Health and Fitness Tracker For Canine Pets

Hai Nguyen, Matt Horton, Dominic Vu

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — In today's day and age, everyone's concern for health and longevity is increasing and consumers are looking at more ways to enhance their quality of life. A form of enhancement are fitness trackers which are wearable devices that provide the ability to track important vital signs on the go such as heart rate, sleep patterns, calorie intake/output, etc. The problem we see is that the functionality has yet to extend to our pets such as dogs and although there are some options available in the market, what is out does not have much functionality and not optimal. We want to tackle the issue at hand by creating a device for a dog that has the ability to read/record a dogs (1) Heart Rate, (2) Temperature, (3) Steps They've Taken In A Day as well as a (4) GPS locator; all while being able to surface this information to a phone application that will be easily accessible as possible. This will also mean that the device will require Bluetooth capability to communicate to the phone application. The paper will also go into detail about how and why we chose these functions for the device/application.**

***Index Terms* — Bluetooth, Global Positioning System, Accelerometers, Temperature measurements.**

I. Introduction

Every modern day fitness or health tracker that you can purchase on the market right now gives you the basic vital signs that can determine (to an certain extent without the presence of a physician) one's overall health. For example, the iWatch can provide your heart rate, the amount of steps you've taken in a day, your sleep patterns, and much more. If you search on the market for something on the market for you dog though, you will find that your options are very limited and the options that you do have do not provide much functionality. The purpose of this paper and this project is to provide the research, knowledge, and implementation of a fitness/health tracker for a dog and its benefits. After extensive research and advice from a practicing veterinarian, we came up with four functions that would accommodate our needs in providing a useful device for a consumer.

A Heart rate monitor is the first component because it is the most basic function of any device out on the market. Also, one of the most important vital signs that a Veterinarian takes to determine a dog's health is the heart rate. The consumer will be able to read the heart rate on the fly which enables them to take it when the dog is resting or when the dog is active. If the heart rate is too low or too high, the user will be notified through the phone application. The average heart rate (defined by the Veterinarian) during rest is 100-160 BPM. This also depends on their weight, so we will read in the heart rate and the weight (which is defined by the user) and determine if the heart rate is out of range and should contact a professional. This is pretty much on par with all the fitness and health trackers for dogs that are on the market now.

The temperature sensor we've decided to add to the device will also be able to be read on the fly. This is very beneficial for our device and consumer because 1. the market does not provide an all in one device that supports heart rate monitoring and temperature readings and 2. the most common form of temperature reading is through the rectum so we are providing a much less invasive and cumbersome way of acquiring an important vital sign of the dog. The average temperature of a dog is close to that of a human which sits around 101F-102.5F and like the heart rate functionality, we will alert the user if the temperature is out of range and contacting a professional may be necessary. The placement of the temperature sensor will be in the armpit of the dog as that is second most common and accurate way to read a dog's temperature. It is also the most stable and situated position we can put the sensor as we have taken into account the activeness and movement of the dog.

An accelerometer is a great addition to the device as it is a parallel for a human and dog. As most human fitness / health trackers, a step counter is provided as standard functionality and if we can provide the same for a dog, it makes it an interactive and a known functionality between the human and the dog. We will display the steps a dog takes on a daily basis and will reset every 24 hours. We will also alert the user if the dog is not active enough as the average amount of steps a dog should take is _ which is defined by a Veterinarian. We will place this on the back of the dog.

With all this technology on the dog as once, we thought it would make the most sense to include a Global Positioning System on the device. Fifteen percent of pet owners (dog or cat) have lost their pet at least once in the past five years across the United State. 85 percent of those pets were recovered. It is our vision that if every pet in the country had the harness on, the recovery rate would increase up to 100 percent.

