

序説

著者: Teodor Lina Microchip Technology Inc.

AVR® EB系統のマイクロ コントローラは信号測定から事象同期と波形生成までの様々な応用を網羅するように構成された強力な計時器を使います。タイマ/カウンタE型(TCE)は4つまでのパルス幅変調(PWM:Pulse-Width Modulation)を生成でき、複数の波形生成動作を持つ16ビット タイマ/カウンタ周辺機能です。波形拡張(WEX:Waveform Extension)はTCEの能力を強化する波形拡張周辺機能です。

16ビット PWM TCEの能力は正確なプログラム実行タイミング、指令実行、周波数と波形の生成を含みます。TCEは基本計数器と比較チャネルの組から成ります。基本計数器はクロック周期と事象を計数するのに使われ、クロック周期を計数するための事象を許します。比較チャネルは比較一致制御、周波数生成とパルス幅変調を実行するために基本計数器と共に使うことができます。

WEXは16ビット PWM TCEの能力を強化します。この周辺機能は補完波形出力信号を作ってそれらの信号間に沈黙時間を生成し、障害事象を生成して管理し、ポート出力を上書きします。鍵となる特徴は電動機制御のような応用に計時器を使う時に共通する問題をWEXが克服することです。追加される沈黙時間で信号は重複せず、トランジスタの切り替えによって起こる貫通を防ぎます。WEXは安全性が重要な応用での重要な機能である障害状態をも作ることができ、全ての信号を使用者定義状態に駆動します。WEXからの全ての機能はハードウェア内部で起き、全ての状況で予測可能な生成波形にします。

この技術概説はTCEとWEXに対する様々な動作形態で読者を習熟させ、この計時器の特殊性を強調します。機能のより良い理解のためデータシートを参照してください。

この文書の構成は以下の3つの特別な使用事例を網羅します。

- **WEXのTCE独立性を使う4つのPWM信号生成**

異なるデューティ サイクルと尺度調整任意選択で4つのPWM信号を生成するようにTCEを2傾斜PWM動作で初期化します。

- **WEXを使う8つの信号のための模様型生成**

周期的に交互する各出力ピン用に継続する模様を生成するようにWEXを構成設定します。

- **TCEとWEXを使って障害検出を持つ8つの補完波形出力信号を生成**

単一傾斜動作でTCEを初期化して4つのPWM信号を生成します。4つのPWM信号を沈黙時間を持つ8つの補完信号に分割するようにWEXを構成設定します。各PWM信号は0~100%間で異なるデューティ サイクルを持ちます。各比較レジスタの範囲内で、割り込み処理ルーチン(ISR:Interrupt Service Routines)に於いて割り込みデューティ サイクルの増加が起きます。障害事象が250µs毎に起動され、全ての出力を論理Low '0' に駆動します。その後、障害事象が解決され、通常の動作が再開します。その後、この処理を繰り返します。

注: この文書で記述した各使用事例に対して、AVR16EB32で開発した素の物とAVR16EB32で開発したMPLAB®コード構成部(MCC)で生成した物の2つのコード例があります。

この技術概説で記述したのと同じ機能性を持つAVR16EB32用素のコード例は以下のここで見つけることができます。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

この技術概説で記述したのと同じ機能性を持つAVR16EB32用MCC Melody生成コード例は以下のここで見つけることができます。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

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、Microchip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。

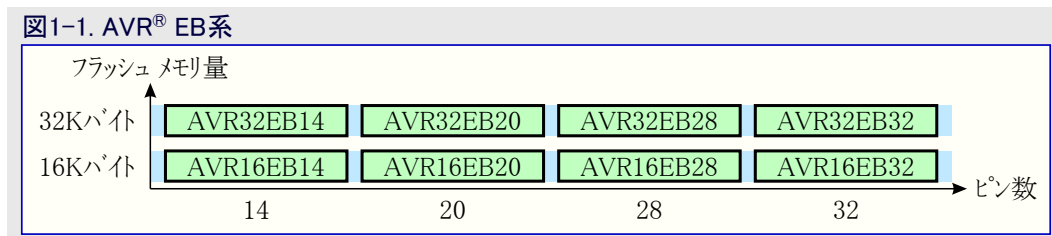
目次

序説	1
1. 関連デバイス	3
2. 概要	4
3. TCEを使うPWM信号生成	6
3.1. 素の実装	6
3.2. MCC Melody実装	11
3.3. 結果	12
4. WEXを使う模様型生成	15
4.1. 素の実装	15
4.2. MCC Melody実装	17
4.3. 結果	19
5. TCEとWEXでの8つのPWM信号生成	20
5.1. 素の実装	20
5.2. MCC Melody実装	26
5.3. 結果	28
6. 参照	30
7. 改訂履歴	31
Microchip情報	32
Microchipウェブサイト	32
製品変更通知サービス	32
お客様支援	32
Microchipデバイス コード保護機能	32
法的通知	32
商標	33
品質管理システム	33
世界的な販売とサービス	34

1. 関連デバイス

本章はこの文書に関連するデバイスを一覧にします。下図はピン数の変種とメモリ量を展開して各種系統デバイスを示します。

- これらのデバイスがピン互換で同じまたはより多くの機能を提供するため、垂直方向移植はコード変更なしで可能です。AVR EB系デバイスでの下方移植はいくつかの周辺機能でのより少ない利用可能実体のためにコード変更を必要とするかもしれません。
- 左への水平方向移植はピン数と利用可能な機能を減らします。
- 異なるフラッシュメモリ量を持つデバイスは一般的に異なるSRAMとEEPROMを持ちます。

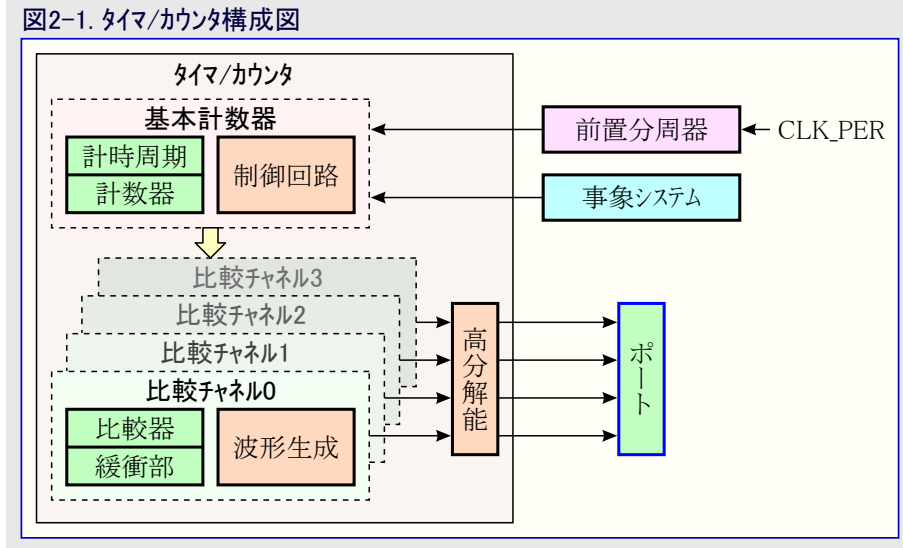


2. 概要

柔軟な16ビットTCEは正確なプログラム実行タイミング、周波数と波形生成だけでなく、指令実行も提供します。

TCE実体は基本計数器と4つの比較チャネルから成ります。クロック刻み(計時器)または各種事象(計数器)に基づき、基本計数器を上昇方向または下降方向の計数に設定することができます。タイマ/カウンタは任意選択前置分周を持つ周辺機能クロックまたは事象システム(EVSY)からクロック駆動/計時することができます。EVSYは方向制御または動作同期に使うこともできます。

計数器はデューティサイクル分解能を入力クロックの8倍に増すことができる高分解能任意選択を含みます。加えて、前置分周された周辺機能クロックとEVSYからの事象は計数器を制御するのに使うことができます。



計数器値は計数が各々、BOTTOMまたはTOPに達した時を決めるのに、0と周期(PER)値と継続的に比較されます。条件の1つが合致すると、計数器は更新され、割り込みが生成されます。計数器は比較(CMPn)レジスタとも比較されます。計数器は割り込み生成と、波形生成動作が選ばれた場合に波形とパルス幅の周期を設定するのにこれらの比較を使います。

計数器、周期、比較のレジスタとそれらの全ての緩衝部は16ビット幅です。緩衝部は計数器(CNT)レジスタが更新される時にだけ各々のレジスタが更新されるのを保証する仕組みの一部です。各緩衝部は各々のレジスタが更新を必要としているかを決めるのに使われる緩衝有効(BV: Buffer Valid)ビットを持ちます。

TCEは事象信号の上昇端や下降端を計数するのに、またはクロック刻み計数を許可するのに、EVSYを使うように構成設定することができます。

TCEは尺度許可(SCALE)任意選択が許可された場合に、比較レジスタに0~2の分数値を書く任意選択を提供します。比較値は周期、振幅、変位の値に基づいて計算されます。

波形生成拡張概要

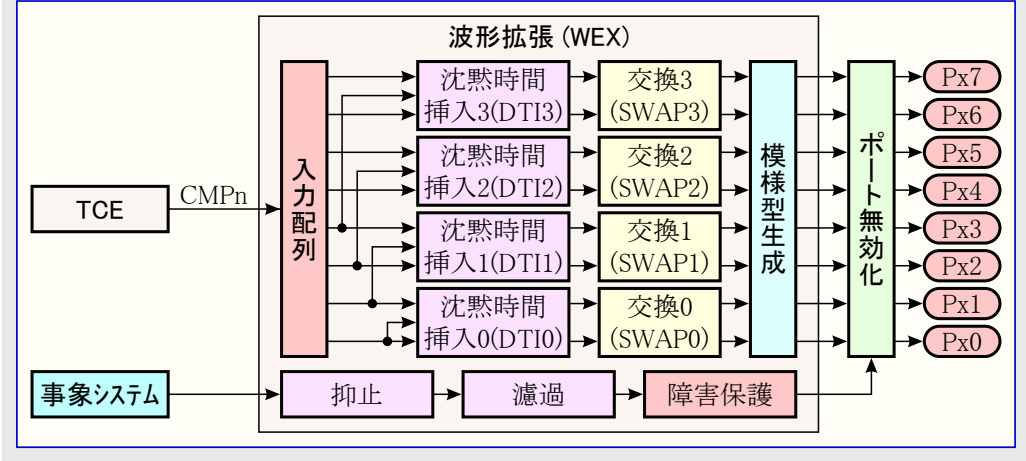
WEXは波形生成(WG: Waveform Generation)動作でのタイマ/カウンタに沈黙時間を持つ補完信号生成のような付加機能を提供します。これらの機能は電動機制御と電力制御の応用に役立ちます。

WEX実体は以下の5つの単位部から構成されます。

- 入力配列
- 沈黙時間挿入部
- 交換部
- 模様型生成部
- 障害保護部

入力配列部使用時に波形出力を違う構成設定でタイマ/カウンタからポートピンに配線することができます。沈黙時間挿入部は計時器の4つの出力を(各々)Low側とHigh側の2つの非重複信号に分割することができます。WEXは計時器の4つのPWM出力から8つの出力信号を作ることができます。Low側とHigh側のピン位置は交換部を使って交換することができ、これは電動機高速減衰用の重要な機能です。模様型生成部は一定の論理値でポートピンを上書きすることができ、タイマ/カウンタから独立して使えます。最大8ピンに対して模様を構成設定することができます。障害保護部は障害事象検出時にWEX出力を定義した状態に設定することができます。障害保護部はEVSYに接続され、事象に障害の起動を許します。

図2-2. 波形拡張構成図



3. TCEを使うPWM信号生成

TCAやTCBのような計時器と比べて時にTCEの鍵となる特性はPWM生成の多用途性と精度です。応用の複雑性に従って様々な構成設定から選ぶことができます。TCEは単一傾斜と2傾斜のPWM生成動作の両方で構成設定することができ、これは定位相、正しい位相のPWM、より高い最大動作周波数の間での相反関係を許します。また、高速PWMでTCEは不具合なしのPWMを保証する緩衝の仕組みを持ちます。TCEの別の機能は振幅と変位を使ってPWM信号のデューティサイクルを尺度調整する能力です。

TCEとTCBは共に高い最大動作周波数でのPWM信号生成に使うことができます。選択可能な方向に基づく2傾斜PWM能力のため、TCEだけが重要な応用で使うことができます。2傾斜PWMはデューティサイクルが変更される時にパルス中央位置を変えません。従って、位相は常に一定です。この特徴はそれが複数PWM信号の同時整流(転流)によって引き起こされる切り替え雑音を避けるため、電動機制御応用に関して重要です。この雑音はトランジスタ切り替えから来ます。異なるデューティサイクルでの2傾斜PWM信号の使用はこの雑音様式を防ぐことができます。

緩衝の仕組みは周期と各比較のレジスタに対する緩衝部を含みます。予期せぬ長いパルスが回路短絡を引き起こし得る重要な応用ではこれらの緩衝部を使うことが重要です。更に、緩衝部の存在は同じ計時器で異なる比較チャンネルを使う2つの周辺機能間での同期損失を防ぐことができます。けれども、周期と比較のレジスタを直接更新できると言う事実を考慮すると、緩衝の仕組みは使用者によって避けられ得ます。以下の波形は緩衝される動作と緩衝されない動作の違いを図解します。

