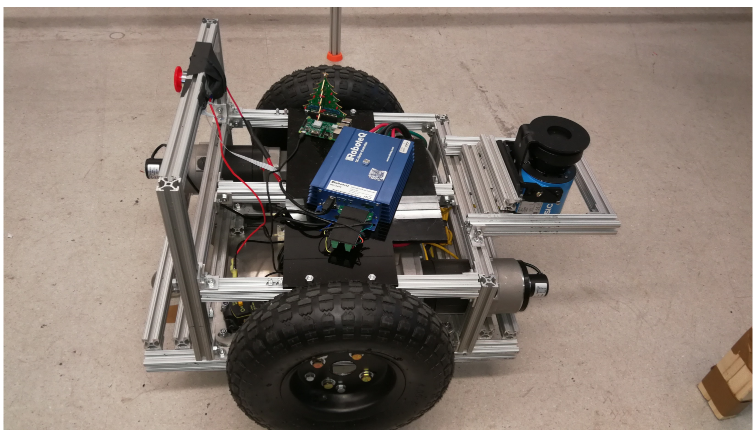
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON

ARCHITECTURAL DESIGN SPECIFICATION
SENIOR DESIGN I
SUMMER/FALL 2017



TEAM UGV
UNMANNED GROUND VEHICLE

CHRIS COLLANDER
DARRELL RASCO
CHASE HUFFMAN
PAUL ASYN

Revision History

Revision	Date	Author(s)	Description
0.1	8.08.2017	CC	document creation, flow diagram
0.2	8.10.2017	DR	introduction, overview
0.3	8.14.2017	CH	overview, definitions, lidar subsystems, hardware interface
1.0	8.15.2017	PA	Y layer, hardware interface, clean up, release

CONTENTS

List of Figures

List of Tables

1 Introduction

Our product is an unmanned ground vehicle(UGV). Our product will have three different architectual layers. The first layer will be consist of the LIDAR and its sensors which will be referred to as the LIDAR layer. The second layer deals with the PC, called the PC layer. This layer specifically deals with processing the LIDAR data, mapping the data, and processing user commands. The last layer is the Hardware interface layer. This layer specifically deals with processing the commands from the computer and turning them into actions by the motors.

2 System Overview

2.1 LIDAR Layer Description

The LIDAR layer will be the sensory layer for our UGV. The LIDAR uses an infrared transmitter that emits infrared light pulses into the environment. The LIDAR reciever measures the differences in wavelengths and laser return times to make a digital representation of the environment.

2.2 PC Layer Description

The PC layer takes in the LIDAR data through the a rosnode, which is processed by the Hector slam algorithm. The hector slam algorithm maps the LIDAR data. The PC will process user commands with the map data to publish motor commands to the Rasberry pi.

2.3 Hardware interface layer Description

The hardware interface layer deals specifically with processing movement commands from the PC into motor actions.

3 Subsystem Definitions & Data Flow

This section breaks down your layer abstraction to another level of detail. Here you grapically represent the logical subsytems that compose each layer and show the interactions/interfaces between those subsystems. A subsystem can be thought of as a programming unit that implements one of the major functions of the layer. It, therefore, has data elements that serve as source/sinks for other subsystems. The logical data elements that flow between subsystems need to be explicitly defined at this point, beginning with a data flow-like diagram based on the block diagram.

