

第8 α 章 PIC16F88を用いたステッピングモータ の速度制御

本稿のWebページ

目次

- 8 α -1. PIC16F88を用いたステッピングモータ制御の実験回路図
回路図
立体配線図
完成写真
- 8 α -2. ステッピングモータの定速駆動
- 8 α -3. タイマ0割り込みによる制御周期管理
- 8 α -4. A/D変換モジュール
- 8 α -5. ステッピングモータの速度制御

PICアセンブラ入門演習の[第8章](#)にPIC16F84Aを用いたステッピングモータの速度制御回路とアセンブラプログラムを紹介したところ、とても多くの方からアクセスを受けています。

本章では[PIC16F88](#)を用いたステッピングモータの速度制御回路とアセンブラプログラムを紹介します。この速度制御は、PIC16F88内蔵の[A/D変換器](#)により回転数指令値を読み込み、その値を用いてタイマ0の割り込み周期を設定してステッピングモータの速度制御を行っています。

PIC16F88は吉田靖幸氏（トランジスタ技術2012年6月号p.74）によると、2011年に秋月電子通商で最も売れたPICマイコンです。ユーザの根強い支持があると推測されます。

DIPタイプのPIC16F88とPIC16F84Aでは，共通モジュールのピン配置は同じです．このため第0章の図19の回路図においてPIC16F88をPIC16F84Aと置き換えて，速度制御用の外部回路を一部変更するだけでステッピングモータの制御ができます．

次のスライドから始まる図1～4に本章の制御のために改造した回路図と立体配線図を示します．図中の一点鎖線で囲った部分に変更箇所です．第0章の図19の可変抵抗 VR_1 ，抵抗 R_1 ， R_2 ，トランジスタ Tr_1 ，コンデンサ C からなる回路を可変抵抗 VR_1 のみの回路に変更してあります．

