BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM

KHOA ĐIỆN – ĐIỆN TỬ

BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP

VI XỬ LÝ 2

ĐỀ TÀI: ***BÁO CÁO VI XỬ LÝ II***

LỚP 061012

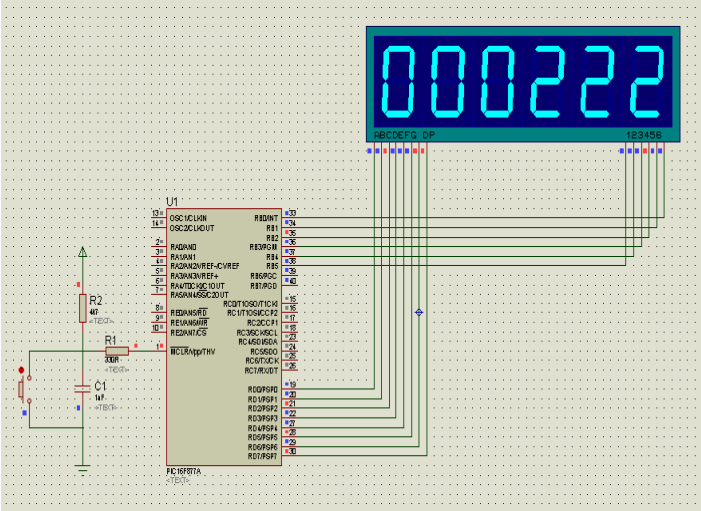
Người soạn: Nguyễn Duy Tưởng

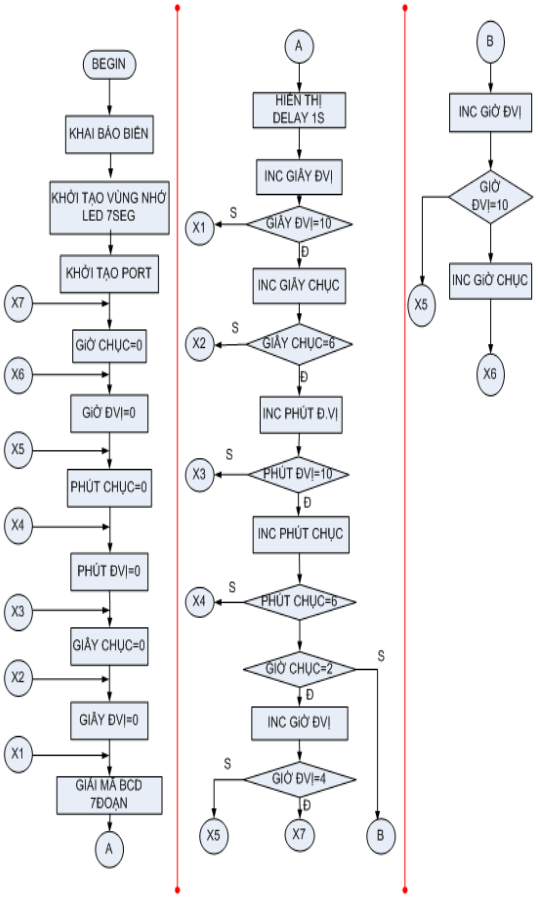
Tp. Hồ Chí Minh 1/2010

MỤC LỤC

BÁO CÁO: VI XU LY 2

BÀI 1 : THIẾT KẾ ĐỒNG HỒ SỐ DÙNG PIC 16F877A ĐẾM GIỜ, PHÚT ,GIÂY DÙNG NGÔN NGỮ ASSEMBLY





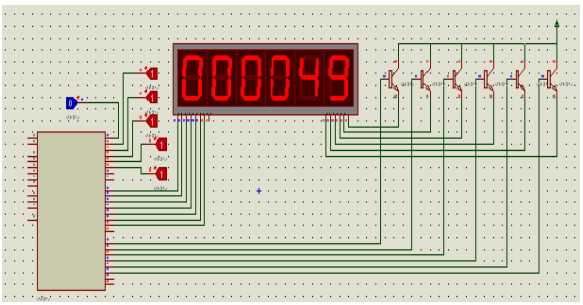

```
; BAO CAO VI XU LY 2
;Ten chuong trinh: dong ho so dung VDK Pic 16F877A
;***********************************************
TITLE " dong ho so dung PIC 16F877A"
PROCESSOR P16F877A
INCLUDE <P16F877A.inc>
__CONFIG _CP_OFF & _PWRTE_ON & _WDT_OFF &
_HS_OSC & _LVP_OFF
;***********************************************
;chuong trinh
;---------------------------------------------
;---------- khai bao bien----------
CBLOCK              0x020
COUNT1
COUNT2
COUNT3
GIAY_DV
GIAY_CHUC
PHUT_DV
PHUT_CHUC
GIO_DV
GIO_CHUC
BIEN_GIAY_DV
BIEN_GIAY_CHUC
BIEN_PHUT_DV
BIEN_PHUT_CHUC
BIEN_GIO_DV
BIEN_GIO_CHUC
ENDC
;*************************************
ORG                 0x000
CLRF                STATUS
MOVLW               0x00
MOVWF               PCLATH
GOTO                START
;*************************************
;-----------------------------------------------------
; KHOI TAO
*****************************************
;KHOI TAO PORT B DIEU KHIEN 6 TRANSISTOR QUET HANG
;KHOI TAO PORT D KET NOI VOI 8 CHAN CUA LED 7 DOAN
```

```
START       ORG 0x006
BCF         STATUS,RP1
BSF         STATUS,RP0      ; LUA CHON BANK 1
CLRF        TRISB           ; PORT B XUAT DATA
CLRF        TRISD           ; PORT D XUAT DATA
BCF         STATUS,RP0      ; TRO VE BANK 0

;**************************************
;chuongtrinhchinh

;**************************************
;=======KHOI TAO GIA TRI BAN DAU========
X7    MOVLW   0x00
      MOVWF   GIO_CHUC  ;CHO GIO PHUT GIAY = 0
X6    MOVLW   0x00
      MOVWF   GIO_DV
X5    MOVLW   0x00
      MOVWF   PHUT_CHUC
X4    MOVLW   0x00
      MOVWF   PHUT_DV
X3    MOVLW   0x00
      MOVWF   GIAY_CHUC
X2    MOVLW   0x00
      MOVWF   GIAY_DV

;===================================

X1    CALL    BCD_7DOAN
      CALL    DELAY_HIEN_THI
;================================

INCF      GIAY_DV       ;TANG GIAY_DV
MOVF      GIAY_DV,0     ;NAP GIAY_DV VAO  T G  W
XORLW     D'10'
BTFSS     STATUS,Z      ;kiem tra bit Z va nhay neu
Z=1(bo qua lenh GOTO) co nghia la giay don vi=10
GOTO      X1

INCF      GIAY_CHUC     ;TANG GIAY_CHUC
MOVF      GIAY_CHUC,0   ;NAP GIAY_CHUC VAO TG W
XORLW     D'6'          ;SO SANH GIAY_CHUC VOI 6,Z=1
NEU GIAY_CHUC=6,NGUOC LAI Z=0
BTFSS     STATUS,Z      ;KIEM TRA Z,NHAY KHI Z=1
GOTO      X2            ;VE X2 NEU Z=0

INCF      PHUT_DV       ;TANG PHUT_DV
MOVF      PHUT_DV,0     ;NAP PHUT_DV VAO TG W
XORLW     D'10'         ;SO SANH PHUT_DV VOI 10,Z=1
NEU GIAY_CHUC=10,NGUOC LAI Z=0
BTFSS     STATUS,Z      ;KIEM TRA Z,NHAY KHI Z=1
GOTO      X3            ;VE X3 NEU Z=0

INCF      PHUT_CHUC     ;TANG PHUT_CHUC
MOVF      PHUT_CHUC,0   ;NAP PHUT_CHUC VAO TG W
XORLW     D'6'          ;SO SANH PHUT_CHUC VOI
6,Z=1 NEU GIAY_CHUC=6,NGUOC LAI Z=0
BTFSS     STATUS,Z      ;KIEM TRA Z,NHAY KHI Z=1
GOTO      X4            ;VE X4 NEU Z=0
;=======================================
MOVF      GIO_CHUC,0
XORLW     D'2'
BTFSS     STATUS,Z
GOTO      X10
GOTO      X11

X10   INCF      GIO_DV
      MOVF      GIO_DV,0
      XORLW     D'10'
      BTFSS     STATUS,Z
      GOTO      X5
      INCF      GIO_CHUC
      GOTO      X6
X11   INCF      GIO_DV
      MOVF      GIO_DV,0
      XORLW     D'4'
      BTFSS     STATUS,Z
      GOTO      X5
      GOTO      X7
;=======================================
BCD_7DOAN
MOVF      GIAY_DV,0     ;MOV (GIAY_DV) VAO W
CALL      TABLE
MOVWF     BIEN_GIAY_DV  ;MOV (GIAY_DV) VAO
BIEN_GIAY_DV

MOVF      GIAY_CHUC,0   ;MOV (GIAY_CHUC)
VAO W
CALL      TABLE
MOVWF     BIEN_GIAY_CHUC ;MOV (GIAY_CHUC)
VAO BIEN_GIAY_CHUC

MOVF      PHUT_DV,0     ;MOV (PHUT_DV) VAO W
CALL      TABLE
MOVWF     BIEN_PHUT_DV  ;MOV (PHUT_DV)VAO BIEN
PHUT_DV

MOVF      PHUT_CHUC,0   ;MOV (PHUT_CHUC)
VAO W
CALL      TABLE
MOVWF     BIEN_PHUT_CHUC ;MOV (PHUT_CHUC)
VAO BIEN_PHUT_CHUC
```

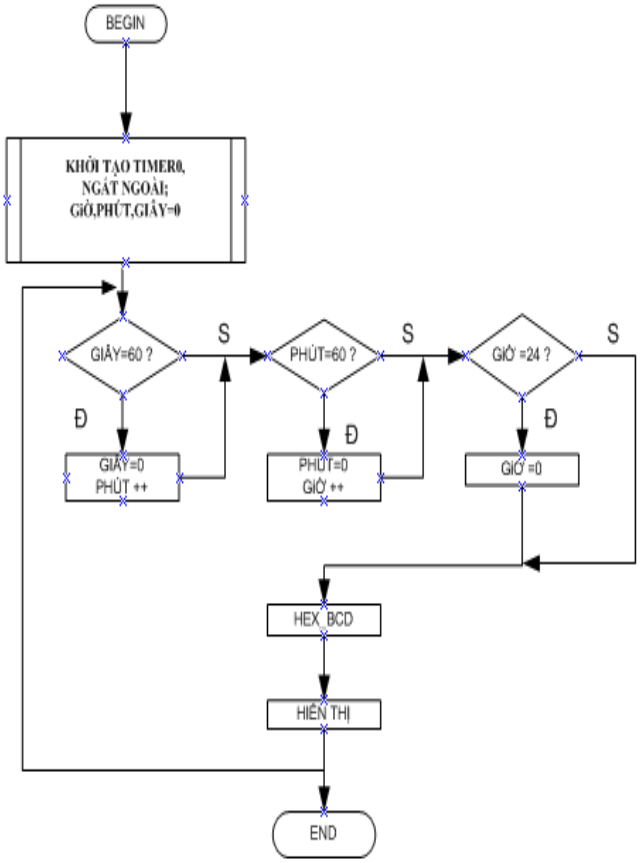
```
MOVF        GIO_DV,0          ;MOV (GIO_DV) VAO W
CALL        TABLE
MOVWF   BIEN_GIO_DV     ;MOV (GIO_DV) VAO BIEN_GIO_DV

MOVF        GIO_CHUC,0      ;MOV (GIO_CHUC) VAO W
CALL        TABLE
MOVWF   BIEN_GIO_CHUC     ;MOV (GIO_CHUC) VAO BIEN_GIO_CHUC

RETURN

;=========================================
DELAY_HIEN_THI
MOVLW       0x2             ;NAP 01 VAO W
MOVWF       COUNT1          ;MOV 01 VA0 COUNT1
DEL1
MOVLW       0x18            ;NAP 1 VAO W
MOVWF       COUNT2          ;MOV 1 VAO COUNT2
DEL2
CALL        HIENTHI
DECFSZ      COUNT2          ;GIAM COUNT2,KHI BANG 0 THI NHAY XUONG GIAM COUNT1
GOTO DEL2
DECFSZ      COUNT1          ;GIAM COUNT1,KHI BANG 0 THI BO QUA LENH KE
GOTO        DEL1
RETURN

;=========================================
HIENTHI
MOVF        BIEN_GIAY_DV,0      ;HIEN THI GIAY-DONVI
MOVWF           PORTD
MOVLW           0x1
MOVWF           PORTB
CALL            DELAY
;==================================
MOVLW           0X00            ;CHONG LEM
MOVWF           PORTB
MOVLW           0xff
MOVWF           PORTD
;==================================
MOVF        BIEN_GIAY_CHUC,0    ;HIEN THI GIAY_CHUC
MOVWF           PORTD
MOVLW           0x2
MOVWF           PORTB
CALL            DELAY

MOVLW           0X00
MOVWF           PORTB
MOVLW           0xff
MOVWF           PORTD

MOVF        BIEN_PHUT_DV,0  ;HIEN THI PHUT-DON VI
MOVWF           PORTD
MOVLW           0x4
MOVWF           PORTB
CALL            DELAY

MOVLW       0X00
MOVWF       PORTB
MOVLW       0XFF
MOVWF       PORTD

MOVF        BIEN_PHUT_CHUC,0    ;HIÊN THI PHUT_CHUC
MOVWF       PORTD
MOVLW       0x8
MOVWF       PORTB
CALL        DELAY

MOVLW       0X00
MOVWF       PORTB
MOVLW       0XFF
MOVWF       PORTD

MOVF        BIEN_GIO_DV,0       ;HIEN THI GIO_DONVI
MOVWF       PORTD
MOVLW       0x10
MOVWF       PORTB
CALL            DELAY

MOVLW       0X00
MOVWF       PORTB
MOVLW       0XFF
MOVWF       PORTD

MOVF        BIEN_GIO_CHUC,0     ;HIEN THI GIO_CHUC
MOVWF       PORTD
MOVLW       0x20
MOVWF       PORTB
CALL        DELAY

MOVLW       0X00
MOVWF       PORTB
MOVLW       0XFF
MOVWF       PORTD

RETURN
;=======================================
DELAY
MOVLW       0xFF
MOVWF       COUNT3
DELAY1
DECFSZ      COUNT3
```

GOTO DELAY1
RETURN
;=====================================
TABLE
ADDWF PCL,1
DT 0xC0, 0xF9, 0xA4, 0xB0, 0x99, 0x92, 0x82, 0xF8, 0x80, 0x90
END

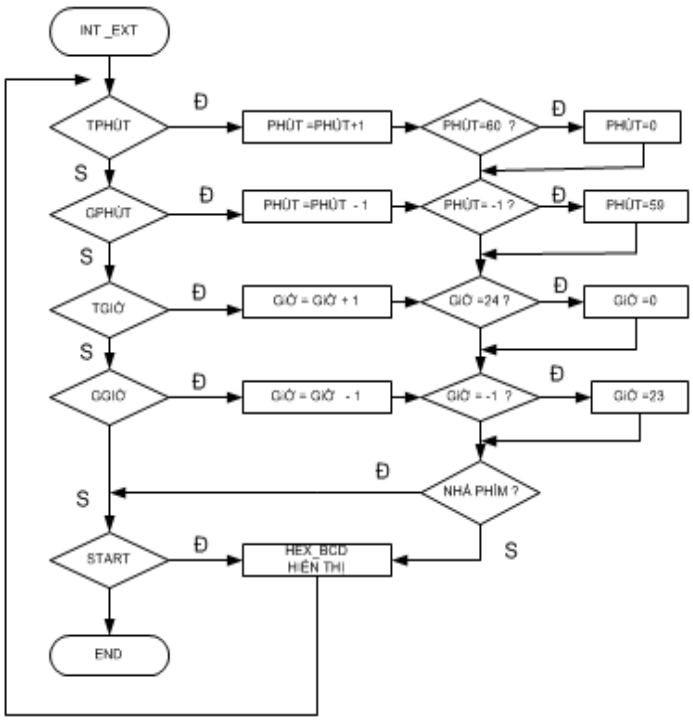
Bài 2:Dùng PIC 16F877a điều khiển 6 led 7 đoạn đếm giờ ,phút ,giây. Dùng C



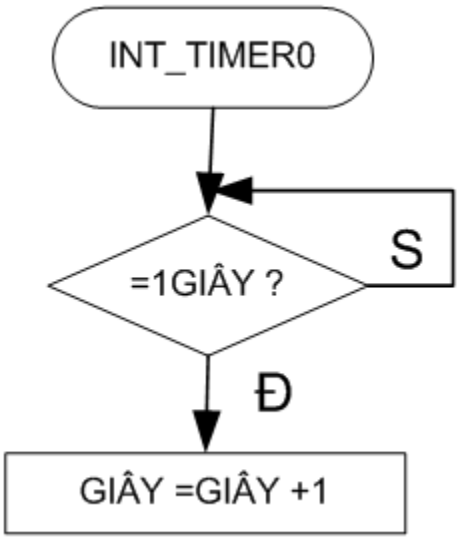
chương trình chính



Chương trình ngắt



Ngắt timer 0

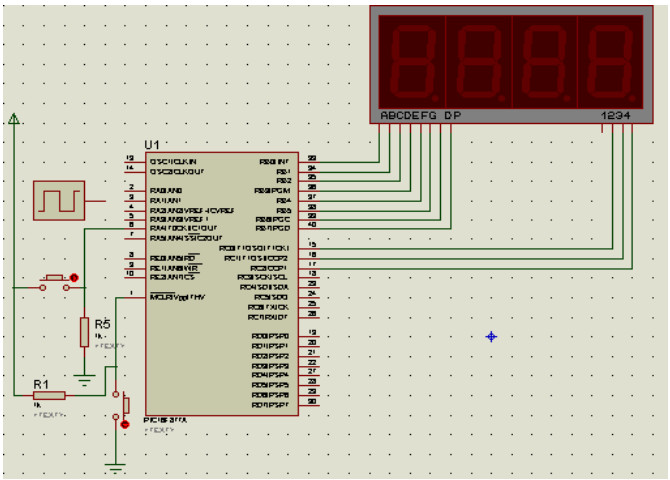


CHƯƠNG TRÌNH

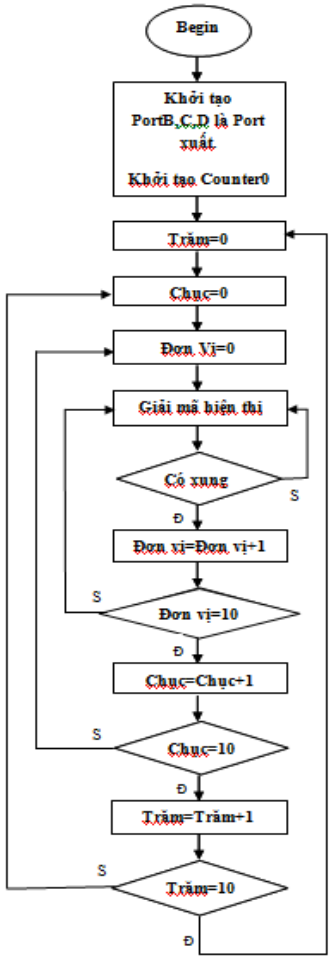
```
#include <16F877A.h>
#include<khaibaotghi_16f877a.h>
#fuses NOWDT,PUT,XT,NOPROTECT,HS,NOLVP
#use delay(clock=1000000)
#use fast_io(b)
#use fast_io(d)
#use fast_io(c)
#bit start=portb.1
#bit tphuc=portb.2
#bit gphuc=portb.3
#bit tgio=portb.4
#bit ggio=portb.5
int16 count;
int8
giay,phuc,gio,dvgiay,chgiay,dvphuc,chphuc,dvgio,chgio,i;
const unsigned char
dig[]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x9
0}
  //ma 7 doan
void hex_bcd()
{chgiay=giay/10;
  dvgiay=giay%10;
  chphuc=phuc/10;
  dvphuc=phuc%10;
  chgio=gio/10;
  dvgio=gio%10;}

void nhienthi()
{ i=0;
while(i<5)
  { output_high(pin_d0);
    portc=dig[dvgiay];
    delay_ms(1);
    output_low(pin_d0);
    output_high(pin_d1);
    portc=dig[chgiay];
    delay_ms(1);
    output_low(pin_d1);

    output_high(pin_d2);
    portc=dig[dvphuc];
    delay_ms(1);
    output_low(pin_d2);

    output_high(pin_d3);
    portc=dig[chphuc];
```

```
    delay_ms(1);
    output_low(pin_d3);

    output_high(pin_d4);
    portc=dig[dvgio];
    delay_ms(1);
    output_low(pin_d4);

    output_high(pin_d5);
    portc=dig[chgio];
    delay_ms(1);
    output_low(pin_d5);
    i=i+1;  }}
#int_ext
void ngat_ngoai()
{ while (true)
  {if (tphuc==0)
    {phuc=phuc+1;
      goto loop;
    }
    if (gphuc==0)
    {phuc=phuc-1;
      goto loop;
    }
    if (tgio==0)
    {gio=gio+1;
      goto loop;
    }
    if (ggio==0)
    { gio=gio-1;
      goto loop; }
      goto lap1;

loop:
    if (phuc==60)
    {phuc=0; }
    if (phuc==-1)
    { phuc=59;}
     if (gio==24)
    { gio=0; }
    if (gio==-1)
    {  gio=23;}
lap:  nhienthi();
    if (tphuc==0||gphuc==0||tgio==0||ggio==0)
      {goto lap;}
lap1:
      if (start==0)
      { break; }
      hex_bcd();
      nhienthi(); }}
//Chuong trinh ngat TMR0
#int_timer0
void interrupt_timer0()
{ set_timer0(6);
  ++count;
  if(count ==500)
  {count=0;
    giay=giay+1;  }}

void hienthi()
{ i=0;
  while(i<5)
  {output_high(pin_d0);
    portc=dig[dvgiay];
    delay_ms(1);
    output_low(pin_d0);
    output_high(pin_d1);
    portc=dig[chgiay];
    delay_ms(1);
    output_low(pin_d1);
    output_high(pin_d2);
    portc=dig[dvphuc];
    delay_ms(1);
    output_low(pin_d2);
    output_high(pin_d3);
    portc=dig[chphuc];
    delay_ms(1);
    output_low(pin_d3);
    output_high(pin_d4);
    portc=dig[dvgio];
    delay_ms(1);
    output_low(pin_d4);
    output_high(pin_d5);
    portc=dig[chgio];
    delay_ms(1);
    output_low(pin_d5);
    i=i+1; }}
//Chuong trinh chinh
```

```
void main(void)
{ trisb=1;
  trisd=0;
  trisc=0;
  intcon=0xb0;//ngat ngoai va ngat t0
  enable_interrupts(int_timer0);//chopheptmr0
  setup_timer_0(RTCC_INTERNAL|RTCC_DIV_2);
   enable_interrupts(global);
  giay =0;
  phuc=0;
  gio=0;
while(true){
    if(giay==60)
        { giay=0;
        phuc=phuc+1; }
   if (phuc==60)
       {phuc=0;
       gio=gio+1;}
   if (gio==24)
      { gio=0;}
   hex_bcd();
   hienthi();}}
```



LƯU ĐỒ

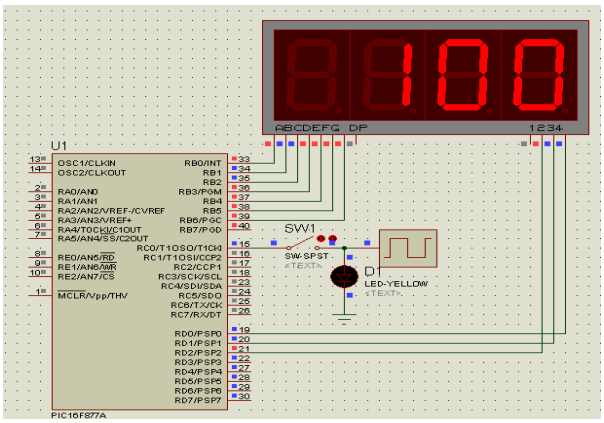


BÀI 3: SỬ DỤNG VI ĐIỀU KHIỂN PIC16F877A ĐỂ ĐẾM SẢN PHẨM SỬ DỤNG TIMER0.SỬ DỤNG NGÔN NGỮ ASSEMBLY

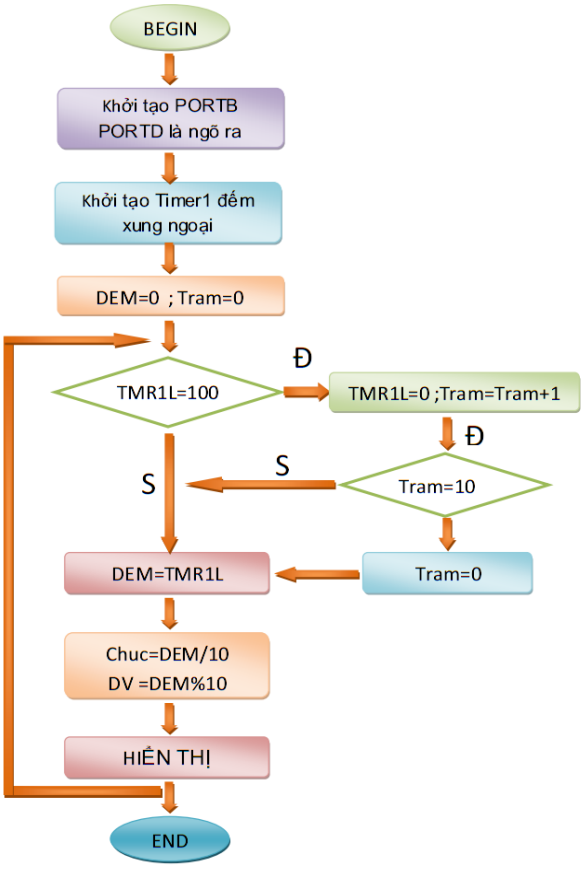
CHƯƠNG TRÌNH

```
title "dem san pham"
processor p16f877a
include <P16f877a.inc>
__CONFIG _WDT_OFF
;==================================
;Chuong trinh chinh
;==================================
;----------khai bao bien--------
      dvi       equ    0x20
      chuc      equ    0x21
      tram      equ    0x22
      tam       equ 0x23
      bien1     equ 0x24
      bien2     equ 0x25
      bien3     equ 0x26
      dem       equ 0x27
;------------------------------
    org 0x000
    goto start
;------------------------------
;khoi tao Port va Timer0
;------------------------------
      start org 0x005
      banksel     TRISB
      clrf        TRISB
      clrf        TRISC
      movlw       b'00101000'
      movwf       OPTION_REG
      banksel     PORTB
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
      clrf        TMR0
      movlw       0x00
      movwf       tam
lb4   movlw       0x00
      movwf       tram
lb3   movlw       0x00
      movwf       chuc
lb2   movlw       0x00
      movwf       dvi
lb1   call        giaimahienthi
      movf        TMR0,0
      xorwf       tam,0
      btfsc       STATUS,Z
      goto        lb1
      movf        TMR0,0
      movwf       tam
      incf        dvi
      movf        dvi,0
      xorlw       d'10'
      btfss       STATUS,Z
      goto        lb1
      incf        chuc
      movf        chuc,0
      xorlw       d'10'
      btfss       STATUS,Z
      goto        lb2
      incf        tram
      movf        tram,0
      xorlw       d'10'
      btfss       STATUS,Z
      goto        lb3
      goto        lb4
;----------------------------------------
;chuong trinh con giai ma hien thi
;----------------------------------------
giaimahienthi
      movf       dvi,0
      call       table
      movwf      PORTB
      movlw      b'00000100'
      movwf      PORTC
      call       delay
      clrf       PORTC
      call       delay
      movf       chuc,0
      call       table
      movwf      PORTB
      movlw      b'00000010'
      movwf      PORTC
      call       delay
      clrf       PORTC
      call       delay
      movf       tram,0
      call       table
      movwf      PORTB
      movlw      b'00000001'
      movwf      PORTC
      call       delay
      clrf       PORTC
      call       delay
return
;----------------------------------------
```

```
;chuong trinh con delay
;-----------------------------------------
delay
        movlw       0xff
        movwf       dem
d1      decfsz      dem
        goto        d1
return
;-----------------------------------------
;table
;-----------------------------------------
        table   addwf       PCL,1
                DT
0x40,0x79,0x24,0x30,0x19,0x12,0x02,0x78,0x00,0x10
end
;$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$
```



LƯU ĐỒ

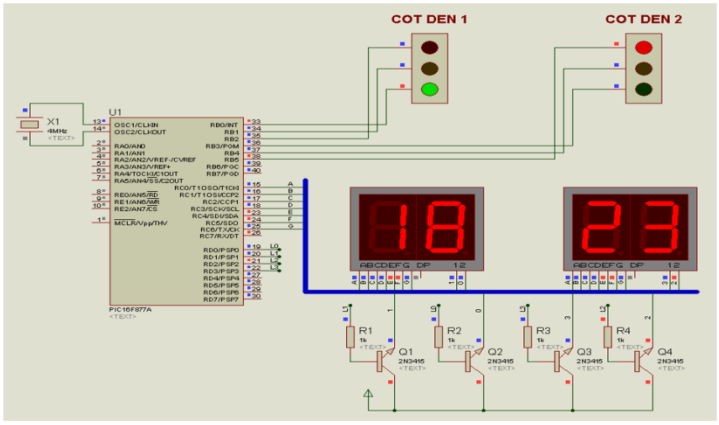


BÀI 4:Sử dụng PIC 16F887A kết nối 3led7 đoạn làm mạch đếm sản phẩm (Timer 1) – Dùng ngôn ngữ lập trình C

Chương trình:

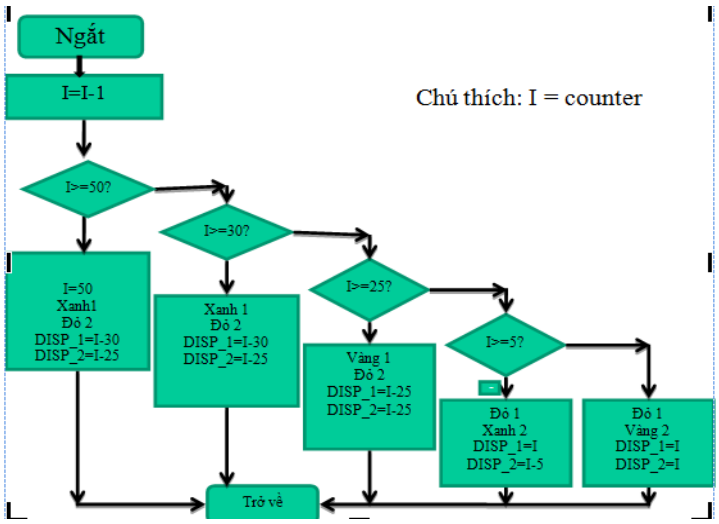
```
#include <16F877A.h>
#include <KhaiBaoTGhi_16F877A.h>
#fuses NOWDT,PUT,HS,NOPROTECT,NOLVP
#use delay(clock=20000000)
#use fast_io(b)
#use fast_io(d)
#byte dem=TMR1L
int8    tram,chuc=0,dv=0,ht=0;
const unsigned char
dig[]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x8
0,0x90};
//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
void khoitao()
{
trisb=0x0; //khoi tao portb & portd la ngo ra
trisd=0x0; //
T1CON=0X06; //chon bo chia 1:1,bo dao dong bat dong bo
dem=0;tram=0;
}
//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
void hienthi()
{
switc(ht)
{
case 0: portb=dig[dv];
portd=0x01;
delay_us(20);
portb=0xff;
break;
case 1: portb=dig[chuc];
portd=0x02;
delay_us(20);
portb=0xff;
break;
case 2: portb=dig[tram];
portd=0x04;
delay_us(20);
portb=0xff;
}
ht++;
if (ht==3) ht=0;
}

//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
void main()
{
khoitao();
bit_set(T1CON,0);
while(1)
{
if (dem==100)
{
bit_clear(T1CON,0); //xoa bit TMR1ON de thay doi gia tri TMR1L
dem=0;
tram=tram+1;
if (tram==10)
tram=0;
bit_set(T1CON,0);
}
chuc=dem/10;
dv=dem%10;
hienthi();
}
}
```

BÀI 5: Dùng PIC16F877A điều khiển đèn giao thông, thời gian đèn xanh là 20 giây, đỏ là 25 giây, vàng 5 giây, dùng ngôn ngữ assembly. Dùng timer 1 làm bộ định thời.

