

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____

วันเดือนปี _____/_____/_____

รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

การทดลองที่ 7 การประยุกต์ใช้งานไมโครคอนโทรลเลอร์กับวงจรอิเล็กทรอนิกส์

วัตถุประสงค์การทดลอง

1. เพื่อศึกษาบอร์ดไมโครคอนโทรลเลอร์ Arduino
2. เพื่อศึกษาการเชื่อมต่อ Arduino กับวงจรอิเล็กทรอนิกส์
3. เพื่อศึกษาการประยุกต์ใช้ทรานซิสเตอร์



อุปกรณ์ และเครื่องมือที่ใช้ในการทดลอง

ลำดับ	อุปกรณ์	จำนวน
1	ดิจิตอลมัลติมิเตอร์	1
2	เครื่องคอมพิวเตอร์ Notebook (ให้นักศึกษานำมาเอง)	1
3	ความต้านทาน 220 Ω	1
4	ความต้านทาน 10 k Ω	6
5	หลอด LED	6
6	สวิตช์แบบกดติดปลายดัด	1
7	Potentiometer 10K	1
8	LM35	1

7.1 บอร์ด Arduino

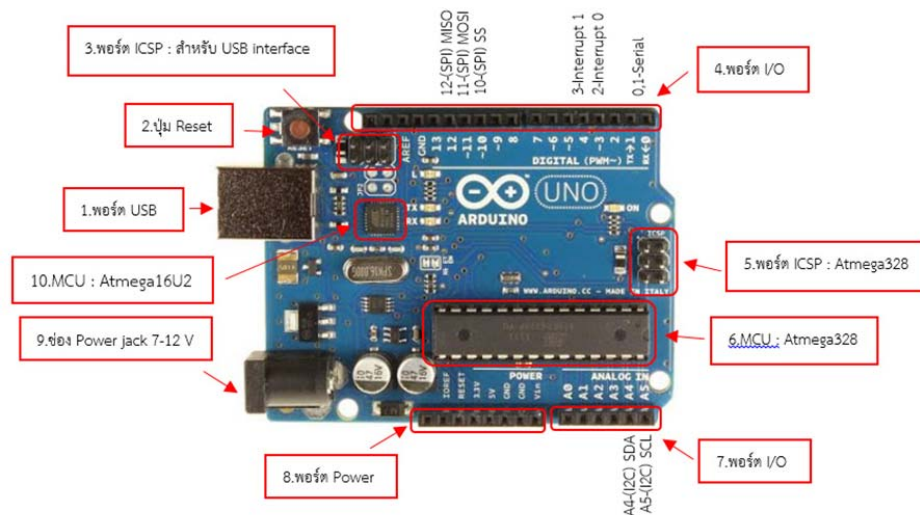
ไมโครคอนโทรลเลอร์ (MCU) เป็นไมโครโพรเซสเซอร์ชนิดหนึ่งที่มีทั้งหน่วยความจำ (ROM/RAM) และ I/O อยู่ในตัวไมโครโพรเซสเซอร์ ทำให้การนำไปใช้งานเชื่อมต่อกับอุปกรณ์ภายนอกสามารถทำได้ง่าย เพราะไม่ต้องต่อกับหน่วยความจำภายนอก และ I/O Controller เพิ่มเติมอีก ไมโคร คอนโทรลเลอร์ที่นิยมใช้มีหลายตระกูล เช่น PIC, AVR, ARM

Arduino เป็นบอร์ดที่ใช้ไมโครคอนโทรลเลอร์ตระกูล AVR มาพัฒนา โดยมีจุดเด่น คือ พัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software นอกจากนั้นบอร์ด Arduino ยังออกแบบมาให้ใช้งานได้ง่าย ใช้ภาษา C ในการใช้งาน นอกจากนั้นยังมีไลบรารีจำนวนมากให้เรียกใช้งาน ทำให้ใช้งานได้ง่ายขึ้น นอกจากนั้นยังมีราคาถูก ทำให้แพร่หลายและนิยมใช้กันมาก

ปัจจุบันคำว่า Arduino มีความหมายกว้างออกไป โดยได้กลายเป็นชื่อเรียก Platform โดยมีการพัฒนาบอร์ดในหลากหลายรูปแบบ หลายขนาด เพื่อให้เลือกใช้งานตามความต้องการ โดยบอร์ดแต่ละรุ่น จะมีขนาดความจุของหน่วยความจำ ความเร็วของไมโครคอนโทรลเลอร์ และจำนวนขาที่สามารถเชื่อมต่ออุปกรณ์แตกต่างกันไป

