

;"Hello" in Japanese on LCD

```

        INCLUDE "p16F84.inc"
        list p=16F84

Memory   EQU      0x0C
DX        EQU      Memory+0      ; "DX" File Register for LCD
DXX       EQU      Memory+1      ; "DXX" File Register for LCD
WORK1    EQU      Memory+2      ; File Register for Cursor Position (1st line)
WORK2    EQU      Memory+3      ; File Register for Cursor Position (2nd line)
TIME1    EQU      Memory+4
TIME2    EQU      Memory+5
CHARA    EQU      Memory+6
CUR1     EQU      Memory+7
CUR2     EQU      Memory+8

        ORG      0
        GOTO    START
        ORG      4

START

        MOVLW   B'10000000'
        MOVWF   CUR1      ;Cursor position(1st line)
        MOVLW   B'11000000'
        MOVWF   CUR2      ;Cursor position(2nd line)

;-----Port Setting-----
        BSF     STATUS, RP0      ;selection of Bank1
        MOVLW   0x00
        MOVWF   TRISA           ;PORTA: output
        MOVLW   B'00000111'
        MOVWF   TRISB           ;RB0-2: input, RB3-7: output
        BCF     STATUS, RP0      ;selection of Bank0

;-----
;-----
        CALL    ini_lcd         ;Initialization
;-----

```

液晶は16文字×2行の文字を表示できる。
 上の行は左からB'10000000'番地から
 B'10001111'番地が割り当てられ、
 下の行は左からB'11000000'番地から
 B'11001111'番地が割り当てられている。
 ここではCUR1に液晶の上の行の左端の番地を、
 CUR2に下の行の左端の番地を指定している。

液晶の初期化サブプログラム
 液晶の初期化はややこしいので
 サブプログラムをそのまま
 使ってください。

PORTAのRA0-4を出力ポートに指定。
 RA0 E (1:液晶に表示用データ転送可
 0:表示用データ転送不可)
 RA1 RS (1:番地指定モード
 0:データ転送モード)
 PORTBのRB0-2を入力ポートに指定。
 (SW1 RB0, SW2 RB1, SW3 RB2)
 PORTBのRB3-7を出力ポートに指定。
 (RB4-7:液晶表示用データを出力)

-----Main Program-----

```
MOVF CUR1,0 } ;Cursor position on the 1st line
MOVWF WORK1 }
CALL SETMODE } ;Cursor position setting routine
MOVLW 0xBA } ;Chara Code -> Working Register
CALL PUTCHAR } ;Displaying "Chara" routine
;-----
INCF WORK1,1 }
MOVF WORK1,0 }
CALL SETMODE }
MOVLW 0xDD }
CALL PUTCHAR }
;-----
INCF WORK1,1 }
MOVF WORK1,0 }
CALL SETMODE }
MOVLW 0xC6 }
CALL PUTCHAR }
;-----
INCF WORK1,1 }
MOVF WORK1,0 }
CALL SETMODE }
MOVLW 0xC1 }
CALL PUTCHAR }
;-----
INCF WORK1,1 }
MOVF WORK1,0 }
CALL SETMODE }
MOVLW 0xCA }
CALL PUTCHAR }
;-----
MOVF CUR2,0 } ;Cursor position on the 2nd line
MOVWF WORK2 }
CALL SETMODE } ;Cursor position setting routine
MOVLW 0xD6 } ;Chara Code -> Working Register
CALL PUTCHAR } ;Displaying "Chara" routine
;-----
INCF WORK2,1 }
MOVF WORK2,0 }
CALL SETMODE }
MOVLW 0xDB }
CALL PUTCHAR }
;-----
INCF WORK2,1 }
MOVF WORK2,0 }
CALL SETMODE }
MOVLW 0xBC }
CALL PUTCHAR }
;-----
INCF WORK2,1 }
MOVF WORK2,0 }
CALL SETMODE }
MOVLW 0xB8 }
CALL PUTCHAR }
LOOP1 GOTO LOOP1
```

液晶の上の行の左端の番地を指定

指定番地に “コ” のデータを転送
液晶は自動的に “コ” を表示する .

