บทที่ 1
บทนำ

1.1 ปัญหาและที่มาของโครงงาน

เนื่องจากในปัจจุบันมีจำนวนประชากรเพิ่มมากขึ้นและมีความเป็นไปได้สูงที่ทางผู้ให้บริการในด้านต่าง ๆ เช่น ห้างสรรพสินค้า ห้องสมุด โรงพยาบาล ฯ ย่อมอยากรู้จำนวนคนเข้าออกและจำนวนคนที่อยู่ภายในอาคารเพื่อที่จะได้พัฒนาหรือปรับปรุงสถานที่ที่ให้บริการของตัวเองให้มีประสิทธิภาพมากขึ้น หลักการทำงานของวงจรอินฟราเรด ใช้กับตัวส่งแบบโมดูล รับสำเร็จรูป(3ขา) จะส่งด้วยความถี่ 40KHzโดยประมาณ ประโยชน์เพื่อเป็นความถี่หลักในการ ตรวจรับว่าเป็นสัญญาณตัวจริงไม่ใช้สัญญาณรบกวน ตัวรับแบบโมดูล (3ขา) โมดูลจะรับ สัญญาณที่กระพริบจะกระพริบด้วยความถี่ประมาณ 40KHz ถ้าค่าตรงก็จะให้เอาท์พุทที่ขาเอาท์พุทเป็น 0 หรือ 1 หลักการของมันก็มีแค่ส่งแสงอินฟราเรดไปยังวัตถุที่ต้องการตรวจจับ ถ้าพบวัตถุนั้นก็จะ สะท้อนแสงกลับมายังตัวรับ สีที่สะท้อนได้ดีที่สุดก็คือสีขาว ถ้าเป็นสีดำจะถูกดูดกลืนได้มากกว่า

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อนับและแสดงจำนวนคนเข้าออก ทราบจำนวนคนที่อยู่ภายในอาคารได้จริง
2. เพื่อศึกษาและออกแบบระบบ Sensor ไร้สายต้นแบบ ที่ต่อเชื่อมกับระบบ Infrared Sensor และ Smart Floors

1.3 ขอบเขตการทำงาน

1. ศึกษาการทำงานของ Sensor ทั้งสองตัว
2. เขียนโปรแกรมควบคุมการนับของ Sensor ทั้งสองตัว
3. เขียนโปรแกรมเพื่อแสดงกราฟและจำนวนคน
4. ศึกษาการรับส่งข้อมูลแบบไร้สาย

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาค้นคว้าหาข้อมูล
2. เขียนโครงการและเสนอโครงการกับอาจารย์ที่ปรึกษา
3. หาซื้ออุปกรณ์ที่ใช้ในโครงการนี้
4. เขียนโปรแกรมที่ใช้กับ Sensor และ Arduino
5. ทดสอบการทดลองเพื่อให้ได้ตามวัตถุประสงค์
6. สรุปผลการทดลองและเขียนรายงาน
7. นำเสนอโครงงาน

1.5 ผลที่คาดว่าจะได้รับจากโครงงาน

1. สามารถใช้นับจำนวนคนได้จริง
2. สามารถทำงานเป็นทีมได้
3. สามารถนำทักษะการแก้ไขปัญหาเฉพาะหน้าและความรู้ที่ได้ไปใช้ในการประกอบวิชาชีพได้

บทที่ 2
หลักการและทฤษฎี

2.1 บทนำ

ในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องที่ใช้ในการทำโครงงานนี้ โดยในส่วนนี้จะกล่าวถึงหลักการทำงานของเซ็นเซอร์อินฟราเรด,ตัวรับส่งสัญญาณแบบไร้สาย **Xbee** และ บอร์ดไมโครคอนโทรลเลอร์ **Arduino** ที่เราได้นำมาประยุกต์ใช้กับโครงงานนี้

เซ็นเซอร์อินฟราเรด (Infrared Sensor) เป็นอุปกรณ์ที่ใช้สำหรับป้องกันการชนหรือกระทบกับสิ่งกีดขวางที่อยู่ข้างหน้า ตัวบอร์ดอินฟราเรดใช้ได้กับบอร์ดไมโครคอนโทรลเลอร์ทั่วไปได้การทำงานของบอร์ดใช้ Sharp IR Module ที่ประกอบด้วย Amplifiers, Filte, Limiter ทำหน้าที่ Detect หรือตรวจจับสัญญาณที่เข้ามา ตัวตรวจจับจะตอบสนองต่อความถี่ที่ได้กำหนดไว้เพื่อช่วยกำจัดเสียง แสงสว่าง และแสงไฟที่จะเข้ามารบกวนการทำงานของตัว Detector บนบอร์ดจะเห็นว่ามีตัว IR LED อยู่ 2 ตัวทั้งทางด้านซ้ายและด้านขวา ทำหน้าที่ส่งสัญญาณความถี่ออกไปโดยใช้ความถี่ที่ 34-40KHz ที่ถูกกำหนดไว้แล้ว เมื่อความถี่ที่ถูกส่งออกไปกระทบกับวัตถุที่อยู่ข้างหน้าก็จะสะท้อนความถี่ของแสงอินฟราเรดกลับมายังตัว Detector โดยจะตรวจจับวัตถุได้ทั้งทางด้านซ้ายและทางด้านขวาซึ่งส่วนใหญ่จะใช้ให้กับ IR LED ตัวบอร์ดได้ติดตั้ง Regulator 5V สำหรับปรับแรงดันไฟไว้แล้ว จึงไม่ต้องใช้แหล่งจ่ายไฟ 5V จากภายนอกตัวอินฟราเรดเซ็นเซอร์สามารถปรับขอบเขตความกว้างของแสงได้ 5-8 นิ้ว

ในการนำไปใช้งานเป็นอุปกรณ์การพัฒนาหุ่นยนต์ที่ใช้สำหรับค้นหาวัตถุหรือป้องกันการชนกับสิ่งกีดขวางที่อยู่ข้างหน้า ตัวบอร์ดเซ็นเซอร์อินฟราเรดใช้กับบอร์ดไมโครคอนโทรลเลอร์ทั่วไปได้และสามารถนำบอร์ดเซ็นเซอร์อินฟราเรดนี้ไปใช้สร้างหุ่นยนต์ประเภทซูโม่เพื่อค้นหาคู่ต่อสู้ในเกมการแข่งขันได้อีกด้วย

2.2 อุปกรณ์ที่ใช้สำหรับการออกแบบ

2.2.1 เซ็นเซอร์อินฟราเรด (Infrared Sensor)

รูปที่ 2.1 Infrared Distance Sensor [E18-D80NK]

อุปกรณ์ตัวนี้ใช้แสง Infrared ในการค้นหาวัตถุที่อยู่ด้านหน้า Sensor โดยส่งสัญญาณออกและรับสัญญาณที่สะท้อนกลับ วัดระยะได้ใยช่วง 3-80ซม. และมุมในการวัด ±15 องศา

การทำงานของบอร์ดใช้ Sharp IR Module ที่ประกอบด้วย Amplifiers, Filter, Limiter ทำหน้าที่ Detect หรือตรวจจับสัญญาณที่เข้ามา ตัวตรวจจับจะตอบสนองต่อความถี่ที่ได้กำหนดไว้เพื่อช่วยกำจัดเสียง แสงสว่าง และแสงไฟที่จะเข้ามารบกวนการทำงานของตัว Detector บน บอร์ดจะเห็นว่ามีตัว IR LED อยู่ 2 ตัวทั้งทางด้านซ้ายและด้านขวา ทำหน้าที่ส่งสัญญาณ ความถี่ออกไปโดยใช้ความถี่ที่ 34-40KHz ที่ถูกกำหนดไว้แล้ว เมื่อความถี่ที่ถูกส่งออก ไปกระทบกับวัตถุที่อยู่ข้างหน้าก็จะสะท้อนความถี่ของแสงอินฟราเรดกลับมายังตัว Detector โดยจะตรวจจับวัตถุได้ทั้งทางด้านซ้ายและทางด้านขวาซึ่งส่วนใหญ่จะใช้ให้กับ IR LED ตัวบอร์ดได้ติดตั้ง Regulator 5V สำหรับปรับแรงดันไฟไว้แล้วจึงไม่ต้องใช้ แหล่งจ่ายไฟ 5V จากภายนอกตัวเซ็นเซอร์อินฟราเรดสามารถปรับขอบเขตความกว้างของแสงได้ 5-8 นิ้ว

2.2.1.1 โครงสร้างพื้นฐานของเซ็นเซอร์อินฟราเรด

รูปที่ 2.2 โครงสร้างพื้นฐานของ Infrared distance sensor (E18-D80NK)

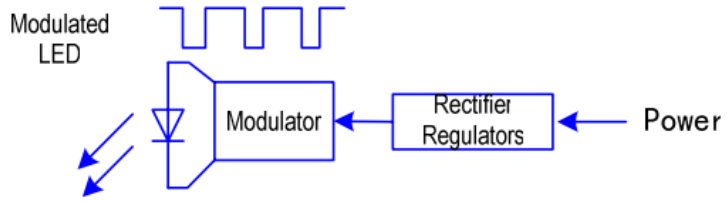
ตารางที่ 2.1 แสดงรายละเอียดของขาเซ็นเซอร์อินฟราเรด

Pin	
Red	+5V DC
Green	Ground
Yellow	Output

ตารางที่ 2.2 แสดงคุณสมบัติของเซ็นเซอร์อินฟราเรด

Features	
Power Supply	5V DC
Supply Current DC	<25mA
Maximum Load Current	100mA (Open-Collector NPN Pull Down Output)
Response Time	<2ms
Diameter	17MM
Pointing Angle	≤ 15 °
Effective From	3-80CM Adjustable
Detection of Objects	Transparent or Opaque
Working Environment Temperature	-25℃+55℃
Case Material	Plastic
Lead Length	45CM

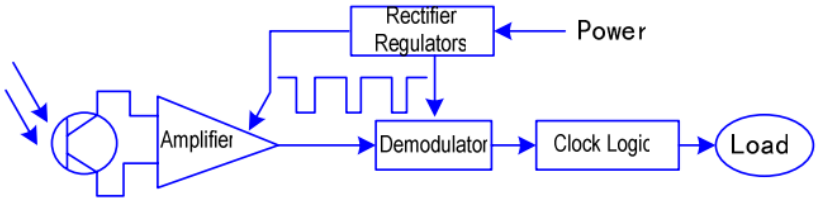
2.2.1.2 ตัวส่ง(Transmitter)



รูปที่ 2.3 ตัวส่ง(Transmitter)

อินฟราเรด (มองด้วยตาเปล่าไม่เห็น) และให้ความเข้มแสงสูงสุดที่เฉพาะค่าความถี่เท่านั้น LED ประเภทนี้มีลักษณะเหมือน LED ทั่วไป มี 2 ขา คือ แอโนด กับ แคโทด ดังนั้นการต่อใช้งาน ก็เหมือนกรณี LED ทั่วไป LED ที่ให้แสงอินฟราเรดแต่ละชนิด สามารถทนกระแสสูงสุด (mA) ได้แตกต่างกัน

2.2.1.3 ตัวรับ(Receiver)



รูปที่ 2.4 ตัวรับ(Receiver)

โฟโต้ทรานซิสเตอร์ (Phototransistor) โฟโต้ทรานซิสเตอร์เป็นทรานซิสเตอร์ที่รวมเอาโฟโต้ไดโอดมาไว้ภายในวงจรเดียวกัน ดังแสดงในรูปด้านล่าง โดยให้โฟโต้ไดโอดทำหน้าที่เป็นตัวไบแอสกระแสให้แก่ขาเบสของทรานซิสเตอร์ ดังนั้น เมื่อมีแสงตกกระทบที่โฟโต้ไดโอด จะเกิดการนำกระแสที่ขาเบส ทรานซิสเตอร์จึงสามรถนำกระแสได้

โฟโต้ทรานซิสเตอร์จะมีกระแสรั่วไหลมากกว่าโฟโต้ไดโอดเล็กน้อย และสามารถนำกระแสได้มากกว่า อย่างไรก็ตาม แม้ความเร็วในการทำงานของโฟโต้ทรานซิสเตอร์จะมากกว่า LDR แต่ก็ยังน้อยกว่าโฟโต้ไดโอด ดังนั้น โฟโต้ทรานซิสเตอร์จึงจำกัดการใช้งานอยู่ในวงจรที่มีความเร็วในการทำงานไม่เกิน 100 กิโลเฮิรตซ์

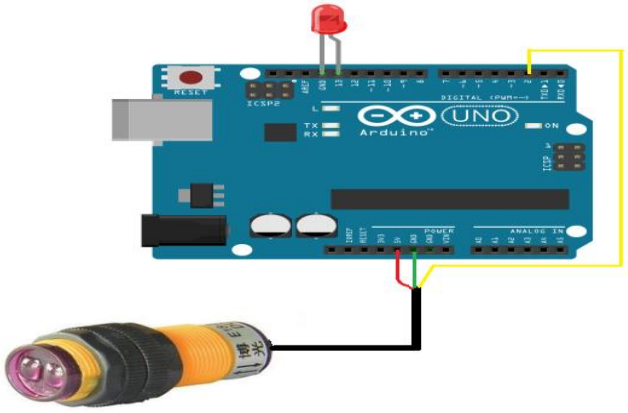
2.2.1.4 ข้อดีของคลื่นอินฟราเรด

- ใช้พลังงานน้อยจึงนิยมใช้กับเครื่อง Laptops, โทรศัพท์
- แผงวงจรควบคุมราคาต่ำ (Low Circuitry Cost) เรียบง่ายและสามารถเชื่อมต่อกับระบบอื่นได้อย่างรวดเร็ว
- มีความปลอดภัยในการเรื่องข้อมูลสูง ลักษณะการส่งคลื่น(Directionality of the Beam)จะไม่รั่วไปที่เครื่องรับตัวอื่นในขณะที่ส่งสัญญาณ
- คลื่นแทรกจากเครื่องใช้ไฟฟ้าใกล้เคียงมีน้อย (High Noise Immunity)

2.2.1.5 ข้อเสียของอินฟราเรด

- เครื่องส่ง(Transmitter) และเครื่องรับ (Receiver) ต้องอยู่ในแนวเดียวกัน คือต้องเห็นว่าอยู่ในแนวเดียวกัน
- คลื่นจะถูกกันโดยวัตถุทั่วไปได้ง่ายเช่น คน กำแพง ต้นไม้ ทำให้สื่อสารไม่ได้
- ระยะทางการสื่อสารจะน้อย ประสิทธิภาพจะตกลงถ้าระยะทางมากขึ้น
- สภาพอากาศ เช่นหมอก แสงอาทิตย์แรงๆ ฝนและมลภาวะมีผลต่อประสิทธิภาพการสื่อสาร
- อัตราการส่งข้อมูลจะช้ากว่าแบบใช้สายไฟทั่วไป

2.2.1.6 การเชื่อมต่อเซ็นเซอร์อินฟราเรดกับบอร์ดไมโครคอนโทรลเลอร์เบื้องต้น



รูปที่ 2.5 การเชื่อมต่อเซนเซอร์อินฟราเรดกับบอร์ดไมโครคอนโทรลเลอร์เบื้องต้น

ขั้นตอนที่ 1 : เชื่อมต่อเซ็นเซอร์อินฟราเรดกับบอร์ดไมโครคอนโทรลเลอร์ ดังรูป 2.2.1(ฉ)

โดย - ขา Vcc ของเซ็นเซอร์ (สายสีแดง) ต่อกับ ขาแรงดัน 5 V ของบอร์ดไมโครคอนโทรลเลอร์

- ขา GND ของเซ็นเซอร์ (สายสีเขียว) ต่อกับ ขา GND ของบอร์ดไมโครคอนโทรลเลอร์

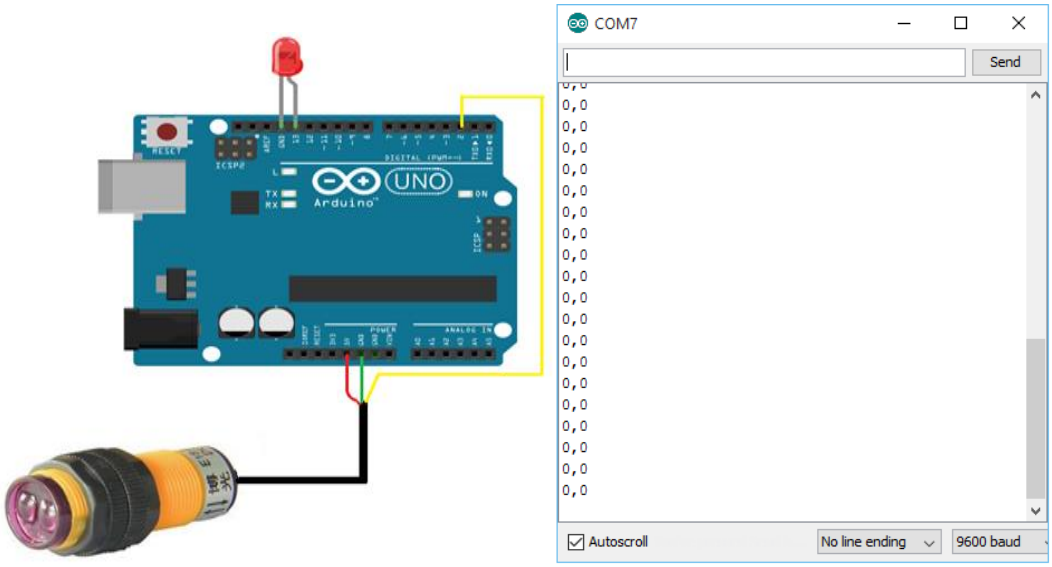
- ขา Signal ของเซ็นเซอร์ (สายสีเหลือง) ต่อกับ ขาดิจิตอลหมายเลข 2 ของบอร์ดไมโครคอนโทรลเลอร์

ขั้นตอนที่ 2 : ทำการเขียนโค้ดโปรแกรมและรันโปรแกรม

```
const int sensors1=10;

const int sensors2=9;

// the setup function runs once when you press reset or power the board

void setup() {

pinMode(sensors1,INPUT);

pinMode(sensors2,INPUT);

Serial.begin(9600);

}

// the loop function runs over and over again forever

void loop() {

int state1=digitalRead(sensors1);

int state2=digitalRead(sensors2);

if(state1==LOW){

Serial.print("0");

}else{
```

```
Serial.print("500");
 }
 if(state1==HIGH){
 state2=LOW;
 delay(1000);
 }
 Serial.print(",");
 if(state2==LOW){
 Serial.print("0");
 }else{
 Serial.print("500");
  }
 if(state2==HIGH){
  state1=LOW;
  delay(1000);
  }
  Serial.println();
 delay(10);
}
```

ขั้นตอนที่ 3 : ไปที่เมนู Serial Monitor ของโปรแกรม Arduino IDE ซึ่งจะได้ผลลัพธ์ดังรูปที่ 2.6



รูปที่ 2.6 ผลลัพธ์จากการรันโปรแกรมที่ Serial Monitor

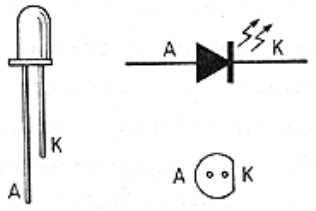
2.2.2 ไดโอดเปล่งแสง (Light Emitting Diode , LED)

รูปที่ 2.7 ไดโอดเปล่งแสง (LED)

ไดโอดเปล่งแสง (Light-Emitting Diode , LED) เป็นอุปกรณ์สารกึ่งตัวนำอย่างหนึ่ง จัดอยู่ในจำพวกไดโอด ที่สามารถเปล่งแสงในช่วงสเปกตรัมแคบ เมื่อถูกไบอัสทางไฟฟ้าในทิศทางไปข้างหน้า ปรากฏการณ์นี้อยู่ในรูปของ Electroluminescence สีของแสงที่เปล่งออกมานั้นขึ้นอยู่กับองค์ประกอบทางเคมีของวัสดุ กึ่งตัวนำที่ใช้และเปล่งแสงได้ใกล้ช่วงอัลตราไวโอเลตช่วงแสงที่มองเห็นและช่วงอินฟราเรดผู้พัฒนาไดโอดเปล่งแสงขึ้นเป็นคนแรก คือนิกโฮโลยัก

(Nick Holonyak Jr.) (เกิด ค.ศ. 1928) แห่งบริษัทเจเนรัล อิเล็กทริก (General Electric Company) โดยได้พัฒนาไดโอดเปล่งแสงในช่วงแสงที่มองเห็น และสามารถใช้งานได้ในเชิงปฏิบัติเป็นครั้งแรก เมื่อ ค.ศ. 1962

ไดโอดเปล่งแสงหรือ LED นี้สามารถนำไปใช้งานในการแสดงผลโดยทั่วไป ถ้าไม่ต้องการความสว่างมากเมื่อเปรียบเทียบกับหลอดไฟธรรมดาแล้วจะเห็นว่าไดโอดเปล่งแสงนี้ สามารถทำงานโดยใช้แรงดันและกระแสไฟที่น้อยกว่ามากปกติจะใช้กระแสอยู่ช่วงระหว่าง 5 – 20 mA



