# บทที่ 3

# การออกแบบ และขั้นตอนการดำเนินงานวิจัย

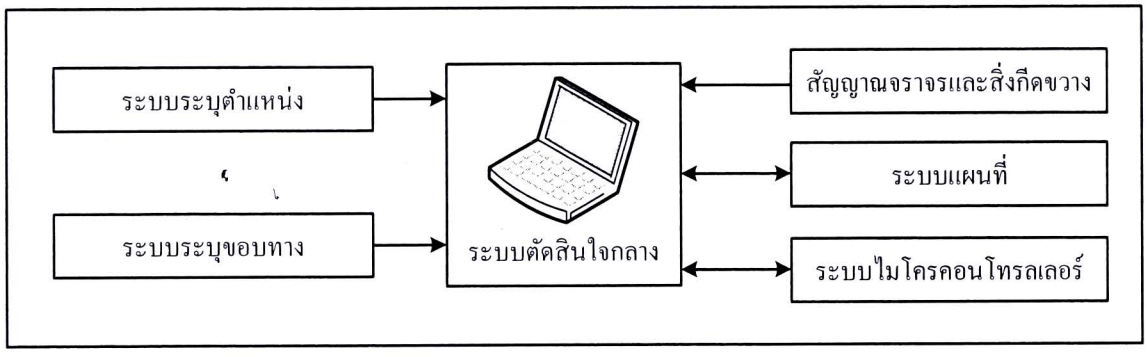
ในการออกแบบโครงสร้างการทำงานของระบบรถอัจฉริยะสำหรับการเคลื่อนที่อัตโนมัตินั้น จำเป็นต้องรู้ถึงโครงสร้างโดยรวมทั้งหมดของรถเคลื่อนที่อัตโนมัติ เพื่อให้ทราบถึงแนวทาง การออกแบบและการปฏิบัติงาน ซึ่งในการปฏิบัติงานในส่วนย่อยต่างๆ จะมีการพัฒนาระบบฮาร์ดแวร์ ควบคู่ไปกับการพัฒนาระบบซอฟต์แวร์ สำหรับการพัฒนาระบบต่างๆ ในลักษณะนี้จะทำให้เห็นข้อเด่นและข้อด้อยของระบบฮาร์ดแวร์ และระบบซอฟต์แวร์ เมื่อเกิดปัญหาข้อผิดพลาดหรือความขัดแย้งกันของทั้งสองระบบ การแก้ปัญหาจะสามารถทำได้อย่างทันท่วงที และหาข้อสรุปที่ลงตัวกันได้ โดยไม่จำเป็นต้องทำการเปลี่ยนโครงสร้างของระบบใดมากนัก อีกประการหนึ่งการนำอุปกรณ์ตรวจจับชนิดต่างกันมาใช้งานร่วมกัน โดยแต่ละอุปกรณ์ก็จะมีหน้าที่การทำงานที่แตกต่างกันออกไปตามลักษณะการใช้งานของอุปกรณ์นั้นๆ การโปรแกรมให้อุปกรณ์ทุกตัวสามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพ จะทำโดยระบบตัดสินใจกลางซึ่งมีหน้าที่เป็นศูนย์กลางการควบคุม และการเชื่อมต่อของระบบทั้งหมดเข้าด้วยกัน โดยระบบตัดสินใจกลางจะรับค่าจากอุปกรณ์ตรวจจับต่างๆ ที่ติดตั้งไว้บนตัวรถอัตโนมัติมาประมวลผล และวิเคราะห์ค่าที่นำเข้ามา หลังจากนั้นระบบตัดสินใจกลางจะส่งคำสั่งควบคุมไปยังอุปกรณ์ต่างๆ เพื่อให้รถสามารถทำงาน และเคลื่อนที่ไปได้โดยอัตโนมัติ เพื่อให้สะดวกในการพัฒนาระบบรถอัจฉริยะได้แบ่งการดำเนินการออกเป็น 7 ส่วน ดังนี้

- การออกแบบโครงสร้างการทำงานของระบบรถอัจฉริยะโดยรวม
- การดัดแปลงและปรับปรุงระบบรถไฟฟ้า
- ระบบตัดสินใจกลาง
- ระบบระบุขอบทาง
- ระบบระบุตำแหน่ง
- ระบบระบุทิศทางด้วยแผนที่
- ระบบวิเคราะห์สัญญาณจราจรและสิ่งกีดขวาง

โดยแต่ละส่วนจะอธิบายแผนการดำเนินการ ข้อมูลเชิงเทคนิค รวมถึงอุปกรณ์ที่ใช้ในการพัฒนาระบบรถอัตโนมัติ และตำแหน่งในการติดตั้งอุปกรณ์ ซึ่งจะขยายความในหัวข้อถัดไป

### 3.1 การออกแบบระบบรถอัจฉริยะโดยรวม

ระบบโครงสร้างของระบบรถอัจฉริยะ เกิดจากการนำระบบที่ใช้ในการวิเคราะห์สภาพแวดล้อมของรถขณะเคลื่อนที่ และระบบไมโครคอนโทรลเลอร์ มาทำการเชื่อมต่อกันโดยมีระบบตัดสินใจกลางเป็นศูนย์กลางการเชื่อมต่อระบบย่อยต่างๆ เพื่อทำการวิเคราะห์ข้อมูลที่เข้ามาจากระบบย่อย แล้วส่งคำสั่งควบคุมให้กับไมโครคอนโทรลเลอร์ควบคุมอุปกรณ์ที่ติดตั้งบนตัวรถสั่งงานให้รถเคลื่อนที่ไปบนถนนได้โดยอัตโนมัติ ซึ่งสามารถอธิบายทิศทางการส่งข้อมูลและการเชื่อมต่อของระบบย่อยกับระบบตัดสินใจกลาง ดังแสดงในรูปที่ 3.1

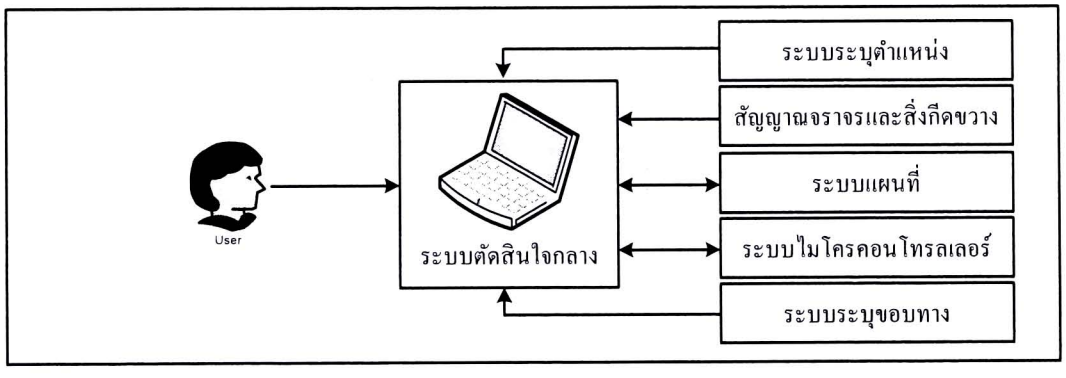



รูปที่ 3.1 โครงสร้างระบบรถอัจฉริยะ

จากรูปที่ 3.1 จะเห็นความสัมพันธ์ของทั้ง 5 ระบบย่อย และระบบตัดสินใจกลางเชื่อมต่อเข้าด้วยกันเพื่อทำงานร่วมกันอย่างเป็นระบบ ทำให้เกิดระบบรถอัตโนมัติขึ้นมา โดยจำแนกอธิบายแต่ละส่วนได้ดังนี้

#### 3.1.1 แผนการทำงานของระบบตัดสินใจกลาง

ระบบตัดสินใจกลาง คือ โปรแกรมที่ทำงานอยู่บนเครื่องประมวลผล ทำหน้าที่วิเคราะห์ข้อมูลที่รับมาจากระบบย่อยต่างๆ และส่งคำสั่งควบคุมต่างๆ ให้ระบบ ดังแสดงในรูปที่ 3.2

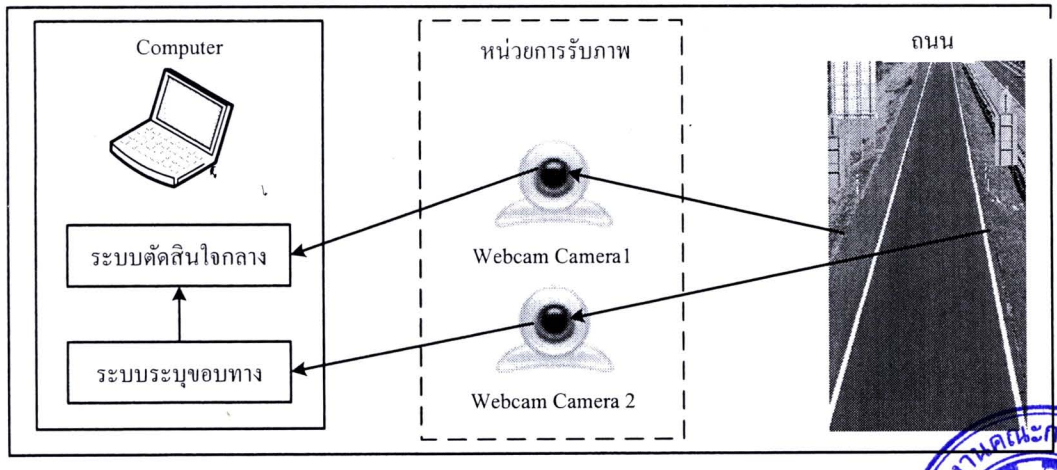



รูปที่ 3.2 ระบบการเชื่อมต่อของระบบตัดสินใจกลาง

จากรูปที่ 3.2 ผู้ใช้ต้องกำหนดค่าทิศทางการเคลื่อนที่ และแผนที่ให้กับระบบตัดสินใจกลาง หากไม่มีการกำหนดค่าหรือคำสั่งให้เคลื่อนที่ รถก็จะไม่สามารถทำการเคลื่อนที่ได้เอง

### 3.1.2 แผนการทำงานของระบบระบุขอบทาง

ระบบระบุขอบทางทำหน้าที่วิเคราะห์ภาพที่ได้มาจากกล้องเว็บแคม แล้วส่งผลการวิเคราะห์ให้กับระบบตัดสินใจกลาง เพื่อเป็นข้อมูลสำหรับใช้วิเคราะห์การเคลื่อนที่ของระบบรถอัตโนมัติ ดังรูปที่ 3.3

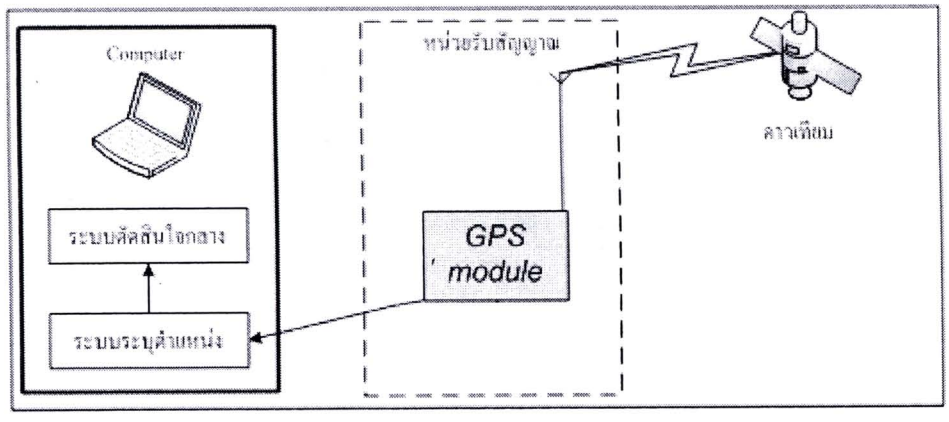



รูปที่ 3.3 ระบบการเชื่อมต่อของระบบระบุขอบทาง

สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

### 3.1.3 แผนการทำงานระบบระบุตำแหน่ง

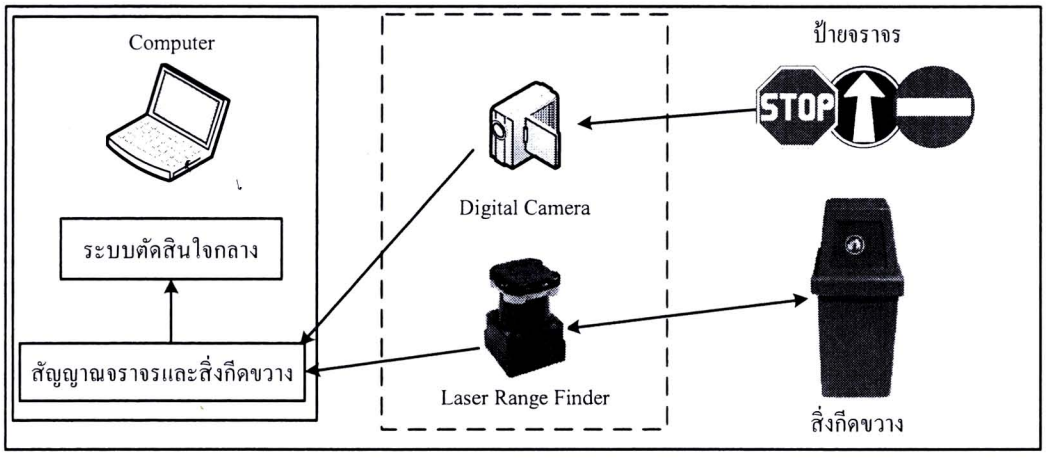
ระบบระบุตำแหน่งทำหน้าที่วิเคราะห์สัญญาณที่รับมาจากอุปกรณ์ GPS แล้วส่งผลการวิเคราะห์ให้กับระบบตัดสินใจกลางเพื่อระบุตำแหน่งของรถ ดังรูปที่ 3.4




รูปที่ 3.4 ระบบการเชื่อมต่อของระบบระบุตำแหน่ง

### 3.1.4 แผนการทำงานของระบบวิเคราะห์สัญญาณจราจรและสิ่งกีดขวาง

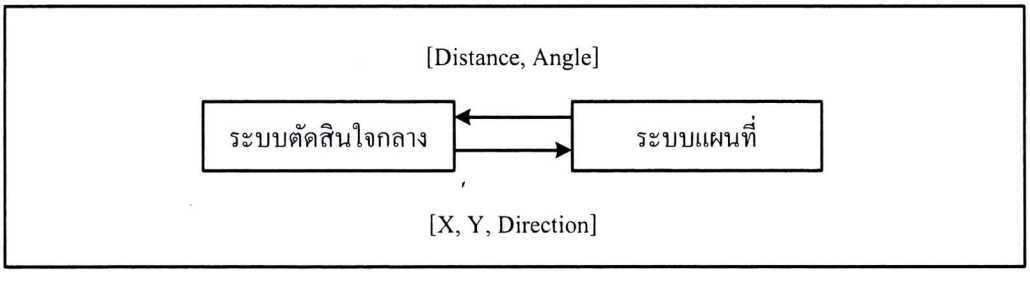
ระบบวิเคราะห์สัญญาณจราจรและสิ่งกีดขวาง จะมีหน้าที่การทำงานอยู่ 2 ลักษณะ คือ วิเคราะห์ป้ายจราจร เพื่อระบุความหมายของป้ายจราจร โดยใช้การประมวลผลภาพที่ถ่ายได้จากกล้องดิจิตอล แล้วนำไปวิเคราะห์ว่าป้ายนั้นมีความหมายอย่างไร อีกลักษณะหนึ่ง คือ การระบุสิ่งกีดขวาง โดยวิเคราะห์ค่าข้อมูลที่ได้จากอุปกรณ์ Laser Range Finder เพื่อระบุว่ามีสิ่งกีดขวางหรือไม่ แล้วส่งผลการวิเคราะห์ให้กับระบบตัดสินใจกลางเพื่อใช้เป็นข้อมูลในการตัดสินใจว่าจะเคลื่อนที่อย่างไร ดังรูปที่ 3.5




รูปที่ 3.5 ระบบการเชื่อมต่อของระบบวิเคราะห์สัญญาณจราจรและสิ่งกีดขวาง

### 3.1.5 แผนการทำงานของระบบแผนที่

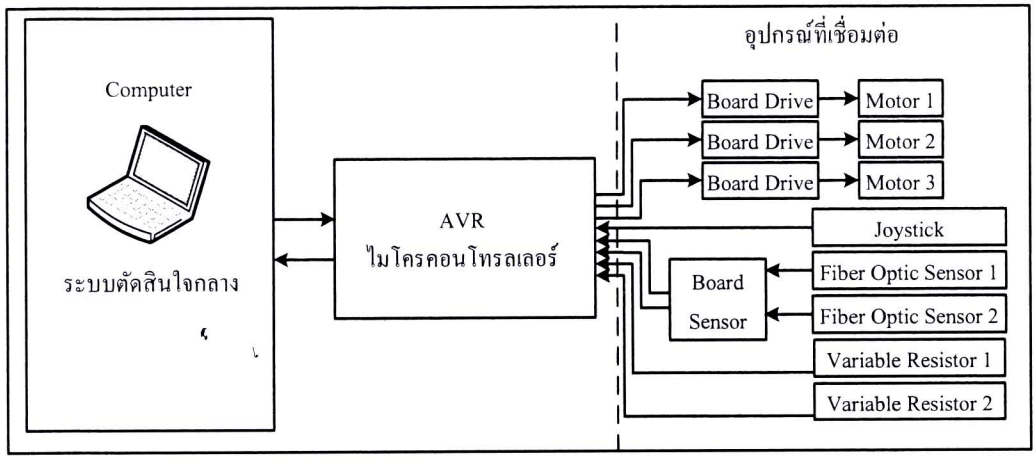
ระบบแผนที่ทำหน้าที่วิเคราะห์สภาพของถนนเมื่อถึงทางแยกหรือทางโค้ง ระบบตัดสินใจกลางจะสอบถามมายังระบบแผนที่ โดยระบบตัดสินใจกลางจะส่งค่าตำแหน่งและพิกัดของรถจากสัญญาณที่รับจาก GPS ให้กับระบบแผนที่ ซึ่งจะทำการคำนวณและส่งทิศทางการเคลื่อนที่กลับไปให้ระบบตัดสินใจกลางดังรูปที่ 3.6




รูปที่ 3.6 ระบบการเชื่อมต่อของระบบแผนที่

### 3.1.6 แผนการทำงานของระบบไมโครคอนโทรลเลอร์

ระบบไมโครคอนโทรลเลอร์ ทำหน้าที่ควบคุมอุปกรณ์ฮาร์ดแวร์ต่างๆ ที่ติดตั้งไว้บนรถ โดยรับคำสั่งจากระบบตัดสินใจกลางและส่งค่าจากอุปกรณ์ตรวจจับต่างๆไปให้ระบบตัดสินใจกลางเพื่อใช้ในการวิเคราะห์มุมการเลี้ยว และความเร็วของรถ ดังรูปที่ 3.7




รูปที่ 3.7 ระบบการเชื่อมต่อของระบบไมโครคอนโทรลเลอร์

## 3.2 การดัดแปลงและปรับปรุงระบบรถไฟฟ้า

ในการพัฒนาระบบรถอัจฉริยะนั้น ผู้พัฒนาจำเป็นต้องทำการปรับปรุงและดัดแปลงสภาพรถให้อยู่ในรูปแบบที่เหมาะสมสำหรับการวิจัย เพราะเดิมทีแล้วรถที่จะนำมาพัฒนาเป็นรถระบบไฟฟ้าที่ใช้ขับขี่ในสนามกอล์ฟโดยมีคนบังคับมาก่อน เพื่อให้การพัฒนาระบบรถอัจฉริยะเป็นไปตามขอบเขตของงานวิจัย คือ สามารถเคลื่อนที่ไปบนถนนได้โดยอยู่ในสภาพไร้คนขับ ดังนั้นระบบสั่งงานทั้งหมดไปที่ตัวรถ จะถูกเปลี่ยนจากระบบการสั่งงานจากมนุษย์โดยตรง มาเป็นการสั่งงานผ่านระบบไฟฟ้าทั้งหมด อาทิเช่น จากเดิมการเปลี่ยนองศาหน้ารถจะใช้แรงของมนุษย์ ออกแรงกระทำไปที่พวงมาลัยทำให้เกิดแรงบิดส่งไปยังแกนพวงมาลัย แกนพวงมาลัยจะส่งกำลังไปยังชุดบังคับล้อให้ล้อรถเกิดการเปลี่ยนองศาทำให้รถเกิดการเลี้ยวเมื่อต้องการให้รถสามารถขับเคลื่อนและทำงานต่างๆได้เองอย่างอัตโนมัติ ผู้พัฒนาจึงนำระบบไมโครคอนโทรลเลอร์ เข้ามาเป็นตัวควบคุมสั่งงาน และรับ-ส่งค่าข้อมูลที่นำเข้ามาจากอุปกรณ์ตรวจจับที่ผู้พัฒนาได้ ทำการติดตั้งไว้ ค่าข้อมูลเหล่านี้จะถูกนำไปวิเคราะห์ด้วยระบบตัดสินใจกลาง โดยข้อมูลที่นำเข้ามาเหล่านี้เปรียบเสมือนกับประสาทการรับรู้ของมนุษย์ ที่จะนำมาใช้ในการพิจารณาในการขับขี่รถให้สามารถเคลื่อนที่ไปได้อย่างปลอดภัย โดยมีการใช้สมองกล และอุปกรณ์ตรวจจับต่างๆ มาแทนระบบประสาทการรับรู้และการตัดสินใจของมนุษย์นั่นเอง