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____
รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

บอร์ด Arduino ที่ใช้ในการทดลองนี้เป็นรุ่น UNO R3 ซึ่งใช้ชิพ ATmega328 เป็นไมโครคอนโทรลเลอร์ 8 บิต ทำงานที่ความถี่ 16 MHz มีหน่วยความจำแฟลช (ROM) 32 KB และมีหน่วยความจำแรม 2 KB มีขา I/O แบบดิจิตอลจำนวน 14 ขา (สามารถใช้งานในรูปแบบ PWM ได้ 6 ขา) มีขา I/O แบบแอนะล็อก 6 ขา (ใช้ ADC 10 บิต) มีพอร์ตอนุกรม (Serial UART 1) จำนวน 1 ชุด พอร์ตอนุกรม I2C จำนวน 1 ชุด และพอร์ตอนุกรม SPI จำนวน 1 ชุด



รูปที่ 1 Arduino UNO R3 (ภาพจาก thaieasyelec)

รูปที่ 1 แสดงส่วนประกอบต่างๆ ของ Arduino UNO R3 โดยมีรายละเอียดดังนี้

- 1) พอร์ต USB ใช้เชื่อมต่อกับเครื่องคอมพิวเตอร์ผ่านพอร์ต USB และใช้เป็นแหล่งจ่ายไฟของบอร์ดได้
- 2) ปุ่ม Reset ใช้สำหรับเริ่มการทำงานใหม่
- 3) พอร์ต ICSP (บอร์ด UNO จะมี MCU อีกตัวหนึ่ง คือ Atmega16U2 โดยทำหน้าที่เป็นตัวเชื่อมต่อ USB โดยสามารถจำลองเป็นพอร์ตอนุกรม (Visual Com port) ได้ โดยพอร์ต ICSP จะใช้ในการโปรแกรม MCU Atmega16U2)
- 4) พอร์ต I/O แบบดิจิตอล (บางพอร์ตของ Arduino สามารถใช้งานได้หลายหน้าที่)
- 5) พอร์ต ICSP สำหรับใช้ในการโปรแกรม ROM ATmega328
- 6) MCU ATmega328
- 7) พอร์ต I/O แบบแอนะล็อก
- 8) พอร์ตสำหรับจ่ายไฟ โดยสามารถจ่ายได้ทั้ง +5v และ +3.3v
- 9) ช่อง Power Jack 7-12 V ใช้สำหรับจ่ายไฟเข้าบอร์ด กรณีที่ไม่ได้ต่อกับ USB
- 10) MCU Atmega16U2

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____

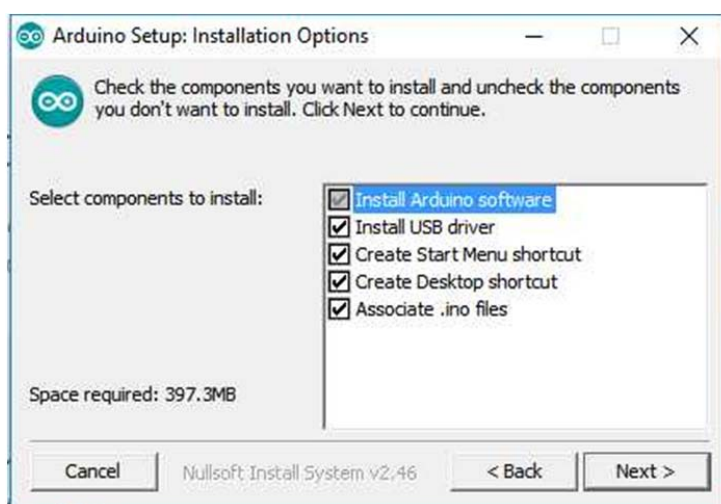
วันเดือนปี _____/_____/_____

รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

7.2 การติดตั้ง Arduino IDE

โปรแกรม Arduino IDE เป็นโปรแกรมที่ใช้ในการเขียนโปรแกรม เพื่อดาวน์โหลดโปรแกรมลงบอร์ด Arduino การติดตั้งโปรแกรม Arduino IDE มีขั้นตอนดังนี้

