



8ビット AVR マイクロコントローラ

ATmega16M1, ATmega32M1, ATmega64M1

完全 データシート

序説

Atmel® ATmega16M1/32M1/64M1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、ATmega16M1/32M1/64M1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**131命令**(多くは1周期で実行)
 - 32個の1バイト長**汎用レジスタ**
 - 完全なスタティック動作
 - 1MHz当たり、1MIPSに達する高速動作
 - 2周期実行の乗算命令
- データ メモリと不揮発性プログラム メモリ
 - 実装自己書き換え可能な16/32/64Kバイト(8/16/32K語)フラッシュ メモリ内蔵
 - 512/1K/2Kバイトの**EEPROM**
 - 1K/2K/4K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**
- **デバッグWIRE内蔵デバッグ機能**
- 6つのメッセージ項目を持つ**CAN 2.0Aと2.0B 制御器** - ISO16845保証
- **LIN2.1と1.3の制御器**、または**8ビットUART**
- 1つの**12ビット高速電力段制御器(PSC)**
 - 柔軟な沈黙時間付き非重複反転PWM出力ピン
 - 可変PWMデューティ比と可変PWM周波数
 - 全PWMレジスタの同期更新
 - 緊急事態用の自動停止機能
- 内蔵周辺機能
 - 独立した前置分周器、比較機能付き、1つの汎用**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つの汎用**16ビット タイマ/カウンタ**
 - 主装置/従装置動作**SPI直列インターフェース**
 - **10ビット A/D変換器**
 - 11個までのシングル エンド入力チャネルと、3つの完全な差動チャネル対
 - **設定可能な利得**(差動チャネルでの×5, ×10, ×20, ×40)
 - 内部基準電圧
 - 電源電圧直接測定
 - 可変電圧基準(アナログ比較器、A/D変換器)用**10ビット D/A変換器**
 - 可変検出閾値を持つ4つの**アナログ比較器**
 - 100µA ±2%電流源(LIN節点(ノード)認証)
 - ピン変化での割り込みと起動復帰
 - 独立したチップ上の発振器を持つ設定可能な**ウォッチドッグ タイマ**
 - チップ上の**温度感知器**
- 特殊マイクロ コントローラ機能
 - **アイドル、A/D変換雑音低減、パワーダウン、スタンバイ**の4つの低消費動作
 - 電源ONリセット回路と設定可能な**低電圧検出器(BOD)**
 - **SPIポート経由での実装書き込み**
 - **CAN動作(16MHz)用高精度クリスタル用発振器**
 - **校正可能な内蔵RC発振器 (8MHz)**
 - 高速PWM(32MHz, 64MHz)とCPU(16MHz)用の**チップ上PLL**
- 動作電圧: 2.7~5.5V
- 拡張された温度範囲: -40~85°C
- 動作速度
 - 0~8MHz/2.7~4.5V
 - 0~16MHz/4.5~5.5V

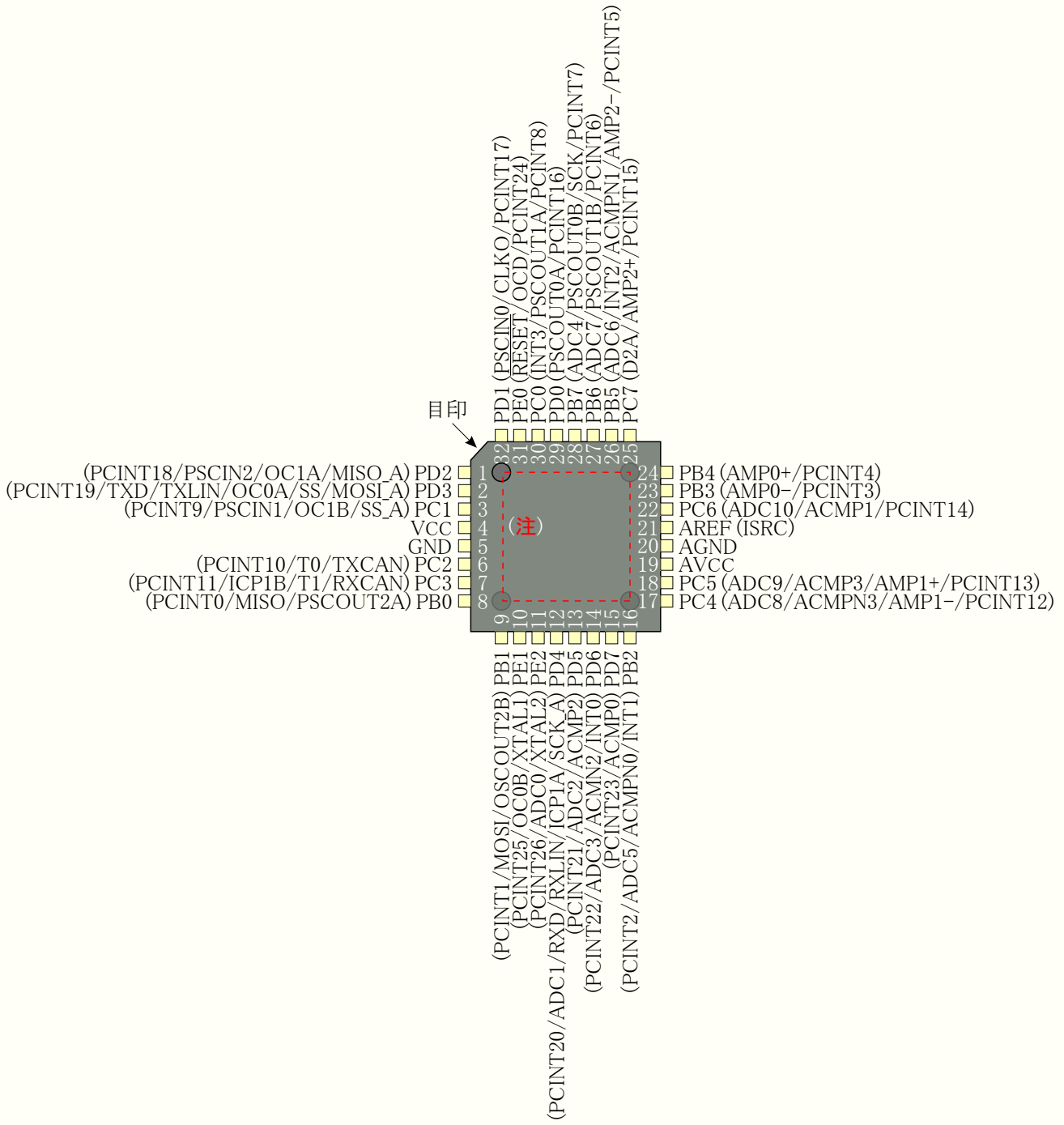
(訳注) 原書が他のデバイスから安易に複製して作成されたため、実際のデバイスと異なる記述が多数存在します。該当箇所が多すぎるため、本書ではその殆どで特に断りなく修正しています。

序説	1	11.9. リセット関係用レジスタ	39
特徴	2	12. INT – 割り込み	41
1. ピン配置	5	12.1. ATmega16M1/32M1/64M1の割り込みベクタ	41
1.1. ピン概要	6	12.2. ベクタ移動用レジスタ	44
2. 注文情報	7	13. EXTINT – 外部割り込み	45
3. 概要	8	13.1. ピン変化割り込みタイミング	45
3.1. 構成図	8	13.2. 外部割り込み用レジスタ	46
3.2. ピン説明	8	14. 入出力ポート	51
4. 資料	9	14.1. 概要	51
5. コード例について	9	14.2. 標準デジタル入出力としてのポート	51
6. データ保持力	9	14.3. 交換ポート機能	54
7. AVR CPU コア	10	14.4. I/Oポート用レジスタ	63
7.1. 概要	10	15. TC0 – 8ビット タイマ/カウンタ0 (PWM)	67
7.2. ALU (Arithmetic Logic Unit)	10	15.1. 特徴	67
7.3. ステータスレジスタ	11	15.2. 概要	67
7.4. 汎用レジスタファイル	12	15.3. タイマ/カウンタのクロック	68
7.5. スタックポインタ	12	15.4. 計数器部	68
7.6. 16ビットレジスタのアクセス	13	15.5. 比較出力部	69
7.7. 命令実行タイミング	14	15.6. 比較一致出力部	70
7.8. リセットと割り込みの扱い	14	15.7. 動作種別	71
8. AVRのメモリ	16	15.8. タイマ/カウンタのタイミング	74
8.1. 概要	16	15.9. 8ビット タイマ/カウンタ0用レジスタ	75
8.2. 実装書き換え可能なプログラム用フラッシュメモリ	16	16. TC1 – 16ビット タイマ/カウンタ1 (PWM)	80
8.3. データ用SRAMメモリ	16	16.1. 概要	80
8.4. データ用EEPROMメモリ	17	16.2. 特徴	80
8.5. I/Oメモリ (レジスタ)	17	16.3. 構成図	80
8.6. メモリ関係レジスタ	19	16.4. 定義	81
9. システムクロックとクロック選択	23	16.5. 関係レジスタ	81
9.1. クロックシステムとその配給	23	16.6. 16ビット タイマ/カウンタレジスタのアクセス	82
9.2. クロック元	24	16.7. タイマ/カウンタのクロック	84
9.3. 既定のクロック元	24	16.8. 計数器部	84
9.4. クリスタル用低電力発振器	24	16.9. 捕獲入力部	85
9.5. 校正付き内蔵RC発振器	25	16.10. 比較出力部	86
9.6. PLL	25	16.11. 比較一致出力部	87
9.7. 128kHz内部発振器	26	16.12. 動作種別	88
9.8. 外部クロック信号	26	16.13. タイマ/カウンタのタイミング	92
9.9. システムクロック出力緩衝部	27	16.14. 16ビット タイマ/カウンタ1用レジスタ	93
9.10. システムクロック前置分周器	27	17. タイマ/カウンタ0と1の前置分周器	99
9.11. クロック関係用レジスタ	28	17.1. 内部クロック元	99
10. PM – 電力管理と休止形態	30	17.2. 前値分周器リセット	99
10.1. 休止形態種別	30	17.3. 外部クロック元	99
10.2. アイドル動作	30	17.4. タイマ/カウンタ捕獲ピン・前値分周器用レジスタ	100
10.3. A/D変換雑音低減動作	30	18. PSC – 電力段制御器	101
10.4. パワーダウン動作	31	18.1. 特徴	101
10.5. スタンバイ動作	31	18.2. 概要	101
10.6. 電力削減(電力削減レジスタ)	31	18.3. 16ビットレジスタのアクセス	101
10.7. 消費電力の最小化	31	18.4. 電力段制御器(PSC)説明	102
10.8. 電力管理用レジスタ	33	18.5. 機能説明	102
11. SCRST – システム制御とリセット	35	18.6. 値の更新	104
11.1. AVRのリセット	35	18.7. 重複保護	104
11.2. リセット元	35	18.8. 信号説明	105
11.3. 電源ONリセット	35	18.9. 電力段制御器(PSC)入力	106
11.4. 外部リセット	36	18.10. PSC入力動作1~5: タイミング変更なしで出力非活動	107
11.5. 低電圧検出(BOD)リセット	36	18.11. PSC入力動作6,7: PSC停止、ソフトウェア操作待機	107
11.6. ウォッチドッグシステムリセット	36	18.12. アナログ同期化	107
11.7. 内部基準電圧	36	18.13. 割り込みの扱い	107
11.8. ウォッチドッグタイマ	37		

18.14.	PSCクロック元	108	26.5.	デバッグWIREの制限	202
18.15.	割り込み	108	26.6.	デバッグWIRE用レジスタ	203
18.16.	電力段制御器(PSC)用レジスタ	109	27.	BTLDR – ブートローダ支援 (RWW自己プログラミング)	204
19.	SPI – 直列周辺インターフェース	116	27.1.	特徴	204
19.1.	特徴	116	27.2.	概要	204
19.2.	概要	116	27.3.	フラッシュメモリの応用領域とブートローダ領域	204
19.3.	SSピンの機能	118	27.4.	書き中に読みが可能な領域と不可能な領域	204
19.4.	データ転送形式	119	27.5.	ブートローダ施錠ビット	206
19.5.	SPI用レジスタ	120	27.6.	ブートローダプログラムへの移行	206
20.	CAN – 制御器域網	123	27.7.	自己プログラミングでのフラッシュアドレス指定	207
20.1.	特徴	123	27.8.	フラッシュメモリの自己プログラミング	207
20.2.	概要	123	27.9.	ブートローダ用レジスタ	213
20.3.	CAN規約	123	28.	MEMPRG – メモリプログラミング	215
20.4.	CAN制御器	127	28.1.	プログラムメモリとデータメモリ用施錠ビット	215
20.5.	CANチャネル	127	28.2.	ヒューズビット	216
20.6.	メッセージ項目	129	28.3.	識票バイト	217
20.7.	CANタイム	131	28.4.	校正バイト	217
20.8.	異常管理	131	28.5.	ページ容量	217
20.9.	割り込み	133	28.6.	並列プログラミング	218
20.10.	CANポーレート設定例	134	28.7.	並列プログラミング手順	219
20.11.	CAN用レジスタ	135	28.8.	直列プログラミング	224
21.	LIN/UART	152	28.9.	直列プログラミング手順	224
21.1.	特徴	152	29.	電気的特性	227
21.2.	概要	152	29.1.	絶対最大定格	227
21.3.	LIN規約	152	29.2.	DC特性	227
21.4.	LIN/UART制御器	154	29.3.	速度勾配	228
21.5.	LIN/UART説明	157	29.4.	クロック特性	228
21.6.	LIN/UART用レジスタ	165	29.5.	システムとリセットの特性	229
22.	ADC – A/D変換器	172	29.6.	PLL特性	230
22.1.	特徴	172	29.7.	SPI タイミング特性	230
22.2.	概要	172	29.8.	A/D変換器特性	231
22.3.	操作	174	29.9.	並列プログラミング特性	232
22.4.	変換の開始	174	30.	代表特性	234
22.5.	前置分周と変換タイミング	175	30.1.	ピンプルアップ	234
22.6.	チャネル変更と基準電圧選択	176	30.2.	ピン駆動部能力	234
22.7.	雑音低減機能	177	30.3.	ピン閾値とヒステリシス	236
22.8.	A/D変換の結果	179	30.4.	BOD閾値とアナログ比較器ヒステリシス	239
22.9.	温度測定	180	30.5.	内部発振器周波数	239
22.10.	増幅器	181	30.6.	電力削減レジスタの使用	240
22.11.	A/D変換用レジスタ	183	31.	レジスタ要約	241
23.	ISRC – 電流源	189	32.	命令要約	245
23.1.	特徴	189	33.	障害情報	248
23.2.	代表的な応用	189	34.	外圍器情報	249
23.3.	電流源用レジスタ	190	35.	データシート改訂履歴	250
24.	AC – アナログ比較器	191			
24.1.	特徴	191			
24.2.	概要	191			
24.3.	ADC増幅器の使い方	193			
24.4.	アナログ比較器用レジスタ	194			
25.	DAC – D/A変換器	198			
25.1.	特徴	198			
25.2.	概要	198			
25.3.	操作	198			
25.4.	変換の開始	198			
25.5.	D/A変換用レジスタ	200			
26.	DBG – デバッグWIRE内蔵デバッグ機能	202			
26.1.	特徴	202			
26.2.	概要	202			
26.3.	物理インターフェース	202			
26.4.	ソフトウェア中断点	202			

1. ピン配置

図1-1. TQFP32・QFN32



注: QFNの底面パッドはGNDに半田付けされるべきです。

1.1. ピン概要

表1-1. ピン割り当て説明

ピン番号	ピン名	形式分類	名称、機能、代替機能
5	GND	電源	接地: 0V基準
20	AGND	電源	アナログ接地: アナログ部用0V基準
4	VCC	電源	電力供給:
19	AVCC	電源	アナログ電力供給: これはアナログ部用の供給電力電圧です。
21	AREF	電源	アナログ基準電圧: これはA/D変換器用基準電圧です。 通常使用で本ピンは接続されなければなりません。 ISRC (電流源出力)
8	PB0	I/O	MISO (SPI 主側入力/従側出力), PSCOUT2A(注1) (PSC2 出力A), PCINT0 (ピン変化割り込み0入力)
9	PB1	I/O	MOSI (SPI 主側出力/従側入力), PSCOUT2B(注1) (PSC2 出力B), PCINT1 (ピン変化割り込み1入力)
16	PB2	I/O	ADC5 (アナログ入力チャネル5), INT1 (外部割り込み1入力), ACPN0 (アナログ比較器0 反転入力), PCINT2 (ピン変化割り込み2入力)
23	PB3	I/O	AMP0- (アナログ差動増幅器0 反転入力), PCINT3 (ピン変化割り込み3入力)
24	PB4	I/O	AMP0+ (アナログ差動増幅器0 非反転入力), PCINT4 (ピン変化割り込み4入力)
26	PB5	I/O	ADC6 (アナログ入力チャネル6), INT2 (外部割り込み2入力), ACPN1 (アナログ比較器1 反転入力), AMP2- (アナログ差動増幅器2 反転入力), PCINT5 (ピン変化割り込み5入力)
27	PB6	I/O	ADC7 (アナログ入力チャネル7), PSCOUT1B(注1) (PSC1 出力B), PCINT6 (ピン変化割り込み6入力)
28	PB7	I/O	SCK (SPI クロック入出力), ADC4 (アナログ入力チャネル4), PSCOUT0B(注1) (電力段階制御器(PSC)0 出力B), PCINT7 (ピン変化割り込み7入力)
30	PC0	I/O	INT3 (外部割り込み3入力), PSCOUT1A(注1) (PSC1 出力A), PCINT8 (ピン変化割り込み8入力)
3	PC1	I/O	OC1B (タイマ/カウンタ1比較B出力), PSCIN1 (PSC1 デジタル入力), SS_A (代替SPI 従装置選択入力), PCINT9 (ピン変化割り込み9入力)
6	PC2	I/O	T0 (タイマ/カウンタ0外部クロック入力), TXCAN (CAN送信出力), PCINT10 (ピン変化割り込み10入力)
7	PC3	I/O	T1 (タイマ/カウンタ1外部クロック入力), RXCAN (CAN受信出力), ICPIB (タイマ/カウンタ1捕獲起動代替入力), PCINT11 (ピン変化割り込み11入力)
17	PC4	I/O	ADC8 (アナログ入力チャネル8), AMP1- (アナログ差動増幅器1 反転入力), ACPN3(注2) (アナログ比較器3 反転入力), PCINT12 (ピン変化割り込み12入力)
18	PC5	I/O	ADC9 (アナログ入力チャネル9), AMP1+ (アナログ差動増幅器1 非反転入力), ACPN3 (アナログ比較器3 非反転入力), PCINT13 (ピン変化割り込み13入力)
22	PC6	I/O	ADC10 (アナログ入力チャネル10), ACPN1 (アナログ比較器1 非反転入力), PCINT14 (ピン変化割り込み14入力)
25	PC7	I/O	D2A (D/A変換器出力), AMP2+ (アナログ差動増幅器2 非反転入力), PCINT15 (ピン変化割り込み15入力)
29	PD0	I/O	PSCOUT0A(注1) (PSC0 出力A), PCINT16 (ピン変化割り込み16入力)
32	PD1	I/O	CLKO (システムクロック出力), PSCIN0 (PSC0 デジタル入力), PCINT17 (ピン変化割り込み17入力)
1	PD2	I/O	OC1A (タイマ/カウンタ1比較A出力), MISO_A (プログラミングと代替SPI 主側入力/従側出力), PSCIN2 (PSC2 デジタル入力), PCINT18 (ピン変化割り込み18入力)
2	PD3	I/O	TXD (UART送信データ出力), TXLIN (LIN送信出力), OC0A (タイマ/カウンタ0比較A出力), SS (SPI 従装置選択入力), MOSLA (プログラミングと代替SPI 主側出力/従側入力), PCINT19 (ピン変化割り込み19入力)
12	PD4	I/O	RXD (UART受信データ入力), RXLIN (LIN受信入力), ICPIA (タイマ/カウンタ1捕獲起動入力), ADC1 (アナログ入力チャネル1), SCK_A (プログラミングと代替SPI クロック入出力), PCINT20 (ピン変化割り込み20入力)
13	PD5	I/O	ADC2 (アナログ入力チャネル2), ACPN2 (アナログ比較器2 非反転入力), PCINT21 (ピン変化割り込み21入力)
14	PD6	I/O	ADC3 (アナログ入力チャネル3), ACPN2 (アナログ比較器2 反転入力), INT0 (外部割り込み0入力), PCINT22 (ピン変化割り込み22入力)
15	PD7	I/O	ACP0 (アナログ比較器0 非反転入力), PCINT23 (ピン変化割り込み23入力)
2	PE0	I/O	RESET (外部リセット入力), OCD (内蔵デバッグWIRE機能入出力), PCINT24 (ピン変化割り込み24入力)
10	PE1	I/O	XTAL1 (発振増幅器入力), OC0B (タイマ/カウンタ0比較B出力), PCINT25 (ピン変化割り込み25入力)
11	PE2	I/O	XTAL2 (発振増幅器出力), ADC0 (アナログ入力チャネル0), PCINT26 (ピン変化割り込み26入力)

注1: Atmel ATmega32M1/64M1だけです。

注2: 技術試供品でのACP3交換機能はPC4に配置されていません。これはPE2に配置されています。

2. 注文情報

デバイス	速度(MHz)	電源電圧	注文符号	外囲器	動作範囲
ATmega16M1	16	2.7~5.5V	ATmega16M1-AU	32A	工業用 (-40°C~85°C)
			ATmega16M1-MU	PV	
ATmega32M1	16	2.7~5.5V	ATmega32M1-AU	32A	工業用 (-40°C~85°C)
			ATmega32M1-MU	PV	
ATmega64M1	16	2.7~5.5V	ATmega64M1-AU	32A	工業用 (-40°C~85°C)
			ATmega64M1-MU	PV	

注: 全外囲器が鉛フリー、完全にLHF。

外囲器形式	
32A	32リード、1.0mm厚 プラスティック4方向平板外囲器 (TQFP)
PV	32パッド、7×7×1mm 0.65mmピッチ 4方向平板リードなし外囲器 (QFN)

3. 概要

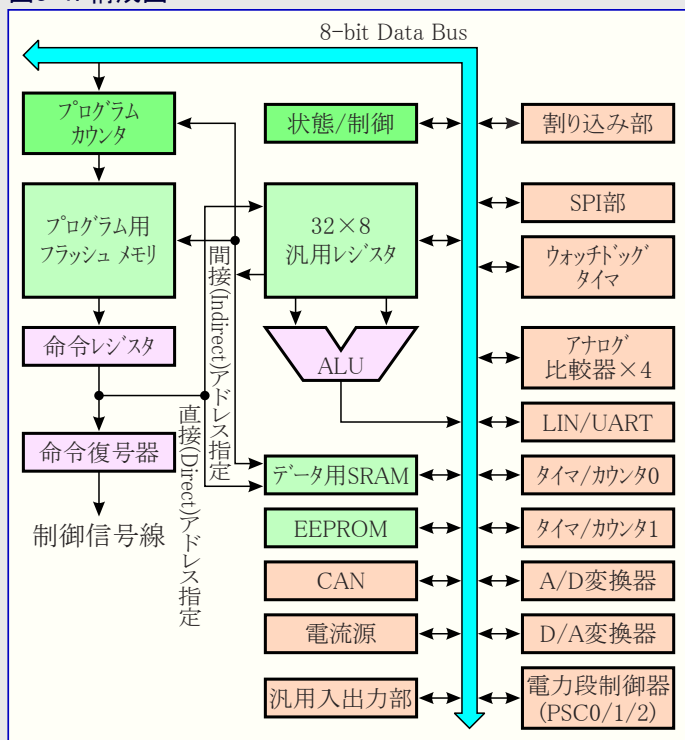
Atmel ATmega16M1/32M1/64M1はAVR RISC構造の低消費CMOS 8ビットマイクロコントローラです。1周期で実行する強力な命令はMHzあたり1 MIPSにも達し、実行速度対電力消費の最適化が容易に行えます。

3.1. 構成図

AVR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て最大10倍の単位処理量向上効果があります。

Atmel ATmega16M1/32M1/64M1は次の特徴、書き込み中読める能力を持つ16/32/64Kバイトの実装書き換え可能なフラッシュメモリと512/1K/2KバイトのEEPROM、1K/2K/4KバイトのSRAM、27本の汎用入出力線、32個の汎用レジスタ、1つの電動機電力段制御器(PSC)、比較動作やPWMを含む柔軟性のある2つのタイマ/カウンタ、ハードウェアLIN動作付きの1つのUART、設定可能な増幅器を持つ2つの差動入力段付きの任意選択差動入力付き11チャンネルの10ビットA/D変換器、10ビットD/A変換器、設定可能な別個の内部発振器付きウォッチドッグタイマ、SPI直列ポート、内蔵デバッグシステム、ソフトウェアで選べる4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、CAN、LIN/UART、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。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、A/D変換器を除く全ての周辺機能とCPUが停止します。スタンバイ動作では水晶発振子/セラミック振動子用発振器が動作し、一方デバイスのその他は休止します。これは低消費電力と非常に速い起動の組み合わせを許します。

図3-1. 構成図



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こ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プログラムは真の「書き込み中の読み出し可」動作により、応用領域フラッシュメモリ更新中も実行を継続します。モリシックチップ上の実装書き換え可能なフラッシュメモリと、8ビットRISC型CPUの組み合わせによるAtmel ATmega16M1/32M1/64M1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。

Atmel ATmega16M1/32M1/64M1 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。

3.2. ピン説明

3.2.1. VCC

デジタル電源ピン。

3.2.2. GND

接地ピン。

3.2.3. PB7~PB0 (ポートB)

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

ポートBは「ポートBの交換機能」で一覧されるATmega16M1/32M1/64M1の様々な特殊機能も扱います。詳細については「ポートBの交換機能」を参照してください。

関連リンク [55頁の「ポートBの交換機能」](#)

3.2.4. PC7~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8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ピンには吐き出し電流が流れます。リセット条件が有効になるとき、クロックが動作していなくても、ポートCピンはHi-Zになります。

ポートCは「[ポートCの交換機能](#)」で一覧されるATmega16M1/32M1/64M1の特殊機能も扱います。詳細については「[ポートCの交換機能](#)」を参照してください。

[関連リンク](#) [57ページの「ポートCの交換機能」](#)

3.2.5. PD7~PD0 (ポートD)

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

ポートDは「[ポートDの交換機能](#)」で一覧されるAtmel ATmega16M1/32M1/64M1の様々な特殊機能も扱います。詳細については「[ポートDの交換機能](#)」を参照してください。

[関連リンク](#) [59ページの「ポートDの交換機能」](#)

3.2.6. PE2~PE0 (ポートE) RESET/XTAL1/XTAL2

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

RSTDISBLヒューズがプログラム(0)されると、PE0はI/Oピンとして使われます。PE0の電気的特性がポートE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、PE0はリセット入力として使われます。クロックが動作していても、最小パルス幅より長いこのピンのLowレベルはリセットを生成します。最小パルス幅は「[システムとリセットの特性](#)」で与えられます。より短いパルスはリセットの生成が保証されません。

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

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

ポートEの様々な特殊機能は「[ポートEの交換機能](#)」で詳しく述べられます。

[関連リンク](#) [23ページの「クロックシステムとその配給」](#)
[62ページの「ポートEの交換機能」](#)

3.2.7. AVCC

AVCCはA/D変換器、D/A変換器、電流源用の供給電圧(電源)ピンです。例えばA/D変換やD/A変換が使われなくても、外部的にVCCへ接続されるべきです。A/D変換が使われる場合、[低域通過濾波器を通してVCCに接続](#)されるべきです。

3.2.8. AREF

AREFはA/D変換器用アナログ基準(電圧)ピンです。

4. 資料

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

5. 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。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。

6. 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。

7. AVR CPU コア

7.1. 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、この項内で後述されます。

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

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

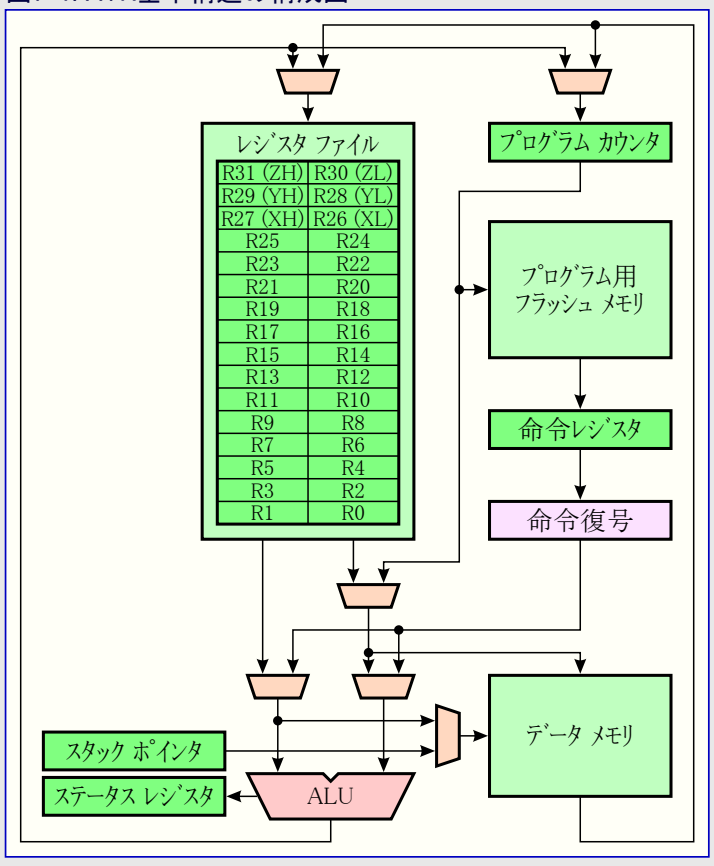
I/Oメモリ空間は制御レジスタ、SPI、他のI/O機能としてCPU周辺機能用の64アドレスを含みます。I/Oメモリは直接またはレジスタファイルの次のデータ空間位置\$20~\$5Fとしてアクセスできます。加えて、本デバイスにはST/STS/STDとLD/LDS/LDD命令だけ使えるSRAM内の\$60~\$FFに拡張I/O空間があります。

7.2. ALU (Arithmetic Logic Unit)

高性能なAVRのALUは32個の全汎用レジスタに直接接続され動作します。汎用レジスタ間または汎用レジスタと即値間の演算操作は単一クロック周期内で実行されます。ALU操作は算術演算、論理演算、ビット操作の3つの主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 [245頁の「命令要約」](#)

図7-1. AVR基本構造の構成図



7.3. ステータス レジスタ

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

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

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

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	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 – I : 全割り込み許可 (Global Interrupt Enable)

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

● ビット6 – T : ビット変数 (Bit Copy Storage)

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

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

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

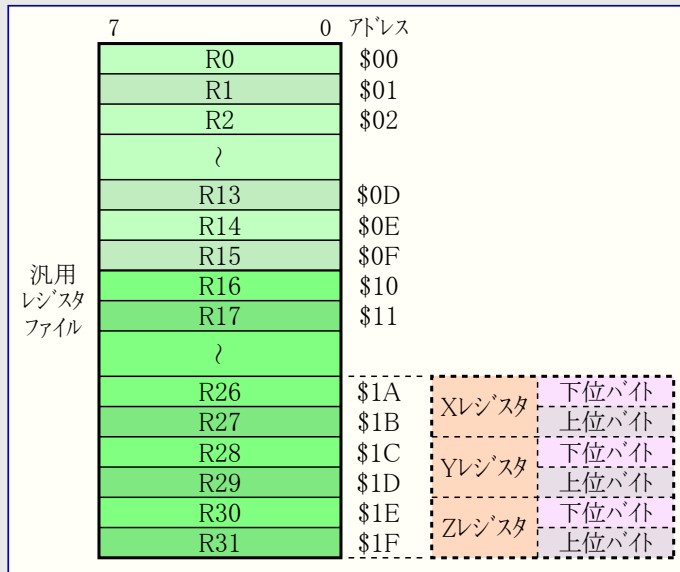
7.4. 汎用レジスタ ファイル

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

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

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

図7-2. AVR CPU 汎用レジスタ構成図

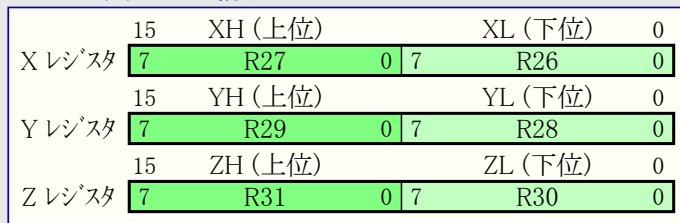


7.4.1. Xレジスタ, Yレジスタ, Zレジスタ

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)。

図7-3. X,Y,Zレジスタ構成図



7.5. スタック ポインタ

スタックは主に一時データの保存、局所変数の保存、割り込みとサブルーチン呼び出し後の戻りアドレスの保存に使われます。スタック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7-1. 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。

7.5.1. SPH,SPL – スタックポインタレジスタ上位/下位バイト (Stack Pointer Register Low and High byte)

SPHとSPLのレジスタ対は16ビット値のSPを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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とSPL (SP)

変位 : \$5D (\$3D)

リセット : \$04FF(ATmega16M1), \$08FF(ATmega32M1), \$10FF(ATmega64M1)

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	15	14	13	12	11	10	9	8
	-		-	(SP12)	(SP11)	SP10~8		
アクセス種別	R	R	R	R/W	R/W	R/W	R/W	R/W
リセット値	0	0	0	0,0,1	0,1,0	1,0,0	0	0
ビット	7	6	5	4	3	2	1	0
	SP7~0							
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12~0 – SP12~0 : スタックポインタ値 (Stack Pointer value)

SPHとSPLはSPに組み合わせられます。ATmega16M1はSP12とSP11が、ATmega32M1はSP11が利用不能です。

(訳注) 原書で7.5.1~3.のATmega16M1/32M1/64M1に対する各SPレジスタ記述は7.5.1.として纏めました。)

7.6. 16ビットレジスタのアクセス

AVRのデータバスは8ビット幅で、故に16ビットレジスタのアクセスは非分断操作が必要です。これらのレジスタは2つの読みまたは書きの操作を用いてバイトアクセスされなければなりません。16ビットレジスタは16ビットバスを用いて8ビットバスと1つの一時レジスタに接続されます。

書き込み操作については、16ビットレジスタの上位バイトが下位バイトに先立って書かれなければなりません。そしてその上位バイトは一時レジスタに書かれます。16ビットレジスタの下位バイトが書かれる時に、同じクロック周期で一時レジスタが16ビットレジスタの上位バイトに複写されます。

読み込み操作については、16ビットレジスタの下位バイトが上位バイトに先立って読まれなければなりません。CPUによって下位バイトが読まれる時に、下位バイトが読まれるのと同じクロック周期で16ビットレジスタの上位バイトが一時レジスタに複写されます。その後上位バイトが読まれる時は、この一時レジスタから読みます。

これはレジスタの読み書き時に16ビットレジスタの上位と下位のバイトが常に同時にアクセスされることを保証します。

非分断16ビットレジスタ読み書き操作中に割り込みが起動されて同じレジスタをアクセスする場合、この時間制限手順を不正にし得ます。これを防ぐために、16ビットレジスタの読み書き時に割り込みを禁止することができます。

(訳注) 原書の本文の記述はXMEGA用の記述をそのまま複写したものとと思われます。そのため、書き込み操作の上位と下位のバイトの書き順が他のAVRと逆になっています。これによって、書き込み操作に関する他の殆どの部分での記述と矛盾が生じています。本書ではこの書き込み操作に関する書き順を矛盾が生じないように、本項と関連する他の一部の記述を修正しています。

7.7. 命令実行タイミング

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

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

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7-4. 命令の取得と実行の並列動作

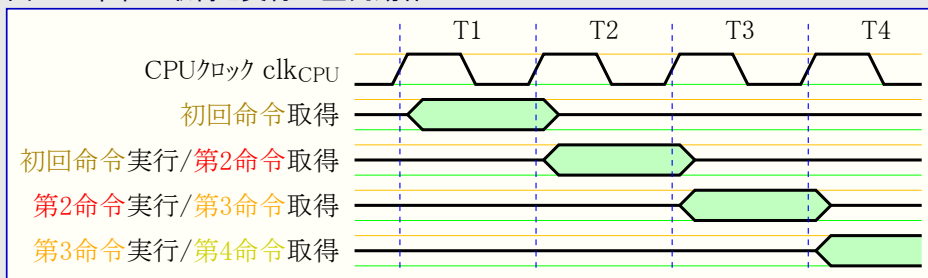
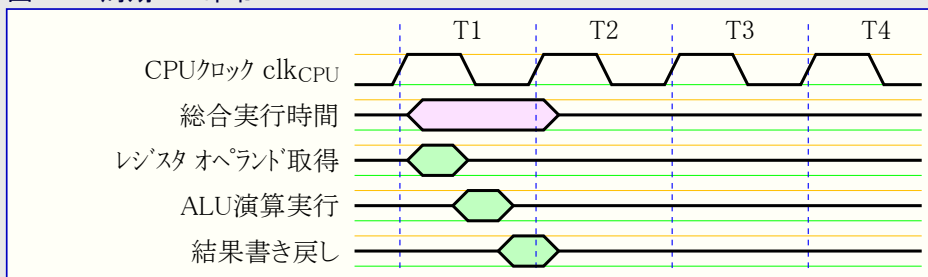


図7-5. 1周期ALU命令



7.8. リセットと割り込みの扱い

AVRは多くの異なる割り込み元を提供します。これらの割り込みと独立したリセット ベクタ各々はプログラム メモリ空間内に独立したプログラム ベクタを持ちます。全ての割り込みは割り込みを許可するために、ステータスレジスタ(SREG)の全割り込み許可(I)ビットと共に論理1が書かれなければならない個別の許可ビットを割り当てられます。BLB02またはBLB12 ブート施錠ビットがプログラム(0)されると、プログラム カウンタ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。それらは決められた優先順位を持ちます。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アナログ比較器0(ANACOMP0)割り込みです。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、復帰時の回復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、これがどう使用できるかを示します。

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せん。

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 [215頁の「MEMPROG - メモリプログラミング」](#)
[204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」](#)

7.8.1. 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クロック周期要します。複数周期命令実行中に割り込みが起こると、その割り込みが扱われる前に、この命令が完了されます。MCUが休止形態の時に割り込みが起こると、割り込み実行応答時間は4クロック周期増やされます。この増加は選んだ休止形態からの起動時間に加えてです。割り込み処理ルーチンからの復帰は4クロック周期要します。これらの4クロック周期中、プログラムカウンタ(PC:2バイト)がスタックから取り戻され(ポップ)、スタックポインタは増加され(+2)、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されます。

8. AVRのメモリ

8.1. 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地址です。

8.2. 実装書き換え可能なプログラム用フラッシュメモリ

ATmega16M1/32M1/64M1はプログラム保存用に実装書き換え可能な16/32/64Kバイトのフラッシュメモリをチップ上に含みます。全てのAVR命令が16または32ビット幅のため、このフラッシュメモリは8/16/32K×16ビットとして構成されます。

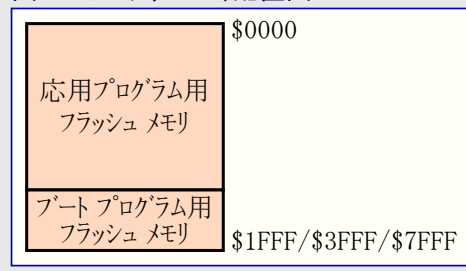
フラッシュメモリは最低10,000回の消去/書き込み回数の耐久性があります。ATmega16M1/32M1/64M1のプログラムカウンタ(PC)は13/14/15ビット幅、故に8/16/32Kプログラムメモリ位置のアドレス指定です。ブートプログラム領域の操作と関係するソフトウェア保護用ブート施錠ビットは「**BTLDR - ブートローダ - 書き込み中読み出し可能な自己プログラミング**」で詳細に記述されます。SPIピンを使うフラッシュデータ直列書き込みの記述については「**MEMPROG - メモリプログラミング**」を参照してください。

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 [204頁の「BTLDR - ブートローダ支援 - 書き中読み可能な自己プログラミング」](#)
[215頁の「MEMPROG - メモリプログラミング」](#)
[14頁の「命令実行タイミング」](#)

図8-1. プログラムメモリ配置図



8.3. 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1280/2304/4352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、内部データSRAMに充てます。先頭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1024/2048/4096位置は内部データ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。

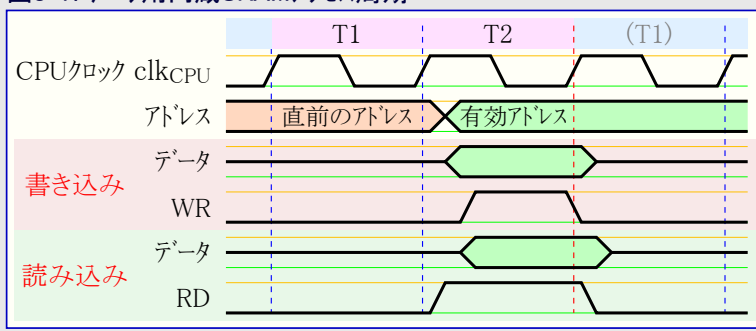
本デバイスの32個の汎用レジスタ、64個のI/Oレジスタ、160個の拡張I/Oレジスタ、1/2/4Kバイトのデータ用内蔵SRAMはこれら全てのアドレス指定種別を通して全部アクセスできます。

8.3.1. データメモリアクセスタイミング

データ用内蔵SRAMアクセスは右図記載のように2 clk_{CPU}周期で実行されます。

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

図8-7. データ用内蔵SRAMアクセス周期



8.4. データ用EEPROMメモリ

ATmega16M1/32M1/64M1は512/1K/2Kバイトのデータ用EEPROMを含みます。それは単一バイトが読み書きできる分離したデータ空間として構成されます。EEPROMは最低100,000回の消去/書き込み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 205頁の「MEMPROG – メモリプログラミング」

8.4.1. EEPROMアクセス

EEPROMアクセスレジスタはI/O空間でアクセス可能です。

EEPROM書き込み(訳注:原文はアクセス)時間は表8-2.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**次バイトが書ける時を検知**してください。使用者コードがEEPROMに書く命令を含む場合、いくつかの予防処置が取られねばなりません。嚴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「**非分離バイト書き込み**」と「**分離バイト書き込み**」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。

8.4.2. 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 $\overline{\text{RESET}}$ 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。

8.5. 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 215頁の「MEMPROG – メモリプログラミング」

241頁の「レジスタ要約」

245頁の「命令要約」

8.5.1. 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。

8.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8-1.で与えられます。EEPEビットは消去と書き込み動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

8.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。

8.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8-1.で与えられます)。EEPEビットは消去動作が完了されるまで設定(1)に留まります。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

8.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8-1.で与えられます)。EEPEビットは書き込み動作が完了されるまで設定(1)に留まります。書かれるべき位置が書き込み前に消去されてしまっていなければ、元の保存したデータは失ったとみなされなければなりません。デバイスがプログラミング動作中、他のどのEEPROM操作の実行も不可能です。

8.6. メモリ関係レジスタ

8.6.1. EEARH,EEARL – EEPROMアドレスレジスタ上位/下位バイト (EEPROM Address Register Low and High byte)

EEARHとEEARLのレジスタ対は16ビット値のEEARを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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とEEARL (EEAR)

変位 : \$41 (\$21)

リセット : '0000000x xxxxxxxx'(ATmega16M1), '000000xx xxxxxxxx'(ATmega32M1), '00000xxx xxxxxxxx'(ATmega64M1)

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	15	14	13	12	11	10	9	8
	-	-	-	-	-	(EEAR10)	(EEAR9)	EEAR8
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	不定	不定	不定
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8~0 – EEAR8~0 : EEPROMアドレス (EEPROM Address)

EEPROMアドレスレジスタ(EEARHとEEARL)は512/1K/2K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0~511/1023/2047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。

注: EEAR9はATmega32M1/64M1でだけ、EEAR10はATmega64M1でだけ有効です。

8.6.2. 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。

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。

8.6.3. 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。

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EEPROMnへのどの書き込みも無視されます。リセット中、EEPROMnビットはEEPROMがプログラミング作業中を除いて'00'にリセットされます。

表8-1. EEPROMプログラミング種別

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EEPROMn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)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8-2. EEPROM書き込み時間

項目	校正付き内蔵RC発振器周期数	代表
EEPROM書き込み(CPU)	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。

8.6.4. 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 : \$3A (\$1A)

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\$1A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)

8.6.5. 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 : \$39 (\$19)

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\$19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)

8.6.6. 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

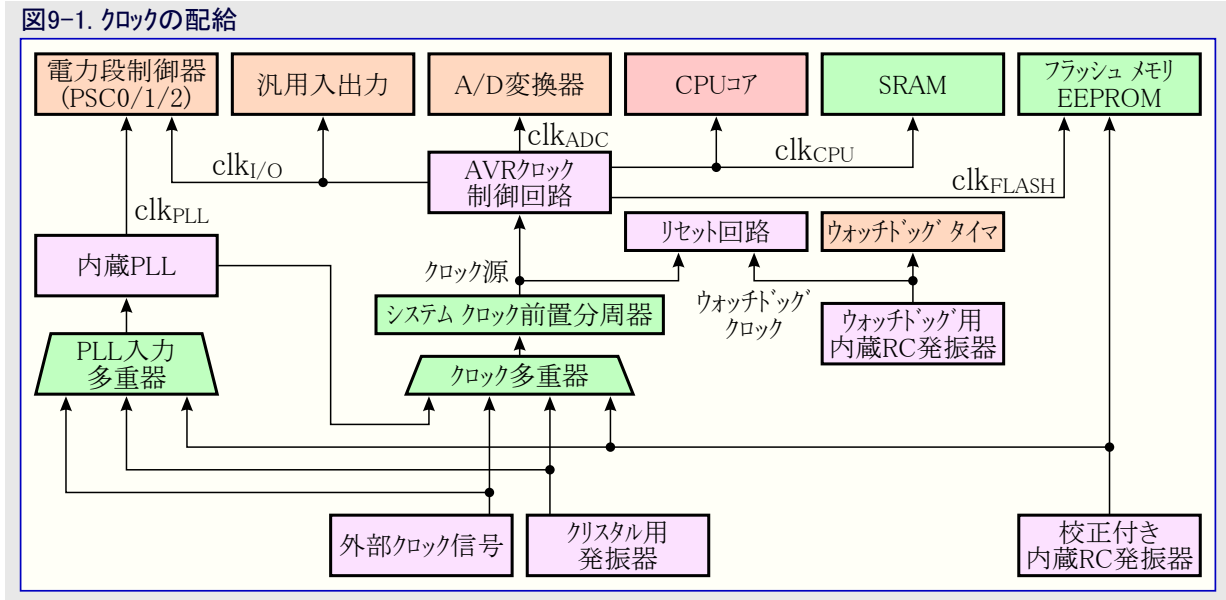
- ビット7~0 – GPIOR07~0 : 汎用I/Oレジスタ0 (General Purpose I/O)

9. システム クロックとクロック選択

9.1. 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。



9.1.1. CPUクロック - clk_{CPU}

CPUクロックはAVRコアの動作と関係する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。

9.1.2. 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、特定の部分は $clk_{I/O}$ が停止される時に非同期で実行されます(**訳注**:整合性のため本行修正)。

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。

9.1.3. フラッシュクロック - clk_{FLASH}

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

9.1.4. PLLクロック - clk_{PLL}

PLLクロックは64/32MHzから直接的にクロック駆動されるのを高速周辺機能に許します。16MHzクロックもCPU用に分周されます。

9.1.5. 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。

9.2. クロック元

本デバイスには右図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。

各クロック選択に対する様々な選択は次項で得られます。CPUがパワーダウンから起動する時、選んだクロック元は命令実行開始前に安定な発振器動作を保証する起動時間に使われます。CPUがリセットから始まる時、これらは通常動作開始前に安定電圧へ達するのを電源に許す付加遅延です。ウォッチドッグ発振器はこの起動時間の実時間部のタイミングに使われます。各計時完了に使われるウォッチドッグ(WDT)発振器の周期数は右下の表で示されます。

表9-1. クロック種別選択

システム クロック	PLL入力	CKSEL3~0
外部クリスタル/セラミック発振子	校正付き内蔵RC発振器	1111~1000
(予約)	(予約)	011x
4分周PLL出力	外部クリスタル/セラミック発振子	0101
外部クリスタル/セラミック発振子	外部クリスタル/セラミック発振子	0100
4分周PLL出力 (16MHz)	校正付き内蔵RC発振器	0011
校正付き内蔵RC発振器	校正付き内蔵RC発振器	0010
4分周PLL出力	外部クロック信号	0001
外部クロック信号	校正付き内蔵RC発振器	0000

注: 1=非プログラム、0=プログラム

表9-2. WDT発振器の代表計時完了値、周期数

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

9.3. 既定のクロック元

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

9.4. クリスタル用低電力発振器

XTAL1とXTAL2は下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。クリスタル発振子またはセラミック振動子のどちらでも使えます。

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ありません。

C1とC2はクリスタル発振子とセラミック振動子の両方について常に等しくすべきです。このコンデンサの最適値は使うクリスタル発振子やセラミック振動子、浮遊容量の量、その環境の電磁雑音に依存します。クリスタル発振子使用に対するコンデンサ選択について初期の指針のいくつかは表9-3.で与えられます。セラミック振動子については製造業者によって与えられたコンデンサ値が使われるべきです。

低電力発振器は示された周波数範囲で各々最適化された3つの異なる種別で動作できます。この動作種別は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。

図9-2. クリスタル発振子接続図

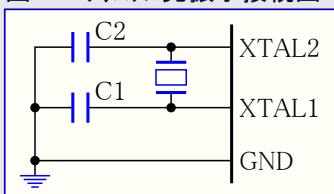


表9-3. クリスタル用低電力発振器動作種別

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

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

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

表9-4. クリスタル発振子/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を保証します。デバイスの最高周波数付近での動作でないとき、応用にとって起動での周波数安定性が重要でない場合はクリスタル発振子も使えます。

9.5. 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マとリセット付加遅延計時器に使われません。

表9-5. 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9-6. 校正付き内蔵RC発振器用起動遅延時間選択表

SUT1,0	パワーダウンからの 起動遅延時間	リセットからの付加遅延 時間 (VCC=5.0V)	推奨使用法
0 0	6×CK	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 [27頁の「システムクロック前置分周器」](#)
[217頁の「校正バイト」](#)
[28頁の「OSCCAL - 発振校正レジスタ」](#)

9.6. PLL

高い精度と周波数のPWM波形を生成するために、電力段階制御器(PSC)は高周波数クロック入力が必要です。このクロックはPLLによって生成されます。全てのPWM精度を維持するため、PLLの周波数係数はソフトウェアによって設定されなければなりません。8MHzのシステムクロックでのPLL出力は32MHzまたは64MHzです。

9.6.1. 内部PLL

Atmel ATmega16M1/32M1/64M1の内部PLLは公称1MHzから64倍されたクロック周波数を生成します。この1MHzのPLL入力クロック元は1MHzに分周された内蔵RC発振器の出力です。

PLLはこのRC発振器に固定化し、発振校正レジスタ(OSCCAL)経由のRC発振器調整は同時に高速周辺機能クロックも調整します。けれども分周したRC発振器が1MHzより高い周波数にされるのが可能でも、高速周辺機能クロックは(最悪の場合)70MHzで飽和し、最高周波数での発振に留まります。この場合のPLLがRC発振器クロックのどれにも固定化されないことに注意されるべきです。

従ってPLLを正しい動作範囲に保持するため、1MHzより高い周波数にOSCCAL補正を行わないことが推奨されます。内部PLLはPLL制御/状態レジスタ(PLLCSR)のPLL許可(PLLE)ビットが設定(1)される時だけ許可されます。PLLCSRのPLL固定化(PLOCK)ビットはPLLが固定化されている時に設定(1)されます。

内部1MHz RC発振器とPLLはパワーダウンとスタンバイの休止形態でOFFへ切り替えられます。

図9-3. 高速周辺機能クロック(CLKPLL)構成図

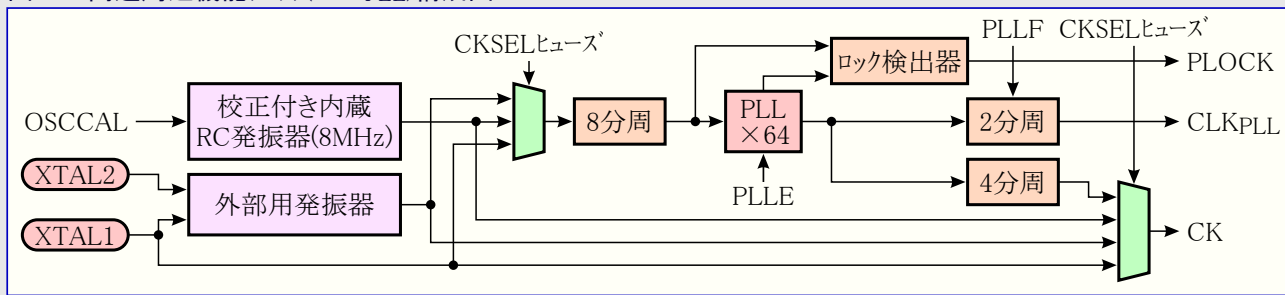


表9-7. PLLシステム クロック用起動遅延時間選択表

CKSEL3~0	SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間 (VCC=5.0V)
	0 0	1K × CK	14 × CK
0 1 0 1 (外部水晶/セラミック発振子)	0 1	1K × CK	14 × CK + 4ms
	1 0	16K × CK	14 × CK + 4ms
	1 1	16K × CK	14 × CK + 64ms
	0 0	1K × CK	14 × CK
0 0 1 1 (校正付き内蔵RC発振器)	0 1	1K × CK	14 × CK + 4ms
	1 0	1K × CK	14 × CK + 64ms
	1 1	16K × CK	14 × CK
	0 0	6 × CK (注)	14 × CK
0 0 0 1 (外部クロック信号)	0 1	6 × CK (注)	14 × CK + 4ms
	1 0	6 × CK (注)	14 × CK + 64ms
	1 1	(予約)	

注: この値は正しい再起動を提供せず、このクロック機構でパワーダウン動作を使ってはいけません。

9.7. 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。このクロックはウォッチドッグ発振器によって使われます。

9.8. 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XTAL1が右図で示されるように駆動されるべきです。外部クロックでデバイスを走行するためには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。

このクロック元が選ばれると、起動時間は下表で示されるようにSUTヒューズによって決定されません。

表9-8. 外部クロック信号動作種別

CKSEL3~0	周波数範囲
0 0 0 0	0~16MHz

図9-4. 外部クロック信号駆動接続図

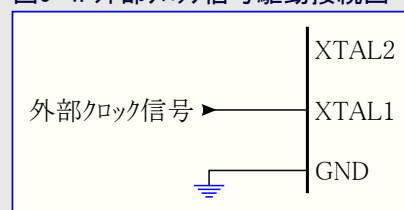


表9-9. 外部クロック信号駆動用起動遅延時間選択表

SUT1,0	パワーダウンからの起動遅延時間	リセットからの付加遅延時間 (VCC=5.0V)	推奨使用法
0 0	6 × CK	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。2%より大きな変化が必要とされる場合、変化中にMCUがリセットに保たれることを保証してください。

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。

9.9. 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われると、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。

9.10. 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。clk_{CPU}、clk_{FLASH}、clk_{I/O}、clk_{ADC}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プルカウンタは分周されないクロック周波数で走行し、CPUのクロック周波数より速いかもしれません。従って例え(カウンタ値が)読めるとしても、前置分周器の状態を決めることはできず、1から他へのクロック分周値切り替えを行う正確な時間は必ずしも予測できません。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 [28頁の「CLKPR - クロック前置分周レジスタ」](#)

9.11. クロック関係レジスタ

9.11.1. 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。

9.11.2. PLLCSR – PLL制御/状態レジスタ (PLL 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 : PLLCSR

変位 : \$49 (\$29)

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\$29です。

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	PLL F	PLL E	PLL OCK
アクセス種別	R	R	R	R	R	R/W	R/W	R
リセット値	0	0	0	0	0	0	0	0

● ビット2 – PLLF : PLL周波数係数 (PLL Factor)

PLLFビットはPLLの分周係数選択に使われます。

値	説明
1	PLLFが設定(1)されると、PLL出力は64MHzです。
0	PLLFが解除(0)されると、PLL出力は32MHzです。

● ビット1 – PLLE : PLL許可 (PLL Enable)

PLLEが設定(1)されると、PLLが起動され、必要とされるなら、PLL基準クロックとして内蔵RC発振器が起動されます。システムクロック元としてPLLが選ばれると、このビットに対する値は常に1です。

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

PLOCKビットが設定(1)されると、PLLが基準クロックに固定化され、高速周辺機能に対するCLKPLLの許可は安全です。PLLが許可された後、PLLが固定化するのに約100µsかかります。

9.11.3. CLKPR – クロック前置分周レジスタ (Clock Prescale 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9-10. クロック前置分周器選択

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

10. PM – 電力管理と休止形態

10.1. 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10-1. 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					発振器 主クロック 供給元	復帰起動要因 (割り込み・リセット)					
	clk CPU	clk FLASH	clk IO	clk ADC	clk PLL		INT 0~3	電力段 制御器 (PSC)	SPM EEPROM 操作可	A/D 変換 完了	ウォッチ ドッグ	その他 I/O
アイドル			○	○	○	○	○	○	○	○	○	○
A/D変換雑音低減				○	○	○	②	○	○	○	○	
パワーダウン							②				○	
スタンバイ(注1)						○	②				○	

注1: クロック元として選ばれた外部クリスタル発振子またはセラミック振動子でだけ推奨されます。

② INT0~3のレベル割り込みだけです。

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

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系統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ると、MCUは起動し、リセットベクタから実行します。

10.2. アイドル動作

休止種別選択(SM2~0)ビットが'000'を書かれる時の**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、UART、アナログ比較器、A/D変換器、タイマ/カウンタ、ウォッチドッグ、割り込み機構の継続動作を許します。この休止形態は基本的にclkCPUとclkFLASHを停止する一方、他のクロックに走行を許します。

アイドル動作はMCUにタイマ溢れやUARTの送信完了などの内部割り込みだけでなく、外部で起動された割り込みからの起動も可能にします。アナログ比較器割り込みからの起動が必要とされないなら、アナログ比較器n制御(ACnCON)レジスタの**アナログ比較器n許可(ACnEN)ビット**を解除(0)することによってアナログ比較器を電源断にできます。これはアイドル動作での消費電力を削減します。

関連リンク [194頁の「ACnCON – アナログ比較器n制御レジスタ」](#)

10.3. A/D変換雑音低減動作

SM2~0ビットが'001'を書かれる時の**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、(クロック元が外部(T0またはT1)ならば)タイマ/カウンタ(脚注1参照)、ウォッチドッグの(許可されていれば)継続動作を許します。この休止形態は基本的にclkI/O、clkCPU、clkFLASHを停止する一方、他のクロックに走行を許します。

これはA/D変換に対する雑音環境を改善し、より高い分解能の測定を可能にします。A/D変換器が許可されている場合、この動作に移行すると、変換が自動的に始まります。A/D変換完了割り込みからの他、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
- タイマ/カウンタの割り込み (脚注1参照)
- 電力段制御器(PSC)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 (脚注2参照)

- (訳注) 1. 原書では存在しない非同期タイマ/カウンタを記載していますが、本書では改訂Eでの記述に基づいて変更しています。けれども、例え本文記述のように外部クロック入力時でも、タイマ/カウンタ本体部分がclkI/Oで動作するため、clkI/Oが停止されるA/D変換雑音低減動作ではタイマ/カウンタが動作せず、従ってタイマ/カウンタ割り込みでCPUを起こすことはできないでしょう。
2. 図13-1.で示されるピン変化割り込みでは上記同様にclkI/Oが停止されると動作しないでしょう。けれども図以外にそれ用の回路となっている場合は動作する(CPUを起こせる)でしょう。

10.4. パワータウン動作

休止種別選択(SM2~0)ビットが'010'を書かれると、SLEEP命令はMCUをパワータウン動作へ移行させます。この動作では外部発振器が停止される一方で、外部割り込み、ウォッチドッグ機能は(許可ならば)継続して動作します。

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
- INTnの外部レベル割り込み
- ピン変化割り込み (前頁の脚注2参照)

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タ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。

パワータ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 [24頁の「クロック元」](#)
[45頁の「EXTINT - 外部割り込み」](#)

10.5. スタンバイ動作

外部クリスタル発振子/セラミック振動子クロック種別が選ばれ、SM2~0ビットが'110'のとき、SLEEP命令はMCUをスタンバイ動作へ移行させます。この動作は(外部クリスタル用)発振器が走行(動作)を保たれる例外を除いてパワータ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。

10.6. 電力削減

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

10.7. 消費電力の最小化

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

10.7.1. A/D変換器 (ADC)

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

関連リンク [172頁の「ADC - A/D変換器」](#)

10.7.2. アナログ比較器

アイドル動作へ移行するとき、アナログ比較器は使われないなら、禁止されるべきです。A/D変換雑音削減動作へ移行するとき、アナログ比較器は禁止されるべきです。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 [191頁の「AC - アナログ比較器」](#)

10.7.3. 低電圧検出器 (BOD)

低電圧検出器(BOD)が応用で必要とされないなら、この単位部はOFFにされるべきです。低電圧検出器がBODLEVELヒューズによって許可されていると全休止形態で許可され、故に常時電力を消費します。これはより深い休止形態での総消費電流にとって重要な一因になります。

関連リンク [36頁の「低電圧検出\(BOD\)リセット」](#)

10.7.4. 内部基準電圧

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

関連リンク [36頁の「内部基準電圧」](#)

10.7.5. ウォッチドッグ タイマ

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

関連リンク [37頁の「ウォッチドッグ タイマ」](#)

10.7.6. 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 [53頁の「デジタル入力許可と休止形態」](#)
[186頁の「DIDR0 – デジタル入力禁止レジスタ0」](#)
[187頁の「DIDR1 – デジタル入力禁止レジスタ1」](#)

10.7.7. 内蔵デバッグ機能 (dW)

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

10.8. 電力管理用レジスタ

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

この休止形態制御レジスタは電力管理用の制御ビットを含みます。

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4つの休止形態の1つを選びます。

表10-2. 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	(予約)
1 0 0	(予約)
1 0 1	(予約)
1 1 0	スタンバイ動作 (注)
1 1 1	(予約)

注: スタンバイ動作は外部クリスタル発振子またはセラミック振動子で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。

10.8.2. 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
	-	PRCAN	PRPSC	PRTIM1	PRTIM0	PRSPI	PRLIN	PRADC
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 – PRCAN : 制御器域網(CAN)電力削減 (Power Reduction CAN)

このビットへの論理1書き込みはこの単位部へのクロック停止によってCANの消費を削減します。CANの再起動時、CAN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5 – PRPSC : 電力段制御器(PSC)電力削減 (Power Reduction PSC)

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

● ビット4 – PRTIM1 : タイマ/カウンタ1電力削減 (Power Reduction Timer/Counter1)

このビットへの論理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3 – 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2 – PRSPI : 直列周辺インターフェース電力削減 (Power Reduction Serial Peripheral Interface)**

このビットへの論理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 – PRLIN : LIN電力削減 (Power Reduction LIN)**

このビットへの論理1書き込みはその部分へのクロック停止によって**LIN/UART**制御器の消費を削減します。LIN/UART制御器の再起動時、LIN/UART制御器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。

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

11.1. AVRのリセット

リセット中、全てのI/Oレジスタはそれらの初期値に設定され、プログラムはリセットベクタから実行を開始します。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。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SUTとCKSELの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。

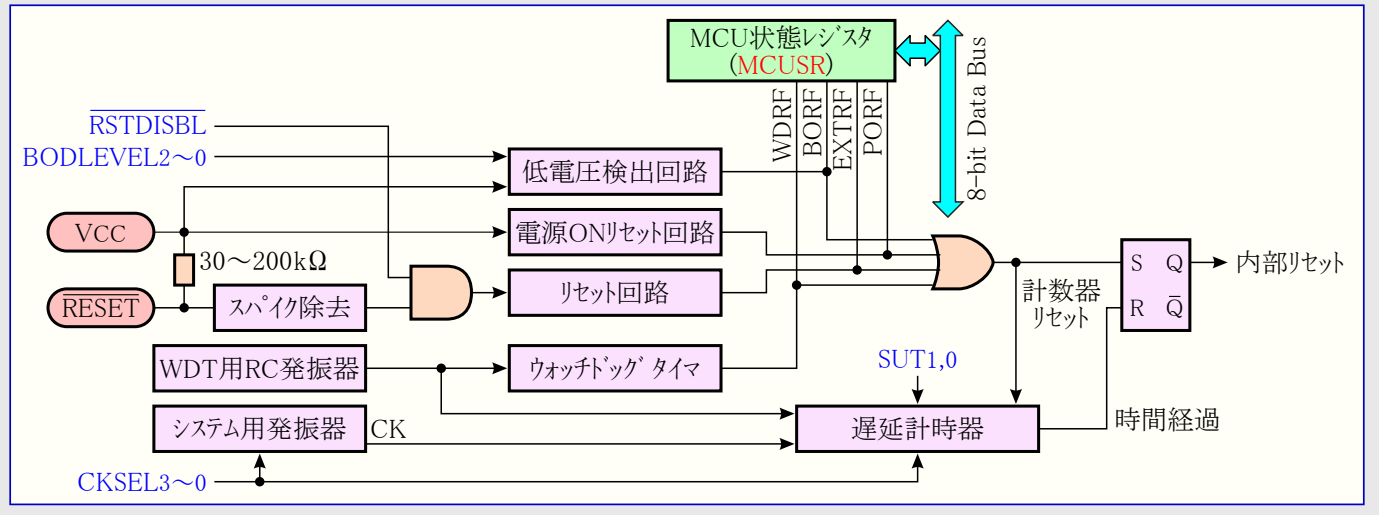
関連リンク 23頁の「システムクロックとクロック選択」

11.2. 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(VPOT)**未満でMCUがリセットされます。
- **外部リセット** …… $\overline{\text{RESET}}$ 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(VCC)が**低電圧検出電圧(VBOT)**未満でMCUがリセットされます。

図11-1. リセット回路構成



11.3. 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(VPOT)への到達はVCCの上昇後にデバイスがどのくらいリセットを保つかを決める遅延計数器(タイマ)を起動します。VCCがこの検出電圧以下に低下すると、リセット信号はどんな遅延もなく再び有効にされます。

図11-2. 内蔵電源ONリセット (RESETはVCCに接続)

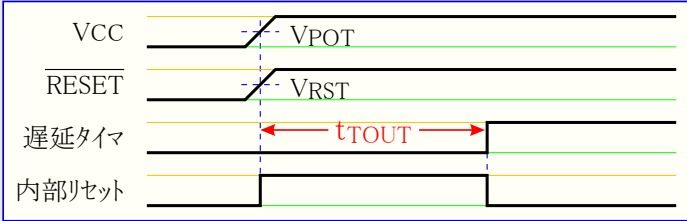
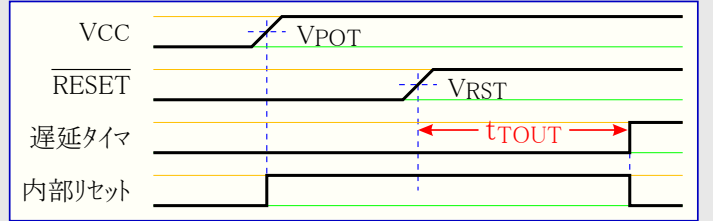


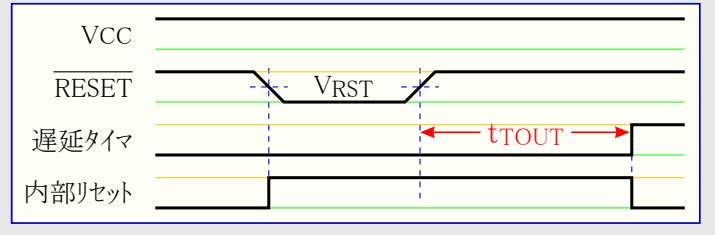
図11-3. 外部RESET信号による延長電源ONリセット



11.4. 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(VRST)に達すると(遅延タイマを起動し)、遅延タイマは遅延時間(t_{TOUT})経過後にMCUを始動します。外部リセットはRSTDISBLヒューズによって禁止することができます。

図11-4. 動作中の外部リセット



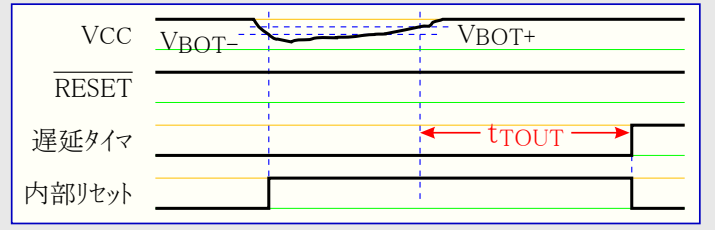
11.5. 低電圧(ブラウンアウト)検出リセット

本デバイス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)回路を持ちます。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 $V_{BOT+} = V_{BOT} + V_{HYST}/2$ 、 $V_{BOT-} = V_{BOT} - V_{HYST}/2$ 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_{BOT-})、低電圧リセットが直ちに有効とされます。VCCが起動電圧以上に上昇すると(右図のV_{BOT+})、(遅延タイマが起動され)遅延タイマは遅延時間(t_{TOUT})経過後にMCUを始動します。

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

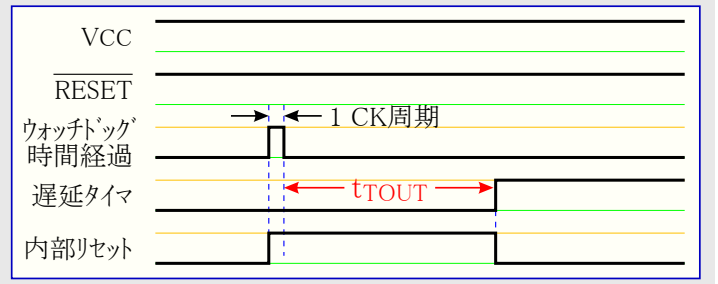
図11-5. 動作中の低電圧検出リセット



11.6. 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)1CK周期幅の短いリセットパルスを生成します。このパルスの下降端で遅延タイマは遅延時間(t_{TOUT})の計時を始めます。

図11-6. 動作中のウォッチドッグ システム リセット



11.7. 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。

11.7.1. 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 (A/D変換多重器選択(ADMUX)レジスタの基準電圧選択(REFS1,0)=11)
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)
4. D/A変換部動作許可時 (D/A変換制御(DACON)レジスタのD/A動作許可(DAEN)=1)

従って低電圧検出(BOD)が許可されていない時に、REFS1,0の設定(=11)またはA/D変換部許可(ADEN=1)またはD/A変換部許可(DAEN=1)後、使用者はアナログ比較器またはA/D変換器またはD/A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4つの状態を避けられます。

11.8. 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。

11.8.1. 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

11.8.2. 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。

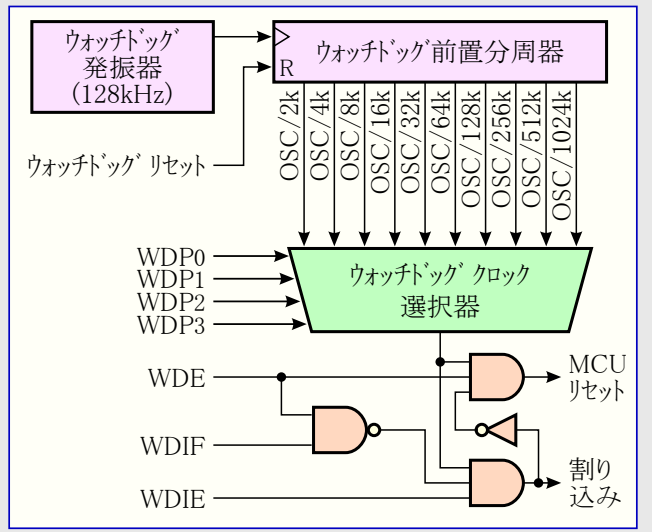
割り込み動作種別では、タイマ計時完了時にWDTが割り込みを生じます。この割り込みは休止形態からデバイスを起動するためや、一般的なシステム タイマとしても使えます。1つの例は或る動作に対して許された最大時間を制限することで、その動作が予測されたより長く走行する時に割り込みを生じます。システム リセット動作種別ではタイマ計時完了時にWDTがリセットを生じます。これは一般的にコード外走行の場合の中断を防止するのに使われます。3つ目の動作種別は先に割り込みを生じ、その後にシステム リセット動作種別に切り替えることで、他の2つの動作種別の組み合わせとなる、割り込み及びシステム リセット動作種別です。この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

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

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

1. 同じ操作(命令)でウォッチドッグ 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11-7. 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<<WDPO) ;WDE=1,計時間隔=0.5s値を取得
            STS   WDTCSR, R16 ;0.5s監視間隔リセット動作開始
            SEI                   ;全割り込み許可
            RET                    ;呼び出し元へ復帰
    
```

C言語プログラム例

```

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

注: 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。

11.9. リセット関係レジスタ

11.9.1. 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 : \$0000xxxx (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。

11.9.2. WDTCSR – ウォッチドッグ タイマ制御レジスタ (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11-1. ウォッチドッグ タイマ設定

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

注: WDTONヒューズは0でプログラム、1で非プログラムに設定です。

● ビット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11-2. ウォッチドッグ前置分周選択

WDP3	0								1							
	0				1				0				1			
WDP2	0		1		0		1		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						

12. 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。

一般的に、

- 各割り込みベクタは2命令語を占有します。
- リセットベクタはBOOTRSTヒューズによって、割り込みベクタ開始アドレスは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 [14頁の「リセットと割り込みの扱い」](#)

12.1. ATmega16M1/32M1/64M1の割り込みベクタ

表12-1. リセットと割り込みのベクタ

ベクタ番号	プログラム アドレス (注2)	発生元	備考
1	\$0000 (注1)	リセット	電源ON, WDT, BOD等の各種リセット
2	\$0002	アナログ比較器0 ANA_COMP0	アナログ比較器0出力遷移
3	\$0004	アナログ比較器1 ANA_COMP1	アナログ比較器1出力遷移
4	\$0006	アナログ比較器2 ANA_COMP2	アナログ比較器2出力遷移
5	\$0008	アナログ比較器3 ANA_COMP3	アナログ比較器3出力遷移
6	\$000A	電力段制御器 PSC_FAULT	電力段制御器(PSC) 異常発生
7	\$000C	電力段制御器 PSC_EC	電力段制御器(PSC) 周期終了(比較一致)
8	\$000E	INT0	外部割り込み要求0
9	\$0010	INT1	外部割り込み要求1
10	\$0012	INT2	外部割り込み要求2
11	\$0014	INT3	外部割り込み要求3
12	\$0016	タイマ/カウンタ1 CAPT1	タイマ/カウンタ1捕獲発生
13	\$0018	タイマ/カウンタ1 COMP1A	タイマ/カウンタ1比較A一致
14	\$001A	タイマ/カウンタ1 COMP1B	タイマ/カウンタ1比較B一致
15	\$001C	タイマ/カウンタ1 OVF1	タイマ/カウンタ1溢れ
16	\$001E	タイマ/カウンタ0 COMP0A	タイマ/カウンタ0比較A一致
17	\$0020	タイマ/カウンタ0 COMP0B	タイマ/カウンタ0比較B一致
18	\$0022	タイマ/カウンタ0 OVF0	タイマ/カウンタ0溢れ
19	\$0024	CAN INT	CAN MOB,集中,全般異常発生
20	\$0026	CAN TOVF	CAN 時間超過
21	\$0028	LIN TC	LIN 転送完了
22	\$002A	LIN ERR	LIN 異常発生
23	\$002C	PCINT0 (PCI0)	ピン変化0群割り込み要求
24	\$002E	PCINT1 (PCI1)	ピン変化1群割り込み要求
25	\$0030	PCINT2 (PCI2)	ピン変化2群割り込み要求
26	\$0032	PCINT3 (PCI3)	ピン変化3群割り込み要求
27	\$0034	SPI STC	SPI 転送完了
28	\$0036	A/D変換器 ADC	A/D変換完了
29	\$0038	ウォッチドッグ WDT	ウォッチドッグ計時完了
30	\$003A	EEPROM EE_RDY	EEPROM 操作可
31	\$003C	SPM命令 SPM_RDY	SPM命令操作可

注1: BOOTRSTヒューズがプログラム(0)されると、デバイスはリセットでブートローダアドレスへ飛びます。「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)ビットが設定(1)されると、割り込みベクタはブートフラッシュ領域先頭(部)へ移動されます。そして各割り込みベクタのアドレスは、この表のアドレスがブートフラッシュ領域の先頭アドレスに加算されます。

右表は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12-2. リセットと割り込みベクタの配置

BOOTRST	IVSEL	リセットベクタアドレス	割り込みベクタ先頭アドレス
非プログラム(1)	0	\$0000	\$0002
	1	\$0000	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	\$0002
	1	ブート領域先頭アドレス	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27-7.で示されます。

