Polytechnic university of Turin

Department of control and computer engineering (DAUIN)

Master of Science in mechatronic engineering

Master Degree Thesis

"Development and Establishment of a Software Quality Test Procedure for ROS Software Releases Controlling Autonomous Transport Robots"

Insystems Automation GmbH
Berlin, Germany

University Tutor:
Prof.Marcello Chiaberge

Company Tutor:
Mr.Henry Stubert

Author:
Afshin Arjmand

April 2018

Abstract

Autonomous Guided Vehicle systems (AGVs) are widely used for transporting material in manufacturing and warehouse applications. AGVs offer many advantages over other systems which makes them unique and most usable vehicles in manufactures. This systems have complex design due to different purpose of using from customer requirements.
Nowadays testing an AGV is one of the challenging part of production. The AGV is design, programming by the robot developers, so, to see the robot is working properly, and as expected, it needs many hard test to be approve by the company. During my thesis, I developed a test process for the AGVs in the company that is reusable and repeatable for all kind of AGVs. In chapter1 You are getting to know the prototype of an AGV that I have done testing and main devices in an AGV explained then I defined the test method that divided in three test levels: static/dynamic context, sensing/Acting and the knowledge of a tester that could influence the testing. In chapter2.1 There is a modeling hierarchy, which defined due to the state of art of testing in chapter1.3. The visual and I/O checks and calibration of devices in the AGV are in test level 1 which means the AGV is in static context with sensing. The next level is testing in a static/dynamic context and sensing/Acting that the tests of Driving Behavior, Fine Positioning, Battery Loop and Load/Unload are testing. At the end of the test process the customers have rights to choose their AGVs which are suitable for their manufacture. The most important choice for them is to choose the weight that an AGV can handle for carrying their product.

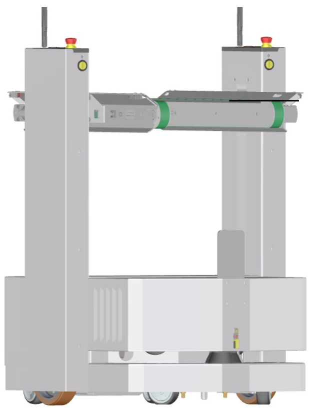


Figure 1: AGV

Acknowledgment

This thesis would not be the same without the generosity and contribution of a large number of people. First and foremost, I wish to thank my advisor, Mr. Henry Stubert, director of InSystems company, for its trust and for the possibility of developing this work on the challenging field of Battery Management Systems for AGVs. He has been supportive since the days I began working ROS Team.
I am very grateful to my Tutor at polytechnic university of Turin, Professor Marcello Chiaberge, I appreciate very much his suggestions and the time he spent to support me, helping to achieve this important milestone.
Special thanks to Mr. Jan Stefan Zernickel Project Manager at Insystems who gave me extensive guidance regarding many practical issues.
I am also very grateful to all my friends and colleagues of the ROS team and the workshop team who were always supportive.
Finally, last but not least, I would like to express my greatest gratitude to my family and also my girlfriend for their continuous support.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Mobile autonomous robots are progressively entering the mass market. Thus, manufacturers have to perform tests on series of robots. Therefore, tests should be repeatable and as much automated as possible. Tests are also performed for purpose of repairing robots. This calls for reusing tests.

Mobile autonomous robots are finding their way to the mass market. Thus, robot manufacturers have to switch from the prototype building mode to industrial production line. Multiple instances of the same robot have to be built fast enough to meet the customers demand. These robots are supposed to be exact copies of the original prototype, and thus exhibit the very same behavior.

Since tests are likely to be conducted on a large number of robots, they should be repeatable. Running a given test several times should consist in making the tested robots perform the same actions under the same circumstances, in the same environment. Automation can help ensuring repeatability, while speeding up the test process.

Testing robots is also important for repair. Technicians have to test a robot to diagnose the actual defect and figure out the source of dysfunction. Repair should rely on tests to identify which robot behavior or parts do not comply with the specification.

Test reuse can be pushed a step further, by focusing on behaviors rather than on robots. Indeed, behaviors matter more that the way they are implemented, at least from the customer point of view. Thus, tests that evaluate the way robots conform to a behavior, should be reusable and apply to different robots, even if they are built out of different parts.

1.1 Insystems Automation GmbH

InSystems Automation develops innovative automatic solutions and special machines for production, material flow and quality tests. Their range of services includes all tasks, from creating the specification sheet, to electrical projects, installation and programming and commissioning, maintenance and service. Their customer have a single and professional contact person, from planning to the completion of machines and plants. The company was founded 1999 by the managing directors Henry Stubert and Torsten Gast and grew constantly since. Meanwhile, more than 50 employees work at InSystems. The company is located in the science center Berlin-Adlershof and has offices, a workshop, an online shop and a showroom. Since 2012, InSystems specialized on the production of autonomous navigating transport robots, which are designed for loads from 30 to 1.000 kg, according to the customer request, and implemented as a fleet into an existing production control. The transport robots are developed under the name proANT. The company provide a variety of facilities to work on, give you enough time to get along well with your tasks in different environments.

During my Thesis I work with most efficient team which they are developers and experties of the Robot Development team helped to understand perfectly the system of an AGV. Being in this team gives me ideas to provide to the company which they are happy to hear about different ideas, then discussing about it. If something wrong is in your work they generously gives suggestions, so in this environment of work, you are free to do your tasks and learn from them every single day. The method of testing in the company is up-to-date like other companies.

The InSystems focusing on every single step of producing an AGV to make it high quality for the customer, each team are working along each other to not miss anything.

In section 1.2 the whole system of an AGV which is use as a prototype completely explained to understand better the system.

In chapter 2 the test procedure is defined as main test managing to declare our testing methods which is defined in section 1.3 .

The most important part that I work is that to develop and bring every test as measurable test so the other testers or technicians can use the same test methods to collect data and campare them to other versions.

1.2 System Overview

Automated guided vehicles (AGVs), have become increasingly sophisticated and used in many different industries. From warehouse to amusement rides, AGVs are used extensively and require to be tested accurately before selling it to the customer. Nowadays testing have an important role in manufacturing and each company has their own methods of testing which makes them special from other companies. Testing an AGV need cooperation of several teams which come to one hand. A tester needs to know how devices are working, effect, How to analyze data, and dependent on perspective view of a tester gives the final decision to the final release AGV. Figure 1.1 shows the prototype AGV which Robot Developmnet team is working on.

The robot that I was using is the prototype of the most popular AGV of the company. The aim is to put box on the robot and moving around. There are groups of AGVs in the company which have same mechanical body but different purposes.

Figure 1.1: Prototype AGV

1.2.1 Core

The TANK-700 Series fanless embedded system is powered by the Intel® 32nm mobile Core^{TM} i7/i5/i3 or Celeron® processor, uses the Intel® QM67 chipset and has 2.0 GB of DDR3 memory. It supports dual display via VGA and HDMI. Two SATA 6Gb/s, two USB 3.0 and four USB 2.0 ports provide flexible expansion options. Serial device connectivity is provided by six RS-232 and two RS-422/485 ports.

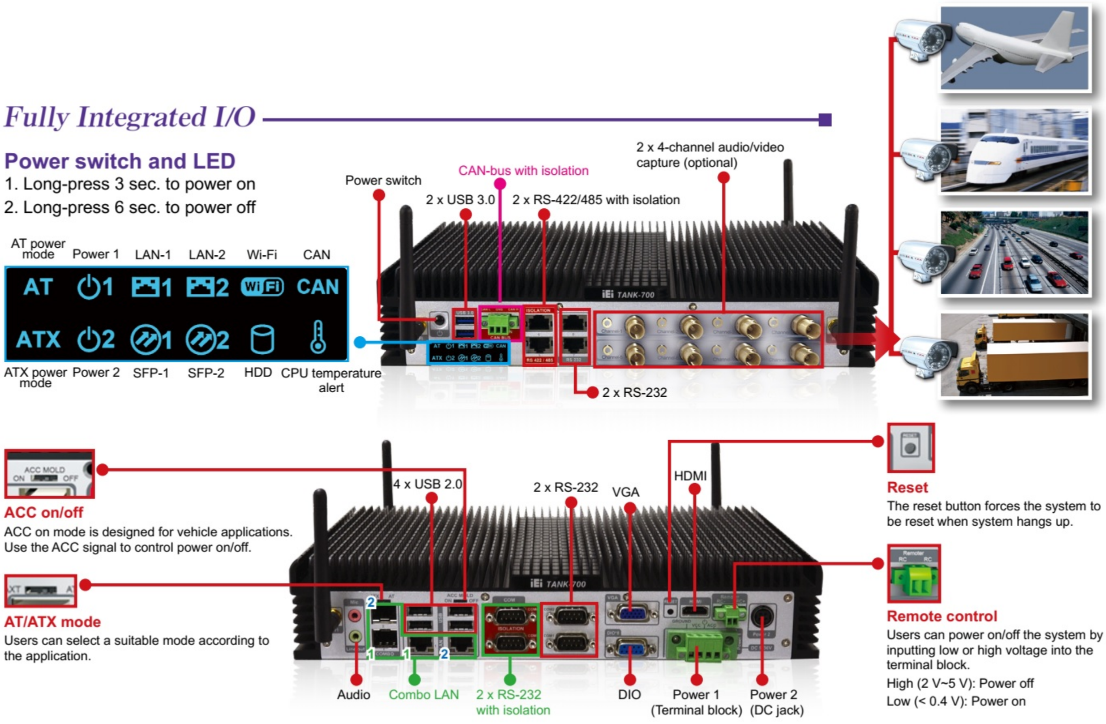


Figure 1.2: iEi Tank-700

1.2.2 Battery

The Energy Storage System (ESS) is a key component for Automated Guided vehicles (AGV). This includes the battery and all the management and monitoring systems that compose the Battery Management System (BMS). Lithium-based batteries are considered as the most advanced batteries technology, which can be designed for high energy or high-power storage systems.

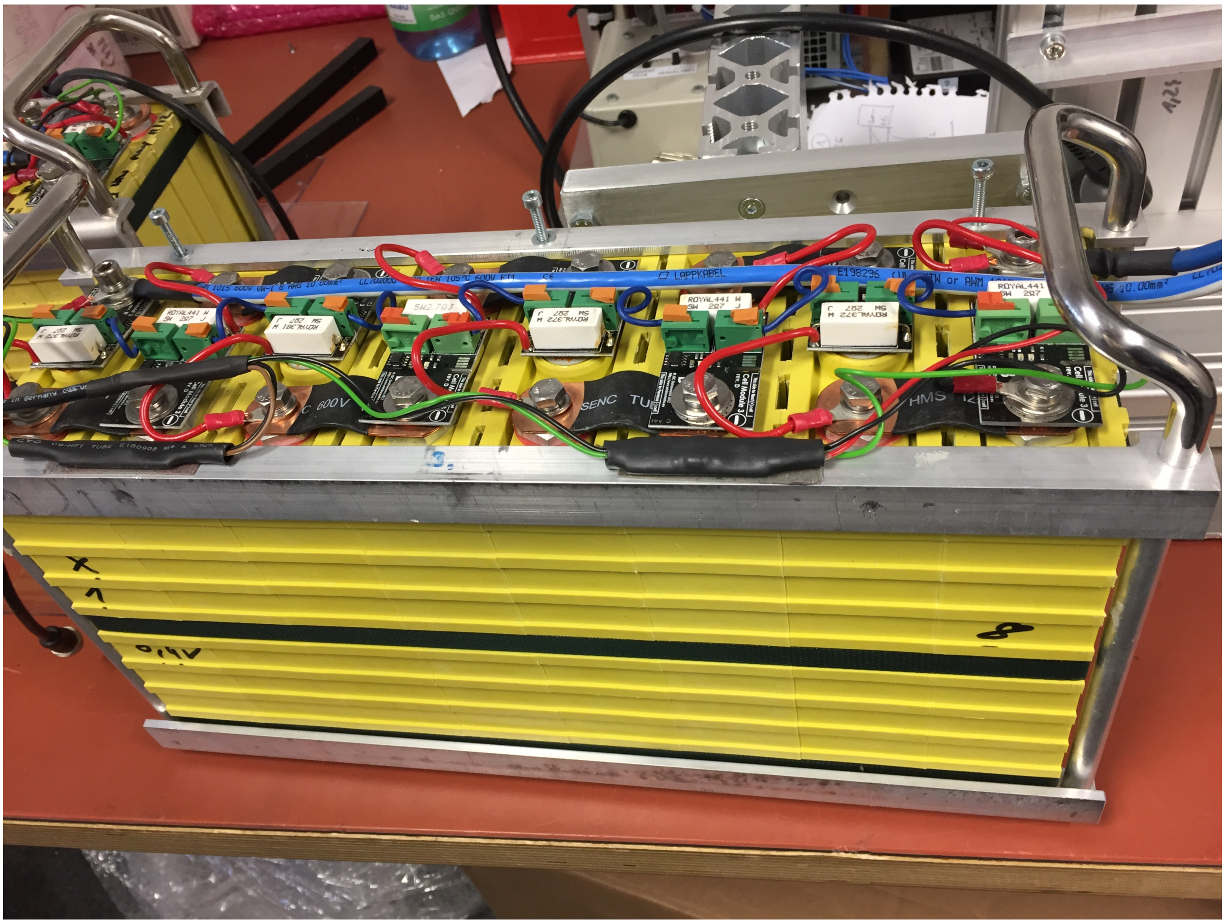


Figure 1.3: BMS

1.2.3 Laser Scanner

The AGV without the laser scanner has no sense to move. The laser scanner is responsible for scanning a map, detecting obstacles to avoid hitting them. The laser scanner sick S300 below is one of the most usable devices in AGVs.

