2.004 Lab 5 Intro: Active Gimbal Control

Fall, 2021

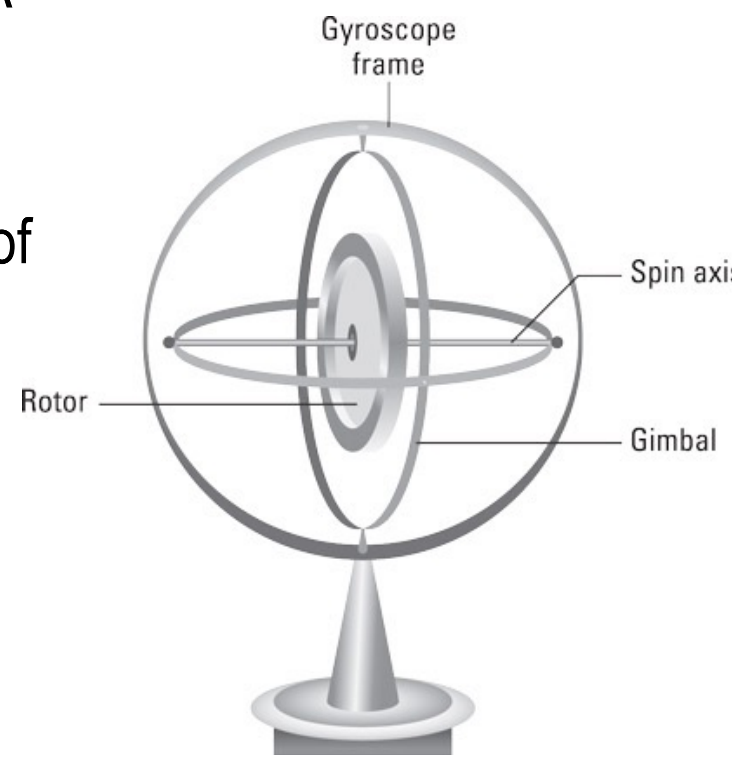
Today’s Tasks

- Objective:
  - Stabilize both pitch and roll motions of a dual-axis gimbal system using IMU sensor feedback.
- Design a PD controller to achieve better stability based on a P controller response.
- Examine the effect of using different signals for the derivative control action.
- Extra Credit Task: synchronize the motion of your gimbal system to track an oscillating mass suspended by a spring.
- Deliverable:

– Lab 5 report

Gimbals

A gimbal is a pivoted support that allows the rotation of an object about a single axis. A set of three gimbals, one mounted on the other with orthogonal pivot axes, may be used to allow an object mounted on the innermost gimbal to remain independent of the rotation of its support



Camera Gimbal Systems

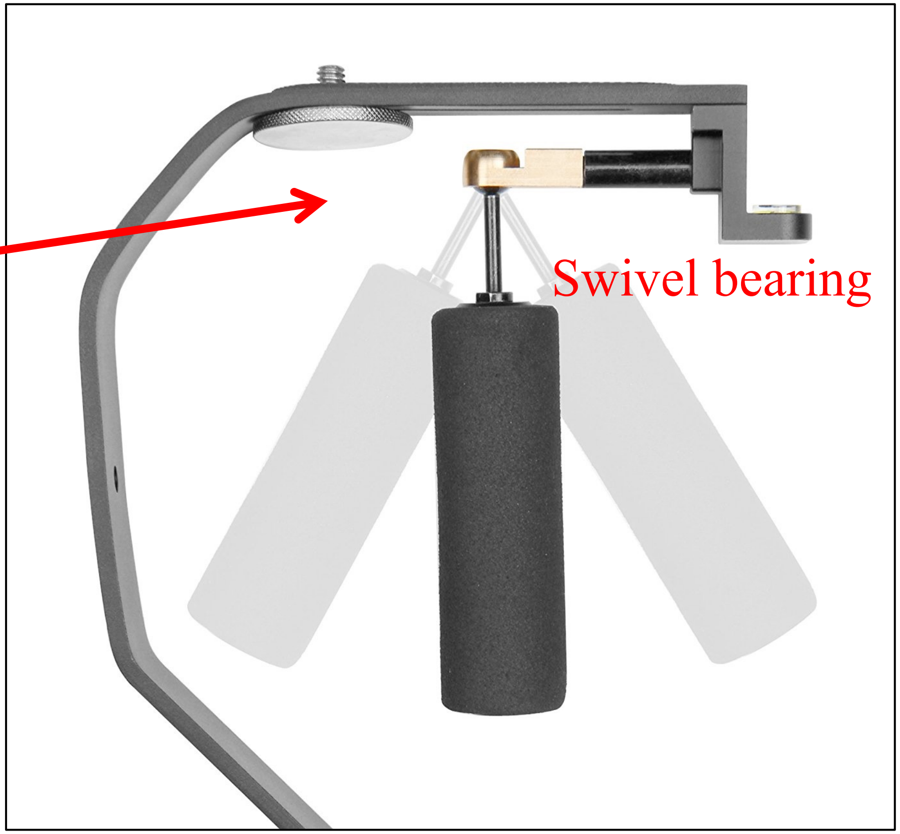
Handheld 3-axis gimbals are used in stabilization systems designed to give the camera operator the independence of handheld shooting without camera vibration or shake.