- 1) ดาวน์โหลดโปรแกรม Arduino IDE ที่ <https://arduino.cc/en/Main/Software> และติดตั้งลงในเครื่องคอมพิวเตอร์
- 2) ในหน้า Installation Options ให้เลือกติดตั้งทั้งหมด รวมทั้ง USB driver ตามรูปที่ 2



รูปที่ 2 การติดตั้ง Arduino IDE

- 3) เมื่อติดตั้งเสร็จจะปรากฏหน้าต่างดังนี้

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____
วันเดือนปี _____ / _____ / _____
รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

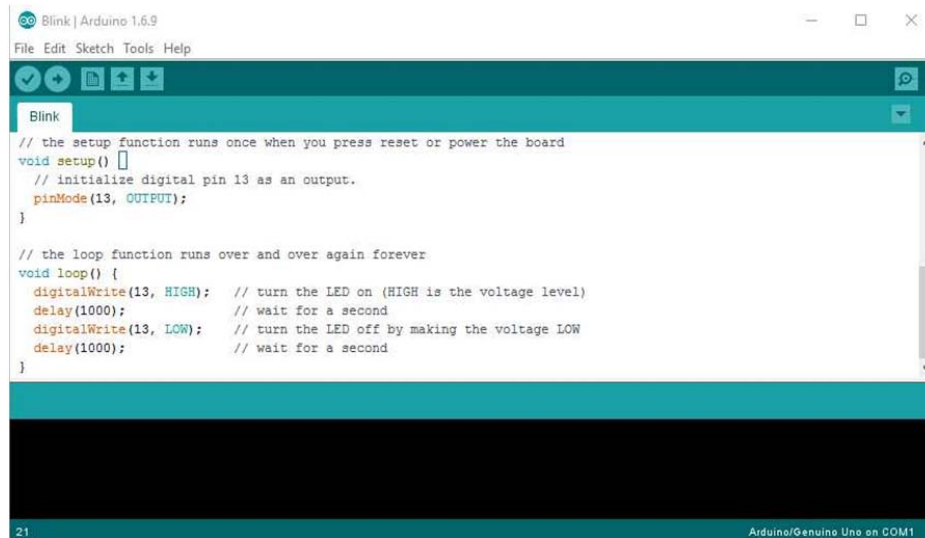


รูปที่ 3 หน้าจอหลังการติดตั้ง Arduino IDE

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____
รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

การทดลองที่ 1 ไฟกระพริบ

- 1) เชื่อมต่อบอร์ดกับ Arduino ผ่านสาย USB
- 2) ที่ Arduino IDE ให้โหลดโปรแกรมตัวอย่างมารัน โดยเลือกที่ File -> Examples -> 01.Basics -> Blink โปรแกรมจะแสดงหน้าจอ



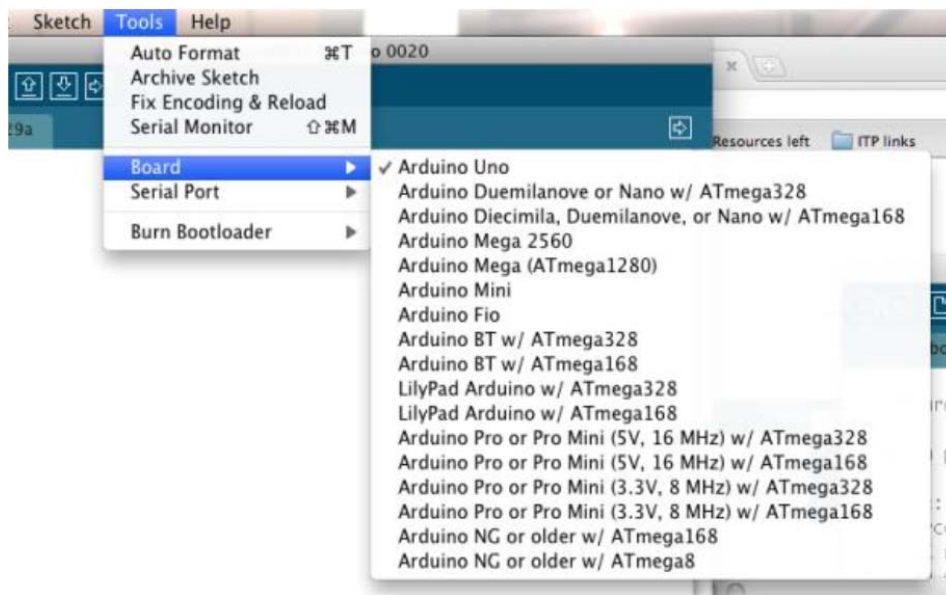
รูปที่ 3 หน้าจอ Arduino IDE ที่แสดงโปรแกรม Blink

