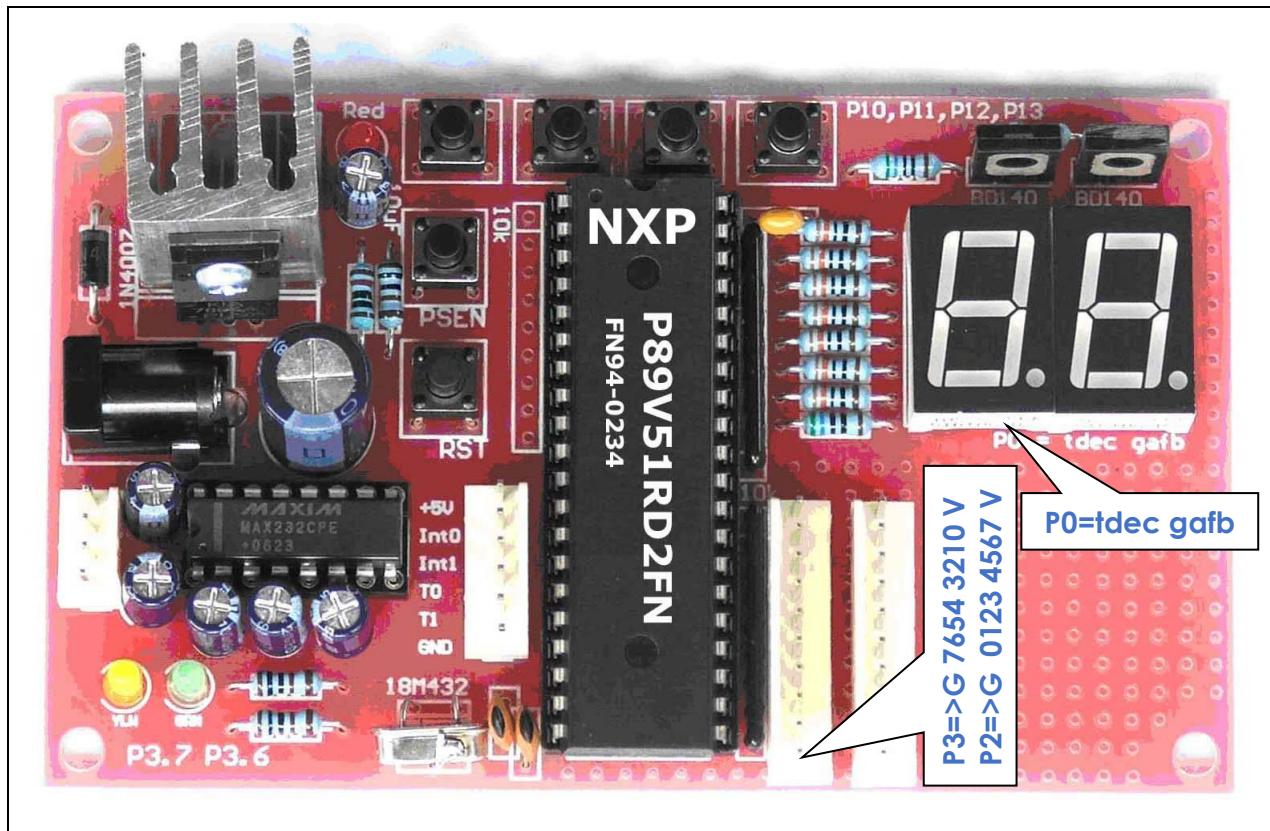


ภาคผนวก 1/2: การทดลองใช้งาน SUT-V5.0 MCS-51 Board

- โครงสร้างวงจรสำหรับอิอร์ด SUT-V5x: MCU-P89V51RD2 ดังนี้



- พัฒนาโปรแกรมด้วย Assembly for MCS51 ผ่าน MIDE-51

- พัฒนาโปรแกรมด้วยภาษา BASIC ผ่านโปรแกรม BSCOM-51

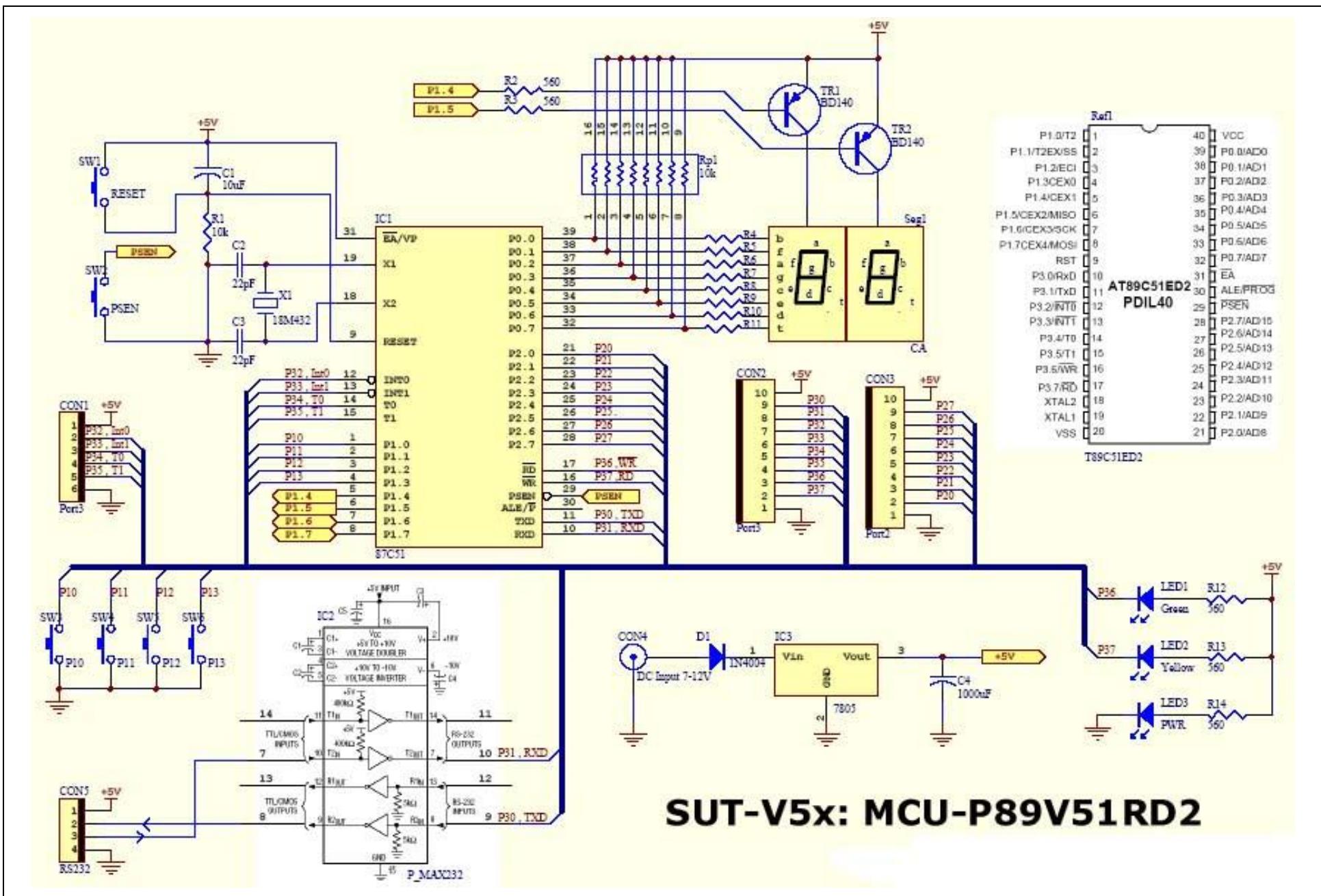
https://www.mcselec.com/index.php?option=com_docman&task=cat_view&gid=98&Itemid=54 > Install "DEMO" + Install "V2.0.14.0"

