ECE 445

Fall 2017

Design Review

Recovery-Monitoring Knee Brace

Team #40

Locker D10

Members: Dennis Ryu [dryu3], Dong Hyun Lee [dlee134], Jong Yoon Lee [jlee642]

TA: Dongwei Shi [dshi9]

Table of Contents

1. Introduction

1.1 Objective

Thanks to modern technology, it is easy to encounter a wide variety of wearable fitness devices such as Fitbit and Apple Watch in the market. Such devices are designed for average consumers who wish to track their lifestyle by counting steps or measuring heartbeats. But it is rare to find a product for the actual patients who require both the real-time monitoring of a wearable device and the hard protection of a brace.

Personally, one of our teammates ruptured his front knee ACL and received reconstruction surgery a few years ago. After ACL surgery, it is common to wear a knee brace for about two to three months for protection from outside impacts, fast recovery, and restriction of movement. For a patient who is situated in rehabilitation after surgery, knee protection is an imperative recovery stage, but is often overlooked. One cannot deny that a brace is also cumbersome to put on in the first place. There should be a viable alternative for such people in need.

1.2 Background

With that said, our group aims to make a wearable device for people who require a knee brace by adding a health monitoring system onto an existing knee brace. The fundamental purpose is to protect the knee, but by adding a monitoring system we want to provide data and a platform for both doctor and patients so they can easily check the current status/progress of the injury.

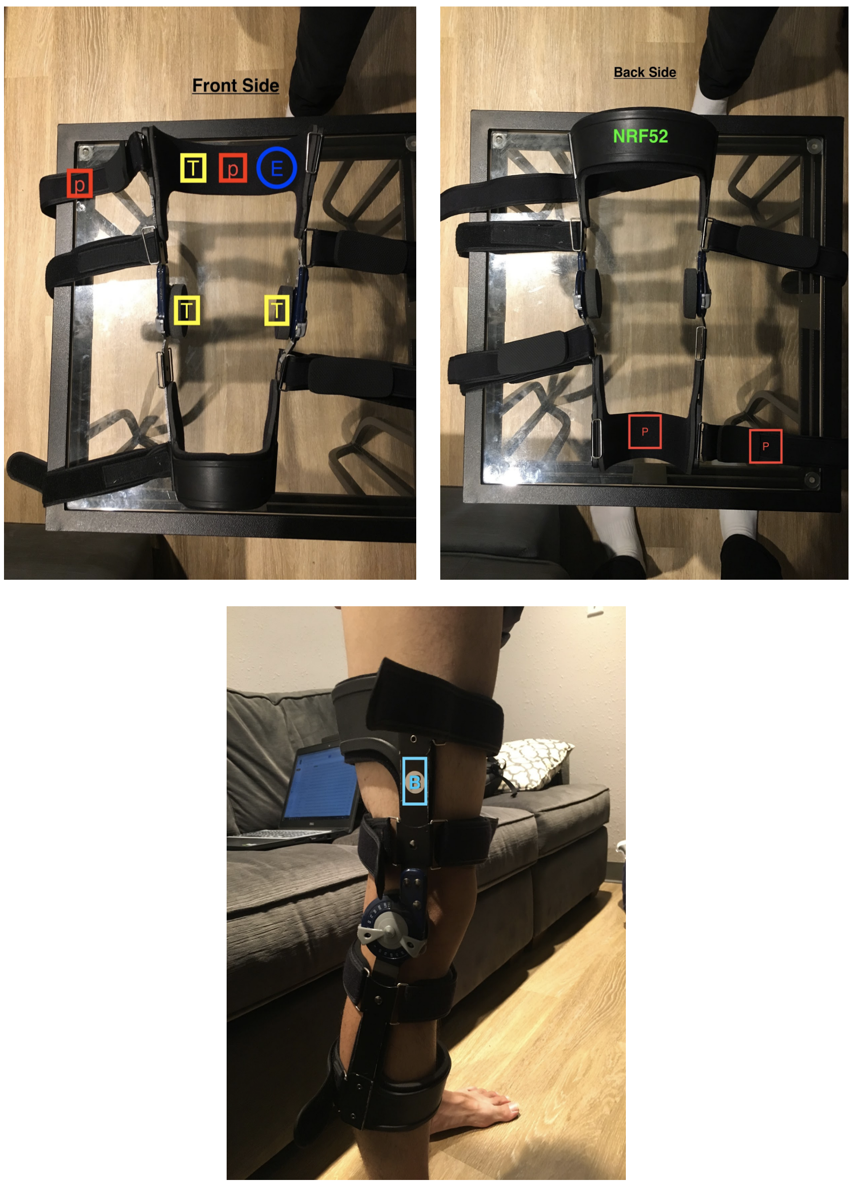
A feedback system will notify the wearer when the brace needs to be more tightly bound, and remind via. mobile interface when not worn for extended periods of time. Because the brace should also provide useful information for a professional looking over the patient, heat and EMG sensors placed on the inside of the brace will stream data to a mobile device via. Bluetooth. Marketable aspects include a minimal change in the overall weight of the brace, and the small but sufficiently ample lifespan of the battery we will be using as our power supply.

1.3 High-Level Requirements List

- Brace must be able to stream data continuously for a 48-hour period.
- EMG sensors signal must be able to determine too much muscle movement & temperature sensors must capture readings accurately
- Pressure sensors must determine whether the straps are securely fit

2. Design

Recovery-Monitoring Knee Brace uses a knee brace that is commonly used for knee-injured patients. The basic frame is a lightweight metal with multiple straps that go around the patient's knee. There are three straps on each side of the brace with an angle control dial at the middle. Figure 2 shows specific sensor locations on the front side. T is for temperature, located on each side of the knee to measure the current temperature. In addition to those sensors, another is located on the upper side of the knee brace to compare the body temperature and knee temperature. The pressure sensors are located underneath the straps to measure optimal pressure for proper wear. The power module is going to be on the side of the knee brace as located on the Figure 4. Our PCB board will be back located on back of the knee brace.



