Real Time Swim Instructor

By

KonradAntoniuk

Senior Project

ELECTRICAL ENGINEERING DEPARTMENT

California Polytechnic State University

San Luis Obispo

June 2014

Contents

Figures

Tables

Abstract

Both competitive and recreational swimmers want an efficient swim stroke but often lack consistent feedback with technical insight. Competitive swimmers rely on coaches for instruction but have to stop swimming to listen. The Real Time Swim Instructor project provides live feedback to a swimmer so they can feel their stroke correction*while* they're swimming. The developing swimmer feels the difference between correct and incorrect form instantaneously.

The project takes on the responsibility of measuring the moving body, accurately determining stroke improvement, functioning under water, and providing recognizable feedback. Additionally, the user can review their swim from data stored in memory. The device does not restrict or alter the swimmer's natural stroke so they can swim normally with the gear as they do without it.

Chapter 1

Introduction

Although every swimmer's stroke differs, many common swim strokeissues depend on arm position alignment. Swimmers that bend there arm inwardstoo much are crossing over resulting in unequal hip rotation or crooked swimming. Bending the arm out too far from the center also affects the swimmer's direction by applying additional forces perpendicular to the intended direction. The most efficient stroke takes advantage of directing the swimmer with the entire magnitudeof the generated force in a straight line. The perfect stroke is humanly impossible but the best swimmers are the ones who have mastered swimming with the highest efficiency stroke fast.

An interest in electrical engineering and 15 years of competitive swimming experience led to the idea of creating the real time swim instructor. The main goal of the device for this project is to help eliminate the elementary errors associated with arm position. Most swimmers do not know when they're swimming inefficiently because they've never felt the strength of a good stroke. Additionally, they do not receive enough constant attention and feedback from coaches when they're practicing with 30 other swimmers. From my experience helping coach stroke technique, I've found as little as 5 minutes can help a person improve their stroke, feeling like they can pull more water and swim easier through the water. This project allows the user to continuously work on their stroke and feel the difference between improper and proper for as long as they want.

In order for the real time swim instructor to successfully improve stroke technique, the project must meet the customer needs. Personal experience and feedback from Cal Poly University swim team coaches and swimmers assisted in deciding that accuracy, underwater functionality, and a hydrodynamic profile are requirements the customer would expect along with user-friendly setup and reliable feedback. Accuracy makes sure the product does not misinform the swimmer with perfect stroke. Underwater functionality assures the device works for swimmers in the water. Hydrodynamic profile limits how much drag the device creates. The user-friendly setup ensures the product does not restrict the swimmer's ability with the goal in mind that the swimmer has the same stroke wearing the device as without it. And finally, a reliable form of feedback tells the swimmer how to fix their stroke. Without feedback, the device has no functionality in the pool for the swimmer.

Chapter 2

Requirements and Specifications

For each of the marketing requirements derived from the customer needs, the engineering specifications listed in table 1 fulfill the project's needs.

Table 1: Requirements and specifications table.

Marketing Requirements

1. User friendly
2. Hydrodynamic profile
3. Works underwater
4. Provides feedback
5. Accurate

Marketing Requirements	Engineering Specifications	Justification
1,2,3	The system functions wirelessly.	Designing a swimmer friendly system entails making it wireless since excess wiring creates discomfort and unwanted drag in the water.
4,5	Sensors recognize 5-15 degree shifts of arm position to within a degree of actual position.	Sensors need to be able to recognize when the swimmer's arm crosses over their body. A crossover ranges from 5-15 degrees off of a straight arm above the swimmer's head.
1,2,3	The system carries voltage signals and processes data at least 5m underwater.	Swimmers swim in the water so the wearable electronics need to be waterproof.
4,5	The system captures sensor data of at least a 100Hz sampling rate.	The entire body is in motion and swimmers seek to shave off hundredths of a second from their times so .01s sampling rate provides a good amount of data.
1,3,5	The entire system functions from a 5V supply.	The system is going to be wireless and most microcontrollers require a 5V voltage supply. Also a low voltage system consumes less power.
1,4	Provides live user feedback and access to at least 5 minutes of data.	The device provides enough meaningful data so the swimmer can fix their stroke in and out of the pool.
1,2	Wearable electronics are within half inch of skin and weigh less than a half pound each.	This specification keeps the hardware from affecting the swimmer's stroke and prevents excessive water drag due to the systems additional surface area.

The deliverables of the project are shown in table 2 and listed as a reminder of important due dates for completion of the senior project. Additionally a Gantt chart displaying the allocated time for each stage of the project can be found in Appendix B.

Table 2: Deliverables

Delivery Date	Deliverable Description
12/9/13	Final Planning Report
3/7/14	EE 463 report
2/20/14	Design Review
3/7/14	EE 463 demo
5/28/14	EE 464 demo
TBD	ABET Sr. Project Analysis
5/29/14	Sr. Project Expo Poster
5/28/14	EE 464 Report

Chapter 3

Design

The initial project design analysis from a level zero diagram standpoint is shown in Figure 1. Table 3 shows the functionality for the module and each input and output in the level zero diagram.

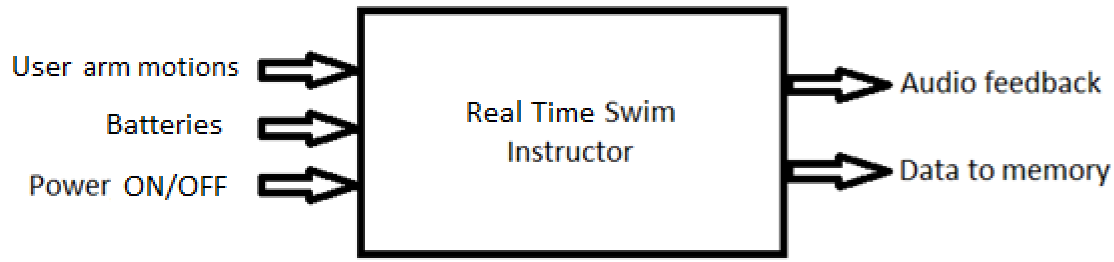

Figure 1: Level 0 block diagram

Table 3: Functional level 0 input and output requirements

Module	Real Time Swim Instructor
Inputs	-User arm motions: The angles of each elbow and arm are measured.The measured angle controls the voltage signal captured by the microprocessor. -Batteries: Device is powered through batteries. -Power ON/OFF: Power switch that turns power off and on to conserve power when the device is not in use.
Outputs	-Audio feedback: Beats with varying tone and frequency provide user feedback. -Data: StoredMCU data is accessible.
Functionality	Acquires user input signals and determines feedback for stroke correction. Batteries power the entire system. The user receives real time audio feedback. System logs sensor data to accessible MCU memory. On/Off switch conserves power when the device is being used or not.

