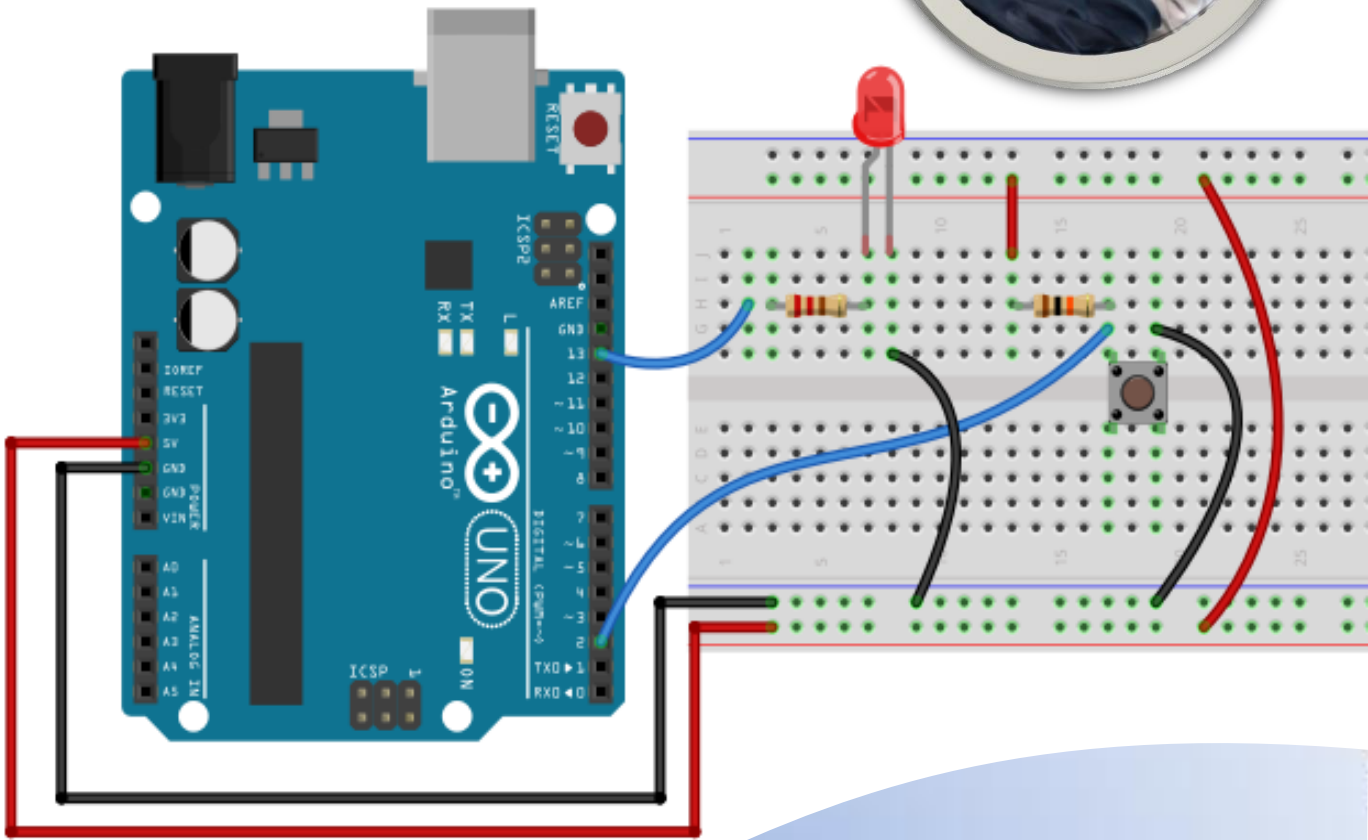




ใบงานการทดลอง

วิชาไมโครคอนโทรลเลอร์

รหัสวิชา 3105-2007



นายสง่า คุณำ

แผนกวิชาช่างอิเล็กทรอนิกส์ วิทยาลัยเทคนิคพัทยา

คำนำ

ใบงานการทดลองฉบับนี้เป็นส่วนหนึ่งของ วิชาไมโครคอนโทรลเลอร์ รหัสวิชา 3105-2007 หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง พุทธศักราช 2557 ประเภทวิชาช่างอุตสาหกรรม สาขาวิชาช่างอิเล็กทรอนิกส์ โดยครอบคลุมจุดประสงค์รายวิชา สมรรถนะรายวิชา คำอธิบายรายวิชา เพื่อต้องการพัฒนาการจัดการเรียนการสอนและการวัดผลและประเมินผลตามสภาพจริง โดยมุ่งเน้นทักษะผู้เรียนเป็นสำคัญ เรียนรู้จากการฝึกปฏิบัติจริงมาใช้ในการเรียนการสอน ในการเรียบเรียงได้แบ่งใบงานการทดลองออกเป็น 18 ใบงาน

ผู้จัดทำหวังว่าเอกสารใบงานการทดลองฉบับนี้ จะอำนวยความสะดวกต่อการเรียนการสอน และช่วยให้การเรียนการสอนมีประสิทธิภาพและบรรลุตามวัตถุประสงค์ นักเรียนนักศึกษา มีผลการเรียนที่ดีขึ้น เกิดทักษะการปฏิบัติงานมากขึ้น

นายสง่า คำคำ
วิทยาลัยเทคนิคพัทธยา

สารบัญ

	หน้า
คำนำ	ก
สารบัญ	๗
ใบงานการทดลองที่ 1 เรื่องงานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	1
ใบงานการทดลองที่ 2 เรื่องงานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	9
ใบงานการทดลองที่ 3 เรื่องงานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	18
ใบงานการทดลองที่ 4 เรื่องงานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่าง ๆ	23
ใบงานการทดลองที่ 5 เรื่องงานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่าง ๆ	31
ใบงานการทดลองที่ 6 เรื่องงานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	42
ใบงานการทดลองที่ 7 เรื่องงานโปรแกรมควบคุม Buzzer	49
ใบงานการทดลองที่ 8 เรื่องงานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	57
ใบงานการทดลองที่ 9 เรื่องงานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	64
ใบงานการทดลองที่ 10 เรื่องงานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	80
ใบงานการทดลองที่ 11 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	86
ใบงานการทดลองที่ 12 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	92
ใบงานการทดลองที่ 13 เรื่องงานโปรแกรมการเชื่อมต่อกับ PIR Sensor	103
ใบงานการทดลองที่ 14 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	107
ใบงานการทดลองที่ 15 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	115
ใบงานการทดลองที่ 16 เรื่องงานการเขียนโปรแกรมควบคุมมอเตอร์	123
ใบงานการทดลองที่ 17 เรื่องงานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	133
ใบงานการทดลองที่ 18 เรื่องงานโปรแกรม Smart Farm system เบื้องต้น	139
บรรณานุกรม	

สารบัญรูปภาพ


รูปที่	หน้า
1.1 แสดงวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3	2
1.2 แสดงหน้าต่างของโปรแกรม Arduino IDE ที่พิมพ์โค้ดเสร็จแล้ว	3
1.3 แสดงเมนูต่างๆ ของโปรแกรม Arduino IDE	3
1.4 แสดงขั้นตอนการเลือกบอร์ดเพื่อใช้งาน	4
1.5 แสดงการเข้าไปที่เมนู File -> Preferences	4
1.6 แสดงการ Compile โค้ดและที่อยู่ของโปรแกรม *.HEX	5
1.7 แสดงขั้นตอนการนำ ไฟล์ *.HEX มาใช้กับโปรแกรม Proteus 7 Professional	5
1.8 แสดงการจำลองการทำงานของวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3	6
1.9 แสดงการต่อวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3 ต่อบอร์ดจริง	6
1.10 แสดงการเลือก COM port ที่ใช้ในการติดต่อกับบอร์ด Arduino UNO R3	7
1.11 แสดงการ Upload โปรแกรมลงบอร์ด Arduino UNO R3	7
2.1 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง	10
2.2 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง	11
2.3 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB	12
2.4 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB	14
2.5 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment	15
2.6 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment	17
3.1 แสดงการออกแบบวงจรรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	19
3.2 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง	21
4.1 แสดงการออกแบบวงจรรับค่า Analog INPUT มาควบคุม LED	24
4.2 แสดงวงจรรับค่า Analog INPUT มาควบคุม LED	27
4.3 แสดงการออกแบบวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB	28
4.4 แสดงวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB	29
5.1 แสดงการออกแบบวงจรรับค่า Push Switch มาควบคุม LED	32
5.2 แสดงวงจรรับค่า Push Switch มาควบคุม LED	35
5.3 แสดงการออกแบบวงจรรับค่า Push Switch มาควบคุม LED 7-Segment	36

สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
5.4 แสดงวงจรรับค่า Push Switch มาควบคุม LED 7-Segment	40
6.1 แสดงการออกแบบวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment	43
6.2 แสดงวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment	47
7.1 แสดงการออกแบบวงจร LED 7-Segment และ Buzzer	50
7.2 แสดงวงจร LED 7-Segment และ Buzzer	55
8.1 แสดงการออกแบบวงจรควบคุม Buzzer ร่วมกับ Keypad	58
8.2 แสดงวงจรควบคุม Buzzer ร่วมกับ Keypad	62
9.1 แสดงการออกแบบวงจรแสดงข้อมูลที่ LCD	65
9.2 แสดงแสดงข้อมูลที่ LCD	70
9.3 แสดงการออกแบบวงจรรับค่า Analog แสดงข้อมูลที่ LCD	71
9.4 แสดงวงจรรับค่า Analog แสดงข้อมูลที่ LCD	74
9.5 แสดงการออกแบบวงจรรับค่าจาก keypad แสดงข้อมูลที่ LCD	75
9.6 แสดงวงจรรับค่าจาก keypad แสดงข้อมูลที่ LCD	78
10.1 แสดงการออกแบบวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	81
10.2 แสดงวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	84
11.1 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	87
11.2 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	90
12.1 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	93
12.2 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	96
12.3 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer	97
12.4 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer	101
13.1 แสดงการออกแบบวงจรเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD	104
13.2 แสดงวงจรเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD	106
14.1 แสดงการออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LED 7-Segment	108
14.2 แสดงรับค่าจาก DS18B20 มาแสดงที่ LED 7-Segment	111

สารบัญรูปภาพ(ต่อ)

รูปที่		หน้า
14.3	แสดงการออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LCD	112
14.4	แสดงรับค่าจาก DS18B20 มาแสดงที่ LCD	114
15.1	แสดงการออกแบบวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มา แสดงผลที่ LED 7-Segment	116
15.2	แสดงวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7- Segment	119
15.3	แสดงการออกแบบวงจรรับค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผล ที่ LCD	120
15.4	แสดงวงจรรับค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงที่ LCD	122
16.1	แสดงการออกแบบวงจรการควบคุม Stepper Motor ด้วย Keypad	124
16.2	แสดงวงจรการควบคุม Stepper Motor ด้วย Keypad	126
16.3	แสดงการออกแบบวงจรควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้	127
16.4	แสดงวงจรควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้	128
16.5	แสดงการออกแบบวงจรควบคุม DC Motor ด้วยการกด Pushbutton	130
16.6	แสดงวงจรควบคุม DC Motor ด้วยการกด Pushbutton	131
17.1	แสดงการออกแบบวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	134
17.2	แสดงวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	137
18.1	แสดงการออกแบบวงจรวงจร Smart Farm system เบื้องต้น	140
18.2	แสดงวงจร Smart Farm system เบื้องต้น	144

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	1

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 1 เรื่องงานโปรแกรม Proteus และ Arduino IDE เบื้องต้นตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์


1. เพื่อให้สามารถใช้งานเครื่องมือต่าง ๆ ของโปรแกรม Proteus ได้อย่างถูกต้อง
2. เพื่อให้สามารถใช้โปรแกรม Proteus ในการจำลองการทำงานของวงจรอิเล็กทรอนิกส์ได้อย่างถูกต้อง
3. เพื่อให้สามารถใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมภาษา C เบื้องต้นได้อย่างถูกต้อง
4. เพื่อให้สามารถใช้งานบอร์ด Arduino UNO R3 เบื้องต้นได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

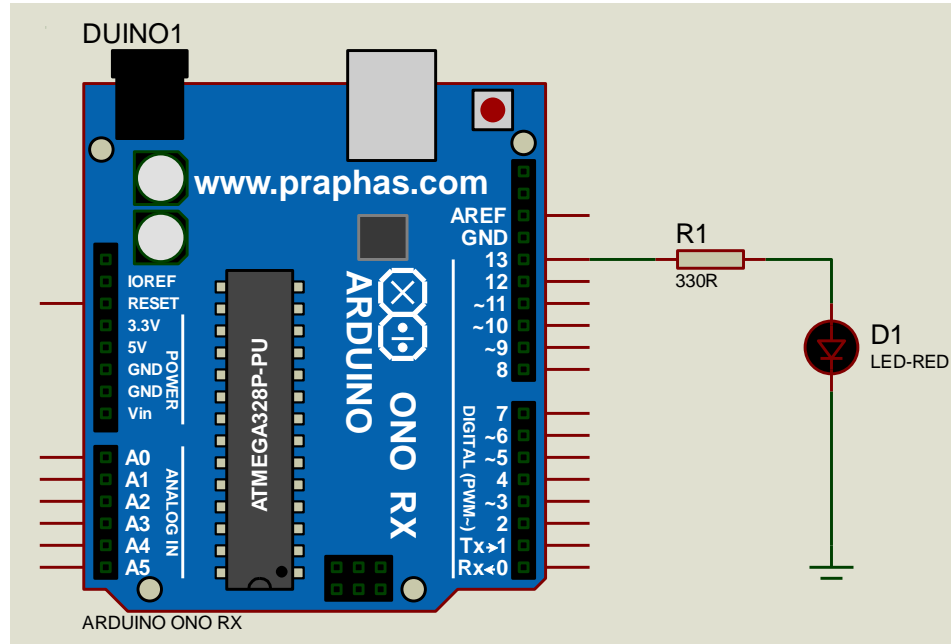
- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 | 1 | ชุด |
| 5. สายต่อวงจร | 1 | ชุด |
| 6. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 7. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

1. ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
2. ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
3. ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
4. ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
5. ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	2

วงจรประกอบการทดลอง



รูปที่ 1.1 แสดงวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3


ลำดับขั้นตอนการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรไฟกระพริบ 1 ดวง แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 1.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า
3. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้

```

1 int led = 13;
2 void setup()
3 {
4   pinMode(led, OUTPUT);
5 }
6 void loop()
7 {
8   digitalWrite(led, HIGH);
9   delay(1000);
10  digitalWrite(led, LOW);
11  delay(1000);
12 }

```


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	3



```

EX01 | Arduino 1.8.4
File Edit Sketch Tools Help
EX01 $
1 int led = 13;
2 void setup()
3 {
4   pinMode(led, OUTPUT);
5 }
6 void loop()
7 {
8   digitalWrite(led, HIGH);
9   delay(1000);
10  digitalWrite(led, LOW);
11  delay(1000);
12 }

Invalid library found in C:\Program Files


11 Arduino/Genuino Uno on COM4

```

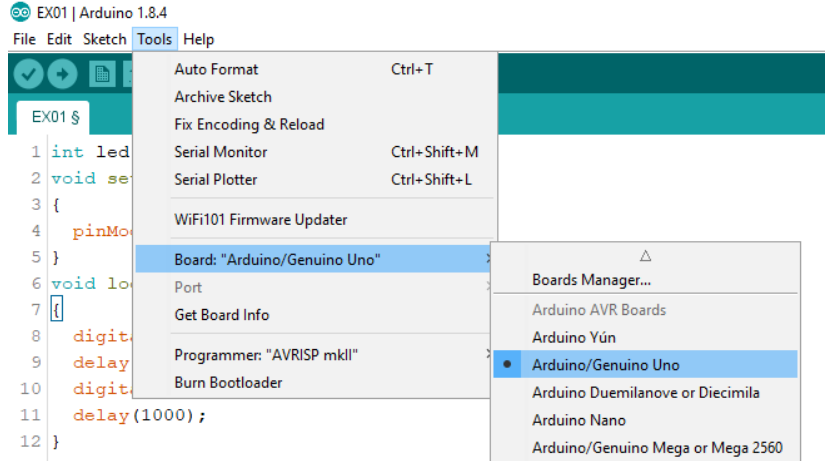
รูปที่ 1.2 แสดงหน้าต่างของโปรแกรม Arduino IDE ที่พิมพ์โค้ดเสร็จแล้ว



รูปที่ 1.3 แสดงเมนูต่างๆ ของโปรแกรม Arduino IDE

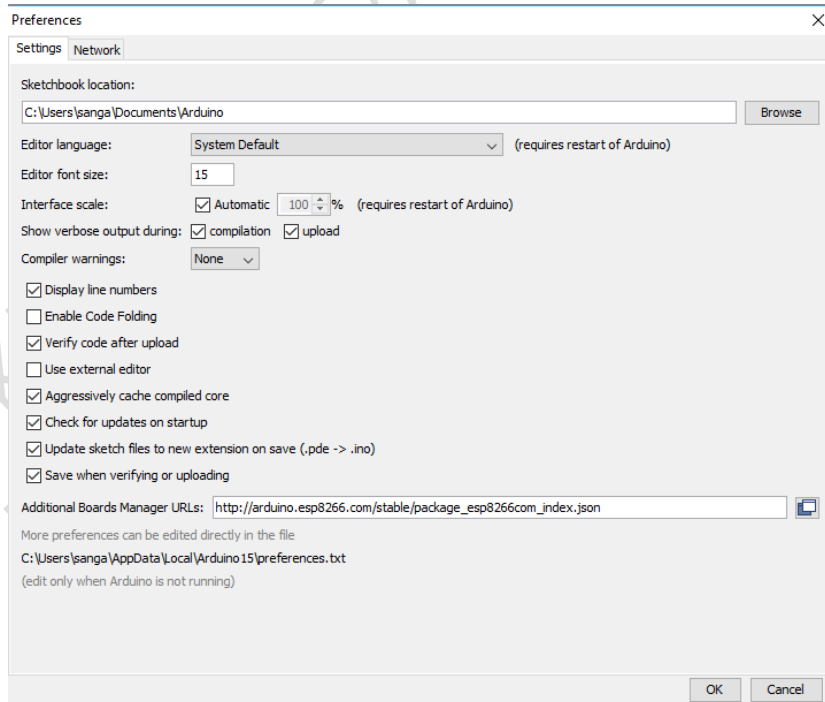
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	4

4. เลือกบอร์ดที่ใช้งานเป็น Arduino Uno ดังรูปที่ 1.4




รูปที่ 1.4 แสดงขั้นตอนการเลือกบอร์ดเพื่อใช้งาน

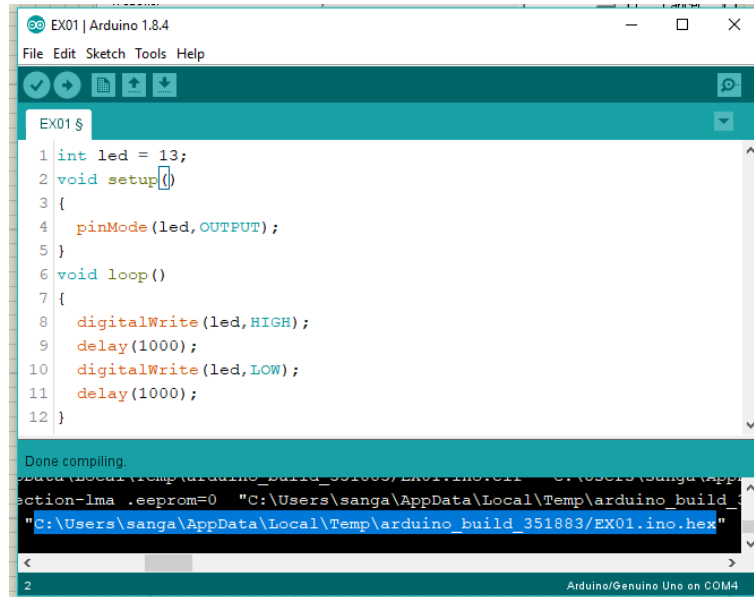
5. ให้เขาไปที่เมนู File -> Preferences แล้วเลือกของ compilation เพื่อให้แสดงที่อยู่ของไฟล์ *.HEX ที่ได้จากการ Compile ดังแสดงในรูปที่ 5 (การเลือกนี้ จะทำเพียงครั้งเดียว เนื่องจาก โปรแกรมจะจำการเลือก คำนีไว้ตลอด จนกว่าจะมีการเปลี่ยนแปลงภายหลัง)



รูปที่ 1.5 แสดงการเขาไปที่เมนู File -> Preferences

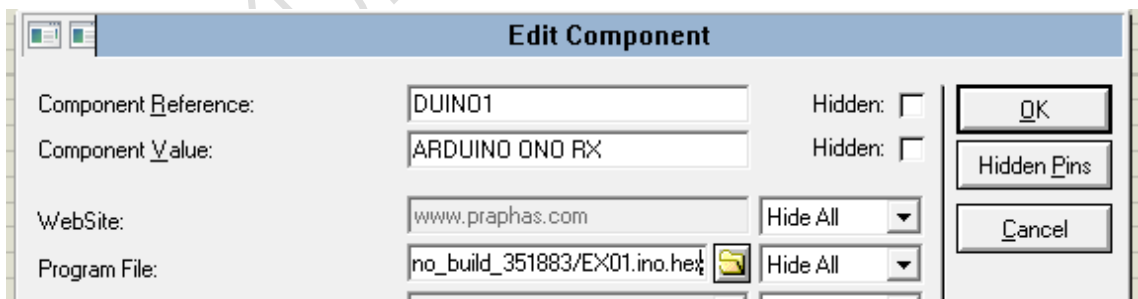
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	5

6. เลือกเครื่องมือ Verify เพื่อทำการ Compile โค้ด ถ้าพิมพ์โค้ดได้ถูกต้อง จะมีข้อความแจ้งว่า Done compiling พร้อมทั้งแสดงตำแหน่งที่อยู่ของไฟล์ดังรูปที่ 1.6




รูปที่ 1.6 แสดงการ Compile โค้ดและที่อยู่ของโปรแกรม *.HEX

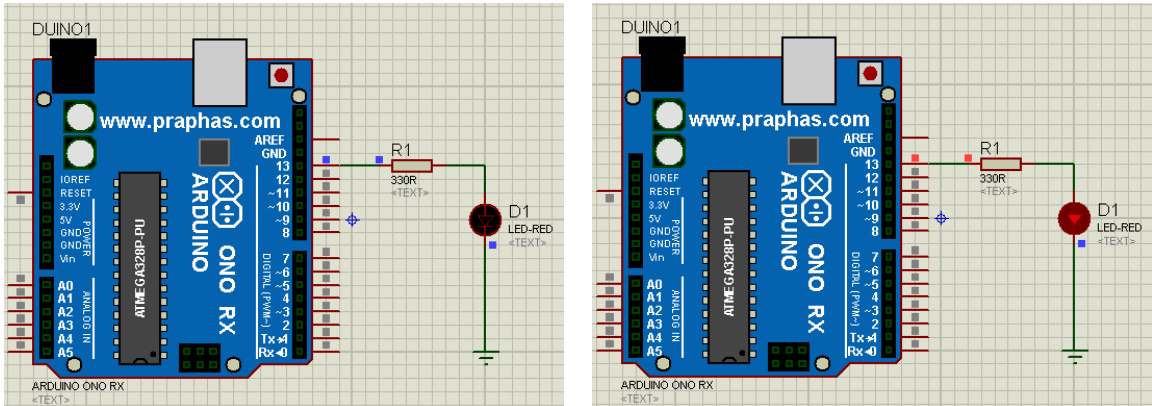
7. เมื่อได้ไฟล์ *.HEX มาแล้ว ให้นำกลับไปโปรแกรม Proteus 7 Professional และให้ดับเบิลคลิกที่บอร์ด Arduino UNO R3 เพื่อเป็นการกำหนดชื่อไฟล์ *.HEX ที่ต้องการ RUN บนบอร์ด โดยให้เลือกไฟล์ *.HEX ที่ได้จาก ขั้นตอนที่แล้ว แสดงดังรูปที่ 1.7



รูปที่ 1.7 แสดงขั้นตอนการนำ ไฟล์ *.HEX มาใช้กับโปรแกรม Proteus 7 Professional

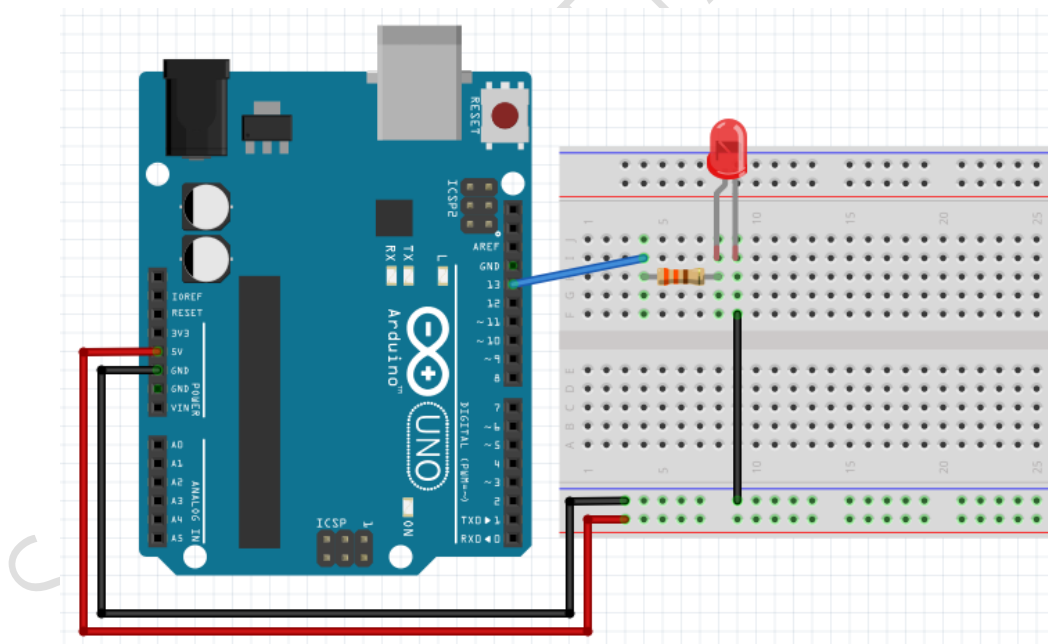
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มทดลองการทำงาน ซึ่งจะแสดงผลการทำงานดังรูปที่ 1.8 โดยหลอด LED จะติด-ดับ สลับกันทุก ๆ 1 วินาที ดังรูปที่ 1.8

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	6




รูปที่ 1.8 แสดงการจำลองการทำงานของวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3

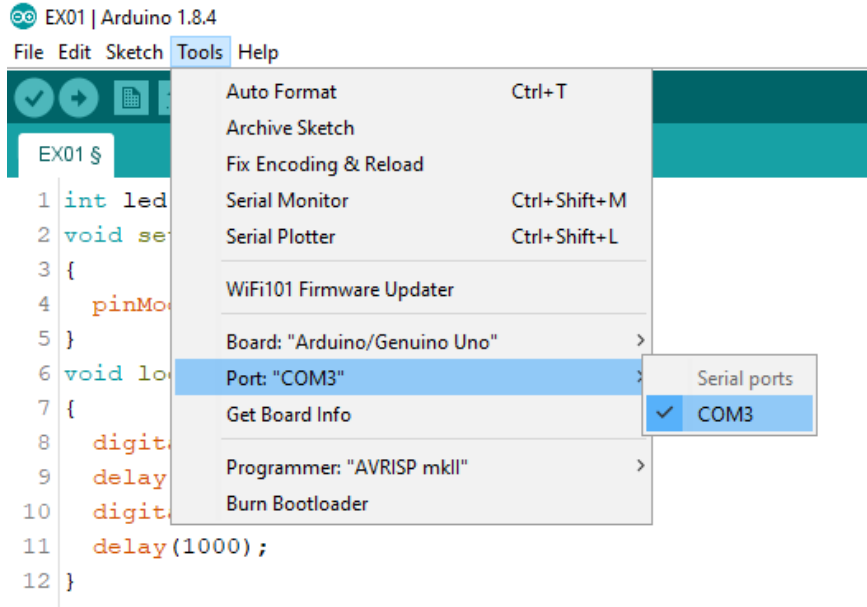
9. ประกอบวงจรไฟกระพริบ 1 ดวง โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่ 1.9



รูปที่ 1.9 แสดงการต่อวงจรไฟกระพริบ 1 ดวงโดยใช้บอร์ด Arduino UNO R3 ต่อบอร์ดจริง

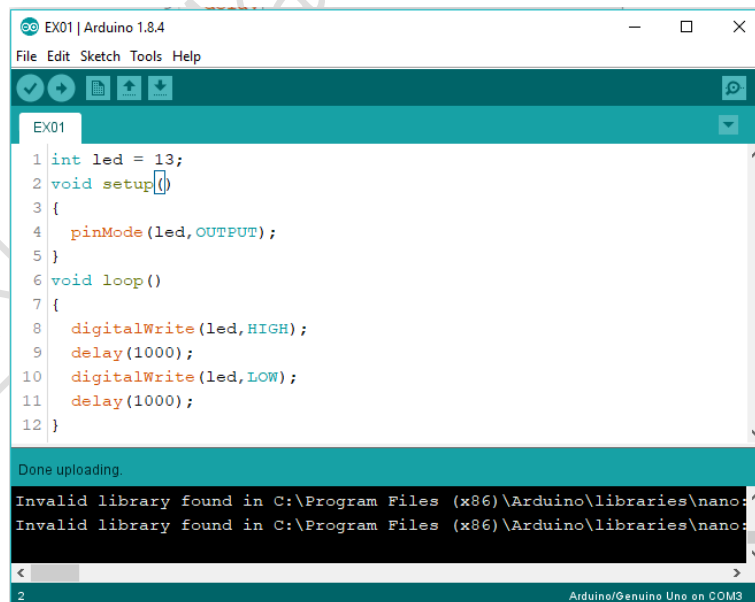
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 1
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Proteus และ Arduino IDE เบื้องต้น	7

10. ให้เลือก COM port ที่ใช้ในการติดต่อกับบอร์ด Arduino ในตัวอย่างนี้ใช้ COM3 ดังรูปที่ 1.10




รูปที่ 1.10 แสดงการเลือก COM port ที่ใช้ในการติดต่อกับบอร์ด Arduino UNO R3

11. กดปุ่มเครื่องหมายลูกศร เพื่อ Upload โปรแกรมลงบอร์ด Arduino UNO R3 แสดงดัง รูปที่ 1.11



รูปที่ 1.11 แสดงการ Upload โปรแกรมลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	9	

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 2 เรื่องงานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจรการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 | 1 | ชุด |
| 5. สายต่อวงจร | 1 | ชุด |
| 6. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 7. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

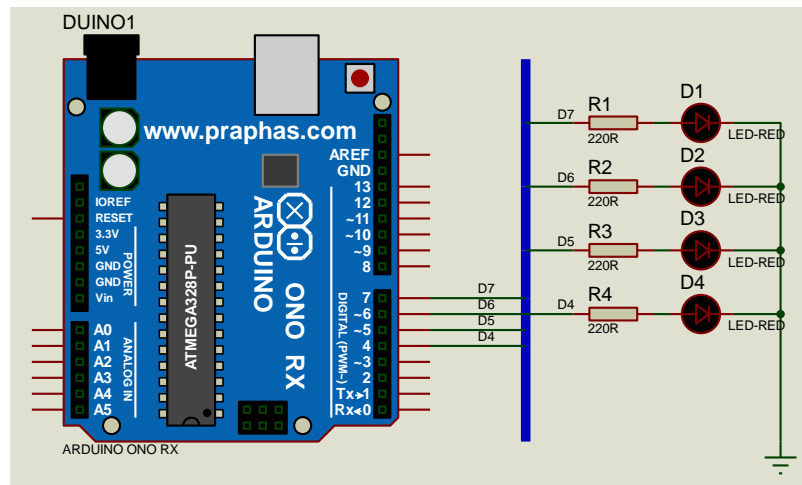
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	10

ลำดับขั้นการทดลอง

ตอนที่ 1 โปรแกรมเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวงแบบไซบอร์ด Arduino UNO R3 แสดงดังรูปที่ 2.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 2.1 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง

3. บันทึกไฟล์ชื่อ LAB2_1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้


EX01 | Arduino 1.8.4

File Edit Sketch Tools Help

```

EX01 §
1 int led1 = 7;int led2 = 6;int led3 = 5;int led4 = 4;
2 void setup()
3 {
4     pinMode(led1,OUTPUT);pinMode(led2,OUTPUT);
5     pinMode(led3,OUTPUT);pinMode(led4,OUTPUT);
6 }
7 void loop()
8 {
9     digitalWrite(led1,HIGH);digitalWrite(led2,LOW);
10    digitalWrite(led3,LOW);digitalWrite(led4,LOW);
11    delay(1000);

```


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	11

```

12 digitalWrite (led1, LOW) ; digitalWrite (led2, HIGH) ;
13 digitalWrite (led3, LOW) ; digitalWrite (led4, LOW) ;
14 delay (1000) ;
15 digitalWrite (led1, LOW) ; digitalWrite (led2, LOW) ;
16 digitalWrite (led3, HIGH) ; digitalWrite (led4, LOW) ;
17 delay (1000) ;
18 digitalWrite (led1, LOW) ; digitalWrite (led2, LOW) ;
19 digitalWrite (led3, LOW) ; digitalWrite (led4, HIGH) ;
20 delay (1000) ;
21 }

```

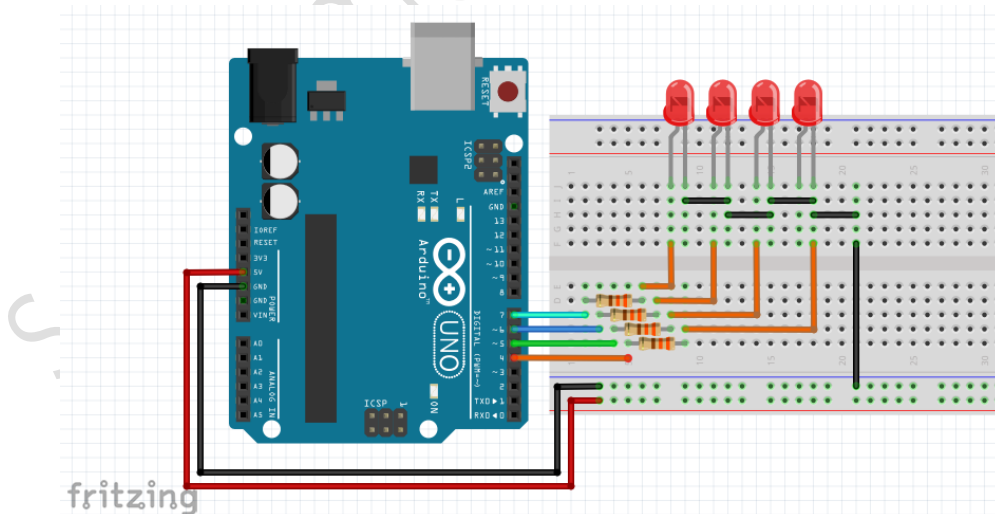
5. บันทึกไฟล์โค้ด ชื่อ LAB2_1.ino
6. ทำการ Compile โค้ด LAB2_1.ino
7. นำไฟล์ LAB2_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....