Figures 1, 2, 3: Front/Back/Right Sides of Knee Brace and Sensor Locations

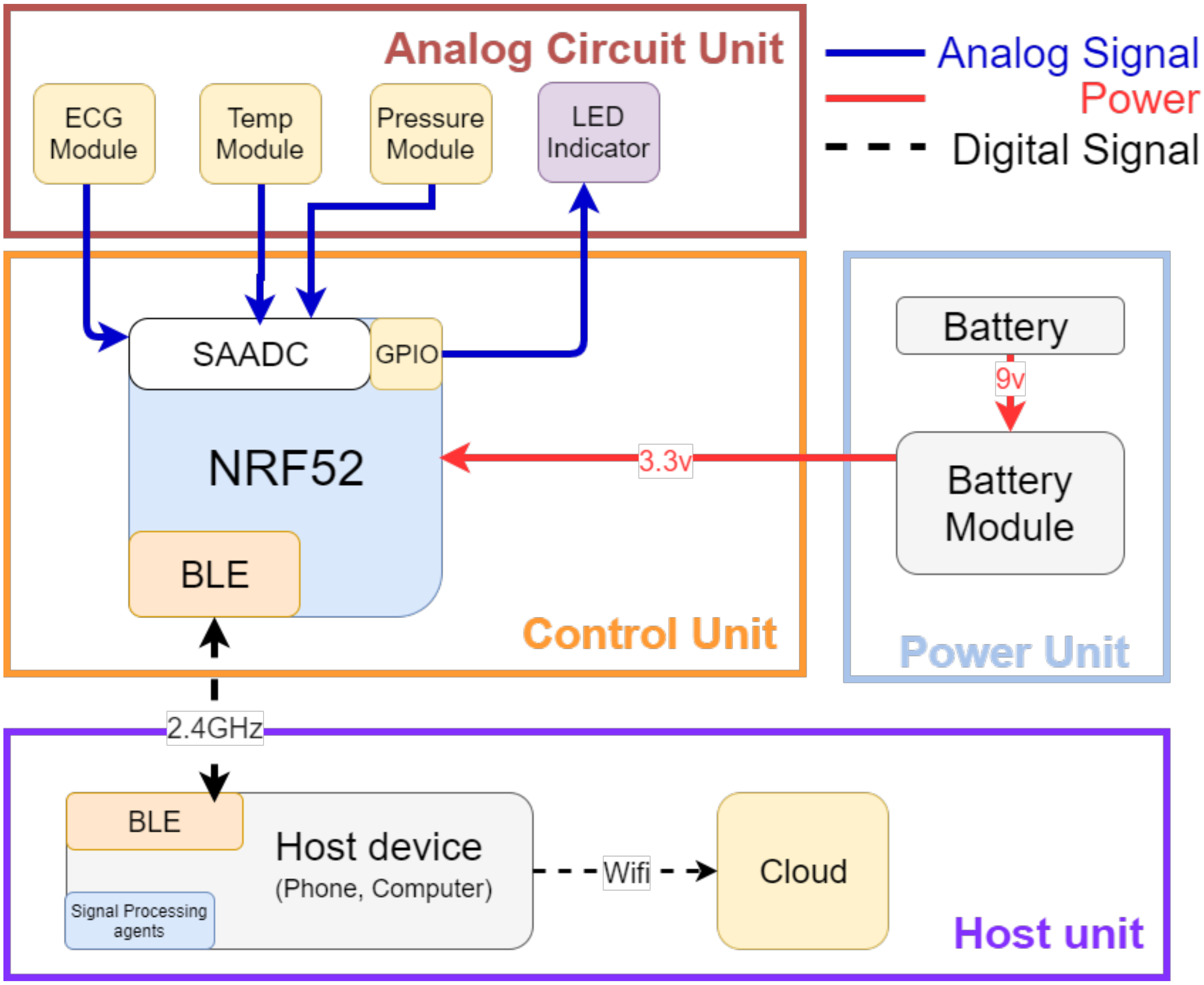


Figure 4: Diagram of Overall Design

2.1 Control Unit

This will be the main module of our project device. It must be a device that can collect all of our bio-signals (pressure, temperature, ECG) from the analog circuit, meaning it should have an analog-to-digital converter. Because it is difficult to execute all of the required signal processing on the chip, we need a System-on-Chip (SoC) that has a Bluetooth module incorporated to communicate with the host device. Our control unit also must consume low power.

2.1.1 nRF52832 SoC

The nRF52 SoC embeds a powerful yet low-power ARM® Cortex®-M4 processor with a 2.4 GHz RF transceiver. With the 2.4 GHz RF transceiver, it has the capability to use standard low-energy wireless protocols such as ANT and BLE. This makes the chip particularly appealing to low-power applications such as ours. The SoC is also packed with such peripherals that will enable us to achieve our goals and though satisfying our requirement as a SoC.

Requirements	Validation
1. Maximum of 5 mA of current consumption. 2. The chip needs to be reversed engineered and designed on the PCB	1. Utilize a power analyzer to measure average current consumption. 2. Power the NRF52 on the PCB with the binky code example. Validate that the led blinks and can connect to a host device

SAADC: The SoC comes with a differential successive approximation register (SAR) analog-to-digital converter. It has eight different channels which means that we can collect up to 8 different signals simultaneously. The sampling rate can be high as 100K samples/second[2], which is much higher than our target rate of 1000 samples/second for ECG and 10 samples/second for pressure and temperature.

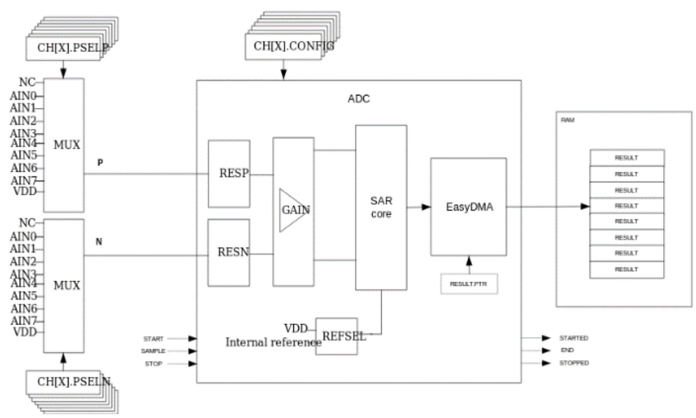


Figure 5: Overall Structure of SAADIC

Requirements	Validation
1. SAADC will need to support 8 ADC to obtain all our signal. 2. Support 1K sampling rate of 14-bit resolution	1.Unit test getting different signal level for all 8 different pins and checking if the values that is obtain doesn't not affect each other 2. Unit test of sampling at 1K of 14 bit resolution and checking the result through. ±5 of accuracy in timing will be checked with an

	JLINK RTT(Real time debugger).