周期レジスタを直接更新する(緩衝なし動作)の場合、計時器は新しい閾値を既に通過しているかもしれず、故に最大値への計数を続けるでしょう。それは更新の問題を引き起こし得る異常に長いパルスを引き起こすでしょう。また、2つ以上の比較チャンネルを使って1つが更新される場合、起動元間の同期が失われ得ます。これらの有り得る問題の全てを防ぐため、緩衝の仕組みの使用が必要とされます。緩衝部は新しい値を保持し、計時器が更新される時に従って、それを比較または周期のレジスタに転送します。全ての値の同時変更で、言及した問題は消滅します。

図3-1. 緩衝なし2傾斜動作

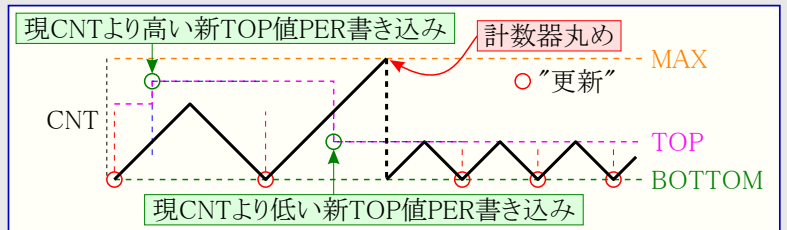
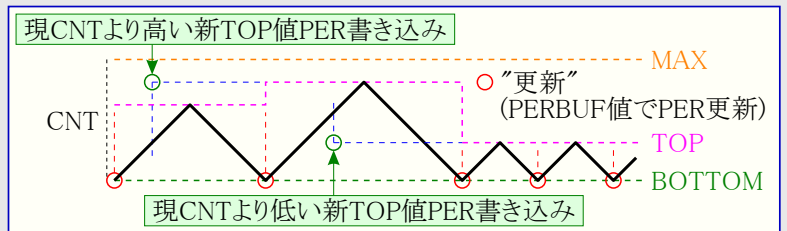


図3-2. 緩衝使用周期変更



下は上で言及した緩衝の仕組みを使って10kHzで20%、40%、60%、80%のデューティサイクルを持つ4つのPWM信号を生成するようにTCE実体を設定する方法の例です。比較レジスタで設定される値尺度調整機能と高分機能がこの例で強調されます。PWM信号分解能を3ビットまで増やすことができます。

この例では、デューティサイクルの最大範囲を0~100%から0~75%、0~50%、0~150%に変更することができます。デューティサイクルは周期値に基づいて計算されます。周期が常に100%でデューティサイクルの最大値です。100%を超える全てのデューティサイクルは周期の値である100%で飽和されます。周期の0~50%に対するデューティサイクルの範囲を選ぶと、20%、40%、60%、80%のデューティサイクル値を持つ4つのPWM信号に対して始めに生成されるデューティサイクルは今や10%、20%、30%、40%のデューティサイクルに尺度調整された値を持ちます。提案された例は走行時の10ms毎にTCEの比較レジスタの尺度調整値を変更します。デューティサイクルの範囲は尺度調整値に応じて変更され、次の2つの項が望む動きを行うためのTCE周辺機能の構成方法を詳述します。最初の項は素のコードを使ってTCEを構成設定する方法を説明します。後続する項はMCC Melodyを使ってTCEを構成設定する方法を詳述します。最後の項は結果を含みません。

3.1. 素の実装

1. TCE0の高分解能機能を使うのにCLKCTRL周辺機能が構成設定されなければなりません。次のレジスタ設定が必要とされます。

図3-3. MCLKCTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	CLKOUT			CLKSEL3~0				
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	0	0

- ビット3~0 - CLKSEL3~0 : クロック選択 (Clock Select)

このビット領域は主クロック(CLK_MAIN)用の供給元を選びます。

値	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	その他
名称	OSCHF	OSC32K	XOSC32K	EXTCLK	PLL	-
説明	内部高周波数発振器	32.768kHz内部発振器	XOSC32KCTRLAのSELビットに応じて32.768kHz外部クロックまたは32.768kHz外部クリスタル用発振器	外部クロック	PLL発振器	(予約)

図3-4. MCLKCTRLBレジスタ

ビット	7	6	5	4	3	2	1	0
			PBDIV	PDIV3~0				PEN
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	1	0	0	0	1

● ビット5 - PBDIV : 前知分周器B分周 (Prescaler B Division)

このビットが'1'の場合、CLK_PERとCLK_CPUに対するクロック周波数はCLK_PER4クロックに呼応して設定されます。前置分周器B分周(PBDIV)ビットと前置分周器許可(PEN)ビットが'1'の時はPDIVに対して2ⁿに一致する前置分周設定だけが利用可能です。

値	0	1
名称	NONE	DIV4
説明	CLK_PERとCLK_CPUに対して分周なし	CLK_PERとCLK_CPUに対して4分周

● ビット0 - PEN : 前置分周器許可 (Prescaler Enable)

このビットは主クロック(CLK_MAIN)前置分周器が許可されるかを制御します。

値	0	1
説明	CLK_MAIN前置分周器禁止	CLK_MAIN前置分周器許可、分周比は前置分周器分周値(PDIV)ビット領域によって制御されます。

図3-5. PLLCTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	RUNSTDBY	SOURCE1,0		SOURCEDIV1,0			MULFAC1,0	
アクセス種別	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット4,3 - SOURCEDIV1,0 : PLL用供給元分周選択 (Select Source Division for PLL)

このビット領域は供給元選択(SELHF)ビットが'0'の時に外部高周波数発振器(XOSCHF)の始動時間を制御します。

値	0 0	0 1	1 0	1 1
名称	NONE	DIV2	DIV4	DIV6
説明	分周なし、公称供給元周波数: 2.5~5.5MHz	2分周、公称供給元周波数: 5~11MHz	4分周、公称供給元周波数: 10~22MHz	6分周、公称供給元周波数: 15~33MHz

● ビット1,0 - MULFAC1,0 : 逓倍係数 (Multiplication Factor)

このビット領域は位相固定化閉路用逓倍係数を制御します。

値	0 0	0 1	1 0	1 1
名称	DISABLE	-	8X	16X
説明	PLL禁止	(予約)	8逓倍	16逓倍

2. ポート多重器のTCE対応レジスタ(TCEROUTEA)は単位部出力を違うポートに配線されるように設定することができます。この場合はポートAが選ばれ、これは既定ポートでもあります。

```
PORTMUX.TCEROUTEA = 0x0;
```

図3-6. TGEROUTEALレジスタ

ビット	7	6	5	4	3	2	1	0
					TCE03~0			
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

● ビット3~0 – TCE03~0 : TCE0信号 (TCE0 Signals)

これらのビット領域はTCE0用ピン位置を制御します。

値	名称	説明							
		WO0	WO1	WO2	WO3	WO4	WO5	WO6	WO7
0 0 0 0	PORTA	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
0 0 1 0	PORTC	PC0	PC1	PC2	PC3	-	-	-	-
0 0 1 1	PORTD	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7
0 1 0 1	PORTF	PF0	PF1	PF2	PF3	PF4	PF5	-	-
1 0 0 0	PORTC2	PA0	PA1	PC0	PC1	PC2	PC3	-	-
1 0 0 1	PORTA2	PA2	PA3	PA4	PA5	PA6	PA7	-	-
その他	-	(予約)							

3. 制御B(CTRLB)レジスタは比較チャンネルの許可ビットと波形生成動作を決めるビット領域を含みます。この例では0,1,2,3のチャンネルが2傾斜PWM動作で使われます。

```
TCE0.CTRLB |= (TCE_CMP0EN_bm | TCE_CMP1EN_bm | TCE_CMP2EN_bm | TCE_CMP3EN_bm);
```

```
TCE0.CTRLB |= TCE_WGMODE_DSBOTh_gc;
```

図3-7. CTRLBレジスタ

ビット	7	6	5	4	3	2	1	0
	CMP3EN	CMP2EN	CMP1EN	CMP0EN	ALUPD	WGMODE2~0		
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 – CMPnEN : 比較n許可 (Compare n Enable)

FRQ(周波数)とPWMの波形生成動作で比較n許可(CMPnEN)ビットは対応するポート出力レジスタでの値を無効にしてWOnに対応するピンでのTCEn波形出力を利用可能にします。

注: 波形拡張(WEX)が許可される場合、WEX出力がTCEn波形出力に優先します。

値	0	1
説明	波形出力WOnは対応するピンで利用できません。	波形出力WOnはFRQとPWMの波形生成動作で対応するピンの出力値に優先します。

● ビット2~0 – WGMODE2~0 : 波形生成動作 (Waveform Generation Mode)

このビット領域は波形生成動作を選び、計数器の計数進行、TOP値、UPDATE条件、割り込み条件、生成される波形の形式を制御します。標準形態の動作では波形生成が全く実行されません。他の全ての動作形態に対して対応する比較n許可(CMPnEN)ビットを設定(1)する場合、波形生成部出力がポートピンに直結されるだけです。ポートピンの方向は出力として設定されなければなりません。

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
名称	NORMAL	FRQ	-	SINGLESLOPE	-	DSTOP	DSBOTH	DSBOTTOM
説明 (動作形態)	標準	周波数	(予約)	1傾斜PWM	(予約)	TOPでの溢れでの2傾斜PWM	TOPとBOTTOMでの溢れでの2傾斜PWM	BOTTOMでの溢れでの2傾斜PWM

4. クロック刻み下降計数(減少)するように計数器を設定するために制御E設定(CTRLESET)レジスタの計数方向(DIR)ビットを'1'に設定してください。DIRビットの既定値は'0'です。

```
TCE0.CTRLESET = TCE_DIR_bm;
```


図3-8. CLRLESETレジスタ

ビット	7	6	5	4	3	2	1	0
					CMD1,0		LUPD	DIR
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0 - DIR : 計数方向 (Counter Direction)

通常、このビットは波形生成動作または事象活動によってハードウェアで制御されますが、ソフトウェアからも変更することができます。このビットへの'1'書き込みはDIRビットを設定(1)し、計数器は下降計数します。

値	0	1
説明	計数器は上昇計数 (増加)	計数器は下降計数 (減少)

5. 尺度動作の振幅(AMP)と変位(OFFSET)のレジスタの使用を許可し、尺度調整方法を底(BOTTOM)に設定してください。デューティサイクル値は0%~100%のデューティサイクルに尺度調整されます。比較(CMPn)レジスタに絶対値を書く代わりに、これを0~2間の分数値として書くことが可能です。

分数値を比較(CMPn)または比較緩衝(CMPnBUF)のレジスタに書く時に、最初にその値は振幅によって乗算されてその後に変位が加えられます。高分解機能はPWMの分解能を2つの追加ビットで4倍、または3つの追加ビットで8倍に増すことができます。この機能は非常に低い周波数で有用です。高分解機能はクロック周期段階を4または8倍短縮します。例えば、高分解能がOFFで20MHzの周波数を持つクロックに対して、1クロック周期は50ns($T[s]=1/f[Hz]=1/20,000,000$)です。4倍の高分解能を持つこのクロックに対して、1クロック周期は12.5nsです。このクロックは8倍の高分解能で6.25nsの1クロック周期を持ちます。

TCE0_CTRLD = TCE_HREN_4X_gc | TCE_SCALE_bm | TCE_AMPEN_bm | TCE_SCALEMODE_BOTTOM_gc;

比較(CMPn)レジスタに書かれる値を尺度調整するための式が下で説明されます。

式3.1. $Scaled_{CMP} = CMP_{frac} \times AMP + OFFSET$: 中央(CENTER)尺度調整動作を使う尺度値

式3.2. $Scaled_{CMP} = CMP_{frac} \times AMP$: 底(BOTTOM)尺度調整動作を使う尺度値

式3.3. $Scaled_{CMP} = CMP_{frac} \times AMP + OFFSET$: 頂上(TOP)尺度調整動作を使う尺度値

式3.4. $Scaled_{CMP} = CMP_{frac} \times AMP$: 頂上底(TOPBOTTOM)尺度調整動作を使う尺度値

CMP_{frac} はAVR16EB16/20/28/32 AVR® EB系統データシートの「表23-6. 尺度動作での実際の比較値」で言及された4つの異なる値を使います。

図3-9. CTRLDレジスタ

ビット	7	6	5	4	3	2	1	0
	HREN1,0		SCALEMODE1,0		AMPEN	SCALE		
アクセス種別	R/W	R/W	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7,6 - HREN1,0 : 高分解能許可 (High Resolution Enable)

このビット領域は高分解能を許可します。高分解能はデューティサイクル分解能を増します。

値	00	01	10	11
名称	OFF	4X	8X	-
説明	高分解能禁止	4倍(2ビット)増加分解能	8倍(3ビット)増加分解能	(予約)

● ビット5,4 - SCALEMODE1,0 : 尺度動作形態 (Scale mode)

このビット領域は変位(OFFSET)がどう生成されるかを定義し、尺度調整が許可された時の比較レジスタ書き込み時に使われます。

値	00	01	10	11
名称	CENTER	BOTTOM	TOP	TOPBOTTOM
説明	書かれた比較値が50%デューティサイクルで中央になるように尺度調整される変位を生成	書かれる比較値が0%デューティサイクルでBOTTOMになるように尺度調整される変位を生成	書かれる比較値が100%デューティサイクルでTOPになるように尺度調整される変位を生成	50%>の書かれる比較値はTOPに、50%<の書かれる比較値はBOTTOMになるように尺度調整され、0%値は比較値0を、100%値はTOP値に等しい比較値を与えます。

● ビット3 – AMPEN : 振幅制御許可 (Amplitude Control Enable)

このビットの設定(1)は振幅(AMP)と変位(OFFSET)のレジスタの使用を許可します。このビットの設定(1)は尺度許可(SCALE)ビットが'1'を書かれた場合にだけ機能します。