Figure 1.4: S 300

1.2.4 3D Camera

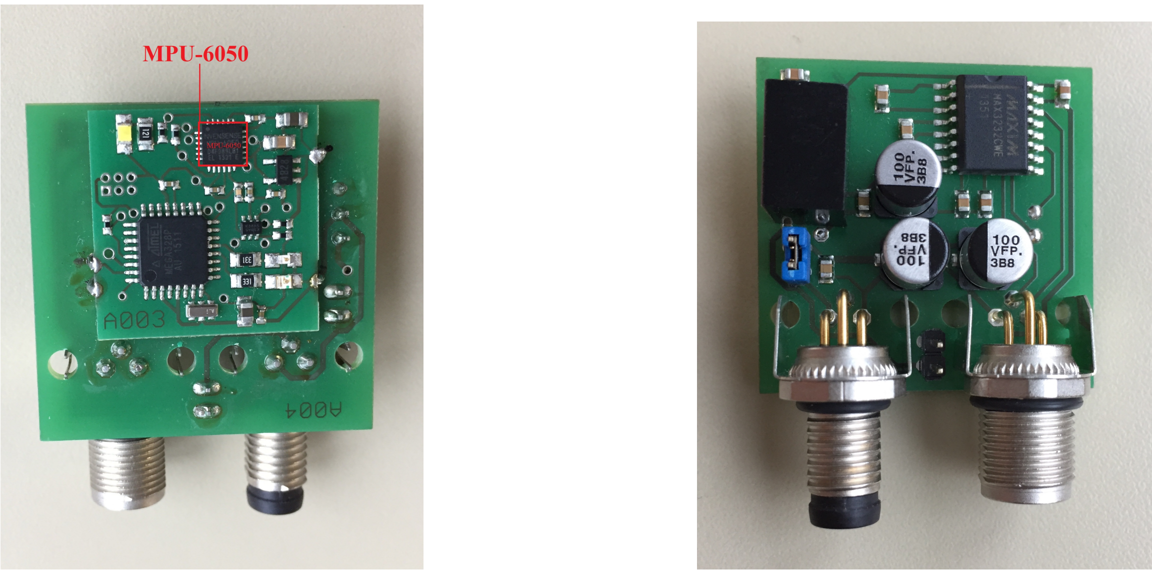
As long as the vehicle only has a 2D laser scanner it is limited to see a rather small part of its surroundings. For example, forklifts are too flat to be seen and tables are too high. This can cause crashes because the route is planned through obstacles.

Figure 1.5: 3D camera

1.2.5 Gyroscope

The MPU-6050^{TM} parts are the world's first MotionTracking devices designed for the low power, low cost, and high-performance requirements of smartphones, tablets and wearable sensors. The MPU-6050 incorporates InvenSense's MotionFusion^{TM} and run-time calibration firmware that enables manufacturers to eliminate the costly and complex selection, qualification, and system level integration of discrete devices in motion-enabled products, guaranteeing that sensor fusion algorithms and calibration procedures deliver optimal performance for consumers.

The MPU-6050 devices combine a 3-axis gyroscope and a 3-axis accelerometer on the same silicon die, together with an onboard Digital Motion Processor which processes complex 6-axis MotionFusion algorithms. The device can access external magnetometers or other sensors through an auxiliary master bus, allowing the devices to gather a full set of sensor data without intervention from the system processor.



(a) (b)

Figure 1.6: Gyroscope - MPU 6050

1.2.6 Network Protocol

The network structure is standalone configuration with the SCALANCE W access point. This configuration does not require a server and the SCALANCE W access point does not have a connection to a wired Ethernet. Within its transmission range, the SCALANCE W700 forwards data from one WLAN node to another. The wireless network has a unique name. All the devices exchanging data within this network must be configured with this name.

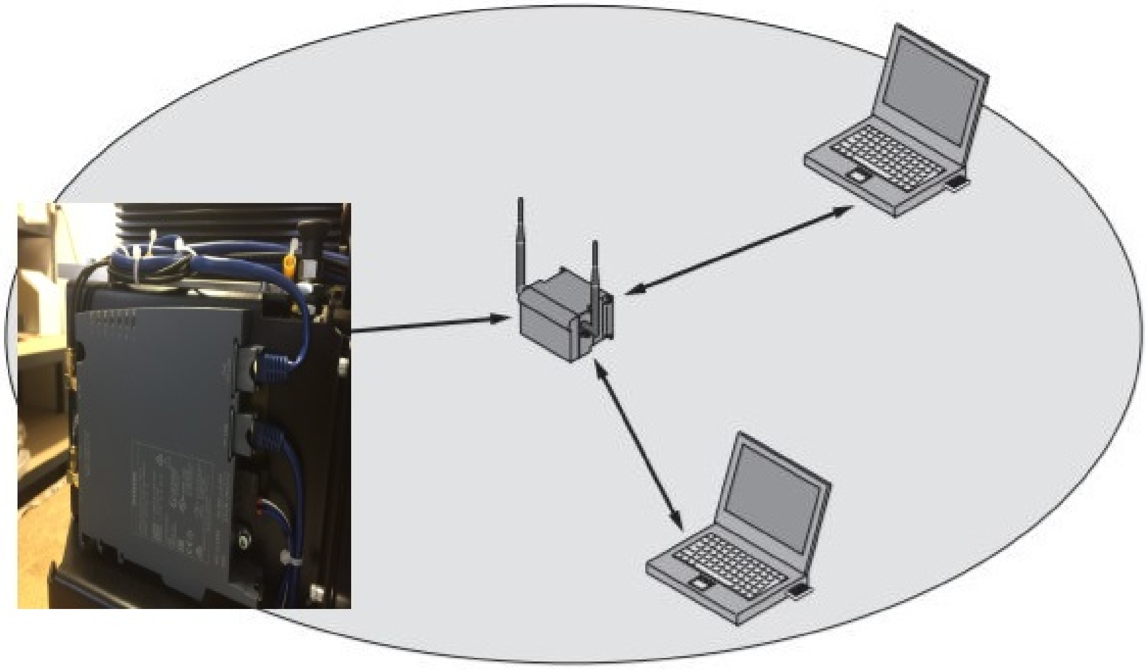


Figure 1.7: Scalance W700

1.3 State Of Art of Testing In the Market

In this section we look at to the new method in the market which is defined for Autonomous Robots. It describes the process of conducting tests while maximizing safety for operator, robot and other equipments.

Robots' tests should be repeatable and reusable and conducted in a way to maximize safety.Risks to harm human operators, or to damage the robot or other equipments should be minimal. In this test process, three test dimensions are identified:

1- The robot activity (Sensing Vs. Acting): A robot can either sense its environment, or it can sense the environment while acting. since acting relies on feedback obtained through sensors, tests involving actuators requires fully functional sensors.

2- Context Area (static Vs. dynamic): The context could be static or dynamic. A static context does not change while the robot performs its mission. A dynamic context is one that evolves over time.

3- Testers knowledge of the context: To test a robot, tester should know the environment (e.g. dimensions of the arena as well as locations and sizes of obstacles) with some levels of accuracy. Based on this knowledge, testers can predict values of sensed data that reflect a correct behavior of the robot. Still, to test a robot more thoroughly, testers need to experiment with different contexts, including ones for which they have little knowledge or imprecise one.

To help understanding the behavior of the robot, tests should be first done in an context totally controlled by developers. It's easier to detect faulty behaviors and trigger security halts. Once a robot passes those tests, testers can proceed with experiments in less controlled environments.

This is the case of tests which check sensing capabilities in an unknown context that can be either static or dynamic. Indeed, since the context is not controlled by developers, it makes it difficult for them to define meaningful tests, since they can not predict valid values are sensed data.

Figure 2.1 shows the resulting tests order organized in three levels. Upper levels are prerequisites for lower ones. This means that a test should not be performed on a robot until all tests in the levels below pass.

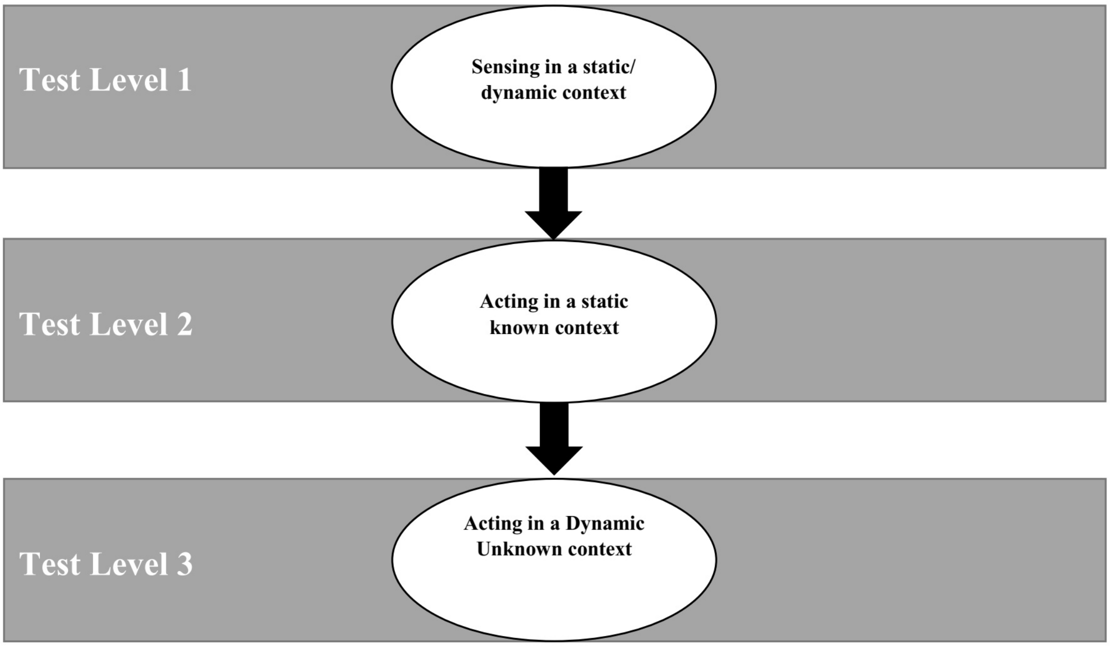


Figure 1.8: Three test levels

We now present each test level:

1- Testing robot sensing capabilities in a static known environment. This test suite ensures that data produced by sensors matches the expected accuracy level.

1.2- Testing robot sensing capabilities in a dynamic known environment. Such an environment is controlled by developers, even if it is dynamic. For example a room with some moving entities following a known path or with the light changing according to some known patterns. Compared to sensing in a static environment, a dynamic one introduces the time factor. In a dynamic environment, tests are more about the frequency of collected data than their accuracy. For example, tests in this level may evaluate if a robot can collect data fast enough to detect potential obstacles.

2- Testing robot action in a static known environment. Here the goal is to ensure that the robot actuators work properly, and that the robot performs basic actions. Still these tests can make use of sensors, since the robot is likely to behave based on perception. Besides, testers should ensure that they have some solutions to stop the robot in case of emergency (e.g. an emergency stop button).

3- This level contains two complementary kinds of tests related to robot action in both static unknown and dynamic known environments.

