รายงานสหกิจศึกษาฉบับสมบูรณ์

ชุดอุปกรณ์ตรวจวัดมุมระหว่างหัวยิงรังสีและแผ่นตรวจจับรังสี

Implement orientation detector using BNO 055 for X- ray imaging Alignment

นางสาวเบญญทิพย์ มาศงามเมือง

ภาควิชาวิศวกรรมชีวการแพทย์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงงานสหกิจศึกษา ชุดอุปกรณ์ตรวจวัดมุมระหว่างหัวยิงรังสี และแผ่นตรวจจับรังสี

ชื่อ-สกุล นักศึกษา นางสาวเบญญทิพย์ มาศงามเมือง

คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมชีวการแพทย์

ชื่อ-สกุล อาจารย์นิเทศ ดร.ตรีสุคนธ์ ตรีบุพชาติสกุล

ชื่อ-สกุล ผู้นิเทศงาน นายอุดมชัย เตชะวิภู

สถานประกอบการ สำนักงานพัฒนาวิทยาศาสตร์ และเทคโนโลยีแห่งชาติ (สวทช.)

บทคัดย่อ

โครงการนี้ทดลองศึกษาการทำงานของ motion sensor module เพื่อนำมาประยุกต์ใช้ในการเปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสี ซึ่งความแม่นยำของการตั้งหัวหลอดเอกซเรย์กับฉากรับรังสีให้ขนานกันมีผลต่อประสิทธิภาพในการใช้งานเครื่องเอกซเรย์ โดยโครงการนี้ศึกษาการทำงานของ motion sensor module และทำการเขียนโปรแกรมเพื่อตรวจสอบพิกัดของหัวหลอดเอกซเรย์ และพิกัดของฉากรับรังสี แล้วนำค่าที่ได้มาตรวจสอบประสิทธิภาพ และความแม่นยำของค่ามุมที่ได้ในสภาวะต่าง ๆ ของ motion sensor module

โดยเริ่มจากทำการรวบรวมข้อมูล ศึกษาระบบการทำงาน ออกแบบพัฒนาอุปกรณ์ที่ใช้ออกแบบพัฒนาโปรแกรมการทำงานของชุดอุปกรณ์ และทดสอบโปรแกรมการทำงานของชุดอุปกรณ์

จากการทดสอบชุดอุปกรณ์ตรวจวัดมุมระหว่างหัวหลอดเอกซเรย์ และฉากรับรังสีแสดงให้เห็นว่าชุดอุปกรณ์สามารถบอกมุมของของหัวยิงรังสี และแผ่นตรวจจับรังสีได้ตามขอบเขตที่ได้วางไว้ คือสามารถบอกตำแหน่ง ค่ามุมของหัวยิงรังสี และแผ่นตรวจจับรังสี ออกมาในรูปแบบของค่า Roll, Pitch, Yaw

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cooperative Title: Implement orientation detector using BNO 055 for X-ray imaging Alignment

Student intern name: Ms.Benyathip Masngammuang

Faculty: Engineering **Department:** Biomedical Engineering

Advisor name: Dr.Treesukon Treebupachatsakul

Mentor name: Mr.Udomchai Techavipoo

Company: National Science And Technology Development Agency (NSTDA)

ABSTRACT

The objectives of project were to study the operation of the motion sensor module in order to compare the position of the X-ray generator with the X-ray detector. The accuracy of setting the x-ray generator and the detector into the parallel position affects the efficiency of the X-ray machine. This project studies the operation of the motion sensor module and writes a program to check the position of the X-ray generator and the position of the X-ray detector. Then, the obtained values were analyzed to check the efficiency and the accuracy of the system in various conditions of the motion sensor module. The process of this project starts from collecting data, studying the working system, designing and developing the device and the operating program, as well as testing the program of operation of the device. From the test of the angle measuring device between the X-ray generator and the X-ray detector, it is found that the device can indicate the angle of the X-ray generator and the X-ray detector according to the scope that has been set to report the position, angle of X-ray generator and X-ray detector in the form of Roll, Pitch, Yaw.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานเล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ดร.ตรีสุคนธ์ ตรีบุพชาติสกุล ที่ให้ความช่วยเหลือ ให้คำปรึกษาชี้แนะแนวทางการทำงาน ให้ความช่วยเหลือในการแก้ไขปัญหาที่เกิดระหว่างการทำงานตลอดจนมอบความรู้สึกที่ดีแก่ข้าพเจ้า

ขอขอบคุณนางเสาวภาคย์ ธงวิจิตรมณี และคณะทำงานในสังกัด ทีมวิจัยระบบสร้างภาพทางการแพทย์ สำนักงานพัฒนาวิทยาศาสตร์ และเทคโนโลยีแห่งชาติ (สวทช.) ที่ให้ความรู้ ทักษะ และมอบประสบการณ์ในการทำงานตลอดระยะเวลา 4 เดือนแก่ข้าพเจ้า

ขอขอบคุณ นายอุดมชัย เตชะวิภู นายนริศ อัศวเลิศศักดิ์ และนายฐิติวัฒน์ เย็นวิชัย ที่คอยให้ความรู้ คำปรึกษา และชี้แนะแนวทางต่างๆ

สุดท้ายขอขอบคุณเพื่อนๆ ของข้าพเจ้า ที่เรียนสาขาวิศวกรรมชีวการแพทย์ ที่ให้การช่วยเหลือซึ่งกันและกัน และให้กำลังใจที่ดีเสมอมา ตลอดจนบุคคลอื่นๆที่มิได้กล่าวนามไว้ ณ ที่นี้ ที่มีส่วนช่วยให้โครงงานเล่มนี้สำเร็จลุล่วงไปด้วยดี

สำหรับคุณงามความดีอันใดที่เกิดจากโครงงานฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดา มารดา ครูอาจารย์ทุกท่าน ซึ่งเป็นที่รักและเคารพอย่างยิ่ง ที่คอยถ่ายทอดประสิทธ์ประสาทความรู้และประสบการณ์ที่ดีให้แก่ข้าพเจ้าตลอดมา

เบญญทิพย์ มาศงามเมือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง **หน้า**

บทคัดย่อ I

ABSTRACT II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญภาพ VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญ 1

1.2 วัตถุประสงค์ของโครงงาน 2

1.3 ขอบเขตของโครงงาน 2

1.4 แผนการดำเนินโครงงาน 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 4

2.1 การถ่ายภาพรังสี 4

2.2 การวัดหาพิกัดแนวทิศทางการวางตัว 5

2.3 เซ็นเซอร์ต่างๆที่สามารถใช้วัดหาพิกัด 5

2.4 อุปกรณ์ Electronic ที่สามารถใช้ระบุพิกัดได้ 8

2.5 ทฤษฎีเกี่ยวกับไมโครคอนโทรเลอร์ 9

2.6 อุปกรณ์ที่ใช้ในการแสดงผลในโครงงาน 14

2.7 อุปกรณ์ที่ใช้ในการสื่อสารในโครงงาน 17

2.8 การสื่อสารของอุปกรณ์และไมโครคอนโทรเลอร์ 19

บทที่ 3 วิธีดำเนินงานโครงงาน 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง **หน้า**

3.1 อุปกรณ์ที่ใช้ในการดำเนินงานโครงงาน 23

3.2 การออกแบบและสร้างอุปกรณ์ 23

3.3 การออกแบบโปรแกรม 30

3.4 การเขียนโปรแกรม 35

3.5 ฟังก์ชั่นเกี่ยวกับการสื่อสารผ่านพอร์ตอนุกรม 37

บทที่ 4 ผลการดำเนินงานโครงงาน 41

4.1 แผนการดำเนินงาน 41

4.2 ทดสอบอุปกรณ์ 41

4.3 ทดสอบการสื่อสาร 43

4.4 ทดสอบการทำงานของชุดอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสี 45

บทที่ 5 สรุปผลการดำเนินงาน 48

5.1 สรุปผล 48

5.2 ปัญหาและอุปสรรค 49

5.4 แนวทางการแก้ไข 49

เอกสารอ้างอิง 50

ภาคผนวก ก 53

ภาคผนวก ข 65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

เรื่อง **หน้า**

ตารางที่ 1.1 แผนดำเนินงานโครงงาน 3

ตารางที่ 3.1 ขาที่ใช้เชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์ 25

ตารางที่ 3.2 ขาที่ใช้เชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์ 26

ตารางที่ 3.3 ขาที่ใช้เชื่อมจอแสดงผล LCD 2004 กับไมโครคอนโทรเลอร์ 26

ตารางที่ 3.4 ขาที่ใช้เชื่อม buzzer กับไมโครคอนโทรเลอร์ 27

ตารางที่ 3.5 ขาที่ใช้เชื่อม switch กับไมโครคอนโทรเลอร์ 27

ตารางที่ 3.6 ขาที่ใช้เชื่อม LED กับไมโครคอนโทรเลอร์ 28

ตารางที่ 3.7 ขาที่ใช้เชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์ 29

ตารางที่ 3.8 ขาที่ใช้เชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์ 30

ตารางที่ 4.1 แสดงค่าเฉลี่ยของมุมที่สามารถวัดได้จากกล่อง A และ กล่อง B 45

ตารางที่ 4.2 แสดงค่าเฉลี่ยของข้อมูล 30 ค่า 46

ตารางที่ 4.3 แสดงค่าเบี่งเบนมาตรฐานของข้อมูล 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

เรื่อง **หน้า**

ภาพที่ 2.1 ตำแหน่งการวางตัวของวัตถุในระนาบ 3 มิติ 5
ภาพที่ 2.2 เซ็นเซอร์วัดมุมเอียง 6
ภาพที่ 2.3 Gyroscope 6
ภาพที่ 2.5 การไหลของกระแส 7
ภาพที่ 2.6 การวัดความต่างศักย์ของสนามแม่เหล็ก 7
ภาพที่ 2.7 เซ็นเซอร์ GY-89 8
ภาพที่ 2.8 เซ็นเซอร์ GY-521 MPU6050 8
ภาพที่ 2.9 เซ็นเซอร์ BNO 055 9
ภาพที่ 2.10 บอร์ด Arduino UNO R3 10
ภาพที่ 2.11 ส่วนประกอบของบอร์ด Arduino UNO R3 10
ภาพที่ 2.12 โครงสร้างไมโครคอนโทรเลอร์ ATmega328 11
ภาพที่ 2.13 บอร์ด Arduino Mega 2560 12
ภาพที่ 2.14 ส่วนประกอบของบอร์ด Arduino Mega 2560 13
ภาพที่ 2.15 LCD 14
ภาพที่ 2.16 การเชื่อมต่อ LCD กับ Arduino 15
ภาพที่ 2.17 Buzzer 15
ภาพที่ 2.18 LED 16
ภาพที่ 2.19 บลูทูธ HM-10 BLE 4.0 17
ภาพที่ 2.20 การเชื่อมต่ออุปกรณ์ แบบ I2C 20
ภาพที่ 3.1 Block diagram Device A 24
ภาพที่ 3.2 การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์ 24
ภาพที่ 3.3 การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์ 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

เรื่อง **หน้า**

ภาพที่ 3.4 การเชื่อมจอแสดงผล LCD 2004 กับไมโครคอนโทรเลอร์ 26

ภาพที่ 3.5 การเชื่อม buzzer กับไมโครคอนโทรเลอร์ 26

ภาพที่ 3.6 การเชื่อม button กับไมโครคอนโทรเลอร์ 27

ภาพที่ 3.7 การเชื่อม LED กับไมโครคอนโทรเลอร์ 28

ภาพที่ 3.8 Block diagram Device B 28

ภาพที่ 3.9 การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์ 29

ภาพที่ 3.10 การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์ 30

ภาพที่ 3.11 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A 31

ภาพที่ 3.12 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A (ต่อ) 32

ภาพที่ 3.13 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A (ต่อ) 33

ภาพที่ 3.13 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device B 34

ภาพที่ 3.14 รูปแบบการเขียนโปรแกรมบน Arduino 35

ภาพที่ 3.15 เลือกรุ่นบอร์ด Arduino ที่ต้องการ upload 35

ภาพที่ 3.16 เลือกหมายเลข Com port ของบอร์ด 36

ภาพที่ 3.17 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้องและ Compile โค้ดโปรแกรม 36

ภาพที่ 3.18 upload โค้ดโปรแกรมสำเร็จ 36

ภาพที่ 4.1 แผนการดำเนินงานช่วงแรก 41

ภาพที่ 4.2 แผนการดำเนินงานช่วงที่สอง 41

ภาพที่ 4.3 Serial monitor การทำงานของ Device A ที่สามารถแสดงข้อมูลของเซ็นเซอร์ที่ติดไว้กับทั้ง Device A และแสดงค่าที่รับมาจากเซ็นเซอร์ ที่ Device B ส่งข้อมูลมา 42

ภาพที่ 4.4 LCD 2004 การทำงานของ Device A ที่สามารถแสดงข้อมูลของเซ็นเซอร์ที่ติดไว้กับทั้ง Device A และแสดงค่าที่รับมาจากเซ็นเซอร์ ที่ Device B ส่งข้อมูลมา 42

ภาพที่ 4.5 serial monitor ที่แสดงค่าส่งออกมาจากเซ็นเซอร์เป็น float หรือ ทศนิยม 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

เรื่อง **หน้า**

ภาพที่ 4.6 การนำเซ็นเซอร์ BNO 055 เชื่อมกับบอร์ด Arduino..... 44

ภาพที่ 4.7 การเชื่อมต่อบอร์ด Arduino Mega และArduino Uno R3 เข้าด้วยกัน..... 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญ

ด้วยห้องปฏิบัติการวิจัยเอกซเรย์ซีทีและการสร้างภาพทางการแพทย์ (MIS) ได้มีการพัฒนาเครื่องดิจิทัลเอกซเรย์แบบเคลื่อนที่ และได้เริ่มพัฒนาคุณลักษณะพิเศษของเครื่องเพื่อให้ผู้ใช้งานมีความสะดวกในการใช้งานยิ่งขึ้น โดยเครื่องเอกซเรย์ดิจิทัลแบบเคลื่อนที่ต้องการความแม่นยำในการตั้งฉากรับรังสีกับหัวหลอดเอกซเรย์ เนื่องจาก การถ่ายภาพรังสีทั่วไป (conventional radiograph) นับเป็นสิ่งจำเป็นและมีความสำคัญอย่างยิ่ง ในการช่วยแพทย์วินิจฉัยโรคได้อย่างถูกต้อง แม่นยำ และรวดเร็ว

การวัดหาพิกัดแนวทิศทางการวางตัว (orientation) ของวัตถุ เป็นขั้นตอนที่มีความจำเป็นและมีความสำคัญอย่างมากในงานด้านต่างๆ อย่างหลากหลาย เช่นการหาทิศทางในการเดินทาง (navigation) ของยานอากาศที่ถูกต้อง การควบคุมหุ่นยนต์ (robotics) การติดตามการเคลื่อนไหวของบุคคลเพื่อการศึกษาและวิเคราะห์ (human motion tracking and analysis) การใช้งานเหล่านี้จำเป็นต้องสามารถทำการวัดให้ข้อมูลได้อย่างถูกต้อง และต่อเนื่องเป็นระยะเวลานาน สัญญาณข้อมูลที่ได้รับอาจส่งผ่านช่องทางการสื่อสารแบบไร้สายหรือใช้ผ่านสาย มายังอุปกรณ์ประมวลผลกลางที่มีอยู่แล้ว เช่น notebook PC คุณสมบัติต่างๆที่ต้องการในการใช้งานเหล่านี้จะต้องมี การเลือกใช้เทคโนโลยีที่เหมาะสมสอดคล้องกับความต้องการ และสิ่งที่สำคัญที่จะต้องพิจารณาอีกประการหนึ่งคือการใช้พลังงานต่ำเพื่อลดค่าใช้จ่ายในระยะยาว และราคาที่เหมาะสม เทคโนโลยีที่เกี่ยวข้องทางด้านการวัดหาพิกัดแนวทิศทางการวางตัว (orientation) ส่วนใหญ่ จะใช้เซ็นเซอร์ (Sensor) ที่เรียกว่าไอเอ็มยู (IMU : Inertial Measurement Unit) ซึ่งในตัวไอเอ็มยู ประกอบด้วย เซ็นเซอร์ที่ใช้วัดค่าความเร่ง (accelerometer) และไจโรสโคป (gyroscope) เพื่อวัดความเร็วเชิงมุม เป็นหลัก โดยอาจมีเซ็นเซอร์อื่นๆ เช่นเซ็นเซอร์วัดสนามแม่เหล็ก (magnetometer) และอุณหภูมิ ประกอบอยู่ด้วย รวมเป็นโมดูลเดียวกัน เนื่องจากไอเอ็มยูประกอบด้วยเซ็นเซอร์อย่างน้อยสองตัว และต่างชนิดกัน แต่ท้ายที่สุดแล้วสิ่งที่ต้องการทราบคือ พิกัดแนวทิศทางการวางตัว ที่วัดเทียบกับแกนอ้างอิงหลักแต่ละแกน ด้วยกายภาพแบบนี้ทำให้การนำไอเอ็มยูไปใช้งานไม่ใช่เรื่องง่ายเลย เนื่องจากเซ็นเซอร์แต่ละชนิดมีความไวต่อสัญญาณและมีค่าความคลาดเคลื่อนต่างกัน การใช้งานจึงมักจะรวมสัญญาณจากเซ็นเซอร์ต่างๆ เหล่านี้เข้าด้วยกัน(data fusion) เพื่อให้ได้ค่าที่ถูกต้องที่สุดในสภาวะต่างกัน เพราะต้องรวมเอาต์พุตจากเซ็นเซอร์ทั้งสอง/สามแบบ (data fusion) เพื่อตีความออกมาเป็นค่ามุมให้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงการนี้ เราจึงเลือกใช้อุปกรณ์ motion sensor module เข้ามาเพื่อเปรียบเทียบพิกัดแนวทิศทางการวางตัว (orientation) ของหัวหลอดเอกซเรย์กับฉากรับรังสีของเครื่องเอกซเรย์ดิจิทัลแบบเคลื่อนที่ ได้เลือกใช้เซ็นเซอร์ BNO 055 ซึ่งเป็นโมดูลที่รวมเอาเซ็นเซอร์ 3 ตัว เข้ามาอยู่รวมในชิปเดียวกัน คือ Gyroscope, Accelerometer, และ Geomagnetic หรือเรียกอีกอย่างหนึ่งว่า Magnetometer โดยเซ็นเซอร์ BNO 055 ที่เลือกใช้ มีการรวมเอาต์พุตจากเซ็นเซอร์ชนิดต่างๆ (data fusion) ภายในโมดูล โดยเราสามารถเรียกข้อมูลจากเซ็นเซอร์ได้แบบ I2C ทำให้ง่ายต่อการพัฒนาร่วมกับ microcontroller ต่าง ๆ เราเลือกใช้บอร์ดควบคุม Arduino Uno ในการสื่อสารระหว่าง motion sensor module และ คอมพิวเตอร์

วัตถุประสงค์ของโครงงาน

1.1.1 เพื่อศึกษาทฤษฎีและหลักการโมดูลเซ็นเซอร์ BNO 055 เพื่อนำไปประยุกต์ใช้ประโยชน์ในงานด้านต่างๆ

1.1.2 เพื่อที่จะสามารถออกแบบโปรแกรมและอุปกรณ์สำหรับบอกพิกัดของหัวยิงรังสีและแผ่นตรวจจับรังสีได้

1.1.3 เพื่อศึกษาทฤษฎีและหลักการของไมโครคอนโทรเลอร์ และนำไปประยุกต์ใช้ในด้านต่างๆ

ขอบเขตของโครงงาน

1.1.4 ชุดอุปกรณ์ตรวจวัดมุมระหว่างหัวยิงรังสี และแผ่นตรวจจับรังสี ของเครื่องเอกซเรย์ดิจิตอล

1.1.5 โดยใช้ไมโครคอนโทรลเลอร์ตระกูล AVR โดยมีชิป ATMEGA328P ในการพัฒนาอุปกรณ์

1.1.6 สามารถบอกพิกัดแนวทิศทางการวางตัว (orientation) ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการดำเนินโครงงาน

แผนการดำเนินโครงงานแสดงในตารางที่ 1.1

ที่	แผนการ	ระยะเวลา															
		ส.ค.				ก.ย.				ต.ค.				พ.ย			
1	ศึกษาหัวข้อโครงงาน																
2	ศึกษา ค้นคว้า ทฤษฎี ข้อมูลที่เกี่ยวข้อง																
3	เปรียบเทียบข้อดีและข้อด้อยของเซ็นเซอร์																
4	ออกแบบและทดสอบโปรแกรม																
5	ออกแบบและทดสอบอุปกรณ์																
6	นำชุดอุปกรณ์ไปทำการทดลอง																
7	เก็บรวบรวมข้อมูล วิเคราะห์ข้อมูล																
8	สรุป และอภิปราย ผลการดำเนินโครงการ																

ตารางที่ 1.1 แผนดำเนินงานโครงงาน

ประโยชน์ที่คาดว่าจะได้รับ

1.1.7 สามารถสร้างสรรค์ สิ่งประดิษฐ์โดยใช้ทักษะทางด้านวิศวกรรมและเทคโนโลยี สามารถนำใช้งานที่เป็นประโยชน์ได้จริง

1.1.8 ได้รับความรู้ความเข้าใจเกี่ยวกับการใช้งานโมดูลเซ็นเซอร์ BNO 055 ร่วมกับไมโครคอนโทรลเลอร์

1.1.9 สามารถนำความรู้ไปประยุกต์ใช้กับอุปกรณ์ทางการแพทย์ชนิดอื่นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 การถ่ายภาพรังสี

การถ่ายภาพรังสีทั่วไป (conventional radiograph) นับเป็นสิ่งจำเป็นและมีความสำคัญอย่างยิ่ง ในการช่วยแพทย์วินิจฉัยโรคได้อย่างถูกต้อง แม่นยำ และรวดเร็ว โรคบางอย่างสามารถวินิจฉัยจากภาพถ่าย รังสีได้โดยตรง โดยไม่ต้องส่งตรวจทางห้องปฏิบัติการอื่นเพิ่มเติมอีก เช่นปอดบวม (pneumonia) นิ่วในระบบทางเดินปัสสาวะ (renal calculi) เป็นต้น

การถ่ายภาพรังสีมีความสำคัญในการวินิจฉัยโรค ได้มาก เช่นการเอกซเรย์ปอด โดยดูพยาธิสภาพของปอด ตัวอย่างโรคที่ตรวจพบได้จากการเอกซเรย์ เช่น วัณโรค ถุงลมโป่งพอง เนื้องอกในปอด เป็นต้น นอกจากนี้ ยังมีบทบาทสำคัญคือ การประเมินความรุนแรงของ โรค (Assess severity) การตัดสินใจเลือกวิธีการรักษา (Determine treatment) การประเมินผลการรักษาและ ติดตามผลการรักษา (Assess response and follow up) นอกจากนี้ภาพถ่ายรังสีสามารถนำไปประกอบการวิเคราะห์หาสาเหตุของปอดบวมได้ และในสถานการณ์

