ECE 445

Spring 2017

Design Review Document

Turning Tracker for Pressure Ulcers

Team 32 - Brad Kearbey and Robert Paul

TA: Jose Rodrigo Sanchez Vicarte

1 Introduction

1.1) **Objective**

Pressure ulcers, or bedsores, is a common skin condition affecting millions of bedridden patients in the US[9]. The condition is caused by the compression of blood capillaries supplying the skin, leading to necrosis, or death, of skin cells.

Pressure ulcers are a huge problem for many elderly patients, predominantly in low-budget nursing homes. While there are many factors that go into the accumulation of pressure ulcers, one of the main causes is the inaction of nursing assistants (CNAs) or other medical staff to flip their patients regularly[3][4][5]. Because ulcers can develop within 2-6 hours (depending on the patient) of stagnant positioning, in the US patients are necessarily to be flipped once every two hours[8]. One of the simplest causes of such high levels of bedsores lies within the organization of nursing staff. Our research shows that beyond all other factors, medical staff tend to become overwhelmed with their other duties, and simply forget to flip their patients, or flip their patients far later than necessary[3]. This is the case, however, with the more ethical of medical staff. There have been known cases of staff purposely missing flips[3]- beith laziness or ignorance. In all cases, though, medical staff are trained to react to ‘beeping’ or ‘buzzing’ which we hope to enact as the standard of patient flipping.

The following design document makes a step forward in solving the bedsore ‘pandemic’ - specifically, in lower-budget nursing homes. Our proposed Turning Tracker pad is to be placed underneath respective patients in order to monitor each patient’s two-hour flip timer. The pad consists of multiple force sensors that keep track of the patient's position on the bed. When a significant weight distribution is shifted - i.e. a flip is registered-, the Turning Tracker will reset the two-hour timer on that particular pad. When the timer is depleted, however, a signal is sent to each staff monitoring that particular pad identifying that a patient needs to be flipped. If a flip is not registered after the required 2-hours, a mechanical buzzer on the device will sound. At which point, the patient would call his or her nurse.

1.2) **Background**

According to the U.S. Department of Health & Human Services[1], pressure ulcers affect 2.5 million a year, costing $9.1-$11.6 billion a year in the U.S. There are more than 60,000 deaths and more than 17,000 lawsuits filed each year directly relating to pressure ulcers.

As of 2017, there are, not surprisingly, multiple attempted solutions to this problem.

One being the *alternating pressure air mattress* (APAM). According to a belguish study by the Department of Public Health at Ghent University, they found that “there was no significant difference in incidence of pressure ulcers (grade 2–4) between the experimental [group using APAM] (15.6%) and control group (15.3%) [group on a visco-elastic foam mattress (Tempur®), flipped every 4 hours].”[2] Because many nursing homes do not carry Tempurpedic mattresses and patients are flipped every *two* hours in the U.S.[3], this study lead us to believe APAM’s simply do not solve the problem.

Another solution includes a *self-turning bed*. According to Christina Paul, an Occupational therapist of five years, self-turing beds are extremely rare to find in most nursing homes[3]. Our research has shown that they are simply too expensive for widespread use in nursing homes.

With all of this being said, there is just reason that many nursing homes rely solely on their staff to flip patients. The most steadfast, cheap, and simple solution to the bedsore pandemic is simply reminding the nurses and CNAs to flip their patients. As stated above, medical staff react to ‘beeping’ and ‘buzzing’ on a much bigger scale than a mental note, or schedule.

1.3) **Requirements**

- The product *must* reliably **measure** if and when a patient is flipped
- The product *must* reliably **communicate** if and when a patient is not flipped within the two-hour window
- The product *must* be **safe** from all harm to users

2 Design

2.1) **Block Diagram**

Our system requires four different modules to meet all the higher level requirements that are listed above. This design contains a power module, control unit, sensor module and communication module. These modules are shown below in figure 1. The power module is responsible for charging a battery that will power the rest of the circuit with a regulated 3.3 V. Our sensors will be the only devices inside the pad and are in charge of measuring a patient's weight distribution. The microcontroller will use the sensor data, the battery charging data, and the Wifi IC data to decide when a person has been flipped and when to notify a nurse to flip a patient. Finally the communication module will send and receive data to the microcontroller to ensure reliable communication to the nurses regarding turning their patients.

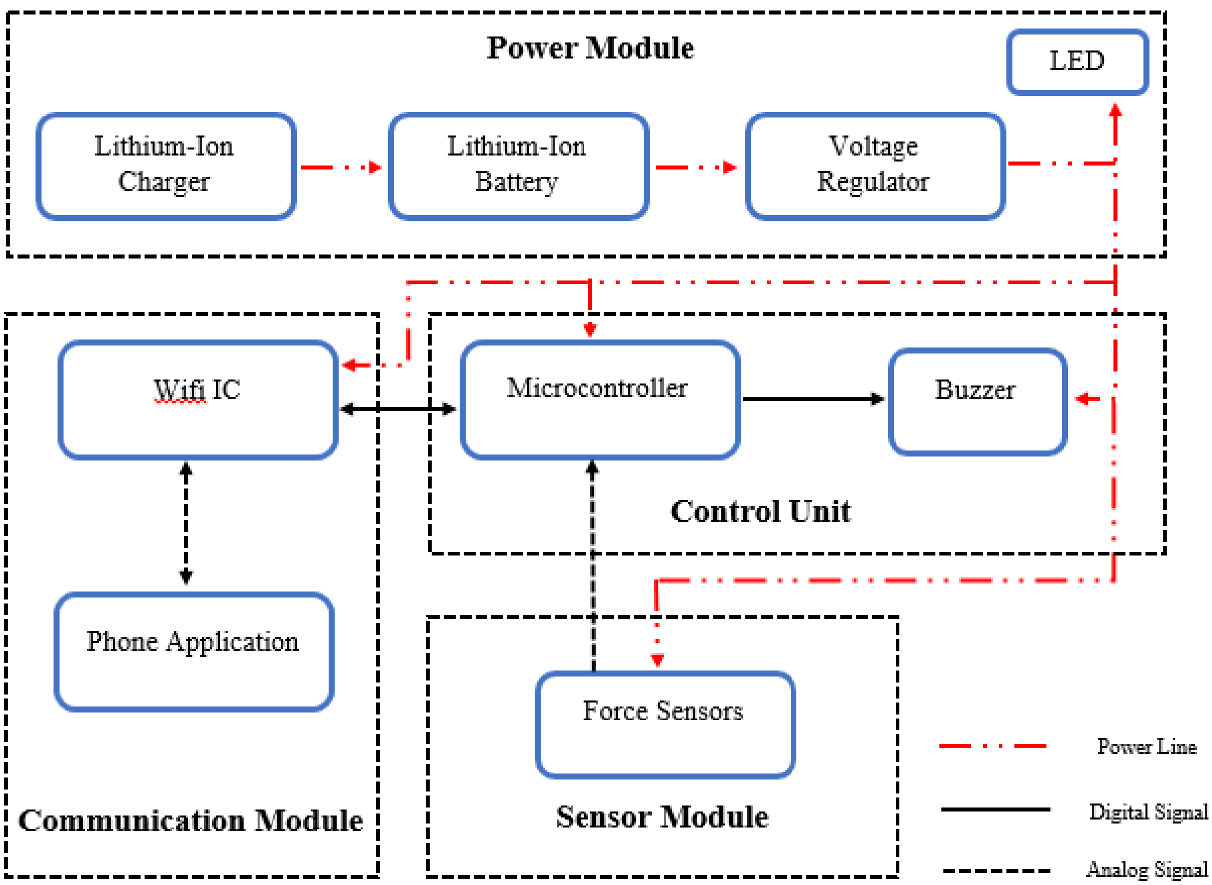


Figure 1. Block Diagram

2.2) **Physical Design**

2.2.1) **General Design**

The rough sketch depicted in Fig. 2 depicts what the final product will look like from the outside. It is meant to demonstrate the vision of the final result. From the outside, the only components visible are the pad itself, the power cable, the power converter, the Velcro straps, and our module pouch (we'll call the Hub), which will hang off the bed and house all "uncomfortable" electrical components. The adjustable straps, to be used with the railings of the bed, will ensure the pad stays in the appropriate area of the bed.