● ビット2 – SCALE : 尺度許可 (Scale Enable)

このビットの設定(1)は比較、比較緩衝、振幅のレジスタへの分数値書き込みを許します。尺度調整が許可された場合、比較(TCEn.CMPn)と比較緩衝(TCEn.CMPnBUF)のレジスタは振幅(TCEn.AMP)または定期(TCEn.PER)のレジスタで乗算されます。振幅(TCEn.AMP)レジスタへの書き込みは定期(TCEn.PER)レジスタで乗算されます。書かれる値は0~2-2⁻¹⁵間の範囲を提供するUQ1.15形式でなければなりません。

値	0	1
名称	NORMAL	FRACTIONAL
説明	AMPとCMPnのレジスタ書き込みに絶対値を使用	AMPとCMPnのレジスタ書き込みに分数値を使用

6. 周期(PER)は周期レジスタの緩衝部です。これは式3.5を使ってPWM信号の周波数を設定するのに使われます。

$$\text{式3.5. } f_{DS\ PWM} [Hz] = \frac{N \times f_{CLK} [Hz]}{2 \times \text{TCE前置分周数} \times \text{TCEPER値}}$$

Nは分解能機能に応じて1,4,8の値を持つことができます。

N=1の場合、高分解能機能はOFFにされます。

N=4の場合、分解能が4倍されます。

N=8の場合、分解能が8倍されます。

この例では分解能が4倍にされます。従って、周期(PER)レジスタに書かれる値は式3.6を使って計算されます。

$$\text{式3.6. } \text{TCEPER値} = \frac{4 \times f_{CLK} [Hz]}{2 \times \text{TCE前置分周数} \times f_{DS\ PWM} [Hz]} = \frac{4 \times 20000000}{2 \times 1 \times 10000} \doteq 4000 = 0xFA0$$

TCE0.PER = 0xFA0;

7. 比較(CMPn)レジスタはデューティサイクルを設定するのに使われます。比較レジスタの値は20%、40%、60%、80%のデューティサイクルが望まれるため、20%、40%、60%、80%の1つです。比較レジスタ内に分数値を書くのに緩衝レジスタを使う時は、それらの値が走行時に上書きされます。0~PERの範囲の値を書いてCMPnレジスタに分数値を書く間の違いを強調するために、これらの値は初期化中に0に設定することができます。例えば、尺度動作なしでの20%デューティサイクルに対するCMP0レジスタに書かれる値は、4倍の高分解能で10kHzの周波数または100μsの周期に等しい、与えられた0xFA0のPERレジスタに対して0x320です。尺度動作許可で20%デューティサイクルのPWMに対してCMP0レジスタに書かれる値は0x1999で、このデューティは周期(PER)、振幅(AMP)、変位(OFFSET)のレジスタの値を使ってハードウェアで計算されます。

TCE0.CMP0 = 0x320;

TCE0.CMP1 = 0x640;

TCE0.CMP2 = 0x960;

TCE0.CMP3 = 0xC80;

8. 振幅(AMP)レジスタに望む値を書くことによって振幅を設定してください。

TCE0.AMP = 0x8000;

9. 変位(OFFSET)レジスタに望む値を書くことによって変位を設定してください。

TCE0.OFFSET = 0x00;

10. 制御A(CTRLA)レジスタのクロック選択(CLKSEL)ビット領域を変更することによって前置分周器(分周数)を1に設定してください。計数器を開始するには同じレジスタの許可(ENABLE)ビットを設定(1)しなければなりません。

TCE0.CTRLA = TCE_CLKSEL_DIV1_gc;

TCE0.CTRLA |= TCE_ENABLE_bm;

図3-10. CTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	RUNSTDBY					CLKSEL2~0		ENABLE
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	0	0

● ビット3~1 - CLKSEL2~0 : クロック選択 (Clock Select)

これらのビットはタイマ/カウンタに対するクロック周波数を選びます。

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
名称	DIV1	DIV2	DIV4	DIV8	DIV16	DIV64	DIV256	DIV1024
説明 (f_{TCE})	f_{CLK_PER}	$f_{CLK_PER}/2$	$f_{CLK_PER}/4$	$f_{CLK_PER}/8$	$f_{CLK_PER}/16$	$f_{CLK_PER}/64$	$f_{CLK_PER}/256$	$f_{CLK_PER}/1024$

● ビット0 - ENABLE : 許可 (Enable)

値	0	1
名称	DISABLED	ENABLED
説明	周辺機能は禁止されます。	周辺機能は許可されます。

11. その後、ポートAの0~3(PA0~3)ピンはポートの方向設定(DIRSET)レジスタの対応するビットに'1'を書くことによって出力として設定されます。

```
PORTA.DIRSET = PIN0_bm | PIN1_bm | PIN2_bm | PIN3_bm;
```

12. 振幅と比較の値で変更が起きる時に交互切り替えするのに別のピンを出力として追加してください。このようにして変更を観測します。このピンは実行時間中に交互切り替えします。

```
PORTD.DIRSET = PIN5_bm;
```

13. デューティサイクル尺度調整を見るために走行時間中に振幅(AMP)レジスタの値を変更しなければなりません。振幅レジスタに新しい値の書き込み後、比較(CMPn)レジスタに(元の)旧値を書かなければなりません。これは比較レジスタに新しく値が書かれた時にだけ、デューティサイクル尺度調整更新が起こるからです。

3.2. MCC Melody実装

MPLABコード構成部Melody、MCC Melodyを使ってこのプロジェクトを生成するには次の手順に従ってください(MCCクラスは支援されません)。

- AVR16EB23に対してMPLAB® X IDEプロジェクトを作成してください。
- ツールバーからMCCを開いてください(ここでMCCプラグイン インストールでのより多くの情報を見つけてください)。
- MCC Content Manager(MCC内容管理)ウィザードでMCC Melodyを選び、その後にFinishをクリックしてください。
- Project Resources(プロジェクト資源)⇒System(システム)⇒CLKCTRLへ行ってください。
 - Prescaler enable(前置分周器許可)鈕を禁止してください。
 - Clock Select(クロック選択) : PLL Oscillator(PLL発振器)
 - PLL Multiplication(PLL倍率) : Multiply by 16(16倍)
 - PLL Source Division(PLL供給元分周) : DIV4(4分周)
 - Prescaler B division PDIVB(前置分周器B分周 PDIVB) : DIV4(4分周)
- Device Resources(デバイス資源)⇒Drivers(ドライバ)⇒Timer(計時器)からTCE単位部を追加し、その後に以下の構成設定を行ってください。
 - Module Enable(単位部許可) : 既定で許可されなければなりません。そうでなければ、鈕を切り替えてください(許可なら青になります)。
 - Clock Selection(クロック選択) : (既定で)System clock(システムクロック)、(分周器は1 - System clockでなければなりません)。
 - Waveform Generation Mode(波形生成動作) : TOPとBOTTOMでの溢れを持つ2傾斜PWM動作(DSBOTH)
 - Requested Period [s](要求周期) : 0.001
 - Duty Cycle 0 [%](デューティサイクル 0) : 20
 - Duty Cycle 1 [%](デューティサイクル 1) : 40
 - Duty Cycle 2 [%](デューティサイクル 2) : 60
 - Duty Cycle 3 [%](デューティサイクル 3) : 80
 - Waveform Output n(波形出力n) : 波形出力0,1,2,3に対する許可列で枠をチェックしてください。
 - Duty Cycle high Resolution(デューティサイクル高分解能) : 4Xによって増された分解能
 - Scale mode(尺度動作) : CMP値はBOTTOM、0% DCから尺度調整されます。
 - Scaled Writing to registers(レジスタへの尺度調整した書き込み) : Fractional(分数)
 - Amplitude Control Enable(振幅制御許可) : 鈕を切り替えてください(許可されたなら青になります)。
 - Amplitude Value(振幅値) : 1

6. **Pin Grid View**(ピン格子表示)タブのPORTAでTCE_WO0～3ピンが出力として固定化されているかを調べてください。**Waveform Output n**(波形出力n)で**Enable**(許可)列での枠がチェックされていると、ピンも固定化されます。PORTを変更するには、**Pin Grid View**で別のPORTからピンをクリックしてください。振幅と比較の値変更を見るために、PORTDのPIN5を出力として選び、その後にそれを切り替えてください。
7. **Project Resources**(プロジェクト資源)タブで、指定したドライバと構成設定の全てをMCCが生成するように**Generate**(生成)鈕をクリックしてください。
8. 下で示されるように**main.c**ファイルを編集してください。
インクルードファイル部に次のように追加してください。

```
#include "mcc_generated_files/system/system.h"
#include <util/delay.h>
```

次のようにマクロ定義を追加してください。

```
/* 周期(PERIOD)、比較(CMPn)、振幅(AMP)、変位(OFFSET)のレジスタに対して計算された値 */
#define DUTY_CYCLE_20_PERCENT          (0x1999)
#define DUTY_CYCLE_40_PERCENT          (0x3333)
#define DUTY_CYCLE_60_PERCENT          (0x4CCC)
#define DUTY_CYCLE_80_PERCENT          (0x660C)
#define AMPLITUDE_MAX_DCY_50_PERCENT   (0x4000)
#define AMPLITUDE_MAX_DCY_75_PERCENT   (0x6000)
#define AMPLITUDE_MAX_DCY_100_PERCENT  (0x8000)
#define AMPLITUDE_MAX_DCY_150_PERCENT  (0xC000)
```

次のように関数を追加してください。

```
void Amplitude_Value_Set(uint16_t value)
{
    /* 振幅の値を設定 */
    TCE0_AmplitudeSet(value);

    /* デューティサイクルが新振幅値に従って尺度調整されるようにCMPBUFレジスタの値を再書き込み */
    TCE0_CompareAllChannelsBufferedSet(DUTY_CYCLE_20_PERCENT, DUTY_CYCLE_40_PERCENT,
                                        DUTY_CYCLE_60_PERCENT, DUTY_CYCLE_80_PERCENT);

    /* 振幅値が変更された時を知るためにPD5ピンを切り替え */
    IO_PD5_Toggle();
}
```

次のように主関数を編集してください。

```
int main(void)
{
    SYSTEM_Initialize();

    while(1)
    {
        Amplitude_Value_Set(AMPLITUDE_MAX_DCY_50_PERCENT);
        _delay_ms(10);
        Amplitude_Value_Set(AMPLITUDE_MAX_DCY_75_PERCENT);
        _delay_ms(10);
        Amplitude_Value_Set(AMPLITUDE_MAX_DCY_100_PERCENT);
        _delay_ms(10);
        Amplitude_Value_Set(AMPLITUDE_MAX_DCY_150_PERCENT);
        _delay_ms(10);
    }
}
```

9. 今や、プロジェクトはMPLAB X IDEから構築して走行することができます。走行時、比較レジスタに対する尺度値が10ms毎に変更され、それによってデューティサイクルの範囲が変更されます。