GPIO: The general purpose input/output will be simply used as an indication that will display the status of the device, such as the battery level or connectivity, to a host device. This will be done by connecting LEDs to the GPIO pins and signaling them when necessary. Since we have more than 32 pins on the board[2], we have plenty of pins to toggle the LEDs.

Requirements	Validation
1. Since GPIO will be used for Status LED indicator, it has to provide digital signal to control LED indicator.	1. Section 2.2.4 (Status LED Indicator) will be used to validated the working property of GPIO. LED, itself, could be checked with power source but working property of status LED (turning it off and on for power status) is GPIO controlled. 2. GPIO also can be checked separately by using a oscilloscope to the digital input and output pin on GPIO. Starts the code in the GPIO and check Vcc on the pin.

Timer: The SoC also has a Timer module the runs on the high-frequency clock source which is typically at 16MHz. This extreme high-precision clock will enable us to have a precise interval for the ADC operation and will also schedule periodic jobs for us. This timer will be suitable as it has a resolution of 62.5 ns and a tolerance of less than +- 40 ppm[2].

Requirements	Validation
1. Deliver stable clock signals for the necessary peripherals with ±1%.	1. Unit test to count the ticks for a period and compare it with the expected count. 2. Have an LED that will blink every second and compare it with an actual timer.

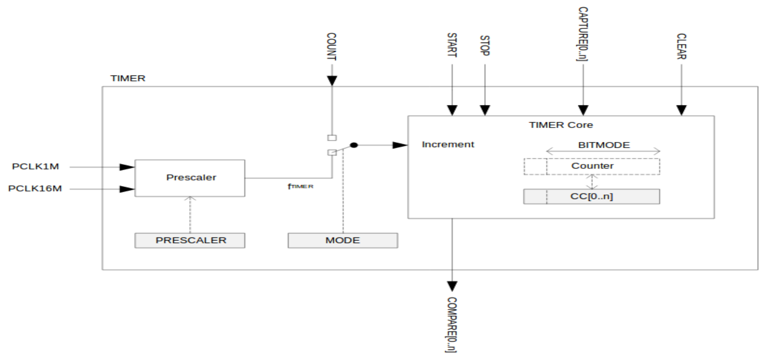


Figure 6: Overall Clock System Structure

Bluetooth: The SoC includes a 2.4 GHz RF transceiver. In order for us to use it, the vendor provides a binary software implementing ble (bluetooth low energy) protocols. This software stack is called 'softdevice'[2] and is required in every project involving the use of ble. Our project will work in unison with the soft device stack to stream the necessary data to the host device. We will package our 14-bit SAADC result into a 16 bit buffer Since this protocol can support up to 16 KBytes/Sec[3], it easily satisfies our data rate requirement as follow.

Calculation

2 EMG Data : $\frac{1000 \text{ sample}}{\text{second}} * \frac{2Bytes}{sample} = 2KBytes/Sec$

3 Temperature data : $3 * \frac{.5 \text{ sample}}{\text{second}} * \frac{2Bytes}{sample} = 3Bytes/Sec$

6 Pressure data : $6 * \frac{10 \text{ sample}}{\text{second}} * \frac{2Bytes}{sample} = 120Bytes/Sec$

Total: $2123\ Bytes/Sec$

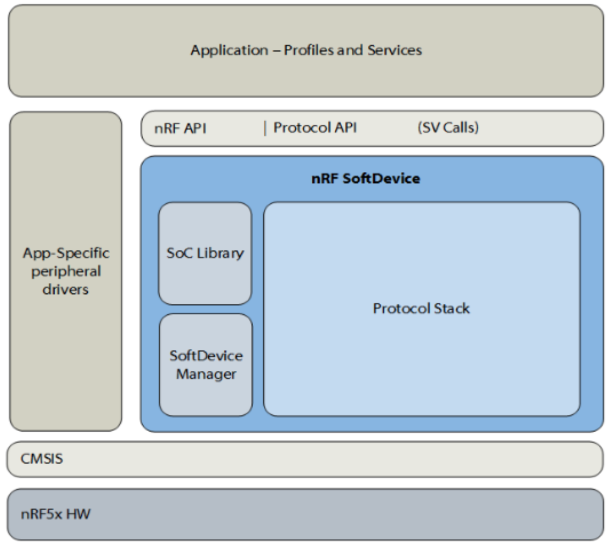


Figure 7: Software Structure Diagram

Requirements	Validation
Support minimum of 2162 Bytes/second of data transfer to the host device Less than 1% of package loss when the data is being streamed to the host device. At least 5 meters in range with stable connectivity. -70dBm or higher value will guarantee stable connection[1].	1. Data transfer rate test and package loss test will be done by creating a unit test for it with quantitative result indicating the current limits 2. Range stability will be tested with measured Broadcasting Power value we get from various distance.

2.2 Analog Circuit Unit

The ultimate goal of the analog circuit unit is to integrate the multiple sensors that will capture our bio-signals and send them into our microcontroller. The sensors are located throughout different parts of the knee brace and will measure the necessary values such as temperature, pressure, and usage of muscle. To capture a better bio signal from each of the sensors after implementation, we used instrumentation amplifiers such as INA333 and OPA2333, which consists with two INA333 components. It has a low offset voltage, low drift, and low noise. It is important to capture a precise value for our needs.

