# Project Proposal

# ECE 445

# Project 23 - Full Movement Gaming Mouse

# Drake Bernhard and Michael Bindon

1 Introduction

1.1 Objective

The traditional mouse has the ability to point on a 2-D surface, has a few buttons, and can scroll with a center scroller. This commonly must be supplemented with other input for higher end computer uses. A mouse with more functionality could be very useful to multiple groups of people. CAD drafting and similar programs have 3D space to navigate through, and computer games commonly have 5 axis of direction (2 for looking and 3 for movement). Our goal is to build a mouse that can allow for 4 axis of movement comfortably and compactly while not altering the basic design of the mouse to a point where it interferes with everyday uses. This project will attempt to solve this through building a mouse which contains a joystick device on the thumb rest area.

1.2 Background

Presently on the market there are a few computer gaming specific mouses that use a joystick on the thumb. The largest problem with these existing mouses is the awkwardness of the thumb movements whether through use of a true joystick or through a circle of click buttons [6]. Many of the joystick based mouses take time to get a feel for what ‘true forward’ is. Our mouse aims to provide a more natural feeling thumb joystick while keeping the remainder of the mouse normal.

1.3 Requirements

* Joystick feels natural on the hand during use and provides smooth function during gameplay.
* Mouse use must be familiar or intuitive as well as easy to move in daily use. The mouse should also be comfortable enough for continuous uses of hours on end for long gaming runs as well as CAD projects.
* The mouse will have to be sufficiently responsive for computer gamers. We will strive for less than 15 ms of latency with our mouse, which is delay of a single 60Hz frame and is considered good, even for gaming.

2 Design

2.1 Physical Design

The physical design of this computer mouse will be made from a 3D printed plastic with dimensions of 100mm x 60mm x 50mm. The main component of this project, the joystick, will be able to vertical ±15mm from its resting position. Additionally, the joystick will be able to rotate ±10° from the horizontal position.