本デバイス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 ANA_COMP0	;アナログ比較器0出力遷移
\$0004		JMP ANA_COMP1	;アナログ比較器1出力遷移
\$0006		JMP ANA_COMP2	;アナログ比較器2出力遷移
\$0008		JMP ANA_COMP3	;アナログ比較器3出力遷移
\$000A		JMP PSC_FAULT	;電力段制御器(PSC) 異常発生
\$000C		JMP PSC_EC	;電力段制御器(PSC) 周期終了(比較一致)
\$000E		JMP EXT_INT0	;外部割り込み要求0
\$0010		JMP EXT_INT1	;外部割り込み要求1
\$0012		JMP EXT_INT2	;外部割り込み要求2
\$0014		JMP EXT_INT3	;外部割り込み要求3
\$0016		JMP TIM1_CAPT	;タイマ/カウンタ1捕獲発生
\$0018		JMP TIM1_COMP_A	;タイマ/カウンタ1比較A一致
\$001A		JMP TIM1_COMP_B	;タイマ/カウンタ1比較B一致
\$001C		JMP TIM1_OVF	;タイマ/カウンタ1溢れ
\$001E		JMP TIM0_COMP_A	;タイマ/カウンタ0比較A一致
\$0020		JMP TIM0_COMP_B	;タイマ/カウンタ0比較B一致
\$0022		JMP TIM0_OVF	;タイマ/カウンタ0溢れ
\$0024		JMP CAN_INT	;CAN MOB,集中,全般異常発生
\$0026		JMP CAN_TOVF	;CAN 時間超過
\$0028		JMP LIN_TC	;LIN 転送完了
\$002A		JMP LIN_ERR	;LIN 異常発生
\$002C		JMP PCINT0	;ピン変化0群割り込み要求
\$002E		JMP PCINT1	;ピン変化1群割り込み要求
\$0030		JMP PCINT2	;ピン変化2群割り込み要求
\$0032		JMP PCINT3	;ピン変化3群割り込み要求
\$0034		JMP SPI_STC	;SPI転送完了
\$0036		JMP ADC	;A/D変換完了
\$0038		JMP WDT_OVF	;ウォッチドッグ計時完了
\$003A		JMP EE_RDY	;EEPROM操作可
\$003C		JMP SPM_RDY	;SPM命令操作可
;			
\$003E	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得
\$003F		OUT SPH, R16	;スタックポインタ上位を初期化
\$0040		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$0041		OUT SPL, R16	;スタックポインタ下位を初期化
		}	;以下、I/O初期化など

BOOTRSTヒューズが非プログラム(1)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立って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)	;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	\$1C02/\$3C02/\$7C02	;ブートプログラム領域が2Kバイトの場合
\$1C02/\$3C02/\$7C02		JMP ANA_COMP0	;アナログ比較器0出力遷移
\$1C04/\$3C04/\$7C04		JMP ANA_COMP1	;アナログ比較器1出力遷移
		}	
\$1C3C/\$3C3C/\$7C3C		JMP SPM_RDY	;SPM命令操作可

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

アドレス	ラベル	命令	注釈
	. ORG	\$0002	;割り込みベクタ先頭
\$0002		JMP ANA_COMP0	;アナログ比較器0出力遷移
\$0004		JMP ANA_COMP1	;アナログ比較器1出力遷移
		}	
\$003C		JMP SPM_RDY	;SPM命令操作可
		}	;以下、プログラムなど
	. ORG	\$1C00/\$3C00/\$7C00	;ブートプログラム領域が2Kバイトの場合
\$1C00/\$3C00/\$7C00	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/\$7C01		OUT SPH, R16	;スタック ポインタ上位を初期化
\$1C02/\$3C02/\$7C02		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$1C03/\$3C03/\$7C03		OUT SPL, R16	;スタック ポインタ下位を初期化
		}	;以下、I/O初期化など

BOOTRSTヒューズがプログラム(0)、ブート領域容量が2Kバイトに設定され、どの割り込みが許可されるのにも先立って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/\$7C00	;ブートプログラム領域が2Kバイトの場合
\$1C00/\$3C00/\$7C00		JMP RESET	;各種リセット (BOOTRSTヒューズ=0)
\$1C02/\$3C02/\$7C02		JMP ANA_COMP0	;アナログ比較器0出力遷移
\$1C04/\$3C04/\$7C04		JMP ANA_COMP1	;アナログ比較器1出力遷移
		}	
\$1C3C/\$3C3C/\$7C3C		JMP SPM_RDY	;SPM命令操作可
;			
\$1C3E/\$3C3E/\$7C3E	RESET:	LDI R16, HIGH (RAMEND)	;RAM最終アドレス上位を取得 (プログラム開始)
\$1C3F/\$3C3F/\$7C3F		OUT SPH, R16	;スタック ポインタ上位を初期化
\$1C40/\$3C40/\$7C40		LDI R16, LOW (RAMEND)	;RAM最終アドレス下位を取得
\$1C41/\$3C41/\$7C41		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

12.2. 割り込みベクタ移動関係レジスタ

12.2.1. 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。

12.2.2. 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
	SPIPS	-	-	PUD	-	-	IVSEL	IVCE
アクセス種別	R/W	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タ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ヒューズ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タ領域に配置され、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、ブートロータ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 */
}
```

13. 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。

PCI3ピン変化割り込みは許可したPCINT24~26の何れかが切り替わると起動します。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PCINT8~15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, PCMSK3)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を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を起動するのに使えることを意味します。I/Oクロックはアイドル動作を除く全休止形態で停止されます。

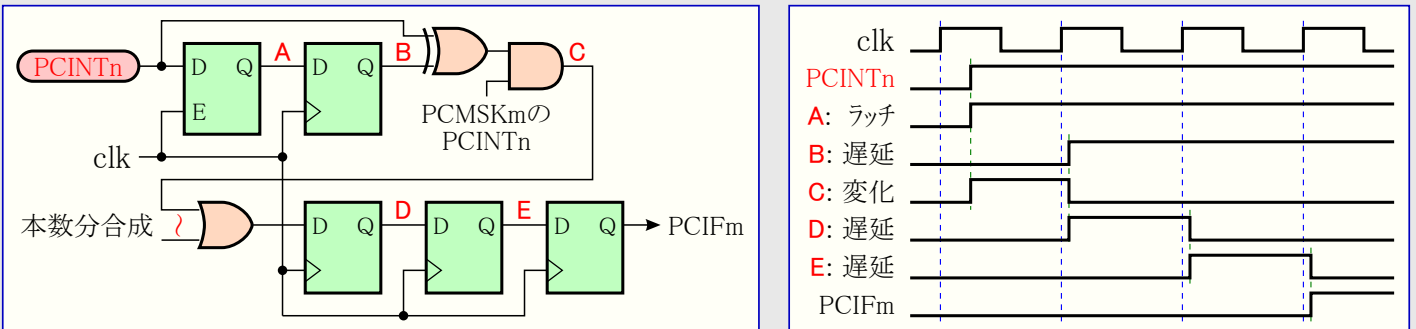
注: レベル起動割り込みがパワーダウン動作からの起動に使われる場合、この必要としたレベルはレベル割り込みを起動する完全な起動復帰のため、MCUに対して充分長く保持されなければならないことに注意してください。このレベルが起動時間の最後に先立って消滅すると、MCUは今まで通り起動しますが、割り込みが生成されません。起動時間はSUTとCKSELのヒューズで定義されます。

関連リンク 27頁の「システムクロックとクロック選択」

13.1. 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。

図13-1. ピン変化割り込みタイミング



13.2. 外部割り込み用レジスタ

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

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
	ISC31,0		ISC21,0		ISC11,0		ISC01,0	
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7,6 – ISC31,0 : 外部割り込み3条件制御 (Interrupt Sense Control 3)

外部割り込み3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3許可(INT3)ビットが設定(1)される場合のINT3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INT3ピンの端(エッジ)とレベルは右表で定義されます。INT3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13-1. 外部割り込み3(INT3)割り込み条件

ISC31	ISC30	割り込み発生条件
0	0	INT3ピンがLowレベルで発生。
0	1	INT3ピンの論理変化(両端)。
1	0	INT3ピンの下降端で発生。
1	1	INT3ピンの上昇端で発生。

● ビット5,4 – ISC21,0 : 外部割り込み2条件制御 (Interrupt Sense Control 2)

外部割り込み2はステータスレジスタ(SREG)の全割り込み許可(I)ビットと外部割り込み許可レジスタ(EIMSK)の外部割り込み2許可(INT2)ビットが設定(1)される場合のINT2外部ピンによって活性(有効)にされます。割り込みを活性にする外部INT2ピンの端(エッジ)とレベルは右表で定義されます。INT2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13-2. 外部割り込み2(INT2)割り込み条件

ISC21	ISC20	割り込み発生条件
0	0	INT2ピンがLowレベルで発生。
0	1	INT2ピンの論理変化(両端)。
1	0	INT2ピンの下降端で発生。
1	1	INT2ピンの上昇端で発生。

● 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13-3. 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13-4. 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。

13.2.2. 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
	-	-	-	-	INT3	INT2	INT1	INT0
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 – INT3 : 外部割り込み3許可 (External Interrupt Request 3 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、INT3ビットが設定(1)されると、INT3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3のビット1と0(ISC31,0)は、この外部割り込みがINT3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INT3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3に対応する割り込みはINT3割り込みベクタから実行されます。

● ビット2 – INT2 : 外部割り込み2許可 (External Interrupt Request 2 Enable)

ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)され、INT2ビットが設定(1)されると、INT2外部ピン割り込みが許可されます。外部割り込み制御レジスタA(EICRA)の割り込み条件制御2のビット1と0(ISC21,0)は、この外部割り込みがINT2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INT2ピンが出力として設定されても、このピンの動きは割り込み要求を引き起こします。外部割り込み要求2に対応する割り込みはINT2割り込みベクタから実行されます。

● 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。

13.2.3. 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
	-	-	-	-	INTF3	INTF2	INTF1	INTF0
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 – INTF3 : 外部割り込み3要求フラグ (External Interrupt Flag 3)

INT3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、INTF3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3許可(INT3)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。INT3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2 – INTF2 : 外部割り込み2要求フラグ (External Interrupt Flag 2)

INT2ピン上の端(エッジ)または論理変化が割り込み要求を起動すると、INTF2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2許可(INT2)ビットが設定(1)なら、MCUは対応する割り込みベクタへ飛びます。このフラグは割り込みルーチンが実行されると解除(0)されます。代わりにこのフラグは1を書くことによっても解除(0)できます。INT2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。

13.2.4. 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
	-	-	-	-	PCIE3	PCIE2	PCIE1	PCIE0
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 – PCIE3 : ピン変化3群割り込み許可** (Pin Change Interrupt Enable 3)

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、PCIE3ビットが設定(1)されると、ピン変化3群割り込みが許可されます。許可したPCINT24~26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PCI3割り込みベクタから実行されます。PCINT24~26ピンはピン変化割り込み許可レジスタ3(PCMSK3)によって個別に許可されます。

● **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PCINT8~15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。PCINT8~15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。

13.2.5. 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
	-	-	-	-	PCIF3	PCIF2	PCIF1	PCIF0
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 – PCIF3 : ピン変化3群割り込み要求フラグ** (Pin Change Interrupt Flag 3)

PCINT24~26ピンの何れかの論理変化が割り込み要求を起動すると、PCIF3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3群割り込み許可(PCIE3)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)

PCINT8～15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。

13.2.6. PCMSK3 – ピン変化割り込み許可レジスタ3 (Pin Change Mask Register 3)

名称 : PCMSK3

変位 : \$6D

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	PCINT26	PCINT25	PCINT24
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2～0 – PCINT26～PCINT24 : ピン変化割り込み26～24許可** (Pin Change Enable Mask 26～24)

各PCINT24～26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。PCINT24～26とピン変化割り込み制御レジスタ(PCICR)の**PCIE3**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。PCINT24～26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。

13.2.7. 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 : \$6C

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	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 – 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。

13.2.8. 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 : \$6B
リセット : \$00
特質 : -

ビット	7	6	5	4	3	2	1	0
	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
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 – PCINT15~PCINT8 : ピン変化割り込み15~8許可 (Pin Change Enable Mask 15~8)

各PCINT8~15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。PCINT8~15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。PCINT8~15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。

13.2.9. 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 : \$6A
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	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 – 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。

14. 入出力ポート

14.1. 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向をも無意識に変更することなく、1つのポートピンの方向を変更できることを意味します。(出力として設定されていれば)駆動値を変更、または(入力として設定されていれば)プルアップ抵抗を許可/禁止する時にも同じく適用されます。各出力緩衝部は高い吐き出し(ソース)と吸い込み(シンク)の両能力で対称的な駆動特性を持ちます。このピン駆動部はLED(表示器)を直接駆動するのに十分な強さです。全てのポートピン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。

14.2. 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14-1. 入出力ピン等価回路

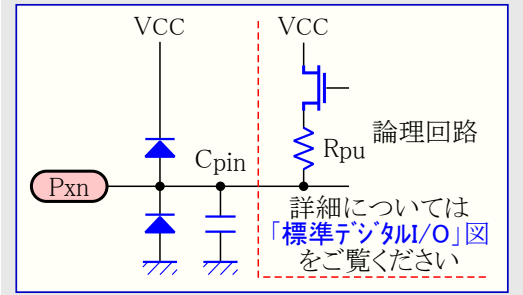
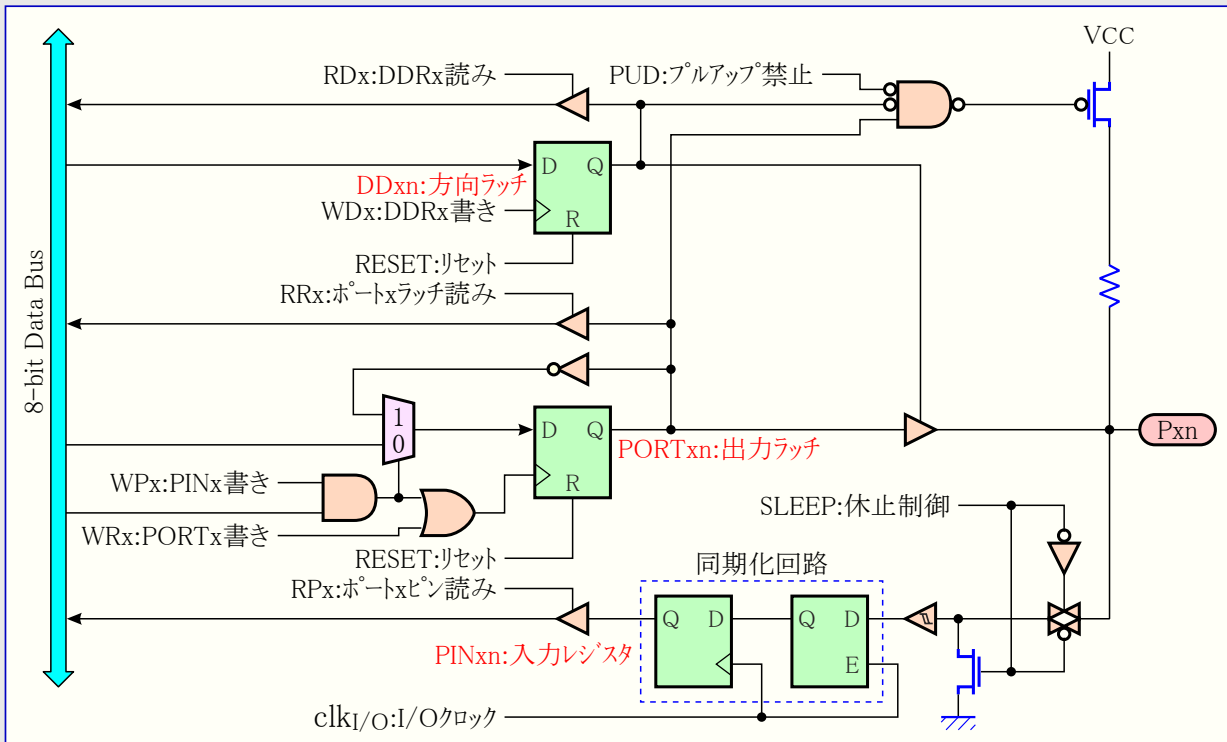


図14-2. 標準デジタル入出力回路構成



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

14.2.1. ピンの設定

各ポートピンは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。

14.2.2. 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。

14.2.3. 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

14.2.4. 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14-3. 外部供給ピン値読み込み時の同期化

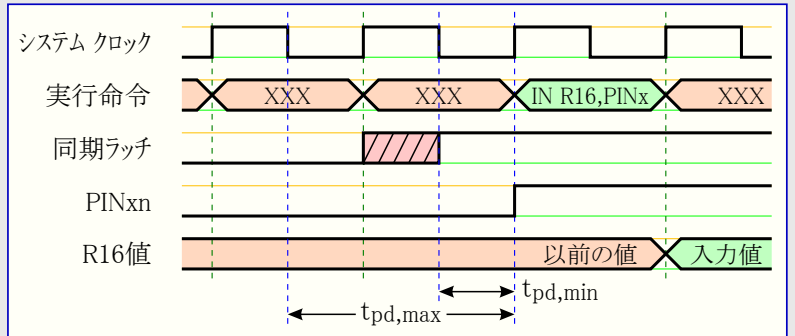
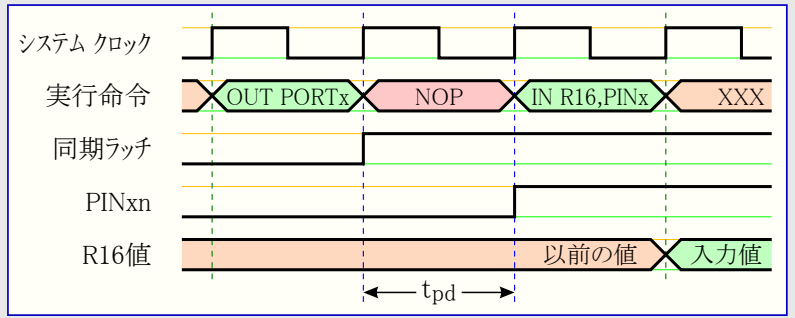


図14-4. プログラムで設定したピン値読み戻し時の同期化



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。

14.2.5. デジタル入力許可と休止形態

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

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

14.2.6. 未接続ピン

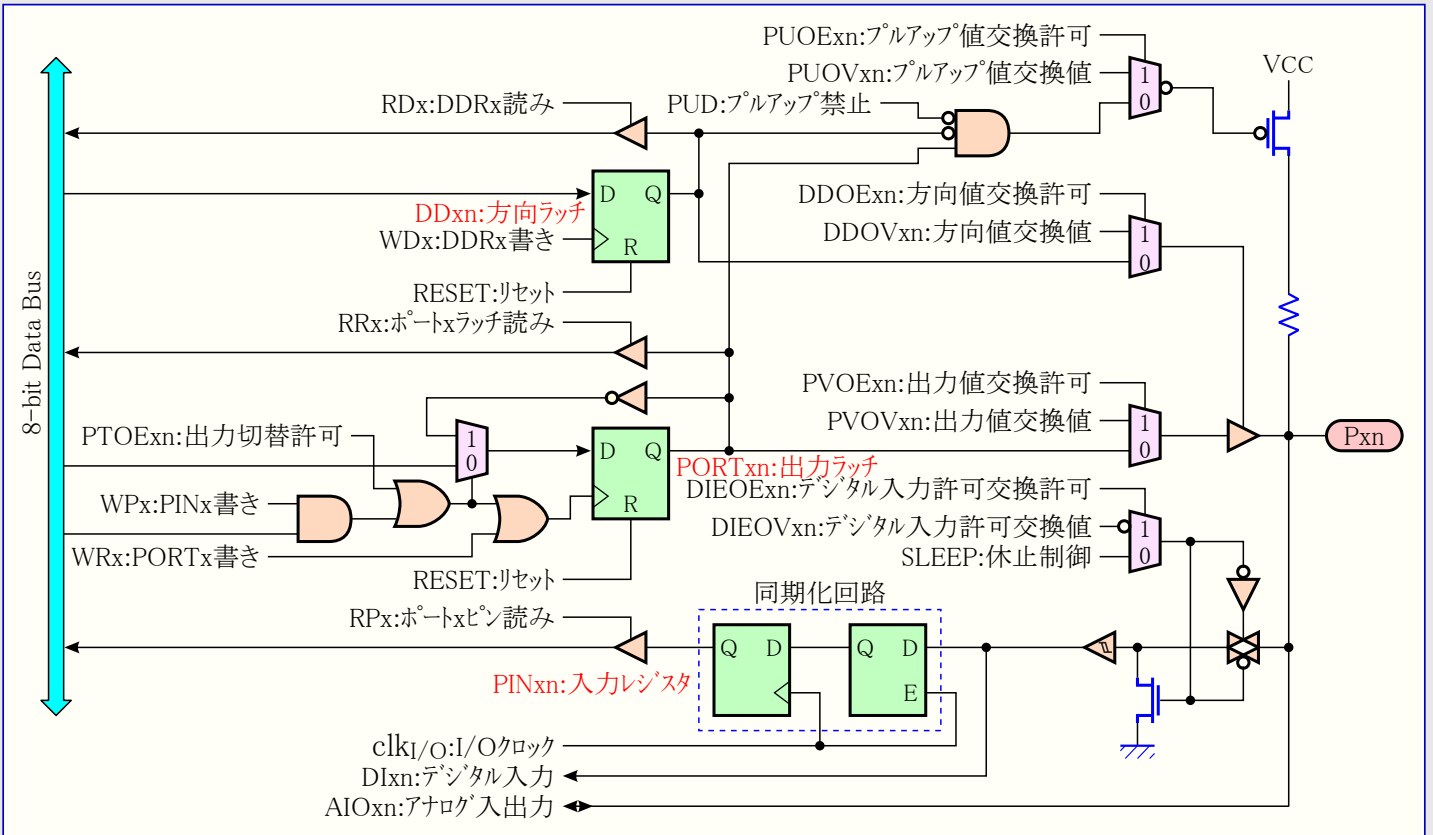
いくつかのピンが未使用にされる場合、それらのピンが定義されたレベルを持つのを保証することが推奨されます。例え上記のような深い休止形態で多くのデジタル入力が禁止されるとしても、デジタル入力が許可される他の全ての動作(リセット、活動動作、アイドル動作)で消費電流削減のため、浮き状態入力は避けられるべきです。

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。

14.3. 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14-2.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14-5. 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14-2. 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。

14.3.1. ポートBの交換機能

交換機能を持つポートBピンは下表で示されます。

表14-3. ポートBピンの交換機能

ポートピン	交換機能	ポートピン	交換機能
PB7	SCK (SPI 直列クロック 主側出力/従側入力)	PB3	AMP0- (アナログ差動増幅器0 反転入力)
	ADC4 (A/D変換アナログ入力チャネル4)		PCINT3 (ピン変化割り込み3入力)
PB6	PSCOUT0B (電力段制御器0 B出力)	PB2	ADC5 (A/D変換アナログ入力チャネル5)
	PCINT7 (ピン変化割り込み7入力)		INT1 (外部割り込み1入力)
	ADC7 (A/D変換アナログ入力チャネル7)		ACMPN0 (アナログ比較器0 反転入力)
PB5	PSCOUT1B (電力段制御器1 B出力)	PB1	PCINT2 (ピン変化割り込み2入力)
	PCINT6 (ピン変化割り込み6入力)		MOSI (SPI 主側データ出力/従側データ入力)
	ADC6 (A/D変換アナログ入力チャネル6)		PSCOUT2B (電力段制御器2 B出力)
	INT2 (外部割り込み2入力)		PCINT1 (ピン変化割り込み1入力)
PB4	ACMPN1 (アナログ比較器1 反転入力)	PB0	MISO (SPI 主側データ入力/従側データ出力)
	AMP2- (アナログ差動増幅器2 反転入力)		PSCOUT2A (電力段制御器2 A出力)
	PCINT5 (ピン変化割り込み5入力)		PCINT0 (ピン変化割り込み0入力)
	AMP0+ (アナログ差動増幅器0 非反転入力)		
	PCINT4 (ピン変化割り込み4入力)		

交換ピンの設定は次のとおりです。

- SCK/ADC4/PSCOUT0B/PCINT7 – ポートB ビット7 : PB7
 - SCK : SPI用の主装置クロック出力、従装置クロック入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB7設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB7によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB7によって制御できます。
 - ADC4 : A/D変換チャネル4入力。
 - PSCOUT0B : 電力段制御器(PSC)0のB出力。
 - 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扱えます。
- ICP1B/ADC7/PSCOUT1B/PCINT6 – ポートB ビット6 : PB6
 - ADC7 : A/D変換チャネル7入力。
 - PSCOUT1B : 電力段制御器(PSC)1のB出力。
 - 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扱えます。
- ADC6/INT2/ACMPN1/AMP2-/PCINT5 – ポートB ビット5 : PB5
 - ADC6 : A/D変換チャネル6入力。
 - INT2 : 外部割り込み2入力。本ピンはMCUへの外部割り込み元として扱えます。
 - ACMPN1 : アナログ比較器1反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - AMP2- : A/D変換器用アナログ差動増幅器2反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- AMP0+/PCINT4 – ポートB ビット4 : PB4
 - AMP0+ : A/D変換器用アナログ差動増幅器0非反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- AMP0-/PCINT3 – ポートB ビット3 : PB3
 - AMP0- : A/D変換器用アナログ差動増幅器0反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- ADC5/INT1/ACMPN0/PCINT2 – ポートB ビット2 : PB2
 - ADC5 : A/D変換チャネル5入力。
 - INT1 : 外部割り込み1入力。本ピンはMCUへの外部割り込み元として扱えます。

- ACMPN0: アナログ比較器0反転入力。アナログ比較器機能を邪魔するデジタルポート機能を避けるため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
- 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扱えます。
- MOSI/PSCOUT2B/PCINT1 - ポートBビット1: PB1
 - MOSI: SPI用の主装置データ出力、従装置データ入力。SPIが従装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB1設定に拘らず、入力として設定されます。SPIが主装置として許可されると、このピンのデータ方向はDDB1によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB1とPUDビットによって制御できます。
 - PSCOUT2B: 電力段制御器(PSC)2のB出力。
 - 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扱えます。
- MISO/PSCOUT2A/PCINT0 - ポートBビット0: PB0
 - MISO: SPI用の主装置データ入力、従装置データ出力。SPIが主装置として許可されると、このピンはポートB方向レジスタ(DDRB)のDDB0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DDB0によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だポートB出力レジスタ(PORTB)のPORTB0とPUDビットによって制御できます。
 - PSCOUT2A: 電力段制御器(PSC)2のA出力。
 - 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扱えます。

下表は下表ポートBの交換機能を図14-5.で示される交換信号に関連付けます。

表14-4. ポートB7~4の交換機能用交換信号

信号名	PB7/SCK/ADC4/ PSCOUT0B/PCINT7	PB6/ADC7/ PSCOUT1B/PCINT6	PB5/ADC6/INT2/ ACMPN1/AMP2-/PCINT5	PB4/AMP0+/PCINT4
PUOE	SPE・MSTR・SPIPS	0	0	0
PUOV	PB7・PUD・SPIPS	0	0	0
DDOE	SPE・MSTR・SPIPS +PSCen01	PSCen11	0	0
DDOV	PSCen01	1	0	0
PVOE	SPE・MSTR・SPIPS	PSCen11	0	0
PVOV	PSCout01・SPIPS +PSCout01・PSCen01 ・SPIPS	PSCOUT11	0	0
PTOE	-	-	-	-
DIEOE	ADC4D	ADC7D	ADC6D+INT2許可	AMP0ND
DIEOV	0	0	INT2許可	0
DI	内部リセット・SPIPS・SCK入力	ICP1B入力	INT2入力	-
AIO	ADC4	ADC7	ADC6	AMP0+

表14-5. ポートB3~0の交換機能用交換信号

信号名	PB3/AMP0-/PCINT3	PB2/ADC5/INT1/ ACMPN0/PCINT2	PB1/MOSI/PSCOUT2B/ PCINT1	PB0/MISO/PSCOUT2A/ PCINT0
PUOE	0	0	-	-
PUOV	0	0	-	-
DDOE	0	0	-	-
DDOV	0	0	-	-
PVOE	0	0	-	-
PVOV	0	0	-	-
PTOE	-	-	-	-
DIEOE	AMP0ND	ADC5D+INT1許可	0	0
DIEOV	0	INT1許可	0	0
DI	-	INT1入力	内部リセット・SPIPS・MOSI入力	内部リセット・SPIPS・MOSI入力
AIO	AMP0-	ADC5	-	-

14.3.2. ホートCの交換機能

交換機能を持つポートピンは下表で示されます。

表14-6. ホート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
PC7	D2A (D/A変換器出力) AMP2+ (アナログ差動増幅器2 非反転入力) PCINT15 (ピン変化割り込み15入力)	PC3	T1 (タイマ/カウンタ1 外部クロック入力) RXCAN (CAN受信データ入力) ICP1B (タイマ/カウンタ1 捕獲起動代替入力) PCINT11 (ピン変化割り込み11入力)
PC6	ADC10 (A/D変換アナログ入力チャネル10) ACMP1 (アナログ比較器1 非反転入力) PCINT14 (ピン変化割り込み14入力)	PC2	T0 (タイマ/カウンタ0 外部クロック入力) TXCAN (CAN送信データ出力) PCINT10 (ピン変化割り込み10入力)
PC5	ADC9 (A/D変換アナログ入力チャネル9) AMP1+ (アナログ差動増幅器1 非反転入力) ACMP3 (アナログ比較器3 非反転入力) PCINT13 (ピン変化割り込み13入力)	PC1	OC1B (タイマ/カウンタ1 比較B一致/PWM出力) PSCIN1 (電力段制御器1 デジタル入力) SS_A (SPI 代替従装置選択入力) PCINT9 (ピン変化割り込み9入力)
PC4	ADC8 (A/D変換アナログ入力チャネル8) AMP1- (アナログ差動増幅器1 反転入力) ACMPN3 (アナログ比較器3 反転入力) (注) PCINT12 (ピン変化割り込み12入力)	PC0	INT3 (外部割り込み3入力) PSCOUT1A (電力段制御器1 A出力) PCINT8 (ピン変化割り込み8入力)

注: 技術試供品(AT90PWM324と記されたデバイス)ではACMPN3交換機能がPC4ではなくPE2に配置されています。

交換ピンの設定は次のとおりです。

- D2A/AMP2+/PCINT15 – ホートC ビット7 : PC7
 - D2A : D/A変換器の出力。
 - AMP2+ : A/D変換器用アナログ差動増幅器2非反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - PCINT15 : ピン変化割り込み15入力。PC7ピンは外部割り込み元として扱えます。
- ADC10/ACMP1/PCINT14 – ホートC ビット6 : PC6
 - ADC10 : A/D変換チャネル10入力。
 - ACMP1 : アナログ比較器1 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- ADC9/AMP1+/ACMP3/PCINT13 – ホートC ビット5 : PC5
 - ADC9 : A/D変換チャネル9入力。
 - AMP1+ : A/D変換器用アナログ差動増幅器1非反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - ACMP3 : アナログ比較器3 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- ADC8/AMP1-/ACMPN3/PCINT12 – ホートC ビット4 : PC4
 - ADC8 : A/D変換チャネル8入力。
 - AMP1- : A/D変換器用アナログ差動増幅器1反転入力。アナログ増幅器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - ACMPN3 : アナログ比較器3 反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- T1/RXCAN/ICP1B/PCINT11 – ホートC ビット3 : PC3
 - T1 : タイマ/カウンタ1の外部クロック入力ピンです。
 - RXCAN : CAN受信データ入力ピン。
 - ICP1B : タイマ/カウンタ1の捕獲起動代替入力。PC3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。
 - 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扱えます。

- T0/TXCAN/PCINT10 – ホートC ビット2 : PC2
 - T0 : タイマ/カウンタ0の外部クロック入力ピンです。
 - TXCAN : CAN送信データ出力ピン。
 - 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扱えます。
- OC1B/PSCIN1/ $\overline{SS_A}$ /PCINT9 – ホートC ビット1 : PC1
 - OC1B : タイマ/カウンタ1の比較B一致出力。本ピンはタイマ/カウンタ1の比較B一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(DDC1=1)されなければなりません。このOC1Bピンはタイマ機能のPWM動作用出力ピンでもあります。
 - PSCIN1 : 電力段制御器1のデジタル入力。
 - $\overline{SS_A}$: SPI 代替従装置選択入力。SPIが従装置として許可されると、このピンはDDC1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DDC1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PORTC1によって制御できます。
 - 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扱えます。
- INT3/PSCOUT1A/PCINT8 – ホートC ビット0 : PC0
 - INT3 : 外部割り込み3入力。本ピンはMCUへの外部割り込み元として扱えます。
 - PSCOUT1A : 電力段制御器(PSC)1のA出力。
 - 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扱えます。

下表はホートCの交換機能を図14-5.で示される交換信号に関連付けます。

表14-7. ホートC7~4の交換機能用交換信号

信号名	PC7/D2A/AMP2+/ PCINT15	PC6/ADC10/ACMP1/ PCINT14	PC5/ADC9/AMP1+/ ACMP3/PCINT13	PC4/ADC8/AMP1-/ ACMPN3/PCINT12
PUOE	0	0	0	-
PUOV	0	0	0	-
DDOE	-	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	-
PVOV	0	0	0	-
PTOE	-	-	-	-
DIOE	DAEN	ADC10D	ADC9D	ADC8D
DIOV	0	0	0	0
DI	-	-	-	-
AIO	-	ADC10amp1	ADC9amp1+/ACMP3	ADC8amp1-/ACMPN3

表14-8. ホートC3~0の交換機能用交換信号

信号名	PC3/T1/RXCAN/ICP1B/ PCINT11	PC2/T0/TXCAN/PCINT10	PC1/OC1B/PSCIN1/ $\overline{SS_A}$ /PCINT9	PC0/INT3/PSCOUT1A/ PCINT8
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	PSCen23	PSCen22	0	PSCen10
DDOV	1	1	0	1
PVOE	PSCen23	PSCen22	OC1B許可	PSCen10
PVOV	PSCout23	PSCout22	OC1B	PSCout10
PTOE	-	-	-	-
DIOE	-	-	-	INT3許可
DIOV	-	-	-	INT3許可
DI	T1入力	T0入力	PSCin1/ $\overline{SS_A}$	INT3入力
AIO	-	-	-	-

14.3.3. ポートDの交換機能

交換機能を持つポートDピンは下表で示されます。

表14-9. ポートD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
PD7	ACMP0 (アナログ比較器0 非反転入力) PCINT23 (ピン変化割り込み23入力)	PD3	TXD/TXLIN (LIN/UART 送信データ出力) OC0A (タイマ/カウンタ0 比較A一致/PWM出力) SS (SPI 従装置選択入力) MOSLA (SPI 代替データ 主側出力/従側入力) PCINT19 (ピン変化割り込み19入力)
PD6	INT0 (外部割り込み0入力) ADC3 (A/D変換アナログ入力チャネル3) ACMPN2 (アナログ比較器2 反転入力) PCINT22 (ピン変化割り込み22入力)		PD2
PD5	ADC2 (A/D変換アナログ入力チャネル2) ACMP2 (アナログ比較器2 非反転入力) PCINT21 (ピン変化割り込み21入力)	PD1	
PD4	RXD/RXLIN (LIN/UART 受信データ入力) ICP1A (タイマ/カウンタ1 捕獲起動入力) SCK_A (SPI 代替直列クロック入出力) ADC1 (A/D変換アナログ入力チャネル1) PCINT20 (ピン変化割り込み20入力)	PD0	PSCOUT0A (電力段制御器0 A出力) 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。

- ACMP0/PCINT23 – ポートD ビット7 : PD7
 - ACMP0 : アナログ比較器0 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- INT0/ADC3/ACMPN2/PCINT22 – ポートD ビット6 : PD6
 - INT0 : 外部割り込み0入力。本ピンはMCUへの外部割り込み元として扱えます。
 - ADC3 : A/D変換チャネル3入力。
 - ACMPN2 : アナログ比較器2 反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- ADC2/ACMP2/PCINT21 – ポートD ビット5 : PD5
 - ADC2 : A/D変換チャネル2入力。
 - ACMP2 : アナログ比較器2 非反転入力。アナログ比較器機能を邪魔するデジタルポート機能为了避免のため、内部プルアップがOFFにされた入力としてポートピンを設定してください。
 - 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扱えます。
- RXD/RXLIN/ICP1A/SCK_A/ADC1/PCINT20 – ポートD ビット4 : PD4
 - RXD/RXLIN : LIN/UART受信データ(LIN/UART用データ入力ピン)。LIN/UART受信部が許可されると、このピンはDDRDのDDD4の値に拘らず、入力として設定されます。UARTがこのピンを入力に強制するとき、プルアップは未だPORTD4ビットによって制御できます。
 - ICP1A : タイマ/カウンタ1の捕獲起動入力。本ピンはタイマ/カウンタ1用捕獲起動入力ピンとして動作できます。
 - SCK_A : SPI用の代替主装置クロック出力、従装置クロック入力。SPIが従装置として許可されると、このピンはDDD4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DDD4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PORTD4によって制御できます。
 - ADC1 : A/D変換チャネル1入力。
 - 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扱えます。

- TXD/TXLIN/OC0A/ \overline{SS} /MOSI_A/PCINT19 – ポートD ビット3 : PD3

- TXD/TXLIN : LIN/UART送信データ(LIN/UART用データ出力ピン)。LIN/UART送信部が許可されると、ピンはDDD3値に拘らず、出力として設定されます。
- OC0A : タイマ/カウンタ0の比較A一致出力。本ピンはタイマ/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、この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OC0Aピンはタイマ機能のPWM動作用出力ピンでもあります。
- \overline{SS} : SPI従装置選択入力。SPIが従装置として許可されると、このピンはDDD3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DDD3によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だPORTD3によって制御できません。
- MOSI_A : SPI用の代替主装置データ出力、従装置データ入力。SPIが従装置として許可されると、このピンはDDD3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DDD3によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だPORTD3によって制御できません。
- 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扱えます。

- OC1A/MISO_A/PSCIN2/PCINT18 – ポートD ビット2 : PD2

- OC1A : タイマ/カウンタ1の比較A一致出力。本ピンはタイマ/カウンタ1の比較A一致用外部出力として扱えます。この機能を扱うため、このピンは出力として設定(DDD2=1)されなければなりません。このOC1Aピンはタイマ機能のPWM動作用出力ピンでもあります。
- MISO_A : SPI用の代替主装置データ入力、従装置データ出力。SPIが主装置として許可されると、このピンはDDD2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DDD2によって制御されます。このピンがSPIによって入力を強制される時、プルアップは未だPORTD2によって制御できません。
- PSCIN2 : 電力段制御器2のデジタル入力。
- 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扱えます。

- CLKO/PSCIN0/PCINT17 – ポートD ビット1 : PD1

- CLKO : システムクロック出力。分周したシステムクロックがこのピンに出力できます。分周したシステムクロックはCKOUTヒューズがプログラム(0)されると、PORTD1とDDD1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。
- PSCIN0 : 電力段制御器0のデジタル入力。
- 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扱えます。

- PSCOUT0A/PCINT16 – ポートD ビット0 : PD0

- PSCOUT0A : 電力段制御器(PSC)0のA出力。
- 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扱えます。

下表はポートDの交換機能を図14-5.で示される交換信号に関連付けます。

表14-10. ポートD7~4の交換機能用交換信号

信号名	PD7/ACMP0/PCINT23	PD6/INT0/ADC3/ ACMPN2/PCINT22	PD5/ADC2/ACMP2/ PCINT21	PD4/RXD/RXLIN/ICP1A/ SCK_A/ADC1/PCINT20
PUOE	0	0	0	RXEN+SPE· $\overline{\text{MSTR}}$ ·SPIPS
PUOV	0	0	0	PD4·PUD
DDOE	0	0	0	RXEN+SPE· $\overline{\text{MSTR}}$ ·SPIPS
DDOV	0	0	0	0
PVOE	0	0	0	SPE· $\overline{\text{MSTR}}$ ·SPIPS
PVOV	0	0	0	-
PTOE	-	-	-	-
DIEOE	ACMP0D	ADC3D+INT0許可	ADC2D	ADC1D
DIEOV	0	-	0	0
DI	-	INT0入力	-	ICP1A入力
AIO	ACMP0D	ADC3/ACMPM	ADC2/ACOMP2	ADC1

表14-11. ポートD3~0の交換機能用交換信号

信号名	PD3/TXD/TXLIN/OC0A/ SS/MISO_A/PCINT19	PD2/OC1A/MISO_A/ PSCIN2/PCINT18	PD1/CLKO/PSCIN0/ PCINT17	PD0/PSCOUT0A/PCINT16
PUOE	TXEN+SPE· $\overline{\text{MSTR}}$ ·SPIPS	-	0	SPE· $\overline{\text{MSTR}}$ ·SPIPS
PUOV	$\overline{\text{TXEN}}$ +SPE· $\overline{\text{MSTR}}$ ·SPIPS ·PD3·PUD	-	0	PD0·PUD
DDOE	TXEN+SPE· $\overline{\text{MSTR}}$ ·SPIPS	-	0	PSCen00+SPE· $\overline{\text{MSTR}}$ ·SPIPS
DDOV	TXEN	0	0	PSCen00
PVOE	TXEN+OC0en+ SPE· $\overline{\text{MSTR}}$ ·SPIPS	-	0	PSCen00+UMSEL
PVOV	TXEN·TXD+TXEN·(OC0en· OC0+OC0en·SPIPS·MOSI)	-	0	-
PTOE	-	-	-	-
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	SS/MOSLA入力	-	-	SS_A入力
AIO	-	-	-	-

14.3.4. ポートEの交換機能

交換機能を持つポートピンは下表で示されます。

表14-12. ポートEピンの交換機能

ポートピン	交換機能
PE2	XTAL2 (主クロック用発振増幅器出力) ADC0 (A/D変換アナログ入力チャネル0) PCINT26 (ピン変化割り込み26入力)
PE1	XTAL1 (主クロック用発振増幅器入力) OC0B (タイマ/カウンタ0 比較B一致/PWM出力) PCINT25 (ピン変化割り込み25入力)
PE0	$\overline{\text{RESET}}$ (外部リセット入力) OCD (内蔵デバッグ機能用入出力) PCINT24 (ピン変化割り込み24入力)

注: 技術試供品(AT90PWM324と記されたデバイス)ではACMPN3交換機能がPC4ではなくPE2に配置されています。

交換ピンの設定は次のとおりです。

• XTAL2/ADC0/PCINT26 – ポートE ビット2 : PE2

- XTAL2 : 主クロック用発振増幅器出力。水晶用低電力発振器用のクロックピンとして使われます。クロックピンとして使われると、本ピンはI/Oピンとして使えません。
- ADC0 : A/D変換チャネル0入力。
- PCINT26 : ピン変化割り込み26入力。PE2ピンは外部割り込み元として扱えます。

• XTAL1/OC0B/PCINT25 – ポートE ビット1 : PE1

- XTAL1 : 主クロック用発振増幅器入力。校正付き内蔵RC発振器を除く全てのチップクロック元用に使われます。クロックピンとして使われると、本ピンはI/Oピンとして使えません。
- OC0B : タイマ/カウンタ0の比較B一致出力。本ピンはタイマ/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、このピンは出力として設定(DDE1=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動作出力ピンでもあります。
- PCINT25 : ピン変化割り込み25入力。PE1ピンは外部割り込み元として扱えます。

• $\overline{\text{RESET}}$ /OCD/PCINT24 – ポートE ビット0 : PE0

- $\overline{\text{RESET}}$: 外部リセット入力。RSTDISBLヒューズがプログラム(0)されると、本ピンは標準の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、本ピンにリセット回路が接続され、本ピンはI/Oピンとして使えません。
PE0がリセットピンとして使われると、PORTE0, DDE0, PINE0は全て0を読みます。
- PCINT24 : ピン変化割り込み24入力。PE0ピンは外部割り込み元として扱えます。

下表はポートEの交換機能を図14-5.で示される交換信号に関連付けます。

表14-13. ポートE2~0の交換機能用交換信号

信号名	PE2/XTAL2/ADC0/PCINT26	PE1/XTAL1/OC0B/PCINT25	PE0/ $\overline{\text{RESET}}$ /OCD/PCINT24
PUOE	0	0	0
PUOV	0	0	0
DDOE	0	0	0
DDOV	0	0	0
PVOE	0	OC0B許可	0
PVOV	0	OC0B	0
PTOE	-	-	0
DIEOE	ADC0D	0	0
DIEOV	0	0	0
DI	-	-	-
AIO	クロック発振器出力/ADC0	クロック発振器入力/クロック信号入力	-

14.4. I/Oポート用レジスタ

14.4.1. 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
	SPIPS	–	–	PUD	–	–	IVSEL	IVCE
アクセス種別	R/W	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。

14.4.2. 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	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 – PORTB7～0 : ポートB出力 (Port B Data)

14.4.3. 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	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 – DDB7～0 : ポートBデータ方向 (Port B Data Direction)

14.4.4. 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	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 – 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。「[ピン出力交互切り替え](#)」をご覧ください。

14.4.5. 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
	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
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 – PORTC6～0 : ポートC出力 (Port C Data)

14.4.6. 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
	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
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 – DDC6～0 : ポートCデータ方向 (Port C Data Direction)

14.4.7. 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
	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
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 – 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。「[ピン出力交互切り替え](#)」をご覧ください。

14.4.8. PORTD – ポート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	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 – PORTD7～0 : ポートD出力 (Port D Data)

14.4.9. DDRD – ポート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	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 – DDD7～0 : ポートDデータ方向 (Port D Data Direction)

14.4.10. PIND – ポート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	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 – PIND7～0 : ポート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。

14.4.11. PORTE – ポートE出力レジスタ (Port E 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 : PORTE

変位 : \$2E (\$0E)

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\$0Eです。

ビット	7	6	5	4	3	2	1	0
	–	–	–	–	–	PORTE2	PORTE1	PORTE0
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2～0 – PORTE2～0 : ポートE出力 (Port E Data)

14.4.12. DDRE – ポートE方向レジスタ (Port E 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 : DDRE

変位 : \$2D (\$0D)

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\$0Dです。

ビット	7	6	5	4	3	2	1	0
	–	–	–	–	–	DDE2	DDE1	DDE0
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2～0 – DDE2～0 : ポートEデータ方向 (Port E Data Direction)

14.4.13. PINE – ポートE入力レジスタ (Port E 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 : PINE

変位 : \$2C (\$0C)

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\$0Cです。

ビット	7	6	5	4	3	2	1	0
	–	–	–	–	–	PINE2	PINE1	PINE0
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	不定	不定	不定

- ビット2～0 – PINE2～0 : ポートE入力 (Port E 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。

15. TCO – 8ビット タイマ/カウンタ0 (PWM付き)

関連リンク 99頁の「タイマ/カウンタ0と1の前置分周器」

15.1. 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 (TOV, OCFnA, OCFB)

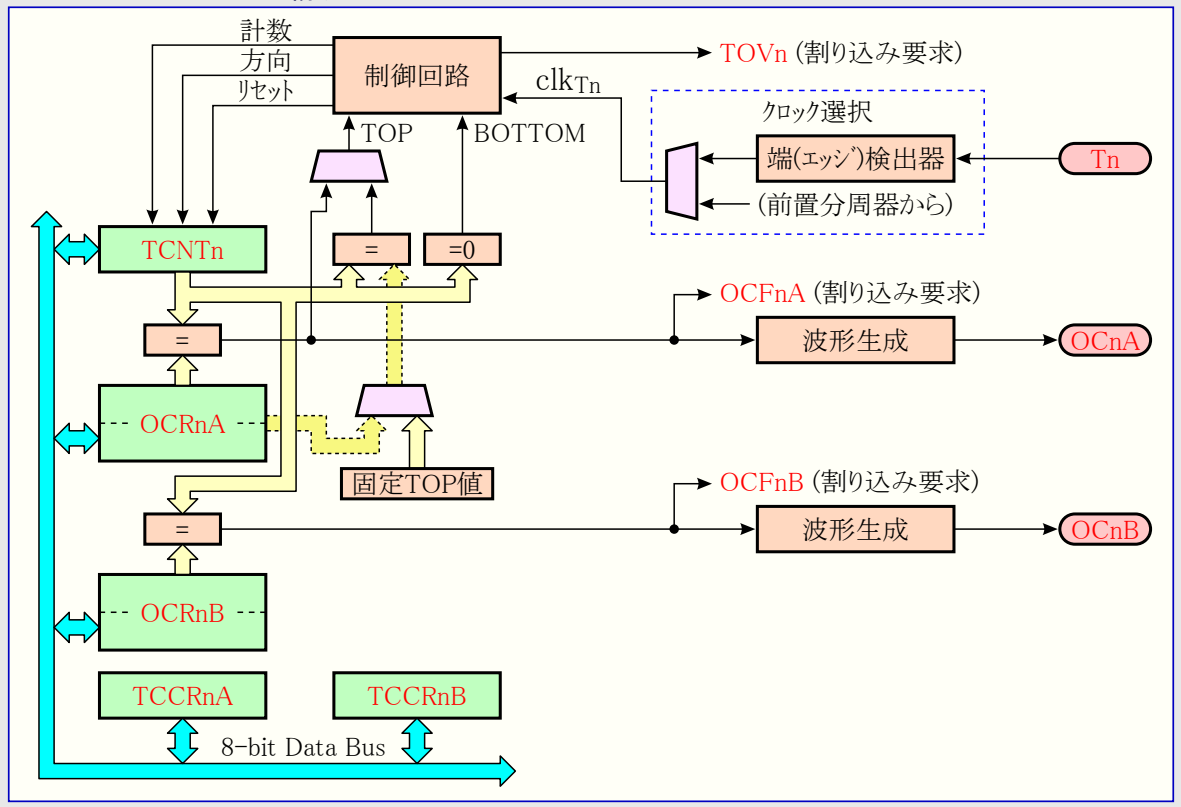
15.2. 概要

タイマ/カウンタ0(TCO)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のPRTIM0(PRR.PRTIM0)ビットが0を書かれる時に許可されます。

図15-1. 8ビットタイマ/カウンタ構成図



15.2.1. 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15-1. 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。

15.2.2. 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(clk_{T0})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(OCFAとOCFB)も設定(1)します。

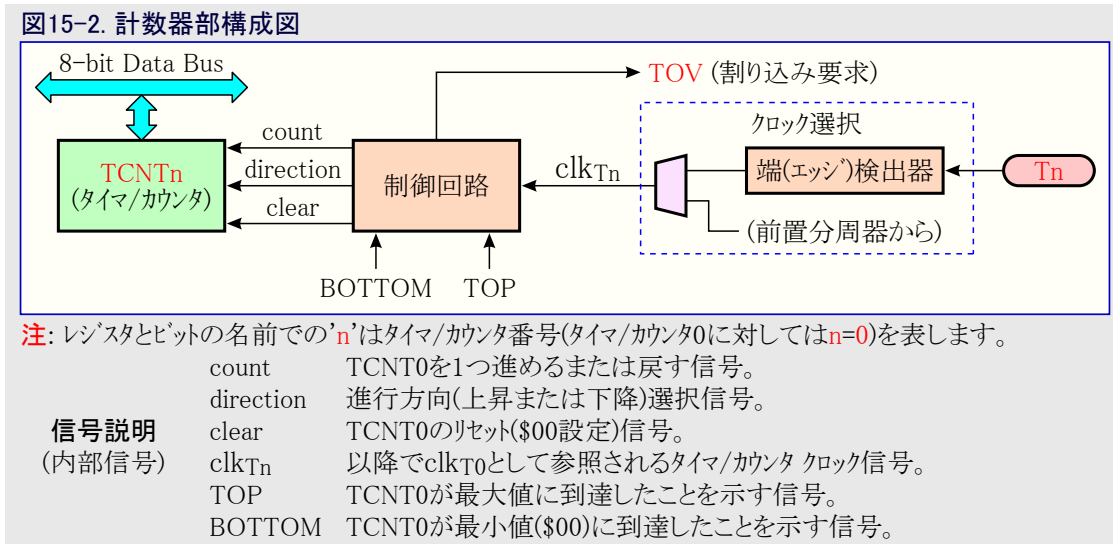
15.3. タイマ/カウンタのクロック

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択されます。

関連リンク [99頁の「タイマ/カウンタ0と1の前置分周器」](#)

15.4. 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(clk_{T0})で解除(\$00)、増加(+1)、または減少(-1)されます。 clk_{T0} 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、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(clk_{T0})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は「[動作種別](#)」をご覧ください。

タイマ/カウンタ溢れ(TOV)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。TOVはCPU割り込み発生に使えます。

15.5. 比較出力部

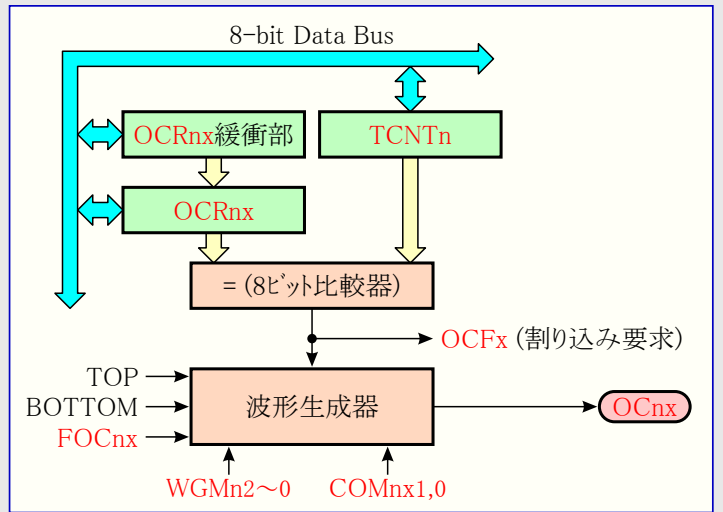
この8ビット比較器はTCNT0と比較レジスタ(OCR0AとOCR0B)を継続的に比較します。TCNT0がOCR0AまたはOCR0Bと等しければ比較器は一致を指示します。この一致は次のタイマ/カウンタクロック周期で比較割り込み要求フラグ(OCFAまたはOCFB)を設定(1)します。対応する割り込みが許可(I=1, OCIEAまたはOCIEB=1)されているならば、その比較割り込み要求フラグは比較割り込みを発生します。比較割り込み要求フラグは割り込みが実行されると自動的に解除(0)されます。代わりにこのフラグはこのI/Oビット位置に1を書くことによってソフトウェアでも解除(0)できます。波形生成器は波形生成種別(WGM02~0)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15-3. 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/カウンタに対してはn=0)、「x'は比較出力部(AまたはB)を表します。

15.5.1. 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(OCFx)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)。

15.5.2. 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。

15.5.3. 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。

15.6. 比較一致出力部

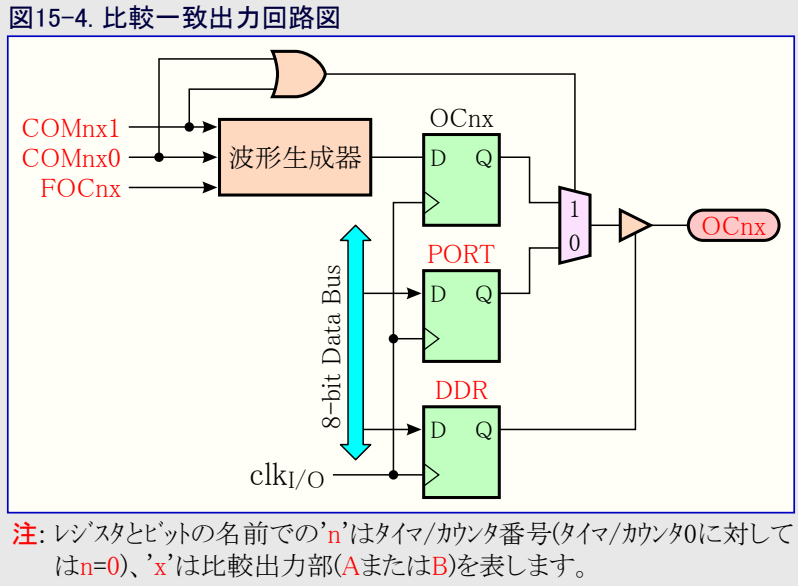
タイマ/カウンタ0制御レジスタAの比較出力選択(TCCR0A.COM0x1,0)ビットは以下の2つの機能を持ちます。

- 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。

15.6.1. 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。

15.7. 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 [70頁の「比較一致出力部」](#)
[74頁の「タイマ/カウンタ0のタイミング」](#)

15.7.1. 標準動作

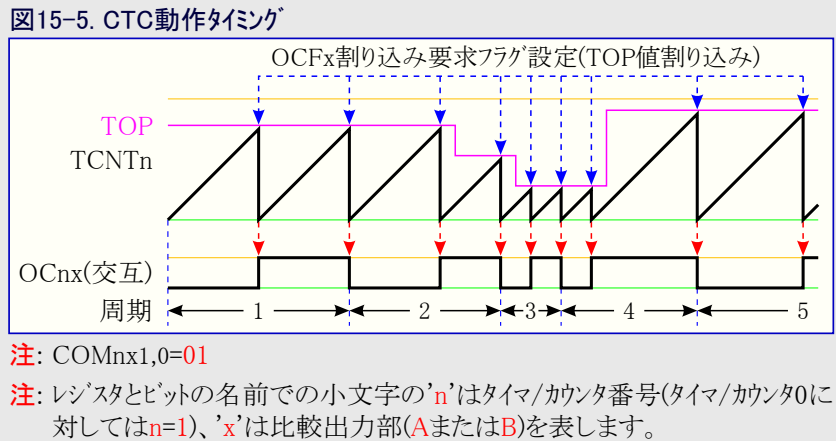
最も単純な動作種別が標準動作(WGM02~0=000)です。この動作種別での計数方向は常に上昇(+)で、カウンタ解除は実行されません。カウンタは8ビット最大値(TOP=\$FF)を通過すると単に範囲を超え、そして\$00(BOTTOM)から再び始めます。通常動作でのタイマ/カウンタ溢れ(TOV)フラグはTCNT0が\$00になるのと同じタイマ/カウンタクロック周期で設定(1)されます。この場合、TOVフラグは設定(1)のみで解除(0)されないことを除いて第9ビットのように動きます。けれどもTOVフラグを自動的に解除(0)するタイマ/カウンタ溢れ割り込みと組み合わせたタイマ/カウンタの分解能はソ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

15.7.2. 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、そしてその後にカウンタ(TCNT0)は解除(\$00)されます。



OCFA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。

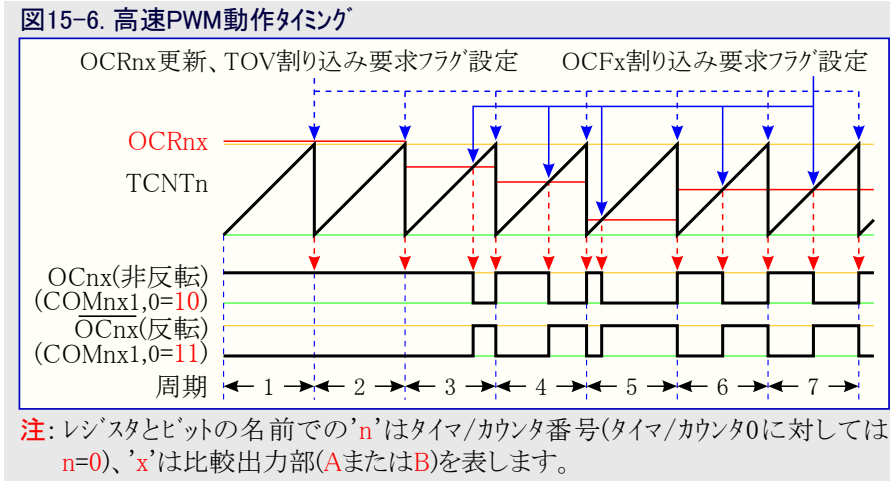
標準動作と同じように、タイマ/カウンタ溢れ(TOV)フラグはカウンタがMAXから\$00へ計数するのと同じタイマ/カウンタクロック周期で設定(1)されます。

15.7.3. 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(OCFx)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(TOV)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。

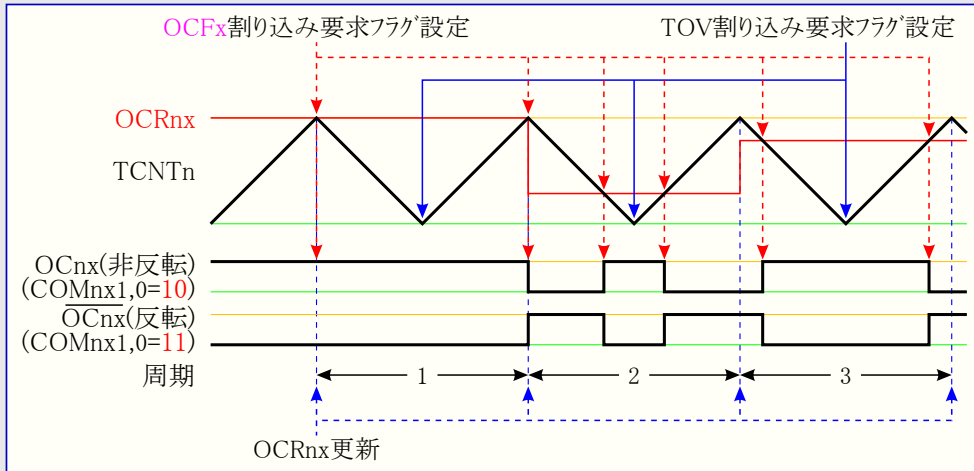
(訳補:WGM02~0=111の場合については、)デューティ比50%での周波数の波形出力は高速PWM動作で比較一致毎に論理反転するOC0A(COM0A1,0=01)選択によって達成できます。生成された波形はOCR0Aが\$00に設定される時に $f_{OC0x} = f_{clk_I/O} / 2$ の最大周波数です。この特性は高速PWM動作で比較出力部の2重緩衝が許可されることを除き、CTC動作でのOC0A交互出力(COM0A1,0=01)と同じです。

15.7.4. 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15-7. 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(TOV)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。

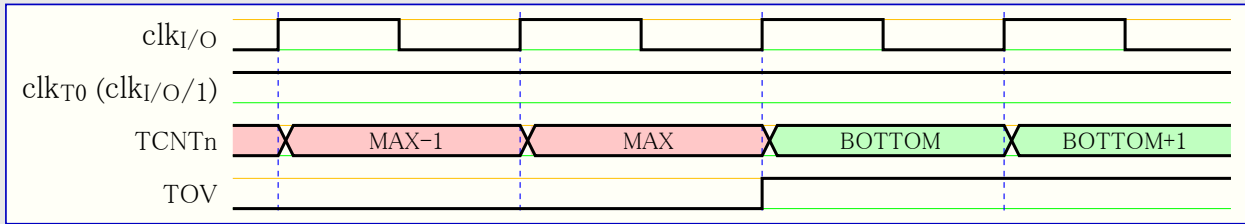
- タイミング図のようにOCR0xはTOPからその値を変更します。OCR0x値がTOPのとき、OCnxピン値は下降計数での比較一致の結果と同じです(訳補: L→H、直前がHのため、常にH)。BOTTOMを挟む対称を保証するため、(変更直後の)TOP(位置)でのOCnx値は上昇計数での比較一致の結果(H→L)と一致しなければなりません。
- タイマ/カウンタがOCR0x値よりも高い値から数え始め、そしてその理屈のために比較一致、従ってOCnxは上昇途中で起こされるであろう変更を起こしません。(訳補: 従って上記同様、TOP位置で(直前がHならば)H→L遷移が生じます。)

15.8. 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。

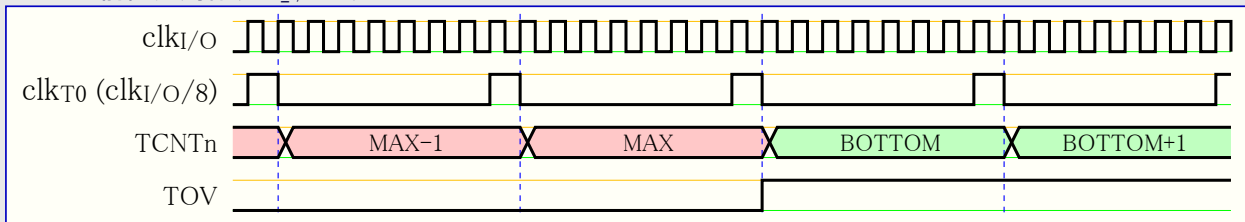
図15-8. 前置分周なし(1/1)のタイマ/カウンタタイミング



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。

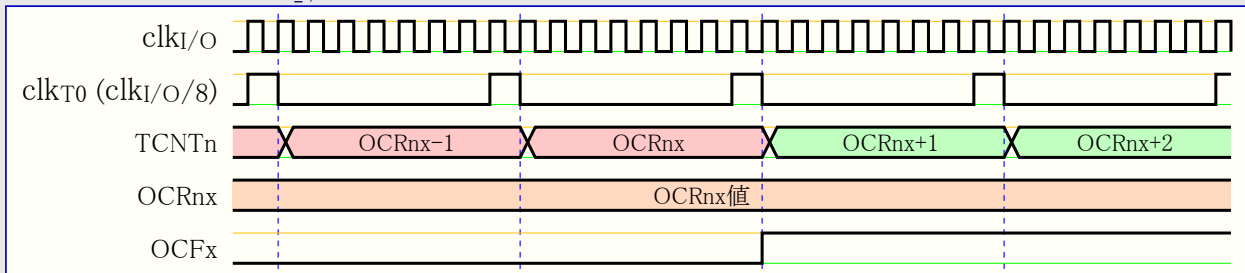
図15-9. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタタイミング



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OCFAと全動作種別でのOCFBの設定を示します。

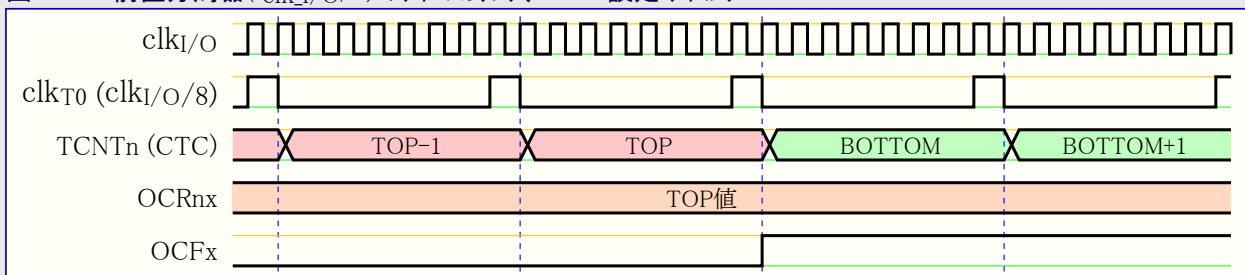
図15-10. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、OCFx設定タイミング



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OCFAの設定を示します。

図15-11. 前置分周器($f_{clk_{I/O}/8}$)のタイマ/カウンタ、OCFA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。

15.9. 8ビット タイマ/カウンタ0用レジスタ

15.9.1. 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15-2. 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15-3. 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15-4. 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15-5. 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ピントル(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15-6. 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15-7. 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**(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15-8. 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	TOV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。

15.9.2. 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15-9. 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 _{I/O} (前置分周なし)
0	1	0	clk _{I/O} /8 (8分周)
0	1	1	clk _{I/O} /64 (64分周)
1	0	0	clk _{I/O} /256 (256分周)
1	0	1	clk _{I/O} /1024 (1024分周)
1	1	0	T0ピンの下降端 (外部クロック)
1	1	1	T0ピンの上昇端 (外部クロック)

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

15.9.3. TCNT0 – タイマ/カウンタ0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0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。

15.9.4. OCR0A – タイマ/カウンタ0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0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。

15.9.5. OCR0B – タイマ/カウンタ0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0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。

15.9.6. 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
	-	-	-	-	-	OCIEB	OCIEA	TOIE
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 – OCIEB : 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)**

OCIEB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(OCFB)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 – OCIEA : 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)**

OCIEA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(OCFA)が設定(1)されると、対応する割り込みが実行されます。

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

TOIE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(TOV)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。

15.9.7. 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
	-	-	-	-	-	OCFB	OCFA	TOV
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 – OCFB : 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)**

OCFB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、OCFB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OCFB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(OCIEB)ビット、OCFB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 – OCFA : 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)**

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

TOV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、TOV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TOV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(TOIE)ビット、TOV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)されません。

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15-8を参照してください。

16. TC1 – 16ビット タイマ/カウンタ1 (PWM付き)

関連リンク [99頁の「タイマ/カウンタ0と1の前置分周器」](#)

16.1. 概要

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

この16ビット タイマ/カウンタの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**赤字(訳注: 原文太字)**で示されます。デバイス仕様のI/Oレジスタとビット位置は「[16ビット タイマ/カウンタ1用レジスタ](#)」で示されます。実際のI/Oピンの配置については「[ピン配置](#)」記述を参照してください。

関連リンク [51頁の「入出力ポート」](#)
[5頁の「ピン配置」](#)

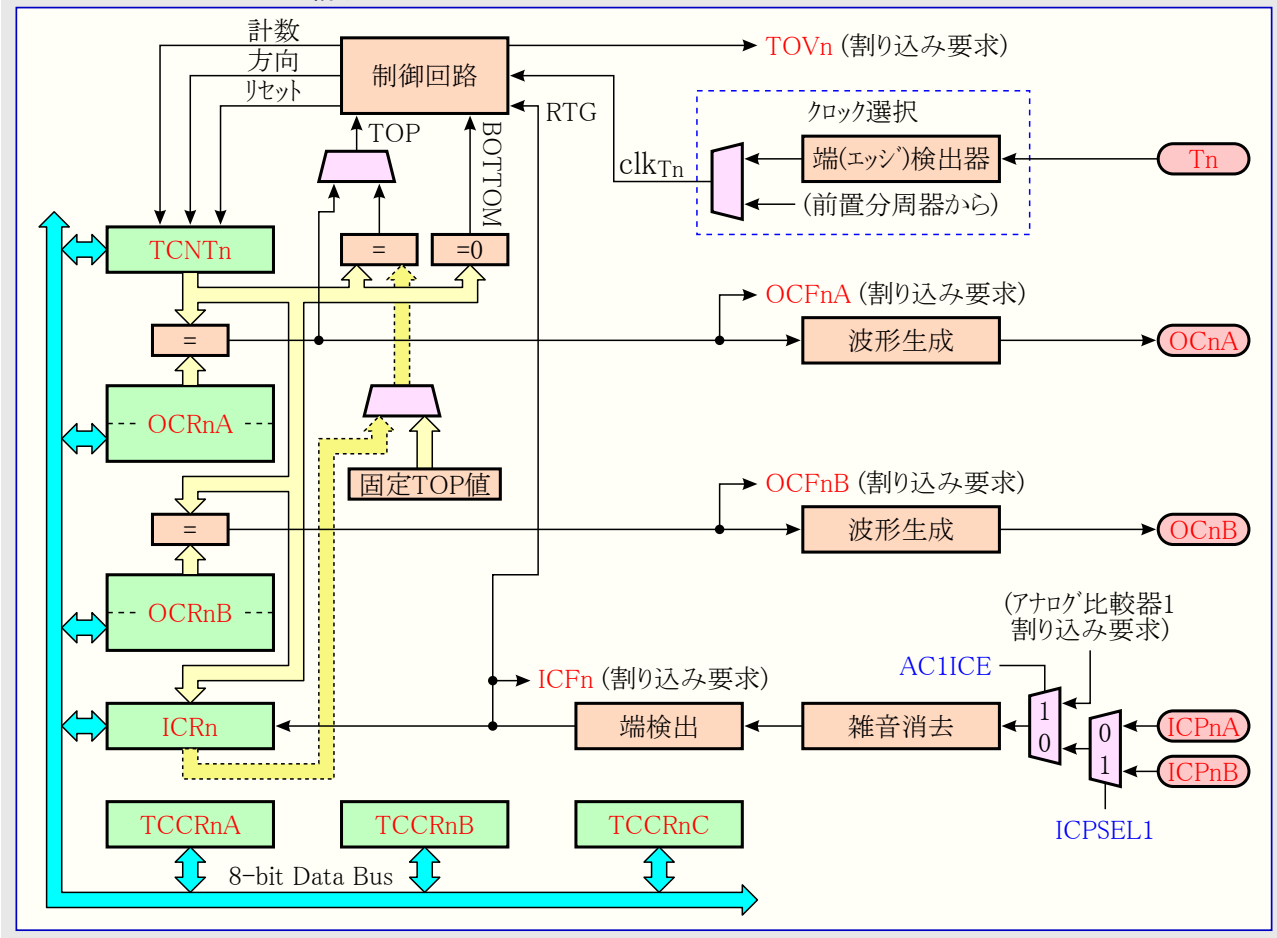
16.2. 特徴

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

16.3. 構成図

電力削減レジスタのTC1電力削減(PRR.PRTIM1)ビットはタイマ/カウンタ1部を許可するために0を書かれなければなりません。

図16-1. 16ビット タイマ/カウンタ構成図



実際のピン配置については関連リンクをご覧ください。

16.4. 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。

- $n=1$ 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16-1. 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。

16.5. 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 タイマ/カウンタ レジスタのアクセス」項で記述されます。タイマ/カウンタ制御レジスタ(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(OCFA, OCFB)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。

16.6. 16ビット タイマ/カウンタ 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(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りません。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りません。

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言語を使う時にコンパイラが16ビット アクセスを扱うことに注意してください。

```
アセンブリ言語プログラム例
~
LDI    R17, $01           ; [16ビット($01FF)書き込み]
LDI    R16, $FF           ; $01FFの上位バイト値取得
OUT    TCNT1H, R17        ; $01FFの下位バイト値取得
OUT    TCNT1L, R16        ; 上位バイト設定(一時レジスタ)
~                               ; 下位バイト設定(一時レジスタ⇒上位バイト)
IN     R16, TCNT1L        ; [16ビット読み込み]
IN     R17, TCNT1H        ; 下位バイト取得(上位バイト⇒一時レジスタ)
~                               ; 上位バイト取得(一時レジスタ)
~                               ;

C言語プログラム例
unsigned int i;
~                               /* */
TCNT1 = 0x1FF;                /* 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りません。

次のコード例はTCNT1レジスタ内容の非分断読み込み実行法を示します。OCR1A,OCR1B,ICR1レジスタは同じ原理を使用する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 [9頁の「コード例について」](#)

16.6.1. 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。

16.7. タイマ/カウンタのクロック

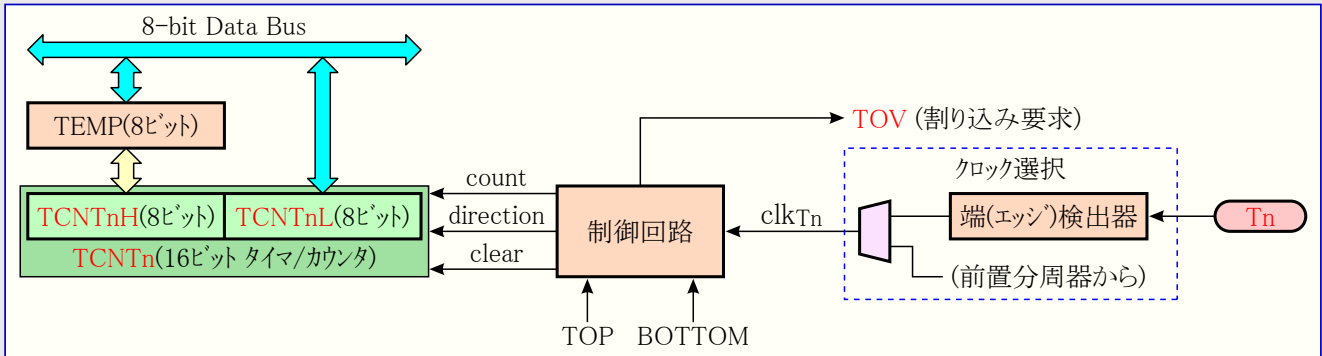
このタイマ/カウンタは内部または外部のクロック元によってクロック駆動できます。このクロック元はタイマ/カウンタ制御レジスタB(TCCR1B)のクロック選択(CS12~0)ビットによって制御されるクロック選択論理回路で選ばれます。

関連リンク 99頁の「タイマ/カウンタ0と1の前置分周器」

16.8. 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16-2. 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。