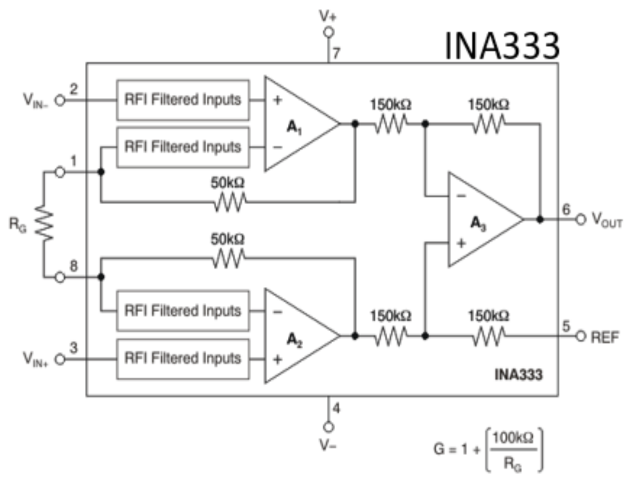


Figure 8: Schematic of INA333

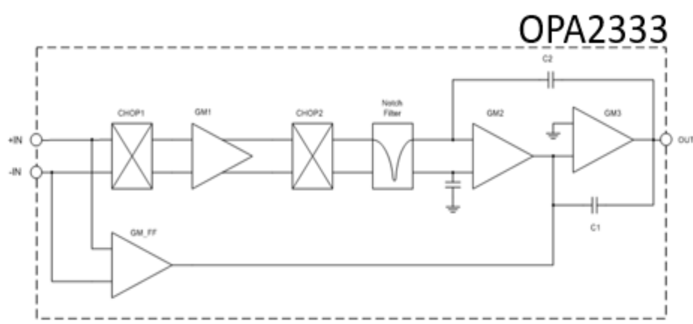


Figure 9: Schematic of OPA2333

2.2.1 Temperature Sensor and its Circuit

The overall function of the temperature sensor is to measure the temperature of the knee and compare it with that of another part of the body, and see whether the knee is currently swollen up or not. The temperature of the general knee will be around 30~40°Celsius. Since it is really important to capture a precise temperature difference of ±2° Celsius, Metal Resistance Temperature Detectors are an optimal choice. RTD is able to measure temperature range between -200 to 850°Celsius and has a sensitivity of 3850 ppm per Celsius. RTD (M-Series 222) from Heraeus Sensor Technology USA is the one that we are going to use under the circuit called a Wheatstone Bridge Circuit, which is proper for our thermistor. As we said in the introduction of the analog circuit unit, we will complement this with an INP333 if we require amplification.

Requirements	Validation
Temperature Sensor has to capture precise temperature of user's body in real time with precision of ±0.5° Celsius with actual real world temperature.	1. Measure the temperature of User's Knee with temperature sensor. 2. Use external thermometer and measure the Knee's temperature. 3. Compare the knee temperature from temperature sensor and actual thermometer to ensure that temperature sensor is working properly.

Schematic

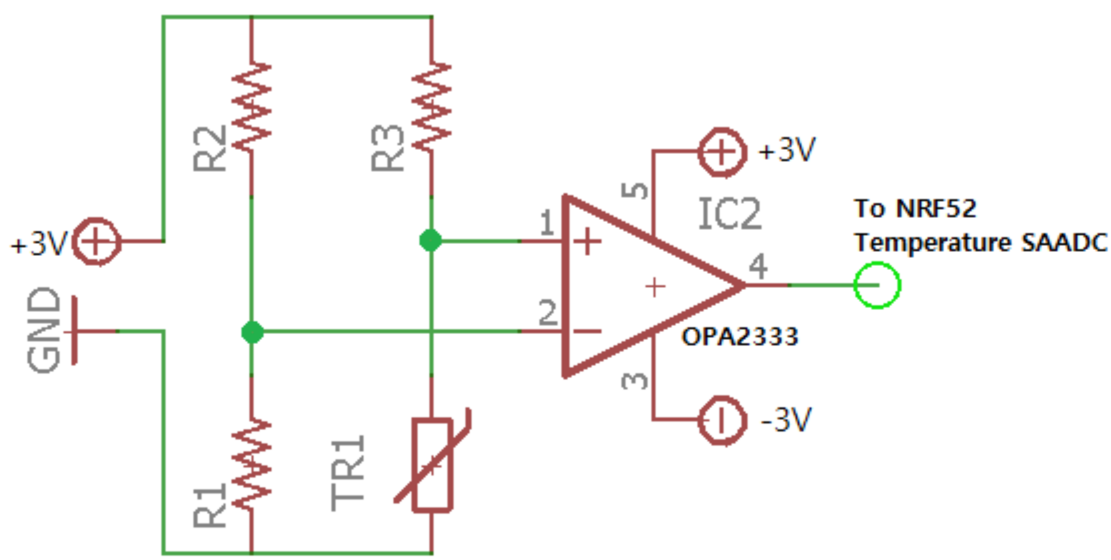


Figure 10: Schematic of Temperature Sensor (Wheatstone Bridge Circuit)

Calculation

$\mathrm{TCR} = 3850\ \mathrm{ppm/K}$ $\mathrm{TCR} = \frac{R_2 - R_1}{R_1(T_2 - T_1)} 10^{-6}$	$R_1 = room\ temperature\ (\Omega)$ $R_2 = operating\ temperature\ (\Omega)$ $T_1 = room\ temperature\ (\mathrm{K})$ $T_2 = operating\ temperature\ (\mathrm{K})$

$R_T = R_O[1 + AT + BT^2]$	$R_T = resistance\ at\ measured\ temperature$ $R_O = resistance\ in\ \Omega\ at\ t = 0°\mathrm{C}$ $A = 3.9083 \times 10^{-3}\ /°\mathrm{C}$ $\mathrm{B} = 5.7750 \times 10^{-7}\ /°\mathrm{C}^2$

2.2.2 Pressure Sensor and its Circuit

The Recovery-Monitoring Knee Brace will have pressure sensors under the strap of the knee brace (Appendix A has a pressure sensor placement). Pressure sensors will monitor whether the user has properly worn the knee brace or not. The degree of tightness of the strap will be measured through pressure sensors, and they will send an analog signal (voltage) into the nRF52. Sensors much be flexible and sensitive enough to measure the miniscule pressure difference on the strap. The adequate operating force measurement range should be between 0.04lbs to 4.5lbs. The pressure sensor will be an FSR 406 from Interlink Electronics

Requirements	Validation
Pressure Sensor has to distinguish between perfect fit and loose fit of straps within User's knee.	1. Tightly put straps of Knee brace into the User and check the host device for a status of the straps. 2. Loosely put straps of Knee brace into the User and warning has to show up into our host device because pressure sensor indicates low pressure.