Diving a level deeper into the design reveals the parts resposnsible for function inside of the module as shown in Figure 2. Table 4 reveals the functions of each subsystem.

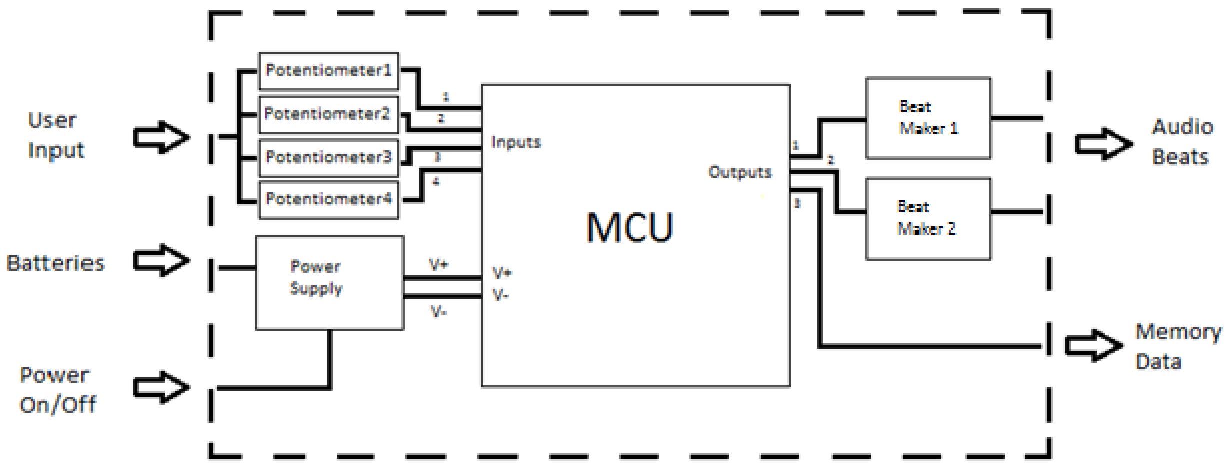

Figure 2: Level 1 block diagram.

Table 4: Level 1 design functionalities.

Module	Potentiometers 1-4
Inputs	Angle measurement of elbow or arm.
Outputs	Varying resistance is interpreted by MCU.
Functionality	The potentiometer converts the user's body's angles into resistance.
Module	Power Supply
Inputs	-Batteries: Voltage power supply -Switch: Turns device ON/OFF
Outputs	- V+: Provides 5V supply in respect to negative node. - V- : Acts as a ground to the positive voltage node.
Functionality	Provides switchable power to the MCU.
Module	MCU
Inputs	-Resistance inputs: Reads resistance as measured by potentiometers. -V+/V-: Voltage rails from power supply that power the MCU.
Outputs	-1,2: Provide varying frequencies. -3: Access to data in memory.
Functionality	The microcontroller, powered by the power supply, determines the necessary frequencies to output according to the varying resistances measured by the inputs. The input data is saved in memory and can be sent through a separate output.
Module	Beat Makers 1,2
Inputs	-1,2: Frequency
Outputs	-Audio: Provides feedback to the user.
Functionality	The beat maker produces audio beats in sync with the input frequency.

References

1. X. Yang, *Underwater Acoustic Sensor Networks*. Boca Raton: Auerbach Publications, 2010. This book informs the reader of underwater acoustic sensors which may be used as a form of feedback in my system. Published in 2010, the book may be considered up to date with modern technology.

2. P. Petrov, 2013, Oct. 16. *Measure two resistive sensors or multiple switches with a single Schmitt* [online]. Available: www.edn.com/design/sensors/4422744/Measure-two-resistive-sensors-or-multiple-switches-with-a-single-Schmitt. This is a design idea that would reduce the wiring in my system since it provides 2 sources of data with one path and eliminates the need for an ADC. This design idea was published on the EDN Network website, which is a trusted source provided by Professor Braun.

3. *HEF40106B Hex Inverting Schmitt Trigger*. NXP Semiconductors. 2011, Nov. 21. This is the datasheet of the Schmitt trigger that was used in the design implementation cited earlier and may be of use if indeed will be implemented. The data sheet comes from NXP Semiconductors which is a successful company that produces semiconductor devices.

4. *Tilt Sensors Information.* IHS GlobalSpec [online]. 2013, October 20. Available: www.globalspec.com/learnmore/sensors_transducers_detectors. This site provided valuable info on sensors that would work with my project. HIS GlobalSpec is a professional company that

5. M. Cho. (Aug. 2010). "A low-power real-time operating system for ARC (actual remote control) wearable device." *IEEE Transactions onConsumerElectronics*. [On-line]. 56(3), pp. 1602-1609. Available: ieeexplore.ieee.org/xpl/article [Oct. 21, 2013]. Low power microcontrollers are desirable for any project with wearable electronics that would like to extend battery life. This journal entry was found on the IEEE website, a trusted source for EE's everywhere.

6. L.J. Medeiros, H.C. Basso, and R.A.C. Altafim. "Multi-layer piezoelectret hydrophone for ultrasonic applications," 2012 IEEE International Ultrasonics Symposium, Dresden, 2012. A hydrophone deals with communicating underwater which may be a feasible method of providing feedback in my project. This conference entry was also found on the IEEE website, a reliable source.

7. S.M. Retchin and M. Lenhardt. "Recreational Bone Conduction Audio Device, System." U.S. Patent 8 111 860, Feb. 7, 2012. This patent deals with bone conduction which can provide audio to a swimmer submersed in water which is also a viable form of feedback. This is a patent registered with the Unites States government's patent office.

8. *ITG-3200 Product Specification Revision 1.4*. InvenSense Inc. 2010, Mar. 3. This datasheet provides valuable data for implementing a well-functioning gyro sensor. The datasheet was provided by the sparkfun website which has sold thousands of sensors.

9. Y. Hsiao, M. Huang, and C. Wang. “Development of MSP430-Based Underwater Acoustic Recorder with Multi-MCU Framework,” Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies, Tokyo, 2007. This entry provides valuable information to past submersible electronics with microcontrollers. This entry was also found on the IEEE website.