3.1- Testing robot action in static unknown environment. The goal of these tests is to increase confidence in the tested robot, by making it faces many different situations. Thus, testers have more chance to detect unexpected or unwanted behaviors in situations uncovered by tests in a static known environment. An example of such tests, is putting a vacuum cleaner robot in a room full of obstacles and check that it does not get stuck in between.

3.2- Testing robot action in a dynamic known environment. At this stage, testers can predict sensed data. Since the environment is supposed to be known, they can also predict when events occur and test if the robot performs the right actions. Tests do not only ensure that the robot does the right action, but they also verify that the behavior is fast enough. For example, knowing trajectories of all entities of a given environment, one can test obstacle avoidance in a dynamic environment.

3.3- Testing robot action in a dynamic unknown environment. This level represents the final tests. It aims at confronting tested robots to as many different situations as possible, especially ones unforeseen by testers. Tests are considered as passed if the robot can cope fast enough with encountered events. An example of a such test is making a mobile robot search for a person based on face recognition, in a place many people passing by (e.g. a bus station).

Chapter 2

Test Process

The whole methodology of testing at InSystems is going to be Develope as it shown by Figure 2.1 . Many steps are defined for testing an AGV which is reusable for all other kind of AGVs in the company. I go through each step individually to see the results out of this tests.
The tests are prepared in the method of testing which is defined in chapter 1.3.

2.1 Modelling Hierarchy

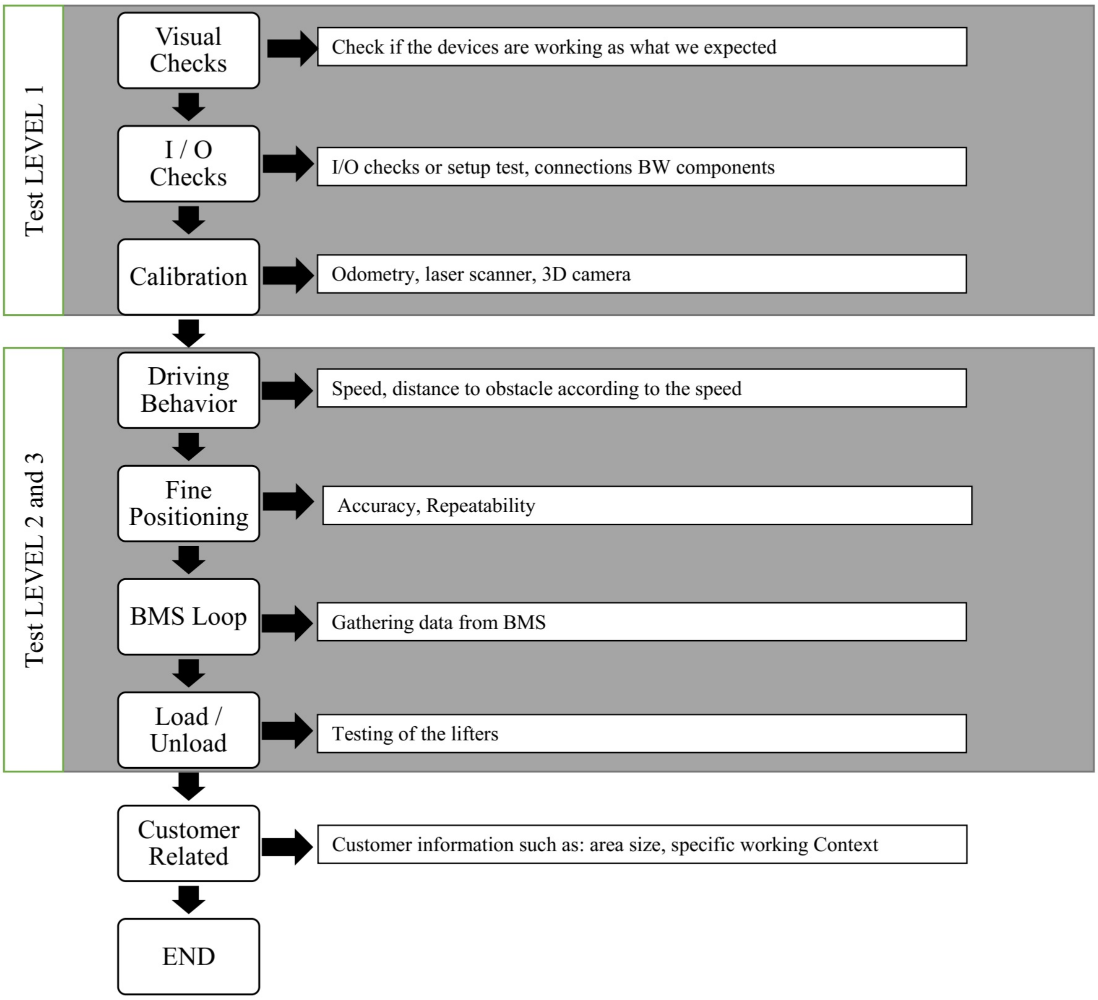


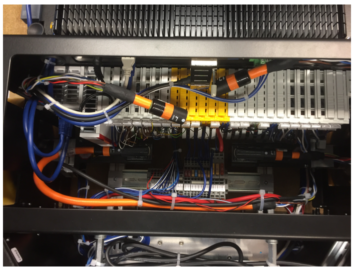
Figure 2.1: Test process

The tests are divided in two test levels. The visual, I/O checks and calibration are necessary to be test in the level that robot could sensed in static or dynamic context.

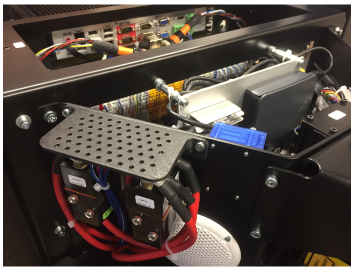
Driving Behavior, Fine Positioning and Battery loop must be tested while acting in a Static or Dynamic context to see the differential behavior of an AGV.
Driving behavior which will be discussed in Chapter 3

2.2 Visual Checks

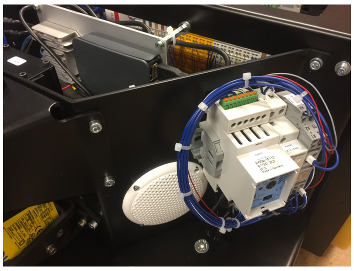
Visual checks means to check all tools and devices in the robot by testers' vision. Figures below shows different part of an AGV checked by detail.



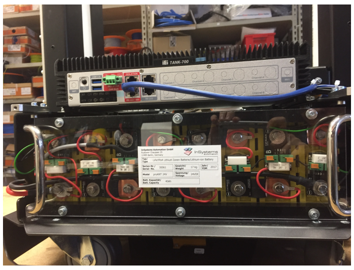
(a) PLC



(b) Right Side.



(c) Left Side.



(d) Battery.

(e) Laser Scanner.

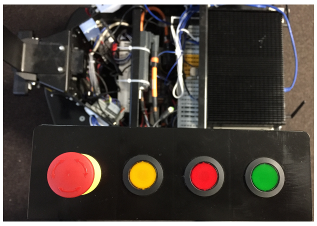
(f) Wlan Client.

(g) Wheels.

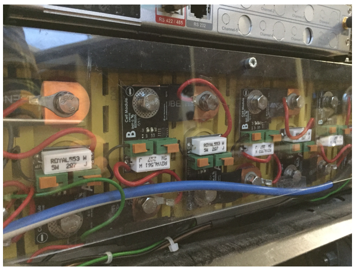
Figure 2.2: General View

2.3 Input / Output Checks

At this part each AGV have to be checked for connections and the shape of connections must be clear, if failure happened between connection it's easier to check and repair before running the whole system.
AGV system I/O such as PLC,pushbuttons, sensors, Conveyor

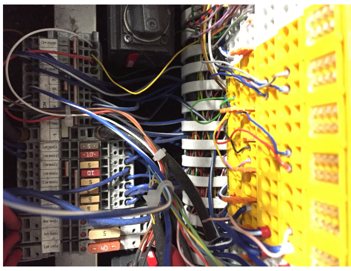


(a) Pushbottons.

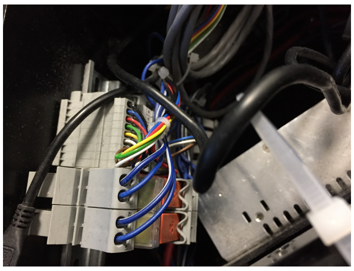


(b) Battery Connection.

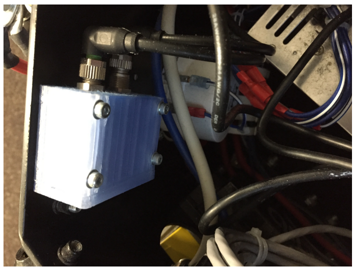
(c) Core.



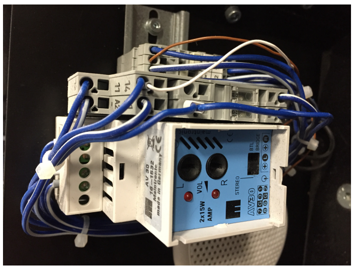
(d) PLC.



(e) PLC.



(f) Gyro connection.



(g) Amplifier for speakers.

Figure 2.3: Connections

2.4 Calibration Test

Calibration process of an AGV is needed to be fix before setting whole system.
In particular 3D camera, Laser Scanner, BMS and Odometry are the main of the system which needs to be calibrated.

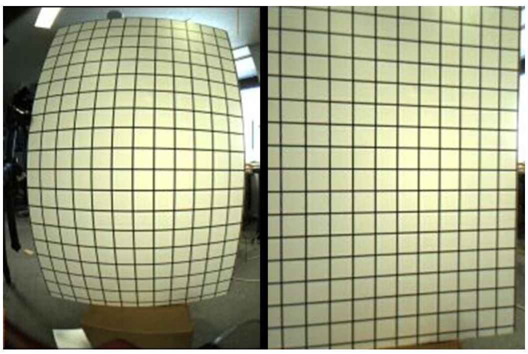
2.4.1 3D Camera

camera calibration is the process of obtaining the fundamental parameters of a camera. to estimate the camera parameters we should find the projection of 3D world points in a 2D image points. To get these correspondences we need multiple images that of a calibration pattern, such as a checkerboard pattern, from different angles and distances. The camera parameters can be divided into intrinsics and extrinsics:

- **Intrinsics:** Or camera matrix, these parameters depend on the camera and independent from the actual world frame
  - fx,fy: Focal length of the camera lens in both axes. These are normally expressed in pixels.
  - cx,cy: Optical center of the sensor(expressed in pixels).
  - k1,k2,p1,p2: distortion coefficients.

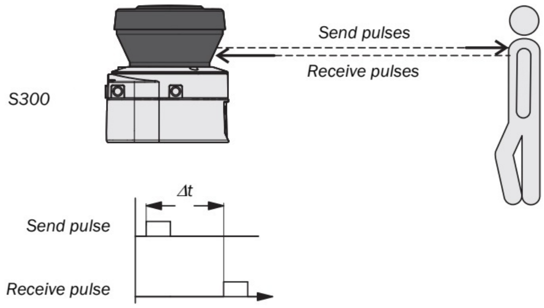
- **Extrinsics:** these parameters depend on the actual world frame, they represent the location and orientation of the camera in relation to the world frame represented by rotation and translation vectors.