รูปที่ 2.8 สัญลักษณ์ และตัวถังของไดโอดเปล่งแสง

ไดโอดเปล่งแสงนี้มีรูปร่างและขนาดต่างๆกันตามการใช้งาน ไดโอดเปล่งแสงแบบหัวมนโดยปกติ จะมีขนาดเส้นผ่านศูนย์กลาง 3 มิลลิเมตร และ 5 มิลลิเมตร ซึ่งทำด้วยพลาสติกโปร่งแสงดูตัวอย่างในรูปที่ 2.7 และถ้าเป็นแบบสี่เหลี่ยมจะมี ขนาด 5*2 มิลลิเมตรมุมในการมองเห็นของไดโอดเปล่งแสงแบบหัวมนนี้จะอยู่ในช่วง 20 – 40 องศา แต่ถ้าเป็นไดโอดเปล่งแสงแบบสี่เหลี่ยมมุมในการมองเห็นจะเพิ่มขึ้นเป็นถึง 100 องศา คุณสมบัติของไดโอดเปล่งแสงโดยทั่วไปแสดงไว้ดังตารางหน้าถัดไป

ตารางที่ 2.3 คุณสมบัติทั่วไปของไดโอดเปล่งแสง

Parameter	standard	Type of LED		
		standard	high efficiency	high itensity
Diameter (mm)	3	5	5	5
Max. forward Current (mA)	40	30	30	30
Typical forward Current (mA)	12	10	7	10
Typical forward Voltage drop (V)	2.1	2.0	1.8	2.2
Max. reverse voltage (V)	5	3	5	5
Max.Power dissipation (mW)	150	100	27	135
Peak wavelenght (nm)	690	635	635	635

รูปที่ 2.9 วงจรใช้งานปกติ ของไดโอดเปล่งแสง

ปกติการใช้งานไดโอดเปล่งแสงก็จะต่อดังรูปที่ 2.9 สามารถคำนวณได้ดังสูตร R = VF คือ แรงดันตกคร่อมไดโอดเปล่งแสงขณะนำกระแสเพื่อความสะดวกในการคำนวณจะกำหนดค่า VF เท่ากับ 2 V จะได้ค่าของตัวต้านทานที่ใช้ต่ออนุกรมกับไดโอดเปล่งแสงที่แรงดันต่างๆ ดังตารางข้างล่างนี้

ตารางที่ 2.4 กำหนดค่าตัวต้านทานอนุกรมของไดโอดเปล่งแสง

Supply Voltage (V)	Low power LED(5 mA nom)	Standard LED (10 mA nom)	High power LED (20 mA nom)
3	220	180	56
5	680	270	150
6	820	390	220
9	1.5k	680	390
12	2.2k	1k	560
15	2.7k	1.2k	680
18	3.3k	1.5k	820
24	4.7k	2.2k	1.2k

ตัวอย่าง

สมมติว่าต้องการนำไดโอดเปล่งแสงไปใช้กับแรงดัน 21 โวลต์ และมีแรงดันตกคร่อมประมาณ 2.2 โวลต์ยอมให้กระแสผ่าน 15 mA ลองคำนวณค่าของตัวต้านทานที่จะนำมาต่ออนุกรมกับไดโอดเปล่งแสงจากสูตรข้างต้น R = ค่าความต้านทานที่มีขนาดใกล้เคียงคือ 1.2 กิโลโอห์ม และค่ากำลังสูญเสียของตัวต้านทาน ก็สามารถประมาณได้จาก

$$P = I^2R$$

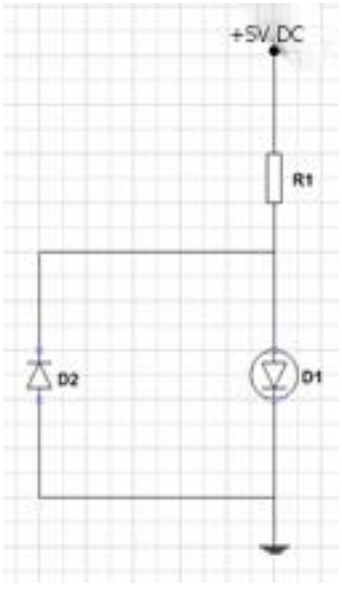
$$= (15\ mA)^2 * 1.2\ kW$$

$$= 0.27\ W$$

ดังนั้นควรเลือกตัวต้านทานให้มีอัตรากำลังสูญเสียตั้งแต่ 0.5 วัตต์ขึ้นไป

2.2.1 ข้อจำกัดในการใช้ไดโอดเปล่งแสง

- ข้อควรระวังอย่างหนึ่งในการใช้งานไดโอดเปล่งแสงก็คือแรงดันย้อนกลับจะต้องมีค่า ไม่เกิน 5 V
- สำหรับการใช้งานบางอย่างที่ใช้กับแบตเตอรี่นั้นจะต้องดูจำนวนของไดโอดเปล่งแสงที่ใช้ด้วย ถ้าต้องการให้ใช้ได้นานๆ ปกติจะกำหนดให้ไดโอดเปล่งแสงดวงหนึ่งกินกระแสเพียง 5 mA
- สำหรับไดโอดเปล่งแสงสีเหลืองและสีเขียวโดยปกติจะให้ความสว่างน้อยกว่าไดโอดเปล่งแสงสีแดงที่ระดับกระแสเท่ากัน ถ้าต้องการให้ระดับความสว่างออกมาเท่ากันในกรณีที่ใช้ไดโอดเปล่งแสงสีแตกต่างกันจะต้องเปลี่ยนค่าตัวต้านทานจำกัดกระแสที่อนุกรมโดยคำนวณหาได้ตามสูตรปกติ จากนั้นลดค่าที่คำนวณได้ลงไปอีก 10 - 15 เปอร์เซ็นต์แต่ก็ควรระวังปริมาณกระแสไฟฟ้าต้องไม่ให้เกินค่าทนได้สูงสุดที่กำหนดไว้ด้วย



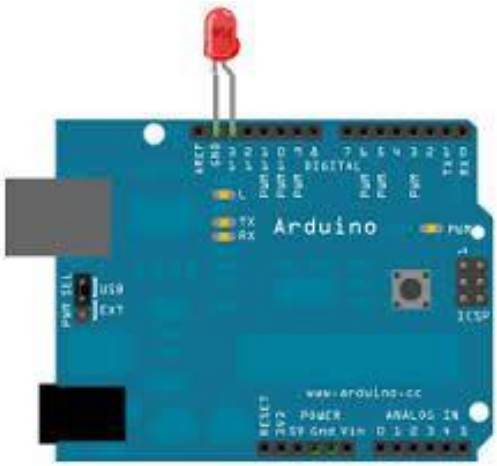
รูปที่ 2.10 การนำไดโอดเปล่งแสงมาใช้กับไฟสลับแรงดันต่ำ

จากรูปที่ 2.10 ไดโอดเปล่งแสงก็สามารถไปนำใช้กับแหล่งจ่ายไฟฟ้ากระแสสลับแรงดันต่ำขนาดน้อยกว่า 50 โวลต์ได้ โดยการใช้ไดโอดต่อขนานกับไดโอดเปล่งแสง

2.2.2 การเชื่อมต่อไดโอดเปล่งแสง(LED) เข้ากับบอร์ดไมโครคอนโทรลเลอร์ เบื้องต้น

ขั้นตอนที่ 1 : เชื่อมต่อไดโอดเปล่งแสง (LED) เข้ากับบอร์ดไมโครคอนโทรลเลอร์ดังรูปโดย

- ขั้วแคโทดของไดโอดเปล่งแสงต่อกับขาดิจิตอลหมายเลข 13 ของบอร์ดไมโครคอนโทรลเลอร์
- ขั้วแอโนดของไดโอดเปล่งแสงต่อกับขา GND ของบอร์ดไมโครคอนโทรลเลอร์



รูปที่ 2.11 การเชื่อมบอร์ด Arduino กับ ไดโอดเปล่งแสง (LED) เบื้องต้น

ขั้นตอนที่ 2 : ทำการเขียนโค้ดโปรแกรมทดสอบดังนี้

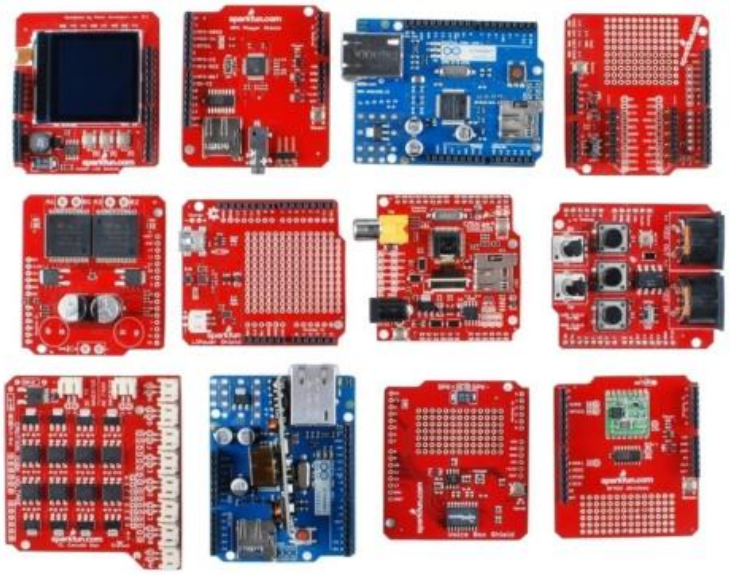
```
int led = 13;
void setup() {
pinMode(led, OUTPUT);
}
void loop() {
digitalWrite(led, HIGH);
delay(1000);
digitalWrite(led, LOW);
delay(1000);
}
```

ขั้นตอนที่ 3 : เมื่อทำรันโค้ดโปรแกรมแล้ว จะได้ผลลัพธ์ดังภาพ

รูปที่ 2.12 ผลลัพธ์จากรันโปแกรมของการเชื่อมต่อ LED กับบอร์ด Arduino

2.2.3 ชุดควบคุมบอร์ดไมโครคอนโทรลเลอร์ (Microcontroller)

บอร์ดไมโครคอนโทรลเลอร์ Arduino



รูปที่ 2.13 บอร์ดไมโครคอนโทรลเลอร์ Arduino รุ่นต่างๆ

บอร์ด Arduino เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัวบอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษาทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์ เสริมต่างๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิคส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม(Arduino Shield) ประเภทต่าง ๆ เช่น Arduino XBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield, Arduino GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย

2.2.3.1 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม

- ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น
- มี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง
- Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
- มีราคาไม่แพง
- Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได

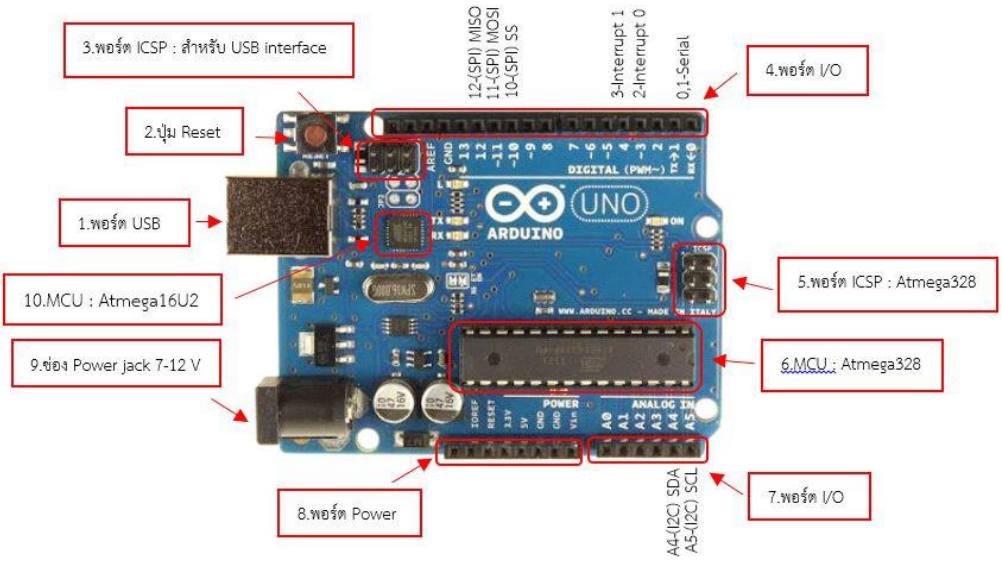
2.2.3.2 บอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในโครงงาน

Arduino Uno R3

รูปที่ 2.14 บอร์ด Arduino Uno R3

เป็นบอร์ด Arduino ที่ได้รับความนิยมมากที่สุดเนื่องจากราคาไม่แพง ส่วนใหญ่โปรเจคและ Library ต่างๆ ที่ พัฒนาขึ้นมา Support จะอ้างอิงกับบอร์ดนี้เป็นหลัก และข้อดีอีกอย่างคือ กรณีที่ MCU เสีย ผู้ใช้งานสามารถซื้อมาเปลี่ยนเองได้ง่าย

2.2.3.3 โครงสร้างพื้นฐานของบอร์ด Arduino Uno R3



รูปที่ 2.15 โครงสร้างพื้นฐานของบอร์ด Arduino Uno R3

โครงสร้างพื้นฐานของบอร์ด Arduino Uno R3 นั้น ประกอบด้วย

1. **USB Port :** ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด

2. **Reset Button** : เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่

3. **ICSP Port** : Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com Port บน Atmega16U2

4. **I/O Port :** Digital I/O ตั้งแต่ ขา D0 ถึง D13 นอกจากนี้บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วยเช่น Pin0,1 เป็นขา Tx , Rx Serial , Pin3,5,6,9,10 และ 11 เป็นขา PWM

5. **ICSP Port :** Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Boot Loader

6. **MCU :** Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino

7. **I/O Port :** นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็นช่องรับสัญญาณอนาล็อกตั้งแต่ขา A0-A5

8. **Power Port :** ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง+3.3V,+5V, GND, Vin

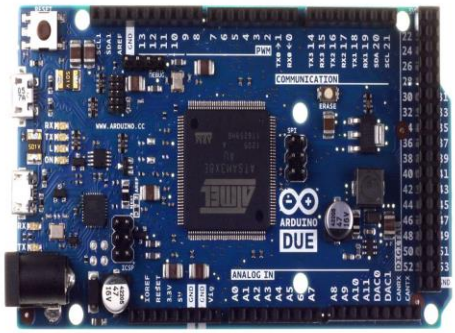
9. **Power Jack :** รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V

10. **MCU ของ Atmega16U2** : เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

คุณสมบัติทั่วไปของบอร์ด Arduino Uno R3

- ใช้ชิพ ATmega328 รันที่ความถี่ 16 MHz
- หน่วยความจำแฟลช 32 KB แรม 2 KB
- บอร์ดใช้ไฟเลียง 7 ถึง 12 V
- มีระดับแรงดันไฟฟ้าในการทำงานและขาสัญญาณอยู่ที่ 5 V
- (TTL) มี Digital Input / Output 14 ขา (เป็น PWM ได้ 6 ขา)
- มี Analog Input 6 ขา Serial UART 1 ชุด เป็นพอร์ตสื่อสารอนุกรม
- มี ชุดI2C 1 ชุด SPI 1 ชุด

ArduinoDUE

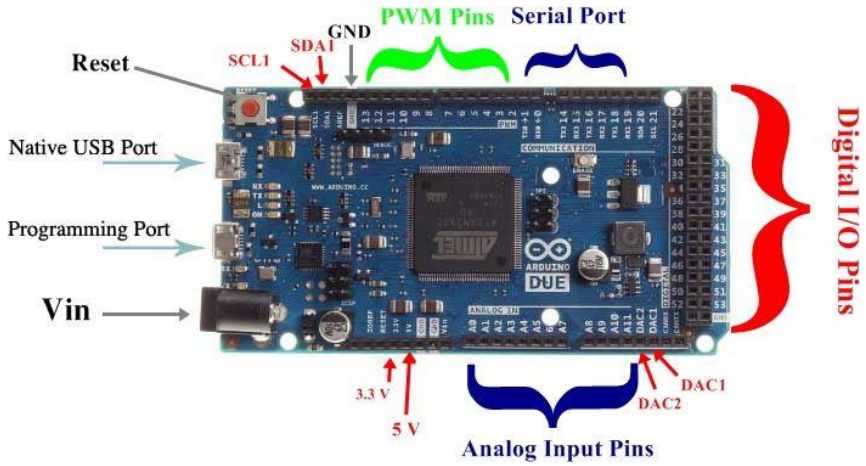


รูปที่ 2.16 บอร์ด Arduino DUE

เป็นบอร์ด Arduino ที่เปลี่ยนชิป MCU ใหมซึ้งจากเดิมเป็นตระกูล AVR เปลี่ยนเป็นเบอร์ AT91SAM3X8E(ตระกูล ARM Cortex-M3) แทน ทำให้การประมวลผลเร็วขึ้น แต่ยังคงรูปแบบโค้ดโปรแกรมของ Arduino ที่ง่ายอยู่

ข้อควรระวัง : เนื่องจาก MCU เป็นเบอร์ที่ต่างกับ Arduino Uno R3 อาจะทำให้บอร์ด Shield บางชนิดหรือ Library ใช้ร่วมกันกับบอร์ด Arduino Leonardo ไม่ได้ ผู้ใช้งานจำเป็นต้องตรวจสอบก่อนใช้งาน

2.2.3.4 โครงสร้างพื้นฐานของบอร์ด Arduino DUE



รูปที่ 2.17 โครงสร้างพื้นฐานของบอร์ด Arduino DUE

โครงสร้างของบอร์ด Arduino DUE นั้นประกอบด้วย

1. **USBPort** : ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด

2. **Reset Button** : เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มมีการทำงานใหม่

3. **ICSP Port** ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com Port บน Atmega16U2

4. **I/O Port** : Digital I/O ตั้งแต่ ขา D0 ถึง D13 นอกจากนี้บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วยเช่น Pin0,1 เป็นขา Tx , Rx Serial, Pin14,15,16,17,18 และ 19 เป็นขา PWM

5. **ICSP Port :** Atmega16U2 เป็นพอร์ตที่ ใช้โปรแกรม Boot loade

6. **MCU:** AT91SAM3X8E เป็น MCU ที่ใช้บนบอร์ด Arduino

7. **I/OPort** : นอกจากจะเป็น Digital I/O แล้วยังเปลี่ยนเป็นช่องรับสัญญาณอนาล็อก ตั้งแต่ ขา A0-A11

8. **Power Port** : ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin

9. **Power Jack** : รับไฟจาก Adapter โดยที แรงดันอยู่ระหว่าง 7-12 V

10. **MCU** ของ Atmega16U2 เป็น MCU ที่ทำหน้าทีเป็น USB to Serial โดย AT91SAM3X8E จะติดต่อกับ Computer ผ่านAtmega16U2

11. **AREF** : เป็นขาอ้างอิงสำหรับแรงดับของขาอนาล็อก

12**. Native USB Port** : เป็นพอร์ตที่สามารถใช้เชื่อมต่อกับอุปกรณ์ ต่างๆใด เช่น เมาส์ คีย์บอร์ด และโทรศัพท์มือถือ เป็นต้น

13**. Programming Port :** เป็นพอร์ตที่ใช้เชื่อมต่อกับ USB Port โดยสามารถเป็นได้ทั้งขาที่ใช้สำหรับรับข้อมูลจากการเขียนโค้ดโปรแกรม Arduino IDE และ เป็นขาที่ สามารถเชื่อมกับไฟเลี้ยงจาก Computer ได้อีกด้วย

คุณสมบัติทั่วไปของบอร์ด Arduino DUE

- ไมโครคอนโทรลเลอร์ : AT91SAM3X8E จาก Atmel หน่วยความจำแฟลช 512 กิโลไบต์ แรม 96 กิโลไบต์ มีความสามารถ USB-OTG ทำหน้าที่เป็นอุปกรณ์ USB โฮสต์ เพื่อต่อกับเมาส์ คีย์บอร์ด และสมาร์ตโฟนได้
- ความถี่สัญญาณนาฬิกา : 84MHz
- ไฟเลี้ยงทำงาน : 3.3V
- ย่านไฟเลี้ยงอินพุต : 7 ถึง 12V
- พอร์ตอินพุตเอาต์พุต : 54 ช่อง (รวมเอาต์พุต PWM 12 ช่อง)
- อินพุตอนาล็อก : 12 ช่อง ความละเอียด 12 บิต รับแรงดัน 0 ถึง +3.3V
- เอาต์พุตอนาล็อก (DAC) : 2 ช่อง ความละเอียด 12 บิ ต ให้แรงดัน 0 ถึง +3.3V
- บัสสื่อสารข้อมูล : I^2C 2 ชุด, SPI, JTAG, CAN, UART 4 ชุด
- พอร์ตติดต่อคอมพิวเตอร์ : พอร์ต USB โดยใช้สาย microUSB

2.2.4 อะคริลิค หรือ อะคริลิคเรซิน (Acrylic Resins)