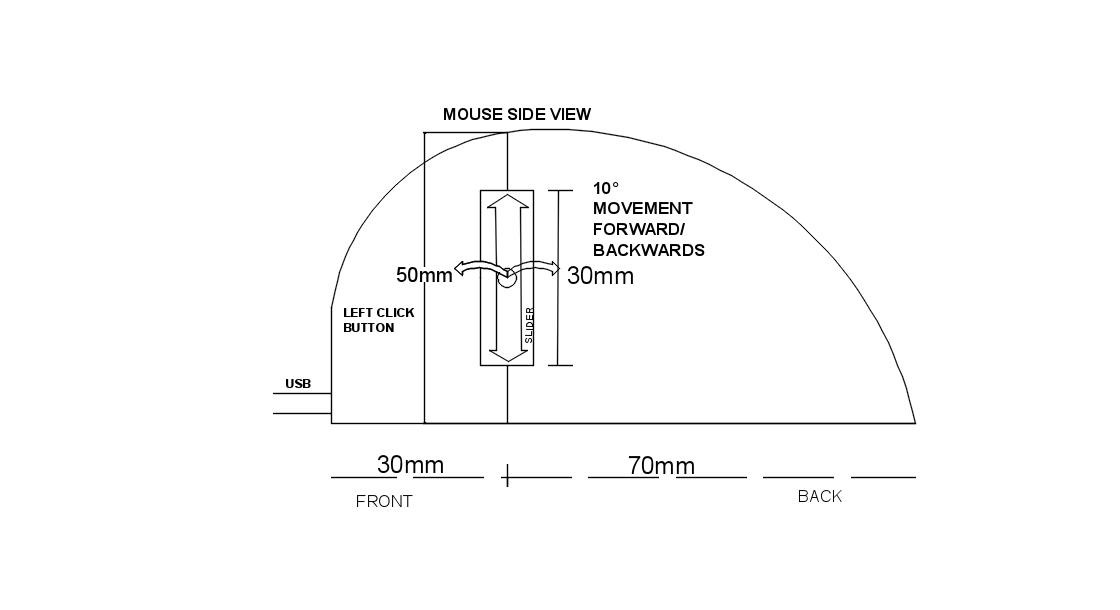
****

Figure #1. Mouse Side View

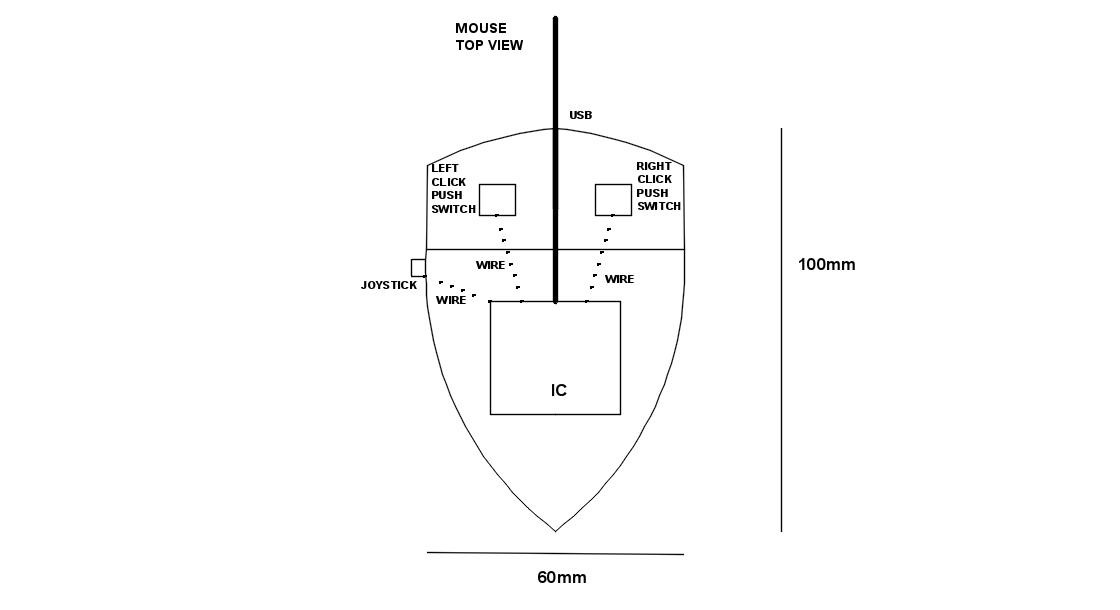
****

Figure #2. Mouse Top View

The preceding figures give our current projections of key size requirements for providing maximum usability and comfort. An ergonomic fit is an important aspect of the project and tends to be difficult normally, it will be more difficult with the added thumb use. Current dimensions are from measurements made on hand movements as well as typical mouse sizes. The exact final dimensions will be determined through feedback from a variety of trials on volunteers. A finished product will need multiple sizes, at least a small, medium, and large size for different hand lengths. Ideally the mouse would allow for custom fitting, most likely through adjustment screws.

2.3 Block Diagram

For a successful full movement gaming mouse, the product will have a control unit, electrical sensors, mechanical input in the form of a joystick, and power source coming from the user’s computer. The power source, from the USB port, will be able to power all the electrical sensors and the MCU with 5V and 50mA. The electrical sensors will measure the movement of the mouse, both left and right clicks, and finally an analog sensor to measure the mechanical movement of the joystick. Finally, the control unit will be in the form of a microcontroller and it’s responsible for gathering data from the sensors and passing it along to the host computer’s hub.