8α-1. ステッピングモータ制御の実験回路図

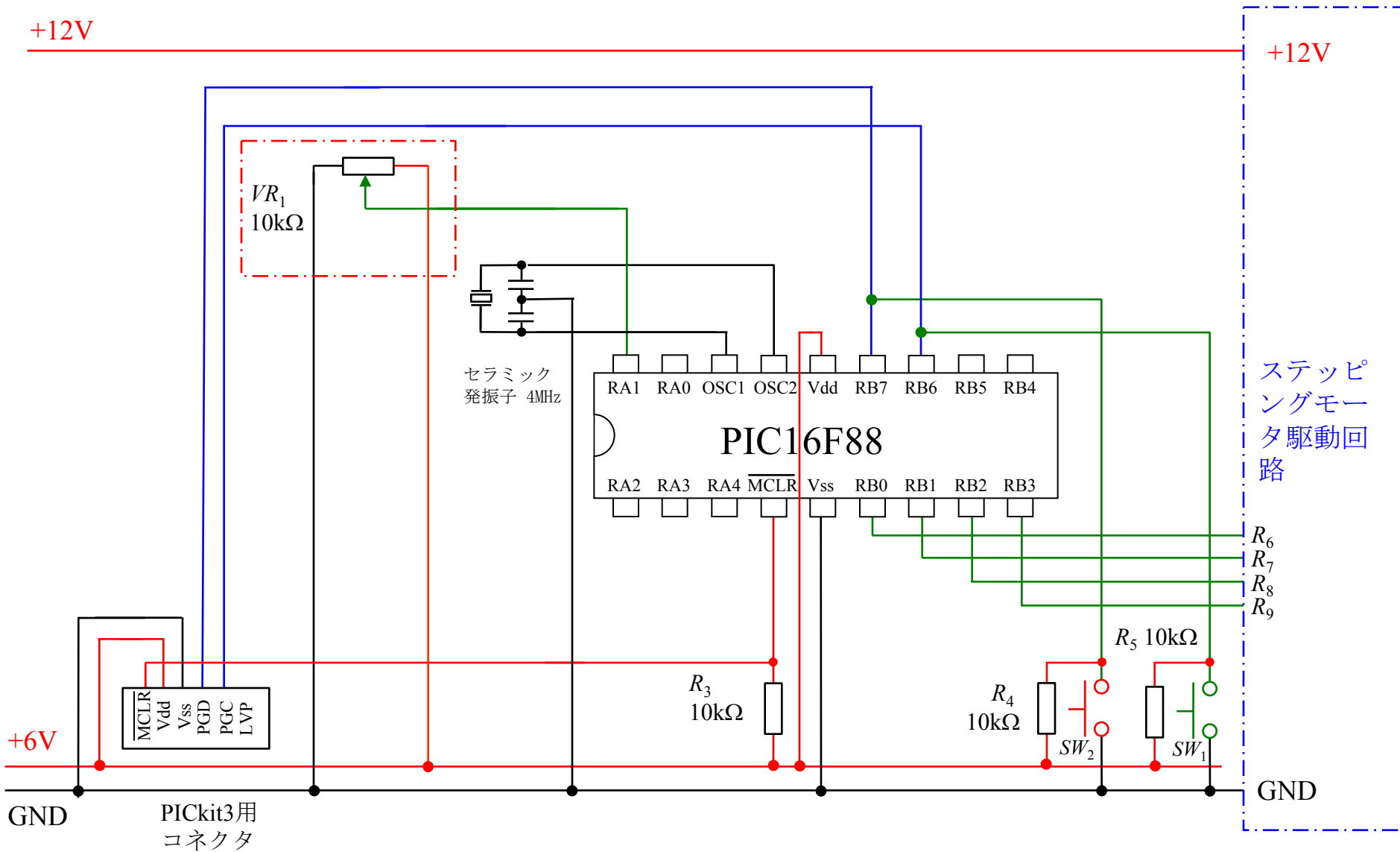


図1 マイコン回路

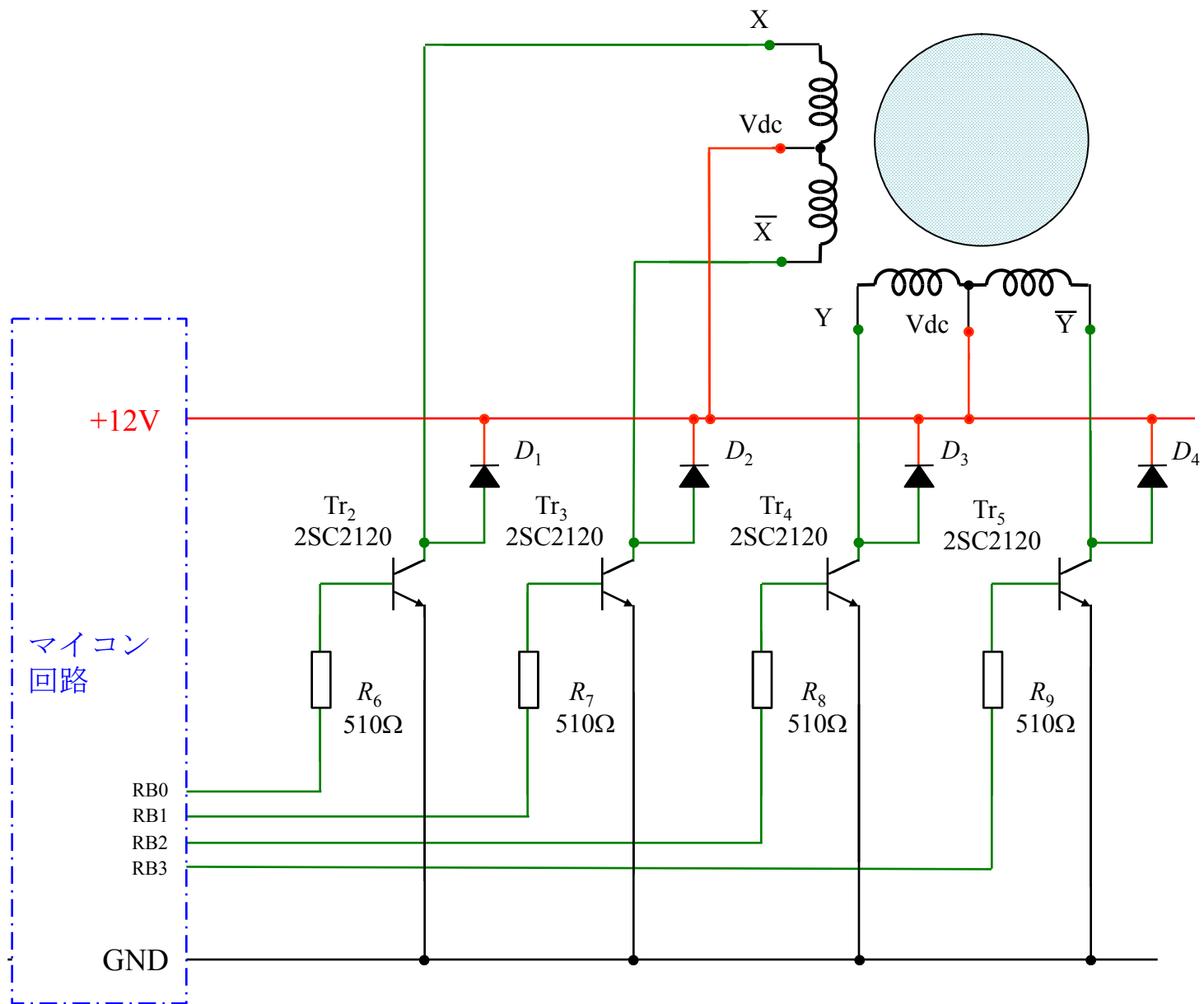


図2 ステッピングモータ駆動回路

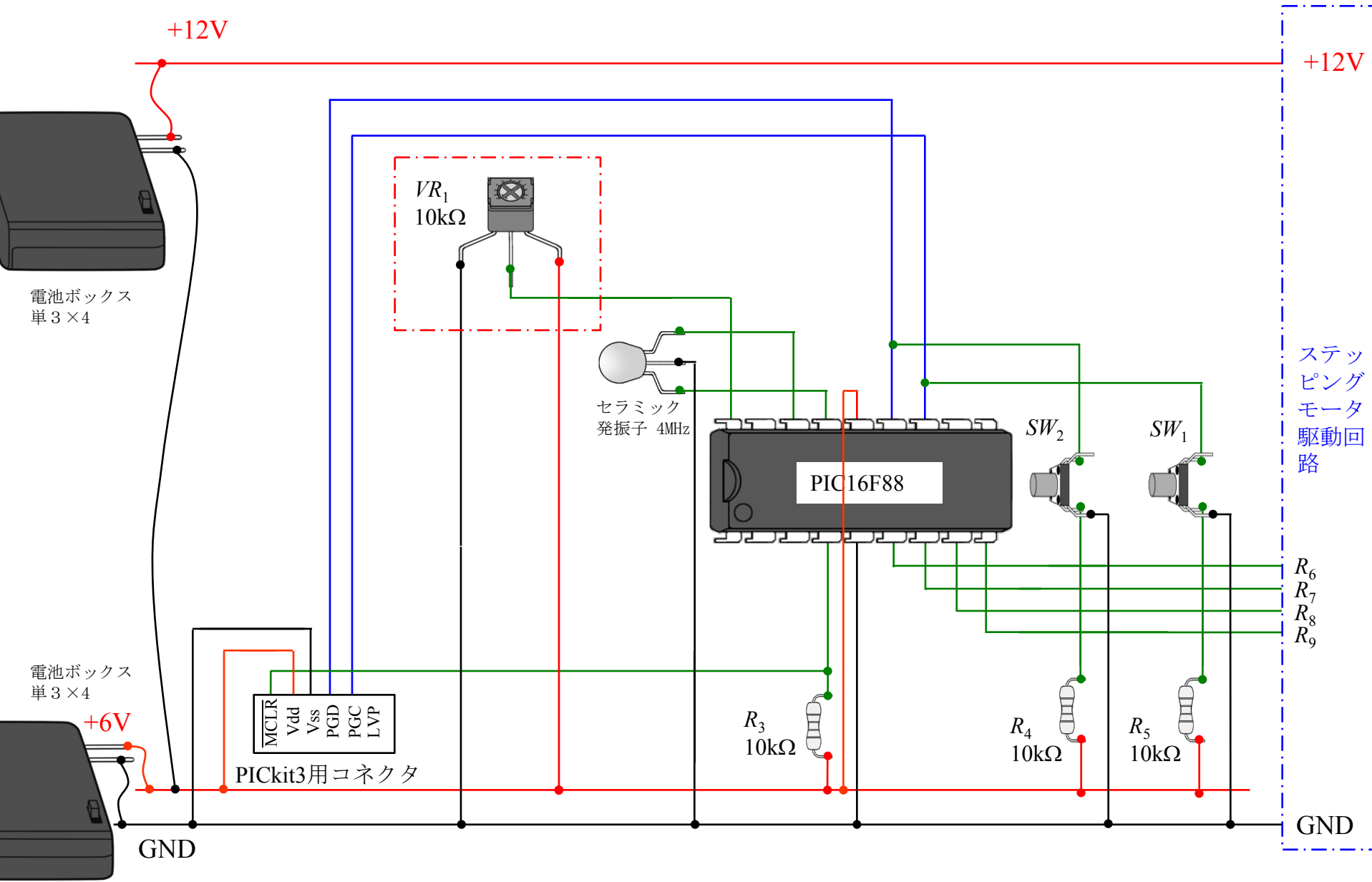


図3 マイコン回路の立体配線図

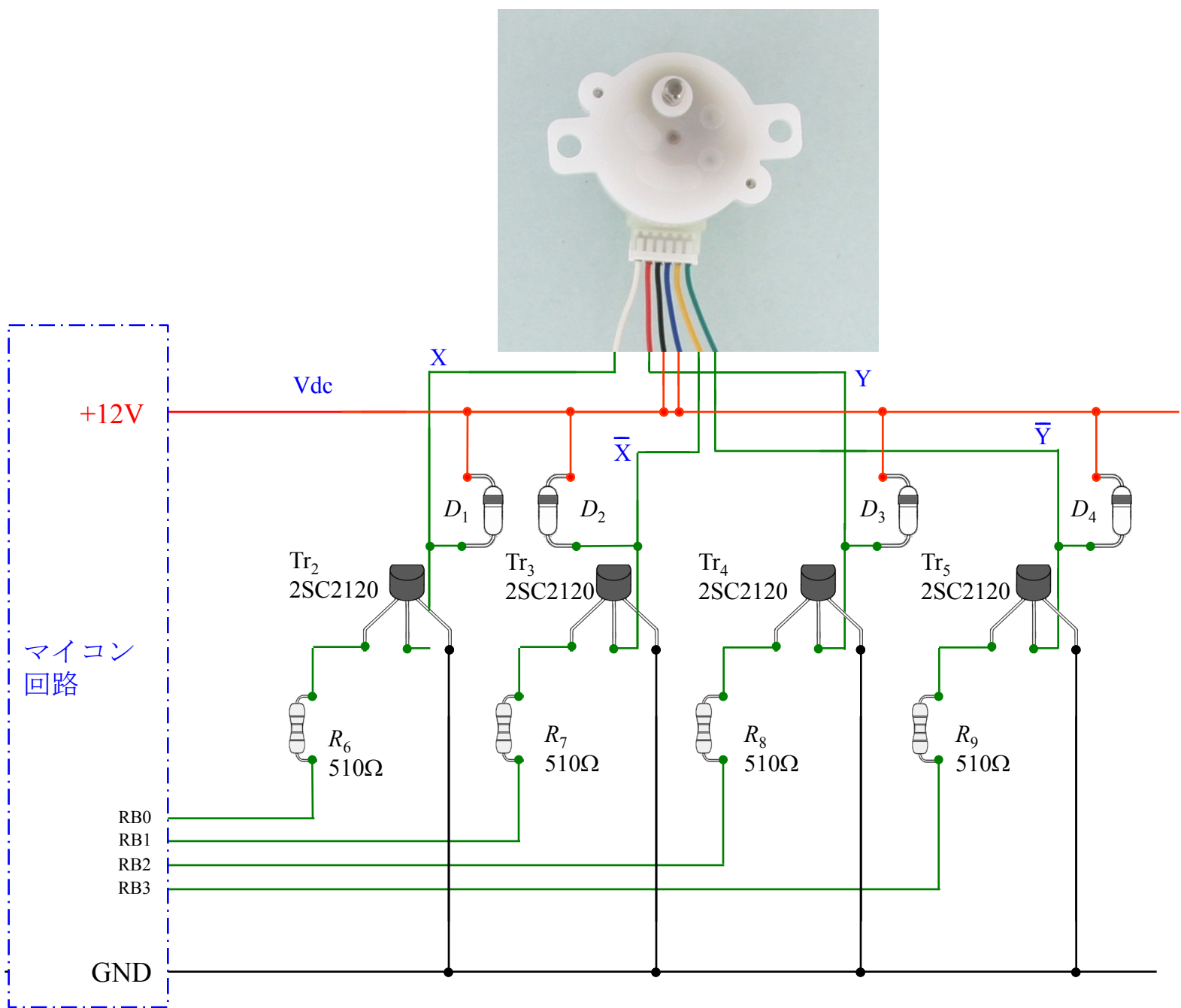


図4ステッピングモータ駆動回路の立体配線図

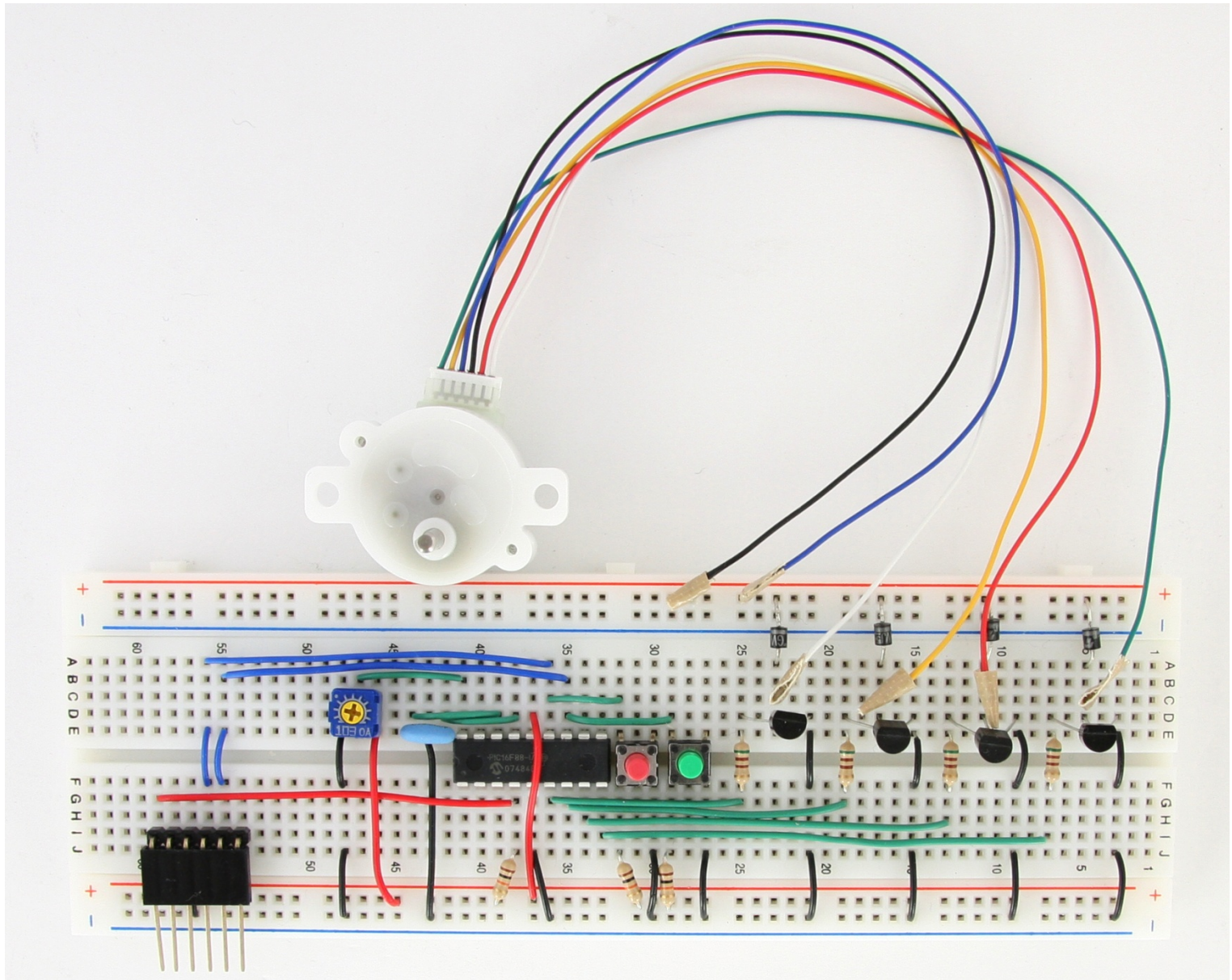
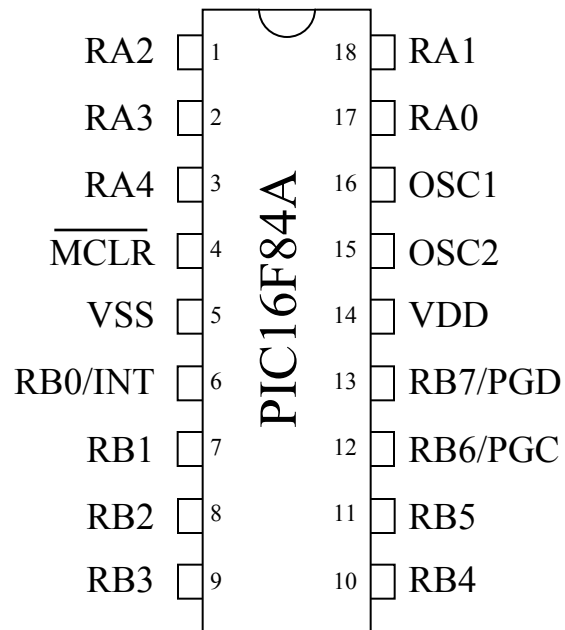
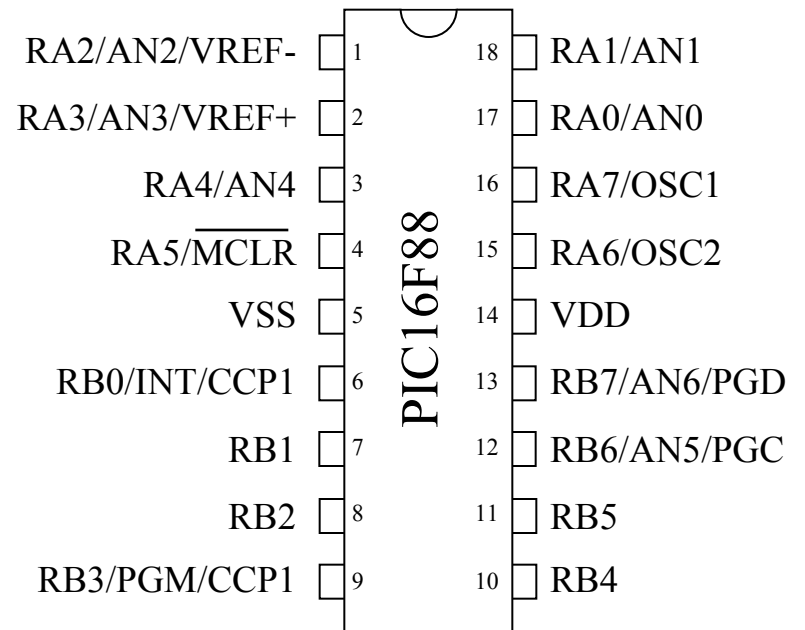


図5 完成写真 (マイコン回路+ステッピングモータ駆動回路)



(a) PIC16F84Aのピン配置



(b) PIC16F88のピン配置

図6 PIC16F88とPIC16F84Aのピン配置

8α-2. ステッピングモータの定速駆動

本節では[第8章](#) 8.1節のPIC16F84A用のステッピングモータの定速駆動プログラムをPIC16F88用書き直す。PIC16F84A用プログラムとの違いを朱書きで示す。

SW1を押すとステッピングモータが時計方向に回転

SW2を押すと反時計方向に回転

ステッピングモータの回転速度は一定

第8章8.1節のPIC16F84Aのプログラムとの違いを朱書きで示す。

注意 CONFIGの前のアンダーバーは2つあります。

;Stepping Motor Control Program (Speed fixed)

INCLUDE"p16F88.inc"

list p=16F88

__CONFIG __CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF
__CONFIG __CONFIG2, _FCMEN_OFF & _IESO_OFF

Register	EQU	0x20	
MEM1	EQU	Register +0	;MEM1 at 20
TIME1	EQU	Register +1	;TIME1 at 21
TIME2	EQU	Register +2	;TIME2 at 22
TIME3	EQU	Register +3	;TIME3 at 23

ORG 0
GOTO START ;Main Program starts at START

ORG 4

START

;Setting of Port B

BSF STATUS,RP0
BCF STATUS,RP1 ;Selection of Bank 1
MOVLW B'11000000'
MOVWF TRISB ;RB0-5 -> Output Port, RB6,7 -> Input Port
BCF ANSEL, 6 ;RB7/AN6 Digital Input
BCF ANSEL, 5 ;RB6/AN5 Digital Input

BCF STATUS,RP0
BCF STATUS,RP1 ;Selection of Bank 0

MOVLW B'0000101'
MOVWF PORTB ;'00001010' -> (W)
;(W) -> (PORTB)

詳細はp.15~19に

第8章8.1節のPIC16F84Aのプログラムと全く一緒

; Main Program

```
STEP1      MOVF      PORTB,0           ;(RB)->(W)
           ANDLW     B'11000000'      ;(W) and 11000000 -> (W)
           MOVWF     MEM1             ;(W) -> (MEM1)

           BTFSS     MEM1,7           ;If the 7th bit = 1, Then skip
           CALL      RotateR
           BTFSS     MEM1,6           ;If the 6th bit = 1, Then skip
           CALL      RotateL
           GOTO      STEP1
```

;End of Main Program

;Sub Routine1

```
RotateR