- พัฒนาโปรแกรมด้วยภาษา C ผ่านโปรแกรม RIDE-51

- MCS-51 ที่ทดสอบเป็นของ Philips เบอร์ P89V51RD2 การ Upload HEX File สามารถทำด้วย Flash Magic Program Version 11.01

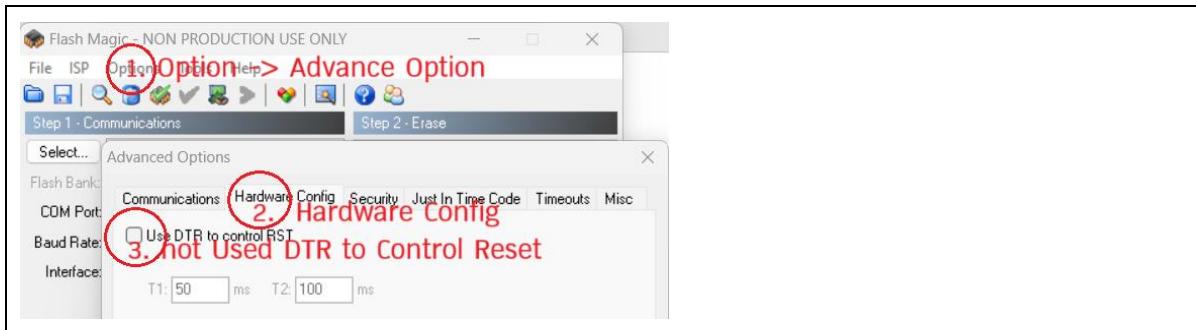
- Download Flad Magic -- V 11.01.5005 →

<https://www.flashmagictool.com/download.html&d=11.01/FlashMagic.exe>

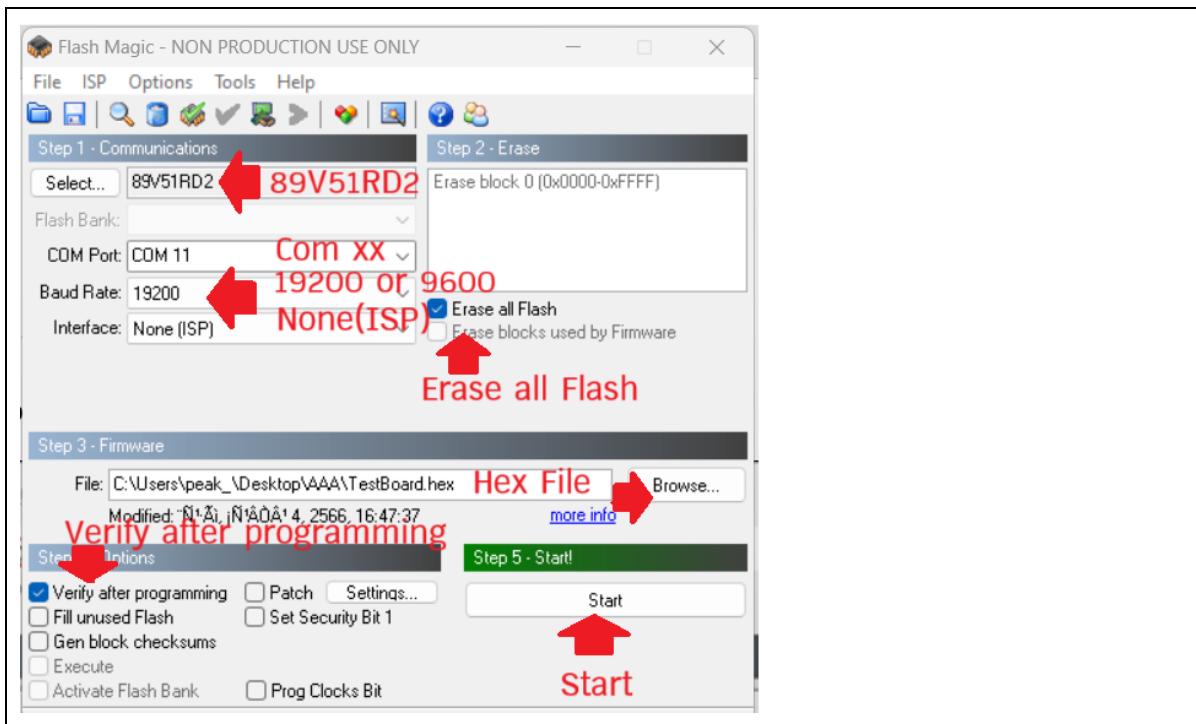


7. การใช้งาน Flash Magic

7.1 Open Flash Magic and Config



7.2 ให้กำหนดค่า CPU, ComX, Baud, None(ISP), Erase.., Hex File, Verify.. แล้ว ยังไงกด Start