.....

10. ประกอบวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวงโดยใช้บอร์ด Arduino UNO R3 ดังรูปที่ 2.2



รูปที่ 2.2 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED 4 ดวง

11. Upload โปรแกรม LAB2_1.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	12

12. สังเกตและบันทึกผลการทดลอง

.....

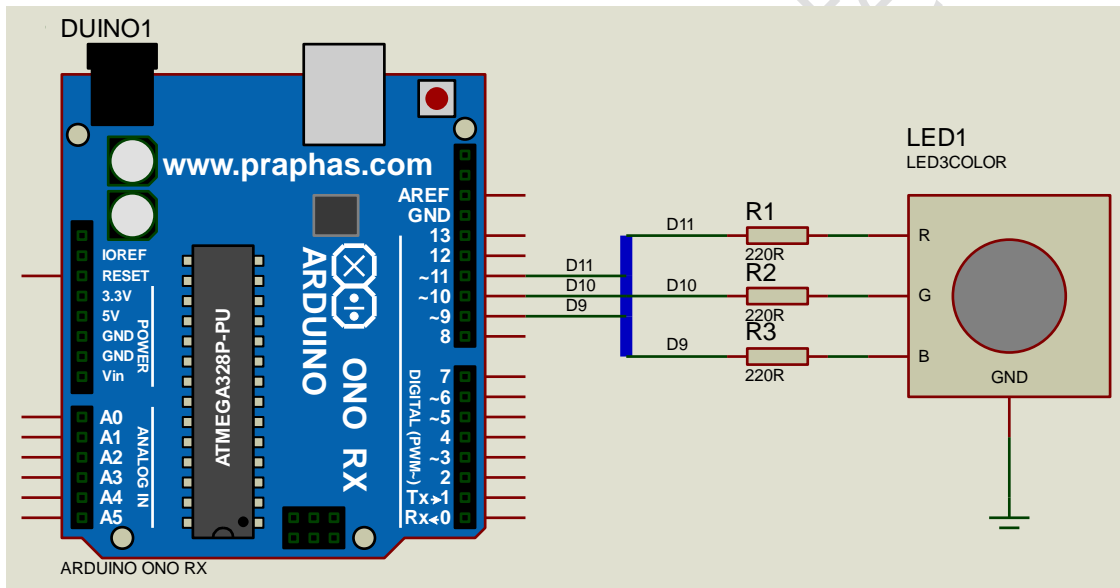
.....

.....

.....

ตอนที่ 2 โปรแกรมเชื่อมต่อเอาต์พุตด้วย LED RGB

13. ออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 2.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 2.3 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB


14. บันทึกไฟล์ชื่อ LAB2_2

15. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้

```

1 int redPin = 11;
2 int greenPin = 10;
3 int bluePin = 9;
4 void setup()
5 {
6     pinMode(redPin, OUTPUT);
7     pinMode(greenPin, OUTPUT);
8     pinMode(bluePin, OUTPUT);
9 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	13

```

10 void loop()
11 {
12     analogWrite (redPin, 255);
13     analogWrite (greenPin, 0);
14     analogWrite (bluePin, 0);
15     delay (1000);
16     analogWrite (redPin, 0);
17     analogWrite (greenPin, 255);
18     analogWrite (bluePin, 0);
19     delay (1000);
20     analogWrite (redPin, 0);
21     analogWrite (greenPin, 0);
22     analogWrite (bluePin, 255);
23     delay (1000);
24     analogWrite (redPin, 255);
25     analogWrite (greenPin, 255);
26     analogWrite (bluePin, 0);
27     delay (1000);
28     analogWrite (redPin, 255);
29     analogWrite (greenPin, 0);
30     analogWrite (bluePin, 255);
31     delay (1000);
32     analogWrite (redPin, 0);
33     analogWrite (greenPin, 255);
34     analogWrite (bluePin, 255);
35     delay (1000);
36 }

```

16. บันทึกไฟล์โค้ด ชื่อ LAB2_2.ino
17. ทำการ Compile โค้ด LAB2_2.ino
18. นำไฟล์ LAB2_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
19. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
20. สังเกตและบันทึกผลการทดลอง


.....

.....

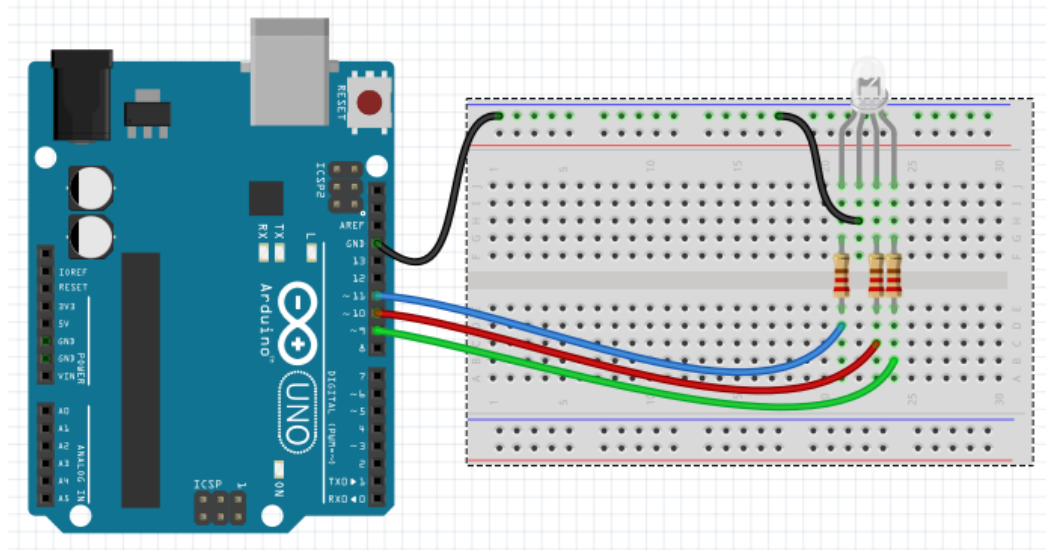
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	14

21. ประกอบวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่ 2.4



รูปที่ 2.4 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED RGB

22. Upload โปรแกรม LAB2_2.ino ลงบอร์ด Arduino UNO R3

23. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....


.....

.....

.....

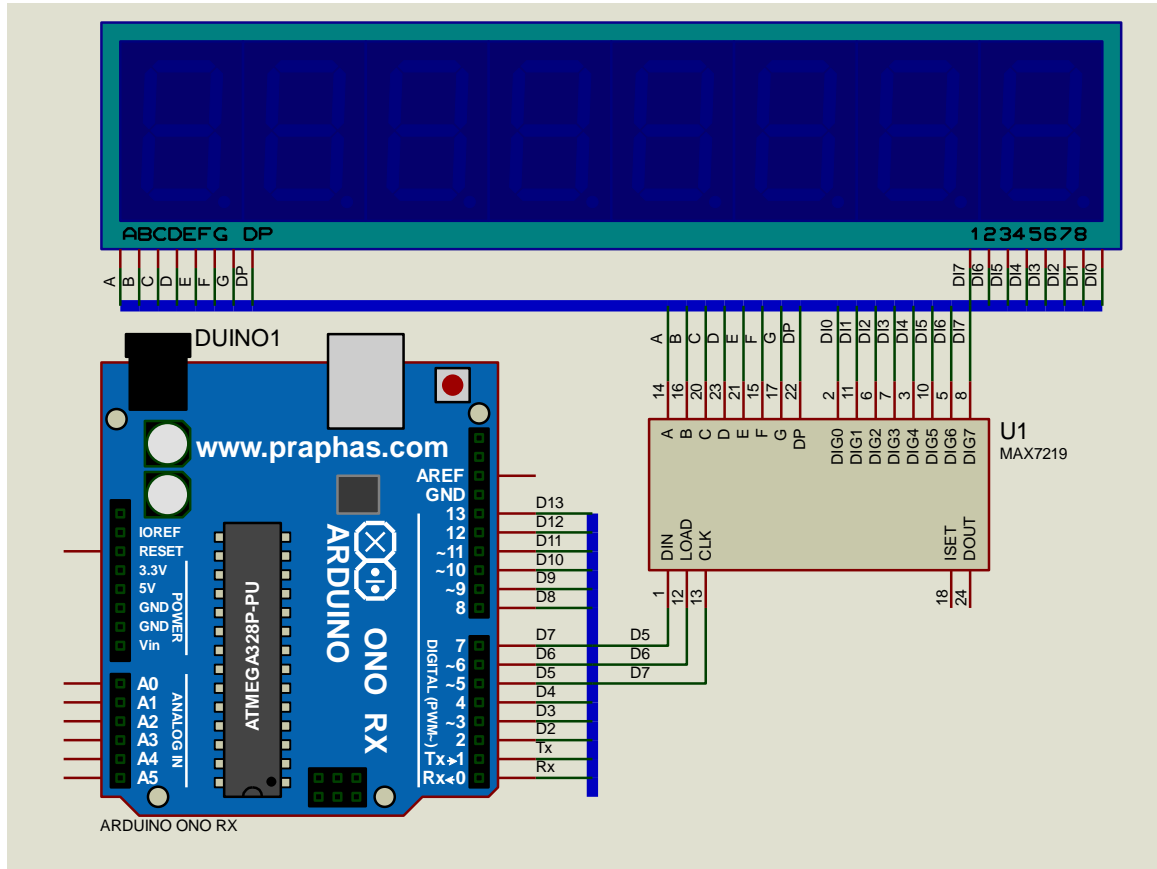
.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	15


ตอนที่ 3 โปรแกรมเชื่อมต่อเอาต์พุตด้วย LED 7-Segment

24. ออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment แบบไซบอร์ด Arduino UNO R3 แสดงดังรูปที่ 2.5 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 2.5 แสดงการออกแบบวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment

25. บันทึกไฟล์ชื่อ LAB2_3
 26. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาท์พุตด้วย LED แบบต่าง ๆ	16

```

1 #include "LedControl.h"
2 LedControl lc=LedControl(5,7,6,1);
3 int num = 0,seg1,seg2;
4 void setup()
5 {
6   lc.shutdown(0,false);
7   lc.setIntensity(0,5);
8   lc.clearDisplay(0);
9 }
10 void loop()
11 {
12   seg1=num%10;
13   seg2=num/10;
14   lc.setDigit(0,0,seg1,false);
15   lc.setDigit(0,1,seg2,false);
16   delay(1000);
17   num=num+1;
18   if (num > 99)
19   {
20     num = 0;
21   }
22 }

```

27. บันทึกไฟล์โค้ด ชื่อ LAB2_3.ino
28. ทำการ Compile โค้ด LAB2_3.ino
29. นำไฟล์ LAB2_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
30. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
31. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....


.....

.....

.....

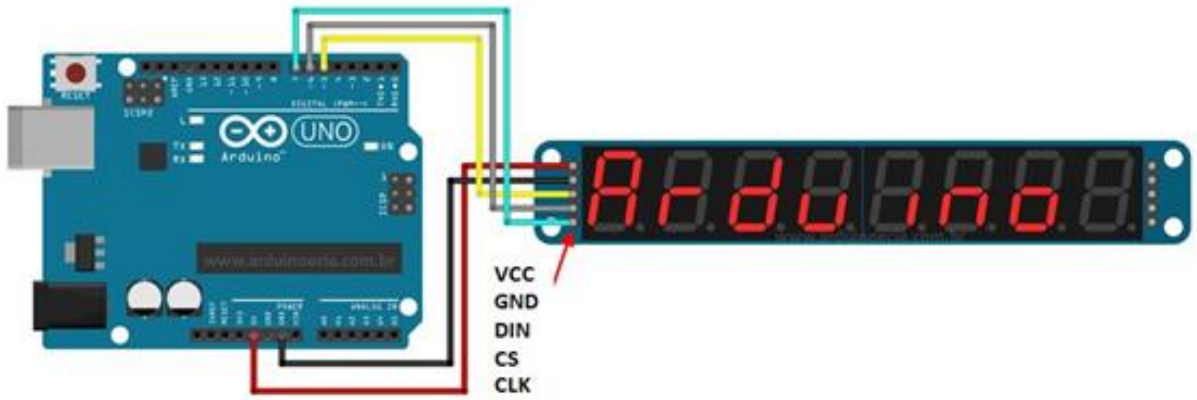
.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 2
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อเอาต์พุตด้วย LED แบบต่าง ๆ	17

32. ประกอบวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่

2.6



รูปที่ 2.6 แสดงวงจรเชื่อมต่อเอาต์พุตด้วย LED 7-Segment

33. Upload โปรแกรม LAB2_3.ino ลงบอร์ด Arduino UNO R3

34. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

35. สรุปผลการทดลอง

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 3	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	18	

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 3 เรื่องงานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรการรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจรการรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 | 1 | ชุด |
| 5. สายต่อวงจร | 1 | ชุด |
| 6. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 7. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

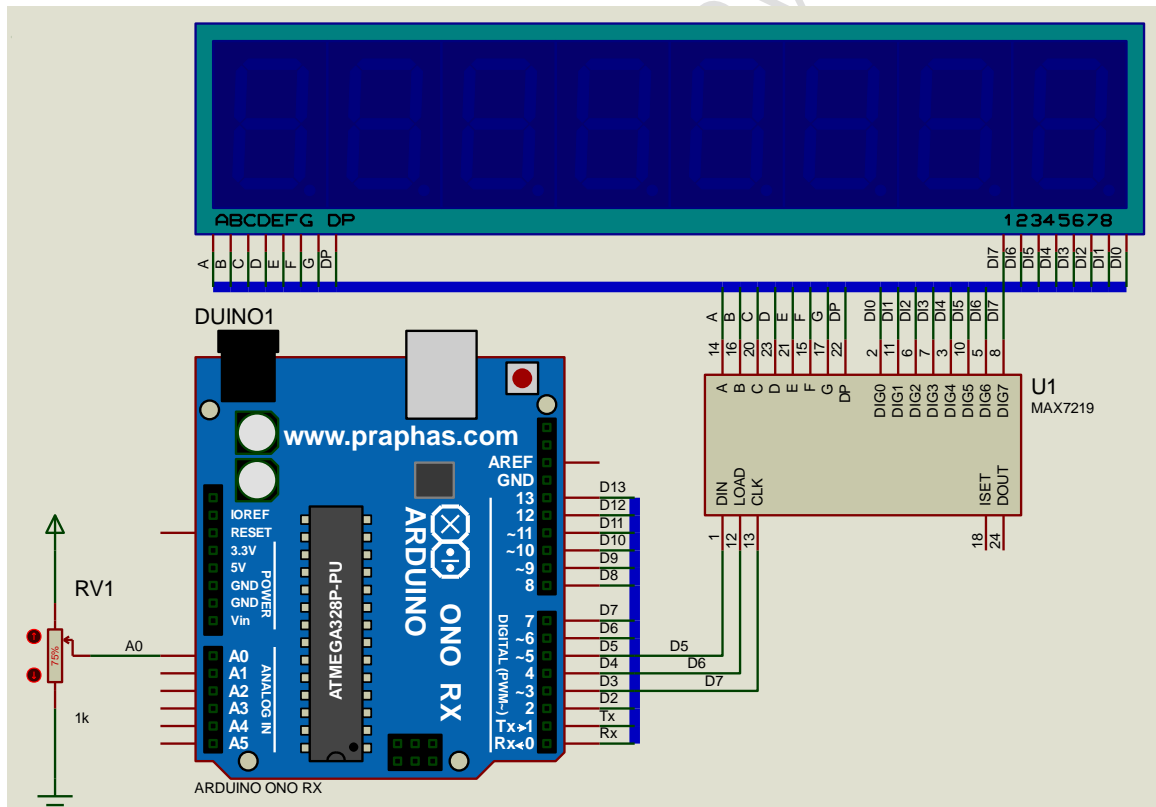
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 3
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	19

- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย


ลำดับขั้นการทดลอง

- เปิดโปรแกรม Proteus 7 Professional
- ออกแบบวงจรรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 3.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 3.1 แสดงวงจรรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment

- บันทึกไฟล์ชื่อ LAB3_1

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 3
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	20

4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้


```

LAB3_1
1 #include "LedControl.h"
2 LedControl lc=LedControl(5,7,6,1);
3 // Pin 5->DIN, 7->CLK, 6->CS(LOAD), 1 = No.of devices
4 void show4digit(int num)
5 {
6     int seg1,seg2,seg3,seg4;
7     seg1 = ((num%1000)%100)%10;
8     seg2 = ((num%1000)%100)/10;
9     seg3 = (num%1000)/100;
10    seg4 = num/1000;
11    lc.setDigit(0,0,seg1,false);
12    if (num>=10)
13        lc.setDigit(0,1,seg2,false);
14        if (num>=100)
15            lc.setDigit(0,2,seg3,false);
16        if (num>=1000)
17            lc.setDigit(0,3,seg4,false);
18    delay(300);
19 }
20 void setup()
21 {
22     Serial.begin(9600);
23     lc.shutdown(0,false);
24     lc.setIntensity(0,5);
25     lc.clearDisplay(0);
26 }
27 void loop()
28 {
29     int num;
30     num = analogRead(A0);
31     Serial.print("analogRead=");Serial.println(num);
32     lc.clearDisplay(0);
33     show4digit(num);
34 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB3_1.ino

6. ทำการ Compile โค้ด LAB3_1.ino

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 3
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment	21

7. นำไฟล์ LAB3_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

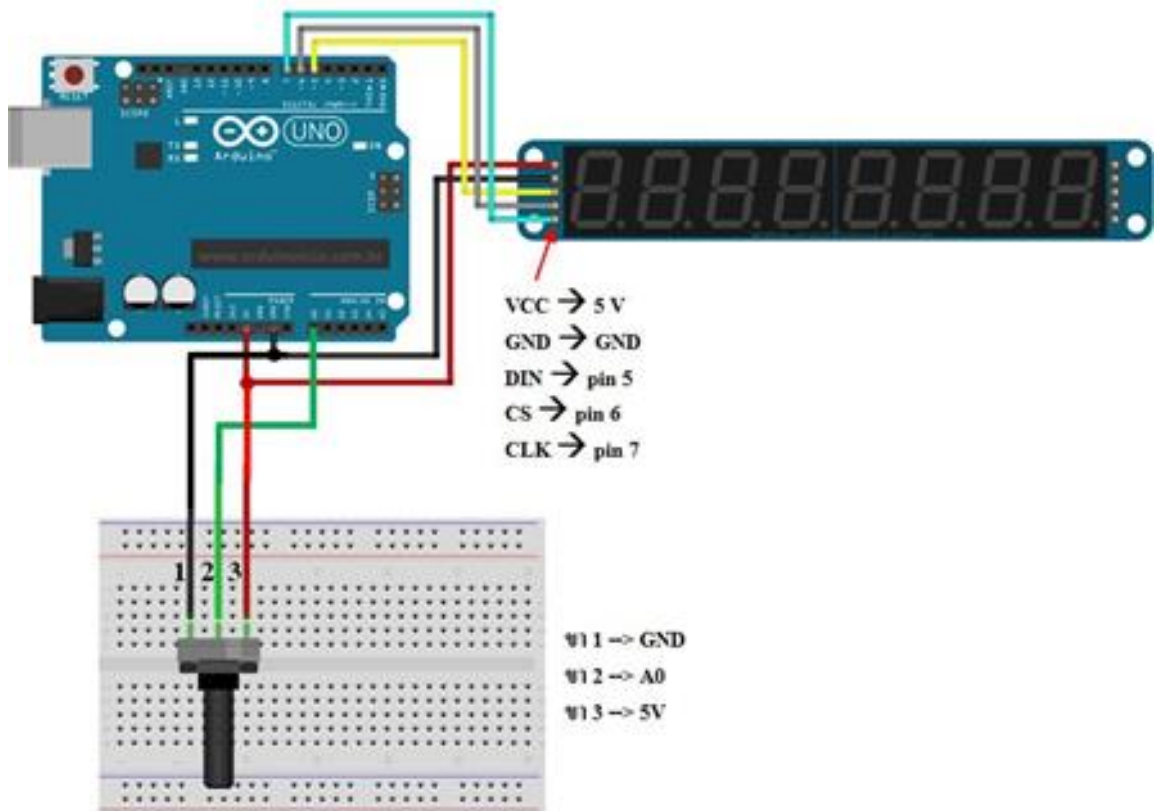
.....

.....

.....


.....

10. ประกอบวงจรรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่ 3.2



รูปที่ 3.2 แสดงวงจรเชื่อมต่อเอาท์พุทด้วย LED 4 ดวง

11. Upload โปรแกรม LAB3_1.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	23

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 4 เรื่องงานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรการรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานการรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรการรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 | 1 | ชุด |
| 5. สายต่อวงจร | 1 | ชุด |
| 6. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 7. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือชิปต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้

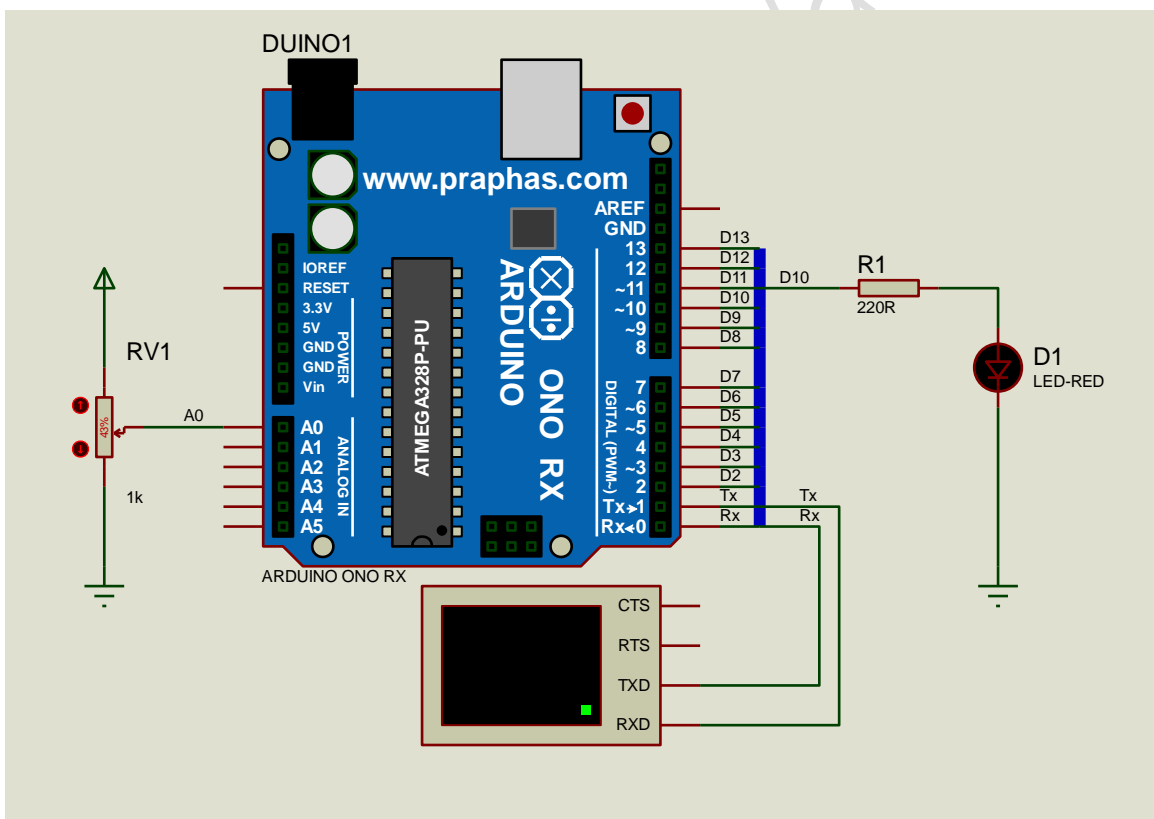
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	24

- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

ลำดับขั้นตอนการทดลอง


ตอนที่ 1 โปรแกรมการรับค่า Analog INPUT มาควบคุม LED

- เปิดโปรแกรม Proteus 7 Professional
- ออกแบบวงจรรับค่า Analog INPUT มาควบคุม LED แบบไซบอร์ด Arduino UNO R3 แสดงดังรูปที่ 4.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 4.1 แสดงวงจรรับค่า Analog INPUT มาควบคุม LED

- บันทึกไฟล์ชื่อ LAB4_1
- เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	25

LAB4_1

```

1 int led = 10;
2 void setup()
3 {
4   Serial.begin(9600);
5   pinMode(led, OUTPUT);
6 }
7 void loop()
8 {
9   int x,y;
10  x = analogRead(A0);
11  y = map(x, 0, 1023, 0, 255);
12  Serial.print("x = ");Serial.println(x);
13  Serial.print("y = ");Serial.println(y);
14  analogWrite(led, y);
15 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB4_1.ino
6. ทำการ Compile โค้ด LAB4_1.ino
7. นำไฟล์ LAB4_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


10. จากนั้นพิมพ์โค้ดต่อไปนี้

LAB4_2

```

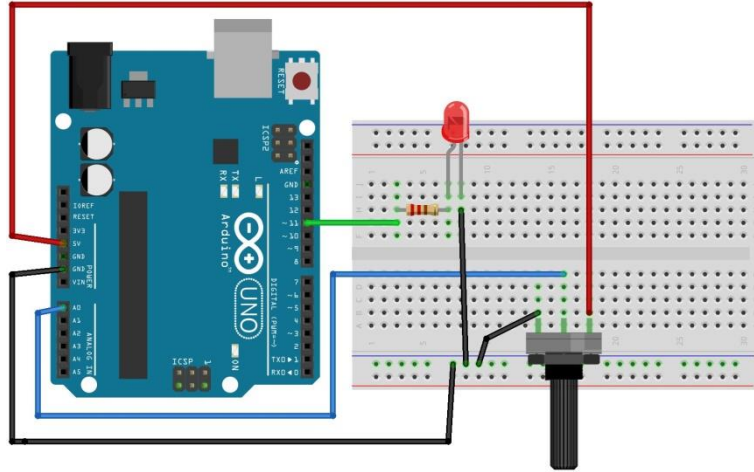
1 int led = 11;
2 void setup()
3 {
4   Serial.begin(9600);
5   pinMode(led, OUTPUT);
6 }

```


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	27

16. ประกอบวงจรรับค่า Analog INPUT มาควบคุม LED โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่

4.2



รูปที่ 4.2 แสดงวงจรรับค่า Analog INPUT มาควบคุม LED

17. Upload โปรแกรม LAB4_1.ino ลงบอร์ด Arduino UNO R3

18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

19. Upload โปรแกรม LAB4_2.ino ลงบอร์ด Arduino UNO R3


20. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

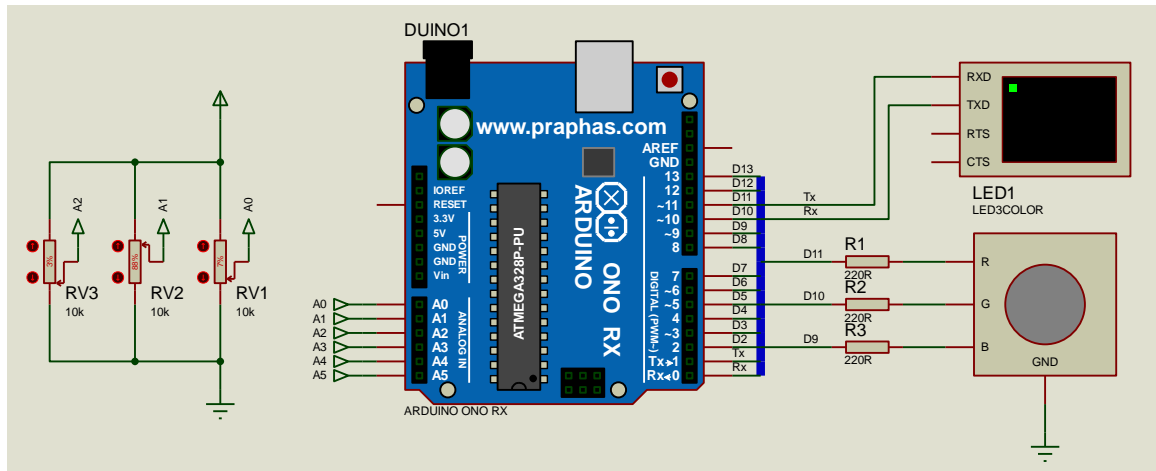
.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	28

ตอนที่ 2 โปรแกรมการรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB

21. ออกแบบวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB แบบไฮบริด

Arduino UNO R3 แสดงดังรูปที่ 4.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 4.3 แสดงวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB


22. บันทึกไฟล์ชื่อ LAB4_2

23. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดต่อไปนี้

```

1 int red = 11;
2 int green = 10;
3 int blue = 9;
4 void setup()
5 {
6   Serial.begin(9600);
7   pinMode (red, OUTPUT);
8   pinMode (green, OUTPUT);
9   pinMode (blue, OUTPUT);
10 }
11 void loop()
12 {
13   int x, y, z, r, s, t;
14   x = analogRead (A0);
15   y = analogRead (A1);
16   z = analogRead (A2);

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	29

```

17  r = map(x, 0, 1023, 0, 255);
18  s = map(y, 0, 1023, 0, 255);
19  t = map(z, 0, 1023, 0, 255);
20  Serial.print("r = ");Serial.println(r);
21  Serial.print("s = ");Serial.println(s);
22  Serial.print("t = ");Serial.println(t);
23  analogWrite(red, r);
24  analogWrite(green, s);
25  analogWrite(blue, t);
26  }

```

24. บันทึกไฟล์โค้ด ชื่อ LAB4_3.ino
25. ทำการ Compile โค้ด LAB4_3.ino
26. นำไฟล์ LAB4_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
27. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
28. สังเกตและบันทึกผลการทดลอง

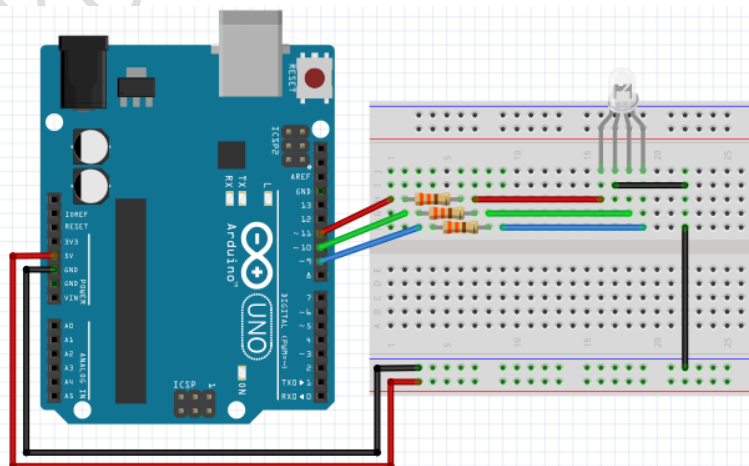
.....

