

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย

DEVELOPMENT OF THE ANDROID APPLICATION FOR THE RISK
AREA EXPLORATION CAR CONTROL VIA WIRELESS NETWORK

นางสาวกัญญ์ณัฏฐ์

ปุตตะรังษิ

นายพรสวรรค์

สิทธิชัยวัฒนา

โครงการวิทยุกรรมนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมศาสตรบัณฑิต


สาขาวิศวกรรมไฟฟ้า แขนงวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ปีการศึกษา ๒๕๕๗

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย

DEVELOPMENT OF THE ANDROID APPLICATION FOR THE RISK AREA
EXPLORATION CAR CONTROL VIA WIRELESS NETWORK



นางสาวกัญญ์ณัฏฐ์ ปุสสระรัมย์
นายพรสวรรค์ ลิทธิชัยวัฒนา

โครงการวิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า แขนงวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ปีการศึกษา 2557

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย

ปีการศึกษา 2557

โดย

นางสาวกัญญ์ณัฏฐ์ ปุสสระรัมย์
นายพรสวรรค์ สิทธิชัยวัฒนา

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์วัชรชัย วิริยะสุทธีวงศ์

บทคัดย่อ

โครงการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย มีวัตถุประสงค์เพื่อสร้างและออกแบบแอปพลิเคชัน ที่สามารถควบคุมรถบังคับให้สำรวจพื้นที่ต่างๆได้ โดยผ่านระบบเครือข่ายไร้สาย (Wi-Fi) ในส่วนของรถบังคับได้ออกแบบให้สามารถเคลื่อนที่ได้ในแนวระนาบ ซึ่งมีเซ็นเซอร์ที่ใช้ในการตรวจจับก๊าซไวไฟ และมีกล้อง IP Camera ที่ใช้ในการส่งสัญญาณภาพจากรถสำรวจมายังหน้าจอแท็บเล็ต โดยควบคุมบังคับผ่านทางแท็บเล็ตบนระบบปฏิบัติการแอนดรอยด์ ในการทดลองแอปพลิเคชันสามารถทำการเชื่อมต่อกับรถสำรวจและสามารถควบคุมการเคลื่อนที่ของรถสำรวจได้อย่างสมบูรณ์ จากการทดลองค่าเวลาที่ไ้การเคลื่อนที่ของรถสำรวจในแผนที่จำลองเทียบผลที่ไ้กับค่าเวลาที่คำนวณได้จริง ปรากฏว่า มีค่าเปอร์เซ็นต์ความคลาดเคลื่อนอยู่ที่ 14.20% การทดลองระยะทางในการควบคุมการทำงานของรถสำรวจ จากผลการทดลองสามารถควบคุมรถสำรวจได้ไกลที่สุดระยะ 50 เมตร ในบริเวณที่โล่งแจ้งและสามารถควบคุมรถสำรวจได้ระยะ 20 เมตร ในบริเวณภายในอาคาร การทดสอบในส่วนของกล้อง IP Camera สามารถแสดงผลและบังคับทิศทางการหมุนได้ถูกต้องในทุกๆตำแหน่งองศา และในส่วนของ การทดลองการตรวจจับก๊าซไวไฟ ระบบสามารถทำงานได้อย่างมีประสิทธิภาพ ซึ่งแสดงผลการตรวจจับวัตถุที่มีสารไวไฟ เช่น ไฟแช็ค แอลกอฮอล์ และยาฉีดกำจัดยุง ได้อย่างถูกต้อง

คำสำคัญ รถบังคับ / แอนดรอยด์ แอปพลิเคชัน / ก๊าซไวไฟ

**DEVELOPMENT OF THE ANDROID APPLICATION FOR THE RISK AREA
EXPLORATION CAR CONTROL VIA WIRELESS NETWORK**

Academic Year 2014

By

Ms. Kannanut

Mr. Pornsawan

Pussarungsi

Sitthichaiwattana

Project Report Advisor

Asst. Prof. Watcharachai Wiriyasuttiwong

ABSTRACT

This project is the development of applications on Android operating system for controlling the surveying car in risky area through a wireless network. The main objective is to design and build an application that can control the surveying car to explore in any area through a wireless network(Wi-Fi). In terms of the surveying car, It is designed to move in the horizontal direction. It have a sensor that can detect flammable gas and it have IP camera for sending the picture signal from the surveying car to the screen on the tablet by controlling through on Android tablet operating system. In the experiment of application can connect with the surveying car and control the movement of the surveying car perfectly. The results of experiment show that the time for the movement of the surveying car in the model map compare with the real-time calculated that the percentage of error is 14.20%. The experiment of trial distance for controlling the surveying car show that the longest distance is 50 meters in the open space and 20 meters in the building. In the experiment of IP camera show that it can be displayed and controlled the direction of rotation completely in every position degrees. In the experiment of detecting flammable gas show that the system can operate efficiently. It can capture flammable gas accurately such as Alcohol lighter, Mosquito spray.

Keyword Radio Controlled Car / Android Application / Flammable Gas

กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้สำเร็จลุล่วงไปด้วยดี ด้วยความช่วยเหลืออย่างดียิ่งจากท่านอาจารย์ ผศ. วัชรชัย วิริยะสุทธิวงษ์ ที่ปรึกษาโครงการ ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆอันเป็นประโยชน์อย่างยิ่งในการทำโครงการ อีกทั้งยังช่วยแก้ปัญหาต่างๆที่เกิดขึ้นระหว่างการดำเนินงานอีกด้วย ขอขอบพระคุณคณาจารย์คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมไฟฟ้า และบุคลากรทุกท่าน ที่ให้คำปรึกษาและอำนวยความสะดวกต่างๆเพื่อให้โครงการนี้สำเร็จลุล่วงด้วยดี ขอขอบพระคุณผู้ทรงคุณวุฒิทุกท่านที่ให้ความอนุเคราะห์ในการตรวจสอบ ให้คำแนะนำ แก้ไขโครงการให้มีคุณภาพ ทั้งยังวิจารณ์ผลงานทำให้โครงการมีความสมบูรณ์มากยิ่งขึ้น นอกจากนี้ขอขอบคุณเพื่อนๆ สาขาวิศวกรรมไฟฟ้าทุกคนที่เป็นกำลังใจและให้ความช่วยเหลือในการทำโครงการเรื่องนี้ สุดท้ายนี้ ขอขอบพระคุณบิดามารดาและครอบครัวที่เลี้ยงดู อบรมสั่งสอน ตลอดจนช่วยเหลือและให้กำลังใจเสมอมา



สารบัญ

| | หน้า |
|--|-----------|
| บทคัดย่อภาษาไทย | ก |
| บทคัดย่อภาษาอังกฤษ | ข |
| กิตติกรรมประกาศ | ค |
| สารบัญตาราง | ฉ |
| สารบัญภาพ | ช |
| | |
| บทที่ 1 บทนำ | 1 |
| 1.1 ความเป็นมาของโครงการ | 1 |
| 1.2 วัตถุประสงค์ | 2 |
| 1.3 ขอบเขตของโครงการ | 2 |
| 1.4 ประโยชน์ที่คาดว่าจะได้รับ | 2 |
| บทที่ 2 ทฤษฎีที่เกี่ยวข้อง | 3 |
| 2.1 ทฤษฎีทางด้านฮาร์ดแวร์ | 3 |
| 2.2 ทฤษฎีทางด้านซอฟต์แวร์ | 12 |
| บทที่ 3 การวิเคราะห์และออกแบบระบบ | 13 |
| 3.1 หลักการทำงานด้านฮาร์ดแวร์ | 16 |
| 3.2 หลักการทำงานด้านซอฟต์แวร์ | 17 |
| บทที่ 4 การทดสอบและผลการทดสอบโครงการ | 26 |
| 4.1 การเคลื่อนที่ของรถสำรวจในพื้นที่ทางตรง | 26 |
| 4.2 การเคลื่อนที่ของรถสำรวจในแผนที่จำลอง | 28 |
| 4.3 ระยะเวลาในการควบคุมการทำงานของรถสำรวจ | 30 |
| 4.4 การควบคุมบังคับทิศทางการหมุนของกล้อง IP Camera | 31 |
| 4.5 การตรวจจับก๊าซไวไฟ MQ7 Gas Sensor Module | 35 |

สารบัญ(ต่อ)

| | หน้า |
|-------------------------------------|------|
| บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ | 38 |
| 5.1 สรุปผลของโครงการ | 38 |
| 5.2 ปัญหาและข้อเสนอแนะ | 39 |
| เอกสารอ้างอิง | 40 |
| ภาคผนวก | 41 |
| ประวัติผู้จัดทำโครงการ | 70 |



สารบัญตาราง

| ตาราง | หน้า |
|--|------|
| 2.1 การเชื่อมต่อ Motor Driver (EVO24V9.1) | 7 |
| 2.2 การควบคุมการทำงาน Motor Driver (EVO24V9.1) | 8 |
| 4.1 ผลการทดสอบการเคลื่อนที่ของรถสำรวจในทางตรง | 27 |
| 4.2 ผลการทดสอบการเคลื่อนที่ของรถสำรวจในแผนที่จำลอง | 30 |
| 4.3 ผลการทดสอบระยะทางในการควบคุมการทำงานของรถสำรวจ | 30 |
| 4.4 ผลการทดสอบการควบคุมทิศทางการหมุนของกล้อง IP Camera | 32 |
| 4.5 ผลการทดสอบการตรวจจับก๊าซไฟไว | 35 |



สารบัญภาพ

| รูป | หน้า |
|---|------|
| 2.1 ส่วนประกอบต่างๆของ Servo motor | 4 |
| 2.2 การป้อนสัญญาณความกว้างของพัลส์ | 4 |
| 2.3 ความกว้างของสัญญาณของพัลส์ | 5 |
| 2.4 ตัวอย่าง EVO24V9.1 | 6 |
| 2.5 การต่อใช้งาน EVO24V9.1 | 7 |
| 2.6 ตัวอย่าง Flyport Ethernet Module | 9 |
| 2.7 ตัวอย่าง USB Nest | 11 |
| 2.8 ตัวอย่าง miniUSB Programmer | 11 |
| 3.1 แผนภาพ Block Diagram ของระบบ | 13 |
| 3.2 ทิศทางการไหลของข้อมูลตรวจสอบ | 14 |
| 3.3 Flyport Ethernet Module | 14 |
| 3.4 วงจรการเชื่อมต่อ Flyport Ethernet | 15 |
| 3.5 แผนภาพ Block Diagram ของ Motor Driver EVO24V9.1 | 15 |
| 3.6 Motor Driver EVO24V9.1 | 15 |
| 3.7 วงจรการเชื่อมต่อ Drive Motor EVO24V9.1 | 16 |
| 3.8 แผนภาพ Block Diagram ของ MQ7 Gas Sensor | 16 |
| 3.9 MQ7 Gas Sensor Module | 17 |
| 3.10 วงจรการเชื่อมต่อ MQ7 Gas Sensor Module | 17 |
| 3.11 แผนภาพ Block Diagram ของ IP Camera | 18 |
| 3.12 วงจรการเชื่อมต่อ MQ7 Gas Sensor Module | 18 |
| 3.13 วงจรควบคุมการเปิด-ปิดกล้อง IP Camera | 18 |
| 3.14 วงจรควบคุมการหมุนของกล้อง IP Camera | 19 |
| 3.15 แผนภาพ Block Diagram ของ Access point | 19 |

สารบัญภาพ(ต่อ)

| รูป | หน้า |
|--|------|
| 3.16 ZyXEL WAP3205 v2 | 20 |
| 3.17 แผนภาพ Block Diagram ของ Lamp | 20 |
| 3.18 หลอดไฟ LED | 21 |
| 3.19 วงจรควบคุมการเปิด-ปิดไฟที่ด้านหน้าของหุ่นยนต์ | 21 |
| 3.20 การติดตั้งอุปกรณ์ต่างๆภายในรถสำรวจ | 22 |
| 3.21 การออกแบบหน้าจอการควบคุมหุ่นยนต์ | 23 |
| 3.22 สัญลักษณ์แสดงการแจ้งเตือนเมื่อจับก๊าซไวไฟได้ | 24 |
| 3.23 ภาพจากกล้อง IP Camera ในที่ที่มีแสงสว่างมากหรือเวลากลางวัน | 24 |
| 3.24 ภาพจากกล้อง IP Camera ในที่ที่มีแสงน้อยหรือเวลากลางคืน | 25 |
| 4.1 การเคลื่อนที่ในทางตรงบนพื้นที่มีน้ำขัง | 27 |
| 4.2 การเคลื่อนที่ในทางตรงบนพื้นทราย | 28 |
| 4.3 การเคลื่อนที่ในทางตรงบนพื้นที่มีก้อนหิน | 28 |
| 4.4 แผนที่จำลองที่ใช้ในการทดสอบรถสำรวจ | 29 |
| 4.5 แผนที่จากสถานที่จริงที่ใช้ในการทดสอบรถสำรวจ | 29 |
| 4.6 การทดสอบการควบคุมการหมุนของกล้อง IP Camera | 31 |
| 4.7 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 0 องศา | 32 |
| 4.8 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 45 องศา | 33 |
| 4.9 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 90 องศา | 33 |
| 4.10 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 135 องศา | 34 |
| 4.11 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 180 องศา | 35 |
| 4.12 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้ไฟแช็ค | 36 |
| 4.13 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้แอลกอฮอล์ | 36 |
| 4.14 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้แก๊สกระป๋องและยาฉีดกำจัดยุง | 37 |

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

การเกิดเพลิงไหม้ ดึกถล่ม หรืออุบัติเหตุจากสารเคมีนั้นสามารถเกิดขึ้นได้อย่างไม่คาดคิดในปัจจุบัน ทำให้เกิดการสูญเสียทั้งชีวิตและทรัพย์สินของผู้ประสบภัยจากเหตุการณ์ดังกล่าว ในการเข้าช่วยเหลือผู้ประสบภัยนั้น ปกติทั่วไปแล้วเจ้าหน้าที่หน่วยกู้ภัยจะเป็นผู้ทำการช่วยเหลือ อย่างไรก็ตาม การเข้าช่วยเหลือนั้นบางพื้นที่หรือบางเหตุการณ์ที่มีความอันตรายและเสี่ยงเกินกว่าที่จะช่วยเหลือได้ ดังนั้นจึงได้มีการนำเทคโนโลยีมาเพื่อช่วยลดอันตรายที่เกิดจากการเข้าช่วยเหลือผู้ประสบภัยให้กับเจ้าหน้าที่หน่วยกู้ภัย เช่น หุ่นยนต์กู้ภัย โดรนสำหรับการสำรวจ และรถบังคับวิทยุติดกล้องสำหรับสำรวจภาคพื้นดิน เป็นต้น ปัจจุบันหุ่นยนต์ โดรน และรถบังคับ วิทยุมีประสิทธิภาพที่ดีมากยิ่งขึ้น ใช้วัสดุที่แข็งแรงในการสร้างและประกอบ จึงทำให้หุ่นยนต์ โดรน และรถบังคับ มีสมรรถนะที่ดีขึ้น และมีการพัฒนาเพื่อให้สามารถทำงานแทนมนุษย์ได้ โดยนำไปใช้ในการเข้าถึงพื้นที่ที่มีความเสี่ยงหรือพื้นที่ที่มนุษย์ไม่สามารถเข้าถึงได้ แต่หุ่นยนต์ยังมีข้อเสียตรงที่มีราคาแพง และมีกลไกในการทำงานที่ซับซ้อน ส่วนโดรนนั้นก็เช่นกันเพราะนอกจากจะมีราคาแพงแล้ว ยังต้องใช้ผู้เชี่ยวชาญในการบังคับและเกิดเสียหายได้ง่าย ทำให้รถบังคับคือสิ่งที่น่าสนใจมากที่สุด ในส่วนของรถบังคับนั้นมีราคาที่ค่อนข้างถูก มีความแข็งแรงต่อการใช้งาน และการใช้งานเป็นที่แพร่หลายจึงง่ายต่อการนำไปใช้งานได้

จึงสามารถที่จะนำไปพัฒนาใช้งานต่อได้ เช่น สามารถติดกล้องที่มีคุณภาพ มีความคมชัด และสามารถส่งข้อมูลผ่านเครือข่ายไร้สาย Wi-Fi ได้ ซึ่งเครือข่ายไร้สาย Wi-Fi นั้นมีข้อดีกว่า Bluetooth คือ สามารถส่งข้อมูลได้เร็ว ส่งในระยะทางที่ไกลกว่าและเป็นระบบที่มีการทำงานคล้ายกับระบบ Network แบบมีสายมากที่สุด โดยในปัจจุบันนั้นได้มีการนำมาใช้งานถึงมาตรฐาน IEEE 802.11ac แล้ว และในอนาคตจะมีการพัฒนาให้มีประสิทธิภาพดียิ่งขึ้น สำหรับสมาร์ตโฟนในปัจจุบันนั้นการเข้าใช้งานอินเทอร์เน็ตผ่านเครือข่ายไร้สาย Wi-Fi สามารถทำได้สะดวกรวดเร็วเกือบทุกที่และทุกเวลา ด้วยเหตุนี้โครงการนี้จึงได้ทำการประยุกต์ความรู้ด้านเทคโนโลยี Wi-Fi กับ

สมาร์ทโฟนที่ทำงานด้วยระบบปฏิบัติการแอนดรอยด์ เข้าด้วยกันเพื่อนำไปใช้ในการควบคุมรถบังคับให้เคลื่อนที่ไปในทิศทางต่างๆหรือเข้าถึงพื้นที่ที่ที่ยากลำบาก และเพื่อพัฒนาให้แอปพลิเคชันสามารถใช้งานร่วมกันกับแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์โดยผ่านเครือข่ายไร้สาย Wi-Fi ได้

1.2 วัตถุประสงค์

1.2.1 เพื่อพัฒนาแอปพลิเคชันที่ใช้ในการควบคุมรถบังคับให้เข้าถึงพื้นที่ที่ที่ยากลำบากได้

1.2.2 เพื่อพัฒนาแอปพลิเคชันให้มีการแสดงสัญญาณภาพจากกล้อง IP Camera มายังหน้าจอแสดงผลของแท็บเล็ต และสามารถควบคุมทิศทางรถของกล้อง IP Camera ที่ติดตั้งอยู่บนรถบังคับได้

1.2.3 เพื่อพัฒนาแอปพลิเคชันในการแจ้งเตือนการตรวจก๊าซไวไฟที่ติดตั้งอยู่บนรถบังคับ

1.3 ขอบเขตของโครงการ

1.3.1 ระบบสามารถควบคุมรถบังคับด้วยแท็บเล็ตบนระบบปฏิบัติการแอนดรอยด์

1.3.2 ระบบสามารถควบคุมรถบังคับให้เคลื่อนที่ไปในทิศทางต่างๆ ได้ในแนวระนาบผ่านระบบเครือข่ายไร้สาย (Wi-Fi)

1.3.3 ระบบมีการส่งสัญญาณภาพจากรถบังคับมายังหน้าจอแสดงผลของแท็บเล็ต

1.3.4 ระบบสามารถตรวจก๊าซไวไฟและแจ้งเตือนไปยังหน้าจอแสดงผลของแท็บเล็ต

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 แอปพลิเคชันสามารถทำการควบคุมรถบังคับให้เข้าถึงพื้นที่ที่ที่ยากลำบากได้

1.4.2 แอปพลิเคชันสามารถแสดงสัญญาณภาพของกล้อง IP Camera มายังหน้าจอแสดงผลของแท็บเล็ตได้ และสามารถบังคับทิศทางรถของกล้อง IP Camera ที่ติดตั้งอยู่บนรถบังคับได้

1.4.3 แอปพลิเคชันสามารถทำการบอกสถานะของก๊าซไวไฟและแจ้งเตือนมายังหน้าจอแสดงผลของแท็บเล็ตได้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ในการออกแบบแอปพลิเคชันและรถบังคับนั้น จำเป็นต้องใช้หลักการและทฤษฎีต่างๆ ที่เกี่ยวข้องมาช่วยในการสร้าง ซึ่งในการพิจารณาหลักการและทฤษฎีที่จะนำมาประยุกต์ใช้ในการสร้างแอปพลิเคชันและรถบังคับ ให้ทำงานได้อย่างมีประสิทธิภาพนั้น แบ่งเป็น 2 ส่วนคือ ฮาร์ดแวร์ และซอฟต์แวร์ จึงจำเป็นต้องอาศัยหลักการและทฤษฎีดังต่อไปนี้

2.1 ทฤษฎีทางด้านฮาร์ดแวร์

ในการสร้างรถสำรวจพื้นที่เสี่ยงภัยนั้น ต้องใช้อุปกรณ์อิเล็กทรอนิกส์หลายอย่าง จึงจำเป็นต้องใช้หลักการและทฤษฎีต่างๆที่เกี่ยวข้องมาช่วยในการสร้าง ซึ่งมีอุปกรณ์ต่างๆดังนี้

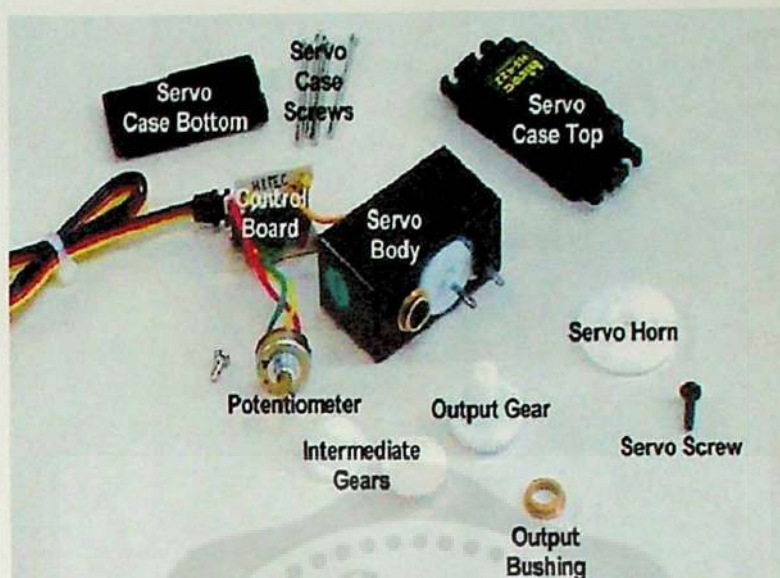
2.1.1 ทฤษฎี DC motor และ Servo motor

2.1.1.1 หลักการเบื้องต้นของมอเตอร์

มอเตอร์คือ เครื่องกลไฟฟ้าที่เปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานกล โดยทั่วไปมอเตอร์สามารถแบ่งออกเป็น 2 ชนิดได้แก่ มอเตอร์ไฟฟ้ากระแสตรงหรือ DC motor (Direct Current motor) และมอเตอร์ไฟฟ้ากระแสสลับ หรือ AC motor (Alternating Current motor) โดยสามารถที่จะควบคุมการทำงานของมอเตอร์ทั้ง 2 ชนิดให้เป็นไปตามที่เราต้องการได้ การที่จะให้มอเตอร์หมุนไปในทิศทางใดนั้นขึ้นอยู่กับความสัมพันธ์ระหว่างทิศทางของเส้นแรงแม่เหล็กระหว่างขั้วเหนือและขั้วใต้กับทิศของกระแสไฟฟ้าที่ไหลผ่านขดลวดอาร์เมเจอร์