รูปที่ 2.18 แผ่นอะคริลิค หรือ อะคริลิคเรซิน (Acrylic Resins)

2.2.4.1 อะคริลิค/แผ่นอะคริลิค

ถือเป็นผลิตภัณฑ์พอลิเมอร์ที่ถูกนำมาใช้งานอย่างกว้างขวางในภาคอุตสาหกรรม สำหรับเป็นสารตั้งตั้งในการผลิตผลิตภัณฑ์ต่างๆส่งจำหน่ายแก่ภาคครัวเรือน หรือ ภาคอุตสาหกรรมที่เกี่ยวข้อง ทั้งในรูปอะคริลิคเหลวสำหรับภาคอุตสาหกรรม และแผ่นอะคริลิคหรือพลาสติกอะคริลิคสำหรับงานในด้านต่างๆ

อะคริลิค หรือ อะคริลิคเรซิน (Acrylic Resins) เป็นพอลิเมอร์ และ โคพอลิเมอร์ที่เตรียมได้จากกรดอะคริลิค และอนุพันธ์ของกรดอะคริลิค และเอสเทอร์ของ กรดอะคริลิค มีสูตรโครงสร้าง คือ CH_2= CHR โดยใช้สารตั้งต้น ได้แก่ Methyl Acrylate, Ethyl Acrylate และ Methyl Methacrylate ผลิตออกมาเป็นอะคริลิคที่นิยมใช้มากคือ Polymethyl Methacrylate (PMMA)

2.2.4.2 ประเภทของอะคริลิค

แบ่งได้เป็นประเภทต่างๆ คือ

1. อะคริลิคของแข็ง เป็นเทอร์โมพลาสติกเรซินที่อยู่ในรูปของแข็ง อาจเป็นเม็ดอะคริลิคหรือขึ้นรูปเป็นแผ่น เช่น แผ่นอะคริลิคหรือพลาสติกอะคริลิค ถือเป็นโฮโมพอลิเมอร์ของเมทาคริเลตเอสเทอร์ หรือโคพอลิเมอร์ของเมทาคริเลต
2. อะคริลิคเหลวเป็นอะคริลิคที่อยู่ในรูปสารละลายที่ผลิตได้จากกระบวนการพอลิเมอร์ไรเซชันแบบสารละลาย
3. อะคริลิคอิมัลชัน ใช้เป็นส่วนผสมของสีทาบ้าน มักผลิตในลักษณะของลาเท็กซ์

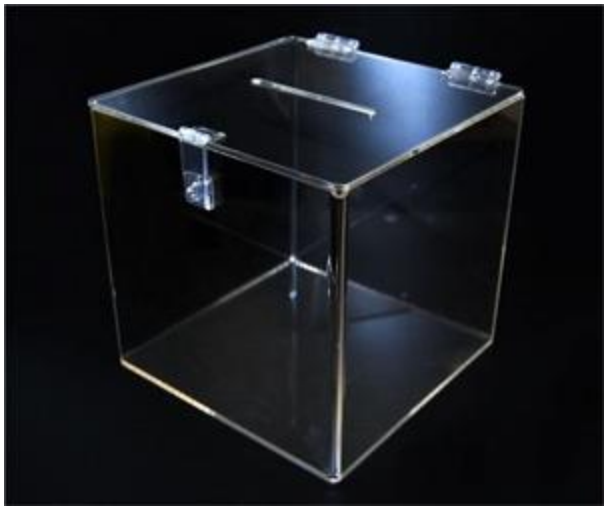
2.2.4.3 การใช้ประโยชน์ของอะคริลิค

- **อะคริลิคเหลว และอิมัลชันอะคริลิค**

มักใช้เป็นสารตั้งต้นในกระบวนการผลิตผลิตภัณฑ์ที่เกี่ยวข้องกับพลาสติก และใช้เป็นส่วนผสมของผลิตภัณฑ์อื่นๆ เช่น สี สารเคลือบวัสดุ เป็นต้น

- แผ่นอะคริลิคหรือพลาสิกอะคริลิค

เป็นผลิตภัณฑ์ของอะคริลิคของแข็งที่ถูกขึ้นรูปให้มีลักษณะเป็นแผ่น อาจเป็นแผ่นใสหรือแผ่นมีสีต่างๆ นิยมใช้งานมากในภาครัวเรือน เช่น ทำป้าย ทำเครื่องตกแต่งบ้าน วัสดุตกแต่งบ้าน ชั้นโชว์ ตู้เลี้ยงปลา ถังขยะ ของเด็กเล่น กรอบรูป กล่อง เป็นต้น



รูปที่ 2.19 การใช้ประโยชน์ของอะคริลิคในการทำผลิตภัณฑ์ต่างๆ

2.2.4.4 สมบัติทั่วไปของแผ่นอะคริลิค

– มีความโปร่งใสคล้ายกระจก

– ทนทานต่อแรงกระแทก แรงกด และสภาพแวดล้อมดินฟ้าอากาศ

– ทนทานต่อสารเคมีหลายชนิด ยกเว้นสารตัวทำละลาย และกรดที่ส่วนมากมีผลต่ออะคริลิค รวมถึงด่างแก่ทุกชนิด

– สามารถเติมแต่งด้วยสี ให้มีสีสันได้ตามความต้องการ

– มีจุดอ่อนตัวต่ำ ทนต่อความร้อน และมีความเหนียว

– มีสภาพคงรูปที่ดี และทนต่อการขีดข่วน

– เป็นฉนวนไฟฟ้า และฉนวนป้องกันความร้อนที่ดี

– ไม่ดูดความชื้น

2.2.4.5 การผลิตแผ่นอะคริลิค

การผลิตแผ่นอะคริลิคด้วยเทคนิคนี้ สามารถผลิตได้ในหลายลักษณะทั้งผลิตภัณฑ์ที่โปร่งแสงและทึบแสง สามารถผลิตได้หลากหลายสี และมีผิวหน้าหลายรูปแบบทั้งมันและไม่มัน

แผ่นอะคริลิคหรือพลาสติกอะคริลิค ผลิตได้จากอะคริลิคของแข็งด้วยกระบวนการการหล่อ สามารถทำได้ 2 วิธี ได้แก่

– กระบวนการหล่อแบบต่อเนื่อง (Continuous Casting Process)

– กระบวนการหล่อแบบไม่ต่อเนื่อง (Batch Casting Process) เช่น กระบวนการหล่อแบบเซลล์ (Cell Casting Process)

ความแตกต่างของทั้งสองวิธี คือ กระบวนการหล่อแบบต่อเนื่องจะเกิดในขั้นตอนลำเลียงของสายพานสองเส้นที่กำลังเคลื่อนที่ ส่วนกระบวนการหล่อแบบไม่ต่อเนื่องจะเกิดในเบ้าหล่อหรือในแม่พิมพ์

2.2.4.6 สารเคมีที่มีผลต่ออะคริลิค

<u>สารเคมีที่มีผลรุนแรงอะคริลิค</u>

– กรดไฮโดรคลอริก 40%
– กรดไฮโดรฟลูออริก 40%
– กรดไนตริก 40%
– กรดซัลฟูริก 98%
– ด่างแก่ทุกชนิด
– คลอโรฟอร์ม
– ไตคลอโรอีทิลลีน
– เมทิลลีนคลอไรด์
– คาร์บอนเตตระคลอไรด์
– อะซิโตน
– เมทิล
– เอทิล
– คีโตน
– โทลูอีน

<u>สารเคมีที่มีผลน้อยต่ออะคริลิค</u>

– กรดอะซิตริก 5%
– กรดโครมิก 40%
– กรดซิตริก 10%
– กรดไฮโดรฟลูออริก 10%
– กรดไนตริก 10%
– กรดซัลฟูริก 30%
– แอมโมเนียมไฮดรอกไซด์ 28%
– โซเดียมคาร์บอเนต 20%
– โซเดียมไฮดรอกไซด์ 60%

2.2.5 XBee

รูปที่ 2.20 XBee

ลักษณะการทำงานของ XBee

แบ่งได้เป็น 3 แบบ ดังนี้

1. Coordinator มีหน้าที่สร้างการสื่อสาร เชื่อมโยงเครือข่าย ระหว่าง End device กับ Router หรือ Coordinator กับ Coordinator ด้วยกัน หรือ Coordinator กับ Router กำหนดตำแหน่งที่อยู่ให้กับอุปกรณ์ที่อยู่ในวงเครือข่าย ไม่ให้ซ้ำ กัน ดูแลจัดการเรื่อง การจัดการเส้นทาง ซึ่งเทียบได้กับ Full Functionality Device: FFD

2. End-Device เป็นอุปกรณ์ปลายทางสุด ซึ่งจะรับใช้สัญญาณจากเซ็นเซอร์ที่ปลายทาง โดยที่ใช้พลังงานต่ำในการทำงานเทียบได้กับ Reduced Functionality Device : RFD หรือ Full Functionality Device : FFD บางกรณีขึ้นอยู่กับเซ็นเซอร์ที่ใช้

3. Router มีหน้าที่รับส่งข้อมูล ในเส้นทางต่างๆของเครือข่ายและหาเส้นทางที่ใกล้ที่สุด ซึ่งจะต้องใช้เวลาน้อยที่สุด

คุณสมบัติโดยทั่วไป

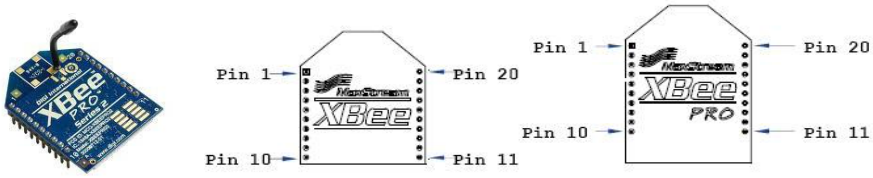
- ความถี่ในการทำงาน: 2.4 GHz
- สายอากาศ: มีสายอากาศแบบ Whip
- ระยะทำการในร่ม: สูงสุด 300 ฟุต หรือประมาณ 100 เมตร
- ระยะทำการกลางแจ้ง (แบบ Line-of-Sight) : สูงสุดถึง 1 ไมล์ หรือประมาณ 1,500 เมตร

- กำลังส่ง: 60 mW (18dBm)
- ความไวในการรับส่งสัญญาณ: -100dBm (1% Packet Error Rate)
- การทำงานของขาพอร์ต: สามารถกำหนดผ่านทางซอฟต์แวร์ X-CTU เพื่อให้ทำงานเป็น
- อินพุตอนาล็อกสำหรับวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล ความละเอียด 10 บิต
- อินพุตเอาต์พุตดิจิตอล
- ขนาด: 0.96 x 1.297 นิ้ว หรือ 2.438 x 3.294 เซนติเมตร
- ไฟเลี้ยง: 2.8 ถึง 3.3 V
- กระแสไฟฟ้า: เมื่อส่งข้อมูล 215mA รับข้อมูล 55mA น้อยกว่า 10μA ในโหมดลดพลังงาน ที่ไฟเลี้ยง +3.3 V
- อุณหภูมิใช้งาน: -40 ถึง 85 ºC

คุณสมบัติด้านการสื่อสารข้อมูล

- สามารถทำงานเป็นอุปกรณ์มาสเตอร์และสเลฟได้
- อัตราถ่ายทอดข้อมูลผ่านคลื่นวิทยุ: 250,000 บิตต่อวินาที
- อัตราการถ่ายทอดข้อมูลอนุกรม (บอดเรต) : 1,200 ถึง 115,200 บิตต่อวินาที
- รูปแบบโครงข่ายข้อมูลที่รองรับ: จุดต่อจุด (Point-to-Point) จุดต่อหลายจุด (Point-to-Multipoint) และเข้ากันได้กับอุปกรณ์ตามมาตรฐานฐาน IEEE 802.15.4
- ทางเลือกแอดเดรส: PAN ID ช่อง (Channel) และแอดเดรส (Addresses) สำหรับแอดเดรส สามารถกำหนดรหัสแอดเดรสได้มากถึง 65,000 รหัส
- เทคโนโลยีในการกระจายคลื่น : DSSS (Direct Sequence Spread Spectrum)
- รองรับการทำงานทั้งแบบ API และ AT Command สามารถกำหนดได้ผ่านทางซอร์ฟแวร์ X-CTU
- การรับรองมาตรฐาน : สหรัฐอเมริกา (FCC Part 15.247) OUR-XBEEPRO
    - แคนาดา (IC) 4214A XBEEPRO
    - ยุโรป (CE) ETSI (ที่กำลังส่งสูงสุด 10 dBm)
    - ญี่ปุ่น 005NYCA0378 (ที่กำลังส่งสูงสุด 10 dBm)

การจัดขาโมดูล XBee และฟังก์ชันในการทำงาน



รูปที่ 2.21 XBee และ การจัดขาของ XBee

ตารางที่ 2.5 การจัดขาโมดูล XBee และฟังก์ชันในการทำงาน

ขาที่	ชื่อขา/การทำงาน
1	Vcc ขาต่อไฟเลี้ยง +3.3V
2	DOUT : ขาเอาต์พุตส่งข้อมูลอนุกรม
3	DIN : ขาอินพุตรับข้อมูลอนุกรม
4	D08 : ขาเอาต์พุตดิจิตอล ช่อ 8
5	RESET : ขารีเซตหลัก (Active “0”)
6	PWM0/RSSI : ขาเอาต์พุต PWM ช่อง 0 และขาเอาต์พุตแสดงความแรงของการรับสัญญาณ
7	PWM1 : ขาเอาต์พุต PWM ช่อง 1
8	ไมใช้งาน
9	DTR/SLEEP_RQ/DI8 : ขาอินพุตรับสัญญาณให้หยุดทำงานเข้าโหมดสลีป หรือ เป็นขาอินพุตดิจิตอลช่อง 8
10	GND : ขาต่อกราวด์
11	AD4/DIO4 : ขาอินพุตอนาล็อก 4 หรือขาอินพุตเอาต์พุตดิจิตอล 4
12	CTS/DIO7 : อินพุตรับสัญญาณแจ้งการส่งข้อมูลจากโฮสต์ (Clear-To-Send) ใช้ในการควบคุมจังหวะการรับส่งข้อมูล หรือเป็นขาอินพุตเอาต์พุตดิจิตอล 7
13	ON/SLEEP : ขาแสดงสถานการณ์ทำงาน “1” : อยูในโหมดทำงานปกติ, “2” : อยูในโหมดสลีป
14	VREF : ขาต่อแรงดันอ้างอิงสำหรับโมดูลแปลงสัญญาณอนาล็อกเป็นดิจิตอลภายใน Xbee
15	Associated/AD5/DIO5 : ขาแสดงสถานการณ์เชื่อมต่อ หรือขาอินพุตอนาล็อก 5 หรือ ขาอินพุตเอาต์พุตดิจิตอล 5

ตารางที่ 2.5 การจัดขาโมดูล XBee และฟังก์ชันในการทำงาน(ต่อ)

16	RTS/AD6/DIO6 : ขาเอาต์พุตแจ้งความพร้อมในการส่งข้อมูล (Ready-To-Send) ใช้ควบคุมจังหวะการรับส่งข้อมูล หรือเป็นขาอินพุตอนาล็อก 6 หรือเป็นขาอินพุตดิจิตอล 6
17	AD3/DIO3 : ขาอินพุตอนาล็อก 3 หรือ ขาอินพุตเอาต์พุตดิจิตอล 3
18	AD3/DIO2 : ขาอินพุตอนาล็อก 2 หรือ ขาอินพุตเอาต์พุตดิจิตอล 2
19	AD3/DIO1 : ขาอินพุตอนาล็อก 1 หรือ ขาอินพุตเอาต์พุตดิจิตอล 1
20	ADO/DIO0 : ขาอินพุตอนาล็อก 0 หรือ ขาอินพุตเอาต์พุตดิจิตอล 0

การเขียนโปรแกรมควบคุมระบบการทำงานของฮาร์ดแวร์และการสร้างอุปกรณ์ฮาร์ดแวร์

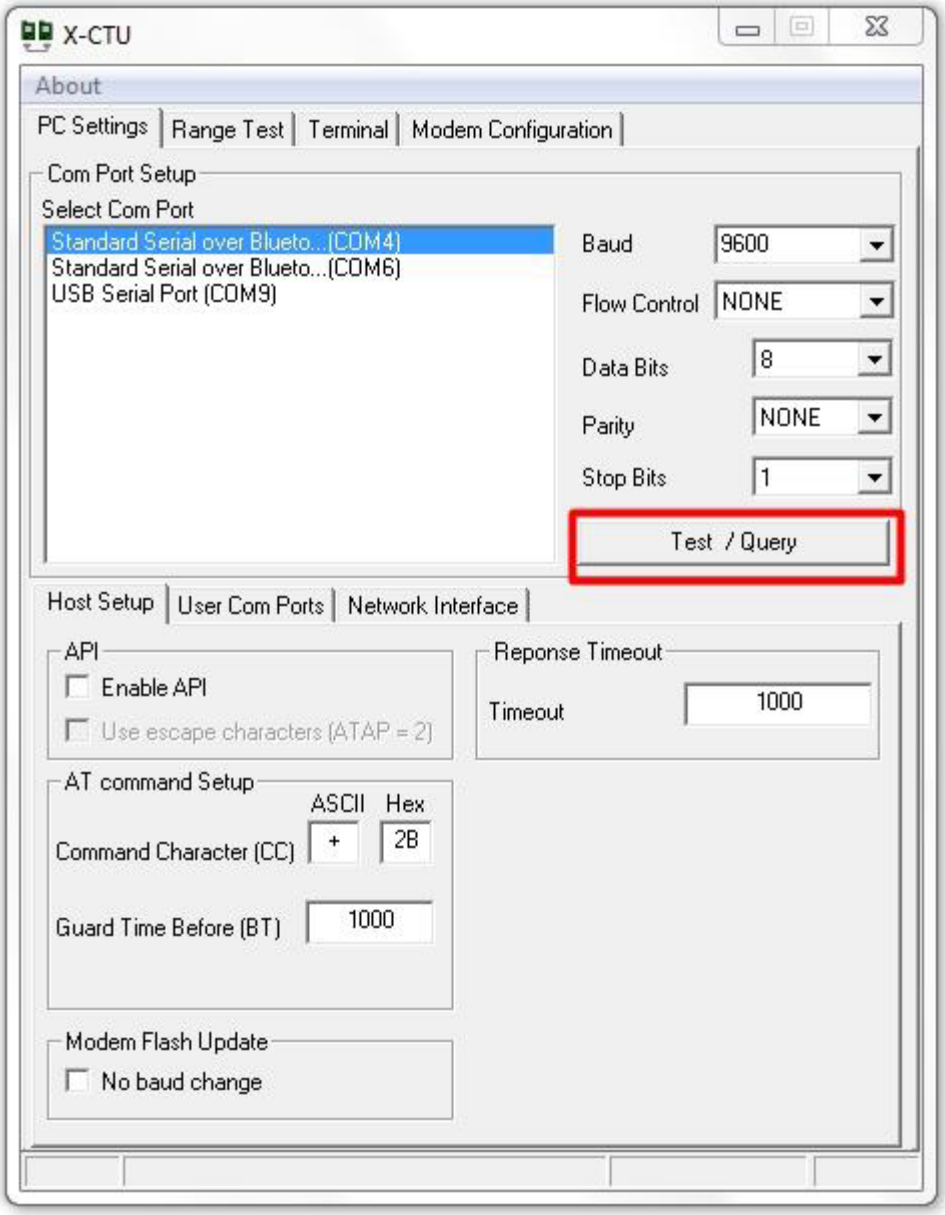
ในการเขียนโปรแกรมนั้น ในภาครับและภาคส่ง เราใช้ XBee ในการส่งข้อมูลหากัน ดังนั้นเราจะต้องทำให้ XBee ภาครับและภาคส่งสามารถติดต่อกันได้ โดยใช้โปรแกรม X-CTU ทำได้ดังนี้

1. ติดตั้งโปรแกรม X-CTU เวอร์ชันล่าสุดลงในคอมพิวเตอร์

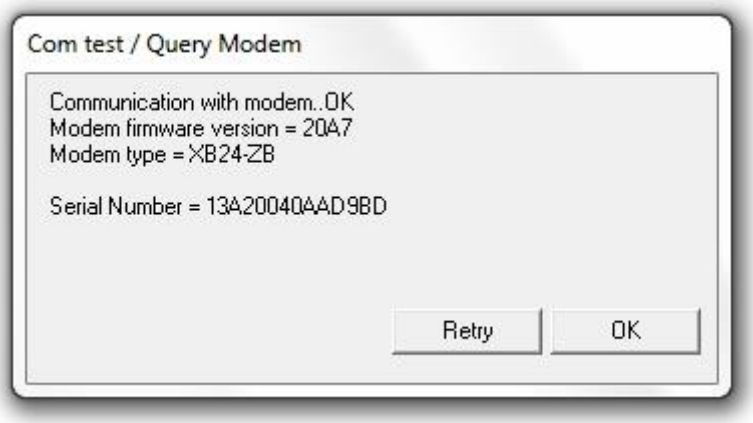
2. ติดตั้งโมดูล XBee ลงบนบอร์ด XBee USB Adapter ดังรูปที่ 2.21