ปัจจุบัน มีโรคอุบัติใหม่ของระบบทางเดินหายใจเกิดขึ้น อยู่เป็นประจำ เช่น โรคซาร์ ไข้หวัดนก และไข้หวัด 2009 เป็นต้น ซึ่งโรคเหล่านี้สามารถใช้การถ่ายภาพรังสีทรวงอก ช่วยในการวินิจฉัยได้ ปัจจุบัน การนำโปรแกรมคอมพิวเตอร์ที่ทันสมัย และสามารถประมวลผลสร้างภาพโดยเฉพาะการสร้าง ภาพด้วยเครื่องเอกซเรย์ทั่วไป มาใช้ในกลุ่มงานคือ เครื่องอ่านและแปลงสัญญาณภาพเอกซเรย์ระบบ คอมพิวเตอร์แบบ CR (Computed Radiography) ซึ่ง เครื่องเอกซเรย์ชนิดนี้มีลักษณะเดียวกับเครื่องเอกซเรย์ ทั่วไปเพียงแต่ส่วนที่เป็นตลับฟิล์ม (Cassette) จะบรรจุ แผ่นรับภาพที่เรียกว่า Imaging plate แทนแผ่นฟิล์ม และ เมื่อให้รังสีกับผู้ป่วยแล้ว นำแผ่น Imaging plate ไปเข้า เครื่องอ่านคือ Imaging reader เพื่ออ่านข้อมูลบนแผ่น เมื่ออ่านสัญญาณ ภาพในแผ่นImaging plate แล้วจะ ส่งภาพเข้าสู่ระบบคอมพิวเตอร์ คือ ระบบจัดเก็บและ สื่อสารข้อมูลการแพทย์ (PACS : Picture Archiving and Communication System) แพทย์สามารถดูภาพทาง จอคอมพิวเตอร์ต่อไป หรือจะเลือกพิมพ์ภาพฟิล์ม ออกมาแบบเดิมก็ได้

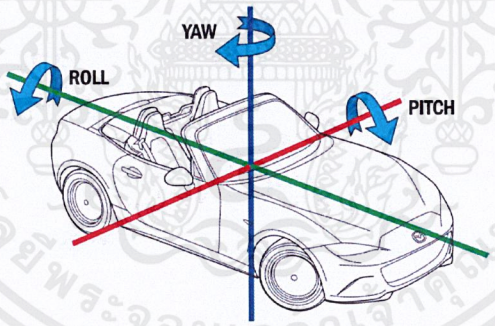
การพัฒนาภาพถ่ายรังสี ให้มีคุณภาพตามมาตรฐาน จะทำให้การแปลผลและการวินิจฉัยโรคของแพทย์ เป็นไปอย่างแม่นยำ รวดเร็ว มีประสิทธิภาพ ลดปริมาณรังสี ลดเวลา และลดค่าใช้จ่ายของโรงพยาบาล ที่เกิดจากภาพเสีย หรือต้องถ่ายภาพซ้ำ จึงมีเกณฑ์การประเมิน เพื่อใช้เป็นเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกณฑ์ประเมินระบบ ประกันคุณภาพ และควบคุมคุณภาพงานรังสีวินิจฉัย ดังกล่าวครอบคลุมประเด็นสำคัญด้านความปลอดภัย ด้านบริการรังสีวินิจฉัย ด้านคุณภาพบริการ ซึ่งเป้าหมายที่สำคัญของโรงพยาบาลคือการให้บริการที่มีคุณภาพและเป็นแนวทางการเพิ่มความพึงพอใจแก่ผู้ป่วย ต้องปรับบทบาทภารกิจและกลยุทธ์เพื่อเป็นองค์กรสมัยใหม่

2.2 การวัดหาพิกัดแนวทิศทางการวางตัว

การวัดหาพิกัดแนวทิศทางการวางตัว (Orientation) หรือบางครั้งก็เรียกว่า Attitude คือตำแหน่งการวางตัวของวัตถุในระนาบ 3 มิติ โดยนิยมบอกด้วยมุม roll, pitch และ yaw ที่มีหน่วยเป็น องศา (degree) หรือ เรเดียน (radian) (ดูภาพที่ ประกอบ) โดยมุม roll pitch และ yaw สามารถคำนวนได้จากค่าความเร่งเพียงอย่างเดียว หรือ ใช้ทั้งค่าความเร่ง (accelerometer) ค่าความเร็วเชิงมุม (gyroscope) และสนามแม่เหล็ก(magnetometer) นำมาผสม (fusion) กันก็ได้ ทั้งนี้การทำ fusion จะได้ค่า roll pitch และ yaw ที่เสถียรกว่าแบบใช้ค่าความเร่งที่วัดได้จาก accelerometer เพียงอย่างเดียว



ภาพที่ 2.1 ตำแหน่งการวางตัวของวัตถุในระนาบ 3 มิติ

2.3 เซ็นเซอร์ต่างๆที่สามารถใช้วัดหาพิกัด

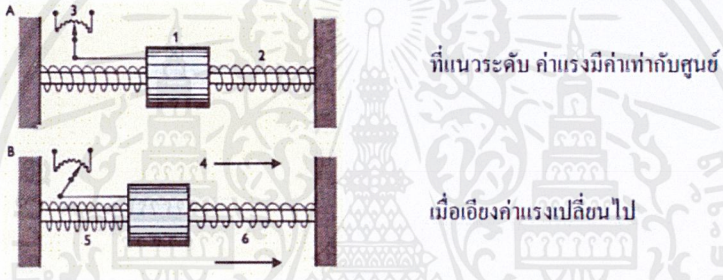
เซ็นเซอร์ที่สามารถใช้ในการตรวจสอบพิกัดได้ คือ IMU ย่อมาจาก Inertial Measurement Unit ซึ่งก็คือ หน่วยตรวจวัดความเคลื่อนไหวภายใน มีหน้าที่รายงานตำแหน่งการขยับเขยื้อนของวัตถุตามแนวการเคลื่อนไหวซึ่งปกติแล้ว วัตถุทุกชนิดจะมีแนวการเคลื่อนไหวในสามแนวแกน นั่นคือ Roll, Pitch และ Yaw โดยการตรวจจับทั้งสามแกนนั้น จะเป็นการทำงานร่วมกันของเซ็นเซอร์ทั้งสามชนิด นั่นคือ เซ็นเซอร์ตรวจวัดลักษณะการเคลื่อนไหว ด้วยตัววัดความเร่ง (Accelerometer), เซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจวัดการหมุน (Gyroscope) และ เซ็นเซอร์ระบุทิศทางด้วยการวัดสนามแม่เหล็ก (Magnetometer) ทำงานร่วมกันสามอย่าง เพื่อส่งข้อมูลออกมา

2.3.1 Accelerometer

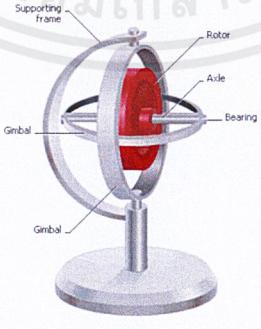
Accelerometer ทำหน้าที่วัดมุมเอียงโดยอาศัยการวัดแรงที่กระทำต่อน้ำหนัก อ้างอิงที่เกิดจากแรงโน้มถ่วงโลกซึ่งแรงโน้มถ่วงของโลกจะชี้ไปที่แกนกลางโลกเสมอ ตามกฎของนิวตัน แรง (F) = มวล (m) x อัตราเร่ง (a) โดยที่ a = 1g = 9.81 m/s^2 เรานิยมใช้ค่าเหล่านี้ในการบอกความเอียงเป็นองศา ณ ตำแหน่งนั้นทุกๆครั้ง ที่มีการหมุนหรือเอียง ตำแหน่งของชิพก็จะเปลี่ยน ส่งผลให้จุดต่างๆ บนตัวเซ็นเซอร์มีความเร่งเชิงเส้นไม่เท่ากันขึ้นอยู่กับองศาความเอียง ซึ่งความเร่งเชิงเส้นเป็นค่าที่เอาไว้เทียบกับค่า g ของโลก ดังนั้น เราจึงสามารถบอกได้ว่าตัวเซ็นเซอร์เอียงกี่องศา โดยเทียบจากค่า g นั่นเอง



ภาพที่ 2.2 เซ็นเซอร์วัดมุมเอียง

2.3.2 ไจโรสโคป (Gyroscope)

ไจโรสโคป (Gyroscope) ทำหน้าที่วัดอัตราการเปลี่ยนมุมเอียง ถ้าไม่มีการเคลื่อนที่ อัตรการเปลี่ยนแปลงมุมจะมีค่าเท่ากับศูนย์



ภาพที่ 2.3 Gyroscope

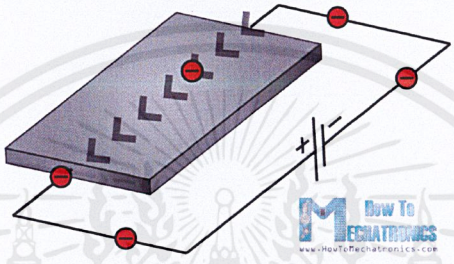
เป็นค่าที่จะเปลี่ยนแปลงก็ต่อเมื่อมีการหมุนเซ็นเซอร์ในแกนต่างๆ และค่าจะกลับมาที่ประมาณ 90 องศาเหมือนเดิม เมื่อหยุดหมุน ซึ่งจะไม่ได้ค้างค่าตำแหน่งไว้เหมือนกับ Accelerometer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก Gyroscope ทำงานโดยการวัดความเร็วเชิงมุมที่หมุนรอบแกนต่างๆของมัน ไม่มีความเกี่ยวข้องกับค่า g มันจึงทำงานเฉพาะตอนที่มีการหมุน (ความเร็วเชิงมุมไม่เป็น 0) และไม่สามารถค้างค่าไว้ได้เมื่อหยุดหมุนแล้ว

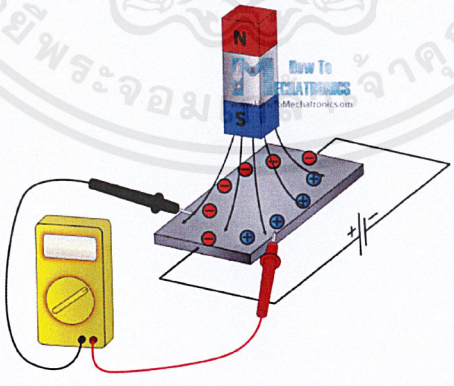
2.3.3 Magnetometer

Magnetometer เป็นเซ็นเซอร์ในกลุ่ม MEMS (Micro Electro Mechanical Systems) โดยการทำงานของเซ็นเซอร์ส่วนใหญ่ จะใช้หลักการของ Hall Effect คือ จะสร้างแผ่นนำกระแสขึ้นมาแผ่นหนึ่ง โดยหากอยู่ในภาวะปกติ กระแสจะไหลจากด้านหนึ่งไปอีกด้านหนึ่งในแนวตรง ตามรูป



ภาพที่ 2.5 การไหลของกระแส

หากเรามีแผ่นนำไฟฟ้าอย่างที่แสดงในภาพและเราตั้งค่ากระแสให้ไหลผ่านอิเล็กตรอนจะไหลตรงจากด้านหนึ่งไปอีกด้านหนึ่งของแผ่น แต่หากนำสนามแม่เหล็กมาใกล้แผ่นนำกระแสนี้ แรงแม่เหล็กจะรบกวนการเดินทางของอิเล็กตรอน และอิเล็กตรอนจะเบี่ยงเบนไปทางด้านหนึ่งของแผ่นและขั้วบวกไปยังอีกด้านหนึ่งของแผ่น แสดงว่าเราจะสามารถวัดความต่างศักย์นี้ได้ โดยใส่มิเตอร์ระหว่างทั้งสองด้านนี้เราจะได้แรงดันไฟฟ้าซึ่งขึ้นอยู่กับความแรงของสนามแม่เหล็กและทิศทางของมัน



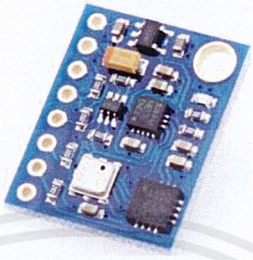
ภาพที่ 2.6 การวัดความต่างศักย์ของสนามแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 อุปกรณ์ Electronic ที่สามารถใช้ระบุพิกัดได้

อุปกรณ์ Electronic ที่สามารถใช้ระบุพิกัดได้ จำเป็นต้องใช้การรวมกันของเซ็นเซอร์ต่างๆ ที่ประกอบไปด้วยเซ็นเซอร์ที่ได้กล่าวอ้างไป ดังนั้นจึงได้ศึกษาอุปกรณ์ Electronic ที่สามารถใช้ได้ ดังนี้

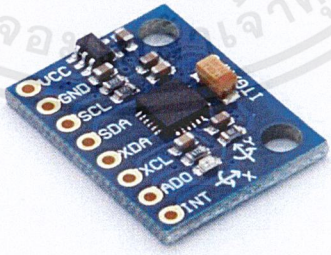
2.4.1 GY-89



ภาพที่ 2.7 เซ็นเซอร์ GY-89

GY-89 คือโมดูลเซ็นเซอร์ที่ตรวจจับการเคลื่อนไหว ที่สามารถระบุพิกัดได้ ที่มีชิพประมวลผลบนตัว 3 ชิพ คือ LSM303D, L3GD20 และ BMP180 โดย LSM303D สามารถตรวจวัดจากความเร่งเชิงเส้นแบบสามมิติดิจิตอล (3D digital linear acceleration) และแม่เหล็กแบบสามมิติดิจิตอล (3D digital magnetic sensor), L3GD20 สามารถตรวจวัดอัตราเร็วเชิงมุม แบบ 3 แกน และ BMP180 สามารถตรวจวัดความดันดิจิตอลที่มีความแม่นยำสูง ถือเป็น อุปกรณ์แบบ 10 DOF (10 Degrees of Freedom) คืออุปกรณ์ที่สามารถตรวจวัดค่าได้ทั้ง 10 แกน

2.4.2 GY-521 MPU6050

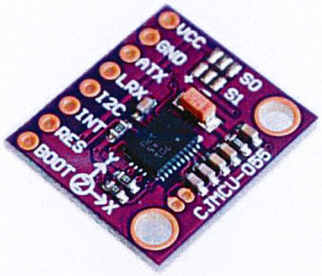


ภาพที่ 2.8 เซ็นเซอร์ GY-521 MPU6050

GY-521 MPU6050 (MPU6050 คือชื่อชิพประมวลผลบนตัว GY-521) คือโมดูลเซ็นเซอร์ที่ตรวจจับการเคลื่อนไหวและความเอียงของวัตถุ ที่สามารถระบุพิกัดได้ โดยตรวจวัดจากความเร่งเชิงเส้น (Linear Acceleration) และ ความเร็วเชิงมุม (Angular Velocity หรือใช้ Gyroscope) ถือเป็นอุปกรณ์แบบ 6 DOF (6 Degrees of Freedom) คืออุปกรณ์ที่สามารถตรวจวัดค่าได้ทั้ง 6 แกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 BNO 055



ภาพที่ 2.9 เซ็นเซอร์ BNO 055

BNO 055 คือโมดูลเซ็นเซอร์อัจฉริยะที่กำหนดทิศทางแบบสมบูรณ์ ที่สามารถระบุพิกัดได้โดยการตรวจวัดแบบใช้เซ็นเซอร์สามแกนสามตัวพร้อมกัน โดยวัดจากความเร่งเชิงเส้น (ผ่าน linear accelerometer), การเร่งความเร็วด้วยการหมุน (ผ่าน gyroscope) และความแรงของสนามแม่เหล็กภายใน (ผ่าน magnetometer) ถือเป็นอุปกรณ์แบบ 9 DOF (9 Degrees of Freedom) คืออุปกรณ์ที่สามารถตรวจวัดค่าได้ทั้ง 9 แกน และมีอัลกอริทึมการฟิวชั่นของเซ็นเซอร์ต่างๆ ภายในโมดูล

การที่ BNO 055 มีอัลกอริทึมฟิวชั่นภายในโมดูล ทำให้ IMU นี้ เป็นตัวเลือกที่เหมาะสมสำหรับโครงงานนี้ เนื่องจากช่วยในการคำนวณหาพิกัดได้ง่ายขึ้น

2.5 ทฤษฎีเกี่ยวกับไมโครคอนโทรเลอร์

2.5.1 ความหมายของไมโครคอนโทรเลอร์

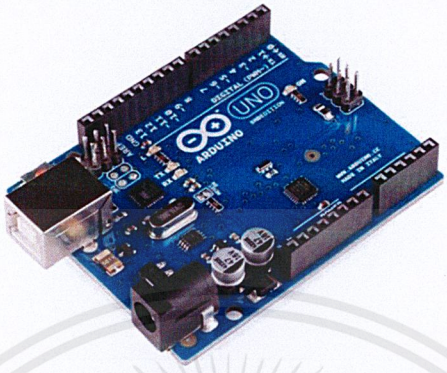
ไมโครคอนโทรเลอร์ (Microcontroller) คือ อุปกรณ์ประเภทสารกึ่งตัวนำที่รวบรวมฟังก์ชันการทำงานต่างๆไว้ภายในตัวของมันเอง โดยมีโครงสร้างใกล้เคียงกับคอมพิวเตอร์ คือ ภายในประกอบด้วยหน่วยรับข้อมูล และโปรแกรม หน่วยประมวลผล หน่วยความจำ หน่วยแสดงผล ซึ่งส่วนประกอบเหล่านี้มีความสมบูรณ์ในตัวของมันเอง ทำให้มีขนาดเล็ก และสามารถเขียนโปรแกรมควบคุมการทำงานของอุปกรณ์ต่างๆ ที่เชื่อมต่อกับตัวมันง่ายต่อการนำไปประยุกต์ใช้งาน

ปัจจุบันเทคโนโลยีไมโครคอนโทรลเลอร์ ได้มีการพัฒนาให้มีความสามารถในการประมวลผลข้อมูลได้รวดเร็วและมีราคาไม่แพง ง่ายต่อการศึกษาและออกแบบ สามารถเขียนโปรแกรมชุดคำสั่งได้หลายภาษา ตลอดจนมีโปรแกรมช่วยสนับสนุนจำนวนมาก เพื่อช่วยในการออกแบบพัฒนาระบบจึงทำให้ไมโครคอนโทรลเลอร์ มีการมาใช้งานอย่างแพร่หลายในวงจรอิเล็กทรอนิกส์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์มีหลายตระกูล ให้เลือกใช้ตามความถนัด เช่น ตระกูล AVR ตระกูล PIC ตระกูล MCS-51 ตระกูล Z-80 และตระกูล Arduino เป็นต้น

2.5.2 บอร์ด Arduino UNO R3

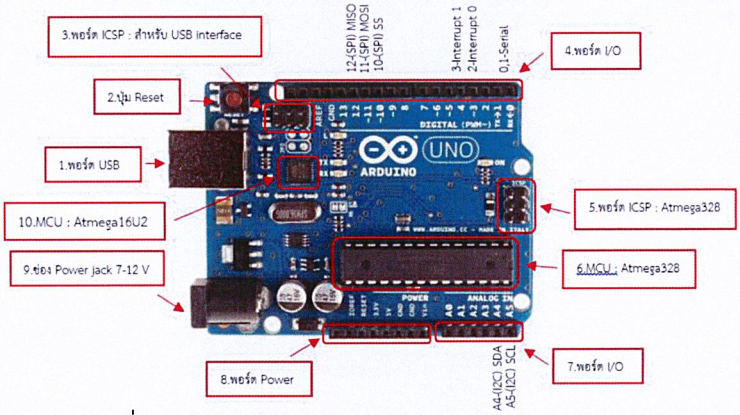


ภาพที่ 2.10 บอร์ด Arduino UNO R3

โครงงานนี้เลือกใช้ไมโครคอนโทรเลอร์ตระกูล Arduino ซึ่งเป็นไมโครคอนโทรเลอร์บอร์ดแบบสำเร็จรูป โดยมี Open Source ที่สามารถนำไปพัฒนาต่อเป็นอุปกรณ์ต่างๆได้ และความสามารถในการเพิ่ม Boot Loader เข้าไป จึงทำให้การ Upload Code เข้าตัวบอร์ดสามารถทำได้ง่ายขึ้น และยังมีการพัฒนา Software ที่ใช้ในการควบคุมตัวบอร์ดของ Arduino เป็น ภาษา C++ ที่โปรแกรมเมอร์ส่วนใหญ่มีความคุ้นเคยในการใช้งาน

Arduino เป็น platform ของ I/O บอร์ดอย่างง่ายๆ ที่มี I/O ขั้นพื้นฐานที่พอเพียงกับการใช้งาน และ การเรียนรู้โดยตัวบอร์ดจะมาพร้อมกับชุดคำสั่งที่ใช้ควบคุม port I/O ไม่ว่าจะเป็น port digital, port analog, PWM และ Serial port ซึ่ง Arduino ออกแบบจากไมโครคอมพิวเตอร์ชิพเดี่ยว และมีโปรแกรมพัฒนาสำหรับเขียนโปรแกรมให้บอร์ด สามารถประยุกต์ทำเป็นตัวควบคุมการทำงานของอุปกรณ์ต่างๆ มากมาย Arduino เป็นภาษาอิตาลี ซึ่งใช้เป็นชื่อของโครงการพัฒนา

2.4.2.1 ส่วนประกอบของบอร์ด Arduino UNO R3

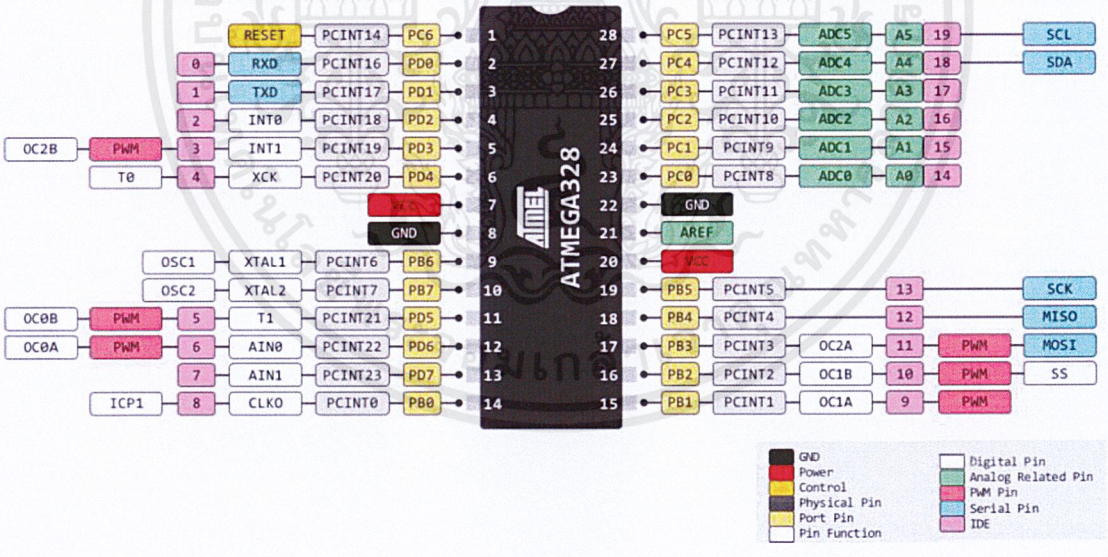


ภาพที่ 2.11 ส่วนประกอบของบอร์ด Arduino UNO R3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. **USB Port:** ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2. **Reset Button:** เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3. **ICSP Port** ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4. **I/O Port:** Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx,Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
5. **ICSP Port:** Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. **MCU:** Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7. **I/OPort:** นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
8. **Power Port:** ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอกประกอบด้วยขา ไฟเลี้ยง +3.3 V, +5V, GND, V_{in}
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