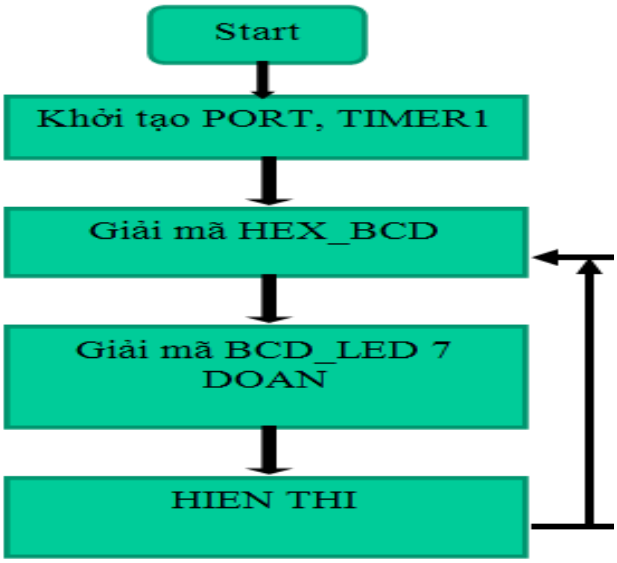


I. Lưu đồ giải thuật

a. Chương trình ngắt



b. Chương trình chính



CHƯƠNG TRÌNH

```
processor       P16f877a
include         <P16f877a.inc>
_CONFIG     _CP_OFF & _WDT_OFF & _BODEN_OFF
& _PWRTE_ON & _RC_OSC & _WRT_OFF &
_LVP_ON & _CPD_OFF
;================================
;       KHAI BAO HANG VA BIEN
;================================
;Dinh nghia cac gia tri su dung hien thi trang thai
;den giao thong tren hai cot
DENGT       EQU    PORTB
XANH_1      EQU 0
VANG_1      EQU    1
DO_1   EQU    2
XANH_2      EQU 3
VANG_2      EQU    4
DO_2   EQU    5
;Khai bao cac hang su dung dieu khien thoi gian hien thi
DATA_LED    EQU    PORTC
CONTROL_LED         EQU    PORTD
LED_DV_1    EQU    0x01
LED_CHUC_1 EQU    0x02
LED_DV_2    EQU    0x04
LED_CHUC_2 EQU 0x08
;Hang so kiem tra du 1 giay
DIV_1S              EQU    0
```

```
;-----KHAI BAO BIEN----------
cblock  0x20
        STATUS_TEMP
        W_TEMP

        TGCOT_1     ;Gia tri thoi gian dua ra 2 cot den
        TGCOT_2

        DV_COT1     ;Gia tri BCD cua so giay
        CHUC_COT1
        DV_COT2
        CHUC_COT2

        temp_1      ;Bien tam su dung trong chuong
        temp_2      ;trinh chuyen doi HEX sang BCD

        MA_CHUC_1   ;Bien luu tru ma 7 doan cua cac led
        MA_DV_1
        MA_CHUC_2
        MA_DV_2

        btam1               ;Bien tam su dung trong
        btam2               ;chuong trinh phuc vu ngat
        counter

        THANHGHI_TAM    ;Thanh ghi de kiem tra ngat
lan 2
        dem                         ;Bien dung cho
chuong trinh delay
        endc
;---------------------------------
;Tao MACRO luu va phuc hoi cac
;thanh ghi quan trong
PUSH    MACRO
        MOVWF       W_TEMP
        SWAPF       STATUS,W
        MOVWF       STATUS_TEMP
        ENDM

POP             MACRO
        SWAPF       STATUS_TEMP,W
        MOVWF       STATUS
        SWAPF       W_TEMP,F
        SWAPF       W_TEMP,W
```

```
        ENDM
;Ket thuc MACRO
;================================
;       CHUONG TRINH NGAT
;================================
NGAT org      0x04
        PUSH
        BANKSEL     PIR1
        BTFSS PIR1, TMR1IF   ;Kiem tra co ngat
        GOTO  exit_int              ;Neu khong phai
ngat do timer1 thi thoat
        BCF    PIR1, TMR1IF         ;Xoa co ngat
         BANKSEL    TMR1H    ;Khoi tao lai gia tri timer
        MOVLW       high d'3036'
        MOVWF       TMR1H
        MOVLW       low  d'3036'
        MOVWF       TMR1L

        BTFSS THANHGHI_TAM, DIV_1S
        GOTO  notfull_1s
        ;Chuong trinh phuc vu ngat thuc
        ;hien cac trang thai cua den giao thong
        ;va gia tri thoi gian de dua ra hien thi
        BCF         THANHGHI_TAM, DIV_1S
        BANKSEL     DENGT
        CLRF   DENGT

        DECF   counter, 1
        MOVLW       0x00
        XORWF       counter, W
        BTFSS STATUS, Z
        GOTO  X1_D2
;-------------------------
        MOVLW       d'50'
        MOVWF       counter
        BSF         DENGT, XANH_1
        BSF         DENGT, DO_2
        MOVLW       d'30'
        MOVWF       btam1
        MOVLW       d'25'
        MOVWF       btam2
        CALL   laythoigian
        GOTO  exit_int
;-------------------------
X1_D2
        MOVLW       d'30'
        SUBWF       counter, 0
        BTFSS STATUS, C
        GOTO  V1_D2
        BANKSEL     DENGT
        BSF         DENGT, XANH_1
        BSF         DENGT, DO_2
        MOVLW       d'30'
        MOVWF       btam1
        MOVLW       d'25'
        MOVWF       btam2
        CALL   laythoigian
        GOTO  exit_int
;-------------------------
V1_D2
        MOVLW       d'25'
        SUBWF       counter, W
        BTFSS STATUS, C
        GOTO  D1_X2
        BANKSEL     DENGT
        BSF         DENGT, VANG_1
        BSF         DENGT, DO_2
        MOVLW       d'25'
        MOVWF       btam1
        MOVLW       d'25'
        MOVWF       btam2
        CALL   laythoigian
        GOTO  exit_int
;-------------------------
D1_X2
        MOVLW       d'5'
        SUBWF       counter, W
        BTFSS STATUS, C
        GOTO  D1_V2
        BANKSEL     DENGT
        BSF         DENGT, DO_1
        BSF         DENGT, XANH_2
        MOVLW       d'0'
        MOVWF       btam1
        MOVLW       d'5'
        MOVWF       btam2
        CALL   laythoigian
        GOTO  exit_int
```

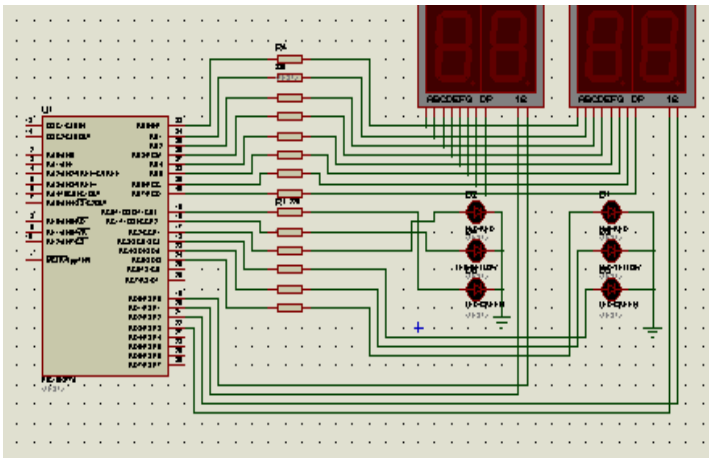
```
;-------------------------
D1_V2
        BANKSEL     DENGT
        BSF         DENGT, DO_1
        BSF         DENGT, VANG_2
        MOVLW       d'0'
        MOVWF       btam1
        MOVLW       d'0'
        MOVWF       btam2
        CALL   laythoigian

notfull_1s
        BSF     THANHGHI_TAM, DIV_1S
exit_int
        POP
        RETFIE
;=====KET THUC CHUONG TRINH NGAT=====

;---------------------------------
;       CHUONG TRINH LAY THOI GIAN
;---------------------------------
laythoigian
        MOVF  btam1, W
        SUBWF       counter, W
        MOVWF       TGCOT_1

        MOVF  btam2, W
        SUBWF       counter, W
        MOVWF       TGCOT_2
        RETURN
;================================
;       CHUONG TRINH CHINH
;================================
org             0x000
                goto    main

;CHUONG TRINH BAT DAU O DAY
Chuongtrinhchinh
org             0x090
main
        BANKSEL     TRISC
        CLRF   TRISB
        CLRF   TRISC
        CLRF   TRISD
        BCF         STATUS, RP0    ;chon bank 0
        BCF         STATUS, RP1
        MOVLW       d'50'
        MOVWF       counter
        CALL   khoitaotimer1
loop
        CALL   hex_bcd_1 ;Giai ma TGCOT_1 sang BCD
        CALL   hex_bcd_2 ;Giai ma TGCOT_2 sang BCD
        CALL   bcd_7doan
        CALL   hienthi
        GOTO   loop
;=======================================
;       KHOI TAO TIMER 1
;=======================================
khoitaotimer1
        CLRF   T1CON
        CLRF   INTCON
        CLRF   TMR1H
        CLRF   TMR1L
        BSF    STATUS, RP0          ;chon bank 1
        CLRF   PIE1
        BSF    PIE1,TMR1IE   ;cho phep ngat timer 1
        BCF    STATUS, RP0          ;chon bank 0
        CLRF   PIR1                 ;xoa cac co ngat

        MOVLW  0xC0         ;cho phep ngat toan cuc va
        MOVWF       INTCON        ;ngat ngoai vi
        MOVLW       0x30
        MOVWF       T1CON    ;0011 0000: Timer1 su
dung Fosc/4,;PSA 1:8
        BANKSEL     TMR1H
        MOVLW       high d'3036'
        MOVWF       TMR1H
        MOVLW       low  d'3036'
        MOVWF       TMR1L
        BSF         T1CON, TMR1ON
        RETURN
;====KET THUC KHOI TAO TIMER 1====
;=CHUONG TRINH CHUYEN DOI HEX SANG
BCD==
;================================
;CHUYEN DOI O COT 1
hex_bcd_1
        BCF         STATUS, RP0
```

```
        CLRF   DV_COT1
        CLRF   CHUC_COT1
        MOVF  TGCOT_1, W
        MOVWF        temp_1
BCD1
        MOVLW        .10
        SUBWF        temp_1, W
        BTFSS STATUS, C
        GOTO  BCD2
        INCF    CHUC_COT1
        MOVLW        .10
        SUBWF        temp_1, F
        ;temp_1 = temp_1 - 10
        GOTO  BCD1
BCD2
        MOVF  temp_1, W
        MOVWF        DV_COT1
        RETURN
;CHUYEN DOI O COT 2
hex_bcd_2
        BCF             STATUS, RP0
        CLRF   DV_COT2
        CLRF   CHUC_COT2
        MOVF  TGCOT_2, W
        MOVWF        temp_2
BCD3
        MOVLW        .10
        SUBWF        temp_2, W
        BTFSS STATUS, C
        GOTO  BCD4
        INCF    CHUC_COT2
        MOVLW        .10
        SUBWF        temp_2, F
        ;temp_2 = temp_2 - 10
        GOTO  BCD3
BCD4
        MOVF  temp_2, W
        MOVWF        DV_COT2
        RETURN
;=====KET THUC HEX SANG BCD==========
;====================================
;=CHUONG TRINH CHUYEN DOI BCD SANG LED 7 DOAN=
;========================================
bcd_7doan
;Lay ma 7 doan cho led hien thi hang chuc cot 1
        MOVF  CHUC_COT1, 0
        CALL   table
        MOVWF        MA_CHUC_1
;Lay ma 7 doan cho led hien thi hang don vi cot 1
        MOVF  DV_COT1, 0
        CALL   table
        MOVWF        MA_DV_1
;Lay ma 7 doan cho led hien thi hang chuc cot 2
        MOVF  CHUC_COT2, 0
        CALL   table
        MOVWF        MA_CHUC_2
;Lay ma 7 doan cho led hien thi hang don vi cot 2
        MOVF  DV_COT2, 0
        CALL   table
        MOVWF        MA_DV_2
        RETURN
;-------------------------------------
;BANG LAY MA LED 7 DOAN
table   ADDWF        PCL, F
                RETLW        0XC0   ;0
                RETLW        0XF9   ;1
                RETLW        0XA4   ;2
                RETLW        0XB0   ;3
                RETLW        0X99   ;4
                RETLW        0X92   ;5
                RETLW        0X82   ;6
                RETLW        0XF8   ;7
                RETLW        0X80   ;8
                RETLW        0X90   ;9
;-------------------------------------
;==KET THUC BCD SANG 7 DOAN===

;====================================
;CHUONG TRINH QUET LED 7 DOAN HIEN THI==
;========================================
hienthi
        MOVF  MA_DV_1, 0
        MOVWF        DATA_LED
        MOVLW        LED_DV_1
        MOVWF        CONTROL_LED
        CALL   delay
        CLRF   CONTROL_LED
```

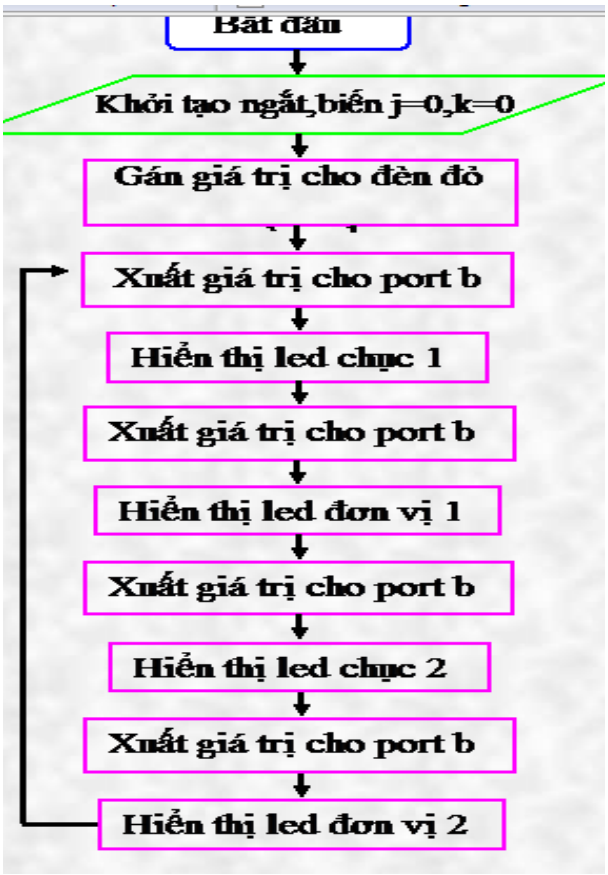
```
        CALL   delay

        MOVF  MA_CHUC_1, 0
        MOVWF       DATA_LED
        MOVLW       LED_CHUC_1
        MOVWF       CONTROL_LED
        CALL   delay
        CLRF   CONTROL_LED
        CALL   delay

        MOVF  MA_DV_2, 0
        MOVWF       DATA_LED
        MOVLW       LED_DV_2
        MOVWF       CONTROL_LED
        CALL   delay
        CLRF   CONTROL_LED
        CALL   delay

        MOVF  MA_CHUC_2, 0
        MOVWF       DATA_LED
        MOVLW       LED_CHUC_2
        MOVWF       CONTROL_LED
        CALL   delay
        CLRF   CONTROL_LED
        CALL   delay
        RETURN
;===KET THUC HIEN THI=======
;--------------------------------
;       CHUONG TRINH DELAY
;--------------------------------
delay
        MOVLW       0xFF
        MOVWF       dem
del     DECFSZ      dem
        GOTO  del
        RETURN

        END
;=====KET THUC CHUONG TRINH=====
```

BÀI 6: Thiết kế đèn giao thông dùng PIC16F877A điều khiển 4 Led 7 đoạn, 6 Led đơn. Thời gian đèn đỏ 25 giây, đèn xanh 20 giây, đèn vàng 5 giây,Timer 1 và dùng ngôn ngữ C

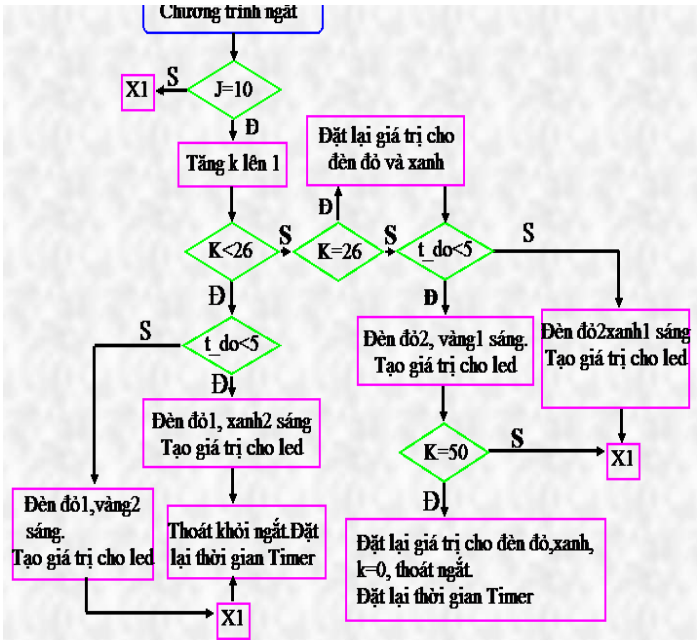
THEO PHƯƠNG PHAP QUET LED



CHƯƠNG TRÌNH CHÍNH



CHƯƠNG TRÌNH NGẮT:

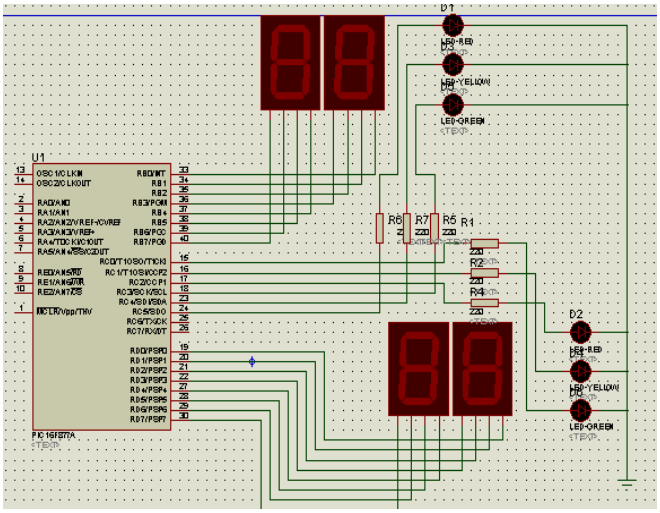


CHƯƠNG TRÌNH:

```
#include <16f877a.h>
#include <KhaiBaoTGhi_16F877A.h>
#fuses  nowdt, put, hs, noprotect, nolvp
#use delay (clock=20000000)
#use   fast_io(b)
#use   fast_io(d)
int8  j,k,chuc1,chuc2,dvi1,dvi2;
int16 t_do,a,t_xanh,b;
const unsigned char
dig[]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x9
0};
#INT_TIMER1
VOID NGAT()      // bat dau chuong trinh ngat
 {  J++;
   IF(J==10)         // khi j=10 thi thoi gian dung 1s
    { k=k+1;   // bien k dung de xac dinh thoi gian cho cot 1
      if((k<26)&&(k>0))
        { a=t_do;
          t_do=t_do-1;
          b=t_xanh;
          t_xanh=t_xanh-1;
          if(a<5)      // thoi gian hien thi cho led vang
            { portc=0x22;    // Den v2 va d1 sang
              b=a;
            }
          else
           portc=0x21;      // den x2 va den d1 sang
        }
       if((k<26)&&(k>24))     // khi du 25s thi chuyen sang
cot 2
         { t_do=24;
           t_xanh=19;
         }
       if(k>25)              // bat dau hien thi cho cot 2
         { b=t_do;
           t_do=t_do-1;
           a=t_xanh;
           t_xanh=t_xanh-1;
           if (b<5)          // khi den xanh du 20s thi den vang
bat dau sang
             { portc=0x14;    // den do 2 va den vang 1 sang
               a=b;
             }
           else
           portc=0x0c;       // den do 2 va den xanh 1 sang
         }
           chuc1=a/10;
           dvi1=a%10;
           chuc2=b/10;
           dvi2=b%10;

       if(k>49)          // khi cot 2 xong 25s thi chuyen ve cot 1
         { k=0;            // khoi tao lai gia tri dem ban dau
           t_do=24;
           t_xanh=19;
         }
        SET_TIMER1(3035);     // cai lai thoi gian cho timer1
        j=0;
    }
    SET_TIMER1(3035);      // cai lai thoi gian cho timer1
  }

void  main()               // bat dau chuong trinh chinh
  {  set_tris_b(0x00);      // khoi tao cho port b,c,d la ngo ra
     set_tris_d(0x00);
     set_tris_c(0x00);
     j=0;                  // khoi tao cho bien dem timer1
```

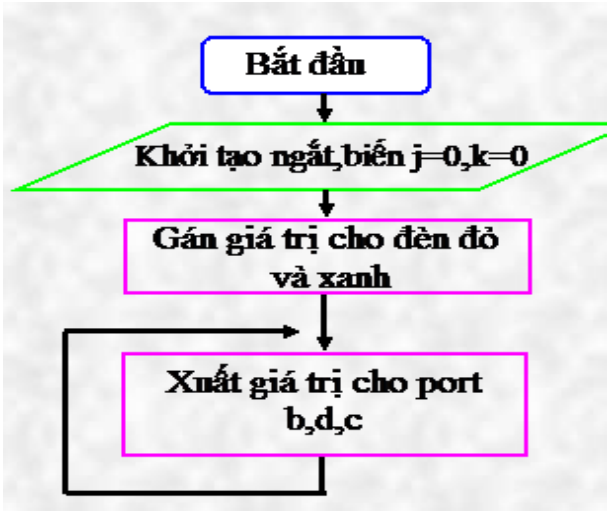
```
k=0;                  // khoi tao cho bien dem cua tung cot
t_do=24;          // gan gia tri ban dau cho den do va xanh
t_xanh=19;
SET_TIMER1(3035);      //dat so dem cho timer1
T1CON=0x35;           //khoi tao gia tri cho timer
enable_interrupts(INT_TIMER1);      //tran thi ngat
enable_interrupts(global);           //cho phep ngat timer1
WHILE(TRUE)
  { portb=dig[chuc1];         //dua gia tri chuc1 ra portb
    portd=0xf8;               //chi led o vi tri chuc1 la sang
    delay_ms(8);             //dung de nhin thay duoc led sang
    portd=0xf0;               //chong lem

    portb=dig[dvi1];          //dua gia tri dvi1 ra portb
    portd=0xf4;               //chi led o vi tri dvi1 la sang
    delay_ms(8);
    portd=0xf0;

    portb=dig[chuc2];         //dua gia tri chuc2 ra portb
    portd=0xf2;               //chi led o vi tri chuc2 la sang
    delay_ms(8);
    portd=0xf0;

    portb=dig[dvi2];          //dua gia tri dvi2 ra portb
    portd=0xf1;               //chi led o vi tri dvi2 la sang
    delay_ms(8);
    portd=0xf0;
   }
}
```

BÀI 6: THEO PHƯƠNG PHÁP GIẢI MÃ

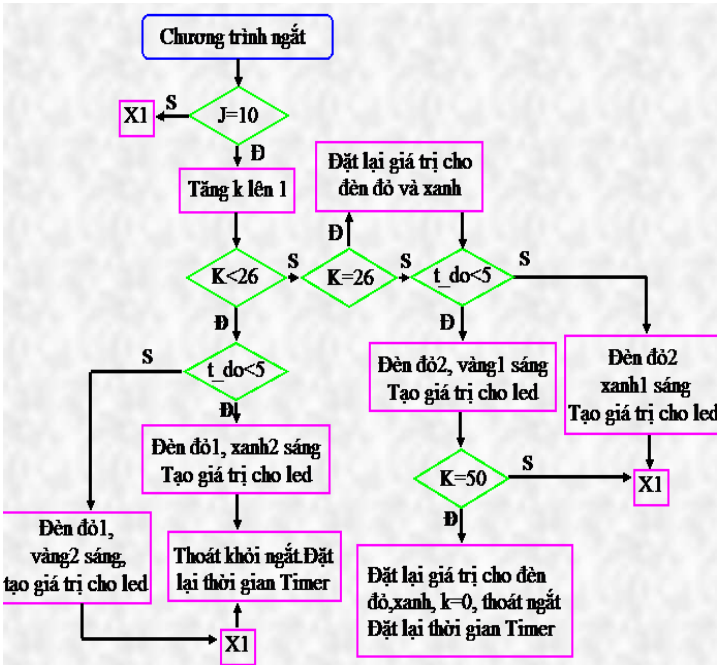


LƯU ĐỒ:

CHƯƠNG TRÌNH CHÍNH



CHƯƠNG TRÌNH NGẮT

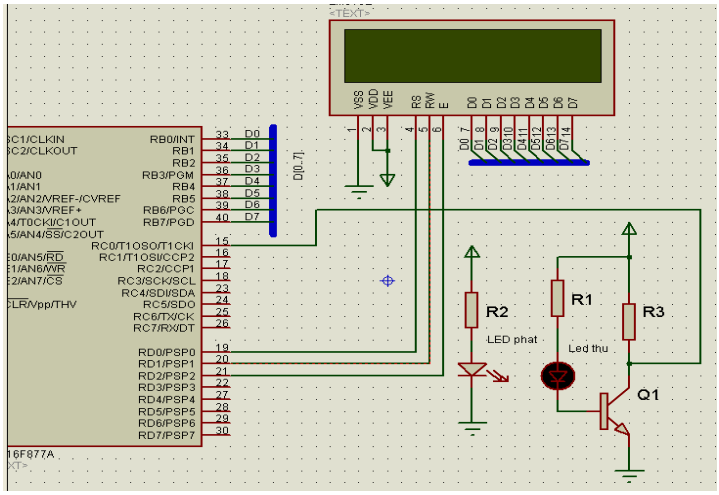


CHƯƠNG TRÌNH:

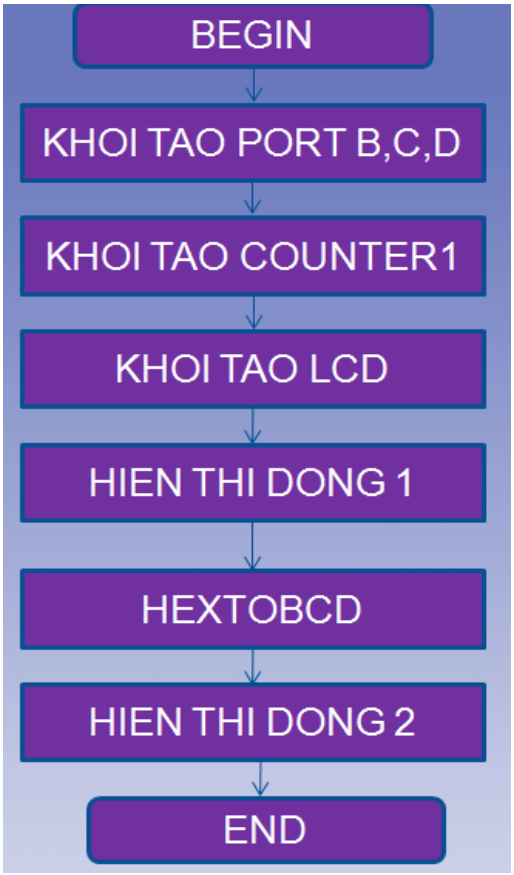
```
#include <16f877a.h>
#include <KhaiBaoTGhi_16F877A.h>
#fuses  nowdt, put, hs, noprotect, nolvp
#use delay (clock=20000000)
#use   fast_io(b)
#use   fast_io(d)
int8  j,k,chuc1,xuat1,chuc2,dvi1,dvi2,xuat2;
int16 t_do,a,t_xanh,b;
#INT_TIMER1
VOID NGAT()                  // bat dau chuong trinh ngat
 {  J++;
   IF(J==10)                 // khi j=10 thi thoi gian dung 1s
    { k=k+1;    // bien k dung de xac dinh thoi gian cho cot 1
      if((k<26)&&(k>0))
        { a=t_do;
          t_do=t_do-1;
          b=t_xanh;
          t_xanh=t_xanh-1;
          if(a<5)           // thoi gian hien thi cho led vang
             { portc=0x22;    // den vang 2 va do 1 sang
               b=a;
             }
         else
          portc=0x21;       // den xanh 2 va den do 1 sang
        }
      if((k<26)&&(k>24)) // khi du 25s thi chuyen sang cot 2
        { t_do=24;
          t_xanh=19;
        }
      if(k>25)              // bat dau hien thi cho cot 2
        { b=t_do;
          t_do=t_do-1;
          a=t_xanh;
          t_xanh=t_xanh-1;
          if (b<5)       // khi den xdu 20s thi den v bat dau sang
            { portc=0x14;    // den do 2 va den vang 1 sang
              a=b;
            }
          else
          portc=0x0c;       // den do 2 va den xanh 1 sang
        }
          chuc1=a/10;
          dvi1=a%10;
          chuc2=b/10;
          dvi2=b%10;
          chuc1=chuc1<<4;
          xuat1=chuc1|dvi1;
          chuc2=chuc2<<4;
          xuat2=chuc2|dvi2;

      if(k>49)          // khi cot 2 xong 25s thi chuyen ve cot 1
        { k=0;            // khoi tao lai gia tri dem ban dau
          t_do=24;
          t_xanh=19;
        }
      SET_TIMER1(3035);     // cai lai thoi gian cho timer1
       j=0;
    }
    SET_TIMER1(3035); // cai lai thoi gian cho timer1
 }

void  main()                 // bat dau chuong trinh chinh
  {  set_tris_b(0x00);//khoi tao cho port b,c,d la ngo ra
     set_tris_d(0x00);
     set_tris_c(0x00);
     j=0;                    // khoi tao cho bien dem timer1
     k=0;                 // khoi tao cho bien dem cua tung cot
     t_do=24;   // gan gia tri ban dau cho den d va x
```

```
    t_xanh=19;
    SET_TIMER1(3035);      //dat so dem cho timer1
    T1CON=0x35;            //khoi tao gia tri cho timer
    enable_interrupts(INT_TIMER1);      //tran thi ngat
    enable_interrupts(global);   //cho phep ngat timer1
    WHILE(TRUE)
      { portb=xuat1;                  //dua gia tri ra port b,d
        portd=xuat2;                  //de hien thi
      }
}
```

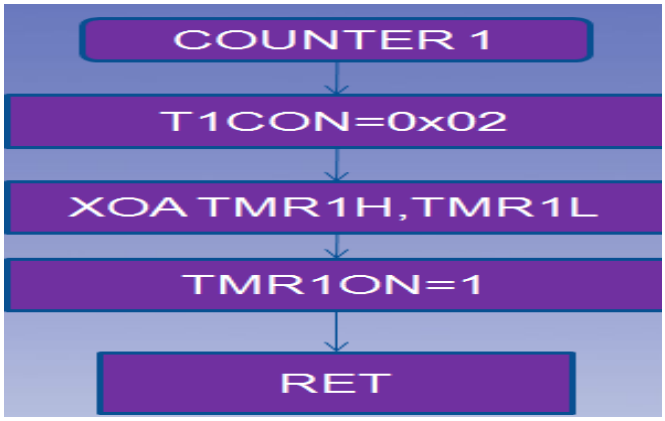
BÀI 7: MẠCH ĐẾM SẢN PHẨM HIỂN THỊ TRÊN LCD.DÙNG NGÔN NGỮ ASSEMBLY



LƯU ĐỒ CHÍNH



KHỞI TẠO COUNTER

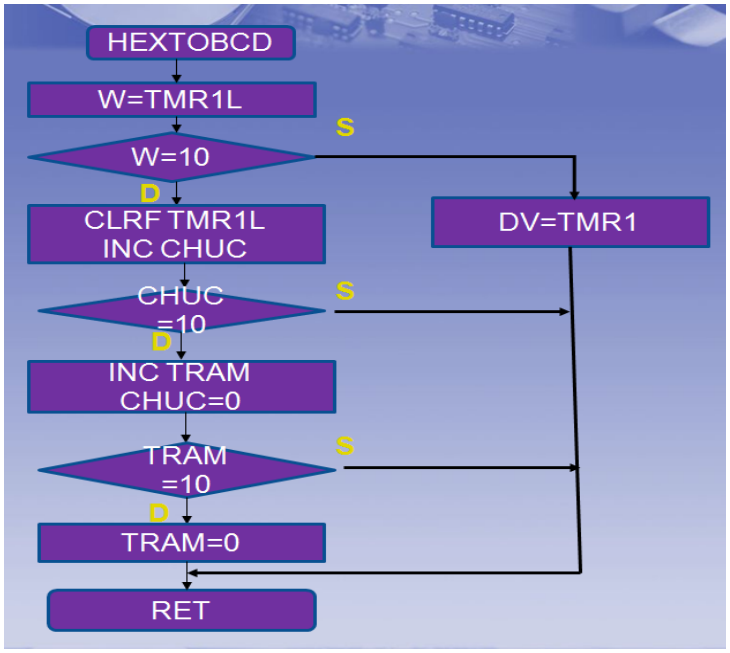


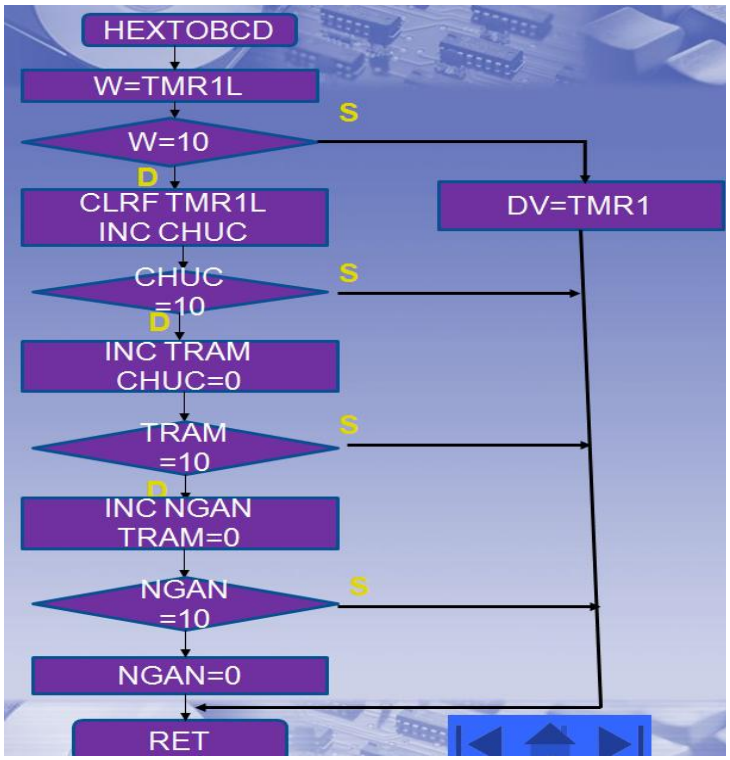
KHỞI TẠO LCD

HIỂN THỊ DÒNG 1:

HIỂN THỊ DÒNG 2:

GIẢI MÃ HEX => BCD:





CHƯƠNG TRÌNH:

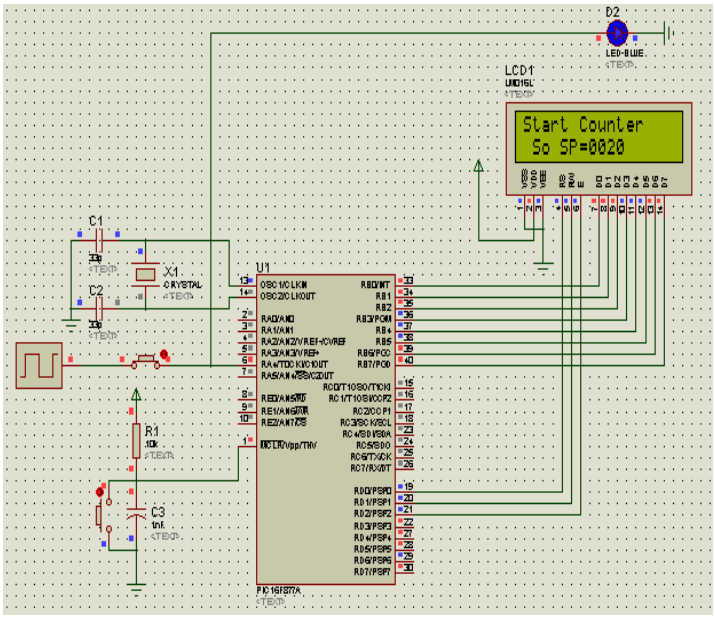
```
title   "DIEM SAN PHAM HIEN THI LCD"
processor                p16f877a
include          <p16f877a.inc>
__CONFIG
          _CP_OFF&_WDT_OFF&_LVP_OFF&_PWRTE_
ON&_HS_OSC&_BODEN_OFF&_CPD_OFF
;*************************************
;KHOI TAO CAC BIEN
;*************************************
COUNT1              EQU          0x20
COUNT2              EQU          0x21
COUNT3              EQU          0x22
COUNT               EQU          0x23
A              EQU          0x24
TAM            EQU          0x25
TR             EQU          0x26
CH             EQU          0x27
DV             EQU          0x28
NG             EQU          0x29
;*********************************
                ORG           0x000
                CLRF          STATUS
                MOVLW         0x00
                MOVWF         PCLATH
                GOTO          START
;*************************************
;KHOI TAO PORTD,PORTB
;*************************************
                ORG 0x005
START           BCF           STATUS,RP1
                BSF           STATUS,RP0
                CLRF          TRISB
                CLRF        TRISD


;*********************************
;VONG LAP CHUONG TRINH CHINH
;*********************************
MAIN
                CALL          KTCOUNTER1
                CALL          KHOITAOLCD
                MOVLW               D'0'
                MOVWF               NG
```

```
                MOVLW       D'0'
                MOVWF       TR
                MOVLW       D'0'
                MOVWF       CH
                MOVLW       D'0'
                MOVWF       DV
                CALL        DONG1

LOOP            CALL        HEXTOBCD
                CALL        DONG2
                GOTO        LOOP
;************************************************
KTCOUNTER1
                BCF         STATUS,RP1
                BCF         STATUS,RP0
        ;CHON BANK 0
                MOVLW       B'00000010'
        ;KHOI TAO COUNTER
                MOVWF       T1CON
                CLRF        TMR1H
        ;XOA 8 BIT CAO
                CLRF        TMR1L
        ;XOA 8 BIT THAP
                BSF         T1CON,TMR1ON
        ;cho couter bt dau dem khi co xung
RETURN
;************************************************
KHOITAOLCD
                MOVLW       0X38
        ;Khoi tao hang 2 matrix 5x8.
                MOVWF       A
                CALL        GHIMADK
                CALL        DELAY40mS

                MOVLW       0X0C
        ;Bat man hinh LCD.
                MOVWF       A
                CALL        GHIMADK
                CALL        DELAY40mS

                MOVLW       0X01
        ;Xoa man hinh va dua con tro ve dau dong thu 1.
                MOVWF       A
                CALL        GHIMADK
                CALL        DELAY40mS
RETURN
;************************************************
HEXTOBCD

                MOVF        TMR1L,0
; copy noi dung TMR1L  vao w
                MOVWF       TAM
;copy noi dung w vao TAM
                MOVLW       D'10'
;w=10
                XORWF       TAM,0
                BTFSS       STATUS,Z
;nhay neu z=1<=> kq=0
                GOTO        X1

                CLRF        TMR1L
                INCF        CH
                MOVLW       D'10'
                XORWF       CH,0
                BTFSS       STATUS,Z
;nhay neu z=1<=> kp=0  (CH=10)
                GOTO        X2

                INCF        TR
                CLRF        CH
                MOVLW       D'10'
                XORWF       TR,0
                BTFSS       STATUS,Z
                GOTO        X1

                INCF        NG
                CLRF        TR
                MOVLW       D'10'
                XORWF       NG,0
                BTFSS       STATUS,Z
                GOTO        X2
                CLRF        NG
                GOTO        X2
X1              MOVF        TMR1L,0
                MOVWF       DV

X2              RETURN
```

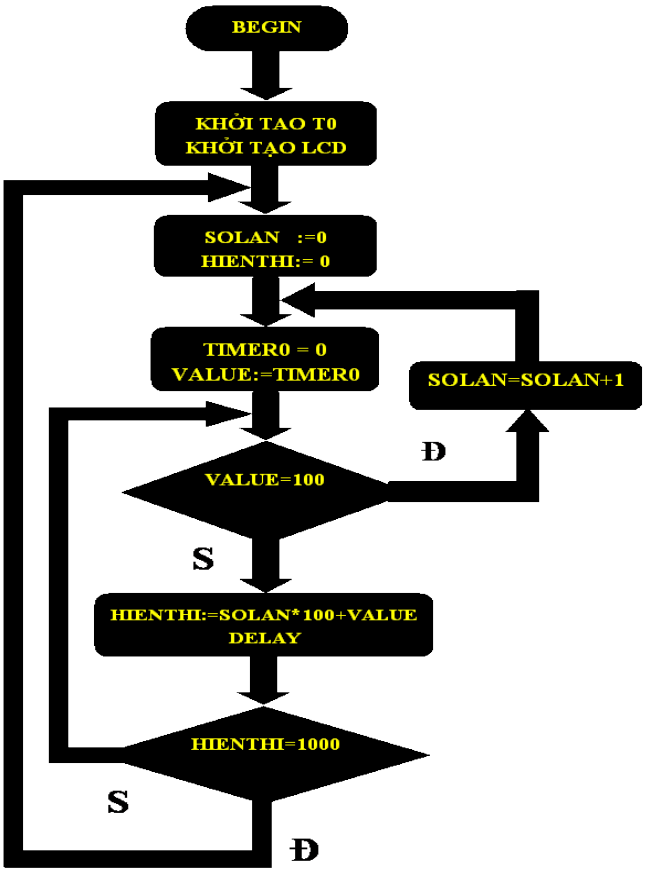
```
;************************************************
GHIMADK     MOVF    A,0
;dk LCD hoat dong
            MOVWF   PORTB
            BCF     PORTD,0
            BCF     PORTD,1
            BSF     PORTD,2
            BCF     PORTD,2
RETURN
;************************************************
DONG1
            MOVLW       0X01
    ;Xoa man hinh va dua con tro ve dau dong thu 1.
            MOVWF       A
            CALL        GHIMADK
            CALL        DELAY

            MOVLW   0X80
    ;hien thi dong 1" SO san pham la"
            MOVWF   A
            CALL    GHIMADK
            CALL    DELAY

            CLRF    COUNT
DONG11      MOVF    COUNT,0
            CALL    TABLE
            MOVWF       A
            CALL    GHIDATA
            CALL    DELAY
            INCF    COUNT,0
            XORLW       D'16'
            BTFSS   STATUS,Z
            GOTO    LB1
            RETURN
LB1         INCF    COUNT,1
            GOTO    DONG11
;************************************************
DONG2
            MOVLW       0XC5
    ;DUA CON TRO HANG 5 CUA DONG 2
            MOVWF       A
            CALL    GHIMADK
            CALL    DELAY
            MOVF    NG,0
            MOVWF       A
            MOVLW   B'00110000'     ;giai ma ra
LCD
            ADDWF       A,1
            CALL    GHIDATA
            CALL    DELAY

            MOVF    TR,0
            MOVWF       A
            MOVLW   B'00110000'     ;giai ma ra
LCD
            ADDWF       A,1
            CALL    GHIDATA
            CALL    DELAY

            MOVF    CH,0
            MOVWF       A
            MOVLW   B'00110000'     ;giai ma ra
LCD
            ADDWF       A,1
            CALL    GHIDATA
            CALL    DELAY

            MOVF    DV,0
            MOVWF       A
            MOVLW   B'00110000'
            ADDWF       A,1
            CALL    GHIDATA
            CALL    DELAY

RETURN
;************************************************
GHIDATA
            MOVF    A,0
;chuong trinh hien thi LCD
            MOVWF       PORTB
            BSF         PORTD,0
;RS
            BCF         PORTD,1
;RW
            BSF         PORTD,2
;E
```

```
                BCF             PORTD,2
                RETURN

;*************************************
DELAY40mS       MOVLW           D'255'
        ;delay 40ms(tan so 20MHz)
                MOVWF           COUNT1
DEL1            MOVLW           0xFF
                MOVWF           COUNT2
DEL2            DECFSZ          COUNT2
                GOTO            DEL2
                DECFSZ          COUNT1
                GOTO            DEL1
                RETURN

;************************************

DELAY           MOVLW           D'255'
                MOVWF           COUNT3
LB              DECFSZ          COUNT3
                GOTO            LB
                RETURN
;*********************************************
TABLE           ADDWF           PCL,1
                DT      " SO SAN PHAM LA:"
                END
;*****************************************
;KET THUC CHUONG TRINH
;*****************************************
```

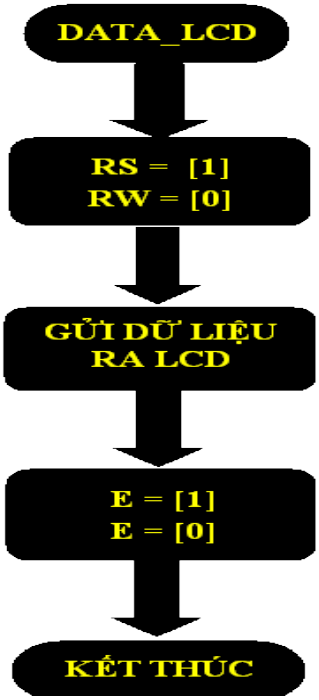
BÀI 8: MẠCH ĐẾM SẢN PHẨM HIỂN THỊ TRÊN LCD.DÙNG NGÔN NGỮ C



LƯU ĐỒ



LƯU ĐỒ XUẤT LCD:



ghi du lieu

LƯU ĐỒ XUẤT MÃ LỆNH RA LCD:

Chương trình đếm sản phẩm dùng PIC 16F877A

```
//==============================
// Title : Dem san pham dung Timer 0 hien thi LCD 16x2
// Writer : Nhom 8 _ 061170
// Hardware : PIC 16F877A
// Complier : CCS C
//==============================

#include <16F877A.h>
#include <def_16f877a.h>
#use delay( clock=20000000 )

//------------Define signals-----------------------

#define RS RD0    //Ket noi chan RS cua LCD voi chan RD0 cua PIC
#define RW RD1    //Ket noi chan RW cua LCD voi chan RD1 cua PIC
#define E RD2     //Ket noi chan E cua LCD voi chan RD2 cua PIC
#define LCD PORTB //Ket noi LCD voi chan RB0-RB7 cua PIC

void cmd_lcd( int8 command ) ;
void data_lcd( int8 data ) ;
Void convert(int16 value) ;

int8 value ;
int8 donvi,chuc,tram,ngan ;        int 8, int 16

int16  solan,hienthi ;

//==================================
//------------ Sub-Program ----------------------------
//==================================

   //Chuong trinh con gui command toi LCD
void cmd_lcd( int8 command )
{
RS = 0 ;    //Bus se ket noi voi thanh ghi lenh IR de nhan lenh
RW = 0 ;  //LCD o che do ghi   ghi du lieu len lcd
LCD = command ;

E = 1 ;     //Tao ra xung high-to-low o che do ghi
E = 0 ;

delay_ms( 10 ) ;
}

  //Chuong trinh con gui du lieu hien thi len LCD
void data_lcd( int8 data )  ghi dl vao thanh ghi DR để hiển thị lên LCD
{
RS = 1 ;    //Bus se ket noi voi thanh ghi du lieu DR
RW = 0 ;   //LCD o che do ghi
```

```
LCD = data ;

E = 1 ;       //Tao ra xung high-to-low o che do ghi
E = 0 ;

delay_ms( 10 ) ;
}
```

//Chuong trinh con chuyen doi gia tri thanh ky tu hien thi LCD

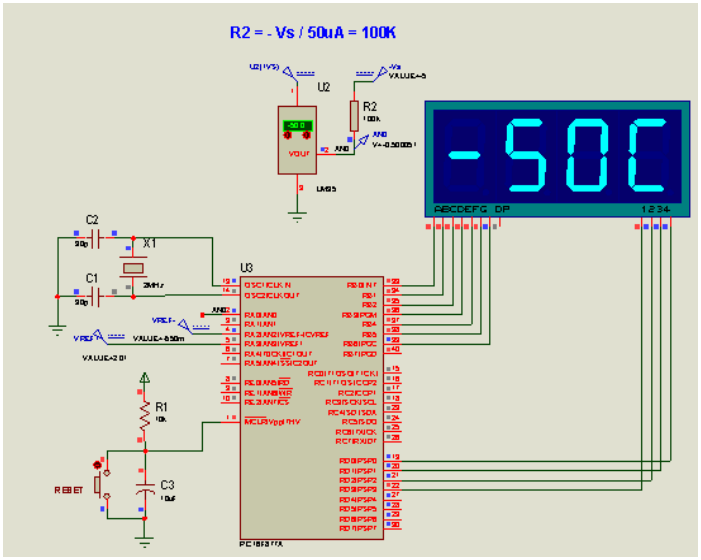
```
Void convert(int16 hienthi)
{

Int16 temp ;
temp = hienthi ;    // Gan bien temp cho gia tri trong thanh ghi TMR0
 ngan = temp / 1000 + 0x30 ;

temp = temp % 1000 ;

tram = temp / 100 + 0x30 ;
temp = temp % 100 ;

chuc = temp / 10 + 0x30 ;
donvi = temp % 10 + 0x30 ;
}

//====================================
//----------------------Main program-------------------------
//====================================

Void main()
{
//====================================
// Khoi tao Timer0
//====================================

Set_timer0(0) ;
//setup_timer_0(RTCC_EXT_L_TO_H) ;
//Chon bit canh lên trên chân RA4
T0CS = 1 ;      //Cho phep nhan xung o chan T0CKI
T0SE = 0 ;      //Cho phep xung vao chan T0CKI tich cuc canh len

TMR0IE = 0 ;  //Khong cho phep ngat T0
delay_ms(10) ;

//========================
// Khoi tao LCD
//========================

TRISB = 0 ;        PORT B,D LA XUAT
TRISD = 0 ;

delay_ms( 15 ) ;       //Cho cho den khi nguon cua LCD dat 4.5v
cmd_lcd( 0x0C ) ;
```

//LCD hien thi, khong hien thi con tro, con tro khong nhap nhay.