To put it all together, we are including Bluetooth functionality to be able to communicate with the phone application. There were a lot of options we had when deciding how to surface all the information to the consumer, for example, lcd screen, on the actual device, uploading the information to a cloud based database system where we can extract and display the information to a website, or simply creating an app that the user can interact with the device real time. The LCD screen posed a lot of concerns such as price, usability, and integrity of the device under stress and rugged use. Also, the cloud based method does provide an easy-to-use methodology as they user would need to access the internet whenever they wanted to acquire knowledge on the dog's reading.

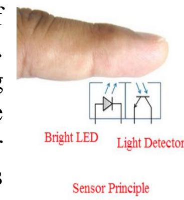
II. System Components

A. ATMEL ATMega328P Microcontroller

The ATmega328P is a low-power C-MOS 8-bit microcontroller based on the AVR enhanced RISC architecture. This microcontroller executes instructions in a single clock cycle, and ATmega328P achieves throughputs approaching 1 MIPS per MHz which is useful in allowing the system designer to optimize power consumption versus processing speed. The ATmega328P chip comes with many different features. Features such as: 32 KB of In-System Programmable Flash with R.W.W. capabilities, 1 KB of EEPROM, 2 KB SRAM, 23 general purpose input/output lines, 32 general purpose registers, three timers and three counters with compare modes, internal/external interrupts, a serial programmable USART, a byte-oriented 2-wire SI, an SPI serial port, a 6-channel 10-bit ADC, a watchdog timer with its own internal Oscillator, and many different power saving modes.

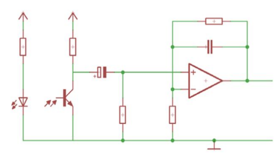
The microcontroller is at the center of the design. The different criteria that were important to the design of our project helped us narrow down which MCU we chose for our Smart Harness. The main factors to consider were the low voltage, ample memory, sufficient number of pins and inclusion of a board for testing purposes. Taking into account all these reasons we developed our Smart Harness using the ATMega328P chip which was accompanied by the Arduino Uno development board. This chip was Arduino compatible and allowed us to complete all of our coding tasks much more easily than we would have otherwise. Seeing as though this chip is the central hub of the hardware design it was acceptable for our group to have this particular component cost a little more as the quality that comes with it is of high importance.

B. Heart Rate Sensor SEN 11574



A basic heartbeat sensor consists of a light emitting diode and a detector. This detector can be a light detecting resistor or photodiode, and when the tissue of the body such as a finger or earlobe is hit with the light, it is reflected. The light is then received by the detector and based on the reflection, the detector will output an electrical signal that is proportional to the heart rate.

The heart rate sensor we selected for our design was the SEN 11574. The sensor itself was small and would be easy to conceal in the harness. Among the many different sensors we looked at this particular item had the lowest current draw at only 4mA, far less than the other sensors. This would allow us to keep our power consumption by the sensor at a minimum. The sensor was also accompanied by a long cable, which allowed for easy placement on the harness. Lastly, the sensor was only a fraction of the price of the other sensors, priced at only $5.00.



C. Temperature Sensor DS18B20

The main use of the DS18B20 is its direct-to-digital temperature sensor. It uses a digital thermometer and this type of thermometer is mercury free. Digital thermometers contain a thermistor inside them which is used to obtain the temperature. A thermistor is a thermal resistor that changes its impedance when the temperature changes. The accuracy of the temperature sensor is customizable. There are four different accuracy settings. The bits 9, 10, 11, and 12 bits are customizable and the corresponding accuracies are 0.5°C, 0.25°C, 0.125°C, and 0.0625°C. Upon startup, the initial setting is 12-bit, or 0.0625°C.The DS18B20 turns on in a low-power idle mode, and to initiate a measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Once the conversion has taken place, the measured data is then stored in a 2-byte register and the temperature sensor returns back to its initial state.

The DS18B20 measured data is initially set to output in degrees Celsius; for Fahrenheit readings, a lookup table or conversion routine is required to be used. The measured information is stored in the memory as a 16-bit sign-extended two's complement number in the temperature register (see Figure C). The sign bit, shown using the letter S, is used to show if the reading is positive or negative: for positive numbers S = 0 and for negative numbers S = 1.

