Perfect Posture

ECE 445 Design Document

Apoorva Josyula, Rohan Kanianchalil, Julianna Gecsey
Jgecsey2@illinois.edu, josyula3@illinois.edu, rohansk2@illinois.du

Group 38

TA: Daniel Vargas

Spring 2021

1 Introduction

1.1 Objective

In the time of Covid-19, we are seeing a rise of the “WFH” lifestyle as more and more non-essential workers are having their offices be shut done and their work be moved into a remote setting. According to an April Facebook survey from the American Chiropractic Association, 92 percent of chiropractors (out of 213 respondents) said that patients report more neck pain, back pain or other musculoskeletal issues since the stay-at-home guidance began [1]. The most common posture related issues that are arising are coming from the upper back to neck area, Scott Bautch, the president of the American Chiropractic Association’s Council on Occupational Health, says that as screen time has exploded, we’re more at risk of “Text Neck” and “Selfie Elbow” [1]. We recognize that this issue will only continue to get worse as more jobs continue to become remote/require more time spent sitting in front of a desk, thus our solution is to present our device “Perfect Posture”. Through the means of the time, we get in ECE 445, we plan to create a wearable posture device that can be attached to a user’s back that will accurately assess a person’s posture and the alert the user on the potential necessary adjustment they will need to make on their posture. Our goal is to help train a user to recognize back problems and eventually create the habit of working with perfect posture.

1.2 Background

The products for creating “perfect posture” devices are endless, from back braces to even single system wearable devices that one can wear right on their back. These products however all lack in holistically assessing a person’s posture. When it comes to physical braces, the external mechanics of a back brace force a person’s body/muscles to rely on the physical weight of the brace to hold up their back posture. This can be harmful as the body will become dependent on that physical brace to help keep them upright and could led to in some cases muscle dystrophia with the lack of use of muscle in the back region. [2]. Other wearable smart devices, such as the “Upright Go” are helpful in eliminating the need for a physical back brace however they typically only focus one aspect of a person’s spine. As studies have shown, the result of bad back posture is not just isolated to one specific area of a person’s spine but has to deal with the holistic approach of maintaining one’s spinal curvature. [3]. These posture devices don’t take into account how a person’s spine can be held “upright” but fail to maintain a healthy curvature of the total spine. [3].

This is one of the many reasons why we have created our device to not just be a posture sensor that detects whether a person’s upper spine is being held upright, but to calculate a person’s spinal curvature to see if they are maintaining the healthy amount of tension and bend in their upright position. [3].

Back problems are typically excluded to being thought of as issues with lower back angles which neglects solving all posture issues since posture follows through with the neck. While the lower back is important the upper lordosis and cervical spine connect to form an important C-shape that is proportional in all humans. While an action like sitting in an office chair may not have many negative effects on the lower back, it is very common for people to position themselves into the “head forward” position in which this C-shape is no longer maintained [2]. Although this position does not cause many issues in the beginning, over time this causes kyphosis, muscle dystrophy, and is linked to future lower back issues. In order to determine the severity of one’s head forward position doctors measure a patient’s craniovertebral angle which is between the tragus of the ear and the C7 vertebrae of the spine [3].

We recognize that it is humanly impossible to never move from your given posture, there will be times that will require a person to slouch for a bit. Our device shouldn’t be going off every second if a person were to just bend down to pick up a pencil. Thus, there is leeway in how long a person can be in a certain position without it being considered “detrimental” to that person’s given posture.

With the knowledge of how we need to be maintaining these angles and for how long can time elapse before a person engages in bad posture, we as ECE students will create a device that will accurately be able to detect deviances from these important angles and create a system that will help alert the user on when they have been continuously in a bad posture.

1.3 High Level Requirements

- Use the accelerometer and gyroscope to calculate the change in degree from the ideal C-Shape of the spine.

- Data taken from the microprocessor via Bluetooth will be processed and interpreted visually through an app on an individual preferred device.

- Through a vibrational motor within our physical device, will notify the user over a 2-minute interval whether their posture is off the ideal degree.

2. Design

2.1 Block Diagram

Our wearable device will require the following four subsystems to be considered successful: A sensor/feedback system, a control unit, a power system, and an app interface. The layout is shown below in Fig. 1. The power system will ensure that the system can be powered for the extended period of time a user might be spent working with the proper 3.3V. Our sensor system will help to read a person’s spinal position and our feedback system will help to alert the user when that spinal position is off after 2 minutes in that position. The control unit will consist of our microcontroller (the ATMEGA328 chip) that will read sensor data and relay them to our wireless transmitter (Bluetooth). It will also be in charge of alerting our feedback system when it has gone over the allotted allowed time for being in bad posture. Our app interface will have the users posture data in an interactive way for the user to see the trends in their posture.