信号説明 (内部信号)	説明
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)。

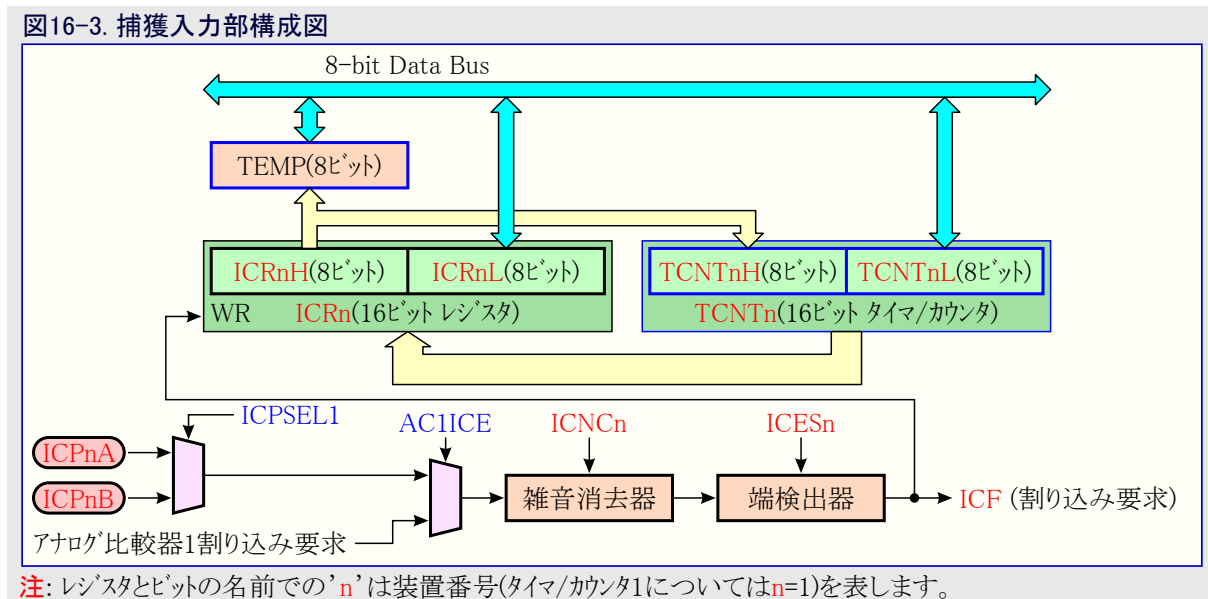
計数順序(方法)はタイマ/カウンタ制御レジスタA(TCCR1A)とTCCR1Bの波形生成種別(TCCR1B.WGM13,2とTCCR1A.WGM11,0)ビットの設定によって決定されます。これらはカウンタ動作(計数)方法と波形がOC1x比較出力に生成される方法間の接続に近いものです。進化した計数順序と波形生成についてより多くの詳細に対しては「動作種別」をご覧ください。

TC1のタイマ/カウンタ溢れ(TOV)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。TOVはCPU割り込み発生に使えます。

16.9. 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(ICP1AかICP1B)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1出力(AC1O)で論理レベルの変化(出来事)が起き、その変化が端(エッジ)検出器の設定を追認すると、捕獲が起動されます。捕獲が起動されると、カウンタ(TCNT1)の16ビット値が捕獲レジスタ(ICR1)に書かれます。捕獲割り込み要求フラグ(ICF)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, ICIE=1)ならば捕獲割り込み要求フラグは捕獲割り込みを発生します。ICF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)、その後に上位バイト(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、この一時レジスタをアクセスします。

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

「16ビットタイマ/カウンタレジスタのアクセス」もご覧ください。

16.9.1. 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(ICP1AまたはICP1B)ピンです。タイマ/カウンタ1は捕獲入力部用起動元としてアナログ比較器出力を代わりに使えます。アナログ比較器はアナログ比較器1制御(AC1CON)レジスタのアナログ比較器1捕獲起動許可(AC1ICE)ビットの設定(=1)によって起動元として選ばれます。起動元を変更することが捕獲を起動し得ることに気付いてください。従って捕獲割り込み要求フラグ(ICF)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1出力(AC1O)の両入力、T1ピンについてと同じ技法を使って採取されます。端(エッジ)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 99ページの「タイマ/カウンタ0と1の前置分周器」

16.9.2. 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。

16.9.3. 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(ICF)**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ICF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)、ICFの解除(0)は必要とされません。

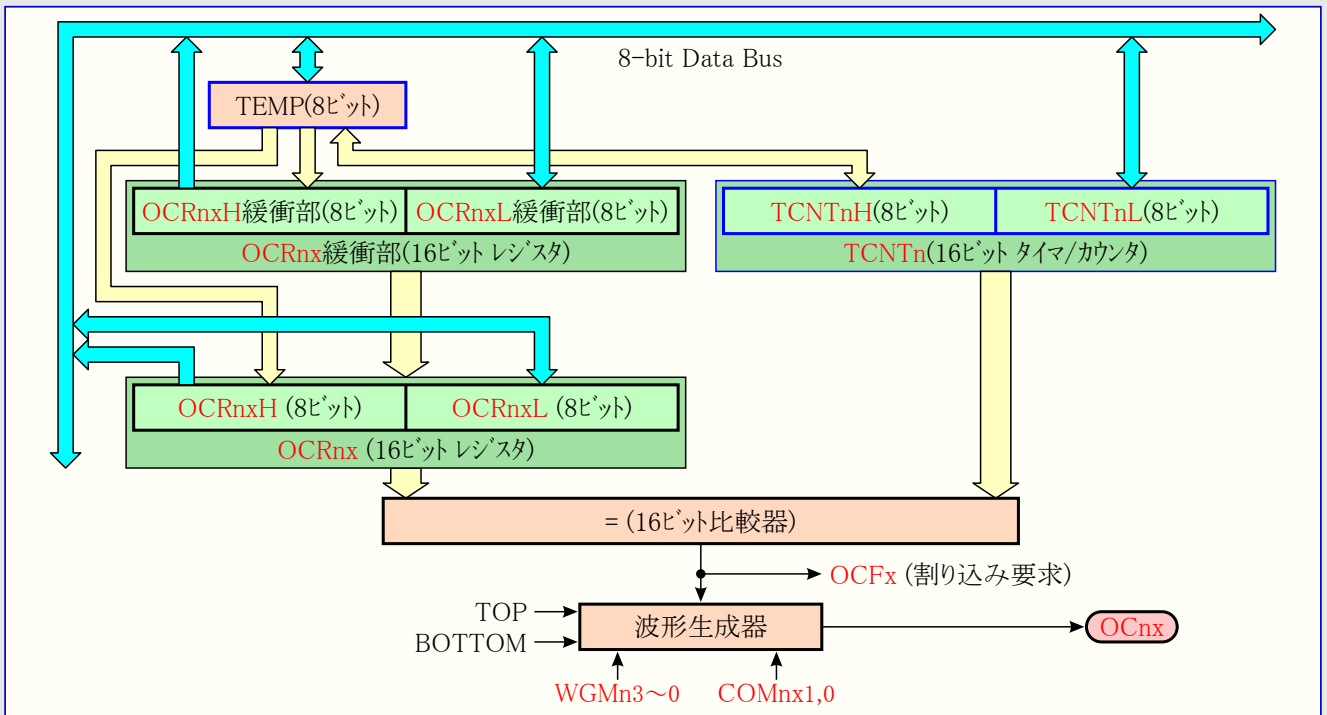
16.10. 比較出力部

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.OCFx)**を設定(1)します。許可(I=1, TIMSK1.OCIEx=1)ならば、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。OCFx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OCFx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16-4. 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 タイマ/カウンタ 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。

16.10.1. 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(OCFx)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)。

16.10.2. 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。

16.10.3. 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否かに拘らず、何れかの比較出力部を使う場合、TCNT1を変更する時に危険を伴います。TCNT1に書かれた値がOCR1x値と同じ場合、比較一致は失われ(一致が発生せず)、不正な波形生成に終わります。可変TOP値のPWM動作でTOPに等しいTCNT1を書いてはいけません。(行った場合)TOPに対する比較一致は無視され、カウンタは\$FFFFへ(計数を)続けます。同様にカウンタが下降計数のとき、BOTTOMに等しいTCNT1値を書いてはいけません。

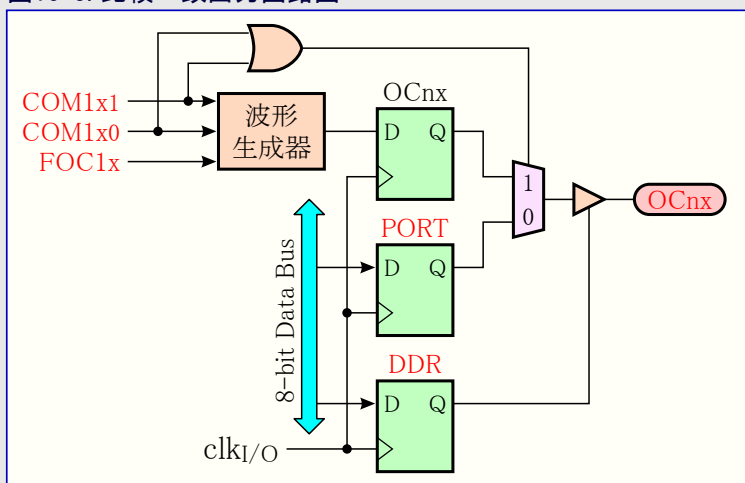
OC1xの初期設定はポートピンに対するポート方向レジスタを出力に設定する前に行われるべきです。OC1x値を設定する一番簡単な方法は標準動作で強制変更(FOC1x)ストロブビットを使うことです。波形生成種別間を変更する時であっても、OC1x(内部)レジスタはその値を保ちます。

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。

16.11. 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16-5. 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。

16.11.1. 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)ストロ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。

16.12. 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 87頁の「比較一致出力部」
92頁の「タイマ/カウンタのタイミング」

16.12.1. 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.TOV)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、TOV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TOV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。

16.12.2. 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、OCFAまたはICF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。

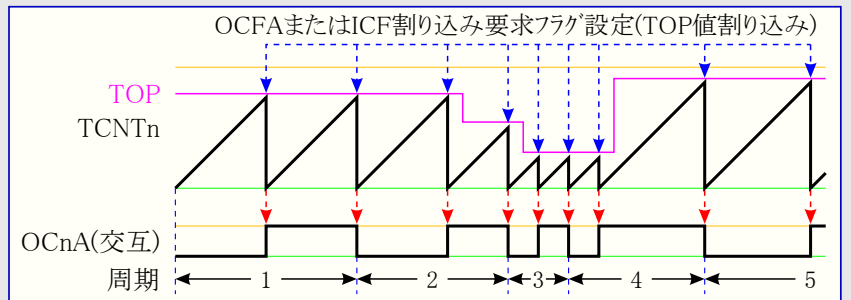
CTC動作で波形出力を生成するため、OC1A出力は比較出力選択(COM1A1,0)ビットを交互動作(=01)に設定することによって各比較一致での論理レベル交互切替に設定できます。OC1A値はそのピンに対するデータ方向が出力(DDR_OC1A=1)に設定されない限り、ポートピンで見えないでしょう。生成された波形はOCR1Aが0(\$0000)に設定される時に $f_{OC1A} = f_{clk_I/O} / 2$ の最大周波数を得ます。生成波形周波数は次式によって定義されます。

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

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(TOV)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16-6. 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。

16.12.3. 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。

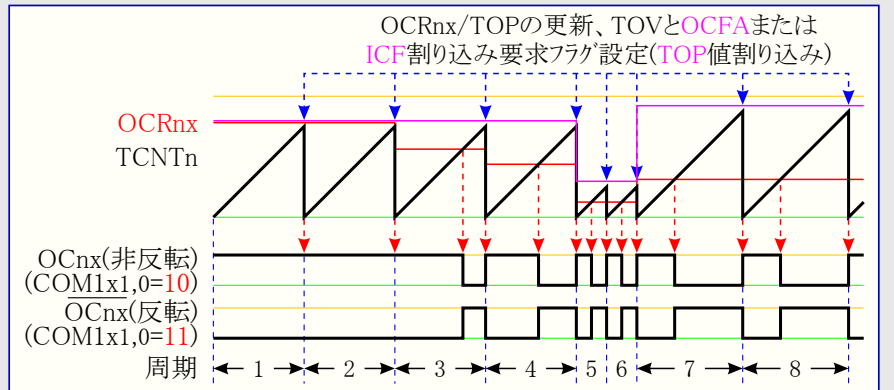
$$R_{FPWM} = \frac{\log(TOP+1)}{\log 2}$$

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(OCFx)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(TOV)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、OCFAまたはICF割り込み要求フラグはTOV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。

図16-7. 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。

ICR1がTOP値を定義するのに使われる時にICR1を更新する手順はOCR1Aの更新と異なります。ICR1は2重緩衝されません。これは前置分周なしまたは低い前置分周値でカウンタが走行している時にICR1が小さな値に変更される場合、書かれた新しいICR1値がTCNT1の現在値よりも小さくなる危険を意味します。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りません。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TOV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、OCR1A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。

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。

16.12.4. 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。

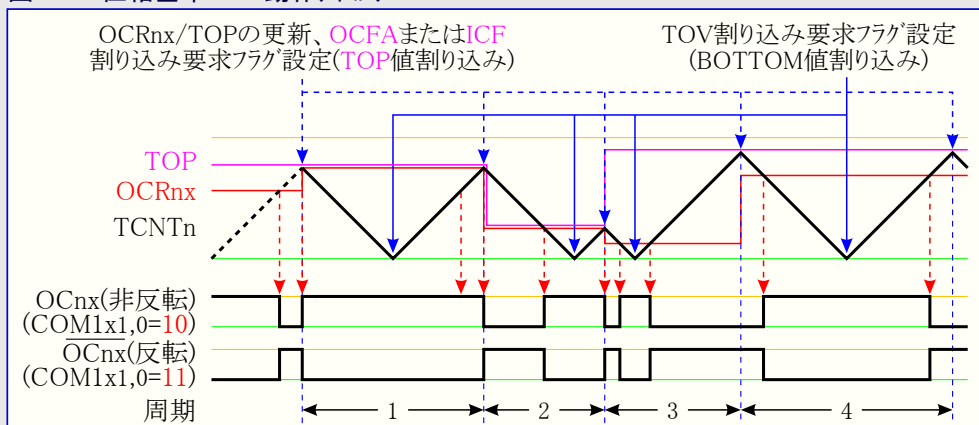
位相基準PWM動作のPWM分解能は8,9,10ビット固定、またはOCR1AかICR1のどちらかによって定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は右式を使うことによって計算できます。

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

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(OCFx)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(TOV)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OCFAまたはICF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16-8. 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。

16.12.5. 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16-8と下のタイミング図をご覧ください)。

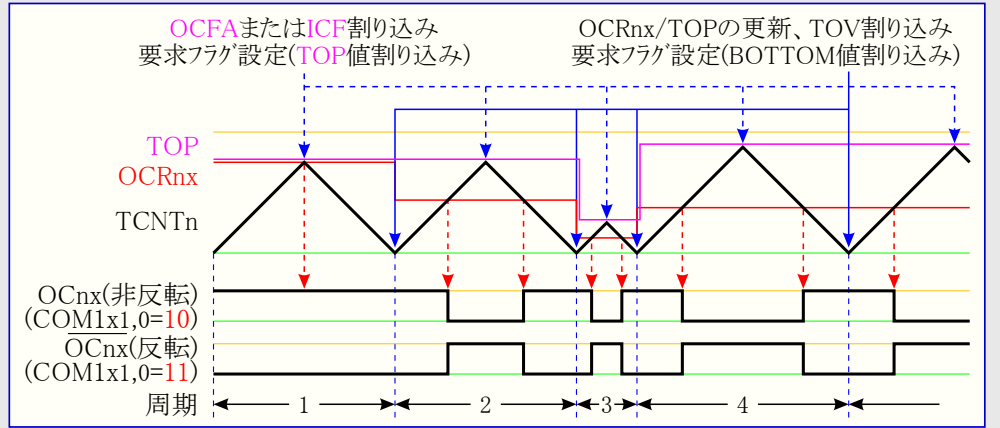
位相/周波数基準PWM動作のPWM分解能はOCR1AかICR1のどちらかで定義できます。許された最小分解能は2ビット(OCR1AまたはICR1が\$0003設定)、最大分解能は16ビット(OCR1AまたはICR1がMAX設定)です。ビットでのPWM分解能は右式を使うことによって計算できます。

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

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(OCFx)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(TOV)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、OCFAまたはICF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16-9. 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。

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

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。

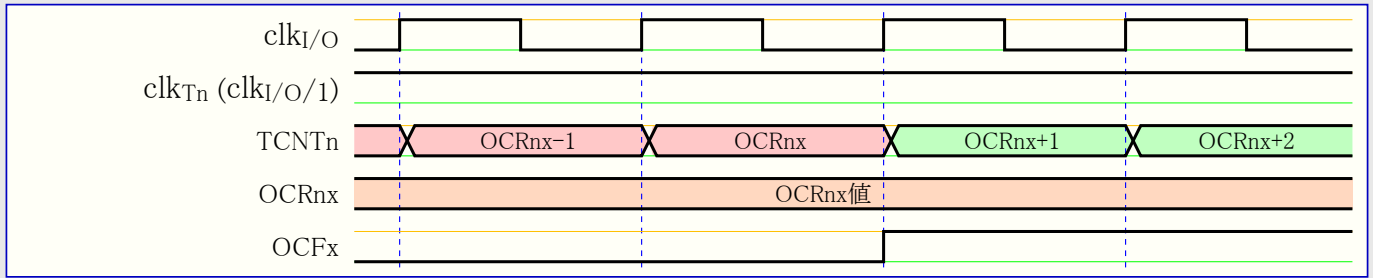
・Nは前置分周数(1,8,64,256,1024)を表します

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。

16.13. 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OCFxの設定についてのタイミング図を示します。

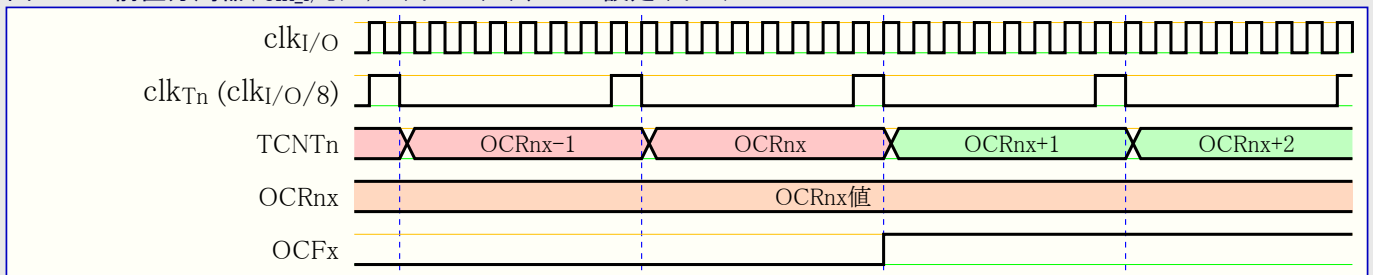
図16-10. 前置分周なし(1/1)のタイマ/カウンタ、OCFx設定 タイミング



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。

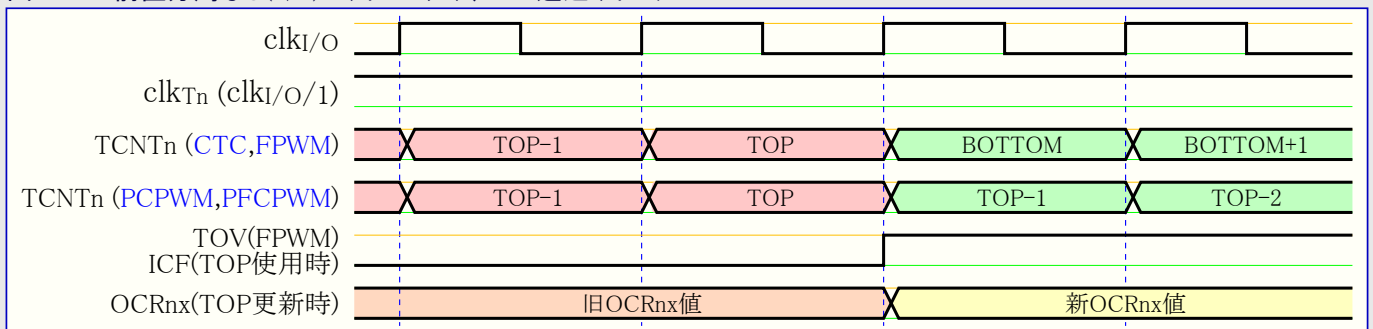
図16-11. 前置分周器($f_{clk_{1/O/8}}$)のタイマ/カウンタ、OCFx設定 タイミング



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でどのように置き換えられます。BOTTOMでTOV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。

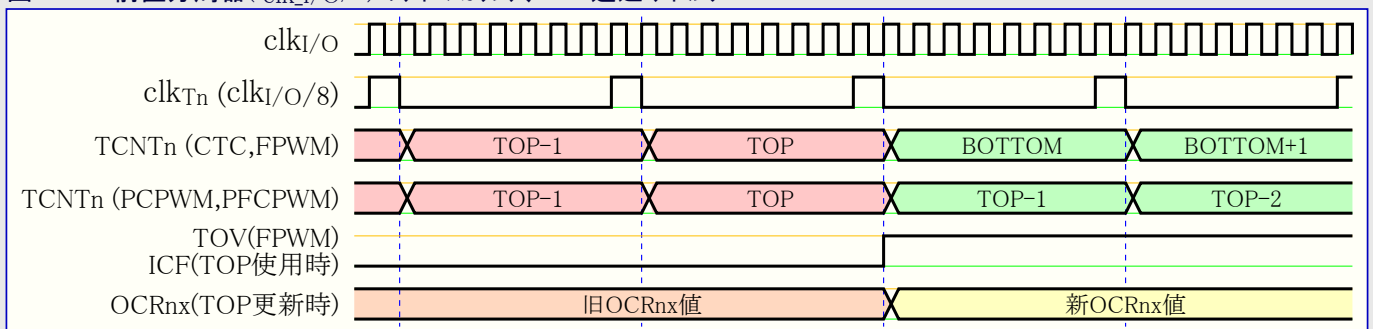
図16-12. 前置分周なし(1/1)のタイマ/カウンタ、TOP近辺 タイミング



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16-13. 前置分周器($f_{clk_{1/O/8}}$)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。

16.14. 16ビット タイマ/カウンタ1用レジスタ

16.14.1. 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16-2. 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16-3. 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16-4. 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16-5. 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	TOV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。

16.14.2. 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(ICF)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16-10と図16-11を参照してください。

表16-6. 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)

16.14.3. 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を書くと波形生成部で直ちに比較一致が強制されます。OC1x出力はCOM1x1,0ビット設定に従って変更されます。FOC1A/FOC1Bビットがスロープとして実行されることに注意してください。それによって強制された比較の効果を決めるのはCOM1x1,0ビットに存在する値です。

FOC1A/FOC1Bスロープは何れの割り込みの生成もTOPとしてOCR1Aを使う比較一致タイマ解除(CTC)動作でのタイマ/カウンタの解除(\$0000)も行いません。FOC1A/FOC1Bビットは常に0として読みます。

16.14.4. TCNT1H,TCNT1L - タイマ/カウンタ1計数值上位/下位バイト (TC1 Counter Value Low and High byte)

TCNT1HとTCNT1Lのレジスタ対は16ビット値のTCNT1を表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

名称 : TCNT1HとTCNT1L (TCNT1)

変位 : \$84

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	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	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15~0 - TCNT115~0 : タイマ/カウンタ1計数值 (Timer/Counter Counter Value)

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有されます。詳細については「16ビットタイマ/カウンタ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。

16.14.5. OCR1AH,OCR1AL – タイマ/カウンタ1比較レジスタ上位/下位バイト (Output Compare Register 1 A Low and High byte)

OCR1AHとOCR1ALのレジスタ対は16ビット値のOCR1Aを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「[16ビットレジスタのアクセス](#)」をご覧ください。

名称 : OCR1AHとOCR1AL (OCR1A)

変位 : \$88

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	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	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15~0 – OCR1A15~0 : タイマ/カウンタ1比較A値 (Output Compare A Value)

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OC1A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 タイマ/カウンタ レジスタのアクセス](#)」を参照してください。

16.14.6. OCR1BH,OCR1BL – タイマ/カウンタ1比較Bレジスタ上位/下位バイト (Output Compare Register 1 B Low and High byte)

OCR1BHとOCR1BLのレジスタ対は16ビット値のOCR1Bを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「[16ビットレジスタのアクセス](#)」をご覧ください。

名称 : OCR1BHとOCR1BL (OCR1B)

変位 : \$8A

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	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	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15~0 – OCR1B15~0 : タイマ/カウンタ1比較B値 (Output Compare B Value)

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OC1B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 タイマ/カウンタ レジスタのアクセス](#)」を参照してください。

16.14.7. ICR1H,ICR1L – タイマ/カウンタ1捕獲レジスタ上位/下位バイト (Input Capture Register 1 Low and High byte)

ICR1HとICR1Lのレジスタ対は16ビット値のICR1を表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

名称 : ICR1HとICR1L (ICR1)

変位 : \$86

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	ICR1 _{15~8}							
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	6	5	4	3	2	1	0
	ICR1 _{7~0}							
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15~0 – ICR1_{15~0} : タイマ/カウンタ1捕獲値 (Input Capture Value)

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ビットタイマ/カウンタレジスタのアクセス」を参照してください。

16.14.8. 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
	-	-	ICIE	-	-	OCIEB	OCIEA	TOIE
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 – ICIE : 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を書かれて、ステータスレジスタ(SREG)の全割り込み許可(I)ビットが設定(1)されると、タイマ/カウンタ1捕獲割り込みが許可されます。タイマ/カウンタ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(ICF)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 – OCIEB : 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(OCFB)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 – OCIEA : 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(OCFA)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 – TOIE : 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(TOV)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。

16.14.9. 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\$16です。

ビット	7	6	5	4	3	2	1	0
	–	–	ICF	–	–	OCFB	OCFA	TOV
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 – ICF : 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、ICF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、ICF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ICFは解除(0)できます。

● ビット2 – OCFB : 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)ストロブがOCFB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、OCFB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OCFBは解除(0)できます。

● ビット1 – OCFA : 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)ストロブがOCFA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、OCFA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OCFAは解除(0)できます。

● ビット0 – TOV : 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TOVフラグはタイマ/カウンタ1溢れ時に設定(1)されます。他のWGM13～0ビット設定を使う時のTOV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、TOV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TOVは解除(0)できます。

17. 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 [67頁の「TC0 - 8ビット タイマ/カウンタ0 \(PWM付き\)」](#)
[80頁の「TC1 - 16ビット タイマ/カウンタ1 \(PWM付き\)」](#)

17.1. 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。

17.2. 前置分周器リセット

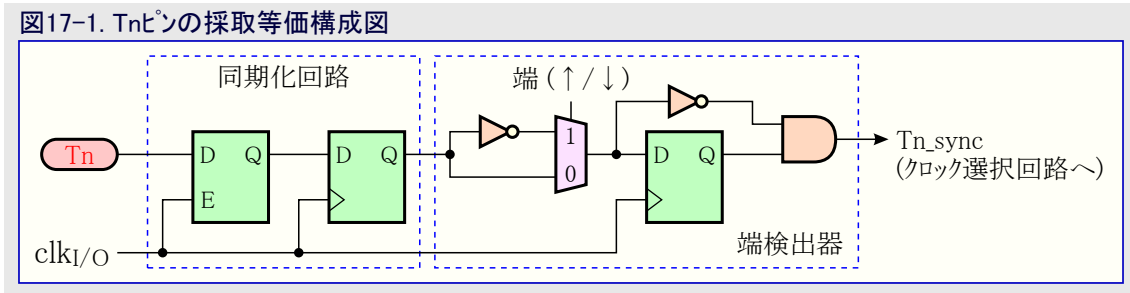
この前置分周器は自由走行で(換言するとタイマ/カウンタのクロック選択論理回路と無関係に動作する)、タイマ/カウンタ0とタイマ/カウンタ1によって共有されます。前置分周器はタイマ/カウンタのクロック選択によって影響を及ぼされないため、前置分周器の状態は前置分周したクロックが使われる状況に対して密接に関係します。タイマ/カウンタが許可され、前置分周器によってクロック駆動される(CSn2~0=5~2)とき、前置分周加工の一例が生じます。タイマ/カウンタが許可される時から最初の計数が起きるまでのシステム クロック周期数は、Nが前置分周値(8, 64, 256, 1024)とすると、 $1 \sim N+1$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

17.3. 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。



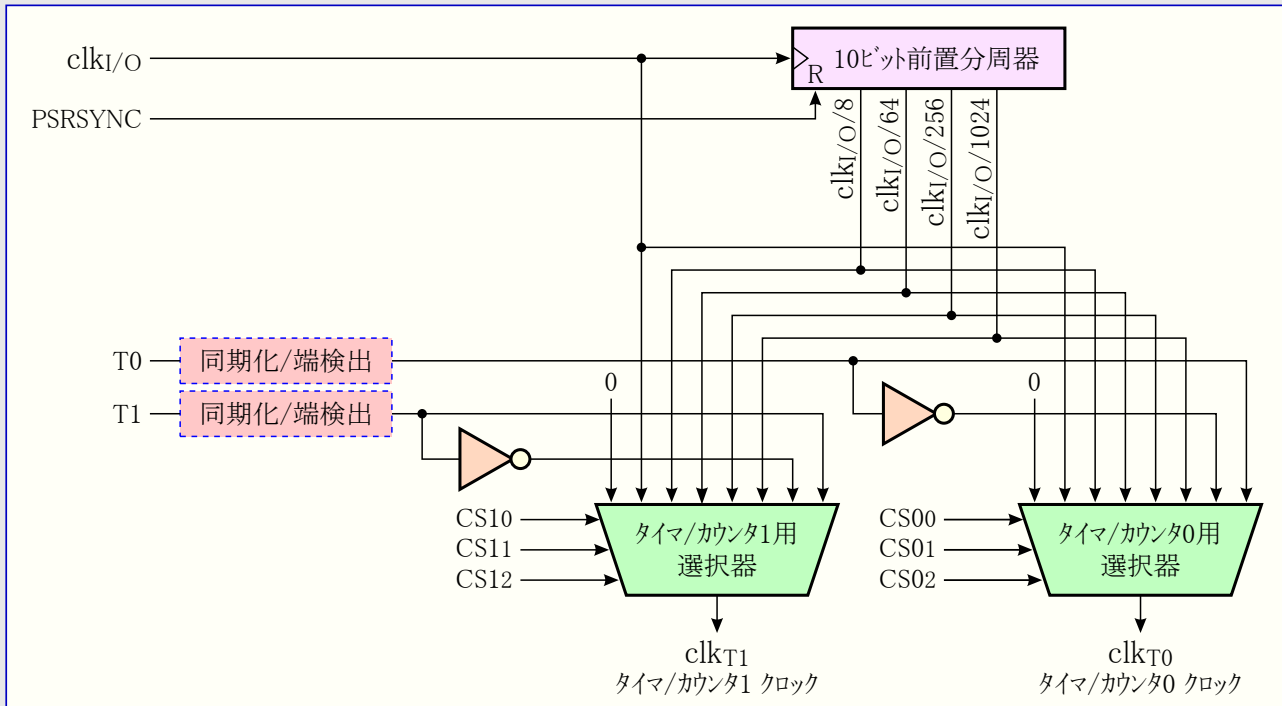
同期化と端検出器論理回路はTnピンへ印加された端から計数が更新されるまでに2.5~3.5システム クロック周期の遅延をもたらします。

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17-2. 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。

17.4. タイマ/カウンタ捕獲ピン・前置分周器用レジスタ

17.4.1. 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	ICPSEL1	-	-	-	-	-	PSRSYNC
アクセス種別	R/W	R/W	R	R	R	R	R	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PSRSYNCへ書かれる値は保持され、従って対応する前置分周器リセット信号の有効を保持します。これは対応するタイマ/カウンタを停止し、設定中にそれらの1つが進行する危険なしに同じ値に設定できることを保証します。TSMビットが0を書かれると、PSRSYNC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6 – ICPSEL1 : タイマ/カウンタ1 捕獲起動ピン選択 (Timer/Counter1 Input Capture selection)

タイマ/カウンタ1の捕獲機能はICP1A(PD4)とICP1B(PC3)の2つの利用可能な入力を持ちます。

表17-1. タイマ/カウンタ1捕獲起動入力ピン選択

ICPSEL1	説明
0	捕獲起動入力ピンとしてICP1A選択
1	捕獲起動入力ピンとしてICP1B選択

● 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0,1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0,1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言及したタイマ/カウンタに影響を及ぼすことに注意してください。

18. PSC – 電力段制御器 (Power Stage Controller)

18.1. 特徴

- 設定可能な6つの相互補完出力を持つPWM波形生成機能 (3つの半ブリッジの制御が可能)
- 設定可能な沈黙時間制御
- 最大12ビット分解能のPWM
- 64MHzまでのPWMクロック周波数 (PLL経由)
- 設定可能なA/D変換起動
- 自動重複保護
- 3つの異常時安全保護緊急入力 (全出力をHi-Zまたは非活性状態(ヒューズ設定可能)に強制)
- 中央整列と端(エッジ)整列の同期動作

18.2. 概要

この電力段制御器は高性能波形制御器です。

本章でのレジスタとビット参照の多くは一般形で書かれます。

- 小文字の'n'は電力段制御器(PSC)単位部番号の置き換えで、この場合は0,1,2です。けれどもプログラムに於いてレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えば単位部0のPOCRnSAHレジスタにアクセスするためのPOCR0SAHのように)。
- 小文字の'x'は電力段制御器(PSC)部分の置き換えで、この場合はAまたはBです。けれどもプログラムに於いてレジスタまたはビット定義に使う時は正確な形式が使われなければなりません(例えばA部分のOCR0SxHにアクセスするためのOCR0SAHのように)。

電力段制御器(PSC)の目的は外部の電力インターフェースを制御することです。これは例えば3つの半ブリッジを駆動するために6つの出力を持ちます。この特徴は非同期またはブラシレスDC(BLDC)電動機駆動、照明システムなどのような応用に関する3相波形の生成を許します。

PSCは3つの入力も持ち、この目的は高速緊急停止能力を提供することです。

PSC出力は“High活性”または“Low活性”として設定可能です。以降の例に於ける全てのタイミング図は“High活性”極性で与えられます。

18.3. 16ビット レジスタのアクセス

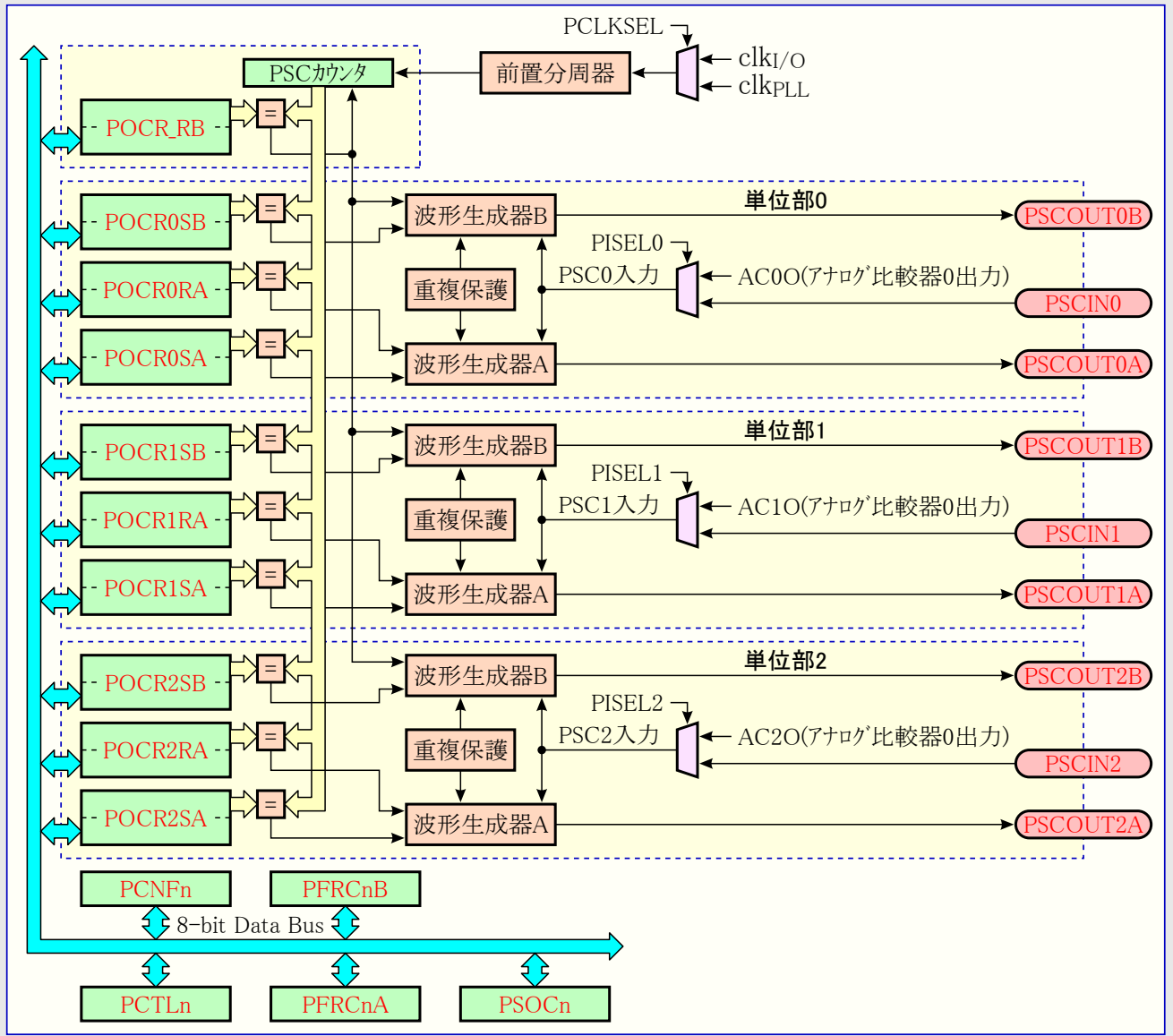
いくつかのPSCレジスタは16ビット レジスタです。これらのレジスタはAVR CPUによって8ビット データ バス経由でアクセスすることができます。16ビット レジスタは2つの読みまたは書きの操作を用いてバイトでアクセスされなければなりません。PSCは16ビット アクセスの上位バイトの一時的な格納用に単一の8ビット レジスタを持ちます。PSCの全ての16ビット レジスタ間で同一レジスタが共用されます。下位バイトのアクセスが16ビットの読みまたは書きの操作を起動します。CPUによって16ビット レジスタの上位バイトが書かれる時に、その上位バイトは一時レジスタに格納され、そして下位バイト書き込みは同じクロック周期で上下両バイトが16ビット レジスタ内に複写されます。CPUによって16ビット レジスタの下位バイトが読まれる時に、16ビット レジスタの上位バイトは下位バイトが読まれるのと同じクロック周期で一時レジスタ内に複写されます。

16ビット書き込みを行うには、下位バイトの前に上位バイトが書かれなければなりません。16ビット読み込みについては、上位バイトの前に下位バイトが読まれなければなりません。

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

18.4. 電力段制御器(PSC)説明

図18-1. 電力段制御器(PSC)構成図



PSCは自由走行の12ビット計数器(PSCカウンタ)の使用に基づきます。この計数器はPSC比較RB(POCR_RB)レジスタの内容によって決められる頂上値へ上昇計数でき、そして別(次)の周期のために、選んだ動作形態に従って下降計数または0にリセットします。

上の構成図で見ることができるよう、PSCは3つの単位部から成ります。

3つのPSC単位部の各々是对称的な2つの実体として見ることができます。A部と名付けられた1つの実体がPSCOUTnA出力を生成し、B部と名付けられた2つ目の実体がPSCOUTnB出力を生成します。

各単位部は対応する入力を管理する自身のPSC入力回路を持ちます。

18.5. 機能説明

18.5.1. 制御波形の生成

一般的に3相電動機の駆動は6つのPWM信号の生成が必要です。電動機の色度やトルクを調節する、または3つの電圧線での望む波形(台形、正弦状など)を生成ために、これらの信号のデューティサイクルは個別に制御されなければなりません。

短絡(cross conduction:交差伝導)や温度超過の場合に、波形生成器の出力を直ちに禁止することができる入力を持つことが望まれます。

これらの考慮は照明、DC/DC変換器などのように電力系を駆動するのにPWM信号が必要な多くのシステムに対して共通です。

18.5.2. 波形周期

3つの単位部の各々は出力信号を共同で構成する2つの波形生成器を持ちます。

波形の始めの部分はA部またはPSCOUTnA出力に関係します。この波形部分は後続図での補助周期Aに対応します。

2つ目の波形の次の(後半)部分はB部またはPSCOUTnB出力に関係します。この波形部分は後続図での補助周期Bに対応します。

(周期の)全体波形は補助周期Bの最後で終了され、次周期のために波形生成器レジスタの設定へのどの変更もここで実行されます。

電力段階制御器(PSC)は2つの動作形態(1傾斜形態または中央整列形態)の1つに構成設定することができます。この構成設定は全ての波形生成器の動作に影響を及ぼします。

図18-2. 1傾斜形態での周期表現図

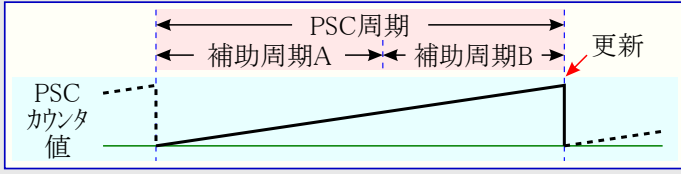
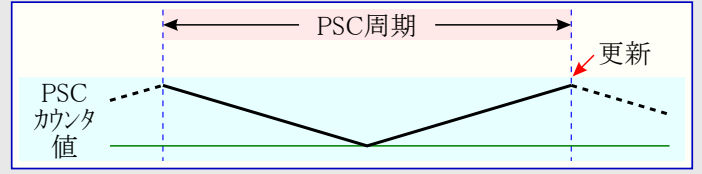


図18-3. 中央整列形態での周期表現図



上図はPSCカウンタ内に保持される値を図形的に図解します。中央整列形態は下降そして上昇と計数する1傾斜形態のようです。

波形生成器レジスタの更新が傾斜形態に拘らずPSC周期の最後で行われることに注意してください。

18.5.3. 動作形態説明

出力信号の波形と長さは走行形態と(比較nSA(POCRnSA)、比較nRA(POCRnRA)、比較nSB(POCRnSB)、比較RB(POCRnRB))レジスタ内に保持される項目によって決められます。次の2つの形態が可能です。

- **1傾斜形態:** この形態では3つのPSCOUTnB出力が端整列され、専用レジスタに同じ値を設定する時にPSCOUTnAも端整列にすることができます。

この形態でのPWM周波数は中央整列形態PWM周波数の2倍です。

- **中央整列形態:** この形態では6つのPSC出力の全てが周期中央で整列されます。3つの単位部で同じデューティサイクルを使う時を除き、出力端は整列されません。このためPSC出力は同じ時に転流(整流)を行わず、従ってこれらの出力によって駆動される系は転流(整流)雑音を生じません。

この形態でのPWM周波数は1傾斜形態PWM周波数より2倍遅くなります。

18.5.3.1. 1傾斜形態

以下の図はPSC周期上で1傾斜形態での結果としてのPSCOUTnAとPSCOUTnBの出力を示します。

$$\text{ON時間A} = (\text{POCRnRA} - \text{POCRnSA}) \times 1/f_{\text{CLKPSC}}$$

$$\text{ON時間B} = (\text{POCRnRB} - \text{POCRnSB}) \times 1/f_{\text{CLKPSC}}$$

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

$$\text{沈黙時間B} = (\text{POCRnSB} - \text{POCRnRA}) \times 1/f_{\text{CLKPSC}}$$

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

重複保護が禁止された場合、1傾斜形態に於いて互いに重複するように構成設定することができますが、一般的な使用に於いて、これは望まれることではありません。

図18-4. 1傾斜動作でのPSCOUTnAとPSCOUTnB基本波形

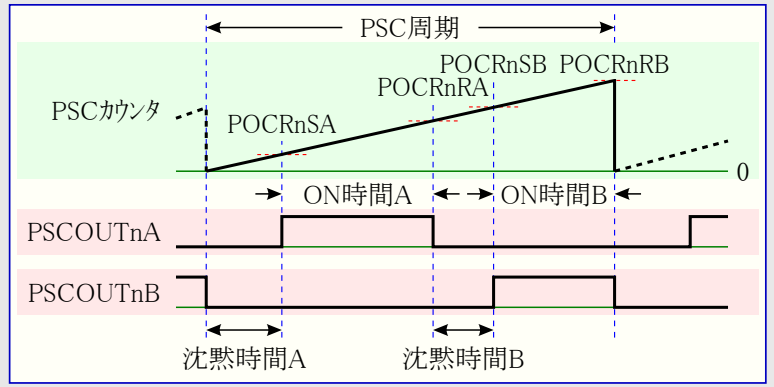
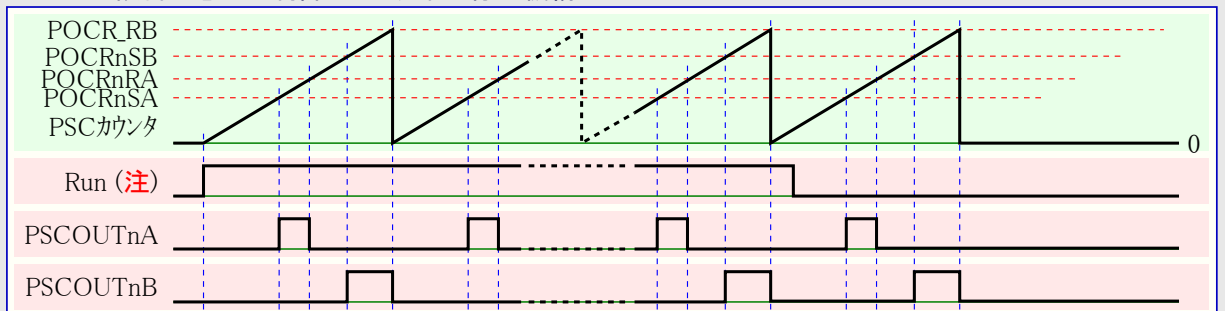


図18-5. 1傾斜形態での制御された走行と停止機構



注: 「PCTL - PSC制御レジスタ」をご覧ください(PSC周期完了許可(PCCYC)=1)。

18.5.3.2. 中央整列形態

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

$$\text{ON時間0} = 2 \times \text{POCRnSA} \times 1 / f_{\text{CLKPSC}}$$

$$\text{ON時間1} = 2 \times (\text{POCRnRB} - \text{POCRnSB} + 1) \times 1 / f_{\text{CLKPSC}}$$

$$\text{沈黙時間} = (\text{POCRnSB} - \text{POCRnSA}) \times 1 / f_{\text{CLKPSC}}$$

$$\text{PSC周期} = 2 \times (\text{POCRnRB} + 1) \times 1 / f_{\text{CLKPSC}}$$

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

中央整列形態でのPOCRnRAはPSC出力波形タイミング制御に(1傾斜形態でのように)必要とされません。これはA/D変換同期調節のために自由に使用されることをPOCRnRAに許します(「アナログ同期化」をご覧ください)。

図18-6. 中央整列形態でのPSCOUTnAとPSCOUTnB基本波形

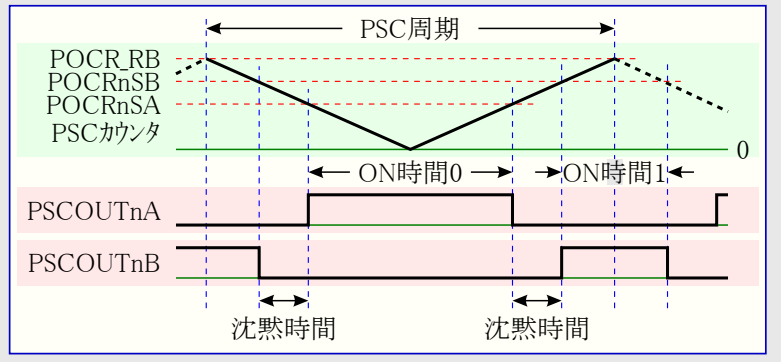
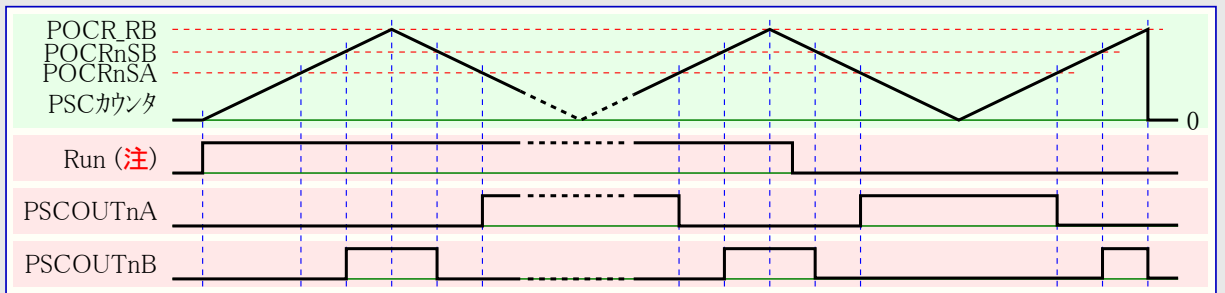


図18-7. 中央整列形態での制御された走行と停止機構

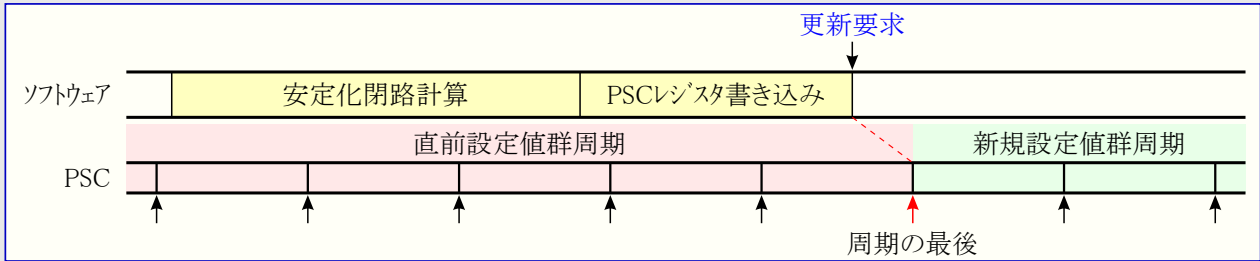


注: 「PCTL - PSC制御レジスタ」をご覧ください(PSC周期完了許可(PCCYC)=1)。

18.6. 値の更新

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

図18-8. 完全なPSC主機の最後での更新



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

18.6.1. 値の同期更新

新規タイミング値やPSC出力構成はPSC周期中に書けます。施錠(PULOCK)構成設定ビットによって新規値群全体がPSC周期の最後に続いて考慮され(扱われ)ます。

PULOCK構成設定ビットが設定(1)される時にそれらは更新されません。PSC内部レジスタの更新はPULOCKビットが0に開放された場合にPSC周期の最後で行なわれます。

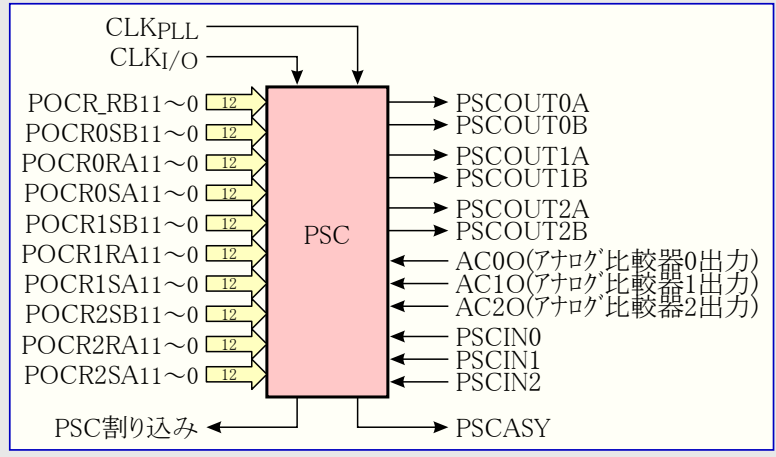
PULOCKで同期更新されるレジスタはPSC出力構成設定(POC)、PSCn比較SA(POCRnSAH/L)、PSCn比較RA(POCRnRAH/L)、PSCn比較SB(POCRnSBH/L)、PSC比較RB(OCRnRBH/L)のレジスタです。

18.7. 重複保護

重複保護によって同じ単位部の2つの出力は同時に活性にすることができません。このために短絡(cross conduction: 交差伝導)を生成し得ません。この機能はPSC単位部n入力制御(PMICn)レジスタのPSC重複許可(POVENn)ビットによって不活動にすることができます。

18.8. 信号説明

図18-9. PSC外部接続構成図



18.8.1. 入力説明

表18-1. 内部入力

名称	説明	形式/ビット数
POCSR_RB11~0	B部で信号をリセットする比較値 (PSCOUTnB)	12ビットレジスタ
POCSRnSB11~0	B部で信号をセットする比較値 (PSCOUTnB)	12ビットレジスタ
POCSRnRA11~0	A部で信号をリセットする比較値 (PSCOUTnA)	12ビットレジスタ
POCSRnSA11~0	A部で信号をセットする比較値 (PSCOUTnA)	12ビットレジスタ
CLKI/O	I/Oクロックからのクロック入力	信号
CLKPLL	PLLからのクロック入力	信号
AC0O	アナログ比較器0出力	信号
AC1O	アナログ比較器1出力	信号
AC2O	アナログ比較器2出力	信号

表18-2. PSC部外入力

名称	説明	形式/ビット数
PSCIN0	障害/再起動機能に使われる入力0	信号
PSCIN1	障害/再起動機能に使われる入力1	信号
PSCIN2	障害/再起動機能に使われる入力2	信号

18.8.2. 出力説明

表18-3. PSC部出力

名称	説明	形式/ビット数
PSCOUT0A	PSC0単位部A部出力	信号
PSCOUT0B	PSC0単位部B部出力	信号
PSCOUT1A	PSC1単位部A部出力	信号
PSCOUT1B	PSC1単位部B部出力	信号
PSCOUT2A	PSC2単位部A部出力	信号
PSCOUT2B	PSC2単位部B部出力	信号

表18-4. 内部出力

名称	説明	形式/ビット数
PSCn割り込み	PSC割り込み要求 (溢れ、障害の2種)	信号
PSCnASY	A/D変換器(+増幅器)同期 (「アナログ同期化」参照)	信号

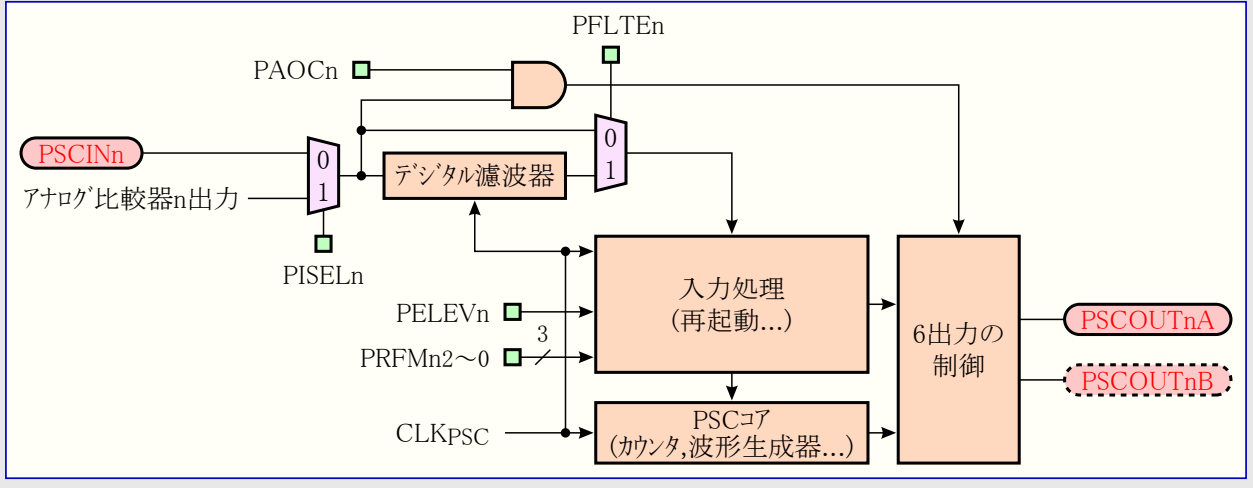
18.9. 電力段制御器(PSC)入力

PSCの詳細情報についてはAtmelのウェブサイトで購入可能な「AVR138:Atmega32M1系列電力段制御器(PSC)料理本」応用記述を参照してください。

PSCの0,1,2の各単位部は1つのPSC入力を考慮するための自身の系を持ちます。PSC単位部 n 入力制御(PMICn)レジスタに従って、PSCIN n 入力は再起動または異常入力の動きにすることができます。

A部またはB部の各々はPSC単位部 n 入力制御(PMICn)レジスタによっても構成設定されます。

図18-10. PSC入力部



18.9.1. PSC入力構成設定

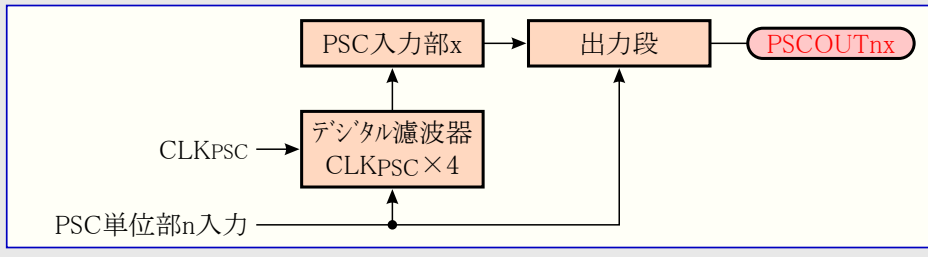
PSC入力構成設定は構成設定レジスタ内のビット設定によって行なわれます。

18.9.1.1. 濾波器許可

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

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

図18-11. PSC入力濾波器



18.9.1.2. 信号極性 (訳注:無効記述削除によって内容修正)

有効端(エッジ動作)または有効レベル(レベル動作)が選べます。「PMICn - PSC単位部 n 入力制御レジスタ」項のPELEV n ビット記述をご覧ください。

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

- 1傾斜動作でのPSC n 入力は傾斜全体で動作します。

18.9.1.3. 入力形態動作

3つの構成設定ビット(PRFMn2~0)で、PSC入力の形態を定義することが可能です。

表18-5. PSC入力形態動作 (注: n=0,1,2)

PRFMn2~0	説明
0 0 0	動きなし、PSC入力は無効にされます。
0 0 1	単位部nのA出力非活動
0 1 0	単位部nのB出力非活動
0 1 1	単位部nのAとBの出力非活動
1 0 x	全PSC出力非活動
1 1 x	PSC停止、ソフトウェアの動き待機

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

18.10. PSC入力形態1~5: タイミング変更なしで出力非活動

図18-12. 形態1~5でのPSCn入力対PSCの動き

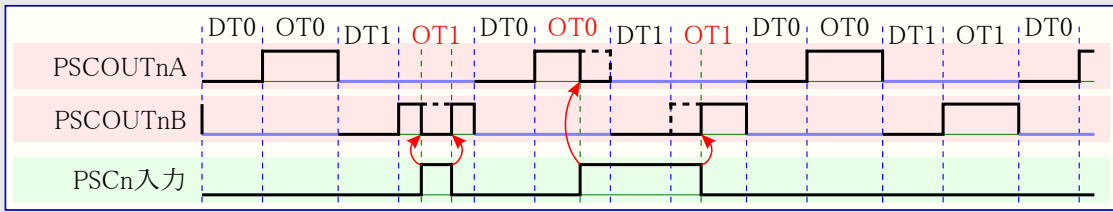
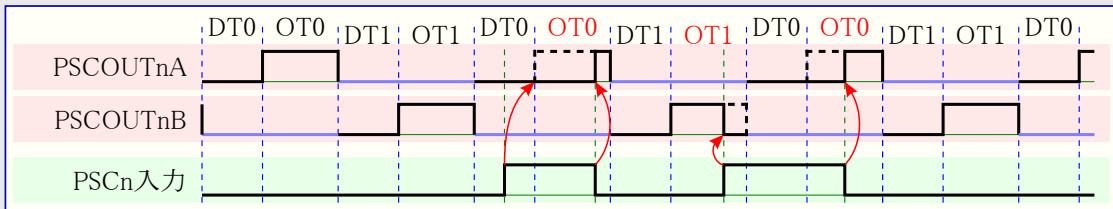


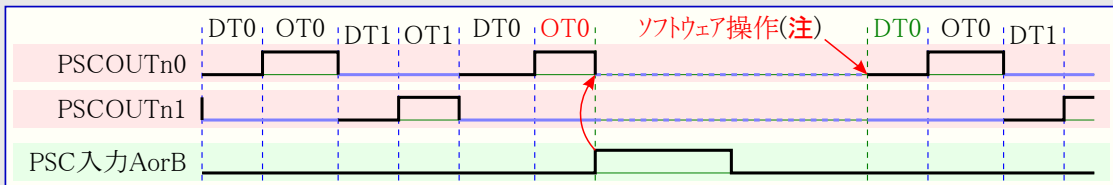
図18-13. 形態1~5でのPSCn入力対PSCの動き



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

18.11. PSC入力形態6,7: PSC停止、ソフトウェアの動き待機

図18-14. 障害形態6,7でのPSCn入力対PSCの動き



注: ソフトウェア操作はPSC制御(PCTL)レジスタのPRUNEビットの設定(1)です。

障害形態6,7使用でのPSCn入力は沈黙時間0(DT0)/ON時間0(OT0)または沈黙時間1(DT1)/ON時間1(OT1)に関係なく動作します。

18.12. アナログ同期化

各PSC単位部はA/D変換器(ADC)の採取&保持を同期化するための信号を生成します。同期化は測定にとって必須です。

この信号はPSCOUTnAまたはPSCOUTnB出力の全ての上昇端または下降端から選ぶことができます。

中央整列動作では比較nRA(POCRnRA)レジスタが使われず、故にADCの同期指定に使うことができます。その場合の最小値は1です。

18.13. 割り込みの扱い

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

割り込み元一覧:

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

18.14. PSCクロック元

各PSCは2つのクロック入力を持ちます。

- PLLからのCLKPLL
- CLKI/O

PSC制御(PCTL)レジスタのPCLKSELビットはクロック元選択に使われます。

PSC制御(PCTL)レジスタのPPRE1,0ビットはクロックの分周係数選択に使われます。

図18-15. クロック選択

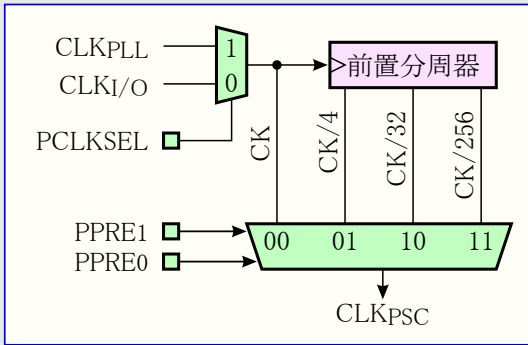


表18-6. クロック元と前置分周の選択

PCLKSEL	PPRE1	PPRE0	CLKPSC _n 出力
0	0	0	CLKI/O
0	0	1	CLKI/O/4
0	1	0	CLKI/O/32
0	1	1	CLKI/O/256
1	0	0	CLKPLL
1	0	1	CLKPLL/4
1	1	0	CLKPLL/32
1	1	1	CLKPLL/256

18.15. 割り込み

本項はATmega16M1/32M1/64M1で行なわれる割り込み動作の仕様を記述します。

18.15.1. 割り込みベクタの一覧

PSCは2つの割り込みベクタを提供します。

- PSC_EC (End of Cycle : 周期終了) : 許可で且つPOCR_RBとの一致で発生
- PSC_Fault (Fault Event : 障害事象) : 許可で且つ、PSC入力障害事象を検出した時に発生

18.15.2. ATmega16M1/32M1/64M1のPSC割り込みベクタ

表18-7. PSC割り込みベクタ

ベクタ番号	プログラム アドレス	発生元	備考
-	-	-	-
6	\$000A	電力段制御器 PSC_FAULT	電力段制御器(PSC) 異常発生
7	\$000C	電力段制御器0 PSC0_EC	電力段制御器(PSC) 周期終了(比較一致)
-	-	-	-

18.16. 電力段制御器(PSC)用レジスタ

レジスタはPSC単位部0について説明されます。これらは単位部1と単位部2に対して同じです。

18.16.1. POC – PSC出力構成設定レジスタ (PSC Output Configuration)

名称 : POC

変位 : \$B6

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	POEN2B	POEN2A	POEN1B	POEN1A	POEN0B	POEN0A
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5 – POEN2B : PSC単位部2 B部出力許可 (PSC Output 2B Enable)

値	説明
0	PSCOUT2Bに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT2Bに影響を及ぼすI/OピンはPSC単位部2波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

● ビット4 – POEN2A : PSC単位部2 A部出力許可 (PSC Output 2A Enable)

値	説明
0	PSCOUT2Aに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT2Aに影響を及ぼすI/OピンはPSC単位部2波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

● ビット3 – POEN1B : PSC単位部1 B部出力許可 (PSC Output 1B Enable)

値	説明
0	PSCOUT1Bに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT1Bに影響を及ぼすI/OピンはPSC単位部1波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

● ビット2 – POEN1A : PSC単位部1 A部出力許可 (PSC Output 1A Enable)

値	説明
0	PSCOUT1Aに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT1Aに影響を及ぼすI/OピンはPSC単位部1波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

● ビット1 – POEN0B : PSC単位部0 B部出力許可 (PSC Output 0B Enable)

値	説明
0	PSCOUT0Bに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT0Bに影響を及ぼすI/OピンはPSC単位部0波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

● ビット0 – POEN0A : PSC単位部0 A部出力許可 (PSC Output 0A Enable)

値	説明
0	PSCOUT0Aに影響を及ぼすI/Oピンは標準ポートとして動作します。
1	PSCOUT0Aに影響を及ぼすI/OピンはPSC単位部0波形生成器B出力へ接続され、PSC操作に従って設定(1)と解除(0)を行いません。

18.16.2. PSYNC – PSC同期構成設定レジスタ (PSC Synchro Configuration)

名称 : PSYNC
 変位 : \$B4
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	PSYNC21,0		PSYNC11,0		PSYNC01,0	
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5,4・3,2・1,0 – PSYNCn1,0 : 単位部n A/D変換器用同期出力選択 (Synchronization Out for ADC Selection)

同期化用に単位部nからA/D変換器へ送る信号を生成する信号元と極性を選びます。

動作形態	値	説明
1傾斜 形態	00	PSCOUTnAの先行端(OCRnSAとの比較一致)で信号送出。
	01	PSCOUTnAの後行端(OCRnRAとの比較一致またはA部での障害/再起動)で信号送出。
	10	PSCOUTnBの先行端(OCRnSBとの比較一致)で信号送出。
	11	PSCOUTnBの後行端(OCRnRBとの比較一致またはB部での障害/再起動)で信号送出。
中央整列 形態	00	(PSC下降計数中の)OCRnRAとの比較一致で信号送出。OCRnRAの最小値は1でなければなりません。
	01	(PSC上昇計数中の)OCRnRAとの比較一致で信号送出。OCRnRAの最小値は1でなければなりません。
	10	同期化信号なし。
	11	同期化信号なし。

18.16.3. POCRnSAH,POCRnSAL (POCRnSA) – PSCn比較SAレジスタ (PSC Output Compare SA Register)

注: n=単位部番号に対応する0~2

POCRnSAHとPOCRnSALのレジスタ対は16ビット値のPOCRnSAを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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

比較レジスタは16ビットと12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の16ビットまたは12ビットの全レジスタで共用されます。

名称 : POCR0SA : POCR1SA : POCR2SA
 変位 : \$A0 : \$A6 : \$AC
 リセット : \$0000
 特質 : -

ビット	15	14	13	12	11	10	9	8
	-	-	-	-	POCRnSA11~8			
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	6	5	4	3	2	1	0
	POCRnSA7~0							
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11~0 – POCRnSA11~0 : PSCn比較SA値 (PSC Output Compare SA Value)

18.16.4. POCRnRAH,POCRnRAL (POCRnRA) – PSCn比較RALレジスタ (PSC Output Compare RA Register)

注: n=単位部番号に対応する0~2

POCRnRAHとPOCRnRALのレジスタ対は16ビット値のPOCRnRAを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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

比較レジスタは16ビットと12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の16ビットまたは12ビットの全レジスタで共用されます。

名称 : POCR0RA : POCR1RA : POCR2RA

変位 : \$A2 : \$A8 : \$AE

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	POCRnRA11~8							
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	6	5	4	3	2	1	0
	POCRnRA7~0							
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11~0 – POCRnRA11~0 : PSCn比較RA値 (PSC Output Compare RA Value)

18.16.5. POCRnSBH,POCRnSBL (POCRnSB) – PSCn比較SBLレジスタ (PSC Output Compare SB Register)

注: n=単位部番号に対応する0~2

POCRnSBHとPOCRnSBLのレジスタ対は16ビット値のPOCRnSBを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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

比較レジスタは16ビットと12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の16ビットまたは12ビットの全レジスタで共用されます。

名称 : POCR0SB : POCR1SB : POCR2SB

変位 : \$A4 : \$AA : \$B0

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	POCRnSB11~8							
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	6	5	4	3	2	1	0
	POCRnSB7~0							
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11~0 – POCRnSB11~0 : PSCn比較SB値 (PSC Output Compare SB Value)

18.16.6. POOCR_RBH,POOCR_RBL (POOCR_RB) – PSC比較RBレジスタ (PSC Output Compare RB Register)

POOCR_RBHとPOOCR_RBLのレジスタ対は16ビット値のPOOCR_RBを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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

比較レジスタは16ビットと12ビットの容量です。CPUがこれらのレジスタに書く時に上位と下位のバイトが同時に書かれるのを保証するために、このアクセスは8ビット上位バイト一時レジスタ(TEMP)を使って実行されます。この一時レジスタは他の16ビットまたは12ビットの全レジスタで共用されます。

名称 : POOCR_RB

変位 : \$B2

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	POOCR_RB11~8							
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	6	5	4	3	2	1	0
	POOCR_RB7~0							
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11~0 – POOCR_RB11~0 : PSC比較RB値 (PSC Output Compare RB Value)

18.16.7. PCNF – PSC構成設定レジスタ (PSC Configuration Register)

名称 : PCNF

変位 : \$B5

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	PULOCK	PMODE	POPB	POPA	-	-
アクセス種別	R	R	R/W	R/W	R/W	R/W	R	R
リセット値	0	0	0	0	0	0	0	0

- ビット5 – PULOCK : PSC更新施錠 (PSC Update Lock)

このビットが設定(1)されると、比較(POCRnSA, POCRnRA, POCRnSB, POOCR_RB)レジスタ、PSC出力構成設定(POC)レジスタはPSC周期を妨げることなく書けます。PSC内部レジスタの更新はPULOCKビットが0に開放される場合に行なわれます。

- ビット4 – PMODE : PSC動作形態 (PSC Mode)

PSCの形態を選びます。

値	説明
0	1傾斜動作 (端整列)
1	中央整列動作

- ビット3 – POPB : PSC B出力極性 (PSC B Output Polarity)

値	説明
0	PSC単位部B出力はLow活性です。
1	PSC単位部B出力はHigh活性です。

- ビット2 – POPA : PSC A出力極性 (PSC A Output Polarity)

値	説明
0	PSC単位部A出力はLow活性です。
1	PSC単位部A出力はHigh活性です。

18.16.8. PCTL – PSC制御レジスタ (PSC Control Register)

名称 : PCTL
 変位 : \$B7
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	PPRE1,0		PCLKSEL	-	-	-	PCCYC	PRUN
アクセス種別	R/W	R/W	R/W	R	R	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7,6 – PPRE1,0 : PSC前置分周選択 (PSC Prescaler Select)

この2ビットはPSC入力クロック分周係数を選びます。全ての生成波形がこの係数によって変更されます。

値	説明
00	PSC入力クロックで分周器なし
01	PSC入力クロックを4分周
10	PSC入力クロックを32分周
11	PSC入力クロックを256分周

● ビット5 – PCLKSEL : PSC入力クロック選択 (PSC Input Clock Select)

このビットは高速クロックまたは低速クロックのどちらかを選びます。

値	説明
0	低速クロック(CLK _{I/O})を選ぶにはこのビットを解除(0)してください。
1	高速クロック(CLK _{PLL})を選ぶにはこのビットを設定(1)してください。

● ビット1 – PCCYC : PSC周期完了許可 (PSC Complete Cycle)

このビットが設定(1)されると、PSCはPRUNの解除(0)によって要求された停止操作に先立って波形周期全体を完了します。

● ビット0 – PRUN : PSC走行許可 (PSC Run)

このビットの1書き込みはPSCを始動します。

18.16.9. PMICn – PSC単位部n入力制御レジスタ (PSC Module n Input Control Register)

入力制御レジスタはPSCの2つ(A部とB部)の再起動/障害部の構成設定に使われます。この2つの部分は同一(構造)で、故に同じ方法で設定されます。

名称 : PMIC0 : PMIC1 : PMIC2
 変位 : \$B8 : \$B9 : \$BA
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	POVEN _n	PISEL _n	PELEV _n	PFLTE _n	PAOC _n	PRFM _{n2~0}		
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 – POVEN_n : PSC単位部n重複許可 (PSC Module n Overlap Enable)

重複保護を不活動にするにはこのビットを設定(1)してください。

● ビット6 – PISEL_n : PSC単位部n入力選択 (PSC Module n Input Select)

値	説明
0	障害/再起動部の入力としてPSCIN _n (ピン)を選ぶにはこのビットを解除(0)してください。
1	障害/再起動部の入力としてアナログ比較器 _n 出力を選ぶにはこのビットを設定(1)してください。

● ビット5 – PELEV_n : PSC単位部n入力レベル選択 (PSC Module n Input Level Selector)

値	説明
0	選んだ入力のLowレベルが障害機能に対して意味ある事象を生成します。
1	選んだ入力のHighレベルが障害機能に対して意味ある事象を生成します。

● **ビット4 – PFLTE_n : PSC単位部_n入力濾波器許可 (PSC Module _n Input Filter Enable)**

このビットの(1)設定は捕獲入力雑音消去器を活性(有効)にします。雑音消去器が活動(状態)にされると、再起動ピンからの入力は濾波されます。この濾波機能はその出力を変更するのに再起動ピンの4連続同一値採取を必要とします。従って捕獲入力は雑音消去器が許可される時に4発振器周期遅らされます。

● **ビット3 – PAOC_n : PSC単位部_n非同期出力制御 (PSC Module _n Asynchronous Output Control)**

このビットが解除(0)されると、障害入力はPSC単位部_nのAとBの出力を直接動かすことができます。

● **ビット2~0 – PRFM_{n2~0} : PSC単位部_n入力形態 (PSC Module _n Input Mode)**

これらの3ビットはPSC入力の動作形態を定義します。

値	説明
000	動きなし、PSC入力は無効にされます。
001	単位部 _n のA出力非活動
010	単位部 _n のB出力非活動
011	単位部 _n のAとBの出力非活動
10x	全PSC出力非活動
11x	PSC停止、ソフトウェアの動き待機

18.16.10. PIM – PSC割り込み許可レジスタ (PSC Interrupt Mask Register)

名称 : PIM
 変位 : \$BB
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	PEVE2	PEVE1	PEVE0	PEOPE
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 – PEVE2 : PSC単位部2外部事象割り込み許可 (PSC External Event 2 Interrupt Enable)**

このビットが設定(1)されると、単位部2で障害を生成し得る外部事象は割り込みも生成します。

● **ビット2 – PEVE1 : PSC単位部1外部事象割り込み許可 (PSC External Event 1 Interrupt Enable)**

このビットが設定(1)されると、単位部1で障害を生成し得る外部事象は割り込みも生成します。

● **ビット1 – PEVE0 : PSC単位部0外部事象割り込み許可 (PSC External Event 0 Interrupt Enable)**

このビットが設定(1)されると、単位部0で障害を生成し得る外部事象は割り込みも生成します。

● **ビット0 – PEOPE : PSC周期終了割り込み許可 (PSC End Of Cycle Interrupt Enable)**

このビットが設定(1)されると、PSCが周期全体(PSC周期)の最後に達する時に割り込みが生成されます。

18.16.11. PIFR – PSC割り込み要求フラグレジスタ (PSC Interrupt Flag Register)

名称 : PIFR
 変位 : \$BC
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	PEV2	PEV1	PEV0	PEOP
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 – PEV2 : PSC単位部2外部事象割り込み要求フラグ (PSC External Event 2 Interrupt)**

このビットは単位部2で障害を生成し得る外部事象が起こる時にハードウェアによって設定(1)されます。これはこの位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEVE2ビット=0)、読むことができます。

● **ビット2 – PEV1 : PSC単位部1外部事象割り込み要求フラグ (PSC External Event 1 Interrupt)**

このビットは単位部1で障害を生成し得る外部事象が起こる時にハードウェアによって設定(1)されます。これはこの位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEVE1ビット=0)、読むことができます。

- ビット1- PEV0 : PSC単位部0外部事象割り込み要求フラグ (PSC External Event 0 Interrupt)

このビットは単位部0で障害を生成し得る外部事象が起こる時にハードウェアによって設定(1)されます。これはこの位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEVE0ビット=0)、読むことができます。

- ビット0 - PEOPE : PSC周期終了割り込み要求フラグ (PSC End Of Cycle Interrupt)

このビットは”PSC周期の終り”発生時にハードウェアによって設定(1)されます。これはこの位置に1を書くことでソフトウェアによって解除(0)されなければなりません。このビットは例え対応する割り込みが許可されてなくても(PEOPEビット=0)、読むことができます。

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

19.1. 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

19.2. 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。

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間相互連結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SCK信号線に生成します。データは常にMOSI(Master Out Slave In)信号線を主装置から従装置へ、MISO(Master In Slave Out)信号線を従装置から主装置へ移動されます。各データパケット後、主装置はSS(従装置選択)ピンをHighへ引き上げることによって従装置と同期を取ります。