.....


.....

.....

29. ประกอบวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB โดยใช้บอร์ด Arduino UNO R3 ดังรูปที่ 4.4



รูปที่ 4.4 แสดงวงจรรับค่า Analog INPUT มาควบคุมการผสมสีของ LED RGB

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 4	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ	30	

30. Upload โปรแกรม LAB4_3.ino ลงบอร์ด Arduino UNO R3

31. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

32. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	31

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 5 เรื่องงานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ

- อธิบายหลักการทำงานวงจรการรับค่า Push Switch มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานการรับค่า Push Switch มาควบคุม LED แบบต่างๆ ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการรับค่า Push Switch มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรการรับค่า Push Switch มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการรับค่า Push Switch มาควบคุม LED แบบต่างๆ ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

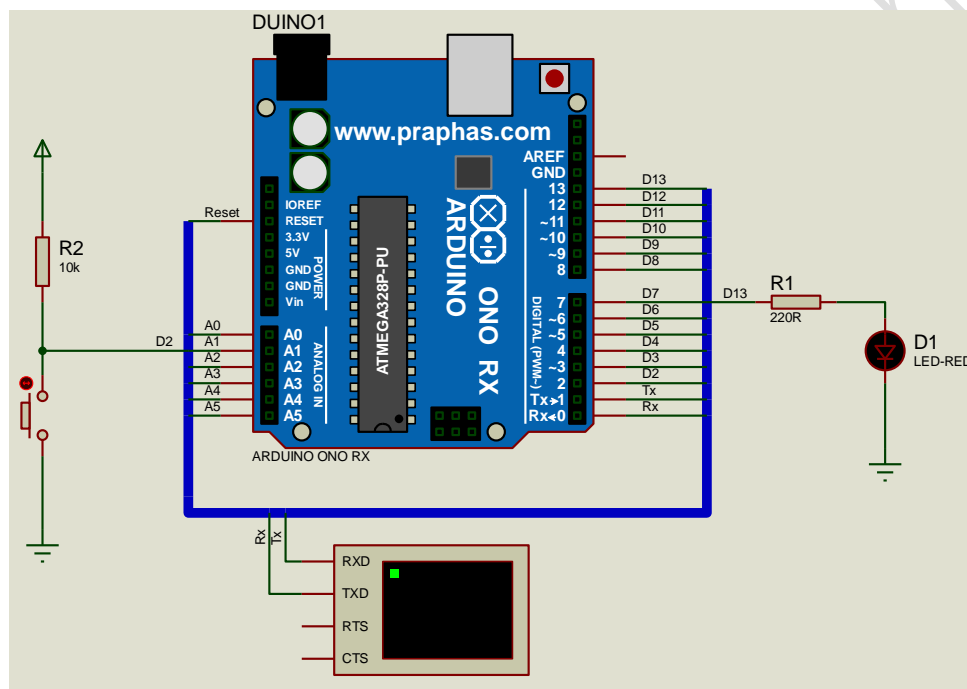
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายไหลต USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	32

ลำดับขั้นการทดลอง

ตอนที่ 1 โปรแกรมการรับค่า Push Switch มาควบคุม LED


1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจร Push Switch มาควบคุม LED แบบไซบอร์ด Arduino UNO R3 แสดงดังรูปที่ 5.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 5.1 แสดงการออกแบบวงจรรับค่า Push Switch มาควบคุม LED

3. บันทึกไฟล์ชื่อ LAB5-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่า Push Switch มาควบคุม LED
ต่อไปนี่

```
Lab5_1
1 int led = 13;
2 int sw = 2;
3 void setup()
4 {
5   Serial.begin(9600);
6   pinMode(led, OUTPUT);
```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	33

```

7   pinMode (sw, INPUT);
8   }
9
10  void loop()
11  {
12    int x;
13    x = digitalRead (sw);
14    Serial.print ("x = ");Serial.println (x);
15    digitalWrite (led, !x);
16  }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB5_1.ino
6. ทำการ Compile โค้ด LAB5_1.ino
7. นำไฟล์ LAB5_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....


.....

10. พิมพ์โค้ดโปรแกรมการรับค่าจาก Push Switch มาควบคุม LED แบบใช้ if ต่อไปนี้

```

LAB5_2§
1  int led = 13;int sw = 2;int ledstatus = LOW;
2  void setup()
3  {
4    Serial.begin (9600);
5    pinMode (led, OUTPUT);
6    pinMode (sw, INPUT);
7  }
8  void loop()
9  {
10     int x = digitalRead (sw);
11     Serial.print ("x = ");Serial.println (x);
12     if (x==0)
13     {
14         delay (350);
15         if (x==0)
16         {
17             ledstatus = !ledstatus;
18             digitalWrite (led, ledstatus);
19         }
20     }
21 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	34

11. บันทึกไฟล์โค้ด ชื่อ LAB5_2.ino
12. ทำการ Compile โค้ด LAB5_2.ino
13. นำไฟล์ LAB5_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

16. พิมพ์โค้ดการรับค่าจาก Push Switch มาควบคุม LED แบบอินเทอร์รัปต์ ต่อไปนี้

Lab5_3§

```

1 int led = 13;
2 int sw = 2;
3 int ledstatus = LOW;
4 void abc()
5 {
6     ledstatus = !ledstatus;
7     digitalWrite(led, ledstatus);
8 }
9 void setup()
10 {
11     pinMode(led, OUTPUT);
12     pinMode(sw, INPUT);
13     attachInterrupt(0, abc, FALLING);
14 }
15 void loop()
16 {
17
18 }

```

17. บันทึกไฟล์โค้ด ชื่อ LAB5_3.ino
18. ทำการ Compile โค้ด LAB5_3.ino
19. นำไฟล์ LAB5_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
20. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	35

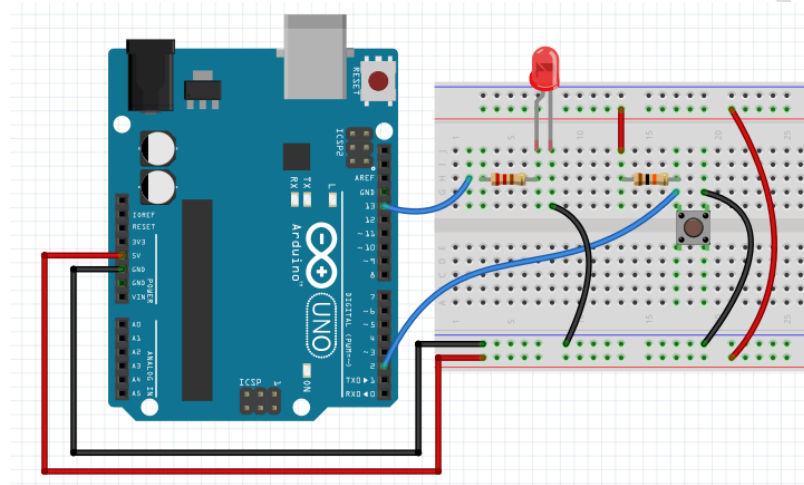
21. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

22. ประกอบวงจรรับค่า Push Switch มาควบคุม LED โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 5.2



รูปที่ 5.2 แสดงวงจรรับค่า Push Switch มาควบคุม LED

23. Upload โปรแกรม LAB5_1.ino ลงบอร์ด Arduino UNO R3

24. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

25. Upload โปรแกรม LAB5_2.ino ลงบอร์ด Arduino UNO R3

26. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	36

27. Upload โปรแกรม LAB5_3.ino ลงบอร์ด Arduino UNO R3

28. สังเกตและบันทึกผลการทดลอง

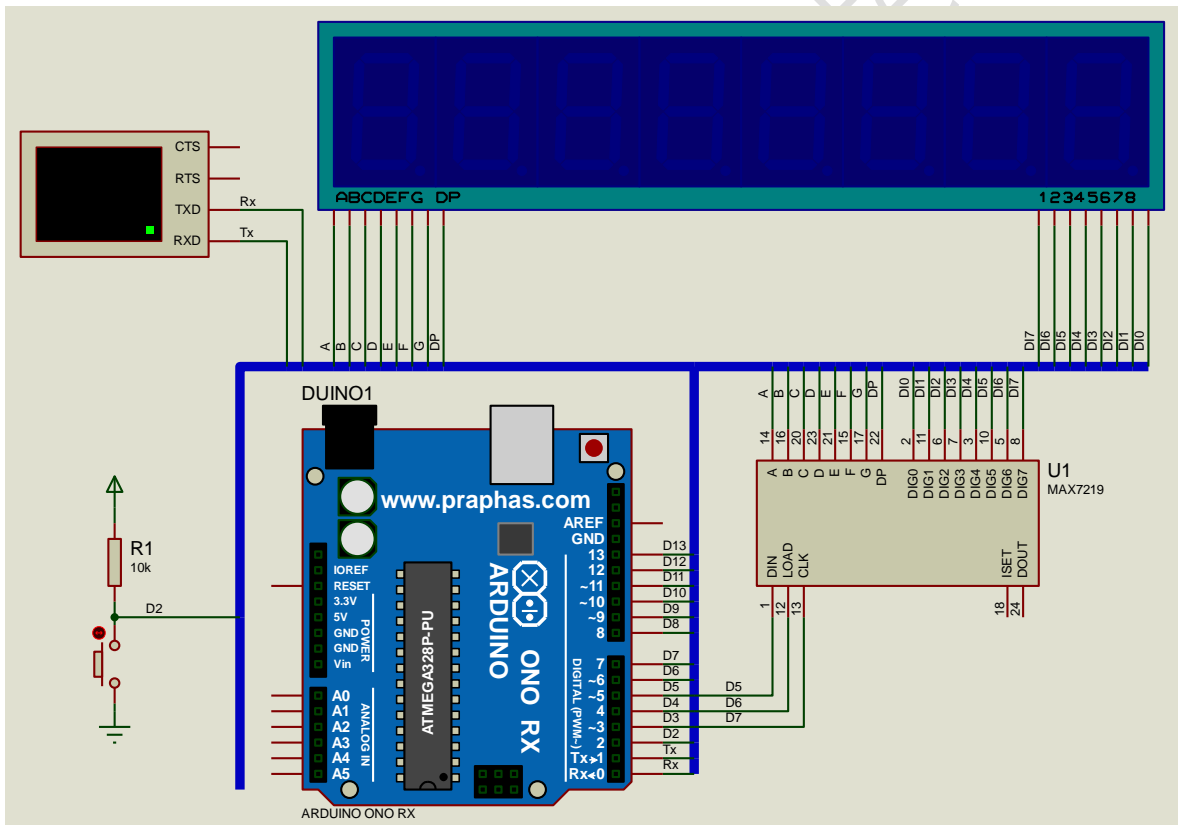
.....

.....

.....

ตอนที่ 2 โปรแกรมการรับค่า Push Switch มาควบคุม LED 7-Segment


29. ออกแบบวงจรรับค่า Push Switch มาควบคุม LED 7-Segment แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 5.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 5.3 แสดงการออกแบบวงจรรับค่า Push Switch มาควบคุม LED 7-Segment

30. บันทึกไฟล์ชื่อ LAB5_2

31. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมการรับค่า Push Switch มาแสดงที่ LED 7-Segment แบบนับขึ้น ต่อไปนี้


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	37

Lab5_4

```

1 #include "LedControl.h"
2 int count=0;
3 int sw = 2;
4 LedControl lc=LedControl(5,7,6,1);
5 // Pin 5->DIN, 7->CLK, 6->CS (LOAD), 1 = No.of devices
6
7 void show3digit(int num)
8 {
9     int seg1,seg2,seg3;
10    seg1 = (num%100)%10;
11    seg2 = (num%100)/10;
12    seg3 = num/100;
13    lc.setDigit(0,0,seg1,false);
14    if (num>=10)
15        lc.setDigit(0,1,seg2,false);
16    if (num>=100)
17        lc.setDigit(0,2,seg3,false);
18 }
19 void setup()
20 {
21     Serial.begin(9600);
22     lc.shutdown(0,false);
23     lc.setIntensity(0,5);
24     lc.clearDisplay(0);
25     pinMode(sw,INPUT);
26 }
27
28 void loop()
29 {
30     int x = digitalRead(sw);
31     if (x==0)
32     {
33         delay(300);
34         if (x==0)
35         {
36             count=count+1;
37         }
38     }
39     if (count>999)
40         count=0;
41     show3digit(count);
42     Serial.print("count = ");Serial.println(count);
43 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	38

32. บันทึกไฟล์โค้ด ชื่อ LAB5_4.ino
33. ทำการ Compile โค้ด LAB5_4.ino
34. นำไฟล์ LAB5_4.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
35. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
36. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


37. โปรแกรม Arduino IDE พิมพ์โค้ดโปรแกรมการรับค่า Push Switch มาแสดงที่ LED 7-Segment แบบสุ่ม ต่อไปนี้

Lab5_5§

```

1 #include "LedControl.h"
2 int count=0;
3 int sw = 2;
4 int number;
5 LedControl lc=LedControl(5,7,6,1);
6 // Pin 5->DIN, 7->CLK, 6->CS(LOAD), 1 = No.of devices
7
8 void show3digit(int num)
9 {
10     int seg1,seg2,seg3;
11     seg1 = (num%100)%10;
12     seg2 = (num%100)/10;
13     seg3 = num/100;
14     lc.setDigit(0,0,seg1,false);
15     if (num>=10)
16         lc.setDigit(0,1,seg2,false);
17     if (num>=100)
18         lc.setDigit(0,2,seg3,false);
19 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	39

```

20 void setup()
21 {
22     lc.shutdown(0, false);
23     lc.setIntensity(0, 5);
24     lc.clearDisplay(0);
25     pinMode(sw, INPUT);
26     randomSeed(analogRead(0));
27     number = random(1000);
28 }
29 void loop()
30 {
31     int x = digitalRead(sw);
32     if (x==0)
33     {
34         delay(300);
35         if (x==0)
36         {
37             number = random(1000);
38         }
39     }
40 show3digit(number);
41 }

```

38. บันทึกไฟล์โค้ด ชื่อ LAB5_5.ino
39. ทำการ Compile โค้ด LAB5_5.ino
40. นำไฟล์ LAB5_5.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
41. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
42. สังเกตและบันทึกผลการทดลอง

.....

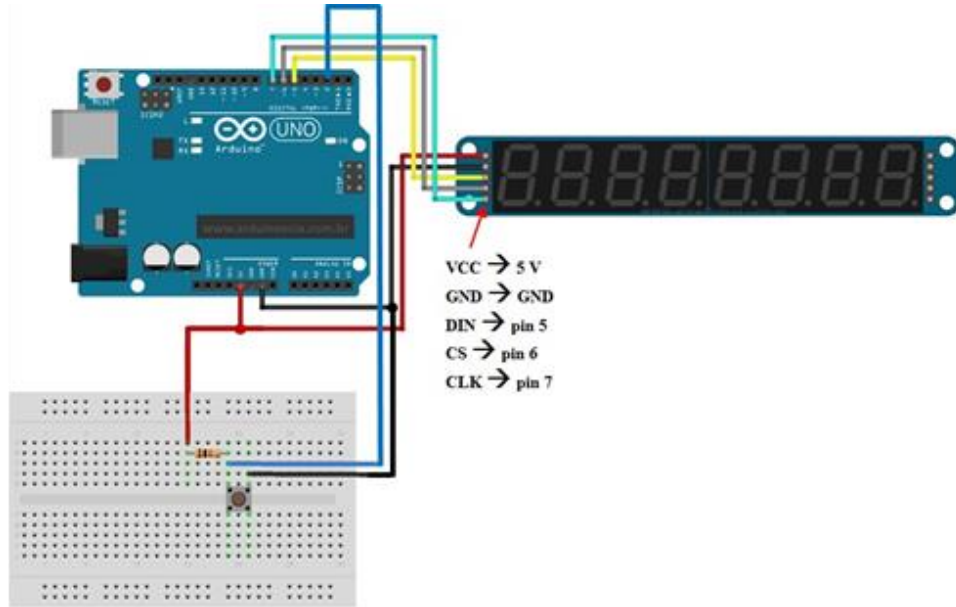
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	40

43. ประกอบวงจรรับค่า Push Switch มาควบคุม LED 7-Segment แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 5.3 ดังรูปที่ 5.4



รูปที่ 5.4 แสดงวงจรรับค่า Push Switch มาควบคุม LED 7-Segment

44. Upload โปรแกรม LAB5_4.ino ลงบอร์ด Arduino UNO R3

45. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

46. Upload โปรแกรม LAB5_5.ino ลงบอร์ด Arduino UNO R3

47. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 5
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ	41

48. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	42

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 6 เรื่องงานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรการรับค่า Keypad มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานการรับค่า Keypad มาแสดงผลที่ LED 7-Segment ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการรับค่า Keypad มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรการรับค่า Keypad มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการรับค่า Keypad มาแสดงผลที่ LED 7-Segment ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

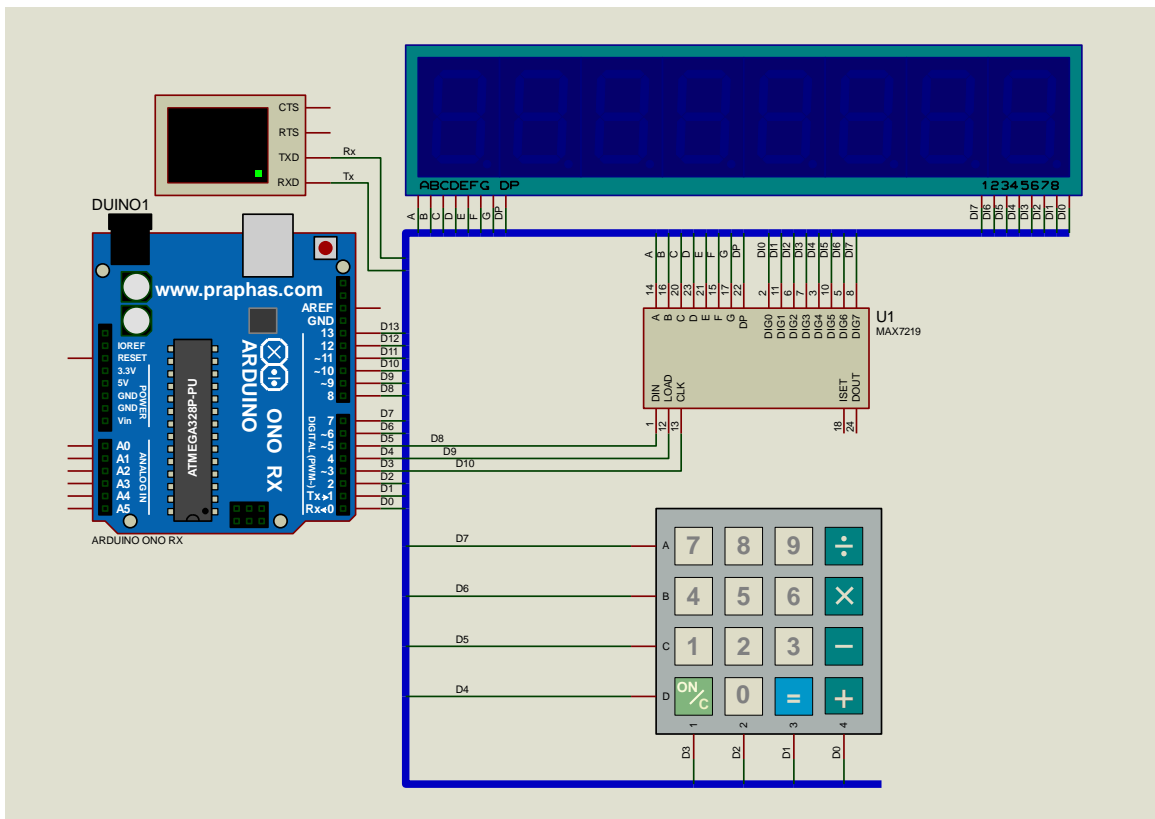
ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายไหลต USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	43


ลำดับขั้นการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment แบบใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 6.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 6.1 แสดงการออกแบบวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment

3. บันทึกไฟล์ชื่อ LAB6-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดรับค่า Keypad มาแสดงผลที่ LED 7-Segment 1 หลัก ต่อไปนี้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	44

Lab6_1

```

1 #include "LedControl.h"
2 #include "Keypad.h"
3 char keys[4][4]={
4   {'7','8','9','A'},
5   {'4','5','6','B'},
6   {'1','2','3','C'},
7   {'E','0','F','D'}};
8 byte rowPins[] = {7,6,5,4};
9 byte colPins[] = {3,2,1,0};
10 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
11 LedControl lc=LedControl(8,10,9,1);
12 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
13 void setup()
14 {
15   lc.shutdown(0,false);
16   lc.setIntensity(0,5);
17   lc.clearDisplay(0);
18 }
19 void loop()
20 {
21   char key = keypad.getKey();
22   if (key != NO_KEY)
23   {
24     lc.setChar(0,0,key,false);
25   }
26 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB6_1.ino
6. ทำการ Compile โค้ด LAB6_1.ino
7. นำไฟล์ LAB6_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	45


10. พิมพ์โค้ดโปรแกรมการรับค่า Keypad มาแสดงผลที่ LED 7-Segment 4 หลัก ต่อไปนี้

```

Lab6_2§
1 #include "LedControl.h"
2 #include "Keypad.h"
3 char keys[4][4]={
4   {'7','8','9','A'},
5   {'4','5','6','B'},
6   {'1','2','3','C'},
7   {'*','0','#','D'}};
8 byte rowPins[] = {7,6,5,4};
9 byte colPins[] = {3,2,1,0};
10 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
11 LedControl lc=LedControl(8,10,9,1);
12 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
13 int m=7;
14 void setup()
15 {
16   lc.shutdown(0, false);
17   lc.setIntensity(0,5);
18   lc.clearDisplay(0);
19 }
20 void loop()
21 {
22   char key = keypad.getKey();
23   if (key != NO_KEY)
24   {
25     if (m==7)
26       lc.clearDisplay(0);
27     lc.setChar(0,m, key, false);
28     m = m-1;
29     if (m < 0)
30       m = 7;
31   }
32 }

```

11. บันทึกไฟล์โค้ด ชื่อ LAB6_2.ino
12. ทำการ Compile โค้ด LAB6_2.ino
13. นำไฟล์ LAB6_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มตนการจำลองการทำงาน

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	46

15. สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

16. พิมพ์โค้ดโปรแกรมการรับเลขรหัสผ่าน 4 หลัก ต่อไปนี้

```

Lab6_3
1 #include "LedControl.h"
2 #include "Keypad.h"
3 char keys[4][4]={
4   {'7','8','9','A'},
5   {'4','5','6','B'},
6   {'1','2','3','C'},
7   {'*','0','#','D'}};
8 byte rowPins[] = {7,6,5,4};
9 byte colPins[] = {3,2,1,0};
10 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
11 LedControl lc=LedControl(8,10,9,1);
12 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
13 int m=3;
14 char num[4]; //num[0] to num[3]
15 void setup()
16 {
17   lc.shutdown(0,false);
18   lc.setIntensity(0,5);
19   lc.clearDisplay(0);
20 }
21 void loop()
22 {
23   char key = keypad.getKey();
24   if (key != NO_KEY)
25   {
26     if (m==3)
27       lc.clearDisplay(0);
28     lc.setChar(0,m,key,false);
29     num[m]=key;
30     m = m-1;
31     if (m < 0)
32     {
33       if ((num[3]=='4') && (num[2]=='0') && (num[1]=='6') && (num[0]=='9'))
34       {
35         lc.setChar(0,7,'0',false);
36         lc.setChar(0,6,'P',false);
37       }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	47

```

38     else
39     {
40         lc.setChar(0,7,'-',false);
41         lc.setChar(0,6,'-',false);
42     }
43     m = 3;
44 }
45 }
46 }

```

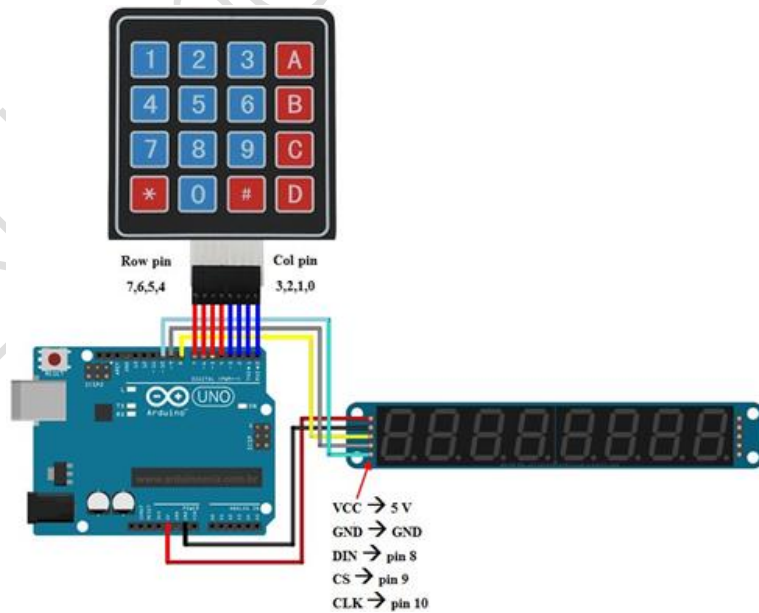
17. บันทึกไฟล์โค้ด ชื่อ LAB6_3.ino
18. ทำการ Compile โค้ด LAB6_3.ino
19. นำไฟล์ LAB6_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
20. จากนั้นกดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
21. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

22. ประกอบวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 6.2



รูปที่ 6.2 แสดงวงจรรับค่า Keypad มาแสดงผลที่ LED 7-Segment

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 6	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment	48	

23. Upload โปรแกรม LAB6_1.ino ลงบอร์ด Arduino UNO R3

24. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

25. Upload โปรแกรม LAB6_2.ino ลงบอร์ด Arduino UNO R3

26. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

27. Upload โปรแกรม LAB6_3.ino ลงบอร์ด Arduino UNO R3

28. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

29. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	49

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 7 เรื่องโปรแกรมควบคุม Buzzer ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


1. อธิบายหลักการทำงานของวงจรควบคุม Buzzer ได้อย่างถูกต้อง
2. ออกแบบและจำลองการทำงานของวงจรควบคุม Buzzer ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
3. ประกอบและติดตั้งวงจรควบคุม Buzzer ได้อย่างถูกต้อง
4. เขียนโปรแกรมควบคุมวงจรรวม Buzzer ได้อย่างถูกต้อง
5. ทดสอบและบำรุงรักษา อุปกรณ์ วงจรควบคุม Buzzer ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

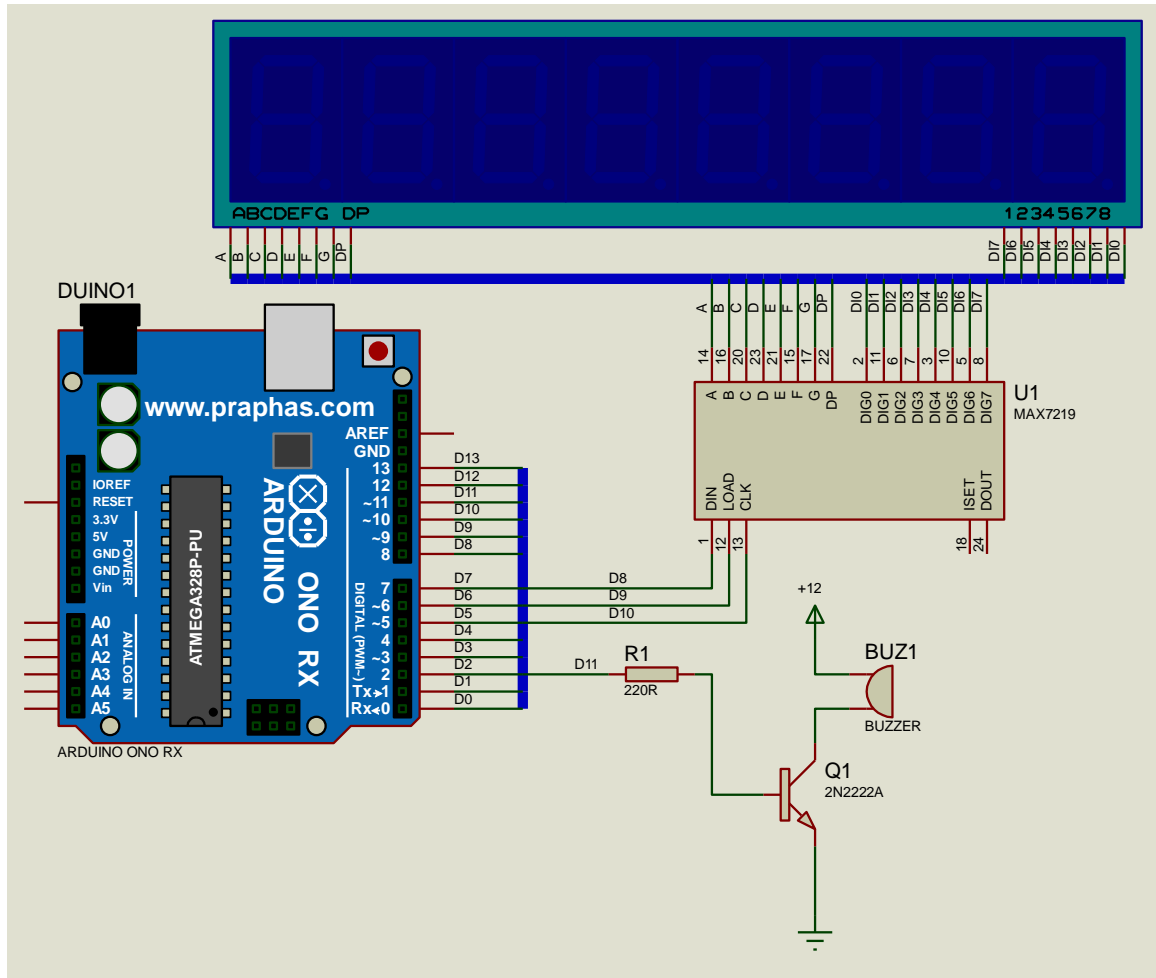
ข้อห้ามและข้อควรระวัง

1. ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
2. ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
3. ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั่ว ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
4. ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
5. ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	50


ลำดับขั้นการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจร LED 7-Segment และ Buzzer โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 7.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 7.1 แสดงการออกแบบวงจร LED 7-Segment และ Buzzer

3. บันทึกไฟล์ชื่อ LAB7-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมเล่นเสียง โด เร มี ฟา ซอล ลา ที่ ต่อไปนี้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	51


