Electrical and Biomedical 4BI6

Final Report

Development of a Novel Wearable Posture Correction Apparatus Using Advanced Accelerometry Techniques

by

David Zhitomirsky (0654728) (Partner Nanxi Zha)

Department of Electrical and Computer Engineering

Faculty Advisor: Prof. Doyle, Supervisor: Prof. deBruin

Electrical and Biomedical Engineering Project Report

Submitted in partial fulfillment of the requirements

for the degree of Bachelor of Engineering

McMaster University

Hamilton, Ontario, Canada

April, 2010

ABSTRACT

A novel posture correction apparatus was developed with the aim of stroke patient rehabilitation, specifically designed for patients with bodily neglect. The apparatus is based on an accelerometer system, capable of tracking tilt in three dimensions. The apparatus is portable and worn on the chest, and is extendable in design to other parts of the body. The approach to tilt sensing and posture correction taken here relies on small DC motor actuators implemented to notify the user of incorrect posture, based on their direction of tilt. The design takes into account patient differences, and has calibration procedures to accommodate for different users and prevent sensor drift. Tilt is also continuously monitored on a remote computer, where signals from the accelerometer are transmitted wirelessly via a microcontroller and RF module. It was found that the system could reliably track tilt and thus posture, and the feedback mechanism was effective at notifying the user about incorrect posture. Thus, this mountable system is an effective way to track and monitor posture, and is especially applicable to stroke patient rehabilitation.

ACKNOWLEDGEMENTS

This work was made possible under the guidance of Professor deBruin, who gave invaluable advice in regards to sensor implementation and system design.

TABLE OF CONTENTS

Contents

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Background

Biomedical engineering refers to using engineering practices and problem solving, integration of simple systems to form practical and complex apparatuses, with the aim of improving health care and aiding health providers by simplifying arduous and time consuming medical processes. A medical field deserving much attention deals with gait and posture, as it is directly applicable to patients with disorders as well as the average person. Improper posture may lead to problems such as thoracic spine pain, lower back pain, and neck pain [1]. As well, patients recovering from stroke or similar conditions require monitoring of posture [2]. As such, numerous methods of human position tracking have been developed; these include force platforms, camera systems, magnetic systems and ultrasound [3]. However, these methods are quite complex and expensive, require skilled technicians to oversee trials, and are confined to the laboratory environment. A more attractive solution is the use of body-mounted sensors, such as accelerometers or gyroscopes, with the former being a much cheaper solution. Accelerometers measure linear acceleration and gyroscopes detect rotation, a combination of these two may yield a complete motion system [4]. However, to reduce costs, it is desirable to implement a system capable of producing similar performance using only accelerometers. Several recent studies have shown that accelerometers can be used to track both static and dynamic activities, and this has shown much promise in gait and balance analysis systems [3-10]. As well, research has also been dedicated to posture correction of stroke patients [2, 11]. An accelerometer is capable of transducing static information directly related to posture, as well as dynamic data attributed to motion, as such, it will be possible to regulate and adjust posture as the subject performs regular daily tasks. Several studies attempted to track posture changes; however, few have specifically targeted a complete wearable system. A complete posture tracking system with intelligent feedback mechanisms has yet to be implemented, as well as, testing and adjustment studies based on patient interactions using such a system have yet to be carried out.

1.2 Objective

The aim of this project is to develop a novel wearable posture correction apparatus using advanced techniques of accelerometer sensor positioning, sensor information relay systems and integrated effectors capable of aiding in body positioning. The focus is primarily on developing the hardware aspect of the apparatus, which includes sensor integration and information gathering, as well as output devices to alert subject and aid in posture correction. This is a joint project, and the software component will be handled by Nanxi Zha.

1.3 Approach to Problem

The primary objective of this project is to develop a posture tracking system using accelerometers and to isolate a clear signal for processing purposes. As well, positioning of feedback actuators will be decided based upon physiological data and common posture irregularities. Secondary objectives will include a fully independent system capable of doing both processing and feedback, involving an RF link, microcontroller and feedback actuators as part of the wearable device. Feasibility for these objectives comes from previous studies done in the field involving the implementation of a wireless activity monitoring system using accelerometers. The general approach will be as follows: determine an effective location for the placement of accelerometers on the body, process the accelerometer signal and perform necessary analogue to digital conversion, process the accelerometer data, and finally output result via actuators that will aid patient in posture correction.

1.4 Scope of the Project

The scope of this project is to provide a functional and self contained posture correction system using accelerometers. The implementation will involve one accelerometer, and will allow posture tracking using a device mounted to the chest.

Motor actuators will be used as feedback mechanisms for posture correction. As well, device will have wireless capabilities and a 3D tracking software component performed by the other member of the group. The device is expected to provide tilt sensing of up to 90 degrees from the normal posture stance, and is capable of operating in both the forward and backward, as well as the right and left directions, so that tilt can be detected three dimensionally.

2 Literature Review

2.1 Importance of Posture Correction

Proper posture is relevant to any human being, regardless of age or gender. Maintaining and correcting posture affects the person's physical function, level of abilities and wellbeing. Improper posture may lead to poor postural habits that may ultimately result in muscle imbalance and muscular strain [13]. Furthermore, proper posture may lead to balance in the body, and less fatigue doing regular tasks.

Postural problems typically arise when limbs or spine has been configured in unnatural positions, and joints are experiencing abnormal forces. This results in fatigue and back pain, and symptoms may become chronic. Poor posture acts by shortening muscle fibers, and results in the imbalance of the skeletal system. Particular areas which are affected are the spine and knees, and development of osteoarthritis may cause fusing of bone in abnormal positions due to calcification [13]. It is also believe that performing tasks in the same position, such as sitting at a workplace desk, may lead to muscle imbalance [14, 15]. This is particularly relevant in the workplace and affects a great portion of the population. This muscle imbalance causes certain muscles to be underused and other muscles to be over used and may lead to long term complications.

Stroke is the leading cause of disability in North America, and posture correction plays an important role in patient rehabilitation [16]. The most pertinent part of such rehabilitation is regaining the ability to walk through training of the lower extremities, and proper posture is integral for proper rehabilitation. Post stroke effects can produce effects such as reduced limb function, difficulty walking, and neglect of one side of the body [17]. As such, research has been dedicated to providing patient feedback to correct posture and aid in the rehabilitation process. Weakness or neglect of one side of the body is referred to as hemiparesis, and is a serious concern as it can affect posture due to the patient being unaware of their body positioning. This creates balance impairments, and risk of falling, which has high economic circumstances [18]. Dynamic postural control systems for hemiparesis patients to prevent falls have been investigated [19], and remain an important area of research.

2.2 Relevant Work Conducted in the Field

Significant attention has been given to using accelerometry for the detection of posture, gait and overall body movement [10]. Many groups have focused on implementing fall prevention algorithms using gyroscopes and accelerometers [20, 8, 12]. Seeing as accelerometers provide a direct measure of acceleration, a fall signal is easily detectable and appropriate emergency response apparatuses may be coupled to such devices. Generally, such systems operate by means of tracking regular walking motion, and rely on acceleration thresholds to detect falls. Challenges come in detecting true falls, and often angle and orientation are required in conjunction with acceleration data.

Research has been conducted to study walking patterns and proper gait [5, 11, 6], where accelerometers are placed on key extremities as well as places on the trunk such as thighs. These systems have shown to provide accurate gait information and are of great use in the medical field, especially for posture control during motion. Overall body posture has been reported in many instances [4, 9, 7], where accelerometer systems are used as angle detectors and evaluate body position with respect to some ideal position. These systems offer an attractive method in evaluating posture, and have been shown to be accurate. However, functional wearable devices have yet to be implemented using these approaches.

There has been limited attention to stroke patient recovery in such systems, but some groups have focused on stroke patient rehabilitation and posture correction during walking [2]. As well, fully wearable systems have been constructed for real-time, wireless posture and gait monitoring [8]. A wearable system targeted specifically at stroke patient recovery has been reported [2], but has seen limited testing and efficacy at correcting posture. The system works via LED and noise feedback, and has accelerometer placement on various parts of the body, measuring tilt via angles.

2.3 Tilt Sensing with Accelerometers

Freescale Semiconductor has provided materials related to its MMA7260Q series accelerometers. These application notes are adaptable to most accelerometers and define how to convert accelerometer voltages into accelerations [21]. Subsequently, using acceleration data it is possible to track angular position of the accelerometer using one, two or three axes [22]. As well proper calibration procedures are required to find necessary offsets, which are used to calculate accelerations [23]. The basic concepts related to accelerometry are that the output voltage is directly proportional to the acceleration. Once the voltage is known, it is possible to convert this voltage into acceleration, by first subtracting the offset voltage and then scaling this figure by the sensitivity (which converts voltages to accelerations). Voltage offsets are sometimes difficult to determine, especially in 3 axes systems, and thus special calibration procedures are required. Once the accelerations are known along each axes, using some trigonometry it is possible to interpret these accelerations as projection of the gravity vector onto the accelerometer axes. Having these projections is essential to determining angles. If one or two axes solutions are used, it is possible to track angular rotations along a single axis. One accelerometer systems are not very accurate and have different sensitivities depending on the angle of tilt. Two axes systems are optimal for single axis tilt and have a consistent sensitivity over a range of angles. Finally to measure tilt in 3D, it is necessary to have at least 3 axes oriented orthogonally, and optimally one chooses to have a single, 3-axis accelerometer to accomplish this task. It is then possible to fully measure tilt using this set-up. Basic operation of devices to monitor tilt rely on determining tilt left and right, as well as forward and backward. This is determine based on the angle between 2 of the the accelerometer axes and the horizontal plane, as well as the angle between one of the axis and the vertical plane. Using a microprocessor, it is possible to conduct real-time tracking of these angles and provide tilt information.

3 Proper Posture

3.1 Anatomy of Proper Posture

To maintain the best posture, the spine must be in a balanced or neutral position. Conditions such as lower back pain, sciatica, and whiplash are greatly affected by the alignment of the spine and thus posture [24]. Ideal posture allows maximal movement of joins and prevents unnecessary straining of muscles. Figures 1 and 2 show proper postures from various stances. This posture entails holding head up straight, shoulders and hips level; chin parallel to the floor, and shoulders in line with the ears [24]. Additional figures 4 and 5 have been included for improper posture. Essentially, proper spinal posture relies on 3 major curves, shown in figure 3, and maintenance of these may ensure proper posture.

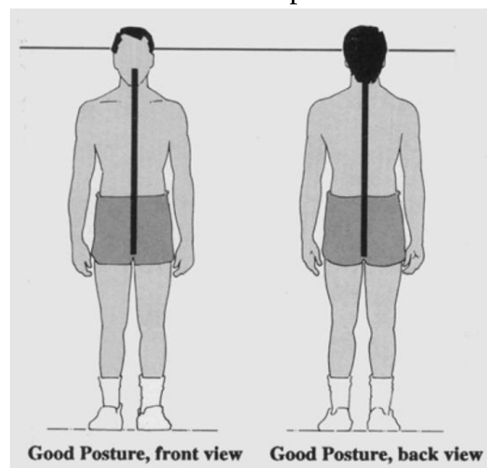


Figure 1. Standing Posture [25]

3.2 Maintaining Proper Posture

Maintenance of proper posture relies on proper positioning of head, shoulders, spine, hips and legs. The proper alignment of these body parts with respect to each other is key for attaining proper posture. The most relevant posture is involved with standing up and sitting, as such it is the focus of this discussion. Tables 1 and 2 provide a summary of requirements to maintain proper posture during standing and sitting respectively, and outline how the best possible posture may be achieved.

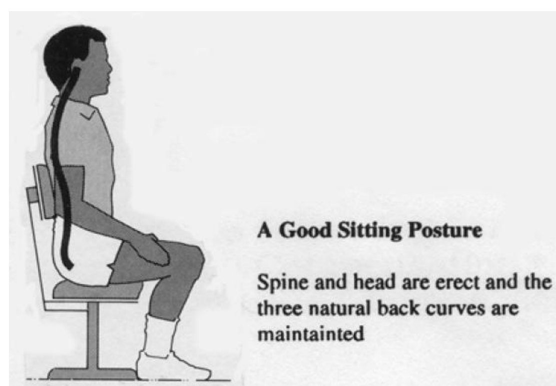


Figure 2. Proper Sitting Posture

Table 1. Proper Standing Posture Methods [25]

Proper Standing Posture
• Feet shoulder width apart
• Thighs elongated
• Knees unlocked
• Lean slightly back to keep straight
• Lift breastbone slightly
• Chin level parallel to floor

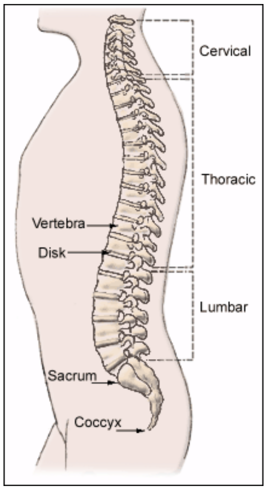


Figure 3. Spinal Curvatures [27]

Evidently, proper standing posture adjusts the body such that the curvature requirements are met. Sitting posture is quite similar but requires specific adjustment of the legs.

Table 2. Proper Sitting Posture Methods

Proper Sitting Posture
• Feet resting on floor
• Hips and knees resting at 90 degrees
• Slight arch in the lower back
• Lift breastbone slightly
• Chin level parallel to floor

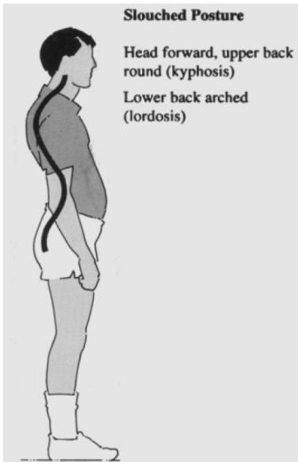


Figure 4. Improper Standing Posture

Through trunk tilt monitoring, it will be possible to detect improper posture based on these criteria. The patient will be instructed to assume proper posture, after which the monitoring system will detect in which directions or tilt angles the patient is moving, so that posture may be corrected accordingly.