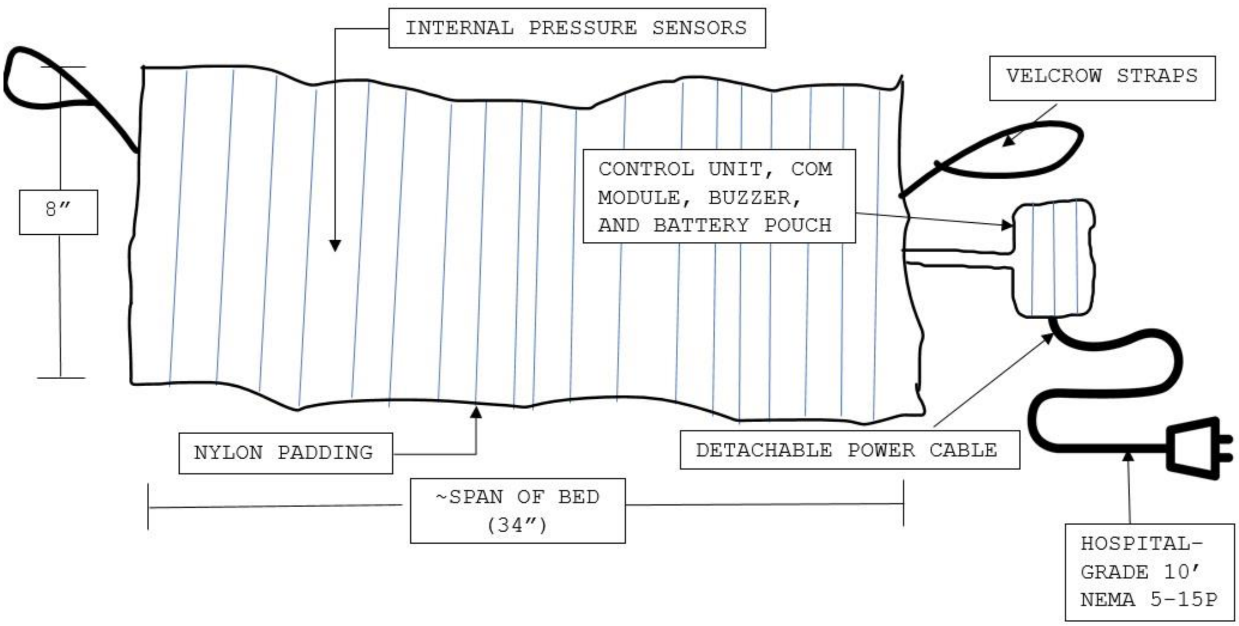


Figure 2. Practical Sketch of Final Product

2.2.2) **Sensor Layout**

Fig. 3, drawn to scale, shows the mechanical layout of the ideally stretched out pad. This cutaway drawing shows the placement of the force sensors relative to themselves and the pad. Many components are purposely left out of this layout.

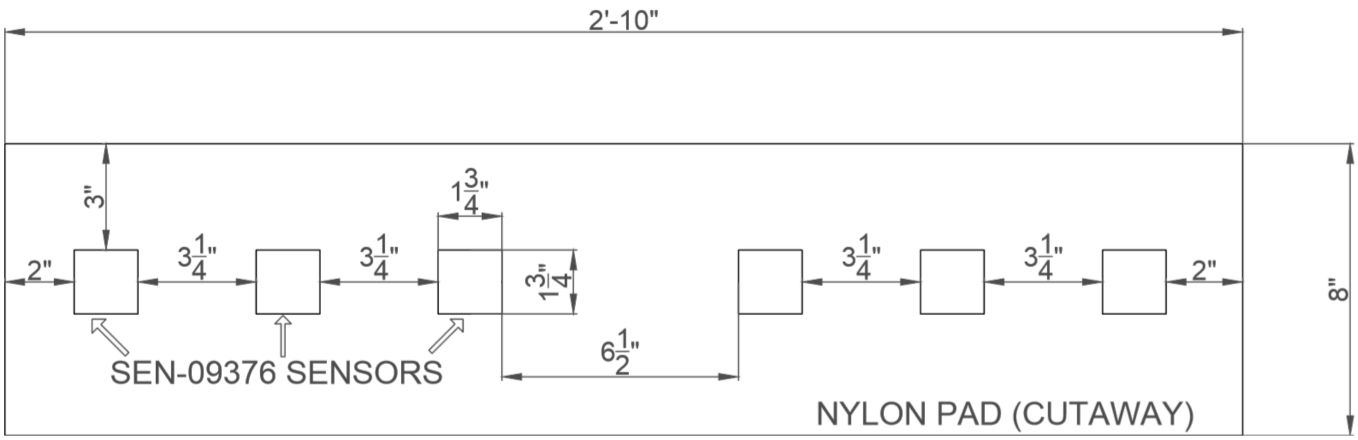


Figure 3. Mechanical Placement of Force Sensors

2.2.3) **Side Pouch (The Hub)**

The side pouch, to be hung on the side of the bed, will be attached to the main sensor pad via a durable, flexible link. The both the link and the Hub will be encumbered in the exact same manner as the main pad (anti-static sheet housed in nylon). The power will be supplied through the Hub and the port will be protected by a rubber flap, as to protect from moisture, spilling, etc. Because we have no way to estimate the exact size of our finished PCB at this point in the design process, we have no way to tell exact dimensions of the Hub. However, we know that we want a 1”X12” link from the base of the hub, to the side of the pad. The thickness is perfect for housing all needed wires and the length isn’t too short or too long for nursing beds[7] and will not pose a nuisance to medical staff or patients.