รูปที่ 2.22 การติดตั้งโมดูล XBee ลงบนบอร์ด XBee USB Adapter

3. เปิดโปรแกรม X-CTU และกดปุ่ม Test เพื่อทดสอบการเชื่อมต่อระหว่าง XBee กับโปรแกรม X-CTU ว่าสามารถติดต่อกันได้หรือไม่ ดังรูปที่ 2.23 หากติดต่อกันได้จะปรากฏหน้าต่างแจ้งผลการติดต่อและข้อมูลของ XBee เบื้องต้นดังรูปที่ 2.24

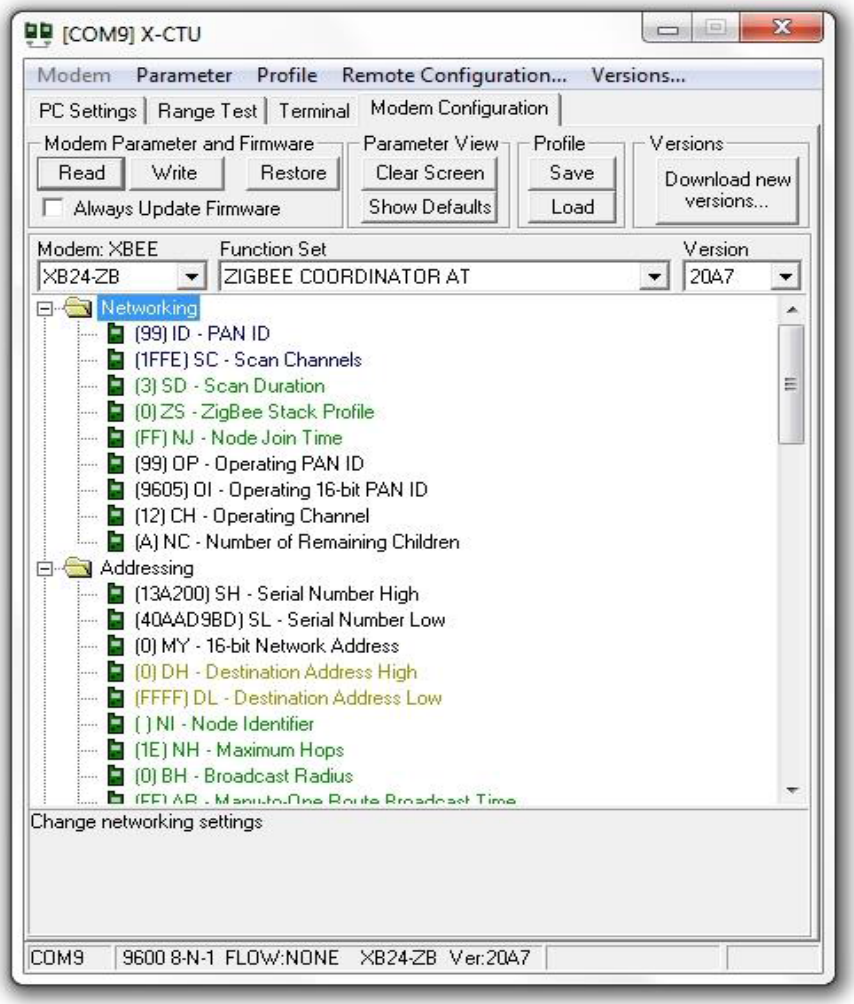


รูปที่ 2.23 หน้าต่างโปรแกรม X-CTU



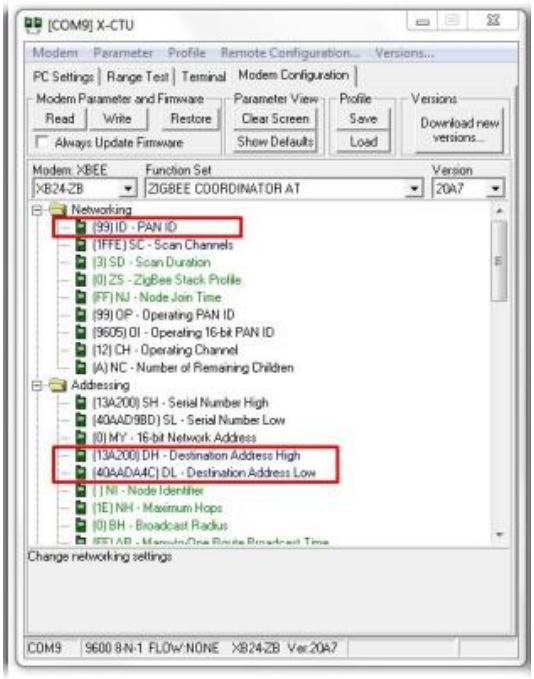
รูปที่ 2.24 หน้าต่างแจ้งผลการติดต่อ

4. หลังจากนั้นคลิกไปที่แท็ป Modem Configuration แล้วกดปุ่ม Read ในกรอบ Modem Parameter and Firmware จะปรากฏข้อมูลชื่อรุ่นของโมดูล XBee (Modem Type) ชื่อฟังก์ชันหมายเลขเวอร์ชันของเฟิร์มแวร์ และค่าพารามิเตอร์ต่างๆ ดังรูปที่ 2.25

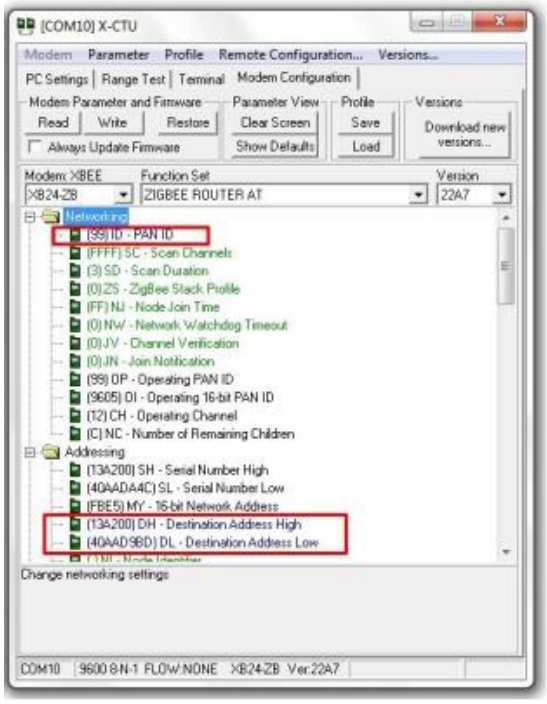


รูปที่ 2.25 ผลการอ่านค่าจากโมดูล XBee ก่อนการตั้งค่า

5. ตั้งค่าการรับส่งข้อมูลให้กับ XBee โดยทำ การตั้งค่าให้ 1 ตัวเป็นโหมด Coordinator (Rx) และ อีก 1 ตัวเป็นโหมด Router (Tx) โดยมีการเชื่อมต่อกันแบบเมช

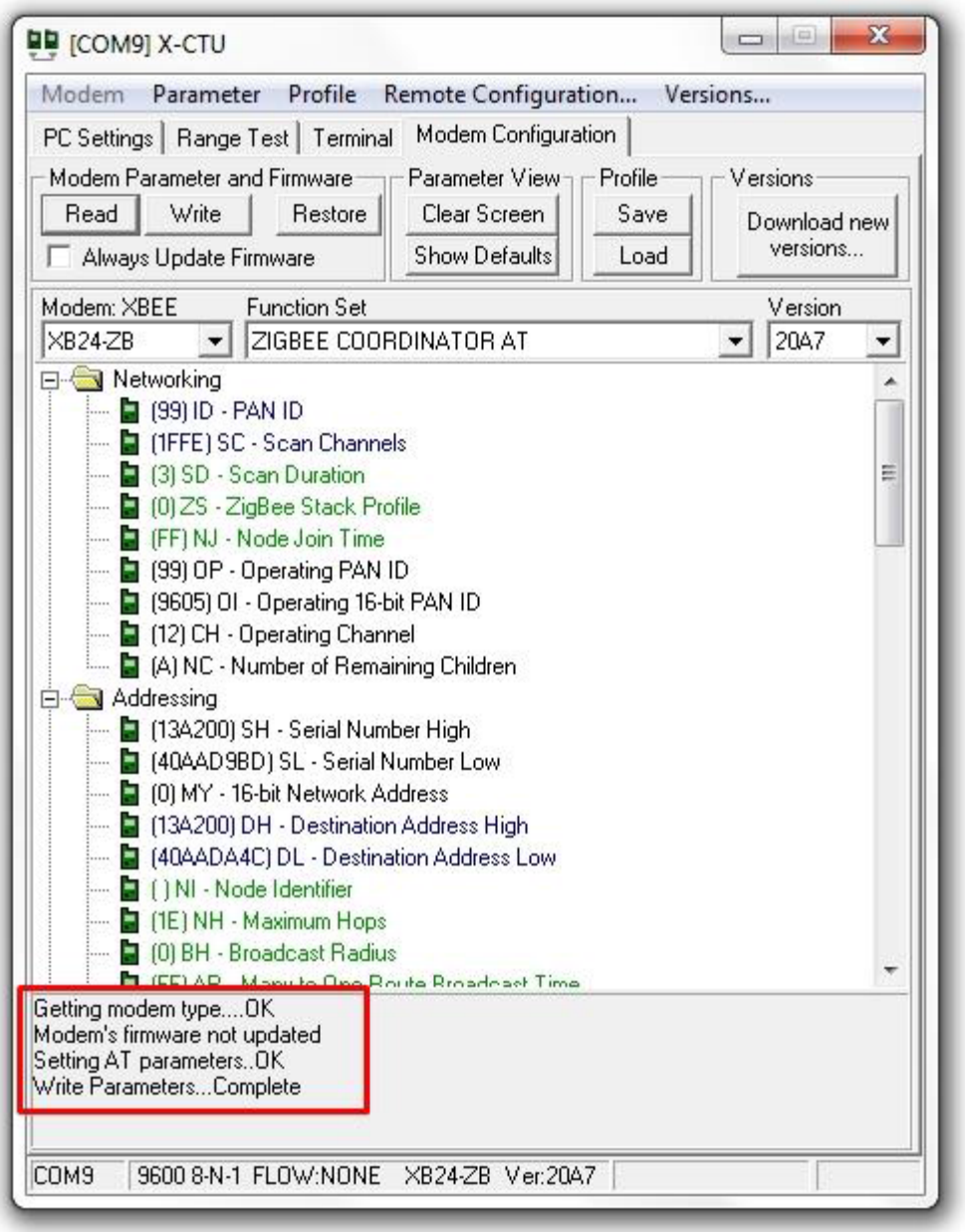


รูปที่ 2.26 การตั้งค่า XBee แบบ Coordinator



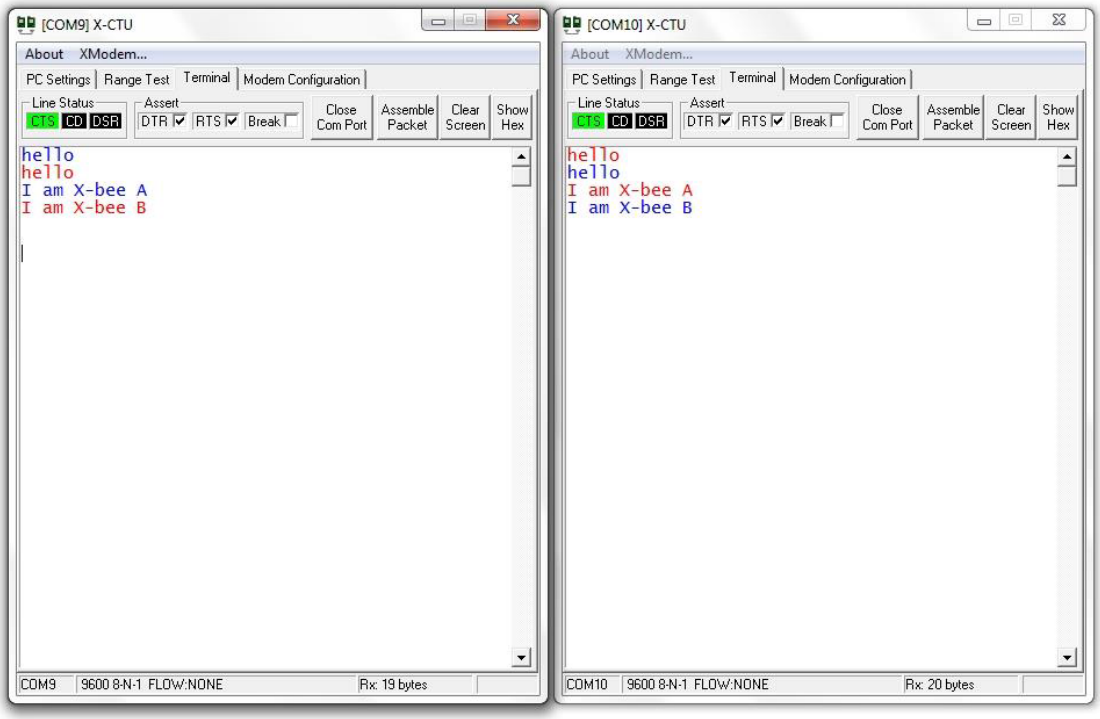
รูปที่ 2.27 การตั้งค่า Xbee แบบ Router

6. จากนั้นกดปุ่ม Write รอสักครู่ สังเกตด้านล่างของหน้าต่าง Modem Configuration จะแสดงข้อความเพื่อแจ้งสถานการณ์ทำงาน หากเป็นดังรูปที่ 2.28 แสดงว่าการกำหนดค่าเสร็จสมบูรณ์



รูปที่ 2.28 แสดงข้อความแจ้งสถานการณ์ทำงาน

7. เราสามารทดสอบว่า XBee สามารถรับส่งข้อมูลหากันได้หรือไม่ โดยคลิกไปที่แท็ป Terminal ของทั้งสองตัว แล้วพิมพ์ข้อความ หากรับส่งข้อมูลกันได้ จะปรากฏดังรูปที่ 2.29 โดยที่ข้อความที่ส่งจะเป็นสีแดง และข้อความที่ได้รับเป็นสีน้ำเงิน



รูปที่ 2.29 XBee มีการรับส่งข้อมูลผ่านโปรแกรม X-CTU

2.3 สรุป

เซ็นเซอร์ไร้สายสำหรับนับจำนวนคนภายในอาคารด้วย Infrared Sensor และ Smart Floors นั้น มีองค์ประกอบสำคัญ คือ เซ็นเซอร์อินฟราเรด (Infrared Sensor) โดยอาศัยคลื่นแสงอินฟราเรดมาใช้ทำการตรวจจับวัตถุ ซึ่งประกอบด้วย ตัวส่งคือไดโอดเปล่งแสง (LED) และตัวรับคือ Phototransistors

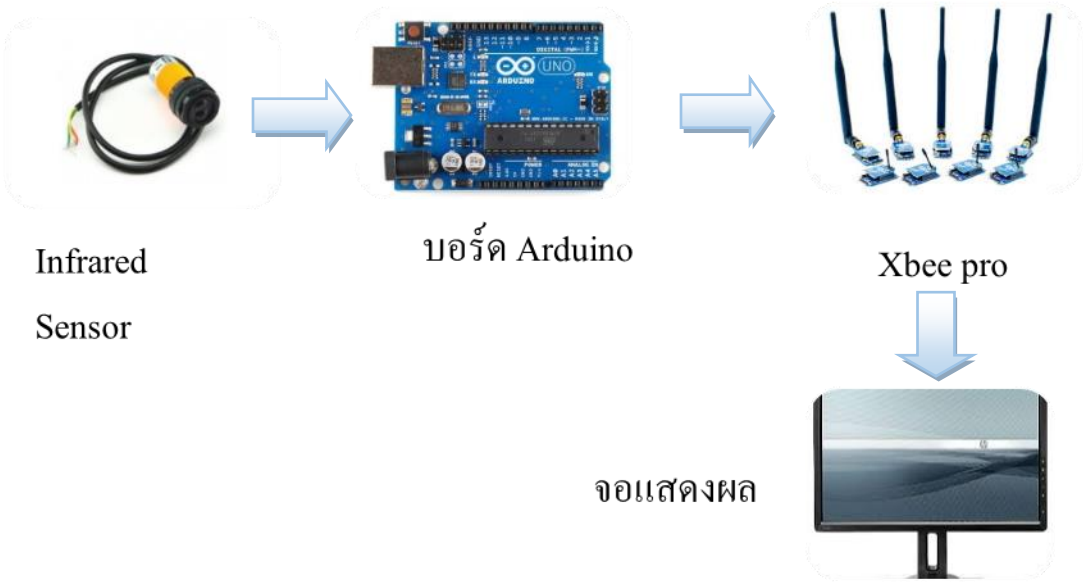
บทที่ 3
การออกแบบโครงงาน

3.1 บทนำ

เนื้อหาในบทนี้จะเป็นการกล่าวถึงการออกแบบโครงสร้างและการสร้างวงจรภายในอุปกรณ์สำหรับการนับจำนวนคนด้วย Infrared Sensor สำหรับนำมาตรวจจับบุคคลเข้าออกอาคาร แล้วทำการทดสอบการใช้งานได้จริงของอุปกรณ์แต่ละตัวที่กลุ่มผู้ทดสอบได้นำมาประกอบเป็นอุปกรณ์สำหรับการนับจำนวนคนด้วย Infrared Sensor มาเป็นวงจรภายในวงจร 3 ส่วนดังนี้ Infrared Sensor ส่งสัญญาณดิจิตอลมายังบอร์ด Arduino บอร์ด Arduino แปลงค่าสัญญาณตามโค้ดที่เขียนไว้แล้วทำการส่งค่าไปที่ Xbee ภาคส่ง ส่งค่าไปที่ Xbee ภาครับเพื่อทำการประมวลผลโดยโปรแกรม Processing 2.2.1 และทำการแสดงผลที่ได้ออกทางหน้าจอ

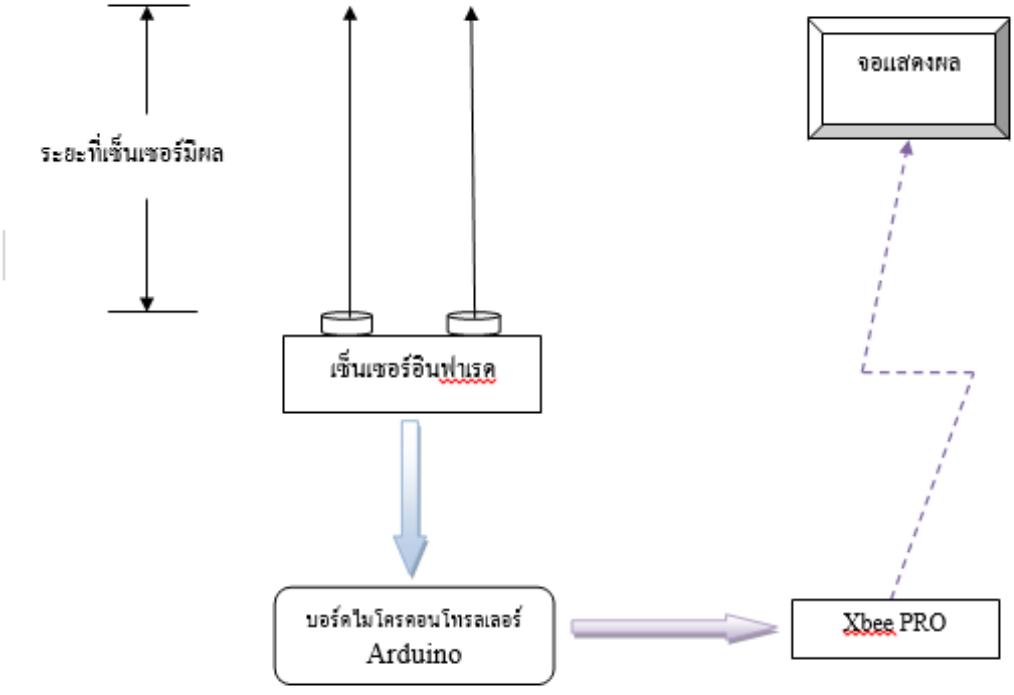
3.2 องค์ประกอบและหลักการทำงานของเครื่องนับจำนวนคน

หลักการของส่วนต่างๆของเครื่องนับคนเขาออกซึ่งอาศัยเซ็นเซอร์ 2 ตัว ตั้งชื่อเป็น Sensor1 และSensor2 วางเรียงกันซึ่งอาศัยหลักการสะท้อนเมื่อมีวัตถุต่างๆมาตัดขว้าง ถ้ามีการเดินเข้ามาทำให้ Sensor1 ถูกตัดก่อนหรือเกิดการสะท้อนจากวัตถุก่อน Sensor2 จะทำให้โปรแกรมที่เขียนมาให้นับเป็นหนึ่งคนเข้าเรื่อยๆตามหลักการเดิมถ้ามีการตัดอีก แต่ถ้า Sensor2 ถูกตัดก่อนหรือเกิดการสะท้อนจากวัตถุก่อน Sensor1 จะทำให้โปรแกรมที่เขียนมาให้นับเป็นหนึ่งคนออกเรื่อยๆตามหลักการเดิมถ้ามีการตัดซ้ำอีกแล้วจะทำการลบกันเพื่อหาจำนวนคนที่เหลืออยู่ในสถานที่นั้นๆและส่งผลที่ได้มายังสถานีฐานผ่านระบบ Xbee ที่ย่าน ความถี่ 2.4 GHz และแสดงผลด้วย Real-time plot และ Animation ซึ่งสร้างด้วยโปรแกรม Processing 2.2.1