Passive Gimbals

Counter weight

Passive Gimbals

Counter weight



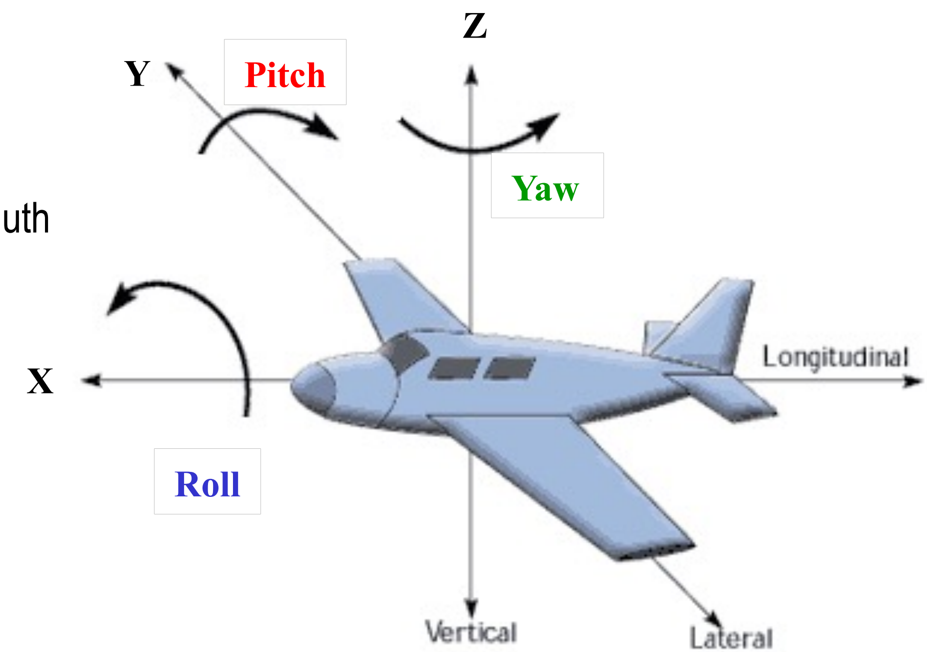
Active Gimbals

- Active gimbals have the ability to keep the camera level on all axes as the camera operator moves the camera.
- An inertial measurement unit (IMU) senses the movements of the camera and a controller stabilizes the camera by sending commands to three actuators.
- With proper feedback control and tracking algorithms, the stabilizer is able to notice the difference between deliberate movement such as pans and tracking shots from unwanted shake.
- It allows the camera to seem as if it is floating through the air.
- In addition to handheld shooting, gimbals can be mounted to cars and other vehicles such as drones, to minimize vibrations or other unexpected movements.

Attitude Control

- Active gimbal control requires controlling the attitude of an object.
- Attitude provides information about an object's orientation with respect to the local level frame (horizontal plane) and true north.

Wing up/down = Roll
Nose up/down = Pitch
Nose left/right = Heading/Yaw/Azimuth



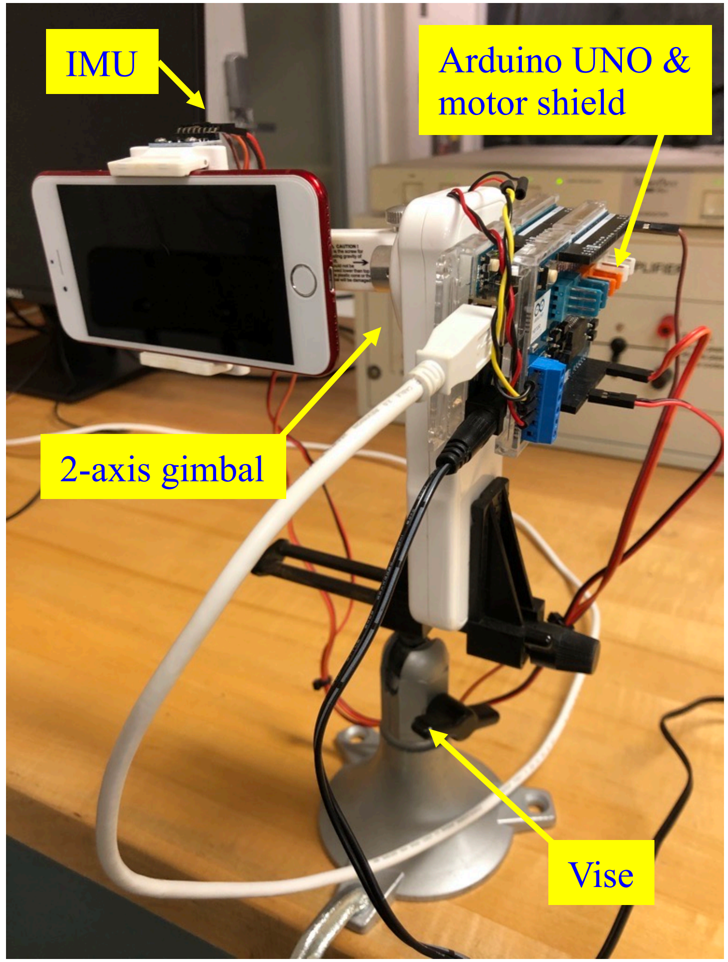
Motion Sensing Using IMU's

- Small devices used to indicate changing orientation in smart phones, video game remotes, quad-copters, etc.

- These devices contains gyroscopes combined with accelerometers and/or compasses (magnetometers) and are referred to as an *IMU*, or *Inertial Measurement Unit*

- The number of sensor inputs in an IMU are referred to as "DOF" (Degrees of Freedom), so a chip with a 3-axis gyroscope and a 3-axis accelerometer would be a 6-DOF IMU.

