บทที่ 3

วิธีดำเนินงานวิจัย

การดำเนินงานวิจัยเพื่อให้ง่ายต่อการเพิ่มประสิทธิภาพของรถขนส่งสินค้าอัตโนมัติ ผู้วิจัยได้มีแนวคิดในการเพิ่มประสิทธิภาพของรถขนส่งสินค้าอัตโนมัติโดยสามารถเคลื่อนที่ไปยังจุดหมายด้วยการนำทางจากระบบปฏิบัติการหุ่นยนต์แทนการใช้เส้นแถบแม่เหล็กและคำนวณเส้นทางการเคลื่อนที่ใหม่เมื่อมีสิ่งกีดขวาง โดยเริ่มต้นจากศึกษาการออกแบบรถขนส่งสินค้าอัตโนมัติ ศึกษาการใช้งานระบบปฏิบัติการหุ่นยนต์ การใช้งานเซ็นเซอร์ต่างๆ การสร้างแผนที่โดยใช้เซ็นเซอร์ในการสร้าง รวมไปถึงการศึกษาการเขียนภาษาที่ต้องใช้ในการเขียนโปรแกรมทั้งภาษา xml , c และ python โดยผู้วิจัยได้แบ่งขั้นตอนการดำเนินงานวิจัยดังนี้

3.1 การเขียนโปรแกรมควบคุมรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์ด้วยอาดูโน่

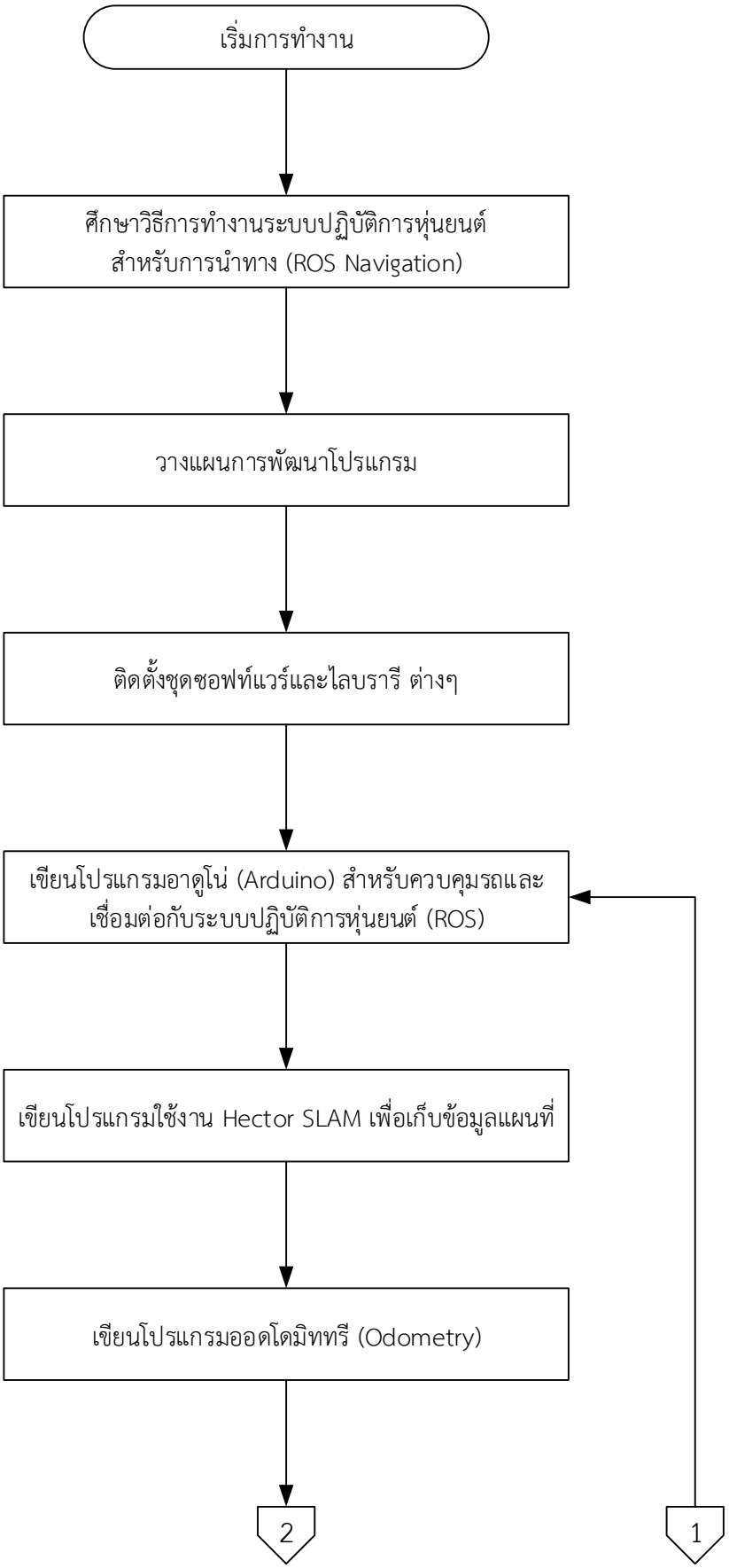
3.2 การเขียนโปรแกรมสร้างแผนที่ด้วย Hector SLAM