- 3) ทำการเลือกบอร์ดโดยเลือกที่ Tools -> Board -> Arduino Uno



กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____

วันเดือนปี _____/_____/_____

รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____



รูปที่ 4 หน้าจอ Arduino IDE ที่แสดงการเลือกบอร์ด

- 4) ทำการเลือกพอร์ตโดยเลือก Tools -> Serial Port -> COMx (x เป็นหมายเลขพอร์ตที่ได้จากการติดตั้ง USB Driver โดยให้ดูจาก Device Manager)
- 5) กดที่ปุ่ม  เพื่อ Compile โปรแกรม หากโปรแกรมไม่มีข้อผิดพลาดให้กดที่  เพื่อดาวน์โหลดโปรแกรมลงบอร์ด หากโปรแกรมแสดงว่า Done Uploading. แสดงว่าโปรแกรมสามารถ Upload เรียบร้อยแล้ว ซึ่งจะเห็น LED กระพริบบนบอร์ด
- 6) หากดูที่โปรแกรมใน Arduino IDE จะพบว่าโปรแกรมแบ่งออกเป็น 2 ส่วน คือ ส่วน setup() และ loop()
 - ส่วน setup() จะเป็นส่วนของโปรแกรมที่จะรันเพียงครั้งเดียว ตอนเริ่มต้นโปรแกรม ดังนั้นจะใช้สำหรับการ setup ของเรา เช่น pinMode(13, OUTPUT); หมายถึงกำหนดให้ขา 13 ทำหน้าที่เป็น OUTPUT (ขา 13 จะต่อกับ LED บนบอร์ด)
 - ส่วน loop() เป็นส่วนของโปรแกรมที่จะรันไปเรื่อยๆ โดยเมื่อทำงานจบรอบแล้ว ก็จะมาเริ่มต้นทำใหม่อีกครั้ง โดย digitalWrite(13, HIGH) หมายถึงให้ set logic high ที่ขา 13 ซึ่งเป็นขา LED ซึ่งจะมีผลให้ LED ติด จากนั้นจะ delay 1000ms (1 วินาที) แล้ว set logic low ทำให้ LEDดับ LED จึงกระพริบไปเรื่อยๆ
- 7) ให้นักศึกษาแก้ไขโปรแกรม โดยให้จังหวะการกระพริบ เป็น Pattern ตามที่นักศึกษากำหนด จากนั้นให้อาจารย์ตรวจ

ลายเซ็น.....

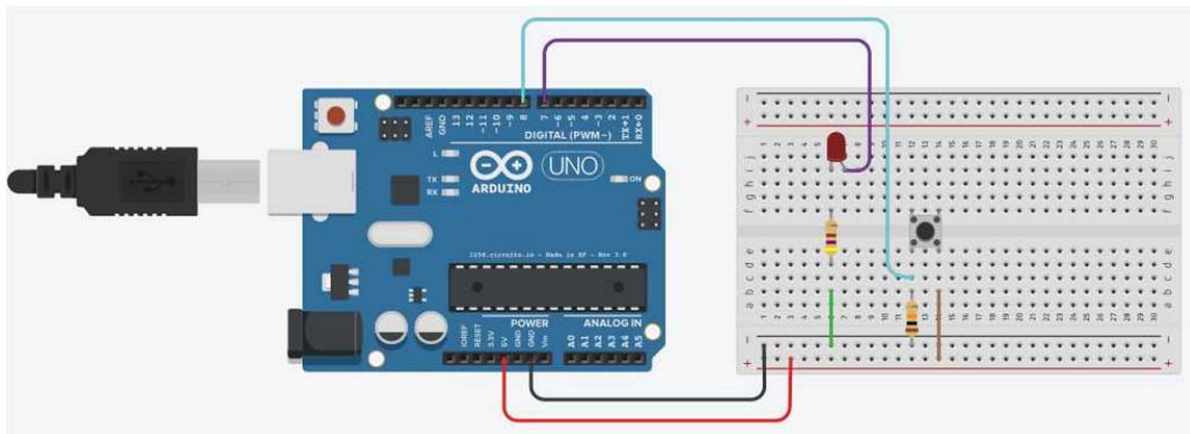
การทดลองที่ 2 อินพุตและ LED (Digital I/O)

