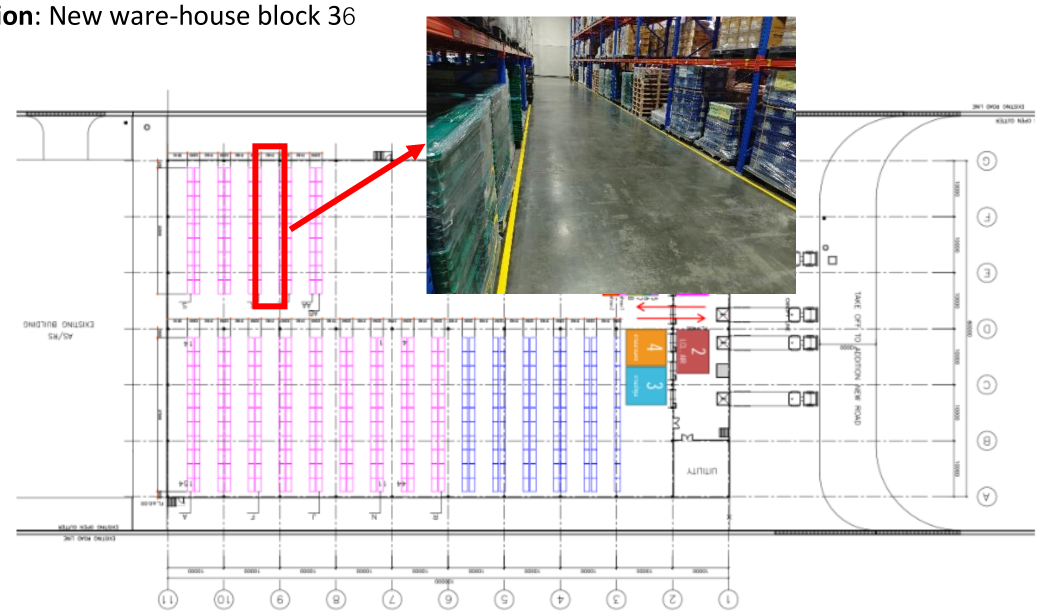
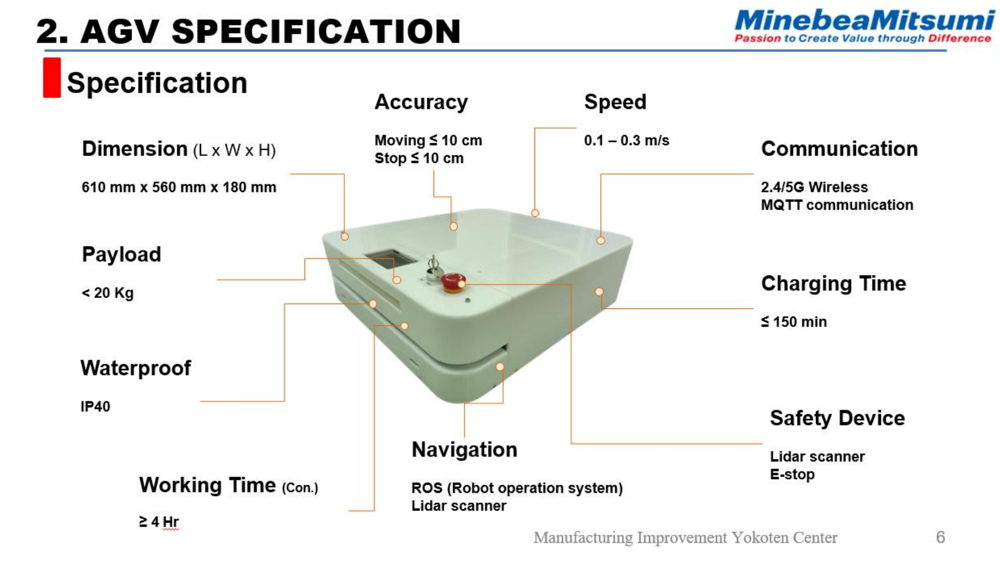
Objective: To test opportunity for AGV surveying in New ware-house LopBuri

Location: New ware-house block 36



Equipment: AGV MICKY50

Specification:



Testing Condition:

1. Create map by using ROS gmapping and slam [1].
2. Marking way point and automatically moving AGV to goal point.
3. Avoidance Forklift standing in the front of AGV (AGV move to rare direction of Forklift).
4. Avoidance Forklift standing in the front of AGV (AGV move to front direction of Forklift).
5. Avoidance Forklift that moving in the front of AGV (AGV move to front direction of Forklift).