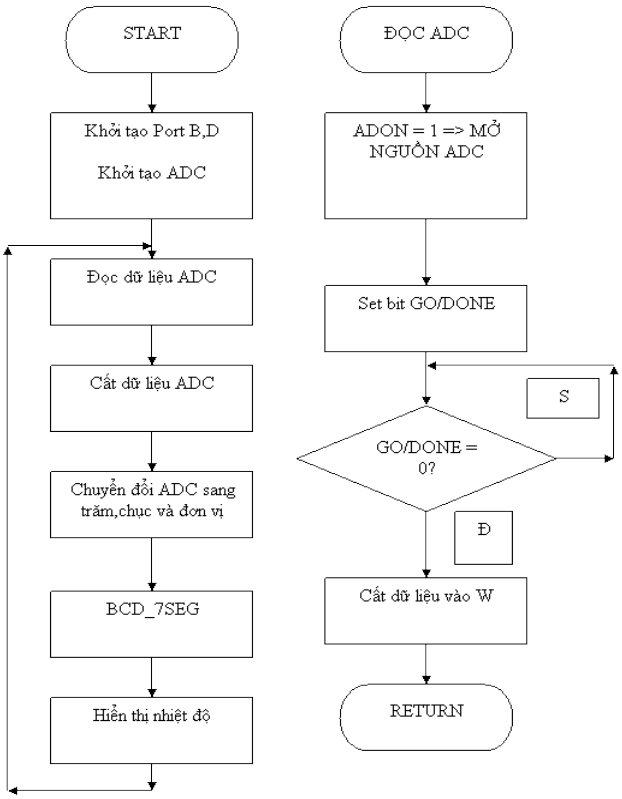
```
cmd_lcd( 0x38 ) ;       //LCD su dung Bus 8bit,hien thi 2 hang  function
cmd_lcd( 0x01 ) ;       //Xoa man hinh LCD  clear display
cmd_lcd( 0x06 ) ;       // Noi dung khong dich chuyen  entry mode set

//########################################
cmd_lcd( 0x01 ) ;  clear display: dua con tro ve goc trai
                   dau hang 1
cmd_lcd( 0x80 ) ;       //Con tro o dau dong 1
printf(data_lcd,"Group 8 - Hello!");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;     con tro ve dau dong
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Member:");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Xuan Len");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Huynh Nhu");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Thanh Thuy");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Minh Quang");

delay_ms( 1000 ) ;
cmd_lcd( 0x01 ) ;
```

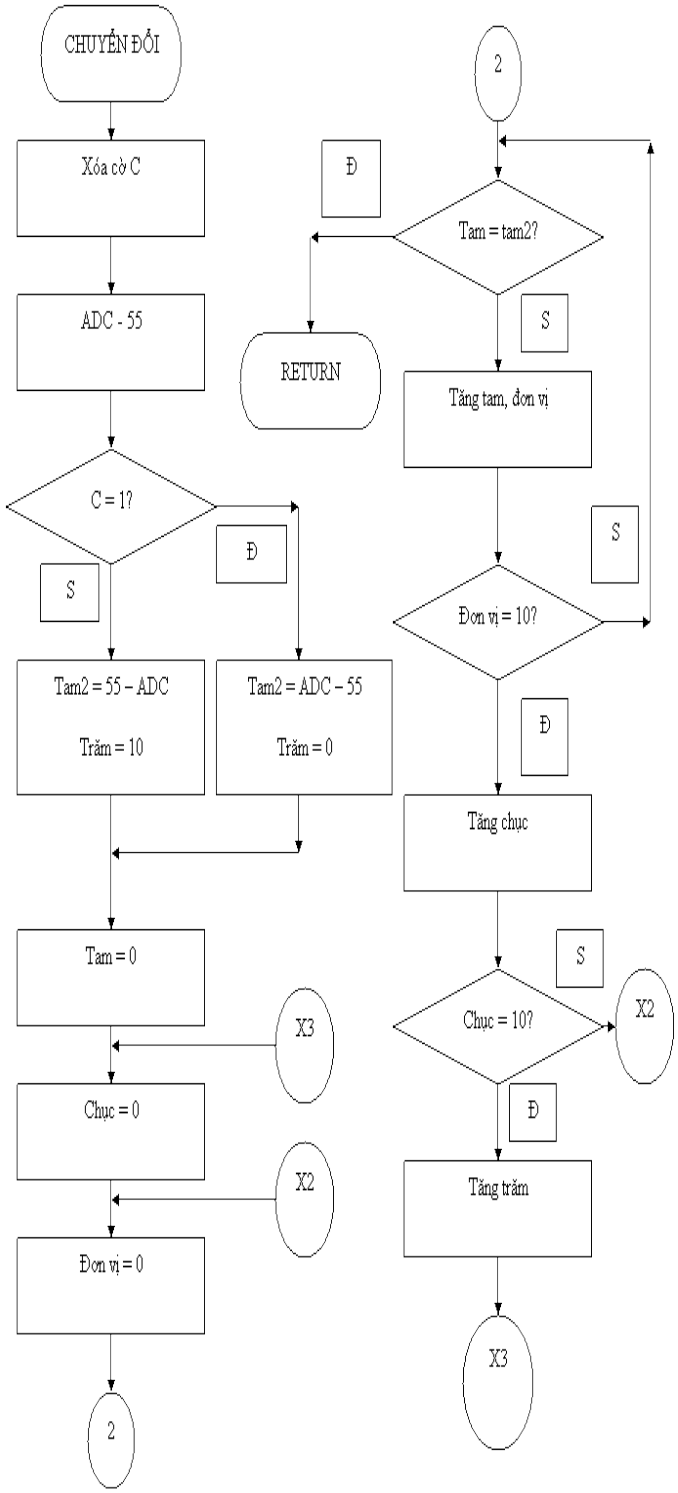
```
cmd_lcd( 0x80 ) ;
printf(data_lcd,"The Bao");

delay_ms( 1000 ) ;

cmd_lcd( 0x01 ) ;
cmd_lcd( 0x80 ) ;
printf(data_lcd,"Start Counter");

delay_ms( 500 ) ;

cmd_lcd( 0xC0 ) ;          //Contro o dau dong 2
printf(data_lcd," So SP= ") ;

Set_timer0(0) ;            //Gia tri cua TMR0 = 0

While (1)
{
cmd_lcd( 0xC0 + 0x07 ) ;   //Cho con tro nhay toi vi tri can hien thi

value = get_timer0() ;     //Dua gia tri trong TMR0 vao bien Value

if(value == 100)
{ solan++                  ;
  set_timer0(0)            ;
  value=0                  ;
}

hienthi=solan*100+value;

if(hienthi==1000)

{   solan=0                ;
    Set_timer0(0)          ;

    hienthi=0              ;
}

     convert(hienthi)      ;

      data_lcd(ngan)       ;

      data_lcd(tram)       ;

      data_lcd(chuc)       ;

      data_lcd(donvi)      ;

}
}
```

BÀI 9: ĐO NHIỆT ĐỘ DÙNG CẢM BiẾN LM35 VỚI PIC 16F877A DÙNG NGÔN NGỮ ASSEMBLY



LƯU ĐỒ





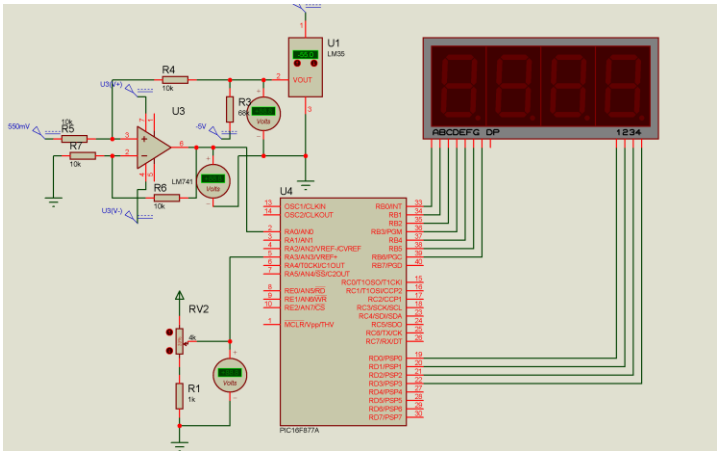
CHƯƠNG TRÌNH

```
title "chuong trinh hien thi nhiet do"
  processor p16f877a
  include <p16f877a.inc>
  __CONFIG
_CP_OFF&_PWRTE_ON&_WDT_OFF&_HS_OSC&_LV
P_OFF
  ;++++++++++++++++++++++++++++++++++++++++
  ; cac bien duoc su dung
 ;++++++++++++++++++++++++++++++++++++++++
  cblock 0x020
  count1
  count2
  count3
  tram
  chuc
  dvi
  tam
  tam2
  bien1
  bien2
  bien3
  so55
  dem
  kqADC
  endc
 ;---------- CT ----------
          org 0x005
          goto start
 ;--------- khoi tao port b,d ---------
 start
          bcf STATUS,RP1
          bsf STATUS,RP0 ; chon bank 1
          clrf TRISB
          clrf TRISD ; lam ngo xuat du lieu

 ;---------- khoi tao ngo vao ADC ----------
 ADC
          movlw 0x0F
          movwf ADCON1 ;chon AN2=Vref- va AN3=Vref+
          bcf STATUS,RP0 ; tro ve bank 0
 ;---------- MAIN ---------
 main
          movlw 0x01
          call docADC
           movwf kqADC
          call chuyen
```

```
        call bcd_7seg
        call delhthi
        goto main
 ;-------- doc ADC dung bit GO/DONE --------
        docADC
        movwf ADCON0
        movlw 0x14
        movwf dem
        delay12
        decfsz dem,1
        goto delay12
        bsf ADCON0,2
        gone btfsc ADCON0,2
        goto gone
         movf ADRESL,0
        return
 ;--------- chuyen doi ---------
 chuyen   bcf STATUS,C
        movlw 0x37
         subwf kqADC,0
        Btfss STATUS,C ;neu phep tru kq duong C=1 va nguoc lai
        goto x4                 ;ADC < 55 nhay den x4 de tao dau "-"
         movwf  tam2
        Movlw  0x00
        Movwf  tam
        Movlw  0x00
        Movwf  tram
 goto X3
x4
        Movlw           0x00
        Movwf            tam
        movlw           0x37
         movwf          so55
        movf            kqADC,0
        subwf           so55,0
        movwf            tam2
        movlw           0x0a
        movwf           tram
X3
        Movlw           0x00
        Movwf           chuc
X2       Movlw          0x00
        Movwf           dvi
X1
        Movf            tam,0
        Xorwf           tam2,0
        Btfss           STATUS,Z
        Goto            X5
        Goto            X6
X5
        Incf tam,1
        Incf dvi,1
        Movf dvi,0
        Xorlw 0x0A
        Btfss STATUS,Z
 Goto X1

        Incf chuc,1
        Movf chuc,0
         Xorlw 0x0A
        Btfss STATUS,Z
 Goto X2

        Incf  tram,1
        Movf tram,0
        Xorlw 0x0A
        Btfss  STATUS,Z
 Goto X3
X6      return
 ;--------- bcd_7seg -----------
 bcd_7seg
        movf dvi,0
        call table
         movwf bien1
        movf chuc,0
        call table
        movwf bien2
        movf tram,0
        call table
         movwf bien3
 ;---------- del_hthi -----------
 delhthi
        movlw 0x01
        movwf count1
del1

          movlw 0x10
        movwf count2
 del2

        call hthi
          decfsz count2,1
          goto del2
          ecfsz count1,1
          goto del1
          return

 ;---------- hien thi ----------
 hthi
```

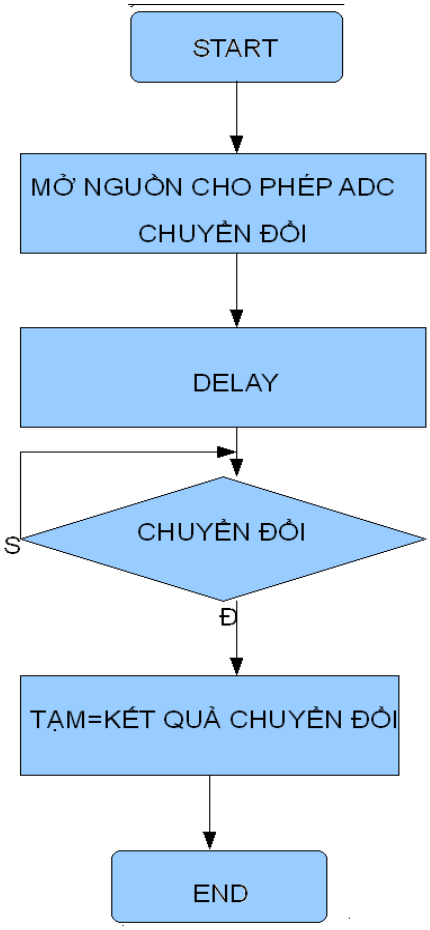
```
   Movlw 0xC6
   Movwf PORTB
          Movlw 0x01
          Movwf PORTD
          Call delay
           Movlw 0xff
          Movwf PORTB

          movf bien1,0
           movwf PORTB
           movlw 0x02
          movwf PORTD
          call delay
          movlw 0xff
          movwf PORTB

           movf bien2,0
           movwf PORTB
          movlw 0x04
          movwf PORTD
          call delay
          movlw 0xff
          movwf PORTB

          movf bien3,0
           movwf PORTB
           movlw 0x08
           movwf PORTD
          call delay
          movlw 0xff
           movwf PORTB
           return
;--------- delay ---------
   delay
           movlw 0x20
           movwf count3
           del3

           decfsz count3
          goto del3
          return
   ;--------- table ----------
   table
   addwf PCL,1
   DT
0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90,0xbf
   end
```

BÀI 10: DÙNG PIC16F877A GIAO TIẾP VỚI LED 7 ĐOẠN VÀ CẢM BIẾN NHIỆT ĐỘ LM35 NỐI NGÕ VÀO KÊNH THỨ 0.LẬP TRÌNH ĐO NHIỆT ĐỘ DÙNG NGÔN NGỮ C.

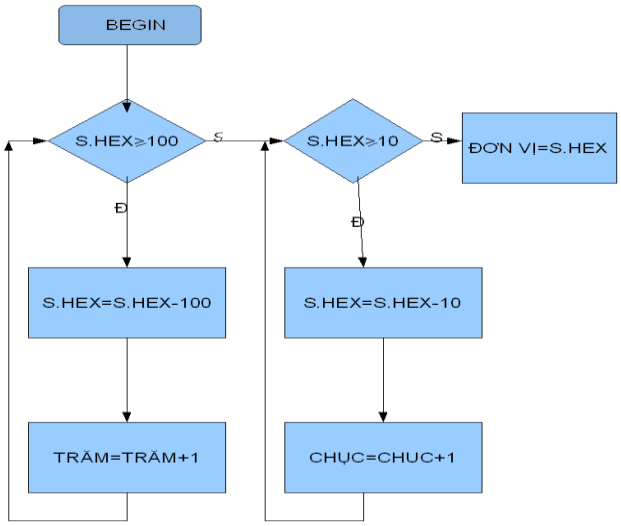


LƯU ĐỒ:

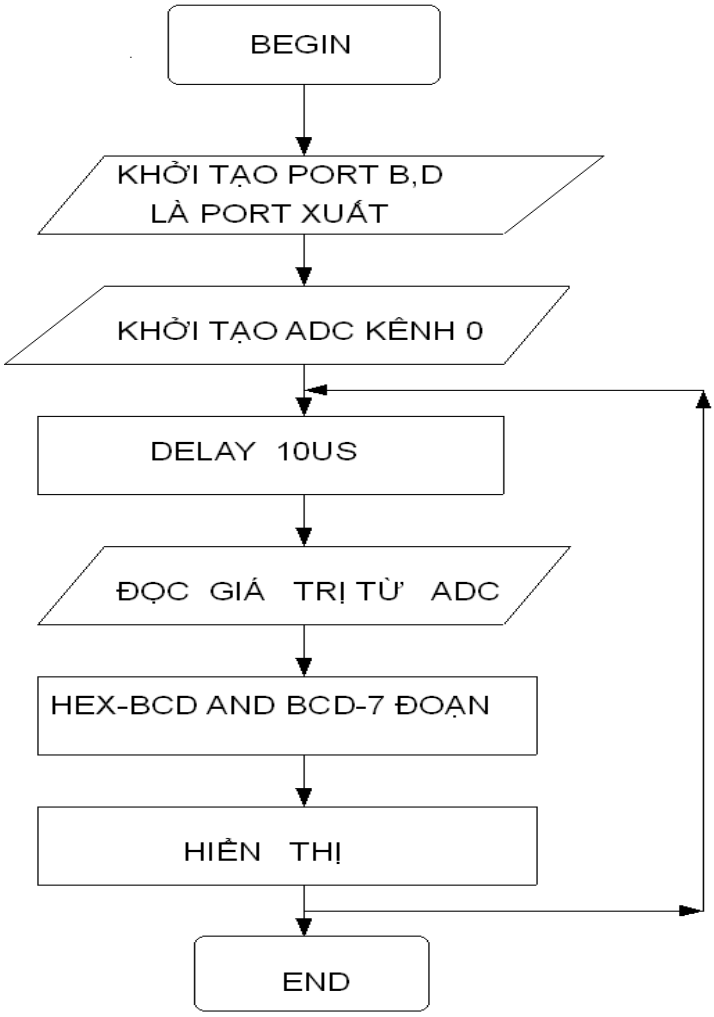
LƯU ĐỒ REAL ADC



LƯU ĐỒ HEX TO BCD:



LƯU ĐỒ CHUONG TRÌNH CHÍNH:



CHƯƠNG TRÌNH

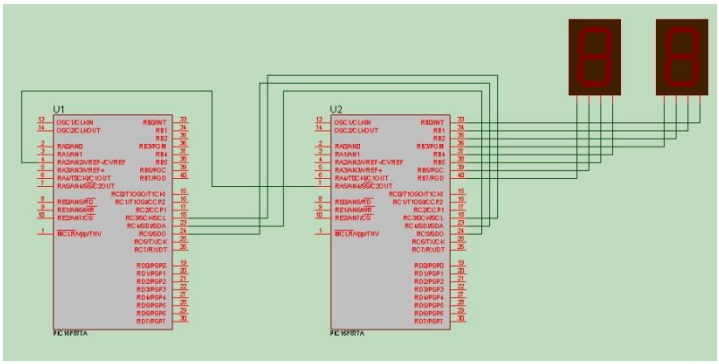
```
#include <16F877a.h>

#include <KhaiBaoTGhi_16F877A.h>
#fuses NOWDT,PUT,HS,NOPROTECT,NOLVP
#use delay(clock=20000000)

#device ADC=8
#use fast_io(d)
#use fast_io(b)
int8 i,tram,chuc,donvi;
signed int16 tam,doc;
const unsigned char
tra[]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x9
0,0xbf,0xff,0xc6};

void hex_bcd()  //chuyen doi so hex sang so bcd

{
    tram=tam/100;
   tam=tam%100;
   chuc=tam/10;
   donvi=tam%10;
}
void hienthi()
{
 i=0;
 while(i<200)
{
    portb=tra[chuc];//xuat hang chuc ra port
    portd=0x02;
   delay_us(5);
   portd=0x00;

   portb=tra[donvi];//xuat hang don vi
   portd=0x04;
   delay_us(5);
   portd=0x00;

   portb=tra[12];// xuat do c
   portd=0x08;
  delay_us(5);
   portd=0x00;

   if (doc<55)
{
    portb=tra[10];//xuat dau am port
    portd=0x01;
    delay_us(5);
```

```
portd=0x00;
}

else if (doc<155)

{

 portb=tra[11];      //xoa so 0 vo nghia
portd=0x01 ;

}
else
{
portb=tra[tram];    // xuat hang tram
portd=0x01;
delay_us(5);
portd=0x00;
}

i++;
}
}
```

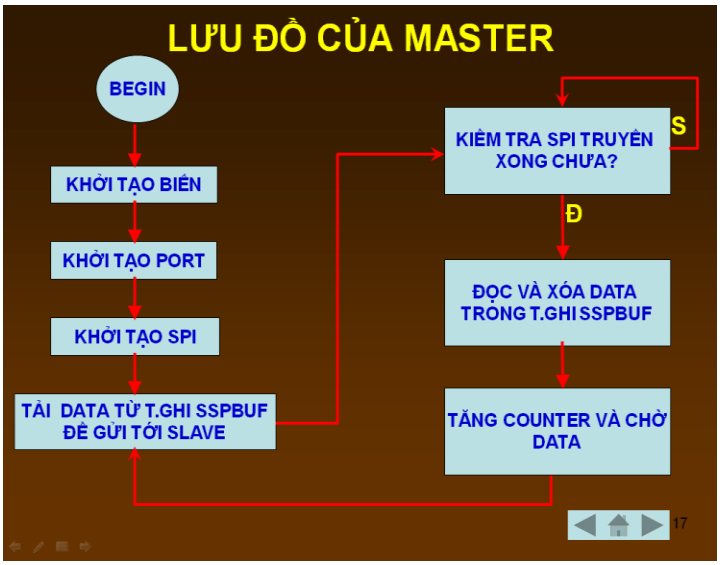
void main()

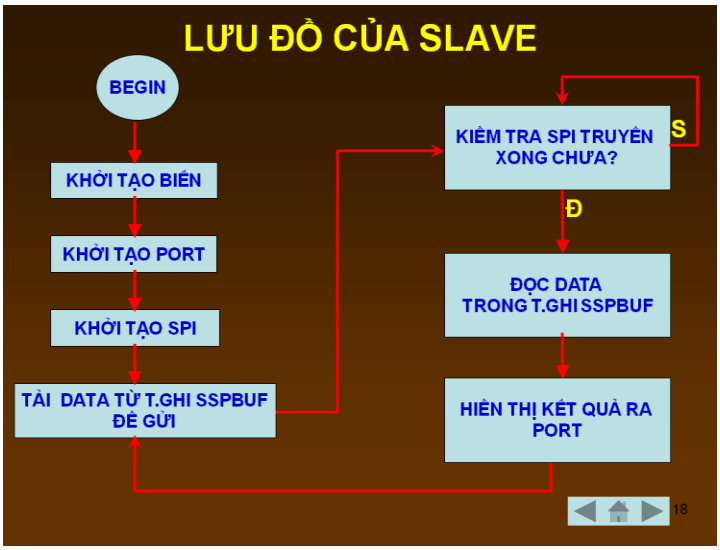
```
{

set_tris_d(0x00); //thiet lap xuat cho port d
set_tris_b(0x00); //thiet lap xuat cho port b



setup_adc(adc_clock_internal); //thoi gian lay mau bang xung clock IC(2-6us)

setup_adc_ports(an0_an1_vss_vref); //A0 A1 VRef+=A3,VRef-=0

set_adc_channel(0); //chon chan 0 doc tin hieu analog
delay_us(10); //delay 10us roi moi dung ham read_ADC   dam bao ket qua dung
while(1)
{
  doc=read_adc();
   tam=abs(doc-55);
   hex_bcd();
   hienthi();}
}
```

BÀI 11: KHẢO SÁT TRUYỀN DỮ LIỆU SPI CỦA PIC16F877A .SỬ DỤNG NGÔN NGỮ ASSEMBLY



LƯU ĐỒ





CHƯƠNG TRÌNH

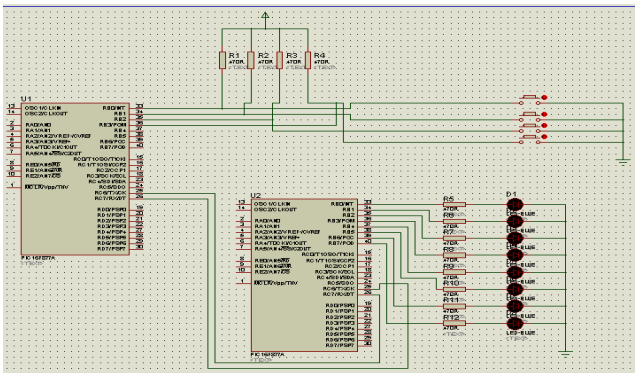
CHƯƠNG TRÌNH CỦA MASTER

```
list p=16F877A
#include "p16F877A.inc"
__CONFIG _CP_OFF & _DEBUG_OFF & _WRT_OFF&
_CPD_OFF& _LVP_OFF& _BODEN_OFF&
_PWRTE_ON& _WDT_OFF& _XT_OSC
Ctr0 EQU 0x20 ; Bien dem – gui du lieu den spi
Dly0 EQU 0x21 ; Bien delay 0
Dly1 EQU 0x22 ; Bien delay 1
#define SS PORTA,2 ; Gan chan chon slave (RA2)
ORG 0x000 ;

;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
; Khoi tao ho tro SPI
BANKSEL TRISA ; Chon bank 1
movlw 0x00 ; Khoi tao port A
movwf TRISA ; Xuat port A
movlw 0x06 ; Tat che do A/D (tuong tu)(trg 92)
movwf ADCON1 ; cho phep port A su dung nhu la port
xuat nhap so

;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
; Khoi tao SPI
BANKSEL TRISC ; Chon bank 1
movlw  b'00010000' ; SCK chan ra xung (Master),SDI chan nhan
data
movwf TRISC ; SDO chan ra data
movlw  b'01000000' ; du lieu dc lay mau tai thoi diem giua xung
clock
movwf SSPSTAT ; du lieu dc lay mau tai thoi diem giua
xung clock
BANKSEL SSPCON ; Chon bank 0
movlw b'00110001' ; 0001 SPI master mode,tan so xung
clock bang
;Fosc /16
movwf SSPCON ; bat che do SSP
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Send_DT     bcf SS ; Ngo ra cho phep lua chon chip (muc
thap)
movf Ctr0,W ; lay gia tri cua bien dem gan vao t.ghi W
movwf SSPBUF ; W => SSPBUF
BANKSEL SSPSTAT ; Chon bank 1
Char1    btfss SSPSTAT,BF ; kiem tra da truyen xong
chua?
Goto     Char1 ; nhay lai kiem tra tai nhan char1 neu chua
truyen xong
BANKSEL SSPBUF ; Chon bank 0
movf SSPBUF,W ; doc data tu t.ghi SSPBUF =>W
; du lieu nay ko dc dung
bsf     SS ; ngat ngo ra lua chon slave (muc cao la xoa)
incf    Ctr0,F ; tang bien dem

;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Delay          movlw 0xF0 ;
                movwf Dly1 ;
               movlw 0x0F ;
                movwf Dly0
DlyLoop        decfsz Dly0,F ;
               Goto     DlyLoop ;
               Decfsz   Dly1,F ;
 goto     DlyLoop ;
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

goto Send_DT ; quay lai gui data tiep theo (byte)
END
```

CHƯƠNG TRÌNH CỦA SLAVE

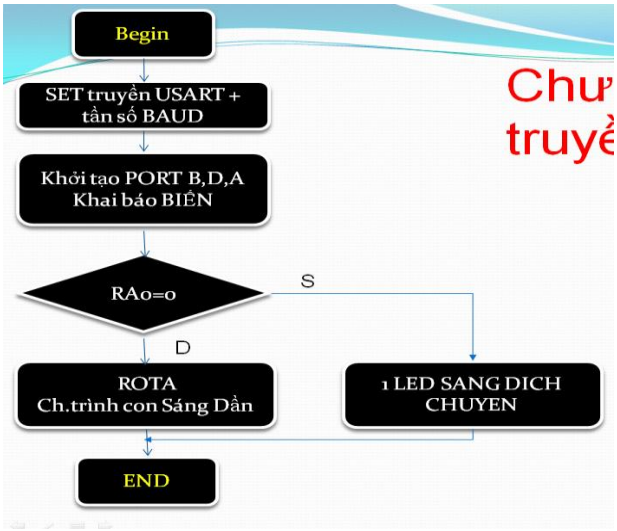
```
list p=16F877A
#include "p16F877A.inc"
__CONFIG _CP_OFF & _DEBUG_OFF &
_WRT_OFF & _CPD_OFF& _LVP_OFF &
_BODEN_OFF & _PWRTE_ON & _WDT_OFF &
_XT_OSC
ORG  0 ;
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
; Khoi tao ho tro SPI
BANKSEL TRISA ; Chon bank 1
movlw D'5' ; (trg 74)
movwf TRISA ; cho phep chan SS nhan du lieu
movlw 0x06 ; Tat che do A/D
movwf ADCON1 ; cho phep port A su dung nhu la
port xuat nhap so
```

BÀI 12: TRUYỀN DỮ LIỆU USART SỮ DỤNG ASSEMBLY

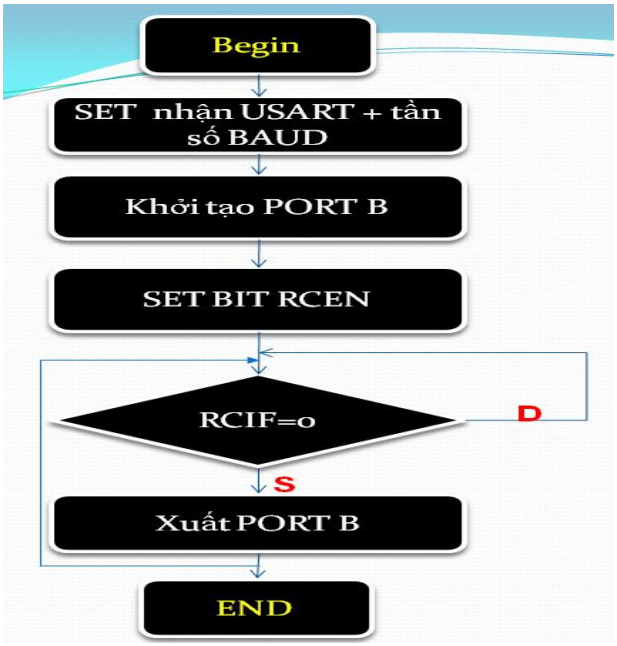


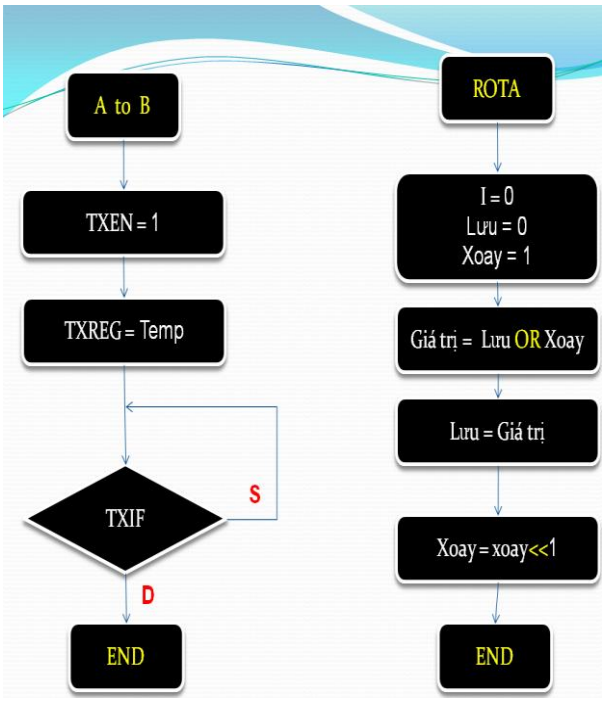
```
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
; khoi tao port ngo ra
BANKSEL TRISB ;
movlw 0x00 ; khoi tao port B la port xuat
movwf TRISB ;


;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
; khoi tao SPI
BANKSEL TRISC ; Chon bank 1
movlw B'00011000' ; SCK la ngo vao xung clock (Slave), SDI ngo vao ;du lieu
movwf TRISC ; SDO la ngo ra du lieu (b.gom cac ngo con lai cua port C)
movlw B'01000000' ; du lieu dc lay mau tai thoi diem giua xung clock
movwf SSPSTAT ; du lieu dc lay mau tai thoi diem giua xung clock
BANKSEL SSPCON ; Chon bank 0
movlw B'00110100' ; 0100 Mode 1,1 SPI Slave Mode,
movwf SSPCON ; bat che do SSP
;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
KT      movlw SSPSTAT ; Kiem tra gian tiep
movwf FSR ; W=>FSR (trg 63)
btfss INDF,BF ; kiem tra bit BF trong t.ghi INDF (trg 63)
goto KT ;
RX_Data
BANKSEL SSPBUF ; chon bank 0
movf SSPBUF,W ; SSPBUF=>W
BANKSEL PORTB ; chon bank 0
movwf PORTB ; Xuat du lieu ra port B

;xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
goto KT ; quay lai nhan byte tiep theo
END
```

