Automatic 3-Axis Camera Stabilizing Gimbal

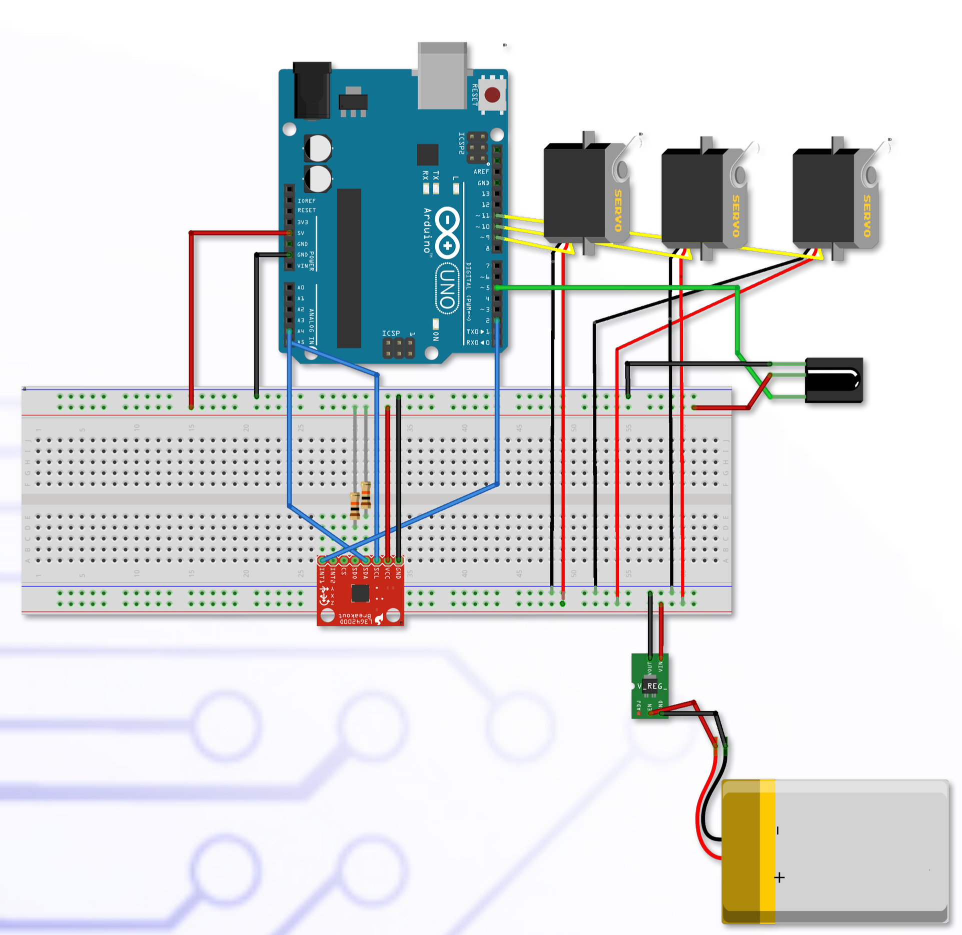
Daniel Hughes ‘18, Daming Xing ‘18
Faculty Advisors: Professor Harry Blaise, Professor Kevin Huang
Trinity College, 300 Summit Street, Hartford, CT 06106, USA

Abstract

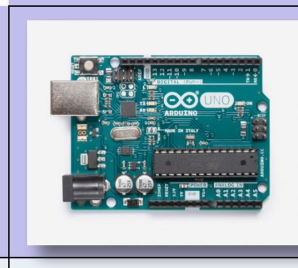
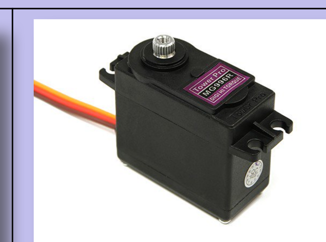
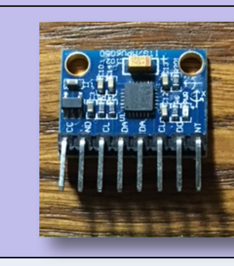
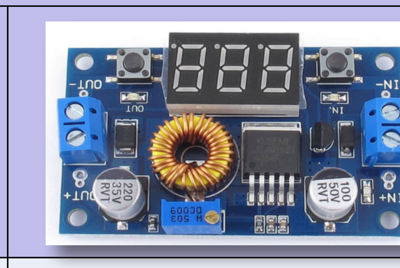
Camera stabilizing gimbals are highly specialized and extremely expensive pieces of photography and film equipment. The user base largely consists of industry professionals who demand stable footage on the move in spaces where cranes and other cumbersome devices are inaccessible. Ranging anywhere from $500 to more than $1000, there is a clear barrier into the market, especially for amateur film makers or hobbyists seeking an affordable yet effective self-stabilizing camera gimbal. Due to recent advancements in mobile film technology, stabilizing gimbals for small video devices are in particularly high demand. The aim of this project is to design, manufacture and test a small mobile camera stabilizing gimbal. The device is capable of stabilization about the pitch, roll and yaw axes and is controlled by an Arduino Uno microprocessor. Furthermore, the camera orientation stabilization is actuated via three commodity servo motors mounted to a custom 3-D printed ABS frame, thus affording an economic alternative entirely manufactured for under $300. Beyond everyday applications, this project has potential impact in the realms of professional film-making, aerospace, and defense.