### 3.2.1 ข้อมูลของรถที่นำมาพัฒนาเป็นรถอัจฉริยะ

รูปที่ 3.8 รถไฟฟ้าที่นำมาพัฒนา

รถที่นำมาพัฒนาในงานวิจัย เดิมเป็นรถกอล์ฟไฟฟ้า ที่เคยใช้งานมาก่อนในสนามกอล์ฟ ดังแสดงในรูปที่ 3.8 โดยมีข้อมูลรายละเอียดที่เกี่ยวข้องกับตัวรถ ดังแสดงในตารางที่ 3.1

ตารางที่ 3.1 รายละเอียดของรถไฟฟ้าที่ถูกใช้ดัดแปลงเป็นรถอัจฉริยะ

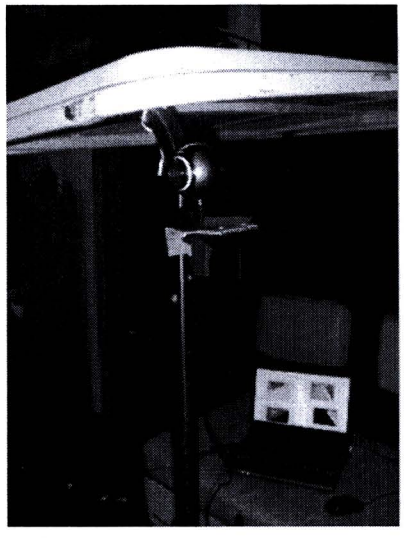
| Specs of electric golf car. | |
|---|---|
| Motor (DC) | 3KW |
| Battery(Deep cycle) | 36 Volt |
| Seat | 2 |
| Speed | 25 km/hr |
| Climbing ability | 20° |
| Dimension(m) WxLxH | 1.2 x 2.4 x 1.8 |
| Turning radius(m) | 3.5 |
| Chassis | Steel |
| Weight | 300 kg |

### 3.2.2 การติดตั้งอุปกรณ์ตรวจจับ

อุปกรณ์ตรวจจับต่างๆ ที่นำมาใช้ในการพัฒนาระบบรถอัจฉริยะในครั้งนี้ ประกอบไปด้วยอุปกรณ์ที่ทำหน้าที่แตกต่างกันไปตามความต้องการที่จะนำมาใช้ในการรับค่าจากอุปกรณ์เหล่านี้ ซึ่งในการติดตั้งอุปกรณ์ต่างๆ จะพิจารณาจากความต้องการของระบบว่าต้องมีการใช้งานสัญญาณการตรวจจับจากอะไรบ้าง แล้วทำการติดตั้งอุปกรณ์ตรวจจับนั้นๆ ลงบน ตัวรถ ซึ่งในการดำเนินการได้มีการติดตั้งอุปกรณ์ต่างๆ ดังนี้

**การติดตั้งกล้องสำหรับตรวจจับขอบทาง**

การทำงานของกล้องที่ใช้กับระบบรถอัจฉริยะนี้ถือเป็นอุปกรณ์ที่มีความสำคัญมากอุปกรณ์หนึ่งเนื่องจากอุปกรณ์นี้เปรียบได้กับระบบประสาทการมองเห็นของมนุษย์ ซึ่งจำเป็นมากในการนำมาใช้พิจารณาการสั่งงานควบคุมระบบต่างๆ ให้รถสามารถเคลื่อนที่อยู่บนถนนได้โดยอัตโนมัติ เมื่อสมองกลสามารถมองเห็นภาพได้ ก็สามารถที่จะนำค่าที่ได้จากภาพมาทำการวิเคราะห์และตัดสินใจสั่งงานการเคลื่อนที่ให้กับระบบได้ กล้องที่ใช้เป็นกล้อง Logitech รูปที่ 3.9 แสดงการติดตั้งกล้องไว้บนตัวรถ

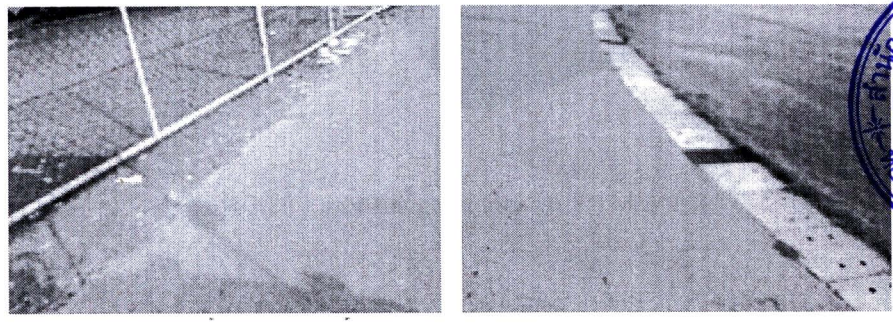


รูปที่ 3.9 ตำแหน่งที่ติดตั้งกล้องเว็บแคม

ในการติดตั้งกล้องเว็บแคม ได้ทำการหาตำแหน่งที่เหมาะสมของกล้องหลังจากพิจารณาเห็นว่าให้ติดตั้งในตำแหน่งที่สูงกว่าเดิมประมาณ 15 เซนติเมตร จากระบบเดิม [13] โดยสูงจากพื้นที่ระยะ 160 เซนติเมตร และมีมุมเอียงลง 15 องศา เพื่อลดปัญหาเรื่องของเงาของหลังคารถเมื่อเวลาแสงแดดส่องพาดไปบนหลังคารถแล้วเงาของหลังคารถ จะทอดไปบนถนนด้านข้างตัวรถ หาก

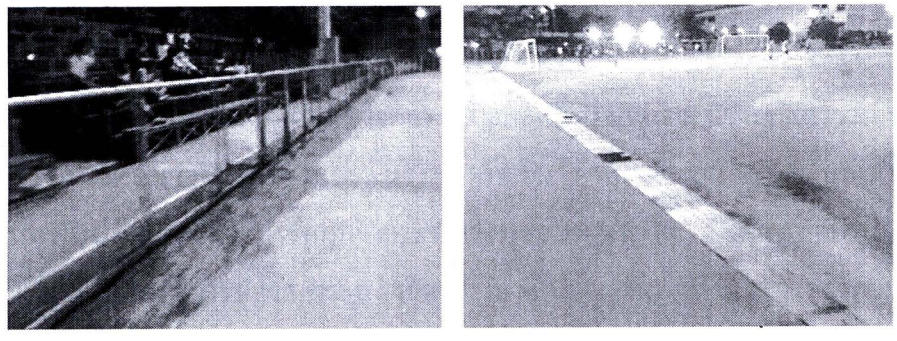
กล้องอยู่ตำแหน่งเดิมโปรแกรมที่ใช้ตรวจหาขอบถนน จะตรวจพบเงาของหลังคารถเป็นขอบถนน และส่งค่าที่พบไปให้ระบบตัดสินใจกลาง ระบบตัดสินใจกลางจะคิดว่ารถวิ่งเข้าใกล้ขอบถนน และจะส่งคำสั่งควบคุมให้รถเลี้ยวออกจากขอบถนน ซึ่งจริงๆ แล้วภาพที่นำเข้ามาจากกล้อง และนำไปวิเคราะห์ไม่ใช่ขอบถนนจริงๆ แต่เป็นค่าขอบที่เกิดจากความแตกต่างระหว่างเงาของวัตถุ กับถนน ซึ่งตรงจุดนี้โปรแกรมไม่สามารถแยกแยะขอบที่เกิดจากเงาของวัตถุ กับขอบของพื้นผิวถนนจริงได้ สิ่งที่เกิดขึ้นคือ เมื่อรถเคลื่อนที่มายังบริเวณที่รถจะตรวจพบเงาของหลังคารถ รถจะหักหลบเงาตัวเองจนตกลงข้างทางและไม่สามารถเคลื่อนที่ต่อได้

การทดสอบโดยนำรถไปขับบริเวณสนามฟุตบอลแล้วบันทึกภาพ โดยภาพที่ได้จากการบันทึกภาพจากตำแหน่งกล้องเดิม จะทำให้เห็นเงาของหลังคารถ ดังรูปที่ 3.10 และภาพที่ได้จากกล้องในตำแหน่งใหม่ จะมีวิสัยทัศน์ที่ไกลกว่าภาพที่ถ่ายได้จากตำแหน่งกล้องเดิม และภาพที่ได้จะไม่เห็นเงาของหลังคารถ ดังแสดงในรูปที่ 3.11



สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

รูปที่ 3.10 ภาพที่ถ่ายจากตำแหน่งกล้องในระบบเดิม ซึ่งติดห่างจากหลังคา



รูปที่ 3.11 ภาพที่ถ่ายจากตำแหน่งกล้องในระบบใหม่ ซึ่งติดสูงใกล้หลังคา

**การติดตั้ง Laser Range Finder**

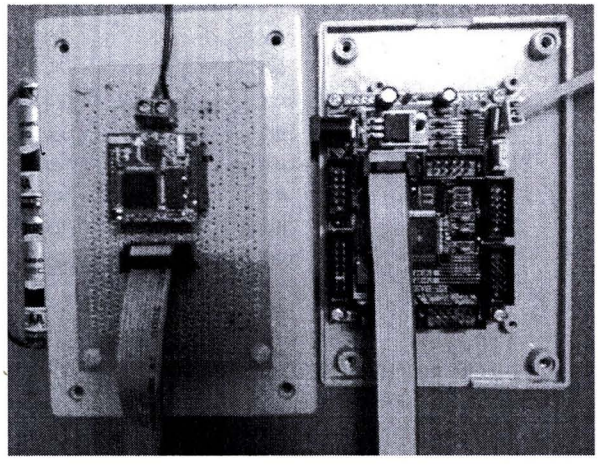
อุปกรณ์ที่นำมาใช้พัฒนาระบบรถอัจฉริยะครั้งนี้ใช้ Laser Range Finder รุ่น UTM-30LX [23] ซึ่งสามารถวัดระยะทางได้สูงสุด 30 เมตร มุมกว้างสูงสุด 270 องศา ในการอ่านแต่ละครั้งจะอ่านได้ 1080 steps เป็นอุปกรณ์ของบริษัท HOKUYO ของประเทศญี่ปุ่น ดังแสดงในรูปที่ 3.12 ซึ่ง Laser Range ของบริษัท HOKUYO นี้เป็น Laser Range ที่มีขนาดเล็กและมีราคาไม่สูงมาก แต่มีประสิทธิภาพที่ค่อนข้างจะใกล้เคียงกับ Laser Range ของบริษัท SICK

รูปที่ 3.12 Laser Range Finder รุ่น UTM-30LX

Laser Range Finder เป็นอุปกรณ์ตรวจจับอีกชนิดหนึ่งที่นำมาใช้ในการพัฒนาระบบ รถอัตโนมัติ เนื่องจากในการสั่งงานการเคลื่อนที่ของรถอัตโนมัติสิ่งที่จำเป็นรองลงมาจากการมองเห็นก็คือ การประเมินหรือการระบุระยะห่างของหน้ารถ กับวัตถุต่างๆ เพื่อใช้ในการวิเคราะห์แนวทางการเคลื่อนที่ หรือวิเคราะห์ความเสี่ยงที่จะทำการเคลื่อนที่ไปข้างหน้า เพราะอุปกรณ์ Laser Range Finder นี้จะเป็นอุปกรณ์ที่ทำหน้าที่อ่านค่าระยะห่าง และขนาดของวัตถุที่อยู่ด้านหน้ารถที่ตรวจพบแล้วส่งค่าดังกล่าวให้ระบบตัดสินใจกลาง ระบบตัดสินใจกลางจะทำการวิเคราะห์ และทราบทันทีว่าหน้ารถมีวัตถุขวางอยู่ หลังจากนั้นระบบตัดสินใจกลางจะทำการกำหนดเส้นทางการเคลื่อนที่ขึ้นมาใหม่ โดยจะไม่ใช้ค่าข้อมูลจากระบบระบุขอบทางมาใช้ ในการตัดสินใจเพียงอย่างเดียว แต่จะใช้ค่าข้อมูลที่รับมาจากระบบระบุสิ่งกีดขวางมาเป็นข้อมูลหลักที่ใช้ในการสั่งควบคุมให้รถเคลื่อนที่ผ่านวัตถุที่ตรวจพบในทิศทางใดทิศทางหนึ่ง โดยระบบตัดสินใจกลางจะนำค่าที่อ่านได้จากระบบระบุสิ่งกีดขวาง ไปคำนวณกับระบบระบุขอบทาง ซึ่งจะทำให้รู้ขนาดความกว้างจากขอบถนนด้านใดด้านหนึ่งที่กว้างกว่า ระบบตัดสินใจกลางก็จะตัดสินใจให้รถเคลื่อนที่ไปทางนั้น เพื่อหลีกเลี่ยงการหลบวัตถุจนรถตกขอบถนน

**การติดตั้งชุดรับสัญญาณ Digital Compass Module**

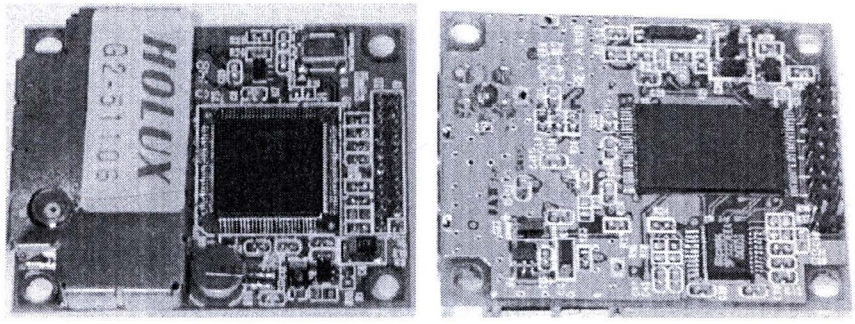
อุปกรณ์นี้เป็นอุปกรณ์ที่มีหน้าที่บอกองศาหน้ารถขณะทำการเคลื่อนที่โดยทำการติดตั้งอุปกรณ์ Digital Compass Module โดยใช้บัส $I^2C$ เชื่อมต่อกับไมโครคอนโทรลเลอร์โดยใช้สายสัญญาณ 2 เส้นได้แก่ ขา SDA (รับและส่งข้อมูล) และ SCL (ขาสัญญาณนาฬิกา) โดยขาสัญญาณทั้งสองจะต้องต่อตัวต้านทานพูลอัปไว้เพื่อกำหนดสถานะลอจิก 1 ให้กับระบบบัส ซึ่งการใช้งานจะตั้งไว้บริเวณคอนโซลหน้ารถ เมื่อรถเคลื่อนที่องศาหน้ารถที่ได้จะมีความถูกต้องแม่นยำกว่านำไปติดตั้งบริเวณอื่น รูปที่ 3.13 แสดงการเชื่อมต่อระหว่างชุดเข็มทิศอิเล็กทรอนิกส์กับบอร์ดควบคุม



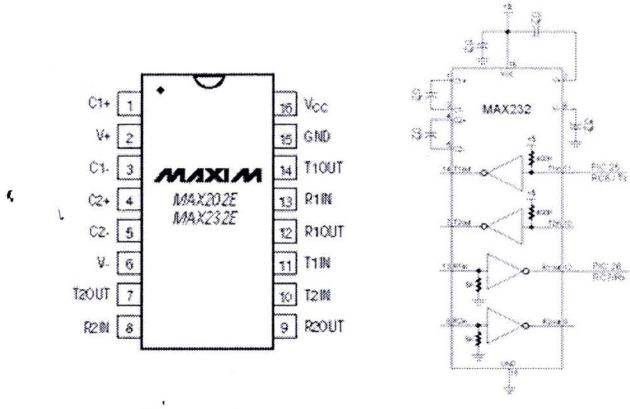
รูปที่ 3.13 Digital Compass Module เชื่อมต่อกับไมโครคอนโทรลเลอร์

**การติดตั้งอุปกรณ์ GPS**

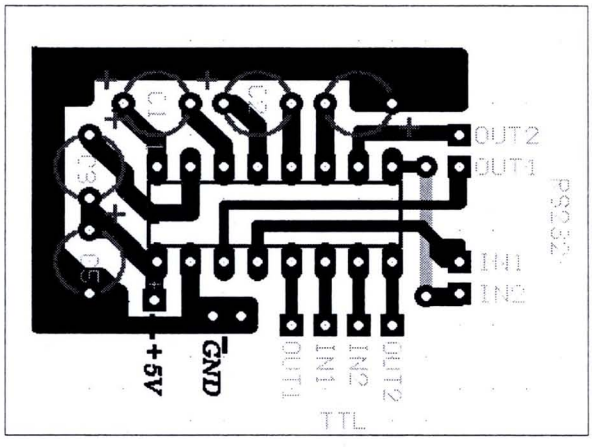
ชุดรับ-ส่ง สัญญาณ GPS Module GM-82 เป็นส่วนรับข้อมูลจากดาวเทียม 4-12 ดวง ทำหน้าที่รับข้อมูลที่ใช้ในการระบุตำแหน่งในการนำข้อมูลไปใช้นั้น ต้องทำการเชื่อมต่อสัญญาณจากขา TX (ขาที่ 11) และทำการป้อนแรงดันไฟฟ้า 5 Volt (ขาที่ 2) และกราว์(ขา ที่ 10) ซึ่งโดยปรกติ GM–82 จะส่งข้อมูลเป็นแบบดิจิตอล คือ ลอจิก '0' เป็น 0 Volt และ ลอจิก '1' เป็น 5 Volt (TTL - transistor-to-transistor logic) เมื่อนำไปต่อสัญญาณเข้าทางพอร์ตอนุกรม จำเป็นต้องมี IC สำหรับแปลงระดับแรงดันโดยใช้ IC Max232 สำหรับการแปลงระดับสัญญาณให้มีค่า 3 ถึง 15 Volt ให้ลอจิก '1' และ 0 ถึง –15 Volt ให้ลอจิก '0' รูปที่ 3.14 แสดงบอร์ด GPS GM-82 รูปที่ 3.15 – 3.17 แสดงวงจรการเชื่อมต่อภายในของ MAX232 และวงจรที่ใช้ต่อกับบอร์ดไมโครคอนโทรลเลอร์



