ASV Documentation

In addition to Teledyne's manuals…

I. Contact Info

John Lee (johlee@hmc.edu), Jane Wu (jhwu@hmc.edu)

II. Useful Documents

Q-Boat Quick Start Guide
Q-Boat Manual
ADCP Manual

III. Overview

We have adapted the Oceanserver Q-Boat to enable autonomous control without purchasing Teledyne's Autonomous Control Unit. This document contains information on how to deploy the ASV and all hardware components that were modified.

IV. Manual RC Control

If autonomous control is not desired, the ASV can be controlled using RC without any modifications. Just make sure the Raspberry Pi is not powered (so it doesn't send any messages to the motors or servo).

V. Autonomous Control

See https://github.com/hmc-lair/asv-river-mapping

VI. Using BOTH RC and Autonomous Control (very important)

While the Pi is turned on (...and connected to the VCU), the RC transmitter can only be in differential control mode ("L Steer 2 Throttles"). This is because when in "L Steer R Throttle" mode, the RC transmitter is constantly sending messages to the servo so the default servo position is straight. The Pi also sends messages to the servo, so this will cause interference/break the servo.

VII. Hardware Components

A. Vessel Control Unit (VCU)

The VCU comes with the Q-Boat, and communicates directly with the motors/servo. There is a 25-pin serial expansion port on the side that you can connect to for controlling the motors/servo. If an Autonomous Control Unit was not purchased, the VCU comes with a plug for the serial port.

The serial port has custom pins. Do not connect any store-bought 25-pin serial cable to it, or the servo *will* burn out (spoken from experience). As a result, we have created a custom serial connector to enable communication between the VCU and the Arduino/Pi. There is a separate serial cable for the left

(electrical taped) and right motors. The servo signal (green) and ground (black) wires also come out of the connector.

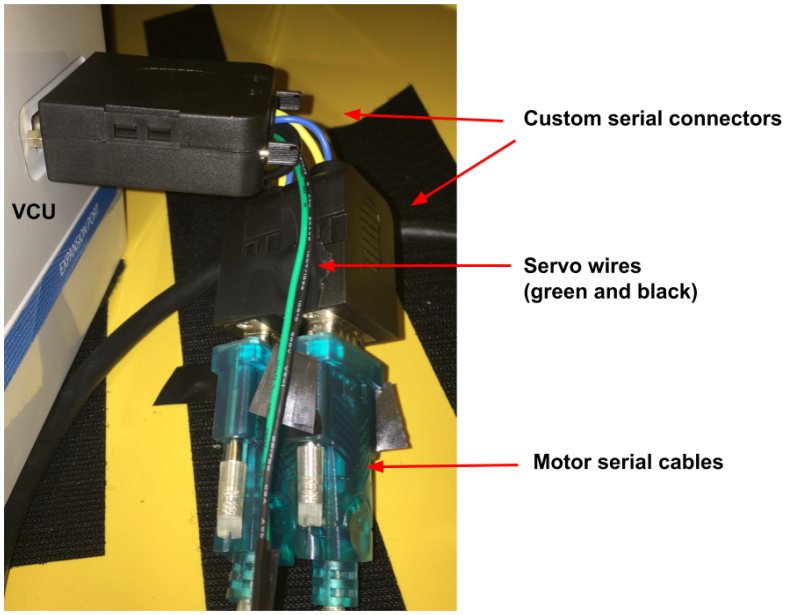

Fig. 1: VCU custom serial connector.

B. Arduino Uno

The Arduino Uno connects directly to the servo and the IMU. It also communicates via USB with the Raspberry Pi to send and receive data.

Arduino Pinout:

1. Servo:
    a. Pin 11 - Signal
    b. GND - Ground
2. IMU:
    a. A4 - SDA
    b. A5 - SCL
    c. 5V - VIN
    d. GND - Ground

C. Raspberry Pi

The Pi is where all the computation on the ASV takes place.See code documentation for details.

USB ports on the Pi:

- Arduino
- ADCP
- GPS

- Left and right motors
- XBee

Note: If the power bank is not sufficiently charged, the Pi will complain about low voltage (the power LED will light up red, even after startup). This will sometimes prevent you from ssh-ing into the Pi. Charge the power bank and this problem should go away.