Problem Statement

The overall goal of this project is to design, manufacture and evaluate an automatic 3-axis stabilizing camera gimbal for use with small electronic video devices such as mobile phones and GoPro cameras.

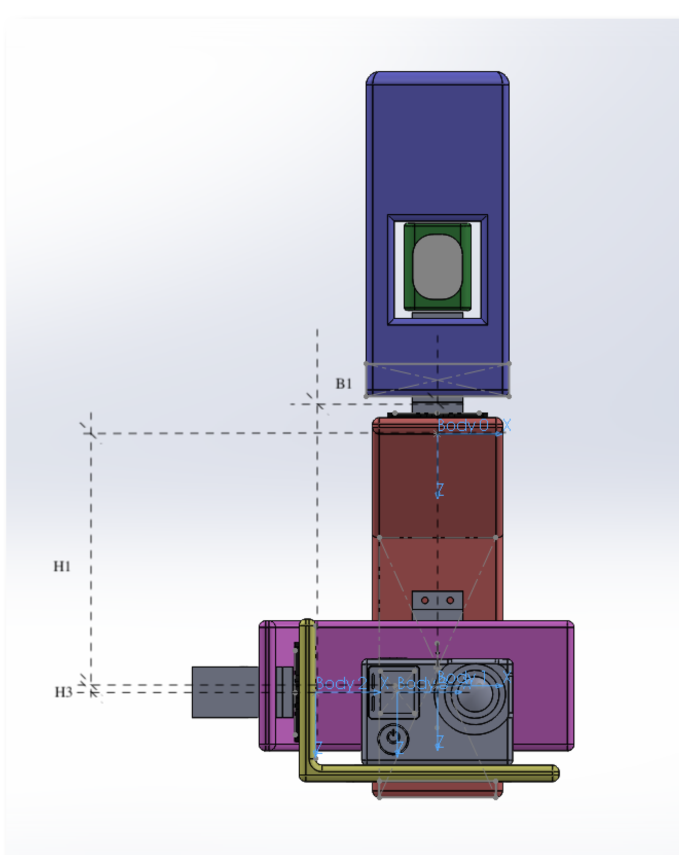


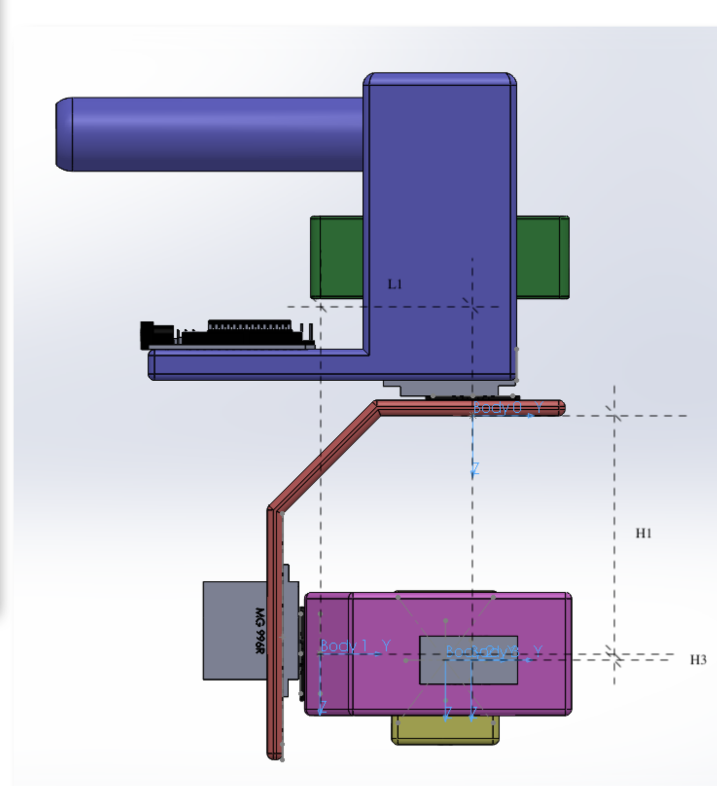
Circuitry

	Component				
Specification					
Equipment	Arduino Uno	Qunqi MG996R Metal Gear Torque Digital Servo	MPU6050 IMU Sensor	Tattu 1550mAh 14.8V LiPo Battery	Buck Converter
Advantages	• Basic C/C++ • 25g • 7.6 x 1.9 x 6.4cm	• 55g • 15kg/cm Torque • 0.4 x 0.19 x 0.43cm	• I2C Output (Use with C/C++) • 2.3 – 3.5 V • Angular Rotation and Acceleration Values	• High performance battery 14.8V • >150 cycle lifespan • Rapid recharging	• Output Current 0-5V • Output Voltage 1.25 – 36V DC • 96% Nominal Efficiency • 6.6 x 3.9 x 1.8 cm
Cost	$22.99	$10.99	$6.99	$25.99	$3.99

Kinematics