           MOVLW     B'00000101'      ;'00000101' -> (W)
           MOVWF     PORTB            ;(W) -> (PORTB)
           CALL      COUNT1
           MOVLW     B'00000110'
           MOVWF     PORTB
           CALL      COUNT1
           MOVLW     B'00001010'
           MOVWF     PORTB
           CALL      COUNT1
           MOVLW     B'00001001'
           MOVWF     PORTB
           CALL      COUNT1
           RETURN
```

;Sub Program2

RotateL

```

    MOVLW    B'00000101'
    MOVWF   PORTB
    CALL    COUNT1
    MOVLW   B'00001001'
    MOVWF   PORTB
    CALL    COUNT1
    MOVLW   B'00001010'
    MOVWF   PORTB
    CALL    COUNT1
    MOVLW   B'00000110'
    MOVWF   PORTB
    CALL    COUNT1
    RETURN

```

;Idling timer

```

COUNT1    MOVLW    0x10
            MOVWF   TIME1

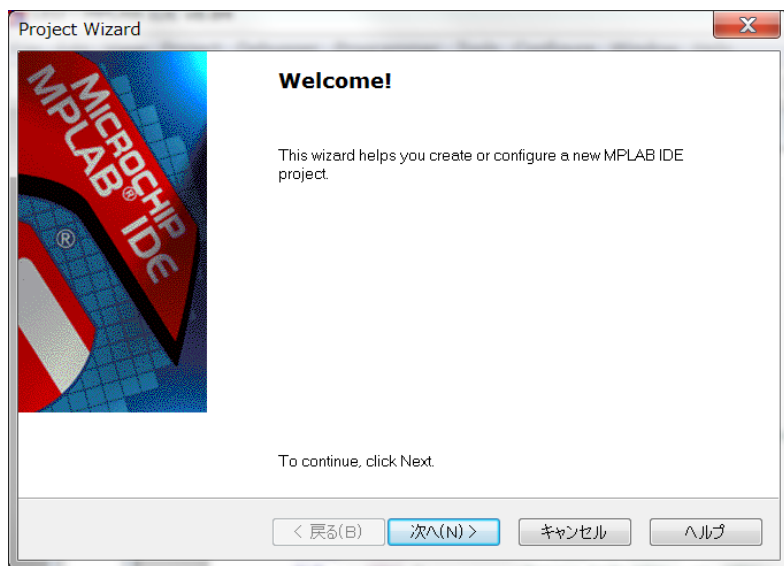
STEPM      MOVWF   TIME2
STEPM1     MOVWF   TIME3
STEPM2     DECFSZ  TIME3,1
            GOTO    STEPM2
            DECFSZ  TIME2,1
            GOTO    STEPM1
            DECFSZ  TIME1,1
            GOTO    STEPM
            RETURN

```

END

新しいプロジェクトの設定

MPLAB IDE v8.84をダブルクリック → Project → Project Wizard →



[第0章 p.19](#)と同様に設定する。

ステッピングモータの定速駆動プログラム(本章p.11)の詳細説明(1)

;Stepping Motor Control Program (Speed fixed)

```
INCLUDE"P16F88.INC"
```

```
list p=16F88
```

PIC16F88とする。

```
_CONFIG _CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF  
_CONFIG _CONFIG2, _FCMEN_OFF & _IESO_OFF
```

コンフィギュレーション：
PIC16F88のデータシートによると、PIC16F88にはCONFIGURATION WORD REGISTERがCONFIG1とCONFIG2の2つある。
それぞれを設定する必要がある。各用語の説明はMicrochip MPASM Suiteのフォルダ内のP16F88.INC内に記載されている。

工場出荷時にはLVPはONに設定されている。このとき9番ピンはPGMが有効となっていて、入出力ピンRB3としては使えない。データシートのBLOCK DIAGRAM OF RB3/PGM/CCP1 PIN参照。
そこで、LVPをOFFとする。LVP (Low Voltage Programming)は、5[V]レベルでのプログラム書き込みを可能とする機能であるが、PICKit3を利用する場合には必要ない。

ステッピングモータの定速駆動プログラム(本章p.11)の詳細説明(2)

Register	EQU	0x20	
MEM1	EQU	Register +0	;MEM1 at 20
TIME1	EQU	Register +1	;TIME1 at 21
TIME2	EQU	Register +2	;TIME2 at 22
TIME3	EQU	Register +3	;TIME3 at 23

PIC16F88にはファイルレジスタが4Bankある(次ページの図7参照).