7.3 กดปุ่ม Reset ที่ MCS-51 Board ค้างไว้ แล้วกด Start ที่โปรแกรมจนกว่าขึ้น **Reset the ..** แล้วปล่อยปุ่ม Reset โปรแกรมจะทำการ Upload Hex File ไปยังบอร์ด รอจนกว่าจะขึ้น **Finished**



7.4 กดและปล่อย ปุ่ม Reset ที่ MCS-51 Board อีกครั้งเพื่อเริ่มการทำงาน

Assembly for MCS-51

8. ไฟกระพริบที่ P3.7, P3.6 LED

```
ORG    0000H
JMP    0100H

ORG    0100H
MOV    SP,#2FH
LOOP: SETB   P3.7
      CLR    P3.6
      CALL   DELAY
      CALL   DELAY
      CLR    P3.7
      SETB   P3.6
      CALL   DELAY
      CALL   DELAY
      JMP    LOOP

Delay: MOV    B,#00H
       MOV    A,#00H
_DLY00: DJNZ  Acc,_DLY00
         DJNZ  B,_DLY00
         RET

END
```

9. การแสดงผลที่ 7_Segment ด้านขวาฝื้อ

```

LF_SEG EQU P1.4
RG_SEG EQU P1.5
PT_SEG EQU P0

        ORG 0000H
        JMP 0100H

        ORG 0100H
        MOV SP,#2FH
        SETB LF_SEG
        CLR RG_SEG
        MOV R7,#0
LOOP:   MOV A,R7
        ANL A,#0FH
        MOV DPTR,#T_SEG
        MOVC A,@A+DPTR
        CPL A
        MOV PT_SEG,A
        CALL DELAY
        CALL DELAY
        INC R7
        JMP LOOP

T_SEG:  DB 01110111B, 00010001B, 00001000B, 00001000B
        DB 00001000B, 00001000B, 00001000B, 00001000B
        DB 00001000B, 00001000B, 00001000B, 00001000B
        DB 00001000B, 00001000B, 00001000B, 00001000B

Delay:  MOV B,#00H
        MOV A,#00H
_DLY00: DJNZ Acc,_DLY00
        DJNZ B,_DLY00
        RET

        END
    
```

10. กดสวิตช์ P1.0 แสดง 0, กดสวิตช์ P1.1 แสดง 1, ไม่กดสวิตช์แสดง -

```

LF_SEG      EQU      P1.4
RG_SEG      EQU      P1.5
PT_SEG      EQU      P0

        ORG      0000H
        JMP      0100H

        ORG      0100H
        MOV      SP,#2FH
        SETB    LF_SEG
        CLR      RG_SEG
        ORL      P1,#00001111B

LOOP:      MOV      A,#01110111B
        JNB      P1.0,SHOW
        MOV      A,#00010001B
        JNB      P1.1,SHOW
        MOV      A,#00001000B

SHOW:      CPL      A
        MOV      PT_SEG,A
        CALL    DELAY
        JMP      LOOP

Delay:     MOV      B,#00H
        MOV      A,#00H
_DLY00:   DJNZ    Acc,_DLY00
        DJNZ    B,_DLY00
        RET

        END

```

11. ไม่กดสวิตช์แสดง -, กดสวิตช์ P1.0 แสดง 0, กดสวิตช์ P1.1 แสดง 1 + กดมากกว่า 2 สวิตช์แสดง E

	LF_SEG	EQU	P1.4
	RG_SEG	EQU	P1.5
	PT_SEG	EQU	P0
		ORG	0000H
		JMP	0100H
		ORG	0100H
		MOV	SP,#2FH
		SETB	LF_SEG
		CLR	RG_SEG
		ORL	P1,#00001111B
LOOP:	MOV	A,P1	
	ANL	A,#0FH	
	MOV	R7,A	
	MOV	B,#00001000B	
	MOV	A,R7	
	XRL	A,#00001111B	
	JZ	SHOW	
	MOV	B,#01110111B	
	MOV	A,R7	
	XRL	A,#00001110B	
	JZ	SHOW	
	MOV	B,#00010001B	
	MOV	A,R7	
	XRL	A,#00001101B	
	JZ	SHOW	
SHOW:	MOV	B,#01101110B	
	MOV	A,B	
	CPL	A	
	MOV	PT_SEG,A	
	CALL	DELAY	
	JMP	LOOP	
Delay:	MOV	B,#00H	
	MOV	A,#00H	
_DLY00:	DJNZ	Acc,_DLY00	
	DJNZ	B,_DLY00	
	RET		
	END		

