジスタに完全なフレームが存在すると)、この移動レジスタの内容は受信緩衝部内へ移されます。受信緩衝部はUSARTデータレジスタ(UDRn)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、UDRnから読むデータの上位ビットは0で覆われます。

次のコード例はUSART制御/状態レジスタA(UCSRnA)の受信完了(RXC)フラグのポーリングを基準とした簡単な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    UCSRnA, RXC          ;受信完了でスキップ
           RJMP    USART_Rx          ;受信完了待機
;
           IN     R16, UDRn          ;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 ( !(UCSRnA & (1<<RXC)) ); /* 受信完了待機 */
    return UDRn;                    /* 受信データ取得 */
}
```

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。

関連リンク [14頁の「コード例について」](#)

## 24.8.2. 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(UDRn)から下位バイトを読むのに先立って第9ビットがUSART制御/状態レジスタB(UCSRnB)の受信データビット8(RXB8)ビットから読まなければならない。この規則はフレーミング異常(FE)、オーバラン発生(DOR)、パリティ誤り(UPE)状態フラグにも適用されます。USART制御/状態レジスタA(UCSRnA)から状態情報を読み、その後UDRnからデータを読んでください。UDRn 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,UPEビット全てが切り替わります。

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

### アセンブリ言語プログラム例

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

### C言語プログラム例

```
unsigned int USART_Receive(void)
{
    unsigned char status, resh, resl; /* 一時変数定義 */
    while ( !(UCSRnA & (1<<RXC)) ); /* 受信完了待機 */
    status = UCSRnA; /* 状態フラグ取得 */
    resh = UCSRnB; /* 受信第9ビット取得 */
    resl = UDRn; /* 受信データ取得 */
    if ( status & ((1<<FE) | (1<<DOR) | (1<<UPE)) ) 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 */
    return ((resh<<8) | resl); /* 結果9ビットデータ取得/復帰 */
}
```

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

[関連リンク](#) 14頁の「コード例について」

## 24.8.3. 受信完了フラグと割り込み

USART受信部には受信部の状態を示す1つのフラグがあります。

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

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

#### 24.8.4. 受信異常フラグ

USART受信にはフレーミング異常(FE)、データオーバラン発生(DOR)、パリティ誤り(UPE)の3つの異常フラグがあります。これら全ては**USART制御/状態レジスタA(UCSRnA)**を読むことでアクセスできます。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(UDRn)**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、UCSRnAは受信緩衝部(UDRn)の前に読まなければならない。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、UCSRnA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らない。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(UCSRnC)**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、UCSRnAに書くとき、常にこのビットを**0**に設定してください。

**データオーバラン発生(DOR)フラグ**は受信緩衝部が一杯状態のためのデータ消失を示します。データオーバランは受信緩衝部(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UDRnから読んだフレームと次にUDRn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ます。将来のデバイスとの共通性のため、UCSRnAに書く時は常にこのビットに**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(UPE)フラグ**は受信緩衝部内の次の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、UPEフラグは常に**0**が読めます。将来のデバイスとの共通性のため、UCSRnAに書くとき、常にこのビットを**0**に設定してください。より多くの詳細については「**パリティビットの計算**」と下の「**パリティ検査器**」をご覧ください。

#### 24.8.5. パリティ検査器

パリティ検査器は**USART制御/状態レジスタC(UCSRnC)**の**パリティ種別上位ビット(UPM1)**が**1**を書かれる時に活性(有効)です。実行されるべきパリティ検査の形式(偶数または奇数)は**UCSRnC.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**USART制御/状態レジスタA(UCSRnA)**のパリティ誤り(UPE)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。

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

#### 24.8.6. 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(即ち、**USART制御/状態レジスタB(UCSRnB)**の**USART受信許可(RXEN)ビット**が**0**を書かれる)と、受信部はもはやRXDn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。

#### 24.8.7. 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(UCSRnA)**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(UDRn)**I/O位置を読んでください。

次のコード例は受信緩衝部の破棄方法を示します。

アセンブリ言語プログラム例		
USART_Flush:	SBIS UCSRnA, RXC	;未読データありでスキップ
	RET	;未読データなしで復帰
;		
	IN R16, UDRn	;データ受信
	RJMP USART_Flush	;未読データなしまで継続
C言語プログラム例		
void USART_Flush(void)		
{		
	unsigned char dummy;	/* 一時変数定義 */
	while ( UCSRnA & (1<<RXC) ) dummy=UDRn;	/* 未読データ読み捨て */
}		

関連リンク [14ページの「コード例について」](#)

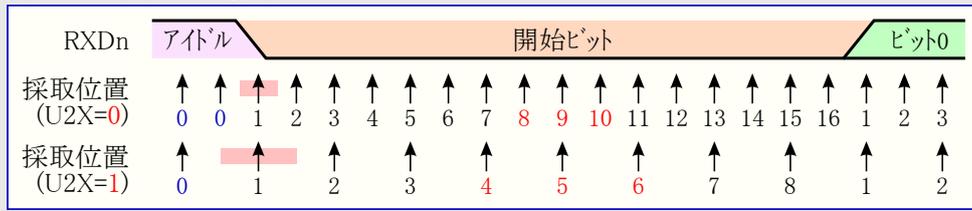
## 24.9. 非同期受信

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ホーレートクロックをRXDn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ホ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。

### 24.9.1. 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。下図は到着フレームの開始ビットの採取手順を図解します。採取速度は標準動作でホーレートの16倍、倍速動作で8倍です。赤帯(訳注:原文は水平矢印)は採取処理のための同期変量を図示します。倍速動作(UCSRnA.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RXDn信号がアイドル(即ち、通信の動きなし)の間に行われる採取です。

図24-5. 開始ビットの採取

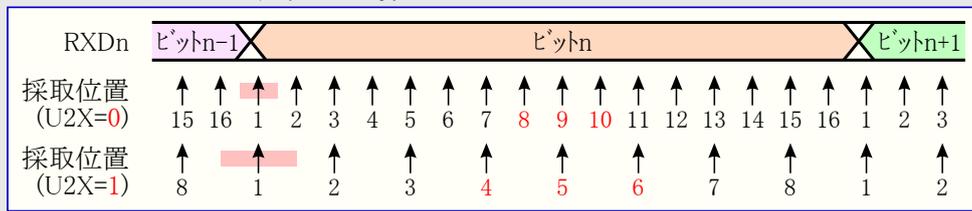


クロック再生論理回路がRXDn信号線でHigh(アイドル)からLow(開始)の遷移を検出すると、開始ビット検出手順が開始されます。図で示されるように採取1は最初のLow採取を意味します。その後クロック再生回路は有効な開始ビットが受信されるかを定めるために、標準動作に対して採取8,9,10、倍速動作に対して採取4,5,6(図の赤字(訳注:原文は箱枠内))で示された採取番号)を使います。これら3回で2回以上の採取が論理Highレベルの場合(多数決)、この開始ビットは尖頭雑音として捨てられ、受信部はRXDn上で次のHighからLowの遷移を探し始めます。けれども、有効な開始ビットが検出された場合、ク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

### 24.9.2. 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動作で16段、倍速動作で8段の順列回路を使います。下図はデータビットとパリティビットの採取を示します。各採取は再生部の各段に等しい番号を与えられます。

図24-6. データビットとパリティビットの採取



受信したビットの論理値の決定は受信したビット中央の3採取で論理値の多数決を取るによって行われます。2つまたは全3つの中央採取(赤字(訳注:原文は箱枠内))のそれらの採取番号によって記されるそれらが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つまたは全3つの採取が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RXDnピンの到着信号に対して低域通過濾波器(ローパスフィルタ)のように働きます。この再生手順はその後に完全なフレームが受信されるまで繰り返されます。これには最初の(第1)停止ビットを含みます。受信部はフレームの最初の停止ビットだけを使います。

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

図24-7. 停止ビットの採取と次の開始ビットの採取



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

新規フレームの開始ビットを示すHighからLowへの遷移は多数決に使った最後のビット後に実現できます。標準動作での最初のLowレベル採取は上図のA点で獲得され得ます。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。

### 24.9.3. 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ポ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ポーレートが類似した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(下の推奨をご覧ください)。

次式は到着データ速度と内部受信部ポーレート間の比率計算に役立ちます。

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

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<sub>F</sub> : 多数決に使う最初の採取番号 (標準速=8、倍速=4)  
 S<sub>M</sub> : 多数決に使う中心の採取番号 (標準速=9、倍速=5)  
 R<sub>slow</sub> : は受信側ポ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<sub>fast</sub> : は受信側ポ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。

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

表24-2. 標準速と倍速での受信部ポーレート推奨最大許容誤差

D	標準速動作 (U2X=0)				倍速動作 (U2X=1)			
	R <sub>slow</sub> (%)	R <sub>fast</sub> (%)	総合許容誤差(%)	推奨許容誤差(%)	R <sub>slow</sub> (%)	R <sub>fast</sub> (%)	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24-2.に標準速、表24-3.に倍速を記載していますが、比較が容易なように表24-2.として纏めました。

受信部ポーレートの推奨最大許容誤差は最大総合許容誤差を送信部と受信部で等分割するという仮定の下で作られました。

受信部ポ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(OSC)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水晶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。ポーレート発振器は欲したポ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UBRRn値が使えます。

## 24.10. 複数プロセッサ通信動作

USART制御/状態レジスタA(UCSRnA)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(UCSRnB)の受信第9(RXB8)ビットがアドレスとデータのフレームを識別するのに使われます。フレーム種別(最初の停止または第9)ビットが1の時にフレームはアドレスを含みます。フレーム種別ビットが0の時にそのフレームはデータフレームです。

複数プロセッサ通信動作は主MCUからのデータを多くの従MCUで受信することを可能にします。これはどのMCUがアドレス指定されるかを検出するため、最初にアドレスフレームを調べることによって行われます。特定の従MCUがアドレス指定されたなら、そのMCUは後続するデータフレームを通常のように受信し、一方その他の従MCUは他のアドレスフレームが受信されるまで受信したフレームを無視します。

### 24.10.1. MPCMn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。UCSRnB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(UCSRnA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UCSRnA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(UDRn)を読み、選ばれたかを判定します。選ばれた場合はUCSRnAのMPCMビットを解除(0)し、そうでなければ(非選択の場合は)MPCMビット設定を保ち、次のアドレスフレームを待ちます。
4. アドレス指定されたMCUは新規アドレスフレームが受信されるまで全データフレームを受信します。MPCMビットが未だ設定(1)されている他の従CPUはこのデータフレームを無視します。
5. 最後のデータフレームがアドレス指定されたMCUによって受信されると、アドレス指定されたMCUはMPCMビットを設定(1)し、主装置からの新規アドレスフレームを待ちます。以降、処理は手順2.からを繰り返します。

5~8ビットデータフレーム形式のどの使用も可能ですが、受信側が使うnとn+1ビットデータフレーム形式間を切り替えなければならないため非実用的です。これは送信部と受信部が同じデータビット長設定を使うため、全二重(フルデュプレックス)動作を困難にします。5~8ビットデータフレームが使われる場合、最初の停止ビットがフレーム種別を示すのに使われるので、送信部は2停止ビット使用(USBS=1)に設定されなければなりません。

MPCMビットを設定(1)または解除(0)するのに読み-修正-書き(リード 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。

(訳注) 本デバイスではUCSRnAのI/OアドレスがSBI,CBI命令適用範囲外なので上記記述は不適切です。但し、命令の組み合わせによる同様処理の場合に対して、上記注意の本意は適切(有効)です。

## 24.11. ホーレート設定例

標準的な水晶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ホーレートは、下表で一覧されるようにUBRRn設定を使うことによって生成できます。目的のホーレートに対して設定誤差1.5%(標準速:U2X=0)、1.0%(倍速:U2X=1)以上を赤字で示します(訳注:原書の本文は0.5%未満を太字)。より高い誤差率でも受け入れ可能ですが、特に長い直列フレームで誤差率が高いと、受信部は雑音耐性が低下します(「非同期での動作範囲」項もご覧ください)。誤差率は次式を使って計算されます。

$$\text{誤差率(\%)} = \left( \frac{\text{UBRRn設定ホーレート(最近似値)}}{\text{目的のホーレート}} - 1 \right) \times 100(\%)$$

表24-4. Xtal、ホーレート対UBRRnH,UBRRnL設定 (UBRR=UBRRnH:UBRRnL)

ホーレート (bps)	1MHz				1.8432MHz				2MHz				2.4576MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	51	0.2	103	0.2	95	0.0	191	0.0	103	0.2	207	0.2	127	0.0	255	0.0
2400	25	0.2	51	0.2	47	0.0	95	0.0	51	0.2	103	0.2	63	0.0	127	0.0
4800	12	0.2	25	0.2	23	0.0	47	0.0	25	0.2	51	0.2	31	0.0	63	0.0
9600	6	-7.0	12	0.2	11	0.0	23	0.0	12	0.2	25	0.2	15	0.0	31	0.0
14400	3	8.5	8	-3.5	7	0.0	15	0.0	8	-3.5	16	2.1	10	-3.0	20	1.6
19200	2	8.5	6	-7.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	0.0	15	0.0
28800	1	8.5	3	8.5	3	0.0	7	0.0	3	8.5	8	-3.5	4	6.7	10	-3.0
38400	1	-18.6	2	8.5	2	0.0	5	0.0	2	8.5	6	-7.0	3	0.0	7	0.0
57600	0	8.5	1	8.5	1	0.0	3	0.0	1	8.5	3	8.5	2	-11.1	4	6.7
76800	0	-18.6	1	-18.6	1	-25.0	2	0.0	1	-18.6	2	8.5	1	0.0	3	0.0
115.2k	-	-	0	8.5	0	0.0	1	0.0	0	8.5	1	8.5	0	33.3	2	-11.1
230.4k	-	-	0	-45.7	0	-50.0	0	0.0	0	-45.7	0	8.5	0	-33.3	0	33.3
250k	-	-	-	-	-	-	0	-7.8	-	-	0	0.0	-	-	0	22.9
最高速	62500		125k		115.2k		230.4k		125k		250k		153.6k		307.2k	

ホーレート (bps)	3.2768MHz				3.6864MHz				4MHz				4.608MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	170	-0.2	340	0.1	191	0.0	384	0.0	207	0.2	416	-0.1	239	0.0	479	0.0
2400	84	0.4	170	-0.2	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0
4800	42	-0.8	84	0.4	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0
9600	20	1.6	42	-0.8	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0
14400	13	1.6	27	1.6	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0
19200	10	-3.0	20	1.6	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0
28800	6	1.6	13	1.6	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0
38400	4	6.7	10	-3.0	5	0.0	11	0.0	6	-7.0	12	0.2	7	-6.3	14	0.0
57600	3	-11.1	6	1.6	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0
76800	2	-11.1	4	6.7	2	0.0	5	0.0	2	8.5	6	-7.0	3	-6.3	7	-6.3
115.2k	1	-11.1	3	-11.1	1	0.0	3	0.0	1	8.5	3	8.5	2	-16.7	4	0.0
230.4k	0	-11.1	1	-11.1	0	0.0	1	0.0	0	8.5	1	8.5	0	25.0	2	-16.7
250k	-	-	1	-18.1	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2
500k	-	-	0	-18.1	-	-	0	-7.8	-	-	0	0.0	0	-42.4	0	15.2
1M	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	-42.4
最高速	204.8k		409.6k		230.4k		460.8k		250k		500k		288k		576k	

注: 最高速はUBRR=0、誤差=0.0%です。以降の周波数は次頁へ続く。

(訳注) 原書では頁割された表毎に表番号24-4.~7.となっていますが、共通性から纏めて表24-4.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24-4 (続き). Xtal、ホーレート対UBRRnH,UBRRnL設定 (UBRR=UBRRnH:UBRRnL)

ホーレート (bps)	4.9152MHz				6.144MHz				7.3728MHz				8MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	255	0.0	511	0.0	319	0.0	639	0.0	383	0.0	767	0.0	416	-0.1	832	0.0
2400	127	0.0	255	0.0	159	0.0	319	0.0	191	0.0	383	0.0	207	0.2	416	-0.1
4800	63	0.0	127	0.0	79	0.0	159	0.0	95	0.0	191	0.0	103	0.2	207	0.2
9600	31	0.0	63	0.0	39	0.0	79	0.0	47	0.0	95	0.0	51	0.2	103	0.2
14400	20	1.6	42	-0.8	26	-1.2	52	0.6	31	0.0	63	0.0	34	-0.8	68	0.6
19200	15	0.0	31	0.0	19	0.0	39	0.0	23	0.0	47	0.0	25	0.2	51	0.2
28800	10	-3.0	20	1.6	12	2.6	26	-1.2	15	0.0	31	0.0	16	2.1	34	-0.8
38400	7	0.0	15	0.0	9	0.0	19	0.0	11	0.0	23	0.0	12	0.2	25	0.2
57600	4	6.7	10	-3.0	6	-4.8	12	2.6	7	0.0	15	0.0	8	-3.5	16	2.1
76800	3	0.0	7	0.0	4	0.0	9	0.0	5	0.0	11	0.0	6	-7.0	12	0.2
115.2k	2	-11.1	4	6.7	2	11.1	6	-4.8	3	0.0	7	0.0	3	8.5	8	-3.5
230.4k	1	33.3	2	-11.1	1	-16.7	2	11.1	1	0.0	3	0.0	1	8.5	3	8.5
250k	1	22.9	2	-18.1	1	-23.2	2	2.4	1	-7.8	3	0.0	1	0.0	3	0.0
500k	0	-38.6	0	22.9	0	-23.2	1	-23.2	0	-7.8	1	-7.8	0	0.0	1	0.0
1M	-	-	0	-38.6	-	-	0	-23.2	-	-	0	-7.8	-	-	0	0.0
最高速	307.2k		614.4k		384k		768k		460.8k		921.6k		500k		1M	

ホーレート (bps)	9.216MHz				9.8304MHz				10MHz				11.0592MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	479	0.0	959	0.0	511	0.0	1023	0.0	520	-0.0	1041	-0.0	575	0.0	1151	0.0
2400	239	0.0	479	0.0	255	0.0	511	0.0	259	0.2	520	-0.0	287	0.0	575	0.0
4800	119	0.0	239	0.0	127	0.0	255	0.0	129	0.2	259	0.2	143	0.0	287	0.0
9600	59	0.0	119	0.0	63	0.0	127	0.0	64	0.2	129	0.2	71	0.0	143	0.0
14400	39	0.0	79	0.0	42	-0.8	84	0.4	42	0.9	86	-0.2	47	0.0	95	0.0
19200	29	0.0	59	0.0	31	0.0	63	0.0	32	-1.4	64	0.2	35	0.0	71	0.0
28800	19	0.0	39	0.0	20	1.6	42	-0.8	21	-1.4	42	0.9	23	0.0	47	0.0
38400	14	0.0	29	0.0	15	0.0	31	0.0	15	1.8	32	-1.4	17	0.0	35	0.0
57600	9	0.0	19	0.0	10	-3.0	20	1.6	10	-1.4	21	-1.4	11	0.0	23	0.0
76800	7	6.7	14	0.0	7	0.0	15	0.0	7	1.8	15	1.8	8	0.0	17	0.0
115.2k	4	0.0	9	0.0	4	6.7	10	-3.0	4	8.5	10	-1.4	5	0.0	11	0.0
230.4k	2	-16.7	4	0.0	2	-11.1	4	6.7	2	-9.6	4	8.5	2	0.0	5	0.0
250k	1	15.2	4	-7.8	1	-23.2	4	-1.7	2	-16.7	4	0.0	2	-7.8	5	-7.8
500k	0	15.2	1	15.2	0	22.9	1	22.9	0	25.0	2	-16.7	0	38.2	2	-7.8
1M	0	-42.4	0	15.2	0	-38.6	0	22.9	0	-37.5	0	25.0	0	-30.9	0	38.2
最高速	576k		1.152M		614.4k		1.2288M		625k		1.25M		691.2k		1.3824M	

ホーレート (bps)	14.7456MHz				16MHz				18.4320MHz				20MHz			
	U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1		U2X=0		U2X=1	
	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)	UBRR	誤差(%)
1200	767	0.0	1533	0.0	832	0.0	1666	0.0	959	0.0	1919	0.0	1041	0.0	2082	0.0
2400	383	0.0	767	0.0	416	-0.1	832	0.0	479	0.0	959	0.0	520	0.0	1041	0.0
4800	191	0.0	383	0.0	207	0.2	416	-0.1	239	0.0	479	0.0	259	0.2	520	0.0
9600	95	0.0	191	0.0	103	0.2	207	0.2	119	0.0	239	0.0	129	0.2	259	0.2
14400	63	0.0	127	0.0	68	0.6	138	-0.1	79	0.0	159	0.0	86	-0.2	173	-0.2
19200	47	0.0	95	0.0	51	0.2	103	0.2	59	0.0	119	0.0	64	0.2	129	0.2
28800	31	0.0	63	0.0	34	-0.8	68	0.6	39	0.0	79	0.0	42	0.9	86	-0.2
38400	23	0.0	47	0.0	25	0.2	51	0.2	29	0.0	59	0.0	32	-1.4	64	0.2
57600	15	0.0	31	0.0	16	2.1	34	-0.8	19	0.0	39	0.0	21	-1.4	42	0.9
76800	11	0.0	23	0.0	12	0.2	25	0.2	14	0.0	29	0.0	15	1.7	32	-1.4
115.2k	7	0.0	15	0.0	8	-3.5	16	2.1	9	0.0	19	0.0	10	-1.4	21	-1.4
230.4k	3	0.0	7	0.0	3	8.5	8	-3.5	4	0.0	9	0.0	4	8.5	10	-1.4
250k	3	-7.8	6	5.3	3	0.0	7	0.0	4	-7.8	8	2.4	4	0.0	9	0.0
500k	1	-7.8	3	-7.8	1	0.0	3	0.0	1	15.2	4	-7.8	1	25.0	4	0.0
1M	0	-7.8	1	-7.8	0	0.0	1	0.0	0	15.2	1	15.2	0	25.0	1	25.0
最高速	921.6k		1.8432M		1M		2M		1.152M		2.304M		1.25M		2.5M	

注: 最高速はUBRR=0、誤差=0.0%です。

## 24.12. USART用レジスタ

### 24.12.1. UDRn – USARTnデータレジスタ (USART I/O Data Register n)

同じI/Oアドレスを共用するUSART受信データ緩衝レジスタとUSART送信データ緩衝レジスタはUSARTデータレジスタまたはUDRnとして引用しました。送信データ緩衝レジスタ(TXB)はUDRnレジスタ位置に書かれるデータの転送先です。UDRnレジスタ位置読み込みは受信データ緩衝レジスタ(RXB)の内容を返します。

5～7ビットデータでの上位未使用ビットは送信部によって無視され、受信部によって0に設定されます。

送信緩衝部はUCSRnA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UDRn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。その後データはTXDnピンで直列送信されます。

受信緩衝部は2段のFIFOから成ります。このFIFOは受信緩衝部がアクセスされる毎にその状態を切り替えます。この受信緩衝部の動きのため、この位置に読み-修正-書き(リード モデファイライト)命令(SBIとCBI)を使えません。ビット検査(SBICとSBIS)命令もFIFOの状態を換えるので、これらの命令を使う時は注意してください。(訳注:適用範囲外命令のため、この注意は不適切です。)

名称 : UDR0

変位 : \$C6

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	TXB/RXB7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット7~0 – TXB7~0/RXB7~0 : USART送受信データ (USART Transmit / Receive Data Buffer)

### 24.12.2. UCSRnA – USARTn制御/状態レジスタA (USART Control and Status Register n A)

名称 : UCSR0A

変位 : \$C0

リセット : \$20

特質 : -

ビット	7	6	5	4	3	2	1	0
	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM
アクセス種別	R	R/W	R	R	R	R	R/W	R/W
リセット値	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(UCSRnB)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(UDRn)に新規データが現存しない時に設定(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(UCSRnB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(UDRn)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(UCSRnBの送信データレジスタ空き割り込み許可(UDRIE)ビットをご覧ください)。送信部が準備できているのを示すため、リセット後のUDREは設定(1)です。

- ビット4 – FE : フレーミング異常フラグ (Framing Error)

受信緩衝部の次データが受信した時にフレーミング異常(換言すると、受信緩衝部で次データの最初の(第1)停止ビットがLow)だった場合、このビットが設定(1)されます。このフラグは受信緩衝部(UDRn)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。UCSRnAに書くとき、常にこのビットを0に設定してください。

このビットは主装置SPI動作(MSPIM)で保留されています。

- ビット3 – DOR : データオーバーラン発生フラグ (Data OverRun)

このビットはオーバーラン状態が検出されると設定(1)されます。受信緩衝部(2フレーム分)が一杯で、新規フレームが受信移動レジスタ内で待機中に新規開始ビットが検出されるとデータオーバーランが起きます。UCSRnAに書くとき、常にこのビットを0に設定してください。

このビットは主装置SPI動作(MSPIM)で保留されています。

● **ビット2 – UPE : パリティ誤りフラグ (USART 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(UDRn)が読まれるまで有効です。UCSRnAに書くとき、常にこのビットを0に設定してください。

このビットは主装置SPI動作(MSPIM)で保留されています。

● **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。

このビットは主装置SPI動作(MSPIM)で保留されています。

● **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受信部によって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。詳細については「[複数プロセッサ通信動作](#)」を参照してください。

このビットは主装置SPI動作(MSPIM)で保留されています。

**24.12.3. UCSRnB – USARTn制御/状態レジスタB (USART Control and Status Register n B)**

名称 : UCSR0B

変位 : \$C1

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
リセット値	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)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRnA)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、UCSRnA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)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRnA)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、UCSRnA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)**

このビットへの1書き込みはUSART制御/状態レジスタA(UCSRnA)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、UCSRnA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)**

このビットへの1書き込みはUSART受信(部)を許可します。受信部は許可されるとTXDnピンの標準ポート動作を無効にします。受信の禁止は受信緩衝部を破棄し、フレーミング異常(FE)、オーバラン(DOR)、パリティ誤り(UPE)のフラグを無効にします。

● **ビット3 – TXEN : 送信許可 (Transmitter Enable)**

このビットへの1書き込みはUSART送信(部)を許可します。送信部は許可されるとTXDnピンの標準ポート動作を無効にします。送信の禁止(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TXDnポート(の標準I/O機能)を無効にしません。

● **ビット2 – UCSZ2 : データビット長選択2 (Character Size)**

USART制御/状態レジスタC(UCSRnC)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。

このビットは主装置SPI動作(MSPIM)で保留されています。

● **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。UDRnから下位ビットを読む前に読んでください。

このビットは主装置SPI動作(MSPIM)で保留されています。

● **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。UDRnへ下位ビットを書く前に書いてください。

このビットは主装置SPI動作(MSPIM)で保留されています。

**24.12.4. UCSRnC – USARTn制御/状態レジスタC (USART Control and Status Register n C)**

名称 : UCSR0C

変位 : \$C2

リセット : \$06

特質 : -

ビット	7	6	5	4	3	2	1	0
	UMSEL1,0		UPM1,0		USBS	UCSZ1 UDORD	UCSZ0 UCPHA	UCPOL
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	1	1	0

● **ビット7,6 – UMSEL1,0 : USART動作選択 (USART Mode Select)**

このビットはUSART動作種別を選びます。

表24-8. USART動作選択

UMSEL1,0	動作種別
0 0	非同期動作
0 1	同期動作
1 0	(予約)
1 1	主装置SPI (MSPIM) (注)

**注:** UDORD、UCPHA、UCPOLは主装置SPI動作(MSPIM)許可での同じ書き込み操作で設定することができます。

● **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ります。受信部は到着データからパリティ値を生成し、UPM0設定と比較します。不一致が検出されると、USART制御/状態レジスタA(UCSRnA)でパリティ誤り(UPE)フラグが設定(1)されます。

これらのビットは主装置SPI動作(MSPIM)で保留されています。

表24-9. パリティ選択

UPM1,0	パリティ動作
0 0	禁止
0 0	(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)。

このビットは主装置SPI動作(MSPIM)で保留されています。

表24-10. 停止ビット選択

USBS	停止ビット数
0	1ビット
1	2ビット

● **ビット2 – UCSZ1/UDORD : データビット長選択 (Character Size) / データ順選択 (Data Order)**

UCSZ1:USART動作 : USART制御/状態レジスタB(UCSRnB)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24-11. データビット長選択

UCSZ2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
データビット数	5ビット	6ビット	7ビット	8ビット	(予約)	(予約)	(予約)	9ビット

UDORD: 主装置SPI動作 : 1に設定されるとデータ語のLSBが最初に転送されます。0に設定されるとデータ語のMSBが最初に転送されます。詳細については「USARTでのSPI動作」の「フレーム形式」を参照してください。

● **ビット1 – UCSZ0/UCPHA : データビット長選択 (Character Size) / クロック位相選択 (Clock Phase)**

UCSZ0:USART動作 : UCSZ1を参照してください。

UDORD: 主装置SPI動作 : クロック位相選択(UCPHA)ビットの設定はデータがXCKn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。詳細については「USARTでのSPI動作」の「SPIデータ形態とタイミング」を参照してください。

● ビット0 – UCPOL : クロック極性選択 (Clock Polarity)

USART動作 : このビットは同期動作に対してだけ使われます。非同期動作が使われるとき、このビットに0を書いてください。UCPOLビットは同期クロック(XCKn)、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24-12. XCKクロック極性選択

UCPOL	送信データ変更 (TXDnピン出力)	受信データ採取 (RXDnピン入力)
0	XCKnの上昇端	XCKnの下降端
1	XCKnの下降端	XCKnの上昇端

主装置SPI動作 : UCPOLビットはXCKnクロックの極性を設定します。UCPOLとクロック位相選択(UCPHA)ビットの組み合わせ設定がデータ転送のタイミングを決めます。詳細については「USARTでのSPI動作」の「SPIデータ形態とタイミング」を参照してください。

24.12.5. UBRRnL – USARTnボーレートレジスタ下位 (USART Baud Rate Register n Low)

名称 : UBRR0L  
変位 : \$C4  
リセット : \$00  
特質 : -

ビット	7	6	5	4	3	2	1	0
	UBRR7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7~0 – UBRR7~0 : ボーレート分周値下位 (USART Baud Rate low)

UBRRnHとUBRRnLを合わせたUBRRnはUSARTのボーレートを含む12ビットレジスタです。UBRRnHがUSARTボーレートの上位4ビットを含み、UBRRnL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。UBRRnL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。

24.12.6. UBRRnH – USARTnボーレートレジスタ上位 (USART Baud Rate Register n High)

名称 : UBRR0H  
変位 : \$C5  
リセット : \$00  
特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	UBRR11~8			
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット3~0 – UBRR11~8 : ボーレート分周値上位 (USART Baud Rate high)

USARTボーレートレジスタ下位(UBRRnL)を参照してください。

## 25. USARTSPI – USARTでのSPI動作 (USART in SPI Mode)

### 25.1. 特徴

- 全二重動作、3線同期データ転送
- 主装置動作
- 4つ全てのSPI転送形式(動作種別0,1,2,3)支援
- LSBまたはMSB先行データ転送(データ順設定)
- 順列動作(2重緩衝)
- 高分解能ボーレート発振器
- 高速動作( $f_{XCKmax}=f_{CK}/2$ )
- 柔軟な割り込み生成

### 25.2. 概要

USART(Universal Synchronous and Asynchronous Receiver and Transmitter)は主装置SPI互換動作に設定できます。USART動作選択(UMSEL1,0)ビットの11設定は主装置SPI(MSPIM)論理回路でのUSARTを許可します。この動作種別でのSPI主装置制御論理回路はUSART資源を直接制御します。これらの資源には送受信の移動レジスタと緩衝部、ボーレート発生器を含みます。パリティ発生/検査論理回路、データ/クロック再生論理回路、送受信制御論理回路は禁止されます。USART送受信論理回路は普通のSPI転送制御論理回路に置き換えられます。けれどもピン制御論理回路と割り込み生成論理回路は両動作種別で全く同じです。

I/Oレジスタ位置は両動作種別で同じです。けれどもMSPIM使用時、制御レジスタの一部の機能が変わります。

### 25.3. クロック生成

クロック生成論理回路は送受信部に対する基準クロックを生成します。USARTのMSPIM動作種別については内部クロック生成(即ち主装置動作)だけが支援されます。従って、USARTでMSPIMを正しく動作するにはXCKnピンに対するデータ方向レジスタ(DDR\_XCKn)が1(即ち出力)に設定されなければなりません。なるべくならDDR\_XCKnはUSARTでのMSPIMが許可(換言するとTXENとRXENが1に設定)される前に設定されるべきです。

MSPIM動作で使われる内部クロック生成はUSART同期主装置動作と同一です。右表は同期主装置動作用のボーレートやUBRRn設定を計算する式を含みます。

表25-1. ボーレートレジスタ(UBRRn)値計算式

動作種別	ボーレート計算式	UBRRn値計算式
同期主装置動作	$BAUD = \frac{f_{OSC}}{2 \times (UBRRn + 1)}$	$UBRRn = \frac{f_{OSC}}{2 \times BAUD} - 1$

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)

UBRRn : UBRRnHとUBRRnLレジスタ値 (0~4095)

$f_{OSC}$  : システム発振器クロック周波数

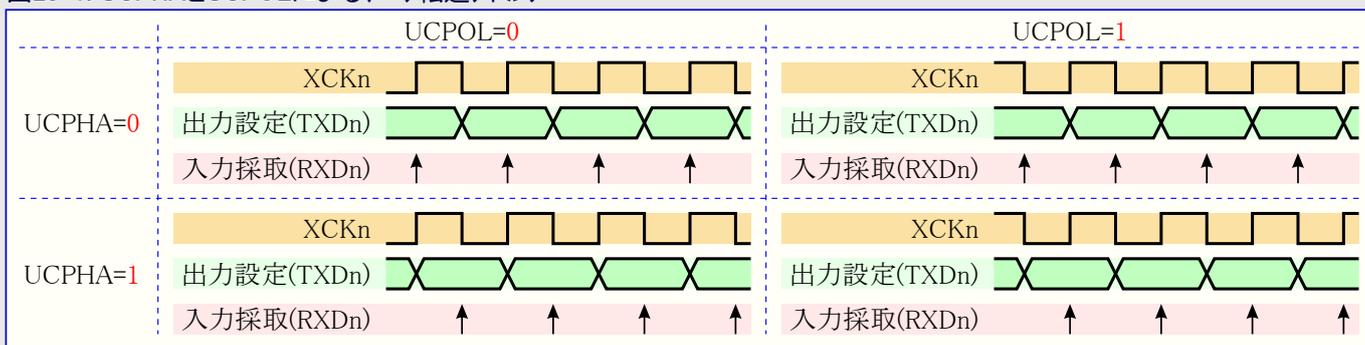
### 25.4. SPIデータ形態とタイミング

直列データに関してはクロック位相(UCPHA)とクロック極性(UCPOL)制御ビットによって決定されるXCKn(SCK)位相と極性で4つの組み合わせがあります。このデータ転送タイミング図は下図で示されます。データビットは安定のためにデータ信号に対して十分な時間を保証するXCKn信号の反対端で移動出力と(入力)ラッチが行われます。UCPOLとUCPHAの機能は下表で要約されます。これらビットのどの変更も送受信両方に対して実行中の通信を不正とすることに注意してください。

表25-2. UCPOL,UCPHA機能動作

SPI動作種別番号	UCPOL	UCPHA	XCKn(SCK)先行端	XCKn(SCK)後行端
0	0	0	入力採取/上昇端	出力設定/下降端
1	0	1	出力設定/上昇端	入力採取/下降端
2	1	0	入力採取/下降端	出力設定/上昇端
3	1	1	出力設定/下降端	入力採取/上昇端

図25-1. UCPHAとUCPOLによるデータ転送タイミング



## 25.5. フレーム形式

MSPIMの直列フレームは8データビット/1フレームに固定されます。USARTでのMSPIM動作は2つの有効なフレーム形式を持ちます。

- MSB先行 8ビット データ
- LSB先行 8ビット データ

フレームは最下位(LSB)または最上位(MSB)のデータビットで始まります。その後次のデータビットが最後の最上位(MSB)または最下位(LSB)データビットまで合計8ビット続きます。完全なフレームが送信されると、新規フレームが直ちに後続するか、または通信線がアイドル状態(high)に設定されるでしょう。

USART制御/状態レジスタC(UCSRnC)のデータ順選択(UDORD)ビットはUSARTでのMSPIMによって使われるフレーム形式を設定します。送受信部は同じ設定を使います。これらビットのどれかの設定変更が送受信部両方に対して実行中の通信を不正にすることに注意してください。

16ビット データ通信はUDRnに2バイト データを書くことによって達せられます。その後のUSART送信完了割り込み(TXC)はこの16ビット値が移動出力されてしまったことを示します。

### 25.5.1. USART MSPIM初期化

USARTでのMSPIM動作はどれかの通信が行われ得るのに先立って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、主装置動作操作(DDR\_XCKnの1)設定、フレーム形式設定、送受信部許可から成ります。送信部だけが独立して操作できます。割り込み駆動USART操作に関して初期化を行う時にステータスレジスタの全割り込み許可(I)ビットが解除(0)される(そして全割り込みが禁止される)べきです。

**注:** XCKn出力の初期化を直ちに保証するため、ボーレートレジスタ(UBRRn)は送信部が許可される時に0でなければなりません。標準動作のUSART操作と逆に、UBRRnは送信部が許可されて最初の送信が開始される前に、希望する値を書かれなければなりません。この初期化がリセット後直に行なわれるなら、UBRRnが0にリセットされているので、送信部許可前にUBRRnを0に設定することは必要ありません。

ボーレート、データ転送形式またはフレーム形式の変更を伴う再初期化を行う前に、レジスタが変更される期間中に進行中の送信がないことを確実にしてください。USART制御/状態レジスタA(UCSRnA)の送信完了(TXC)フラグは送信部の全転送完了検査に使い、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(UDRn)が書かれる)前にTXCフラグが解除(0)されなければならないことに注意してください。

次の簡単なUSART初期化コード例が示すアセンブリ言語とC言語の関数は機能的に同じです。この例はボーリングを使う(割り込み不許可)と仮定します。ボーレート(UBRRn)値は関数の引数として与えられます。アセンブリ言語でのボーレート引数はR17:R16レジスタに格納されると仮定されます。

#### アセンブリ言語プログラム例

```
USART_Init: CLR     R18                ;0値取得
             OUT     UBRRnH, R18      ;ボーレート設定(上位バイト)=0
             OUT     UBRRnL, R18      ;ボーレート設定(下位バイト)=0 (クロック停止)
             SBI     XCKn_DDR, XCKn   ;XCKnポートピン出力設定
             LDI     R18, (1<<UMSEL1) | (1<<UMSEL0) | (0<<UCPHA) | (0<<UCPOL)
             ;動作種別値を取得
             OUT     UCSRnC, R18      ;MSPI,データ種別0設定
             LDI     R18, (1<<RXEN) | (1<<TXEN)
             ;送受信許可値を取得
             OUT     UCSRnB, R18      ;送受信許可
             OUT     UBRRnH, R17      ;ボーレート設定(上位バイト)
             OUT     UBRRnL, R16      ;ボーレート設定(下位バイト)
             RET
```

#### C言語プログラム例

```
void USART_Init(unsigned int baud)
{
    UBRRn = 0;                /* ボーレート設定(クロック停止) */
    XCKn_DDR |= (1<<XCKn);    /* XCKnポートピン出力設定 */
    UCSRnC = (1<<UMSEL1) | (1<<UMSEL0) | (0<<UCPHA) | (0<<UCPOL);
    ;動作種別0設定 */
    UCSRnB = (1<<RXEN) | (1<<TXEN);
    ;送受信許可 */
    UBRRn = baud;            /* ボーレート設定 */
}
```

関連リンク [14頁の「コード例について」](#)

## 25.6. データ転送

USARTでのMSPI動作の使用は送信部が許可されること、換言するとUSART制御/状態レジスタB(UCSRnB)で送信許可(TXEN)ビットが1に設定される必要があります。送信部が許可されると、TXDnピンの標準ピン動作は無視され、送信部の直列出力としての機能を与えられます。受信部の許可は任意選択で、UCSRnBの受信許可(RXEN)ビットの設定(=1)によって行なわれます。受信部が許可されると、RXDnピンの標準ピン動作は無視され、受信部の直列入力としての機能を与えられます。XCKnは両方の場合で転送クロックとして使われます。

初期化後、USARTはデータ転送を行なう準備が整います。データ転送はUSARTデータレジスタ(UDRn)I/O位置に書くことによって開始されます。送信部が転送クロックを制御するため、これは送受信データ両方の状態についてです。UDRnに書かれた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送信緩衝部から移動レジスタへ移されます。

**注:** 入力緩衝部に於いて送信されたデータバイト数との同期を保つために、送信された各バイトに対して一度、UDRnが読まなければならない。入力緩衝操作は通常のUSART動作と同じで、換言すると、オーバーランが起きると、緩衝部内の先頭ではなく最後に受信したデータが失われます。これは第1バイト、第2、第3、第4バイトの順で4バイトのデータが送信され、全ての転送が完了される前にDDRnが読めない場合、第1バイトではなく、受信されるべき第3バイトが失われます。

次のコード例はUSART制御/状態レジスタA(UCSRnA)の送信データレジスタ空き(UDRE)フラグのポーリングに基いた、簡単なUSARTでのMSPIM転送関数を示します。この関数が使われ得る前にUSARTが初期化されなければならない。アセンブリ言語での送るべきデータはR16レジスタに格納されると仮定され、受信したデータは関数復帰後、同じR16レジスタで利用できます。

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。その後、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待ちます。

### アセンブリ言語プログラム例

```
USART_MSPI: SBIS    UCSRnA, UDRE          ;送信緩衝部空きでスキップ
             RJMP    USART_MSPI         ;送信緩衝部空き待機
;
             OUT     UDRn, R16          ;データ送信(送信開始)
USART_SPIR: SBIS    UCSRnA, RXC          ;受信完了でスキップ
             RJMP    USART_SPIR        ;受信完了待機
;
             IN     R16, UDRn          ;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

```
void USART_MSPI_M_Transfer(unsigned int data)
{
    while ( !(UCSRnA & (1<<UDRE)) );    /* 送信緩衝部空き待機 */
    UDRn = data;                          /* データ送信(送信開始) */
    while ( !(UCSRnA & (1<<RXC)) );    /* 受信完了待機 */
    return UDRn;                          /* 受信データ取得 */
}
```

関連リンク [14頁の「コード例について」](#)

### 25.6.1. 送受信フラグと割り込み

USARTのMSPI動作での受信完了(RXC)、送信完了(TXC)、送信データレジスタ空き(UDRE)フラグと対応する割り込みは通常のUSART操作と機能的に同一です。けれども受信異常状態フラグ(FE,DOR,UPE)は使えず、常に0として読みます。

### 25.6.2. 送受信の禁止

USARTのMSPI動作での送受信部の禁止は通常のUSART操作と機能的に同一です。

## 25.7. USARTでのMSPIMとSPIの比較

USARTでのMSPI動作は次に関してSPIと完全な互換性があります。

- 主装置動作タイミング図
- クロック極性選択(UCPOL)ビットはSPIのSCK極性選択(CPOL)ビットと機能的に同じです。
- クロック位相選択(UCPHA)ビットはSPIのSCK位相選択(CPHA)ビットと機能的に同じです。
- データ順選択(UDORD)ビットはSPIのデータ順選択(DORD)ビットと機能的に同じです。

けれどもUSARTでのMSPI動作がUSART資源を再使用するため、USARTでのMSPI動作はSPIと比較して多少異なります。加えて制御レジスタビットの差異、主装置動作だけがUSARTでのMSPI動作によって支援されること、2つの部間で異なる次の特質があります。

- USARTでのMSPI動作は送信部の(2重)緩衝部を含みます。SPIは緩衝部を持ちません。
- USARTのMSPI動作での受信部は追加の緩衝段を含みます。
- SPIの上書き(WCOL)ビットはUSARTでのMSPI動作に含まれません。
- SPIの倍速許可(SPI2X)ビットは含まれません。しかし、対応するボーレートレジスタ(UBRRn)設定によって同じ効果が達せられます。
- 割り込みタイミングに互換性はありません。
- USARTでのMSPI動作が主装置動作だけのため、ピン制御が異なります。

USARTのMSPI動作とSPIでのピンは右表で示されます。

表25-3. USARTでのMSPIMとSPIのピン比較

USART MSPIM	SPI	備考
TXDn	MOSI	主装置出力のみ
RXDn	MISO	主装置入力のみ
XCKn	SCK	(機能的に同一)
該当なし	$\overline{SS}$	USARTでのMSPIMで未支援

## 25.8. MSPIMでのUSART用レジスタ

USART用レジスタ記述を参照してください。

関連リンク [140頁の「USART用レジスタ」](#)

## 26. TWI – 2線直列インターフェース (2-wire Serial Interface)

### 26.1. 特徴

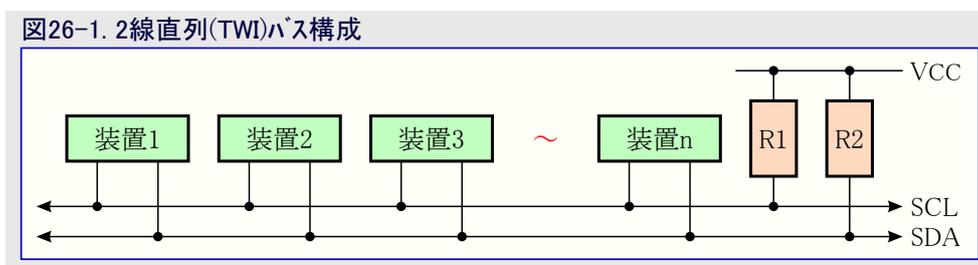
- 2本のバス信号線のみ必要な、単純ながら強力な柔軟な通信インターフェース
- 主装置動作と従装置動作の両方を支援
- 送信装置または受信装置として動作可能
- 7ビットのアドレス空間が128までの異なる従装置アドレスを許容
- 複数主装置の調停支援
- 400kHzまでのデータ転送速度
- 上昇/下降(スレーブ)制限された出力駆動回路
- バス信号線のスパイクを排除する雑音除去回路
- 一斉呼び出しを含む完全に設定変更可能な従装置アドレスの支援
- AVRが休止形態の時のアドレス認証(一致)起動
- Philips I<sup>2</sup>C規約互換

電力削減レジスタのTWI電力削減(PRR.PRTWI)ビットは2線直列インターフェースを許可するために0を書かれなければなりません。

関連リンク 35頁の「PM – 電力管理と休止形態」

### 26.2. 2線直列インターフェースバスの定義

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



#### 26.2.1. TWI用語定義

次の定義は本章で度々使われます。

表26-1. TWI用語定義

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

#### 26.2.2. 電気的な相互接続

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

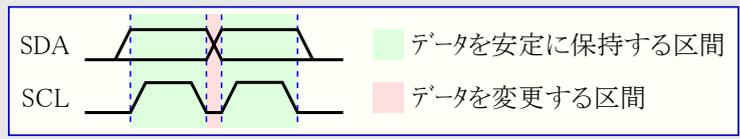
このバスに接続できる装置数は7ビットの従装置アドレス空間と400pFのバス容量制限によってのみ制限されます。TWIの電気的特性の詳細仕様は「2線直列インターフェース特性」で与えられます。そこで与えられる2組の異なる仕様は、1つがバス速度100kHz以下に関するもので、もう1つはバス速度400kHzまでに関して有効です。

## 26.3. データ転送とフレーム形式

### 26.3.1. ビット転送

TWIバスに転送される各データビットはクロック信号線のハルスを伴います。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。

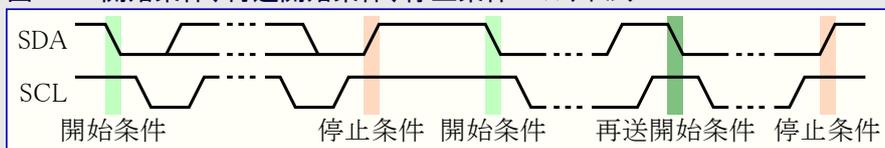
図26-2. データの有効性



### 26.3.2. 開始条件と停止条件

主装置がデータ転送の開始と終了を行います。転送は主装置がバスに**開始条件**を起こすと開始され、主装置が**停止条件**を起こすと終了されます。**開始条件**と**停止条件**間はバスが使用中と考えられ、他の主装置はバスの制御獲得を試みるべきではありません。**開始条件**と**停止条件**間で新規**開始条件**が起こされると特別な状態が起きます。これは**再送開始条件**として引用され、主装置がバスの制御を手放さずに新規転送を始めたい時に使われます。**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。

図26-3. 開始条件、再送開始条件、停止条件 バス タイミング



### 26.3.3. アドレス パケット形式

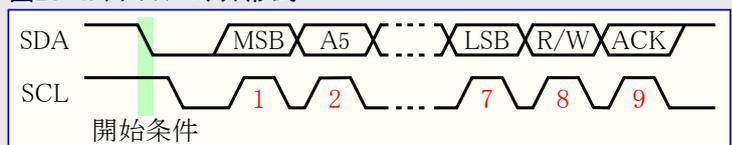
TWIバスに送信した全てのアドレス パケットは7ビットのアドレスビット、1ビットの方向(Read/Write)制御ビット、1ビットの応答ビットから成る9ビットです。方向(R/W)ビットが設定(1)されると読み出し操作が実行され、さもなければ書き込み操作が実行されるべきです。従装置がアドレス指定されたことを認証すると、9番目のSCL(ACK)周期でSDAをLowへ引くことによって確認応答すべきです。アドレス指定された従装置が忙しいまたはその他の理由で主装置の要求を扱えない場合、**確認応答(ACK)**クロック周期でSDA信号線をHighのままにすべきです。主装置はその後に**停止条件**または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

アドレスビットの最上位ビット(MSB)が最初に送信されます。従装置アドレスは設計者によって自由に割り当てられますが、'0000 000'のアドレス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くW(方向が書き込み)ビットがバスに送信されると、一斉呼び出しに**応答**する設定の全ての従装置は、ACK周期でSDA信号線をLowに引き込みます。そして後続のデータ パケットは一斉呼び出しに**確認応答**した全従装置によって受信されます。一斉呼び出しアドレスに続くR(方向が読み出し)ビットの送信は、従装置それぞれが異なるデータの送信を始めた場合の衝突の原因となるので意味がないことに注意してください。

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

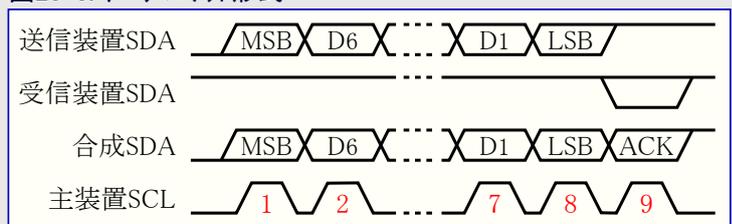
図26-4. アドレス パケット形式



### 26.3.4. データ パケット形式

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

図26-5. データ パケット形式

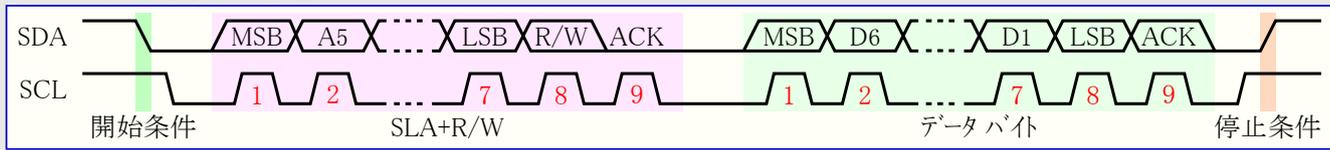


### 26.3.5. 転送内でのアドレスパケットとデータパケットの組み合わせ

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

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

図26-6. 代表的なデータ転送



### 26.4. 複数主装置バスシステムの調停と同期

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

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

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

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

図26-7. 複数主装置間でのSCL同期化

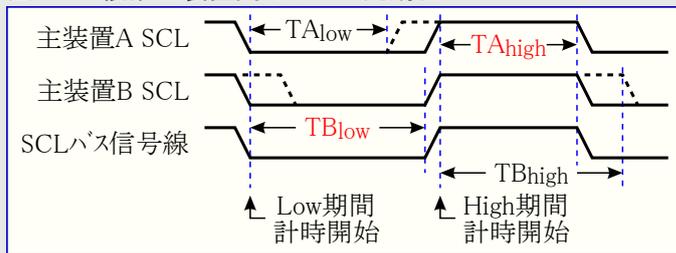
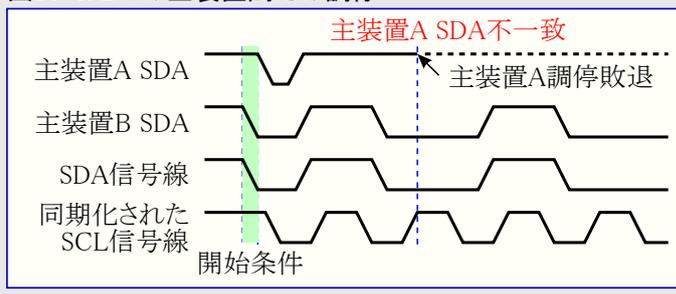


図26-8. 2つの主装置間での調停



調停が次の状態間で許されないことに注意してください。

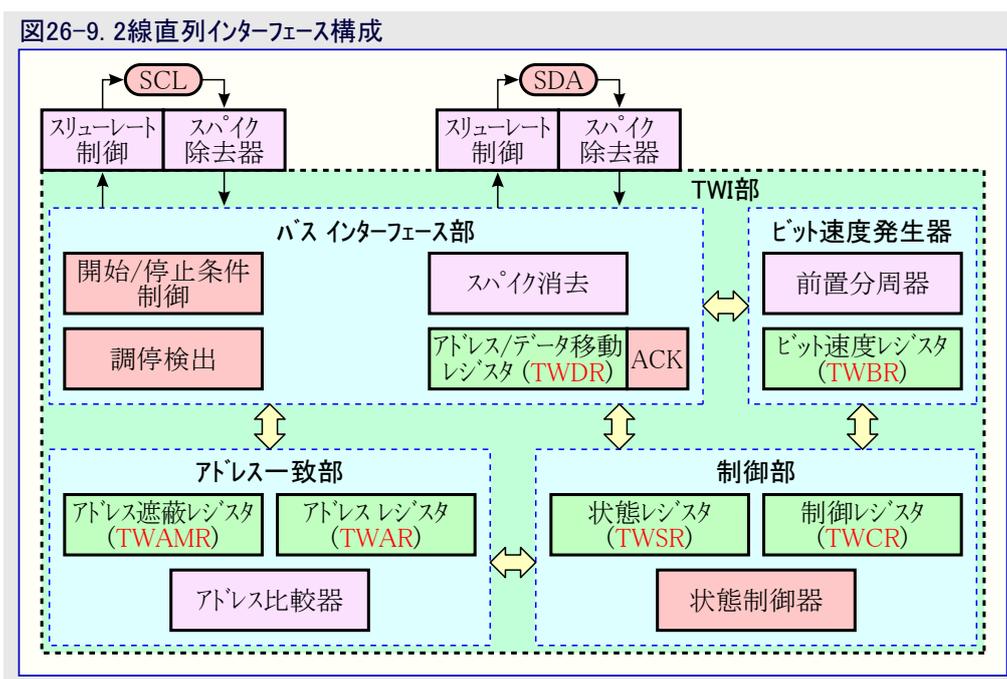
- 再送開始条件とデータビット間
- 停止条件とデータビット間
- 再送開始条件と停止条件間