Lab7_1

```

1 #include "LedControl.h"
2 #define C4 262
3 #define D4 294
4 #define E4 330
5 #define F4 349
6 #define G4 392
7 #define A4 440
8 #define B4 494
9 #define C5 523
10 int melody[] = {C4,D4,E4,F4,G4,A4,B4,C5};
11 float beats[] ={1,1,1,1,1,1,1,1};
12 int buzzerpin = 11;
13 int timestop = 70;
14 LedControl lc=LedControl(8,10,9,1);
15 // Pin 8->DIN, 10->CLK, 9->CS(LOAD), 1 = No.of devices
16 void setup()
17 {
18   lc.shutdown(0,false);
19   lc.setIntensity(0,5);
20   lc.clearDisplay(0);
21   int dl = 500;
22   pinMode(buzzerpin,OUTPUT);
23   int numnote;
24   numnote = sizeof(melody)/2;
25   for (int i=0;i<numnote;i++)
26   {
27     lc.setChar(0,7-i,'-',false);
28     tone(buzzerpin, melody[i],dl*beats[i]);
29     delay(dl*beats[i]);
30     digitalWrite(buzzerpin,HIGH);
31     delay(timestop);
32   }
33 }
34 void loop()
35 {
36 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB7_1.ino
6. ทำการ Compile โค้ด LAB7_1.ino
7. นำไฟล์ LAB7_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	52

9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


.....

10. พิมพ์โค้ดโปรแกรมเล่นเพลง Happy birth day ต่อไปนี้

```

Lab7_2 pitches.h
1 #include "LedControl.h"
2 #define C4 262
3 #define D4 294
4 #define E4 330
5 #define F4 349
6 #define FS4 370
7 #define G4 392
8 #define A4 440
9 #define B4 494
10 #define C5 523
11 #define D5 587
12 int melody[] = {D4, D4, E4, D4, G4, FS4, D4, D4, E4, D4, A4, G4, D4, D4, D5,
13   B4, G4, FS4, E4, C5, C5, B4, G4, A4, G4};
14 float beats[] = {0.5, 0.5, 1, 1, 1, 2, 0.5, 0.5, 1, 1, 1, 2, 0.5, 0.5, 1, 1,
15   1, 1, 1, 0.5, 0.5, 1, 1, 1, 2};
16 int buzzerpin = 11;
17 int timestop = 70;
18 LedControl lc=LedControl(8,10,9,1);
19 // Pin 8->DIN, 10->CLK, 9->CS(LOAD), 1 = No.of devices
20 void setup()
21 {
22   lc.shutdown(0, false);
23   lc.setIntensity(0, 5);
24   lc.clearDisplay(0);
25   int dl = 400;
26   pinMode(buzzerpin, OUTPUT);
27   lc.setChar(0, 7, 'H', false);
28   lc.setChar(0, 6, 'b', false);
29   lc.setChar(0, 5, 'd', false);
30   int numnote;
31   numnote = sizeof(melody)/2;

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	53

```

32 for (int i=0;i<numnote;i++)
33 {
34     tone(buzzerpin, melody[i],dl*beats[i]);
35     delay(dl*beats[i]);
36     digitalWrite(buzzerpin,HIGH);
37     delay(timestop);
38 }
39 }
40 void loop()
41 {}

```

11. บันทึกไฟล์โค้ด ชื่อ LAB7_2.ino
12. ทำการ Compile โค้ด LAB7_2.ino
13. นำไฟล์ LAB7_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

16. พิมพ์โค้ดโปรแกรมเล่นเพลง หนุมานีสี่ ต่อไปนี้

```

Lab7_3$ pitches.h
1 #include "LedControl.h"
2 #define C4 262
3 #define D4 294
4 #define E4 330
5 #define F4 349
6 #define G4 392
7 int melody[] = {E4, D4, C4, D4, E4, E4, E4, D4, D4, D4, E4, G4, G4,
8                 E4, D4, C4, D4, E4, E4, E4, D4, D4, E4, D4, C4};
9 float beats[] = {1, 1, 1, 1, 1, 1, 2, 1, 1, 2, 1, 1, 2, 1,
10                 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 4};
11 int buzzerpin = 11;
12 int timestop = 70;
13 LedControl lc=LedControl(8,10,9,1);
14 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	54

```

15 void setup()
16 {
17   lc.shutdown(0, false);
18   lc.setIntensity(0, 5);
19   lc.clearDisplay(0);
20   int dl = 250;
21   pinMode(buzzerpin, OUTPUT);
22   int numnote;
23   numnote = sizeof(melody)/2;
24   for (int i=0;i<numnote;i++)
25   {
26     lc.setDigit(0, 0, beats[i], false);
27     tone(buzzerpin, melody[i], dl*beats[i]);
28     delay(dl*beats[i]);
29     digitalWrite(buzzerpin, HIGH);
30     delay(timestop);
31   }
32 }
33 void loop()
34 {
35 }

```

17. บันทึกไฟล์โค้ด ชื่อ LAB7_3.ino
18. ทำการ Compile โค้ด LAB7_3.ino
19. นำไฟล์ LAB7_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
20. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
21. สังเกตและบันทึกผลการทดลอง

.....

.....


.....

.....

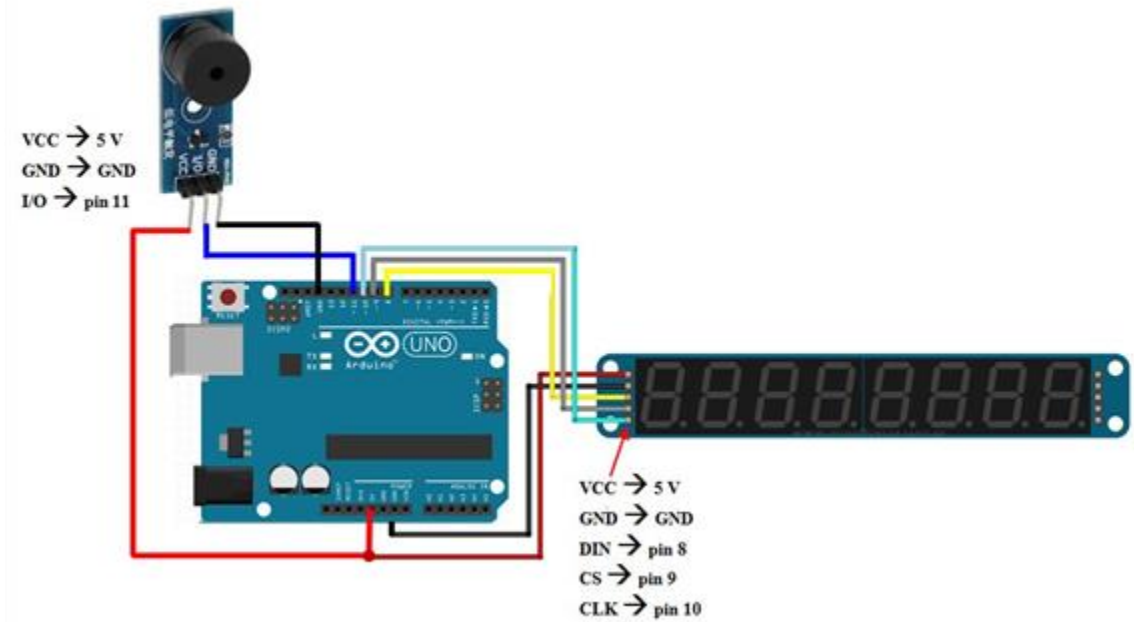
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	55

22. ประกอบวงจร LED 7-Segment และ Buzzer โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 7.2



รูปที่ 7.2 แสดงวงจร LED 7-Segment และ Buzzer

23. Upload โปรแกรม LAB7_1.ino ลงบอร์ด Arduino UNO R3

24. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

25. Upload โปรแกรม LAB7_2.ino ลงบอร์ด Arduino UNO R3

26. สังเกตและบันทึกผลการทดลอง


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 7	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมควบคุม Buzzer	56	


27. Upload โปรแกรม LAB7_3.ino ลงบอร์ด Arduino UNO R3

28. สังเกตและบันทึกผลการทดลอง

.....
.....
.....
.....

29. สรุปผลการทดลอง

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	57

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 8 เรื่องงานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรควบคุม Buzzer ร่วมกับ Keypad ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรควบคุม Buzzer ร่วมกับ Keypad ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรควบคุม Buzzer ร่วมกับ Keypad ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรควบคุม Buzzer ร่วมกับ Keypad ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรควบคุม Buzzer ร่วมกับ Keypad ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

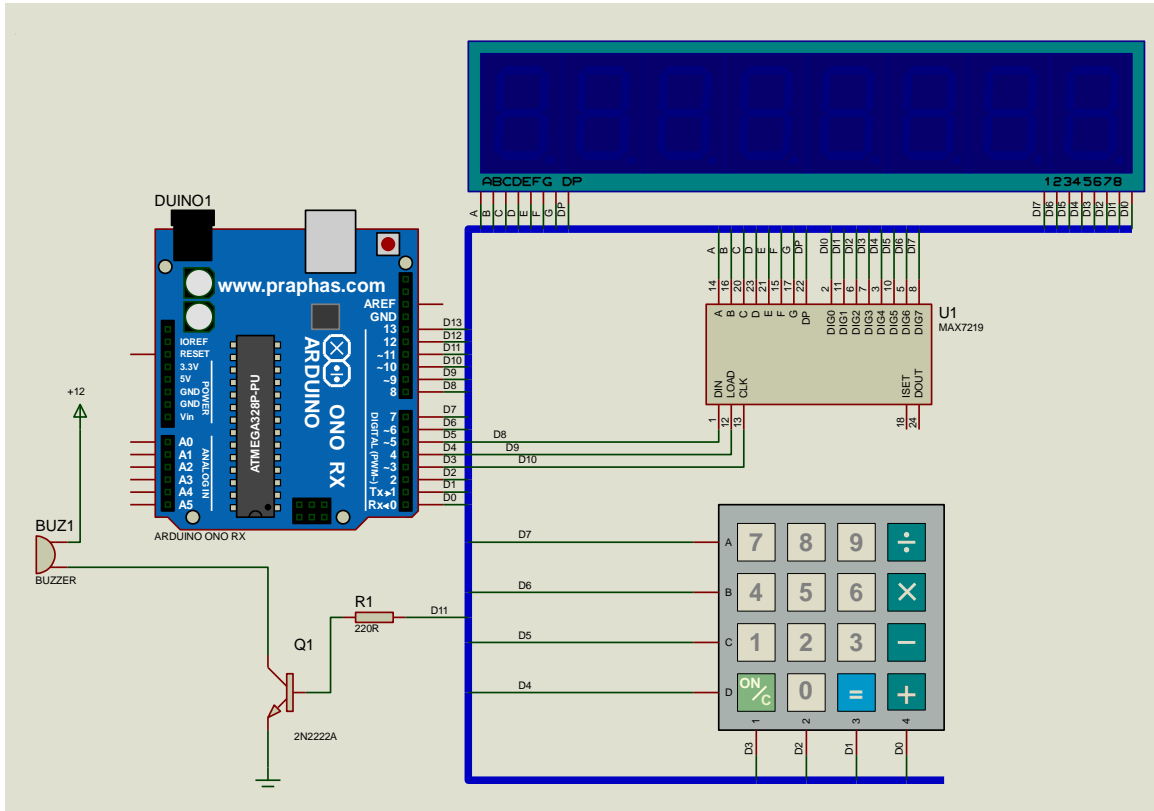
ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายไหลต USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	58

ลำดับขั้นการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรควบคุม Buzzer ร่วมกับ Keypad โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 8.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 8.1 แสดงการออกแบบวงจรควบคุม Buzzer ร่วมกับ Keypad


3. บันทึกไฟล์ชื่อ LAB8-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมเล่นน้ำตามการกด keypad

Lab8_1

```

1 #include "LedControl.h"
2 #include "Keypad.h"
3 #define C4 262
4 #define D4 294
5 #define E4 330
6 #define F4 349

```



	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	59

```

7 #define G4 392
8 #define A4 440
9 #define B4 494
10 #define C5 523
11 char keys[4][4]={
12     {7,8,9,0},
13     {4,5,6,0},
14     {1,2,3,0},
15     {0,0,0,0}};
16 byte rowPins[] = {7,6,5,4};
17 byte colPins[] = {3,2,1,0};
18 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
19 int melody[] = {C4,D4,E4,F4,G4,A4,B4,C5};
20 int dl = 300;
21 int buzzerpin = 11;
22 LedControl lc=LedControl(8,10,9,1);
23 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
24 void setup()
25 {
26     lc.shutdown(0,false);
27     lc.setIntensity(0,5);
28     lc.clearDisplay(0);
29     pinMode(buzzerpin,OUTPUT);
30     digitalWrite(buzzerpin,HIGH);
31 }
32 void loop()
33 {
34     char key = keypad.getKey();
35     if ((key != NO_KEY) && (key < 9))
36     {
37         lc.setDigit(0,0,key,false);
38         tone(buzzerpin, melody[key-1],dl);
39         delay(dl);
40         digitalWrite(buzzerpin,HIGH);
41     }
42 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB8_1.ino
6. ทำการ Compile โค้ด LAB8_1.ino
7. นำไฟล์ LAB8_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	60

9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


.....

10. พิมพ์โค้ดโปรแกรมเกมทายตัวเลข 00 ถึง 99 ต่อไปนี้

```

Lab8_2
1 #include "LedControl.h"
2 #include "Keypad.h"
3 #define A3 220
4 #define B3 247
5 #define C4 262
6 #define D4 294
7 #define E4 330
8 #define F4 349
9 #define G4 392
10 #define A4 440
11 #define ST 0
12 int melody[] = {G4,G4,G4,G4,G4,G4,E4,D4,E4,G4,C4,G4,E4,
13 D4,E4,C4,D4,C4,A3,C4,C4,C4,A3,C4,ST,C4,A3,C4,C4,C4,A3,
14 C4,ST,G4,A4,G4,E4,D4,C4,ST};
15 float beats[] = {0.5,0.5,0.5,0.5,1.5,0.5,0.5,0.5,0.5,0.5,
16 1.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,
17 0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,1,1,1,1};
18 int buzzerpin = 11;
19 int timestop = 70;
20 char keys[4][4]={
21   {'7','8','9','0'},
22   {'4','5','6','0'},
23   {'1','2','3','0'},
24   {'0','0','0','0'}};
25 byte rowPins[] = {7,6,5,4};
26 byte colPins[] = {3,2,1,0};
27 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
28 LedControl lc=LedControl(8,10,9,1);
29 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
30 int m=1;
31 int num[2]; //num[0] to num[1]
32 char number;
33 int pw;


```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	61

```

34 void setup()
35 {
36   lc.shutdown(0, false);
37   lc.setIntensity(0, 5);
38   lc.clearDisplay(0);
39   pinMode(buzzerpin, OUTPUT);
40   digitalWrite(buzzerpin, HIGH);
41   randomSeed(analogRead(0));
42   pw = random(100);
43 }
44 void loop()
45 {
46   char key = keypad.getKey();
47   if (key != NO_KEY)
48   {
49     if (m==1)
50       lc.clearDisplay(0);
51     lc.setChar(0, m, key, false);
52     num[m]=key;
53     m = m-1;
54     if (m < 0)
55     {
56       delay(500);
57       number = ((num[1]-48)*10)+(num[0]-48);
58       if (number == pw)
59       {
60         lc.setChar(0, 7, '0', false);
61         lc.setChar(0, 6, 'H', false);
62         int dl = 350;
63         int numnote;
64         numnote = sizeof(melody)/2;
65         for (int i=0; i<numnote; i++)
66         {
67           if (melody[i]>0)
68             tone(buzzerpin, melody[i], dl*beats[i]);
69           delay(dl*beats[i]);
70           digitalWrite(buzzerpin, HIGH);
71           delay(timestop);
72         }
73       }
74       else if (number < pw)
75       {
76         lc.setChar(0, 7, 'P', false);
77         lc.setChar(0, 6, 'P', false);

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	62

```

78     }
79     else if (number > pw)
80     {
81         lc.setChar(0,7,'L',false);
82         lc.setChar(0,6,'L',false);
83     }
84     m = 1;
85 }
86 }
87 }

```

11. บันทึกไฟล์โค้ด ชื่อ LAB8_2.ino
12. ทำการ Compile โค้ด LAB8_2.ino
13. นำไฟล์ LAB8_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มตนการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

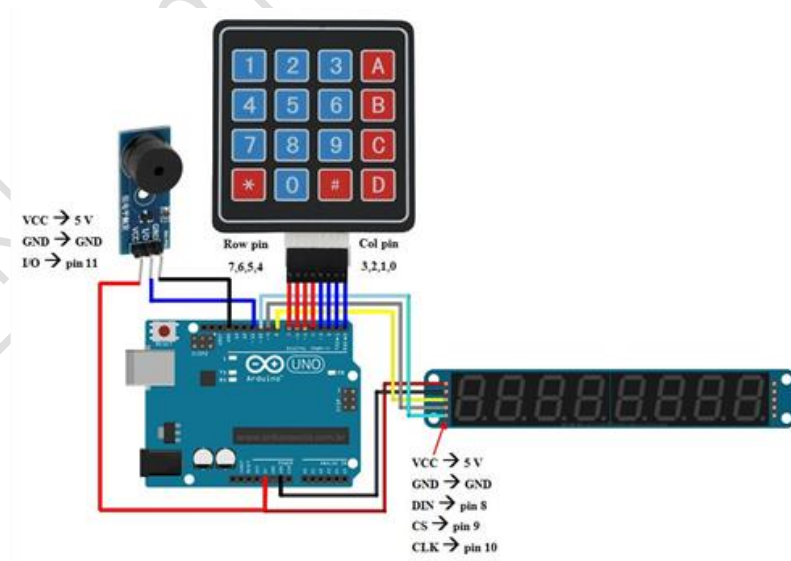
.....

.....


.....

.....

16. ประกอบควบคุม Buzzer ร่วมกับ Keypad โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 7.2



รูปที่ 8.2 แสดงวงจรควบคุม Buzzer ร่วมกับ Keypad

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 8	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad	63	

17. Upload โปรแกรม LAB8_1.ino ลงบอร์ด Arduino UNO R3

18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

19. Upload โปรแกรม LAB8_2.ino ลงบอร์ด Arduino UNO R3

20. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

21. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	64	

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 9 เรื่องงานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจรรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้

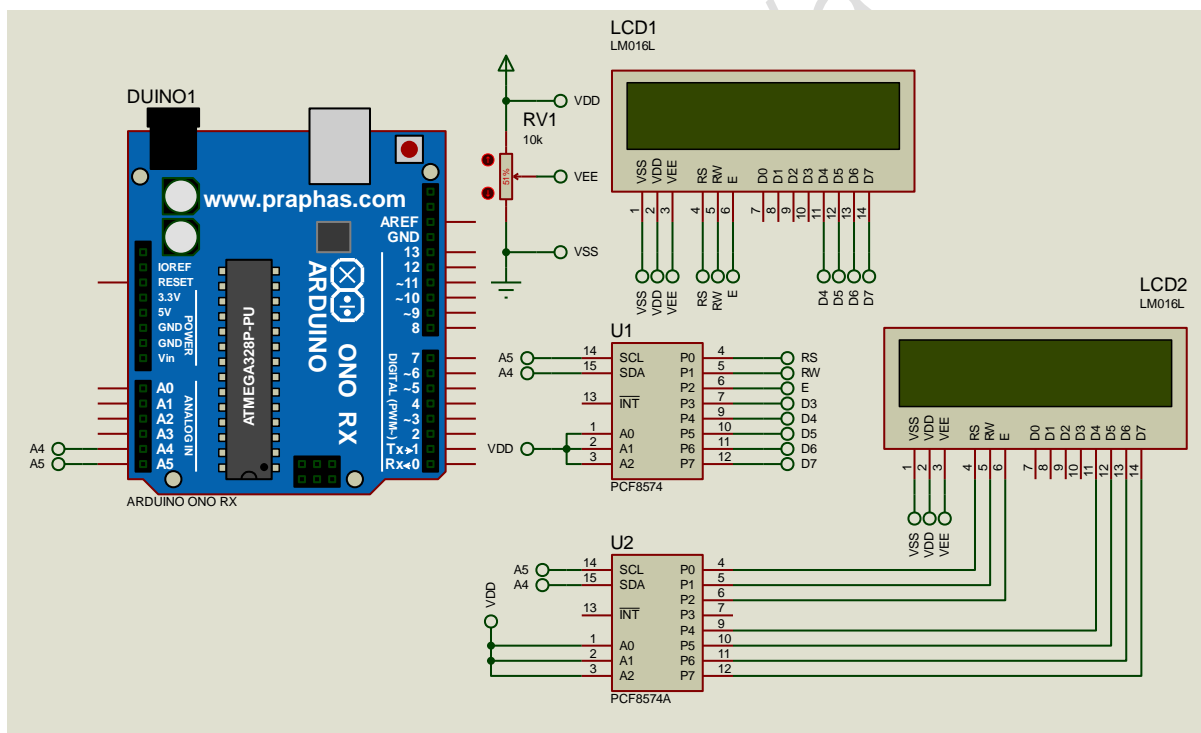
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	

- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

ลำดับขั้นการทดลอง


ตอนที่ 1 โปรแกรมแสดงข้อมูลที่ LCD

- เปิดโปรแกรม Proteus 7 Professional
- ออกแบบวงจรแสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 9.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 9.1 แสดงการออกแบบวงจรแสดงข้อมูลที่ LCD

- บันทึกไฟล์ชื่อ LAB9-1
- เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงข้อมูลที่ LCD

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	66

Lab9_1 §

```

1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3 LiquidCrystal_I2C lcd(0x27,16,2);
4 void setup()
5 {
6   lcd.backlight();
7   lcd.begin(); //LiquidCrystal_I2C
8 }
9 void loop()
10 {
11   lcd.setCursor(0,0);
12   lcd.print("Hello LCD I2C");
13   lcd.setCursor(0,1);
14   lcd.print("Pattayatech");
15 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB9_1.ino
6. ทำการ Compile โค้ด LAB9_1.ino
7. นำไฟล์ LAB9_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	67

10. พิมพ์โค้ดโปรแกรมการแสดงผลค่าตัวเลขที่เก็บอยู่ในตัวแปรต่อไปนี้

```

Lab9_2§
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int x=48;
4 int y=-79;
5 float z=8.74586;
6 void setup()
7 {
8 lcd.begin();
9 }
10 void loop()
11 {
12 lcd.setCursor(0,0);
13 lcd.print("x = ");
14 lcd.print(x);
15 lcd.setCursor(8,0);
16 lcd.print("y = ");
17 lcd.print(y);
18 lcd.setCursor(0,1);
19 lcd.print("z = ");
20 lcd.print(z,4);
21 }

```

11. บันทึกไฟล์โค้ด ชื่อ LAB9_2.ino
12. ทำการ Compile โค้ด LAB9_2.ino
13. นำไฟล์ LAB9_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	68

16. พิมพ์โค้ดโปรแกรมการแสดงผลสัญลักษณ์พิเศษที่สร้างขึ้นมาเองต่อไปนี้

```

Lab9_3
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 byte heart[8] = {0x00,0x0A,0x1F,0x1F,0x0E,0x04,0x00,0x00};
4 byte smile[8] = {0x00,0x11,0x00,0x00,0x11,0x0E,0x00,0x00};
5 void setup()
6 {
7   lcd.begin();
8   lcd.createChar(0,heart);
9   lcd.createChar(1,smile);
10 }
11 void loop()
12 {
13   lcd.setCursor(0,0);
14   lcd.write(0);
15   lcd.setCursor(0,1);
16   lcd.write(1);
17 }

```

17. บันทึกไฟล์โค้ด ชื่อ LAB9_3.ino
18. ทำการ Compile โค้ด LAB9_3.ino
19. นำไฟล์ LAB9_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
20. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
21. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	69

22. พิมพ์โค้ดโปรแกรมแสดงข้อความที่เลื่อนไปมาได้ต่อไปนี้

```

Lab9_4
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 void setup()
4 {
5   lcd.begin();
6 }
7 void loop()
8 {
9   lcd.home();
10  lcd.print("ECS");
11  for (int i=0;i<13;i++)
12  {
13    delay(400);
14    lcd.scrollDisplayRight();
15  }
16  for (int i=0;i<13;i++)
17  {
18    delay(400);
19    lcd.scrollDisplayLeft();
20  }
21 }

```

23. บันทึกไฟล์โค้ด ชื่อ LAB9_4.ino
24. ทำการ Compile โค้ด LAB9_4.ino
25. นำไฟล์ LAB9_4.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
26. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
27. สังเกตและบันทึกผลการทดลอง


.....

.....

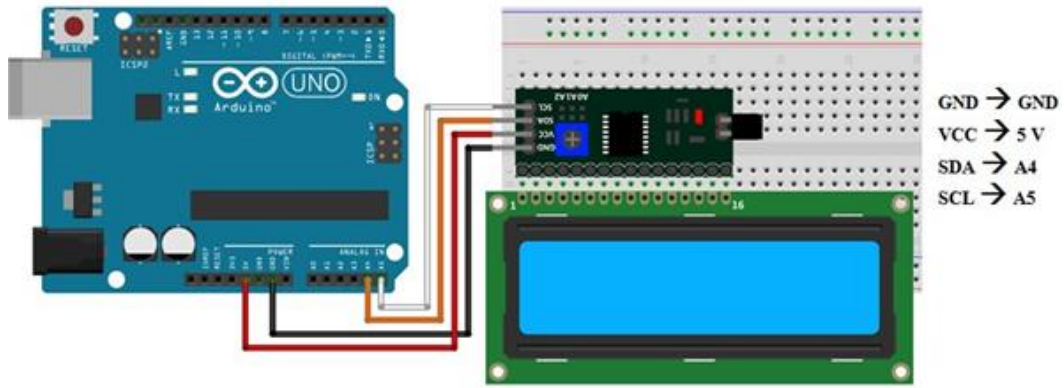
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	70

28. ประกอบวงจรแสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 9.2



รูปที่ 9.2 แสดงแสดงข้อมูลที่ LCD

29. Upload โปรแกรม LAB9_1.ino ลงบอร์ด Arduino UNO R3

30. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

31. Upload โปรแกรม LAB9_2.ino ลงบอร์ด Arduino UNO R3

32. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	71

33. Upload โปรแกรม LAB9_1.ino ลงบอร์ด Arduino UNO R3

34. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

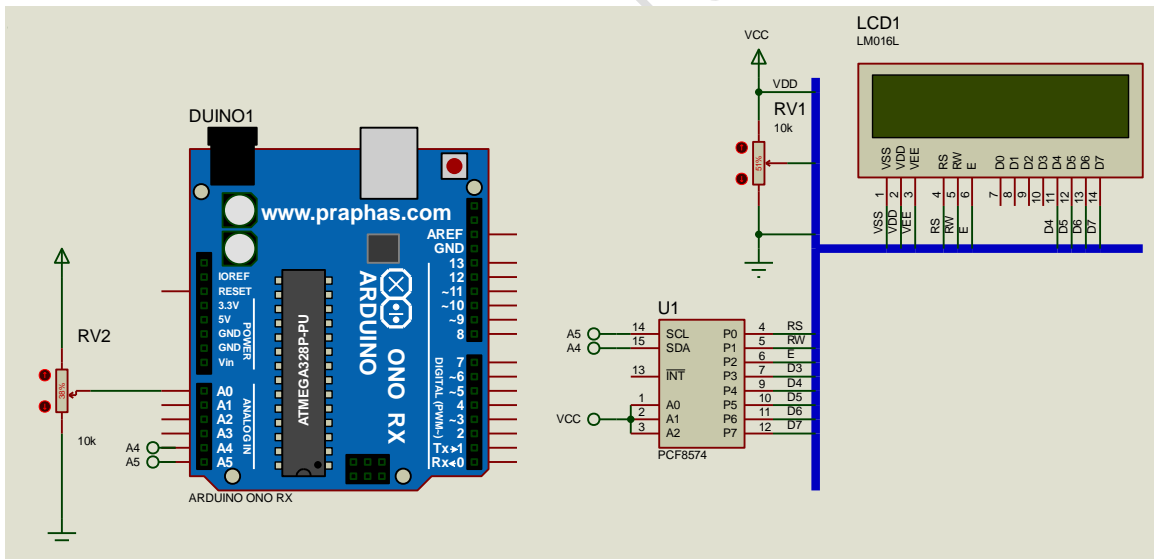
.....