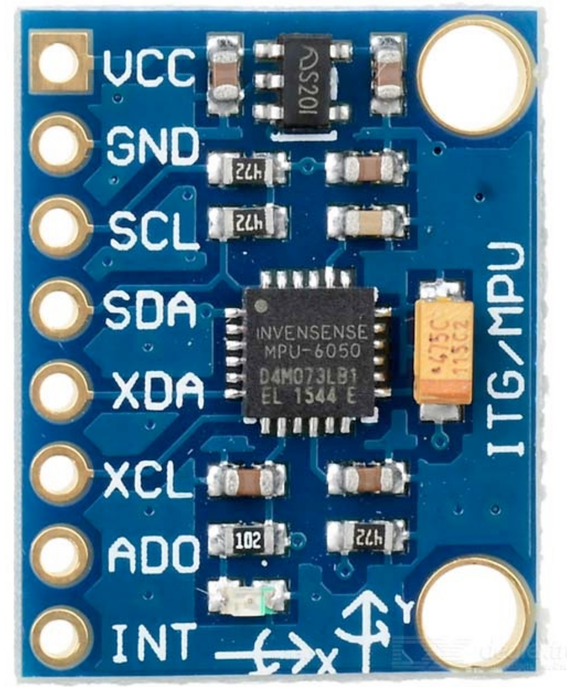
2.004 Dual-Axis Gimbal Setup



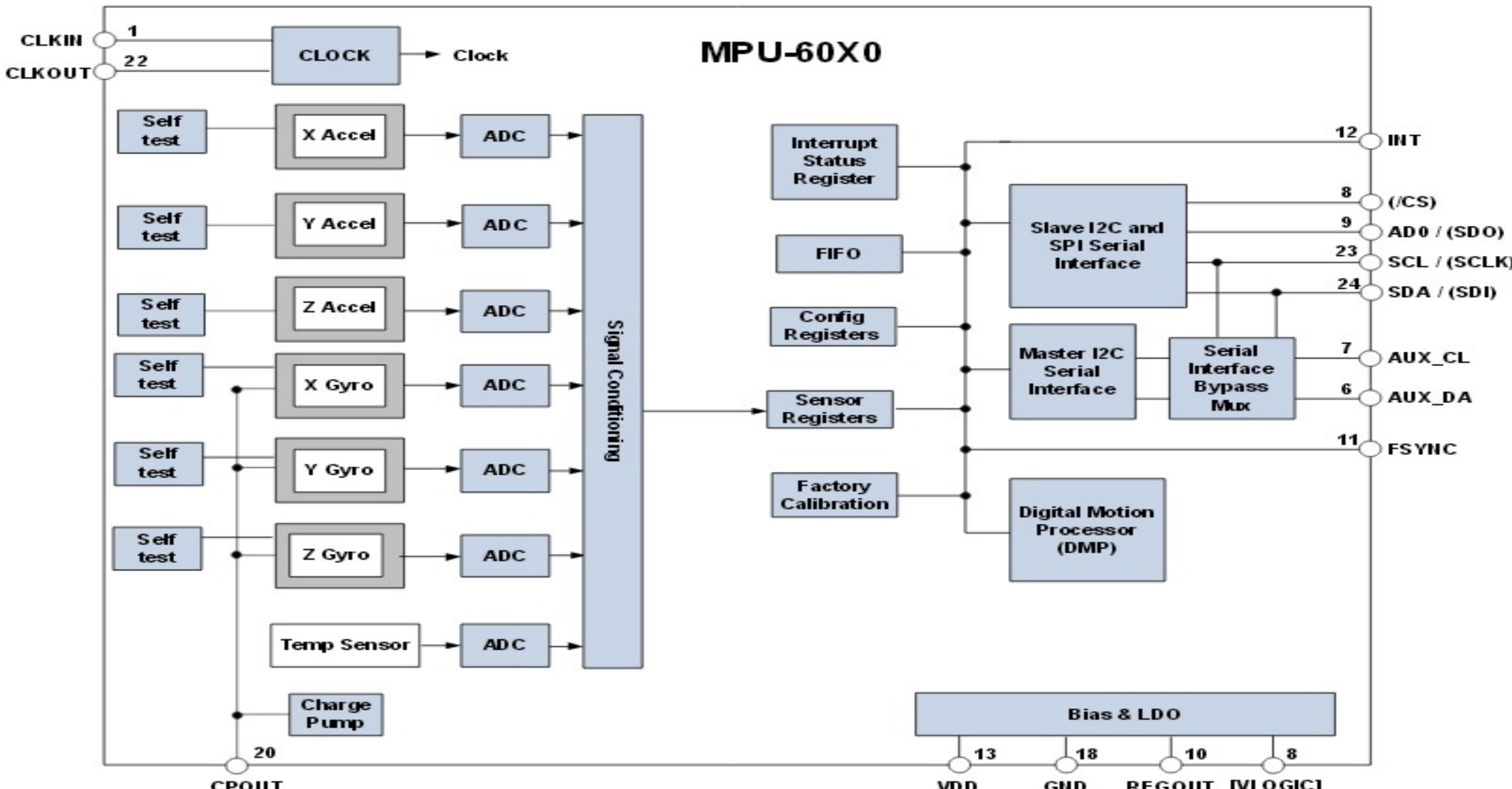
The gimbal structure is based on the X-CAM SIGHT2 2 Axis Handheld Gimbal for Smart Phone.

IMU: MPU-6050 (by TDK InvenSense)