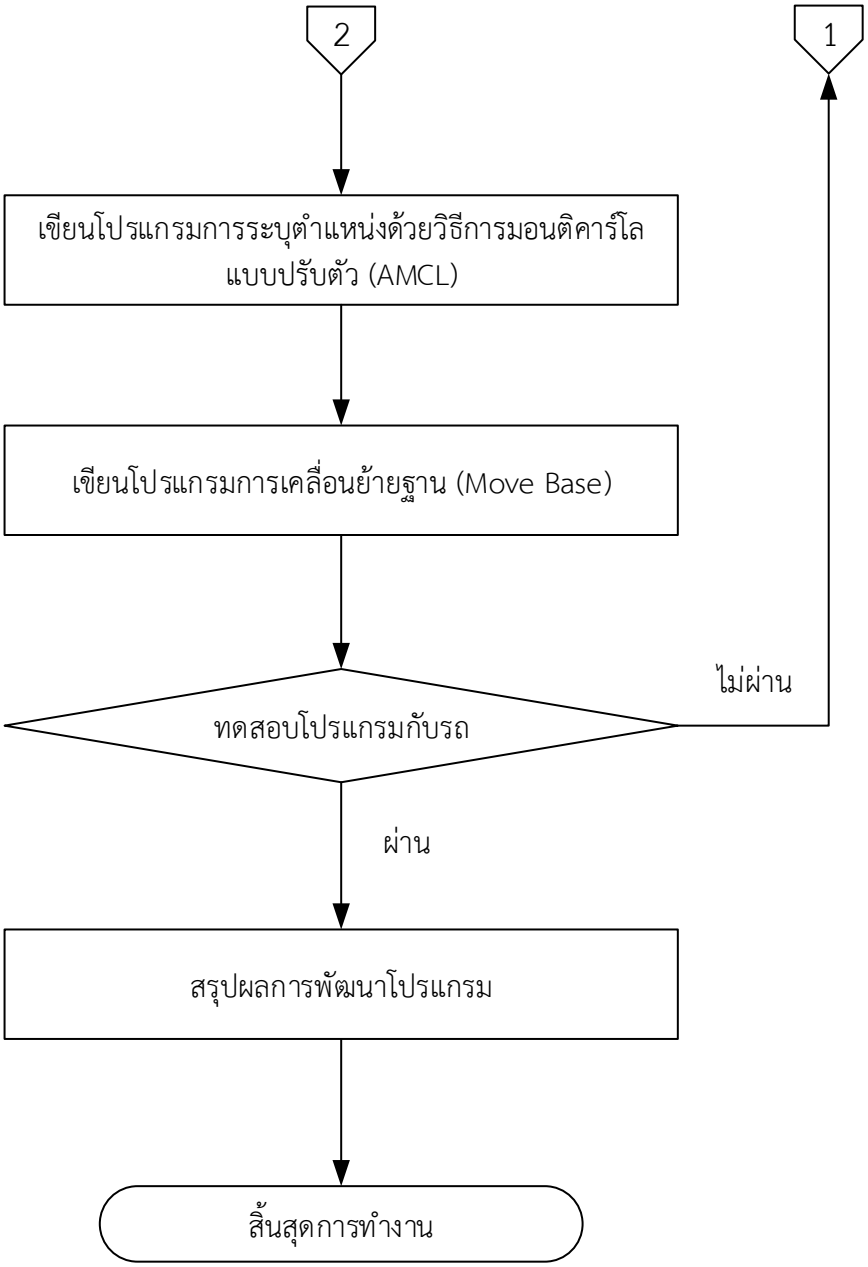
3.3 การเขียนโปรแกรมสั่งงานรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์เพื่อเคลื่อนที่ไปยังจุดหมายและหลบหลีกสิ่งกีดขวาง

3.4 การใช้งานของระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

3.5 การทดลองหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

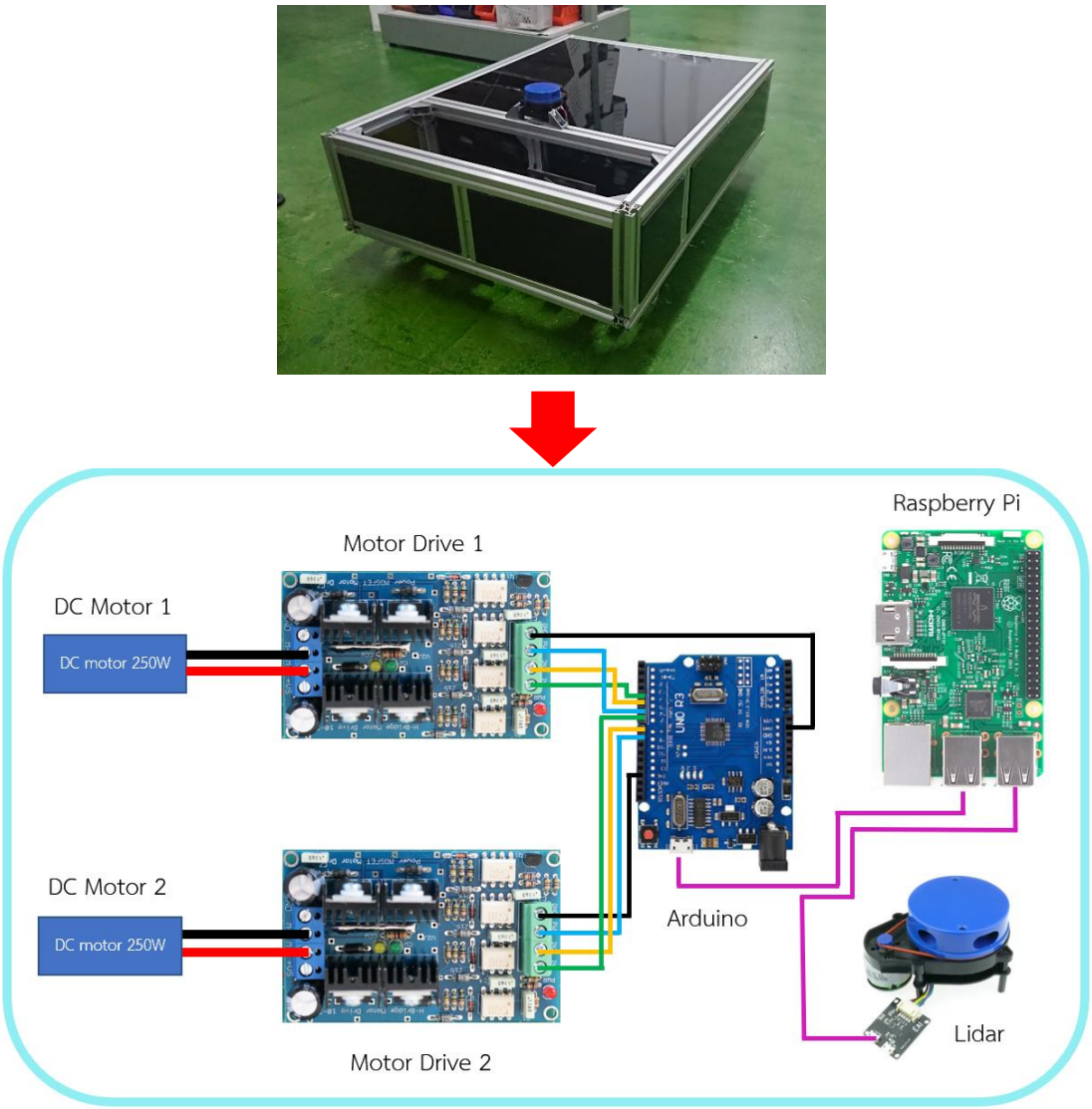
ในการดำเนินการวิจัยผู้วิจัยได้ทำการเขียนโปรแกรมการทำงานต่างๆ เพื่อให้ได้ประสิทธิภาพสูงสุดในการดำเนินงานวิจัยนี้ ซึ่งผู้วิจัยได้แบ่งขั้นตอนต่างๆ ในการดำเนินงานเพื่อศึกษาหาวิธีการที่จะทำให้รถขนส่งสินค้าอัตโนมัติมีการค้นหาเส้นทางใหม่และมีความรวดเร็วในการเคลื่อนที่ไปยังจุดหมายโดยขั้นตอนต่างๆ ผู้วิจัยได้จัดทำเอาไว้แสดงในแผนผังการดำเนินงานวิจัยดังแสดงในภาพที่ 3.1


