

Lab 2 : การเขียนโปรแกรมภาษาซีกับชิพ AVR

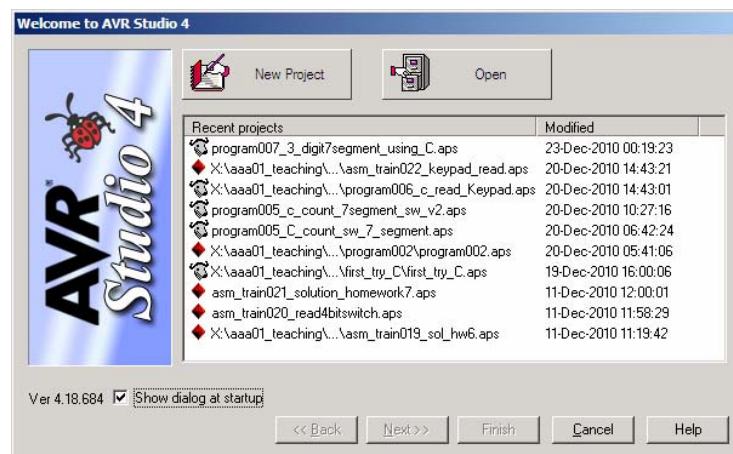
2.1 แนะนำการใช้ซอฟต์แวร์ AVRStudio พัฒนาโปรแกรมด้วยภาษาซี

โปรแกรม AVRStudio เป็นซอฟต์แวร์ที่พัฒนาโดย ATMEAL ซึ่งแจกจ่ายให้ใช้งานได้ฟรี นอกจากนี้จะสามารถใช้พัฒนาโปรแกรมภาษาแอสเซมบลีได้แล้ว ยังสามารถใช้พัฒนาซอฟต์แวร์ด้วยภาษาซีได้ด้วย โดยจะทำงานร่วมกับโปรแกรม WinAVR (เป็นซอฟต์แวร์อีกตัวหนึ่งซึ่งเป็นฟรีแวร์) โดยใช้สภาพแวดล้อม AVRStudio เป็นหลักในการทำงาน

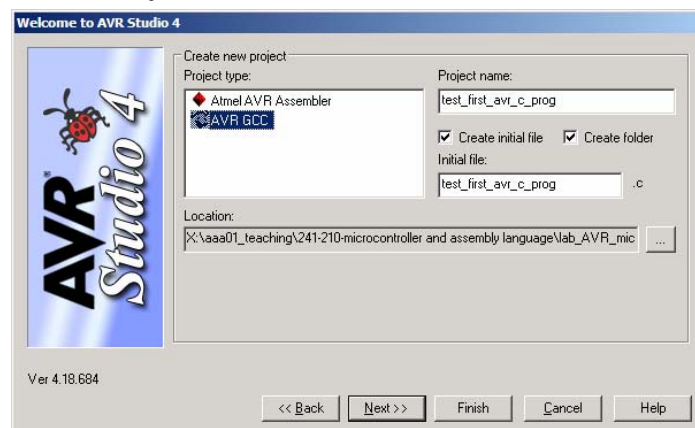
เรียกใช้งาน โปรแกรมจาก Start menu->All Programs->ATmel AVR Tools->AVR Studio 4



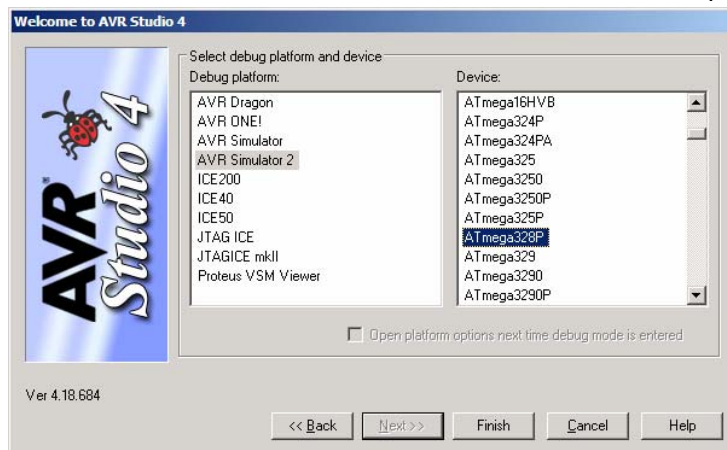
โปรแกรมจะแสดงหน้าต่าง ดังรูป ให้เลือกกดปุ่ม New Project