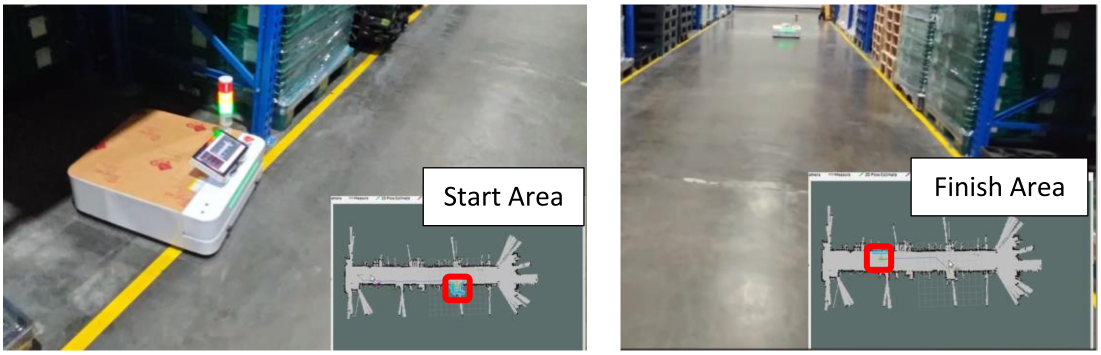
Testing Result:

1. Create map by using ROS Gmapping and slam[1]

Picture 1. Map New Ware-House Block 36

From the picture 1. AGV successfully creates a map of the warehouse area using Gmapping and SLAM. [1]. The left picture shown that the black area is the obstacle that AGV cannot move and gray area is a moving area that AGV can move to it.

2. Marking way point and automatically moving AGV to goal point



Picture 2. Marking way point result (https://youtu.be/Vt0BKcrXfkE)

*****Video link, people who have link only can see this video**

From the picture 2. AGV is able to move from start area to finish area is correctly. The accuracy of distance < 10 cm and angle < 10 degree as specification.

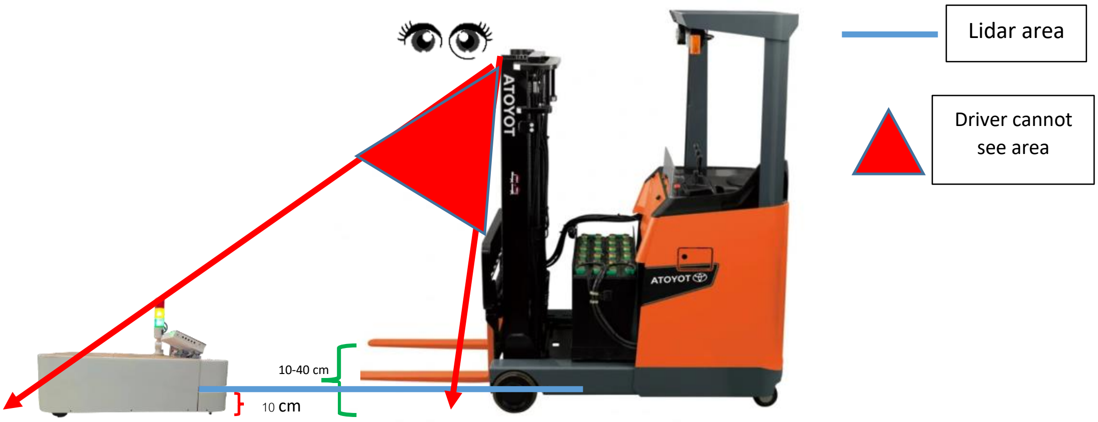
3. Avoidance Forklift standing in the front of AGV (AGV move to rare direction of Forklift)

Picture 3. Avoidance Forklift standing in the front of AGV (https://youtube.com/shorts/Tyl7NCnSfH8?feature=share)

*****Video link, people who have link only can see this video**

From the picture 3. AGV can avoid stationary forklifts by moving around them from the rear direction effectively with distance about 10-30 cm that is acceptable distance for working.

