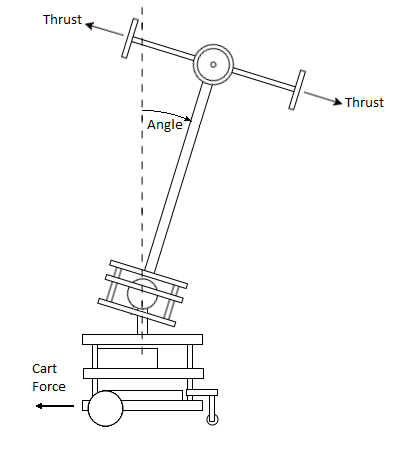
**1 Introduction** – Talk about what it is I’m doing. Why I’m doing it. What I expect to see. Aims/objectives. Report structure.

**1.1 Project conception**

The project details the steps taken in the construction, the control and the stabilisation of the Quadrotor Top Inverted Pendulum (Q-TIP) designed by Grifo[REF]. Figure XXX illustrates the concept design of the system.

FIGURE XXX 

The system consists of a pendulum and a cart connected by a 3 degree of freedom (DOF) ball joint. The cart moves around by instruction sent by the operator via a mobile application (app). The pendulum arm is then stabilised by four propulsion units placed laterally at the top of the pendulum, each unit consisting of a motor and a propeller. The project is defined by a series of aims and objectives.

**1.2 Project goals**

Aims: Finish construction of the robot. Apply controller to stabilise the pendulum arm. Develop a control interface for the car.

Objectives: Decide on and order necessary parts. Develop code to control the cart. Develop a mobile application for communicating with the cart via Bluetooth. CAD model and 3D print sensor mounts. Laser cut new motor mounts and IR sensor target plate. Read data from the MPU6050 registers. Setup and calibrate the Pololu IR distance sensors. Filter and combine sensor data in a complementary filter. Connect hardware, mechanical and electrical. Stabilise the pendulum arm with PID control. Combine the filter, controller and mobile app to drive the entire system while stabilising the pendulum upright. Implement essential safety procedures.

The report is structured as follows:

**2 Background** – The inverted pendulum on a cart is a widely researched example of a control system; it is most commonly studied in a single plane, with a single rotational DOF and translational DOF.

Grifo selected, designed and manufactured the major components of Q-TIP detailed in his thesis. He designed the ball joint connecting the pendulum and cart to have a variable DOF, varying from 1 to 3. He also selected the parts to be used in the system. Overall the system he designed adds up to 6 DOF.

Uhing originally designed, created and controlled a propeller mounted inverted pendulum on an omnidirectional ground robot[REF]; he implemented both PID and LQR control to successfully stabilise the system.

**2 Method** – (what did I do?):

* 3D printed parts.
* Hardware(new parts, powering(network diagram), soldering, machining, wiring, ESCs, ). – common issues
* Software(Arduino code, I2C library, DMP). – common issues
* Interface(App, Bluetooth). – common issues
* Control(DMP, complementary filter, feedback).
* Safety(Battery go boom, precautions, pictures).
* Operating Procedure
* Equations of motion (2 dof pendulum. Others?)
* Simulink models. Simscape models.

**2.1 Overview** – Q-TIP had to be constructed before any testing could be carried out. This required ordering the remaining parts, and machining the cart to attach them. Computer aided design (CAD) software was used to model and 3D print the sensor mounts to be attached to Q-TIP without any additional machining. The power distribution board (PDB) required soldering together, and the power cables for the electronic speed controllers (ESCs) were then soldered onto it and connected to the ESCs by bullet connectors. A switch was installed to toggle the main battery power supply on and off.