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

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

## 26.5. TWI部の概要

下図で示されるように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。



### 26.5.1. SCLとSDAピン

これらのピンはAVR TWIをMCUシステムのその他とインターフェースします。出力駆動部はTWI仕様に適合させるためのスローレート(上昇/下降)制限器を含みます。入力段は50nsよりも短いスパイクを除去するスパイク除去部を含みます。「[入出力ポート](#)」章で説明したようにAVRパッドの内部プルアップはSCLとSDAの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。

### 26.5.2. ビット速度発生器

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

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

TWBR : TWIビット速度レジスタ値  
前置分周値 : TWI状態レジスタの前置分周器選択(TWPS)をご覧ください。

**注:** プルアップ抵抗値はSCL周波数とバス信号線の容量性負荷に応じて選ばれるべきです。適切なプルアップ抵抗の値については「[2線直列インターフェース特性](#)」をご覧ください。

関連リンク [217頁の「2線直列インターフェース特性」](#)

### 26.5.3. バス インターフェース部

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

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

TWIが主装置として送信を始めると、調停検出ハードウェアは調停が進行中かを決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。その後正しい処置が行われ、適切な状態符号が生成されます。

#### 26.5.4. アドレス一致部

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

#### 26.5.5. 制御部

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

TWI割り込み要求フラグ(TWINT)は次の場合に設定(1)されます。

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

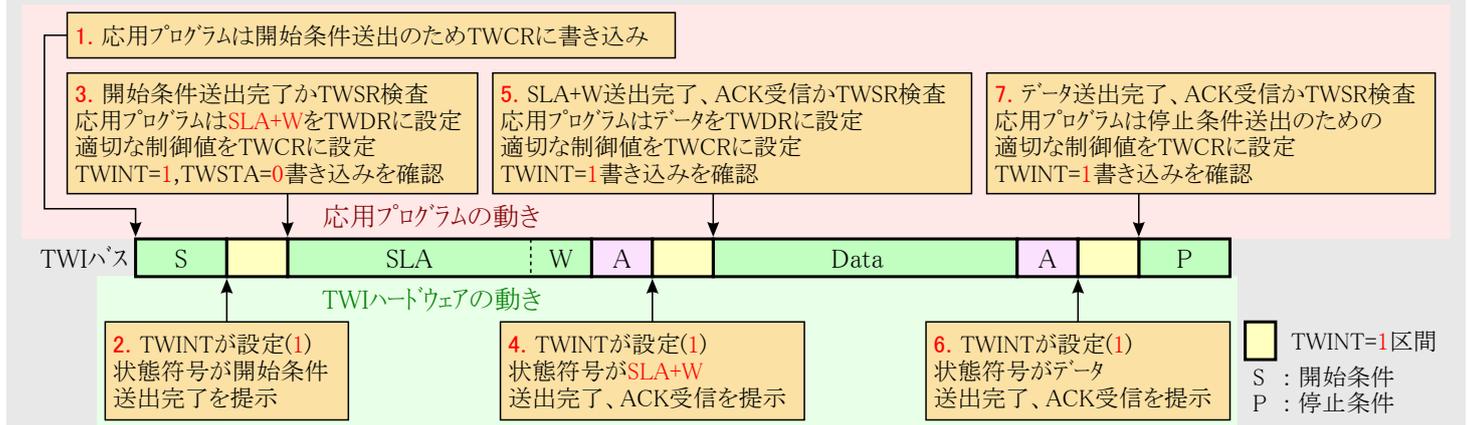
## 26.6. TWIの使用法

AVR TWIはハバ志向で割り込みが基本です。割り込みはハバの受信や**開始条件**の送出のような全てのバスの事象後に起こります。TWIは割り込みが基本のため、応用ソフトウェアはTWIハバ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割り込み要求フラグ(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バ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フラグが設定(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バ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バス周期で何を行うべきかを決定できます。

下図は応用(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を図解します。この例では主装置が単一データハバを従装置に送信しようとしています。より詳細な説明は本項の後に続きます。簡単なコード例が**以降の表**で示されます。

図26-10. 代表的な送信での応用プログラムとTWIのインターフェース



1. TWI送信の最初の段階は**開始条件**を送出することです。これはTWIハードウェアに**開始条件**送出を命じる特別な値をTWCR内に書くことによって行います。どんな値を書くかは後で記述されます。けれども、書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みは、このフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。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**開始条件**が正常に送出されてしまったことを示す状態符号に更新されます。
3. 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が他を示している場合、応用ソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、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がアドレスとデータの両方に使われることを思い出してください。TWDRが望む**SLA+W**に設定されてしまった後、TWDRにある**SLA+W**の送信をTWIハードウェアへ命じる特別な値がTWCRに書かれなければなりません。どんな値を書くかは後で記述されます。けれども書かれる値でTWINTビットが設定(1)されることが重要です。TWINTへの1書き込みがこのフラグを解除(0)します。TWCRでTWINTビットが設定(1)されている限り、TWIはどんな動作も始めません。応用(ソフトウェア)がTWINTを解除(0)した後、TWIは直ちにアドレスパケットの送信を始めます。
4. アドレスパケットが送信されてしまうと、TWCRでTWINTフラグが設定(1)され、TWSRはアドレスパケットが正常に送信されたことを示す状態符号に更新されます。この状態符号は従装置がパケットに**応答**したかどうかも反映します。
5. 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**ビット値であることを確認するためにTWSRの値を直ぐに検査すべきです。TWSRが他を示している場合、応用ソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、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**ビットの値であることを確認するためにTWSRの値を直ぐに検査すべきです。TWSRが他を示している場合、応用ソフトウェアは異常ルーチン呼び出すような或る特別な動きを講じるかもしれません。期待した状態符号だと仮定すると、応用(ソフトウェア)は**停止条件**の送出をTWIハードウェアへ命じる特別な値をTWCRに書か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**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)されたなら、使用者は次のTWIバス周期に関連した値で(必要な)全てのTWIレジスタを更新しなければなりません。例で示されるように**TWDR**は次のTWIバス周期で送信されるべき値を設定されなければなりません。
- (必要な)全てのTWIレジスタを更新し、その他保留中の応用ソフトウェアの処理が完了してしまった後に**TWCR**が書かれます。TWCR書き込み時、TWINTビットが設定(1)されるべきです。TWINTへの1書き込みはこのフラグを解除(0)します。TWCR設定によってどの動作が指定されても、TWIはその(TWINT=0)後に実行を始めます。

下表はアセンブリ言語とC言語の実装例を一覧にします。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.	<pre>LDI R16, (1&lt;&lt;TWINT)   (1&lt;&lt;TWSTA)       (1&lt;&lt;TWEN) OUT TWCR, R16</pre>	<pre>TWCR = (1&lt;&lt;TWINT)   (1&lt;&lt;TWSTA)       (1&lt;&lt;TWEN);</pre>	; 開始条件送出
2.	<pre>WAIT1: IN R16, TWCR         SBRS R16, TWINT         RJMP WAIT1</pre>	<pre>while (!(TWCR &amp; (1&lt;&lt;TWINT)));</pre>	; TWINT=1まで待機 ; (開始条件送出完了待機)
3.	<pre>IN R16, TWSR ANDI R16, \$F8 CPI R16, START BRNE ERROR</pre>	<pre>if ((TWSR &amp; 0xF8) != START)     ERROR();</pre>	; 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
	<pre>LDI R16, SLA_W OUT TWDR, R16 LDI R16, (1&lt;&lt;TWINT)   (1&lt;&lt;TWEN) OUT TWCR, R16</pre>	<pre>TWDR = SLA_W; TWCR = (1&lt;&lt;TWINT)   (1&lt;&lt;TWEN);</pre>	; 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.	<pre>WAIT2: IN R16, TWCR         SBRS R16, TWINT         RJMP WAIT2</pre>	<pre>while (!(TWCR &amp; (1&lt;&lt;TWINT)));</pre>	; TWINT=1まで待機 ; (SLA+W送出完了と ; ACK/NACK受信完了待機)
5.	<pre>IN R16, TWSR ANDI R16, \$F8 CPI R16, MT_SLA_ACK BRNE ERROR</pre>	<pre>if ((TWSR &amp; 0xF8) != MT_SLA_ACK)     ERROR();</pre>	; 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
	<pre>LDI R16, DATA OUT TWDR, R16 LDI R16, (1&lt;&lt;TWINT)   (1&lt;&lt;TWEN) OUT TWCR, R16</pre>	<pre>TWDR = DATA; TWCR = (1&lt;&lt;TWINT)   (1&lt;&lt;TWEN);</pre>	; 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.	<pre>WAIT3: IN R16, TWCR         SBRS R16, TWINT         RJMP WAIT3</pre>	<pre>while (!(TWCR &amp; (1&lt;&lt;TWINT)));</pre>	; TWINT=1まで待機 ; (データ送出完了と ; ACK/NACK受信完了待機)
7.	<pre>IN R16, TWSR ANDI R16, \$F8 CPI R16, MT_DATA_ACK BRNE ERROR</pre>	<pre>if ((TWSR &amp; 0xF8) != MT_DATA_ACK)     ERROR();</pre>	; 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
	<pre>LDI R16, (1&lt;&lt;TWINT)   (1&lt;&lt;TWSTO)       (1&lt;&lt;TWEN) OUT TWCR, R16</pre>	<pre>TWCR = (1&lt;&lt;TWINT)   (1&lt;&lt;TWSTO)       (1&lt;&lt;TWEN);</pre>	; 停止条件送出

## 26.7. 転送種別

TWIは以下の4つの主な動作種別の1つで動けます。

- 送信主装置(MT)
- 受信主装置(MR)
- 送信従装置(ST)
- 受信従装置(SR)

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

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

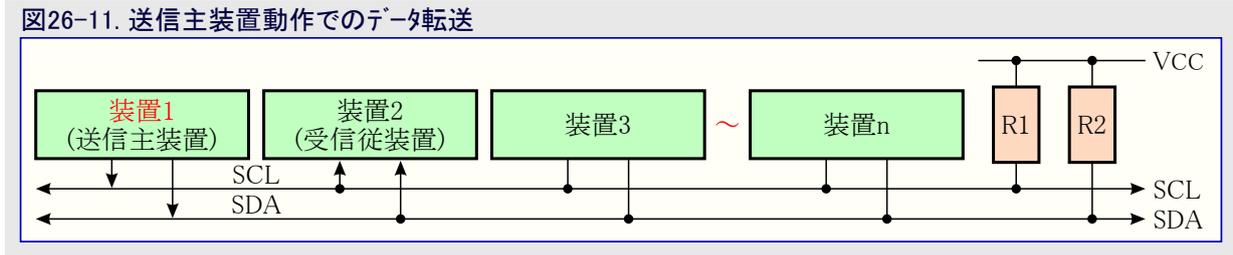
S	開始(START)条件
Rs	再送開始(REPEATED START)条件
R	読み出し指定ビット (SDA=High)
W	書き込み指定ビット (SDA=Low)
A	確認応答(ACK)ビット (SDA=Low)
$\bar{A}$	非確認応答(NACK)ビット (SDA=High)
Data	8ビットデータバイト
P	停止(STOP)条件
SLA	従装置アドレス

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

TWI割り込み要求フラグ(TWINT)が設定(1)される時のTWI状態レジスタ(TWSR)の状態符号は適切なソフトウェア動作を決めるのに使われます。各状態符号に対する必要なソフトウェア動作や後続の直列転送の詳細は各動作の状態表の下で与えられます。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。

## 26.7.1. 送信主装置動作

送信主装置(MT)動作では何バイトかのデータが受信従装置へ送信されます(下図をご覧ください)。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MTまたは受信主装置(MR)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MT動作へ移行し、**SLA+R**が送信されるとMR動作へ移行します。本項で言及する全ての状態符号は**前置分周選択ビットが0**か、または**0**で遮蔽されることが前提です。



開始条件はTWCR=1x10x10x形式の値をTWI制御レジスタ(TWCR)に書くことによって送出されます。

- TWI許可(TWCR.TWEN)ビットは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。
- 開始条件生成許可(TWCR.TWSTA)ビットは**開始条件**を送出するために**1**を書かれなければなりません。
- TWI割り込み要求(TWCR.TWINT)フラグはこのフラグを解除(**0**)するために**1**を書かれなければなりません。

その後に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、TWI状態レジスタ(TWSR)の状態符号が\$08になります(以降の**状態符号表**をご覧ください)。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TWIデータレジスタ(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TWCR.TWINTビットは(そこへ**1**を書くことによって)解除(**0**)されるべきです。これはTWCR=1x00x10x形式の値をTWCRに書くことによって達成されます。

**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以降の**状態符号表**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(そこへ**1**を書くことによって)解除(**0**)されるべきです。これはTWCR=1x00x10x形式の値をTWCRに書くことによって達成されます。

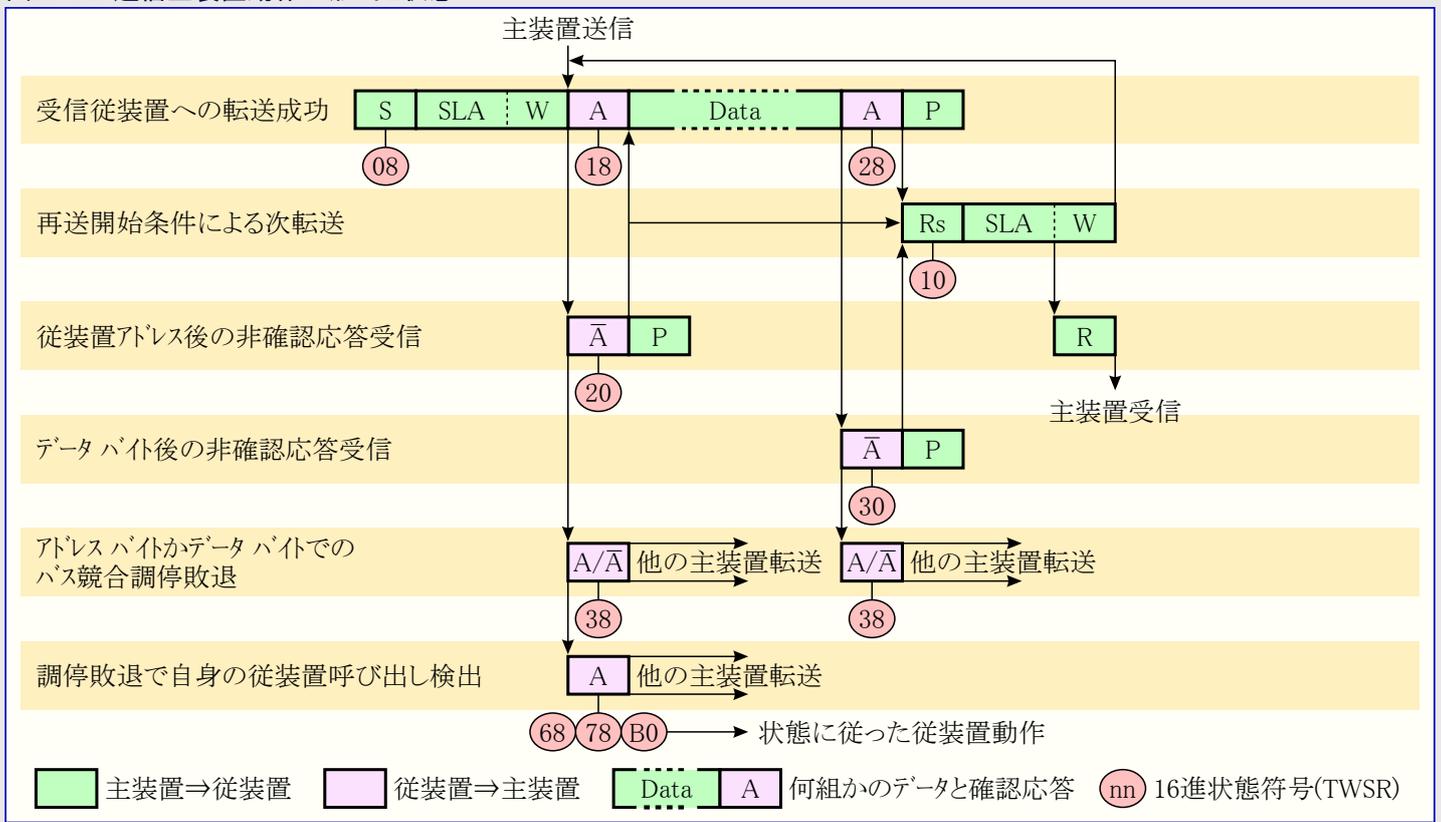
最後のバイトが送られてしまうまでこの手順が繰り返され、この転送は**停止条件**または**再送開始条件**のどちらかを生成することによって終了されます。**再送開始条件**は通常**開始条件**値(TWCR=1x10x10x)を書くことによって達成されます。**停止条件**はTWCR=1x01x10x形式の値を書くことによって生成されます。

**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26-2. 送信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

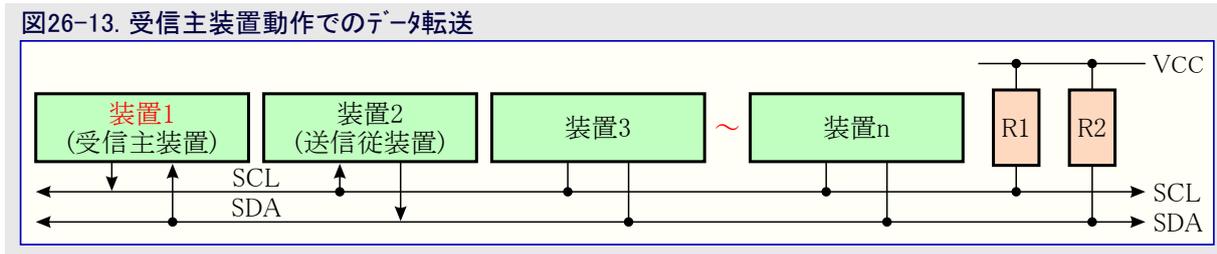
状態符号 (TWSR)	直前の動作と バスの状態	ソフトウェアの対応					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$08	開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+W設定	0	0	1	X	SLA+W送信、ACKかNACK受信
		SLA+R設定	0	0	1	X	SLA+R送信、受信主装置動作へ移行
\$18	SLA+W送信 ACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
		なし	0	1	1	X	停止条件送信、TWSTO=0
\$20	SLA+W送信 NACK受信	データ設定	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
		なし	1	0	1	X	再送開始条件送信
\$28	データバイト送信 ACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	0	1	1	X	停止条件送信、TWSTO=0
		なし	1	1	1	X	停止条件→開始条件送信、TWSTO=0
\$30	データバイト送信 NACK受信	データ設定	0	0	1	X	データ送信、ACKかNACK受信
		なし	1	0	1	X	再送開始条件送信
		なし	0	1	1	X	停止条件送信、TWSTO=0
\$38	SLA+W, データバイトで バス競合調停敗退	データ設定	1	1	1	X	停止条件→開始条件送信、TWSTO=0
		なし	0	0	1	X	バス開放、未指定従装置動作へ移行
		なし	1	0	1	X	バス開放時に開始条件送信

図26-12. 送信主装置動作の形式と状態



## 26.7.2. 受信主装置動作

受信主装置(MR)動作では何バイトかのデータが送信従装置から受信されます(次図をご覧ください)。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バケットの形式が送信主装置(MT)またはMR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MT動作へ移行し、**SLA+R**が送信されるとMR動作へ移行します。本項で言及する全ての状態符号は**前置分周選択ビット**が0か、または0で遮蔽されることが前提です。



開始条件はTWCR=1x10x10x形式の値をTWI制御レジスタ(TWCR)に書くことによって送出されます。

- TWI許可(TWCR.TWEN)ビットは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。
- 開始条件生成許可(TWCR.TWSTA)ビットは開始条件を送出するために1を書かれなければなりません。
- TWI割り込み要求(TWCR.TWINT)フラグはそこへ1を書くことによって解除(0)されなければなりません。

その後、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になります(以降の**状態符号表**をご覧ください)。**MR**動作へ移行するには**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フラグは(そこへ1を書くことによって)解除(0)されるべきです。これはTWCR=1x00x10x形式の値をTWCRに書くことによって達成されます。

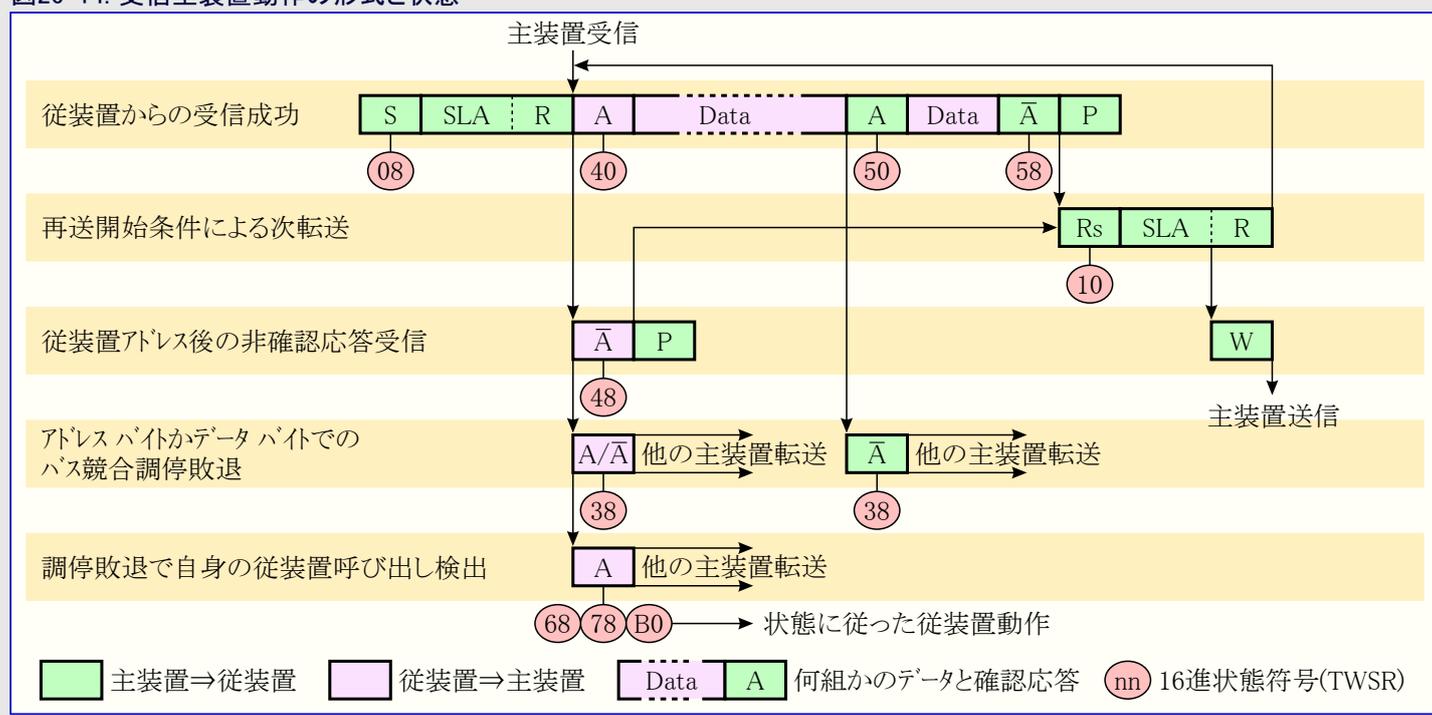
**SLA+R**が送信されて応答ビットが受信されてしまうと、TWINTが再び設定(1)され、TWSRの状態符号の数値が利用可能になります。主装置動作で可能性のある状態符号は\$38,\$40,\$48です。これら状態符号の各々に対する適切な動作は以降の**状態符号表**で詳述されます。ハードウェアによってTWINTフラグが設定(1)されると、受信したデータがTWDRから読めます。この手順は最後のバイトが受信されてしまうまで繰り返されます。最後のバイトが受信されてしまった後、受信主装置は最後に受信したデータバイト後の**NACK**送信によって送信従装置へ通知すべきです。この転送は**停止条件**または**再送開始条件**を生成することによって終了されます。**再送開始条件**はTWCR=1x10x10x形式の値を再びTWCRに書くことによって送出されます。**停止条件**はTWCR=1x01x10xを書くことによって生成されます。

**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26-3. 受信主装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

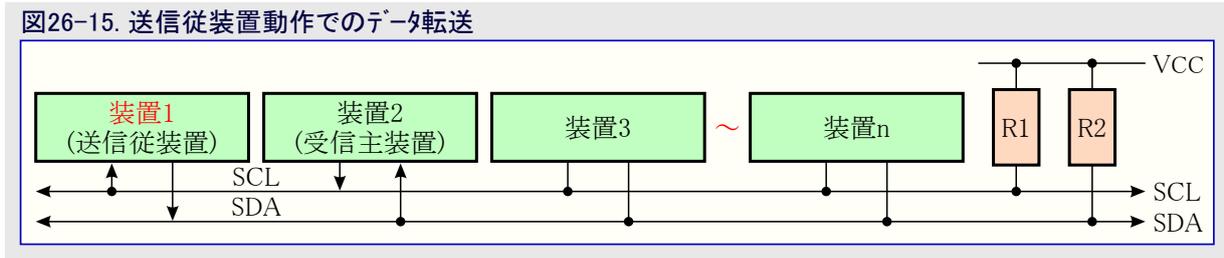
状態符号 (TWSR)	直前の動作と バスの状態	ソフトウェアの対応					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$08	開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
\$10	再送開始条件送信	SLA+R設定	0	0	1	X	SLA+R送信、ACKかNACK受信
		SLA+W設定	0	0	1	X	SLA+W送信、送信主装置動作へ移行
\$38	SLA+Rで調停敗退 またはNACK受信	なし	0	0	1	X	バス開放、未指定従装置動作へ移行
			1	0	1	X	バス開放時に開始条件送信
\$40	SLA+R送信 ACK受信	なし	0	0	1	0	データ受信、NACK応答
			0	0	1	1	データ受信、ACK応答
\$48	SLA+R送信 NACK受信	なし	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
\$50	データバイト受信 ACK応答	データ取得	1	1	1	X	停止条件→開始条件送信、TWSTO=0
			0	0	1	0	データ受信、NACK応答
\$58	データバイト受信 NACK応答	データ取得	0	0	1	1	データ受信、ACK応答
			0	1	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	1	0	1	X	再送開始条件送信
			0	1	1	X	停止条件送信、TWSTO=0
			1	1	1	X	停止条件→開始条件送信、TWSTO=0

図26-14. 受信主装置動作の形式と状態



### 26.7.3. 送信従装置動作

送信従装置(ST)動作では下図で示されるように、何バイトかのデータが送信主装置へ送信されます。本項で言及する全ての状態符号は前置分周選択ビットが0か、または0で遮蔽されることが前提です。



ST動作を始めるにはTWI(従装置)アドレスレジスタ(TWAR)とTWI制御レジスタ(TWCR)が次のように初期化されなければなりません。

TWARの上位7ビット(TWAR.TWA6~0)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。TWARの最下位(TWAR.TWGCE)ビットが1を書かれた場合、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。

TWCRはTWCR=0100010x形式の値を保持しなければなりません。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。TWCR.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。TWCR.TWSTAとTWCR.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、さもなければ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以降の状態符号表で詳述されます。ST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をご覧ください)。

転送中にTWCR.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。

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

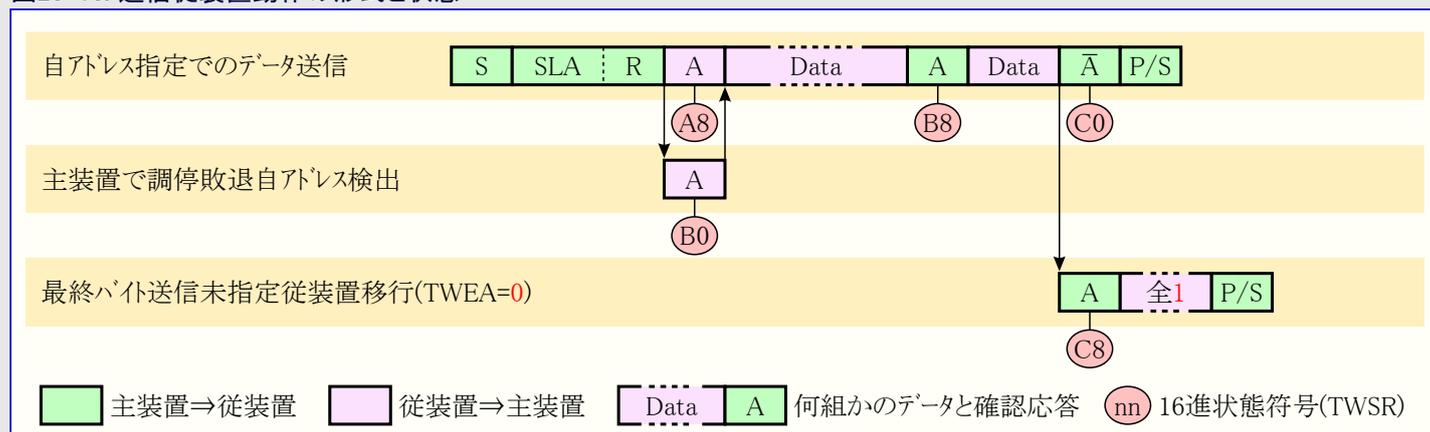
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが(それへ1を書くことによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれ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

注: (アイドル動作を除く)これらの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)はバスで渡す最後のバイトを反映しません。

表26-4. 送信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

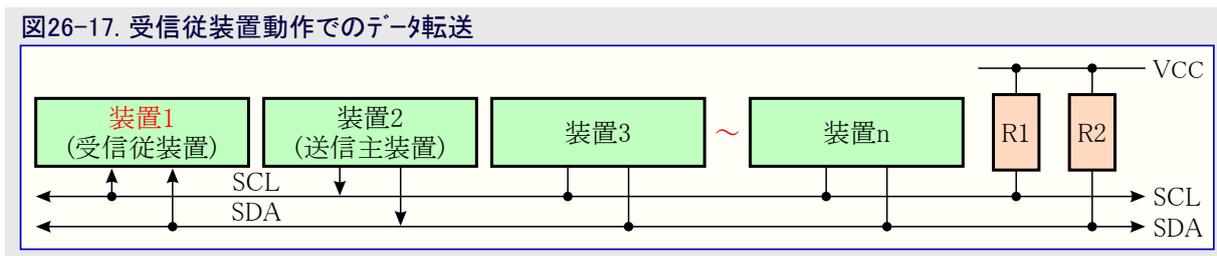
状態符号 (TWSR)	直前の動作と バスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$A8	自宛SLA+R受信 ACK応答	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$B0	主装置のSLA+R/Wで 調停敗退/自宛SLA+ R受信/ACK応答	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$B8	データバイト送信 ACK受信	データ設定	X	0	1	0	最終データバイト送信、NACK受信予定
			X	0	1	1	データバイト送信、ACK受信予定
\$C0	データバイト送信 NACK受信	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$C8	最終データバイト送信 (TWEA=0) ACK受信	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信

図26-16. 送信従装置動作の形式と状態



## 26.7.4. 受信従装置動作

受信従装置(SR)動作では何バイトかのデータが送信主装置から受信されます(下図をご覧ください)。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。



SR動作を始めるにはTWI(従装置)アドレスレジスタ(TWAR)とTWI制御レジスタ(TWCR)が次のように初期化されなければなりません。

TWARの上位7ビット(TWAR.TWA6~0)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。TWARの最下位(TWAR.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。

TWCRはTWCR=0100010x形式の値を保持しなければなりません。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。TWCR.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。TWCR.TWSTAとTWCR.TWSTOは0を書かれなければなりません。

TWARとTWCRが初期化されてしまうと、TWIは自身の従装置アドレス(または許可なら、一斉呼び出しアドレス)とそれに続くデータ方向ビットによってアドレス指定されるまで待機します。方向ビットが0(W)ならばTWIは受信従装置で動作し、さもなく(1(R)ならば送信従装置へ移行されます。自身の従装置アドレスとWビットが受信されてしまった後にTWINTフラグが設定(1)され、TWSRから有効な状態符号が読めます。以降の状態符号表で詳述されるように、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。SR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TWCR.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)を返しません。これは従装置がこれ以上受信できないことを示すのに使えます。TWEAが0の間中、TWIは自身の従装置アドレスに応答しませんが、2線直列バスは未だ監視され、アドレス認証はTWEAの設定(1)によって何時でも再開できます。これはTWEAビットがTWIを2線直列バスから一時的に隔離するのに使えることを意味します。

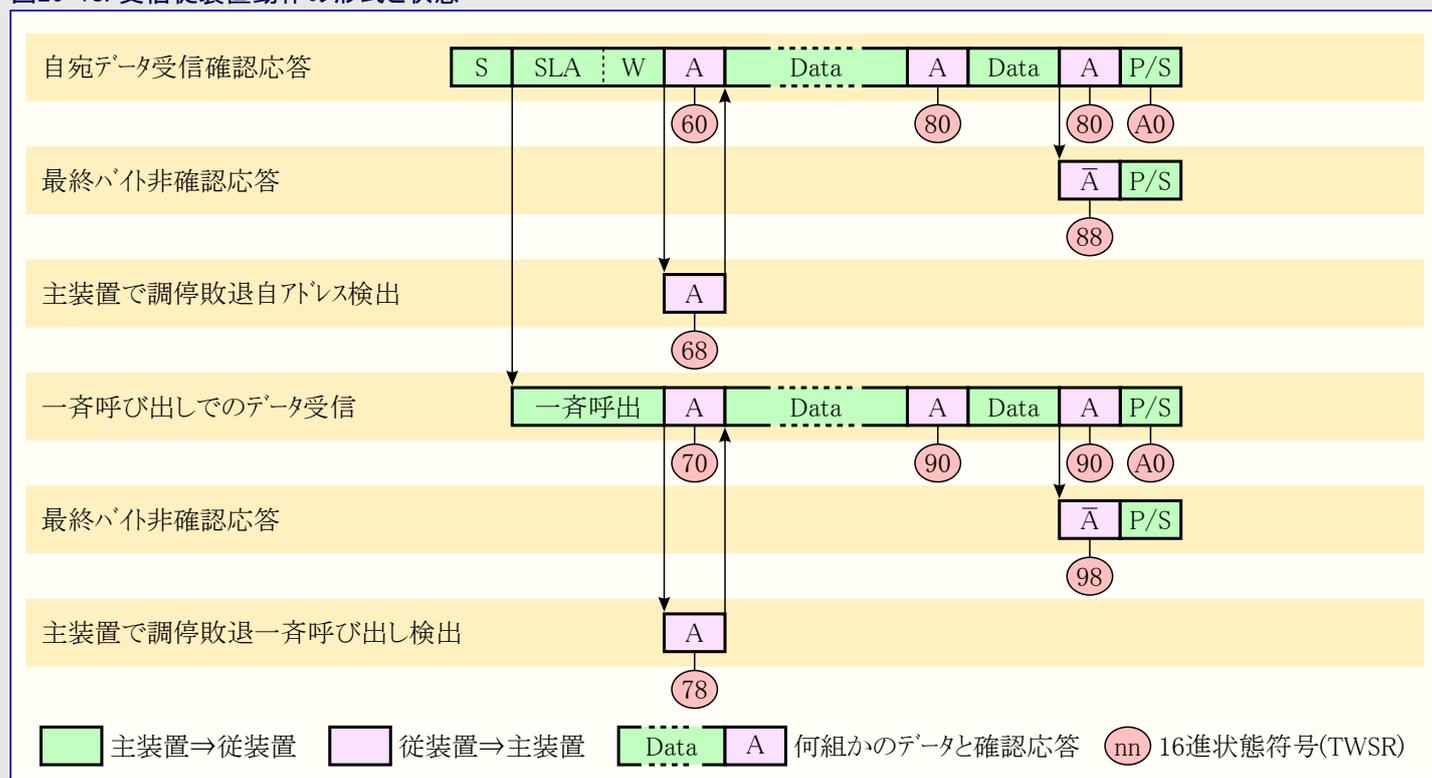
アイドル動作を除く休止形態ではTWIへのクロック系がOFFにされます。TWEAビットが設定(1)されていると、このインターフェースはクロック元として2線直列バスクロック(SCL)を使うことにより、自身の従装置アドレスと一斉呼び出しに未だ確認応答できます。その後デバイスが休止形態から起動し、TWIは起動中からTWINTフラグが(それへ1を書くことによって)解除(0)されるまでSCLクロックをLowに保ちます。その後のデータ受信はAVRクロックが通常通り走行することで通常通りに行われます。AVRが長い起動時間に設定されていると、SCL信号線が長時間Lowに保持され、他のデータ送信を阻止するかもしれ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

注: 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)はバスで渡す最後のバイトを反映しません。

表26-5. 受信従装置動作の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

状態符号 (TWSR)	直前の動作とバスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$60	自宛SLA+W受信 ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$68	主装置のSLA+R/Wで 調停敗退/自宛SLA+W 受信/ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$70	一斉呼び出し受信 ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$78	主装置のSLA+R/Wで 調停敗退/一斉呼び 出し受信/ACK応答	なし	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$80	自宛データバイト受信 ACK応答	データ取得	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$88	自宛データバイト受信 NACK応答	データ取得	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$90	一斉呼び出しのデータ バイト受信/ACK応答	データ取得	X	0	1	0	データ受信、NACK応答
			X	0	1	1	データ受信、ACK応答
\$98	一斉呼び出しのデータ バイト受信 NACK応答	データ取得	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信
\$A0	自指定中の 停止条件または 再送開始条件検出	なし	0	0	1	0	未指定従装置動作へ移行、応答禁止
			0	0	1	1	未指定従装置動作へ移行、応答対応
			1	0	1	0	未指定従装置動作へ移行、応答禁止 バス開放で開始条件送信
			1	0	1	1	未指定従装置動作へ移行、応答対応 バス開放で開始条件送信

図26-18. 受信従装置動作の形式と状態



## 26.7.5. その他の状態

定義したTWI状態に従わない2つの状態符号があります。本項内の表をご覧ください。

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)されて、TWINTが論理1書き込みによって解除(0)されなければなりません。これはTWIをアドレス指定されていない従装置動作にさせ、TWSTOビットを解除(0)させます(TWCRの他のビットは影響されません)。SDAとSCL信号進は開放され、停止条件は送出されません。

表26-6. その他の状態符号 (注: TWSRの前置分周選択ビットは0の前提)

状態符号 (TWSR)	直前の動作と バスの状態	ソフトウェアの応答					TWCR設定によるハードウェア動作
		TWDR操作	TWCR設定				
			TWSTA	TWSTO	TWINT	TWEA	
\$F8	適切な状態情報なし TWINT=0	なし	-	-	-	-	待機または現在の転送続行
\$00	不正な開始条件/停止条件でのバス異常	なし	0	1	1	X	停止条件を送出せずにバスを開放 TWSTO=0

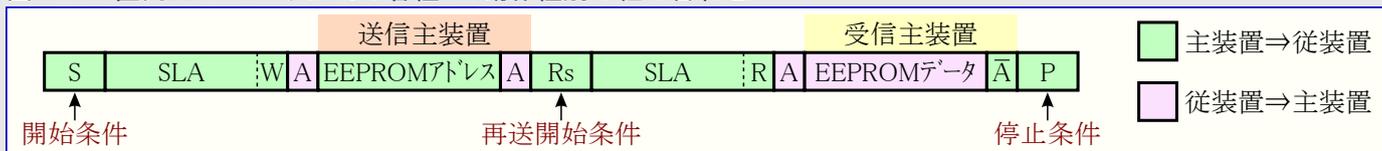
## 26.7.6. 各種TWI動作種別の組み合わせ

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

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

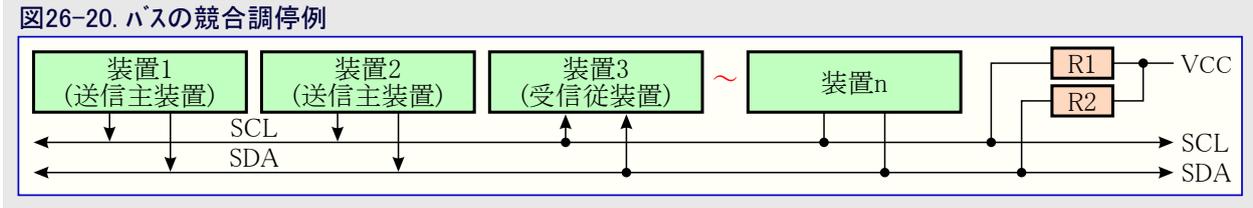
データが主装置から従装置へとその逆の両方向へ転送されることに注意してください。主装置はどの場所を読みたいかを従装置に指示しなければならず、送信主装置動作の使用を必要とします。その後にデータを従装置から読まねばならず、受信主装置動作の使用を意味します。従って転送方向が切り替えられなければなりません。主装置はこれら全ての段階中にバスの制御を保持しなければならず、この手順は排他的(非分断)操作として行われるべきです。複数主装置システムでこの原則に違反すると、他の主装置が2と3の段階間でEEPROM内のデータポインタを変更するかもしれず、(元の)主装置は不正なデータ位置を読むでしょう。このような転送方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

図26-19. 直列EEPROMアクセスでの各種TWI動作種別の組み合わせ



## 26.8. 複数主装置システムでのバス競合と調停

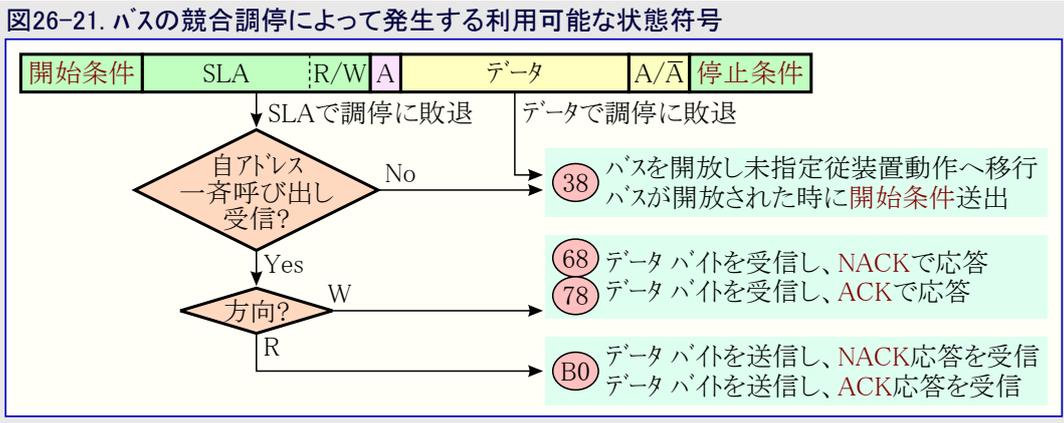
複数の主装置が同じバスに接続されると、それらの1つまたはそれ以上によって同時に送信が開始されるかもしれません。TWIは主装置の1つが転送を続けることを許され、手順内でデータが失われないような方法でこのような状態が扱われることを標準で保証します。2つの主装置が受信従装置へデータを送信することを試みる場合の調停状況の例は以下で図示されます。



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つて新規開始条件を送出します。
- 複数の主装置が異なる従装置をアクセスする場合。この場合、SLAビット内で調停が起きます。他の主装置がSDAに0を出力する間に1を出力しようとする主装置が調停に敗れます。SLA内で調停に敗れた主装置は勝った主装置によってアドレス指定されるかを確認するために従装置動作へ切り替えます。アドレス指定されると、R/Wビットの値によって受信従装置(SR)動作または送信従装置(ST)動作へ切り替えます。アドレス指定されないなら、応用ソフトウェアの処置によって未指定従装置動作に切り替えるか、またはバスが開放になるまで待つて新規開始条件を送出します。

これは次図で要約されます。利用可能な状態符号は楕円(訳注:原文は円)で与えられます。



## 26.9. TWI用レジスタ

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

名称 : TWBR  
 変位 : \$B8  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TWBR7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	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。

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

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

名称 : TWCR  
 変位 : \$BC  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
アクセス種別	R/W	R/W	R/W	R/W	R	R/W	R	R/W
リセット値	0	0	0	0	0	0	0	0

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

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

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

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

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

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

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

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

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

TWENビットはTWI動作を許可し、TWIインターフェースを活性(有効)にします。TWENが1を書かれると、TWIはSCLとSDAのピンに接続したI/Oピンを制御できるようになり、スパイク濾波器とスロ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。

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

名称 : TWSR  
 変位 : \$B9  
 リセット : \$F8  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TWS7~3				-	TWPS1,0		
アクセス種別	R	R	R	R	R	R	R/W	R/W
リセット値	1	1	1	1	1	0	0	0

● **ビット7~3 – TWS7~3 : TWI状態 (TWI Status)**

TWS7~3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「**転送種別**」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「**ビット速度発生器**」を参照してください。TWPS1,0の値はこの式で使われます。

表26-7. TWIビット速度前置分周器選択

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

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

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

名称 : TWDR  
 変位 : \$BB  
 リセット : \$FF  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TWD7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	1	1	1	1	1	1	1	1

● **ビット7~0 – TWD7~0 : TWIデータ (TWI Data)**

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

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

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。一致が見つかる割込み要求が生成されます。

名称 : TWAR

変位 : \$BA

リセット : \$FE

特質 : -

ビット	7	6	5	4	3	2	1	0
	TWA6~0							TWGCE
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	1	1	1	1	1	1	1	0

● 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。

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

名称 : TWAMR

変位 : \$BD

リセット : \$00

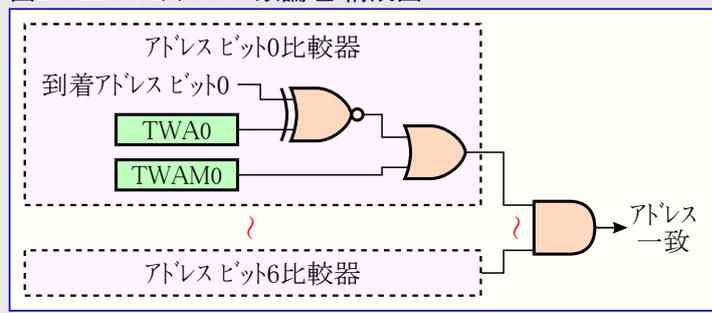
特質 : -

ビット	7	6	5	4	3	2	1	0
	TWAM6~0							-
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
リセット値	0	0	0	0	0	0	0	0

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

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

図26-22. TWIアドレス一致論理構成図



## 27. AC – アナログ比較器 (Analog Comparator)

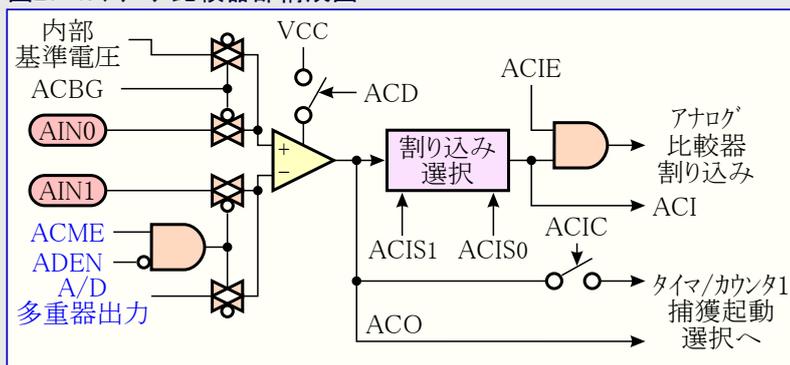
### 27.1. 概要

アナログ比較器は非反転入力AIN0ピンと反転入力AIN1ピンの入力値を比較します。非反転AIN0ピンの電圧が反転AIN1ピンの電圧よりも高い時にACSRのアナログ比較器出力(ACO)ビット(任意でポートCのPC5)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右で示されます。

ADC入力(A/D多重器出力)の使用を可能とするには、電力削減レジスタのA/D変換器電力削減(PRR.PRADC)ビットが0を書かれなければなりません。

関連リンク [35頁の「PM – 電力管理と休止形態」](#)  
[37頁の「消費電力の最小化」](#)

図27-1. アナログ比較器部構成図



注: アナログ比較器ピン配置については「ピン配置」と「入出力ポート」の記述を参照してください。

### 27.2. 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、故にこの機能を利用するにはA/D変換部がOFF(動作禁止)にされなければなりません。A/D変換制御/状態レジスタB(ADCSR)のアナログ比較器多重器許可(ACME)ビットが1で、A/D変換部がOFF(ADCSRAのADENビットが0)にされていれば、右表で示されるようにA/D多重器選択(ADMUX)レジスタのA/Dチャンネル選択(MUX2~0)ビットの下位3ビットがアナログ比較器への反転入力を取り替えるための入力ピンを選びます。ADCSR.ACME=0またはADCSRA.ADEN=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27-1. アナログ比較器反転入力選択

ACME	ADEN	MUX2~0	アナログ比較器反転入力	
0	x	x x x	AIN1	
	1	x x x		ADC0
		0 0 0		ADC1
1	0	0 0 1	ADC2	
		0 1 0	ADC3	
		0 1 1	ADC4	
		1 0 0	ADC5	
		1 0 1	ADC6	
		1 1 0	ADC7	
		1 1 1	ADC7	

(訳補) ADC6とADC7はTQFPとQFN/MLF32外囲器でだけ有効です。

### 27.3. アナログ比較器用レジスタ

#### 27.3.1. ADCSRB – A/D変換制御/状態レジスタB (ADC Control and Status Register B)

名称 : ADCSRB  
 変位 : \$7B  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	ACME	-	-	-	ADTS2~0		
アクセス種別	R	R/W	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット6 – 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択します。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「アナログ比較器入力選択」をご覧ください。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : ACSR  
 変位 : \$50 (\$30)  
 リセット : '00x00000'

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$30です。

ビット	7	6	5	4	3	2	1	0
	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1,0	
アクセス種別	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	不定	0	0	0	0	0

#### ● ビット7 – ACD : アナログ比較器禁止 (Analog Comparator Disable)

このビットが論理1を書かれると、アナログ比較器への電力がOFFにされます。このビットはアナログ比較器をOFFにするために何時でも設定(1)できます。これは活動動作やアイドル動作で電力消費を削減します。ACDビットを変更する時にACSRでアナログ比較器割り込み許可(ACIE)ビットを解除(0)することによってアナログ比較器割り込みが禁止されなければなりません。さもなければ、このビットが変更される時に割り込みが起り得ます。

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

このビットが設定(1)されると、内部基準電圧(公称1.1V)がアナログ比較器への非反転入力に置き換わります。本ビットが解除(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。内部基準電圧がアナログ比較器入力として使われるとき、電圧の安定に一定時間を必要とします。安定(待機を)しない場合、最初は不正値を与えるかもしれません。

#### ● 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、その後直接ACOへ接続されます。この同期化は1～2クロック周期の遅延をもたらします。

#### ● ビット4 – ACI : アナログ比較器割り込み要求フラグ (Analog Comparator Interrupt Flag)

このビットは比較器出力での出来事がACSRのアナログ比較器割り込み条件(ACIS1,0)ビットによって定義した割り込み方法で起動する時に設定(1)されます。ACSRのアナログ比較器割り込み許可(ACIE)ビットが設定(1)され、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されていると、アナログ比較器割り込みルーチンが実行されます。対応する割り込み処理ベクタを実行すると、ACIはハードウェアによって解除(0)されます。代わりにこのフラグへ論理1を書くことによってもACIは解除(0)されます。

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

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

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

論理1を書かれると、このビットはアナログ比較器によって起動されるタイマ/カウンタ1の捕獲機能を許可します。この場合、比較器出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音除去機能と端(エッジ)選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。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1割り込み許可レジスタ(TIMSK1)の捕獲割り込み許可(ICIE)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。

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27-2. アナログ比較器割り込み条件選択

ACIS1	ACIS0	割り込み発生条件
0	0	比較器出力の変移 (トグル)
0	1	(予約)
1	0	比較器出力の下降端
1	1	比較器出力の上昇端

### 27.3.3. DIDR1 – デジタル入力禁止レジスタ1 (Digital Input Disable Register 1)

名称 : DIDR1

変位 : \$7F

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	AIN1D	AIN0D
アクセス種別	R	R	R	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット1 – AIN1D : AIN1デジタル入力禁止 (AIN1 Digital Input Disable)

● ビット0 – AIN0D : AIN0デジタル入力禁止 (AIN0 Digital Input Disable)

このビットが論理1を書かれると、AIN1/0ピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、対応するポート入力レジスタのビット(PINx)は常に0として読みます。AIN1/0ピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされない時にデジタル入力緩衝部での消費電力を削減するため、このビットは論理1を書かれるべきです。

## 28. ADC – A/D変換器 (Analog to Digital Converter)

### 28.1. 特徴

- 10ビット分解能
- 積分非直線性誤差0.5 LSB
- 絶対精度±2 LSB
- 変換時間13~260µs (50kHz~1MHz変換クロック)
- 76.9kSPS(採取/s)まで (最大分解能で15kSPSまで)
- 6チャンネルのシングルエンド入力多重器内蔵
- 2チャンネルの追加シングルエンド入力多重器内蔵 (TQFP, QFN/MLF32外囲器のみ)
- 温度感知器入力チャンネル
- A/D変換結果読み出しに対する任意の左揃え
- 0~VCC A/D変換入力電圧範囲
- 選択可能な1.1V A/D変換基準電圧
- 連続と単独の変換動作
- 割り込み元の自動起動によるA/D変換開始
- A/D変換完了割り込み
- 休止形態雑音低減機能

### 28.2. 概要

本デバイスは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Aのピンから構成された8つのシングルエンド電圧入力を許す8チャンネルアナログ多重器に接続されます。このシングルエンド電圧入力は0V(GND)が基準です。