主装置として設定されると、SPIインターフェースにはSS信号線の自動制御がありません。これは通信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。

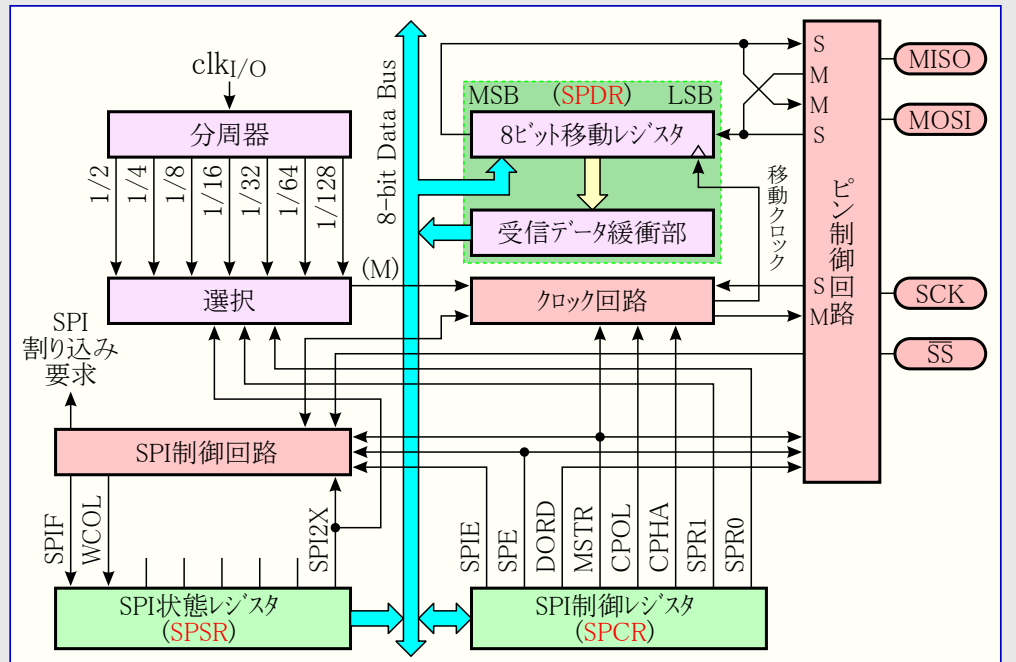
従装置として設定されると、SPIインターフェースはSSピンがHighに駆動される限り、MISOをHi-Zにした休止状態に留まります。この状態でプログラムはSPIデータレジスタ(SPDR)の内容を更新できますが、そのデータはSSピンがLowに駆動されるまでSCKピンでの到着クロックパルスによって移動出力されません。1バイトが完全に移動されてしまうと転送完了フラグ(SPIF)が設定(1)されます。SPCRでSPI割り込み許可(SPIE)ビットが設定(1)されていれば割り込みが要求されます。従装置は受信データを読む前にSPDR内へ送られるべき次のデータの配置を続けられます。最後の到着バイトはその後の使用のため、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19-1. 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。

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19-2. SPI 主装置/従装置の連結

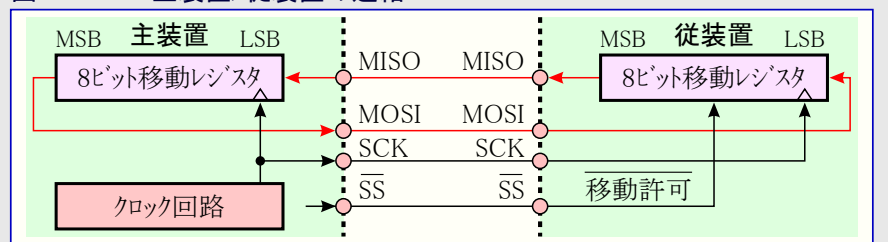


表19-1. SPIピン方向規定

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 [9頁の「コード例について」](#)

19.3. \overline{SS} ピンの機能

19.3.1. 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。

19.3.2. 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。

19.4. 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19-2. 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19-3. SPIデータ転送形式 (CPHA=0)

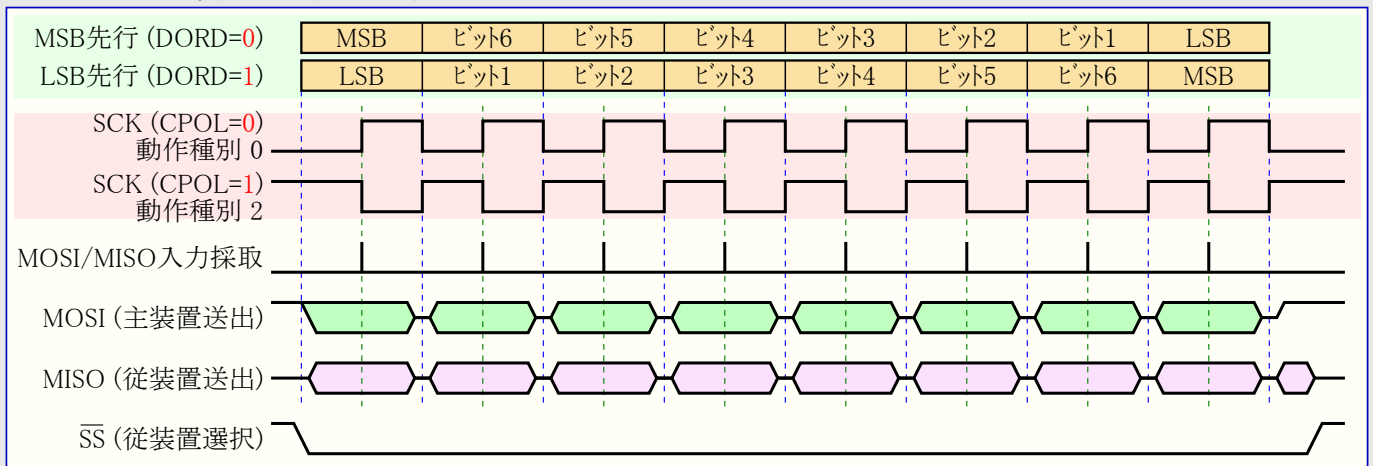
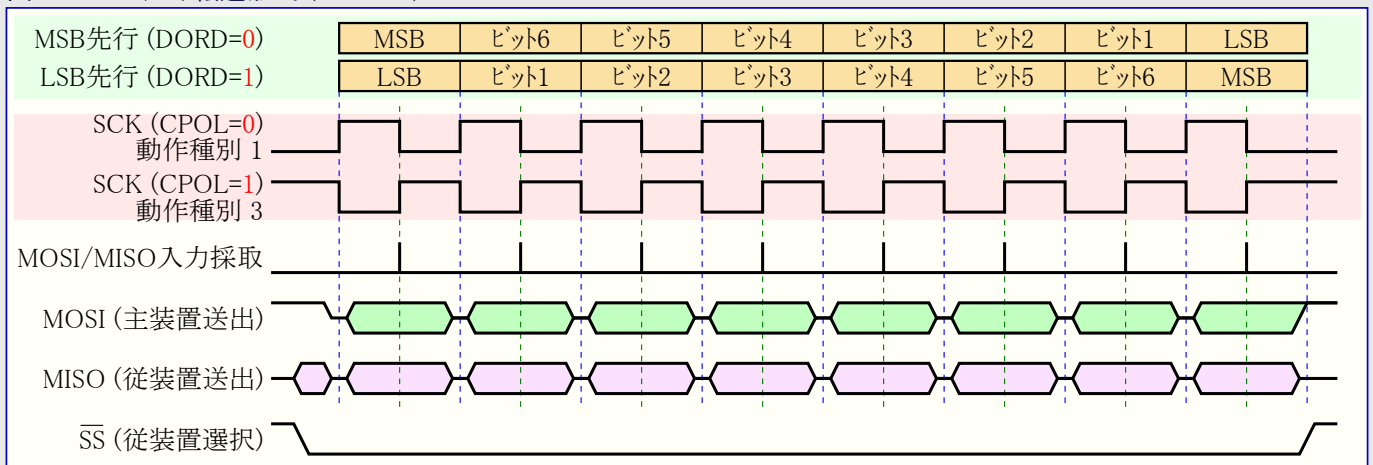


図19-4. SPIデータ転送形式 (CPHA=1)



19.5. SPI用レジスタ

19.5.1. 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
	SPIPS	–	–	PUD	–	–	IVSEL	IVCE
アクセス種別	R/W	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7 – SPIPS : SPIピン再割付 (SPI Pin Redirection)

MCUCRのSPIPS(SPI Pin Select:SPIピン選択)によってSPIピンは再割付することができます。プログラミングポートが常に代替SPIポートに配置されることに注意してください。

値	説明
0	SPI信号はMISO,MOSI,SCK,SSのピンに割り付けられます。
1	SPI信号はMISO_A,MOSI_A,SCK_A,SS_Aの代替ピンに割り付けられます。

19.5.2. 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(I)ビットが設定(1)されて、SPI状態レジスタ(SPSR)でSPI割り込み要求フラグ(SPIF)が設定(1)されると、このビットがSPI割り込みを実行させます。

● ビット6 – SPE : SPI許可 (SPI Enable)

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

● ビット5 – DORD : データ順序選択 (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19-3と図19-4を参照してください。CPOL機能は右で要約されます。

表19-3. 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19-3と図19-4を参照してください。CPHA機能は右で要約されます。

表19-4. CPHA機能動作

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

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

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

表19-5. SCKn速度選択 (fosc=CPUクロック周波数)

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

19.5.3. 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、その後にSPIデータレジスタ(SPDR)をアクセスすることによってもSPIFフラグは解除(0)されます。

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

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

● **ビット0 – SPI2X : SPIn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19-5参照)。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fosc(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。

19.5.4. 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)間のデータ転送に用いら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。

20. CAN – 制御器域網 (Controller Area Network)

20.1. 特徴

- 完全なCAN制御器
- CAN規格改訂2.0Aと改訂2.0Bに完全適合
- それら自体とでの6種のMOB(メッセージ目的項目)
 - 11ビットの識別票(改訂2.0A)、29ビットの識別票(改訂2.0B)
 - 11ビットの識別遮蔽(改訂2.0A)、29ビットの識別遮蔽(改訂2.0B)
 - 8バイトのデータ緩衝部(静的割り当て)
 - 送受信フレーム緩衝または自動応答設定
 - 時間印
- 8MHzでの1Mビット/s最大転送速度
- 時間起動通信(TTC)タイマ
- 聴取動作(調査または自動速度)

20.2. 概要

制御器域網(CAN)規約は非常に高い安全性での実時間、直列、広域通報規約です。Atmel ATmega16M1/32M1/64M1のCAN制御器はCAN仕様2.0A部と2.0B部に完全適合します。これはISO/OSI参照モードに従ったCANバス規約のカーネル実装に必要な機能を提供します。

- データリンク層
 - 論理リンク制御(LLC)補助層
 - 中間アクセス制御(MAC)補助層
- 物理層
 - 物理信号(PLS)補助層
 - 物理中間接合(PMA) – 未支援
 - 中間独立インターフェース(MDI) – 未支援

このCAN制御器は全フレーム形式(データ、リモート、エラー、オーバーロード)の扱いと1Mビット/sのビット速度を達成できます。

20.3. CAN規約

CAN規約は低速についてISO11519-2、高速についてISO11898で定義された万国標準です。

20.3.1. 原理

CANは広域通信機構に基きます。この広域通信はメッセージ指向送信規約の使用によって達成されます。これらのメッセージはメッセージ識別子の使用によって識別されます。メッセージ識別子のようなものは網全体内で唯一でなければならず、それは内容だけでなく、メッセージの優先順も定義します。

メッセージで送られた優先権は各メッセージの識別子によって指定される緊急性の低い他のメッセージと比較します。優先順は2進値に対応する形式でシステム設計中に制定され、動的に変更できません。最低2進数の識別子が最高優先順です。

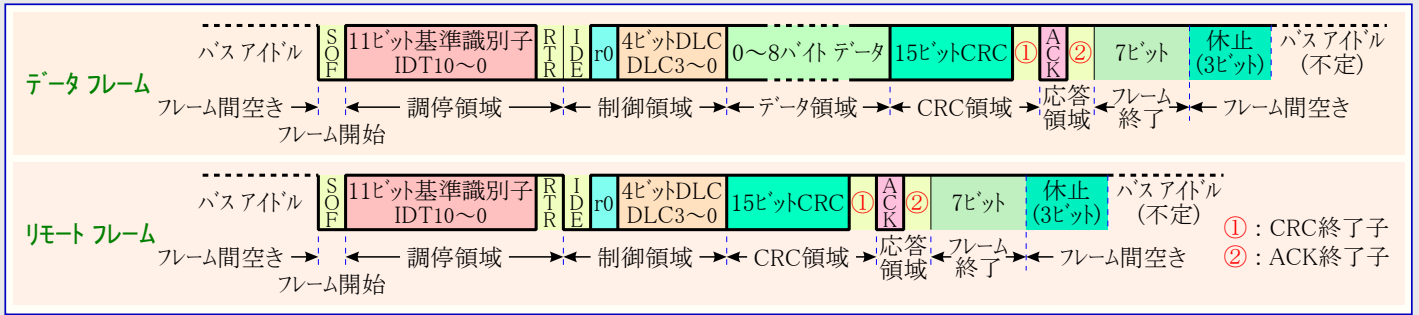
バス入出力の衝突は各節点でのビットに対するバス上のビットレベル監視による、混乱した識別子上のビット単位の調停によって解決されます。これは'優性'状態が'劣性'状態に上書きすることによる'ワイアードAND'構造によって起きます。バス割り当てに関する競合は劣性送信で且つ優性観測の全節点によって無くされます。全ての敗者(誤補:前行条件一致節点)は自動的に最高優先権メッセージの受信者となり、バスが再び利用可能になるまで再送信を試みてはなりません。

20.3.2. メッセージ形式

CAN規約は本質的に識別子の長さだけが異なる2つのメッセージフレーム形式を支援します。CAN2.0Aとしても知られるCAN標準フレームは識別子に関して11ビット長を支援し、CAN2.0Bとしても知られるCAN拡張フレームは識別子に関して29ビット長を支援します。

20.3.2.1. CAN標準フレーム

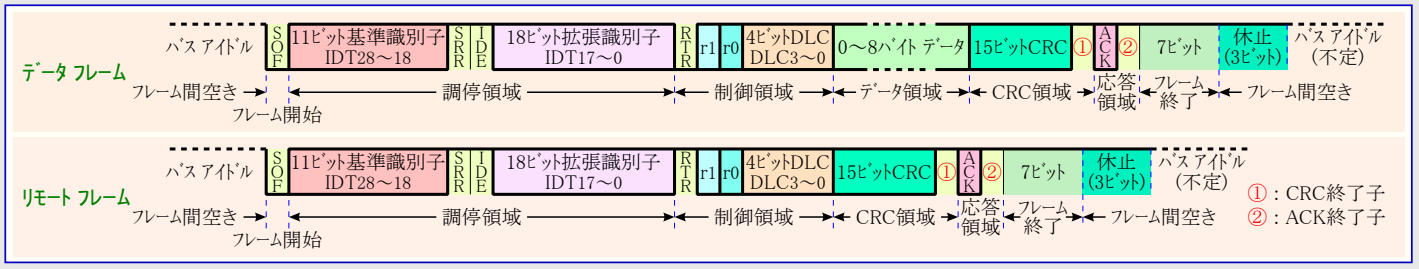
図20-1. CAN標準フレーム



CAN標準フレーム形式でのメッセージはフレーム開始(SOF:Start Of Frame)で始まり、これには識別子と、リモートフレームと呼ばれるデータ要求フレームとデータフレームとを区別するのに使う間接(リモート)送信要求(RTR:Remote Transmission Request)ビットから成る調停領域が後続します。後続する制御領域は識別子拡張(IDE:Identifier Extension)ビットと後続するデータ領域内のデータバイト数を示すのに使うデータ長符号(DLC:Data Length Code)を含みます。リモートフレームでのDLCは要求するデータバイト数を含みます。後続のデータ領域は8バイトまでのデータを保持できます。フレームの保全は後続する巡回冗長検査(CRC:Cyclic Redundant Check)計算によって保証されます。応答領域はACKスロットとACK終了子から成ります。ACKスロットのビットは**劣性**として送信され、この時に正しくデータを受けた受信装置によって**優性**として書き込まれます。正しいメッセージは受け入れ検査の結果に拘らず、受信装置によって(肯定)応答されます。メッセージの終了はフレーム終了(EOF:End Of Frame)によって示されます。フレーム間空き(IFS:Intermission Frame Space)は継続メッセージを分離する最小ビット数です。どの節点によっても後続するバスアクセスがないなら、バスはアイドルに留まります。

20.3.2.2. CAN拡張フレーム

図20-2. CAN拡張フレーム



CAN拡張フレーム形式でのメッセージはCAN標準フレーム形式でのメッセージと殆ど同じです。違いは使う識別子の長さです。この識別子は既存の11ビット識別子(基準識別子)と18ビットの拡張(拡張識別子)で作られます。CAN標準フレーム形式とCAN拡張フレーム形式の区別はCAN標準フレーム形式の場合に**優性**、他の場合に**劣性**として送信されるIDEビットの使用によって行われます。

20.3.2.3. フレーム形式の同時共存

2つの形式は1つのバス上に同時共存しななければならず、それは同じ識別子/基準識別子で違う形式でのバスアクセス衝突の場合、バス上でより高い優先権を持つメッセージにされます。CAN標準フレーム形式のメッセージは拡張形式のメッセージに対して常に優先権を持ちます。

CAN部で利用可能な3つの異なる種類は次のとおりです。

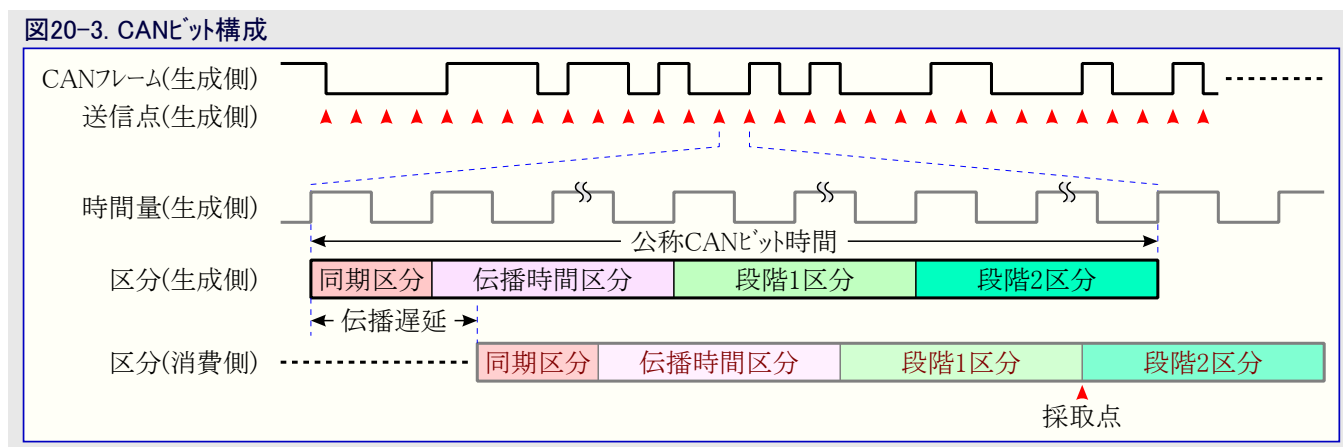
- 2.0A 29ビットIDを異常として考慮
- 2.0B受動 29ビットID無効
- 2.0B能動 11ビットと29ビットIDメッセージの取り扱い

20.3.3. CANビットタイミング

最終ビットまでの正しい採取を保証するためにCAN節点はフレーム全体を通して再同期を行う必要があります。これはSOFの下降端での各メッセージの開始と劣性から優性への各端で行われます。

20.3.3.1. ビット構成

1つのCANビット時間は4つの非重複区分として記されます。各区分は時間量の整数倍で構成されます。時間量(TQ:Time Quantum)はCAN節点で使われる最小独立タイミングです。



20.3.3.2. 同期区分

最初の区分は様々なバス節点と同期するために使われます。

送信に於ける本区分の開始で現在のビットレベルが出力されます。直前ビットと現在ビット間でビット状態変化があるなら、その後に受信する節点によって本区分内でのバス状態変化の発生が予測されます。

20.3.3.3. 伝播時間区分

本区分は網を渡る信号遅延に対する補償に使われます。

これはバス節点の送信部を通り、バス信号線の信号伝播遅延に関する保障に必要です。

20.3.3.4. 段階1区分 (PS1)

段階1区分は端(エッジ)段階異常に対する補償に使われます。

この区分は再同期中に延長されるかもしれません。

20.3.3.5. 採取点

採取点はバスレベルが読まれ、各ビットの値として判断される時の位置です。この位置は段階1区分の最後(2つの段階区分の間)です。

20.3.3.6. 段階2区分 (PS2)

本区分は端(エッジ)段階異常に対する補償に使われます。

この区分は再同期中に短縮されるかもしれませんが、この長さは少なくとも情報処理時間(IPT:Information Processing Time)でなければならず、段階1区分の長さよりも長くはできません。

20.3.3.7. 情報処理時間 (IPT)

これは採取したビットのビットレベルを決める論理回路に関して必要な時間です。

IPTは採取点で始まり、時間量(TQ)で計測され、AtmelのCANに関しては2TQで確定されます。段階2区分(PS2)も採取点で始まり、これがビット時間の最終区分なので、最小PS2がIPTよりも短くなることはありません。

20.3.3.8. ビット延長

再同期の結果として発振器誤差に対する補償のために段階1区分が延長されるか、または段階2区分が短縮されるかもしれません。例えば送信側発振器が受信側発振器よりも遅い場合、再同期に対して使う次の下降端は遅れるかもしれません。故に採取点とビット時間の最後を補正するために段階1区分が延長されます。

20.3.3.9. ビット短縮

これに反して送信側発振器が受信側よりも速い場合、再同期に対して使う次の下降端は早すぎるかもしれません。故にビットN+1に対する採取点とビット時間の最後を補正するためにビットNの段階2区分が短縮されます。

20.3.3.10. 同期飛躍幅

段階区分の延長または短縮の量の限度は再同期飛躍幅によって設定されます。

この区分は段階2区分よりも長くできません。

20.3.3.11. 採取点設定

採取点の設定(プログラム)はバスに合わせた特性の調整を許します。

容易な採取は段階2区分でのより多くの時間量(TQ)、故に同期飛躍幅が最大に設定できるのを許します。ビット時間の延長または短縮に関して、この最大量は節点発振器誤差への感度を減少し、従ってセラミック振動子のような低価格発振器が使えます。

遅延採取はより貧弱なバス接続形態と最大バス長を許す伝播時間区分内で、より多くの時間量(TQ)を許します。

20.3.3.12. 同期

確実な同期は開始ビットの劣性から優性への遷移で起きます。ビット時間はこの端(エッジ)から改めて開始されます。

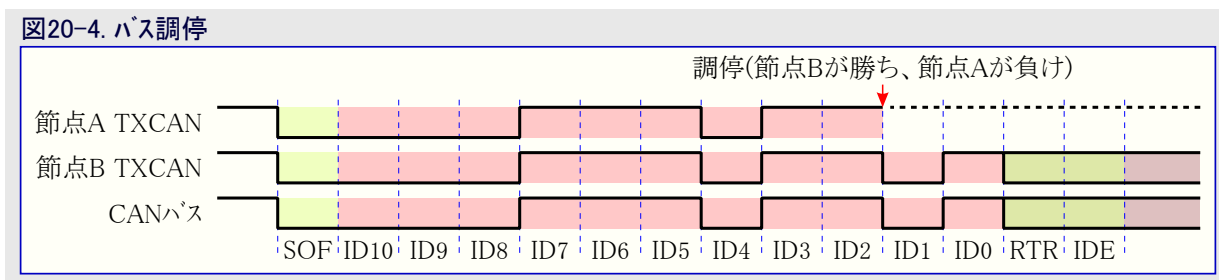
再同期はメッセージ内の同期区分で劣性から優性への端が起きない時に起こります。

20.3.4. 調停

CAN規約はメッセージ優先権調停付き搬送波監視多重入出力(CSMAAMP:Carrier Sense Multiple Access with Arbitration on Message Priority)と呼ばれる概念に従うバス入出力を扱います。

送信中、CANバスでの調停はより高い優先権のCAN識別子を持つデバイスとの競争で負け得ます。この調停の概念は複数節点によって同時に開始された送信メッセージの衝突をなくし、最も重要なメッセージが時間損失なしで最初に送られるのを保証します。

バス入出力の衝突は殆どが識別子の値に対する調停領域間に解決されます。同じ識別子を持つデータフレームとリモートフレームが同時に開始される場合、リモートフレームに対してデータフレームが勝ちます(RTRビット参照)。



20.3.5. 異常

CAN規約は何れかの異常が起きると、直ちにそれを知らせます。メッセージ段階での3つとビット段階での2つの異常検出機構が実装されています。

20.3.5.1. メッセージ段階での異常

• 巡回冗長検査 (CRC)

CRCは送信の最後での冗長検査ビット付加によってフレーム内の情報を保護します。受信側でこれらのビットは再計算され、受信したビットに対して検査されます。それらが一致しなければCRC異常になってしまいます。

• フレーム検査

この機構は固定形式に対するビット領域とフレーム容量の検査によって送信されたフレーム構造を照合します。フレーム検査によって検出した異常は「形式異常」と呼ばれます。

• ACK異常

既に記述したように受信したフレームは肯定応答を通して全ての受信装置によって応答されます。メッセージの送信装置によって応答が受信されない場合、ACK異常が示されます。

20.3.5.2. ビット段階での異常

• 監視

送信側の異常検出能力はバス信号の監視に基きます。各送信節点はバスレベルも監視し、従って送信ビットと受信ビット間の違いを検出します。これは信頼できる広域的な異常と送信側への局所的な異常の検出を可能にします。

• ビット要素

個別ビットの符号化はビットレベルで検査されます。CANで使うビット表現はビット符号化で最大効率を保証する0復帰なし(NRZ: Non Return to Zero)符号化です。同期端(エッジ)はビット要素の意味によって保証されます。

20.3.5.3. 異常の合図

上記の機構を使う最低1つの節点で1つまたは複数の異常が発見されると、現在の送信は「異常フラグ」の送出によって中断されます。これはメッセージを受け入れている他の節点を保護し、従って網全体でデータの一貫性を保証します。誤ったメッセージの送信が中断された後、送信側は自動的に送信を再試行します。

20.4. CAN制御器

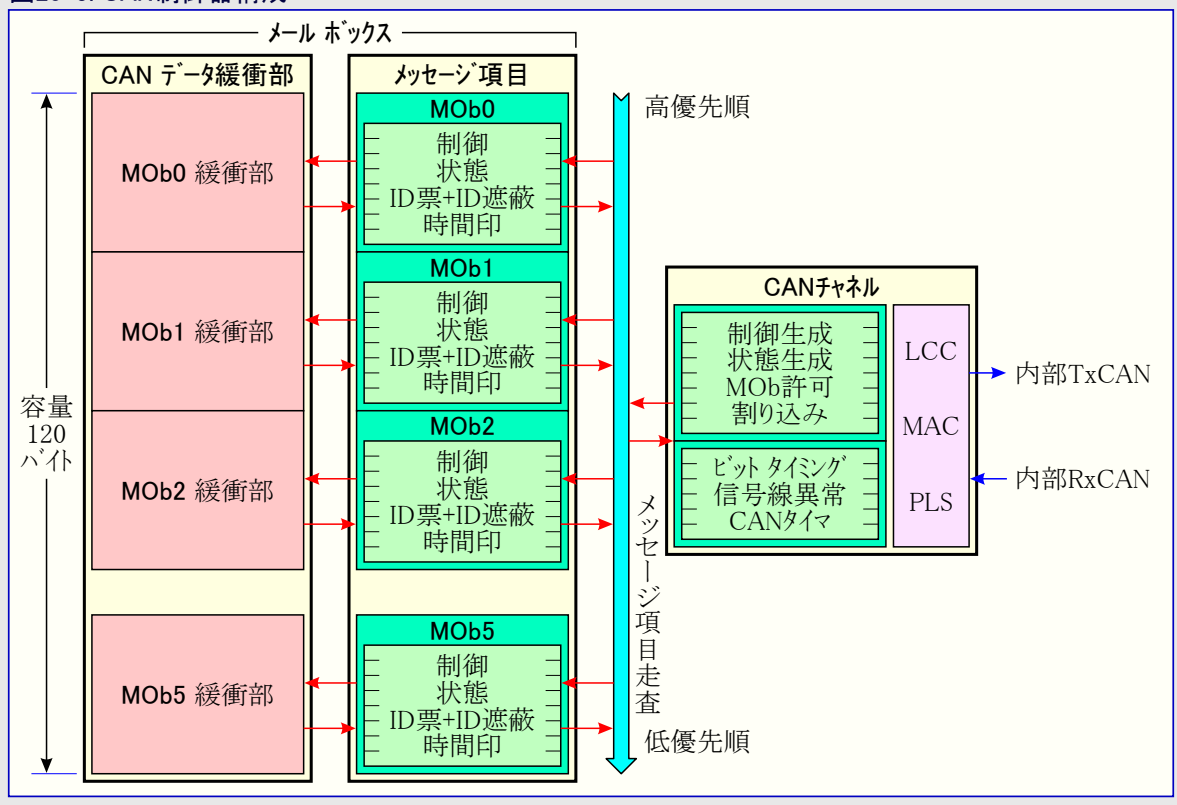
Atmel ATmega16M1/32M1/64M1内に実装されたCAN制御器は改訂2.0B能動を提供します。

この完全CAN制御器は便利な受容濾過とメッセージ管理に対して完全なハードウェアを提供します。送受信されるべき各メッセージに対し、この単位部はメッセージに関する全情報(例えば識別子、データバイトなど)が格納されるメッセージ項目(Message object)と呼ばれるようなものを含みます。

周辺機器の初期化の間、応用(ソフトウェア)はどれが送るべきメッセージなのか、どれが受信されるべきなのかを定義します。CAN制御器が設定した(受信)メッセージ項目の識別子の1つと一致するメッセージを受信するだけなら、そのメッセージが格納され、応用(ソフトウェア)は割り込みによって通知されます。他の優位点は到着リモートフレームが完全CAN制御器によって対応するデータフレームで自動的に応答できることです。この方法でのCPU負荷は基本的なCAN処理法に比べて大きく軽減されます。

完全CAN制御器の使用は高レートと多くのメッセージでの高いバス負荷を扱えます。

図20-5. CAN制御器構成



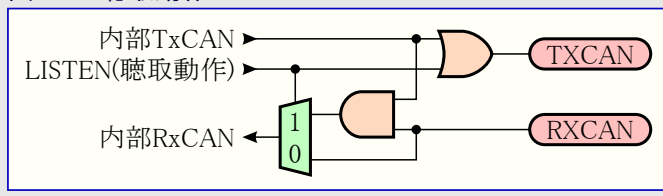
20.5. CANチャンネル

20.5.1. 形態

CANチャンネルは以下にできます。

- 許可動作
 - この動作では
 - CANチャンネル(内部TxCANとRxCAN)が許可されます。
 - 入力クロックが許可されます。
- 待機動作
 - 待機動作では
 - 送信部が定期的に(内部TxCAN上に)劣性レベルを供給し、受信部は禁止されます。
 - 入力クロックが許可されます。
 - ページとレジスタはアクセス可能のままです。
- 聴取動作
 - この動作はCANチャンネルに対して透過です。
 - ハードウェア短絡帰帰(内部RxCANに内部TxCAN)を許可。
 - TxCAN出力ピンに劣性レベルを供給します。
 - RxCAN入力ピンを禁止しません。
 - TECとRECの異常計数器を凍結(一時停止)します。

図20-6. 聴取動作



20.5.2. ビット タイミング

CANチャネルの有限状態機構(FSM:Finite State Machine)は時間量(TQ)に同期する必要があります。故にビット タイミングに対する入力クロックはCANチャネルのFSMに使うクロックです。

領域と区分の略号は次のとおりです。

- BRP : ボーレート前置分周器
- TQ : 時間量(ボーレート前置分周器出力)
- SYNS: 1TQ長の同期区分
- PRS : 伝播時間区分は1~8TQ長に設定可能です。
- PHS1: 段階1区分は1~8TQ長に設定可能です。
- PHS2: 段階2区分はIPT~PHS1のTQ長に設定可能です。
- IPT : 情報処理時間は2TQです。
- SJW : (再)同期飛躍幅は1から、4またはPHS1の小さい方の値までが設定可能です。

ビット時間内の総TQ数は最低の8から25までで設定されなければなりません。

図20-7. 採取点と送出点

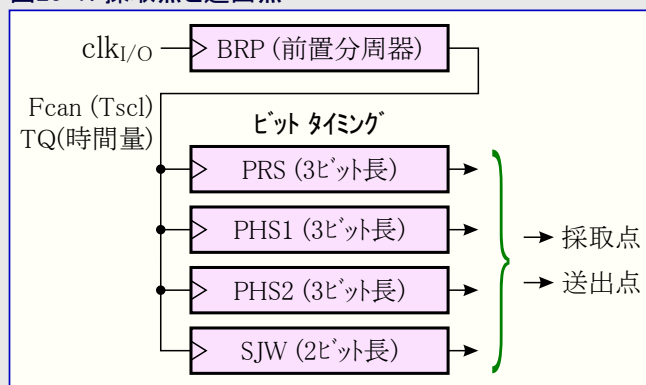
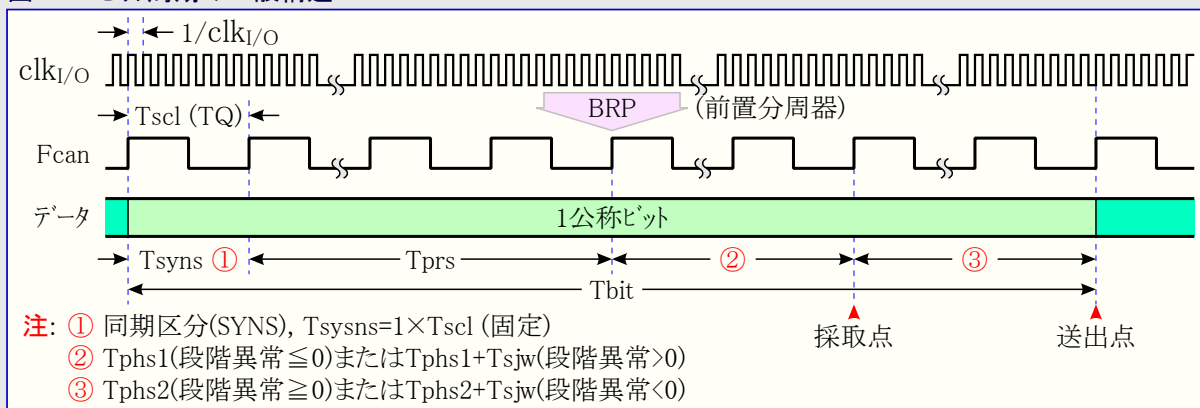


図20-8. ビット周期の一般構造



20.5.3. ボーレート

ボーレート前置分周器なしでの採取点は1時間量(TQ)速くなります。これはISO16845に従った試験法の失敗を引き起こします。これには1時間量(TQ)長くした段階1区分と、補償のために2時間量(TQ)短くした段階2区分が必要です。

ボーレート選択はTbit計算によって行われます。

$$T_{bit} = T_{syns} + T_{prs} + T_{phs1} + T_{phs2} \quad (\text{注1})$$

- $T_{syns} = 1 \times T_{sc1} = (BRP5 \sim 0 + 1) \div clk_{I/O} = 1TQ$
- $T_{prs} = (1 \sim 8) \times T_{sc1} = (PRS2 \sim 0 + 1) \times T_{sc1}$
- $T_{phs1} = (1 \sim 8) \times T_{sc1} = (PHS12 \sim 0 + 1) \times T_{sc1}$
- $T_{phs2} = (1 \sim 8) \times T_{sc1} = (PHS22 \sim 0 + 1) \times T_{sc1}$ (注2)
- $T_{sjw} = (1 \sim 4) \times T_{sc1} = (SJW1, 0 + 1) \times T_{sc1}$

注1: ビット時間内の総Tsc1(TQ)数は8~25でなければなりません。

注2: PHS22~0は1~PHS12~0間に設定可能です。

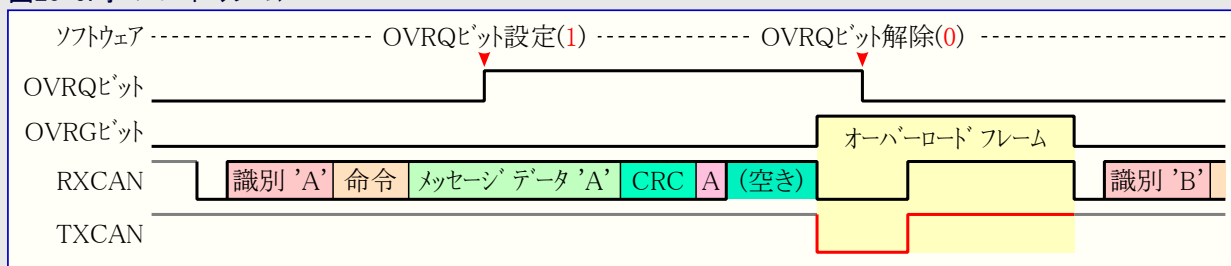
20.5.4. 失敗制限

「異常管理」をご覧ください。

20.5.5. オーバーロード フレーム

オーバーロード フレームはオーバーロード要求(OVRQ)の設定(1)によって送出されます。次の受信後、CANチャネルはCAN仕様に従ってオーバーロード フレームを送信します。状態またはフラグはオーバーロード フレームが送信されている限り、(OVRQが)設定(1)されます。

図20-9. オーバーロード フレーム



20.6. メッセージ項目

メッセージ項目(MOb:Message Objects)はCANフレーム記述子です。それはCANフレームが扱う全情報を含みます。これはMOBがCANメッセージを項目のように記述するのを許すために略記されることを意味します。MOBの組はソフトウェアの作業負荷を軽減するだけでなく、送受信のためのメッセージが予め定義されるメールボックスの前置部です。

MOBは独立していますが、複数一致の場合、下位順に優先権が与えられます。動作種別は次のとおりです。

- 禁止動作
- 送信動作
- 受信動作
- 自動応答動作
- フレーム緩衝受信動作

20.6.1. MOB数

本デバイスは6個のMOBを持ち、これらは0から5まで番号付けされます。

20.6.2. 動作種別

リセット後の既定動作はありません。

全てのMOBは動作種別を制御するために自身の領域を持ちます。CAN周辺機能を許可する前に各MOBは構成設定されなければなりません(例えば禁止動作, CONMOB1,0=00)。

表20-1. MOB構成設定

MOB構成	応答有効	RTRTAG	動作種別	
0	0	x	x	禁止
0	1	x	0	送信データ フレーム
		x	1	送信リモート フレーム
1	0	x	0	受信データ フレーム
		0	1	受信リモート フレーム
		1		受信リモートフレーム後、送信データ フレーム(応答)
1	1	x	x	フレーム緩衝受信動作

20.6.2.1. 禁止

この動作でのMOBは自由(未使用)です。

20.6.2.2. 送信データ & リモート フレーム

- 各種領域が送信前に初期化されなければなりません。
 - 識別票 (IDT)
 - 識別子拡張 (IDE)
 - リモート送信要求 (RTRTAG)
 - データ長符号 (DLC)
 - 予約ビット値 (RBnTAG)
 - メッセージのバイト データ (MSG)
- MOBはMOB構成設定(CONMOB1,0)が設定されると、データまたはリモート フレーム送信の準備が整います。
- そしてCANチャネルは送信設定の全MOBを走査し、最上位優先権を持つMOBを探して、その送信を試みます。
- 送信が完了すると、TXOKフラグが設定(1)されます(割り込み)。
- 全てのパラメータとデータは新規初期化までMOBで利用可能です。

20.6.2.3. 受信データ & リモート フレーム

- 各種領域が受信前に初期化されなければなりません。
 - 識別票 (IDT)
 - 識別子遮蔽 (IDMSK)
 - 識別子拡張 (IDE)
 - 識別子拡張遮蔽 (IDEMSK)
 - リモート送信要求 (RTRTAG)
 - リモート送信要求遮蔽 (RTRMSK)
 - データ長符号 (DLC)
 - 予約ビット値 (RBnTAG)
- MOBはMOB構成(CONMOB1,0)が設定されると、データまたはリモート フレーム受信の準備が整います。
- CAN網でフレーム識別子が受信されると、CANチャネルは受信設定の全MOBを走査し、一致する最上位優先権を持つMOBの検索を試みます。
- 見つけると、一致したMOBのIDT,IDE,DLCが到着(フレーム)値で更新されます。
- 一旦受信が完了すると、(リモート フレームを除いて)受信したメッセージのバイト データが一致したMOBのデータ緩衝部内に格納され、RXOKフラグが設定(1)されます(割り込み)。
- 全てのパラメータとデータは新規初期化までMOBで利用可能です。

20.6.2.4. 自動応答

リモート フレームへの(データ フレーム)応答は期待したリモート フレームの受信後、自動的に送信できます。

1. 各種領域がリモート フレーム受信前に初期化されなければなりません。
 - 「受信データ & リモート フレーム」で説明されたものと同じ流れでの**応答有効(RPLV)ビット**
2. リモート フレームが一致すると、**リモート送信要求(RTRTAG)**と**応答有効(RPLV)ビット**は自動的に解除(0)されます。この時にフラグ(と割り込み)は設定(1)されません。CANデータ緩衝部が到着リモート フレームによって使われないため、MOBはそれ以上の設定なしで送信動作での準備ができています。受信したリモート フレームのIDT,IDE,その他の識票とDLCは応答に使われます。
3. 応答送信が完了すると、**TXOKフラグ**が設定(1)されます(割り込み)。
4. 全てのパラメータとデータは新規初期化までMOBで利用可能です。

20.6.2.5. フレーム緩衝受信動作

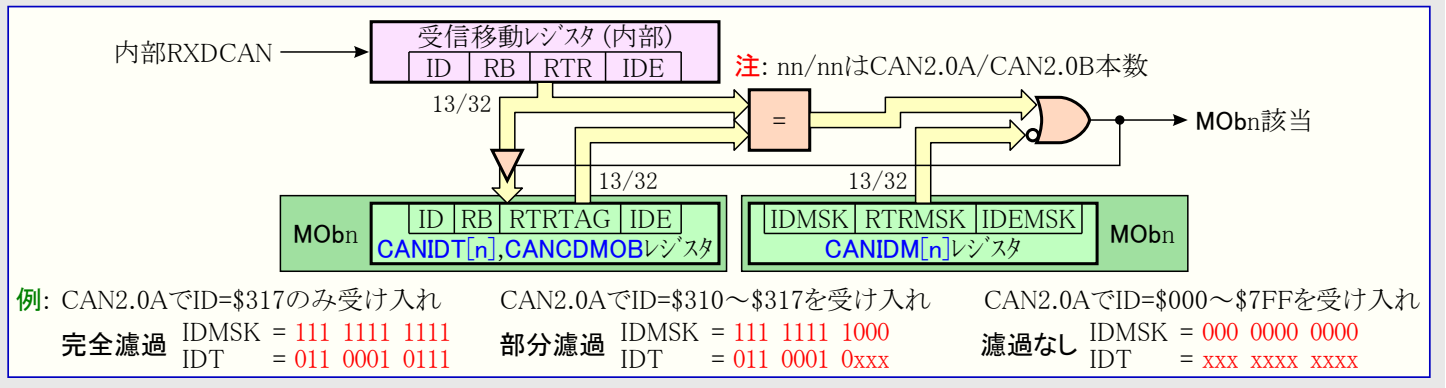
この動作は複数フレーム受信に有用です。MOB間の優先順位はこれらの到着フレームに対する管理を提供します。MOBがこの動作に設定されると、(未使用MOBを含めて)1組のMOBが作成されます。この動作の設定のため、1組だけが利用可能です。フレーム緩衝完了(BXOK)フラグ(または割り込み)は組内の全MOBがそれら用のCANフレームを受信してしまった時にだけ設定(1)されます。

1. フレーム緩衝受信動作のMOBは標準受信動作でのMOBのように初期化される必要があります。
2. MOBはそれら各々の**MOB構成設定(CONMOB1,0)**が設定されると、データ(またはリモート)フレーム受信の準備が整います。
3. CAN網でフレーム識別子が受信されると、CANチャネルは受信設定の全MOBを走査し、一致する最上位優先権を持つMOBの検索を試みます。
4. 見つけると、一致したMOBのIDT,IDE,DLCが到着(フレーム)値で更新されます。
5. 一旦受信が完了すると、(リモート フレームを除いて)受信したメッセージのバイト データが一致したMOBのデータ緩衝部内に格納され、**RXOKフラグ**が設定(1)されます(割り込み)。
6. 組内の最後のMOBの受信が完了すると、**フレーム緩衝完了(BXOK)フラグ**が設定(1)されます(割り込み)。BXOKフラグは組内の全CONMOB1,0領域が再書き込みされる場合にだけ解除(0)され得ます。
7. 全てのパラメータとデータは新規初期化までMOBで利用可能です。

20.6.3. 受容濾過

受信該当(換言すれば、比較遮蔽を考慮する一方で指定したIDT,RTRTAG,RBnTAG,IDEと受信したID,RTR,RBn,IDE間の完全な比較一致)で、MOB内のIDT,RTRTAG,RBnTAG,IDEは受信値で更新されます(レジスタ書き)。

図20-10. 受容濾過部構成図



20.6.4. MOBページ

全てのMOBは場所を節約するためにページに割り付けられます。このページ番号はMOB番号です。このページ番号はCANページMOB (CANPAGE)レジスタで設定されます。番号15は工場検査用に予約されています。

最優先MOB(CANHPMOB)レジスタはMOB割り込み状態(CANSIT1,2)レジスタ内で最高優先権を持つMOBを与えます。これはCANPAGEレジスタに対する直接入力(設定)を提供するために構成されています。CANHPMOBはCANSIT1,2レジスタを符号化するため、対応する許可ビット(ENRX,ENTX,ENERR)が許可される場合にだけ更新されます。図20-14をご覧ください。

20.6.5. CAN データ緩衝部

レジスタ位置を保つため、CANデータ緩衝部はMOB選択で(アドレス ポインタでアクセス可能な)FIFOのようなものに見えます。これは未制御アクセスの危険の減少も可能にします。

MOB毎に1つのFIFOです。このFIFOはCANメッセージ レジスタによってMOBページでアクセスされます。

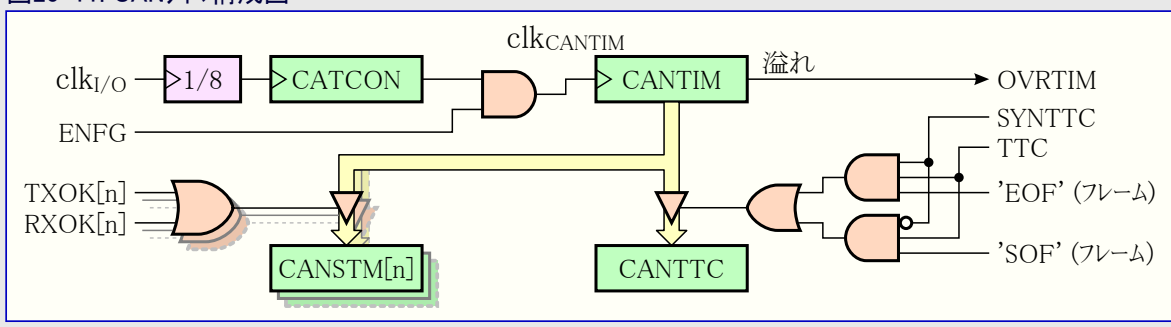
FIFO CANデータ緩衝部指示子(INDX2~0)は欲したバイト データへのアドレス ポインタです。このバイト データは読み書きできます。データ指示子はFIFO CANデータ緩衝部指示子自動進行(AINC)ビットが解除(0)なら、毎アクセス後、自動的に進行されます。転回が実装され、データ指示子=7後はデータ指示子=0です。

CANフレームの最初のバイトはデータ指示子=0に格納され、2番目はデータ指示子=1、以下同様です。

20.7. CANタイマ

設定可能な16ビット タイマはメッセージ印と時間起動通信(TTC:Timer Trigger Communication)に使われます。

図20-11. CANタイマ構成図



20.7.1. 前置分周器

8ビット前置分周器はCANタイマ制御(CANTCON)レジスタによって初期化されます。それは8分周したclkI/O周波数を受けます。CAN制御器が許可されているなら、それはCANタイマにclkCANTIM周波数を供給します。

$$Tclk_{CANTIM} = Tclk_{I/O} \times 8 \times (CANTCON7 \sim 0 + 1)$$

20.7.2. 16ビット タイマ

このタイマはCAN制御器が許可される(ENFG=1)時に\$0000から計数を開始します。このタイマが\$FFFFから\$0000へ転回する時に割り込み(OVRTIM)が生成されます。

20.7.3. 時間起動

2つの同期化動作がTTC(CAN総合制御(CANGCON)レジスタの時間起動通信(TTC)ビット)に対して実装されます。

- フレーム開始(SOF)での同期 (CANGCONレジスタのTTC同期化(SYNTTC)ビット=0)
- フレーム終了(EOF)での同期 (CANGCONレジスタのSYNTTCビット=1)

TTC動作でのフレームは例え異常が発生しても、一度送信されます。

20.7.4. メッセージ刻印

このタイマ値の捕獲はフレームを送受信するMOBで行われます。管理する全てのMOBが刻印され、受信(送信)したフレームの刻印はRXOK(TXOK)で起こります。

20.8. 異常管理

20.8.1. 失敗限界

CANチャネルは以下の3状態の1つになるかもしれません。

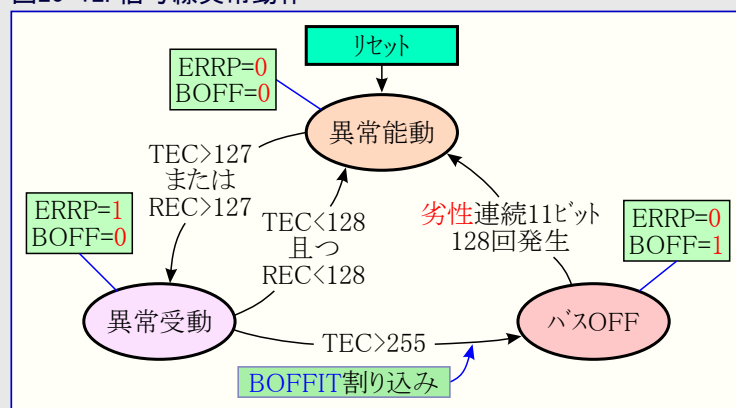
- 異常能動 (既定)

CANマクロ(セル)が異常を検出する時にCANチャネルはバス通信を離れて能動異常フレームを送れます。
- 異常受動

CANチャネルは能動異常フレームを送れません。それはバス通信を離れますが、異常検出時に受動異常フレームが送られます。また、送信後に異常受動部は更なる送信を始める前に待機します。
- バスOFF

CANチャネルはバスに対してどんな影響を与えることも許されません。

図20-12. 信号線異常動作



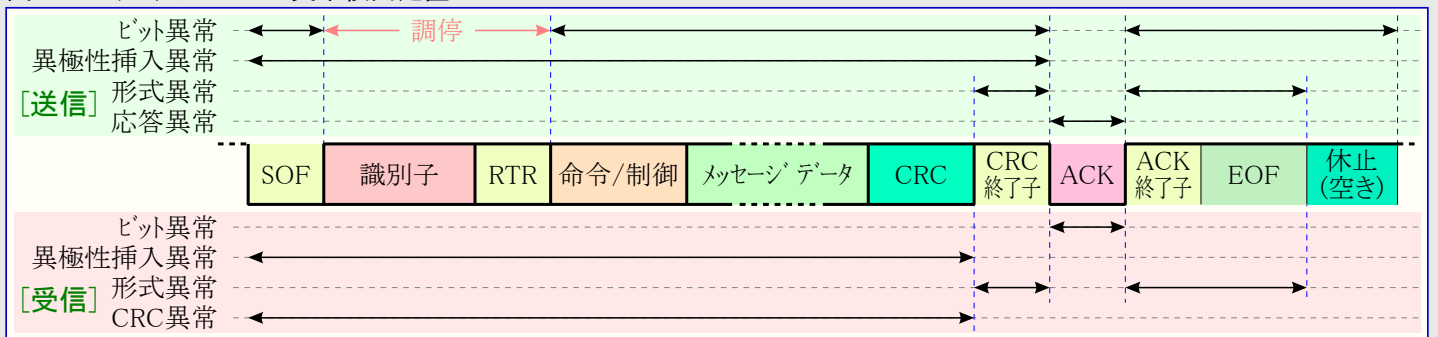
注: 与えられたメッセージ転送中に1つよりも多くのREC/TEC変更が加えられるかもしれません。

失敗限度に関して送信異常計数器(CANTEC)と受信異常計数器(CANREC)が実装されています。CAN総合状態(CANGSTA)レジスタのバスOFF動作(BOFF)と異常受動動作(ERRP)ビットはCANチャネルの状態情報を与えます。BOFFの1設定は割り込みを生成するかもしれません。

20.8.2. 異常種別

- **BERR**: ビット異常。監視したビット値が送ったビット値と違います。
 - 注: 例外:
 - 調停領域と応答スロット中に**優性**ビットとして監視した**劣性**送出ビット
 - 異常フレーム送出中の**優性**ビット検出
- **SERR**: 異極性挿入異常。同一極性で5ビットを越える連続を検出。
- **CERR**: CRC異常(受信のみ)。受信部は完全に異極性挿入解除をした受信メッセージのフレーム開始からデータ領域まででCRC検査を実行します。この検査が異極性挿入解除をしたCRC領域と不一致の場合にCRC異常が設定(1)されます。
- **FERR**: 形式異常。形式異常は以下のビット領域の固定形式の1つ以上の違反の結果です。
 - CRC終了子
 - 応答終了子
 - フレーム終了
 - 異常終了子
 - オーバーロード終了子
- **AERR**: 応答異常(送信のみ)。応答スロットで**優性**ビット未検出。

図20-13. データフレームでの異常検出处置



20.8.3. 異常設定

CANチャネルはCAN網上でいくつかの異常を検出できます。

- 送信側:
 - 異常はMOB段階で設定されます。
- 受信側:
 - 識別一致後:
 - 異常はMOB段階で設定されます。
 - 識別不一致または未だ不一致:
 - 異常は総合段階で設定されます。

異常検出後、CANチャネルは網上に異常フレームを送出します。CANチャネルが異常フレームを検出すると、自身の異常フレームを送出します。

20.9. 割り込み

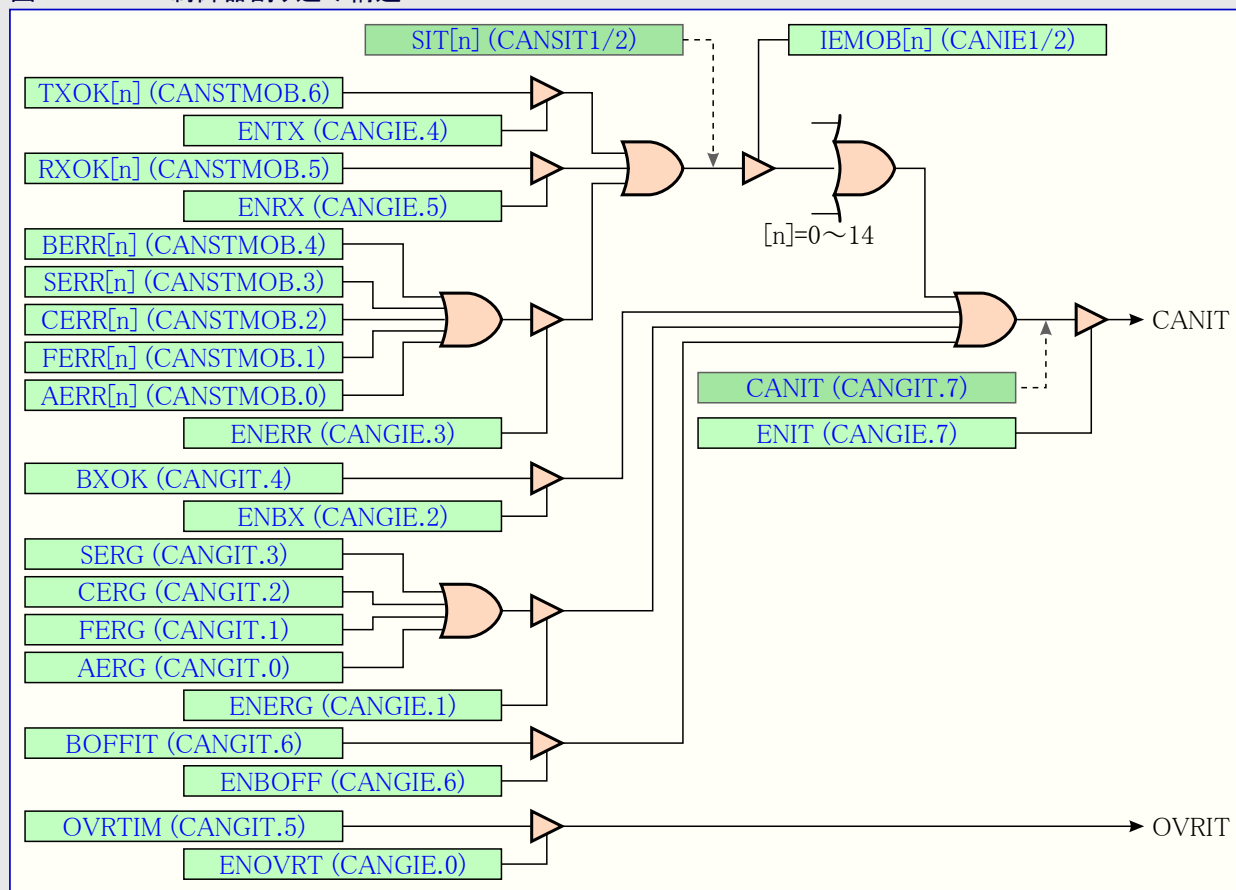
20.9.1. 割り込み構造

各種割り込みは次のとおりです。

- 受信正常完了割り込み
- 送信正常完了割り込み
- 異常割り込み (ビット異常、異極性挿入異常、CRC異常、形式異常、応答異常)
- フレーム緩衝満杯割り込み
- バスOFF設定割り込み
- CANタイマ計時完了割り込み

総合割り込み許可はCAN全割り込み許可(ENIT)ビットによって提供され、CANタイマ計時完了に対する特別な割り込み許可はCANタイマ経過割り込み許可(ENOVRT)ビットによって提供されます。

図20-14. CAN制御器割り込み構造



20.9.2. 割り込みの動き

割り込みが起きると、CAN MOb状態(CANSTMOB)レジスタの対応ビットまたはCAN総合割り込み要求(CANGIT)レジスタの割り込み要求フラグが設定(1)されます。総合割り込み許可(CANGIE)レジスタで受信割り込み許可(ENRX)/送信割り込み許可(ENTX)/MOb異常割り込み許可(ENERR)ビットが設定(1)なら、対応するMOb状態フラグがMOb割り込み状態(CANSIT1,2)レジスタで設定(1)されます。

MOb割り込みに応答するため、CAN MOb状態(CANSTMOB)レジスタの対応(RXOK, TXOK, ...)ビットは応用ソフトウェアで解除(0)されなければなりません。この操作は読み-修正-書きソフトウェア ルーチンを必要とします。

総合割り込みに応答するため、CANGITレジスタの対応(BXOK, BOFFIT, ...)ビットは応用ソフトウェアで解除(0)されなければなりません。この操作はこれらの割り込み要求フラグへの論理1書き込みで行われます(論理0書き込みは割り込み要求フラグ値を変更しません)。

CANタイマ経過割り込み要求(OVRTIM)フラグはCANGITレジスタの他の割り込み元でリセット(0)され、OVRTIM用の割り込み処理ルーチンへの移行でもリセット(0)されます。

CAN節点が送信でフレーム内に形式異常を検出すると、ビット異常も起されます。従って両方が同じ異常(原因)のため、連続的に2つの割り込みが起き得ます。

自分のCANSTMOBレジスタの設定(1)でMOb異常が起こると、CANGITレジスタでの総合異常は設定(1)されません。

20.10. CANホーレート設定例

CANバスは高ホーレートに対して殊の外非常に正確なタイミングが必要です。CAN動作に対しては外部クリスタルの使用だけが推奨されま
す。(タイミング記述については128頁の「ビット タイミング」と「ホーレート」、CANビット タイミング レジスタについては142頁と143頁を参照)

表20-2. 主な周波数に対するCANホーレート設定例

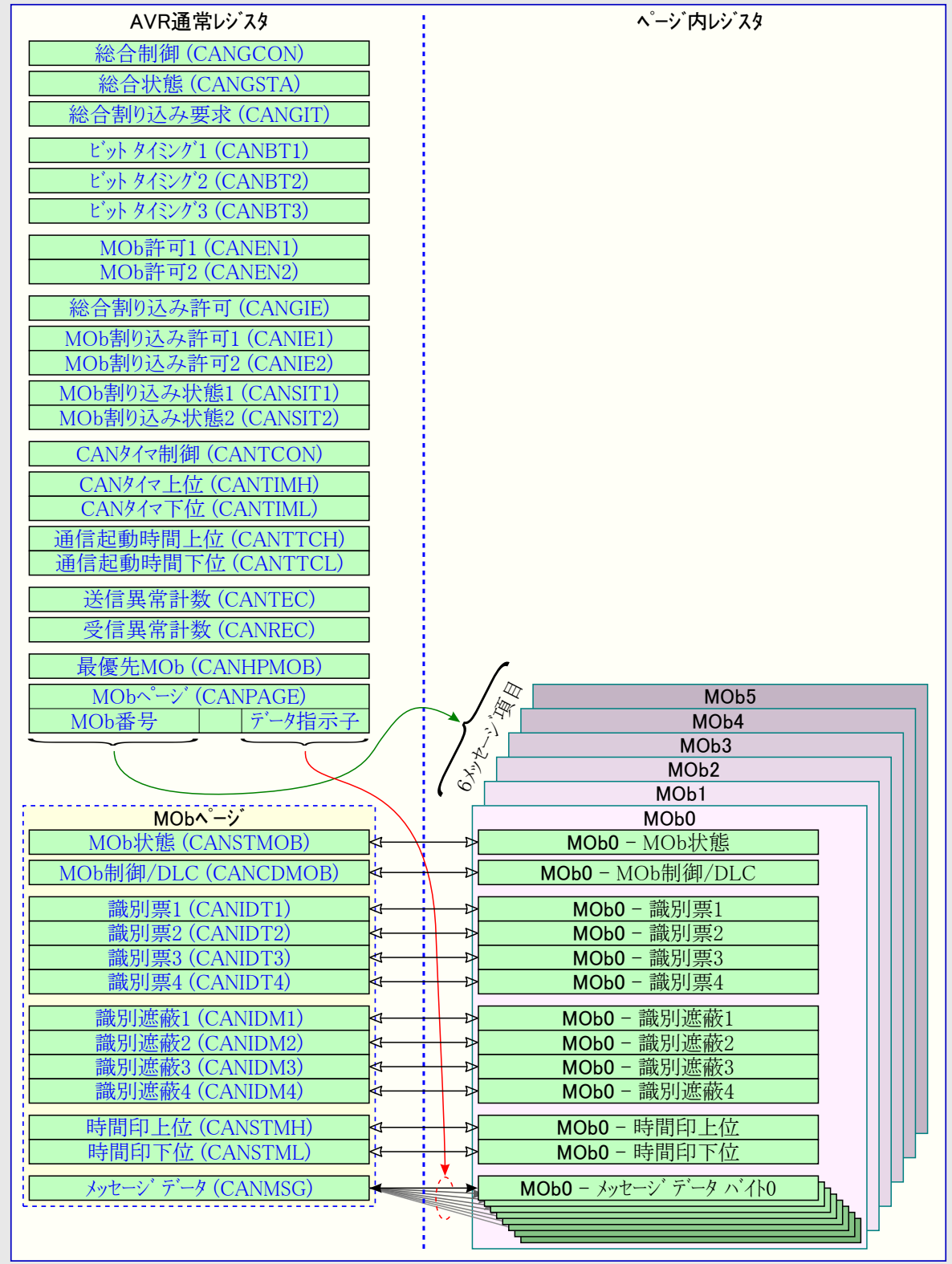
fclk _{I/O} (MHz)	CAN速度 (kbps)	内容			区分 (TQ)				レジスタ		
		採取点 (%)	TQ (ns)	TQ数/ビット	Tprs	Tphs1	Tphs2	Tsjw	CANBT1	CANBT2	CANBT3
16.000	1000	69 (注1)	62.5	16	7	4	4	1	\$00	\$0C	\$36 (注2)
		75	125	8	3	2	2	1	\$02	\$04	\$13
	500	75	125	16	7	4	3	1	\$02	\$0C	\$37
			250	8	3	2	2	1	\$06	\$04	\$13
	250	75	250	16	7	4	4	1	\$06	\$0C	\$37
			500	8	3	2	2	1	\$0E	\$04	\$13
	200	75	312.5	16	7	4	4	1	\$08	\$0C	\$37
			625	8	3	2	2	1	\$12	\$04	\$13
	125	75	500	16	7	4	4	1	\$0E	\$0C	\$37
			1000	8	3	2	2	1	\$1E	\$04	\$13
	100	75	625	16	7	4	4	1	\$12	\$0C	\$37
			1250	8	3	2	2	1	\$26	\$04	\$13
12.000	1000	67 (注1)	83.333	12	5	3	3	1	\$00	\$08	\$24 (注2)
	500	75	166.667	12	5	3	3	1	\$02	\$08	\$25
			250	8	3	2	2	1	\$04	\$04	\$13
	250	75	250	16	7	4	4	1	\$04	\$0C	\$37
			500	8	3	2	2	1	\$0A	\$04	\$13
	200	75	250	20	8	6	5	1	\$04	\$0E	\$4B
			416.667	12	5	3	3	1	\$08	\$08	\$25
	125	75	500	16	7	4	4	1	\$0A	\$0C	\$37
			1000	8	3	2	2	1	\$16	\$04	\$13
	100	75	500	20	8	6	5	1	\$0A	\$0E	\$4B
			833.333	12	5	3	3	1	\$12	\$08	\$25
	8.000	1000	63 (注1)	125	8	3	2	2	1	\$00	\$04
500		69 (注1)	125	16	7	4	4	1	\$00	\$0C	\$36 (注2)
		75	250	8	3	2	2	1	\$02	\$04	\$13
250		75	250	16	7	4	4	1	\$02	\$0C	\$37
			500	8	3	2	2	1	\$06	\$04	\$13
200		75	250	20	8	6	5	1	\$02	\$0E	\$4B
			625	8	3	2	2	1	\$08	\$04	\$13
125		75	500	16	7	4	4	1	\$06	\$0C	\$37
			1000	8	3	2	2	1	\$0E	\$04	\$13
100		75	625	16	7	4	4	1	\$08	\$0C	\$37
			1250	8	3	2	2	1	\$12	\$04	\$13
1000		応用不能									
500	67 (注1)	166.667	12	5	3	3	1	\$00	\$08	\$24 (注2)	
250	75	333.333	12	5	3	3	1	\$02	\$08	\$25	
		500	8	3	2	2	1	\$04	\$04	\$13	
200	80	333.333	15	7	4	3	1	\$02	\$0C	\$35	
		500	10	4	3	2	1	\$04	\$06	\$23	
125	75	500	16	7	4	4	1	\$04	\$0C	\$37	
		1000	8	3	2	2	1	\$0A	\$04	\$13	
100	75	500	20	8	6	5	1	\$04	\$0E	\$4B	
		833.333	12	5	3	3	1	\$08	\$08	\$25	
1000	応用不能										
500	63 (注1)	250	8	3	2	2	1	\$00	\$04	\$12 (注2)	
250	69 (注1)	250	16	7	4	4	1	\$00	\$0C	\$36 (注2)	
	75	500	8	3	2	2	1	\$02	\$04	\$13	
200	70 (注1)	250	20	8	6	5	1	\$00	\$0E	\$4A (注2)	
125	75	500	16	7	4	4	1	\$02	\$0C	\$37	
		1000	8	3	2	2	1	\$06	\$04	\$13	
100	75	500	20	8	6	5	1	\$02	\$0E	\$4B	
		1250	8	3	2	2	1	\$08	\$04	\$13	

注1: 128頁の「20.5.3. ホーレート」項をご覧ください。

注2: 143頁の「ビット0 - SMP : 採取点」項をご覧ください。

20.11. CAN用レジスタ

図20-15. レジスタ構造



20.11.1. CANGCON – CAN総合制御レジスタ (CAN General Control Register)

名称 : CANGCON

変位 : \$D8

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	ABRQ	OVRQ	TTC	SYNTTC	LISTEN	TEST	ENA/STB	SWRES
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 – ABRQ : 中断停止要求 (Aort Request)

これは自動でリセット(=0)するビットではありません。

値	説明
0	要求なし。
1	中断停止要求: MOB許可1(CANEN1)とMOB許可2(CANEN2)レジスタのリセットが行われます。保留通信は直ちに禁止され、進行中の通信は通常のように終了され、適切な状態フラグを設定します。 注: MOB制御/DLC(CANCDMOB)レジスタが無変化に留まります。

● ビット6 – OVRQ : オーバーロード フレーム要求 (Overload Frame Request)

これは自動でリセット(=0)するビットではありません。オーバーロード フレームは総合状態(CANGSTA)レジスタのオーバーロード フレーム送出中(OVRG)フラグ監視で追跡することができます。

値	説明
0	要求なし。
1	オーバーロード フレーム要求: 次の受信フレーム後にオーバーロード フレームを送信します。

● ビット5 – TTC : 時間起動通信許可 (Time Trigger Communication)

値	説明
0	時間起動通信(TTC)なし。
1	時間起動通信(TTC)動作。

● ビット4 – SYNTTC : 時間起動通信同期化 (Synchronization of TTC)

このビットは時間起動通信動作でだけ使われます。

値	説明
0	時間起動通信(TTC)タイマはフレーム開始(SOF)で捕獲されます。
1	時間起動通信(TTC)タイマはフレーム終了(EOF)の最終ビットで捕獲されます。

● ビット3 – LISTEN : 聴取動作 (Listening Mode)

値	説明
0	聴取動作なし。
1	聴取動作。

● ビット2 – TEST : 試験動作 (Test Mode)

注: このビットが設定(1)された場合、CANは動作不良になるかもしれません。

値	説明
0	試験動作なし。
1	試験動作: 工場試験用を意図したもので、お客様での使用を意図したものではありません。

● ビット1 – ENA/STB : 許可動作/待機動作 (Enable/Standby Mode)

このビットが命令で且つ直ぐに効果を発揮しないため、総合状態(CANGSTA)レジスタの許可フラグ(ENFG)が選んだ動作の真の状態を与えます。

値	説明
0	待機動作: (存在するなら、)進行中の通信は通常のように終了され、CANチャネルは凍結(一時停止)されます。(全てのMOBのメッセージ項目構成設定(CONMOB1,0)ビットは無変化)。送信部は継続的に劣性レベルを供給します。この動作での受信部は許可されませんが、全レジスタとメールボックスはCPUからアクセス可のままです。 注: 受信中に適用された待機動作は進行中の受信の誤りや制御器を不正状態にするかもしれません。ソフトウェアリセット(SWRES)が適用されると、制御器はこの状態から正しく再開(リセット)します。リセットを考慮しないなら、可能な解決策は待機動作へ移行する前に受信部作業中(RXBSY)がなくなるまで待つことです。最善の解決策は最初に中断停止要求(ABRQ)を与え、そして待機動作へ移行する前に受信部作業中(RXBSY)がなくなるまで待つことです。何れの場合も、この待機動作の動きはCANバス上に全く影響を及ぼしません。
1	要求なし。

● ビット0 – SWRES : ソフトウェアリセット要求 (Software Reset Request)

この自動リセット(=0)ビットはCAN制御器をリセットするだけです。

値	説明
0	リセットなし。
1	リセット: このリセットはハードウェアリセットとORされます。

20.11.2. CANGSTA – CAN総合状態レジスタ (CAN General Status Register)

名称 : CANGSTA

変位 : \$D9

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	OVRG	-	TXBSY	RXBSY	ENFG	BOFF	ERRP
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6 – OVRG : オーバーロード フレーム送信中フラグ (Overload Frame Flag)

このフラグは割り込みを生成しません。

値	説明
0	オーバーロード フレームなし。
1	オーバーロード フレーム: 生成したオーバーロード フレームが送出されている限り、ハードウェアによって設定(1)されます。

● ビット4 – TXBSY : 送信部作業中フラグ (Transmitter Busy)

このフラグは割り込みを生成しません。

値	説明
0	送信部作業なし。
1	送信部作業中: フレーム(データ、リモート、オーバーロード、異常フレーム)または応答(ACK)領域が送出されている限り、ハードウェアによって設定(1)されます。フレーム間空きが送出されている時も設定(1)されます。

● ビット3 – RXBSY : 受信部作業中フラグ (Receiver Busy)

このフラグは割り込みを生成しません。

値	説明
0	受信部作業なし。
1	受信部作業中: フレームが受信または監視されている限り、ハードウェアによって設定(1)されます。

● ビット2 – ENFG : 許可フラグ (Enable Flag)

このフラグは割り込みを生成しません。

値	説明
0	CAN制御器禁止: 許可/待機命令が直ぐに効果を発揮しないので、この状態フラグが選んだ動作の真の状況を与えます。
1	CAN制御器許可。

● ビット1 – BOFF : バスOFF動作フラグ (Bus Off Mode)

BOFFはCANチャネルの状態情報を与えます。バスOFF動作への移行だけ、バスOFF割り込み(BOFFIT)を生成します。

値	説明
0	非バスOFF動作。
1	バスOFF動作。

● ビット0 – ERRP : 異常受動動作フラグ (Error Passive Mode)

ERRPはCANチャネルの状態情報を与えます。このフラグは割り込みを生成しません

値	説明
0	非異常受動動作。
1	異常受動動作。

20.11.3. CANGIT – CAN総合割り込み要求レジスタ (CAN General Interrupt Register)

名称 : CANGSTA

変位 : \$DA

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	CANIT	BOFFIT	OVRTIM	BXOK	SERG	CERG	FERG	AERG
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7 – CANIT : 総合割り込み要求フラグ (General Interrupt Flag)

これは読み込み専用ビットです。

値	説明
0	割り込みなし。
1	CAN割り込み: CANタイマ経過(OVRTIM)割り込みを除く全CAN制御器割り込みの写しです。

● ビット6 – BOFFIT : バスOFF割り込み要求フラグ (Bus Off Interrupt Flag)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。BOFFITフラグはCANが(異常能動動作から来る)バスOFF動作への移行時だけ設定(1)されます。

値	説明
0	割り込みなし。
1	CANがバスOFF動作へ移行する時のバスOFF割り込み。

● ビット5 – OVRTIM : CANタイマ経過割り込み要求フラグ (Overrun CAN Timer)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。CANタイマ経過割り込み処理ルーチン移行でも、この割り込み要求フラグをリセット(0)します。

値	説明
0	割り込みなし。
1	CANタイマ経過割り込み: CANタイマが\$FFFFから\$0000に切り換わる時に設定(1)します。

● ビット4 – BXOK : フレーム緩衝受信割り込み要求フラグ (Frame Buffer Receive Interrupt)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。BXOKフラグは緩衝部のMOBの全メッセージ項目構成設定(CONMOB1.0)領域が先に再書き込みされてしまっている場合にだけ解除(0)できます。

値	説明
0	割り込みなし。
1	集中受信割り込み: フレーム緩衝受信完了時に設定(1)します。

● ビット3 – SERG : 総合異極性挿入異常割り込み要求フラグ (Stuff Error Geneal)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。

値	説明
0	割り込みなし。
1	異極性挿入異常割り込み: 同一極性で連続5ビット超過を検出。

● ビット2 – CERG : 総合CRC異常割り込み要求フラグ (CRC Error Geneal)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。

値	説明
0	割り込みなし。
1	CRC異常割り込み: 異極性挿入を解除したメッセージでのCRC検査がCRC領域と合致しません。

● ビット1 – FERG : 総合形式異常割り込み要求フラグ (Form Error Geneal)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。

値	説明
0	割り込みなし。
1	形式異常割り込み: CRC終了子、応答終了子、またはフレーム終了(EOF)で固定形式の1つ以上の違反。

● ビット0 – AERG : 総合応答異常割り込み要求フラグ (Acknowledgment Error Geneal)

論理1書き込みがこの割り込み要求フラグをリセット(0)します。

値	説明
0	割り込みなし。
1	応答異常割り込み: 応答スロットで優性ビット未検出。

20.11.4. CANGIE – CAN総合割り込み許可レジスタ (CAN General Interrupt Enable Register)

名称 : CANGIE

変位 : \$DB

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	ENIT	ENBOFF	ENRX	ENTX	ENERR	ENBX	ENERG	ENOVRT
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 – ENIT : 全CAN割り込み許可 (Enable all Interrupts)

CANタイマ経過割り込み(OVRTIM)を除きます。

値	説明
0	割り込み禁止。
1	総合割り込み(CANIT)許可。

● ビット6 – ENBOFF : バスOFF割り込み許可 (Enable Bus Off Interrupt)