10. *Water-Proof Connectors*. Edac Inc. May 2012. Waterproof connectors will be required for my project to function. Edac is a Canadian company that has been in business for 46 years producing waterproof connectors as well as other supplies.

Chapter 4

Implementation

Since the end of the introductory design class EE460, the Real Time Swim Instructor project has transitioned from a theoretical design to an actual hardware prototype over the course of 5 months. This chapter focuses on the final design, changes made to the original, and the hardware acquired for assembly. Review of the final design begins with an updated flow chart describing the system functionality as shown in figure 3.

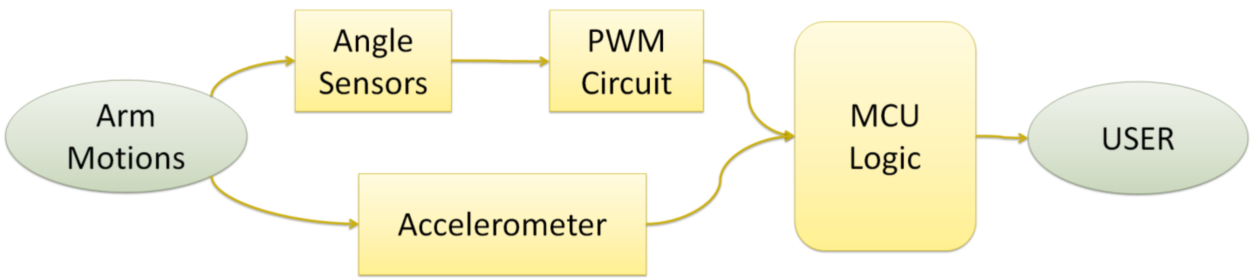

Figure 3: Flow chart describing the final design. This figure updates figure 2 with a new PWM Circuit stage.

Angle Sensors

Beginning with angle sensors, the original design had 4 potentiometers connected directly to the microcontroller. Realizing regular potentiometers would not be able to withstand the mechanical forces of following arm motion, the angle sensors require a special design, one that is waterproof and can smoothly track the changing angles of the elbow and shoulder. Additionally, the project simplified by choosing to work with only two angle sensors versus four, this way the project would only have to focus on one arm instead of both. This also supports the fact that swimmers usually have a problem of crossing over with only one arm.

In researching for a better potentiometer to use, the Sensofoil product line was found. One of Sensofoil company's focuses is membrane potentiometer technology in which they produce IP65 rated thin potentiometers. The IP65 rating helps protect against dust and low pressure water, thus helping satisfy this project's waterproof specification. Having a thin potentiometer also adheres to the hydro dynamic profile specification of the project. Therefore, three Sensofoil PET Foil radial 270 degree 40mm diameter potentiometers,two for actual implementation and another for testing and precaution, were ordered.

To better seal the potentiometers from water, a polyethylene block was trimmed and drilled to encase the potentiometer. Using spare audio cable, conductors were soldered to each terminal of the potentiometer. An additional layer of plastic was laid on the touch membrane to further waterproof the pot. The prepped rotary potentiometer was then covered by glue within the plastic enclosure to ultimately prevent water from reaching the electrical components of the angle sensors. A second thin

plastic pieceis connected on top of the pivot point to provide a free rotating arm. A wiper screwed into the second piece keeps contact with the potentiometer membrane to track the rotary position.

With each arm of the angle sensor attached to proper positions on the arm or body, the angle between the two will be captured as a resistance between 0 and lab measured 2.95 kilo-ohms. Since the potentiometer has a rotary range of 270 degrees, the sensor changes its resistance at a rate of 10.96 ohms per degree. For each sensor, the wiper position is specialized to either the elbow or shoulder so that the limits of the resistive range aren't reached. For example, the elbow cannot rotate more than 180 degrees without breaking, so the sensor will be placed to measure between 45 and 225 degrees. In the shoulder case, since the issue of crossing over occurs when the arm is extended above the head of the swimmer, the target angle is how far off center the arm points, which can range between +/- 90 degrees. Since the shoulder has a greater range of motion in various directions while swimming, the sensor goes off the 270 degree track during portions of the stroke. Such a problem won't matter since the target resistance will be caught in the proper position as ensured by the accelerometer discussed later in the report.

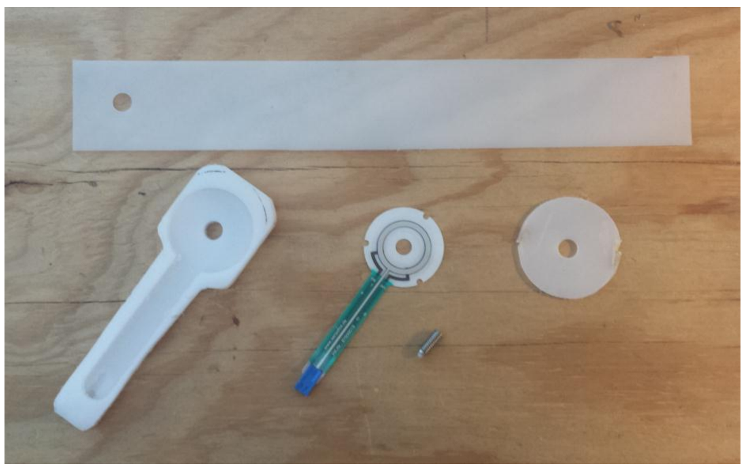

Figure 4: Components used to build the angle sensor.

PWM Circuit

Originally not part of the theoretical design, the final design includes a PWM circuit. The pulse width modulation circuit transforms the angle data from a resistance into an electric signal which carries the angle information as high and low pulses of a square wave. Pulse widths directly relate to the measured angle through a Schmitt trigger switching RC circuit, the same one referenced earlier as an innovative solution.

The circuit works off a 5V supply and consists of an inverting Schmitt trigger with 2 looping branches; one branch charges the capacitor while the other discharges. The capacitor's charge oscillates between the 2 and 3 volt input threshold levels, resulting in inverted high and low levels of 1 and 4 volts at the output. Each rotary variable resistor controls the pulse width by directly changing the RC time constant. Simulation results reveal the magic behind the oscillating square wave as shown in figures 5,6.