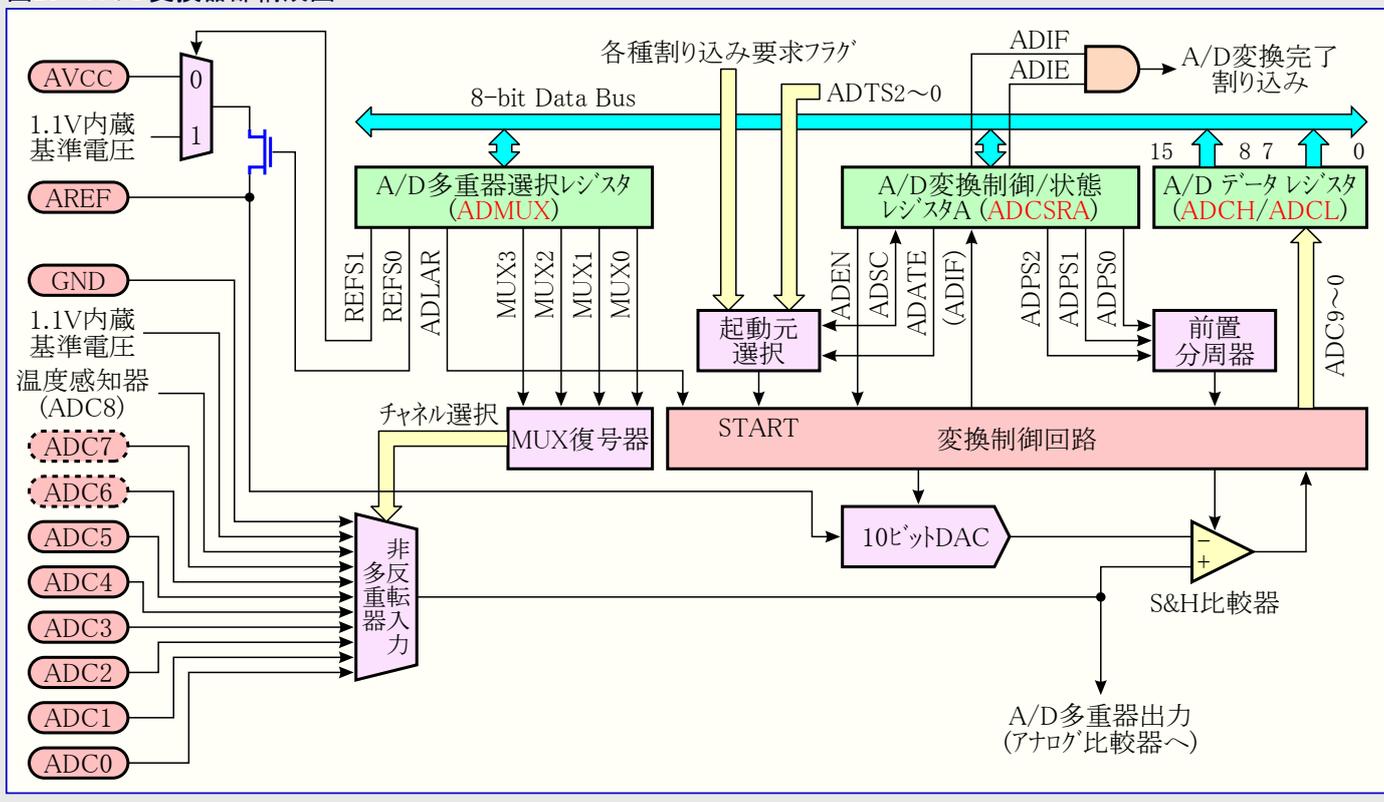
A/D変換器はA/D変換器への入力電圧が変換中に一定の値で保持されることを保証する採取&保持(S/H)回路を含みます。A/D変換部の構成図は下で示されます。

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「[雑音低減技術](#)」項をご覧ください。

電力削減レジスタのADC電力削減(PRR.PRADC)ビットはA/D変換部を許可するために0を書かれなければなりません。

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1.1V基準電圧がAREFピンに接続できます。従ってこの内部基準電圧は雑音耐性を改善するためにAREFピンで外部コンデンサによってデカップ(雑音結合減少)することができます。

図28-1. A/D変換器部構成図



アナログ入力チャンネルはA/D多重器選択(ADMUX)レジスタのチャンネル選択(MUX3~0)ビットへの書き込みによって選ばれます。GNDと固定基準電圧(1.1V内蔵基準電圧(VBG))だけでなく、どのADC入力ピン(ADC7~0)もがA/D変換器のシングルエンド入力として選べます。A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットに1を書くことによって(動作が)許可されます。基準電圧と入力チャンネルの選択は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データレジスタ(ADCH,ADCL)で示される10ビットの結果を生成します。既定では、この結果は右揃え(16ビットのビット0側10ビット)で表されますが、ADMUXで左揃え選択(ADLAR)ビットを設定(1)することにより、任意で左揃え(16ビットのビット15側10ビット)で表せます。

この結果が左揃え補正され、8ビットを越える精度が必要とされない場合はADCHを読むことで足りる。さもなければデータレジスタの内容が同じ変換に属することからの結果を保証するため、ADCLが初めに、次にADCHが読まれなければなりません。一度ADCLが読まれると、A/D変換器からのA/Dデータレジスタ(ADCH,ADCL)アクセスが阻止されます。これはADCLが読まれてしまい、ADCHが読まれる前に次の(第2の)変換が完了すると、どちらのレジスタ(ADCH,ADCL)も更新されず、第2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。

関連リンク [35頁の「PM - 電力管理と休止形態」](#)  
[36頁の「電力削減レジスタ」](#)

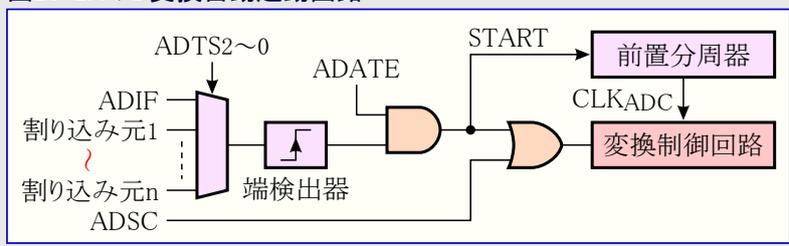
## 28.3. 変換の開始

単独変換は電力削減レジスタ(PRR)のA/D変換器電力削減(PRADC)ビットに0を書き、A/D変換制御/状態レジスタA(ADCSRA)の変換開始(ADSC)ビットに1を書くことによって開始されます。ADSC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ンネルが選ばれると、A/D変換部はそのチャンネル変更を実行する前に現在の変換を済ませます。

代わりに、変換は様々な起動元によって自動的に起動できます。自動起動はA/D変換自動起動許可(ADCSRA.ADATE)ビットの設定(1)によって許可されます。起動元はA/D変換制御/状態レジスタB(ADCSRB)のA/D変換起動元選択(ADTS2~0)ビットの設定によって選ばれます。利用可能な起動元の一覧についてはADCSRB.ADTSの記述をご覧ください。

選んだ起動信号上の上昇端が起きると、A/D変換用前置分周器がリセットし、変換が開始されます。これは一定間隔での変換開始の方法を提供します。変換完了時、起動信号が未だ設定(1)されている場合、新規の変換は開始されません。変換中にこの起動信号上で別の上昇端が起きると、その端(エッジ)は無視されます。指定した割り込みが禁止またはAVR ステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)でも、割り込み要求フラグが設定(1)されることに注意してください。従って割り込みを起こさずに変換が起動できます。けれども次の割り込み要因で新規変換を起動するために、割り込み要求フラグは解除(0)されなければなりません。

図28-2. A/D変換自動起動回路



起動元としてA/D変換完了割り込み要求フラグ(ADIF)を使うことは、A/D変換器に実行中の変換が完了されると直ぐに新規変換を開始させます。そのためA/D変換器は連続動作で動き、継続的な採取(変換)とA/Dデータレジスタを更新します。最初の変換はADCSRAでADSCビットに1を書くことによって始めなければなりません。この動作でのA/D変換器はA/D変換完了割り込み要求フラグ(ADIF)が解除(0)されるかどうかにかかわらず、連続的な変換を実行します。

自動起動が許可されている場合、ADCSRAのADSCビットに1を書くことによって単独変換を開始できます。ADSCは変換が進行中かを決めるためにも使えます。ADSCビットは変換がどう開始されたかに拘らず、変換中は1として読みます。

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

## 28.4. 前置分周と変換タイミング

既定での逐次比較回路は最大分解能を得るのに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(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のA/D許可(ADEN)ビットに1を書くことによって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=1である限り走行を保ち、ADEN=0の時に継続的にリセットします。

ADCSRAのA/D変換開始(ADSC)ビットに1を書くことによってシングル エント 入力の変換を起動すると、その変換は直後の変換クロックの上昇端で始まります。

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にされる(即ちADCSRAのADENが1を書かれた)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。

内部基準電圧がA/D変換器への入力として使われるとき、電圧の安定に一定時間を必要とします。安定(待機を)しない場合は初回変換後の最初の読み込み値は不正になるかもしれません。

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(ADATE=0)では同時にADCSRAのADSCビットが解除(0)されます。その後ソフトウェアは再びADCSRAのADSCを設定(1)でき、新規変換は変換クロックの最初の上昇端で開始されます。

自動起動が使われると、前置分周器は起動要因発生時にリセットされます。これは起動要因から変換開始までの一定の遅延を保証します。この動作での採取&保持は起動要因となる信号の上昇後、2変換クロック周期で採取が行われます。同期化論理回路(端(エッジ)検出器)に対して、追加の3 CPUクロック周期が費やされます。

連続変換動作(ADATE=1)では変換完了後直ちに新規変換が開始され、一方ADSCは1に留まります。以降のA/D変換時間の表もご覧ください。

図28-3. A/D変換前置分周器部構成

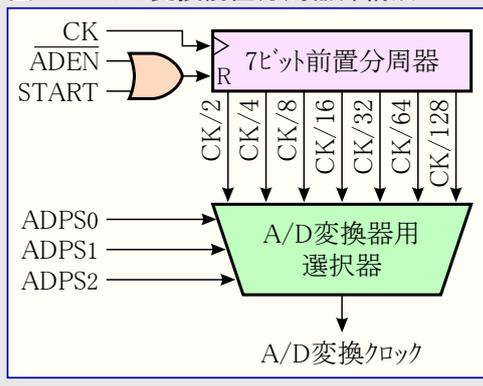


図28-4. 初回変換タイミング (単独変換動作)

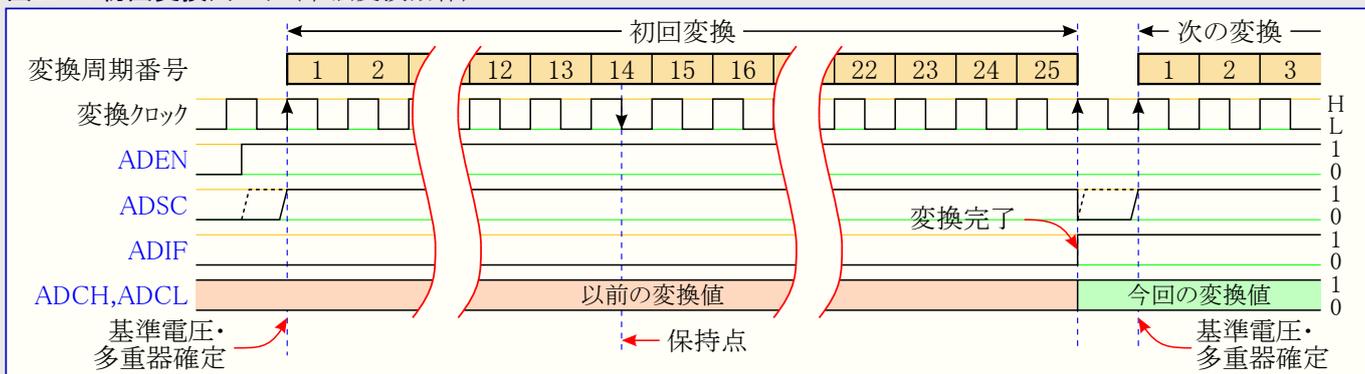


図28-5. 通常変換タイミング (単独変換動作)

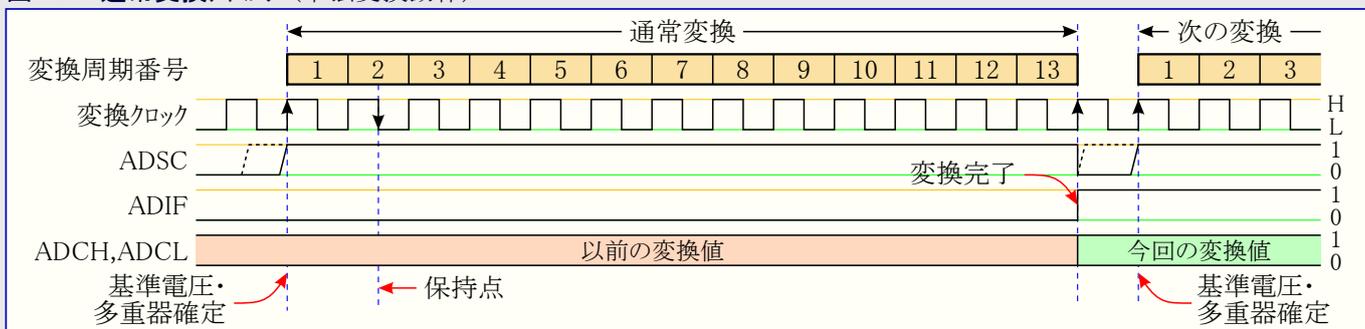


図28-6. 通常変換タイミング\* (自動起動変換動作)

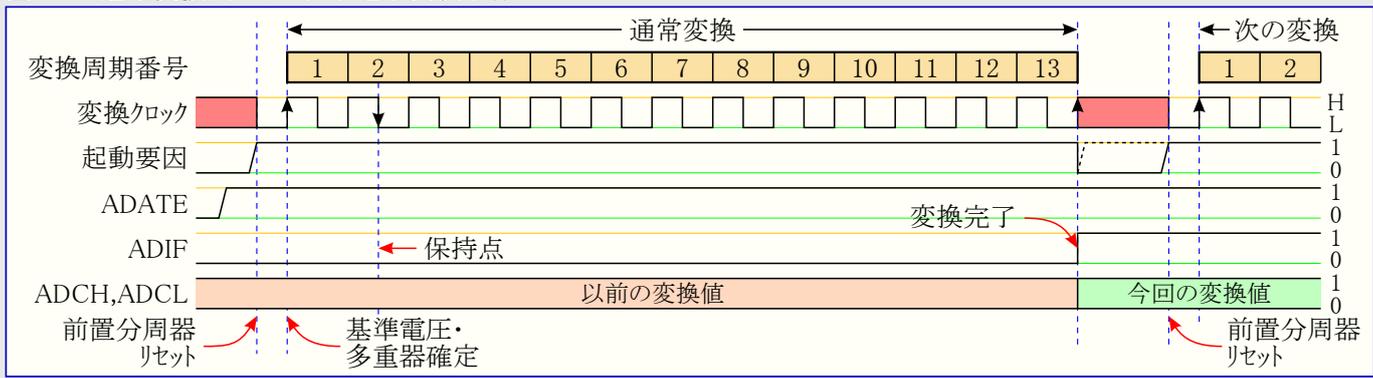


図28-7. 連続変換動作タイミング\*

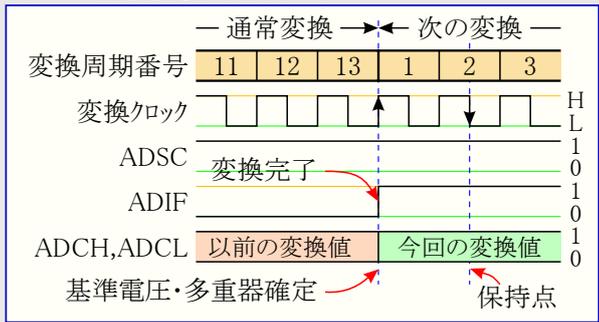


表28-1. A/D変換時間

変換種別	保持点	変換時間
初回変換	13.5	25
シングルエンド入力通常変換	1.5	13
自動起動変換	2	13.5

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。

## 28.5. チャネル変更と基準電圧選択

A/D多重器選択(ADMUX)レジスタのチャネル選択(MUX3~0)ビットと基準電圧選択(REFS1,0)ビットはCPUが乱順にアクセスするための一時レジスタを通して単独緩衝されます。これはチャネルと基準電圧の選択が変換中の安全なところでだけ行うのを保証します。チャネルと基準電圧の選択は変換が開始されるまで継続的に更新されます。一旦変換が始まると、A/D変換器に対して十分な採取/変換時間を保証するためにチャネルと基準電圧の選択は固定されます。継続的な更新は(ADCSRAのADIFの設定(1))によって示される変換完了前の最後の変換クロック周期で再開します。A/D変換制御/状態レジスタ(ADCSRA)の変換開始(ADSC)ビットが書かれた後の次の変換クロックの上昇端で変換が始まることに注意してください。従って使用者はADCSRA.ADSCビットが書かれた後、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。

自動起動が使われる場合、起動要因の正確な時間は確定できません。変換が新規設定によって影響されるように制御するにはADMUXの更新時に特別な注意が祓われなければなりません。

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

1. ADENまたはADATEが解除(0)されているとき。
    - 1-1. 変換開始後、最低1変換クロック周期経過後の変換中。
    - 1-2. 変換後から、変換起動元として使った割り込みフラグが解除(0)される直前まで。
- これら条件の1つでADMUXを更新すると、新設定は次のA/D変換に影響を及ぼします。

### 28.5.1. A/D入力チャネル

チャネル選択を変更する時に使用者は正しいチャネルが選ばれる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。使用者は連続変換動作中に新しいチャネルや基準電圧選択を書かないことが推奨されます。

差動増幅チャネルへ切り替える時に自動変位(オフセット)消去回路用の設定時間が必要なので、最初の変換結果は貧弱な正確さとなってしまうかもしれません。使用者は最初の変換結果をなるべくなら無視すべきです。

## 28.5.2. A/D変換基準電圧

このA/D変換用の基準電圧( $V_{REF}$ )はA/D変換に対する変換範囲を示します。 $V_{REF}$ を越えるシングル エント'入力チャネルは\$3FFで打ち切る符号に帰着します。 $V_{REF}$ はAVCC、内部1.1V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してA/D変換部に接続されます。1.1Vの内部基準電圧は内蔵基準(バンドギャップ)電圧( $V_{BG}$ )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。 $V_{REF}$ (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。 $V_{REF}$ は高インピーダンス出力で、容量性負荷のみがシステム内で接続されるべきであることに注意してください。

使用者がAREFピ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AREFピ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1.1V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ない、使用者はこの結果を破棄することが推奨されます。

## 28.6. 雑音低減機能

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

1. A/D変換部が許可( $ADEN=1$ )され、変換中でない( $ADSC=0$ )ことを確認してください。単独変換動作が選択( $ADATE=0$ )され、且つA/D変換完了割り込みが許可( $ADIE=1$ )されていないとなりません。
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。

**注:** 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変換部は自動的にOFFへ切り替えられません。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ADCSRAのADENへ0を書くことが推奨されます。

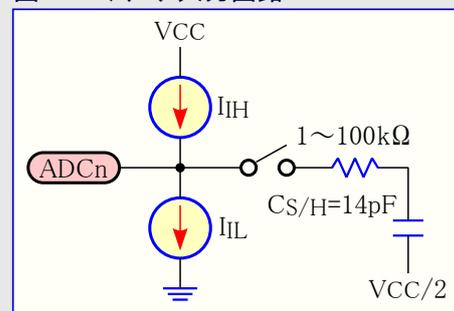
### 28.6.1. アナログ入力回路

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

A/D変換部は概ね10k $\Omega$ 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( $f_{ADC}/2$ )よりも高い信号成分が存在すべきではありません。使用者はADC入力として信号を印加する前に低域通過濾波器(ローパスフィルタ)で高い周波数成分を取り除くことが推奨されます。

図28-8. アナログ入力回路

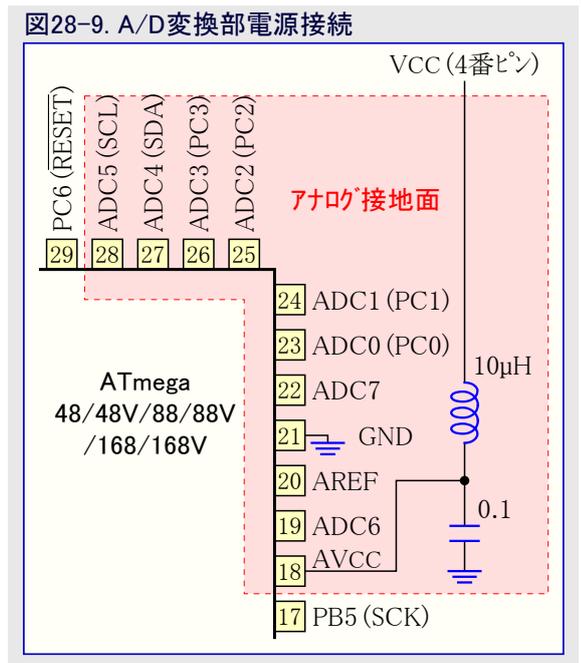


## 28.6.2. アナログ雑音低減技術

デバイス内外のデジタル回路がアナログ測定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。アナログ信号線がアナログGND面上を走ることに注意し、高速切り替えデジタル信号線から充分離すことを守ってください。
2. デバイスのAVCCピンは右図で示されるように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、これらは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) 図28-9は原図に対し修正され、ピン名は部分的に省略されています。



## 28.6.3. A/D変換の精度定義

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

### • 変位(オフセット)誤差 - 図28-10.

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

### • 利得誤差 - 図28-11.

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

### • 積分非直線性誤差 (INL) - 図28-12.

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

### • 微分非直線性誤差 (DNL) - 図28-13.

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

### • 量子化誤差

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

### • 絶対精度

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

図28-10. 変位(オフセット)誤差

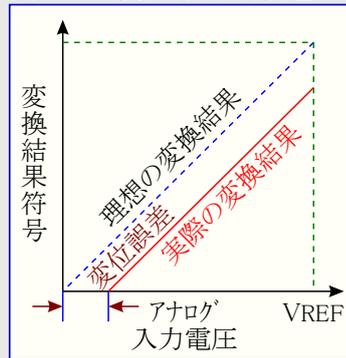


図28-11. 利得誤差

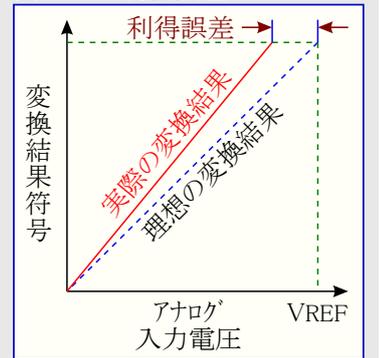


図28-12. 積分非直線性誤差

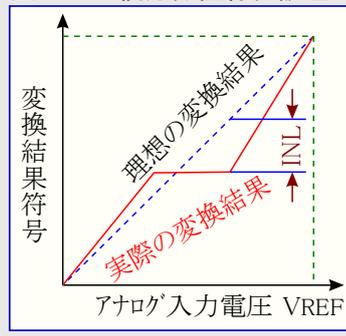
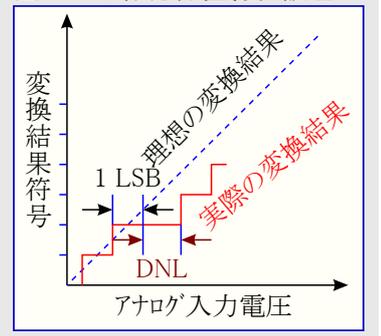


図28-13. 微分非直線性誤差



## 28.7. A/D変換の結果

変換完了(ADCSRAのADIFの設定(1))後、変換結果はA/Dデータレジスタ(ADCH, ADCL)で得られます。

シングルエンド入力変換での結果は右式で示されます。

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

$V_{IN}$ は選んだ入力ピンの電圧、 $V_{REF}$ は選んだ基準電圧です(ADMUXのREFSとMUXの記述もご覧ください)。

\$000はアナログGNDを表し、\$3FFは選んだ基準電圧-1 LSBを表します。

## 28.8. 温度測定

温度測定はシングル エント 温度感知器チャネルに連結されるチップ上の温度感知器に基づきます。A/D多重器選択(ADMUX)レジスタのチャンネル選択(MUX3~0)ビットへの'1000'書き込みによる温度感知器チャネル選択が温度感知器を許可します。温度感知器測定でのA/D変換器基準電圧源に対しては1.1V内部基準電圧が選ばなければなりません。温度感知器が許可されると、A/D変換器は温度感知器上の電圧を測定するために単独変換動作(ADATE=0)で使うことができます。

測定した電圧は右表で記述されたように温度に対して直線的関係を持ちます。電圧感度は概ね1mV/°Cで、温度測定の精度は±10°Cです。

右表に記載した値は代表値です。けれども、製法変化のため、温度感知器出力電圧は或るチップと別のチップで変化します。より正確な結果の達成を可能とするために温度測定は応用ソフトウェアで校正することができます。ソフトウェア校正は量産検査の一部として各チップに対して校正値が測定され、レジスタまたはEEPROM内に格納されることが必要です。ソフトウェア校正は右式を使って行うことができます。

ここでADCH:ADCLはA/D変換器データレジスタ、 $k$ は固定係数(訳補:希望温度形式に依存)、 $T_{OS}$ は量産検査の一部として決定され、EEPROM内に格納される温度感知器変位(オフセット)値です。

表28-2. 温度対感知器出力電圧(代表条件)

温度(°C)	-45°C	+25°C	+85°C
電圧(mV)	242mV	314mV	380mV

$$T(\text{温度}) = \frac{(\text{ADCH} \ll 8 | \text{ADCL}) - T_{OS}}{k}$$

## 28.9. A/D変換用レジスタ

### 28.9.1. ADMUX – A/D多重器選択レジスタ (ADC Multiplexer Select Register)

名称 : ADMUX  
 変位 : \$7C  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	REFS1,0		ADLAR	-	MUX3~0			
アクセス種別	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
リセット値	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28-3. A/D変換部の基準電圧選択

REFS1	REFS0	基準電圧
0	0	AREFピンの外部基準電圧 (AVCCと内部基準電圧は切り離されます。)
0	1	AVCC (内部基準電圧は切り離され、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1.1V基準電圧 (AVCCは切り離され、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。さもない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変換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「A/D変換データレジスタ下位」と「A/D変換データレジスタ上位」をご覧ください。

#### ● ビット3~0 – MUX3~0 : A/Dチャネル選択 (Analog Channel Selection)

これらのビットの値はA/D変換器にどのアナログ入力が入力が接続されるかを選びます。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が設定(1)される)まで実施しません。

表28-4. アナログ入力チャネル選択

MUX3~0	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001~1101	1110	1111
アナログ入力チャネル	ADC0	ADC1	ADC2	ADC3	ADC4	ADC5	(ADC6)	(ADC7)	ADC8	(予約)	1.1V	0V
備考	PC0	PC1	PC2	PC3	PC4	PC5	ADC6	ADC7	温度		V <sub>BG</sub>	GND

(訳注) PDIPとQFN/MLF28外周器は6チャネルのため、ADC6,7はありません。

### 28.9.2. ADCSRA – A/D制御/状態レジスタA (ADC Control and Status Register A)

名称 : ADCSRA  
 変位 : \$7A  
 リセット : \$00  
 特質 : -

ビット	7	6	5	4	3	2	1	0
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2~0		
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	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にされます。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 – ADATE : A/D変換自動起動許可 (ADC Auto Trigger Enable)

このビットが1を書かれると、A/D変換の自動起動が許可されます。A/D変換器は選んだ起動信号の上昇端で変換を開始します。この起動元はA/D変換制御/状態レジスタB(ADCSR)のA/D変換起動要因選択(ADTS2~0)ビット設定によって選ばれます。

●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、このフラグが設定(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(ADIF)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ベクタを実行する時にADIF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(リード モディファイ ライト)を行うと、保留中の割り込みが禁止され得ることに注意してください。これはSBI,CBI命令が使われる場合にも適用されます(訳注:アドレス範囲外のため、本行は不適切です)。

●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28-5. 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

28.9.3. ADCSRB – A/D制御/状態レジスタB (ADC Control and Status Register B)

名称 : ADCSRB

変位 : \$7B

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	ACME	-	-	-	ADTS2~0		
アクセス種別	R	R/W	R	R	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

●ビット6 – 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「アナログ比較器入力選択」をご覧ください。

●ビット2~0 – ADTS2~0 : A/D変換自動起動要因選択 (ADC Auto Trigger Source)

A/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットが1を書かれると、これらのビット値はどの起動元がA/D変換を起動するかを選びます。ADATEが解除(0)されると、ADTS2~0設定は無効です。変換は選んだ割り込みフラグの上昇端によって起動されます。解除(0)されている起動元から設定(1)されている起動元への切り替えが、起動信号上に上昇端を生成することに注意してください。ADCSRAのA/D許可(ADEN)ビットが設定(1)されているなら、これが変換を開始させます。連続変換動作(ADTS2~0=0)への切り替えは、例えばA/D変換完了割り込み要求フラグが設定(1)されていても、起動事象を引き起こしません。

表28-6. A/D変換自動起動元選択

ADTS2~0	起動元
0 0 0	連続変換動作
0 0 1	アナログ比較器
0 1 0	外部割り込み要求0
0 1 1	タイマ/カウンタ0比較A一致
1 0 0	タイマ/カウンタ0溢れ
1 0 1	タイマ/カウンタ1比較B一致
1 1 0	タイマ/カウンタ1溢れ
1 1 1	タイマ/カウンタ1捕獲要求

### 28.9.4. ADCL – A/D変換データレジスタ下位 (ADC Data Register Low) [ADLAR=0]

A/D変換が完了すると、その結果がADCHとADCLの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が先に、その後にADCHが読まれなければなりません。

A/D多重器選択レジスタ(ADMUX)の左揃え選択(ADLAR)ビットとA/Dチャネル選択(MUX3~0)ビットはこのレジ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)ならば結果は左揃えにされます。ADLARが解除(0:既定)ならば結果は右揃えにされます。

名称 : ADCL  
 変位 : \$78  
 リセット : \$00  
 特質 : ADLAR=0

ビット	7	6	5	4	3	2	1	0
	ADC7~0							
アクセス種別	R	R	R	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – ADC7~0 : A/D変換結果 (ADC Conversion result)

これらのビットは変換での結果を表します。詳細については「[A/D変換の結果](#)」を参照してください。

### 28.9.5. ADCH – A/D変換データレジスタ上位 (ADC Data Register High) [ADLAR=0]

名称 : ADCH  
 変位 : \$79  
 リセット : \$00  
 特質 : ADLAR=0

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ADC9,8	
アクセス種別	R	R	R	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

#### ● ビット1,0 – ADC9,8 : A/D変換結果 (ADC Conversion result)

A/D変換データレジスタ下位(ADCL)を参照してください。

### 28.9.6. ADCL – A/D変換データレジスタ下位 (ADC Data Register Low) [ADLAR=1]

名称 : ADCL  
 変位 : \$78  
 リセット : \$00  
 特質 : ADLAR=1

ビット	7	6	5	4	3	2	1	0
	ADC1,0		-	-	-	-	-	-
アクセス種別	R	R	R	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7,6 – ADC1,0 : A/D変換結果 (ADC Conversion result)

A/D変換データレジスタ下位(ADCL)を参照してください。

### 28.9.7. ADCH – A/D変換データレジスタ上位 (ADC Data Register High) [ADLAR=1]

名称 : ADCH  
 変位 : \$79  
 リセット : \$00  
 特質 : ADLAR=1

ビット	7	6	5	4	3	2	1	0
	ADC9~2							
アクセス種別	R	R	R	R	R	R	R	R
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – ADC9~2 : A/D変換結果 (ADC Conversion result)

A/D変換データレジスタ下位(ADCL)を参照してください。

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

各々のビットは論理1を書かれると、対応するADCnピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、対応するポート入力レジスタのビット(PINx)は常に0として読みます。アナログ信号がADCnピンに印加され、そのピンからのデジタル入力が必要とされない時にデジタル入力緩衝部での消費電力を削減するため、そのビットは論理1を書かれるべきです。

名称 : DIDR0

変位 : \$7E

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D
アクセス種別	R	R	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット5 – ADC5 : ADC5デジタル入力禁止 (ADC5 Digital Input Disable)
- ビット4 – ADC4 : ADC4デジタル入力禁止 (ADC4 Digital Input Disable)
- ビット3 – ADC3 : ADC3デジタル入力禁止 (ADC3 Digital Input Disable)
- ビット2 – ADC2 : ADC2デジタル入力禁止 (ADC2 Digital Input Disable)
- ビット1 – ADC1 : ADC1デジタル入力禁止 (ADC1 Digital Input Disable)
- ビット0 – ADC0 : ADC0デジタル入力禁止 (ADC0 Digital Input Disable)

## 29. DBG – デバッグWIRE 内蔵デバッグ システム (debugWIRE On-chip Debug System)

### 29.1. 特徴

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

### 29.2. 概要

デバッグWIRE内蔵デバッグ システムはプログラムの流れを制御してCPUでのAVR命令を実行し、各種不揮発性メモリをプログラミングするのに双方向インターフェースの線を使います。

### 29.3. 物理インターフェース

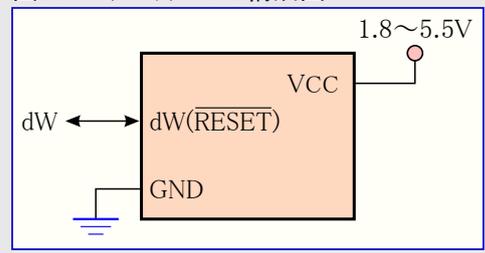
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

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

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

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

図29-1. デバッグWIRE構成図



### 29.4. ソフトウェア中断点 (ブレークポイント)

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

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

### 29.5. デバッグWIREの制限

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

デバッグWIREシステムはSPI部とシステムクロックを共用します。従って電力削減レジスタ(PRR)のPRSPIビットはデバッグ時に設定(1)されてはなりません。PRSPIビットの設定(1)はデバッグWIRE部へのクロックを禁止し、デバイスの固着を引き起こすかもしれません。

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

## 29.6. デバッグWIRE用レジスタ

次項はデバッグWIREで使うレジスタを記述します。

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : DWDR

変位 : \$51 (\$31)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$31です。

ビット	7	6	5	4	3	2	1	0
	DWDR7~0							
アクセス種別	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7~0 – DWDR7~0 : デバッグWIREデータ (deBugWire Data)

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

## 30. フラッシュメモリの自己プログラミング – ATmega48/V

### 30.1. 概要

ATmega48/Vでは書き込み中の読み出し可能(Read-While-Write)の支援と独立したブートローダ領域がありません。SPM命令はフラッシュメモリ全体で実行することができます。

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

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

手段1 (ページ消去前の一時緩衝部格納)

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

手段2 (ページ消去後の一時緩衝部格納)

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

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

#### 30.1.1. SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタ(R31:R30)にアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'0000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGE(Z11~6)に書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- ページ消去中、CPUは停止されます。

**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SPMCSRへ書く前に割り込みを禁止すべきです。

#### 30.1.2. ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタ(R31:R30)にアドレス、R1:R0にデータを設定してSPMCSRに'0000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORD(Z5~1)の内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのRWWSREビット(=1)書き込みによって自動的に消去されます。システムリセット後も消去されています。一時緩衝部を消去せずに各アドレスへ複数回書くことができないことに注意してください。

SPMページ設定操作の途中でEEPROMが書かれると、設定した全データが失われます。

#### 30.1.3. ページ書き込みの実行

ページ書き込みを行うにはZポインタ(R31:R30)にアドレスを設定してSPMCSRに'0000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGE(Z11~6)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。

- ページ書き込み中、CPUは停止されます。

### 30.2. 自己プログラミング中のフラッシュメモリのアドレス指定

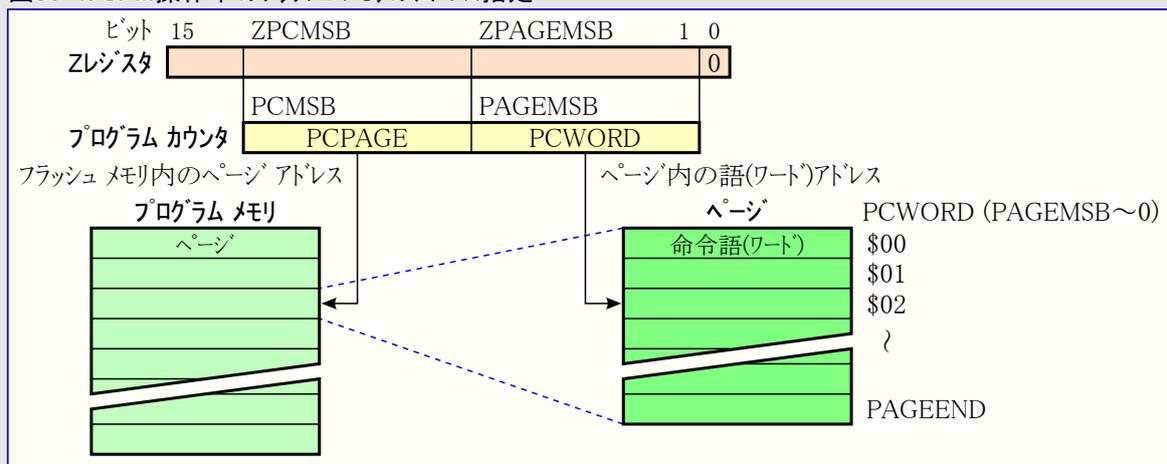
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。

ビット	15	14	13	12	11	10	9	8
ZH (R31)	Z15	Z14	Z13	Z12	Z11	Z10	Z9	Z8
ビット	7	6	5	4	3	2	1	0
ZL (R30)	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

フラッシュメモリがページで構成されるため(「メモリプログラミング」章の「ページ容量」参照)、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つように扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは次図で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってソフトウェアはページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。

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

図30-1. SPM操作中のフラッシュメモリのアドレス指定



注: 本図で使用した各変数は「メモリプログラミング」章の「ページ容量」で一覧されます。

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

EEPROM書き込み動作がフラッシュメモリへの全ソフトウェアプログラミングを妨げることに注意してください。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタのEEPROMプログラム許可(EPCR.EEPE)ビットを検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前にこのビットが解除(0)されているのを確認することが推奨されます。

### 30.2.2. ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビット(LB)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、SPMCSRのSPM操作許可(SPMCSR.SPMEN)とブート施錠ビット設定(SPMCSR.BLBSET)ビットを設定(1)してください。SPMENとBLBSETビットがSPMCSRに設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SPMENとBLBSETビットは施錠ビット読み出しの完了、または3 CPU周期内にLPM命令が実行されないか、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SPMENとBLBSETビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

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

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

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様にヒューズ上位ビット(FHB)を読むにはZポインタに\$0003を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるようにヒューズ上位ビット(FHB)の値が転送先レジスタに格納されます。ヒューズ上位ビットの配置と詳細な記述については「メモリプログラミング」章のヒューズ上位バイトの表を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

同様に拡張ヒューズビット(EFB)を読むにはZポインタに\$0002を設定してください。SPMENとBLBSETビットがSPMCSRで設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるように拡張ヒューズビット(EFB)の値が転送先レジスタに格納されます。拡張ヒューズビットの配置と詳細な記述については「メモリプログラミング」章の拡張ヒューズバイトの表を参照してください。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	-	-	EFB0

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

関連リンク [202頁の「ヒューズビット」](#)  
[203頁の「ヒューズのラッチ」](#)

### 30.2.3. フラッシュメモリデータ化けの防止

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

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

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

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

### 30.2.4. 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。次表はCPUからのフラッシュメモリアクセスに対する代表的なプログラミング時間を示します。

表30-1. SPM命令によるフラッシュメモリのプログラミング時間

項目	最小	最大
SPM命令によるフラッシュ書き込み (ページ消去、ページ書き込み)	3.2ms	3.4ms

注: 最小と最大の時間は(項目の)個別操作毎に対してです。

### 30.2.5. アセンブリ言語による簡単なブートローダ例 - ATmega48/V

ATmega48/VではSPMCSRのRWWSBビットが常に0として読まれることに注意してください。それでも書き中読み可(Read-While-Write)を支援するデバイスとの互換性を保証するため、コード例で示されるようにこのビットを検査することが推奨されます。

このルーチンはRAMからフラッシュメモリへ1ページのデータを書きます。RAM内の最初のデータ位置はYレジスタによって指示され、フラッシュメモリ内の最初のデータ位置はZレジスタによって指示されます。異常処理は含まれません。使用レジスタはR0,R1,TMP,CNTL,CNTH,SPMCでレジスタの保存と復帰はこのルーチン内に含まれず、レジスタ使用はコード量を犠牲にすれば最適化できます。割り込みが禁止される前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更となります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令	注釈
	. EQU PGSZB = PAGESIZE*2	; PGSZBはページ内のバイト数です。(PAGESIZEは語(ワード)数)
	. ORG SMALLBOOTSTART	;
WRPG:	LDI SPMC, (1<<PGERS)+(1<<SPMEN)	; [ ページ消去 ]
	RCALL SPMJ	; ページ消去SPMCSR値を取得
		; ページ消去
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN)	; [ RWW領域読み出し再許可 ]
	RCALL SPMJ	; RWW領域読み出し許可SPMCSR値を取得
		; RWW領域読み出し許可
	LDI CNTL, LOW(PGSZB)	; [ RAMからフラッシュページ一時緩衝部へ転送 ]
	LDI CNTH, HIGH(PGSZB)	; バイト計数器を初期化
WLP:	LD R0, Y+	; (削除)
	LD R1, Y+	; RAM上の下位データを取得(ポインタ進行)
	LDI SPMC, (1<<SPMEN)	; RAM上の上位データを取得(ポインタ進行)
	RCALL SPMJ	; ページ一時緩衝部書き込みSPMCSR値を取得
	ADIW ZH:ZL, 2	; ページ一時緩衝部書き込みSPMCSR値を取得
	SBIW CNTH:CNTH, 2	; 対応語(ワード)データをページ一時緩衝部に設定
	BRNE WLP	; 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 ZL, LOW(PGSZB)	; [ ページ書き込み ]
	SBCI ZH, HIGH(PGSZB)	; ページ一時緩衝部先頭にポインタを復帰
	LDI SPMC, (1<<PGWRT)+(1<<SPMEN)	; (削除)
	RCALL SPMJ	; フラッシュ書き込みSPMCSR値を取得
		; フラッシュメモリページ書き込み
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN)	; [ RWW領域読み出し再許可 ]
	RCALL SPMJ	; RWW領域読み出し許可SPMCSR値を取得
		; RWW領域読み出し許可
		; [ 読み戻し照合(任意) ]
	LDI CNTL, LOW(PGSZB)	; バイト計数器を初期化
	LDI CNTH, HIGH(PGSZB)	; (削除)
	SUBI YL, LOW(PGSZB)	; RAMデータ先頭にポインタを復帰
	SBCI YH, HIGH(PGSZB)	;
RLP:	LPM R0, Z+	; フラッシュメモリから1バイト取得(ポインタ進行)
	LD R1, Y+	; RAMから1バイトデータ取得(ポインタ進行)
	CPSE R0, R1	; 値一致でスキップ
	RJMP ERROR	; 不一致で異常処理へ
;		;
	SBIW CNTH:CNTH, 1	; 計数器を減数 (SUBI)
	BRNE RLP	; 指定バイト数分継続
:		; [ RWW領域へ復帰 ]
RTN:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRS TMP, RWWSB	; RWW領域多忙でスキップ
	RET	; 準備可で呼び出し元へ復帰
;		; [ RWW領域読み出し再許可 ]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN)	; RWW領域読み出し許可SPMCSR値を取得
	RCALL SPMJ	; RWW領域読み出し許可
	RJMP RTN	; RWW領域準備可まで待機へ
		; [ SPM命令実行サブルーチン ]
SPMJ:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRC TMP, SPMEN	; 操作可能(直前のSPM完了)でスキップ
	RJMP SPMJ	; 操作可まで待機
;		;
	IN TMP, SREG	; ステータスレジスタ値を保存
WAIT:	CLI	; 全割り込み禁止
	SBIC EECR, EEPE	; EEPROMプログラミング中以外でスキップ
	RJMP WAIT	; EEPROMプログラミング完了まで待機
;		;
	OUT SPMCSR, SPMC	; SPM動作指定
	SPM	; 対応SPM動作実行
	OUT SREG, TMP	; ステータスレジスタ値を復帰
	RET	; 呼び出し元へ復帰

## 30.3. 自己プログラミング用レジスタ – ATmega48/V

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

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

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPMCSR

変位 : \$57 (\$37)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$37です。

ビット	7	6	5	4	3	2	1	0
	SPMIE	RWWSB	(SIGRD)	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN
アクセス種別	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIEビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みはSPMCSRのSPM操作許可(SPMCSR.SPMEN)ビットが解除(0)されている限り実行されます。EEPROM書き込みとSPM命令の間、本割り込みは生成されません。

#### ● ビット6 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)

このビットは書き中読み可(Read-While-Write)を支援するデバイスとの互換用です。これは常に0として読みます。

#### ● ビット5 – SIGRD : 識票列読み出し (Signature Row Read)

このビットがSPMENと同時に1を書かれると、次の3クロック周期内のLPM命令は識票列からバイトを転送先レジスタに読みます。「ソフトウェアからの識票列読み出し」を参照してください。SIGRDとSPMENが設定(1)された後の4クロック周期内のSPM命令は無効です。この操作は将来の使用に対して予約されており、使われるべきではありません。

#### ● ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

ATmega48/Vでの本ビットの機能はATmega88/VとATmega168/Vでの機能の部分集合です。ページ一時緩衝部を満たしている間にRWWSREビットが1を書かれると、ページ一時緩衝部は消去され、データが失われます。

#### ● ビット3 – BLBSET : ブート施錠ビット設定 (Boot Lock Bits Set)

ATmega48/Vでの本ビットの機能はATmega88/VとATmega168/Vでの機能の部分集合です。SPMCSRでBLBSETとSPMENが設定(1)された後の3クロック周期内のLPM命令は(Zポイントの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。「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」を参照してください。

#### ● ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポイントの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ全体の書き込み動作中、CPUは停止されます。

#### ● ビット1 – PGERS : ページ消去 (Page Erase)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポイントの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ全体の消去中、CPUは停止されます。

#### ● ビット0 – SPMEN : SPM操作許可 (Store Program Memory Enable)

このビットは次の4クロック周期間SPM命令を許可します。本ビットが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ポイントでアドレス指定したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SPMENビットはSPM命令の完了、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ消去とページ書き込み中、SPMENビットはその動作が完了されるまで1に留まります。

下位5ビットに10001, 01001, 00101, 00011, 00001以外のどんな組み合わせを書いても無効です。

## 31. BTLDR – ブート ロード 支援 – 書き込み中読み出し可能な自己プログラミング – ATmega88/V/168/V

### 31.1. 特徴

- 書き込み中読める(Read-While-Write)自己プログラミング
- 柔軟性のあるブート ロード メモリ容量
- 高い安全性 (柔軟な保護用の独立したブート施錠ビット)
- リセット ベクタ選択用の独立したヒューズ
- 最適化されたページ容量 (注1)
- 効率的なコード手法
- 効率的な読み-変更-書き(リード モデファイ ライト)支援

**注1:** ページはプログラミング中に使われる多数のバイトから成るフラッシュ メモリの区画です(「ページ容量」のフラッシュ メモリのページ数とページの語数の表をご覧ください)。このページ構成は通常動作に影響を及ぼしません。

**関連リンク** 203頁の「ページ容量」

### 31.2. 概要

本デバイスに於いて、ブート ロード 支援はMCU自身によるプログラム コードのダウンロードとアップロード用の真の書き込み中の読み出しが可能な自己プログラミング機構を提供します。この特徴はフラッシュ メモリに常駐するブート ロード 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 ロード プログラムはフラッシュ メモリ内にコードを書き(プログラム)、コードを読み、またはプログラム メモリからコードを読むのに、利用可能なデータ インターフェースと関連する規約のどれもが使えます。ブート ロード 領域内のプログラム コードはブート ロード 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 ロード 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 ロード メモリの容量はヒューズで設定可能で、ブート ロード 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。

### 31.3. フラッシュ メモリの応用領域とブート ロード 領域

フラッシュ メモリは応用領域とブート ロード 領域の2つの主な領域で構成されます。各領域の容量は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。

#### 31.3.1. 応用領域

応用領域は応用コードを格納するのに使われるフラッシュ メモリの領域です。応用領域用保護基準は応用ブート施錠ビット(ブート施錠ビット0)によって選べます。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 ロード コードも決して格納し得ません。

#### 31.3.2. ブート ロード 領域 (BLS)

応用領域が応用コード格納用に使われるのに対して、SPM命令はBLSから実行する時にだけプログラミングを始められるので、ブート ロード ソフトウェアはBLSに格納されなければなりません。SPM命令はBLS自身を含む全てのフラッシュ メモリにアクセスできます。ブート ロード 領域用保護基準はブート ロード 施錠ビット(ブート施錠ビット1)によって選べます。

### 31.4. フラッシュ メモリの書き込み中に読み出し可能な領域と不能な領域

どちらのアドレスがプログラミングされるかによって、CPUが書き込み中の読み出しを支援するか、ブート ロード ソフトウェアが更新中にCPUが停止されるかのどちらです。上で記述されるようなBOOTSZヒューズによって設定可能な2つの領域に加え、フラッシュ メモリは書き込み中読み出し可能な(RWW)領域と書き込み中読み出し不能な(NRWW)領域の2つの固定領域にも分けられます。RWWとNRWW領域間の境界は「ブート ロード パラメータ」項と図31-2.で与えられます。この2つの領域間の主な違いを次に示します。

- RWW領域側に配置されたページを消去または書くとき、NRWW領域はその動作中に読むことができます。
- NRWW領域側に配置されたページを消去または書くとき、その全ての動作中にCPUは停止されます。

ブート ロード ソフトウェア動作中、使用者ソフトウェアはRWW領域側に配置されたどのコードも決して読めません。「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 ロード 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。

**関連リンク** 198頁の「ブート ロード パラメータ」

**(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。

### 31.4.1. RWW – 書き込み中読み出し可能領域

ブートローダソフトウェア更新がRWW領域側のページをプログラミングする場合、フラッシュメモリからコードを読むことが可能ですが、NRWW領域に配置されるコードだけです。プログラミング実行中、そのソフトウェアはRWW領域が決して読まれないことを保証しなければなりません。使用者ソフトウェアがプログラミング中に(例えば、CALL、JMP、LPM系命令または割り込みによって)RWW領域側に配置されるコードを読むと、そのソフトウェアは未知の状態へ行き着くかもしれません。これを避けるために割り込みは禁止またはブートローダ領域へ移動のどちらかにされるべきです。ブートローダ領域は常にNRWW領域に配置されます。RWW領域が読み出しに対して妨げられている限り、SPM命令制御/状態レジスタ(SPMCSR)のRWW領域多忙(RWWSB)ビットは論理1として読みます。プログラミングが完了した後、RWW領域に配置したコードを読む前にRWWSBはソフトウェアによって解除(0)されなければなりません。RWWSBを解除(0)する方法の詳細については本章内の「SPMCSR – SPM命令制御/状態レジスタ」をご覧ください。

### 31.4.2. NRWW – 書き込み中読み出し不能領域

NRWW領域に配置したコードはブートローダソフトウェアがRWW領域内のページを更新する時に読めます。ブートローダコードがNRWW領域を更新するとき、全てのページ消去またはページ書き込み動作中にCPUが停止されます。

表31-1. 書き込み中読み出し可能機能

プログラミング中にZポイントで指定される領域	プログラミング中に読める領域	CPU動作	RWW機能支援
RWW領域	NRWW領域	通常動作	あり
NRWW領域	なし	停止	なし

図31-1. RWW領域とNRWW領域の関係

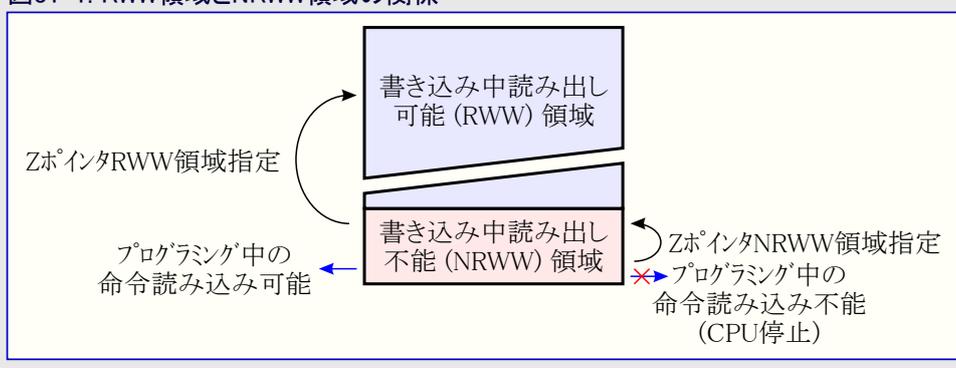
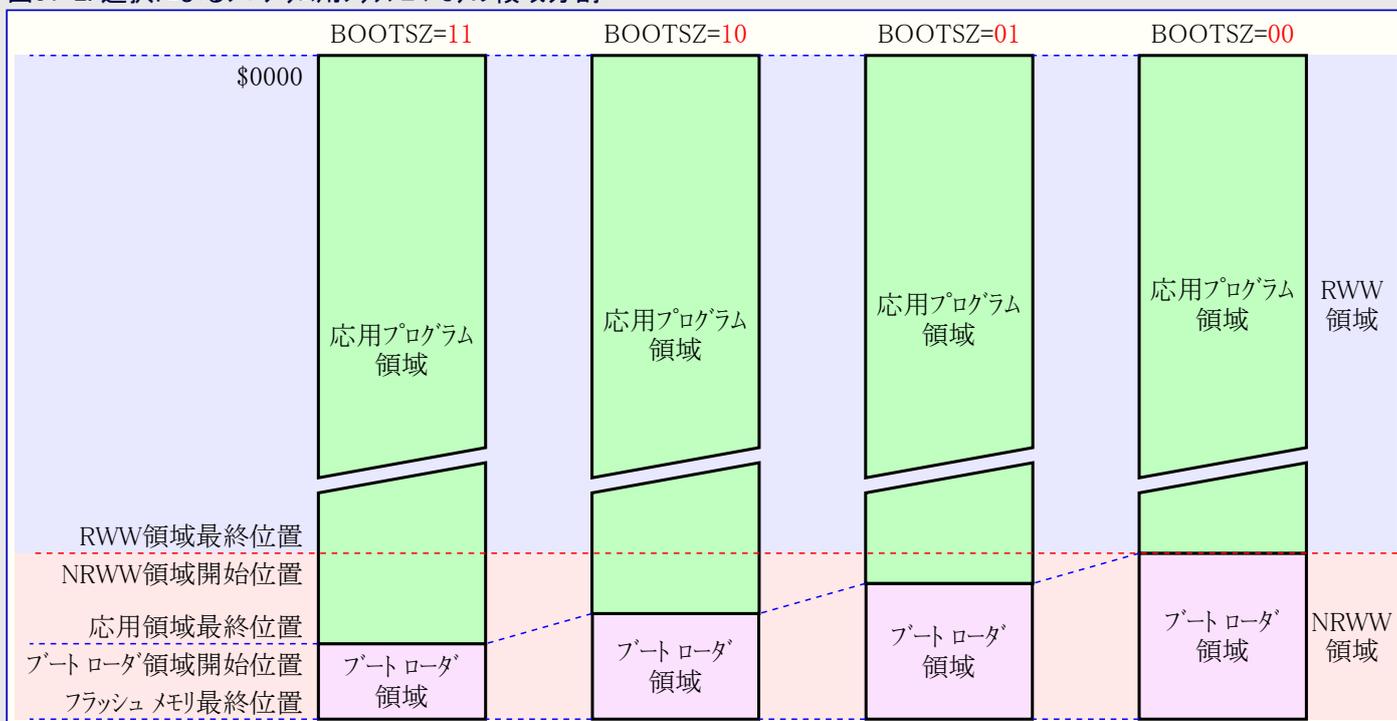