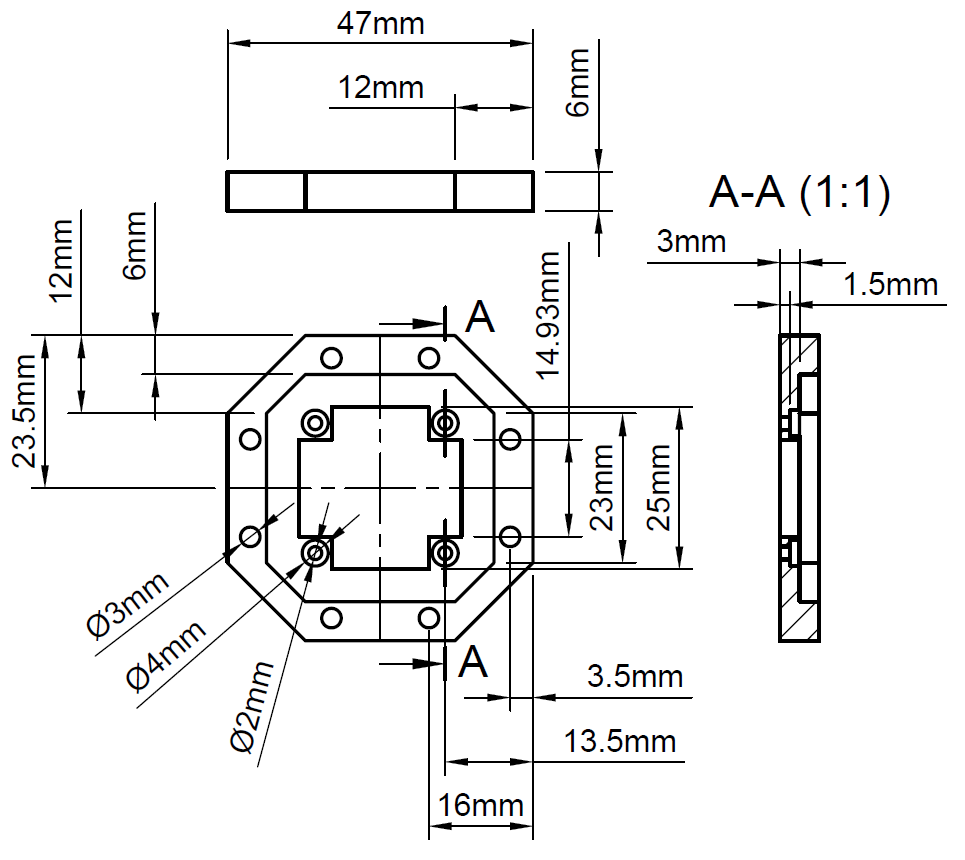
The Arduino was wired up to all the peripherals with header cables for small power supplies and data exchange. Code was uploaded to the microcontroller via the Arduino interactive development environment (IDE); the code managed the control of the ESCs, ground motors, communicated with the interface and received data from the sensors. A mobile app was used as the human user interface with Q-TIP, allowing the user to send inputs to the Arduino for cart control.

The code was split into the following sections:

* Definition of variables.
* Setup of ESCs, sensors, controller gains, interface with ground motors.
* Reading and processing sensor registers.
* Complementary filter for sensor data.
* PID feedback controller from filter sensor data to ESCs.
* Bluetooth command to cart motor actuation.
* Printing data to the serial monitor.

**2.2 3D printed parts**: Pololu IR distance sensor mounts. MPU6050 mount, reprinted cone, cut acrylic plate, redesign motor mounts.

Sensor mounts had to be designed for the Pololu infrared (IR) distance sensors and the MPU6050 gyroscope and accelerometer. These were designed in Autodesk Fusion 360. The models were designed as to fix to the top of the pendulum arm without the need for any additions machining. Figure XXX shows the MPU6050 mount, which is placed at the top of the pendulum arm, at the central axis of the quadrotor.