Figure 2.4: Distortion removal based on the camera matrix, undistorted image on the right side

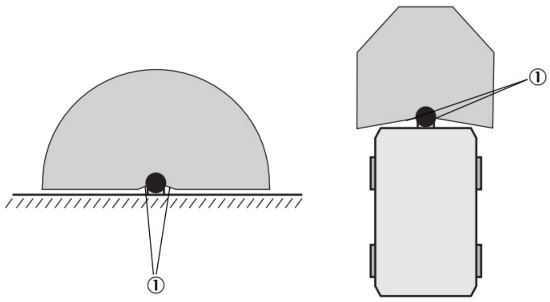


2.4.2 Laser Scanner

The calibration of a laser scanner is adjustable by connection of the laser or manually by a measurement tool which set the laser scanner. The calibration of laser scanner is needed to have straight protective field, parallel to the ground.
figure 2.5 shows laser scanner detect an obstacle Static or dynamic in the protective fields by sending pulses. The protected fields are define as 3 safety fields as shown



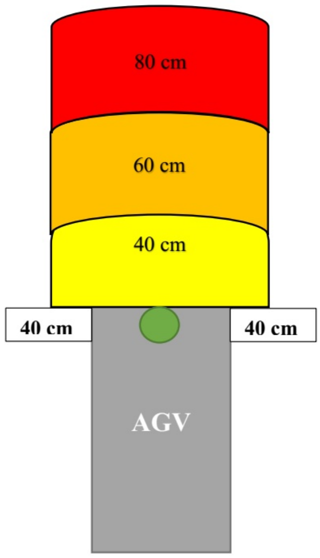
(a) Detecting Object



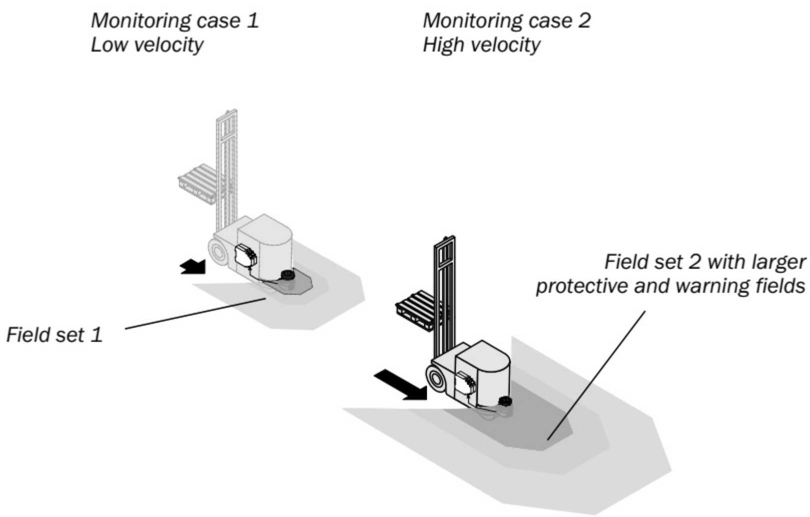
(b) Protection Area.

Figure 2.5: Laser Scanner detection

in figure 2.6, the first field is called protective field with 40cm, the second and third fields are warning fields with 60 and 80 centimeters respectively. The size of the fields are changeable due to the process-dependent hazardous area protection or velocitydependent vehicle monitoring is possible for example. The laser scanner field are changing due to the velocity as shown in figure 2.6 at the low velocity the field are more or the less the same as it defined but at the hight velocity the laser scanner field become smaller due to the high velocity. In conclusion the field are related to velocity.



(a) Safety Fields



(b) Two monitoring cases on an AGV

Figure 2.6: Laser Scanner safety Fields

2.4.3 BMS Balancing

Battery balancing and battery redistribution refer to techniques that maximize the capacity of a battery pack with multiple cells (usually in series) to make all of capacity available for use and increase each cell's longevity. A battery balancer or battery regulator is a device in a battery pack that performs battery balancing. Balancers are often found in lithium-ion battery packs for cell phones and laptop computers. They can also be found in battery electric vehicle battery packs such as AGVs that we are using.

2.4.4 Odometry

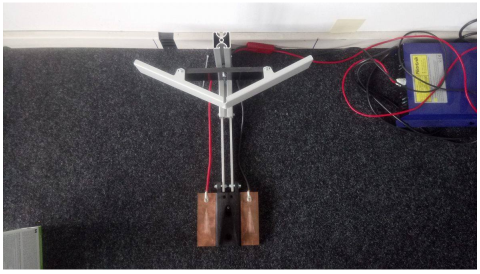
Odometry is the most widely used navigation method for mobile robot positioning; it provides good short-term accuracy, is inexpensive, and allows very high sampling rates. However, the fundamental idea of odometry is the integration of incremental motion information over time, which leads inevitably to the unbounded accumulation of errors. Specifically, orientation errors will cause large lateral position errors, which increase proportionally with the distance traveled by the robot. Despite these limitations, most researchers agree that odometry is an important part of a robot navigation system and that navigation tasks will be simplified if odometric accuracy can be improved.

2.5 Driving Behavior

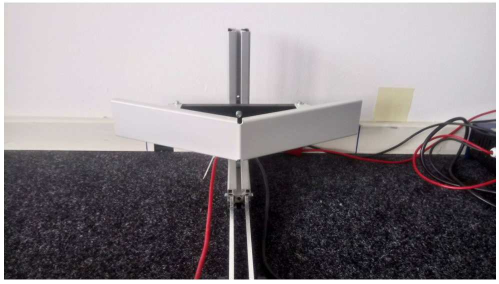
One of the most important aspects of autonomous systems is the handling of materials in industrial environments. As every industrial environments are different such as objects inside or the size of alleys, this features could effect on the driving behavior of an AGV.
What we expect from driving behavior of an AGV is to in a direct/straight line without any obstacle drive with high velocity, then in confronting with static/dynamic obstacles in specific distance the velocity must decrease to avoid hitting objects.

2.6 Fine Positioning

Fine positioning is the act of approaching a marker with minimal distance and orientation error.The actual used solution for fine positioning is scanning a triangle with the laser scanner to get its pose. In this section we will briefly describe this solution discuss the flaws and the advantages of this method.



(a) Top View.



(b) Frontal View.

Figure 2.7: Fine positioning triangle on a battery charging station

2.6.1 Description of the current solution

The current solution is relying on the laser scanner which provides a point cloud. After that triangular shapes are fetched inside the point cloud, that means the algorithm fetch consecutive points that forms a triangular shape. And then a procedure of triangle filtering starts where we keep only the shapes that meet our predefined attributes such as the length of the triangle's sides and the angle between those two sides. From there an algorithm starts calculating the position and orientation of the angle in the laser scanner coordinate system.

2.6.2 Flaws of the current solution

The main problem with fine positioning using triangles is that it is hard to differentiate between different triangles insight, that means in some cases the robot can approach the wrong conveyor since the triangles doesn't have IDs. Moreover, the triangles can't be detected properly unless the laser scanner is detecting both sides of the triangle. Also in an industrial environments the robot can be confused while performing fine positioning due to the triangular shape of the marker, considering that the environments has many shapes similar to the marker. The current solution requires also hardware deployment in the environment that can cause mechanical problems such as fixing a triangle on a conveyor.

2.6.3 Strengths of the current solution

In the current solution they are using a laser scanner which has a 260 degrees wide field of view, this could drastically improve the accuracy of the fine positioning since it is relying on actual readings from the laser scanner instead of estimating the readings. Likewise the high readings rate of the laser scanner enhances further the accuracy, and the robot can approach the triangle rapidly without risking moving faster than calculating its position.

2.7 BMS Loop

After testing all other tests which make the AGV to be safe in the field, we need to be sure that the heart of the Robot is enough safe to provide it to the customer. So, for this case a particular test is needed.

2.8 Load / Unload

Related to the customer requirements the weight of loading is changeable arises from 50 to 1000 kg. The duty of Load/Unload is carry a box from station A to station B. In this aspect we use many sensors to control the velocity.

2.9 Customer Related

Every customer have different point of view and purpose for ordering an AGV. The company has several customers with different requests due to their workplace. To be clearer here I have some examples of them, an AGV which works in a warehouse to do a processing job of movements to move objects from one station to the others.

Figure 2.8: ProAnt 390

Figure 2.9: ProAnt 490

Figure 2.10: ProAnt 485

Figure 2.8 and 2.9 are designed for a customer which asked for an AGV to carry boxes from one station to other. The ProAnt 490 is the new version that company provided to the same customer.
Figure 2.10 designed for another customer with different approach.

Chapter 3

Developed test process

Each step in chapter 2 needs a test procedure to be tested. In test procedure the expectation of test is different from reality. Despite passing, there would be many fails before achieving the expected result . All the test in the test process have specific scenarios but the same structure:

Test Scenario: What is this test about?The purpose of the test.

Minimal Requirements: The minimal requirements that is needed

Pretest: What pretest has to be performed.

Test scenario description: Describe the planning, design, execution, and documentation associated with the acceptance test effort. List the major acceptance test activities.

Data Collection and preparation: What data has to be collected? Analyzing data gathered from the test

Pass/ Fail Criteria:At the end, what is the final behavior/conclusion out of tests to analyze and resolve any fail.

3.1 Visual Checks

3.1.1. Test Scenario

The aim out of this test is to have an assembled robot with testers' vision which checked all the devices and appearance of the AGV.

3.1.2. Minimal Requirements

- Robot Platform

3.1.3. Pretest

- None

3.1.4. Test Scenario Description

The complete assembled AGV must be in an empty and secure context which tester could have good view on the AGV.

3.1.5. Data Collection and Preparation

Checklist of appearance of the AGV As the tester must be informed of each devices/tools inside and outside of a robot, then it must be clear for him if fault in connections or in the body appearance to consider it as fault in the checklist.

3.1.6. Pass/Fail Criteria

In pass/fail criteria of visual checks we have a checklist to consider, below you can see the visual checklist:

3.2 Input / Output Checks

3.2.1. Test Scenario

The aim is to find and resolve the error at the in entire robot. The tester must check all devices used in the robot platform.I/O checks requires to check all devices connections to other devices which generate a specific behavior such as

3.2.2. Minimal Requirements

-Robot platform: A complete assembled robot which is ready to work.

-PLC softwares such as: TwinCat for Beckhoff devices and CodSys.

3.2.3. Pretest

- Visual checks

3.2.4. Test Scenario Description

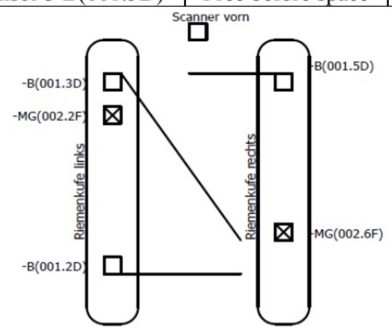
The robot must be tested in an empty space of at least 3 meter width and 5 meter length. Testing requires specific softwares to check and control the devices in use. The most significant part is working with PLC that controls

3.2.5. Data Collection and Preparation

The I/O checklist provides the pass/fail criteria. Blow you can see the I/O checklist:

Button	Function	I/O
One and Quit	Green lamp	
	Turn on FTF	
	Acknowledge error	
Function 1	Stroke in min/max	
	Drive position	
	> Press 5s	
Function 2	Synchronize hub	
	> Press for 3s	
	Error No .: 110	
System OFF	red lamp	
	Shut down the FTF	
	Turn off FTF	
	without core-off	
	> Press 5s	
Release brake	blue lamp	
	Release brakes	
load	yellow lamp	
	Charging relay on / off	
Emergency stop on the left	Brake left / right active	
	System OFF lamp	
Emergency stop on the right	Brake left / right active	
	System OFF lamp	
Motor		
Press "Forward"	Rotates	
Press "Backward"	Rotates	
Actual Position		
Actual Velocity		
Scanner	Align Sick Scanner	

		I/O
Service ports		
USB		
LAN		
SENSORS		
Sensor 1 B(001.2D)	Free after space	
Sensor 2 B(001.3D)	Free space	
Sensor 3 B(001.5D)	Free before space	



		I/O
Bands		
Activate the band on the left	Press "Forward"	
	Press "Backward"	
Activate the band on the right	Press "Forward"	
	Press "Backward"	
Charger	Single Cell spg	
	Single cell temp	
	Set SOC	
	Set Temp.	

Figure 3.1: I/O Checklist

3.2.6. Pass/Fail Criteria

1- The emergency stop failed to respond, so the conductor of connection between PLC and emergency stop has changed.

2- The sound bip, when the AGV is moving backward was not activated

3- The softwares check the connection between tools such as lights. The two sided lights of the robot were blinking 50 ms but the standard is 500 ms. The programmer/tester changed the blinking time to 500 ms by developing the code.

3.3 Calibration

3.3.1 3D Camera

3.3.1.1 Test Scenario

To distinguish between the floor and flat obstacles it is important to have precise values for the cameras height and orientation towards the floor. The node features an iteration that can calculate those values if the camera looks on an empty part of the floor.

3.3.1.2 Minimal Requirements

- Robot Platform

- 3D camera

3.3.1.3 Pretest

None

3.3.1.4 Test Scenario Description

The calibration used to precisely determine the mounting parameters (height of the camera, inclination in the direction of travel and inclination perpendicular to the direction of travel). The following steps must be followed:

1 -Connect to the 3dD Camera via Ros

2-Make sure the vehicle is looking at a straight, open area.

3-Start the console with Ctrl + Alt + T.

4-Enter the command "cd / home / pi / Desktop / CollisionProtection /" and confirm with Enter.

5-Enter the command "sudo ./a.out" and confirm with Enter.

6-Press "Enter", "c", "Enter" to start the configuration.

7-Wait a few seconds until the parameters below do not change any more.