汎用レジスタ(General Purpose Register)は

Bank 1の 20h~7Fh番地

Bank 2の A0h~ Efh番地

Bank 3の 120h~16Fh番地

Bank 4の1A0h~1EFh番地

にある. 上記の設定ではBank 0の20h~23h番地を使う設定である. これらのメモリにアクセスするためには事前にBank 0を選定しておかなければならない.

TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh		18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh		18Fh
T1CON	10h	OSCTUNE	90h		110h		190h
TMR2	11h		91h				
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPADDD	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
	1Bh	ANSEL	9Bh				
	1Ch	CMCON	9Ch				
	1Dh	CVRCON	9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh				
	20h		A0h				
General Purpose Register 96バイト		General Purpose Register 80バイト		General Purpose Register 80バイト	11Fh	General Purpose Register 80バイト	19Fh
					120h		1A0h
			Efh		16Fh		1EFh
			F0h		170h		1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

図7 File Register Map

ステッピングモータの定速駆動プログラム(本章p.11)の詳細説明(3)

```
ORG    0                ;電源が入るとこの番地からプログラム実行を開始する.
GOTO   START

ORG    4

START
;Setting of Port B
BSF    STATUS,RP0
BCF    STATUS,RP1      ;Selection of Bank 1
MOVLW  B'11000000'
MOVWF  TRISB           ;RB0-5 -> Output Port,
                       ;RB6,7 -> Input Port
BCF    ANSEL, 6        ;RB7/AN6 Digital Input
BCF    ANSEL, 5        ;RB6/AN5 Digital Input
```

図7によると TRISB が Bank 1にあるので, Bank 1を選定している.
Bankの選定は, データシートの ARITHMETIC STATUS REGISTERによると,
RP1, RP0 = 11 Bank3
 = 10 Bank2
 = 01 Bank1
 = 00 Bank0
である. なお, STATUS レジスタは4バンクに共通している.

データシートの TRISB REGISTERによると,
TRISB bit = 1 → PORTB ピン 入力ポート
 = 0 → " 出力ポート

ANSEL:ANALOG SELECT REGISTERによると
ANSELbit = 1 → Analog I/O
 = 0 → Digital I/O

ステッピングモータの定速駆動プログラム(本章p.11)の詳細説明(4)

```
BCF     STATUS,RP0
BCF     STATUS,RP1      ;Selection of Bank 0

MOVLW  B'00000101'     ;'00001010' -> (W)
MOVWF  PORTB           ;(W) -> (PORTB)
```

図7よりPORTBはBank 0にある。そこで、Bank 0を選定している。

また、MEM1, TIME1, TIME2, TIME3はBank 0の20h~23h番地を指定した。ここでBank 0を選定したことで、以降Bankの選定を変えない限り、これらのレジスタにもアクセス可能である。

8α-3. タイマ0割り込みによる制御周期管理

;タイマ0による割り込みプログラム

RB0(6番ピン) に1と0を周期的に出力するプログラム。新しいところを朱書きで示す。詳細は本章 p.22以降

```

INCLUDE"p16F88.inc"
list p=16F88

__CONFIG __CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF
__CONFIG __CONFIG2, _FCMEN_OFF & _IESO_OFF

ORG      0          ;電源が入るとこの番地からプログラム実行を開始する。
GOTO    START

ORG      4          ;割り込みがかかるとこの番地から開始する。
CALL    Timer0_interrupt
RETFIE          ;割り込み処理ルーチンからメインプログラムへ復帰

START
;ポートBの設定
BSF     STATUS,RP0
BCF     STATUS,RP1          ;バンク1の選択
MOVLW  B'00000000'
MOVWF  TRISB          ;RB0-7を出力ポートに設定

;タイマ0の設定
BCF     OPTION_REG, T0CS          ;システムクロック(FOSC)を選択. 実際には FOSC/4 = 4MHz/4 = 1MHz
BCF     OPTION_REG, PSA          ;プリスケアラをタイマ0用に設定. (PSA = 1とすると, プリスケアラはWDT専用となる.)
BSF     OPTION_REG, PS2
;
BSF     OPTION_REG, PS1
;
BSF     OPTION_REG, PS0          ;PS2 PS1 PS0 = 111 とすることでタイマ0のクロックをFOSC/4/256 と設定.

;割込みの設定
BSF     INTCON, GIE          ;マスクされていない全ての割込みを可とする.
BSF     INTCON, PEIE          ;マスクされていない全ての周辺モジュールからの割込みを可とする.
BSF     INTCON, TMR0IE          ;タイマ0の割込みを可とする.
BCF     INTCON, TMR0IF          ;タイマ0の割込みフラグをクリアする.

BCF     STATUS,RP0
BCF     STATUS,RP1          ;バンク0の選択

```

タイマ0による割り込みプログラム(つづき)

;メインルーチン

STEP1 GOTO STEP1 ;STEP1にジャンプすることを繰り返す永久ループ

;サブルーチン

Timer0_interrupt

MOVLW B'00000001' ;'00000001' -> (W)

MOVWF PORTB ;(W) -> (PORTB)

MOVLW B'00000000'

MOVWF PORTB

MOVLW B'11100000' ;11100000 ->(W)

;タイマ0はこの値を初期値としてカウントアップする.

;(次の割り込みは11111111 -> 00000000 となるタイミング)

MOVWF TMR0 ;(W) -> TMR0

BCF INTCON, TMR0IF ;TMR0による再割り込みを可能とする.

RETURN

END

タイマ0による割り込みプログラム(本章p.20, 21)の詳細説明(1)

```
ORG    0                ;電源が入るとこの番地からプログラム実行を開始する.  
GOTO   START  
  
ORG    4                ;割り込みがかかるとこの番地から開始する.  
CALL   Timer0_interrupt ;Timer0_interruptルーチンへの分岐  
RETFIE ;割り込み処理ルーチンからメインルーチンへ復帰
```

START

;ポートBの設定

```
BSF    STATUS,RP0  
BCF    STATUS,RP1      ;バンク1の選択  
MOVLW  B'00000000'  
MOVWF  TRISB          ;RB0-7を出力ポートに設定
```

タイマ0による割り込みプログラム(本章p.20, 21)の詳細説明(2)

;タイマ0の設定

```
BCF    OPTION_REG, T0CS ;システムクロック(FOSC)を選択.  
;実際には FOSC/4 = 4MHz/4 = 1MHz  
BCF    OPTION_REG, PSA ;プリスケアラをタイマ0用に設定.  
; (PSA = 1とすると, プリスケアラはWDT専用となる.)  
BSF    OPTION_REG, PS2 ; PS2 PS1 PS0 = 111 とすることで  
BSF    OPTION_REG, PS1 ;タイマ0のクロックをFOSC/4/256 と設定.  
BSF    OPTION_REG, PS0 ;
```

データシートによるとOPTION_REG Register (図7 Bank1, 3に共通) の各ビットによりタイマ0のクロックの選定, タイマ0のプリスケアラの設定ができる。上のプログラムの設定によりタイマ0の入力クロックは, セラミック発振子に4 [MHz]のものを用いた場合, $4 \text{ [MHz]}/4/256 = 3.906 \text{ [kHz]}$ となる。

このクロックによりタイマ0をカウントアップして, タイマ0がオーバフロー (B'1111111' → B'0000000')するタイミングで 割り込みをかけることができる (ORG 4からのプログラムを実行する)。割り込み処理プログラムの中でタイマ0の値を例えばB'11100000' と設定すれば,
 $(B'10000000' - B'11100000')/3.906 \text{ [kHz]} = (256 - 224)/3.906 \text{ [kHz]} = 8.193 \text{ [ms]}$
後に再び割り込みがかけられる。