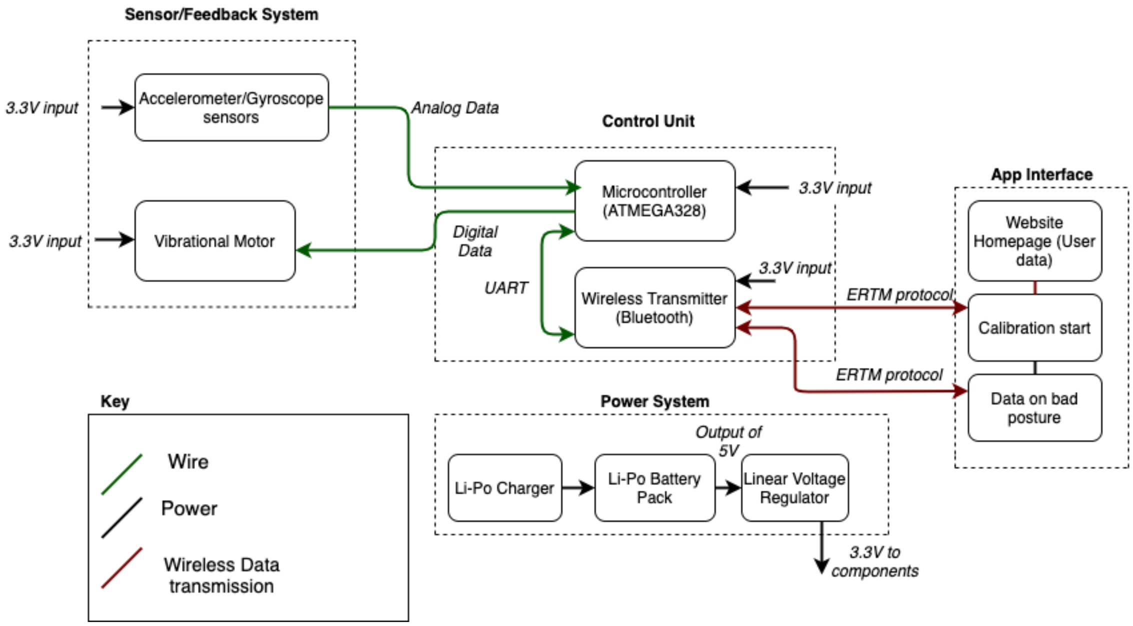


Fig 1. Block Diagram of Perfect Posture

2.3 Physical Design

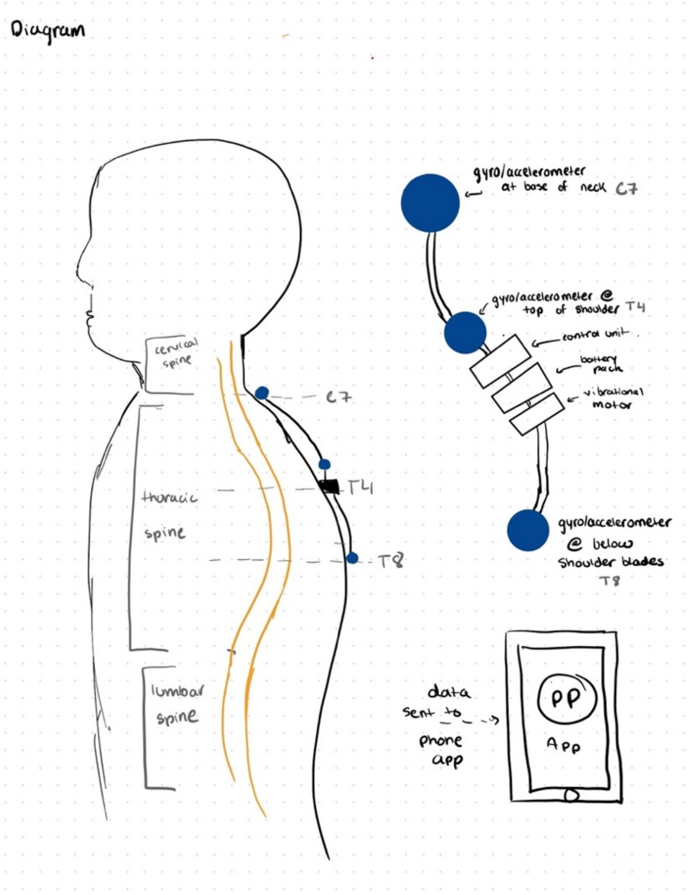


Diagram
cervical spine
thoracic spine
lumbar spine
C7
T4
T8
gyro/accelerometer at base of neck C7
gyro/accelerometer @ top of shoulder T4
control unit
battery pack
vibrational motor
gyro/accelerometer @ below shoulder blades T8
data sent to phone app
PP
App