12. ทดสอบการแสดงผลแบบสองหลัก

```

        ORG    0000H
        JMP    0100H

        ORG    0100H
        MOV    SP,#2FH

        MOV    R7,#00

LOOP:   MOV    R6,#2
        MOV    R5,#00
_WAIT:  CALL   SHOW7
        DJNZ   R5,_WAIT
        DJNZ   R6,_WAIT
        INC    R7
        JMP    LOOP

SHOW7:  SETB   P1.4
        CLR    P1.5
        MOV    A,R7
        CALL   _7SHOW
        CLR    P1.4
        SETB   P1.5
        MOV    A,R7
        SWAP   A
_7SHOW: MOV    DPTR,#X_SEG
        ANL    A,#0FH
        MOVC  A,@A+DPTR
        CPL    A
        MOV    P0,A
        CLR    A
        DJNZ   Acc,$
        RET

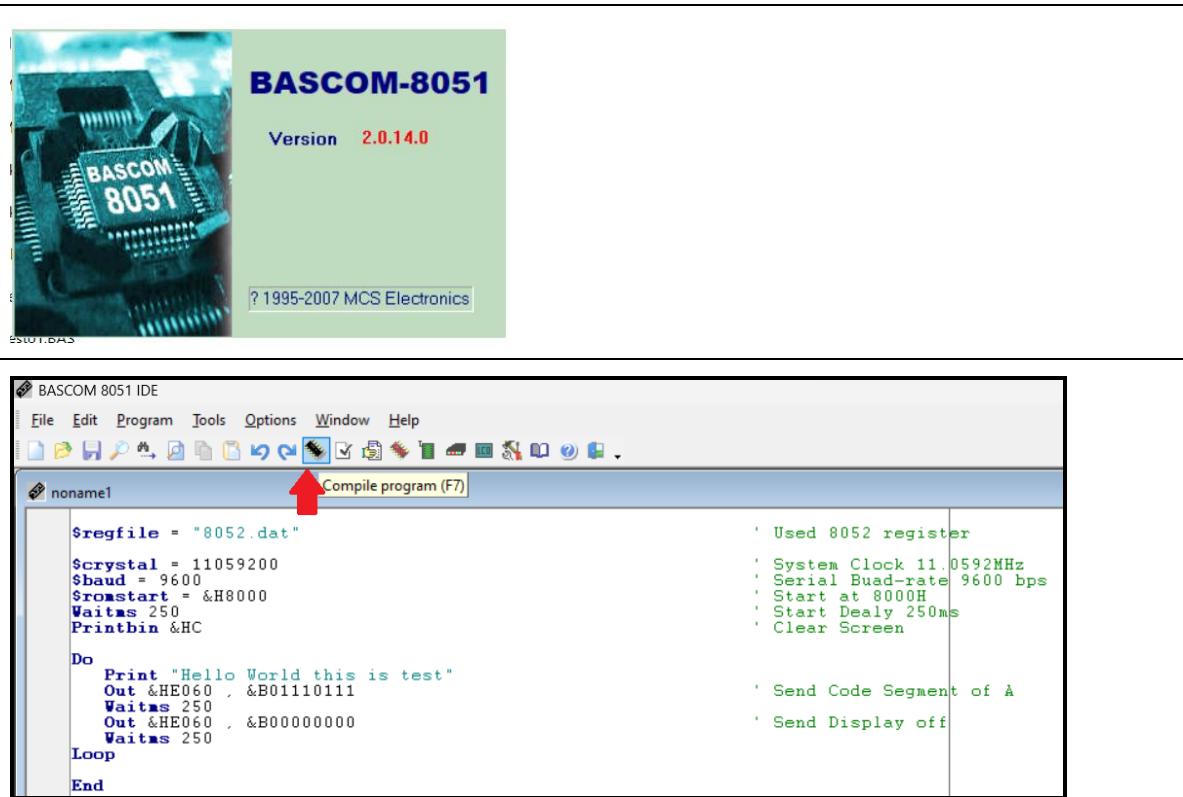
X_SEG:  DB     77H, 11H, 6DH, 5DH, 1BH, 5EH, 7EH, 15H ; Code 0123 4567
        DB     7FH, 5FH, 3FH, 7AH, 66H, 79H, 6EH, 2EH ; Code 89ab cdef

        END

```

BASCOM-51

13. Install BASCOM-DEMO
14. Install BASCOM V2.0.14.0
15. BASCOM Test01 (ETT-8032 Board) ส่งข้อความ “Hello” เพื่อทดสอบการใช้งาน BASCOM
 - Open BASCOM – 8051 → Edit → Save “Test01.BAS” → Compile Program
 - Load code to ETT-8032 Board via Hyper Terminal



The screenshot shows the BASCOM-8051 IDE interface. At the top, there is a logo featuring a microcontroller chip labeled "BASCOM 8051" and the text "BASCOM-8051 Version 2.0.14.0". Below the logo, it says "© 1995-2007 MCS Electronics". The main window displays a BASIC-like program code. A red arrow points to the "Compile program (F7)" button in the toolbar above the code editor. The code itself is as follows:

```

$regfile = "8052.dat"           ' Used 8052 register
$crystal = 11059200            ' System Clock 11.0592MHz
$baud = 9600                   ' Serial Buad-rate 9600 bps
$romstart = &H8000             ' Start at 8000H
$waitms 250                   ' Start Dealy 250ms
$printbin &HC                 ' Clear Screen

Do
    Print "Hello World this is test"
    Out &HE060, &B01110111     ' Send Code Segment of A
    $waitms 250
    Out &HE060, &B00000000     ' Send Display off
    $waitms 250
Loop

End

```

Below the IDE window, the same program code is displayed again, likely representing the compiled output or a copy of the original code.