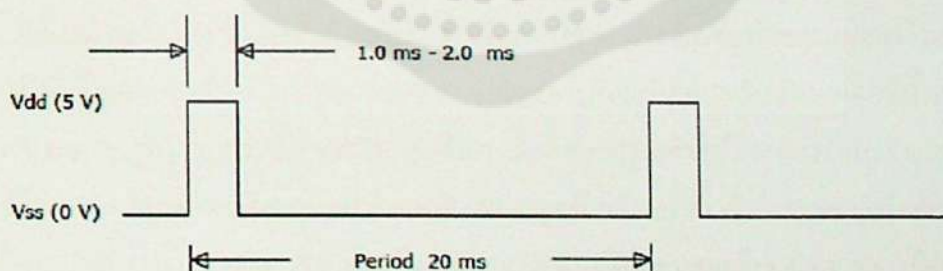
2.1.1.2 หลักการทำงานของ Servo motor

Servo motor เป็นมอเตอร์ไฟฟ้ากระแสตรง (DC motor) ที่ถูกประกอบรวมกับชุดเกียร์ และส่วนควบคุมต่างๆ ไว้ในโมดูลเดียวกันหรือภายในกล่องพลาสติกเดียวกันโดยมอเตอร์ชนิดนี้จะมีสายต่อใช้งานเพียง 3 เส้นเท่านั้น คือ VCC, GND และสายสัญญาณควบคุม (Control line) ซึ่งสามารถควบคุมให้มอเตอร์หมุนซ้ายหรือขวาได้จากสายสัญญาณเพียงเส้นเดียว โดยสัญญาณที่ใช้ควบคุมนี้จะเป็นสัญญาณพัลส์วามอด (PWM) แบบ TTL level ระดับแรงดันที่จ่ายให้มอเตอร์นี้จะอยู่ในช่วงประมาณ 4 ถึง 6 โวลต์ ขึ้นอยู่กับคุณสมบัติของมอเตอร์แต่ละตัว

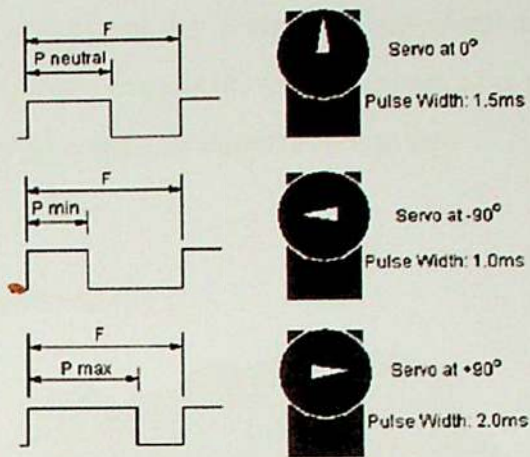


รูปที่ 2.1 ส่วนประกอบต่างๆของ Servo motor

การควบคุมการทำงานของ Servo motor ทำได้โดย การป้อนสัญญาณความกว้างของพัลส์ให้กับมอเตอร์ซึ่งตำแหน่งและทิศทางการหมุนของมอเตอร์นี้จะขึ้นอยู่กับขนาดของความกว้างของพัลส์นั้นๆ โดยทั่วไปแล้วความกว้างของสัญญาณพัลส์จะมีจุดให้อ้างอิง 3 จุด ดังรูปที่ 2.2 และรูปที่ 2.3



รูปที่ 2.2 การป้อนสัญญาณความกว้างของพัลส์



รูปที่ 2.3 ความกว้างของสัญญาณของพัลส์

สัญญาณความกว้างพัลส์ขนาด 1.5 ms จะควบคุมให้ Servo motor หมุนไปอยู่ที่ตำแหน่งมุม 0 องศา หรือ จุดกึ่งกลางของมอเตอร์

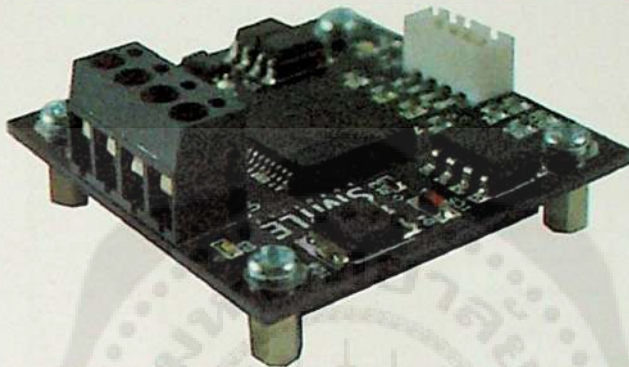
สัญญาณความกว้างพัลส์ขนาด 1 ms จะควบคุมให้ Servo motor หมุนไปอยู่ที่ตำแหน่งมุม - 90 องศา หรือในทิศทางทวนเข็มนาฬิกา

สัญญาณความกว้างพัลส์ขนาด 2 ms จะควบคุมให้ Servo motor หมุนไปอยู่ที่ตำแหน่งมุม + 90 องศา หรือในทิศทางตามเข็มนาฬิกา

หมายเหตุ ค่าความกว้างพัลส์และระยะของสากการหมุนของมอเตอร์ที่อธิบายด้านบนนั้นเป็นเพียงค่า ประมาณเท่านั้น ทั้งนี้ระยะการหมุน และขนาดของพัลส์ที่ควบคุมการทำงานของมอเตอร์ในแต่ละยี่ห้ออาจจะไม่เท่ากัน ดังนั้นในการใช้งานจึงควรศึกษารายละเอียดของมอเตอร์ในแต่ละรุ่นที่นำมาใช้ ซึ่งโดยปกติแล้วรายละเอียดต่างๆ ของมอเตอร์มักจะมีติดมากับตัวมอเตอร์นั้นๆ อยู่แล้ว ส่วนการที่จะควบคุมให้มอเตอร์หมุนเป็นมุมอื่นๆ นั้นก็สามารถทำได้โดยการป้อนสัญญาณพัลส์เป็นระดับความกว้างต่างๆ โดยอ้างอิงจากจุด ทั้ง 3 จุดที่กล่าวมานี้ ตัวอย่างเช่น ถ้าต้องการให้มอเตอร์หมุนไปที่มุม - 45 องศา ก็จะต้องป้อนสัญญาณพัลส์ที่มีความกว้าง 1.25 ms เป็นต้น และสัญญาณพัลส์นี้จะต้องจ่ายให้มอเตอร์ทุกๆ 20 ms เพื่อรักษาสภาพตำแหน่งของมอเตอร์ไว้โดยหลักการก็คือ จะอาศัยการเปรียบเทียบช่วงเวลาของความกว้างพัลส์ที่จ่ายให้กับมอเตอร์ทางขาสัญญาณควบคุมกับค่าเวลาของวงจร RC ภายในบอร์ดควบคุมในตัวของมอเตอร์ ซึ่งค่าเวลาของวงจร RC นี้จะมีการเปลี่ยนแปลงตามการหมุนของมอเตอร์ เนื่องจากตัวต้านทานปรับค่าจะถูกยึดติดอยู่กับแกนหมุนของมอเตอร์ ซึ่งการหมุนของมอเตอร์จะทำให้ค่าความต้านทานของตัวต้านทานปรับค่า (VR) เปลี่ยนแปลงไปเป็นผลทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงตามไปด้วย โดยในขณะที่เราป้อนสัญญาณความกว้างพัลส์ให้กับมอเตอร์ทางขาสัญญาณควบคุม สัญญาณนี้จะถูก

นำไปเปรียบเทียบกับค่าเวลาของวงจร RC หากค่าทั้ง 2 ไม่เท่ากันมอเตอร์ก็จะหมุนทำให้ค่าเวลาของวงจร RC เปลี่ยนแปลงจนกระทั่งค่าเวลาความกว้างพัลส์ของ วงจร RC เปลี่ยนแปลงจนเท่ากับสัญญาณพัลส์ทางควบคุม (Control line) มอเตอร์จึงจะหยุดหมุน

2.1.2 Motor Driver (EVO24V9.1)



รูปที่ 2.4 ตัวอย่าง EVO24V9.1

บอร์ดขับ DC Motor รุ่น EVO24V9.1 เป็นบอร์ดขับ DC Motor ที่ใช้ไอซี Full bridge motor drive เบอร์ VN3SP30 ของบริษัท ST Microelectronic ซึ่งสามารถขับกระแสได้สูงสุด 30A ตัวบอร์ดถูกออกแบบให้สามารถใช้แรงดันไฟฟ้าอินพุตสูงสุดที่ 28V มีวงจรป้องกันสัญญาณรบกวนจากมอเตอร์ด้วย Opto Isolator มีวงจรป้องกันการต่อไฟเลี้ยงกลับขั้ว มีความทนทานสูงบอร์ดมีขนาดเล็กน้ำหนักเบาใช้งานง่าย สามารถต่อสัญญาณควบคุมจากไมโครคอนโทรลเลอร์หรือวงจรควบคุมอื่นๆได้ เหมาะสำหรับใช้ขับ DC Motor สำหรับหุ่นยนต์หรือประยุกต์ใช้ควบคุมอุปกรณ์ไฟฟ้าอื่นๆ

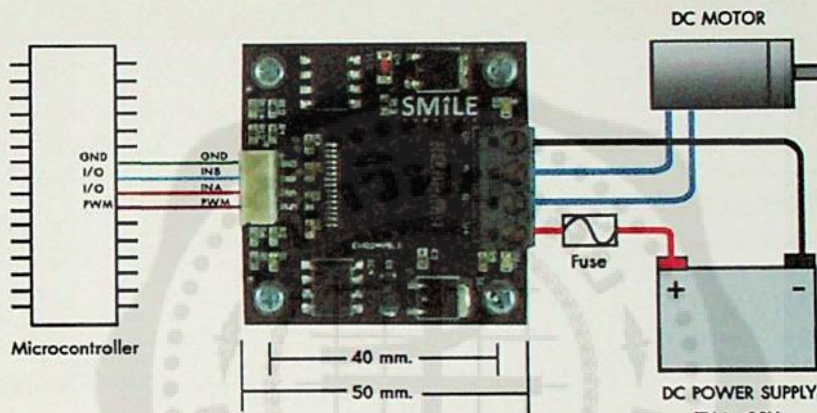
2.1.2.1 คุณสมบัติ EVO24V9.1

Full bridge motor drive เบอร์ VN3SP30 ขับกระแสสูงสุดชั่วขณะ 30 A ขับกระแสไฟฟ้าต่อเนื่อง 9 A ที่แรงดัน 24 VDC แรงดันไฟฟ้าอินพุต VCC 7-28 VDC แรงดันไฟฟ้าเอาต์พุต 0.98 x VCC ควบคุมความเร็วมอเตอร์ด้วยการปรับความกว้างของสัญญาณ PWM ที่ความถี่ไม่เกิน 5 kHz แยกสัญญาณควบคุมไฟฟ้าด้วย Opto Isolator กลับทางหมุนด้วยสัญญาณลอจิกอินพุต 3-5.5 V แสดงสถานะทิศทางหมุนด้วย LED มีวงจรป้องกันการจ่ายไฟเลี้ยงกลับขั้ว ตัดการทำงานเมื่อมี

อุณหภูมิสูงเกินกว่า 100 °C มีฮีทซิงค์ช่วยระบายความร้อน ขนาด PCB 50x50 mm. และน้ำหนักรวม 22 g.

2.1.2.2 การต่อใช้งานและลักษณะสัญญาณที่ใช้ควบคุม

การต่อสัญญาณควบคุมเข้ากับตัวบอร์ดใช้ CONNECTOR: 2.50 mm. 04 PINS HOUSING (FEMALE) เบอร์ 200104/RH และใช้ CRIMP TERMINAL CONNECTOR 2.50mm. เบอร์ 2001T/RH



รูปที่ 2.5 การต่อใช้งาน EVO24V9.1

ตารางที่ 2.1 การเชื่อมต่อ Motor Driver (EVO24V9.1)

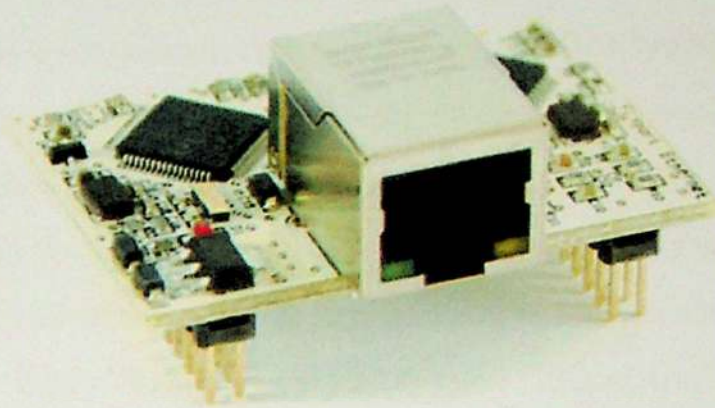
| PIN | Signal | Description |
|------|--------------------|---|
| V+ | +VCC | ใช้ไฟเลี้ยงระบบที่แรงดันไฟ +7...+28 VDC |
| V- | Power GND | กราวด์ของไฟเลี้ยง |
| A | Motor winding (+M) | แรงดันไฟฟ้าเอาต์พุตมอเตอร์ |
| B | Motor winding (-M) | แรงดันไฟฟ้าเอาต์พุตมอเตอร์ |
| IN A | IN A | สัญญาณลอจิกอินพุต หมุนตามเข็มนาฬิกา |
| IN B | IN B | สัญญาณลอจิกอินพุต หมุนทวนเข็มนาฬิกา |
| PWM | PWM | สัญญาณพัลส์วิดท์มอดูเลต (PWM) อินพุต |
| GND | GND | กราวด์ของสัญญาณควบคุม |

ตารางที่ 2.2 การควบคุมการทำงาน Motor Driver (EVO24V9.1)

| PWM | IN A | IN B | Operating mode |
|-----|------|------|------------------------|
| ON | 0 | 0 | Free Run Stop |
| ON | 1 | 1 | Free Run Stop |
| OFF | 1 | 1 | Brake to VCC |
| OFF | 0 | 0 | Brake to GND |
| ON | 1 | 0 | Clockwise (CW) |
| ON | 0 | 1 | Counterclockwise (CCW) |

2.1.3 Flyport Ethernet module

Flyport เป็น โมดูลการสื่อสาร ที่พัฒนาโดย openPicus ประเทศอิตาลี มี ไมโครคอนโทรลเลอร์ PIC24FJ ขนาด 16bit ในตัว โดยมีการโปรแกรม bootloader ไว้แล้ว ทำให้สามารถโปรแกรมสั่งงานด้วย openPicus IDE ที่ทางผู้ผลิตได้เตรียมไว้ให้ผ่านทาง serial port ให้กับ โมดูลได้โดยตรง ทำให้การใช้งานไม่จำเป็นต้องต่อไมโครคอนโทรลเลอร์ภายนอกเพิ่มเติมอีก นอกจากนี้ทางผู้ผลิตยังได้ปล่อย license ของฮาร์ดแวร์ (Open Hardware) สำหรับให้นักพัฒนานำไปต่อยอดได้ด้วย จึงทำให้โมดูลนี้กำลังจะเป็นที่นิยมและรู้จักไปทั่วโลก โดยเฉพาะกับนักพัฒนาที่ต้องการสร้างระบบควบคุมหรือสั่งงานอุปกรณ์ต่างๆ ผ่านระบบ Wi-Fi Ethernet หรือ GPRS ให้ทำงานได้ตามคำสั่งที่ต้องการ



รูปที่ 2.6 ตัวอย่าง Flyport Ethernet module

2.1.2.1 ความน่าสนใจของ Flyport

ลดเวลาในการพัฒนาอุปกรณ์: ด้วย IDE ที่ทางผู้ผลิตได้พัฒนาขึ้นมาและให้ download มาใช้งานได้ฟรี จะช่วยลดระยะเวลาในการพัฒนาอุปกรณ์และโปรแกรมจาก 6 เดือน ให้เหลือเพียง 1-3 อาทิตย์ได้

เชื่อถือได้: ทางผู้ผลิตจะมีการสนับสนุนและช่วยเหลือในปัญหาต่างๆของนักพัฒนาอยู่ตลอดเวลาโดยจะมีกระดานสนทนาเพื่อตอบข้อซักถามต่างๆที่มาจากนักพัฒนาทั่วโลก

ลดค่าใช้จ่าย: เนื่องจาก Flyport มีไมโครคอนโทรลเลอร์ในตัวอยู่แล้ว จึงไม่จำเป็นต้องต่อกับไมโครคอนโทรลเลอร์ภายนอกอีก ซึ่งจะเป็นการประหยัดค่าใช้จ่ายในการพัฒนาอุปกรณ์ลงได้

เคลื่อนย้ายและต่อใช้งานได้ง่าย: เนื่องจากโมดูล Flyport มีขนาดเล็ก และทำงานคล้ายกับไมโครคอนโทรลเลอร์ตัวหนึ่ง มีขาพอร์ตเป็น Header pin ให้ต่อใช้งานได้ง่ายสะดวก และสามารถถอดเปลี่ยนหรือเคลื่อนย้ายไปยังบอร์ดอิเล็กทรอนิกส์อื่นๆ ได้ง่าย

เป็น Embedded Web Server: ทำให้นักพัฒนาสามารถสร้างระบบควบคุมหรือสั่งงานอุปกรณ์ผ่าน Web browser ด้วย Smartphone Tablet หรือคอมพิวเตอร์ได้

สร้างตัว Programmer ได้ง่าย: ในการ flash โปรแกรมให้กับ Flyport ไม่จำเป็นต้องมี USB Nest หรือ miniUSB Programmer จากทางผู้ผลิต หรือตัว USB-to-Serial ซึ่งนักพัฒนาสามารถที่จะออกแบบและสร้างเองได้ไม่ยาก และหาได้ง่ายทั่วไปตามท้องตลาด

สนับสนุน PPS (Peripheral Pin Select): Flyport สามารถปรับเปลี่ยนการทำงานของฮาร์ดแวร์ตามที่ต้องการได้ โดยขาพอร์ตแต่ละขาสามารถเปลี่ยนแปลงการทำงานใหม่ได้ด้วยซอฟต์แวร์ เช่น การกำหนดให้ทำงานเป็น Digital I/O, UARTs, SPI, PWMs หรือ Interrupts ซึ่งจะช่วยให้มีความยืดหยุ่นในการใช้งานมากขึ้น และยังสามารถสร้างบอร์ดได้หลากหลายรูปแบบ

Framework และ IDE: openPicus ได้พัฒนา Flyport framework บนฐานของ freeRTOS ทำให้ระบบการจัดการสื่อสารกับ stack และแอปพลิเคชันของ Flyport เป็นไปอย่างมีประสิทธิภาพ และด้วย IDE ที่ผู้ผลิตได้เตรียมไว้ให้ จะช่วยให้นักพัฒนาไม่ต้องลำบากในการใช้งานระบบ freeRTOSที่มีความซับซ้อนและยุ่งยาก ดังนั้นทำให้สามารถมุ่งเน้นในการพัฒนาเฉพาะในส่วนที่เป็นแอปพลิเคชันได้อย่างเต็มที่

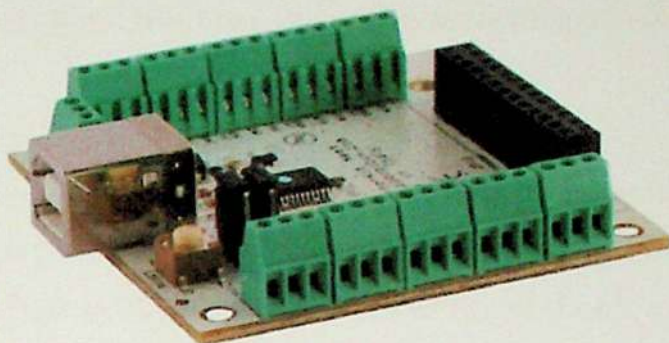
2.1.3.2 คุณสมบัติด้าน Hardware

ใช้ MCU ของบริษัท Microchip ตระกูล PIC24FJ ขนาด 16bit หน่วยความจำแบบ Flash ขนาด 256KB, RAM ขนาด 16KB มีความเร็วในการทำงานมากถึง 16Mips ที่ความถี่ 32MHz สามารถต่อแรงดันไฟเลี้ยง 5V หรือ 3.3V ให้กับบอร์ดได้ ระดับแรงดันไฟอ้างอิง ADC อยู่ที่ 2.048V มี Clock ภายนอก 32MHz มี Oscillator 32.768KHz สำหรับ RTCC ภายนอก มีพอร์ตต่อใช้งานและขยายได้มากถึง 18 Digital I/O, 4 Analog inputs (10bits ADC), 4 UARTs, SPI, I2C มีสายอากาศภายในแบบ Microstrip และยังสามารถปรับเปลี่ยนเพื่อต่อกับสายอากาศภายนอกด้วยคอนเนคเตอร์แบบ UFL ได้

2.1.3.3 คุณสมบัติด้าน Software

ใช้ภาษา C ในการพัฒนา ใช้ openPicus IDE ในการเขียนโปรแกรมและ config ค่าการทำงานต่างๆ และใช้ในการโปรแกรมคำสั่งให้กับโมดูล ทำให้สะดวกและง่ายต่อการพัฒนาเป็นอย่างมาก สนับสนุนการรับ-ส่ง ข้อมูลได้หลากหลายโปรโตคอล ทั้ง HTTP (Web server), TCP, UDP, FTP, SMTP การโปรแกรมคำสั่งให้กับ Flyport

ในการโปรแกรมคำสั่งการทำงานให้กับ Flyport จะต้องต่อกับอุปกรณ์ที่เป็น USB-to-Serial ซึ่งนักพัฒนาสามารถสร้างเองได้ แต่เพื่อความสะดวกแนะนำให้ใช้ USB Nest หรือ miniUSB Programmer ที่ทาง openPicus ได้เตรียมไว้ให้แล้ว โดยสามารถต่อสาย USB เข้ากับคอมพิวเตอร์เพื่อทำการโปรแกรมผ่าน IDE ให้กับโมดูลได้เลย



รูปที่ 2.7 ตัวอย่าง USB Nest



รูปที่ 2.8 ตัวอย่าง miniUSB Programmer

นักพัฒนาจำเป็นต้องติดตั้งเครื่องมือที่จำเป็น 2 ตัว เพื่อให้สามารถพัฒนาโปรแกรมบน Flyport ได้ซึ่งดาวน์โหลดได้จากเว็บไซต์ <http://www.openpicus.com/site/tools> ได้แก่

Microchip C30 compiler v3.24 เป็นตัว compiler ภาษา C สำหรับไมโครคอนโทรลเลอร์ของ Microchip

openPicus IDE 2.3 เป็น IDE สำหรับช่วยในการเขียนโปรแกรม ตั้งค่าการทำงาน และ โปรแกรมคำสั่งให้กับโมดูล Flyport