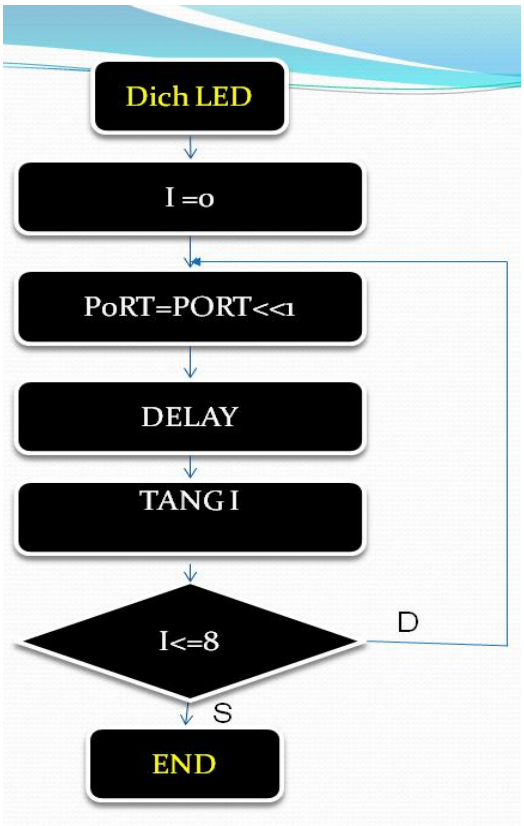
CHƯƠNG TRÌNH TRUYỀN PIC A



CHƯƠNG TRÌNH NHẬN PIC B







CHƯƠNG TRÌNH

CHƯƠNG TRÌNH TRUYỀN

```
title   "TRUYEN USART"
processor               p16f877a
include         <p16f877a.inc>
__CONFIG
        _CP_OFF&_WDT_OFF&_LVP_OFF&_PWRTE_
ON&_HS_OSC&_BODEN_OFF&_CPD_OFF
;**************************************
;KHAI BAO CAC HANG SO
;**************************************
SW1     EQU     0
SW2     EQU     1
SW3     EQU     2
SW4     EQU     3
LED1    EQU     4
LED2    EQU     5
LED3    EQU     6
LED4    EQU     7
;**************************************
;KHAI BAO BIEN
;**************************************
COUNT           EQU     0x20
COUNT1          EQU     0x21
COUNTa          EQU     0x22
COUNTb          EQU     0x23
;**************************************
;CHUONG TRINH CHINH
;**************************************
        ORG     0x000
        GOTO    Start
Start
        bcf     STATUS,RP1
        bsf     STATUS,RP0      ; Chon bank 1
        clrf    PORTB
        clrf    PORTD
        bsf     STATUS,RP0      ;chon bank 0
        movlw   B'00001111'
        movwf   TRISB
        movlw   0x00
        movwf   TRISD
        bsf     STATUS,RP0      ; Chon bank 1
        call    khoitaousart
```

```
        bcf     STATUS,RP0   ; Chon bank 0
Loop1
        clrf    COUNT
        call    CHECK_KEY
Loop2
        movf    COUNT,W
        btfsc   PORTB,LED1
        call    TABLE1

        btfsc   PORTB,LED2
        call    TABLE2

        btfsc   PORTB,LED3
        call    TABLE3

        btfsc   PORTB,LED4
        call    TABLE4
        movwf   PORTD
        call    transmitAtoB
        call    DELAY
        incf    COUNT,0
        xorlw   D'8'
        btfsc   STATUS,Z
        goto    Loop1

        incf    COUNT,1
        goto    Loop2
;**************************************
;SANG TU PHAI SANG TRAI
;**************************************
TABLE1
        addwf   PCL,F
        retlw   b'10000000'
        retlw   b'11000000'
        retlw   b'11100000'
        retlw   b'11110000'
        retlw   b'11111000'
        retlw   b'11111100'
        retlw   b'11111110'
        retlw   b'11111111'
;**************************************
;SANG TU GIUA RA 2 BEN
;**************************************
TABLE2
        addwf   PCL,F

        retlw   b'00011000'
        retlw   b'00100100'
        retlw   b'01000010'
        retlw   b'10000001'
        retlw   b'01000010'
        retlw   b'00100100'
        retlw   b'00011000'
        retlw   b'00100100'
;**************************************
;SANG XEN KE 2 DIEM
;**************************************
TABLE3
        addwf   PCL,F

        retlw   b'10000000'
        retlw   b'11000000'
        retlw   b'01100000'
        retlw   b'00110000'
        retlw   b'10011000'
        retlw   b'11001100'
        retlw   b'01100110'
        retlw   b'11011011'
;**************************************
;SANG DAN TU TRAI SANG PHAI
;**************************************
TABLE4
        addwf   PCL,F
        retlw   b'00000001'
        retlw   b'00000011'
        retlw   b'00000111'
        retlw   b'00001111'
        retlw   b'00011111'
        retlw   b'00111111'
        retlw   b'01111111'
        retlw   b'11111111'
;**************************************
;CHECK_KEY
;**************************************
CHECK_KEY
        btfss   PORTB,SW1
        call    SWITCH1
```

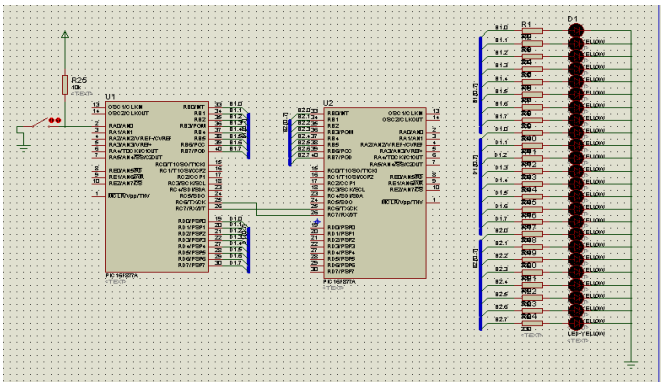
```
        btfss   PORTB,SW2
        call    SWITCH2

        btfss   PORTB,SW3
        call    SWITCH3

        btfss   PORTB,SW4
        call    SWITCH4
        return
;**************************************
;CHECK_SWITCH
;**************************************
SWITCH1
        clrf    PORTB
        bsf     PORTB,LED1
        return
SWITCH2
        clrf    PORTB
        bsf     PORTB,LED2
        return
SWITCH3
        clrf    PORTB
        bsf     PORTB,LED3
        return
SWITCH4
        clrf    PORTB
        bsf     PORTB,LED4
        return
;**************************************
;DELAY
;**************************************
DELAY
        movlw   D'300'
        movwf   COUNT1
D1
        movlw   0xC7
        movwf   COUNTa
        movlw   0x01
        movwf   COUNTb
DELAY_0
        decfsz  COUNTa,1
        goto    $+2
        decfsz  COUNTb,1
        goto    DELAY_0
        decfsz  COUNT1,1
        goto    D1
        return
;**************************************
;KhoiTaoUSART:
;**************************************
khoitaousart
        movlw   D'129'          ;Gia tri toc do baud ung voi
9600, BRGH = 1, xtal 20 MHz
;       bcf     STATUS,RP1
;       bsf     STATUS,RP0      ;Chon bank cho SPBRG

        movwf   SPBRG           ;Dat toc do baud = 9600
        movlw   0x24            ;Che do: high-speed, cho
phep xuat TXEN=1, truyen bat dong bo
        movwf   TXSTA           ;TXSTA cung bank voi
SPBRG

        movlw   0x90            ;Che do: nhan lien tuc, 8-
bit, dung cac chan TX/RX
        BCF     STATUS,RP0      ; Chon bank RCSTA
        bank0

        movwf   RCSTA
        return
;**************************************
;chuongtrinhTRUYENA_B:
;**************************************
transmitAtoB
        MOVWF       TXREG
        btfsc   PIR1,10
        GOTO    transmitAtoB
        RETURN
;**************************************
END
```

CHƯƠNG TRÌNH NHẬN

```
title   "NHAN USART"
processor               p16f877a
include         <p16f877a.inc>
__CONFIG
        _CP_OFF&_WDT_OFF&_LVP_OFF&_PWRTE_
ON&_HS_OSC&_BODEN_OFF&_CPD_OFF
```

```
;**************************************
;CHUONG TRINH CHINH
;**************************************
        ORG     0x000
        GOTO  Start
Start
        bcf     STATUS,RP1
        bsf     STATUS,RP0   ; Chon bank 1
        clrf    PORTB
        bsf     STATUS,RP0   ;chon bank 0
        movlw   B'00000000'
        movwf   TRISB
        bsf     STATUS,RP0   ; Chon bank 1
        call    khoitaousart
loop    call    transmitAtoB
        MOVWF       PORTB
        goto    loop
;**************************************
;KhoiTaoUSART
;**************************************
khoitaousart
        movlw   D'129'          ;Gia tri toc do baud ung voi
9600, BRGH = 1, xtal 20 MHz
        movwf   SPBRG           ;Dat toc do baud = 9600
        movlw   0x04            ;Che do: high-speed, truyen
bat dong bo
        movwf   TXSTA           ;TXSTA cung bank voi
SPBRG
        movlw   0x90            ;Che do: nhan lien tuc, 8-
bit, dung cac chan TX/RX
        BCF     STATUS,RP0   ; Chon bank RCSTA bank0
        movwf   RCSTA
        return
;**************************************
;chuongtrinhTRUYENA_B
;**************************************
transmitAtoB
        btfsc   PIR1,20
        GOTO    transmitAtoB
        MOVF    RCREG,w
        RETURN
;**************************************
end
```

BÀI 12 SỮ DỤNG C:



TRUYỀN:

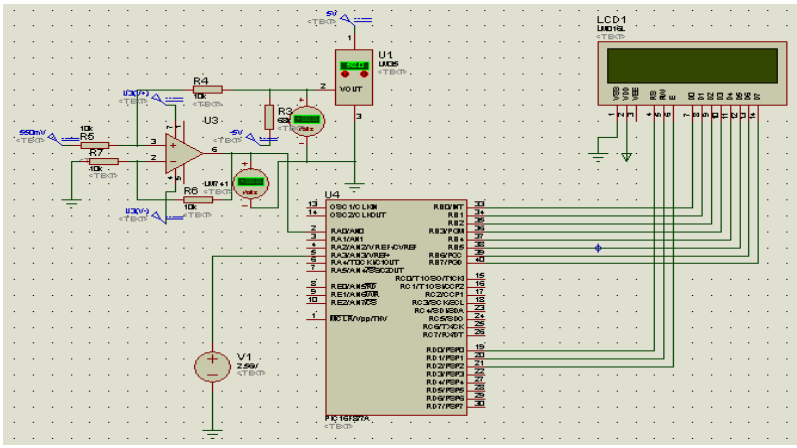
```
  #include<16f877a.h>
  #include<KhaiBaoTGhi_16F877A.H>
  #fuses NOWDT,PUT,HS,NOPROTECT,NOLVP
  #use delay(clock=20000000)
  #use rs232(baud=9600,xmit=pin_C6,rcv=pin_C7)  //khai
bao toc do baud su dung chuan rs232
  #use  fast_io(b)                  //khai bao dung port b
  #use  fast_io(d)
  #use  fast_io(a)                  //kb dung port d
  #locate b_B=0x22
  int8 i,giatri,luu,xoay;
  //=====================================
  //chuong trinh con
  //=====================================
   void A_to_B()
  {
   TXEN=1;        //cho phep truyen du lieu   TXSTA<5>
   TXREG=b_B;   // load data vao TG dem truyen
    while(~TXIF)    // cho den khi load xog  PIR1<4>
  }
    void rota()
  {
   giatri=luu|xoay;   // luu or xoay
   luu=giatri;
   xoay=xoay<<1;
  }
//=========================================
  //chuong trinh chinh
//=========================================
  void main ()
```

```
{
  set_tris_b(0x00);  // set portB out
  set_tris_d(0x00);  // set port D out
  set_tris_a(0x01);
while(1)
  {
    if(ra0==0)
  {
    i=0;
    luu=0;
    xoay=1;
    portb=0;
    portd=0;
    b_B=0;
    A_TO_B();          //goi b_B  qua portb cua PIC 2
    delay_ms(100);
    for(i=1;i<=8;i++)
      {
      rota();
      portb=giatri;
      delay_ms(100);
      }

    i=0;
    luu=0;
    xoay=1;
     for(i=1;i<=8;i++)
      {
       rota();
       portd=giatri;
       delay_ms(100);
      }
     i=0;
    luu=0;
    xoay=1;
    for(i=1;i<=8;i++)
      {  rota();
       b_B=giatri;
       A_TO_B();
       delay_ms(100);
      }
  }
else
    {
    i=0;
    portb=0x00;
    portd=0x00;
    b_B=0x00;
    A_TO_B();          //goi b_B  qua portb cua PIC 2
    delay_ms(100);
    portb=0x01;
    delay_ms(100);
    for(i=1;i<=8;i++)
    {
     delay_ms(100);
     portb=portb<<1;
    }
     portb=0x00;
     i=0;
     portd=0x01;
     for(i=1;i<=8;i++)
     {

      delay_ms(100);
      portd=portd<<1;
     }
    portd=0x00;
    i=0;
    b_B=0x01;
    for(i=1;i<=8;i++)
     {
      A_TO_B();
      delay_ms(100);
      b_B=b_B<<1;
     }

      }
    }
  }
```

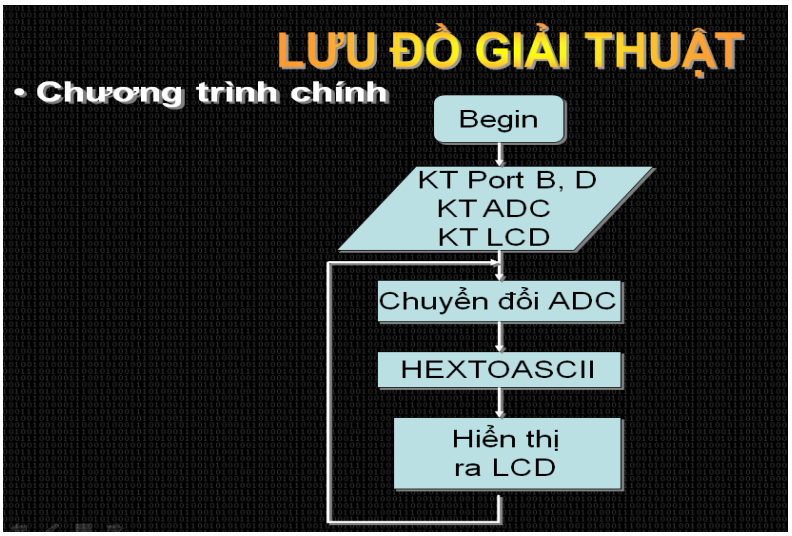
NHẬN:

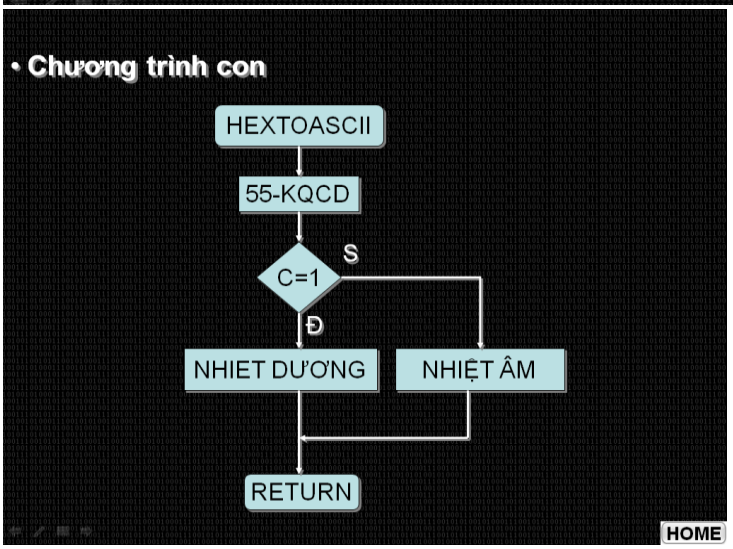
```
#include<16f877a.h>
  #include<KhaiBaoTGhi_16F877A.H>
  #fuses NOWDT,PUT,HS,NOPROTECT,NOLVP
  #use delay(clock=20000000)
  #use rs232(baud=9600,xmit=pin_C6,rcv=pin_C7)  //khai bao toc do baud
  #use fast_io(b)
```

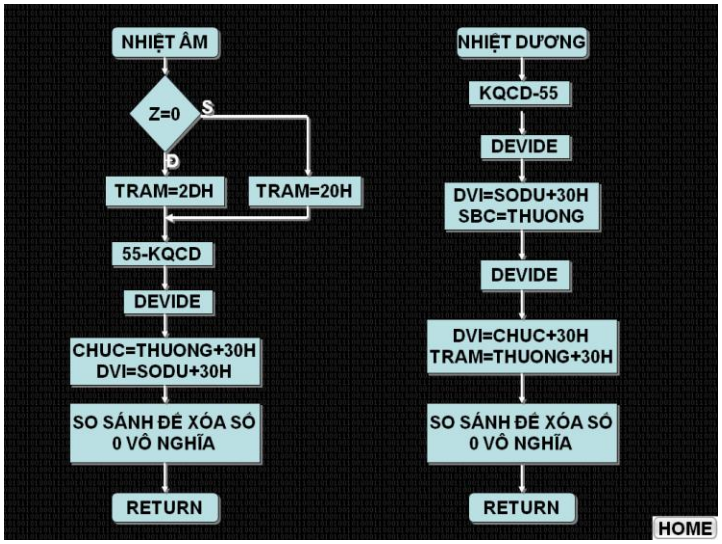
BÀI 13: Sử dụng Pic 16F877A để đo nhiệt độ với LM35, hiển thị LCD,ngõ vào kênh AN0.SỬ DỤNG ASSEMBLY



```
//=========================================
  // chuong trinh chinh

//=========================================
  main()
  {
   set_tris_b(0x00);
   RCEN=1;     //cho phep nhan du lieu
   while(1)
   {
   while(~RCIF)
      {
      }
    portb= rcreg;
    rcreg=0;
   }
```







CHƯƠNG TRÌNH:

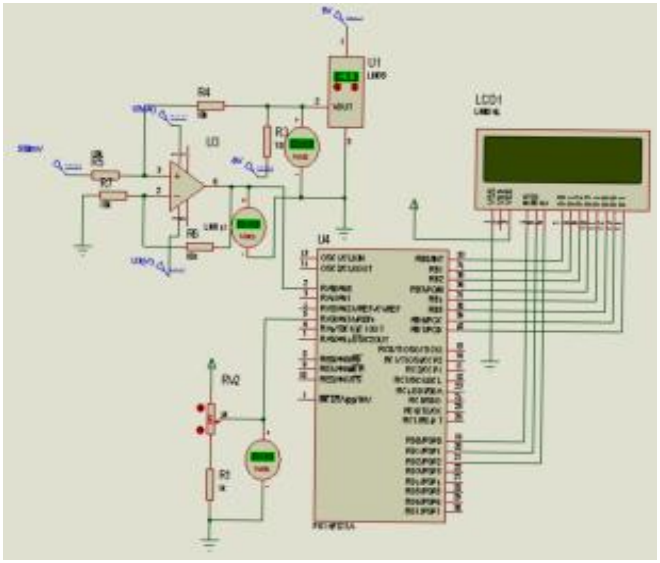
```
; Ten chuong trinh :Mach do nhiet do dung cam bien LM35 hien thi LCD
; Mo ta phan cung  : Dung PIC16f877aA - thach anh 20MHz
;    :cam bien nhiet LM35 mac vao kenh AN0 cua port0
;    :LCD giao tiep 8 bit.
;    :RS, RW, E mac vao E mac vao 3 chan RD0, RD1 RD2
;=========================================
  TITLE   "Mach Do Nhiet Dung Cam Bien LM35 Hien Thi LCD.asm"
  PROCESSOR P16F877A
  INCLUDE  <P16F877A.inc>
  __CONFIG     _CP_OFF & _PWRTE_ON & _XT_OSC & _WDT_OFF & _HS_OSC & _LVP_OFF
;=========================================
        CBLOCK        0x30
        COUNT
        TRAM
        CHUC
        DVI
        KQCD
        SOBICHIA
        SOCHIA
        SODU
        THUONG
        COUNT1
        COUNT2
        COUNT3
        COUNT4
        A
        TAM
        ENDC
;=========================================
        ORG    0X0000
        GOTO  START
START
        BANKSEL TRISB
        CLRF TRISB    ;PORTB XUAT LED
        CLRF TRISD
        CALL KTADC
        CALL KTLCD
MAIN
        CALL   CHUYENDOI
        CALL   HEXTOASCII
        CALL   DONG1
        CALL   DONG2
        GOTO  MAIN
;*************************************************
KTADC
        BANKSEL ADCON1
        MOVLW 0X01
;NGO VAO ANALOG,RIGHT JUSTIFY,Fosc/2
        MOVWF ADCON1
        BANKSEL ADCON0
        MOVLW         0XC1    ;CHON RA0 LAM NGO VAO,ADON=1,BAT ADC
        MOVWF ADCON0
        CALL DELAY1
;CHO 1 KHOANG THOI GIAN TRUOC KHI BAT DAU
;*************************************
;KIEM TRA KET QUA
CHUYENDOI
        BSF ADCON0,GO
        ;SET BIT "GO",BAT DAU CONVERT
        BTFSC ADCON0,GO
        ;KIEM TRA BIT "DONE" CUA ADCON0
        GOTO $-1;NEU BANG 1 THI CHO TIEP TUC
        MOVF ADRESL,W
        MOVWF KQCD

        RETURN
;*************************************************
DELAY1
        MOVLW D'10'
```

```
            MOVWF COUNT
BACK
            MOVLW D'4'
            MOVWF COUNT1
            DECFSZ COUNT1,F
            GOTO $-1
            DECFSZ COUNT,F
            GOTO BACK
            RETURN
;***********************************************
;chuong trinh con chuyen ma hex sang MA LCD
;---------------------------------------------------------------------
HEXTOASCII
        MOVF KQCD,0
        SUBLW       D'55'               ;so sanh
ket qua voi 55
        BTFSS STATUS,C          ;so sanh voi co c
        GOTO NHIETDUONG                 ;co
C=0, tuc phep toan co kq <0
        GOTO NHIETAM                ;co C=1,
tuc phep toan co kq >0
NHIETAM     ;giai ma nhiet do am tu -55 den -1 do C
        BTFSC       STATUS,Z
        GOTO Y1
        MOVLW       B'00101101'
        MOVWF       TRAM                ;ma asscii
cua dau '-'
        GOTO Y2
Y1      MOVLW       B'00100000'
        MOVWF       TRAM
Y2      MOVLW       D'55'
        MOVWF       TAM
        MOVF KQCD, 0
        SUBWF       TAM,  0
        MOVWF       SOBICHIA
        MOVLW       D'10'
        MOVWF       SOCHIA
        CALL  DIVIDE
        MOVF SODU, 0
        ADDLW       B'00110000'
        MOVWF       DVI
        MOVF THUONG,      0
        ADDLW       B'00110000'
        MOVWF       CHUC
        MOVF CHUC, 0;kiem tra hang chuc de xoa so 0 vo
nghia
        XORLW B'00110000'  ;so sanh voi so 0
        BTFSS STATUS,       Z
        GOTO EXIT   ;thoat chuong trinh con neu chuc
khac 0
        MOVLW       B'00100000'     ;neu chuc la 0 thi
hien thi khoang trang
        MOVWF       CHUC
        GOTO EXIT
NHIETDUONG
        MOVLW       D'55'
        SUBWF       KQCD,0
        MOVWF       SOBICHIA;sobichia = kqcd - 55
        MOVLW       D'10'
        MOVWF       SOCHIA
        CALL  DIVIDE
        MOVF SODU, 0
        ADDLW       B'00110000'
        MOVWF       DVI
        MOVF THUONG,      0
        MOVWF       SOBICHIA
        CALL  DIVIDE
        MOVF SODU, 0
        ADDLW       B'00110000'
        MOVWF       CHUC
        MOVF THUONG,      0
        ADDLW       B'00110000'
        MOVWF       TRAM
        MOVF TRAM, 0                ;kiem tra hang
tram de xoa so 0 vo nghia
        XORLW       B'00110000'     ;so sanh voi so 0
        BTFSS STATUS,       Z
        GOTO EXIT   ;thoat chuong trinh con neu tram
khac 0
        MOVLW       B'00100000'             ;neu tram
la 0 thi hien thi khoang trang
        MOVWF       TRAM
        MOVF CHUC, 0        ;kiem tra hang chuc de xoa
so 0 vo nghia
        XORLW       B'00110000'     ;so sanh voi so 0
        BTFSS STATUS,       Z
        GOTO EXIT                   ;thoat chuong trinh
con neu chuc khac 0
```

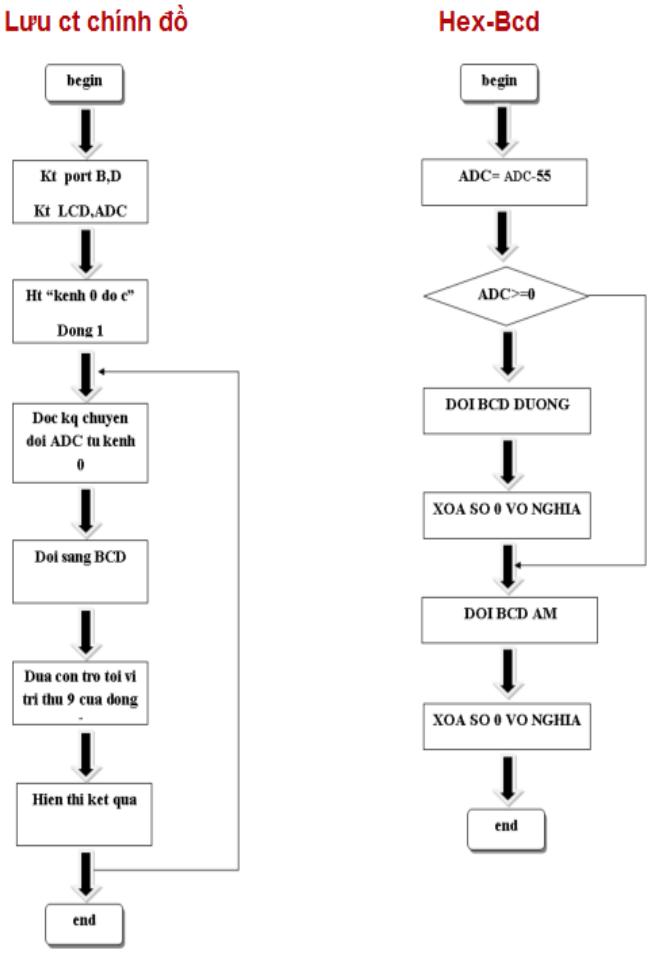
```
        MOVLW       B'00100000'
        ;neu chuc la 0 thi hien thi khoang trang
        MOVWF       CHUC
        GOTO  EXIT
EXIT
        RETURN
;-------------------------------------------------------------------
;chuong trinh con divide
;-------------------------------------------------------------------
DIVIDE
        CLRF   THUONG
        MOVF  SOBICHIA,0
        MOVWF       SODU
X       MOVF  SOCHIA,       0
        SUBWF       SODU,0          ;W=sodu-
sochia, c=1 neu kq>=0 va nguoc lai
        BTFSS STATUS,       C       ;bo qua lenh ke neu
c=1, kq duong
        GOTO  EXIT1                 ;thoat khoi chuong
trinh con neu kq am
        INCF    THUONG,1 ;tang thuong so len 1
        MOVWF       SODU
        GOTO  X
EXIT1
        RETURN
;-------------------------------------------------------------------
;chuong trinh con khoi tao LCD
;-------------------------------------------------------------------
KTLCD
        MOVLW       0x38;Set 8 bit mode, hien thi 2
hang, kieu ki tu 5x8
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY40MS
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY40MS
        MOVLW       0X0C    ;display on/off control
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY40MS
        MOVLW       0X01    ;tat hien thi,con tro doi ve
goc trai
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY40MS
        RETURN
GHIMADK
        MOVF  A,0
        MOVWF       PORTB
        BCF         PORTD,0
        BCF         PORTD,1
        BSF         PORTD,2
        BCF         PORTD,2
        RETURN
;-------------------------------------------------------------------
;chuong trinh con ghi nhiet do ra dong 1
;-------------------------------------------------------------------
DONG1
        MOVLW       0x80    ;ve dau dong thu nhat
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY
        CLRF   COUNT4
LABEL1
        MOVF  COUNT4,       0
        CALL   TABLE1
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        INCF    COUNT4,0
        XORLW       D'10'
        BTFSC STATUS,       Z
        GOTO  L1
        INCF    COUNT4,1
        GOTO  LABEL1
L1      MOVF  TRAM, 0
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        MOVF  CHUC, 0
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        MOVF  DVI,    0
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
```

