EOG Dino Game

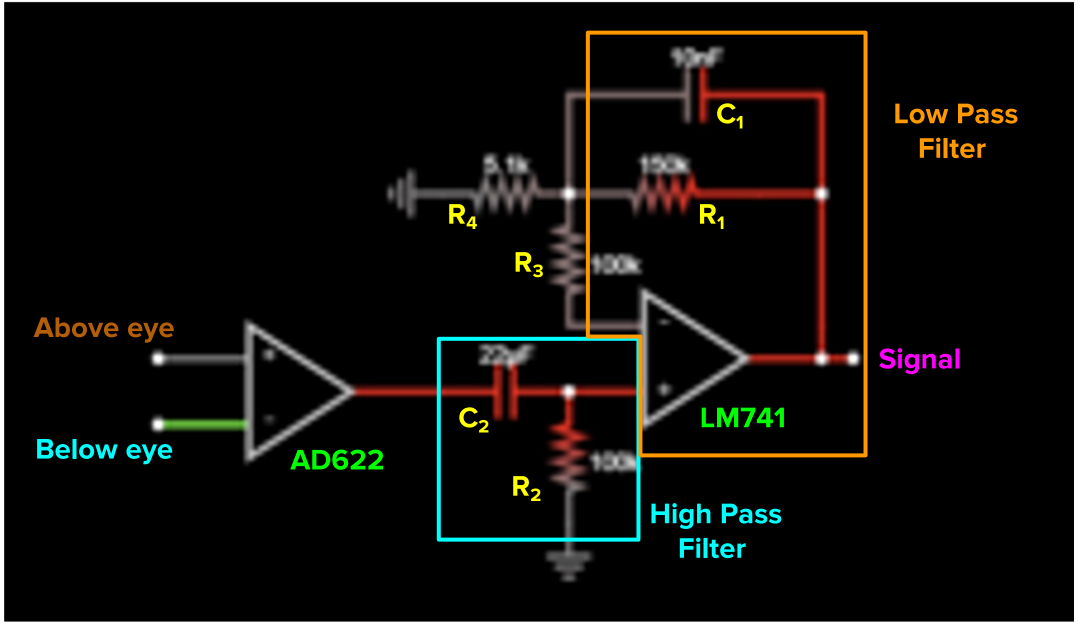
BENG 186B WI23
William Chen, Wesley Hsu, Katelyn Kennedy, Edward McGee, Jerry Qiao

Our Idea

- Playing the ‘No Internet’ Dinosaur game by looking up or down
- EOG Sensors ➔ ELECTROOCULOGRAPHY CIRCUIT ➔ ARDUINO ➔ DINO GAME

Circuit

9V power supply to both op amps



Assumptions: ideal components, variation in electrode placement is not significant

Circuit Specifications

	Theoretical	Measured
R_g	100kΩ	98.4kΩ
R_1	150kΩ	166.4kΩ
R_2	100kΩ	99.1kΩ
R_3	100kΩ	99.8kΩ
R_4	5.1kΩ	5.05kΩ
C_1	10nF	9.701nF
C_2	22μF	20.6μF

Circuit Components and Reasoning

- High Pass Filter: R_2 and C_2
- Low Pass Filter: LM741 non-inverting amplifier with R_1 and C_1
- The low pass filter filters out high frequency noise
- The high pass filter blocks out the low frequency noise, and filters out any DC drift
- The two filters form a band-pass filter that filters out the common-mode noise.
- R_3 matches R_2 in resistance, does not affect the gain

Circuit Theoretical and Measured Calculations

Theoretical Gain = $\left(1+\frac{R_1}{R_4}\right)\left(\frac{50.5k}{R_g}+1\right)=\left(1+\frac{150k\Omega}{5.1k\Omega}\right)\left(\frac{50.5k}{100k\Omega}+1\right)=45.77$

Measured Gain = $\left(1+\frac{R_1}{R_4}\right)\left(\frac{50.5k}{R_g}+1\right)=\left(1+\frac{166.4k\Omega}{5.05k\Omega}\right)\left(\frac{50.5k}{98.4k\Omega}+1\right)=51.37$

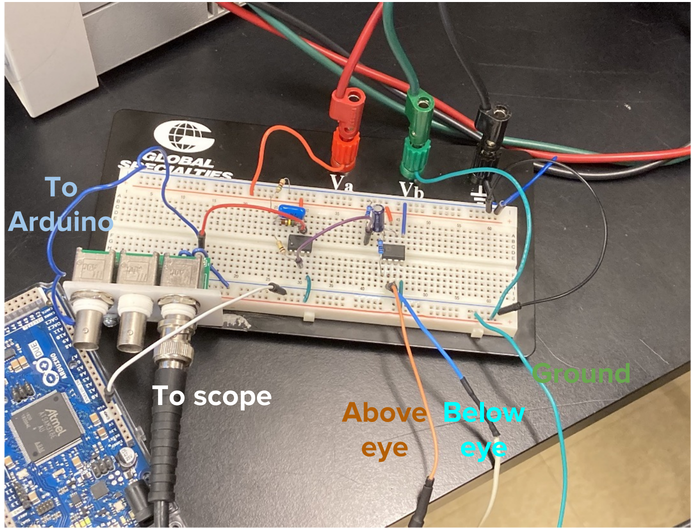
Theoretical High-Pass Cutoff: $f_c=\frac{1}{2\pi * R_2 * C_2}=\frac{1}{2\pi * 100k\Omega * 22\mu F}=0.072\ Hz$