Fig 2. Physical Design

An important thing to note about our physical design is that we want to keep this relatively small. The individual sensor components and vibrational motor will be no large than a 4x4cm design. Each of our components will be encased in a 3-D printed plastic case. The control unit and battery pack will have the leeway to be a bit larger (around 11x8cm design). The weight in order to not affect the individual posture will be no larger than 150 grams. We will use a soft textile to run our wires through. That will connect the different components without irritating the users back. The attachment to the actual person's back will consist of a skin safe adhesive a user can either attach directly to their back or on top of their clothes.

2.4 Functional Overview & Block Diagram Requirements

2.4.1 Senor/Feedback System

Our sensor subsystem will consist of a gyroscope and an accelerometer. The combination of these two sensors will give us a raw data value of the x,y,z plane position of where the sensors are at. Three sensor units will be placed along the spine to ensure we can read an accurate picture of how a person's "C-shaped" upper spine position looks like. The sensors will relay their positional data to the microcontroller, where it will store the changes of values when there is significant movement.

The feedback system will be our vibrational motor. The vibrational motor will be in charge of alerting the user whenever their posture is off for a significant amount of time. It will receive either a High signal (indicating the posture is off so turn on motor) or Low signal (posture is fine for now so stop motor). The vibrational motor will be set off again whenever the time spent in the bad posture continues to increase.

Accelerometer Requirements and Verification

Requirement	Verification
- Have a steady output of readings that doesn't deviate more than 0.2g when in a stable position for more than 2 seconds.	1. Test accelerometer by first powering the device to a 5.0V power in the VCC pin, ground to GND pin, and the data pins onto an Arduino. 2. Connect Arduino to computer and have it be connected to an Arduino Serial Monitor. 3. Write program that will print out the raw g values onto the monitor. 4. Lay down accelerometer without moving and check to see if reading is saying 0.5g in the X, Y axis and 1g in the Z axis (with variation from 0.2g). 5. Rotating about the x-axis, rotate accelerometer 90 degrees, and check to see if the reading is showing change

	of Y axis to be 1g and see if the value is kept at 1g for 2 seconds. 6. Rotate to 180 degrees now, and check if the values now read, 0.5g, 0.5g, 0.0g in the x,y,z plane and are steady at that reading for more than 2 seconds.

Gyroscope Requirement and Verification

Requirement	Verification
- Have a steady output of readings that doesn't deviate more than 2 degrees when in a stable position for more than 2 seconds.	1. Test gyroscope by first powering the device to a 5.0V power in the VCC pin, ground to GND pin, and the data pins onto an Arduino. 2. Connect Arduino to computer and have it be connected to an Arduino Serial Monitor. 3. Write program that will print out the raw angle values onto the monitor. 4. Lay gyroscope flat on stable surface and have the monitor read 0 degrees (with 2-degree variation). 5. Rotate gyroscope to 90 degrees and see if value is accurate on the serial reading and holds the value steady for more than 2 seconds. 6. Rotate gyroscope to 180 degrees and see if value is accurate on the serial reading and holds the value steady for more than 2 seconds.

Vibrational Motor Requirement and Verification

Requirement	Verification
- When microcontroller sends HIGH signal to motor, the motor will vibrate for 10 seconds to indicate bad posture.	1. Connect vibrational motor to a 5V power source and connect to our microcontroller chip as well. 2. With our microcontroller connected from USB to our computer, we will write an Arduino program that is going to turn on the digital pin to HIGH for 10 seconds and then LOW for 10 seconds.

	3. Hold vibrational motor feel for vibrations when switched onto high. 4. Record the time the vibrational motor is held on HIGH to see if it stays on for 10 seconds.

2.4.2 Power System

The power subsystem consists of a Li-Po Charger, Li-Po Battery Pack, and a Linear Voltage regulator all connected to one another through wires, and eventually attached to the feedback subsystem. This subsystem ensures that necessary power is delivered to each component meanwhile regulates the voltage delivered to provide user safety.

The Li-Po Battery Pack itself outputs at 3.7V at 1000mAh. The Li-Po Charger will help charge the battery without having to change out the Li-Po Battery Pack every time a user wants to use the device. Our linear voltage regulator will help decrease the 3.7V output to be 3.3V which is ideal for the components of our device.