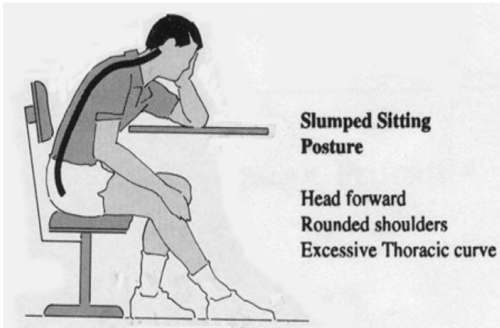


Figure 5. Improper Sitting Posture

4 Basic Design

4.1 Layout of Apparatus

Design of the tilt sensing apparatus involves using an accelerometer interfaced with a microcontroller, which is then interfaced with a feedback system. The basic concept of the apparatus is shown in Figure 6. The user is responsible for placing the posture correction system on their body and then assuming proper posture as outlined in this documentation or provided by a health professional. The system is responsible for tracking the user's tilt, and outputting appropriate feedback to notify the user of improper posture. The user is then responsible to adjust their posture based on the information provided from the feedback actuators of the posture correction system.

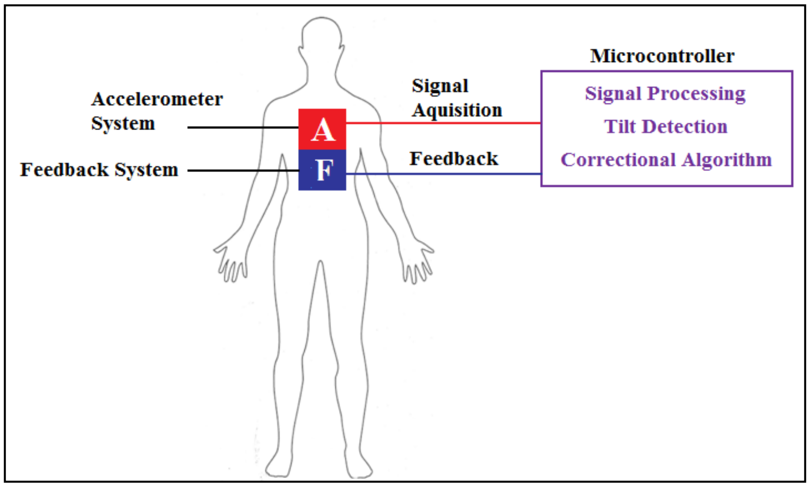


Figure 6. Apparatus Concept

The accelerometer, microcontroller, and feedback system are all placed on the user's chest, and are locally powered by batteries. As such, the entire system is wearable and able to be taken outside the lab environment. The accelerometer is attached such that there is maximal tilt sensing when the subject is moving, preferably at the largest possible distance from the hips. The system is packaged such that the user may adjust the location of all the components with easy. The microcontroller interprets the voltage signals from the accelerometer via an analogue to digital converter, and calculates relevant tilt angles, as per Appendix A. Once the tilt angles are known, a posture correction algorithm compares the tilt angles with hardcoded thresholds, and determines appropriate actions to be carried out via the feedback system. The feedback system consist of small DC motors which vibrate to notify the user of a change in posture.

4.2 Accelerometer Interfacing

In order to detect tilt and thus posture, it is necessary to use a three-axis accelerometer and all three of its axes. As well, it is necessary to supply the appropriate power and ground connections. Figure 7 outlines the basic connections required for accelerometer interfacing.

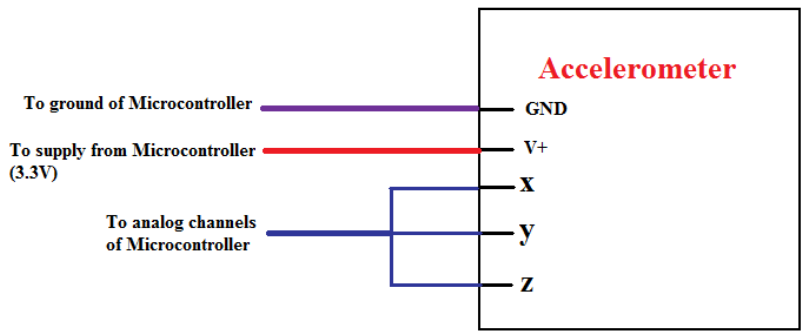


Figure 7. Accelerometer Interfacing

All the connections from the accelerometer go to the microcontroller. The microcontroller has built in modules to allow for processing of analog signals and can provide a constant power supply to the accelerometer. In the event that the microcontroller cannot provide adequate power, a separate battery with a voltage regulator may be used to power the accelerometer, and this is shown in Figure 8. The

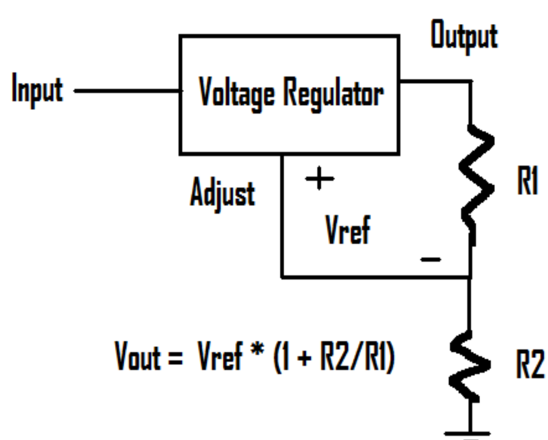


Figure 8. Voltage Regulator

voltage is scaled down through resistors R1 and R2. This voltage regulator may be used in conjunction with a battery to supply power to any portions of the device. However, it is not used in the project due to redundancy.

4.3 Microcontroller

The microcontroller receives signals from the accelerometer and outputs digital signals to actuators in the feedback network. Figure 9 shows the layout and pins of the microcontroller. Here 4 output pins drive the motors, and 1 output pins drive LEDs.

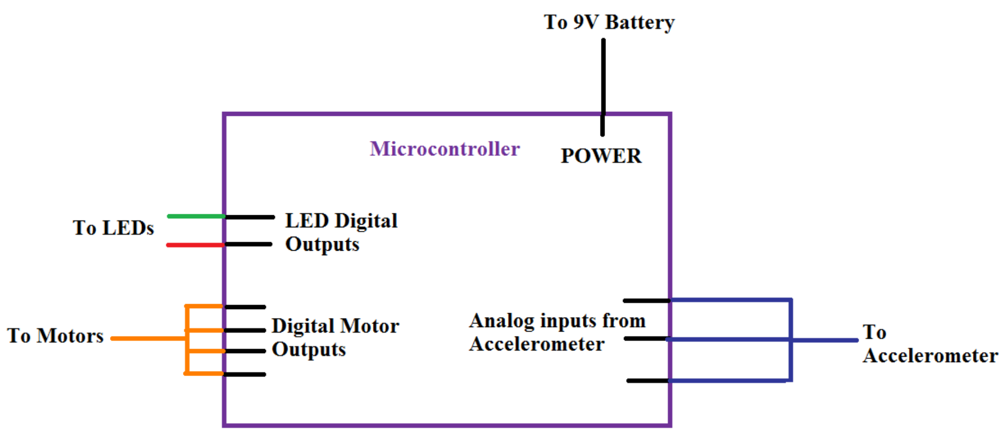


Figure 9. Microcontroller Interfacing

The microcontroller has its own ground and multiple power output pins as well (3.3V, 5V) for the particular model in use (see detailed implementation). Four motors are connected to give information about left, right, forward, backward tilt sensing. 1 LED will keep track of whether posture is inappropriate (red). Motors corresponding to the particular degree of tilt will vibrate and inform the user that they are tilting too much in a particular direction and posture needs to be re-adjusted. The LED is a way for observers to see if posture is being maintained correctly or not based on input from the accelerometer. The microcontroller has been tested and verified that it can provide real-time information in regards to posture changes in seated and standing positions (which require at least 100 ms to change). The microcontroller comes with a specialized electronics perforation board, that allows the accelerometer and actuators to be directly connected above it, such that the entire system is housed on one local circuit. This is advantageous as mounting the device involves mounting only one package, as opposed to mounting the accelerometer, actuators and microcontroller separately. Power will be supplied to the microcontroller via a 9V battery and will be housed in the same packaging.

4.4 Motor Actuators and LEDs

The microcontroller will supply feedback information via LED and motor actuators. The LED will be attached directly to the microcontroller, as they do not require large current draws. The motors will be powered via buffer circuit and Darlington Drivers from a 9V source, as to avoid damage to the microprocessor, should there be a large current surge. The motors will be placed on the user's chest, above, below, to the left and to the right of the accelerometer system. These correspond to backward, forward, left and right tilting based on the motor outputs.

LED will use a simple circuit that powers it from a source through a sufficient resistance; this regulates the intensity of the LED. The circuit used is depicted in Figure 10. A red LED will correspond to incorrect posture. The LEDs will be attached to the special perf-board circuit housed on top of the microcontroller.

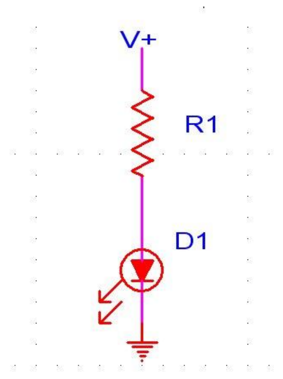


Figure 10. Simple LED Circuit

Motors will be treated slightly differently, and powered via buffer circuits and Darlington Drivers. The microcontroller will output a voltage to cause the Darlington Drivers to connect force appropriate voltage across the motors; the voltage may be scaled down from 9V via voltage dividers and then buffered. The buffer will use an OP-AMP to use this same voltage to drive the motors.

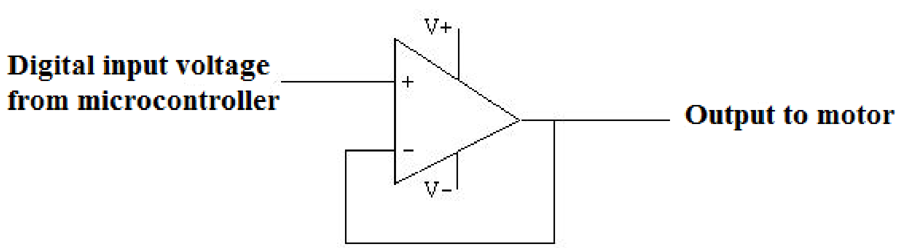


Figure 11. Buffer For Driving Motor

4.5 Wireless Module and Imaging

As this is a joint project, the visualization is handled by Nanxi Zha. She will be wirelessly transmitting the information from the processor regarding the angles to a computer. Using labview, she will use the angle information to determine tilt, and then output this tilt data via LabView and a 3D visualization scheme. In this way, it will be possible to visually track the motions of the user in real-time, as they are moving and tilting. The wireless module will be connected to the microcontroller within the accelerometer set-up, and will communicate with a second microcontroller connected to a remote laptop. The timing required for this operation has been verified and is sufficient to track tilt in 3D.

5 Detailed Implementation

5.1 Parts List

The following is a summary of the parts required for the full implementation of this project, as well as a brief description of operation and connection procedures.

Accelerometer

The accelerometer used is the ADXL335 by Analog Devices, which come pre-soldered onto a breakout board chip. The axes are drawn onto the break-out board itself, and show the appropriate directions. The relevant pins are pins x,y,z which provide voltages related to acceleration, and GND and VCC pins for powering the accelerometer.

Figure 12. ADXL355

Microcontroller

The microcontroller used is the Arduino Duemillanove, which comes on a pre-built module with pin connections, power connections, and a USB interface. Three analog pins are used to received in the input analog voltage, and digital pins are used to output appropriate signals to drive the LEDs. The microcontroller also has a Protoshield attachment, which is a perforation board raising the pins of the microcontroller and allowing for custom circuitry to be implemented.

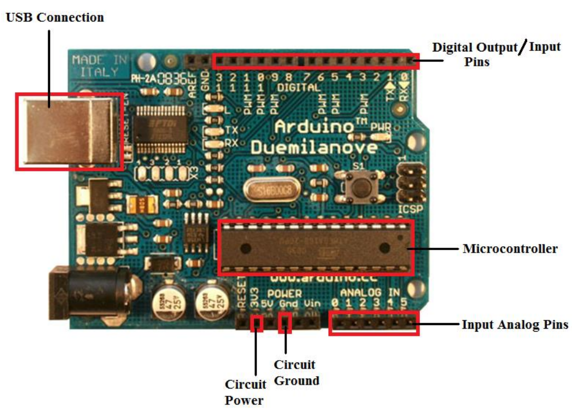


Figure 13. Arduino Microcontroller

Miniature Vibration Motors

Solarbotics VPM2 Miniature Disc Motors were used in order to supply sufficient feedback for this device. Four motors were obtained and connected in order to provide feedback about direction of tilt. Flexible, shielded electrical wire was soldered onto the ends of the motor wires, and then implement alongside the rest of the circuitry.

Figure 14. Vibration DC Motors

OP-AMP for Buffer Circuit

The OP-AMP used is a high precision OP-AMP, Burr-Brown (TI) OPA277P. By setting up the circuit as a buffer, any input on the positive pin will be reproduced on the output. The output is capable of maintain a constant voltage to drive a motor.

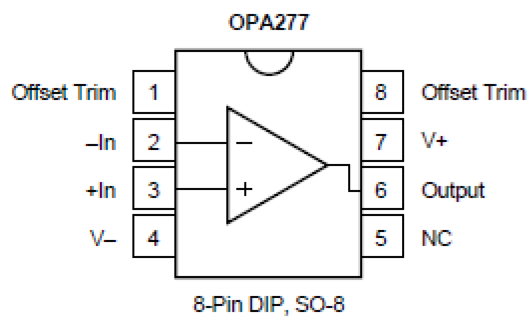


Figure 15. OP-AMP Pin Specification

Darlington Drivers

In order to drive the motors, they must have an on and off state, regulated by the microcontroller, hence, drivers based on transistors (as switching mechanisms, current amplifiers) may be used. A voltage output from the microcontroller may be used to switch motors on and off. The Toshiba TD62003AP Darlington Drivers were used. The numbered blocks on the left in the diagram represent the digital inputs, the blocks on the right are the ground of the motors, and the other end of the motor is connected to voltage high.

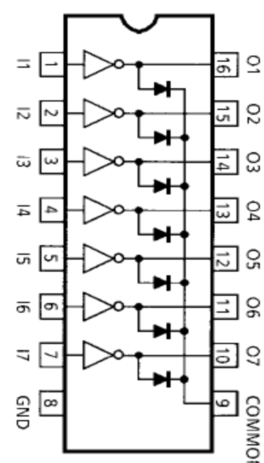


Figure 16. Darlington Drivers

Other Parts:

Other common parts were used, such as buttons, LEDs, wiring, resistors. These are generic types and do not require a tight specification. Wireless module and associated parts are handled by the other group member responsible for tilt imaging and software component.

