

การทดลองที่ 3 การรับส่งข้อมูลแบบขนานและอนุกรม

วัตถุประสงค์

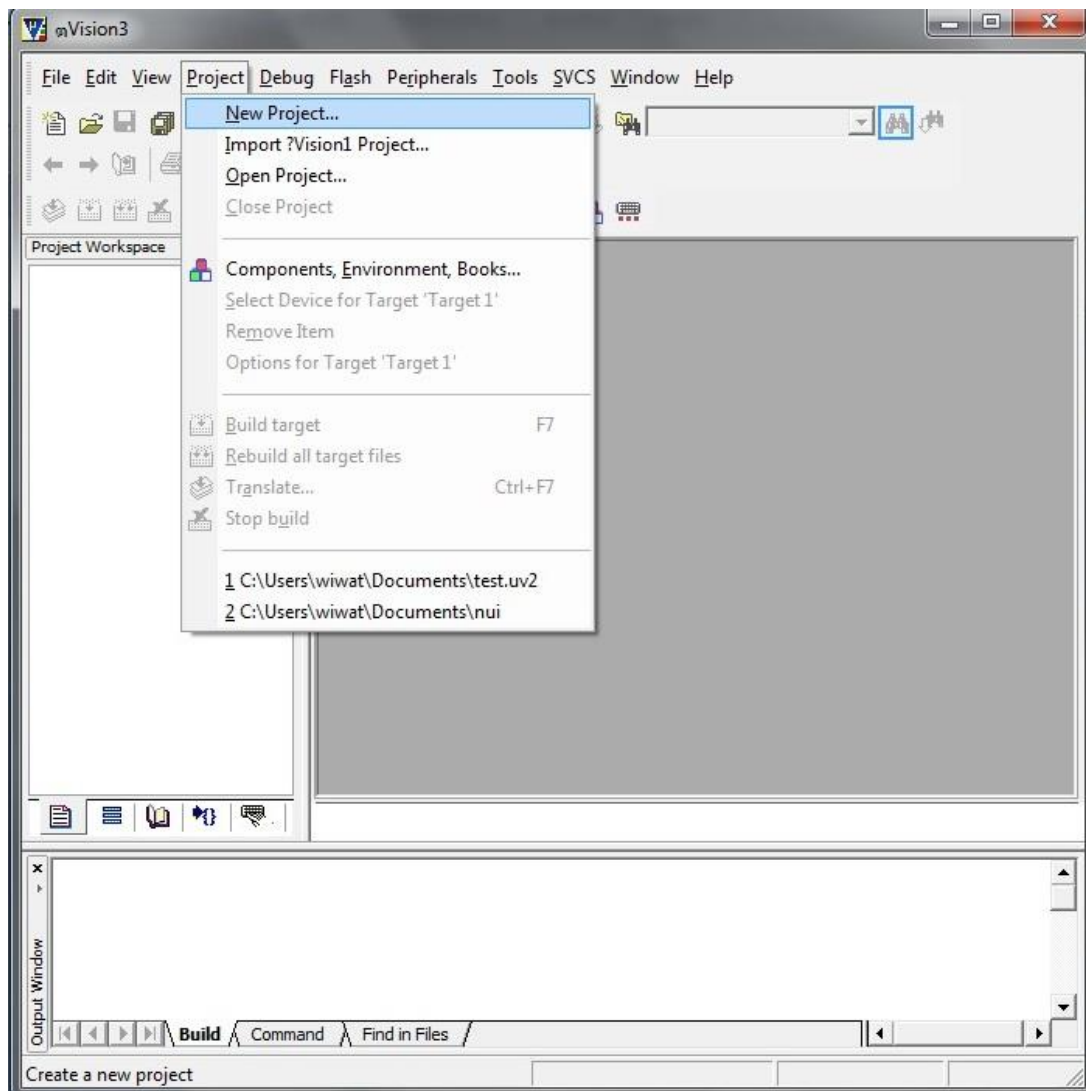
1. เพื่อให้นิสิตสามารถเขียนโปรแกรมภาษา C อย่างง่ายในการควบคุมไมโครคอนโทรลเลอร์ได้
2. เพื่อให้นิสิตเข้าใจการทำงานการรับส่งข้อมูลแบบขนานและอนุกรมเพื่อติดต่อระหว่างไมโครคอนโทรลเลอร์
3. เพื่อให้นิสิตเข้าใจการทำงานการรับส่งข้อมูลแบบขนานและอนุกรมเพื่อติดต่อกับ PC