2.2.4) **Material, Straps, and Production**

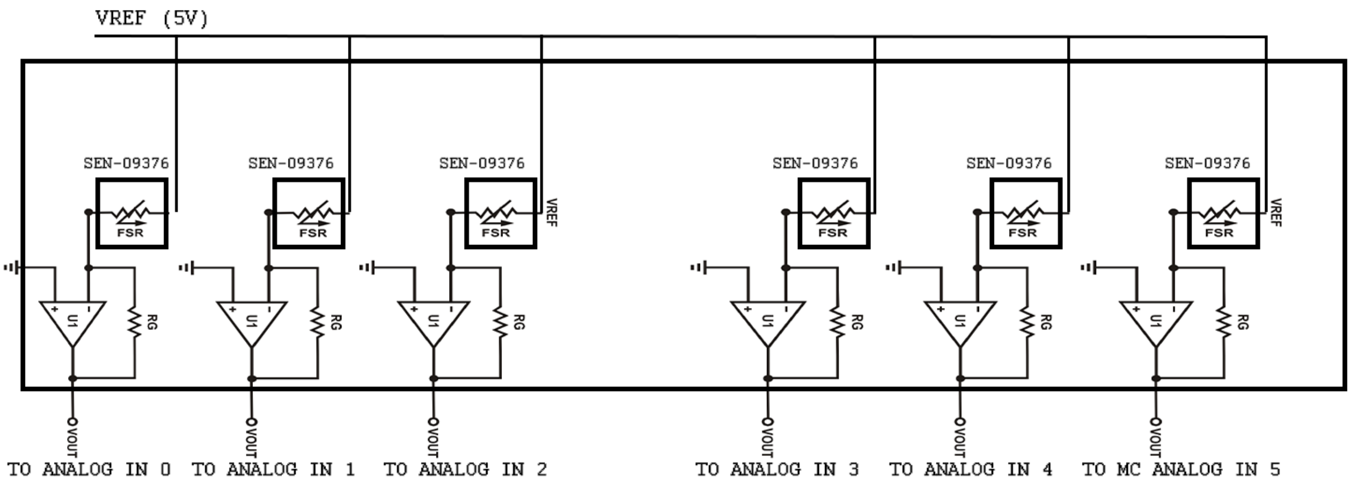
The sensors will be taped with Scotch™ brand double-sided laminating adhesive (to avoid air bubbles) to an anti-static Uline™ foam sheet, which will be housed by the outside, nylon material.. The outgoing Vout and incoming Vref wires will be adhesively glued to the same foam sheet. All components within the foam sheets will be deflated and taped together (this is how we will test the sensors. The outer-Nylon housing with then be sewn in such a way to waterproof the foam inside. The adjustable straps and pouch will be sewn and attached to the pad in a similar manner. The straps will be placed in the geographic center of the height of the pad (4" up on both sides - for maximum stability) and the pouch will be placed 2" up from the bottom of the pad on the right side.

2.3) **Block Design**

2.3.1) **Sensor Module**

Our sensor module, depicted in Fig. 4, will be housed in an 8 X 32" pad made of nylon for its durable, waterproof properties (see 2.2.2 for sensor layout). The pad will be designed to sit under the buttocks area of the patient and will send Vout to each respective analog-in of the microcontroller (see Control Unit) as shifts in weight distribution occur with each individual sensor. We have one row of six sensors, three sensors on one side and three on the other. This is a design choice due to the three main positions for patients: lying mostly on the right side, lying evenly distributed between the two sides, and lying on the left side. The dimensions 8 X 32" are relevant because 8" in the approximate height of the buttocks and 32" is the width of the most thin hospital beds[11], with 2" cut off from each side.

The sensors we are using are Sparkfun SEN-09376. These sensors have a large area relative to other sensors on the market, and can handle loads from 100g- 10kg[10], which is just the range we desire(see calculations). As described above, we will be using 4.7k resistors due to the sensor's apparent linearity at higher force (around 1 kg)[10]. Note that Fig. 4 shows the sensor pad to scale, but not the electrical components, such as the OP amps and RG resistors. In fact, these components will be housed in the Hub (side pouch) of the pad (also part of the Sensor Module). The only electrical components running through the pad are the Vref wires coming from the Power Module, and the Vout data wires going into the Control Unit. The Sensor Module is powered by the Power Module.



PAD AND SENSORS DRAWN TO SCALE, ELECRICAL COMPONENTS ARE NOT
RG = 4.7kΩ
SENSORS: 1.75 x 1.75 INCHES

Figure 4. Sensor Module Schematic

	Requirements	Verification
1.	Pad is waterproof from 3 cups of water.	A. Take the pad to an area where spillage can be collected such as a drain or hover over a dish. Leave the pad plugged in B. Take 3 cups of water and gently lather the top of the pad C. Let liquid sit for 5 minutes D. Make sure the buzzer does not sound. If so FAIL, if not, dry the pad and PASS E. As an alternative, dry the pad and check voltages going out from the sensors. If any sensor reads 0V, FAIL, else PASS
2.	Sensors do not malfunction at weights up to 120kg	A. Obtain twelve 10 kg weights or equivelant B. Obtain a firm mattress or enough firm pillows to support the whole 8X34" area of the pad C. Place the pad on the support D. Linearly distribute the weights on the pad E. Wait 5 minutes F. Make sure the buzzer does not sound. If so FAIL, if not, PASS G. As an alternative, check voltages going out from the sensors. If any sensor reads 0V, FAIL, else PASS
3.	Pad stays in place upon 10 flips	A. Obtain a bed with a sheeted mattress and railings, as hospital-grade beds have B. Stretch the pad the length of the bed at a reasonable tautness. The pad will be placed under the buttox so place accordingly C. Strap the adjustable velcro straps to the railings of the bed, keeping taut D. Mark the initial position of the strap E. Lay on the bed and a trained medical staff flip you 10 times. If the velcro comes undone, the strap moves >1 inch in any direction, or the pad becomes unplugged mark as FAIL, else PASS
4.	Sensors and wires do not lead to discomfort	A. If any discomfort from Requirement 3 was felt in the flipping process from the internals of the pad, mark FAIL, else PASS
5.	Sensors do not move in a way that would lead to 0V being sent to the MC	A. If a buzzer was heard at any point during the verification of Requirement 3, FAIL, else PASS

Sensor Module Calculations

The calculations that are being shown were used to estimate how many sensors we would need for our pad. We were not able to obtain the sensors we are intending to use to take realistic data, so instead we used a bathroom scale to take very rough measurements of how your weight changes when you are standing up and laying down (as you would be if you were laying on our pad) on a hardwood floor versus when you are on a firmer bed.

Measured Weight While Varying Position and Scale Surface	
Condition	**Measured Weight (lbs)**
Standing on hardwood	169.6
Standing on firm bed	15.2
Laying on hardwood	69.2
Laying on firm bed	10.01

Figure 5. Table of measured weight on bathroom scale when standing and laying on a hard and soft surface.

We then took these measurements and compared how the weight changed from measuring on the hardwood versus a firm bed for each position.

$$\frac{Standing\ on\ Bed}{Standing\ on\ Hardwood} = \frac{15.2lbs}{169.6lb} = 8.96\% \quad (1)$$

$$\frac{Laying\ on\ Bed}{Laying\ on\ Hardwood} = \frac{10.0lbs}{69.2lb} = 14.45\% \quad (2)$$

Comparing these two percentages we can see the bathroom scale was not very accurate when taking measurements on a softer surface, but it is accurate enough to estimate the amount of need sensors.
We then compared the measured standing weight on a hard floor versus the laying weight on a firm bed. This percentage can be used for estimating the average weight that is distributed near a person’s hip.

$$\frac{Laying\ on\ Bed}{Standing\ on\ Hardwood} = \frac{10.0lbs}{169.6lb} = 5.896\% \quad (3)$$

Then we took the average weight of a person of 80.7 kg

$$Average\ Weight\ Distributed\ Near\ Hips = Average\ Weight\ of\ Person \times 5.896\% \qquad (4)$$

$$Average\ Weight\ Distributed\ Near\ Hips = 80.7\ kg \times 5.896\% = 4.76\ kg \qquad (5)$$