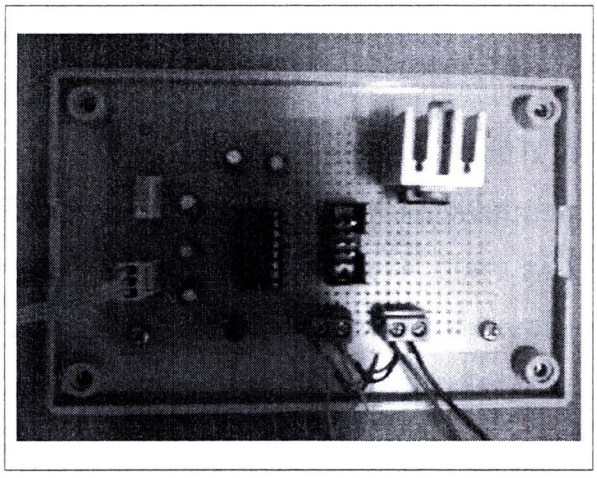

รูปที่ 3.14 บอร์ด GPS GM-82




รูปที่ 3.15 IC และวงจรภายในของ Max-232



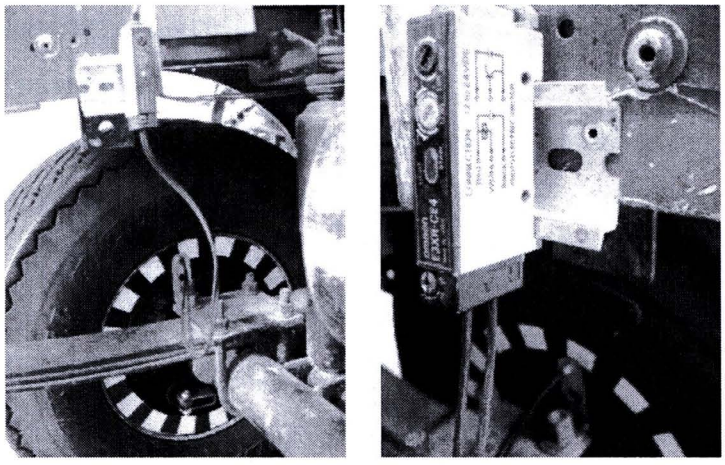

รูปที่ 3.16 วงจรการต่อใช้งาน MAX-232



รูปที่ 3.17 บอร์ด MAX-232 ที่พัฒนาพร้อมใช้งาน

**การติดตั้งอุปกรณ์ Wheel Encoder**

ในการติดตั้งจะมีอยู่ด้วยกันสองตัว คือ จานสะท้อนแสง จะทำการติดตั้งไว้บริเวณกะทะล้อของล้อหลังทั้งสองล้อวัสดุที่ใช้คือ แผ่นฟิวเจอร์บอร์ดสีดำติดสติกเกอร์สีขาวที่ตัดเป็นชิ้น 16 ชิ้น อุปกรณ์อีกชิ้นหนึ่งคือ Fiber Optic Sensor จะทำการติดตั้งไว้บริเวณใกล้ๆ กับจานสะท้อนแสงโดยให้แนวการสะท้อนแสงทำมุมตั้งฉากกับจานสะท้อนแสง ดังแสดงในรูปที่ 3.18



รูปที่ 3.18 ตำแหน่งที่ติดตั้งอุปกรณ์ Wheel Encoder

### 3.2.3 ระบบการควบคุมรถแบบบังคับด้วยแท่นควบคุม

**การสั่งควบคุมรถด้วยแท่นควบคุมระยะใกล้**

ผู้พัฒนาได้ออกแบบแท่นควบคุมแบบใช้สายโดยจะต่อเข้ากับไมโครคอนโทรเลอร์ AVR ที่ PORT PA.1 และ PA.2 บนแท่นควบคุมจะประกอบไปด้วยสวิตซ์แบบกด เมื่อมีการกดสวิตซ์วงจรจะปล่อยสัญญาณลอจิก "1" ให้กับไมโครคอนโทรลเลอร์ที่ PORT PA.1 ซึ่งในส่วนนี้จะเป็นค่า Input ให้กับโปรแกรมที่พัฒนาไว้ในระบบไมโครคอนโทรลเลอร์ และเมื่อมีการกดสวิตซ์ไมโครคอนโทรลเลอร์จะส่งสัญญาณลอจิก "1" ให้กับขา In0 ของไดร์เวอร์บอร์ดของมอเตอร์ เพื่อสั่งงานมอเตอร์ที่ได้ทำการติดตั้งไว้ สำหรับขับเฟืองบริเวณแกนพวงมาลัย ให้เกิดการหมุนของแกนบังคับเลี้ยว ซึ่งแกนบังคับเลี้ยวจะส่งแรงไปยังก้านบังคับล้อทำให้รถเกิดการเลี้ยวไปในทิศทางซ้าย ซึ่งในการบังคับเลี้ยวใน ทิศทางขวาก็ใช้ลักษณะเดียวกันแตกต่างกันที่ AVR ไมโครคอนโทรลเลอร์จะส่งสัญญาณลอจิก "1" ให้กับขา In1 ของไดร์เวอร์บอร์ดแทน

**การควบคุมรถด้วยแท่นควบคุมระยะไกล**

ใช้สัญญาณวิทยุบังคับระบบ RF (Radio Frequency) ย่านความถี่ 2.4 GHz แบบเดียวกับที่ใช้ในรถบังคับหรือเครื่องบินบังคับวิทยุ เพราะสามารถหาซื้อและใช้งานได้สะดวกซึ่งจะมีอยู่ด้วยกัน 2 ส่วน คือ

1. **ภาคส่งสัญญาณ TX (TRANSMITTER)**

   ภาคส่งสัญญาณวิทยุ RF เครื่องส่งจะถูกติดตั้ง อยู่บนแท่นควบคุมแบบไร้สาย เป็นตัวส่งสัญญาณขนาด 2-Chanel สามารถส่งคำสั่งควบคุมได้พร้อมกันสูงสุด 2 คำสั่ง

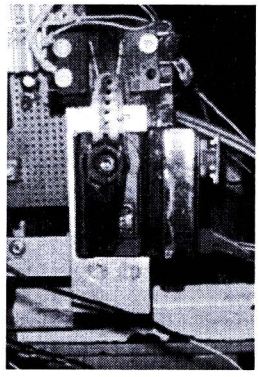
2. **ภาครับสัญญาณ RX (RECIEVER)**

   ภาครับสัญญาณ จะรอรับสัญญาณจากตัว ส่งสัญญาณที่ติดตั้งอยู่บนแท่นควบคุมไร้สาย เมื่อมีการส่งสัญญาณควบคุมออกมาตัวรับสัญญาณจะทำการแปลงสัญญาณที่รับเข้ามาแล้วส่งสัญญาณควบคุมในรูปแบบของความถี่ (Hz) ไปควบคุม Servo motor ให้ทำงานตามความต้องการโดยสั่งให้ Servo motor ทำงานมากน้อยแค่ไหนขึ้นอยู่กับผู้บังคับที่ป้อนสัญญาณควบคุมให้กับตัวส่งสัญญาณ

**การสั่งงานระบบไมโครคอนโทรลเลอร์ผ่านสัญญาณวิทยุ**

การส่งสัญญาณในลักษณะของคลื่นวิทยุ RF ไปที่ระบบไมโครคอนโทรลเลอร์ AVR ไม่สามารถทำได้ เพราะ AVR ไม่สามารถรับค่า Input ที่เป็นคลื่นความถี่ (Hz) ได้ ดังนั้น จึงใช้แขนของ Servo motor มาเป็นตัวกระตุ้นสัญญาณการควบคุมการเลี้ยว ลักษณะเดียวกันกับการกดสวิตซ์บนแท่นบังคับ ขั้นตอนการออกแบบวงจรจะทำการติดตั้ง Limit-switch ไว้ใกล้กับแขนของ Servo motor แล้วใช้การหมุนของแขน Servo Motor ไปกด Limit-switch เพื่อใช้เป็นสัญญาณ Input ให้กับ

ไมโครคอนโทรลเลอร์ ภาพที่แสดงด่านล่างคือตำแหน่งการวางอุปกรณ์ Limit-switch กับแขน Servo Motor



รูปที่ 3.19 ตำแหน่งการวางอุปกรณ์ Limit-switch เพื่อบังคับการเลี้ยว

### 3.2.4 การควบคุมรถโดยใช้ไมโครคอนโทรลเลอร์

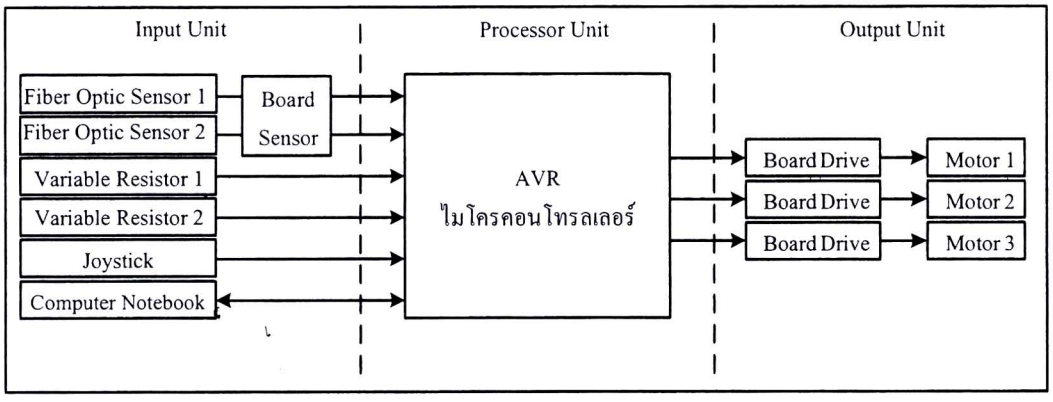
การออกแบบระบบการเคลื่อนที่ของรถจากเดิมในส่วนของระบบขับเคลื่อน ระบบบังคับเลี้ยว ระบบเบรค และอื่นๆ จะใช้การออกแรงจากมนุษย์ เพื่อสั่งให้รถเคลื่อนที่ หรือหยุด แต่ในระบบของรถอัจฉริยะจะใช้ระบบไฟฟ้าเข้ามาแทนที่ระบบการสั่งงานของมนุษย์ทั้งหมด การสั่งให้รถเคลื่อนที่ไปข้างหน้าจากเดิม มนุษย์จะใช้เท้าเหยียบไปที่คันเร่งของรถกอล์ฟ แต่เมื่อเปลี่ยนมาเป็นระบบอัตโนมัติการเหยียบคันเร่งจะถูกสั่งงานผ่านไมโครคอนโทรลเลอร์ โดยใช้การหมุนของอุปกรณ์มอเตอร์ดึงคันเร่งให้เสมือนกับการเหยียบด้วยเท้าของมนุษย์ หรืออาจกล่าวได้ว่า ใช้อุปกรณ์มอเตอร์ และอุปกรณ์ตรวจจับต่างๆ มาทำหน้าที่แทนการรับรู้ และการสั่งงานโดยการเลียนแบบพฤติกรรมการสั่งงานของมนุษย์

**การวิเคราะห์และการออกแบบระบบควบคุม**

จากการวิเคราะห์และออกแบบผู้พัฒนาได้เลือกใช้อุปกรณ์ไมโครคอนโทรลเลอร์ตระกูล Atmel AVR-128 เป็นตัวกลางการรับส่งค่าข้อมูลที่รับมาจากอุปกรณ์ตรวจจับต่างๆ และควบคุมการทำงานของอุปกรณ์มอเตอร์ เนื่องจากบอร์ดไมโครคอนโทรลเลอร์รุ่นนี้มีพอร์ตให้เลือกใช้งานอย่างหลากหลาย มีความนิยมจากผู้ใช้ในการนำมาพัฒนาเป็นอุปกรณ์ที่ใช้ในงานระบบควบคุมต่างๆ สามารถค้นหาข้อมูลเพื่อใช้เป็นแนวทางในการพัฒนาให้กับโครงงานวิจัยระบบรถอัจฉริยะทั้งในด้านซอฟต์แวร์ และฮาร์ดแวร์ ได้เป็นอย่างดี

**การออกแบบระบบฮาร์ดแวร์**

ในการออกแบบระบบฮาร์ดแวร์ที่ใช้ในการควบคุมการทำงานของระบบรถอัจฉริยะ ผู้พัฒนาได้พิจารณาเงื่อนไขลักษณะการทำงานของรถ ตัวอุปกรณ์ตรวจจับ และอุปกรณ์ใช้งานต่างๆ ที่จำเป็นในระบบ แล้วทำการออกแบบโครงสร้างความสัมพันธ์ของระบบฮาร์ดแวร์ได้ ดังแสดงในรูปที่ 3.20




รูปที่ 3.20 โครงสร้างความสัมพันธ์ของระบบฮาร์ดแวร์

จากรูปที่ 3.20 ระบบฮาร์ดแวร์จะมีหน่วยการทำงานอยู่ 3 หน่วยหลัก คือ

**1. Input Unit**

เป็นหน่วยที่จะส่งค่าข้อมูลให้ Processor Unit โดยหน่วยนี้จะประกอบไปด้วยอุปกรณ์ต่างๆ ที่ต้องการนำค่าข้อมูลนั้นๆ มาใช้ในการประมวลผลจากไดอะแกรม ในส่วนนี้จะประกอบไปด้วยอุปกรณ์ตรวจจับ 2 ชนิดคือ

- Fiber Optic Sensor นำมาใช้ในระบบ Wheel Encoder ล้อหลังซ้ายและขวา ซึ่งการเชื่อมต่อเข้ากับไมโครคอนโทรลเลอร์ต้องผ่าน Sensor Board ก่อนเพื่อปรับสัญญาณให้มีความเสถียรและระดับแรงดันให้เหมาะสม
- Variable Resistor นำมาใช้ในระบบวิเคราะห์มุมการเลี้ยว และระบบการเร่งความเร็วของรถ
- Joystick เป็นอุปกรณ์ที่ใช้ควบคุมการเคลื่อนที่ของรถโดยมนุษย์ควบคุม
- Computer Notebook เป็นหน่วยส่งคำสั่งควบคุมให้ไมโครคอนโทรลเลอร์สั่งงานอุปกรณ์ต่างๆ ตามที่ระบบตัดสินใจกลางต้องการ

**2. Processor Unit**

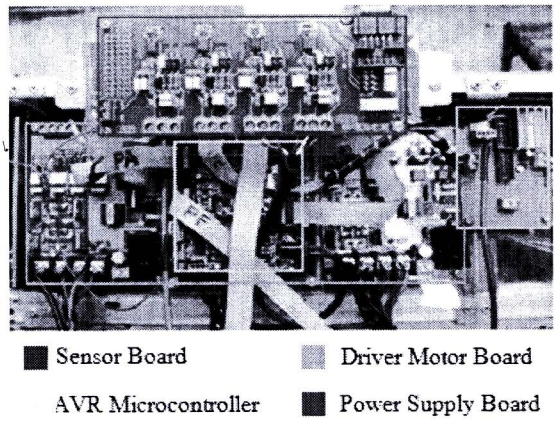
เป็นหน่วยประมวลผลโดยจะรับค่าจาก Input Unit เข้ามาวิเคราะห์แล้วสั่งงานไปยัง Output Unit ให้ทำงานตามที่ได้โปรแกรมไว้

**3. Output Unit**

เป็นหน่วยแสดงผลจากการประมวลผลของ Processor Unit อุปกรณ์ ในส่วนนี้จะประกอบไปด้วย Driver Board สามบอร์ดเพื่อใช้ในการสั่งงานมอเตอร์ที่ติดตั้งไว้บนตัวรถเพื่อทำงานแทนการสั่งงานของมนุษย์ไปยังรถ 3 ลักษณะคือ การบังคับคันเร่ง, การบังคับเบรค และการบังคับเลี้ยว

**การสร้างอุปกรณ์ฮาร์ดแวร์ และการเชื่อมต่อ**

หลังจากทำการออกแบบและทำการสร้างอุปกรณ์ ฮาร์ดแวร์ต่างๆ ที่จำเป็นกับระบบของงานวิจัยนี้แล้ว จากนั้นทำการติดตั้งฮาร์ดแวร์ต่างๆ ลงบนรถกอล์ฟ และเชื่อมต่ออุปกรณ์ต่างๆ เข้าด้วยกันตามโครงสร้างความสัมพันธ์ของระบบฮาร์ดแวร์ที่ได้ทำการออกแบบ ดังรูปที่ 3.21




รูปที่ 3.21 บอร์ดอิเล็กทรอนิกส์สำหรับการควบคุมและรูปแบบการเชื่อมต่อเข้าด้วยกัน

**3.2.5 เขียนโปรแกรมควบคุมระบบฮาร์ดแวร์**

เพื่อให้ระบบฮาร์ดแวร์ที่ได้ทำการออกแบบสามารถทำงานได้ ดังนั้นจึงต้องพัฒนาโปรแกรมขึ้นมารองรับการทำงานในส่วนนี้ ซึ่งโปรแกรมที่พัฒนาในส่วนของระบบควบคุมจะเป็นภาษา C โดยสามารถจำแนกการพัฒนาได้ 2 แบบดังนี้

**1. เขียนโปรแกรมควบคุมอุปกรณ์ ด้วยไมโครคอนโทรลเลอร์**

หลักการในการสั่งควบคุมอุปกรณ์ต่างๆ จะเป็นการกำหนดเงื่อนไขอย่างใดอย่างหนึ่งแล้วสั่งการไปยังตัวฮาร์ดแวร์ ผ่านพอร์ตต่างๆ ที่ทำเชื่อมต่อกับอุปกรณ์นั้นๆ โดยเมื่อสั่งลอจิกเป็น “1” (+5 Volt) อุปกรณ์ที่ต่ออยู่ก็จะทำงาน หากไม่ต้องการให้ทำงานก็จะสั่งให้พอร์ตนั้นเป็นลอจิก “0” (0 Volt)

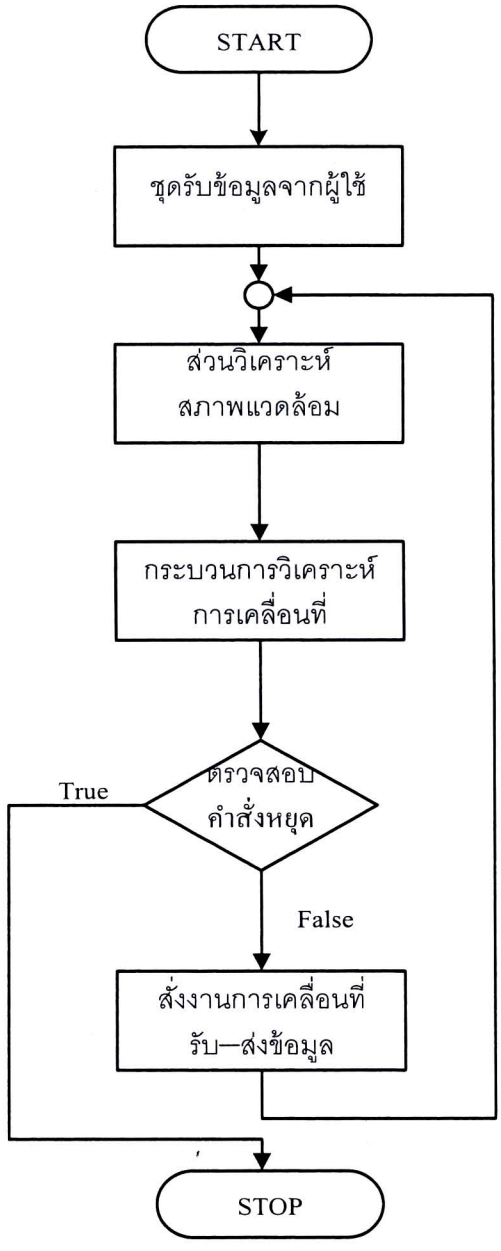
**2. เขียนโปรแกรมรับ-ส่งคำสั่งจากระบบตัดสินใจกลาง**

พัฒนาโปรแกรมให้สามารถทำงานร่วมกับระบบตัดสินใจกลาง โดยระบบตัดสินใจกลางจะส่งคำสั่งควบคุมต่างๆ มาให้ ดังนั้นการเขียนโปรแกรมจะเป็นการรับค่าจากพอร์ตอนุกรมมาทำการแปลความหมาย แล้วจึงสั่งคำสั่งควบคุมที่ระบบตัดสินใจกลางส่งมา ไปยังอุปกรณ์ที่ทำการติดตั้งให้ทำงานตามต้องการ

### 3.3 ระบบตัดสินใจกลาง

ส่วนของระบบตัดสินใจกลางจะเป็นซอฟต์แวร์ที่ทำงานอยู่บนเครื่องคอมพิวเตอร์ทำหน้าที่เป็นศูนย์กลางการควบคุม และการเชื่อมต่อของระบบอัตโนมัติต่างๆ เข้าด้วยกัน โดยมีระบบระบุขอบทาง ระบบระบุตำแหน่ง ระบบแผนที่ ระบบระบุสิ่งกีดขวางและวิเคราะห์สัญญาณป้ายจราจร และระบบคอนโทรลเลอร์ [13]