อุปกรณ์ในการทดลอง

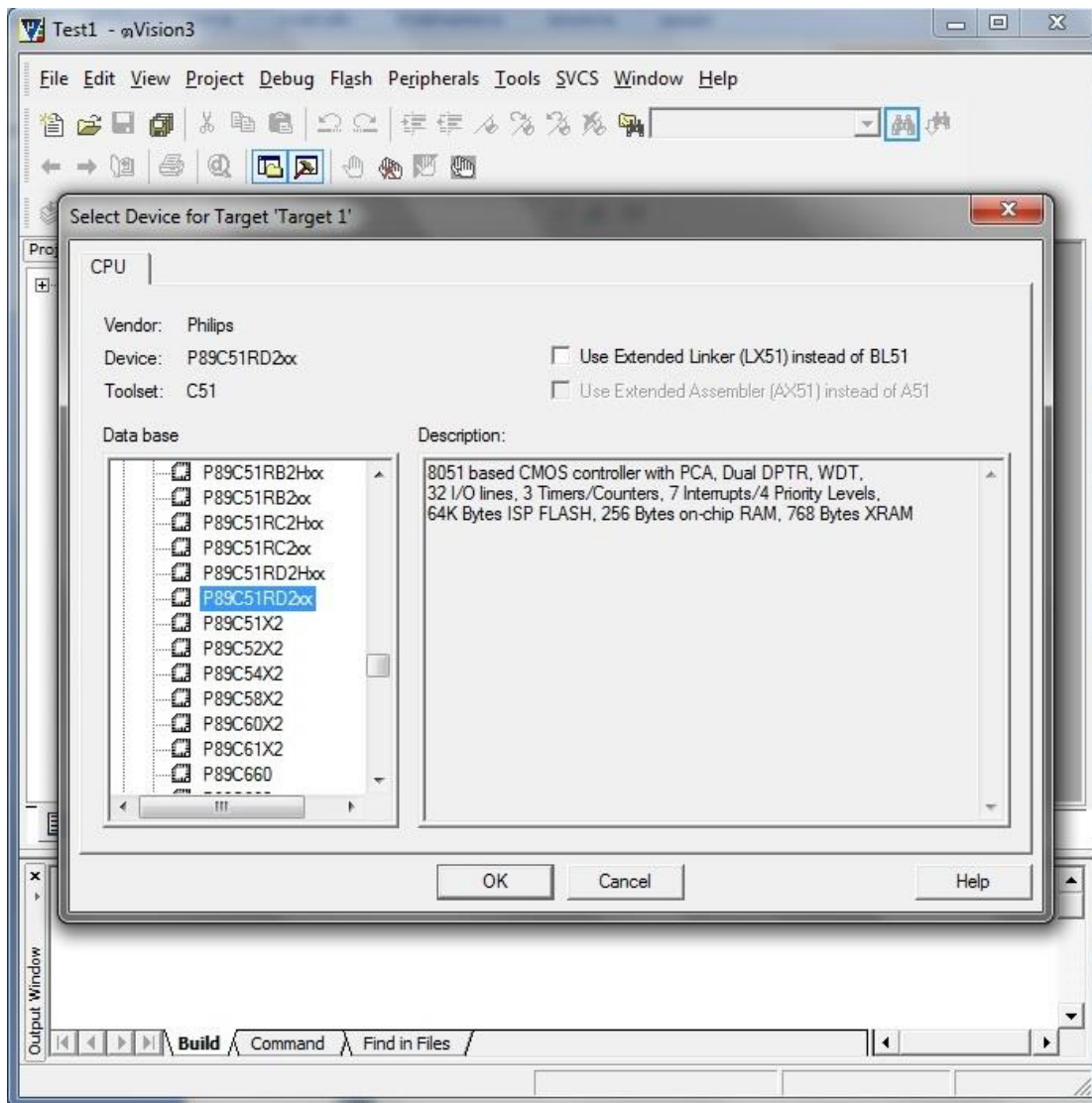
1. ชุดทดลองไมโครคอนโทรลเลอร์ SILA-START-C51
2. เครื่องคอมพิวเตอร์ PC พร้อมโปรแกรมสำหรับการเขียนและคอมไพล์ภาษา C - โปรแกรม Keil51 v.xx
3. สายต่อพอร์ตอนุกรม
4. บอร์ดใช้งาน MCS-51

วิธีการทดลอง

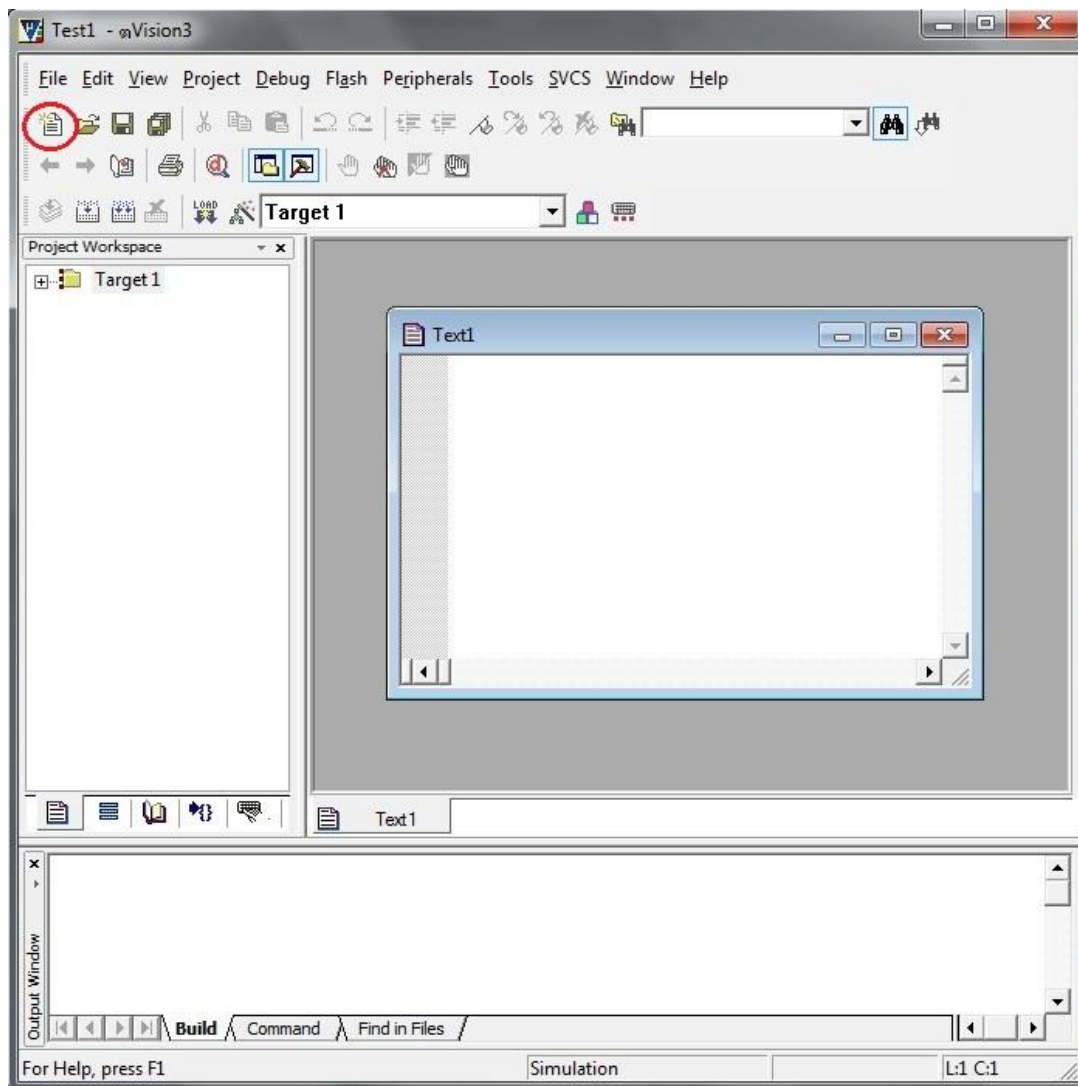
1. ใช้ IDE สำหรับพัฒนาชุดคำสั่งของ MCS-51 ด้วยภาษา C โดยใช้ Keil51 เพื่อเขียนโปรแกรม Lab03_x.c
2. สร้างโปรเจกต์ใหม่โดยเลือก New project ตั้งชื่อเป็น Lab03 แล้วกด ok