Based on our requirements and the comparison between MCP9700, MCP9808, and DS18B20, we decided to choose the DS18B20. An important factor in our decision was that this particular temperature sensor was designed to be waterproof. Since this sensor will be in constant skin contact this was vital. Another important factor was the cable was quite long. This saves our group the trouble of needing to extend the wire so that it reaches the correct location on the dog's body. Finally, the power consumption of this device is quite low at a range of which is from 27µW - 45µW which will help keep our overall power consumption at a minimum.

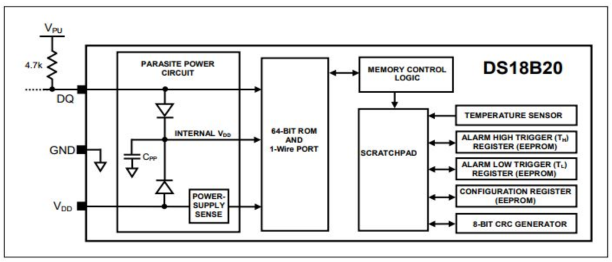


Figure C DS18B20 block diagram

D. Adafruit MMA8451 Accelerometer

An accelerometer detects motion in three-dimensional space. There are many types of accelerometers, according to there are some use the piezoelectric effect. This means they contain tiny crystals that get stressed by acceleration, which causes voltage to be produced.

The Accelerometer we used for our project was the Adafruit Triple-Axis Accelerometer MMA8451 module. Some of the features of this accelerometer are I^2C digital output interface (operates to 2.25 MHz with 4.7 kΩ pullup), Output Data Rates (ODR) from 1.56 Hz to 800 Hz, ±2g/±4g/±8g dynamically selectable full-scale, 14-bit and 8-bit digital output and current consumption: 6 µA to 165 µA.

Many of the accelerometers have very similar functionality and precision as well as voltage ranges that were fairly close. For the project the group chose this module because, while not the cheapest, it was the the best fit due to its higher precision coupled with a moderate cost as well as meeting our requirements. It also has added feature of being Arduino compatible which made our software much easier to design.

E. Global Positioning System Fona 808

GPS is a device which is using four satellites to locate the position of an object on earth with time, longitude, latitude, number of satellites, and altitude. GSM (Global System for Mobile Communication) is the device is used to communicate with mobile phone. In this project, we will use both GPS and GSM as an add on function in order to locate the position of the harness in the situation if the dog is get lost. We will use the device Fona808 for our project to track the harness location.

First of all, Fona 808 is designed by Adafruit Industries integrated with both GSM (Sim808) for 2G mobile communication, and MT3336 chipset for GPS tracking with 165 dBm tracking sensitivity which is needed for our project. SIM808 will send SMS or MMS to the mobile phone with the longitude from MT3336 to get the location of the device.

Secondly, Fona 808 will have a separate battery for GSM because during the active mode the GSM can consume up to 2 Amp which based on design for SMS. Meanwhile the GPS will using 20mA continuously from the battery on PCB according to Adafruit Industries.

Finally, because of the time limit, our group decide to spend $75.85 dollars from $100 dollars budget to build the add on GPS tracking system on https://www.adafruit.com.

F. SH-HC-08 Bluetooth Module

A Bluetooth module is a module that uses radio waves to transmit data from one device to another. The SH-HC-08 is a Bluetooth UART communication module. A universal asynchronous receiver-transmitter (UART) is a computer hardware module for asynchronous serial communication in which the data type and transmission speeds are configurable. The module also uses Bluetooth specification version 4.0 BLE Bluetooth protocol. The wireless frequency is 2.4 GHz ISM, modulation is GFSK. The maximum transmit power is 4 dBm, and the receiving sensitivity is -93 dBm. It also has been able to achieve 80 meters of super long distance communication using an IPhone 4S and an open environment, but for the purposes of our project, the strong 8 meter range will be sufficient as we are not trying to test the limitations of the module. The module also has a size of 26.9mm x 13mm x 2.2mm and uses the CC2540 chip.