図31-2. 選択によるプログラム用フラッシュメモリの領域分割



関連リンク [198頁の「ブートローダパラメータ」](#)

### 31.5. ブートローダ施錠ビット

ブートローダ能力が必要とされないなら、フラッシュメモリ全体が応用コード用に利用可能です。ブートローダは個別に設定可能な独立した2組のブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

使用者は以下を選べます。

- MCUによって更新するソフトウェアからフラッシュメモリ全体を保護
- MCUによって更新するソフトウェアからフラッシュメモリのブートローダ領域だけを保護
- MCUによって更新するソフトウェアからフラッシュメモリの応用領域だけを保護
- フラッシュメモリ全体で更新するソフトウェアを許可

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31-2. 応用領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB0 動作種別	BLB02	BLB01	保護種別
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注)
4	0	1	ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注)

注: BLB02=0で、割り込みベクタがブートローダ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

表31-3. ブートローダ領域に対する保護種別 (0=プログラム、1=非プログラム)

BLB1 動作種別	BLB12	BLB11	保護種別
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注)
4	0	1	応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注)

注: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

### 31.6. ブートローダプログラムへの移行

ブートローダへの移行は応用プログラムから分岐(Jump)または呼び出し(Call)によって行います。これはUSARTやSPIインターフェース経由で受信した指令のような起点によって始められるかもしれませんが、代わりに、リセット後にリセットベクタがブートローダ領域開始アドレスを指示するようにブートリセット(BOOTRST)ヒューズをプログラム(0)することができます。この場合、ブートローダがリセット後に開始されます。応用コードが設定された(書かれた)後、そのプログラム(ブートローダ)は応用コードの実行を始めることができます。このヒューズはMCU自身によって変更できません。これは一旦ブートリセットヒューズがプログラム(0)されると、リセットベクタは常にブートローダリセットを指示し、このヒューズが直列プログラミングまたは並列プログラミングを通してのみ変更できることを意味します。

表31-4. ブートリセットヒューズ (0=プログラム、1=非プログラム)

BOOTRST	リセット後実行開始アドレス (リセットベクタ)	
0	ブートローダリセット	ブートローダ開始アドレス (「ブートローダパラメータ」で記述されるように)
1	応用リセット	\$0000

### 31.7. 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

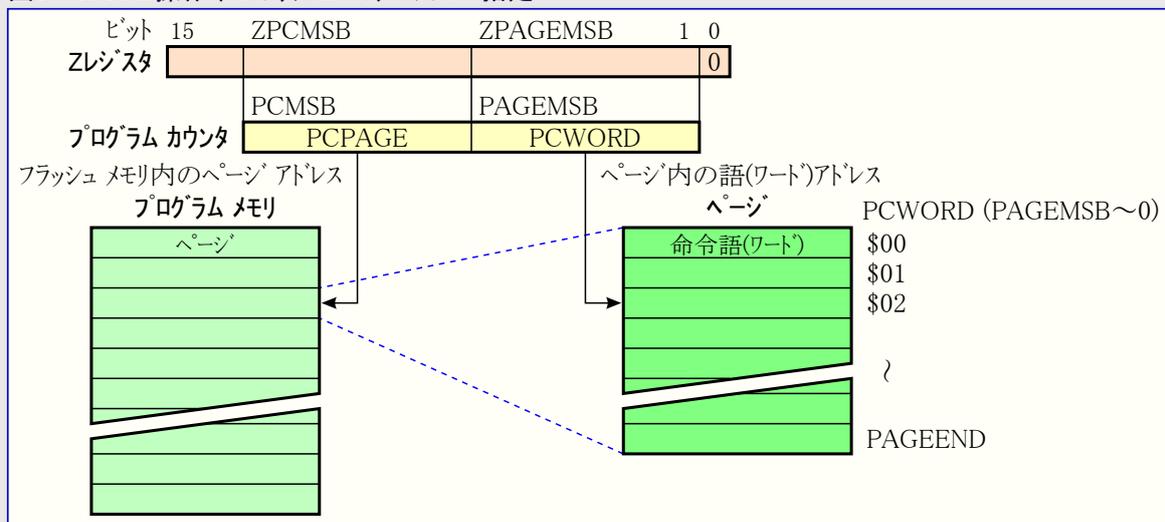
Zポイント(レジスタ)はSPM命令でのアドレス指定に使われます。

ビット	15	14	13	12	11	10	9	8
ZH (R31)	Z15	Z14	Z13	Z12	Z11	Z10	Z9	Z8
ビット	7	6	5	4	3	2	1	0
ZL (R30)	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

フラッシュメモリがページで構成されるため、プログラムカウンタ(アドレスポインタ)は2つの違う領域を持つように扱われます。1つの領域は下位側ビットから成り、ページ内の語(ワード)をアドレス指定し、一方上位側ビットはそのページをアドレス指定します。これは次図で示されます。ページ消去とページ書き込み操作が個別にアドレス指定されることに注意してください。従ってブートローダソフトウェアはページ消去とページ書き込み操作の両方で同じページをアドレス指定することが最も重要です。一旦プログラミング操作が開始されると、このアドレスはラッチされ、Zポイントは他の操作に使えます。

Zポイントを使わない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ポイントを使います。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ポイントの最下位ビット(Z0)も使われます。

図31-3. SPM操作中のフラッシュメモリのアドレス指定



注: 図内で使った各変数は表31-9.で一覧されます。

### 31.8. フラッシュメモリの自己プログラミング

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

手段1 (ページ消去前の一時緩衝部格納)

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

手段2 (ページ消去後の一時緩衝部格納)

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

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

### 31.8.1. SPM命令によるページ消去の実行

ページ消去を実行するにはZポインタにアドレスを設定してSPM命令制御/状態レジスタ(SPMCSR)に'X0000011'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスはZポインタのPCPAGEに書かれなければなりません。この操作中、Zポインタの他のビットは無視されます。

- RWW領域のページ消去 : ページ消去中、NRWW領域は読めます。
- NRWW領域のページ消去 : ページ消去中、CPUは停止されます。

### 31.8.2. ページ一時緩衝部の設定 (ページ設定)

命令語(ワード)を(ページ一時緩衝部に)書くにはZポインタにアドレス、R1:R0にデータを設定してSPMCSRに'00000001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。ZポインタのPCWORD(Z5~1)の内容は一時緩衝部のデータのアドレスに使われます。一時緩衝部はページ書き込み操作後、またはSPMCSRのRWWSREビット(SPMCSR.RWWSRE)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はできません。

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。

### 31.8.3. ページ書き込みの実行

ページ書き込みを行うにはZポインタにアドレスを設定してSPMCSRに'X0000101'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。R1とR0のデータは無視されます。ページアドレスは(Zポインタの)PCPAGE(Z12~6)に書かれなければなりません。この操作中にZポインタの他のビットは0を書かれなければなりません。

- RWW領域のページ書き込み : ページ書き込み中、NRWW領域は読めます。
- NRWW領域のページ書き込み : ページ書き込み中、CPUは停止されます。

### 31.8.4. SPM操作可割り込みの使用方法

SPM操作可割り込みが許可されると、SPMCSRのSPMENビット(SPMCSR.SPMEN)が解除(0)されている時にSPM操作可割り込みが継続的に発生します。これはソフトウェアでSPMCSRをポーリングする代わりにこの割り込みが使えることを意味します。SPM操作可割り込みを使うとき、割り込みが読み出しに対して防がれる時にRWW領域をアクセスするのを避けるために、割り込みベクタはブートローダ領域(BLS)へ移動されるべきです。割り込み(ベクタ)の移動法は「INT - 割り込み」章で記述されます。

関連リンク 45頁の「INT - 割り込み」

### 31.8.5. ブートローダ領域(BLS)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(BLS)に更新を許す場合、特別な注意が払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。

### 31.8.6. 自己プログラミング中のRWW領域読み込みの防止

自己プログラミング中(ページ消去もページ書き込みも)、RWW領域は読み出しに対して常に防がれます。使用者ソフトウェアそれ自身が自己プログラミング操作中にこの領域がアドレス指定されるのを防止しなければなりません。SPMCSRのRWWSB(SPMCSR.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「INT - 割り込み」章で記述されるようにブートローダ領域(BLS)へ移動されるべきか、または割り込みが禁止されなければなりません。プログラミングが完了した後にRWW領域をアドレス指定する前に、使用者ソフトウェアはSPMCSR.RWWSREの書き込みによってSPMCSR.RWWSBを解除(0)しなければなりません。例については「7セプティ言語による簡単なブートローダ例」を参照してください。

関連リンク 45頁の「INT - 割り込み」

### 31.8.7. SPM命令によるブートローダ施錠ビットと一般施錠ビットの設定

ブートローダ施錠ビットと一般施錠ビットを設定(0)するには希望したデータをR0に設定してSPMCSRに'X0001001'を書き、SPMCSR書き込み後4クロック周期内にSPM命令を実行してください。

ビット	7	6	5	4	3	2	1	0
R0	1	1	BLB12	BLB11	BLB02	BLB01	LB2	LB1

「ブートローダ施錠ビット」項の表はフラッシュメモリのアクセスに影響を及ぼすブートローダ施錠ビットの各種設定法を示します。

R0のビット5~0が解除(0)される場合、SPMCSRでSPMENビット(SPMCSR.SPMEN)とブート施錠ビット設定(BLBSET)ビット(SPMCSR.BLBSET)が設定(1)された後の4クロック周期内にSPM命令が実行されると、対応する施錠ビットがプログラム(0)されます。この操作中、Zポインタは関係ありませんが、将来との共通性のため、(施錠ビット読み出しに使われるのと同じ)\$0001でZポインタを設定することが推奨されます。将来との共通性のため、施錠ビット書き込み時に、R0のビット7,6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。

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

EEPROM書き込み動作はフラッシュメモリへの全ソフトウェアプログラミングを妨げます。ソフトウェアからのヒューズと施錠ビット読み出しもEEPROM書き込み動作中、妨げられます。使用者はEEPROM制御レジスタ(EECR)のEEPROMプログラム許可(EEPE)ビット(EECR.EEPE)を検査し、SPM命令制御/状態レジスタ(SPMCSR)へ書く前にこのビットが解除(0)されているのを確認することが推奨されます。

### 31.8.9. ソフトウェアからのヒューズビットと施錠ビットの読み出し

ソフトウェアからヒューズと施錠ビット(LB)の両方を読むことができます。施錠ビットを読むにはZポインタに\$0001を設定してSPMCSRのSPMEN(SPMCSR.SPMEN)とブート施錠ビット設定(BLBSET)(SPMCSR.BLBSET)のビットを設定(1)してください。SPMCSRでSPMEN(SPMCSR.SPMEN)とBLBSET(SPMCSR.BLBSET)のビットが設定された後、3 CPU周期内にLPM命令が実行されると、施錠ビットの値が転送先レジスタに格納されます。SPMCSR.SPMENとSPMCSR.BLBSETのビットは施錠ビット読み出しの完了で、または3 CPU周期内にLPM命令が実行されないか、または4 CPU周期内にSPM命令が実行されない場合、自動的に解除(0)されます。SPMCSR.SPMENとSPMCSR.BLBSETのビットが解除(0)されると、LPMは命令一式手引書で記述されるように動作します。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	BLB12	BLB11	BLB02	BLB01	LB2	LB1

ヒューズ下位ビット(FLB)を読む手順は上記の施錠ビット読み出しと同様です。ヒューズ下位ビットを読み出すにはZポインタに\$0000を設定してSPMCSRのSPMEN(SPMCSR.SPMEN)とBLBSET(SPMCSR.BLBSET)のビットを設定(1)してください。SPMCSR.SPMENとSPMCSR.BLBSETのビットが設定された後、3 CPU周期内にLPM命令が実行されると、以下で示されるようにヒューズ下位ビット(FLB)の値が転送先レジスタに格納されます。

ビット	7	6	5	4	3	2	1	0
Rd	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

同様に、ヒューズ上位ビット(FHB)を読むにはZポインタに\$0003を設定してください。SPMCSR.SPMENとSPMCSR.BLBSETのビットが設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるようにヒューズ上位ビット(FHB)の値が転送先レジスタに格納されます。

ビット	7	6	5	4	3	2	1	0
Rd	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

拡張ヒューズビット(EFB)を読む時はZポインタに\$0002を設定してください。SPMCSR.SPMENとSPMCSR.BLBSETのビットが設定(1)された後、3周期内にLPM命令が実行されると、以下で示されるように拡張ヒューズビット(EFB)の値が転送先レジスタに格納されます。

ビット	7	6	5	4	3	2	1	0
Rd	-	-	-	-	-	EFB2	EFB1	EFB0

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

関連リンク [202頁の「ヒューズビット」](#)

### 31.8.10. ソフトウェアからの識票列読み出し

ソフトウェアから識票列を読むには右表で与えられる識票バイトアドレスをZポインタに設定し、SPMCSRで識票列読み出し(SIGRD)(SPMCSR.SIGRD)とSPMEN(SPMCSR.SPMEN)のビットを設定(1)してください。SPMCSR.SIGRDとSPMCSR.SPMENのビットが設定された後、3 CPU周期内にLPM命令が実行されると、識票バイト値が転送先レジスタに格納されます。SPMCSR.SIGRDとSPMCSR.SPMENのビットは識票バイト読み出しの完了、または3 CPU周期内にLPM命令が実行されない場合、自動的に解除(0)されます。SPMCSR.SIGRDとSPMCSR.SPMENのビットが解除(0)されると、LPMはAVR命令一式説明で記述されるように動作します。

表31-5. 識票列アドレス一覧

識票バイト	Zポインタアドレス
デバイス識票バイト 1	\$0000
デバイス識票バイト 2	\$0002
デバイス識票バイト 3	\$0004
RC発振器校正値	\$0001

注: 他の全てのアドレスは将来の使用に對して予約されています。

### 31.8.11. フラッシュメモリデータ化けの防止

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

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

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

- そのシステムでファームウェア更新が必要ない場合、どんなファームウェア更新をも防ぐためにファームウェア施錠ビットをプログラム(0)してください。
- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアをパワーダウン休止動作に保ってください。これはCPUが命令の復号と実行の試みを防ぎ、SPMCSR、従ってフラッシュメモリを予期せぬ書き込みから効果的に保護します。

### 31.8.12. SPM命令使用時のフラッシュメモリ用プログラミング(書き込み)時間

校正された内蔵RC発振器がフラッシュメモリアクセス時間に使われます。次表はCPUからのフラッシュメモリアクセスに対する代表的なプログラミング時間を示します。

表31-6. SPM命令によるフラッシュメモリのプログラミング時間

項目	最小	最大
SPM命令によるフラッシュ書き込み (ページ消去、ページ書き込み、施錠ビット書き込み)	3.2ms	3.4ms

注: 最小と最大の時間は(項目の)個別操作毎に対してです。

### 31.8.13. アセンブリ言語による簡単なブートローダ例 - ATmega88/V/168/V

このルーチンはRAMからフラッシュメモリへ1ページのデータを書きます。RAM内の最初のデータ位置はYレジスタによって指示され、フラッシュメモリ内の最初のデータ位置はZレジスタによって指示されます。異常処理は含まれません。このルーチン(少なくともSPMJサブルーチン)はブートローダ領域側に配置されなければなりません。NRWW領域側のコードだけが自己プログラミング(ページ消去とページ書き込み)中に読めます。使用レジスタはR0,R1,TMP,CNTL,CNTH,SPMCで、レジスタの保存と復帰はこのルーチン内に含まれず、使用レジスタはコード量を犠牲にすれば最適化できます。割り込み表がブートローダ領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更になります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令	注釈
	.EQU PGSZB = PAGESIZE*2	; PGSZBはページ内のバイト数です。(PAGESIZEは語(ワード)数)
	.ORG SMALLBOOTSTART	;
WRPG:	LDI SPMC, (1<<PGERS)+(1<<SPMEN)	; [ページ消去]
	RCALL SPMJ	; ページ消去SPMCSR値を取得
		; ページ消去
		; [RWW領域読み出し再許可]
	LDI SPMC, (1<<RWWSRE)+(1<<SPMEN)	; RWW領域読み出し許可SPMCSR値を取得
	RCALL SPMJ	; RWW領域読み出し許可
		; [RAMからフラッシュページ一時緩衝部へ転送]
		; バイト計数器を初期化
		; (削除)
WLP:	LDI CNTL, LOW(PGSZB)	; RAM上の下位データを取得(ポインタ進行)
	LDI CNTH, HIGH(PGSZB)	; RAM上の上位データを取得(ポインタ進行)
	LD R0, Y+	; ページ一時緩衝部書き込みSPMCSR値を取得
	LD R1, Y+	; ページ一時緩衝部書き込みSPMCSR値を取得
	LDI SPMC, (1<<SPMEN)	; 対応語(ワード)データをページ一時緩衝部に設定
	RCALL SPMJ	; ページ一時緩衝部ポインタ進行
	ADIW ZH: ZL, 2	; 計数器を減数 (SUBI)
	SBIW CNTH: CNTL, 2	; 指定バイト数分継続
	BRNE WLP	; [ページ書き込み]
		; ページ一時緩衝部先頭にポインタを復帰
		; (削除)
	SUBI ZL, LOW(PGSZB)	; フラッシュ書き込みSPMCSR値を取得
	SBCI ZH, HIGH(PGSZB)	; フラッシュメモリページ書き込み
	LDI SPMC, (1<<PGWRT)+(1<<SPMEN)	; [RWW領域読み出し再許可]
	RCALL SPMJ	; RWW領域読み出し許可SPMCSR値を取得
		; RWW領域読み出し許可
		; [読み戻し照合(任意)]
		; バイト計数器を初期化
		; (削除)
	LDI CNTL, LOW(PGSZB)	; RAMデータ先頭にポインタを復帰
	LDI CNTH, HIGH(PGSZB)	;
	SUBI YL, LOW(PGSZB)	;
	SBCI YH, HIGH(PGSZB)	;
RLP:	LPM R0, Z+	; フラッシュメモリから1バイト取得(ポインタ進行)
	LD R1, Y+	; RAMから1バイトデータを取得(ポインタ進行)
	CPSE R0, R1	; 値一致でスキップ
	RJMP ERROR	; 不一致で異常処理へ
		;
		; 計数器を減数 (SUBI)
		; 指定バイト数分継続
		; [RWW領域へ復帰]
RTN:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRS TMP, RWWSB	; RWW領域多忙でスキップ
	RET	; 準備可で呼び出し元へ復帰
		; [RWW領域読み出し再許可]
		; RWW領域読み出し許可SPMCSR値を取得
		; RWW領域読み出し許可
		; RWW領域準備可まで待機へ
		; [SPM命令実行サブルーチン]
SPMJ:	IN TMP, SPMCSR	; SPM命令制御/状態レジスタ値を取得
	SBRC TMP, SPMEN	; 操作可能(直前のSPM完了)でスキップ
	RJMP SPMJ	; 操作可まで待機
		;
		; ステータスレジスタ値を保存
		; 全割り込み禁止
WAIT:	IN TMP, SREG	; EEPROMプログラミング中以外でスキップ
	CLI	; EEPROMプログラミング完了まで待機
	SBIC EECR, EEPE	;
	RJMP WAIT	;
		;
		; 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
		;
	OUT SPMCSR, SPMC	;
	SPM	;
	OUT SREG, TMP	;
	RET	;

### 31.8.14. ブートローダパラメータ

以下の表は自己プログラミングの記述で使ったパラメータが与えられます。

表31-7. 応用領域とブートローダ領域の分割設定 (上段:ATmega88/V, 下段:ATmega168/V)

BOOTSZ1	BOOTSZ0	ブートローダ領域		アドレス範囲		
		容量 (語)	ページ数	応用プログラム領域	ブートローダプログラム領域	ブートリセットアドレス
1	1	128	4	\$0000~\$0F7F	\$0F80~\$0FFF	\$0F80
		128	2	\$0000~\$1F7F	\$1F80~\$1FFF	\$1F80
1	0	256	8	\$0000~\$0EFF	\$0F00~\$0FFF	\$0F00
		256	4	\$0000~\$1EFF	\$1F00~\$1FFF	\$1F00
0	1	512	16	\$0000~\$0DFF	\$0E00~\$0FFF	\$0E00
		512	8	\$0000~\$1DFF	\$1E00~\$1FFF	\$1E00
0	0	1024	32	\$0000~\$0BFF	\$0C00~\$0FFF	\$0C00
		1024	16	\$0000~\$1BFF	\$1C00~\$1FFF	\$1C00

注: 各種BOOTSZヒューズ設定は図31-2.で示されます。

(訳注) 原書の表31-7.(ATmega88/V)と表31-10.(ATmega168/V)は表31-7.として纏めました。

表31-8. RWW領域とNRWW領域の範囲 (上段:ATmega88/V, 下段:ATmega168/V)

領域	ページ数	アドレス範囲
書き込み中読み出し可能(RWW)領域	96	\$0000~\$0BFF
	112	\$0000~\$1BFF
書き込み中読み出し不能(NRWW)領域	32	\$0C00~\$0FFF
	16	\$1C00~\$1FFF

注: これら2つの領域についての詳細に関しては「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」を参照してください。

(訳注) 原書の表31-8.(ATmega88/V)と表31-11.(ATmega168/V)は表31-8.として纏めました。

表31-9. 図31-3.で使った各変数説明とZポイントの配置 (上段:ATmega88/V, 下段:ATmega168/V)

変数名	対応値		意味
	PC	Zポイント(注)	
PCMSB	PC11		プログラムカウンタの最上位ビット。 (プログラムカウンタは12ビット、PC11~0) (プログラムカウンタは13ビット、PC12~0)
	PC12		
PAGEMSB	PC4		1ページ内に使われる 語(ワード)アドレスの最上位ビット。 (ページ内の32語には5ビット PC4~0が必要) (ページ内の64語には6ビット PC5~0が必要)
	PC5		
ZPCMSB		Z12 Z13	PCMSBに配置される(対応する)Zレジスタ内のビット。 (Z0が使われないため、ZPCMSB=PCMSB+1)
ZPAGEMSB		Z5	PAGEMSBに配置される(対応する)Zレジスタ内のビット。 (Z0が使われないため、ZPAGEMSB=PAGEMSB+1)
		Z6	
PCPAGE	PC11~5 PC12~6	Z12~6 Z13~7	プログラムカウンタ ページ アドレス: ページ消去とページ書き込み用のページ選択
PCWORD	PC4~0	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。)
	PC5~0	Z6~1	

注: Z15~13(ATmega88/V), Z15~14(ATmega168/V): 常に無視されます。

Z0: 全てのSPM命令に対して0であるべきで、LPM命令に対するバイト選択です。

自己プログラミング中のZポイントの使用については「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」を参照してください。

(訳注) 原書の表31-9.(ATmega88/V)と表31-12.(ATmega168/V)は表31-9.として纏めました。

## 31.9. ブートローダ関係レジスタ – ATmega88/V/168/V

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

このレジスタはブートローダ動作を制御するために必要とされる制御ビットを含みます。

LD系とST系の命令を使い、データ空間としてI/Oレジスタをアクセスする時は、提供された変位が使われなければなりません。I/O特定命令のINとOUTを使う時は、この変位が\$20で減算され、I/Oアドレスの変位は\$00～\$3F内になります。

名称 : SPMCSR

変位 : \$57 (\$37)

リセット : \$00

特質 : I/O特定命令でI/Oレジスタとしてアクセスする時の変位アドレスは\$37です。

ビット	7	6	5	4	3	2	1	0
	SPMIE	RWWSB	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN
アクセス種別	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

#### ● ビット7 – SPMIE : SPM操作可割り込み許可 (SPM Interrupt Enable)

SPMIEビットが1を書かれ、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されているなら、SPM操作可割り込みが許可されます。SPM操作可割り込みはSPMCSRのSPM操作許可(SPMEN)ビットが解除(0)されている限り実行されます。

#### ● ビット6 – RWWSB : RWW領域多忙 (Read-While-Write Section Busy)

RWW領域に自己プログラミング(ページ消去またはページ書き込み)操作が開始されると、RWWSBがハードウェアによって設定(1)されます。RWWSBビットが設定(1)されると、RWW領域はアクセスできません。自己プログラミングが完了された後にRWWSREビットが1を書かれると、このRWWSBビットは解除(0)されます。同様に、ページ設定操作が開始されると、RWWSBビットは自動的に解除(0)されます。

#### ● ビット5 – SIGRD : 識票列読み出し (Signature Row Read)

このビットがSPMENと同時に1を書かれると、次の3クロック周期内のLPM命令は識票列からバイトを転送先レジスタに読みます。本章内の「ソフトウェアからの識票列読み出し」を参照してください。SIGRDとSPMENが設定(1)された後の4クロック周期内のSPM命令は無効です。この操作は将来の使用に対して予約されており、使われるべきではありません。

#### ● ビット4 – RWWSRE : RWW領域読み出し許可 (Read-While-Write Section Read Enable)

RWW領域にプログラミング(ページ消去またはページ書き込み)すると、RWW領域は(RWWSBがハードウェアによって設定(1)され)読み出しに対して防がれます。RWW領域を再許可するために使用者ソフトウェアはプログラミングが完了(SPMENが解除(0)されるまで待たねばなりません。その後、RWWSREビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令がRWW領域を再許可します。ページ消去またはページ書き込みでフラッシュメモリが多忙(SPMEN=1)の間、RWW領域は再許可できません。フラッシュメモリが設定されている間にRWWSREビットが書かれると、フラッシュメモリ設定操作は失敗し、設定したデータが失われます。

#### ● ビット3 – BLBSET : ブート施錠ビット設定 (Boot Lock Bits Set)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令は、R0のデータに従って一般施錠とブート施錠ビットを設定します。R1のデータとZポインタのアドレスは無視されます。BLBSETビットは施錠ビットの設定完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。

SPMCSRでBLBSET(SPMCSR.BLBSET)とSPMEN(SPMCSR.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。本章内の「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」を参照してください。

#### ● ビット2 – PGWRT : ページ書き込み (Page Write)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令は一時緩衝部に格納したデータでページ書き込みを実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGWRTビットはページ書き込みの完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の書き込み動作中にCPUは停止されます。

#### ● ビット1 – PGERS : ページ消去 (Page Erase)

このビットがSPMENと同時に1を書かれると、次の4クロック周期内のSPM命令はページ消去を実行します。ページアドレスはZポインタの上位部から取得されます。R1とR0のデータは無視されます。PGERSビットはページ消去の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。NRWW領域がアドレス指定されると、ページ全体の消去中にCPUは停止されます。

#### ● ビット0 – SPMEN : SPM操作許可 (Store Program Memory Enable)

このビットは次の4クロック周期間のSPM命令を許可します。このビットがRWWSRE、BLBSET、PGWRT、PGERSのどれかと共に1を書かれると、続くSPM命令は特別な意味を持ちます(上の記述をご覧ください)。SPMENだけが書かれると、続くSPM命令はZポインタによってアドレス指定したページ一時緩衝部へR1:R0の値を格納します。Zレジスタの最下位ビットは無視されます。SPMENビットはSPM命令の完了で、または4クロック周期内にSPM命令が実行されない場合、自動的に解除(0)されます。ページ消去とページ書き込み中、SPMENビットはその動作が完了されるまで1に留まります。

下位6ビットに100001, 010001, 001001, 000101, 000011, 000001以外のどんな組み合わせを書いても無効です。

## 32. MEMPROG – メモリプログラミング

### 32.1. プログラムメモリとデータメモリ用施錠ビット

ATmega48/Vは2つ、他のデバイスは6つの施錠ビットを提供します。これらは非プログラム(1)のままか「施錠ビットの保護種別」の表で一覽される付加機能を得るためにプログラム(0)することができます。この施錠ビットはチップ消去指令でのみ1に消去することができます。

ATmega48/Vは独立したブートローダ領域を持たず、SELFPRGENヒューズがプログラム(0)されると、SPM命令がフラッシュメモリ全体に対して許可され、さもなければSPM命令が禁止されます。

表32-1. 施錠ビットバイトの内容

名称	ビット番号	意味	既定値 (注1)
-	7		1 (非プログラム)
-	6		1 (非プログラム)
BLB12 (注2)	5	ブートローダ領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB11 (注2)	4		1 (非プログラム)
BLB02 (注2)	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 (注2)	2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注1: 0はプログラム、1は非プログラムを意味します。

注2: ATmega88/VとATmega168/Vだけです。

表32-2. 施錠ビットの保護種別

LB 種別	LB2	LB1	外部プログラミングに対する保護
1	1	1	メモリ施錠機能は機能しません。
2	1	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)が禁止されます。ヒューズビットが固定されます。(注1)
3	0	0	フラッシュ、EEPROMの更なるプログラミング(書き込み)と照合(読み出し)が禁止されます。ヒューズビットとブート施錠ビットが固定されます。(注1)

注: 0はプログラム、1は非プログラムを意味します。

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

表32-3. ブート施錠ビットの保護種別 (ATmega88/VとATmega168/Vだけです。)

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1)
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1)
LB1 種別	BLB12	BLB11	フラッシュメモリのブートローダプログラム領域に対する保護
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注2)
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2)

注: 0はプログラム、1は非プログラムを意味します。

注1: 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2: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

(訳注) 原書の表32-3.と表32-4.は表32-3.として纏めました。

## 32.2. ヒューズ ビット

本デバイスは3つのヒューズ バイトを持ちます。以下の表は全てのヒューズの概略機能とヒューズ バイト内でどのように配置されるかを示します。ヒューズはプログラムされると、論理0として読まれることに注意してください。

表32-5. ATmega48/V 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
-	7~1		1 (非プログラム)
SELFPRGEN	0	自己プログラミング機能許可。	1 (非プログラム) 自己プログラミング不許可

表32-6. ATmega88/V/168/V 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
-	7~3		1 (非プログラム)
BOOTSZ1	2	ブート ロータ容量選択。 (注1)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOOTRST	0	リセット ベクタ(応用領域またはブート ロータ領域)選択。	1 (非プログラム) 応用領域

注1: BOOTSZ1,0既定値は最大ブート ロータ容量になります。表31-7をご覧ください。

表32-7. ヒューズ 上位バイト一覧

名称	ビット	意味	既定値
RSTDISBL(注1)	7	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
DWEN	6	デバッグ WIRE機能許可。	1 (非プログラム) デバッグ WIRE不許可
SPIEN (注2)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON (注3)	4	ウォッチドッグ タイマ常時有効。	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BODLEVEL2	2		1 (非プログラム)
BODLEVEL1	1	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注4)	1 (非プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: RSTDISBLヒューズの記述については「入出力ポート」章の「ポートCの交換機能」を参照してください。

注2: SPIENヒューズは直列プログラミングでアクセスできません。

注3: 詳細については「WDTCSR - ウォッチドッグ タイマ制御レジスタ」を参照してください。

注4: BODLEVELヒューズの復号については「システムとリセットの特性」内の「BODLEVELヒューズ設定」表を参照してください。

表32-8. ヒューズ 下位バイト一覧

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

注1: SUT1,0の既定値は既定クロック元に対する最大起動時間になります。詳細については「システム クロックとクロック選択」章の「校正付き内蔵RC発振器」の「校正付き内蔵RC発振器用起動遅延時間選択」表をご覧ください。

注2: CKSEL3~0の既定設定は8MHz校正付き内蔵RC発振器になります。詳細については「システム クロックとクロック選択」章の「校正付き内蔵RC発振器」の「校正付き内蔵RC発振器動作種別」表をご覧ください。

注3: CKOUTヒューズはシステム クロックにポートB0へ出力することを許します。詳細については「システム クロックとクロック選択」章の「クロック出力緩衝部」を参照してください。

注4: 詳細については「システム クロックとクロック選択」章の「システム クロック前置分周器」を参照してください。

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。

- 関連リンク
- 59頁の「交換ポート機能」
  - 32頁の「校正付き内蔵RC発振器」
  - 44頁の「WDTCSR - ウォッチドッグ タイマ制御レジスタ」
  - 215頁の「システムとリセットの特性」

### 32.2.1. ヒューズのラッチ

ヒューズ値はデバイスがプログラミング動作へ移行する時にラッチされ、ヒューズ値の変更はデバイスがプログラミング動作を去るまで無効です。これは一旦プログラム(0)されると直ぐに効果があるEESAVEヒューズには適用されません。ヒューズは通常動作での電源投入でもラッチされません。

### 32.3. 識票バイト

デバイスは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。本デバイス用の識票バイトは右表で与えられます。

表32-9. デバイスの識別番号(ID)

部品番号	識票バイト アドレス		
	\$0000	\$0001	\$0002
ATmega48/V	\$1E	\$92	\$05
ATmega88/V	\$1E	\$93	\$0A
ATmega168/V	\$1E	\$94	\$06

### 32.4. 校正バイト

本デバイスは内蔵RC発振器用に1バイトの校正値を持ちます。このバイトは識票アドレス空間でアドレス\$0000の上位バイトにあります。リセット中、校正付き内蔵RC発振器の正しい周波数を保証するために、このバイトが発振校正(OSCCAL)レジスタへ自動的に書かれます。

関連リンク [32ページの「校正付き内蔵RC発振器」](#)

### 32.5. ページ容量

表32-10. フラッシュメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	PCMSB
ATmega48/V	2K語 (4Kバイト)	32語	PC4~0	64	PC10~5	10
ATmega88/V	4K語 (8Kバイト)	32語	PC4~0	128	PC11~5	11
ATmega168/V	8K語 (16Kバイト)	64語	PC5~0	128	PC12~6	12

表32-11. EEPROMメモリのページ数とページの語数

デバイス	全容量	ページ容量	PCWORD	ページ数	PCPAGE	EEAMSB
ATmega48/V	256バイト	4バイト	EEA1~0	64	EEA7~2	7
ATmega88/V	512バイト	4バイト	EEA1~0	128	EEA8~2	8
ATmega168/V	512バイト	4バイト	EEA1~0	128	EEA8~2	8

## 32.6. 並列プログラミング

この項は本デバイスに於けるプログラム用フラッシュメモリ、データ用EEPROM、メモリの**施錠ビット**、**ヒューズビット**の並列プログラミングと照合の方法を記述します。特記事項を除いて、パルス幅は最低250nsと仮定されます。

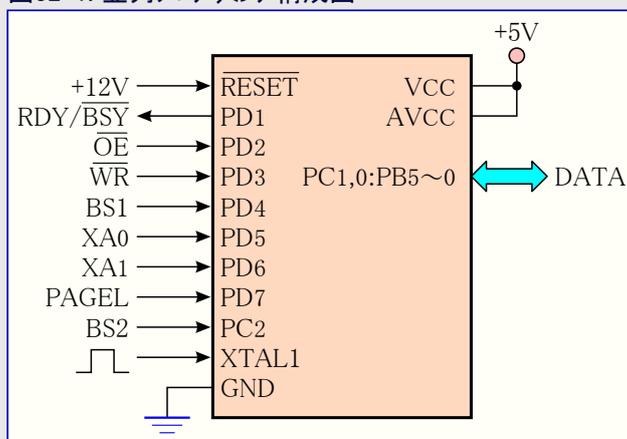
### 32.6.1. 信号名

本章で本デバイス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。本章内の「並列プログラミング構成図」図と「信号名とピン名の関係」表を参照してください。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「XA0とXA1の符号化(機能)」表で示されます。

WRまたはOEパルス送出時、設定された指令が決める動作が実行されます。各種指令は「**指令バイトのビット符号化**」表で示されます。

図32-1. 並列プログラミング構成図



注: VCC-0.3V<AVCC<VCC+0.3Vですが、AVCCは常に4.5~5.5Vであるべきです。

表32-12. 信号名とピン名の関係

信号名	ピン名	入出力	機能
RDY/BSY	PD1	出力	0(Low): 多忙(プログラミング中) 1(High): 準備可(指令受付可)
OE	PD2	入力	出力許可(負論理)
WR	PD3	入力	書き込みパルス(負論理)
BS1	PD4	入力	上位/下位バイト選択1 (0:下位, 1:上位) (一般用)
XA0	PD5	入力	XTAL動作ビット0
XA1	PD6	入力	XTAL動作ビット1
PAGEL	PD7	入力	プログラムメモリとEEPROMデータをページ一時緩衝部に設定
BS2	PC2	入力	上位/下位バイト選択2 (0:下位, 1:上位) (ヒューズビット用)
DATA	PC1,0:PB5~0	入出力	双方向データバス (OE=Low時出力)

表32-13. プログラミング動作移行時ピン値

ピン名	シンボル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

表32-14. 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32-15. 指令バイトのビット符号化

指令バイト	指令の機能
\$80 (1000 0000)	チップ消去
\$40 (0100 0000)	ヒューズビット書き込み
\$20 (0010 0000)	施錠ビット書き込み
\$10 (0001 0000)	フラッシュメモリ書き込み
\$11 (0001 0001)	EEPROM書き込み
\$08 (0000 1000)	識票バイト、校正バイト読み出し
\$04 (0000 0100)	ヒューズビット、施錠ビット読み出し
\$02 (0000 0010)	フラッシュメモリ読み出し
\$03 (0000 0011)	EEPROM読み出し

## 32.7. 並列プログラミング手順

### 32.7.1. 並列プログラミング動作への移行

次に示す方法がデバイスを(高電圧)並列プログラミング動作にします。

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、表32-13で一覧されるProg\_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5~5.5Vを印加します。それから20 $\mu\text{s}$ 以内にVCCが最低1.8Vに達することを保証してください。
- ③ 20~60 $\mu\text{s}$ 待ち、 $\overline{\text{RESET}}$ に11.5~12.5Vを印加します。
- ④ Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu\text{s}$ 、Prog\_enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立って少なくとも300 $\mu\text{s}$ 間待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

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

- ① VCCを0V、 $\overline{\text{RESET}}$ ピンを0V、表32-13で一覧されるProg\_enableピンを全てLow(0)に設定します。
- ② VCCとGND間に4.5~5.5Vを印加します。
- ③ VCCを監視し、0.9~1.1Vに達したら直ぐ、 $\overline{\text{RESET}}$ に11.5~12.5Vを印加します。
- ④ Prog\_enable識別がラッチされてしまうのを確実にするため、高電圧が印加されてしまった後、最低10 $\mu\text{s}$ 、Prog\_enableピンを無変化に保ちます。
- ⑤ 如何なる並列プログラミング指令を与えるのにも先立ってVCCが実際に4.5~5.5Vに達するまで待ちます。
- ⑥ デバイスの電源を落とすか、 $\overline{\text{RESET}}$ ピンを0Vに持ってくることによってプログラミング動作を抜けます。

### 32.7.2. 効率的なプログラミングへの考慮

設定した指令とアドレスはプログラミング中、維持されます。効率的なプログラミングを行うために次を考慮すべきです。

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

### 32.7.3. チップ消去

チップ消去はフラッシュメモリ、SRAM、EEPROM、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:** EESAVEヒューズがプログラム(0)されていると、EEPROMはチップ消去中、保護されます。

「チップ消去」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$80(1000 0000)に設定します。これはチップ消去指令です。
- ④ XTAL1に正パルスを与えます。これはチップ消去指令を設定します。
- ⑤  $\overline{\text{WR}}$ に負パルスを与えます。これはチップ消去を開始します。RDY/ $\overline{\text{BSY}}$ がLow(0)になります。
- ⑥ 次の指令を設定する前にRDY/ $\overline{\text{BSY}}$ がHigh(1)になるまで待ちます。

### 32.7.4. フラッシュ メモリ書き込み (次頁の図32-3.タイミングを参照)

フラッシュ メモリは表32-10のようにページで構成されます。フラッシュ メモリへ書く時にプログラム データはページ緩衝部にラッチされます。これは同時に書かれることをプログラム データの1ページに許します。次の手順は完全なフラッシュ メモリの書き込み方法を記述します。

#### A. 「フラッシュ メモリ書き込み」指令設定

- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② BS1をLow(0)に設定します。
- ③ DATAを\$10(0001 0000)に設定します。これはフラッシュ メモリ書き込み指令です。
- ④ XTAL1に正パルスを与えます。これはフラッシュ メモリ書き込み指令を設定します。

#### B. アドレス下位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をLow(0)に設定します。これは下位アドレス(バイト)を選びます。
- ③ DATAにアドレス下位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス下位バイトを設定します。

#### C. データ下位バイト設定

- ① XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ② DATAにデータ下位バイト(\$00～\$FF)を設定します。
- ③ XTAL1に正パルスを与えます。これはデータ下位バイトを設定します。

#### D. データ上位バイト設定

- ① BS1をHigh(1)に設定します。これは上位バイトを選びます。
- ② XA1をLow(0)、XA0をHigh(1)に設定します。これはデータ設定を許可します。
- ③ DATAにデータ上位バイト(\$00～\$FF)を設定します。
- ④ XTAL1に正パルスを与えます。これはデータ上位バイトを設定します。

#### E. 語(ワード)データをページ一時緩衝部に設定

- ① BS1をHigh(1)にします。これは上位バイトを選びます。
- ② PAGELに正パルスを与えます。これは語データをページ一時緩衝部にラッチ(設定)します。

#### F. 緩衝部全体が満たされるか、または(必要な)全てのデータが緩衝部内に設定されるまで、B～Eを繰り返し

アドレス内の下位ビットがページ内の語(ワード)位置を指示する一方、上位ビットがフラッシュ メモリ内のページをアドレス指定します。これは以下の図で図解されます。ページ内の語アドレスに8ビット未満が必要とされる場合(ページ容量<256)、アドレス下位バイトの最上位(側)ビットがページ書き込み実行時のページ アドレスに使われることに注意してください。

#### G. アドレス上位バイト設定

- ① XA1をLow(0)、XA0をLow(0)に設定します。これはアドレス設定を許可します。
- ② BS1をHigh(1)に設定します。これは上位アドレス(バイト)を選びます。
- ③ DATAにアドレス上位バイト(\$00～\$07/\$0F/\$1F)を設定します。
- ④ XTAL1に正パルスを与えます。これはアドレス上位バイトを設定します。

#### H. ページ書き込み

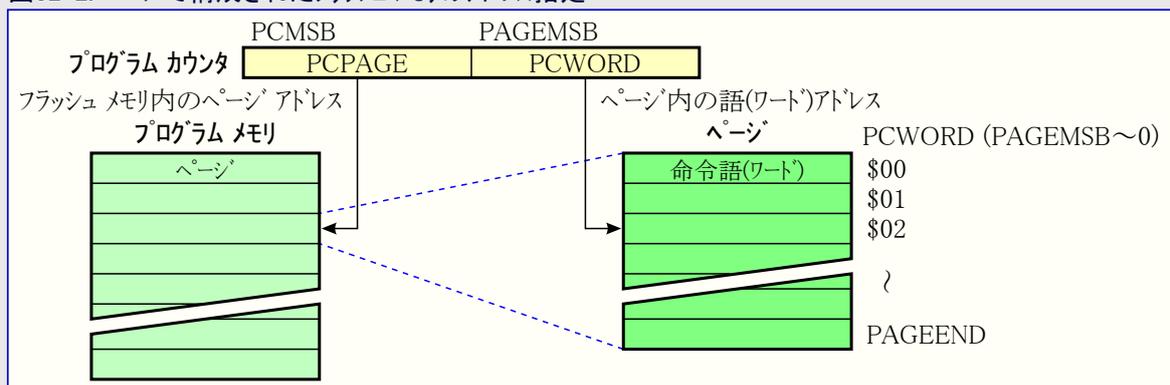
- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/BSYがLow(0)になります。
- ③ RDY/BSYがHigh(1)になるまで待ちます。

#### 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

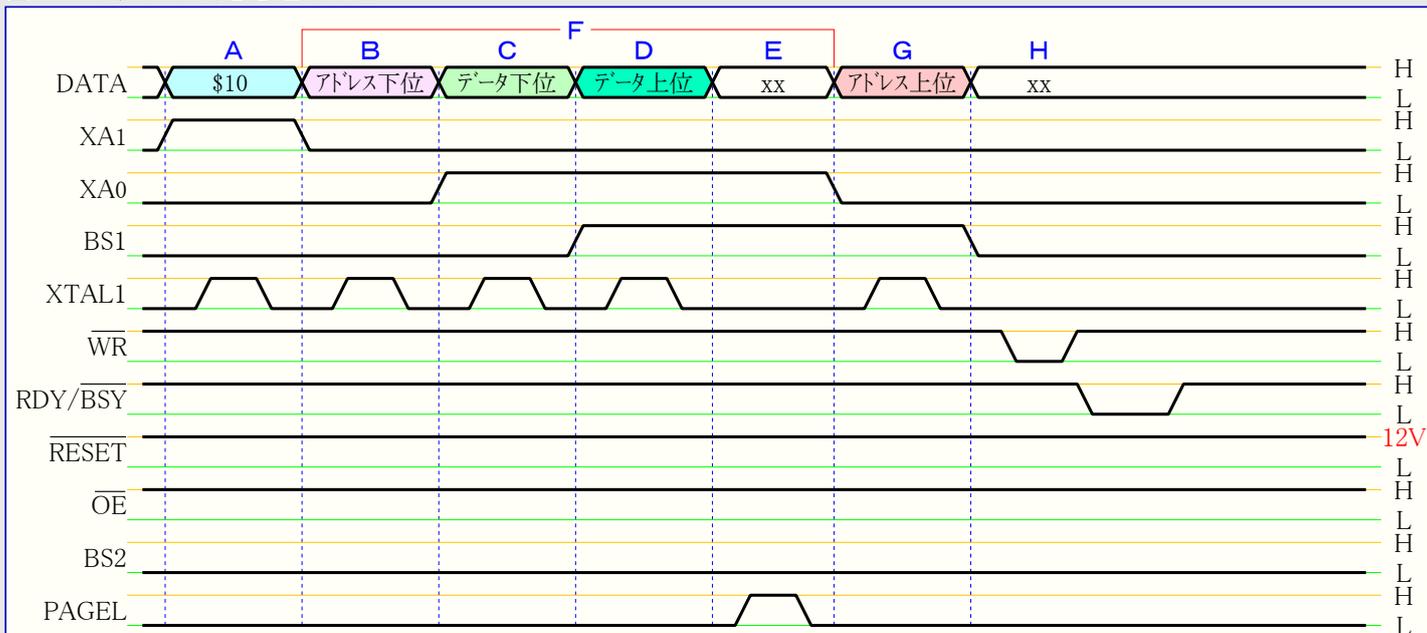
- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図32-2. ページで構成されたフラッシュ メモリのアドレス指定



注: PCPAGEとPCWORDは表32-10.で一覧されます。

図32-3. フラッシュメモリ書き込みタイミング



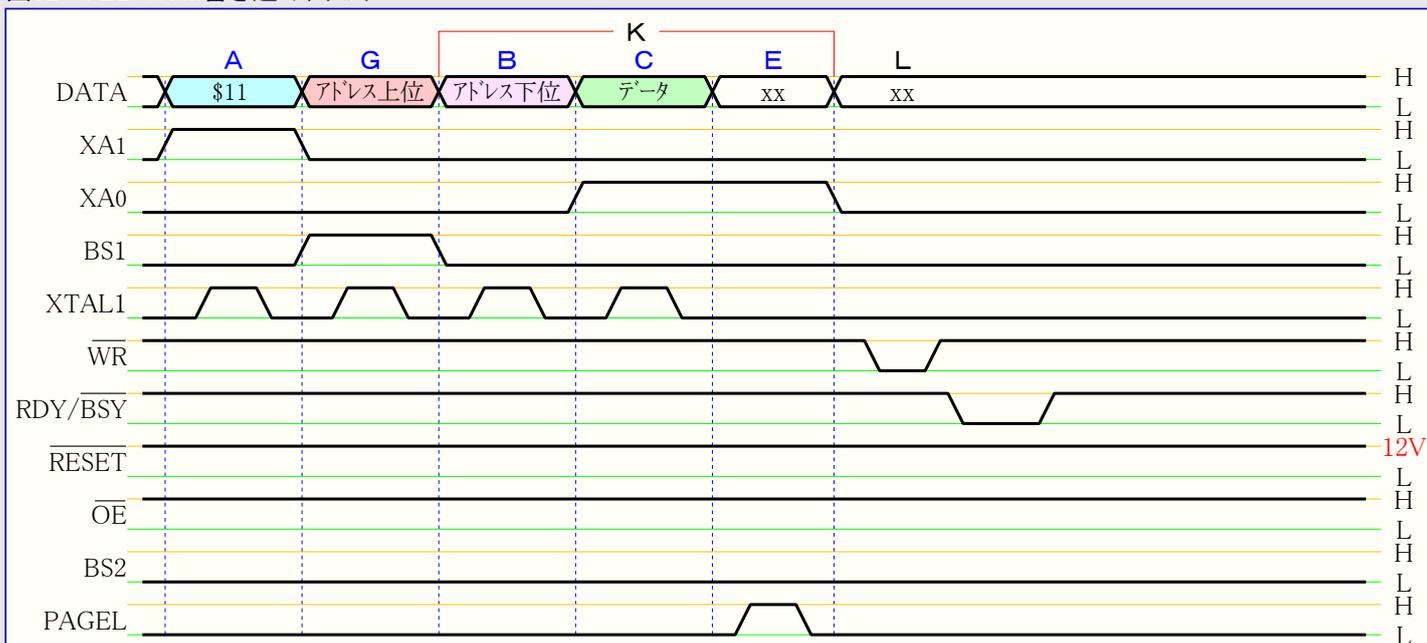
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

### 32.7.5. EEPROM書き込み

EEPROMは表32-11のようにページで構成されます。EEPROMを書く時にデータはページ緩衝部にラッチされます。これは同時に書かれることをデータの1ページに許します。データ用EEPROMメモリの書き込み方法は次のとおりです。(指令、アドレス、データ設定の詳細については「フラッシュメモリの書き込み」を参照。図32-4タイミング参照。)

1. EEPROM書き込み指令\$11(0001 0001)を設定します。(「フラッシュメモリ書き込み」のAを参照)
  2. アドレス上位バイト(\$00～\$00/\$01/\$01)を設定します。(「フラッシュメモリ書き込み」のGを参照)
  3. アドレス下位バイト(\$00～\$FF)を設定します。(「フラッシュメモリ書き込み」のBを参照)
  4. データバイト(\$00～\$FF)を設定します。(「フラッシュメモリ書き込み」のCを参照)
  5. データをラッチします(PAGELに正パルスを与えます)。(「フラッシュメモリ書き込み」のEを参照)
- K. 緩衝部全体が満たされるまで3～5を繰り返します。
- L. EEPROMページ書き込み
- ① BS1をLow(0)に設定します。
  - ② WRに負パルスを与えます。これはEEPROMページ書き込みを開始します。RDY/BSYがLow(0)になります。
  - ③ 次のページを書く前に、RDY/BSYがHigh(1)になるまで待ちます。

図32-4. EEPROM書き込みタイミング



### 32.7.6. フラッシュ メリ読み出し

フラッシュ メリの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については「フラッシュ メリの書き込み」を参照)

1. フラッシュ メリ読み出し指令\$02(0000 0010)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$07/\$0F/\$1F)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。フラッシュ メリ語(ワード)の下位バイトが直ぐにDATAで読めます。
5. BS1をHigh(1)に設定します。フラッシュ メリ語(ワード)の上位バイトが直ぐにDATAで読めます。
6. OEをHigh(1)に設定します。DATAはHi-Zになります。

### 32.7.7. EEPROM読み出し

データ用EEPROMの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については「フラッシュ メリの書き込み」を参照)

1. EEPROM読み出し指令\$03(0000 0011)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. アドレス上位バイト(\$00~\$00/\$01/\$01)を設定します。(「フラッシュ メリ書き込み」のGを参照)
3. アドレス下位バイト(\$00~\$FF)を設定します。(「フラッシュ メリ書き込み」のBを参照)
4. BS1をLow(0)、OEをLow(0)に設定します。EEPROMのバイト データが直ぐにDATAで読めます。
5. OEをHigh(1)に設定します。DATAはHi-Zになります。