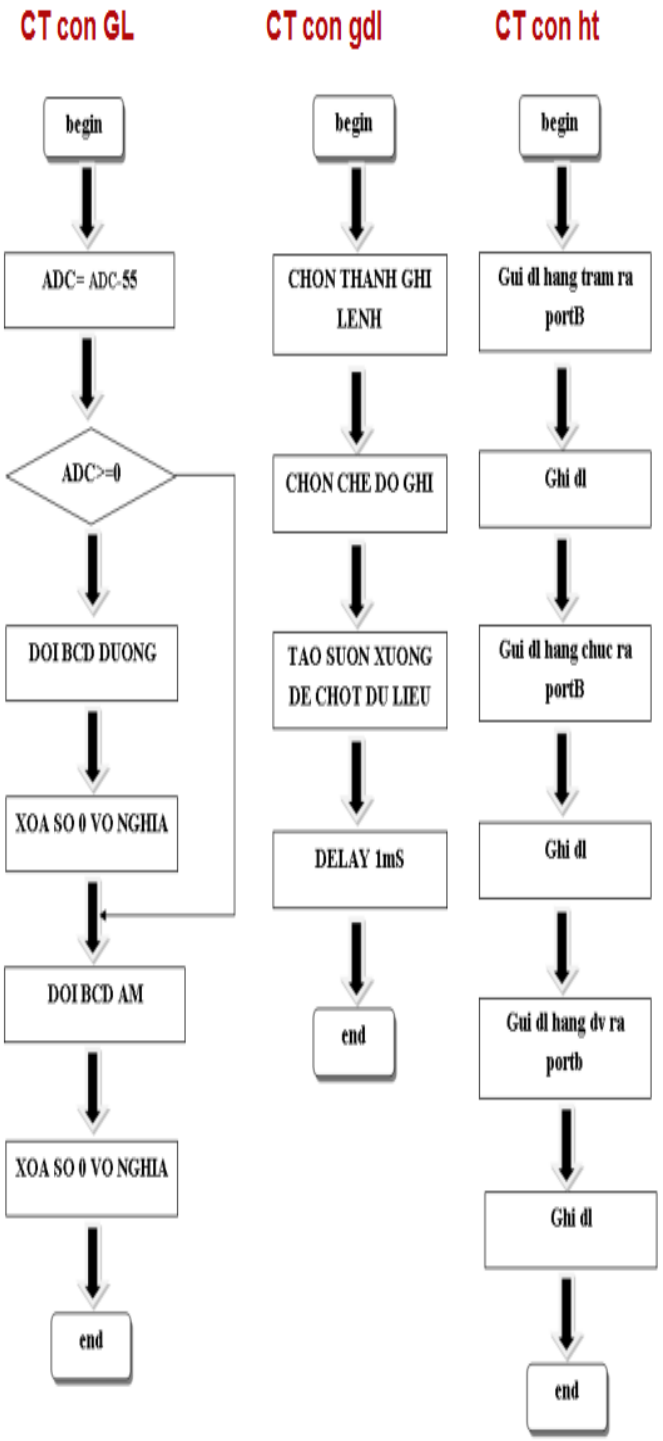
```
        CLRF   COUNT4
LABEL2
        MOVLW       B'11011111'
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        MOVF COUNT4,        0
        CALL   TABLE2
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        RETURN
;-------------------------------------------------------------------
;chuong trinh con hien thi chu dong 2
;-------------------------------------------------------------------
DONG2
        MOVLW       0xc1
        MOVWF       A
        CALL   GHIMADK
        CALL   DELAY
        CLRF   COUNT4
LABEL3
        MOVF COUNT4,        0
        CALL   TABLE3
        MOVWF       A
        CALL   GHIDL
        CALL   DELAY
        INCF   COUNT4,0
        XORLW       D'7'
        BTFSC STATUS,       Z
        GOTO  EX1
        INCF   COUNT4,1
        GOTO  LABEL3
EX1     RETURN

;-------------------------------------------------------------------
;bang ma ki tu cua tabel
;-------------------------------------------------------------------
TABLE1
        ADDWF       PCL,    1
        DT          "NHIET DO: "
TABLE2
        ADDWF       PCL,    1
        DT          "C"
TABLE3
        ADDWF       PCL,    1
        DT          "NHOM 13"
;-------------------------------------------------------------------
;chuong trinh con ghi ki tu ra LCD
;-------------------------------------------------------------------
GHIDL
        MOVF  A,0
        MOVWF       PORTB
        BSF         PORTD,0
        BCF         PORTD,1
        BSF         PORTD,2
        BCF         PORTD,2
        RETURN
;-------------------------------------------------------------------
;cac chuong trinh con delay
;-------------------------------------------------------------------
DELAY       MOVLW       D'255'
            MOVWF       COUNT3
DELA1 DECFSZ        COUNT3
            GOTO  DELA1
            RETURN
DELAY40MS
        MOVLW D'255'
        MOVLW COUNT1
DE1     MOVLW       0XFF
        MOVWF       COUNT2
DE2     DECFSZ COUNT2
        GOTO  DE2
        DECFSZ      COUNT1
        GOTO  DE1
        RETURN
END
```

BÀI 14: Đo nhiệt độ dùng LM35 kết nối với pic hiển thị trên LCD dùng kênh 0.SỮ DỤNG C



LƯU ĐỒ



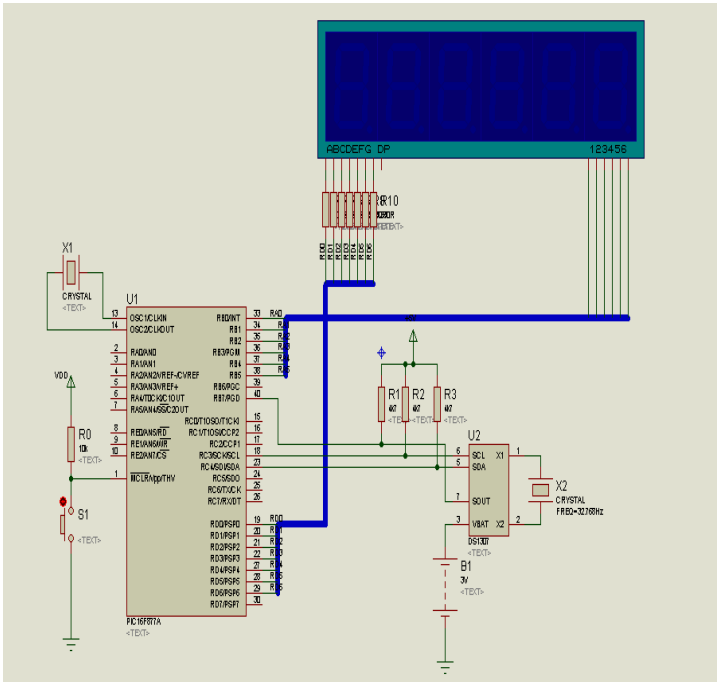


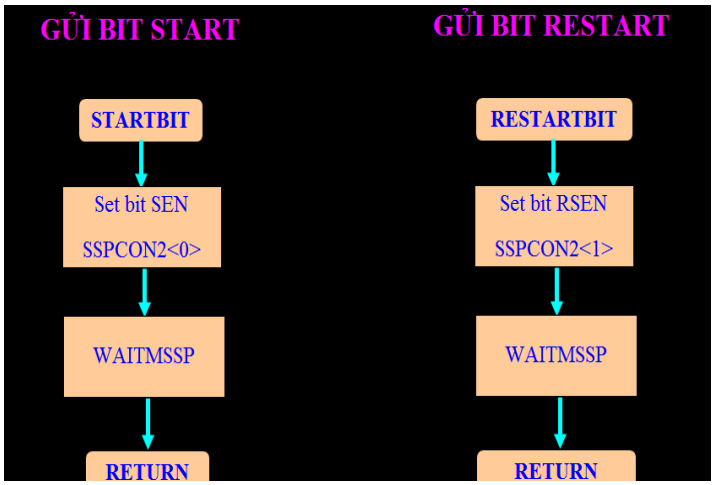
CHƯƠNG TRÌNH

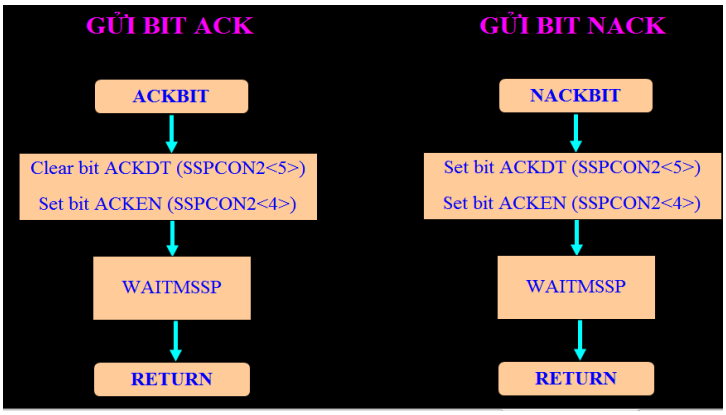
```
#include <16f877a.h>
#include "def_877a.h"
#fuses nowdt,noprotect,nolvp,put,hs
#use delay(clock=20000000)
#define RS rD0
#define RW rD1
#define E  rD2
#define LCD PORTB
signed int16 ADC0,tam;
int i=0;
int dv=0;
int chuc=0;
int tram=0;
const unsigned char mht[]={'0','1','2','3','4','5','6','7','8','9','-',' '};
const unsigned char nd0[]="kenh 0:    do C";

void gl(void)
{
    RS = 0;
    RW=0;
    E=1;
    E=0;
    delay_ms(1);
}
void gdl(void)
{
   Rw=0;
   Rs=1;
   E=1;
   E=0;
   DELAY_MS(1);
}

void hex_bcd(signed int16 adc )
{
   ADC = ADC-55;
   if(ADC>=0)
   {
      dv=ADC%10;
      tam=ADC/10;
      chuc=tam%10;
      tram=tam/10;
```

```
      if(tram==0)
        {
         tram=13;
         if(chuc==0) chuc=13;
        }
  }
  else
   {
     ADC=abs(ADC);
     tram=10;
     chuc=ADC/10;
     dv=ADC%10;
     if(chuc==0) chuc=13;
   }
}

void ht()
{
    LCD=mht[tram];
    gdl();
    LCD=mht[chuc];
    gdl();
    LCD=mht[dv];
    gdl();
}

void kt_lcd()
{
    LCD = 0x38;
    gl();
    LCD =0x38;
    gl();
    LCD=0x0C;
    gl();
}
void main()
{
  trisb=0;
  trisd=0;
 setup_adc(ADC_CLOCK_INTERNAL);
 setup_adc_ports(an0_an1_vss_vref);

 kt_LCD();
 LCD=0x80;
 gl();
 while(i<=15)
 {
  LCD=nd0[i];
  gdl();
  i++;
 }
 i=0;
 while(1)
 {
   set_ADC_channel(0);
   delay_us(10);
   ADC0=read_ADC();
   hex_bcd(adc0);
   LCD=0x88;
   gl();
   ht();
 }
}
```

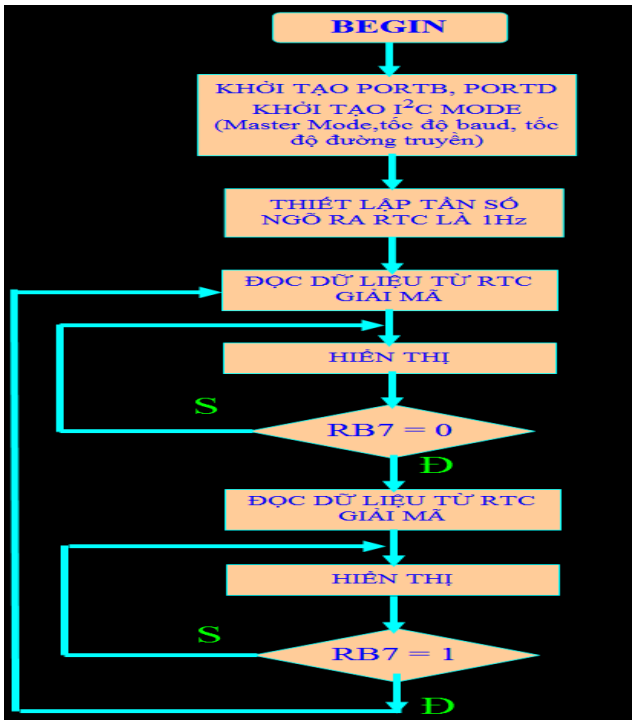
BÀI 15: DÙNG PIC 16F877A GIAO TIẾP VỚI REALTIME DS13B07 VÀ 6 LED 7 ĐOẠN VIẾT CT ĐỒNG HỒ SỐ DÙNG NGÔN NGỮ ASSEMBLY

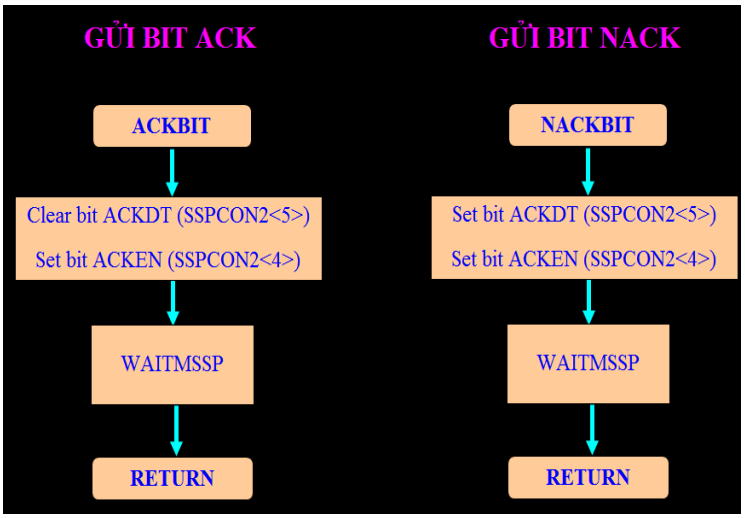


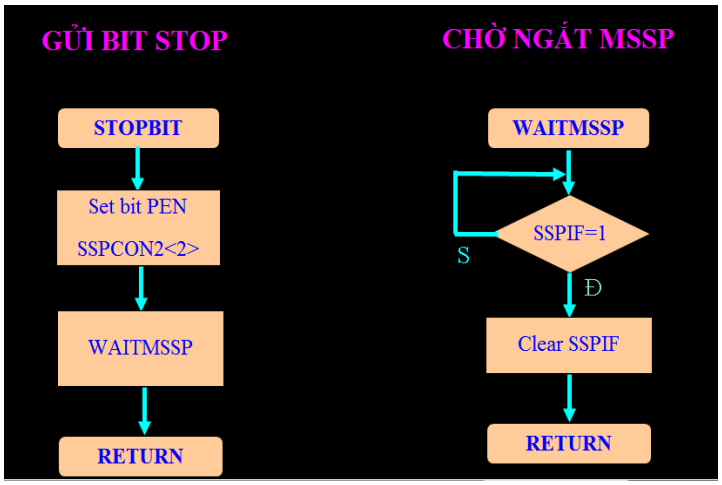


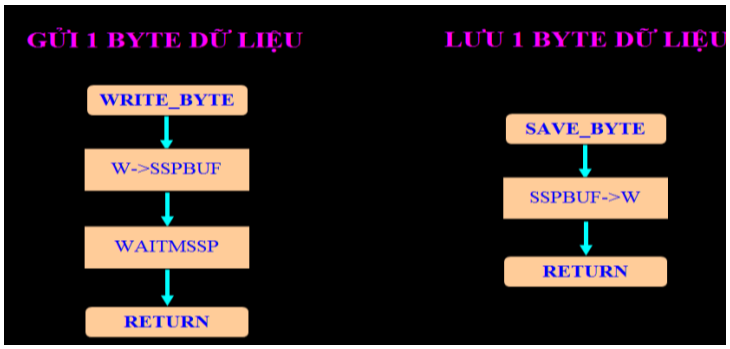


CHƯƠNG TRÌNH CHÍNH









CHƯƠNG TRÌNH

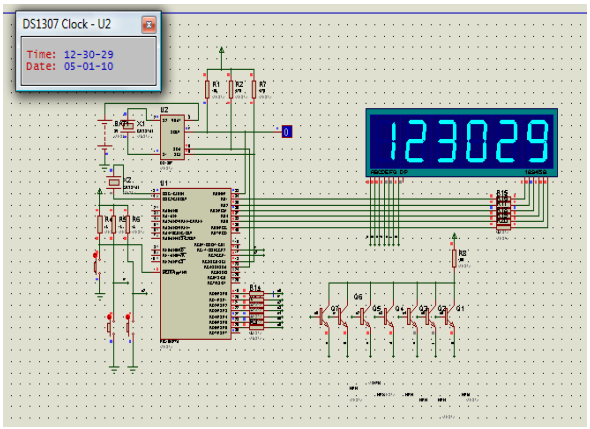
```
title "GIAO TIEP RTC DS1307 HIEN THI THOI GIAN THUC"
processor p16f877a
include <P16f877a.inc>
__CONFIG _CP_OFF & _PWRTE_ON & _WDT_OFF & _HS_OSC & _LVP_OFF
;=================================
;----------KHAI BAO BIEN--------
        CBLOCK      0X20
        RAM_SEC
        RAM_MIN
        RAM_HOU
        RAM_SEC1
        RAM_MIN1
        RAM_HOU1
        RAM_SEC2
        RAM_MIN2
        RAM_HOU2
        TEMP        ; BIEN DUNG CHO CT DELAY
        ENDC
;=========================================
==============================
        ORG   0X00
        GOTO  MAIN
MAIN
;KHOI TAO PORTD
        BCF   STATUS,RP1
        BSF   STATUS,RP0
        CLRF  TRISD         ;THIET LAP PORTD LA NGO RA DE DUA DU LIEU DEN LED
        MOVLW       0X80            ;RB0->RB6 LA NGO RA DE QUET LED
        MOVWF       TRISB           ;CHAN RB7 LA NGO VAO CHO TIN HIEU DIEU KHIEN
        BCF   STATUS,RP0
;KHOI TAO I2C MASTER MODE
        BANKSEL SSPCON              ; chon MSSP che do MASTER MODE
        MOVLW   B'00101000' ; Thiet lap MSSP cho chan PORTC o che do I2C
        MOVWF   SSPCON
        BANKSEL SSPSTAT             ; Thiet lap che do Chuan cua toc do BAUD(100KHz)
        MOVLW   B'10000000'
```

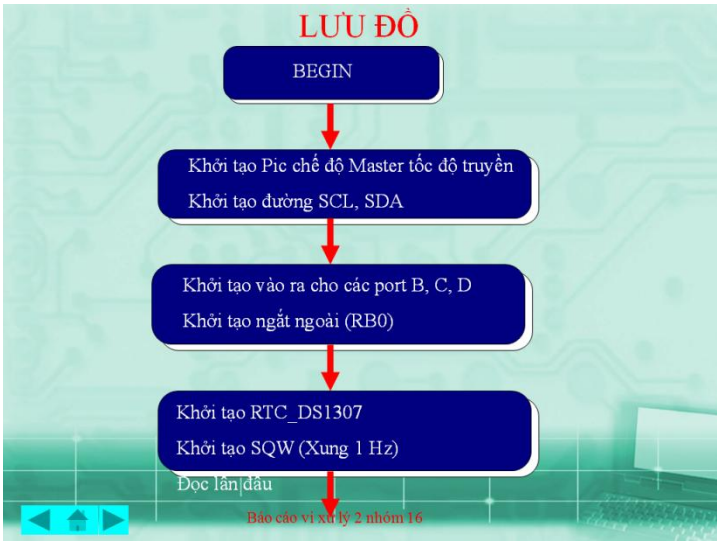
```
        MOVWF   SSPSTAT
        BANKSEL SSPADD          ;
SSPADD=F(osc)/(tocdoBaud)
        MOVLW   0X28            ; Tinh toan gia tri
cho thiet lap toc do 0X28=4MHz/100KHz
        MOVWF   SSPADD
;=======================================
============================
;THIET LAP TAN SO NGO RA CUA DS1307 LA 1Hz
        CALL    STARTBIT        ; GUI BIT START

        MOVLW   0XD0                    ; GUI
DIA CHI CHO PHEP RTC NHAN
        CALL    WRITE_BYTE
        MOVLW   0x07        ; GUI DIA CHI TG DIEU
KHIEN NGO RA SQW/OUT CUA RTC
        CALL    WRITE_BYTE
        MOVLW       0X10                ; DU
LIEU CHON TAN SO NGO RA SQW/OUT CUA RTC LA
1Hz
        CALL    WRITE_BYTE
        CALL    STOPBIT
;=======================================
============================
START
        CALL    READ_RTC
        CALL    GIAIMA
        CALL    HIENTHI
        BTFSS PORTB,7
        GOTO  $-2
        CALL    READ_RTC
        CALL    GIAIMA
        CALL    HIENTHI
        BTFSC PORTB,7
        GOTO  $-2
        GOTO  START
;=======================================
============================
READ_RTC
        CALL    STARTBIT
        MOVLW   0XD0                    ; GUI DIA
CHI CHO PHEP RTC NHAN
        CALL    WRITE_BYTE
        MOVLW   0x00        ; gui dia chi o nho GIAY
        CALL    WRITE_BYTE
        CALL    RESTARTBIT  ; GUI BIT RESTART
        MOVLW   0xD1; GUI DIA CHI CHO PHEP RTC GUI
        CALL    WRITE_BYTE
;---------------------------------------------------------------
; Qua trinh doc du lieu tu RTC
        CALL    READ_I2C    ; Doc du lieu SECOND
        CALL    ACKBIT
        CALL    SAVE_BYTE
        MOVWF   RAM_SEC  ; Luu du lieu GIAY
        CALL    READ_I2C    ; Doc du lieu MINUTE
        CALL    ACKBIT
        CALL    SAVE_BYTE
        MOVWF   RAM_MIN  ; Luu du lieu PHUT
        CALL    READ_I2C
        CALL    NACKBIT
        CALL    SAVE_BYTE
        MOVWF   RAM_HOU ;Luu du lieu HOU
        CALL    STOPBIT
        RETURN
;=======================================
WRITE_BYTE
        BANKSEL SSPBUF
        MOVWF   SSPBUF
        CALL    WAITMSSP
        RETURN
READ_I2C
        BSF     STATUS,RP0
        BTFSC   SSPSTAT,2           ; kiem tra bit R/W, xem
qua trinh Transmit da ket thuc chua
        GOTO    $-1
        BANKSEL SSPCON2
        BSF     SSPCON2,RCEN        ; cho phep RECEIVE
Mode (I2C )
        CALL    WAITMSSP            ; Doi cho den khi I2C
thuc hien xong TUC LA NHAN DU LIEU XONG
        RETURN
SAVE_BYTE
        BANKSEL     SSPBUF
        MOVF    SSPBUF,W
        RETURN
;=======================================
STARTBIT
        BANKSEL SSPCON2
        BSF     SSPCON2,SEN     ; gui bit START
```

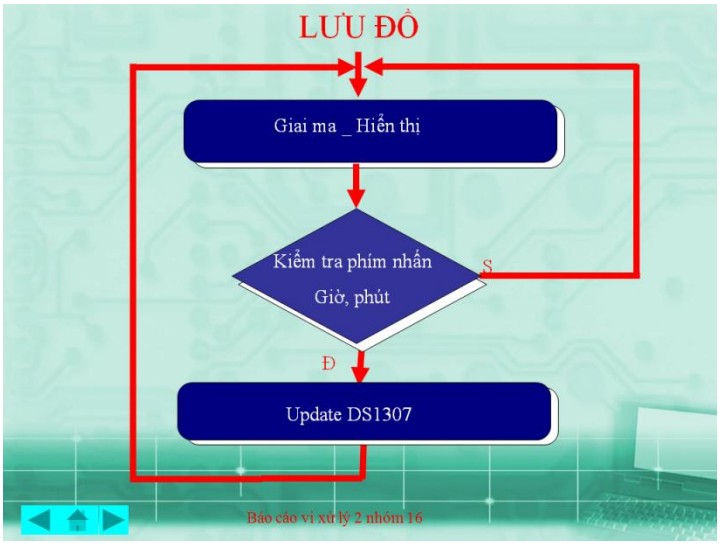
```
        CALL    WAITMSSP
        RETURN
RESTARTBIT
        BANKSEL SSPCON2
        BSF     SSPCON2,RSEN      ; gui bit RESTART
        CALL    WAITMSSP
        RETURN
ACKBIT
        BANKSEL SSPCON2
        BCF     SSPCON2,ACKDT     ; Chon gui ACK
        BSF     SSPCON2,ACKEN     ; Gui
        CALL    WAITMSSP
        RETURN
NACKBIT
        BANKSEL SSPCON2
        BSF     SSPCON2,ACKDT     ; Chon gui NACK
        BSF     SSPCON2,ACKEN     ; Gui di
        CALL    WAITMSSP
        RETURN
STOPBIT
        BANKSEL SSPCON2
        BSF     SSPCON2,PEN       ; gui bit STOP
        CALL    WAITMSSP
        RETURN
WAITMSSP
        BANKSEL PIR1
        BTFSS   PIR1,SSPIF        ; Kiem tra trang thai co
        GOTO    $-1               ; Chua hoan thanh
        BCF     PIR1,SSPIF        ; Da hoan thanh cong
viec, luc nay co the chuyen sang hoat dong moi
        RETURN
;=======================================
==============================
GIAIMA
;HEX TO BCD
        MOVLW       0X0F
        ANDWF       RAM_SEC,0           ;LAY 4BIT
THAP
        MOVWF       RAM_SEC1
        SWAPF       RAM_SEC,0
        ANDLW       0X0F
        MOVWF       RAM_SEC2
        MOVLW       0X0F
        ANDWF       RAM_MIN,0    ;LAY 4BIT THAP
        MOVWF       RAM_MIN1
        SWAPF       RAM_MIN,0
        ANDLW       0X0F
        MOVWF       RAM_MIN2

        MOVLW       0X0F
        ANDWF       RAM_HOU,0    ;LAY 4BIT THAP
        MOVWF       RAM_HOU1
        SWAPF       RAM_HOU,0
        ANDLW       0X0F
        MOVWF       RAM_HOU2;LAY MA 7 DOAN
        MOVF RAM_SEC1,0
        CALL   TABLE
        MOVWF       RAM_SEC1
        MOVF RAM_SEC2,0
        CALL   TABLE
        MOVWF       RAM_SEC2
        MOVF RAM_MIN1,0
        CALL   TABLE
        MOVWF       RAM_MIN1
        MOVF RAM_MIN2,0
        CALL   TABLE
        MOVWF       RAM_MIN2
        MOVF RAM_HOU1,0
        CALL   TABLE
        MOVWF       RAM_HOU1
        MOVF RAM_HOU2,0
        CALL   TABLE
        MOVWF       RAM_HOU2
        RETURN
;=======================================
HIENTHI
;-------HIEN THI GIAY 1
        MOVF RAM_SEC1,0
        MOVWF       PORTD
        BSF         PORTB,5
        CALL   DELAY
        BCF         PORTB,5
        CALL   DELAY
;-------HIEN THI GIAY 2
        MOVF RAM_SEC2,0
        MOVWF       PORTD
        BSF         PORTB,4
        CALL   DELAY
```

