TU/e Technische Universiteit
Eindhoven
University of Technology

Department of Mechanical Engineering

Embedded motion control 2019

(4SC020)

Design document – Group 2

Student name:	**Student ID:**
Bob Clephas	1271431
Tom van de laar	1265938
Job Meijer	1268155
Marcel van Wensveen	1253085
Anish Kumar Govada	1348701

Version 1, May 2019

Introduction

To complete the two assignments for the course “Embedded motion control” specific software must be written. In this design document the global architecture of the software is explained, and the given constraints and hardware is listed. This document is a first draft and will be updated during the project.

Requirements and specifications

The requirements and related specifications are listed in the following table. The listed specifications are required for the final assignment, the underlined specifications are needed for the final challenge.

Requirements	Specifications
Accomplish predefined high-level tasks	1. Find the exit (Back wheels over the finish line) 1. Reach a predefined cabinet
Knowledge of the environment	1. Location of walls (via corner points) 2. Location of the doors (via corner points) 3. Location of the cabinets (location: TBD) 4. At what level – 2D (Top view) 5. Accuracy (< 0.1 m resolution)* 6. Shapes (axis aligned Lines)
Knowing where the robot is in the environment	1. Level (2D) 2. XY precision (<0.1 m)* 3. Orientation precision (<10 degree)*
Being able to move	1. 0.5 [m/s] translational speed / 1.2 [rad/sec] 2. Maximum acceleration / braking TBD after first experiment session.* 3. path following (<0.1m)*
Do not bump into the wall	1. Stay at least (0.1m)* of the wall
Do not stand still for longer than 30 sec	1. It is not allowed to stand still for longer than 30 seconds
Finish as fast as possible	1. Within 5 minutes
Coding language	1. Only allowed to write code in C++ coding language 2. GIT version control must be used

* Initial guess, exact specifications must be determined after first experiment

Components and functions

The components and their functions are split in software components and hardware components.

Software components

Software Components	General functionality
World model	- Storing all the relevant data (Map / tasks / position). - Data communication between the other components (All data goes through the world model)

Preceptor	- Reading sensor data and creating a local map. - Locate the robot on this local map. - ~~Align global map with local map.~~
Task manager	- Dividing upcoming tasks to other modules when current task is finished. - Interrupt a task if needed (safety / Human input)
Path planner	- Create a path from the combined map, current position and the desired position
Drive controller	- Actuates the robot such that is follows the planned path (keep speed and acceleration in mind)

Hardware components

Hardware Components	**General functionality**
PICO Robotic platform - Jazz telepresence robot	General framework with all the hardware. This framework will allow to execute the assignments
Sensors: - Laser range finder - Wheel encoders - 170^{0} wide angle camera	 Scan environment and detect objects Determine the traveled distance by the wheels Can be used for vision system (object detection for example)
Actuators: - Holonomic base (omni-wheels) - Pan-tilt unit for head	 Allows the robot to move on the ground Can be used to move the head with the display and camera
Computer - Intel I7 processor - OS: Ubuntu 16.04 (64-Bit) - ROS with own software layer	 Perform computations Software that allows execution of programs Allows to easily make connections between software and hardware

Environment

The environments for both assignments will meet the following specifications:

Escape room challenge

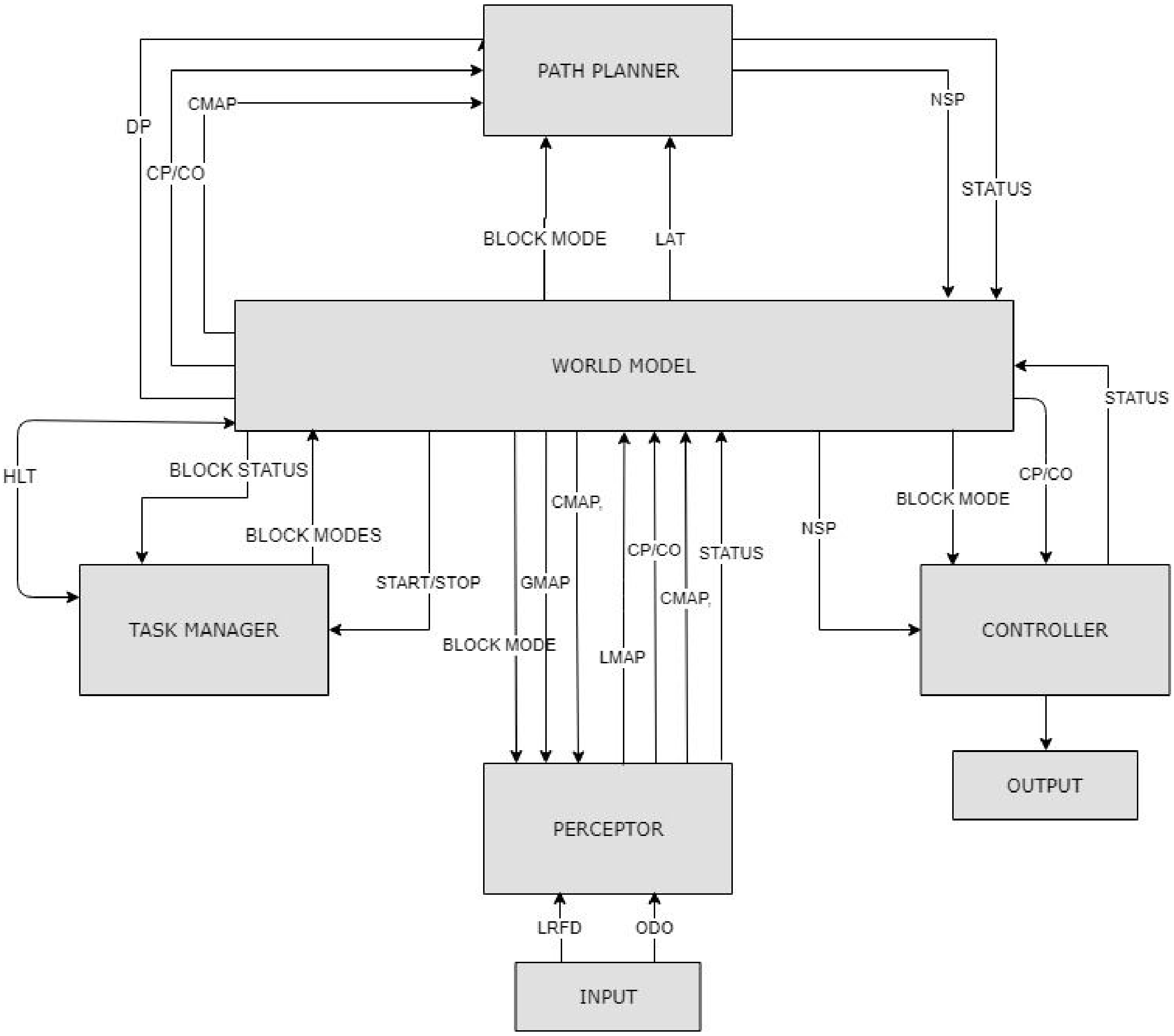
- Rectangular room, unknown dimensions. One opening with a corridor
- Starting point and orientation is random, but equal for all groups
- Opening will be perpendicular to the room
- Far end of the corridor will be open
- Wall will not be perfectly straight, walls of the corridor will not be perfectly parallel
- Finish line is at least 3 meters away from the corridor opening, walls of the corridor will be a little bit longer

Final challenge

- Walls will be perpendicular to each other
- Dynamic elements will be in the area
- Not all objects will have the same orientation
- Multiple rooms with doors

Interface

The overall software is split in several building blocks.



KEY :

LRFD : LASER RANGE FINDER DATA

LMAP : LOCAL MAP

CMAP : COMBINED MAP

DP : DESTINATION POSITION

LAT : LOCATION ACTIVE TASK

BLOCK MODE : INITIALIZE/ STAND BY/ EXECUTE

ODO : ODOMETRY DATA

GMAP : GLOBAL MAP

CP/CO : CURRENT POSITION/ ORIENTATION

NSP : NEXT SET OF POSITIONS (OPTIMAL PATH)

HLT : HIGH LEVEL TASK

BLOCK STATUS : ERROR/ BUSY/ FINISH