3.3. 結果

次には4つのPWM信号がどう見え、振幅と変位を使ってデューティサイクルがどう作成されるかを示すためにいくつかのロジックアナライザ捕獲が示されます。

図3-11. 0~100%のPERIODに尺度調整されたデューティサイクルの範囲

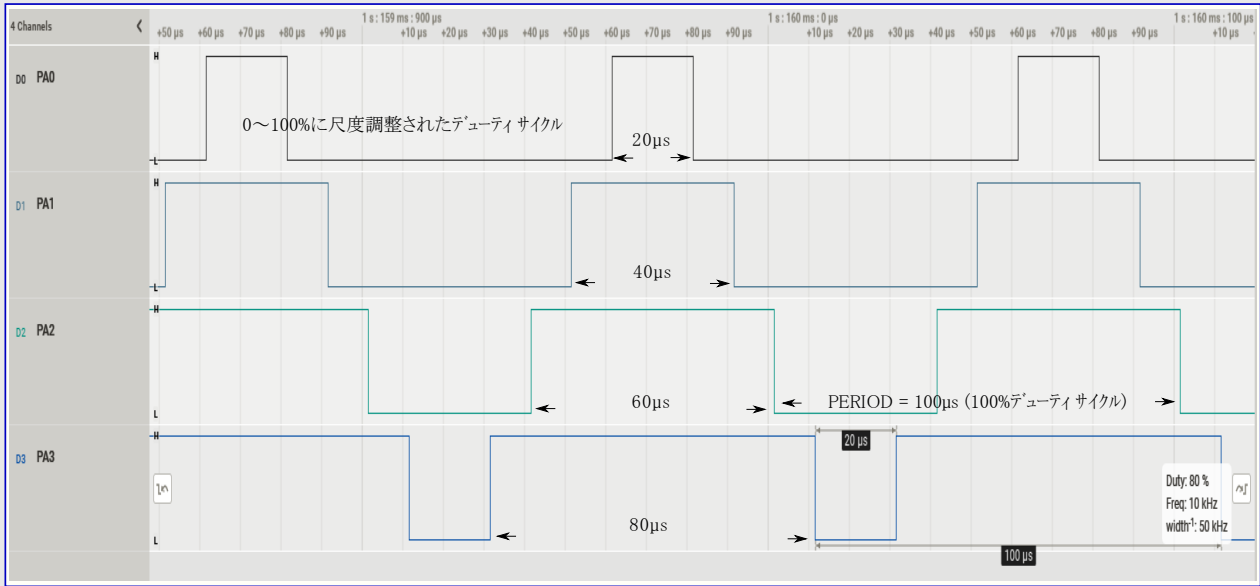


図3-12. 0~50%のPERIODに尺度調整されたデューティサイクルの範囲

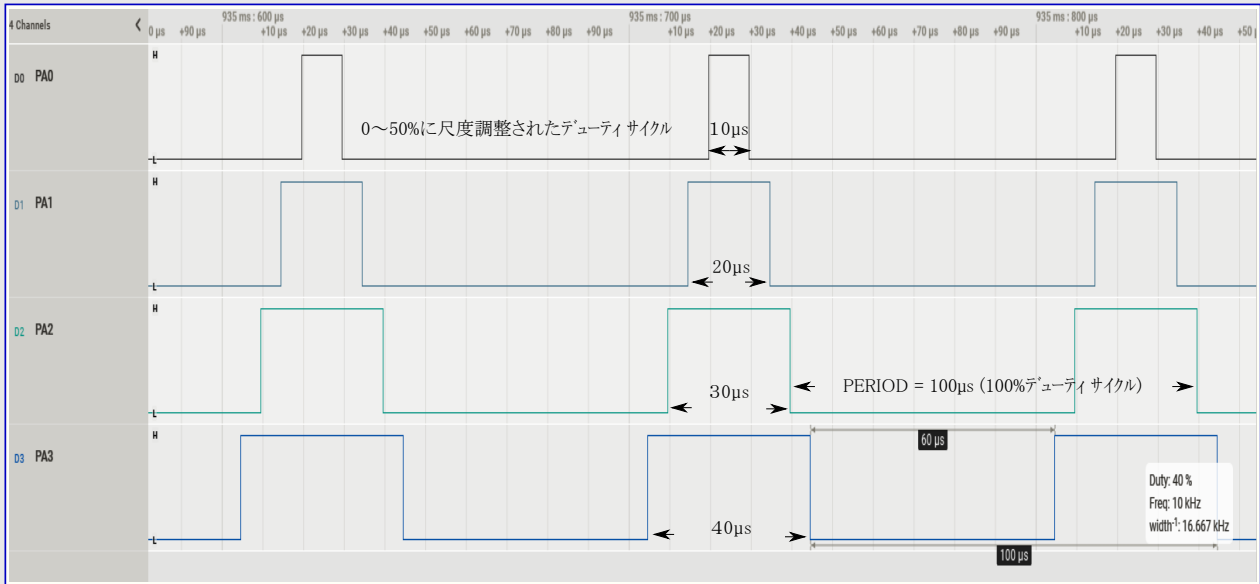
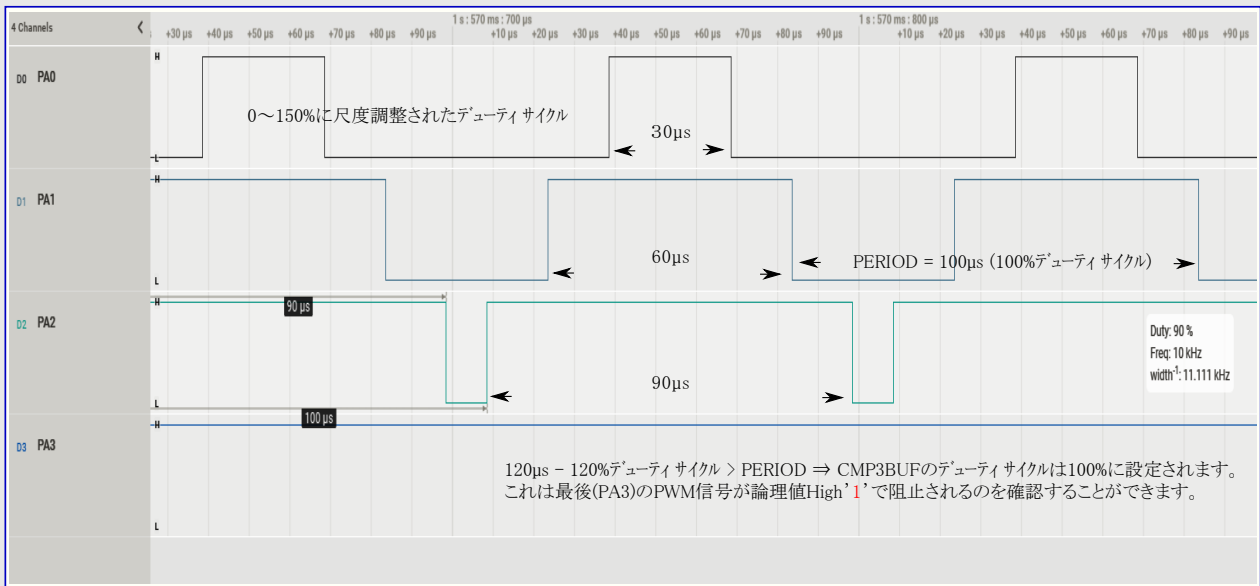


図3-13. 0~150%のPERIODに尺度調整されたデューティサイクルの範囲



本章で記述されたのと同じ機能性を持つAVR16EB32用素のコード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

本章で記述されたのと同じ機能性を持つMCCで生成したAVR16EB32用コード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

4. WEXを使う模様型生成

波形拡張の基本的な使用事例は周期的に変更する出力信号に対して模様型を設定することです。この模様型生成動作はポートピンの上書きにWEXを使いたい場合に役立ちます。模様型生成動作でWEXを使う時にTCEからの出力ピンが制御されます。模様型生成動作を許可して適切な符号を含む望む模様型を設定しなければなりません。

初期化を含む基本的な例が下で提供されます。このプログラムはソフトウェア遅延を使って25µs毎に10回、ピンを(相補的な模様型)で切り替えます。更に250µs後、各段階間で5µsの遅延付きで階段模様が生成されます。更に250µs後、この処理を繰り返します。

下で記述されるように、ピンは波形拡張の初期化前に方向(DIR)レジスタの対応するビットを設定(1)することによって出力として構成設定されなければなりません。ポートAの0~7(PA0~7)ピンが選ばれました。模様は信号出力を見るためにロジックアナライザや8つのLEDを使って上手く観測されます。

次の2つの項が望む動きを行うためのWEX周辺機能構成設定を説明します。最初の項は素のコード'を使ってWEXを構成設定する方法を記述します。2つ目の項はMCC Melodyを使ってWEXを構成設定する方法を説明します。最後の項は結果を含みます。

4.1. 素の実装

1. 制御A(CTRLA)レジスタでの模様型生成動作許可はTCEが生成した波形を上書きします。

```
WEX0. CTRLA = WEX_PGM_bm;
```

図4-1. CTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	PGM	INMX2~0			DTI3EN	DTI2EN	DTI1EN	DTI0EN
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 - PGM : 模様型生成動作 (Pattern Generation Mode)**

このビットへの'1'書き込みは模様型生成動作を許可します。模様型生成動作では模様型を格納するのに沈黙時間緩衝レジスタが使われます。故にこの動作でこれらの緩衝部は利用できません。

2. 全ての信号出力は模様型生成動作で上書きされるように許可されなければなりません。全ての信号は模様型生成動作上書き(PGMOVR)レジスタのビットと1対1の関係を持ちます。

```
WEX0. PGMOVR = WEX_PGMOVR0_bm | WEX_PGMOVR1_bm | WEX_PGMOVR2_bm | WEX_PGMOVR3_bm |
                WEX_PGMOVR4_bm | WEX_PGMOVR5_bm | WEX_PGMOVR6_bm | WEX_PGMOVR7_bm;
```

図4-2. PGMOVRレジスタ

ビット	7	6	5	4	3	2	1	0
	PGMOVR7	PGMOVR6	PGMOVR5	PGMOVR4	PGMOVR3	PGMOVR2	PGMOVR1	PGMOVR0
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 - PGMOVRn : 模様型生成動作上書き (Pattern Generation Mode Override)**

このレジスタは模様型生成動作(PGM)を許可します。ビットが'1'に設定された場合、模様型生成動作出力(PGMOUT)レジスタの対応するビットがSWAP部からの出力を上書きする値を指定します。

下表はこのビット領域の各ビットに対して利用可能な構成設定を示します。

値	0	1
説明	ピン(Pxn)出力は上書きされません。	ピン(Pxn)出力はPGMOUTレジスタの値で上書きされます。

3. 模様型生成動作出力(PGMOUT)レジスタは全ての信号に対する模様型を保持します。全ての信号がPGMOUTレジスタのビットと1対1の関係を持つことは言及する価値があります。例えば、模様型0xAAを達成するには、PGMOUTレジスタでビット1,3,5,7を論理High'1'に設定してください。

```
WEX0. PGMOUT = WEX_PGMOUT1_bm | WEX_PGMOUT3_bm | WEX_PGMOUT5_bm | WEX_PGMOUT7_bm;
```

図4-3. PGMOUTレジスタ

ビット	7	6	5	4	3	2	1	0
	PGMOUT7	PGMOUT6	PGMOUT5	PGMOUT4	PGMOUT3	PGMOUT2	PGMOUT1	PGMOUT0
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 – PGMOUTn : 模様型生成動作出力 (Pattern Generation Mode Output)

このレジスタはWEXが模様型生成動作の時に実施する上書き値を保持します。

下表はこのビット領域の各ビットに対して利用可能な構成設定を示します。

値	0	1
説明	ピンn(Pxn)の波形出力値はLowに駆動されます。	ピンn(Pxn)の波形出力値はHighに駆動されます。

4. WEXを模様型生成動作に構成設定後、PGMOUTレジスタを使って各信号の状態を交互切り替えることができます。

```
WEX0.PGMOUT = ~WEX0.PGMOUT;
```

5. 切り替え発生時を見るために遅延を追加してください。

```
_delay_us(25);
```

6. ポートの方向設定(DIRSET)レジスタの対応するビットに'1'を書くことによってポートAの0~7(PA0~7)ピンは出力として設定されます。これらのGPIOは見ることのできる出力を得るためだけに構成設定されます。

```
PORTA.DIRSET = PIN0_bm | PIN1_bm | PIN2_bm | PIN3_bm | PIN4_bm | PIN5_bm | PIN6_bm | PIN7_bm;
```

7. ポート多重器(PORTMUX)で既定のポートAピンを選ぶように設定を行ってください。

```
PORTMUX.TCEROUTEA = 0x0;
```

図4-4. TCEROUTEAレジスタ

ビット	7	6	5	4	3	2	1	0	
	TCE03~0								
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W	
リセット値	0	0	0	0	0	0	0	0	

● ビット3~0 – TCE03~0 : TCE0信号 (TCE0 Signals)

これらのビット領域はTCE0用ピン位置を制御します。

値	名称	説明							
		WO0	WO1	WO2	WO3	WO4	WO5	WO6	WO7
0 0 0 0	PORTA	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
0 0 1 0	PORTC	PC0	PC1	PC2	PC3	-	-	-	-
0 0 1 1	PORTD	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7
0 1 0 1	PORTF	PF0	PF1	PF2	PF3	PF4	PF5	-	-
1 0 0 0	PORTC2	PA0	PA1	PC0	PC1	PC2	PC3	-	-
1 0 0 1	PORTA2	PA2	PA3	PA4	PA5	PA6	PA7	-	-
その他	-	(予約)							

8. ポートA上書きを許可するために出力上書き許可(OUTOVEN)レジスタで対応するビットを設定(1)することによってWEXの出力を許可してください。

```
WEX0.OUTOVEN = WEX_OUTOVEN0_bm | WEX_OUTOVEN1_bm | WEX_OUTOVEN2_bm | WEX_OUTOVEN3_bm |
WEX_OUTOVEN4_bm | WEX_OUTOVEN5_bm | WEX_OUTOVEN6_bm | WEX_OUTOVEN7_bm;
```

図4-5. OUTOVENレジスタ

ビット	7	6	5	4	3	2	1	0
	OUTOVEN7	OUTOVEN6	OUTOVEN5	OUTOVEN4	OUTOVEN3	OUTOVEN2	OUTOVEN1	OUTOVEN0
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 – OUTOVENn : 出力上書き許可 (Output Override Enable)

このビット領域は対応するポート出力レジスタ(即ち、波形出力と1対1の関係)の上書きを許可します。

このレジスタは障害保護が許可された(障害制御(WEXn.FAULTCTRL)レジスタの障害検出活動(FDACT)ビット領域がNONE以外の場合に解除(0)することができません。

9. この例の最後の段階は定義された模様型を更新するのにTCEからWEXに渡す供給元クロックが必要とされるため、TCE単位部を許可することです。

```
TCE0.CTRLA = TCE_ENABLE_bm;
```

4.2. MCC Melody実装

MPLABコード構成部Melody、MCC Melodyを使ってこのプロジェクトを生成するには次の手順に従ってください(MCCクラスは支援されません)。

- AVR16EB23に対してMPLAB® X IDEプロジェクトを作成してください。
- ツールバーからMCCを開いてください(ここでもMCCプラグイン インストールでのより多くの情報を見つけてください)。
- MCC Content Manager(MCC内容管理)ウィザードでMCC Melodyを選び、その後にFinishをクリックしてください。
- Device Resources(デバイス資源)⇒Drivers(ドライバ)⇒Timer(計時器)へ行ってください。TCE単位部を追加し、その後に以下の構成設定を行ってください。
 - Module Enable(単位部許可)： 鈕を切り替えてください(許可なら青になります)。
- Device Resources(デバイス資源)⇒Drivers(ドライバ)へ行ってください。WEX単位部を追加し、その後に以下の構成設定を行ってください。
 - Input Matrix(入力配列)： Direct(直接)
 - Update Source(更新供給元)： TCE (出力信号に対する更新条件はTCE単位部になります)。
 - Override Settings(上書き設定)： 波形出力0～7に対するOutput Enable(出力許可)列で全ての枠をチェックしてください。
 - Pattern Generation Mode Enable(模様型生成動作許可)： 鈕を切り替えてください(許可なら青になります)。
 - Pattern Generation Actions(模様型生成活動)： 模様型を設定するためにOverride Enable(上書き許可)列から全ての枠をチェックして各出力を望む状態(LOWまたはHIGH)に設定してください。
- Pin Grid View(ピン格子表示)タブのPORTAでTCE_WO0～7ピンが出力として固定化されているかを調べてください。Override Settings(上書き設定)でOutput Enable(出力許可)列での枠がチェックされていると、ピンも固定されます。PORTを変更するには、Pin Grid Viewで別のPORTからピンをクリックしてください。
- Project Resources(プロジェクト資源)タブで、指定したドライバと構成設定の全てをMCCが生成するようにGenerate(生成)鈕をクリックしてください。
- 下で示されるようにmain.cファイルを編集してください。
インクルード ファイル部に次のように追加してください。

```
#include "mcc_generated_files/system/system.h"
#include <util/delay.h>
```