We intend to use Force Sensitive Resistor SEN-09376 which can measure 0-1000 kg. So we use 6 sensor, each sensor will see on average:

$$\frac{Average\ Weight\ Distributed\ Near\ Hips}{Number\ of\ Sensors} = \frac{4.76\ kg}{6} = 793.33\ g/sensor \qquad (6)$$

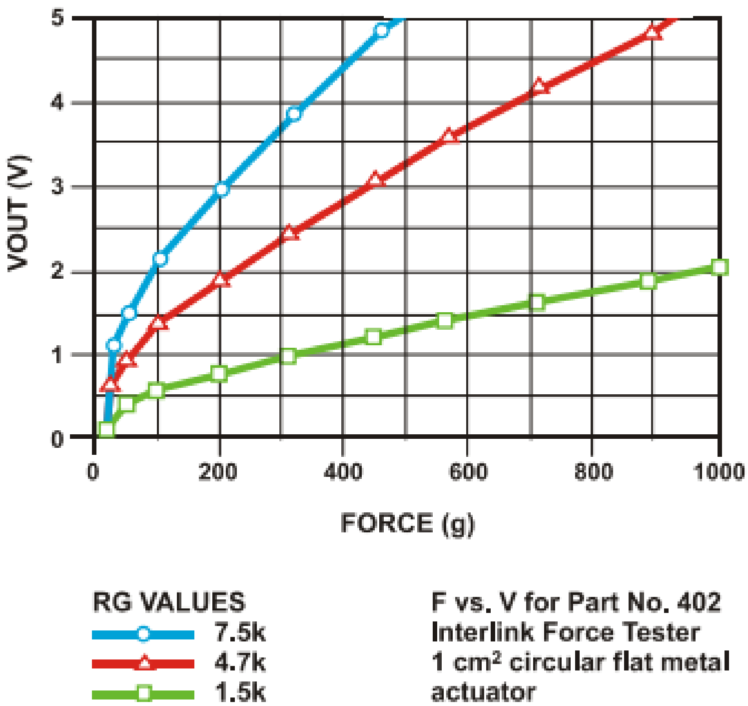


Figure 6. Voltage vs. Force for Different Resistor Values of SEN-09376

We used this plot to decide what value for RG we will be using which is shown in our circuit schematic. We will choose 4.7k because this value has the most linear plot which will allow the sensor to most accurately see changes in force.

2.3.2) **Power Module**

The power module is responsible for suppling our system with a constant 3.3 V. To meet the standards and regulations of a nursing home, our design would normally obtain its power from a hospital grade 10’ Nema 5-15 plug. However, to demo our prototype we will use any USB port to supply power to our lithium-ion battery charger. Our complete design will draw around 300 mA which I will breakdown when discussing our battery decision.

Lithium-Ion Battery Charger

As mentioned above, our prototype will send power from an USB port to a lithium-ion battery charging. The battery charger we have chosen is the Micro Lipo LiPoly Charger simply because of its low cost, small size, and because it’s ability to fully charge our battery in 6.25 hours.

$$\frac{Battery\ Capacity}{Charge\ Rate} \times 125\% = \frac{2500mAh}{500mA} \times 125\% = 6.25\ hours \tag{7}$$

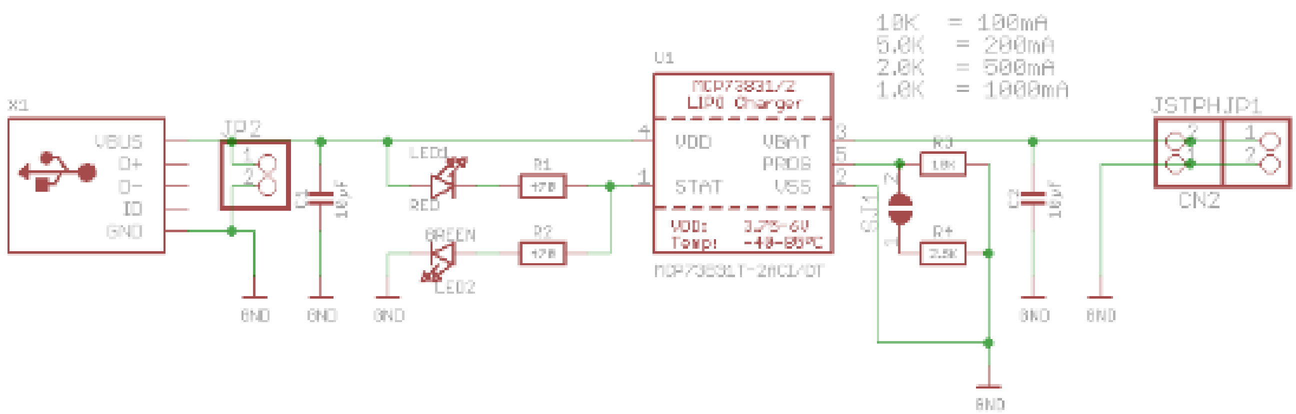


Figure 7. Battery Charger Schematic[12]

	Requirements	**Verification**
1.	Be able to fully charge a 2500mAh battery within 8 hours. (length of battery life)	A. Fully drain the battery B. Test that the battery is outputting 0V with a multimeter. C. Plug battery into charger and use a stopwatch to measure the time it take for the battery to reach 4.2 volts (fully charged voltage).

Lithium-Ion Battery

In a nursing home, it is possible that the patients' beds are moved to other rooms, so we knew we would have to use a battery so our device could have isolated power. We also did not want nurses to have to constantly be buying new batteries, so we went with a lithium-ion battery since it is rechargeable. The total power budget we calculated was 2500 mAh. The Wi-Fi module uses 250 mA, the microcontroller uses 20 mA, the buzzer uses 3.5 mA, the sensors uses 1 mA each (6 mA), and the LED uses 15 mA. Our complete design uses a total of 294.5 A. The needed capacity of the battery calculation is shown below in equation 8.

$$\frac{Battery\ Capacity}{Design\ Current\ Draw} = Hours\ of\ Use\ = \frac{2500mAh}{294.5} = 8.48\ hours \quad (8)$$

Our device will be able to last a typical shift in a nursing home before the battery needs to be recharged. This battery supplies 4.2V-3.7V.

	Requirements	Verification
1.	Stores 2.35-2.5 Ah of charge	A. Connect a fully charged battery to a constant current test circuit. B. Discharge the battery at 300 mA for 8 hours. C. Use a multimeter to make sure the battery voltage is above 3.7 V

Voltage Regulator

Our voltage regulator will supply a constant 3.3V to the rest of our system. The voltage regulator will take in 4.2-3.7V from the lithium ion battery. We will use the LM1117 3.3V 800mA voltage regulator chip to handle the maximum current from our system.

	Requirements	Verification
1.	Must maintain an output voltage of 3.3V +/- 1% from a input voltage varying from 4.2-3.7 V.	A. Construct a constant current circuit using the voltage regulating chip and draw 300 mA of current. B. Use an oscilloscope to verify that the output voltage maintains a 3.3V +/- 1% range

LED

The LED in the power module will be used to verify that our system is still powered. This is done by connecting it to the output of the voltage regulator, and will be lit as long as there is power to our system. The LED that will be using is a basic green 5mm LED.

	Requirements	Verification
1.	Must not draw more than 20mA of current.	A. Use a multimeter to test the current running through the LED.