2.5.3 ATmega328



ภาพที่ 2.12 โครงสร้างไมโครคอนโทรเลอร์ ATmega328

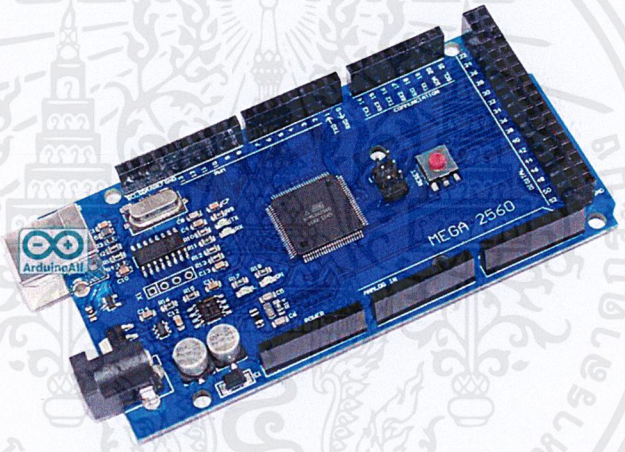
ATMEGA328P เป็นไมโครคอนโทรเลอร์ที่ใช้ในบอร์ด Arduino เป็นซีพียูแบบ RISC (Reduced Instruction Set Computer) ขนาด 8 บิต มีคุณสมบัติเด่นดังนี้

- หน่วยความจำโปรแกรมแบบ FLASH ขนาด 32 กิโลไบต์
- หน่วยความจำข้อมูลแบบ SRAM ขนาด 2 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หน่วยความจำข้อมูลแบบ EEPROM ขนาด 1 กิโลไบต์
- สนับสนุนการเชื่อมต่อแบบ I2C bus
- พอร์ตอินพุตเอาต์พุตจำนวน 23 บิต
- วงจรสื่อสารอนุกรม
- วงจรนับ/จับเวลาขนาด 8 บิตจำนวน 2 ตัวและขนาด 16 บิตจำนวน1 ตัว
- สนับสนุนช่องสัญญาณสำหรับสร้าง Pulse Width Modulation (PWM) จำนวน 6 ช่อง
- วงจรแปลงอนาลอกเป็นดิจิตอลขนาด10 บิตในตัวจำนวน8 ช่อง
- ทำงานได้ตั้งแต่ย่านแรงดัน 1.8-5.5 Volts
- ความถี่ใช้งานสูงสุด 20 MHz

2.5.4 บอร์ด Arduino Mega 2560



ภาพที่ 2.13 บอร์ด Arduino Mega 2560

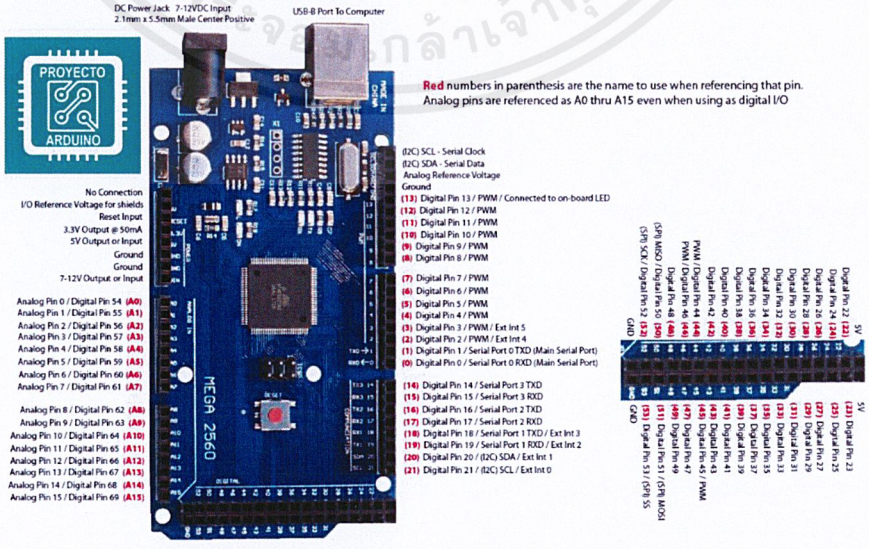
บอร์ด Arduino Mega 2560 จะเหมือนกับ Arduino Mega ADK ต่างกันตรงที่บนบอร์ดไม่มี USB Host มาให้ การโปรแกรมยังต้องทำผ่านโปรโตคอล UART อยู่บนบอร์ดใช้ชิปไอซีไมโครคอนโทรเลอร์เบอร์ ATmega2560 เป็นบอร์ด Arduino ที่ออกแบบมาสำหรับงานที่ต้องใช้ IO มากกว่า Arduino Uno R3 เช่น งานที่ต้องการรับสัญญาณจาก Sensor หรือควบคุมมอเตอร์ Servo หลายๆตัว ทำให้ Pin IO ของบอร์ด Arduino Uno R3 ไม่สามารถรองรับได้ ทั้งนี้บอร์ด Mega 2560 R3 ยังมีความหน่วยความจำแบบ Flash มากกว่า Arduino Uno R3 ทำให้สามารถเขียนโค้ดโปรแกรมเข้าไปได้มากกว่า ในความเร็วของ MCU ที่ เท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจำเพาะ

ชิปไอซีไมโครคอนโทรเลอร์	ATmega2560
ใช้แรงดันไฟฟ้า	5 V
รองรับการจ่ายแรงดันไฟฟ้า (ที่แนะนำ)	7 – 12 V
รองรับการจ่ายแรงดันไฟฟ้า (ที่จำกัด)	6 – 20 V
พอร์ต Digital I/O	54 พอร์ต (มี 15 พอร์ต PWM output)
พอร์ต Analog Input	16 พอร์ต
กระแสไฟฟ้ารวมที่จ่ายได้ในทุกพอร์ต	40 mA
กระแสไปที่จ่ายได้ในพอร์ต 3.3V	50 mA
พื้นที่โปรแกรมภายใน	256 KB แต่ 8 KB ถูกใช้โดย Boot Loader
พื้นที่แรม	8 KB
พื้นที่หน่วยความจำถาวร (EEPROM)	4 KB
ความถี่คริสตัล	16 MHz

2.4.3.1 ส่วนประกอบของบอร์ด Arduino Mega 2560

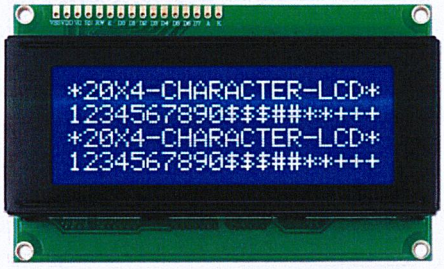


ภาพที่ 2.14 ส่วนประกอบของบอร์ด Arduino Mega 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 อุปกรณ์ที่ใช้ในการแสดงผลในโครงงาน

2.6.1 LCD



ภาพที่ 2.15 LCD

คำว่า LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตอลเหลว หลักการคือด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า Backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึก ก็จะทำให้ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ Backlight แสดงขึ้นมาบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้ จะมีสีที่แตกต่างกันตามสีของผลึกคริสตอล เช่น สีเขียว หรือ สีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือสีขาว แล้วพบกับพื้นหลังสีต่างๆกัน

จอ LCD จะแบ่งเป็น 2 แบบใหญ่ๆตามลักษณะการแสดงผลดังนี้

1. Character LCD เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัว เช่น จอ LCD ขนาด 16x2 หมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 16 ตัว และมีทั้งหมด 2 บรรทัดให้ใช้งาน ส่วน 20x4 จะหมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 20 ตัว และมีทั้งหมด 4 บรรทัด

2. Graphic LCD เป็นจอที่สามารถกำหนดได้ว่าจะให้แต่ละจุดบนหน้าจอกั้นแสง หรือปล่อยแสงออกไป ทำให้จอนี้สามารถสร้างรูปขึ้นมาบนหน้าจอได้ การระบุขนาดจะระบุในลักษณะของจำนวนจุด (Pixels) ในแต่ละแนว เช่น 128x64 หมายถึงจอที่มีจำนวนจุดตามแนวนอน 128 จุด และมีจุดตามแนวตั้ง 64 จุด

การเชื่อมต่อกับจอ Character LCD

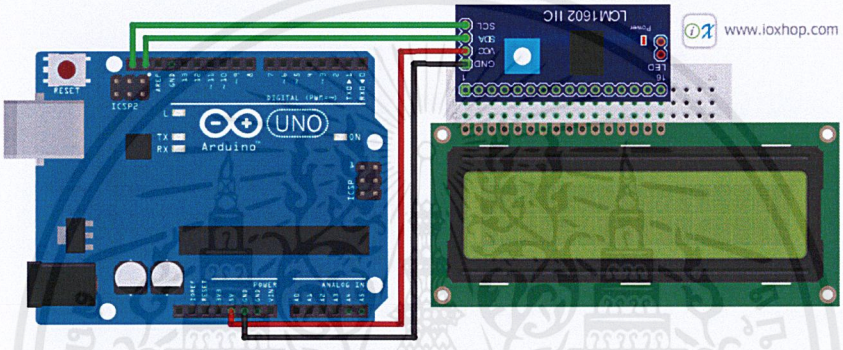
การเชื่อมต่อจะมีด้วยกัน 2 แบบ คือ

- การเชื่อมต่อแบบขนาน - เป็นการเชื่อมต่อจอ LCD เข้ากับบอร์ด Arduino โดยตรง โดยจะแบ่งเป็นการเชื่อมต่อแบบ 4 บิต และการเชื่อมต่อแบบ 8 บิต ใน Arduino จะนิยมเชื่อมต่อแบบ 4 บิต เนื่องจากใช้สายในการเชื่อมต่อน้อยกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเชื่อมต่อแบบอนุกรม - เป็นการเชื่อต่อกับจอ LCD ผ่านโมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I^2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I^2C ทำให้ใช่สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่างๆออกมาได้

ในโครงงานนี้เราเลือกใช้การเชื่อมต่อแบบอนุกรม (LCD I^2C) การเชื่อมต่อแบบอนุกรม จะใช้งานโมดูล I^2C Serial Interface Board Module มาเชื่อมต่อระหว่าง Arduino กับจอ LCD วงจรที่เชื่อมต่อจะเป็นดังนี้



ภาพที่ 2.16 การเชื่อมต่อ LCD กับ Arduino

2.6.2 Buzzer

ภาพที่ 2.17 Buzzer

ลำโพงบัซเซอร์เป็นอุปกรณ์ที่ให้กำเนิดเสียงทำหน้าที่เปลี่ยนสัญญาณไฟฟ้าให้อยู่ในรูปสัญญาณเสียง ลำโพงบัซเซอร์มีอยู่ 2 ประเภท ได้แก่

1. แบบแอคทีฟ (Active Buzzer) ลำโพงชนิดนี้มีวงจรกำเนิดความถี่อยู่ภายใน สามารถสร้างสัญญาณเสียงเตือนได้ทันทีเพียงแค่จ่ายแรงดันไฟฟ้าเข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. แบบพาสซีฟ (Passive Buzzer) ลำโพงชนิดนี้ทำงานเหมือนลำโพงขนาดเล็ก คือ ถ้าป้อนแรงดันไฟฟ้ากระแสตรงเข้าไปไม่มีเสียงถ้าต้องการให้มีสัญญาณเสียงต้องทำการป้อนสัญญาณความถี่เข้าไป ลำโพงชนิดนี้สามารถกำเนิดเสียงที่มีความแตกต่างกันตามความถี่ที่ป้อนเข้ามา

ในโครงงานนี้เป็นการใช้งาน Arduino กับลำโพงบัซเซอร์แบบพาสซีฟ ดังนั้นการใช้งานต้องทำการ เขียนโปรแกรมเพื่อส่งความถี่จาก Arduino เข้าไปยังลำโพงบัซเซอร์ ปัจจุบันเครื่องใช้ไฟฟ้าโดยทั่วไปมีลำโพงบัซเซอร์ติดตั้งอยู่ภายในด้วย เช่น ในคอมพิวเตอร์ใช้ลำโพงบัซเซอร์เพื่อส่งสัญญาณให้ทราบว่าสถานะของคอมพิวเตอร์มีปัญหาอะไร หรือในเครื่องซักผ้าอัตโนมัติ เครื่องปรับอากาศ เมื่อทำการกดปุ่มบนเครื่องหรือรีโมตคอนโทรลจะได้ยินเสียงปืนดังขึ้นมาด้วยเพื่อบอกให้รู้ว่าขณะนี้ทำการกด สวิตช์แล้ว

2.6.3 LED

ภาพที่ 2.18 LED

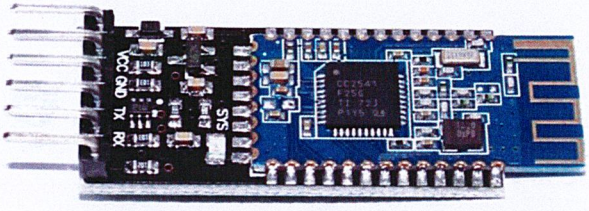
LED คือไดโอดเปล่งแสง ย่อมาจากคำว่า (Light-Emitting Diode) ซึ่งสามารถเปล่งแสงออกมาได้แสงที่เปล่งออกมาประกอบด้วยคลื่นความถี่เดียวและ เฟสต่อเนื่องกัน ซึ่งต่างกับแสงธรรมดาที่ตาคนมองเห็น โดย หลอด LED สามารถเปล่งแสงได้เมื่อจ่ายกระแสไฟฟ้าเข้าเพียงเล็กน้อยเท่านั้นและประสิทธิภาพในการให้แสงสว่างก็ยังดีกว่าหลอดไฟขนาดเล็กทั่วๆ ไป

ในโครงงานนี้เป็นการใช้งาน Arduino กับ LED ในการแสดงผลเมื่อค่าพิกัดที่ได้รับมามีค่าตรงกับที่บันทึกไว้ใน EEPROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 อุปกรณ์ที่ใช้ในการสื่อสารในโครงงาน

2.7.1 บลูทูธ HM-10 BLE 4.0



ภาพที่ 2.19 บลูทูธ HM-10 BLE 4.0

Bluetooth คือ มาตรฐานการสื่อสารแบบไร้สาย สำหรับการรับ/ส่งข้อมูลขนาดเล็กภายในข่ายงานส่วนบุคคล (Personal Area Network, PAN) เริ่มการพัฒนาโดย Ericsson ตั้งแต่ปี ค.ศ. 1994 จนมาถึงปัจจุบันเป็นรุ่นที่ 5

Bluetooth Low Energy (BLE บางแห่งเรียก Bluetooth Smart) ถูกออกแบบมาเพื่อความประหยัด (low power, low bandwidth, low cost และ low complexity) เป็นส่วนหนึ่งของมาตรฐาน Bluetooth 4.0 ปัจจุบันแนวคิดการสร้างระบบ Internet Of Things (IoT) ได้เติบโตขึ้นมาและมีการนำเอา BLE มาติดตั้งกับอุปกรณ์อิเล็คทรอนิกส์อื่น เช่น Raspberry Pi, Arduino เป็นต้น ทำให้เกิดการเชื่อมโยงจากข่ายงาน PAN Bluetooth ออกไปยังข่ายงาน IoT ด้วย

คุณสมบัติของอุปกรณ์ Bluetooth 4.0 BLE

- เชื่อมต่อสื่อสาร รับ-ส่งข้อมูลสูงสุด 110 เมตรในพื้นที่โล่ง
- ใช้ความถี่ 2.4GHz
- Modulation : GFSK (Gaussian Frequency Shift Keying)
- Sensitivity : < -84dBm ที่ 0.1% BER
- อัตราการรับส่ง : Asynchronous 6 kbps, Synchronous 6kbps
- ความปลอดภัยในการรับส่ง : Authentication and encryption
- Support Service : Central & Peripheral UUID FFE0, FFE1
- Power consumption : โหมด sleep อัตโนมัติ และ โหมด stand by กินกระแสอยู่ที่ 400 ไมโครแอมป์ ถึง 1.5 มิลลิแอมป์, ขณะรับส่งข้อมูล กินกระแส 8.5 มิลลิแอมป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ไฟเลี้ยงโมดูล : +3.3VDC 50 มิลลิแอมป์
- ขนาด : 26.9mm x 13mm x 2.2mm
- Bluetooth Certification : ROHS REACH BQB

AT Commanders ของบลูทูธ

- AT (Test command)
- AT+BAUD (Query/Set Baud rate)
- AT+CHK (Query/Set parity)
- AT+STOP (Query/Set stop bit)
- AT+UART (Query/Set uart rate,parity, stop bit)
- AT+PIO (Query/Set PIO pins status Long command)
- AT+PIO (Query/Set a PIO pin sttus Short command)
- AT+NAME (Query/Set device friendly name)
- AT+PIN (Query/Set device password code)
- AT+DEFAULT (Reset device settings)
- AT+RESTART (Restart device)
- AT+ROLE (Query/Set device mode, Master or Slave)
- AT+CLEAR (Clear remote device address if has)
- AT+CONLAST (Try to connect last connect succeed device)
- AT+VERSION (Show software version information)
- AT+HELP (Show help information)
- AT+RADD (Query remote device address)
- AT+LADD (Query self address)
- AT+IMME (Query/Set Whether the device boot immediately)
- AT+WORK (if device not working, start work, use with AT+IMME command)
- AT+TCON (Query/Set Try to connect remote times)
- AT+TYPE (Query/Set device work type, transceiver mode or remote mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- AT+START (Switch remote control mode to transceiver mode)
- AT+BUFF (Query/Set How to use buffer data, Duing mode switching time)
- AT+FILT (Query/Set device filter when device searching) A
- T+COD (Query/Set Class of Device. eg: phone, headset etc.)

2.8 การสื่อสารของอุปกรณ์และไมโครคอนโทรเลอร์

การสื่อสารระหว่าง Microcontroller กับอุปกรณ์ภายนอกต่างๆ สามารถแบ่งได้สองชนิดคือ

- การสื่อสารแบบขนาน (Parallel Communication) ทำได้โดยการส่งข้อมูลจากผู้ส่ง (Transmitter) ไปยังผู้รับ (Receiver) ออกมาทีละ 1 Byte หรือ 8 Bits ซึ่งอาจมี Bit เพิ่มเติมสำหรับควบคุมการสื่อสาร ซึ่งมีข้อดีคือ อัตราการรับส่งข้อมูลสูง ตัวอย่างการใช้งาน STM32F4DISCOVERY กับการสื่อสารแบบขนาน เช่น การเชื่อมต่อจอแสดงผล LCD การเชื่อมต่อกล้อง และการเชื่อมต่อกับหน่วยความจำเสริม (External RAM Memory) ตัวกลางระหว่างอุปกรณ์ทั้งสองเครื่องต้องใช้กลุ่มสายส่งเป็นจำนวนมาก (Bus) จึงไม่เหมาะสำหรับงานที่ต้องใช้อุปกรณ์ภายนอกเป็นจำนวนมาก
- การสื่อสารแบบอนุกรม (Serial Communication) เป็นการรับส่งออกมาทีละ Bit จึงมีความล่าช้ากว่าการสื่อสารแบบขนาน อย่างก็ไรตาม ตัวกลางสำหรับการสื่อสารอาจจะใช้สายส่งเพียงคู่เดียว จึงมีค่าใช้จ่ายน้อยกว่าและทำให้อุปกรณ์มีขนาดเล็ก นอกจากนี้ การสื่อสารแบบอนุกรมสามารถสื่อสารแบบเครือข่ายได้ จึงเป็นที่นิยมมากกว่า

Arduino มีโมดูลการสื่อสารหลายชนิด ได้แก่ UART, I2C, SPI, USB เป็นต้น ในบทนี้จะกล่าวถึงโมดูลการสื่อสารแบบ UART และ I2C

2.8.1 UART

UART ย่อมาจาก **U**niversal **A**synchronous **R**eceiver **T**ransmitter ใช้การสื่อสารแบบอนุกรมโดยเป็นแบบอะซิงโคนัสใช้สายสัญญาณเพียง 2 เส้น คือ Tx และ Rx ในการรับ – ส่งข้อมูล โดยเส้น Tx จะเป็นเส้นที่ใช้ส่งข้อมูล และเส้น Rx เป็นเส้นที่ใช้รับข้อมูล เมื่อนำไปต่อใช้งานจะต้องต่อไขว้กันระหว่างอุปกรณ์และไมโครคอนโทรลเลอร์ และต้องกำหนดความเร็วในการสื่อสารซึ่งสามารถเลือกได้ดังนี้ 300 600 1200 2400 4800 9600 14400 19200 28800 38400 57600 และ 115200 ทั้งนี้ความเร็วที่กำหนดมักขึ้นอยู่กับอุปกรณ์ที่จะเชื่อมต่อด้วยโดยมักจะกำหนดเป็น 9600 เมื่ออุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดเป็น 9600 ที่ตัวไมโครคอนโทรลเลอร์ต้องกำหนดเป็น 9600 ด้วย จึงจะสามารถรับ-ส่งข้อมูลได้ถูกต้อง

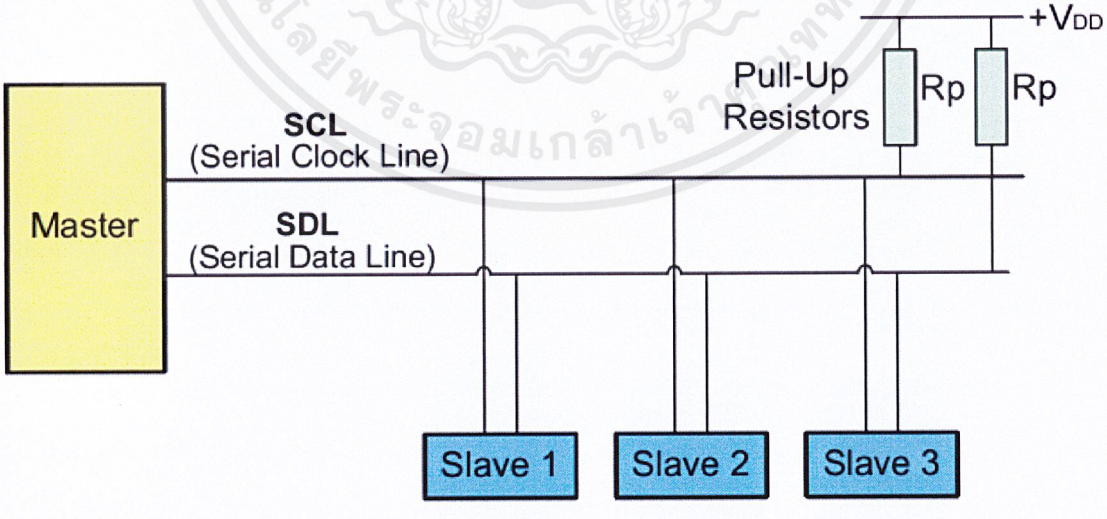
อุปกรณ์ที่ใช้ UART ในการสื่อสารที่เห็นได้บ่อย ๆ คือ GPS, GSM/3G/4G Module, Bluetooth รวมไปถึงการสื่อสารกับคอมพิวเตอร์ ในการอัพโหลดโปรแกรมโดยมีชิปไอซีแปลง USB เป็น UART ทำให้สามารถใช้งาน UART ผ่าน USB เพื่ออัพโหลดโปรแกรมได้