次のようにマクロ定義を追加してください。

```
/* PGMOUTレジスタに書かれる模様型 */
#define COMPLEMENTARY_PATTERN (WEX_PGMOUT6_bm | WEX_PGMOUT4_bm |
                               WEX_PGMOUT2_bm | WEX_PGMOUT0_bm)

#define STAIRCASE0_PATTERN (WEX_PGMOUT0_bm)
#define STAIRCASE1_PATTERN (WEX_PGMOUT1_bm)
#define STAIRCASE2_PATTERN (WEX_PGMOUT2_bm)
#define STAIRCASE3_PATTERN (WEX_PGMOUT3_bm)
#define STAIRCASE4_PATTERN (WEX_PGMOUT4_bm)
#define STAIRCASE5_PATTERN (WEX_PGMOUT5_bm)
#define STAIRCASE6_PATTERN (WEX_PGMOUT6_bm)
#define STAIRCASE7_PATTERN (WEX_PGMOUT7_bm)
#define PATTERN_RESET (0x00)
```

次のように関数を追加してください。

```
void Complementary_Pattern_Set(void)
{
    WEX0_PatternGenerationOutputSet (COMPLEMENTARY_PATTERN);
    uint8_t complementary_pattern = COMPLEMENTARY_PATTERN;
    _delay_us (25);

    /* 相補的信号模様型 */
    for (uint8_t i = 0; i < 9; i++)
    {
        /* Complementary_pattern変数は全ての段階で変更 */
        complementary_pattern = ~complementary_pattern;
    }
}
```

```

    /* WEXの出力の各々に対して模様型を切り替え */
    WEX0_PatternGenerationOutputSet (complementary_pattern);

    /* 切り替え可視化のために追加されたソフトウェア遅延 */
    _delay_us (25);
}

/* 全ての信号を論理Low'0'にし、相補的模様の1つから階段模様への遷移を見るために250µs待機 */
WEX0_PatternGenerationOutputSet (PATTERN_RESET);
}

void Stairs_Pattern_Set(void)
{
    /* 各信号は階段増加模様を生成するために昇順で一度に1つずつLowからHighに切り替わります。 */
    WEX0_PatternGenerationOutputSet (STAIRCASE7_PATTERN);

    /* 増加を可視化するために追加されたソフトウェア遅延 */
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE6_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE5_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE4_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE3_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE2_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE1_PATTERN);
    _delay_us (5);

    /* 各信号は階段減少模様を生成するために降順で一度に1つずつLowからHighに切り替わります。 */
    WEX0_PatternGenerationOutputSet (STAIRCASE0_PATTERN);

    /* 減少を可視化するために追加されたソフトウェア遅延 */
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE1_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE2_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE3_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE4_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE5_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE6_PATTERN);
    _delay_us (5);
    WEX0_PatternGenerationOutputSet (STAIRCASE7_PATTERN);
    _delay_us (5);

    /* 全ての信号を論理Low'0'にし、階段模様から相補的模様の1つへの遷移を見るために250µs待機 */
    WEX0_PatternGenerationOutputSet (PATTERN_RESET);
}

```

次のように主関数を編集してください。

```

int main(void)
{
    SYSTEM_Initialize();
}

```

```

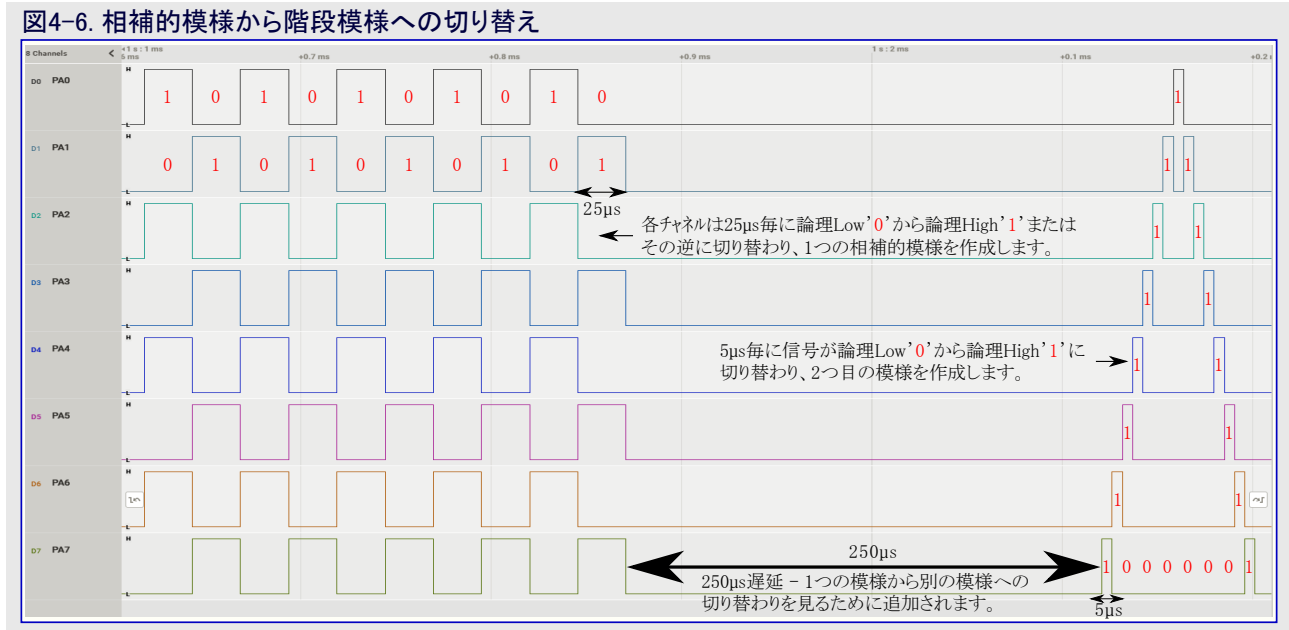
while(1)
{
    Complementary_Pattern_Set();
    _delay_us(250);
    Stairs_Pattern_Set();
    _delay_us(250);
}

```

9. プロジェクトは今やMPLAB X IDEを使って構築して走行することができます。

4.3. 結果

生成した模様型がどう見えるかを示すためにロジックアナライザ捕獲が示されます。



本章で記述されたのと同じ機能性を持つAVR16EB32用素のコード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

本章で記述されたのと同じ機能性を持つMCCで生成したAVR16EB32用コード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

5. TCEとWEXでの8つのPWM信号生成

TCEは4つのチャンネルで正確で多用途のPWM信号を生成し、重複しない補完波形を生成することができます。

WEXはTCEによって与えられた4つの比較チャンネルを8つまでのPWM信号生成能力へ拡張するように構成設定することができます。

下は緩衝を使う可変デューティサイクルを持ち20kHzで8つの補完PWM信号を生成するようにTCEとWEXの実体を設定する方法の例で、参考までにTCEとWEXを使うPWM信号生成部分を使います。信号は追加した沈黙時間のために重複なしでの2つの対で、トランジスタ切り替えでの貫通電流を防ぐため電動機制御で重要な機能です。比較レジスタの更新は各チャンネルに対する比較一致割り込み中に起こります。この例では更に障害保護機能も強調されます。ソフトウェア事象が起動されると、それは障害事象を模倣し、全ての信号がLowに駆動され、これは1ms毎に発生します。これを行うためにWEXを障害事象検出用に、ソフトウェア事象を生成するのにEVSYSを構成設定してください。

次の2つの項が望む動きを行うためのTCEとWEXの周辺機能構成設定方法を説明します。最初の項は素のコードを使ってTCEとWEXの構成設定を明確にします。2つ目の項はMCC Melodyを使うTCEとWEXの構成設定を説明します。最後の項は結果を含みます。

5.1. 素の実装

1. クロック周辺機能が20MHzの周波数で動かなければなりません。既定クロック速度は3.33MHzで、既定で6分周の前置分周器を持ちます。このコード例は前置分数器を禁止してクロックを最大速度で使うことが必要です。

```
_PROTECTED_WRITE (CLKCTRL.MCLKCTRLB, CLKCTRL.MCLKCTRLB & ~CLKCTRL_PEN_bm);
```

図5-1. CLKCTRLのMCLKCTRLBレジスタ

ビット	7	6	5	4	3	2	1	0	
	PBDIV		PDIV3~0						PEN
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	1	0	0	0	1	

- ビット0 - PEN : 前置分周器許可 (Prescaler Enable)

このビットは主クロック(CLK_MAIN)前置分周器が許可されるかを制御します。

値	0	1
説明	CLK_MAIN前置分周器禁止	CLK_MAIN前置分周器許可、分周比は前置分周器分周値(PDIV)ビット領域によって制御されます。

2. ポート多重器のTCEとWEXに対応するレジスタは単位部出力を違うポートに配線することができます。この場合はポートAが選ばれ、これは既定ポートでもあります。

```
PORTMUX.TCROUTEA = 0x0;
```

図5-2. TCROUTEALレジスタ

ビット	7	6	5	4	3	2	1	0
	TCE03~0				TCE03~0			
アクセス種別	R	R	R	R	R/W	R/W	R/W	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット3~0 - TCE03~0 : TCE0信号 (TCE0 Signals)

これらのビット領域はTCE0用ピン位置を制御します。

値	名称	説明							
		WO0	WO1	WO2	WO3	WO4	WO5	WO6	WO7
0 0 0 0	PORTA	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
0 0 1 0	PORTC	PC0	PC1	PC2	PC3	-	-	-	-
0 0 1 1	PORTD	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7
0 1 0 1	PORTF	PF0	PF1	PF2	PF3	PF4	PF5	-	-
1 0 0 0	PORTC2	PA0	PA1	PC0	PC1	PC2	PC3	-	-
1 0 0 1	PORTA2	PA2	PA3	PA4	PA5	PA6	PA7	-	-
その他	-	(予約)							

3. 制御B(CTRLB)レジスタは比較チャンネルの許可ビットと波形生成動作を決めるビット領域を含みます。この例では0,1,2,3のチャンネルが単一傾斜PWM動作で使われます。

```
TCE0.CTRLB |= (TCE_CMP0EN_bm | TCE_CMP1EN_bm | TCE_CMP2EN_bm | TCE_CMP3EN_bm);
```

```
TCE0.CTRLB |= TCE_WGMODE_SINGLESLOPE_gc;
```

図5-3. TCEのCTRLBレジスタ

ビット	7	6	5	4	3	2	1	0			
	CMP3EN		CMP2EN		CMP1EN		CMP0EN		ALUPD	WGMODE2~0	
アクセス種別	R/W	R/W	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	0	0	

● ビット7~4 - CMPnEN : 比較n許可 (Compare n Enable)

FRQ(周波数)とPWMの波形生成動作で比較n許可(CMPnEN)ビットは対応するポート出力レジスタでの値を無効にしてWOnに対応するピンでのTCEn波形出力を利用可能にします。

注: 波形拡張(WEX)が許可される場合、WEX出力がTCEn波形出力に優先します。

値	0	1
説明	波形出力WOnは対応するピンで利用できません。	波形出力WOnはFRQとPWMの波形生成動作で対応するピンの出力値に優先します。

● ビット2~0 - WGMODE2~0 : 波形生成動作 (Waveform Generation Mode)

このビット領域は波形生成動作を選び、計数器の計数進行、TOP値、UPDATE条件、割り込み条件、生成される波形の形式を制御します。標準形態の動作では波形生成が全く実行されません。他の全ての動作形態に対して対応する比較n許可(CMPnEN)ビットを設定(1)する場合、波形生成部出力がポートピンに直結されるだけです。ポートピンの方向は出力として設定されなければなりません。

値	000	001	010	011	100	101	110	111
名称	NORMAL	FRQ	-	SINGLESLOPE	-	DSTOP	DSBOTH	DSBOTTOM
説明 (動作形態)	標準	周波数	(予約)	1傾斜PWM	(予約)	TOPでの溢れでの2傾斜PWM	TOPとBOTTOMでの溢れでの2傾斜PWM	BOTTOMでの溢れでの2傾斜PWM

4. クロック刻み下降計数(減少)するように計数器を設定するためにTCEの制御E設定(CTRLSET)レジスタの計数方向(DIR)ビットを'1'に設定してください。DIRビットの既定値は'0'です。

```
TCE0.CTRLSET = TCE_DIR_bm;
```

図5-4. TCEのCLRLESETレジスタ

ビット	7	6	5	4	3	2	1	0
					CMD1,0		LUPD	DIR
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0 - DIR : 計数方向 (Counter Direction)

通常、このビットは波形生成動作または事象活動によってハードウェアで制御されますが、ソフトウェアからも変更することができます。このビットへの'1'書き込みはDIRビットを設定(1)し、計数器は下降計数します。