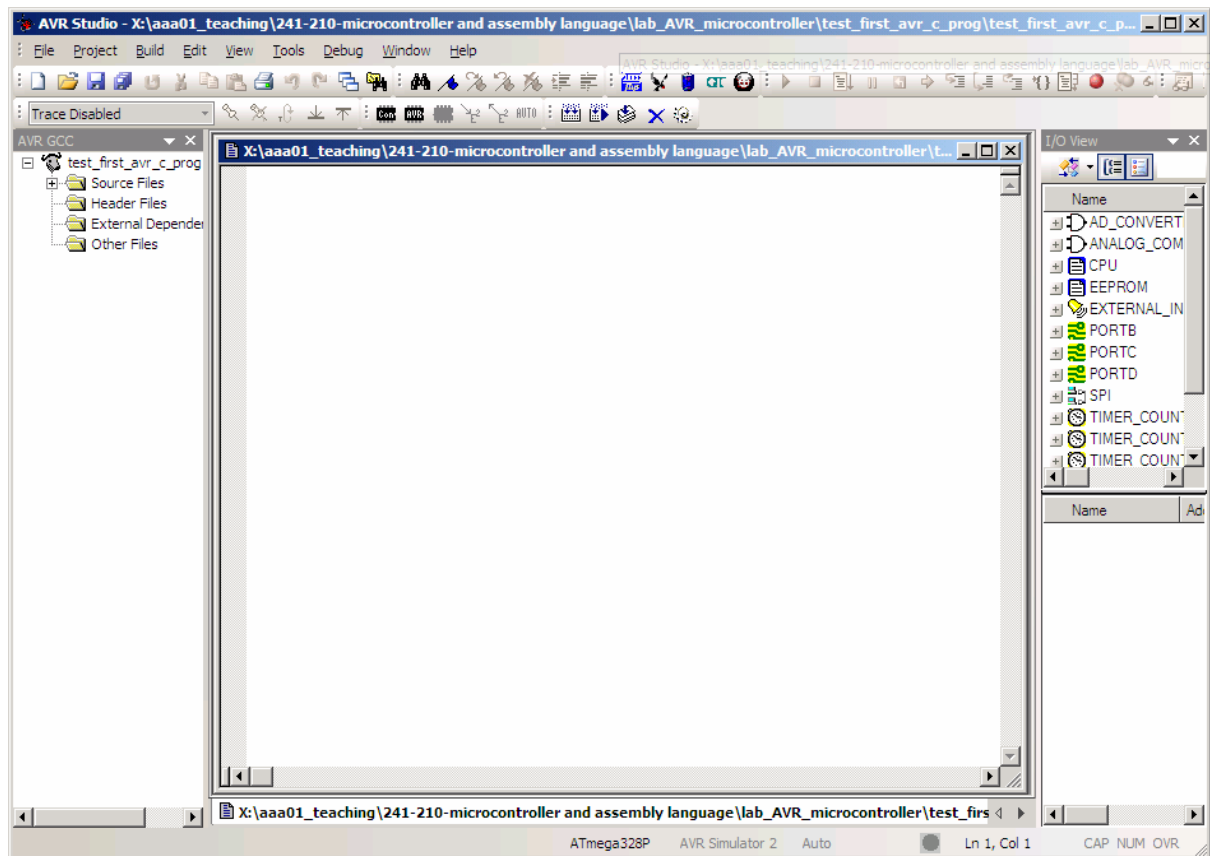
เลือกสร้างโปรเจกต์ด้วยภาษาซี ดังรูป โดยเลือกชนิดของ Project เป็น AVR GCC ป้อนชื่อโปรเจกต์ลงไป