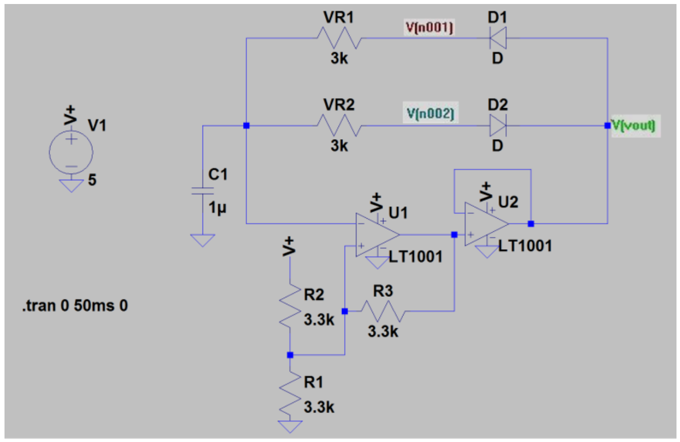

Figure 5: PWM circuit schematic simulated in LTSpice. Annotations added to match nodes to signal.

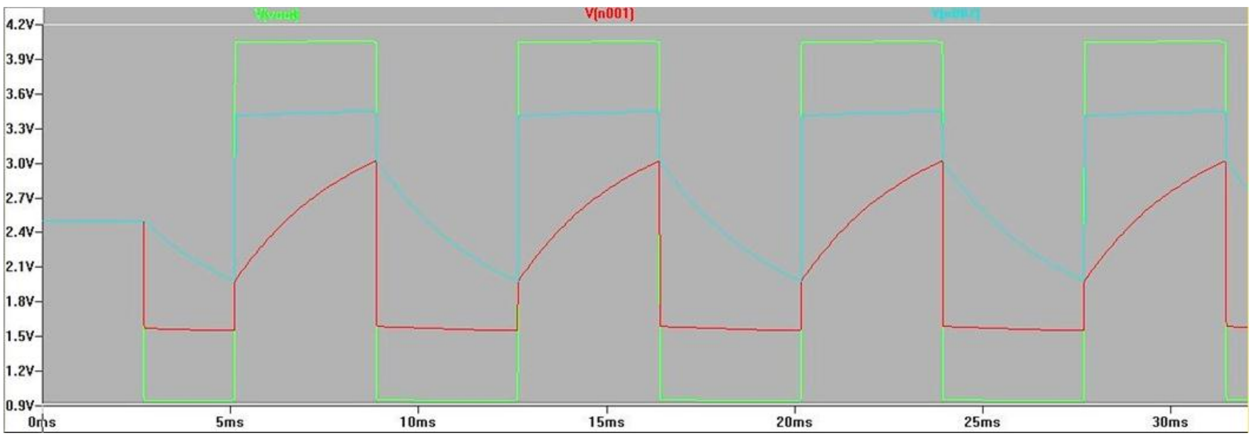

Figure 6: Resulting simulated waveforms for PWM circuit in LTSpice.

Designing the circuit to meet specifications was a bit difficult. Since the variable resistors are our rotary potentiometers that range between 0 and 2.95 kilo-ohms, the undetermined variables left to solve for are the capacitance and operating frequency. From the specs, a minimum 100Hz sampling rate was specified. The other limiting factor comes from the read time of the microcontroller. If the square wave oscillates faster than the MCU can read a high or low signal, then the project won't function properly and accuracy is lost. Therefore, each pulse width must be at least 1 microsecond in duration, given from Arduino Uno technical data that reading a port takes .76 microseconds. The following calculations show justification for the chosen capacitor value:

Using a 1 uF capacitor,

$$Vc = Vo * e^{-\frac{t}{RC}} \textit{equation for discharging capacitor}$$

$$t = -RC * \ln\left(\frac{Vc}{Vo}\right) \textit{solving for t}$$

$$\ln\left(\frac{Vc}{Vo}\right) = -1.3867 \textit{is a constant and will be replaced by its calculated value}$$

from simulation for better hand calculation accuracy

Therefore,

$$t = (3kohms)(1uF) * 1.3867 = 4.16\ milliseconds$$

$T_{MAX} = 2 * t = 8.32\ ms$ *worst case high + low time = max period*

$f_{MIN} = 1 / T_{MAX} = 240.38\ Hz$ *minimum sampling frequency*

f_{MIN} meets minimum 100 Hz specification. Simulation results calculate f_{MIN} = 132.1 Hz. Still meets the criterion! Further calculation determines the resolution the MCU can read from the angle sensors.

Sincemicros() function measures pulse width in multiples of 4us, the minimum noticeable change in angle must result in 4us change.

Therefore,

$\Delta t = 4us = 1.3867 * \Delta R * 1uF$ *inserting known values*

$\Delta R = 2.885\ ohms$ *solving for minimum noticeable change in R*

Since we know the minimum noticeable change in resistance, we can calculate the # of noticeable angle changes in our rotary potentiometer.

 # ofintervals$= Total\ R / \Delta R = \frac{2.95kohms}{2.885ohms} = 1022.7$

Angle resolution = $270°/1022.7$ = $0.264°$

The system notices 0.264° changes which meets our 5-15° angle measurement specification.

As a result from the calculation shown, a 1 microfarad capacitor works well enough for the PWM circuit.

To construct the circuit, a general purpose LM741 op-amp is used for the inverting Schmitt trigger configuration followed by a simple voltage follower configuration using a LMC662 dual operational amplifier. Lab testing on the final hardware implementation revealed the Schmitt trigger voltage levels as a low of 2.256V and a high of 3.337V. The resulting square wave at the output of the voltage follower found the levels to be a high of 4.3V and a low of 1.953V. Considering the Arduino input port determines high and low levels based on a 3V threshold, the analog to digital PWM circuit interface works. Given the angle sensors and PWM circuit are set up in their current configuration, regarding arrangement of connections between potentiometers and circuit board, the elbow sensor, when pointed straight, measured a max low pulse width of 2.8262ms and a min low pulse width of 1.39ms when the elbow makes a 90° angle. The shoulder measured a minimum high pulse width of 356.8us when the arm points 90° left of center and a max pulse width of 1.3682ms when the arm points 90°right

of center. Considering those angles are the max range the user's arms will move, the measured minimum frequency of the device is 170Hz. The assembled prototype circuit is shown in figure 7.