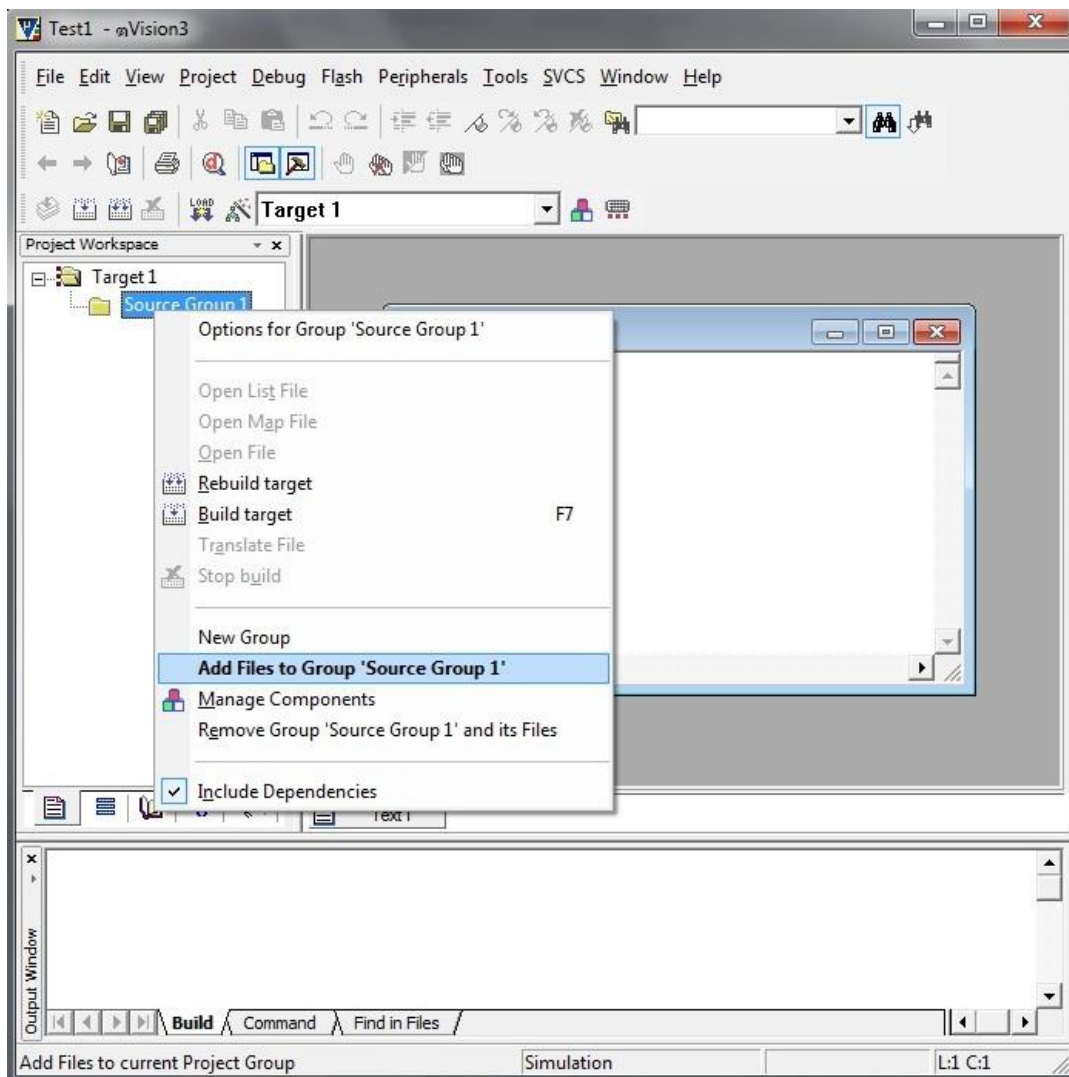
3. เลือก CPU : Philips P89C51RD2xx



4. จากนั้นคลิกที่ปุ่มวงกลมสีแดง เพื่อสร้างหน้าต่างสำหรับเขียนโปรแกรม(หน้าต่าง Text1)
สร้างไฟล์ใหม่ เขียนโปรแกรมและ save โดยใช้ชื่อ Lab03_x.c (x แทนหมายเลขการทดลอง)