Li-Po Charger Requirements and Verification

Requirements	Verification
- Charger should be outputting 4.2V +/- 0.03V at 400mA +/- 10mA when plugged into USB charging plug.	1. Plug in charger from USB end to computer to start charger. 2. Connect an oscilloscope to the charger. One terminal is at the negative end of the charger, while the other terminal is at the positive end. This will help us to measure voltage across the charger. 3. Unplug oscilloscope and plug in ammeter in series with the charger. 4. Once in series, measure current through and check to see if there is a steady output current of 400mA +/- 10mA.
- Charger should stop outputting voltage supply to Li-Po battery once battery reached max capacity at 3.9V +/- 0.03V	1. Connect an uncharged Li-Po battery to Li-Po charger, for a full charge of the Li-Po battery at the current rate of 400mA, that should take from anywhere from 50 minutes to 70 minutes. 2. Taking the oscilloscope, probe the charger across the output terminals of the charger. One terminal end is at the positive terminal end of the charge,

	while the other is at the negative terminal end of the charger. 3. Once battery is fully charged, the output voltage across the charger should be 0V. 4. Remove the oscilloscope, and place on ammeter in series with the charger. This reading of the current, once the battery is fully charged should read 0A.

Li-Po Battery Pack Requirements and Verification

Requirement	Verification
- At full charged capacity, battery should be outputting a voltage charge of 3.7V +/- 0.2 V at 400 mA for maximum current	1. Use an uncharged Li-Po battery, and plug into a Li-Po charger. The LED indicator on the Li-Po charger will show when the Li-Po battery has been fully charged (around 50 to 70 minutes). 2. Once charged, connect an oscilloscope to the battery (one end to the positive terminal of battery, other end to the negative terminal of battery). 3. Graph the voltage readings to ensure, the output voltage is around 3.7V +/- 0.2V. 4. To measure current output, with the oscilloscope already connected, connect the battery to a 10ohm resistor, and check current output on graph.
- _Fully charged, the battery should last around 4 +/- 1 hours with continuous usage.	1. To verify the usage of the battery, with the full circuit's components connected, keep oscilloscope plugged into the ends of the battery as described above. 2. Check to see where the oscilloscope current readings and voltage readings are at. 3. Given under the specs of the battery, the battery should run at 400mAh at continuous use, so divide total current use of the circuit by 400mAh to determine, length of time the battery is

	able to sustain continuous use of the device.

Linear Voltage Regulator Requirements and Verification

Requirements	Verification
- Given a voltage of 3.7 +/- 0.4V, our linear voltage regulator must output a voltage of 3.3 +/- 0.05V.	1. Use a function generator to supply the given voltage output. The output from the voltage generator should be 3.7V +/- 0.4V. 2. Attach an oscilloscope where one end is attached to the positive terminal of the voltage regulator, and the other end is attached to the negative terminal of the voltage regulator. 3. Check the graph results from the oscilloscope to see what the output voltage from the linear voltage regulator is showing. The results should be around 3.3V +/- 0.05V

2.4.3 Control Unit

The components of our control unit will consist of a microcontroller chip (ATMega328) along with our wireless transmission, which in our case will be our Bluetooth Module.

For our microcontroller, we want it to read and store the sensor data, along from an I2C bus, and be able to convert the raw g-force and gyroscope values into angular data. It has to compare changes from its original calibrated values to determine if a timer needs to be set for when to send the HIGH signal to our vibrational motor. This time interval will be 2 minutes to allow leeway if the users bad posture is only temporary. That signal and timing will help indicate for a user whether they are in bad posture. All of our angular data information will also be sent to our Bluetooth module so that when a user wants to pair their smart device, they can visually see the changes being made in their posture and how much they improve each time they use the device.

The Bluetooth module is responsible for receiving digital values that it gets from our microcontroller and sending it to a user's paired device, through the form of a front-end app. The exact Bluetooth module we will be using will be the JY-MCU BT_Board following the ERTM protocol between Bluetooth module and computer.

Microcontroller Requirement and Verification

Requirements	Verification
- Be able to read raw sensor data and calculate an angle approximation via I2C bus at a minimal rate of 5Hz.	1. Attach our sensors by the analog pins on the microcontroller.

	2. Flash user defined code onto microcontroller that will take inputs from sensors as inputs to angle calculator function. 3. Place sensors at predetermined angles (using protractor for best angle reading). 4. On the Arduino serial monitor, wait to see if the function code will return the predetermined angles. 5. Using an oscilloscope probe the negative and positive output terminals of microcontroller. 6. Determine on graph to see frequency rate.

Bluetooth Module Requirement and Verification

Requirements	Verification
- Bluetooth sends values of angular values received from the microcontroller to the computer front-end app. - Communicate with UART from the microcontroller and Bluetooth Module - Results are reflected on prototype app	1. Attaching the connection between the microcontroller and the Bluetooth (using the TXD pins) and set test angle values to the microcontroller. 2. Have Bluetooth be connected to computer through a preliminary test app. Have the test app print continuous values in terminal received from the Bluetooth module to see if the angle values are being reflected from the pre-determined ones.