### 3.3.1 การออกแบบการทำงานของระบบตัดสินใจกลาง



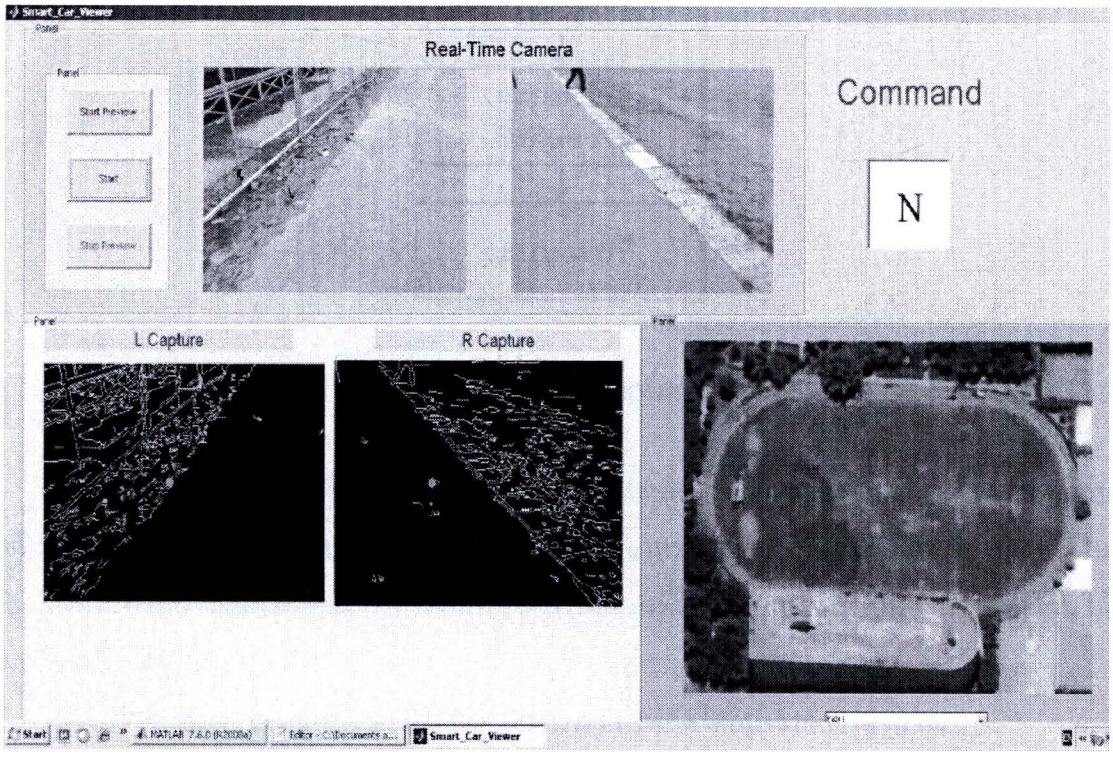

สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

รูปที่ 3.22 ระบบการดำเนินงานโดยรวม

จากรูปที่ 3.22 เป็นระบบการดำเนินงานโดยรวม ซึ่งจะพบว่าการทำงานนี้สามารถแบ่งได้เป็น 4 ส่วนหลักดังนี้

**1. ส่วนรับข้อมูลจากผู้ใช้**

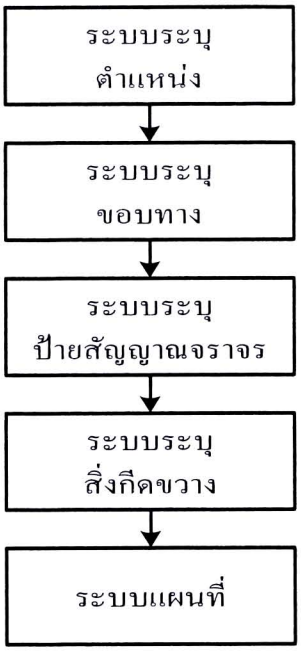
ผู้ใช้ต้องทำการเลือกแผนที่ให้กับตัวรถ วิธีนี้ผู้ใช้สามารถเลือกแผนที่ได้ตามต้องการ โดยไม่ต้องทำการแก้ไขในส่วนของโปรแกรม ซึ่งใช้ Popup menu เลือกสนามที่ต้องการ และเมื่อทำการเลือกสนามที่ต้องการแล้ว ก็สามารถกดปุ่มเพื่อให้ระบบเริ่มทำงานได้ ดังรูปที่ 3.23




รูปที่ 3.23 ส่วน GUI ของผู้ใช้โปรแกรม

โปรแกรมนี้จะถูกติดตั้งอยู่บนเครื่องคอมพิวเตอร์ที่อยู่บนรถ โดยหน้าต่างด้านบนของ GUI จะแสดงภาพจากกล้องทั้งด้านซ้ายและขวา โดยแสดงผลและปรับปรุงภาพต่อเนื่องตลอดเวลา ภาพที่ได้จะถูกนำมาประมวลผลและหาระยะขอบทาง ดังแสดงในหน้าต่างด้านล่างซ้าย สำหรับหน้าต่างด้านขวาล่างเป็นการแสดงแผนที่ของเส้นทางทั้งหมด

**2. ส่วนวิเคราะห์สภาพแวดล้อม**

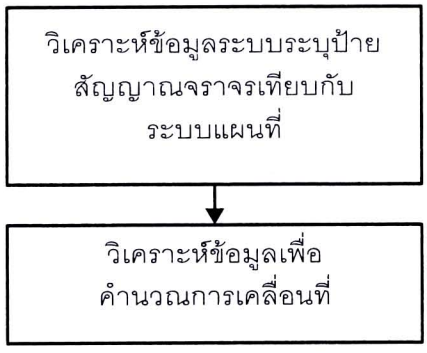



รูปที่ 3.24 ขั้นตอนการวิเคราะห์สภาพแวดล้อม

จากรูปที่ 3.24 เป็นการทำงานในขณะทดสอบการเคลื่อนที่ ในส่วนนี้ เมื่อมีการใช้งานทดสอบจริง ก็สามารถเข้าไปทำการประมวลผลจากค่าที่วัดได้จากระบบต่างๆ โดยไม่ต้องเข้าไปแก้ไขในส่วนของโปรแกรม แต่ในขณะทำการทดสอบการเคลื่อนที่แบบจำลอง ระบบก็สามารถทำงานได้เสมือนจริงโดยใช้ข้อมูล ดังนี้

- **ระบบระบุตำแหน่ง** จะมีค่าองศาหน้ารถ มีค่าพิกัดโดยมาจากการคำนวณ
- **ระบบระบุขอบทาง** จะได้มาจากการจำลองการติดตั้งกล้อง และคำนวณหาระยะขอบทางจากรถถึงขอบทางในระบบจำลอง
- **ระบบระบุป้ายสัญญาณจราจร** จะมีการตั้งค่าพิกัดและสัญญาณของป้ายจากส่วนการรับข้อมูลจากผู้ใช้ซึ่งการทำงานของระบบนี้จะทำการเตือนเหมือนกับระบบการทำงานจริงโดยจะเตือนมาเมื่อตรวจพบสัญญาณจราจรในระยะที่กำหนด
- **ระบบระบุสิ่งกีดขวาง** จะมีการตั้งค่าพิกัดของสิ่งกีดขวางจากส่วนการรับข้อมูลจากผู้ใช้ซึ่งการทำงานของระบบนี้จะทำการเตือนเหมือนกับระบบการทำงานจริงโดยจะเตือนมาเมื่อถึงตรวจพบสิ่งกีดขวางในระยะที่กำหนด
- **ระบบแผนที่** จะมีการตั้งค่าพิกัดของทางแยกต่างๆ และทิศทางที่ควรไปจากส่วนการรับข้อมูลจากผู้ใช้ซึ่งการทำงานของระบบนี้จะทำการเตือนเหมือนกับระบบการทำงานจริงโดยจะเตือนมาเมื่อตัวรถมาถึงระยะที่กำหนด

### 3. ส่วนการวิเคราะห์การเคลื่อนที่




รูปที่ 3.25 ขั้นตอนการวิเคราะห์การเคลื่อนที่

**การวิเคราะห์ข้อมูลระบบระบุป้ายสัญญาณจราจรเทียบกับระบบแผนที่**

เนื่องจากข้อมูลในการบอกทิศทางที่จะเคลื่อนที่ไปได้นั้นสามารถให้ข้อมูลมาได้จากทั้ง 2 ระบบ คือ ระบบระบุป้ายสัญญาณจราจรกับระบบแผนที่ ซึ่งในบางกรณีที่อาจเกิดขึ้นคือทั้ง 2 ระบบให้ค่าทิศทางมาไม่เหมือนกัน การวิเคราะห์นั้นก็จะดูค่าความเชื่อมั่นทั้ง 2 ระบบหากเท่ากันก็จะเชื่อระบบแผนที่ เนื่องจากระบบแผนที่นั้นได้มีข้อมูลของพิกัด ณ ขณะนั้นและยังมีข้อมูลของพื้นที่ของเส้นทางนั้นอยู่

**การวิเคราะห์ข้อมูลเพื่อคำนวณการเคลื่อนที่**

จากข้อมูลที่ได้รับจากระบบต่างๆ นั้น สามารถนำมาวิเคราะห์เพื่อที่จะกำหนดทิศทางการเคลื่อนที่ความเร็วและสัญญาณไฟท้าย ซึ่งในส่วนความเร็วการเคลื่อนที่กับทิศทางที่ได้รับทั้งจากระบบการทำงานจริง หรือการทดลองนั้นสามารถวิเคราะห์ได้โดยกระบวนการเดียวกัน เนื่องจากการวิเคราะห์นั้นได้ใช้การคำนวณโดยคิดเป็นสัดส่วนของค่าขอบทาง และสัดส่วนของทิศทางการเคลื่อนที่กับความเร็ว โดยจะมีการตั้งค่าหากกรณีที่องศาของรถหันไปทางขอบทางมากเกินไปด้วย โดยมีตัวอย่างโปรแกรมที่ใช้ในการคำนวณ ดังรูปที่ 3.26

```
turnMagnify = findMagnify( dl, dr, Magnify, rcaml );
direcA = turnMagnify * (lcaml – rcaml) / maxTurning;
speedA = abs ( ( maxSpeed – abs ( direcA) )/ turnMagnify;
turndegree = direcA;
turnsh = direcA;
```

รูปที่ 3.26 ตัวอย่างคำสั่งที่ใช้ในการคำนวณการเคลื่อนที่

turnMagnify : อัตราการเลี้ยวว่าควรเลี้ยวมากหรือน้อย

rcaml : ระยะห่างจากตัวรถถึงขอบทางทางขวาหน่วยเป็นเมตร

lcaml : ระยะห่างจากตัวรถถึงขอบทางทางซ้ายหน่วยเป็นเมตร

maxTurning : อัตราการเลี้ยวสูงสุด

maxSpeed : อัตราความเร็วสูงสุด

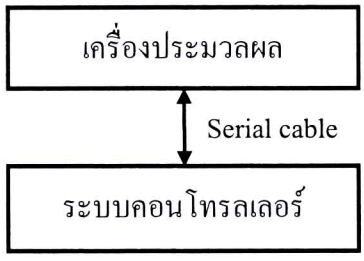
direcA : ทิศทางการเลี้ยวที่คำนวณ

speedA : ความเร็วที่คำนวณ

lcaml – rcaml : ทำให้รถวิ่งกลางถนนได้

**4. ส่วนการสั่งงานการเคลื่อนที่ และรับ-ส่งข้อมูล**

ส่วนการสั่งงานการเคลื่อนที่ของระบบที่ใช้รับ-ส่งข้อมูลเป็นการสั่งงานไปยังระบบคอนโทรลเลอร์และรับค่าการเคลื่อนที่ของตัวรถมา การเชื่อมต่อระหว่างเครื่องประมวลผลกับระบบคอนโทรลเลอร์จะทำผ่าน Serial Port




รูปที่ 3.27 ขั้นตอนการสั่งงานการเคลื่อนที่และรับ–ส่งข้อมูล

### 3.3.2 การเชื่อมต่อระบบตัดสินใจกลางกับส่วนต่างๆ

เนื่องจากระบบตัดสินใจกลางเป็นศูนย์กลางในการรับข้อมูลที่นำเข้าจากอุปกรณ์ตรวจจับและส่งคำสั่งควบคุมไปยังระบบควบคุมอีกด้วย ทำให้เกิดการเชื่อมต่อกับระบบตัดสินใจกลางในรูปแบบที่ต่างกัน ทั้งฮาร์ดแวร์และซอฟต์แวร์สามารถจำแนกได้ดังนี้

**ส่วนการรับส่ง-ข้อมูลระหว่างเครื่องประมวลผลกับไมโครคอนโทรลเลอร์**

ส่วนการรับส่งข้อมูลระหว่างเครื่องประมวลผลกับระบบไมโครคอนโทรลเลอร์จะทำผ่านพอร์ตอนุกรมหรือ Serial port หลังจากการวิเคราะห์ จะได้ทิศทางการเคลื่อนที่และความเร็วของรถแล้วจึงสั่งให้รถทำตามโดยผ่าน Serial port ไปยังคอนโทรลเลอร์ และรับข้อมูลของรถกลับมาเพื่อนำมาวิเคราะห์ในครั้งต่อไปโดยในโปรแกรม Matlab นั้นทำการเปิด Serial port ในการทำงานใช้เวลาประมาณ 7-9 วินาที จึงได้ทำการเปิด Serial port เพื่อใช้ในการรับส่งข้อมูลตั้งแต่ก่อนสั่งงานให้รถเคลื่อนที่ รูปที่ 3.28 ตัวอย่างคำสั่งที่ใช้งานเกี่ยวกับ Serial port

```
mySerial  =  serial ( 'Com4' ) ;
set (mySerial, 'InputBufferSize' , 8000) ;
fopen (mySerial) ;
myData  = [ [' SP ' speed ' , '] [ ' TU ' turn ' , '] [ ' SEN10, ;' ] ] ;
mySerial. Terminator  = ' CR ' ;
fwrite (mySerial ,  [myData  char (13) ]  ) ;
sensorData  =  fscanf (mySerial,  '%s ' ) ;
```

รูปที่ 3.28 ตัวอย่างคำสั่งการใช้งาน Serial port ด้วยโปรแกรม Matlab

mySerial : ตั้งค่า Serial port ที่ต้องการเชื่อมต่อ

8000 : เลขขนาดข้อมูลที่กำหนดไว้

myData : ตัวแปรข้อมูลที่ส่งไปยังไมโครคอนโทรลเลอร์

mySerial.Terminator : เมื่อ Serial port พบคำสั่ง Enter จะรู้ว่าเป็นการหมดชุดคำสั่ง

fwrite : ส่งคำสั่งไปยังไมโครคอนโทรลเลอร์

sensorData : รับข้อมูลจากไมโครคอนโทรลเลอร์

**รูปแบบการรับส่งข้อมูลระหว่างระบบตัดสินใจกับระบบต่างๆ**

รูปแบบการรับส่งข้อมูลของระบบตัดสินใจกับระบบต่างๆได้ถูกออกแบบไว้ เพื่อให้ระบบต่างๆ นั้นสามารถรับส่งข้อมูลได้ถูกต้อง ตรงกัน ซึ่งระหว่างระบบตัดสินใจกับระบบขอบทาง ระบบสิ่งกีดขวางและสัญญาณจราจร ระบบระบุตำแหน่ง และระบบแผนที่นั้น ค่าที่รับส่งกันจะอยู่ในรูปแบบของ parameter เนื่องจากเป็นโปรแกรมที่ใช้อยู่ในตัวประมวลผลเดียวกัน ส่วนระหว่างระบบตัดสินใจกับระบบคอนโทรลนั้นค่าที่รับส่งกันผ่านสาย Serial นั้นได้มีการออกแบบการรับส่งเป็นชุดคำสั่งซึ่งรูปแบบที่ได้ออกแบบไว้นั้นเน้นให้สามารถเรียกใช้และทำความเข้าใจได้ง่าย

**ระบบตัดสินใจ – ระบบขอบทาง**

สิ่งที่ระบบตัดสินใจต้องการจากส่วนของระบบขอบทาง คือข้อมูลของระยะความกว้างของถนนจากกล้อง และสิ่งแปลกปลอมที่ระบบขอบทางตรวจเจอ โดยรูปแบบของข้อมูลที่รับจากระบบขอบทางเป็นรูปแบบดังนี้ [Direction] Ex : (L)

- parameter ที่1 คือ ทิศทางการบังคับเลี้ยว ซึ่งมี 3 แบบคือ L R และ N หมายถึงให้เลี้ยวซ้าย เลี้ยวขวา และตรง ตามลำดับ

**ระบบตัดสินใจ – ระบบสัญญาณจราจร**

สิ่งที่ระบบตัดสินใจต้องการจากส่วนของระบบสัญญาณจราจร คือ ข้อมูลของระยะของสัญญาณจราจรที่พบ และความหมายของสัญญาณจราจร โดยรูปแบบของข้อมูลที่รับจากระบบสัญญาณจราจรเป็นรูปแบบดังนี้ [Sign, Distance_ Label] Ex : (TR, 5.2)

- parameter ที่ 1 คือ สัญญาณจราจรอะไรที่ตรวจพบค่าที่เป็นได้คือ TL, TR
- parameter ที่ 2 คือ ระยะของสัญญาณจราจรที่พบถึงตัวรถ มีหน่วยเป็นเมตร

**ระบบตัดสินใจ – ระบบสิ่งกีดขวาง**

สิ่งที่ระบบตัดสินใจต้องการจากส่วนของระบบสิ่งกีดขวาง คือข้อมูลของระยะและทิศทางของวัตถุ โดยรูปแบบของข้อมูลที่รับจากระบบสิ่งกีดขวางเป็นรูปแบบดังนี้ [dir, Distance_Obstacle] Ex : (4.5, 5.2)

- parameter ที่ 1 คือ ทิศทางของสิ่งกีดขวางหน่วยเป็นองศา
- parameter ที่ 2 คือ ระยะของสิ่งกีดขวาง มีหน่วยเป็นเมตร

**ระบบตัดสินใจ – ระบบระบุตำแหน่ง**

สิ่งที่ระบบตัดสินใจต้องการจากส่วนของระบบระบุตำแหน่ง คือค่าพิกัดของรถเป็น2 มิติ ตามแนวแกน x, y โดยพิกัด (0,0) คือมุมซ้ายล่างของแผนที่ และข้อมูลของทิศทางของรถ โดยรูปแบบของข้อมูลที่รับจากระบบระบุตำแหน่งเป็นรูปแบบ ดังนี้ Ex : (150 200 30)

- parameter ที่ 1 คือ ตำแหน่งของตัวรถในแนวแกน x
- parameter ที่ 2 คือ ตำแหน่งของตัวรถในแนวแกน y
- parameter ที่ 3 คือ ทิศทางของตัวรถโดยมีค่าเป็น -180 ถึง 180 โดยที่ 0 คือ ทิศเหนือแล้วนับมาทางทวนเข็มนาฬิกา

**ระบบตัดสินใจ – ระบบแผนที่**

สิ่งที่ระบบตัดสินใจติดต่อกับระบบแผนที่แบ่งเป็น 2 ส่วนคือส่วนของการส่งข้อมูล และส่วนของการรับข้อมูล

ส่วนของการส่งข้อมูลจากระบบตัดสินใจให้ระบบแผนที่คือค่าพิกัดตัวรถในแนวแกน X Y ค่าองศาตัวรถในขณะนั้น ดังนี้ [x, y, Direction] Ex : ([400 180 0])

- parameter ที่ 1 คือ ระยะจากพิกัดของตัวรถในแนวแกน X
- parameter ที่ 2 คือ ระยะจากพิกัดของตัวรถในแนวแกน Y
- parameter ที่ 3 คือ องศาของตัวรถในขณะนั้น

ส่วนของการรับข้อมูลสิ่งที่ระบบตัดสินใจต้องการจากส่วนของระบบแผนที่ คือ ข้อมูลของระยะจากพิกัดของตัวรถถึงพิกัดที่จะทำการเลี้ยว (ทางแยก) ที่ได้เคยบันทึกพิกัดไว้แล้ว และทิศทางที่ควรจะเลี้ยว ดังนี้ [Distance, Angle] Ex : (250 30)

- parameter ที่ 1 คือ ระยะจาก่พิกัดของตัวรถถึงพิกัดที่จะทำการเลี้ยว มีหน่วยเป็นเซนติเมตร
- parameter ที่ 2 คือ ทิศทางที่จะเลี้ยวโดยที่ 0 คือทิศเหนือแล้วนับมาทางทวนเข็มนาฬิกา

**ระบบตัดสินใจ – ระบบคอนโทรลเลอร์**

สิ่งที่ระบบตัดสินใจติดต่อกับระบบคอนโทรลเลอร์แบ่งเป็น 3 รูปแบบ คือ ส่วนของการสั่งงานของระบบตัดสินใจไปยังระบบคอนโทรลเลอร์ ส่วนของการร้องขอข้อมูลจากระบบตัดสินใจโดยค่าที่ขอเป็นค่าของระบบเซ็นเซอร์ต่างๆ และส่วนที่ระบบคอนโทรลเลอร์ส่งมาให้ระบบตัดสินใจ ซึ่งมองคำสั่งเป็นชุดๆ แต่ละชุดจะขั้นด้วยเครื่องหมายจุลภาค (,) ซึ่งแต่ละชุดคำสั่งเป็นดังนี้