2.2 ทฤษฎีทางด้านซอฟต์แวร์

ในส่วนของซอฟต์แวร์จะแบ่งออกเป็น 2 ส่วน คือการใช้ภาษา C ในการเขียนโปรแกรมใส่ไมโครคอนโทรลเลอร์และใช้ภาษา Java ในการสร้างและออกแบบแอปพลิเคชัน ซึ่งมีรายละเอียดดังนี้

2.2.1 ภาษา C controller

ภาษาซีเป็นภาษาเขียนโปรแกรมระบบเชิงคำสั่ง (หรือเชิงกระบวนการ) ถูกออกแบบขึ้นเพื่อใช้แปลด้วยตัวแปลโปรแกรมแบบการเชื่อมโยงที่ตรงไปตรงมา สามารถเข้าถึงหน่วยความจำในระดับล่าง เพื่อสร้างภาษาที่จับคู่อย่างมีประสิทธิภาพกับชุดคำสั่งเครื่อง และแทบไม่ต้องการสนับสนุนใดๆ ขณะทำงาน ภาษาซีจึงเป็นประโยชน์สำหรับหลายโปรแกรมที่ก่อนหน้านี้เคยเขียนในภาษาแอสเซมบลีมาก่อนหากไม่คำนึงถึงความสามารถในระดับล่าง ภาษานี้ถูกออกแบบขึ้นเพื่อส่งเสริมการเขียนโปรแกรมที่ไม่ขึ้นอยู่กับเครื่องใดเครื่องหนึ่ง Machine-independent โปรแกรมภาษาซีที่เขียนขึ้นตามมาตรฐานและเคลื่อนย้ายได้ สามารถแปลได้บนแพลตฟอร์มคอมพิวเตอร์และระบบปฏิบัติการต่างๆ อย่างกว้างขวาง โดยแก้ไขรหัสต้นฉบับเพียงเล็กน้อยหรือไม่ต้องแก้ไขเลย ภาษานี้สามารถใช้ได้บนแพลตฟอร์มได้หลากหลายตั้งแต่ไมโครคอนโทรลเลอร์ฝังตัวไปจนถึงซูเปอร์คอมพิวเตอร์

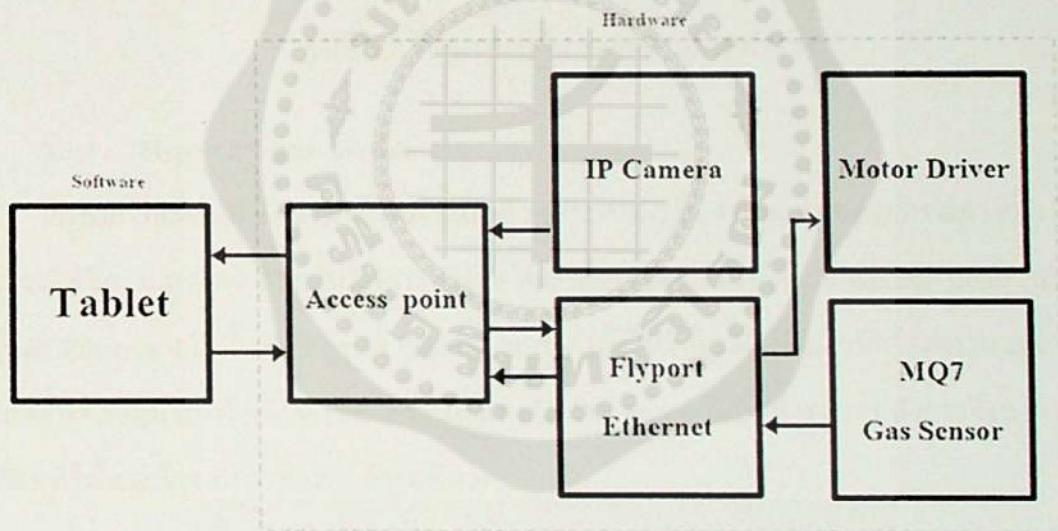
2.2.2 ภาษา Java programming

Java หรือ Java programming language คือภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่บริษัท ซัน ไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส C++ โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ “จาวา” ซึ่งเป็นชื่อกาแฟแทน จุดเด่นของภาษา Java อยู่ที่ผู้เขียนโปรแกรมสามารถใช้หลักการของ Object-Oriented Programming มาพัฒนาโปรแกรมของตนด้วย Java ได้ ภาษา Java เป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้นคลาสคือที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ (State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior)

บทที่ 3

การออกแบบและหลักการทำงาน

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ สำหรับควบคุมรดน้ำในพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย มีวัตถุประสงค์เพื่อพัฒนาแอปพลิเคชันที่ใช้ในการควบคุมรดน้ำให้เข้าถึงพื้นที่ที่ยากลำบากได้ ให้มีการแสดงสัญญาณภาพของกล้อง IP Camera มายังหน้าจอแสดงผลของแท็บเล็ต สามารถควบคุมทิศทางการหมุนของกล้อง IP Camera และสามารถตรวจจับก๊าซไวไฟเพื่อแจ้งเตือนมายังหน้าจอของแท็บเล็ตได้ โดยแอปพลิเคชันนี้สามารถแบ่งออกเป็น 2 ส่วนใหญ่ๆ ได้ คือ ส่วนของฮาร์ดแวร์ และส่วนของซอฟต์แวร์ ดังรูปที่ 3.1

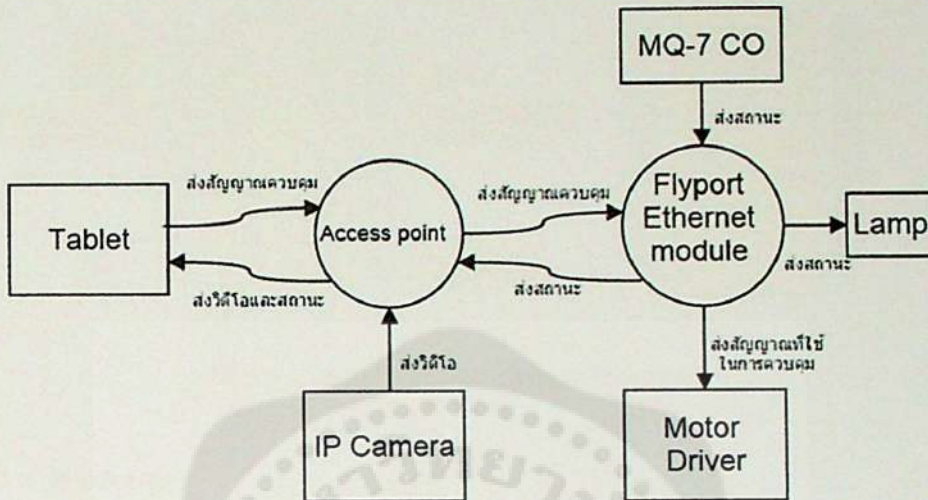


รูปที่ 3.1 แผนภาพ Block Diagram ของระบบ

3.1 หลักการทำงานด้านฮาร์ดแวร์

การทำงานของระบบที่ใช้ในการออกแบบและสร้างรดน้ำในพื้นที่เสี่ยงภัย คือ เมื่อทำการเปิดสวิตช์ของรดน้ำแล้ว วงจรควบคุมรดน้ำ, Access point, MQ7 และ IP Camera จะเริ่มทำงาน เมื่ออุปกรณ์ทุกอย่างทำงานพร้อมทุกตัวแล้ว Access point จะทำการกระจายสัญญาณออกไปเพื่อรอ

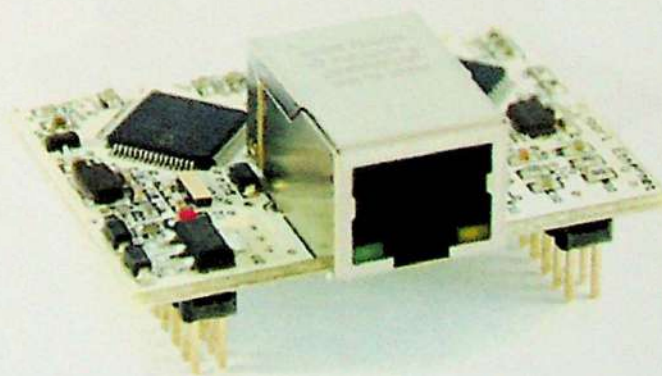
การเชื่อมต่อจากแท็บเล็ต ซึ่งผู้ควบคุมจะต้องทำการเชื่อมต่อสัญญาณกับ Access point เพื่อที่จะทำการควบคุมการทำงานของรถสำรวจ



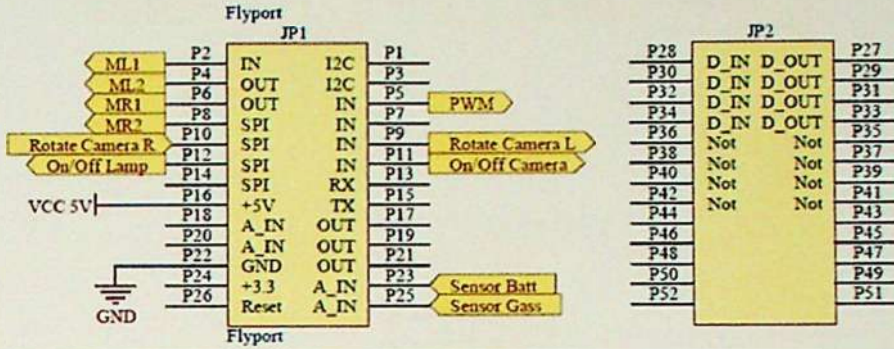
รูปที่ 3.2 ทิศทางการไหลของข้อมูลรถสำรวจ

3.1.1 Flyport Ethernet module

Flyport Ethernet module มีหน้าที่เป็นศูนย์กลางในการรับส่งข้อมูลของอุปกรณ์ต่างๆในรถสำรวจให้มีการแสดงผล และมีหลักการทำงานโดยแท็บเล็ตจะส่งคำสั่งผ่าน Access point มายัง Flyport Ethernet จากนั้นตัว Flyport Ethernet จะส่งคำสั่งไปให้อุปกรณ์ต่างๆมีการทำงาน เช่น ส่งคำสั่งการควบคุมไปให้ Motor Driver เพื่อสั่งการให้ DC Motor มีการทำงาน รถสำรวจจึงสามารถเคลื่อนที่ได้ และมีวงจรการทำงาน ดังรูปที่ 3.4



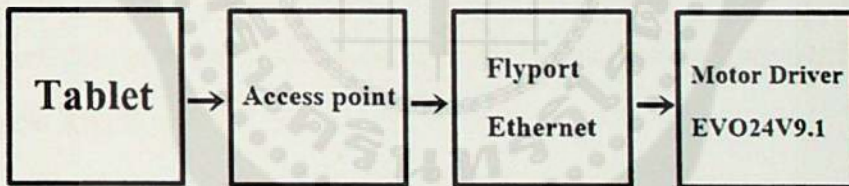
รูปที่ 3.3 Flyport Ethernet module



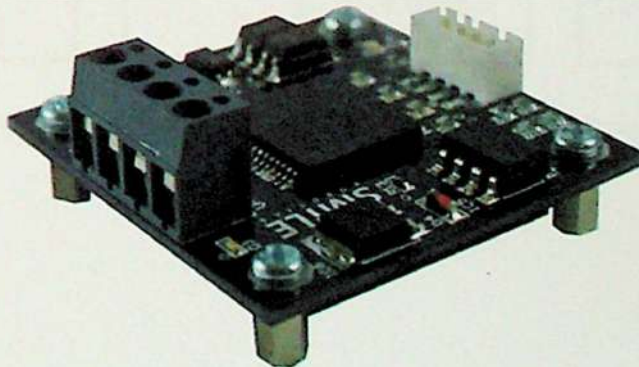
รูปที่ 3.4 วงจรการเชื่อมต่อ Flyport Ethernet

3.1.2 Motor Driver

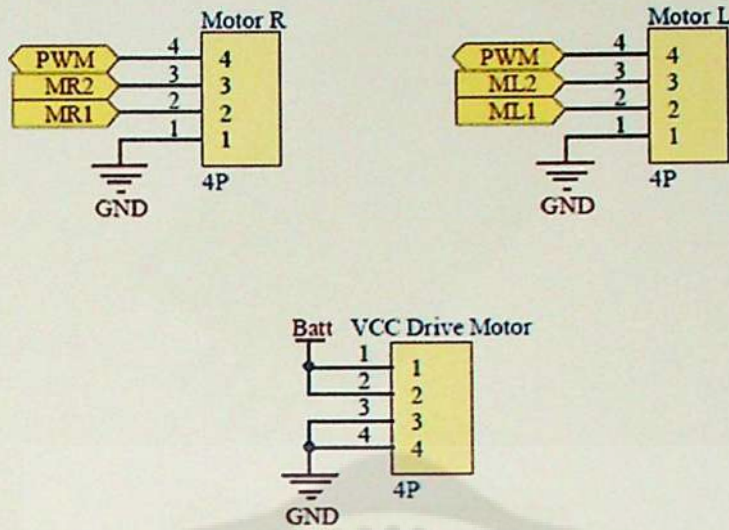
Motor Driver รุ่น EVO24V9.1 มีหน้าที่เป็นตัวควบคุมการทำงานของ DC Motor และมีหลักการทำงาน โดยรับคำสั่งมาจาก Flyport Ethernet เพื่อไปควบคุมบังคับให้ DC Motor มีการทำงาน จึงทำให้รถสำรวจสามารถเคลื่อนที่ได้ และมีวงจรการทำงาน ดังรูปที่ 3.7



รูปที่ 3.5 แผนภาพ Block Diagram ของ Motor Driver EVO24V9.1



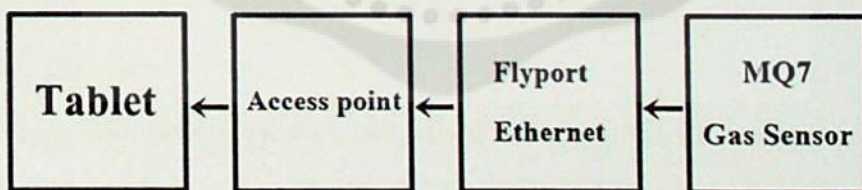
รูปที่ 3.6 Motor Driver EVO24V9.1



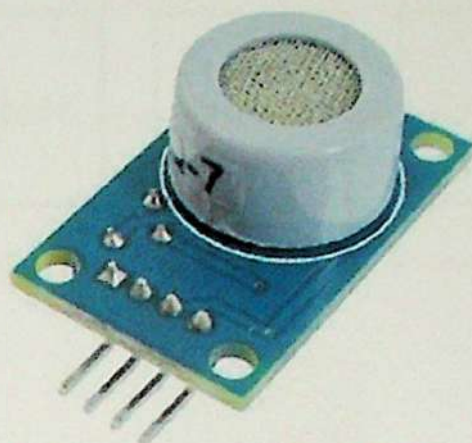
รูปที่ 3.7 วงจรการเชื่อมต่อ Drive Motor EVO24V9.1

3.1.3 MQ7 Gas Sensor Module

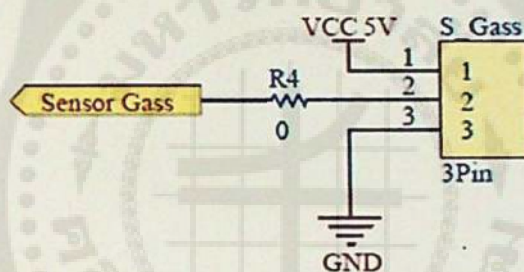
MQ7 Gas Sensor Module มีหน้าที่เป็นเซ็นเซอร์ที่ใช้ในการตรวจจับก๊าซไวไฟ และมีหลักการทำงาน คือเมื่อเซ็นเซอร์จับได้ว่าในบริเวณนั้นมีก๊าซไวไฟก็จะส่งค่าไปยัง Flyport Ethernet จากนั้น Flyport Ethernet จะส่งค่าไปยัง Access point เพื่อไปแสดงผลยังหน้าจอของแท็บเล็ต และมีวงจรการทำงาน ดังรูปที่ 3.10



รูปที่ 3.8 แผนภาพ Block Diagram ของ MQ7 Gas Sensor



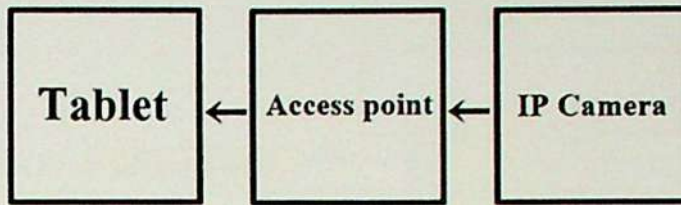
รูปที่ 3.9 MQ7 Gas Sensor Module



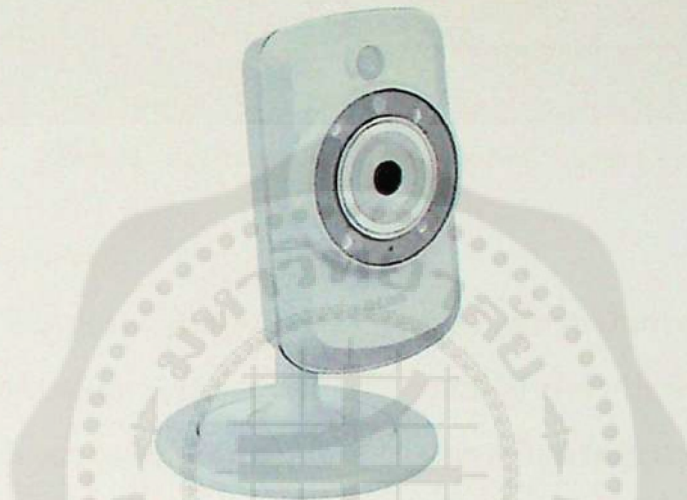
รูปที่ 3.10 วงจรการเชื่อมต่อ MQ7 Gas Sensor Module

3.1.4 IP Camera

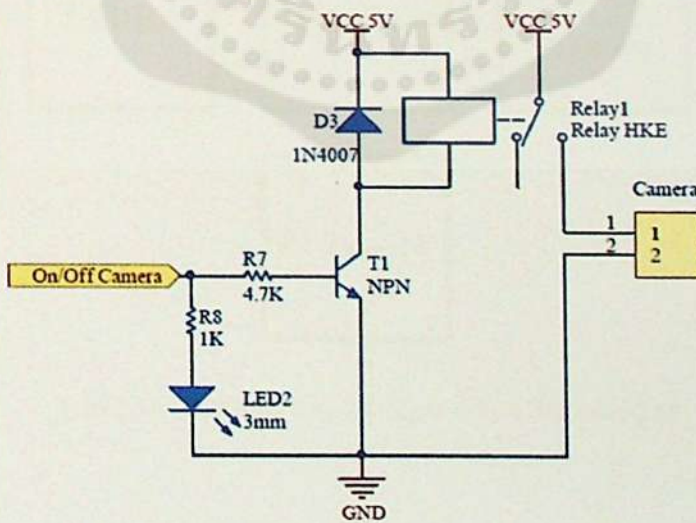
IP Camera ยี่ห้อ D-Link รุ่น DCS-942L มีหน้าที่ส่งสัญญาณภาพและเสียงไปยังแท็บเล็ต มีหลักการทำงาน คือกล้อง IP Camera จะเชื่อมต่อและส่งข้อมูลผ่าน Access point โดยตรง ซึ่งจะส่งสัญญาณภาพและเสียงผ่าน Access point เพื่อไปแสดงผลยังหน้าจอของแท็บเล็ต และสามารถควบคุมทิศทางการหมุนของกล้อง IP Camera ได้ โดยมีวงจรการทำงาน ดังรูปที่ 3.13 และรูปที่ 3.14



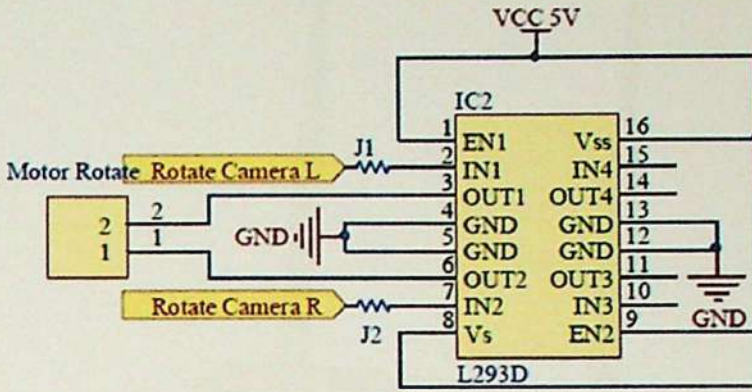
รูปที่ 3.11 แผนภาพ Block Diagram ของ IP Camera



รูปที่ 3.12 วงจรการเชื่อมต่อ MQ7 Gas Sensor Module



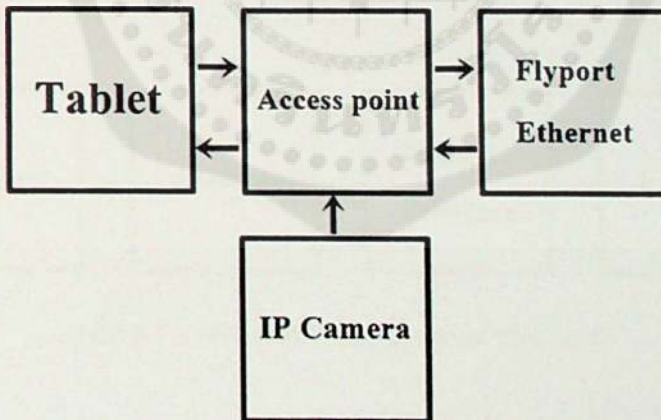
รูปที่ 3.13 วงจรควบคุมการเปิด-ปิดกล้อง IP Camera



รูปที่ 3.14 วงจรควบคุมการหมุนของกล้อง IP Camera

3.1.5 Access point

Access point ยี่ห้อ ZyXEL รุ่น WAP3205 v2 มีหน้าที่เป็นตัวกลางในการรับส่งข้อมูล และมีหลักการทำงาน คือตัว Access point จะกระจายสัญญาณออกไป เพื่อรอการเชื่อมต่อจากแท็บเล็ต เมื่อทำการเชื่อมต่อกันแล้ว Access point จะเป็นตัวกลางในการรับส่งข้อมูลระหว่างซอฟต์แวร์กับ ฮาร์ดแวร์ได้



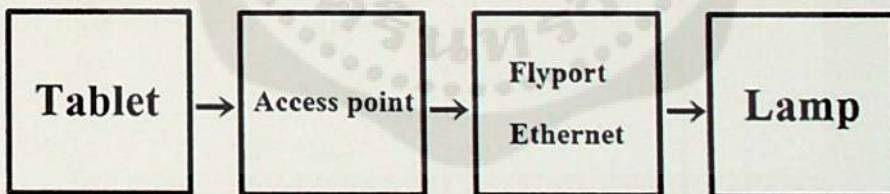
รูปที่ 3.15 แผนภาพ Block Diagram ของ Access point