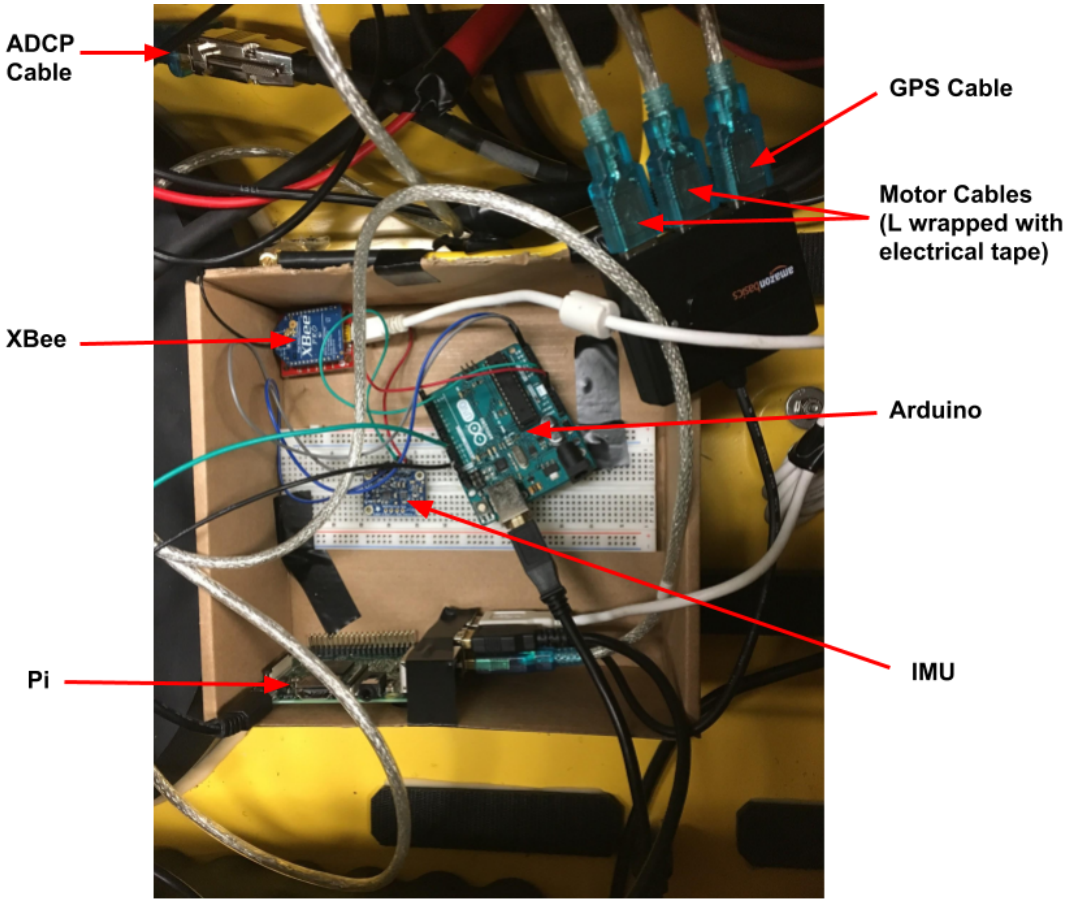

Fig. 2: Raspberry Pi connections to various hardware components.

D. XBee

The XBee is used to communicate between the Pi and a shoreside computer, which is connected to a different XBee. There is an antenna for the XBee on the exterior of the boat (identical to RC antenna).

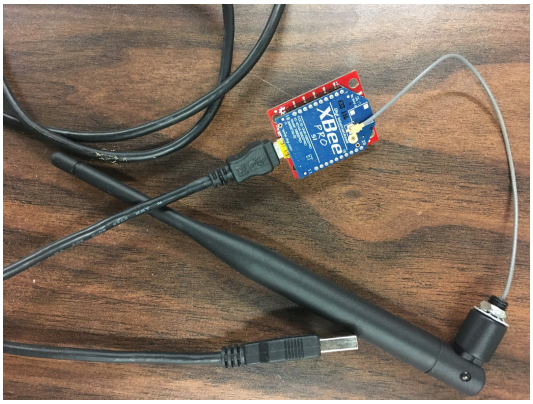

Fig. 3: Example XBee with antenna and USB cable.