- 1) ให้ต่อวงจรตามรูป (ใช้ R 220 Ω ต่อกับ LED และ R 10 k Ω ต่อกับสวิตช์)

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____

วันเดือนปี _____/_____/_____

รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____



รูปที่ 5การต่อ Arduino กับ LED และ Switch

2) ให้เขียนโปรแกรมต่อไปนี้ จากนั้นโหลดลงบอร์ด แล้วทดสอบการทำงาน

```
1) const int BUTTON = 8;      //the number of the pushbutton pin
2) const int LED = 7;         //the number of the LED pin
3) int buttonState = 0;
4) int state = 0;
5)
6) void setup() {
7)   //put your setup code here, to run once:
8)   //initialize the LED pin as an output:
9)   pinMode(LED, OUTPUT);
10)  //initialize the pushbutton pin as an input:
11)  pinMode(BUTTON, INPUT);
12) }

13) void loop() {
14)  //put your main code here, to run repeatedly:
15)  buttonState = digitalRead(BUTTON);
16)
17)  if (buttonState == HIGH) {
18)    //turn LED on:
19)    state = 1 - state;
20)  }
21)
22)  if (state == 1) {
23)    digitalWrite(LED, HIGH);
24)  } else {
25)    //turn LED off:
26)    digitalWrite(LED, LOW);
27)  }
28) }
```

จงตอบคำถามต่อไปนี้

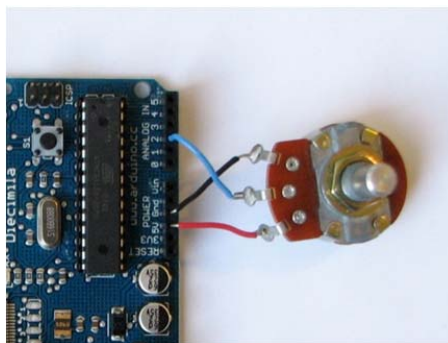
กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____
วันเดือนปี _____/_____/_____
รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

1.1 กระแสที่มากที่สุดที่ LED สามารถรับได้ คือ 20 mA ที่ 2.0v ให้นักศึกษาคำนวณค่า R ต่ำที่สุดที่สามารถใช้กับวงจรนี้ได้

1.2 ให้อธิบายว่าเหตุใดจึงต้องต่อ R 10 k Ω กับสวิตช์

การทดลองที่ 3 การอ่านค่าอนาล็อก (Analog Read)

1) ให้ต่อ Potentiometer กับ A2 ของ Arduino ตามรูป



รูปที่ 6 การต่อ Arduino กับ Potentiometer

2) ให้เขียนโปรแกรมต่อไปนี้ จากนั้นโหลดลงบอร์ด แล้วทดสอบการทำงาน

```
1) //the setup routine runs once when you press reset:
2) void setup{
3) //initialize serial communication at 9600 bits per second:
4)   Serial.begin(9600);
5) }

6) //the loop routine runs over and over again forever:
7) void loop{
8)   //read the input on analog pin 0:
9)   int sensorValue = analogRead(A2);
10)  //print out the value you read:
11)  Serial.println(sensorValue);
12)  delay(1);        //delay in between reads for stability
```

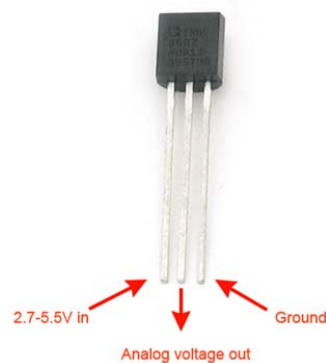

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____ วันเดือนปี _____ / _____ / _____
 รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____
 13) }

คำสั่ง `Serial.begin(9600)` เป็นคำสั่งที่ใช้กำหนดความเร็วในการรับส่งข้อมูลระหว่างบอร์ด Arduino และเครื่องพีซี คำสั่ง `analogRead()` เป็นคำสั่งที่ใช้อ่านค่าจากพอร์ตแอนะล็อก โดยจะเป็นค่าที่อยู่ระหว่าง 0-1023 ซึ่งเป็นค่าที่แปลงจากค่า 0-5 V โดยใช้ ADC 10 บิต คำสั่ง `Serial.println()` เป็นคำสั่งที่ใช้ในการส่งค่าข้อมูลไปยัง PC โดยจะแสดงในหน้าต่าง Serial Monitor (เรียกดูโดย Tools -> Serial Monitor)