5.2 Circuit Board Layout

The following is a block diagram of the circuit board representing the various components.

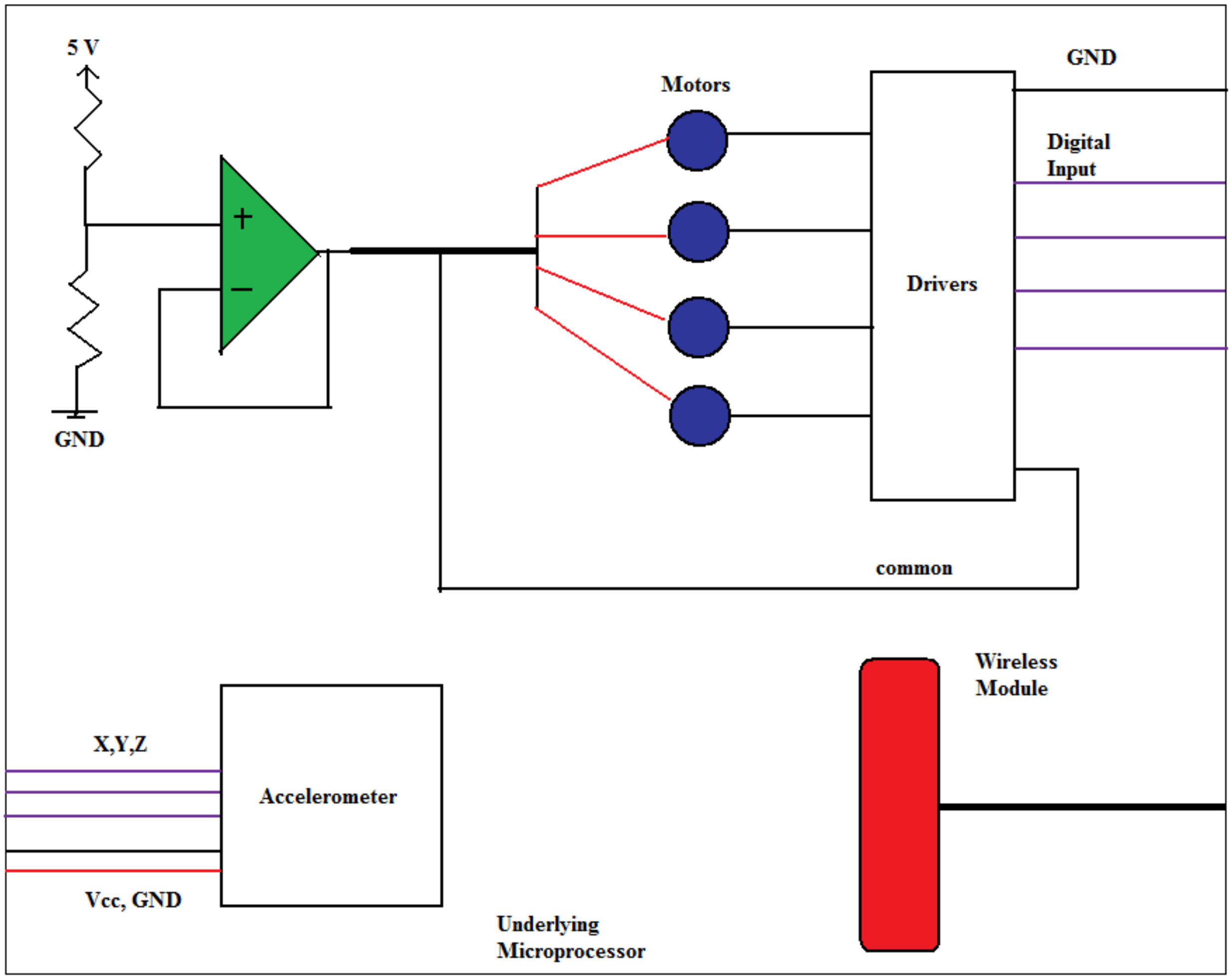


Figure 17. Circuit Board Layout

The perforation board sits on top of the microcontroller, and the devices are soldered onto it. The accelerometer is soldered perfectly flat onto the board, and its x,y,z inputs go into the analog pins of the microcontroller (x,y,z corresponds to pints 3, 2, 1 respectively). Then power is supplied to the accelerometer from the 3.3V out on the microcontroller.

The OP-AMP serves as a buffer, and the 5V supplied by the microcontroller is scaled down further to 2.5V, in order to supply the motors.

The motor positive wire is connected to the OP-AMP output, and the negative wire is connected to the output of the driver. The driver is connect to the output of the OP-AMP through a common pin, and digital inputs control the operation of the driver. When a digital pin is high, the driver will connect the negative end of the motor to ground, thus causing 2.5V to be across the motor.

The wireless module communicates accelerometer outputs in real time to a remote laptop which displays a 3D interpretation of the tilt. The figure below shows the implementation of the design on a circuit board.

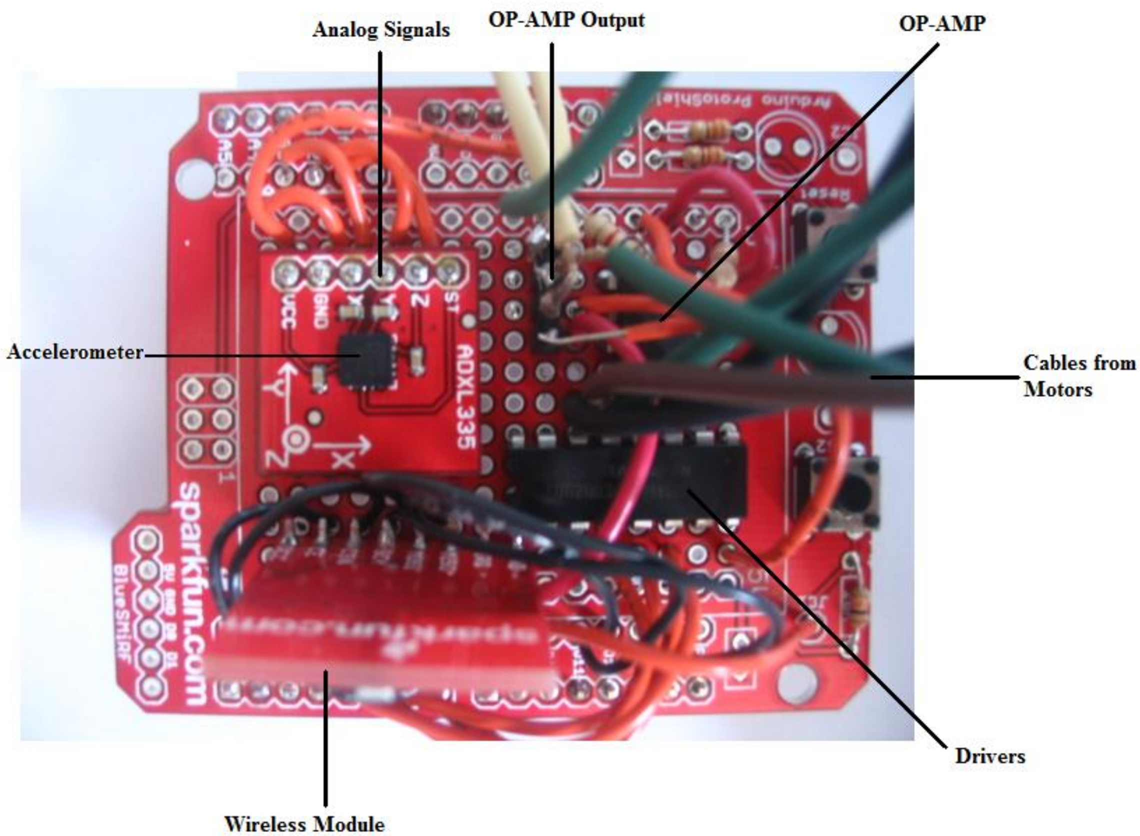


Figure 18. Circuit Board

This is fitted onto the microcontroller with special headers, and the entire assembly is what is placed on the user's chest. Power is supplied via VIN pin, which can take a 9V source and scales it down to 5V via a built in voltage regulator.

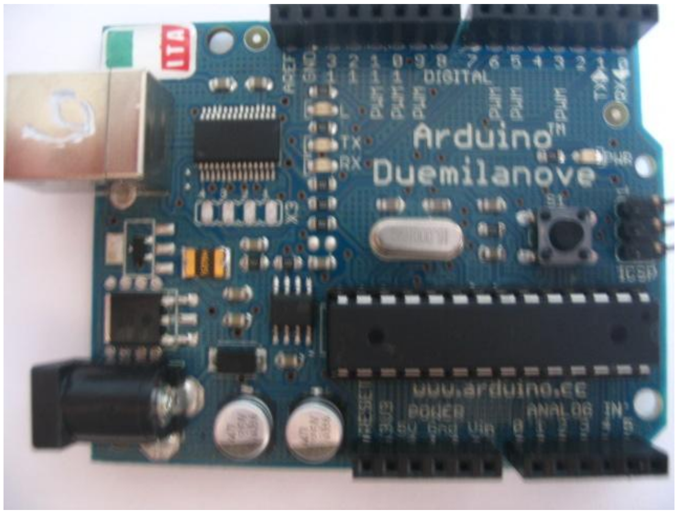


Figure 19. Arduino Board

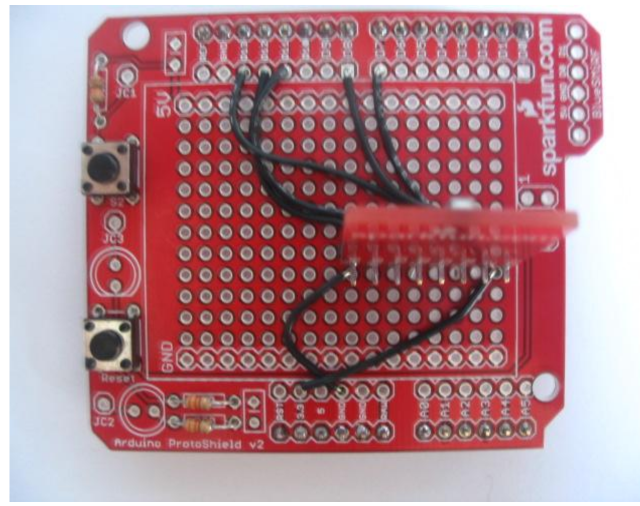


Figure 20. Wireless Module on Separate Board

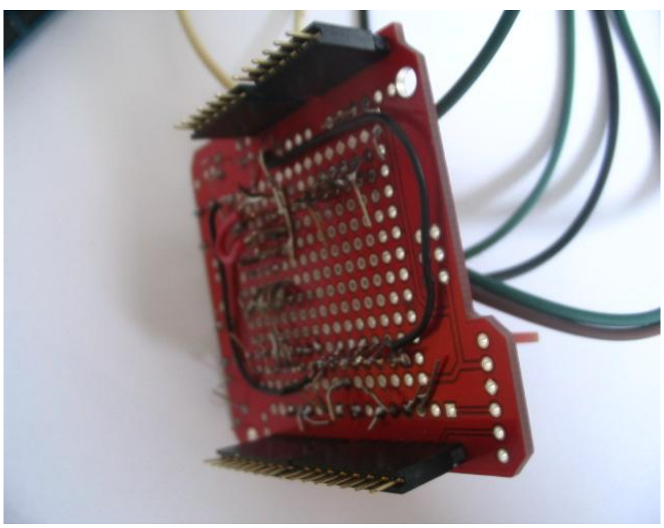


Figure 21. Connection Headers

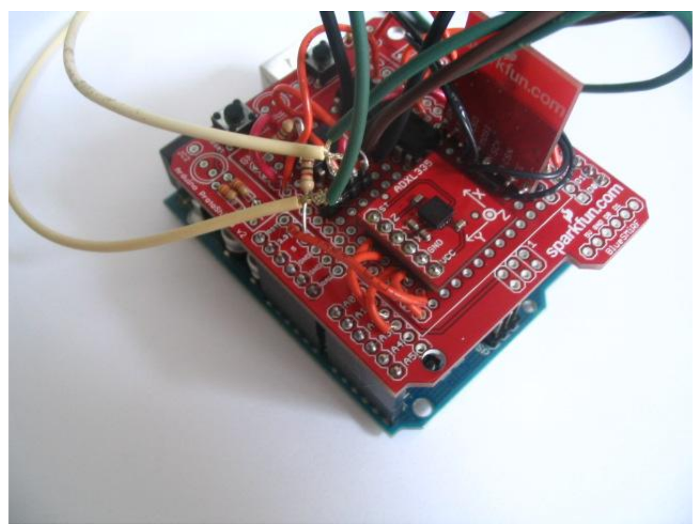


Figure 22. Board sitting on top of Arduino

5.3 Accelerometry and Tilt Sensing

It is possible to track tilt using 1 axis, 2 axes, or 3 axes, detailed descriptions of these methods are presented in Appendix A. This project concerns itself with implementation of tilt in three dimensions. The device does not measure rotation however, so it is assumed the subject is not rotating at the hips and posture regulation is limited to correction of the spine.

The basic principles of tilt sensing in three dimensions is finding 3 angles, which describe e the tilt with respect to various axes and planes. The block show to the right is the accelerometer on a three dimensional coordinate system. The accelerometer has its own axes, which are labeled X Y and Z. The accelerometer outputs a signal proportional to the acceleration experience along an axis. This is measured based on a force on a spring, so then the interpretation of acceleration in the static case is the gravity vector being split up into components along the axes of the accelerometer.