Schematic

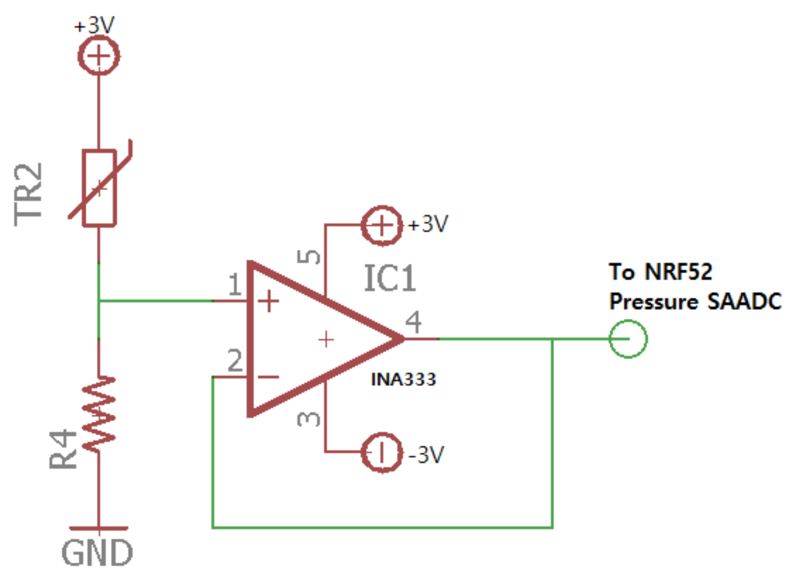


Figure 11: Schematic of Pressure Sensor

Calculation

$$V_{out} = \frac{R_m}{R_m + R_{FSR}} V_{in}$$

Plot

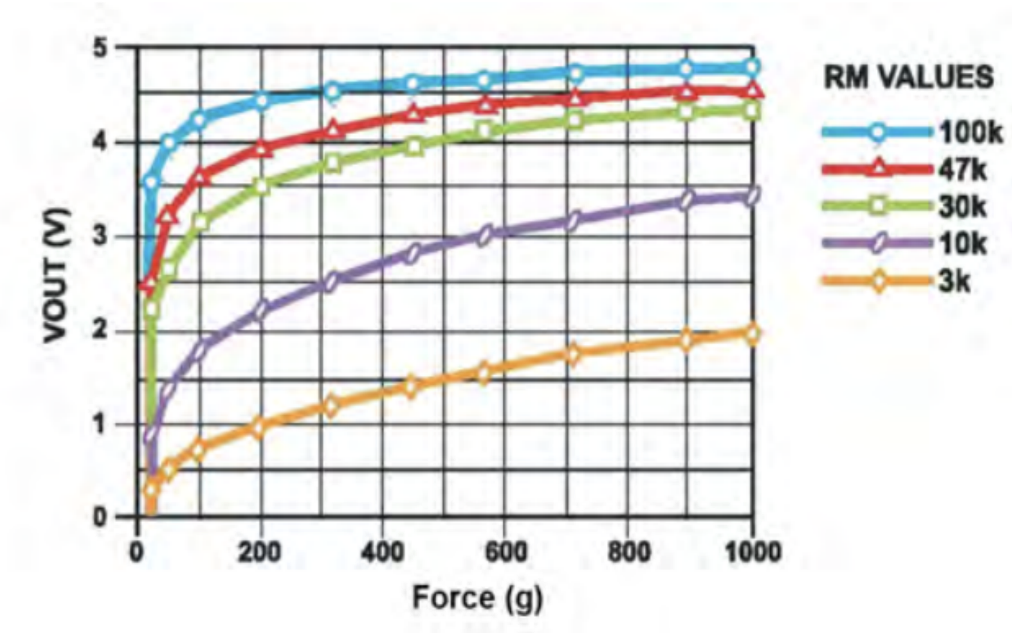


Figure 12: Plot Vout from Pressure Sensor Circuit with different Resistor Value

2.2.3 EMG Sensor

The purpose of the EMG sensor is to measure the usage of muscle. We are going to implement a two-electrode EMG sensor to indicate the usage of the leg muscle to prevent the user from excessive usage. Figure 1 shows the EMG sensor placement. Two electrodes will come out of the sensor and will be attached to the skin. We will use INA333 for preamplification and different types of filter such as 1st order High Pass filter to capture a better signal.

Requirements	Validation
1. EMG Sensor has to capture the signal increases with an increase in muscle activity/effort.	1. Using a host device within the EMG sensor which connected into one of our team member and compare the signal difference between flexed muscle and relaxed muscle.

Schematic

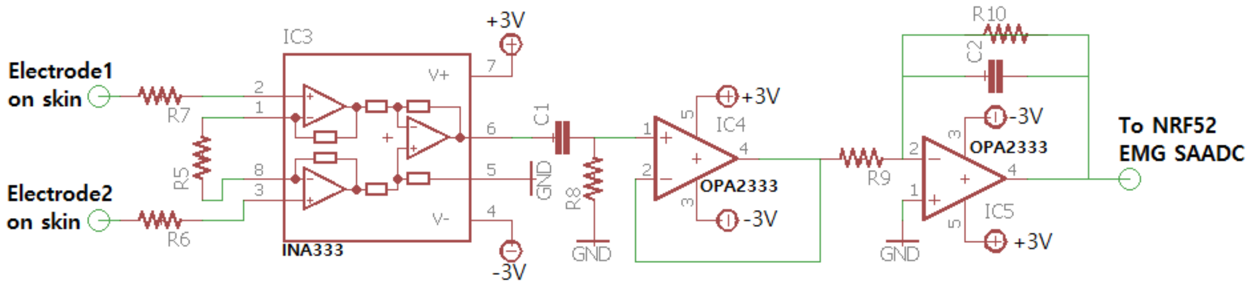


Figure 13: Schematic of EMG Sensor

2.2.4 Status LED Indicator

The Status LED indicator will be a simple LED circuitry with the functionality of displaying whether device is turned on or off. It will be connected into the GPIO from the microcontroller and power will be supplied from the power module. The LED indicator will work as we desire. The LED indicator will have a low duty cycle (<%5) and consume less than 1mA of current draw.