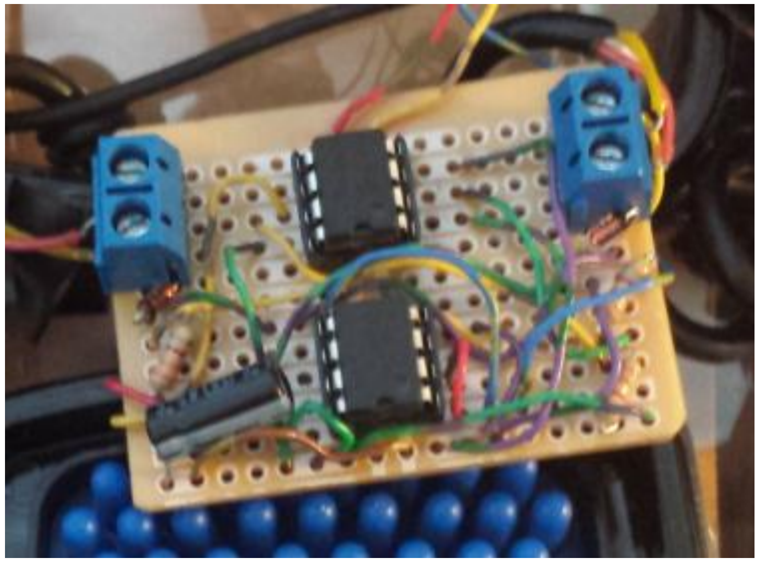

Figure 7: Photo showing the pulse width modulation circuit.

MCU Logic

Following the voltage output of the PWM circuit to the microcontroller, the final design as seen in figure 3 shows how the angle measurements are converted to a digital form. By measuring the high and low pulse widths, the MCU stores the durations digitally in a form that can make simple arithmetic calculations. In order to capture the necessary data at the right time, the MCU also determines when the user's stroke is in proper position by reading accelerometer data using I2C communication. The equations implemented in the code were determined from setting the limits on what constitutes a "correct" swim stroke that is not too wide and does not crossover. In the case that the right arm (the arm chosen for testing) is determined to be wide right, a couple of beats are sent to the left ear to signal the user to correct their arm motion a little to the left. In the other case when the user's arm crosses over the center axis of their body, beats are sent to the user's right ear to signal the user to correct themselves by straightening their arm or adjusting their arm position to the right.The equations for each are shown in the order discussed:

$$\textit{too wide when : } \angle Shoulder > 120°$$

$$\textit{crossing when : } \angle Elbow + 2 * \angle Shoulder \geq 270°$$

**Elbow angle is calculated digitally as 0 °at a right angle and 90°when fully extended.*

***Shoulder angle is calculated digitally as 0 ° when $90°$ left of center and $180°$ when $90°$ right of center.*

The MCU used for this project is an Arduino Uno. This particular microcontroller is well known for easy implementation into DIY projects and was already available at the time of picking an MCU. Since no unnecessarily complicated functions had to be implemented for this project's purpose, the Uno model was chosen. Coding in C also simplified the project. The programming follows a simple routine as shown in figure 8. The code created can be found in section C of the Appendices.

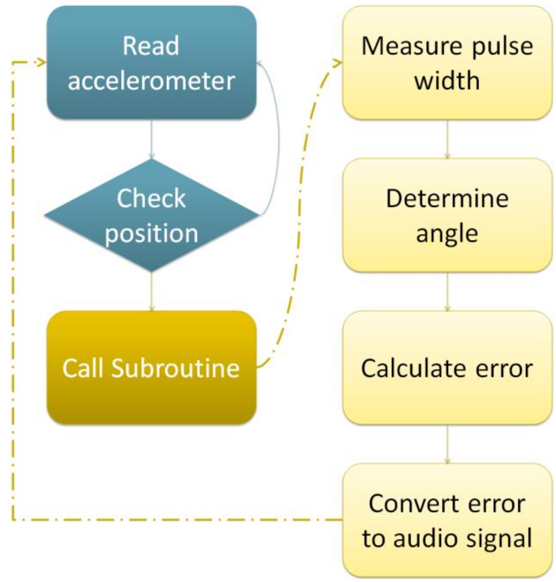

Figure 8: Program outline displaying process for determining user stroke correction.

Accelerometer

The final component of the function process is the accelerometer. Attached within the space of the microcontroller for space efficiency, the OSEPP ADXL345 Accelerometer is responsible for outputting the arm's acceleration the entire time. The accelerometer is set to measure +/- 2g in the X, Y, and Z axis in a 13 bit format. Since the accelerometer can detect the gravity vector, the arm's position can be identified. For the purpose of this project, the ideal position is found when the gravity vector lies entirely in the Z axis. From the serial data output to a computerduringsensor testing, the correct values for the ideal position were found. These values are checked by the MCU with an extended buffer range to allow the controller to determine if the accelerometer is in the general position. In order to transfer its data, the sensor communicates using I2C protocol which had already been provided by the company available for download and use on an Arduino Uno. This allowed for easy setup by hooking up only four cables to the MCU ports; Power, I2C data, ground, and I2C clock.

Waterproof Case + Headphones

To keep all the electrical components except for the angle sensors away from the water, they were all inserted into H2Oaudio's Amphibx Fit. The lightweight waterproof armband for iPhone and Large MP3 players keeps the objects stored inside separated from the water by a strong, durable, and flexible see-through plastic. The case also features a built in audio cable connected to a waterproof port on the outside, intended for use with waterproof headphones. The enclosure was able to fit the 9V battery that powers the arduino, the microcontroller with its attached accelerometer, and the PWM circuit board. The only downfall to the case is the fact that it only guarantees waterproof protection above 3.6 meters below the surface. Fortunately, the enclosure prevents the device from sinking to the bottom since it can float. A slight modification was made to allow the angle sensors to connect to the PWM circuit inside of the case. By hole-punching two holes in the plastic, the speaker wires go straight through the holes which afterwards are sealed using rubber and a silicone sealant. This process keeps the case as waterproof as before.

Along with purchasing H2O Audio's waterproof case, I bought X1's Surge Sportwrap Waterproof Headphones (X-1 was formerly H2O Audio so the two products are actually made by the same company but different name). The headphones and case were built specifically for each other. Altogether, these products proved to be easy to work with and an effective solution to protecting electronics from water.