The total rotation matrix for the gimbal with R_n denoting the rotation vector about axis n, and $\boldsymbol{\theta}_n$ denoting the counterclockwise rotation about axis n. Parameters H_n, B_n and L_n are unique physical parameters of this gimbal.





$$R_1^0 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad d_1^0 = \begin{bmatrix} L1 * \sin(\theta_1) \\ L1 * \cos(\theta_1) \\ H1 \end{bmatrix}$$

$$R_2^1 = \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) \\ 0 & 1 & 0 \\ -\sin(\theta_2) & 0 & \cos(\theta_2) \end{bmatrix} \qquad d_2^1 = \begin{bmatrix} B1 * \cos(\theta_2) \\ L1 \\ B1 * \sin(\theta_2) \end{bmatrix}$$

$$R_3^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_3) & -\sin(\theta_3) \\ 0 & \sin(\theta_3) & \cos(\theta_3) \end{bmatrix} \qquad d_3^2 = \begin{bmatrix} -B1 \\ H3 * \sin(\theta_3) \\ H3 * \cos(\theta_3) \end{bmatrix}$$

$$T_1^0 = \begin{bmatrix} R_1^0 & d_1^0 \\ 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} R_2^1 & d_2^1 \\ 0 & 1 \end{bmatrix}$$

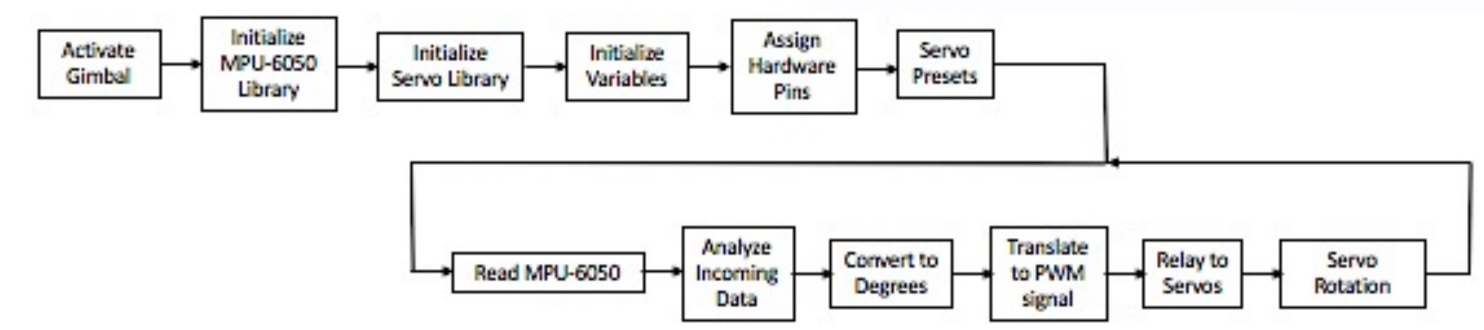
$$T_3^2 = \begin{bmatrix} R_3^2 & d_3^2 \\ 0 & 1 \end{bmatrix}$$