รูปที่ 3.1 ขั้นตอนการทำงานของวงจรทั้งหมด

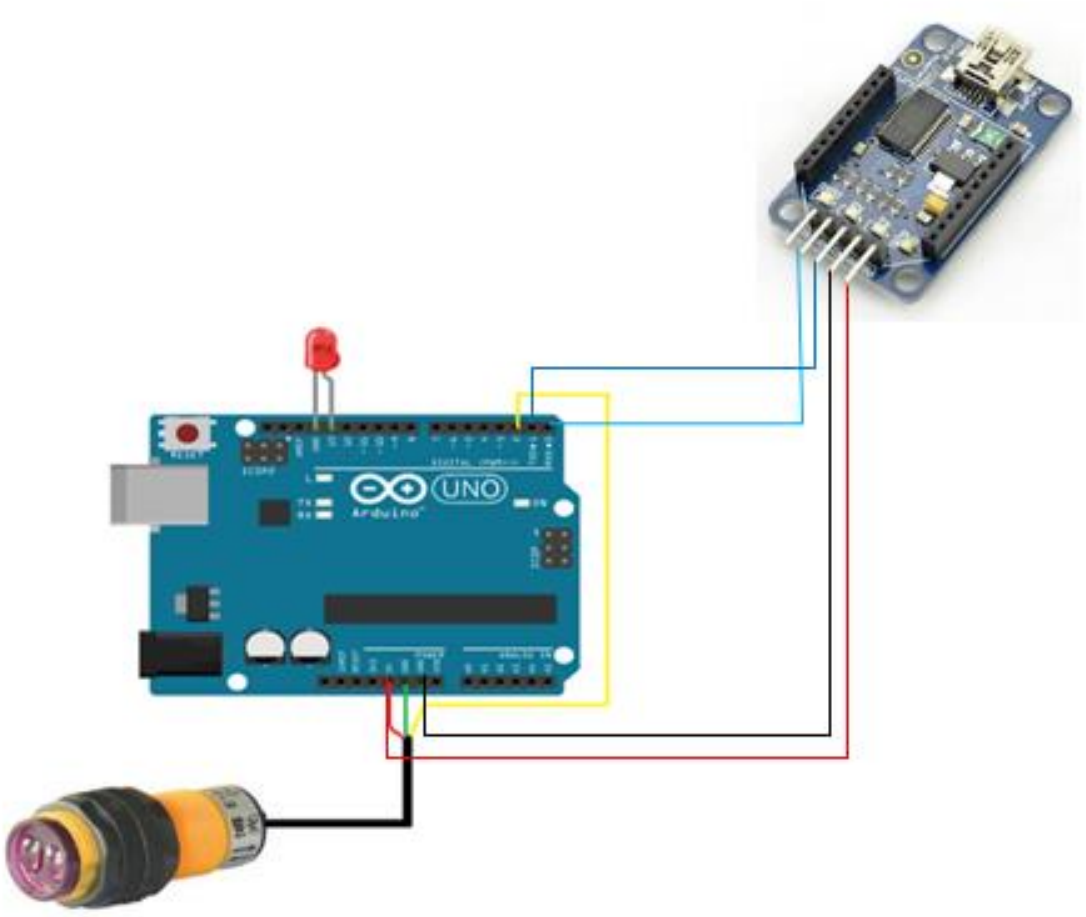
3.3 วงจรควบคุมการทำงานทั้งหมด



รูปที่ 3.2 แผนไดอะแกรมของวงจรควบคุมการทำงานต่างๆ

วงจรควบคุมการนับจำนวนคนดังกล่าวเป็นวงจรที่ตั้งอยู่บริเวรประตูทางเข้าออกทางเดียว ทำหน้าที่นับจำนวนคนเข้าออก โดยวงจรจะมีการตอบสนองเมื่อมีคนเดินผ่านหรือวัตถุตัดผ่านในระยะหวังผลของเซ็นเซอร์ เมื่อมีการตัดโปรแกรม Processing จะทำหน้าที่ประมวลผลและทำการแสดงค่าที่ได้ออกทางจอแสดงผลดังรูปที่ 3.2

3.3.1 การเชื่อมต่ออุปกรณ์ต่างๆภายในวงจร



รูปที่ 3.3 การเชื่อมต่ออุปกรณ์ต่างๆภายในวงจร

การเชื่อมต่อสาย XBee

- เชื่อมต่อสาย Power (สีแดง) เข้ากับขาไฟเลี้ยง 5 V ของบอร์ดไมโครคอนโทรลเลอร์
- เชื่อมต่อสาย GND (สีดำ) เข้ากับขา GND ของบอร์ดไมโครคอนโทรลเลอร์
- เชื่อมต่อสาย Tx (สีน้ำเงิน) เข้ากับ ขาดิจิตอลหมายเลข 0 ของบอร์ดไมโครคอนโทรลเลอร์
- เชื่อมต่อสาย Rx (สีน้ำเงินอ่อน) เข้ากับ ขาดิจิตอลหมายเลข 1 ของบอร์ดไมโครคอนโทรลเลอร์ที่ เซ็นเซอร์อินฟราเรด

การเชื่อมต่อสายเซ็นเซอร์อินฟราเรด

- เชื่อมต่อสาย Vcc (สีแดง) ของเซ็นเซอร์ต่อกับขาแรงดัน 5 V ของบอร์ดไมโครคอนโทรลเลอร์
- เชื่อมต่อสาย GND (สีเขียว) ของเซ็นเซอร์ ต่อกับ ขา GND ของบอร์ดไมโครคอนโทรลเลอร์
- เชื่อมต่อสาย Signal (สีเหลือง) ของเซ็นเซอร์ เข้ากับ ขาดิจิตอลหมายเลย 2 ของบอร์ดไมโครคอนโทรลเลอร์

3.3.2 โค้ดโปแกรมสำหรับควบคุมการทำงาน

```
const int sensors1=10;
const int sensors2=9;
// the setup function runs once when you press reset or power the board
void setup() {
 pinMode(sensors1,INPUT);
 pinMode(sensors2,INPUT);
 Serial.begin(9600);
}
// the loop function runs over and over again forever
void loop() {
 int state1=digitalRead(sensors1);
int state2=digitalRead(sensors2);
if(state1==HIGH){
```

```
Serial.print("0");                ; ถ้าเซ็นเซอร์ตัวที่ 1 มีค่าเป็น HIGH ให้แสดงค่า “0”
}else{
Serial.print("500");              ; ถ้าเซ็นเซอร์ตัวที่ 1 มีค่าเป็น LOW ให้แสดงค่า “500”
if(state2==LOW){
Serial.print("0"); }              ; ถ้าเซ็นเซอร์ตัวที่ 2 มีค่าเป็น LOW ให้แสดงค่า “0”
delay(1500);                      ; ดีเลย์ 1500 ms

 }
Serial.print(",");
 if(state2==HIGH){
Serial.print("0");                ; ถ้าเซ็นเซอร์ตัวที่ 2 มีค่าเป็น LOW ให้แสดงค่า “0”
}else{
Serial.print("500");              ; ถ้าเซ็นเซอร์ตัวที่ 2 มีค่าเป็น LOW ให้แสดงค่า “500”
state1=0;                         ; ให้เซ็นเซอร์ตัวที่ 1 มีค่าเท่ากับ “0”
delay(1500);                      ; ดีเลย์ 1500 ms
}
Serial.println();
delay(10);                        ; ดีเลย์ 10 ms
}
```

3.3.3 โค้ดโปรแกรมสำหรับการแสดงผลโดย Processing

```
import processing.serial.* ;
import gifAnimation.*;     // import the gifAnimation library
Serial myport ;
Gif myAnimation;           // create Gif object called myAnimation
int val, screen_increment, old_x = 0, old_y = 0, old_x1 = 0, old_y1 = 0;
String inString ;
```

```
int lf = 10 ; //Linefeed in ASCII
int numSensors = 2; //we will be expecting data feed from 2 sensors
int sensors[]; //array to read the 2 values
int pSensors[]; //array to store previous readings used for comparing actual reading with
last one
int time = 0, t0=0 , t1=0, flag0 = 0, flag1 = 0, In = 0, Out = 0 ,total=0 ;
int val0_old = 600 , slope0, val1_old = 600, slope1, peaks0 = 0, peaks1 = 0 ;
int screenstep = 3 ;
PImage img;
void setup()
{
size(displayWidth-100, 800);
 myAnimation = new Gif(this, "Gif_in_out_2.gif"); // load animated GIF file from
                                                  // the data folder, replace
                                                  // "img01.gif" with the name of
                                                  // your GIF file
 myAnimation.play();                  // play the animated GIF
println(Serial.list());
String portName = Serial.list()[0];
myport = new Serial(this, portName, 9600);
myport.bufferUntil(lf); //read bytes until you get linefeed (ASCII 10)
background(288,24,24);
strokeWeight(1) ;
stroke(255,255,255);
```

```
line(100,10,100,600)  ;

textSize(30) ;

text("ADC values from Infrared sensor", 400, 670);

textSize(15) ;

text("by School of Telecommunication  Engineering,  SUT", displayWidth-480,  690);

textSize(20) ;

text(0, 40, 600); line(90,600,  100,600) ;

text(1, 40, 525); line(90,525,  100,525) ;

text(200, 40, 400); line(90,400,  100,400) ;

text(300, 40, 300); line(90,300,  100,300) ;

text(400, 40, 200); line(90,200,  100,200) ;

text(500, 40, 100); line(90,100,  100,100) ;

text(600, 40, 20); line(90,5,  100,5) ;

line(100,600,displayWidth-100,600)  ;

text(100, 100, 630); line(100,600,  100,610) ;

text(200, 200, 630); line(200,600,  200,610) ;

text(300, 300, 630); line(300,600,  300,610) ;

text(400, 400, 630); line(400,600,  400,610) ;

text(500, 500, 630); line(500,600,  500,610) ;

text(600, 600, 630); line(600,600,  600,610) ;

text(700, 700, 630); line(700,600,  700,610) ;
```

```
text(800, 800, 630); line(800,600,  800,610) ;

text(900, 900, 630); line(900,600,  900,610) ;

text(1000, 1000, 630); line(1000,600,  1000,610) ;

text(1100, 1100, 630); line(1100,600,  1100,610) ;

text(1200, 1200, 630); line(1200,600,  1200,610) ;

text(1300, 1300, 630); line(1300,600,  1300,610) ;

text(1400, 1400, 630); line(1400,600,  1400,610) ;

text(1500, 1500, 630); line(1500,600,  1500,610) ;

text(1600, 1600, 630); line(1600,600,  1600,610) ;

}

void draw()

{

{

int s = second();

int m = minute();

int h = hour();

textSize(25);

fill(55);

text(h+":"+m+":"+s, 220, 50);}

if((pSensors != null)&&(sensors  != null))  {

}
```

```
}

void serialEvent(Serial  myport)

{

 inString = myport.readString() ;

if (inString !=null)

{

inString = trim(inString) ;

pSensors = sensors ;

sensors = int(split(inString,','));

}

for (int sensorNum = 0; sensorNum < sensors.length;  sensorNum++) {

print("Sensor " + sensorNum + ": " + sensors[sensorNum] + "\t");

}

println();

val = int(inString) ;

int val0 = sensors[0],val1  = sensors[1];

time = time + 1 ;

strokeWeight(5) ;

stroke(255,255,255);

color c = color(255, 204, 0);  // Define color 'c'

fill(50); //textSize(50) ;
```

```
textSize(20) ;

image(myAnimation, 200, 10);

fill(50) ;

textSize(50) ;

text(In, 280, 270) ;

fill(255);

text(Out, 980, 270) ;

int visitors = total ;

fill(288,24,24);

text(visitors, 530, 170) ;

strokeWeight(5) ;

stroke(255,255,255);

line(old_x, old_y, screen_increment, 600-val0) ;

old_x = screen_increment ;

old_y = 600 - val0;

 screen_increment = screen_increment+10 ;

stroke(0,0,0);

line(old_x1, old_y1, screen_increment, 600-val1) ;

old_x1 = screen_increment ;

old_y1 = 600 - val1;

if (val0 > val1){
```

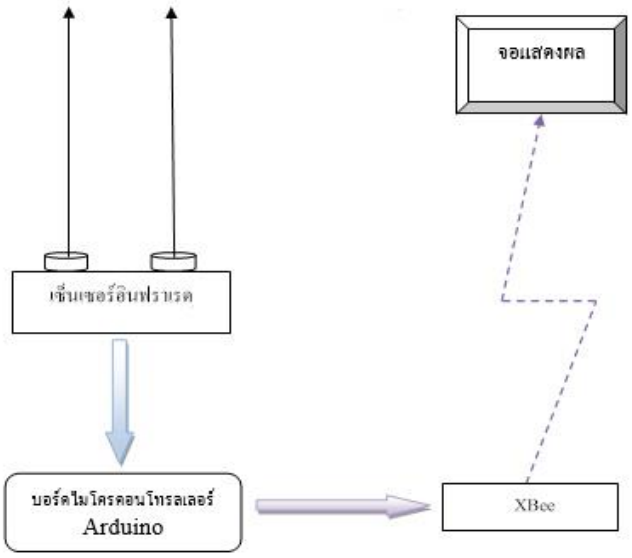
```
In = In + 1 ;

}

if( val1 > val0 ) {

Out = Out + 1 ;

}

total=In - Out;

screen_increment = screen_increment+ 5 ;

textSize(20) ;

if (screen_increment > displayWidth-100)

{

pause();

background(288,24,24);

strokeWeight(1) ;

stroke(255,255,255);

line(100,10,100,600) ;

fill(255,255,255);

textSize(20) ;

text(0, 40, 600); line(90,600, 100,600) ;

text(100, 40, 500); line(90,500, 100,500) ;

text(200, 40, 400); line(90,400, 100,400) ;

text(300, 40, 300); line(90,300, 100,300) ;
```

```
text(400, 40, 200); line(90,200,  100,200) ;

text(500, 40, 100); line(90,100,  100,100) ;

text(600, 40, 10); line(90,5,  100,5) ;

line(100,600,displayWidth-100,600)  ;

text(100, 100, 630); line(100,600,  100,610) ;

text(200, 200, 630); line(200,600,  200,610) ;

text(300, 300, 630); line(300,600,  300,610) ;

text(400, 400, 630); line(400,600,  400,610) ;

text(500, 500, 630); line(500,600,  500,610) ;

text(600, 600, 630); line(600,600,  600,610) ;

text(700, 700, 630); line(700,600,  700,610) ;

text(800, 800, 630); line(800,600,  800,610) ;

text(900, 900, 630); line(900,600,  900,610) ;

text(1000, 1000, 630); line(1000,600,  1000,610) ;

text(1100, 1100, 630); line(1100,600,  1100,610) ;

text(1200, 1200, 630); line(1200,600,  1200,610) ;

text(1300, 1300, 630); line(1300,600,  1300,610) ;

text(1400, 1400, 630); line(1400,600,  1400,610) ;

text(1500, 1500, 630); line(1500,600,  1500,610) ;

text(1600, 1600, 630); line(1600,600,  1600,610) ;

textSize(30) ;
```

```
text("ADC values from Infrared sensors", 300, 30);

 textSize(15) ;

text("by School of Telecommunication  Engineering,  SUT", displayWidth-500,  560);

textSize(20) ;

strokeWeight(5) ;

creen_increment = -50 ;

old_x = -50 ;

old_y = 0 ;

old_x1 = -50 ;

old_y1 = 0 ;

}

}
```

3.3.6 แผนไดอะแกรมสำหรับแสดงการทดสอบสำหรับ Infrared Sensor

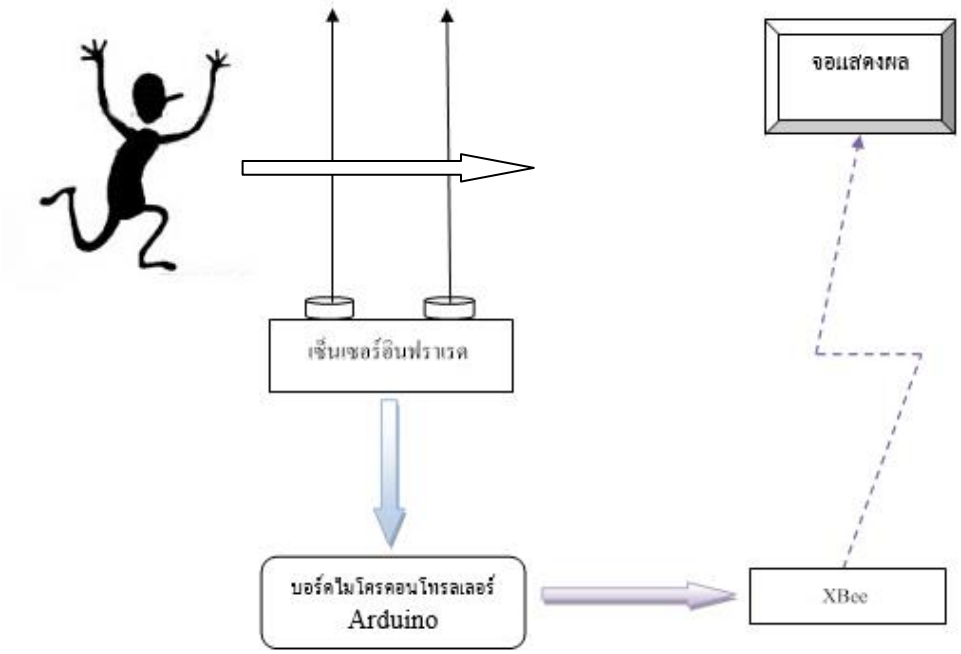
3.3.6.1 แผนไดอะแกรมแสดงการทำงานขณะไม่มีคนเดินเข้าภายในอาคาร สำหรับ Infrared Sensor



รูปที่ 3.4 แผนไดอะแกรมแสดงการทำงานขณะไม่มีคนเดินเข้าภายในอาคาร สำหรับ Infrared Sensor

รูปที่ 3.5 การแสดงผลแบบ Animation ขณะไม่มีคนเดินเข้าภายในอาคารสำหรับ Infrared Sensor

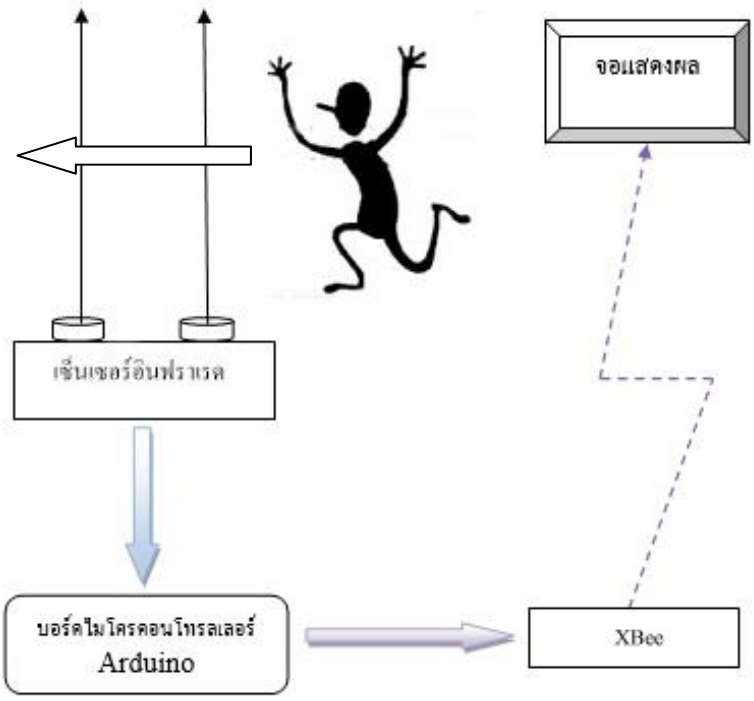
3.3.6.2 แผนไดอะแกรมแสดงการเดินเข้าภายในอาคาร