2.8.2 I2C

I2C ย่อมาจาก Inter-IntegratedCircuit คิดค้นโดย Philip semiconductor หรือ NXP semiconductor จุดประสงค์เพื่อรับส่งข้อมูลความเร็วต่ำระหว่างอุปกรณ์ ต่างๆ เช่นเครื่องบันทึกเสียง โทรศัพท์มือถือ รวมถึง หน่วยความจำอย่าง EEPROM เป็นต้น

การสื่อสารแบบ I2C มีจุดเด่นคือ การเชื่อมต่อกันเป็นระบบบัสและรับส่งข้อมูลระหว่างอุปกรณ์ได้เป็นจำนวนมาก โดยใช้สายสัญญาณเพียงสองเส้นดังภาพ ซึ่งสามารถลดจำนวนสายสัญญาณ และขนาดของอุปกรณ์ จึงเป็นที่นิยม

การสื่อสาร I2C ใช้สายส่งดิจิตอลแบบ Bidirectional Open-drain line ซึ่งสามารถรับส่งข้อมูลได้ในเส้นเดียวกับ จำนวน 2 ชุด คือ Serial Data Line (SDA) ใช้ทำหน้าที่ส่งข้อมูล และ Serial Clock Line (SCL) ทำหน้าที่ส่งสัญญาณนาฬิกา ความเร็วของการรับส่งข้อมูลนั้นขึ้นอยู่กับความถี่ของสัญญาณนาฬิกาในสาย SCL



ภาพที่ 2.20 การเชื่อมต่ออุปกรณ์ แบบ I2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากสายสัญญาณของอุปกรณ์เป็นแบบ open-drain ซึ่งมีแรงดันไฟเท่ากับ GND เมื่อสถานะของโลจิกเป็นศูนย์ และเป็น High impedance เมื่อสถานะโลจิกเป็นหนึ่ง จึงจำเป็นต้องต่อ Pulled-up resistors และมีไฟเลี้ยงภายนอก V_{DD} (3.3V หรือ 5V)

จากภาพ แสดงตัวอย่างการเชื่อมต่ออุปกรณ์แบบ I2C บัส โดยมีอุปกรณ์รับส่งข้อมูลหลัก (Master) และมีอุปกรณ์ย่อย (Slave) จ่านวน 3 ชุด มาต่อพ่วง เมื่อเริ่มต้นการทำงาน อุปกรณ์ Master จะระบุคำสั่งว่าต้องการอ่านข้อมูล (Read) หรือส่งข้อมูล (Write) ไปยังอุปกรณ์ Slave การระบุอุปกรณ์ Slave ที่ต้องการจะสื่อสาร จะใช้ Address 7 บิต หรือ 10 บิต ซึ่งระบุไว้ใน Datasheet ของผู้ผลิต และกำหนดโดยผู้ใช้งานได้ทั้ง Hardware หรือ Software

การสื่อสารแบบ I2C มีรูปแบบดังรูป 1-2 และมีขั้นตอนการรับส่งข้อมูลเป็นลำดับ ดังนี้

1. เพื่อเริ่มต้นสถานะการสื่อสารแบบ I2C อุปกรณ์ Master เริ่มจะส่งคำสั่ง START ซึ่งจะทำให้อุปกรณ์ Slave ทุกตัวที่อยู่ใน Bus เตรียมพร้อมรับข้อมูลจากสายส่งข้อมูล (SDA)
2. อุปกรณ์ Master ส่ง Address ขนาด 7 บิตและตามด้วยคำสั่งอ่านหรือเขียน (Read/Write) อีก 1 บิต
3. อุปกรณ์ Slave ทุกตัวใน Bus จะตรวจสอบ Address ในสายส่งข้อมูลว่าตรงกับ Address ของตน หรือไม่หากตรงกัน จะส่งสัญญาณ ACK ขนาด 1 บิต กลับไปยังอุปกรณ์ Master เพื่อเตรียมพร้อมดำเนินการต่อไป
4. การรับส่งข้อมูลระหว่าง Master และ Slave ที่ระบุ Address จะด่าเนินการอย่างต่อเนื่อง ในกรณีที่ Master ส่งคำสั่ง Read อุปกรณ์ Slave จะส่งข้อมูลเป็นชุดๆ ชุดละ 8 บิต (1 ไบต์) เมื่ออุปกรณ์ Master จะส่งสัญญาณ ACK เมื่อได้รับทุกๆ ไบต์
5. อุปกรณ์ Master จะส่งคำสั่ง STOP เพื่อสิ้นสุดสถานะการสื่อสารแบบ I2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีดำเนินงานโครงงาน

โครงงานนี้จัดทำขึ้นเพื่อพัฒนาชุดอุปกรณ์ตรวจวัดมุมระหว่างหัวยิงรังสี และแผ่นตรวจจับรังสีของเครื่องเอกซเรย์ดิจิตอล แบบ DR (Digital Radiography) โดยการตรวจสอบมุมจากโมดูลเซ็นเซอร์ แล้วนำค่าที่ได้มาปรับตำแหน่งของหัวยิงรังสี และแผ่นตรวจจับรังสี เพื่อให้อยู่ในตำแหน่งที่ตั้งฉากกัน โดยขั้นตอนการดำเนินโครงงานจะประกอบด้วย การออกแบบและสร้างอุปกรณ์ การเขียนโปรแกรม และการออกแบบโปรแกรมของชุดอุปกรณ์

โดยโครงงานนี้ เราเลือกใช้อุปกรณ์ motion sensor module เข้ามาเพื่อเปรียบเทียบพิกัดของหัวหลอดเอกซเรย์กับฉากรับรังสีของเครื่องเอกซเรย์ดิจิทัลแบบเคลื่อนที่ เราเลือกใช้อุปกรณ์ IMU BNO 055 ซึ่งเป็นโมดูลที่รวมเอาเซ็นเซอร์ต่าง ๆ เข้ามาอยู่รวมกัน คือ Accelerometer, Gyroscope และ Magnetometer โดยเราสามารถเรียกข้อมูลจากเซ็นเซอร์ได้แบบ I2C ทำให้ง่ายต่อการพัฒนาร่วมกับ microcontroller ต่างๆ เราเลือกใช้บอร์ดควบคุม Arduino Uno ในการสื่อสารระหว่าง motion sensor module และ คอมพิวเตอร์ Arduino Uno เป็นบอร์ด Microcontroller ตระกูล Arduino ชนิดหนึ่งซึ่ง Arduino มีแพลตฟอร์มที่สามารถใช้งานได้ง่ายและมีผู้พัฒนาเยอะทำให้มีไลบรารี่ที่หลากหลาย มีความยืดหยุ่นทางด้านSoftwareเพราะเปิดได้ทั้งบน Mac, Windows, และ Linux สามารถพัฒนาเพื่อเชื่อมต่อกับอุปกรณ์หรือเซ็นเซอร์ต่าง ๆ ผ่าน Pins บนตัวบอร์ด เหมาะสำหรับผู้เริ่มต้นจะศึกษาเขียนโปรแกรมและหุ่นยนต์

หลังจากได้ค่าจากเซ็นเซอร์ทั้งสาม เราจะเขียนโปรแกรมเพื่อนำค่าเหล่านั้นมาคำนวณเพื่อหาพิกัดการหมุนในสามแนวแกน (roll, pitch, yaw) ซึ่ง roll คือการหมุนรอบแกน x ที่พุ่งออกจากหน้าของเรา, pitch คือการหมุนรอบแกน y ที่พุ่งจากด้านขวาไปด้านซ้ายของเรา, และ yaw คือการหมุนรอบแกน z ที่พุ่งจากล่างขึ้นบน หลังจากนั้น เราจะนำค่าพิกัดการหมุน (roll, pitch, yaw) ที่ได้มาคำนวณหาค่าความคลาดเคลื่อนในรูปแบบต่างและสรุปผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 อุปกรณ์ที่ใช้ในการดำเนินงานโครงงาน

อุปกรณ์ต่างๆ ที่ใช้ในอุปกรณ์ตรวจวัดมุมระหว่างหัวยิงรังสี และแผ่นตรวจจับรังสี จะต้องกำหนดขนาดให้เหมาะสมกับอุปกรณ์แต่ละชนิดที่ใช้ และให้สามารถบรรจุลงกล่องเอนกประสงค์ ซึ่งในกล่องอุปกรณ์เป็น 2 กล่อง คือ Device A และ Device B และมีอุปกรณ์ที่ใช้ทั้งหมด ดังนี้

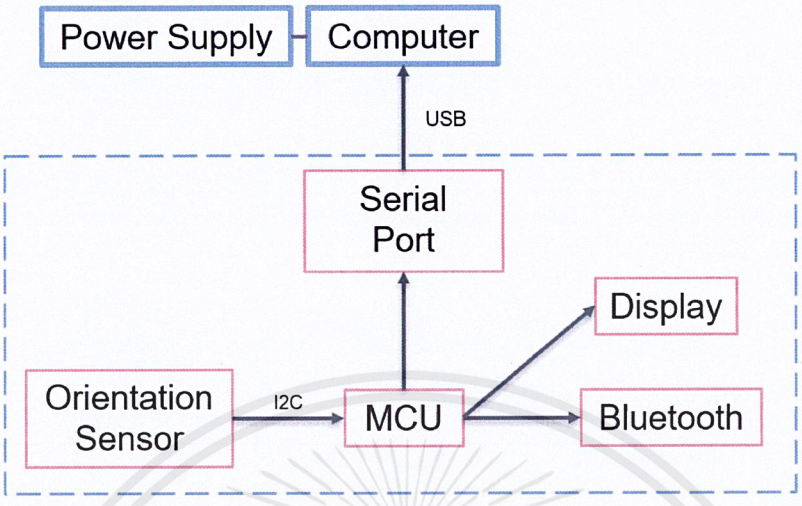
1) กล่องเอนกประสงค์ 2 กล่อง สำหรับบรรจุอุปกรณ์ต่างๆลงไปในกล่อง
2) บอร์ด Arduino UNO R3 1 บอร์ด สำหรับ Device B
3) บอร์ด Arduino Mega 1 บอร์ด สำหรับ Device A
4) เซ็นเซอร์ BNO 055 2 ตัว สำหรับบอกค่ามุมพิกัด ในรูปแบบของ Euler's angle
5) เซ็นเซอร์บลูทูธ HM-10 BLE 4.0 2 ตัว สำหรับสื่อสารข้อมูลระหว่าง Device A และ Device B
6) โพรโทบอร์ด 400 holes 2 ชิ้น สำหรับวางอุปกรณ์ทุกชนิดที่ต้องใช้ใน Device A และ Device B
7) จอแสดงผล LCD 2004 1 ชิ้น สำหรับแสดงผลค่ามุมพิกัด ในรูปแบบของ Euler's angle ทั้งข้อมูลของ Device A และDevice B
8) Buzzer 1 ตัว สำหรับเตือน เมื่อค่ามุมพิกัดขนานกัน ตามที่ได้กำหนดเงื่อนไขไว้ โดยติดไว้ที่ Device A
9) Button 1 ตัว สำหรับบันทึกข้อมูลค่ามุมพิกัด ที่เป็นค่าที่เหมาะสมสำหรับใช้ในการถ่ายภาพรังสี โดยติดไว้ที่ Device A
10) LED 1 ตัว สำหรับบอกเมื่ออุปกรณ์อยู่ที่ตำแหน่ง ที่เป็นค่ามุมพิกัดที่ได้บันทึกไว้ โดยติดไว้ที่ Device A
11) สายไฟจัมเปอร์ สำหรับเชื่อมอุปกรณ์ทุกชนิดกับบอร์ดไมโครคอนโทรเลอร์ทั้ง Device A และDevice B
12) ชิ้นงาน 3D print สำหรับปรับองศา เพื่อทำการทดสอบองศาของอุปกรณ์

3.2 การออกแบบและสร้างอุปกรณ์

โดยในการออกแบบสร้างอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสีของเครื่องเอกซเรย์ดิจิทัลแบบเคลื่อนที่ โดยการใช้บอร์ด Arduino รับพิกัดการหมุนในแนวสามแกน (roll, pitch, yaw) จากเซ็นเซอร์ BNO 055 และนำไปติดไว้กับที่หัวหลอดเอกซเรย์ ซึ่งเรียกว่า Device A กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

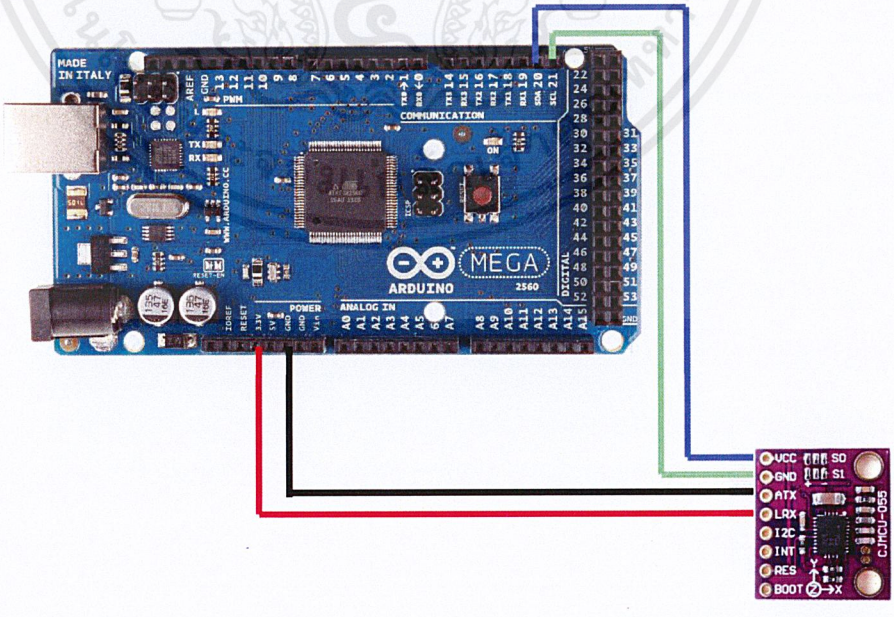
ฉากรับรังสี ซึ่งเรียกว่า Device B โดยสื่อสารกันผ่านเซ็นเซอร์บลูทูธเพื่อส่งข้อมูลไปแสดงผลยังหน้าจอ LCD ที่ติดไว้ที่หัวหลอดเอกซเรย์



ภาพที่ 3.1 Block diagram Device A

จากภาพ Block diagram Device A จะประกอบด้วย เซ็นเซอร์ที่ใช้สำหรับบอกทิศทางแนวการวางตัว (orientation), บอร์ด Arduino ที่ใช้คอมพิวเตอร์ในการ upload code และให้ power แก่บอร์ดเพื่อทำงาน, เซ็นเซอร์บลูทูธสำหรับสื่อสารระหว่าง Device A และ Device B และจอแสดงผลที่เป็น LCD ที่แสดงค่าพิกัดการหมุน (roll, pitch, yaw)

- การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์



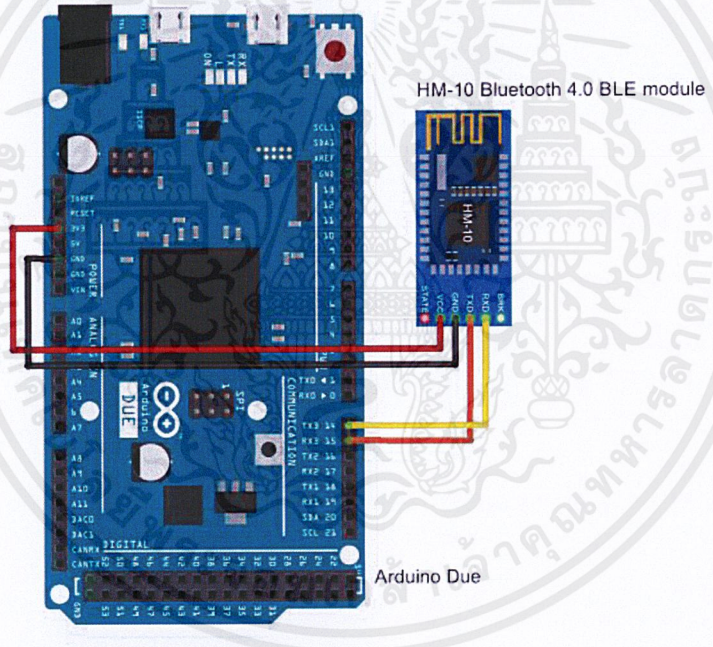
ภาพที่ 3.2 การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arduino Mega	BNO 055
5V	VCC
GND	GND
SDA	SDA
SCL	SCL

ตารางที่ 3.1 ขาที่ใช้เชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์

- การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์



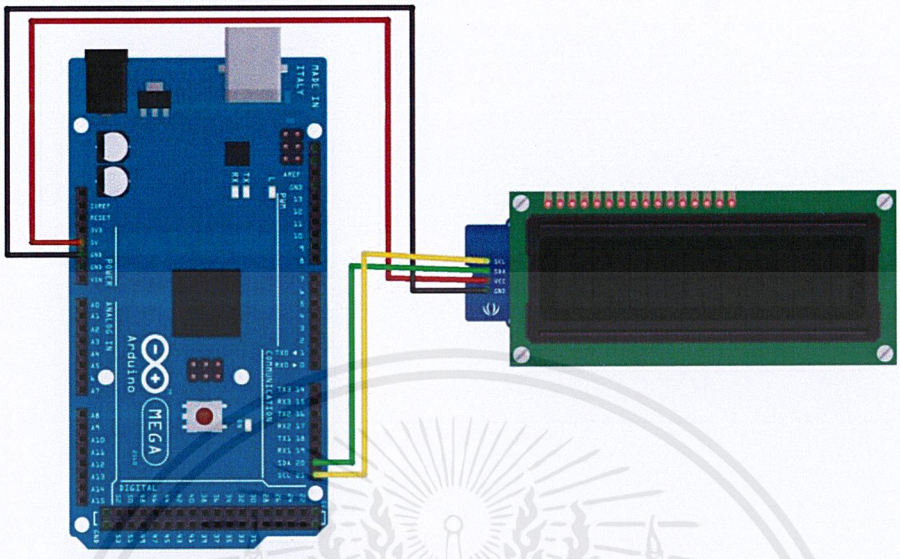
ภาพที่ 3.3 การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์

Arduino Mega	HM-10 BLE 4.0
5V	VCC
GND	GND
RX1	TXD
TX1	RXD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 ขาที่ใช้เชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์

- การเชื่อมจอแสดงผล LCD 2004 กับไมโครคอนโทรเลอร์

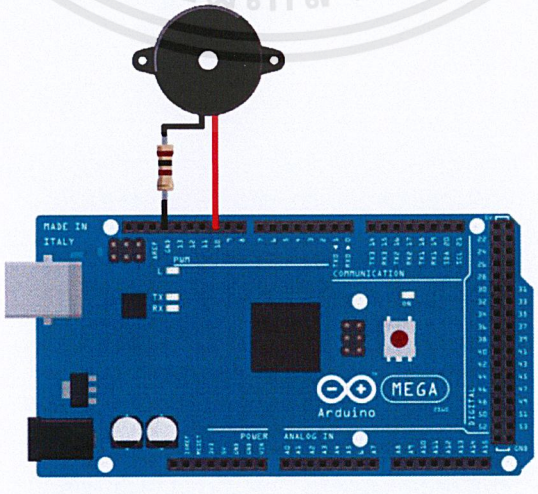


ภาพที่ 3.4 การเชื่อมจอแสดงผล LCD 2004 กับไมโครคอนโทรเลอร์

Arduino Mega	LCD 2004
5V	VCC
GND	GND
20	SDA
21	SCL

ตารางที่ 3.3 ขาที่ใช้เชื่อมจอแสดงผล LCD 2004 กับไมโครคอนโทรเลอร์

- การเชื่อม buzzer กับไมโครคอนโทรเลอร์



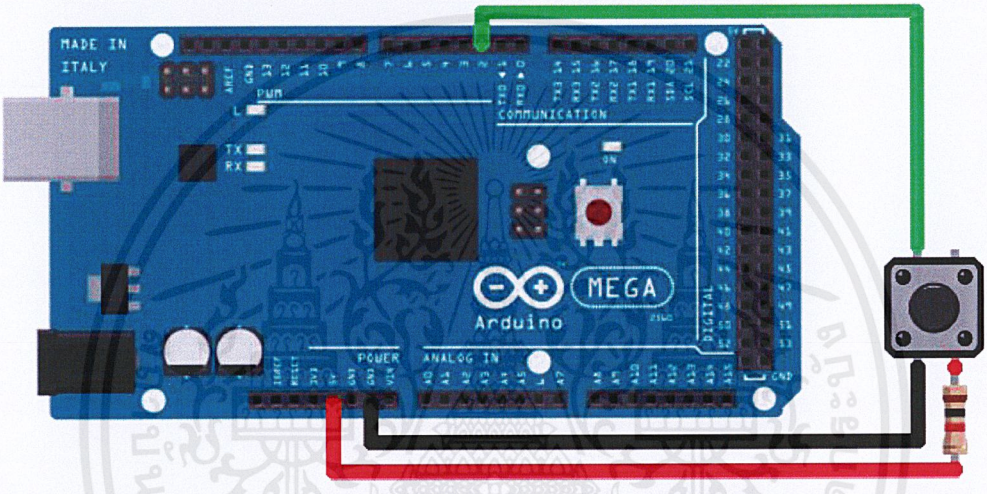
ภาพที่ 3.5 การเชื่อม buzzer กับไมโครคอนโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arduino Mega	Buzzer
2	VCC
GND	GND

ตารางที่ 3.4 ขาที่ใช้เชื่อม buzzer กับไมโครคอนโทรเลอร์

- การเชื่อม button กับไมโครคอนโทรเลอร์



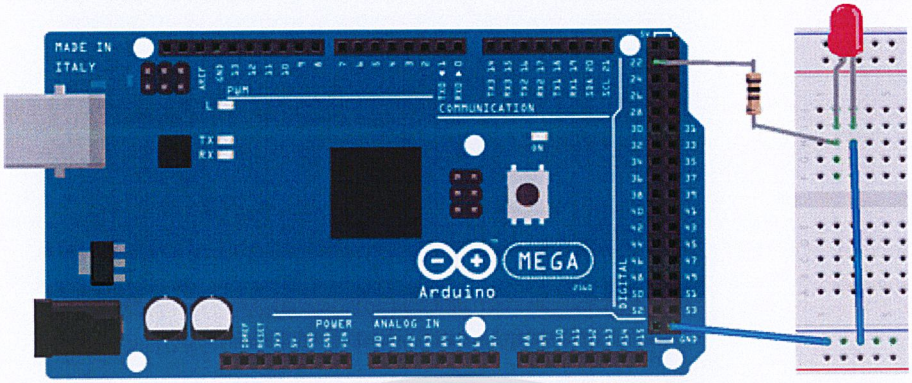
ภาพที่ 3.6 การเชื่อม button กับไมโครคอนโทรเลอร์

Arduino Mega	Switch
3	VCC
GND	GND

ตารางที่ 3.5 ขาที่ใช้เชื่อม switch กับไมโครคอนโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