値	0	1
説明	計数器は上昇計数(増加)	計数器は下降計数(減少)

5. 周期(PER)はTCEの周期レジスタの緩衝部です。これは式5.1を使ってPWM信号の周波数を設定するのに使われます。

$$\text{式5.1. } f_{SS \text{ PWM}} [\text{Hz}] = \frac{f_{\text{CLK}} [\text{Hz}]}{\text{TCE前置分周数} \times (\text{TCEPER値} + 1)}$$

この例の目標値を考慮すると、

$$\text{TCEPER値} = \frac{f_{\text{CLK}} [\text{Hz}]}{\text{TCE前置分周数} \times f_{SS \text{ PWM}} [\text{Hz}]} = \frac{20000000}{1 \times 20000} \doteq 1000 = 0x3E8$$

/* PERレジスタは常に望む値-1、1000-1=999で設定されます(50µs周期に対する望む値)。*/
TCE0.PER = 0x3E7;

6. 比較(CMPn)レジスタはデューティサイクルを設定する能力があるその緩衝部を使って更新されます。比較レジスタの値は20%、40%、60%、80%のように任意のデューティサイクルを持つように設定することができます。けれども、最初の割り込み発生時にそれらの緩衝部を使ってCMPnレジスタを更新することによって走行時にそれらの値が書き込まれます。故にこれらの値を'0'に設定します。

```
TCE0.CMP0 = 0x00;
```

```
TCE0.CMP1 = 0x00;
```

```
TCE0.CMP2 = 0x00;
```

```
TCE0.CMP3 = 0x00;
```

7. 計時器開始後、デューティサイクルは比較レジスタの値がカウンタ値と一致する時毎に増します。デューティサイクルの範囲は0~100%の間です。割り込み処理ルーチン(ISR)中に、これは全ての比較チャンネルに対して起こります。比較0チャンネルに対するISRは次のとおりです。

```
ISR(TCE0_CMP0_vect)
{
    /* 割り込み要求フラグ解除 */
    TCE0.INTFLAGS = TCE_CMP0_bm;

    static uint16_t duty_cycle = 0;

    /* 割り込みでデューティ サイクル更新 */
    duty_cycle += 5;
    if(duty_cycle >= MAX_DUTY_CYCLE)
        duty_cycle = 0;
    TCE0.CMP0BUF = duty_cycle;
}
```

8. TCEの比較レジスタの各々に対して比較一致での割り込みを許可してください。

```
TCE0.INTCTRL = (TCE_CMP0_bm | TCE_CMP1_bm | TCE_CMP2_bm | TCE_CMP3_bm);
```

図5-5. TCEのINTCTRLレジスタ

ビット	7	6	5	4	3	2	1	0
	CMP3	CMP2	CMP1	CMP0				OVF
アクセス種別	R/W	R/W	R/W	R/W	R	R	R	R/W
リセット値	0	0	0	0	0	0	0	0

- ビット7,6,5,4 - CMPn : 比較チャンネルn割り込み要求フラグ (Compare Channel n Interrupt Flag)

比較割り込み要求(CMPn)フラグは対応する比較チャンネルでの比較一致で設定(1)されます。全ての動作形態に対して、CMPnフラグは計数(TCEn.CNT)レジスタと対応する比較(TCEn.CMPn)レジスタ間で比較一致が起こる時に設定(1)されます。CMPnフラグは自動的に解除(0)されず、そのビット位置に'1'を書くことによってだけ解除(0)されます。

9. TCEカウンタの初期値を'0'に設定してください。

```
TCE0.CNT = 0x00;
```

10. 制御A(CTRLA)レジスタのクロック選択(CLKSEL)ビット領域を変更することによって前置分周器を'0'に設定してください。カウンタを開始するため、同じレジスタの許可(ENABLE)ビットを設定(1)しなければなりません。

```
TCE0.CTRLA = TCE_CLKSEL_DIV1_gc;
```

TCE構成設定の最後の段階はこの単位部を許可することです。

```
TCE0.CTRLA |= TCE_ENABLE_bm;
```

図5-6. TCEのCTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	RUNSTDBY					CLKSEL2~0		ENABLE
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	0	0

- ビット3~1 - CLKSEL2~0 : クロック選択 (Clock Select)

これらのビットはタイマ/カウンタに対するクロック周波数を選びます。

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
名称	DIV1	DIV2	DIV4	DIV8	DIV16	DIV64	DIV256	DIV1024
説明 (f _{TCE})	f _{CLK_PER}	f _{CLK_PER} /2	f _{CLK_PER} /4	f _{CLK_PER} /8	f _{CLK_PER} /16	f _{CLK_PER} /64	f _{CLK_PER} /256	f _{CLK_PER} /1024

- ビット0 - ENABLE : 許可 (Enable)

値	0	1
名称	DISABLED	ENABLED
説明	周辺機能は禁止されます。	周辺機能は許可されます。

11. WEXの出力配線配列を直接動作(DIRECT)に設定し、2つの補完信号の各対に対して沈黙時間挿入を許可してください。これらの設定を行った後、WEXは8つの補完信号を生成するのにTCEから生成された4つのPWM信号を使います。

```
WEX0. CTRLA = WEX_INMX_DIRECT_gc | WEX_DTI0EN_bm | WEX_DTI1EN_bm | WEX_DTI2EN_bm | WEX_DTI3EN_bm;
```

図5-7. WEXのCTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	PGM	INMX2~0			DTI3EN	DTI2EN	DTI1EN	DTI0EN
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6~4 - INMX2~0 : 入力配列 (Input Matrix)

このビット領域はWEX内部単位部に対するタイマ/カウンタ波形生成入力の配列配線を定義します。より多くの情報については表24-1を参照してください。

値	0 0	0 1	1 0	1 1
名称	DIRECT	-	CWCMA	CWCMB
説明	TCE0から直接	(予約)	共通波形チャネル動作A、単一パルス幅変調(PWM)チャネルでの波形出力(WO)	共通波形チャネル動作B、2つのPWMチャネルでの波形出力(WO)

- ビット3,2,1,0 - DTInEN : 沈黙時間挿入CMPn許可 (Dead-Time Insertion CMPn Enable)

値	0	1
名称	DISABLED	ENABLED
説明	沈黙時間が挿入されません。	沈黙時間が挿入されます。

12. 沈黙時間レジスタを使って望む沈黙時間の値を設定してください。沈黙時間は周辺機能クロック刻みで測られます。下の式は沈黙時間を計算する方法を説明します。

```
WEX0. DTLS = 0x03;
```

```
WEX0. DTHS = 0x05;
```

図5-8. WEXの沈黙時間レジスタ : 沈黙時間Low側、High側、両側

DTLS	7~0	DTLS7~0		
DTHS	7~0	DTHS7~0		
DTBOTH	7~0	DTBOTH7~0		

応用に対して対称的な沈黙時間を達成するには最後のレジスタを使ってください。この例では非対称の沈黙時間を作成するため、Low側とHigh側の沈黙時間レジスタが使われます。

沈黙時間を計算するのに使われる式は次のとおりです。

$$\text{式5.2. } \text{clock_tick [ns]} = \frac{1}{f_{\text{CLK}} [\text{Hz}]} \times \text{TCE前置分周数} \times 10^9$$

クロック刻みをnsに変換すると、例えば次のようになります。

```
fCLK [Hz] = 20MHz
TCE前置分周数 = 4
望む沈黙時間 [ns] = 500
```

$$\text{clock_tick [ns]} = \frac{1}{20000000} \times 4 \times 10^9 = 250$$

$$\text{レジスタ値} = \text{沈黙時間 [ns]} \div \text{clock_tick [ns]} = 500 \div 250 = 2$$

13. 障害制御(FaultCTRL)レジスタで障害再開動作条件と障害信号駆動様式を設定してください。

```
/* 障害再開動作をラッチ動作に設定 - 障害は障害条件が有効な限り有効です。 */
/* ラッチ動作の障害から再開するにはソフトウェア解消指令を使わなければなりません。 */
WEX0.FAULTCTRL = WEX_FDMODE_LATCHED_gc;
```

```
/* 障害検出時に全てのピンを論理Low'0'に駆動 */
WEX0.FAULTCTRL |= WEX_FDACT_LOW_gc;
```

図5-9. WEXのFAULTCTRLレジスタ

ビット	7	6	5	4	3	2	1	0
	FDDBD					FDMODE	FDACT1,0	
アクセス種別	R/W	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 - FDMODE : 障害検出再開動作 (Fault Detection Restart Mode)

値	0	1
名称	LATCHED	CBC
説明	ラッチ動作。出力は障害条件がもはや有効でなく、事象入力x障害検出フラグ(INTFLAGSレジスタのFDDEVx)がソフトウェアによって解除(0)されるまで障害状態に留まります。	周期毎動作。波形出力は障害条件がもはや有効でない時まで障害状態に留まります。

- ビット1,0 - FDACT1,0 : 障害検出活動 (Fault Detection Action)

値	00	01	10	11
名称	NONE	LOW	-	CUSTOM
説明	なし、障害保護禁止	全ピンをLowに駆動	(予約)	全ピンを障害駆動(WEXn.FAULTDRV)と障害出力(WEXn.FAULTOUT)のレジスタによって定義された設定に駆動

14. 事象制御A(EVCTRLA)レジスタを使ってWEXの事象入力Aを許可してください。

```
WEX0.EVCTRLA = WEX_FAULTEI_bm;
```

図5-10. WEXのEVCTRLAレジスタ

ビット	7	6	5	4	3	2	1	0	
					FILTER2~0			BLANK	FAULTEI
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	0	0	0	

- ビット0 - FAULTEI : 障害事象入力許可 (Fault Event Input Enable)

このビットへの'1'書き込みは障害条件用の起動元として事象入力A/B/Cを許可します。

注: 障害事象入力は波形を駆動するタイマ/カウンタが許可される前は考慮されません。

15. 障害割り込み(障害検出)を許可してWEXのISR中に障害フラグを解除(0)してください。

```
WEX0.INTCTRL = WEX_FAULTDET_bm;
```

```
/* WEX将棋割り込み処理ルーチン(ISR) */
ISR(WEX0_FDVEVA_vect)
{
    /* 割り込み要求フラグと障害事象フラグを解除(0) */
    WEX0.INTFLAGS = WEX_FDVEVA_bm | WEX_FAULTDET_bm;
}
```

図5-11. WEXのINTCTRLとINTFLAGSのレジスタ

ビット	7	6	5	4	3	2	1	0
								FAULTDET
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0 – FAULTDET : 障害検出割り込み許可 (Fault Detection Interrupt Enable)**

このビットへの'1'書き込みは障害検出割り込みを許可します。TCE0割り込みとの共存はありません。

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

● **ビット2 – FDFEVA : 事象入力A障害検出フラグ (Fault Detection Flag Event input A)**

このビットは事象入力Aでの障害検出で設定(1)されます。このビットはこの位置に'1'を書くことによって解除(0)されます。

● **ビット0 – FAULTDET : 障害検出割り込み要求フラグ (Fault Detection interrupt Flag)**

このビットは障害検出の正端で設定(1)されます。このビットはこの位置に'1'を書くことによって解除(0)されます。

16. チャンネル0で事象を生成するように事象システム(EVSYN)を構成設定してください。チャンネル0での事象生成部の使用部にWEXを構成設定してください。ソフトウェア事象を使って障害を生成してください。障害を止めるため、EVSYNからの障害生成部チャンネルをOFFにしてソフトウェア指令を使って障害状況を解消してください。

```
/* チャンネル0での事象生成部の使用部としてWEX0を設定、これはこの例ではソフトウェア事象です。 */
EVSYN.USERWEXA = EVSYN_USER_CHANNEL0_gc;
```

```
/* ソフトウェア障害作成、ロジアナで見えるため主繰り返しで繰り返し。これはソフトウェア指令を使って生成された事象です。 */
EVSYN.SWEVENTA = EVSYN_SWEVENTA_CHO_gc;
```

```
/* ソフトウェア指令を使って障害条件を解消 */
WEX0.CTRLC = WEX_CMD_FAULTCLR_gc;
```

17. 出力上書き許可(OUTOVEN)レジスタを使って8つのPWM信号を生成するようにWEXの出力を許可してください。

```
/* WEXの出力を許可 */
WEX0.OUTOVEN = WEX_OUTOVEN0_bm | WEX_OUTOVEN1_bm | WEX_OUTOVEN2_bm | WEX_OUTOVEN3_bm |
WEX_OUTOVEN4_bm | WEX_OUTOVEN5_bm | WEX_OUTOVEN6_bm | WEX_OUTOVEN7_bm;
```

図5-12. WEXのOUTOVENレジスタ

ビット	7	6	5	4	3	2	1	0
	OUTOVEN7	OUTOVEN6	OUTOVEN5	OUTOVEN4	OUTOVEN3	OUTOVEN2	OUTOVEN1	OUTOVEN0
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 – OUTOVENn : 出力上書き許可 (Output Override Enable)**

このビット領域は対応するポート出力レジスタ(即ち、波形出力と1対1の関係)の上書きを許可します。

このレジスタは障害保護が許可された(障害制御(WEXn.FAULTCTRL)レジスタの障害検出活動(FDACT)ビット領域がNONE以外の場合に解除(0)することができません。

18. その後、ポートの方向設定(DIRSET)レジスタの対応するビットに'1'を書くことによってポートAの0~7(PA0~7)ピンを出力として設定してください。

```
PORTA.DIRSET = PIN0_bm | PIN1_bm | PIN2_bm | PIN3_bm | PIN4_bm | PIN5_bm | PIN6_bm | PIN7_bm;
```