รูปที่ 3.6 แผนไดอะแกรมแสดงการเดินเข้าภายในอาคารสำหรับ Infrared Sensor

รูปที่ 3.7 การแสดงผลแบบ Animation ขณะเดินเข้าในอาคารสำหรับ Infrared Sensor

3.3.6.3 แผนไดอะแกรมแสดงการเดินออกนอกอาคารสำหรับ Infrared Sensor

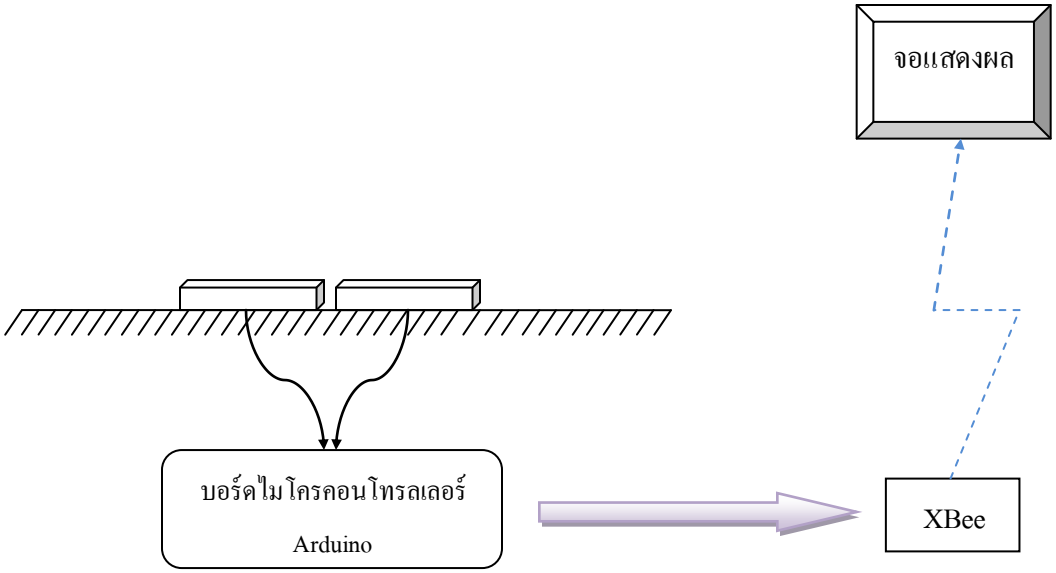


รูปที่ 3.8 แผนไดอะแกรมแสดงการเดินออกนอกอาคารสำหรับ Infrared Sensor

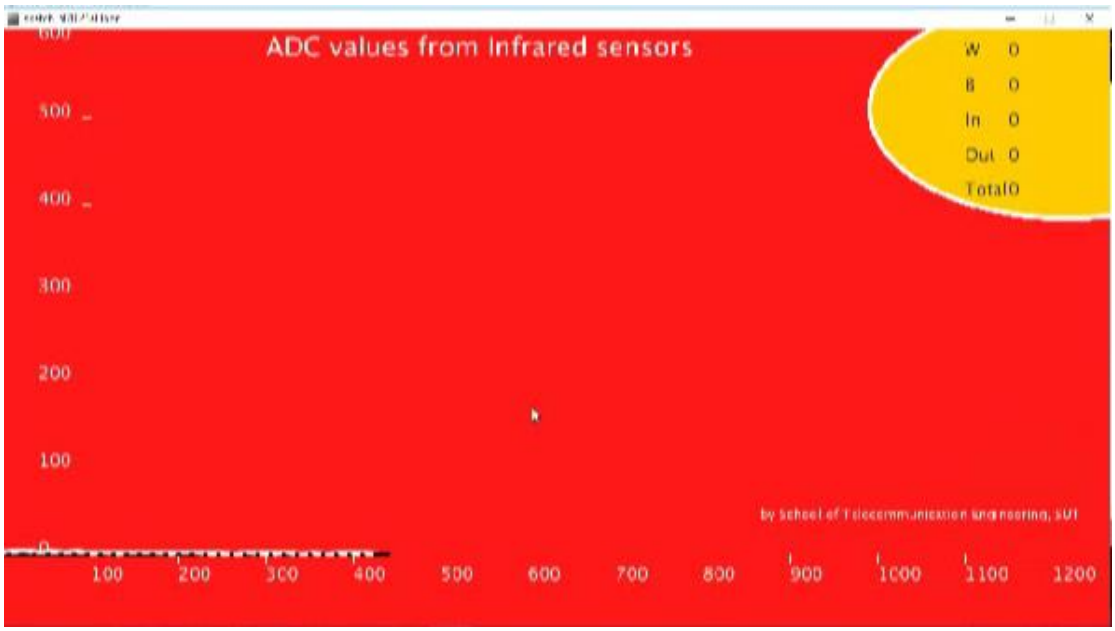
รูปที่ 3.9 การแสดงผลแบบ Animation ขณะเดินออกนอกอาคารสำหรับ Infrared Sensor

3.3.7 แผนไดอะแกรมสำหรับแสดงการทดสอบสำหรับ Smart Floors

3.3.7.1 แผนไดอะแกรมแสดงการทำงานขณะไม่มีคนเดินเข้าภายในอาคาร สำหรับ Smart Floors

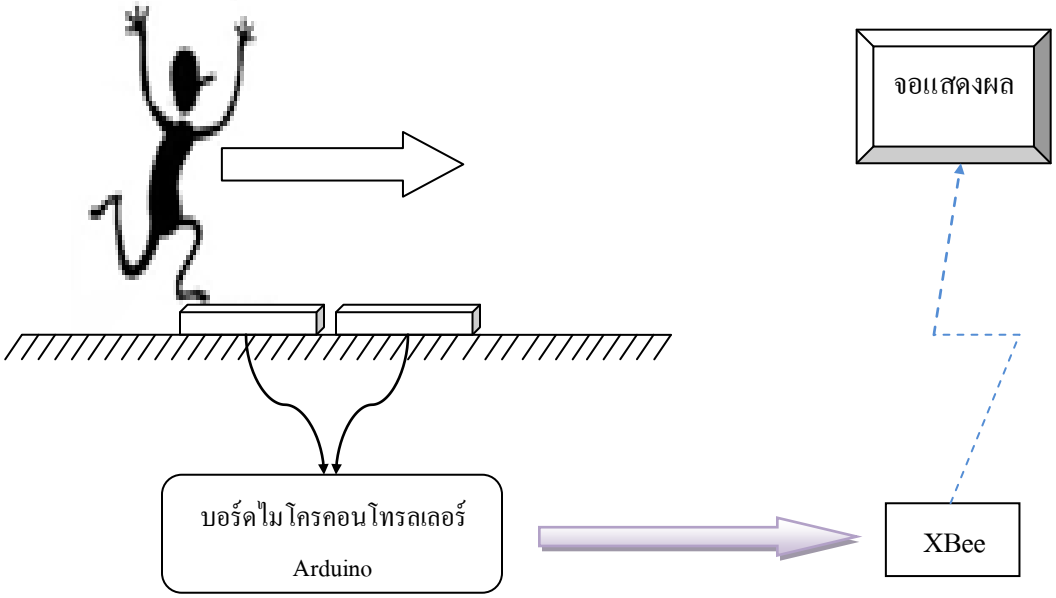


รูปที่ 3.10 แผนไดอะแกรมแสดงการทำงานขณะไม่มีคนเดินเข้าภายในอาคารสำหรับ Smart Floors

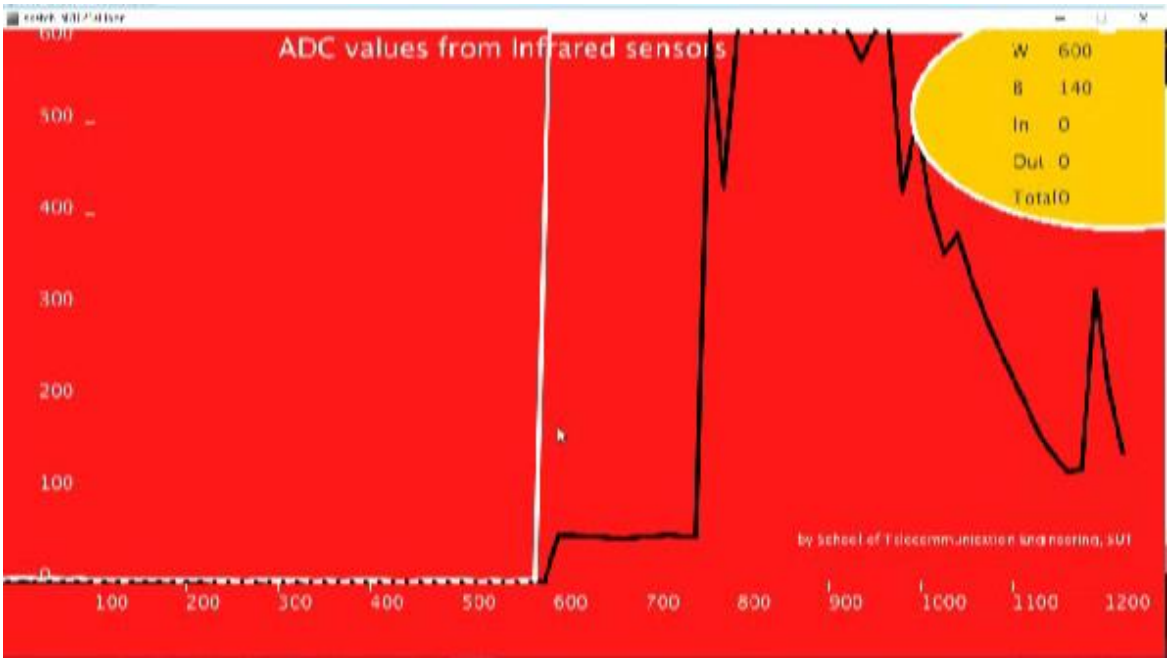


รูปที่ 3.11 การแสดงผลแบบ Animation ขณะไม่มีคนเดินเข้าภายในอาคารสำหรับ Smart Floors

3.3.7.2 แผนไดอะแกรมแสดงการทำงานขณะมีคนเดินเข้าภายในอาคาร สำหรับ Smart Floors

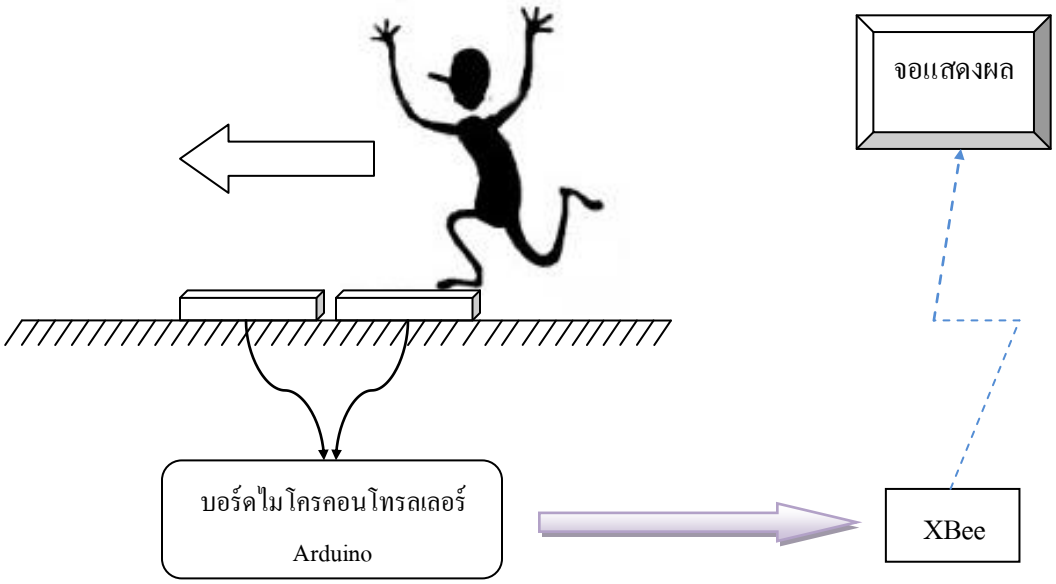


รูปที่ 3.12 แผนไดอะแกรมแสดงการทำงานขณะมีคนเดินเข้าภายในอาคารสำหรับ Smart Floors

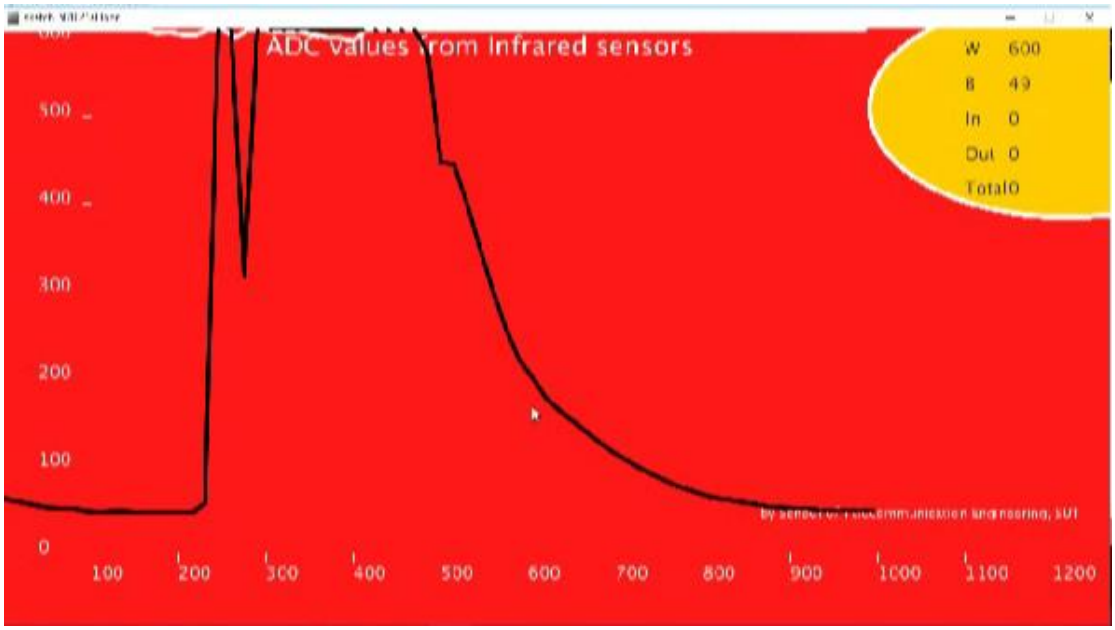


รูปที่ 3.13 การแสดงผลแบบกราฟขณะเดินเข้าในอาคารสำหรับ Smart Floors

3.3.7.3 แผนไดอะแกรมแสดงการทำงานขณะมีคนเดินออกนอกอาคารอาคาร สำหรับ Smart Floors



รูปที่ 3.14 แผนไดอะแกรมแสดงการทำงานขณะมีคนเดินออกนอกอาคารสำหรับ Smart Floors



รูปที่ 3.15 การแสดงผลแบบกราฟขณะเดินออกนอกอาคารสำหรับ Smart Floors

จากการทดสอบการใช้ Smart Floors นับจำนวนคนเข้าออกอาคารตอนเริ่มทำโครงงานนี้ ปัญหาที่พบคือความต่างศักย์ที่ตัว Smart Floors มีการเพิ่มลดเนื่องจากการเหยียบหรือแรงกดทับต่างๆทำให้กราฟที่ได้มีความไม่สม่ำเสมอจึงทำให้คณะผู้จัดโครงงานนี้ไม่สามารถนำ Smart Floors มาใช้ในการนับจำนวนคนภายในอาคารได้และได้เปลี่ยนมาใช้เป็น Infrared Sensor ในการนับจำนวนคนเข้าออกอาคารแทน Smart Floors

บทที่ 4

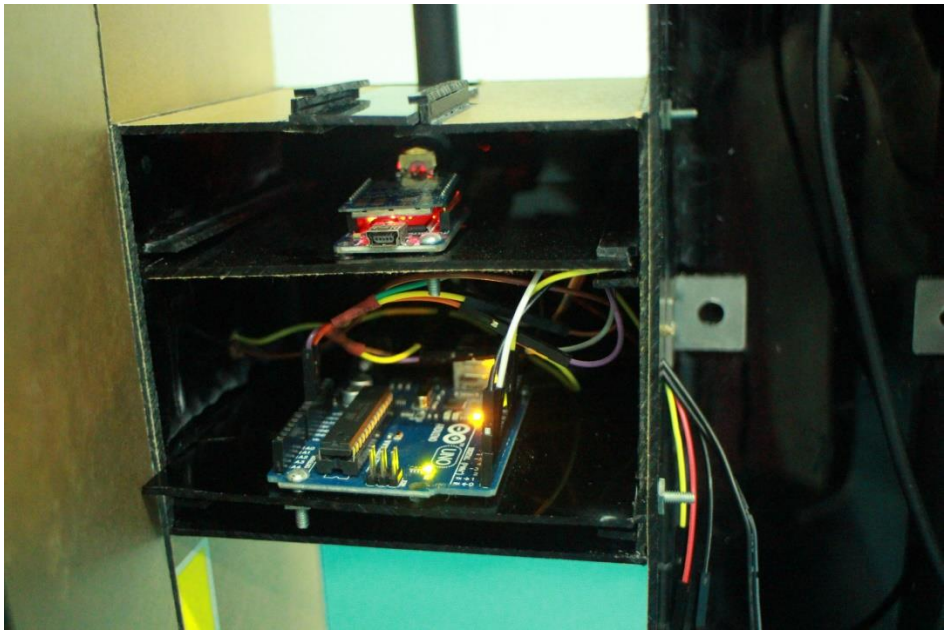
ผลการทดสอบ

4.1 บทนำ

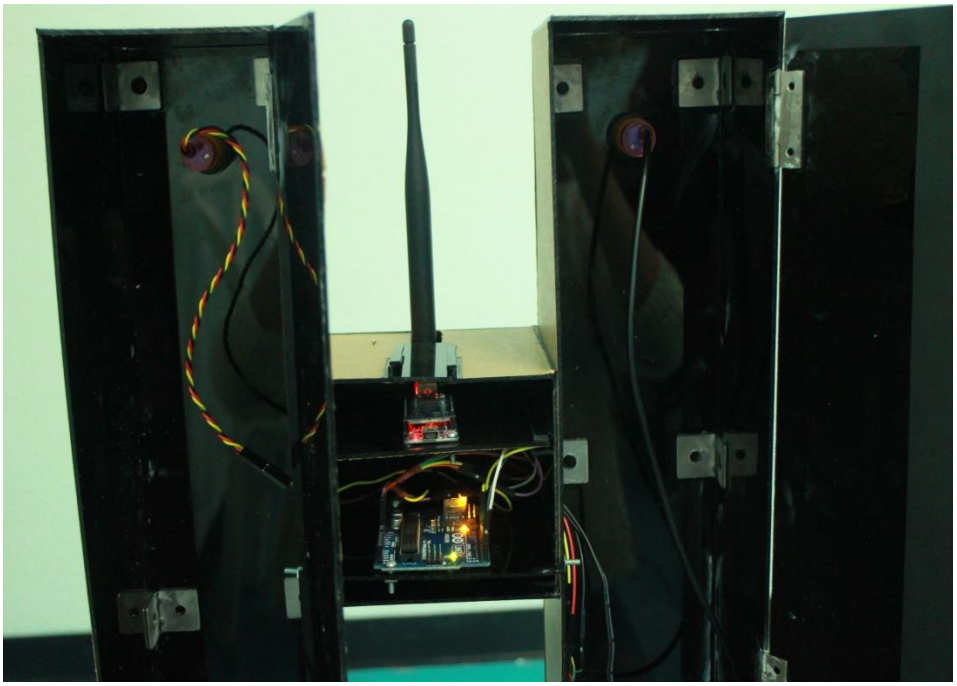
เนื้อหาในบทนี้จะเป็นการกล่าวถึงการออกแบบโครงสร้างและการสร้างวงจรภายในอุปกรณ์สำหรับการนับจำนวนคนด้วย Infrared Sensor สำหรับนำมาตรวจจับบุคคลเข้าออกอาคาร แล้วทำการทดสอบการใช้งานได้จริงของอุปกรณ์แต่ละตัวที่กลุ่มผู้ทดสอบได้นำมาประกอบเป็นอุปกรณ์สำหรับการนับจำนวนคนด้วย Infrared Sensor มาเป็นวงจรภายในวงจร 3 ส่วนดังนี้ Infrared Sensor ส่งสัญญาณดิจิตอลมายังบอร์ด Arduino บอร์ด Arduino แปลงค่าสัญญาณตามโค้ดที่เขียนแล้วส่งมาไปที่ Xbee และ Xbee ส่งค่าไปที่ Xbee ภาครับเพื่อทำการแสดงผลที่ได้ออกทางหน้าจอ หลักการของส่วนต่างๆของเครื่องนับคนเขาออกซึ่งอาศัยเซ็นเซอร์ 2 ตัว ตั้งชื่อเป็น Sensor1 และSensor2 วางเรียงกันซึ่งอาศัยหลักการสะท้อนเมื่อมีวัตถุต่างๆมาตัดขว้าง ถ้ามีการเดินเข้ามาทำให้ Sensor1 ถูกตัดก่อนหรือเกิดการสะท้อนจากวัตถุก่อน Sensor2 จะทำให้โปรแกรมที่เขียนมาให้นับเป็นหนึ่งคนเข้าเรื่อยๆตามหลักการเดิมถ้ามีการตัดอีก แต่ถ้า Sensor2 ถูกตัดก่อนหรือเกิดการสะท้อนจากวัตถุก่อน Sensor1 จะทำให้โปรแกรมที่เขียนมาให้นับเป็นหนึ่งคนออกเรื่อยๆตามหลักการเดิมถ้ามีการตัดซ้ำอีกแล้วจะทำการลบกันเพื่อหาจำนวนคนที่เหลืออยู่ในสถานที่นั้นๆและส่งผลที่ได้มายังสถานีฐานผ่านระบบ Xbee ที่ย่าน ความถี่ 2.4 GHz และแสดงผลด้วย Real-time plot และ Animation ซึ่งสร้างด้วยโปรแกรม Processing 2.2.1