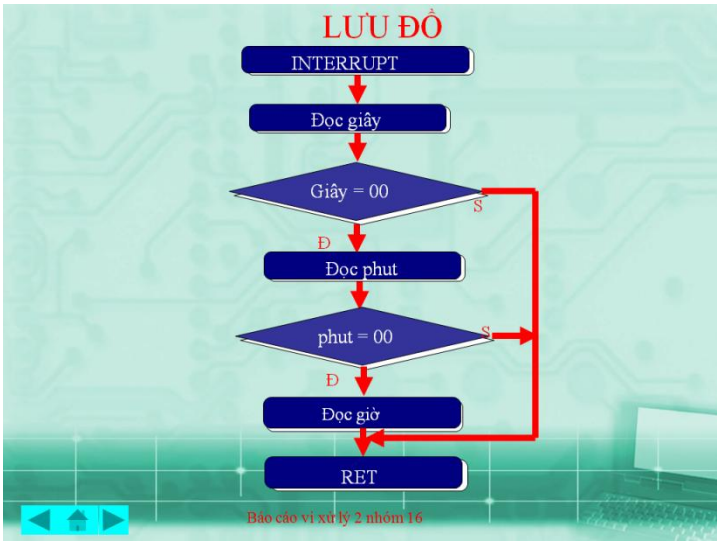
```
        BCF         PORTB,4
        CALL   DELAY
;-------HIEN THI PHUT 1
        MOVF  RAM_MIN1,0
        MOVWF       PORTD
        BSF         PORTB,3
        CALL   DELAY
        BCF         PORTB,3
        CALL   DELAY
;-------HIEN THI PHUT 2
        MOVF  RAM_MIN2,0
        MOVWF       PORTD
        BSF         PORTB,2
        CALL   DELAY
        BCF         PORTB,2
        CALL   DELAY
;-------HIEN THI GIO 1
        MOVF  RAM_HOU1,0
        MOVWF       PORTD
        BSF         PORTB,1
        CALL   DELAY
        BCF         PORTB,1
        CALL   DELAY
;-------HIEN THI GIO 2
        MOVF  RAM_HOU2,0
        MOVWF       PORTD
        BSF         PORTB,0
        CALL   DELAY
        BCF         PORTB,0
        CALL   DELAY
        RETURN
;=========================================
DELAY
        MOVLW       D'50'
        MOVWF       TEMP
        DECFSZ      TEMP
        GOTO  $-1
        RETLW       0X00
;=========================================
TABLE
        ADDWF       PCL,1
        DT
0x40,0x79,0x24,0x30,0x19,0x12,0x02,0x78,0x00,0x10
END
```

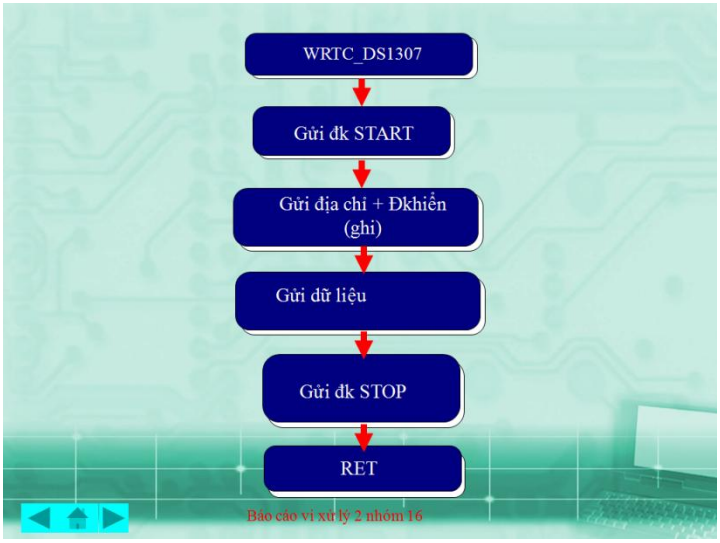
BÀI 16: Viết chương trình đồng hồ số bằng ngôn ngữ C, dùng PIC16F877A giao tiếp với RTC_DS1307. Hiển thị trên 6 led 7 đoạn











Chương trình

```
#include "16f877a.h"
#include "def_877a.h"
#fuses NOPROTECT,NOBROWNOUT,NOLVP,PUT,NOWDT
#use delay(clock=16000000)
#use i2c(Master,Fast,sda=PIN_C4,scl=PIN_C3)
#use fast_io(b)
#use fast_io(d)
#use fast_io(c)
#byte portb=0x06
#byte portd=0x08
#byte portc=0x07
//cac dinh nghia
#bit gio  = portc.1
#bit phut = portc.2
#bit led0 = portb.1
#bit led1 = portb.2
#bit led2 = portb.3
#bit led3 = portb.4
#bit led4 = portb.5
#bit led5 = portb.6
int8 sec,min,hour,data;
int8 tam=0;
int8 secdv,secc,mindv,minc,hourdv,hourc;
const int code_led[]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f};
#int_EXT
void  EXT_isr(void)
{
    i2c_start();
    i2c_write(0xD0);          // WR to RTC
    i2c_write(0x00);          // REG 0
    i2c_start();
    i2c_write(0xD1);          // RD from RTC
    sec = i2c_read(0);
    i2c_stop();
    if ( sec == 0x00)
    {
      i2c_start();
      i2c_write(0xD0);          // WR to RTC
      i2c_write(0x01);          // REG 0
```

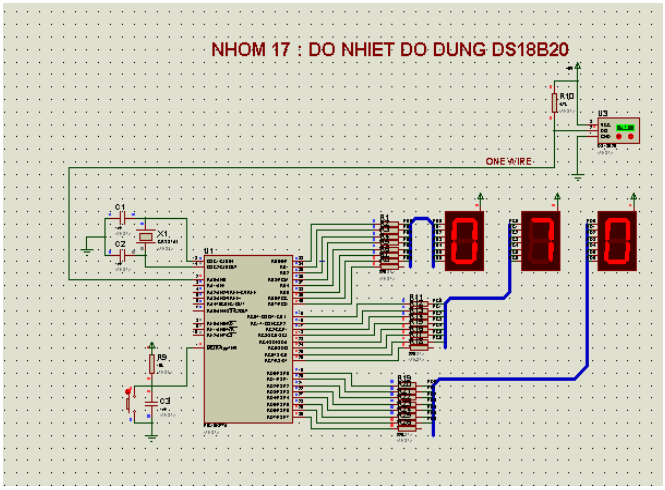
```
      i2c_start();
      i2c_write(0xD1);          // RD from RTC
      min = i2c_read(0);
      i2c_stop();
    if (min == 0x00)
      {
        i2c_start();
        i2c_write(0xD0);          // WR to RTC
        i2c_write(0x02);          // REG 0
        i2c_start();
        i2c_write(0xD1);          // RD from RTC
        hour = i2c_read(0);
        i2c_stop();
      }
    }
}

void write_RTC()
{
    i2c_start();
    i2c_write(0xD0);          // WR to RTC
    i2c_write(0x00);          // REG 0
    i2c_write(0x26);          // Start oscillator with current
"seconds value
    i2c_write(0x30);          // REG 1
    i2c_write(0x12);          // REG 2
    i2c_stop();
    i2c_start();
    i2c_write(0xD0);          // WR to RTC
    i2c_write(0x07);          // Control Register
    i2c_write(0x10);          // squarewave output pin 1Hz
    i2c_stop();
    i2c_start();
    i2c_write(0xD0);          // WR to RTC
    i2c_write(0x00);          // REG 0
    i2c_stop();
    i2c_start();
    i2c_write(0xD1);          // RD from RTC
    sec = i2c_read();
    min = i2c_read();
    hour = i2c_read();
    i2c_stop();
}
```

```
void hienthi()
{
  secdv = sec & 0x0f;
  secc = (sec & 0x70)>>4;
  portd=code_led[secdv];
  led5=0;
  delay_ms(2);
  led5=1;
  portd=code_led[secc];
  led4=0;
  delay_ms(2);
  led4=1;

  mindv = min & 0x0F;
  minc  =(min & 0x70)>>4;
  portd=code_led[mindv];
  led3=0;
  delay_ms(2);
  led3=1;
  portd=code_led[minc];
  led2=0;
  delay_ms(2);
  led2=1;

  hourdv = hour & 0x0F;
  hourc  =(hour & 0x30)>>4;
  portd=code_led[hourdv];
  led1=0;
  delay_ms(2);
  led1=1;
  portd=code_led[hourc];
  led0=0;
  delay_ms(2);
  led0=1;
}
void update_ds1307(unsigned int gtri)
{
    if(gtri == 01)
    {
    i2c_start();
    i2c_write(0xd0);
    i2c_write(0x01);          // ghi du lieu bat dau tu vi tri 01
    data=mindv+(minc<<4);
    i2c_stop();
```

```
    }
    if(gtri == 02)
    {
    i2c_start();
    i2c_write(0xd0);
    i2c_write(0x02);
    i2c_write(data);
    data=hourdv+(hourc<<4);
    i2c_write(data);
    i2c_stop();
    }
}
void set_incgio(void)
{
    hourdv++;                // tang gio
    if(hourdv == 10)          // khi hang don vi gio = 10
    {
      hourc++;
      hourdv = 0;
    }
    if((hourc == 2)&(hourdv == 4))   // khi gio = 24
    {
      hourc = 0;
      hourdv = 0;
    }
    hour = (hourc<<4)| hourdv;
 }

void set_incphut(void)
{
    mindv++;                        // tang phut
    if(mindv == 10)                  // khi phut don vi = 10
    {
       mindv = 0;
       minc++;
      if(minc == 6)                  // khi phut chuc = 6
       {
         minc = 0;
         mindv = 0;
        }
    }
    min = (minc<<4)|mindv;
 }
```

```
void main()
{
    enable_interrupts(INT_EXT);   // enable ext interrupt
    ext_int_edge(0,H_TO_L);       // set external interrupt to
falling edge
    enable_interrupts(GLOBAL);
    set_tris_d(0x00);
    set_tris_c(0x1f);
    set_tris_b(0x01);
    write_RTC();
  while (TRUE)
  {
     hienthi();
     if (phut==0)
     {
       tam++;
      }
     if (tam ==15)
       {
         set_incphut();
         update_ds1307(01);
         tam=0;
       }

     if (gio==0)
     {
        tam++;

     }
     if (tam ==15)
         {
           set_incgio();
           update_ds1307(02);
           tam=0;
         }

  }
}
```

Bài 17:giao tiếp DS18B20 và 3 led 7 đoạn .viết chương trình đo nhiệt độ băng ngôn ngữ ASSEMBLY.



Chương trình:

```
        PROCESSOR   P16F877A
        INCLUDE             <P16F877A.INC>
        __CONFIG    _CP_OFF & _WDT_OFF &
_HS_OSC & _PWRTE_ON
        #INCLUDE <P16F877A.INC>
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                CBLOCK  20H
                PDBYTE
        ; Kiem tra qua trinh Reset
;-------Lay du lieu tu DS-----------------------------------------
                BYTE0,BYTE1         ; Luu ket qua
                RBYTE0,RBYTE1       ; Xu ly ket qua
                XOAY    ; XOAY 4 lan de lay ket qua
;-------Dùng cho Delay--------------------------------------------
                DELAY_TEMP0
                DELAY_TEMP1
                DELAY_TEMP2
;-------Ghi du lieu vao DS----------------------------------------
        BYTE_8  ; Bien trung gian de ghi lenh vao DS
;-------Hien thi--------------------------------------------------
                TRAM,CHUC,DONVI
                TAM
                BDATA
                DAU
                ENDC
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                ORG     0X000
                GOTO            START
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
START
                ORG     0X0005

                BCF     STATUS,RP1
                BSF     STATUS,RP0 ; Select Bank1

                CLRF    TRISA    ; PortA as output
                CLRF    TRISB    ; PortB as output
                CLRF    TRISC    ; PortC as output
                CLRF    TRISD     ; PortD as output
                MOVLW   B'00000110'   ; Set portA  as
digital port
                MOVWF           ADCON1 ;
                BCF             STATUS,RP0          ;
Select Bank0

                CLRF    PORTB
                CLRF    PORTC
                CLRF    PORTD
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
MAIN

;-------Initate 18B20------------------------------
                CALL    RESET_18B20         ; Khoi tao
DS18B20 (Reset and wait for prensence pulse)
                BTFSS PDBYTE,0      ; PDBYTE=1 ->
Reset thành công
                GOTO  MAIN          ; PDBYTE=0 ->
Quay lai tu dau
;-------SKIP ROM----------------------------------------


        MOVLW       0CCH    ; Truy cap thang den
DS18B20
        MOVWF       BYTE_8; BYTE_8 làm trung gian
de dich byte 0CCH vào RA0
        CALL    WRITE_18B20         ;
;-------CONVERT-----------------------------
        MOVLW       44H     ; Đo và chuyen doi giá tri
nhiet do thành so
        MOVWF       BYTE_8; BYTE_8 làm trung gian
de dich byte 44H vào RA0
        CALL    WRITE_18B20 ;
        CALL    DELAY_762MS; Cho chuyen doi xong

;-------Initate 18B20----------------------
        CALL    RESET_18B20         ; Khoi tao
DS18B20 (Reset and wait for prensence pulse)
        BTFSS PDBYTE,0              ; PDBYTE=1 ->
Reset thành công
        GOTO  MAIN  ; PDBYTE=0 -> Quay lai tu dau
;-------SKIP ROM-------------------------------------------------
        MOVLW  0CCH   ; Truy cap thang den DS18B20
        MOVWF  BYTE_8 ; BYTE_8 làm trung gian de
dich byte 0CCH vào RA0
        CALL    WRITE_18B20         ;
;-------READ SCRATCHPAD-----------------------------
```

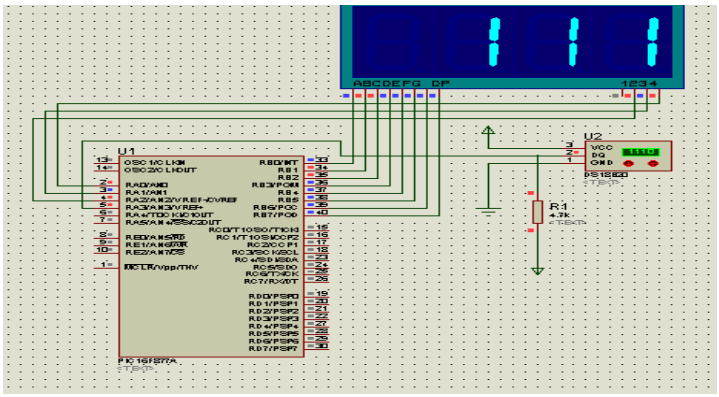
```
        MOVLW  0BEH  ; Đoc noi dung bo nho nháp
        MOVWF        BYTE_8 ; BYTE_8 làm trung gian de dich byte 0BEH vào RA0
        CALL   WRITE_18B20          ;
;-------Lay du lieu---------------------------------
        CALL   READ_18B20  ; Read 2 byte BDATA (Luu vao Byte0 & Byte1)
;-------Initate 18B20-------------------------
 CALL           RESET_18B20          ; Reset Ds18B20 ve trang thái nghi (bo 7 bit còn lai)
        BTFSS PDBYTE,0; PDBYTE=1 -> Reset thành công
        GOTO  MAIN  ; PDBYTE=0 -> Quay lai tu dau
;-------Xu ly ket qua---------------------------------
        MOVF  BYTE0,W       ; RBYTTE0 = BYTE0 , BYTE0 BYTE1 co dinh
        MOVWF  RBYTE0 ; RBYTTE0 = BYTE0
        MOVF  BYTE1,W       ; RBYTTE1 = BYTE1
        MOVWF  RBYTE1; RBYTTE1 = BYTE1
        CALL   LOC_KETQUA  ; Loc lay ket qua, luu vào bien BDATA
;-------Hien thi-------------------------
                CALL   HEXTOBCD
                CALL   HIENTHI
                 GOTO          MAIN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
RESET_18B20
                CLRF   PDBYTE        ; Xóa PDBYTE (bit kiem tra Presence Pulse)
               CALL    OW_LOW
                CALL   DELAY_485    ; 480us MINIMUM
        CALL   OW_HIGH_Z            ; Release the BUS
        CALL   DELAY_62
        BTFSS PORTA,0; RA0=0 -> Có xung Presence
        INCF    PDBYTE
        CALL   DELAY_242
   RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
OW_HIGH_Z
        BSF    STATUS,RP0 ; Chon Bank1 cua bo nho du lieu
        BSF    TRISA,0  ; RA0 as OUTPUT -> DQ o trang thái HIGH_Z
        BCF    STATUS,RP0    ; Chon BANK0
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
OW_LOW
        BSF    STATUS,RP0  ; Chon Bank1 cua bo nho du lieu
         BCF    TRISA,0          ; RA0 as OUTPUT
         BCF    STATUS,RP0   ; Chon BANK0
         BCF    PORTA,0                 ; RA0=0

        RETURN                                  ;
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
WRITE_18B20           ; 3 giai doan
;------Dich 8 bit BYTE_8 vào RA0 de thuc hien lenh---------
        MOVLW         08H     ; Dich 8 bit vào RA0
        MOVWF         TAM             ;
        BCF     STATUS,C  ; Co C lam cau noi de dich
LB_WRITE
;-------Giai doan 1: Khoi tao----------------------
        CALL   OW_LOW       ; Kéo QD xuong muc thap
        CALL   DELAY_17     ; 15us MINIMUM
;-------Giai doan 2: Ghi du lieu-----------------------------
        RRF  BYTE_8,1; Dich 1bit LSB cua BYTE_8 vào C
        BTFSS  STATUS,C; C=0 thì WRITE_0, C=1 thì WRITE_1
        GOTO  $+2
        BSF             PORTA,0; Neu C=1 -> RA0=1
        CALL   DELAY_47      ; Delay 47 cho ghi xong
;-------Giai doan 3: Release the Bus-----------------------
        BANKSEL TRISA
                BSF     TRISA,0   ; RA0 as INPUT
                BANKSEL PORTA   ; 1us
                NOP
        ; Cho thêm 1us
;-------Dich bit tiep theo---------------------------------
        DECFSZ  TAM,1
                GOTO            LB_WRITE
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
READ_18B20                      ; 3 giai doan
                MOVLW         D'16'    ; Đoc 16 bit tu RA0
                MOVWF         TAM
LB_READ
;-------Giai doan 1: Khoi tao-----------------------------------
```

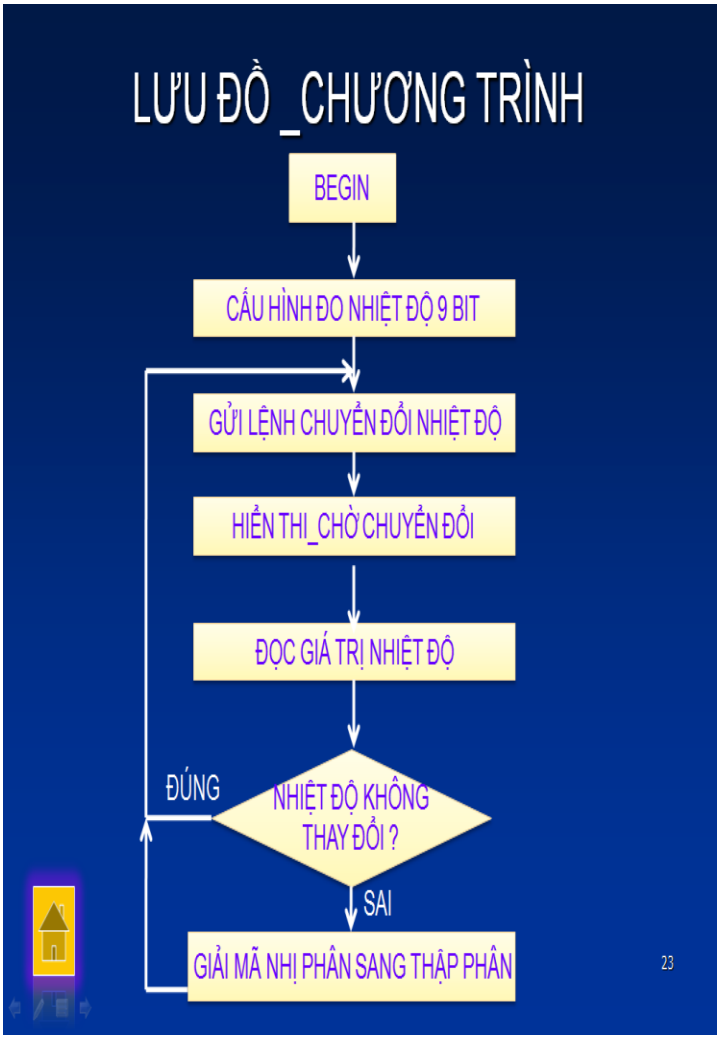
```
            CALL  OW_LOW
            NOP
;-------Giai doan 2: Release the Bus----------
            CALL  OW_HIGH_Z   ; Tu luc RA0 o
trang thai HIGH_Z den khi ket thuc ctr con la 3us
            CALL  DELAY_11; Cho them 11us de lay
du lieu trong khoang 14us cua TimeSlot
;-------Giai doan 2: Lay du lieu-----------------------------
            BTFSS       PORTA,0; RA0=1 thì C=1
            GOTO        $+3; RA0=0 thì C giu
nguyên (=0)
            BSF   STATUS,C
            GOTO        $+2
            BCF   STATUS,C  ; C=0 khi DQ=0
             RRF   BYTE1,1  ; Dich C vào BYTE1 và
BYTE0 (16 lan)
            RRF   BYTE0,1  ; Dich C vào BYTE1 và
BYTE0 (16 lan)
            CALL  DELAY_47 ; Cho het TimeSlot
;-------Đoc bit tiep theo--------------------
            DECFSZ      TAM,1
            GOTO  LB_READ
            RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
LOC_KETQUA
      BTFSS BYTE1,7  ; BYTE1.7 = 1 thì là so âm
      GOTO  EXIT   ; BYTE1.7 = 0 thì là so duong
      COMF  RBYTE1,F          ; Bù 2 RBYTE1
      COMF  RBYTE0,F          ; Bù 2 RBYTE0
      MOVLW       0X01        ; Cong thêm 1
      ADDWF       RBYTE0,F          ;
;---------------------------------------------------------------------
      BTFSS STATUS,C      ; Neu C=1 thì phai cong 1
cho RBYTE1 do phép toán truoc bi tràn
      GOTO  EXIT; (Chi dành cho truong hop hien thi so
thap phân)
      MOVLW       0X01
      ADDWF       RBYTE1,F
;---------------------------------------------------------------------
EXIT
      MOVLW       D'4'      ; Ket qua nguyên lay trong
BYTE0<4:7> và BYTE1<0:3>
      MOVWF       XOAY  ; Do dó phai xoay 4 lan
LOOP
      RRF         RBYTE1,1
      RRF         RBYTE0,1
      DECFSZ   XOAY,1; Xoay du 4 lan? (Bo 4 bit thap)
      GOTO  LOOP
       MOVF RBYTE0,0  ; Ket qua nam trong BYTE0
      MOVWF       BDATA; Đua ket qua vào BDATA
      BCF   BDATA,7  ; Chi lay 7 bit du lieu

            RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
HEXTOBCD
      CLRF  TRAM        ; TRAM=0
      CLRF  CHUC        ; CHUC=0
      CLRF  DONVI       ; DON VI=0
XET1
      MOVLW       .100    ; W=100
      SUBWF       BDATA,0; XET BDATA-W
      BTFSS STATUS,C ; BANG CACH XET CO C=1 -
> BDATA>100
      GOTO  XET2  ; BDATA<100 THI DI DEN XET2
      MOVLW .100; CON NEUBDATA >= 100 THI
      SUBWF       BDATA,1; BDATA=DATA-W (VOI
W=100)
      INCF  TRAM,1; TRAM=TRAM+1 (DATA SO
LAN TAM LON HON 100)
            GOTO  XET1                ; TIEP
TUC QUAY LAI XET1 NHU TREN
XET2
            MOVLW       .10
      ; W=10
            SUBWF       BDATA,W
      ; XETBDATA-W
            BTFSS STATUS,C            ; BANG
CACH XET CO C=1 -> BDATA>10
            GOTO  THOATXET            ; NEU
KHONG TRU HET TUC LA TAM<10 THI DI DEN
THOATXET
            MOVLW       .10
      ; CON NEUBDATA >= 100 THI
            SUBWF       BDATA,1
      ; BDATA=DATA-W (VOI W=10)
            INCF  CHUC,1                    ;
CHUC=CHUC+1 (DATA SO LAN TAM LON HON 10)
```

```
        GOTO  XET2   ; TIEP TUC QUAY LAI XET2 NHU TREN
THOATXET
        MOVF BDATA,0        ; SAU CUNG THIBDATA<10 THI TA CHUA VAO DONVI
        MOVWF       DONVI         ; DONVI=W
        RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
HIENTHI
        BTFSS BYTE1,7; BYTE1.7 =1 -> So âm
        GOTO  DUONG; BYTE1.7 =0 -> So duong
        MOVLW 0XBF; Mã dau tru (common anode)
        MOVWF       PORTB         ; Đua vào PORTB
        GOTO  AM       ; Không hien thi TRAM
DUONG
        MOVF  TRAM,W                ; W=TRAM
        CALL   TABLE ; Giai mã TRAM sang LED 7 doan
        MOVWF  PORTB    ; Xuat ra PORTB
AM
        MOVF  CHUC,W        ; W=CHUC
        CALL   TABLE ; Giai mã CHUC sang LED 7 doan
        MOVWF       PORTC   ; Xuat ra PORTC

        MOVF  DONVI,W       ; W=DONVI
        CALL   TABLE; Giai mã DONVI sang LED 7 doan
        MOVWF  PORTD        ; Xuat ra PORTD
        RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_242
                MOVLW       D'79'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_62
                MOVLW       D'19'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_47
                MOVLW       D'14'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_17
                MOVLW       D'4'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_11
                MOVLW       D'2'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_485
                MOVLW       D'160'
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY_762MS
        ;(762.242MS)
                MOVLW       D'50'
                MOVWF       DELAY_TEMP2
                MOVWF       DELAY_TEMP1
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                DECFSZ      DELAY_TEMP1,1
                GOTO  $-4
                DECFSZ      DELAY_TEMP2,1
                GOTO  $-7
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DELAY300MS
                MOVLW       D'40'
                MOVWF       DELAY_TEMP2
                MOVWF       DELAY_TEMP1
```

```
                MOVWF       DELAY_TEMP0
                DECFSZ      DELAY_TEMP0,1
                GOTO  $-1
                DECFSZ      DELAY_TEMP1,1
                GOTO  $-4
                DECFSZ      DELAY_TEMP2,1
                GOTO  $-7
                RETURN
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
TABLE
        ADDWF       PCL,1
        DT
        0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,
90H     ;common anode
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 END
```

Bài 18: Dùng PIC16F877A giao tiếp với cảm biến nhiệt độ DS18B20 đo nhiệt độ hiển thị ra Led 7 đoạn , dùng C