เลือกซีพียูรุ่น ATmega328P และสภาพแวดล้อม AVR Simulator2 จากนั้นกดปุ่ม Finish





จะขึ้นสภาพแวดล้อมการพัฒนาโปรแกรมดังรูป

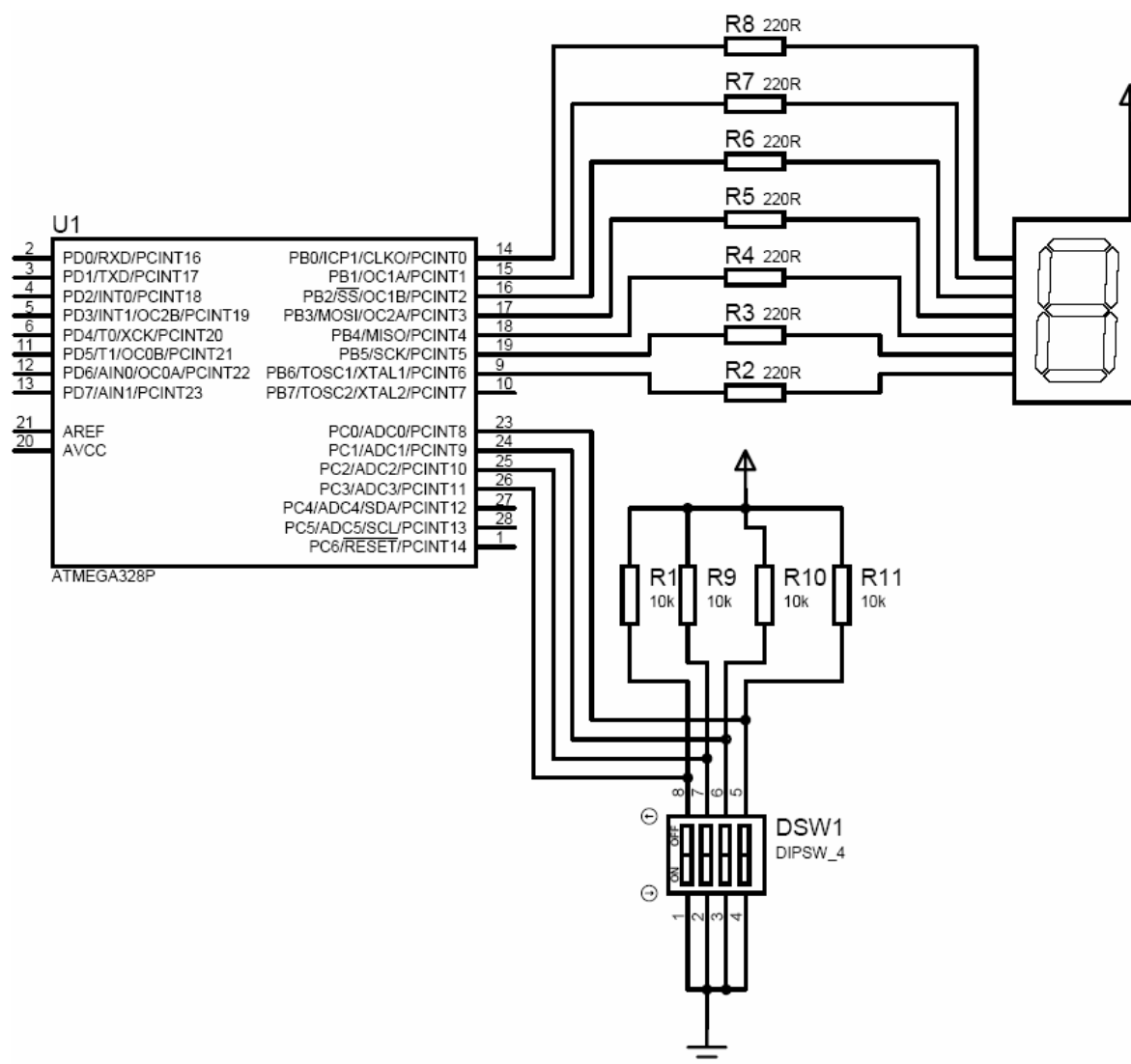


2.2 การทดลอง

2.2.1 การออกแบบวงจรอ่านค่าจากสวิตช์และแสดงผลออกทาง 7-segment LED

ให้นักศึกษาเปิดโปรแกรม ISIS ในชุดซอฟต์แวร์ Proteus คลิกเลือก New Design เลือกขนาดของกระดาษเป็น Landscape A4 ทำการวาดวงจรเพื่อทำการอ่านค่าจากสวิตช์ดังรูปที่ 2.1 กดปุ่ม  จากนั้นดับเบิลคลิกที่  เพื่อเลือกใช้อุปกรณ์จากไลบรารีต่อไปนี้

