Swimming Swarm

Multi-Agent Underwater Robotics System

ECE 4011/4012 Capstone Design Project

User Manual

Team Hydra

Advisor | Dr. Mick West

Priyansh Bhatnagar

Computer Engineer

Trevor Jones

Electrical Engineer

Hemanth Koralla

Computer Engineer

Aneri Muni

Electrical Engineer

Vishnu Perumal

Electrical Engineer

Michael Tatum

Computer Engineer

May 2nd, 2018

Table of Contents

1. Purpose of this Document

This is a technical guide for use with the swimming swarm prototype agent, the HydraBot. In this document is a default wiring setup table for the electronic components in each HydraBot unit, considerations for programming the bot’s microcontrollers, and suggestions for reliably maintaining the waterproof chassis.

2. Wiring of Internal Components

Component	Qty	Connects To	Connection Description	Comments
Raspberry Pi	1	-Arduino Uno -Power Distribution Board	To Uno: USB To Dist. Board: Pin 2 to 12V+ Pin 6 to 12V-	
Arduino Uno	1	-Raspberry Pi -IR receivers -IR transceivers -ESCs		The Arduino interfaces and receives power from the Pi with a USB cable
Power Distribution Board	1	-Battery -Raspberry Pi -ESCs	To Battery: Battery+ pad to p+ Battery- pad to p- To Raspberry Pi: 12V+ to pin 2 on Pi 12V- to pin 6 on Pi	
Battery with protection board	1	-Power Distribution Board	To Distribution Board: p+ to battery+ p- to battery-	
Raspberry Pi Camera	1	-Raspberry Pi		
IR Receivers	4	-Arduino Uno	To Uno: Pin 1 to digital pin 11 Pin 2 to Gnd Pin 3 to 5V	The four IR receivers are connected in parallel.
IR Transceiver	1	-Arduino Uno	To Uno: Pin 1 to 5V Pin 2 to digital pin 3 Pin 3 not connected Pin 4 to Gnd Pin 5 to 5V Pin 6 to Gnd	
ESCs	2	-Power Distribution Board -Arduino Uno	To Dist Board: Red Power Wire to + Black Gnd Wire to - Arduino Uno: Gnd to Gnd Signal to digital pin 8(left motor),9(right motor)	Each ESC's black and red power wires are soldered to the distribution board. There are four ESC pad sets simply labeled + and - on the distribution board and it does not matter which sets are used. The left and right ESCs are controlled individually by the Arduino, so correctly matching the signal wires to the Arduino pins is crucial for the bot to move the way it should.

3. Programming the HydraBot

Each HydraBot is equipped with two microcontrollers and is thus executing two separate programs while operating. The Raspberry Pi is by default running a Python 3 script which is continuously processing images from the Pi camera in search of its neighboring bots, and is in charge of telling the Arduino how and when to control the bot's motors by simple serial commands sent through the USB connection. The Arduino is running a C++ script which waits for serial commands from the Pi to tell it which direction to move the HydraBot. IR communication is also programmed onto the Arduino where the Pi is only in charge of telling the Arduino to turn it's IR communication routines on or off.

Uploading your own code to either microcontroller is thus as simple as interfacing and uploading code to an Arduino Uno or Raspberry Pi, both of which have numerous resources and guides available online to help getting started. Nothing was done in team Hydra's design which changes the normal routine of programming each microcontroller individually, and the only considerations that must be taken when loading new code to either the Arduino or the Pi is that it works with the other microcontroller's programming, and that any changes to the programmed pinout of sensors and ESCs be physically made as well.

4. Maintaining a Waterproof Chassis

The HydraBot is a prototype that is intended to be relatively easy to modify. This means having simple access to the internal components of the bot for uploading code or adding new sensors etc. As such the waterproofing of the chassis is not as robust as the final design's will hopefully be, and therefore the user must be vigilant in maintaining the chassis' waterproofing whenever using the bot.

- Securing the Lid - The top of the HydraBot is secured by applying a generous amount of petroleum jelly along the entirety of the point of contact between the lid and body, snapping down the four clips around the perimeter, and then tightening the seal at the round edges by placing rubber bands around the entire chassis so that looking at the top of the bot the user sees an 'X' across the lid. It is recommended that the petroleum jelly be reapplied frequently because it will wear off from repeatedly removing and securing the lid.

- Removing Lid - It is recommended that the user be careful to not spill water into the chassis when removing the lid, as there will be some water resting on top of it. After removing the lid and making sure hands are dry, check the bottom of the container for any dampness or drops that may have accrued during any recent operation. If any moisture is found there may be a leak in the waterproof chassis.

- Finding a Leak - If it is suspected that the waterproofing of the chassis has been compromised it is recommended that the user shut down all of the bot's electronic systems and remove anything not bound to the chassis. If a leak has sprung somewhere in the chassis it is important to find and reseal it before further operating the HydraBot in water. To find a leak in the chassis, push the empty chassis into water and look for escaping air bubbles on the exterior of the container. The source of the bubbles is the point of leakage and it must be resealed.

- Sealing a Leaky Chassis - If a leak had been found let the chassis dry completely before attempting to reseal the weak point. Once dry, apply a generous amount of silicone-based sealant to the weak point and let cure for at least 24 hours. Once cured, the chassis can be put in water again to test for leaks. If no leaks are found the user has successfully resealed the chassis, and the HydraBot may resume operation.