値	説明
0	割り込み禁止。
1	バスOFF割り込み(BOFFIT)許可。

● ビット5 – ENRX : 受信割り込み許可 (Enable Receive Interrupt)

値	説明
0	割り込み禁止。
1	受信割り込み(RXOK[n])許可。

● ビット4 – ENTX : 送信割り込み許可 (Enable Transmit Interrupt)

値	説明
0	割り込み禁止。
1	送信割り込み(TXOK[n])許可。

● ビット3 – ENERR : MOB異常割り込み許可 (Enable MOB Errors Interrupt)

値	説明
0	割り込み禁止。
1	MOB異常割り込み(BERR[n],SERR[n],CERR[n],FERR[n],AERR[n])許可。

● ビット2 – ENBX : フレーム緩衝割り込み許可 (Enable Frame Buffer Interrupt)

値	説明
0	割り込み禁止。
1	フレーム緩衝割り込み(BXOK)許可。

● ビット1 – ENERG : 総合異常割り込み許可 (Enable General Errors Interrupt)

値	説明
0	割り込み禁止。
1	総合異常割り込み(SERG,CERG,FERG,AERG)許可。

● ビット0 – ENOVRT : CANタイマ経過割り込み許可 (Enable CANTimer Overrun Interrupt)

値	説明
0	割り込み禁止。
1	CANタイマ経過割り込み(OVRTIM)許可。

20.11.5. CANEN1, CANEN2 (CANEN) – CAN MOb許可状態レジスタ (CAN Enable MOb Register)

名称 : CANEN1 : CANEN2 (CANEN)

変位 : \$DC : \$DD

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	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	7	6	5	4	3	2	1	0
	-	-	ENMOB5	ENMOB4	ENMOB3	ENMOB2	ENMOB1	ENMOB0
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5~0 – ENMOB5~0 : MOb許可状態フラグ (Enable MOb)

これらのビットはMObの有効性を提供します。これはMOb(換言すると、MOb制御/DLC(CANCDMOB)レジスタのメッセージ項目構成設定(CONMOB1,0)ビット)が許可されている場合、1に設定されます。一旦送信完了(TXOK)または受信完了(RXOK)が1に設定されると(自動応答に対するTXOKも)、対応するENMOBnがリセット(0)されます。ENMOBnは中断停止または待機動作に適用する禁止動作でのMOb構成設定でも0に設定されます。

値	説明
0	メッセージ項目(MOb)禁止: MObは新規送信または受信に利用可能。
1	メッセージ項目(MOb)許可: MOb使用中。

20.11.6. CANIE1, CANIE2 (CANIE) – CAN MOb割り込み許可レジスタ (CAN Enable Interrupt MOb Register)

名称 : CANIE1 : CANIE2 (CANIE)

変位 : \$DE : \$DF

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	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	7	6	5	4	3	2	1	0
	-	-	IEMOB5	IEMOB4	IEMOB3	IEMOB2	IEMOB1	IEMOB0
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5~0 – IEMOB5~0 : MOb割り込み許可 (Interrupt Enable by MOb)

例: CANIE2=0000 1100はMOb2と3の割り込みを許可。

値	説明
0	割り込み禁止。
1	MOb割り込み許可。

20.11.7. CANSIT1, CANSIT2 (CANSIT) – CAN MOB割り込み状態レジスタ (CAN Status Interrupt MOB Register)

名称 : CANSIT1 : CANSIT2 (CANSIT)

変位 : \$E0 : \$E1

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	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	7	6	5	4	3	2	1	0
	-	-	SIT5	SIT4	SIT3	SIT2	SIT1	SIT0
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5~0 – SIT5~0 : MOB割り込み状態フラグ* (Status of Interrupt by MOB)

例: CANIE2=0010 0001はMOB0と5が割り込みを要求。

値	説明
0	割り込みなし。
1	MOB割り込みあり。

20.11.8. CANBT1 – CANビット タイミング レジスタ1 (CAN Bit Timing Register 1)

名称 : CANBT1

変位 : \$E2

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	BRP5~0						-
アクセス種別	R	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6~1 – BRP5~0 : ホールレート前置分周選択 (Baud Rate Prescaler)

CAN制御システムクロック周期(Tscl)は設定変更可能で、個々のビットタイミングを決めます。

$$T_{scl} = \frac{BRP5 \sim 0 + 1}{clk_{I/O} \text{周波数}}$$

20.11.9. CANBT2 – CANビット タイミング レジスタ2 (CAN Bit Timing Register 2)

名称 : CANBT2

変位 : \$E3

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	SJW1,0		-	PRS2~0			-
アクセス種別	R	R/W	R/W	R	R/W	R/W	R/W	R
リセット値	0	0	0	0	0	0	0	0

● ビット6,5 – SJW1,0 : 再同期跳躍幅 (Re-Synchronization Jump Width)

異なるバス制御器のクロック発振器間の位相差を補償するため、制御器は現在の転送の何れかの適切な信号端で再同期しなければなりません。同期跳躍幅は最大クロック周期数を定義します。ビット周期は再同期化によって短縮または延長されるかもしれません。

$$T_{sjw} = T_{scl} \times (SJW1,0 + 1)$$

● ビット3~1 – PRS2~0 : 伝播時間区分 (Propagation Time Segment)

このビット時間部分は網内での物理的な遅延時間に対する補償に使われます。これはバス線上での信号伝播時間、入力比較器遅延、出力駆動部遅延の合計の2倍です。

$$T_{prs} = T_{scl} \times (PRS2 \sim 0 + 1)$$

20.11.10. CANBT3 – CANビット タイミング レジスタ3 (CAN Bit Timing Register 3)

名称 : CANBT3
 変位 : \$E4
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	PHS22~0			PHS12~0			SMP
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

● ビット6~4 – PHS22~0 : 段階2区分 (Phase Segment 2)

この段階は位相端(エッジ)異常に対する補償に使われます。この区分は再同期跳躍幅によって短縮されるかもしれません。PHS22~0は1~PHS12~0であるべきです。

$$T_{phs2} = T_{scl} \times (PHS22 \sim 0 + 1)$$

● ビット3~1 – PHS12~0 : 段階1区分 (Phase Segment 1)

この段階は位相端(エッジ)異常に対する補償に使われます。この区分は再同期跳躍幅によって延長されるかもしれません。

$$T_{phs1} = T_{scl} \times (PHS12 \sim 0 + 1)$$

● ビット0 – SMP : 採取点 (Sample Point(s))

この任意選択はTXCAN入力ピン上に起こり得る雑音の濾波を許します。

SMP=1構成設定はBRP5~0=0でT_{scl}(TQ)=T_{clkI/O}のため矛盾します。BRP=0の場合、SMPは解除(0)されなければなりません。

値	説明
0	採取は使用者が設定した採取点(SP)で一度起きます。
1	3点での採取設定、最初の採取は使用者が設定した採取点(SP)の2T _{clkI/O} クロック前、SPの1T _{clkI/O} クロック前で再び、最後にSPで起きます。そしてビット値は3採取の多数決によって決定されます。

20.11.11. CANTCON – CANタイマ制御レジスタ (CAN Timer Control Register)

名称 : CANTCON
 変位 : \$E5
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	TPRSC7~0							
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 – TPRSC7~0 : CANタイマ前置分周選択 (CAN Timer Prescaler)

CANタイマ上位カウンタに対する前置分周器は0~255範囲です。これはCAN制御器が許可の場合にCANタイマへクロックを供給します。

$$T_{clkCANTIM} = T_{clkI/O} \times 8 \times (CANTCON7 \sim 0 + 1)$$

20.11.12. CANTIMH, CANTIML (CANTIM) – CANタイマレジスタ (CAN Timer Register)

名称 : CANTIML : CANTIMH (CANTIM)

変位 : \$E6 : \$E7

リセット : \$00

特質 : -

ビット	15	14	13	12	11	10	9	8
CANTIM15~8								
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	6	5	4	3	2	1	0
CANTIM7~0								
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15~0 – CANTIM15~0 : CANタイマ計数值 (CAN Timer Conut)

CANタイマ計数器は0~65535の範囲です。

20.11.13. CANTTCH, CANTTCL (CANTTC) – CAN時間起動通信(TTC)タイマレジスタ (CAN TTC Timer Register)

名称 : CANTTCL : CANTTCH (CANTTC)

変位 : \$E8 : \$E9

リセット : \$00

特質 : -

ビット	15	14	13	12	11	10	9	8
TIMTTC15~8								
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	7	6	5	4	3	2	1	0
TIMTTC7~0								
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15~0 – CANTTC15~0 : 時間起動通信(TTC)タイマ計数值 (TTC Timer Conut)

CAN TTCタイマ計数器は0~65535の範囲です。

20.11.14. CANTEC – CAN送信異常計数レジスタ (CAN Transmit Error Counter Register)

名称 : CANTEC

変位 : \$EA

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
TEC7~0								
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 – TEC7~0 : 送信異常計数值 (Transmit Error Count)

CAN送信異常計数器は0~255の範囲です。

20.11.15. CANREC – CAN受信異常計数レジスタ (CAN Receive Error Counter Register)

名称 : CANREC
 変位 : \$EB
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	REC7~0							
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 – REC7~0 : 受信異常計数値 (Receive Error Count)

CAN受信異常計数器は0~255の範囲です。

20.11.16. CANHPMOB – CAN最優先MOBレジスタ (CAN Highest Priority MOB Register)

名称 : CANHPMOB
 変位 : \$EC
 リセット : \$F0
 特質 : -

ビット	7	6	5	4	3	2	1	0
	HPMOB3~0				CGP3	CGP2	CGP1	CGP0
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	1	1	1	1	0	0	0	0

- ビット7~4 – HPMOB3~0 : 最優先MOB番号 (Highest Priority MOB Number)

MOB割り込み状態(CANSIT1,2)レジスタで最優先権を持つMOBです。CANSIT1,2=全0(MOB割り込みなし)なら、返り値は1111です。

注: 「MOB優先権」と「メッセージID優先権」を混同しないでください。

- ビット3~0 – CGP3~0 : 汎用ビット (General Purpose Bits)

これらのビットは欲したMOBページ(CANPAGE)レジスタ設定(換言すると、AINCとINDX2~0設定)との一致用に予め設定することができます。

20.11.17. CANPAGE – CAN MOBページレジスタ (CAN Page MOB Register)

名称 : CANPAGE
 変位 : \$ED
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	MOBNB3~0				AINC	INDX2~0		
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7~4 – MOBNB3~0 : MOB番号 (MOB Number)

MOB番号選択、利用可能な番号は0~5です。

注: 全てのAVR CANデバイスとの共通性のため、MOBNB3は常に0を書かれなければなりません。

- ビット3 – AINC : データ緩衝部指示子自動進行(負論理) (Auto Increment of the FIFO CAN Data Buffer Index (Active Low))

値	説明
0	指示子自動進行(+1) (既定値)。
1	指示子自動進行なし。

- ビット2~0 – INDX2~0 : データ緩衝部指示子 (FIFO CAN Data Buffer Index)

定義したMOBに対するFIFO内のCANバイトデータのバイト位置です。

20.11.18. CANSTMOB – MOB状態レジスタ (CAN MOB Status Register)

名称 : CANSTMOB

変位 : \$EE

リセット : \$xx

特質 : -

ビット	7	6	5	4	3	2	1	0
	DLCW	TXOK	RXOK	BERR	SERR	CERR	FERR	AERR
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	-	-	-	-	-	-	-	-

● ビット7 – DLCW : データ長符号警告 (Data Length Code Warning)

到着メッセージが予期したデータ長符号(DLC)を持っていません。どのフレーム形式であっても、MOB制御/DLC(CANCDMOB)レジスタのDLC領域は受信したDLCで更新されます。

● ビット6 – TXOK : 送信完了 (Transmit OK)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

送信によって許可された通信が完了されました。TXOKはフレーム終了(EOF)領域の最後で立ちます(0⇒1)。制御器がフレーム送信の準備ができた時に生成側として2つ以上のメッセージ項目(MOB)が許可される場合、先に下位MOB指示子(0~5)が配給されます。

● ビット5 – RXOK : 受信完了 (Receive OK)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

受信によって許可された通信が完了されました。RXOKはフレーム終了(EOF)領域の第6ビットの最後で立ちます(0⇒1)。2つ以上のメッセージ項目(MOB)受信一致の場合、先に下位MOB指示子(0~5)が更新されます。

● ビット4 – BERR : ビット異常(送信のみ) (Bit Error (Only in Transmission))

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

監視値が送ったビット値と異なります。

例外: 調停領域間に**優性**ビットとして送った**劣性**ビット観測と、異常フレーム送信中の応答スロットでの**優性**ビット検出。

● ビット3 – SERR : 異極性挿入異常 (Stuff Error)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

同一極性での5ビットを越える連続の検出。(訳注:原書本位置行が本項先頭行と重複するため削除)

● ビット2 – CERR : CRC異常 (CRC Error)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

受信部は異極性挿入を解除した受信メッセージ毎にフレーム開始からデータ領域までのCRC検査を実行します。この検査が異極性挿入を解除したCRC領域と不一致の場合にCRC異常が設定(1)されます。

● ビット1 – FERR : 形式異常 (Form Error)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

形式異常は以下のビット領域での固定形式の1つ以上の違反の結果です。

- CRC終了子
- 応答終了子
- フレーム終了(EOF)

● ビット0 – AERR : 応答異常 (Acknowledgment)

このフラグは割り込みを生成できます。これはCANSTMOBレジスタ全体で読み-修正-書きソフトウェア ルーチンを使って解除(0)されなければなりません。

応答スロットで**優性**ビットの検出がありません。

20.11.19. CANCDMOB – MOB制御/DLCレジスタ (CAN MOB Control and DLC Register)

名称 : CANCDMOB

変位 : \$EF

リセット : \$xx

特質 : -

ビット	7	6	5	4	3	2	1	0
	CONMOB1,0		RPLV	IDE	DLC3~0			
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	-	-	-	-	-	-	-	-

● ビット7,6 – CONMOB1,0 : メッセージ項目構成設定 (Configuration of Message Object)

これらのビットは実行すべき通信を設定します(リセット後初期値なし)。

これらのビットは一旦通信が実行されても解除(00)されません。使用者は新規の通信を許可するのに構成設定を再書き込みしなければなりません。

- この操作はフレーム緩衝受信完了割り込み要求(BXOK)フラグをリセット(0)できる必要があります。
- この操作はMOB許可状態(CANEN1,2)レジスタの対応ビットも設定(1)します。

値	説明
00	禁止
01	送信許可
10	受信許可
11	フレーム緩衝受信許可

● ビット5 – RPLV : 応答有効 (Reply Valid)

リモート フレーム受信後の自動応答動作で使います。

値	説明
0	応答準備不可
1	応答準備可、応答有効

● ビット4 – IDE : 識別子拡張 (Identifier Extension)

送出的ためのデータまたはリモート フレームのIDEビットです。このビットは受信したデータまたはリモート フレームの対応値で更新されます。

値	説明
0	CAN規格改訂2.0A (識別子長=11ビット)
1	CAN規格改訂2.0B (識別子長=29ビット)

● ビット3~0 – DLC3~0 : データ長符号 (Data Length Code)

メッセージのデータ領域のバイト数です。

送出的ためのデータまたはリモート フレームのDLC領域の値です。DLCの範囲は0~8です。DLC領域>8の場合、DLC=8を実施します。

この領域は受信したデータまたはリモート フレームの対応値で更新されます。期待したDLCが到着DLCと異なる場合、MOB状態(CANSTMOB)レジスタでデータ長符号警告(DLCW)が出現します。

20.11.20. CANIDT1, CANIDT2, CANIDT3, CANIDT4 (CANIDT) – CAN識別票レジスタ (CAN Identifier Tag Registers) V2.0A部

名称 : CANIDT1 : CANIDT2 : CANIDT3 : CANIDT4 (CANIDT)

変位 : \$F0 : \$F1 : \$F2 : \$F3

リセット : \$xx

特質 : -

ビット	31	30	29	28	27	26	25	24
	IDT10~3							
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	-	-	-	-	-	-	-	-
ビット	23	22	21	20	19	18	17	16
	IDT2~0			-	-	-	-	-
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	-	-	-	-	-	-	-	-
ビット	15	14	13	12	11	10	9	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	-	-	-	-	-	-	-	-
ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	RTRTAG	-	RB0TAG
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	-	-	-	-	-	-	-	-

● ビット31~21 – IDT10~0 : 識別子票 (Identifier Tag)

送出のためのデータまたはリモートフレームの識別子領域です。この領域は受信したデータまたはリモートフレームの対応値で更新されます。

● ビット2 – RTRTAG : リモート送信要求票 (Remote Transmission Request Tag)

送出のためのデータまたはリモートフレームのRTRビットです。このビットは受信したデータまたはリモートフレームの対応値で更新されます。自動応答動作の場合、応答を送信する前にこのビットは自動的にリセット(0)されます。

● ビット0 – RB0TAG : 予約ビット0票 (Reserved Bit 0 Tag)

送出のためのデータまたはリモートフレームのRB0ビットです。このビットは受信したデータまたはリモートフレームの対応値で更新されます。

20.11.21. CANIDT1, CANIDT2, CANIDT3, CANIDT4 (CANIDT) – CAN識別票レジスタ (CAN Identifier Tag Registers) V2.0B部

名称 : CANIDT1 : CANIDT2 : CANIDT3 : CANIDT4 (CANIDT)

変位 : \$F0 : \$F1 : \$F2 : \$F3

リセット : \$xx

特質 : -

ビット	31	30	29	28	27	26	25	24
	IDT28~21							
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	-	-	-	-	-	-	-	-
ビット	23	22	21	20	19	18	17	16
	IDT20~13							
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	-	-	-	-	-	-	-	-
ビット	15	14	13	12	11	10	9	8
	IDT12~5							
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	-	-	-	-	-	-	-	-
ビット	7	6	5	4	3	2	1	0
	IDT4~0					RTRTAG	RB1TAG	RB0TAG
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	-	-	-	-	-	-	-	-

● ビット31~3 – IDT28~0 : 識別子票 (Identifier Tag)

送出のためのデータまたはリモートフレームの識別子領域です。この領域は受信したデータまたはリモートフレームの対応値で更新されます。

● ビット2 – RTRTAG : リモート送信要求票 (Remote Transmission Request Tag)

送出のためのデータまたはリモートフレームのRTRビットです。このビットは受信したデータまたはリモートフレームの対応値で更新されます。自動応答動作の場合、応答を送信する前にこのビットは自動的にリセット(0)されます。

● ビット1 – RB1TAG : 予約ビット1票 (Reserved Bit 1 Tag)

送出のためのデータまたはリモートフレームのRB1ビットです。このビットは受信したデータまたはリモートフレームの対応値で更新されます。

● ビット0 – RB0TAG : 予約ビット0票 (Reserved Bit 0 Tag)

送出のためのデータまたはリモートフレームのRB0ビットです。このビットは受信したデータまたはリモートフレームの対応値で更新されます。

20.11.22. CANIDM1, CANIDM2, CANIDM3, CANIDM4 (CANIDM) – CAN識別遮蔽レジスタ (CAN Identifier Mask Registers)
V2.0A部

名称 : CANIDM1 : CANIDM2 : CANIDM3 : CANIDM4 (CANIDM)

変位 : \$F4 : \$F5 : \$F6 : \$F7

リセット : \$xx

特質 : -

ビット	31	30	29	28	27	26	25	24
	IDMSK10~3							
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	-	-	-	-	-	-	-	-
ビット	23	22	21	20	19	18	17	16
	IDMSK2~0							
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	-	-	-	-	-	-	-	-
ビット	15	14	13	12	11	10	9	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	-	-	-	-	-	-	-	-
ビット	7	6	5	4	3	2	1	0
	-	-	-	-	-	RTRMSK	-	IDEMSK
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	-	-	-	-	-	-	-	-

● ビット31~21 – IDMSK10~0 : 識別子遮蔽 (Identifier Mask)

値	説明
0	強制比較一致
1	ビット比較許可

● ビット2 – RTRMSK : リモート送信要求遮蔽 (Remote Transmission Request Mask)

値	説明
0	強制比較一致
1	ビット比較許可

● ビット0 – IDEMSK : 識別子拡張遮蔽 (Identifier Extension Mask)

値	説明
0	強制比較一致
1	ビット比較許可

20.11.23. CANIDM1, CANIDM2, CANIDM3, CANIDM4 (CANIDM) – CAN識別遮蔽レジスタ (CAN Identifier Mask Registers) V2.0B部

名称 : CANIDM1 : CANIDM2 : CANIDM3 : CANIDM4 (CANIDM)

変位 : \$F4 : \$F5 : \$F6 : \$F7

リセット : \$xx

特質 : -

ビット	31	30	29	28	27	26	25	24
	IDMSK28~21							
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	-	-	-	-	-	-	-	-
ビット	23	22	21	20	19	18	17	16
	IDMSK20~13							
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	-	-	-	-	-	-	-	-
ビット	15	14	13	12	11	10	9	8
	IDMSK12~5							
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	-	-	-	-	-	-	-	-
ビット	7	6	5	4	3	2	1	0
	IDMSK4~0					RTRMSK	-	IDEMSK
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	-	-	-	-	-	-	-	-

● ビット31~3 – IDMSK28~0 : 識別子遮蔽 (Identifier Mask)

値	説明
0	強制比較一致
1	ビット比較許可

● ビット2 – RTRMSK : リモート送信要求遮蔽 (Remote Transmission Request Mask)

値	説明
0	強制比較一致
1	ビット比較許可

● ビット0 – IDEMSK : 識別子拡張遮蔽 (Identifier Extension Mask)

値	説明
0	強制比較一致
1	ビット比較許可

20.11.24. CANSTMH, CANSTML (CANSTM) – CAN時間印レジスタ (CAN Time Stamp Register)

名称 : CANISTML : CANSTMH (CANSTM)

変位 : \$F8 : \$F9

リセット : \$xx

特質 : -

ビット	15	14	13	12	11	10	9	8
	TIMSTM15~8							
アクセス種別	R	R	R	R	R	R	R	R
リセット値	-	-	-	-	-	-	-	-
ビット	7	6	5	4	3	2	1	0
	TIMSTM7~0							
アクセス種別	R	R	R	R	R	R	R	R
リセット値	-	-	-	-	-	-	-	-

● ビット15~0 – TIMSTM15~0 : 時間印計数値 (Time Stamp Conut)

CAN時間印計数器は0~65535の範囲です。

20.11.25. CANMSG – CANメッセージ データ レジスタ (CAN Data Message Register)

名称 : CANMSG

変位 : \$FA

リセット : \$xx

特質 : -

ビット	7	6	5	4	3	2	1	0
	REC7~0							
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	-	-	-	-	-	-	-	-

● ビット7~0 – MSG7~0 : メッセージ データ (Message Data)

このレジスタはMOBページ(CANPAGE)レジスタで指示されたCANバイト データを含みます。

MOBページ レジスタ書き込み後、このバイトは予め定義した識別子(MOBNB3~0)+指示子(INDX2~0)の指定したメッセージ位置と等価です。自動進行が使われると、このデータ レジスタの読みまたは書きの最後で指示子(INDX2~0)が自動進行(+1)されます。

計数の範囲は繰り返し終りなしでの8です(0,1,・・・,7,0,・・・)。

21. LIN/UART – 局所相互連結網制御器またはUART

21.1. 特徴

21.1.1. LIN

- LIN2.1(LIN1.3互換)のハードウェア実装
- LIN2.1仕様の「LIN作業の流れの概念」に基づく、小さく、CPU効率的で独立した主/従ルーチン
- 自動的なLIN先頭部処理と無関係なLINフレームの濾過
- 自動LIN応答処理
- 拡張されたLIN異常検出と合図
- ハードウェアによるフレーム時間超過検出
- 「データ内中断」支援能力
- 正しいフレームの完全性を保証するための自動再同期
- 十分に柔軟な拡張フレーム支援能力

21.1.2. UART

- 全二重動作(独立した直列受信と送信の処理)
- 非同期動作
- 高分解能ボーレート生成器
- 8ビットデータ、奇/偶/なしのパリティビット、1停止ビットのフレームのハードウェア支援
- データオーバーランとフレーミング異常の検出

21.2. 概要

局所相互連結網(LIN:Local Interconnect Network)は分配された車載応用に於ける電子機械節点(ノード)の制御を効果的に支援する直列通信規約です。LINバスの主な特性は以下です。

- 複数従装置を持つ単一主装置の概念
- 一般的なUART/SCIインターフェースに基づく安価なシリコン実装
- 従節点(ノード)での自己同期
- 先行して計算可能な信号伝播時間を持つ確定的信号転送
- 最大20kビット/sの速度

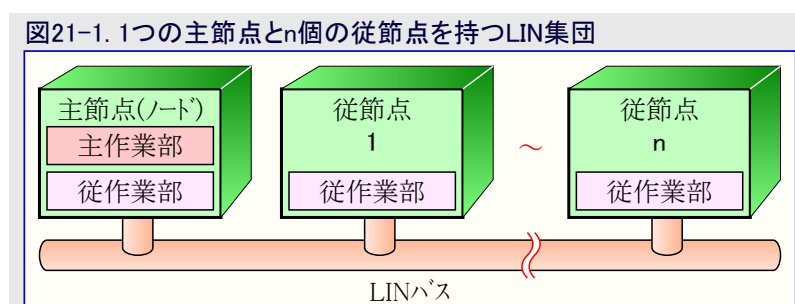
LINはCANの帯域や多能性が必要とされない費用効率的なバス通信を提供します。信号線の駆動部/受信部の仕様はISO9141 NRZ規格に一致する必要があります。

LINが必要とされない場合、この制御器は代わりにUART(Universal Asynchronous serial Receiver and Transmitter)として設定することができます。

21.3. LIN規約

21.3.1. 主作業と従作業

LIN集団は1つの主作業と多数の従作業から成ります。主節点(ノード)は従作業だけでなく主作業も含みます。他の全ての節点は従作業だけを含みます。



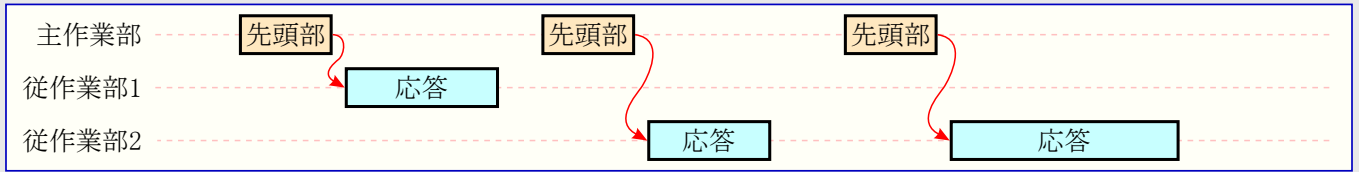
主作業はバス上にどのフレームが転送されるべきかとその時を決めます。従作業は各フレームによって転送されたデータを提供します。主と従の両作業はフレーム処理部の一部です。

21.3.2. フレーム

フレームは(主作業によって供給される)先頭部と(従作業によって供給される)応答から成ります。

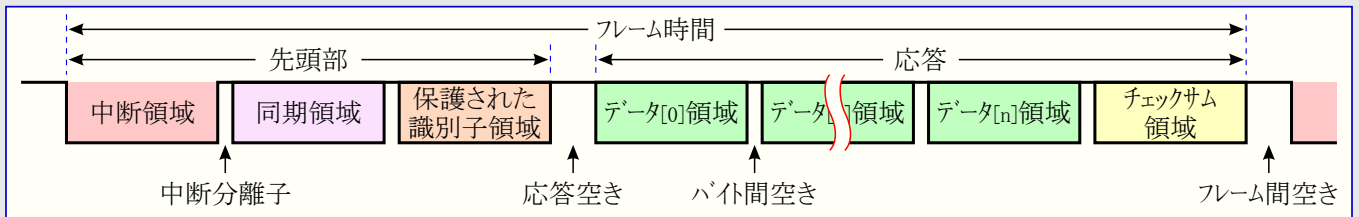
先頭部は保護された識別子(PROTECTED IDENTIFIER)領域が後続する、中断(BREAK)と同期(SYNC)の領域から成ります。識別子はそのフレームの目的を一意に定義します。識別子に関連する応答を提供するように指定された従作業部はそれを送出します。応答はデータ(DATA)領域とチェックサム(CHECKSUM)領域から成ります。

図21-2. LINフレームに於ける主/従作業部の動き



その識別子に関連するデータを待っている従作業部は、その応答を受信してチェックサム照合後に転送されたデータを使います。

図21-3. LINフレームの構造



注: 各バイト領域はLSB先行の連続バイトとして送信されます。

21.3.3. データ転送

フレームに於いて信号または診断メッセージの2つのデータ形式が転送されるかもしれません。

- 信号
信号はフレームのデータ領域内に詰められた数量値またはバイト配列です。信号は同じ識別子を持つ全てのフレームのデータ領域の常に同じ位置に存在します。
- 診断メッセージ
診断メッセージは2つの予約された識別子を持つフレームで転送されます。データ領域の解釈はデータ領域自体だけでなく通信節点(ノード)の状態にも依存します。

21.3.4. 計画表

(主節点(ノード)に於ける)主作業は計画表に基づいてフレーム先頭部を送信します。計画表は各先頭部に対する識別子と、フレームの開始と後続するフレーム間の間隔を指定します。主応用は異なる計画表を用い、それらの中から選ぶかもしれません。

21.3.5. LIN1.3との互換性

LIN2.1はLIN1.3の上位仕様です。

LIN2.1の主節点(ノード)はLIN1.3とLIN2.1両方の従節点から成る集合を処理することができます。そして主装置はLIN1.3従装置から新しいLIN2.1機能を要求することを避けます。

- 強化されたチェックサム
- 形態再設定と診断
- 自動ポーレート検出
- “応答異常”状態監視

LIN2.1従節点はLIN1.3主節点で操作することができません(例えば、LIN1.3主装置は強化されたチェックサムを支援しません)。

LIN2.1物理層はLIN1.3物理層と過去互換です。しかし、別物ではありません。LIN2.1物理層は大きな必要条件を整え、換言すると、LIN2.1物理層を使う主節点はLIN1.3集合で動作することができます。

21.4. LIN/UART制御器

LIN/UART制御器は3つの主機能に分けられます。

- Tx LIN先頭部機能
- Rx LIN先頭部機能
- LIN応答機能

これらの機能は主に2つの役務を使います。

- Rx役務
- Tx役務

これら2つの役務が基本的にUART役務なので、制御器はUART機能に切り替えることもできます。

21.4.1. LIN概要

LIN/UART制御器はLINソフトウェア応用構造に可能な限り近く合致するように設計されています。LINソフトウェア応用は多数の従作業と1つの主作業の独立した作業として開発されます(「計画表」参照)。Atmel ATmega16M1/32M1/64M1はこの見通しに従います。主作業と従作業間の繋がりは一旦新しい識別子が利用可能になると割り込みルーチンが呼ばれる交差点だけです。従って、主節点では主と従の両作業を収容し、Tx LIN先頭部機能は識別子の存在する従作業に警報を出します。同様に、従節点ではRx LIN先頭部機能が識別子の存在する従作業に警報を出します。

従作業が識別子の存在を警告される時に、応答で何を行うのかを知るため、最初にそれを分析しなければなりません。ハードウェアのフラグが60(\$3C)~63(\$3F)の特定識別子の1つの存在を認識します。

LIN通信に関しては、管理のために以下の4つの割り込みだけが必要です。

- LIDOK : 新しいLIN識別子利用可
- LRXOK : LIN応答受信
- LTXOK : LIN応答送信
- LERR : LIN異常

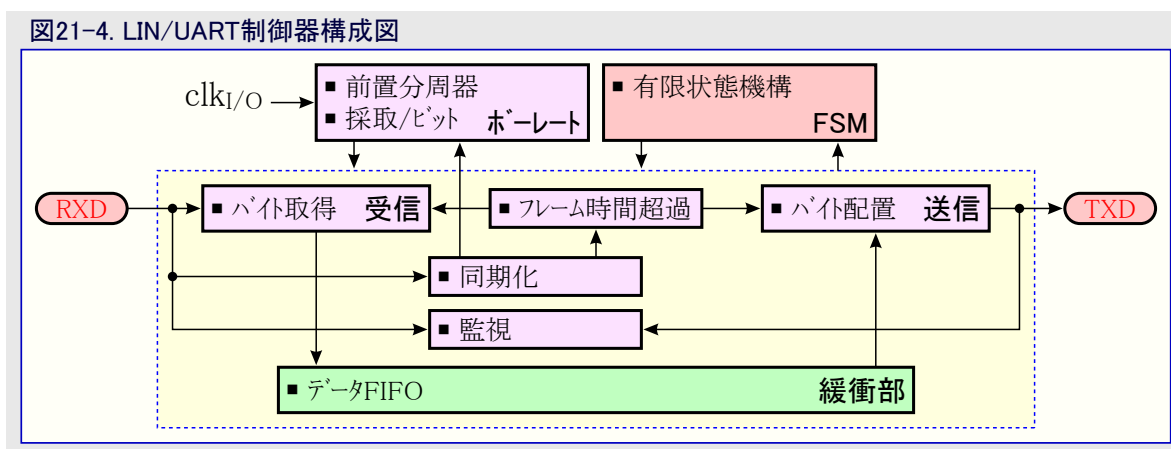
起動管理はLIN2.1に関して最小5ビットのLow(\$F0)、LIN1.3に関して8ビットのLow(\$80)を送出する節点(ノード)と、UART起動能力を使って自動化することができます。LIN起動信号でのピン変化割り込みは休止形態の1つのデバイスを抜けるのに使うこともできます。

62(\$3E)と63(\$3F)の拡張フレーム識別子は使用者定義メッセージ形式と将来のLIN形式の組み込みを許すために予約されています。UARTによって提供されるバイト転送形態はLIN規約に適合するLIN従装置の上位互換を保証します。

21.4.2. UART概要

LIN/UART制御器は伝統的なUARTとしても機能することができます。既定により、UARTは全二重制御器として動作します。これは試験目的に局所戻し閉路を持ちます。UARTは送信用に1文字と受信用に2文字分の緩衝する能力を持ちます。受信緩衝部は1つの8ビット直列レジスタと後続する1つの独立した緩衝部レジスタから作られます。応用が文字を置くまたは得る時の自動的なフラグ管理が実装され、故にソフトウェアの付随作業を減らします。送信と受信の役務が独立しているため、使用者は2つの役務の1つが使われない時に1つのデバイスピンを節約することができます。UARTはどんなクロック周波数と目的ボーレートでも2%の最大誤差を提供する強化されたボーレート生成器を持ちます。

21.4.3. LIN/UART制御器構造



21.4.4. LIN/UART指令概要

図21-5. LIN/UART指令依存性

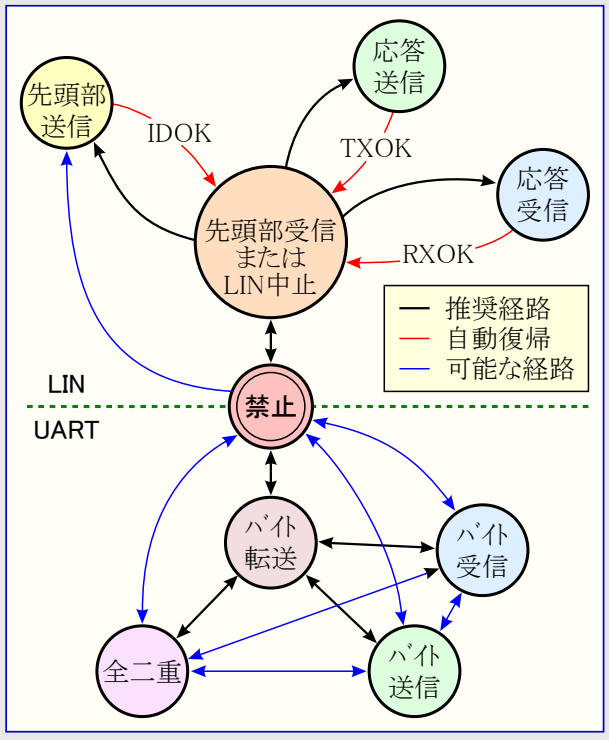


表21-1. LIN/UART指令一覧

LENA	LCMD2~0	指令	注釈
0	x x x	周辺機能禁止	
1	0 0 0	先頭部受信 - LIN中止	LIN撤回
	0 0 1	先頭部送信	送信後、LCMD2~0=000
	0 1 0	応答受信	受信後、LCMD2~0=000
	0 1 1	応答送信	送信後、LCMD2~0=000
	1 0 0	パケット転送	CRCなし、時間超過監視なし、LTXDL=LRXDL=0 (LINDLRは読み込み専用レジスタ)
	1 0 1	パケット送信	
	1 1 0	パケット受信	
	1 1 1	全二重	

21.4.5. 許可/禁止

LINCRレジスタの許可(LENA)ビットの設定(1)がLIN/UART制御器を許可します。LIN/UART制御器を禁止するにはLENAビットが0を書かれなければならない。待ち状態が全く実装されないため、禁止指令は直ちに行われます。

21.4.6. LIN指令

LIN制御レジスタ(LINCR)の動作形態(LCMD2)ビットの解除(0)がLIN指令を許可します。

表21-1.で示されるように、LINCRレジスタの指令(LCMD1,0)ビットによって制御される4つの機能が利用可能です(図21-5.参照)。

21.4.6.1. 先頭部受信/LIN中止機能

この機能(または状態)は主に制御器の撤回形態です。

制御器が主作業を実行しなければならない時に、この状態は先頭部送信指令を許可する前の開始位置です。

制御器が従作業だけを実行しなければならない時に、LIN先頭部検出/獲得が背面機能として許可されます。このような獲得(先頭部受信機能)の最後に於いて、自動的に適切なフラグが設定(1)され、LIN1.3ではLINデータ長レジスタ(LINDLR)が符号化されていない長さ値で設定されます。

この状態は応答送信または応答受信の指令を許可する前の開始位置でもあります。

機能(換言すると、先頭部送信、応答送信、または応答受信)の走行はLIN制御レジスタ(LINCR)の指令(LCMD1,0)ビットを解除(00)することによって中止することができます。この場合、他のソフトウェア作業に通知するために、LIN異常レジスタ(LINERR)の中断(LABORT)フラグが設定(1)されます。待ち状態が全く実装されないため、中止指令は直ちに行われます。

先頭部受信機能は以下に対して責任があります。

- 中断(BREAK)領域検出
- 同期(SYNC)領域を分析するハードウェア再同期化
- 保護された識別子(PROTECTED IDENTIFIER)領域の受信、パリティ制御とLIN1.3の場合に於けるLINDLRレジスタの更新
- フレーム時間超過監視の開始
- LIN通信の完全性検査

21.4.6.2. 先頭部送信機能

LIN規約に従って、主作業だけがこの機能を許可しなければなりません。先頭部は設定されたポーレート(LINポーレートレジスタ(LINBRR)とLINビットタイミングレジスタ(LINBTR)参照)と適切な間隔で送出されます。

制御器は以下に対して責任があります。

- 中断(BREAK)領域の送信 - 13個の優先ビット
- 同期(SYNC)領域の送信 - 文字(\$55)
- 保護された識別子(PROTECTED IDENTIFIER)領域の送信。これはLIN識別子レジスタ(LINIDR)の完全な内容です(検査ビット自動内包)。

この送信の最後に於いて、制御器は適切なフラグを設定した後で自動的に先頭部受信/LIN中止状態(換言すると、指令(LCMD1,0)ビット=00)に復帰します。この機能は制御器を先頭部受信機能後と同じ設定にします。これはLIN1.3に於いて先頭部送信機能の最後でLINデータ長レジスタ(LINDLR)が符号化されていない長さ値で設定されることを意味します。

この機能中、制御器は以下に対しても責任があります。

- フレーム時間超過監視の開始
- LIN通信の完全性検査

21.4.6.3. 応答送信と応答受信の機能

これらの機能はLIN節点(ノード)の従作業によって開始されます。これらは先頭部送信(主作業)後、または先頭部受信(従作業に属すと見做される)後に使われなければなりません。応答送信指令が送られると、送信が始まります。応答受信指令は先頭バイトの最終直列ビットの受信(停止ビットの前)までに送ることができます。

LIN1.3では先頭部間隔がLINデータ長レジスタ(LINDLR)を構成設定します。LIN2.1では使用者がLINDLRレジスタを、応答受信に対してLIN受信データ長(LRXDL3~0)または応答送信に対してLIN送信データ長(LTXDL3~0)のどちらかを構成設定しなければなりません。

指令開始時、制御器はチェックサム計算に対して正しい規則を適用するためにLIN制御レジスタ(LINCR)のLIN1.3形態(LIN13)ビットを調べます。データ(DATA)バイトと保護された識別子(PROTECTED IDENTIFIER)に渡るチェックサム計算は拡張チェックサムと呼ばれ、LIN2.1従装置との通信に用いられます。データ(DATA)バイトだけに渡るチェックサム計算は古典的チェックサムと呼ばれ、LIN1.3従装置との通信に用いられます。60~63(\$3C~\$3F)の識別子が常に古典的チェックサムを使うことに注意してください。

この送受信の最後に於いて、適切なフラグを設定した後で制御器は自動的に先頭部受信/LIN中止状態(換言すると、指令(LCMD1,0)ビット=00)に復帰します。

LIN異常が起きた場合、送信または受信は中止され、適切なフラグが設定され、そしてLINバスは劣性状態のままにされます。

これらの関数中、制御器は以下に対して責任があります。

- チェックサム操作部の初期化
- チェックサム計算の更新と共に'n'データの送信または受信
- チェックサム領域の送信または検査
- フレーム時間超過監視の開始
- LIN通信の完全性検査

制御器が応答を送信または受信の間に、中断(BREAK)と同期(SYNC)の領域を検知し、この新先頭部の識別子を記録することができます。勿論、直前の応答での特定の異常はこの識別子受信と共に保持されます。

21.4.6.4. LIN応答のデータの扱い

LIN応答のデータ用にFIFOデータ緩衝器が用いられます。LINデータ緩衝部選択(LINSEL)レジスタの全項目設定後、LINデータ(LINDAT)レジスタへの繰り返しアクセスがデータ読み込みまたはデータ書き込みを実行します(「データ管理」をご覧ください)。

LINDLRのLIN受信データ長(LRXDL3~0)とLIN送信データ長(LTXDL3~0)がデータアクセスに関連しないことに注意してください。

21.4.7. UART指令

LIN制御レジスタ(LINCR)の動作形態(LCMD2)ビットの設定(1)がUART指令を許可します。

- バイト送信とバイト受信の役割は表21-1.で示されるように独立です。
- バイト転送: UARTが選ばれますが、送受信の両役割は禁止されます。
- バイト受信: 受信役割が許可されるだけで、送信役割は禁止されます。
- バイト送信: 受信役割が許可されるだけで、受信役割は禁止されます。
- 全二重: UARTが許可され、送受信の両役割が許可されます。

この役割の組み合わせはLINCRレジスタの指令(LCMD1,0)ビットによって制御されます(図21-5.をご覧ください)。

21.4.7.1. データの扱い

LIN通信に使われるFIFOはUARTアクセス中に禁止されます。それでLINデータ長レジスタ(LINDLR)のLIN受信データ長(LRXDL3~0)とLIN送信データ長(LTXDL3~0)は無関係です。そしてデータレジスタとしてLINデータ(LINDAT)レジスタが使われ、LINデータ緩衝部選択(LINSEL)レジスタは無関係ありません。

21.4.7.2. 受信役務

一旦この役務が許可されると、使用者はLINSIRの**受信実行割り込み要求(LRXOK)フラグ**によって到着文字を警告されます。LINデータ(LINDAT)レジスタ読み込みはこのフラグを解除(0)して緩衝部の第2段目を開放します。使用者がそれを読むことなく到着文字が無関係と見做すなら、このフラグを直接解除(0)することができます(166頁の「LINSIR - LIN状態/割り込み要求レジスタ」で記述される特定フラグの管理をご覧ください)。

受信役務の本質的な構造が2バイトの緩衝器を提供します。1つ目は直列⇒並列変換用に使われ、2つ目は変換結果を受け取ります。LINDATレジスタ読み込みがこの第2緩衝バイトに至ります。2バイトの緩衝器が満杯の場合、新規到着文字は既に記録された2つ目を上書きします。そしてLIN異常レジスタ(LINERR)の**オーバーラン異常(LOVERR)フラグ**が読み込み時にこの文字と連動します。

LINERRレジスタの**フレーミング異常(LFERR)フラグ**はフレーミング異常の場合に設定(1)されます。

21.4.7.3. 送信役務

この役務が許可されると、使用者はLINデータ(LINDAT)レジスタを書くことによって文字を送ります。自動的にLINSIRの**送信実行割り込み要求(LTXOK)フラグ**が解除(0)されます。これは直列送信の最後で立ちます。送るべき新規文字がなければ、LTXOKフラグは独立して(別に)解除(0)することができます(「LINSIR - LIN状態/割り込み要求レジスタ」で記述される特定フラグの管理をご覧ください)。

送信の緩衝はありません。

この役務によって検知される異常はありません。

21.5. LIN/UART説明

21.5.1. リセット

AVRコアのリセット論理回路信号はLIN/UART制御器もリセットします。存在するリセットの別の形式、ソフトウェアリセットはLIN制御レジスタ(LINCR)のソフトウェアリセット(LSWRES)ビットによって制御されます。この自己リセットビットは部分的なリセットを実行します。

表21-2. LIN/UARTレジスタのリセット

レジスタ名称	レジスタ略称	システム リセット値	LSWRES ソフトウェア リセット値	注釈
LIN制御レジスタ	LINCR	0000 0000	0000 0000	
LIN状態/割り込み要求レジスタ	LINSIR	0000 0000	0000 0000	
LIN割り込み許可レジスタ	LINENIR	0000 0000	xxxx 0000	
LIN異常レジスタ	LINERR	0000 0000	0000 0000	
LINビットタイミングレジスタ	LINBTR	0010 0000	0010 0000	x=不定 u=不変
LINポーレート下位レジスタ	LINBRRL	0000 0000	uuuu uuuu	
LINポーレート上位レジスタ	LINBRRH	0000 0000	xxxx uuuu	
LINデータ長レジスタ	LINDLR	0000 0000	0000 0000	
LIN識別子レジスタ	LINIDR	1000 0000	1000 0000	
LINデータ緩衝部選択レジスタ	LINSEL	0000 0000	xxxx 0000	
LINデータレジスタ	LINDAT	0000 0000	0000 0000	

21.5.2. クロック

I/Oクロック信号($clk_{I/O}$)はLIN/UART制御器をもクロック駆動します。これはその唯一のクロックです。

21.5.3. LIN規約選択

LIN規約を選ぶのにLIN制御レジスタ(LINCR)のLIN1.3形態(LIN13)ビットが使われます。

- LIN13=0 (既定) : LIN2.1規約
- LIN13=1 : LIN1.3規約

制御器はチェックサム(LIN2.1での拡張チェックサム、LIN1.3での古典的チェックサム)計算でLIN13ビットを調べます。「**応答送信と応答受信の機能**」をご覧ください。

このビットはUART指令に対して無関係です。

21.5.4. 構成設定

(LINまたはUART)形態に依存して、LIN制御レジスタ(LINCR)の構成設定(LCONF1,0)ビットは制御器を以下の構成設定で設定します。

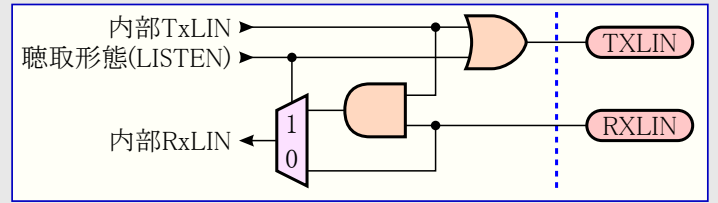
表21-3. 動作形態に対する構成設定表

LCONF1,0	設定形態	
	LIN動作形態	UART動作形態
0 0	LIN標準構成設定 (既定)	8ビット データ, パリティなし, 1停止ビット
0 1	CRC領域の検出または送信なし	8ビット データ, 偶数パリティ, 1停止ビット
1 0	フレーム時間超過監視禁止	8ビット データ, 奇数パリティ, 1停止ビット
1 1	聴取形態動作	聴取形態動作(8ビット データ, パリティなし, 1停止ビット)

LIN構成設定は設定したLIN規約と無関係です。

聴取形態動作は内部TxLINと内部RxLINを共に接続します。この動作形態ではTXLIN出力ピンが禁止され、RXLIN入力は常に許可されます。同じ仕組みはUART形態でも利用可能です。

図21-6. 聴取形態動作

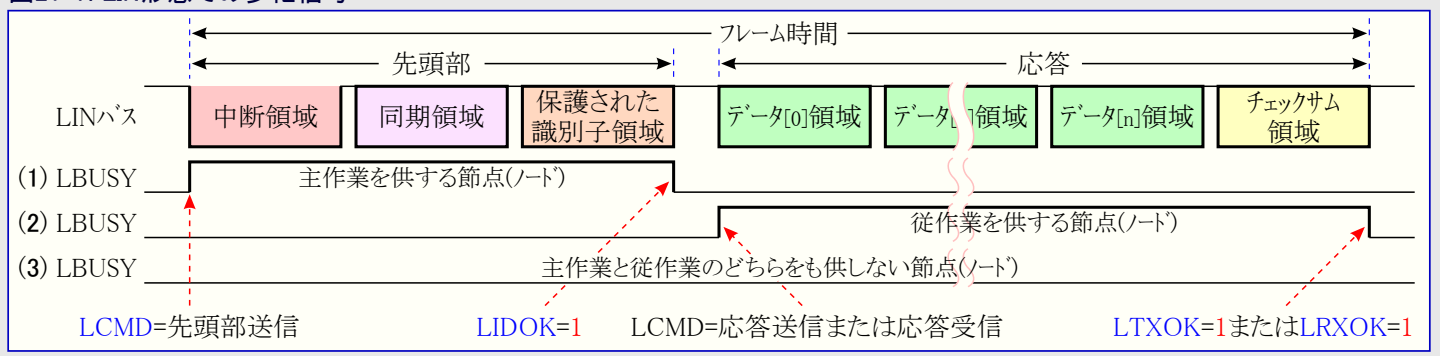


21.5.5. 多忙信号

LIN状態/割り込み要求レジスタ(LINSIR)の多忙信号(LBUSY)フラグは多忙(BUSY)信号の投影です。これはハードウェアによって設定(1)と解除(0)が成されます。これは制御器がLINまたはUART通信で多忙なことを合図します。

21.5.5.1. LIN形態での多忙信号

図21-7. LIN形態での多忙信号



多忙信号が設定(1)されると、いくつかのレジスタが施錠され、使用者書き込みが許されません。

- LIN制御レジスタ(LINCR) - 形態/指令(LCMD2~0)、許可(LENA)、ソフトウェアリセット(LSWRES)を除く
- LINポーレートレジスタ(LINBRRH, LINBRRL)
- LINデータ長レジスタ(LINDLR)
- LIN識別子レジスタ(LINIDR)
- LINデータレジスタ(LINDAT)

多忙信号が設定(1)されている場合に利用可能な指令は以下だけです。

- LCMD1,0=00、中止指令はバイトの最後で処理されます。
- LENA=0やLCMD2=0、禁止指示は直ちに処理されます。
- LSWRES=0、リセット指示は直ちに処理されます。

多忙信号(=1)中に別の指令が許可された場合に新しい指令は有効にされず、LIN異常レジスタ(LINERR)のオーバーラン異常(LOVERR)フラグが設定(1)されることに注意してください。進行中の転送は割り込まれません。

21.5.5.2. UART形態での多忙信号

バイト送信中に多忙信号が設定(1)されます。これは書き込みからいくつかのレジスタを施錠します。

- LIN制御レジスタ(LINCR) - 形態/指令(LCMD2~0)、許可(LENA)、ソフトウェアリセット(LSWRES)を除く
- LINデータレジスタ(LINDAT)

バイト受信中に多忙信号は生成されません。

21.5.6. ビット タイミング

21.5.6.1. ボーレート生成器

ボーレートは秒当たりのビット数(bps)での転送速度を定義します。

- BAUD : ボーレート(bps)
- $clk_{I/O}$: システム/Oクロック周波数
- LDIV11~0 : LINボーレートレジスタ(LINBRRH, LINBRRL)の内容(0~4095)。入力クロックとして $clk_{I/O}$ を受け取る前置分周器。
- LBT5~0 : LINビット タイミング レジスタ(LINBTR)の下位側(0~63)はLINまたはUARTでの採取番号です(既定値=32)。

ボーレートを計算するための式は次です。

$$BAUD = fclk_{I/O} \div LBT5 \sim 0 \times (LDIV11 \sim 0 + 1)$$

LIN分周値(LDIV11~0)設定用の式は次です。

$$LDIV11 \sim 0 = (fclk_{I/O} \div LBT5 \sim 0 \times BAUD) - 1$$

受信に於いて3採取の多数決が行われることに留意してください。

21.5.6.2. LIN形態での再同期

先頭部受信待機時、LINBTRのLINビット タイミング(LBT5~0)は32です。再同期は中断(BREAK)検出時に始まります。中断(BREAK)量が範囲(最小=11、標準=13、最大28ビット)内でなければ、中断(BREAK)は拒絶されます。再同期は受信した先頭部の同期(SYNC)領域(\$55)に対してLBT5~0を補正することによって行われます。その後には新規LBT5~0値を用いて保護された識別子(PROTECTED IDENTIFIER)が採取されます。この制御器に実装された再同期は±20%のクロック偏差を許容し、ボーレートを±2%範囲に補正します。

新しいLBT5~0値は応答の最後まで使われます。そしてLBT5~0は次の先頭部のために32にリセットされます。

LINBTRレジスタはクロック発振器の再校正に用いることができます。

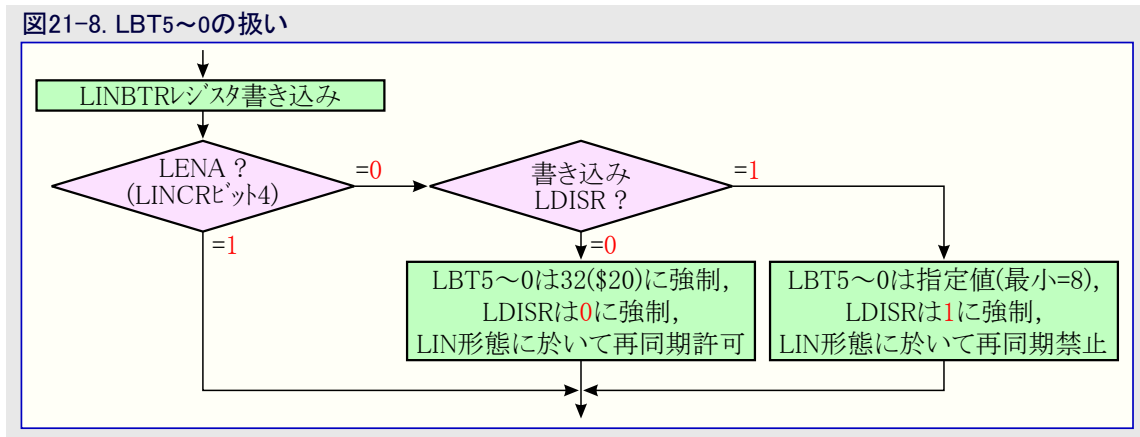
LIN節点(ノード)が主装置として許可されている場合に再同期は実行されません。

21.5.6.3. LBT5~0の扱い

LINBTRレジスタのビット タイミング再同期禁止(LDISR)ビットは以下のために使われます。

- LBT5~0の設定を許可するため(特にUART形態の場合でボーレートを手動補正するため)。
- 試験目的のためにLIN従形態で再同期を禁止。

この処理のためにLIN制御レジスタ(LINCR)の許可(LENA)ビットが重要であることに注意してください。



21.5.7. データ長

「LIN指令」は応答送受信に先立って、LINデータ長レジスタ(LINDLR)のLIN受信データ長(LRXDL3~0)またはLIN送信データ長(LTXDL3~0)領域を自動的にどう設定するか、または設定方法を記述します。

応答送信の場合のLRXDL3~0は既に成功裏に送られたバイト数を数えるため、ハードウェアによって使われます。

応答受信の場合のLTXDL3~0は既に成功裏に受け取ったバイト数を数えるため、ハードウェアによって使われます。

異常が起きた場合、この情報はLINメッセージを回復するためのプログラム作成者に有用です。

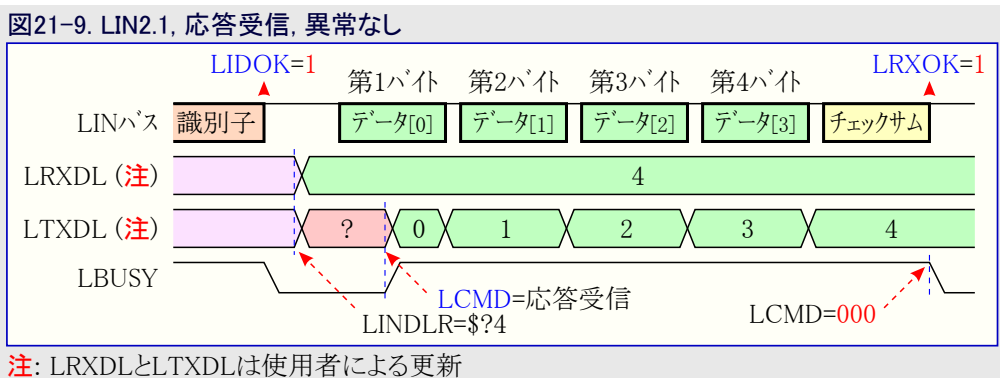
21.5.7.1. LIN2.1でのデータ長

- LTXDL3~0=0なら、チェックサム(CHECKSUM)だけが送られました。
- LRXDL3~0=0なら、最初に受信したバイトがチェックサム(CHECKSUM)として解釈されました。
- LTXDL3~0またはLRXDL3~0が>8なら、その値は指令設定後で最初のバイトの送信または受信に先立って8に強制されます。

21.5.7.2. LIN1.3でのデータ長

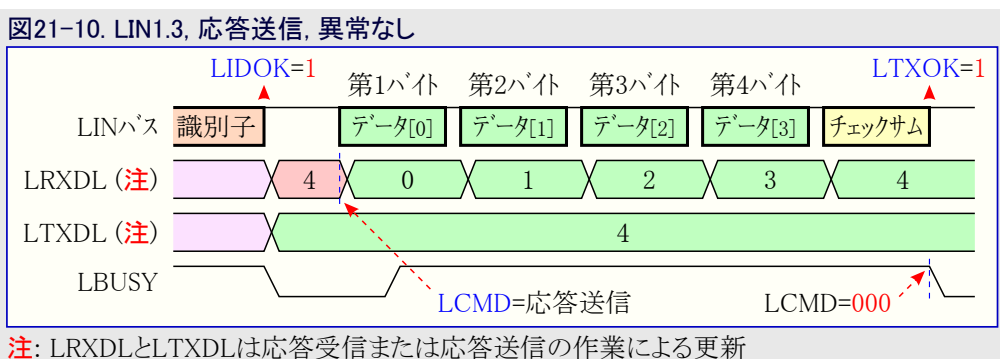
- LRXDLとLTXDLの両領域は受信した保護された識別子(PROTECTED IDENTIFIER)に含まれるデータ長符号の復号により、識別子割り込み要求(LIDOK)フラグを設定(1)する前にハードウェアで更新されます(LRXDL=LTXDL)。
- 上の構造経路では、0または>8の長さは有り得ません。

21.5.7.3. 応答受信でのデータ長



- 使用者は応答受信指令を設定する前にLIN受信データ長(LRXDL)領域を初期化します。
- 応答受信指令設定後、LIN送信データ長(LTXDL)はハードウェアによってリセットされます。
- LRXDL領域は受信(多忙合図)中、無変化に留まります。
- LTXDL領域は(多忙合図)中、受信したバイト数を数えます。
- 異常が起きた場合、受信を停止して対応する異常フラグを設定(1)し、そしてLTXDLは異常なしで受信したバイト数を与えます。
- 異常が起きなかった場合、チェックサム(CHECKSUM)の受信後に受信実行割り込み要求(LRXOK)フラグが設定(1)され、LRXDLは無変化です(LTXDL=LRXDL)。

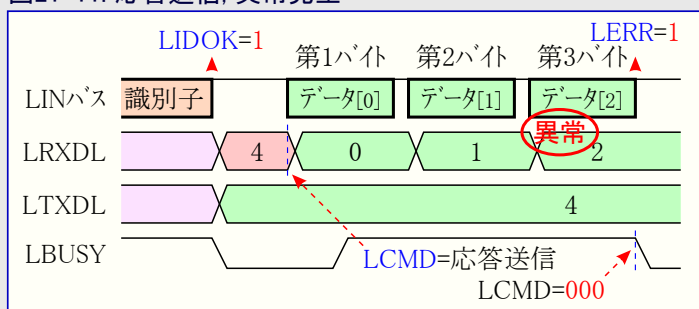
21.5.7.4. 応答送信でのデータ長



- 使用者は応答送信指令を設定する前にLIN送信データ長(LTXDL)領域を初期化します。
- 応答送信指令設定後、LIN受信データ長(LRXDL)はハードウェアによってリセットされます。
- LTXDL領域は送信(多忙合図)中、無変化に留まります。
- LRXDL領域は(多忙合図)中、送信したバイト数を数えます。
- 異常が起きた場合、送信を停止して対応する異常フラグを設定(1)し、そしてLRXDLは異常なしで送信したバイト数を与えます。
- 異常が起きなかった場合、チェックサム(CHECKSUM)の送信後に送信実行割り込み要求(LTXOK)フラグが設定(1)され、LTXDLは無変化です(LRXDL=LTXDL)。

21.5.7.5. 異常後のデータ長

図21-11. 応答送信, 異常発生



注: 応答での情報(例えばバイトでの異常)はバイトの直並列化の終りでだけ利用可能です。

21.5.7.6. UART形態でのデータ長

- UART形態はLRXDLとLTXDLを0に強制し、LINデータ長レジスタ(LINDLR)書き込みを禁止します。
- リセット後にLRXDLとLTXDLも0を強制されることに注意してください。

21.5.8. xxOKフラグ

LINSIRレジスタには3つのxxOKフラグがあります。

- LIN識別子OK(LIDOK)

先頭部送信または先頭部受信のどちらかにより、先頭部の終りで設定(1)されます。LIN1.3ではLIDOK生成前に制御器がLINデータ長レジスタ(LINDLR)のLIN受信データ長(LRXDL)とLIN送信データ長(LTXDL)を更新します。

- LIN応答受信完了(LRXOK)

LIN形態で応答受信機能による応答の終りで、UART形態で一旦文字が受信されると設定(1)されます。

- LIN応答送信完了(LTXOK)

LIN形態で応答送信機能による応答の終りで、UART形態で一旦文字が送信されてしまうと設定(1)されます。

これらのフラグは対応する割り込み許可ビットがLIN割り込み許可レジスタ(LINENIR)で設定(1)されているなら、割り込みを生成することができます(162頁の「割り込み」項をご覧ください)。

21.5.9. xxERRフラグ

LIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグはLIN異常レジスタ(LINERR)の全ビットの論理和(OR)です(「割り込み」をご覧ください)。

- LINビット異常(LBERR)

バス上にビットを送出する装置はバスの監視も行います。LINビット異常は監視されたビット値が送出したビット値と異なる時にフラグを立てられます。LINビット異常検出後の送信は中止されます。

- LINチェックサム異常(LCERR)

LINチェックサム異常は全ての受信データバイト(LIN2.1ではそれと保護された識別子)の総和の256の剰余の反転とチェックサムの加算が\$FFの結果でない場合にフラグを立てられます。

- LINパリティ異常(LPERR)

識別子(IDENTIFIER)領域内のLINパリティ異常はパリティビットの値が識別子値と一致しない場合にフラグを立てられます。ハードウェアはどんな修正の義務も負いません。けれども、LIN従装置応用は以下としてこれを解決しなければなりません。

- (パリティビットが不正な)既知の識別子、または
- 無視されるべき不正な識別子、または
- 新しい識別子

- LIN同期異常(LSERR)

LIN同期異常は従装置が与えられた許容外で同期(SYNC)領域の端(エッジ)を検出した場合にフラグを立てられます。

- LINフレーミング異常(LFERR)

フレーミング異常は優性の停止(STOP)ビットが採取された場合にフラグを立てられます。UART形態と同じ機能です。

- LINフレーム時間超過異常(LTOERR)

時間超過異常は同期(SYNC)と識別子(IDENTIFIER)の領域の送信によって、どれかの従作業によるメッセージフレームが最大長(TFrame_Maximum)内で完全に完了されなかった場合にフラグを立てられます(「フレーム時間超過」項をご覧ください)。

- LINオーバーラン異常(LOVERR)

オーバーラン異常は'多忙信号'が存在中に(LIN中断以外の)新規指令が入力された場合にフラグを立てられます。UART形態でのオーバーラン異常は受信したバイトが直列入力緩衝器に格納されたバイトを上書きした場合にフラグを立てられます。

- LIN中止(LABORT)

LIN転送中止は'多忙信号'が存在中の以前のLIN中止指令(LCMD2~0=000)を反映します。

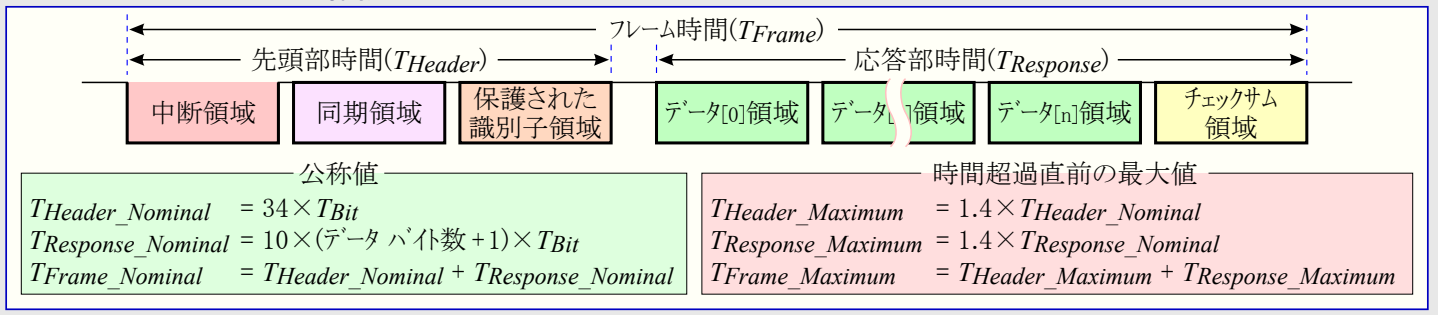
各LIN異常後、LIN制御器はその直前の活動を停止し、図21-11.で図解されるように撤回形態(LCMD2~0=000)に戻ります。

LINSIRレジスタのLERRへの書き込みはLERRビットとLINERRレジスタの全ビットをリセットします。

21.5.10. フレーム時間超過

LIN規約に従い、フレーム時間超過は $T_{Frame} > T_{Frame_Maximum}$ の場合にフラグを立てられます。この機能はLIN/UART制御器で実装されます。

図21-12. LINタイミングとフレーム時間超過



21.5.11. データ内での中断

LIN規約に従い、LIN/UART制御器は例えば応答のバイトで中断(BREAK)が部分的に重なっても、中断/同期(BREAK/SYNC)領域手順を検出することができます。中断/同期(BREAK/SYNC)領域手順が起これば、進行中の転送は中止され、新規フレームの処理が始まります。

- 従節点(ノード)では、異常が生成されます(換言すると、**応答送信**の場合に**ビット異常(LBERR)**、**応答受信**の場合に**フレーミング異常(LFERR)**)。データ異常の情報も利用可能で、「**異常後のデータ長**」を参照してください。
- 主節点(ノード)では、このフレームの中止に関して使用者(コード)が責任を持ちます。これを行うため、主作業は最初に進行中の通信を中止(LCMD2~0ビットを解除(=000:LIN中止指令))し、そして**先頭部送信**指令を与えます。この場合、**中止異常(LABORT)フラグ**が設定(1)されます。

従節点では、LIN/UART制御器が(中止した)応答を処理していた時に利用可能だった同期設定で中断(BREAK)検出が処理されます。けれども例の如く再同期が再始動し、時間超過値が僅かに不正確となり得ます。

21.5.12. チェックサム

フレームの最後の領域がチェックサムです。

LIN2.1に於けるチェックサムは保護された識別子と全データバイトに渡る桁溢れを伴う総和の論理反転8ビットを含みます。この計算は拡張チェックサムと呼ばれます。

$$\text{チェックサム} = 255 - \left(\text{unsigned char} \left(\sum_0^n \text{データ}_n \right) + \text{保護された識別子} \right) + \text{unsigned char} \left(\left(\sum_0^n \text{データ}_n \right) + \text{保護された識別子} \right) \gg 8$$

LIN1.3に於けるチェックサムは全データバイトに渡る桁溢れを伴う総和の論理反転8ビットを含みます。この計算は古典的チェックサムと呼ばれます。

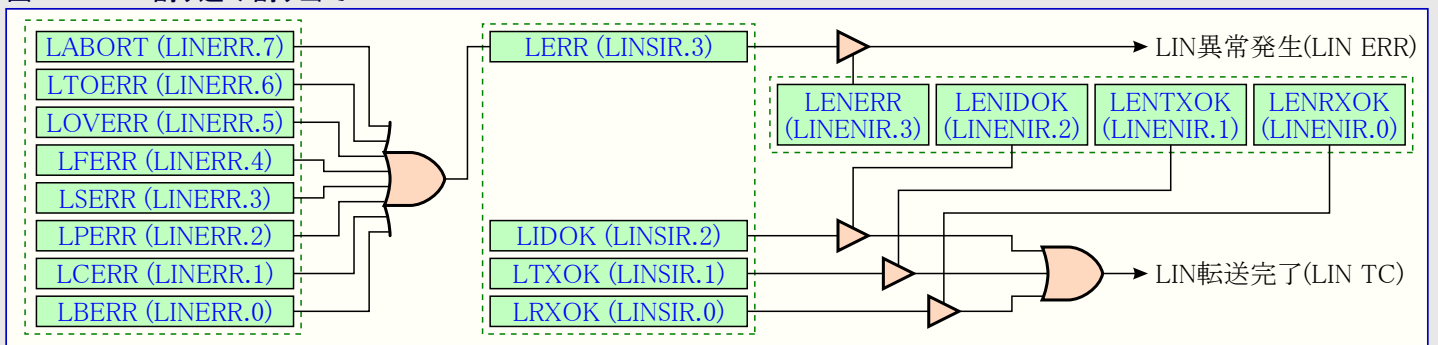
$$\text{チェックサム} = 255 - \left(\text{unsigned char} \left(\sum_0^n \text{データ}_n \right) + \text{unsigned char} \left(\left(\sum_0^n \text{データ}_n \right) \gg 8 \right) \right)$$

60(\$3C)~63(\$3F)のフレーム識別子は常に古典的チェックサムを使います。

21.5.13. 割り込み

下図で示されるように、LIN状態/割り込み要求レジスタ(LINSIR)の4つの通信フラグが2つ割り込みを駆動するように組み合わせられます。これらのフラグの各々はLIN割り込み許可レジスタ(LINENIR)に各々の割り込み許可ビットを持ちます(「**xxOKフラグ**」と「**xxERRフラグ**」をご覧ください)。

図21-13. LIN割り込み割り当て



21.5.14. メッセージ濾過

全ての識別子に基づくメッセージ濾過は実装されていません。識別子として\$3C,\$3D,\$3E,\$3Fを持つフレーム先頭部だけがLIN状態/割り込み要求レジスタ(LINSIR)で利用可能です。

60(\$3C)~63(\$3F)の識別子を持つメッセージは古典的チェックサム(データバイトだけに渡る総和)を用いること、LIN規約は言っています。ソフトウェアには、この期待したチェックサムを供給/検査するため、LIN1.3形態(LIN13)ビットを正確に切り替える(応答送信または応答受信の指令直後に、CRCの計算に識別子領域の挿入を含めるか否かの)責任があります。

表21-4. フレーム情報

LIDST2~0	フレーム情報
0 x x	特定識別子なし
1 0 0	60(\$3C)識別子
1 0 1	61(\$3D)識別子
1 1 0	62(\$3E)識別子
1 1 1	63(\$3F)識別子

21.5.15. データ管理

21.5.15.1. LIN FIFOデータ緩衝器

レジスタ配置位置を保つため、LINデータ緩衝器は(アドレス指示器でアクセス可能な)FIFOのように見えます。このFIFOはLINデータ緩衝器選択(LINSEL)レジスタのFIFO LINデータ緩衝器指示子(LINDX2~0)領域経由でLINデータレジスタ(LINDAT)を通してアクセスされます。

データ指示子(LINDX2~0)は必要とするデータバイトへの位置指示子です。データバイトは読み書きできます。データ緩衝器指示子自動進行(LAINC, Low活性)が解除(0)されている場合、データ指示子は各LINDATアクセス後、自動的に増加されます。超過巻き戻りが実装され、データ指示子=7後、それはデータ指示子=0になります。さもなければ、またはLAINCビットが設定(1)されている場合、各LINDATアクセス前にデータ指示子は書き込み(更新)が必要です。

LINフレームの先頭バイトはデータ指示子=0に、2つ目はデータ指示子=1に、以下同様に格納されます。にも拘らず、LINSELは使用前に使用者によって初期化されなければなりません。

21.5.15.2. UARTデータレジスタ

LINDATレジスタがデータレジスタです(緩衝なし、FIFOなし)。書き込みアクセスではLINDATがデータ出力用で、読み込みアクセスではLINDATがデータ入力用です。

UART形態に於けるLINSELレジスタは未使用です。

21.5.16. OCD支援

本節はOCDによって支援されるLIN/UART制御器の動き(換言すると、AVR Studio®でのI/Oウィンドウの動き)を記述します。

1. LIN制御レジスタ(LINCR)

- LINCRのビット6~0は読み書きアクセス可能です。
- ソフトウェアリセット(LSWRES)は常に自己リセットビット(実行には1マイクロコントローラ周期が必要)

2. LIN状態/割り込み要求レジスタ(LINSIR)

- 識別子状態(LIDST2~0)と多忙信号(LBUSY)は常に読み込み専用です。
- 異常割り込み要求(LERR)とLxxOKのビットは(実行でのようではなく、1または0書き込みによって直接設定(1)または解除(0)する)直接アクセス可能です。
- LERRの解除(0)がLINERRの全ビットをリセット(0)し、LERRの設定(1)がLINERRの全ビットを設定(1)することに注意してください。

3. LIN割り込み許可レジスタ(LINENIR)

- 全ビットが読み書きアクセス可能です。

4. LIN異常レジスタ(LINERR)

- 全ビットが読み書きアクセス可能です。
- LINSIRのLERR割り込み要求フラグを提供するためにLINERRのビットがORされることに注意してください。

5. LINビットタイミングレジスタ(LINBTR)

- LINビットタイミング(LBT5~0)はビットタイミング再同期禁止(LDISR)が設定(1)されている場合にだけ読み書きアクセスできます。
- LDISRがリセット(0)なら、LBT5~0は変更不能です。

6. LINボーレートレジスタ(LINBRRH,LINBRRL)

- 全ビットが読み書きアクセス可能です。

7. LINデータ長レジスタ(LINDLR)

- 全ビットが読み書きアクセス可能です。

8. LIN識別子レジスタ(LINIDR)

- 全ビットが読み書きアクセス可能です。
- パリティ(LP1,0)は読み込み専用で、動作進行中に常に更新されます。

9. LINデータ緩衝器選択(LINSEL)レジスタ

- 全ビットが読み書きアクセス可能です。

10. LINデータ(LINDAT)レジスタ

- 全ビットが読み書きアクセス可能です。
- データ緩衝器指示子自動進行(LAINC)は自動増加にもう効果を持たず、FIFO全体へのアクセスはLINSELのデータ指示子(LINDX2~0)を設定することで行われます。

注: デバッガの中断(ブレーク)が起こると、LIN/UART制御器の状態機構が(フレーム時間超過を含めて)停止され、更なる通信が不正になるかもしれません。

21.6. LIN/UART用レジスタ

21.6.1. LINCR – LIN制御レジスタ (LIN Control Register)

名称 : LINCR
 変位 : \$C8
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	LSWRES	LIN13	LCONF1,0	LENA	LCMD2~0			
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 – LSWRES : ソフトウェア リセット (Software Reset)

値	説明
0	動きなし
1	ソフトウェア リセット(このビットはリセット手順の最後で自己リセット(0)されます。)

● ビット6 – LIN13 : LIN1.3形態 (LIN1.3 mode)

値	説明
0	LIN2.1 (既定)
1	LIN1.3

● ビット5,4 – LCONF1,0 : 構成設定 (Configuration)

LINとUARTの形態に関する構成設定は下で示されます。

値	動作形態	
	LIN	UART
0 0 (既定値)	LIN標準形態 (聴取動作形態-OFF、CRC-ON、フレーム時間監視-ON)	8ビットデータ、パリティなし (聴取動作形態-OFF)
0 1	CRCなし、時間監視なし (聴取動作形態-OFF)	8ビットデータ、偶数パリティ (聴取動作形態-OFF)
1 0	フレーム時間監視なし (聴取動作形態-OFF、CRC-ON)	8ビットデータ、奇数パリティ (聴取動作形態-OFF)
1 1	聴取動作形態 (CRC-ON、フレーム時間監視-ON)	聴取動作形態、8ビットデータ、パリティなし