ตอนที่ 2 โปรแกรมรับค่า Analog แสดงข้อมูลที่ LCD

35. เปิดโปรแกรม Proteus 7 Professional

36. ออกแบบวงจรรับค่า Analog แสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่


9.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 9.3 แสดงการออกแบบวงจรรับค่า Analog แสดงข้อมูลที่ LCD

37. บันทึกไฟล์ชื่อ LAB9-2

38. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงค่าที่อ่านได้จาก Analog Input

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	72

Lab9_5


```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aVal=0,tempaVal=0;
4 void setup()
5 {
6 lcd.begin();
7 }
8 void loop()
9 {
10 aVal = analogRead(A0);
11 lcd.setCursor(0,0);
12 if (tempaVal != aVal)
13 {
14   lcd.clear();
15   lcd.print(aVal);
16   delay(300);
17 }
18 tempaVal = aVal;
19 }

```

39. บันทึกไฟล์โค้ด ชื่อ LAB9_5.ino
40. ทำการ Compile โค้ด LAB9_5.ino
41. นำไฟล์ LAB9_5.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
42. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
43. สังเกตและบันทึกผลการทดลอง

44. พิมพ์โค้ดโปรแกรมการแสดงผลเครื่องหมาย * ตามการหมุนตัวด้านทานปรับค่าได้ต่อไปนี้

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	73

Lab9_6.ino

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aVal=0,tempx=0,x;
4 void setup()
5 {
6   lcd.begin();
7 }
8 void loop()
9 {
10  aVal = analogRead(A0);
11  lcd.setCursor(0,0);
12  x = map(aVal,0,1023,1,17);
13  if (tempx != x)
14  {
15    lcd.clear();
16    for(int i=0;i<x;i++)
17    {
18      lcd.print("*");
19    }
20  }
21  tempx = x;
22 }

```


45. บันทึกไฟล์โค้ด ชื่อ LAB9_6.ino
46. ทำการ Compile โค้ด LAB9_6.ino
47. นำไฟล์ LAB9_6.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
48. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
49. สังเกตและบันทึกผลการทดลอง

.....

.....

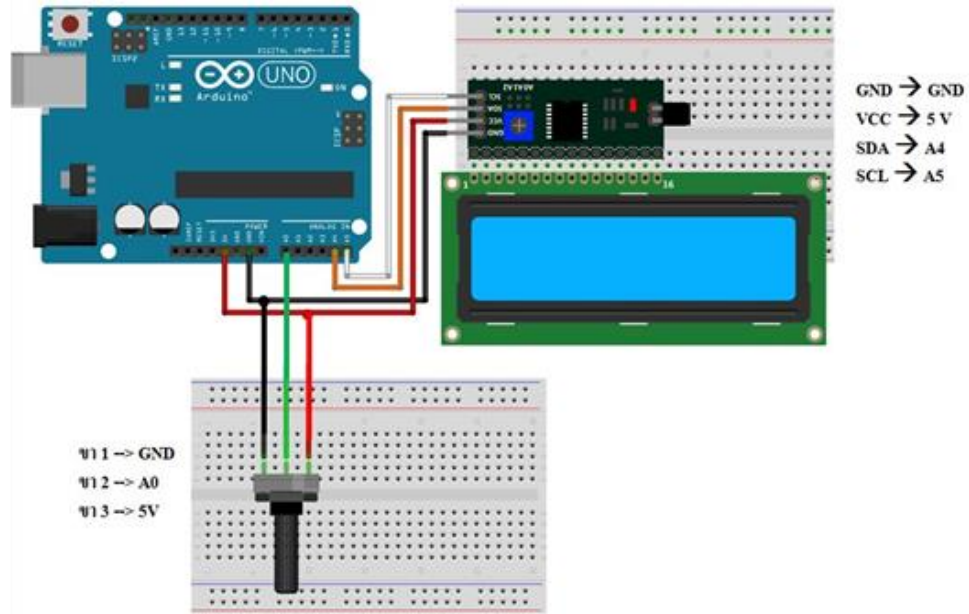
.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	74

50. ประกอบวงจรจรรยาับค่า Analog แสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่

9.4



รูปที่ 9.4 แสดงวงจรรับค่า Analog แสดงข้อมูลที่ LCD

51. Upload โปรแกรม LAB9_5.ino ลงบอร์ด Arduino UNO R3

52. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

53. Upload โปรแกรม LAB9_6.ino ลงบอร์ด Arduino UNO R3


54. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

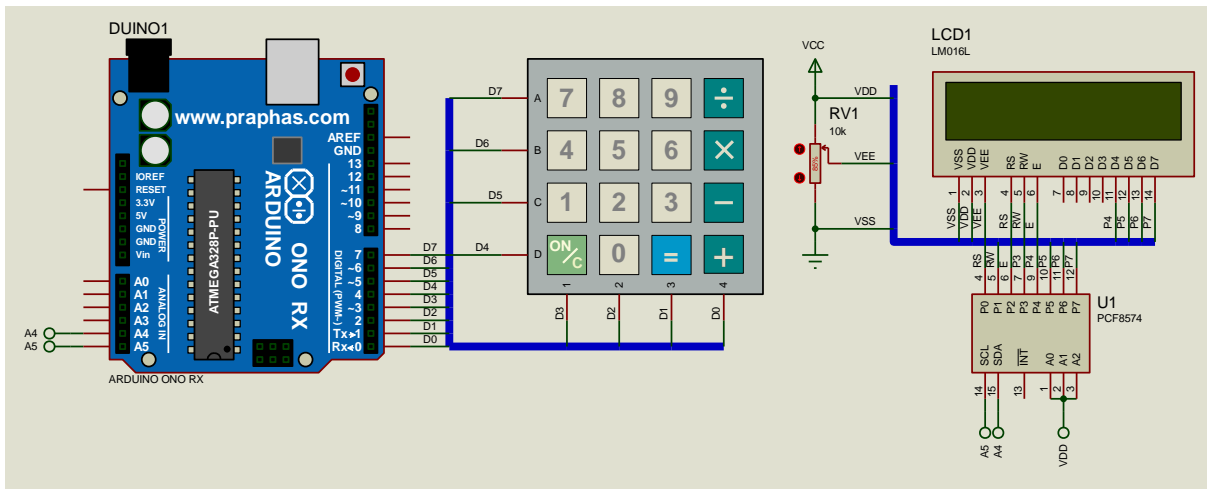
.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	

ตอนที่ 3 โปรแกรมรับค่าจาก keypad แสดงข้อมูลที่ LCD

55. เปิดโปรแกรม Proteus 7 Professional

56. ออกแบบวงจรรับค่า จาก keypad แสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 9.5 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 9.5 แสดงการออกแบบวงจรรับค่าจาก keypad แสดงข้อมูลที่ LCD


57. บันทึกไฟล์ชื่อ LAB9-3

58. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงค่าที่รับมาจาก keypad

```

Lab9_7
1 #include <Keypad.h>
2 #include <LiquidCrystal_I2C.h>
3 LiquidCrystal_I2C lcd(0x27,16,2);
4 char keys[4][4] = {
5   {'7','8','9','A'},
6   {'4','5','6','B'},
7   {'1','2','3','C'},
8   {'*','0','#','D'}};
9 byte rowPins[4] = {7,6,5,4};
10 byte colPins[4] = {3,2,1,0};
11 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
12 void setup()
13 {
14   lcd.begin();
15   lcd.setCursor(0,0);
16 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	76

```

17 void loop()
18 {
19   char key = keypad.getKey();
20   if (key != NO_KEY)
21   {
22     lcd.print(key);
23   }
24 }

```

59. บันทึกไฟล์โค้ด ชื่อ LAB9_7.ino
60. ทำการ Compile โค้ด LAB9_7.ino
61. นำไฟล์ LAB9_7.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
62. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
63. สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

64. พิมพ์โค้ดโปรแกรมรหัสผ่าน 4 หลักต่อไปนี้

```

Lab9_8
1 #include <Keypad.h>
2 #include <LiquidCrystal_I2C.h>
3 LiquidCrystal_I2C lcd(0x27,16,2);
4 char keys[4][4] = {
5   {'1','2','3','A'},
6   {'4','5','6','B'},
7   {'7','8','9','C'},
8   {'*','0','#','D'}};
9 byte rowPins[4] = {7,6,5,4};
10 byte colPins[4] = {3,2,1,0};
11 char num[4];
12 int m=3;
13 Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
14 void setup()
15 {
16   lcd.begin();
17   lcd.setCursor(0,0);

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	77

```

18  lcd.print("PW : ");
19  lcd.setCursor(5,0);
20  lcd.cursor();
21  }
22  void loop()
23  {
24  char key = keypad.getKey();
25  if ((key != NO_KEY) && (key>=48) && (key<=57))
26  {
27  lcd.print(key);
28  num[m]=key;
29  m--;
30  if(m<0)
31  {
32  delay(500);
33  lcd.setCursor(0,1);
34  if ((num[3]=='4') && (num[2]=='5') && (num[1]=='9') && (num[0]=='6'))
35  {
36  lcd.print("OPEN");
37  }
38  else
39  {
40  lcd.print("CLOSE");
41  }
42  delay(1500);
43  m=3;
44  lcd.clear();
45  lcd.setCursor(0,0);
46  lcd.print("PW : ");
47  lcd.setCursor(5,0);
48  }
49  }
50  }


```

65. บันทึกไฟล์โค้ด ชื่อ LAB9_8.ino

66. ทำการ Compile โค้ด LAB9_8.ino

67. นำไฟล์ LAB9_8.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional

68. จากนั้นให้กดปุ่ม Play เพื่อเริ่มตนการจำลองการทำงาน

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	78

69. สังเกตและบันทึกผลการทดลอง

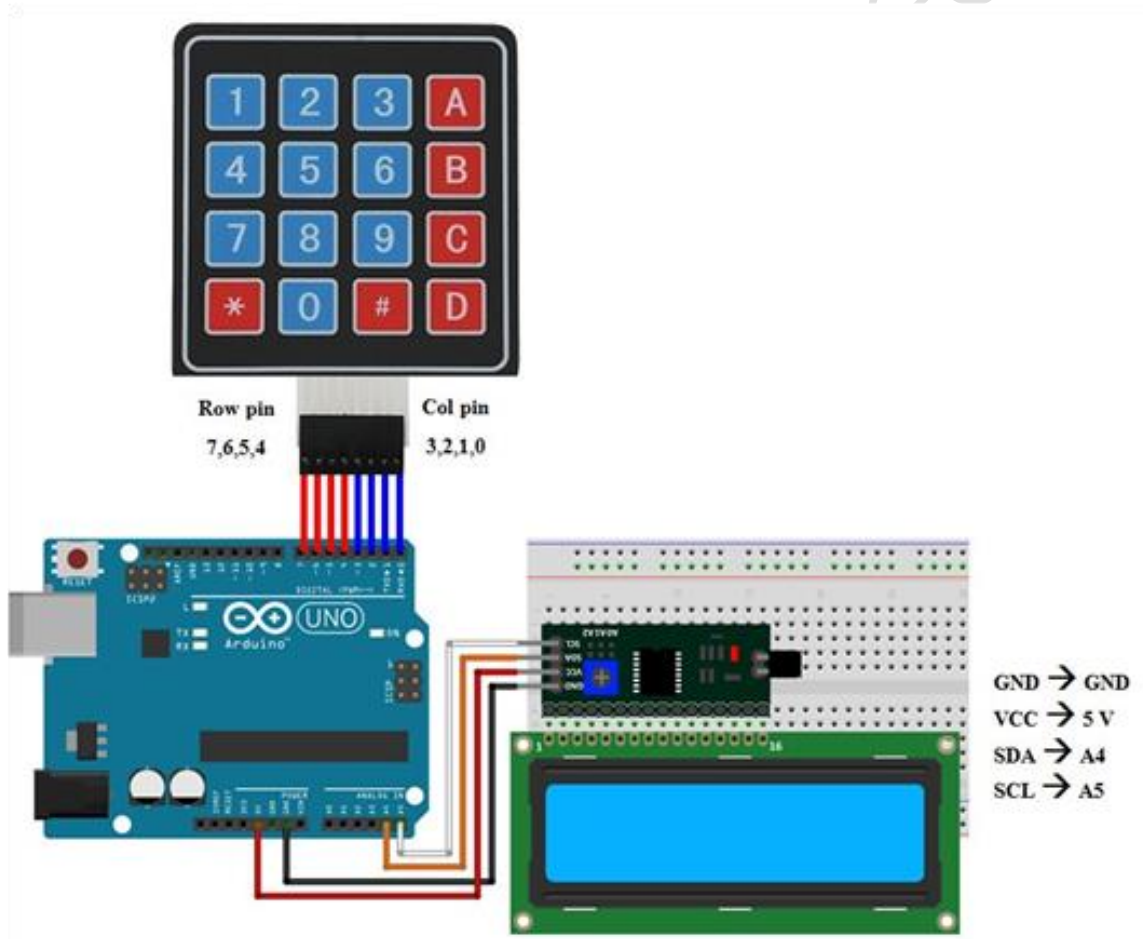
.....

.....


.....

.....

70. ประกอบวงจรรับค่าจาก keypad แสดงข้อมูลที่ LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 9.6



รูปที่ 9.6 แสดงวงจรรับค่าจาก keypad แสดงข้อมูลที่ LCD

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 9
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD	79

71. Upload โปรแกรม LAB9_7.ino ลงบอร์ด Arduino UNO R3

72. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

73. Upload โปรแกรม LAB9_8.ino ลงบอร์ด Arduino UNO R3

74. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

75. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	80

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 10 เรื่องงานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

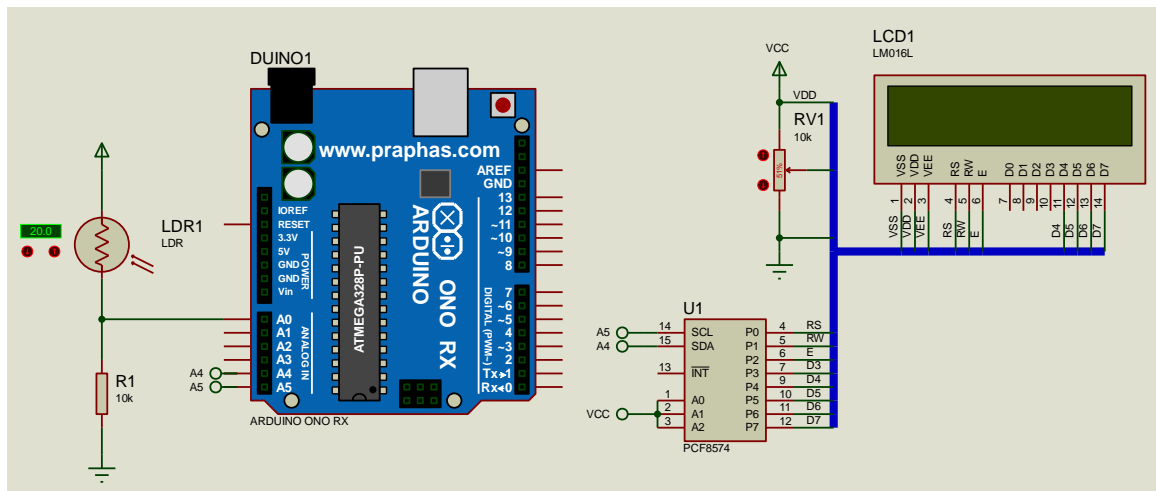
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทั้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	81

ลำดับขั้นการทดลอง

ตอนที่ 1 โปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบ วงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 10.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 10.1 แสดงการออกแบบวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD


3. บันทึกไฟล์ชื่อ LAB10-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงค่าจาก LDR แบบแบ่งแรงดัน

Lab10_1

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aVal=0,tempaVal=0;
4 void setup()
5 {
6   lcd.begin();
7 }
8 void loop()
9 {
10  aVal = analogRead(A0);
11

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	82

```

12  if (tempaVal != aVal)
13  {
14      lcd.clear();
15      lcd.setCursor(0,0);
16      lcd.print("LDR");
17      lcd.setCursor(0,1);
18      lcd.print(aVal);
19      delay(300);
20  }
21  tempaVal = aVal;
22  }

```

- บันทึกไฟล์โค้ด ชื่อ LAB10_1.ino
- ทำการ Compile โค้ด LAB10_1.ino
- นำไฟล์ LAB10_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
- จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
- สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

- พิมพ์โค้ดโปรแกรมการรับค่าจาก LDR มาตรวจสอบเงื่อนไขต่อไปนี้

```

Lab10_2§
1  #include <LiquidCrystal_I2C.h>
2  LiquidCrystal_I2C lcd(0x27,16,2);
3  int aVal=0,tempaVal=0;
4  void setup()
5  {
6      lcd.begin();
7  }
8  void loop()
9  {
10     aVal = analogRead(A0);

```


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	83

```

11  if (tempaVal != aVal)
12  {
13      lcd.clear();
14      lcd.setCursor(0,0);
15      lcd.print(aVal);
16      lcd.setCursor(0,1);
17      if (aVal<100)
18          lcd.print("Dark");
19      else
20          lcd.print("Light");
21      delay(300);
22  }
23  tempaVal = aVal;
24  }

```

11. บันทึกไฟล์โค้ด ชื่อ LAB10_2.ino
12. ทำการ Compile โค้ด LAB10_2.ino
13. นำไฟล์ LAB10_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

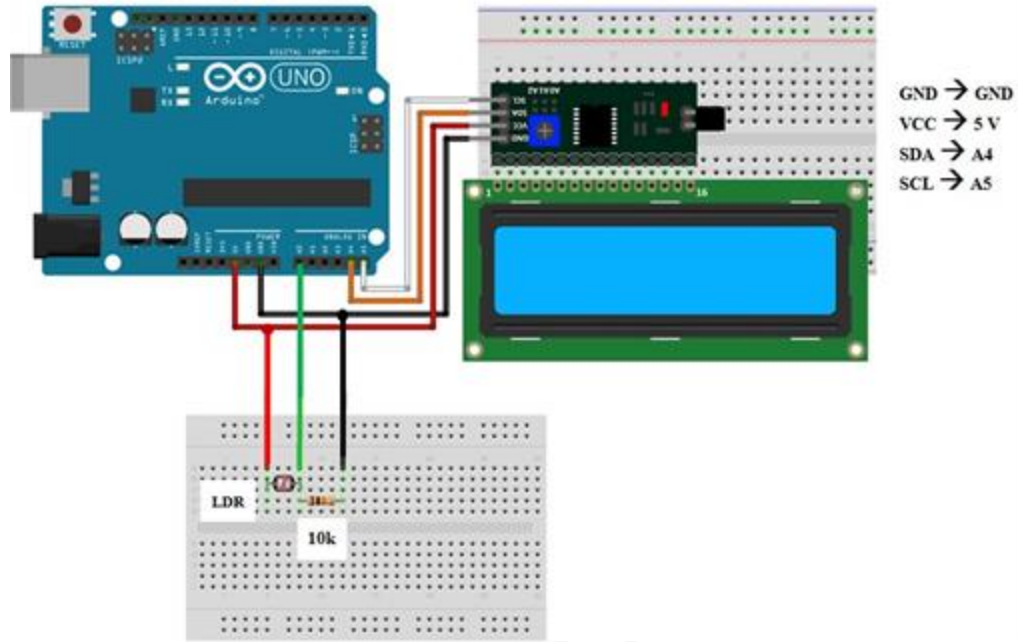
.....

.....

.....

16. ประกอบวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 10.2

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	84



รูปที่ 10.2 แสดงวงจรการเชื่อมต่อกับ LDR แสดงผลด้วย LCD

17. Upload โปรแกรม LAB10_1.ino ลงบอร์ด Arduino UNO R3

18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

19. Upload โปรแกรม LAB10_2.ino ลงบอร์ด Arduino UNO R3

20. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 10
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD	85

21. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	86

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 11 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือชิปต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้

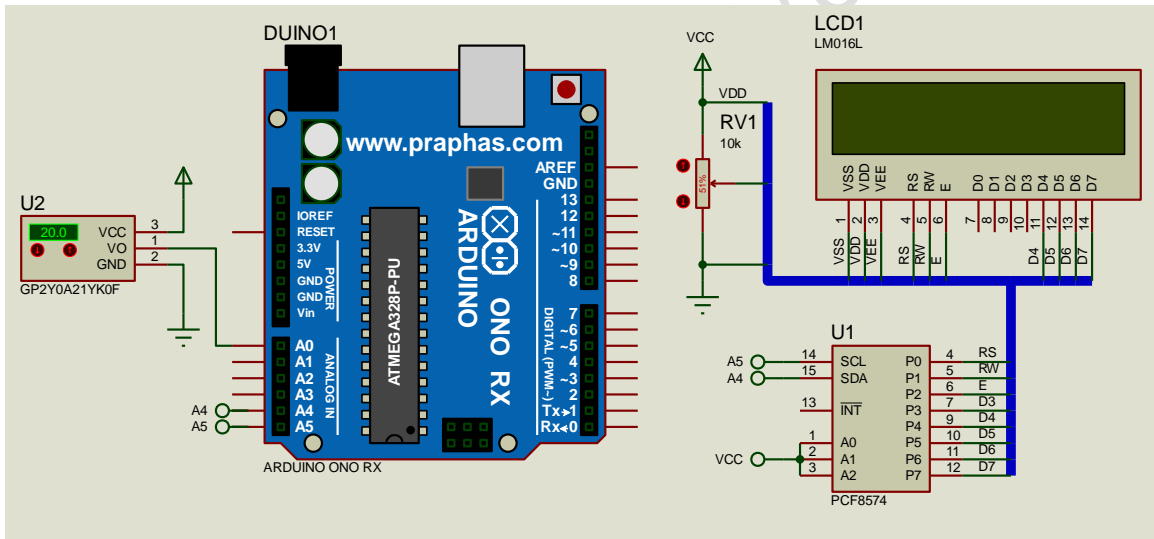
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	87

- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

ลำดับขั้นการทดลอง


ตอนที่ 1 โปรแกรมการเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21

- เปิดโปรแกรม Proteus 7 Professional
- ออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 11.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 11.1 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21

- บันทึกไฟล์ชื่อ LAB11-1
- เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงค่า 0-1023 ที่อ่านได้จาก GP2Y0A21

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	88

Lab11_1

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aval,temp_aval=0;
4 void setup()
5 {
6   lcd.begin();
7   lcd.setCursor(0,0);
8   lcd.print("aval = ");
9 }
10 void loop()
11 {
12   aval = analogRead(A0);
13   if (temp_aval != aval)
14   {
15     lcd.setCursor(7,0);
16     lcd.print("  ");
17   }
18   lcd.setCursor(7,0);
19   lcd.print(aval);
20   temp_aval = aval;
21   delay(300);
22 }

```


5. บันทึกไฟล์โค้ด ชื่อ LAB11_1.ino
6. ทำการ Compile โค้ด LAB11_1.ino
7. นำไฟล์ LAB11_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	89

10. พิมพ์โค้ดโปรแกรมแสดงข้อความ Hi_Pattayatech เมื่อระยะทางอยู่ในช่วงที่กำหนดต่อไปนี้

Lab11_2§


```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aval,temp_aval=0;
4 void setup()
5 {
6   lcd.begin();
7   lcd.setCursor(0,0);
8   lcd.print("aval = ");
9 }
10 void loop()
11 {
12   aval = analogRead(A0);
13   if (temp_aval != aval)
14   {
15     lcd.setCursor(7,0);
16     lcd.print("  ");
17   }
18   lcd.setCursor(7,0);
19   lcd.print(aval);
20   if (aval>400)
21   {
22     lcd.setCursor(0,1);
23     lcd.print("Hi_Pattayatech");
24   }
25   else
26   {
27     lcd.setCursor(0,1);
28     lcd.print("-----");
29   }
30   temp_aval = aval;
31   delay(300);
32 }

```

11. บันทึกไฟล์โค้ด ชื่อ LAB11_2.ino

12. ทำการ Compile โค้ด LAB11_2.ino

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21	90

13. นำไฟล์ LAB11_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

.....

.....

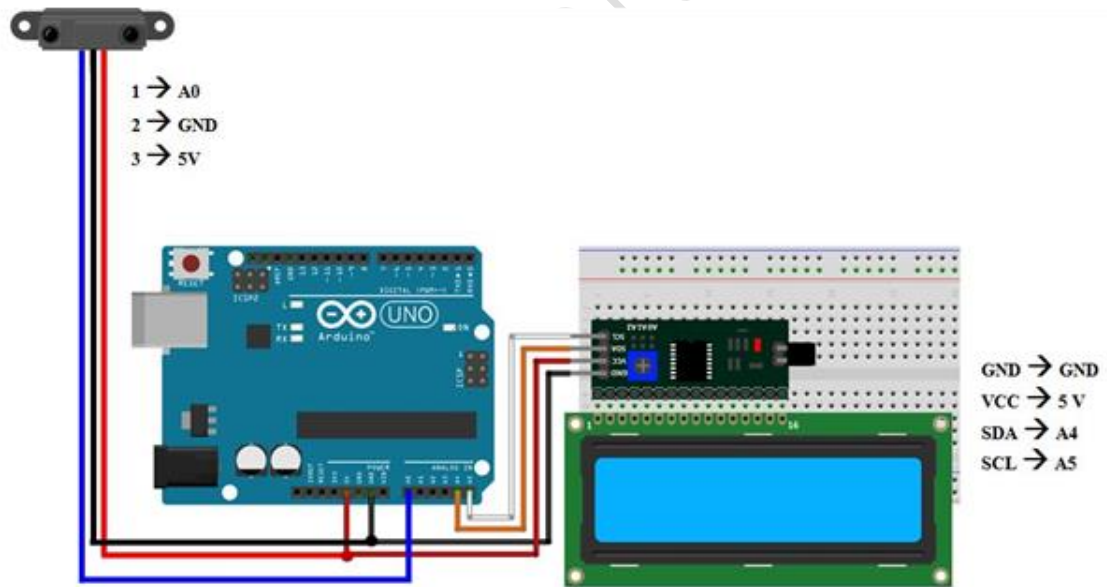
.....

.....

.....


.....

16. ประกอบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21 โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 11.2



รูปที่ 11.2 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21

17. Upload โปรแกรม LAB11_1.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 11
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟาเรด GP2Y0A21	91

18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

19. Upload โปรแกรม LAB11_2.ino ลงบอร์ด Arduino UNO R3

20. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

21. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	92

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 11 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ วงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือชิปต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้

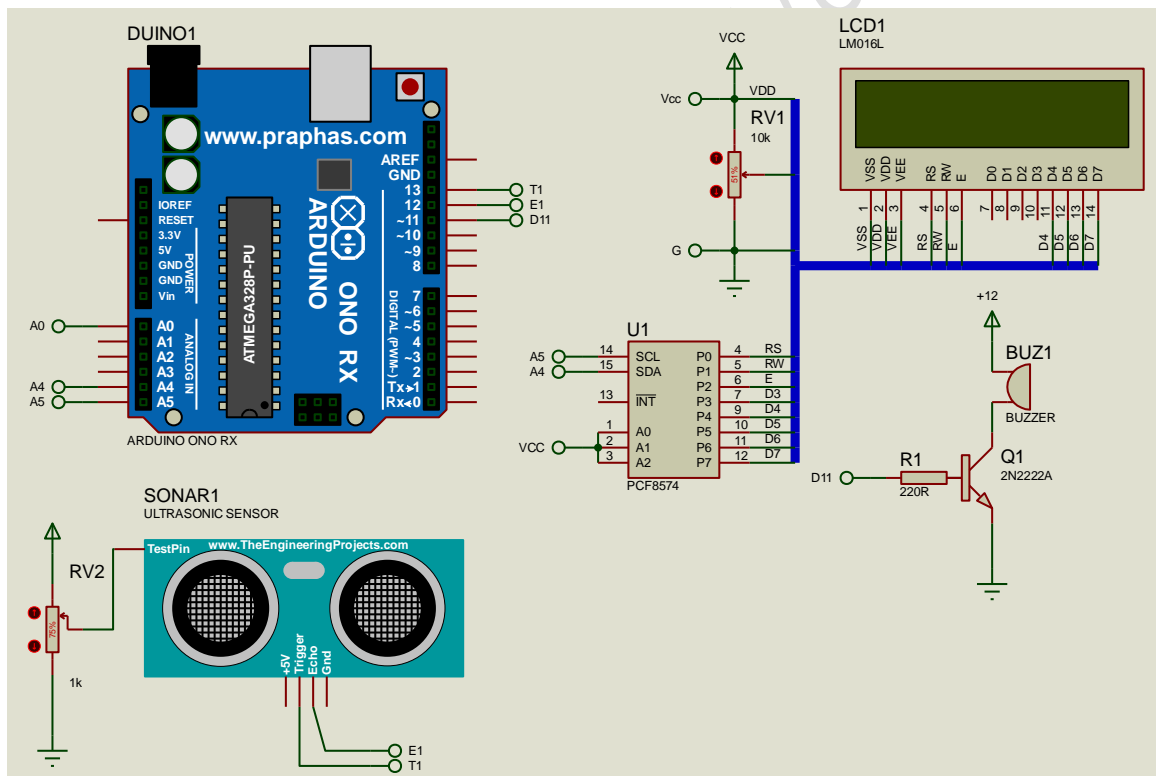
	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	

- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

ลำดับขั้นการทดลอง


ตอนที่ 1 โปรแกรมการเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04

- เปิดโปรแกรม Proteus 7 Professional
- ออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 12.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 12.1 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04

- บันทึกไฟล์ชื่อ LAB12-1

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	94

4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแสดงค่าระยะทางที่อ่านได้ HC-SR04


Lab12_1

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int triggerPin = 13;
4 int echoPin = 12;
5 long duration,distCM,temp_distCM=0;
6 void setup()
7 {
8   lcd.begin();
9   lcd.setCursor(0,0);
10  lcd.print("Dist =      cm");
11  pinMode(triggerPin,OUTPUT);
12  pinMode(echoPin,INPUT);
13 }
14 void loop()
15 {
16   digitalWrite(triggerPin,LOW);
17   delayMicroseconds(2);
18   digitalWrite(triggerPin,HIGH);
19   delayMicroseconds(10);
20   digitalWrite(triggerPin,LOW);
21   duration = pulseIn(echoPin,HIGH);
22   distCM = duration/58;
23   if (temp_distCM != distCM)
24   {
25     lcd.setCursor(7,0);
26     lcd.print("    ");
27   }
28   lcd.setCursor(7,0);
29   lcd.print(distCM);
30   temp_distCM = distCM;
31   delay(300);
32 }

```

- บันทึกไฟล์โค้ด ชื่อ LAB12_1.ino
- ทำการ Compile โค้ด LAB12_1.ino
- นำไฟล์ LAB12_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	95

8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

10. พิมพ์โค้ดโปรแกรมแสดง * ตามค่าระยะทางที่อ่านได้

```

Lab12_2
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int triggerPin = 13;
4 int echoPin = 12;
5 long duration,distCM;
6 void setup()
7 {
8   lcd.begin();
9   pinMode(triggerPin,OUTPUT);
10  pinMode(echoPin,INPUT);
11 }
12 void loop()
13 {
14   digitalWrite(triggerPin,LOW);
15   delayMicroseconds(2);
16   digitalWrite(triggerPin,HIGH);
17   delayMicroseconds(10);
18   digitalWrite(triggerPin,LOW);
19   duration = pulseIn(echoPin,HIGH);
20   distCM = duration/58;
21   int x = constrain(distCM,4,19);
22   lcd.clear();
23   lcd.home();
24   for (int i=4;i<=x;i++)
25     lcd.print("*");
26   delay(150);
27 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	96

11. บันทึกไฟล์โค้ด ชื่อ LAB12_2.ino
12. ทำการ Compile โค้ด LAB12_2.ino
13. นำไฟล์ LAB12_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

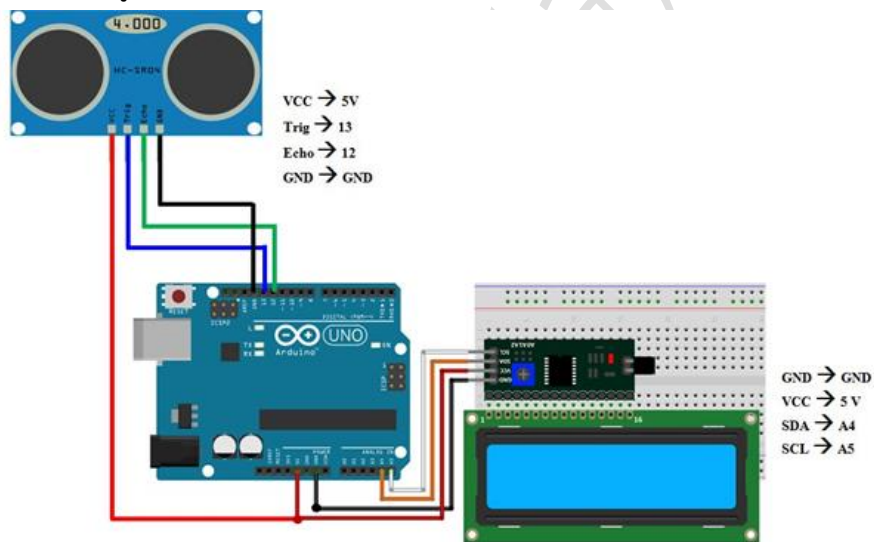
.....

.....

.....

.....

16. ประกอบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 12.2




รูปที่ 12.2 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04

17. Upload โปรแกรม LAB12_1.ino ลงบอร์ด Arduino UNO R3
18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	97

19. Upload โปรแกรม LAB12_2.ino ลงบอร์ด Arduino UNO R3

20. สังเกตและบันทึกผลการทดลอง

.....

.....

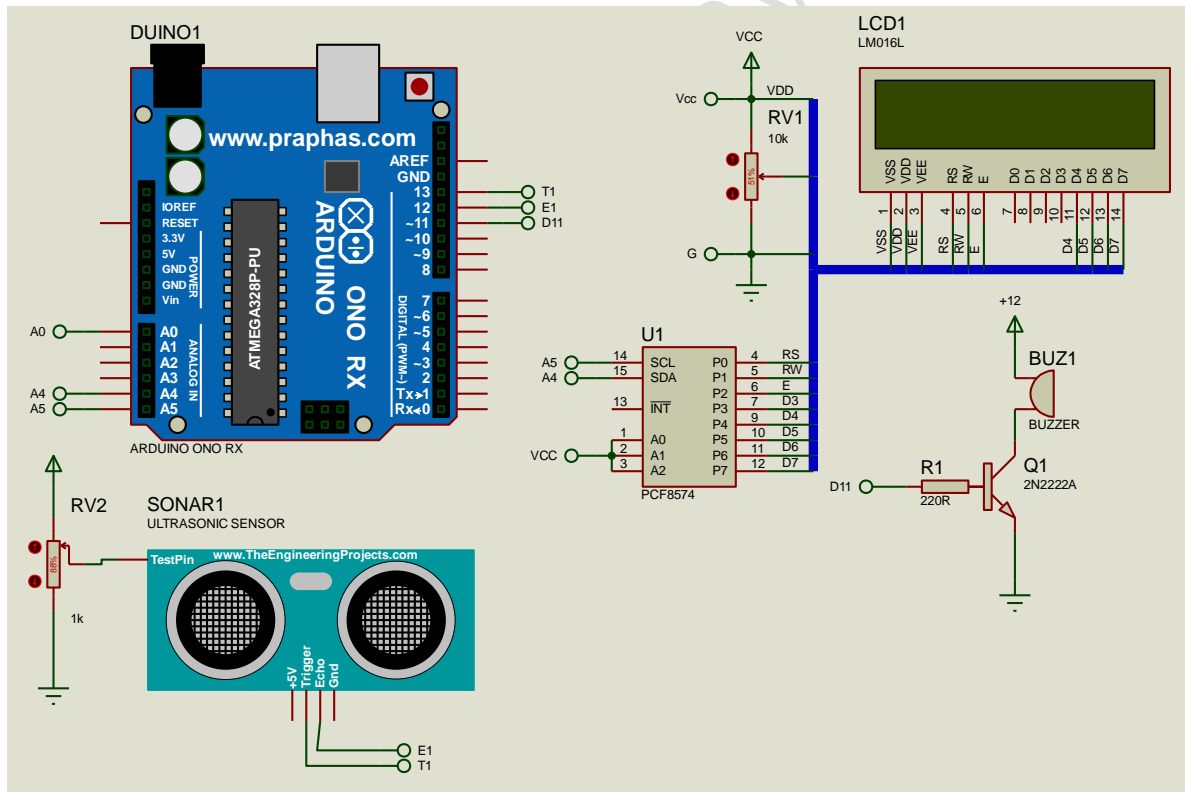
.....

.....

