
DACでの開始に際して

序説

著者: Victor Berzan, Microchip Technology Inc.

デジタル⇒アナログ変換器(D/A変換器:DAC)はデータ(DAC.DATA)レジスタに書かれたデジタル値をアナログ電圧に変換します。この出力は物理的なピンに接続することや内部的に使うことができます。変換範囲はGNDと選ばれた内部参照基準電圧(VREF)間です。

この技術概要はtinyAVR® 1系マイクロコントローラ(MCU)でD/A変換器がどう動くかを記述します。これは以下の使用事例を網羅します。

- **一定のアナログ信号を生成:**

DACを初期化、電圧参照基準を設定、特定の一定電圧を出力するようにDACを設定します。

- **正弦波信号を生成:**

DACを初期化、電圧参照基準を設定、正弦波の標本を繰り返して出力します。

- **ADCで内部的にDACを読み込み:**

DACとADCを初期化、電圧参照基準を設定、DACを読むようにADCを設定、DAC出力を増して各段階に対してADCでそれを読み込みます。

- **AC用負入力としてDACを使用:**

ADCとACを初期化、DACからACの負入力を取るようにACを設定、特定の値を出力するようにDACを設定します。ACはDAC電圧とACの正入力ピン上の電圧を比較し、比較結果に従って出力ピンをHighまたはLowに設定します。

注: コード例はATtiny817 Xplained Mini (ATTINY817-XMINI)で開発されました。

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
1.1. tinyAVR® 1系統	3
2. 概要	3
3. 一定のアナログ信号を生成	3
4. 正弦波信号を生成	5
5. ADCで内部的にDACを読み込み	5
6. AC用負入力としてDACを使用	7
7. 参照	8
8. 追補	8
Microchipウェブ サイト	14
お客様への変更通知サービス	14
お客様支援	14
Microchipデバイス コード保護機能	14
法的通知	14
商標	15
DNVによって認証された品質管理システム	15
世界的な販売とサービス	16

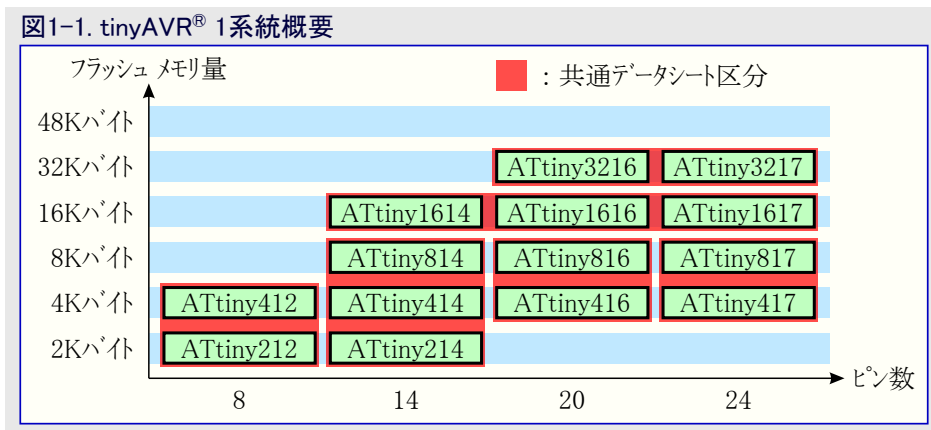
1. 関連デバイス

本章はこの資料に関連するデバイスを一覧にします。

1.1. tinyAVR[®] 1系統

下図はピン配置変種とメモリ量を展開してtinyAVR[®] 1系統デバイスを示します。

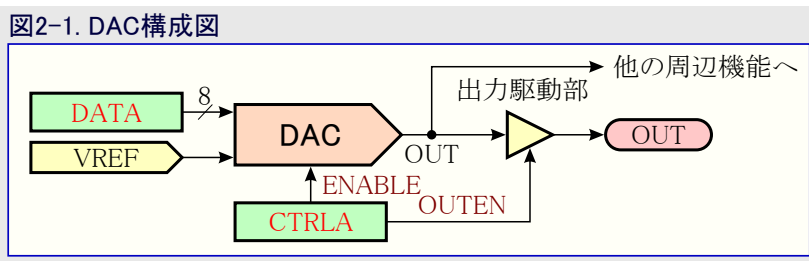
- これらのデバイスがピン互換で同じまたはより多くの機能を提供するため、垂直上方向移植はコード変更なしに可能です。下方向移植はより少ない利用可能ないくつかの周辺機能の実体のためにコード変更が必要かもしれません。
- 左への水平方向移植はピン数、従って利用可能な機能を減らします。