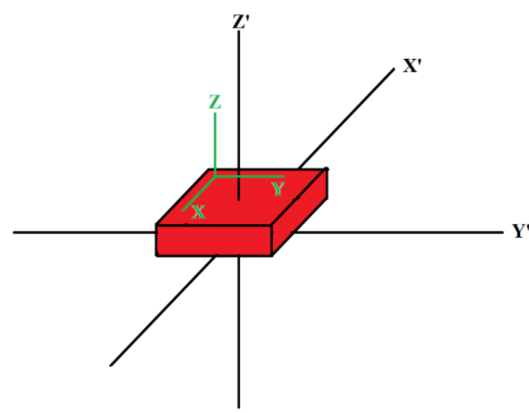


Figure 23. Tilt Sensing

To find tilt, we calculate the acceleration along each axis, and then using trigonometry determine the angles required to produce such accelerations along the various axes. To find the acceleration along a single axis, the following equation is used:

$$a = \frac{V \quad - V_{off}}{\boldsymbol{S}}$$

Once this is known, we are interested in 3 angles; the angle between the Z’ axis and the Z axis, the angle between the X axis and the X’, Y’ plane, and the angle between the Y axis and the X’,Y’ plane. Using trigonometry, it is possible to show that these work out to:

$$\delta = tan^{-1} \frac{\sqrt{a_X^2 + a_Y^2}}{a_Z}$$

$$\alpha = tan^{-1} \frac{a_X}{\sqrt{a_Z^2 + a_Y^2}}$$

$$\beta = tan^{-1} \frac{a_Y}{\sqrt{a_Z^2 + a_X^2}}$$

Where δ is angle between Z and Z', α is between X and X',Y' plane, and β is between Y and X', Y plane. Complete derivation and more details are available in Appendix A.

5.4 Implementation with the Microcontroller

In order to implement with a microcontroller, the following algorithm was used:

Arduino Posture Correction Algorithm

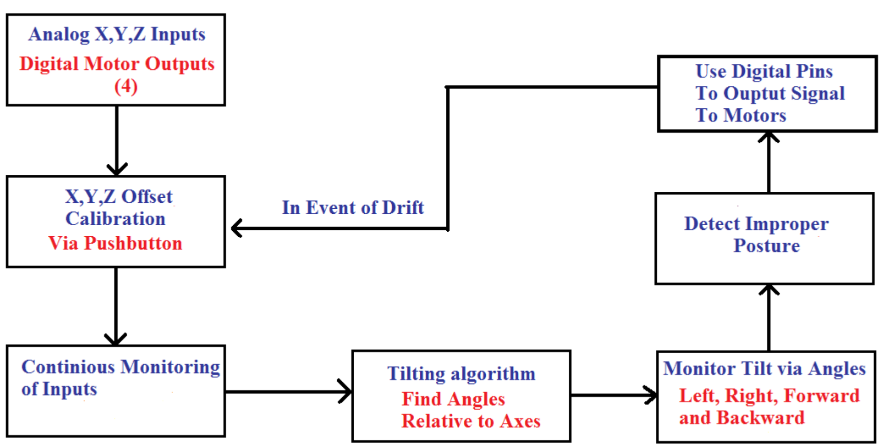


Figure 24. Microcontroller flow chart

Full details of implementation are given in Appendix B. Basic signal acquisition is done by using the microcontroller's analog input pins. A calibration button is included to perform a calibration algorithm which obtains the offsets of the axes based on an averaging scheme. The input is then modified and converted to usable accelerations along each axis. A tilting algorithm is used to determine the angles, as discussed in section 5.3 and Appendix A. The tilt angles determine whether the user is tilting beyond a set threshold, and in what direction. This signifies detection of improper posture, and digital pins are used to output signals to motors to notify user of incorrect posture. Finally, in event of sensor drift, the device may be recalibrated via the calibrate button.

5.5 Packaging

Packaging of the device involved taking the device from figure 22 and placing it inside a plastic box with a cover. Holes are made on the sides of the device to run the motors out to plastic pads/holders, which house the motors. The motors are placed on sponge in order to accommodate for people of different sizes. This method allows to place the motors snug against the body so that vibration is felt. An LED is placed on top of the box and lights whenever posture is incorrect.

In order to anchor the device on the body, it is necessary that it is vertical and tight against the body. In order to this, straps were attached via Velcro to the box and a wearable harness was made to fit people of various sizes. The harness system is shown in figure 26.

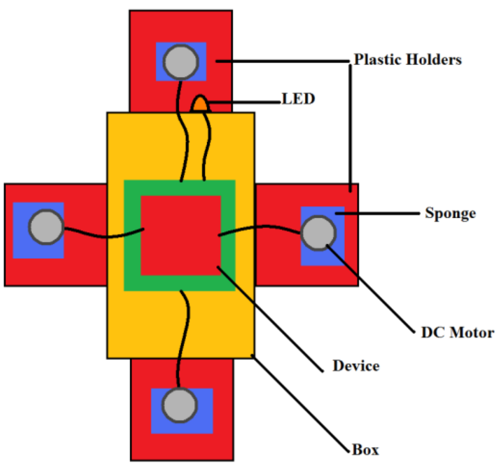


Figure 26. Housing for Device

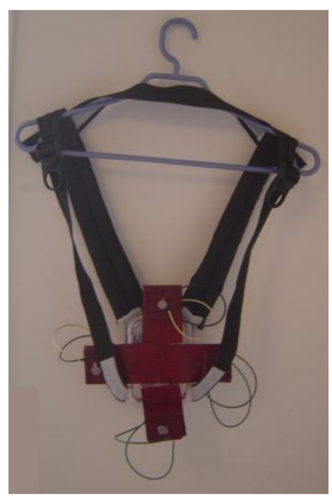


Figure 25. Photo of Harness and Encasing

5.6 Operating Instructions

1. Anchor device on body using harness, the device should end up on chest, with device x-axis pointing up
2. Motors should be snug against the chest
3. Stand or sit with appropriate posture
4. Press the calibrate button, which is a built in button on the right side if facing the device
5. If tilting generates angles above certain thresholds, the LED will light up and the motor corresponding to the direction of tilt will begin to vibrate
6. Return to proper posture position, and the motors will cease to vibrate and LED will turn off

6 Testing of Apparatus

6.1 Preliminary Testing

Preliminary testing was first done with the accelerometer itself. The accelerometer was anchored to a rotating board with finite angle measurements. Typical signals obtained for a controlled rotation are shown.

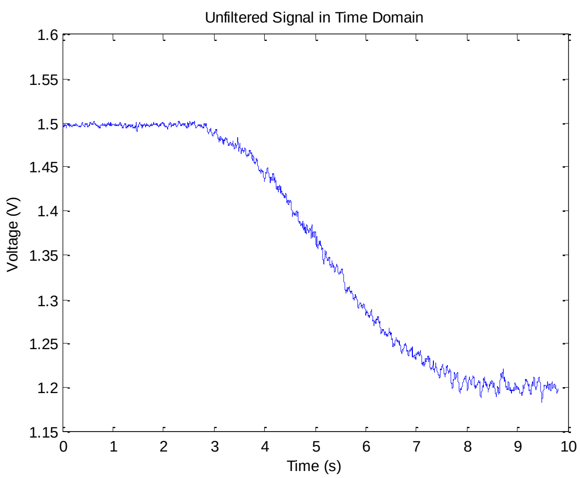


Figure 28. Signal from rotations

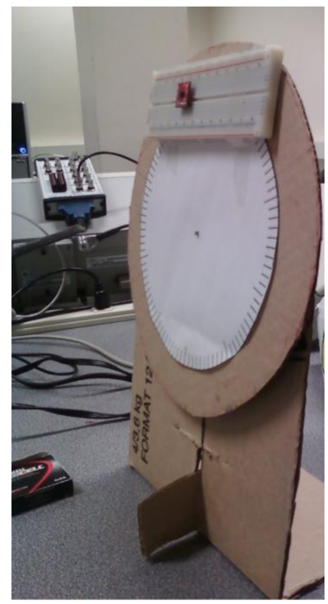


Figure 27. Early Testing Set-Up

Using a single axis tilt algorithm, it was possible to draw out a controlled tilt path, eventually this was extended to 3 angles and 3D tilt sensing. In 3D tilt sensing, the angles changed in a similar manner, going from 0 to a maximum of +/-90 degrees. The angle paths are similar to figure 28, and simultaneous tracking of angles allows to gage the overall tilt of the spine.

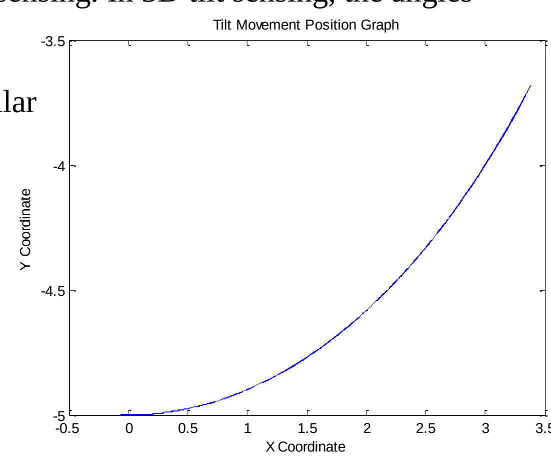


Figure 29. Tilt Path for 90 Degrees

Upon constructing the final device, it was important to test how well the theoretical angles matched with the actual angles, once the device was in use and being handled on the macro scale. To do this, controlled angle experiments were performed and percentage errors across various tilt angles were done. It was found that the error in the angle is approximately 10% in the worst case. However, since the tolerance on “proper posture” is fairly large, this error is not a major concern for posture correction.

Table 3. Tilt angle errors

Theoretical Angle (degrees)	Actual Angle (degrees)	% Error
-90	-89.04	1.06667
-60	-56.76	5.4
-30	-26.96	10.1333
0	4.27	N/A*
30	28.95	3.5
60	66.03	10.05
90	86.98	3.35556

*N/A due to division by the theoretical value, which is zero.

6.2 Subject Testing

Several testing subjects with different physical attributes were investigated, and the harness and device was fitted successfully. It was found that tilt sensing was accurate between the subjects, and they all had to tilt in relatively the same manner. The device was able to keep calibrated after continuous monitoring for over 30 minutes, and the motor operation was reproducible in most cases. Subjects could detect motors and distinguish between the directions depending on their location.

It was not possible to do extensive testing or clinical studies due to time constraints, but from the limited subject testing it was found that this system performed well to correct posture, and if miniaturized significantly, will be a non-intrusive method of daily tilt monitoring.

It was found that the wireless transmission links were working, and that tilt could be monitored in real-time on the laptop computer as the subject was moving around. This is beneficial if a physician is interested in the tilt patterns of patients and serves as a precise form of visualization of tilt.

Figure 31. Laptop and Receiver

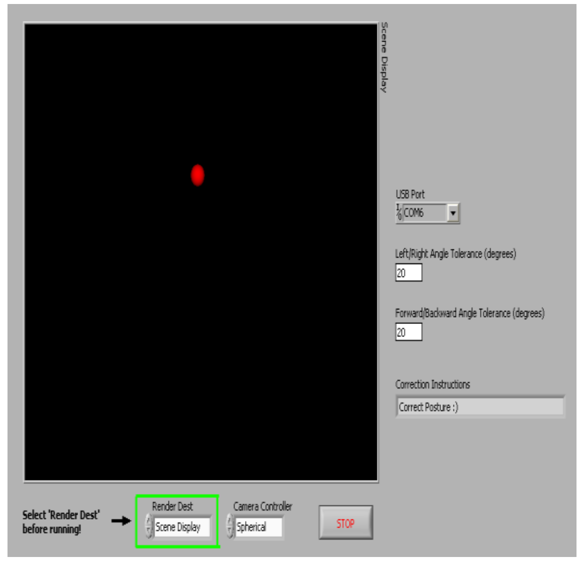


Figure 30. 3D Tilt Visualization

Figure 32. Device With Harness

7 Results and Discussion

7.1 Discussion of results

It was found that tilt sensing using accelerometers is a cheap and effective way of monitoring posture and feedback mechanisms such as motors are very effective. The LED is an effective way of letting both the subject and anyone monitoring the subject know if posture is incorrect.

The error on the tilt angles was found to be minor, and did not affect the general operation of the device. However, longer testing periods of an hour or more are required to give a truly accurate representation of the robustness of operation of the device. The successful operation for 30 minutes and the fact that the device may be easily re-calibrated makes this a sufficiently robust system for posture correction.

The system was able to accommodate subjects of various sizes. This is due to the adjustable harness. As well, posture correction accomplished by the device was very similar between subjects, and did not vary significantly due to differences in tilting.

Motor vibration served as an effective feedback mechanism. As subjects tilted, the feedback from the motors not only gave information about incorrect posture, but indicated the direction which was being deviated. As such, it was natural for the subjects to correct their posture in the opposite direction.

7.2 Discussion of Shortcomings

The first prototype has several shortcomings in terms of size and accuracy. Though the system is capable of providing posture correction feedback sufficiently well, it is quite difficult to quantify it exactly and the only way to control the tilt algorithm is by setting angle thresholds.

The size of the apparatus is quite large, and cumbersome to keep on the body during regular activity. It is inevitable that such a system is best miniaturized, and possibly best integrated into clothing. As well, a better method of keeping the motors close to the body would allow for better feedback.

Finally, discrepancies between male and female subjects may occur due to different anatomy, as the device is being positioned on the chest. To avoid this, a different location, such as the back, may be chosen upon miniaturization. The drawback is that it is more difficult for the user to mount the device on themselves. Alternatively, the motors may be brought out to the extremities of the body where anatomical features of males and females are the same.

8 Conclusion and Future Work

A robust and practical posture correction device has been constructed using accelerometer techniques. The system tracks tilt angles associated with the spine, and notifies the user of incorrect posture. It was found that the accuracy of the system is sufficient to monitor posture, and the motors are an effective and way to provide feedback without irritating the subject. Use of the Arduino Duemillanove has allowed for a simple integration of all the hardware on a single, portable platform, and is expandable to accommodate for future work.

The current developed technology allows for pre-programmed tilt posture correction based on set algorithms. However, to expand this system, it is possible to include input from the user, as to how posture is to be tracked. A preliminary application of this is letting the user set threshold angles for tilt in various directions. Furthermore, being a MEMS device, this particular model of accelerometer may cause errors due to improper mounting. Improvements in packaging and calibration may resolve these issues. Finally, the system currently uses only 1 accelerometer and track the orientation of the spine. Future work may include accelerometers placed at various extremities and joints, to fully monitor body posture. Furthermore, simultaneous data collection from all these accelerometers will give rise to new algorithms for posture detection and correction.