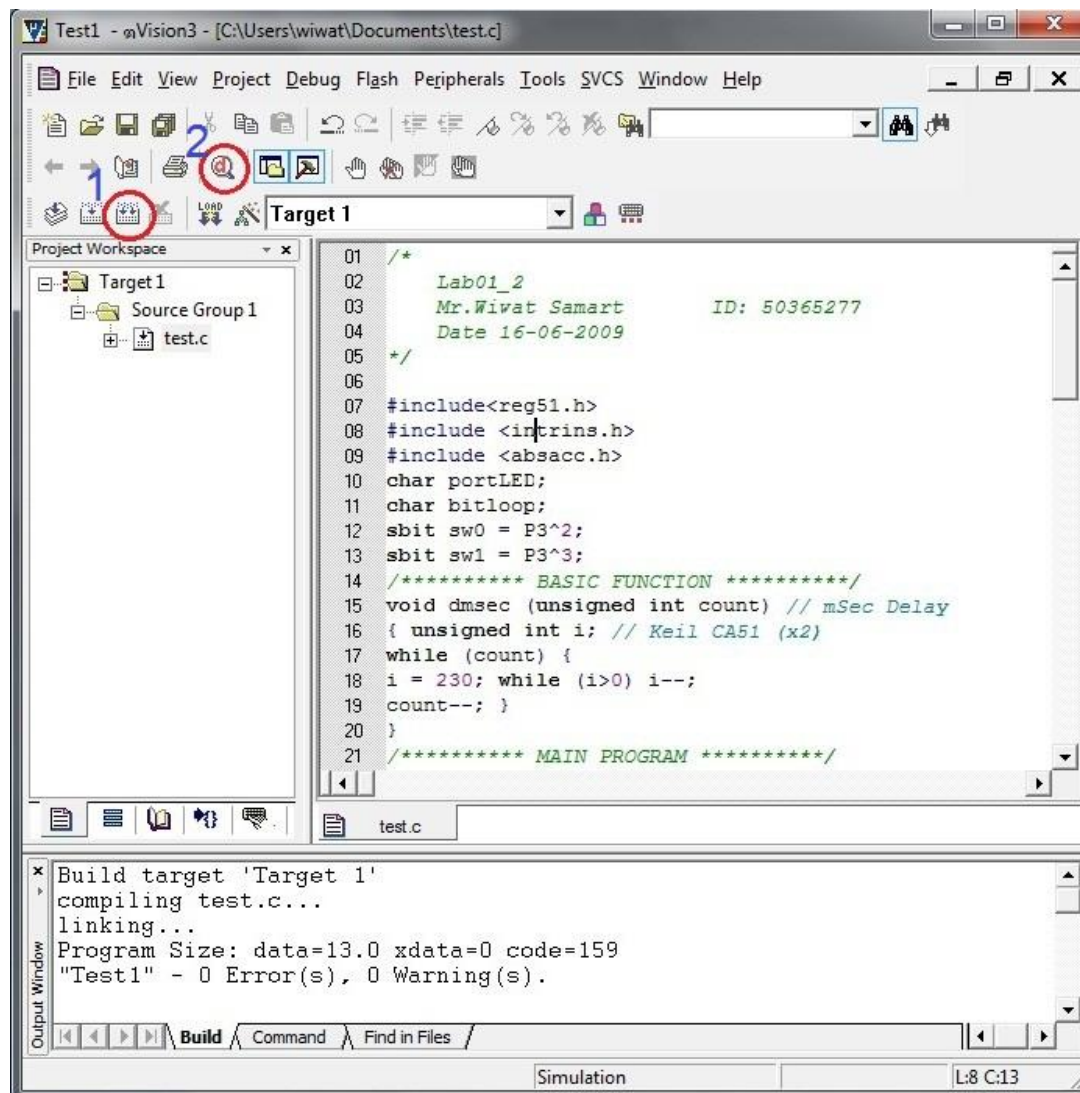
5. คลิกขวาที่ Source Group 1 แล้วเลือก Add file to group 'Source Group 1' แล้วเลือก File Lab03_x.c



6. เขียนโปรแกรมให้เสร็จแล้วคลิกที่ วงกลมที่ 1 เพื่อทำการ compile ถ้าหากว่าไม่พบ Error โดยที่ถ้าเกิด error ขึ้น โปรแกรมจะแสดงว่า error ก็ที่ (ดู windows output ด้านล่างของโปรแกรม) คลิกที่ วงกลมที่ 2 เพื่อให้ keil ตรวจสอบ และทำการ debug พร้อมทั้งตรวจสอบขนาดของโปรแกรมที่เราสร้างขึ้นและพร้อมสำหรับการทดสอบการทำงานของโปรแกรม

7. Build โปรแกรมที่เขียน โดยก่อนหน้านั้น click ขวา ที่ Target1 เลือก Options for Target 'Target1' ที่ Tab Output เลือก create hex file โดยเลือกที่ check box 'Create Hex file'

File ที่ได้จะมีชื่อเหมือนกับชื่อโปรเจ็ค คือ Lab03.hex สามารถเปลี่ยนแปลงชื่อได้โดยแก้ไขชื่อ output file ลงในช่อง 'Name of Executable'



8. เมื่อ Compile และ Build ผ่านโดยไม่มีข้อผิดพลาด (error) จะได้ HEX file เพื่อนำไปใช้งานตามที่กำหนดไว้ในแต่ละการทดลอง

9. บันทึกผลการทดลอง และอธิบายการทำงานของโปรแกรม และทำออกมาในรูปแบบรายงาน

10. ให้นิสิตทำการทดลองที่เหลือ โดยเมื่อทำการทดลองใหม่ให้สร้างไฟล์ใหม่ เขียนโปรแกรมและ save เป็นชื่อใหม่ เช่น สร้างไฟล์ใหม่ชื่อ Lab03_2.c (2 แทนหมายเลขการทดลองที่ 2)

11. จากนั้น Click ที่ Source group แล้ว click ขวาที่ Lab03_1.c ซึ่งอยู่ใน Source group เลือก Remove file 'Lab03_1.c' เพื่อเอา file Lab03_1.c ออกจาก project

12. Click ขวาที่ Source group เลือก Add file to group แล้วเลือก File Lab03_2.c

13. ทำเช่นนี้จนครบทุกการทดลอง ถ้าการทดลองใดมีการบ้านหรือแบบฝึกหัดท้ายการทดลอง ให้ตอบคำถามให้ชัดเจน และถ้าหากเป็นการเขียนโปรแกรม ให้ทำในรูปแบบเช่นเดียวกันกับการทดลอง และแนบมาในรายงานการทดลองพร้อมกันด้วย

วิธีการใช้ SILA-START-C51

ให้ต่อสายอนุกรมจากพอร์ตอนุกรมของเครื่องคอมพิวเตอร์ PC (COM1) ไปยังพอร์ตอนุกรมของ SILA-START-C51

a. ที่เครื่อง PC พิมพ์คำสั่งใน DOS เพื่อทำการกำหนดค่าในติดต่อทางพอร์ตอนุกรมดังนี้