เริ่มการทำงาน
ศึกษาวิธีการทำงานระบบปฏิบัติการหุ่นยนต์ สำหรับการนำทาง (ROS Navigation)
วางแผนการพัฒนาโปรแกรม
ติดตั้งชุดซอฟท์แวร์และไลบรารี ต่างๆ
เขียนโปรแกรมอาดูโน่ (Arduino) สำหรับควบคุมรถและเชื่อมต่อกับระบบปฏิบัติการหุ่นยนต์ (ROS)
เขียนโปรแกรมใช้งาน Hector SLAM เพื่อเก็บข้อมูลแผนที่
เขียนโปรแกรมออดโดมิททรี (Odometry)
2
1



ภาพที่ 3.1 แผนภาพแสดงขั้นตอนการดำเนินงานวิจัยการประยุกต์ใช้ระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

จากภาพที่ 3.1 แสดงแผนผังการดำเนินงานวิจัยการประยุกต์ใช้ระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม จากแผนผังการดำเนินงานวิจัยได้เริ่มศึกษาการออกแบบรถ ส่วนประกอบโครงสร้างของรถและวิธีการขับเคลื่อนเพื่อให้เข้าใจหลักการควบคุมรถสำหรับการเขียนโปรแกรมในการควบคุมต่อไป จากนั้นศึกษาวิธีการทำงานของระบบปฏิบัติการหุ่นยนต์สำหรับการนำทาง ซึ่งมีทั้งวิธีการติดตั้งชุดซอฟต์แวร์และไลบรารี่ การใช้คำสั่งต่างๆ ที่เกี่ยวข้องทั้งหมด เมื่อเข้าใจระบบดีแล้วจึงได้วางแผนการพัฒนาโปรแกรมอย่างเป็นระบบ โดยอันดับแรกได้ทำการติดตั้งชุดซอฟท์แวร์และไลบรารี่ต่างๆลงบนราสเบอร์รี่พาย (Raspberry Pi) และอาดูโน่ (Arduino) อุปกรณ์ทั้งสองชนิดนี้เป็นคอนโทรลเลอร์สำคัญในการควบคุมรถ จะทำหน้าที่เป็นเซิร์ฟเวอร์ของระบบปฏิบัติการหุ่นยนต์การประมวลผลคำสั่งและไลบรารี่เกี่ยวกับการนำทางทั้งหมด ส่วนอาดูโน่จะทำหน้าที่รับข้อมูลความเร็วของรถจากราสเบอรี่พาย เพื่อนำมาสั่งงานมอเตอร์ให้เคลื่อนที่ต่อไป เมื่อทำการติดตั้งชุดซอฟต์แวร์ทั้งหมดจึงเริ่มต้นการเขียนโปรแกรม โดยเริ่มในส่วนของอาดูโน่ทดลองรับค่าความเร็วของรถจากราสเบอรี่พายและทำการขับเคลื่อนรถตามข้อมูลความเร็วของรถที่ได้รับมา เมื่อโปรแกรมส่วนของอาดูโน่เสร็จแล้ว จึงทำโปรแกรมในส่วนของราสเบอรี่พาย สร้างแผนที่สำหรับการเดินรถโดยใช้ซอฟท์แวร์ที่ชื่อว่า Hector SLAM ซึ่งเป็น Open Source สำหรับสร้างแผนที่ตัวหนึ่งในระบบปฏิบัติการหุ่นยนต์ เขียนโปรแกรมในส่วนออดโดมิททรีด้วยไลบรารี่ RF2o Odometry สำหรับสร้างข้อมูลความเร็วจริงของรถจากไลดาร์เพื่อนำมาใช้ในการคำนวณต่างๆที่เกี่ยวข้อง ต่อมาเขียนโปรแกรมการทำการระบุตำแหน่งด้วยวิธีการมอนติคาร์โลแบบปรับตัว เพื่อให้ระบบปฏิบัติการหุ่นยนต์รู้ตำแหน่งปัจจุบันของรถในแผนที่ จากนั้นเขียนโปรแกรมการการเคลื่อนย้ายฐานเพื่อสั่งงานให้รถเคลื่อนที่ไปยังจุดหมายที่กำหนดได้ โดยในโปรแกรมการเคลื่อนย้ายฐานจะมีทั้งการรับค่าแผนที่จากการระบุตำแหน่งพร้อมกับการสร้างแผนที่ ข้อมูลความเร็วของรถจากออดโดมิททรีตำแหน่งปัจจุบันของรถจากการระบุตำแหน่งด้วยวิธีการมอนติคาร์โลแบบปรับตัวและข้อมูลสิ่งกีดขวางจากการทำ การคำนวณแผนที่นำข้อมูลทั้งหมดมาคำนวณเป็นเส้นทางการเดินรถด้วยไลบรารี่ผู้วางแผนแล้วส่งค่าความเร็วที่รถจะต้องเคลื่อนที่ไปยังอาดูโน่ จนรถเดินทางถึงจุดหมายจึงสั่งรถหยุด เมื่อโปรแกรมทำงานได้ถูกต้องทั้งหมดจึงทำการทดลองสร้างแผนที่สั่งให้รถเดินทางไปยังจุดหมายต่างๆ ในแผนที่ ด้วยรูปแบบต่างๆกันหลายแบบเพื่อเก็บข้อมูลผลการทดลองและสรุปผลการพัฒนาโปรแกรมต่อไป

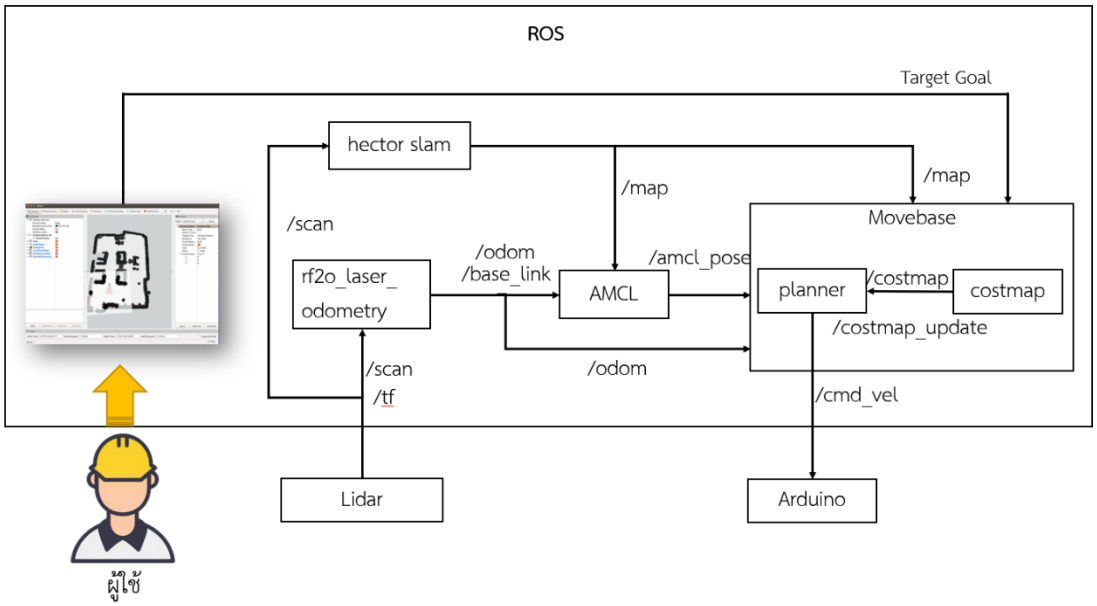


ภาพที่ 3.2 แผนภาพแสดงฮาร์ดแวร์ของการประยุกต์ใช้ระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