Requirements	Verification
1. Status LED1 has to show power on/power off status of the device. 2. Status LED2 has to show that device is properly connected into host device or not.	1. Take out the battery and check the LED and put battery back into power module to check that LED light signal kicks in. 2. Connect the microcontroller with host device and see the LED2 lights on or not.

2.3 Power Supply

The power supply has to provide a steady 3.3 voltage input for the microcontroller and multiple bio sensors when the device is on. Our main power source will be a 9 Volt battery, but we are going to use a Buck converter to step down the 9V into 3.3V.

2.3.1 9 Volt Battery

Requirements	Verification
1. The 9V battery has to provide fixed 9V in the tolerance of ±3%.	1. Use multimeter to ensure that the 9 Volt Battery providing proper voltage with tolerance of ±3%. 2. Check the voltage value on the multimeter and make sure that battery is providing fixed voltage value in the tolerance range.

2.3.2 Buck Converter

Requirements	Validation
1. Buck Converter has to provide steady 3.3v with ±3% from the battery. 2. Buck Converter has to provide about 600mA into the device.	1. First connect the power supply into the Buck Converter as a power source and change the input voltage from 5V to 10V within the range of DC/DC converter allows. 2. Measure the value from voltage output of the Buck converter and verify that the voltage is providing fixed 3.3Volt.

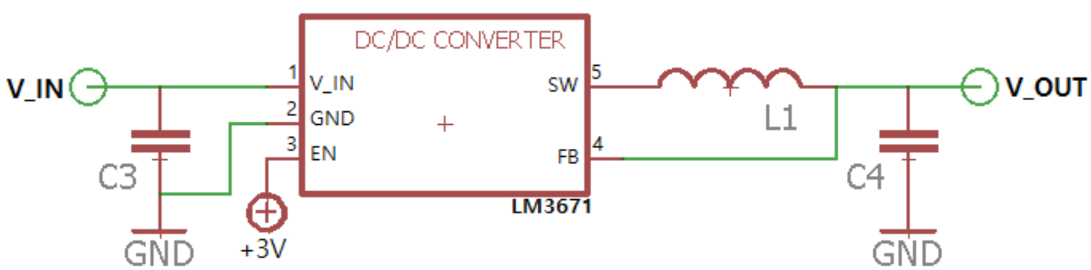


Figure 14: Schematic of DC-DC Converter

3. Software

We will design a two-node system that consist of a Host and a device. For the device node, the goal is to put the device in lower power mode(sleep) as much as possible to allow long operation. To achieve this goal, we are going to utilize the interrupt based SAADC available. Also, the device will use the Bluetooth Low Energy soft device provided by the vendor to send burst of BLE packages to the host device. The host device has multiple tasks it needs to take on when data from the device comes in. We will have multiple agent with different tasks to run concurrently to achieve the necessary real-time tasks.