C:\> MODE COM1 9600 , n , 8 , 1 , p (ทำครั้งเดียวเมื่อเปิด PC หรือ เครื่องมีปัญหา)

b. เสียบปลั๊กชุดทดลอง SILA-START-C51 แล้วกดคำสั่งบนบอร์ดทดลองเพื่อรับข้อมูลจากเครื่อง PC ดังนี้

10.1 ทำการลบข้อมูลโปรแกรมที่มีอยู่ก่อนหน้านี้ โดย

MON → BK/ER → 0 (เลือก bank ที่ต้องการลบ)

10.2 ทำการรรับข้อมูลโปรแกรมจากทาง PC โดย

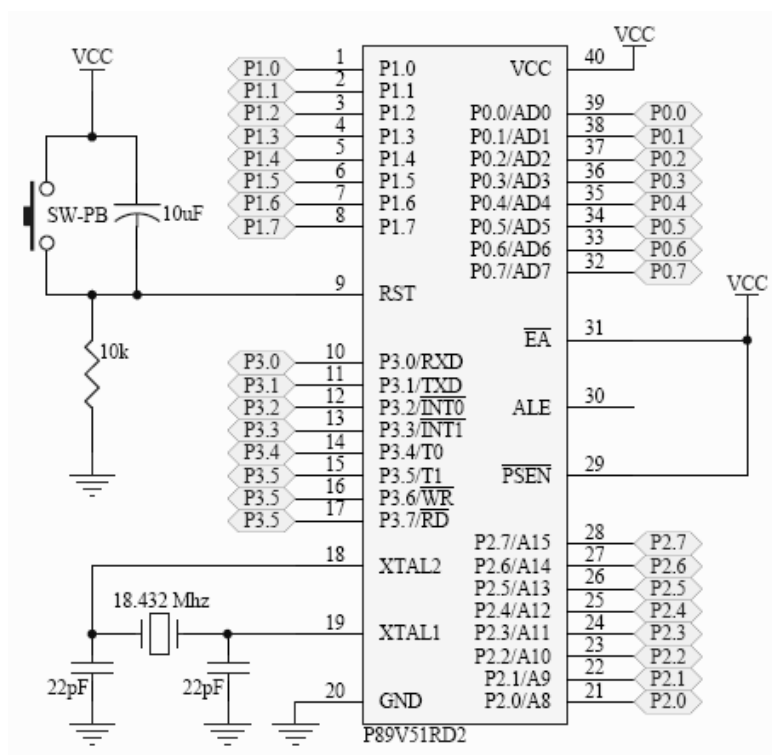
MON → DNLD → FILLD

c. ที่เครื่อง PC พิมพ์คำสั่งเพื่อส่งโปรแกรมที่คอมไพล์แล้วผ่านทางพอร์ตอนุกรมดังนี้

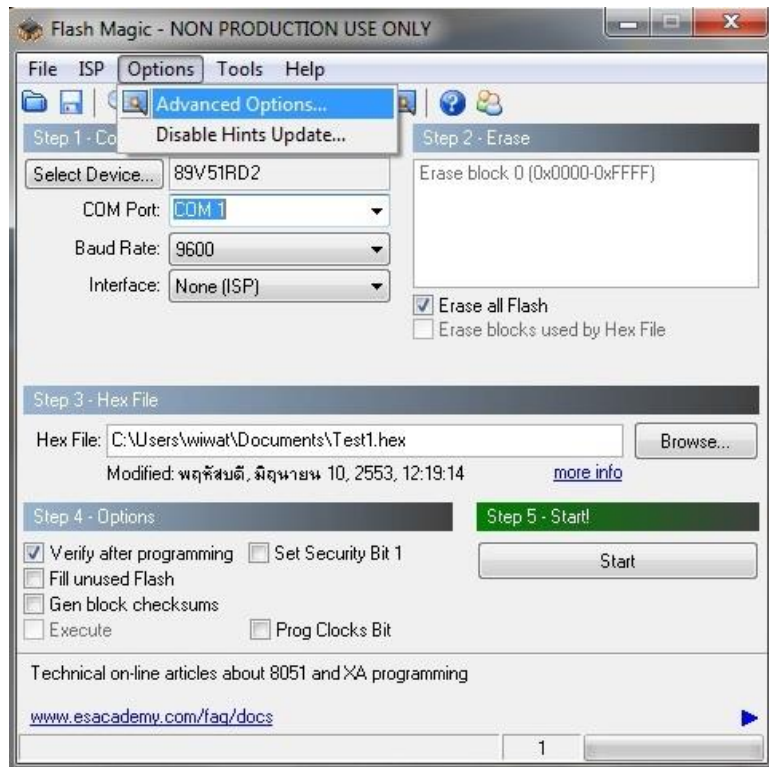
C:\> COPY LAB03.HEX COM1

d. กดปุ่ม RUN และเลือก Address ที่ต้องการ (โดยปรกติจะตั้งไว้ที่ 0000 H) ที่ชุดทดลอง SILA-START-C51 แล้วรอดูผล