จากภาพที่ 3.2 แผนภาพแสดงระบบในส่วนของฮาร์ดแวร์ของการประยุกต์ใช้ระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม จากภาพไลดาร์เชื่อมต่อกับราสเบอร์รี่พายผ่านยูเอสบีพอร์ต (USB Port) อาดูโน่เชื่อมต่อกับราสเบอร์รี่พายยูเอสบีพอร์ต โดยใช้คำสั่ง rosrun rosserial_python serial ในการเชื่อมพอร์ตเข้ากับระบบปฏิบัติการหุ่นยนต์ ซึ่งมีการรับส่งข้อมูลโดยการแปลงยูเอสบีพอร์ตเป็น Serial Port และมีการส่งสัญญาณ PWM ไปยังวงจรขับเคลื่อนมอเตอร์เพื่อควบคุมความเร็วรอบมอเตอร์ทั้งสองตัว และส่งสัญญาณส่งออกของดิจิตอลไปที่ช่อง IN1 และ IN2 ของตัวขับมอเตอร์เพื่อกำหนดทิศทางการหมุนของมอเตอร์

ตารางที่ 3.1 การเชื่อมต่อสัญญาณราสเบอรี่พายและอาดูโน่

Arduino	Motor Drive
PIN4	IN1 Motor Drive 1
PIN5	IN2 Motor Drive 1
PIN6	PWM Motor Drive 1
PIN7	IN1 Motor Drive 2
PIN8	IN2 Motor Drive 2
PIN9	PWM Motor Drive 2

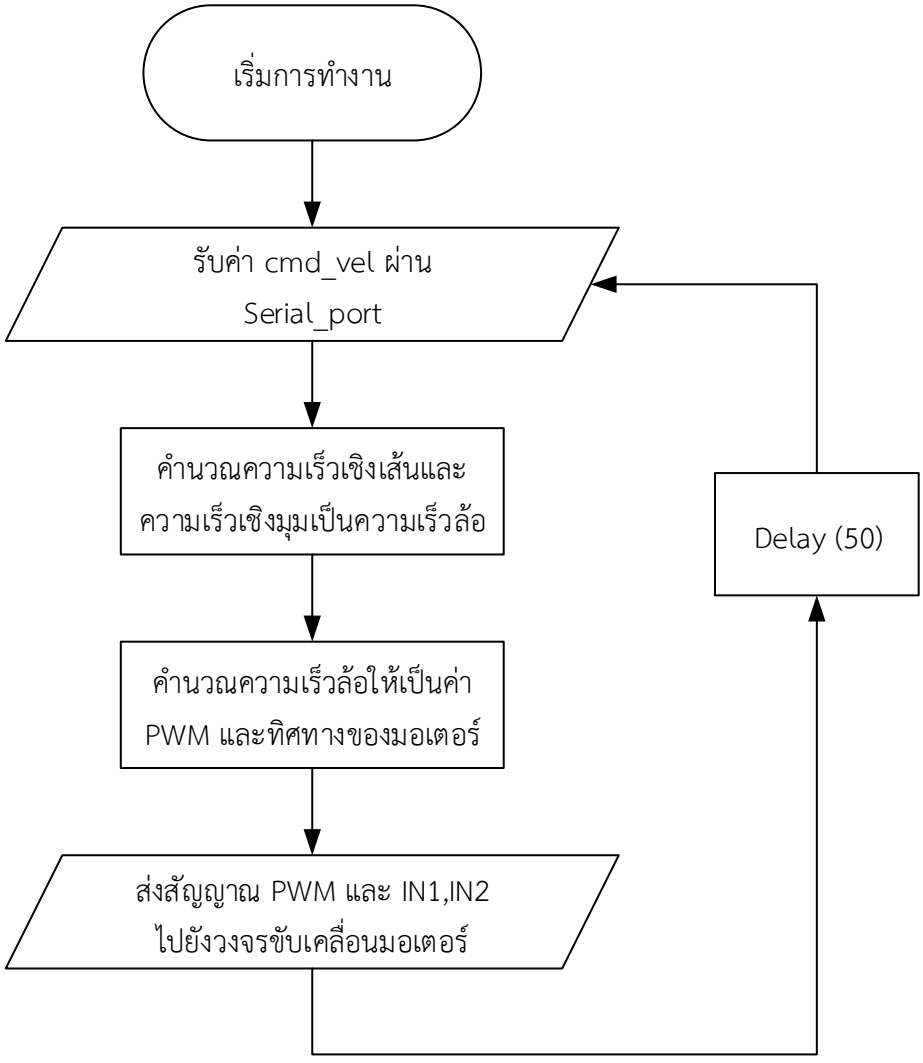


ภาพที่ 3.3 แผนผังการทำงานของระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

จากภาพที่ 3.3 ผู้ใช้งานเข้ามาสั่งรถให้เคลื่อนที่ไปจุดปลายทางโดยการรีโมตเข้ามาใช้งานผ่านโปรแกรมอาร์วิช โปรแกรมอาร์วิชจะส่งพิกัดของแผนที่ไปให้จุดต่อการเคลื่อนย้ายฐาน จากนั้นจุดต่อการเคลื่อนย้ายฐาน รับหัวข้อ map , amcl_pose และ odom มาคำนวณหาเส้นทางการเคลื่อนที่และหลบหลีกสิ่งกีดขวาง คำนวณเป็นความเร็วในการเคลื่อนที่เชิงเส้นและเชิงมุมของรถเพื่อไปยังจุดหมายส่งให้อาดูโน่ในหัวข้อ cmd_vel โดยที่ map ได้มาจากจุดต่อ hector_slam ซึ่งข้อมูลในหัวข้อ scan มาจากอุปกรณ์ไลดาร์ที่ส่งข้อมูลเข้าสู่ระบบปฏิบัติการหุ่นยนต์ในหัวข้อ scan และ tf

amcl_pose คือข้อมูลตำแหน่งปัจจุบันของรถบนแผนที่ ได้จากจุดต่อโดยการระบุตำแหน่งด้วยวิธีการมอนติคาร์โลแบบปรับตัว ซึ่งนำข้อมูลจากหัวข้อหัวข้อ map , odom และ base_link มาคำนวณหาตำแหน่งของรถบนแผนที่ odom คือข้อมูลการเคลื่อนที่ของรถ ได้จากจุดต่อ rf2o_laser_odometry ซึ่งนำข้อมูลจากหัวข้อ scan และ tf จากไลดาร์มาคำนวณเป็นข้อมูลการเคลื่อนที่ของรถและส่งออกไปในหัวข้อ odom และ base_link