This module had the lowest current draw of all of the modules which were researched with a value around 9mA. We also wanted our module to be compatible Arduino, and this item accomplished that. This made our coding much simpler as we did not have to start our algorithms from nothing. This module allowed us to gather all the information that the sensors measured and send them to our mobile device. The information was then used in our mobile application and was viewable by the user.

G. 9V Alkaline Battery

The battery we chose for this project, was a 9V alkaline battery. This battery can be located in any store and would provide sufficient battery life for the design. This allowed for 15+ hours of functionality and will be paired with a voltage regulator. This will step down the voltage so the components can operate safely without being damaged. In the future, other battery options to extend the life can be implemented by changing the battery and regulator, but the for the purposes of our project the 9 V was a cheap and effective option.

H. Circuit Enclosure

To providing a safe enclosure was important for our final design. The enclosure we used was obtained through eBay. It had dimensions of 5”x2.5”x1”. It is made of durable ABS plastic and will allow for the PCB to be mounted inside. To allow the sensors to safely take measurements we had to drill holes into this enclosure. Inside the holes were placed gland joints or cord grips. The gland joints used were PG7 and will help keep our PCB safe from water and dust.

I. Harness

The design required a fabric harness to place our design on. A tactical harness was the best option as it has durability and plenty of surface area to mount the device.

III. System Concept

With the exception of the fabric harness, enclosure and the GPS module, all of the different hardware parts will most likely be incorporated into the PCB. The heart of the design is the microcontroller in the center of the flowchart. This module will process all the data that comes from the peripherals. The ATMega328P chip that was chosen contains 28 pins that will be used to connect that different components. The components that were placed on the PCB along with the microcontroller are the bluetooth, GPS, and the accelerometer.

The design will require a clock signal to run. For this purpose a 16MHz crystal was incorporated into our design. This crystal will also be added to our second PCB design and was the main reason our initial PCB design failed. During our first prototype it was discovered that our design had been lacking a clock signal when it was separated from the development board, but once we added in the clock the design worked as it was intended to.

The overall design flowchart is shown in Figure 1. This shows all of the components that were included in our design. The PCB contains the MCU, bluetooth, GPS, and accelerometer. All of the other hardware components will be connected to the PCB or included in the circuit enclosure, except the mobile device.

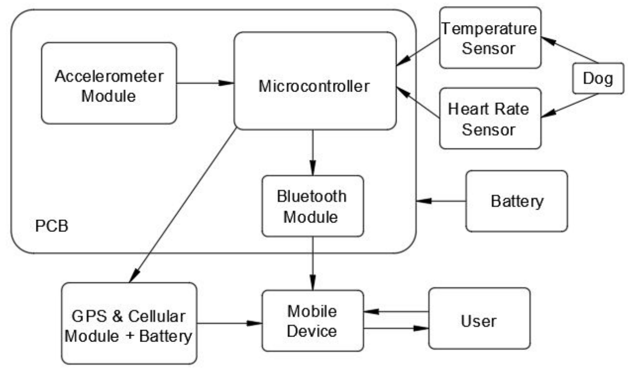


Figure 1: Overall Design Flowchart

IV. Hardware Design

The power source of this project we will be the 9V Alkaline battery because of the weight, and the convenience to replace the battery. Because all of the sensors and devices on the PCB use 5V as the power supply, we have to convert 9V to 5V. Moreover, We will use the Buck converter (MP1584en) instead of the linear voltage regulator (L7805CV) from our initial plan.

There are two main reasons why we use the buck converter. First of all, the buck converter will save our battery energy by 90%; meanwhile, the linear voltage regulator will waste a lot of energy from the battery. The formula is calculated by

$$P_{loss}(L7805) = Vin*Iregulator$$

Vin is 9V, Iregulator is 0.8mA. The total power loss is 11.25mW. According to Texas instrument, MOSFET is reduce the power loss of by 100%.