8-Press "Enter", "s", "Enter" to save the parameters.

9-Press "Enter", "g", "Enter" to finish the graphical interface.

3.3.1.5 Data Collection and Preparation

In the upper part of the graphic surface are all points whose height ¿= 0 mm is represented by "0" and all whose height ¡0 mm is marked with _. The differences are due to the measurement inaccuracies. The characters "0" and _ should be distributed chaotically over the output after successful configuration, as shown in Figure 3.2.

Figure 3.2: 3D Camera Calibration

3.3.1.6 Pass/Fail Criteria

1-Make sure that the Camera is straight forward.

2-If the procedure failed for calibration redo the steps from beginning.

3.3.2 Laser Scanner

3.3.2.1 Test Scenario

This test is aimed to adjust the laser scanner by measuring the width from different distances. The expected result out of this test is to have corrected mounted which makes the laser scanner to be straight forward and parallel to the ground.

3.3.2.2 Minimal Requirements

- Robot Platform

- Laser Receiver: The laser receiver have two modes of precision:
Coarse precision with ± 2mm and Fine precision with ± 1mm. The unit can be adjusted for either a fine or course band, which will help improve the accuracy.

3.3.2.3 Pretest

None

3.3.2.4 Test Scenario Description

The following setup should be executed:

- Having a space in front of the robot with at least 3 meters length and approximate 3 meters width. 3.3

- Setting the laser receiver with distances of 1 and 3 meters in front of laser scanner and the same distance for the corners with 45 The Measuring of laser scanner is done by a laser receiver Topcon LS-80L.

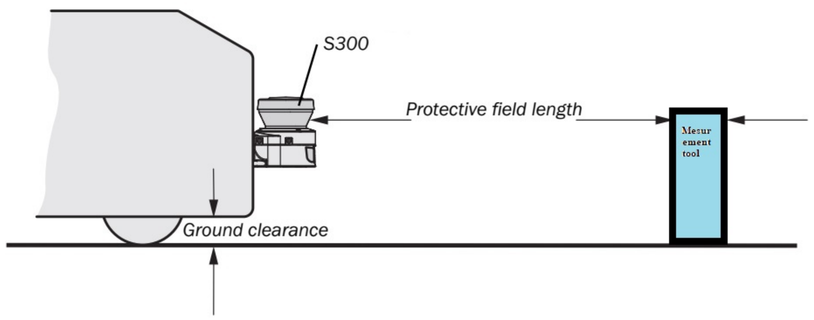


Figure 3.3: Laser Scanner Calibration

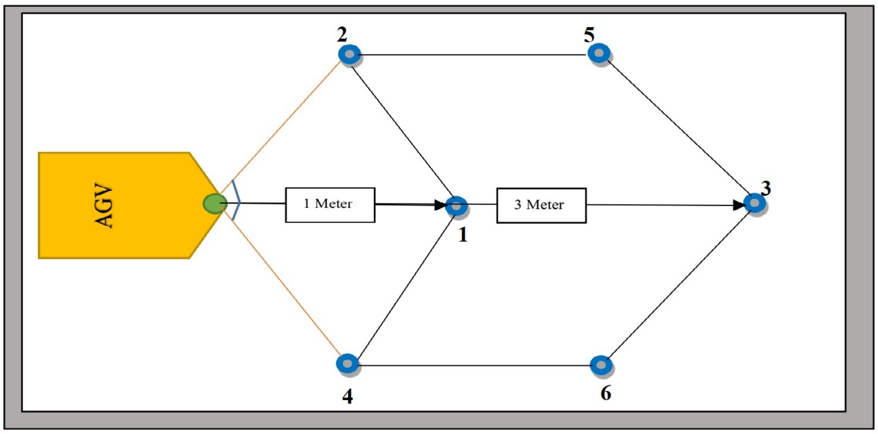


Figure 3.4: Laser Scanner adjustment

In Figure 3.4 There are 6 points of Adjustments:

Point 1: The laser receiver is in front of laser scanner with minimum distance of 1 meter.

Point 2:The laser receiver is in the corner with approximate of 45 degree to laser scanner with minimum distance of 1 meter

Point 3:The laser receiver is in front of laser scanner with minimum distance of 3 meter.

Point 4:The laser receiver is in the corner with approximate of 45 degree to laser scanner with minimum distance of 1 meter

Point 5:The laser receiver is in the corner with approximate of 45 degree to laser scanner with minimum distance of 3 meter

Point 6:The laser receiver is in the corner with approximate of 45 degree to laser scanner with minimum distance of 3 meter

3.3.2.5 Data Collection and Preparation

- Measuring upward and downward of the laser fields.

The table 3.1 shows a measuring field of alignment for a laser scanner which gathered from laser receiver in different points and distances.

Points	Downward	Upward
1	7.8	11.8
2	8	11.7
3	8	10.8
4	8	11.7
5	8.3	10.6
6	8.3	10.6

Table 3.1: Measuring Fields Of A Laser Scanner

Note that the standard range of upward and downward is approximate 11.5 cm.

3.3.2.6 Pass/Fail Criteria

The field of laser scanner was straight forward and parallel to the ground.
The test that we did was not correct,it was parallel to the ground but the Standard range dor upward and downward is 11.5 cm. which for our test is 10 cm.
If the test failed to be alighned/parallel to the ground with standard range, there are two screws in both side of laser scanner that can be loose to modify it by moving up or down.

3.3.3 Odometry

3.3.4.1 Test Scenario

This test is aimed to test the odometry.Quantitative measurement of odometry in Autonomous Robots is difficult. Lack of well-defined measuring procedure due to poor calibration of Autonomous Robots, to solve this problem is to develop a method for quantitative measurement. The purpose of this test is the merching of odometry.

3.3.4.2 Minimal Requirements

- Robot platform

- Gyroscope

- TwinCat: Software System

- Wheel Encoder

3.3.4.3 Pretest

-None

3.3.4.4 Test Scenario Description

The robot must drive between 2 goals , certain velocity and certain distance. The following setup should be executed:

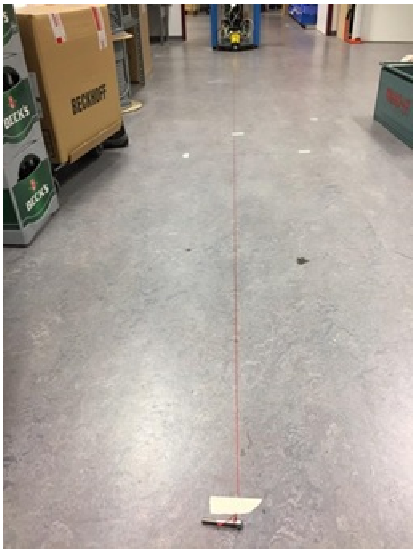
1-Having space in an area without obstacles.

2-Creating a map of the contex with minimum of 5 meters length and approximate 2 meters width.

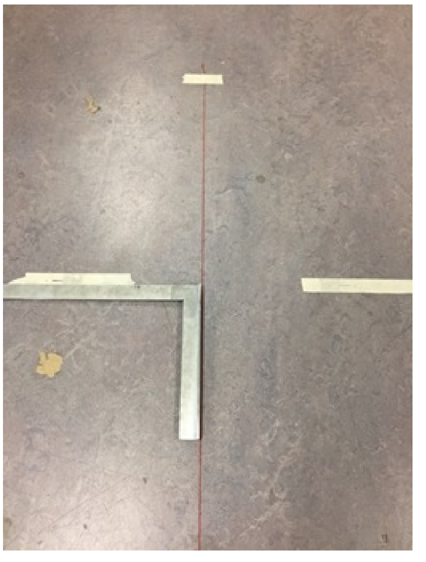
3-Record starting point for AGV by means of stop angle across the direction of driving.

4-set Velocity = 250mm/s

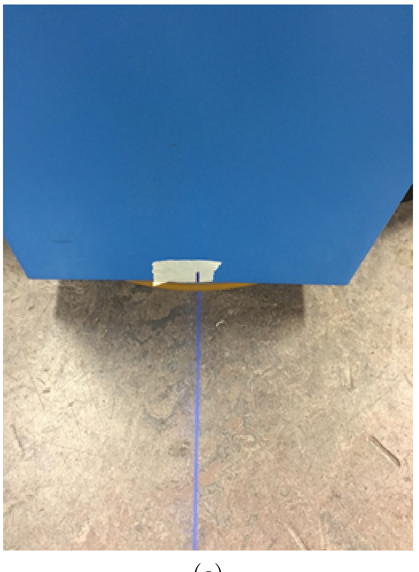
5-Distance = 5000mm



(a)



(b)



(c)

(d)

Figure 3.5: Odometry

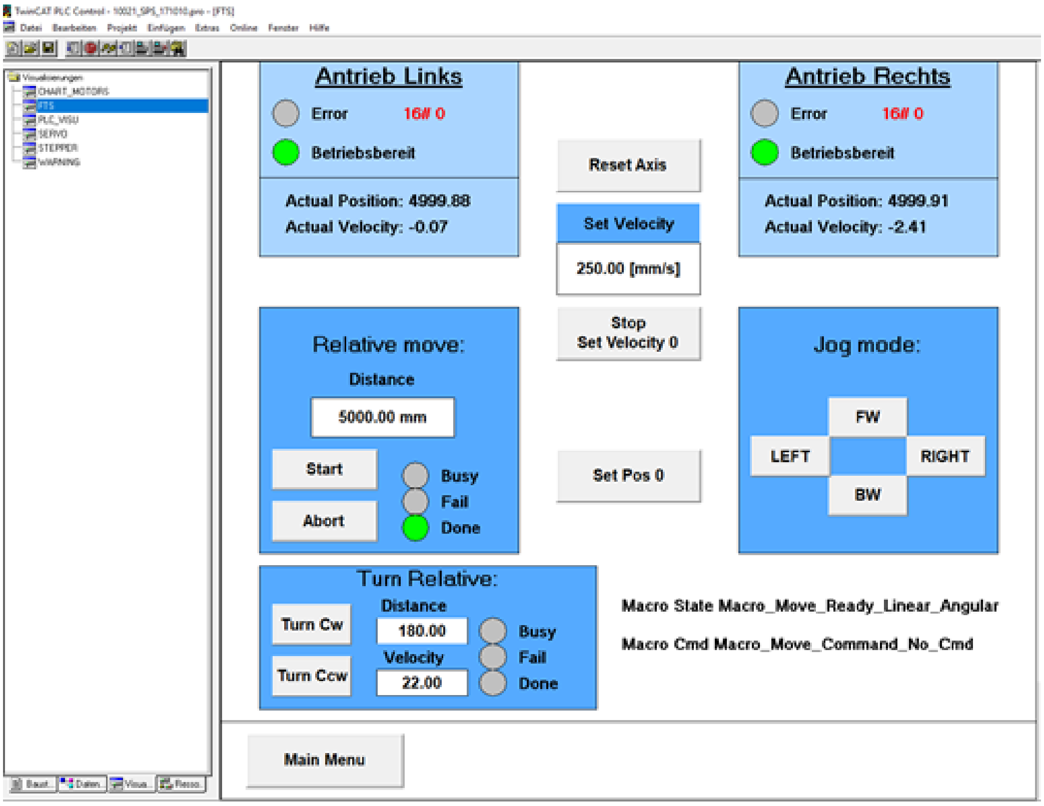


Figure 3.6: Moving Via Sps Program

3.3.4.5 Data Collection and Preparation

- Measuring the real distance between start and stop points

- Measuring the distance to the robot center to the marked line (drift)

- Scale Velocity := 0.2339,

- Scale Position := 0.00001685,

- Position Tolerance := 6.0,

Move Param: 1-Acceleration = 700; 2-Deceleration = 700

3.3.4.6 Pass/Fail Criteria

If Fail occurred for distance or turning is to change the parameters to get the right measurement as we expected.

3.4 Fine Positioning

3.5.1. Test Scenario

The aim of this test is to find the certain triangle with certain distance.The expected result is to have a reasonable accuracy of fine pos, by testing in the specific context.

3.5.2. Minimal Requirements

- Triangle with specific Degree

- Robot Platform

- Presence-plus Camera

- Presence-Plus Vision Software

3.5.3. Pretest

- laser Scanner calibration

- odometry

3.5.4. Test Scenario Description

The Robot by fine positioning try to recognize a certain triangle (e.g. charge station) in an area. The following setup should be executed:

- Creating a map of the environment with minimum of 5 meters length and approximate 2 meters width.

- Have space in an area without obstacles.