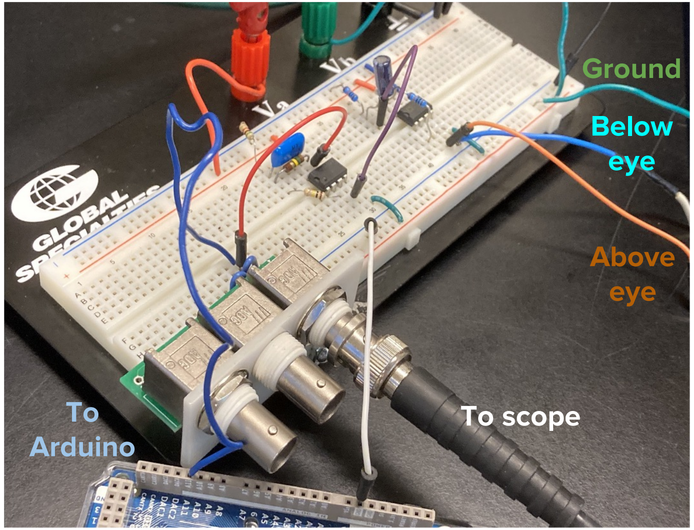
Measured High-Pass Cutoff: $f_c=\frac{1}{2\pi * R_2 * C_2}=\frac{1}{2\pi * 99.1k\Omega * 20.6\mu F}=0.078\ Hz$

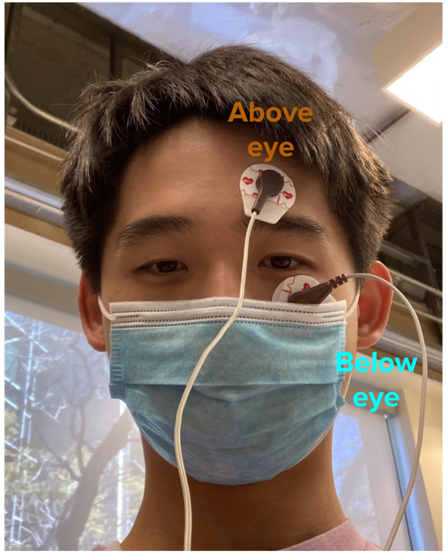
Theoretical Low-Pass Cutoff: $f_c=\frac{1}{2\pi * R_1 * C_1}=\frac{1}{2\pi * 150k\Omega * 10nF}=106.103\ Hz$

Measured Low-Pass Cutoff: $f_c=\frac{1}{2\pi * R_1 * C_1}=\frac{1}{2\pi * 166.4k\Omega * 9.701nF}=98.59\ Hz$