16. BASCOM Test02 (ETT-8032 Board) – นับเลขที่ EO60H ด้วยการเปิดตารางโดย BASCOM-51

```
'=====
$regfile = "8052.dat"           ' Used 8052 register
$crystal = 11059200            ' System Clock 11.0592MHz
$baud = 9600                   ' Serial Buad-rate 9600 bps
$romstart = &H8000             ' Start at 8000H

Dim Data_show As Byte
Dim Data_index As Byte
Waitms 250                      ' Start Dealy 250ms
Printbin &HC                     ' Clear Screen

Do
    For Data_index = 0 To &HF
        Print "Data index = ";
        Print Data_index
        Data_show = Lookup(data_index , Segment_table)
        Out &HE060 , Data_show
        Wait 1
    Next
Loop

'=====
Segment_table:                 ' tgfe dcba
    Data &B00111111 , &B00000110 , &B01011011 , &B01001111  ' Code 0123
    Data &B01100110 , &B01101101 , &B01111101 , &B00000111  ' Code 4567
    Data &B01111111 , &B01101111 , &B01110111 , &B01111100  ' Code 89ab
    Data &B00111001 , &B01011110 , &B01111001 , &B01110001  ' Code cdef

End
```

17. BASCOM Test03 (ETT-8032 Board) – ใช้ P3.4, P3.5 ในการนับขีน-หัวลง

```

'=====
$regfile = "8052.dat"           ' Used 8052 register
$crystal = 11059200             ' System Clock 11.0592MHz
$baud = 9600                     ' Serial Buad-rate 9600 bps
$romstart = &H8000              ' Start at 8000H

Dim Data_Show As Byte
Dim Data_Index As Byte
Waitms 250                      ' Start Dealy 250ms
Printbin &HC                    ' Clear Screen
Data_Index = 8                   ' Initial Data
Gosub Display_Data
Do
    Debounce P3.4 , 0 , Increment , Sub
    Debounce P3.5 , 0 , Decrement , Sub
Loop

'=====
Increment:
Incr Data_Index
If Data_Index > 15 Then Data_Index = 0
Gosub Display_Data
Bitwait P3.4 , Set
Return

'=====
Decrement:
Decr Data_Index
If Data_Index > 15 Then Data_Index = 15
Gosub Display_Data
Bitwait P3.5 , Set
Return

'=====
Display_Data:
Data_Show = Lookup(Data_Index , Segment_Table)
Out &HE060 , Data_Show
Print "Data index = ";
Print Data_Index
Return

'=====
Segment_Table:                  ' tgfe dcba
    Data &B00111111 , &B00000110 , &B01011011 , &B01001111      ' Code 0123
    Data &B01100110 , &B01101101 , &B01111101 , &B00000111      ' Code 4567
    Data &B01111111 , &B01101111 , &B01110111 , &B01111100      ' Code 89ab
    Data &B00111001 , &B01011110 , &B01111001 , &B01110001      ' Code cdef

End

```

18. BASCOM Test04 (SUT V5.0 Board) – เริ่มต้นใช้งาน

```
$regfile = "8052.dat"           ' Used 8052 register
$crystal = 18432000             ' System Clock 18.432MHz
$baud = 9600                     ' Serial Buad-rate 9600 bps
Waitms 250                      ' Start Dealy 250ms
Printbin &HC
P1.5 = 1
P1.4 = 0

Dim Xdata As Byte

Do
    Print "Hello World this is test"
    Xdata = &B00111111           ' Send Code Segment of A
    P0 = Not Xdata               ' Inverse
    Reset P3.6
    Reset P3.7
    Waitms 250                  ' Send Display off
    P0 = Not Xdata               ' Inverse
    Set P3.6
    Set P3.7
    Waitms 250
Loop

End
```

19. BASCOM Test02 (SUT V5.0 Board) – นับเลขที่ LEFT Segment ด้วยการเปิดตาราง

```

$regfile = "8052.dat"           ' Used 8052 register
$crystal = 18432000            ' System Clock 18.432 MHz
$baud = 9600                   ' Serial Buad-rate 9600 bps

Dim Data_show As Byte
Dim Data_index As Byte
Waitms 250                     ' Start Dealy 250ms
Printbin &HC                    ' Clear Screen
Set P1.4
Reset P1.5

Do
    For Data_index = 0 To &HF
        Print "Data index = ";
        Print Data_index
        Data_show = Lookup(data_index, Segment_table)
        P0 = Not Data_show
        Wait 1
    Next
Loop

'=====
Segment_table:                 ' tdec gafb
    Data    &H77, &H11, &H6D, &H5D, &H1B, &H5E, &H7E, &H15      ' Code 0123 4567
    Data    &H7F, &H5F, &H3F, &H7A, &H66, &H79, &H6E, &H2E      ' Code 89ab cdef

End