### 32.7.8. ヒューズ ビット書き込み (訳注:原書での拡張/上位/下位ヒューズ項を1つに纏めました。)

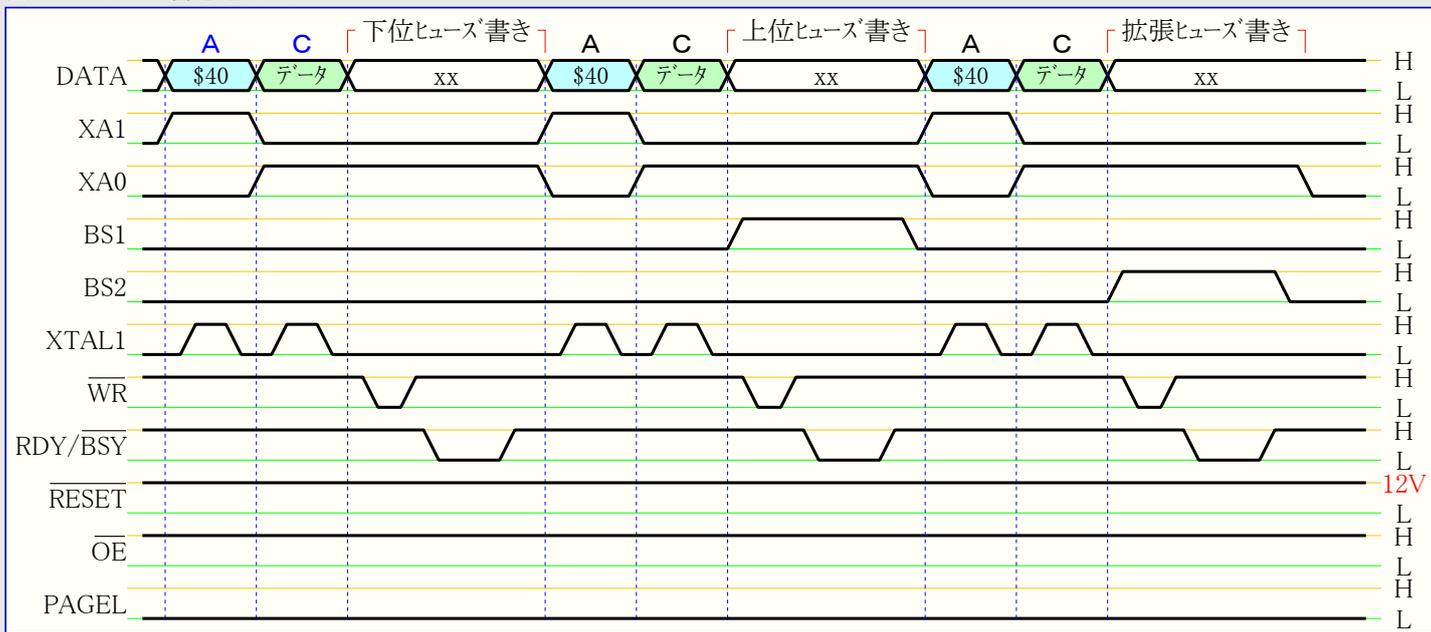
各ヒューズ バイトの書き込み方法は次のとおりです。(指令とデータ設定の詳細については「フラッシュ メリの書き込み」を参照)

1. ヒューズ ビット書き込み指令\$40(0100 0000)を設定します。(「フラッシュ メリ書き込み」のAを参照)
2. データ下位バイトを設定します。0=プログラム,1=非プログラム(消去)です。(「フラッシュ メリ書き込み」のCを参照)
3. BS1とBS2を右表の目的バイトに対応する設定にします。
4. WRに負パルスを与え、RDY/BSYがHighになるまで待ちます。
5. 3.でHighに設定したBS1、BS2をLow(0)に戻します。これはデータ下位バイトを選びます。

表A. ヒューズ バイト対応BS1,BS2設定

ヒューズ バイト	BS1	BS2
拡張バイト	Low(0)	High(1)
上位バイト	High(1)	Low(0)
下位バイト	Low(0)	Low(0)

図32-5. ヒューズ書き込みタイミング



### 32.7.9. 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「フラッシュメモリの書き込み」を参照)

1. 施錠ビット書き込み指令\$20(0010 0000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. データ下位バイト(としてデータ)を設定します。0=プログラム,1=無変化です。LB保護種別3が設定(LB1とLB2がプログラム(0))されると、どの外部的なプログラミング動作種別によってもブート施錠ビットはプログラミングできません。(「フラッシュメモリの書き込み」のCを参照)
3.  $\overline{WR}$ に負パルスを与え、RDY/BSYがHighになるまで待ちます。

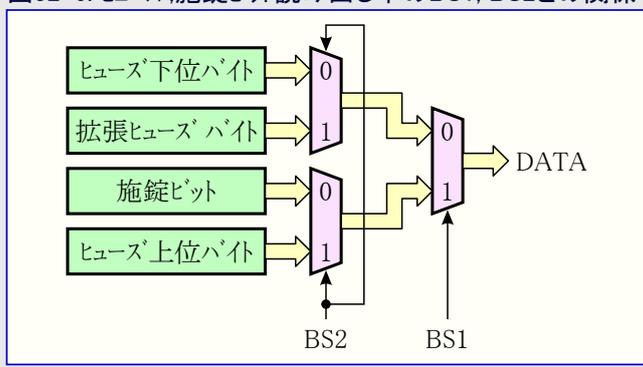
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

#### 32.7.10. ヒューズビットと施錠ビットの読み出し

ヒューズビットと施錠ビットの読み出し方法は次のとおりです。(指令設定の詳細については「フラッシュメモリの書き込み」を参照)

1. ヒューズビットと施錠ビットの読み出し指令\$04(0000 0100)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. BS1とBS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ下位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
3. BS1とBS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。ヒューズ上位ビットの状態が直ぐにDATAで読めます。(0=プログラム)
4. BS1をLow(0)、BS2をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。拡張ヒューズビットの状態が直ぐにDATAで読めます。(0=プログラム)
5. BS1をhigh(1)、BS2をLow(0)、 $\overline{OE}$ をLow(0)に設定します。施錠ビットの状態が直ぐにDATAで読めます。(0=プログラム)
6.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

図32-6. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



#### 32.7.11. 識票バイト読み出し

識票バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については「フラッシュメモリの書き込み」を参照)

1. 識票バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイト(\$00~\$02)を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をLow(0)、 $\overline{OE}$ をLow(0)に設定します。選んだ識票バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

#### 32.7.12. 校正バイト読み出し

校正バイトの読み出し方法は次のとおりです。(指令とアドレス設定の詳細については「フラッシュメモリの書き込み」を参照)

1. 校正バイト読み出し指令\$08(0000 1000)を設定します。(「フラッシュメモリの書き込み」のAを参照)
2. アドレス下位バイトに\$00を設定します。(「フラッシュメモリの書き込み」のBを参照)
3. BS1をHigh(1)、 $\overline{OE}$ をLow(0)に設定します。校正バイトが直ぐにDATAで読めます。
4.  $\overline{OE}$ をHigh(1)に設定します。DATAはHi-Zになります。

#### 32.7.13. 並列プログラミング特性

並列プログラミング特性については「並列プログラミング特性」を参照してください。

関連リンク 219頁の「並列プログラミング特性」

## 32.8. 直列プログラミング

フラッシュメモリとEEPROMの両方はRESETがGNDに引かれている間に直列SPIバスを使ってプログラミングできます。この直列インターフェースはSCK入力、MOSI入力、MISO出力から成ります。RESETがLowに設定された後、書き込み/消去操作が実行され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。

### 32.8.1. 直列プログラミング用ピン配置

表32-16. 直列プログラミング用ピン配置

信号名	ピン名	入出力	機能
MOSI	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

注: 上表でSPIプログラミング用のピン配置がご覧されます。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。

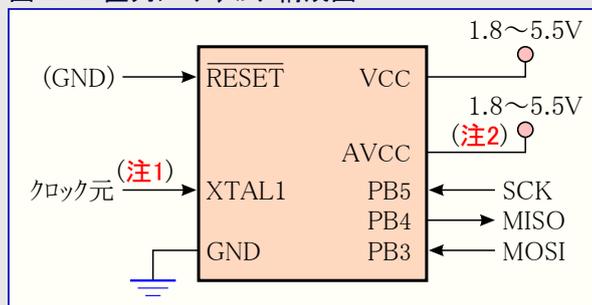
EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(直列プログラミングのみ)、チップ消去命令を初めに実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全てのメモリ位置の内容を\$FFにします。

CKSELヒューズによる有効なクロックが与えられなければなりません。直列クロック(SCK)入力のLowとHighの区間の最小値は次のように定義されます。

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

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

図32-7. 直列プログラミング構成図



注1: デバイスが内蔵発振器で動作する場合、XTAL1ピンにクロック元を接続する必要はありません。

注2: VCC-0.3V < AVCC < VCC+0.3Vですが、AVCCは常に1.8~5.5V内にすべきです。

## 32.9. 直列プログラミング手順

本デバイスに直列データを書く時にデータはSCKの上昇端で行われ、本デバイスから読む時にデータはSCKの下降端で行われます。タイミングの詳細については「直列プログラミングバイト通信波形」図を参照してください。

直列プログラミング動作での本デバイスのプログラミングと照合は次手順が推奨されます(表32-18.の直列プログラミング命令一式をご覧ください)。

### 1. 電源投入手順:

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

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

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

4. フラッシュメモリは1ページ単位で書かれます。ページ容量は表32-10.で得られます。このメモリページはページ設定命令と共にアドレスの下位5/5/6+1ビットとデータを供給することによって1バイトずつ設定されます。ページが正しく設定されるのを保証するため、与えられたアドレスにデータ上位バイトが適用される前にデータ下位バイトが設定されなければなりません。プログラムメモリのページはアドレスの上位6/7/7ビットを含むページ書き込み命令の設定によって(フラッシュメモリに)格納されます。ポーリング(BSY/RDY)が使われない場合、使用者は次のページを行う前に最低tWD\_FLASH待たなければなりません。フラッシュ書き込み操作完了前の(ポーリング以外の)直列プログラミングインターフェースでのアクセスは不正な書き込み結果になり得ます。

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

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

**ページ単位:** EEPROMの1ページはEEPROMページ設定命令と共にアドレス下位2ビットとデータを供給することによって1バイトずつ設定されます。EEPROMページはアドレスの上位6/7/7ビットを含むEEPROMページ書き込み命令によって(EEPROMに)格納されます。EEPROMページアクセス使用時、EEPROMページ設定命令で設定したバイト位置だけが変更されます。残りの位置は無変化で留まります。ポーリング(BSY/RDY)が使われない場合、次ページ(表32-11.参照)を行う前に最低tWD\_EEPROM待たなければなりません。チップ消去されたデバイスでの\$FFデータは書かれる必要がありません。

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

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

### 8. 電源OFF手順 (必要とされるならば)

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

表32-17. ヒューズ、フラッシュ、EEPROM次位置書き込み前の待機時間

シンボル	最低待機時間	備考
tWD_FUSE	4.5ms	ヒューズ書き込み
tWD_FLASH	2.6ms	フラッシュメモリ書き込み
tWD_EEPROM	3.6ms	EEPROM書き込み
tWD_ERASE	10.5ms	チップ消去

### 32.9.1. 直列プログラミング命令一式

この項は命令一式を記述します。

表32-18. 直列プログラミング命令一式

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

注1: 全命令が全デバイスで利用可能な訳ではありません。

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

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

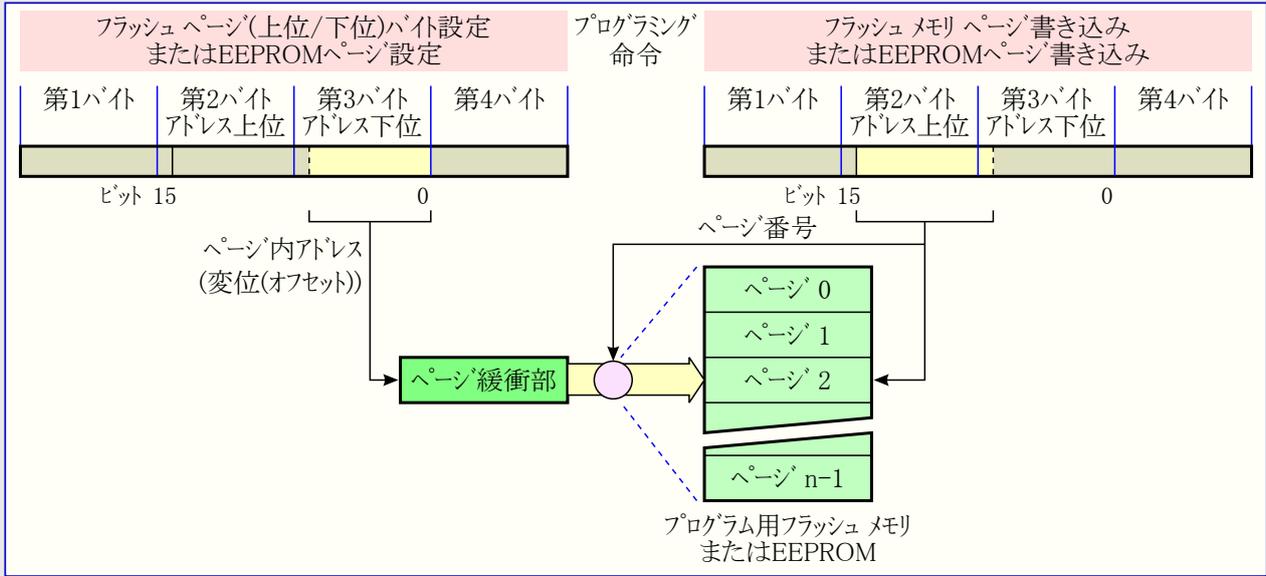
- ヒューズ、施錠ビット、識票バイト、校正バイト、ページ容量については対応項を参照してください。
- プログラミングと書き込み器に関する応用記述については <http://www.atmel.com/avr> をご覧ください。
- 第4バイトの赤背景はホスト読み込み(デバイス出力)を示します。

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

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

データがページ緩衝部に格納された後にEEPROMページをプログラムしてください。以降の図をご覧ください。

図32-8. 直列プログラミング命令例



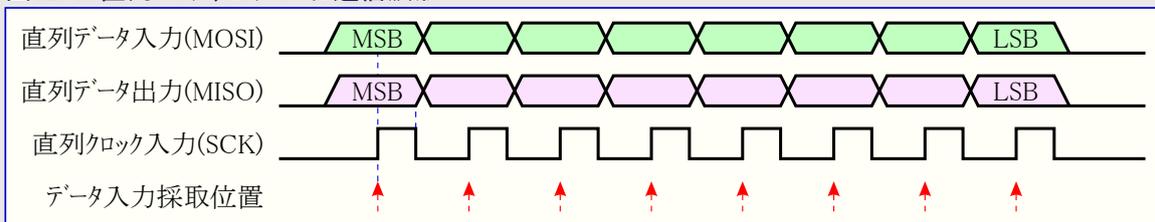
(訳補) フラッシュメモリ、EEPROM、ページ緩衝部内のアドレス(位置)指定に使われるビットはメモリ容量とページ構成に依存します。ATmega48/V/88/V/168/Vでのこれらの指定方法は次表で要約されます。

表B. アドレス(第2,3バイト)指定法

命令	第2バイト	第3バイト	備考
拡張アドレス設定			該当命令なし
フラッシュページ内バイト設定	0000 0000	000L LLLL 00LL LLLL	ATmega48/V/88/V : L=PC4~0 ATmega168/V : L=PC5~0
EEPROMページ内バイト設定	0000 0000	0000 00LL	ATmega48/V/88/V/168/V : L=EEA1~0
フラッシュメモリ読み出し	0000 0HHH 0000 HHHH 000H HHHH	LLLL LLLL	ATmega48/V : H=PC10~8, L=PC7~0 ATmega88/V : H=PC11~8, L=PC7~0 ATmega168/V : H=PC12~8, L=PC7~0
EEPROM読み出し	0000 0000 0000 000H	LLLL LLLL	ATmega48/V : L=EEA7~0 ATmega88/V/168/V : H=EEA8, L=EEA7~0
フラッシュページ書き込み	0000 0HHH 0000 HHHH 000H HHHH	LLL0 0000 LLL0 0000 LL00 0000	ATmega48/V : H=PC10~8, L=PC7~5 ATmega88/V : H=PC11~8, L=PC7~5 ATmega168/V : H=PC12~8, L=PC7~6
EEPROMバイト書き込み	0000 0000 0000 000H	LLLL LLLL	ATmega48/V : L=EEA7~0 ATmega88/V/168/V : H=EEA8, L=EEA7~0
EEPROMページ書き込み	0000 0000 0000 000H	LLLL LL00	ATmega48/V : L=EEA7~2 ATmega88/V/168/V : H=EEA8, L=EEA7~2

### 32.9.2. 直列プログラミング特性

図32-9. 直列プログラミングバイト通信波形



SPI部の特性については「SPIタイミング特性」を参照してください。

## 33. 電気的特性

### 33.1. 絶対最大定格

表33-1. 絶対最大定格

動作温度	-55°C~125°C
保存温度	-65°C~150°C
RESETを除くピン許容電圧	-0.5V~VCC+0.5V
RESETピン許容電圧	-0.5V~13.0V
最大動作電圧	6.0V
入出力ピン毎のDC電流	40.0mA
VCCとGNDピンのDC電流	200.0mA

注: 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。

### 33.2. DC特性

表33-2. 共通DC特性 TA=-40°C~85°C, VCC=1.8V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V <sub>IL</sub>	Lowレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	-0.5		0.2VCC (注1)	V
		VCC=2.4~5.5V	-0.5		0.3VCC (注1)	
V <sub>IL1</sub>	Lowレベル入力電圧 (XTAL1)	VCC=1.8~5.5V	-0.5		0.1VCC (注1)	V
V <sub>IL2</sub>	Lowレベル入力電圧 (RESET)	VCC=1.8~5.5V	-0.5		0.1VCC (注1)	
V <sub>IL3</sub>	Lowレベル入力電圧 (I/OとしてのRESET)	VCC=1.8~2.4V	-0.5		0.2VCC (注1)	V
		VCC=2.4~5.5V	-0.5		0.3VCC (注1)	
V <sub>IH</sub>	Highレベル入力電圧 (XTAL1,RESETを除く)	VCC=1.8~2.4V	0.7VCC (注2)		VCC+0.5	V
		VCC=2.4~5.5V	0.6VCC (注2)		VCC+0.5	
V <sub>IH1</sub>	Highレベル入力電圧 (XTAL1)	VCC=1.8~2.4V	0.8VCC (注2)		VCC+0.5	V
		VCC=2.4~5.5V	0.7VCC (注2)		VCC+0.5	
V <sub>IH2</sub>	Highレベル入力電圧 (RESET)	VCC=1.8~5.5V	0.9VCC (注2)		VCC+0.5	V
V <sub>IH3</sub>	Highレベル入力電圧 (I/OとしてのRESET)	VCC=1.8~2.4V	0.7VCC (注2)		VCC+0.5	
		VCC=2.4~5.5V	0.6VCC (注2)		VCC+0.5	
V <sub>OL</sub>	Lowレベル出力電圧 (RESETを除く) (注3)	IOL=20mA, VCC=5V			0.9	V
		IOL=10mA, VCC=3V			0.6	
V <sub>OH</sub>	Highレベル出力電圧 (RESETを除く) (注4)	IOH=-20mA, VCC=5V	4.2			V
		IOH=-10mA, VCC=3V	2.3			
I <sub>IL</sub>	I/OピンLowレベル入力漏れ電流	VCC=5.5V			1	μA
I <sub>IH</sub>	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			1	
R <sub>RST</sub>	RESETピンプルアップ抵抗		30		60	kΩ
R <sub>PU</sub>	I/Oピンプルアップ抵抗		20		50	
V <sub>ACIO</sub>	アナログ比較器入力変位(オフセット)電圧	VCC=5V, Vin=VCC/2		<10	40	mV
I <sub>ACLK</sub>	アナログ比較器入力漏れ電流		-50		50	
t <sub>ACPD</sub>	アナログ比較器伝播遅延時間	VCC=2.7V		750		ns
		VCC=4.0V		500		

注1: Lowレベルの認識が保証される最高電圧です。

注2: Highレベルの認識が保証される最低電圧です。

注3: 各I/Oポートは安定状態(非過渡時)に於いては検査条件(VCC=3Vで10mA, VCC=5Vで20mA)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. ポートC5~0, ADC7,6のIOLの合計が100mAを超えるべきではありません。
2. ポートC6, D4~0のIOLの合計が100mAを超えるべきではありません。
3. ポートB7~0, D7~5のIOLの合計が100mAを超えるべきではありません。

IOLが検査条件を超える場合、V<sub>OL</sub>も仕様書での値を超えます。表の検査条件より大きな吸い込み電流は保証されません。

注4は次頁をご覧ください。

**注3:** 各I/Oポートは安定状態(非過渡時)に於いては検査条件(VCC=3Vで10mA, VCC=5Vで20mA)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. ポートC6~0、D4~0、ADC7のIOHの合計が150mAを超えるべきではありません。
2. ポートB7~0、D7~5、ADC6のIOHの合計が150mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件より大きな吐き出し電流は保証されません。

関連リンク [37ページの「消費電力の最小化」](#)

### 33.2.1. DC特性 - 消費電力

表33-3. DC特性 (TA=-40°C~85°C, VCC=1.8V~5.5V (特記事項を除く))

シンボル	項目	条件	最小	代表(注1)	最大(注3)	単位
ICC	活動動作消費電流 (電力削減レジスタ(PRR)全ビット=1)	VCC=2V, 1MHz		-	0.55	mA
		VCC=3V, 4MHz		-	3.5	
		VCC=5V, 8MHz		-	12	
	アイドル動作消費電流 (電力削減レジスタ(PRR)全ビット=1)	VCC=2V, 1MHz		0.25	0.5	
		VCC=3V, 4MHz		-	1.5	
		VCC=5V, 8MHz		-	5.5	
パワーダウン動作消費電流 (注2)	VCC=3V, WDT有効			8	μA	
	VCC=3V, WDT禁止			1		

**注1:** 25°Cでの代表値です。

**注2:** この消費電流は入力漏れ電流を含みます。

**注3:** 最大値は特性付けされた値で製造に於いて上限検査されません。

### 33.3. 速度勾配

最高周波数は動作電圧に依存します。最高周波数対VCCの図で示されるように最高周波数対動作電圧曲線は1.8~2.7Vと2.7~4.5V間で直線です。

図33-1. ATmega48V/88V/168Vの最高周波数対VCC

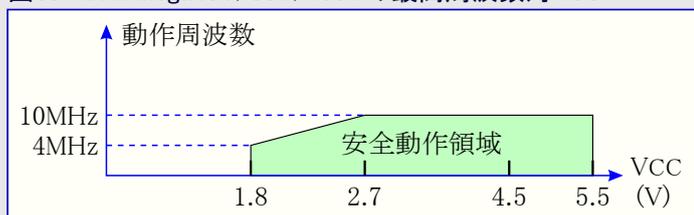
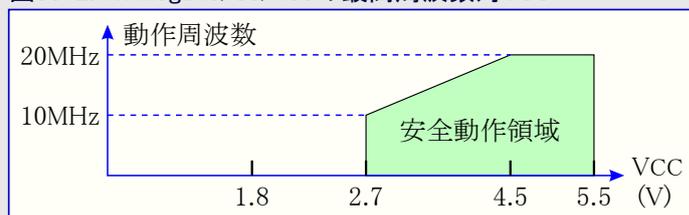


図33-2. ATmega48/88/168の最高周波数対VCC



### 33.4. クロック特性

関連リンク [32ページの「校正付き内蔵RC発振器」](#)

#### 33.4.1. 校正付き内蔵RC発振器精度

表33-4. 校正付き内蔵RC発振器の校正精度

校正種別	周波数	VCC	温度	校正精度
工場校正	8.0MHz	3V	25°C	±10%
使用者校正	7.3~8.1MHz	1.8~5.5V(注1), 2.7~5.5V(注2)	-40~85°C	±1%

**注1:** ATmega48V/88V/168Vに対する電圧範囲

**注2:** ATmega48/88/168に対する電圧範囲

### 33.4.2. 外部クロック信号駆動

図33-3. 外部クロック駆動波形

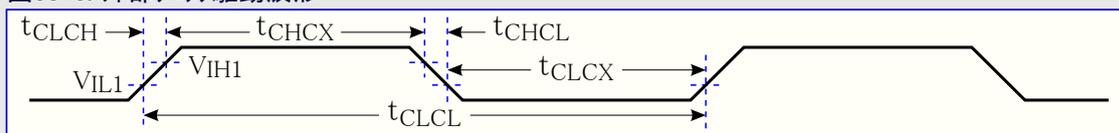


表33-5. 外部クロック特性

シンボル	項目	VCC=1.8~5.5V		VCC=2.7~5.5V		VCC=4.5~5.5V		単位
		最小	最大	最小	最大	最小	最大	
$1/t_{CLCL}$	クロック周波数	0	4	0	10	0	20	MHz
$t_{CLCL}$	クロック周期	250		100		50		
$t_{CHCX}$	Highレベル時間	100		40		20		ns
$t_{CLCX}$	Lowレベル時間	100		40		20		
$t_{CLCH}$	上昇時間		2.0		1.6		0.5	$\mu$ s
$t_{CHCL}$	下降時間		2.0		1.6		0.5	$\mu$ s
$\Delta t_{CLCL}$	隣接クロック周期間の変化率		2		2		2	%

### 33.5. システムとリセットの特性

表33-6. リセット、低電圧検出(BOD)、内部基準電圧の特性

シンボル	項目	条件	最小	代表	最大	単位
$V_{POT}$	上昇時電源ONリセット閾値電圧	$T_A = -40 \sim 85^\circ\text{C}$	0.7	1.0	1.4	V
	下降時電源ONリセット閾値電圧 (注1)		0.05	0.9	1.3	
$SR_{ON}$	電源ON電圧傾斜率		0.01		4.5	V/ms
$V_{RST}$	RESETピン閾値電圧		$0.2V_{CC}$		$0.9V_{CC}$	V
$t_{RST}$	RESETピンでの最小パルス幅		2.5			$\mu$ s
$V_{HYST}$	低電圧検出ヒステリシス電圧			50		mV
$t_{BOD}$	最小低電圧検出時間			2		$\mu$ s
$V_{BG}$	基準電圧	$V_{CC} = 2.7\text{V}$ $T_A = 25^\circ\text{C}$	1.0	1.1	1.2	V
$t_{BG}$	起動時間			40	70	$\mu$ s
$I_{BG}$	消費電流			15		$\mu$ A

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

注: 値は指針の意味だけです。

表33-7. BODLEVELヒューズ<sup>\*</sup> ( $V_{BOT}$ ) 設定 (注)

BODLEVEL2~0	最小	代表	最大	単位
1 1 1	低電圧検出(BOD)リセット禁止			
1 1 0	1.7	1.8	2.0	V
1 0 1	2.5	2.7	2.9	
1 0 0	4.1	4.3	4.5	
0 0 0 ~ 0 1 1	(予約)			

注: いくつかのデバイスで $V_{BOT}$ が公称最低動作電圧以下の可能性があります。この状態のデバイスについては、製造検査中、 $V_{CC} = V_{BOT}$ に落として検査されます。これはマイクロコントローラの正しい動作がもはや保証されない電圧になる前に、低電圧検出(BOD)リセットが起きることを保証します。この検査はATmega48/88/168についてBODLEVEL=101と100を、ATmega48V/88V/168VについてBODLEVEL=110と101を使って実行されます。

### 33.6. SPIタイミング特性

図33-4. SPI タイミング必要条件 (主装置動作)

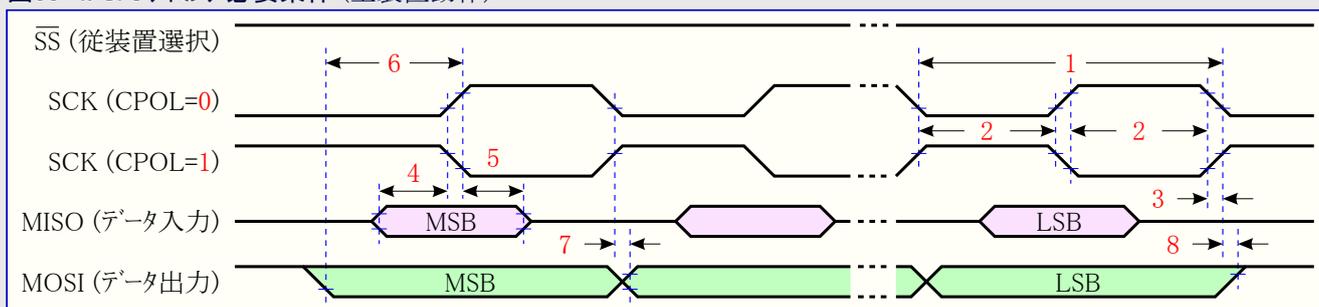


図33-5. SPI タイミング必要条件 (従装置動作)

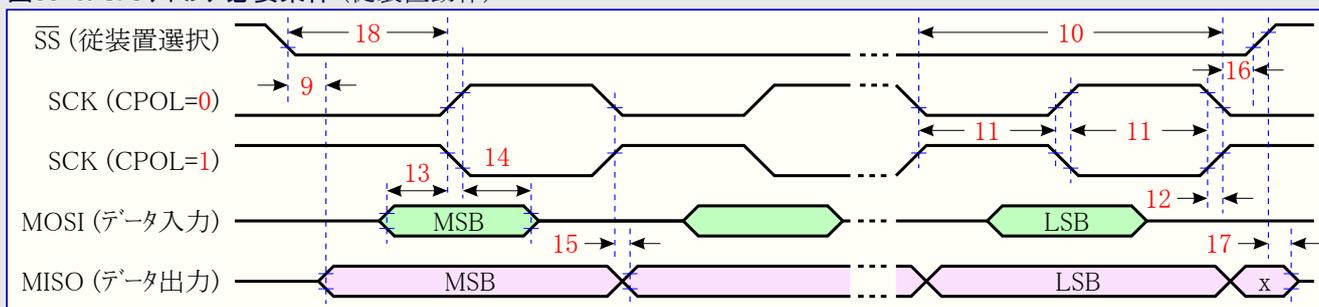


表33-8. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表23-5.参照		
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		$0.5 \cdot t_{SCK}$		ns
7	SCKからの出力遅延時間	主装置		10		
8	SCKからのHigh出力時間	主装置		10		
9	SS ↓からの出力遅延時間	従装置		15		
10	SCK周期	従装置	$4 \cdot t_{CK}$			
11	SCK High/Low期間 (注)	従装置	$2 \cdot t_{CK}$			
12	SCK上昇/下降時間	従装置			1.6	μs
13	入力データ 準備時間	従装置	10			
14	入力データ 保持時間	従装置	$t_{CK}$			
15	SCKからの出力遅延時間	従装置		15		
16	SCKからのSS ↑遅延時間	従装置	20			ns
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	$2 \cdot t_{CK}$			

注: SPIプログラミングでの最小SCK High/Low期間は、 $2t_{CLCL}$ ( $t_{CK} < 12\text{MHz}$ )、 $3t_{CLCL}$ ( $t_{CK} \geq 12\text{MHz}$ )です。

### 33.7. 2線直列インターフェース特性

本項の表は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。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。

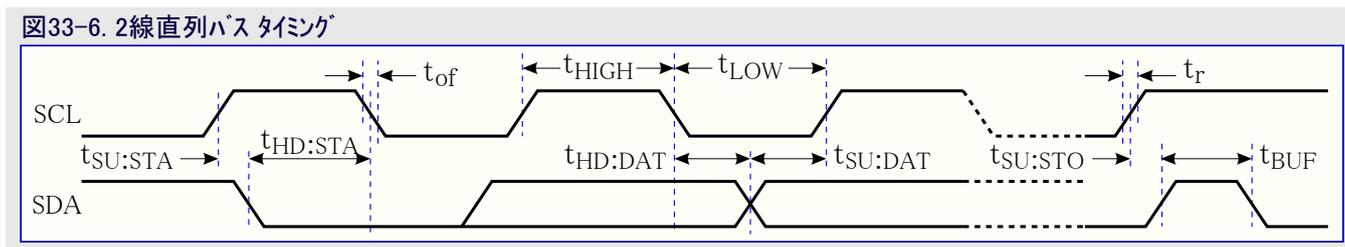


表33-9. 2線直列インターフェース必要条件

シンボル	項目	条件	最小	最大	単位
V <sub>IL</sub>	Lowレベル入力電圧		-0.5	0.3V <sub>CC</sub>	V
V <sub>IH</sub>	Highレベル入力電圧		0.7V <sub>CC</sub>	V <sub>CC</sub> +0.5	
V <sub>hys</sub> ①	シュミットトリガ入力ヒステリシス電圧		② 0.05V <sub>CC</sub>		ns
V <sub>OL</sub> ①	Lowレベル出力電圧	I <sub>OL</sub> =3mA	0	0.4	
t <sub>r</sub> ①	出力上昇時間(V <sub>ILmin</sub> →V <sub>IHmax</sub> )		② 20+0.1Cb ③	300	ns
t <sub>of</sub> ①	出力下降時間(V <sub>IHmin</sub> →V <sub>ILmax</sub> )	10pF<Cb<400pF	③ ② 20+0.1Cb ③	250	
t <sub>SP</sub> ①	入力パルス最小幅(尖頭消去濾波)		0	② 50	μA
I <sub>i</sub>	入力電流(ピン単位)	0.1V <sub>CC</sub> <V <sub>i</sub> <0.9V <sub>CC</sub>	-10	10	
C <sub>i</sub> ①	ピン入力容量			10	pF
f <sub>SCL</sub>	SCLクロック周波数	④⑤ f <sub>CK</sub> >max(16f <sub>SCL</sub> ,250kHz)	0	400	kHz
R <sub>p</sub>	プルアップ抵抗値	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	(V <sub>CC</sub> -0.4V)/3mA (V <sub>CC</sub> -0.4V)/3mA	1000ns/Cb 300ns/Cb	Ω
t <sub>HD:STA</sub>	(再送)開始条件保持時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.0 0.6		μs
t <sub>LOW</sub>	SCLクロックLowレベル時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.7 1.3		
t <sub>HIGH</sub>	SCLクロックHighレベル時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.0 0.6		ns
t <sub>SU:STA</sub>	再送開始条件準備時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.7 0.6		
t <sub>HD:DAT</sub>	データ保持時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	0 0	3.45 0.9	ns
t <sub>SU:DAT</sub>	データ準備時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	250 100		
t <sub>SU:STO</sub>	停止条件準備時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.0 0.6		μs
t <sub>BUF</sub>	停止条件→開始条件間バス開放時間	f <sub>SCL</sub> ≤ 100kHz f <sub>SCL</sub> > 100kHz	4.7 1.3		

① 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<sub>SCL</sub> > 100kHz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<sub>CK</sub>はCPU(システム)クロック周波数です。

⑤ この必要条件は全ての2線直列インターフェース動作に適用します。2線直列バスに接続した他の装置は一般的なf<sub>SCL</sub>必要条件に従うことだけを必要とします。

### 33.8. A/D変換器特性

表33-12. A/D変換特性

シンボル	項目	条件	最小	代表	最大	単位
	分解能			10		ビット
	絶対精度	変換クロック=200kHz		2		
		変換クロック=1MHz		4.5		
	(積分非直線性誤差、 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 VREF=4V 変換クロック=200kHz 雑音低減動作		2		
		変換クロック=1MHz 雑音低減動作		4.5		LSB
	積分非直線性誤差			0.5		
	微分非直線性誤差	VCC=4V、VREF=4V		0.25		
	利得誤差	変換クロック=200kHz		2		
	変位(オフセット)(ゼロ)誤差			2		
	変換時間	連続変換動作	13		260	μs
	変換クロック周波数		0.05		1	MHz
AVCC	アナログ供給電圧 (注)		VCC-0.3		VCC+0.3	
VREF	基準電圧		1.0		AVCC	V
VIN	入力電圧		GND		VREF	
	入力周波数帯域			38.5		kHz
VINT	内蔵基準電圧		1.0	1.1	1.2	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			100		MΩ

注: 絶対最小/最大AVCCは1.8/5.5Vです。

### 33.9. 並列プログラミング特性

図33-7. 並列プログラミング タイミング (一般的な必要条件)

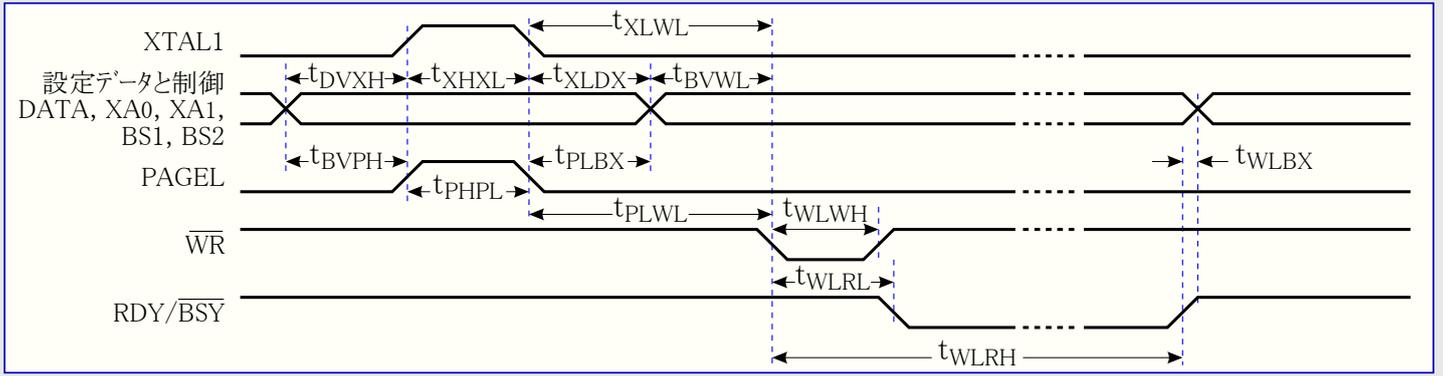
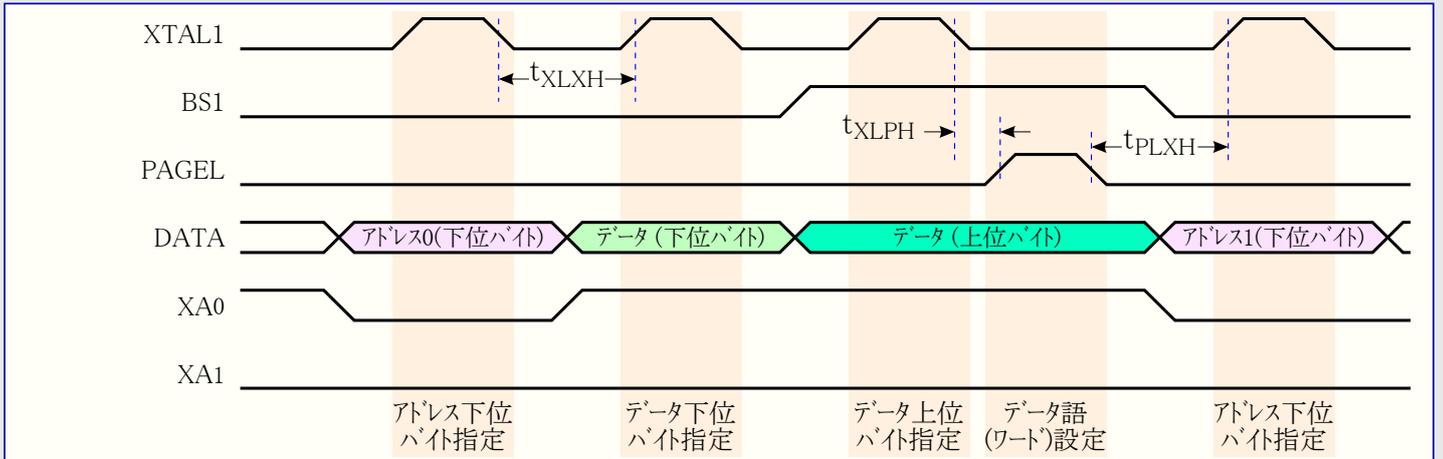
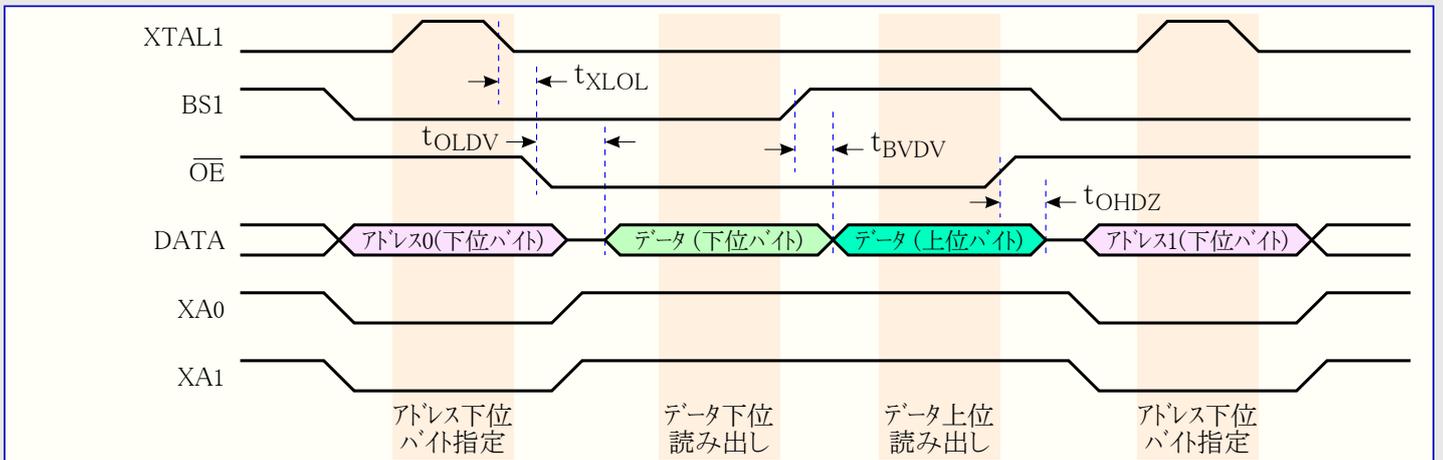


図33-8. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図33-7.で示されるタイミング必要条件(即ち、 $t_{DVXH}$ 、 $t_{XHXL}$ 、 $t_{XLDX}$ )は設定操作にも適用されます。

図33-9. 並列プログラミング タイミング (同一ページ読み出しでの必要条件)



注: 図33-7.で示されるタイミング必要条件(即ち、 $t_{DVXH}$ 、 $t_{XHXL}$ 、 $t_{XLDX}$ )は読み出し操作にも適用されます。

表33-11. 並列プログラミング特性 (VCC=5V±10%)

シンボル	項目	最小	代表	最大	単位
V <sub>PP</sub>	プログラミング許可電圧	11.5		12.5	V
I <sub>PP</sub>	プログラミング許可電流			250	μA
t <sub>DVXH</sub>	XTAL1 ↑ に対するデータと制御の準備時間	67			
t <sub>XLXH</sub>	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
t <sub>XHXL</sub>	XTAL1 Highパルス幅	150			
t <sub>XLDX</sub>	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
t <sub>XLWL</sub>	XTAL1パルス ↓ 後の $\overline{WR}$ ↓ 待機時間	0			
t <sub>XLPH</sub>	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
t <sub>PLXH</sub>	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			ns
t <sub>BVPH</sub>	PAGELパルス ↑ に対するBS1準備時間	67			
t <sub>PHPL</sub>	PAGEL Highパルス幅	150			
t <sub>PLBX</sub>	PAGELパルス ↓ 後のBS1保持時間	67			
t <sub>WL BX</sub>	RDY/ $\overline{BSY}$ ↑ 後のBS1,BS2保持時間	67			
t <sub>PLWL</sub>	PAGELパルス ↓ 後の $\overline{WR}$ パルス ↓ 待機時間	67			
t <sub>BVWL</sub>	$\overline{WR}$ パルス ↓ に対するBS1準備時間	67			
t <sub>WLWH</sub>	$\overline{WR}$ Lowパルス幅	150			
t <sub>WLRL</sub>	$\overline{WR}$ パルス ↓ 後のRDY/ $\overline{BSY}$ ↓ 遅延時間	0		1	μs
t <sub>WLRH</sub>	書き込み時間 ( $\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注1)	3.7		4.5	ms
t <sub>WLRH_CE</sub>	チップ消去時間 ( $\overline{WR}$ ↓ からRDY/ $\overline{BSY}$ ↑) (注2)	7.5		9.0	ms
t <sub>XLCL</sub>	XTAL1パルス ↓ 後の $\overline{OE}$ ↓ 待機時間	0			
t <sub>BVDV</sub>	BS1有効からのDATA遅延時間	0		250	ns
t <sub>OLDV</sub>	$\overline{OE}$ ↓ 後のDATA出力遅延時間			250	
t <sub>OHDZ</sub>	$\overline{OE}$ ↑ 後のDATA Hi-Z遅延時間			250	

注1: フラッシュメモリ、EEPROM、ヒューズビット、施錠ビット書き込み指令に対して有効です。

注2: チップ消去指令に対して有効です。

## 34. 代表特性

以下の図は代表的な特性を示します。これらの図は製造中に検査されていません。全ての消費電流測定は全I/Oピンを入力として設定した内部プルアップ許可で行われています。電源幅振幅の方形波発振器がクロック源として使われています。

活動動作とアイドル動作の全ての消費電流測定は電力削減レジスタ(PRR)で全ビットが設定(1)で行なわれ、従って関係周辺機能部はOFFにされます。アナログ比較器もこれらの測定中に禁止されます。パワーダウン動作での消費電力はクロック選択と無関係です。

消費電流は動作電圧、動作周波数、I/Oピンの負荷、I/Oピンの切り替え速度、命令実行、周囲温度のような様々な要素の関数です。支配的な要素は動作電圧と動作周波数です。

容量性負荷のピンの引き込み電流は(1つのピンに対して)  $C_L(\text{負荷容量}) \times V_{CC}(\text{動作電圧}) \times f(\text{I/Oピンの平均切り替え周波数})$  として推測できます。

デバイスは検査範囲よりも高い周波数特性を示します。デバイスは注文番号が示す周波数よりも高い周波数での機能特性を保証されません。

ウォッチドッグタイマ許可のパワーダウン動作での消費電流とウォッチドッグタイマ禁止のパワーダウン動作での消費電流間の違いは、ウォッチドッグタイマによって引き込んだ(消費した)差電流を表します。

### 34.1. 活動動作消費電流

図34-1. 活動動作消費電流 対 周波数 (100kHz~1MHz)

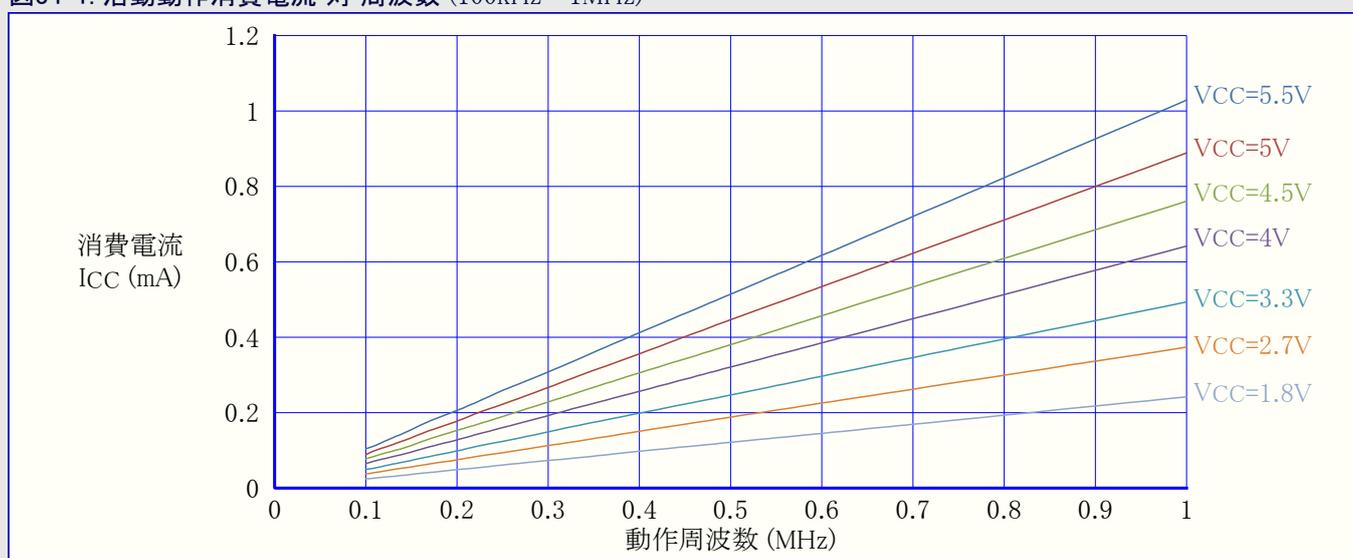


図34-2. 活動動作消費電流 対 周波数 (1MHz~24MHz)

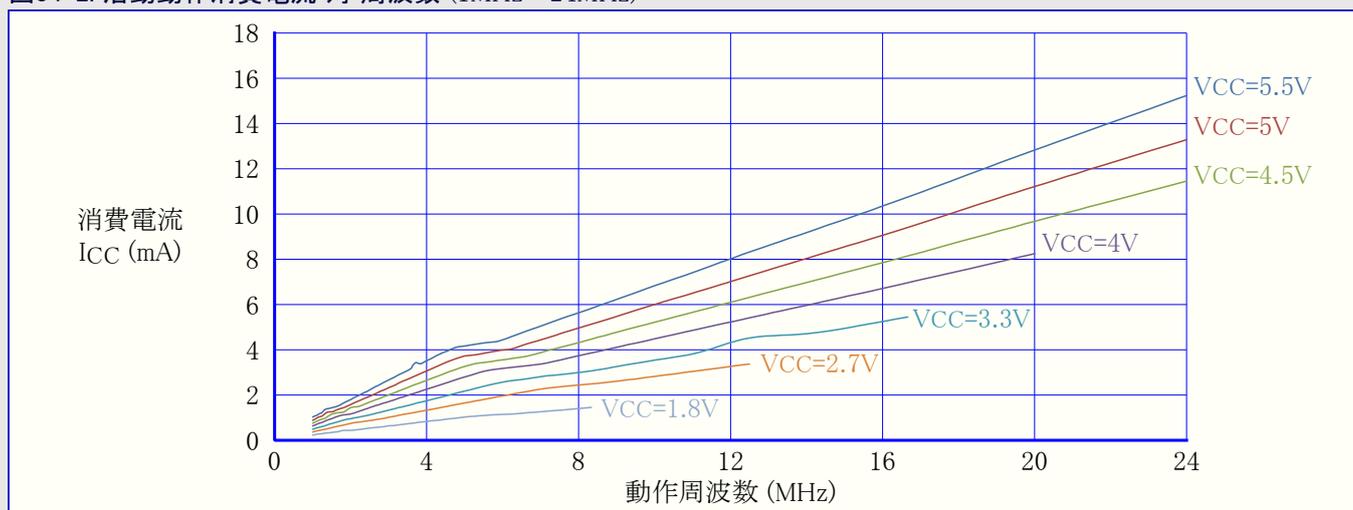


図34-3. 活動動作消費電流 対 動作電圧 (内蔵WDT発振器,128kHz)

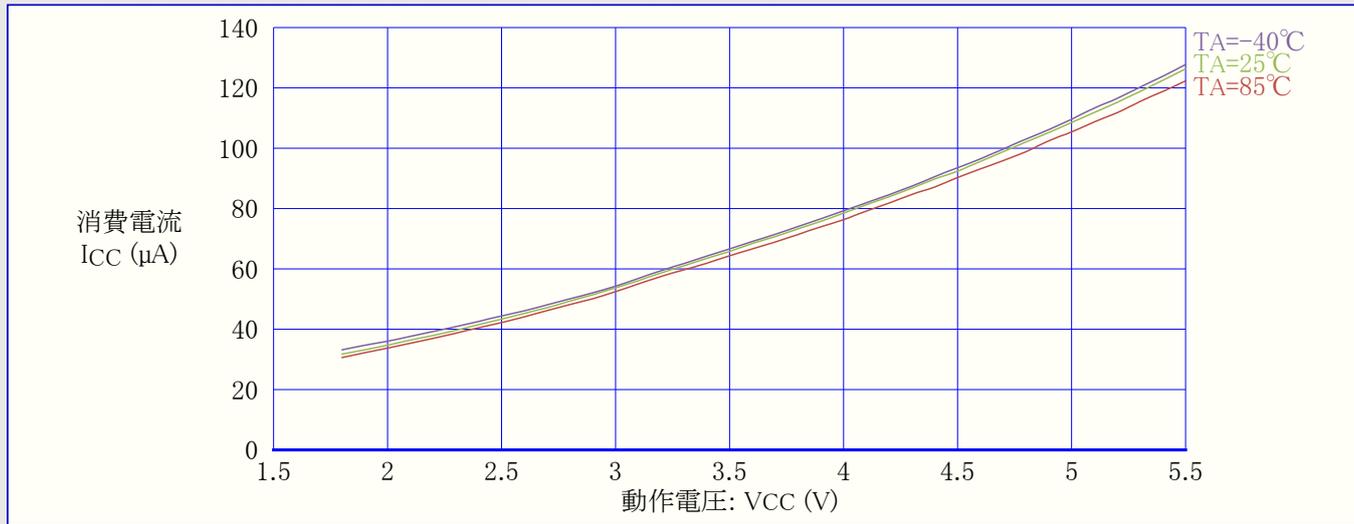


図34-4. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)