APPENDIX A: Accelerometer Tilt Sensing

A.1 Inside the accelerometer

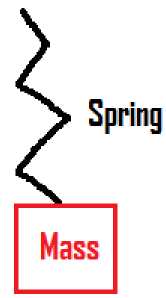


Figure 1. Spring mass system

The fundamental working principal behind the accelerometer is analogous to the mass-spring system. The spring is attached to the mass which is suspended vertically, while the other end of the spring is held fixed. The spring exerts a force on the mass equal and opposite to force of gravity in order to keep the mass in equilibrium. Drawing a free body diagram and including forces from gravity and the spring, we may write an equilibrium expression:

$$ma = -kx + mg$$

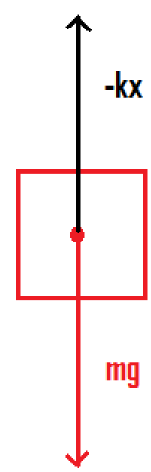


Figure 2. Conceptual free body diagram

Where **k** is the spring constant, **x** is the stretch in the spring (positive values indicate the spring is lengthened), **m** is the object mass, **g** is acceleration due to gravity, and **a** is the acceleration of the system. Note that that positive acceleration is defined as downward. –**kx** implied the spring exerts a force opposite to the direction it is stretched, implying that it attempts to compensate for gravity such that acceleration is zero in equilibrium.

In dynamic problems, one would orient the accelerometer such that the force of gravity is orthogonal to the sensing axis, so that there is zero acceleration in the sensing direction at rest; this is to avoid angular offset errors. The same concept is applied in static problems to ensure the greatest sensitivity for an angular range (more on this later).

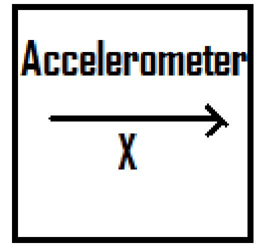


Figure 3. 0g Configuration

The accelerometer is a device that outputs a voltage on each of its pins corresponding to the acceleration experienced in a particular direction. For example, consider the accelerometer to the right. This is a uniaxial accelerometer with direction X as shown. The device will have a pin producing a voltage from the internal components that corresponds to acceleration in direction X. Let's assume that the accelerometer to the right is stationary, and the force of gravity is acting downward. Then, since the force is orthogonal to the sensing axis, we expect the output to correspond to a case where the acceleration is 0. Note that the voltage itself may not be zero and thus these devices may require external calibration based on certain criteria. Now let's consider a case where the accelerometer is oriented in a different manner, and the sensing axis is oriented downward.

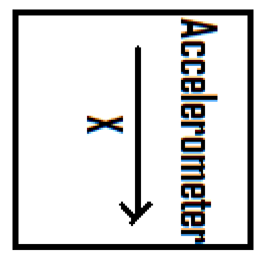


Figure 4. +1g Configuration

In this scenario, the force of gravity will be directly acting on the sensing axis, and we expect a voltage output to correspond to +g. If the sensing axis was pointing up we'd expect an output corresponding to –g.

In order to determine the expected voltage, the device has a parameter called sensitivity **(S)**. This is usually denoted as **mV/g**, where **g** is the acceleration due to gravity. So if we multiply **S** by the acceleration, we can get an expected voltage reading for +g and –g configurations discussed above. Note that the output voltage will be of the form,

$$V_{out} = V_{off} + S * g$$

where the above equation is for the +g configuration. There is an offset voltage present in all such measurements, and care should be taken to compensate for this. In general acceleration from the accelerometer may be found via:

$$a = \frac{V \quad - V_{off}}{\boldsymbol{S}}$$

A.2 Basic compensation for offset errors

When designing a tilt system that is going to rely on some sort of standard start position, with respect to which all other angles will be measured, it is important to compensate for offset errors. Once the “start” position of the device is known, the voltage at that position is measured to be the “offset voltage”. Using this voltage and by comparison with other voltages output from the accelerometer, we may deduce the angle of tilt. However, even though we expect our offset voltage to remain consistent, this is in fact not true due to errors arising from mounting the assembly, temperature, and even aging. Hence, it is necessary to implement some sort of adaptive auto-zero technique so that the measurement results are consistent and reproducible. It is obvious that if the offset voltage is different from the assumed offset voltage, then there will be an error in the overall voltage:

$$V_{out} + \Delta V_{out} = V_{off} + \Delta V_{off} + S * a$$

It is unlikely that this offset voltage is known, and when calculations are performed it will affect the tilt measurement. Hence, the only sure way of eliminating such offsets is by calibrating the device each time it is used under different circumstances or after prolonged periods of time. To perform such a calibration, the 0g voltage output level is required, as it the location of greatest linearity and minimal error for the accelerometer. There are several methods which may be used to accomplish this. The most prominent method is to rotate the accelerometer in one axis, from +1g to -1g. Assuming the sensitivity is linear, we may take readings at these configurations and subtract them, then dividing by 2 yields the actual sensitivity of the device. This is because sensitivity is given in units of mV/g, and rotating from -1g to +1g covers a range of 2g; dividing the difference by 2 yields voltage expectation per 1 g. Then to find the 0g or “offset” voltage, we may subtract the sensitivity from the +1g or -1g voltages. Considering the fact that this must be done for all 3 axis, this may become a nuisance. Hence, a simpler and more practical method of implementation would be to simply place the device in a configuration such that 2 of the axis are in the 0g configuration, and 1 axis is in a + or –

1g configuration. Then, the two 0g axis may be auto-calibrated to their output voltage, and to find the 0g voltage of the remaining axis, we may add or subtract the sensitivity for that axis (depending on +/- g configuration).

A.5 Tilt Measurement using 1 Axis

When the accelerometer is configured in orientation other than 0g, +1g or -1g, it becomes slightly less trivial to determine the anticipate voltage reading. Let's take a scenario where the accelerometer is a uniaxial device placed on a ramp inclined at an angle θ, show to the right. A good way to conceptualize the sensing axis X is being a spring oriented in the X direction, fixed to experience forces only parallel to the sensing axis. In actuality, the voltage produced by such devices is through a changing resistance or capacitance that is regulated by the stretch of these springs.

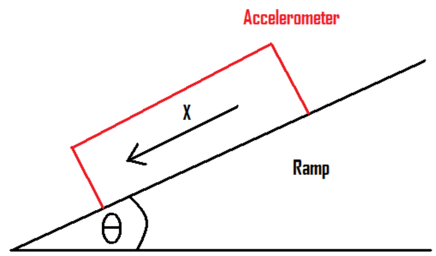


Figure 5. Accelerometer inclined at an angle θ

As before, we draw a free body diagram to illustrate the scenario. In this way it will be possible to determine the forces acting in the X direction. Note that this force will be some fraction of the g force.

In **Figure 5** we use observe that the required force is Fx and it is directed parallel to the ramp and in the direction of the sensing axis. Using trigonometry we obtain the following relationship:

$$a_x = g * \sin(\theta)$$

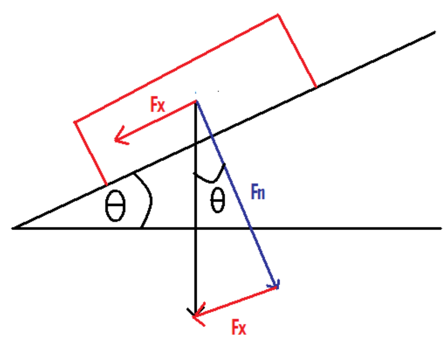


Figure 5. Free body diagram of titled set-up

Now we can write an entire expression for the expected voltage.

$$V_x = V_{off} + S * \sin(\theta) * g\ mV$$

Since S is in units of mV/g, we can omit g from the equation so that we are simply left with,

$$V_x = V_{off} + \boldsymbol{S} * \sin(\theta)\ mV$$

where **S** is just the nominal value of the sensitivity.

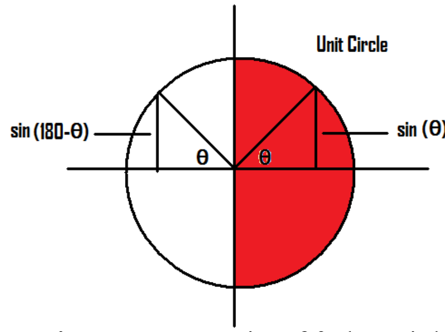


Figure 5. Domain of θ that yields unique results

Naturally, for tilt sensing applications we would like to isolate θ in order to measure the angle of inclination and thus determine tilt, rearranging the preceding equation yields,

$$\theta = sin^{-1}\left(\frac{V_x - V_{off}}{\boldsymbol{S}}\right)$$

There exist some physical limitations here. The range of allowable angles is between -90 and 90 degrees. This is because it will not be possible to discern angles of >90 degrees seeing as sin(θ) = sin(180- θ), similar conditions hold for when the angle is <-90 degrees.

A.4 Tilt Measurement using 2 Axes

The single axis tilt sensing is an effective way to get quick tilt measurements but has a few drawbacks. The greatest drawback is the limit on the angles which can be measured; it only allows measurement of a 180 degree range. The second drawback is the difference in sensitivity at various angles. For example, if the accelerometer is in the **0g** for the x-axis, tilting by one degree causes an acceleration of 17.45 mg (mili-‘g’); here we have considered sin(1)-sin(0) as a means to investigate sensitivity (the sin of the angle is exactly the acceleration, as shown before). On the other hand, if we are going from 89 to 90 degrees, the change in acceleration required for a 1 degree change is 0.15 mg. Hence if our accelerometer experiences errors, these errors will be more pronounced in certain angle ranges (namely approaching 90). Ideally, it will be favorable to have constant tilt sensitivity, that is, equal acceleration experienced for a 1 degree tilt regardless of

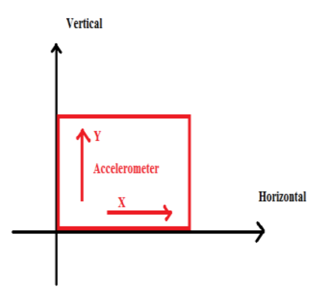


Figure 6. Orientation of accelerometer

orientation. In order to accomplish this, a 2 or 3 axis accelerometer may be used by using 2 of the axis.

The approach here is to orient the accelerometer in such a way that one of the axis (y, for example) is exactly parallel to the force of gravity, and the other axis is then constrained to being perpendicular and in the **0g** configuration (x in this case). The diagram below depicts this scenario:

Now, the accelerometer is tilted counterclockwise by an angle , and a free body diagram of the situation is drawn in figure 7. The gravity vector, **g**, is labelled in blue, and point downward. It can be split onto an orthogonal basis, which are the axis of the accelerometer, by projecting the vector onto the accelerometer axis.

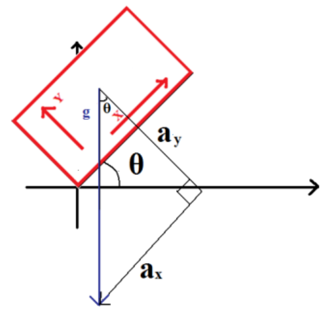


Figure 7. Accelerometer tilted counterclockwise by an angle free body diagram

This is shown by vectors $\mathbf{a_y}$ and $\mathbf{a_x}$**.** Since the only acceleration is due to gravity, the output voltages of the accelerometer will be indicative of the acceleration vectors along the axis of the accelerometer. Solving for these accelerations,

$$a_x = g * \sin(\theta)$$

$$a_y = g * \cos(\theta)$$

Now, the expected voltages from the accelerometer may be expressed as,

$$V_x = V_{offx} + \boldsymbol{S} * \sin(\theta) \ mV$$

$$V_y = V_{offy} + \boldsymbol{S} * \cos(\theta) \ mV$$

To solve for the angle, we apply the fact that,

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)}$$

In order to use this result, we rearrange the previous voltage equations as follows,

$$V_x - V_{offx} = \boldsymbol{S_x} * \sin(\theta)\ mV$$

$$V_y - V_{offy} = \boldsymbol{S_y} * \cos(\theta)\ mV$$

Now dividing the top equation by the bottom equations we have,

$$\frac{V_x - V_{offx}}{V_y - V_{offy}} = \frac{\boldsymbol{S_x}}{\boldsymbol{S_y}}\tan(\theta)$$

Finally we find theta by using the following equation,

$$\theta = tan^{-1}(\left(\frac{V_x - V_{offx}}{V_y - V_{offy}}\right) * \frac{\boldsymbol{S_y}}{\boldsymbol{S_x}})$$

By using both cos and sin dependence, we have effectively caused a more uniform and substantial sensitivity range. This is because when sin is at its lowest sensitivity (at 90 degrees) cos is at its highest (0 degrees), and vice versa. Hence this method allows reducing sensitivity errors. Additionally, we are now in a position to investigate 360 degrees of tilt. The following is a diagram which shows what occurs when we tilt the accelerometer counterclockwise through 360 degrees, where the angle is measured from the +horizontal axis to x-axis edge of the accelerometer,

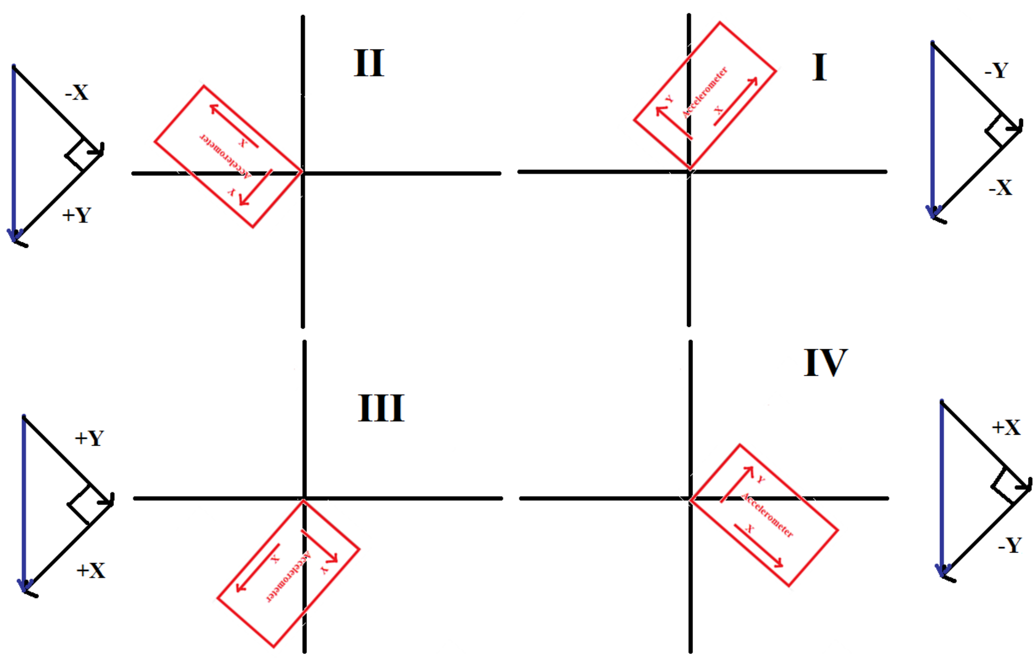


Figure 8. Accelerometer rotated through 360 degrees

The roman numerals represent the angular quadrants (ie. I is between 0-90, II 90-180 etc). Also, the vector sum diagrams accompanying each quadrant represent the vectors which sum to the gravity vector in each case. Assuming X and Y are positive values, then these vector diagrams show the signs of the acceleration in each quadrant based on the orientation of x and y axis on the accelerometer. For example, in quadrant 1, the first vector in the vector diagram is parallel to the Y vector on the accelerometer, and is pointing in the opposite direction as Y, hence we can expect a negative acceleration in the Y direction. The same is true for the second vector and the X acceleration. So then, given the signs of the accelerations, we may know which quadrant the angle is in, but there remains the problem of finding the exact angle, and for this we consider the tan function. The tan function is unique between -90 and +90, and then the waveform repeats for other angles outside this range. Hence an inverse of a certain value may mean several different angles. We discriminate against this by knowing the signs of the X and Y acceleration, but we must determine how to obtain an angle in the right quadrant.