19. 全ての単位部が構成設定された後、全体割り込みを許可してください。

```
/* 全体割り込み許可 */
sei();
```


5.2. MCC Melody実装

MPLABコード構成部、MCC Melodyを使ってこのプロジェクトを生成するには次の手順に従ってください(MCCクラスは支援されません)。

1. AVR16EB23に対してMPLAB® X IDEプロジェクトを作成してください。
2. ツールバーからMCCを開いてください(ここでMCCプラグイン インストールでのより多くの情報を見つけてください)。
3. **MCC Content Manager**(MCC内容管理)ウィザードで**MCC Melody**を選び、その後に**Finish**をクリックしてください。
4. **Project Resources**(プロジェクト資源)⇒**System**(システム)⇒**Interrupt Manager**(割り込み管理部)へ行ってください。**Global Interrupt Enable**(全体割り込み許可)鈕を切り替えてください。
5. **Project Resources**(プロジェクト資源)⇒**System**(システム)⇒**CLKCTRL**へ行ってください。**Prescaler Enable**(前置分周器許可)鈕を禁止してください。
6. **Device Resources**(デバイス資源)⇒**Drivers**(ドライバ)⇒**Timer**(計時器)でTCE単位部を追加し、その後に以下の構成設定を行ってください。
 - **Enable Timer**(計時器許可)：おそらく既定で許可されます。そうでなければ、鈕を切り替えてください(許可なら青になります)。
 - **Clock Divider**(クロック分周器)：(既定で)**System clock**(システム クロック)、(分周器は1 - System clockでなければなりません)。
 - **Waveform Generation Mode**(波形生成動作)：単一傾斜PWM動作(**SINGLESLOPE**)
 - **Requested Period [s]**(要求周期)：0.00005
 - **Waveform Output n**(波形出力n)：波形出力0,1,2,3に対する許可列で枠をチェックしてください。
 - **Generate ISR**(ISR生成)：鈕を切り替えてください(許可なら青になります)。
 - **Enable Compare 0 Interrupt**(比較0割り込み許可)：鈕を切り替えてください(許可なら青になります)。
 - **Enable Compare 1 Interrupt**(比較1割り込み許可)：鈕を切り替えてください(許可なら青になります)。
 - **Enable Compare 2 Interrupt**(比較2割り込み許可)：鈕を切り替えてください(許可なら青になります)。
 - **Enable Compare 3 Interrupt**(比較3割り込み許可)：鈕を切り替えてください(許可なら青になります)。
7. **Device Resources**(デバイス資源)⇒**Drivers**(ドライバ)でWEX単位部を追加し、その後に以下の構成設定を行ってください。
 - **Module Enable**(単位部許可)：既定で許可されなければなりません。そうでなければ、鈕を切り替えてください(許可なら青になります)。
 - **Input Matrix**(入力配列)：**Direct**(直接)
 - **Update Source**(更新供給元)：TCE (出力信号に対する更新条件はTCEによって指示されます。)
 - **Override Settings**(上書き設定)：波形出力0～7に対する**Output Enable**(出力許可)列で全ての枠をチェックしてください。
 - **Dead-time Inersion Channel 0 Enable**(沈黙時間挿入チャンネル0許可)：鈕を切り替えてください(許可なら青になります)。
 - **Dead-time Inersion Channel 1 Enable**(沈黙時間挿入チャンネル1許可)：鈕を切り替えてください(許可なら青になります)。
 - **Dead-time Inersion Channel 2 Enable**(沈黙時間挿入チャンネル2許可)：鈕を切り替えてください(許可なら青になります)。
 - **Dead-time Inersion Channel 3 Enable**(沈黙時間挿入チャンネル3許可)：鈕を切り替えてください(許可なら青になります)。
 - **Requested Dead-time Low Side (μs)**(Low側要求沈黙時間)：0.150
 - **Requested Dead-time High Side (μs)**(High側要求沈黙時間)：0.250
 - **Fault Event Input A**(障害事象入力A)：鈕を切り替えてください(許可なら青になります)。
 - **Fault Enable**(障害許可)：鈕を切り替えてください(許可なら青になります)。
 - **Fault Interrupt Enable**(障害割り込み許可)：鈕を切り替えてください(許可なら青になります)。
 - **Fault Detection Restart Mode**(障害検出再開動作)：Cycle-by Cycle(周回毎)
 - **Fault Detection Action**(障害検出活動)：Low
8. **Device Resources**(デバイス資源)⇒**Drivers**(ドライバ)でEVSYS単位部を追加し、その後に次のように構成設定を行ってください。
 - **CHANNELS**(チャンネル)：CHANNEL0(チャンネル0)
 - **USERS**(使用部)：WEXA (CHANNELSタブでCHANNEL0長方形を選び、USERSタブのWEXA長方形へドラッグ&ドロップしてください。)
9. **Pin Grid View**(ピン格子表示)タブのPORTAでTCE_WO0～7ピンが出力として固定化されているかを調べてください。**Waveform Output n**(波形出力n)で**Enable**(許可)列での枠がチェックされていると、ピンも固定化されます。PORTを変更するには、**Pin Grid View**で別のPORTからピンをクリックしてください。
10. **Project Resources**(プロジェクト資源)タブで、指定したドライバと構成設定の全てをMCCが生成するように**Generate**(生成)鈕をクリックしてください。
11. 下で示されるように**main.c**ファイルを編集してください。
インクルード ファイル部に次のように追加してください。

```
#include "mcc_generated_files/system/system.h"
#include <util/delay.h>
```


次のようにマクロ定義を追加してください。

```
#define TCE_PERIOD                (0x3E8)
#define MAX_DUTY_CYCLE            (0x3DE)
```

割り込み処理ルーチン(ISR)呼び戻し関数を追加してください。

```
/* ISRルーチンで呼ばれる呼び戻し関数 */
void UserCallback_CMP0(void)
{
    static uint16_t duty_cycle = 0;

    /* 割り込みでデューティサイクル更新 */
    duty_cycle += 5;
    if(duty_cycle >= MAX_DUTY_CYCLE)
        duty_cycle = 0;
    TCE0_PWM_BufferedDutyCycle0Set(duty_cycle);
}

/* ISRルーチンで呼ばれる呼び戻し関数 */
void UserCallback_CMP1(void)
{
    static uint16_t duty_cycle = 0;

    /* 割り込みでデューティサイクル更新 */
    duty_cycle += 15;
    if(duty_cycle >= MAX_DUTY_CYCLE)
        duty_cycle = 0;
    TCE0_PWM_BufferedDutyCycle1Set(duty_cycle);
}

/* ISRルーチンで呼ばれる呼び戻し関数 */
void UserCallback_CMP2(void)
{
    static uint16_t duty_cycle = 0;

    /* 割り込みでデューティサイクル更新 */
    duty_cycle += 25;
    if(duty_cycle >= MAX_DUTY_CYCLE)
        duty_cycle = 0;
    TCE0_PWM_BufferedDutyCycle2Set(duty_cycle);
}

/* ISRルーチンで呼ばれる呼び戻し関数 */
void UserCallback_CMP3(void)
{
    static uint16_t duty_cycle = 0;

    /* 割り込みでデューティサイクル更新 */
    duty_cycle += 35;
    if(duty_cycle >= MAX_DUTY_CYCLE)
        duty_cycle = 0;
    TCE0_PWM_BufferedDutyCycle3Set(duty_cycle);
}
```

次のように関数を追加してください。

```
void Create_Fault(void)
{
    /* 障害作成、ロジックアナライザで見るため、主繰り返しで繰り返し。これはソフトウェア指令を使って生成された事象です。 */
    EVSYS_SoftwareEventASet(EVSYS_SWEVENTA_CHO_gc);
}

void Clear_Fault(void)
```

```

{
  /* ソフトウェア指令を使って障害条件を解消 */
  WEX0_SoftwareCommand(WEX_CMD_FAULTCLR_gc);
}

```

次のように主関数を編集してください。

```

int main(void)
{
  SYSTEM_Initialize();

  TCE0_Compare0CallbackRegister(UserCallback_CMP0);
  TCE0_Compare1CallbackRegister(UserCallback_CMP1);
  TCE0_Compare2CallbackRegister(UserCallback_CMP2);
  TCE0_Compare3CallbackRegister(UserCallback_CMP3);

  while(1)
  {
    Create_Fault();
    _delay_us(250);
    Clear_Fault();
    _delay_us(250);
  }
}

```

12. 今や、プロジェクトはMPLAB X IDEを使って構築して走行することができます。走行中、比較レジスタに対する尺度値が10ms毎に変更され、それによってデューティサイクルの範囲が変更されます。

5.3. 結果

下の図は生成した8つの補完PWM信号、0~100%へ増加するデューティサイクル、障害事象中に何が起こるか、沈黙時間期間の様子を実演するためにいくつかのロジックアナライザが捕獲が示されます。