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の量を持ちます。

2. 概要

DACは毎秒350,000採取(350ksp/s)の変換能力の8ビット抵抗列型DACが特徴です。DACは変換用の上限として内部基準電圧(VREF)を使います。DACは5kΩと/または30pFの負荷を駆動することができる高い駆動能力を持つ1つの連続時間出力を持ちます。DAC変換はデータ変換(DAC.DAT)レジスタに書くことによって応用から開始することができます。



3. 一定のアナログ信号を生成

DACは一定のアナログ信号を生成するのに使うことができます。正の参照基準として基準電圧(VREF)周辺機能の出力を使います。DAC出力は0Vから $(255 \times VREF) / 256$ の範囲です。

VREFは予め定義された値の一覧から選ぶことができます。

図3-1. VREF.CTRLA.DAC0REFSEL値と関連するVREF電圧

ビット	7	6	5	4	3	2	1	0
	ADC0REFSEL2~0					DAC0REFSEL2~0		
アクセス種別	R	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2~0 - DAC0REFSEL2~0 : AC0とDAC0基準選択 (DAC0 and AC0 Reference Select)

これらのビットはD/A変換器(DAC0)とアナログ比較器(AC0)用の基準電圧を選びます。

値	000	001	010	011	100	101	110	111
説明	0.55V	1.1V	2.5V	4.3V	1.5V	(予約)		

最も広い変化範囲を持つように4.34Vが選ばれました。

```
VREF_CTRLA |= VREF_DACOREFSEL_4V34_gc;
VREF_CTRLB |= VREF_DACOREFEN_bm;
```

VREF許可後に25 μ sの遅延が推奨されます。

図3-2. 内部電圧参照基準特性

シンボル	項目	条件	Min	Typ	Max	単位
t_{start}	始動時間		-	25	-	μ s
VDDINT055V	INT055Vに対する電源電圧範囲		1.8	-	5.5	V
VDDINT11V	INT11Vに対する電源電圧範囲		1.8	-	5.5	
VDDINT15V	INT15Vに対する電源電圧範囲		1.9	-	5.5	
VDDINT25V	INT25Vに対する電源電圧範囲		2.9	-	5.5	
VDDINT43V	INT43Vに対する電源電圧範囲		4.75	-	5.5	

```
_delay_us(25);
```

DAC出力は他の周辺機能によって内部的に使う、または出力ピンに繋げることができます。ATtiny817についてはDAC出力がPA6ピンに接続されます(図3-3をご覧ください)。

図3-3. ポート機能多重化

24ビット VQFN	ピン名 (注1,2)	その他/特殊	ADC0	PTC (注3)	AC0	DAC0	USART0
23	PA0	RESET/UPDI	AIN0				
24	PA1	BREAK	AIN1				TxD
1	PA2	EVOUT0	AIN2				RxD
2	PA3	EXTCLK	AIN3				XCK
3	GND						
4	VDD						
5	PA4		AIN4	X0/Y0			XDIR
6	PA5		AIN5	X1/Y1	OUT		
7	PA6		AIN6	X2/Y2	AINN0	OUT	
8	PA7		AIN7	X3/Y3	AINP0		
9	PB7						

DAC出力ピンはこれの負荷を減らすため、禁止されたデジタル入力緩衝部とプルアップ抵抗を持つことが必要です。

```
PORTA.PIN6CTRL &= ~PORT_ISC_gm;
PORTA.PIN6CTRL |= PORT_ISC_INPUT_DISABLE_gc;
PORTA.PIN6CTRL &= ~PORT_PULLUPEN_bm;
```

この特別な例でDACに対して望まれる出力は0.5425Vです。これを達成するため以下の式が適用されます。

$$DATA = VOUT \times 256 / VREF = 0.5425V \times 256 / 4.34V = 32 = 0x20$$

初期化でDAC0.DATAレジスタを書くことは任意選択ですが、初めからDACに指定した電圧を出力させることは有用で有り得ます。

```
DAC0.DATA = 0x20;
```