● ビット3 – LENA : 許可 (Enable)

値	説明
0	(LINとUARTの両形態)禁止
1	(LINとUARTの両形態)許可

● ビット2~0 – LCMD2~0 : 形態と指令 (Command and mode)

指令は許可(LENA)が設定(1)されている場合にだけ利用可能です。

値	説明
000	LIN先頭部受信 – LIN中止
001	LIN先頭部送信
010	LIN応答受信
011	LIN応答送信
100	UARTバイト転送
1x1	UARTバイト送信
11x	UARTバイト受信

21.6.2. LINSIR – LIN状態/割り込み要求レジスタ (LIN Status and Interrupt Register)

名称 : LINSIR

変位 : \$C9

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	LIDST2~0			LBUSY	LERR	LIDOK	LTXOK	LRXOK
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7~5 – LIDST2~0 : 識別子状態 (Identifier Status)

値	説明
0xx	特定識別子以外
100	60(\$3C)識別子
101	61(\$3D)識別子
110	62(\$3E)識別子
111	63(\$3F)識別子

● ビット4 – LBUSY : 多忙信号 (Busy Signal)

値	説明
0	多忙でない
1	送信または受信で多忙

● ビット3 – LERR : 異常割り込み要求フラグ (Error Interrupt)

これはLIN異常レジスタ(LINERR)の論理和(OR)です。このビットはLIN割り込み許可レジスタ(LINENIR)で該当する許可ビットの異常割り込み許可(LENERR)が設定(1)されている場合に割り込みを生成します。

この割り込みをリセットするため、使用者は1を書くことによってこのビットを解除(0)します。LERRのリセットはLIN異常レジスタ(LINERR)の全ビットもリセットします。UART形態では、LINデータ(LINDAT)レジスタを読むことによってもこのビットが解除(0)されます。

値	説明
0	異常なし
1	異常発生

● ビット2 – LIDOK : 識別子割り込み要求フラグ (Identifier Interrupt)

このビットはLIN割り込み許可レジスタ(LINENIR)で該当する許可ビットの識別子割り込み許可(LENIDOK)ビットが設定(1)されている場合に割り込みを生成します。

この割り込みをリセットするため、使用者は1を書くことによってこのビットを解除(0)します。

値	説明
0	識別子なし
1	従作業 : 識別子存在、主作業 : 先頭部送信完了

● ビット1 – LTXOK : 送信実行割り込み要求フラグ (Transmit Performed Interrupt)

このビットはLIN割り込み許可レジスタ(LINENIR)で該当する許可ビットの送信実行割り込み許可(LENTXOK)ビットが設定(1)されている場合に割り込みを生成します。

この割り込みをリセットするため、使用者は1を書くことによってこのビットを解除(0)します。UART形態では、LINDATレジスタに書くことによってもこのビットが解除(0)されます。

値	説明
0	送信なし
1	応答送信完了

● ビット0 – LRXOK : 受信実行割り込み要求フラグ (Receive Performed Interrupt)

このビットはLIN割り込み許可レジスタ(LINENIR)で該当する許可ビットの受信実行割り込み許可(LENRXOK)ビットが設定(1)されている場合に割り込みを生成します。

この割り込みをリセットするため、使用者は1を書くことによってこのビットを解除(0)します。UART形態では、LINデータ(LINDAT)レジスタを読むことによってこのビットが解除(0)されます。

値	説明
0	受信なし
1	応答受信完了

21.6.3. LINENIR – LIN割り込み許可レジスタ (LIN Enable Interrupt Register)

名称 : LINENIR

変位 : \$CA

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	LENERR	LENIDOK	LENTXOK	LENRXOK
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 – LENERR : 異常割り込み許可 (Enable Error Interrupt)

値	説明
0	異常割り込み遮蔽
1	異常割り込み許可

● ビット2 – LENIDOK : 識別子割り込み許可 (Enable Identifier Interrupt)

値	説明
0	識別子割り込み遮蔽
1	識別子割り込み許可

● ビット1 – LENTXOK : 送信実行割り込み許可 (Enable Transmit Performed Interrupt)

値	説明
0	送信実行割り込み遮蔽
1	送信実行割り込み許可

● ビット0 – LENRXOK : 受信実行割り込み許可 (Enable Receive Performed Interrupt)

値	説明
0	受信実行割り込み遮蔽
1	受信実行割り込み許可

21.6.4. LINERR – LIN異常レジスタ (LIN Error Register)

名称 : LINERR

変位 : \$CB

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	LABORT	LTOERR	LOVERR	LFERR	LSERR	LPERR	LCERR	LBERR
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 – LABORT : 中止フラグ (Abort Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	警告なし
1	LIN中止指令発生

● ビット6 – LTOERR : フレーム時間超過異常フラグ (Frame_Time_Out Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	フレーム時間超過異常

● ビット5 – LOVERR : オーバーラン異常フラグ (Overrun Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	オーバーラン異常

● ビット4 – LFERR : フレーミング異常フラグ (Framing Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	フレーミング異常

● ビット3 – LSERR : 同期異常フラグ (Synchronization Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	同期異常

● ビット2 – LPERR : パリティ異常フラグ (Parity Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	パリティ異常

● ビット1 - LCERR : チェックサム異常フラグ (Checksum Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	チェックサム異常

● ビット0 - LBERR : ビット異常フラグ (Bit Error Flag)

このビットはLIN状態/割り込み要求レジスタ(LINSIR)の異常割り込み要求(LERR)フラグが解除(0)される時に解除(0)されます。

値	説明
0	異常なし
1	ビット異常

21.6.5. LINBTR - LINビット タイミング レジスタ (LIN Bit Timing Register)

名称 : LINBTR

変位 : \$CC

リセット : \$20

特質 : -

ビット	7	6	5	4	3	2	1	0
	LDISR	-	LBT5~0					
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	1	0	0	0	0	0

● ビット7 - LDISR : ビット タイミング再同期禁止 (Disable Bit Timing Re-synchronization)

値	説明
0	ビット タイミング再同期許可 (既定)
1	ビット タイミング再同期禁止

● ビット5~0 - LBT5~0 : LINビット タイミング (LIN Bit Timing)

ビットの採取数(位置)を与えます。

$$\text{採取時間} = (1 \div \text{clkI/O}) \times (\text{LDIV}_{11 \sim 0} + 1)$$

既定値:LBT5~0=32、最小値:LBT5~0=8、最大値:LBT5~0=63

21.6.6. LINBRRH, LINBRRL (LINBRR) - LINボーレート レジスタ (LIN Baud Rate Register)

名称 : LINBRRL : LINBRRH (LINBRR)

変位 : \$CD : \$CE

リセット : \$0000

特質 : -

ビット	15	14	13	12	11	10	9	8
	-	-	-	-	LDIV11~8			
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	6	5	4	3	2	1	0
	LDIV7~0							
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11~0 - LDIV11~0 : 前置分周係数 (Scaling of clkI/O Frequency)

LDIV値は適切なLINまたはUARTのボーレートを達成するために入力するclkI/O周波数を分周するのに使われます。

21.6.7. LINDLR – LINデータ長レジスタ (LIN Data Length Register)

名称 : LINDLR
 変位 : \$CF
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	LTXDL3~0				LRXDL3~0			
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7~4 – LTXDL3~0 : 送信データ長 (LIN Transmit Data Length)**

LIN形態でのこの領域は送信されるべきバイト数を与えます(最大8で留め置かれます)。

UART形態でのこの領域は未使用です。

● **ビット3~0 – LRXDL3~0 : 受信データ長 (LIN Receive Data Length)**

LIN形態でのこの領域は受信されるべきバイト数を与えます(最大8で留め置かれます)。

UART形態でのこの領域は未使用です。

21.6.8. LINIDR – LIN識別子レジスタ (LIN Identifier Register)

名称 : LINIDR
 変位 : \$D0
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	LP1,0		LID5/LDL1	LID4/LDL0	LID3~0			
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7,6 – LP1,0 : パリティ (Parity)**

LIN形態では、 $LP1 = \text{!(LID5} \wedge \text{LID4} \wedge \text{LID3} \wedge \text{LID1)}$ 、 $LP0 = \text{LID4} \wedge \text{LID2} \wedge \text{LID1} \wedge \text{LID0}$

UART形態でのこの領域は未使用です。

● **ビット5,4 – LDL1,0 : LIN1.3データ長 (LIN1.3 Data Length)**

LIN1.3形態では下表のようになります。

LIN2.1形態でこれらのビットは6ビット識別子に属します(長さは転送されません。)

UART形態でのこの領域は未使用です。

値	説明
00	2バイト応答
01	2バイト応答
10	4バイト応答
11	8バイト応答

● **ビット5/3~0 – LID5/3~0 : LIN1.3/2.1識別子 (LIN1.3/2.1 Identifier)**

LIN1.3形態 : 4ビット識別子

LIN2.1形態 : 6ビット識別子(長さは転送されません。)

UART形態でのこの領域は未使用です。

21.6.9. LINSEL – LINデータ緩衝器選択レジスタ (LIN Data Buffer Selection Register)

名称 : LINSEL
 変位 : \$D1
 リセット : \$x0
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	-	-	-	LAINC	LINDX2~0		
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

● **ビット3 – LAINC : データ緩衝器指示子の自動進行 (Auto Increment of Data Buffer Index)**

LIN形態では下表のようになります。

UART形態でのこのビットは未使用です。

値	説明
0	FIFOデータ緩衝器位置指示子の自動進行 (既定)
1	自動進行なし

● **ビット2~0 – LINDX2~0 : FIFO LINデータ緩衝器位置指示子 (FIFO LIN Data Buffer Index)**

LIN形態では、FIFOデータ緩衝器ないのLIN応答データバイトの位置(指示子)です。FIFOデータ緩衝器はLINデータ(LINDAT)レジスタを通してアクセスされます。

UART形態でのこの領域は未使用です。

20.6.10. LINDAT – LINデータレジスタ (LIN Data Register)

名称 : LINDAT
 変位 : \$D2
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	LDATA7~0							
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 – LDATA7~0 : LINデータ入出力 (LIN Data in / Data out)**

LIN形態では、FIFOデータ緩衝器のポートです。

UART形態では、データレジスタ(データ緩衝器なし、FIFOなし)です。

- 書き込みアクセスはデータ出力
- 読み込みアクセスはデータ入力

22. ADC – A/D変換器

22.1. 特徴

- 10ビット分解能
- 積分非直線性誤差0.5 LSB
- 絶対精度±2 LSB
- 変換時間8～250µs
- 最大分解能で120kSPS(採取/s)まで
- 11チャンネルのシングル エント 入力多重器内蔵
- 5%精度で5,10,20,40倍に設定可能な3つの差動入力チャンネル
- 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2.56V A/D変換基準電圧
- 連続と単独の変換動作
- 割り込み元の自動起動によるA/D変換開始
- A/D変換完了割り込み
- 休止形態雑音低減機能
- 温度感知器
- LINアドレス感知(ISRC電圧測定)
- VCC電圧測定

22.2. 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11のシングル エント 電圧入力を許す15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 エント 電圧入力は0V(GND)が基準です。

デバイスはA/D変換前に差動入力電圧で14dB(×5)、20dB(×10)、26dB(×20)、32dB(×40)の増幅段を提供する設定変更可能な利得段が装備された3つの差動電圧入力増幅器も支援します。増幅したチャンネルでは8ビット分解能が期待できます。

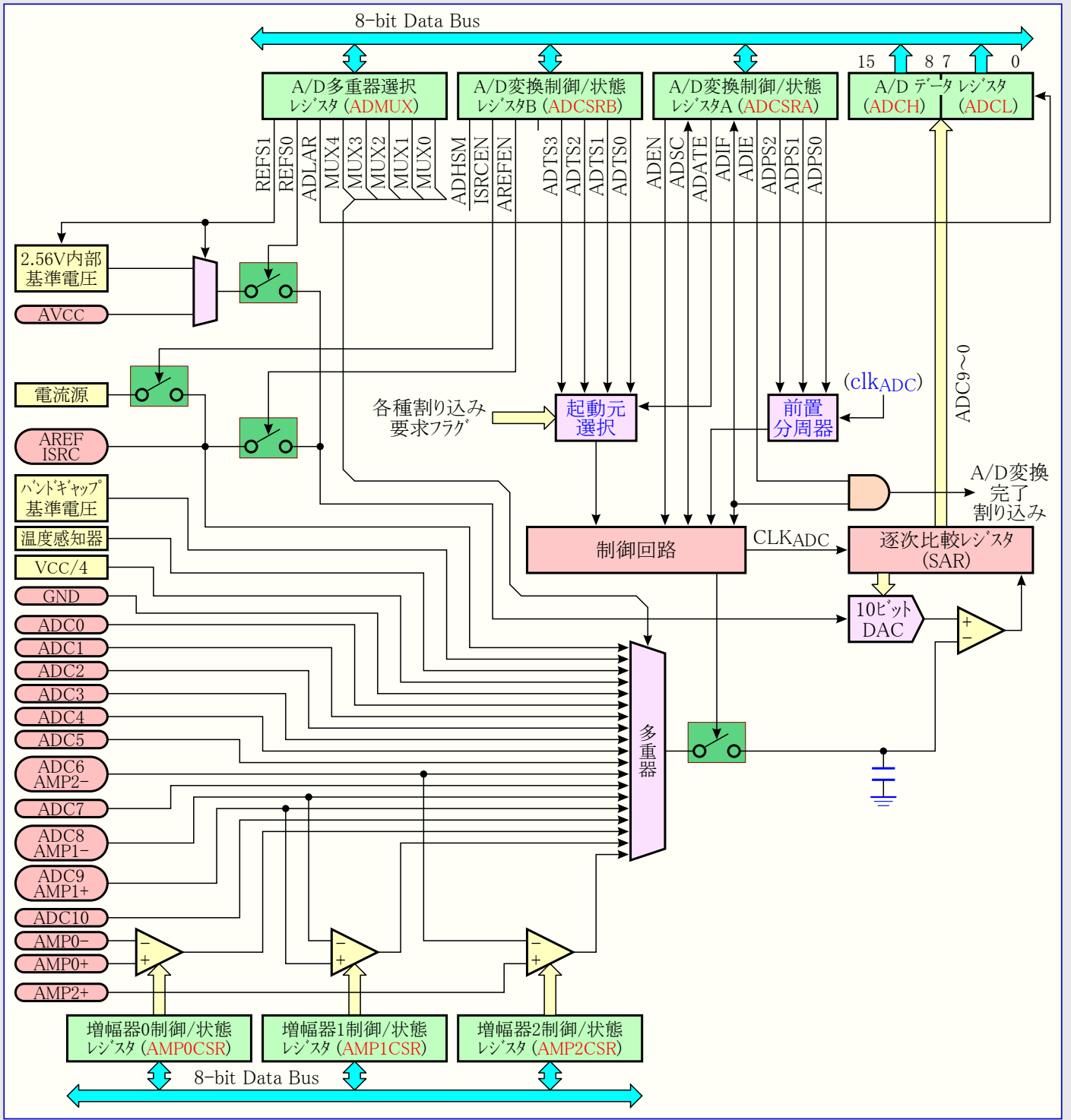
このA/D変換器はA/D変換器への入力電圧が変換中に一定の値で保持されることを保証する採取&保持(S/H)回路を含みます。A/D変換部の構成図は以下で示されます。

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2.56V基準電圧を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22-1. A/D変換器部構成図



関連リンク [30頁の「PM - 電力管理と休止形態」](#)
[31頁の「電力削減レジスタ」](#)

22.3. 操作

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

アナログ入力チャンネルはADMUXのチャンネル選択(MUX4~0)ビットへの書き込みによって選ばれます。GNDとバントギャップ固定基準電圧だけでなく、どのADC入力ピン(ADC10~0)もA/D変換器のシングルエンド入力として選べます。

A/D変換部はA/D変換制御/状態レジスタA(ADCSRA)のA/D許可(ADEN)ビットの設定(1)によって(動作が)許可されます。基準電圧はA/D変換部が許可されるかされないかがどうでも、ADMUXレジスタのREFS1とREFS0ビットによって設定されます。ADENが解除(0)されているとA/D変換部は電力を消費しないので、節電をする休止形態へ移行する前にA/D変換部をOFFに切り替えることが推奨されます。

A/D変換部はA/Dデータレジスタ(ADCH,ADCL)で示される10ビットの結果を生成します。既定では、この結果は右揃え(16ビットのビット0側10ビット)で表されますが、ADMUXで左揃え選択(ADLAR)ビットを設定(1)することにより、任意で左揃え(16ビットのビット15側10ビット)で表せます。

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

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

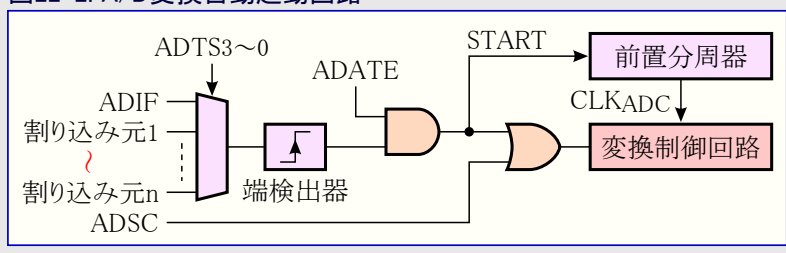
22.4. 変換の開始

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22-2. 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。連続動作は増幅したチャンネルで許されません。

自動起動が許可されている場合、ADCSRAのADSCビットに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)

22.5. 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50～2MHzの入力クロック周波数を必要とします。10ビットよりも低い分解能が必要とされるなら、A/D変換器への入力クロック周波数はより高い採取速度を得るために2MHzよりも高くできます。

A/D変換部は100kHz以上のどんなCPUクロックからも受け入れ可能なA/D変換クロック周波数を生成する前置分周器を含みます。この前置分周はA/D変換制御/状態レジスタ(ADCSRA)のA/Dクロック選択(ADPS2～0)ビットによって設定されます。前置分周器はADCSRA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15.5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3.5変換クロック周期、初回変換の開始後13.5変換クロック周期で行われます。変換が完了すると、結果がA/Dデータレジスタ(ADCH,ADCL)に書かれ、ADCSRAのA/D変換完了割り込み要求フラグ(ADIF)が設定(1)されます。単独変換動作(ADATE=0)では同時にADCSRAの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22-3. A/D変換前置分周器部構成

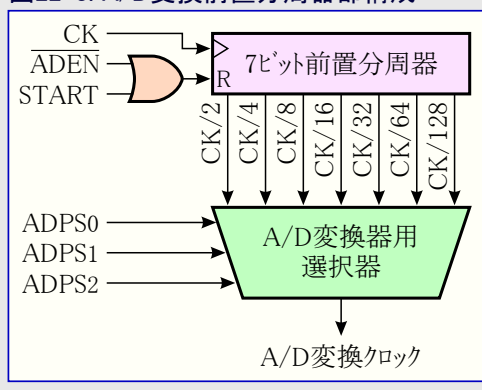


図22-4. 初回変換タイミング (単独変換動作)

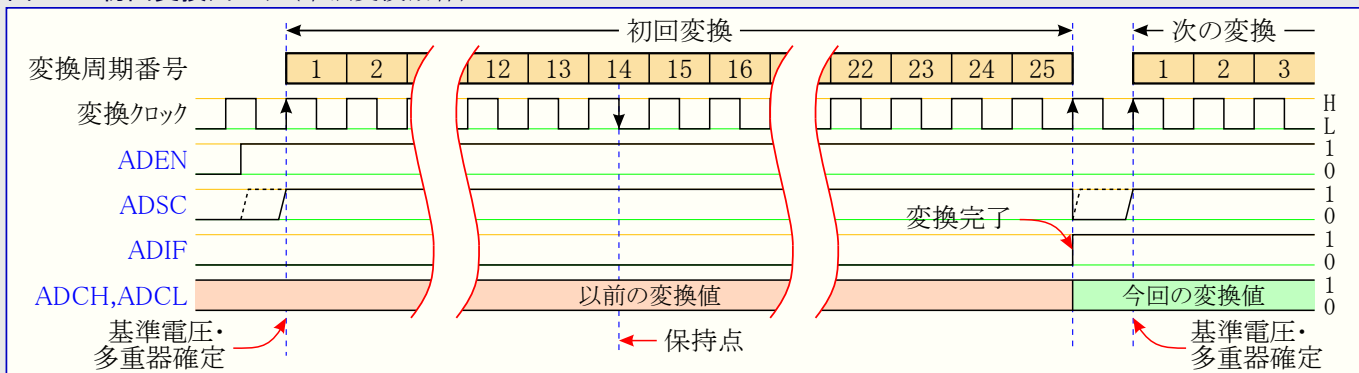
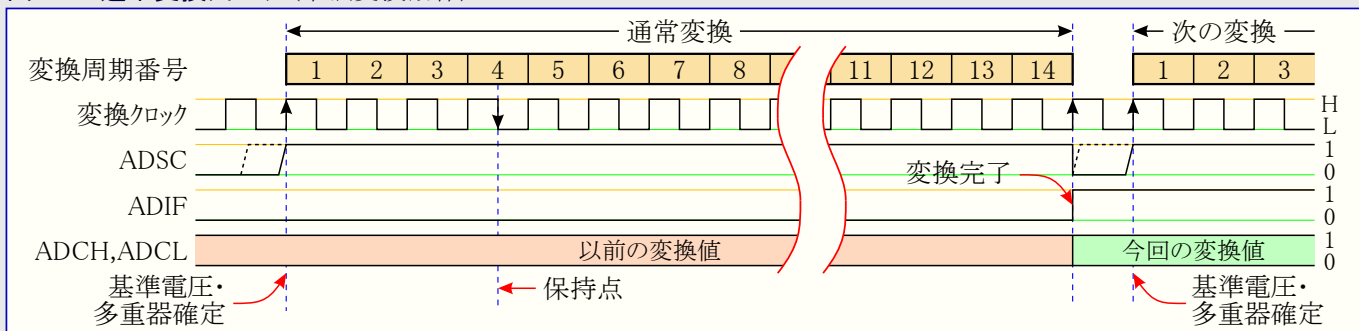


図22-5. 通常変換タイミング (単独変換動作)



(訳注) 上記本文中の通常変換クロック数を表22-1.に従って原書の13から15.5に変更しています。図22-5.は原書に対して保持点を修正していますが他は原書に従っています。この図の変換クロック数は14ですが、表22-1.では15.5と記載されていて矛盾します。

図22-6. 通常変換タイミング* (自動起動変換動作)

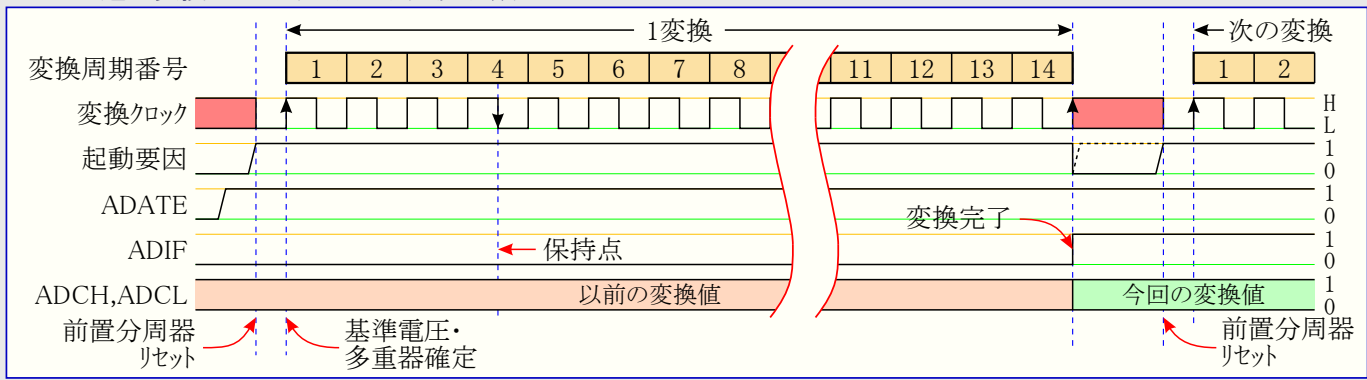


図22-7. 連続変換動作タイミング*

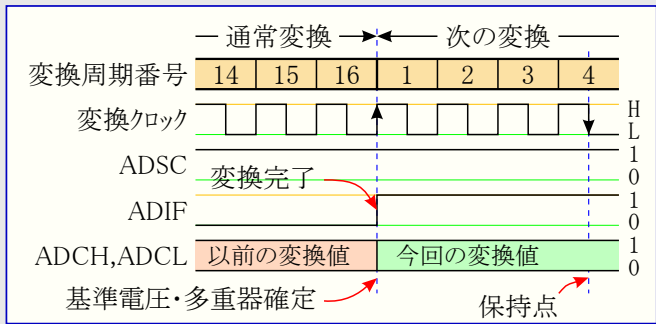


表22-1. A/D変換時間

変換種別	保持点	変換時間
初回変換	13.5	25
シングルエントリ通常変換	3.5	15.5
自動起動変換	2	16

(訳注) 図22-6.と図22-7.は原書に従っていますが、図22-6.の変換クロック数は14で表22-1.の16と矛盾し、図22-7.の保持点は3.5で表22-1.の2と矛盾します。

22.6. 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(MUX4~0)ビットと基準電圧選択(REFS1,0)ビットはCPUが乱順で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払われなければなりません。

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

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。

差動入力チャネルを変更する時に特別な注意が払われるべきです。一旦差動入力チャネルが選ばれてしまうと、利得段は新しい値に安定するのに125µs程度かかるかもしれません。従って変換は新規差動入力チャネル選択後の最初の125µs内に開始されるべきではありません。または、この期間内に得た変換結果は破棄されるべきです。(ADMUXのREFS1,0ビットの変更による)A/D変換器基準電圧変更後の最初の差動入力変換に対して、同じ設定(安定)時間が厳守されるべきです。

22.6.1. 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。
- 連続変換動作で、増幅器が増幅チャネル変換の最後でADSCビットを解除(0)するので、各変換の最後でADSCビットがソフトウェアによって再び設定(1)される場合を除き、連続変換動作の使用は不可能です。

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いかも知れません。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。

22.6.2. 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2.56V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してA/D変換部に接続されます。2.56V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2.56V基準電圧間の切り替えができます。基準電圧源切り替え後の最初のA/D変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

AREFピンは電流源出力(ISRC)との交換機能です。電流源が選ばれると、AREFピンは内部基準電圧網に接続されません。「ADCSRBA/D変換制御/状態レジスタ」項のアナログ基準電圧ピン許可(AREFEN)と電流源許可(ISRCEN)のビットをご覧ください。

差動入力チャネルが使われる場合、選んだ基準電圧は「電気的特性」の「A/D変換器特性」で示されるよりもAVCCに近くすべきではありません。

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

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ADCSRのADENへ0を書くことが推奨されます。

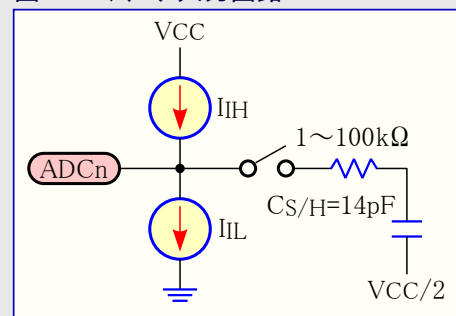
22.7.1. アナログ入力回路

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 Ω 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22-8. アナログ入力回路

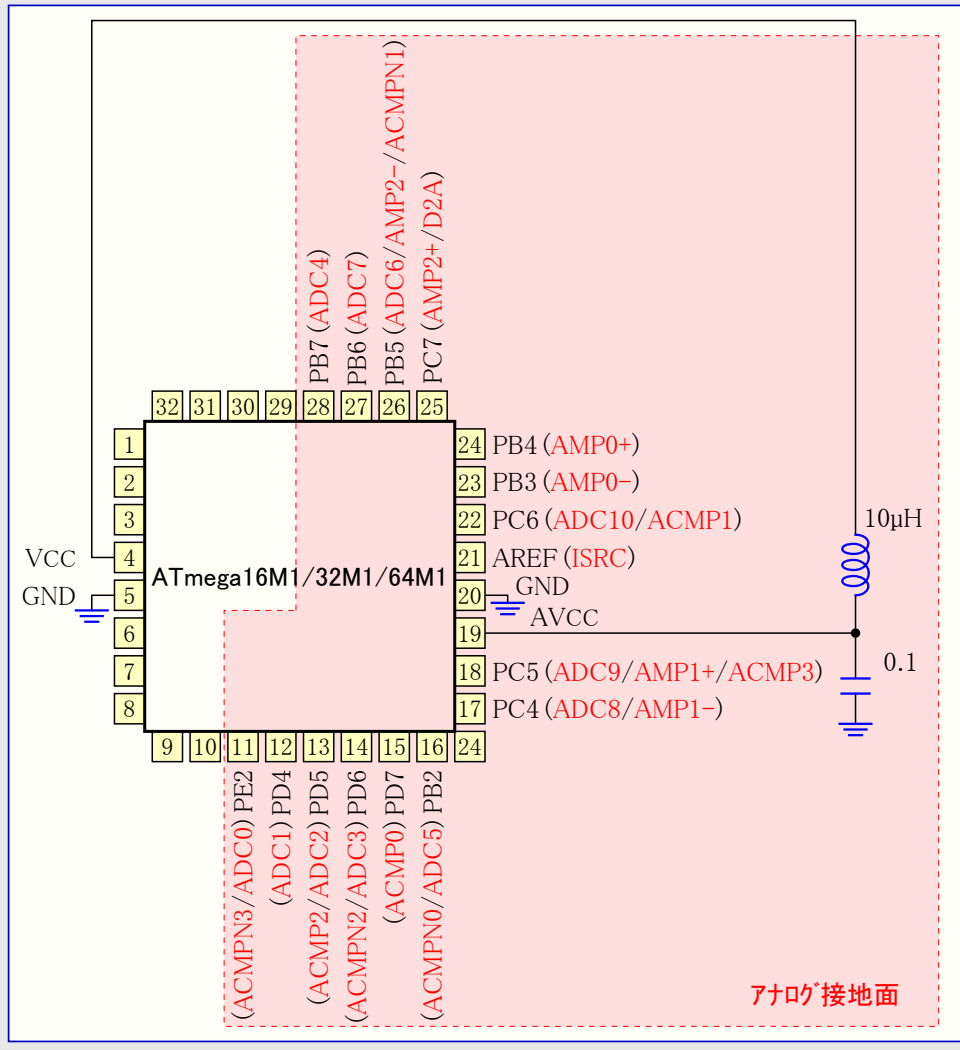


22.7.2. アナログ雑音低減技術

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ピンは下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ADCポートピンがデジタル出力として使われる場合、これらが変換進行中に切り替わらないことが重要です。

図22-9. A/D変換部電源接続



22.7.3. 変位(オフセット)補償の仕組み

利得段には差動測定の変位(オフセット)をできるだけ無効にする組み込み変位補正回路があります。アナログ経路内の残留変位は両入力未接続でAMPnIS_{BIT}を使う両差動入力短絡によって直接的に計測できます(「AMPnCSR - 増幅器n制御/状態レジスタ」をご覧ください)。その後、この残留変位は計測結果からソフトウェアで減算できます。この変位補正に基いたソフトウェア手法の使用はどのチャンネルの変位も1 LSB以下に減少することができます。

22.7.4. A/D変換の精度定義

シングルエンド入力電圧のnビットA/D変換はGNDとVREF間を2ⁿ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2ⁿ-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22-10.

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22-11.

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22-12.

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22-13.

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±0.5 LSBです。

• 絶対精度

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

図22-10. 変位(オフセット)誤差

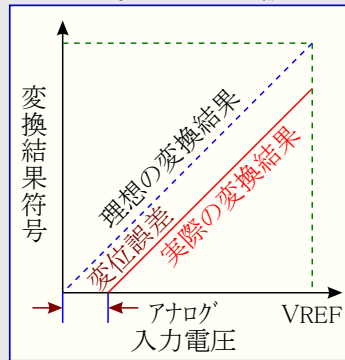


図22-11. 利得誤差

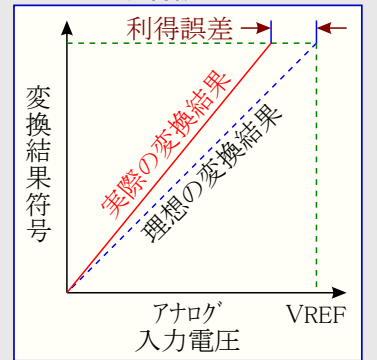


図22-12. 積分非直線性誤差

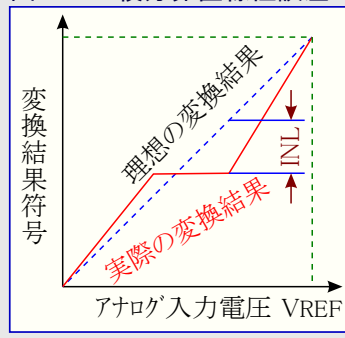
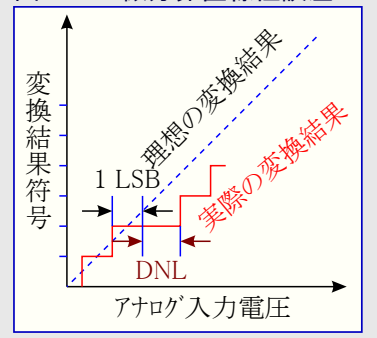


図22-13. 微分非直線性誤差



22.8. 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。

VINは選んだ入力ピンの電圧で、VREF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。

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

差動チャンネルが使われると、その結果は右式で示されます。VPOSは非反転入力ピンの電圧、VNEGは反転入力ピンの電圧、GAINは選んだ倍率、VREFは選んだ基準電圧です。この結果は-512(\$200)~+511(\$1FF)の2の補数形式で示されます。使用者が結果の符号を素早く検査したいなら、結果のMSB(ADCHのADC9)を読むことで充分なことに注目してください。このビットが1ならばその結果は-、0ならばその結果は+です。下図は差動入力範囲の符号化を示します。

$$ADC = \frac{(V_{POS} - V_{NEG}) \times GAIN \times 512}{V_{REF}}$$

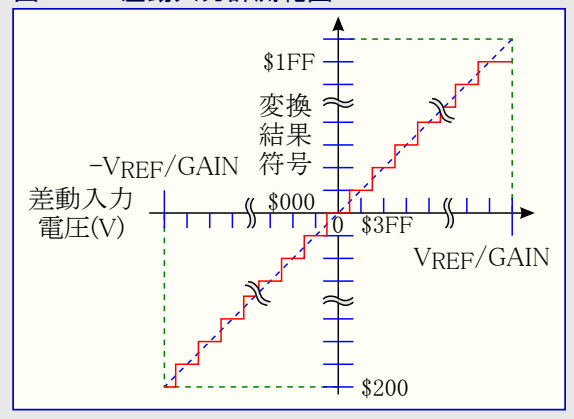
下表は差動入力チャンネルの組(ADCn-ADCm)が基準電圧(VREF)で選ばれた場合の出力符号の結果を示します。

表22-2. 入力電圧と出力符号の関係

VADCn	読み出し符号	対応する10進値
VADCm+VREF/GAIN	\$1FF	511
VADCm+(511/512)VREF/GAIN	\$1FF	511
VADCm+(510/512)VREF/GAIN	\$1FE	510
}	}	}
VADCm+(1/512)VREF/GAIN	\$001	1
VADCm	\$000	0
VADCm-(1/512)VREF/GAIN	\$3FF	-1
}	}	}
VADCm-(511/512)VREF/GAIN	\$201	-511
VADCm-VREF/GAIN	\$200	-512

例: ADMUX=\$ED(ADC3とADC2差動10倍入力、VREF=2.56V、左揃え)で、ADC3が300mV、ADC2が500mVの場合、A/D変換値=512×10×(300-500)÷2560=-400=\$270
従って、ADCH=\$9C、ADCL=\$00、ADLAR=0なら、ADCH=\$02、ADCL=\$70。

図22-14. 差動入力計測範囲



22.9. 温度測定

温度測定はシングルエンド温度感知器チャネルと対にされた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A/Dチャネル選択(MUX4~0)ビットに'1000'を書くことによる温度感知器チャネル選択がこの温度感知器を許可します。温度感知器測定でのA/D変換基準電圧源用に内部2.56V基準電圧も選ばれなければなりません。温度測定が許可されると、A/D変換器は温度感知器上の電圧を測定するのに単独変換動作で使うことができます。

測定した電圧は右表で記述される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2.5mV/°Cで精度は使用者校正の方法に依存します。代表的に、単一温度校正後の測定精度は、室温での校正と仮定して±10°Cです。より良い精度は校正に対して2つの温度点を使うことによって達成されます。

表22-3. 温度対感知器出力電圧(代表条件)

温度(°C)	-40°C	+25°C	+85°C
電圧(mV)	600mV	762mV	912mV

右表に記載した値は代表値です。然しながら製法変化のため、温度感知器出力電圧は或るチップと別のチップで変化します。より正確な結果達成を可能とするために温度測定は応用ソフトウェアで校正することができます。校正ソフトウェア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。校正ソフトウェアは次式を使って行なえます。

$$T(\text{温度}) = k \times (\text{ADCH} \ll 8 \mid \text{ADCL}) + T_{OS}$$

ここでADCH:ADCLはA/D変換器データレジスタ、 T はケルビンでの絶対温度、 k は固定傾斜係数、 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。代表的に、 k は非常に1.0に近く、1点校正でこの係数は省略されるかもしれませんが。より高い精度が必要とされる場所では、この傾斜係数は2つの温度での測定に基づいて評価されるべきです。

22.9.1. 使用者校正

ソフトウェア校正は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次式を使って行うことができます。

$$T(\text{温度}) = ((\text{ADCH} \ll 8 \mid \text{ADCL}) - T_{OS}) \div k$$

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} は決定されてEEPROM内に格納される温度感知器変位(オフセット)値です。

22.9.2. 製造時校正

識票列で利用可能な校正値を使うこともできます。

校正値は概ね85°Cの高温での検査中に測定された値から決められます。

摂氏での温度は次式を使って計算することができます。

$$T(\text{温度}) = ((\text{ADCH} \ll 8 \mid \text{ADCL}) \times TSGAIN) + TSOFFSET - 273$$

ここで、

- ADCH:ADCLはA/D変換器データレジスタ
 - $TSGAIN$ は温度感知器利得(1一定、または符号なし固定小数点数の\$80=10進数の1.0)
 - $TSOFFSET$ は温度感知器変位修正項(2の補数符号付きバイト)
- です。

22.10. 増幅器

Atmel ATmega16M1/32M1/64M1は5,10,20,40倍の利得段付きの3つの作動増幅チャネルが特徴です。

この増幅器がスイッチ キャパシタ増幅器のため、それらは本文書で増幅器同期クロックと呼ばれる同期信号によってクロック駆動される必要があります。

結果精度を保証するため、この増幅器入力には最低4増幅器同期クロック周期間、採取点で静定入力値を持つ必要があります。

増幅変換は電力段制御器(PSC)事象または8 A/D変換クロックに等しい内部クロックCKADCに同期化され得ます。この同期化がA/D変換クロックの8分周で行なわれる場合、この同期化は採取&保持がCKADC2の特定位相で起きるといような方法でA/D変換器インターフェースによって自動的に行われます。CKADC2がLowの時に使用者によって開始した変換(即ち全ての単独変換と連続変換の最初)はシングル エンド入力変換(前置分周した次のクロック周期から13変換クロック周期)と同じ変換時間になります。CKADC2がHighの時に使用者によって開始した変換は同期化機構のため14変換クロック周期になります。

増幅器を使うための一般的な方法は増幅器n制御/状態レジスタ(AMPnCSR)の起動元選択(AMPnTS2~0)ビット経由で同期化クロックを選ぶことです。その後、増幅器はONに切り替えでき、増幅は各同期事象で行なわれます。

増幅チャネルで増幅したA/D変換を始めるには表22-5.で詳述されるようにA/D変換多重器選択(ADMUX)レジスタを設定しなければなりません。

A/D変換開始の必要条件はA/D変換制御/状態レジスタA(ADCSRA)でA/D変換開始(ADSC)ビットを設定(1)することによって行なわれます。

その変換が達成されないまでの間、別チャネルでの変換を開始することは不可能です。

増幅器同期化動作を良く理解するため、下で2つのタイミング図の例が示されます。

変換がADSCビットによって要求されると直ぐにA/D変換が開始されます。増幅器出力がA/D変換器の採取段階の間に変更される場合、実行中の変換は中止され、増幅器の出力が安定すると直ぐに改めて開始されます。これは高速応答時間を保証します。唯一行う注意はADCclk/4よりも低い起動信号(PSC)周波数を保証することです。

図22-15. 増幅器同期化タイミング構成図 (アナログ入力信号での変更との)

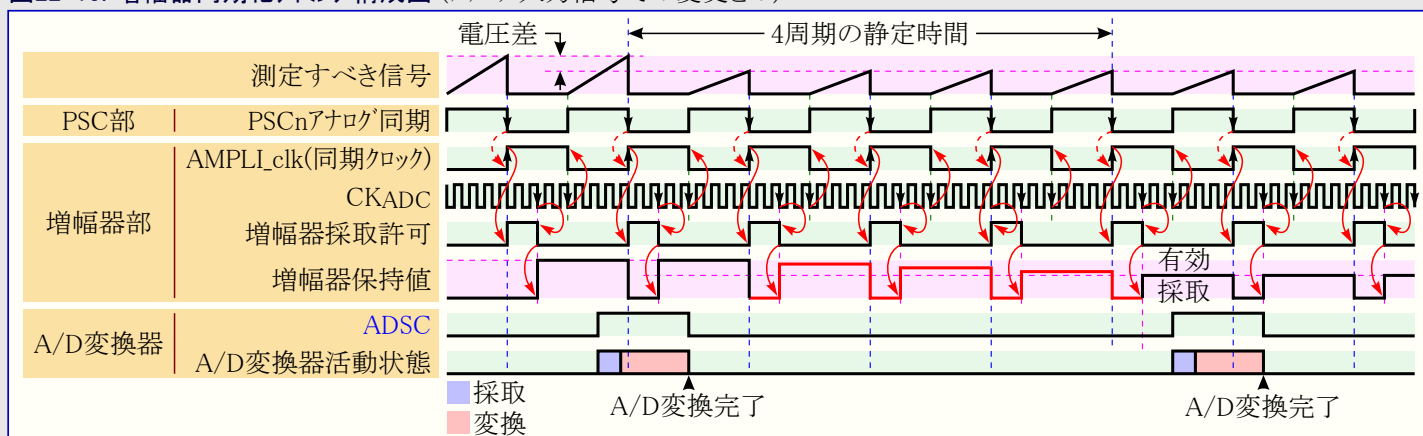
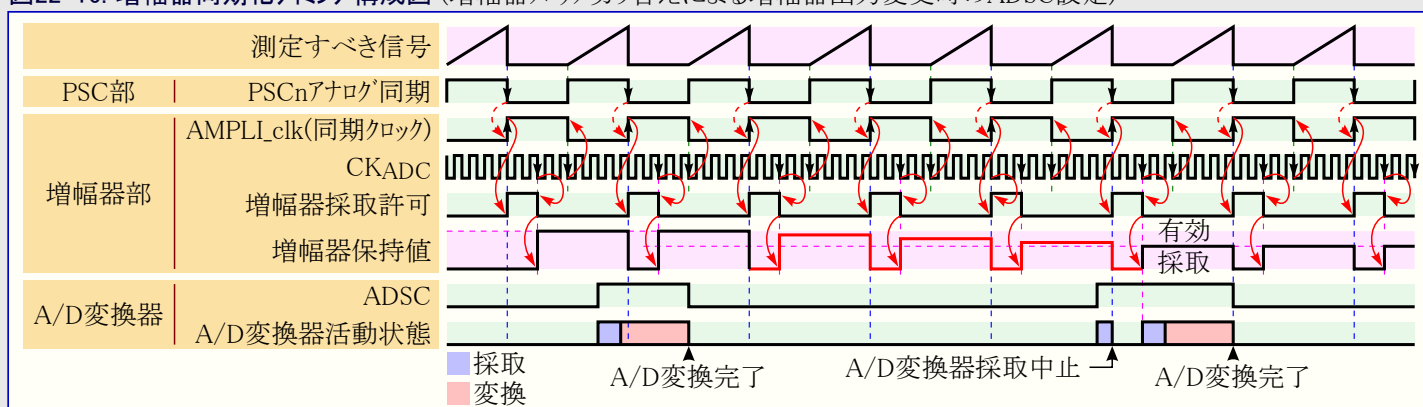


図22-16. 増幅器同期化タイミング構成図 (増幅器クロック切り替えによる増幅器出力変更時のADSC設定)

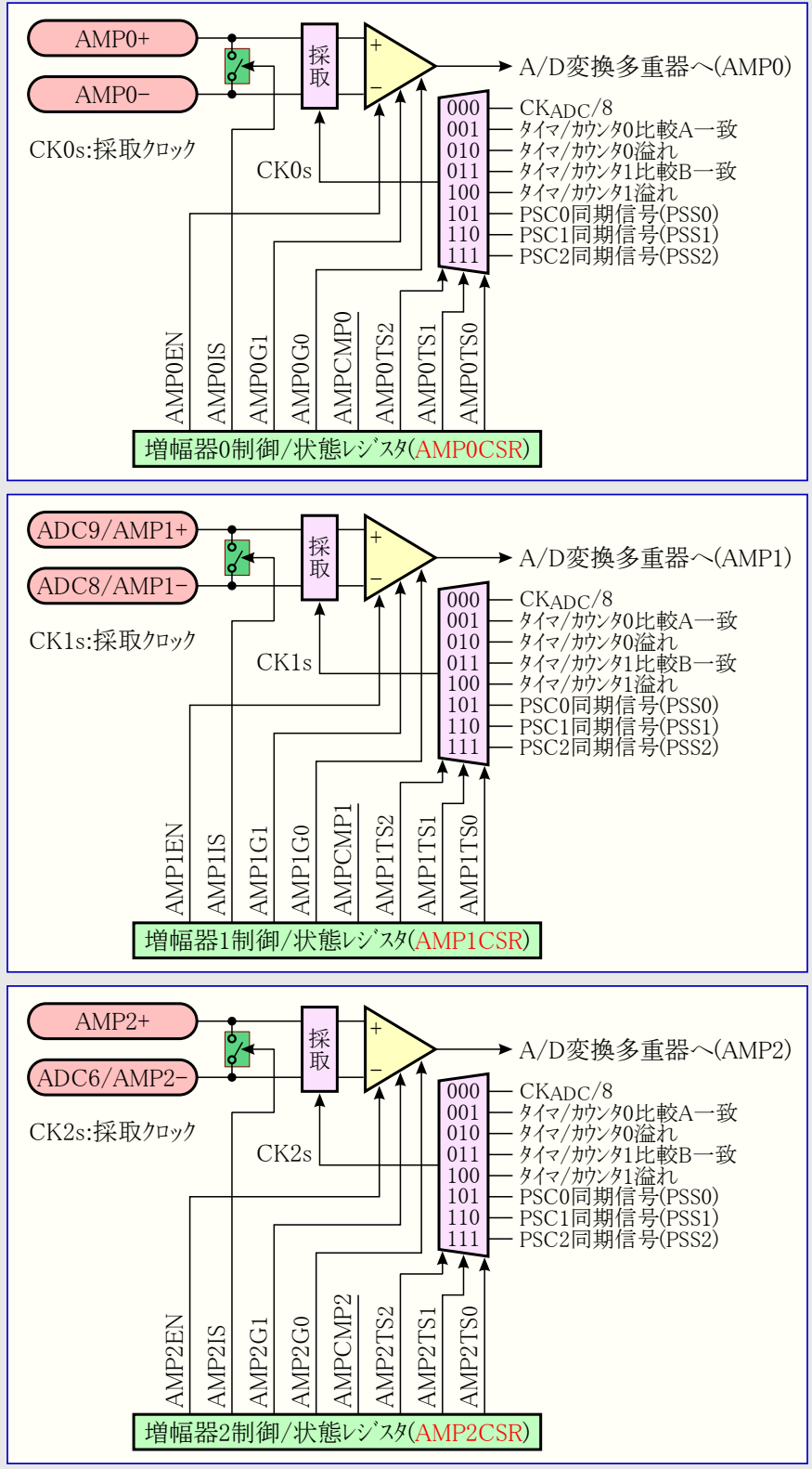


増幅器同期化動作を良く理解するため、タイミング図の例が上の図で示されます。

増幅チャネルでの自動起動変換も可能です。この場合の変換はA/D変換制御/状態レジスタB(ADCSRB)の自動起動要因選択(ADTS3~0)ビットで選んだ最後の自動起動事象に後続する最初の増幅器クロック事象で開始されます。自動起動変換での連続変換動作は各変換後にソフトウェアによってADCSRAのADSCビットが設定(1)される場合を除いて不可能です。

3つの増幅器の構成図が下で示されます。

図22-17. 増幅器構成図



22.11. A/D変換用レジスタ

22.11.1. 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	MUX4~0				
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7,6 – REFS1,0 : 基準電圧選択 (ADC VREF Selection Bits)

これらのビットはA/D変換器用の基準電圧を決めます。変換中にこれらのビットが変更されると、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。内部電圧基準(AVCCまたは内部2.56V基準電圧)任意選択は外部基準電圧がAREFピンに印加される場合に使われません。

表22-4. A/D変換部の基準電圧選択

ISRCEN	AREFEN	REFS1,0	基準電圧
0	1	0 0	AREFピンの外部基準電圧 (内部の基準電圧は切り離されます。)
	1	0 1	AVCC (AREFピンにデカップ用コンデンサが接続できます。)
	0	0 1	AVCC (AREFピンにデカップ用コンデンサを接続できません。)
	1	1 0	(予約)
	1	1 1	内部2.56V基準電圧 (AREFピンにデカップ用コンデンサが接続できます。)
x	0	1 1	内部2.56V基準電圧

内部VREF(AVCCまたは内部2.56V基準電圧)が選ばれる場合、これを必要とするアナログ機能が(活動に)設定されると直ぐにONへ切り替わります。(訳注:先行する内容が重複する行を削除)

● ビット5 – ADLAR : A/D変換 左揃え選択 (ADC Left Adjust Result)

ADLARビットはA/Dデータレジスタ内の変換結果の配置に影響を及ぼします。結果を左揃えにするにはADLARに1を書いてください。さもなければ結果は右揃えです。ADLARビットの変更はどんな進行中の変換にも拘らず、直ちにA/D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4~0 – MUX4~0 : A/Dチャネル選択 (Analog Channel Select Bits 4~0)

これら5ビットはどのアナログ入力A/D変換器入力に接続されるかを決めます。これらのビットが変換中に変更される場合、その変更は変換が完了する(ADCSRAのADIF=1)まで実施しません。

表22-5. アナログ入力チャネル選択

MUX4~0	A/D変換器入力	MUX4~0	A/D変換器入力
0 0 0 0 0	ADC0 (PE2)	0 1 0 1 0	ADC10 (PC6)
0 0 0 0 1	ADC1 (PD4)	0 1 0 1 1	温度感知器
0 0 0 1 0	ADC2 (PD5)	0 1 1 0 0	VCC/4
0 0 0 1 1	ADC3 (PD6)	0 1 1 0 1	電流源(ISRC)電圧
0 0 1 0 0	ADC4 (PB7)	0 1 1 1 0	AMP0 (-はPB3,+はPB4)
0 0 1 0 1	ADC5 (PB2)	0 1 1 1 1	AMP1 (-はPC4/ADC8,+はPC5/ADC9)
0 0 1 1 0	ADC6 (PB5)	1 0 0 0 0	AMP2 (-はPB5/ADC6,+はPC7)
0 0 1 1 1	ADC7 (PB6)	1 0 0 0 1	バンドギャップ電圧(VBG)
0 1 0 0 0	ADC8 (PC4)	1 0 0 1 0	0V(GND)
0 1 0 0 1	ADC9 (PC5)	10011~11111	(予約)

22.11.2. ADCSRA – A/D変換 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切り替えは変換の最後で実施します。

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

単独変換動作で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。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。単独変換動作へ戻すにはこのビットを解除(0)してください。この起動元はA/D変換制御/状態レジスタB(ADCSRB)のA/D変換起動要因選択(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22-6. A/D変換 前置分周器選択

ADPS2~0	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
分周数	2	2	4	8	16	32	64	128

22.11.3. 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
	ADHSM	ISRCEN	AREFEN	-	ADTS3~0			
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 – ADHSM : 高速動作 (ADC High Speed Mode)

このビットの1書き込みがA/D変換器高速動作を許可します。200kHzよりも高いA/D変換クロック周波数で変換をしたい場合、このビットを設定(1)してください。

● ビット6 – ISRCEN : 電流源許可 (Current Source Enable)

AREFピンに100μAの電流を吐き出すにはこのビットを設定(1)してください。AREFピンをアナログ基準電圧ピンとして使うにはこのビットを解除(0)してください。

● **ビット5 – AREFEN : アナログ基準電圧ピン許可 (Analog Reference pin Enable)**

AREFピンに内部AREF回路を接続するにはこのビットを設定(1)してください。AREFピンから内部AREF回路を切断するにはこのビットを解除(0)してください。

● **ビット3~0 – ADTS3~0 : A/D変換自動起動要因選択 (ADC Auto Trigger Source)**

これらのビットはA/D変換器が自動起動動作で動く場合にだけ必要です。これはA/D変換制御/状態レジスタA(ADCSRA)のA/D変換自動起動許可(ADATE)ビットが設定(1)される場合の意味です。

これら4ビットは変換開始起動を生成する割り込み事象を選びます。変換の開始は割り込みが許可であろうとなかろうと、選んだ割り込み要求フラグの上昇端によって生成されます。電力段制御器(PSC)nアナログ同期事象の場合、フラグがありません。従って、この場合の変換は直前の変換が完了され、この起動事象が現れる毎に開始します。

値	起動元	値	起動元
0 0 0 0	連続変換動作	1 0 0 0	電力段制御器(PSC)単位部1同期信号
0 0 0 1	外部割り込み要求0	1 0 0 1	電力段制御器(PSC)単位部2同期信号
0 0 1 0	タイマ/カウンタ0比較A一致	1 0 1 0	アナログ比較器0
0 0 1 1	タイマ/カウンタ0溢れ	1 0 1 1	アナログ比較器1
0 1 0 0	タイマ/カウンタ1比較B一致	1 1 0 0	アナログ比較器2
0 1 0 1	タイマ/カウンタ1溢れ	1 1 0 1	アナログ比較器3
0 1 1 0	タイマ/カウンタ1捕獲要求	1 1 1 0	(予約)
0 1 1 1	電力段制御器(PSC)単位部0同期信号	1 1 1 1	(予約)

22.11.4. ADCH,ADCL – A/D変換データレジスタ上位/下位バイト (ADC Data Register Low and High byte) [ADLAR=0]

ADCHとADCLのレジスタ対は16ビット値のADCデータレジスタを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「[16ビットレジスタのアクセス](#)」をご覧ください。

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(MUX4~0)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 : ADCHとADCL
 変位 : \$78
 リセット : \$0000
 特質 : ADLAR=0

ビット	15	14	13	12	11	10	9	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	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9~0 – ADC9~0 : 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。

22.11.5. ADCH,ADCL – A/D変換データレジスタ上位/下位バイト (ADC Data Register Low and High byte) [ADLAR=1]

ADCHとADCLのレジスタ対は16ビット値のADCデータレジスタを表します。下位バイト[7~0](接尾辞L)は変位原点でアクセスできます。上位バイト[15~8](接尾辞H)は変位+1でアクセスすることができます。16ビットレジスタの読み書きのより多くの詳細については「16ビットレジスタのアクセス」をご覧ください。

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(MUX4~0)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 : ADCHとADCL

変位 : \$78

リセット : \$0000

特質 : ADLAR=1

ビット	15	14	13	12	11	10	9	8
	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	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15~6 – ADC9~0 : 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。

22.11.6. 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にそのビットは論理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
	ADC7D	ADC6D AMP2ND ACMPN1D	ADC5D ACMPN0D	ADC4D	ADC3D ACMPN2D	ADC2D ACMP2D	ADC1D	ADC0D ACMPN3D
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 – ADC7~0D : ADC7~0 デジタル入力禁止 (ADC7~0 Digital Input Disable)

● ビット6 – AMP2ND : AMP2N デジタル入力禁止 (AMP2N Digital Input Disable)

22.11.7. DIDR1 – デジタル入力禁止レジスタ1 (Digital Input Disable Register 1)

各々のビットは論理1を書かれると、対応するアナログピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されると、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。アナログ信号がアナログピンに印加され、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にそのビットは論理1を書かれるべきです。

名称 : DIDR1
 変位 : \$7F
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	-	AMP2PD	ACMP0D	AMP0PD	AMP0ND	ADC10D ACMP1D	ADC9D AMP1PD ACMP3D	ADC8D AMP1ND
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6,4,3,1,0 – AMPnxD : AMPnxD デジタル入力禁止 (AMPnxD Digital Input Disable)
- ビット2~0 – ADC10~8D : ADC10~8 デジタル入力禁止 (ADC10~8 Digital Input Disable)


22.11.8. AMPnCSR – 増幅器n制御/状態レジスタ (Amplifier n Control and Status register)

名称 : AMP0CSR : AMP1CSR : AMP2CSR
 変位 : \$75 : \$76 : \$77
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	AMPnEN	AMPnIS	AMPnG1,0		AMPCMPn	AMPnTS2~0		
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 – AMPnEN : 増幅器n許可 (Amplifier n Enable)

増幅器nを許可するにはこのビットを設定(1)してください。増幅器nを禁止するにはこのビットを解除(0)してください。変換走行中のこのビットの解除(0)は変換の最後で実施します。

 **警告:** AMPnEN解除(0)時、常にAMPnTS2~0を解除(=00)してください。

- ビット6 – AMPnIS : 増幅器n入力切替(短絡) (Amplifier n Input Shunt)

増幅器n入力(間)を短絡するにはこのビットを設定(1)してください。増幅器nを通常使用するにはこのビットを解除(0)してください。

- ビット5,4 – AMPnG1,0 : 増幅器n利得選択 (Amplifier n Gain Selection Bits)

これら2ビットは増幅器nの利得を決めます。正確な結果を保証するため、利得値が変更された後、最低4増幅器同期クロック周期の間、静定入力値を持つことが必要です。

値	説明
00	利得5 (×5)
01	利得10 (×10)
10	利得20 (×20)
11	利得40 (×40)

- ビット3 – AMPCMPn : 増幅器n 比較器n接続 (Amplifier n – Comparator n connection)

増幅器nを比較器nの非反転入力に接続するにはこのビットを設定(1)してください。この構成設定では比較器クロックが増幅器クロックに適用され、増幅器n起動元選択(AMPnTS2~0)ビットは無効です。増幅器nを標準使用するにはこのビットを解除(0)してください。

● ビット2~0 – AMPnTS2~0 : 増幅器n 起動元選択 (Amplifier n Trigger Source Selection Bits)

これらのビットは増幅器nに対してクロックを生成する事象を選びます。このクロック供給元は増幅チャネルでの変換を始めるのに必要です。

値	説明
000	A/D変換クロック/8で自動同期
001	タイマ/カウンタ0の比較A一致で同期化起動
010	タイマ/カウンタ0溢れで同期化起動
011	タイマ/カウンタ1の比較B一致で同期化起動
100	タイマ/カウンタ1溢れで同期化起動
101	電力制御器(PSC)単位部0同期信号(PSS0)で同期化起動
110	電力制御器(PSC)単位部1同期信号(PSS1)で同期化起動
111	電力制御器(PSC)単位部2同期信号(PSS2)で同期化起動

23. ISRC – 電流源

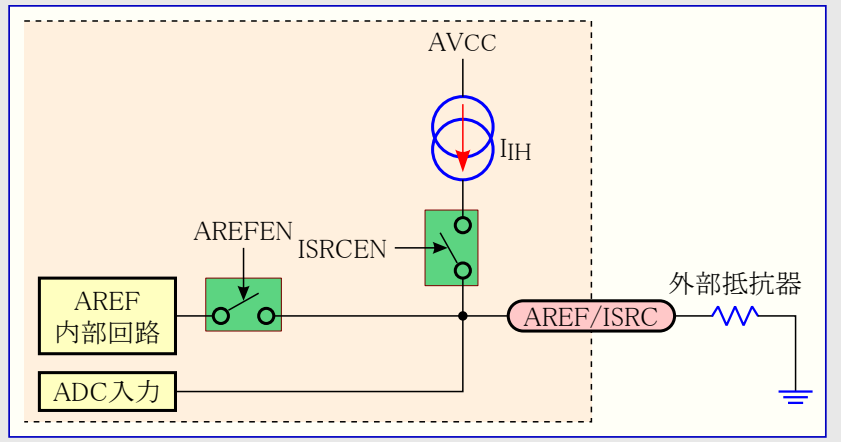
23.1. 特徴

- 100 μ A定電流源
- $\pm 2\%$ 絶対精度

Atmel ATmega16M1/32M1/64M1は100 μ A $\pm 2\%$ の電流源が特徴です。リセット後または要求で、電流は外部抵抗器を通して流れます。その電圧はA/D変換器と共用する専用ピンで測定することができます。誤差0.5%以下の直列抵抗器の使用が推奨されます。大きな値に対してデバイスを保護するために、A/D変換器(ADC)は最初の測定を実行するのに内部基準電圧としてAVCCで構成設定されなければなりません。その後、結果を改善するために直前の測定値に従って他の内部基準電圧を選ぶことができます。

A/D変換制御/状態レジスタ(ADCSR)の電流源許可(ISRCEN)ビットが設定(1)されると、ISRCピンは100 μ Aを吐き出します。さもなければ、このピンは初期の機能を維持します。

図23-1. 電流源構成図



23.2. 代表的な応用

23.2.1. LIN電流源

集合内のLIN節点(ノード)の構成設定中に、集合の全ての節点に対して動的に固有の物理アドレスにする必要があるかもしれません。それを行う方法はLIN規約に記述されていません。

電流源はLIN節点によって支援される応用に対して物理アドレスを関連付けする素晴らしい解決策を提供します。集合内のLIN節点構成設定に完全な動的節点構成設定を使うことができます。

Atmel ATmega16M1/32M1/64M1は電流源と共に使われる外部抵抗器を持つことを提唱します。デバイスはA/D変換器経由で抵抗の境界に対する電圧を測定します。結果の電圧は節点がLIN通信に参加する時に使う通信処理部の物理アドレスを定義します。

車載応用では分配された電圧が非常に乱されます。ATmega16M1/32M1/64M1の内部電流源解決策はどんな種類の電圧変動に対してもアドレス検出を免疫性にします。

表23-1. 87アドレス用($\pm 5\%$)抵抗値例 (AVCC=5V(注))

物理アドレス	抵抗値 (R _{load})	測定電圧 (V)		
		最小	代表	最大
0	1k Ω		0.1	
1	2.2k Ω		0.22	
2	3.3k Ω		0.33	
3	4.7k Ω		0.47	
4	6.8k Ω		0.68	
5	10k Ω		1	
6	15k Ω		1.5	
7	22k Ω		2.2	

注: 5V範囲での最大抵抗値=30k Ω
3V範囲での最大抵抗値=15k Ω

表23-2. 167アドレス用($\pm 1\%$)抵抗値例 (AVCC=5V(注))

物理アドレス	抵抗値 (R _{load})	測定電圧 (V)		
		最小	代表	最大
0	2k Ω		0.2	
1	2.4k Ω		0.24	
2	2.7k Ω		0.27	
3	3.3k Ω		0.33	
4	3.9k Ω		0.39	
5	4.7k Ω		0.47	
6	5.6k Ω		0.56	
7	6.8k Ω		0.68	
8	8.2k Ω		0.82	
9	9.1k Ω		0.91	
10	11k Ω		1.1	
11	13k Ω		1.3	
12	16k Ω		1.6	
13	18k Ω		1.8	
14	20k Ω		2	
15	24k Ω		2.4	

注: 5V範囲での最大抵抗値=30k Ω
3V範囲での最大抵抗値=15k Ω

23.2.2. 外部デバイス用電圧基準

電流源と共に使われる外部抵抗器は外部デバイス用の電圧基準として使うことができます。電流源精度(2%)よりも低い誤差の直列抵抗器の使用が推奨されます。前表は標準抵抗値を用いた電圧基準の例を与えます。

23.2.3. 内部アナログ比較器用閾値基準

電流源と共に使われる外部抵抗器は内部アナログ比較器(ご覧ください。)用の電圧基準として使うことができます。これはAIN1(アナログ比較器非反転入力ピン)だけでなく、AIN0(アナログ比較器反転入力ピン)にも接続することができます。電流源精度(2%)よりも低い誤差の直列抵抗器の使用が推奨されます。前表は標準抵抗値を用いた電圧基準の例を与えます。

23.3. 電流源用レジスタ

23.3.1. 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
	ADHSM	ISRCEN	AREFEN	-	ADTS3~0			
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6 – ISRCEN : 電流源許可 (Current Source Enable)

AREFピンに100 μ Aの電流を吐き出すにはこのビットを設定(1)してください。AREFピンをアナログ基準電圧ピンとして使うにはこのビットを解除(0)してください。

- ビット5 – AREFEN : アナログ基準電圧ピン許可 (Analog Reference pin Enable)

AREFピンに内部AREF回路を接続するにはこのビットを設定(1)してください。AREFピンから内部AREF回路を切断するにはこのビットを解除(0)してください。

24. AC – アナログ比較器 (Analog Comparator)

24.1. 特徴

- 4つのアナログ比較器
- 高速クロック駆動の比較器
- ±30mVのヒステリシス
- 4つの基準レベル
- 構成設定可能な割り込み生成

24.2. 概要

Atmel ATmega16M1/32M1/64M1は4つの高速アナログ比較器が特徴です。アナログ比較器はACMPn非反転ピンとACMPMまたはACMPn反転ピンの入力値を比較します。

各比較器には非反転入力専用入力があり、各比較器の反転入力は次のように設定できます。

- A/D変換多重器選択(ADMUX)レジスタの基準電圧選択(REFS1,0)ビットで選んだVREFによって定義された4つの内部基準電圧内の固定値
- 内蔵D/A変換器で生成した値
- 外部ACMPMnアナログ入力

ACMPn非反転ピンの電圧がアナログ比較器n制御(ACnCON)レジスタの反転入力選択(ACnM2~0)ビットによって選択した反転入力電圧よりも高い場合にアナログ比較器n出力(ACnO)が設定(1)されます。

この比較器はクロック駆動比較器です。新規比較は(アナログ比較器0制御レジスタ(AC0CON)のアナログ比較器クロック選択(ACCKSEL)ビットに依存して)CLKI/OまたはCLKI/O/2の下降端で行なわれます。「AC0CON – アナログ比較器0制御レジスタ」をご覧ください。

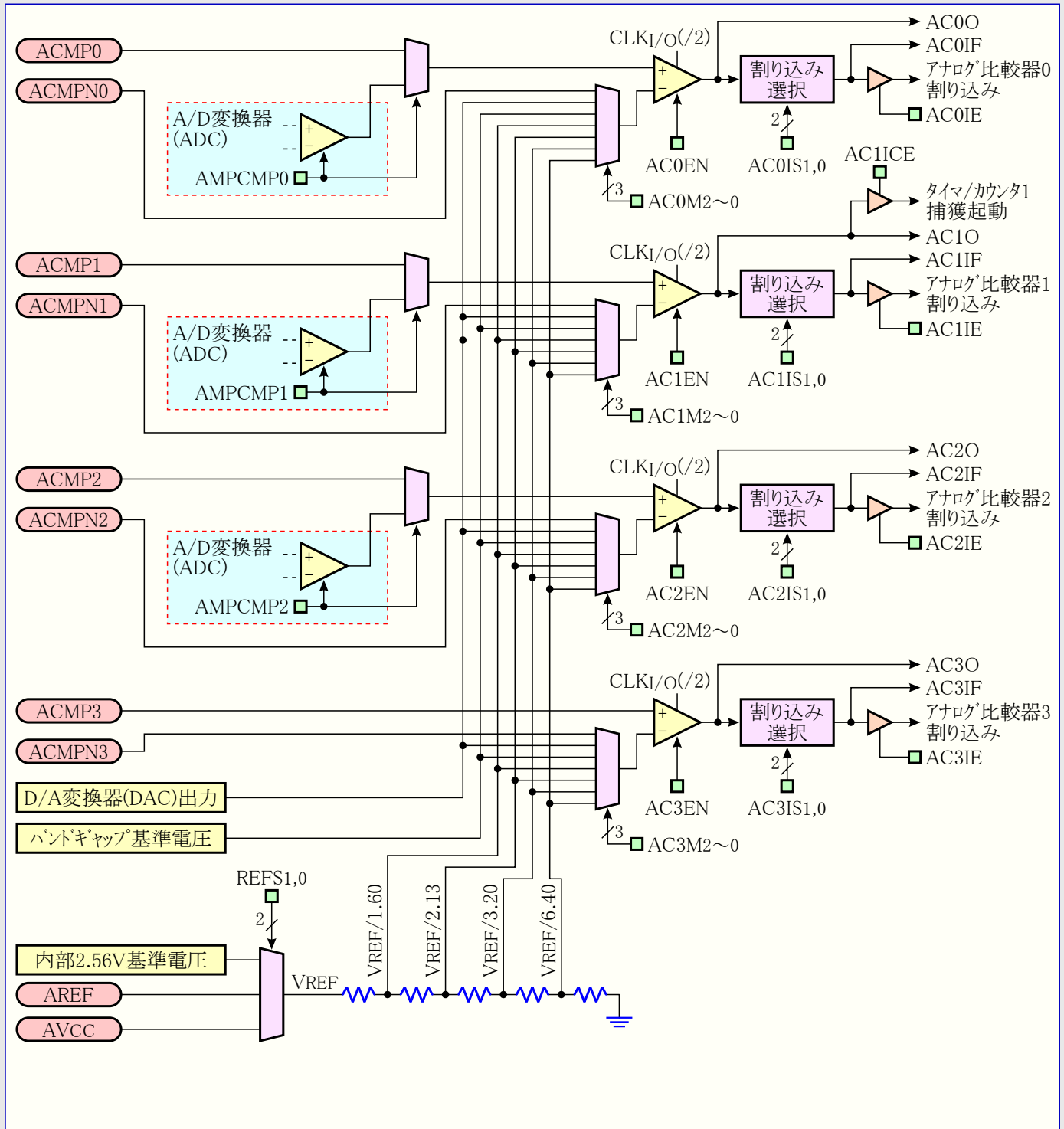
各比較器はアナログ比較器専用の独立した割り込みを起動できます。加えて、使用者は比較器出力の上昇端、下降端、切り替わり(両端)での割り込み起動を選べます。

この割り込み要求フラグはA/D変換器やD/A変換器の同期に使うこともできます。

更に比較器1の比較機出力はタイマ/カウンタ1の捕獲機能の起動に設定できます。

4つの比較器とそれらの周辺回路の構成図は以下で示されます。

図24-1. アナログ比較器構成図



- 注:
- A/D変換器(ADC)多重器出力、「[ADMUX - A/D多重器選択レジスタ](#)」をご覧ください。
 - アナログ比較器ピン配置については「[ピン配置](#)」を参照してください。
 - VREF電圧はA/D多重器選択レジスタ(ADMUX)で定義されます。

関連リンク [183頁の「ADMUX - A/D多重器選択レジスタ」](#)
[5頁の「ピン配置」](#)
[6頁の「ピン概要」](#)

24.3. ADC増幅器の使い方

増幅器0比較器0接続(AMPCMP0)構成設定ビットにより、比較器0の非反転入力が増幅器0出力に接続することができます。その場合は比較器0のクロックが増幅器0のクロックに適用されます。

増幅器1比較器1接続(AMPCMP1)構成設定ビットにより、比較器1の非反転入力が増幅器1出力に接続することができます。その場合は比較器1のクロックが増幅器1のクロックに適用されます。

増幅器2比較器2接続(AMPCMP2)構成設定ビットにより、比較器2の非反転入力が増幅器2出力に接続することができます。その場合は比較器2のクロックが増幅器2のクロックに適用されます。

関連リンク [187頁の「AMPnCSR - 増幅器n制御/状態レジスタ」](#)

24.4. アナログ比較器用レジスタ

24.4.1. AC0CON – アナログ比較器0制御レジスタ (Analog Comparator 0 Control Register)

名称 : AC0CON
 変位 : \$94
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	AC0EN	AC0IE	AC0IS1,0		AC0CKSEL	AC0M2~0		
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

24.4.2. AC1CON – アナログ比較器1制御レジスタ (Analog Comparator 1 Control Register)

名称 : AC1CON
 変位 : \$95
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	AC1EN	AC1IE	AC1IS1,0		AC1ICE	AC1M2~0		
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

24.4.3. AC2CON – アナログ比較器2制御レジスタ (Analog Comparator 2 Control Register)

名称 : AC2CON
 変位 : \$96
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	AC2EN	AC2IE	AC2IS1,0		-	AC2M2~0		
アクセス種別	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

24.4.4. AC3CON – アナログ比較器3制御レジスタ (Analog Comparator 3 Control Register)

名称 : AC3CON
 変位 : \$97
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	AC3EN	AC3IE	AC3IS1,0		-	AC3M2~0		
アクセス種別	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7 – ACnEN : アナログ比較器n 動作許可 (Analog Comparator n Enable)

値	説明
0	アナログ比較器nを禁止
1	アナログ比較器nを許可

● ビット6 – ACnIE : アナログ比較器n 割り込み許可 (Analog Comparator n Interrupt Enable)

値	説明
0	アナログ比較器n割り込みを禁止
1	アナログ比較器n割り込みを許可

(訳注) 原書のAC0CON, AC1CON, AC2CON, AC3CONの各ビット説明はACn~として纏めました。

● ビット5,4 – ACnIS1,0 : アナログ比較器n 割り込み条件選択 (Analog Comparator n Interrupt Select bits)

これら2ビットは割り込み起動の感知を決めます。

値	説明
00	比較器n出力の変移 (トグル)
01	(予約)
10	比較器n出力の下降端
11	比較器n出力の上昇端

● ビット3 – ACCKSEL : アナログ比較器クロック選択 (Analog Comparator Clock Select) (アナログ比較器0のみ)

値	説明
0	比較器クロックとしてclk _{I/O} を使用
1	比較器クロックとしてPLL出力を使用

● ビット3 – AC1ICE : アナログ比較器1捕獲起動許可 (Analog Comparator 1 Input Capture Enable) (アナログ比較器1のみ)

アナログ比較器1の事象でタイマ/カウンタ1の捕獲機能を許可するにはこのビットを設定(1)してください。この場合、比較器1出力は比較器にタイマ/カウンタ1捕獲割り込みの雑音消去機能と端(エッジ)選択機能を利用させる捕獲入力前置論理回路へ直接的に接続されます。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。

タイマ/カウンタ1制御レジスタB(TCCR1B)のICES1ビットが1に設定の場合に考慮されるのはAC10の上昇端がタイマ/カウンタ1の捕獲起動事象で、ICES1が0に設定の場合にはそれが下降端です。

この機能を禁止するにはこのビットを解除(0)してください。この場合、アナログ比較器1と捕獲機能間の接続は存在しません。

● ビット2~0 – ACnM2~0 : アナログ比較器n 反転入力選択 (Analog Comparator n Multiplexer register)

これら2ビットはアナログ比較器nの反転入力の入力を決めます。

値	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
説明	VREF/6.40	VREF/3.20	VREF/2.13	VREF/1.60	バンドキャップ 基準電圧 (1.1V)	D/A変換器 出力	アナログ比較器 反転入力 (ACMPMピン)	(予約)

24.4.5. ACSR – アナログ比較器状態レジスタ (Analog Comparator Status Register)

名称 : ACSR

変位 : \$50 (\$30)

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\$30です。

ビット	7	6	5	4	3	2	1	0
	AC3IF	AC2IF	AC1IF	AC0IF	AC3O	AC2O	AC1O	AC0O
アクセス種別	R/W	R/W	R/W	R/W	R	R	R	R
リセット値	0	0	0	0	不定	不定	不定	不定

● ビット7 – AC3IF : アナログ比較器3 割り込み要求フラグ (Analog Comparator 3 Interrupt Flag)

このビットはアナログ比較器3出力がアナログ比較器3制御レジスタ(AC3CON)の割り込み条件選択(AC3IS1,0)ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC3CONレジスタの割り込み許可(AC3IE)ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期にも使えます。

● ビット6 – AC2IF : アナログ比較器2 割り込み要求フラグ (Analog Comparator 2 Interrupt Flag)

このビットはアナログ比較器2出力がアナログ比較器2制御レジスタ(AC2CON)の割り込み条件選択(AC2IS1,0)ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC2CONレジスタの割り込み許可(AC2IE)ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期にも使えます。

● ビット5 – AC1IF : アナログ比較器1 割り込み要求フラグ (Analog Comparator 1 Interrupt Flag)

このビットはアナログ比較器1出力がアナログ比較器1制御レジスタ(AC1CON)の割り込み条件選択(AC1IS1,0)ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC1CONレジスタの割り込み許可(AC1IE)ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期にも使えます。

● ビット4 – AC0IF : アナログ比較器0 割り込み要求フラグ (Analog Comparator 0 Interrupt Flag)

このビットはアナログ比較器0出力がアナログ比較器0制御レジスタ(AC0CON)の割り込み条件選択(AC0IS1,0)ビットによって定義した割り込み動作種別で起動する時にハードウェアによって設定(1)されます。