3.1 Device Flowchart

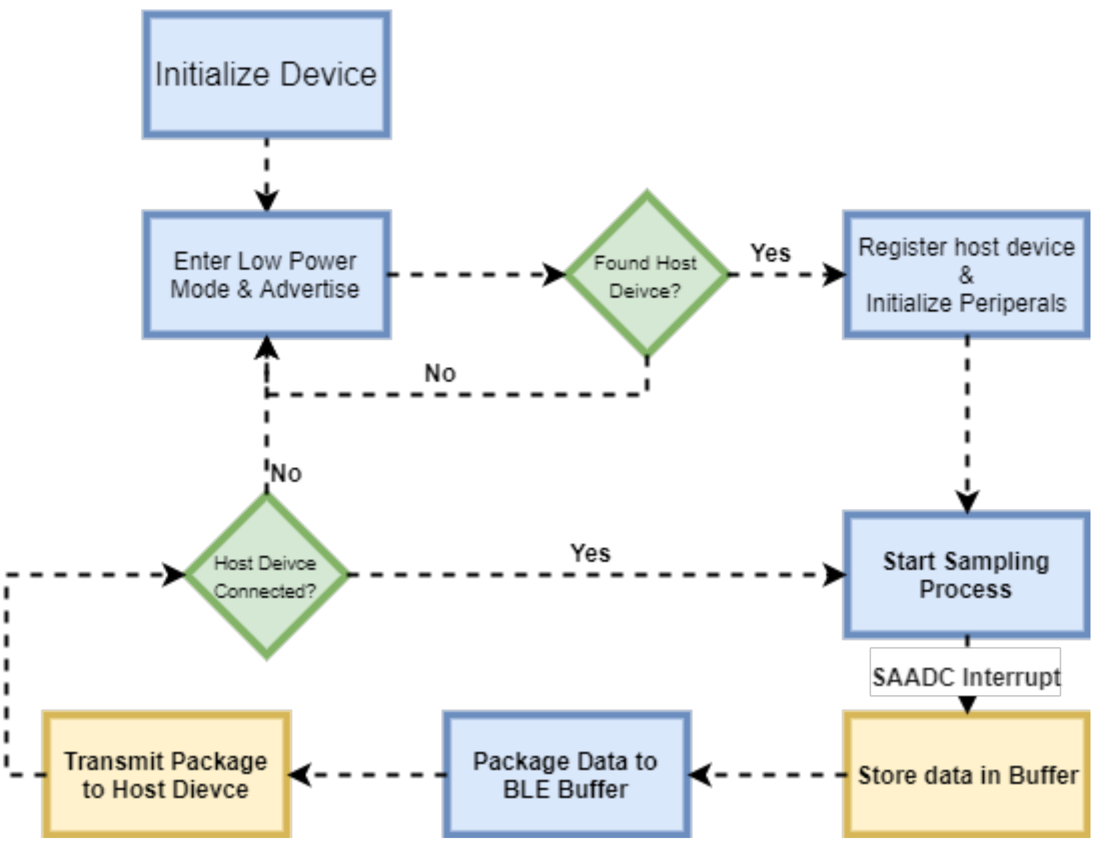


Figure 15: Flowchart of Device

3.2 Host Flowchart

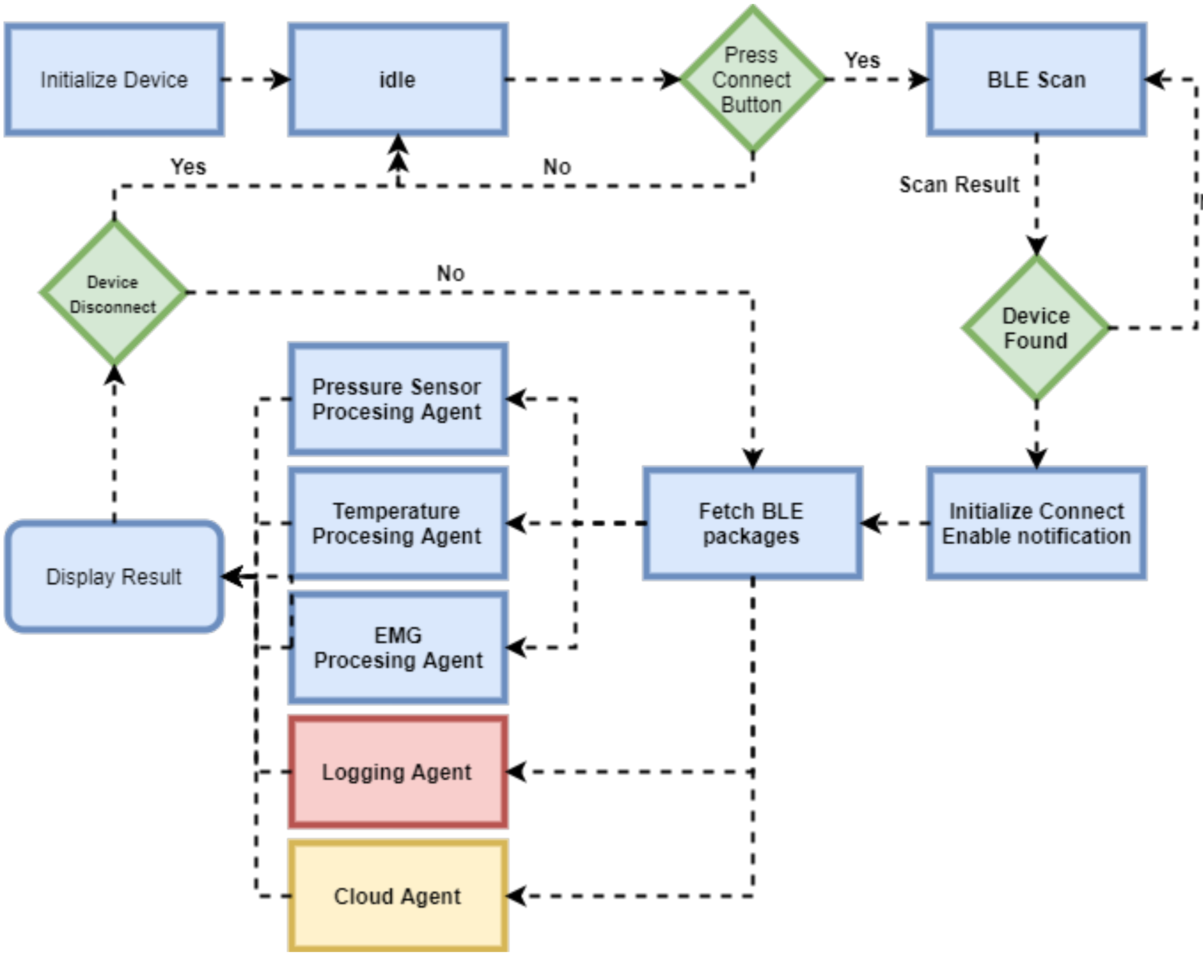


Figure 16: Flowchart of Host

4. Tolerance Analysis

Knee monitoring systems require precise bio-sensing. The device consists of many different sensors and it is important to capture the clear values for the sake of the user. One of the most important sensor components in the device is the temperature sensor which captures the body temperature and knee temperature in real-time. However, there is one good thing about our device: calibration through microcontroller. The temperature sensor delivers analog signals into the microcontroller and we have to perform post-calculations to change the voltage into a proper temperature value. Section 2.2.1 is enclosed with calculations and equations that are required for a voltage-to-temperature change.

700 Platinum RTD has its own tolerance measured already in the datasheet. R_0 which is Nominal Resistance at 0° Celsius can have ±0.12% differences [3]. Also, we have to consider the heat dissipation in the temperature sensor, which can cause an altered value in voltage. The usage of the Wheatstone Bridge circuit is to minimize the heat dissipation and increase

sensitivity in the sensor. By adding another comparator voltage divider in parallel, the temperature sensor will be using a differential voltage measurement around zero that helps for a better improvement in sensitivity of sensor, while reducing current flow through the component. The following equation is for the bridge circuit when it is balanced (starting point).

$$V_{out} = V_{in}\left(\frac{R_T}{R_T + R_3} - \frac{R_2}{R_2 + R_1}\right)$$

$$\text{When } V_{out} = 0, \quad \frac{R_3}{R_T} = \frac{R_1}{R_2}$$

Since our device is going to measure the temperature around 20 degree Celsius, we calculate the R_{20} value using the equation from the section 2.2.1.

$$R_T = R_O[1 + AT + BT^2]$$

$$R_{20} = 100\Omega[1 + 3.9083\times10^{-3}(20°\text{C}) + 5.7750\times10^{-7}(20°\text{C})^2] = 100\Omega(1.0783)$$

However, R_O has tolerance of ±0.12%, and R_{20} can have a tolerance of $R_{20}0$, too. $R_{20} = 94.8904\Omega \sim 120.7696\Omega$. Using correct R_{20}value, we have to choose the $R_1, R_2, and\ R_3$ values.