2.3.3) **Control Unit**

The Control Unit, consisting of our very important microcontroller, will garner information from the six sensors of the Sensor Module and make the call if a patient has been flipped (more specifics below). If so, it will send a UART (D5 - TX pin) signal to the communication module which will, in turn, send the signal the appropriate packet to the given healthcare professional's mobile device (see section 2.3.4**:** Communication Module). The control unit also keeps track of the two-hour timer for its respective Turning Tracker, rather the phone. This was a choice made for prevention of multiple tracking agents receiving multiple timers based on communication strength from the device. The Control Unit also harbors the emergency buzzer. The Control Unit is powered by the Power Module.

Buzzer

The buzzer we will be using is the Adafruit LLC 1739. The buzzer is necessary to alert the patient and/or medical staff that, in short, "something is wrong". The buzzer will sound is such emergency situations as:

- Malfunction of sensors (0V is read for 1 minute of any of the analog inputs of the microcontroller)
- No acknowledge signal returned by any mobile device; In some cases, the Communication Module will try to send the TCP/IP packet to one or more healthcare professionals. If no acknowledge signal is sent back after 15 minutes of trying to send the signal every minute, the buzzer will sound
- Battery is low

This buzzer makes a single tone that can be made once or repeated based on what is wrong with our device.

	Requirements	Verification
1.	The buzzer sounds when a voltage of 3.3V is sent to it.	A. Use a programmable power source to send 3.3 V to the buzzer. B. Ensure that a tone can be heard.

Microcontroller

The microcontroller of choice is the AVR ATmega328 with Arduino Optiboot (Uno). Many factors have gone into this choice including[13]:

- Compatibility with the Arduino Uno; we can directly compile our code from the Arduino onto our microcontroller, and remove the microcontroller for use with our PCB. This is a huge factor in debugging our controller, and will cut much of our debugging time. Plus the Arduino IDE is relatively intuitive and easy to use, with much documentation online. We also will not need to purchase a programmer, which can exceed $50.
- Atmel is a great company with many online resources, code snippets, and examples online. The ATmegas series gets nothing but good reviews from Electrical and Computer engineers
- 6 necessary analog inputs for our 6 sensors
- Cheap, costing around $5 USD
- Fast, clocking at 20 MHz
- Three timers, making our 2-hour timer able to be its own task without the danger of unwanted interrupts
- 2 UART outputs, being necessary for communication with our Communication Module
- Plenty of memory; 2kB of SRAM, 32kB of ROM and 1024B of EEPROM. We are going to have a somewhat large program which makes this desirable