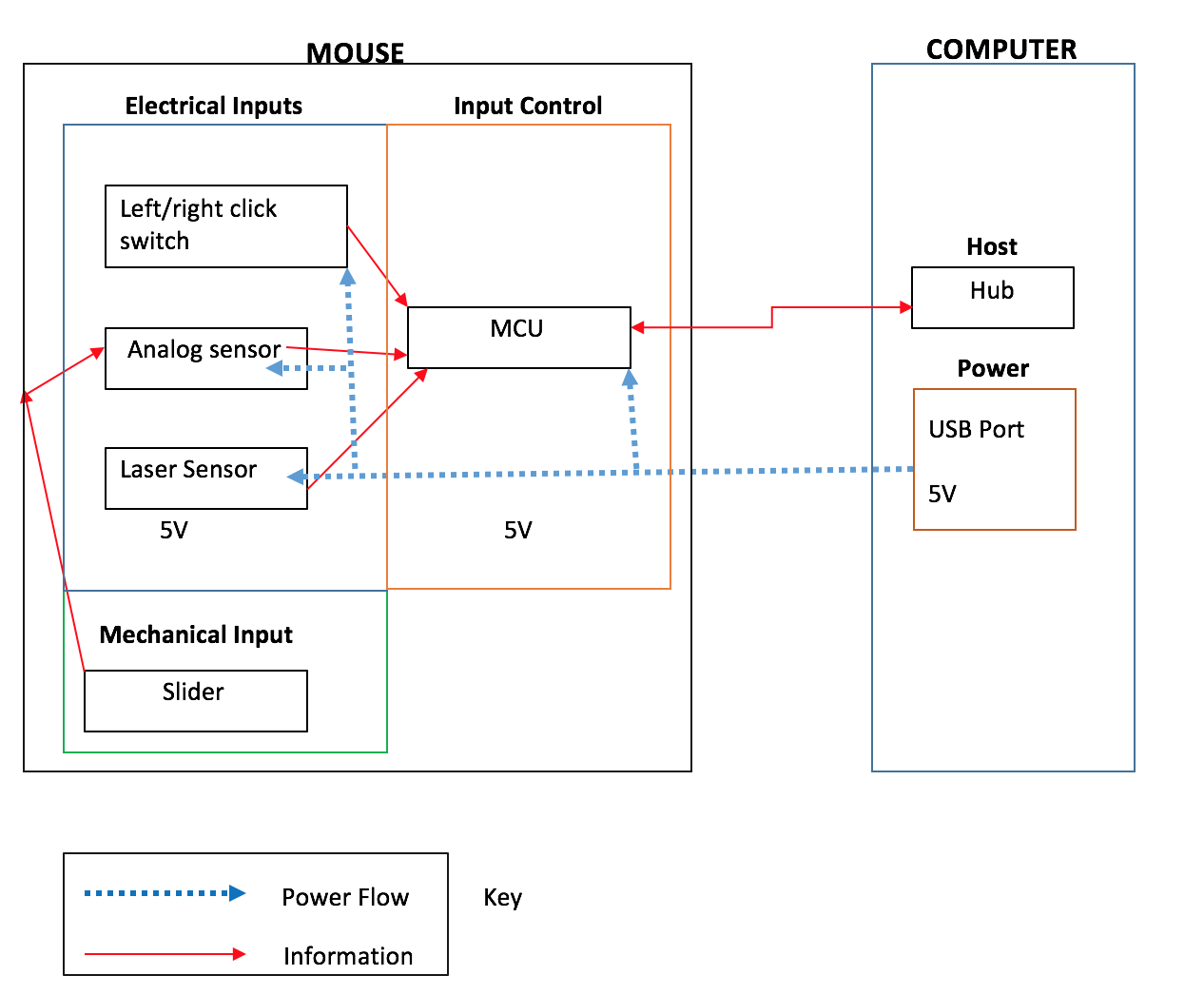


Figure #3. Block Diagram

2.4 Functional Overview and Block Requirements

2.4.1 USB Port

The mouse requires a means to communicate data with the external computer as well as a supply of power. The USB connection provides both of those and is the standard choice for wired computer peripherals. Our design will require an estimated data rate of a few (less than 20) kB/sec, and 250mW power draw (50 mA at 5V).

USB 2.0 can provide 5V through a single cord and a maximum of 500mA, for a total of 2.5W. USB high speed is 480MBits/sec and full speed in 12MBits/sec [3]. The USB selection will have more than enough ability to beat both requirements. Given that USB is also the universal choice for wired mouses and almost all computer peripherals we will use USB protocol to enable better customer acceptance.

|  |  |
| --- | --- |
| Requirement | Verifications |
| 1. Zener diode within mouse’s circuit will protect 5.25V over voltage | 1. A. Multimeter probes will be placed across the Zener diode   B. Test voltages from 4.75-5.25V  C. Multimeter indicates minimum voltage drop above 5.25V |
| 1. A 2 µF will be placed from source to ground to protect transient voltage drop no more than 330mV | 1. A. Oscilloscope will be placed across source and ground to measure voltage   B. Voltage will be measured for 1 second after plugging the device into the active power supply  C. Oscilloscope should indicate voltage no more than 330mV |

Figure #4. USB Requirements

2.4.2 Joystick / Joystick Sensor

The joystick is the mechanical device that translates user actions into signals for communication to the external computer. The goal of the joystick is for the user to be able to use it comfortably with his thumb. The joystick should be able to move forward, backwards, left, right, and all possible combinations of directions. In addition, when the user is not moving the joystick, the joystick should remain in the original position and not drift in one direction.