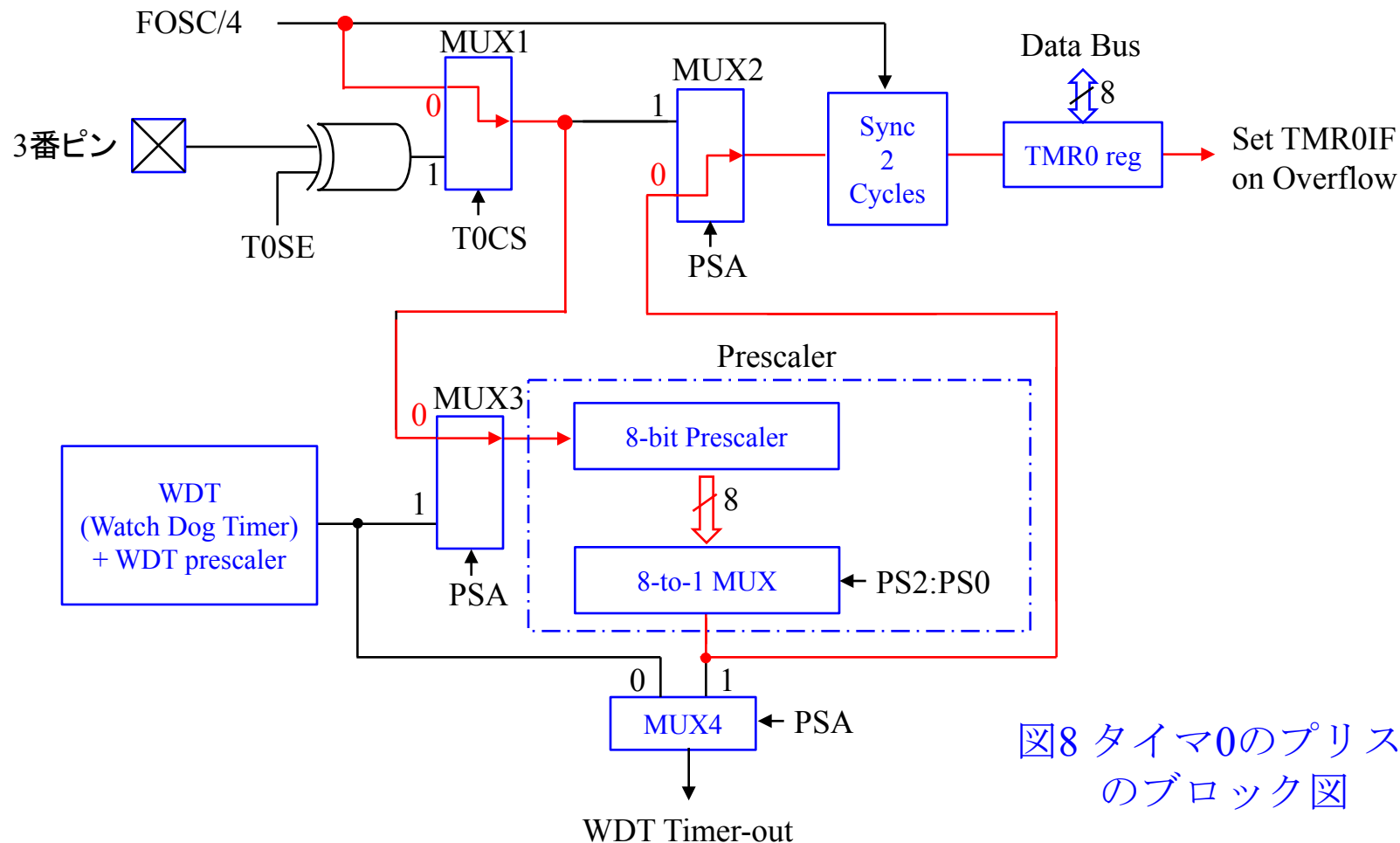


図8 タイマ0のプリスケアラのブロック図

MUX:マルチプレクサ(Multiplexer), 複数の入力のいずれかを選んで出力する, 入力切り替え器

例えばMUX1はT0CS=0のとき入力のFOSC/4を出力する. T0CS=1のときはXORの値を出力する.

Prescaler: プリスケアラ, クロックの分周器.

上図の設定例ではタイマ0に入る前のクロックを $(1/2)^n$ 倍($n=1\sim 8$)する. n はPS2~PS0により設定できる.

WDT: ウォッチドッグタイマ(Watch Dog Timer), コンフィギュレーションにて(`_CONFIG_CONFIG1_WDT_ON`)とすると, WDTが起動する. 設定時間内にプログラムがCLRWD命令を実行して, WDTとそのWDT prescalerをクリア(0を代入)しないとプログラム実行が強制リセットされる. プログラムが暴走した場合などにWDTは有効. 番犬タイマという意味.

Sync 2 Cycles: Data BusからTMR0 registerに値を書き込む際に, FOSC/4の2クロックの間, TMR0 registerの入力(MUX2の出力)は無視される.

タイマ0による割り込みプログラム(本章p.20, 21)の詳細説明(4)

;割り込みの設定

BSF	INTCON, GIE	;マスクされていない全ての割り込みを可とする.
BSF	INTCON, PEIE	;マスクされていない全ての周辺モジュールからの割り込みを可とする.
BSF	INTCON, TMR0IE	;タイマ0の割り込みを可とする.
BCF	INTCON, TMR0IF	;タイマ0の割り込みフラグをクリアする.
BCF	STATUS, RP0	
BCF	STATUS, RP1	;バンク0の選択

図7よりPORTBはBank 0にある。割り込み処理ルーチンにてPORTBを利用するので、Bank 0を選択しておく。

データシートによるとINTCON Register (図7 Bank0~3に共通) の各ビットによりタイマ0の割り込みを設定できる。

上記のGIE, PEIE, TMR0IEのビットを1にセットし、TMR0IFを0にクリアすることで、タイマ0がオーバフローしたときに、割り込みをかけることができる。なお、タイマ0がオーバフローした際にはTMR0IFはセットされるので、割り込み処理プログラムの中で、TMR0IFをクリアしておく必要がある。これによりタイマ0による割り込みを続け続けることができる。

タイマ0による割り込みプログラム(本章p.20, 21)の詳細説明(5)

;サブルーチン

Timer0_interrupt

```
MOVLW B'00000001'  
MOVWF PORTB  
MOVLW B'00000000'  
MOVWF PORTB
```

```
;'00000001' -> (W)  
;(W) -> (PORTB)
```

RB0(6番ピン)に1を出力する。

RB0に0を出力する。

```
MOVLW B'11100000'
```

```
;11100000 ->(W)
```

;タイマ0はこの値を初期値としてカウントアップする。

```
MOVWF TMR0
```

```
;(W) -> TMR0
```

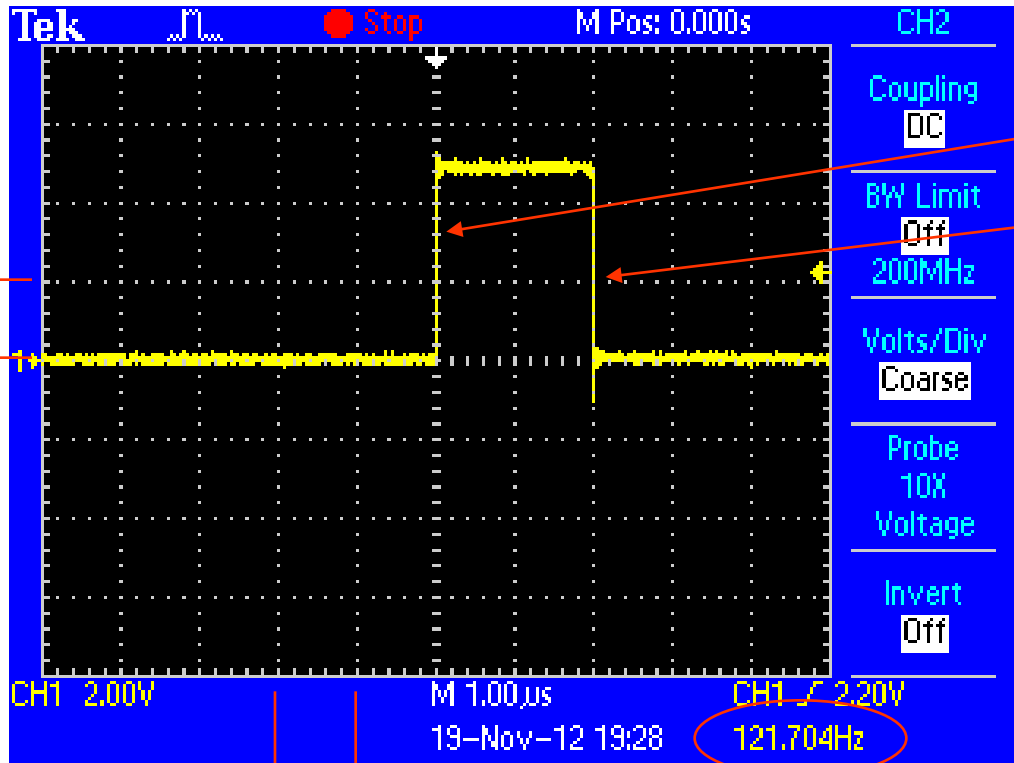
```
BCF INTCON, TMR0IF
```

;TMR0による再割り込みを可能とする。

```
RETURN
```

タイマ0がオーバフローした際にはTMR0IFはセットされるので、割り込み処理プログラムの中で、TMR0IFをクリアしておく必要がある。これにより再びタイマ0による割り込みをかけることができる。

タイマ0の初期値をB'11100000' = 224に設定。この値からカウントアップしてオーバフロー(B'11111111' → B'00000000')するタイミングで割り込みをかけることができる。割り込み周期は
 $(B'100000000' - B'111000000') / 3.906[\text{kHz}] = (256 - 224) / 3.906[\text{kHz}] = 8.193 [\text{ms}]$
となる。



```

MOVLW B'00000001'
MOVWF PORTB
MOVLW B'00000000'
MOVWF PORTB

```

FOSC/4 = 1 [MHz] → 1 [μs]を1サイクルとして，各命令の実行に1サイクルを要する．1を出力してから0を出力するまでに2サイクル= 2[μs]を要していることが分かる．

オシロスコープの周波数カウンタ機能による
割り込み周波数の計測結果.
割り込み周波数 = 121.704[Hz]

割り込み周期計算値
 $(256 - 224) / 3.906[\text{kHz}] = 8.193 [\text{ms}]$
 割り込み周波数計算値
 $1 / 8.193[\text{ms}] = 122.1 [\text{Hz}]$

誤差の主な原因はセラミック発振子の発振周波数が4[MHz]よりわずかに低いことによる.

図9 タイマ0による割り込みプログラム実行時の6番ピンの出力波形

8α-4. A/D変換モジュール

: A/D変換プログラム

```
INCLUDE"p16F88.inc"  
list p=16F88
```

```
__CONFIG __CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF  
__CONFIG __CONFIG2, _FCMEN_OFF & _IESO_OFF
```

```
ORG 0  
GOTO START ;Main Program starts from START
```

```
ORG 4  
CALL Timer0_interrupt  
RETFIE
```

START

;Setting of Port B

```
BSF STATUS,RP0  
BCF STATUS,RP1 ;Selection of Bank 1  
MOVLW B'00000000'  
MOVWF TRISB ;RB0-7 -> Output Port  
MOVLW B'00000010'  
MOVWF TRISA ;RA1 -> Input Port
```

;Setting of Timer0

```
BCF OPTION_REG, T0CS ;Internal instruction cycle clock (CLKO)  
BCF OPTION_REG, PSA ;Internal instruction cycle clock (CLKO)  
BSF OPTION_REG, PS2 ;  
BSF OPTION_REG, PS1 ;  
BSF OPTION_REG, PS0 ;Timer0 Rate 1:256
```

AN1(18番ピン) からアナログ電圧をA/D変換モジュールにより読み込み、タイマ0による割り込み周期をアナログ電圧に応じて可変とするプログラム。8α-3節のタイマ0による割り込みプログラムに対して新しく追加した箇所を朱書きで示す。詳細は本章

[p.31](#)以降

A/D変換プログラム(つづき)

;Setting of Interrupt

```
BSF    INTCON, GIE           ;Enables all unmasked interrupts
BSF    INTCON, PEIE        ;Enables all unmasked peripheral interrupts
BSF    INTCON, TMR0IE     ;Enables the TMR0 interrupt
BCF    INTCON, TMR0IF     ;TMR0 register did not overflow
```