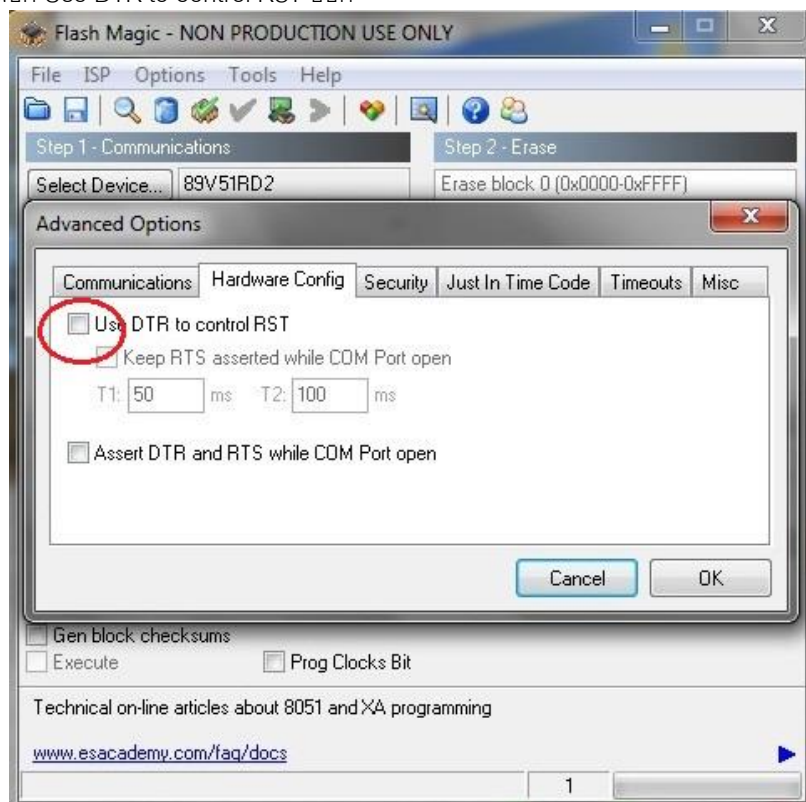
วิธีการใช้บอร์ด MCS-51



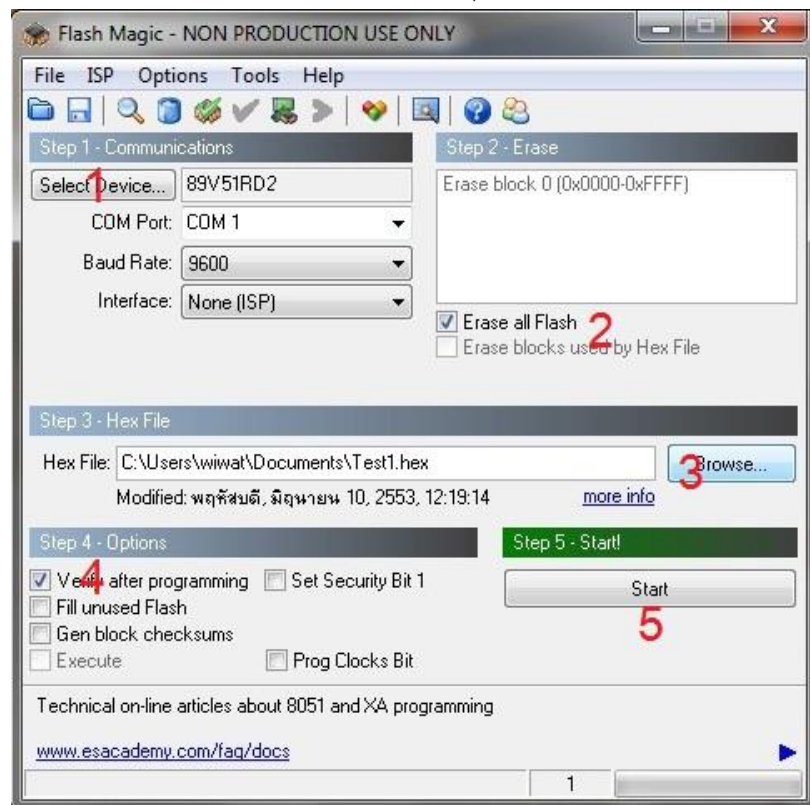
- a. ต่อสายจาก com1 ของ computer เข้ากับบอร์ดโดย**ระวังให้ connector ด้านที่เป็น ground ตรงกับเครื่องหมาย** - ทำการโหลดผ่านโปรแกรม Flash magic ดังนี้
- b. เปิดโปรแกรม Flash magic โดยทำการตั้งค่าต่างๆ ดังนี้ เลือกที่ Options----> Advanced Options



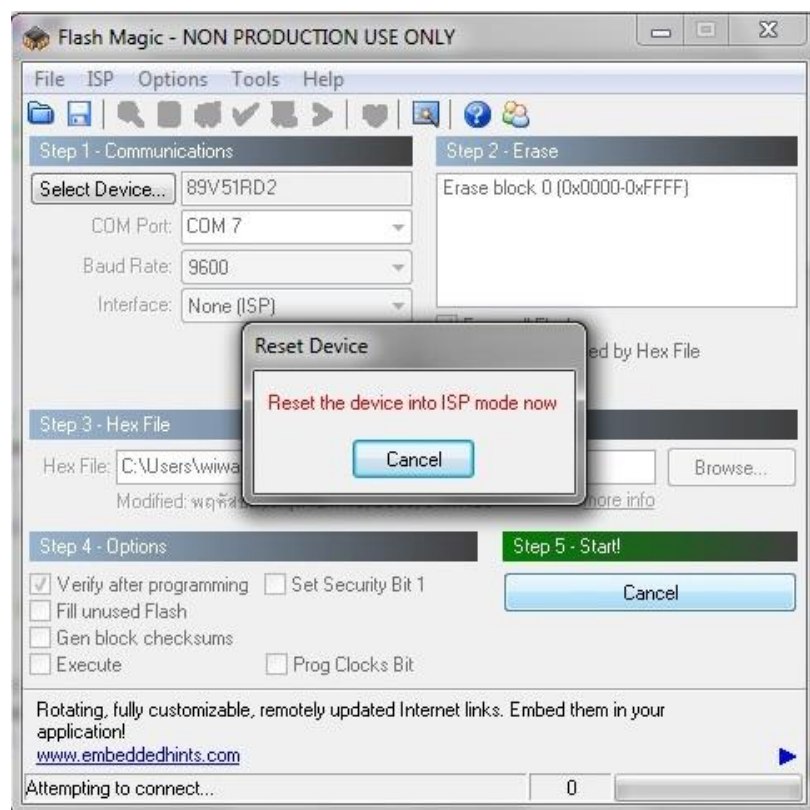
- c. คลิกเลือก Use DTR to control RST ออก



- d. ที่หมายเลข 1 เลือกรุ่นของ MCS-51 และ com port ให้ตรงกับที่เราใช้ ตั้งค่าอื่นๆ ตามหมายเลข 2 และ 4 ตามรูปพร้อมทั้ง Browse ไฟล์ที่หมายเลข 3 เลือก Hex file จากนั้นคลิกเลือก Start ที่หมายเลข 5 แล้ว โปรแกรมจะให้ Reset MCS-51 เพื่อโหลด Hex file ลงอุปกรณ์