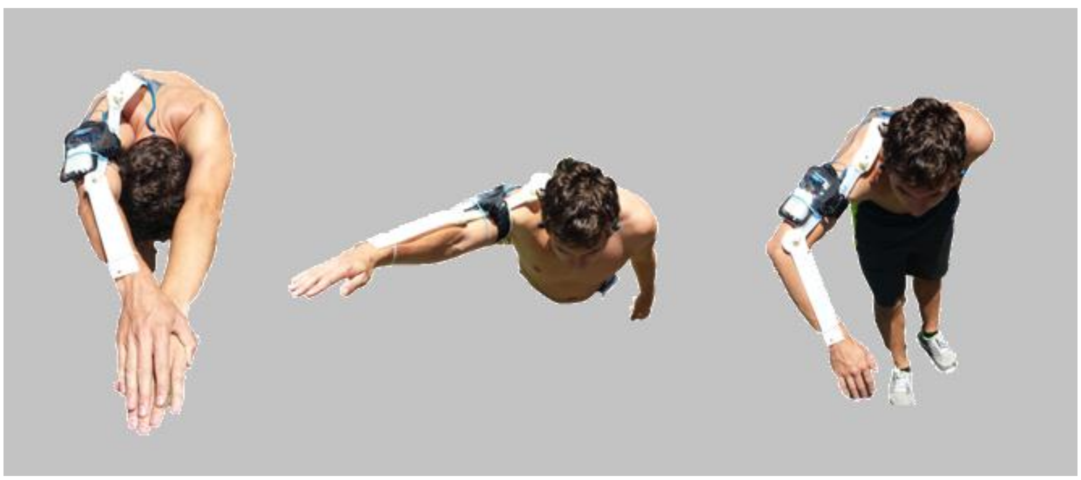

Figure 9: Real Time Swim Instructor completed and being tested.

Conclusion

Looking back at the original ideas and the final implementation, the project's direction of development meandered in the beginning but settled down on a straight course over time. Some of the specifications originating from 8 months before completion of the final design don't make sense but the rest helped determine the final design's component values. The following is a review of each original specification.

The first specification called for a functioning wireless system. Early thoughts on the project leaned toward a device that would require heavy computational power, therefore requiring to design a form of communication between a computer on land and the device in the water. Eventually, the design moved past the need for an actual computer once the scope of stroke improvement became limited to fixing a stroke that crosses over. Technically, this engineering specification is met since the Real Time Swim Instructor can be used anyplace as long as the battery is supplying sufficient power.

The next spec required the system to recognize 5-15 degree shifts of arm position to within a degree of actual position. Extrapolated from self-measurement and experimentation, the proper position of a swimmer's arm varies from person to person, thus allowing the arm to be located in a general area between 5-15 degrees. Earlier calculation from the PWM circuit portion of the report determined the device can recognize .264 degree shifts in position, satisfying the accuracy requirement of recognizing correct arm position with less than 1 degree error.

The third requirement entailed making the device waterproof to at least a 5m depth. This spec stemmed from the thought that if the device were to fall off in the pool and sink to the bottom, it would need to withstand the water pressure at the bottom of the pool, which in some pools can reach 5 meters of depth in the so called deep ends of the pool. In a way the device both satisfies and fails this requirement. The waterproof enclosure that contains all of the electrical components is limited to 3.6 meters underwater, which simply is less than the specified 5 meters. On the other hand, the case floats and would prevent the device from ever sinking to 3.6 meter depths, let alone 5.

Competitive swimming measures race times down to the hundredths of seconds; therefore it seemed suitable to specify the system should at least operate at a 100 Hz sampling rate. This means that updated data should be available every .01 seconds. Following this guideline, a range of acceptable capacitor values were determined. The final lab testing of the output waveforms worst case proved that the minimum operational frequency of the system was greater than 100 Hz at 170 Hz.

Knowing Arduino's could supply 5V, the next spec stems from the idea that the entire system had to work off a 5V source. Ideally, a 9V source specification would work better since 9V batteries store more energy and fit in easily into packaging due to their boxed shape. Additionally, 5V batteries are not readily available. In order to construct a 5V supply, several AA batteries and a special housing would be needed to construct the 5V as a series of batteries. Such a design would be inefficient. In the final implementation, a 9V battery sufficiently satisfies the power demands. Updated specs for future devices will be called upon to work off 9V supplies.

Going back to the idea of using a computer, in the event that the swimmer wanted to record their stroke data, another specification asked to provide live user feedback and access to the last 5 minutes of collected data. The collected data would have been available for use to create plots or models that could graph the stroke improvements. Nevertheless, the design was simplified to providing live user feedback, which would effectively close the control loop and ensure the user learned proper hand placement in the water.

The final specification called for the device to be wearable with a non-obtrusive hydro dynamic profile that kept the device with a half inch distance of the skin. The goal was to make sure the swimmer could feel as comfortable swimming with the device as without it. Unfortunately, dimensions of the prototype do not meet the requirement as certain areas feature a 1.5 inch layer of hardware. Future improvements would warrant the assistance of a mechanical or industrial engineer to design the mechanics and product dimensions.

Overall, the prototype proved that a real time swim instructor is feasible. Although not all of the initial specifications were met, a better grasp of more important design factors was attained. With the prototype finished and working, the future of the project focuses on improvements that can be made and additional features that may be added. The design and construction of the Real Time Swim Instructor is a demonstration of my capabilities and a capstone of the knowledge acquired during my undergraduate tenure at Cal Poly. This experience encompasses what it truly means *to learn by doing*.

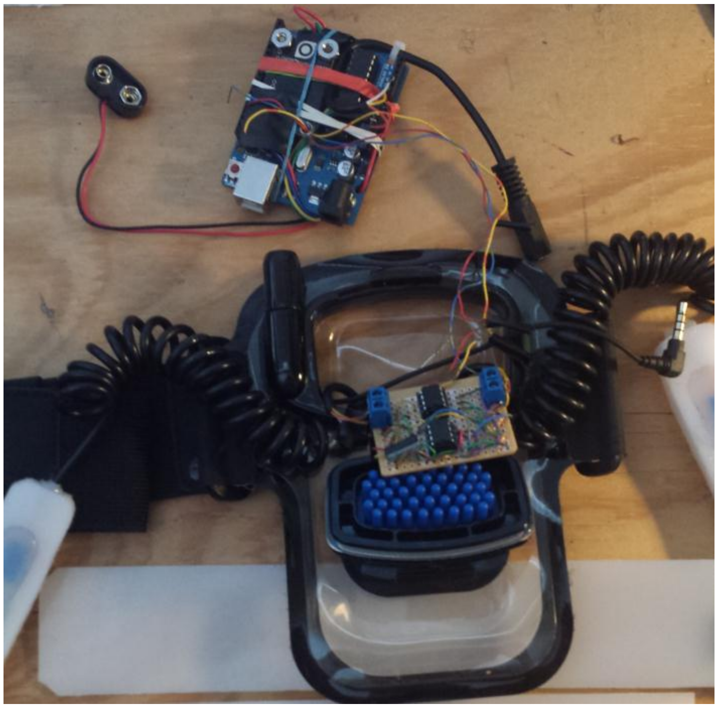

Figure 10: Final design with all of the components connected outside of the case.