- Setting a fixed triangle (e.g. charge station)

- setting the camera on top of the Triangle to take photos of each fine pose as shown in figure

Figure 3.7: Context area

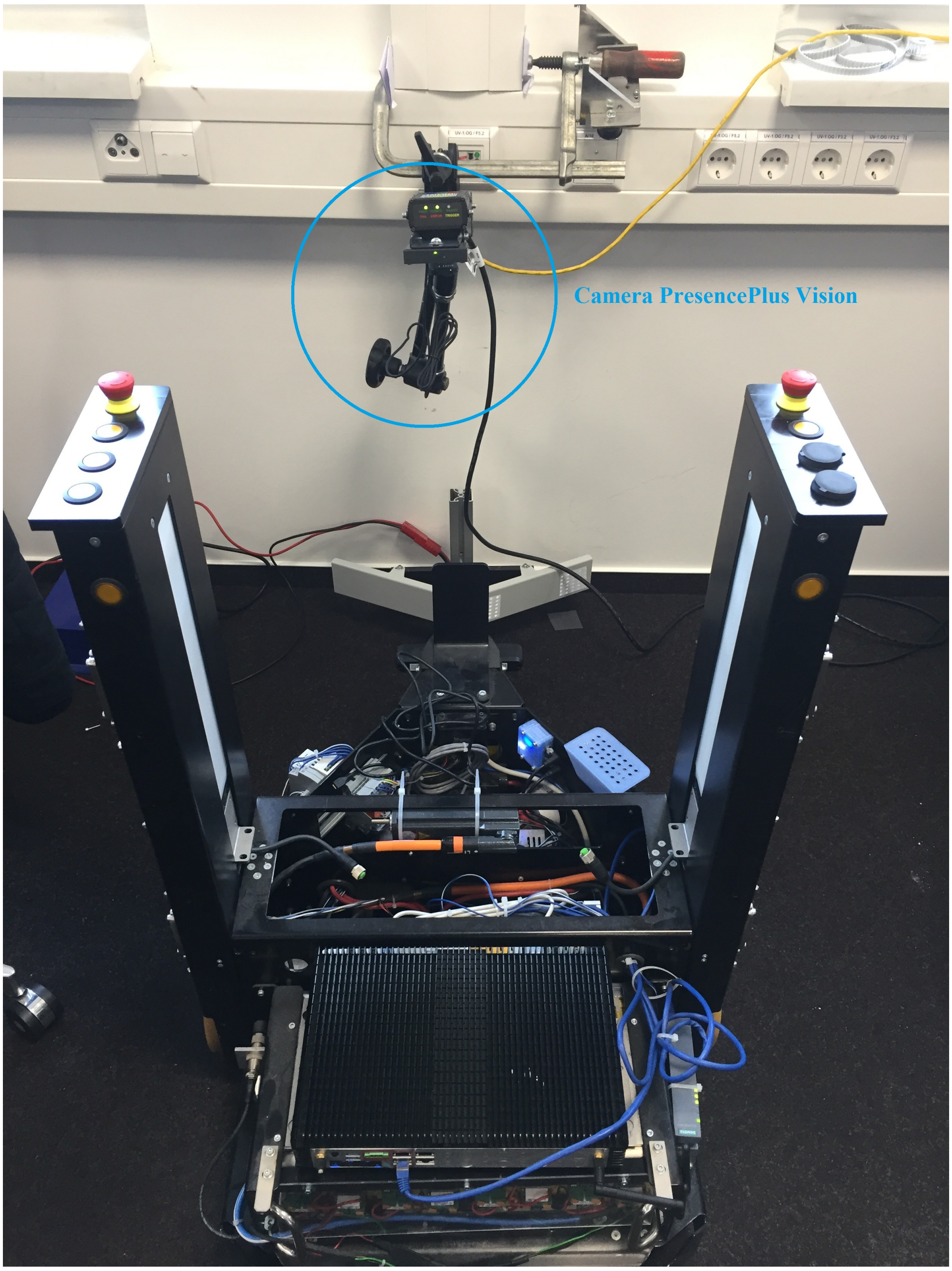


Figure 3.8: Presence-Plus Vision

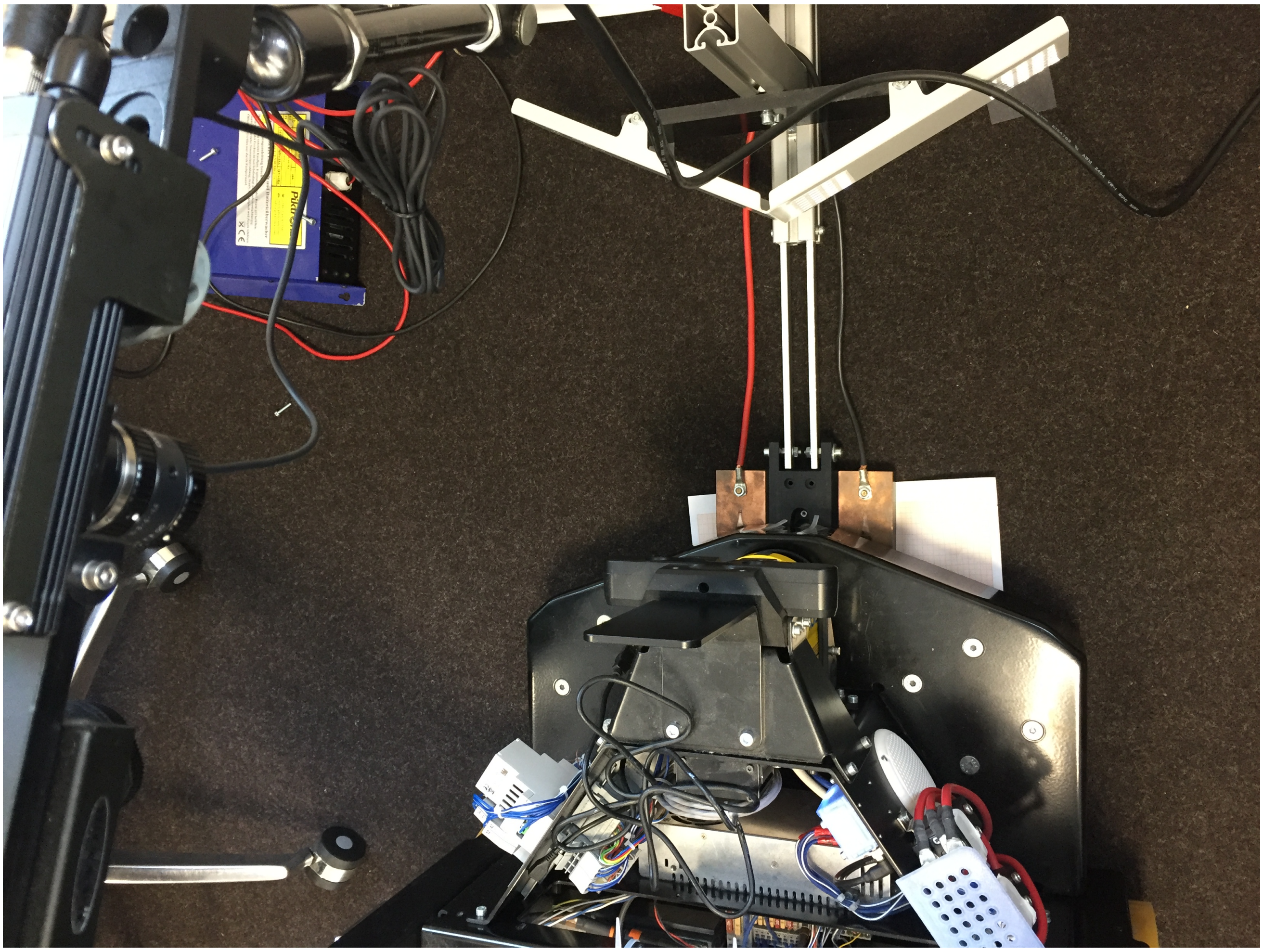
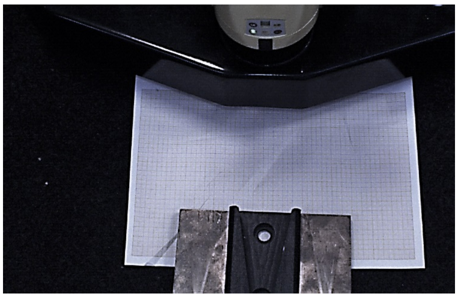
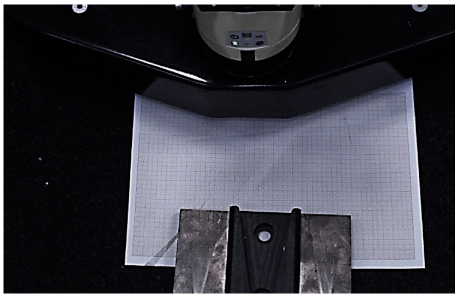


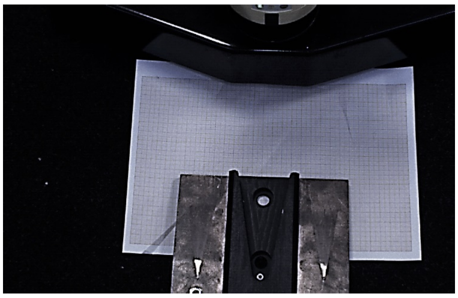
Figure 3.9: Top View

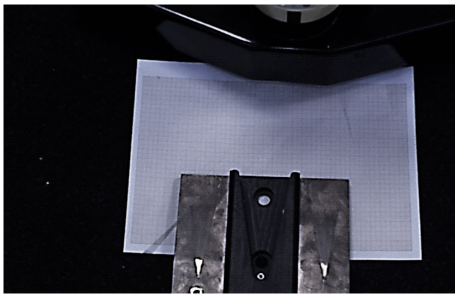
3.5.5. Data Collection and Preparation

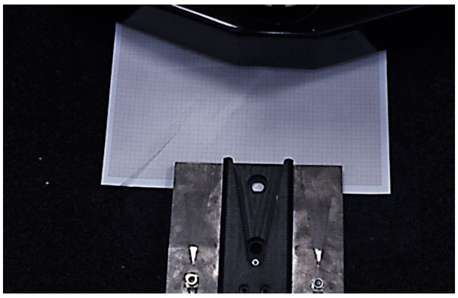
The photos below shows that the accuracy of fine pose in not 100% but good enough to finishing the fine pose. The test has done many times, to check the accuracy, as we can see all 8 photos are different the reason is laser scanner merging error.

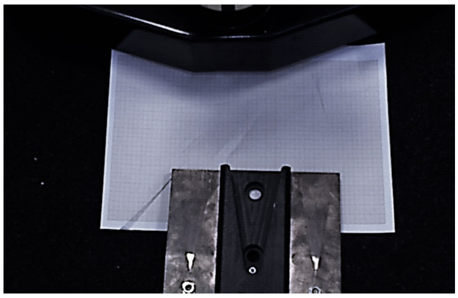

(a)

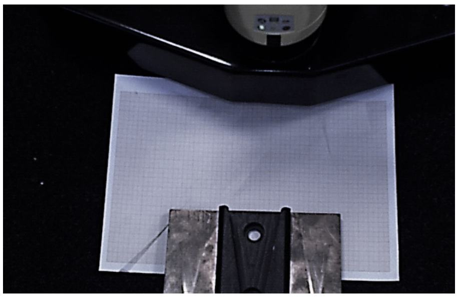

(b)


(c)


(d)


(e)


(f)


(g)

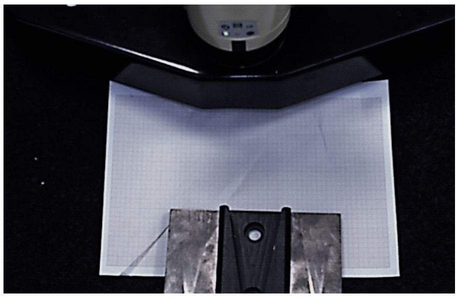

(h)

Figure 3.10: Fine pose

3.5.6. Pass/Fail Criteria

When failed occurred there many reasons to be checked:

1-laser scanner noise

2-Marching of Odometry

3.5 BMS Loop

3.6.1. Test Scenario

This test is aimed to test the BMS behavior over time. The battery status relative to predefined environment conditions will be tested and the data will be analyzed with different software version and ProANT types.
The expected results out of this test is to have graph analysis of different varibales such as SOC variation, voltage variation, and time stamps relative to different sub-scenario conditions and changes.