```

20. BASCOM Test02 (SUT V5.0 Board) – ใช้ P1.0, P1.3 ในการนับขี๊น-นับลง

```
'=====
$regfile = "8052.dat"           ' Used 8052 register
$crystal = 18432000            ' System Clock 18.432MHz
$baud = 9600                   ' Serial Buad-rate 9600 bps

Dim Data_show As Byte
Dim Data_index As Byte
Waitms 250                     ' Start Dealy 250ms
Printbin &HC                   ' Clear Screen
Data_index = 8                  ' Initial Data
Gosub Display_data
Do
    Debounce P1.0 , 0 , Increment , Sub
    Debounce P1.3 , 0 , Decrement , Sub
Loop

'=====
Increment:
Incr Data_index
If Data_index > 15 Then Data_index = 0
Gosub Display_data
Bitwait P1.0 , Set
Return

'=====
Decrement:
Decr Data_index
If Data_index > 15 Then Data_index = 15
Gosub Display_data
Bitwait P1.3 , Set
Return

'=====
Display_data:
Data_show = Lookup(data_index , Segment_table)
Set P1.4
Reset P1.5
P0 = Not Data_show
Print "Data index = ";
Print Data_index
Return

'=====
Segment_table:                 ' tdec gafb
Data      &H77 , &H11 , &H6D , &H5D , &H1B , &H5E , &H7E , &H15      ' Code 0123 4567
Data      &H7F , &H5F , &H3F , &H7A , &H66 , &H79 , &H6E , &H2E      ' Code 89ab cdef

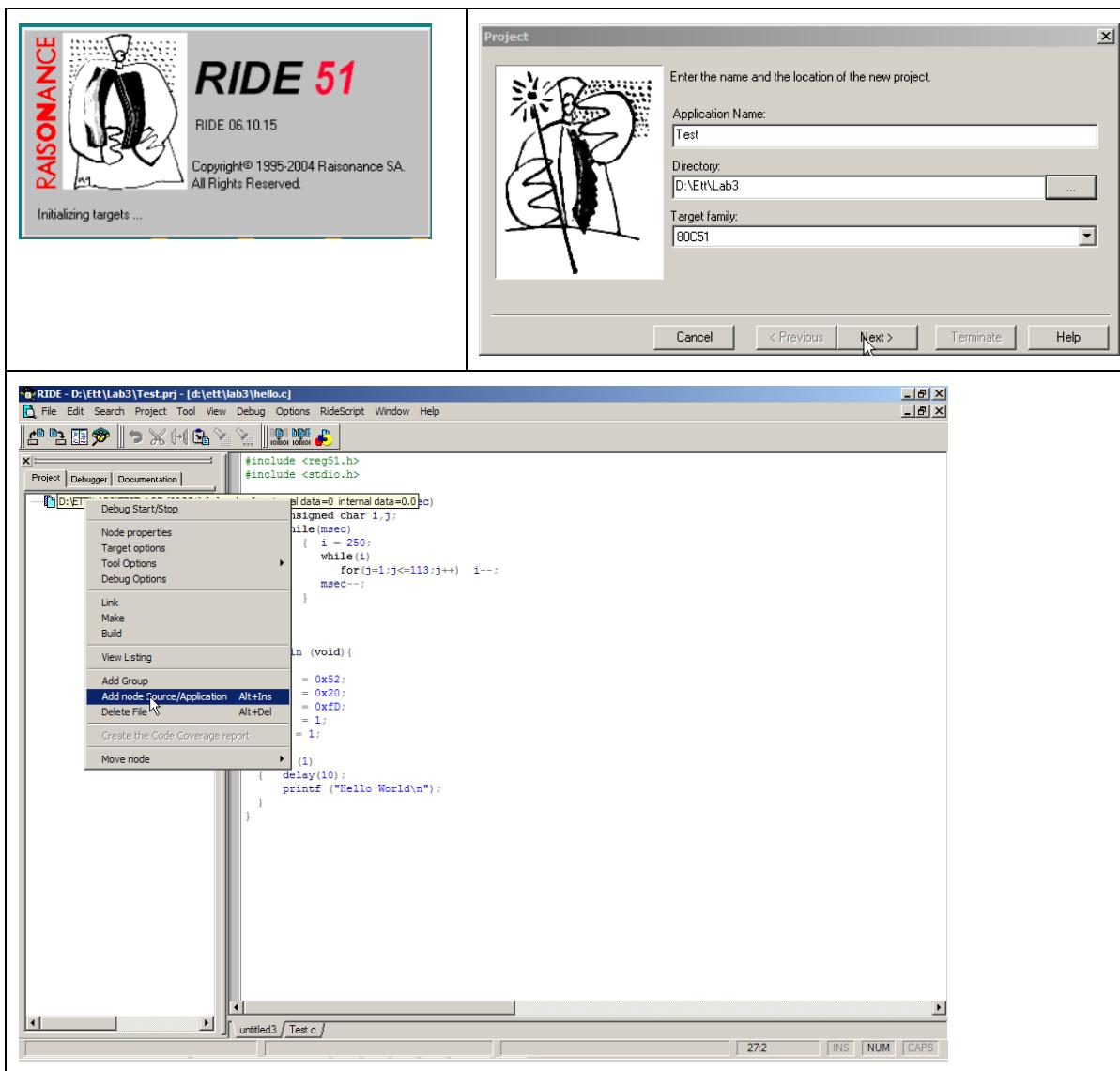
End
```

RIDE-51

21. ติดตั้งโปรแกรม PKit 6.1 -- RIDE-51

22. การสร้างโปรเจ็คใหม่

- Project → New → Application Name: Test, Directory: C:\Ett\Lab3, Target family: P80C51
→ Terminate
- File → New → C Files // and Save as “C:\Ett\Lab3\Prog_11.C”
- Right Click at Project → Add node/Source File Select “Prog_11.C”



<p>กรณี SUT-V5 Board → ORG 0000H</p> <p>กรณี ET-8032 Board → ORG 8000H ตั้งค่าโดย</p> <ul style="list-style-type: none"> → Option → Project → Select LX51 → Linker <p>Set:</p> <p>Starting Address Code = 8000</p> <p>Absolute Address Offset = 8000</p>	<p>หาก New Project ไม่ต้องกำหนดค่า เพราะเป็น Default</p>
--	--

23. Prog_11.C, Prog_12.C, Prog_13.C - ทดสอบการทำงานบน ET-8032 Board ให้ตั้งค่า Strat Code และ Absolute Offset เป็น 8000 ที่หน้าต่าง Linker
24. หลังจากสร้างโปรเจ็ค, Add C File และตั้งค่า Start Code, Offset Code = 8000 แล้วให้ปรับโปรแกรม Prog_11.C แล้วทำการคอมpile { Project → Build All } โหลดไปที่บอร์ดด้วย Hyper Terminal เพื่อทดสอบการทำงาน
25. โปรแกรม “Prog_11.C” ส่งข้อความ“Hello” เพื่อทดสอบการใช้งาน RIDE-51 ดูผลการทำงานที่ Hyper Terminal ที่ Baud rate =9600,n,8,1

```
//== Prog_11.C =====
#include <reg51.h>
#include <stdio.h>
unsigned int Count;

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
        {
            for(i=1;i<=250;i++)
                for(j=1;j<=113;j++);
            msec--;
        }
}

//=====
void main (void)
{
    SCON = 0x52; // Serial Config n,8,1
    TMOD = 0x20; // Set Timer for buad Gen.
    TH1 = 0xfd; // Buad rate 9600bps@11.0592MHz
    TR1 = 1; // Start Timer1
    TI = 1;
    Delay(2);
    while (1)
        {
            printf ("Hello World this Ride-C51 %d\n",Count);
            Count++;
            Delay(2);
        }
}
//=====
```