Assuming we only want to deal with positive angles, if the accelerometer is in the first quadrant configuration, then the tan inverse function will give the correct angle. Now we consider the second quadrant with angles between 90-180, we know that in such a situation, we are going to end up with angles between -90 to 0 according to the tan inverse function. Since we are expecting angles between 90 and 180, we must add 180 degrees. In quadrant III, angles are 180 to 270, and the tan inverse function produces angles between 0-90. Hence again we add 180 degrees. Finally in quadrant IV, we expect angles from inverse function to be

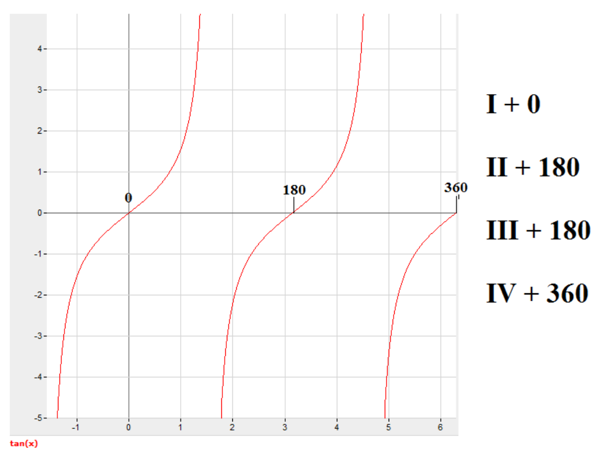


Figure 9. Tan function and adjustment for various quadrants

between 0 and -90, but we require angles between 270-360, hence we must add 360 degrees to obtain the appropriate angle. A summary of this is available in the diagram to the right.

A.5 Practical 3D tilt tracking application with 3-axis accelerometer

Using the theory, it is possible to implement tilt sensing for a simple block in space that the accelerometer is attached to. As discussed before, in order to track tilt along a single axis, it is best to use 2 axis as opposed to 1 due to the increased and uniform sensitivity as well as the greater range of angles that may be covered. To the right is a sample illustration how this may be done. The green axes on the red block are the accelerometer axes (X, Y, Z). The other axes (X', Y', Z') are our axes of interest with respect to which tilt will be measured. Since there are no forces acting parallel to the X',Y' plane, it is not possible to track tilt about the Z' axis. However, if we consider X, Z and Y, Z as pairs, it is possible to track angles about the Y' and X' axes respectively. The X,Z pair will give the angle between X and X', while the Y,Z pair will give the angle between Y and Y'. However a problem that quickly arises is that it is not possible to simultaneously track both these angles since the Z axis must rotate around only one of the coordinate axes (X', Y'). Therefore, it is necessary to move to a 3-axis solution that will simultaneously make use of all 3 axes. If we consider the X' Y' plane as "ground" in the above illustration, and Z' as an axis pointing in the direction of the force of gravity, then we may let the angles of interest be the angle between X and ground, Y and ground, and Z and the gravity vector (or the Z' axis). Here the angles will always range between 0 and 90 degrees regardless of the situation, and will give a complete description of tilt. Note that we cannot determine rotation about the vertical axis because force components of gravity projected on the axis are unaffected by this motion. Once again, we require

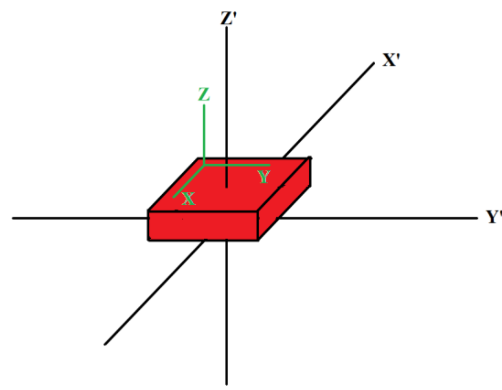


Figure 10. 3D tilt tracking

finding the projection of the gravity vector onto all three of the axes, and we begin by considering a series of diagrams where the block has been tilted,

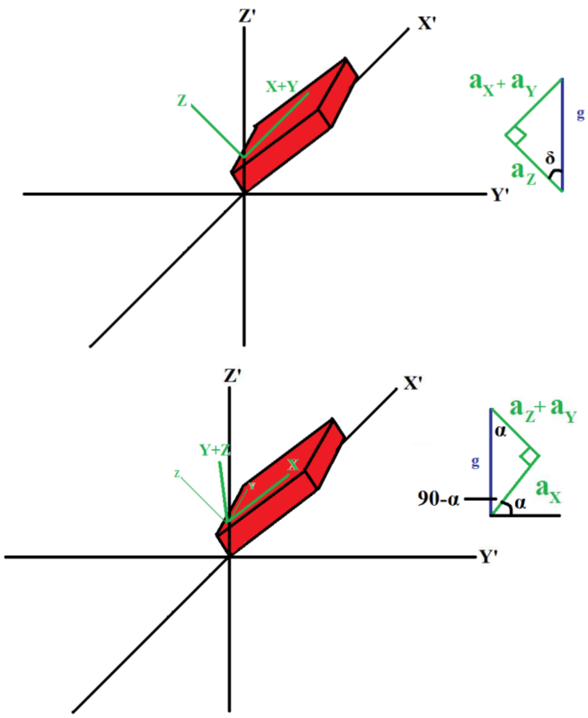


Figure 11. 3D tilt angle determination

Let δ be the angle the angle between the accelerometer Z axis and gravity (or Z'). Here we added X and Y components of acceleration to get a new vector orthogonal to the acceleration vector in the Z direction. Knowing that the magnitude of the sum vector is $\sqrt{a_X^2 + a_Y^2}$, we may write the following equation,

$$\delta = tan^{-1} \frac{\sqrt{a_X^2 + a_Y^2}}{a_Z}$$

Note that the voltages obtained from the accelerometer will need to be converted to accelerations, as shown in the previous sections.

Also from the figure about, let α and β be the angles between the X and Y accelerometer axes and ground, respectively, and hence their calculations are analogous. The specific case of α is shown, employing methods similar to the ones use for finding δ. To find the remaining quantities,

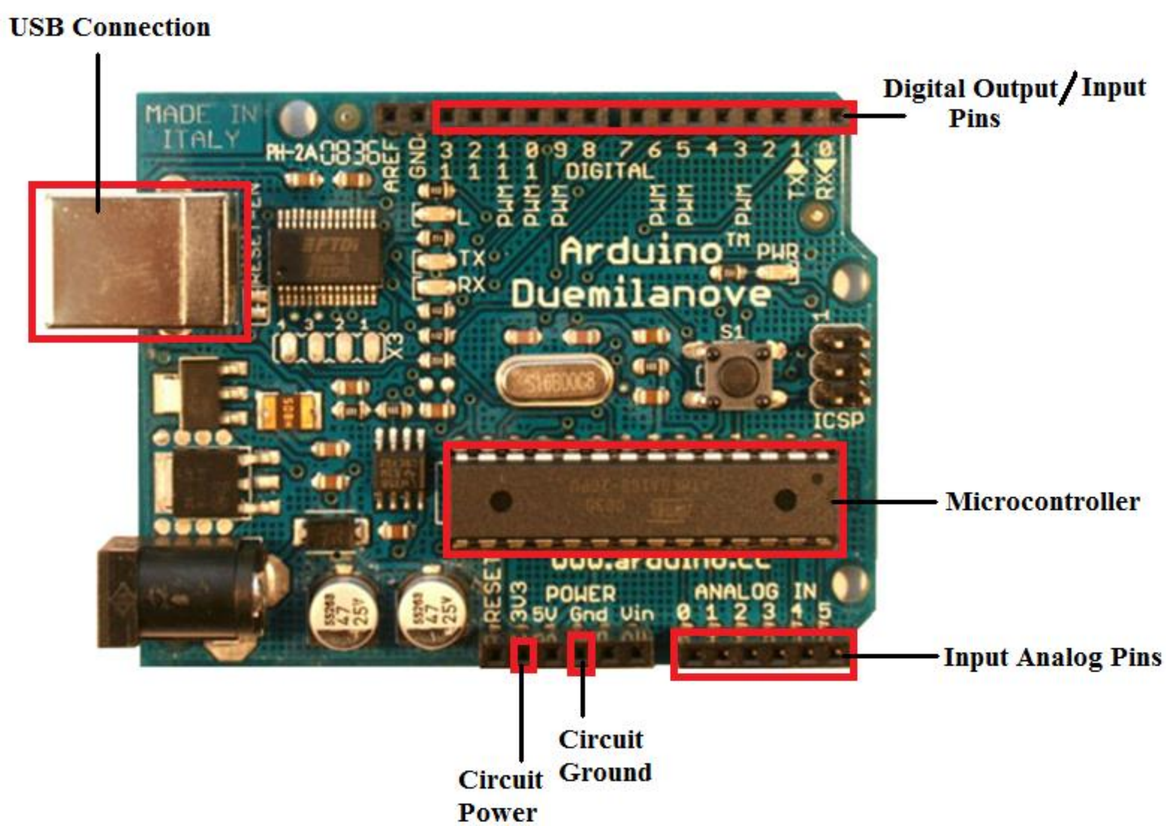
$$\alpha = tan^{-1}\ \frac{a_X}{\sqrt{a_Z^2 + a_Y^2}}$$

$$\beta = tan^{-1}\ \frac{a_Y}{\sqrt{a_Z^2 + a_X^2}}$$

APPENDIX B: Accelerometer Tilt Sensing With Arduino Duemilanove

B.1 Understanding the Microcontroller

The particular model of Arduino Microcontroller chosen for this application is the **Arduino Duemilanove USB Microcontroller Module,** based on Atmega324



architecture.

This particular module comes pre-assembled with the necessary pin connections and various power options, but for the purposes of tilt implementation, the focus is on the areas highlighted in red. The first thing to note is that the module has a USB connection, which allows the processor to be programmed on any PC with a suitable connection. Additionally, the USB connection supplies power to the module, and allows for a 3.3 V output and a 5 V output. These outputs are very convenient when it comes to powering the accelerometers which are later interfaced with the microcontroller. In this case, we use the 3.3 V output (due to power maximums on accelerometer) and the ground labeled above, to supply our accelerometer circuit. Next we connect the accelerometer outputs (x,y,z pins) to **Input Analog Pins** of the microcontroller. Each pin will provide an independent accessible value based on the output voltage of the accelerometer. The A/D converter on the Arduino reports the voltage in steps, so for a 10 bit A/D converter, there are 2^{10} -1= 1023 steps, or independent voltage levels that the A/D can resolve. Once values for pins x,y,z are obtained in the microprocessor, it is possible to use an **Integrated Development Environment** (IDE) to upload code to the microcontroller. Finally, the **Digital Output/Input Pins** provide a constant voltage (low or high) to drive external actuators, in this case, vibration motors. As well, the digital pins may be used as input pins as well, and using a pushbutton it is possible to calibrate the accelerometer tilt sensing to account for offsets.

B.2 Tilt Sensing and Arduino IDE

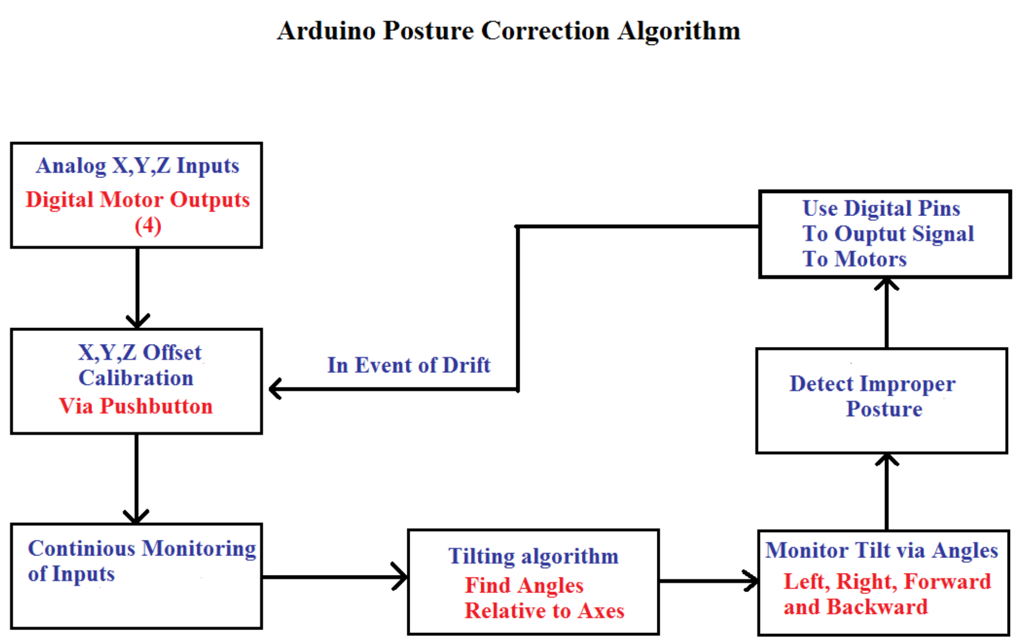


Figure 2. Block Diagram of
Posture Correction Implementation

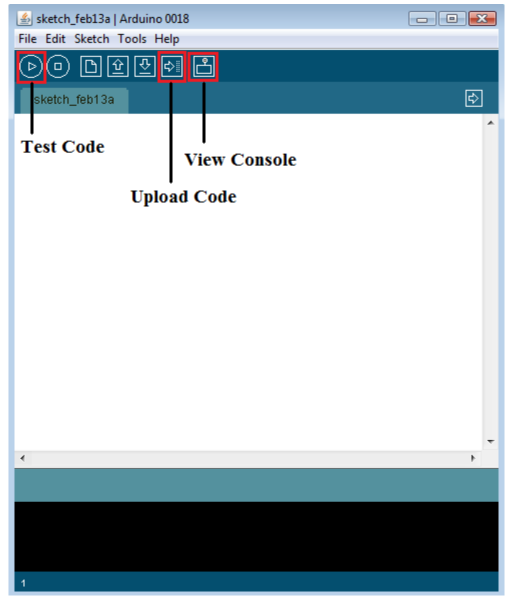


Figure 3. Layout of IDE

Above is a representation of the algorithm used to implement tilt sensing via the microcontroller. The first step is to define inputs to obtain the signal from the microcontroller, and also define outputs to control the vibration discs. In order to do this, programming of the microcontroller may be done by downloading the freely available **Arduino IDE Software,** available here: http://arduino.cc/en/Main/Software. Things to note are highlighted in red. The "play" button tests code before it is uploaded to the microcontroller. Once the code is errorless, it may be uploaded via the button indicated above. It is essential that nothing is attached the microcontroller during programming. Finally, for testing purposes it is possible to write outputs or inputs to the console, which can be accessed as shown.