液晶の上の行の左から 2 番目の番地を指定

指定番地に “ン” のデータを転送
液晶は自動的に “ン” を表示する .

BA = “コ”
DD = “ン”
C6 = “ニ”
C1 = “チ”
CA = “ハ”
D6 = “ヨ”
DB = “ロ”
BC = “シ”
B8 = “ク”

;Cursor position on the 2nd line
;Cursor position setting routine
;Chara Code -> Working Register
;Displaying "Chara" routine

```

;-----Cursor Pos. Setting Mode-----
SETMODE      BCF      PORTA,1
              MOVWF   DXX
              CALL    WR_LCD4
              RETURN

```

;RS = 0 (LCD: Cursor position setting mode)

RA1='0' RS='0':番地指定モード
ワーキングレジスタにある番地を液晶に指定するサブプログラム
WR_LCD4のサブプログラムはDXXに入れた内容の8ビットを4ビットずつ液晶に書き込むサブプログラムであるが、ややこしいのでそのまま使ってください。

```

;----Character Display Sub Program-----
PUTCHAR      BSF      PORTA,1
              MOVWF   DXX
              CALL    WR_LCD4
              RETURN

```

RA1='1' RS='1':データ転送モード
ワーキングレジスタにある文字コードを液晶に送り表示するサブプログラム

```

WR_LCD8      MOVF      DX,0
              MOVWF   PORTB
              NOP
              BSF     PORTA,0
              NOP
              BCF     PORTA,0
              MOVLW  D'20'
              call   wait_ms
              RETURN

```

WR_LCD4のサブプログラムはDXXに入れた内容の8ビットを4ビットずつ液晶に書き込むサブプログラムであるが、ややこしいのでそのまま使ってください。

```

WR_LCD4      MOVF      DXX,0
              MOVWF   DX
              CALL    WR_LCD8
              SWAPF   DX,1
              CALL    WR_LCD8
              RETURN

```

```

;-----Waiting time Sub Program-----
wait_ms      MOVWF   TIME1
STEPM        MOVLW   0xFF
              MOVWF   TIME2
STEPM1       MOVLW   0xFF
              DECFSZ  TIME2,1
              GOTO    STEPM1
              DECFSZ  TIME1,1
              GOTO    STEPM
              RETURN

```

時間稼ぎのサブプログラム。
このサブプログラムをコールするときにワーキングレジスタにある値をTIME1に入れ、'TIME1の値' × 0xFF × 0xFF回 GOTO STEPM1 ↔ STEPM1の間の計算を繰り返す。

```

;-----Initialization-----
ini_lcd    MOVLW    D'255'
           call     wait_ms
           MOVLW    B'00000000'
           MOVWF    PORTA
           MOVLW    B'00110000'
           MOVWF    DX
           CALL     WR_LCD8
           MOVLW    D'255'
           call     wait_ms
           MOVLW    B'00110000'
           MOVWF    DX
           CALL     WR_LCD8
           MOVLW    D'255'
           call     wait_ms
           MOVLW    B'00110000'
           MOVWF    DX
           CALL     WR_LCD8

           MOVLW    B'00100000'
           MOVWF    DX
           CALL     WR_LCD8
           MOVLW    B'00101000'
           MOVWF    DXX
           CALL     WR_LCD4
           MOVLW    B'00000001'
           MOVWF    DXX
           CALL     WR_LCD4
           MOVLW    D'255'
           call     wait_ms
           MOVLW    B'00000110'
           MOVWF    DXX
           CALL     WR_LCD4
           MOVLW    B'00001110'
           MOVWF    DXX
           CALL     WR_LCD4
           RETURN
           END

```

液晶初期化サブプログラム
 ややこしいのでそのまま使ってください。