5. Cost

5.1 Parts

PARTS				
Part Name	**Distributor**	**Unit Cost**	**Quantity**	**Total**
Temperature Sensor 32208548	Heraeus Sensor Technology	$_4.53	3	$13.59
Pressure Sensor 30-73258	Interlink Electronics	$_7.95	4	$31.8
INA333AIDGKR chip	Texas Instruments	$_4.60	1	$_4.60
OPA2333AIDR chip	Texas Instruments	$_4.34	9	$38.79
LM3671 chip	Texas Instruments	$_1.44	1	$_1.44
Skin-contact Electrodes	Covidien Kendall	$11.86	2	$23.72
Various Resistors, Capacitors, Inductors	DigiKey	$_0.75	19, 4, 1	$18.00
			Analog Circuit Unit	$131.94
9V Alkaline Battery	Newegg	$_7.47	1	$_7.47
			Power Unit	$_7.47
nRF52 Microcontroller	Nordic Semiconductor	$_2.11	2	$_4.22
			Control Unit	$_4.22
			PARTS TOTAL	$143.63

5.2 Labor

Name	**Hourly Rate**	**Total Time Investested**	**Total Hourly Payment x 2.5 = Total**
Dong Hyun Lee	$30	200	$15000
John Lee	$30	200	$15000
Dennis Ryu	$30	200	$15000
		Grand Total	$45000

6. Schedule

Week	Hardware (Dong Hyun)	Hardware (Dennis)	Software (John)
10/02	**Mock Design Review**	**Mock Design Review**	**Mock Design Review**
10/09	**Design Review**	**Design Review**	**Design Review**
10/16	Build temperature and pressure sensors	Build power module Complete pin assignments for microcontroller	Prototype the NRF52 chip to collect SAADC data and construct the BLE layer to communicating with host device
10/23	Build EMG sensor	Build modules for parts in EagleCAD and start schematic	Build baseline PC GUI that can graph the incoming signal data from the NRF52.
10/30	Calibrate temperature, pressure and EMG sensor	Finish EagleCAD layout for entire circuit	Program a data logger for sensor calibration and Integration with HW team
11/06	Help optimize EagleCAD PCB layout	Complete final PCB design for submission	Prototype the various agents in the host device(data processing agents, cloud agent).
11/13	Quantify data from temperature, pressure, and EMG sensors	Solder components onto PCB & assemble leg brace	Finalize on GUI design and optimize the NRf52 chip for power efficiency.
11/20	Make necessary signal processing changes	Field test the device for adjustments	Finalize the algorithm that will be used in the data processing agents.
11/27	Identify any hardware mishaps	Sign-up for presentation	Create and test all the unit test for final checkup
12/04	Finalize demonstration	Discuss rough draft for final paper	Commit final version with all the bugs fixed and code refactored
12/11	Discuss future enhancements for device	Complete final paper	Submit software scripts

7. Safety statements

Considering the scope of our project and the IEEE Code of Ethics, we wish to make a few statements regarding the standards.

[1] To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;

As we will be using a 9V alkaline battery as our power source, which has on average a capacity of 1000mAh. Electrical concerns such as discharge will be at a minimum, and the only mechanical concerns relevant will be of the existing knee brace currently out in the public. Some lab safety concerns we may have are related to soldering when creating our electrical circuit, both on a test breadboard and on the final PCB.

[5] To improve the understanding of technology; its appropriate application, and potential consequences;

Our knee brace serves to make comfortable the lives of the injured, and is founded on the notion of helping. But we must be careful not to overlook important medical standards that our device may miss out upon.

[6] To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;

During the final stages of our project, we will need to solder the components onto our printed circuit board. Only students qualified and who have passed the lab safety trainings should be eligible in performing tasks that involve the use of potentially harmful lab equipment.

[7] to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;

In order to seek improvement and success in constructing our knee brace, we must be open to criticism and embrace the public opinion.

8. References

[1] Devzone.nordicsemi.com. (2017). Maximum RSSI value of -42 dBm - Nordic Developer Zone. [online] Available at: https://devzone.nordicsemi.com/question/64899/maximum-rssi-value-of-42-dbm/ [Accessed 6 Oct. 2017].

[2] Infocenter.nordicsemi.com. (2017). Nordic Semiconductor Infocenter. [online] Available at: https://infocenter.nordicsemi.com/index.jsp [Accessed 3 Oct. 2017].

[3] interlinkelectronics. (2017). *Pressure Sensor FSR400 Series Datasheet*. [online] Available at: http://www.interlinkelectronics.com/datasheets/Datasheet_FSR.pdf [Accessed 3 Oct. 2017].

[4] Punchthrough.com. (2017). Maximizing BLE Throughput on iOS and Android - Punch Through. [online] Available at: https://punchthrough.com/blog/posts/maximizing-ble-throughput-on-ios-and-android [Accessed 6 Oct. 2017].

[5] Sensing.honeywell.com. (2017). *Installation Instructions for the 700 Series Platinum RTD Temperature Sensors*. [online] Available at: https://sensing.honeywell.com/index.php?ci_id=50017 [Accessed 6 Oct. 2017].

[6] Sensing.honeywell.com. (2017). *700 Series Platinum RTDs Data Sheet*. [online] Available at: https://sensing.honeywell.com/index.php?ci_id=50019 [Accessed 6 Oct. 2017].

[7] Texas Instruments. (2017). *LM3671/-Q1 2-MHz, 600-mA Step-Down DC-DC Converter*. [online] Available at: https://cdn-shop.adafruit.com/product-files/2745/P2745_Datasheet.pdf [Accessed 4 Oct. 2017].

[8] Texas Instrument. (2017). *INA333 Data Sheet*. [online] Available at: http://www.ti.com/lit/ds/sbos445c/sbos445c.pdf [Accessed 3 Oct. 2017].