Appendices

A: *Senior Project Analysis*

1. Functional Requirements

   The device needs to function wirelessly from a power source. The device sends feedback while the swimmer is swimming to provide stroke correction while the person is swimming. The device lasts longer than a couple hours since elite practices last between 2-3 hours in the pool. The device accurately measures body positions and angles in the water from 5 to 15 degrees to determine the swimmer's exact stroke perfections. Additionally, the device is self-sustainable through at least 12 hours of practice before the batteries require replacement.

2. Primary Constraints

   The device needs to function underwater. The microcontroller needs to function within a waterproof casing. Choosing sensors that function underwater are also required. Determining the best approach to correcting swim stroke also takes time since there are many ways of sending feedback to the user. In addition, the device allows free, unrestricted movement, and unhampered swimming motions.

3. Economic

   Both casual and professional swimmers would be interested in purchasing this device since stroke perfection is sought after by all amateur level and above swimmers. This device would be used every swim practice.Thus, items used routinely convincebigger club teams to purchase the product. The most expensive project parts are waterproofing the electrical components and determining the feedback system, which can range from audio feedback to nerve excitation. Additional equipment for the project can include safety utilities for the feedback system.

Table 5: Cost estimates.

Item	Quantity	Price
Potentiometers	3 @ $25.00	$75.00
Sensors	1 @ $20.00	$18.00
Microcontroller	1 @ $36.00	$36.00
Water proof case	1 set @ $7.00	$7.00
Waterproof headphones	1 set @ $37.00	$37.00
Labor	80 hours @ $15.00	$1200.00

Total Cost of parts:	$173.00
+ **Cost of labor:**	+ $1200.00
Total:	$1373.00

4. Commercial Manufacturability

USA Swimming's nearly 300,000 members make an excellent target group for marketing the Real Time Swim Instructor. A safe estimate for the number of devices sold per year is around 400. at this point manufacturing costs per device average around $250. Customers can purchase the product for $300resulting in a $50 dollar profit margin per device. The yearly profit estimates to about 20,000. Additional user costs include electricity for powering the device. An exact source for power isn't determined yet but users can expect battery life to last at least 6 hours of continuous use.

5. Environmental Impacts

The environmental impacts associated with manufacturing include the continued consumption of fossil fuels in developing semiconductor devices. Computers powered by electricity acquire power from coal burning plants. Also, batteries which power the device are reported to be detrimental to the well-being of the environment. This product indirectly keeps water consumption at unnecessary levels since swimmers require pools filled with hundreds of thousands of gallons. Water is essential to all living things and continued use of water for non-essential purposes carries negative consequences to the depleted environment.

6. Manufacturability

The biggest factor in manufacturing the product is waterproofing the device. Water has conductive properties that interfere with current flow and have a high potential to harm electronic circuits. Today's technology implements an insulating surface around the electronics to keep water out but any damage to the casing has the potential to let it leak and destroy the device. Therefore finding a suitable, safe, and reliable way to waterproof the device is a necessity and has all to do with manufacturing engineering.

In addition, the system has several subcomponents spread out on the user's upper body. Pieces include angle sensors for the arms, beat makers by the ears, and a MCU at the midpoint of all other components. Manufacturing a spread out system challenges the packaging process and manufacturing order. The device fits people of different sizes so assembling a universal tool requires adjustable components. Developing components with varying lengths pose challenging questions in determining maximum and minimum lengths.

7. Sustainability

The device requires low maintenance, because it only needs to be charged. The rest of the device functions are meant to help improve swim stroke. The device affects the sustainable uses of electricity and water since electrical power keeps the device working and swimmers will want large bodies of water to swim in to use the device for its intended purpose. The best upgrade available for the current design includes installing video feedback to the swimmer through their goggle lens. Unfortunately, video feedback requires fitting a tiny screen into the

lens which is hard enough to implement without mentioning the complexity and processing power for such an upgrade.

8. Ethics

Based upon the framework of utilitarianism, my device provides for the greater good. This project allows people, who want to become better swimmers, achieve that goal, and when more people can achieve their intermediate goals, our society becomes happier. On the other hand, this project continues the use of swimming pools which as mentioned earlier have a high water demand. In the future, if we do not find a sustainable source of water, the environment around us will crumble, ecosystems will die, and that's when this project indirectly contributes to the deterioration of our society, serving the masses for the worse.

In respect to the IEEE Code of Ethics, as a contributor to this project, I accept the responsibilities of providing a safe product, improving my technical competence in understanding the technology I am working with, and fully disclose all relevant information honestly and to the best of my abilities.

9. Health and Safety

Some of the technology that can be implemented in the final design provides feedback to the user. The form of this feedback has to interact with the user and work with their senses of feeling, seeing, or hearing. There may be reason to be cautious of health concerns with that part of the project but until a final design is chosen that adheres to those safety concerns, the public can assume this device is safe. Additionally, as with all products that can be used in a body of water, this is not a lifesaving device so do not depend on it.

On the other hand, the device keeps people swimming, thus promoting physical fitness. Swimming provides both aerobic and anaerobic exercise while the body experiences low impacts in the water. Compared to basketball, running, and all physical sports, swimming has the lowest harmful impact on the body since the effects of gravity on the body are weaker in the water than on solid surfaces.

Additionally, an electronic swim instructor teaches people how to swim better. This enhances the swimmer safety in the water and users involved in emergency situations in water. The more experienced a swimmer is, the easier that swimmer can save a person in distress in the water.

10. Social and Political

The direct stakeholders in this project are the swimmers and coaches that purchase the product. In exchange for tradable currency, these stakeholders will be able to continue their swimming career with assistance from the Real Time Swim Instructor. That assistance will result in time saved practicing with an inefficient swim stroke and the continued attention to improving technique. This device potentially increases the swimmer's chances of outperforming the competition, thus possibly creating an inequality between swimmers with access to the device and those without. The potential inequality would depend on economic power, if the

device isn’t affordable, and the political power of the coaches, their decision in providing the device to their swimmers.

11. Development

Most of the research focused on finding existing waterproofed, wearable,electronic devices. I discovered that there isn’t a lot of information available on the subject and that it took effort to search deeper. The fields of wearable electronics and underwater devices have room for research and development that go further than water proof watches. In order to figure out how to adequately protect my circuit from the water, contacting experienced manufacturers will aid the success of that deliverable.