- e. กดปุ่ม reset ที่บอร์ดไมโครคอนโทรลเลอร์



การทดลองที่ 3.1

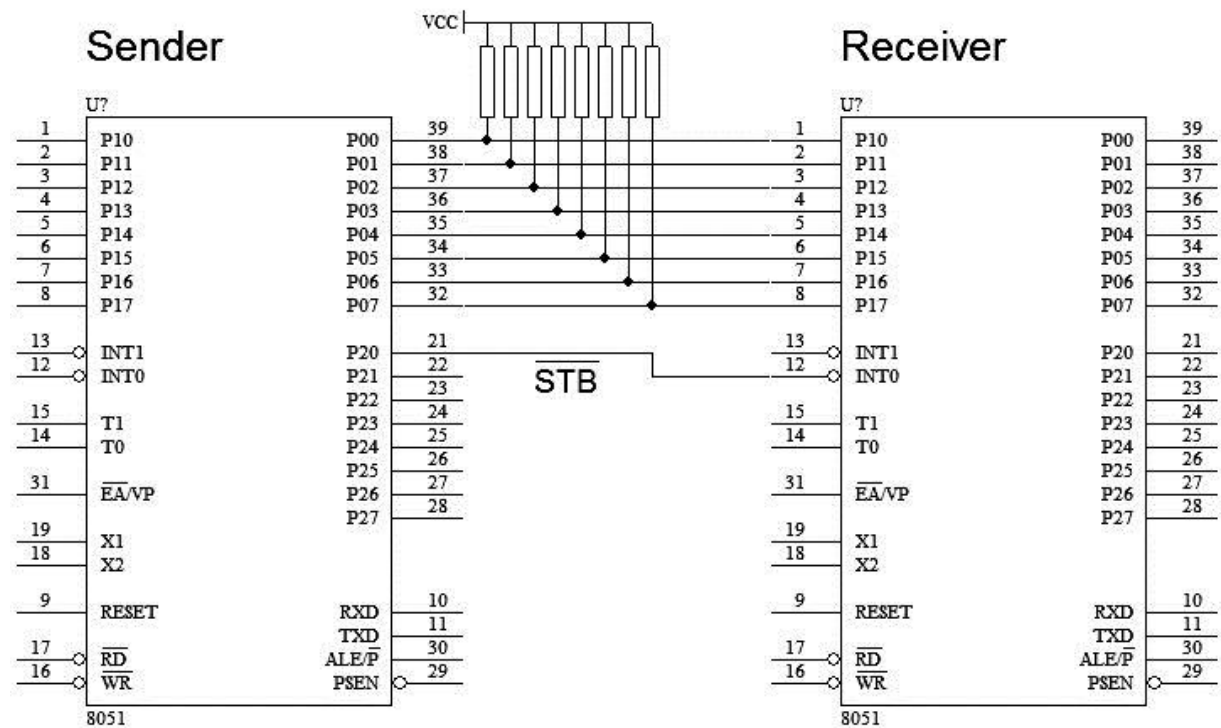
1. ทำการทดลองโดยใช้บอร์ด MCS-51 และ Program Flash magic
2. เมื่อ load hex file ลงในบอร์ด MCS-51 แล้ว ที่ computer เลือก Accessories → communications → Hyper terminal ตั้งชื่อ Lab3 เลือก com1 ใช้ค่า bit/sec 9600 แล้วสั่ง run program
3. เมื่อต้องการแก้ไข program และ load hex file ลงในบอร์ด microcontroller อีกครั้ง ต้อง disconnect Hyper terminal ก่อน เมื่อ load เสร็จแล้วจึง connect Hyper terminal เพื่อ run program

การทดลองที่ 3.2 รับข้อมูลเข้าด้วย serial interrupt

ทำการทดลองเหมือนตอนที่ 3.3 ข้อ

การทดลองที่ 3.3

1. สร้าง lab03s.hex จาก ไฟล์ lab03_3s.c และ load lab03s.hex ลงในบอร์ด MCS-51(1)
2. เอาสาย serial ออกแล้ว
3. สร้าง lab03r.hex จาก ไฟล์ lab03_3r1. c และ load lab03r.hex ลงในบอร์ด MCS-51(2)
4. ต่อ P0 ของบอร์ด MCS-51(1) เข้ากับ P1 ของบอร์ด MCS-51(2)
5. ต่อ port 1.0 ของบอร์ด MCS-51(1) เข้ากับ port 3.2 ของบอร์ด MCS-51(2)
6. กด reset ที่บอร์ด MCS-51 ค้างไว้ เลือก run program บนบอร์ด MCS-51(2) แล้วจึงปล่อยปุ่ม reset ที่บอร์ด MCS-51(1)
7. ทำการทดลอง ซ้ำโดยเปลี่ยนเป็นไฟล์ lab03_3r2.c



**** (P2 ไม่ทำงาน ใช้ P1 แทน ไม่มีเวลาแก้รูป)**

การทดลองที่ 3.1

```
/* Filename    lab03_1.C
```

Description Test Sending message by RS232

Clock 11.0592 Mhz

Compiler Keil CA51 v2.20a

* /

```
#include <reg51.h>
```

```
#include <stdio.h>
```

```
void start232 (void);
```

```
void delay(unsigned int d);
```

```
void wait (void);
```

```
void main()
```

```
{
    start232();

    printf("Microcontroller and Microcomputer Interfacing
    \n\nr");

    SBUF = 'a';

    wait();

    SBUF = 'b';

    wait();

    SBUF = 13;

    wait();

    SBUF = 0x31;
```

<pre> while(1) { } // end while(1) // end main //*****Functions***** void start232 (void) { SCON = 0x52; //Setup serial port control register Mode 1: //8-bit uart var baud rate REN: enable receiver // TI: buffer empty TMOD = 0x20; //Set M1 for 8-bit autoreload timer PCON = 0x80; //Set SMOD bit in PCON //doubles the baud rate TH1 = -6; //Set auto-reload value for timer1 19200 baud with 11.0592 MHz xtal RI = 0; </pre>	<pre> TR1 = 1; // Start timer 1 } void delay(unsigned int d) { unsigned int i,j; for(i=0;i<=d;i++) for(j=0;j<=1000;j++); } void wait (void) { while(TI==0); delay(50); TI=0; } </pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

