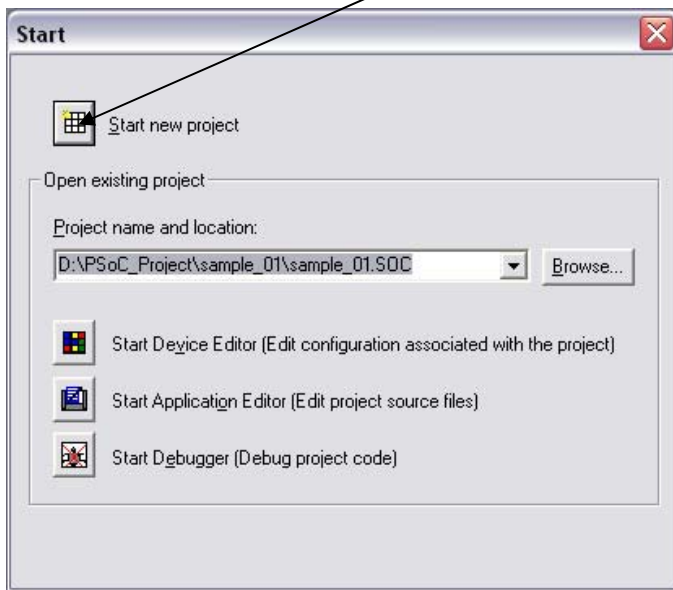


# การทดลองที่ 1

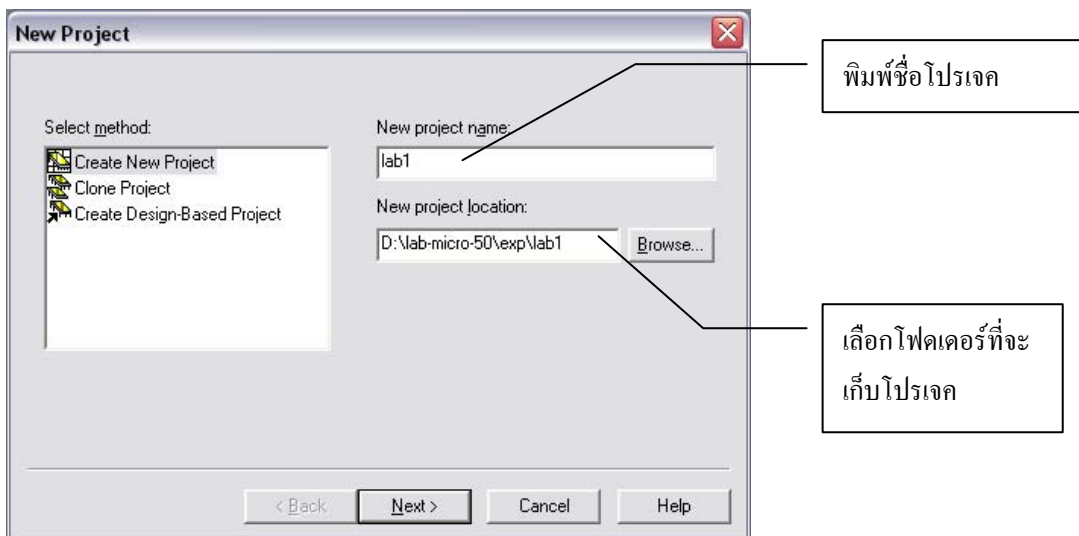
## การเชื่อมต่อไมโครคอนโทรลเลอร์ PSoC กับแอลอีดีและสวิตช์