My 15 years of swimming experience benefit the programming and angle measuring parts of the project. The best knowledge comes from firsthand experience so when designing those parts, I know where the arms need to be positioned and where they’re going. Understanding the entire process helps in coding a tuned device that will measure a wide range of values. Without the prior swim experience, understanding what to program would require a lot more time for research.

Additionally, a helpful find came from an electronics design website. Using two resistive sensors, a Schmitt trigger, and a capacitor, two sensors could be measured by the same input. This benefits the project by presenting an alternative to additional wiring and complexity.

B: *Gantt Chart*

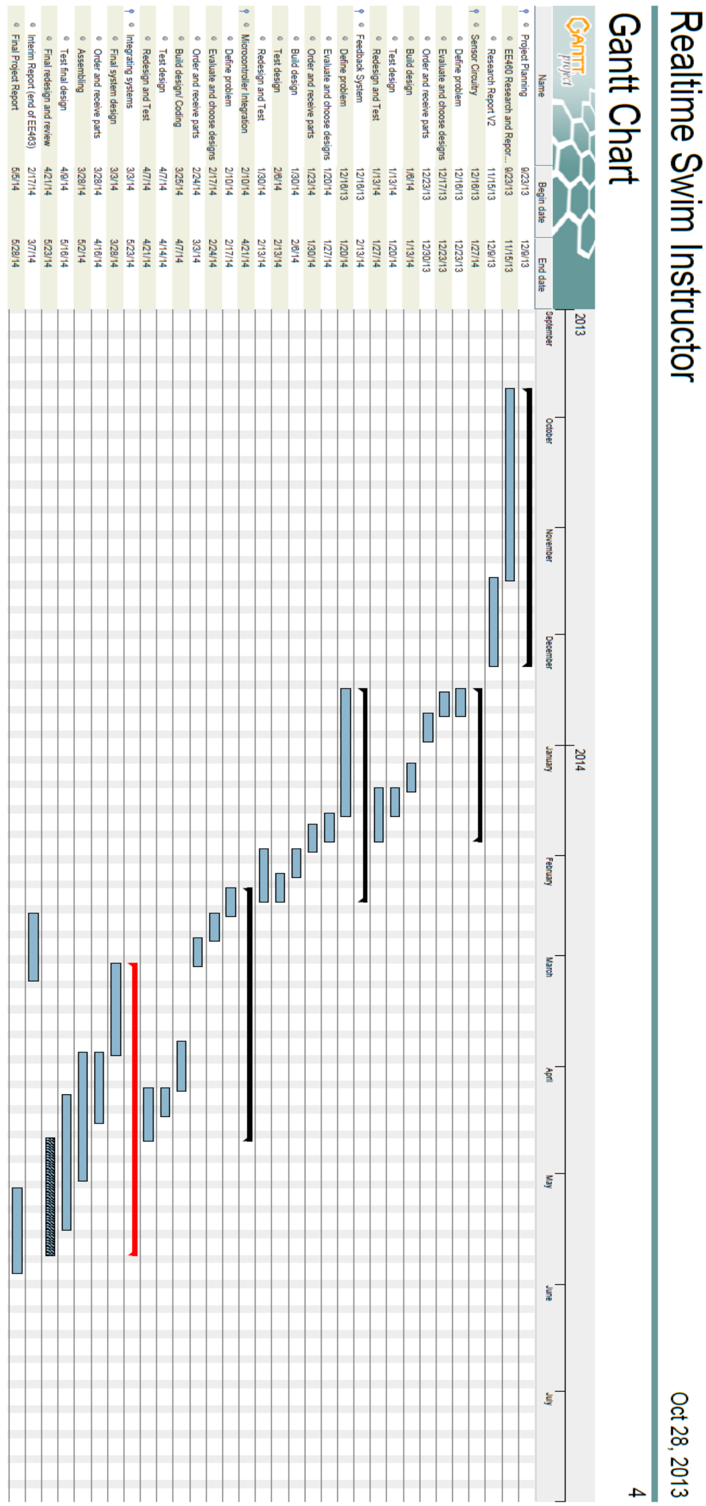

Figure 11: Gantt Chart

C: C code for Arduino Uno

```
/**
Title: Real Time Swim Instructor Code
Programmer: KonradAntoniuk
Version: 29 May 2014
*/

#include "Wire.h"
#include "I2Cdev.h"
#include "ADXL345.h"

ADXL345 accel;

int16_t ax, ay,az;

#define PWMpin 12
#define NOTE_C4  262
#define beatDuration 4
#define signalLeft A1
#define signalRight A3

void setup(){
// join I2C bus
Wire.begin();

//initialize serial communication
Serial.begin(38400);

//initialize accelerometer sensor
Serial.println("Initializing I2C devices...");
accel.initialize();

// verify connection
Serial.println("Testing device connections...");
Serial.println(accel.testConnection()?"ADXL345 connection
successful":"ADXL345 connection failed");

// pin mode outputs for headphones
pinMode(signalLeft, OUTPUT);
pinMode(signalRight, OUTPUT);

}

void loop(){
//read accelerometer
accel.getAcceleration(&ax,&ay,&az);

// display tab-separated accel x/y/z values
Serial.print("accel:\t");
Serial.print(ax);Serial.print("\t");
Serial.print(ay);Serial.print("\t");
Serial.println(az);

// check position
if(ax <75&& ax >-75&& ay <75&& ay >-75&&az<-230&&az>-330)
runSubRoutine();
```

```
}

voidrunSubRoutine(){
//find arm position
intelbowAngle=determineAngle(LOW);
intshoulderAngle=determineAngle(HIGH);

determineError(elbowAngle,shoulderAngle);
}

intdetermineAngle(int part){
if(part == LOW)
returnelbowCalculation(pulseIn(PWMpin, LOW));
else
returnshoulderCalculation(pulseIn(PWMpin, HIGH));
}

intelbowCalculation(unsignedlongpulseWidth){
//calculations using numbers from testing
float temp =(pulseWidth-1390)*90.0/1436.0;
return(int) temp;
}

intshoulderCalculation(unsignedlongpulseWidth){
//calculations using numbers from testing
float temp =(pulseWidth-357)*180.0/1011.0;
return(int) temp;
}

voiddetermineError(intelbowAngle,intshoulderAngle){
//shoulder angle error case
if(shoulderAngle>120)
playBeat(signalRight);

//crossing over error case
if(elbowAngle+2*shoulderAngle>=270)
playBeat(signalLeft);
}

voidplayBeat(int port){
for(int i =0; i <3; i++){
tone(port, NOTE_C4,beatDuration);
delay(beatDuration);
}
}
```