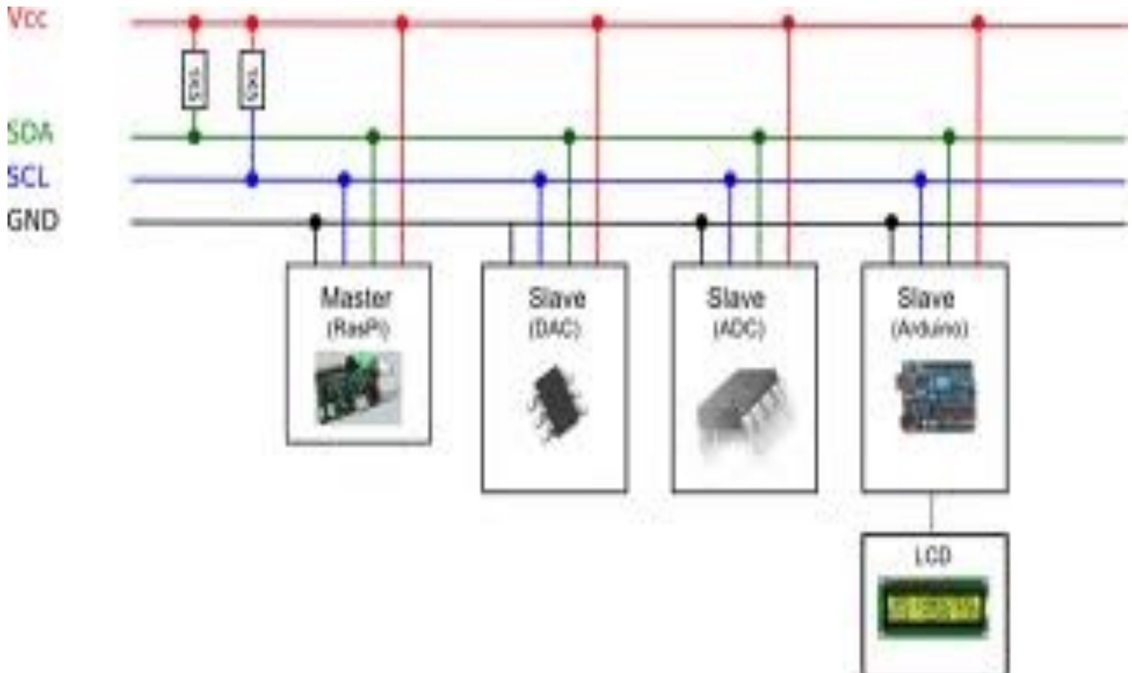
- The MPU-6050 is the world's first integrated 6-axis Motion Tracking device.
- It combines a 3-axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor™ (DMP) all in a small 4x4x0.9mm package.
- It uses a standard I2C bus for data transmission.
  - With it's I2C bus, it can accept inputs from an external 3-axis compass to provide a complete 9-axis Motion Fusion output.
- A number of different breakout boards are available containing the MPU-6050 chip, we have the GY-521 version of IMU.



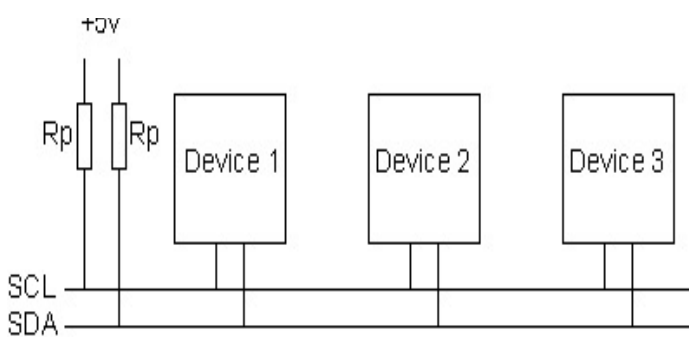
MPU-6050 Functional Block Diagram



I²C



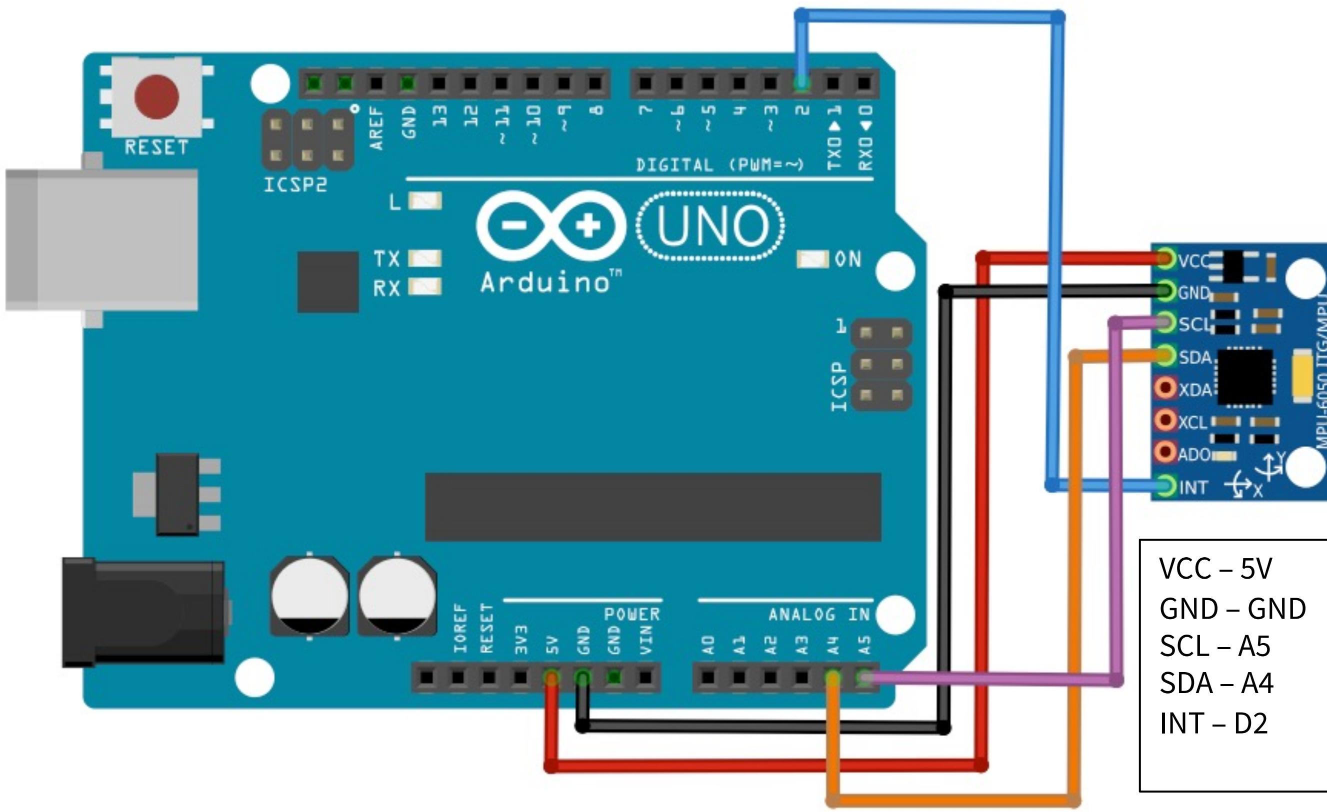
http://quick2wire.com/articles/i2c-and-spi/