The feel of the joystick is one of the largest parts of being able to define this project as a success. Given the limited movement of the resting thumb, the joystick will have to be able to respond to very small movements (millimeters). The user will also need to have a good feel on the joystick. Specifically, the user will need to be able to feel the difference between true forward and an action that would move a few degrees off from straight forward. The joystick will need to work from power supplied to the board.

The analog sensor will be able to read one axis of the joystick’s position and convert this to analog voltage that the microcontroller can read. In order to complete this task, the analog sensor will be able to tell the direction and magnitude at which the joystick is pushed. Importantly, this sensor will need to be compact to fit either in the mouse or on the joystick. The off the shelf mouse allows at most 20 mm of thumb travel vertically. As a higher limit, your thumb only can comfortably move at most 40 mm vertically. We aim to allow a large travel length which will allow for precision while not requiring stretching and arcing movements. Our acceptable travel range at a minimum will require 20 mm of travel, 10 mm each direction. This should allow for a user to have a good feel for a continuous range of motion with larger ranges offering more precise movement. We will also have a maximum of 30 mm for usable travel due to mouse physical space as well as thumb reach and to avoid arcs in thumb travel that would interfere with forward and backward movement.

For sensor accuracy we experimentally determined it would be difficult to achieve better resolution of thumb feel than 1 mm and therefore set our accuracy requirement as needing to be able to distinguish 0.5 mm of movement reliably. Any worse than 1 mm would be a failure and any higher accuracy than 0.5 mm is not likely to be valuable to the end user.

At this time we have found a variety of variable resistor sliders that should meet or exceed our design criteria. We will select the final slider based on feel and comfort. We will also match an appropriate resistor in series to make a voltage divider that allows for a full range of values that are distinguishable using the MCU A/D converter.

The forward /backward movement of the joystick will be controlled by a small rocker switch. The thumb on the side of the mouse has much less usable forward and backward movement without requiring a change in grip. The full range of motion is estimated to be a maximum of 15 mm. Due to this limited range we will be using a 3 position rocker switch to control forward/backward/rest. This switch should revert to a center position when not having pressure applied. We have found a few parts that meet these requirements and will select the final part based on feel.

|  |  |
| --- | --- |
| Requirements | Verifications |
| 1. The slider will be able to move 30mm for comfort | 1. A. Measure 5 other people’s thumb mobility to confirm comfortability of 20mm range |
| 1. Rocker switch is able to distinguish rest/forward/backwards | 1. Simple circuit to prove that the rocker positions are unique. |
| 1. The slider will be able to distinguish movement of 1mm | 1. Analog voltage measurements of finished voltage divider by MCU after small measured movements of the slider. |

Figure #5. Joystick Requirements

2.4.3 Click Left and Click Right Buttons

The click left and click right buttons are designed to register when the user presses down on the mouse. This will be implemented by two electrical pushbuttons. Both switches will operate independently and be assigned to specific input pins of our microcontroller. The switches need to be able to handle fast clicking for the common double clicking and also be able to hold up to tens of thousands of clicks over the life of the mouse. Clicking as fast as 10 clicks per second is not uncommon and likely to be more regulated by the plastic housing than the electrical push buttons themselves. Nonetheless, any push button required will have to be able to be pressed rapidly.

We found estimates that an average user will click 2 million times in their life. Our mouse should be able to hold up to at least 1 million clicks. For reference, the highest click life gaming mice are designed to handle 5 million clicks [4].

|  |  |
| --- | --- |
| Requirements | Verifications |
| 1. Click buttons register when they are pressed | 1. A. Digital multimeter will be placed across the right/left click switches connected to 5V source   B. When switches are closed multimeter will read 0V and when open the multimeter will indicate 5V drop. |
| 1. Click buttons will need to be able to register at least 10 distinct clicks per second. | 2. Use of counting circuit coupled with servo instructions for precise button clicks and data confirmation. |

Figure #6. Click Button Requirements

2.4.4 Optical Position Sensor and LED

The purpose of the optical sensor is to read the user’s movement of the mouse and to transfer the information to the microcontroller. Since this mouse is designed to be higher end, this laser sensor should be sensitive to small movements of the mouse. Mouses have ratings of DPI (dots per inch) which correlates how many on screen pixels will correspond to 1 inch of mouse movement. Given high resolution gaming has 2560x1440 resolution the mouse will be responsible for going 2500 pixels all while staying within the confines of a typical mouse pad. Normal mouse movement stays within 2 inches of center so 4” of total travel means that we will need accurate resolution of 2500/4 inch movements. This equates to a required 650 DPI. The upper limit on computer mouse DPI is roughly 4000 which would allow for a one inch movement to traverse an entire 4K monitor. Many computer mice have adjustable DPI which allows for a feel that works for the user without performance degrading software such as mouse acceleration.