;Setting of AD Converter

```
MOVLW  B'00000010'
MOVWF  ANSEL                ;AN1:Analog I/O
BCF    ADCON1, ADFM        ;Left Justified
BCF    ADCON1, ADCS2      ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF    ADCON1, VCFG1     ;Voltage Reference = VSS
BCF    ADCON1, VCFG0     ;Voltage Reference = VDD

BCF    STATUS, RP0
BCF    STATUS, RP1        ;Selection of Bank 0

BCF    ADCON0, ADCS1     ;
BSF    ADCON0, ADCS0     ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF    ADCON0, CHS2
BCF    ADCON0, CHS1
BSF    ADCON0, CHS0     ;CHS = 001 -> AN1
BSF    ADCON0, ADON     ;A/D converter module is operating
```

A/D変換プログラム(つづき)

```
; Main Program

STEP1      GOTO      STEP1

;End of Main Program

;Sub Routine1
Timer0_interrupt
      MOVLW      B'00000001'      ;'00000001' -> (W)
      MOVWF     PORTB            ;(W) -> (PORTB)

      BSF       ADCON0, GO        ;AD Conversion Start
      BTFSC    ADCON0, 2
      GOTO     AD_Wait           ;Wait until AD conv is done

      MOVF     ADRESH, 0         ;ADRESH -> (W)
      MOVWF    TMR0             ;(W) -> TMR0

      MOVLW    B'00000000'
      MOVWF    PORTB

      BCF      INTCON, TMR0IF    ;TMR0 register did not overflow

      RETURN

      END
```

A/D変換プログラム(本章p.28~30)の詳細説明(1)

ANSEL:ANALOG SELECT REGISTERによると
 ANSELbit = 1 → Analog I/O
 = 0 → Digital I/O

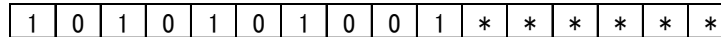
;Setting of AD Converter

```

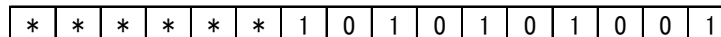
MOV LW  B'00000010'
MOV WF  ANSEL           ;AN1:Analog I/O
BCF     ADCON1, ADFM    ;Left Justified
BCF     ADCON1, ADCS2   ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF     ADCON1, VCFG1   ;Voltage Reference = VSS
BCF     ADCON1, VCFG0   ;Voltage Reference = VDD
    
```

データシートのADCON1 Register(図10のBank 1にある)によると、A/D変換結果は (ADDRESS, ADDRESSLの2つの8ビットレジスタからなる) 16ビットのレジスタに格納される。A/D変換結果は10ビットである、今、その値がB'1010101001'であるとする。16ビットのレジスタへの入れ方は

ADFM = 0 のとき左寄せ.



ADFM = 1 のとき右寄せ.
 となる.



ADCS2: 次ページに解説

VCFG1 = 0 のとき参照電圧は 5番ピンのVSS
 = 1 " 1番ピンのVREF-
 VCFG0 = 0 のとき参照電圧は 14番ピンのVDD
 = 1 " 2番ピンのVREF+

図11参照

A/D変換プログラム(本章p.28~30)の詳細説明(2)

```
BCF      STATUS,RP0
BCF      STATUS,RP1          ;Selection of Bank 0

BCF      ADCON0,ADCS1      ;
BSF      ADCON0,ADCS0      ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF      ADCON0,CHS2
BCF      ADCON0,CHS1
BSF      ADCON0,CHS0      ;CHS = 001 -> AN1
BSF      ADCON0,ADON      ;A/D converter module is operating
```

データシートのADCON0 Register(図10のBank 0にある)によると,

A/D変換モジュールのクロック周波数(F_{AD})は

ADCS2 ADCS1 ADCS0 = 000 のとき $F_{AD} = FOSC/2$
= 001 のとき $F_{AD} = FOSC/8$
= 010 のとき $F_{AD} = FOSC/32$
= 011 のとき $F_{AD} = FRC$ (内蔵のA/Dモジュール用RC発振器の出力)
= 100 のとき $F_{AD} = FOSC/4$
= 101 のとき $F_{AD} = FOSC/16$
= 110 のとき $F_{AD} = FOSC/64$
= 111 のとき $F_{AD} = FRC$ (内蔵のA/Dモジュール用RC発振器の出力)

となる。ただし、セラミック発振子が4 [MHz]のとき $FOSC = 4$ [MHz]。A/D変換モジュールのクロック周期($T_{AD} = 1/F_{AD}$)は、データシートの T_{AD} vs. Maximum Device Operating Frequencyによると、 $1.6 \mu s \leq T_{AD} \leq 6.4 \mu s$ でないといけない。よって、 $FOSC = 4$ [MHz] のとき、 T_{AD} として $8/FOSC = 2 \mu s$ 、 $16/FOSC = 4 \mu s$ の二通りの設定が選べる。上のプログラムでは $F_{AD} = FOSC/8$ を選定している。

CHS2 CHS1 CHS0 = 000 ~ 110に応じてAN0~AN6を選定 (図11参照)

ADON = 1 のときA/D変換モジュールの電源が入る。

TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh		18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADDRH	10Fh		18Fh
T1CON	10h	OSCTUNE	90h		110h		190h
TMR2	11h		91h				
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPADD	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
	1Bh	ANSEL	9Bh				
	1Ch	CMCON	9Ch				
	1Dh	CVRCON	9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96バイト	20h	General Purpose Register 80バイト	A0h	General Purpose Register 80バイト	120h	General Purpose Register 80バイト	1A0h
			Efh		16Fh		1EFh
			F0h		170h		1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