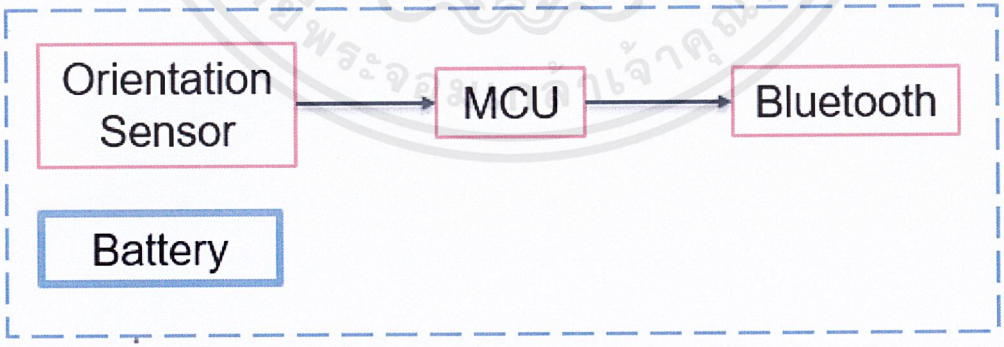
- การเชื่อม LED กับไมโครคอนโทรเลอร์



ภาพที่ 3.7 การเชื่อม LED กับไมโครคอนโทรเลอร์

Arduino Mega	Switch
4	VCC
GND	GND

ตารางที่ 3.6 ขาที่ใช้เชื่อม LED กับไมโครคอนโทรเลอร์

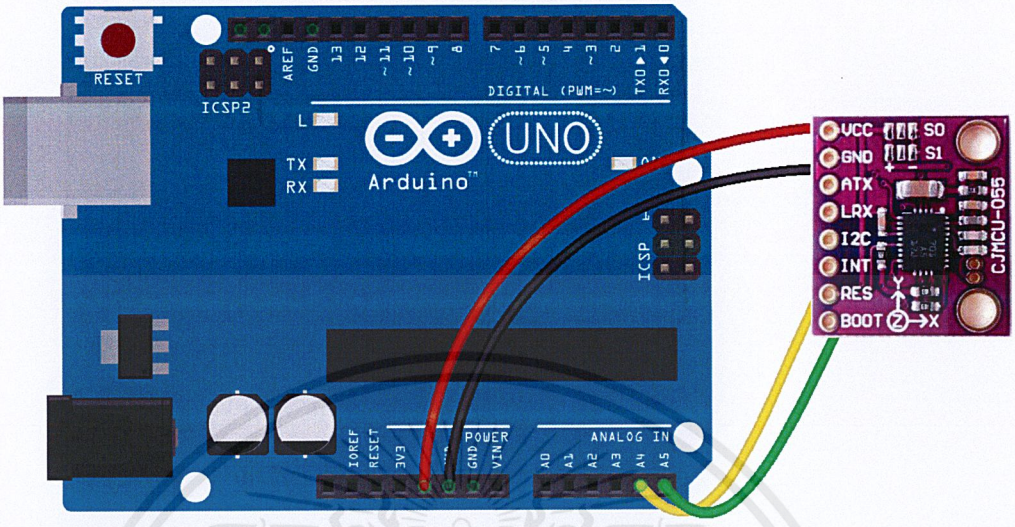


ภาพที่ 3.8 Block diagram Device B

จากภาพ Block diagram Device B จะประกอบด้วย เซ็นเซอร์ที่ใช้สำหรับบอกทิศทางแนวการวางตัว (orientation), บอร์ด Arduino ที่ใช้ power จากแบตเตอรี่ให้แก่บอร์ดเพื่อทำงาน, เซ็นเซอร์บลูทูธสำหรับสื่อสารระหว่าง Device A และ Device B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์



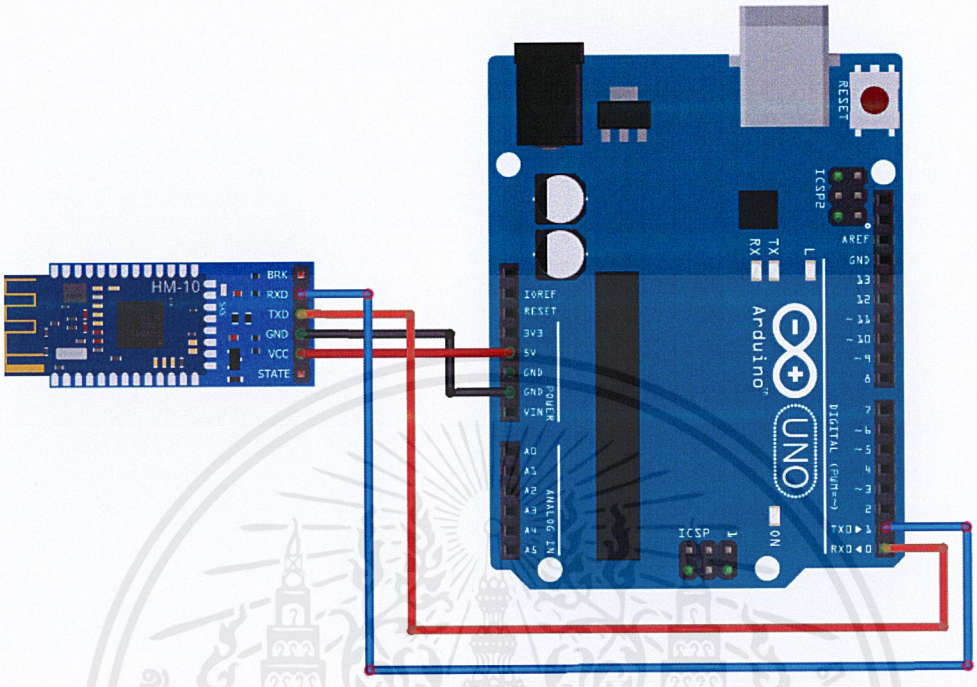
ภาพที่ 3.9 การเชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์

Arduino UNO R3	BNO 055
5V	VCC
GND	GND
A4	SDA
A5	SCL

ตารางที่ 3.7 ขาที่ใช้เชื่อมเซ็นเซอร์ BNO 055 กับไมโครคอนโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์



ภาพที่ 3.10 การเชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์

Arduino UNO R3	HM-10 BLE 4.0
5V	VCC
GND	GND
10	TXD
11	RXD

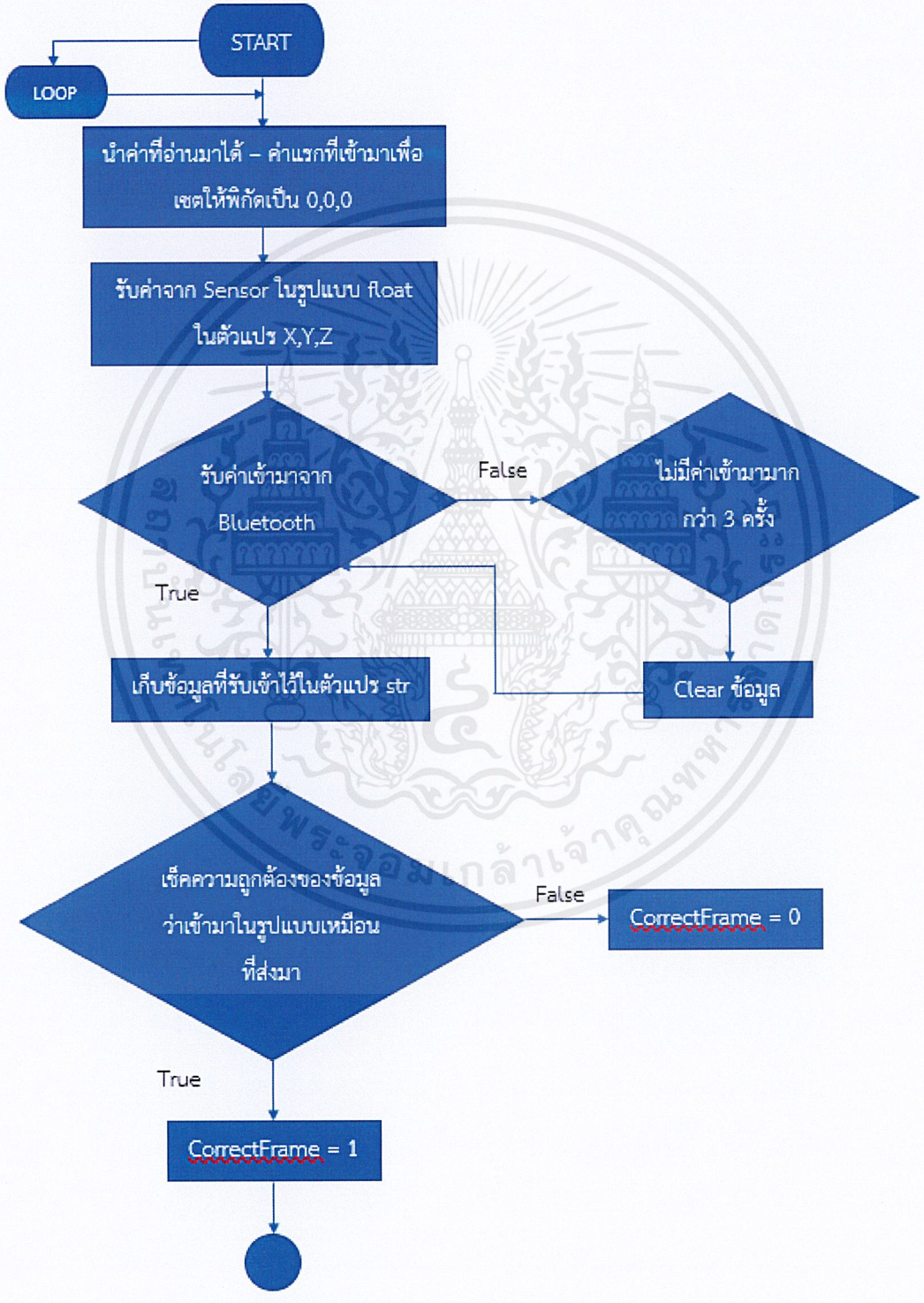
ตารางที่ 3.8 ขาที่ใช้เชื่อมเซ็นเซอร์บลูทูธ HM-10 BLE 4.0 กับไมโครคอนโทรเลอร์

3.3 การออกแบบโปรแกรม

ก่อนทำการเขียนโปรแกรมต้องทำการเขียนผังงานขั้นตอนการทำงานของโปรแกรมก่อน ซึ่งขั้นตอนแรกต้องทำการกำหนดค่าเริ่มต้นให้พอร์ตอินพุท เอาท์พุท กำหนดความเร็วในการสื่อสารข้อมูลผ่านระบบบัส UART ให้ทำงานที่ 9600 bps กำหนดเรียกใช้ไลบรารี่ Wire.h เพื่อใช้ในการสื่อสารข้อมูล

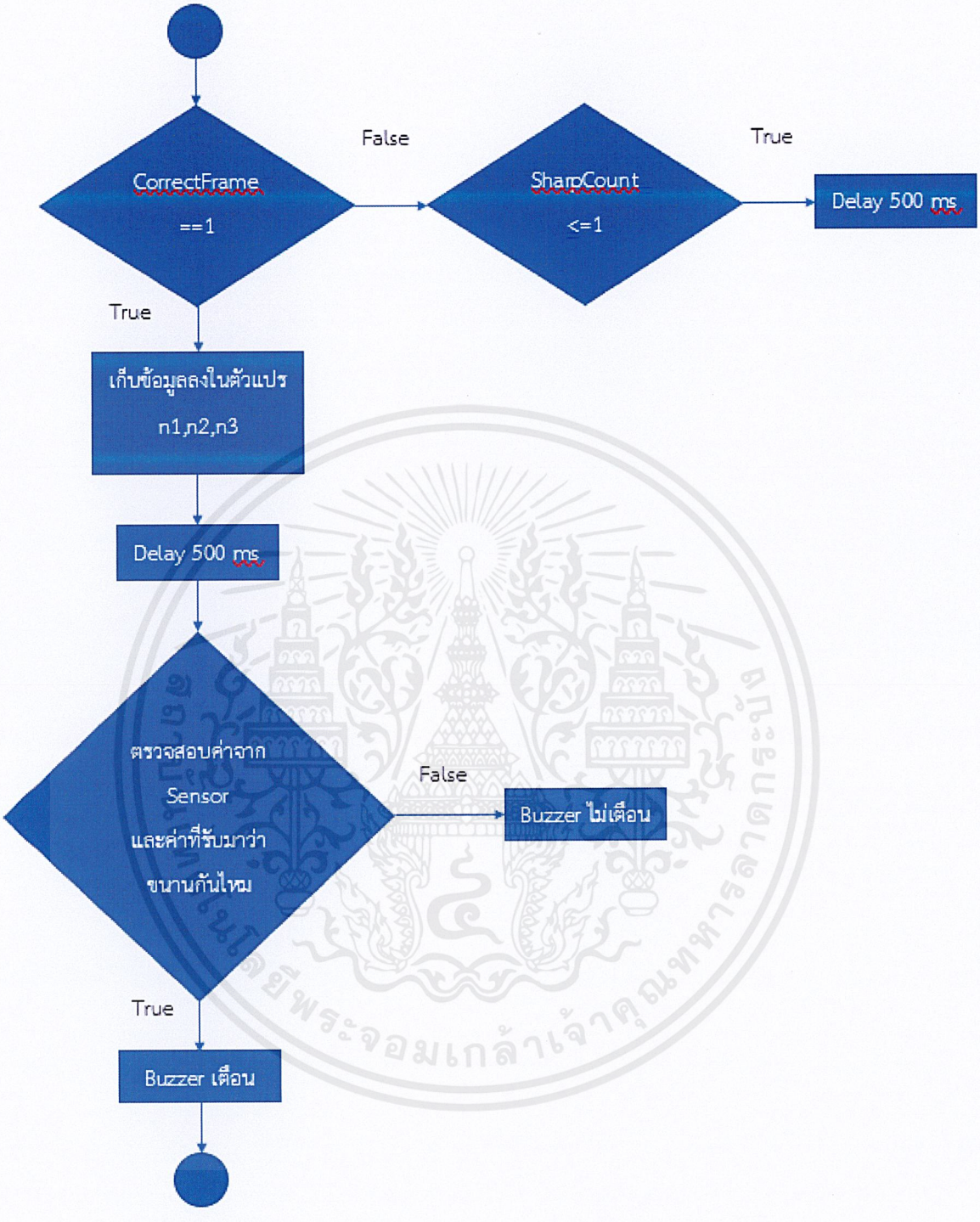
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านระบบบัส I2C และกำหนดเรียกใช้ไลบรารี่ DFRobot_BNO055.h เพื่อใช้ในการนำค่าข้อมูลของเซ็นเซอร์ BNO 055 มาใช้ ที่มีความละเอียดสูง ข้อมูลที่ได้มีค่าทศนิยม 2 ตำแหน่ง โดยนำข้อมูลที่ได้ของอุปกรณ์ B ทำการส่งค่าไปยัง อุปกรณ์ A เพื่อแสดงผลที่ หน้าจอที่ติดไว้ยังอุปกรณ์ A เพื่อสำหรับแสดงข้อมูล ของอุปกรณ์ A และ B แล้วนำมาเปรียบเทียบกัน



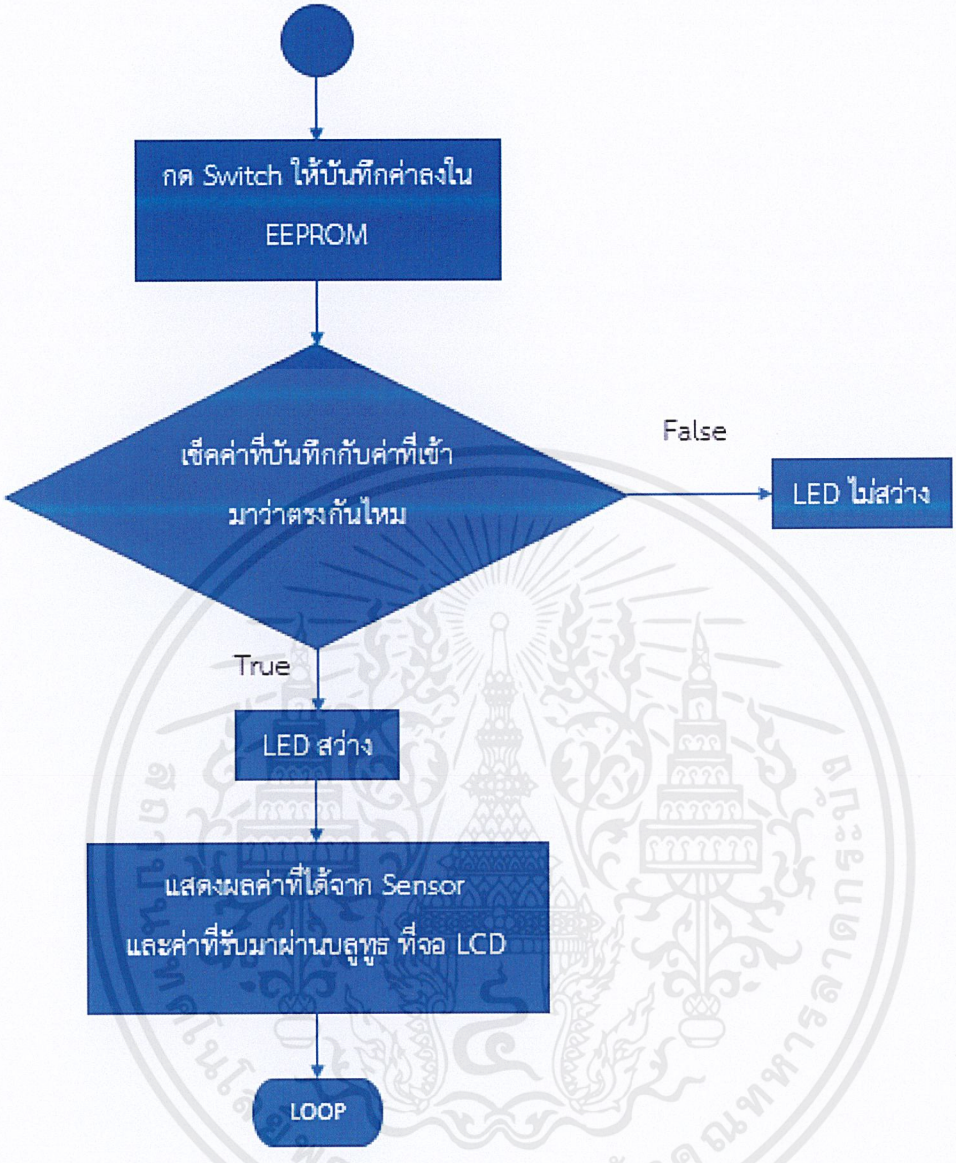
ภาพที่ 3.11 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



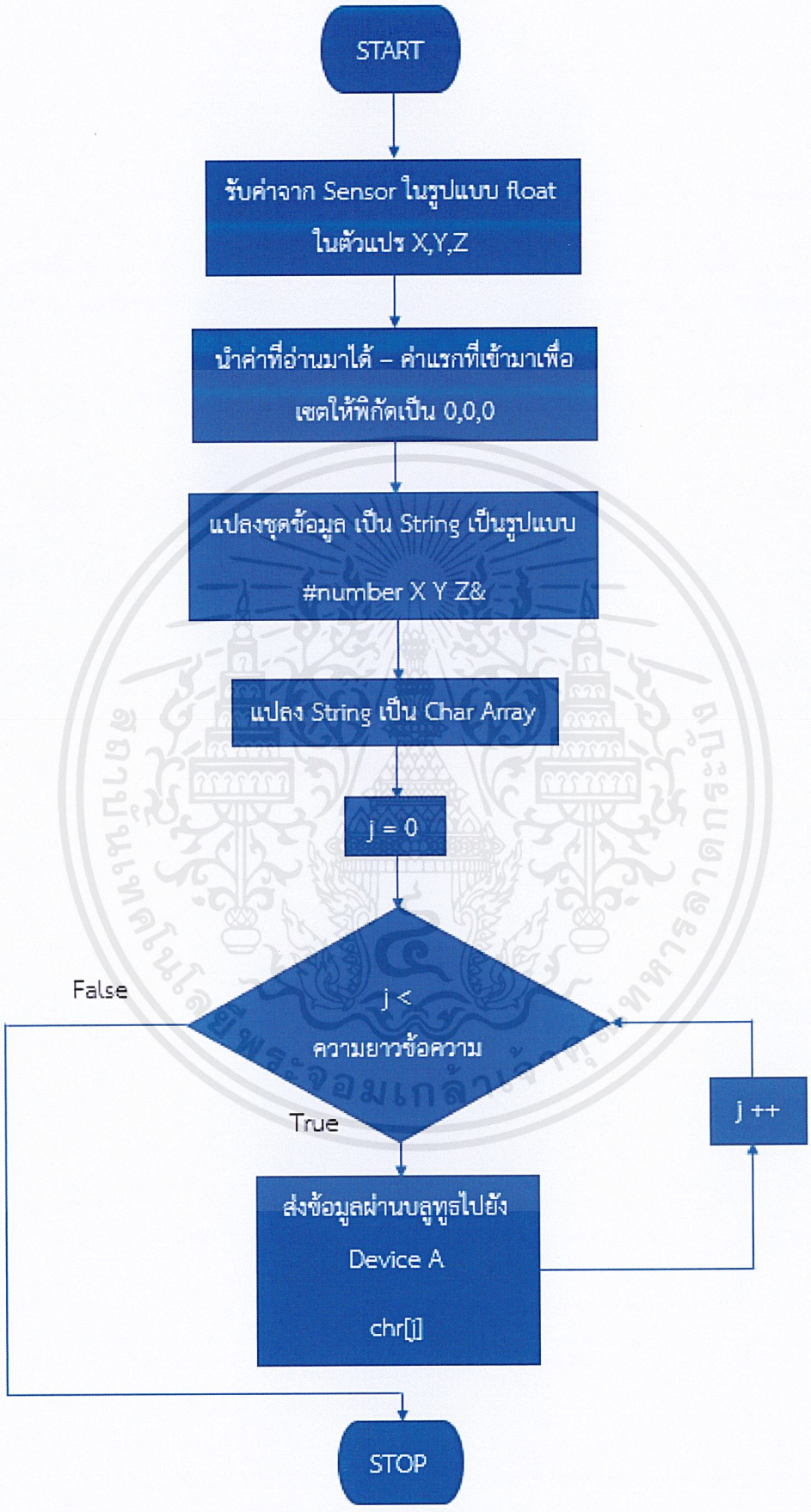
ภาพที่ 3.12 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.13 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device A (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