B.3 Accelerometer Signal Acquisition

Any Arduino program must start with the following template:

```
//pin and other variables
void setup() //basic setup parameters
{
}
void loop() //main loop as microcontroller operates in real-time
{
}
```

Signal acquisition is done by assigning pins to certain variables, and then using functions related to analogue input to read in the analog input.

```
//pin and other variables
const int xpin = 1;
const int ypin = 2;
const int zpin = 3;
```

The variable names can be arbitrary, as long as the type is "int". Any function related to analogue input will interpret the variable value as the pin number, and will read the respective pin (as labeled on module). Here we choose 3 pins for a single accelerometer, with x,y,z designations.

```
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  Serial.print(analogRead(xpin));
  Serial.print(analogRead(ypin));
  Serial.print(analogRead(zpin));
  delay(100);
}
```

The rest of the code prints out the pin values using **Serial.print**, and reads in the values using **analogueRead(variableName)**. However, these variables need to be stored, and so we can define some integer variables to hold these values:

```
int varX;
int varY;
int varZ;

void setup()
{
  Serial.begin(9600);
}
void loop()
{
  varX = analogRead(xpin);
  varY = analogRead(ypin);
  varZ = analogRead(zpin);
  delay(100);
}
```

To complete this section, we construct a simple circuit of the accelerometer and make appropriate connections to the microcontroller, and investigate the output using the serial console:

The following code was pre-loaded into the microprocessor:

```
const int xpin = 1;
const int ypin = 2;
const int zpin = 3;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  Serial.print("X value: ");
  Serial.print(analogRead(xpin));
  Serial.print("\t Y value: ");
  Serial.print(analogRead(ypin));
  Serial.print("\t Z Value: ");
  Serial.print(analogRead(zpin));
  Serial.println();
  delay(100);
}
```

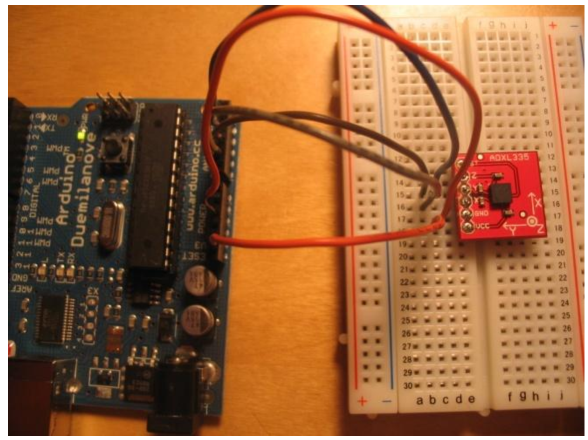


Figure 4. Accelerometer and Microcontroller Interface

It is important to realize that the outputs for these pins are not measured in volts, but rather as a step size representative of a discrete scale describing the full-scale voltage. For example, if the X value was 365, then the true voltage on that pin would be:

$$v = \frac{\#\,steps}{\#\max steps} * V_{fullscale} = \frac{365}{1023} * 3.3V = 1.177\ V$$

However, in our tilt sensing applications, the actual voltage from the accelerometer is required, and appropriate conversion must be done.

B.4 Calibration

Seeing as the tilt sensing apparatus will be worn by different people with different physical attributes, it is important to calibrate the apparatus to account for this variation. The accelerometer will be mounted such that 2 axes are perpendicular to the g force and 1 axis is directly parallel. As such, 2 of the axes will be easy to calibrate, because they will be near the 0 g configuration, but the third axis will be difficult and will require some estimation.

It would be helpful if calibration may be done at any point during the operation of the apparatus, and so we may use a pushbutton switch to enter a calibration routine. Seeing as how we do not want to constantly poll the switch, we may use interrupts and interrupt routines to accomplish this task. Interrupts are internal constructs that allow the processor to respond to external inputs the moment they happen, and then execute code dedicated to the interrupt. For our application, **Digital Pin 0** will serve as the interrupt pin. The pushbutton will be attached to this pin, and will be in the **low** configuration. The interrupt will respond to a rising edge of the button, and the switch will need to be denounced as well to avoid multiple activations of the interrupt. The following is a basic template for constructing interrupts on the Arduino:

```
volatile int interruptVar;
void setup()
{
  Serial.begin(9600);
  attachInterrupt(0, routine, MODE);
}
void loop()
{
  //main program code
}
void routine()
{
  // operates on volatile variables, interruptVar
}
```

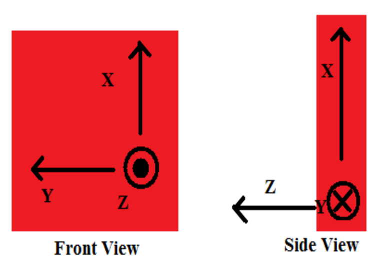


Figure 5. Device orientation on subject

In the above code, we first initialize a volatile variable which we can modify in our interrupt service routine (code to be executed when interrupt is invoked). In the setup routine, we use **attachInterrupt** to assign the interrupt to **Digital Pin 0** and "routine" is code which will be run as soon as the interrupt is called. Finally, MODE refers to how the interrupt is called, and MODE may be replaced with the following parameters depending on the requirements:

Now for the actual calibration of the accelerometer, it is first important to choose how the device is to be mounted on the person, and this will be done as seen in **figure 5.** As such, the X and Z axes will track right and left motion, and Y and X axes will track backwards and forward motions. Thus, the Y and Z axes are easily calibrated by setting their value at interrupt to the offset value. The X offset value maybe found by adding the sensitivity value to value obtained from the X pin (more details are available in the **Accelerometer Tilt Sensing** note). In order to ensure an accurate offset value, we will take a sample average of several values over a period of 1-2 seconds. The following is the code implementation of the calibration routine for the accelerometer:

```
//set analog input pins

float offx;
float offy;
float offz;
float Xsensitivity;
int i;
volatile CalibrateFlag=0; //variable to indicate whether interrupt has occured

void setup()
{
  Serial.begin(9600);
  attachInterrupt(0, routine, RISING);  //react on rising edge, or upon button press
}
void loop()
{
  if(CalibrateFlag==1){  //if interrupt is called
   offx = 0;
    offy = 0;
    offz = 0;
for(i=0;i<10;i++){ //find average value of offset for calibration
 offx = offx + analogRead(xpin);
offy = offy + analogRead(ypin);
offz = offz + analogRead(zpin);
delay(50);
                    }
offx = offx/10 *3.3/1024 - sensitivityX; //special case for X axis sensitivity
offy = offy/10 *3.3/1024;
offz= offz/10*3.3/1024;
CalibrateFlag=0; //reset calibration flag
         }

//rest of program
}
void routine()
{
  CalibrateFlag = 1;
}
```

In this case, we have a **CalibrateFlag** which changes if the interrupt is called. It is better to have the interrupt routine inside the main code rather than a separate function because it saves on computational time and causes less delays. Note that the offset values are also converted to voltages seeing as the sensitivities of the accelerometer are related to units of volts. Next we combine the code for monitoring of inputs and the calibration, and move onto finding the required angles to estimate tilt.

However, during testing it was found that such interrupts are difficult in terms of switch debouncing, and therefore a simple polling method was used in order to calibrate, and was implemented as follows:

```
//set digital pin 0 as input
int pin0 = 0;

void setup()
{
  Serial.begin(9600);
  //initialize pin0 as input
  pinMode(pin0, INPUT);
}

if(digitalRead(pin0)==1){  //calibration switch is pressed
//do same code required for calibration as long as pin0 is high
}
```

B.5 Finding the angles

As per the **Accelerometer Tilt Sensing** document, we require to find angles of the axes with respect to the vertical, axes and the horizontal plane. In this case, the X axis will track angles with respect to the vertical, and Y and Z axes will track angles with respect to the horizontal. We investigate pairs of angles at a t time to determine tilt direction. Let the angles of the X axis relative to the vertical be **δ**, and the Y and Z axes relative to the horizontal plane be **α** and **β** respectively. Then, by observing α and δ as a pair, we detect tilting left and right. By observing β and δ as a pair, we observe forward and backward motion.

The following is a full implementation of tracking the analogue inputs, conversion to voltages, then conversion to workable value to determine the angles.

```
//set analog input pins
//set calibration variables
#include <math.h>
#include <float.h>

//set offsets
float offx=0;
float offy=0;
float offz=0;

//set variables to hold analog inputs
float Xval=0;
float Yval=0;
float Zval=0;

//set accelerations
float ax;
float ay;
float az;

//angles
float alpha=0;
float beta = 0;
float delta = 0;

//accelerometer device sensitivities
float sensitivity=0.22;

void setup()
{
  Serial.begin(9600);
  //calibration setup here
}
void loop()
{
// calibration to be done here
//read in values from accelerometer
  Xval = analogRead(xpin);
  Yval = analogRead(ypin);
  Zval = analogRead(zpin);

// find accelerations
ax = (Xval*3.3/1023 – offx)/ sensitivity;
ay = (Yval*3.3/1023 – offy)/ sensitivity;
az= (Zval*3.3/1023 – offz)/ sensitivity;

//find the angles and convert to degrees
  alpha =   atan(az/sqrt(pow(ay,2)+pow(ax,2)))*(180/3.14592);
  beta =   atan(ay/sqrt(pow(az,2)+pow(ax,2)))* (180/3.14592);
  delta = atan(sqrt(pow(ay,2)+pow(az,2))/ax)* (180/3.14592);
}
```

B.6 Determining degree of tilt

A brief algorithm to accomplish such tilt sensing is as follows:

- Check **δ**, if it deviates significantly from offset then tilting is occurring

- If **δ** deviates significantly, then check **α**, if above positive angle threshold the tilting backward, if lesser than negative threshold, then tilting backward

- If **δ** deviates significantly, then check **β**, if above positive threshold, then tilting left, if below negative threshold then right

```
//variable declarations as in part E
//flags to determine direction of tilt
FlagLR =0; //flag to determine left and right tilt
FlagFB = 0; //flag to determine forward and backward tilt
//thresholds on angles that indicate tilt adjustment is required
alphaThresh ;
betaThresh ;
deltaThresh ;
void setup()
{
  Serial.begin(9600);
  }
void loop()
{
//calibration and signal acquisition and processing as in part E

if(delta>=deltaThresh){
                    if(beta>betaThresh) FlagLR = 1; //tilting left
                    else if (beta<-betaThresh) FlagLR=-1;//tilting right

                    if(alpha>alphaThresh) FlagFB= -1; //tilting backward
        else if (alpha<-alphaThresh) FlagFB = 1; //tilting forward

                    }
else{
  FlagLR=0;
  FlagFB=0;
}
//set front and back vibration
 if(FlagFB==1)
  digitalWrite(frontVib, HIGH);   // set actuator motor on
 else if(FlagFB==-1)
 digitalWrite(backVib, HIGH);
  else{
                    digitalWrite(frontVib, LOW);
         digitalWrite(backVib, LOW);
}
//set left and right vibration
if(FlagLR==1)
  digitalWrite(leftVib, HIGH);   // set actuator motor on
 else if(FlagLR==-1)
 digitalWrite(rightVib, HIGH);
  else{
                    digitalWrite(leftVib, LOW);
        digitalWrite(rightVib, LOW);
        }
}
```

Here we define thresholds to detect when the angles are too great and need to be adjusted. Settings flags to -1 or +1 indicates the direction of tilt, when the flag is 0 it indicates that no tilt is occurring. It is worth noting that both negative and positive angles are produced for **α** and **β** and depending on the sign we can determine whether tilt is occurring left or right, or forward or backward.

B.7 Posture Correction via Vibration Actuators

Once we know that the angles are above threshold and tilting is occurring in a certain direction, it is possible to let the user know in which direction tilting is occurring and how the user may correct the posture. For this we use vibration actuators, which will vibrate based on the direction the user is leaning in. The vibration sensors will be attached to a module on the chest as seen in **Figure 6.** To cause vibration, we attach the actuators to **Digital Pins 1-4**. The following code is a complete working program that implements tilt sensing and feedback using the Arduino Microcontroller and the ADXL335 accelerometer. Tilt angles are also available via serial console view.