RIDE - C:\Users\peak\Desktop\AAA\pkTest.prj - [c:\users\peak\Desktop\aaa\c_code.c]

File Edit Search Project Tool View Debug Options RideScript Window Help

Project | Debugger | Documentation |

C:\USERS\PEAK_\DESKTOP\AAA\PKTEST.AOF

- c_code.c [RC51] code=71 const=31 xdata:
 - c\RIDE\Inc\51\reg51.h
 - c\RIDE\Inc\stdio.h
 - c\RIDE\Inc\stdarg.h
 - c\RIDE\Inc\51\MAIN_C51.h

```
===== Prog_11.C =====
#include <reg51.h>
#include <stdio.h>
unsigned int Count;

=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

=====
void main (void)
{
    SCON = 0x52;      // Serial Config n,8,1
    TMOD = 0x20;      // Set Timer for baud Gen.
    TH1  = 0xfb;      // Baud rate 9600bps@18.432MHz
    TR1  = 1;          // Start Timer1
    TI   = 1;
    Delay(2);
    while (1)
    {
        printf ("Hello World this Ride-C51 %d\n",Count);
        Count++;
        Delay(2);
    }
}
=====
```

Make | Debug | Grep | Script |

- Running RC51 on c:\users\peak\Desktop\aaa\c_code.c
- Running LX51 on c:\users\peak\Desktop\aaa\pktest.aof
- Running the tool c\RIDE\Bin\oh51v32.dll [HEX] on c:\users\peak\Desktop\aaa\pktest.AOF
 - ◆ GENERATING STANDARD HEX FILE: C:\USERS\PEAK_\DESKTOP\AAA\PKTEST.HEX

x - HyperTerminal

Name: hyperterm.dll

File Edit View Call Transfer Help

Connected 0:00:12 Auto detect 9600 8-N-1 SCROLL CAPS NUM Capture Print echo

```
Hello World this Ride-C51 800
Hello World this Ride-C51 801
Hello World this Ride-C51 802
Hello World this Ride-C51 803
Hello World this Ride-C51 804
Hello World this Ride-C51 805
Hello World this Ride-C51 806
Hello World this Ride-C51 807
Hello World this Ride-C51 808
Hello World this Ride-C51 809
Hello World this Ride-C51 810
Hello World this Ride-C51 811
Hello World this Ride-C51 812
Hello World this Ride-C51 813
Hello World this Ride-C51 814
Hello World this Ride-C51 815
Hello World this Ride-C51 816
Hello World this Ride-C51 817
Hello World this Ride-C51 818
Hello World this Ride-C51 819
Hello World this Ride-C51 820
Hello World this Ride-C51 821
Hello World this Ride-C51 822
-
```

26. โปรแกรม “Prog_12.C” นำบล็อกที่ E060H เพื่อทดสอบการติดต่อกับ External Port

```

//== Prog_12.C =====
#include <reg51.h>
#include <stdio.h>
xdata at 0xE060 PortSeg;
unsigned int Count;
code unsigned char CodeSeg[] = {
    0x3f, 0x06, 0x5b, 0x4f, 0x66,
    0x6d, 0x7d, 0x07, 0x7f, 0x6f};

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

//=====
void main (void)
{
    SCON    = 0x52; // Serial Config n,8,1
    TMOD   = 0x20; // Set Timer for baud Gen.
    TH1    = 0xfd; // Baud rate 9600bps@11.0592MHz
    TR1    = 1; // Start Timer1
    TI    = 1;
    Delay(1);
    while (1)
    {
        for(Count=0; Count<=9; Count++)
        {
            printf ("Counter = %d\n",Count);
            PortSeg = CodeSeg[Count];
            Delay(2);
        }
    }
}
//=====

```

27. โปรแกรม “Prog_13.C” ใช้ P3.4, P3.5 ในการนับขึ้น-ลง เพื่อให้มีแนวคิดในการเขียนโปรแกรม

```

//== Prog_13.C =====
#include <reg51.h>
#include <stdio.h>
xdata at 0xe060 PortSeg;
unsigned int Count;
sbit P_Incress = P3^4;
sbit P_Decress = P3^5;
code unsigned char CodeSeg[] = {    0x3f, 0x06, 0x5b, 0x4f, 0x66,
                                    0x6d, 0x7d, 0x07, 0x7f, 0x6f};

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

//=====
void Send_Display(void)
{
    printf ("Hello World this Ride-C51 %d\n",Count);
    PortSeg = CodeSeg[Count];
    Delay(1);
}

//=====
void main (void)
{
    SCON = 0x52; // Serial Config n,8,1
    TMOD = 0x20; // Set Timer for buad Gen.
    TH1 = 0xfd; // Buad rate 9600bps@11.0592MHz
    TR1 = 1; // Start Timer1
    TI = 1;
    Delay(1);
    P_Incress = 1; //Set for Input Bit
    P_Decress = 1; //Set for Input Bit
    Count = 8;
    Send_Display();
    while (1)
    {
        if (P_Incress == 0)
        {
            Count++;
            if (Count>=10) Count = 0;
            Send_Display(); while(~P_Incress); // Wait Until Port_Incress = ~0
        }

        if (P_Decress == 0)
        {
            Count--;
            if (Count>=10) Count = 9;
            Send_Display(); while(~P_Decress); // Wait Until Port_Decress = ~0
        }
    }
}
//=====

```

28. Prog_2x.C - ทดสอบการทำงานบน SUT-V5 Board ไม่ต้องตั้งค่า Strat Code และ Absolute Offset
เนื่องจากเป็นค่า default ที่ 0000
29. หลังจากสร้างโปรเจ็ค, Add C File และให้ปรับโปรแกรม Prog_21.C และทำการคอมไพล์ { Project → Build All } โดยดูไปที่บอร์ดด้วย Flash Magic เพื่อทดสอบการทำงาน
30. โปรแกรม “Prog_21.C” ทดสอบควบคุม LED

```

//=====
// Program : Prog_21.C Output Test
// By: Wichai Srisuruk
//=====

#include <reg52.h>

sbit GRN_LED = P3^6;
sbit RED_LED = P3^7;

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++) //not :
                msec--;
    }
}

//=====
void main (void)
{
    P3 = 0xff;
    while(1)
    {
        RED_LED = 1;
        GRN_LED = 0;
        Delay(4);
        RED_LED = 0;
        GRN_LED = 1;
        Delay(4);
    }
}