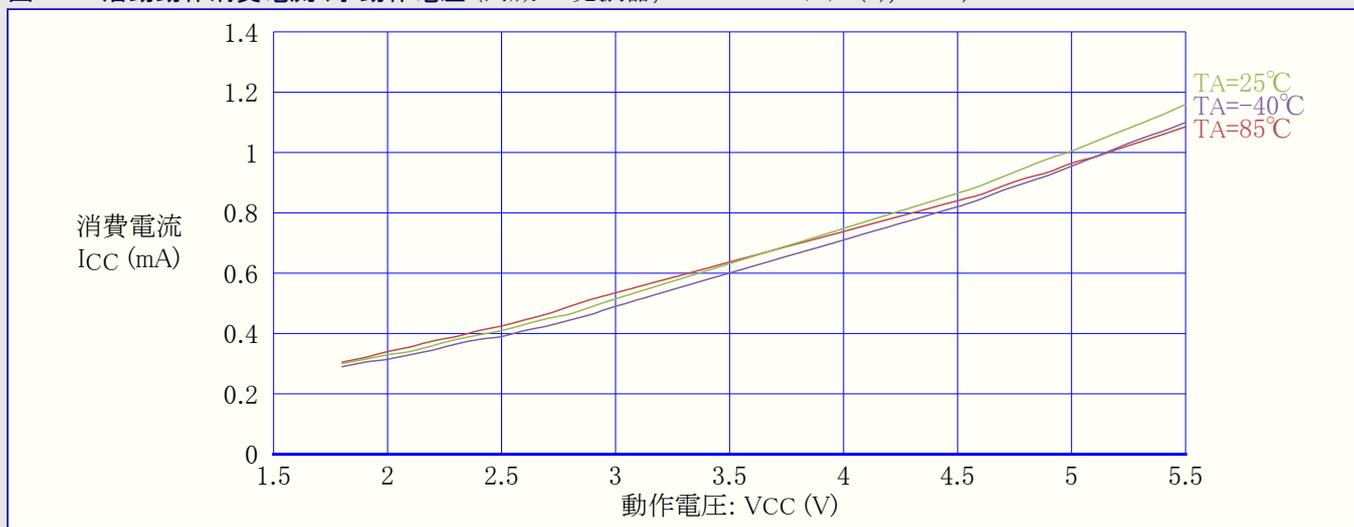


図34-5. 活動動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

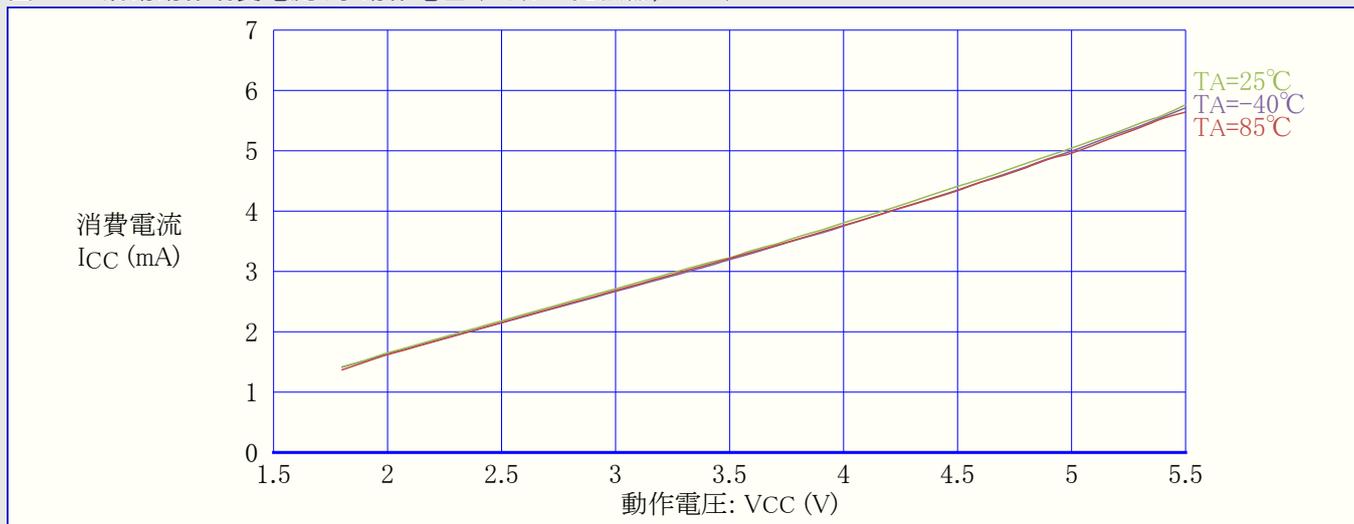
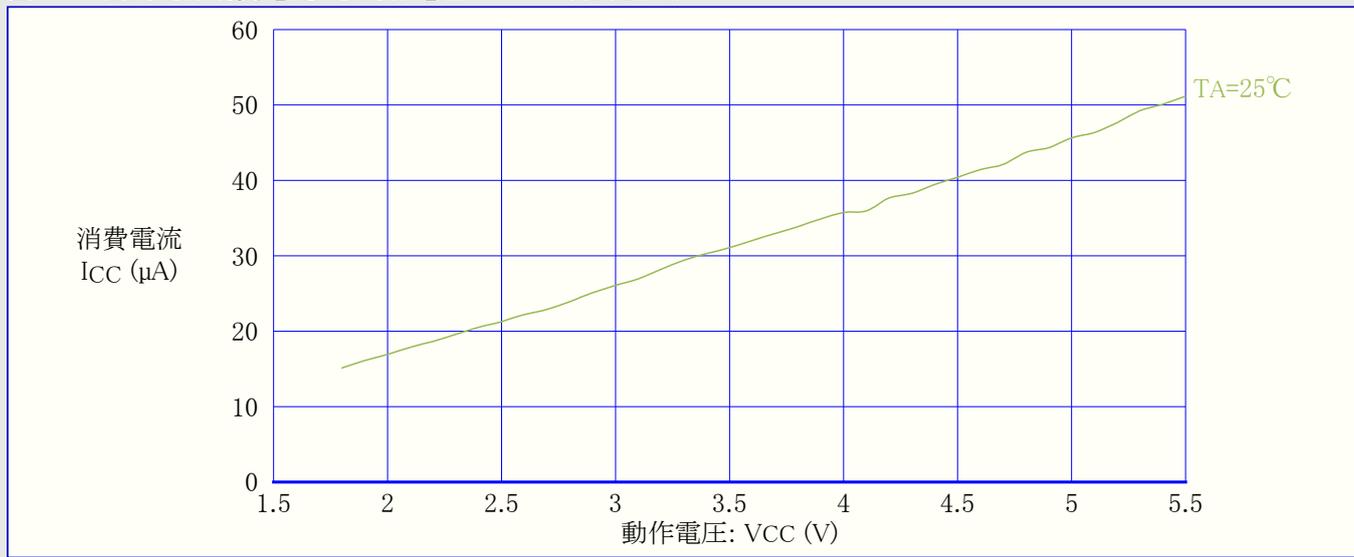


図34-6. 活動動作消費電流 対 動作電圧 (32kHz外部発振器)



### 34.2. アイドル動作消費電流

図34-7. アイドル動作消費電流 対 周波数 (100kHz~1MHz)

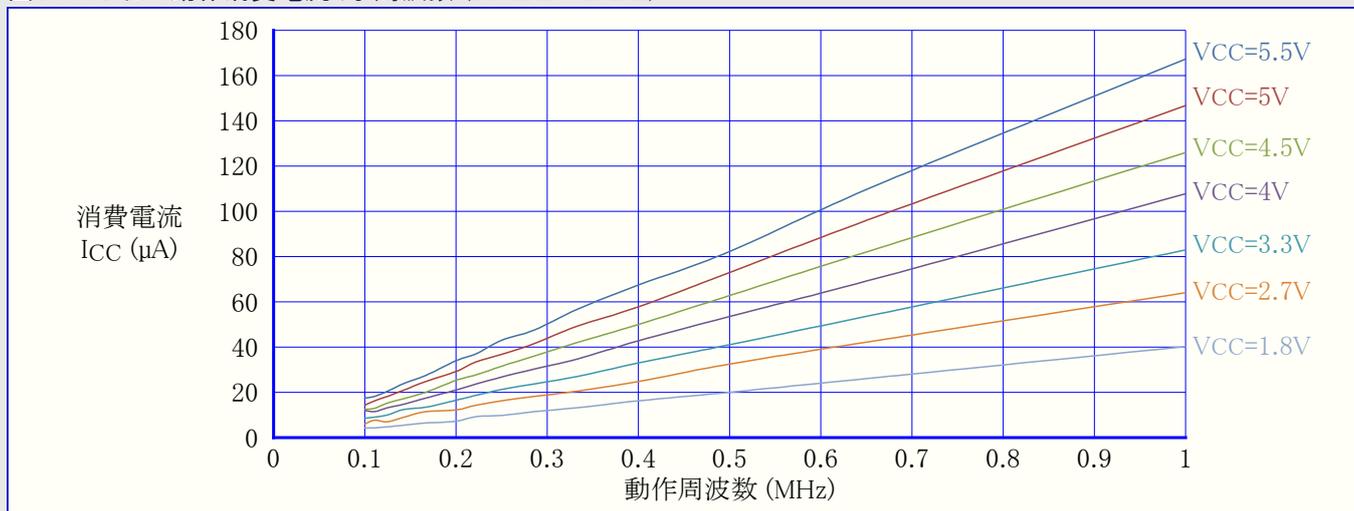


図34-8. アイドル動作消費電流 対 周波数 (1MHz~24MHz)

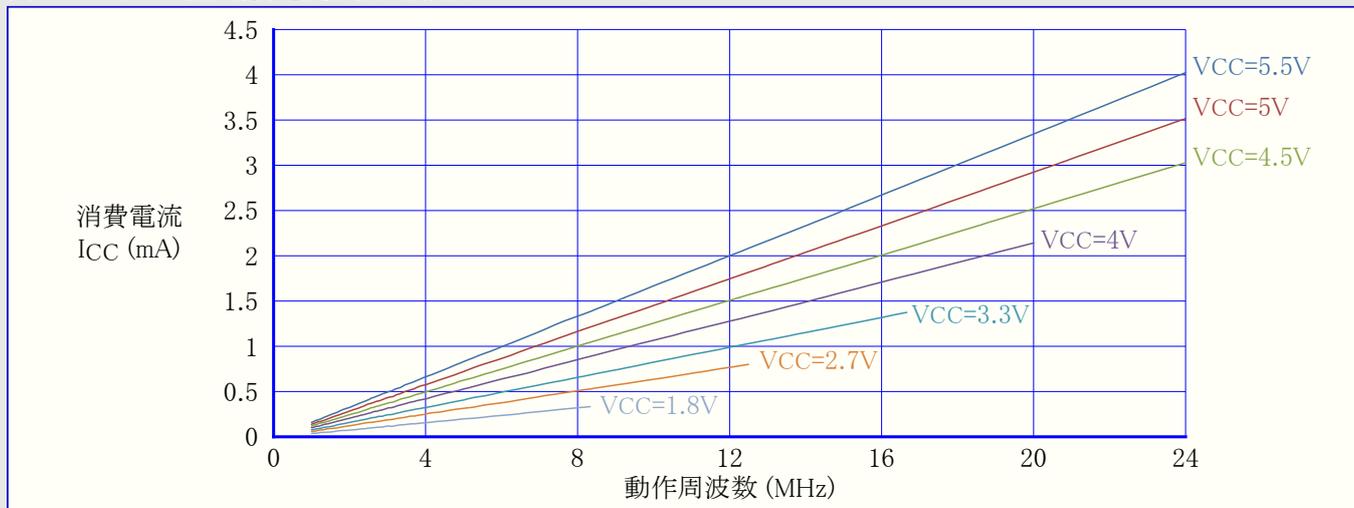


図34-9. アイドル動作消費電流 対 動作電圧 (内蔵WDT発振器,128kHz)

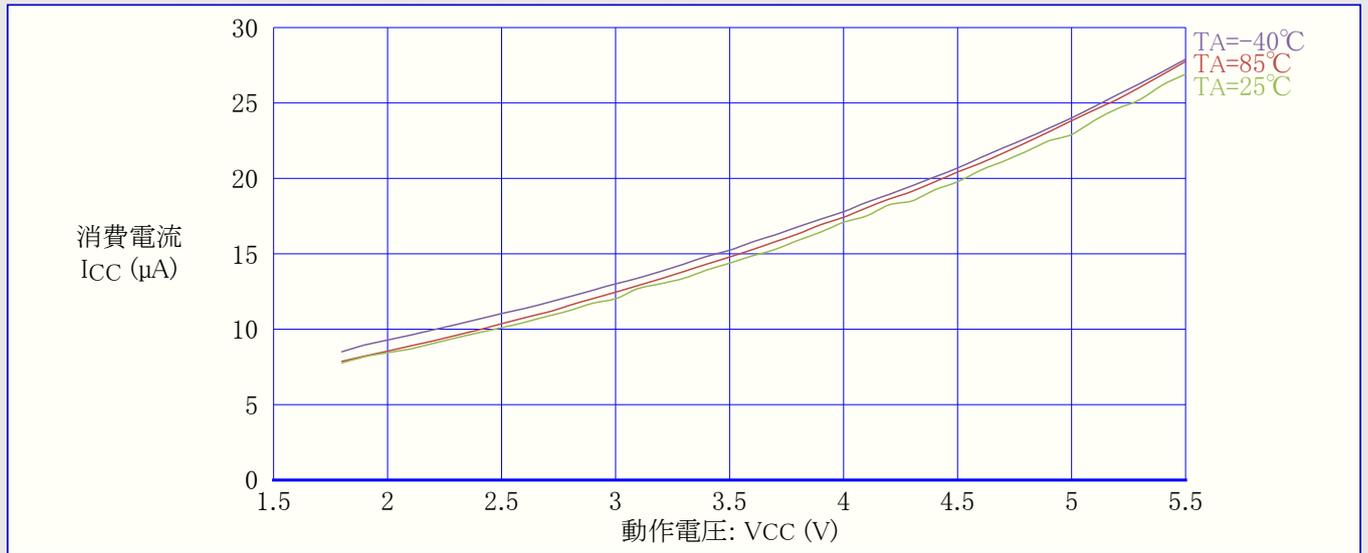


図34-10. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,CKDIV8=プログラム(0),1MHz)

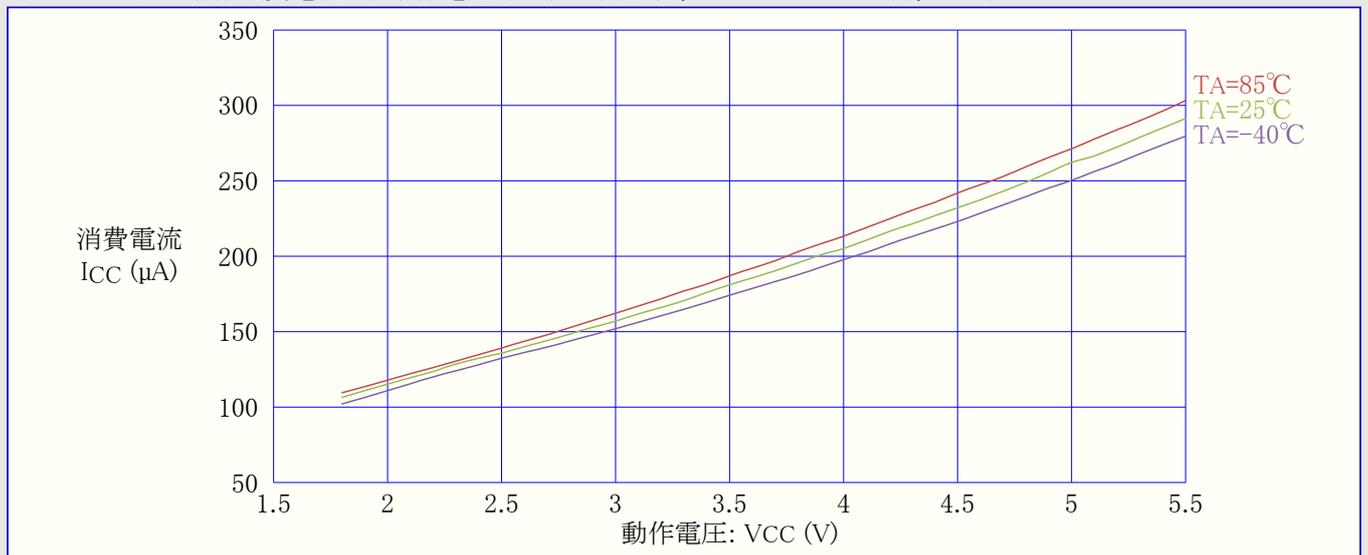


図34-11. アイドル動作消費電流 対 動作電圧 (内蔵RC発振器,8MHz)

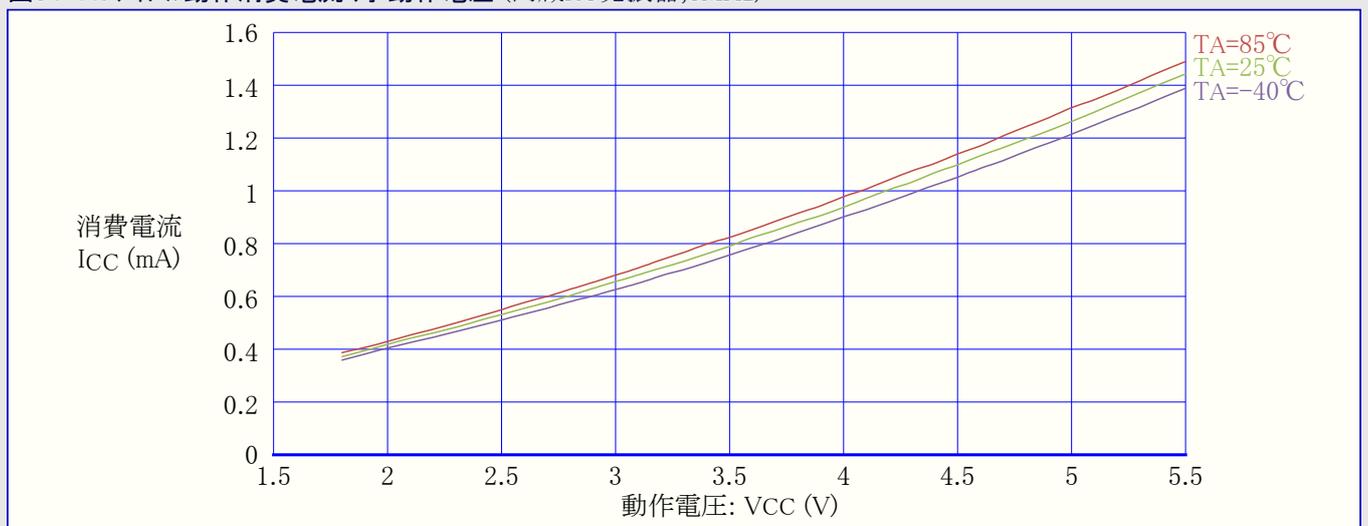
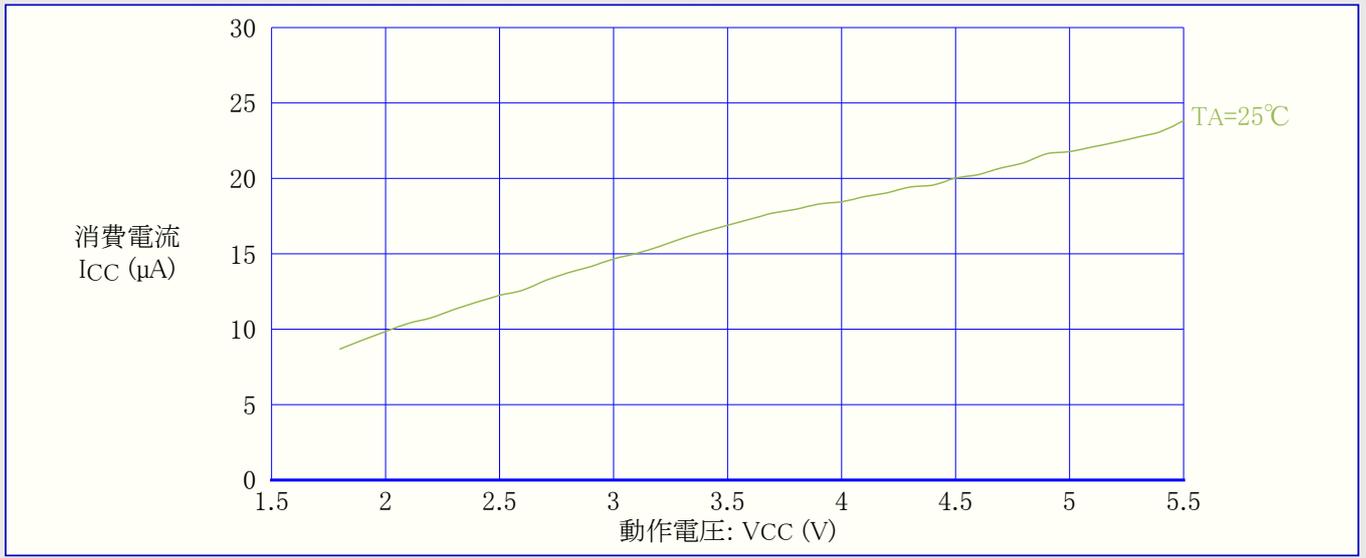


図34-12. アイドル動作消費電流 対 動作電圧 (32kHz外部発振器)



### 34.3. 周辺機能部供給電流

以下の表と式は活動動作とアイドル動作で個別周辺機能部に対する追加消費電流の計算に使えます。周辺機能部の許可や禁止は電力削減レジスタによって制御されます。詳細については「PRR – 電力削減レジスタ」をご覧ください。

表34-1. 各部追加消費電流 (絶対値: µA)

PRR内ビット	1MHz,2V	4MHz,3V	8MHz,5V
PRUSART0	8.0	51	220
PRTWI	12	75	315
PRTIM2	11	72	300
PRTIM1	5.0	32	130
PRTIM0	4.0	24	100
PRSPI	15	95	400
PRADC	12	75	315

表34-2. 各部追加消費電流 (相対値:%)

PRR内ビット	活動動作(図34-1,図34-2)	アイドル動作(図34-7,図34-8)
PRUSART0	3.3	18
PRTWI	4.8	26
PRTIM2	4.7	25
PRTIM1	2.0	11
PRTIM0	1.6	8.5
PRSPI	6.1	33
PRADC	4.9	26

表34-1.で一覽される以外のVCCと周波数設定については表34-2.からの数値を元に代表的な消費電流を計算できます。

例1: VCC=3V, f=1MHzでUSART0, TWI, タイマ/カウンタ1が許可されたアイドル動作での予測される消費電流を計算します。表34-2.のアイドル動作列からUSART0が18%、TWIが26%、タイマ/カウンタ1が11%追加する必要があります。図34-7.を読み、VCC=3V, f=1MHzでのアイドル動作消費電流が約0.075mAであることを得ます。USART0, TWI, タイマ/カウンタ1許可のアイドル動作での総消費電流を得ます。

$$\text{総消費電流} = 0.075\text{mA} \times (1 + 0.18 + 0.26 + 0.11) \approx 0.116\text{mA}$$

例2: 例1と同じ条件ですが、代わりに活動動作です。表34-2.の活動動作列からUSART0が3.3%、TWIが4.8%、タイマ/カウンタ1が2.0%追加する必要があります。図34-1.を読み、VCC=3V, f=1MHzでの活動動作消費電流が約0.42mAであることを得ます。USART0, TWI, タイマ/カウンタ1許可の活動動作での総消費電流を得ます。

$$\text{総消費電流} = 0.42\text{mA} \times (1 + 0.033 + 0.048 + 0.02) \approx 0.46\text{mA}$$

例3: VCC=3.6V, f=10MHzで全周辺機能部許可の活動動作での予測消費電流を計算します。周辺機能部を除く活動動作消費電流が約4.0mAであることを得ます(図34-2.より)。そして表34-2.の活動動作列からの数値を使い、総消費電流を得ます。

$$\text{総消費電流} = 4.0\text{mA} \times (1 + 0.033 + 0.048 + 0.047 + 0.02 + 0.016 + 0.061 + 0.049) \approx 5.1\text{mA}$$

### 34.4. パワーダウン動作消費電流

図34-13. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)

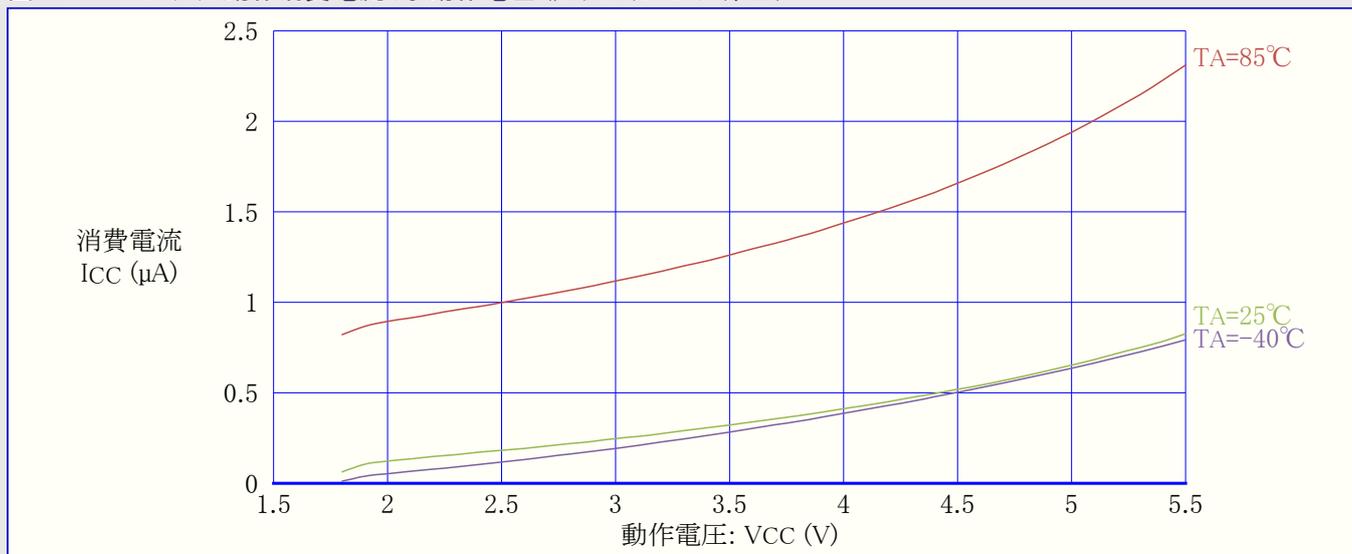
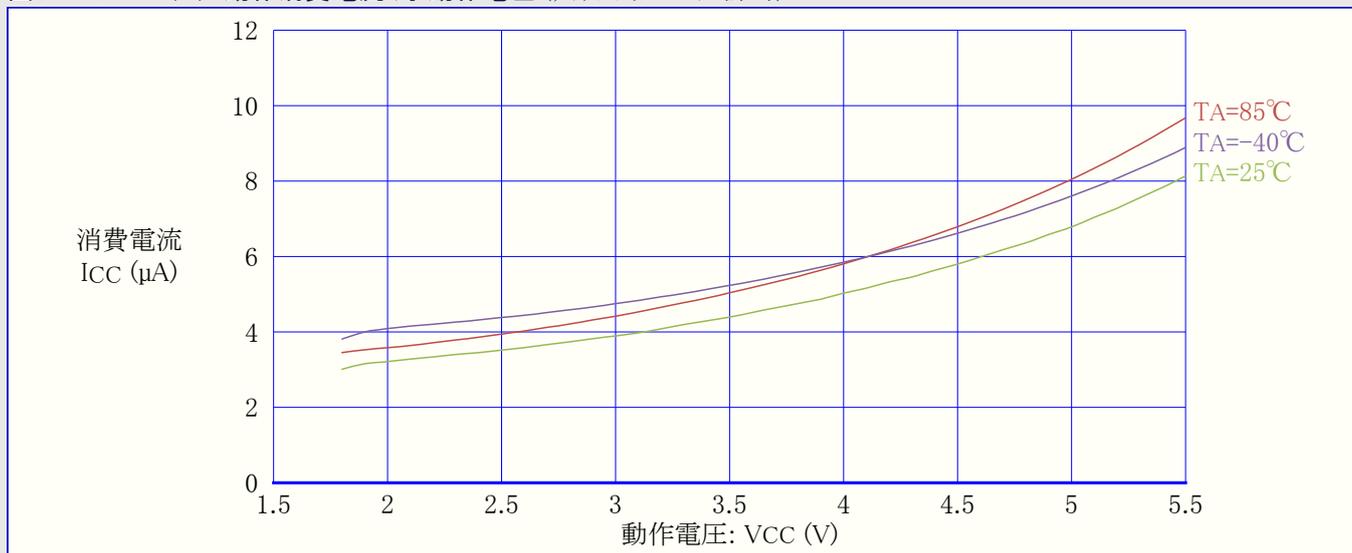
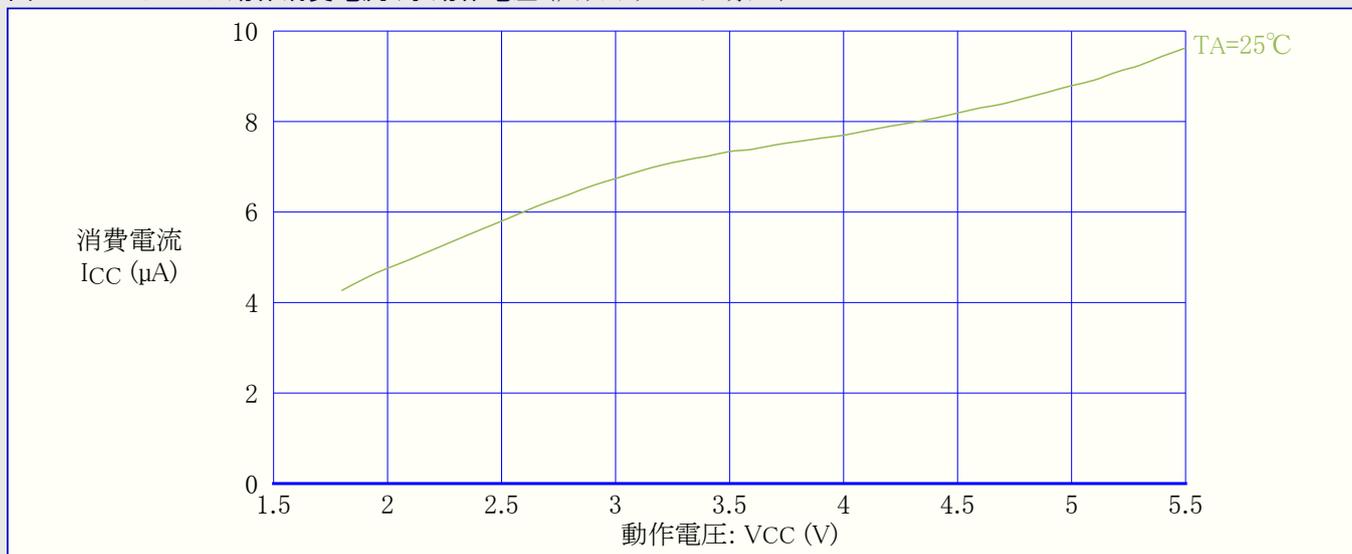


図34-14. パワーダウン動作消費電流 対 動作電圧 (ウォッチドッグ タイマ許可)



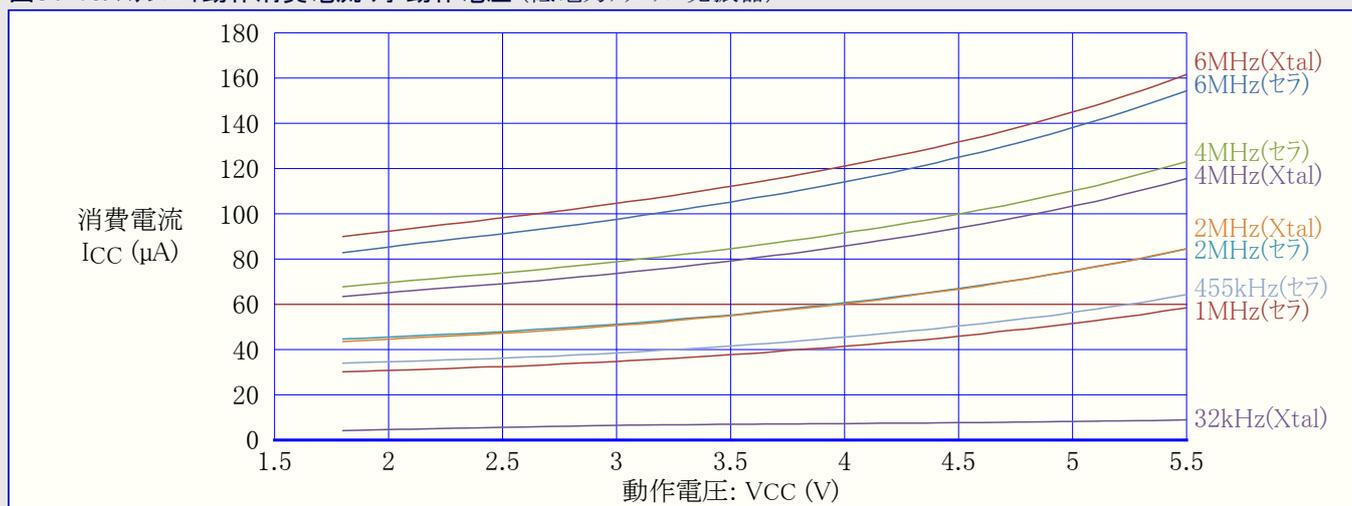
### 34.5. パワーセーブ動作消費電流

図34-15. パワーセーブ動作消費電流 対 動作電圧 (ウォッチドッグ タイマ禁止)



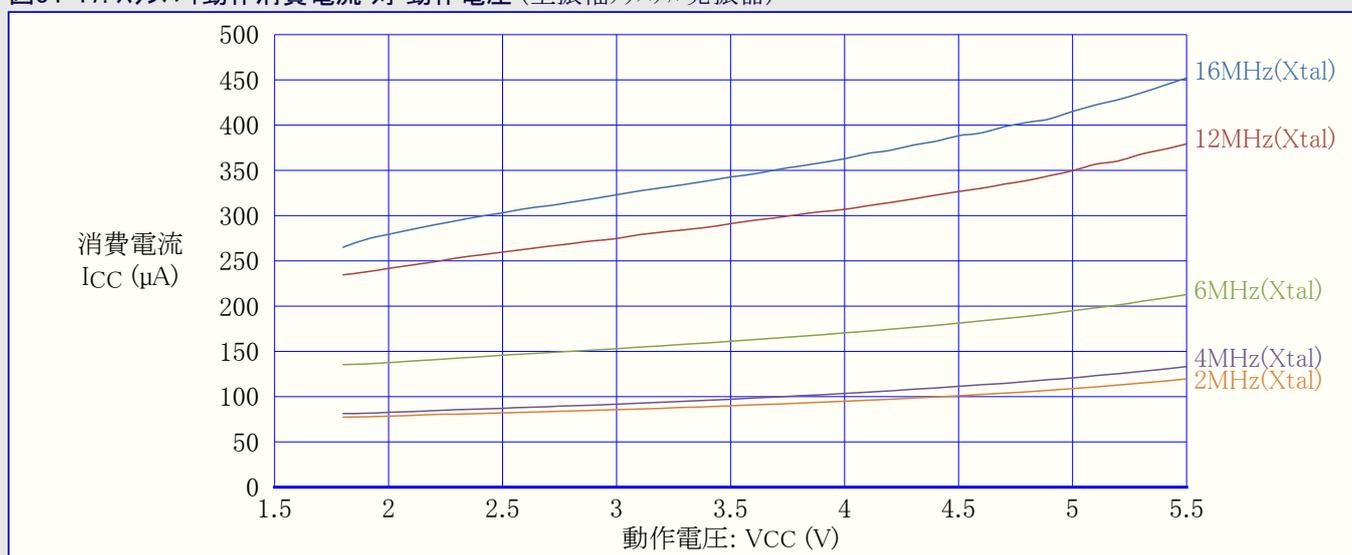
### 34.6. スタンバイ動作消費電流

図34-16. スタンバイ動作消費電流 対 動作電圧 (低電力クリスタル発振器)



注: セラはセラミック振動子

図34-17. スタンバイ動作消費電流 対 動作電圧 (全振幅クリスタル発振器)



### 34.7. ピンプルアップ

図34-18. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

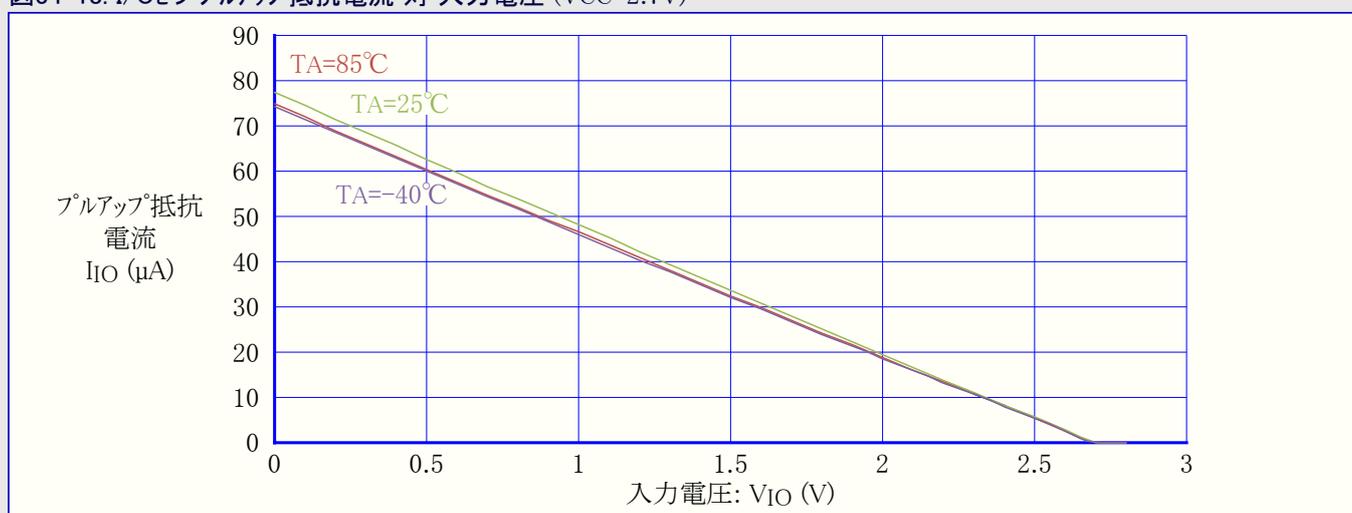


図34-19. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

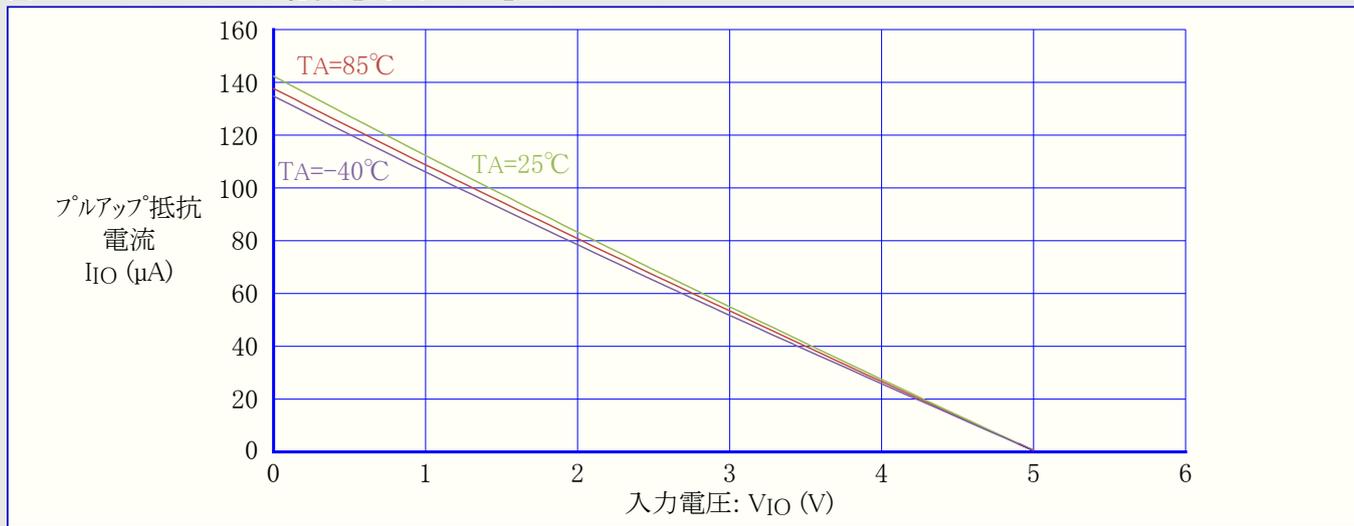


図34-20. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

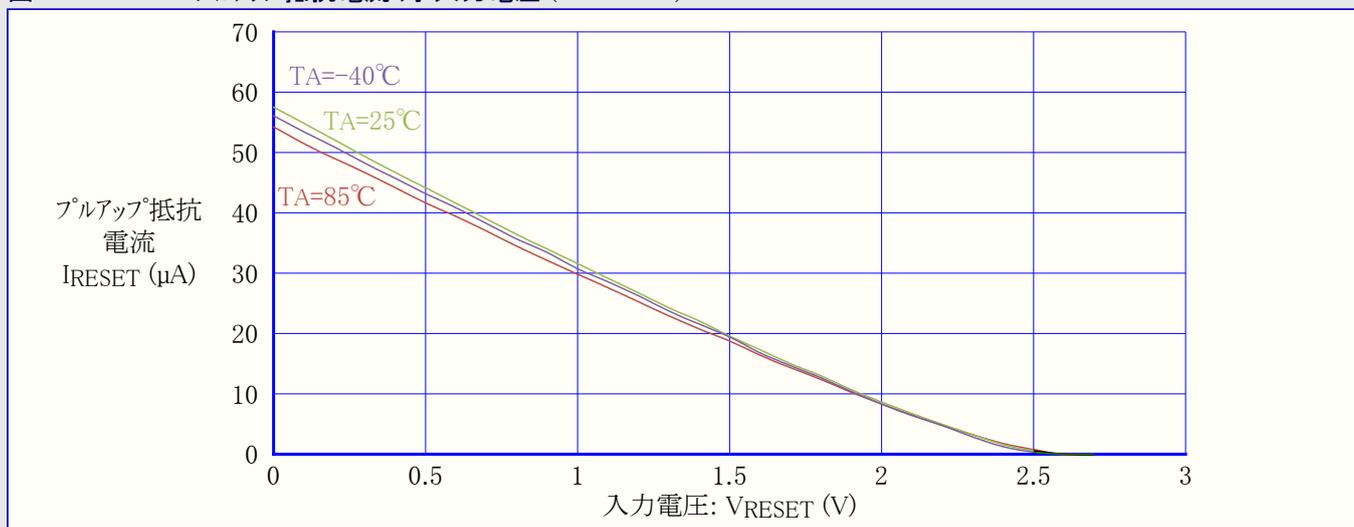
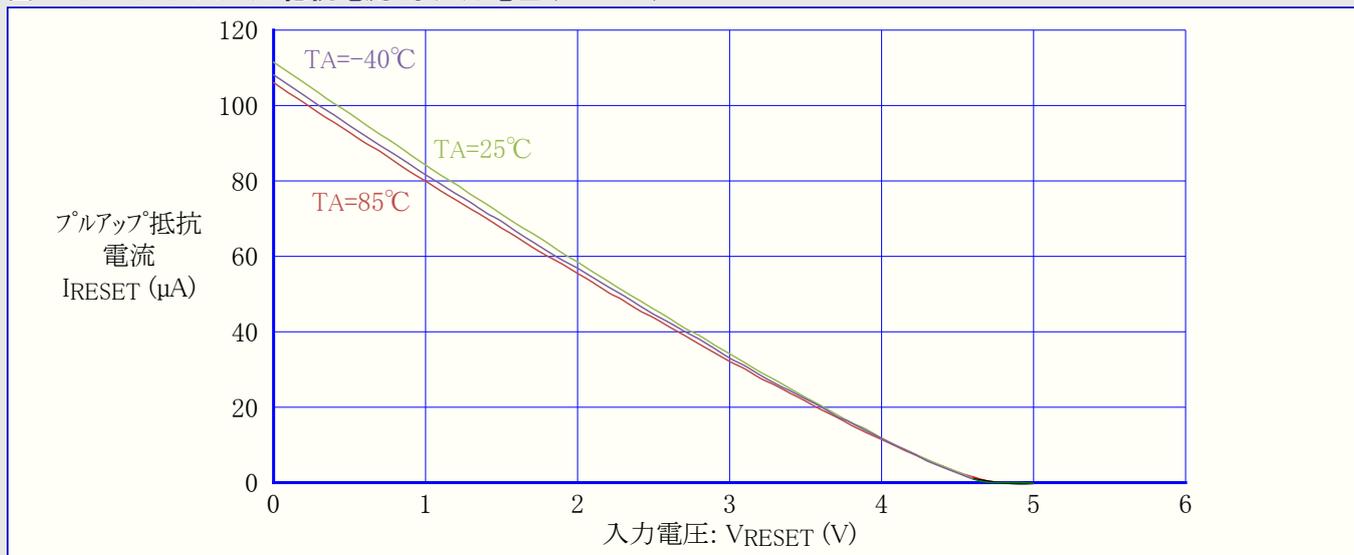


図34-21. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)



### 34.8. ピン駆動能力

図34-22. I/Oピン吸い込み電流 対 出力電圧 (VCC=1.8V)

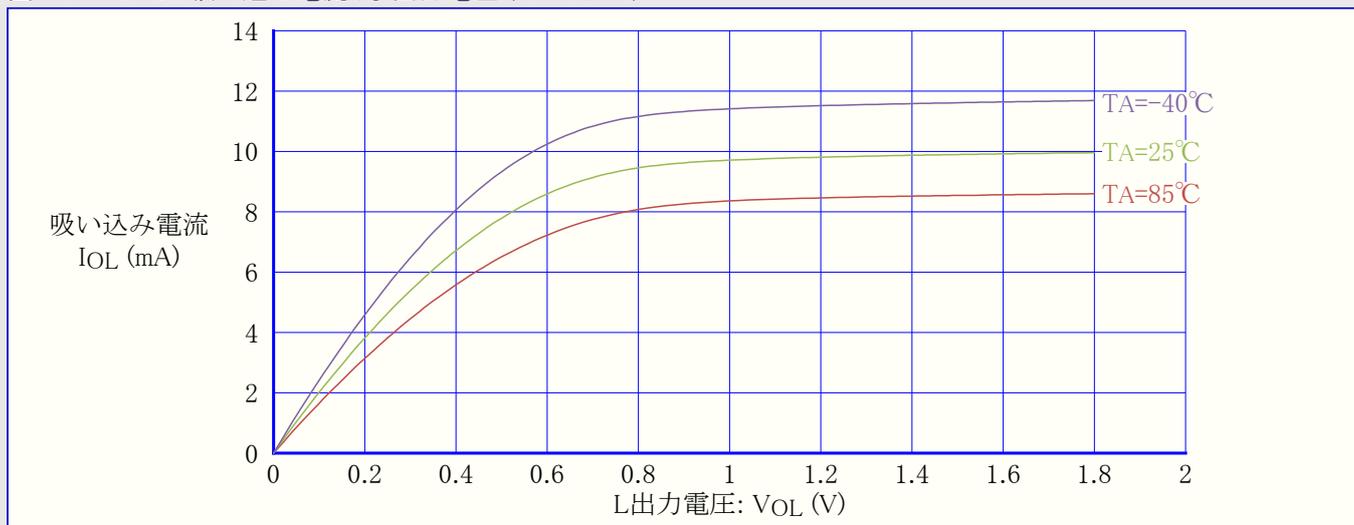


図34-23. I/Oピン吸い込み電流 対 出力電圧 (VCC=2.7V)

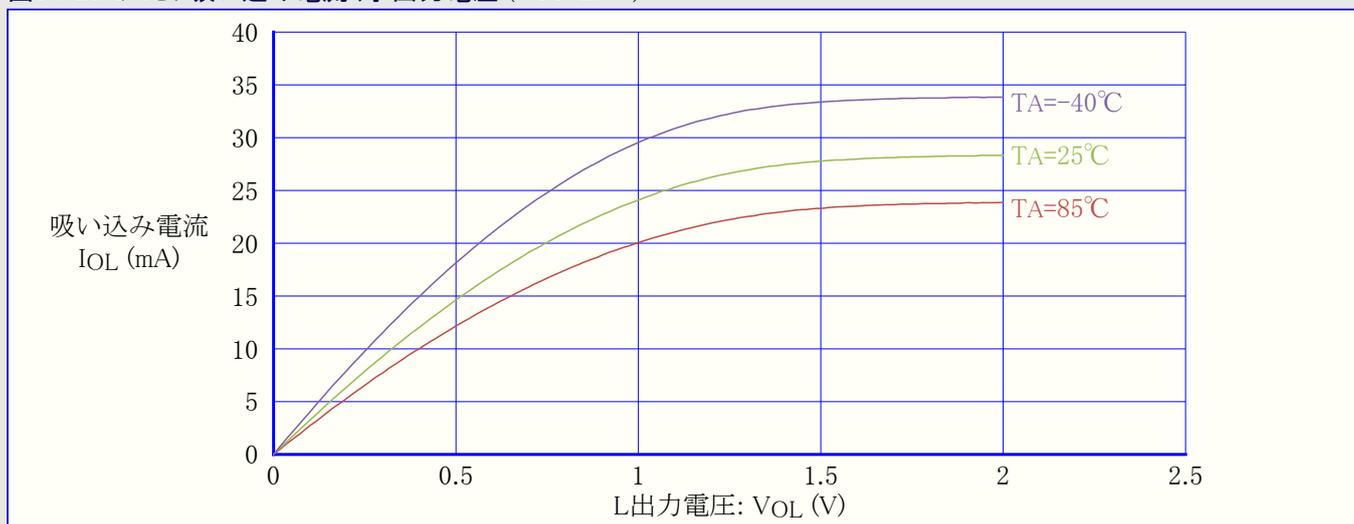


図34-24. I/Oピン吸い込み電流 対 出力電圧 (VCC=5V)

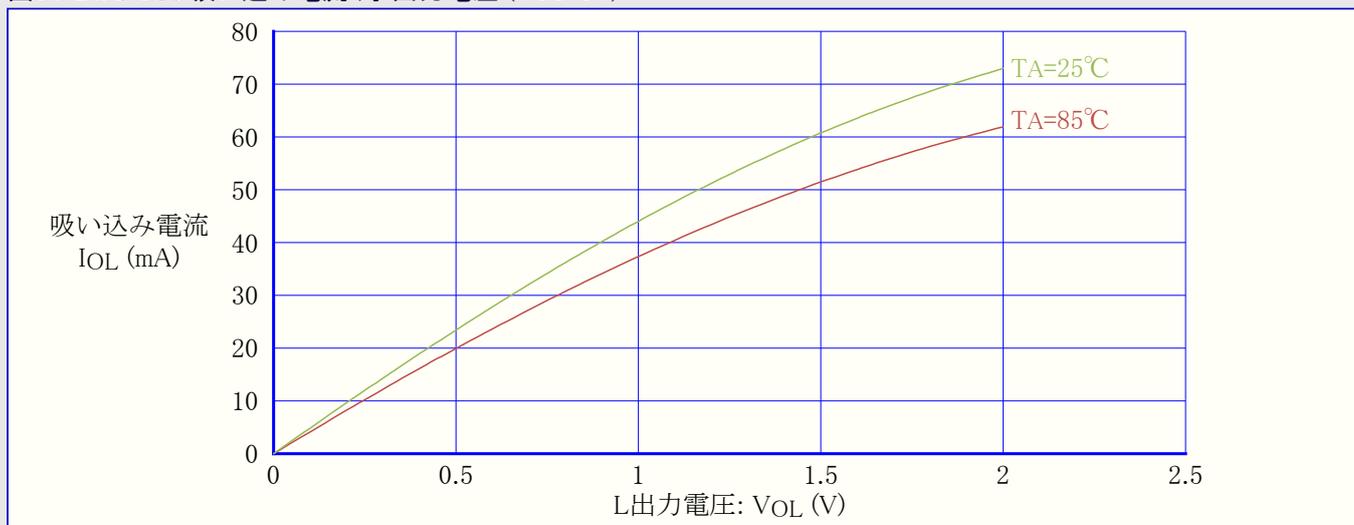


図34-25. I/Oピン吐き出し電流 対 出力電圧 (VCC=1.8V)

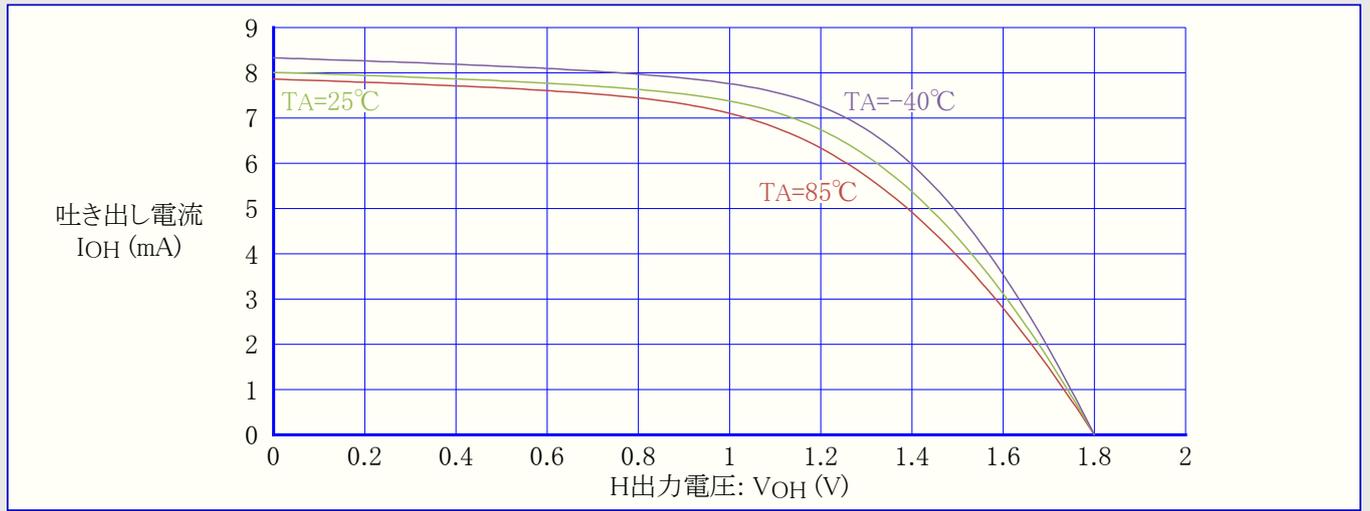


図34-26. I/Oピン吐き出し電流 対 出力電圧 (VCC=2.7V)