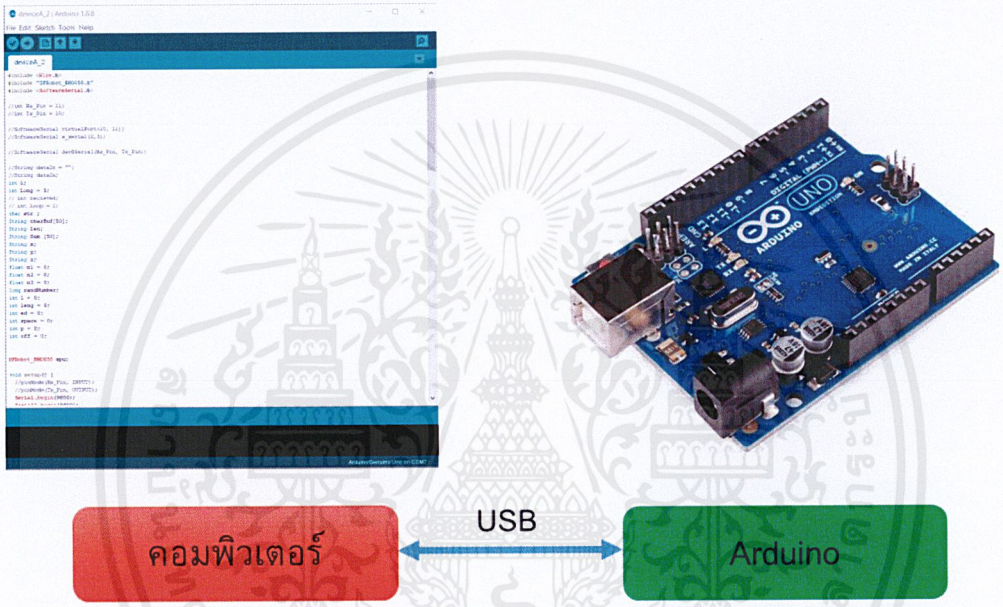


ภาพที่ 3.13 ผังงานขั้นตอนการทำงานของโปรแกรมของ Device B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

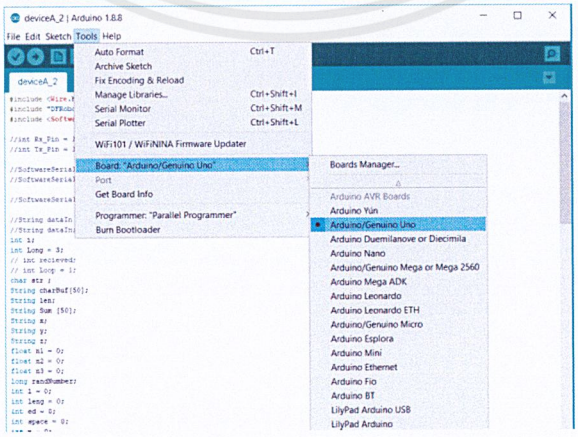
3.4 การเขียนโปรแกรม

การเขียนโปรแกรมในโครงงานนี้จะเขียนบน Arduino IDE ชื่อ Arduino 1.8.8 ซึ่งสามารถดาวน์โหลดได้ฟรีจากเว็บไซต์ของ Arduino ได้เลยที่ https://www.arduino.cc/en/Main/Software และเลือกดาวน์โหลดให้ตรงกับซอฟต์แวร์ระบบของคอมพิวเตอร์ที่เราใช้ ซึ่งสามารถใช้ได้ทั้ง Windows, Mac OS X, และ Linux จากนั้นก็สร้างโปรแกรมขึ้นมา โดยตั้งชื่อว่า Device A และ Device B สำหรับอุปกรณ์ทั้ง 2 ตัว ที่ติดอยู่กับหัวหลอดเอกซเรย์ และฉากรับรังสีของเครื่องเอกซเรย์ โดยสื่อสารกันระหว่างโปรแกรมที่เขียนกับบอร์ด Arduino โดยใช้ USB



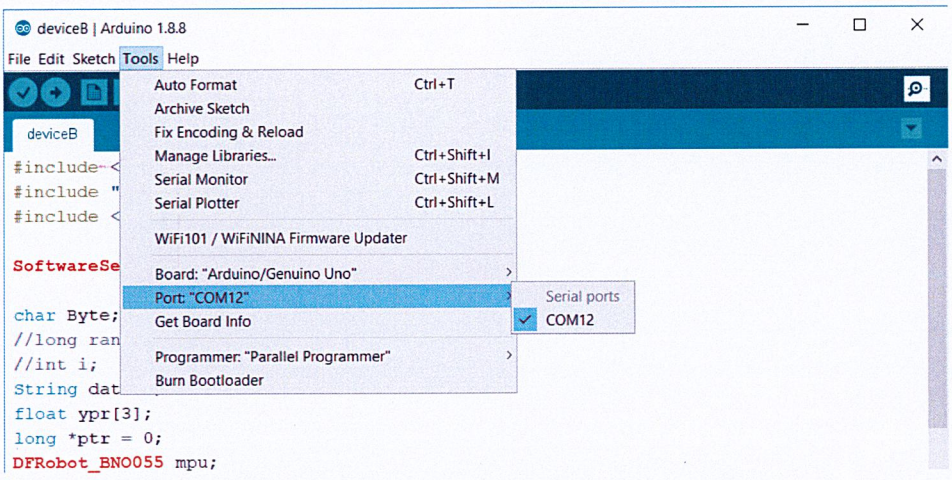
ภาพที่ 3.14 รูปแบบการเขียนโปรแกรมบน Arduino

หลังจากเขียนโปรแกรมสำหรับ Device A และ Device B เสร็จเรียบร้อยแล้ว ให้เลือกรุ่นบอร์ด Arduino ที่ใช้ และหมายเลข Com port



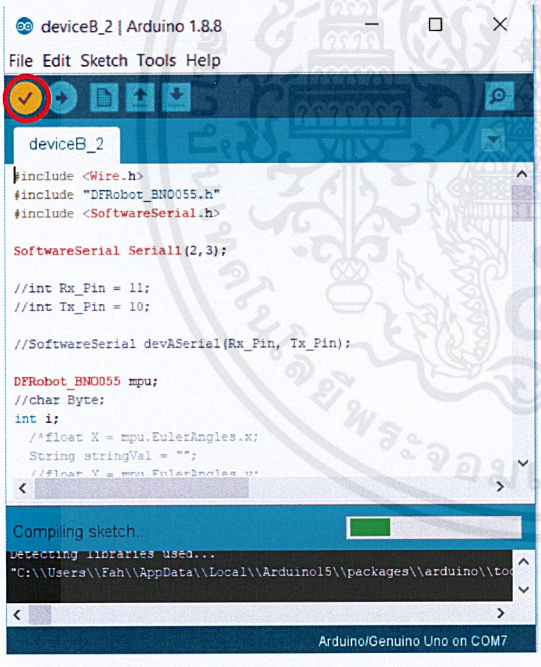
ภาพที่ 3.15 เลือกรุ่นบอร์ด Arduino ที่ต้องการ upload

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

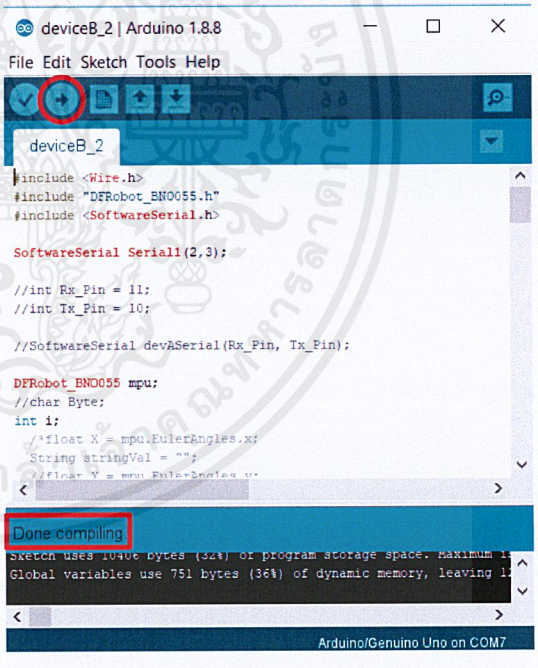


ภาพที่ 3.16 เลือกหมายเลข Com port ของบอร์ด

จากนั้นกดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง และ Compile โค้ดโปรแกรมจากนั้นกดปุ่ม Up- load โค้ด โปรแกรมไปยังบอร์ด Arduino ผ่านทางสาย USB เมื่ออับโหลดเรียบร้อยแล้ว จะแสดงข้อความแถบข้างล่าง “Done uploading” และบอร์ดจะเริ่มทำงานตามที่เขียนโปรแกรมไว้



ภาพที่ 3.17 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้องและ Compile โค้ดโปรแกรม



ภาพที่ 3.18 upload โค้ดโปรแกรมสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ฟังก์ชั่นเกี่ยวกับการสื่อสารผ่านพอร์ตอนุกรม

ใช้สำหรับสื่อสารข้อมูลระหว่างฮาร์ดแวร์ Arduino กับคอมพิวเตอร์ หรืออุปกรณ์อื่นๆ โดยจะแบ่ง พอร์ตสำหรับเชื่อมต่อออกเป็น 2 ส่วนคือ ส่วนแรกติดต่อพอร์ตอนุกรมเสมือน (Virtual Com Port) จากการทำงานของส่วนเชื่อมต่อพอร์ต USB ฟังก์ชั่นที่ใช้คือ Serial

อีกส่วนหนึ่งคือ ขาพอร์ตสื่อสารข้อมูลอนุกรมโดยใช้ขา 10 (RxD) และ 11 (TxD) และฟังก์ชั่นของ Arduino Mega 2560 ที่ใช้ คือ Serial1

3.5.1 Serial.begin (int datarate)

กำหนดค่า baund rate ของการรับส่งข้อมูลอนุกรมในหน่วยบิตต่อวินาที (bits per second : bps) ใช้ค่า ต่อไปนี้ 300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600 หรือ 115200

พารามิเตอร์

Int datarate ในหน่วยบิตต่อวินาที (baud หรือ bps)

3.5.2 Serial.available ( )

ใช้แจ้งว่าได้รับข้อมูลตัวอักษร (characters) แล้ว และพร้อมสำหรับการอ่านไปใช้งาน

ค่าที่ส่งกลับจากฟังก์ชั่น

จำนวนไบต์ที่พร้อมสำหรับการอ่านค่า โดยเก็บข้อมูลในบัฟเฟอร์ตัวรับ ถ้าไม่มีข้อมูลจะมีค่าเป็น 0 ถ้ามี ข้อมูลฟังก์ชั่นจะคืนค่าที่มากว่า 0 โดยบัฟเฟอร์สามารถเก็บข้อมูลได้สูงสุด 128 ไบต์

3.5.3 Serial.read ( )

ใช้อ่านค่าข้อมูลที่ได้รับจากพอร์ตอนุกรม

ค่าที่ส่งกลับจากฟังก์ชั่น

เป็นเลข int ที่เป็นไบต์แรกของข้อมูลที่ได้รับ (หรือเป็น -1 ถ้าไม่มีข้อมูล)

3.5.4 Serial.flush ( )

ใช้ล้างบัฟเฟอร์ตัวรับข้อมูลพอร์ตอนุกรมให้ว่าง

3.5.5 Serial.print (data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ส่งข้อมูลออกทางพอร์ตอนุกรม

พารามิเตอร์

Data – เป็นข้อมูลเลขจำนวนเต็มได้แก่ char, int หรือเลขทศนิยมที่ตัดเศษออกเป็นจำนวนเต็ม

รูปแบบฟังก์ชั่น

คำสั่งนี้สามารถเขียนได้หลายรูปแบบ Serial .print (b) เป็นการเขียนคำสั่งแบบไม่ได้ระบุรูปแบบ จะพิมพ์ค่าตัวแปร b เป็นเลขฐานสิบ โดยพิมพ์ ตัวอักษรรหัส ASCII การพิมพ์ข้อมูลจากฟังก์ชั่น Serial.print() และ Serial.write() ในรูปแบบต่างๆ แสดงผ่าน ทางหน้าต่าง Serial Monitor

3.5.6 Serial.println(data)

เป็นฟังก์ชั่นพิมพ์ (หรือส่ง) ข้อมูลออกทางพอร์ตอนุกรมตามด้วยรหัส carriage return (รหัส ASCII หมายเลข 13 หรือ \r) และ linefeed (รหัส ASCII หมายเลข 10 หรือ \n) เพื่อให้เกิดการเลื่อนบรรทัด และขึ้นบรรทัดใหม่ หลังจากพิมพ์ข้อความมีรูปแบบเหมือนคำสั่ง Serial.print()

รูปแบบฟังก์ชั่น

Serial.println(b)

3.5.7 Serial. write(data)

ใช้เขียนข้อมูลไบต์ไปยังพอร์ตอนุกรม

รูปแบบฟังก์ชั่น

Serial.write(val)

Serial.write(str)

Serial.write(buf, len)

พารามิเตอร์

val : ค่าที่ส่งไปในรูปแบบไบต์

str : สตริงที่ส่งเป็นชุดข้อมูลเป็นไบต์

buf : อาร์เรย์ที่ส่งเป็นชุดข้อมูลเป็นไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

len : จำนวนไบต์ที่จะส่งจากอาร์เรย์

3.5.8 delay(ms)

เป็นฟังก์ชั่นชะลอการทำงานหรือหน่วงเวลาของโปรแกรมตามเวลาที่กำหนดในหน่วยมิลลิวินาที

พารามิเตอร์

ms - ระยะเวลาที่ต้องการหน่วงเวลาหน่วยเป็นมิลลิวินาที (1000 ms เท่ากับ 1 วินาที)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

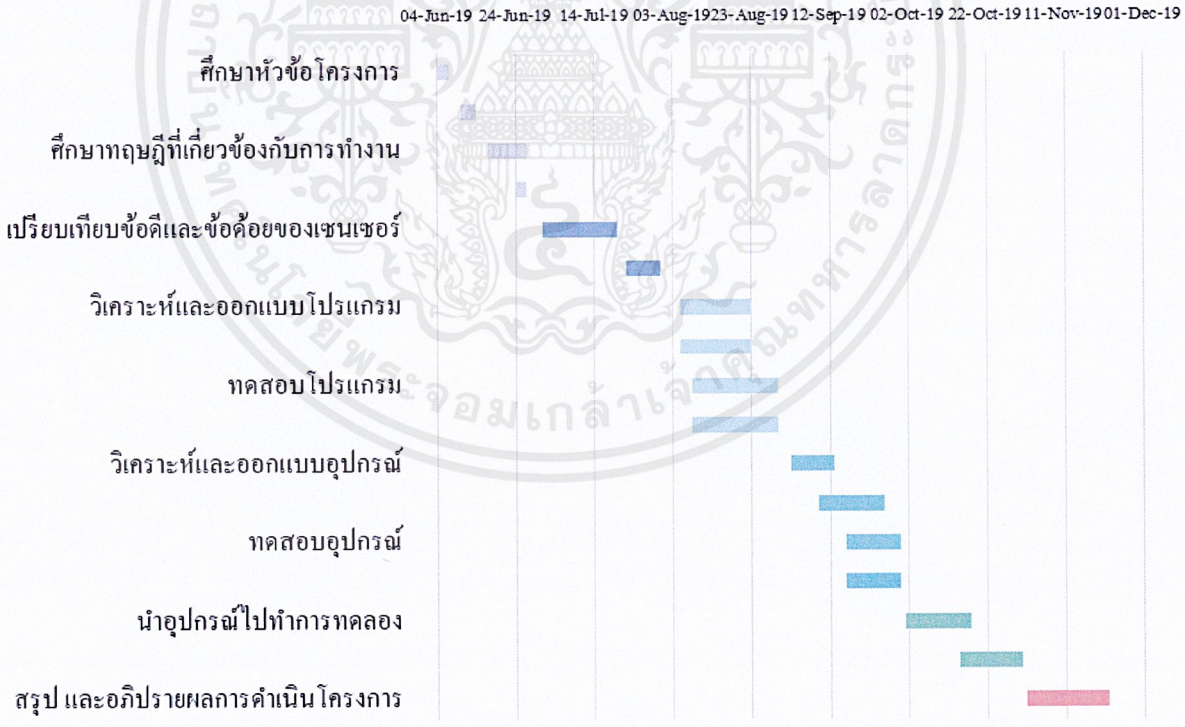
บทที่ 4

ผลการดำเนินงานโครงงาน

การดำเนินงานโครงงาน ในส่วนนี้จะแบ่งออกเป็นสามส่วนหลักๆส่วนแรกเป็นการเขียนโปรแกรมเพื่อนำมาใช้ในการทดสอบ ส่วนที่สองจะเป็นการสร้างชุดไมโครคอนโทรเลอร์เพื่อจะนำไปใช้ในการสร้างอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสีของเครื่องเอกซเรย์ดิจิทัลแบบเคลื่อนที่ ส่วนที่สามจะเป็นการทดสอบการทำงานโปรแกรมที่ได้เขียนขึ้นมาเพื่อหาข้อสรุปผลการทำงานของโปรแกรมเพื่อนำข้อมูลที่ได้มาแก้ไขปรับปรุงโปรแกรม และนำไปวิเคราะห์ข้อมูลและทำการสรุปผลต่อไป

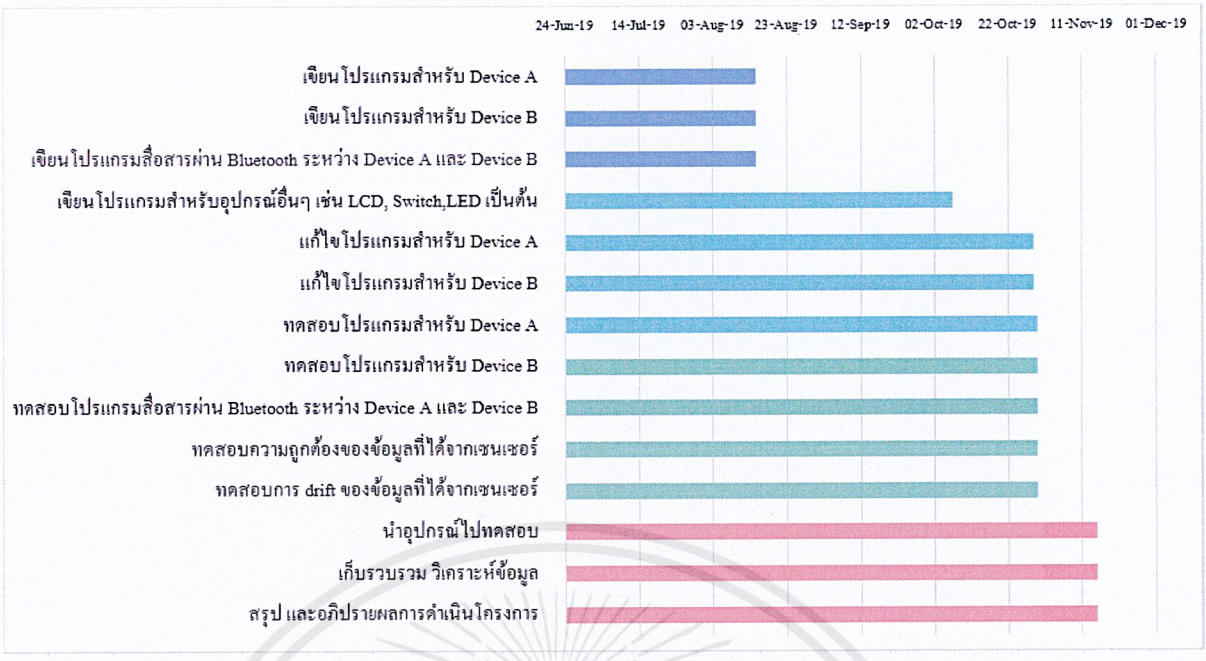
4.1 แผนการดำเนินงาน

แผนการดำเนินงานในช่วงแรกจะดำเนินงานตาม Gantt chart นี้ โดยเริ่มดำเนินจากวันที่ 4 มิถุนายน 2562 การดำเนินงานใน Gantt chart ที่ทำขึ้นในช่วงแรกเป็นการทำงานในเบื้องต้น ดังนั้นจึงได้ทำ Gantt chart ช่วงที่สองขึ้นเพื่อปรับให้ละเอียด และเหมาะสมกับการทำงาน



ภาพที่ 4.1 แผนการดำเนินงานช่วงแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.2 แผนการดำเนินงานช่วงที่สอง

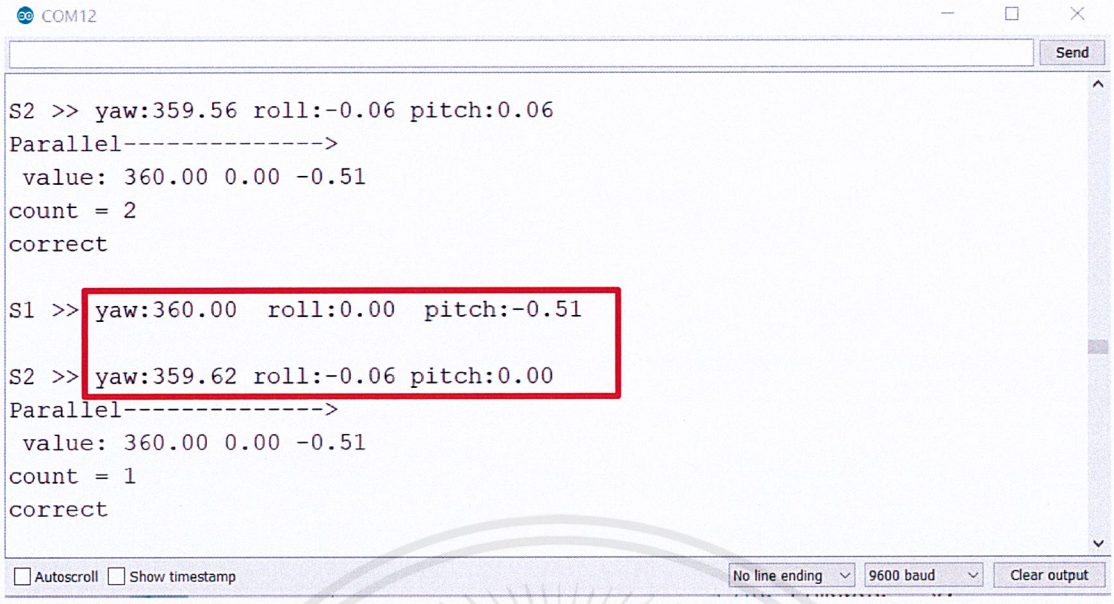
4.2 ทดสอบอุปกรณ์

4.2.1 Device A

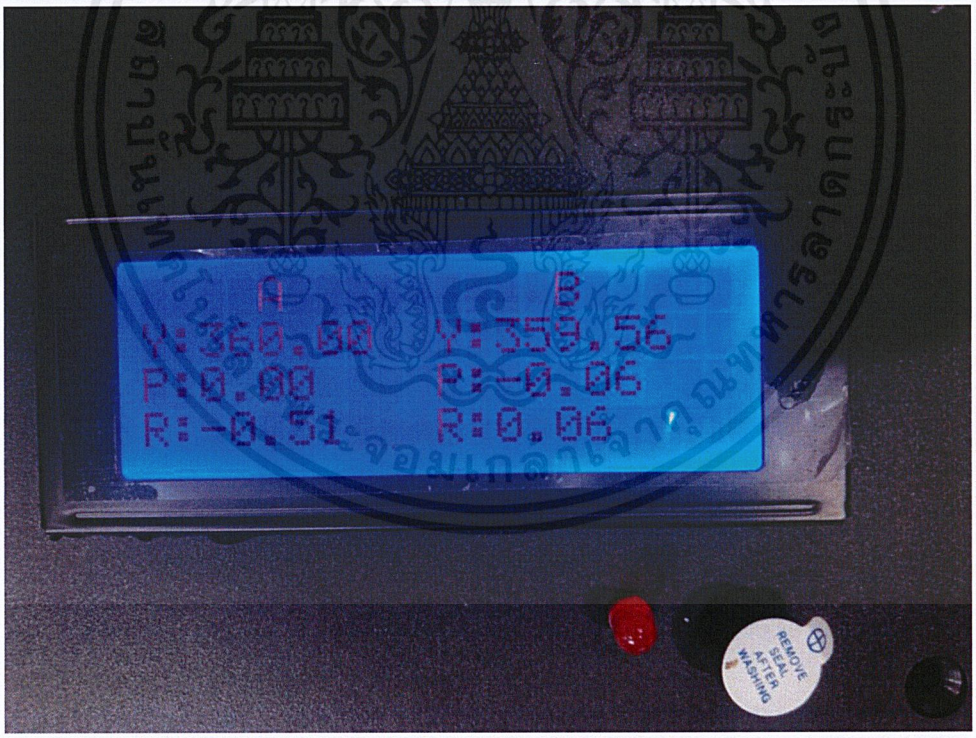
ในส่วนนี้จะทำการทดสอบการทำงานของ Device A โดยการทำงานของ Device A คือแสดงค่า roll, pitch, yaw ของเซ็นเซอร์ BNO 055 ที่ติดอยู่ที่ Device A และแสดงค่า roll, pitch, yaw ที่รับมาจากเซ็นเซอร์ที่ติดอยู่กับ Device B โดยผ่านทางการสื่อสารบลูทูธ