- 3) ให้แก้ไขโปรแกรม โดยเขียนโปรแกรมแปลงจากค่า 0-1024 ที่อ่านได้ ให้แสดงค่ากลับมาเป็นค่าโอห์ม โดยใช้มัลติมิเตอร์วัดค่าประกอบด้วยเพื่อความแม่นยำ เสร็จแล้วให้อาจารย์ตรวจ

ลายเซ็น.....

- 4) ต่อวงจร LM35 กับ Arduino โดย LM35 เป็นไอซีเซนเซอร์สำหรับวัดอุณหภูมิ มี 3 ขา โดยขาที่ 1 จะต่อไฟเลี้ยง ขาที่ 3 จะเป็นกราวด์ และขาที่ 2 จะเป็นค่าอุณหภูมิ โดยในการแปลงค่าเป็นอุณหภูมิ จะใช้สูตร $\text{Temp in } ^\circ\text{C} = (\text{Vout in mV}) / 10$



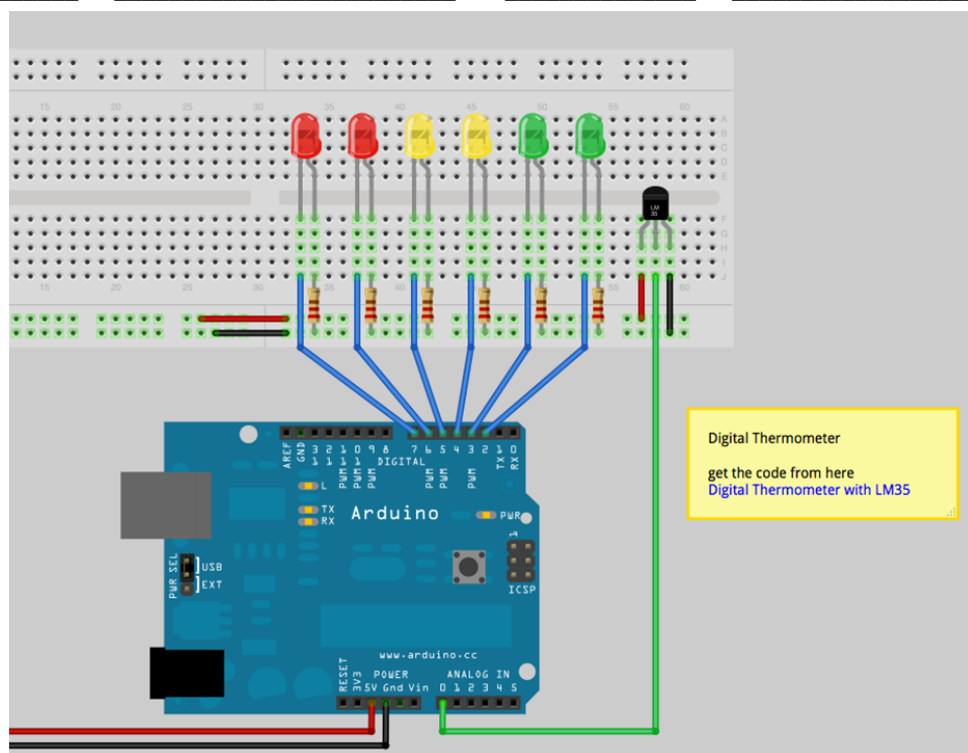
รูปที่ 7 LM35

- 5) ต่อวงจรเพื่อแสดงระดับความร้อนผ่าน LED ตามรูป แล้วเขียนโปรแกรม โดยกำหนดช่วงอุณหภูมิตามความเหมาะสม เสร็จแล้วให้อาจารย์ตรวจ

กลุ่ม(เข้า-บ่าย) _____ กลุ่มที่ _____ ชั้นปีที่ _____ ห้อง _____

รหัส _____ ชื่อ _____ รหัส _____ ชื่อ _____

วันเดือนปี _____/_____/_____



รูปที่ 8 LM35

ลายเซ็น.....

สรุปผลการทดลอง