- **Two wires: SCL and SDA**
  - SCL is the clock line: used to synchronize all data transfers
  - SDA is the data line

http://www.robot-electronics.co.uk/acatalog/I2C_Tutorial.html

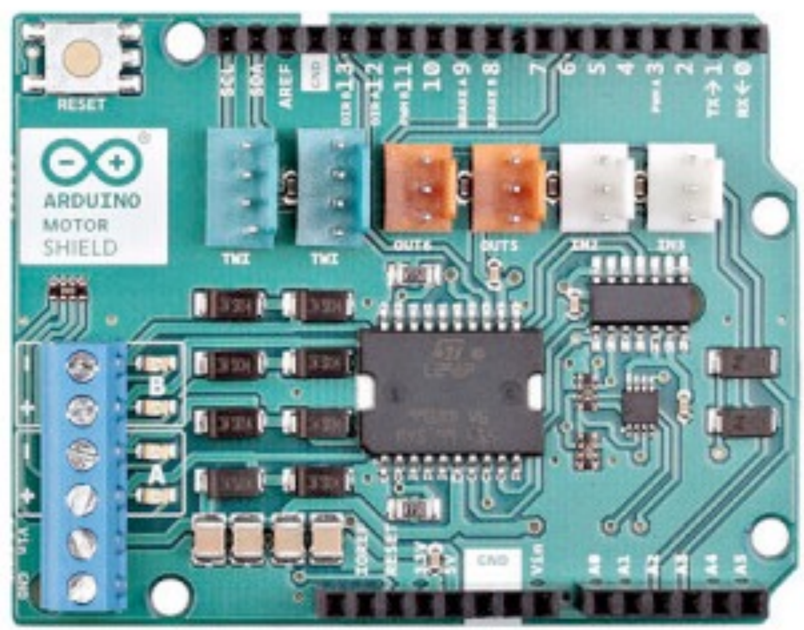
MPU-6050 Arduino Connection Diagram



VCC – 5V
GND – GND
SCL – A5
SDA – A4
INT – D2

Arduino Motor Shield

- Based on the L298P chip, a dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. It lets you drive two DC motors with your Arduino board, controlling the speed and direction of each one independently.
- The shield can supply 2A per channel, for a total of 4A maximum current. Current sensing: 1.65 V/A.



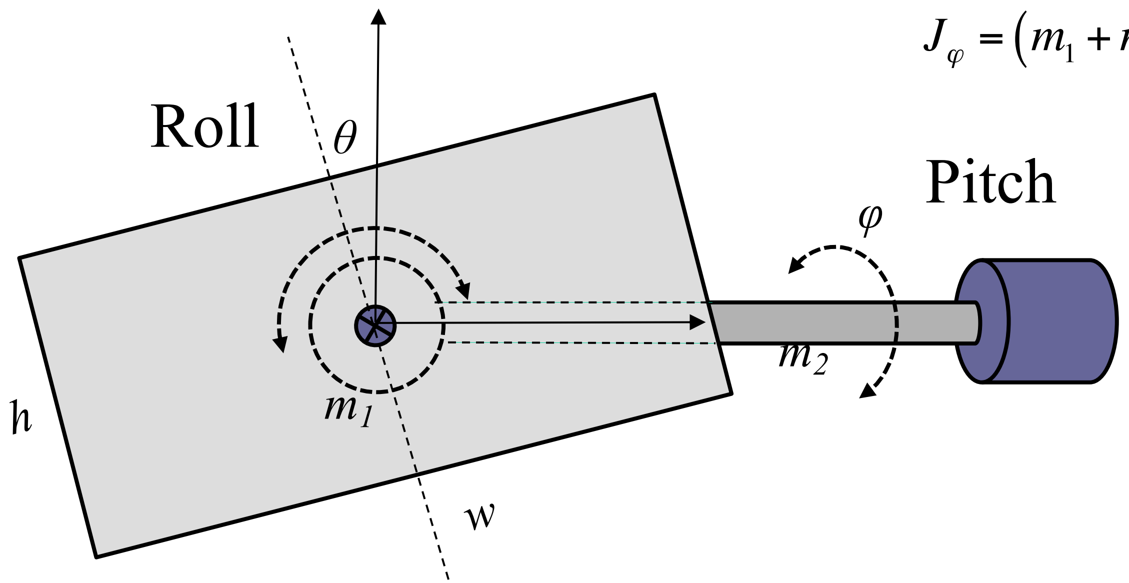
Modeling (Dual-Axis)

For roll: model the phone as a thin rectangular plate of height h, width w and mass m_1 with axis of rotation at the center.