DAC、出力緩衝部、スタンバイで走行を許可します。

```
DAC0.CTRLA = DAC_ENABLE_bm | DAC_OUTEN_bm | DAC_RUNSTDBY_bm;
```



重要: スタンバイで走行動作が許可された場合、DACはMCUがスタンバイ休止動作の時に走行を続けます。

変換の開始

DACが許可(DAC0.CTRLAのENABLE=1)されると、データ(DAC0.DATA)レジスタが書かれると直ぐに変換が始まります。

```
DAC0.DATA = 0x20;
```

変換後、出力はDACが走行している限り、次の変換までDATA \times VREF/256の値を保ちます。VREF選択でのどの変更も(許可されて走行しているなら、)直ちにDAC出力を変更します。



GitHubでコード例を見てください。
貯蔵庫を閲覧するにはクリックしてください。



助言: 完全なコード例は「[追補](#)」章でも利用可能です。

4. 正弦波信号を生成

DACは正弦波信号を生成するのに使うことができます。

VREFとDACの初期化後、出力値は単にDAC0.DATAレジスタに書くことによって変更することができます。

正弦波を生成するため、最初に波形標本が計算されて緩衝部に格納されます。

```
for(uint16_t i = 0; i < SINE_WAVE_STEPS; i++)
{
    sineWave[i] = SINE_DC_OFFSET + SINE_AMPLITUDE * sin(i * M_2PI / SINE_WAVE_STEPS);
}
```

その後それらは指定間隔で繰り返してDAC0.DATAに書かれます。

```
while (1)
{
    DAC0.DATA = sineWave[i++];
    i = i % SINE_WAVE_STEPS;
    _delay_us(STEP_DELAY_MICROS);
}
```



GitHubでコード例を見てください。
貯蔵庫を閲覧するにはクリックしてください。

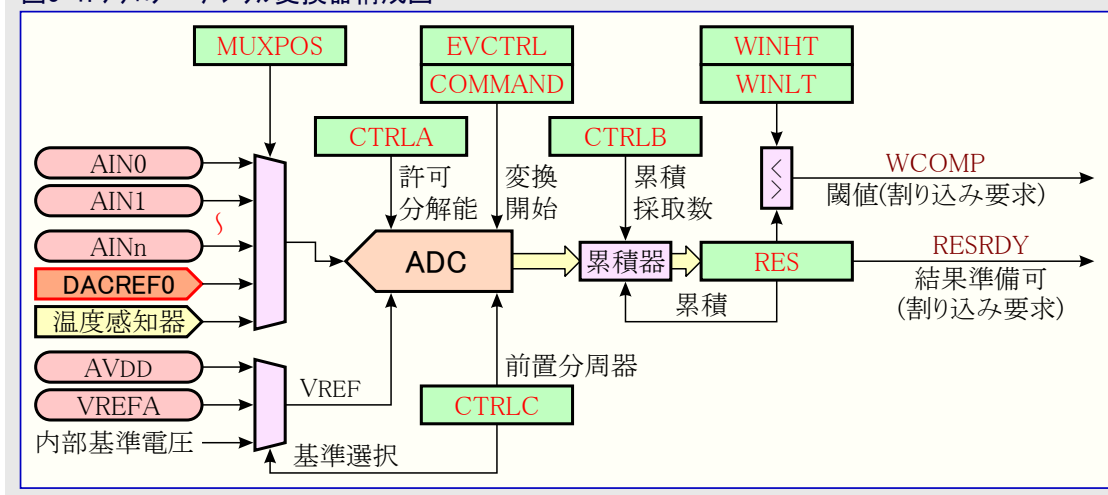


助言: 完全なコード例は「[追補](#)」章でも利用可能です。

5. ADCで内部的にDACを読み込み

図5-1.で見られるように、ADC入力はDACの出力に接続することができます。

図5-1. アナログ⇒デジタル変換器構成図



VREFからの正参照基準でADCを使う場合、対応する参照基準選択もまた構成設定されなければなりません。

図5-2. VREF.CTRLAレジスタ

ビット	7	6	5	4	3	2	1	0
	ADC0REFSEL2~0				DAC0REFSEL2~0			
アクセス種別	R	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6~4 – ADC0REFSEL2~0 : ADC0基準選択 (ADC0 Reference Select)