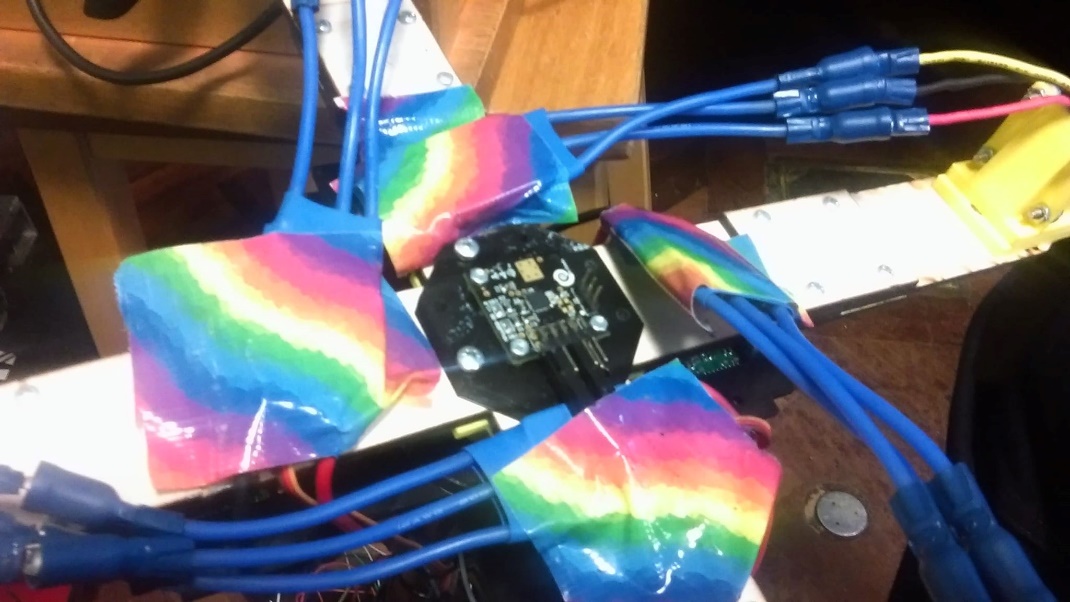


FIGURE XXX and FIGURE XXX

It can be seen from the image that the mount has a hole providing clearance for any protruding parts on the chip, this allows for the chip to be mounted flat on the quadrotor to eliminate angular offset. The dimensions are XXX matching the XXX dimensions from Grifo’s thesis [REF].

Pololu IR sensor mounts were designed to slot rigidly over the quadrotor-to-arm connector (Figure XXX). This consisted of four branches, each slotted for the attachment of an IR sensor which can be mounted parallel or perpendicular to the arm, this is demonstrated in Figure XXX.

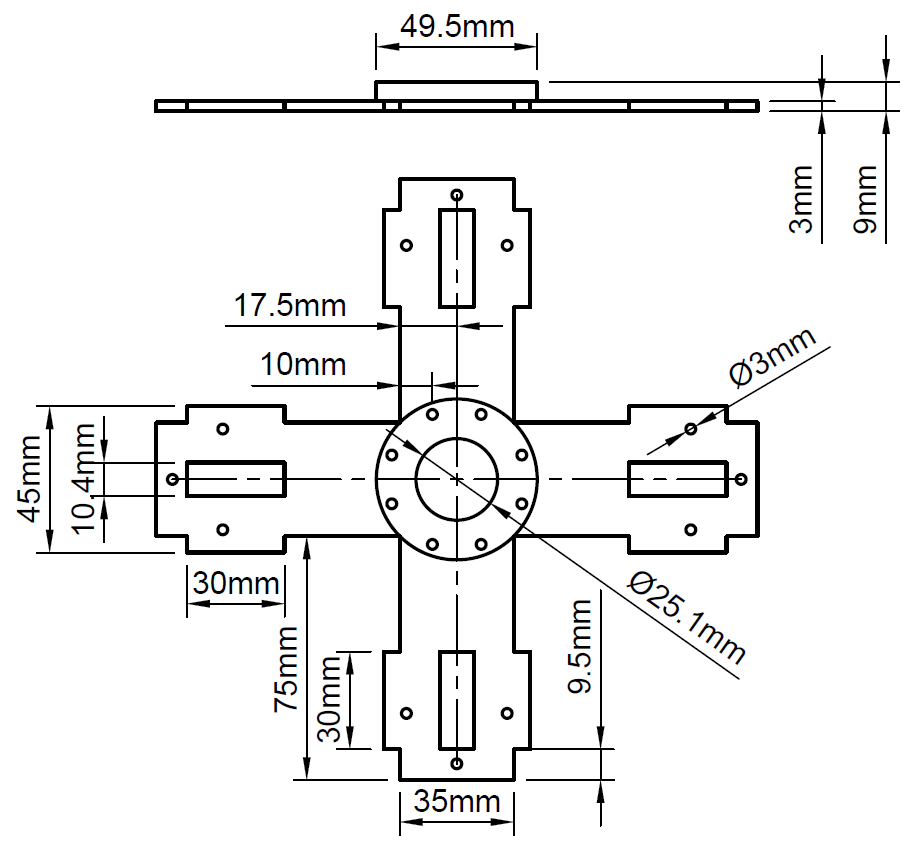


Figure XXX and XXX

The dimensions are XXX

The IR reflector base cone was re-dimensioned to allow for a more reasonable range of motion of the pendulum arm (Figure XXX). An acrylic ring was laser cut and placed on the cone surrounding the arm base to act as a reflective target for the IR sensors, displayed in Figure XXX.