ตอนที่ 2 โปรแกรมการเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer


21. เปิดโปรแกรม Proteus 7 Professional

22. ออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer โดยบอร์ด Arduino UNO R3 แสดงดังรูปที่ 12.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 12.3 แสดงการออกแบบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer

23. บันทึกไฟล์ชื่อ LAB12-2


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	98

24. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมแจ้งเตือนด้วยเสียงเมื่อระยะทางน้อยกว่าค่าที่กำหนดไว้

```

Lab12_3§
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int triggerPin = 13;
4 int echoPin = 12;
5 int buzzerpin = 11;
6 long duration,distCM;
7 void setup()
8 {
9   lcd.begin();
10  pinMode(buzzerpin,OUTPUT);
11  pinMode(triggerPin,OUTPUT);
12  pinMode(echoPin,INPUT);
13  digitalWrite(buzzerpin,HIGH);
14 }
15 void loop()
16 {
17   digitalWrite(triggerPin,LOW);
18   delayMicroseconds(2);
19   digitalWrite(triggerPin,HIGH);
20   delayMicroseconds(10);
21   digitalWrite(triggerPin,LOW);
22   duration = pulseIn(echoPin,HIGH);
23   distCM = duration/58;
24   lcd.home();
25   if (distCM > 400)
26   {
27     lcd.clear();
28     lcd.print("***!!Alarm!!**");
29     tone(buzzerpin,700,500);
30     delay(500);
31     digitalWrite(buzzerpin,HIGH);
32     delay(70);
33     lcd.setCursor(0,1);
34     lcd.print(distCM);
35   }
36   else
37   {
38     lcd.print("<----->");
39     lcd.setCursor(0,1);
40     lcd.print(distCM);
41   }
42 }

```


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	99

25. บันทึกไฟล์โค้ด ชื่อ LAB12_3.ino
26. ทำการ Compile โค้ด LAB12_3.ino
27. นำไฟล์ LAB12_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
28. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
29. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


30. พิมพ์โค้ดโปรแกรมแจ้งเตือนด้วยเสียงที่มีช่วง delay เปลี่ยนแปลงตามระยะทาง

Lab12_4 §

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int triggerPin = 13;
4 int echoPin = 12;
5 int buzzerpin = 11;
6 long duration,distCM;
7 void setup()
8 {
9   lcd.begin();
10  pinMode(buzzerpin,OUTPUT);
11  pinMode(triggerPin,OUTPUT);
12  pinMode(echoPin,INPUT);
13  digitalWrite(buzzerpin,HIGH);
14 }
15 void loop()
16 {
17   digitalWrite(triggerPin,LOW);
18   delayMicroseconds(2);
19   digitalWrite(triggerPin,HIGH);
20   delayMicroseconds(10);
21   digitalWrite(triggerPin,LOW);
22   duration = pulseIn(echoPin,HIGH);
23   distCM = duration/58;

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12	
	ชื่อวิชา	ไมโครคอนโทรลเลอร์		
	รหัสวิชา	3105-2007	หน้าที่	
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	100	

```

24  lcd.home ();
25  if ((distCM>=4) && (distCM<=30))
26  {
27      lcd.clear ();
28      lcd.print ("####Alarm####");
29      int dl = distCM*20;
30      tone (buzzerpin, 600, dl);
31      delay (dl);
32      digitalWrite (buzzerpin, HIGH);
33      delay (70);
34  }
35  else
36  {
37      lcd.print ("  Dist > 30 cm  ");
38  }
39  }

```

31. บันทึกไฟล์โค้ด ชื่อ LAB12_4.ino
32. ทำการ Compile โค้ด LAB12_4.ino
33. นำไฟล์ LAB12_4.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
34. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
35. สังเกตและบันทึกผลการทดลอง

.....

.....

.....


.....

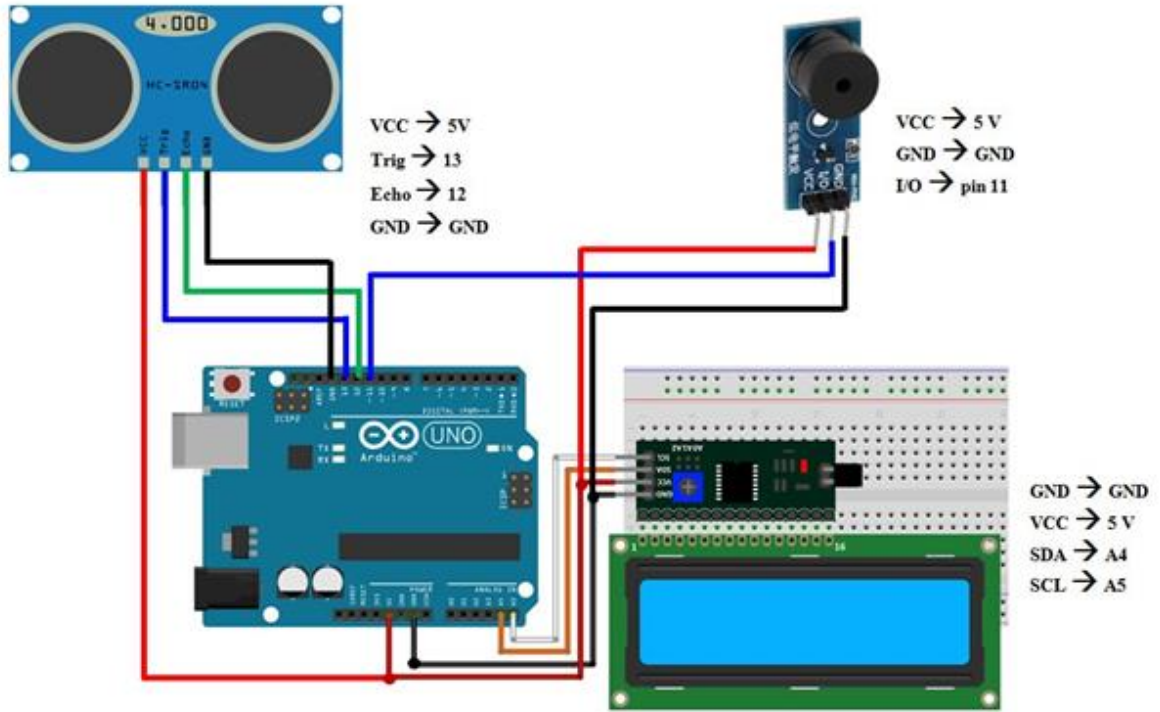
.....

.....

.....

36. ประกอบวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 12.4

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04	101



รูปที่ 12.4 แสดงวงจรเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04 และ Buzzer

37. Upload โปรแกรม LAB12_3.ino ลงบอร์ด Arduino UNO R3

38. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

39. Upload โปรแกรม LAB12_4.ino ลงบอร์ด Arduino UNO R3


40. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 12
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมเชื่อมต่อไมโครวัดระยะทางด้วยอัลตราโซนิก HC-SR04	102

41. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 13
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ PIR Sensor	103

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 13 เรื่องงานโปรแกรมการเชื่อมต่อกับ PIR Sensor ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ

- อธิบายหลักการทำงานของวงจรการเชื่อมต่อกับ PIR Sensor ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรการเชื่อมต่อกับ PIR Sensor ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการเชื่อมต่อกับ PIR Sensor ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมการเชื่อมต่อกับ PIR Sensor ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การเชื่อมต่อกับ PIR Sensor ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

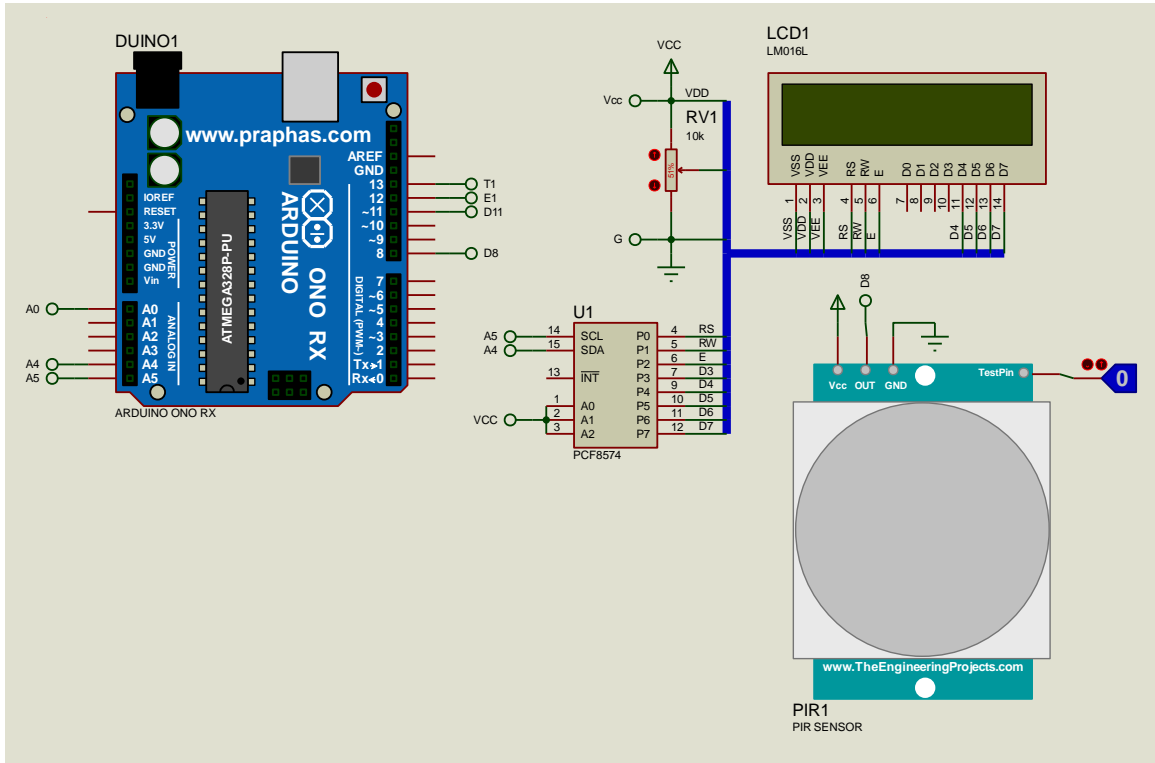
ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 13
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ PIR Sensor	104

ลำดับขั้นการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบการเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD โดยบอร์ด Arduino UNO R3 แสดงดังรูปที่ 13.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 13.1 แสดงการออกแบบวงจรเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD


3. บันทึกไฟล์ชื่อ LAB13-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมสถานะที่อ่านได้จาก PIR

Lab13_1

```

1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int PIRpin = 8;
4 void setup()
5 {
6   lcd.begin();

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 13
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ PIR Sensor	105

```

7   pinMode (PIRpin, INPUT);
8   lcd.home ();
9   lcd.print ("Waiting for PIR");
10  delay (10000);
11  lcd.clear ();
12  }
13 void loop ()
14 {
15   int x = digitalRead (PIRpin);
16   lcd.home ();
17   lcd.print ("PIR = ");
18   lcd.print (x);
19   delay (100);
20 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB13_1.ino
6. ทำการ Compile โค้ด LAB13_1.ino
7. นำไฟล์ LAB12_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

.....


.....

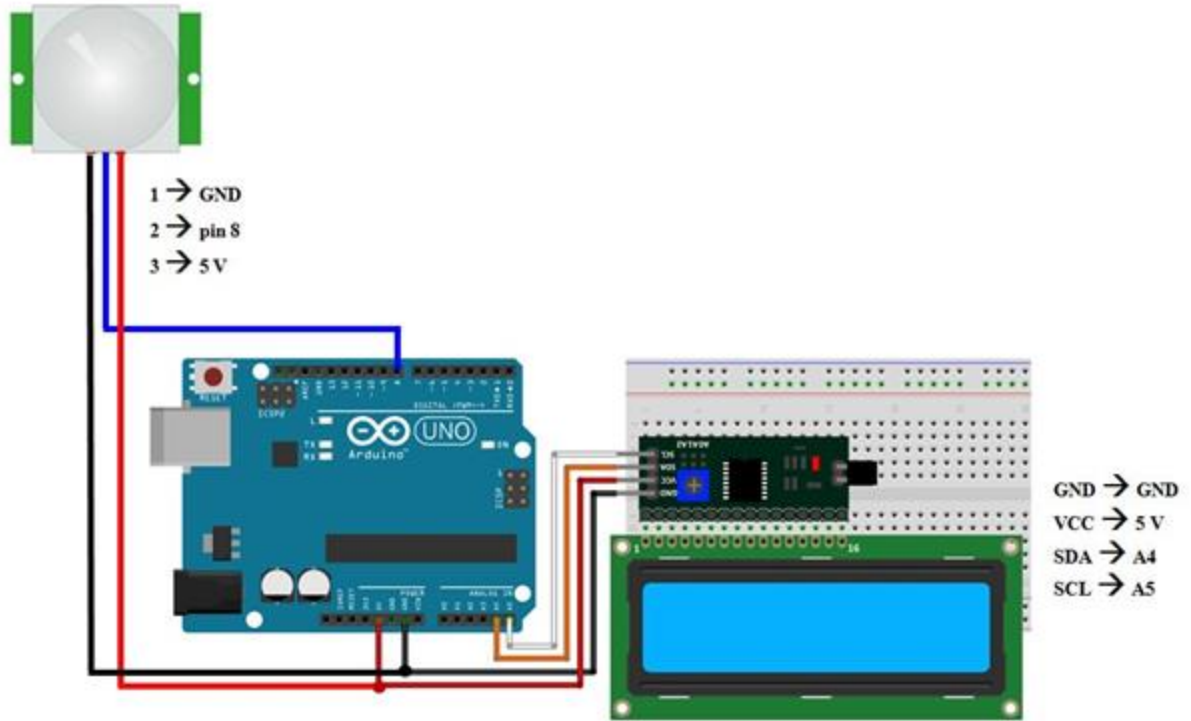
.....

.....

.....

10. ประกอบวงจรเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 13.2

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 13
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการเชื่อมต่อกับ PIR Sensor	106



รูปที่ 13.2 แสดงวงจรเชื่อมต่อกับ PIR Sensor แสดงผลด้วย LCD

- Upload โปรแกรม LAB13_1.ino ลงบอร์ด Arduino UNO R3
- สังเกตและบันทึกผลการทดลอง

.....

.....

.....

- สรุปผลการทดลอง

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	107

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 14 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20 ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานวงจรการอ่านค่าอุณหภูมิด้วย DS18B20 ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจรการอ่านค่าอุณหภูมิด้วย DS18B20 ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการอ่านค่าอุณหภูมิด้วย DS18B20 ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมการอ่านค่าอุณหภูมิด้วย DS18B20 ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การอ่านค่าอุณหภูมิด้วย DS18B20 ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

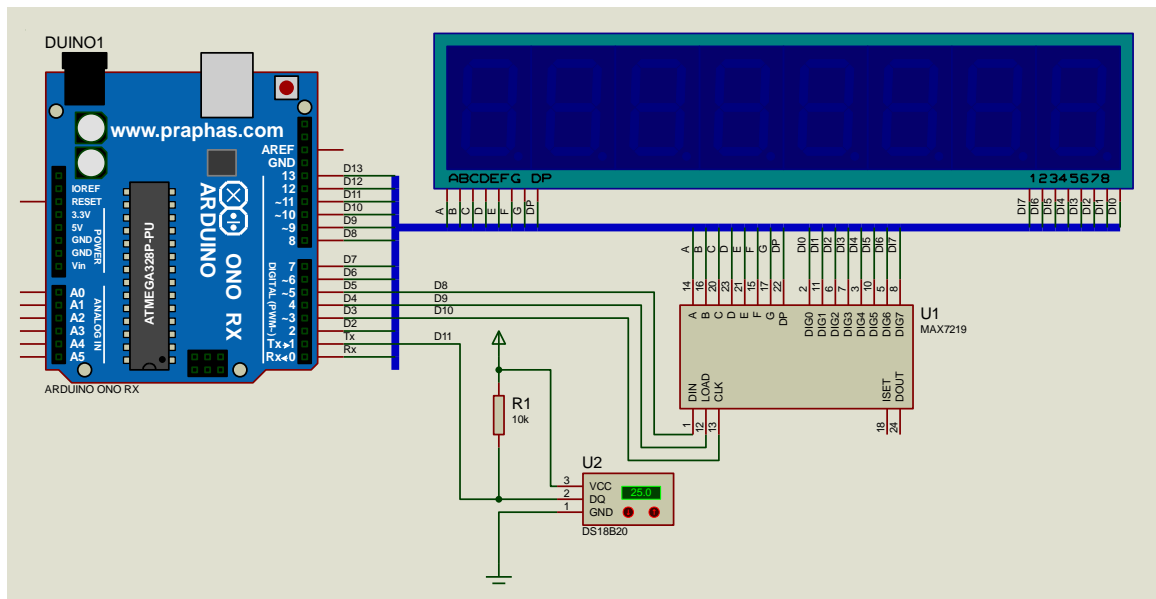
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทั้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	108

ลำดับขั้นการทดลอง

ตอนที่ 1 การอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LED 7-Segment

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 14.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า




รูปที่ 14.1 แสดงการออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LED 7-Segment

3. บันทึกไฟล์ชื่อ LAB14-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าจาก DS18B20 มาแสดงที่ LED 7-Segment

```

Lab14_1
1 #include <LedControl.h>
2 #include <OneWire.h>
3 #include <DallasTemperature.h>
4 #define ONE_WIRE_BUS 11
5 OneWire oneWire(ONE_WIRE_BUS);
6 DallasTemperature sensors(&oneWire);
7 float Tfloat;
8 long Tint;
9 LedControl lc=LedControl(8,10,9,1);

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	109

```

10 // Pin 8->DIN, 10->CLK, 9->CS (LOAD), 1 = No.of devices
11 void show6digit(int num)
12 {
13     int seg1, seg2;
14     seg2 = (((num%100000)%10000)%1000)%100/10;
15     seg1 = (((num%100000)%10000)%1000)%100%10;
16     lc.setDigit(0,0,seg1,false);
17     if (num>=10)
18         lc.setDigit(0,1,seg2,false);
19     delay(300);
20 }
21 void setup(void)
22 {
23     sensors.begin();
24     lc.shutdown(0,false);
25     lc.setIntensity(0,5);
26     lc.clearDisplay(0);
27 }
28 void loop(void)
29 {
30     sensors.requestTemperatures();
31     Tfloat = sensors.getTempCByIndex(0);
32     Tint = int(Tfloat);
33     lc.clearDisplay(0);
34     show6digit(Tint);
35 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB14_1.ino
6. ทำการ Compile โค้ด LAB14_1.ino
7. นำไฟล์ LAB14_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

.....

.....

.....


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	110

10. พิมพ์โค้ดโปรแกรมรับค่าจาก DS18B20 (มีทศนิยม) มาแสดงที่ LED 7-Segment

```

Lab14_2 §
1 #include <LedControl.h>
2 #include <OneWire.h>
3 #include <DallasTemperature.h>
4 #define ONE_WIRE_BUS 11
5 OneWire oneWire(ONE_WIRE_BUS);
6 DallasTemperature sensors(&oneWire);
7 float Tfloat;
8 long Tint;
9 LedControl lc=LedControl(8,10,9,1);
10 // Pin 8->DIN, 10->CLK, 9->CS(LOAD), 1 = No.of devices
11 void show6digit(long num)
12 {
13   int seg1,seg2,seg3,seg4,seg5,seg6;
14   seg6 = (num/100000);
15   seg5 = ((num%100000)/10000);
16   seg4 = (((num%100000)%10000)/1000);
17   seg3 = (((num%100000)%10000)%1000)/100;
18   seg2 = (((((num%100000)%10000)%1000)%100)/10);
19   seg1 = ((((((num%100000)%10000)%1000)%100)%10);
20   lc.setDigit(0,0,seg1,false);
21   lc.setDigit(0,1,seg2,false);
22   lc.setDigit(0,2,seg3,false);
23   lc.setDigit(0,3,seg4,false);
24   lc.setDigit(0,4,seg5,true);
25   if (num>=100000)
26     lc.setDigit(0,5,seg6,false);
27   delay(300);
28 }
29 void setup(void)
30 {
31   sensors.begin();
32   lc.shutdown(0,false);
33   lc.setIntensity(0,5);
34   lc.clearDisplay(0);
35 }
36 void loop(void)
37 {
38   sensors.requestTemperatures();
39   Tfloat = sensors.getTempCByIndex(0);
40   Tint = long(Tfloat*10000);
41   lc.clearDisplay(0);
42   show6digit(Tint);
43 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	111

11. บันทึกไฟล์โค้ด ชื่อ LAB14_2.ino
12. ทำการ Compile โค้ด LAB14_2.ino
13. นำไฟล์ LAB14_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
14. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
15. สังเกตและบันทึกผลการทดลอง

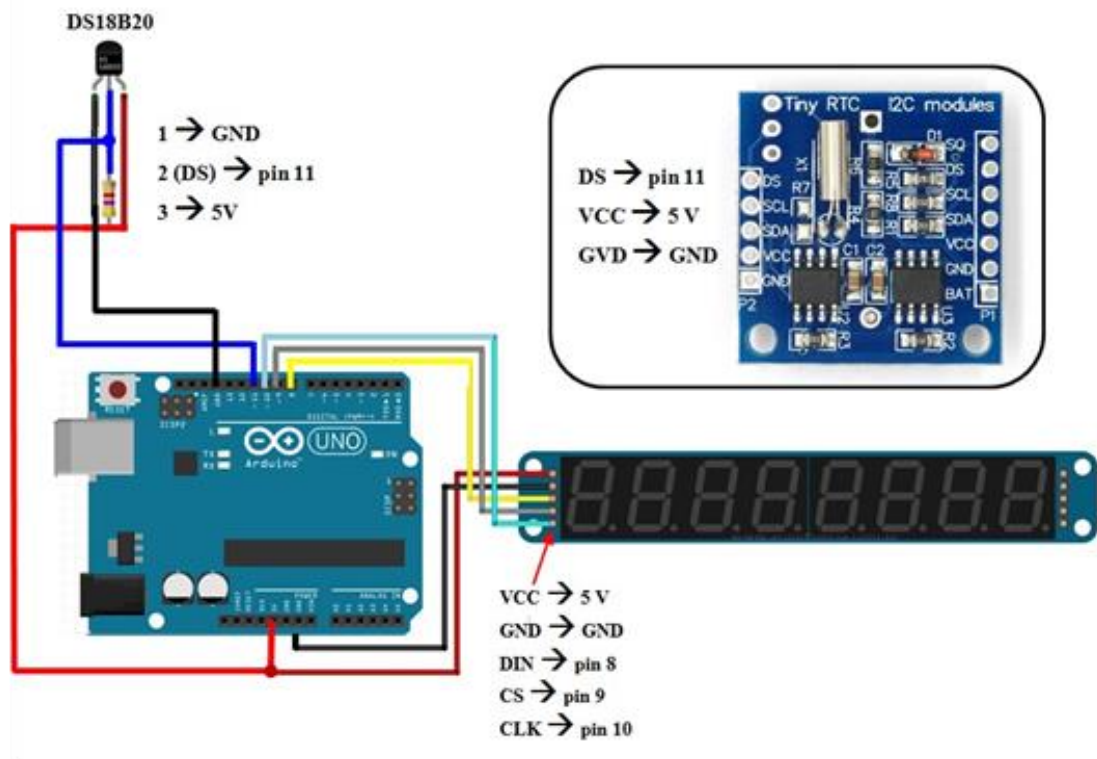
.....

.....

.....


.....

16. ประกอบรับค่าจาก DS18B20 มาแสดงที่ LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 14.2



รูปที่ 14.2 แสดงรับค่าจาก DS18B20 มาแสดงที่ LED 7-Segment

17. Upload โปรแกรม LAB14_1.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	112

18. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

19. Upload โปรแกรม LAB14_2.ino ลงบอร์ด Arduino UNO R3

20. สังเกตและบันทึกผลการทดลอง

.....

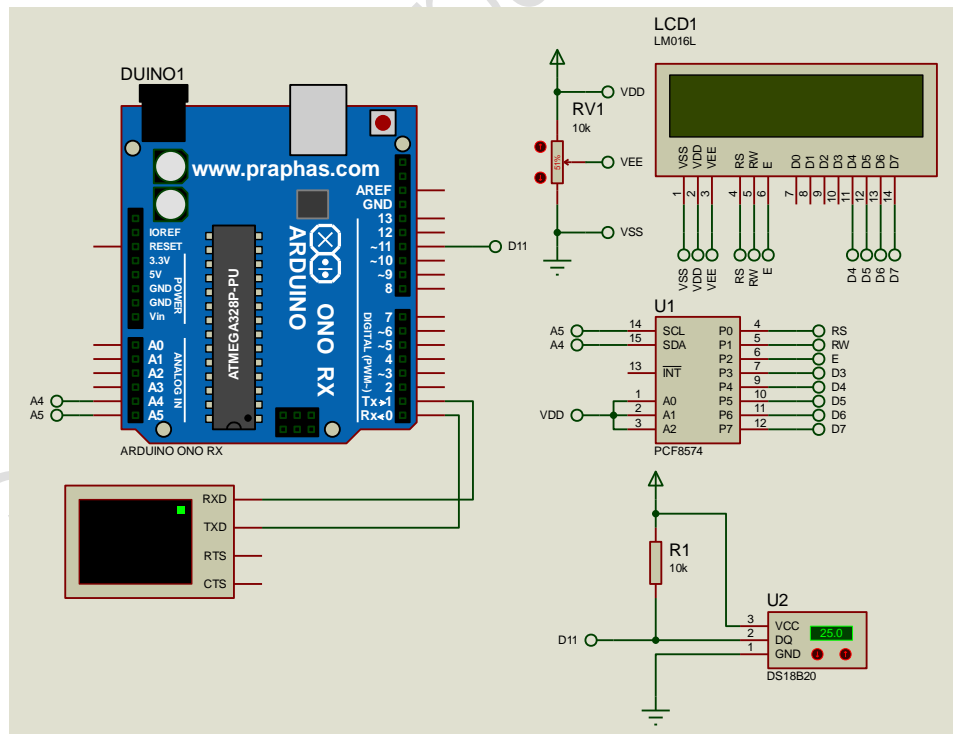
.....

.....


.....

ตอนที่ 2 การอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LCD

21. ออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 14.3 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 14.3 แสดงการออกแบบวงจรอ่านค่าอุณหภูมิจาก DS18B20 มาแสดงผลที่ LCD

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	113

22. บันทึกไฟล์ชื่อ LAB14-2
23. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าจาก DS18B20 มาแสดงที่ LCD

```

Lab14_3
1 #include <LiquidCrystal_I2C.h>
2 #include <OneWire.h>
3 #include <DallasTemperature.h>
4 #define ONE_WIRE_BUS 11
5 OneWire oneWire(ONE_WIRE_BUS);
6 DallasTemperature sensors(&oneWire);
7 float Tfloat;
8 LiquidCrystal_I2C lcd(0x27,16,2);
9 void setup(void)
10 {
11   sensors.begin();
12   lcd.begin();
13   Serial.begin(9600);
14 }
15 void loop(void)
16 {
17   sensors.requestTemperatures();
18   Tfloat = sensors.getTempCByIndex(0);
19   Serial.print(Tfloat);
20   lcd.setCursor(0,0);
21   lcd.print("T = ");
22   lcd.print(Tfloat,4);
23   delay(250);
24 }

```


24. บันทึกไฟล์โค้ด ชื่อ LAB14_3.ino
25. ทำการ Compile โค้ด LAB14_3.ino
26. นำไฟล์ LAB14_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
27. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
28. สังเกตและบันทึกผลการทดลอง

.....

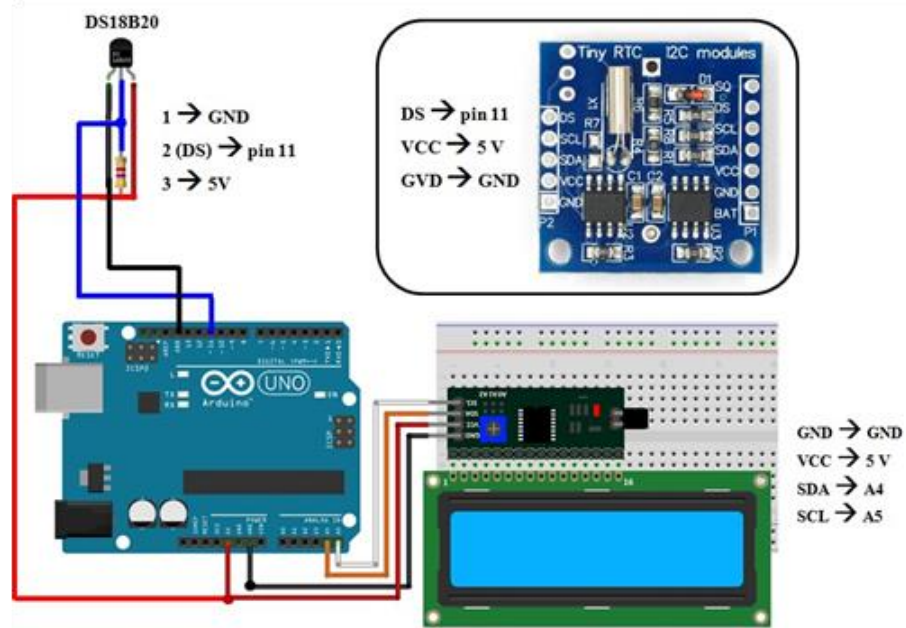
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 14
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20	114

29. ประกอบรับค่าจาก DS18B20 มาแสดงที่ LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 14.4



รูปที่ 14.4 แสดงรับค่าจาก DS18B20 มาแสดงที่ LCD

30. Upload โปรแกรม LAB14_3.ino ลงบอร์ด Arduino UNO R3

31. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

32. สรุปผลการทดลอง


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	115

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 15 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานของวงจรการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|--|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 8 Professional | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

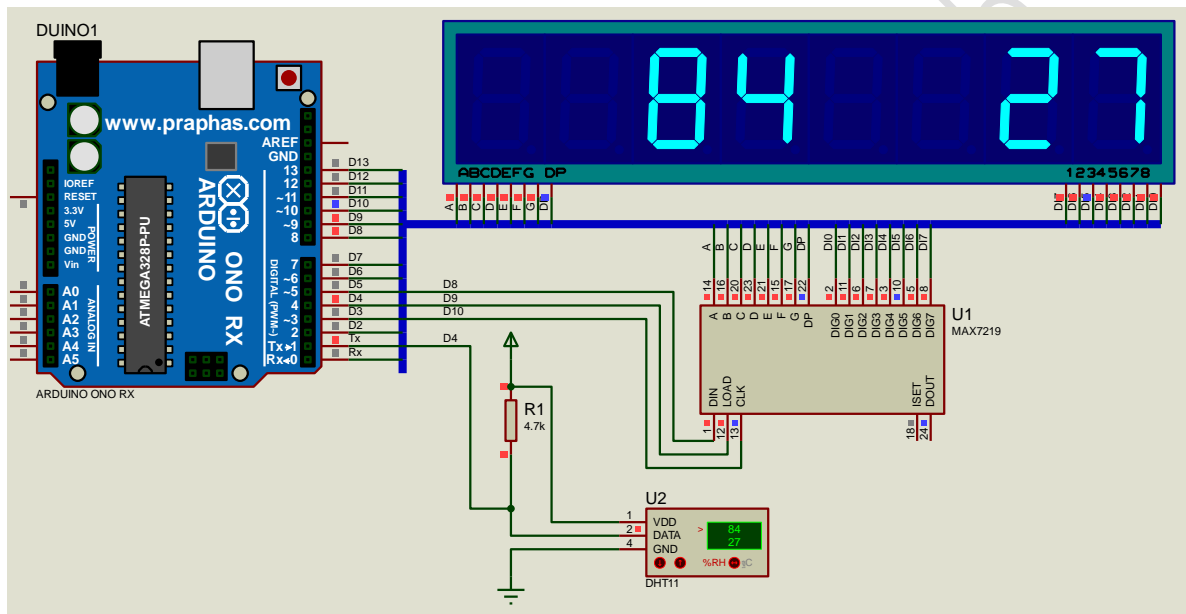
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิดการลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	116

ลำดับขั้นการทดลอง

ตอนที่ 1 การอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7-Segment

1. เปิดโปรแกรม Proteus 8 Professional
2. ออกแบบการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7-Segment โดยใช้อบอร์ด Arduino UNO R3 แสดงดังรูปที่ 15.1 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 15.1 แสดงการออกแบบวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7-Segment


3. บันทึกไฟล์ชื่อ LAB15-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าจาก DHT11 มาแสดงที่ LED 7-Segment

Lab15_1