### (Interfacing the PSoC microcontroller with LED displays and button switches)

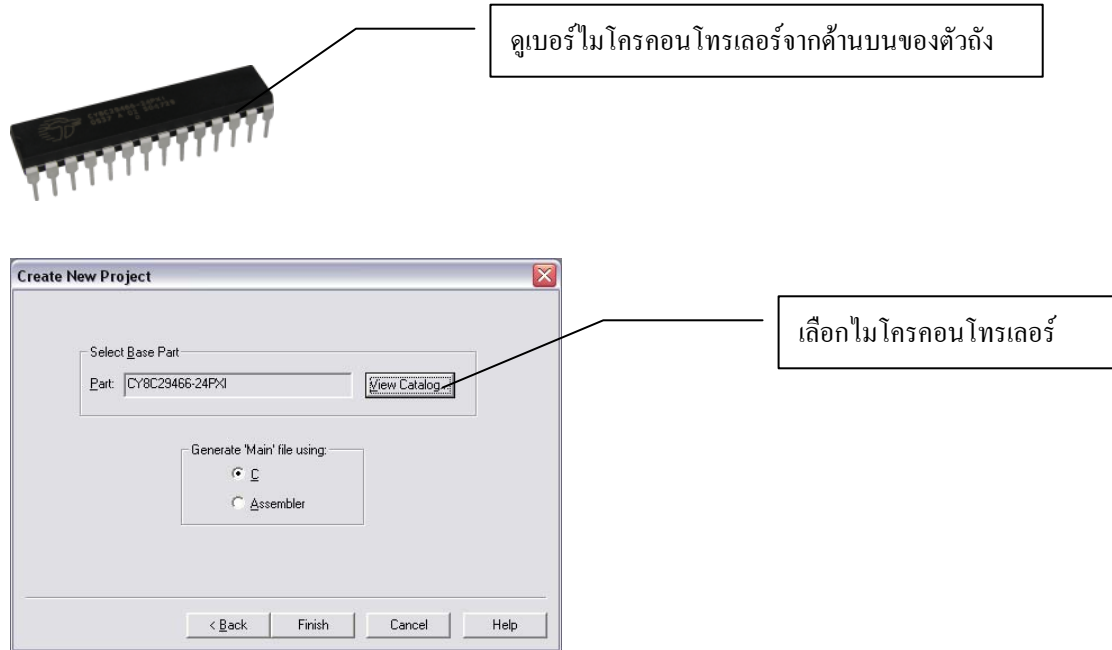
1. รันโปรแกรม PSoC Designer โดยใช้ Start > Program > Cypress MicroSystem > PSoC Designer
2. เมื่อปรากฏหน้าต่าง Start ให้เลือก Start new project




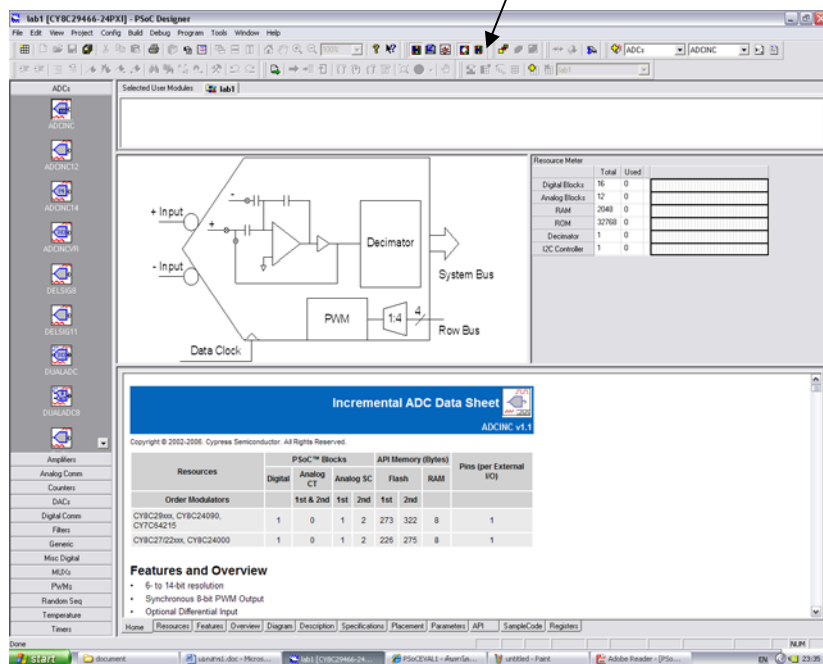
3. ในหน้าต่าง New Project ให้กำหนดโฟลเดอร์ที่จะเก็บโปรเจกต์และตั้งชื่อโปรเจกต์ (จะมีการสร้างโฟลเดอร์ย่อยในโฟลเดอร์ที่ระบุ โดยมีชื่อตามชื่อโปรเจกต์) แล้วคลิกที่ปุ่ม Next