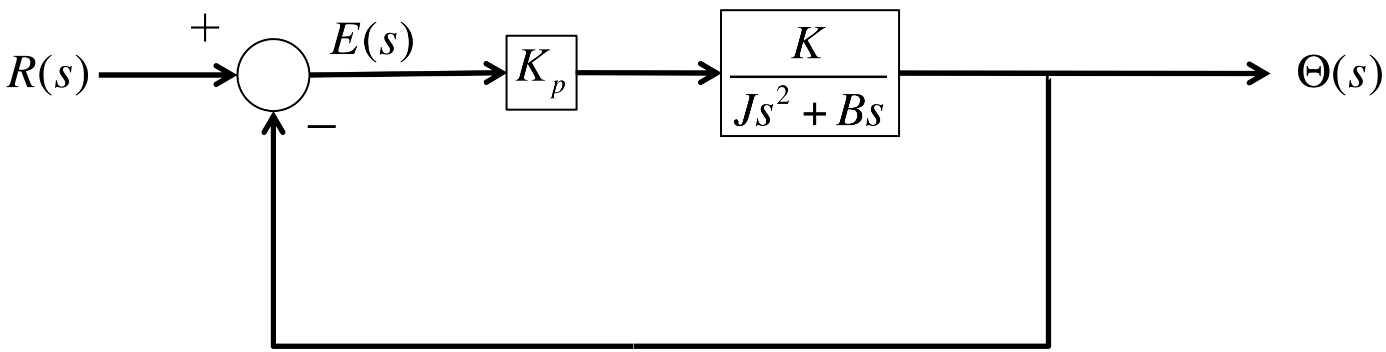
$$J_\theta = \frac{m_1}{12}\left(h^2 + w^2\right)$$

For pitch: model the phone as a point mass of radius $r = h/2$ and mass $(m_1 + m_2)$.

$$J_\varphi = \left(m_1 + m_2\right) r^2$$



Estimate the Plant Model Using P Control

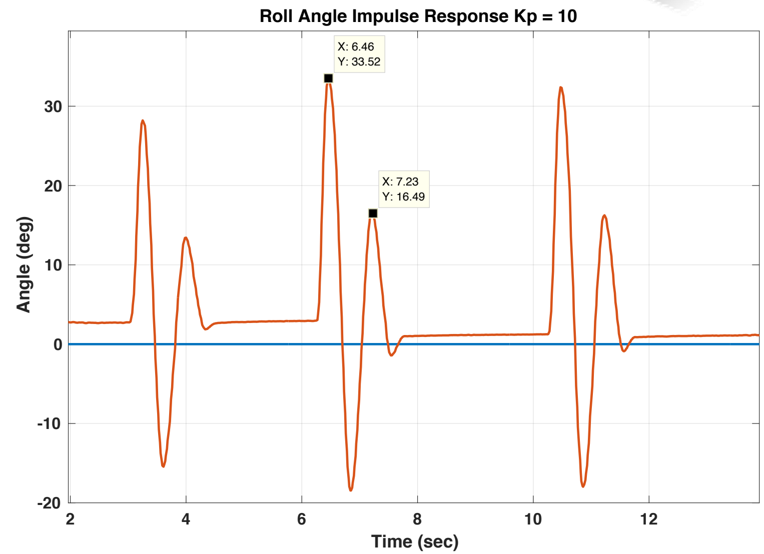


$$G_{cl}(s) = \frac{\Theta(s)}{R(s)} = \frac{\frac{K_p K}{J}}{s^2 + \frac{B}{J}s + \frac{K_p K}{J}} \longleftrightarrow s^2 + 2\zeta\omega_n s + \omega_n^2$$

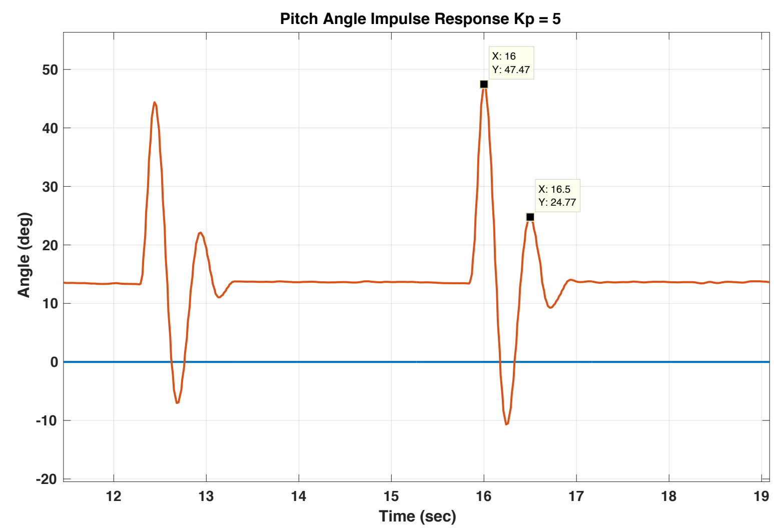
$$\begin{cases} \frac{B}{J} = 2\zeta\omega_n \\ \frac{K}{J} = \frac{\omega_n^2}{K_p} \end{cases}$$

Impulse Response with P Control

For roll: secure the pitch axis with one hand and disturb the roll axis to generate an impulse. Design a PD to stabilize the roll motor.

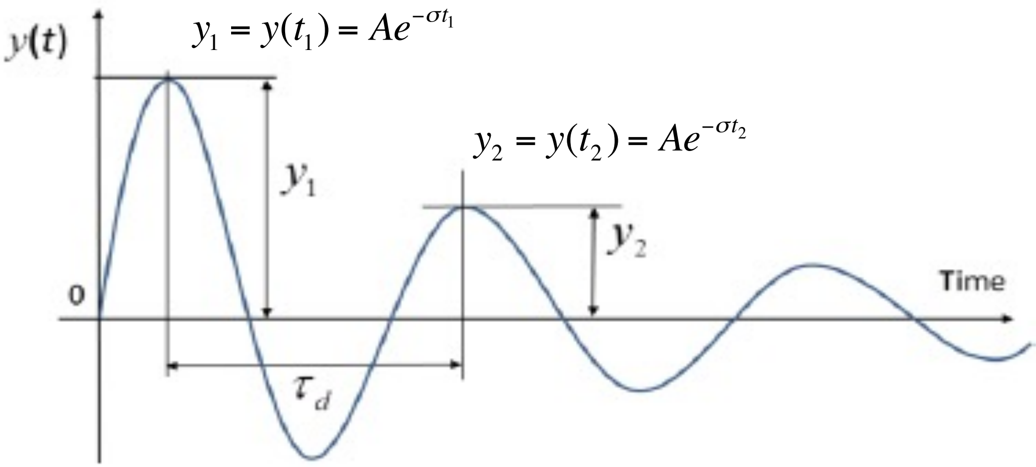


For pitch: while the roll axis is being stabilized by the PD controller, disturb the pitch axis to generate an impulse. Design a PD to stabilize the pitch motor.