Building a Prototype



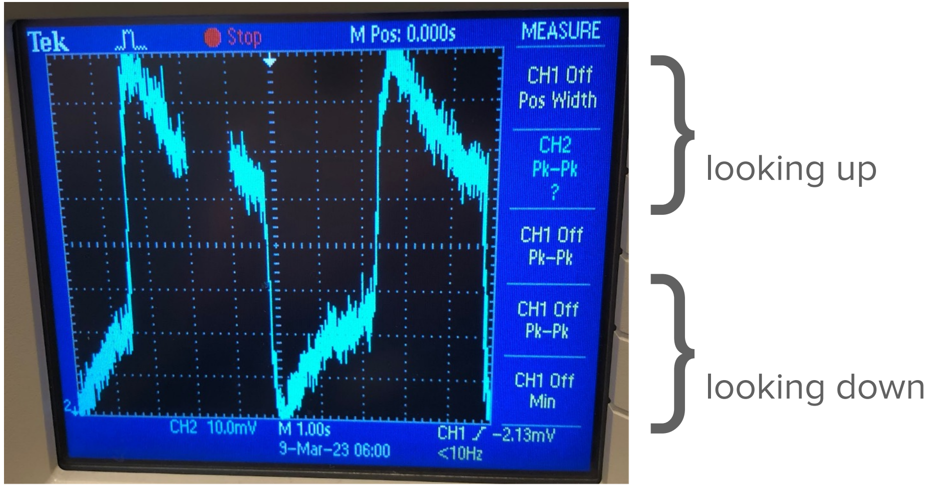




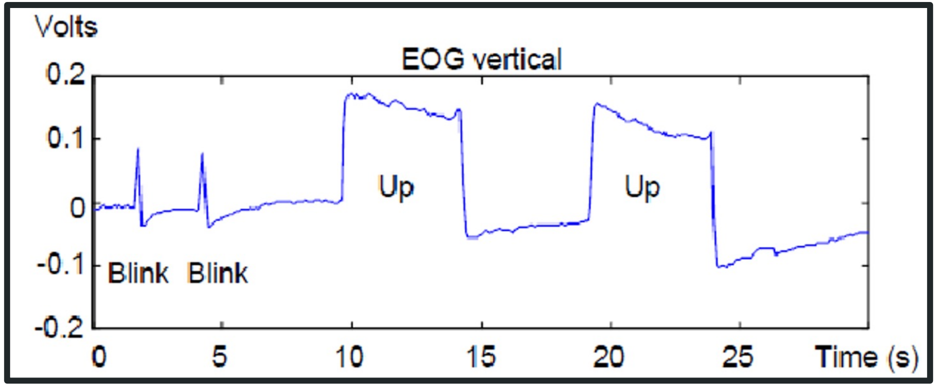
*Ground is placed on hip

- Materials: Arduino Due, eye-tracking probes, computer w/ Google Chrome installed, USB cable, cable to scope, cables to power supply, Breadboard, Op Amps, Jumper wires, Resistors, Capacitors

Pictures of EOG Signals



Electrooculography Circuit Output Signal



Example of Blink vs. Looking Up Signal Amplitudes

1. Define analog input pin

1. Define “up” and “down” threshold for triggering arrow keys

1. Initialize Keyboard library, identify port, read analog input and map it to range of values

1. If mapped value > or < threshold, trigger UP or DOWN arrow key

sketch_mar11a §

```
// Define the analog input pin that reads the eye-tracking signal
const int analogInPin = A11;

// Define the threshold for triggering the "up arrow" key (may need to be adjusted)
const int thresholdUp =8;

// Define the threshold for triggering the "down arrow" key (may need to be adjusted)
const int thresholdDown = 0;

void setup() {
  // Set up serial communication for debugging
  Serial.begin(9600);

  // Set up pin modes
  pinMode(analogInPin, INPUT);



  // Initialize the Keyboard library
Keyboard.begin(&Serial1); // because the Mega 2560 has four hardware serial ports and we need to specify which one to use. In this example, we're using Serial1
}

void loop() {
  // Read the analog input and map it to a range of values
  int analogValue = analogRead(analogInPin);
  int mappedValue = map(analogValue, 0, 1023, 0, 1000);

  // Debugging output
  Serial.print("Analog value: ");
  Serial.print(analogValue);
  Serial.print(", mapped value: ");
  Serial.println(mappedValue);

  // Trigger the appropriate key based on the mapped value
  if (mappedValue > thresholdUp) {
    Keyboard.write(KEY_UP_ARROW);
    delay(50);
  } else if (mappedValue < thresholdDown) {
    Keyboard.write(KEY_DOWN_ARROW);
    delay(50); // may also need to adjust the delay() value depending on how often you want the keyboard keys to be triggered
  }
}
```

Results

The design succeeded in allowing us to easily distinguish the voltage values between looking up and down with a very clear boundary to assign in the software as thresholds for the keyboard inputs.

Limitation: might be challenging to look at the screen while performing inputs as that would require eye movement (especially looking up, since the monitor is rarely higher than eye level).

Advantage: threshold for looking up can be lowered such that blinking can be registered as looking up, which makes inputs more feasible based on an ergonomically normal monitor height.

Industry Significance and Applications

- Locked-In Syndrome
  - Paralyzes victim’s body except eyes
  - Rare and only a few confirmed cases each year in the United States
- Lou Gehrig’s Disease
  - Weakens muscles and nerves for physical function
  - Eye tracking can be a quick and easy work-around for simple technology-related tasks.
- Simplistic control with the eyes ➔ Future complex control systems
  - Horizontal eye movement could additionally be considered as another axis for combination inputs such as left/down or right/up

References & Resources

- Simulation: https://www.falstad.com/circuit/
- RMD;, M Das J;Anosike K;Asuncion. “Locked-in Syndrome.” National Center for Biotechnology Information, U.S. National Library of Medicine, https://pubmed.ncbi.nlm.nih.gov/32644452/.https://www.researchgate.net/figure/B-Eye-movement-and-blinking-detection-vertical-EOG_fig9_261020471.
- AD622 Datasheet: https://www.analog.com/media/en/technical-documentation/data-sheets/AD622.pdf
- LM741 Datasheet: https://www.ti.com/lit/ds/symlink/lm741.pdf

Acknowledgements

- Thank you to Dr. Gert Cauwenberghs and the BENG 186B TA's for lectures & guidance throughout the quarter.
- Thank you to Dr. Bruce Wheeler for providing guidance and circuitry components for our prototype.