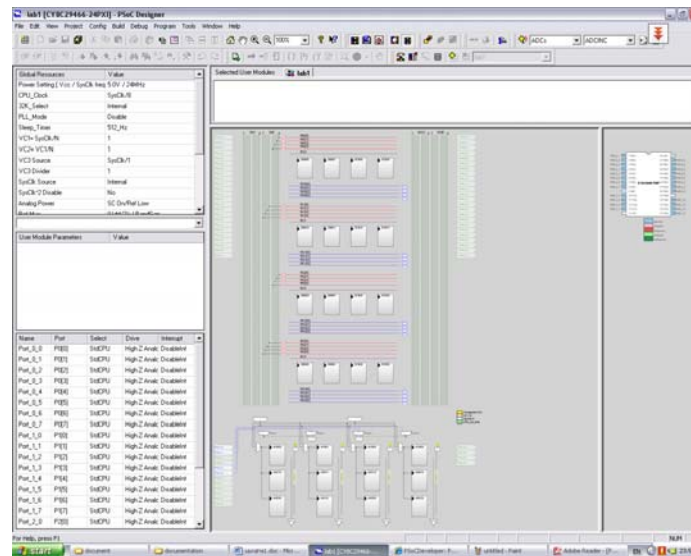
4. ในหน้าต่าง Create New Project ให้เลือก เบอร์ไมโครคอนโทรลเลอร์ ตามที่อยู่ในบอร์ดทดลอง โดยดูจากด้านบนของตัวถัง IC แล้วคลิกปุ่ม View Catalog เพื่อเลือกเบอร์ ในที่นี้ใช้เบอร์ CY8C29466-24PXI ส่วน Main file ให้เลือกเป็นภาษา C ตามรูป แล้วคลิกที่ปุ่ม Finish




5. เมื่อเข้าสู่ PSoC designer ในหน้าแรกจะเป็น User Module Selection View ซึ่งใช้สำหรับการเลือก อุปกรณ์ต่างๆภายในตัวไมโครคอนโทรลเลอร์ แต่สำหรับการทดลองนี้จะยังไม่มีการใช้อุปกรณ์ อื่นๆนอกจากตัวไมโครคอนโทรลเลอร์ ดังนั้นให้คลิก  (Interconnect View) หรือ



6. ในหน้าต่าง Interconnect View นี้ ผู้ใช้สามารถกำหนดค่าพารามิเตอร์ต่างๆของระบบได้ สำหรับการทดลองนี้จะกำหนดเพียงค่า Pin Drive ในส่วน User port



7. การทดลองนี้ จะใช้พอร์ท Port\_1\_0 ถึง Port\_1\_3 สำหรับต่อกับ LED1 ถึง LED4 และ Port\_2\_0 จะต่อกับ Pushbutton switch S1 ที่ต่อขาข้างหนึ่งไว้กับ VCC ดังนั้น Port\_1\_0 ถึง Port\_1\_3 ให้กำหนดค่า Drive เป็น Strong ส่วน Port\_2\_0 กำหนดเป็น Pull Down เมื่อเสร็จแล้วให้กด  (Generate Application) เพื่อสร้างซอร์สไฟล์ต่างๆ

Name	Port	Select	Drive	Interrupt
Port_0_0	P0[0]	StdCPU	High Z Analc	DisableInt
Port_0_1	P0[1]	StdCPU	High Z Analc	DisableInt
Port_0_2	P0[2]	StdCPU	High Z Analc	DisableInt
Port_0_3	P0[3]	StdCPU	High Z Analc	DisableInt
Port_0_4	P0[4]	StdCPU	High Z Analc	DisableInt
Port_0_5	P0[5]	StdCPU	High Z Analc	DisableInt
Port_0_6	P0[6]	StdCPU	High Z Analc	DisableInt
Port_0_7	P0[7]	StdCPU	High Z Analc	DisableInt
Port_1_0	P1[0]	StdCPU	Strong	DisableInt
Port_1_1	P1[1]	StdCPU	Strong	DisableInt
Port_1_2	P1[2]	StdCPU	Strong	DisableInt
Port_1_3	P1[3]	StdCPU	Strong	DisableInt
Port_1_4	P1[4]	StdCPU	High Z Analc	DisableInt
Port_1_5	P1[5]	StdCPU	High Z Analc	DisableInt
Port_1_6	P1[6]	StdCPU	High Z Analc	DisableInt
Port_1_7	P1[7]	StdCPU	High Z Analc	DisableInt
Port_2_0	P2[0]	StdCPU	Pull Down	DisableInt

8. ถ้าการกำหนดค่าไม่ผิดพลาดควรปรากฏข้อความดังนี้ ในแท็บ Results (ตอนล่างของหน้าต่าง)

Starting DRC...