Category	Sub-Category	Device	คำอธิบาย
Microprocessor ICs	AVR Family	ATMEGA328P	ไมโครคอนโทรลเลอร์ AVR
Optoelectronics	7-Segment Displays	7-SEG-COM-CAT	ชนิด Common Cathode
Switches & Relays	Switches	DIPSW-4	Dip Switch 4 ตัวใน 1 package
Resistors	Generic	RES	ตัวต้านทาน



รูปที่ 2.1 วงจรอ่านค่าจากสวิตช์แสดงผลทาง 7-segment LED

2.2.2 การเขียนโปรแกรมภาษาซีเพื่ออ่านค่าจากสวิตช์แสดงผลทาง 7-segment LED

เขียนโปรแกรมภาษาซีเพื่อใช้ควบคุมไมโครคอนโทรลเลอร์ของวงจรในรูปที่ 2.1 โดยตัวโปรแกรมแสดงให้เห็นในรูปที่ 2.2 การทำงานของตัวโปรแกรม จะทำการอ่านค่าจากคิปสวิตช์ขนาด 4 บิตเข้ามาทางพอร์ต C บิตที่ 0-3 และทำการแปลงค่าไบนารีที่ได้ ซึ่งมีค่า 0-15 ไปแสดงผลทางแอลอีดี 7 เซกเมนต์ ค่า 0-F ให้นักศึกษาเขียนโปรแกรมภาษาซีด้วย AVR Studio จากนั้นใช้คำสั่ง Build เพื่อคอมไพล์ให้เป็นภาษาเครื่องของ AVR ซึ่งจะได้ไฟล์เอาต์พุตนามสกุล .HEX

```
#include <avr/io.h>
int main(void)
{
    DDRC = 0x00;    // set port C as input
    DDRB = 0xFF;    // set port B as output

    unsigned char SWITCH_V, DISP;
    unsigned char LOOKUPTB[] = { 0b00111111,
                                0b00000110,
                                0b01011011,
                                0b01001111,
                                0b01100110,
                                0b01101101,
                                0b01111101,
                                0b00000111,
                                0b01111111,
                                0b01101111,
                                0b01110111,
                                0b01111100,
                                0b00111001,
                                0b01011110,
                                0b01111001,
                                0b01110001 };

    while (1)
    {
        //---read input switch via portB
        SWITCH_V = PINC;
        SWITCH_V &= 0x0F; //upper 4 bits are masked
        DISP = LOOKUPTB[SWITCH_V];
        PORTB = ~(DISP);
    }
}
```

รูปที่ 2.2 โปรแกรมภาษาซีสำหรับอ่านค่าจากสวิตช์และแสดงผลทางแอลอีดี 7 เซกเมนต์

รายการ	ลายเซ็น	วัน-เดือน-ปี
#checkpoint1 ทดสอบจำลองการทำงานของโปรแกรมที่เขียนขึ้นด้วยภาษาซีกับวงจรไมโครคอนโทรลเลอร์ในโปรแกรม ISIS7.0		
#checkpoint2 เปรียบเทียบขนาดของโปรแกรมภาษาเครื่องที่ได้จากการเขียนด้วยภาษาแอสเซมบลีในการทดลองที่ 1 (รูป 1.8) และโปรแกรมภาษาซีในรูปที่ 1.2 ขนาดของโปรแกรมภาษาเครื่อง - พัฒนาด้วยภาษาแอสเซมบลี..... - พัฒนาด้วยภาษาซี.....		