- ส่วนการสั่งงานจากระบบตัดสินใจไปยังระบบคอนโทรลเลอร์

ชุดคำสั่งการเคลื่อนที่ S ตามด้วยความเร็ว (-30 ถึง 30 ค่าลบคือการถอยหลัง) ตามด้วยอัตราเร่ง (H,L ซึ่ง H คือทันทีทันใด L คือค่อยเป็นค่อยไป) Ex : S20L

ชุดคำสั่งการเลี้ยว T ตามด้วยองศาการเลี้ยว (-45 ถึง 45 โดยที่ค่าลบคือการเลี้ยวไปทางขวา) Ex : T30

ชุดคำสั่งไฟเลี้ยว Lหรือ R เป็นตัวกำหนดข้างของไฟเลี้ยว (L ข้างซ้าย R ข้างขวา) ตามด้วย 0 หรือ 1 (0 คือสั่งให้ไฟดับ, 1 คือ สั่งให้ไฟติด) Ex : L1

ชุดคำสั่งเสียง HORN ตามด้วยค่าเสียงที่ 0 – 5 (0 คือปิดเสียง 1-5 เป็นเสียงต่างๆ) Ex : HORN3

- ส่วนของการร้องขอข้อมูลจากระบบตัดสินใจ

ชุดคำสั่งร้องขอข้อมูลเซนเซอร์ SE ตามด้วยอัตราการส่งต่อวินาทีมีค่า 1- 9 ซึ่งค่าสำหรับการสั่งให้หยุดการส่งจะเป็น 0 และค่าการร้องขอข้อมูลครั้งเดียวคือ n Ex : SE0, SEn, SE4

- ส่วนที่ระบบคอนโทรลเลอร์ส่งมาให้ระบบตัดสินใจ

ชุดข้อมูลของ Wheel encode ของล้อด้านซ้าย wl ตามด้วยจำนวนระยะที่นับได้ Ex : wl400

ชุดข้อมูลของ Wheel encode ของล้อด้านขวา wl ตามด้วยจำนวนระยะที่นับได้ Ex : wr300

ชุดข้อมูลของทิศทางของหน้ารถ t ตามด้วยองศา (-45 ถึง 45) Ex : t30

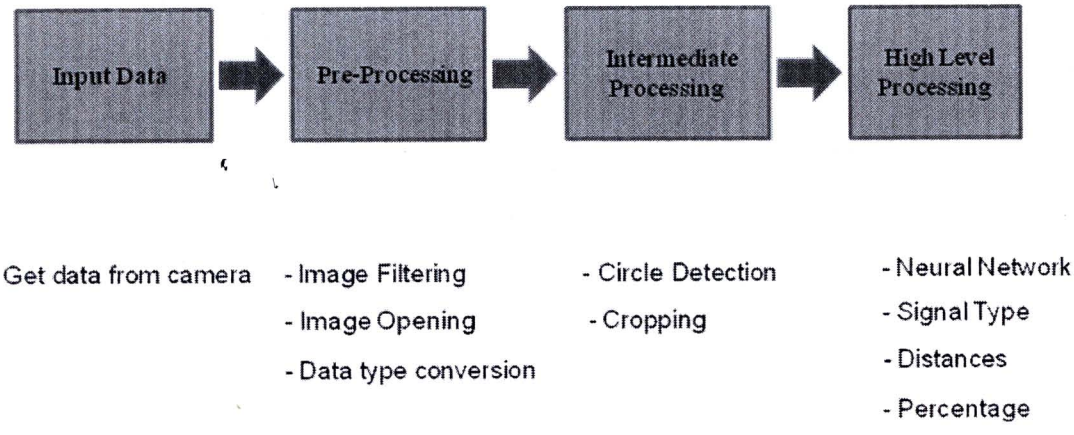
ชุดข้อมูลของทิศทางของตัวรถ d ตามด้วยองศา (0 ถึง 360) Ex : d90

ชุดข้อมูลของรอบเวลาการเก็บข้อมูล ti ตามด้วยรอบเวลา (เป็น1/1000วินาที) Ex : ti800

### 3.4 การวิเคราะห์สัญญาณภาพ

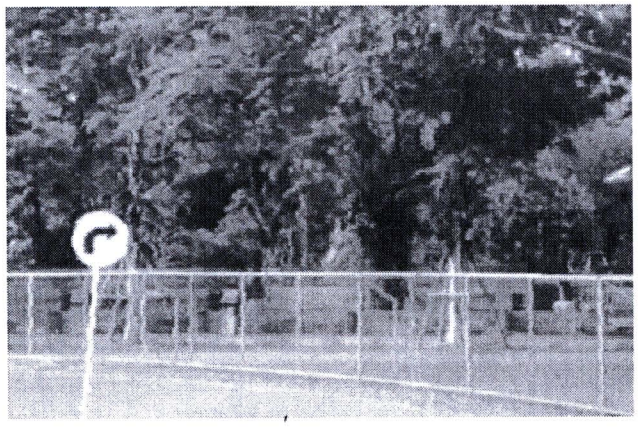
ในส่วนนี้เป็นการวางแผนการทำงานของระบบการวิเคราะห์ภาพเพื่อส่งค่าการวิเคราะห์ภาพให้กับระบบตัดสินใจกลางที่ทำการควบคุมรถอัตโนมัติให้สามารถเคลื่อนที่ไปบนถนน ซึ่งการวิเคราะห์ในส่วนนี้จะเป็นขั้นตอนการทำงานของระบบวิเคราะห์ป้ายจราจร และระบบระบุขอบทาง ซึ่งแต่ละระบบจะอธิบายถึงขั้นตอนการวิเคราะห์โดยการใช้แผนภาพบล็อกไดอะแกรม

#### 3.4.1 ระบบวิเคราะห์ป้ายจราจร




รูปที่ 3.29 ขั้นตอนการวิเคราะห์ของระบบป้ายจราจร

- **Input Data**

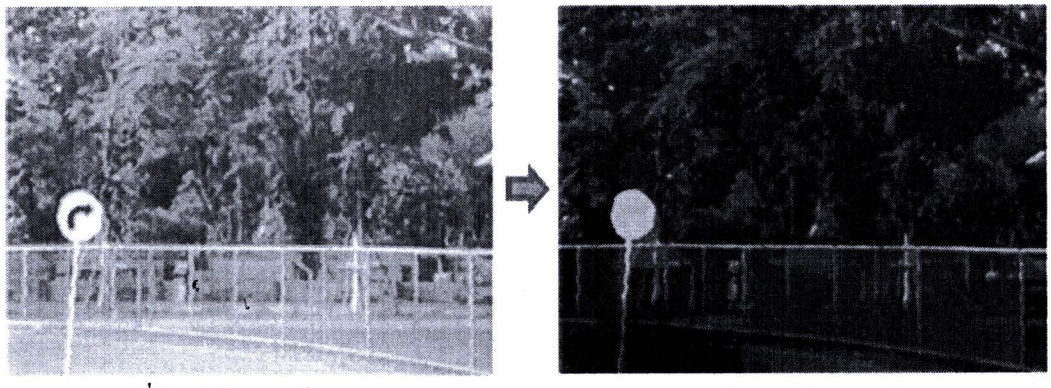


รูปที่ 3.30 ภาพป้ายจราจรที่ได้จากกล้องดิจิตอลชนิดป้ายเลี้ยวขวา

จากรูปที่ 3.30 เป็นภาพที่ได้จากกล้องดิจิตอล โดยลักษณะของป้ายจราจรมีเพียงแค่สีขาวและสีดำ โดยสีขาวนั้นจะเป็นพื้นของป้ายส่วนสีดำนั้นจะเป็นส่วนของสัญลักษณ์ที่อยู่ในป้าย ซึ่งในที่นี้ลักษณะของสัญลักษณ์ในป้ายจะเป็นลูกศรเลี้ยวขวา ความละเอียดของภาพนั้นจะไม่ละเอียดเท่าภาพที่ถ่ายจากกล้อง แต่จะขึ้นอยู่กับคุณภาพของอุปกรณ์ที่ใช้ในการแปลงสัญญาณภาพ (TV Box)

- **Pre-Processing**

**ขั้นที่ 1 Image Filtering (imfill)**

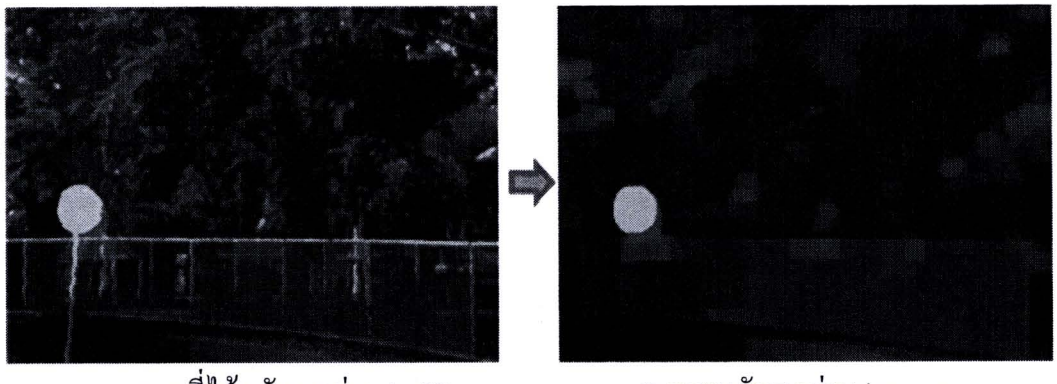


ก.ภาพที่ถ่ายได้จากกล้อง ข.ภาพหลังจากผ่านเทคนิค imfill

รูปที่ 3.31 ภาพป้ายเลี้ยวขวาที่ผ่านการใช้เทคนิคการเติมภาพ (imfill)

ในการประมวลผลขั้นแรกนี้ จะเริ่มต้นโดยการใช้เทคนิคการแปลงให้เป็นภาพสีเทา และการกรองภาพ Image Filtering เพื่อให้เห็นสัญลักษณ์วงกลมชัดเจนซึ่งเหมาะสมในการประมวลผลขั้นต่อไป โดยฟังก์ชั่นที่ใช้คือ imfill

**ขั้นที่ 2 imopen**



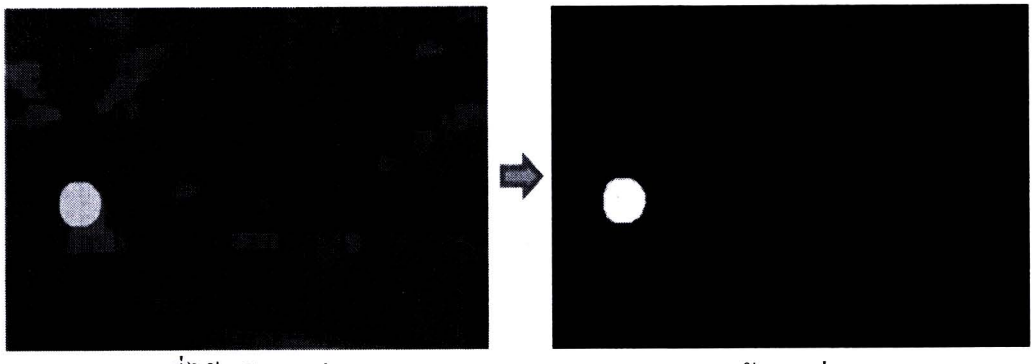
ก.ภาพที่ได้หลังจากผ่าน imfill ข.ภาพหลังจากผ่าน imopen

รูปที่ 3.32 ภาพป้ายเลี้ยวขวาที่ผ่านการใช้เทคนิค Opening

รูปที่ 3.32 เป็นการใช้เทคนิคการเปิดภาพ Opening ซึ่งเป็นเทคนิคหนึ่งที่อยู่ในส่วนของการประมวลผลโครงสร้างของภาพ (Morphological Image Processing) เป็นการทำให้ผิวขอบของวัตถุหรือป้ายจราจรมีความโค้งมนราบเรียบ โดยการตัดหรือทำลายส่วนที่เป็นคอคอด และส่วนที่โผล่

ยื่นออกไป เช่น ส่วนที่เป็นเสาของป้ายจราจร โดยฟังชั่นที่ใช้คือ A = imopen(I_open,SE) โดย SE จะเป็นตัวแปรของรูปแบบโครงสร้าง(Structuring Element) ที่ต้องการลบหรือทำลาย ซึ่งฟังชั่นที่ใช้คือ SE = strel('line',21,0) ซึ่งสิ่งที่ต้องลบออกจากภาพคือส่วนที่เป็นเสาของป้ายจราจรเพราะมีเสาของป้ายนั้นลักษณะเป็นเส้นตรง ดังนั้นชนิดรูปแบบโครงสร้างที่ต้องการลบจะเป็นชนิด 'line' จากนั้นก็จะนำภาพที่ได้ไปสู่การประมวลผลในขั้นต่อไป

**ขั้นที่ 3 Binary image**



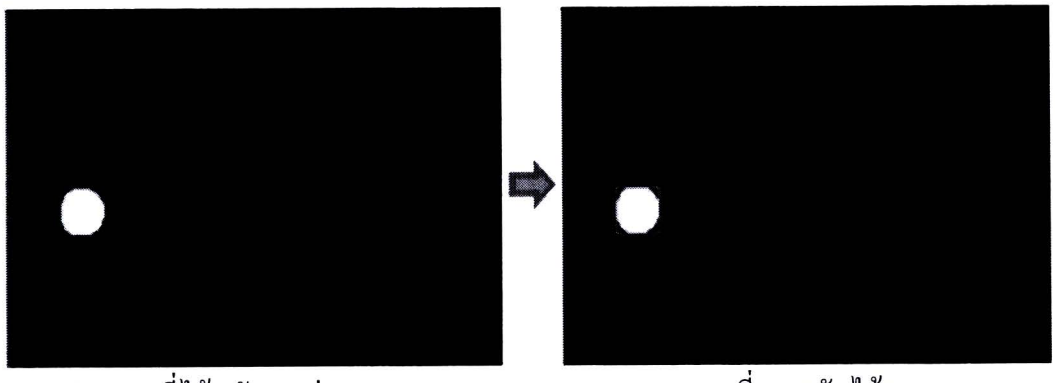
ก.ภาพที่ได้หลังจากผ่าน imopen ข.ภาพหลังจากผ่าน im2bw

รูปที่ 3.33 ภาพป้ายเลี้ยวขวาที่ผ่านการใช้เทคนิค Data Type Conversion

รูปที่ 3.33 เป็นเทคนิคในการแปลงชนิดข้อมูลภาพ Data Type Conversion คือ การทำให้ภาพเป็นชนิด Binary Image ทำให้ภาพสีเหลือเพียงแค่สองสีคือสีขาวและสีดำ โดย pixel ที่เป็นสีดำจะมีค่าสีเท่ากับ 0 และ pixel ที่เป็นสีขาวจะมีค่าสีเท่ากับ 1 ฟังชั่นที่ใช้คือ im2bw

- **Intermediate-Processing**

**ขั้นที่ 4 Circle detection**

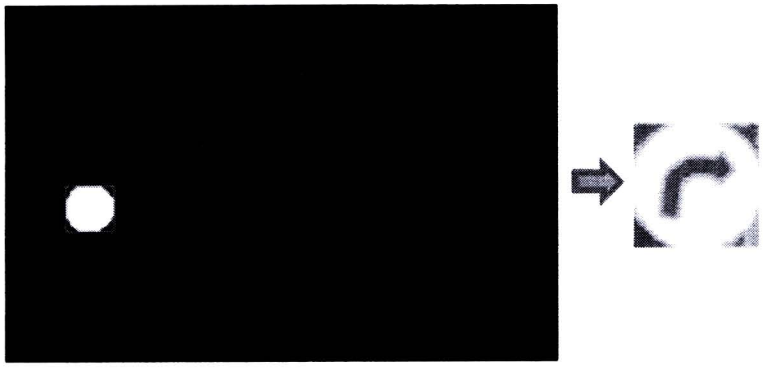


ก.ภาพที่ได้หลังจากผ่าน im2bw ข.วงกลมที่ตรวจจับได้จากภาพ

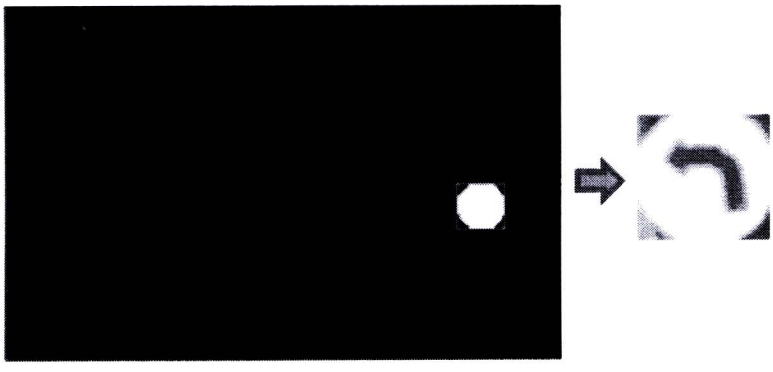
รูปที่ 3.34 ภาพป้ายเลี้ยวขวาที่ผ่านการใช้เทคนิค Circle Detection

ในส่วนนี้จะเป็นส่วนของการประมวลผลขั้นกลาง Intermediate-Processing เมื่อได้รูปภาพที่มาจากการประมวลผลขั้นต้น Pre-Processing แล้วรูปภาพที่ได้จะมีชนิดข้อมูลเป็นแบบ Binary Image ขั้นตอนต่อไปก็จะเป็นการใช้เทคนิคในการตรวจจับวงกลม Circle Detection โดยใช้ฟังชั่น regionprops ซึ่งเป็นฟังชั่นในการวัดคุณสมบัติพื้นที่ของรูปภาพ เมื่อทำการตรวจจับแล้วจะตัดพื้นที่บริเวณวงกลมนั้น Cropping นำไปประมวลผลเพื่อหาสัญลักษณ์ที่อยู่ในป้ายต่อไป

**ขั้นที่ 5 Image cropping**



รูปที่ 3.35 ภาพป้ายเลี้ยวขวาที่ผ่านการใช้เทคนิค Cropping



รูปที่ 3.36 ภาพป้ายเลี้ยวซ้ายที่ผ่านการใช้เทคนิค Cropping

รูปที่ 3.35 และ รูปที่ 3.36 เป็นการตัดแบ่งพื้นที่ใช้งาน Cropping โดยการตัดแยกเอาเฉพาะส่วนที่ได้จากการใช้เทคนิคการตรวจจับวงกลม Circle Detection ซึ่งส่วนที่เป็นวงกลมนั้นก็คือส่วนของป้ายจราจร จากนั้นทำการตัดพื้นที่ส่วนนี้ออก ฟังก์ชั่นที่ใช้คือ imcrop เพื่อที่จะนำภาพที่ได้ไปใช้ในการประมวลผลขั้นต่อไปซึ่งก็คือการประมวลผลละดับสูง High Level Processing เพื่อหาความหมายของภาพต่อไป

- **High Level Processing**

ในส่วนนี้จะเป็นการประมวลผลขั้นสุดท้ายซึ่งจะเป็นการวิเคราะห์รูปภาพป้ายจราจรที่ได้จากการ Cropping มาวิเคราะห์ว่าสัญลักษณ์ที่อยู่ในป้ายนั้นเป็นป้ายอะไร เช่น ป้ายเลี้ยวซ้าย (Turn LEFT) ป้ายเลี้ยวขวา (Turn RIGHT) ซึ่งในส่วนนี้จะนำเอาเทคนิคข่ายงานประสาท Neural Network มาใช้ในการวิเคราะห์สัญลักษณ์ที่อยู่ในป้าย โดยการเรียนรู้จากตัวอย่างที่มีอยู่ เมื่อทำการวิเคราะห์แล้วจะได้ข้อมูลของป้ายจราจรดังรูปที่ 3.37 ซึ่งก็จะมี ชนิดของป้ายจราจร(Sign) ได้ Turn RIGHT(เลี้ยวขวา) และระยะทางของป้าย(Distance_Lable) ได้ 6.1 เมตร และเวลาที่ใช้ในการประมวลผลป้ายเลี้ยวขวานี้ คือ 0.601 วินาที และดังรูปที่ 3.38 ซึ่งก็จะมี ชนิดของป้ายจราจร(Sign) ได้ Turn LEFT(เลี้ยวซ้าย) และระยะทางของป้าย(Distance_Lable) ได้ 5.2 เมตร และเวลาที่ใช้ในการประมวลผลป้ายเลี้ยวซ้ายนี้ คือ 0.720 วินาที ทั้งนี้ระยะทางของป้ายถึงตัวกล้องจะได้มาจากสัดส่วนจำนวน pixel ของสัญลักษณ์ภาพ โดยถ้าในภาพตรวจจับป้ายได้ขนาดเล็กแสดงว่าป้ายอยู่ไกล เนื่องจากขนาดจริงของป้ายได้กำหนดไว้ล่วงหน้า

```
Elapsed time is 0.601 seconds.
Sign_Traffic =

                    Sign : ' Turn RIGHT'

      Distance_Label : 6.1
```