Processing top level rules...


Processing base device rules...

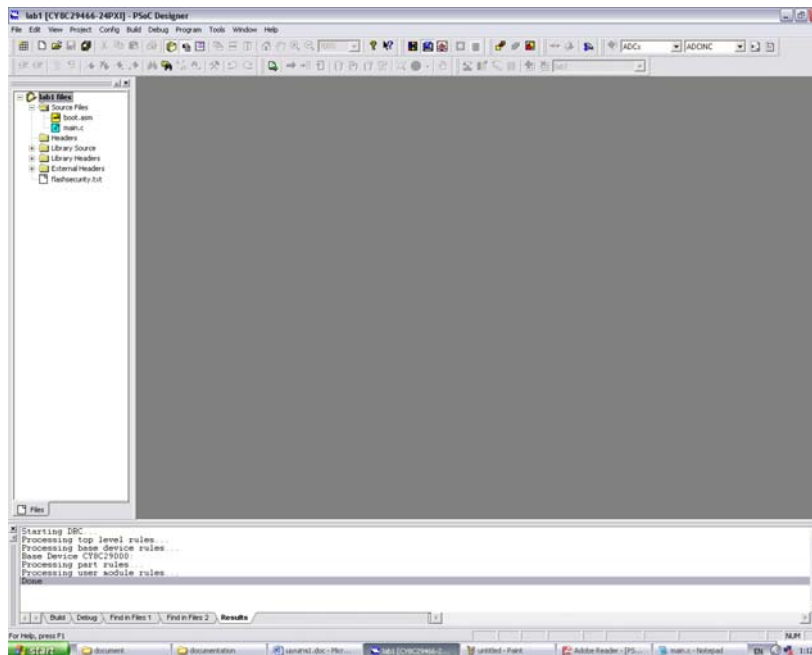
Base Device CY8C29000:

Processing part rules...

Processing user module rules...

Done

ไฟล์ต่างๆ ได้ถูกสร้างขึ้นและพร้อมที่จะให้ปรับแก้ ให้คลิกที่ปุ่ม  (Application Editor) เพื่อเปิดหน้าต่าง Application



9. ให้เปิด Source Files และดับเบิลคลิกที่ main.c เพื่อทำการเขียนโปรแกรม โปรแกรม main.c ที่สร้างโดย PSoC Designer จะมีเพียง โครงร่างที่สำคัญดังนี้


```
//-----  
// C main line  
//-----  
  
#include <msc.h> // part specific constants and macros  
#include "PSOCAPI.h" // PSoC API definitions for all User Modules  
  
void main()  
{  
 // Insert your main routine code here.  
}
```

ให้แก้ไขโปรแกรมดังนี้

โปรแกรมที่ 1.1

```
//-----  
//This project echos the state of port 2 pin 0 to port1 pins 1 and 2  
//  
//When port _2_0 is high,  
// port_1_0 is set low  
// port_1_1 is set high  
//When port_2_0 is low,  
// port_1_0 is set high  
// port_1_1 is set low  
//-----  
  
#include <misc.h> // part specific constants and macros  
#include "PSoCAPI.h" // PSoC API definitions for all User Modules  
  
void main()  
{  
    while (1){  
        if (PRT2DR & 0X01) { //ตรวจสอบค่าที่ Port 2  
            PRT1DR &= 0XFE; // นำค่าที่ Port 1 AND กับค่า FEH แล้วส่งออก Port 1  
            PRT1DR |= 0X02; // นำค่าที่ Port 1 OR กับค่า 02H แล้วส่งออก Port 1  
        }  
        else {  
            PRT1DR |= 0X01;  
            PRT1DR &= 0XFD;  
        }  
    }  
}
```