2nd Order Impulse Response

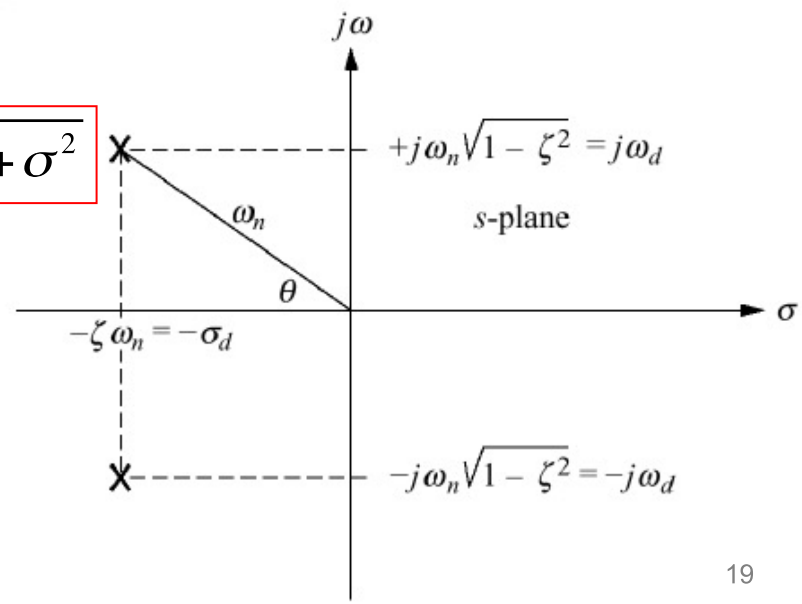
Impulse response: $y(t) = Ae^{-\sigma t}\sin(\omega_d t)$



$$\frac{y_1}{y_2} = \frac{Ae^{-\sigma t_1}}{Ae^{-\sigma t_2}} = \frac{Ae^{-\sigma t_1}}{Ae^{-\sigma (t_1+\tau_d)}} = \frac{Ae^{-\sigma t_1}}{Ae^{-\sigma t_1}e^{-\sigma\tau_d}} = e^{\sigma\tau_d}$$

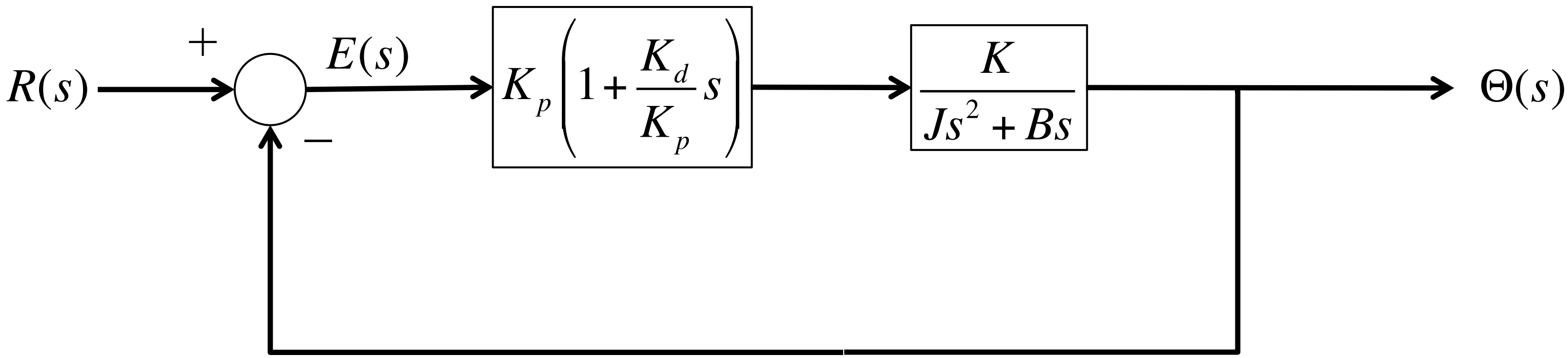
$$\omega_n = \sqrt{\omega_d^2 + \sigma^2}$$