これらのビットはA/D変換器(ADC0)用の基準電圧を選びます。

値	000	001	010	011	100	101	110	111
説明	0.55V	1.1V	2.5V	4.3V	1.5V	(予約)		

● ビット2~0 – DAC0REFSEL2~0 : AC0とDAC0基準選択 (DAC0 and AC0 Reference Select)

これらのビットはD/A変換器(DAC0)とアナログ比較器(AC0)用の基準電圧を選びます。

値	000	001	010	011	100	101	110	111
説明	0.55V	1.1V	2.5V	4.3V	1.5V	(予約)		

完全なVREF初期化は今やこのように見えます。

```
VREF_CTRLA = VREF_DAC0REFSEL_4V34_gc;
VREF_CTRLB = VREF_DAC0REFEN_bm;
VREF_CTRLA = VREF_ADC0REFSEL_4V34_gc;
VREF_CTRLB = VREF_ADC0REFEN_bm;
_delay_us(VREF_STARTUP_MICROS);
```

その後にADCが以下のように初期化されなければなりません。

```
ADC0_CTRLC = ADC_PRESC_DIV4_gc | ADC_REFSEL_INTREF_gc | ADC_SAMPCAP_bm;
ADC0_CTRLA = ADC_ENABLE_bm | ADC_RESSEL_10BIT_gc;
```

ADCでDACを読むにはADCのMUXPOSレジスタが0x1Cに設定されなければなりません。

図5-3. ADC0.MUXPOSでのDAC出力選択

ビット	7	6	5	4	3	2	1	0
	MUXPOS4~0							
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4~0 – MUXPOS4~0 : 多重器正選択 (MUXPOS)

このビット領域はADCにどのシングルエンドアナログ入力接続されるかを選びます。これらのビットが変換中にえられる場合、その変更はこの変換が完了するまで有効ではありません。

値	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011
名称	AIN0	AIN1	AIN2	AIN3	AIN4	AIN5	AIN6	AIN7	AIN8	AIN9	AIN10	AIN11
説明 (アナログ入力ピン)	0	1	2	3	4	5	6	7	8	9	10	11
値	11100		11101					11110		11111		その他
名称	DAC0		INTREF					TEMPSENSE		GND		-
説明	DAC0出力		(VREF周辺機能からの)内部基準電圧					温度感知器		0V (GND)		(予約)

```
ADC0_MUXPOS = ADC_MUXPOS_DAC0_gc;
```

ADC変換はADC0.COMMANDへ対応するビットを書くことによって開始されます。

```
ADC0_COMMAND = ADC_STCONV_bm;
```

変換が終了されると、ハードウェアによってADC0.INTFLAGSのRESRDYが設定(1)されます。

```
while (!(ADC0.INTFLAGS & ADC_RESRDY_bm))
{
    ;
}
```

使用者は'1'を書くことによってこのフラグを解除(0)しなければなりません。

```
ADC0.INTFLAGS = ADC_RESRDY_bm;
```

ADCデータはADC0.RESから読むことができます。

DAC出力は様々な値に設定し、While繰り返し内に於いてADCで読むことができます。

```
while (1)
{
    adcVal = ADC0_read();
    dacVal++;
    DAC0_setVal(dacVal);
}
```