3.6.2. Minimal Requirements

- Battery

- Charge Station

- AIC: It's an interface between robot and user.By AIC The user can set goals, when to do fine pos and making a loop between goals to drive.

3.6.3. Pretest

- 3D camera

- Laser Scanner

- Odometry

- Fine Positioning

3.6.4. Test Scenario Description

The robot must drive between two goals repeatedly. While the Soc or Voltage for instant reaches a specific level must goes to charge station for recharging. The following setup should be executed:

- Having space in an area without obstacles

- Creating a map of the context with minimum of 5 meters length and approximate 2 meters width.

- Setting two fixed goals in the map with a distance of approximate 4 meters between the goals

- Setting a charging station in the area of 2 goals.

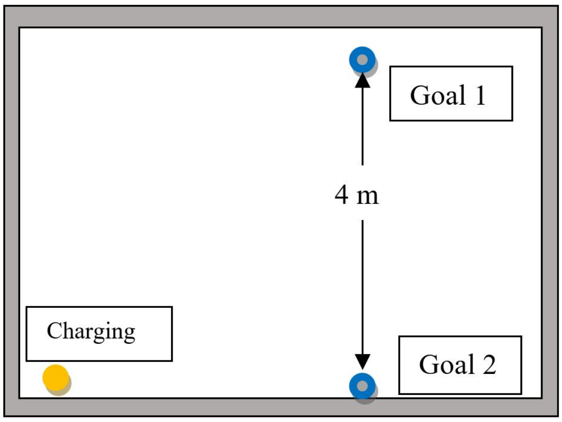


Figure 3.11: BMS Test Context

3.6.5. Data Collection and Preparation

1- Min/Max Voltage with reference to time in one cycle of charging. 2- Soc and Voltage wrt. time in one cycle of charging. 3-Soc and Voltages Wrt. time in a loop. The data are gathered from log file of the AIC.

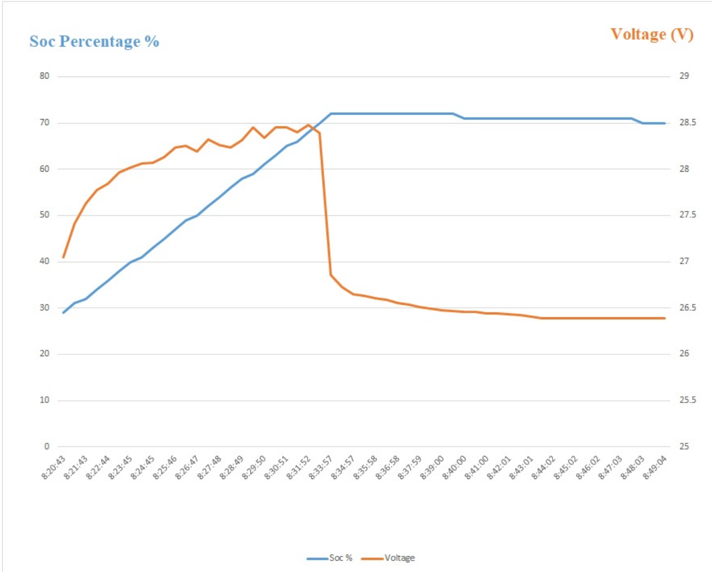


Figure 3.12: Soc and Voltage Wrt. Time

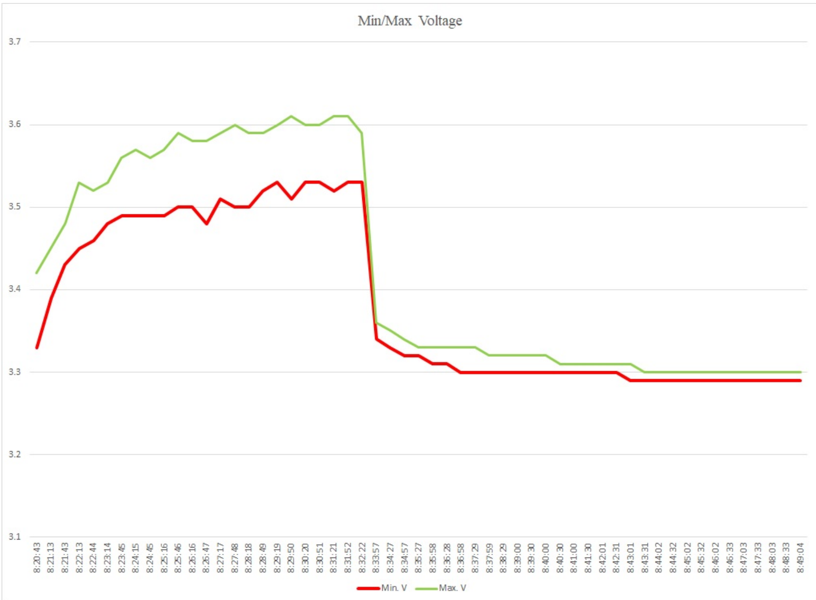


Figure 3.13: Min/Max Voltage Wrt. Time

Figure 3.12 When the battery reaches its limits, it needs to be charged when the Soc reach to 20%, so, the voltage increasing till the Soc% reach 80% ,then the voltage immediately decrease. As we can see on figure 3.13 the green line is maximum voltage and the red line is minimum voltage of one cell in a cycle of charging. The aim is to have minimum threshold between maximum and minimum.

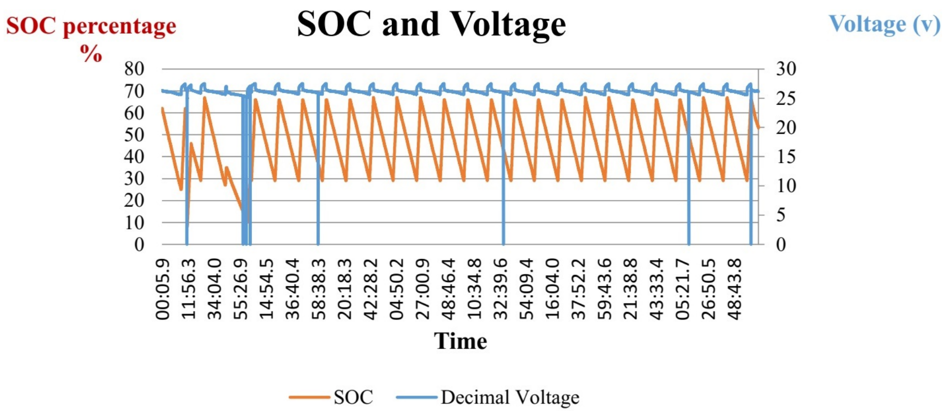


Figure 3.14: Soc and Voltage wrt time

Figure 3.14 shows the BMS behavior of a loop cycle in 2 days of running. The first 5 cycles of charge and discharging are not the same, which shows that the BMS trying to be in a normal cycle, so as we can see the soc% reaches under 20% the voltage increases. Afterward the BMS is in normal cycle and behave as expected.

3.6.6. Pass/Fail Criteria

In the fail criteria the BMS cycles will not be in the same period of cycles due to some reasons:

1-The battery is old.

2-Check which cells are defected.

3.6 Load / Unload

3.7.1. Test Scenario

The aim is to test the lifter by loading and unloading. The very last test is to test the lifter because the robot is ready after many test procedures.

3.7.2. Minimal Requirements

- Robot Platform

- Roller station with the Triangle

- AIC: for ordering to load/unload

3.7.3. Pretest

- 3D camera

- Laser Scanner

- Odometry

- Fine Positioning

- BMS Loop

3.7.4. Test Scenario Description

The following setup should be executed:

- Having space in an area without obstacles

- Creating a map of the context with minimum of 6 meters length and approximate 3 meters width.

-

Figure 3.15: Roller station

3.7.6. Pass/Fail Criteria

In the fail criteria the follow the steps:

1-check the sensors.

2-check the PLC.

Chapter 4

Feedback Process

The feedback process combined with many other tests that in following the other tests. such as:

1-Battery Balancing.

2-Driving Behavior.

4.1 Battery Balancing

4.1.1 Test Scenario

This test is aimed to balance the BMS by testing in two methods; one when the BMS is already connected to the robot and the second method to calibrate it individually.

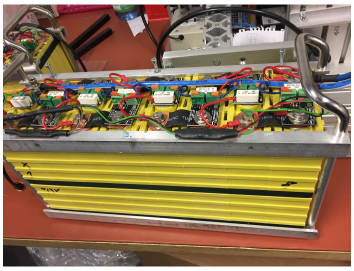
4.1.2 Minimal Requirements

-Charger

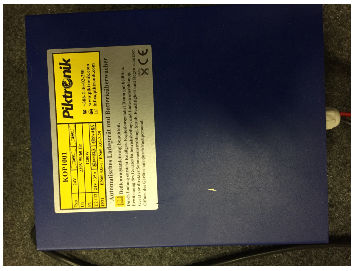
-BMS tester

-Ammeter

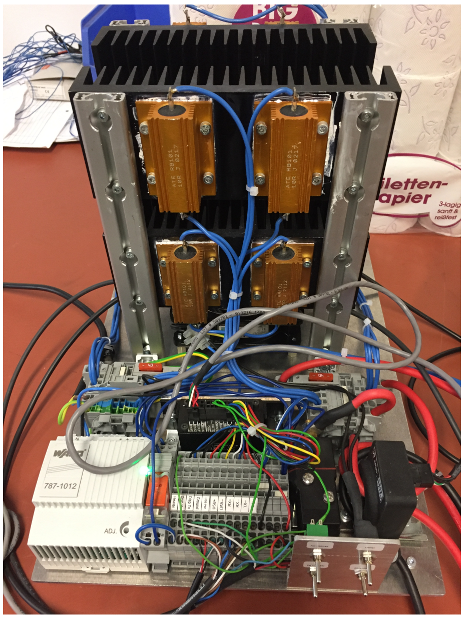
-Emus Software.



(a) BMS Cells.



(b) Charger.



(c) BMS tester station.

Figure 4.1: Requirement tools

4.1.3 Pretest

None

4.1.4 Test Scenario Description

1.First Method: When the BMS needs to be charge we have to connect the BMS to the charger directly.

2.Second Method: We can either balance the BMS individually by connecting to the BMS tester, Charger and Emus Software. Measuring whole cells to check the voltage between 24-30 v othrwise Check each cell individually in case of failure in difference voltages to find the cells' failures.

4.1.5 Data Collection and Preparation

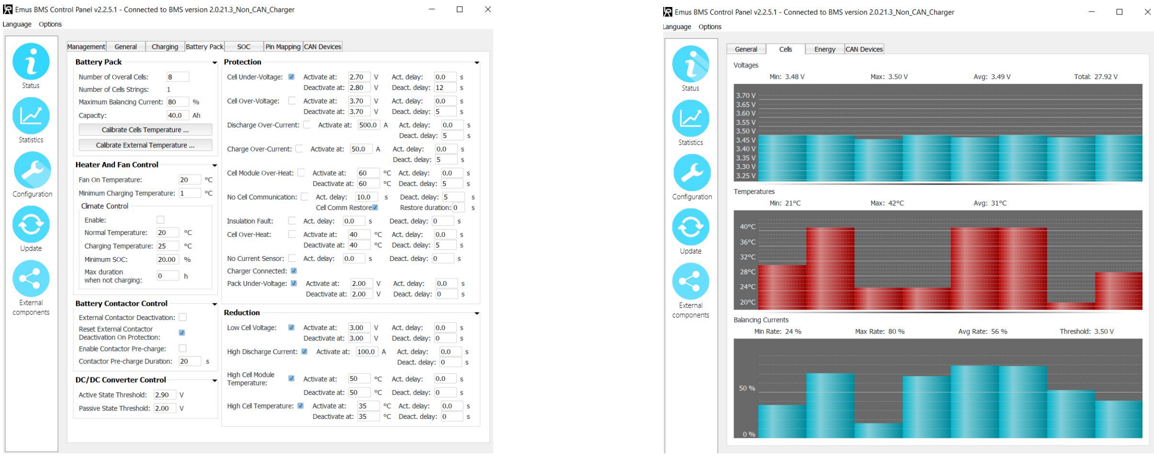
- Data Gathered from Emus Software - Measuring Voltages by Ammeter

First Method: After full charge we can measure each cell by an ammeter. As the circuit is a series circuit all voltages are divided to each cell equally, so as shown in the figure you can see measured voltages from a Balanced BMS.

