MICROCONTROLLER DESIGN FOR AN OMNIDIRECTIONAL MOBILE ROBOT
(PROGRAMMING FOR SIMULTANEOUS LOCALIZATION, MAPPING, AND EXPLORATION)

By

Brian Wijaya
11301057

BACHELOR'S DEGREE
in

MECHANICAL ENGINEERING – MECHATRONIC CONCENTRATION
FACULTY OF ENGINEERING AND INFORMATION TECHNOLOGY

SWISS GERMAN UNIVERSITY

SWISS GERMAN UNIVERSITY
The Prominence Tower
Jalan Jalur Sutera Barat Kav 15, Alam Sutera
Tangerang, Banten 15143 – Indonesia

August 2017

Revision after Thesis Defense on July 27th 2017

STATEMENT BY THE AUTHOR

I hereby declare that this submission is my own work and to the best of my knowledge, it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at any educational institution, except where due acknowledgement is made in the thesis.

Brian Wijaya

Student _____ Date _____

Approved by:

Andreas Schwung

Prof Andreas Schwung 07.08.2017

Thesis Advisor _____ Date _____

Dr. Rusman Rusyadi, B.Eng, M.Sc

Thesis Co-Advisor _____ Date _____

Dr. Ir. Gembong Baskoro, M.Sc

Dean _____ Date _____

SGU® SWISS GERMAN UNIVERSITY

ABSTRACT

MICROCONTROLLER DESIGN FOR AN
OMNIDIRECTIONAL MOBILE ROBOT (SLAM AND EXPLORATION)

By

Brian Wijaya
Prof. Andreas Schwung, Advisor
Dr. Rusman Rusyadi, B.Eng, M.Si, Co-Advisor

SWISS GERMAN UNIVERSITY

This paper is particularly intended to create a map through SLAM algorithm, in particular using Hector SLAM and explore the unknown environment using the Hector Exploration node. SLAM is an algorithm that used for creating a virtual map that can be used for environment sensing. Using the *mecanum* wheel or omnidirectional mobile robot where laser scanner sensor is used. RPLIDAR A2 is chosen for the laser scanner due its low-cost price while still producing a good set image of map. ROS framework is used to program for the robot since it provide ready-to-use packages regarding to SLAM. After the map was created, it will be used for dynamic path planning where it can provide the static map for the basic of the dynamic path planning. Through this process a sufficient map is created.

Keywords: SLAM, ROS, Omnidirectional, Mecanum Wheel.

© Copyright 2017
By Brian Wijaya
All rights reserved

SGU®
SWISS GERMAN UNIVERSITY

DEDICATION

I dedicated my work for my family, God, SGU, and Indonesia

SGU®
SWISS GERMAN UNIVERSITY

ACKNOWLEDGEMENTS

I would like to thank God for this opportunity. Furthermore, I would also like to thank my team in Fachhochschule Sudwestfallen Soest for their support and patience. Prof. Andreas Schwung and Dr. Rusman Rusyadi, B.Eng, M.Sc thanks for helping and guiding me toward this project. Thanks to Prof. Andreas Schwung for the opportunity to be involved in this project, and the chance to complete my degree in mechatronics. And last but not least my family for keep supporting me throughout this period of time.

SGU®
SWISS GERMAN UNIVERSITY

TABLE OF CONTENTS

Page

STATEMENT BY THE AUTHOR 2
ABSTRACT 3
DEDICATION 5
ACKNOWLEDGEMENTS 6
CHAPTER 1 - INTRODUCTION 12
1.1 Background 12
1.2 Research or Design Problems 13
1.3 Objectives 13
1.4 Significance of Study 13
1.5 Scope and limitations 13
1.6 Thesis Overview 14
CHAPTER 2 - LITERATURE REVIEW 15
2.1 Theoretical Perspectives 15
2.1.1 Mecanum Wheel 15
2.1.2. Laser Scanner Range Finder 16
2.1.3 Raspberry Pi 17
2.1.4. Robot Operating System (ROS) 18
2.1.5 Simultaneous Localization and Mapping (SLAM) 19
2.2 Previous Studies 21
CHAPTER 3 – RESEARCH METHODS 24
3.1 Materials and Equipment 24
3.1.1 Laser Scanner 24
3.2 Analytical Method 25
3.2.1 Raspberry PI 25

SWISS GERMAN UNIVERSITY

3.2.2 Simulation 25
3.2.3 RPLIDAR in ROS 29
3.2.4 Hector SLAM 29
3.2.5 Combining simulation and real time 29
3.2.6 Navigation (Exploring) 30
3.2.7 Data Publishing 30
CHAPTER 4 – RESULTS AND DISCUSSIONS 31
4.1 Initial Evaluation 31
4.1.1 Sensor Test 31
4.1.2 Simulation Result 32
4.1.3 Real Time Result 38
4.2 Data Analytical 44
4.2.1 Simulation Data 44
4.2.2 Real Time Data 45
CHAPTER 5 – CONCLUSIONS AND RECCOMENDATIONS 47
5.1 Conclusions 47
5.2 Recommendations 48
GLOSSARY 50
REFERENCES 51
APPENDICES 53
A Simulation Code 53
A.1 Simulation launch file 53
A.2 Costmap.yaml 54
A.3 Mapper.yaml 55
A.4 get_map_client.cpp 56
A.5 Explore.client 57
A.6 navigator.yaml 58
A.7 Operator.yaml 59
A.8 ROS.yaml 59
A.9 Simualtion.rviz 60

SWISS GERMAN UNIVERSITY

A.10 floorplan.inc 68
A.11 set_goal_client.cpp 68
A.12 hokuyo.inc 69
A.13 p3t.inc 70
A.14 simulation.yaml 72
B Real Time Code 73
B.1 Real time launch file 73
B.2 Hector Mapping 73
B.3 rplidar.launch 74
B.4 geotiff.launch 75
B.5 Path2Speed.cpp (Courtesy of Mohammad Rizky) 76
B.6 calc_speed.ino (Courtesy of Julian Bernauer) 81
B.7 final_rt_prog.ino (Courtesy of Julian Bernauer) 83
B.8 read_i2c.ino (Courtesy of Julian Bernauer) 84
B.9 read_serial.ino (Courtesy of Julian Bernauer) 85
B.10 read_serial_raspi.ino (Courtesy of Julian Bernauer) 86
B.11 read_v.ino (Courtesy of Julian Bernauer) 88
B.12 send_data.ino (Courtesy of Julian Bernauer) 88
B.13 exploration.cpp (Courtesy of Mohammad Rizky) 90
B.14 sendspeed.cpp (Courtesy of Mohammad Rizky) 93
C RPLIDAR A2 Data Sheet [14] 97
CURRICULUM VITAE 100

SWISS GERMAN UNIVERSITY