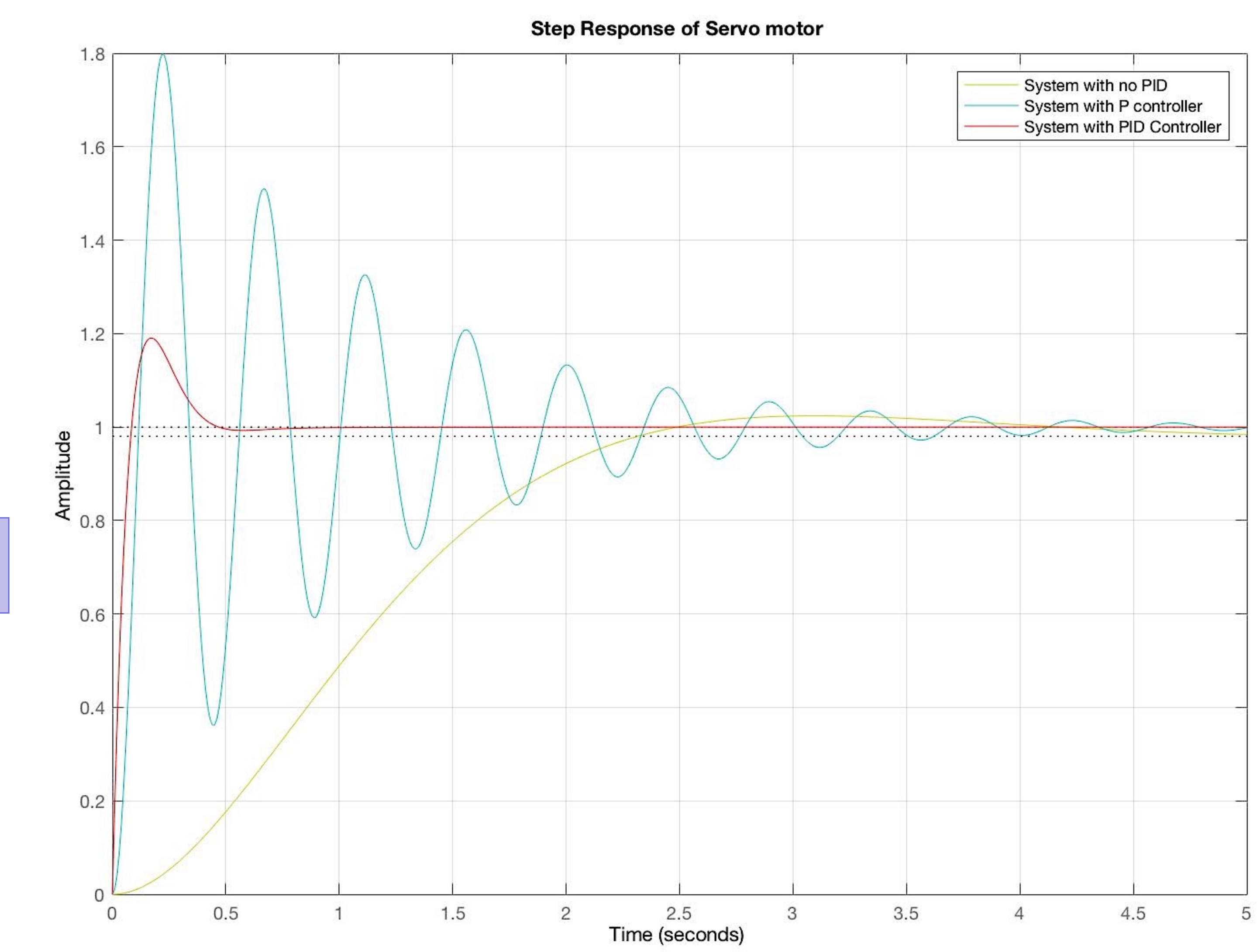
$$T_3^0 = T_1^0 T_2^1 T_3^2$$

$$T_3^0 = \begin{bmatrix} c_1 * c_2 & c_1 * s_2 * s_3 - c_3 * s_1 & s_1 * s_3 + c_1 * c_3 * s_2 & H_3 * s_1 * s_3 \\ c_2 * s_1 & c_1 * c_3 + s_1 * s_2 * s_3 & c_3 * s_1 * s_2 - c_1 * s_3 & 2 * L_1 * c_1 + H_3 * c_1 * s_3 + H_3 * c_3 * s_1 * s_2 \\ -s_3 & c_2 * s_3 & c_2 * c_3 & s_3 + 2 * B_1 * s_2 + H_3 * c_2 * c_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