このビットはAC0CONレジスタの割り込み許可(AC0IE)ビットが設定(1)なら、対応する割り込みベクタが実行される時に解除(0)されます。何れにしても、このビットは論理1書き込みによって解除(0)されます。

このビットはA/D変換器またはD/A変換器の同期にも使えます。

● ビット3 – AC3O : アナログ比較器3 出力 (Analog Comparator 3 Output)

値	説明
0	比較器出力がLow(0)
1	比較器出力がHigh(1)

● ビット2 – AC2O : アナログ比較器2 出力 (Analog Comparator 2 Output)

値	説明
0	比較器出力がLow(0)
1	比較器出力がHigh(1)

● ビット1 – AC1O : アナログ比較器1 出力 (Analog Comparator 1 Output)

値	説明
0	比較器出力がLow(0)
1	比較器出力がHigh(1)

● ビット0 – AC0O : アナログ比較器0 出力 (Analog Comparator 0 Output)

値	説明
0	比較器出力がLow(0)
1	比較器出力がHigh(1)

24.4.6. DIDR0 – デジタル入力禁止レジスタ0 (Digital Input Disable Register 0)

名称 : DIDR0
 変位 : \$7E
 リセット : \$00
 特質 : -

ビット	7	6	5	4	3	2	1	0
	ADC7D	ADC6D AMP2ND ACMPN1D	ADC5D ACMPN0D	ADC4D	ADC3D ACMPN2D	ADC2D ACMP2D	ADC1D	ADC0D ACMPN3D
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6,5,3,2,0 – ACMPN1D,ACMPN0D,ACMPN2D,ACMP2D,ACMPN3D
 : ACMPN1,ACMPN0,ACMPN2,ACMP2,ACMPN3 デジタル入力禁止
 (ACMPN1,ACMPN0,ACMPN2,ACMP2,ACMPN3 Digital Input Disable)

このビットが論理1を書かれると、対応するアナログピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。これらのピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされないとき、このビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

24.4.7. 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
	-	AMP2PD	ACMP0D	AMP0PD	AMP0ND	ADC10D ACMP1D	ADC9D AMP1PD ACMP3D	ADC8D AMP1ND
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5,2,1 – ACMP0D,ACMP1D,ACMP3D : ACMP0,ACMP1,ACMP3 デジタル入力禁止
(ACMP0,ACMP1,ACMP3 Digital Input Disable)

このビットが論理1を書かれると、対応するアナログピンのデジタル入力緩衝部が禁止されます。このビットが設定(1)されるとき、対応するポート入力(PINx)レジスタのビットは常に0として読みます。これらのピンにアナログ信号が印加され、そのピンからのデジタル入力が必要とされないとき、このビットはデジタル入力緩衝部での消費電力を削減するために論理1を書かれるべきです。

25. DAC – D/A変換器

25.1. 特徴

- 10ビット分解能
- 8ビット直線精度
- 100mV～(AVCC-100mV)間±0.5 LSB精度
- 出力電圧(VOUT)=D/A変換器値×基準電圧(VREF)/1023
- アナログ比較器反転入力と/または専用出力駆動部に接続可能
- 約100Ωの出力インピーダンス (故に時定数約1μsに於いて33kΩを越える抵抗で1000pFの並列容量負荷にできます。)

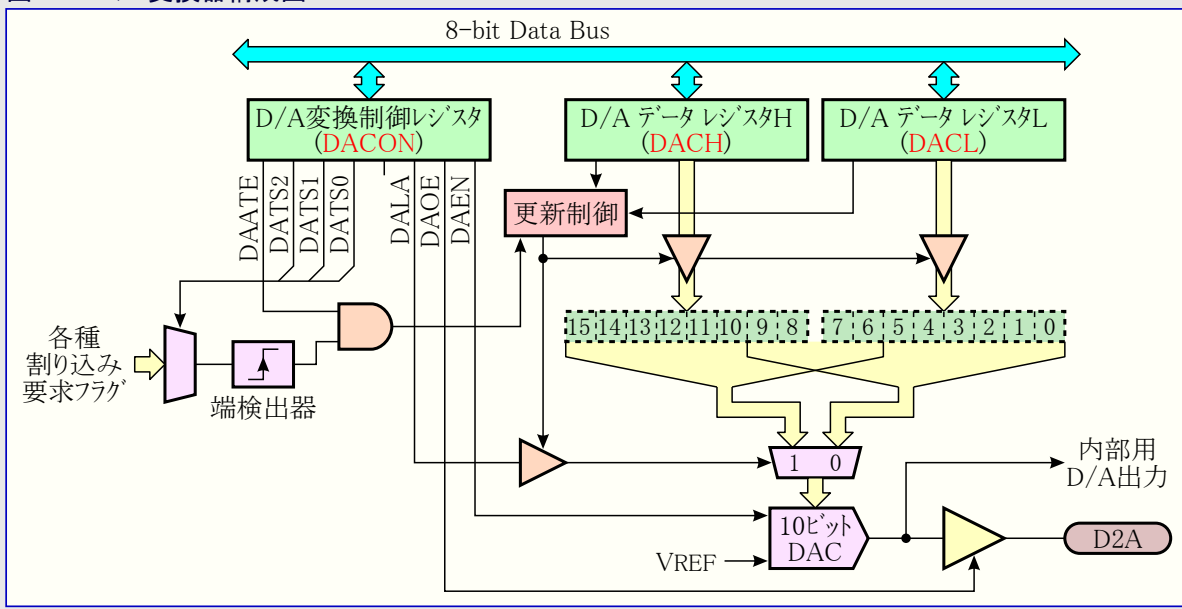
25.2. 概要

Atmel ATmega16M1/32M1/64M1は10ビットD/A変換器が特徴です。このD/A変換器はアナログ比較器に対して使え、且つ/または専用駆動部経由でマイクロコンピュータのD2Aピンに出力できます。

D/A変換器には分離されたアナログ電源供給ピン(AVCC)があります。AVCCはVCCから±0.3Vよりも多く違ってはなりません。このピンの接続方法はA/D変換記述をご覧ください。

基準電圧はA/D変換器で使われるのと同じです。A/D多重器選択レジスタ(ADMUX)をご覧ください。公称2.56Vの内部基準電圧またはAVCCがチップ上で提供されます。この基準電圧は雑音特性向上のため、コンデンサによってAREFピンで外部的にデカップ(雑音分離)できます。

図25-1. D/A変換器構成図



関連リンク [178頁の「アナログ雑音低減技術」](#)
[183頁の「ADMUX – A/D多重器選択レジスタ」](#)

25.3. 操作

D/A変換器はD/Aデータレジスタ値に比例したアナログ信号を生成します。

正確な採取周波数制御を持っているので、各種起動事象を通してD/A変換器入力値を更新することが可能です。

25.4. 変換の開始

D/A変換器はD/A制御(DACON)レジスタによって構成設定されます。DACONレジスタのD/A許可(DAEN)ビットが設定(1)されると直ぐにD/A変換器はDACONレジスタ設定に従ってD/Aデータ(DACH,DACL)レジスタに存在する値を変換します。

代わりに変換器は様々な供給元によって自動的に起動できます。自動起動はDACONのD/A変換自動起動許可(DAATE)ビットの設定(1)によって許可されます。起動元はDACONのD/A変換起動元選択(DATS2~0)ビット設定によって選ばれます(起動元一覧についてはDATSビット記述をご覧ください)。選んだ起動信号で上昇端が起こると、D/A変換器はDACONレジスタ設定に従い、DACHとDACLレジスタに存在する値を変換します。これは固定間隔での変換開始方法を提供します。変換完了時に未だ起動信号が設定(1)されていると、新しい変換は開始されません。変換中に別の起動信号で上昇端が起こる場合、その端(エッジ)は無効です。特定の割り込みが禁止、またはステータスレジスタ(SREG)の全割り込み許可(I)ビットが解除(0)されていても、割り込み要求フラグが設定(1)されることに注意してください。従って変換は割り込みなしで起動できます。とは言え、割り込み要求フラグは次の割り込み事象で新しい(次の)変換を起動するために解除(0)されなければなりません。

25.4.1. D/A変換器基準電圧

A/D変換器用の基準電圧(V_{REF})がD/A変換器に対する変換範囲を示します。 V_{REF} はAVCC、内部2.56V基準電圧、外部AREFピンのどれかとして選べます。

AVCCは受動型スイッチを通してD/A変換部に接続されます。内部2.56V基準電圧は内蔵基準(バンドギャップ)電圧(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D/A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いけません。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2.56V基準電圧間の切り替えができます。基準電圧源切り替え後の最初のD/A変換結果は不正確かもしれず、使用者はこの結果を破棄することが推奨されます。

25.5. D/A変換器用レジスタ

25.5.1. DACON – D/A変換制御レジスタ (Digital to Analog Conversion Control Register)

名称 : DACON

変位 : \$90

リセット : \$00

特質 : -

ビット	7	6	5	4	3	2	1	0
	DAATE	DATS2~0			-	DALA	DAOE	DAEN
アクセス種別	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット7 – DAATE : D/A変換自動起動許可 (DAC Auto Trigger Enable)

DACONレジスタのDATS2~0ビットで選んだ起動信号の上昇端でD/A変換器入力値を更新するにはこのビットを設定(1)してください。値がDACHレジスタへ書かれる時にD/A変換器入力を自動的に更新するにはこれを解除(0)してください。

● ビット6~4 – DATS2~0 : D/A変換自動起動要因選択 (DAC Trigger Selection Bits)

これらのビットはD/A変換器が自動起動動作で作動する場合だけ必要です。これはDAATEビットが設定(1)される場合の意味です。これら3ビットはD/A変換器入力値の更新を生成する割り込み事象を選びます。更新は割り込みが許可であろうとなかろうと、選んだ割り込み要求フラグの上昇端によって生成されます。

値	起動元	値	起動元
0 0 0	アナログ比較器0	1 0 0	タイマ/カウンタ0溢れ
0 0 1	アナログ比較器1	1 0 1	タイマ/カウンタ1比較B一致
0 1 0	外部割り込み要求0	1 1 0	タイマ/カウンタ1溢れ
0 1 1	タイマ/カウンタ0比較A一致	1 1 1	タイマ/カウンタ1捕獲要求

● ビット2 – DALA : D/A変換 左揃え選択 (Digital to Analog Left Adjust)

D/A変換器入力データを左揃えにするにはこのビットを設定(1)してください。D/A変換器入力データを右揃えにするにはこのビットを解除(0)してください。DALAビットはD/A変換データレジスタの形態に影響を及ぼします。このビットの変更は次(から)のDACH書き込みでのD/A変換器出力に影響を及ぼします。

● ビット1 – DAOE : D/A変換 出力許可 (Digital to Analog Output Enable)

D2Aピンに変換結果を出力するにはこのビットを設定(1)してください。D/A変換器を内部的に使うにはこれを解除(0)してください。

● ビット0 – DAEN : D/A変換 許可 (Digital to Analog Enable)

D/A変換器を許可するにはこのビットを設定(1)してください。D/A変換器を禁止するにはこれを解除(0)してください。

25.5.2. DACH,DA CL – D/Aデータレジスタ (Digital to Analog Converter Input Register) [DALA=0]

DACHとDA CLレジスタはアナログ電圧に変換されるべき値を含みます。DA CLレジスタ書き込みはDACHも書かれてしまうまで入力値の更新を禁止します。従ってD/A変換レジスタに10ビット値を書く一般的な方法はまず第1にDA CL、次にDACHを書くことです。8ビット精度だけで容易に作業するために入力値を左揃えにすることが可能です。このようにしてD/A変換値を更新するにはDACHを書くことで充分です。

10ビットD/A変換器で動作するには2つのレジスタが更新されなければなりません。途中の値を避けるため、アナログ信号に変換される実際のD/A変換入力値は届くことができないレジスタ内に緩衝されます。標準動作でのこの隠れたレジスタの更新はDACHレジスタが書かれる時に行なわれます。

D/A変換自動起動許可(DAATE)ビットが設定(1)される場合、D/A変換入力値は起動元選択(DATS2~0)ビットを通して選んだ起動事象で更新されます。

不正なD/A変換入力値を避けるため、更新はDA CLとDACHレジスタの各々を書いてしまった後でだけ行なえます。DACH値書き込みだけによる8ビット形態で動作することが可能です。この場合(訳補:結果的に)、更新は各起動事象で行なわれます。

DAATEビットが解除(0)される場合、このD/A変換器は自動更新動作です。DACHレジスタ書き込みがDACHとDA CLレジスタ値でD/A変換入力値を自動的に更新します。

これはDAATEビット設定がどうであれ、DACHレジスタも更新されてしまうまで、DA CLレジスタの変更がD/A変換器出力に対して無効であることを意味します。従って10ビットで動作するにはDACHに先立ってDA CLが最初に書かれなければなりません。8ビット形態での動作ではDACH書き込みがD/A変換器の更新を許します。

名称 : DA CL : DACH
 変位 : \$91 : \$92
 リセット : \$0000
 特質 : DALA=0

ビット	15	14	13	12	11	10	9	0
	-						DAC9,8	
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	6	5	4	3	2	1	0
	DAC7~0							
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9~0 – DAC9~0 : D/A変換値 (DAC Value)

25.5.3. DACH,DA CL – D/Aデータレジスタ (Digital to Analog Converter Input Register) [DALA=1]

名称 : DA CL : DACH
 変位 : \$91 : \$92
 リセット : \$0000
 特質 : DALA=1

ビット	15	14	13	12	11	10	9	8
	DAC9~2							
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	6	5	4	3	2	1	0
	DAC1,0		-	-	-	-	-	-
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15~6 – DAC9~0 : D/A変換値 (DAC Value)

詳細についてはDALA=0記述をご覧ください。

26. DBG – デバッグWIRE 内蔵デバッグ システム (debugWIRE On-chip Debug System)

26.1. 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

26.2. 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。

26.3. 物理インターフェース

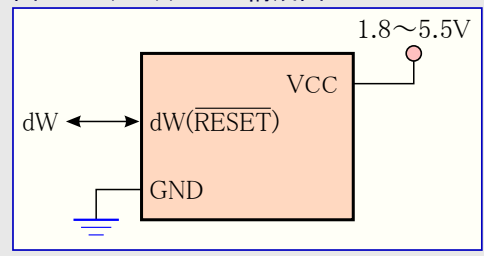
デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、**施錠ビット**が非プログラム(1)にされると、対象デバイス内のデバッグWIREシステムが活性(有効)にされます。RESETポートピンはプルアップ許可のANDタイ(オープンドレイン)双方向I/Oピンとして設定され、対象デバイスとエミュレータ間の通信路になります。

デバッグWIRE構成(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26-1. デバッグWIRE構成図



26.4. 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。

26.5. 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。PRSPI0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。

26.6. 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。

26.6.1. 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。

27. 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

27.1. 特徴

- 書き込み中読める(Read-While-Write)自己プログラミング
- 柔軟性のあるブートローダメモリ容量
- 高い安全性(柔軟な保護用の独立したブート施錠ビット)
- リセットベクタ選択用の独立したヒューズ
- 最適化されたページ容量(注1)
- 効率的なコード手法
- 効率的な読み-変更-書き(リード モデファイ ライト)支援

注1: ページはプログラミング中に使われる多数のバイトから成るフラッシュメモリの区画です(「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 217頁の「ページ容量」

27.2. 概要

本デバイスに於いて、ブートローダ支援はMCU自身によるプログラムコードのダウンロードとアップロード用の真の書き込み中の読み出しが可能な自己プログラミング機構を提供します。この特徴はフラッシュメモリに常駐するブートローダプログラムを使うMCUによって制御される柔軟な応用ソフトウェア更新を可能にします。ブートローダプログラムはフラッシュメモリ内にコードを書き(プログラム)、コードを読み、またはプログラムメモリからコードを読むのに、利用可能なデータインターフェースと関連する規約のどれもが使えます。ブートローダ領域内のプログラムコードはブートローダメモリを含むフラッシュメモリ全体を書く能力を持ちます。従ってブートローダは自身をも変更でき、この機能がそれ以上必要とされないなら、そのコードから自身を消去することもできます。ブートローダメモリの容量はヒューズで設定可能で、ブートローダは個別に設定可能な2組の独立したブート施錠ビットを持ちます。これは異なる保護基準を選ぶ独特な柔軟性を使用者に与えます。

27.3. フラッシュメモリの応用領域とブートローダ領域

フラッシュメモリは応用領域とブートローダ領域の2つの主な領域で構成されます。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。

27.3.1. 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、**Mega32A** 応用領域はどんなブートローダコードも決して格納し得ません。

27.3.2. ブートローダ領域 (BLS)

応用領域が応用コード格納用に使われるのに対して、SPM命令はBLSから実行する時にだけプログラミングを始められるので、ブートローダソフトウェアはBLSに格納されなければなりません。SPM命令はBLS自身を含む全てのフラッシュメモリをアクセスできます。ブートローダ領域用保護基準はブートローダ施錠ビット(ブート施錠ビット1)によって選べます。

27.4. 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、ブートローダソフトウェアが更新中にCPUが停止されるかのどちらです。上で記述されるようなBOOTSZヒューズによって設定可能な2つの領域に加え、フラッシュメモリは書き込み中読み出し可能な(RWW)領域と書き込み中読み出し不能な(NRWW)領域の2つの固定領域にも分けられます。RWWとNRWW領域間の境界は「ブートローダパラメータ」項と図27-2.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。

ブートローダ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、ブートローダソフトウェアが更新中に実際に読まれる領域ではありません。

関連リンク 212頁の「ブートローダパラメータ」

(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。

27.4.1. 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。

27.4.2. 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27-1. 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27-1. RWW領域とNRWW領域の関係

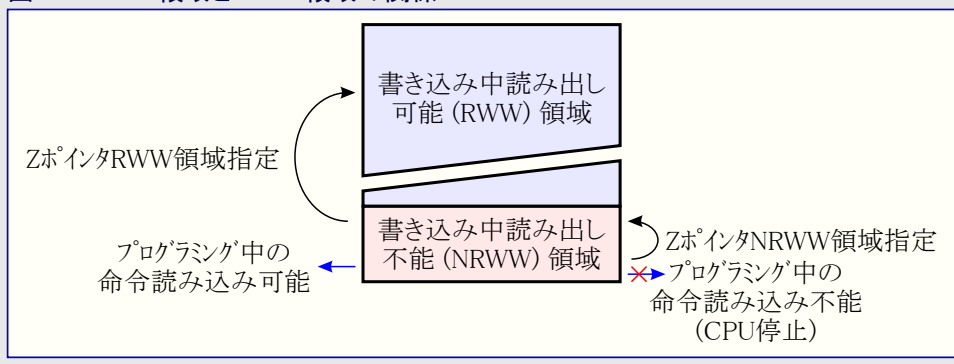
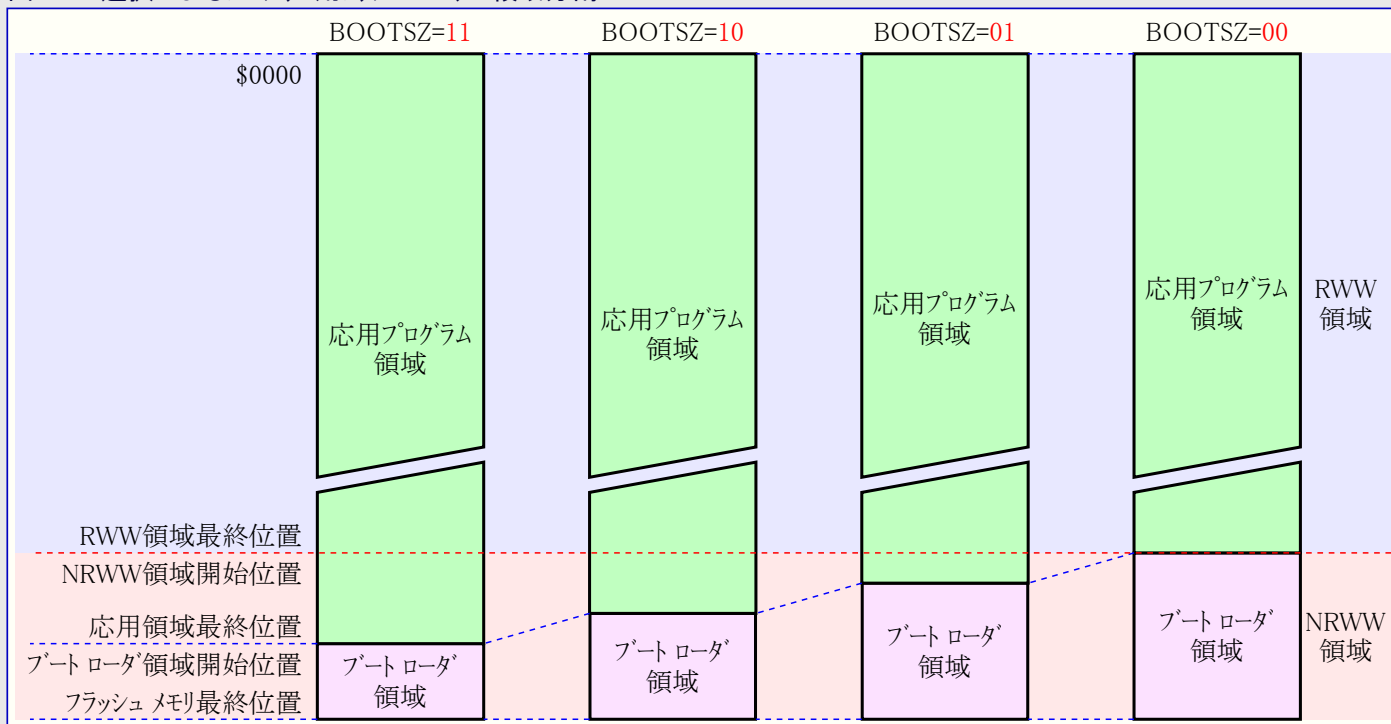


図27-2. 選択によるプログラム用フラッシュメモリの領域分割



関連リンク [212頁の「ブートローダパラメータ」](#)

27.5. 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27-2. 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27-3. 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。

27.6. 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27-4. 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

27.7. 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

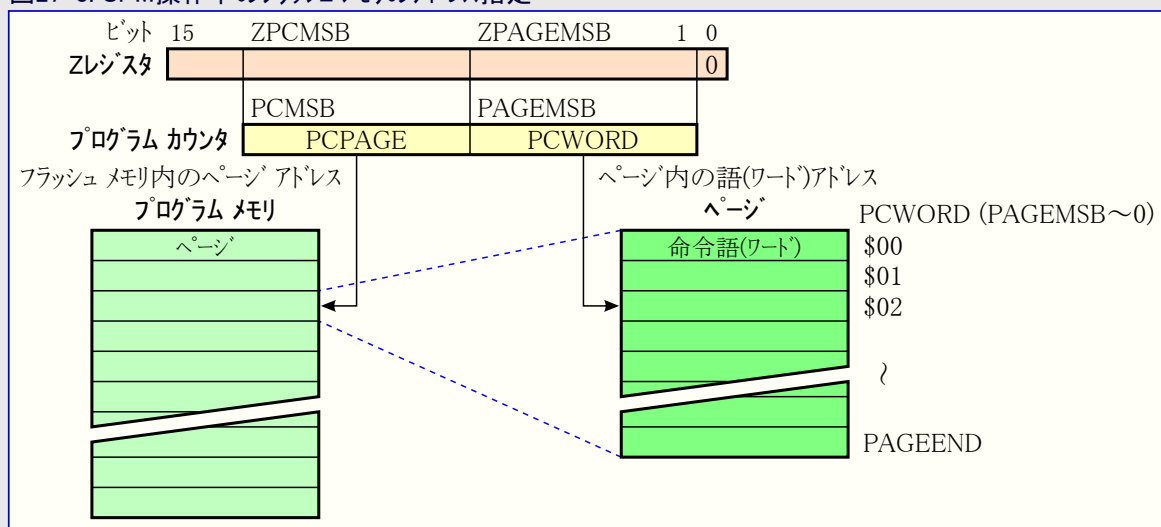
Zポインタ(レジスタ)はSPM命令でのアドレス指定に使われます。Zポインタはレジスタファイル内のZHとZLのZレジスタから成ります。実際に使われるビット数は実装依存です。

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ポインタは他の操作に使えます。

Zポインタを使わないSPM操作はブートローダ施錠ビット設定だけです。この操作でZレジスタの内容は無視され、無効です。LPM命令もアドレスを格納するのにZポインタを使います。この命令はフラッシュメモリのバイト単位をアドレス指定するので、Zポインタの最下位ビット(Z0)も使われます。

図27-3. 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27-9.で一覧されます。

27.8. 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、その後改めて書かれます。手段1.を使う場合、初めにページを読んで必要な変更を行い、その後に変更したデータを書き戻すことをユーザーソフトウェアに許す効率的な読み-修正-書き(リード-モディファイ-ライト)機能をブートローダが提供します。手段2.が使われる場合、ページが既に消去されているため、格納中の旧データを読むことができません。ページ一時緩衝部は乱順でアクセスできます。ページ消去とページ書き込み操作の両方で使われるページアドレスは同じページをアドレス指定することが非常に重要です。「アセンブリ言語による簡単なブートローダ例」を参照してください。

27.8.1. 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。

27.8.2. 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。

27.8.3. 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。

27.8.4. SPM操作可割り込みの使用法

SPM操作可割り込みが許可されると、SPMCSRのSPMENビット(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 41頁の「INT - 割り込み」

27.8.5. 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、特別な注意が祓われなければなりません。ブートローダ自身への予期せぬ書き込みはブートローダ全体を不正にし得て、更にソフトウェアの更新が不可能になるかもしれません。ブートローダ自体の変更が必要ないなら、内部ソフトウェアのどんな変更からもブートローダを保護するためにブート施錠ビット11(BLB11)をプログラム(0)することが推奨されます。

27.8.6. 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 41頁の「INT - 割り込み」

27.8.7. 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。

27.8.8. 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。

27.8.9. 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	EFB7	EFB6	EFB5	EFB4	-	EFB2	EFB1	EFB0

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

関連リンク [216頁の「ヒューズビット」](#)

27.8.10. 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27-5. 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。

27.8.11. 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を使う基板段階の装置と同じで、同じ設計上の解決策が適用されるべきです。

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

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

- そのシステムでファームウェア更新が必要ない場合、どんなファームウェア更新をも防ぐためにファームウェア施錠ビットをプログラム(0)してください。
- 不十分な供給電源電圧の期間中、AVR RESETを活性(Low)に保ってください。これは動作電圧が検出電圧と一致するなら、内部低電圧検出器(BOD)を許可することによって行えます。そうでなければ外部低VCCリセット保護回路が使えます。書き込み操作進行中にリセットが起こると、その書き込み動作は供給電源電圧が充分であれば完了されます。
- 低VCCの期間中、AVRコアをパワーダウン休止動作に保ってください。これはCPUが命令の復号と実行の試みを防ぎ、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。

27.8.12. 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27-6. 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。

27.8.13. アセンブリ言語による簡単なブートローダ例

このルーチンはRAMからフラッシュメモリへ1ページのデータを書きます。RAM内の最初のデータ位置はYレジスタによって指示され、フラッシュメモリ内の最初のデータ位置はZレジスタによって指示されます。異常処理は含まれません。このルーチン(少なくともSPMJサブルーチン)はブートローダ領域側に配置されなければなりません。NRWW領域側のコードだけが自己プログラミング(ページ消去とページ書き込み)中に読めます。使用レジスタはR0,R1,TMP,CNTL,CNTH,SPMCで、レジスタの保存と復帰はこのルーチン内に含まれず、使用レジスタはコード量を犠牲にすれば最適化できます。割り込み表がブートローダ領域に移動されるか、割り込みが禁止されるかのどちらかが前提です。

ページ内データが256バイト以下の場合には計数器上位が不要になります。また関連する命令も変更になります。これらの部分を赤字で示します(訳注:本行は以下のプログラム補正に対応して追加しました)。

ラベル	命令		注釈
	.EQU	PGSZB = PAGESIZE*2	; PGSZBはページ内のバイト数です。(PAGESIZEは語(ワード)数)
	.ORG	SMALLBOOTSTART	;
WRPG:	LDI	SPMC, (1<<PGERS)+(1<<SPMEN)	; [ページ消去]
	RCALL	SPMJ	; ページ消去SPMCSR値を取得
			; ページ消去
			; [RWW領域読み出し再許可]
	LDI	SPMC, (1<<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]
	LDI	CNTL, LOW(PGSZB)	; バイト計数器を初期化
	LDI	CNTH, HIGH(PGSZB)	; (削除)
WLP:	LD	R0, Y+	; RAM上の下位データを取得(ポインタ進行)
	LD	R1, Y+	; RAM上の上位データを取得(ポインタ進行)
	LDI	SPMC, (1<<SPMEN)	; ページ一時緩衝部書き込みSPMCSR値を取得
	RCALL	SPMJ	; 対応語(ワード)データをページ一時緩衝部に設定
	ADIW	ZH: ZL, 2	; ページ一時緩衝部ポインタ進行
	SBIW	CNTH: CNTL, 2	; 計数器を減数 (SUBI)
	BRNE	WLP	; 指定バイト数分継続
			; [ページ書き込み]
	SUBI	ZL, LOW(PGSZB)	; ページ一時緩衝部先頭にポインタを復帰
	SBCI	ZH, HIGH(PGSZB)	; (削除)
	LDI	SPMC, (1<<PGWRT)+(1<<SPMEN)	; フラッシュ書き込みSPMCSR値を取得
	RCALL	SPMJ	; フラッシュメモリページ書き込み
			; [RWW領域読み出し再許可]
	LDI	SPMC, (1<<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
			; [読み戻し照合(任意)]
	LDI	CNTL, LOW(PGSZB)	; バイト計数器を初期化
	LDI	CNTH, HIGH(PGSZB)	; (削除)
	SUBI	YL, LOW(PGSZB)	; RAMデータ先頭にポインタを復帰
	SBCI	YH, HIGH(PGSZB)	;
RLP:	LPM	R0, Z+	; フラッシュメモリから1バイト取得(ポインタ進行)
	LD	R1, Y+	; RAMから1バイトデータを取得(ポインタ進行)
	CPSE	R0, R1	; 値一致でスキップ
	RJMP	ERROR	; 不一致で異常処理へ
;			
	SBIW	CNTH: CNTL, 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
	CLI		; 全割り込み禁止
WAIT:	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

27.8.14. Atmel ATmega16M1/32M1/64M1用ブートローダのパラメータ

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27-7. 応用領域とブートローダ領域の分割設定 (上段:ATmega16M1,中段:ATmega32M1,下段:ATmega64M1)

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	256	4	\$0000~\$1EFF	\$1F00~\$1FFF	\$1F00
		256	4	\$0000~\$3EFF	\$3F00~\$3FFF	\$3F00
		512	4	\$0000~\$7DFF	\$7E00~\$7FFF	\$7E00
1	0	512	8	\$0000~\$1DFF	\$1E00~\$1FFF	\$1E00
		512	8	\$0000~\$3DFF	\$3E00~\$3FFF	\$3E00
		1024	8	\$0000~\$7BFF	\$7C00~\$7FFF	\$7C00
0	1	1024	16	\$0000~\$1BFF	\$1C00~\$1FFF	\$1C00
		1024	16	\$0000~\$3BFF	\$3C00~\$3FFF	\$3C00
		2048	16	\$0000~\$77FF	\$7800~\$7FFF	\$7800
0	0	2048	32	\$0000~\$17FF	\$1800~\$1FFF	\$1800
		2048	32	\$0000~\$37FF	\$3800~\$3FFF	\$3800
		4096	32	\$0000~\$6FFF	\$7000~\$7FFF	\$7000

注: 各種BOOTSZヒューズ設定は図27-2.で示されます。

表27-8. RWW領域とNRWW領域の範囲 (上段:ATmega16M1,中段:ATmega32M1,下段:ATmega64M1)

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~\$17FF
	224	\$0000~\$37FF
	224	\$0000~\$6FFF
書き込み中読み出し不能(NRWW)領域	32	\$1800~\$1FFF
	32	\$3800~\$3FFF
	32	\$7000~\$7FFF

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」をご覧ください。

表27-9. 図27-3.で使った各変数説明とZポイントの配置 (上段:ATmega16M1,中段:ATmega32M1,下段:ATmega64M1)

変数名	対応値		意味
	PC	Zポイント(注)	
PCMSB	PC12		(プログラムカウンタは13ビット、PC12~0)
	PC13		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14ビット、PC13~0)
	PC14		(プログラムカウンタは15ビット、PC14~0)
PAGEMSB	PC5		1ページ内に使われる (ページ内の64語には6ビット PC5~0が必要)
	PC5		語(ワード)アドレスの最上位ビット。(ページ内の64語には6ビット PC5~0が必要)
	PC6		(ページ内の128語には7ビット PC6~0が必要)
ZPCMSB		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)
		Z14	
		Z15	
ZPAGEMSB		Z6	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	
		Z7	
PCPAGE	PC12~6	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
	PC13~6	Z14~7	
	PC14~7	Z15~8	
PCWORD	PC5~0	Z6~1	プログラムカウンタ 語(ワード)アドレス: 一時緩衝部格納用語(ワード)選択 (ページ書き込み操作中は0でなければなりません。)
	PC5~0	Z6~1	
	PC6~0	Z7~1	

注: Z15,Z14: ATmega16M1ではZ15とZ14, ATmega32M1ではZ15, これらは常に無視されます。

Z0: 全てのSPM命令に対して0であるべきで、LPM命令に対するビット選択です。

自己プログラミング中のZポイントの使用についての詳細に関しては「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」をご覧ください。

(訳注) 原書の表27-7.~15.は表27-7.~9.として纏めました。

27.9. ブート ロード用レジスタ

27.9.1. 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 ロード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。

28. MEMPROG – メモリプログラミング

28.1. プログラムメモリとデータメモリ用施錠ビット

本デバイスは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」の表で一覧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。

表28-1. 施錠ビットバイトの内容

名称	ビット番号	意味	既定値 (注)
-	7		1 (非プログラム)
-	6		1 (非プログラム)
BLB12	5	ブートローダ領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB11	4		1 (非プログラム)
BLB02	3	応用領域に対する保護用ブート施錠ビット	1 (非プログラム)
BLB01	2		1 (非プログラム)
LB2	1	フラッシュとEEPROMメモリに対する一般保護用施錠ビット	1 (非プログラム)
LB1	0		1 (非プログラム)

注: 0はプログラム、1は非プログラムを意味します。

表28-2. 施錠ビットの保護種別

メモリ施錠ビット			保護種別
LB 種別	LB2	LB1	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)
LB0 種別	BLB02	BLB01	フラッシュメモリの応用プログラム領域に対する保護
1	1	1	LPM, SPM命令が応用領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令は応用領域に書くことを許されません。
3	0	0	SPM命令による応用領域への書き込みと、ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
4	0	1	ブートローダ領域でのLPM命令による応用領域からの読み込みが許されません。(注2)
LB1 種別	BLB12	BLB11	フラッシュメモリのブートローダプログラム領域に対する保護
1	1	1	LPM, SPM命令がブートローダ領域をアクセスすることに対して制限はありません。
2	1	0	SPM命令はブートローダ領域に書くことを許されません。
3	0	0	SPM命令によるブートローダ領域への書き込みと、応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)
4	0	1	応用領域でのLPM命令によるブートローダ領域からの読み込みが許されません。(注3)

注: 0はプログラム、1は非プログラムを意味します。

注1: 施錠ビットを書く前にヒューズビットとブート施錠ビットを書いてください。

注2: BLB02=0で、割り込みベクタがブートローダ領域に配置されていると、応用領域での実行時に割り込みが禁止されます。

注3: BLB12=0で、割り込みベクタが応用領域に配置されていると、ブートローダ領域での実行時に割り込みが禁止されます。

(訳注) 原書の表28-2、表28-3、表28-4は表28-2として纏めました。

28.2. 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28-5. 拡張ヒューズ バイト一覧

名称	ビット	意味	既定値
-	7	-	1 (非プログラム)
-	6	-	1 (非プログラム)
PSCRB	5	電力段制御器(PSC)出力 リセット時動作	1 (非プログラム) 標準ポート動作
PSCRVA	4	PSCOUTnAリセット値	1 (非プログラム) Low
PSCRVB	3	PSCOUTnBリセット値	1 (非プログラム) Low
BODLEVEL2	2	低電圧検出(BOD)リセットの制御と検出電圧選択。 (注1)	1 (非プログラム)
BODLEVEL1	1		1 (非プログラム)
BODLEVEL0	0		1 (非プログラム)

注1: BODLEVELヒューズの符号化については「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。

28.2.1. リセット中の電力段制御器(PSC)出力の動き

外部部品の安全のため、リセット中のPSC出力状態はPSCRB,PSCRVA,PSCRVBヒューズによってプログラム(設定)できます。

これらのヒューズは拡張ヒューズ バイトに配置されます。

PSCRBヒューズが1に等しい(非プログラム)なら、全てのPSC出力は標準ポート動作を保ちます。PSCRBヒューズが0に等しい(プログラム)なら、全てのPSC出力はリセットでPSCRVAとPSCRVBヒューズ ビットに従ってHighまたはLowレベルに強制されます。後者の場合、PSC出力はPSCレジスタが書かれるまで、この強制された状態を保ちます。

PSCRVA(PSCOUTnAリセット値)はPSCRBヒューズがプログラム(0)された時にPSCOUT0A,PSCOUT1A,PSCOUT2A出力で強制するLowまたはHighの状態を与えます。PSCRVAヒューズが0に等しい(プログラム)なら、PSCOUT0A,PSCOUT1A,PSCOUT2A出力はHigh状態を強制されます。PSCRVAヒューズが1に等しい(非プログラム)なら、PSCOUT0A,PSCOUT1A,PSCOUT2A出力はLow状態を強制されます。

PSCRVB(PSCOUTnBリセット値)はPSCRBヒューズがプログラム(0)された時にPSCOUT0B,PSCOUT1B,PSCOUT2B出力で強制するLowまたはHighの状態を与えます。PSCRVBヒューズが0に等しい(プログラム)なら、PSCOUT0B,PSCOUT1B,PSCOUT2B出力はHigh状態を強制されます。PSCRVBヒューズが1に等しい(非プログラム)なら、PSCOUT0B,PSCOUT1B,PSCOUT2B出力はLow状態を強制されます。

表28-6. リセット中とリセット後にPOCレジスタが書かれるまでのPSC出力の動き

PSCRB	PSCRVA	PSCRVB	PSCOUTnA	PSCOUTnB
非プログラム(1)	x	x	標準ポート	標準ポート
プログラム(0)	非プログラム(1)	非プログラム(1)	Low強制	Low強制
プログラム(0)	非プログラム(1)	プログラム(0)	Low強制	High強制
プログラム(0)	プログラム(0)	非プログラム(1)	High強制	Low強制
プログラム(0)	プログラム(0)	プログラム(0)	High強制	High強制

表28-7. 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	PE0がI/OピンかまたはRESETピンかを選びます。	1 (非プログラム) PE0はRESETピン
DWEN	6	デバッグWIRE機能許可。	1 (非プログラム) デバッグWIRE不許可
SPIEN(注2)	5	直列プログラミング許可。	0 (プログラム) 直列プログラミング許可
WDTON(注3)	4	ウォッチドッグ タイマ常時有効。	1 (非プログラム) WDTはWDTCSRで許可
EESAVE	3	チップ消去からEEPROM内容を保護。	1 (非プログラム) EEPROMは未保護
BOOTSZ1	2	ブート ロータ容量選択。 (注4)	0 (プログラム)
BOOTSZ0	1		0 (プログラム)
BOOTRST	0	リセット ベクタ(応用領域またはブート ロータ領域)選択	1 (非プログラム)

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

注2: SPIENヒューズは直列プログラミングでアクセスできません。

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

注4: BOOTSZ1,0既定値は最大ブート ロータ容量になります。表27-7をご覧ください。

表28-8. 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の既定値は最大起動時間になります。詳細については表9-6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。詳細については表9-1をご覧ください。

注3: CKOUTはポートD1に出力することをシステム クロックに許します。詳細については「クロック出力緩衝部」をご覧ください。

注4: 詳細については「システム クロック前置分周器」をご覧ください。

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

- 関連リンク
- 229ページの「システムとリセットの特性」
 - 54ページの「交換ポート機能」
 - 25ページの「校正付き内蔵RC発振器」
 - 39ページの「WDTCSR - ウォッチドッグ タイマ制御レジスタ」

28.2.2. 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。

28.3. 識票バイト

本デバイスは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28-9. 識票バイト

部品番号	識票バイト アドレス		
	\$0000	\$0001	\$0002
ATmega16M1	\$1E	\$94	\$84
ATmega32M1	\$1E	\$95	\$84
ATmega64M1	\$1E	\$96	\$84

28.4. 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。

28.5. ページ容量

表28-10. 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
ATmega16M1	8K語 (16Kバイト)	64語	PC5~0	128	PC12~6	12
ATmega32M1	16K語 (32Kバイト)	64語	PC5~0	256	PC13~6	13
ATmega64M1	32K語 (64Kバイト)	128語	PC6~0	256	PC14~7	14

表28-11. 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
ATmega16M1	512バイト	4バイト	EEA1~0	128	EEA8~2	8
ATmega32M1	1Kバイト	4バイト	EEA1~0	256	EEA9~2	9
ATmega64M1	2Kバイト	8バイト	EEA2~0	256	EEA10~3	10

28.6. 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。

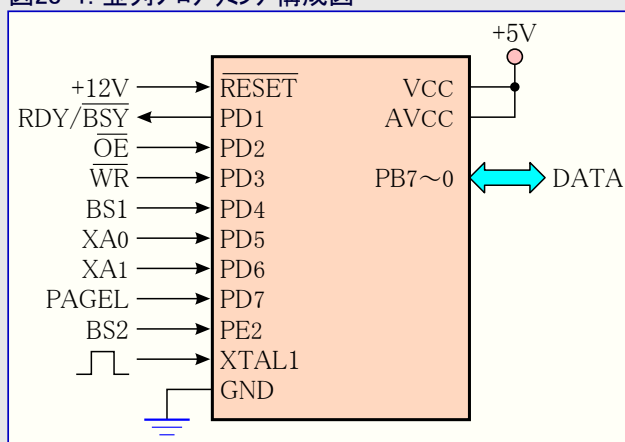
28.6.1. 信号名

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ピンが正ハ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ハルス送出時、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28-1. 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28-12. 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	PE2	入力	上位/下位バイト選択2 (0:下位, 1:上位) (ヒューズビット用)
DATA	PB7~0	入出力	双方向データバス (OE=Low時出力)

表28-13. プログラミング動作移行時のピン値

ピン名	シンボル	値
PAGEL	Prog_enable[3]	0
XA1	Prog_enable[2]	0
XA0	Prog_enable[1]	0
BS1	Prog_enable[0]	0

表28-14. XA0とXA1の符号化(機能)

XA1	XA0	XTAL1ハルス時の動作
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28-15. 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

28.7. 並列プログラミング手順

28.7.1. 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28-13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 μ 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 μ 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 μ 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 μ 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28-13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 μ 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。

28.7.2. 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。

28.7.3. 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。

28.7.4. フラッシュ メモリ書き込み (次頁の図28-3.タイミングを参照)

フラッシュ メモリは表28-10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～\$1F/\$3F/\$7F)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

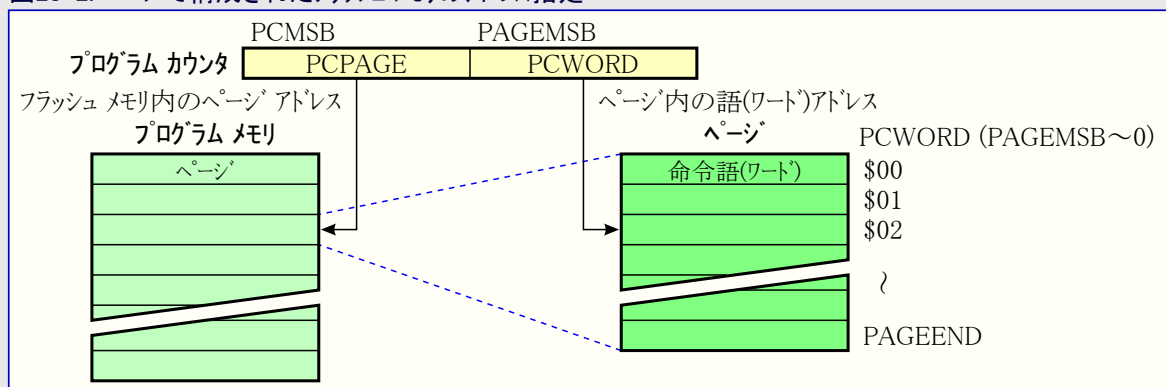
- ① BS1をLow(0)に設定します。
- ② WRに負パルスを与えます。これはデータのページ全体の書き込みを開始します。RDY/BSYがLow(0)になります。
- ③ RDY/BSYがHigh(1)になるまで待ちます。

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

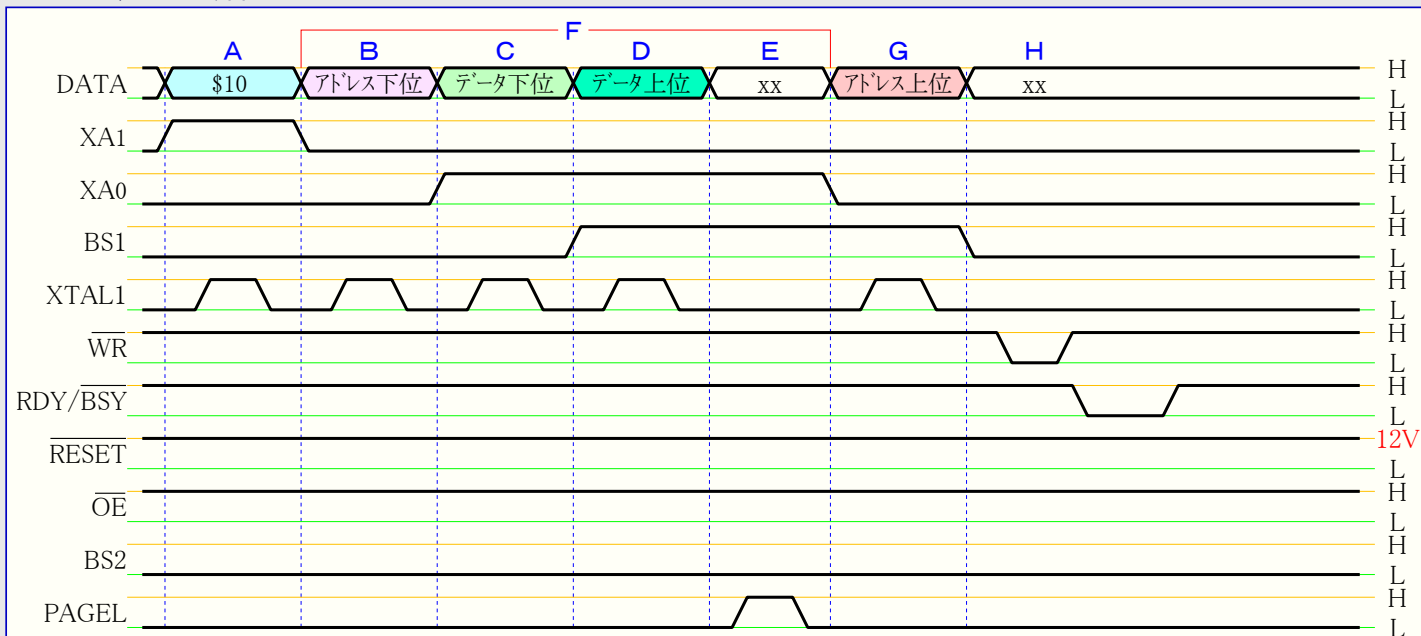
- ① XA1をHigh(1)、XA0をLow(0)に設定します。これは指令設定を許可します。
- ② DATAを\$00(0000 0000)にします。これは無操作指令です。
- ③ XTAL1に正パルスを与えます。これは無操作指令を設定し、内部書き込み信号がリセットされます。

図28-2. 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28-10.で一覧されます。

図28-3. フラッシュメモリ書き込みタイミング



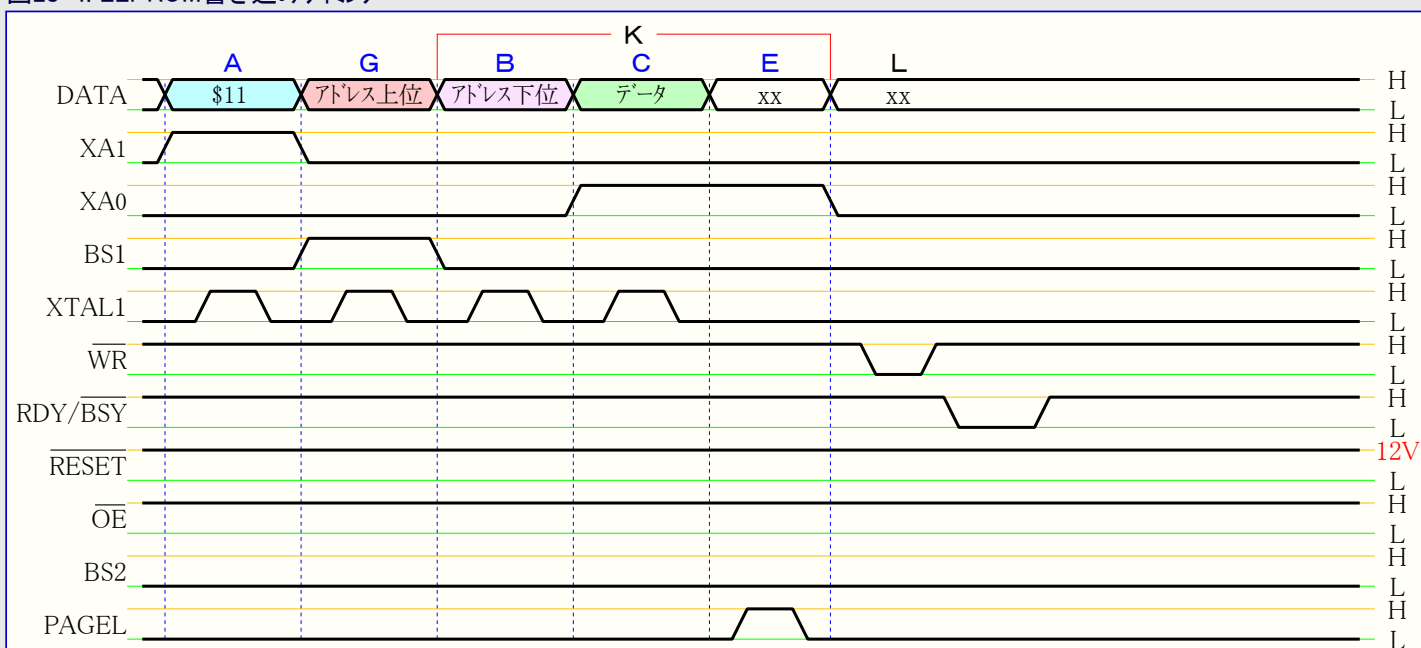
注: xx値は無関係です。A～Hは前記プログラミングを参照してください。

28.7.5. EEPROM書き込み

EEPROMは表28-11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28-4タイミング参照。)

1. EEPROM書き込み指令\$11(0001 0001)を設定します。(「フラッシュメモリ書き込み」のAを参照)
2. アドレス上位バイト(\$00～\$01/\$03/\$07)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28-4. EEPROM書き込みタイミング



28.7.6. 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~\$1F/\$3F/\$7F)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。

28.7.7. 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~\$01/\$03/\$07)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。

28.7.8. 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。)

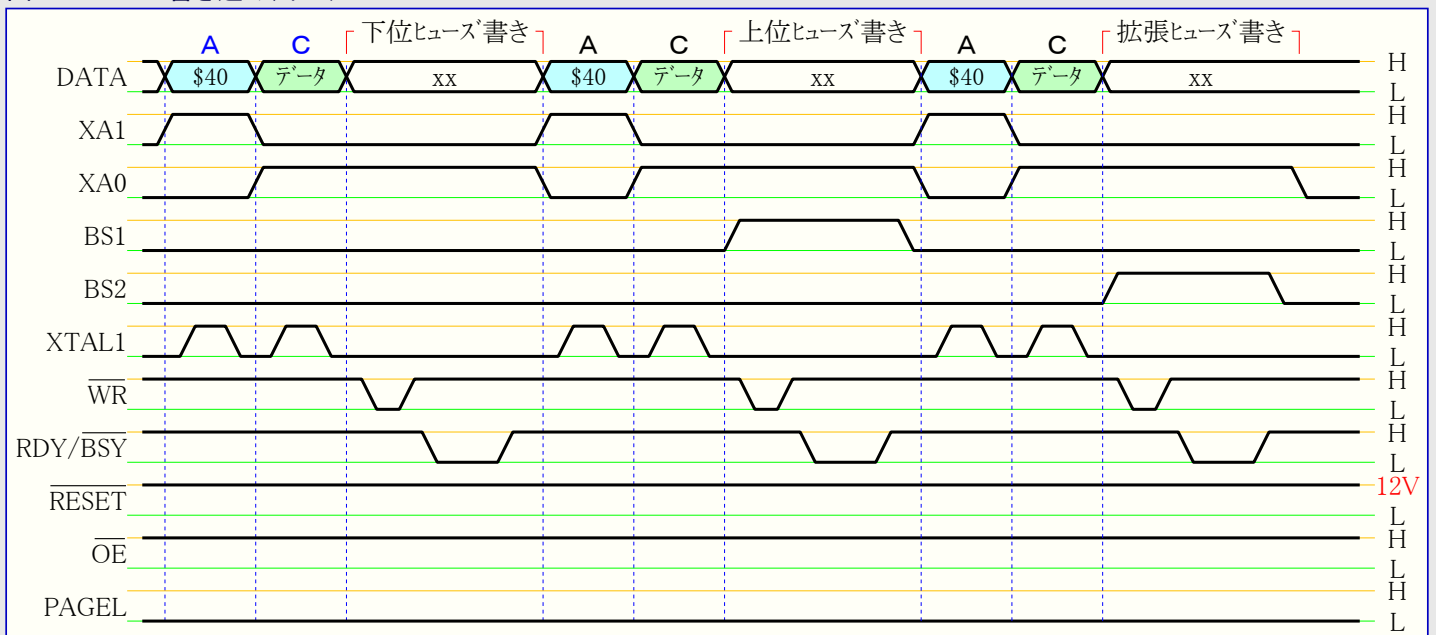
各ヒューズ バイトの書き込み方法は次のとおりです。(指令とデータ設定の詳細については「フラッシュ メリの書き込み」を参照)

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28-5. ヒューズ書き込みタイミング



28.7.9. 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。

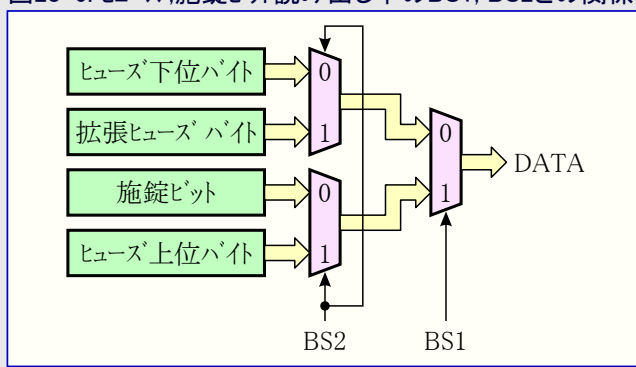
施錠ビットはチップ消去の実行によってのみ解除(1)できます。

28.7.10. 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28-6. ヒューズ、施錠ビット読み出し中のBS1, BS2との関係



28.7.11. 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。

28.7.12. 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。

28.7.13. 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。

28.8. 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。

28.8.1. 直列プログラミング用ピン配置

表28-16. 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
MOSLA	PD3	入力	直列データ入力
MISO_A	PD2	出力	直列データ出力
SCK_A	PD4	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。

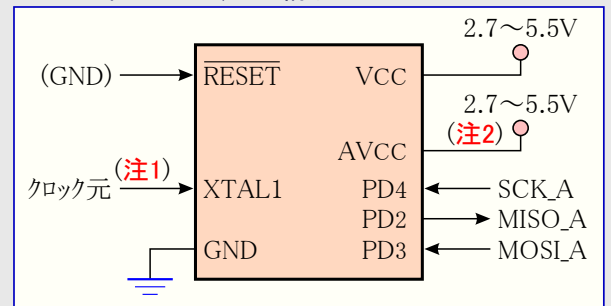
EEPROMをプログラミングする時に自動消去周期が自動書き込み動作内に組み入れられ(直列プログラミングのみ)、チップ消去命令を初めに実行する必要があります。チップ消去操作はプログラム(フラッシュメモリ)とEEPROM両方の全てのメモリ位置の内容を\$FFにします。

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28-7. 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2.7~5.5V内にすべきです。

28.9. 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28-18.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28-10.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6/7+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7/8/8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2/2/3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7/8/8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28-11.参照)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28-17. 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

28.9.1. 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28-18. 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。

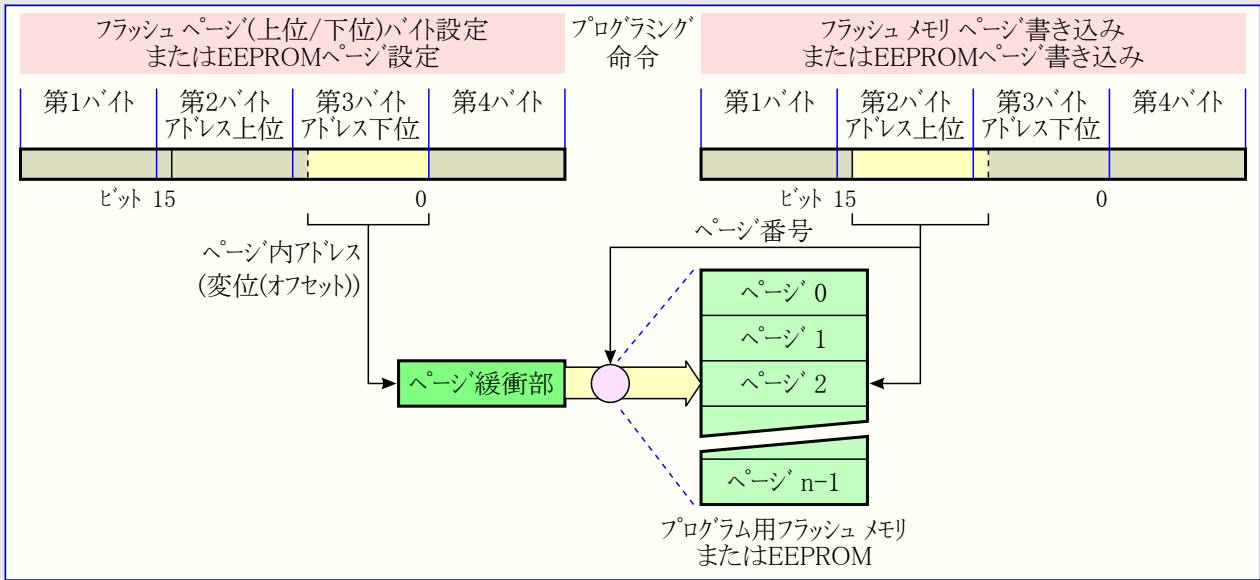
- ・ヒューズ、施錠ビット、識票バイト、校正バイト、ページ容量については対応項を参照してください。
- ・プログラミングと書き込み器に関する応用記述については <http://www.atmel.com/avr> をご覧ください。
- ・第4バイトの赤背景はホスト読み込み(デバイス出力)を示します。

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28-8. 直列プログラミング命令例



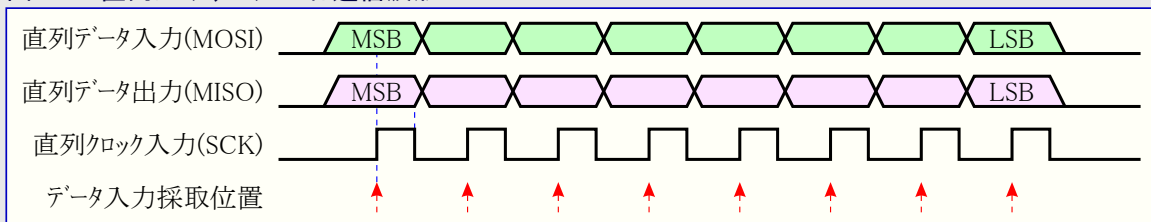
(訳補) フラッシュメモリ、EEPROM、ページ緩衝部内のアドレス(位置)指定に使われるビットはメモリ容量とページ構成に依存します。ATmega16M1/32M1/64M1でのこれらの指定方法は次表で要約されます。

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	00LL LLLL 0LLL LLLL	ATmega16M1/32M1 : L=PC5~0 ATmega64M1 : L=PC6~0
EEPROM ページ内バイト設定	0000 0000	0000 00LL 0000 0LLL	ATmega16M1/32M1 : L=EEA1~0 ATmega64M1 : L=EEA2~0
フラッシュメモリ読み出し	000H HHHH 00HH HHHH 0HHH HHHH	LLLL LLLL	ATmega16M1 : H=PC12~8, L=PC7~0 ATmega32M1 : H=PC13~8, L=PC7~0 ATmega64M1 : H=PC14~8, L=PC7~0
EEPROM読み出し	0000 000H 0000 00HH 0000 0HHH	LLLL LLLL	ATmega16M1 : H=EEA8, L=EEA7~0 ATmega32M1 : H=EEA9~8, L=EEA7~0 ATmega64M1 : H=EEA10~8, L=EEA7~0
フラッシュ ページ書き込み	000H HHHH 00HH HHHH 0HHH HHHH	LL00 0000 LL00 0000 L000 0000	ATmega16M1 : H=PC12~8, L=PC7~6 ATmega32M1 : H=PC13~8, L=PC7~6 ATmega64M1 : H=PC14~8, L=PC7
EEPROMバイト書き込み	0000 000H 0000 00HH 0000 0HHH	LLLL LLLL	ATmega16M1 : H=EEA8, L=EEA7~0 ATmega32M1 : H=EEA9~8, L=EEA7~0 ATmega64M1 : H=EEA10~8, L=EEA7~0
EEPROM ページ書き込み	0000 000H 0000 00HH 0000 0HHH	LLLL LL00 LLLL LL00 LLLL L000	ATmega16M1 : H=EEA8, L=EEA7~2 ATmega32M1 : H=EEA9~8, L=EEA7~2 ATmega64M1 : H=EEA10~8, L=EEA7~3

28.9.2. 直列プログラミング特性

図28-9. 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。

29. 電気的特性

29.1. 絶対最大定格

表29-1. 絶対最大定格

動作温度	-40°C~85°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。

29.2. DC特性

表29-2. DC特性 TA=-40°C~85°C, VCC=2.7V~5.5V (特記事項を除く)

シンボル	項目	条件	最小	代表	最大	単位
V _{IL}	Lowレベル入力電圧 (一般/XTALポート)	ポートB,C,DとI/O使用 のXTAL1,XTAL2	-0.5		0.2VCC (注1)	V
V _{IL1}	Lowレベル入力電圧 (XTAL1)	外部クロック信号選択	-0.5		0.1VCC (注1)	
V _{IL2}	Lowレベル入力電圧 (RESET)	外部リセット入力	-0.5		0.2VCC (注1)	
V _{IL3}	Lowレベル入力電圧 (RESET)	I/Oとして使用	-0.5		0.2VCC (注1)	
V _{IH}	Highレベル入力電圧 (一般/XTALポート)	ポートB,C,DとI/O使用 のXTAL1,XTAL2	0.6VCC (注2)		VCC+0.5	
V _{IH1}	Highレベル入力電圧 (XTAL1)	外部クロック信号選択	0.7VCC (注2)		VCC+0.5	
V _{IH2}	Highレベル入力電圧 (RESET)	外部リセット入力	0.9VCC (注2)		VCC+0.5	
V _{IH3}	Highレベル入力電圧 (RESET)	I/Oとして使用	0.8VCC (注2)		VCC+0.5	
V _{OL}	Lレベル出力電圧 (注3)	I/Oとしての RESET以外	IOL=10mA, VCC=5V		0.7	
			IOL=6mA, VCC=3V		0.5	
V _{OL3}	RESET=I/O		IOL=2.1mA, VCC=5V		0.9	
			IOL=0.8mA, VCC=3V		0.7	
V _{OH}	Hレベル出力電圧 (注4)	I/Oとしての RESET以外	IOH=-10mA, VCC=5V	4.2		
			IOH=-8mA, VCC=3V	2.3		
V _{OH3}	RESET=I/O		IOH=-0.6mA, VCC=5V	3.8		
			IOH=-0.4mA, VCC=3V	2.2		
I _{IL}	I/OピンLowレベル入力漏れ電流	VCC=5.5V			50	nA
I _{IH}	I/OピンHighレベル入力漏れ電流	確実なH/L範囲			50	
RRST	RESETピンプルアップ抵抗		30		200	kΩ
RPU	I/Oピンプルアップ抵抗		20		50	
I _{CC}	活動動作消費電流 (電力削減レジスタ(PRR)全ビット=1)	VCC=3V, 8MHz(RC)		3.8	8	mA
		VCC=5V, 16MHz(外部)		14	30	
	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=3V, 8MHz(RC)		1.5	4	μA
		VCC=5V, 16MHz(外部)		5.5	15	
パワーダウン動作消費電流 (注5)		VCC=3V, TA<90°C		5	15	μA
		WDT有効, TA<105°C		9	90	
		VCC=3V, TA<90°C		2	10	
		WDT禁止, TA<105°C		5	80	
V _{hysr}	アナログ比較器ヒステリシス電圧	VCC=5V, Vin=3V	上昇 下降	25 -35	70	mV
I _{ACLK}	アナログ比較器入力漏れ電流	VCC=5V, Vin=VCC/2		-50	50	
t _{ACPD}	アナログ比較器伝播遅延時間	VCC=2.7V		(注6)		ns
		VCC=5.0V		(注6)		

(注1)~(注6)は次頁を参照してください。

注1: Lowレベルの認識が保証される最高電圧です。

注2: Highレベルの認識が保証される最低電圧です。

注3: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3Vで6mA, VCC=5Vで10mA)よりも多くの吸い込み電流を流すことができますが、次の条件を厳守しなければなりません。

1. ポートB1,0、C3,2、D4、E2,1のIOLの合計が70mAを超えるべきではありません。
2. ポートB7,6、C1,0、D3~0、E0のIOLの合計が70mAを超えるべきではありません。
3. ポートB5~2、C7~4、D7~5のIOLの合計が70mAを超えるべきではありません。

IOLが検査条件を超える場合、VOLも仕様書での値を超えます。表の検査条件よりも大きな吸い込み電流を流すことは保証されません。

注4: 各I/Oポートは安定状態(非過渡時)に於いて検査条件(VCC=3Vで8mA, VCC=5Vで10mA)よりも多くの吐き出し電流を流すことができますが、次の条件を厳守しなければなりません。

1. ポートB1,0、C3,2、D4、E2,1のIOHの合計が100mAを超えるべきではありません。
2. ポートB7,6、C1,0、D3~0、E0のIOHの合計が100mAを超えるべきではありません。
3. ポートB5~2、C7~4、D7~5のIOHの合計が100mAを超えるべきではありません。

IOHが検査条件を超える場合、VOHも仕様書での値を超えます。表の検査条件よりも大きな吐き出し電流を流すことは保証されません。

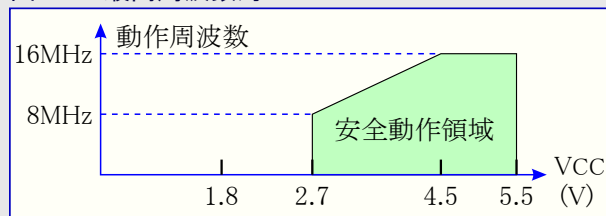
注5: パワーダウン動作に対する最小VCCは2.5Vです。

注6: アナログ比較器伝播遅延は1比較器クロック+30nsと同等です。比較器クロック定義については「[アナログ比較器](#)」をご覧ください。

29.3. 速度勾配

最高周波数は動作電圧に依存します。右図で示されるように最高周波数対動作電圧曲線は2.7~4.5V間で直線です。

図29-1. 最高周波数対VCC



29.4. クロック特性

29.4.1. 校正付き内蔵RC発振器の精度

表29-2. 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	2.7~5.5V	-40~85°C	±1%

29.4.2. 外部クロック信号駆動

図29-2. 外部クロック駆動波形

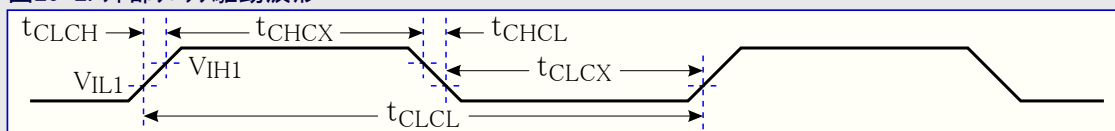


表29-3. 外部クロック特性

シンボル	項目	VCC=2.7~5.5V		VCC=4.5~5.5V		単位
		最小	最大	最小	最大	
$1/t_{CLCL}$	クロック周波数	0	8	0	16	MHz
t_{CLCL}	クロック周期	125		62.5		
t_{CHCX}	Highレベル時間	50		25		ns
t_{CLCX}	Lowレベル時間	50		25		
t_{CLCH}	上昇時間		1.6		0.5	μ s
t_{CHCL}	下降時間		1.6		0.5	
Δt_{CLCL}	隣接クロック周期間の変化率		2		2	%

29.5. システムとリセットの特性

表29-4. リセット、低電圧検出(BOD)(注1)、内部基準電圧(注1)の特性

シンボル	項目	最小	代表	最大	単位
V_{POT}	上昇時電源ONリセット閾値電圧	1.1	1.4	1.7	V
	下降時電源ONリセット閾値電圧 (注2)	0.8	0.9	1.6	
V_{PORMAX}	内部電源ONリセット信号を保证するためのVCC最大開始電圧			0.4	
V_{PORMIN}	内部電源ONリセット信号を保证するためのVCC最小開始電圧	-0.1			
V_{CCRR}	内部電源ONリセットを保证するためのVCC上昇勾配	0.01			V/ms
V_{RST}	RESETピン閾値電圧	0.1VCC		0.9VCC	V
t_{RST}	RESETピンでの最小パルス幅	TBD			μ 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		μ s
V_{BG}	基準電圧		1.1		V
t_{BG}	起動時間		40		μ s
I_{BG}	消費電流		15		μ A

注1: 値は指針の意味だけです。

注2: 上昇に先立ち、リセットを保证するために供給電圧はVPORMINとVPORMAX間でなければなりません。

表29-5. BODLEVEL_{2~0} (V_{BOT}) 設定 (注1,2)

BODLEVEL _{2~0}	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		4.5		V
1 0 1		2.7		
1 0 0		4.3		
0 1 1		4.4		
0 1 0		4.2		
0 0 1		2.8		
0 0 0		2.6		

注1: いくつかのデバイスでV_{BOT}が公称最低動作電圧以下の可能性があります。この状態のデバイスについては、製造検査中、VCC=V_{BOT}に落ちて検査されます。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。この検査は低動作電圧についてBODLEVEL=101と高動作電圧についてBODLEVEL=010を使って実行されます。

注2: 値は指針の意味だけです。

29.6. PLL特性

表29-6. PLL特性 VCC=2.7V~5.5V (特記事項を除く)

シンボル	項目	最小	代表	最大	単位
PLLIF	入力周波数	0.5	1	2	MHz
PLLF	PLL係数(倍率)		64		
PLLLT	固定化時間			64	μs

注: 外部クロック信号または外部発振器接続時、PLL入力周波数は回路の駆動部分(CPUコア,PSC...)の駆動に対応する周波数で出力を提供するよう選ばなければなりません。

29.7. SPIタイミング特性

図29-3. SPI タイミング必要条件 (主装置動作)

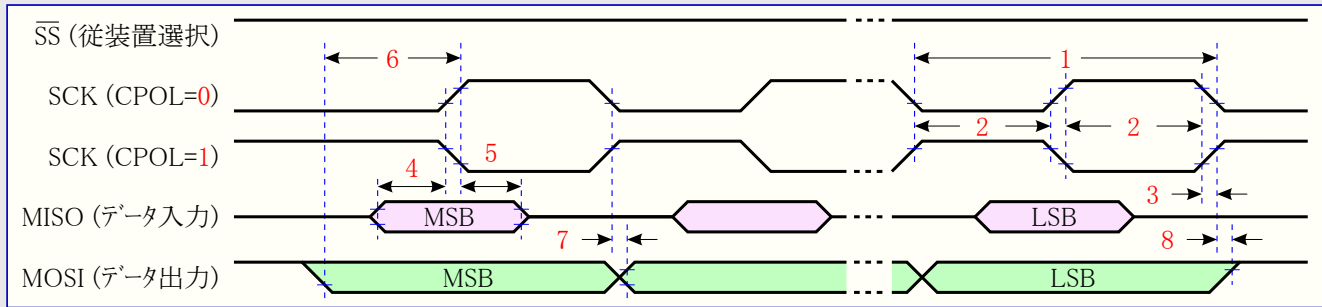


図29-4. SPI タイミング必要条件 (従装置動作)

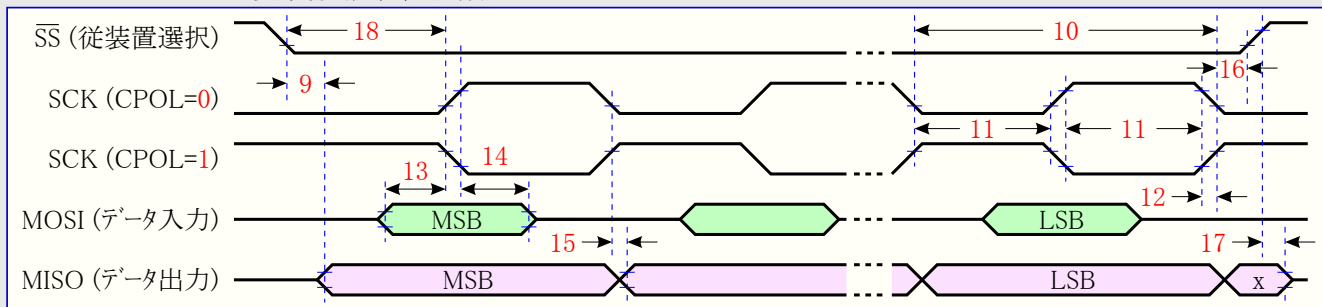


表29-7. SPI タイミング特性

番号	項目	動作種別	最小	代表	最大	単位
1	SCK周期	主装置		表19-5.参照		
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		0.5t _{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	4t _{CK}			
11	SCK High/Low期間 (注)	従装置	2t _{CK}			
12	SCK上昇/下降時間	従装置			16	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	?		10		
18	SS ↓からのSCK遅延時間	従装置	20			

注: SPIプログラミングでの最小SCK High/Low期間は、2t_{CLCL}(t_{CK}<12MHz)、3t_{CLCL}(t_{CK}≥12MHz)です。

29.8. A/D変換器特性

表29-8. A/D変換特性 (TA=-40~+85°C, VCC=2.7~5.5V) (特記事項を除く)

シンボル	項目	条件		最小	代表	最大	単位
シングル エンド入力変換	分解能				10		ビット
	TUE	絶対精度	VCC=5V, VREF=2.56V	変換クロック=1MHz	3.2	5.0	
				変換クロック=2MHz	3.2	5.0	
	INL	積分非直線性誤差	VCC=5V, VREF=2.56V	変換クロック=1MHz	0.7	1.5	
				変換クロック=2MHz	0.8	2.0	
	DNL	微分非直線性誤差	VCC=5V, VREF=2.56V	変換クロック=1MHz	0.5	0.8	
				変換クロック=2MHz	0.6	1.2	LSB
		利得誤差	VCC=5V, VREF=2.56V	変換クロック=1MHz	-8.0	-5.0	0.0
変換クロック=2MHz				-8.0	-5.0	0.0	
	変位(オフセット)(ゼロ)誤差	VCC=5V, VREF=2.56V	変換クロック=1MHz	-2.0	2.5	6.0	
			変換クロック=2MHz	-2.0	2.5	6.0	
VREF	基準電圧			2.56		AVCC	V
差動入力変換	分解能	差動変換、温度=-40~85°C			8		ビット
	TUE	絶対精度	VCC=5V, VREF=2.56V, 変換クロック=2MHz	利得=5倍	1.5	3.5	
				利得=20倍	1.5	4.0	
				利得=40倍	1.5	4.5	
	INL	積分非直線性誤差	VCC=5V, VREF=2.56V	変換クロック=2MHz, 利得=5倍	0.1	1.5	
				変換クロック=2MHz, 利得=20倍	0.2	2.5	
				変換クロック=1MHz, 利得=40倍	0.3	3.5	
				変換クロック=2MHz, 利得=40倍	0.7	4.5	LSB
	DNL	微分非直線性誤差	VCC=5V, VREF=2.56V, 変換クロック=2MHz	利得=5倍	0.1	1.0	
				利得=20倍	0.2	1.5	
				利得=40倍	0.3	2.5	
	利得誤差	VCC=5V, VREF=2.56V, 変換クロック=2MHz	利得=5倍, 10倍	-3.0		3.0	
			利得=20倍, 40倍	-3.0		3.0	
	変位(オフセット)(ゼロ)誤差	VCC=5V, VREF=2.56V, 変換クロック=2MHz	利得=5倍, 10倍	-3.0		3.0	
			利得=20倍, 40倍	-4.0		4.0	
VREF	基準電圧			2.56		AVCC-0.5	V