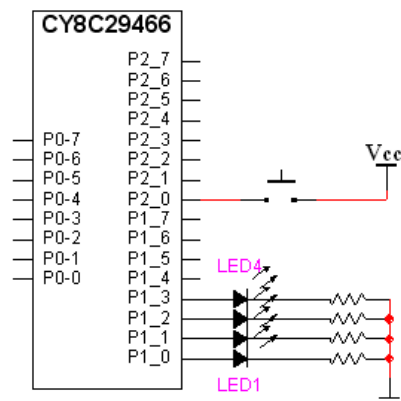
แล้วทำการบันทึกไฟล์

10. ให้ทำการแปลโปรแกรมโดยคลิกที่ปุ่ม  (Build) เพื่อแปลโปรแกรม หากไม่มีข้อผิดพลาดใน Output Window จะปรากฏข้อความ


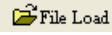

lab1 - 0 error(s) 0 warning(s) 09:46:43

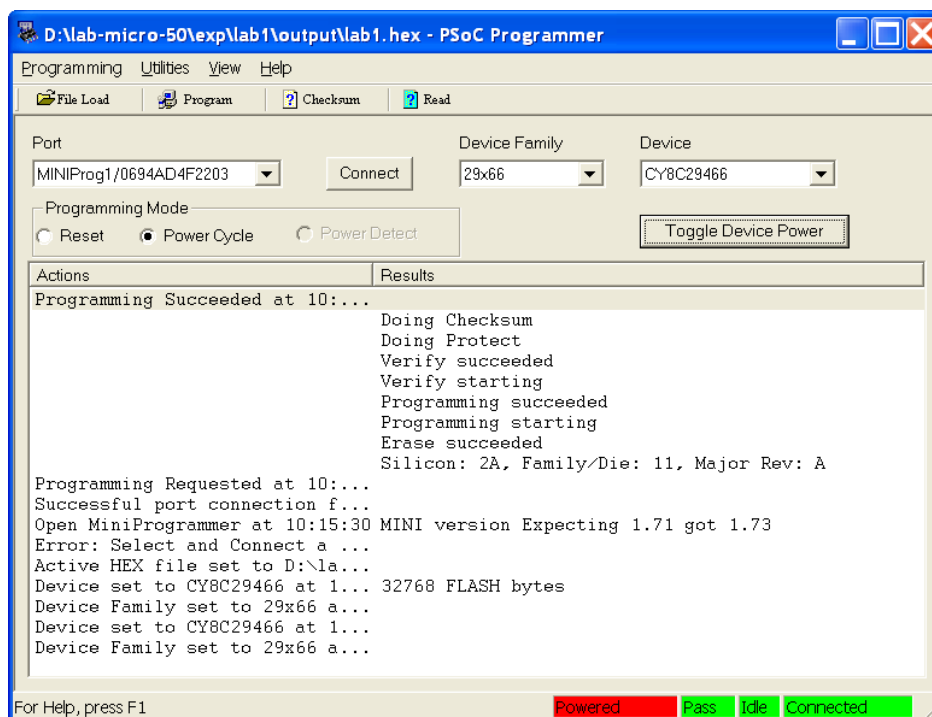
และได้ไฟล์ Hex ที่มีชื่อเหมือนกับชื่อของโปรเจก อยู่ในโฟลเดอร์ Output สำหรับใช้ดาวน์โหลด  
ชิพไมโครคอนโทรลเลอร์

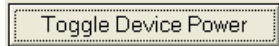
11. การติดตั้งบอร์ด ให้ Jumper JP1 JP2 และ JP3  
ทั้งหมด ไม่ต้องลัดวงจรของขา Jumper เข้าด้วยกัน
12. ให้ต่อวงจรรูป
13. ให้ต่อสาย USB เข้ากับ PSoC MiniProg แล้วเสียบลงในบอร์ดทดลอง ขั้นตอนนี้อธิบายให้ด้วยความ



ถ้า PC เครื่องใดเพิ่งเริ่มทำเป็นครั้งแรก ก็จะถามหาการติดตั้งโปรแกรมสำหรับการเชื่อมต่อแบบ USB ในขั้นตอนนี้ให้เลือกแบบมัติ เมื่อติดตั้งแล้วโปรแกรมก็พร้อมดาวน์โหลดเพื่อทดสอบ