2.6 Schematics

2.6.1 Power System

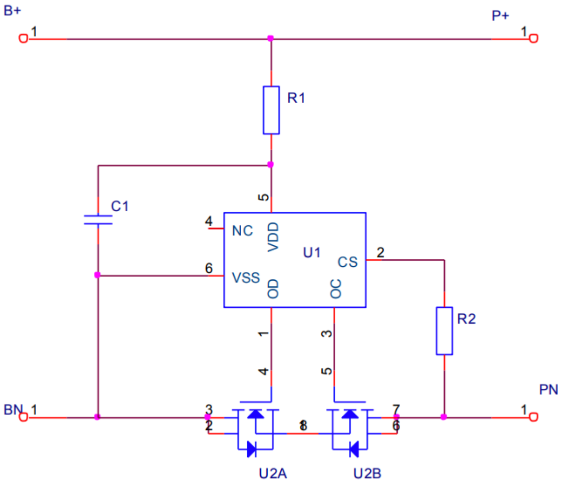
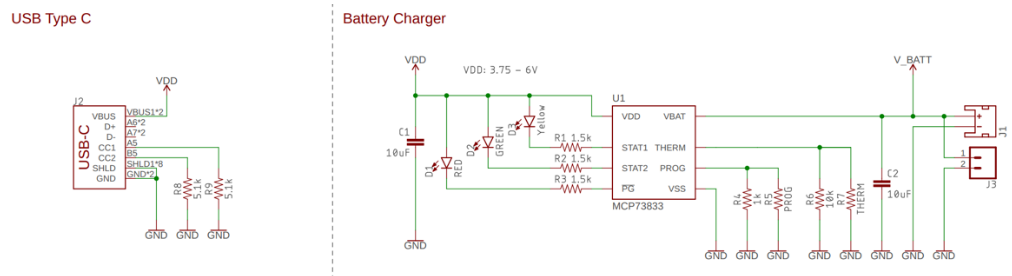


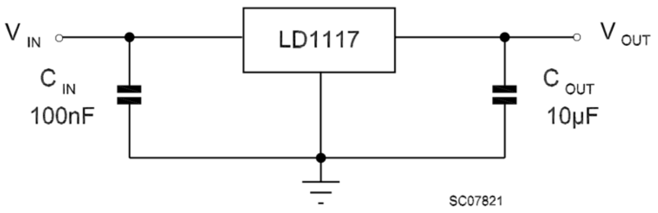
Fig. 3 Circuit Schematic for Li-Po Battery (DTP502535) [4.1]



R_PROG (kΩ) = 1000 / I_REG (mA)

R_PROG (kΩ)	I_REG (mA)
1.00	1000
1.11	900
1.25	800
1.43	700
1.67	600
2.00	500
2.50	400
3.33	300
5.00	200
10.0	100

Fig 4. Circuit Schematic for Li-Po charger (MCP73833/4) [5.1]



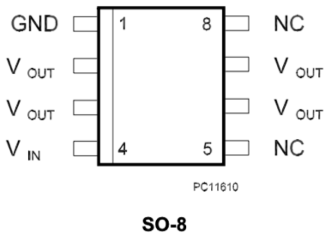


Fig 3. Circuit Schematic of Linear Voltage Regulator (LD1117D33TR). [6.1] [6.1]

Fig 4. Pin Connections

2.6.2 Sensor/Feedback System

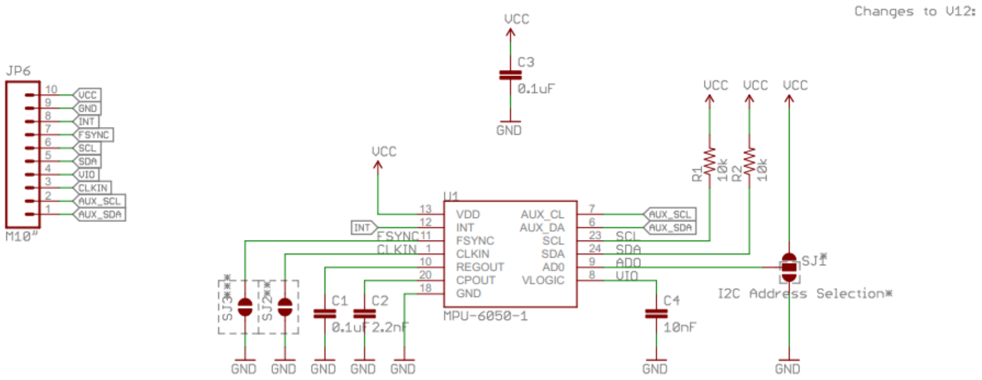


Fig 5. Circuit Schematic for Triple-Axis Accelerometer/Gyroscope (MPU-6050) [7.1]