รูปที่ 3.16 ZyXEL WAP3205 v2

3.1.6 Lamp

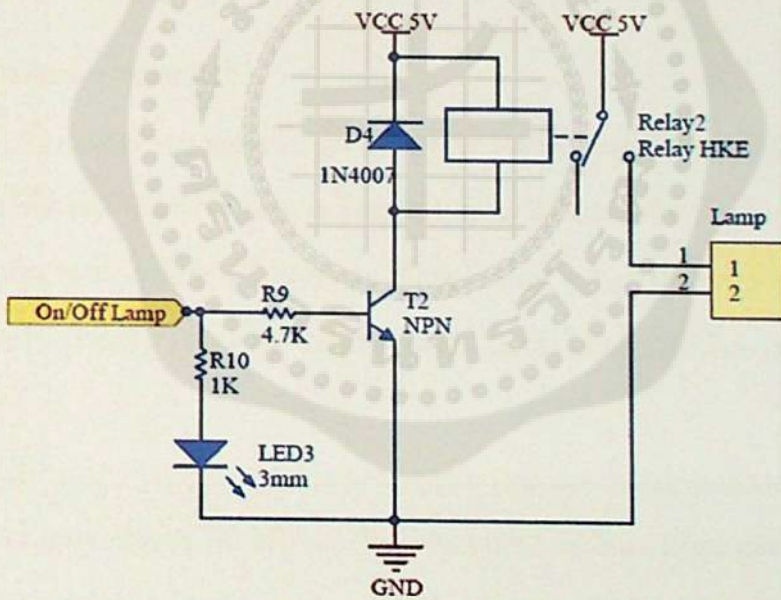
Lamp หรือหลอดไฟ LED มีหน้าที่ให้แสงสว่าง และมีหลักการทำงาน คือรับข้อมูลคำสั่งจาก Flyport Ethernet เพื่อควบคุมการเปิด-ปิด หลอดไฟที่ด้านหน้าของรถสำรวจ โดยมีวงจรการทำงาน ดังรูปที่ 3.14



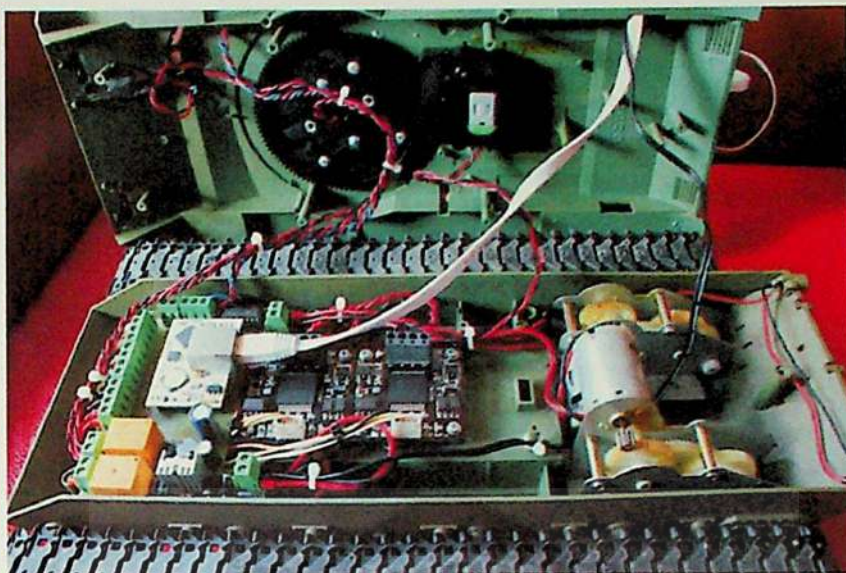
รูปที่ 3.17 แผนภาพ Block Diagram ของ Lamp



รูปที่ 3.18 หลอดไฟ LED



รูปที่ 3.19 วงจรควบคุมการเปิด-ปิดไฟที่ด้านหน้าของหุ่นยนต์



รูปที่ 3.20 การติดตั้งอุปกรณ์ต่างๆภายในรถสำรวจ

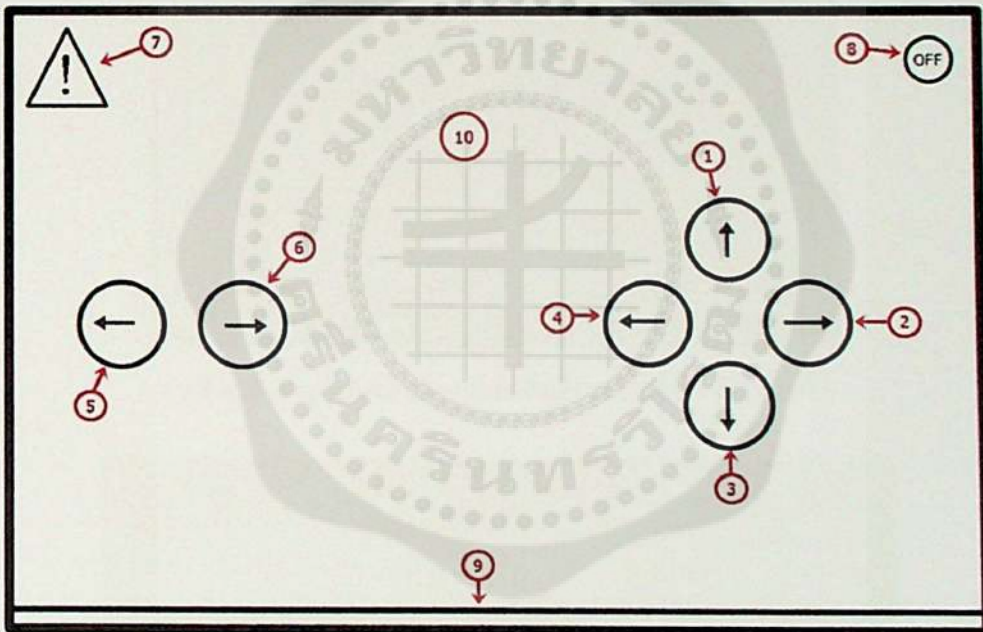
3.2 หลักการทำงานด้านซอฟต์แวร์

การออกแบบในส่วนของ Software นั้น จะต้องใช้ความรู้ความเข้าใจเกี่ยวกับระบบการทำงานของรถสำรวจให้ดีกว่าก่อน เพื่อให้ง่ายต่อการเขียน โปรแกรม ซึ่งในการเขียนโปรแกรมควบคุมการทำงานของรถสำรวจ จะใช้อุปกรณ์สื่อสารแบบไร้สาย Access point เป็นสื่อกลางในการส่งคำสั่งในส่วนควบคุมไปยังรถสำรวจและ IP Camera ได้นั้น จะต้องใช้เครื่องมือในการเขียนโปรแกรม ดังนี้

1) เขียนโปรแกรมควบคุมการทำงานของ Flyport Ethernet module และการสร้างแอปพลิเคชัน เพื่อใช้ในควบคุมรถสำรวจ โดยใช้ภาษา C++ และภาษา Java ในการเขียนชุดคำสั่งควบคุมการทำงานต่างๆ ของรถสำรวจ โดยมีการควบคุมการทำงานของรถสำรวจ เริ่มจากแอปพลิเคชันรับคำสั่งในการควบคุมการทำงานผ่านทางแท็บเล็ต จากนั้นระบบก็จะทำการส่งคำสั่งมายัง Access point เมื่อ Access point รับคำสั่งมาแล้วก็จะทำการประมวลผลเพื่อที่จะส่งคำสั่งให้กับ Flyport Ethernet module ทำงานซึ่งการทำงานของ Flyport Ethernet module จะเป็นการส่งคำสั่งให้กับ Motor Driver อีกทีเพื่อไปควบคุม DC Motor แต่ละตัวให้ทำการเคลื่อน ไหวไปในทิศทางที่ต้องการ

2) การออกแบบแอปพลิเคชัน เป็นการออกแบบการแสดงผลการเชื่อมต่อของรถสำรวจ และ ผู้ใช้งาน คือเป็นการออกแบบแอปพลิเคชัน ที่ใช้สำหรับควบคุมรถสำรวจ โดยการออกแบบแอป

พลิเคชัน ของรถตำรวจจะใช้ภาษา Java ในการออกแบบและสร้างแอปพลิเคชัน รวมไปถึงเครื่องมือ Plug-in ต่างๆ ในโปรแกรม Android Studio ที่จะเข้ามาช่วยให้การทำงานดีขึ้น โดยในการออกแบบ ตัวแสดงผลบนหน้าจอแท็บเล็ตนั้นจะประกอบไปด้วย การแสดงสัญญาณภาพจากกล้อง IP Camera มาบนหน้าจอของแท็บเล็ต การแสดงสัญลักษณ์แจ้งเตือนก๊าซไวไฟบนหน้าจอของแท็บเล็ต การแสดง ระดับพลังงานของแบตเตอรี่ที่ติดตั้งอยู่ในรถบังคับ ปุ่มควบคุมสำหรับการบังคับทิศทาง การเคลื่อนที่ของรถตำรวจ ปุ่มควบคุมการบังคับทิศทางการหมุนของ IP Camera ที่ติดตั้งอยู่บนรถ บังคับ และมีปุ่มควบคุมการเปิด-ปิด หลอดไฟในส่วนด้านหน้าของรถตำรวจ ดังแสดงในรูปที่ 3.21



รูปที่ 3.21 การออกแบบหน้าจอการควบคุมหุ่นยนต์

หมายเลข 1 คือ ปุ่มควบคุมบังคับทิศทางของรถตำรวจให้เคลื่อนที่ไปข้างหน้า
 หมายเลข 2 คือ ปุ่มควบคุมบังคับทิศทางของรถตำรวจให้หมุนตัวไปทางขวา
 หมายเลข 3 คือ ปุ่มควบคุมบังคับทิศทางของรถตำรวจให้เคลื่อนที่ถอยหลัง
 หมายเลข 4 คือ ปุ่มควบคุมบังคับทิศทางของรถตำรวจให้หมุนตัวไปทางซ้าย
 หมายเลข 5 คือ ปุ่มควบคุมบังคับกล้อง IP Camera ให้หมุนกล้องไปทางซ้าย
 หมายเลข 6 คือ ปุ่มควบคุมบังคับกล้อง IP Camera ให้หมุนกล้องไปทางขวา

หมายเลข 7 คือ สัญลักษณ์การแจ้งเตือนเมื่อรถสำรวจสามารถจับแก๊สไวไฟได้ ดังรูปที่ 4.22

หมายเลข 8 คือ ปุ่มควบคุมการเปิด-ปิด หลอดไฟที่อยู่ด้านหน้าของรถสำรวจ

หมายเลข 9 คือ แถบแสดงสถานะพลังงานของแบตเตอรี่

หมายเลข 10 คือ หน้าจอการแสดงผลสัญญาณภาพที่ส่งมาจากกล้อง IP Camera แสดงผลเต็มหน้าจอของแท็บเล็ต ดังรูปที่ 4.23 และรูปที่ 4.24



รูปที่ 4.22 สัญลักษณ์แสดงการแจ้งเตือนเมื่อจับแก๊สไวไฟได้



รูปที่ 4.23 ภาพจากกล้อง IP Camera ในที่ที่มีแสงสว่างมากหรือเวลากลางวัน



รูปที่ 4.24 ภาพจากกล้อง IP Camera ในที่ที่มีแสงน้อยหรือเวลากลางคืน

บทที่ 4

การทดลองและผลการทดลองโครงการ

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ สำหรับควบคุมรถสำรวจพื้นที่เสี่ยง ภัยผ่านเครือข่ายไร้สาย มีวัตถุประสงค์เพื่อพัฒนาแอปพลิเคชันที่ใช้ในการควบคุมรถสำรวจให้ เข้าถึงพื้นที่ที่ยากลำบากได้ ให้มีการแสดงสัญญาณภาพของกล้อง IP Camera มายังหน้าจอแสดงผล ของแท็บเล็ต สามารถควบคุมทิศทางการหมุนของกล้อง IP Camera และสามารถตรวจจับก๊าซไวไฟ เพื่อแจ้งเตือนมายังหน้าจอของแท็บเล็ตได้ โดยการทดลองจะแบ่งออกเป็น 5 ส่วน คือ การเคลื่อนที่ ของรถสำรวจในพื้นที่ทางตรง การเคลื่อนที่ของรถสำรวจในแผนที่จำลอง ระยะทางในการควบคุม การทำงานของรถสำรวจ การควบคุมบังคับทิศทางการหมุนของกล้อง IP Camera และการตรวจจับ ก๊าซไวไฟ MQ7 Gas Sensor Module

4.1 การเคลื่อนที่ของรถสำรวจในพื้นที่ทางตรง

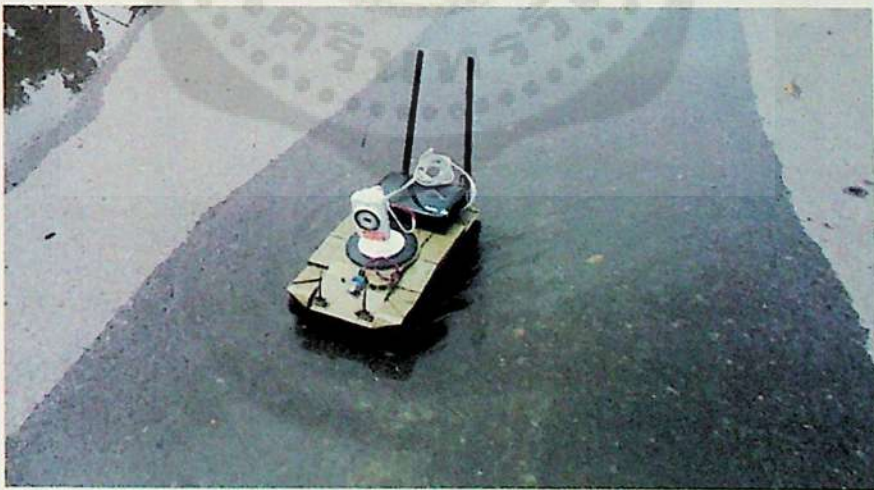
การทดสอบในขั้นตอนนี้เป็นการทดสอบการเคลื่อนที่ในทางตรงของรถสำรวจ มี วัตถุประสงค์เพื่อทดสอบประสิทธิภาพในการเคลื่อนที่ของรถสำรวจและประสิทธิภาพในการ ทำงานของแอปพลิเคชัน โดยมีระยะทางในการทดสอบ 5 เมตร สภาพพื้นผิวที่ใช้ในการทดสอบมี 4 แบบ และเก็บค่าโดยการจับเวลา ซึ่งผลลัพธ์ที่ได้จะนำมาเปรียบเทียบกับค่าทฤษฎีเพื่อคำนวณหาค่า ความคลาดเคลื่อน โดยค่าทฤษฎีที่นำมาเปรียบเทียบหาได้จากสูตร $v = \frac{s}{t}$ เมื่อรถสำรวจมีความเร็ว เท่ากับ 0.33 เมตรต่อวินาที ดังนั้นเวลาที่รถสำรวจเคลื่อนที่ได้ เท่ากับ 15.15 วินาที และหา เปอร์เซ็นต์ความคลาดเคลื่อนได้จากสูตร

$$\text{เปอร์เซ็นต์ความคลาดเคลื่อน (\% Error)} = \frac{(\text{ค่าที่ทดลองได้}-\text{ค่าที่แท้จริง})}{\text{ค่าที่แท้จริง}} \times 100$$

ตารางที่ 4.1 ผลการทดสอบการเคลื่อนที่ของรถสำรวจในทางตรง

| บริเวณการเคลื่อนที่ ระยะทาง 5 เมตร | ผลการทดสอบ | | | เปอร์เซ็นต์ ความคลาด เคลื่อนเฉลี่ย |
|---------------------------------------|--------------|--------------|--------------|--|
| | 1 | 2 | 3 | |
| พื้นที่มีน้ำขัง | 15.17 วินาที | 15.30 วินาที | 15.25 วินาที | 0.59 % |
| พื้นที่ทราย | 15.63 วินาที | 15.67 วินาที | 15.87 วินาที | 3.78 % |
| พื้นต่างระดับ | 15.81 วินาที | 16.01 วินาที | 15.93 วินาที | 5.05 % |
| ก้อนหิน | 16.31 วินาที | 16.51 วินาที | 16.54 วินาที | 8.59 % |

จากตารางที่ 4.1 จากผลการทดลองบนพื้นที่มีน้ำขังมีเปอร์เซ็นต์ความคลาดเคลื่อน 0.59% บนพื้นที่ทรายมีเปอร์เซ็นต์ความคลาดเคลื่อน 3.78% บนพื้นต่างระดับมีเปอร์เซ็นต์ความคลาดเคลื่อน 5.05% และบนพื้นที่มีก้อนหินมีเปอร์เซ็นต์ความคลาดเคลื่อน 8.59% จะเห็นว่าในแต่ละพื้นผิวมีความคลาดเคลื่อนต่างกันไป สาเหตุมาจากปัจจัยหลายอย่าง เช่น สภาพของพื้นผิวที่ไม่ราบเรียบ เกิดการหน่วงเวลาของการส่งข้อมูล และระดับพลังงานที่ลดลงของแบตเตอรี่ในรถสำรวจ เป็นต้น



รูปที่ 4.1 การเคลื่อนที่ในทางตรงบนพื้นที่มีน้ำขัง



รูปที่ 4.2 การเคลื่อนที่ในทางตรงบนพื้นทราย



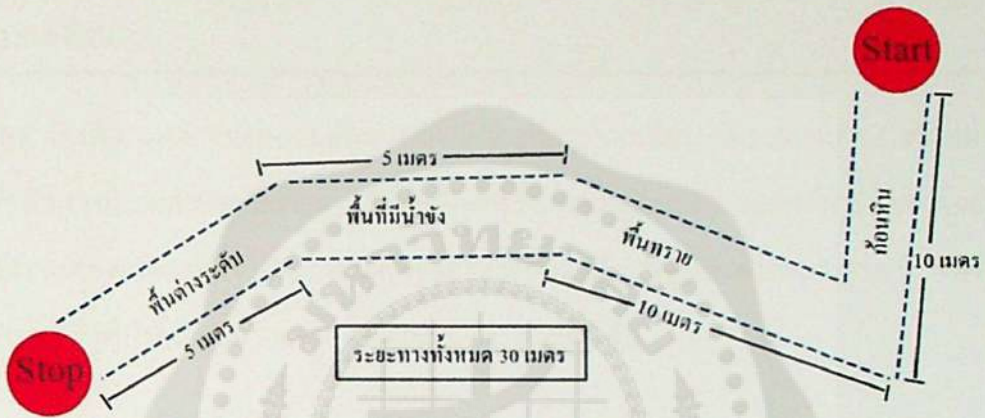
รูปที่ 4.3 การเคลื่อนที่ในทางตรงบนพื้นที่มีก้อนหิน

4.2 การเคลื่อนที่ของรถสำรวจในแผนที่จำลอง

การทดสอบในขั้นตอนนี้เป็นการทดสอบการเคลื่อนที่ของรถสำรวจในแผนที่ที่จำลองขึ้นมา โดยมีวัตถุประสงค์เพื่อทดสอบว่า เมื่อรถสำรวจนำไปใช้งานจริงในพื้นที่ที่ขากลำบากจะมีประสิทธิภาพในการเคลื่อนที่ได้ดีมาน้อยแค่ไหน เมื่อเปรียบเทียบกับค่าทฤษฎี โดยมีระยะทางทั้งหมด 30 เมตร มีสภาพพื้นผิวทั้งหมด 4 แบบ ดังรูปที่ 4.4 ขั้นตอนที่ทดสอบ คือควบคุมบังคับรถสำรวจไปตามเส้นทางที่จำลองขึ้นมา และเก็บค่าโดยการจับเวลา ซึ่งผลลัพธ์ที่ได้จะนำมาเปรียบเทียบกับค่าทฤษฎีเพื่อคำนวณหาค่าความคลาดเคลื่อน โดยค่าทฤษฎีที่นำมาเปรียบเทียบหาได้

จากสูตร $v = \frac{S}{t}$ เมื่อรถสำรวจมีความเร็วเท่ากับ 0.33 เมตรต่อวินาที ดังนั้นเวลาที่รถสำรวจเคลื่อนที่ได้ เท่ากับ 90.9 วินาที และหาเปอร์เซ็นต์ความคลาดเคลื่อนได้จากสูตร

$$\text{เปอร์เซ็นต์ความคลาดเคลื่อน (\% Error)} = \frac{(\text{ค่าที่ทดลองได้} - \text{ค่าที่แท้จริง})}{\text{ค่าที่แท้จริง}} \times 100$$



รูปที่ 4.4 แผนที่จำลองที่ใช้ในการทดสอบรถสำรวจ



รูปที่ 4.5 แผนที่จากสถานที่จริงที่ใช้ในการทดสอบรถสำรวจ

ตารางที่ 4.2 ผลการทดสอบการเคลื่อนที่ของรถสำรวจในแผนที่จำลอง

| บริเวณการเคลื่อนที่ | ผลการทดสอบ | | | ค่าเฉลี่ย |
|--------------------------------|---------------|---------------|---------------|---------------|
| | 1 | 2 | 3 | |
| ระยะทาง 30 เมตร | | | | |
| เวลา | 100.57 วินาที | 103.44 วินาที | 107.44 วินาที | 103.62 วินาที |
| เปอร์เซ็นต์ ความคลาดเคลื่อน | 10.63 % | 13.79 % | 18.19 % | 14.20 % |

จากตารางที่ 4.2 ผลการทดสอบเกิดเปอร์เซ็นต์ความคลาดเคลื่อน 14.20% เนื่องจากสภาพของพื้นผิวที่มีความขากลำบาก มีการหน่วงเวลาในการรับส่งข้อมูลระหว่างซอฟต์แวร์กับฮาร์ดแวร์ ระดับพลังงานของแบตเตอรี่ที่ลดลง และความสามารถของผู้ควบคุมบังคับรถสำรวจ ด้วยปัจจัยต่างๆที่กล่าวมาจึงทำให้เกิดค่าความคลาดเคลื่อนในการทดสอบขึ้น

4.3 ระยะทางในการควบคุมการทำงานของรถสำรวจ

การทดสอบระยะทางในการควบคุมระหว่างแท็บเล็ตกับรถสำรวจนี้ โดยมีวัตถุประสงค์เพื่อหาระยะทางที่แท็บเล็ตสามารถควบคุมรถสำรวจได้ โดยทำการทดสอบ 2 ครั้ง และสถานที่ในการทดสอบจะแตกต่างกัน ดังตารางที่ 4.3