System Architecture

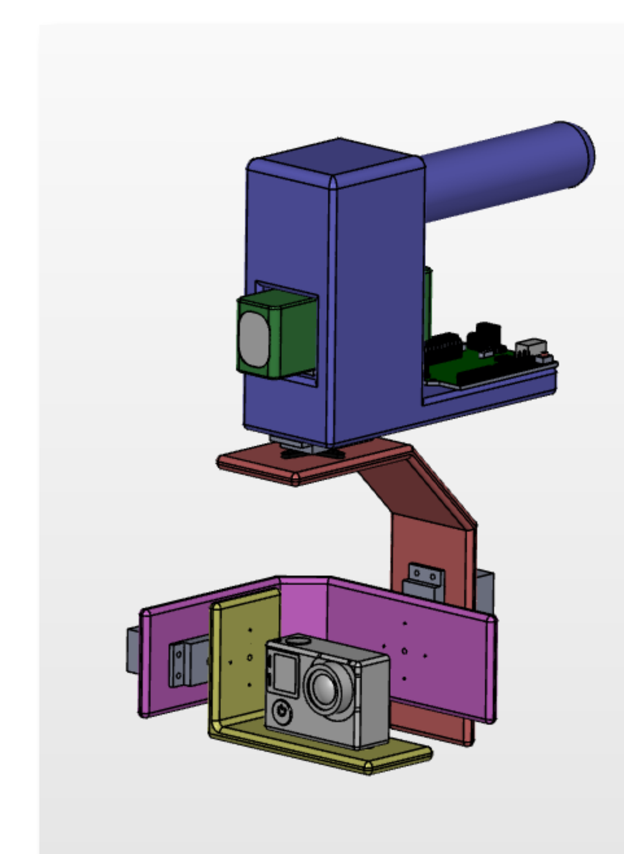
Overall system topology and Servomotor PID response.

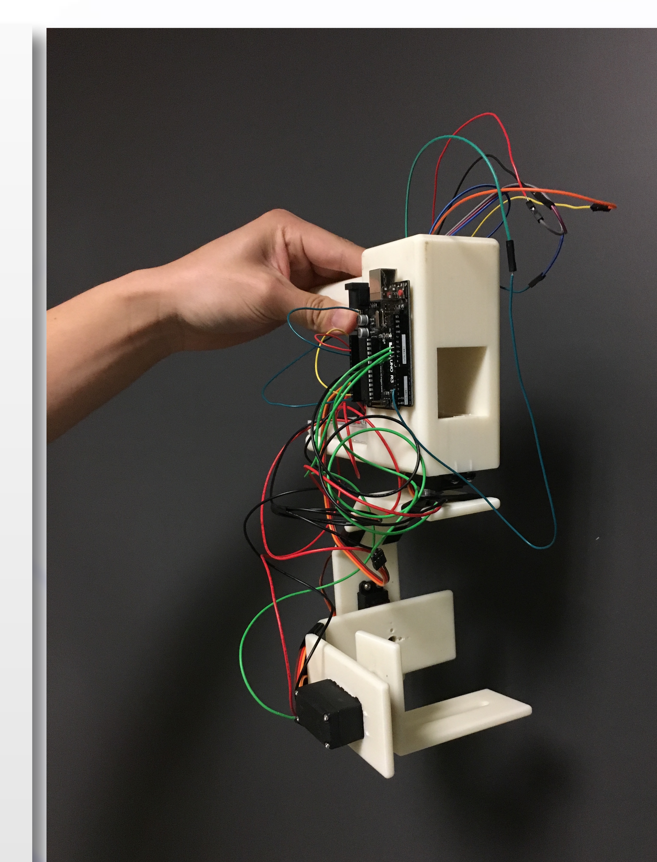




Frame Design

The gimbal arms and handle were all 3D-printed out of ABS plastic.





SOLIDWORKS Multiphysics CAD Model

Acknowledgements

Andrew Muslin for his unrivalled generosity and support with our project hardware from start to finish.
Professor Mertens and Professor Fixel for mentoring us over the course of two semesters.