14. การโปรแกรมลงชิพ ให้ใช้คำสั่ง Program Part ในเมนูโปรแกรม Program > Program Part หรือคลิกที่ปุ่ม 
15. เมื่อปรากฏหน้าต่าง PSoC Programmer ให้ เลือก Port กำหนดไฟล์ที่จะดาวน์โหลดโดยคลิกที่ปุ่ม  กำหนด Device Family กำหนด Device แล้วคลิก  เครื่องโปรแกรมก็จะทำการดาวน์โหลดโปรแกรมลงชิพ ในขั้นตอนนี้จะใช้เวลาพอสมควร



16. เมื่อดาวน์โหลดเสร็จ ให้ทำการทดสอบการทำงาน โดยคลิกที่ปุ่ม  เพื่อจ่ายไฟเลี้ยงเข้าบอร์ดทดลอง สังเกตด้านล่างของหน้าต่าง จะปรากฏแถบสีแดงและคำว่า Powered แต่ ถ้าไม่มีไฟเลี้ยงเข้าบอร์ดทดลอง ที่ตำแหน่งนี้จะเป็นสีเขียวและปรากฏคำว่า Not Powered
17. ให้สังเกตการทำงาน ถ้าไม่กดสวิตช์ S1 LED1 จะติดส่วน LED2 จะดับ แต่ถ้ากดสวิตช์ S1 LED1 จะดับส่วน LED2 จะติด ถ้าเป็นเช่นนี้แสดงว่าการทำงานทุกอย่างถูกต้อง

ให้แก้ไขโปรแกรม ให้เป็นไปตามโปรแกรมที่ 1.2 แล้วทำตามขั้นตอนที่ 10 เป็นต้นไปใหม่

## โปรแกรมที่ 1.2

```
//-----  
// C main line  
//-----  
  
#include <m8c.h> // part specific constants and macros  
#include "PSoC_API.h" // PSoC API definitions for all User Modules  
  
//Port Name  
#define Port1 PRT1DR  
  
void main()  
{  
    BYTE loop=1;  
    WORD i;  
    while(1)  
    {  
        Port1=loop;  
        loop<<=1;  
        if (loop==16) loop=1;  
        for (i=1;i<10000;i++)  
        {  
            asm("nop"); // 4 Cycle  
            asm("nop"); // 4 Cycle  
            asm("nop"); // 4 Cycle  
            asm("nop"); // 4 Cycle  
        }  
    }  
}
```

### คำถาม

1. แต่ละการทดลองมีการทำผิดพลาดอะไรบ้าง
2. ในการทดลองโปรแกรมที่ 1.2 ได้ผลลัพธ์เป็นเช่นไร
3. ให้เขียนโปรแกรมเกี่ยวกับการใช้พอร์ตอินพุทเอาต์พุท มา 1 โปรแกรม

ตัวอย่างไอคอน

Icon	Option	Menu Path	Shortcut	Feature
	Compile/Assemble	Build > Compile/Assemble	[Ctrl] [F7]	Compiles/assembles the most prominent open, active file (.c or .asm)
	Build	Build > Build	[F7]	Builds entire project and links applicable files
	Execute Program			Switches into Debugging subsystem, connects, downloads file, runs... all from one click
	New File	File > New	[Ctrl] [N]	Adds a new file to the project
	Open File	File > Open	[Ctrl] [O]	Opens an existing file in the project
	Indent			Indents specified text
	Outdent			Outdents specified text
	Comment			Comments selected text
	Uncomment			Uncomments selected text
	Toggle Bookmark			Toggles the bookmark: Sets/removes user-defined bookmarks used to navigate source files
	Clear Bookmark			Clears all user-defined bookmarks
	Next Bookmark			Goes to next bookmark
	Previous Bookmark			Goes to previous bookmark
	Find Text	Edit > Find	[Ctrl] [F]	Find specified text
	Replace Text	Edit > Replace	[Ctrl] [H]	Replace specified text
	Find in Files	Edit > Find in Files		Find specified text in specified file(s)
	Repeat Find			Repeats last find
	Set Editor Options			Set options for editor
	Undo	Edit > Undo	[Ctrl] [Z]	Undo last action
	Redo	Edit > Redo	[Ctrl] [Y]	Redo last action