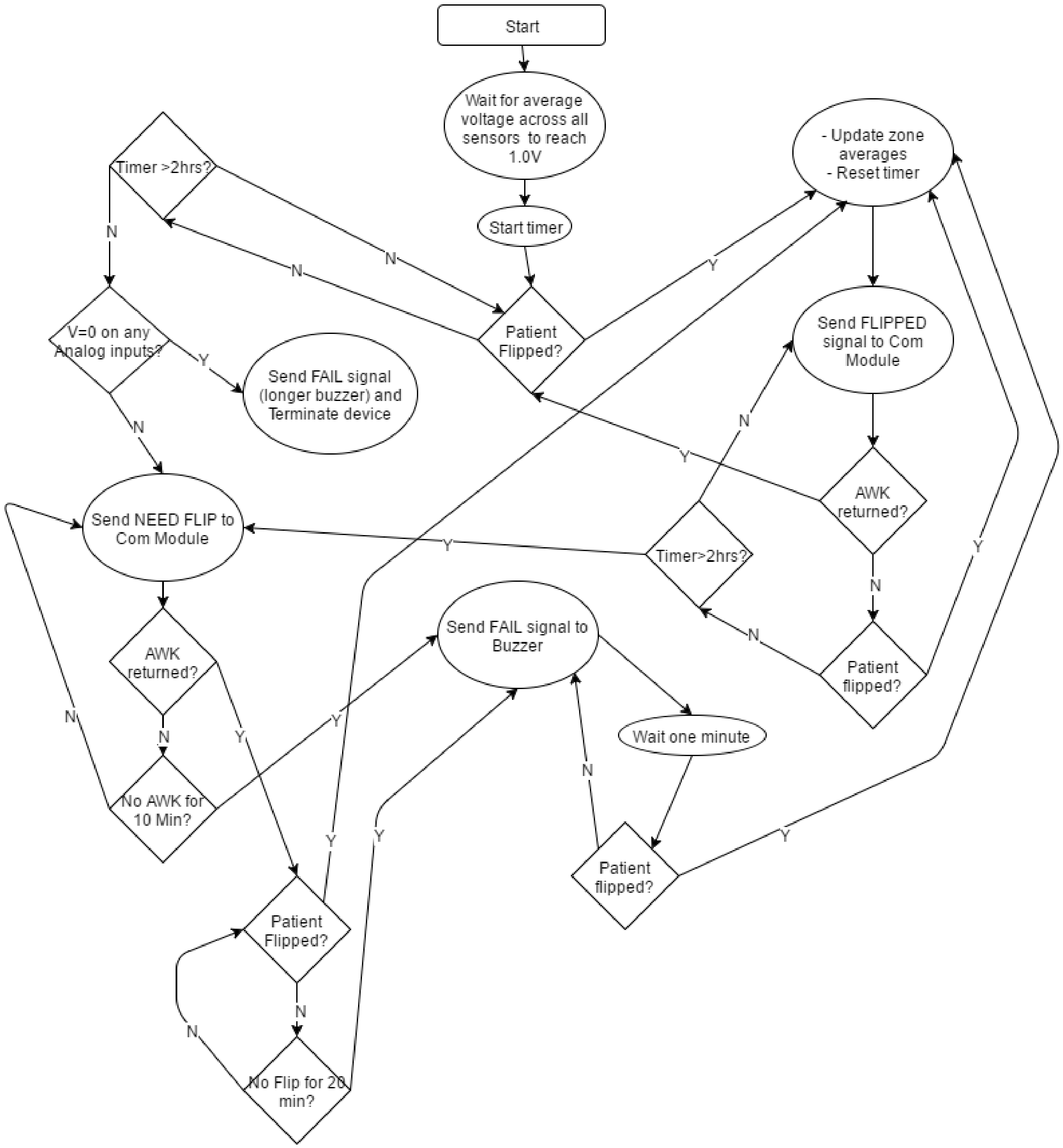


Figure 8. Flow chart of Microcontroller software

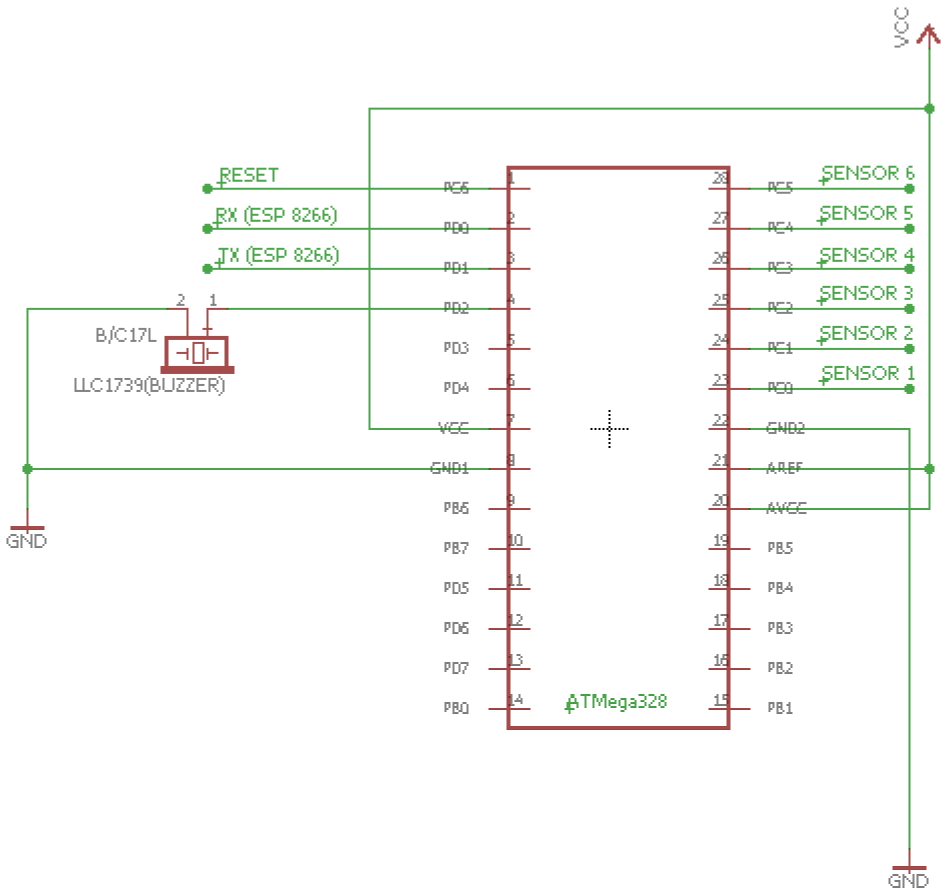


Figure 9. Microcontroller Schematic

Fig 8. shows the basic flow of our sequential Microprocessor program. Note that a more realistic application of this program would call for parallel processing power. The program starts by waiting for a human to lay down on the pad. Then, the two-hour timer is started. The program breaks the six sensors into three groups; zone A consisting of the two far left sensors, zone B consisting of the two middle sensors, and zone C consisting of the two far right sensors. The microcontroller takes the average voltage reading values of each of these zones for 2 minutes, saves the value, and then continues instruction. If an average voltage of any individual zone is shifted by += 50% (as explained in the tolerance analysis) of its original value, a flip is registered. This check happens many, many times through standard instruction.

Once a flip is measured, the new averages are computed over a two minute interval, the timer is restarted, and a FLIPPED signal is sent to the Communication module via the UART RX port. If an acknowledge signal is not registered via the UART (D0 - RX) port of the communication module, the controller continues to try (periodically seeing if another flip occurred in the meantime) to send the acknowledge signal until it another flip occurred or another flip is needed. When a flip is needed, the controller first does a maintenance check to ensure than all sensors are working properly. Then it sends a NEEDS FLIP signal to the Communication module. Similarly to the FLIPPED case, it keeps trying. It the signal fails to send or a patient is not flipped over a certain interval, the buzzer is sounded.

	Requirements	**Verification**
1.	The microcontroller registers when a patient's weight distribution is shifted from any zone by +-50%	A. Through the microcontroller's code, locate the subroutine in which registers a patient's flip B. Modify the code to output a debug signal to PC6 of the microcontroller. C. Load the microcontroller with this modified software D. Hook up the sensors, microcontroller and an LED sprouted out of PC6 E. Place 10 kg on each sensor from analog inputs A0 and A1. F. Wait 5 minutes and place 10 kg on each sensor from analog inputs A4 and A5. G. If the LED lights up PASS, else FAIL
2.	The microcontroller’s clock is uninterrupted and accurate to two hours +- 3 minutes	A. Modify the microcontroller’s code to output a debug signal when the two hour timer is depleted. B. Load the microcontroller with this code C. As before, set up the breadboard appropriately with the controller, the sensors, and the LED sprouting out of PC6 hooked up. Power 3.3V D. On a hard surface, place 5 kg on each sensor E. Start a stopwatch for two hours F. If after two hours +-3 minutes, the LED blinks PASS, else FAIL

2.3.4) **Communication Module**

The Communication Module is essential in the communication between the medical professional and the device itself. This module consists of the integrated Wi-Fi IC and the phone application on the professional's mobile device. The Control Unit outputs two main signals to the Wi-Fi IC by means of UART RX. Either a patient is FLIPPED or a patient NEEDS_FLIP. This signal is sent to the Wi-Fi IC, which, in turn, sends it to the mobile device. As part of the mobile app, when this signal is received, the phone then sends an AWK signal within the packet to the Wi-Fi IC which then sends it to the Control Unit through the UART TX pin for further processing (see Control Module).

The TCP/IP packet that will be sent/received has the form [NULL, ID, AWK, TYPE]. The first entry, NULL is left blank in case we wish to add the room number of the device later. Being that we are focusing our scope to one phone and one device, we left this blank. ID is the ID of the specific device. This is another entry that is irrelevant with one device. The next entry, AWK, is sent as FALSE and returned as TRUE when the mobile device receives (and processes) the signal. The last entry, TYPE, is either FLIPPED or NEEDS_FLIP. FLIPPED is for use with an extenuation of this project, when we want to restart all of the staff's timers. NEEDS_FLIP is important to alert the staff that a flip is necessary. This signal will be sent once a minute for 20 minutes until a flip occurs. Else, a buzzer will sound, at which point the patient will likely call his or her nurse.

Wi-Fi IC

Having cost being one of our main concerns of this project, we have opted for the ESP8266 Wi-Fi chip. This chip is TCP/IP capable, equipped with a built in microcontroller. Range is key with this project and contributes highly to the overall reliability of the device. The ESP8266 has a range of 479 meters[14], which is far more reach than we would need in our applications. The packet size we will be using the minimum 1024-byte packet length along with the built-in 802.11n Wi-Fi protocol.

Being that the communication between human and device must be the utmost reliable, in accordance with health standards[7], we have selected the Adafruit HUZZAH ESP8266 breakout board as the central hub of our Communication Module.

The board has an incredible amount of features that garner it to be most appropriate for our product:

- Cheap, costing about $10
- CE and FCC emitter certified[14], adding reliability to our design
- Deep sleep mode, saving a considerable amount of battery. We can program the board to only wake up only when it must send (and receive) a signal.
- As with our microprocessor the board is Arduino-accessible. We can test, debug, and experiment with our Wifi IC in ways we never could with a raw ESP8266 chip
- Breadboard-friendly, adding to the ease of debugging
- Small, at 38X25X5mm[14]
- PCB-integratable; we can solder this piece directly onto a PCB, saving space in our side pouch
- Thorough documentation, resources, and examples.
- Fast, sending/receiving SPI at 4.5Mbps and UART at 115.2kbps[14]