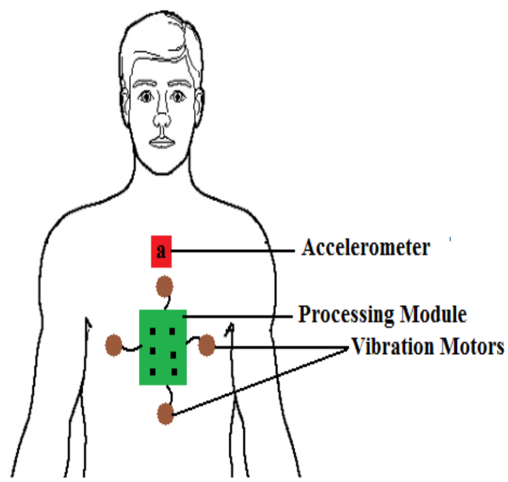


Figure 6. Vibration feedback setup

B.8 COMPLETE CODE

```
##include <math.h>
#include <float.h>
int pin9 = 9;
//set up vibration outputs
int leftVib = 2;
int rightVib =3;
int frontVib =4;
int backVib =5;

//set input analog channels
const int xpin = 3;
const int ypin = 2;
const int zpin = 1;

//set variables to hold analog inputs
float Xval=0;
float Yval=0;
float Zval=0;

//set offsets and calibration stuff
float offx=1.5;
float offy=1.5;
float offz=1.5;
int i;
volatile int CalibrateFlag=0; //variable to indicate whether interrupt has occured

//set accelerations
float ax;
float ay;
float az;

//angles
float alpha=0;
float beta = 0;
float delta = 0;

//accelerometer device sensitivities
float sensitivity=.22;

//flags to determine direction of tilt
int FlagLR=0; //flag to determine left and right tilt
int FlagFB=0; //flag to determine forward and backward tilt

//thresholds on angles that indicate tilt adjustment is required
```

```
float alphaThresh=30;
float betaThresh=30;
float deltaThresh=30;

void setup()
{
  Serial.begin(9600);
      // initialize the digital pin as an output:
  pinMode(leftVib, OUTPUT);
  pinMode(rightVib, OUTPUT);
  pinMode(frontVib, OUTPUT);
  pinMode(backVib, OUTPUT);
  //initialize pin0 as input
  pinMode(pin9, INPUT);

}
void loop()
{
  if(digitalRead(pin9)==0){  //calibration switch is pressed
   offx = 0;
    offy = 0;
    offz = 0;
for(i=0;i<10;i++){ //find average value of offset for calibration
 offx = offx + analogRead(xpin);
offy = offy + analogRead(ypin);
offz = offz + analogRead(zpin);
delay(50);
                  }
offx = offx/10 *3.3/1023 - sensitivity; //special case for X axis sensitivity
offy = offy/10 *3.3/1023;
offz= offz/10*3.3/1023;
          }
//read in values from accelerometer
  Xval = analogRead(xpin);
  Yval = analogRead(ypin);
  Zval = analogRead(zpin);
// find accelerations
ax = (Xval*3.3/1023-offx)/sensitivity;
ay = (Yval*3.3/1023-offy)/sensitivity;
az = (Zval*3.3/1023-offz)/sensitivity;
//find the angles and convert to degrees

  alpha =   atan(az/sqrt(pow(ay,2)+pow(ax,2)))*(180/3.14592);
  beta =   atan(ay/sqrt(pow(az,2)+pow(ax,2)))* (180/3.14592);
  delta = atan(sqrt(pow(ay,2)+pow(az,2))/ax)* (180/3.14592);

 Serial.print("alpha value: ");
```

```
 Serial.print(alpha);
  Serial.print("\t beta value: ");
  Serial.print(beta);
  Serial.print("\t delta Value: ");
  Serial.print(delta);
  Serial.println();
  Serial.print("\t back Value: ");
  Serial.print(digitalRead(backVib));
    Serial.print("\t front Value: ");
  Serial.print(digitalRead(frontVib));
    Serial.print("\t Left Value: ");
  Serial.print(digitalRead(leftVib));
    Serial.print("\t Right Value: ");
  Serial.print(digitalRead(rightVib));
  Serial.println();
  if(delta>=deltaThresh){
		if(beta>betaThresh) FlagLR = 1; //tilting left
		else if (beta<-betaThresh) FlagLR=-1;//tilting right

		if(alpha>alphaThresh) FlagFB= -1; //tilting backward
        else if (alpha<-alphaThresh) FlagFB = 1; //tilting forward
		}
else{
  FlagLR=0;
  FlagFB=0;
}
//set front and back vibration
 if(FlagFB==1)
  digitalWrite(frontVib, HIGH);   // set actuator motor on
 else if(FlagFB==-1)
 digitalWrite(backVib, HIGH);
  else{
		digitalWrite(frontVib, LOW);
         digitalWrite(backVib, LOW);
}

//set left and right vibration
if(FlagLR==1)
  digitalWrite(leftVib, HIGH);   // set actuator motor on
 else if(FlagLR==-1)
 digitalWrite(rightVib, HIGH);
  else{
		digitalWrite(leftVib, LOW);
        digitalWrite(rightVib, LOW);
        }

}
```

REFERENCES

[1] A. Briggs, A. Smith, L. Starker, and P. Bragged, "Thoracic spine pain in the general population: Prevalence, incidence and associated factors in children, adolescents and adults. a systematic review," *BMC Musculoskeletal Disorders*, vol. 10, pp. 77-89, June 2009

[2] S. Arteaga, J. Chevalier, A. Coile, A. William Hill, S. Sali, S. Sudhakhrisnan, S. Kurniawan, "Low-Cost Accelerometry-Based Posture Monitoring System for Stroke Survivors," *Assets '08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, pp. 243-244, 2008

[3] K. Aminian, and B. Najafi, "Capturing human motion using body-fixed sensors: outdoor measurement and clinical applications: Research Articles," *Comput. Animat. Virtual Worlds*, vol. 15, no. 2, pp. 79-94, 2004

[4] H.J. Luinge, and P.H. Veltink, "Measuring orientation of human body segments using miniature gyroscopes and accelerometers", *Medical and Biological Engineering and Computing*, vol. 43, no. 2, pp. 273-282, 2005

[5] R. Foster, L. Lanningham-Foster, J. Levin, "Optimization of accelerometers for measuring walking", *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*, vol. Volume 222, no. 1, pp. 53-60, 2008

[6] R. Moe-Nilssen, J. Helbostad, "Estimation of gait cycle characteristics by trunk accelerometry," *Journal of Biomechanics*, vol. 37, no. 1, pp. 121-126, 2004

[7] R. Mayagoitia, et al, "Standing balance evaluation using a triaxial accelerometer" *Gait & Posture*, vol. 16(1), pp. 55-59, 2002

[8] J. Kavanagh, H. Menz, "Accelerometry: A technique for quantifying movement patterns during walking", *Gait & Posture*, vol. 28, no. 1, pp. 1-15, 2008

[9] F. Foerster, "Detection of posture and motion by accelerometry: a validation study in ambulatory monitoring," *Computers in Human Behavior*, vol. 15, no. 5, pp. 571-583, September 1999

[10] A. Godfrey, R. Conway, D. Meagher, and G. OLaighin, "Direct measurement of human movement by accelerometry." *Medical engineering & physics*, vol. 30, no. 10, pp. 1364-1386, December 2008

[11] C. Mizuike, S. Ohgi, and S. Morita, "Analysis of stroke patient walking dynamics using a tri-axial accelerometer," *Gait & Posture*, vol. 30, no. 1, pp. 60-64, July 2009

[12] D. Jeong, K. Do, W. Chung, "Implementation of the Wireless Activity Monitoring System Using Accelerometer and Fuzzy Classifier", *International Journal of Information Systems for Logistics and Management.* vol. 3, no. 2, pp. 115-120, 2008

[13] J. Swann, "Good Positioning: The Importance of Posture", *Nursing & Residential Care*, September, Vol 11, No 9, pp. 467-469, 2009

[14] R. J. Sanders, "Etiology and pathology", *Hand Clin*, vol. 20 pp. 23–26, 2004

[15] V. Kaushik and N. A. Charpe, "Effect of Body Posture on Stress Experienced by Worker", Stud. Home Comm. Sci., vol. 2 no. 1 pp.1-5, 2008

[16] E.S. Sazonov et al, "Automatic Recognition of Postures and Activities in Stroke Patients", *31st Annual International Conference of the IEEE EMBS*, 2009

[17] C.X. Rosado, L. Simone, "Translational Haptic Feedback for Post-. Stroke Rehabilitation", *IEEE 33rd Annual Northeast Bioengineering Conference*, 2007

[18] C. Oliveira, "Balance control in hemiparetic stroke patients: Main tools for evaluation", *J Rehabil Res* Dev. vol.45 no.8 pp.1215-26, 2008

[19] I. Tetsuo et al, "Dynamic Postural Control inmPatients with Hemiparesis", *American journal of physical medicine & rehabilitation*, vol. 82, no 6, pp. 463-469, 2003

[20] Q. Li, "Accurate, Fast Fall Detection Using Gyroscopes and Accelerometer-Derived Posture Information", *Proceedings of the 2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks*, 2009

[21] M. Clifford and L. Gomez, " Measuring Tilt with Low-g Accelerometers", *Freescale Semiconductor*, Application Note, 2005.

[22] K. Tuck, " Tilt Sensing Using Linear Accelerometers", *Freescale Semiconductor,* Application Note, 2007.

[23] K. Tuck, "Implementing Auto-Zero Calibration Technique for Accelerometers", *Freescale Semiconductor*, Application Note, 2007.

[24] "The Importance of Posture", *Physical Therapy at Briarcliff & Jefferson Valley,* 2006.

[25] Jeanne Markusic, "Maintaining a Healthy Spine – Posture", *SpineUniverse, 2009,* available at: http://www.spineuniverse.com/wellness/ergonomics/maintaining-healthy-spine-posture

VITAE

September 22, 2009

David Zhitomirsky

Home Address: 148 Huntingwood Ave. Dundas, Ontario, Canada, L9H6X7
Tel. 905 807 2358
Email: zhitomd@mcmaster.ca

EDUCATION BACKGROUND

Bachelors of Electrical and Biomedical Engineering – McMaster University, Hamilton, Ontario, Canada (current)
High school Diploma – Highland Secondary School, Dundas, Ontario, Canada (2006)

ACADEMIC RECORD

- **11.9 Academic Cumulative Average based on a 12 point system (4.0 GPA equivalent) (current)**

MAJOR ACADEMIC AWARDS

- **Nominee for the Chancellor's Gold Medal Award in Engineering**
- **Provost Honor Roll Medal for highest possible academic achievement**
- **Canadian Governor General's Medal for highest academic achievement**

OTHER AWARDS

McMaster University

- Ontario Professional Engineers Foundation for Education Scholarship (2009)
- Ashbaugh scholarships (2006-2009)
- McMaster Future Fund In-Course Award (2008)
- University Senate Scholarship (2008)
- University Senate Scholarship (2007)
- Queen Elizabeth Aiming for the Top Scholarship (2006-2009)
- Nortel Entrance Scholarship (2006-2007)
- Engineering Competition Scholarship (2006)

High school

- **Top Graduating Student based on a class of 200**
- Grade 12 subject awards **based on highest grade** in:
  - **Geometry and Discreet**
  - **Calculus**
  - **Physics**
  - **Chemistry**
  - **Biology**
  - **Computer Science and Programming**
  - **Financial Accounting**
- Graduating average of **99%**
- Special honors for participation in **math**, **physics** and **chemistry** contests

Research Interests

Major interest in electronic devices and applications. Specifically, MEMs, optical and chemical sensors, integrated electronics such as hearing aids or visual aids. Interest in microelectronics, materials for transistors and similar devices, device characterization and signal conditioning using custom instrumentation.

Research Experience

Undergraduate Research Opportunities Program

1.Research project in collaboration with Prof. A.Boccaccini, Imperial College (London, UK) on biomedical implants [1]

Synthesis of hydroxyapatite nanoparticles. Electrodeposition of organic-inorganic composites, containing bioglass, hydroxyapatite, chitosan, alginate and proteins. SEM, EDS, TGA, DTA and XRD studies. QCM studies of protein adsorption, FTIR and UV spectroscopy, impedance spectroscopy.

2.Research project in collaboration with Prof.A.Adronov on carbon nanotube-polymer-metal oxide composites

Synthesis of inorganic oxides, dispersion of inorganic nanoparticles and CNT, electrophoretic deposition of polymer-CNT and polymer-metal oxide composites, QCM studies of electrochemical deposition kinetics, SEM, XRD, EDS, optical microscopy, cyclic voltammetry, chronopotentiometry studies.

Natural Sciences and Engineering Reasearch Council of Canada

3. Research project with Prof.J.Deen and Prof. R. Selvaganapathy on Miniaturized Reference Electrode [2]

Microfabrication of a miniaturized microfluidics Silver/Silver Chloride Reference Electrode for use in the BioFET for detection of single stranded DNA. Fabrication of microfluidics device in a clean room environment, use of UV Photolithography with SU-8 substrate. Cyclic voltammetery and potentiostatic measurements to test working performance of reference electrode. Polydimethylsiloxane (PDMS) mould formation for use as base of reference electrode.

Research Skills

Materials processing and device fabrication methods

Sol-gel synthesis, chemical precipitation, electrophoretic deposition, electroplating, electrosynthesis, SU-8 photolithography, PDMS mould formation, nanoporous microcontact printing using PDMS, chemical separation methods, operational amplifier design, microprocessor programming and interface, biomedical instrumentation design, clean room training

Characterization methods

Scanning electron microscopy, energy dispersive spectroscopy, thermogravimetric analysis, differential thermal analysis, X-ray diffraction, optical microscopy, cyclic voltammetry, chronopotentiometry, quartz crystal microbalance, Fourier transform infrared spectroscopy, UV spectroscopy, impedance spectroscopy, electrical circuit analysis, modeling using electrical circuit equivalents, Nuclear Magnetic Resonance

Computer Skills

Matlab, LabView, C++, Java, C, C#, Assembly Language, Microsoft Office Suite (Word, Excel, etc.), Dreamwear and HTML, photoshop, Origin Pro

Publications

[1] D.Zhitomirsky, A.Boccaccini, J.Roether and I.Zhitomirsky. "Electrophoretic deposition of nanocomposites containing bioglass". J.Materials Processing Technology, 209 (2009) 1853-1860

[2] M. W. Shinwari, D. Zhitomirsky, I.A. Deen, P. R. Selvaganapathy and M. J. Deen. "Microfabricated Reference Electrodes and their Biosensing Applications". Submitted on 18/09/2009 to Biosensors and Bioelectronics

Personal Interests

- Classical and Acoustic Guitar
- Arts and Classical Music
- Travel
- Weightlifting and fitness
- Nature and the outdoors