การทดลองที่ 3.2

<pre> /* Filename lab03_2.C Description Test serial interrupt Clock 11.0592 Mhz Compiler Keil CA51 v2.20a */ </pre>	<pre> #include <reg51.h> #include <stdio.h> #include <absacc.h> #include <stdlib.h> </pre>
-------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------

<pre> #define SEG7LED P0 #define SEGCHNL P1 void dusec (unsigned int count); void start232 (void); void startinterrupt(void); void serial_ISR(void); void serial_ISR_Job(void); int i=0; char dataSEG[]={0x3f,0x06,0x5b,0x4f,0x66,0x6d, 0x7d,0x07,0x7f,0x6f,0x40}; //----- void main() { start232(); startinterrupt(); printf("Test Serial Interrupt \n\r"); while(1) { for(i=0;i<8;i++) { </pre>	<pre> SEG7LED = dataSEG[i]; SEGCHNL = i; dusec(1); } } // end while(1) } // end void main /***** FUNCTIONS *****/ void dusec (unsigned int count) { // mSec Delay unsigned int i; // Keil CA51 (x2) while (count) { i = 10; while (i>0) i--; count--; } } void start232 (void) { SCON = 0x52; // set RS232 parameter TMOD = 0x21; // Crystal = 11059200 TH1 = -6; PCON = 0x80; // speed x 2=19200 // TH1 = -12; PCON = 0x80; // speed x 2=9600 TR1 = 1; } </pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<pre> void startinterrupt(void) { EA = 1; ES = 1; } void serial_ISR(void) interrupt 4 { if(RI) { ES = 0; serial_ISR_Job(); RI = 0; ES = 1; } </pre>	<pre> } void serial_ISR_Job(void) { int i; char mm; for(i=0;i<8;i++) { mm=SBUF; SEG7LED = mm; SEGCHNL = i; dusec(5000); } dusec(1000); } </pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

การทดลองที่ 3.3

<pre> /* Filename lab03_3s.C Description Send data Clock 11.0592 Mhz Compiler Keil CA51 v2.20a */ #include<reg51.h> #include <intrins.h> </pre>	<pre> #include <absacc.h> char send_data[]={0x3f,0x06,0x5b,0x4f,0x66, 0x6d,0x7d,0x07,0x7f,0x6f,0x40}; unsigned int bytecounter; //Data counter sbit strobe=P1^0; /***** BASIC FUNCTION *****/ void delays(unsigned int count2) </pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```

{

TMOD = (TMOD & 0xF0) | 0x01;

/* Set T/C0 Mode */

TR0=1;

while (count2)

{if (TF0==1)

{

TR0=0; /* Stop Timer 0*/

TF0=0;

count2--;

TH0 = 0xD8; /* Load Timer with FFFFH- 10000 */

TL0 = 0xEA;

TR0 = 1; /* Start Timer 0*/

}}}

```

```

/*****/

void main()

{

strobe = 1; //set STB to high

delayms(300); //Wait subroutine

for(bytecounter=0;bytecounter<=10;bytecounter++)

{

strobe = 0; //set STB to low

P0 = send_data[bytecounter]; //Send the data

delayms(300); //Wait subroutine

strobe = 1; //set STB to high

delayms(300); //Wait subroutine

}

} // end of main()

```

```

/* Filename lab03_3r1.C

Description Receive data (Polling)

Clock 11.0592 Mhz

Compiler Keil CA51 v2.20a

*/

#include<reg51.h>

#include<intrins.h>

```

```

#include <absacc.h>

#define SEG7 XBYTE [0xD000]

#define DIGIT XBYTE [0xF000]

char rec_data[11];

unsigned int bytecounter,i; //Data counter

sbit strobe=P3^2;

/*****/

void dusec (unsigned int count) { // mSec Delay

```


<pre> unsigned int j; while (count) { j = 10; while (j>0) j--; count--; } } /*****/ void main() { bytecounter=0; P1=0xff; while(1) { if(strobe==0)// Wait until P3.2 = 0 { if (bytecounter<=10) </pre>	<pre> { rec_data[bytecounter] = P1; //receive the data L1: if (strobe==0){goto L1;} // Wait for sender to finish sending old value bytecounter++; // Next data address to be stored } } //end if for(i=0;i<=5;i++) { SEG7 = rec_data[i]; DIGIT = i; dusec(50); } } // end while } // end of main() </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<pre> /* Filename lab03_3r2.C Description Receive data (Interrupt) Clock 11.0592 Mhz Compiler Keil CA51 v2.20a */ </pre>	<pre> #include<reg51.h> #include <intrins.h> #include <absacc.h> #define SEG7 XBYTE [0xD000] #define DIGIT XBYTE [0xF000] </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------

<pre>char rec_data[11]; unsigned int bytecounter,i; //Data counter /*****/ void int0_ISR (void) interrupt 0 { rec_data[bytecounter] = P1; //receive the data bytecounter++; // Next data address to be stored } /*****/ void dusec (unsigned int count) { // uSec Delay unsigned int j; // Keil CA51 (x2) while (count) { j = 10; while (j>0) j--; count--; } } /*****/</pre>	<pre>void main() { P1=0xff; bytecounter=0; IT0=1; // use edged triggering interrupt IE=0x81; // Enable interrupt for INT 0 while(1) { for(i=0;i<=5;i++) { SEG7 = rec_data[i]; DIGIT = i; dusec(50); }// Other Operations } } // end of main()</pre>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Reference

http://www.keil.com/support/man/docs/c166/c166_libref.htm

การบ้าน

1. จากการทดลอง 3.2 เขียนโปรแกรม เพื่อส่งข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์หนึ่งไปยังบอร์ดไมโครคอนโทรลเลอร์อีกบอร์ดหนึ่งแบบอนุกรมพร้อมแสดงข้อมูลที่ส่งบน 7 segment