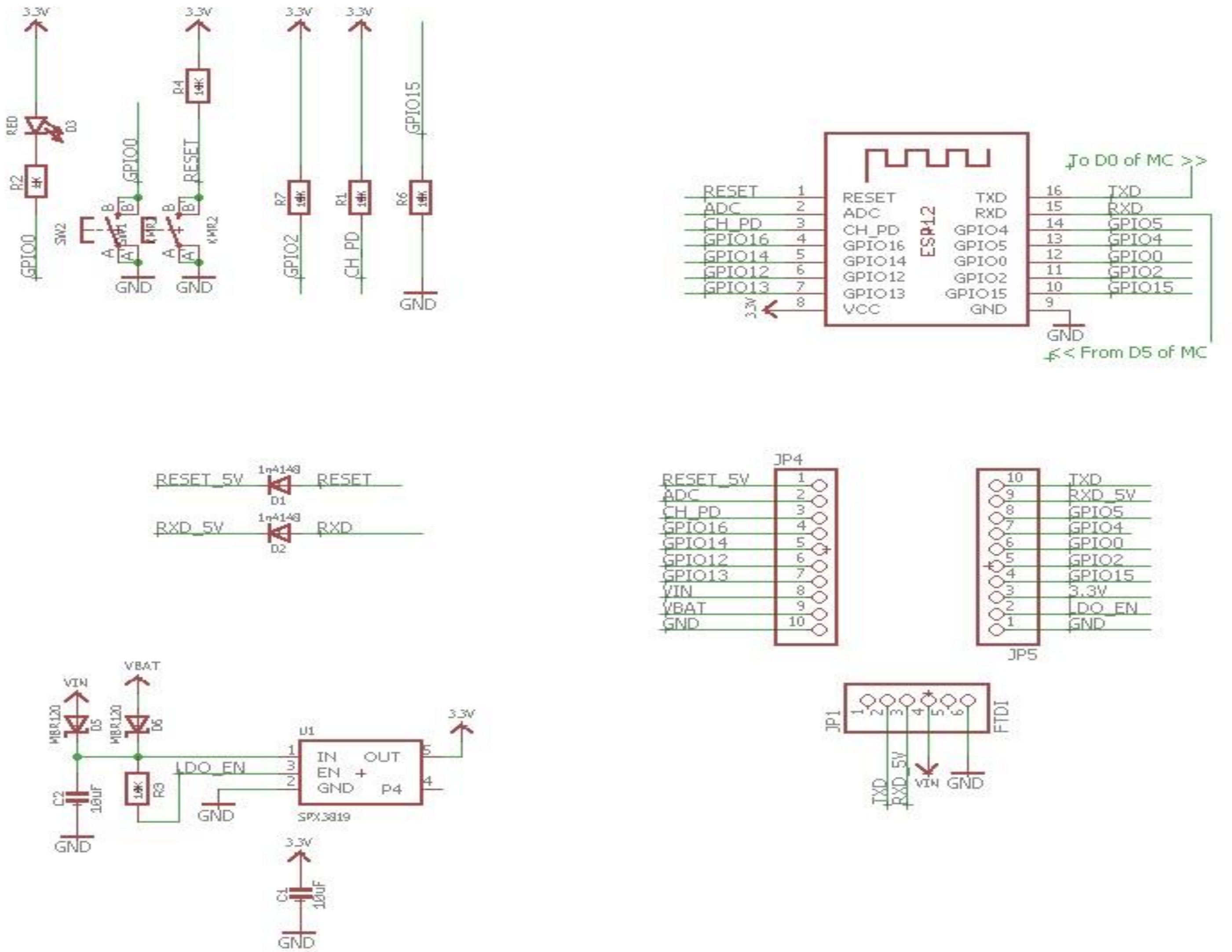


Fig. 10 Schematic of Adafruit ESP8266 breakout[7]

	Requirements	Verification
1.	Wi-Fi IC both sends and receives signal at distance of at least 300 meters	A. Open the pad’s mobile application and link your mobile device with pad B. Separate your phone from the pad by approximately 300 meters within the same Wi-Fi network. As a reference, the Wright entrance of the ECE building is about 300 meters from the Senior Design Lab C. Using Arduino IDE, program Adafruit ESP8266 chip to send a test signal, NEED_FLIP using its built-in debugger D. If your device does not read NEEDS_FLIP, FAIL E. Open the debugger of the Wifi unit and assure that the AWK = TRUE within the packet. PASS if so, else FAIL
2.	Signal is continually sent for 20 minutes when phone is out of range or off network	A. Open the pad’s mobile application and link your mobile device with pad B. Take phone 500 meters or more away from Wi-Fi unit. C. Using a similar procedure as in 1., send a test signal D. Wait 5 minutes, then come back in range E. Assure that a NEEDS_FLIP signal is registered on your mobile app F. If so, PASS else FAIL

Phone Application

The phone application will go hand-in-hand with the communication module. Having such large time and resource constraints this semester, we only plan on having the phone application display a ‘flip’ and a ‘needs flip’ signal for a single pad. If this project were to carry on, we’d like to see a plethora of added features on the application. This user-friendly phone application would show all room numbers graphically next to their respective timers on multiple tracked devices for multiple phones. This goal, however, is a far stretch for two people and two months. Our software for the application will be initially for Android and will be programmed with WebStorm. Simply stated, it receives the TCP/IP packet and displays the data it sees, sending back an acknowledge signal to the device. We have noticed the impossibility with the compatibility of multiple mobile devices at this point without multi-processing power from our micro controller. Each medical person’s timer would be significantly asynchronous without such parallel processor. We would need a processor to monitor if a patient needs flipping and a separate processor to decide if a patient has been flipped

	Requirements	**Verification**
1.	Phone displays “Flip” upon appropriate flip	F. Open the pad’s mobile application and link your mobile device with pad G. Strap pad on firm mattress on lay down with the pad positioned under the buttox area H. Have a certified nurse assistant or other trained medical staff flip you from laying down up to laying on your right side I. Wait 1 minute. If your mobile device reads “Flip” PASS else FAIL
2.	Phone displays “Needs Flip” after two-hours of inaction	G. Open the pad’s mobile application and link your mobile device with pad H. Let pad sit for two-hours I. If your mobile device reads “Flip” FAIL J. If after two-hours +- 5 minutes your phone reads “Needs Flip” PASS else FAIL

2.4) **Tolerance Analysis**

The most apparent and import tolerance of our project to consider is the the tolerance with determining the appropriate weight shift that warrants a “flip” signal to be sent.

We defined in our Control Module section that we there are three zones of sensors, as seen by the microcontroller. Zone 1, being the two far-left sensors, zone 2 being the two middle sensors, and zone 3 being the two far-right sensors. Through this section, we will discuss the appropriate transfer of weight (voltage) from section-to-section in order to accurately define a ‘flip’.

We define three main positions a patient can take on. Position 2 is has most weight distributed between sensor 3 and 4 (zone 2), position 1 has most weight distributed between sensors 2 and 3 (between zone 1 and 2), and position 3 has most weight distributed between sensors 4 and 5 (zone 2 and 3). Let sensors 1, 2, 3, 4, 5, and 6 be the sensors from zone 1 to 3.

Using the data accumulated in section 2.3.1 (Sensor Module)

$$Position\ 1\ weight\ distribution\ = \frac{weight\ near\ waist}{\#\ of\ sensors\ affected} = \frac{4.76kg}{2\ sensors} = 2.38kg/sensor \qquad (9)$$

So in position 1, sensors 2 and 3 see an average of 2.38kg/sensor. With this amount of force on these two sensors, the output voltage will be equal to the reference voltage of 3.3 volts. Since we are using a flexible material sensor 1 will feel a force of on average half of sensor 2 and 3, and sensor 4 will feel a negligible force.

$$Sensor\ 1\ force\ in\ Position\ 1\ = \frac{Sensor\ 2\ Force}{2} = \frac{2.38kg}{2} = 1.19kg/sensor \qquad (10)$$

The shape of the Vout vs Force curve is very linear so with this amount of force, sensor 1 will output a voltage half of the reference voltage, or 1.65 volts.