```

1 #include <LedControl.h>
2 #include <DHT11.h>
3 int pin=4;
4 DHT11 dht11(pin);
5 LedControl lc=LedControl(8,10,9,1);
6 // Pin 8->DIN, 10->CLK, 9->CS(Load), 1 = No.of devices


```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	117

```

7 void show2digit(int h,int t)
8 {
9   int seg1,seg2,seg3,seg4;
10  seg1 = h%10;
11  seg2 = h/10;
12  lc.setDigit(0,4,seg1,false);
13  lc.setDigit(0,5,seg2,false);
14  seg3 = t%10;
15  seg4 = t/10;
16  lc.setDigit(0,0,seg3,false);
17  lc.setDigit(0,1,seg4,false);
18  delay(300);
19 }
20 void setup()
21 {
22   Serial.begin(9600);
23   while (!Serial) {
24     ; // wait for serial port to connect. Needed for Leonardo only
25   }
26   lc.shutdown(0,false);
27   lc.setIntensity(0,5);
28   lc.clearDisplay(0);
29 }
30 void loop()
31 {
32   int err;
33   float temp, humi;
34   if((err=dht11.read(humi, temp))==0)
35   {
36     Serial.print("temperature:");
37     Serial.print(temp);
38     Serial.print(" humidity:");
39     Serial.print(humi);
40     Serial.println();
41   }
42 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	118

```

43   else
44   {
45       Serial.println();
46       Serial.print("Error No :");
47       Serial.print(err);
48       Serial.println();
49   }
50   lc.clearDisplay(0);
51   show2digit(humi, temp);
52 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB15_1.ino
6. ทำการ Compile โค้ด LAB15_1.ino
7. นำไฟล์ LAB15_1.ino.HEX มาใช้กับโปรแกรม Proteus 8 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....


.....

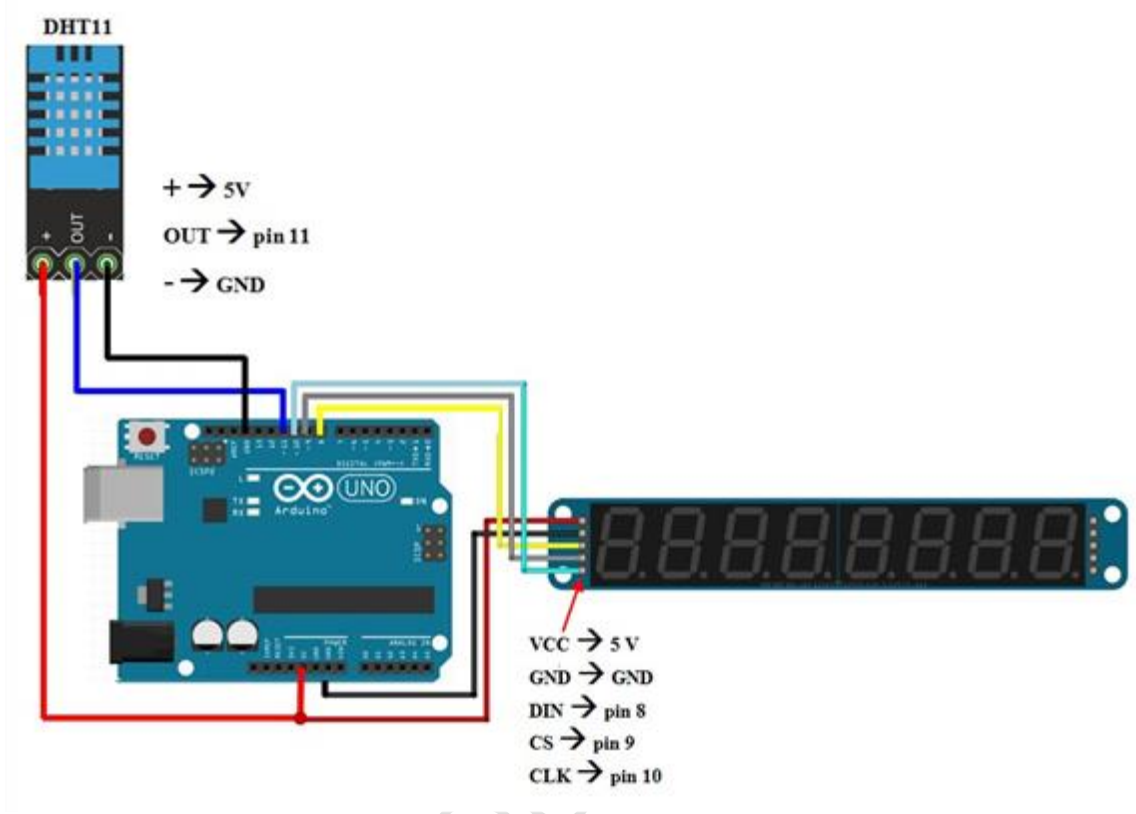
.....

.....

.....

10. ประกอบวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7-Segment โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 15.2

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	119



รูปที่ 15.2 แสดงวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LED 7-Segment

11. Upload โปรแกรม LAB15_1.ino ลงบอร์ด Arduino UNO R3
12. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....

.....


.....

.....

.....

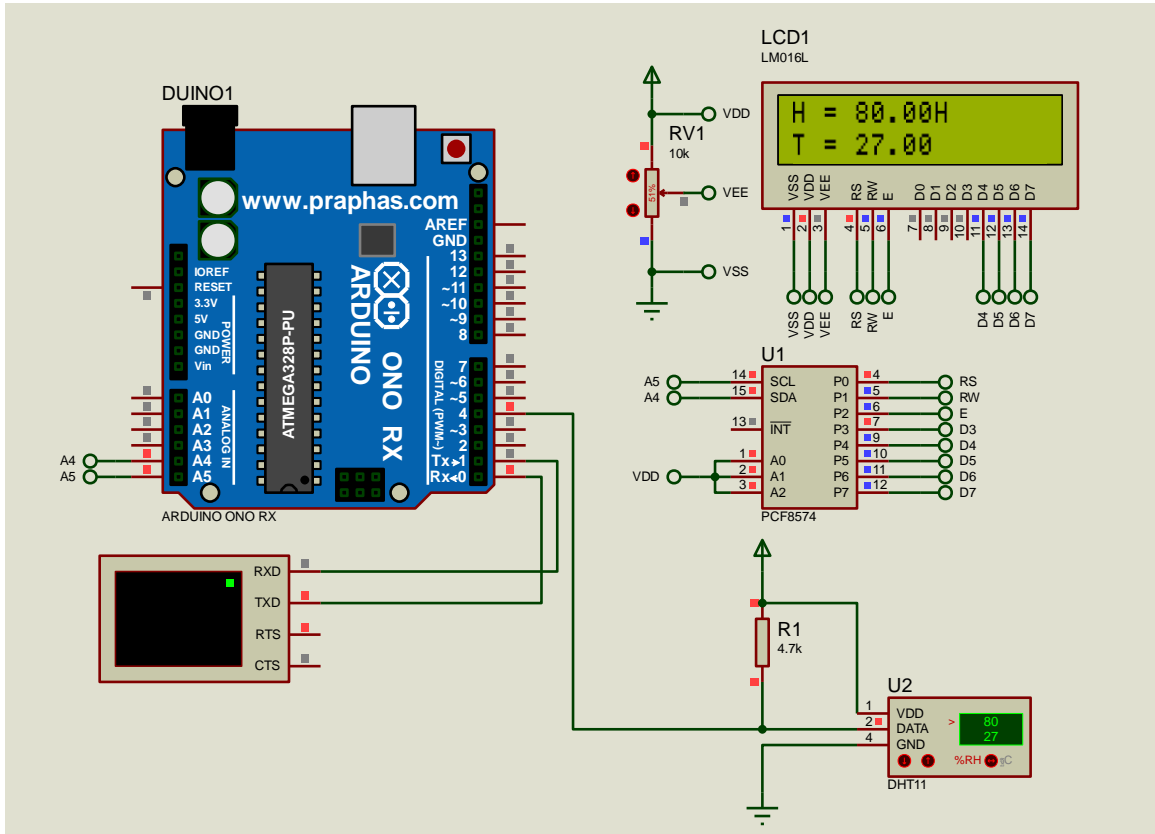
.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	


ตอนที่ 2 การอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LCD

13. ออกแบบวงจรอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LCD โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 15.3 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 15.3 แสดงการออกแบบวงจรรับค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงผลที่ LCD

14. บันทึกไฟล์ชื่อ LAB15-2
15. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงที่ LCD

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	121

Lab15_2§

```

1 #include <LiquidCrystal_I2C.h>
2 #include "DHT.h"
3 DHT dht;
4 int dhtDataPin = 11;
5 LiquidCrystal_I2C lcd(0x27,16,2);
6 void setup()
7 {
8   dht.setup(dhtDataPin);
9   lcd.begin();
10  lcd.setCursor(0,0);lcd.print("H =   %RH");
11  lcd.setCursor(0,1);lcd.print("T =   C");
12 }
13 void loop()
14 {
15   delay(dht.getMinimumSamplingPeriod());
16   int humidity = dht.getHumidity();
17   int temperature = dht.getTemperature();
18   lcd.setCursor(4,0);
19   lcd.print(humidity);
20   lcd.setCursor(4,1);
21   lcd.print(temperature);
22   delay(300);
23 }

```


16. บันทึกไฟล์โค้ด ชื่อ LAB15_2.ino
17. ทำการ Compile โค้ด LAB15_2.ino
18. นำไฟล์ LAB15_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
19. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
20. สังเกตและบันทึกผลการทดลอง

.....

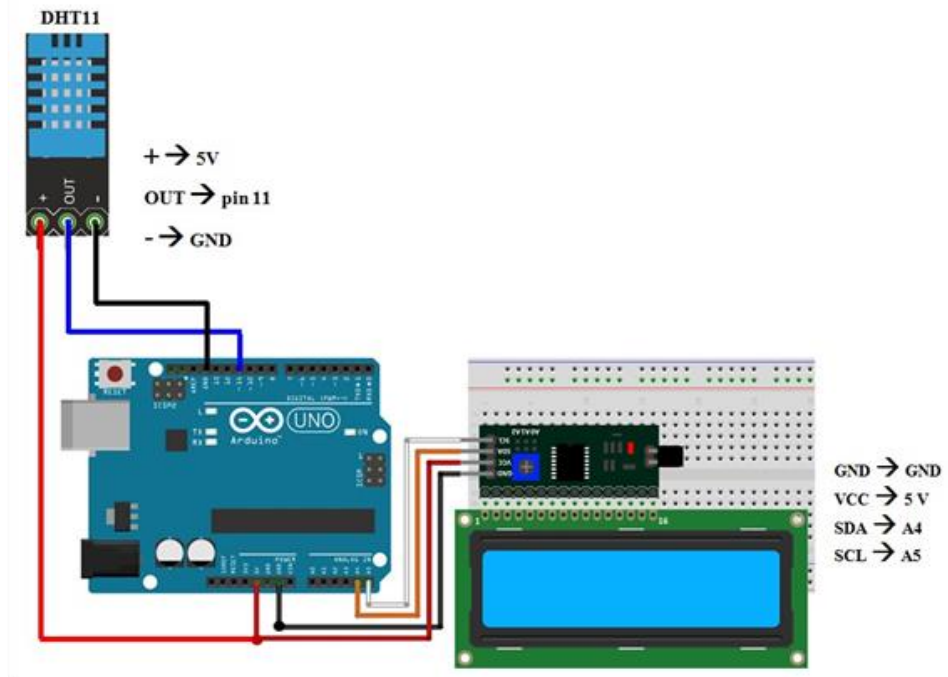
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 15
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11	122

21. ประกอบวงจรรับค่าอุณหภูมิและค่าความชื้นจาก DHT11 มาแสดงที่ LCD โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 15.4



รูปที่ 15.4 แสดงวงจรรับค่าอุณหภูมิและค่าความชื้นจาก DHT11มาแสดงที่ LCD

22. Upload โปรแกรม LAB15_2.ino ลงบอร์ด Arduino UNO R3

23. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

.....


24. สรุปผลการทดลอง

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	123

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 16 เรื่องงานการเขียนโปรแกรมควบคุมมอเตอร์ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทำงานของวงจรโปรแกรมควบคุมมอเตอร์แบบต่าง ๆ ได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจรควบคุมมอเตอร์แบบต่าง ๆ ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรควบคุมมอเตอร์แบบต่าง ๆ ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมมอเตอร์แบบต่าง ๆ ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การควบคุมมอเตอร์แบบต่าง ๆ ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|--|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 8 Professional | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

ข้อห้ามและข้อควรระวัง

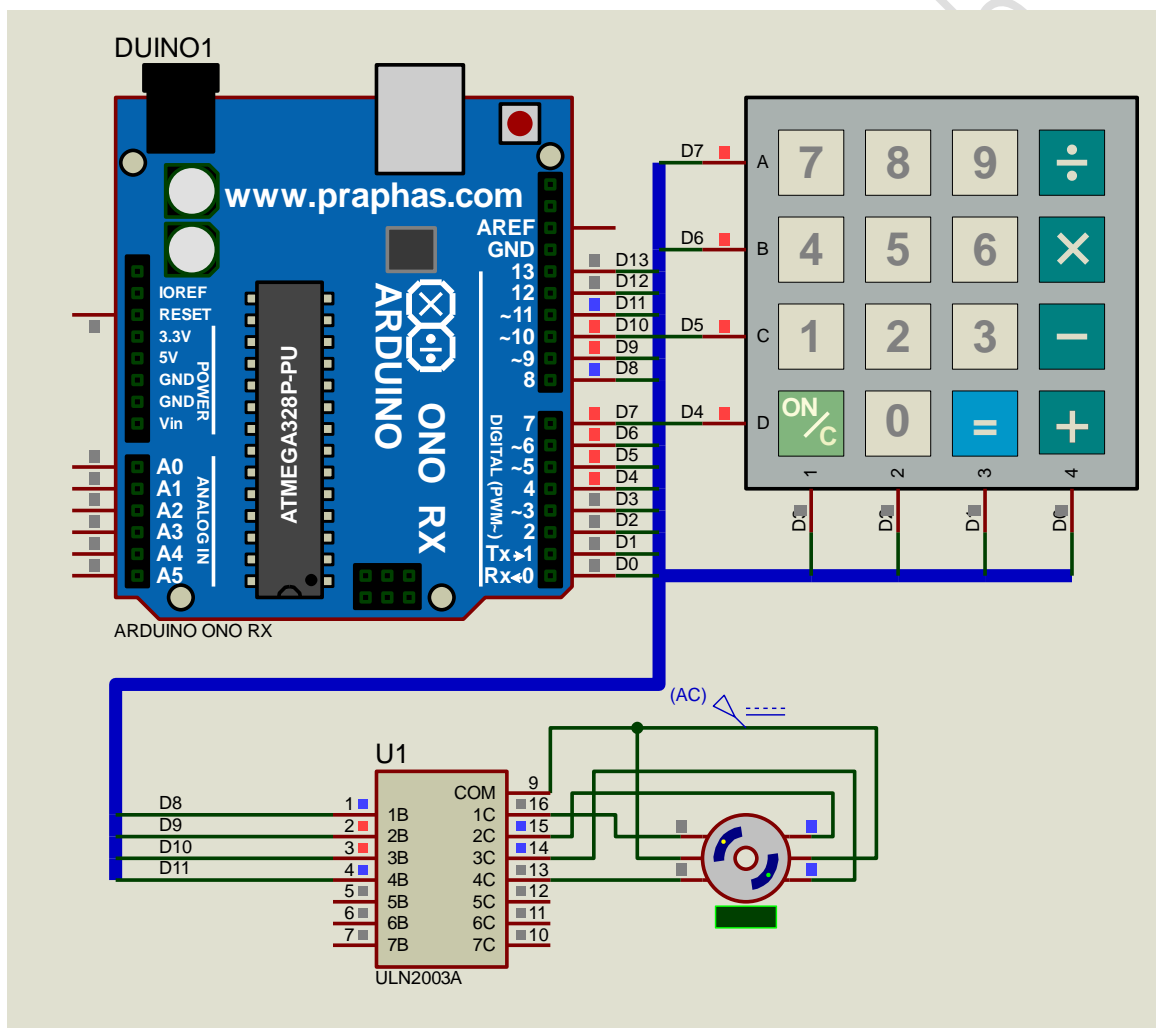
- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซิลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั่ว ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	124

ลำดับขั้นการทดลอง


ตอนที่ 1 การควบคุม Stepper Motor ด้วย Keypad

1. เปิดโปรแกรม Proteus 8 Professional
2. ออกแบบวงจรการควบคุม Stepper Motor ด้วย Keypad โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 16.1 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 16.1 แสดงการออกแบบวงจรการควบคุม Stepper Motor ด้วย Keypad

3. บันทึกไฟล์ชื่อ LAB16-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าจาก DHT11 มาแสดงที่ LED 7-Segment

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	125


Lab16_1 §

```

1 #include <Stepper.h>
2 #include "Keypad.h"
3 #define STEP_ANGLE_4STEP 32 //360/11.25 degree
4 #define STEP_OUT_WITH_GEAR 2048 //32*64
5 Stepper stepper(STEP_ANGLE_4STEP,8,10,9,11);
6 char keys[4][4]={
7   {'7','8','9','A'},
8   {'4','5','6','B'},
9   {'1','2','3','C'},
10  {'*','0','#','D'}};
11 byte rowPins[] = {7,6,5,4};
12 byte colPins[] = {3,2,1,0};
13 int speedmotor = 400;
14 int dirmotor = 1;
15 Keypad keypad = Keypad(makeKeymap(keys),rowPins,colPins,4,4);
16 void setup()
17 {
18 }
19 void loop()
20 {
21   char key = keypad.getKey();
22   if (key != NO_KEY)
23   {
24     if (key == '1')
25       speedmotor = 400;
26     if (key == '2')
27       speedmotor = 700;
28     if (key == '3')
29       speedmotor = 1000;
30     if (key == 'A')
31       dirmotor = 1;
32     if (key == 'B')
33       dirmotor = -1;
34     if (key == 'C')
35       dirmotor = 0;
36   }
37   stepper.setSpeed(speedmotor);
38   stepper.step(dirmotor);
39 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB16_1.ino
6. ทำการ Compile โค้ด LAB16_1.ino
7. นำไฟล์ LAB16_1.ino.HEX มาใช้กับโปรแกรม Proteus 8 Professional

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	126

8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

9. สังเกตและบันทึกผลการทดลอง

.....

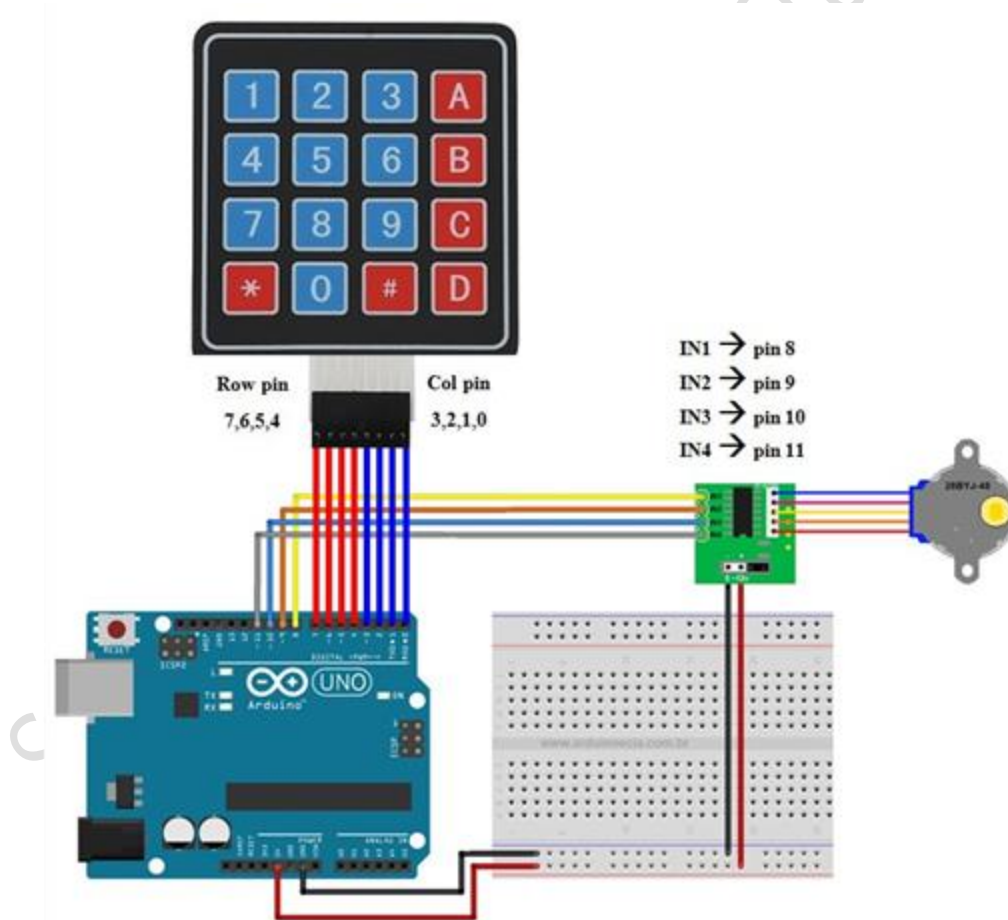
.....

.....


.....

.....

10. ประกอบวงจรการควบคุม Stepper Motor ด้วย Keypad โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 16.2



รูปที่ 16.2 แสดงวงจรการควบคุม Stepper Motor ด้วย Keypad

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	127

11. Upload โปรแกรม LAB16_1.ino ลงบอร์ด Arduino UNO R3

12. สังเกตและบันทึกผลการทดลอง

.....

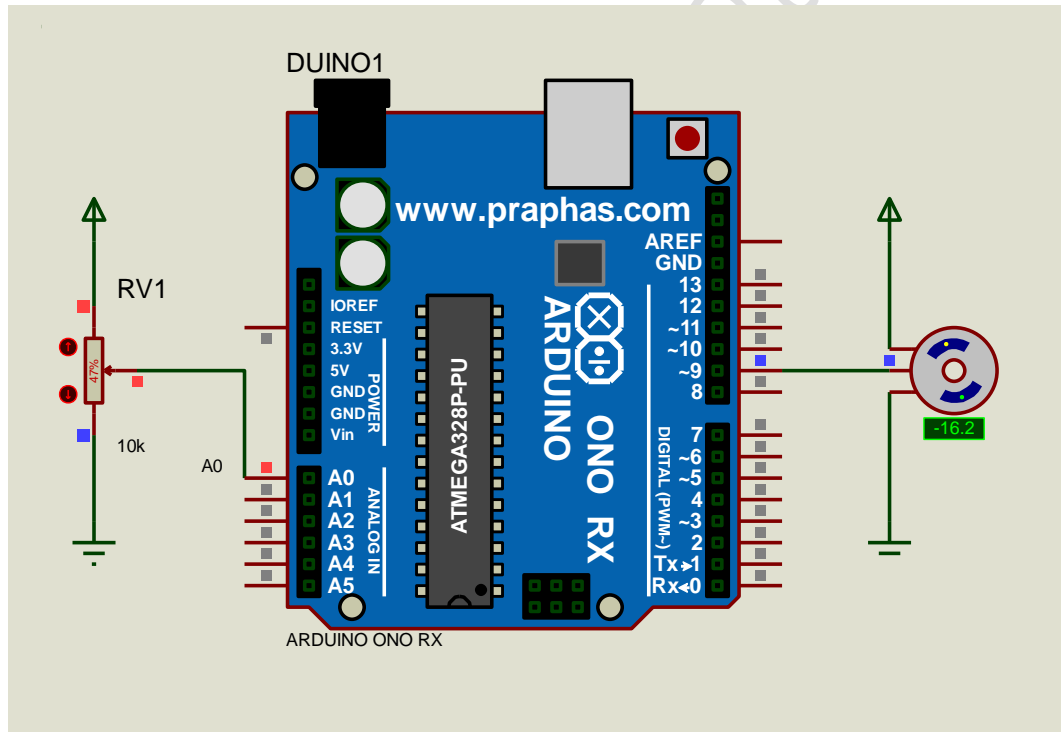
.....

.....

.....

ตอนที่ 2 การควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้


13. ออกแบบวงจรการควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้ โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 16.3 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 16.3 แสดงการออกแบบวงจรควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้

14. บันทึกไฟล์ชื่อ LAB16-2

15. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมรับค่าจากตัวต้านทานปรับค่าได้มาควบคุม Servo Motor

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	128

Lab16_2§

```

1 #include <Servo.h>
2 Servo servo;
3 int x,y;
4 void setup()
5 {
6   servo.attach(9);
7 }
8 void loop()
9 {
10  x = analogRead(A0);
11  y = map(x, 0, 1023, 0, 179);
12  servo.write(y);
13  delay(50);
14 }

```

16. บันทึกไฟล์โค้ด ชื่อ LAB16_2.ino
17. ทำการ Compile โค้ด LAB16_2.ino
18. นำไฟล์ LAB16_2.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
19. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
20. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

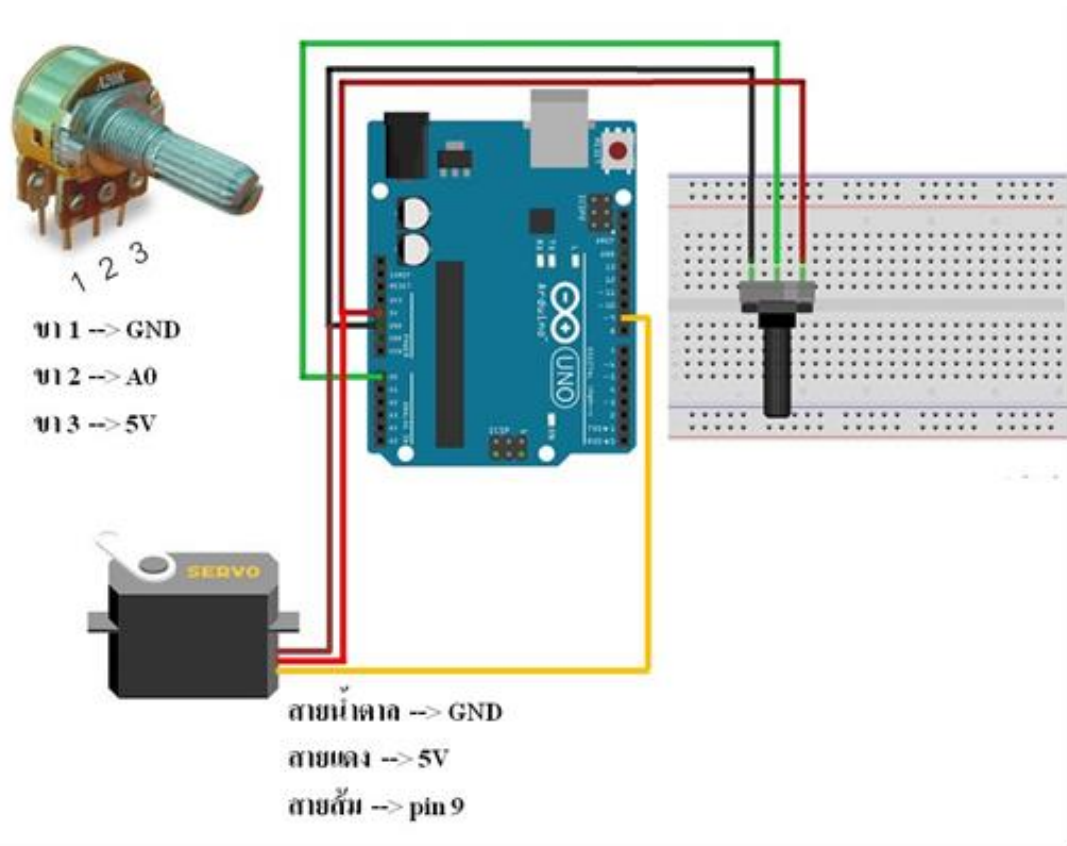
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	129

21. ประกอบวงจรควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้ โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 16.4



รูปที่ 16.4 แสดงวงจรควบคุม Servo Motor ด้วยตัวต้านทานปรับค่าได้

22. Upload โปรแกรม LAB16_2.ino ลงบอร์ด Arduino UNO R3

23. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

.....

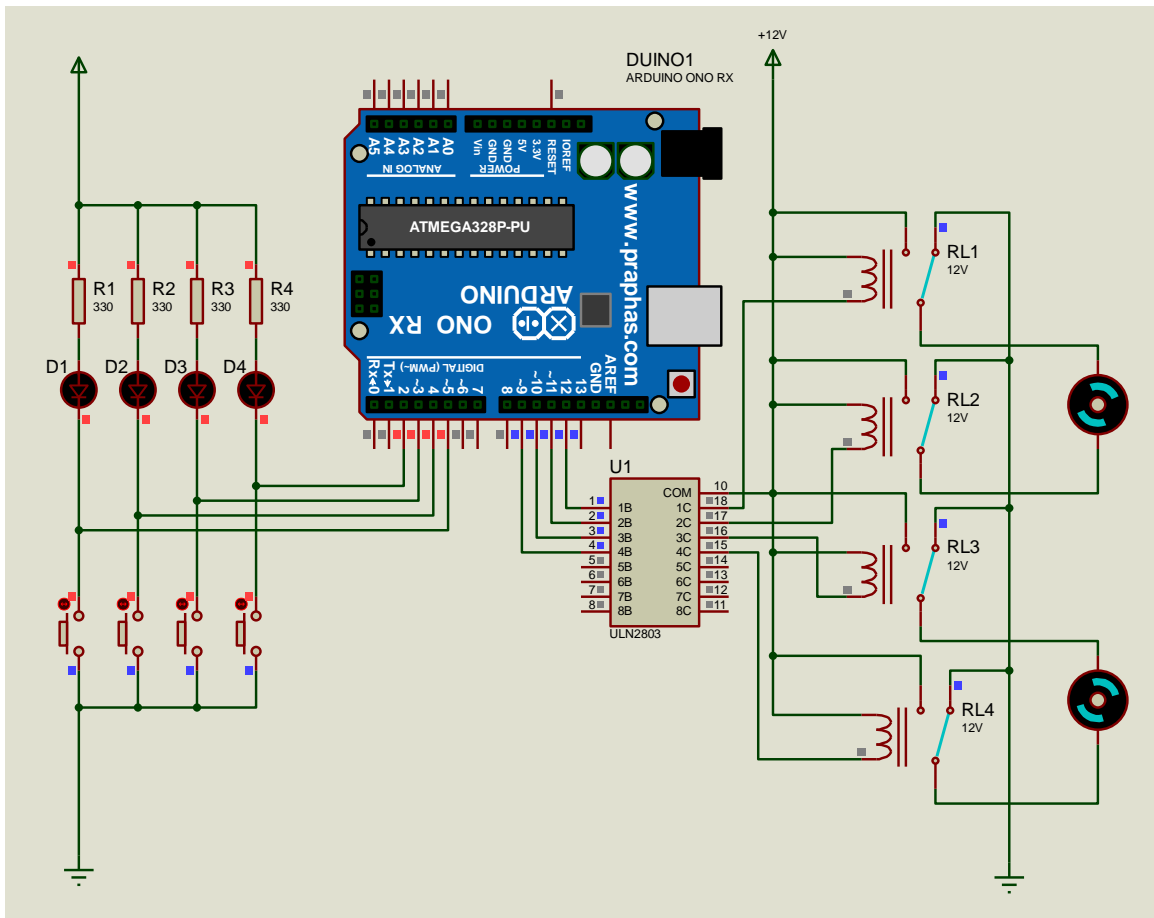
.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	130


ตอนที่ 3 การควบคุม DC Motor ด้วยการกด Pushbutton

24. ออกแบบวงจรควบคุม DC Motor ด้วยการกด Pushbutton โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 16.5 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 16.5 แสดงการออกแบบวงจรควบคุม DC Motor ด้วยการกด Pushbutton

25. บันทึกไฟล์ชื่อ LAB16-3
26. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมควบคุม DC Motor ด้วยการกด Pushbutton
27. บันทึกไฟล์โค้ด ชื่อ LAB16_3.ino
28. ทำการ Compile โค้ด LAB16_3.ino
29. นำไฟล์ LAB16_3.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 16
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานการเขียนโปรแกรมควบคุมมอเตอร์	131

30. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน

31. สังเกตและบันทึกผลการทดลอง

.....