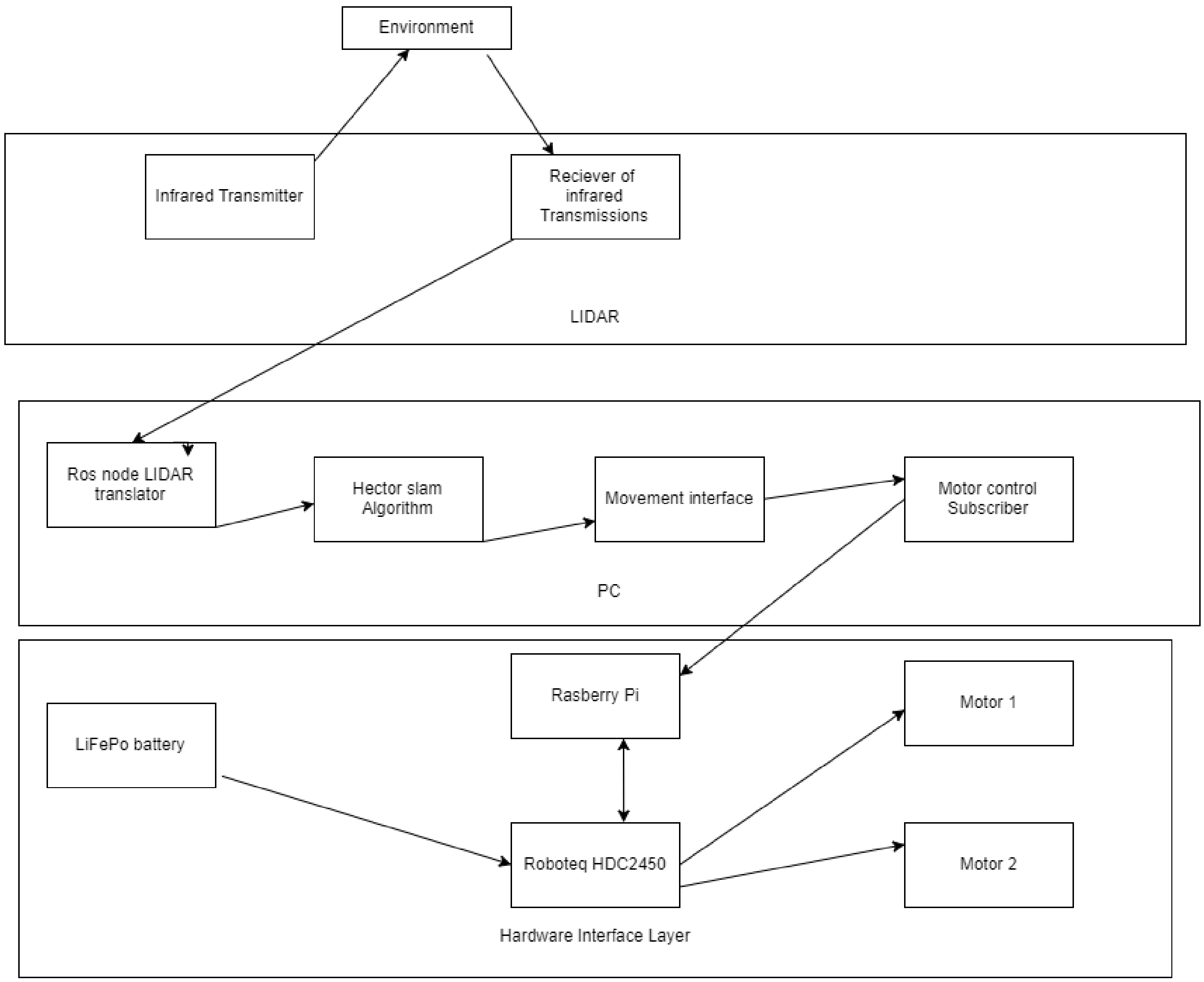


Figure 1: A simple data flow diagram

4 LIDAR Subsystems

4.1 Infrared Transmitter

This subsection is the infrared transmitter. The infrared transmitter transmits pulses of infrared lights into the environment.

4.1.1 Assumptions

The infrared transmitter is properly calibrated to transmit the corrects pulses at the correct intervals.

4.1.2 Responsibilities

To transmit the infrared pulses into the environment.

4.2 Reciever of infrared transmissions

The receiver of infrared transmission measures the differences in wavelengths and laser return times to determine its physical surroundings.

4.2.1 Assumptions

The reciever is properly calibrated, and provides a correct represenation of the physical environment. The ros node LIDAR translator can gather the information from the reciever.

4.2.2 Responsibilities

To collect data about the physical space around the UGV, and send the data to the ros node.

5 Y Layer Subsystems

5.1 Ros node LIDAR translator

This subsection is the Ros node LIDAR translator. The ros node translator takes in LIDAR data and does a basic translation for compatability with the hector slam algorithm.

5.1.1 Assumptions

The ros node LIDAR translator should be able to interface with the LIDAR.

5.1.2 Responsibilities

The responsibility of this subsystem is to properly convert the LIDAR data into data compartible with the hector slam algorithm.

5.1.3 Subsystem Interfaces

Each of the inputs and outputs for the subsystem are defined here. Create a table with an entry for each labelled interface that connects to this subsystem. For each entry, describe any incoming and outgoing data elements will pass through this interface.

5.2 Hector slam algorithm

Uses a 2D SLAM system based on scan matching and 3D navigation all through an inertial sensing system. Used with data input from odometry.

5.2.1 Assumptions

Uses LIDAR to capture odometry distance for mapping.

5.2.2 Responsibilities

System responsible for most of the robot decision-making to base on. Used to figure out paths and obstacle obstruction.

5.3 Movement interface

This subsection is the movement interface. It will take in the mapping data from the hector slam algorithm and user input in conjunction with the map and output the proper motor commands.

5.3.1 Assumptions

The hector slam algorithm inputs the proper mapping data.

5.3.2 Responsibilities

The responsibility of this subsystem is to properly convert the LIDAR data into data compatiable with the hector slam algorithm.

6 Hardware interface layer

6.1 LiFePo battery

6.1.1 Assumptions

The battery provides power, connected in parallel to reach specified requirements..

6.1.2 Responsibilities

The batter provides power.

6.2 Roboteq HDC2450

The roboteq HDC2450 is a electronic DC motor controller used to power and control different motors.

6.2.1 Assumptions

The rototeq HDC2450 can properly connect to the battery. It can provide enough power to power each motor. It can power the rasberry pi.

6.2.2 Responsibilities

The rototeq HDC2450 converts commands from the rasberry pi into motor actions.

6.2.3 Subsystem Interfaces

The rototeq HDC2450 can properly connect to the battery, and the rasberry pi.

6.3 Rasberry pi

A small single board computer.

6.3.1 Assumptions

The rasberry pi can subscribe to the motors topic and relay the commands to the ROBOTEQ HDC2450 motor driver.

6.3.2 Responsibilities

The rototeq HDC2450 converts commands from the rasberry pi into motor actions.

6.3.3 Subsystem Interfaces

The rototeq HDC2450 can properly connect to the battery, and the rasberry pi.

6.4 Motor 1/2

2 motors attached to the UGV.

6.4.1 Assumptions

They can provide enough power to move the UGV.

6.4.2 Responsibilities

Move the ugv.

6.4.3 Subsystem Interfaces

The motors connect to the motor driver, which powers the motors accordingly.

REFERENCES