รูปที่ 3.37 ข้อมูลป้ายเลี้ยวขวาที่ระบบตรวจจับป้ายจราจรวิเคราะห์และส่งให้ระบบตัดสินใจกลาง โดยใช้โปรแกรม MATLAB ประมวลผลจากเครื่องคอมพิวเตอร์ที่อยู่บนรถ

```
Elapsed time is 0.720 seconds.
Sign_Traffic =

                    Sign : ' Turn LEFT '

      Distance_Label : 5.2
```

รูปที่ 3.38 ข้อมูลป้ายเลี้ยวซ้ายที่ระบบตรวจจับป้ายจราจรวิเคราะห์และส่งให้ระบบตัดสินใจกลาง โดยใช้โปรแกรม MATLAB ประมวลผลจากเครื่องคอมพิวเตอร์ที่อยู่บนรถ

**ค่าที่ระบบวิเคราะห์ป้ายจราจรส่งให้ระบบตัดสินใจกลาง**

ข้อมูลที่ระบบวิเคราะห์ป้ายจราจร จะส่งไปยังระบบตัดสินใจกลางมีอยู่ด้วยกัน 2 กรณี ดังต่อไปนี้

From Sign_Traffic = [Sign, Distance_ Label] โดยที่

Sign = TR, TL (สัญลักษณ์ของป้ายจราจร)

Distance_ Label = unit in m (ระยะห่างตัวรถกับป้ายจราจร)

**กรณีที่ 1** สัญลักษณ์ป้ายจราจรเลี้ยวขวา

**ตัวอย่าง:** Sign = TR, Distance_ Label = 6.1

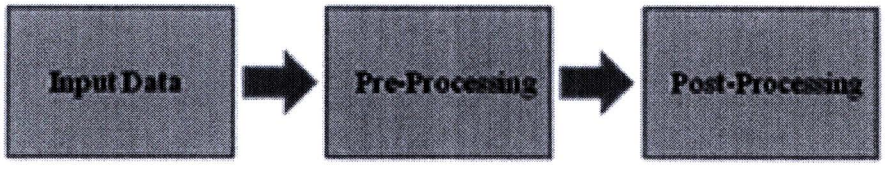
**หมายถึง** สัญลักษณ์ของป้ายจราจรเลี้ยวขวา และมีระยะห่าง 6.1 เมตร ดังรูปที่ 3.37

**กรณีที่ 2** สัญลักษณ์ป้ายจราจรเลี้ยวซ้าย

**ตัวอย่าง:** Sign = TL, Distance_ Label = 5.2

**หมายถึง** สัญลักษณ์ของป้ายจราจรเลี้ยวซ้าย และมีระยะห่าง 5.2 เมตร ดังรูปที่ 3.38

### 3.4.2 ระบบระบุขอบทาง

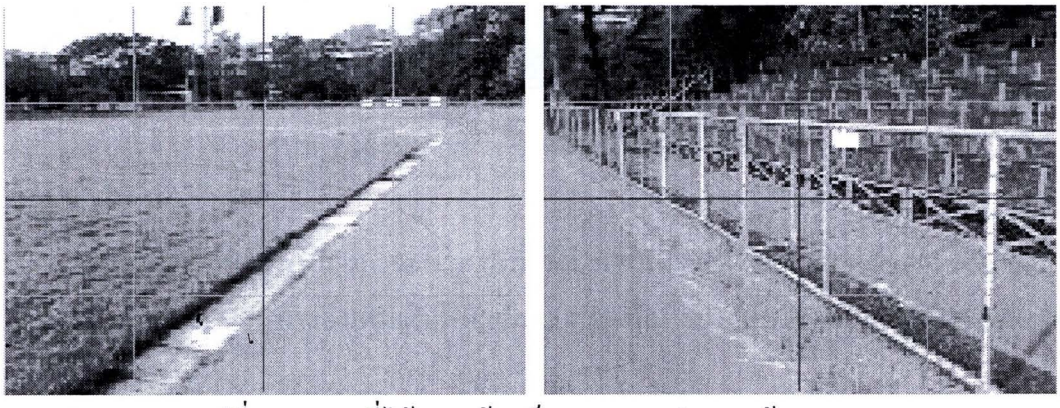



รูปที่ 3.39 ขั้นตอนของระบบระบุขอบทาง

ระบบนี้ประกอบด้วย 3 ขั้นตอนหลักๆ คือ รับภาพจากกล้องจำนวน 2 ตัวทางซ้ายและทางขวา จากนั้นส่งไปสู่กระบนการ Pre-Processing เพื่อให้ได้ขอบทาง และจะคำนวณหาระยะขอบทางในส่วนของ Post-Processing ดังแสดงในรูปที่ 3.39

- **Input Data**

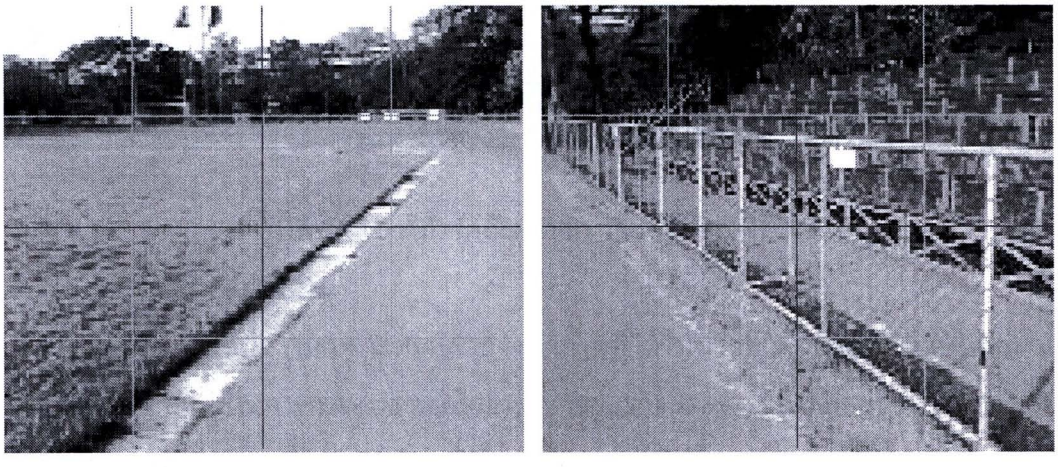
จะเป็นการทำงานของกล้องเว็บแคมเพื่อทำการจับภาพถนน และสร้างเส้นตารางขึ้นมาเพื่อให้ทราบขนาดของภาพโดยแบ่งออกเป็น 4 ส่วน โดยเส้นสีแดงจะเป็นเส้นแบ่งกึ่งกลางของภาพในแนวแกน X และแกน Y ทำให้สะดวกต่อการทดสอบ แล้วภาพที่ได้จะถูกส่งไปยังคอมพิวเตอร์เพื่อทำการประมวลผล



รูปที่ 3.40 ภาพที่ได้การกล้องเว็บแคม และทำการสร้างตาราง

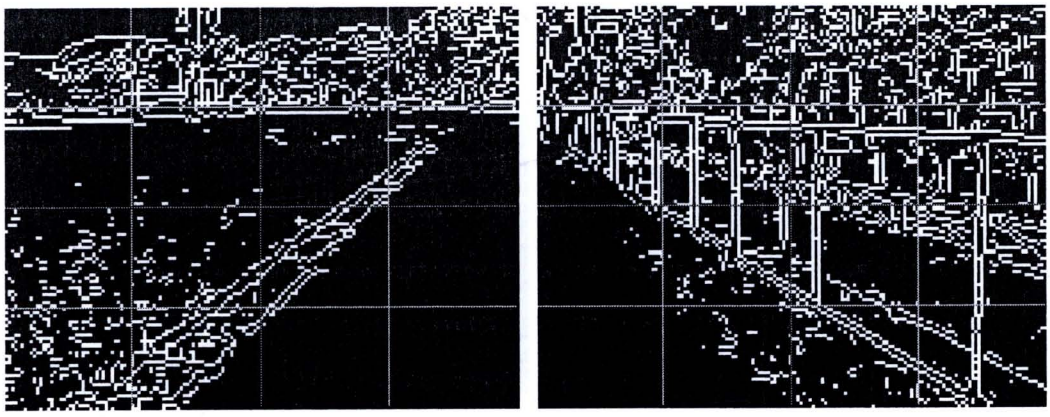
- **Pre-Processing**

การประมวลผลขั้นแรกจะเริ่มทำการแปลงภาพสีชนิด RGB ให้เป็นภาพสีชนิด Grayscale ดังรูปที่ 3.41



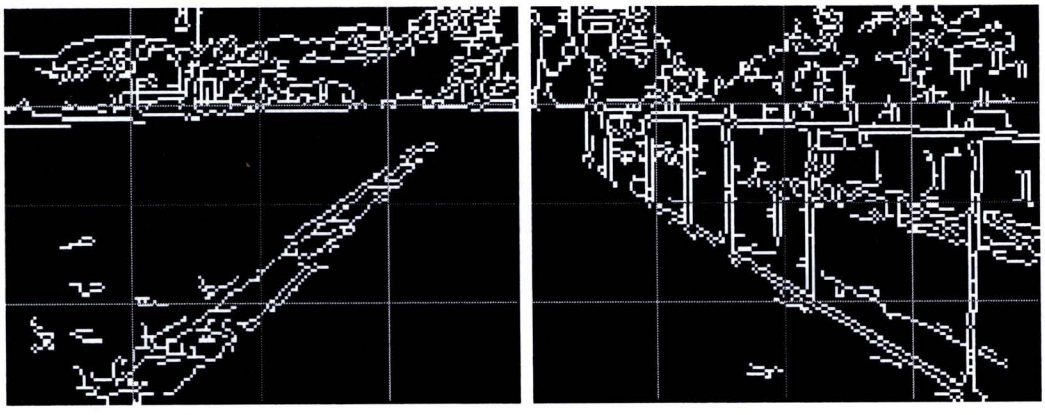
รูปที่ 3.41 การแปลงภาพสีชนิด RGB เป็นภาพสีชนิด Grayscale

การประมวลผลขั้นที่สองจะทำก่ารนำภาพชนิด Grayscale มาเข้าสู่กระบวนการหาขอบภาพ โดยใช้เทคนิค Canny (อ้างอิงเพิ่มเติมที่หัวข้อ 2.4.6) ในการทำภาพให้เป็นขอบภาพ ดังรูปที่ 3.42



รูปที่ 3.42 การหาขอบภาพโดยใช้เทคนิค Canny

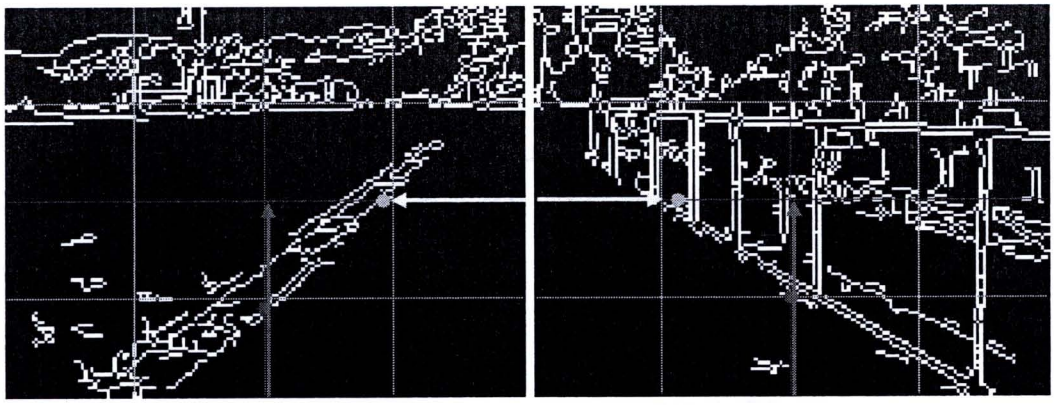
การประมวลผลขั้นที่สาม จะนำภาพที่ได้ทำการแปลงเป็นขอบภาพ ทำการลดสัญญาณรบกวนหรือกำจัด Noise (อ้างอิงเพิ่มเติมที่หัวข้อ 2.4.5) เพื่อที่จะได้เห็นขอบภาพที่ชัดเจน ดังรูปที่ 3.43



รูปที่ 3.43 การกำจัดสัญญาณรบกวนหรือกำจัด Noise

- **Post-Processing**

เมื่อได้ภาพที่มีความเหมาะสมแล้วก็จะนำภาพที่ได้ไปประมวลผลขั้นต่อไปดังแสดงในรูปที่ 3.44 โดยจะใช้เทคนิคในการหาขอบภาพ เพื่อตรวจหาขอบทาง โดยจะใช้จุดสีแดงเป็นการหาค่าขอบภาพที่ได้ในแนวแกน Y โดยจะตรวจสอบจากด้านล่างของภาพไล่ขึ้นบน และจุดสีเขียวจะเป็นการหาค่าขอบภาพที่ได้ในแนวแกน X โดยกล้องซ้ายจะตรวจสอบจากด้านขวาไปด้านซ้ายของภาพ และกล้องขวาจะตรวจสอบจากด้านซ้ายไปด้านขวาของภาพ จะได้ค่าตำแหน่งของขอบ 4 ค่า คือ ค่าที่ได้จากกล้องซ้าย 2 และกล้องขวา 2 ค่า โดยจุดสีแดงจะเป็นค่าของขอบทาง จุดสีเขียวจะเป็นค่าที่ใช้ในการตรวจสอบว่าค่าของจุดสีแดงเป็นค่าของขอบใบไม้ที่อยู่บนถนนหรือไม่ และเป็นค่าที่ช่วยบอกทิศทางการเลี้ยว จากนั้นก็จะได้ค่าของทิศทาง เช่น ค่าการเลี้ยวซ้าย (L) ค่าการเลี้ยวขวา(R) และไม่ทำการเลี้ยว(N) เมื่อทำการวิเคราะห์เสร็จก็จะทำการส่งข้อมูลไปยังระบบตัดสินใจกลางต่อไป



รูปที่ 3.44 การหาค่าของขอบทางในแกน X และแกน Y

หมายเหตุ : กรณีที่ระบบแผนและระบบระบุขอบทาง มีการบอกทิศทางการเลี้ยวแตกต่างกัน เช่น ระบบแผน์ที่บอกให้เลี้ยวซ้าย แต่ระบบระบุขอบทางบอกให้เลี้ยวขวา ซึ่งทิศทางการเลี้ยวที่แตกต่างกันนี้ ระบบตัดสินใจกลางจะทำงานตามทิศทางการเลี้ยวของระบบแผนที่ และถ้าระบบแผนที่ไม่ได้บอกทิศทางการเลี้ยว แต่ระบบระบุขอบทางบอกทิศทางเลี้ยว ระบบตัดสินใจกลางจะทำงานตามทิศทางการเลี้ยวของระบบระบุขอบทาง

**ค่าที่ระบบระบุขอบทางส่งให้ระบบตัดสินใจกลาง**

กรณีที่เกิดขึ้นในระบบระบุขอบทาง และจะมีการส่งข้อมูลไปยังระบบตัดสินใจกลางมีอยู่ด้วยกัน 3 กรณี ดังต่อไปนี้

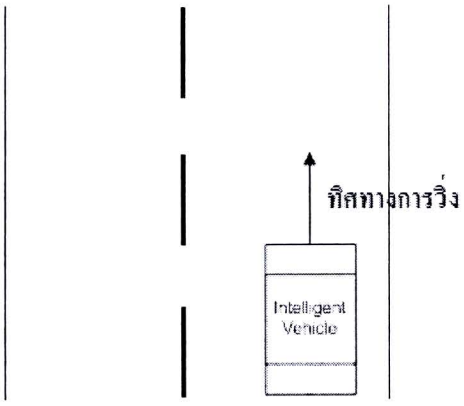
From Wayside = [Direction] โดยที่

Direction = L, R, N (ทิศทางการเลี้ยวของรถ)

**กรณีที่ 1** ตำแหน่งของตัวรถอยู่ในตำแหน่งเลี้ยวซ้าย

**ตัวอย่าง:** Direction = L

**หมายถึง** ให้ระบบตัดสินใจกลางเลี้ยวซ้าย

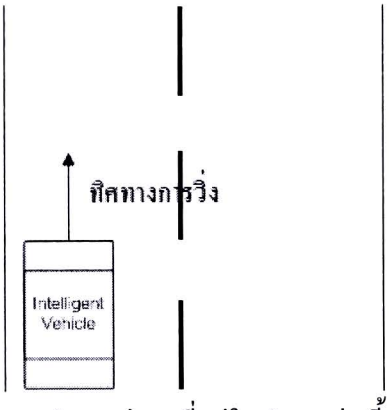



รูปที่ 3.45 ตำแหน่งรถที่อยู่ในตำแหน่งเลี้ยวซ้าย

**กรณีที่ 2** ตำแหน่งของตัวรถอยู่ในตำแหน่งเลี้ยวขวา

**ตัวอย่าง:** Direction = R

**หมายถึง** ให้ระบบตัดสินใจกลางเลี้ยวขวา

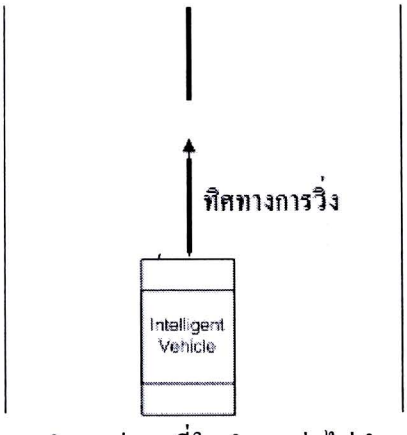



รูปที่ 3.46 ตำแหน่งรถที่อยู่ในตำแหน่งเลี้ยวขวา

**กรณีที่ 3** ตำแหน่งของตัวรถอยู่ในตำแหน่งไม่ทำการเลี้ยว

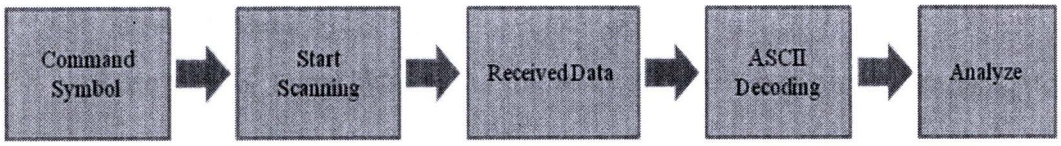
**ตัวอย่าง:** Direction = N

**หมายถึง** ให้ระบบตัดสินใจกลางไม่ทำการเลี้ยว




รูปที่ 3.47 ตำแหน่งรถที่ในตำแหน่งไม่ทำการเลี้ยว

## 3.5 การวิเคราะห์สิ่งกีดขวาง

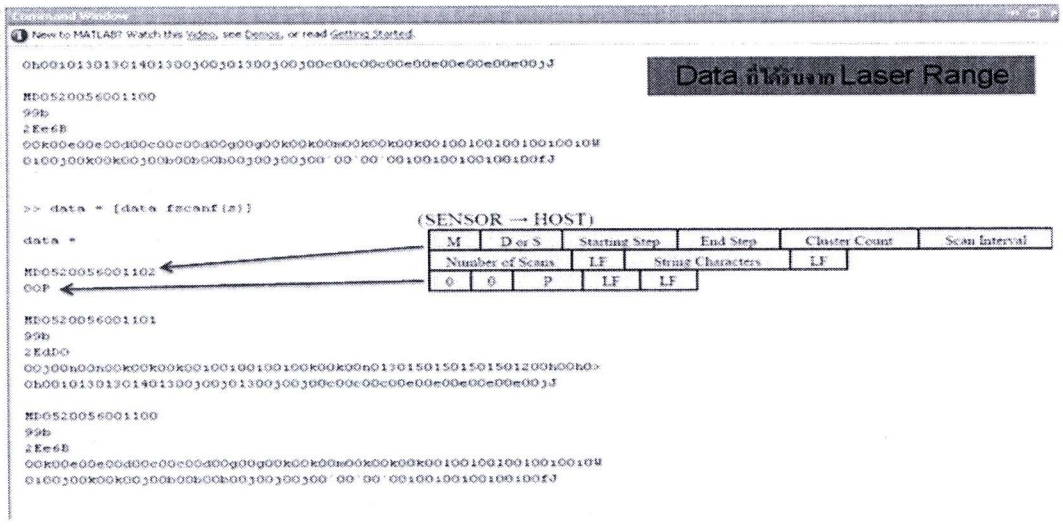



SCIP 2.0 Command | Scan from start step to end step | Data Format | Distances | - Number of object - Size of object - Distances of object