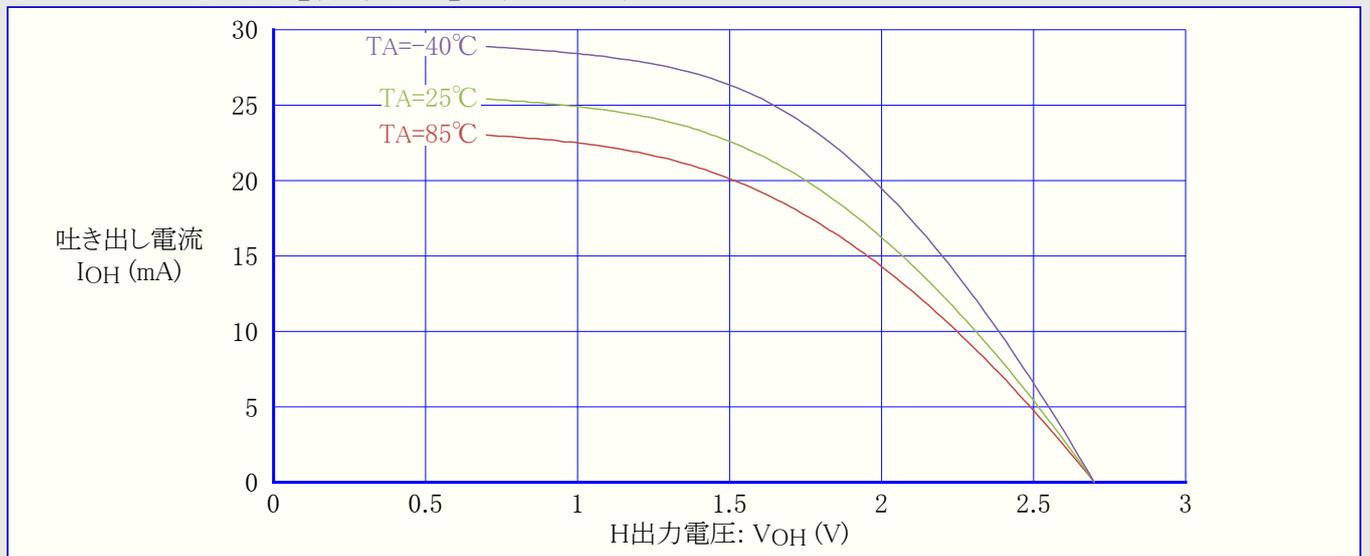
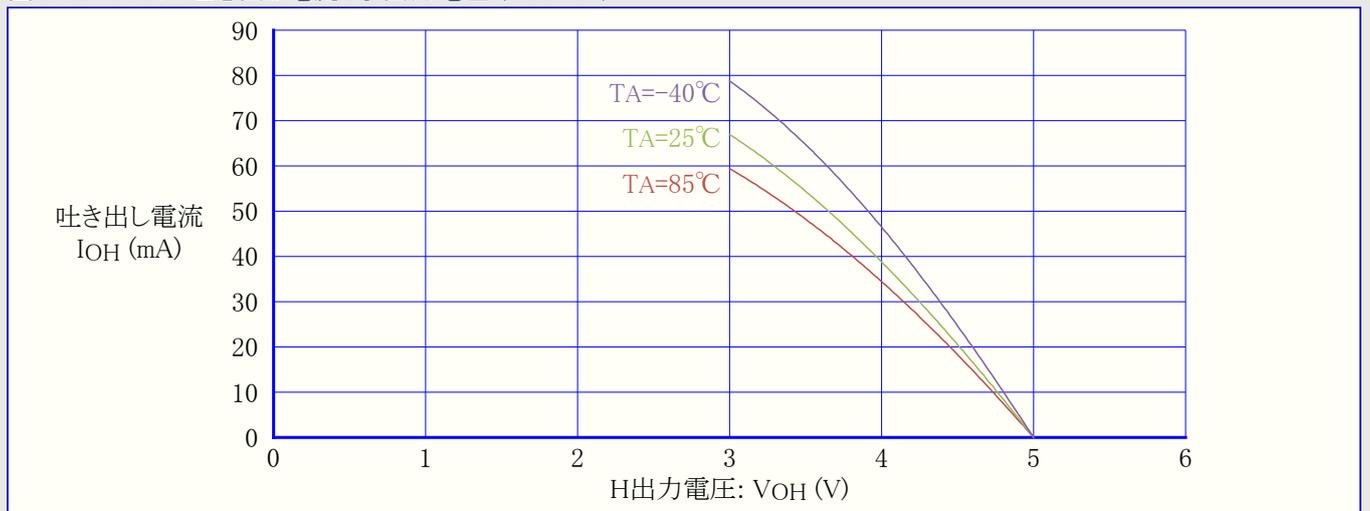


図34-27. I/Oピン吐き出し電流 対 出力電圧 (VCC=5V)



### 34.9. ピン 閾値とヒステリシス

図34-28. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIH, 1読み値)

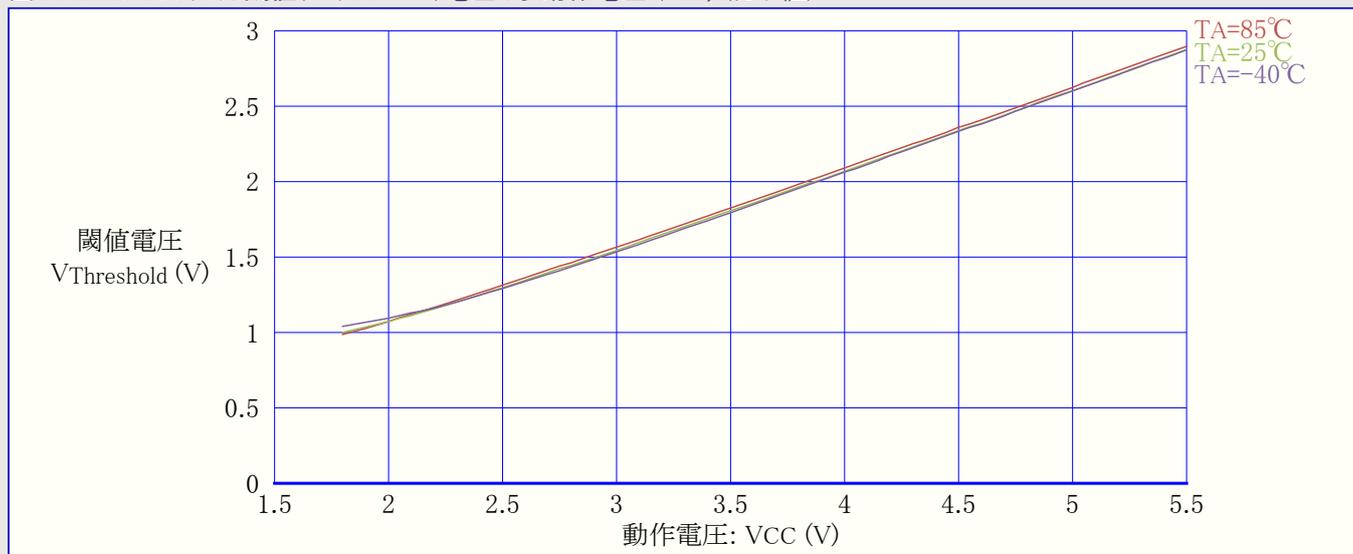


図34-29. I/Oピン入力閾値(スレッショルド)電圧 対 動作電圧 (VIL, 0読み値)

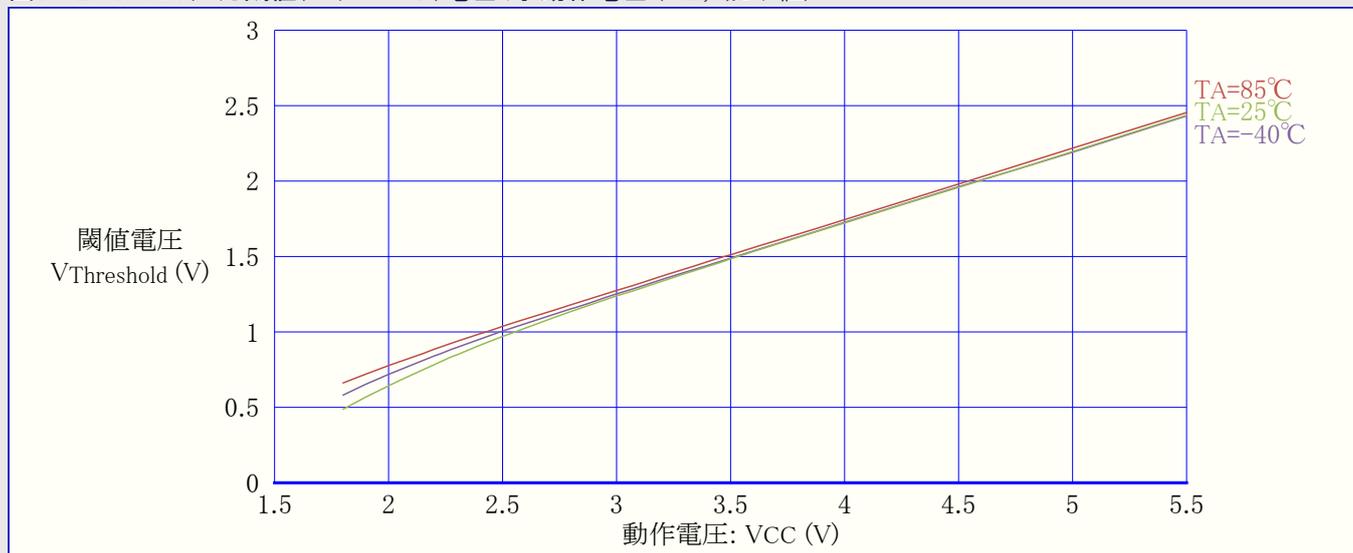


図34-30. I/Oピン入力ヒステリシス電圧 対 動作電圧

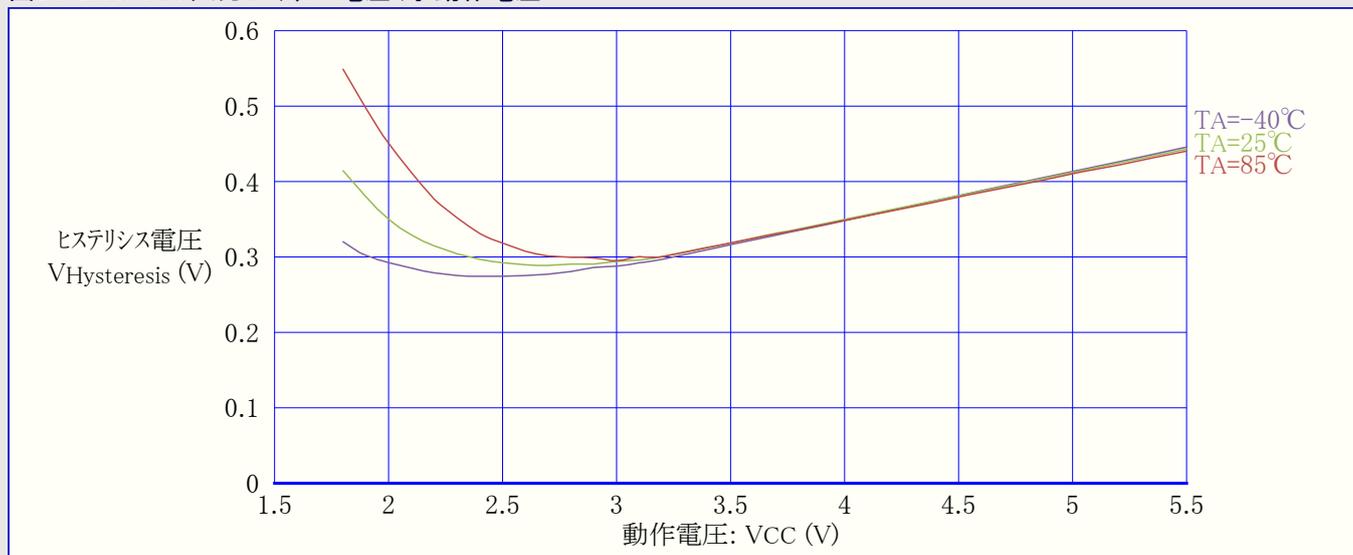


図34-31. RESET入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IH}$ , 1読み値)

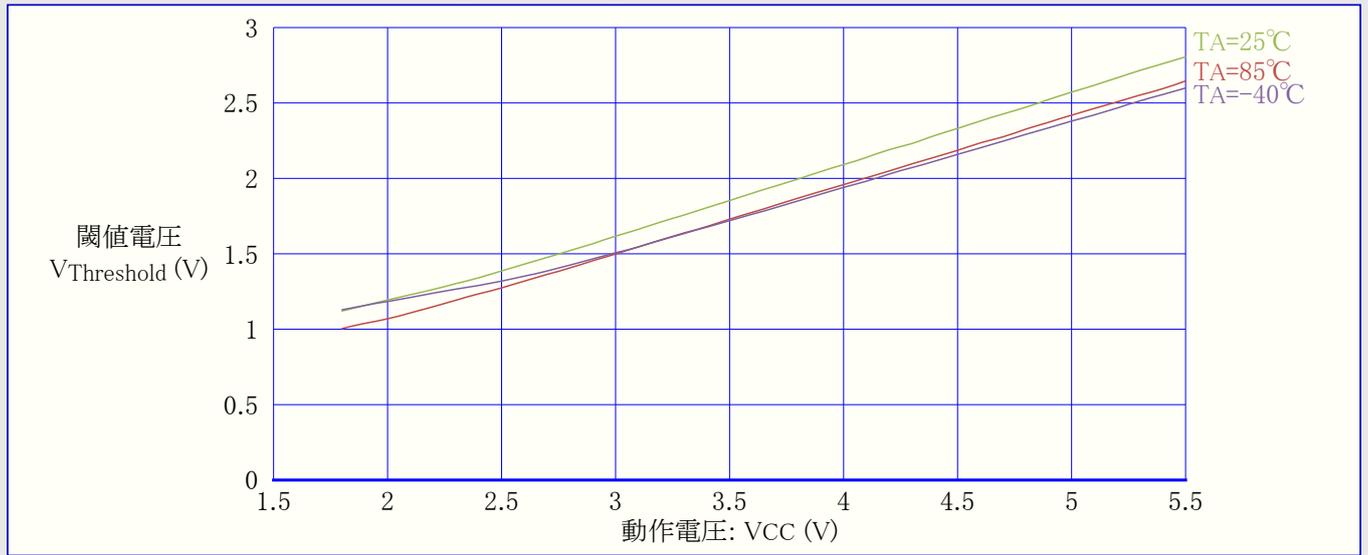


図34-32. RESET入力閾値(スレッシュホールド)電圧 対 動作電圧 ( $V_{IL}$ , 0読み値)

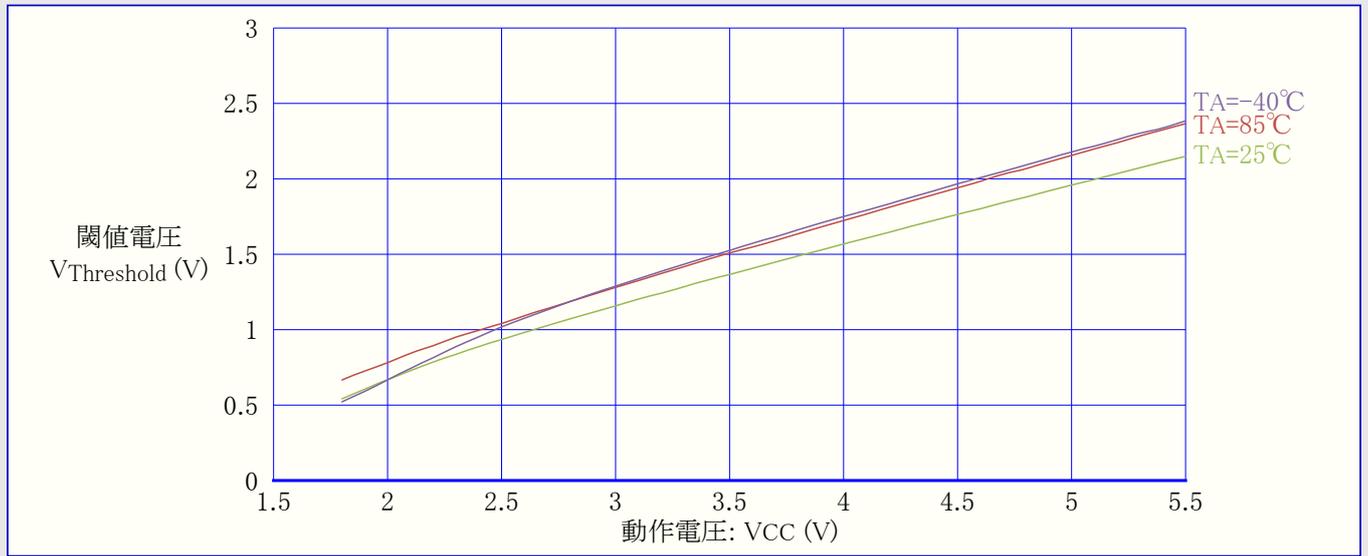
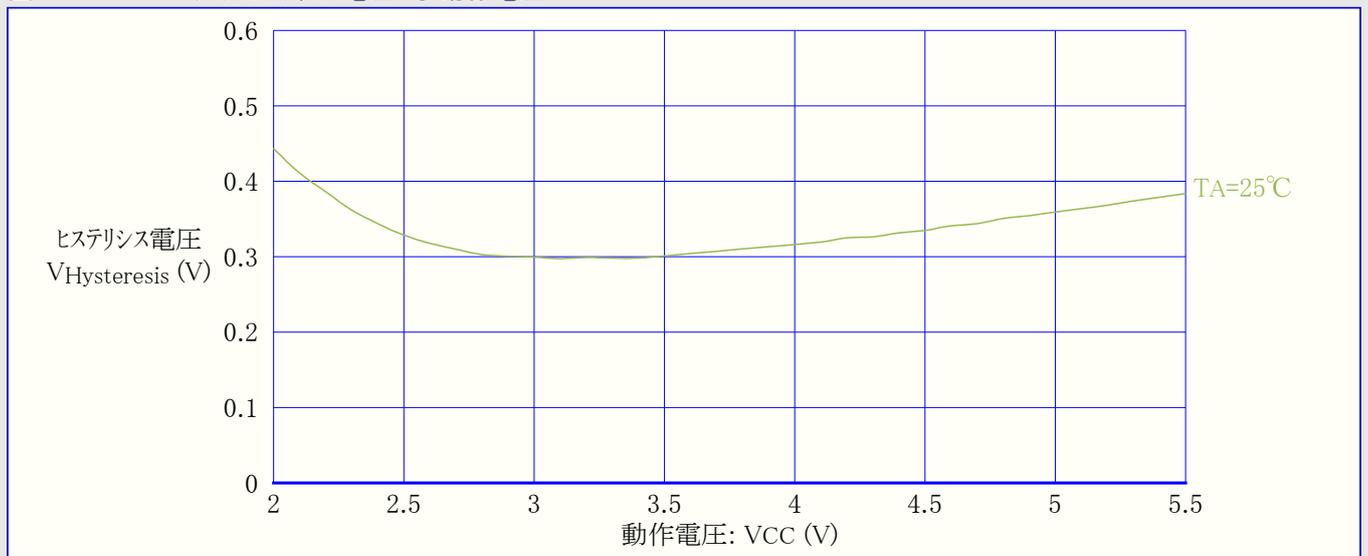


図34-33. RESET入力ヒステリシス電圧 対 動作電圧



### 34.10. 低電圧検出器(BOD)閾値とアナログ比較器変位(オフセット)

図34-34. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧1.8V)

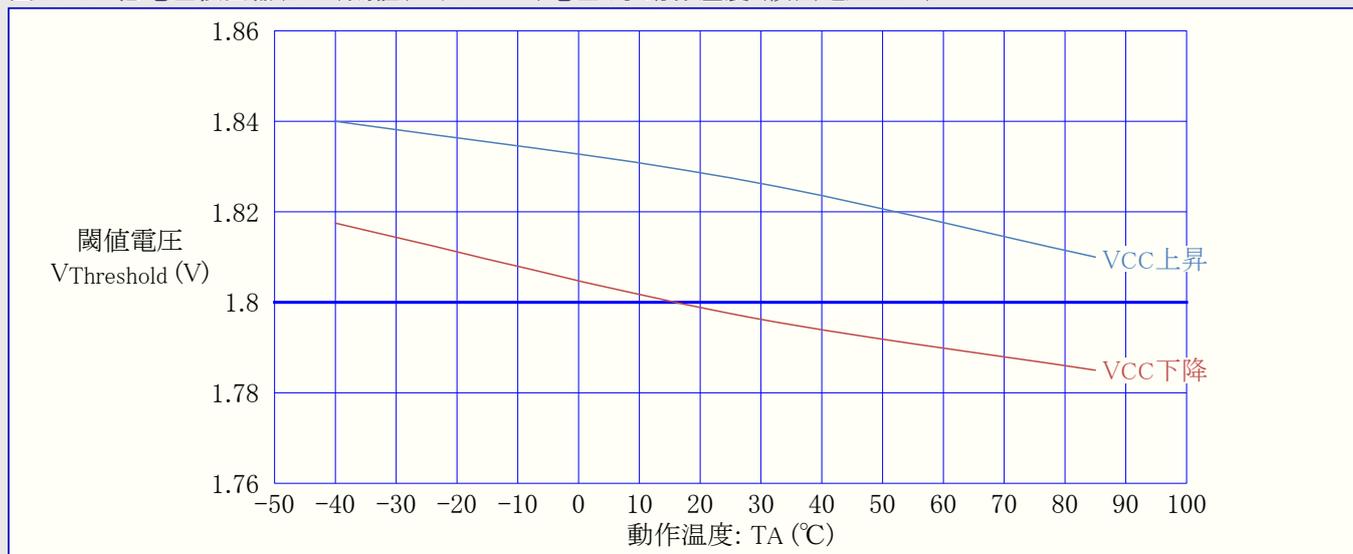


図34-35. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧2.7V)

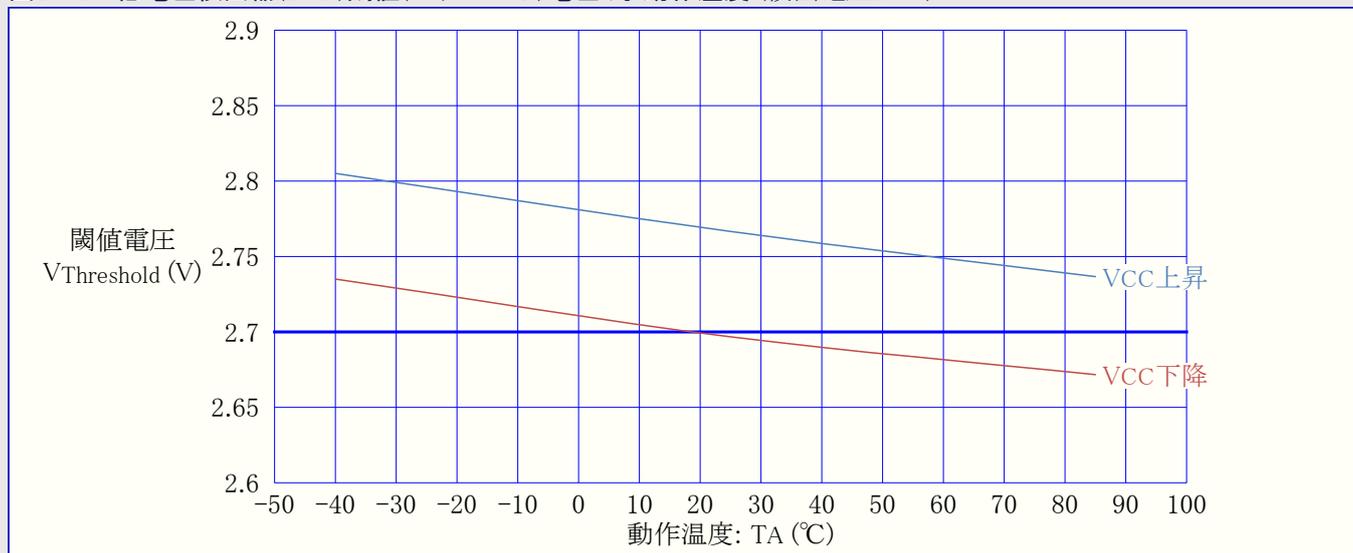


図34-36. 低電圧検出器(BOD)閾値(スレッシュホールド)電圧 対 動作温度 (検出電圧4.3V)

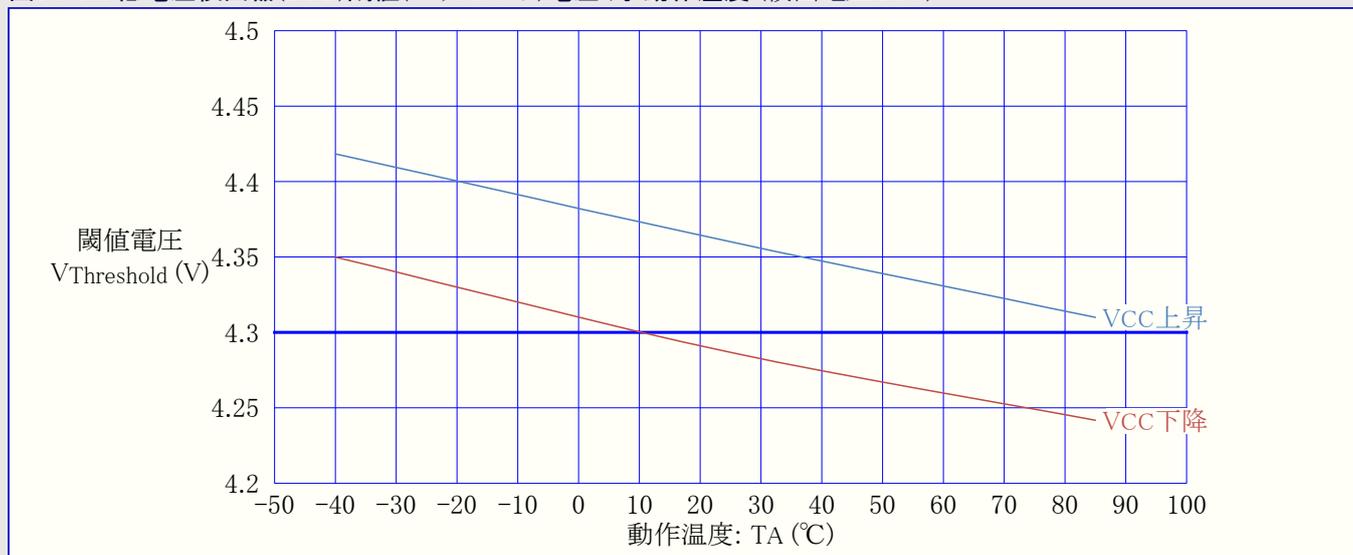


図34-37. 内部基準(バンドギャップ)電圧 対 動作電圧

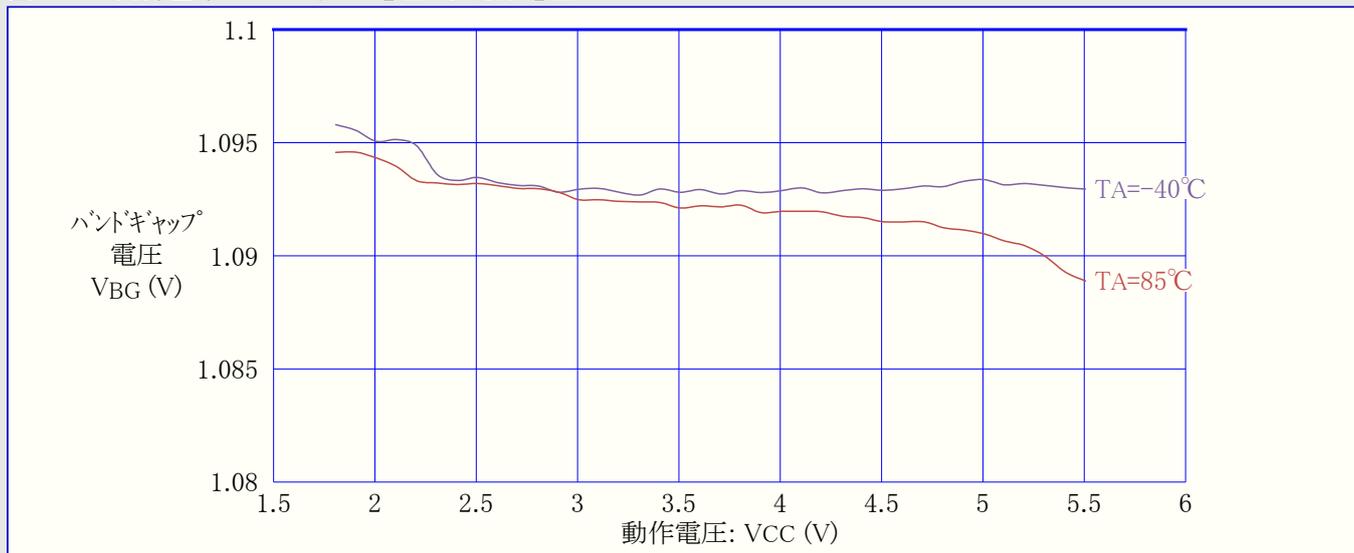


図34-38. アナログ比較器変位(オフセット)電圧 対 同相電圧 (VCC=2.7V)

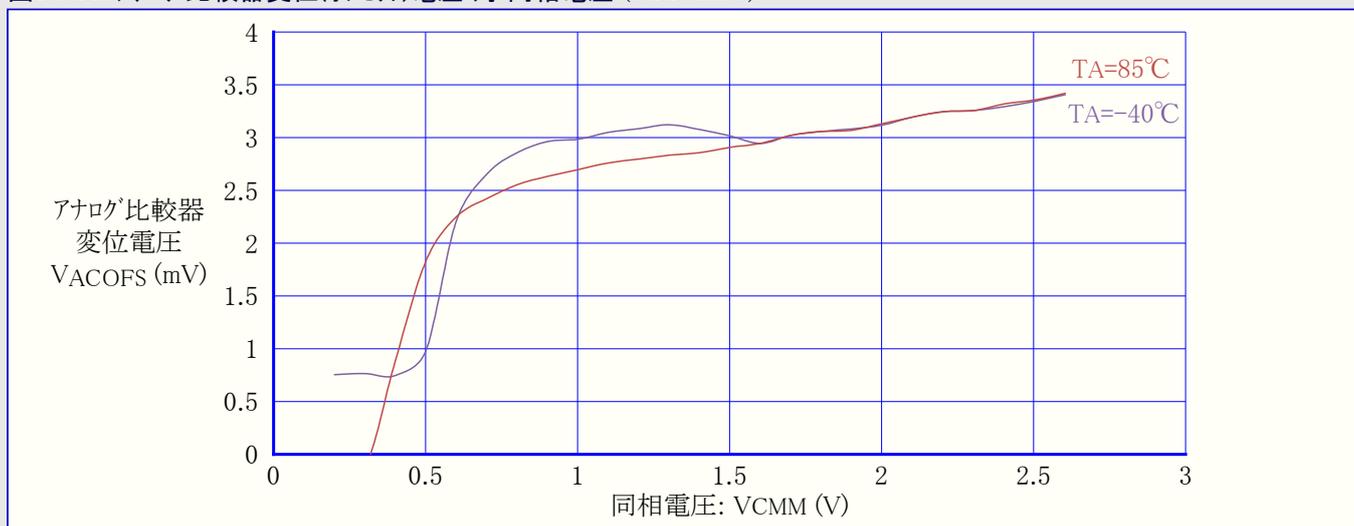
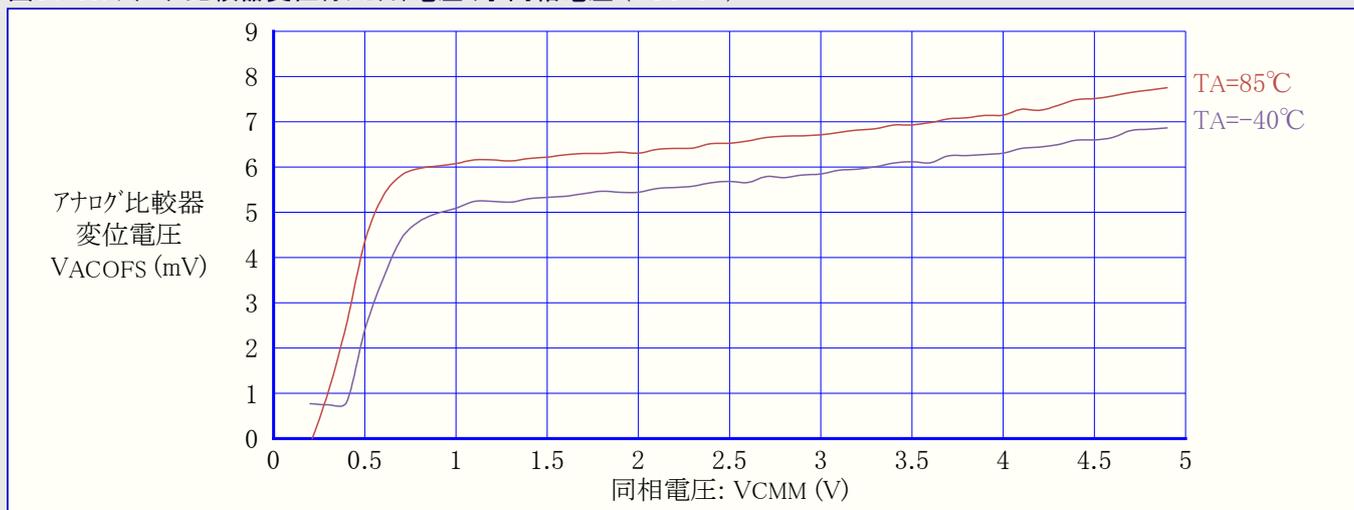


図34-39. アナログ比較器変位(オフセット)電圧 対 同相電圧 (VCC=5V)



### 34.11. 内部発振器周波数

図34-40. ウォッチドッグ発振器周波数 対 動作電圧

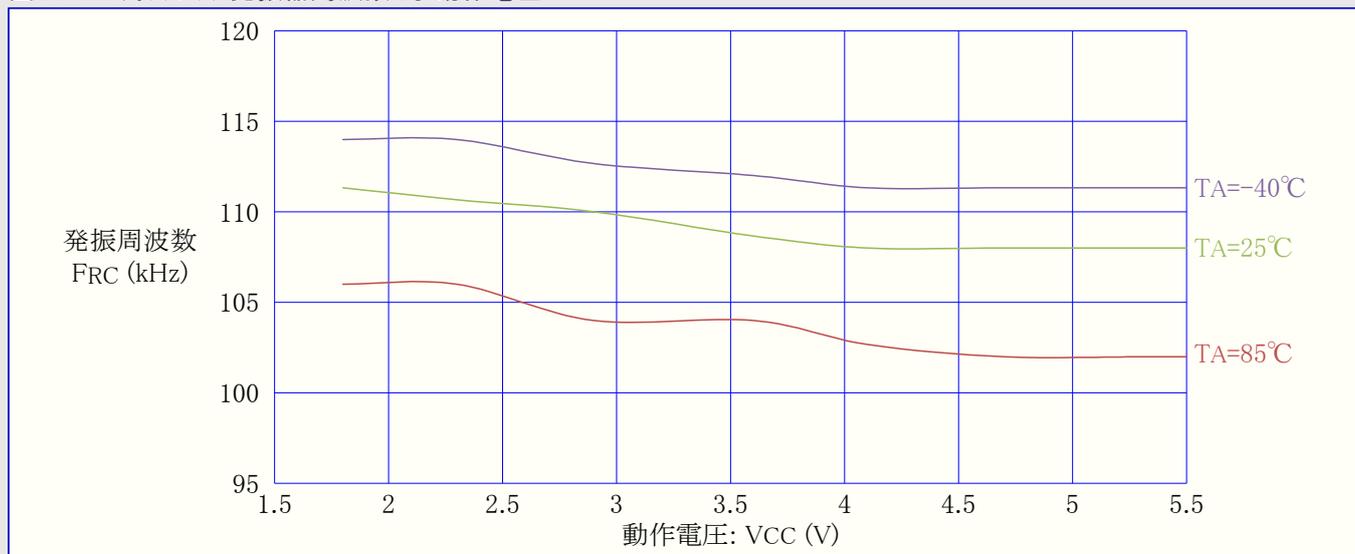


図34-41. 校正済み8MHz内蔵RC発振器周波数 対 動作電圧

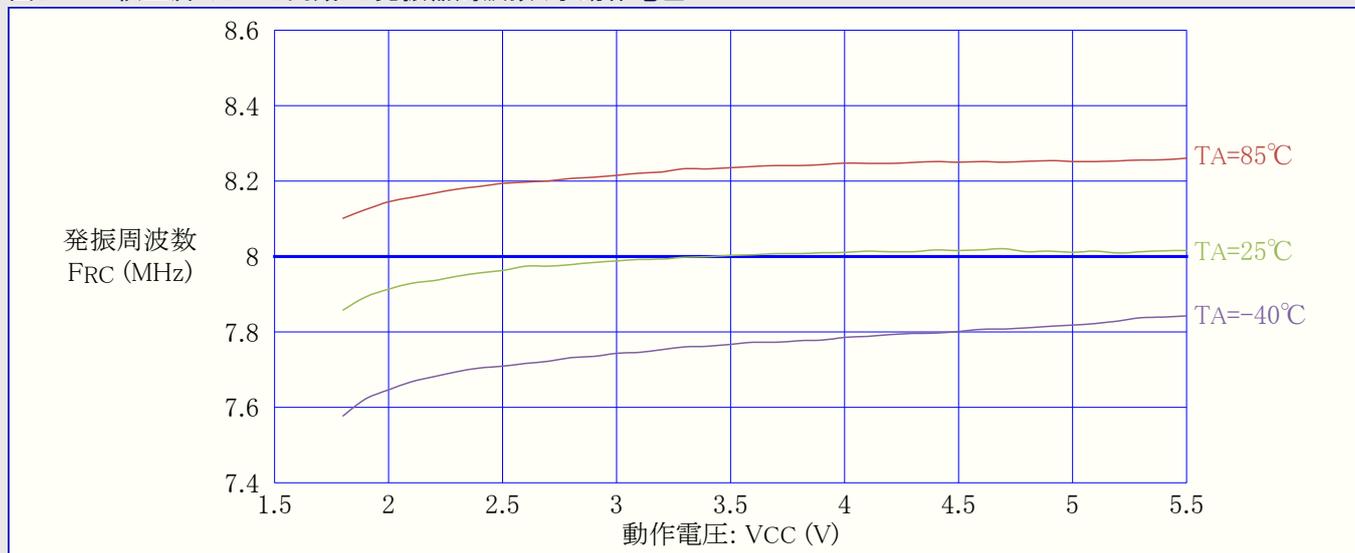


図34-42. 校正済み8MHz内蔵RC発振器周波数 対 動作温度

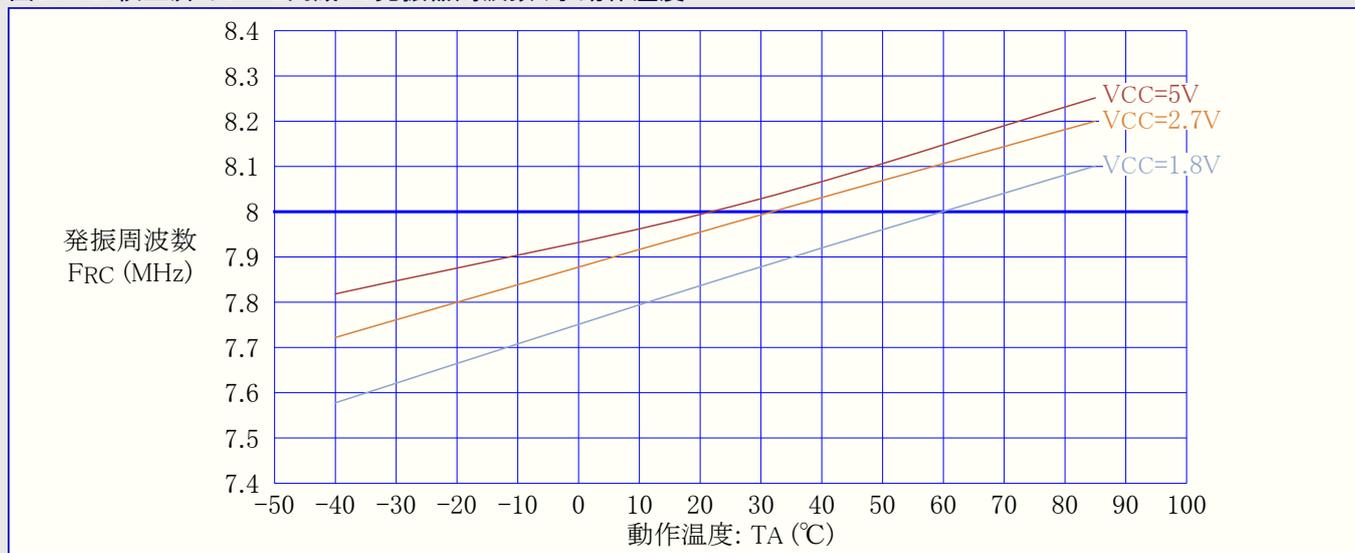
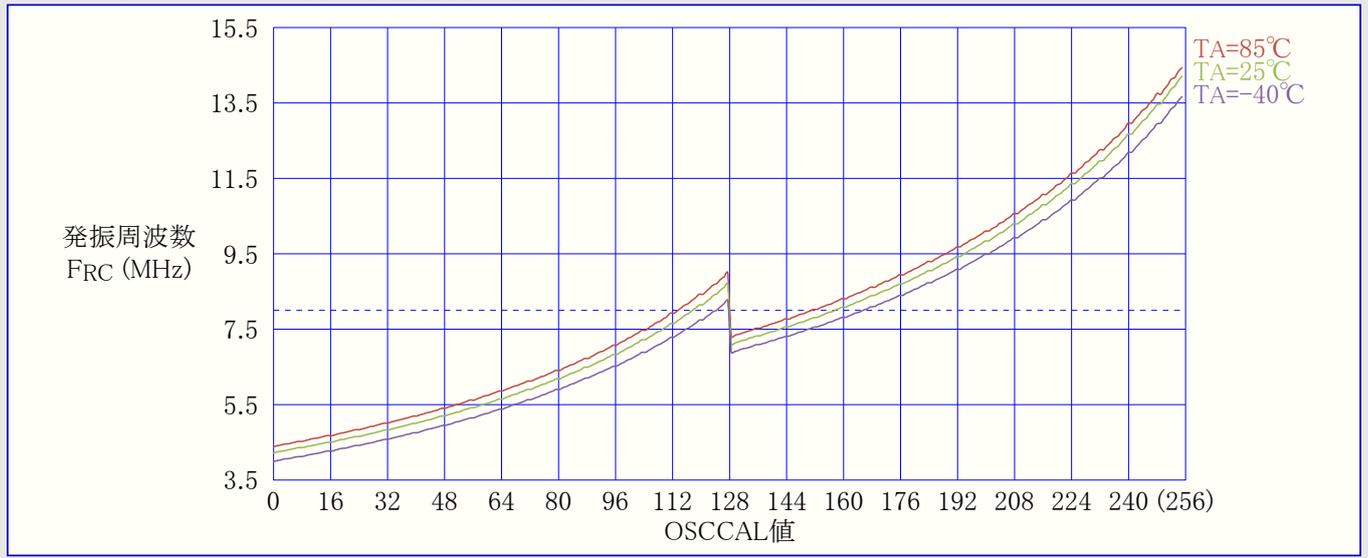


図34-43. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値



### 34.12. 周辺機能部消費電流

図34-44. A/D変換器消費電流 対 動作電圧 (AREF=AVCC)

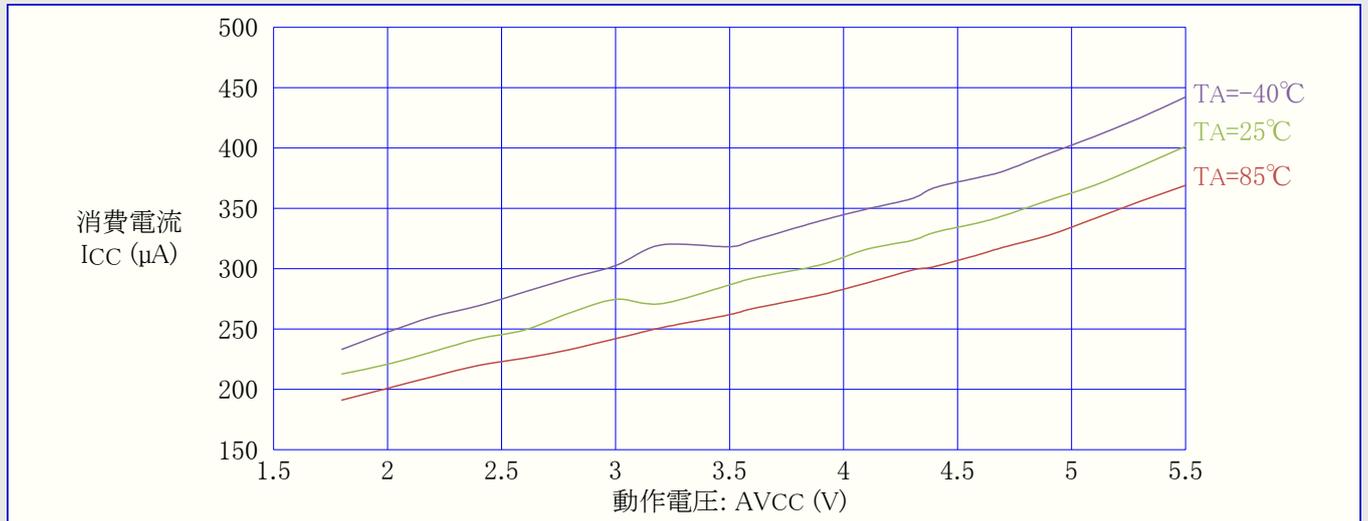


図34-45. 外部基準電圧(AREF)電流 対 動作電圧

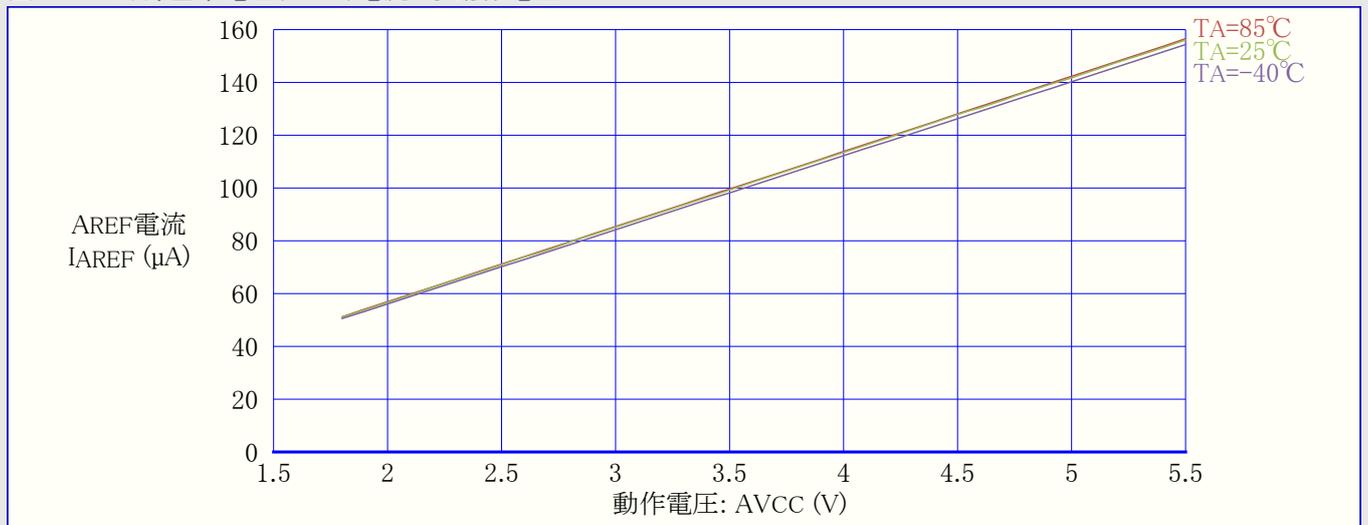


図34-46. アナログ比較器消費電流 対 動作電圧

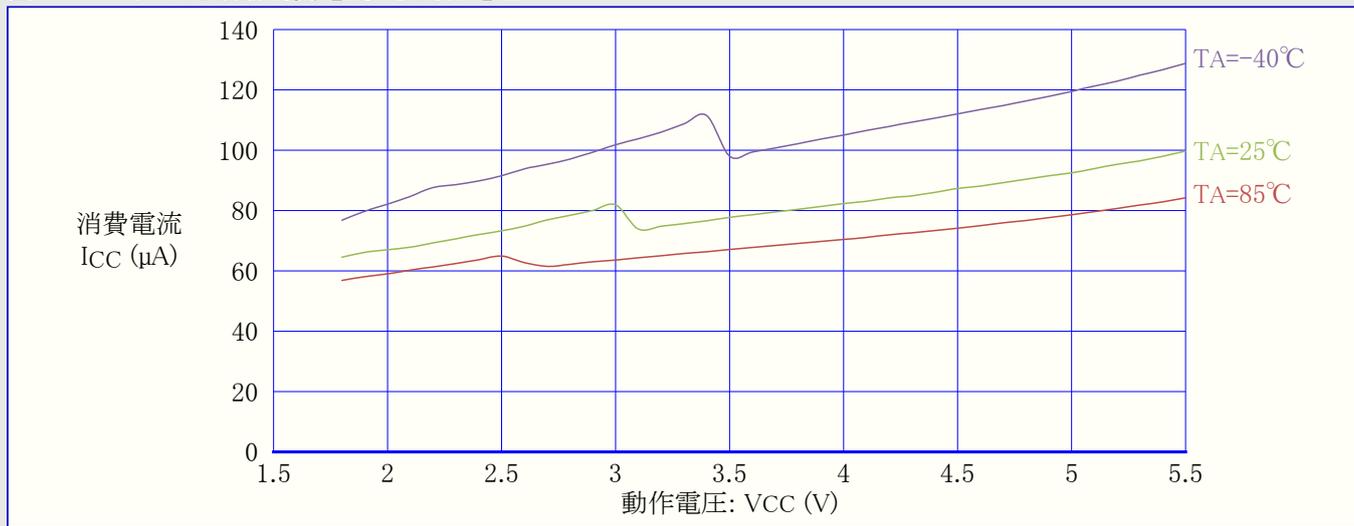


図34-47. 低電圧検出器(BOD)消費電流 対 動作電圧

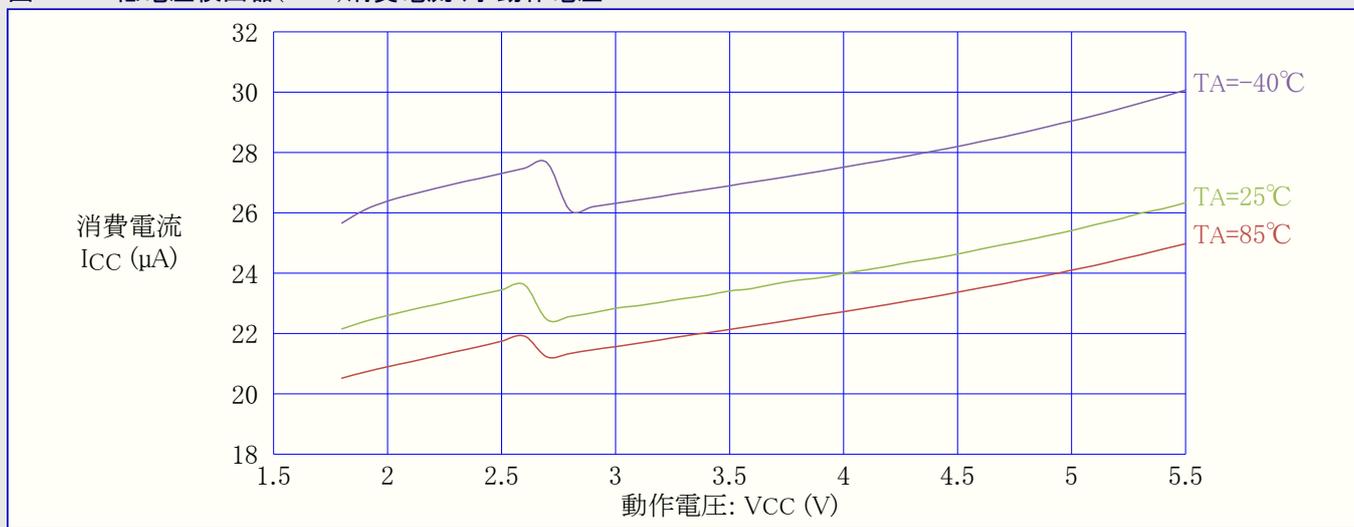
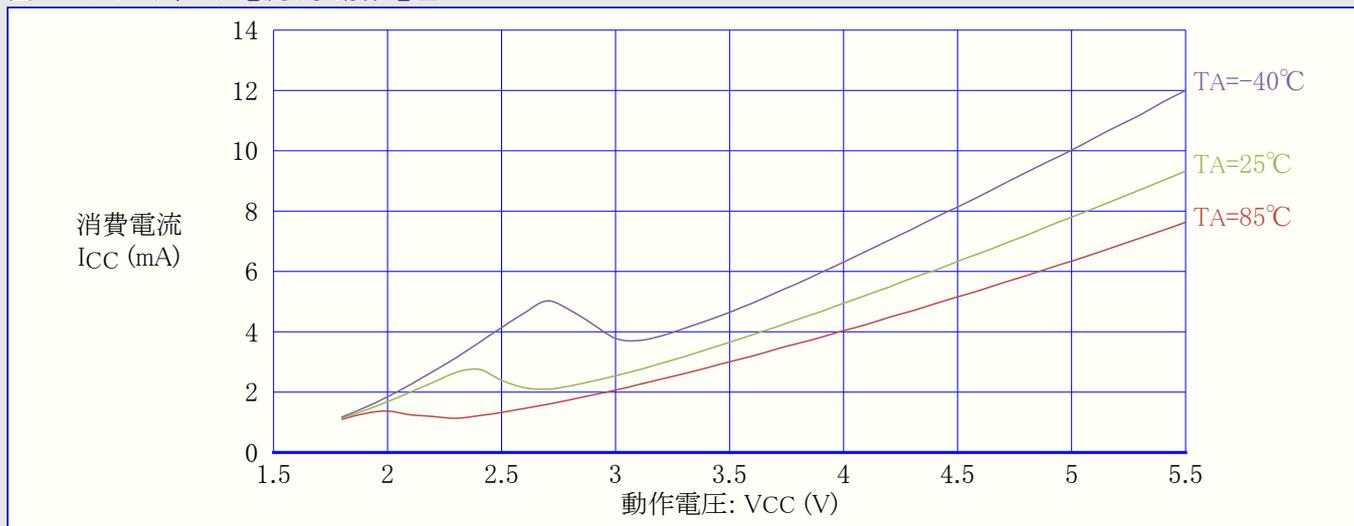


図34-48. プログラミング電流 対 動作電圧



### 34.13. リセット消費電流とリセットパルス幅

図34-49. リセット消費(供給)電流 対 周波数 (100kHz~1MHz、RESETプルアップ電流を除く)