The second reason is the temperature. Linear voltage regulator can release really high temperature. during our test, the temperature of the L7805CV it go up about 135ºC and the device turn off after it cool down, the L7805 work again for every 60 second. As for MP1584en, the temperature keep constant at 45ºC. As a result, our group chose to use the MP1584en as our Buck converter from 9v to 5v for our PCB.

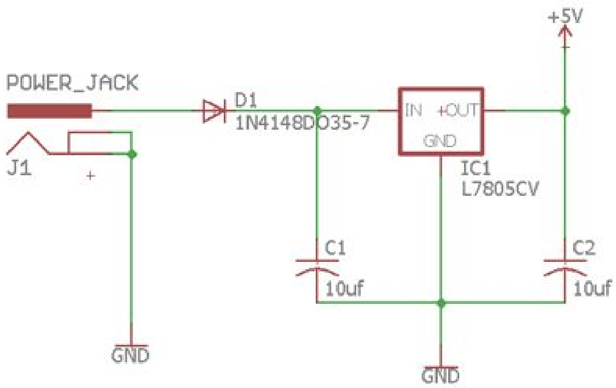


Figure 2: 5 Voltage Regulator Schematic

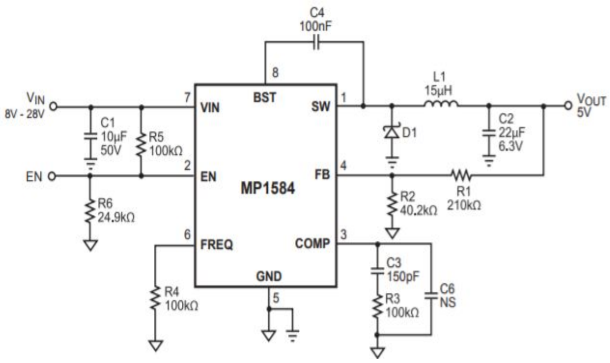


Figure 2.1: Voltage Switching Regulator Schematic from MSP 1584 data Sheet.

When the 9V battery was chosen for this project, we were not focusing on the length of time the battery would last. We were more focused on cheap and reliable power options. While the life of the battery isn't extremely long, it is sufficiently long enough for the purposes of our project. If this design were to go into production, the power system would have to be changed. This would not be a huge undertaking but it should be noted for the future. The battery life can be calculated using the following formula:

$$Battery\ Life\ (Hours)\ =\ \frac{Total\ Capacity\ (mAh)}{Actual\ Current\ Consumption\ (mA)}$$

The total capacity of a 9V alkaline battery is 550mAh. Using the following table, Table 1, the total amount of current consumption can be calculated.

Component Name	Current Consumption (mA)
ATMega328P (16MHz crystal)	17.52mA
DS18B20	1.5 mA
SEN 11574	4 mA
SH-HC-08	9 mA
MMA8451	.165 mA
Resistors, Diodes, etc.	10 mA
GPS(MT3336)	20mA
Total	62.185 mA

Table 1: Current Consumption (Active Mode)

Based on these numbers, the battery will take about 9 hours before it is no longer able to supply power to the device. The entire power system will include the 9V and a switching voltage regulator.This component will be used to step the voltage while also keeping the device from getting to hot. The 9V battery will supply the power to the PCB, which will distribute the power to the different components. Each of the components require 5V to operate, and the switching voltage regulator will step down the voltage so that none of the components will be damaged by the 9V power supply.

The GSM will be using a separate battery system (lithium ion polymer 3.7v 1200mah) which comes with the Fona 808.

Component	Current Consumption (mA)
GSM (Sim808)	25mA
Hours	< 48 Hours

Table 1.1: Current Consumption Fona 808 during low mode activity.