GitHubでコード例を見てください。
貯蔵庫を閲覧するにはクリックしてください。

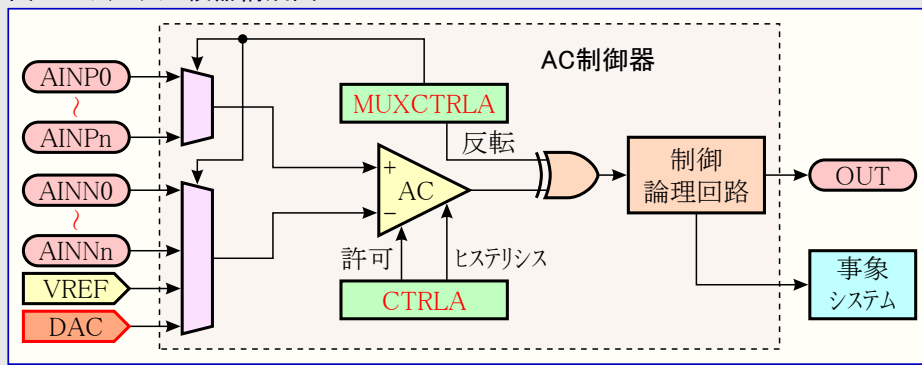


助言: 完全なコード例は「[追補](#)」章でも利用可能です。

6. AC用負入力としてDACを使用

図6-1.で見られるように、ACはDAC出力を負入力として使うことができます。

図6-1. アナログ比較器構成図



DACに使われるVREFはこの特定例で1.5Vに設定されます。

```
VREF.CTRLA |= VREF_DACOREFSEL_1V5_gc;
VREF.CTRLB |= VREF_DACOREFEN_bm;
_delay_us(VREF_STARTUP_MICROS);
```

ACはDACからの負入力とPA7ピンからの正入力を取るように設定されます。

24ビット VQFN	ピン名 (注1,2)	その他/特殊	ADC0	PTC (注3)	AC0	DAC0	USART0
23	PA0	RESET/UPDI	AIN0				
24	PA1	BREAK	AIN1				TxD
1	PA2	EVOUT0	AIN2				RxD
2	PA3	EXTCLK	AIN3				XCK
3	GND						
4	VDD						
5	PA4		AIN4	X0/Y0			XDIR
6	PA5		AIN5	X1/Y1	OUT		
7	PA6		AIN6	X2/Y2	AINN0	OUT	
8	PA7		AIN7	X3/Y3	AINP0		
9	PB7						

```
AC0.MUXCTRLA = AC_MUXNEG_DAC_gc | AC_MUXPOS_PIN0_gc;
```

ACの出力はPA5ピンです。

24ビット VQFN	ピン名 (注1,2)	その他/特殊	ADC0	PTC (注3)	AC0	DAC0	USART0
23	PA0	RESET/UPDI	AIN0				
24	PA1	BREAK	AIN1				TxD
1	PA2	EVOUT0	AIN2				RxD
2	PA3	EXTCLK	AIN3				XCK
3	GND						
4	VDD						
5	PA4		AIN4	X0/Y0			XDIR
6	PA5		AIN5	X1/Y1	OUT		
7	PA6		AIN6	X2/Y2	AINN0	OUT	
8	PA7		AIN7	X3/Y3	AINP0		
9	PB7						

```
AC0.CTRLA = AC_ENABLE_bm | AC_OUTEN_bm | AC_RUNSTDBY_bm | AC_LPMODE_bm;
```

VREF、DAC、AC初期化後、DAC出力はこの特定例に対して1.4Vに設定されます。

```
DAC0_setVal (DAC_DATA_1V4);
```

ACはPA7からの電圧を1.4Vと比較して比較結果に従ってその出力を設定します。



GitHubでコード例を見てください。
貯蔵庫を閲覧するにはクリックしてください。



助言: 完全なコード例は「追補」章でも利用可能です。

7. 参照

1. [ATtiny417/817 - コアから独立した周辺機能とpicoPower®技術を持つAVR®マイクロコントローラ \(DS40001901\)](#)
2. [ATtiny817 Explained Mini基板](#)

8. 追補

例8-1. 一定のアナログ信号を生成するコード例

```
/* 3.33MHz (遅延用に必要) */
#define F_CPU (3333333UL)
/* DAC値 */
#define DAC_EXAMPLE_VALUE (0x20)
/* VREF始動時間 */
#define VREF_STARTUP_MICROS (25)

#include <avr/io.h>
#include <util/delay.h>

void VREF_init(void);
void DAC0_init(void);
void DAC0_setVal(uint8_t val);

void VREF_init(void)
{
    /* 電圧参照基準を4.34V */
    VREF.CTRLA |= VREF_DACOREFSEL_4V34_gc;
    /* DAC0/AC0参照基準許可:許可 */
    VREF.CTRLB |= VREF_DACOREFEN_bm;
    /* VREF始動時間待ち */
    _delay_us(VREF_STARTUP_MICROS);
}
```