図 10 File Register Map

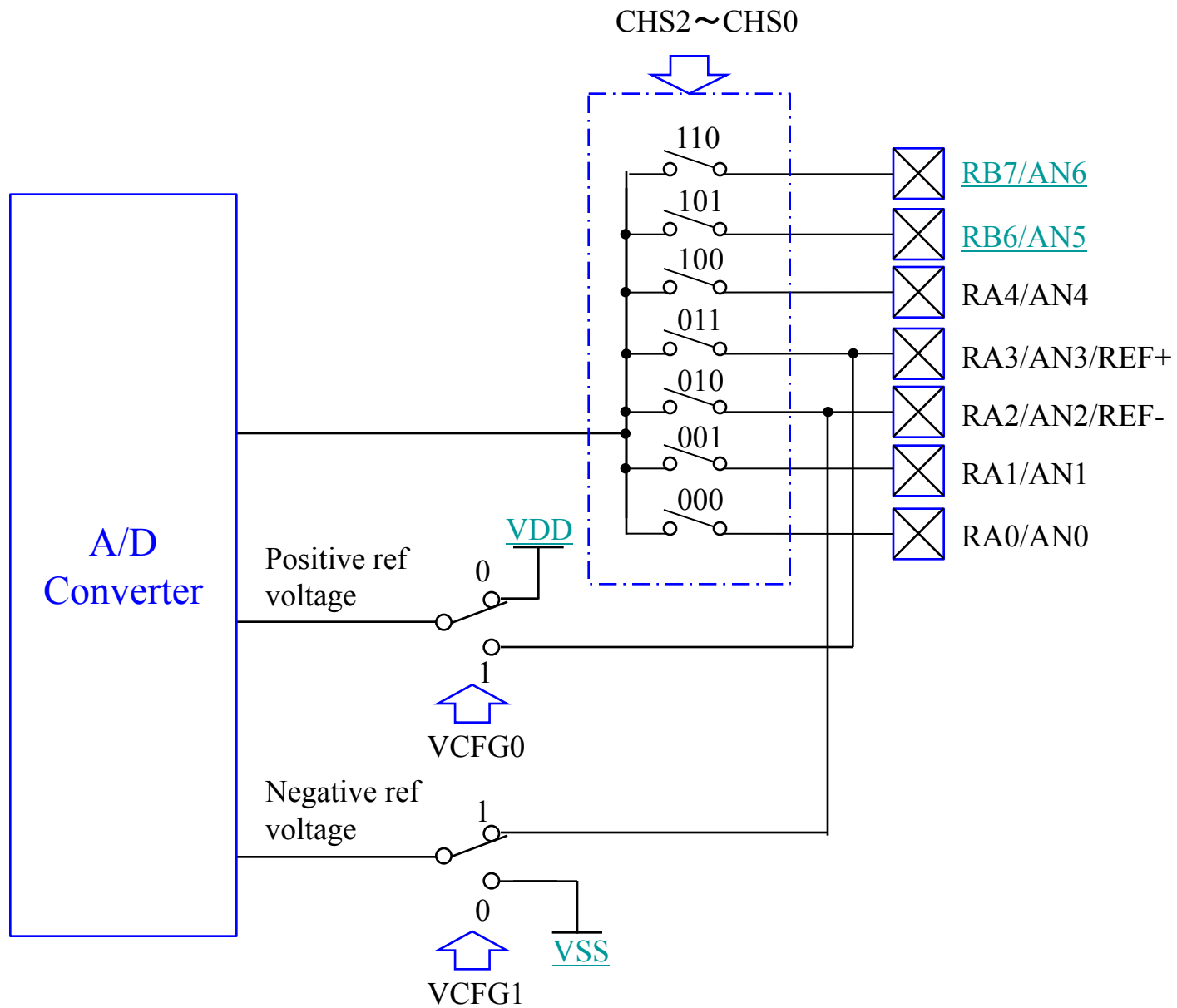


図11 A/D変換モジュールのブロック図

A/D変換プログラム(本章p.28~30)の詳細説明(3)

```
;Sub Routine1
Timer0_interrupt
    MOVLW    B'00000001'        ;'00000001' -> (W)
    MOVWF   PORTB              ;(W) -> (PORTB)

    BSF     ADCON0, GO         ;AD Conversion Start
AD_Wait  BTFSCL ADCON0, 2
    GOTO    AD_Wait           ;Wait until AD conv is done

    MOVF    ADRESH, 0         ;ADRESH -> (W)
    MOVWF   TMR0             ;(W) -> TMR0

    MOVLW   B'00000000'
    MOVWF   PORTB

    BCF     INTCON, TMR0IF    ;TMR0 register did not overflow

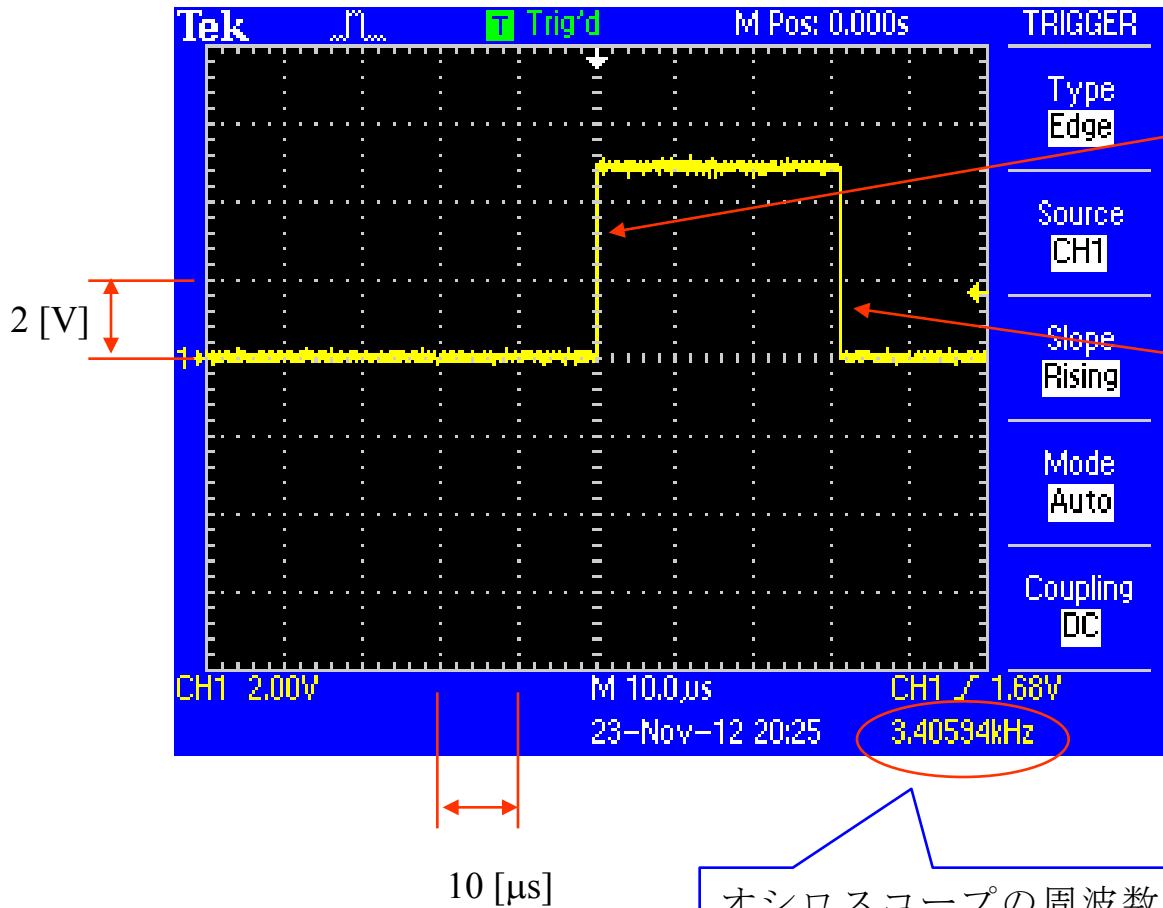
    RETURN
```

データシートのADCON0 Register([図10](#)のBank 0にある)によると、

GO = 1 とするとA/D変換モジュールはA/D変換を開始する。変換が終了するとADCON0 Registerの第2ビットが0にクリアされる。そこで、

```
AD_Wait  BTFSCL    ADCON0, 2
          GOTO     AD_Wait
```

により、変換が終了する（第2ビットが0にクリアされる）まで待つことにする。変換終了後、ADRESHには変換結果の上位8ビットが格納されているので、これを8ビットのTMR0 Registerに転送し、タイマ0のカウントアップの初期値とする。



```

MOVLW    B'00000001'
MOVWF    PORTB
BSF      ADCON0, GO
AD_Wait  BTFSCL ADCON0, 2
GOTO    AD_Wait
MOV      ADRESH, 0
MOVWF   TMR0
MOVLW   B'00000000'
MOVWF   PORTB
  
```

FOSC/4 = 1 [MHz] → 1 [μs]を1サイクルとして、各命令の実行に1サイクル、ジャンプ実行時に2サイクルを要する。A/D変換にはMax 12TAD = 12×2 [μs] = 24 [μs]を要する。
RB0(6番ピン)に1を出力してから0を出力するまでに

```

BSF      1
BTFSCL  24+2
MOV      1
MOVWF   1
MOVLW   1
MOVWF   1
  
```

計 31[μs]を要している。

A/D変換の所要時間はA/D Conversion TAD Cyclesによる。

オシロスコープの周波数カウンタ機能による割り込み周波数の計測結果。

図3のVR₁のつまみを廻すことで、割り込み周波数を15[Hz]~3.4 [kHz]の範囲で可変とする。

図12 タイマ0による割り込みプログラム(A/D変換モジュール) 実行時の6番ピンの出力波形

8α-5. ステッピングモータの速度制御

AN1(18番ピン) からアナログ電圧をA/D変換モジュールにより読み込み、タイマ0による割り込み周期をアナログ電圧に応じて可変とする。割り込み処理プログラムの中にステッピングモータの駆動信号生成プログラムを書くことで、ステッピングモータの速度制御を実現している。8α-4節のA/D変換モジュールに対して新しく追加した箇所を朱書きで示す。詳細は本章p.41以降

;AD Converter program

INCLUDE"p16F88.inc"
list p=16F88

__CONFIG __CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF
__CONFIG __CONFIG2, _FCMEN_OFF & _IESO_OFF

OutMode EQU 0x20 ;OutMode

ORG 0
GOTO START ;Main Program starts from START

ORG 4
CALL Timer0_interrupt
RETFIE

START

;Setting of Port A and B

BSF STATUS,RP0
BCF STATUS,RP1 ;Selection of Bank 1
MOVLW B'00000000'
MOVWF TRISB ;RB0-7 -> Output Port
MOVLW B'00000010'
MOVWF TRISA ;RA1 -> Input Port

;Setting of Timer0

BCF OPTION_REG, T0CS ;Internal instruction cycle clock (CLKO)
BCF OPTION_REG, PSA ;Internal instruction cycle clock (CLKO)
BSF OPTION_REG, PS2 ;
BSF OPTION_REG, PS1 ;
BSF OPTION_REG, PS0 ;Timer0 Rate 1:256

;Setting of Interrupt

```
BSF    INTCON, GIE           ;Enables all unmasked interrupts
BSF    INTCON, PEIE         ;Enables all unmasked peripheral interrupts
BSF    INTCON, TMR0IE       ;Enables the TMR0 interrupt
BCF    INTCON, TMR0IF       ;TMR0 register did not overflow
```

;Setting of AD Converter

```
MOVLW  B'00000010'
MOVWF  ANSEL                 ;AN1:Analog I/O
BCF    ADCON1, ADFM          ;Left Justified
BCF    ADCON1, ADCS2         ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF    ADCON1, VCFG1         ;Voltage Reference = VSS
BCF    ADCON1, VCFG0         ;Voltage Reference = VDD

BCF    STATUS, RP0
BCF    STATUS, RP1           ;Selection of Bank 0

BCF    ADCON0, ADCS1         ;
BSF    ADCON0, ADCS0         ;ADCS2 = 0, ADCS = 01 -> A/D Conversion Clock = FOSC/8
BCF    ADCON0, CHS2
BCF    ADCON0, CHS1
BSF    ADCON0, CHS0          ;CHS = 001 -> AN1
BSF    ADCON0, ADON          ;A/D converter module is operating
```

;Setting of Initial Value of PORTB & the Register 'OutMode'

```
MOVLW  B'00000101'         ;'00000101' -> (W)
MOVWF  PORTB                ;(W) -> (PORTB)

MOVLW  B'00000000'
MOVWF  OutMode
```

; Main Program

STEP1 GOTO STEP1

;End of Main Program

;Sub Routine1

Timer0_interrupt

 MOVLW B'00000100' ;'00000100' -> (W)
 MOVWF PORTA ;(W) -> (PORTA)

AD_Wait BSF ADCON0, GO ;AD Conversion Start

 BTFSC ADCON0, 2
 GOTO AD_Wait ;Wait until AD conv is done

 MOVF ADRESH, 0 ;ADRESH -> (W)
 MOVWF TMR0 ;(W) -> TMR0

 CALL Rotate

 MOVLW B'00000000' ;'00000000' -> (W)
 MOVWF PORTA ;(W) -> (PORTA)

 BCF INTCON, TMR0IF ;TMR0 register did not overflow

 RETURN

```

;Rotating Step Motor routine
Rotate      BTFSS      OutMode,0    ;If the 0-th bit = 1, then skip the next command line
           GOTO       CaseX0
           GOTO       CaseX1

CaseX0      BTFSS      OutMode,1    ;If the 1-th bit = 1, then skip the next command line
           GOTO       Case00
           GOTO       Case10

CaseX1      BTFSS      OutMode,1    ;If the 1-th bit = 1, then skip the next command line
           GOTO       Case01
           GOTO       Case11

Case00      MOVLW      B'00000110'  ; '00000101' -> '00000110'
           GOTO       Fin

Case01      MOVLW      B'00001010'  ; '00000110' -> '00001010'
           GOTO       Fin

Case10      MOVLW      B'00001001'  ; '00001010' -> '00001001'
           GOTO       Fin

Case11      MOVLW      B'00000101'  ; '00001001' -> '00000101'

Fin         MOVWF      PORTB        ;(W) -> (PORTB)

           INCF       OutMode,1    ;(OutMode) + 1 -> (OutMode)

           RETURN

           END

```


ステッピングモータの速度制御プログラム(本章p.37~40)の詳細説明(1)

;AD Converter program

```
INCLUDE"p16F88.inc"  
list p=16F88
```

```
__CONFIG _CONFIG1, _HS_OSC & _WDT_OFF & _PWRTE_OFF & _CP_OFF & _LVP_OFF  
__CONFIG _CONFIG2, _FCMEN_OFF & _IESO_OFF
```

```
OutMode EQU 0x20 ;OutMode
```

汎用レジスタ(General Purpose Register)のBank 1
の 20h番地のレジスタをOutModeと命名

;Setting of Initial Value of PORTB & the Register 'OutMode'

```
MOVLW B'00000101' ;'00000101' -> (W)  
MOVWF PORTB ;(W) -> (PORTB)
```

```
MOVLW B'00000000'  
MOVWF OutMode
```

レジスタOutModeの初期値を0
に設定

RB3~RB0の初期値を設定.
RB3に 0, RB2に1, RB1に0, RB0に1
を出力.