V. User Manual

One of the main goals of this project was to make this harness as easy to use as possible. When the harness is initially utilized by the dog owner, some information will be needed and the user will be prompted to enter this data accordingly. Some of this data will include weight, age, known health issues, and other factors. Once completed, the user should be able to access information about their pet easily and quickly.

Step 1: Pair the Device Before starting the application, the user will attempt to pair their mobile device with the Smart Harness. The Smart Harness will be Bluetooth enabled and should pair easily with an Android device easily (Apple devices may pair, but the app. may not work properly). The user will be notified of the success or failure of pairing and at that point the user would be directed to the setup of the application or prompted to retry to connect.

Step 2: Input Data Once the connection is established, the application will prompt the user to input specific data about their dog. The information that will need to be provided will include but will not be limited to: the dog's name, weight, sex, age, color, breed and other notes that would be beneficial such as health issues or medication that the dog is on. All this information will be added to and displayed on the dog's profile.

Step 3: Safely Strap on the Harness The next thing that the user must do is to make sure that the harness has been fitted to the dog properly and is powered on properly even after bluetooth pairing. If the dog is in danger of being shocked or harmed in any way or if any of the electronic equipment is exposed in an unsafe manner, REMOVE IMMEDIATELY. We all care for our pets and do not want them in any kind of danger.

Step 4: Sensor Placement Once it has been fitted safely, make sure the sensors are placed in the correct locations so that the information will be detected properly. If the information from the sensors is not read correctly, you may get inaccurate information about the dog's health. For instance, if the temperature sensor is improperly attached, you may not know if your canine has a fever. Proper calibration is pertinent to accurate readings.

Step 5: Enjoy the Harness After you have input the data, equipped the dog with the harness, and placed the sensors properly, the data should now begin to read. This data will include the heart rate, temperature, and the amount of steps the dog has taken. You should also be able to track the location of the dog with the GPS. Now you can track the health of your dog!

VI. Embedded Software

The ATmega328p will be programed with the arduino IDE before it will get integrated on our PCB. inside the code. it is divided into four sections which including Bluetooth, Heartbeat sensor, Temperature sensor, Accelerometer, and GPS tracking. The diagram below will explain how the code flow inside ATmega328p chip.

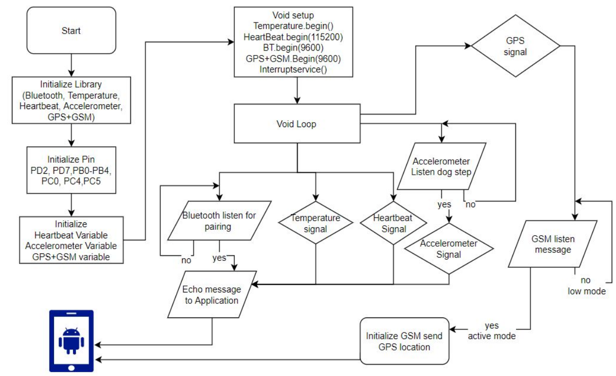


Figure 3 Arduino Code Flow Chart

The flow chart above represent step by step of the code how it will operate inside the ATmega328p. the chip will keep running and send the temperature, heartbeat, and accelerometer signal to the application through bluetooth device. Meanwhile, the GPS and GSM will listen to the message from mobile SMS or MMS in low mode activity until it receive the SMS or MMS the GPS and GSM will turn on active mode. The chip will fetch longitude from GPS, send it to GSM, and GSM will message it to assign Mobile phone through 2G network

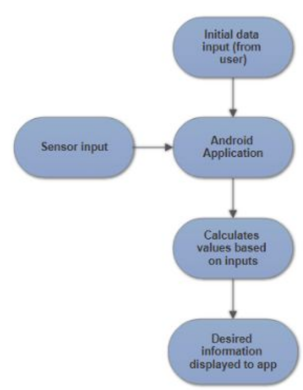


Figure 4: User Manual Flowchart

Figure 3 gives a simple breakdown of the steps involved with using the software. The block diagram is basic and easily understood because the use of the software was designed not to be complicated. The application was designed to be as easy to use as possible, once the initial information of the pet is input, the data is transmitted via Bluetooth to the application easily and will allow for the user to monitor their pet with ease.