29.9. 並列プログラミング特性

図29-5. 並列プログラミング タイミング (一般的な必要条件)

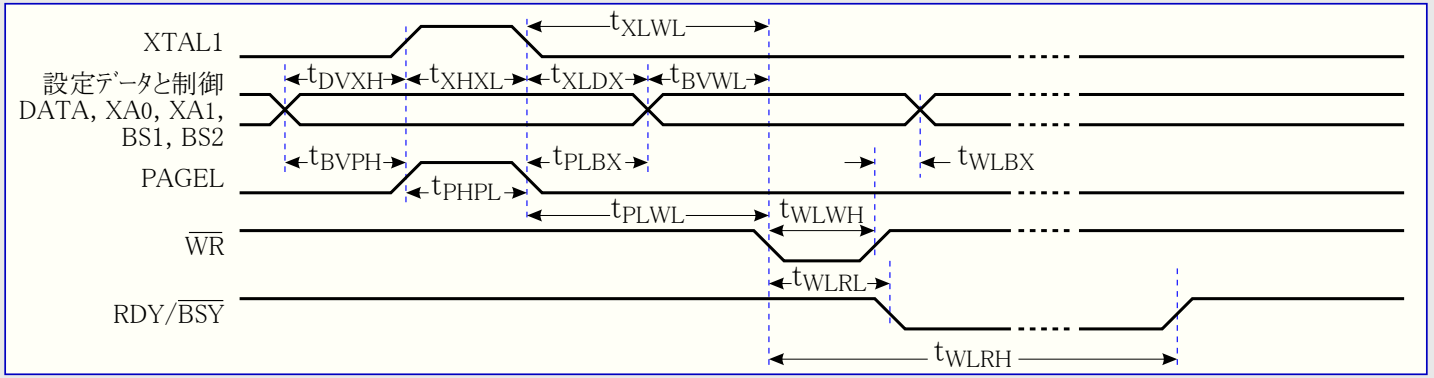
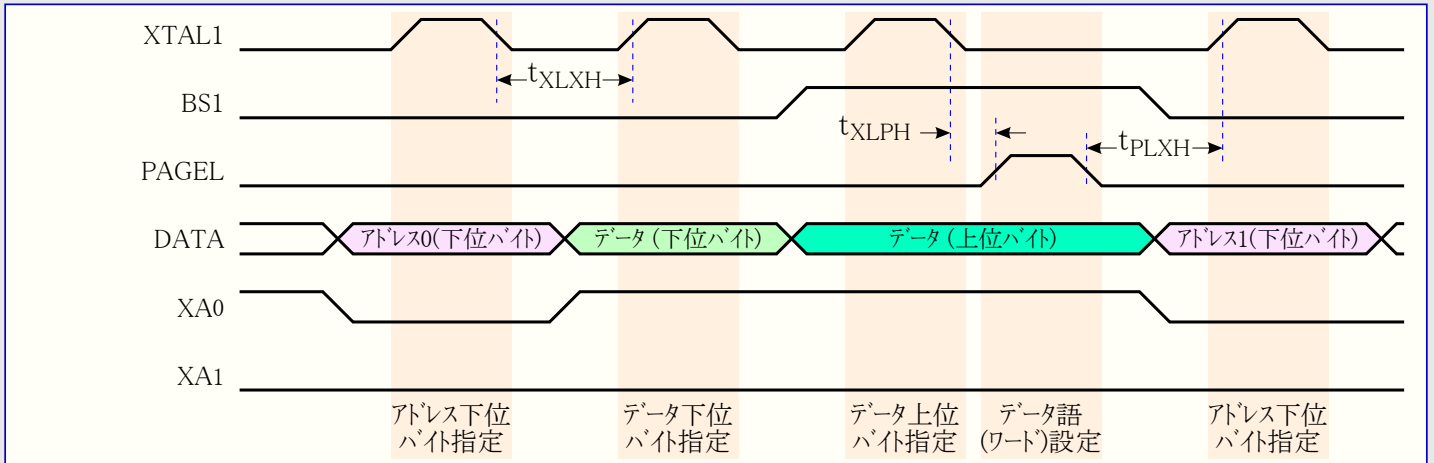
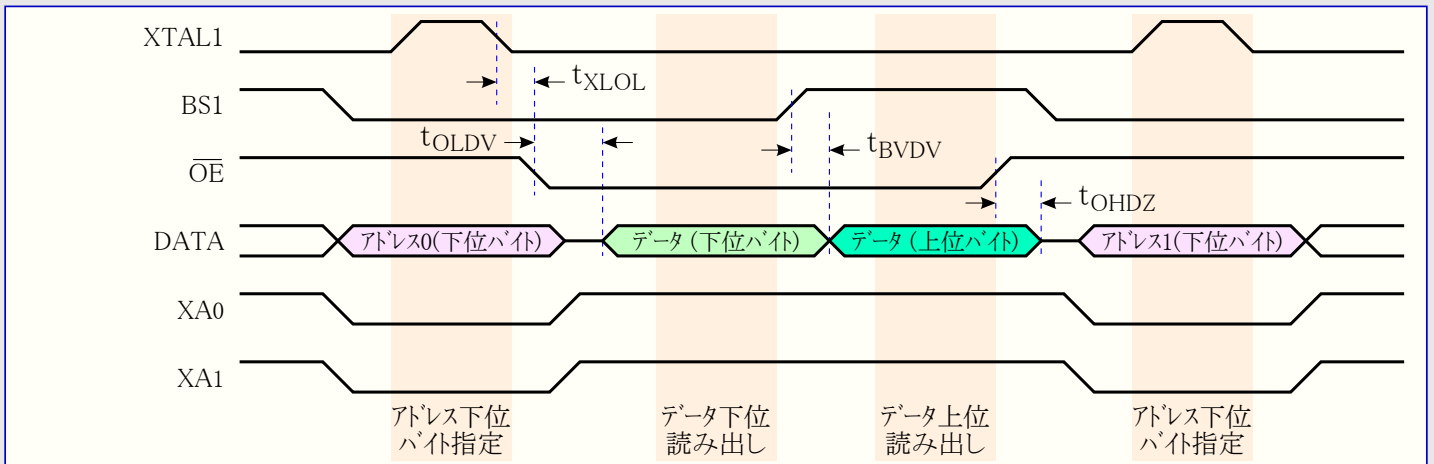


図29-6. 並列プログラミング タイミング (ページ設定での必要条件)



注: 図29-5.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHXL} 、 t_{XLDX})は設定操作にも適用されます。

図29-7. 並列プログラミング タイミング (同一ページ読み出しでの必要条件)



注: 図29-5.で示されるタイミング必要条件(即ち、 t_{DVXH} 、 t_{XHXL} 、 t_{XLDX})は読み出し操作にも適用されます。

表29-9. 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 _{PP}	プログラミング許可電圧	11.5		12.5	V
I _{PP}	プログラミング許可電流			250	μA
t _{DVXH}	XTAL1 ↑ に対するデータと制御の準備時間	67			
t _{XLXH}	XTAL1 ↓ から次XTAL1 ↑ までの待機時間	200			
t _{XHXL}	XTAL1 Highパルス幅	150			
t _{XLDX}	XTAL1パルス ↓ 後のデータと制御の保持時間	67			
t _{XLWL}	XTAL1パルス ↓ 後の \overline{WR} ↓ 待機時間	0			
t _{XLPH}	XTAL1パルス ↓ 後のPAGELパルス ↑ 待機時間	0			
t _{PLXH}	PAGELパルス ↓ 後のXTAL1パルス ↑ 待機時間	150			ns
t _{BVPH}	PAGELパルス ↑ に対するBS1準備時間	67			
t _{PHPL}	PAGEL Highパルス幅	150			
t _{PLBX}	PAGELパルス ↓ 後のBS1保持時間	67			
t _{WL BX}	\overline{WR} パルス ↓ 後のBS1,BS2保持時間	67			
t _{PLWL}	PAGELパルス ↓ 後の \overline{WR} パルス ↓ 待機時間	67			
t _{BVWL}	\overline{WR} パルス ↓ に対するBS1準備時間	67			
t _{WLWH}	\overline{WR} Lowパルス幅	150			
t _{WLRL}	\overline{WR} パルス ↓ 後のRDY/ \overline{BSY} ↓ 遅延時間	0		1	μs
t _{WLRH}	書き込み時間 (\overline{WR} ↓ からRDY/ \overline{BSY} ↑) (注1)	3.7		4.5	ms
t _{WLRH_CE}	チップ消去時間 (\overline{WR} ↓ からRDY/ \overline{BSY} ↑) (注2)	7.5		9	ms
t _{XLCL}	XTAL1パルス ↓ 後の \overline{OE} ↓ 待機時間	0			
t _{BVDV}	BS1有効からのDATA遅延時間	0		250	ns
t _{OLDV}	\overline{OE} ↓ 後のDATA出力遅延時間			250	ns
t _{OHDZ}	\overline{OE} ↑ 後のDATA Hi-Z遅延時間			250	ns

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。

30. 代表特性

30.1. ピンプルアップ

図30-1. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=5V)

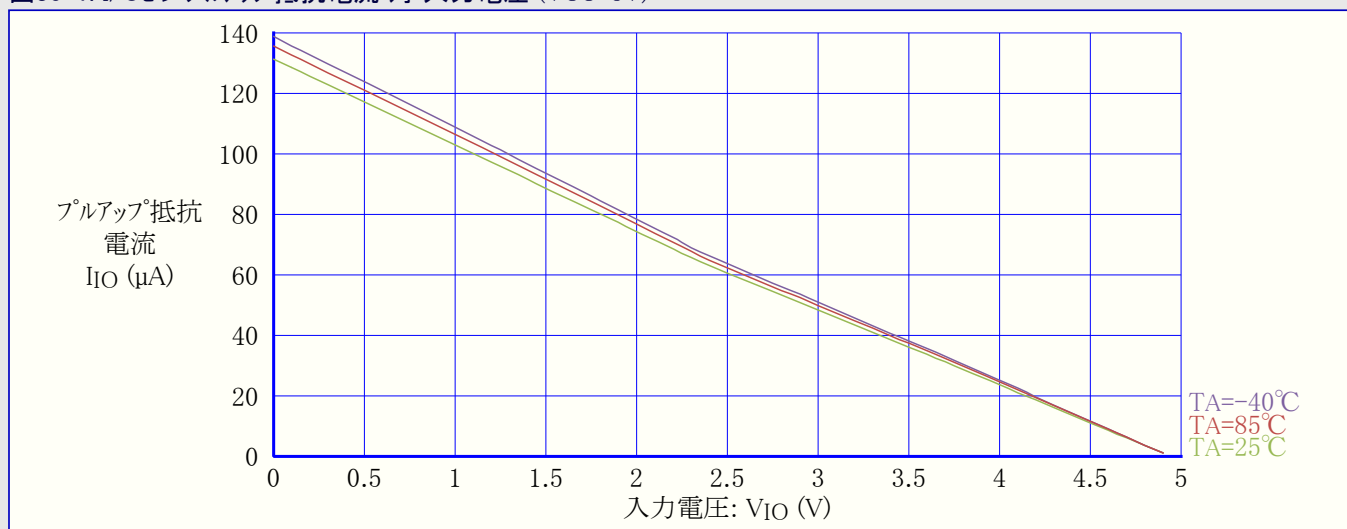
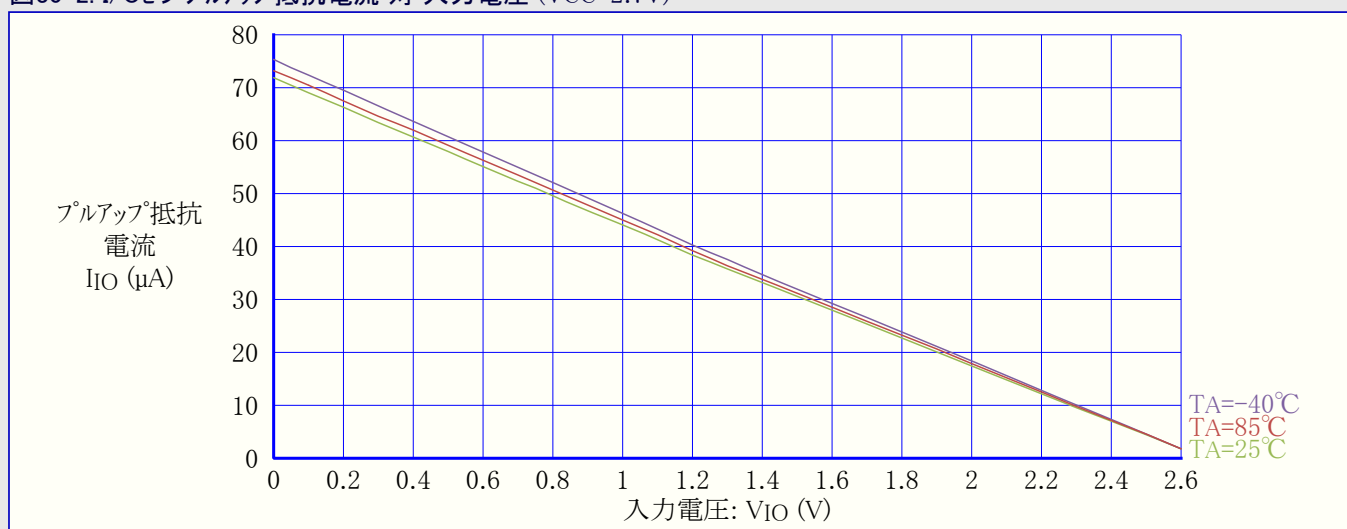


図30-2. I/Oピンプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)



30.2. ピン駆動部能力

図30-3. I/Oピン出力電圧 対 吸い込み電流 (VCC=5V)

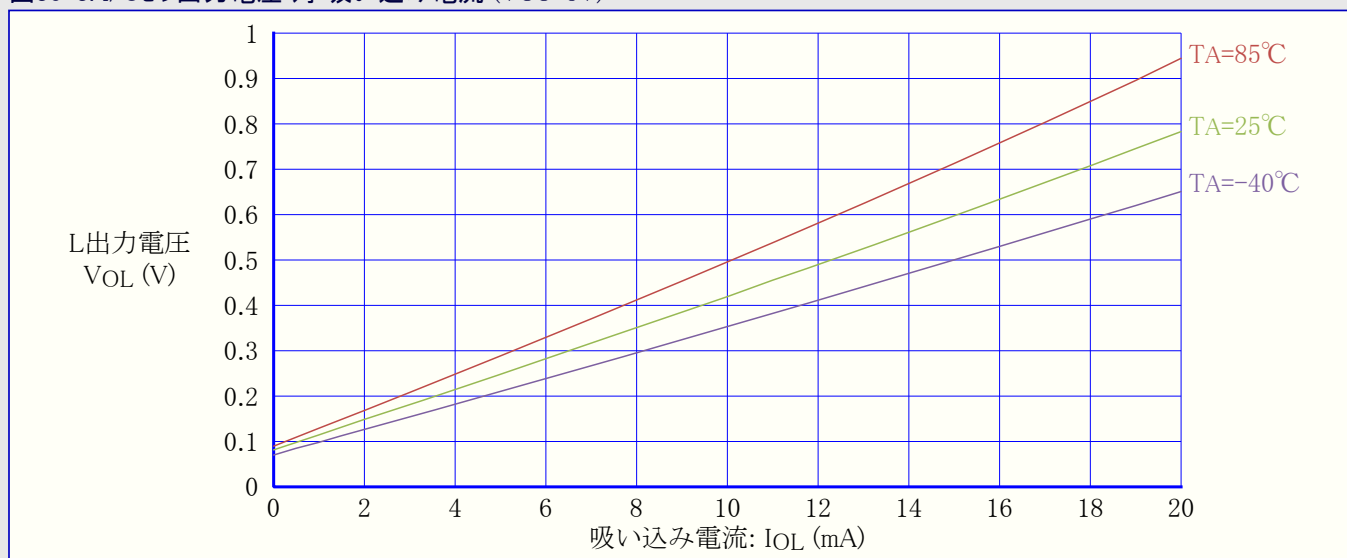


図30-4. I/Oピン出力電圧 対 吸い込み電流 (VCC=3V)

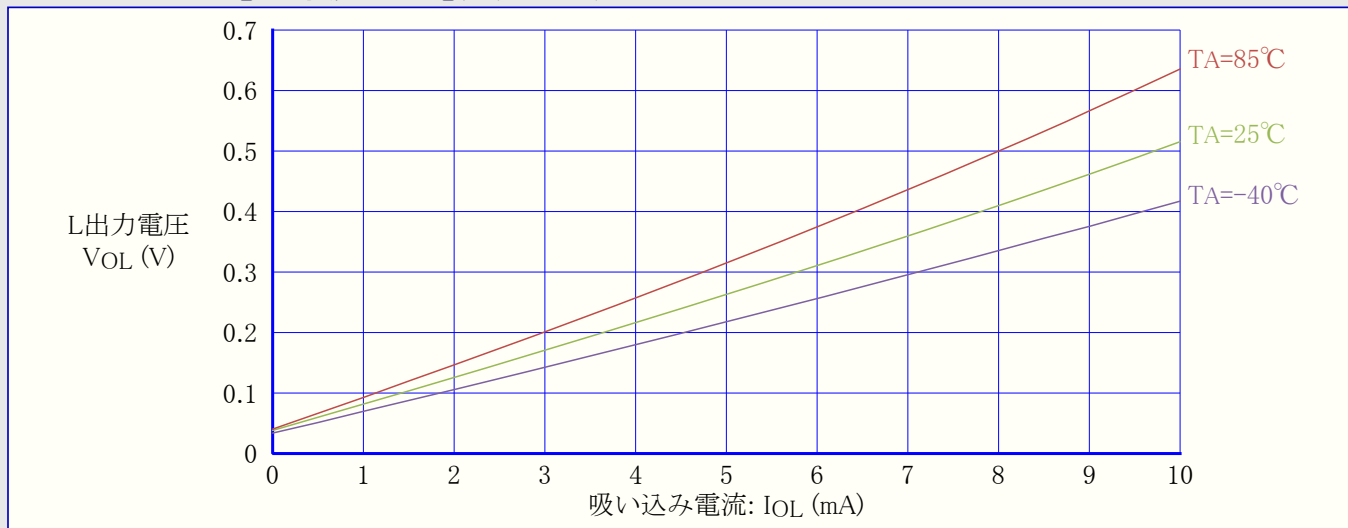


図30-5. I/Oピン出力電圧 対 吐き出し電流 (VCC=5V)

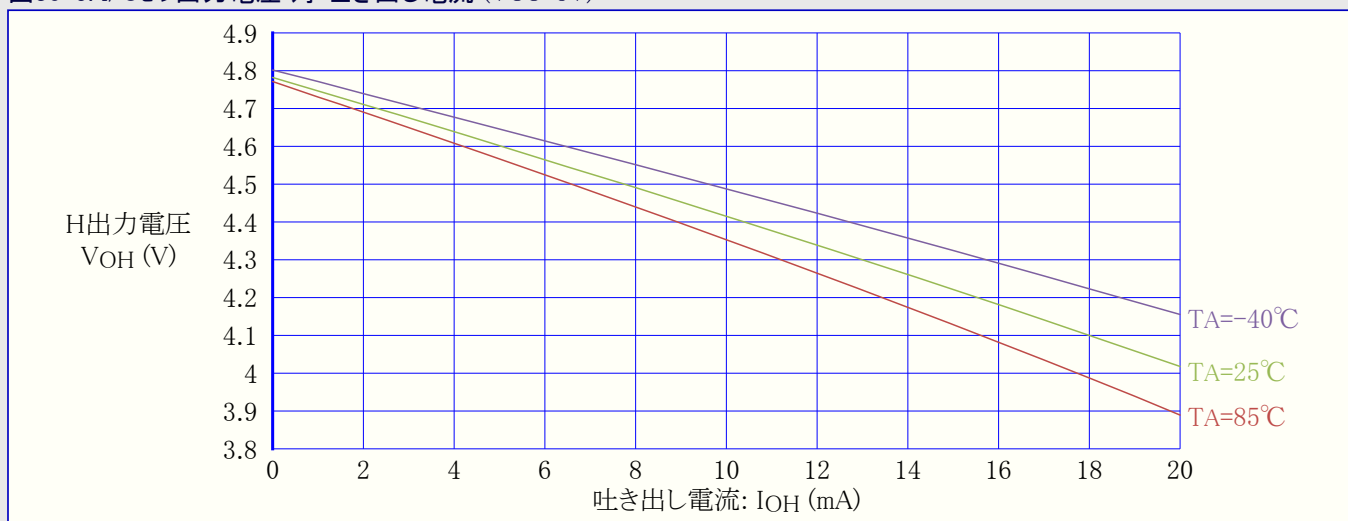
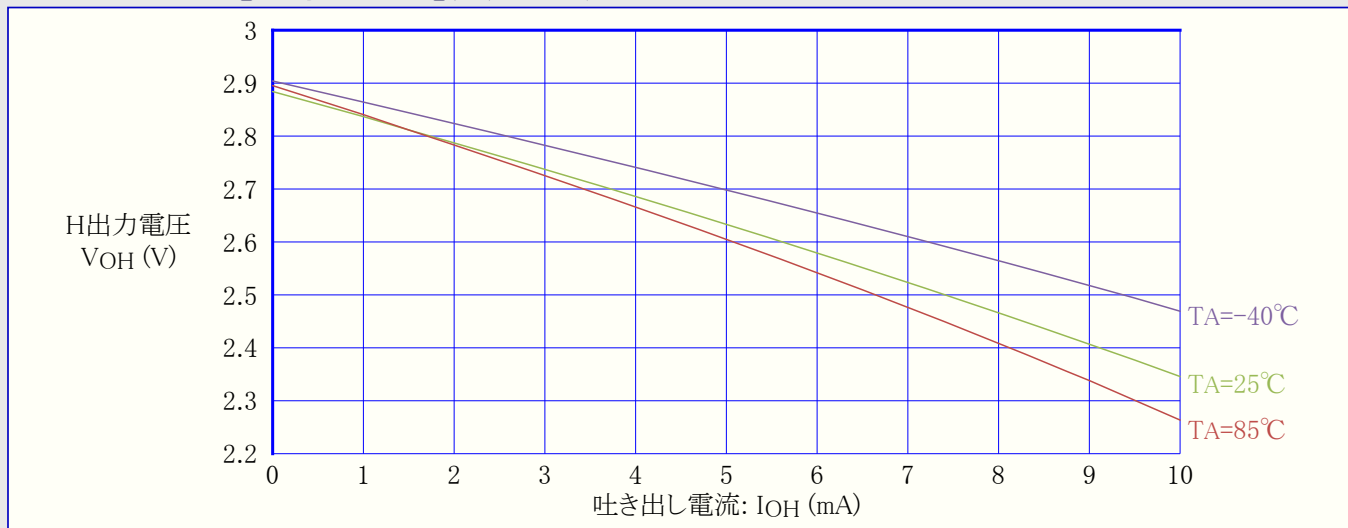


図30-6. I/Oピン出力電圧 対 吐き出し電流 (VCC=3V)



30.3. ピン 閾値とヒステリシス

図30-7. I/Oピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (V_{IH} , 1読み値)

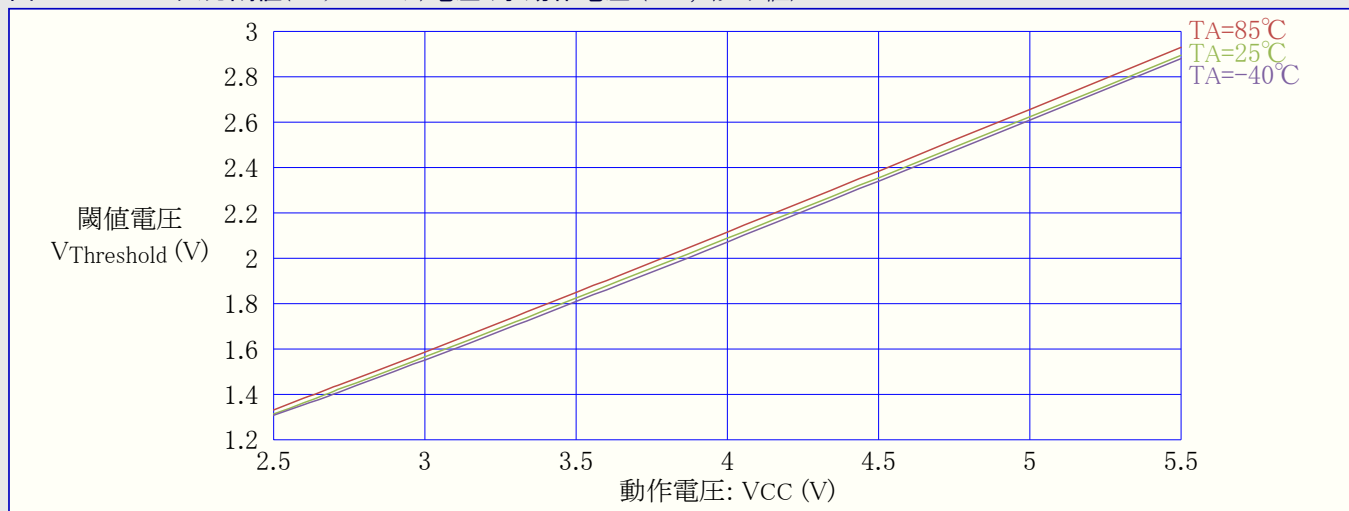


図30-8. I/Oピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (V_{IL} , 0読み値)

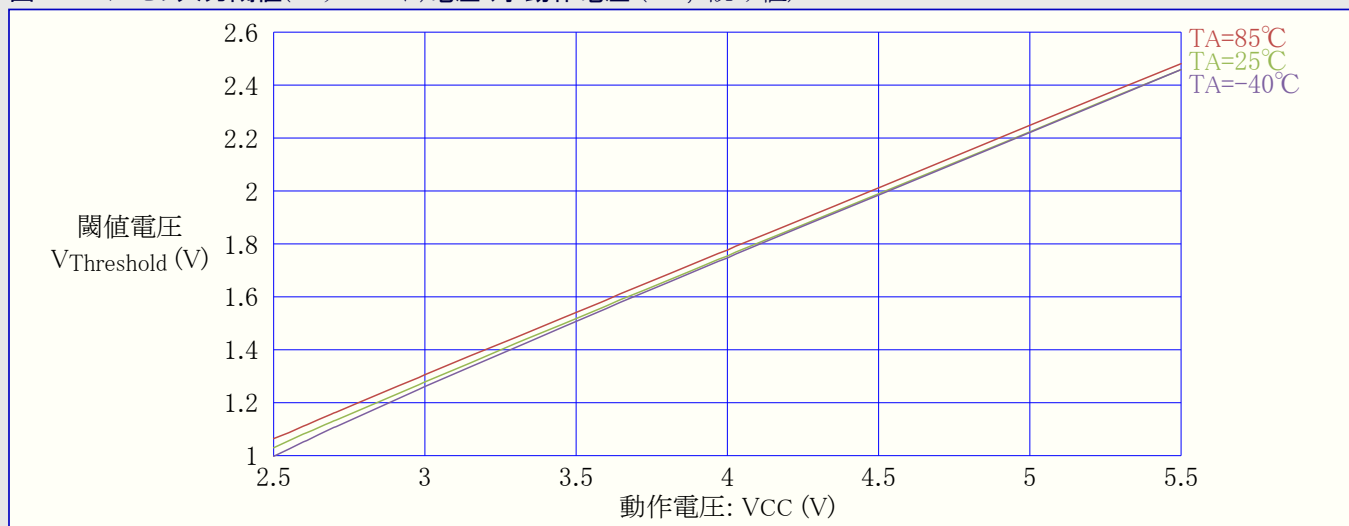


図30-9. I/Oピン入力ヒステリシス電圧 対 動作電圧

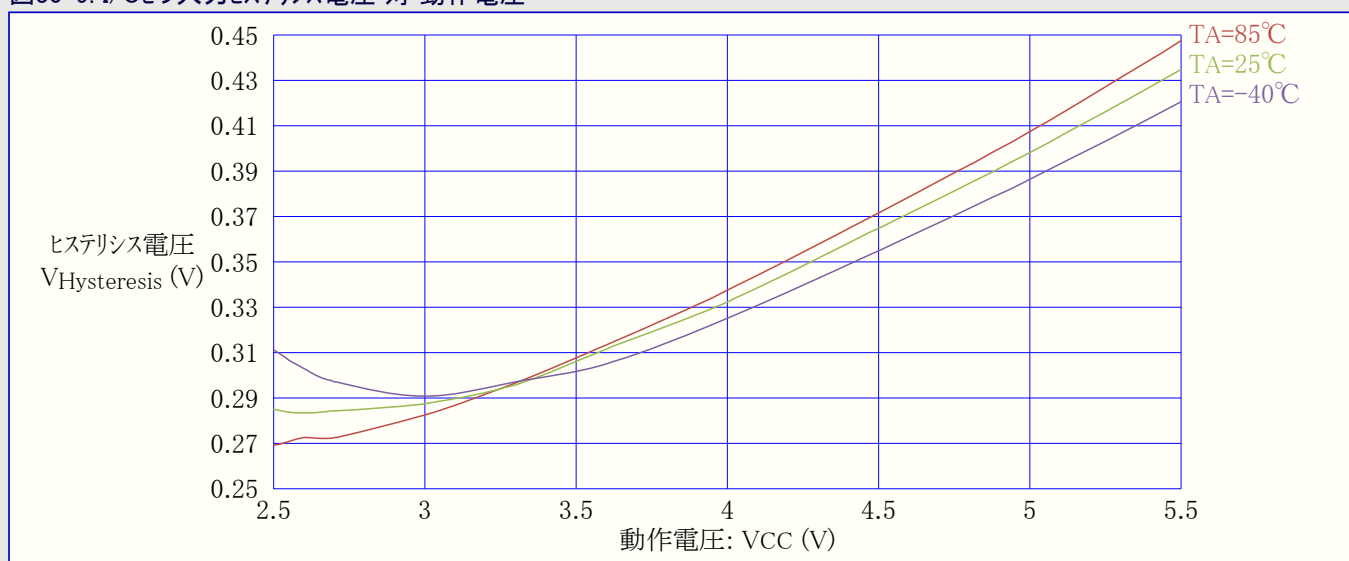


図30-10. RESET入力閾値(スレッショールド)電圧 対 動作電圧 (VIH,1読み値)

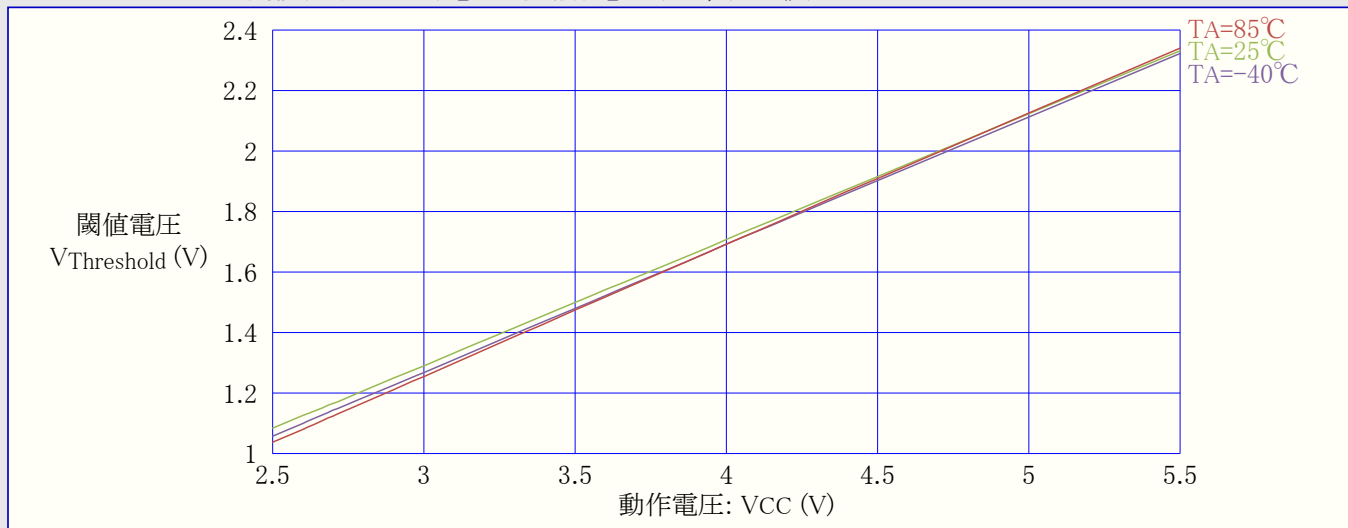


図30-11. RESET入力閾値(スレッショールド)電圧 対 動作電圧 (VIL,0読み値)

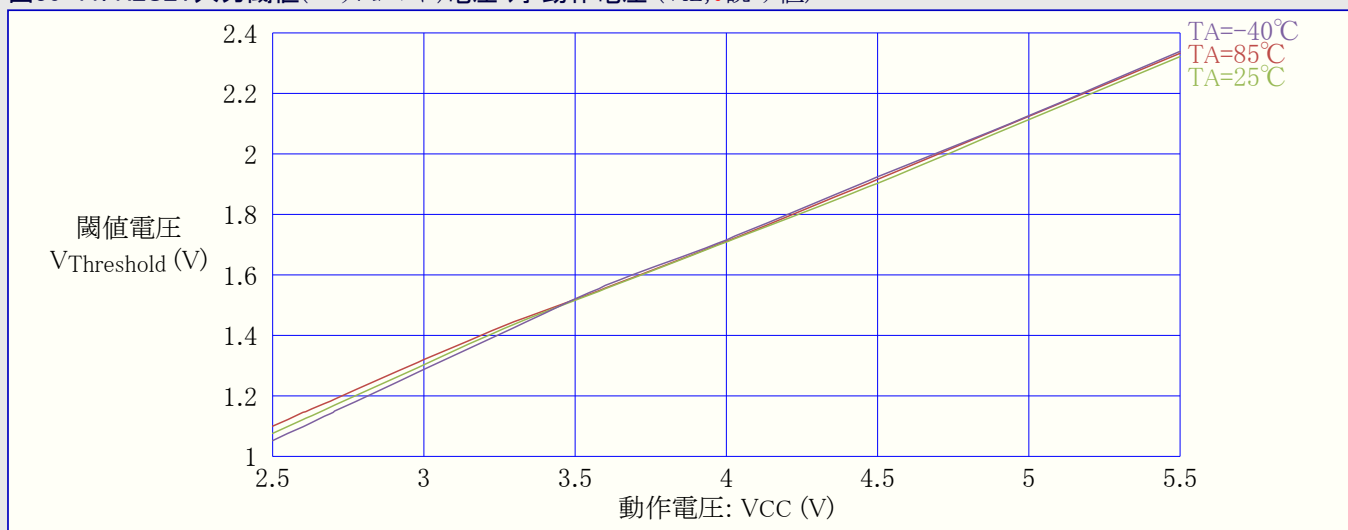


図30-12. RESETプルアップ抵抗電流 対 入力電圧 (VCC=5V)

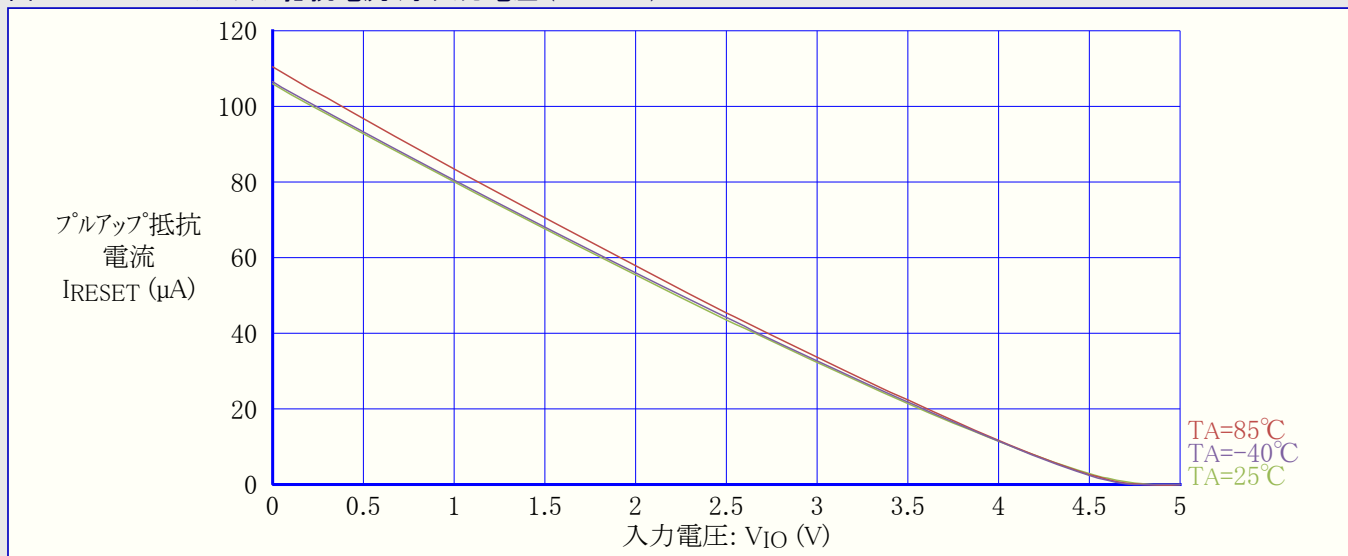


図30-13. RESETプルアップ抵抗電流 対 入力電圧 (VCC=2.7V)

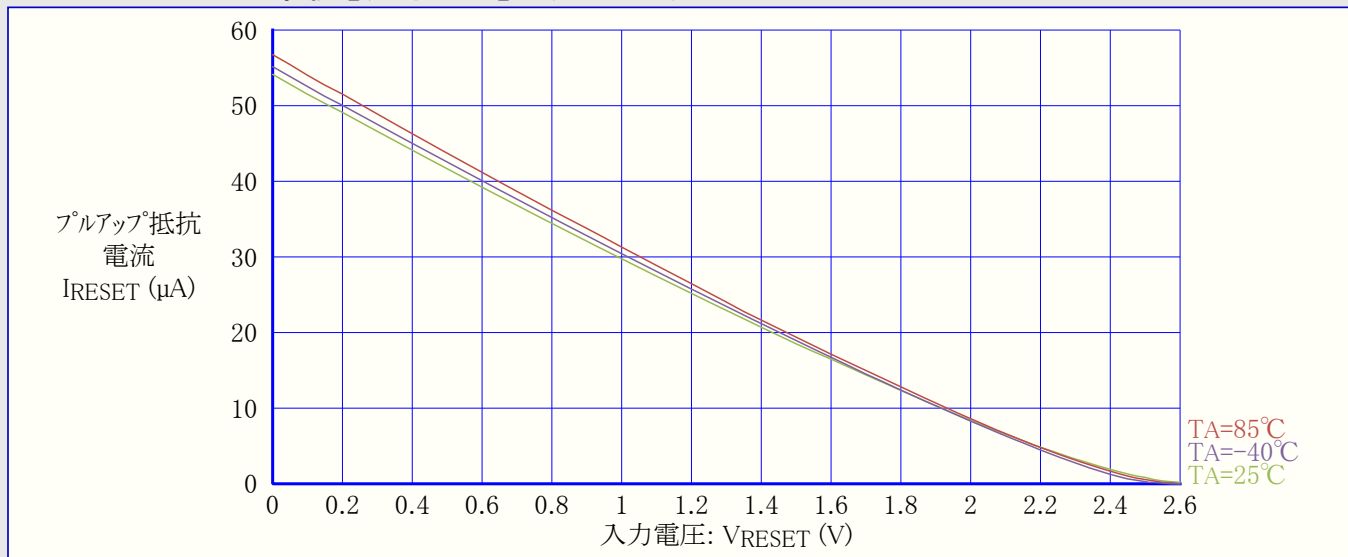


図30-14. XTAL1ピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (VIL,0読み値)

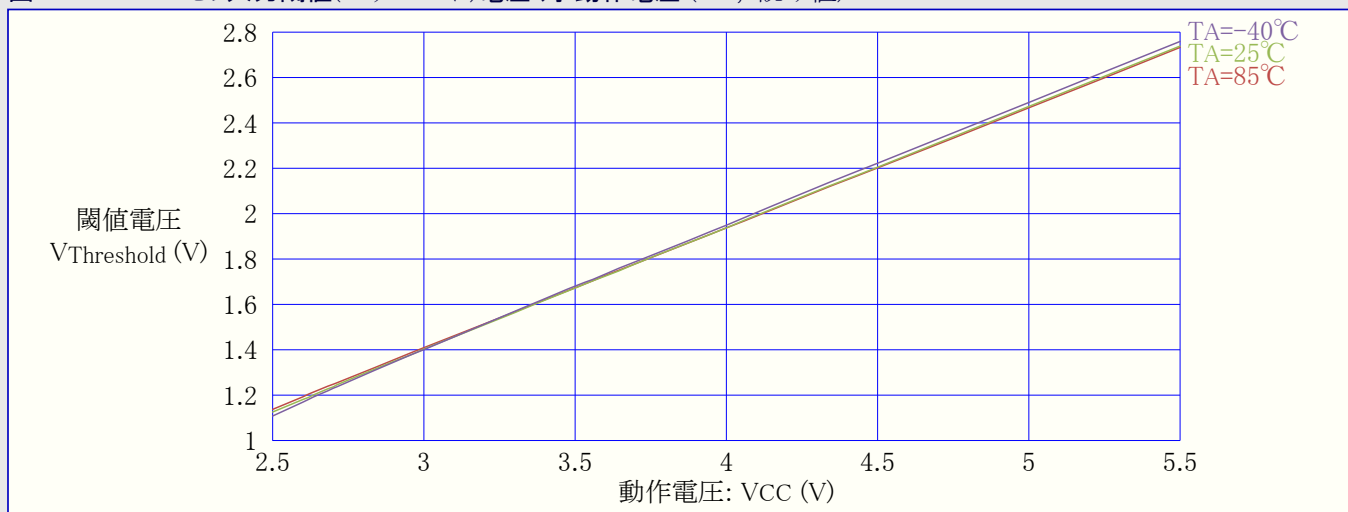
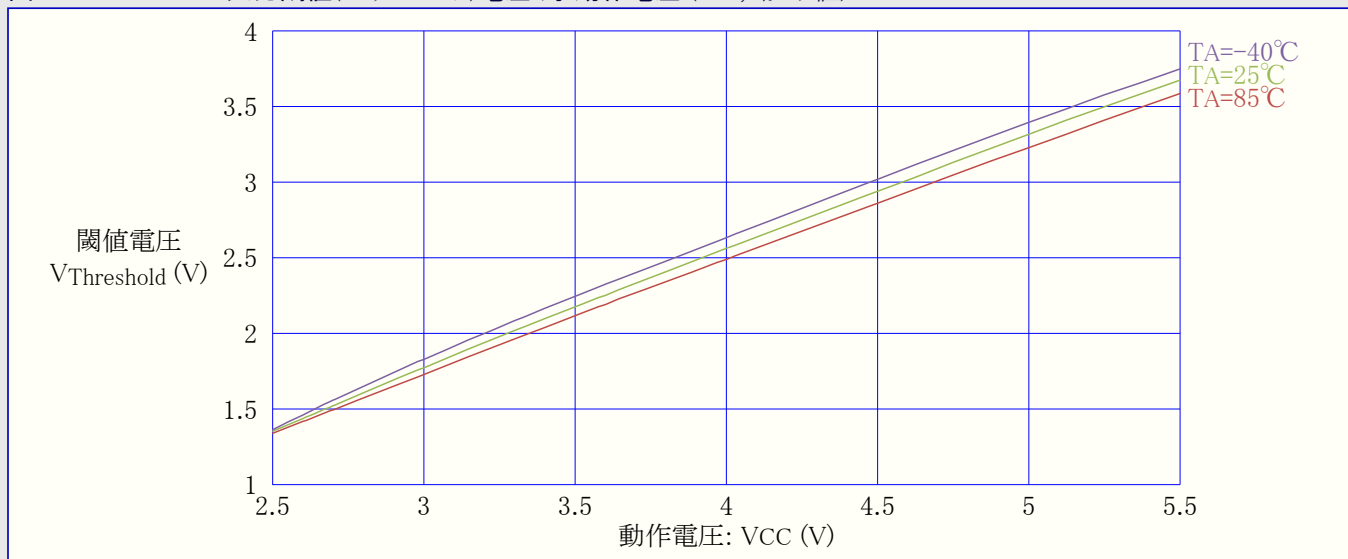
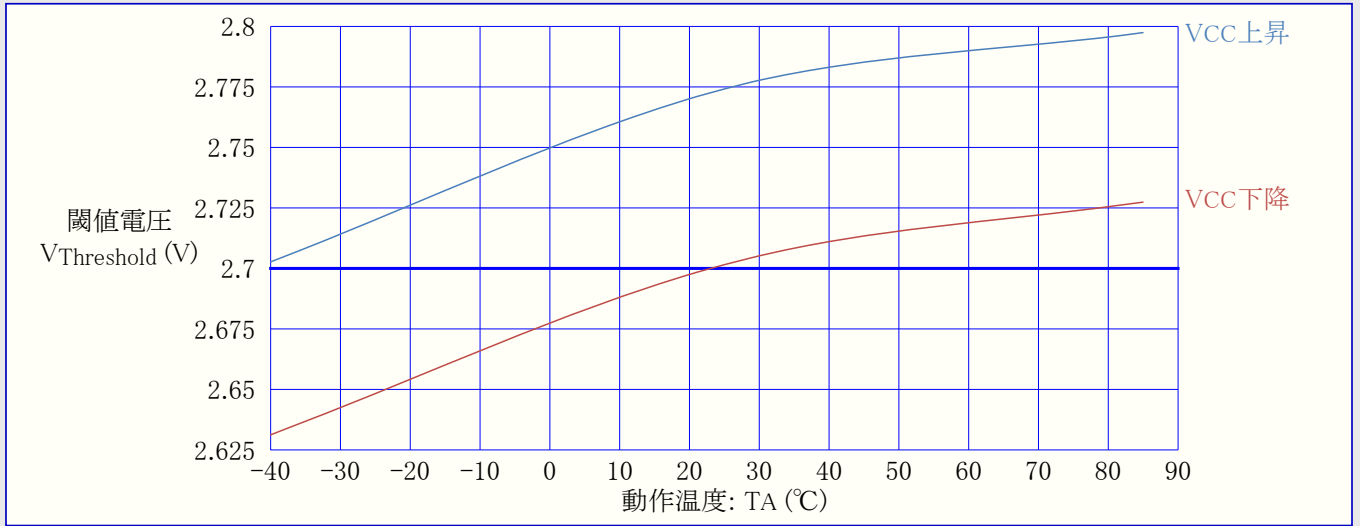


図30-15. XTAL1ピン入力閾値(スレッシュホールド)電圧 対 動作電圧 (VIH,1読み値)



30.4. 低電圧検出器(BOD)閾値とアナログ比較器ヒステリシス

図30-16. 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)



30.5. 内部発振器周波数

図30-17. ウォッチドッグ発振器周波数 対 動作電圧

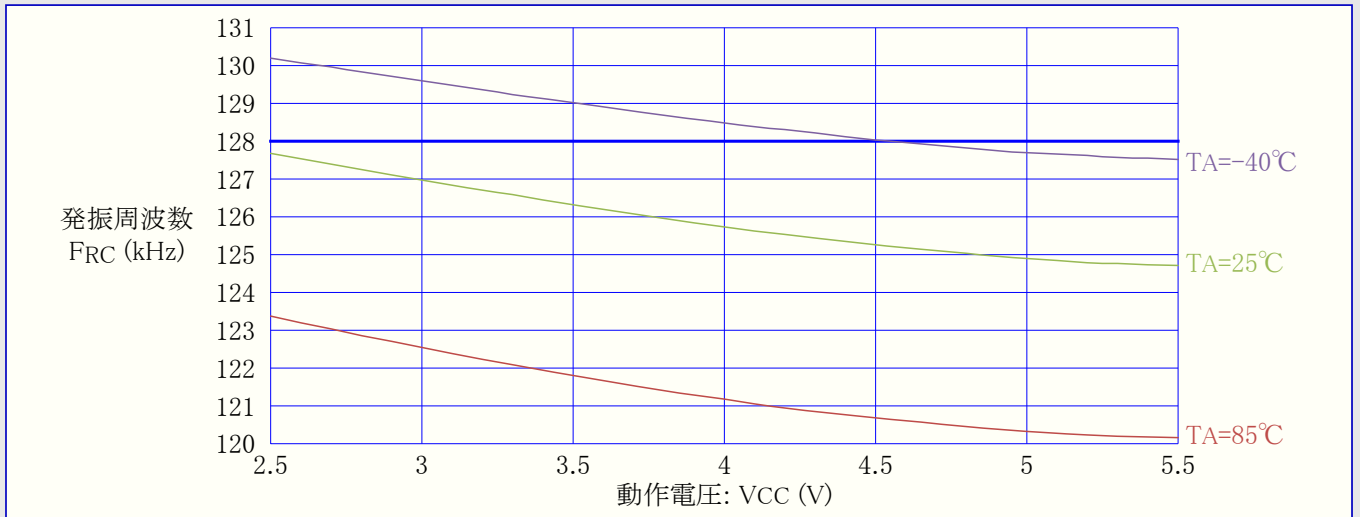


図30-18. ウォッチドッグ発振器周波数 対 動作温度

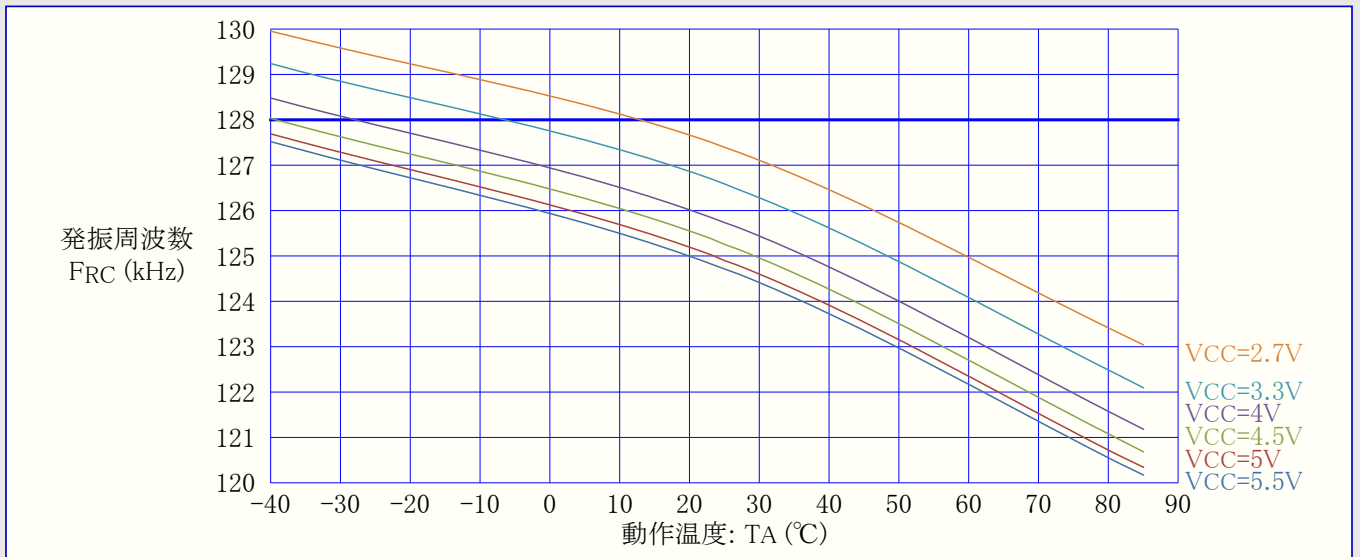


図30-19. 校正付き8MHz内蔵RC発振器周波数 対 動作電圧

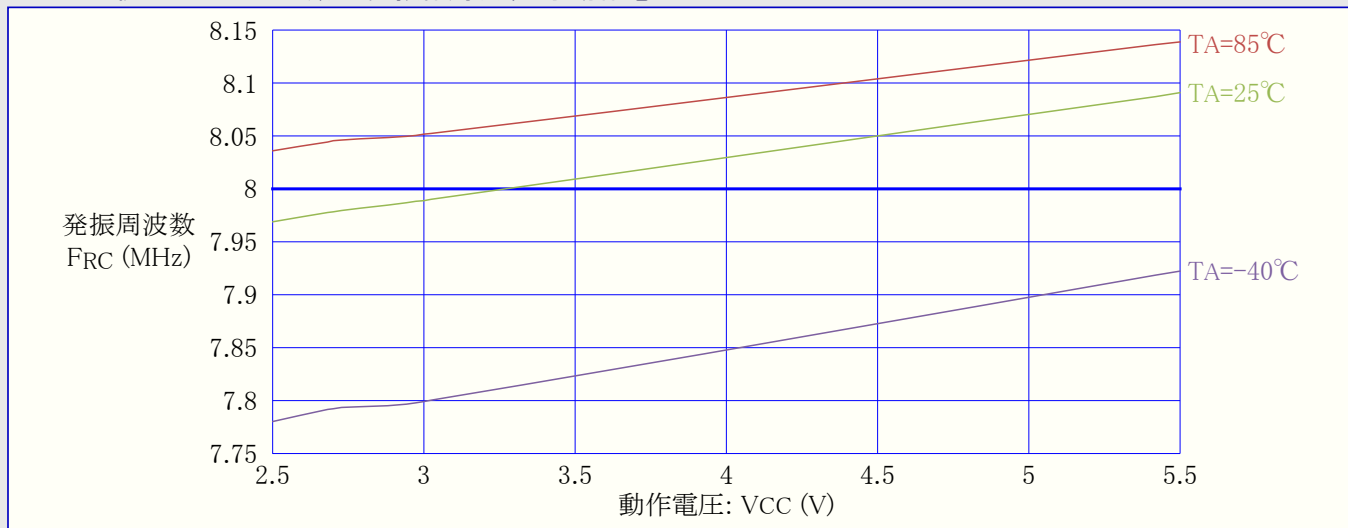
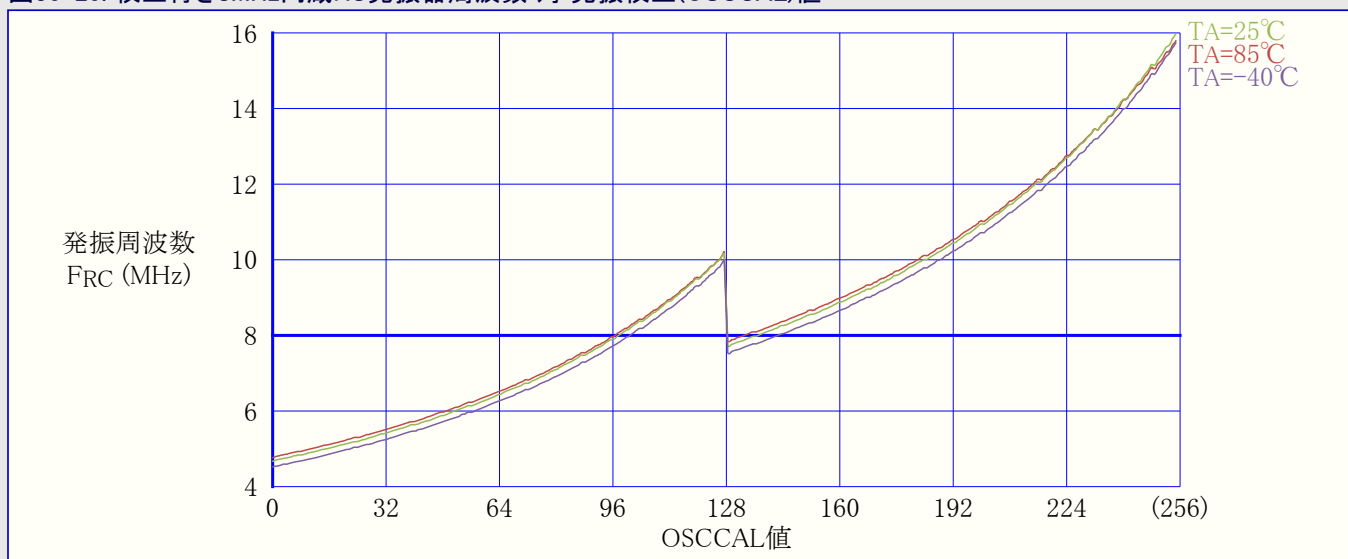


図30-20. 校正付き8MHz内蔵RC発振器周波数 対 発振校正(OSCCAL)値



30.6. 電力削減レジスタの使用

下の表と式(訳注:式は未記載)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。

表30-1. 活動動作とアイドル動作での部追加消費電流 (絶対値: μA)

PRR内ビット	VCC=5V, 16MHz	VCC=3V, 8MHz
PRCAN	13	12
PRPSC	8	7.5
PRTIM1	2	2
PRTIM0	1	1
PRSPI	2	2
PRLIN	5.5	5
PRADC	5	4.5

関連リンク [33頁の「PRR - 電力削減レジスタ」](#)

31. レジスタ要約

拡張1/Oレジスタ領域 (1/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$FF	予約								
\$FE	予約								
\$FD	予約								
\$FC	予約								
\$FB	予約								
\$FA	CANMSG								
\$F9	CANSTMH								
\$F8	CANSTML								
\$F7	CANIDM1								
\$F6	CANIDM2								
\$F5	CANIDM3								
\$F4	CANIDM4								
\$F3	CANIDT1								
\$F2	CANIDT2								
\$F1	CANIDT3								
\$F0	CANIDT4								
\$EF	CANCDMOB								
\$EE	CANSTMOB								
\$ED	CANPAJ								
\$EC	CANHPMOB								
\$EB	CANREC								
\$EA	CANTEC								
\$E9	CANTTCH								
\$E8	CANTTCL								
\$E7	CANTIMH								
\$E6	CANTIML								
\$E5	CANTCON								
\$E4	CANBT3								
\$E3	CANBT2								
\$E2	CANBT1								
\$E1	CANSIT1								
\$E0	CANSIT2								
\$DF	CANIE1								
\$DE	CANIE2								
\$DD	CANEN1								
\$DC	CANEN2								
\$DB	CANGIE								
\$DA	CANGIT								
\$D9	CANGSTA								
\$D8	CANGCON								
\$D7	予約								
\$D6	予約								
\$D5	予約								
\$D4	予約								
\$D3	予約								
\$D2	LINDAT								
\$D1	LINSEL								
\$D0	LINDR								
\$CF	LINDLR								
\$CE	LINBRRH								
\$CD	LINBRRL								
\$CC	LINBTR								
\$CB	LINERR								
\$CA	LINENIR								
\$C9	LINSIR								
\$C8	LINCR								
\$C7	予約								
\$C6	予約								
\$C5	予約								
\$C4	予約								
\$C3	予約								
\$C2	予約								
\$C1	予約								
\$C0	予約								

拡張1/レジスタ領域 (2/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$BF	予約								
\$BE	予約								
\$BD	予約								
\$BC	PIFR	-	-	-	-	PEV2	PEV1	PEV0	PEOP
\$BB	PIM	-	-	-	-	PEVE2	PEVE1	PEVE0	PEOPE
\$BA	PMIC2	POVEN2	PISEL2	PELEV2	PFLTE2	PAOC2		PRFM22~0	
\$B9	PMIC1	POVEN1	PISEL1	PELEV1	PFLTE1	PAOC1		PRFM12~0	
\$B8	PMIC0	POVEN0	PISEL0	PELEV0	PFLTE0	PAOC0		PRFM02~0	
\$B7	PCTL	PPRE1,0		PCLKSEL	-	-	-	PCCYC	PRUN
\$B6	POC	-	-	POEN2B	POEN2A	POEN1B	POEN1A	POEN0B	POEN0A
\$B5	PCNF	-	-	PULOCK	PMODE	POPB	POPA	-	-
\$B4	PSYNC	-	-	PSYNC21,0		PSYNC11,0		PSYNC01,0	
\$B3	POCR_RBH	-	-	-	-	POCR_RB11~8 (電力段制御器 比較RBレジスタ 上位)			
\$B2	POCR_RBL					POCR_RB7~0 (電力段制御器 比較RBレジスタ 下位バイト)			
\$B1	POCR2SBH	-	-	-	-	POCR2SB11~8 (電力段制御器2 比較SBレジスタ 上位)			
\$B0	POCR2SBL					POCR2SB7~0 (電力段制御器2 比較SBレジスタ 下位バイト)			
\$AF	POCR2RAH	-	-	-	-	POCR2RA11~8 (電力段制御器2 比較RAレジスタ 上位)			
\$AE	POCR2RAL					POCR2RA7~0 (電力段制御器2 比較RAレジスタ 下位バイト)			
\$AD	POCR2SAH	-	-	-	-	POCR2SA11~8 (電力段制御器2 比較SAレジスタ 上位)			
\$AC	POCR2SAL					POCR2SA7~0 (電力段制御器2 比較SAレジスタ 下位バイト)			
\$AB	POCR1SBH	-	-	-	-	POCR1SB11~8 (電力段制御器1 比較SBレジスタ 上位)			
\$AA	POCR1SBL					POCR1SB7~0 (電力段制御器1 比較SBレジスタ 下位バイト)			
\$A9	POCR1RAH	-	-	-	-	POCR1RA11~8 (電力段制御器1 比較RAレジスタ 上位)			
\$A8	POCR1RAL					POCR1RA7~0 (電力段制御器1 比較RAレジスタ 下位バイト)			
\$A7	POCR1SAH	-	-	-	-	POCR1SA11~8 (電力段制御器1 比較SAレジスタ 上位)			
\$A6	POCR1SAL					POCR1SA7~0 (電力段制御器1 比較SAレジスタ 下位バイト)			
\$A5	POCR0SBH	-	-	-	-	POCR0SB11~8 (電力段制御器0 比較SBレジスタ 上位)			
\$A4	POCR0SBL					POCR0SB7~0 (電力段制御器0 比較SBレジスタ 下位バイト)			
\$A3	POCR0RAH	-	-	-	-	POCR0RA11~8 (電力段制御器0 比較RAレジスタ 上位)			
\$A2	POCR0RAL					POCR0RA7~0 (電力段制御器0 比較RAレジスタ 下位バイト)			
\$A1	POCR0SAH	-	-	-	-	POCR0SA11~8 (電力段制御器0 比較SAレジスタ 上位)			
\$A0	POCR0SAL					POCR0SA7~0 (電力段制御器0 比較SAレジスタ 下位バイト)			
\$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	AC3CON	AC3EN	AC3IE	AC3IS1,0		-		AC3M2~0	
\$96	AC2CON	AC2EN	AC2IE	AC2IS1,0		-		AC2M2~0	
\$95	AC1CON	AC1EN	AC1IE	AC1IS1,0		AC1ICE		AC1M2~0	
\$94	AC0CON	AC0EN	AC0IE	AC0IS1,0		ACCKSEL		AC0M2~0	
\$93	予約								
\$92	DACH					DAC9,8またはDAC9~2 (D/Aデータレジスタ 上位バイト)			
\$91	DACL					DAC7~0またはDAC1,0 (D/Aデータレジスタ 下位バイト)			
\$90	DACON	DAATE		DATS2~0		-	DALA	DAOE	DAEN
\$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	

拡張1/OLレジスタ領域 (3/3)

アドレス	レジスタ略称	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
\$7F	DIDR1	-	AMP2PD	ACMP0D	AMP0PD	AMP0ND	ADC10D ACMP1D	ADC9D AMP1PD ACMP3D	ADC8D AMP1ND
\$7E	DIDR0	ADC7D	ADC6D AMP2ND ACMPN1D	ADC5D ACMPN0D	ADC4D	ADC3D ACMPN2D	ADC2D ACMP2D	ADC1D	ADC0D ACMPN3D
\$7D	予約								
\$7C	ADMUX	REFS1,0		ADLAR			MUX4~0		
\$7B	ADCSRB	ADHSM	ISRCEN	AREFEN	-		ADTS3~0		
\$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	AMP2CSR	AMP2EN	AMP2IS	AMP2G1,0		AMPCMP2	AMP2TS2~0		
\$76	AMP1CSR	AMP1EN	AMP1IS	AMP1G1,0		AMPCMP1	AMP1TS2~0		
\$75	AMP0CSR	AMP0EN	AMP0IS	AMP0G1,0		AMPCMP0	AMP0TS2~0		
\$74	予約								
\$73	予約								
\$72	予約								
\$71	予約								
\$70	予約								
\$6F	TIMSK1	-	-	ICIE	-	-	OCIEB	OCIEA	TOIE
\$6E	TIMSK0	-	-	-	-	-	OCIEB	OCIEA	TOIE
\$6D	PCMSK3	-	-	-	-	-	PCINT26	PCINT25	PCINT24
\$6C	PCMSK2	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
\$6B	PCMSK1	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
\$6A	PCMSK0	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
\$69	EICRA	ISC31,0		ISC21,0		ISC11,0		ISC01,0	
\$68	PCICR	-	-	-	-	PCIE3	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	-	PRCAN	PRPSC	PRTIM1	PRTIM0	PRSPI	PRLIN	PRADC
\$63	予約								
\$62	予約								
\$61	CLKPR	CLKPCE	-	-	-	CLKPS3~0			
\$60	WDTCSR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2~0		

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	-	-	-	(SP12)	(SP11)	SP10~8		
\$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	SPIPS	-	-	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	AC3IF	AC2IF	AC1IF	AC0IF	AC3O	AC2O	AC1O	AC0O
\$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)	予約								
\$4A (\$2A)	予約								
\$49 (\$29)	PLLCSR	-	-	-	-	-	PLLF	PLLE	PLOCK
\$48 (\$28)	OCR0B	OCR0B7~0 (タイマ/カウンタ0 比較Bレジスタ)							
\$47 (\$27)	OCR0A	OCR0A7~0 (タイマ/カウンタ0 比較Aレジスタ)							
\$46 (\$26)	TCNT0	TCNT07~0 (タイマ/カウンタ0)							
\$45 (\$25)	TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02~0		
\$44 (\$24)	TCCR0A	COM0A1,0		COM0B1,0		-	-	WGM01,0	
\$43 (\$23)	GTCCR	TSM	ICPSEL1	-	-	-	-	-	PSRSYNC
\$42 (\$22)	EEARH	-	-	-	-	-	(EEAR10)	(EEAR9)	EEAR8
\$21 (\$41)	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)	EEDR	-	-	EEP1,0		EERIE	EEMWE	EWE	EERE
\$3E (\$1E)	GPIOR0	GPIOR07~0 (汎用I/Oレジスタ0)							
\$3D (\$1D)	EIMSK	-	-	-	-	INT3	INT2	INT1	INT0
\$3C (\$1C)	EIFR	-	-	-	-	INTF3	INTF2	INTF1	INTF0
\$3B (\$1B)	PCIFR	-	-	-	-	PCIF3	PCIF2	PCIF1	PCIF0
\$3A (\$1A)	GPIOR2	GPIOR27~0 (汎用I/Oレジスタ2)							
\$39 (\$19)	GPIOR1	GPIOR17~0 (汎用I/Oレジスタ1)							
\$38 (\$18)	予約								
\$37 (\$17)	予約								
\$36 (\$16)	TIFR1	-	-	ICF	-	-	OCFB	OCFA	TOV
\$35 (\$15)	TIFR0	-	-	-	-	-	OCFB	OCFA	TOV
\$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)	PORTE	-	-	-	-	-	PORTE2	PORTE1	PORTE0
\$2D (\$0D)	DDRE	-	-	-	-	-	DDE2	DDE1	DDE0
\$2C (\$0C)	PINE	-	-	-	-	-	PINE2	PINE1	PINE0
\$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	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
\$27 (\$07)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
\$26 (\$06)	PINC	PINC7	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: () 付きのビットはATmega64M1で、() 付きのビットはATmega64M1とATmega32M1でのみ利用できます。

32. 命令要約

表32-1. 算術論理演算命令

ニーモニック	オペランド	意味	動作	フラグ*	クロック数
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

表32-2. 分岐命令

ニーモニック	オペランド	意味	動作	フラグ*	クロック数
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	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	3/4(注1)
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	3/4(注1)
CALL	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	4/5(注1)
RET		サブルーチンからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4/5(注1)
RETI		割り込みからの復帰	$PC \leftarrow STACK$	I,T,H,S,V,N,Z,C	4/5(注1)
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
SBRB	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

表32-3. データ移動命令

ニーモニック	オペラント	意味	動作	フラグ	クロック数
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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(注1,2)
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	(注4)
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

表32-4. ビット操作命令

ニーモニック	オペラント	意味	動作	フラグ	クロック数
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,N,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,N,Z,C	1
BCLR	s	ステータスレジスタのビット解除(0)	$SREG(s) \leftarrow 0$	0,0,0,0,0,0,0,0	1
BST	Rr,b	汎用レジスタのビットを一時フラグへ移動	$T \leftarrow Rr(b)$	I,T,H,S,V,N,Z,C	1
BLD	Rd,b	一時フラグを汎用レジスタのビットへ移動	$Rd(b) \leftarrow T$	I,T,H,S,V,N,Z,C	1
SEC		キャリーフラグを設定(1)	$C \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLC		キャリーフラグを解除(0)	$C \leftarrow 0$	I,T,H,S,V,N,Z,0	1
SEN		負フラグを設定(1)	$N \leftarrow 1$	I,T,H,S,V,1,Z,C	1
CLN		負フラグを解除(0)	$N \leftarrow 0$	I,T,H,S,V,0,Z,C	1
SEZ		ゼロフラグを設定(1)	$Z \leftarrow 1$	I,T,H,S,V,N,1,C	1
CLZ		ゼロフラグを解除(0)	$Z \leftarrow 0$	I,T,H,S,V,N,0,C	1
SEI		全割り込み許可	$I \leftarrow 1$	I,T,H,S,V,N,Z,C	1
CLI		全割り込み禁止	$I \leftarrow 0$	0,T,H,S,V,N,Z,C	1
SES		符号フラグを設定(1)	$S \leftarrow 1$	I,T,H,1,V,N,Z,C	1
CLS		符号フラグを解除(0)	$S \leftarrow 0$	I,T,H,0,V,N,Z,C	1
SEV		2の補数溢れフラグを設定(1)	$V \leftarrow 1$	I,T,H,S,1,N,Z,C	1
CLV		2の補数溢れフラグを解除(0)	$V \leftarrow 0$	I,T,H,S,0,N,Z,C	1
SET		一時フラグを設定(1)	$T \leftarrow 1$	I,1,H,S,V,N,Z,C	1
CLT		一時フラグを解除(0)	$T \leftarrow 0$	I,0,H,S,V,N,Z,C	1
SEH		ハーフキャリーフラグを設定(1)	$H \leftarrow 1$	I,T,H,1,S,V,N,Z,C	1
CLH		ハーフキャリーフラグを解除(0)	$H \leftarrow 0$	I,T,0,S,V,N,Z,C	1

表32-5. MCU制御命令

ニーモニック	オペラント	意味	動作	フラグ	クロック数
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	1

凡例

K6 : 6ビット定数 K : 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1: データ メモリ アクセスに対するクロック数は内部SRAMアクセスと仮定し、NVM制御部を通じたアクセスに対しては有効ではありません。(フラッシュメモリやEEPROMのように)NVM制御部を通してメモリをアクセスする時に最低1つの余分な周期が追加されなければなりません。これは他の主権部による同時アクセスやNVM制御部の状態に依存し、それらは1つよりも多くの余分な周期かもしれません。
(訳注:これは基本的に本デバイスには無関係な記述です。)

注2: 下位(64バイト)のI/O空間をアクセスする時に余分な1周期が追加されなければなりません。

注3: 命令は全てのデバイスで利用可能ではありません。(訳注:本書では使用不可命令は除外しているため、本注は無効です。)

注4: デバイス依存。デバイス固有のデータシートをご覧ください。(訳注:これがそのデータシートなので本注は無効です。)

33. 障害情報

本章の改訂版文字は各々Atmel ATmega16M1/32M1/64M1デバイスの改訂版を参照しています。

33.1. ATmega16M1

改訂A : 試供されていません。

33.2. ATmega32M1

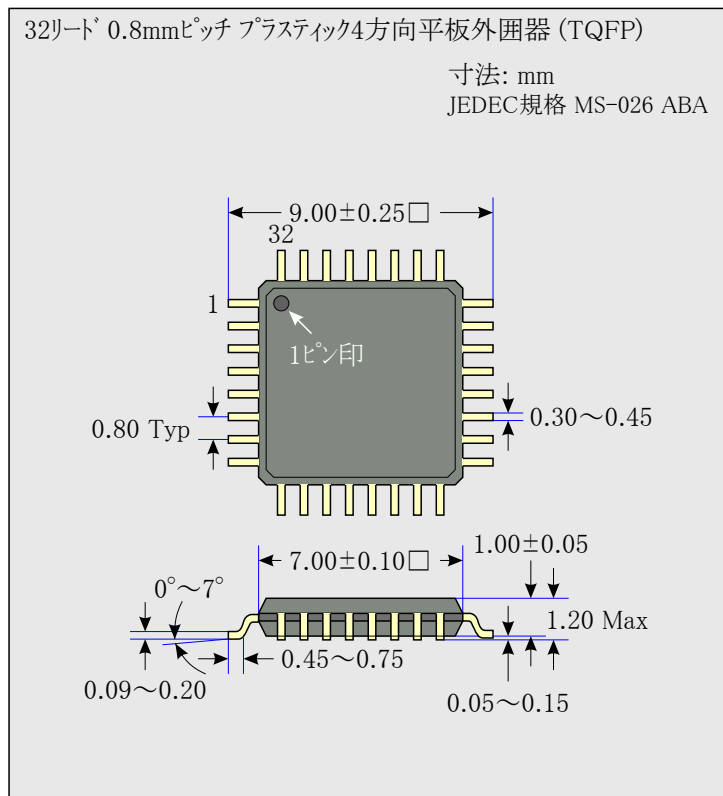
改訂A : 試供されていません。

33.3. ATmega64M1

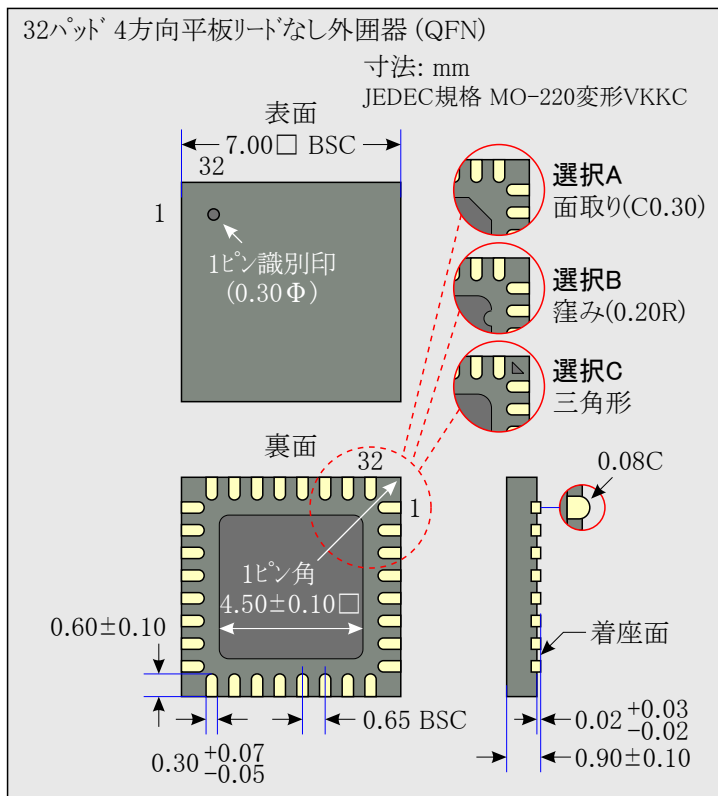
改訂A : 試供されていません。

34. 外圍器情報

34.1. 32A



34.2. PV



35. 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。

35.1. 8209A – 2009年8月

1. 初版

35.2. 8209B – 2009年10月

1. 180頁の「温度測定」を更新
2. 180頁の「製造時校正」を更新

35.3. 8209C – 2010年5月

1. 249頁でPV図で32M1-A外圍器情報図を置換
2. PV外圍器での情報修正で「注文情報」を更新

35.4. 8209D – 2010年11月

1. 1頁の「特徴」で脚注を削除
2. データシートから「お断り」項を削除
3. ヒューズビット読み込みに対する参照修正で225頁の表28-18を更新
4. リンク修正で226頁の「SPI直列プログラミング特性」を更新
5. 231頁の表29-8でRAINとCAIN(シングル エント'入力と作動入力の両方)に対する代表値を追加
6. 241頁の「レジスタ要約」で“PCICR”を追加
7. 編集上の更新
8. Atmelの新しい品質様式指針に従って最終頁を更新

35.5. 8209E – 2012年11月

1. 電気的特性追加 (訳注:代表特性の誤り)
2. Atmelの新ロゴで内容全体を更新
3. 内容編集上の更新

35.6. 8209F – 2016年10月

1. 内容編集上の更新 (訳注:内容も含めて新雛形適用)

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の英語版データシート ATmega16M1/32M1/64M1(Rev.8209F-10/2016)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。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。