3.1 การเขียนโปรแกรมควบคุมรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์ด้วยอาดูโน่



ภาพที่ 3.4 กระบวนการทำงานของอาดูโน่

จากภาพที่ 3.4 เริ่มการทำงานโดยการรับค่า cmd_vel ผ่าน Serial Port จากนั้นคำนวณความเร็วเชิงเส้นและความเร็วเชิงมุมให้เป็นความเร็วล้อด้วยสมการ $v = v_{linear} + \omega R$ ต่อมาคำนวณความเร็วล้อให้เป็นค่า PWM และทิศทางของมอเตอร์ ส่งสัญญาณ PWM และ IN1,IN2 ผ่านพอร์ตเอนกประสงค์ (General Purpose Input/Output :GPIO) ไปยังวงจรขับเคลื่อนมอเตอร์ทั้ง 2 มอเตอร์

3.2 การเขียนโปรแกรมสร้างแผนที่ด้วย Hector SLAM

ก่อนการเคลื่อนที่ต้องเริ่มจากการสร้างแผนที่และการที่ผู้วิจัยจะสร้างแผนที่ได้นั้นใช้ไลดาร์ ซึ่งการทำงานของไลดาร์คือการส่งแสงเลเซอร์ออกไปและจับเวลาที่แสงสะท้อนกลับมา เพื่อนำมาหาระยะทางของสิ่งกีดขวางตามสมการดังนี้

$$S = C\frac{t}{2}$$

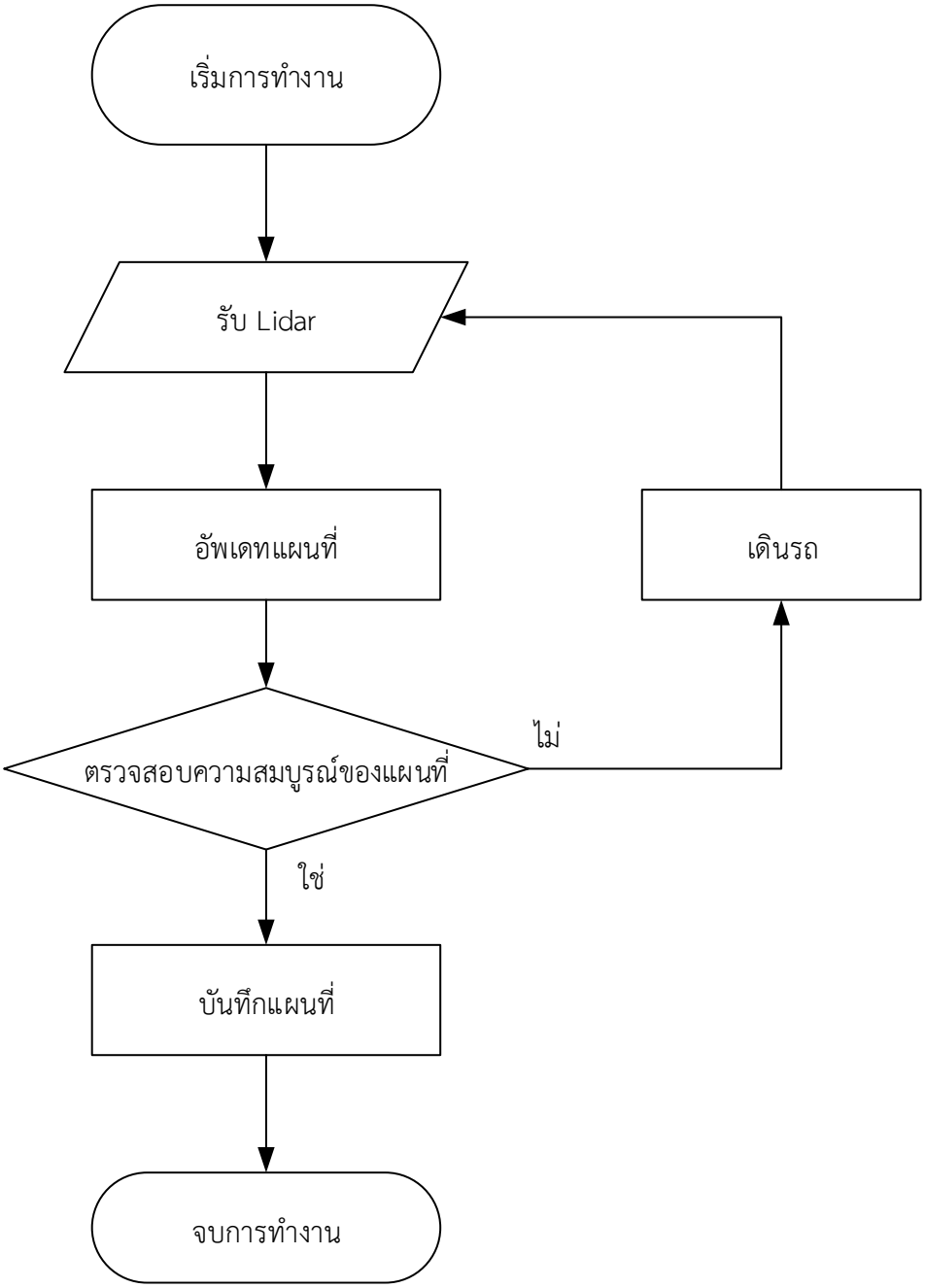
เมื่อ C คือ ความเร็วของแสง

t คือ เวลาการเดินทางของแสงมีการเดินทางไปและกลับ

S คือ ระยะทางของสิ่งกีดขวาง

ผู้วิจัยเลือกใช้งาน Hector SLAM เนื่องจาก Hector SLAM นั้นไม่ใช้ออดโดมิททรี ในการสร้างแผนที่และสะดวกต่อการใช้งาน โดยการรับข้อมูลจากหัวข้อที่เผยแพร่จากไลดาร์ ทำการเปิดอาร์วิชเป็นโปรแกรมที่มาพร้อมการติดตั้งระบบปฏิบัติการหุ่นยนต์มีประโยชน์สำหรับการดูข้อมูลจากหัวข้อที่เผยแพร่ต่างๆ เมื่อทำการเปิดอาร์วิชขึ้นมาแล้วจะเห็นแผนที่ปรากฏขึ้นมาสังเกตุว่าแผนที่ที่ถูกสร้างขึ้นมาเป็นแผนที่ที่สมบูรณ์หรือไม่ หากยังไม่สมบูรณ์จะทำการสั่งให้รถเคลื่อนที่เพื่อเก็บข้อมูลเพิ่มเติม เมื่อแผนที่สมบูรณ์แล้วจะทำการบันทึกแผนที่โดยมีไฟล์ที่บันทึก 2 ไฟล์คือไฟล์นามสกุล pgm และ yaml ซึ่งเรียกว่า Param โดยนามสกุล pgm จะเป็นภาพของแผนที่ส่วนนามสกุล yaml เป็นการบอกพิกัดของแผนที่ซึ่งภาพที่ 3.5 แสดงตัวอย่างแผนที่นามสกุล pgm