We searched for optical or laser sensors that would work while trying to best balance size, performance, usability, and cost. We selected the CJMCU-110 optical sensor to meet each of these different areas best. This product requires external light so we will add in an external LED which will also be mounted on the bottom of the mouse.

|  |  |
| --- | --- |
| Requirements | Verifications |
| 1. Less than 15ms of latency with our mouse | 1. Precise measurement is possible on finished product through having mouse view a changing computer screen and logging mouse response. |
| 1. Optical position sensor will have accuracy of 650 DPI | 2. Test with Arduino to ensure that proper total counts received to corresponding to 1”, 2”, and 3” movements over multiple trials. |

Figure #7. Optical Sensor Requirements

2.4.5 Microcontroller (MCU)

The microcontroller is responsible for handling the incoming information from all the sensors. The microcontroller will also translate the analog values from the joystick into movement commands based on thresholds programmed based on feel at a later date. The chip will also handle communication to the laser sensor and computer through serial communication standards. After handling the information from the inputs, the microcontroller will send the proper information to the computer. The chip should have built in USB capabilities such as interrupt messaging capability and a good foundation of software for using the USB protocols.

Another communication requirement on the microprocessor is that it should have some built in support for serial communication through SPI to support simple communication with the optical sensor. SPI and I2C communication protocols dominate optical sensor communication options so ideally having both as supported features in the microprocessor will enable the best flexibility. The microprocessor will need to be able to accept input from a few digital pins (6 should be expected) as well as an analog input option for reading from the joystick’s analog sensor.

Given the requirements on the inputs the board will need 12 inputs as a minimum in order to support the required features of the mouse. The microcontroller should also ideally be 5V compatible since we will have access to a clean 5V source through the USB connection.

After sorting through these requirements we selected the PIC16F1459 MCU [7].

|  |  |
| --- | --- |
| Requirements | Verifications |
| 1. MCU acknowledges and handle input from Analog and digital pins | 1. Will be confirmed through tests of discrete pushbutton/switch signals as well as analog voltage levels which will be confirmed with a multimeter. |
| 1. MCU is able to handle SPI communication | 1. Will be confirmed through interfacing known working parts with the MCU. |

Figure #8. Microcontroller Requirements

2.4.6 Computer

In this project the computer is the component that requests information from the microcontroller. The computer’s monitor will be able to display the actions of the user’s mouse and joystick. The polling rate of a typical office mouse is 125 Hz, however gaming mouses commonly reach 1000 Hz. In modern computers the increased polling frequency does not have any negative effects on the rest of the computer’s usage while providing slightly better response. Any common computer running any operating system should be able to use our finished mouse provided a driver. The computer places the constraint on our mouse that suggests we cooperate with established standards and common drivers.

|  |  |
| --- | --- |
| Requirements | Verifications |
| 1. The computer will cooperate with common driver protocols | 1. A. The MCU will be programmed as a common driver initially and connected to the computer.   B. The computer will be able to recognize the device plugged into its port |