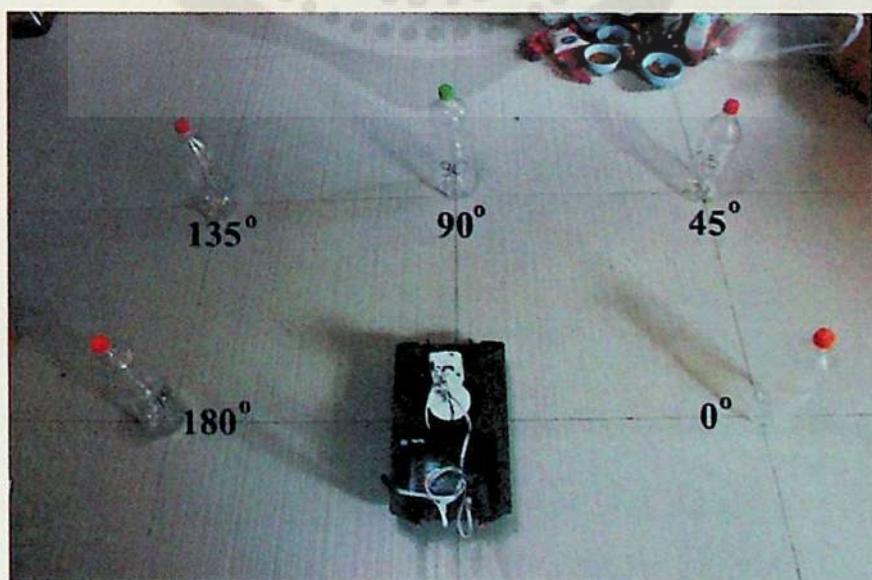
ตารางที่ 4.3 ผลการทดสอบระยะทางในการควบคุมการทำงานของรถสำรวจ

| ระยะทาง (เมตร) | ผลการทดสอบ ครั้งที่ 1 | ผลการทดสอบ ครั้งที่ 2 |
|----------------|-----------------------|-----------------------|
| 10 เมตร | ได้ | ได้ |
| 20 เมตร | ได้ | ได้ |
| 30 เมตร | ไม่ได้ | ได้ |
| 40 เมตร | ไม่ได้ | ได้ |
| 50 เมตร | ไม่ได้ | ได้ |

จากตารางที่ 4.3 เป็นการทดสอบการควบคุมการทำงานของรถสำรวจ ครั้งที่ 1 ในบริเวณที่ทำการทดลองเป็นเขตชุมชนอยู่ในหมู่บ้านเมืองเอก มีหอพักนักศึกษาอยู่หลายแห่ง ซึ่งมีตัวกระจายสัญญาณ Access point อยู่เป็นจำนวนมากและทดสอบการควบคุมในบริเวณภายในอาคารจึงทำให้ผลของการควบคุมการทำงานของรถสำรวจเกิดการรบกวนกันของสัญญาณทำให้การส่งสัญญาณนั้นขาดหายเป็นบางช่วงเนื่องจากเกิดการ Reset ของการเชื่อมต่อสัญญาณ เมื่อสัญญาณขาดหายแล้วยังมีการค้างสถานะของการควบคุมการทำงานของรถสำรวจ ส่งผลให้ระยะทางที่สามารถควบคุมการทำงาน ทำได้ใกล้ การทดสอบ ครั้งที่ 2 ได้มีการเปลี่ยนสถานที่ทดสอบเป็นลานจอดรถยนต์ของโรงเรียนเซนต์โยเซฟเมืองเอก ซึ่งมีบริเวณที่โล่งกว้างและมีตัวกระจายสัญญาณ Access point อยู่น้อย จึงส่งผลให้ระยะการรับส่งสัญญาณ ได้ไกลขึ้น

4.4 การควบคุมบังคับทิศทางการหมุนของกล้อง IP Camera

การทดสอบในขั้นตอนนี้เป็นการทดสอบการควบคุมทิศทางการหมุนของกล้อง IP Camera โดยมีวัตถุประสงค์เพื่อตรวจสอบว่ากล้อง IP Camera สามารถบังคับทิศทางการให้หมุนได้ 180 องศา และแสดงสัญญาณภาพออกมาได้ตรงตามที่กำหนด โดยให้รถสำรวจอยู่ตรงจุดศูนย์กลางและวางขวดน้ำไว้ในตำแหน่งแต่ละองศาที่กำหนด ดังรูปที่ 4.5

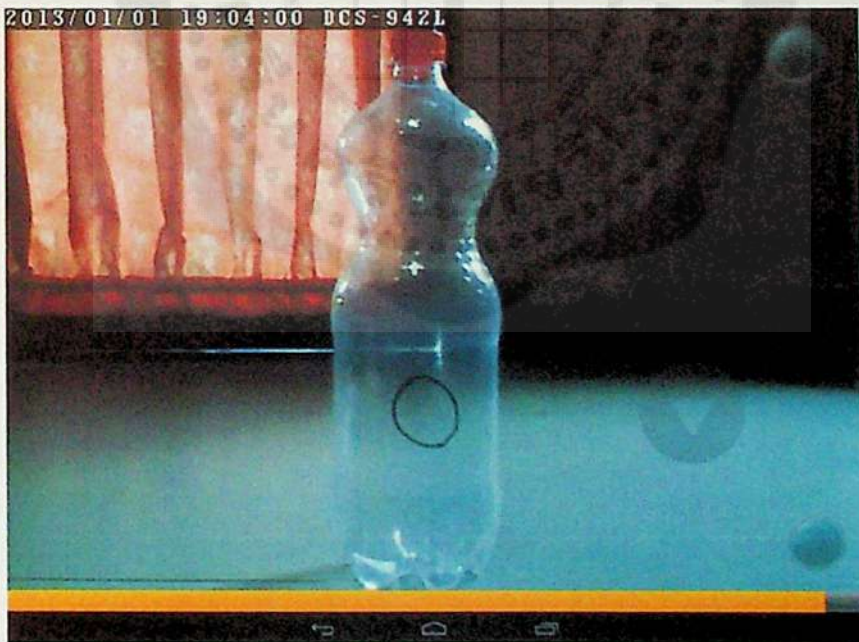


รูปที่ 4.6 การทดสอบการควบคุมการหมุนของกล้อง IP Camera

ตารางที่ 4.4 ผลการทดสอบการควบคุมทิศทางการหมุนของกล้อง IP Camera

| องศา | ผลการทดลอง | | | | | | | | | | เปอร์เซ็นต์การควบคุมและมองเห็น |
|----------|------------|---|---|---|---|---|---|---|---|---|--------------------------------|
| | 1 | | 2 | | 3 | | 4 | | 5 | | |
| | Y | N | Y | N | Y | N | Y | N | Y | N | |
| 0 องศา | / | | / | | / | | / | | / | | 100 % |
| 45 องศา | / | | / | | / | | / | | / | | 100 % |
| 90 องศา | / | | / | | / | | / | | / | | 100 % |
| 135 องศา | / | | / | | / | | / | | / | | 100 % |
| 180 องศา | / | | / | | / | | / | | / | | 100 % |

จากตารางที่ 4.4 ผลการทดสอบกล้อง IP Camera สามารถควบคุมและบังคับทิศทางได้แม่นยำ 100% ในทุกองศาที่ได้ทำการทดลอง



รูปที่ 4.7 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 0 องศา



รูปที่ 4.8 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 45 องศา



รูปที่ 4.9 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 90 องศา



รูปที่ 4.10 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 135 องศา



รูปที่ 4.11 การทดสอบการควบคุมการหมุนของกล้อง IP Camera ที่ตำแหน่ง 180 องศา

4.5 การตรวจจับก๊าซไวไฟ MQ7 Gas Sensor Module

การทดสอบในขั้นตอนนี้เป็นการทดสอบการแสดงผลและการตรวจจับก๊าซไวไฟ โดยมีวัตถุประสงค์เพื่อทดสอบว่า MQ7 Gas Sensor สามารถตรวจจับวัตถุที่เป็นก๊าซไวไฟได้และสามารถส่งค่ามาแสดงผลยังหน้าจอของแท็บเล็ตได้ โดยใช้อุปกรณ์ในการทดสอบคือ ไฟแช็ค แอลกอฮอล์ ขาฉีดกำจัดขยะ ควันทูบไฟ ละอองน้ำ และทราย ดังตารางที่ 4.5

ตารางที่ 4.5 ผลการทดสอบการตรวจจับก๊าซไวไฟ

| วัตถุที่ทดลอง | ผลการทดลอง | | | | | | | | | | เปอร์เซ็นต์การตรวจจับ |
|---------------|------------|---|---|---|---|---|---|---|---|---|-----------------------|
| | 1 | | 2 | | 3 | | 4 | | 5 | | |
| | Y | N | Y | N | Y | N | Y | N | Y | N | |
| ไฟแช็ค | / | | / | | / | | / | | / | | 100 % |
| แอลกอฮอล์ | / | | / | | / | | / | | / | | 100 % |
| ขาฉีดกำจัดขยะ | / | | / | | / | | / | | / | | 100 % |
| ควันทูบไฟ | | / | | / | | / | | / | | / | 0 % |
| ละอองน้ำ | | / | | / | | / | | / | | / | 0 % |
| ทราย | | / | | / | | / | | / | | / | 0 % |

จากตารางที่ 4.5 การตรวจจับก๊าซไวไฟของ MQ7 Gas Sensor สามารถทำงานได้อย่างมีประสิทธิภาพ ซึ่งแสดงผลการตรวจจับวัตถุที่ติดไฟได้อย่าง ไฟแช็ค แอลกอฮอล์ และขาฉีดกำจัดขยะ ได้ครบ 100% และสามารถแสดงผลการแจ้งเตือนมายังหน้าจอของแท็บเล็ตได้เป็นอย่างดี



รูปที่ 4.12 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้ไฟแช็ค



รูปที่ 4.13 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้แอลกอฮอล์



รูปที่ 4.14 ทดสอบ MQ-7 การตรวจจับก๊าซไวไฟ โดยใช้แก๊สกระป๋องและยานีคกำจัดขุง



บทที่ 5

สรุปและข้อเสนอแนะ

5.1 สรุปผลของโครงการ

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย มีวัตถุประสงค์เพื่อพัฒนาแอปพลิเคชันที่ใช้ในการควบคุมรถสำรวจให้เข้าถึงพื้นที่ที่ยากลำบากได้ ให้มีการแสดงสัญญาณภาพของกล้อง IP Camera มายังหน้าจอแสดงผลของแท็บเล็ต สามารถควบคุมทิศทางการหมุนของกล้อง IP Camera และสามารถตรวจจับก๊าซไวไฟเพื่อแจ้งเตือนมายังหน้าจอของแท็บเล็ตได้ โดยการทดลองการเคลื่อนที่ของรถสำรวจในพื้นที่ทางตรง เพื่อทดสอบประสิทธิภาพในการเคลื่อนที่ของรถสำรวจและประสิทธิภาพในการทำงานของแอปพลิเคชัน จากผลการทดลองบนพื้นที่ที่มีน้ำขังมีเปอร์เซ็นต์ความคลาดเคลื่อน 0.59% บนพื้นที่ทรายมีเปอร์เซ็นต์ความคลาดเคลื่อน 3.78% บนพื้นดินระดับมีเปอร์เซ็นต์ความคลาดเคลื่อน 5.05% และบนพื้นที่มีก้อนหินมีเปอร์เซ็นต์ความคลาดเคลื่อน 8.59% จะเห็นว่าในแต่ละพื้นผิวมีค่าความคลาดเคลื่อนต่างกันไป สาเหตุมาจากปัจจัยหลายอย่าง เช่น สภาพของพื้นผิวที่ไม่ราบเรียบ เกิดการหน่วงเวลาของการส่งข้อมูล และระดับพลังงานที่ลดลงของแบตเตอรี่ในรถสำรวจ เป็นต้น การทดลองการเคลื่อนที่ของรถสำรวจในแผนที่จำลอง เพื่อทดสอบว่า เมื่อนำรถสำรวจไปใช้งานจริงในพื้นที่ที่ยากลำบาก รถสำรวจจะมีประสิทธิภาพในการเคลื่อนที่ได้ดีขนาดไหนเมื่อเปรียบเทียบกับค่าทฤษฎี ผลการทดสอบเกิดเปอร์เซ็นต์ความคลาดเคลื่อน 14.20% เนื่องจากสภาพของพื้นผิวที่มีความยากลำบาก มีการหน่วงเวลาในการรับส่งข้อมูลระหว่างซอฟต์แวร์กับฮาร์ดแวร์ ระดับพลังงานของแบตเตอรี่ที่ลดลง และความสามารถของผู้ควบคุมบังคับรถสำรวจ ด้วยปัจจัยต่างๆ ที่กล่าวมาจึงทำให้เกิดค่าความคลาดเคลื่อนในการทดสอบขึ้น การทดลองการควบคุมบังคับทิศทางการหมุนของกล้อง IP Camera เพื่อตรวจสอบว่ากล้อง IP Camera สามารถบังคับทิศทางให้หมุนได้ 180 องศา และแสดงสัญญาณภาพออกมาได้ตรงตามที่ต้องการ จากผลการทดลองกล้อง IP Camera สามารถควบคุมและบังคับทิศทางได้แม่นยำ 100% ในทุกองศาที่ได้ทำการทดลอง และการทดลองการตรวจจับก๊าซไวไฟ MQ7 Gas Sensor เพื่อทดสอบว่า MQ7 Gas Sensor สามารถตรวจจับวัตถุที่เป็น

ก๊าซไฟไวไฟได้และสามารถส่งค่ามาแสดงผลยังหน้าจอของแท็บเล็ตได้ จากผลการทดลองการตรวจจับก๊าซไวไฟของ MQ7 Gas Sensor สามารถทำงานได้อย่างมีประสิทธิภาพ ซึ่งแสดงผลการตรวจจับวัตถุที่ติดไฟได้อย่าง ไฟแช็ค แอลกอฮอล์ และยาฉีดกำจัดยุง ได้ครบ 100% และสามารถแสดงผลการแจ้งเตือนมายังหน้าจอของแท็บเล็ตได้เป็นอย่างดี

5.2 ปัญหาและข้อเสนอแนะ

ในการทดลองโครงการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ สำหรับควบคุมรถสำรวจพื้นที่เสี่ยงภัยผ่านเครือข่ายไร้สาย มีการเกิดปัญหาในการทดลอง คือสัญญาณภาพและเสียงที่ส่งมาจากกล้อง IP Camera มีการหน่วงเวลา ทำให้มีผลต่อการทดลอง คือเมื่อทำการควบคุมรถสำรวจด้วยการมองจากหน้าจอของแท็บเล็ต ทำให้ไม่สามารถควบคุมได้ตามที่ต้องการ เพราะสัญญาณภาพที่ส่งมามีการหน่วงเวลา หน้าจอของแท็บเล็ตกับตำแหน่งของรถสำรวจจึงไม่ตรงกับความเป็นจริง ซึ่งทำให้การควบคุมเป็นไปได้ยากและรถสำรวจอาจจะไปชนเข้ากับสิ่งกีดขวางได้ การนำรถสำรวจพื้นที่เสี่ยงภัยไปพัฒนาต่อ สามารถพัฒนาให้รถสำรวจสามารถเคลื่อนที่ตามวัตถุหรือหลบหลีกวัตถุได้ โดยใช้โซนาร์เซ็นเซอร์ เพื่อให้รถสำรวจมีประสิทธิภาพในการเคลื่อนที่ที่ดีมากขึ้นสามารถหลบหลีกสิ่งของและวัตถุต่างๆที่ขวางทางได้ พัฒนาเรื่องความเร็วในการเคลื่อนที่ ให้มีหลายระดับและสามารถปรับเลือกได้บนสถานีควบคุม เพื่อให้รถสำรวจใช้งานได้ในหลายสภาพพื้นผิวและสามารถเข้าถึงพื้นที่ที่ยากลำบากได้ โดยที่ความเร็วของรถสำรวจเพิ่มมากขึ้นจึงทำให้ลดระยะเวลาในการเข้าถึงพื้นที่ต่างๆได้ดีขึ้น การนำ GPS เข้ามาประยุกต์ใช้เพื่อช่วยในการบอกตำแหน่งของตัวรถสำรวจที่ถูกต้อง สำหรับการเคลื่อนที่ภายนอกอาคาร

เอกสารอ้างอิง

จักรพงษ์ กาวงศ์. (2548). “การควบคุมหุ่นยนต์สำรวจผ่านเว็บเบราว์เซอร์.” ปรินซ์นิพนธ์ปริญา
ตรี. มหาวิทยาลัยเชียงใหม่.

พร้อมเลิศ หล่อวิจิตร. (2557). “คู่มือเขียนแอป Android ฉบับรวมโค้ด.” โปรวิชั่น. กรุงเทพฯ.

จักรชัย โสอินทร์และคณะ. (2555). “Android App Development.” ไอดีซี พรีเมียร์. นนทบุรี.

ชัชวดี พรหมบุตร. “การติดตั้ง JDK และ กำหนด path ให้ Support android Studio” [Online]

Available : <http://androidthai.in.th/conternt-android/238-set-up-and-install-jdk-for-android-studio.html>

ชัชวดี พรหมบุตร. “การติดตั้ง android studio พร้อม HelloWorld” [Online]

Available : <http://androidthai.in.th/conternt-android/246-install-android-studio-and-helloworld.html>

Elec-Creations. “รู้จักกับ Flyport” [Online]

Available : http://www.elec-creations.com/index.php?option=com_content&view=article&id=35:flyport-product&catid=25&Itemid=131

Elec-Creations. “คู่มือการใช้งานเบื้องต้น Flyport WiFi & Ethernet” [PDF FILE]

Available : http://www.elec-creations.com/images/file/Flyport_WiFi_Eth_Manual.pdf



ภาคผนวก

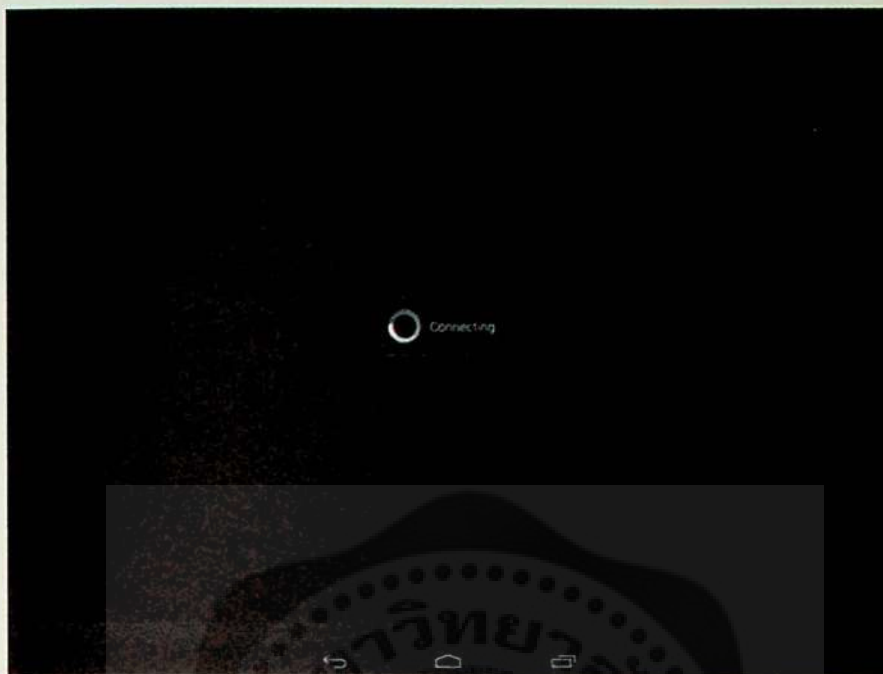
ภาคผนวก ก. วิธีใช้งาน แอปพลิเคชัน

1. เมื่อต้องการใช้งานแอปพลิเคชัน ต้องทำการเปิดการทำงานของหุ่นยนต์ก่อน เพื่อรอรับก่อนเชื่อมจากอุปกรณ์ที่ติดตั้งแอปพลิเคชันไว้ จากนั้นให้ Click ที่รูป icon RobotSWU เพื่อเปิดแอปพลิเคชัน

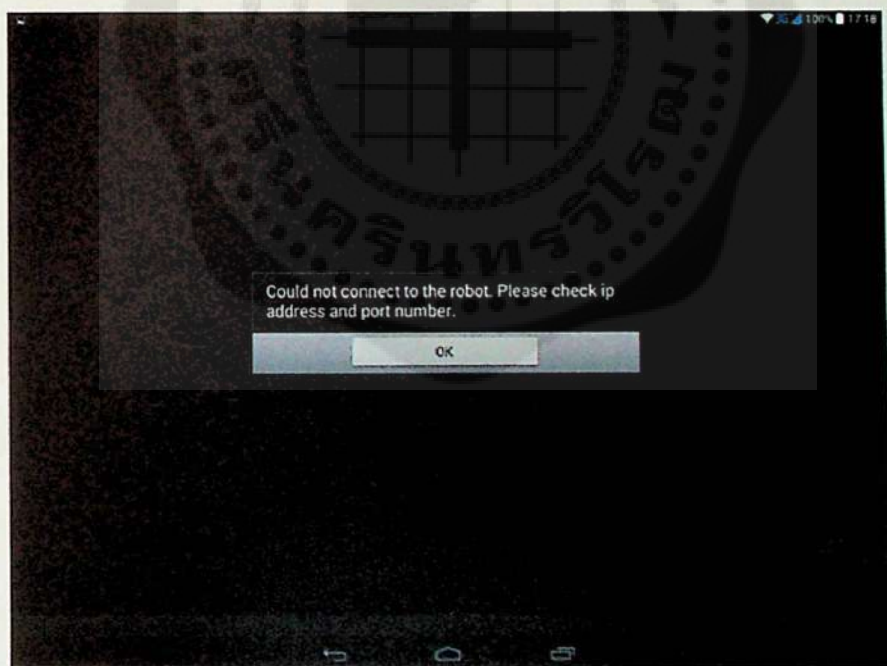


รูปที่ ก1 แสดงวิธีใช้ RobotSWU Application (1)