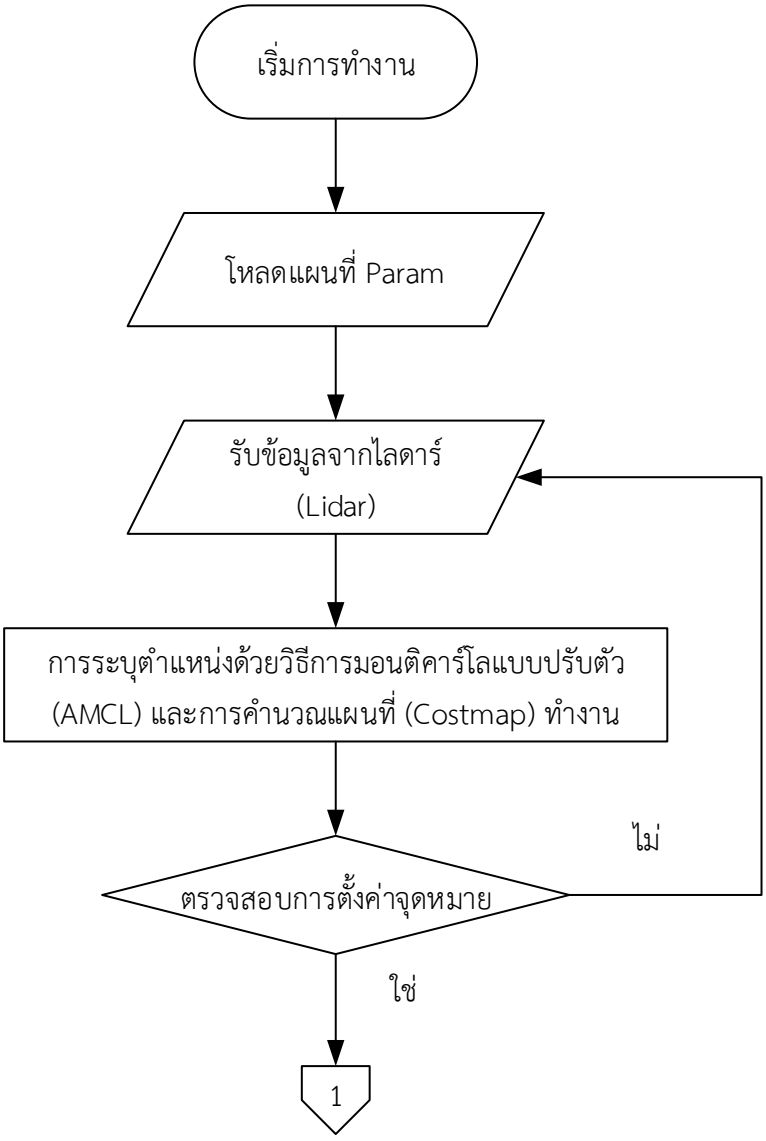
ภาพที่ 3.5 ตัวอย่างแผนที่นามสกุล pgm

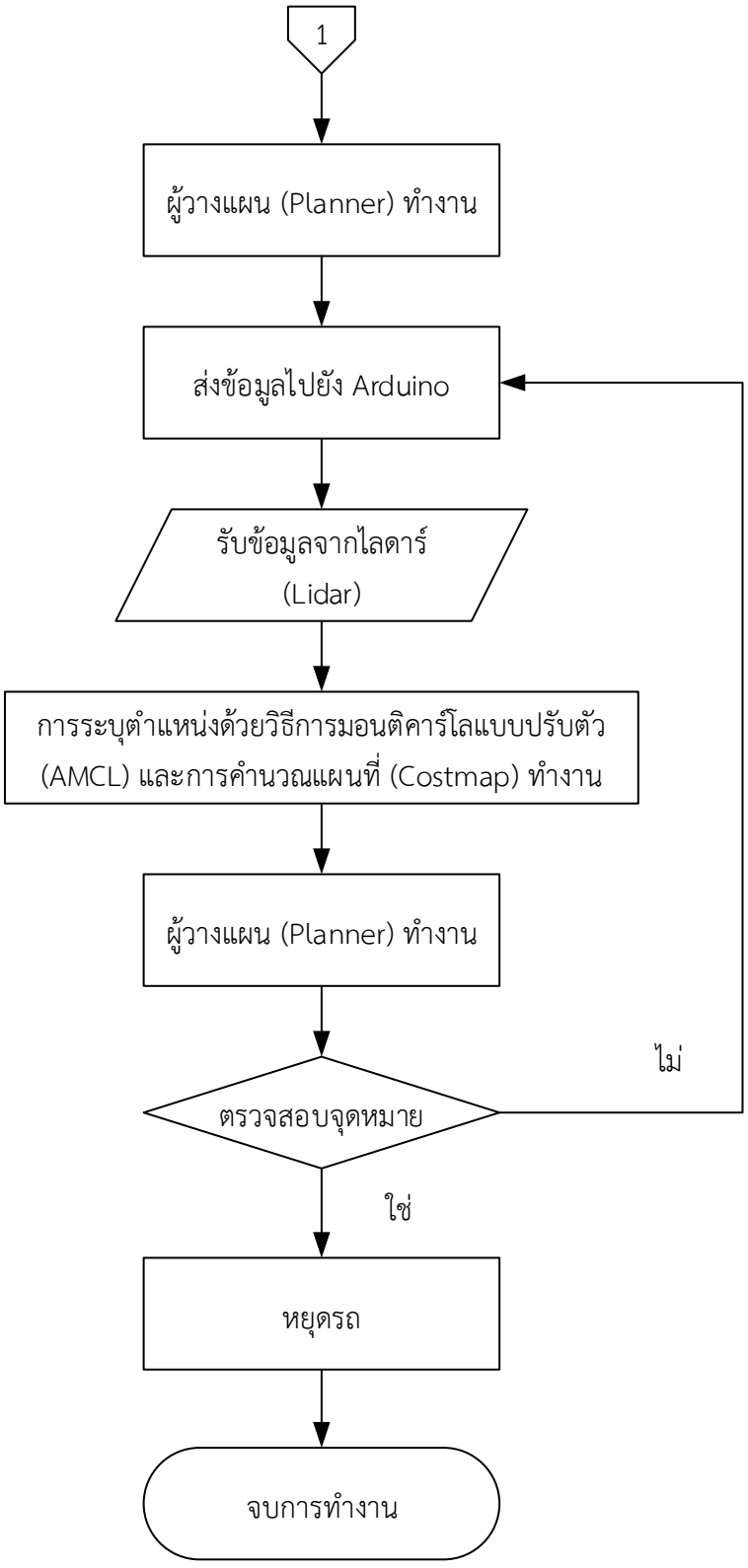


ภาพที่ 3.6 กระบวนการสร้างแผนที่จาก Hector SLAM

จากภาพที่ 3.6 เริ่มการทำงานโดยโปรแกรมการระบุตำแหน่งพร้อมกับการสร้างแผนที่รับข้อมูลจากหัวข้อชื่อ scan จากไลดาร์ทำการคำนวณข้อมูลจากไลดาร์นำมาอัพเดทและแสดงผลแผนที่บนอาร์วิช หากแผนที่ครอบคลุมจุดหมายทั้งหมดที่ต้องการให้รถเคลื่อนที่ทำการบันทึกแผนที่ด้วยคำสั่ง rosrun map_server หากแผนที่ไม่สมบูรณ์ให้สั่งรถเคลื่อนที่เพื่อเก็บแผนที่เพิ่มเติมจนกว่าแผนที่จะเสร็จสมบูรณ์

3.3 การเขียนโปรแกรมสั่งงานรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์เพื่อเคลื่อนที่ไปยังจุดหมายและหลบหลีกสิ่งกีดขวาง





ภาพที่ 3.7 กระบวนการเดินของรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์จากจุดเริ่มต้นไปถึงจุดปลายทาง

จากภาพที่ 3.7 เริ่มการทำงานโหลดแผนที่ที่ได้จาก Hector SLAM และพารามิเตอร์สำหรับทำการคำนวณแผนที่รับข้อมูลจากหัวข้อที่เผยแพร่ชื่อว่า scan และทรานสฟอร์มจากไลดาร์ หลังจากนั้นทำการเปิดโปรแกรมการระบุตำแหน่งด้วยวิธีการมอนติคาร์โลแบบปรับตัวตามด้วยโปรแกรมการคำนวณแผนที่ ซึ่งตอนนี้ระบบปฏิบัติการหุ่นยนต์จะทราบตำแหน่งของรถว่าอยู่จุดไหนของแผนที่พร้อมทั้งมองเห็นสิ่งกีดขวางจากการคำนวณด้วยโกลเบิลโคสต์แมฟและโลเคิลโคสต์แมฟ แสดงผลบนอาร์วิช เมื่อจุดหมายถูกกำหนดโปรแกรมจะเริ่มคำนวณเส้นทางด้วยผู้วางแผนและส่งข้อมูลความเร็วเชิงเส้นและความเร็วเชิงมุมไปในหัวข้อชื่อ cmd_vel ผ่าน Serial port ไปยังอาดูโน่ จากนั้นรับข้อมูลจากหัวข้อชื่อ scan และทรานสฟอร์มจากไลดาร์อีกครั้งเพื่อคำนวณหาตำแหน่งของรถในแผนที่ด้วยการระบุตำแหน่งด้วยวิธีการมอนติคาร์โลแบบปรับตัวพร้อมทั้งคำนวณสิ่งกีดขวางและเส้นทางใหม่เมื่อรถเคลื่อนที่ถึงจุดหมายจะหยุดและจบการทำงาน

3.4 การใช้งานของระบบปฏิบัติการหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

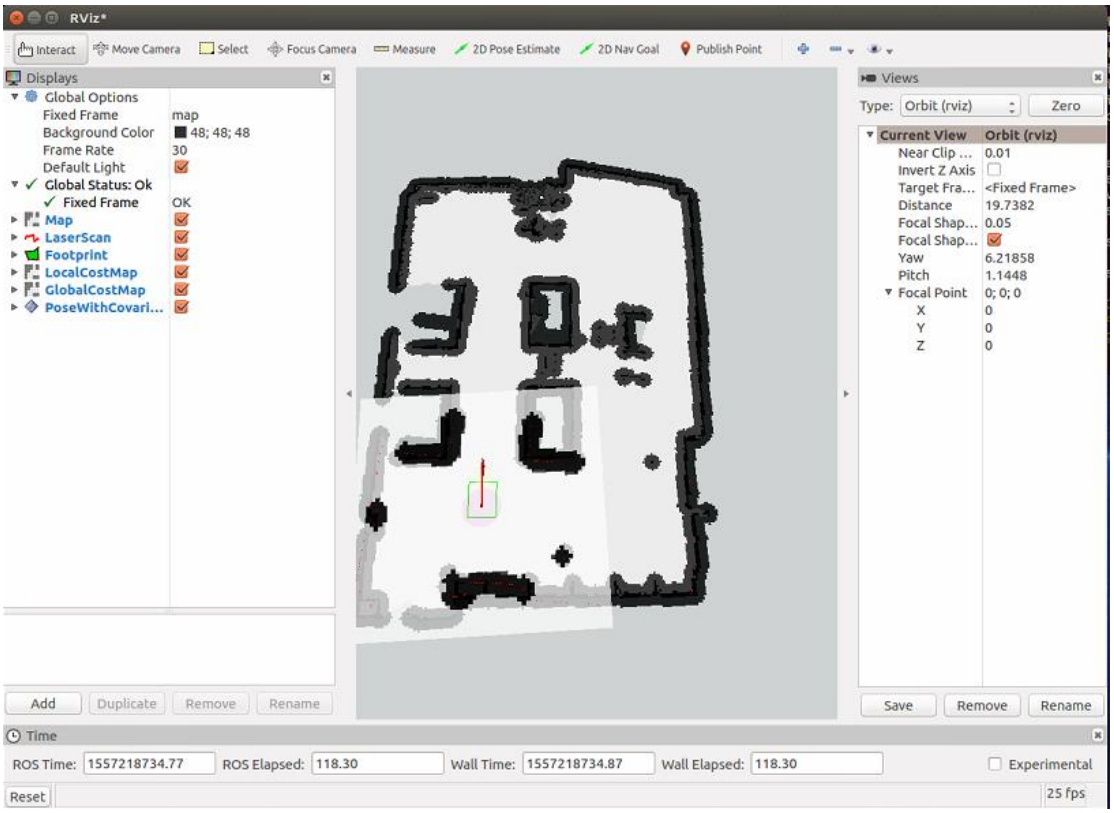
3.4.1 การสร้างแผนที่สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

เริ่มจากเปิดการทำงานของ Hector SLAM โดยใช้คำสั่ง $ roslaunch hector_slam tutorial.launch เมื่อเปิดการทำงานแล้วสั่งให้หุ่นยนต์เดินไปตามจุดหมายทั้งหมดที่ต้องการให้รถเคลื่อนที่โดยใช้คีย์บอร์ดผ่านการควบคุมระยะไกล เมื่อแผนที่ครอบคลุมจุดหมายทั้งหมดที่ต้องการให้รถเคลื่อนที่แล้วทำการบันทึกภาพแผนที่โดยใช้คำสั่ง $ rosrun map_server map_server -f my_map จากภาพที่ 3.8 เป็นแผนที่ที่ครอบคลุมจุดหมายทั้งหมดที่ต้องการให้รถเคลื่อนที่