例8-1 (続き). 一定のアナログ信号を生成するコード例

```

}

void DAC0_init(void)
{
    /* デジタル入力緩衝部禁止 */
    PORTA.PIN6CTRL &= ~PORT_ISC_gm;
    PORTA.PIN6CTRL |= PORT_ISC_INPUT_DISABLE_gc;
    /* プルアップ抵抗禁止 */
    PORTA.PIN6CTRL &= ~PORT_PULLUPEN_bm;

    /* DAC、出力緩衝部、スタンバイで走行を許可 */
    DAC0.CTRLA = DAC_ENABLE_bm | DAC_OUTEN_bm | DAC_RUNSTDBY_bm;
}

void DAC0_setVal(uint8_t val)
{
    DAC0.DATA = val;
}

int main(void)
{
    VREF_init();
    DAC0_init();

    DAC0_setVal(DAC_EXAMPLE_VALUE);

    while (1)
    {
        ;
    }
}

```

例8-2. 正弦波信号を生成するコード例

```

/* 3.33MHz (遅延用に必要) */
#define F_CPU (3333333UL)
/* VREF始動時間 */
#define VREF_STARTUP_MICROS (25)
/* 正弦波周期用段階数 */
#define SINE_WAVE_STEPS (100)
/* 正弦波振幅 */
#define SINE_AMPLITUDE (127)
/* 正弦波DC変位(オフセット) */
#define SINE_DC_OFFSET (128)
/* 2×PI */
#define M_2PI (2 * M_PI)
/* 正弦波の周波数 */
#define OUTPUT_FREQ (50)
/* 合成繰り返し用段階遅延 */
#define STEP_DELAY_MICROS ((1000000 / OUTPUT_FREQ) / SINE_WAVE_STEPS)

#include <avr/io.h>
#include <util/delay.h>
#include <math.h>

/* 正弦波標本を格納する緩衝部 */
uint8_t sineWave[SINE_WAVE_STEPS];

void sineWaveInit(void);
void VREF_init(void);

```

例8-2 (続き). 正弦波信号を生成するコード例

```

void DAC0_init(void);
void DAC0_setVal(uint8_t val);

void sineWaveInit(void)
{
    for(uint16_t i = 0; i < SINE_WAVE_STEPS; i++)
    {
        sineWave[i] = SINE_DC_OFFSET + SINE_AMPLITUDE * sin(i * M_2PI / SINE_WAVE_STEPS);
    }
}

void VREF_init(void)
{
    /* 電圧参照基準を4.34V */
    VREF.CTRLA |= VREF_DACOREFSEL_4V34_gc;
    /* DAC0/AC0参照基準許可:許可 */
    VREF.CTRLB |= VREF_DACOREFEN_bm;
    /* VREF始動時間待ち */
    _delay_us(VREF_STARTUP_MICROS);
}

void DAC0_init(void)
{
    /* デジタル入力緩衝部禁止 */
    PORTA.PIN6CTRL &= ~PORT_ISC_gm;
    PORTA.PIN6CTRL |= PORT_ISC_INPUT_DISABLE_gc;
    /* プルアップ抵抗禁止 */
    PORTA.PIN6CTRL &= ~PORT_PULLUPEN_bm;

    /* 既定値 */
    DAC0.DATA = SINE_DC_OFFSET;
    /* DAC、出力緩衝部、スタンバイで走行を許可 */
    DAC0.CTRLA = DAC_ENABLE_bm | DAC_OUTEN_bm | DAC_RUNSTDBY_bm;
}

void DAC0_setVal(uint8_t val)
{
    DAC0.DATA = val;
}

int main(void)
{
    uint16_t i = 0;

    VREF_init();
    DAC0_init();

    sineWaveInit();

    while (1)
    {
        DAC0_setVal(sineWave[i++]);
        i = i % SINE_WAVE_STEPS;
        _delay_us(STEP_DELAY_MICROS);
    }
}