Figure #9. Computer Requirements

* 1. Supporting Material

2.6.1 Circuit Schematic

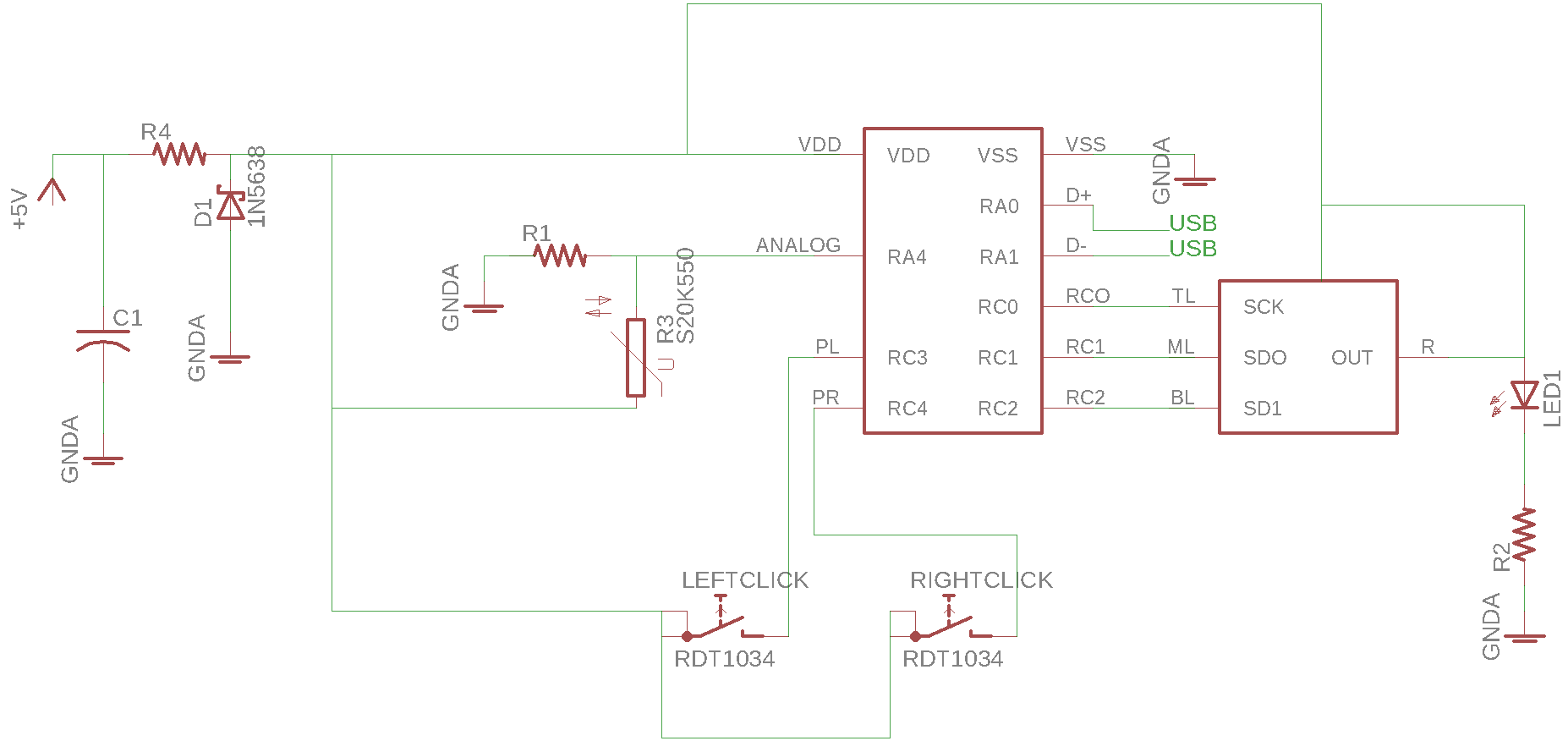


Figure #10. Circuit Schematic

2.6.2 Software Flow Chart

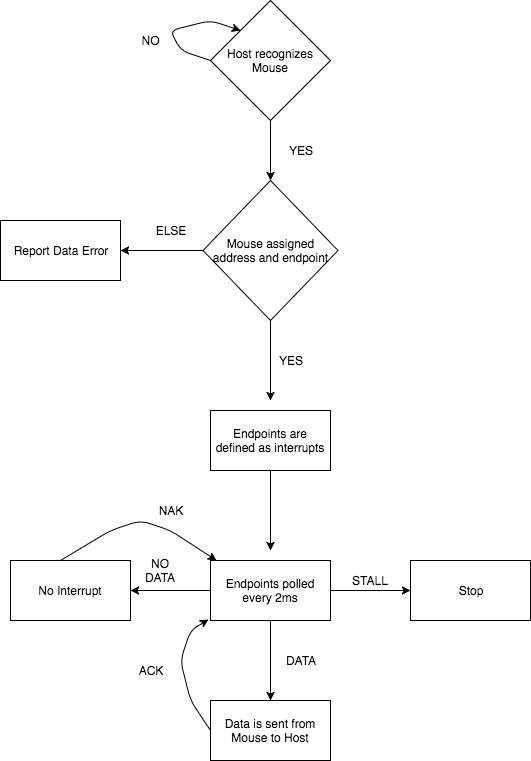


Figure #11. Software Flow Chart

The first step for successful communication with the host to the mouse is for the host to recognize the mouse is plugged into its USB port. Once the host acknowledges the mouse’s presence, the computer will assign the mouse a specific individual address. Next, the mouse will declare how many endpoints are needed and they will be set to interrupt. These endpoints are the communication link from the computer to the actual mouse. The endpoints will contain all the information from the mouse’s sensors and organized by the microcontroller. Additionally, the endpoints will be polled every 2ms. If new data is sent from the mouse is from the mouse to the host the computer will send an ACK signal, declaring that it recognizes the new information. On the other hand, if the host polls the endpoints and no new data is sent the host will reply with the NAK signal. Finally, if the host polls the endpoint and receives a STALL signal it recognizes it needs intervention from the host and will stop.

2.7 Tolerance Analysis

Gaming mice require low latency and high comfort levels. Most other areas of the mouse are sufficiently abstracted away from the user and the computational requirements, while they exist, are far below what even simple 8-bit microcontrollers are capable of. Some additional circuitry is added to protect the components, read the joystick, and provide light for the optical sensor. Since a USB device is allowed to be plugged in when the host is operating a current is allowed to form which can result in a voltage drop. To protect the circuit a capacitor was inserted between the voltage source and ground in parallel with our load. According to USB specifications capacitances ranging from 1µF to 10µF are permitted to regulate voltage while minimizing inrush. The capacitor also prevents voltage spikes due to inductance while unplugging live devices.

The mouse design also contains a simple voltage divider on the analog portion of the joystick. The voltage divider will split a 5V drop across a variable slider resistor and a fixed resister. The final variable resistor will be selected based on feel so until then it is important to ensure that we will be able to match a resister with our slider that will allow for the analog input to be read accurately. It is important to establish a ratio between the max variable resister value and the required fixed resister. Our MCU has an analog