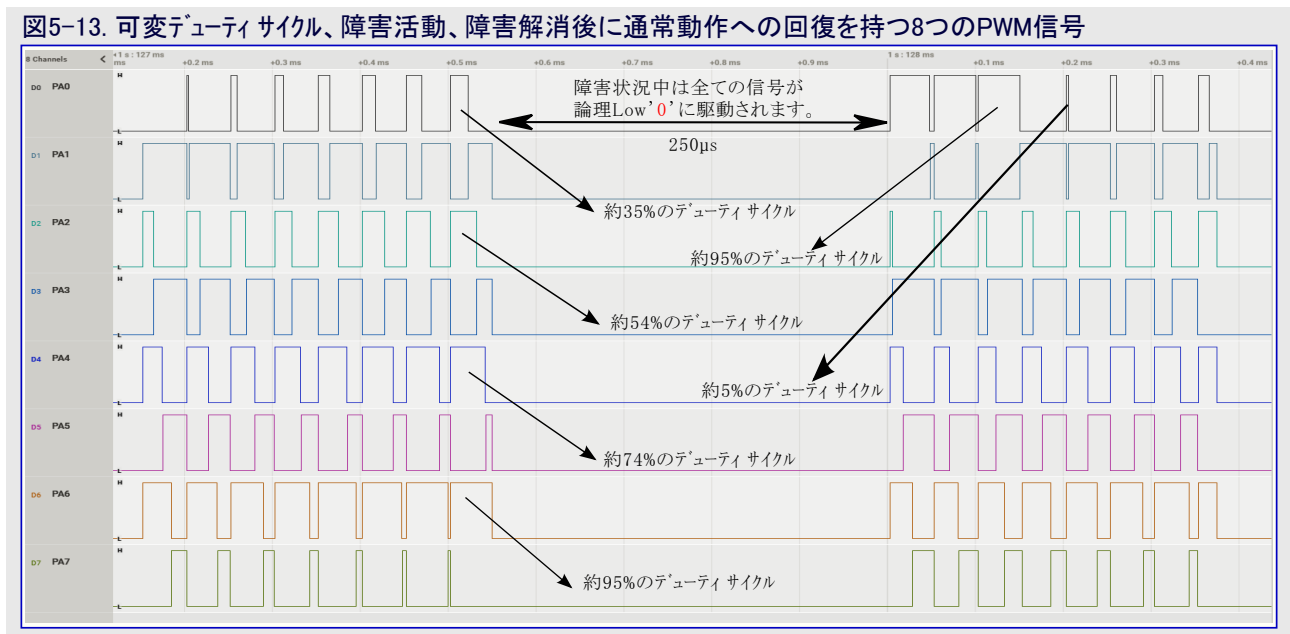


図5-14. 強調された沈黙時間期間

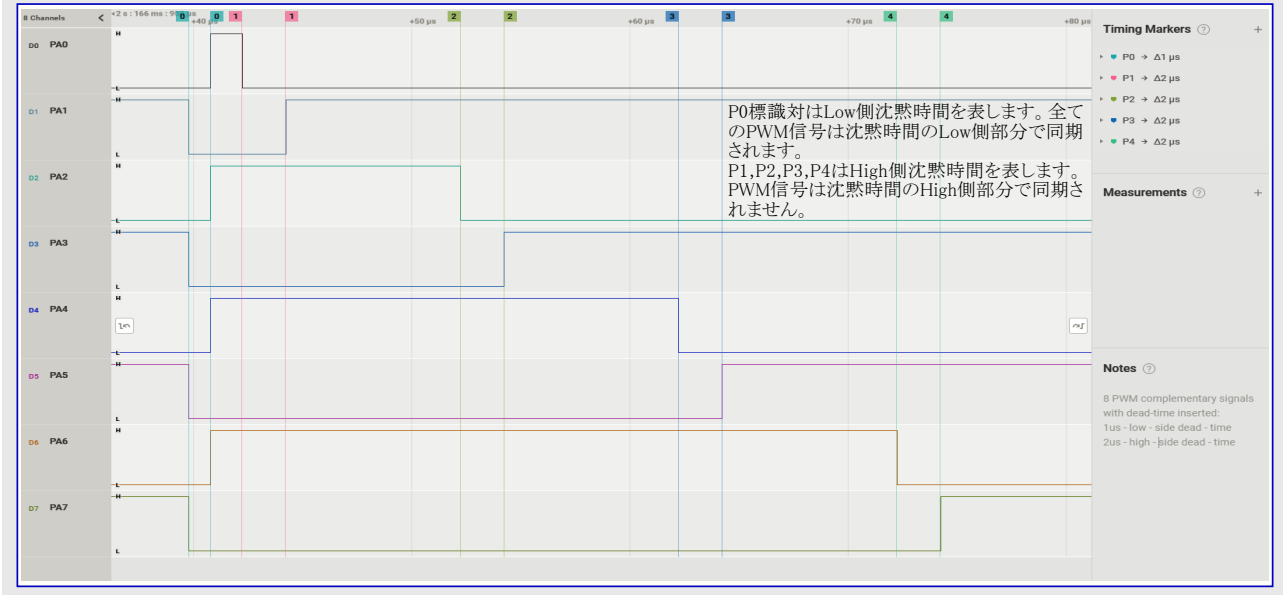
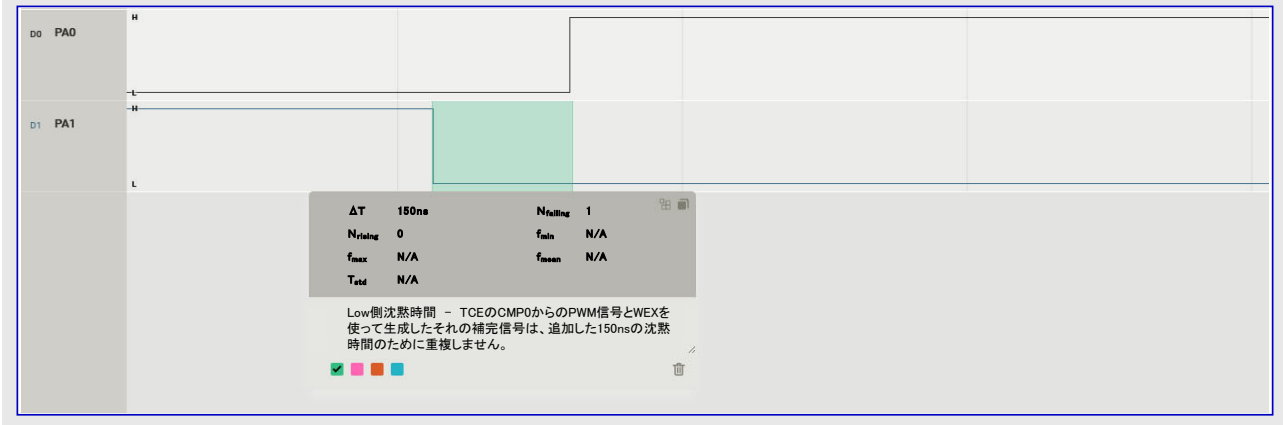


図5-15. Low側沈黙時間期間でより詳細に見学



本章で記述されたのと同じ機能性を持つAVR16EB32用素のコード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

本章で記述されたのと同じ機能性を持つMCCで生成したAVR16EB32用コード例をここで見つけてください。



MPLAB DISCOVERでコード例を見るにはクリックしてください。

6. 参照

TCE動作形態についてより多くの情報に関しては以下のリンクを使ってください。

1. [AVR16EB32製品頁](#)
2. [AVR16EB32 Curiosity Nano評価キット](#)
3. [AVR16EB16/20/28/32 AVR® EB系統データシート](#)

7. 改訂履歴

文書改訂	日付	注釈
A	2023年11月	初版文書公開
B	2024年8月	MULFAC1,0のレジスタ表で画像を更新

Microchip情報

Microchipウェブ サイト

Microchipはwww.microchip.com/で当社のウェブ サイト経由でのオンライン支援を提供します。このウェブ サイトはお客様がファイルや情報を容易に利用可能にするのに使われます。利用可能な情報のいくつかは以下を含みます。

- ・ **製品支援** – データシートと障害情報、応用記述と試供プログラム、設計資源、使用者の手引きとハードウェア支援資料、最新ソフトウェア配布と保管されたソフトウェア
- ・ **一般的な技術支援** – 良くある質問(FAQ)、技術支援要求、オンライン検討グループ、Microchip設計協力課程会員一覧
- ・ **Microchipの事業** – 製品選択器と注文の手引き、最新Microchip報道発表、セミナーとイベントの一覧、Microchip営業所の一覧、代理店と代表する工場

製品変更通知サービス

Microchipの製品変更通知サービスはMicrochip製品を最新に保つのに役立ちます。加入者は指定した製品系統や興味のある開発ツールに関連する変更、更新、改訂、障害情報がある場合に必ず電子メール通知を受け取ります。

登録するにはwww.microchip.com/pcnへ行って登録指示に従ってください。

お客様支援

Microchip製品の使用者は以下のいくつかのチャネルを通して支援を受け取ることができます。

- ・ 代理店または販売会社
- ・ 最寄りの営業所
- ・ 組み込み解決技術者(ESE:Embedded Solutions Engineer)
- ・ 技術支援

お客様は支援に関してこれらの代理店、販売会社、またはESEに連絡を取るべきです。最寄りの営業所もお客様の手助けに利用できます。営業所と位置の一覧はこの資料の後ろに含まれます。

技術支援はwww.microchip.com/supportでのウェブ サイトを通して利用できます。

Microchipデバイス コード保護機能

Microchip製品での以下のコード保護機能の詳細に注意してください。

- ・ Microchip製品はそれら特定のMicrochipデータシートに含まれる仕様に合致します。
- ・ Microchipは動作仕様内で意図した方法と通常条件下で使われる時に、その製品系統が安全であると考えます。
- ・ Microchipはその知的所有権を尊重し、積極的に保護します。Microchip製品のコード保護機能を侵害する試みは固く禁じられ、デジタル ミレニアム著作権法に違反するかもしれません。
- ・ Microchipや他のどの半導体製造業者もそのコードの安全を保証することはできません。コード保護は製品が”破ることができない”ことを当社が保証すると言うことを意味しません。コード保護は常に進化しています。Microchipは当社製品のコード保護機能を継続的に改善することを約束します。

法的通知

この刊行物と契約での情報は設計、試験、応用とのMicrochip製品の統合を含め、Microchip製品でだけ使えます。他の何れの方法でのこの情報の使用はこれらの条件に違反します。デバイス応用などに関する情報は皆さまの便宜のためにだけ提供され、更新によって取り換えられるかもしれません。皆さまの応用が皆さまの仕様に合致するのを保証するのは皆さまの責任です。追加支援については最寄りのMicrochip営業所にお問い合わせ頂くか、www.microchip.com/en-us/support/design-help/client-support-servicesで追加支援を得てください。

この情報はMicrochipによって「現状そのまま」で提供されます。Microchipは非侵害、商品性、特定目的に対する適合性の何れの黙示的保証やその条件、品質、性能に関する保証を含め、明示的にも黙示的にもその情報に関連して書面または表記された書面または黙示の如何なる表明や保証もしません。

如何なる場合においても、Microchipは情報またはその使用に関連するあらゆる種類の間接的、特別的、懲罰的、偶発的または結果的な損失、損害、費用または経費に対して責任を負わないものとします。法律で認められている最大限の範囲で、情報またはその使用に関連する全ての請求に対するMicrochipの全責任は、もしあれば、情報のためにMicrochipへ直接支払った料金を超えないものとします。生命維持や安全応用でのMicrochipデバイスの使用は完全に購入者の危険性で、購入者はそのような使用に起因する全ての損害、請求、訴訟、費用からMicrochipを擁護し、補償し、免責することに同意します。他に言及されない限り、Microchipのどの知的財産権下でも暗黙的または違う方法で許認可は譲渡されません。

商標

Microchipの名前とロゴ、Microchip、Adaptec、AVR、AVR、AVR Freaks、BesTime、BitCloud、CryptoMemory、CryptoRF、dsPIC、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maxStylus、maxTouch、MediaLB、megaAVR、Microsemi、Microsemi、MOST、MOST、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST、Super Flash、Symmetricom、SyncServer、Tachyon、TimeSource、tinyAVR、UNI/O、Vectron、XMEGAは米国と他の国に於けるMicrochip Technology Incorporatedの登録商標です。

AgileSwitch、ClockWorks、The Embedded Control Solutions Company、EtherSynch、Flashtec、Hyper Speed Control、HyperLight Load、IntelliMOS、Libero、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus、Quiet-Wire、SmartFusion、SyncWorld、TimeCesium、TimeHub、TimePictra、TimeProvider、ZLは米国に於けるMicrochip Technology Incorporatedの登録商標です。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、Augmented Switching、BlueSky、BodyCom、Clockstudio、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、EyeOpen、GridTime、IdealBridge、IGaT、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、IntelliMOS、Inter-Chip Connectivity、JitterBlocker、Knob-on-Display、MarginLink、maxCrypto、maxView、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified、MPLIB、MPLINK、mSiC、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICKit、PICtail、Power MOS IV、Power MOS 7、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、RTAX、RTG4、SAM-ICE、Serial Quad I/O、simpleMAP、SimpliPHY、SmartBuffer、SmartHLS、SMART-I.S.、storClad、SQI、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、Trusted Time、TSHARC、Turing、USBCheck、VariSense、Vector Blox、VeriPHY、ViewSpan、WiperLock、XpressConnect、ZENAは米国と他の国に於けるMicrochip Technology Incorporatedの商標です。

SQTPは米国に於けるMicrochip Technology Incorporatedの役務標章です。

Adaptec、Frequency on Demand、Silicon Storage Technology、Symmcomは他の国に於けるMicrochip Technology Inc.の登録商標です。

GestICは他の国に於けるMicrochip Technology Inc.の子会社であるMicrochip Technology Germany II GmbH & Co. KGの登録商標です。

ここで言及した以外の全ての商標はそれら各々の会社の所有物です。

© 2024年、Microchip Technology Incorporatedとその子会社、不許複製

品質管理システム

Microchipの品質管理システムに関する情報についてはwww.microchip.com/qualityを訪ねてください。

日本語© HERO 2024.

本技術概説はMicrochipのTB3339技術概説(DS90003339B-2024年8月)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。

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。

世界的な販売とサービス

米国	亜細亜/太平洋	亜細亜/太平洋	欧州
本社 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 技術支援: www.microchip.com/support ウェブアドレス: www.microchip.com	オーストラリア - シドニー Tel: 61-2-9868-6733 中国 - 北京 Tel: 86-10-8569-7000 中国 - 成都 Tel: 86-28-8665-5511 中国 - 重慶 Tel: 86-23-8980-9588 中国 - 東莞 Tel: 86-769-8702-9880 中国 - 広州 Tel: 86-20-8755-8029 中国 - 杭州 Tel: 86-571-8792-8115 中国 - 香港特别行政区 Tel: 852-2943-5100 中国 - 南京 Tel: 86-25-8473-2460 中国 - 青島 Tel: 86-532-8502-7355 中国 - 上海 Tel: 86-21-3326-8000 中国 - 瀋陽 Tel: 86-24-2334-2829 中国 - 深圳 Tel: 86-755-8864-2200 中国 - 蘇州 Tel: 86-186-6233-1526 中国 - 武漢 Tel: 86-27-5980-5300 中国 - 西安 Tel: 86-29-8833-7252 中国 - 廈門 Tel: 86-592-2388138 中国 - 珠海 Tel: 86-756-3210040	インド - ハンガロール Tel: 91-80-3090-4444 インド - ニューデリー Tel: 91-11-4160-8631 インド - プネー Tel: 91-20-4121-0141 日本 - 大阪 Tel: 81-6-6152-7160 日本 - 東京 Tel: 81-3-6880-3770 韓国 - 大邱 Tel: 82-53-744-4301 韓国 - ソウル Tel: 82-2-554-7200 マレーシア - クアラルンプール Tel: 60-3-7651-7906 マレーシア - ペナン Tel: 60-4-227-8870 フィリピン - マニラ Tel: 63-2-634-9065 シンガポール Tel: 65-6334-8870 台湾 - 新竹 Tel: 886-3-577-8366 台湾 - 高雄 Tel: 886-7-213-7830 台湾 - 台北 Tel: 886-2-2508-8600 タイ - バンコク Tel: 66-2-694-1351 ベトナム - ホーチミン Tel: 84-28-5448-2100	オーストラリア - ウェルズ Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 デンマーク - コペンハーゲン Tel: 45-4485-5910 Fax: 45-4485-2829 フィンランド - エスポー Tel: 358-9-4520-820 フランス - パリ Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 ドイツ - ガルピング Tel: 49-8931-9700 ドイツ - ハーン Tel: 49-2129-3766400 ドイツ - ハイムブロン Tel: 49-7131-72400 ドイツ - カールスルーエ Tel: 49-721-625370 ドイツ - ミュンヘン Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 ドイツ - ローゼンハイム Tel: 49-8031-354-560 イスラエル - ホト ハシャロン Tel: 972-9-775-5100 イタリア - ミラノ Tel: 39-0331-742611 Fax: 39-0331-466781 イタリア - ハトバ Tel: 39-049-7625286 オランダ - テルネン Tel: 31-416-690399 Fax: 31-416-690340 ノルウェー - トロンハイム Tel: 47-72884388 ポーランド - ワルシャワ Tel: 48-22-3325737 ルーマニア - ブカレスト Tel: 40-21-407-87-50 スペイン - マドリッド Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 スウェーデン - イェテボリ Tel: 46-31-704-60-40 スウェーデン - ストックホルム Tel: 46-8-5090-4654 イギリス - ウォーキングム Tel: 44-118-921-5800 Fax: 44-118-921-5820
アトランタ Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 オースチン TX Tel: 512-257-3370 ホストン Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 シカゴ Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 ダラス Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 デトロイト Novi, MI Tel: 248-848-4000 ヒューストン TX Tel: 281-894-5983 インディアナポリス Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 ロサンゼルス Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 ローリー NC Tel: 919-844-7510 ニューヨーク NY Tel: 631-435-6000 サンホセ CA Tel: 408-735-9110 Tel: 408-436-4270 カナダ - トロント Tel: 905-695-1980 Fax: 905-695-2078			