ステッピングモータの速度制御プログラム(本章p.37~40)の詳細説明(2)

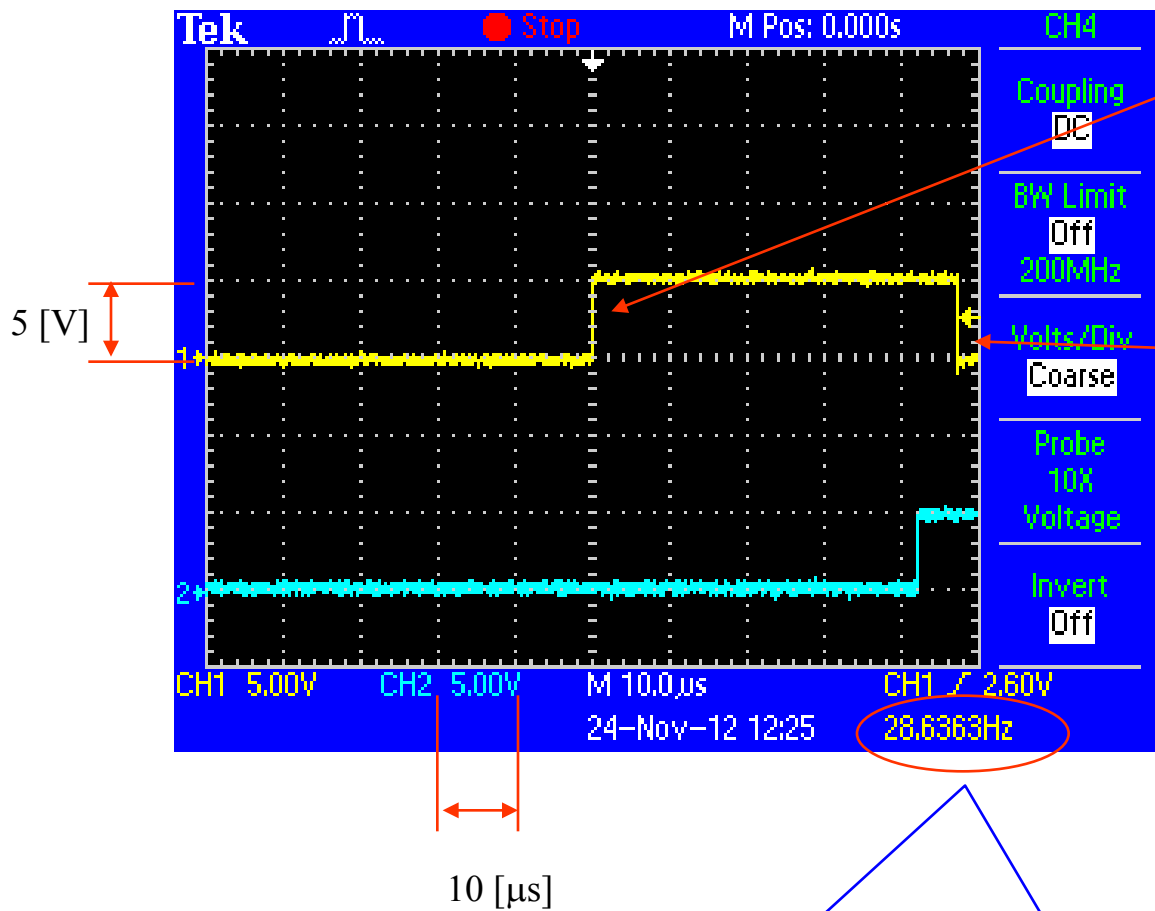
```
;Sub Routine1  
Timer0_interrupt
```

```
                MOVLW    B'00000100'        ;'00000100' -> (W)  
                MOVWF    PORTA              ;(W) -> (PORTA)  
  
                BSF      ADCON0, GO         ;AD Conversion Start  
AD_Wait  
                BTFSC    ADCON0, 2  
                GOTO     AD_Wait           ;Wait until AD conv is done  
  
                MOVF     ADRESH, 0         ;ADRESH -> (W)  
                MOVWF    TMR0             ;(W) -> TMR0  
  
                CALL     Rotate  
  
                MOVLW    B'00000000'        ;'00000000' -> (W)  
                MOVWF    PORTA              ;(W) -> (PORTA)  
  
                BCF      INTCON, TMR0IF    ;TMR0 register did not overflow  
  
                RETURN
```

RA2 (1番ピン)に
1を出力.

P.40のサブルーチン
をコール.

RA2に0を出力.



```

MOVWF PORTA
BSF     ADCON0, GO

BTFSC  ADCON0, 2
GOTO   AD_Wait

MOVF   ADRESH, 0
MOVWF  TMR0
CALL   Rotate
MOVLW  B'00000000'
MOVWF  PORTA

```

AD_Wait

BSF	1
BTFSC	24 + 2
MOVF	1
MOVWF	1
CALL	15 or 16
MOVLW	1
MOVWF	1
計	46 or 47[μs]

を要している。

図13 タイマ0による割り込みプログラム(A/D変換モジュール+ステッピングモータの速度制御) 実行時のRA2(1番ピン)の出力波形(黄)とRB0(6番ピン)の出力波形(青)

図3のVR₁のつまみを廻すことで、割り込み周波数を15[Hz]~3.4 [kHz]の範囲で可変とする。図1のようにステッピングモータの電源電圧12 [V]の場合、タイマ0による割り込み周波数が760[Hz]を超えると、モータはトランジスタのオン/オフの切換の速さ(760/4 = 190 [Hz])について行けなくなって止まってしまう。これを脱調という。

トランジスタTr₂~Tr₄の駆動波形

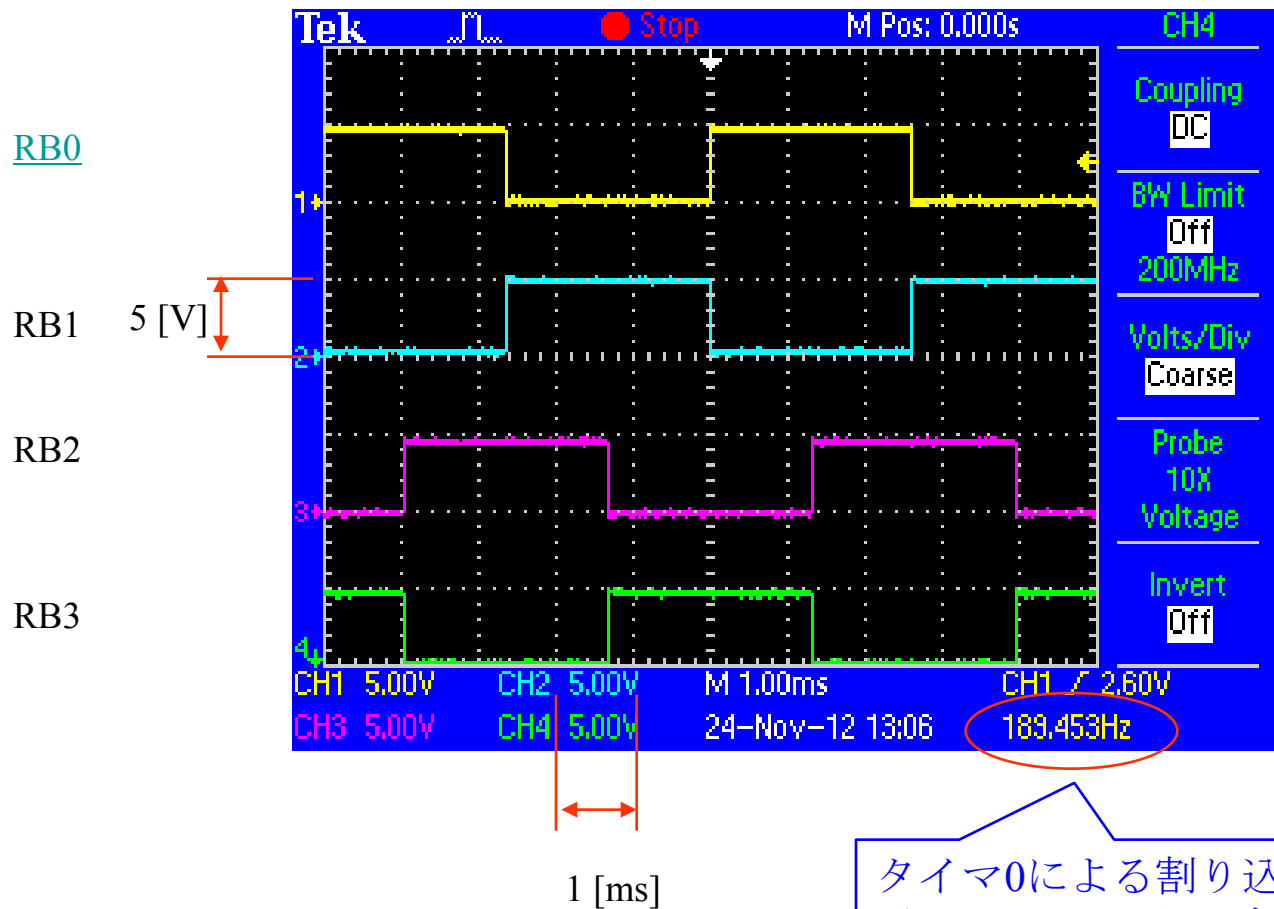


図14 タイマ0による割り込みプログラム(A/D変換モジュール+ステッピングモータの速度制御) 実行時のRB0~RB3(6~9番ピン)の出力波形(順に黄, 青, 紫, 緑)

タイマ0による割り込み周波数が760[Hz]のとき, 各トランジスタのオン/オフの切換周波数は $760/4 = 190$ [Hz]である.

2012年11月

著者： 古橋武
名古屋大学工学研究科計算理工学専攻
furuhashi@cse.nagoya-u.ac.jp