voltage resolution of 10 bits for 5mV increments. We have no more than 30 mm of travel and suspect that half mm movements will be beyond the finest that the thumb can move. For simplicity and extra margin we will assume that we would like 100 unique positions across our thumb movement range. We also will assume that noise may take away from our resolution so we assume that the circuit will still have 8-bit resolution (20mV increments). This still gives us 250 counts. We can fit all the counts we need within a 2V range from the voltage drop across our variable resister. This means given a variable resister that goes from 0 Ω to 10K Ω we would use a resistor with no more than 15K Ω in order to have sufficient voltage drop. At the same time we want the fixed resistor to have as high of resistance as possible to provide the best linearity (using a 1 ohm fixed resistor would result in a large voltage change initially and then almost none following). The precise value of the resistors does not matter as much as the proportion between them. If linearity does prove to be an issue later on in the testing it will always be possible to correct for any linearity issue through MCU processing.

3 Schedule

|  |  |  |
| --- | --- | --- |
|  | Michael | Drake |
| Week 1 2/26 | Begin MCU Programming | Optical sensor Arduino testing |
| Week 2 3/5 | Finish MCU Programming | Analog sensors measurements with Arduino |
| Week 3 3/12 | SPI Communication | Push button verification + Initial PCB design finalization |
| Week 4 3/19 | Host to MCU Communication  Host recognizes mouse | Host to MCU communication |
| Week 5 3/26 | Host to MCU Communication  Host assigns address and endpoints to mouse | Final PCB order |
| Week 6 4/2 | Host to MCU Communication  Computer is able to handle the information to mouse | 3D Modeling and Printing |
| Week 7 4/9 | Flex Week | 3D Print + Test Prototype |
| Week 8 4/16 | Mock Demo + Finish final report | Mock Demo + Finish final report |
| Week 9 4/23 | Present Product | Present Product |

Table #2. Schedule

4 Costs

4.1 Labor

The largest cost in design of this mouse will be the labor so it is important to estimate the man hours and the value of those man hours. Using time spent weekly on the project so far as a guide suggests 10-12 man hours per week will be spent on this project for the duration of the class. The schedule we have set does not suggest this number will need to change significantly. The value of the labor is estimated to be $30/hour given the type of work and level of competence of the labor. This semester should have 12 weeks of true design work for a total estimate cost of $30\*12\*2.5\*12= $10,800.