```

31. โปรแกรม “Prog_22.C” ทดสอบอ่านค่า Switch ควบคุม LED

```

//=====
// Program : Prog_22.C Input Test
// By: Wichai Srisuruk
//=====
#include <reg52.h>

sbit GRN_LED = P3^6;
sbit RED_LED = P3^7;
sbit SW_Left = P1^0;
sbit SW_Right = P1^3;

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++) //not :
                msec--;
    }
}

//=====
void main (void)
{
    SW_Left = 1;
    SW_Right = 1;
    while(1)
    {
        if (SW_Left == 0) {
            RED_LED = 1;
            GRN_LED = 0;
            Delay(4);
            RED_LED = 0;
            GRN_LED = 1;
            Delay(4);
        }
        if (SW_Right == 0) {
            RED_LED = 1;
            GRN_LED = 1;
            Delay(4);
            RED_LED = 0;
            GRN_LED = 0;
            Delay(4);
        }
    }
}

```

32. โปรแกรม “Prog_23.C” ส่งข้อความ“Hello” เพื่อทดสอบการใช้งาน RIDE-51 ดูผลการทำงานที่ Hyper Terminal ที่ Baud rate =9600,n,8,1

```

//== Prog_23.C =====
#include <reg51.h>
#include <stdio.h>
unsigned int Count;

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

//=====
void main (void)
{
    SCON    = 0x52; // Serial Config n,8,1
    TMOD   = 0x20; // Set Timer for buad Gen.
    TH1    = 0xfb; // Buad rate 9600bps@18.432MHz
    TR1    = 1;    // Start Timer1
    TI    = 1;
    Delay(2);
    while (1)
    {
        printf ("Hello World this Ride-C51 %d\n",Count);
        Count++;
        Delay(2);
    }
}
//=====

```

33. โปรแกรม “Prog_24.C” นับเลขที่ 2Digit 7Segment

```

//== Prog_24.C =====
#include <reg51.h>
#include <stdio.h>

sbit CS_Seg0 = P1^4;
sbit CS_Seg1 = P1^5;

unsigned int Count;
code unsigned char CodeSeg[] = { 0x77,0x11,0x6D,0x5D,0x1B,0x5E,0x7E,0x15,
                                0x7F,0x5F,0x3F,0x7A,0x66,0x79,0x6E,0x2E };

//=====
void Delay(unsigned char msec) {
    unsigned char i,j;
    while(msec){
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

//=====
void Seg2DigitDisplay(unsigned char xDelay) {
    unsigned char i, j, k, splitData;
    for(k=1;k<=xDelay;k++) {
        for(j=1;j<=250;j++) {
            splitData = (Count / 10) % 10;
            CS_Seg0 = 0;
            CS_Seg1 = 1;
            P0 = ~(CodeSeg[splitData]);
            for(i=1;i<=250;i++);
            splitData = Count % 10;
            CS_Seg0 = 1;
            CS_Seg1 = 0;
            P0 = ~(CodeSeg[splitData]);
            for(i=1;i<=250;i++);
        }
    }
}

//=====
void main (void)
{
    SCON      = 0x52; // Serial Config n,8,1
    TMOD     = 0x20; // Set Timer for buad Gen.
    TH1      = 0xfb; // Buad rate 9600bps@18.432MHz
    TR1      = 1; // Start Timer1
    TI      = 1;
    Delay(1);
    while (1) {
        printf ("Counter = %d\n",Count);
        Seg2DigitDisplay(2);
        Count++;
    }
}
//=====

```

34. โปรแกรม “Prog_25.C” ใช้ P1.0, P1.3 ในการนับขึ้น-ลง เพื่อให้มีแนวคิดในการเขียนโปรแกรม

```

//=====
// Program : Prog_25.C Test
// By: Wichai Srisuruk
//=====

#include <reg52.h>
sbit P_Incress = P1^0;
sbit P_Decress = P1^3;
sbit CS_Seg0 = P1^4;
sbit CS_Seg1 = P1^5;
sbit LED_Yellow = P3^6;
sbit LED_Green = P3^7;
int Count;
code unsigned char CodeSeg[] = { 0x77,0x11,0x6D,0x5D,0x1B,0x5E,0x7E,0x15,
                                0x7F,0x5F,0x3F,0x7A,0x66,0x79,0x6E,0x2E };

//=====
void Delay(unsigned char msec)
{
    unsigned char i,j;
    while(msec)
    {
        for(i=1;i<=250;i++)
            for(j=1;j<=113;j++);
        msec--;
    }
}

//=====
void Send_Display(void)
{
    CS_Seg0 = 0;
    CS_Seg1 = 0;
    P0 = ~(CodeSeg[Count]);
}

//=====
void main (void)
{
    P_Incress = 1;           //Set for Input Bit
    P_Decress = 1;           //Set for Input Bit
    Count = 8;
    Send_Display();
    while (1)
    {
        LED_Yellow = P_Incress;
        if (P_Incress == 0)
        {
            Count++;
            if (Count>=10)          Count = 0;
            Send_Display();
            Delay(2);
            while(~P_Incress);     // Wait Until Port_Incress = ~0
            Delay(2);
        }

        LED_Green = P_Decress;
        if (P_Decress == 0)
        {
            Count--;
            if (Count>=10)          Count = 9;
            Send_Display();
            Delay(2);
            while(~P_Decress);     // Wait Until Port_Decress = ~0
            Delay(2);
        }
    }
}

```