โดยโปรแกรมสำหรับ Device A เป็นโปรแกรมสำหรับแสดงค่าพิกัดที่อ่านมาได้จากเซ็นเซอร์ที่ติดอยู่กับอุปกรณ์ และที่ค่าพิกัดของเซ็นเซอร์อีกตัวที่รับมาจาก Device B โดยการสื่อสารผ่านบลูทูธที่ส่ง Array ของ char ของข้อมูลเซ็นเซอร์ โปรแกรมสำหรับ Device A จึงต้องทำหน้าที่แปลงกลับมาให้เป็น float แล้วจะแสดงผลออกมาที่หน้าจอคอมพิวเตอร์ โดยเปิด Serial Monitor ผ่านโปรแกรม Arduino และจะแสดงผลออกมาที่ LCD 2004 บนอุปกรณ์ ดังภาพ ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.3 Serial monitor การทำงานของ Device A ที่สามารถแสดงข้อมูลของเซ็นเซอร์ที่ติดไว้กับทั้ง Device A และแสดงค่าที่รับมาจากเซ็นเซอร์ ที่ Device B ส่งข้อมูลมา



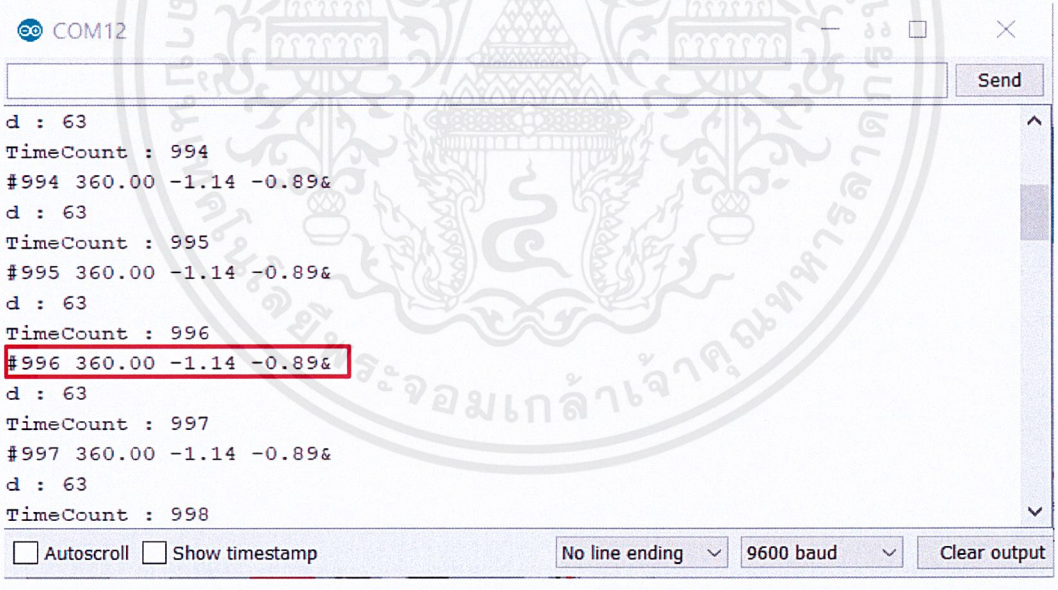
ภาพที่ 4.4 LCD 2004 การทำงานของ Device A ที่สามารถแสดงข้อมูลของเซ็นเซอร์ที่ติดไว้กับทั้ง Device A และแสดงค่าที่รับมาจากเซ็นเซอร์ ที่ Device B ส่งข้อมูลมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 Device B

จากการทดลองโปรแกรม ให้โปรแกรมสามารถใช้งานได้ โดยได้เขียนโปรแกรมสำหรับใช้กับ Device B ดังนี้ โดยจะเป็นโปรแกรมที่ส่งข้อมูลจากเซ็นเซอร์ BNO 055 โดยข้อมูลที่ได้จากเซ็นเซอร์จะอยู่ในรูปแบบของ float ดังนั้นโปรแกรมของ Device B จะทำการแปลงข้อมูลให้ส่งไปในรูปแบบ Array ของ char เพื่อที่จะสามารถส่งข้อมูลผ่านบลูทูธไปยัง Device A ได้ โดยมีโปรโตคอลในการส่งข้อมูลไปยัง Device A คือ “#996 360.00 -1.14 -0.89&” ซึ่งโปรโตคอลนี้ก็จะประกอบไปด้วย

- ‘#’ คือ Start bit ตัวแรกที่ส่งเข้ามาเพื่อบอกว่าหลังจากนี้จะเป็นข้อมูลของเซ็นเซอร์ BNO 055 เข้ามา
- ‘996’ คือ Number Frame จำนวนครั้งที่มีข้อมูลส่งออกไป
- ‘360.00 -1.14 -0.89’ คือ ค่า Yaw, Pitch, Roll ตามลำดับ ที่อ่านได้จากเซ็นเซอร์ BNO 055
- ‘&’ คือ Stop bit ตัวสุดท้ายที่ส่งเข้ามาเพื่อบอกว่าจบชุดข้อมูลของเซ็นเซอร์ BNO 055



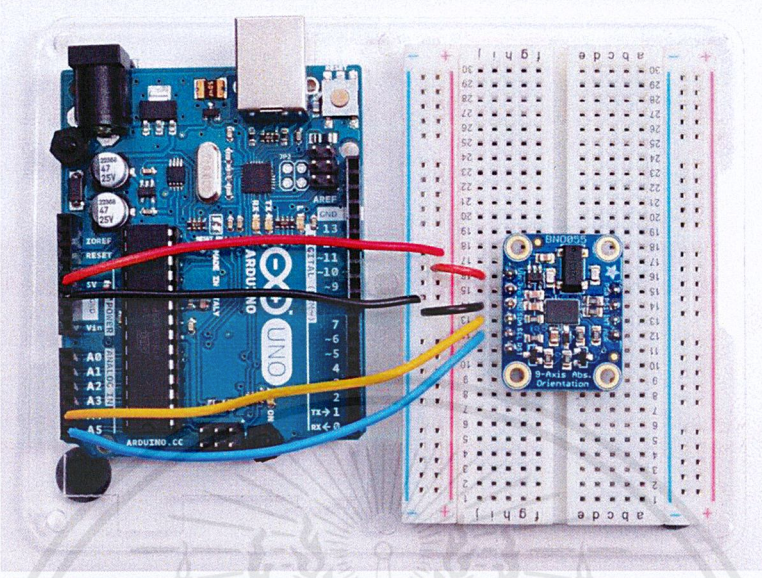
ภาพที่ 4.5 serial monitor ที่แสดงค่าส่งออกมาจากเซ็นเซอร์เป็น float หรือ ทศนิยม

4.3 ทดสอบการสื่อสาร

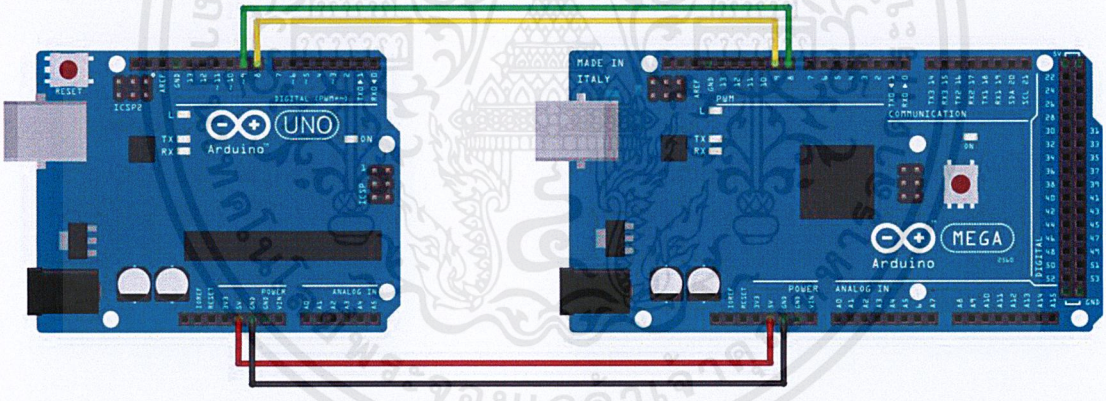
เมื่อนำทั้ง 2 โปรแกรมไป compile และ upload โปรแกรมลงบอร์ด Arduino 2 บอร์ด โดยการนำเอาบอร์ด Arduino Mega ที่ทำหน้าที่เป็น Device A เชื่อมต่อกับเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และนำเอาบอร์ด Arduino Uno R3 ที่ทำหน้าที่เป็น Device B เชื่อมต่อกับเซ็นเซอร์ แล้วเชื่อมบอร์ด 2 บอร์ดนี้เข้าด้วยกัน



ภาพที่ 4.6 การนำเซ็นเซอร์ BNO 055 เชื่อมกับบอร์ด Arduino



ภาพที่ 4.7 การเชื่อมต่อบอร์ด Arduino Mega และArduino Uno R3 เข้าด้วยกัน

จากการทดลองโปรแกรมสำหรับชุดอุปกรณ์ทั้ง Device A และ Device B เมื่อ compile และ upload โปรแกรมแล้ว ก็ทำการการเปิด serial monitor เพื่อตรวจสอบ พบว่าสามารถแสดงค่าที่อ่านมาจากเซ็นเซอร์ที่เชื่อมอยู่กับบอร์ด Arduino Mega ได้ และยังสามารถแสดงค่าที่ส่งมาจากอีกเซ็นเซอร์ที่เชื่อมบอร์ด Arduino Uno R3 ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดสอบการทำงานของชุดอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสี

1) ทดสอบการทำงานของชุดอุปกรณ์ และโปรแกรม โดยเปรียบเทียบ 2 กล่อง ณ องศาต่างๆ

โดยทำการตั้งค่ามุมองศาจริงโดยใช้ชิ้นงาน 3D Print ปรับองศาต่างๆ โดยเปรียบเทียบองศาจริงของมุม Yaw ที่ได้ติดตั้งไว้ ณ องศาต่างๆ คือ 0, 30, 60 และ 90 องศา แล้วเก็บค่าองศาที่ได้จากโปรแกรม จำนวน 30 ค่าเพื่อนำค่าที่ได้มาหาค่าเฉลี่ย

องศาจริง	ค่าเฉลี่ยองศา กล่อง A	ค่าเฉลี่ยองศา กล่อง B
0	0.17	0.32
30	30.75	30.39
60	64.83	61.34
90	88.15	93.37
RMSE	0.95469	0.66851

ตารางที่ 4.1 แสดงค่าเฉลี่ยของมุมที่สามารถวัดได้จากกล่อง A และ กล่อง B

ผลการทดลองถูกแสดงในตาราง จะเห็นว่าองศา yaw จากโปรแกรมที่เขียนไว้สามารถอ่านได้โดยเปรียบเทียบระหว่างสองกล่อง ที่ตั้งไว้องศาเดียวกันกับค่าองศาจริงจะเห็นได้ว่า ค่า error แต่ละกล่อง เมื่อเทียบกับองศาจริงจะมีค่าใกล้เคียงกับองศาจริง และมี RSME (Root Mean Square Error) ของกล่อง A อยู่ที่ 0.95469 และของกล่อง B อยู่ที่ 0.66851 นั่นหมายความว่า Error ของมุมที่วัดได้จะแปรผันกับการตั้งระนาบและองศาที่ระนาบมีการเปลี่ยนแปลง ทั้งนี้ค่า Error อาจเกิดจากการติดตั้งชุดการทดลองที่ไม่ได้มาตรฐานที่เพียงพอทำให้ค่าที่อ่านมาได้ไม่น่าเชื่อถือ

2) ทดสอบชุดอุปกรณ์ และโปรแกรม โดยเปรียบเทียบ Yaw, Pitch และ Roll ที่ 0, 30, 60 และ 90 องศา

โดยทำการตั้งค่ามุมองศาจริงโดยใช้ชิ้นงาน 3D Print ปรับองศา หลังจากนั้นทดลองโดยเปรียบเทียบองศาจริงของมุม Roll, Pitch และ Yaw ที่ได้ติดตั้งไว้ ณ องศาต่างๆ คือ 0, 30, 60 และ 90 องศาหลังจากนั้นเปิดโปรแกรม Arduino เก็บค่าองศาที่ได้จากโปรแกรม จำนวน 30 ค่าเพื่อนำค่าที่ได้มาหาค่าเฉลี่ย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

degree	ตารางแสดงค่าเฉลี่ยของข้อมูล											
	0			30			60			90		
	yaw	roll	pitch	yaw	roll	pitch	yaw	roll	pitch	yaw	roll	pitch
yaw	0.19	-0.19	0.00	-30.68	0.32	0.00	-59.38	0.39	0.00	-89.46	0.28	0.00
roll	360.00	-0.58	0.19	360.00	31.28	-167.29	355.70	60.89	-172.72	357.15	88.22	-173.92
pitch	360.00	0.06	0.19	354.30	-2.72	31.06	-184.81	-4.56	61.70	-183.80	-4.63	89.37

ตารางที่ 4.2 แสดงค่าเฉลี่ยของข้อมูล 30 ค่า

การทดลองที่ถูกแสดงในตาราง พบว่าในแนวแกน roll yaw pitch ในแต่ละองศาที่ทดสอบมีค่าใกล้เคียงกับองศาที่กำหนดไว้เพื่อทดสอบ โดยที่มุม 0 องศา พบว่ามีค่า error ของแกน yaw อยู่ที่ประมาณ 0.19 องศา ค่า error ของแกน roll อยู่ที่ประมาณ - 0.58 องศา และค่า error ของแกน pitch อยู่ที่ประมาณ 0.19 องศา ที่มุม 30 องศา พบว่ามีค่า error ของแกน yaw อยู่ที่ประมาณ - 0.68 องศา ค่า error ของแกน roll อยู่ที่ประมาณ 1.28 องศา และค่า error ของแกน pitch อยู่ที่ประมาณ 1.06 องศา ที่มุม 60 องศา พบว่ามีค่า error ของแกน yaw อยู่ที่ประมาณ - 0.62 องศา ค่า error ของแกน roll อยู่ที่ประมาณ 0.89 องศา และค่า error ของแกน pitch อยู่ที่ประมาณ 1.70 องศา และที่มุม 90 องศา พบว่ามีค่า error ของแกน yaw อยู่ที่ประมาณ - 0.54 องศา ค่า error ของแกน roll อยู่ที่ประมาณ 1.78 องศา และค่า error ของแกน pitch อยู่ที่ประมาณ 0.63 องศา

3) ทดสอบการเกิดการดริฟท์โดยเปรียบเทียบ Yaw, Pitch และ Roll ที่ 0, 30, 60 และ 90 องศา

โดยทำการตั้งค่ามุมองศาจริงโดยใช้ชิ้นงาน 3D Print ปรับองศา หลังจากนั้นทดลองโดยเปรียบเทียบองศาจริงของมุม Roll, Pitch และ Yaw ที่ได้ติดตั้งไว้ ณ องศาต่าง ๆคือ 0, 30, 60 และ 90 องศาหลังจากนั้นเปิดโปรแกรม Arduino เก็บค่าองศาที่ได้จากโปรแกรม จำนวน 30 ค่า และวางทิ้งไว้ที่องศาเดิมเป็นเวลา 60 วินาที เพื่อนำค่าที่ได้มาหาค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน

degree	ตารางแสดงค่าเบี่ยงเบนมาตรฐานของข้อมูล (S.D.)											
	0			30			60			90		
	yaw	roll	pitch	yaw	roll	pitch	yaw	roll	pitch	yaw	roll	pitch
yaw	0.02	0.01	0.03	0.14	0.01	0.00	0.09	0.07	0.00	0.16	0.10	0.00
roll	0.00	0.03	0.00	0.00	0.04	0.03	0.00	0.01	0.00	0.00	0.03	0.02
pitch	0.00	0.00	0.01	0.03	0.01	0.10	0.01	0.02	0.07	0.01	0.03	0.00

ตารางที่ 4.3 แสดงค่าเบี่งเบนมาตรฐานของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ถูกแสดงในตาราง พบว่าในแนวแกน roll yaw pitch ในแต่ละองศาที่ทดสอบและวางทิ้งไว้ เกิดการเปลี่ยนแปลงของค่าองศาที่อ่านได้น้อยมาก โดยสังเกตได้จากค่า S.D. (Standard Deviation) ที่อ่านได้ที่มุม 0 องศา โดยค่า S.D. ของแกน yaw มีค่าประมาณ 0.02 องศา ค่า S.D. ของแกน roll อยู่ที่ประมาณ 0.03 องศา และค่า S.D. ของแกน pitch อยู่ที่ประมาณ 0.01 องศา ที่มุม 30 องศา โดยค่า S.D. ของแกน yaw มีค่าประมาณ 0.14 องศา ค่า S.D. ของแกน roll อยู่ที่ประมาณ 0.04 องศา และค่า S.D. ของแกน pitch อยู่ที่ประมาณ 0.01 องศา ที่มุม 60 องศา โดยค่า S.D. ของแกน yaw มีค่าประมาณ 0.09 องศา ค่า S.D. ของแกน roll อยู่ที่ประมาณ 0.01 องศา และค่า S.D. ของแกน pitch อยู่ที่ประมาณ 0.07 องศา ที่มุม 90 องศา โดยค่า S.D. ของแกน yaw มีค่าประมาณ 0.16 องศา ค่า S.D. ของแกน roll อยู่ที่ประมาณ 0.03 องศา และค่า S.D. ของแกน pitch อยู่ที่ประมาณ 0.00 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงาน

จากการวิเคราะห์ปัญหา ศึกษางานวิจัย ระบบที่เกี่ยวข้อง ออกแบบโปรแกรมและทดสอบ ทำให้ได้ผลทดลองออกมา เมื่อวิเคราะห์ประเมินผลแล้ว จึงสามารถบ่งบอกถึงประสิทธิภาพการทำงานของระบบได้เป็นอย่างดี ในบทนี้จะกล่าวถึงการทดลอง ปัญหา ลา ข้อเสนอแนะในการวิจัย

5.1 สรุปผล

โครงงานนี้ทำเพื่อแก้ไขปัญหาการเกิด distortion ของภาพเอกซเรย์ที่เกิดจากการวางแนวรังสีให้ตรงกับฉากรับรังสี ที่มีความไม่แม่นยำ ซึ่งจากการทดลองใช้งานโปรแกรมและอุปกรณ์ที่ออกแบบ พบว่าการทดสอบอุปกรณ์ทั้ง 2 กล่อง ทั้งกล่อง A และกล่อง B สามารถแสดงค่าเซ็นเซอร์ที่ติดไว้ทั้ง 2 อันได้ จากนั้นเมื่อทดสอบการสื่อสาร พบว่าอุปกรณ์ทั้ง 2 กล่องสามารถสื่อสารกันได้โดยทั้งวิธีใช้สายไฟสื่อสาร และเซ็นเซอร์บลูทูธ โดยสังเกตได้จากการที่ข้อมูลเซ็นเซอร์ที่ติดตั้งไว้ใน กล่อง B ไปแสดงยังจอ LCD และ Serial monitor ของโปรแกรม Arduino ที่ติดไว้กับ กล่อง A ได้ และจากการทดสอบการทำงานของชุดอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสี

โดยเมื่อทดสอบการทำงานของชุดอุปกรณ์ และโปรแกรม พบว่ามีค่า RSME ของกล่อง A อยู่ที่ 0.95469 และของกล่อง B อยู่ที่ 0.66851 ซึ่งค่า RSME เมื่อมีค่าใกล้ 0 จะยิ่งมีความแม่นยำสูง ดังนั้น การทำงานของชุดอุปกรณ์มีความแม่นยำ แต่ควรจะให้ค่า RSME มีค่าน้อยกว่านี้

โดยเมื่อทดสอบชุดอุปกรณ์ ในแนว 3 แกน Yaw, Pitch และ Roll ที่มุม 0, 30, 60, และ 90 องศา พบว่าค่าเฉลี่ยของ 3 แนวแกน ในแต่ละมุม มีค่าใกล้เคียงกับค่ามุมจริงที่ได้ปรับองศาของชิ้นงาน 3D print ไว้ โดยมีค่า error ประมาณ 1%

โดยเมื่อทดสอบการเกิดการดริฟท์ของเซ็นเซอร์ พบว่า มีค่าเบี่ยงเบนมาตรฐานประมาณ 0.058333 ซึ่งมีค่าใกล้เคียง 0 ดังนั้นข้อมูลจากเซ็นเซอร์ ที่สามารถอ่านออกมาได้ มีความแกว่งของข้อมูลน้อยมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นชุดอุปกรณ์เปรียบเทียบพิกัดของหัวหลอดเอกซเรย์ กับฉากรับรังสี สามารถใช้งานในการหาตำแหน่งได้ในเบื้องต้น โดยถ้าอยากให้ประสิทธิภาพของภาพถ่ายสูงขึ้น ควรใช้แสงในการช่วยวางตำแหน่งควบคู่กันไปด้วย

5.2 ปัญหาและอุปสรรค

1) ขาดองค์ความรู้การเขียนโปรแกรมที่ซับซ้อน จึงทำให้เกิดความล่าช้าในการเขียนโปรแกรม

2) การสื่อสารกับผู้ให้ความรู้มีความสับสน จึงทำให้การเขียนโปรแกรม เกิดความล่าช้า

3) อุปกรณ์ที่ใช้ในการทดสอบ อาจมีขนาดใหญ่เกินไป อาจทำให้มีผลต่อการเก็บผลการทดลอง

5.4 แนวทางการแก้ไข

1) ศึกษาองค์ความรู้ด้านการเขียนโปรแกรมที่ซับซ้อนมากยิ่งขึ้น

2) ควรศึกษาการทำงาน และการใช้งานอุปกรณ์ต่างๆที่ใช้ในการทำชุดอุปกรณ์ให้เข้าใจมากยิ่งขึ้น