BMS	Voltage (V)
BMS Cells	26.4
Cell 1	-3.30
Cell 2	3.31
Cell 3	-3.31
Cell 4	3.32
Cell 5	-3.31
Cell 6	3.31
Cell 7	-3.31
Cell 8	3.37

Table 4.1: Balance Voltages of BMS

Second Method: The BMS at the same time connected to the BMS tester(is Connected to the Emus Software by bluetooth) and charger. As shown in Figures **??** . At first The BMS tester will empty the BMS voltages then by connecting to the charger we can collect data from Emus software.



(a) Emus Software. (b) Voltages and Temperatures.

Figure 4.2: Emus Software

4.2 Driving Behavior

4.2.1 Test Scenario

This test is aimed to test the driving behavior of the robot. The expected result out of this test is that, Robot should drive with certain speed between 2 goals and in confronting with dynamic obstacles (ex. Human body) the robot should pass smoothly with certain speed.

4.2.2 Minimal Requirements

- Robot Platform

- Dynamic Obstacles

4.2.3. Pretest

-None

4.2.4. Test Scenario Description

The robot must drive between two static goals repeatedly. While dynamic obstacle (ex. Human body) suddenly goes between 2 goals. The following setup should be executed:

- Having space in an area without obstacles.

- Creating a map of the environment with minimum of 6 meters length and approximate 2 meters width

- Setting two static goals in the map with a distance of approximate 5 meters between the goals

- Dynamic obstacles could be anywhere at any time between two goals.

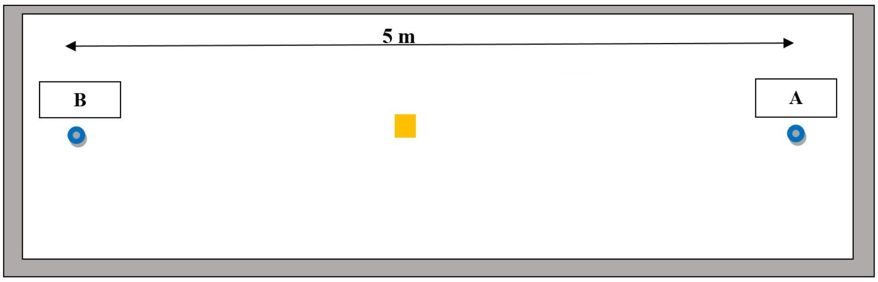


Figure 4.3: Context area

4.2.5. Data Collection and Preparation

- Velocity wrt. dynamic obstacle

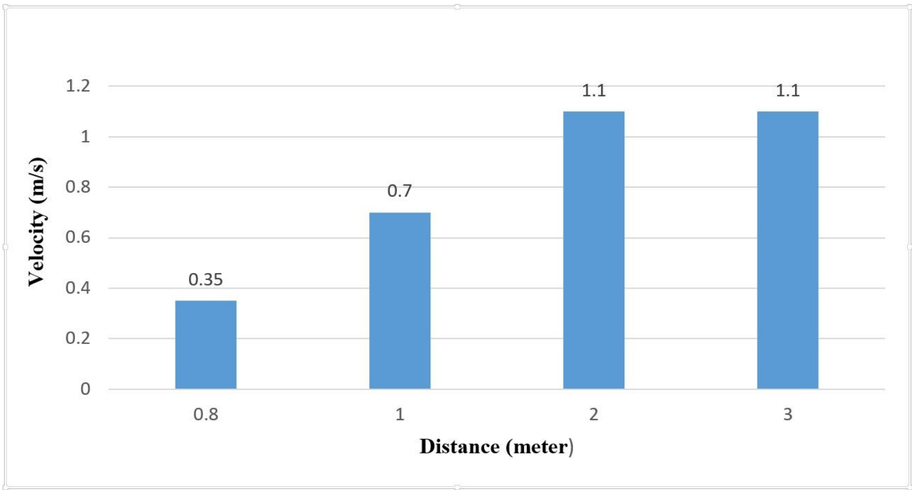


Figure 4.4: Dynamic/Static Obstacles

4.2.1 Second Driving behavior Scenario

4.3.1. Test Scenario

The robot should drive between 2 static goals repeatedly with certain speed and certain footprints, while go through 2 static obstacles.

4.3.2. Minimal Requirements

- Robot Platform

-Static Obstacles

-Laser Scanner

4.3.3. Pretest

- None

4.4.4. Test Scenario Description

The following setup should be executed:

- Having space in an area with fixed obstacles.

- Creating a map of the environment with minimum of 6 meters length and approximate 2 meters width.

- Setting 2 fixed goals in the map with a distance of approximate 5 meters between the goals.

- Setting 2 fixed obstacles in the map with a distance of approximate 1 meter between the obstacles.

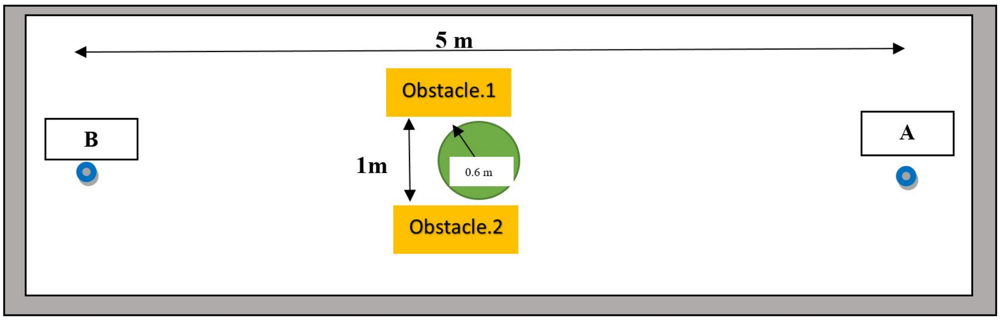


Figure 4.5: Context area

4.4.5. Data Collection and Preparation

- Dynamic footprints and velocities

-Footprints shape: Circular

-Minimum radius: 0.5m

-Maximum radius: 1m

-Minimum translational velocity: 0.3m/s

-Maximum translational velocity: 0.95m/s

-Minimum rotational velocity: 0.3m/s

-Maximum rotational velocity: 0.95m/s

Chapter 5

Future Prospect

In the future the company is planning to make sufficient plans for testing.
The plan is:

1-Working with tests: The test must get update with the technology

2-Working with test executions: The data must be comparable to other tests or with the new version of tests.

3-Managing requirements and defects: The requirements could change to new make new methods for testing.

4-Automated and efficient: It's been a long time that companies test their products manually which takes too much time for each system. The testing we did here is also take manually, integrating with the vehicle for many days due to malfunctions, defects and many more reasons. In the market theres a new way of testing which engineers trying to make it sufficient which is software test.
Test engineers are getting involved with the technology so there is not just manually testing, also it is coding. As every computer engineer is familiar to software testing it is a must for test engineers. For automated testing there is prospect if you are skilled in automating all types of testing like unit testing, integration testing, performance testing, load testing. If you can do this, you are no less than a developer. For manual testing there is none because manual testers are easily replaceable.

1-Unit Testing

2-Integration Testing

3-Performance Testing

4-Load testing

Bibliography

[1] Roger Bostelman, Will Shackleford, Geraldine Cheok, Richard Norcross. *National Institute of Standards and Technology.* [*Standard Test Procedures and Metrics Development for Automated Guided Vehicle Safety Standards*].

[2] Jannik Laval, Rob Alexander. *Mines-Telecom Institute, Mines Douai, France.* [*A Methodology for Testing Mobile Autonomous Robots*].

[3] Jae-Hee Lim, Suk-Hoon Song, Jung-Rye Son, Tae-Yong Kuc , Hong-Seong Park, Hong-Seok Kim. *Sung Kyun Kwan University.* [*An Automated Test Method for Robot Platform and Its Components*].

[4] James Arnold, Luc Fabresse and Noury Bouraqadi. *University of York, UK.* [*Testing Autonomous Robot Control Software Using Procedural Content Generation*].

[5] Andreas Nuchter, Oliver Wulf, and Kai Lingemann. *University of Hannover.* [*Robotic 3D environment Cognition*]. September,2006.

[6] F. Ducatelle, A. Forster, G.A. Di Caro and L.M. Gambardella. [*New Task allocation Methods for Robotic swarms*].

[7] R.M.B. Smit. *University of Eindhoven.* [*Measurement system and control design for AGV localization and motion control*]. November,2016.

[8] Christof Röhrig, University of Applied Sciences and Arts in Dortmund,christof.roehrig@fh-dortmund.de, Germany André Heller, University of Applied Sciences and Arts in Dortmund, andre.heller@fh-dortmund.de, Germany Daniel Heß, University of Applied Sciences and Arts in Dortmund, daniel.hess@fh-dortmund.de, Germany Frank Künemund, University of Applied Sciences and Arts in Dortmund, frank.kuenemund@fh-dortmund.de, Germany [*Global Localization and Position Tracking of Automatic Guided Vehicles using passive RFID Technology*]. June,2014.

[9] Elena Messina, Adam Jacoff, Jean Scholtz, Craig Schlenoff, Hui-Min Huang, Alan Lytle,John Blitch. [*STATEMENT OF REQUIREMENTS FOR URBAN SEARCH AND RESCUE ROBOT PERFORMANCE STANDARDS*]. May,2005.

[10] Joe Jarzombek. *US Department of Homeland Security*. [*Software Assurance: Enabling Security and Resilience throughout the Software Lifecycle*].

[11] Kaiyu Zheng. [*ROS Navigation Tuning Guide*]. September,2016.

[12] R. Bostelman, P. Russo, J. Albus, T. Hong, and R. Madhavan. [*Applications of a 3D Range Camera Towards Healthcare Mobility Aids*]. April,2006.

[13] Frank Dellaert,Dieter Fox,Wolfram Burgard,Sebastian Thrun. *University of Bonn*. [*Monte Carlo Localization for Mobile Robots*].

[14] Gilles Caprari, Kai O. Arras, Roland Siegwart. *Swiss Federal Institute of Technology Lausanne (EPFL)*. [*The Autonomous Miniature Robot Alice: from Prototypes to Applications*].

[15] Todd Litman. *Victoria Transport Policy Institute*. [*Autonomous Vehicle Implementation Predictions*]. July,2017

[16] Dipl.-Ing. Arnd Buschhaus, Andreas Blank, Dr.-Ing. Christian Ziegler, Prof. Dr.-Ing. Jörg Franke. *Friedrich-Alexander-University Erlangen-Nuremberg*. [*Highly Efficient Control System Enabling Robot Accuracy Improvement*].

[17] T. Hong, R. Bostelman, and R. Madhavan. *National Institute of Standards and Technology*. [*Obstacle Detection using a TOF Range Camera for Indoor AGV Navigation*].

[18] Sebastian THRUN, Wolfram BURGARD, and Dieter FOX. *University of Stanford,University of Freiburg,University of Washington*. [*PROBABILISTIC ROBOTICS*].

[19] F. Li, Michael Brady and H. Hu. *University of Oxford*. [*Visual Guidance of an AGV*].

[20] Stephen Balakirsky and Raj Madhavan. *National Institute of Standards and Technology*. [*Advancing Manufacturing Research Through Competitions*].

[21] Roger Bostelman, Richard Norcrossa, Joe Falcoa, Jeremy Marvel. *National Institute of Standards and Technology*. [*Development of Standard Test Methods for Unmanned and Manned Industrial Vehicles Used Near Humans*].

[22] Jeremy A. Marvel and Roger Bostelman. *National Institute of Standards and Technology*. [*Test Methods for the Evaluation of Manufacturing Mobile Manipulator Safety*].January,2016.

[23] Roger Bostelman. *National Institute of Standards and Technology*. [*Navigation Performance Evaluation for Automatic Guided Vehicles*].

[24] Michael Shneier,Roger Bostelman. *National Institute of Standards and Technology.* [*Literature Review of Mobile Robots for Manufacturing*].

[25] Suhua Hsieh and Kuei-Huang Max Lin. *National Taiwan University.* [*AGVS Mechanism*].

[26] J. Borenstein,H.R. Everett,L. Feng, D. Wehe. [*Mobile Robot Positioning Sensors and Techniques*].

[27] S300 Laser scanner.*SICK.Co. Sensor Intelligence.* [*Safety Laser Scanner*].

[28] InSystems Automation GmbH,
`http://www.insystems.de/`

[29] Path Planning for Semi-automated Simulated Robotic Neurosurgery,
`https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4687488/`

[30] A Comparison of Automated Material Transport,
`https://www.ottomotors.com/resources/info/agv-vs-sdv`

[31] InvenSense's 6-axis,
`https://www.invensense.com/products/motion-tracking/6-axis/`