.....

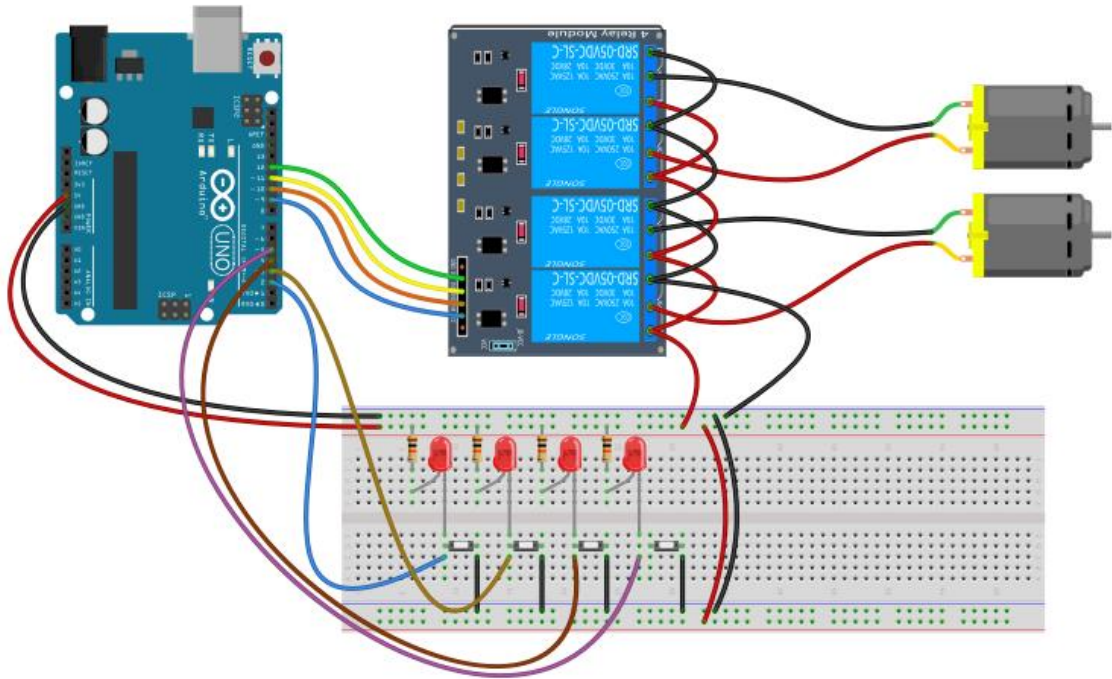
.....

.....

.....


.....

32. ประกอบวงจรควบคุม DC Motor ด้วยการกด Pushbutton โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 16.5



รูปที่ 16.6 แสดงวงจรควบคุม DC Motor ด้วยการกด Pushbutton

33. Upload โปรแกรม LAB16_3.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	133

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 17 เรื่องงานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ ตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ


- อธิบายหลักการทํางานวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ได้อย่างถูกต้อง
- ออกแบบและจำลองการทํางานวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ได้อย่างถูกต้อง
- เขียนโปรแกรมควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การเชื่อมต่อควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์ได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

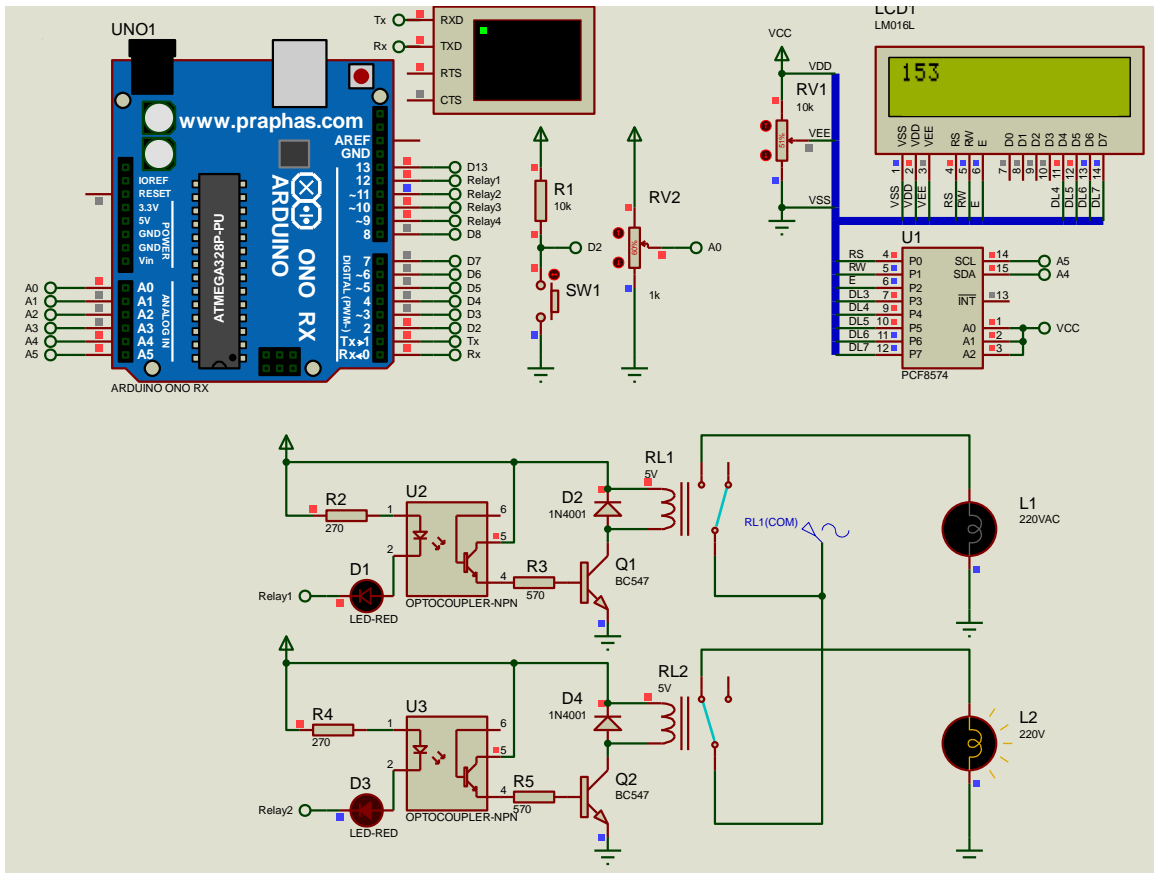
ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายโหนด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	134


ลำดับขั้นตอนการทดลอง

1. เปิดโปรแกรม Proteus 7 Professional
2. ออกแบบวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์โดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 17.1 ด้วยโปรแกรม Proteus 7 Professional หรือดีกว่า



รูปที่ 17.1 แสดงการออกแบบวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์


3. บันทึกไฟล์ชื่อ LAB17-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	135

```

Lab17_1.ino
1 #include <LiquidCrystal_I2C.h>
2 LiquidCrystal_I2C lcd(0x27,16,2);
3 int aVal=0, tempaVal=0;
4 const int SW_ON = 2;
5 const int led_ON = 13; int CSW_ON = 0; int C_CSW_ON =0;
6 int Relay1=12,Relay2=11,Relay3=10,Relay4=9;
7 int sensorPin = A0;
8 int sensorValue = 0;
9 void setup() {
10   lcd.backlight();
11   lcd.begin();
12   Serial.begin(9600);
13   pinMode(led_ON, OUTPUT);pinMode(SW_ON, INPUT);
14   pinMode(Relay1,OUTPUT);pinMode(Relay2,OUTPUT);
15   pinMode(Relay3,OUTPUT);pinMode(Relay4,OUTPUT);
16 //
17   digitalWrite(Relay1,HIGH);digitalWrite(Relay2,HIGH);
18   digitalWrite(Relay3,HIGH);digitalWrite(Relay4,HIGH);
19   lcd.setCursor(0,0);
20   lcd.print("Project2017");
21   lcd.setCursor(0,1);
22   lcd.print("standby");
23 }
24 //END Setup
25 void loop() {
26   CSW_ON = digitalRead(SW_ON);
27   sensorValue = analogRead(sensorPin);//Serial.println(sensorValue);
28   //=====
29   int c_sensorValue1 = map(sensorValue,0,1023,0,255);
30   //Serial.print("c_sensorValue1 = ");Serial.println(c_sensorValue1);
31   int c_sensorValue2 = map(sensorValue,0,1023,255,0);
32   //Serial.print("c_sensorValue2 = ");Serial.println(c_sensorValue2);
33   //=====
34   //check SW_ON
35   if (CSW_ON == LOW) { delay(1000);
36     lcd.clear();
37     C_CSW_ON++;
38     Serial.print("C_CSW_ON = ");Serial.println(C_CSW_ON);
39     //OPEN Project
40     digitalWrite(led_ON, HIGH);
41   //END check SW_ON
42   if(C_CSW_ON == 1){
43     lcd.setCursor(0,0);
44     lcd.print("Power On ");
45     delay(1000);
46     lcd.clear();
47     lcd.print("Power On ");
48     delay(1000);


```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	136

```

49     lcd.clear();
50     }
51 }
52 if(C_CSW_ON == 1){ //OPEN PROJECT
53     digitalWrite(led_ON, HIGH);
54     //delay(1000);
55     digitalWrite(Relay1, HIGH);
56     //ตรวจสอบค่า VR=====
57     //lcd.clear();
58     lcd.setCursor(0,0);
59     lcd.print(c_sensorValue1);
60     lcd.print(" ");
61     //=====
62     if(c_sensorValue1 >= 150){
63         digitalWrite(Relay2, LOW);
64     }else{
65         digitalWrite(Relay2, HIGH);
66     }
67     //=====
68     //RelaySTART();
69 }else{//OFF PROJECT
70     lcd.setCursor(0,0);
71     //lcd.print("Power Off");
72     lcd.setCursor(0,1);
73     digitalWrite(Relay2, HIGH);
74     }
75     //=====
76     //RelaySTART();
77 }else{//OFF PROJECT
78     lcd.setCursor(0,0);
79     //lcd.print("Power Off");
80     lcd.setCursor(0,1);
81     lcd.print("standby");
82     digitalWrite(led_ON, LOW); C_CSW_ON = 0;
83     RelaySTOP();
84 }
85 } //END Loop
86 //=====โปรแกรมย่อย=====
87 void RelaySTOP() {
88     digitalWrite(Relay1, HIGH); digitalWrite(Relay2, HIGH);
89     digitalWrite(Relay3, HIGH); digitalWrite(Relay4, HIGH);
90 }
91 void RelaySTART() {
92     digitalWrite(Relay1, LOW); digitalWrite(Relay2, LOW);
93     digitalWrite(Relay3, LOW); digitalWrite(Relay4, LOW);
94 }

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	137

5. บันทึกไฟล์โค้ด ชื่อ LAB17_1.ino
6. ทำการ Compile โค้ด LAB17_1.ino
7. นำไฟล์ LAB17_1.ino.HEX มาใช้กับโปรแกรม Proteus 7 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....

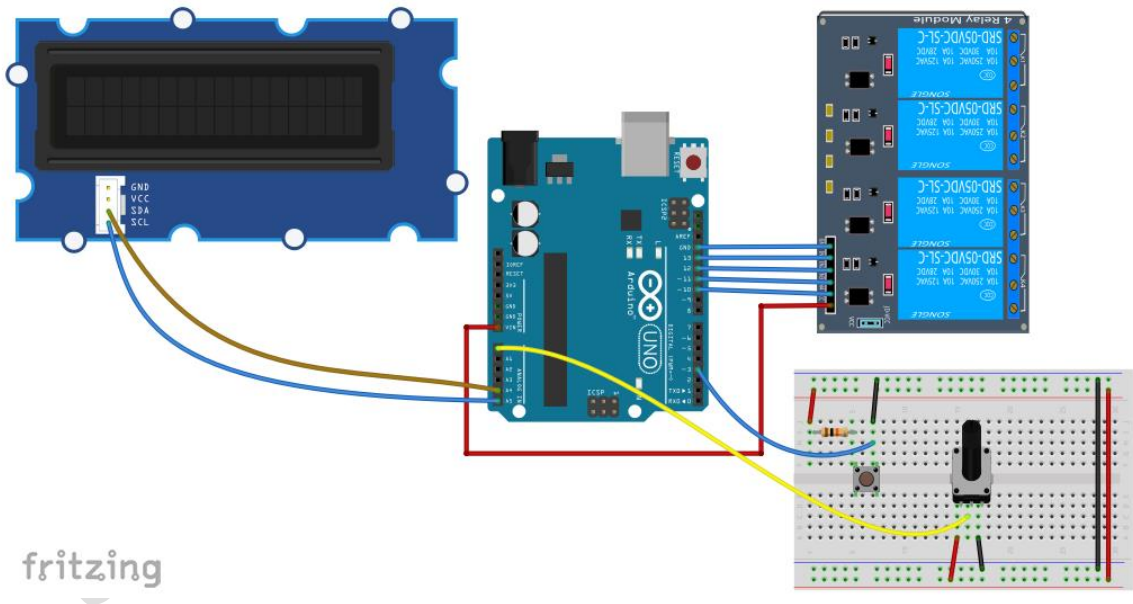
.....

.....

.....


.....

10. ประกอบวงจรวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์โดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 17.2



รูปที่ 17.2 แสดงวงจรควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์

11. Upload โปรแกรม LAB17_1.ino ลงบอร์ด Arduino UNO R3

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 17
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์	138

12. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

13. สรุปผลการทดลอง

.....


.....

.....

.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	139

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 18 เรื่องงานโปรแกรม Smart Farm system เบื้องต้นตามขั้นตอนการปฏิบัติงาน

วัตถุประสงค์ เพื่อให้นักศึกษาสามารถ

- อธิบายหลักการทำงานวงจร Smart Farm system เบื้องต้นได้อย่างถูกต้อง
- ออกแบบและจำลองการทำงานวงจร Smart Farm system เบื้องต้น ด้วยโปรแกรม Proteus ได้อย่างถูกต้อง
- ประกอบและติดตั้งวงจร Smart Farm system เบื้องต้นได้อย่างถูกต้อง
- เขียนโปรแกรม Smart Farm system เบื้องต้นได้อย่างถูกต้อง
- ทดสอบและบำรุงรักษา อุปกรณ์ การเชื่อมต่อวงจร Smart Farm system เบื้องต้นได้อย่างถูกต้อง

เครื่องมือและอุปกรณ์

- | | | |
|---|---|---------|
| 1. โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า | 1 | โปรแกรม |
| 2. โปรแกรม Proteus 7 Professional หรือสูงกว่า | 1 | โปรแกรม |
| 3. สาย USB สำหรับ Arduino Uno R3 | 1 | เส้น |
| 4. ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร | 1 | ชุด |
| 5. เครื่องคอมพิวเตอร์แบบพกพา | 1 | เครื่อง |
| 6. แผงต่อวงจร | 1 | ตัว |

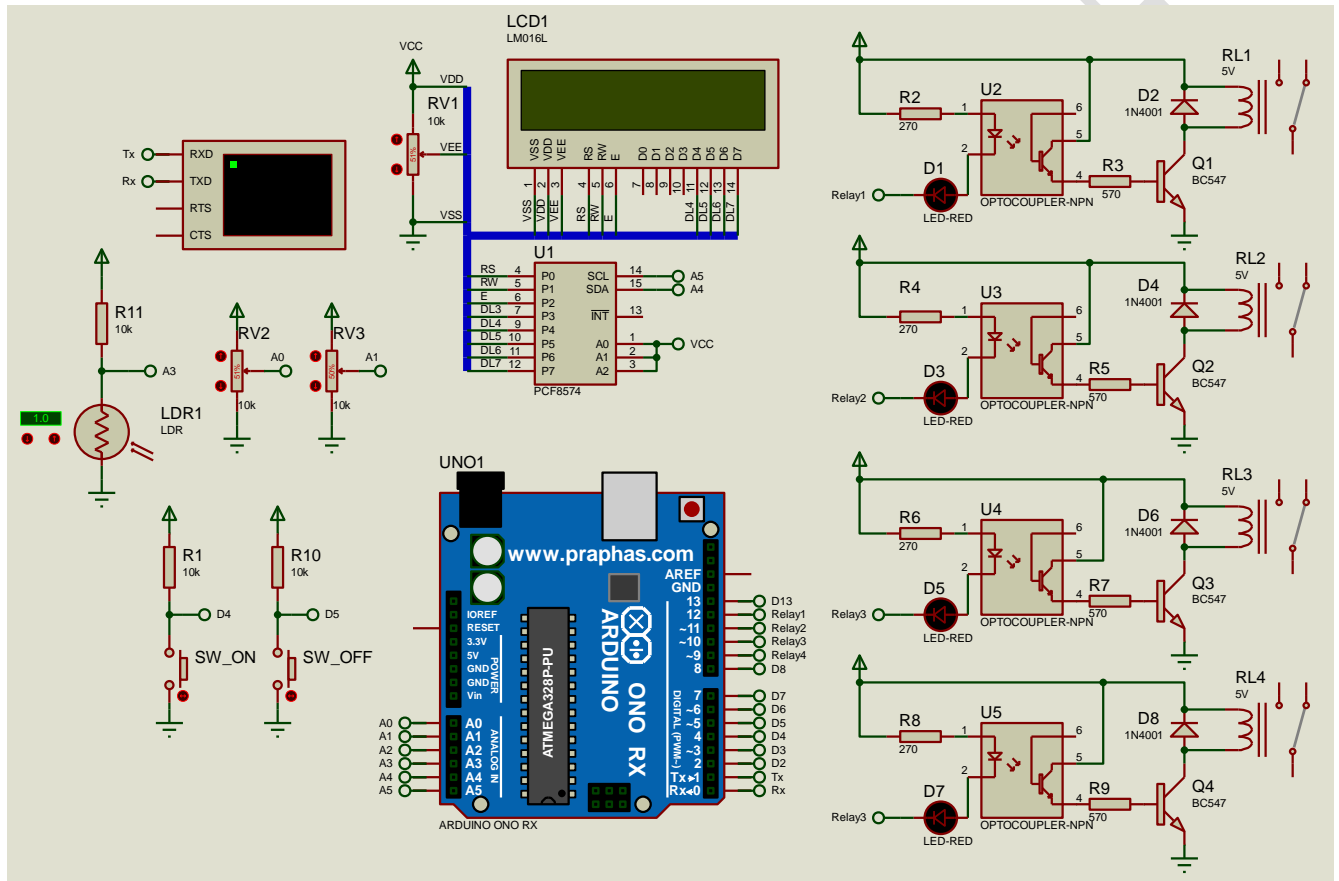
ข้อห้ามและข้อควรระวัง

- ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
- ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือซีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะเพราะอาจเกิด การลัดวงจรของภาคจ่ายไฟได้
- ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ที่งั้ว ควรถอดสายต่อวงจรออกให้หมด เพราะผลการทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
- ไม่ควรถอดสายสายไหลต USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
- ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	


ลำดับขั้นตอนการทดลอง

1. เปิดโปรแกรม Proteus 8 Professional
2. ออกแบบวงจรวงจร Smart Farm system เบื้องต้นโดยใช้บอร์ด Arduino UNO R3 แสดงดังรูปที่ 18.1 ด้วยโปรแกรม Proteus 8 Professional



รูปที่ 18.1 แสดงการออกแบบวงจรวงจร Smart Farm system เบื้องต้น

3. บันทึกไฟล์ชื่อ LAB18-1
4. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมวงจร Smart Farm system เบื้องต้น


	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	141

Lab18_1

```

1 #include <SoftwareSerial.h>
2 #include <DHT11.h>
3 //=====
4 #include <Wire.h>
5 #include <LiquidCrystal_I2C.h>
6 //LiquidCrystal_I2C lcd(0x27, 16, 2);
7 LiquidCrystal_I2C lcd(0x3F, 16, 2);
8 SoftwareSerial ArduinoSerial(3, 2); // RX, TX
9 //=====
10 int SW_ON = 4, SW_OFF = 5, pin=6; //กำหนดขา INPUT
11 int Sun = 9, CSun = 0, Relay1 = 12, Relay2 = 11; //กำหนดขา OUTPUT
12 int ATL = A0, ATH = A1; //กำหนดขา Analog
13 int VTL1 = 0, VTH1 = 0, TL = 0, TH = 0, VRHL = 50, VRHH = 80;
14 DHT11 dht11(pin);
15 //=====
16 void setup()
17 {
18   lcd.begin();
19   Serial.begin(115200);
20   ArduinoSerial.begin(4800);
21   while (!Serial){
22     ;
23   }
24   //=====
25   lcd.setCursor(0, 0); lcd.print("Elec Pattayatech");
26   lcd.setCursor(0, 1); lcd.print("Smart Farm 2016 ");
27   delay(5000);
28   //=====
29   pinMode(Relay1, OUTPUT); pinMode(Relay2, OUTPUT);
30   pinMode(SW_ON, INPUT); pinMode(SW_OFF, INPUT); pinMode(Sun, INPUT);
31   digitalWrite(Relay1, HIGH); digitalWrite(Relay2, HIGH);
32   lcd.setCursor(0, 0); lcd.print("          ");
33   lcd.setCursor(0, 1); lcd.print("          ");
34 }
35 //=====
36 void loop()
37 {
38   while (ArduinoSerial.available() > 0) {
39     float val = ArduinoSerial.parseFloat();
40     if (ArduinoSerial.read() == '\n') {
41       if (val == 10) {
42         digitalWrite(13, 1);
43       } else if (val == 11) {
44         digitalWrite(13, 0);
45       }
46       Serial.println(val);
47     }
48   }


```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	142

```

49 delay(100);
50 //=====ปรับค่าอุณหภูมิห้อง=====
51 CSun = digitalRead(Sun);
52 int err;
53 float temp, humi;
54 if((err=dht11.read(humi, temp))==0)
55 {
56 VTL1 = analogRead(ATL);
57 VTH1 = analogRead(ATH);
58 TL = map(VTL1, 0, 1023, 10, 30);
59 TH = map(VTH1, 0, 1023, 25, 45);
60 Serial.print("TL :");Serial.print(TL);Serial.println("°C");
61 Serial.print("TH :");Serial.print(TH);Serial.println("°C");
62 //=====
63 int Stemp =(int) (temp);int Shumi=(int) (humi); //ตัดทศนิยม
64 Serial.print("temp :");Serial.print(Stemp);Serial.println("°C");
65 Serial.print("humidity :");Serial.print(Shumi);Serial.println("%");
66 Serial.print("SUN :");Serial.println(CSun); //Serial.println();
67 //=====
68 lcd.setCursor(0, 0);lcd.print("TL:");lcd.print(TL);
69 lcd.setCursor(6, 0);lcd.print("TH:");lcd.print(TH);
70 lcd.setCursor(12, 0);lcd.print("Sl:");lcd.print(!CSun);
71 lcd.setCursor(0, 1);lcd.print("TN:");lcd.print(Stemp);
72 lcd.setCursor(6, 1);lcd.print("RH:");lcd.print(Shumi);
73 lcd.setCursor(12, 1);lcd.print("Pl:0");lcd.print(Shumi);
74 delay(1000);
75 //=====
76 //=====ควบคุมอุณหภูมิ=====
77 if(temp < TL && temp < TH ){
78     digitalWrite(Relay1, HIGH); //มีมีหน้าหยุดทำงาน
79     lcd.setCursor(12, 1);lcd.print("Pl:0");
80     delay(100);
81     digitalWrite(Relay2, LOW); //หลอดไฟทำงาน
82     lcd.setCursor(12, 0);lcd.print("Ll:1");
83     delay(60000); //หน่วงเวลา 1 นาที
84 }else if(temp > TL && temp > TH){
85     digitalWrite(Relay2, HIGH); //หลอดไฟไม่ทำงาน
86     lcd.setCursor(12, 0);lcd.print("Ll:0");
87     delay(100);
88     digitalWrite(Relay1, LOW); //มีมีหน้าทำงาน
89     lcd.setCursor(12, 1);lcd.print("Pl:1");
90     delay(60000); //หน่วงเวลา 1 นาที
91 }else if(temp > TL && temp < TH){
92     //=====ควบคุมความชื้น=====
93     if(humi < VRHL && humi < VRHH){
94         digitalWrite(Relay2, HIGH); //หลอดไฟไม่ทำงาน
95         lcd.setCursor(12, 0);lcd.print("Ll:0");
96         delay(100);

```

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	143

```

97     digitalWrite(Relay1, LOW); //ปิดหน้าทำงาน
98     lcd.setCursor(12, 1); lcd.print("P1:1");
99     delay(60000); //หน่วงเวลา 1 นาที
100    }else if(humi > VRHL && humi > VRHH){
101        digitalWrite(Relay1, HIGH); //ปิดหน้าหยุดทำงาน
102        lcd.setCursor(12, 1); lcd.print("P1:0");
103        delay(100);
104        digitalWrite(Relay2, LOW); //หลอดไฟทำงาน
105        lcd.setCursor(12, 0); lcd.print("L1:1");
106        delay(60000); //หน่วงเวลา 1 นาที
107    }else if(humi > VRHL && humi < VRHH){
108        //delay(1000);
109        digitalWrite(Relay1, HIGH); //ปิดหน้าหยุดทำงาน
110        lcd.setCursor(12, 1); lcd.print("P1:0");
111        digitalWrite(Relay2, HIGH); //หลอดไฟไม่ทำงาน
112        lcd.setCursor(12, 0); lcd.print("L1:0");
113        //หน่วงเวลา 1 นาที
114    }
115    //=====ENDควบคุมความชื้น=====
116    }
117    //=====ENDควบคุมอุณหภูมิ=====
118    }
119    else
120    {
121        Serial.println(); Serial.print("Error No :");
122        Serial.print(err); Serial.println();
123    }
124    delay(DHT11_RETRY_DELAY);
125 }

```

5. บันทึกไฟล์โค้ด ชื่อ LAB18_1.ino
6. ทำการ Compile โค้ด LAB18_1.ino
7. นำไฟล์ LAB18_1.ino.HEX มาใช้กับโปรแกรม Proteus 8 Professional
8. จากนั้นให้กดปุ่ม Play เพื่อเริ่มต้นการจำลองการทำงาน
9. สังเกตและบันทึกผลการทดลอง

.....


.....

.....

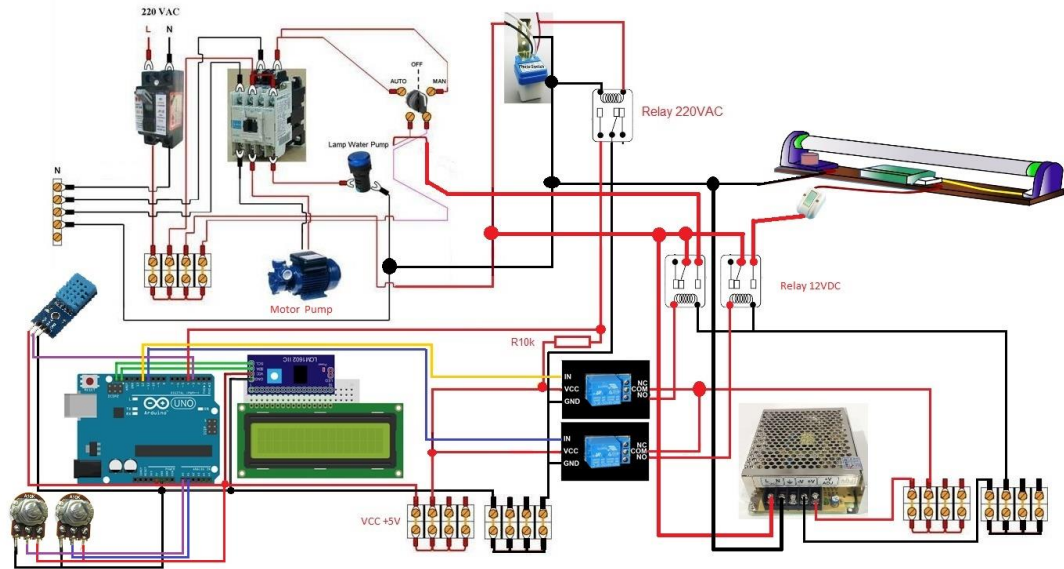
.....

.....

.....

	สาขาวิชา	อิเล็กทรอนิกส์	ใบงานการทดลองที่ 18
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	3105-2007	หน้าที่
	ชื่องาน	งานโปรแกรม Smart Farm system เบื้องต้น	144

10. ประกอบวงจร Smart Farm system เบื้องต้นโดยใช้บอร์ด Arduino UNO R3 ตามรูปที่ 18.2



รูปที่ 18.2 แสดงวงจร Smart Farm system เบื้องต้น

11. Upload โปรแกรม LAB18_1.ino ลงบอร์ด Arduino UNO R3

12. สังเกตและบันทึกผลการทดลอง

.....

.....

.....

13. สรุปผลการทดลอง

.....

.....

.....

.....

.....

บรรณานุกรม

เดชฤทธิ์ มณีธรรม. คัมภีร์การใช้งาน ไมโครคอนโทรลเลอร์ Arduino. กรุงเทพฯ : ซีเอ็ดดูเคชั่น, 2560.

ประภาส สุวรรณเพชร. เรียนรู้และลงเล่น Arduino เบื้องต้น. (เอกสารประกอบการสอน). ชัยภูมิ: วิทยาลัยเทคนิคชัยภูมิ.

เว็บไซต์อ้างอิง

“งานที่มอบหมาย ไมโครคอนโทรลเลอร์.” 2561. [ระบบออนไลน์]. แหล่งที่มา www.praphas.com

“618353 ปฏิบัติการไมโครโปรเซสเซอร์และการเชื่อมต่อ.” 2560. [ระบบออนไลน์]. แหล่งที่มา <https://sites.google.com/site/618353/ls>



ใบงานการทดลอง

วิชาไมโครคอนโทรลเลอร์

รหัสวิชา 3105-2007

เนื้อหาใบงานประกอบด้วย

- ใบงานการทดลองที่ 1 เรื่องงานโปรแกรม Proteus และ Arduino IDE เบื้องต้น
- ใบงานการทดลองที่ 2 เรื่องงานโปรแกรมการเชื่อมต่อเอาท์พุทด้วย LED แบบต่าง ๆ
- ใบงานการทดลองที่ 3 เรื่องงานโปรแกรมรับค่า Analog INPUT มาแสดงผลที่ LED 7-Segment
- ใบงานการทดลองที่ 4 เรื่องงานโปรแกรมรับค่า Analog INPUT มาควบคุม LED แบบต่างๆ
- ใบงานการทดลองที่ 5 เรื่องงานโปรแกรมรับค่า Push Switch มาควบคุม LED แบบต่างๆ
- ใบงานการทดลองที่ 6 เรื่องงานโปรแกรมรับค่า Keypad มาแสดงผลที่ LED 7-Segment
- ใบงานการทดลองที่ 7 เรื่องงานโปรแกรมควบคุม Buzzer
- ใบงานการทดลองที่ 8 เรื่องงานโปรแกรมควบคุม Buzzer ร่วมกับ Keypad
- ใบงานการทดลองที่ 9 เรื่องงานโปรแกรมรับค่าจากอุปกรณ์ INPUT แบบต่าง ๆ แสดงผลด้วย LCD
- ใบงานการทดลองที่ 10 เรื่องงานโปรแกรมการเชื่อมต่อกับ LDR แสดงผลด้วย LCD
- ใบงานการทดลองที่ 11 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยแสงอินฟราเรด GP2Y0A21
- ใบงานการทดลองที่ 12 เรื่องงานโปรแกรมเชื่อมต่อโมดูลวัดระยะทางด้วยอัลตราโซนิก HC-SR04
- ใบงานการทดลองที่ 13 เรื่องงานโปรแกรมการเชื่อมต่อกับ PIR Sensor
- ใบงานการทดลองที่ 14 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิด้วย DS18B20
- ใบงานการทดลองที่ 15 เรื่องงานโปรแกรมการอ่านค่าอุณหภูมิและค่าความชื้นจาก DHT11
- ใบงานการทดลองที่ 16 เรื่องงานการเขียนโปรแกรมควบคุมมอเตอร์
- ใบงานการทดลองที่ 17 เรื่องงานควบคุมเครื่องใช้ไฟฟ้าขนาดใหญ่ด้วยรีเลย์
- ใบงานการทดลองที่ 18 เรื่องงานโปรแกรม Smart Farm system เบื้องต้น

นายสง่า คุณำ

แผนกวิชาช่างอิเล็กทรอนิกส์ วิทยาลัยเทคนิคพัทยา