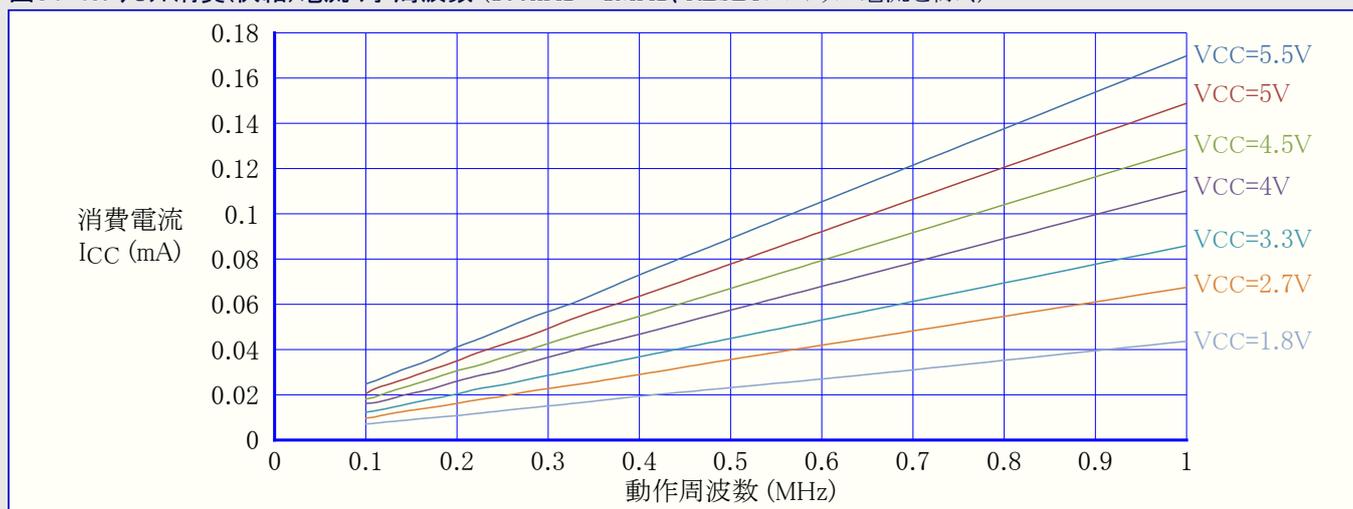


図34-50. リセット消費(供給)電流 対 周波数 (1MHz~24MHz、RESETプルアップ電流を除く)

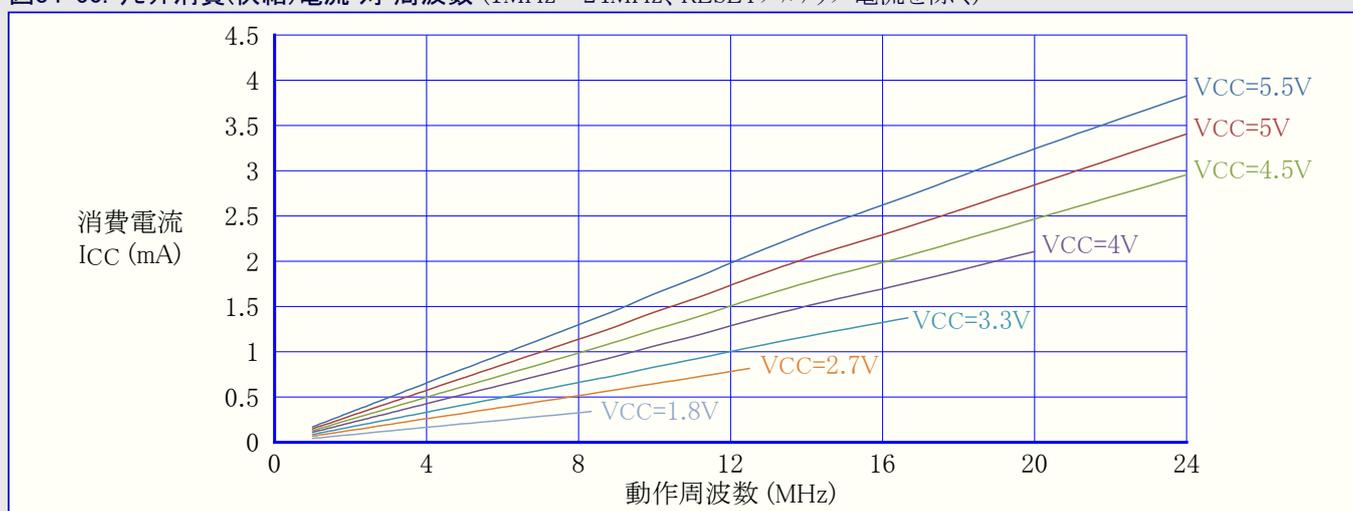
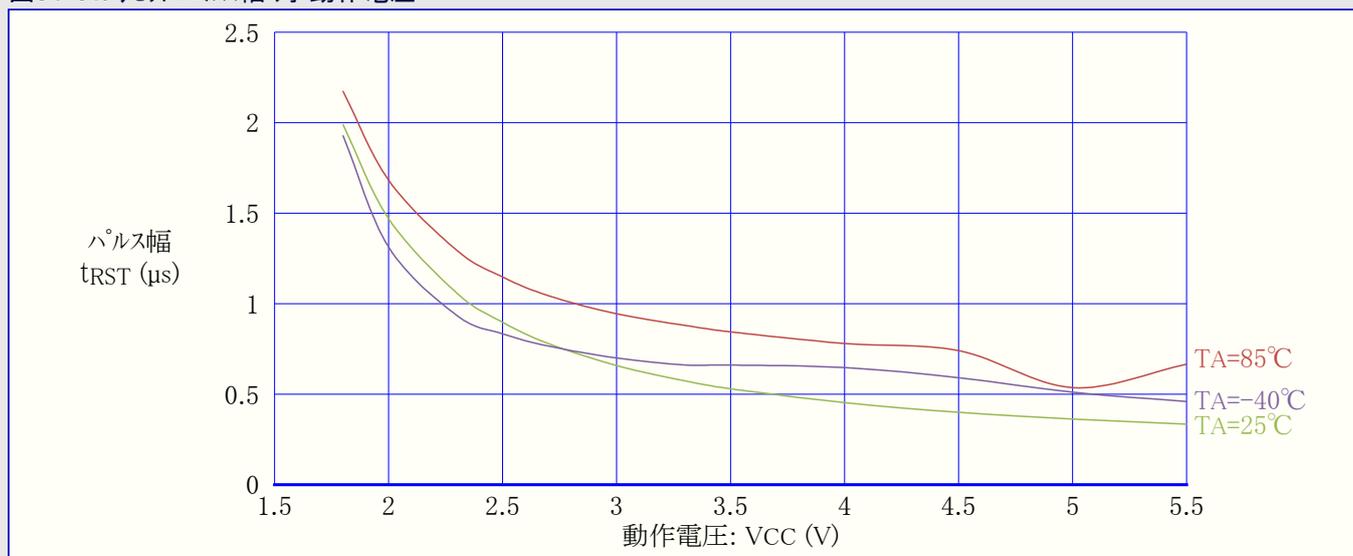


図34-51. リセットパルス幅 対 動作電圧



### 35. レジスタ要約

拡張1/Oレジスタ領域 (1/2)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$FF~\$C8	予約								
\$C7	予約								
\$C6	UDR0	TXB/RXB7~0 (USART0 データレジスタ)							
\$C5	UBRR0H	-	-	-	-	UBRR11~8 (USART0 ボーレートレジスタ 上位)			
\$C4	UBRR0L	UBRR7~0 (USART0 ボーレートレジスタ 下位)							
\$C3	予約								
\$C2	UCSR0C	UMSEL1,0		UPM1,0		USBS	UCSZ1 UDORD	UCSZ0 UCPHA	UCPOL
\$C1	UCSR0B	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8
\$C0	UCSR0A	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM
\$BF	予約								
\$BE	予約								
\$BD	TWAMR	TWAM6~0							
\$BC	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
\$BB	TWDR	TWD7~0 (2線直列インターフェース データレジスタ)							
\$BA	TWAR	TWA6~0							
\$B9	TWSR	TWS7~3				-		TWPS1,0	
\$B8	TWBR	TWBR7~0 (2線直列インターフェース0 ビット速度レジスタ)							
\$B7	予約								
\$B6	ASSR	-	EXCLK	AS2	TCN2UB	OCR2AUB	OCR2BUB	TCR2AUB	TCR2BUB
\$B5	予約								
\$B4	OCR2B	OCR2B7~0 (タイマ/カウンタ2 比較Bレジスタ)							
\$B3	OCR2A	OCR2A7~0 (タイマ/カウンタ2 比較Aレジスタ)							
\$B2	TCNT2	TCNT27~0 (タイマ/カウンタ2 計数值レジスタ)							
\$B1	TCCR2B	FOC2A	FOC2B	-	-	WGM22	CS22~0		
\$B0	TCCR2A	COM2A1,0		COM2B1,0		-	-	WGM21,0	
\$AF	予約								
\$AE	予約								
\$AD	予約								
\$AC	予約								
\$AB	予約								
\$AA	予約								
\$A9	予約								
\$A8	予約								
\$A7	予約								
\$A6	予約								
\$A5	予約								
\$A4	予約								
\$A3	予約								
\$A2	予約								
\$A1	予約								
\$A0	予約								
\$9F	予約								
\$9E	予約								
\$9D	予約								
\$9C	予約								
\$9B	予約								
\$9A	予約								
\$99	予約								
\$98	予約								
\$97	予約								
\$96	予約								
\$95	予約								
\$94	予約								
\$93	予約								
\$92	予約								
\$91	予約								
\$90	予約								

拡張I/Oレジスタ領域 (2/2)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$8F	予約								
\$8E	予約								
\$8D	予約								
\$8C	予約								
\$8B	OCR1BH			OCR1B15~8 (タイマ/カウンタ1 比較Bレジスタ 上位バイト)					
\$8A	OCR1BL			OCR1B7~0 (タイマ/カウンタ1 比較Bレジスタ 下位バイト)					
\$89	OCR1AH			OCR1A15~8 (タイマ/カウンタ1 比較Aレジスタ 上位バイト)					
\$88	OCR1AL			OCR1A7~0 (タイマ/カウンタ1 比較Aレジスタ 下位バイト)					
\$87	ICR1H			ICR115~8 (タイマ/カウンタ1 捕獲レジスタ 上位バイト)					
\$86	ICR1L			ICR17~0 (タイマ/カウンタ1 捕獲レジスタ 下位バイト)					
\$85	TCNT1H			TCNT115~8 (タイマ/カウンタ1 上位バイト)					
\$84	TCNT1L			TCNT17~0 (タイマ/カウンタ1 下位バイト)					
\$83	予約								
\$82	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-
\$81	TCCR1B	ICNC1	ICES1	-	WGM13,2			CS12~0	
\$80	TCCR1A	COM1A1,0		COM1B1,0		-	-	WGM11,0	
\$7F	DIDR1	-	-	-	-	-	-	AIN1D	AIN0D
\$7E	DIDR0	-	-	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D
\$7D	予約								
\$7C	ADMUX	REFS1,0		ADLAR	-	MUX3~0			
\$7B	ADCSRB	-	ACME	-	-	-	ADTS2~0		
\$7A	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2~0		
\$79	ADCH	ADC9~8またはADC9~2 (A/Dデータレジスタ上位バイト)							
\$78	ADCL	ADC7~0またはADC1~0 (A/Dデータレジスタ下位バイト)							
\$77	予約								
\$76	予約								
\$75	予約								
\$74	予約								
\$73	予約								
\$72	予約								
\$71	予約								
\$70	TIMSK2	-	-	-	-	-	OCIE2B	OCIE2A	TOIE2
\$6F	TIMSK1	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1
\$6E	TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0
\$6D	PCMSK2	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
\$6C	PCMSK1	-	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
\$6B	PCMSK0	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
\$6A	予約								
\$69	EICRA	-	-	-	-	ISC11,0		ISC01,0	
\$68	PCICR	-	-	-	-	-	PCIE2	PCIE1	PCIE0
\$67	予約								
\$66	OSCCAL	CAL7~0 (内蔵RC発振器 発振校正値レジスタ)							
\$65	予約								
\$64	PRR	PRTWI	PRTIM2	PRTIM0	PRUSART1	PRTIM1	PRSPI	PRUSART0	PRADC
\$63	予約								
\$62	予約								
\$61	CLKPR	CLKPCE	-	-	-	CLKPS3~0			
\$60	WDTCR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2~0		

- 注意:**
- 将来のデバイスとの共通性のため、アクセスされる場合の予約ビットは0を書かれるべきです。予約したI/Oメモリアドレスは決して書かれるべきではありません。
  - アドレス範囲\$00~\$1F内のI/OレジスタはSBIとCBI命令を使う直接ビットアクセスが可能です。これらのレジスタではSBISとSBIC命令を使うことによって単一ビット値が検査できます。
  - いくつかの状態ビットはそれらへ論理1を書くことによって解除(0)されます。他の多くのAVRと異なり、CBIとSBI命令は指定ビットだけ操作し、故にこのような状態フラグを含むレジスタで使えます。CBIとSBI命令は\$00~\$1Fのレジスタだけで動作します。
  - I/O指定命令INとOUTを使う時はI/Oアドレス\$00~\$3Fが使われなければなりません。LDとST命令を使ってデータ空間としてI/Oレジスタをアドレス指定する時はこれらのアドレスに\$20が加算されなければなりません。ATmega48/V/88/V/168/VはINとOUT命令で予約した64位置で支援できるよりも多くの周辺部(機能)の複合マイクロコントローラです。SRAM(データ空間)内の拡張I/O空間はST/STS/STDとLD/LDS/LDD命令だけが使えます。
  - EEARHレジスタとSPH.SP10, SPMCSR.RWWSB, SPMCSR.RWWSRE, MCUCR.IVSEL, MCUCR.IVCEのレジスタビットはATmega88/VとATmega168/Vでだけ利用可能です。

標準I/Oレジスタ領域

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$5F (\$3F)	SREG	I	T	H	S	V	N	Z	C
\$5E (\$3E)	SPH	-	-	-	-	-	(SP10)	SP9,8	
\$5D (\$3D)	SPL	SP7~0							
\$5C (\$3C)	予約								
\$5B (\$3B)	予約								
\$5A (\$3A)	予約								
\$59 (\$39)	予約								
\$58 (\$38)	予約								
\$57 (\$37)	SPMCSR	SPMIE	(RWWSB)	SIGRD	(RWWSRE)	BLBSET	PGWRT	PGERS	SPMEN
\$56 (\$36)	予約								
\$55 (\$35)	MCUCR	-	-	-	PUD	-	-	(IVSEL)	(IVCE)
\$54 (\$34)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF
\$53 (\$33)	SMCR	-	-	-	-	SM2~0			SE
\$52 (\$32)	予約								
\$51 (\$31)	DWDR	DWDR7~0 (デバッグWIREデータレジスタ)							
\$50 (\$30)	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1,0	
\$4F (\$2F)	予約								
\$4E (\$2E)	SPDR	SPID7~0 (SPIデータレジスタ)							
\$4D (\$2D)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X
\$4C (\$2C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1,0	
\$4B (\$2B)	GPOR2	GPOR27~0 (汎用I/Oレジスタ2)							
\$4A (\$2A)	GPOR1	GPOR17~0 (汎用I/Oレジスタ1)							
\$49 (\$29)	予約								
\$48 (\$28)	OCR0B	OCR0B7~0 (タイマ/カウンタ0 比較レジスタ)							
\$47 (\$27)	OCR0A	OCR0A7~0 (タイマ/カウンタ0 比較レジスタ)							
\$46 (\$26)	TCNT0	TCNT07~0 (タイマ/カウンタ0 計数值レジスタ)							
\$45 (\$25)	TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02~0		
\$44 (\$24)	TCCR0A	COM0A1,0		COM0B1,0		-	-	WGM01,0	
\$43 (\$23)	GTCCR	TSM	-	-	-	-	-	PSRASY	PSRSYNC
\$42 (\$22)	EEARH	-	-	-	-	-	-	-	(EEAR8)
\$41 (\$21)	EEARL	EEAR7~0 (EEPROMアドレスレジスタ下位バイト)							
\$40 (\$20)	EEDR	EEDR7~0 (EEPROMデータレジスタ)							
\$3F (\$1F)	EECR	-	-	EEP1,0		EERIE	EEMPE	EEPE	EERE
\$3E (\$1E)	GPOR0	GPOR07~0 (汎用I/Oレジスタ0)							
\$3D (\$1D)	EIMSK	-	-	-	-	-	-	INT1	INT0
\$3C (\$1C)	EIFR	-	-	-	-	-	-	INTF1	INTF0
\$3B (\$1B)	PCIFR	-	-	-	-	-	PCIF2	PCIF1	PCIF0
\$3A (\$1A)	予約								
\$39 (\$19)	予約								
\$38 (\$18)	予約								
\$37 (\$17)	TIFR2	-	-	-	-	-	OCF2B	OCF2A	TOV2
\$36 (\$16)	TIFR1	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1
\$35 (\$15)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0
\$34 (\$14)	予約								
\$33 (\$13)	予約								
\$32 (\$12)	予約								
\$31 (\$11)	予約								
\$30 (\$10)	予約								
\$2F (\$0F)	予約								
\$2E (\$0E)	予約								
\$2D (\$0D)	予約								
\$2C (\$0C)	予約								
\$2B (\$0B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
\$2A (\$0A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
\$29 (\$09)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0
\$28 (\$08)	PORTC	-	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
\$27 (\$07)	DDRC	-	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
\$26 (\$06)	PINC	-	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
\$25 (\$05)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
\$24 (\$04)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
\$23 (\$03)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0
\$22 (\$02)	予約								
\$21 (\$01)	予約								
\$20 (\$00)	予約								

注: ( )付きビットはATmega88/VとATmega168/Vでだけ利用可能です。

(訳注) 原書本位置の注意は前頁に移動しました。

## 36. 命令要約

ニーモニック	オペラント	意味	動作	フラグ	クロック
<b>算術、論理演算命令</b>					
ADD	Rd,Rr	汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr$	I,T,H,S,V,N,Z,C	1
ADC	Rd,Rr	キャリーを含めた汎用レジスタ間の加算	$Rd \leftarrow Rd + Rr + C$	I,T,H,S,V,N,Z,C	1
ADIW	Rd,K6	即値の語(ワード)長加算	$RdH:RdL \leftarrow RdH:RdL + K6$	I,T,H,S,V,N,Z,C	2
SUB	Rd,Rr	汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr$	I,T,H,S,V,N,Z,C	1
SUBI	Rd,K	汎用レジスタから即値の減算	$Rd \leftarrow Rd - K$	I,T,H,S,V,N,Z,C	1
SBIW	Rd,K6	即値の語(ワード)長減算	$RdH:RdL \leftarrow RdH:RdL - K6$	I,T,H,S,V,N,Z,C	2
SBC	Rd,Rr	キャリーを含めた汎用レジスタ間の減算	$Rd \leftarrow Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
SBCI	Rd,K	汎用レジスタからキャリーと即値の減算	$Rd \leftarrow Rd - K - C$	I,T,H,S,V,N,Z,C	1
AND	Rd,Rr	汎用レジスタ間の論理積(AND)	$Rd \leftarrow Rd \text{ AND } Rr$	I,T,H,S,0,N,Z,C	1
ANDI	Rd,K	汎用レジスタと即値の論理積(AND)	$Rd \leftarrow Rd \text{ AND } K$	I,T,H,S,0,N,Z,C	1
OR	Rd,Rr	汎用レジスタ間の論理和(OR)	$Rd \leftarrow Rd \text{ OR } Rr$	I,T,H,S,0,N,Z,C	1
ORI	Rd,K	汎用レジスタと即値の論理和(OR)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
EOR	Rd,Rr	汎用レジスタ間の排他的論理和(Ex-OR)	$Rd \leftarrow Rd \text{ EOR } Rr$	I,T,H,S,0,N,Z,C	1
COM	Rd	1の補数(論理反転)	$Rd \leftarrow \$FF - Rd$	I,T,H,S,0,N,Z,I	1
NEG	Rd	2の補数	$Rd \leftarrow \$00 - Rd$	I,T,H,S,V,N,Z,C	1
SBR	Rd,K	汎用レジスタの(複数)ビット設定(1)	$Rd \leftarrow Rd \text{ OR } K$	I,T,H,S,0,N,Z,C	1
CBR	Rd,K	汎用レジスタの(複数)ビット解除(0)	$Rd \leftarrow Rd \text{ AND } (\$FF - K)$	I,T,H,S,0,N,Z,C	1
INC	Rd	汎用レジスタの増加(+1)	$Rd \leftarrow Rd + 1$	I,T,H,S,V,N,Z,C	1
DEC	Rd	汎用レジスタの減少(-1)	$Rd \leftarrow Rd - 1$	I,T,H,S,V,N,Z,C	1
TST	Rd	汎用レジスタのゼロとマイナス検査	$Rd \leftarrow Rd \text{ AND } Rd$	I,T,H,S,0,N,Z,C	1
CLR	Rd	汎用レジスタの全0設定(=\$00)	$Rd \leftarrow Rd \text{ EOR } Rd$	I,T,H,0,0,0,I,C	1
SER	Rd	汎用レジスタの全1設定(=\$FF)	$Rd \leftarrow \$FF$	I,T,H,S,V,N,Z,C	1
MUL	Rd,Rr	符号なし間の乗算	$R1:R0 \leftarrow Rd \times Rr$ (U×U)	I,T,H,S,V,N,Z,C	2
MULS	Rd,Rr	符号付き間の乗算	$R1:R0 \leftarrow Rd \times Rr$ (S×S)	I,T,H,S,V,N,Z,C	2
MULSU	Rd,Rr	符号付きと符号なしの乗算	$R1:R0 \leftarrow Rd \times Rr$ (S×U)	I,T,H,S,V,N,Z,C	2
FMUL	Rd,Rr	符号なし間の固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (U×U)	I,T,H,S,V,N,Z,C	2
FMULS	Rd,Rr	符号付き間の固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (S×S)	I,T,H,S,V,N,Z,C	2
FMULSU	Rd,Rr	符号付きと符号なしの固定小数点乗算	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$ (S×U)	I,T,H,S,V,N,Z,C	2
<b>分岐命令</b>					
RJMP	k	相対無条件分岐	$PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	2
IJMP		Zレジスタ間接無条件分岐	$PC \leftarrow Z$	I,T,H,S,V,N,Z,C	2
JMP (注)	k	絶対無条件分岐	$PC \leftarrow k$	I,T,H,S,V,N,Z,C	3
RCALL	k	相対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow PC + k + 1$	I,T,H,S,V,N,Z,C	3
ICALL		Zレジスタ間接サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow Z$	I,T,H,S,V,N,Z,C	3
CALL (注)	k	絶対サブルーチン呼び出し	$STACK \leftarrow PC, PC \leftarrow k$	I,T,H,S,V,N,Z,C	4
RET		サブルーチンからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
RETI		割り込みからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4
CPSE	Rd,Rr	汎用レジスタ間比較、一致でスキップ	$Rd=Rr$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
CP	Rd,Rr	汎用レジスタ間の比較	$Rd - Rr$	I,T,H,S,V,N,Z,C	1
CPC	Rd,Rr	キャリーを含めた汎用レジスタ間の比較	$Rd - Rr - C$	I,T,H,S,V,N,Z,C	1
CPI	Rd,K	汎用レジスタと即値の比較	$Rd - K$	I,T,H,S,V,N,Z,C	1
SBRC	Rr,b	汎用レジスタのビットが解除(0)でスキップ	$Rr(b)=0$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBRS	Rr,b	汎用レジスタのビットが設定(1)でスキップ	$Rr(b)=1$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBIC	P,b	I/Oレジスタのビットが解除(0)でスキップ	$P(b)=0$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
SBIS	P,b	I/Oレジスタのビットが設定(1)でスキップ	$P(b)=1$ なら, $PC \leftarrow PC + 2or3$	I,T,H,S,V,N,Z,C	1/2,3
BRBS	s,k	ステータスフラグが設定(1)で分岐	$SREG(s)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRBC	s,k	ステータスフラグが解除(0)で分岐	$SREG(s)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BREQ	k	一致で分岐	$Z=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRNE	k	不一致で分岐	$Z=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCS	k	キャリーフラグが設定(1)で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRCC	k	キャリーフラグが解除(0)で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRSH	k	符号なしの $\geq$ で分岐	$C=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLO	k	符号なしの $<$ で分岐	$C=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRMI	k	-(マイナス)で分岐	$N=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRPL	k	+(プラス)で分岐	$N=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRGE	k	符号付きの $\geq$ で分岐	$(N \text{ EOR } V)=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRLT	k	符号付きの $<$ で分岐	$(N \text{ EOR } V)=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHS	k	ハーフキャリーフラグが設定(1)で分岐	$H=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRHC	k	ハーフキャリーフラグが解除(0)で分岐	$H=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTS	k	一時フラグが設定(1)で分岐	$T=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRTC	k	一時フラグが解除(0)で分岐	$T=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVS	k	2の補数溢れフラグが設定(1)で分岐	$V=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRVC	k	2の補数溢れフラグが解除(0)で分岐	$V=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRIE	k	割り込み許可で分岐	$I=1$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2
BRID	k	割り込み禁止で分岐	$I=0$ なら, $PC \leftarrow PC + K + 1$	I,T,H,S,V,N,Z,C	1/2

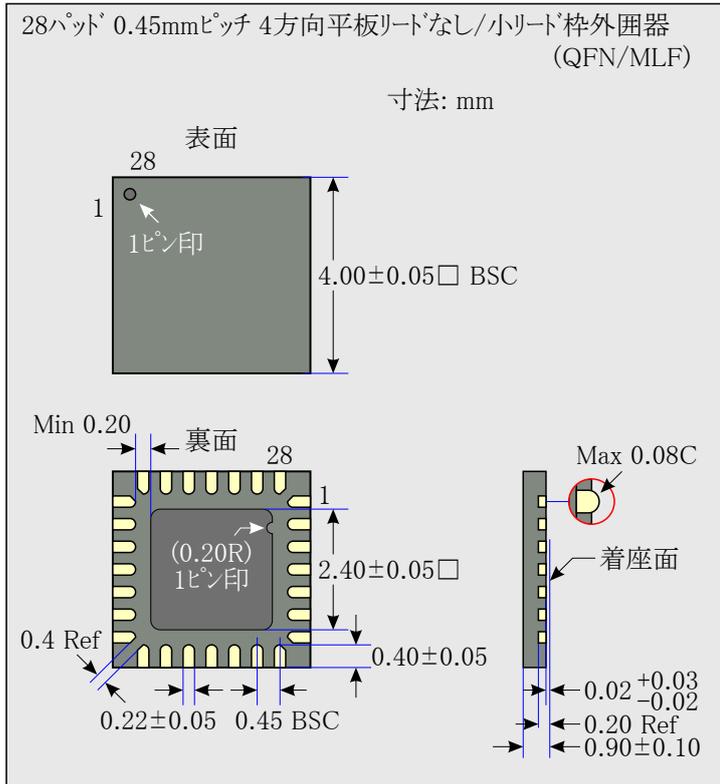
K6, K : 6, 8ビット定数 P : I/Oレジスタ Rd, Rr : 汎用レジスタ(R0~R31) X, Y, Z : X, Y, Zレジスタ  
 b : ビット(0~7) k : アドレス定数(7,12,16ビット) q : 符号なし6ビット定数(変位) s : ステータスフラグ(C,Z,N,V,X,H,T,I)

注: これらの命令はATmega168/Vでだけ利用できます。

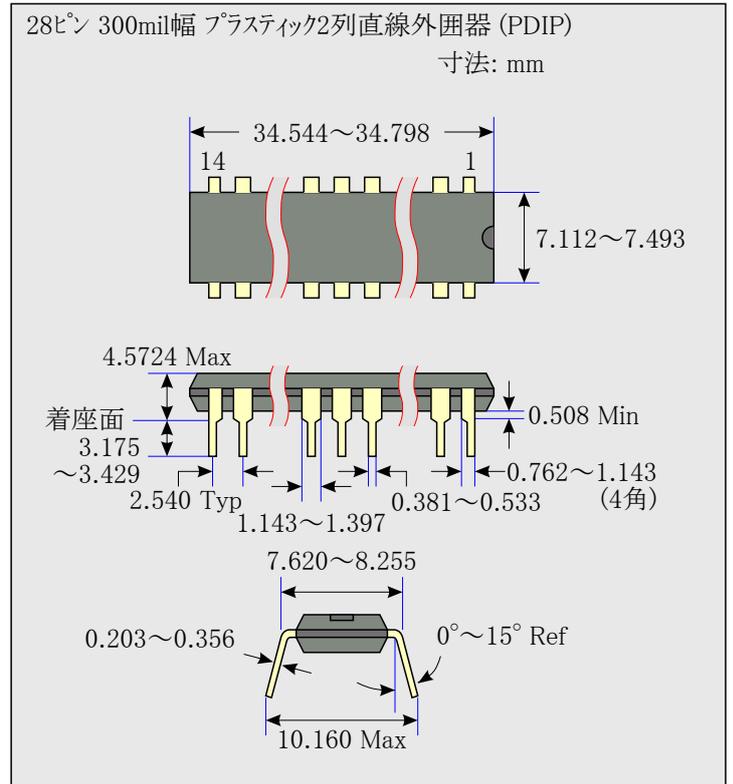
ニーモニック	オペランド	意味	動作	フラグ	クロック
<b>データ移動命令</b>					
MOV	Rd,Rr	汎用レジスタ間の複写	$Rd \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
MOVW	Rd,Rr	汎用レジスタ対間の複写	$Rd+1:Rd \leftarrow Rr+1:Rr$	I,T,H,S,V,N,Z,C	1
LDI	Rd,K	即値の取得	$Rd \leftarrow K$	I,T,H,S,V,N,Z,C	1
LD	Rd,X	Xレジスタ間接での取得	$Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,X+	事後増加付きXレジスタ間接での取得	$Rd \leftarrow (X), X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-X	事前減少付きXレジスタ間接での取得	$X \leftarrow X - 1, Rd \leftarrow (X)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y	Yレジスタ間接での取得	$Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Y+	事後増加付きYレジスタ間接での取得	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Y	事前減少付きYレジスタ間接での取得	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Y+q	変位付きYレジスタ間接での取得	$Rd \leftarrow (Y + q)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z	Zレジスタ間接での取得	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LD	Rd,Z+	事後増加付きZレジスタ間接での取得	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
LD	Rd,-Z	事前減少付きZレジスタ間接での取得	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	2
LDD	Rd,Z+q	変位付きZレジスタ間接での取得	$Rd \leftarrow (Z + q)$	I,T,H,S,V,N,Z,C	2
LDS	Rd,k	データ空間(SRAM)から直接取得	$Rd \leftarrow (k)$	I,T,H,S,V,N,Z,C	2
ST	X,Rr	Xレジスタ間接での設定	$(X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	X+,Rr	事後増加付きXレジスタ間接での設定	$(X) \leftarrow Rr, X \leftarrow X + 1$	I,T,H,S,V,N,Z,C	2
ST	-X,Rr	事前減少付きXレジスタ間接での設定	$X \leftarrow X - 1, (X) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y,Rr	Yレジスタ間接での設定	$(Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Y+,Rr	事後増加付きYレジスタ間接での設定	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	I,T,H,S,V,N,Z,C	2
ST	-Y,Rr	事前減少付きYレジスタ間接での設定	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Y+q,Rr	変位付きYレジスタ間接での設定	$(Y + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z,Rr	Zレジスタ間接での設定	$(Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
ST	Z+,Rr	事後増加付きZレジスタ間接での設定	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	2
ST	-Z,Rr	事前減少付きZレジスタ間接での設定	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STD	Z+q,Rr	変位付きZレジスタ間接での設定	$(Z + q) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
STS	k,Rr	データ空間(SRAM)へ直接設定	$(k) \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
LPM		プログラム領域からZレジスタ間接での取得	$R0 \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z	同上 (任意のレジスタへ)	$Rd \leftarrow (Z)$	I,T,H,S,V,N,Z,C	3
LPM	Rd,Z+	同上 (事後増加付き)	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	I,T,H,S,V,N,Z,C	3
SPM		プログラム領域へZレジスタ間接での設定	$(Z) \leftarrow R1:R0$	I,T,H,S,V,N,Z,C	-
IN	Rd,P	I/Oレジスタからの入力	$Rd \leftarrow P$	I,T,H,S,V,N,Z,C	1
OUT	P,Rr	I/Oレジスタへの出力	$P \leftarrow Rr$	I,T,H,S,V,N,Z,C	1
PUSH	Rr	汎用レジスタをスタックへ保存	$STACK \leftarrow Rr$	I,T,H,S,V,N,Z,C	2
POP	Rd	スタックから汎用レジスタへ復帰	$Rd \leftarrow STACK$	I,T,H,S,V,N,Z,C	2
<b>ビット関係命令</b>					
SBI	P,b	I/Oレジスタのビット設定(1)	$I/O(P,b) \leftarrow 1$	I,T,H,S,V,N,Z,C	2
CBI	P,b	I/Oレジスタのビット解除(0)	$I/O(P,b) \leftarrow 0$	I,T,H,S,V,N,Z,C	2
LSL	Rd	論理的左ビット移動	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	I,T,H,S,V,N,Z,C	1
LSR	Rd	論理的右ビット移動	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	I,T,H,S,V,0,Z,C	1
ROL	Rd	キャリーを含めた左回転	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	I,T,H,S,V,N,Z,C	1
ROR	Rd	キャリーを含めた右回転	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	I,T,H,S,V,N,Z,C	1
ASR	Rd	算術的右ビット移動	$Rd(n) \leftarrow Rd(n+1), n=0\sim6$	I,T,H,S,V,N,Z,C	1
SWAP	Rd	ニブル(4ビット)上位/下位交換	$Rd(7\sim4) \leftrightarrow Rd(3\sim0)$	I,T,H,S,V,N,Z,C	1
BSET	s	ステータスレジスタのビット設定(1)	$SREG(s) \leftarrow 1$	I,T,H,S,V,1,I,I	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLC		キャリーフラグを解除(0)	$C \leftarrow 0$	I,T,H,S,V,N,Z,0	1
SEN		負フラグを設定(1)	$N \leftarrow 1$	I,T,H,S,V,1,Z,C	1
CLN		負フラグを解除(0)	$N \leftarrow 0$	I,T,H,S,V,0,Z,C	1
SEZ		ゼロフラグを設定(1)	$Z \leftarrow 1$	I,T,H,S,V,N,1,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,0,C	1
SEI		全割り込み許可	$I \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLI		全割り込み禁止	$I \leftarrow 0$	0,T,H,S,V,N,Z,C	1
SES		符号フラグを設定(1)	$S \leftarrow 1$	I,T,H,1,V,N,Z,C	1
CLS		符号フラグを解除(0)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	$V \leftarrow 1$	I,T,H,S,V,1,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	$V \leftarrow 0$	I,T,H,S,0,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,1,H,S,V,N,Z,C	1
CLT		一時フラグを解除(0)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1
<b>MCU制御命令</b>					
NOP		無操作		I,T,H,S,V,N,Z,C	1
SLEEP		休止形態開始	休止形態参照	I,T,H,S,V,N,Z,C	1
WDR		ウォッチドッグタイマリセット	ウォッチドッグタイマ参照	I,T,H,S,V,N,Z,C	1
BREAK		一時停止	内蔵デバッグ機能専用	I,T,H,S,V,N,Z,C	N/A

## 37. 外圍器情報

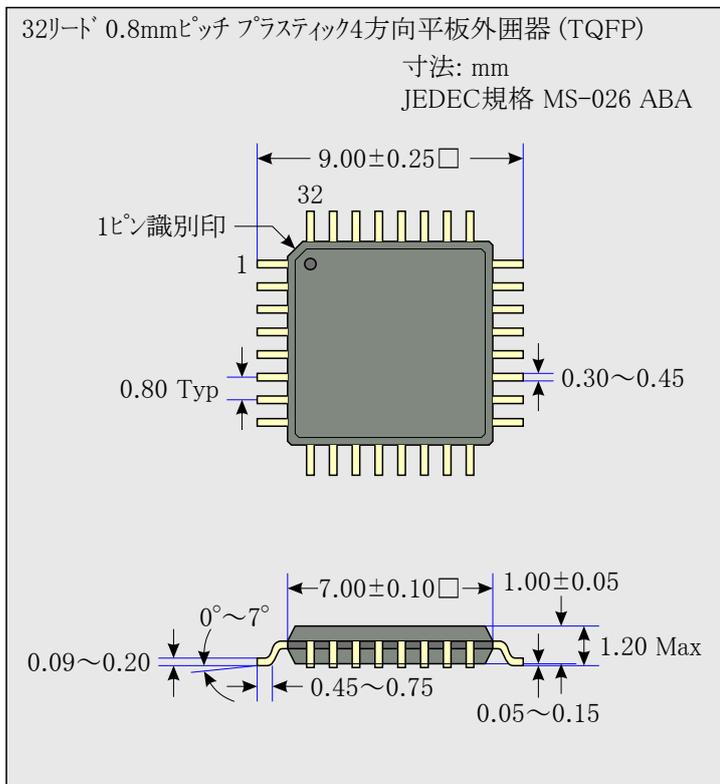
### 37.1. 28M1



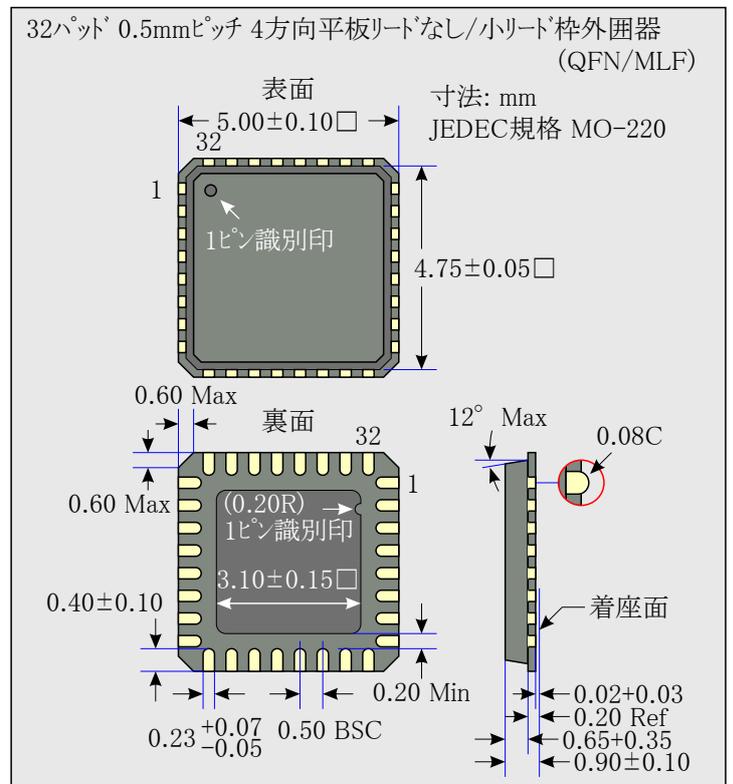
### 37.2. 28P3



### 37.3. 32A



### 37.4. 32M1-A



## 38. 障害情報

この章の改訂番号はATmega48/V/88/V/168/Vデバイスの改訂版を参照してください。

ATmega88/V 改訂B,C : 試供されていません。

• 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	48/V-A~D,88/V-A/D,168/V-A~C
• デバイスがリセットで停止する可能性	48/V-A,88/V-A,168/V-A/B
• 単独消去操作後の不正値読み込み	48/V-A,168/V-A
• 周波数900kHz未満でのEEPROM読み込みが動作しない可能性	48/V-C
• ウォッチドッグ タイマ割り込みが禁止	48/V-A
• 予想よりも多いクリスタル発振器での起動時間	48/V-A
• 外部クロックでのパワーダウン動作で高消費電力	48/V-A
• パワーダウン動作で非同期発振器が停止しない	48/V-A
• 低電圧動作でEEPROM書き込みが動作しない	88/V-A

### 1. 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消失の可能性 (48/V-A~D,88/V-A/D,168/V-A~C)

非同期タイマ/カウンタレジスタ(TCNTx)が\$00の時に同期タイマ/カウンタクロックのタイマ/カウンタレジスタが書かれる場合に割り込みが失われるでしょう。

#### 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、常に非同期タイマ/カウンタレジスタが\$FFまたは\$00のどちらの値でもないことを調べてください。

### 2. デバイスがリセットで停止する可能性 (48/V-A,88/V-A,168/V-A/B)

いくつかのデバイスは内部リセット順次回路が特別な状態の時にリセット信号が印加されると、リセット状態で動かなくなるかもしれません。内部リセット順次回路はリセット後にデバイスが起動する直前の約10nsと、システムクロック前置分周器が切り替わる時の10ns間がこの状態です。この問題は実装書き込み(ISP)中に度々よく見られます。この出来事は走行動作でも理論的な可能性があります。次の3つの場合がデバイスをリセット状態で動かなくするきっかけになり得ます。

- 最初のリセットによってデバイスがリセット状態外になる前の10ns間に2番目のリセットが起きる、2つの連続するリセットの印加。
- ソフトウェアによってシステムクロック前置分周器値が更新される最中の10ns間でのリセットの印加。
- 内部リセット信号を生成するSPIプログラミングの抜け出しが、この状態のきっかけとなり得ます。

最初の2つの場合は通常の動作中に起き得、一方最後の場合はデバイスのプログラミング中にだけ起きます。

#### 対策/対処

最初の場合は1つのリセット元だけが活性(有効)であることを保証することにより、走行動作中に防げます。押し釦の外部リセットが使われる場合、リセット起動時間は起動時間中にリセット信号が完全に跳ねなくなるようなもの(時間)を選ぶべきです。

2つ目の場合はシステムクロック前置分周器を使わないことによって防げます。

3つ目の場合は実装書き込み(ISP)中だけに起きます。最高周波数で内蔵RC発振器を使う時に最もよく見られます。

デバイスがリセット状態で動かなくなったら、デバイスをリセット状態外にするために、電源をOFFにし、その後再びONにしてください。

### 3. 単独消去操作後の不正値読み込み (48/V-A,168/V-A)

2.7V未満の電圧印加で、消去操作のみで消去されたEEPROM位置はプログラム(\$00)として読むかもしれません。

#### 対策/対処

単独消去後にEEPROM位置を読む必要がある場合、その位置を消去するために値として\$FFの非分離書き込みを使ってください。どんな場合も単独書き込み操作は予定通りに使えます。従って消去された位置がプログラミング以前に読まれない限り、特別な考慮の必要はありません。

### 4. 周波数900kHz未満でのEEPROM読み込みが動作しない可能性 (48/V-C)

低内部クロック周波数でのEEPROMからの読み込みは不正なデータ読み込みに終わるかもしれません。

#### 対策/対処

900kHz未満のクロック周波数でのEEPROM読み込みを避けてください。

### 5. ウォッチドッグ タイマ割り込みが禁止 (48/V-A)

ウォッチドッグ タイマ割り込み要求フラグ(WDIF)が次の計時完了発生に先立って解除(0)されなければウォッチドッグが禁止され、割り込み要求フラグは自動的に解除(0)されます。これは割り込みのみの動作種別にだけ当てはまります。ウォッチドッグ計時完了に続く割り込みでウォッチドッグがデバイスリセットに設定されれば、デバイスは正しく動作します。

#### 対策/対処

次のウォッチドッグ計時完了が起こる前に先行する計時完了発生での処理が常に十分な時間であることに注意してください。これは十分に長い計時完了周期を選ぶことによって行われます。

#### 6. 予想よりも多いクリスタル発振器での起動時間 (48/V-A)

外部クリスタル走行時、起動時間のクロック計数部は起動期間に関して予想よりも約2倍多くなります。これはリセットによる起動時にだけ適用します。パワーダウン動作からの起動は影響を及ぼされません。殆どの設定についてクロック計数部は起動時間全体の小さな断片にすぎず、従ってこの問題は無視できます。例外は例えば時計用32kHzクリスタルのような非常に低い周波数のクリスタルを使う時です。

##### 対策/対処

既知の対策はありません。

#### 7. 外部クロックでのパワーダウン動作で高消費電力 (48/V-A)

外部クロック活性でのパワーダウン動作の消費電力は内部RCまたは外部発振器使用時より約10倍多くなります。

##### 対策/対処

デバイスのパワーダウン時、外部クロックを停止してください。

#### 8. パワーダウン動作で非同期発振器が停止しない (48/V-A)

パワーダウン移行時、非同期発振器は停止しません。これは予想よりも高い電力消費を引き起こします。

##### 対策/対処

パワーダウン動作へ移行する前に、操作によって非同期タイマを禁止してください。

#### 9. 低電圧動作でEEPROM書き込みが動作しない (88/V-A)

低電圧でEEPROM書き込みが動作しません。

##### 対策/対処

4.5V未満の電圧でEEPROMを書かないでください。(これは改訂Bで修正されます。)

## 39. データシート改訂履歴

この章内の参照頁番号はこの文書が参照されていることに注意してください。この章内の改訂番号は文書の改訂番号を参照してください。

### 39.1. 改訂2545B – 2004年1月

1. 「特徴」で動作速度と代表消費電力を更新、I/Oと外圍器にPDIPを追加
2. 「スタックポインタ」を(推奨スタックポインタ値としてRAMEND)で更新
3. 「電力削減レジスタ」節、PRRビット使用に関する注意をタイマ/カウンタ、USART、A/D変換器、アナログ比較器に追加
4. 「ウォッチドッグ タイマ」を更新
5. 表20-4.と表20-5.を更新
6. 「TC2 – タイマ/カウンタ2 (PWM,非同期動作付き)」章で「特徴」に比較一致割り込み(OCF2B)を追加
7. 表14-1.、表28-1.、表28-6.、表32-8.を更新、表32-1.に注2を追加、表17-1.で誤植を修正
8. 「代表特性」全体を更新
9. 「障害情報」に3項目追加
10. 次のビットを改名
  - SPMENをSELFPRGENへ
  - PSR2をPSRASYSへ
  - PSR10をPSRSYNCへ
  - ウォッチドッグ リセットをウォッチドッグ システム リセットへ
11. 旧IAR文法を含むCコード例を更新
12. 「SPMCSR – SPM命令制御/状態レジスタ」でBLBSET記述を更新

### 39.2. 改訂2545C – 2004年4月

1. 最大動作速度を変更：12MHzを10MHzへと24MHzを20MHzへ
2. 「速度勾配」を更新
3. 「注文情報」を更新
4. 「障害情報」でATmega88/Vを更新

### 39.3. 改訂2545D – 2004年7月

1. コード例に関してWDTCSRを使った命令を更新
2. 表13-5.、表31-7.と表31-9.、表32-17.、表33-7.を更新
3. 「システム クロック前置分周器」を更新
4. 「TIMSK2 – TC2割り込み許可レジスタ」と「TIFR2 – TC2割り込み要求フラグ レジスタ」を「8ビット タイマ/カウンタ2用レジスタ」に移動
5. 「電氣的な相互接続」でタイミング特性へのリンクを更新
6. 「ビット速度発生器」で式を更新
7. 「ページ容量」を追加
8. 「直列プログラミング手順」を更新
9. ATmega168/Vに対する「注文情報」を更新
10. ATmega88/V/168/Vに対する「障害情報」を更新

### 39.4. 改訂2545E – 2005年2月

1. MLF外圍器を“4方向平板リードなし/極小リード”枠 QFN/MLF”に代替変更
2. 「EECR – EEPROM制御レジスタ」を更新
3. 「校正付き内蔵RC発振器」を更新
4. 「外部クロック信号」を更新
5. 表32-17.、表33-5.、表33-6.、表33-8.を更新
6. 「ピン変化割り込みタイミング」を追加
7. 図19-1.を更新
8. 「SPMCSR – SPM命令制御/状態レジスタ」を更新
9. 「並列プログラミング動作への移行」を更新
10. 「DC特性」を更新
11. 「注文情報」を更新
12. ATmega88/V/168/Vに対する「障害情報」を更新

### 39.5. 改訂2545F – 2005年6月

1. 「資料」を追加
2. 「校正付き内蔵RC発振器」を更新
3. 「直列プログラミング命令一式」を更新
4. 「DC特性」で表の注記を更新
5. 「障害情報」を更新

### 39.6. 改訂2545E – 2005年2月

1. レジスタにアドレスを追加
2. 「校正付き内蔵RC発振器」を更新
3. 表13-15、表14-1、表15-1、表18-3を更新
4. 「A/D変換雑音低減動作」を更新
5. 表14-2に対する注を更新
6. 「ビット2 – PRSPI：直列周辺インターフェース電力削減」を更新
7. 「TCCR0B – タイマ/カウンタ0制御レジスタB」を更新
8. 「高速PWM動作」を更新 (訳注: 本書では原書と異なり本来の形で修正)
9. 「タイマ/カウンタ2 非同期動作」を更新
10. 「SPI – 直列周辺インターフェース」を更新
11. 「UCSRnA – USARTn制御/状態レジスタA」を更新
12. 「ビット速度発生器」を更新
13. 「ビット6 – ACBG：基準電圧選択」を更新
14. 「ADC – A/D変換器」で特徴を更新
15. 「前置分周と変換タイミング」を更新
16. 「デバッグWIREの制限」を更新
17. 表33-4を追加
18. 表20-7、表34-44を更新 (訳注: 表20-7は矛盾回避のため、本書では無視)
19. ATmega48/V改訂Aについての「障害情報」を更新
20. ATmega48/V改訂CとDについての「障害情報」を追加

### 39.7. 改訂2545H – 2006年10月

1. 誤植修正
2. 「特徴」を更新
3. 「校正付き内蔵RC発振器」を更新
4. 「SCRST – システム制御とリセット」を更新
5. 「低電圧検出(BOD)リセット」を更新
6. 「高速PWM動作」を更新
7. 「TCCR1C – タイマ/カウンタ1制御レジスタC」を更新
8. 「SPI – 直列周辺インターフェース」を更新
9. 表19-3、表19-6、表19-8、表20-3、表20-4、表20-5、表22-3、表22-6、表22-8、表32-6を更新
10. 表30-1、表31-6、表32-18に注を追加
11. 「SPM命令によるブートローダ施錠ビットと一般施錠ビットの設定」を更新
12. 「識票ハイ」を更新
13. 「電気的特性」を更新
14. 「障害情報」を更新

### 39.8. 改訂2545I – 2006年11月

1. 「特徴」を更新
2. 「TWI – 2線直列インターフェース」で「特徴」を更新
3. 表33-6で誤植を修正

### 39.9. 改訂2545J – 2006年12月

1. 「特徴」を更新
2. 「ピン配置」図を更新

3. 「注文情報」を更新
4. 「外圍器情報」を更新

### 39.10. 改訂2545K – 2007年4月

1. 「ウォッチバック タイム」への参照を「割り込み」への参照に修正
2. ATmega48/Vに対して「障害情報」を更新
3. 「ADC – A/D変換器」の記述を変更

### 39.11. 改訂2545L – 2007年8月

1. 「特徴」を更新
2. 「MCUCR – MCU制御レジスタ」でコード例を更新
3. 「システムとリセットの特性」を更新
4. 表13-3、表13-5、表13-11、表13-13で注を更新

### 39.12. 改訂2545M – 2007年9月

1. 「データ保持力」を追加
2. 「A/D変換特性」を更新
3. データシートを通して「暫定」を削除

### 39.13. 改訂2545N – 2009年1月

1. 「特徴」を更新、「新規設計に推奨されません。」を追加
2. 「資料」、「データ保持力」、「コード例について」章を「諸注意」共通章下に合併(訳注:後に分離)
3. 図13-3を更新
4. 「システム クロック前置分周器」を更新
5. 「ポートBの交換機能」を更新
6. 「電源ON傾斜率」項を追加
7. 「ピン閾値とヒステリシス」を更新

### 39.14. 改訂2545O – 2009年2月

1. 表33-6で下降時電源ON最小閾値電圧を0.05Vに変更
2. 「システムとリセットの特性」から「電源ON傾斜率」項を削除

### 39.15. 改訂2545P – 2009年2月

1. 表33-6から電源OFF傾斜率を削除

### 39.16. 改訂2545Q – 2009年6月

1. 表題「諸注意」削除。本章の副項目は今や「資料」、「データ保持力」、「コード例について」章に分離
2. 「注文情報」を更新

### 39.17. 改訂2545R – 2009年7月

1. 「障害情報」を更新
2. Atmelの新住所で最終頁を更新

### 39.18. 改訂2545S – 2010年7月

1. 「2線直列インターフェース特性」の表で注の6と7を削除
2. Atmel規格に従って文書更新

### 39.19. 改訂2545T – 2011年5月

1. AIとMIの削除、AURとMUR(テープとリール)の追加によって「注文情報」を更新
2. 参照リンクの追加と修正
3. 新しいAtmel規格に従って文書更新
4. QTouchライブラリ支援の特徴を追加

### 39.20. 改訂2545U – 2015年11月

#### 1. 「障害情報」章で以下を更新

- ATmega48/V改訂E～Kの障害情報を追加
- ATmega88/V改訂E～Kの障害情報を追加
- ATmega168/V改訂D～Kの障害情報を追加

### 39.21. 改訂2545V – 2016年6月

#### 1. 編集上の更新

### 39.22. 改訂2545W – 2016年11月

#### 1. 「障害情報」章更新

- ATmega48/V：改訂E～Kを削除
  - 改訂E～Jのダイは試供されませんでした。
  - 改訂Kのダイは製品供給されませんでした。
- ATmega88/V：改訂E～Kを削除
  - 改訂E～Jのダイは試供されませんでした。
  - 改訂Kのダイは製品供給されませんでした。
- ATmega168/V：改訂D～Kを削除
  - 改訂D～Jのダイは試供されませんでした。
  - 改訂Kのダイは製品供給されませんでした。

Atmel®、Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®、AVR®とその他は米国及び他の国に於けるAtmel Corporationの登録商標または商標です。他の用語と製品名は一般的に他の商標です。

**お断り:** 本資料内の情報はAtmel製品と関連して提供されています。本資料またはAtmel製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。Atmelのウェブサイトに表示する販売の条件とAtmelの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、Atmelはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえAtmelがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益と損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してAtmelに責任がないでしょう。Atmelは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。Atmelはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、Atmel製品は車載応用に対して適当ではなく、使用されるべきではありません。Atmel製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

**安全重視、軍用、車載応用のお断り:** Atmel製品はAtmelが提供する特別に書かれた承諾を除き、そのような製品の機能不全が著しく人に危害を加えたり死に至らしめることがかなり予期されるどんな応用(“安全重視応用”)に対しても設計されず、またそれらとの接続にも使用されません。安全重視応用は限定なしで、生命維持装置とシステム、核施設と武器システムの操作の装置やシステムを含みます。Atmelによって軍用等級として特に明確に示される以外、Atmel製品は軍用や航空宇宙の応用や環境のために設計も意図もされていません。Atmelによって車載等級として特に明確に示される以外、Atmel製品は車載応用での使用のために設計も意図もされていません。

#### © HERO 2022.

本データシートはAtmelのATmega48/V/88/V/168/V英語版データシート(Rev.2545W-11/2016)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

汎用入出力ポートの出力データレジスタとピン入力は、対応関係からの理解の容易さから出力レジスタと入力レジスタで統一表現されています。一部の用語がより適切と思われる名称に変更されています。必要と思われる部分には( )内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。

原書に対して若干構成が異なるため、一部の節/項番号が異なります。