รูปที่ 3.48 ขั้นตอนการวิเคราะห์ของระบบระบุสิ่งกีดขวาง

- **Command Symbol**

การใช้คำสั่งหรือ Command ที่อยู่ใน Protocol SCIP2.0 (อ้างอิงเพิ่มเติมที่หัวข้อ 2.3.2) ในการสั่งการให้ Laser Range Finder ทำงานให้ตรงตามความต้องการ รูปที่ 3.49 แสดงค่ารับส่งค่าระหว่างเครื่องคอมพิวเตอร์และเครื่อง Laser Range



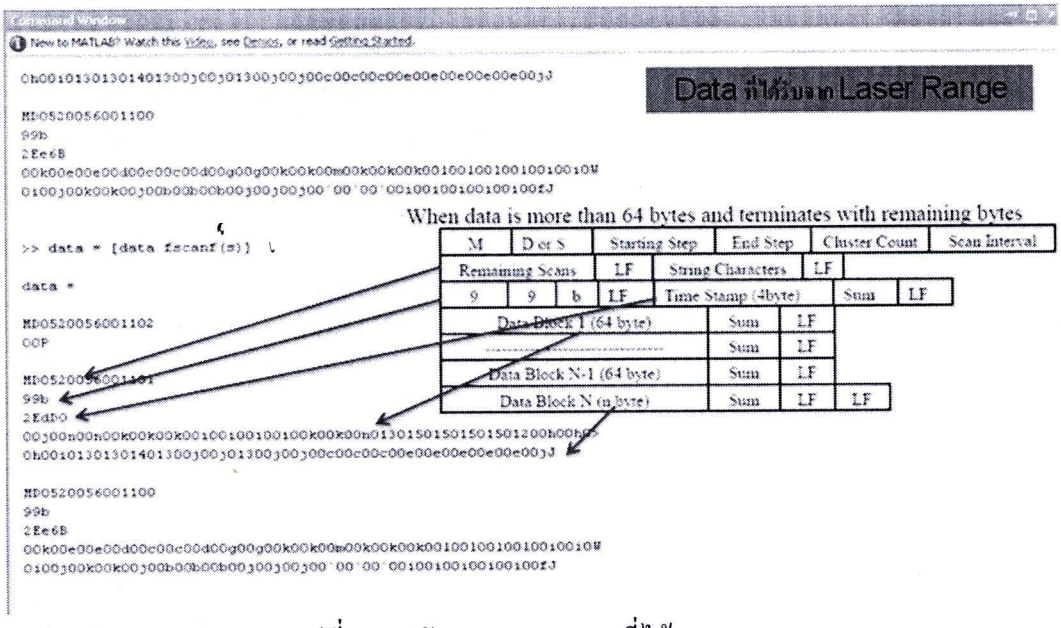

รูปที่ 3.49 ค่า Command Echo ที่ได้จาก Laser Range

- **Start Scanning**

เมื่อ Laser Range ได้รับคำสั่งจะทำการ Scan จากตำแหน่งแรกไปจนถึงตำแหน่งสุดท้าย แล้วส่งค่าของระยะทางที่ได้กลับมายังเครื่องคอมพิวเตอร์เพื่อใช้ในการวิเคราะห์ต่อไป

- **Received Data**

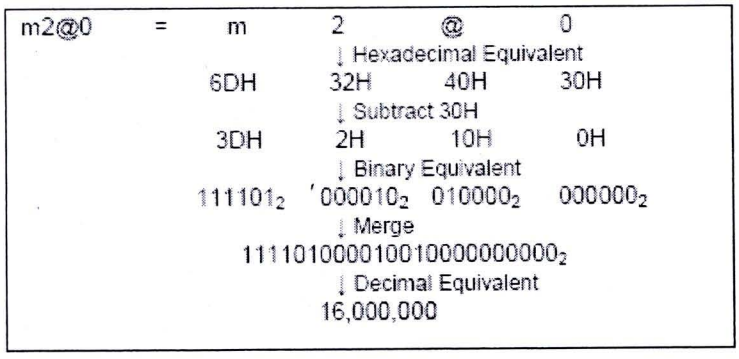
หลังจากที่ Laser Range ทำการ Scan และส่งค่ากลับมานั้น ข้อมูลที่ส่งกลับมาจะอยู่ในรูปของ Data Format และใน Format นั้นจะมีค่าของระยะทางที่ได้รวมอยู่ด้วยแต่ข้อมูลนั้นจะถูกเข้ารหัส ตามรูปแบบของ Protocol SCIP2.0 ดังแสดงในรูปที่ 3.50




รูปที่ 3.50 ลักษณะของ Data ที่ได้จากการ Scan

- **ASCII Decoding**

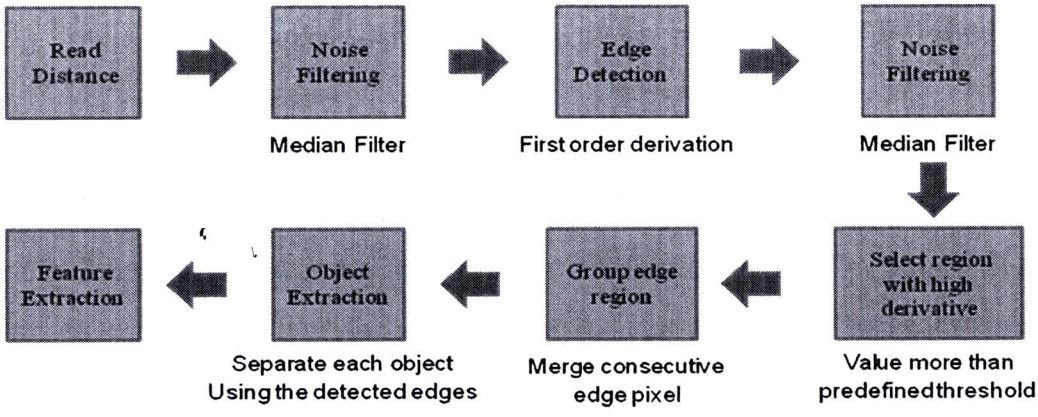
ข้อมูลดิบที่ได้จะต้องผ่านกระบวนการถอดรหัสข้อมูลตามรูปแบบของ Protocol SCIP2.0 ให้อยู่ในรูปของค่าระยะทางที่เป็นเลขฐานสิบ ซึ่งสามารถนำมาใช้ในการวิเคราะห์เพื่อหาขนาดและระยะทางของสิ่งกีดขวาง รูปที่ 3.51 แสดงตัวอย่างการถอดรหัส




รูปที่ 3.51 ตัวอย่างการถอดรหัส

- **Analyze**

การวิเคราะห์เพื่อหาจำนวนของวัตถุที่พบ การวิเคราะห์เพื่อหาขนาดและระยะของแต่ละวัตถุที่พบ จากนั้นก็จะส่งข้อมูลที่ได้ไปยังระบบตัดสินใจต่อไป ซึ่งขั้นตอนในการวิเคราะห์สามารถอธิบายได้ดังรูปที่ 3.52




รูปที่ 3.52 ขั้นตอนการวิเคราะห์หาสิ่งกีดขวาง

- **Read Distance**

การอ่านค่าระยะทางที่ได้จาก Laser Range ซึ่งค่าที่ได้นี้จะอยู่ในลักษณะของเลขฐานสิบและมีหน่วยเป็นมิลลิเมตร

- **Noise Filtering**

การอ่านค่าระยะทางที่ได้มาทำการกำจัด Noise หรือการทำให้ค่าที่ได้นั้นมีการเรียงกันอย่างเป็นระเบียบและยังทำให้เส้นกราฟที่ได้นั้นมีความชัดเจนมากยิ่งขึ้น โดยการใช้เทคนิค median filter คือการคำนวณหาค่ากลางของค่าระยะทางในตำแหน่งปัจจุบันและตำแหน่งถัดไป

- **Edge Detection**

การใช้เทคนิคของ Edge Detection โดยการเอาค่าตำแหน่งปัจจุบันลบจากตำแหน่งด้านซ้ายเพื่อหาส่วนต่างที่เป็นขอบวัตถุ

- **Noise Filtering**

นำจุดที่ได้จาก Edge Detection มาทำการกำจัด Noise หรือการทำให้ค่าที่ได้นั้นมีการเรียงกันอย่างเป็นระเบียบและยังทำให้เส้นกราฟที่ได้นั้นมีความชัดเจนมากยิ่งขึ้น โดยการใช้เทคนิค median filter

- **Select region with high derivative**

กำหนดขอบเขตของ Object โดยการใช้ค่าที่มีการเปลี่ยนแปลงอย่างกะทันหัน เช่น ตำแหน่งที่มีค่ามากอย่างกะทันหันหรือตำแหน่งที่มีค่าน้อยอย่างกะทันหัน

- **Group edge region**

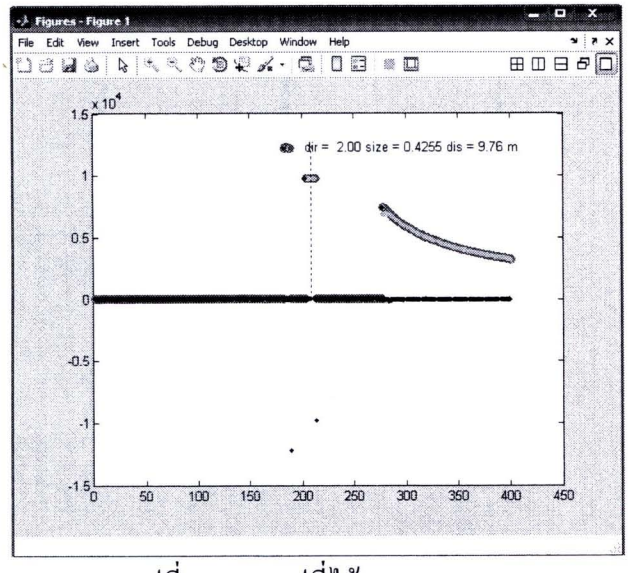
นำจุดที่มีการเปลี่ยนแปลงอย่างกะทันหันในบริเวณที่ใกล้เคียงกันมารวมเข้าด้วยกันเป็นขอบเขตของวัตถุ

- **Object extraction**

การแบ่ง Object โดยกำหนดเส้นคั่นของแต่ละ Object โดยค่าของ edge ที่ได้มาเป็นตัวแบ่ง

- **Feature extraction**

การวิเคราะห์เพื่อเลือก Object ที่มีลักษณะเด่นหรือ Object ที่สนใจ โดยการกำหนดขนาดของวัตถุที่สนใจ ทำให้เหลือเพียงวัตถุที่มีขนาดตรงตามที่ต้องการ แต่ถ้ามีวัตถุที่มีขนาดที่ต้องการอยู่หลายชิ้น ก็จะทำการพิจารณาเฉพาะวัตถุที่อยู่ใกล้ที่สุดเท่านั้น และทำการส่งข้อมูลมุมจากด้านหน้ารถ และระยะทางจากวัตถุถึงรถให้ระบบตัดสินใจกลาง ดังแสดงตัวอย่างการวิเคราะห์และผลลัพธ์ในรูปที่ 3.53 และ 3.54




รูปที่ 3.53 กราฟที่ได้จากการ Scan

```
Obstracles =

        StepStart : 203

        StepEnd : 213

        Size : 0.425

        dir : 2

    Distance_Barrier : 9.760
```

รูปที่ 3.54 ข้อมูลผลการวิเคราะห์ที่ได้จากการ Scan

**ค่าที่ระบบวิเคราะห์สิ่งกีดขวางส่งให้ระบบตัดสินใจกลาง**

กรณีที่เกิดขึ้นในระบบวิเคราะห์สิ่งกีดขวาง และจะมีการส่งข้อมูลไปยังระบบตัดสินใจกลางมีอยู่ด้วยกัน 2 กรณี ดังต่อไปนี้

From Obstacle = [dir, Distance_Obstacle] โดยที่

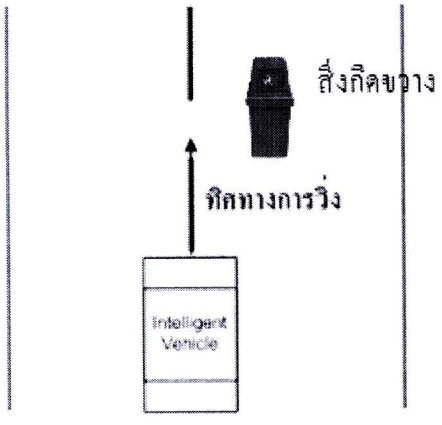
dir = - unit in degree, + unit in degree (บ่งบอกว่าวัตถุอยู่ทางด้านใดของตัวรถและทิศทางกี่องศา)

Distance_Obstacle = unit in m (ระยะห่างตัวรถกับสิ่งกีดขวาง)

**กรณีที่ 1** สิ่งกีดขวางอยู่ทางด้านขวา ดังแสดงตัวอย่างในรูปที่ 3.55

**ตัวอย่าง:** dir = -4.25, Distance_Obstacle = 7.23

**หมายถึง** วัตถุอยู่ทางด้านขวาของรถ มีองศาจากด้านหน้าของรถเท่ากับ 4.25 องศา และมีระยะห่าง 7.23 เมตร

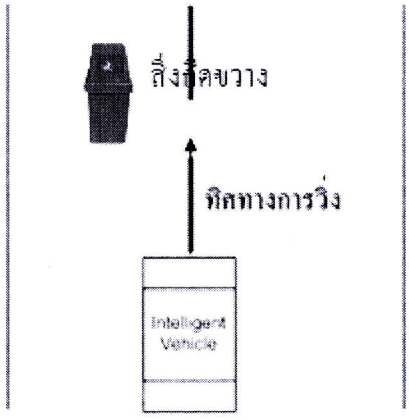



รูปที่ 3.55 สิ่งกีดขวางอยู่ทางด้านขวา

**กรณีที่ 2** สิ่งกีดขวางอยู่ทางด้านซ้าย ดังแสดงตัวอย่างในรูปที่ 3.56

**ตัวอย่าง:** dir = 2, Distance_Obstacle = 9.76

**หมายถึง** วัตถุอยู่ทางด้านซ้ายของรถ มีองศาจากด้านหน้าของรถเท่ากับ 2 องศา และมีระยะห่าง 9.76 เมตร




รูปที่ 3.56 สิ่งกีดขวางอยู่ทางด้านซ้าย

สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

### 3.6 การวิเคราะห์สัญญาณ GPS

เนื่องจากงานวิจัยเพื่อพัฒนารถอัจฉริยะมีการนำระบบต่างๆ เข้ามาร่วมในการตัดสินใจหาเส้นทางการเคลื่อนที่ของรถ ระบบที่นำมาใช้ได้แก่ ระบบระบุตำแหน่ง ระบบนำทาง และระบบแผนที่ ซึ่งแต่ละระบบที่กล่าวมาล้วนจำเป็นที่จะต้องอาศัยการทำงานของเทคโนโลยี GPS ทั้งสิ้น ดังนั้นเนื้อหาในหัวข้อนี้จะกล่าวถึงการวิเคราะห์สัญญาณ GPS เพื่อนำค่าที่ได้จากสัญญาณ GPS ไปใช้ในการวิเคราะห์ระบบต่างๆ ดังที่กล่าวมา

#### 3.6.1 รูปแบบของสัญญาณ GPS

การวางแผนในงานวิจัยนี้ได้เลือกใช้ GPS Module ของ HULUK รุ่น GM-82 เนื่องจาก GPS Module มีการส่งข้อมูลผ่านทางพอร์ตสื่อสารอนุกรม (Serial Port) ซึ่งสามารถนำไปเชื่อมต่อกับเครื่องประมวลผลโดยตรง สำหรับข้อมูลที่ได้จาก GPS Module จะเป็นข้อความ ASCII Code ตามมาตรฐาน NMEA (The National Marine Electronics Association) โดยมี Output NMEA Messages ข้อมูลที่มาจาก GPS Module จะแบ่งเป็น Sentence ต่างๆ โดยแต่ละ Sentence จะขึ้นต้นด้วยเครื่องหมาย $ และจะสิ้นสุด Sentence ด้วยการขึ้นบรรทัดใหม่ ซึ่งเห็นได้ว่ามี Sentence [17] ต่างๆหลายรูปแบบดังตัวอย่าง

$GPGGA,102434,1611.0053,N,10318.3853,E,1,09,1.8,150.2,M,-21.8,M,,*65

$GPGSA,A,3,01,03,11,13,16,19,20,23,,31,,,2.8,1.8,1.3*3D

$GPGSV,3,2,10,16,28,028,39,19,56,175,42,20,24,229,42,23,59,315,47*78

$GPGLL,1611.0053,N,10318.3853,E,102434,A,A*40

$GPBOD,,T,,M,,*47

$GPVTG,235.9,T,236.4,M,9.8,N,18.2,K*7A

$PGRME,10.0,M,13.6,M,16.9,M*15

$PGRMZ,493,f,3*15

$PGRMM,User*58

$GPRTE,1,1,c,*37

$GPRMC,102436,A,1611.0017,N,10318.3802,E,11.5,232.6,040707,0.5,W,A*39

$GPRMB,A,,,,,,,,,,,A,A*0B

ค่าต่างๆที่อ่านได้จาก GPS Module มีความหมายแตกต่างกันซึ่งอธิบายได้ดังนี้

**GGA** (Global Positioning System Fixed Data) ข้อมูลเวลาตำแหน่ง และ fix type data

**GLL** (Geographic Position-Latitude/Longitude) ให้ข้อมูลพิกัด (Latitude, longitude), เวลา UTC ของพิกัดและ สถานะ

**GSA** (GNSS DOP and Active Satellites) ให้ข้อมูล Operating mode จำนวนดาวเทียมที่ใช้ในการคำนวณตำแหน่ง ค่า DOP

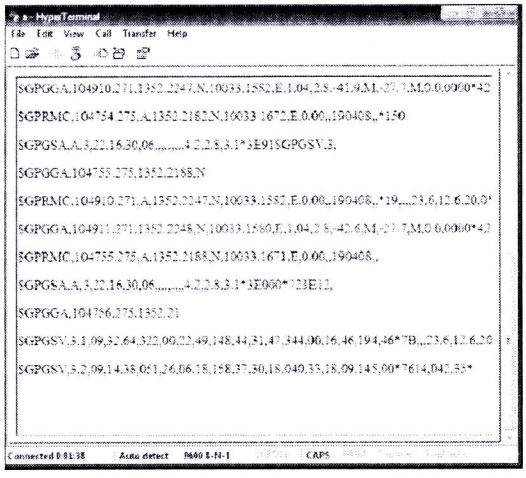
**GSV** (GNSS Satellites in View) ให้ข้อมูลจำนวนดาวเทียมที่จีพีเอสเห็นพร้อมหมายเลข (ID number) ความสูง (elevation) องศาในแนวขอบฟ้า (azimuth) และค่า SNR (Signal-to-Noise-Ratio) หรืออัตราส่วนสัญญาณจีพีเอสเมื่อเทียบกับสัญญาณรบกวน

**RMC** (Recommended Minimum Specific GNSS) ให้ข้อมูล เวลา วันที่ พิกัดมุ่งหน้าไปทางทิศใด และความเร็ว

**VTG** (Course Over Ground and Ground Speed) ให้ข้อมูลทิศทางและความเร็วเมื่อเทียบกับพื้นดิน

### 3.6.2 เขียนโปรแกรมเพื่อรับค่าสัญญาณ GPS

หลังจากทำการวิเคราะห์รูปแบบการให้ข้อมูลของ GPS Module แล้วทำการเชื่อมต่อGPS Module เข้ากับเครื่องคอมพิวเตอร์ ผ่าน USB พอร์ตเมื่อใช้โปรแกรม Hyper Terminal อ่านค่าจาก GPS Moduleได้ค่าดังรูปที่ 3.57