4.2 การทดสอบชิ้นงาน

กลุ่มผู้จัดทำได้จัดทำแบบอุปกรณ์สำหรับที่จะใช้ในการทดสอบดังนี้



รูปที่ 4.1 การติดตั้งบอร์ดที่ใช้ในการทำงานทั้งหมด

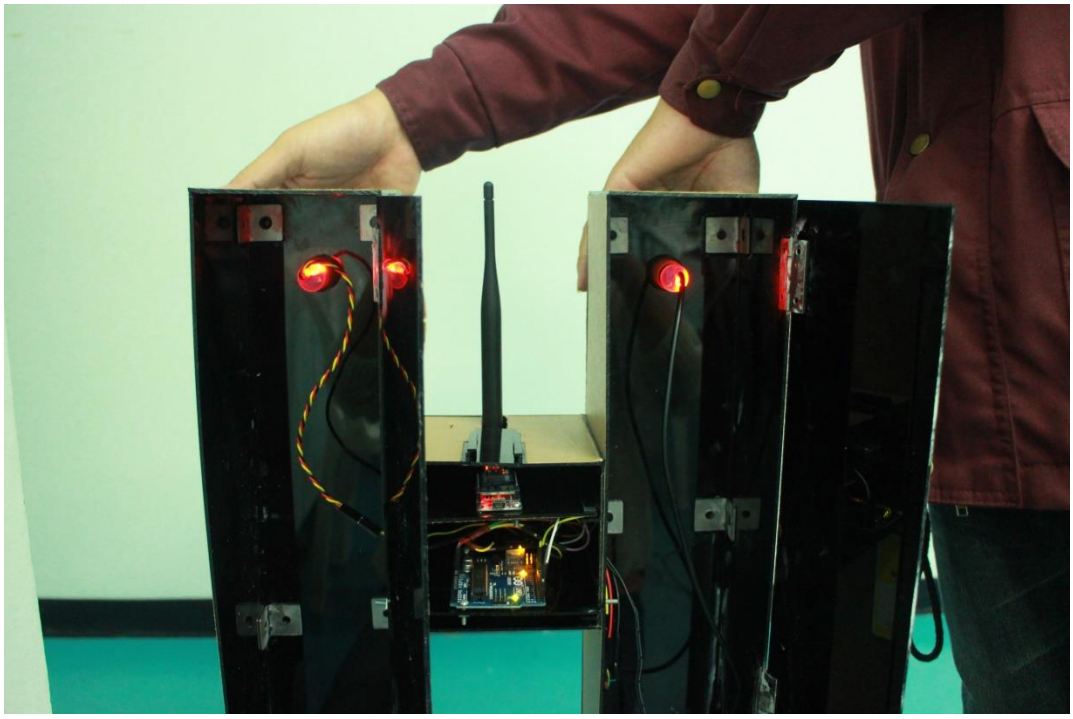


รูปที่ 4.2 การติดตั้งเซ็นเซอร์อินฟราเรด

4.2.1 การทดสอบการตรวจจับของเซนเซอร์

บริเวณด้านหน้าของแท่งเสาสำหรับนับจำนวนคนเข้าออก จะมีเซ็นเซอร์คอยตรวจจับผู้ใช้ที่เดินเข้าออกโดยมีหลักการดังนี้

- ขณะที่ยังไม่มีรถยนต์ผ่านเข้ามาในระยะที่เซนเซอร์ทางเข้าตรวจจับบอร์ดจะสั่งการโดยการส่งค่า 0 ไปที่ Monitor เพื่อทำการแสดงผล
- ขณะที่มีวัตถุผ่านเข้ามาในระยะที่เซ็นเซอร์ทางเข้าตรวจจับบอร์ดจะสั่งการโดยการส่ง 500 ไปที่ Monitor เพื่อทำการแสดงผล



รูปที่ 4.3 ไฟแสดงสถานะการทำงานของเซ็นเซอร์อินฟราเรด

4.2.2 ผลการทดสอบการตรวจจับของเซ็นเซอร์

ตารางที่ 4.1 ผลการทดสอบการตรวจจับของเซ็นเซอร์กรณีทีมีคนผ่านทางเข้า

คนที่	การนับ	
	ถูก	ผิด
1	✔	-
2	✔	-
3	✔	-
4	✔	-
5	✔	-
6	✔	-
7	✔	-
8	✔	-
9	✔	-
10	✔	-

ตารางที่ 4.2 ผลการทดสอบการตรวจจับของเซ็นเซอร์กรณีที่มีคนผ่านทางออก

คนที่	การนับ	
	ถูก	ผิด
1	✔	-
2	✔	-
3	✔	-
4	✔	-
5	✔	-
6	✔	-
7	✔	-
8	✔	-
9	✔	-
10	✔	-

4.3 สรุป

จากผลการทดสอบการตรวจจับของเซ็นเซอร์สำหรับนับคนเข้าในตารางที่ 4.1 และการตรวจจับของเซ็นเซอร์สำหรับนับคนเข้าในตารางที่ 4.2 พบว่าเซ็นเซอร์สองตัวมีนับผิดเป็นบ้างครั้ง เพราะมีรัศมีทำงานไม่เท่ากันขึ้นอยู่กับแสงสว่างและความเข้มของวัตถุ เมื่อมีคนหรือวัตถุตัดผ่านไฟที่เซ็นเซอร์จะติดแต่ถ้าไม่มีไรตัดผ่านไฟที่เซ็นเซอร์ก็จะดับจึงสรุปได้ว่า วงจรที่ใช้ในการทดสอบนี้ทำงานตามจุดประสงค์ที่ผู้ทดสอบวางไว้ได้

บทที่ 5
บทสรุปของโครงงาน

5.1 บทนำ

เนื้อหาในบทนี้เป็นการกล่าวถึงบทสรุปของโครงงาน เรื่อง เซ็นเซอร์ไร้สายสำหรับนับจำนวนคนภายในอาคาร ด้วย Infrared Sensor และ Smart Floors (Wireless Sensors for Counting Visitors in a Building using Infrared Sensors and Smart Floors) ซึ่งจะประกอบไปด้วยปัญหาในการดำเนินงาน วิธีการแก้ไข และ แนวทางในการพัฒนาโครงงานต่อไป

ซึ่งภายในแท่งเสาสำหรับนับจำนวนคนนั้น จะมีส่วนประกอบไปหลัก 3 ส่วนดังนี้

1. เซ็นเซอร์อินฟราเรดสำหรับตรวจจับ
2. บอร์ด Arduino สำหรับรับค่าแล้วประมวลผล
3. Xbee สำหรับส่งค่าที่ได้แบบไร้สายเพื่อแสดงผลจอทางหน้าจอ

บทสรุปของโครงงานเรื่อง เรื่อง เซ็นเซอร์ไร้สายสำหรับนับจำนวนคนภายในอาคาร ด้วย Infrared Sensorและ Smart Floors (Wireless Sensors for Counting Visitors in a Building using Infrared Sensors and Smart Floors) จากการทดสอบการทำงานภายในแท่งเสาสำหรับนับจำนวนคนจะเห็นได้ว่าตัวเซ็นเซอร์นั้นมีรัศมีการตรวจจับที่ไม่แน่นอนและยังขึ้นอยู่กับแสงสว่างอีกทั้งความเข้มของวัตถุอีกด้วย แต่อย่างไรก็ทำให้วงจรที่ใช้ในการทดสอบนี้ทำงานตามจุดประสงค์ที่ผู้ทดสอบวางไว้ได้

5.2 ข้อเสนอแนะ

ในการทดสอบ Infrared Sensor จะมีปัญหาเกิดขึ้นคือ ตัวเซ็นเซอร์นั้นมีรัศมีการตรวจจับที่ไม่แน่นอนและยังขึ้นอยู่กับแสงสว่างอีกทั้งความเข้มของวัตถุอีกด้วย ซึ่งสามารถแก้ปัญหาได้ดังกล่าว ดังนี้

1.ใช้แผ่นสะท้อนแสงเพื่อเพิ่มรัศมีของเซ็นเซอร์ออกไปอีก

2.ลดระยะความกว้างของทางเดิน

5.3 แนวทางการพัฒนา

โครงงานชิ้นนี้เป็นต้นแบบการนับจำนวนคนแบบไร้สายด้วย Infrared Sensor ซึ่งสามารถนำมาประยุกต์ใช้กับระบบต่างๆได้เช่น ระบบป้องกันการชนหรือการขีดข่วนสำหรับรถยนต์ ระบบกันขโมย ระบบนับจำนวนสินค้าในโรงงานอุตสาหกรรมต่าง เป็นต้น รวมทั้งสามารถนำไปต่อยอดกับระบบรักษาความปลอดภัยด้านต่างๆ

ประวัติผู้เขียน

นางสาววสุนธรา ปะมา
เกิดเมื่อวันที่ 2 กรกฎาคม พ.ศ. 2535
ที่อยู่ 336 ม.16 ตำบลปะหลาน อำเภอพยัคฆภูมิพิสัย จังหวัดมหาสารคาม
สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนพยัคฆภูมิวิทยาคาร
ปีการศึกษาที่ 2553
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

นางสาวพัชรพิมล แห้วเพ็ชร์
เกิดเมื่อวันที่ 27 สิงหาคม พ.ศ. 2535
ที่อยู่ 67 หมู่ 1 ตำบลห้วยแห้ง อำเภอบ้านไร่ จังหวัดอุทัยธานี
สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนบ้านไร่วิทยา
ปีการศึกษาที่. 2553
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

นายสมพร สินปรุ
เกิดเมื่อวันที่ 26 กรกฎาคม พ.ศ. 2534
ที่อยู่ 301 หมู่ที 11 บ้านไทยอยู่เย็น ตำบลหนองตะไก้อำเภอหนองบุญมาก
จังหวัดนครราชสีมา
สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนหนองบุนนากพิทยาคม
ปีการศึกษาที่2552
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก

ภาคผนวก ก

การใช้งานโปรแกรม Arduino IDE

1. ทำการดาวน์โหลดโปรแกรม Arduino IDE ที่ ลิงค์ http://arduino.cc/en/Main/Software และทำการติดตั้งโปรแกรม

รูปที่ 1 หน้าต่างเว็บไซต์สำหรับโหลดโปรแกรม Arduino IDE

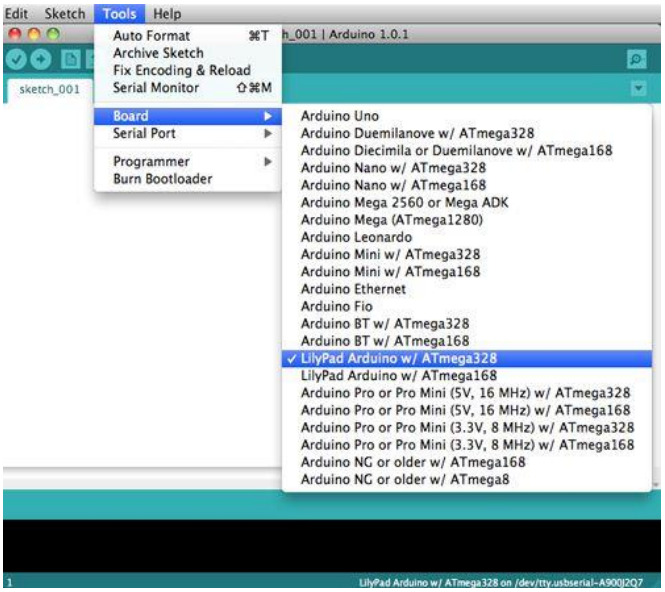
2. คลิกที่ไอคอนของโปรแกรม Arduino IDE ดังรูป

รูปที่ 2 ไอคอนโปรแกรม Arduino IDE

3. เมื่อเปิดโปรแกรมแล้วจะพบกับหน้าต่างของ IDE ดังรูป

รูปที่ 3 หน้าต่างโปรแกรม Arduino IDE

4. ไปที่ Tools->Board แล้วเลือกให้ตรงกับบอร์ดที่ใช้งาน โดยเลือกตาชื่อบอร์ดที่เราต้องการ



รูปที่ 4 ขั้นตอนการเลือกบอร์ดสำหรับเขียนโค้ด

5. เขียนโปรแกรมดังข้อความด้านล่างนี้

```
const int sensors1=10;
const int sensors2=9;
// the setup function runs once when you press reset or power the board
void setup() {
  pinMode(sensors1,INPUT);
  pinMode(sensors2,INPUT);
  Serial.begin(9600);
}
// the loop function runs over and over again forever
void loop() {
  int state1=digitalRead(sensors1);
  int state2=digitalRead(sensors2);
  if(state1==HIGH){
    Serial.print("0");
  }else{
    Serial.print("500");
 }
 Serial.print(",");
if(state2==HIGH){
    Serial.print("0");
    }else{
    Serial.print("500");
  }
Serial.println();
delay(500);
}
```

รูปที่ 5 ตัวอย่างโค้ดที่ใช้อัพโหลดลงบอร์ด Arduino

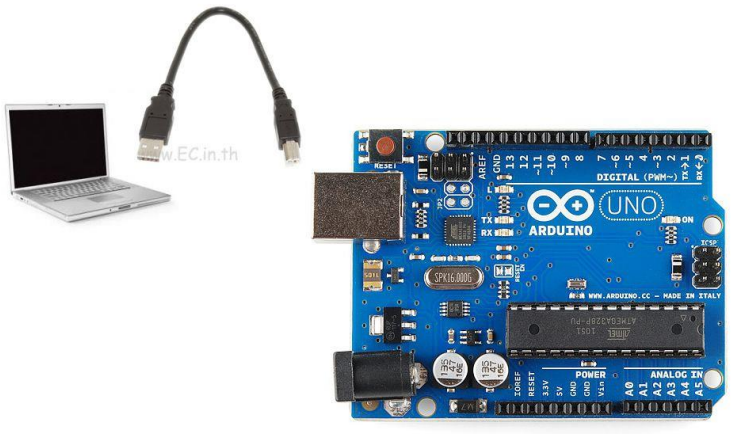
6. จากนั้นคอมไฟล์โปรแกรมโดยไปที่ Sketch->Verify / Compile

รูปที่ 6 ขั้นตอนการตรวจสอบความถูกต้องของโค้ด

7. เมื่อคอมไฟล์เรียบร้อยจะมีข้อความปรากฏดังรูป

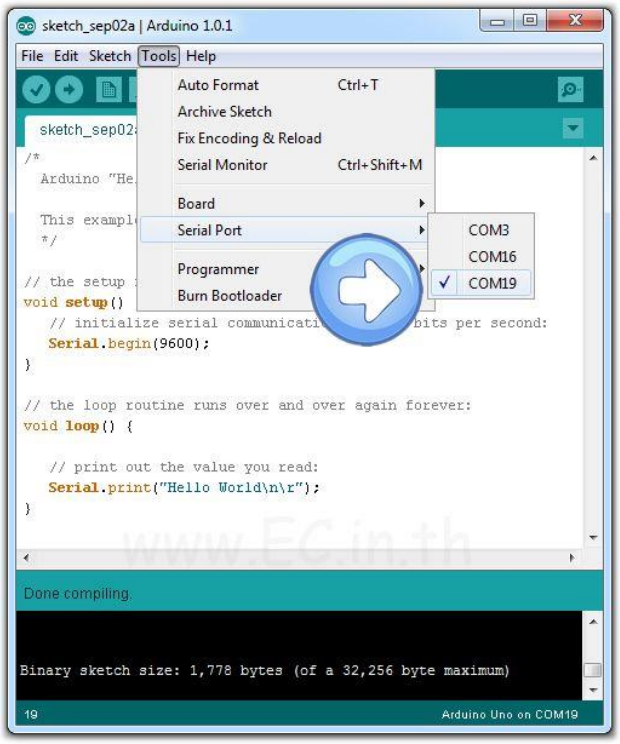
รูปที่ 7 แสดงสถานะความถูกต้องของโค้ด

8.ทำการเชื่อมต่อบอร์ด Arduino ที่เราใช้งานเข้ากับคอมพิวเตอร์ผ่านทางพอร์ต USB



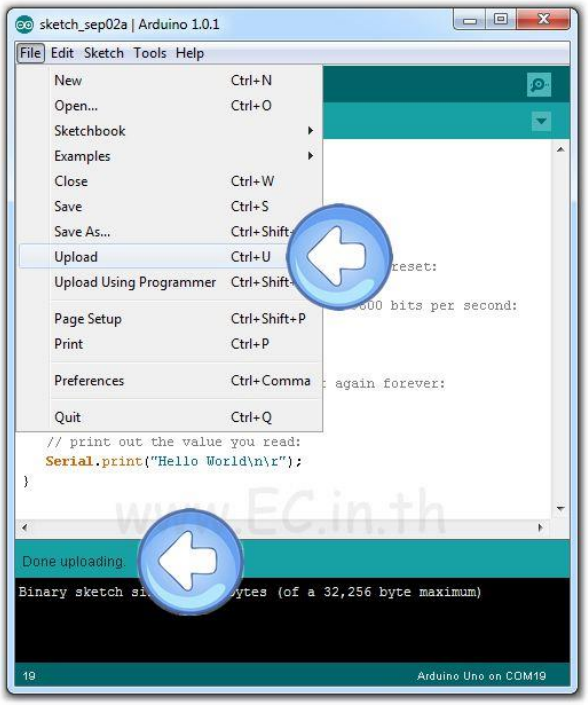
รูปที่ 8 การเชื่อมต่อบอร์ด Arduino เข้ากับคอมพิวเตอร์ผ่านทางพอร์ต USB

9. จากนั้นให้ไปที่ Tools->Serial Port และเลือกให้ตรงกับบอร์ด Arduino ที่เราใช้งานที่ใช้งาน (สำหรับบอร์ด Arduino ของเราโปรแกรมจะเลือกให้อัตโนมัติ)



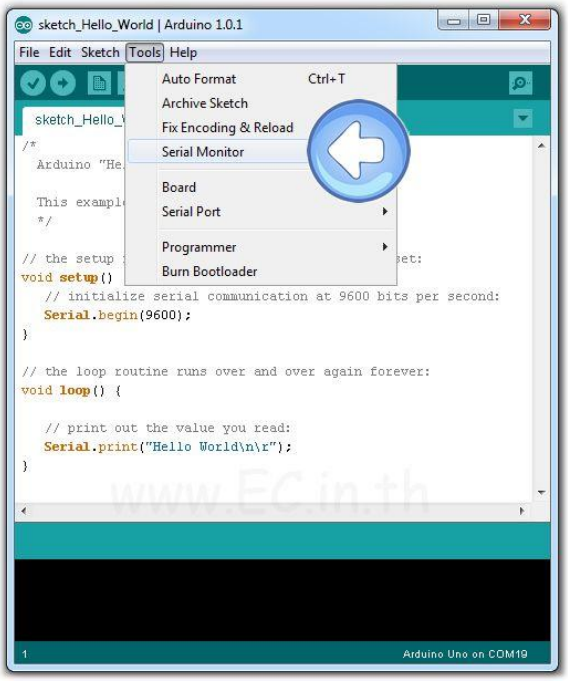
รูปที่ 9 การเลือกพอร์ตให้ตรงกับบอร์ดที่ต้องการใช้งาน

10. โหลดโปรแกรมเข้าบอร์ด Arduino ที่เราใช้งานโดยไปที่ File->Upload



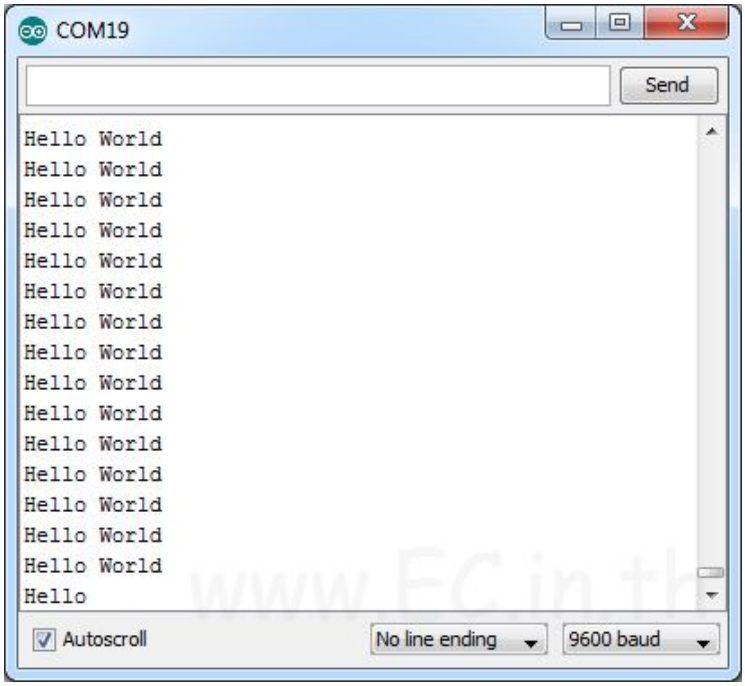
รูปที่ 10 ขั้นตอนการอัพโหลดโค้ดลงบอร์ด Arduino