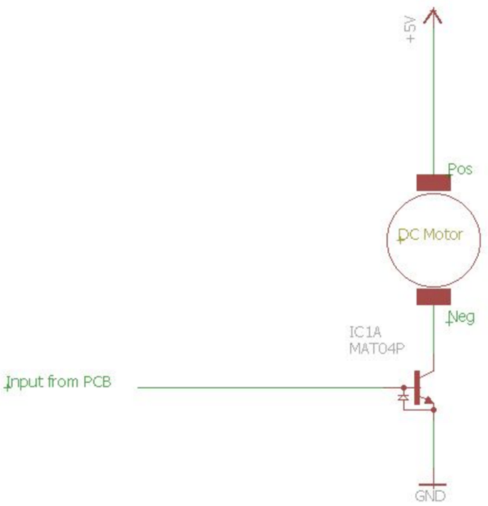


Fig 6. Circuit Schematic for Vibrational Motor (ROB-08449)

2.6.3 Control Unit

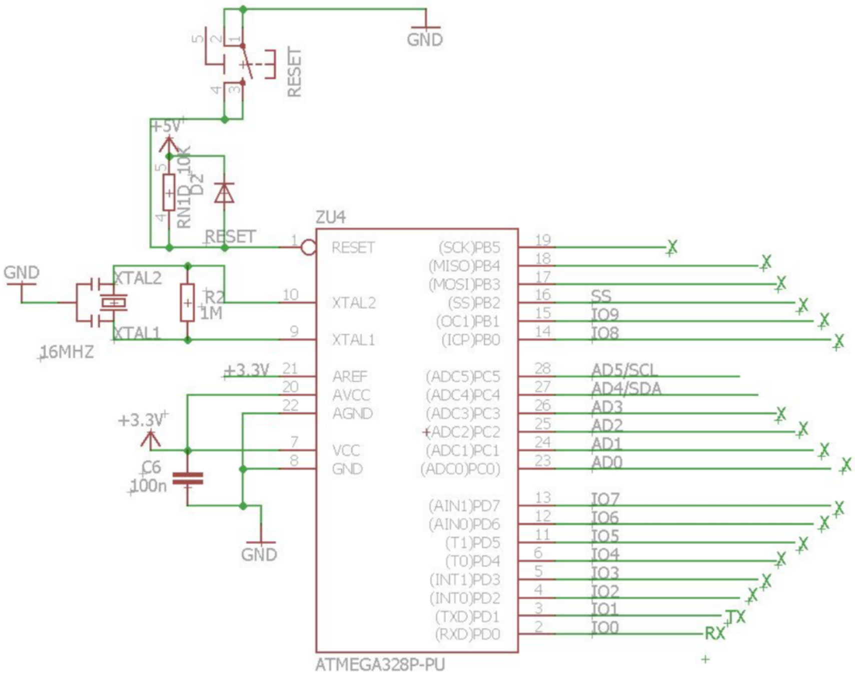


Fig 7. Circuit Schematic for AtMega328 Microcontroller

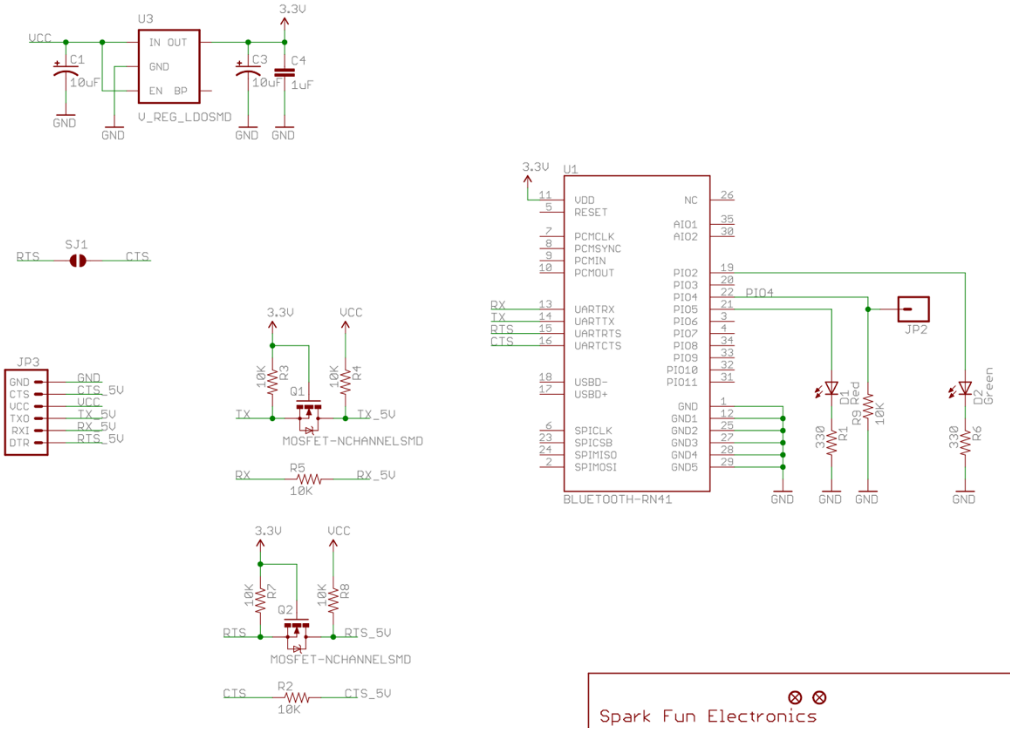


Fig 8. Circuit Schematic for Bluetooth Module (RN-41) [8.1]

2.7 Tolerance Analysis

One key piece of information that we need to keep track of throughout our design process is the actual angle of the user's spine and how far deviant it is from the normal spine angle. The way most medical professionals measure whether an individual has poor posture, or a spine imperfection is the Cobb Angle, which measures the deviation of a spine from its ideal position. If the Cobb angle is in the range of 1-10 degrees, it is indicative of a healthy spine, 10-24 is mild scoliosis, 24-39 is mild scoliosis and anything above 39 degrees is considered severe [9]. The formula to measure the Cobb Angle is

$$Y = 2D \sin\left(\frac{X}{2}\right)$$

Eq 1. Cobb's Angle Formula

where D is the distance between the top vertebrae measured and the bottom vertebrae within the C shape of the spine, X is the angle measured of the curve and Y is the convexo-concave vertebral difference [10]. Because we do not have X-Rays of the individual's spine and each individual vertebra, we strategically placed our gyroscopes and accelerometers in such a way that each encapsulates the top, middle and bottom of the C shape of the spine, giving us a good estimate of the potential Cobb angle. Because the nature of the Cobb angle is two-dimensional, and the spine is three-dimensional, we must do this calculation three times (between the upper and middle, the middle and lower and upper and lower of the spine) to get an accurate depiction of the spine's condition [11].

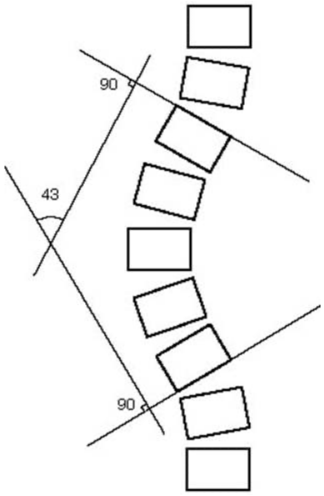