รูปที่ 3.57 ค่าที่อ่านได้จาก GPS Module

เมื่อสามารถรับค่าสัญญาณจาก GPS Module แล้วในส่วนต่อไปจะเป็นการนำค่าที่ได้ มาทำการแยกส่วนที่ต้องการ และไม่ต้องการออกจากกัน จากนั้นจะส่งค่าที่ได้ เพื่อทำการวิเคราะห์แล้วส่งให้กับระบบต่างๆ ตามที่ต้องการต่อไป ในงานวิจัยนี้จะสนใจเฉพาะ Sentence ที่ขึ้นต้นด้วย $GPRMC ซึ่ง Sentence นี้จะมีตำแหน่งที่บอกค่าพิกัดมุ่งหน้า ,ความเร็วเพื่อนำมาคำนวณหาค่าในแนวแกน X และ Y ดังแสดงในรูปที่ 3.58 โดยมีแผนการดำเนินงานการวิเคราะห์สัญญาณ GPS ดังนี้

**ค่าที่ระบบวิเคราะห์สัญญาณ GPS ส่งให้ระบบตัดสินใจกลาง**

กรณีที่เกิดขึ้นในระบบวิเคราะห์ป้ายจราจร และจะมีการส่งข้อมูลไปยังระบบตัดสินใจกลาง มีดังต่อไปนี้

From GPS = [Latitude, Longitude] โดยที่

Latitude = 0 ถึง 90 องศา (เป็นพิกัดที่ใช้บอกตำแหน่งบนพื้นโลกและแบ่งเขตสภาวะอากาศโดยวัดจากเส้นศูนย์สูตร)

Longitude = -180 ถึง 180 องศา (เป็นพิกัดที่ใช้บอกตำแหน่งบนพื้นโลก โดยวัดไปทางตะวันตกหรือตะวันออก)

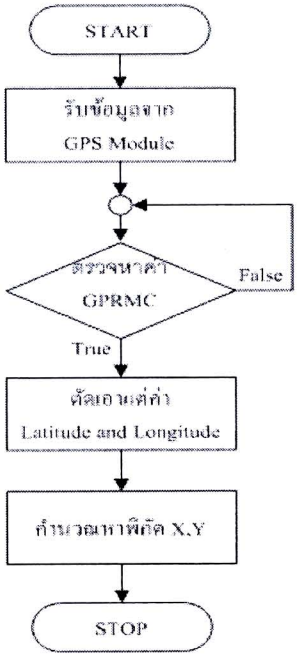
**ตัวอย่าง 1:** Latitude = 37.77, Longitude = -122.41

**หมายถึง** ตำแหน่งที่ได้คือ เส้นศูนย์สูตรที่ 37.77 องศา และวัดไปทางตะวันตก -122.41 องศา

**ตัวอย่าง2:** Latitude = 10.771, Longitude = 99.359

**หมายถึง** ตำแหน่งที่ได้คือ เส้นศูนย์สูตรที่ 10.771 องศา และวัดไปทางตะวันออก 99.359 องศา

**แผนการดำเนินงานการวิเคราะห์สัญญาณ GPS**




รูปที่ 3.58 แผนการดำเนินงานการวิเคราะห์สัญญาณ GPS

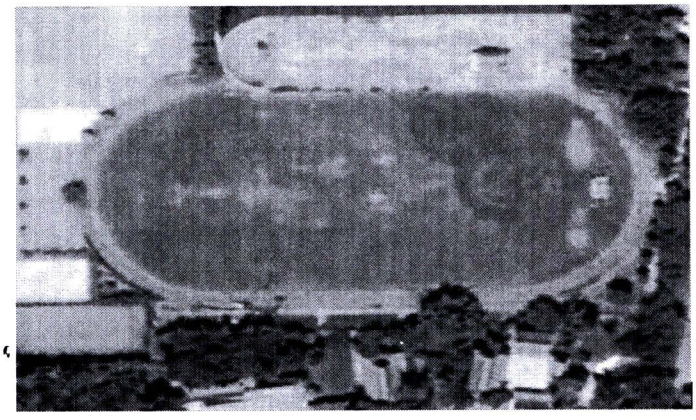
**3.7 ระบบแผนที่**

เป็นการทำงานในส่วนของระบบแผนที่ โดยเริ่มจากส่วนของ Predefined Data คือ การนำข้อมูลที่เป็นไฟล์รูปภาพสนามที่จะใช้ในการทดสอบ ซึ่งเป็นไฟล์รูปภาพนามสกุล .jpg เป็นรูปที่หาจากเว็บไซต์ของการแข่งขันรถอัจฉริยะหรือจาก Google Earth มาทำการกำหนดจุดต่างๆ ที่เป็นทางโค้งหรือทางแยก เพื่อจะบอกทิศทางในการเลี้ยวให้กับระบบตัดสินใจกลาง และรับค่าพิกัดเริ่มต้นเพื่อให้รู้ว่ารถนั้นอยู่ในตำแหน่งไหนของแผนที่ เมื่อมีการทำงานระบบตัดสินใจกลาง จะสอบถามมายังระบบแผนที่ตลอดเวลา ถ้าถึงบริเวณที่เป็นทางโค้งหรือทางแยกระบบแผนที่ก็จะส่งทิศทางการวิ่งกลับไปให้ระบบตัดสินใจกลาง ทำการประมวลผล

### 3.7.1 สนามที่ใช้ในการทดสอบ

สนามที่ใช้ทดสอบระบบแผนที่มีอยู่ด้วยกัน 2 สนามดังนี้

สนามฟุตบอลมหาวิทยาลัยธุรกิจบัณฑิตย์ เป็นสนามที่ใช้ในการทดสอบดังแสดงในรูปที่ 3.59 ซึ่งรอบสนามฟุตบอลนี้มีระยะทางประมาณ 400 เมตร ลู่วิ่งกว้างประมาณ 3 เมตร



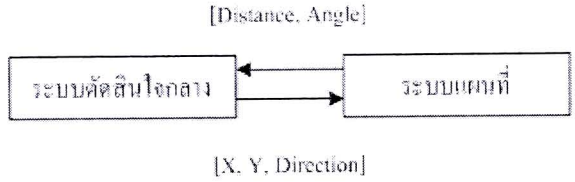
รูปที่ 3.59 สนามฟุตบอลของมหาวิทยาลัยธุรกิจบัณฑิตย์

สนาม Bangkok Racing Circuit (BRC) เป็นสนามที่ใช้ในการแข่งขันดังแสดงในรูปที่ 3.60 โดยข้อมูลของสนามคือ สนามรอบนอกยาวประมาณ 1 กิโลเมตร เส้นทางทั้งหมดยาวประมาณ 2.4 กิโลเมตร ความกว้างของลู่สนามไม่คงที่ กว้างที่สุด 12 เมตร แคบที่สุดประมาณ 6 เมตร

รูปที่ 3.60 สนาม Bangkok Racing Circuit

### 3.7.2 ส่วนของการระบุทิศทางด้วยแผนที่ (Map Referencing)

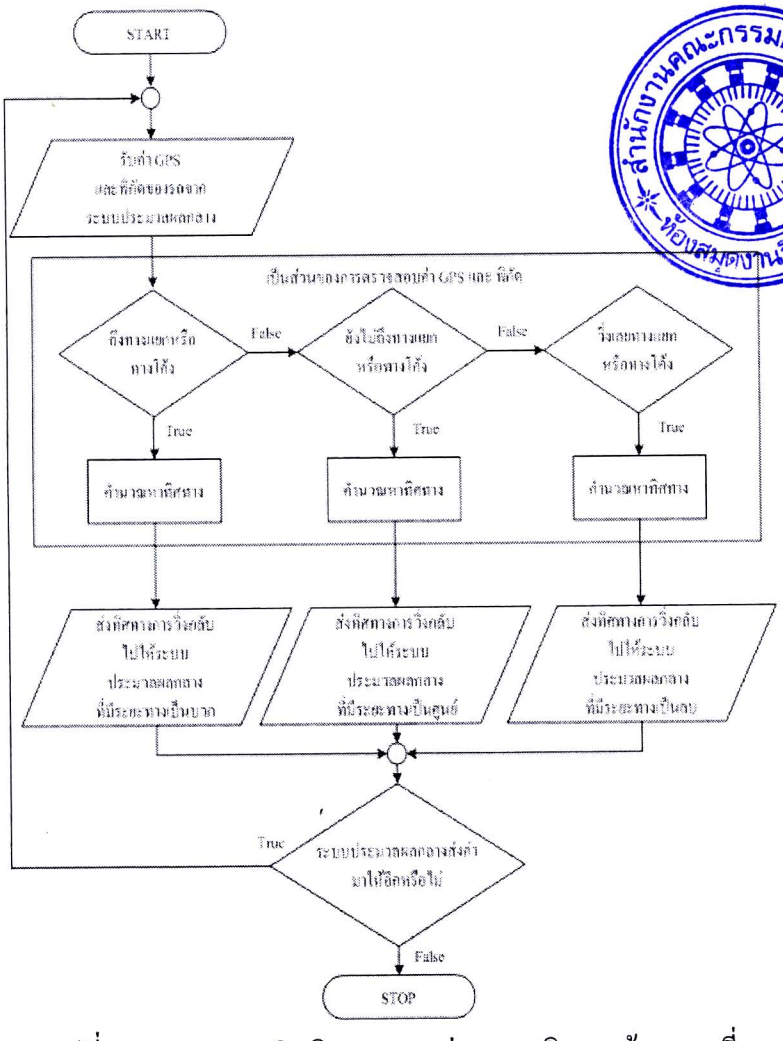
การทำงานของส่วนนี้ เริ่มต้นด้วยระบบตัดสินใจกลางจะทำการส่งค่าพิกัดของรถ และองศาของหน้ารถ เพื่อให้ระบบแผนที่ส่งระยะจากทางโค้งหรือทางแยกถึงตัวรถและมุมที่ใช้ในการเลี้ยวกลับไปให้ระบบตัดสินใจกลาง ดังตัวอย่างในรูปที่ 3.61




รูปที่ 3.61 การทำงานของส่วนระบุทิศทางด้วยแผนที่

เมื่อมีการทำงาน ระบบตัดสินใจกลางจะเริ่มส่งค่าเพื่อทำการสอบถามระยะห่างระหว่างตัวรถกับทางโค้งหรือทางแยก พร้อมกับมุมที่ต้องเลี้ยว ในการทำงานของระบบแผนที่นั้น เมื่อได้รับค่าพิกัดมาแล้ว ต้องมีการคำนวณค่าระยะห่างที่ใกล้เข้าไปเรื่อยๆ เมื่อเข้าใกล้บริเวณที่ต้องเลี้ยวระบบแผนที่ก็จะส่งระยะห่างระหว่างตัวรถ พร้อมกับมุมเลี้ยวให้ระบบตัดสินใจกลาง ดังแสดงในรูปที่ 3.62

**แผนการดำเนินการส่วนของการระบุทิศทางด้วยแผนที่**




สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

รูปที่ 3.62 แผนการดำเนินการของส่วนระบุทิศทางด้วยแผนที่

**รูปแบบข้อมูลที่ระบบตัดสินใจกลางส่งให้ระบบแผนที่**

[x, y, Direction] โดยที่

x = ตำแหน่งในแนวแกน x

y = ตำแหน่งในแนวแกน y

Direction = -180 ถึง 180 (องศาของหน้ารถ)

ตัวอย่าง: (100 200 30)

หมายถึง ตำแหน่ง x, y เป็น 100,200 องศาหน้ารถคือ 30 องศา

**ค่าที่ระบบแผนที่ส่งให้ระบบตัดสินใจกลาง**

[Distance, Angle] โดยที่

Distance = หน่วยเป็นซม. (ตำแหน่งของตัวรถ ถึง ตำแหน่งทางโค้งหรือทางแยก)

Angle = -180 ถึง 180 (องศาที่นำมาอ้างอิงในการเลี้ยว)

ตัวอย่าง: (100 30)

หมายถึง ระยะทางระหว่างตัวรถถึงทางโค้งหรือทางแยกเป็น 100 ซม. และให้ระบบตัดสินใจกลางเลี้ยวซ้าย 30 องศา

กรณีที่เกิดขึ้นในระบบแผนที่และจะมีการส่งข้อมูลกลับไปยังระบบตัดสินใจกลางมีอยู่ด้วยกัน 3 กรณี ดังต่อไปนี้

กรณีที่ 1 ตำแหน่งของตัวรถอยู่ในระยะที่กำหนดไว้

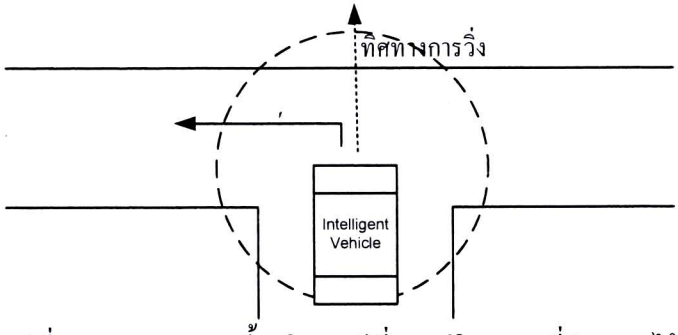
From Map = [Distance, Angle] โดยที่

Distance = หน่วยเป็นซม. (ตำแหน่งของตัวรถ ถึง ตำแหน่งโค้งหรือทางแยก)

Angle = -180 ถึง 180 (องศาที่นำมาอ้างอิงในการเลี้ยว)

ตัวอย่าง: (100 45)

หมายถึง ระยะทางจากตัวรถถึงตำแหน่งทางโค้งหรือทางแยกเป็น 100 ซม. ให้ระบบตัดสินใจกลางเลี้ยวซ้าย 45 องศา




รูปที่ 3.63 การหามุมเลี้ยวในกรณีที่รถอยู่ในระยะที่กำหนดไว้

กรณีที่ 2 ตำแหน่งของตัวรถไม่ได้อยู่ใกล้ทางแยก

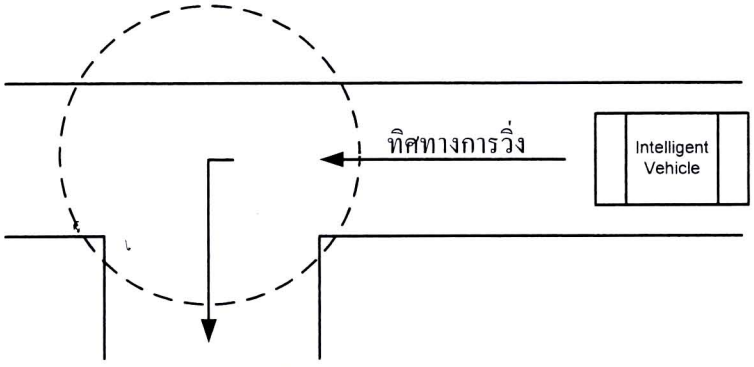
From Map = [Distance, Angle] โดยที่

- Distance = 0

  Angle = 0

ตัวอย่าง: (0 0)

หมายถึง ระบบแผนที่จะส่งค่า 0, 0 กลับไปให้ระบบตัดสินใจกลางเพื่อเป็นการบอกว่ายังไม่ถึงทางแยก




รูปที่ 3.64 การหามุมเลี้ยวในกรณีที่รถไม่ได้อยู่ในระยะที่กำหนดไว้

กรณีที่ 3 ตำแหน่งของตัวรถวิ่งเลยจุดที่กำหนดไว้

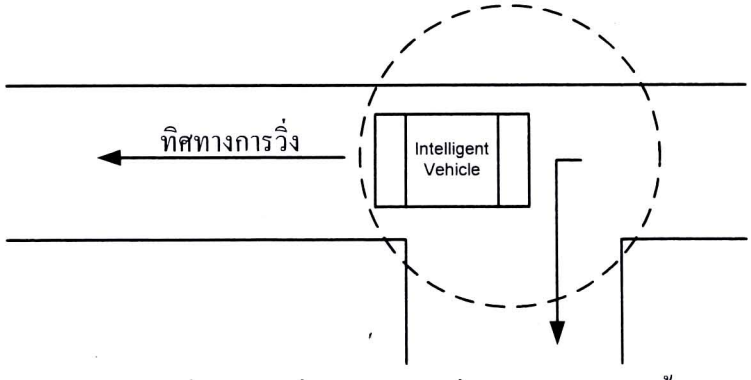
From Map = [-Distance, Angle] โดยที่

-Distance = หน่วยเป็น ซม. (ตำแหน่งของตัวรถ ถึง ตำแหน่งโค้งหรือทางแยก)

Angle = 0-360(องศาที่นำมาอ้างอิงในการเลี้ยว)

ตัวอย่าง: (-100 45)

หมายถึง รถได้วิ่งเลยจุดที่กำหนดไว้ 100 ซม. และต้องเลี้ยวซ้ายด้วยมุม 45 องศา




รูปที่ 3.65 รถวิ่งไปถึงระยะที่กำหนดไว้แต่ไม่เลี้ยว

### 3.7.3 เขียนโปรแกรมติดต่อกับระบบตัดสินใจกลาง

**ทดสอบการสอบถามจากระบบตัดสินใจกลาง**

- ระบบแผนที่ส่งทิศทางการวิ่งกลับไป
- ระบบแผนที่ส่งมุมเลี้ยวกลับไป

**เขียนโปรแกรมติดต่อกับระบบตัดสินใจกลาง**

พัฒนาโปรแกรมให้สามารถทำงานร่วมกับระบบตัดสินใจกลาง โดยระบบตัดสินใจกลางจะมี GUI หลักที่ใช้ได้ทั้งแบบจำลองการทำงาน และแบบการทำงานจริง โดยขั้นตอนหลักๆ ในการเขียนโปรแกรมมีดังนี้

- สร้าง Popup Menu ให้สามารถเลือกแผนที่ได้ทั้งสองสนาม
- สร้าง Axes เพื่อใช้แสดงแผนที่ที่ทำการเลือก
- สร้างปุ่มที่รับค่า Input จากเมาส์เพื่อกำหนดจุดที่เป็นทางโค้งหรือทางแยก
- สร้างปุ่มที่รับค่า Input จากเมาส์เพื่อกำหนดทิศทางในการวิ่งและการเลี้ยว

การจำลองการทำงานจะมีการแสดงจุดที่วิ่งเรื่อยๆ เสมือนว่าเป็นรถยนต์เคลื่อนที่อัตโนมัติที่กำลังวิ่งอยู่บนแผนที่

### 3.8 สรุปขั้นตอนการออกแบบโดยรวม

การออกแบบในแต่ละส่วนของการทำงานจะสามารถจำแนกได้ 2 ลักษณะ คือ การทำงานในส่วนของฮาร์ดแวร์ และซอฟต์แวร์ ซึ่งการทำงานในส่วนของฮาร์ดแวร์หลักๆ แล้วจะทำงานอยู่บนระบบไมโครคอนโทรลเลอร์เกือบทั้งหมด ดังนั้น การเชื่อมต่ออุปกรณ์ต่างๆ เข้าด้วยกันก็จะทำกับบอร์ดไมโครคอนโทรลเลอร์เช่นกัน เมื่อระบบตัดสินใจกลางต้องการติดต่อกับอุปกรณ์ ตัวใดก็จะทำผ่านระบบไมโครคอนโทรลเลอร์ผ่าน Serial port ส่วนอุปกรณ์อื่นๆที่มีพอร์ต USB เป็นปลั๊กอินก็ให้เชื่อมต่อกับเครื่องประมวลผลโดยตรง แล้วให้ระบบซอฟต์แวร์เป็นตัวจัดการกับอุปกรณ์นั้นๆ ให้สามารถติดต่อกับระบบตัดสินใจกลางได้

การทดสอบการทำงาน จะแยกทดสอบเป็นส่วนๆ เพื่อสะดวกในการหาและแก้ไขข้อผิดพลาดได้อย่างถูกต้องตรงจุด ไม่เกิดความสับสน ทั้งนี้การทดสอบระบบฮาร์ดแวร์ จะทำควบคู่ไปกับซอฟต์แวร์ที่พัฒนาขึ้นมาเพื่อรองรับการทำงานของอุปกรณ์นั้นๆโดยเฉพาะ ทั้งแบบอุปกรณ์ที่เชื่อมต่อกับเครื่องประมวลผลโดยตรง และแบบเชื่อมต่อผ่านบอร์ดไมโครคอนโทรลเลอร์ เมื่อทดสอบในแต่ละส่วนจนแน่ใจว่าเป็นไปตามที่ออกแบบ จึงทำการเชื่อมต่ออุปกรณ์ต่างๆ เข้าด้วยกันแล้วทำการทดสอบระบบโดยรวมอีกครั้งหนึ่ง