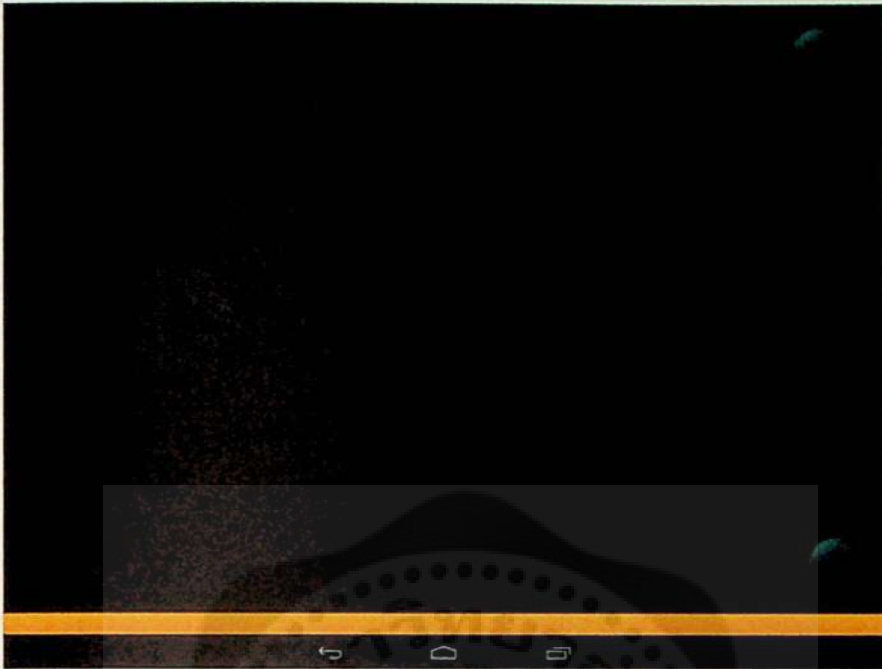
2. เมื่อ Click ที่ icon แล้วจะปรากฏหน้าจอ ดังรูปที่ ก2 แอปพลิเคชันจะทำการตรวจสอบว่าแท็บเล็ตได้มีการเชื่อมต่ออินเทอร์เน็ตหรือได้มีการเชื่อมต่อกับ Access point แล้วหรือยัง ถ้ายังไม่มี การเชื่อมต่อ แอปพลิเคชันจะแจ้งเตือนและจบการทำงาน ดังรูปที่ ก3 แต่ถ้าแอปพลิเคชันทำการเชื่อมต่อกับ Access point ได้แล้ว แอปพลิเคชันจะเข้าสู่โหมดการควบคุมบังคับหุ่นยนต์ได้ ดังรูปที่



รูปที่ ก2 แสดงวิธีใช้ RobotSWU Application (2)



รูปที่ ก3 แสดงวิธีใช้ RobotSWU Application (3)



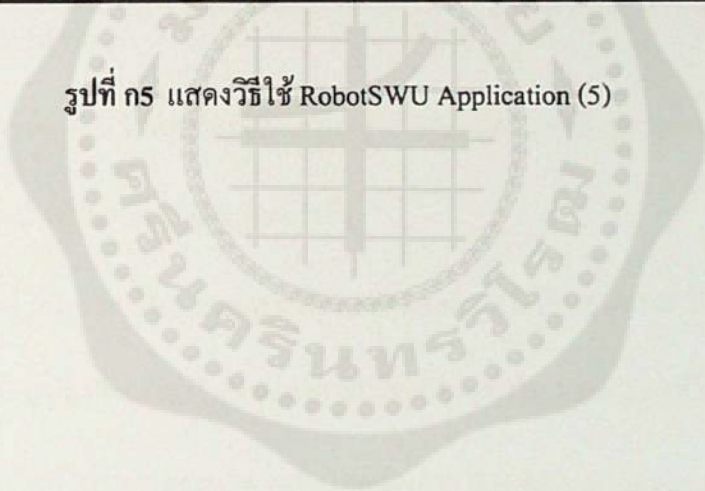
รูปที่ ก4 แสดงวิธีใช้ RobotSWU Application (4)

3. เมื่อเข้าสู่โหมดการควบคุมบังคับได้ หน้าจอจะปรากฏปุ่มทิศทางเพื่อใช้ในการควบคุมบังคับหุ่นยนต์ ดังรูปที่ ก5

- ปุ่มควบคุมบังคับทิศทางของหุ่นยนต์เพื่อให้หุ่นยนต์มีการเคลื่อนที่ไปในทิศทางต่างจะอยู่ทางด้านขวาของหน้าจอ
- ปุ่มควบคุมบังคับทิศทางการหมุนของกล้อง IP Camera จะอยู่ทางด้านซ้ายของหน้าจอ
- ปุ่มที่ใช้ในการ เปิด-ปิด หลอดไฟที่อยู่ด้านหน้าของตัวหุ่นยนต์จะอยู่ด้านบนสุดฝั่งขวา
- สัญญาณภาพและเสียงที่ส่งมาจากกล้อง IP Camera จะแสดงผลอยู่เต็มหน้าจอ
- สัญลักษณ์การแจ้งเตือนเมื่อหุ่นยนต์ทำการตรวจจับก๊าซไวไฟได้ จะอยู่บริเวณด้านบนฝั่งขวา



รูปที่ ก5 แสดงวิธีใช้ RobotSWU Application (5)



ภาคผนวก ข. วิธีใช้งาน openPicus IDE

openPicus IDE (Integrated Development Environment) คือ เครื่องมือสำหรับช่วยในการพัฒนาโปรแกรมบนโมดูล Flyport โดยจะอำนวยความสะดวกในการจัดการกับฟังก์ชันการทำงานต่างๆ ที่มีอยู่ทั้งหมดในโมดูล โดยทาง OpenPicus ได้พัฒนา IDE 2.1 เป็นเวอร์ชันแรก ปัจจุบันได้พัฒนาถึง IDE 2.3 แล้ว ซึ่งทำให้สามารถดึงความสามารถของ Flyport ออกมาได้อย่างเต็มที่ และสนับสนุนการพัฒนาโปรแกรมบน FlyportWiFi, Ethernet และ GPRS

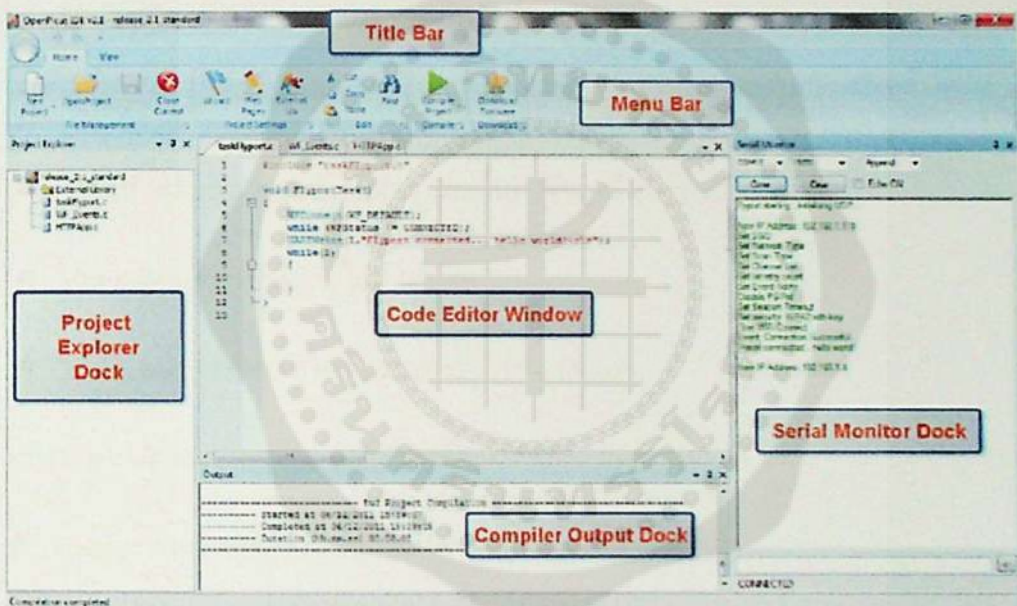
คุณลักษณะของ openPicus IDE

- สามารถตรวจสอบ จัดการ และแก้ไขทุกไฟล์โปรเจกได้ (ทั้งไฟล์แอปพลิเคชัน และ libraries ภายนอก)
- เป็นโปรแกรม editor ที่มีประสิทธิภาพ มีการทำ highlight syntax ตรวจสอบการจบฟังก์ชัน และมีเครื่องมือช่วยเหลือในการเขียนคำสั่ง
- มีเครื่องมือช่วยในการกำหนดคุณสมบัติการทำงานต่างๆ ให้กับอุปกรณ์ โดยทุกครั้งที่มีการแก้ไขเปลี่ยนแปลงการตั้งค่าใหม่ จะทำการปรับเปลี่ยนค่าใน libraries ให้โดยอัตโนมัติ
- ในการคอมไพล์โปรแกรมจะถูกควบคุมโดยตรงจาก C30 Microchip compiler
- สามารถนำเข้าไป Web pages, HTML pages, JavaScript, รูปภาพและไฟล์มีเดียทั้งหมดของหน้าเว็บ เข้าไปอยู่ในโปรเจกได้
- สามารถนำเข้าไป libraries จากภายนอก เพื่อที่จะสร้างฟังก์ชันการทำงานของโปรแกรมขึ้นมาใหม่ได้ เช่น libraries การอินเตอร์เฟสหรือการเชื่อมต่อกับอุปกรณ์ภายนอก เป็นต้น
- สามารถดาวน์โหลดโปรแกรมคำสั่งลงใน Flyport ได้โดยตรง โดยไม่ต้องผ่านบอร์ดแปลงหรือโปรแกรมเสริมอื่น

- มีฟังก์ชันที่ใช้ในการตรวจสอบเพื่อดูการทำงานของ serial port (serial monitor) ซึ่งผู้ใช้งานไม่จำเป็นต้องเปิดโปรแกรม Hyper-Terminal อื่นให้เสียเวลา โดย IDE นี้มีไว้ให้เรียบร้อยแล้ว

- IDE นี้รองรับการพัฒนาโปรแกรมบนโมดูล Flyport WiFi, Ethernet, GPRS

ส่วนประกอบของ openPicus IDE มีพื้นที่สำหรับจัดการกับโปรเจกต์หลายส่วนด้วยกัน โดยแต่ละส่วน ดังรูปที่ ข.1 ได้แก่

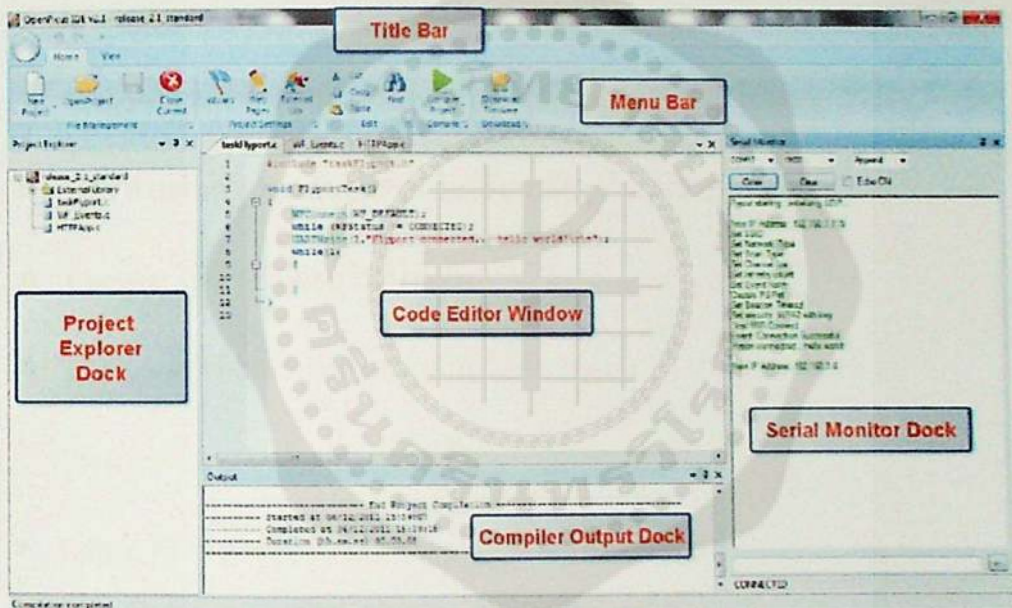


รูปที่ ข.1 ส่วนประกอบของ IDE

- Title Bar เป็นบริเวณที่แสดงชื่อของโปรเจกต์ และเวอร์ชันของ IDE ที่ใช้งาน
- Menu Bar ประกอบด้วย 2 แถบเครื่องมือ
 - Home tab ประกอบด้วยแถบเครื่องมือต่างๆ ได้แก่
 - File Management ประกอบด้วย
 - ❖ New Project สำหรับสร้างโปรเจกต์ใหม่

- มีฟังก์ชันที่ใช้ในการตรวจสอบเพื่อดูการทำงานของ serial port (serial monitor) ซึ่งผู้ใช้งานไม่จำเป็นต้องเปิดโปรแกรม Hyper-Terminal อื่นให้เสียเวลา โดย IDE นี้มีไว้ให้เรียบร้อยแล้ว
- IDE นี้รองรับการพัฒนาโปรแกรมบนโมดูล Flyport WiFi, Ethernet, GPRS

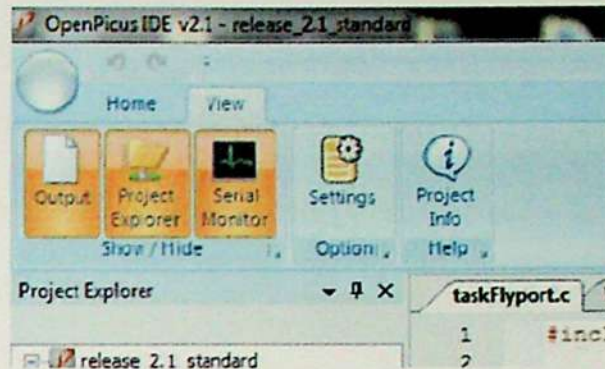
ส่วนประกอบของ openPicus IDE มีพื้นที่สำหรับจัดการกับโปรเจกต์หลายส่วนด้วยกัน โดยแต่ละส่วน ดังรูปที่ ข.1 ได้แก่



รูปที่ ข.1 ส่วนประกอบของ IDE

- Title Bar เป็นบริเวณที่แสดงชื่อของโปรเจกต์ และเวอร์ชันของ IDE ที่ใช้งาน
- Menu Bar ประกอบด้วย 2 แถบเครื่องมือ
- Home tab ประกอบด้วยแถบเครื่องมือต่างๆ ได้แก่
 - File Management ประกอบด้วย
 - ❖ New Project สำหรับสร้างโปรเจกต์ใหม่

- ❖ OpenProject สำหรับเปิดไฟล์โปรเจกต์ที่เก็บไว้ในเครื่อง
- ❖ Save สำหรับบันทึกโปรเจกต์หรือบันทึกการแก้ไขต่างๆ
- ❖ Close Current สำหรับปิดโปรเจกต์ปัจจุบันที่กำลังเปิดใช้งานอยู่
- Project Setting ประกอบด้วย
- ❖ Wizard ใช้ในการ config ค่าการทำงานต่างๆ ให้กับ Flyport
- ❖ Web Pages ใช้ในการนำหน้าเว็บที่เป็นไฟล์ HTML+accompanying CSS, รูปภาพหรือ JavaScript แล้วแปลงเป็นไฟล์ที่ใช้ได้กับ Flyport
- ❖ External Lib ใช้ในการนำเข้าหรือสร้างไฟล์ libraries สำหรับโปรเจกต์
- Edit สำหรับการ Cut/Copy/Past และค้นหาข้อมูลในโปรเจกต์
- Compile ใช้สำหรับคอมไพล์โปรเจกต์
- Download ใช้สำหรับ โพรแกรมคำสั่งให้กับ Flyport
- View tab ประกอบด้วยแถบเครื่องมือ
- Output สำหรับใช้ดูหรือซ่อนหน้าต่างแสดงผลคอนคอมไพล์
- Project Explorer ใช้ดูหรือซ่อนหน้าต่าง Project Explorer
- Serial Monitor ใช้ดูการทำงานบน serial port
- Settings สำหรับเปิดดูการตั้งค่าของ C30 compiler ดังรูปที่ ข2

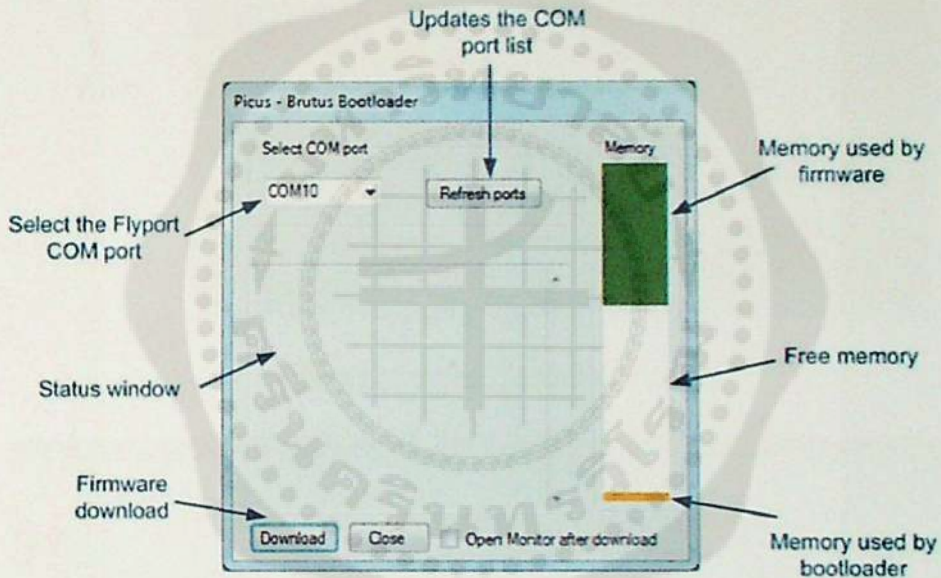


รูปที่ ข2 แถบเครื่องมือ View

- Project explorer เป็นพื้นที่แสดงทุกไฟล์ในโปรเจกซึ่งสามารถแก้ไขได้ ตามความต้องการของผู้ใช้โดยมีไฟล์แอปพลิเคชัน ดังนี้
 - taskFlyport.c เป็นไฟล์โปรแกรมที่ผู้ใช้งานเป็นผู้กำหนดขึ้นเอง
 - WF_events.c เป็นไฟล์สำหรับจัดการกับทุกเหตุการณ์ของ Flyport เช่น การสถาปนากการเชื่อมต่อสำเร็จ หรือไม่สำเร็จ (สำหรับโปรเจก WiFi) เป็นต้น
 - HTTPApp.c เป็นไฟล์สำหรับจัดการกับหน้าเว็บ (กรณีที่ทำเป็น Web server)
 - สำหรับไฟล์อื่นๆ นอกเหนือจากนี้ ก็จะเป็นไฟล์ libraries ภายนอก ซึ่งสามารถแก้ไขใหม่ หรือนำเข้ามาเพิ่มในโปรเจกได้
- Code Editor window เป็นส่วนสำหรับทำการแก้ไขโค้ด โปรแกรมของแต่ละโปรเจก
- Output window เป็นหน้าต่างแสดงผลรายงานต่างๆ ที่เกิดจากการคอมไพล์ และการทำงานของ IDE
- Serial monitor เป็นพื้นที่แสดงข้อมูลการสื่อสารโดยตรงของ Serial port กับ Flyport UART เพียงแค่ทำการเซตค่า baud rate กับ COM port ให้ตรงเท่านั้นก็สามารถดูข้อมูลที่รับ-ส่งได้แล้ว

1. การโหลดโปรแกรมคำสั่งให้กับ Flyport

การโหลดโปรแกรมที่ได้เขียนขึ้นลงโมดูล Flyport จะทำผ่าน serial bootloader ซึ่งการโหลดนั้นทำได้ง่าย โดยผู้ใช้กดไปที่แถบ Download Firmware จากนั้นจะปรากฏหน้าต่างดังรูปด้านล่าง ทำการเลือก COMport แล้วกดปุ่ม Download ได้เลย จากนั้นรอซักพัก จนขึ้นข้อความ successful ที่หน้าต่าง Status window ก็เป็นอันเสร็จเรียบร้อย



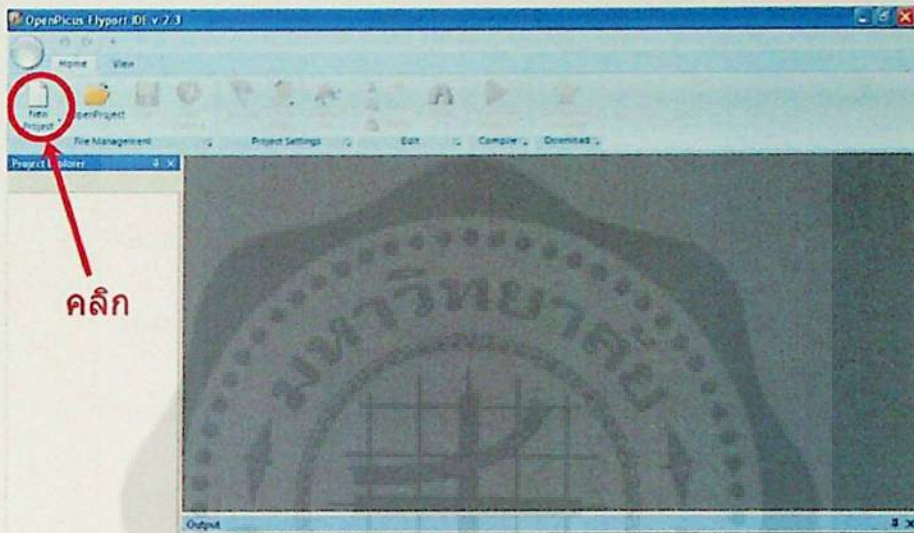
รูปที่ ข3 การ Download Firmware ให้กับ Flyport

2. การเริ่มต้นเขียนโปรแกรม

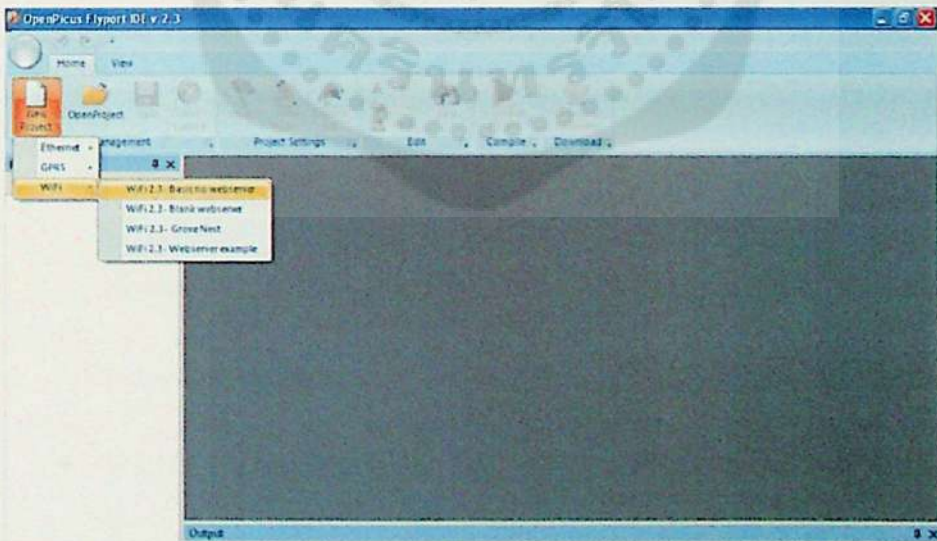
ตัวอย่างการเริ่มเขียนโปรแกรมจะใช้ IDE เวอร์ชันล่าสุด คือ 2.3 ในการอธิบายประกอบความเข้าใจโดยหลังจากที่ผู้ใช้ได้ลงโปรแกรมที่จำเป็นเรียบร้อยแล้ว ต่อไปก็จะเป็นการเริ่มต้นเขียนโปรแกรม