VII. Application Design

There are many options available to providing the desired data to the end user but one method we decided to go with in particular was to integrate a phone application with the device. This provides usability, portability, optimal functionality, and a desirable low cost solution. The application is designed in such a way that we will be accepting data from the user and the device. The user's information that is to be entered will be specific to each individual. For example, weight, age, breed, sex, color, and any additional information that would be pertinent to the user or to a veterinarian. The second dataset will be vital readings from the device which will be heart rate, temperature, step counts, and GPS location. All this information will be surfaced in a way that is readable, meaning the data received will be clearly labeled and defined. We will also provide record tracking which entails a function to store all previous data for a certain period of time. This is important to see any kind of trend that may be useful to note health decline or incline.

All the data received from the device will be transmitted via bluetooth. In the application there is functionality establish connection with the device. After the device and the application has been paired, we will start transmitting data immediately. This data will be absorbed by the application and will only be displayed to the user when the user initializes the vital reading through a button click.

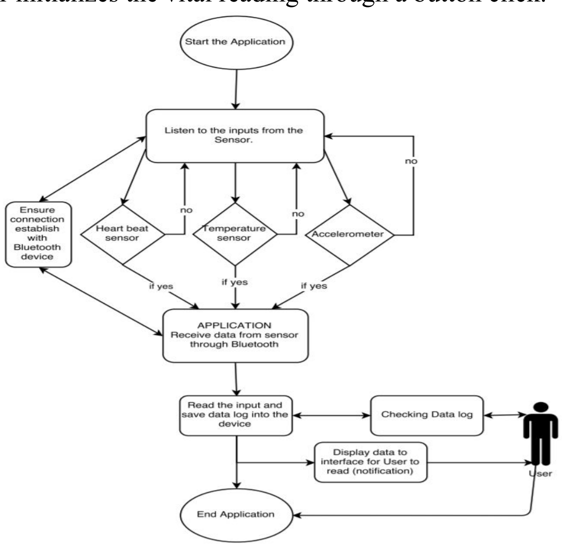


Figure 5: Software Flow Chart

The device has been programmed to transmit data every second so when this data is passed into the application and the reading is initialized, the data will be put into an equation to determine the most accurate reading for each vital sign. The heart rate reading will take 15-30 seconds for it to process the reading before it is displayed to the user. Over the 15-30 second time frame, we will take all the readings and average them and the variability is accounting for miss reading or time for contact adjustment to occur.

The temperature readings will be approached with the same method. We will take the readings from the device over a 15 second time frame at a reading per second and average it over 15 seconds. The temperature reading requires less time because we do not need to account for variability since the temperature sensor will be snug against the dog's body and the error in reading is lower than that of a heart rate sensor.

The step counter will take a little bit more calculation from the application. The accelerometer will transmit data in the form of x, y, and z coordinates. We have determined that the delta of any axis greater than 10 degrees will register as a step. This means if there is movement detected within a second, it will be transmitted to the application and a step count will be incremented. If there is not, it will do nothing. The step counter will require no manual interaction for the user as the heart rate and temperature readings will.

An important feature of the application to note is that we will be providing notifications to the user's phone to enhance interaction and usefulness. Since the data is being transmitted per second, we can store the data up to the past 30 minutes and can see trends of decline or incline of vital signs. For example, if over the past 30 minutes a dog's heart rate has steadily declined, we will send an alert to the user's phone to notify them that there may be something wrong with their dog because of the decline in heart rate and they should contact a professional. This can also apply to a dog's step counts for the day. We will store the amount of steps the dog has taken for that day and compare it to the recommended steps a dog should be taking in a day. A notification to the user will be sent if the steps are below half the recommended amount by mid day and remind them they they are far from hitting their target amount of steps.