4. Avoidance Forklift standing in the front of AGV (AGV move to front direction of Forklift).



Picture 4. Avoidance Forklift standing in the front of AGV

From the picture 4. Distance between LIDAR sensors of AGV (**blue line**) is under the forks of the forklift (10 cm from floor for AGV and 10-40 cm for Forklift). Therefore AGV cannot see forklift while working and if AGV and forklift move towards each other it will cause a collision. In Addition, Eye distance of driver about 1.5-2 meter from the front of Forklift. If AGV move into this area that driver cannot see AGV, it may be cause collision too.

5. Avoidance Forklift that moving in the front of AGV (AGV move to front direction of Forklift) we **don't** test this case because result from 4.

Conclusion:

From the test result we found that AGV can move on testing area (New Ware-House block 36) but when co-working with Forklift it can move through Forklift from rear of forklift only. AGV cannot move through the front of forklift because height between AGV's LIDAR sensor and forks of Forklift, eye distance of driver. It will cause a collision.

Considerations:

1. Set area that AGV can move and without Forklift.
2. Test additional area within New Ware-House.
3. Test additional condition with people by move towards each other.
4. Maximum speed of AGV is 0.3 m/s that less than Forklift (about 1-5 m/s).
5. Maximum speed of AGV is 0.3 m/s that less than worker (about 0.5-1 m/s), is it acceptable?

Reference:

[1]

Gmapping and SLAM in ROS

SLAM (Simultaneous Localization and Mapping)

SLAM is a technique used in robotics and autonomous systems for building a map of an unknown environment while simultaneously keeping track of the robot's location within that environment. The two main components of SLAM are:

Localization: Determining the position and orientation (pose) of the robot within the map.

Mapping: Creating a map of the environment.

SLAM algorithms use sensor data (e.g., from LIDAR, sonar, or cameras) to update both the map and the robot's pose in real-time. The goal is to ensure that the robot can navigate accurately even in previously unexplored areas.

Gmapping in ROS

Gmapping is a widely used SLAM algorithm in the ROS (Robot Operating System) ecosystem. It is an implementation of a laser-based SLAM algorithm based on particle filters, which is designed to create 2D occupancy grid maps from laser scan data. Here are the key components and steps involved in using Gmapping in ROS:

- **Sensor Setup**: Gmapping requires laser scan data, typically provided by a LIDAR sensor. The sensor must be properly calibrated and configured in the ROS environment.
- **Launch File**: A ROS launch file is used to start the Gmapping node along with other necessary nodes like the robot state publisher, the transform (tf) tree, and sensor drivers.
- **Gmapping Node**: The slam_gmapping package provides the Gmapping node. When started, it subscribes to topics for laser scan data and odometry information.
- **TF Tree**: Gmapping relies on the tf (transform) package to keep track of the robot's position and orientation in the world frame. The tf tree should accurately represent the relationship between the robot's base frame, sensor frames, and the world frame.
- **Map Building**: As the robot moves around, Gmapping uses the laser scan data to create an occupancy grid map, which represents the environment. The map is continuously updated based on new sensor data and robot movements.
- **Parameter Tuning**: Gmapping offers various parameters that can be tuned to improve the quality of the generated map, such as the number of particles, linear and angular update thresholds, and the size of the map

Conclusion

Gmapping is a powerful tool within the ROS ecosystem for implementing SLAM with 2D laser scanners. By accurately mapping an environment and localizing within it, robots can perform complex navigation tasks in unknown or dynamic settings. The combination of Gmapping and ROS provides a robust framework for developing autonomous robotic systems.