Fig 10. Depiction of measurement on the Cobb Angle

We previously calibrated each of the gyroscopes to the curve of our users back, so to calculate the angle used in the Cobb angle, we would monitor and receive the change in angle from each gyroscope from its initial calibrated position and do those 3 separate calculation combinations. The position of the Top and Bottom gyroscopes when initially placed on the spine will not deviate, so the D value will remain the same, allowing us to get a fairly accurate representation of our user's spinal condition. Our accelerometer's raw g-force values will not be that helpful, but we can use them to determine orientation about the axis.

$$A_x = tan^{-1}\left(\frac{x}{\sqrt{y^2+z^2}}\right)$$

$$A_y = tan^{-1}\left(\frac{y}{\sqrt{x^2+z^2}}\right)$$

Eq 2: Angles about the x and y axis

These calculations will then be relayed to our microcontroller where the user will be notified if their posture is off and whether their current position may lead to potential back problems. In a medical environment an X-Ray would be necessary to garner the exact severity of one's spine, however, we aren't necessarily treating spinal injuries, instead we are designing a product that will indicate deviations in an attempt to stop those changes before permanent damage occurs.

3 Safety & Ethics

When creating our project, we have to consider a few safety issues as well as some ethical as well.

3.1 Safety

Starting with the issues that could possibly occur during the development of the project is the usage of 3 accelerometers and gyroscopes. With that many being in use, if we are not careful with how things are wired and powered, it may cause a malfunction in our project. Continuing on, with the usage of a microcontroller and a Bluetooth module, if we miscode the microcontroller, we may cause an adverse effect in our project.