4.2 Parts

Given the estimated price of labor, it is important to consider the tradeoff of purchasing parts that may allow for less design effort (especially components that provide sufficient abstraction). Doing so is likely to save in up front development costs, however they will also affect the profitability of the finished product, especially at higher quantities.

|  |  |  |  |
| --- | --- | --- | --- |
| **Manufacturer /Part Number** | **Description** | **Purchased From** | **Quantity @ Cost** |
| CJMCU-110 | Optical Flow Sensor Module | Ebay | 1 @ $15.40 |
| Alps RS15H113CA05 | Slide Potentiometer 15mm | Mouser #  688-RS15H113CA05 | 1 @ $2.55 |
| Sparkfun BOB-12700 | USB A Female Breakout board | Mouser # 474-BOB-12700 | 1 @ $3.95 |
| Bourns PTA1543-2010CIB103 | Slide Potentiometers 10MM | Mouser #  652-PTA15432010CIB10 | 1 @ $1.07 |
| Bourns PTA1543-2010DPB103 | Slide Potentiometer 10MM | Mouser #  652-PTA15432010DPB10 | 1 @ $1.06 |
| NKK Switches M2024TNW01-DH | Rocker Switch ON-ON-ON | Mouser #  633-M2024TNW01-DH | 1 @ $8.18 |
| E-Switch 400AWMSP4R2BLKM6QE | Rocker Switch SPDT | Mouser # 612-40WMSP4R2BLKM6QE | 1 @ $2.98 |
| Microchip Technology PIC16F1459-I/SS | 8 bit MCU, USB, SPI, smnt | Mouser # 579-PIC16F1459-I/SS | 1 @ $1.95 |
| TBD | Printed Circuit Board |  | Est. $50 |
| Common Parts | LED, Click Buttons | ECEB Parts Bins |  |

Table #3. Cost of Parts

Combining our labor cost of $10,800 and parts cost of $87.14 gives the total estimated cost to our project to be $10,887.

5 Safety and Ethics

5.1 Safety

Given the low voltage we will be using as our power supply our biggest safety concern is to the hardware and making sure we aren't exposed to dangerous metals such as lead. We will be ordering RoHs compliant components to prevent unnecessary exposure. We will also take precautions when soldering components, both while prototyping and also for assembling final components to the PCB.

Additionally to protect the mouse, a Zener diode is placed in the device to protect the circuit overvoltage and protecting the host’s port above 5.25V. Also, a large voltage can occur in the circuit when the device is removed due to the USB cable’s inductive properties. To avoid this large voltage spike a capacitor was placed across the voltage source to ground.

To protect the user, a warning will be placed on the mouse. This warning will caution the user to avoid looking into the laser to avoid potentially damaging the user’s eye. The optical sensor selected would require multiple minutes of direct eye exposure to cause damage to the retina.

5.2 Ethics

We do not anticipate that the development of our product would unleash any possibilities of misuse that did not already exist with other computer mice. The largest ethical concern is that we must consider existing products that are similar and be confident that our product is sufficiently differentiated and unique so as to not upset any existing works which would violate number seven in the IEEE Code of Ethics [5].

References

[1] Enhanced Host Controller Interface Specification. [Online]. Available:<https://www.intel.com/content/www/us/en/io/universal-serial-bus/ehci-specification.html>. [Accessed 14-Feb-2018]

[2] Serial Peripheral Interface (SPI). [Online]. Available:<https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi>. [Accessed 14-Feb-2018]

[3] USB Protocols. [Online]. Available:<http://www.beyondlogic.org/usbnutshell/usb3.shtml>. [Accessed 16-Feb-2018]

[4] How many clicks the best gaming mice can handle. [Online]. Available:<https://mybroadband.co.za/news/gaming/208870-how-many-clicks-the-best-gaming-mice-can-handle.html>. [Accessed 16-Feb-2018]

[5] 7.8 IEEE Code of Ethics. [Online]. Available:<https://www.ieee.org/about/corporate/governance/p7-8.html>. [Accessed 17-Feb-2018]

[6] This '4D' mouse has a joystick, but that doesn't make it good. [Online]. Available:

[https://www.pcgamer.com/this-mouse-has-a-joystick-but-that-doesnt-make-it-good/.[Accessed](about:blank) 19-Feb-2017]

[7] PIC16 Microcontroller Datasheet. [Online]. Available: <https://www.microchip.com/wwwproducts/en/PIC16F1459>. [Accessed 4-Mar-2018]