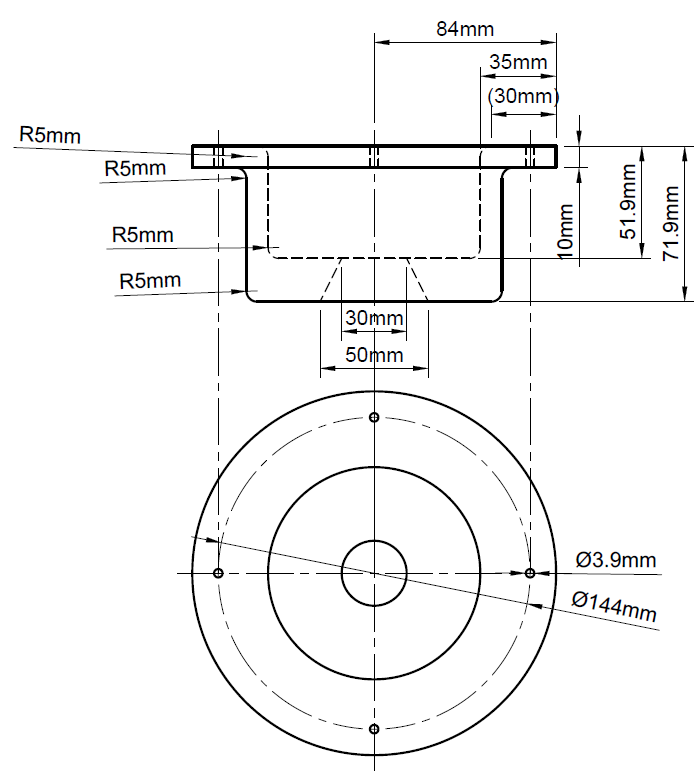
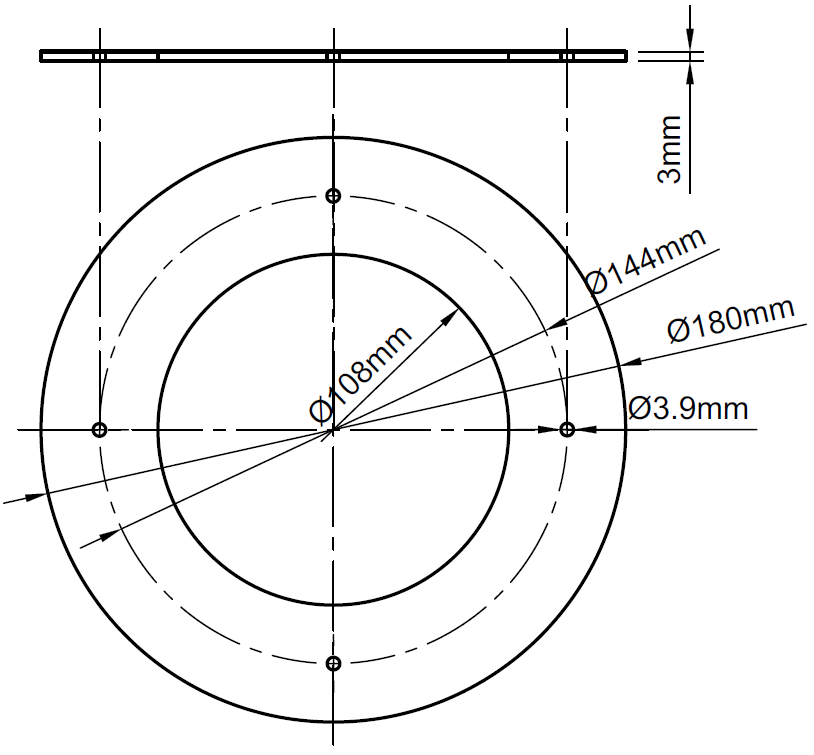


Figure XXX … XXX.

The propulsion unit wooden bases were fragile and replaced with a more sturdy laser cut design as shown in Figure XXX.