ภาพที่ 3.8 ตัวอย่างแผนที่สำหรับรถขนส่งสินค้าอัตโนมัติ

3.4.2 การใช้งานรถขนส่งสินค้าอัตโนมัติ

เริ่มการทำงานโดยใช้คำสั่ง $ roslaunch move_base.launch จะปรากฏหน้าต่างของโปรแกรมอาร์วิชขึ้นมาดังภาพที่ 3.9 เมื่อกดปุ่มชื่อว่า 2D Nav Goal ลากใน Grid โดยหันทิศทางไปในทิศปลายทางที่ต้องการให้รถเคลื่อนที่ไปยังปลายทาง จะปรากฏภาพลูกศรของปลายทางที่ต้องการให้รถเคลื่อนที่ขึ้นมาในโปรแกรมอาร์วิช



ภาพที่ 3.9 หน้าต่างในโปรแกรมอาร์วิช

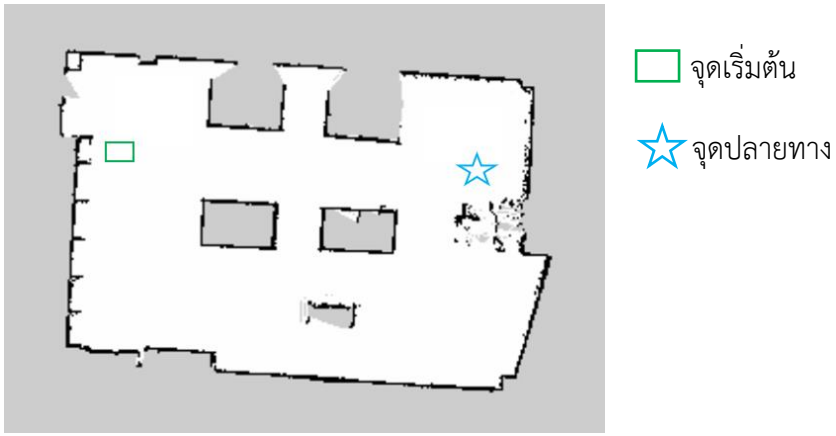
3.5 การทดลองหุ่นยนต์สำหรับรถขนส่งสินค้าอัตโนมัติในโรงงานอุตสาหกรรม

การทดลองเพื่อเพิ่มประสิทธิภาพของรถขนส่งสินค้าอัตโนมัติโดยสามารถเคลื่อนที่ไปยังจุดหมายด้วยการนำทางจากระบบปฏิบัติการหุ่นยนต์แทนการใช้งานเส้นแถบแม่เหล็กและคำนวณเส้นทางการเคลื่อนที่ใหม่เมื่อมีสิ่งกีดขวางซึ่งแบ่งการทดลองออกเป็น 2 ส่วนดังนี้

3.5.1 การทดลองความเร็วของรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์

เริ่มจากการเก็บข้อมูลแผนที่ของห้องที่จะทำการทดลองด้วย hector SLAM ซึ่งแผนที่จะครอบคลุมจุดหมายทั้งหมดที่ผู้วิจัยต้องการให้รถเคลื่อนที่ จากนั้นทดลองให้รถเดินไปยังจุดหมายที่กำหนดในแผนที่จำนวน 20 รอบ โดยให้จุดเริ่มต้นของรถอยู่ในตำแหน่งเดิม จากนั้นบันทึกระยะทางการเคลื่อนที่ของรถขนส่งสินค้าอัตโนมัติและเวลาที่รถใช้ในการเคลื่อนที่ เพื่อนำมาคำนวณความเร็วเฉลี่ยของรถขนส่งสินค้าอัตโนมัติโดยใช้สมการคือ

$$\text{ความเร็วเฉลี่ยของรถ} = \frac{\text{ระยะทางการเคลื่อนที่ของรถ}}{\text{เวลาที่รถใช้ในการเคลื่อนที่}}$$



ภาพที่ 3.10 แผนที่ห้องที่ใช้ในการทดลองความเร็ว

จากภาพที่ 3.10 สภาพแวดล้อมของห้องที่ใช้ในการทดลองความเร็วของรถขนส่งสินค้าอัตโนมัติ มีลักษณะเส้นทางการเคลื่อนที่ไม่ซับซ้อนและไม่มีสิ่งกีดขวางในการเคลื่อนที่ของรถขนส่งสินค้าอัตโนมัติ

3.5.2 การทดลองค้นหาเส้นทางใหม่และการหลบหลีกสิ่งกีดขวางของรถขนส่งสินค้าอัตโนมัติแบบนำทางด้วยระบบปฏิบัติการหุ่นยนต์

เริ่มจากผู้วิจัยจะสั่งให้รถเคลื่อนที่ไปตามจุดต่างๆ บนแผนที่ โดยที่มีสิ่งกีดขวางตามเส้นทางในจุดที่แตกต่างกันไป โดยมีจุดปลายทางอยู่ 12 ตำแหน่งและจุดเริ่มต้นจำนวน 5 ตำแหน่ง และมีรูปแบบทั้งหมด 20 รูปแบบ โดยแต่ละรูปแบบวางสิ่งกีดขวางเพื่อทำการทดสอบการทำงาน ซึ่งสัญลักษณ์ดังต่อไปนี้

ภาพที่ 3.11 ตัวอย่างแผนที่ที่ใช้ในการทดลองค้นหาเส้นทางใหม่และการหลบหลีกสิ่งกีดขวาง

จากภาพที่ 3.11 กรอบสี่เหลี่ยมแสดงถึงจุดเริ่มต้นและดาวคือจุดปลายทางของรถ โดยที่มีจุดห้ามผ่านเกิดขึ้น เส้นประแสดงเส้นทางของรถในการเคลื่อนที่ไปยังจุดปลายทาง ซึ่งผู้วิจัยจะบันทึกระยะทางของการทดลองแบบไม่มีสิ่งกีดขวางและมีสิ่งกีดขวางมาเปรียบเทียบกัน เพื่อให้ทราบระยะทางที่แตกต่างกันของการเคลื่อนที่ของรถ สภาพแวดล้อมของห้องที่ใช้ในการทดลองค้นหาเส้นทางใหม่และการหลบหลีกสิ่งกีดขวางมีสภาพแวดล้อมคล้ายโรงงานอุตสาหกรรมมีเครื่องมือเครื่องจักรและชั้นวางของ โดยมีลักษณะเส้นทางการเคลื่อนที่ซับซ้อน โดยกำหนดจุดเริ่มต้นและจุดหมายได้หลายตำแหน่ง