$$\omega_d = \omega_n\sqrt{1-\zeta^2} = \frac{2\pi}{\tau_d}$$



Design a PD Control to Increase Damping

MITMECHE



A zero at $-\left(\frac{K_p}{K_d}\right)$

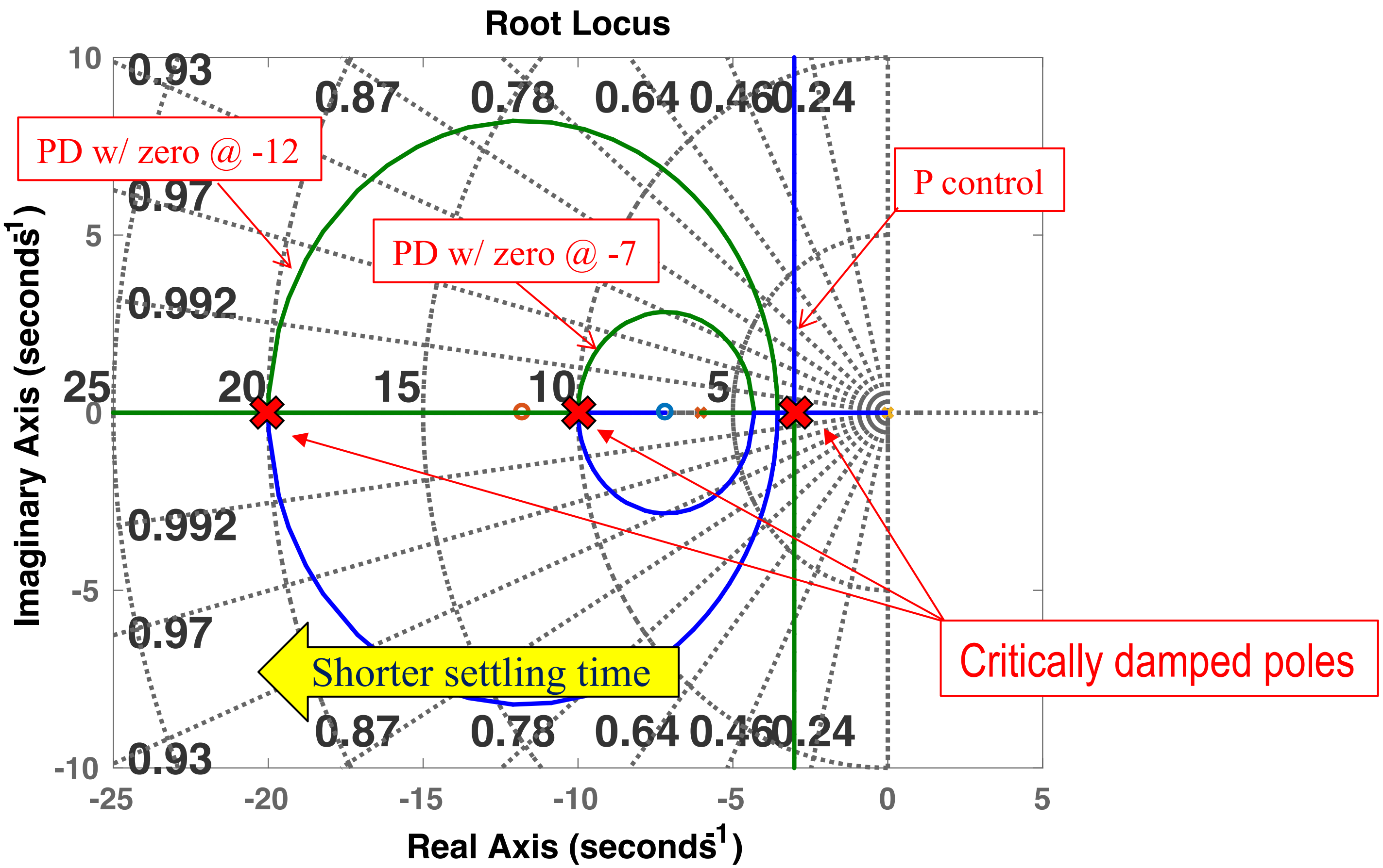
$$G_{cl}(s)=\frac{\Theta(s)}{R(s)}=\frac{\left(K_p+K_d s\right)K}{Js^2+\left(B+K_d K\right)s+K_p K}=\frac{\dfrac{\left(K_p+K_d s\right)K}{J}}{s^2+\dfrac{\left(B+K_d K\right)}{J}s+\dfrac{K_p K}{J}}$$

Add K_d to increase the damping term

$$s^2+2\zeta\omega_n s+\omega_n^2$$

Effect of PD Controller Zero Location

PD zero location: $z = \dfrac{-K_p}{K_d}$



From P Control to PD Control

P control response: $G_{cl}(s)=\dfrac{\Theta(s)}{R(s)}=\dfrac{\dfrac{K_pK}{J}}{s^2+\dfrac{B}{J}s+\dfrac{K_pK}{J}} \longleftrightarrow s^2+2\zeta\omega_n s+\omega_n^2$

Given K_p^o with three unknowns: J, B, K $\Rightarrow \begin{cases} \dfrac{B}{J}=2\zeta\omega_n \\ \dfrac{K}{J}=\dfrac{\omega_n^2}{K_p^o} \end{cases}$

PD control response:

If we pick desired damping ratio ζ' and natural frequency ω_n' for PD response:

$$\Rightarrow \begin{cases} \dfrac{B+K_dK}{J}=\dfrac{B}{J}+\dfrac{K_dK}{J}=2\zeta'\omega_n' \\ \dfrac{K_pK}{J}=\omega_n'^2 \end{cases}$$

Design Goals

- A critically damped dynamic behavior (no overshoot).
- Fast reaction time. Think about how fast the controller has to be to counteract a rapid hand/arm movement when holding the gimbal.
- For example, we can design the gimbal to control motions up to 3 cycles of movements per second (3 Hz or 18.85 rad/s).
- Note that to achieve this frequency or higher you need to use *pitch_rate* and *roll_rate* to compute *Dcontrol* term for their respective PD controller.
- Capture and analyze step responses with PD control to check your design.

FYI: Ziegler–Nichols PID Tuning Method

- An empirical PID controller tuning method.
- Use proportional only control and find $K_p = K_c$ that produces a marginally stable response.
- K_c and the oscillation period T_c are then used to set the P, I, and D gains according to the following heuristic rules:

Controller	Kp	Ki	Kd
P	0.5Kc	0	0
PI	0.45Kc	Kp(1.2/Tc)	0
PD	0.8Kc	0	Kp(Tc/8)
PID	0.6Kc	Kp(2/Tc)	Kp(Tc/8)
Some overshoot	0.33Kc	Kp(2/Tc)	Kp(Tc/3)
No overshoot	0.2Kc	Kp(2/Tc)	Kp(Tc/3)

Fine Tuning of Controller Gains

- **In general:**
  - Increase K_p to improve speed of response (decrease rise time).
  - Increase K_d to reduce overshoot and settling time.
  - Increase K_i to eliminate steady-state error.

A Natural Gimbal???… Galluscam

https://www.youtube.com/watch?v=LEGZ7hGaMNI