```

例8-3. ADCで内部的にDACを読むコード例

```

/* 3.33MHz (遅延用に必要) */
#define F_CPU                (3333333UL)
/* VREF始動時間 */
#define VREF_STARTUP_MICROS  (25)

#include <avr/io.h>
#include <util/delay.h>

void VREF_init(void);
void DAC0_init(void);
void ADC0_init(void);
uint16_t ADC0_read(void);
void DAC0_setVal(uint8_t val);

void VREF_init(void)
{
    /* DAC0/AC0の電圧参照基準を4.34V */
    VREF_CTRLA |= VREF_DACOREFSEL_4V34_gc;
    /* DAC0/AC0参照基準許可:許可 */
    VREF_CTRLB |= VREF_DACOREFEN_bm;
    /* ADC0の電圧参照基準を4.34V */
    VREF_CTRLA |= VREF_ADCOREFSEL_4V34_gc;
    /* ADC0参照基準許可:許可 */
    VREF_CTRLB |= VREF_ADCOREFEN_bm;
    /* VREF始動時間待ち */
    _delay_us(VREF_STARTUP_MICROS);
}

void DAC0_init(void)
{
    /* DACを許可 */
    DAC0.CTRLA = DAC_ENABLE_bm;
}

void ADC0_init(void)
{
    ADC0.CTRLC = ADC_PRESC_DIV4_gc           /* CLK_PERを4分周 */
                | ADC_REFSEL_INTREF_gc      /* 内部参照基準 */
                | ADC_SAMPCAP_bm;          /* 採取容量選択:低 */

    ADC0.CTRLA = ADC_ENABLE_bm              /* ADC許可:許可 */
                | ADC_RESSEL_10BIT_gc;     /* 10ビット動作 */

    /* ADCチャンネル選択 */
    ADC0.MUXPOS = ADC_MUXPOS_DAC0_gc;
}

uint16_t ADC0_read(void)
{
    /* ADC変換開始 */
    ADC0.COMMAND = ADC_STCONV_bm;

    /* ADC変換終了まで待機 */
    while ( !(ADC0.INTFLAGS & ADC_RESRDY_bm) )
    {
        ;
    }

    /* 1書き込みによって割り込み要求フラグ解除(0) */
    ADC0.INTFLAGS = ADC_RESRDY_bm;
}

```

例8-3 (続き). ADCで内部的にDACを読むコード例

```

return ADC0.RES;
}

void DAC0_setVal(uint8_t val)
{
    DAC0.DATA = val;
}

int main(void)
{
    uint8_t dacVal = 0;
    uint16_t adcVal = 0;

    VREF_init();
    DAC0_init();
    ADC0_init();

    /* VREF始動時間待ち */
    _delay_us(VREF_STARTUP_MICROS);

    DAC0_setVal(dacVal);

    while (1)
    {
        adcVal = ADC0_read();

        /* adcValで何かを行ってください。 */

        dacVal++;
        DAC0_setVal(dacVal);
    }
}

```

例8-4. AC用負入力としてDACを使うコード例

```

/* 3.33MHz (遅延用に必要) */
#define F_CPU (3333333UL)
/* VREF=1.5Vに対して1.4V出力 */
#define DAC_DATA_1V4 (239)
/* VREF始動時間 */
#define VREF_STARTUP_MICROS (25)

#include <avr/io.h>
#include <util/delay.h>

void VREF_init(void);
void DAC0_setVal(uint8_t val);
void DAC0_init(void);
void AC0_init(void);

void VREF_init(void)
{
    /* 電圧参照基準を1.5V */
    VREF.CTRLA |= VREF_DACOREFSEL_1V5_gc;
    /* DAC0/AC0参照基準許可:許可 */
    VREF.CTRLB |= VREF_DACOREFEN_bm;
    /* VREF始動時間待ち */
    _delay_us(VREF_STARTUP_MICROS);
}

```

例8-4 (続き). AC用負入力としてDACを使うコード例

```

void DAC0_setVal(uint8_t val)
{
    DAC0.DATA = val;
}

void DAC0_init(void)
{
    /* デジタル入力緩衝部禁止 */
    PORTA.PIN6CTRL &= ~PORT_ISC_gm;
    PORTA.PIN6CTRL |= PORT_ISC_INPUT_DISABLE_gc;
    /* プルアップ抵抗禁止 */
    PORTA.PIN6CTRL &= ~PORT_PULLUPEN_bm;

    /* DAC、出力緩衝部、スタンバイで走行を許可 */
    DAC0.CTRLA = DAC_ENABLE_bm | DAC_OUTEN_bm | DAC_RUNSTDBY_bm;
}

void AC0_init(void)
{
    AC0.MUXCTRLA = AC_MUXNEG_DAC_gc           /* DAC0からの負入力 */
                 | AC_MUXPOS_PIN0_gc;       /* PA7ピンからの正入力 */

    AC0.CTRLA = AC_ENABLE_bm                 /* ACを許可 */
               | AC_OUTEN_bm                 /* PA5での出力許可 */
               | AC_RUNSTDBY_bm             /* スタンバイで走行許可 */
               | AC_LPMODE_bm;              /* 低電力許可 */
}

int main(void)
{
    /* 分圧器⇒PA7でVDD/2を入力 */
    /* ACはPA5のLEDで出力 */
    /* 電圧(VDD)が2.8V(PA7が1.4V)以下の時にLEDがOFFに切り替わります。 */

    VREF_init();
    DAC0_init();
    AC0_init();

    /* VREF=1.5Vに対して1.4V出力 */
    DAC0_setVal(DAC_DATA_1V4);

    while (1)
    {
        ;
    }
}