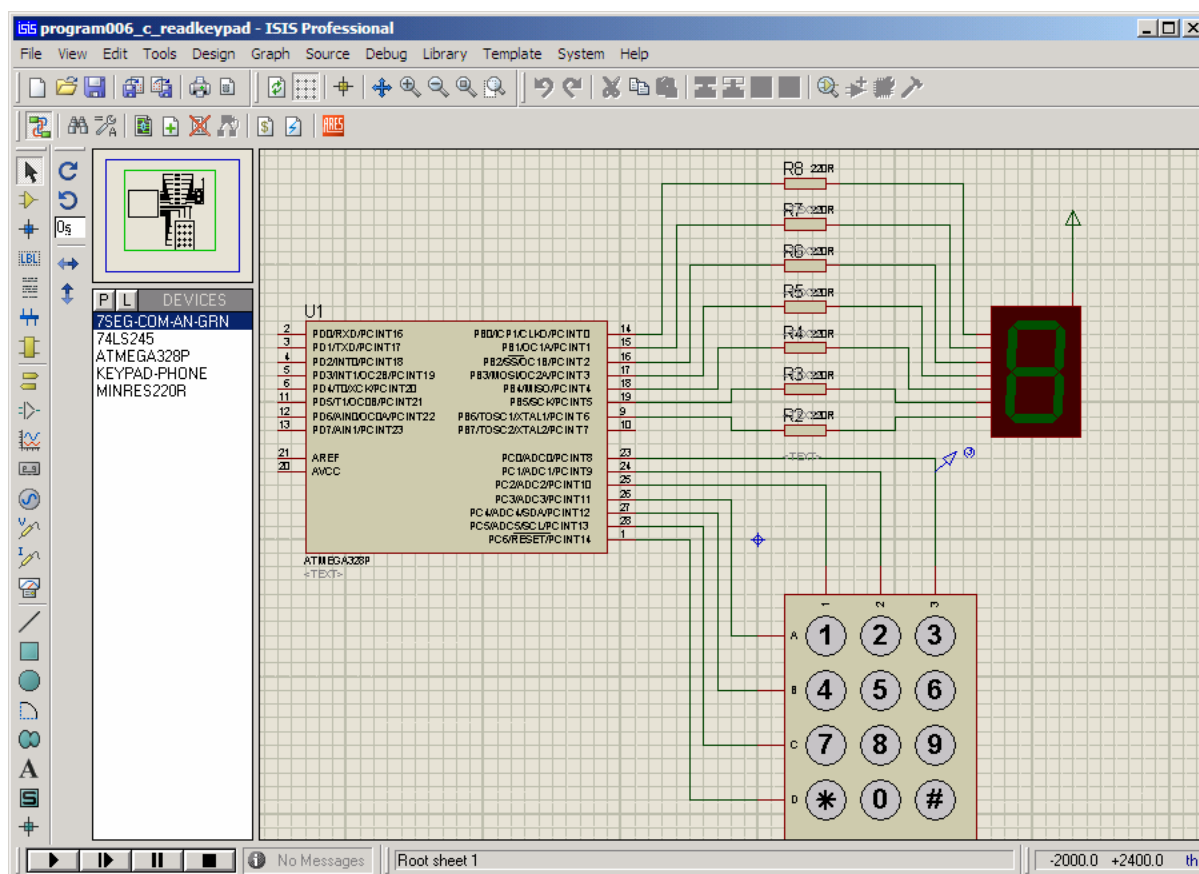
2.2.3 จงออกแบบวงจรและเขียนโปรแกรมควบคุมด้วยภาษาซี#1

จงออกแบบวงจรสำหรับการอ่านค่าจากสวิตช์ 8 ตัว และแสดงผลจำนวนบิตที่เป็นลอจิก 1 ออกทางแอลอีดี 7-segment พร้อมทั้งเขียนโปรแกรมควบคุมด้วยภาษาซี จากนั้นทดสอบความถูกต้องของโปรแกรมด้วยการจำลองการทำงานในโปรแกรม ISIS ของซอฟต์แวร์ชุด Proteus

รายการ	ลายเซ็น	วัน-เดือน-ปี
#checkpoint3 ทดสอบจำลองการทำงานของโปรแกรมที่เขียนขึ้นด้วยภาษาซีกับวงจรไมโครคอนโทรลเลอร์ในโปรแกรม ISIS7.0		

2.2.4 จงออกแบบวงจรและเขียนโปรแกรมควบคุมด้วยภาษาซี#2

จากวงจรดังรูปที่ 2.3 จงเขียนโปรแกรมเพื่อทดสอบการกดสวิตช์ Keypad ด้วยภาษาซี จากนั้นทดสอบความถูกต้องของโปรแกรมด้วยการจำลองการทำงานในโปรแกรม ISIS ของซอฟต์แวร์ชุด Proteus โดยสมมติว่าการกดสวิตช์ไม่มี Bounce เกิดขึ้น (ไม่ต้องทำ Debouncing)



รูปที่ 2.3 วงจรอ่านสวิตช์ Keypad และแสดงผลการกดออกทาง 7-segment

รายการ	ลายเซ็น	วัน-เดือน-ปี
#checkpoint4 ทดสอบจำลองการทำงานของโปรแกรมที่เขียนขึ้นด้วยภาษาซีกับ วงจรไมโครคอนโทรลเลอร์ในโปรแกรม ISIS7.0		
