

'TEMP\_PICMB876 a/d result to rs-232c

```
' Program Name   TEMP_PICMB876.BAS
' Programed by  Tech.Hanzougane Yoshiaki Morohashi
' Date          2002.10.06
' Language      microEngineering Labs, Inc. PICBasic Pro Compiler
' Target Device PIC16F877 on TEMP_PICMB876
```

Include "modedefs.bas"

```
DEFINE ADC_BITS 8           'A/D converter definition
DEFINE ADC_CLOCK 2
DEFINE ADC_SAMPLEUS 50
```

```
OUTPUT PORTA.5
INPUT PORTA.4
SO Var PORTA.5             'Serial Out portB pin RB5
SI Var PORTA.4             'Serial Input portB pin RB4
RC Var PORTC
```

```
OUTPUT PORTC.0             'LCD Back light control bit
OUTPUT PORTC.6
INPUT PORTC.7
SOC6 Var PORTC.6           'Serial Out portc pin RC6
SIC7 Var PORTC.7           'Serial Input portc pin RC7
```

```
define lcd_dreg    portb
define lcd_dbit    0
define lcd_rsreg   portb
define lcd_rsbit   4
define lcd_ereg    portb
define lcd_ebit    5
define lcd_bits    4
define lcd_lines   2
```

```
ADRH Var byte
ADRL Var byte
w0 Var word
wh Var word
whrmd Var word
whrmd2 Var byte
ADRH = 0
ADRL = 0
w0 = 0
wh = 0
whrmd = 0
whrmd2 = 0
```

'-----

```
CPIN var PORTB.6          ' I2C clock pin
DPIN var PORTB.7          ' I2C data pin

B0 var WORD               '24LC256 needs WORD size address
BWRITE var byte           'BWRITE is WRITE data
BREAD var byte            'BREAD is READ data
```

'-----

```

                                temp2.txt
                                'Execute program branch

goto Tx

'Transmit rs-232c temp data

Tx:      SEROUT SO,T9600,["          ",13,10]      'Dummy SEROUT
        PAUSE 1000
        SEROUT SO,T9600,["PICMB876 A/D_temp SEROUT & 24LC256memory Sample",13,10]

        RC.0 = 0

Txrs:    FOR B0 = 0 TO 32766
        RC.0 = 0
        ADRH=0:ADRL=0:wh=0:w0=0
        TRISA = %00011111      '%0001 1111
        ADCON1 = %10000101     '%1000 0101
        ADCIN 0,wh             'ADC conversion START
        ADRH = ADRESH          'A/D result move ADRH
        ADRL = ADRESL          'A/D result move ADRL
        w0 = ADRH * 256 + ADRL
        whrmd = (w0 * 61) // 1250
        whrmd = (whrmd * 80) / 1000
        whrmd2 = whrmd / 10
        w0 = (w0 * 61) / 1250
        IF w0 < 30 Then GOTO LWLIM
        RC.0 = 1
        SEROUT SO,T9600,[#w0, ". " ,#whrmd2,13,10]
LWLIM:   BWRITE = w0             'BWRITE(byte size)<-- w0(word size)
        I2CWRITE DPIN,CPIN,$A0,B0,[BWRITE]
        PAUSE 10                'EEPROM Write Delay(10ms)
        LCDOUT $FE,1, "W_ing..Adrs=",#B0
        LCDOUT $FE,$C0,"WData=",#BWRITE
        PAUSE 500               'LCD display time(500ms)

        NEXT B0

        LCDOUT $FE,1             'LCD cleare

        I2CREAD DPIN,CPIN,$A0,B0,[BREAD]
        PAUSE 10                'EEPROM Read Delay(10ms)
        LCDOUT $FE,1, "R_ing..Adrs=",#B0
        LCDOUT $FE,$C0,"RData=",#BREAD
        PAUSE 500               'LCD display time(500ms)

        PAUSE 2000              'wait 2 sec

        GOTO Txrs

```

.