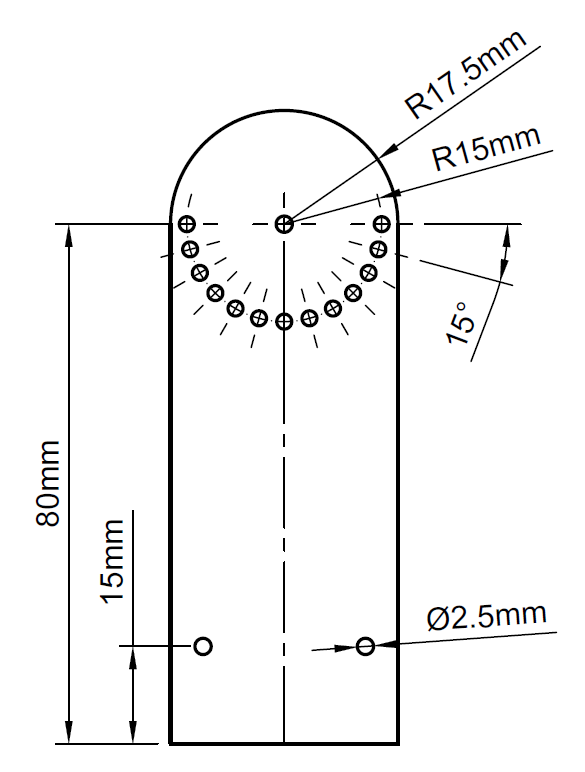


Figure XXX

This design improves the tensile properties of the quadrotor arms, greatly decreasing the risk of fracture when removing and adjusting the motors.

**2.3 Hardware**: Chose switch, connected switch to PCB. Soldered on power cables. Insulated with electrical tape. Adjusted cart dimensions. Added stabilising wheel. Cable management. Arduino setup. Cable lengths.

An 80A fused switch was selected for the battery power supply, this toggled the entire systems power supply on and off. This was position in the 2nd level of the cart, fixed by an M3 screw. A hole was machined in the top plate of the cart for the switch’s lever to pass through for easy access. Power cables of 12 AWG were installed into the switch, one end attached to a female deans connector where the battery is connected, and the other end soldered to the PDB input.



The individual parts of the PDB were soldered together, with the four pairs of ESC power cables soldered onto the PDB outputs. The power cables run adjacent to the copper pipe, both the cables and copper pipe insulated with electrical table to minimise any exposed conductive surfaces.

The XXX cart bolts are replaced with XXX bolts, and the XXX bolts are also replaced with XXX bolts, this allows for the height of the cart to be adjusted to fit the switch. The new dimensions are… XXX. A wooden base was machined to mound a castor wheel. The dimensions are XXX. This is adjustable in height as well in case any wheel needs to be changed.

ESC setup with Arduino

Motor driver setup with Arduino

HC-05 Setup with Arduino

MPU6050 setup with Arduino.

Pololu setup with Arduino.

Cable management.

**2.4 Software**: Arduino IDE used to write code. Code consisted of ESC speed control PWM, MPU6050 register reading (I2C talk about issues), Complementary filter, PID feedback, Wheel motor control PWM, App designed to send signals to Arduino.

**2.5 Interface App:** Created using MIT App inventor. Each button transmits a byte to the Bluetooth HC05 receiver when held down. The receiver then signals the Arduino with the data which processes it into a command for the motor drivers.

**2.6 Control**: Angular velocity readings from the gyroscope, and the acceleration readings from the accelerometer are transformed into angle readings. Control is done by applying a complementary filter (or DMP) to combine these readings. The filtered angle data is processed by PID feedback and used as the input for the ESC speeds.

**2.7 Safety**

**2.8 Operating procedure**

**3. Analysis**

**3.1 Equations of motion**

**3.2 Simulink models**

**3.3 Simscape models**

**3.4 Model validation and simulation results**

**4. Results**

**4.1 Hardware results**

**5. Conclusion**

**5.1 Hardware conclusion**

**5.2 Software conclusion**

**5.3 Future work**

Arduino IDE used. Libraries used. Sensor reading. Filtering(DMP and Complementary). PID. ESC update. Wheel control. App.

Modelling: 1, 2, 3 DOF, Motor modelling.

Simulations: 1 2 3 DOF.

Safety: Stuff from risk assessment. Electrical Batt, Mechanical propeller, Chemical Batt, Soldering, Copper Pipe.

Results and Discussion

Simulation performance vs real life performance.

Conclusion

References

Grifo. Uhing. Complementary filter. I2C library. (other libraries, instructions, resources used).

Appendices