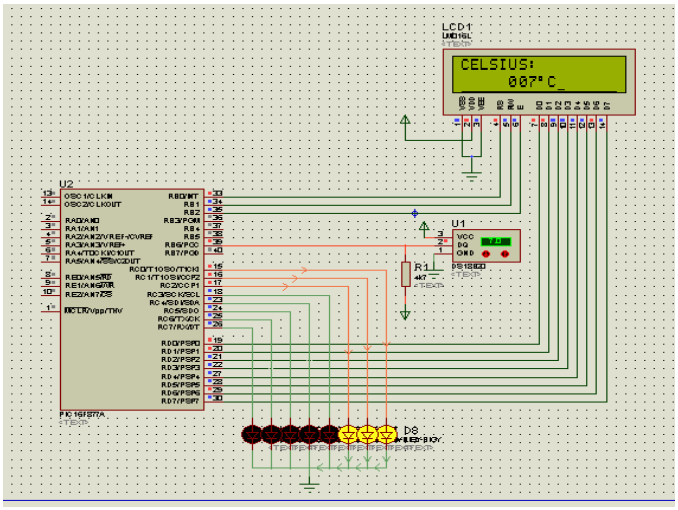


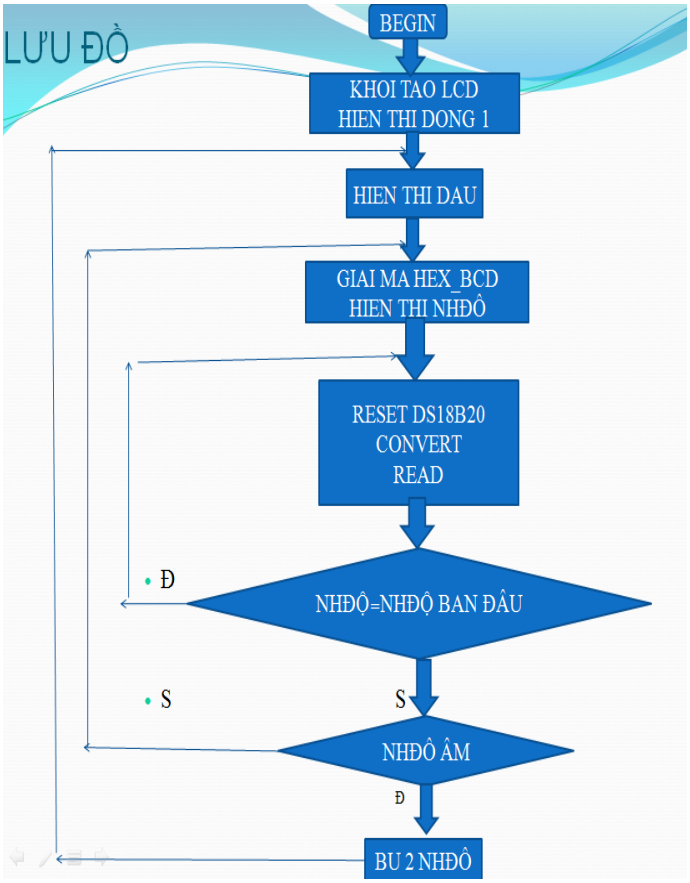
Chương trình

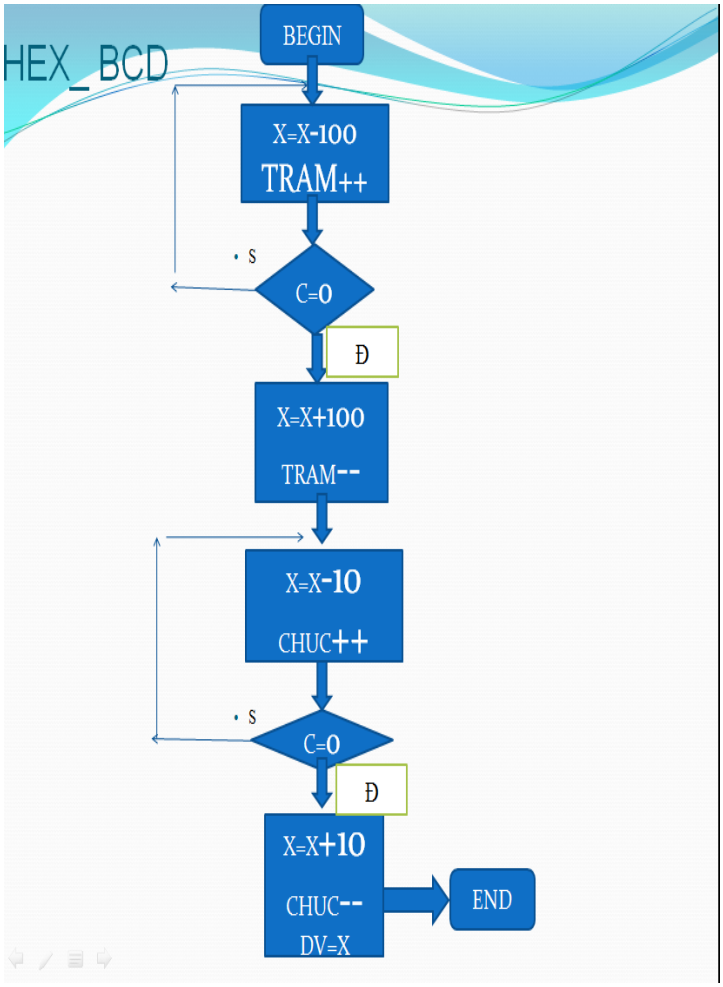
```
 #include <16F877.h>
#fuses HS,NOPROTECT,NOLVP
#use delay(clock=20000000)
#define TOUCH_PIN pin_a3
#include <touch.c>
void hienthi(int8 a,int8 b,int8 c);
int8 const seg7[11] =
{0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,0x40
};
//chuong trinh chinh
=====================
void main() {

        int8 tam1=0,tam2,ss,i;
        int8 led1,led2,led3;
        int16 nd;
        //cài đặt chuyển đổi nhiệt độ 9 bit
        ============================
        touch_present();
        touch_write_byte(0xcc);     // Phat lenh skip ROM
        touch_write_byte(0x4e);     // Phat lenh ghi 3byte
du lieu to Scratchpad
        touch_write_byte(0x0);      // ghi 2 byte rong
        touch_write_byte(0x0);      //
        touch_write_byte(0x1f);     //Cau hinh cho thanh
ghi nhiet do(do phan giai 9 bit)

        touch_present();
        touch_write_byte(0xCC);     // skip ROM
        touch_write_byte(0x48);     // Sao chep Scratchpad
vao Eeprom
/===vòng lặp chương trình chính=======/
   while (1) {
        do {
                if(touch_present())
                {
                touch_write_byte(0xCC);
                touch_write_byte (0x44);  //xuat lenh
chuyen doi nhiet do
                for(i=0;i<7;++i) {
                        hienthi(led1,led2,led3);
                                        }
                touch_present();
                touch_write_byte(0xCC);
                touch_write_byte (0xBE);
        //lenh doc bo nho nhap
                tam1 = touch_read_byte();
        //doc 2 byte nhiet do
                tam2 = touch_read_byte();
                }
           }
         while (tam1==ss);                  //so sanh gia tri
nhiet do doc duoc voi gia tri truoc do
        ss=tam1;
/==chuyen doi nhiet do sang so thap phan====
  nd=make16(tam2,tam1);
  nd=nd>>4;             //dich phai 4 bit
  nd=nd&0x0ff;          //lay 8 bit thap
  if(bit_test(nd,7))    //nếu giá tri nhiệt độ là âm
  {
    led3=10;            //led3 hiển thị dấu "-"
    nd=~(--nd);         //lấy bù 2 giá tri nhiệt độ
    nd=nd&0x0ff;        //xóa byte cao
  }
  else                  //nếu giá trị nhệt độ dương
  {
    led3=nd/100;        //led3 hiển thị số hàng trăm
    nd=nd%100;
  }
    led2=nd/10;         //led2 hiển thị số hàng chục
    led1=nd%10;         //led1 hiển thị số hàng đơn vị
   }
   }


 //chuong trinh hien thi_quet led===
 void hienthi(int8 a,int8 b,int8 c)
{
  output_b(seg7[a]);
  output_low(pin_a0);
  delay_ms(6);
  output_high(pin_a0);
    output_b(seg7[b]);
  output_low(pin_a1);
  delay_ms(6);
  output_high(pin_a1);
```

```
    output_b(seg7[c]);
output_low(pin_a2);
 delay_ms(6);
 output_high(pin_a2);
  }
```

Bài 19: Sử dụng PIC 16F877A đọc nhiệt độ từ cảm biến DS 18B20, hiển thị LCD HD 44780. Ngôn ngữ ASSEMBLY







Chương trình:

```
processor       p16f877a
include         <p16f877a.inc>
__CONFIG        _CP_OFF & _PWRTE_OFF & _WDT_OFF & _HS_OSC & _LVP_OFF

        constant        SKIPROM=0xCC
        constant        CONVERT=0x44
        constant        READ=0xBE

        constant        MODE=0x38
        constant        DISPLAY=0xF
        constant        CEL=0XDF
        constant        DQ=6
        constant        RS=0
        constant        RW=1
        constant        E=2

        LCD             EQU             PORTD
cblock  0x020
        COUNT
        IOBYTE
        BYTE0
        BYTE1
        NEWTEMPER
        TEMPER
        TMP0
        TRAM
        CHUC
        DV
        TAM
        TAM2
        count
        TEMP1
endc
        org             0x000
        bcf             STATUS,RP1
        bsf             STATUS,RP0
        clrf            TRISD
        clrf            TRISC
        bcf             TRISB,RS
        bcf             TRISB,RW
        bcf             TRISB,E
        bcf             STATUS,RP0
;-----------------------------------------------------------
        BCF             PORTB,RS
        BCF             PORTB,RW
        MOVLW           MODE
        CALL   EXEC_DLAY
        MOVLW           DISPLAY
        CALL   EXEC_DLAY

        BSF             PORTB,RS
        MOVLW           D'0'
        MOVWF           TAM2
LOOP    MOVF  TAM2,0
        CALL   TABLE2
        CALL    EXEC_DLAY
        INCF    TAM2,1
        MOVF  TAM2,0
        SUBLW           D'8'
        BTFSS STATUS,Z
```

```
            GOTO  LOOP
            GOTO  DL
DLSIG BCF   PORTB,RS
            MOVLW       0xC4
            CALL  EXEC_DLAY
            BSF         PORTB,RS
            MOVLW       D'10'
            CALL  TABLE1
            CALL        EXEC_DLAY
            GOTO  DL2
DL          BCF         PORTB,RS
            MOVLW       0xC4
            call  EXEC_DLAY
            BSF         PORTB,RS
            MOVLW       ' '
            call  EXEC_DLAY
DL2         CALL  DECODE


MAIN  CALL  RESET
            movlw SKIPROM

            CALL  WRITE
            movlw CONVERT
            CALL  WRITE
            MOVLW       D'50'
            MOVWF       TEMP1
AA          CALL    WAIT5000U
            DECFSZ      TEMP1,F
            GOTO        AA

            CALL  RESET
            movlw SKIPROM
            CALL  WRITE
            movlw READ
            CALL  WRITE
        CALL        READDS
            MOVF  IOBYTE,0
            MOVWF       BYTE0
            CALL  READDS
            MOVF  IOBYTE,0
            MOVWF       BYTE1
            CALL  RESET
            MOVLW       0xF0
            ANDWF       BYTE0,1
            SWAPF       BYTE0,0
            MOVWF       NEWTEMPER
            MOVLW       0X0F
            ANDWF       BYTE1,1
            SWAPF       BYTE1,0
            IORWF NEWTEMPER,1
            MOVF  NEWTEMPER,0
            MOVWF       PORTC
            XORWF       TEMPER,0
            BTFSC STATUS,Z
            GOTO  MAIN
            MOVF  NEWTEMPER,0
            MOVWF       TEMPER
            MOVWF       TAM
            RLF         TAM,0
            BTFSC STATUS,C
            GOTO  SIG
            GOTO  DL
SIG         COMF  TAM,F
            MOVLW       D'1'
            ADDWF       TAM,1
            GOTO  DLSIG
;-------------------------------------------------------
OW_HIZ:MACRO
    BSF     STATUS,RP0
    BSF     TRISB,DQ
    BCF     STATUS,RP0
    ENDM
; -------------------------------------------------------
OW_LO:MACRO
    BSF     STATUS,RP0
    BCF     TRISB,DQ
    BCF     STATUS,RP0
    BCF     PORTB,DQ
    ENDM
; -------------------------------------------------------
 WAIT:MACRO TIME
      MOVLW (TIME/5) - 1        ;1µs to process
      MOVWF  TMP0               ;1µs to process
      CALL   WAIT5U             ;2µs to process
      ENDM
 WAIT5U:
```

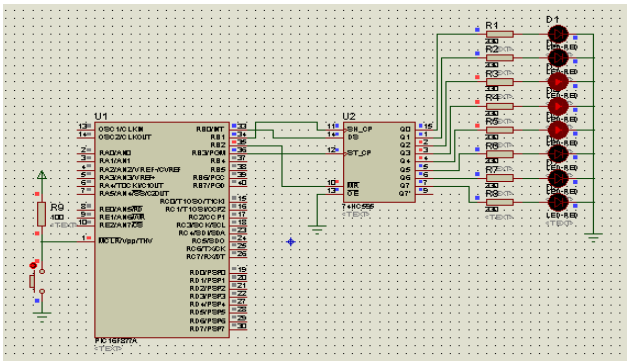
```
      NOP                  ;1µs to process
      NOP                  ;1µs to process
      DECFSZ   TMP0,F      ;1µs if not zero or
      GOTO     WAIT5U      ;2µs to process
      RETLW    0           ;2µs to process
WAIT5000U:
         WAIT  .1000
         WAIT  .1000
         WAIT  .1000
         WAIT  .1000
         WAIT  .1000
         RETLW 0
; -------------------------------------------------------
RESET
TRY      OW_LO
         WAIT  .500
         OW_HIZ
         WAIT  .60
         BTFSC PORTB,DQ
         GOTO  TRY
         WAIT  .240
         BTFSS PORTB,DQ
         GOTO  TRY
         WAIT  .180
   RETLW 0
; -------------------------------------------------------
READDS

   MOVLW    d'8'
   MOVWF    COUNT
RER  OW_LO

   NOP
   OW_HIZ
      NOP
      NOP

   BTFSS    PORTB,DQ
   BCF      STATUS,C
   BTFSC    PORTB,DQ
   BSF      STATUS,C
   RRF      IOBYTE,1
      WAIT  .80
   DECFSZ   COUNT,1
      GOTO  RER
      RETLW 0
; -------------------------------------------------------
WRITE
      MOVWF    IOBYTE
      MOVLW    D'8'
      MOVWF    COUNT
GHI
      OW_LO
      WAIT     .10
      BTFSC    IOBYTE,0
               GOTO  MUCCAO
      GOTO     MUCTHAP
MUCCAO
      OW_HIZ
      WAIT     .50
               RRF   IOBYTE,1
      DECFSZ   COUNT,1
      GOTO     GHI
      RETURN
MUCTHAP
               BCF   PORTB,6
      WAIT     .50
               OW_HIZ
               RRF   IOBYTE,1
      DECFSZ   COUNT,1
      GOTO     GHI
      RETURN
;-------------------------------------------------------------
DECODE   MOVLW   D'0'
         MOVWF   TRAM
         MOVWF   CHUC
         MOVWF   DV
         MOVLW   d'100'
LB1      SUBWF   TAM,1
         INCF    TRAM,1
         BTFSC STATUS,C
         GOTO  LB1
         DECF  TRAM,1
         ADDWF   TAM,1
         MOVLW   d'10'
LB2      SUBWF   TAM,1
         INCF    CHUC,1
         BTFSC STATUS,C
```

```
                GOTO  LB2
                DECF   CHUC,1
                ADDWF          TAM,0
                MOVWF          DV
                movf    TRAM,0
                call    TABLE1
                call       EXEC_DLAY
                movf    CHUC,0
                call       TABLE1
                call    EXEC_DLAY
                movf    DV,0
                call       TABLE1
                call       EXEC_DLAY
                MOVLW          CEL
                call       EXEC_DLAY
                MOVLW          'C'
                call       EXEC_DLAY
                RETLW 0
EXEC_DLAY               movwf  LCD
                        bsf                PORTB,E
                        bcf                PORTB,E
                        movlw  d'7'
                        movwf  count
dela1                   WAIT   .200
                               decfsz   count,1
                               goto     dela1
                               return
;-------------------------------------------------------------------
TABLE2          ADDWF          PCL,1
        DT      "CELSIUS:"

TABLE1          addwf   PCL,1

        DT      "0123456789-"
end
```

Bài 20: pic 16f877a giao tiếp với ic 74HC595

A. SỬ DỤNG NGÔN NGỮ ASSEMBLY

1.test ic 74CH595:

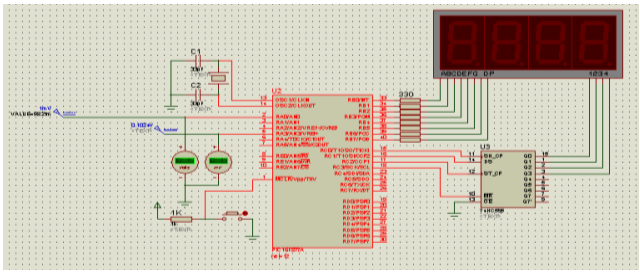


title"giao tiep voi 74HC595"

```
 ;-------------------------------------------------
processor   16f877a
 include  <p16f877a.inc>
 ;--------------------------------------
; khai bao bien
;---------------------------------------------------------------------

sendreg  EQU    0X20 ;chua du lieu can xuat ra ic 74hc595
count    EQU    0X21 ;can de dem so bit du lieu can xuat ra
 tam    equ       0x22
;---------------------------------------------------------------------
;dinh nghia phan cung
;---------------------------------------------------------------------
 #define dataa PORTB,0
#define clock PORTB,1
#define clear PORTB,2
#define latch PORTB,3

 ;---------------------------------------------------------------------
; Chuong trinh chinh
;---------------------------------------------------------------------
    ORG  0x000
    GOTO  start

  start         ;chuong trinh chinh
    BCF  STATUS,RP1
    BCF  STATUS,RP0   ;chon bank 0
    CLRF  PORTB
    BSF  STATUS,RP0   ;chon bank 1
```

```
    MOVLW  0xF0      ;cac pin RB3,RB0 la output
    MOVWF  TRISB     ;cac pin RB7,RB3 la input

    BCF  STATUS,RP0   ;chon bank 0

    MOVLW 0x04
    MOVWF PORTB      ;dua sclr\ len mua logic cao

    BCF    clear     ;reset du lieu trong 74hc595

    NOP              ;clear tac dong canh xuong

    BSF    clear
         movlw  0xff
         movf    tam,1
lb
    MOVf  tam,0
    CALL    serout
    BSF    latch
    NOP
    BCF    latch
decf    tam
    GOTO   lb
serout
    MOVWF    sendreg
    MOVLW    0x08
    MOVWF  count
 testbit
    BCF    dataa
    BTFSC  sendreg,7
    BSF    dataa
    BSF    clock
    NOP

    BCF     clock
    RLF     sendreg,0
    MOVWF   sendreg
    DECFSZ  count,1
    GOTO    testbit

    RETURN
END
```

2. do nhiet do dung adc quet 74595



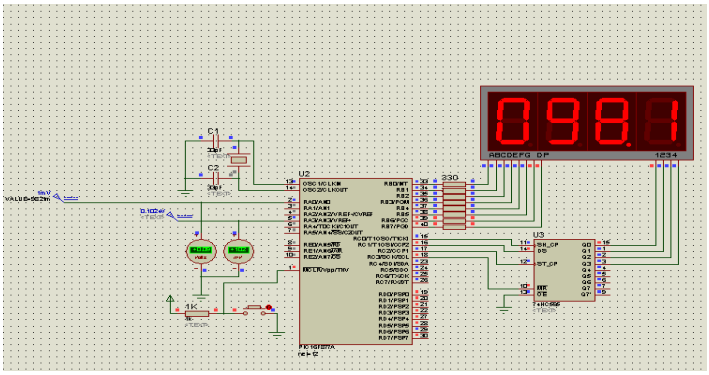
```
;=====================================
        TITLE           "ADC CONVERTER"
        PROCESSOR   PIC16F877A
        INCLUDE<P16F877A.INC>
        __CONFIG _CP_OFF & _WDT_OFF &
_BODEN_OFF & _PWRTE_ON & _HS_OSC &
_WRT_OFF & _CPD_OFF &_LVP_OFF
;=====================================
;KHAI BAO BIEN
;=====================================
CBLOCK      0X020
TR
CH
DV
TP

CAO
THAP

COUNT
TIME
ENDC
;=====================================
;BAT DAU CHUONG TRINH CHINH
;=====================================
        ORG    0X0000
        START
;=====================================
;KHOI TAO CAC PORT
;=====================================
        BANKSEL     TRISB
        CLRF   TRISB  ;KHOI TAO PORT B,C
        CLRF   TRISC
        CLRF   TRISD
;=====================================
;KHOI TAO ADC
```

```
;===================================
MOVLW  B'10000101'; KHOI TAO THANH GHI ADCON1
MOVWF ADCON1;CANH LE PHAI, 2 NGO IN ANALOG,
VREF+=AN3
;===================================
;START GAME ^_________^
;===================================
MAIN
        CALL   DOC_ADC
        CALL   BYTE_BCD
        CALL   BCD_7DOAN
        CALL   HIENTHI
        GOTO  MAIN
;=======================================
;CHUONG TRINH CON DOC ADC
;=======================================
DOC_ADC
        BANKSEL      ADCON0          ;CHUYEN
DEN BANK 0 DEN SU DUNG THANH GHI ADCON0
        MOVLW        B'10000001'     ;CAP
NGUON CHUYEN DOI ADC,FOSC/32,IN ANALOG=AN0
        MOVWF        ADCON0
        MOVLW        .20
        MOVWF        TIME
TG_TRE
        DECFSZ       TIME,F
        GOTO  TG_TRE
        BSF          ADCON0,2
GODONE
        BTFSC ADCON0,2      ;BAT DAU CHUYEN DOI ADC
        GOTO  GODON   ;CHO CHUYEN ADC KET THUC
        MOVF  ADRESH,W
        MOVWF        CAO
        ;LUU DU LIEU CHUYEN DOI VAO BIEN ADC
        BANKSEL      ADRESL
        MOVF  ADRESL,W
        BANKSEL      THAP
        MOVWF        THAP
        RETURN
;=======================================
;CHUONG TRINH CON CHUYEN DU LIEU ADC
SANG SO BCD
;=======================================
BYTE_BCD
        CLRF   TR
        CLRF   CH
        CLRF   DV
SCH
        MOVLW        .100
        SUBWF        THAP,F
        BTFSS STATUS,C
        GOTO  RDV
        INCF   CH
        GOTO  SCH
RDV
        ADDWF        THAP,F
SDV
        MOVLW        .10
        SUBWF        THAP,F
        BTFSS STATUS,C
        GOTO  RTP
        INCF   DV
        GOTO  SDV
RTP
        ADDWF        THAP,W
        MOVWF        TP


        MOVF  CAO,W
        XORLW        0X01
        BTFSS STATUS,Z
        GOTO  XET10
        MOVLW        .2
        ADDWF        CH,F
        MOVLW        .5
        ADDWF        DV,F
        MOVLW        .6
        ADDWF        TP,F
XET10
        MOVF  CAO,W
        XORLW        0X02
        BTFSS STATUS,Z
        GOTO  XET11
        MOVLW        .5
        ADDWF        CH,F
        MOVLW        .1
        ADDWF        DV,F
        MOVLW        .2
        ADDWF        TP,F
```

```
XET11
        MOVF CAO,W
        XORLW       0X03
        BTFSS STATUS,Z
        GOTO NEXT
        MOVLW       .7
        ADDWF       CH,F
        MOVLW       .6
        ADDWF       DV,F
        MOVLW       .8
        ADDWF       TP,F

NEXT
        MOVLW       .10
        SUBWF       TP,F
        BTFSS STATUS,C
        GOTO PTP
        INCF   DV
        GOTO XETDV
PTP
        ADDWF       TP,F
XETDV
        MOVLW       .10
        SUBWF       DV,F
        BTFSS STATUS,C
        GOTO PDV
        INCF   CH
        GOTO XETCH
PDV
        ADDWF       DV,F
XETCH
        MOVLW       .10
        SUBWF       CH,F
        BTFSS STATUS,Z
        GOTO PCH
        INCF   TR
        GOTO THOAT
PCH
        ADDWF       CH,F
THOAT
RETURN
;====================================
;CHUONG TRINH CON CHUYEN MA BCD SANG 7 DOAN
;====================================
BCD_7DOAN
        MOVF TR,W
        CALL  TABLE
        MOVWF       TR

        MOVF CH,W
        CALL  TABLE
        MOVWF       CH

        MOVF DV,W
        CALL  TABLE
        MOVWF       DV
        MOVLW       0X80
        XORWF       DV,F

        MOVF TP,W
        CALL  TABLE
        MOVWF       TP
RETURN
;====================================
;CHUONG TRINH CON HIEN THI
;====================================
HIENTHI
        MOVLW       0X0F
        MOVWF       COUNT
XUAT

        MOVF TR,W   ;XUAT GIA TRI HANG TRAM
        MOVWF       PORTB
        CALL  SH1
        CALL  DELAY
        MOVLW       0XFF
        MOVWF       PORTB

        MOVF CH,W   ;XUAT GIA TRI HANG CHUC
        MOVWF       PORTB
        CALL  SH0
        CALL  DELAY
        MOVLW       0XFF
        MOVWF       PORTB

        MOVF DV,W   ;XUAT GIA TRI DON VI
        MOVWF       PORTB
```

```
        CALL   SH0
        CALL   DELAY
        MOVLW       0XFF
        MOVWF       PORTB

        MOVF  TP,W   ;XUAT GIA TRI THAP PHAN
        MOVWF       PORTB
        CALL   SH0
        CALL   DELAY
        MOVLW       0XFF
        MOVWF       PORTB
        CALL   RST

        DECFSZ      COUNT,F
        GOTO  XUAT
        RETURN
;====================================
;RESET DE CHO DAU RA QUET VE 0 CHUAN BI
CHO CHU KY QUET MOI
;PLEASE READ DATASHEET FOR DETAIL
OPERATION ^_________^
;====================================
RST
        BCF         PORTC,3
        BSF         PORTC,2
        BCF         PORTC,2
        BSF         PORTC,3
RETURN
;====================================
;DICH RA BIT 1 O VI TRI BIT DAU
;READ DATASHEET FOR DETAIL OPERATION
AGAIN ^_____________^ LOLOLOLOOLOLO
;====================================
SH1
        BSF         PORTC,1
        BSF         PORTC,0
        BCF         PORTC,0
        BSF         PORTC,2
        BCF         PORTC,2
RETURN
;====================================
;DICH RA BIT 0 DE DAY 1 CHAY
;SAME HERE,LOLOLOLOOLLOLO .. READ THE
DATASHEET ^_____________^
;====================================
SH0
        BCF         PORTC,1
        BSF         PORTC,0
        BCF         PORTC,0
        BSF         PORTC,2
        BCF         PORTC,2
RETURN
;====================================
;CHUONG TRINH CON DELAY
;====================================
DELAY
        MOVLW       0XFF
        MOVWF       TIME
        DECFSZ      TIME
        GOTO  $-1
        RETURN
;====================================
;THU VIEN MA 7 DOAN
;====================================
TABLE
        ADDWF               PCL,1
        DT
        0XC0,0XF9,0XA4,0XB0,0X99,0X92,0X82,0XF8,0
X80,0X90
END
```

B.SỬ DỤNG NGÔN NGỮ C



```
#INCLUDE <16F877A.H>
#INCLUDE <KhaiBaoTGhi_16F877A.H>
#FUSES NOWDT,NOPROTECT,NOLVP,PUT,HS
#USE DELAY(CLOCK=20000000)
#DEFINE SCK RC0
```

```
#DEFINE SER RC1
#DEFINE RCK RC2
#DEFINE RESET  RC3

UNSIGNED INT16   TAM=0;
INT8  TP,DV,CHUC,TRAM;
CONST UNSIGNED
MALED[10]={0XC0,0XF9,0XA4,0XB0,0X99,0X9
2,0X82,0XF8,0X80,0X90};

VOID HEX_BCD()
{
        TAM=(INT16) ADRESH;
        TAM=(INT16)(TAM<<2)+(INT16)(ADRE
SL>>6);
        TRAM= MALED[(INT8)(TAM/1000)];
        TAM=TAM%1000;
        CHUC= MALED[(INT8)(TAM/100)];
        TAM=TAM%100;
        DV= MALED[(INT8)(TAM/10)];
        TP=MALED[(INT8)(TAM%10)];
}

VOID DICH1(VOID)
{       RESET=1;
        SER=1;
        SCK=1;
        SCK=0;
        RCK=1;
         RCK=0;
}

VOID DICH0(VOID)
{       RESET=1;
         SER=0;
         SCK=1;
        SCK=0;
         RCK=1;
        RCK=0;    }
VOID RSOUT(VOID)
{
         RESET=0;
         RCK=1;
         RCK=0;
}

VOID HIENTHI()
{
        TAM=0;
  WHILE(TAM<100)
//SE XUAT RA 100 LAN TUONG UNG THOI GIAN
  {
    OUTPUT_B(TRAM);//DUA MA LED RA PORT B
    DICH1();
    DELAY_US(100);
    OUTPUT_B(0XFF);//DUA CAC CHAN CUA PORT B
LEN TRANG THAI CHUAN BI NHAN INPUT

    OUTPUT_B(CHUC);//DUA MA LED RA PORT B
    DICH0();
    DELAY_US(100);
    OUTPUT_B(0XFF);//DUA CAC CHAN CUA PORT B
LEN TRANG THAI CHUAN BI NHAN INPUT

    OUTPUT_B(DV);//DUA MA LED RA PORT B
    OUTPUT_BIT( PIN_B7, 0);
    DICH0();
    DELAY_US(100);
    OUTPUT_B(0XFF);//DUA CAC CHAN CUA PORT B
LEN TRANG THAI CHUAN BI NHAN INPUT

    OUTPUT_B(TP);//DUA MA LED RA PORT B
    DICH0();
    DELAY_US(100);
    OUTPUT_B(0XFF);//DUA CAC CHAN CUA PORT B
LEN TRANG THAI CHUAN BI NHAN INPUT
    RSOUT();
    TAM++;//TANG LAN DA XUAT GIO PHUT DAY
   }
}
VOID MAIN()
{
  TRISB=0;
  TRISC=0;
  TRISD=0;
 SETUP_ADC(ADC_CLOCK_INTERNAL);
 SETUP_ADC_PORTS(AN0_VREF_VREF);
 WHILE(1)
 {
```

```
    SET_ADC_CHANNEL(0);
    DELAY_US(10);
    READ_ADC();
    HEX_BCD();
    HIENTHI();
  }
 }
```