```

Microchipウェブ サイト

Microchipは<http://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手段として使われます。お気に入りのインターネット ブラウザを用いてアクセスすることができ、ウェブ サイトは以下の情報を含みます。

- **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相談役プログラム員一覧
- **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

お客様への変更通知サービス

Microchipのお客様通知サービスは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<http://www.microchip.com/>でMicrochipのウェブ サイトをアクセスしてください。”Support”下で”Customer Change Notification”をクリックし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
- 現場応用技術者(FAE:Field Application 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、または現場応用技術者(FAE)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<http://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デバイス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、その製品系統が今日の市場でその種類の最も安全な系統の1つであると考えます。
- コード保護機能を破るのに使われる不正でおそらく違法な方法があります。当社の知る限りこれらの方法の全てはMicrochipのデータシートに含まれた動作仕様外の方法でMicrochip製品を使うことが必要です。おそらく、それを行う人は知的財産の窃盗に関与しています。
- Microchipはそれらのコードの完全性について心配されているお客様と共に働きたいと思います。
- Microchipや他のどの半導体製造業者もそれらのコードの安全を保証することはできません。コード保護は当社が製品を”破ることができない”として保証すると言った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。Microchipのコード保護機能を破る試みはデジタル ミレニアム著作権法に違反するかもしれません。そのような行為があなたのソフトウェアや他の著作物に不正なアクセスを許す場合、その法律下の救済のために訴権を持つかもしれません。

法的通知

デバイス応用などに関してこの刊行物に含まれ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。Microchipはその条件、品質、性能、商品性、目的適合性を含め、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ありません。Microchipはこの情報とそれの使用から生じる全責任を否認します。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、免責に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、Mcirochipロゴ、AnyRate、AVR、AVRロゴ、AVR Freaks、BitCloud、chipKIT、chipKITロゴ、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KeeLoq、KeeLoqロゴ、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOSTロゴ、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32ロゴ、Prochip Designer、QTouch、SAM-BA、SpyNIC、SST、SSTロゴ、SuperFlash、tinyAVR、UNI/O、XMEGAは米国と他の国に於けるMicrochip Technology Incorporatedの登録商標です。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge、Quiet-Wire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、BodyCom、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、INICnet、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNetロゴ、memBrain、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certifiedロゴ、MPLAB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、View Sense、WiperLock、Wireless DNA、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。

Silicon Storage Technology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。

© 2019年、Microchip Technology Incorporated、米国印刷、不許複製

DNVによって認証された品質管理システム

ISO/TS 16949

Microchipはその世界的な本社、アリゾナ州のチャンドラーとテンペ、オレゴン州グラシャムの設計とウェハー製造設備とカリフォルニアとインドの設計センターに対してISO/TS-16949:2009認証を取得しました。当社の品質システムの処理と手続きはPIC[®] MCUとdsPIC[®] DSC、KEELOQ符号飛び回りデバイス、直列EEPROM、マイクロ周辺機能、不揮発性メモリ、アナログ製品用です。加えて、開発システムの設計と製造のためのMicrochipの品質システムはISO 9001:2000認証取得です。

日本語© HERO 2020.

本技術概説はMicrochipのTB3210技術概説(DS90003210A-2019年1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。



MICROCHIP

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: http://www.microchip.com/support ウェブアドレス: www.microchip.com 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	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 - 香港特別行政区 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 - フネー 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: 45-4450-2828 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: 49-7131-67-3636 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 - ラーナナ Tel: 972-9-744-7705 イタリア - ミラノ Tel: 39-0331-742611 Fax: 39-0331-466781 イタリア - ハドバ Tel: 39-049-7625286 オランダ - デルネン 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