- การสร้างโปรเจคใหม่

- ขั้นแรกให้ทำการเปิดโปรแกรม openPicus IDE ขึ้นมาก่อน จากนั้นคลิกที่ New Project ดังรูปที่ ข4
- เลือก WiFi > WiFi 2.3-Basic no webserver ดังรูปที่ ข5



รูปที่ ข4 การสร้างโปรเจกใหม่ (1)



รูปที่ ข5 การสร้างโปรเจกใหม่ (2)

จากรูปจะเห็นว่า IDE มีรูปแบบการสร้างโปรเจกต์ให้สามารถเลือกได้ทั้ง WiFi Ethernet และ GPRS ในที่นี้จะขอยกตัวอย่างเพียง WiFi และในส่วน WiFi ก็ยังสามารถเลือกรูปแบบตามลักษณะการทำงานได้อีกด้วยได้แก่

- WiFi2.3-Basic no webserver: เป็นรูปแบบของการสร้างโปรเจกต์พื้นฐานแบบง่าย โดยไม่มีการสร้าง webserver ซึ่งจะประกอบด้วยไฟล์ testFlyport.c และ WF_Events.c ดังรูปที่ ข6
- WiFi2.3-Blank webserver: รูปแบบการสร้างโปรเจกต์ที่มีการใช้งาน webserver โดยจะมีการสร้างไฟล์ HTTPApp.c เพิ่มขึ้นสำหรับทดสอบเว็บเพจ ซึ่งเป็นส่วนติดต่อกับเว็บเพจที่ทาง openPicus ได้เตรียมไว้ให้นำไปเป็นตัวอย่างในการศึกษา โดยจะรับข้อความเข้าเพียงอย่างเดียวไม่สามารถส่งข้อความออกได้ ซึ่ง HTTPApp.c เป็นเพียงส่วนของฟังก์ชันการจัดการทำงานบน webserver เท่านั้น และไม่ใช้แบบไดนามิก (no dynamic) ดังรูปที่ ข7
- WiFi2.3-GroveNest: เป็นรูปแบบการสร้างโปรเจกต์เพื่อใช้งานกับบอร์ด Grove Nest ซึ่งเป็นบอร์ดที่รองรับการทำงานร่วมกับ Flyport WiFi ในการติดต่อกับเซ็นเซอร์ต่างๆ โดยไฟล์ที่สร้างขึ้นมาจะคล้ายกับรูปแบบ WiFi2.3-Basic no webserver ดังรูปที่ ข8
- WiFi2.3-Webserver example: เป็นรูปแบบการสร้างโปรเจกต์มาตรฐานของ Flyport ที่รวมฟังก์ชันการทำงานต่างๆ ไว้ให้ใช้งาน เช่น IO monitor และ analog inputs monitor โดยมีไฟล์ HTTPApp.c ที่คอยจัดการการทำงานบนเว็บเพจ ที่สำคัญเป็น

แบบไดนามิก (dynamic) ด้วย ซึ่งการสร้างไฟล์ก็จะคล้ายกับ WiFi2.3-Blank
webserver

```

taskFlyport.c  WF_Events.c
1      #include "WF_Events.h"
2
3
4      /*****
5      FUNCTIONS TO HANDLE THE WIFI EVENTS
6      *****/

```

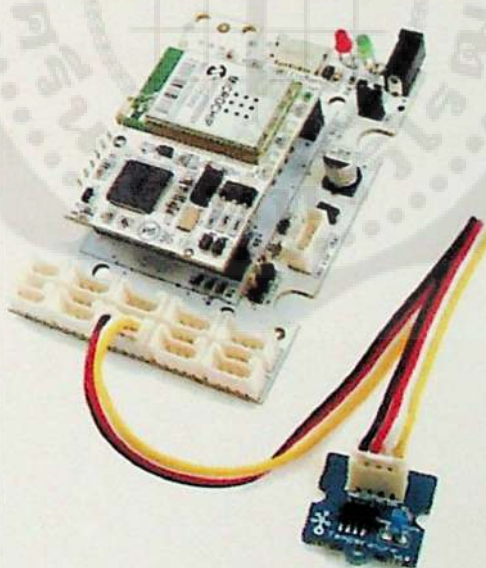
รูปที่ ข6 การสร้างโปรเจกใหม่ (3)

```

taskFlyport.c  WF_Events.c  HTTPApp.c
1      /*****
2      * Software License Agreement
3      *
4      * This is free software; you can redistribute it and/or modify it under
5      * the terms of the GNU General Public License (version 2) as published by
6      * the Free Software Foundation AND MODIFIED BY OpenPicus team.

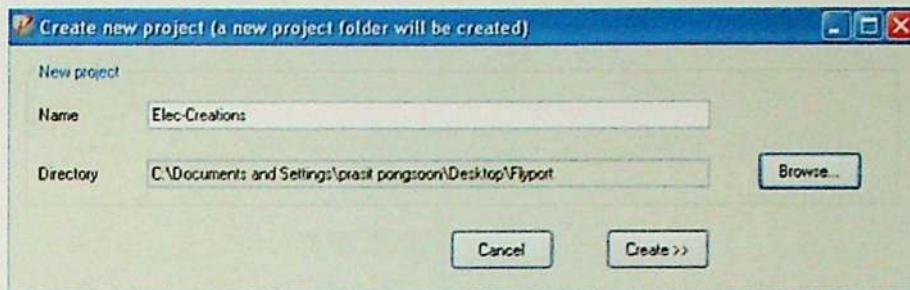
```

รูปที่ ข7 การสร้างโปรเจกใหม่ (4)



รูปที่ ข8 บอร์ด Grove Nest

- หลังจากเลือกรูปแบบการสร้างโปรเจกต์แล้ว จะมีหน้าต่างขึ้นมาให้ทำการตั้งชื่อโปรเจกต์ และเลือก directory ที่ต้องการเก็บ จากนั้นเลือก Create>>



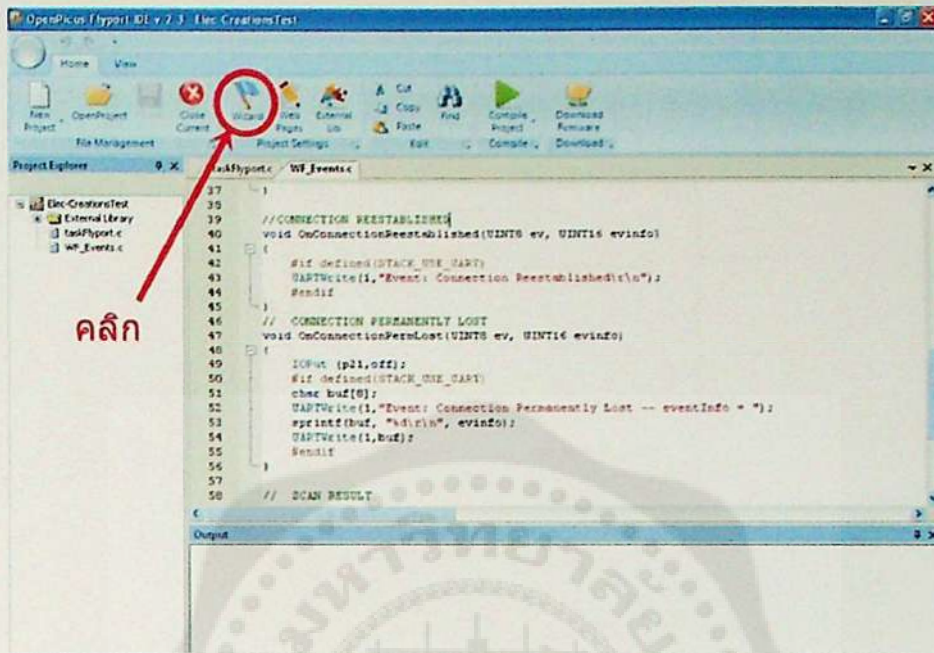
รูปที่ ข9 การสร้างโปรเจกต์ใหม่ (5)

การสร้างรูปแบบโปรเจกต์ที่ได้กล่าวมานั้น จะมีการสร้างฟังก์ชันการทำงานต่างๆ ขึ้นมาให้สำหรับการติดต่อสื่อสารของ Flyport แบบอัตโนมัติ แต่ผู้ใช้สามารถปรับเปลี่ยน แก้ไข หรือเพิ่มเติมไฟล์ต่างๆ ตามที่ต้องการได้

3. การตั้งค่าการทำงานให้กับ Flyport

จากที่เราได้เลือกรูปแบบและสร้างโปรเจกต์ขึ้นมาเป็นที่เรียบร้อยแล้ว ขั้นตอนต่อไปจะเป็นการคอนฟิกค่าต่างๆ ให้กับ Flyport ดังนี้

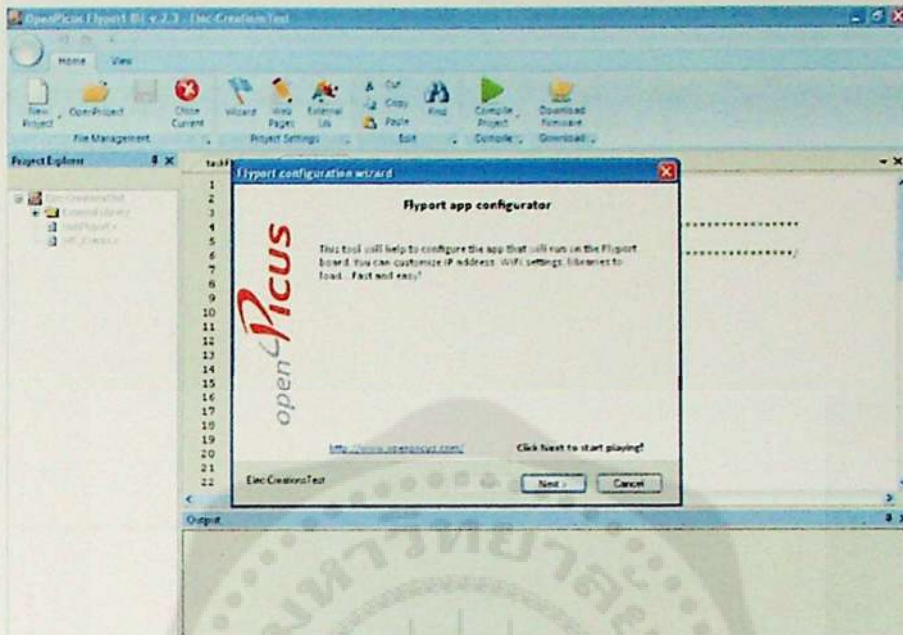
- เข้าไปที่ Wizard



รูปที่ ข10 การตั้งค่าการทำงานให้กับ Flyport (1)

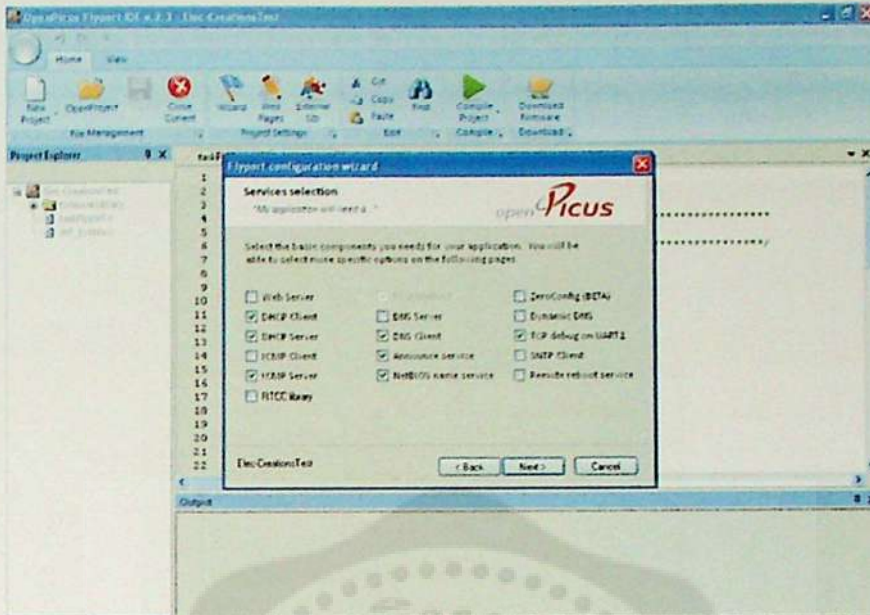
สำหรับการคอนฟิกค่าในเบื้องต้นเราจะใช้ค่าพารามิเตอร์ที่โปรแกรมกำหนดมาให้ โดยจะเป็นค่าเริ่มต้น หรือค่า Default โดยเพียงคลิก Next ไปเรื่อยๆ จนกว่าจะสิ้นสุดขั้นตอน

- หลังจากคลิกเลือก Wizard จะมีหน้าต่างปรากฏขึ้นมาให้ทำการเลือก Next



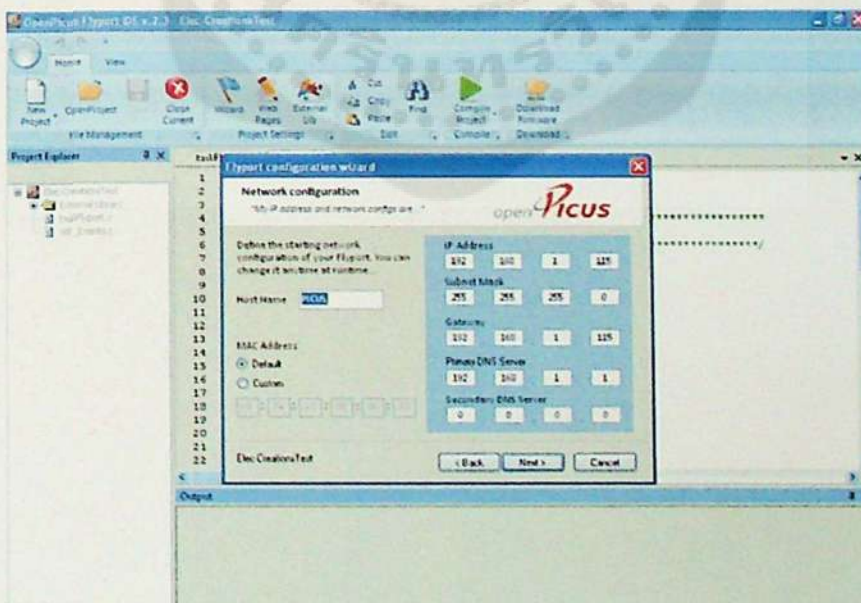
รูปที่ ข11 การตั้งค่าการทำงานให้กับ Flyport (2)

- จากนั้นจะปรากฏหน้าต่าง Services selection สำหรับให้เราเลือกบริการที่ต้องการ (ในที่นี้กำหนดค่าตาม default แต่สำหรับการนำไปใช้งานจริงต้องกำหนดให้มีความเหมาะสม) จากนั้นคลิก Next



รูปที่ ข12 การตั้งค่าการทำงานให้กับ Flyport (3)

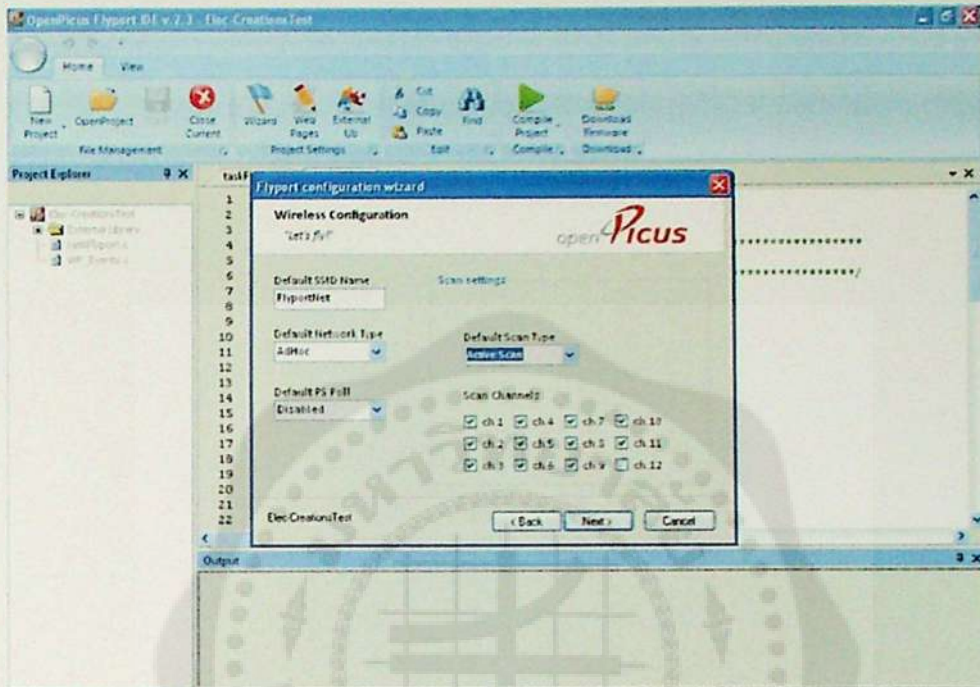
- จากนั้นจะเป็นการตั้งค่าให้กับเครือข่าย (ในที่นี้กำหนดตามค่า default) จากนั้นคลิกNext



รูปที่ ข13 การตั้งค่าการทำงานให้กับ Flyport (4)

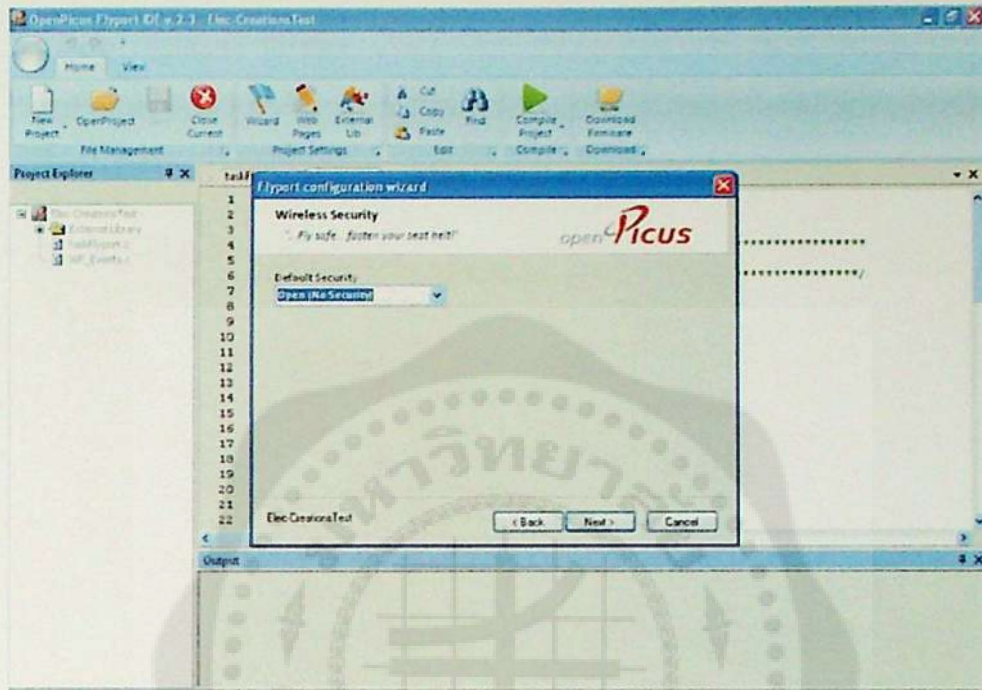
- ต่อไปเป็นการกำหนดค่า Wireless (ในที่นี้กำหนดตามค่า default) จากนั้นคลิก

Next



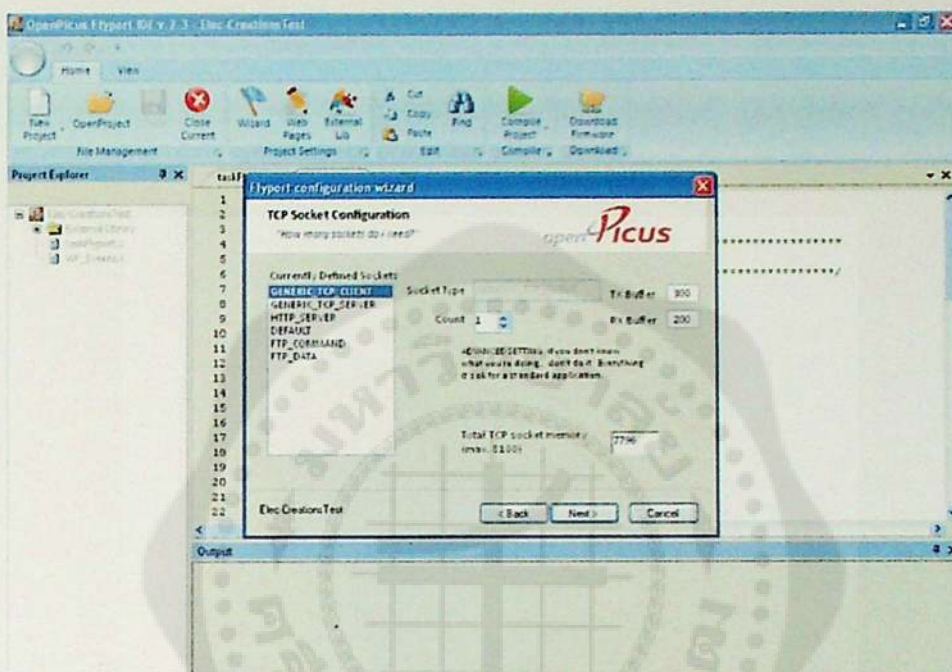
รูปที่ ข14 การตั้งค่าการทำงานให้กับ Flyport (5)

- ต่อไปเป็นการกำหนดรูปแบบการรักษาความปลอดภัย (ในที่นี้กำหนดตามค่า default) จากนั้นคลิก Next



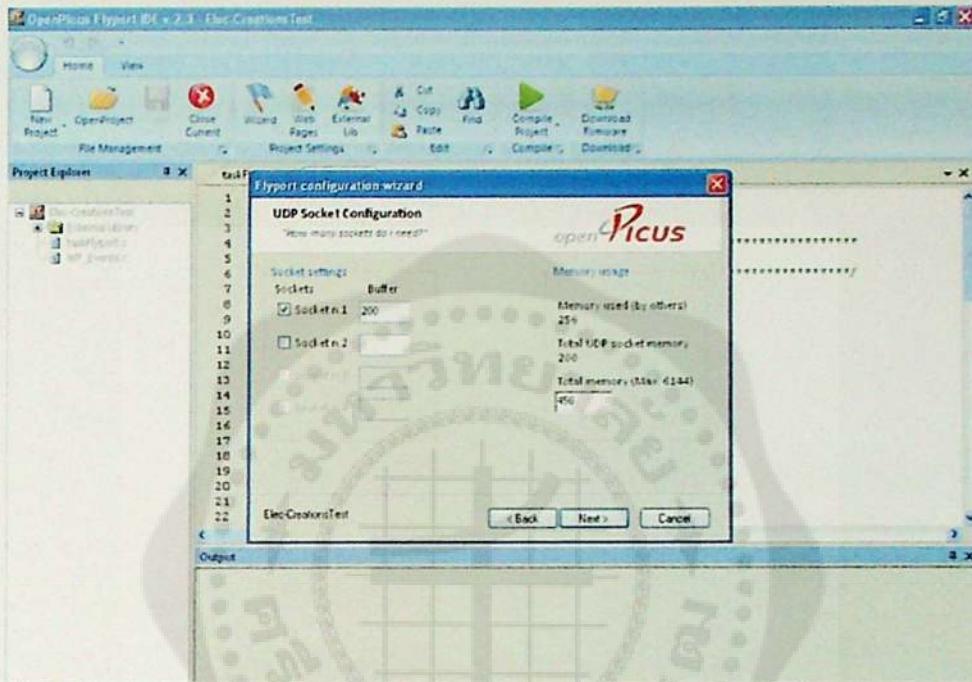
รูปที่ ข15 การตั้งค่าการทำงานให้กับ Flyport (6)