Regarding the physical issues with the device in regard to the user, we have to consider many factors. We must consider the voltage of the battery when designing our implementation so that it may not cause harm to the user. We must also consider how the accelerometer and gyroscope may affect the wearer and the rpm of the vibrational motor so that it doesn't hurt the user.

Finally, a factor to consider is the type of adhesive used when placing the device on the user. There are few ethical codes that we must worry about when we design our project. Because our implementation is physically being put on a person, we have to make sure that we follow code 1.2 of the ACM Code of Ethics: Avoid Harm [12]. We must make sure the adhesive and the components of our invention that are being placed on a person will not harm them in any way and cannot be used in such a way that it negatively impacts the user.

3.2 Ethics

With our project including an app interface, we have to also consider code 1.6 of the ACM Code of Ethics: Respecting Privacy [12]. Our project constantly monitors an individual's posture as well as many calibration factors, such as weight, height, age, etc., so we must keep the user's information confidential as those factors are private to oneself.

One more ethical issue that we should highlight and make sure to follow is the IEEE code of ethics 7.8 subsection 10: to support colleagues and co-workers in following this code of ethics, to strive to ensure the code is upheld, and to not retaliate against individuals reporting a violation.

[13]. This is a code that we have to uphold because we are making something from scratch, which entails 100’s of hours of potential work.

We must make sure that although we must put that amount of effort into our project, that each individual in our group is making sure that they are both physically and emotionally well and ensure that there is a balance between the work needed for this project as well as a healthy off-work life.

References

[1] nytimes.com, “The Pandemic of Work From Home Injuries”, 2020. [Online]. Available: https://www.nytimes.com/2020/09/04/well/live/ergonomics-work-from-home-injuries.html?fbclid=IwAR3y4TwqmDq2ZLAoYVMlpikjyPZH-7xRMcqfMyD-eQoSS3FaI5UZ14nc5zU [Accessed 1-March-2021]

[2] spine-health.com, “How to Measure and Fix Head Posture”, 2018. [Online]. Available: https://www.spine-health.com/conditions/neck-pain/how-measure-and-fix-forward-head-posture.com [Accessed 18-Feb-2021]

[3] spine-heath.com, “Forward Head Posture’s Effect on the Cervical Spine”, 2018. [Online]. Available: https://www.spine-health.com/conditions/neck-pain/forward-head-postures-effect-cervical-spine.com [Accessed 18-Feb-2021]

[4] Sparkfun Electronics, “Polymer Li-ion Rechargeable Battery,”[PDF]. Available: https://cdn.sparkfun.com/datasheets/Prototyping/spe-00-502535-400mah-en-1.0ver.pdf [Accessed 1-March-2021]

[5] Sparkfun Electronics, “Stand-Alone Linear Li-Ion / Li-Polymer Charge Management Controller” [PDF]. Available: https://cdn.sparkfun.com/assets/b/a/7/6/8/MCP73833Datasheet.pdf [Accessed 1-March-2021]

[6] Sparkfun Electronics, “LOW DROP FIXED AND ADJUSTABLE POSITIVE VOLTAGE REGULATORS,” [PDF]. Available: https://www.sparkfun.com/datasheets/Components/LD1117V33.pdf [Accessed 1-March-2021]

[7] Sparkfun Electronics, “MPU-6000 and MPU-6050 Product Specification Revision 3.4,” [PDF]. Available: https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf [Accessed 1-March-2021]

[8] Sparkfun Electronics, “Class 1 Bluetooth® Module,” [PDF]. Available: http://cdn.sparkfun.com/datasheets/Wireless/Bluetooth/Bluetooth-RN-41-DS.pdf [Accessed 1-March-2021]

[9] spine-health.com, “Cobb Angle used to measure Scoliosis Curve”, 2018[Online]. Available: https://www.spine-health.com/conditions/scoliosis/cobb-angle-used-measure-scoliosis-curves [Accessed 1-March-2021]

[10] ncbi.nlm.nih.gov, “Expressing Cobb Angle as a Linear Measurement in Scoliosis”, 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5769934/#:~:text=sin%20(X%2F2)%20 [Accessed 1-March-2021]

[11] Clear-Institute.org, “Cobb Angle”, 2019. [Online]. Available: https://clear-institute.org/learning-about-scoliosis/cobb-

angle/#:~:text=The%20orthopedic%20%E2%80%9CGold%20Standard%E2%80%9D%20for,of%20the%20inferior%20tilted%20vertebra [Accessed 1-March-2021]

[12] acm.org, "ACM Code of Ethics", 2018. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 18- Feb- 2021].

[13] Ieee.org"IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html . [Accessed: 18- Feb- 2021].