11. จากนั้นเปิด Serial Monitor ของ Arduino IDE โดยไปที่ Tools->Serial Monitor



รูปที่ 11 ขั้นตอนในการเลือกใช้การแสดงผล

12. เมื่อเปิด Serial Monitor จะได้ข้อความดังรูป



รูปที่ 12 แสดงผลที่ได้โค้ดที่อัพโหลดลงบอร์ด Arduino

คำสั่งพื้นฐานในชิป

pinMode(ขาที่จะใช้ ,OUTPUT) = ตั้งค่าให้ขานั้นเป็นเอาท์พุทใช้ได้ทั้งอนาล็อกและดิจิตอล

pinMode(ขาที่จะใช้ ,INPUT) = ตั้งค่าให้ขานั้นเป็นอินพุทใช้ได้ทั้งอนาล็อกและดิจิตอล

digitalWrite(ขาที่จะใช้,สถานะที่จะแสดง) = ใช้สั่งขานั้นเป็นเอาท์พุทดิจิตอล เช่น LED หน้าจอ

digitalRead(ขาที่จะใช้) = ใช้สั่งขานั้นรับอินพุทดิจิตอล เช่น สวิตซ์กด เซ็นเซอร์ดิจิตอล

analogWrite(ขาที่จะใช้) = ใช้สั่งขานั้นเป็นเอาท์พุทอนาล็อก เช่น IC ควบคุมความเร็วมอเตอร์

analogRead(ขาที่จะใช้) = ใช้สั่งขานั้นรับอินพุทอนาล็อก เช่น Rเกือกม้า LDR เซ็นเซอร์วัดระยะทาง

delay(เวลาหน่วยเป็นมิลลิวินาที) = ใช้หน่วงเวลาทำงานก่อนทำงานคำสั่งต่อไป

delayMicroseconds(เวลาหน่วยเป็นไมโครวินาที) = ใช้หน่วงเวลาทำงานก่อนทำงานคำสั่งต่อไป

int = ใช้ตั้งชื่อตัวแปรสำหรับตั้งค่าเริ่มต้นที่เป็นตัวเลขหรือจองพื้นที่เก็บตัวเลขหรือตั้งชื่อให้ขา

char = ใช้ตั้งชื่อตัวเปรที่ใช้เก็บตัวอักษร(จองพื้นที่ใช้เก็บอักษร)

Serial.begin(9600) = ตั้งค่าเริ่มต้นเพื่อติดต่อสื่อสารกับคอมพิวเตอร์อัตราเร็ว9600บิตต่อวินาที

Serial.print(“คำพูด”) = ใช้พิมพ์คำพูดเพื่อให้แสดงผลบนจอคอมแบบไม่เว้นบรรทัด

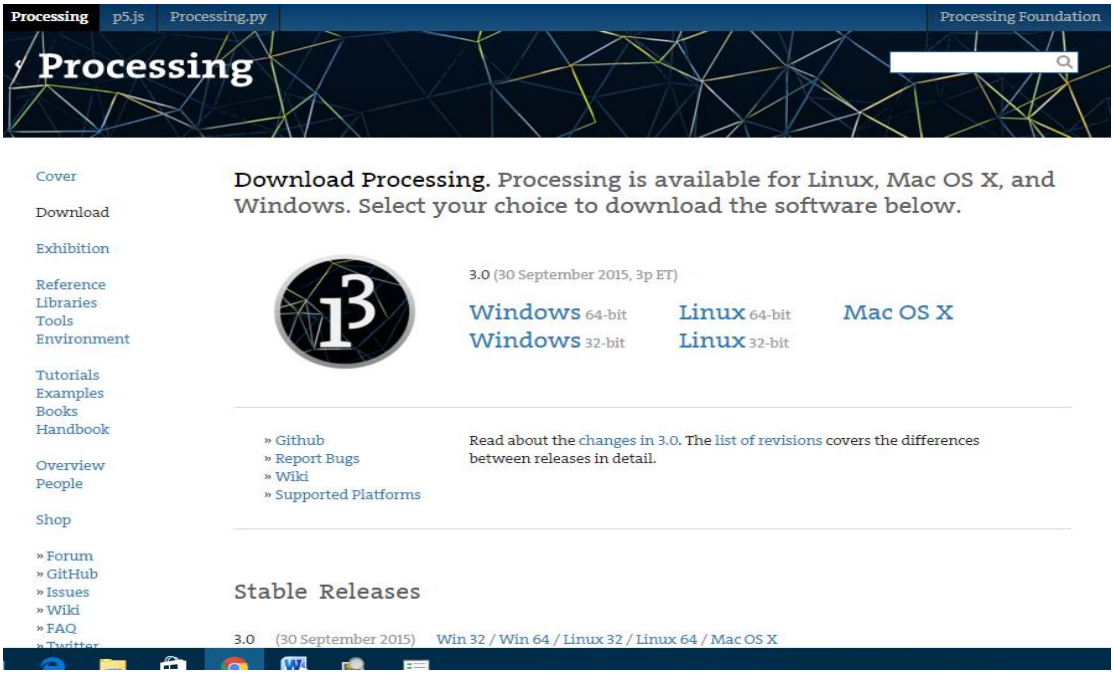
Serial.println(“คำพูด”) = ใช้พิมพ์คำพูดเพื่อให้แสดงผลบนจอคอมแบบเว้นบรรทัด

Serial.available() = ใช้ตรวจสอบว่ามีการกดคีย์บอร์ดหรือไม่

Serial.Read() = ใช้อ่านค่าปุ่มคีย์บอร์ด

การใช้งานโปรแกรม Processing

1.ทำการดาวน์โหลดโปรแกรม Processing ที่ลิงค์ https://processing.org/download/?processing และทำการติดตั้งโปรแกรม



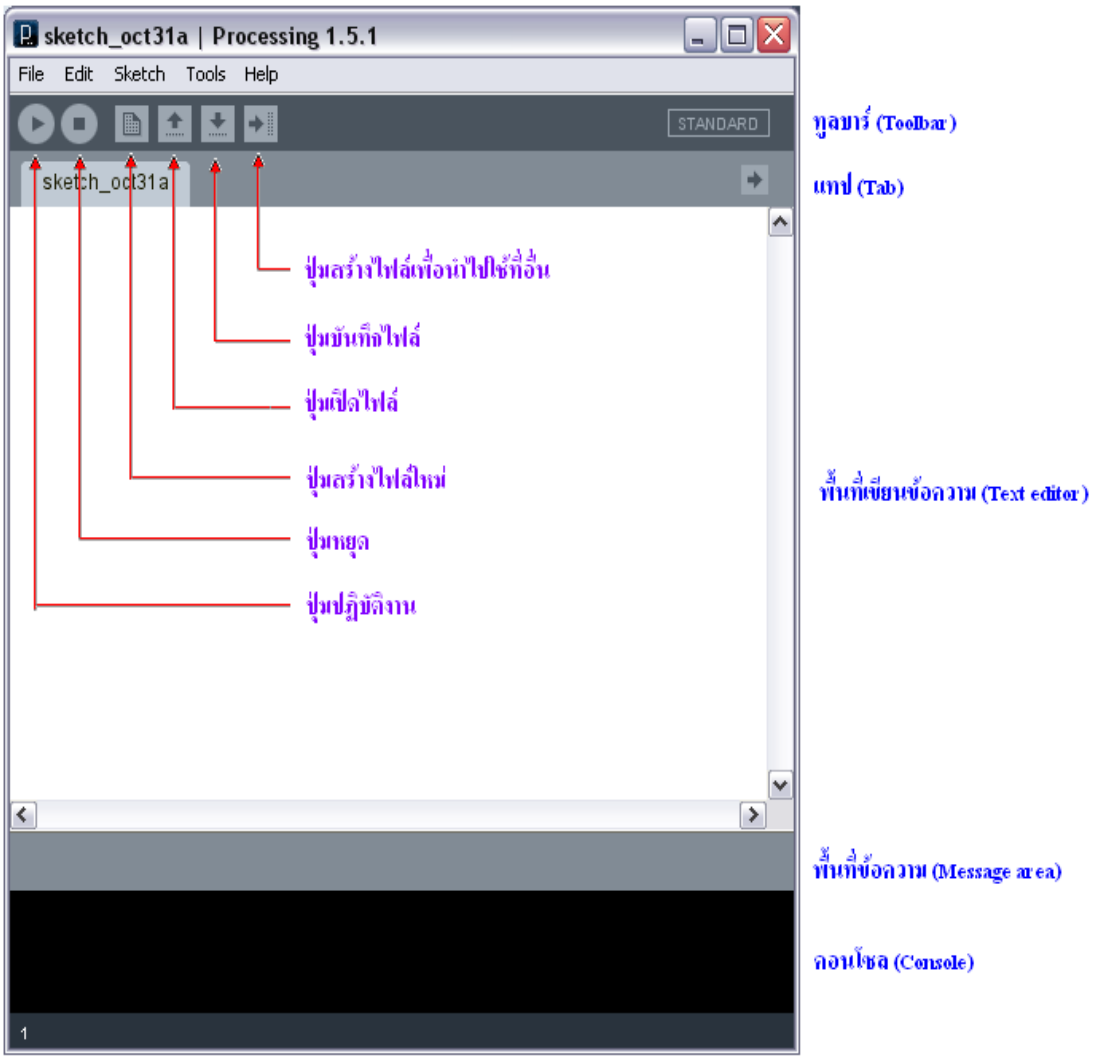
รูปที่ 13 หน้าต่างเว็บไซต์สำหรับโหลดโปรแกรม Processing

2. คลิกที่ไอคอนของโปรแกรม Processing ดังรูป

รูปที่ 14 ไอคอนโปรแกรม Processing

โปรแกรม Processing Shortcut

Processing เป็นซอฟท์แวร์ระบบเปิด เหมาะสำหรับผู้ต้องการพัฒนาโปรแกรมเกี่ยวกับการสร้างภาพเคลื่อนไหวและการมีปฏิสัมพันธ์ สำหรับผู้ที่เคยใช้ชุดพัฒนาโปรแกรม Arduino เมื่อเห็นรูปร่างหน้าตาการอินเตอร์เฟสของ Processing แล้วจะรู้สึกคุ้นเคยมาก ด้วยเพราะเหมือนกันนั่นเอง อีกทั้ง Processing และ Arduino ใช้หลักการในการเขียนโปรแกรมเหมือนกัน โดยมีพื้นฐานมาจากภาษา C/C++ รวมถึงการติดตั้งชุดพัฒนาก็เหมือนกันด้วย



รูปที่ 15 แสดงเครื่องมือต่างๆของโปรแกรม Processing

ฟังก์ชันพื้นฐานของ Processing

background() = การกำหนดสีพื้นหลังของภาพที่จะทำการแสดง

clear() = เป็นการเคลียร์ค่าของการวนลูปแต่ละครั้ง

fill() = เป็นการเติมสีให้รูปทรงต่างๆ

draw() = เป็นฟังก์ชันใช้ในการวนลูป

line() = ฟังก์ชันที่ใช้ในการกำหนดเส้นในแนวต่างๆ

int = ใช้ตั้งชื่อตัวแปรสำหรับตั้งค่าเริ่มต้นที่เป็นตัวเลขหรือจองพื้นที่เก็บตัวเลขหรือตั้งชื่อให้ขา

char = ใช้ตั้งชื่อตัวเปรที่ใช้เก็บตัวอักษร(จองพื้นที่ใช้เก็บอักษร)

float = ใช้ตั้งชื่อตัวเปรที่ใช้เก็บเลขที่เป็นทศนิยม

size() = ใช้ในการกำหนดขนาดหน้าตางที่จะทำการแสดงผล

texe() = ใช้ในการกำหนดจุดที่แสดงผลและขนาดที่ต้องการแสดง

ภาคผนวก ข

ขั้นตอนในการทำโครงสร้างของแท่งเสาสำหรับนับจำนวนคน

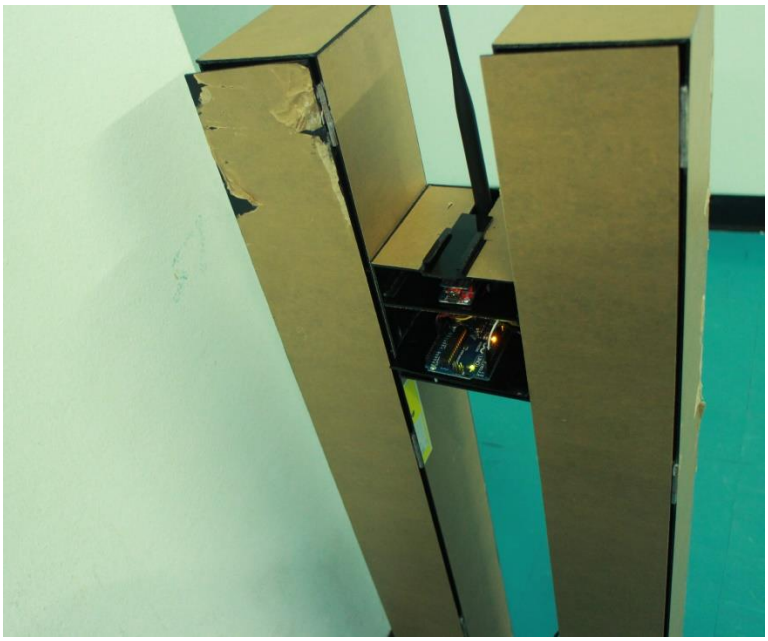
1. ทำการตัดแผ่นอะคริลิค

รูปที่ 16 การตัดแผ่นอะคริลิค

2. เชื่อมแผ่นอะคริลิคที่ตัดไว้เข้าด้วยกันด้วยกาวร้อนและตามด้วยเหล็ก

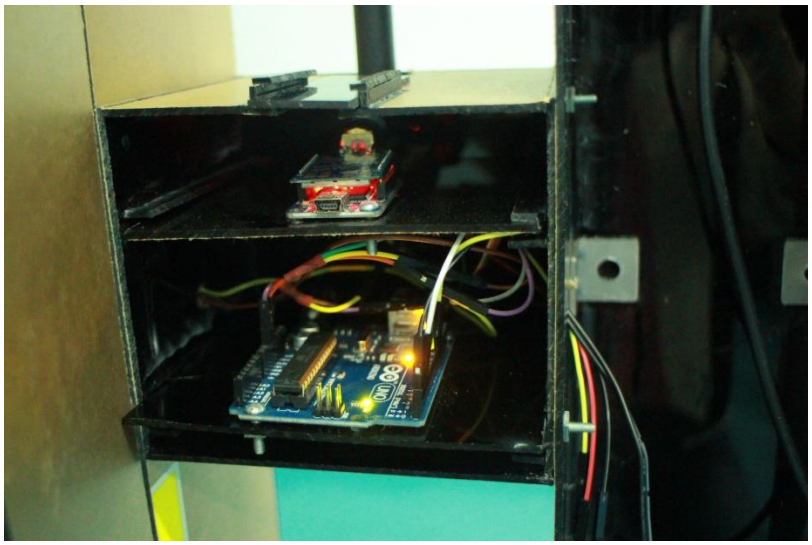
รูปที่ 17 การเชื่อมแผ่นอะคริลิค

3.ทำการประกอบเข้าด้วยกัน



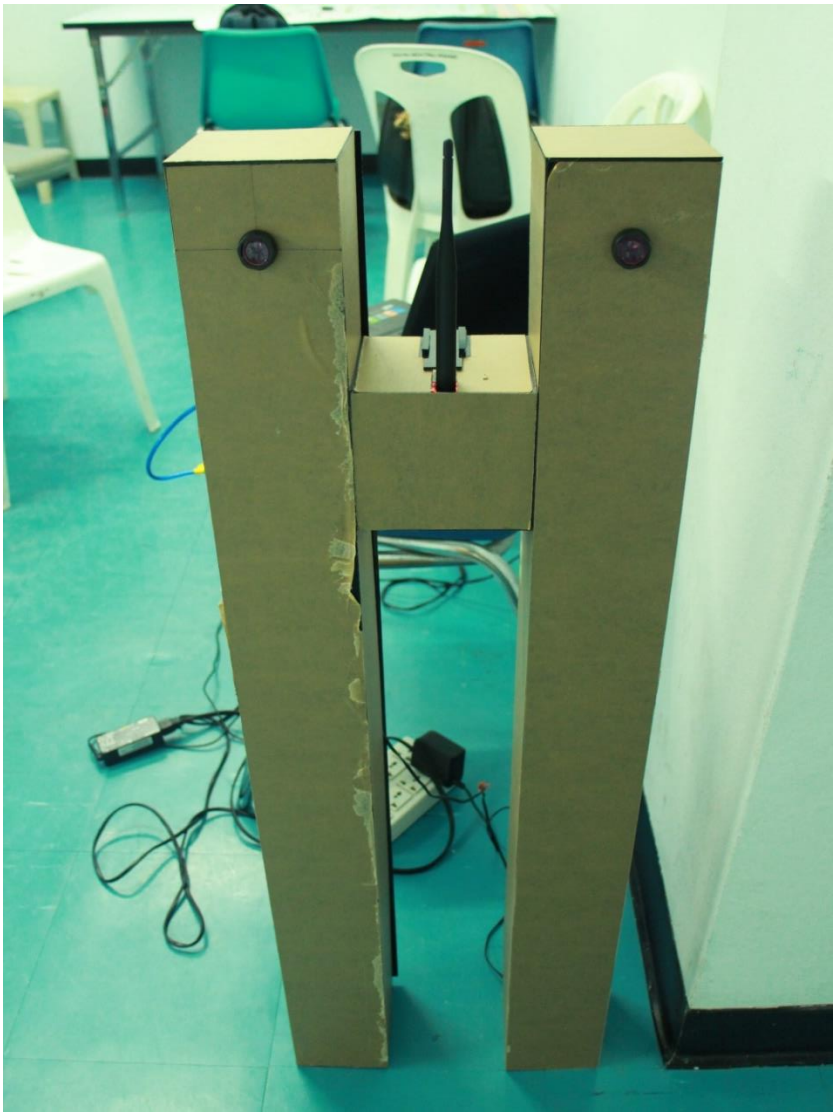
รูปที่ 18 ประกอบแผ่นอะคริลิคเข้าด้วยกัน

4.ทำการใส่เซ็นเซอร์ บอร์ด Arduino และ Xbee



รูปที่ 19 การใส่บอร์ดควบคุมการทำงาน

5.เมื่อนำส่วนต่างๆมาประกอบกันเสร็จจะได้รูปร่างดังรูป



รูปที่ 20 แท่งเสาสำหรับนำจำนวนคนเข้าออกอาคาร

บรรณนานุกรม

[1] การทำงานเซนเซอร์อินฟราเรด

http://www.mynpe.com (เซ็นเซอร์อินฟราเรด (Infrared Sensor))

http://www.thaigoodview.com/node/118154

[2] การทำงานของตัวรับ

http://electronicspocketbook.blogspot.com/2014/01/photo-transistor.html

[3] การทำงานของตัวส่ง

http://cpre.kmutnb.ac.th/esl/learning/index.php?article=ir-sender-receiver

[4] คุณสมบัติของไดโอดเปล่งแสง (LED)

http://electronics.se-ed.com/contents/112s096/112s096_p07.asp

http://th.wikipedia.org/wiki/%E0%B9%84%E0%B8%94%E0%B9%82%E0%B8%AD%E0%B8%94%%B9%80%E0%B8%9B%E0%B8%A5%E0%B9%88%E0%B8%87%E0%B9%81%E0%B8%AA%E0%%B8%87

[5] การใช้งานไดโอดเปล่งแสงกับบอร์ด Arduino

http://www.arduino.cc/en/Tutorial/Blink?from=Tutorial.BlinkingLED

[6] การใช้งานบอร์ด Arduino เบื้องต้น

http://www.thaieasyelec.com/articlewiki/basicelectronics/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1arduino%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3%E0%B9%80%E0%B8%A3%E0%B8%B4%E0%B9%88%E0%B8%A1%E0%B8%95%E0%B9%89%E0%B8%99%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99-arduino.html

https://www.gravitechthai.com/guru2.php?p=260

[7] อะคริลิค

http://www.siamchemi.com/%E0%B8%AD%E0%B8%B0%E0%B8%84%E0%B8%A3%E0%B8%B4%E0%B8%A5%E0%B8%B4%E0%B8%84/

[8] เริ่มต้นเรียนรู้การเขียนโปรแกรมด้วยภาษา Processing

http://cpre.kmutnb.ac.th/esl/2014/08/learn-programming-with-processing/

[9] ข้อมูลของ Xbee

http://www.es.co.th/Schemetic/PDF/XBEE-PRO.pdf

[10]Smart Floors

รูปเล่มโปรเจคของนักศึกษาสาขาวิชาเซรามิกมหาวิทยาลัยเทคโนโลยีสุนารี