- ต่อกำหนดค่า TCP socket ซึ่งจะเป็นการเลือกกำหนดขนาดของ Buffer ในการรับส่งข้อมูลให้กับรูปแบบของ socket ที่ใช้งาน (ในที่นี้กำหนดตามค่า default) จากนั้นคลิก Next



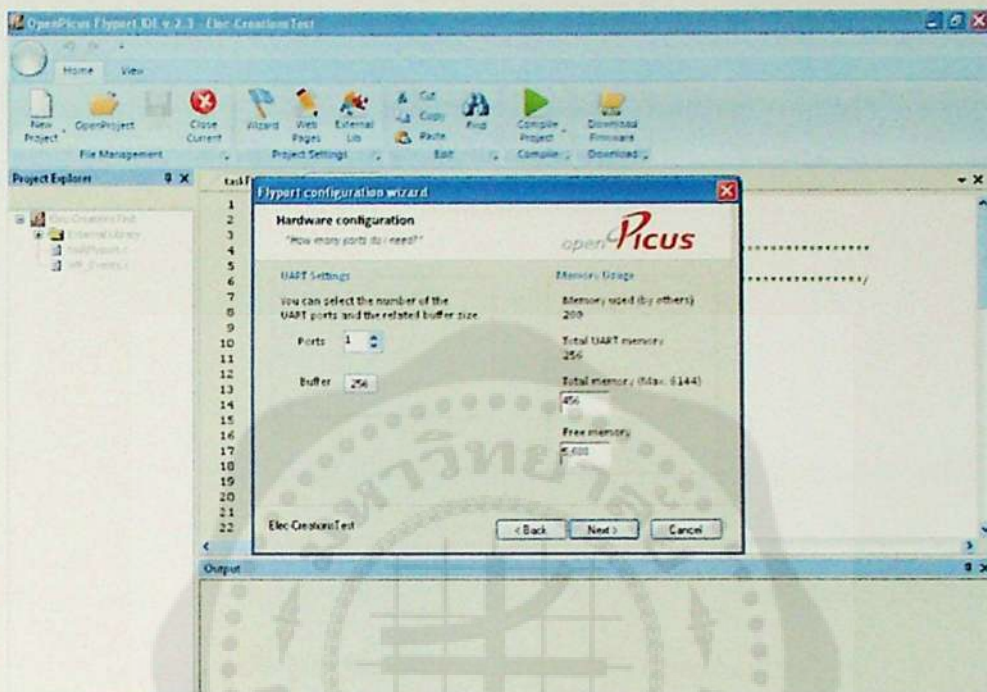
รูปที่ ข16 การตั้งค่าการทำงานให้กับ Flyport (7)

- ต่อไปเป็นกำหนดค่าให้กับ UDP socket ซึ่งเป็นการเลือกหมายเลข socket และกำหนดขนาดของ Buffer ในการรับส่งข้อมูล (ในที่นี้กำหนดตามค่า default) จากนั้นคลิก Next



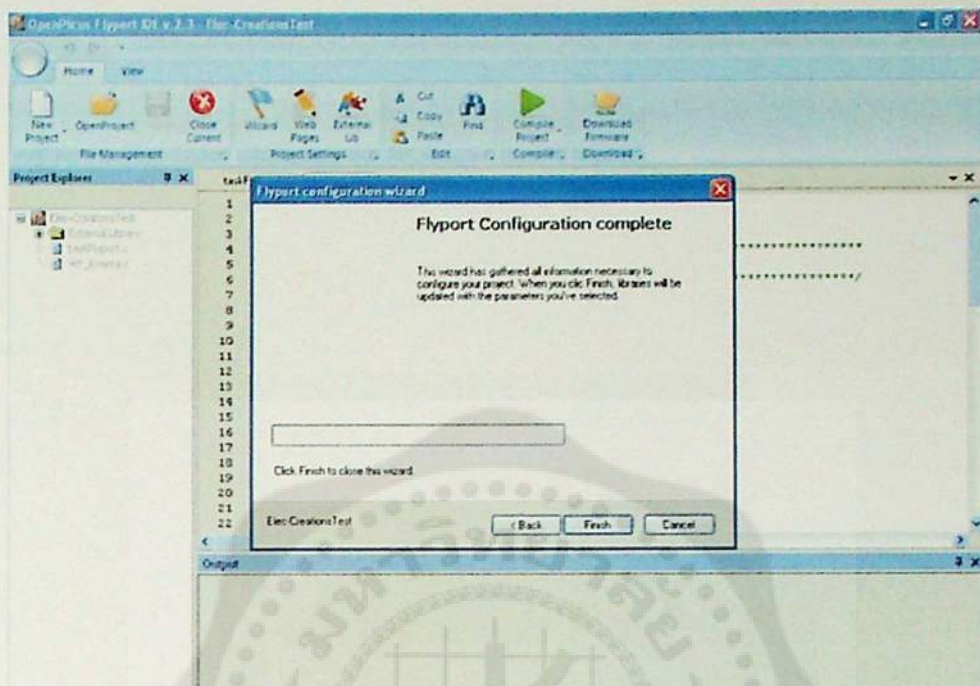
รูปที่ ข17 การตั้งค่าการทำงานให้กับ Flyport (8)

- ต่อไปเป็นการกำหนดขนาดของ Buffer ให้กับแต่ละ UART ซึ่งสามารถใช้งานได้ถึง 4 พอร์ตพร้อมกัน (ในที่นี้กำหนดตามค่า default) จากนั้นคลิก Next



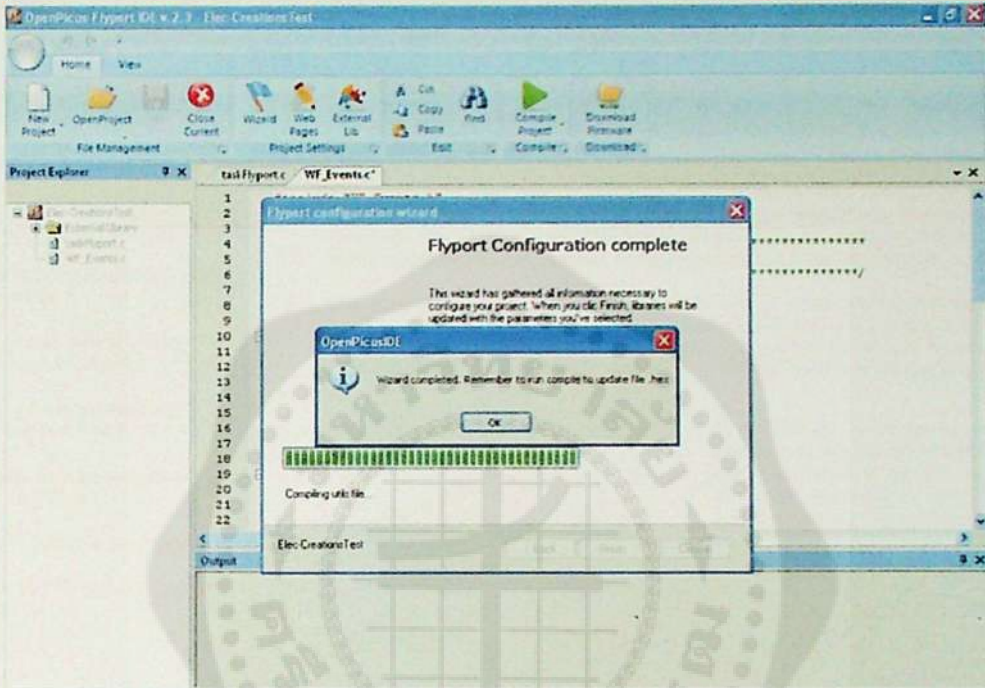
รูปที่ ข18 การตั้งค่าการทำงานให้กับ Flyport (9)

- เมื่อทำการตั้งค่าต่างๆ เสร็จเรียบร้อยแล้ว ให้ทำการคลิก Finish



รูปที่ ข19 การตั้งค่าการทำงานให้กับ Flyport (10)

- จากนั้น IDE จะทำการแก้ไขค่าต่างๆ ใน libraries ตามที่เราได้กำหนดให้โดยอัตโนมัติ ซึ่งต้องทำการรอสักครู่จนกว่าจะขึ้น Wizard completed จึงถือเป็นการเสร็จสิ้นการตั้งค่าด้วย Wizard จากนั้นคลิก OK



รูปที่ 20 การตั้งค่าการทำงานให้กับ Flyport (11)

ภาคผนวก ค. Flyport Ethernet

Connect the embedded world to the internet

Internet

openPICUS

FLYPORT Ethernet

System on module

- Microchip PIC 24F 16bit processor
- 16Mbit Flash Memory for web pages
- Firmware upgrade over Internet
- ENC424J600 10/100 Base-T Ethernet
- Available with or without RJ45 connector onboard
- Easy development with openPicus free IDE
- openPicus framework is based on freeRTOS
- Serial bootloader onboard
- Webserver (customizable)
TCP Socket
UDP Socket
SNTP
SMTP
- 5V or 3.3V power supply
- RTCC onboard
- Remappable pins at runtime
- Digital I/O
- Analog Inputs
- 4 UARTs, 1 SPI, 1 I2C

Applications

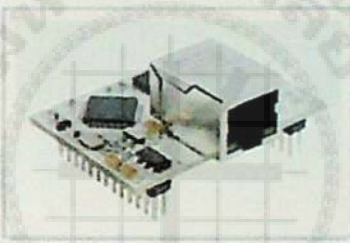
- Webserver based user interfaces to the embedded
- Sensors and automation
- Internet of Things
- Audio over IP
- Building automation and remote control
- Industrial/process management

Introduction

FLYPORT Ethernet is a miniature web server module featuring a fully integrated 10/100 Base-T ethernet interface and several interfaces to the 'real world'.

The module integrates a powerful 16 bit processor which runs custom applications and an Ethernet interface controller with integrated MAC & PHY, hardware cryptographic security and a unique MAC address. An RJ45 connector can be mounted on the module, or routed at external add-ons boards.

The module provides the embedded world with a powerful 'Internet engine' to a browser-based interface over Internet, in a small footprint, at low power and low cost. Real time data can be both displayed and/or updated from a standard web browser, even on smartphone or tablets, because FLYPORT supports dynamic web pages.



FLYPORT is powered by openPicus framework based on FreeRTOS. The free IDE allows to create applications, to import web pages and to compile and download code to the module.

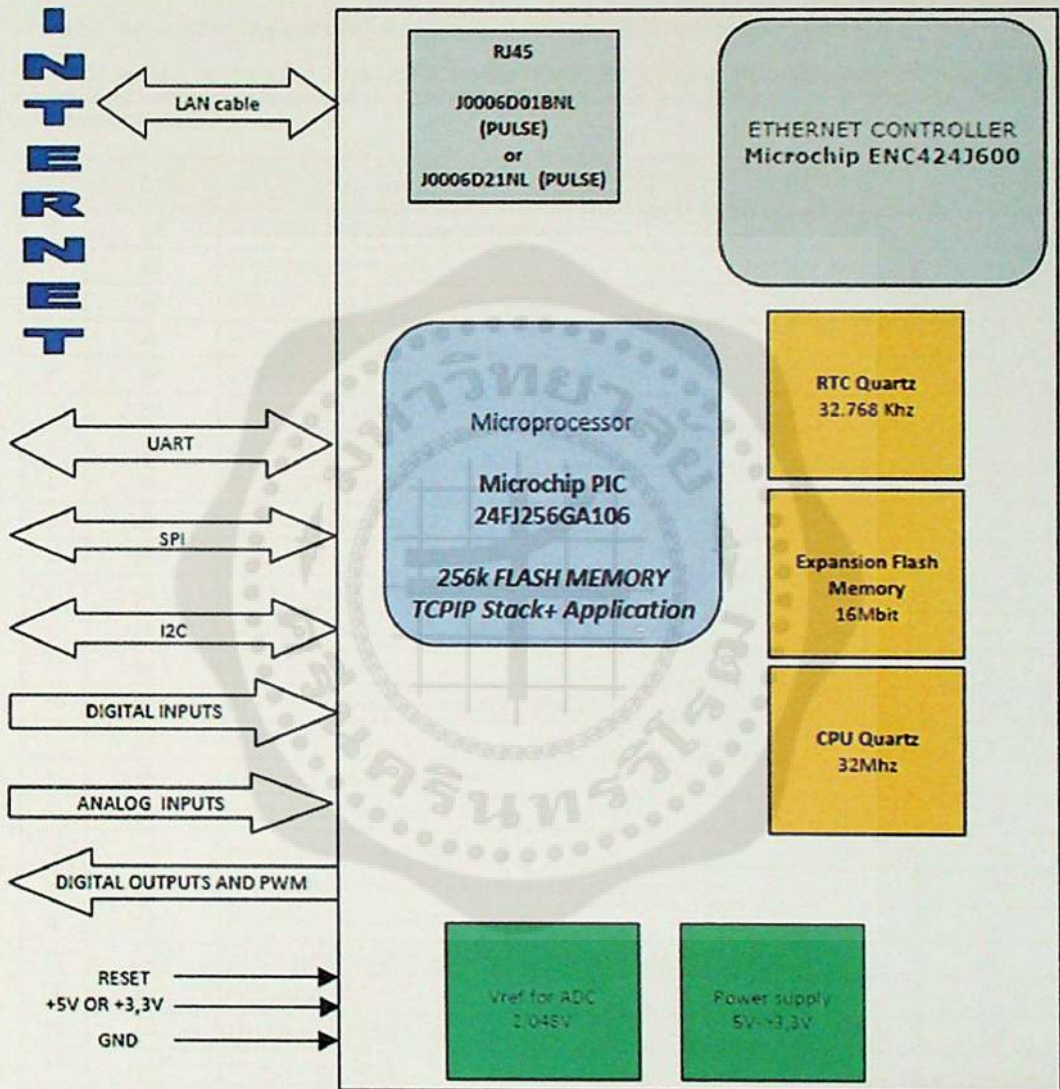
Features

| | |
|------------------|---|
| 16 Bit Processor | PIC24FJ256, 256K Flash, 16K Ram, 16Mips@32Mhz |
| Transceiver | ENC424J600 10/100 Base-T Ethernet controller |
| Power Supply | 5V or 3.3V, integrated LDO |
| Integrated RTC | 32,768 Khz quartz onboard |
| Digital I/O | up to 21, remappable at Runtime |
| Analog In | up to 4, 10bits ADC, Vref=2,048V |
| Communication | up to 4 UARTs, SPI, I2C |
| Flash Memory | SST25VF016B 16Mbit storage Flash memory |
| Connector | 2x26 ways, 2 rows, standard 2.54mm male pin header compatible with SAMTEC ssw-113-01-t-d or FCI 65781-013 |
| Dimensions | 35 x 50 x 25 mm, 15 grams |

Connect the embedded world to the internet

openPICUS

Block Diagram



Connect the embedded world to the internet

Internet

openPICUS

JP1 Connector

JP1 is the main and common connector of each module of the FLYPORT family (Wi-Fi, Ethernet).

FLYPORT modules are based on Microchip PIC processor and offer remappable pins function. User can customize the hardware configuration by firmware. It means that you can have up to 4 UARTs, up to 18 Digital I/Os and 9 independent PWMs.

| Pin | Pin Name | Description (default setting) | 5V Tolerant | Remappable |
|-----|----------|--|-------------|------------|
| 1 | p1 | GPIO (I2C bus Clock signal by default) | Yes | No |
| 2 | p2 | GPIO (input by default) | Yes | Yes |
| 3 | p3 | GPIO (I2C bus Data signal by default) | Yes | No |
| 4 | p4 | GPIO (output by default) | Yes | Yes |
| 5 | p5 | GPIO (input by default) | Yes | Yes |
| 6 | p6 | GPIO (output by default) | Yes | Yes |
| 7 | p7 | GPIO (input by default) | Yes | No |
| 8 | p8 | GPIO (SPI bus Clock SCLK by default) | Yes | Yes |
| 9 | p9 | GPIO (input by default) | Yes | Yes |
| 10 | p10 | GPIO (SPI bus Out SDO by default) | Yes | Yes |
| 11 | p11 | GPIO (input by default) | Yes | Yes |
| 12 | p12 | GPIO (SPI bus In SDI by default) | Yes | Yes |
| 13 | p13 | UART RX input | Yes | Yes |
| 14 | p14 | GPIO (SPI bus chip select CS by default) | Yes | Yes |
| 15 | p15 | UART TX output | Yes | Yes |
| 16 | p16 | +5V Power supply input (note 1) | - | - |
| 17 | p17 | GPIO (output by default) | No | Yes |
| 18 | p18 | Analog input #4 (note 2) | No | Yes |
| 19 | p19 | GPIO (output by default - connected on red Led OUT4) | No | Yes |
| 20 | p20 | Analog input #3 (note 2) | No | Yes |
| 21 | p21 | GPIO (output by default - connected on red Led OUT5) | No | No |
| 22 | p22 | GND (ground) | - | - |
| 23 | p23 | Analog input #1 (note 2) | No | Yes |
| 24 | p24 | +3.3V (see note 1) | - | - |
| 25 | p25 | Analog input #2 (note 2) | No | Yes |
| 26 | p26 | Reset (Active Low) | No | Yes |

Note 1. FLYPORT can be powered at 5V or at 3.3V. If the module is powered by 5V on pin 16, pin 24 is the output of integrated LDO (max output current available:100mA). If powered using a single 3,3V on pin 24, leave pin 16 unconnected!

Note 2. FLYPORT has a precise voltage reference for analog 2,048V: this is the max voltage readable by these pins

Pins 16-18-20-22-24-26 are directly compatible with the Microchip Pickit programmer connector

Connect the embedded world to the internet



Internet

openPICUS

JP2 Connector

JP2 is not available on FLYPORT WI-FI. FLYPORT Ethernet provides some add-on pins on this connector. The signals routed on this connector are coming directly from the microcontroller (to expand number of available I/O pins) and from Ethernet transceiver, to use an external RJ45 connector on your expansion board.

| Pin | Pin Name | Description (default setting) | 5V Tolerant | Remappable |
|-----|----------|-------------------------------|-------------|------------|
| 1 | p27 | Digital output | No | No |
| 2 | p28 | Digital input | No | No |
| 3 | p29 | Digital output | No | No |
| 4 | p30 | Digital input | No | No |
| 5 | p31 | Digital output | No | No |
| 6 | p32 | Digital input | No | No |
| 7 | p33 | Digital output | No | No |
| 8 | p34 | Digital input | Yes | No |
| 9 | - | Not connected | - | - |
| 10 | - | Not connected | - | - |
| 11 | - | Not connected | - | - |
| 12 | - | Not connected | - | - |
| 13 | - | Not connected | - | - |
| 14 | - | Not connected | - | - |
| 15 | - | Not connected | - | - |
| 16 | - | Not connected | - | - |
| 17 | - | Ethernet signal TD- | - | - |
| 18 | - | Ethernet signal TD+ | - | - |
| 19 | - | Ethernet signal TCT | - | - |
| 20 | - | Ethernet signal RD+ | - | - |
| 21 | - | Ethernet signal RCT | - | - |
| 22 | - | Ethernet signal RD- | - | - |
| 23 | - | RJ45 connector pins 7-8 | - | - |
| 24 | - | RJ45 connector pins 4-5 | - | - |
| 25 | - | RJ45 connector LED2 | - | - |
| 26 | - | RJ45 connector LED1 | - | - |

Note. FLYPORT Ethernet comes in 2 version: with RJ45 connector onboard (default) and without RJ connector.

POE: Pins 23, 24 can be used to power supply the module through the ethernet (it's not a real POE), but an external circuitry is needed.

Connect the embedded
world to the internet

Internet

open **PICUS**

Technical Information

| Electrical | | Ethernet 10/100 Base-T | |
|-----------------------|------------------------------|------------------------|----------------------------------|
| Power supply | 5V or 3.3V | Compatibility | 10/100 Base-T |
| Current consumption | Ethernet cable not connected | Max Data Rate (Burst) | 100 Mbit |
| | Ethernet cable connected | MAC | Preprogrammed unique MAC address |
| 96.2 mA | 158.4mA | | |
| Mechanical | | Certifications | |
| Operating Temperature | -20°C...+85°C | CE, FCC in progress | |
| Dimensions | 35*50*25mm | | |

ประวัติย่อประวัติผู้จัดทำโครงการ

ชื่อ-สกุล นางสาวกัญญ์ณฉวี ปุสสะรังษี
วัน เดือน ปีเกิด 4 กันยายน 2535
สถานที่เกิด สุพรรณบุรี
สถานที่อยู่ปัจจุบัน 423 หมู่ 2 ต.บ้าน ไช้ อ.อู่ทอง จ.สุพรรณบุรี
 72160
โทรศัพท์ 086-7505551
E-mail zimz_indiiz@hotmail.com



ประวัติการศึกษา

พ.ศ. 2548 มัธยมศึกษาตอนต้น โรงเรียนกาญจนาภิเษกวิทยาลัย สุพรรณบุรี
 พ.ศ. 2553 มัธยมศึกษาตอนปลาย โรงเรียนกาญจนาภิเษกวิทยาลัย สุพรรณบุรี
 พ.ศ. 2557 กำลังศึกษาระดับปริญญาตรี สาขาวิศวกรรมไฟฟ้า
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ประวัติย่อประวัติผู้จัดทำโครงการ

ชื่อ-สกุล นายพรสวรรค์ สิทธิชัยวัฒนา
 วัน เดือน ปีเกิด 28 มกราคม 2536
 สถานที่เกิด กรุงเทพมหานคร
 สถานที่อยู่ปัจจุบัน 52/208 หมู่ 7 ต.หลักหก อ.เมือง ปทุมธานี
 12000
 โทรศัพท์ 087-0558977
 E-mail auiaui.013@gmail.com



ประวัติการศึกษา

พ.ศ. 2548 มัธยมศึกษาตอนต้น โรงเรียนเซนต์โยเซฟเมืองเอก
 พ.ศ. 2552 มัธยมศึกษาตอนปลาย โรงเรียนพระหฤทัยคอนเมือง
 พ.ศ. 2557 กำลังศึกษาระดับปริญญาตรี สาขาวิศวกรรมไฟฟ้า
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