The most basic functionality that we all agreed that the harness should support is Global Positioning System (GPS). Pets in general get lost often and dogs are no exception. Due to the technical spectrum of the harness, we felt that it would be lacking to not include GPS functionality. There will be an option under "Settings" that will allow the user to locate the device if ever need be. The device utilizes a sim card to communicate its location wirelessly and send a text message of its location. Through the application we will accept the message and use Google Maps API to translate the message and pin -point the location on their map.

The application will be supported on Android platform. The decision to release an Android application was due to the cost of deploying the application to the App store, the amount of resources and documentation provided for the framework, the available API for some of the components on our device, and time spent developing the application which would be significantly less since we are proficient in Java. Apple applications require using their proprietary language Swift.

VIII. Conclusion

The main motive of this project was to develop a device/application that could be beneficial to people and improve their life. We juggled a lot of ideas back and forth and concluded that this harness would be able to impact a lot of people and disrupt the market. Hopefully one day we can see it’s success way beyond our vision and change many people’s lives as well as their pets.

IX. The Engineers

Hai Nguyen - Hai Nguyen is a student at UCF studying for Bachelor’s of Science in Computer Engineering. Also, he has intent to go for Electrical Engineering once he complete his degree in Computer Engineering. He was born at Ho Chi Minh City in Vietnam, and he moved to U.S in 2009. He has been interest in computer at 14 years old. He also wants to obtain the knowledge from school to build his own device. After graduation, Hai want to join U.S army and become an engineer for them. Later, He can pursue his Master Degree in Computer Engineering or Electrical Engineering.

Matthew Horton - Matthew is a student at UCF studying for his Bachelor’s of Science in Electrical Engineering. He was born in the small city of Springfield, Ohio and moved to Florida at the age of 15. His interest in electronics began when he built his first computer around the age of 17 and has always been fascinated with understanding how things work. Matthew is currently employed as an engineering intern producing CAD drawings for engineers at R.L. Plowfield & Associates. Upon graduation Matthew will join EXP global, an international engineering firm, as an electrical engineer.

Dominic Vu - Dominic is a Vietnamese American born in Layton, Utah. Currently he is employed locally at a software company that specializes in contract management as a Junior Developer. The main goal for him is to gain as much experience and knowledge as possible through work and school to eventually land a stable and fulfilling job.

Acknowledgement

The authors wish to acknowledge the assistance and support of the University of Central Florida Electrical Engineering and Computer Science department with special acknowledgements to Dr. Samuel Richie and Dr. Lei Wei.

References

[1] A. (2016, September).*ATmega328/P*[PDF]. San Jose: Atmel Corporation.

[2] Adafruit MMA8451 Accelerometer Breakout. (2017). 1st ed. [ebook] Adafruit. Available at: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-mma8451-accelerometer-breakout.pdf [Accessed 1 Aug. 2017].

[3] Adafruit FONA 808 Cellular + GPS Breakout. (2017). 1st ed. [ebook] © Adafruit Industries. Available at: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-fona-808-cellular-plus-gps-breakout.pdf.

[4] L78 Positive voltage regulator ICs. (2017). 34th ed. [ebook] ST, p.14. Available at: http://www.mouser.com/ds/2/389/l78-974043.pdf.

[5] L78 Positive voltage regulator ICs. (2017). 34th ed. [ebook] ST, p.14. Available at: http://www.mouser.com/ds/2/389/l78-974043.pdf.

[6] MP1584 3A, 1.5MHz, 28V Step-Down Converter. (2017). 1st ed. [ebook] ©MPS. Available at: https://www.monolithicpower.com/Portals/0/Documents/Products/Documents/datasheet/MP1584_r1.0.pdf

[7] *DS18B20*[PDF]. (2015). Maxim Integrated Products, Inc.

[8] *Adafruit Fona 808 Cellular + GPS Shield for Arduino* [PDF]. (Updated 2017, January 1) Lady Fona. Available at: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-fona-808-cellular-plus-gps-shield-for-arduino.pdf