3) ปรับขนาดชุดอุปกรณ์การทดลอง และตำแหน่งการวางเซ็นเซอร์ภายในกล่องเอนกประสงค์ ให้สอดคล้องกันทั้ง 2 กล่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. สมหมาย กันทะเมืองลี้. (2560). **การวิเคราะห์ภาพดิจิตอลทางรังสีที่ไม่สามารถนำไปวินิจฉัยโรคได้ (reject image)**. วารสารสุขภาพภาคประชาชน, 12(4), 28-35.
2. **ความหมายของ IMU (Inertial Measurement Unit)** [ออนไลน์]. 2561 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก http://djigogadgets.com/index.php?route=journal2/blog/post&journal_blog_post_id=34
3. **Accelerometer คืออะไร** [ออนไลน์]. 2554 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.spicydog.org/blog/accelerometer-gyroscope/
4. **Gyroscope คืออะไร** [ออนไลน์]. 2554 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.spicydog.org/blog/accelerometer-gyroscope/
5. **Magnetometer คืออะไร** [ออนไลน์]. 2560 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก http://medium.com/@thanahongsuwan/micro-bit-%E0%B8%81%E0%B8%9A-arduino-ide-5-magnetometer-985b0ba75302
6. **MEMS Magnetometer** [ออนไลน์]. 2559 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://howtomechatronics.com/how-it-works/electrical-engineering/mems-accelerometer-gyroscope-magnetometer-arduino/
7. **GY-521 MPU6050** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.hotmcu.com/gy521-mpu6050-3axis-acceleration-gyroscope-6dof-module-p-83.html
8. **GY-89** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.hotmcu.com/imu-10dof-lsm303dl3gd20-bmp180-p-189.html?cPath=8
9. **BNO055** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://thinkrobotics.in/products/9-dof-absolute-orientation-bno055-sensor
10. **เกี่ยวกับเซ็นเซอร์ BNO055** [ออนไลน์]. 2562 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://th.electronics-council.com/capturing-imu-data-with-bno055-absolute-orientation-sensor-14555
11. **BNO055 IMU Sensor** [ออนไลน์]. 2562 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://it.mathworks.com/help/supportpkg/arduino/ref/bno055imusensor.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. **Arduino คืออะไร** [ออนไลน์]. 2562 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.thaieasyelec.com/article-wiki/latest-blogs/what-is-arduino-ch1.html
13. **Arduino Mega 2560** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 10 พฤศจิกายน 2562]. จาก https:// www.thaieasyelec.com/arduino-mega-2560.html
14. **การใช้งานจอ Character LCD กับ Arduino** [ออนไลน์]. 2562 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.ioxhop.com/article/30/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99%E0%B8%88%E0%B8%AD-character-lcd-%E0%B8%81%E0%B8%B1%E0%B8%9A-arduino-%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%A5%E0%B8%B0%E0%B9%80%E0%B8%AD%E0%B8%B5%E0%B8%A2%E0%B8%94
15. **บลูทูธ HM-10 BLE 4.0** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 10 พฤศจิกายน 2562]. จาก https://www.Myarduino.net/product/773/at-09-bluetooth-ble-module-v4-0-cc2541-hm-10
16. **Bluetooth Low Energy** [ออนไลน์]. 2562 [ค้นเมื่อ 10 พฤศจิกายน 2562]. จาก https://raspberrypi-thailand.blogspot.com/2018/01/bluetooth-low-energy-1.html
17. **การใช้งานพอร์ตสื่อสาร UART** [ออนไลน์]. 2557 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก http://aimagin.com/blog/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99%E0%B8%9E%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%95%E0%B8%AA%E0%B8%B7%E0%B9%88%E0%B8%AD%E0%B8%AA%E0%B8%B2%E0%B8%A3-uart/?lang=th
18. **การใช้งานพอร์ตสื่อสาร I2C** [ออนไลน์]. 2557 [ค้นเมื่อ 8 ตุลาคม 2562]. จาก http://aimagin.com/blog/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99%E0%B8%9E%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%95%E0%B8%AA%E0%B8%B7%E0%B9%88%E0%B8%AD%E0%B8%AA%E0%B8%B2%E0%B8%A3-i2c/?lang=th
19. **LED** [ออนไลน์]. 2562 [ค้นเมื่อ 10 พฤศจิกายน 2562]. จาก https://www.ju-led.com/16574233/led-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84% E0% B8%A3
20. **ฟังก์ชั่นพื้นฐานของ Arduino** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 8 ตุลาคม 2562]. จาก http://www.sbt.ac.th/new/sites/default/files/TNP_Unit_4.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

21. **Arduino Language** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 8 ตุลาคม 2562]. จาก https://www.arduino.cc/reference/en/language/functions/communication/serial/write/

22. **Communicate with Arduino** [ออนไลน์]. 2561 [ค้นเมื่อ 10 ตุลาคม 2562]. จาก https://www.aranacorp.com/en/communicate-with-arduino/

23. **Communication Between Two Arduino** [ออนไลน์]. ม.ป.ป. [ค้นเมื่อ 10 ตุลาคม 2562]. จาก https://www.ahirlabs.com/2018/04/29/communication-between-two-arduino/

24. **Metrics พื้นฐานสำหรับวัดผล RSME** [ออนไลน์]. 2562 [ค้นเมื่อ 10 ตุลาคม 2562]. จาก https://datarockie.com/2019/03/30/top-ten-machine-learning-metrics/

25. **Datasheet BNO 055** [ออนไลน์]. 2557 [ค้นเมื่อ 1 ตุลาคม 2562]. จาก https://cdn-shop.adafruit.com/datasheets/BST_BNO055_DS000_12.pdf

26. **การวัดหาพิกัดแนวทิศทางการวางตัว (orientation)** [ออนไลน์]. 2562 [ค้นเมื่อ 10 ตุลาคม 2562]. จาก https://www.arduitronics.com/article/68/arduino-uno-wifi-rev2-%E0%B8%84%E0%B8%B3%E0%B8%99%E0%B8%A7%E0%B8%99%E0%B8%A1%E0%B8%B8%E0%B8%A1-roll-%E0%B9%81%E0%B8%A5%E0%B8%B0-pitch-%E0%B8%88%E0%B8%B2%E0%B8%81%E0%B8%84%E0%B9%88%E0%B8%B2%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B9%80%E0%B8%A3%E0%B9%88%E0%B8%87%E0%B8%97%E0%B8%B5%E0%B9%88%E0%B8%A7%E0%B8%B1%E0%B8%94%E0%B9%84%E0%B8%94%E0%B9%89%E0%B8%88%E0%B8%B2%E0%B8%81-accelerometer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โค้ดโปรแกรมของ Device A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);

#include <EEPROM.h>

//SoftwareSerial virtualPort(10, 11);

//SoftwareSerial ArduinoMaster(2, 3);

const int LOOP_DELAY_ms = 500;

char str[100];

char f1[7];

char f2[7];

char f3[7];

float n1 = 0;

float n2 = 0;

float n3 = 0;

int fi; //count ระหว่าง spaceLoc[0]-[1]

int sharpCount = 0;

int andCount = 0;

int spaceCount = 0;

int spaceLoc[3];

bool correctFrame = 0;

int TimeSet = 0;

float init_yaw;

float init_pitch;

float init_roll;

float parallelY;

float parallelP;

float parallelR;

int y1Address = 0; // Address Y1

int y2Address = 1; // Address Y2

int p1Address = 2; // Address P1

int p2Address = 3; // Address P2

int r1Address = 4; // Address R1

int r2Address = 5; // Address R2

float Y1Set = 0;

float Y2Set = 0;

float P1Set = 0;

float P2Set = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float R1Set = 0;
float R2Set = 0;
unsigned long count = 0;
//int numberframe = 0;
//String numberframeStr;

//String str;
//char a[50];
//char *stra = &a[0];

#define buzzerPin 2
#define swPin 3
#define LedPin 4
bool swSet = false;
//bool numberframeStatus = false;

DFRobot_BNO055 mpu;

void setup() {
  Serial.begin(9600);
  Serial1.begin(9600);
  lcd.begin();
  //ArduinoMaster.begin(9600);

  pinMode(buzzerPin, OUTPUT);
  digitalWrite(buzzerPin, LOW);
  pinMode(swPin, INPUT_PULLUP);
  pinMode(LedPin, OUTPUT);// initialize
digital pin Ledpin as an output.
  digitalWrite(LedPin, LOW);

  while ( !Serial );

  while (!mpu.init())
  {
    Serial.println("ERROR! Unable to
initialize the chip!0");
    delay(50);
  }
  //
mpu.setMode(mpu.eNORMAL_POWER_
MODE, mpu.eFASTEST_MODE);
  delay(100);
  Serial.println("Read euler angles...");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void loop() {
  float X = mpu.EulerAngles.x;
  float Y = mpu.EulerAngles.y;
  float Z = mpu.EulerAngles.z;
  if (X < 0){
    X = X*(-1.0);
    Serial.print(X);
  }
  if (TimeSet == 3)
  {
    init_yaw = X;
    init_pitch = Y;
    init_roll = Z;
    Serial.print("init : ");
    Serial.print(init_yaw);
    Serial.print(init_pitch);
    Serial.println(init_roll);
  }
  else
  {
    X = X - init_yaw;
    Y = Y - init_pitch;
    Z = Z - init_roll;
    Serial.print(" value: ");
    Serial.print(X);
    Serial.print(" ");
    Serial.print(Y);
    Serial.print(" ");
    Serial.println(Z);
    /*Serial.print(init_yaw);
     Serial.print(" ");
     Serial.print(init_pitch);
     Serial.print(" ");
     Serial.println(init_roll);*/
  }
 TimeSet += 1;
 count++; // นับจำนวนที่เข้า loop
 int len_str = 0;
 Serial.print("count = ");
 Serial.println(count);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if (count > 3) { // ถ้าloopมากกว่า 3 ครั้ง
ให้เคลียค่า
n1,n2,n3,correctFrame,len_str
    n1 = 0;
    n2 = 0;
    n3 = 0;
    correctFrame = 0;
    len_str = 0;
  }
  mpu.readEuler();
  //char chr[100];
  while (Serial1.available() > 0 ) {
    str[len_str] = Serial1.read();
    len_str += 1;
    count = 0; //ให้ count เป็น 0 เมื่อได้รับ
ข้อมูลจาก BT
    //    Serial.print(str);
    //Serial.print("1");
    //    Serial.print("-----------------count
= ");
    //    Serial.println(count);
  }
  //  if (Serial1.available() <= 0 ) {
  //    //Serial.print("NO SEND");
  //    n1 = 0;
  //    n2 = 0;
  //    n3 = 0;
  //
  //  //    Serial.println(n1);
  //  //    Serial.println(n2);
  //  //    Serial.println(n3);
  //
  //  }
  //Serial.print(str);

  if (len_str > 0) {
   //Serial.print("Test");
   for (int j = 0; j < len_str; j++) {
    //Serial.print(str[j]);
   }
   //Serial.print('\n');
   //Serial.print(len_str);
   sharpCount = 0;
   andCount = 0;
   spaceCount = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//numberframeStr = "";


for (int k = 0; k < len_str; k++) {
  //Serial.print(str[k]);
  if (str[k] == '#') {
    sharpCount += 1;
  }
  if (str[k] == '&') {
    andCount += 1;
  }
  if (str[k] == ' ') {
    if (spaceCount < 3) {
      spaceLoc[spaceCount] = k;
    }
    spaceCount += 1;
  }
}
correctFrame = 0;
//      Serial.println(spaceLoc[0]);
//      Serial.println(spaceLoc[1]);
//      Serial.println(spaceLoc[2]);
    //  if ((sharpCount ==
1)&&(andCount == 1)&&(str[0] == '#')
    //  &&(str[len_str] == '&')){
    if ((sharpCount == 1) && (andCount
== 1) && (str[0] == '#')
        && (str[len_str - 1] == '&') &&
(spaceCount == 3)) {

      Serial.println("correct");
      correctFrame = 1;
    }
    else {
      Serial.println("Incorrect");
      correctFrame = 0;
    }

  }

  /////detect str แยกเป็น 3 ค่า/////

  if (correctFrame == 1) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    fi = 0;
    for (int j = 1; j < spaceLoc[0]; j++) {
      //      numberframeStr += str[j];
    }
    for (int j = spaceLoc[0] + 1; j <
spaceLoc[1]  ; j++) {
      f1[fi] = str[j];
      fi += 1;
    }
    fi = 0;
    for (int j = spaceLoc[1] + 1; j <
spaceLoc[2]  ; j++) {
      f2[fi] = str[j];
      fi += 1;
    }
    fi = 0;
    for (int j = spaceLoc[2] + 1; j <
len_str - 1; j++) {
      f3[fi] = str[j];
      fi += 1;
    }
    /////ค่า 3 ค่าแปลงเป็น float/////
    n1 = atof(f1);
    n2 = atof(f2);
    n3 = atof(f3);
    //numberframe =
numberframeStr.toInt();
    // Serial.println();
    // Serial.println(n1);
    // Serial.println(n2);
    // Serial.println(n3);
    //numberframeStatus = false;
    delay(LOOP_DELAY_ms );
  }
  else {
    if (sharpCount > 1) {
    }
    else {
      delay(LOOP_DELAY_ms );
    }
  }

  Serial.println ("");
  // Serial.print ("numberframe ------>");
  // Serial.println (numberframe);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//Serial.print ("term> ");
//Serial.print (i);
//Serial.print("  ");
Serial.print("S1 >> ");
Serial.print("yaw:");
Serial.print(X);
Serial.print("  ");

Serial.print("roll:");
Serial.print(Y);
Serial.print("  ");

Serial.print("pitch:");
Serial.print(Z);
Serial.println("  ");


Serial.println ("");
//Serial.print ("term> ");
//Serial.print (i);
//Serial.print("  ");
Serial.print("S2 >> ");
Serial.print("yaw:");
Serial.print(n1);
Serial.print(" roll:");
Serial.print(n2);
Serial.print(" pitch:");
Serial.println(n3);

//  if (mpu.EulerAngles.x > n1){ //ถ้า
mpu.EulerAngles.x มีค่ามากกว่า n1 ให้เอา
mpu.EulerAngles.x - n1 ไปเก็บไว้ในตัว
แปร parallelY
//    parallelY = X - n1;
//  }
//  else {                    //ถ้า
mpu.EulerAngles.x มีค่าน้อยกว่า n1 ให้เอา
n1 - mpu.EulerAngles.x ไปเก็บไว้ในตัว
แปร parallelY
//    parallelY = n1 - X;
//  }
//  if (mpu.EulerAngles.y > n2){ //ถ้า
mpu.EulerAngles.y มีค่ามากกว่า n2 ให้เอา
mpu.EulerAngles.y - n2 ไปเก็บไว้ในตัว
แปร parallelP
//    parallelP = Y - n2;
//  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// else {                    //ถ้า
mpu.EulerAngles.y มีค่าน้อยกว่า n2 ให้เอา
n2 - mpu.EulerAngles.y ไปเก็บไว้ในตัว
แปร parallelP
//    parallelP = n2 - Y;
// }
// if (mpu.EulerAngles.z > n3){ //ถ้า
mpu.EulerAngles.z มีค่ามากกว่า n3 ให้เอา
mpu.EulerAngles.z - n3 ไปเก็บไว้ในตัว
แปร parallelR
//    parallelR = Z - n3;
// }
// else {                    //ถ้า
mpu.EulerAngles.z มีค่าน้อยกว่า n3 ให้เอา
n3 - mpu.EulerAngles.z ไปเก็บไว้ในตัว
แปร parallelR
//    parallelR = n3 - Z;
// }

if (abs(X - n1)> 183){
  parallelY = 360 - abs(X - n1);
  Serial.println(parallelY);
}
else {
  parallelY = abs(X- n1);
  Serial.println(parallelY);
}
if (abs(Z - n3)> 183){
  parallelP = 360 - abs(Z - n3);
  Serial.println(parallelP);
}
else {
  parallelP = abs(Z- n3);
  Serial.println(parallelP);
}
 // if (parallelY <= 5 && parallelP <= 5
&& parallelR <= 5) { //parallel
  if ((parallelY <= 5 || (parallelY >=178
&& parallelY <=182)) && (parallelP <=
5 || (parallelP >=178 && parallelP
<=182))) { //parallel
    Serial.println("Parallel---------->");
    digitalWrite(buzzerPin, HIGH);
//Buzzer เตือน
    delay(100);
  }
// if (parallelY <= 5 && parallelP <= 5 )
{ //parallel
//    Serial.println("Parallel---------->");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//    digitalWrite(LedPin, HIGH);
//Buzzer เตือน
//    delay(100);
// }
  else {
    Serial.println("NOT Parallel !! ----------
---->");
    digitalWrite(buzzerPin, LOW);
//Buzzer ไม่เตือน
    delay(100);
  }
  swSet = !digitalRead(swPin); //อ่านค่า
SW
  if (swSet) {
    EEPROM.put(y1Address, X); //นำค่า
จาก mpu.EulerAngles.x เก็บไว้ใน
EEPROME ใน Address y1Address
    EEPROM.put(y2Address, n1);
//นำค่าจาก n1 เก็บไว้ใน EEPROME ใน
Address y2Address
    EEPROM.put(p1Address, Y); //นำค่า
จาก mpu.EulerAngles.y เก็บไว้ใน
EEPROME ใน Address p1Address
    EEPROM.put(p2Address, n2);
//นำค่าจาก n2 เก็บไว้ใน EEPROME ใน
Address p2Address
    EEPROM.put(r1Address, Z); //นำค่า
จาก mpu.EulerAngles.z เก็บไว้ใน
EEPROME ใน Address r1Address
    EEPROM.put(r2Address, n3);
//นำค่าจาก n3 เก็บไว้ใน EEPROME ใน
Address r2Address
    EEPROM.get(y1Address, Y1Set); //
อ่านค่า Y1 มาเก็บไว้ตัวแปร Y1Set
    EEPROM.get(y2Address, Y2Set); //
อ่านค่า Y2 มาเก็บไว้ตัวแปร Y2Set
    EEPROM.get(p1Address, P1Set); //
อ่านค่า P1 มาเก็บไว้ตัวแปร P1Set
    EEPROM.get(p2Address, P2Set); //
อ่านค่า P2 มาเก็บไว้ตัวแปร P2Set
    EEPROM.get(r1Address, R1Set); //
อ่านค่า R1 มาเก็บไว้ตัวแปร R1Set
    EEPROM.get(r2Address, R2Set); //
อ่านค่า R2 มาเก็บไว้ตัวแปร R2Set
  }
  if ((Y1Set == X && Y2Set == n1) &&
(P1Set == Y && P2Set == n2) &&
(R1Set == Z && R2Set == n3)) //เช็คว่า
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าที่บันทึก กับค่าที่รับมาตรงกันไหม ถ้าตรง ให้ LED สว่าง

```
    digitalWrite(LedPin, HIGH);
  else
    digitalWrite(LedPin, LOW);



  lcdMain(X, n1, Y, n2, Z, n3);


  //Serial.println();
  //Serial.println(f1);
  //Serial.println(f2);
  //Serial.println(f3);
  // Serial.print('\n');
  // Serial.print(sharpCount);
  // Serial.print('\n');
  // Serial.print(andCount);
  // Serial.print('\n');
  //  Serial.print(spaceCount);
  //  Serial.print('\n');
  //  Serial.print('\n');
  //  while (Serial1.available() <= 0 ) {
  //    Serial.print("1");
  //  }
  //  Serial.print('\n');
  //  while (Serial1.available() > 0 ) {
  //    //delay(100);
  //    str = Serial1.read();
  //    Serial.print(str);
  //  }
  //Serial.println();
  //delay(500);
}

void lcdMain(float yA, float yB, float
pA, float pB, float rA, float rB) {
  //Clear หน้าจอ
  lcd.setCursor(0, 0);
  lcd.print("                ");
  lcd.setCursor(0, 1);
  lcd.print("                ");
  lcd.setCursor(0, 2);
  lcd.print("                ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lcd.setCursor(0, 3);
lcd.print("                ");

lcd.setCursor(4, 0); //กำหนด Cursor
lcd.print("A");
lcd.setCursor(14, 0);
lcd.print("B");
lcd.setCursor(0, 1);
lcd.print("Y:");
lcd.print(yA); //แสดงค่า yA
lcd.setCursor(10, 1);
lcd.print("Y:");
lcd.print(yB); //แสดงค่า yB

lcd.setCursor(0, 2);
lcd.print("P:");
lcd.print(pA); //แสดงค่า pA
lcd.setCursor(10, 2);
lcd.print("P:");
lcd.print(pB); //แสดงค่า pB

lcd.setCursor(0, 3);
lcd.print("R:");
lcd.print(rA); //แสดงค่า rA
lcd.setCursor(10, 3);
lcd.print("R:");
lcd.print(rB); //แสดงค่า rB
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

โค้ดโปรแกรมของ Device B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <Wire.h>
#include <DFRobot_BNO055.h>
#include <SoftwareSerial.h>
DFRobot_BNO055 mpu;
int i;
int TimeCount = 990;
int TimeSet = 0;
int d;
float init_yaw;
float init_pitch;
float init_roll;
//SoftwareSerial virtualPort(10, 11);
//SoftwareSerial ArduinoSlave(2,3);
SoftwareSerial virtualPort(10, 11);

void setup() {
  Serial.begin(9600);
  //ArduinoSlave.begin(9600);
  virtualPort.begin(9600);

  while (!mpu.init())
  {
    Serial.println("ERROR! Unable to
initialize the chip!");
    delay(50);
  }

//mpu.setMode(mpu.eNORMAL_POWE
R_MODE, mpu.eFASTEST_MODE);
  delay(100);
  Serial.println("Read euler angles...");

}

void loop() {
  String ptr;
  //float f = 123.12F;
  float X = mpu.EulerAngles.x;
  float Y = mpu.EulerAngles.y;
  float Z = mpu.EulerAngles.z;
  if (X < 0){
    X = X*(-1.0);
     Serial.print(X);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Serial.print("TimeCount : ");
Serial.println(TimeCount);

if (TimeSet == 3)
{
  init_yaw = X;
  init_pitch = Y;
  init_roll = Z;
  Serial.print("init : ");
  Serial.print(init_yaw);
  Serial.print(init_pitch);
  Serial.println(init_roll);
}
else
{
  X = X - init_yaw;
  Y = Y - init_pitch;
  Z = Z - init_roll;
  Serial.print(" value: ");
  Serial.print(X);
  Serial.print(" ");
  Serial.print(Y);
  Serial.print(" ");
  Serial.println(Z);

  /*Serial.print(init_yaw);
    Serial.print(" ");
    Serial.print(init_pitch);
    Serial.print(" ");
    Serial.println(init_roll);*/
  }
 TimeSet += 1;
 //  Serial.print("Original value: ");
 //  Serial.print(X);
 //  Serial.print(Y);
 //  Serial.println(Z);

 mpu.readEuler();  /* read euler angle
*/
  ptr = "#" + String(TimeCount) + " " +
String(X) + " " + String(Y) + " " +
String(Z) + "&";
  //ptr = "#"+ String(X) + " " + String(Y)
+ " " + String(Z) + "&";
  char chr[25];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ptr.toCharArray(chr, 25);
int i = ptr.length();
Serial.flush();
int d = Serial.availableForWrite();
//  for (int j = 0; j <= i; j++) {
//    Serial.write(chr[j]);
//    virtualPort.write(chr[j]);
//  }
Serial.write(chr, i);
virtualPort.write(chr, i);
Serial.println();
Serial.print("d : ");
Serial.println(d);

TimeCount += 1;
if (TimeCount > 999) {
  TimeCount = 0;
}
/*
  Serial.print(" ");
   //Serial.print("x = ");
   Serial.print(X);
   //Serial.print("y = ");
   Serial.print(" ");
   Serial.print(Y);
   //Serial.print("z = ");
  Serial.print(" ");
  Serial.println(Z);
*/
  delay(1000);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้