In position 2, sensors 3 and 4 see the majority of the force, so now sensors 3 and 4 will see 2.38kg. Sensors 2 and 5 will now see the same force as sensor 1 did in position 1 which is 1.19kg. Now we can average the Vout for each zone in each position to find the required change in voltage for a zone to register it as a flip.

$$Zone\ 1\ Vout\ in\ Position\ 1\ = \frac{Sensor\ 1\ V + Sensor\ 2\ V}{2} = \frac{3.3V + 1.65V}{2} = 2.475V \qquad (11)$$

$$Zone\ 1\ Vout\ in\ Position\ 2\ = \frac{Sensor\ 1\ V + Sensor\ 2\ V}{2} = \frac{0V + 1.65V}{2} = 0.825V \qquad (12)$$

$$Zone\ 2\ Vout\ in\ Position\ 1\ = \frac{Sensor\ 3\ V + Sensor\ 4\ V}{2} = \frac{3.3V + 0V}{2} = 1.65V \qquad (13)$$

$$Zone\ 2\ Vout\ in\ Position\ 2\ \ = \frac{Sensor\ 3\ V + Sensor\ 4\ V}{2} = \frac{3.3V + 3.3V}{2} = 3.3V \tag{14}$$

Now we can see how the zones output voltages change as a position changes.

$$Zone\ 2\ Position\ 1\ vs\ Position\ 2\ \ = 3.3V - 1.65V = 1.65V \tag{15}$$

Zone 2 experiences a smaller change than zone 1, and sees a voltage change of -50% of its previous values. So we have to ensure that our microcontroller registers a flip when it notices at least a average zone voltage change of +/-50 %.

3 Costs and Schedule

3.1) **Cost Analysis**

We have decided that our time is valued at 40 dollars/hour, and we will be spending 15 hours/week individually on average. We expect to finish a working prototype of every module except the mobile app. So a total of 80% of our project will be completed.

$$\frac{40\$}{hr} \times \frac{15\ hr}{wk} \times \frac{16\ weeks}{0.8} \times 2 \times 2.5 = \$60{,}000 \tag{16}$$

The cost for our parts need for our prototype are shown below in **Figure 10**

Part	Cost
Adafruit ESP8266 Communication Modules	$9.95
ATMega328 with Arduino Optiboot (Uno)	$5.50
SEN-09376 Force Sensitive Resistor	$59.75 (6*$9.95)
Adafruit Lithium-Ion Polymer Battery	$14.95
Adafruit Micro Lipo Usb LiIon	$5.95
Adafruit LLC1739 Buzzer	$0.95
LED-Basic Green 5mm	$0.35
PCB(pcbway) 5 boards	$10.00
Anti-static Uline Foam Sheet	$28.00
Water Repellent Nylon Cloth	$18.40
Total Cost of Parts	$153.80

Figure 11 Table of Cost of Parts

The grand total is calculated below:

$$Fixed\ Cost\ +\ Cost\ of\ Parts\ =\ \$60{,}000\ +\ 153.80 = \$60{,}153.80 \tag{17}$$

3.2) **Schedule**

Week	Brad	Robert
2/27/17	Order sensors, foam, nylon, and parts to make the sensor module as well as parts for the power module.	Order parts for the control unit and the communication module.
3/6/17	Assemble the prototype for the sensor module. Being taking readings laying down on pad.	Learn/ begin programming the microcontroller with the created flowchart.
3/13/17	Integrate the sensor module with the with the control unit. Complete and order PCB.	Integrate the sensor module with control unit. Complete and order PCB.
3/20/17	Refine the sensor module based off the test with newly integrated control unit.	Refine the control unit based off the test of the newly integrated sensor module.
3/27/17	Begin soldering parts to the pcb.	Work on the data transmission for the communication module.
4/3/17	Integrate the power module with the entire system.	Continue working on the data transmission for the communication module.
4/10/17	Test the system in its entirety.	Integrate communication module with the control unit.
4/17/17	Continue testing and debugging, prepare for mock demo.	Continue testing and debugging, prepare for mock demo.
4/24/17	Prepare for presentation.	Ensure the system is ready for demo.
5/1/17	Finish the final paper.	Finish the final paper.

4 Ethics and Safety

4.1) **Development Concerns**

As stated in section 1.3, two of our three main requirements call for our product to be *reliable*. If our product doesn't work, people will simply get hurt. As section 1.2 states, more than 17,000 lawsuits are filed each year regarding bedsores. We would never want to be in the middle of this. As an ex-employer of the bio-instrumental industry, our group is very aware of the rigorous testing procedure that is to come. We need to be within 98% confidence that our product *reliably* senses a flip <u>and</u> *reliably* communicates with the nurse. To not have such confidence would be in violation of IEEE 1st and 9th codes of ethics[6]. The prior obligates decisions to be made in the best health and safety of the public. The latter obligates an avoidance of the injuring of others. We will strive to make the most reliable, well-made device we possibly can, and will be honest with those in the unlikely case of failure.

4.2) **Operational Concerns**

Continuing on with IEEE 1st code of ethics[6], we would have to accurately and promptly inform the public of the reliability of our product, and the risk involved in using it (there is always a failure rate). We would let them use honest data (conforming to the 3rd code of ethics [6]) in their decision to adapt the device.

Because we are designing the sensors to work for those who are qualified at turning patients, we would want to make sure all staff that use this product are certified to flip patients correctly. If not, it is possible that a flip could be mis-registered, leading to a fault in a flip cycle. This conforms with the 6th code of IEEE code of ethics[6]; we would have to issue a “disclosure of pertinent limitations[6]” to make sure those who use the device are certified healthcare providers.

We would never want to see our device as a way for medical staff to cheat their way out of work, or “fake a flip”. With this in mind, we will make it harder to cheat a flip, then to perform a flip altogether. For this, we will not rely on any accelerometer in our design (to avoid cheating by shaking), and will make sure continuous force differential is the main sensing device used to measure a flip (as seen in the Control Module).

4.3) **Healthcare Standards and Federal Regulation**

Besides sensing and communicating, power is also a concern. We have to make sure we have a reliable source of power that would never die without warning, or pose any electrical risk to the patient (conforming with the 9th code of ethics[6]). This is why we may choose to use a rechargeable battery with an outside converter, which buzzes when low in power[7].

References

[1]K. Vanderwee, M. Grypdonck and T. Defloor, "Alternating pressure air mattresses as prevention for pressure ulcers: a literature review", *PubMed Health*, 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0025720/. [Accessed: 24- Feb- 2017].

[2]"1. Are we ready for this change? | Agency for Healthcare Research & Quality", *Ahrq.gov*, 2017. [Online]. Available: https://www.ahrq.gov/professionals/systems/hospital/pressureulcertoolkit/putool1.html. [Accessed: 24- Feb- 2017].

[3]C. Paul, "Pressure Sore Insights", 2017.

[4]B. Aria, "Problems in the Speech Therapy", 2016.

[5]A. Yassin, "Problems in RES", 2016.

[6]"IEEE IEEE Code of Ethics", *Ieee.org*, 2017. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 24- Feb- 2017].

[7]*Electrical Requirements for Health Care Facilities*, 1st ed. Los Angeles: Techbreifs, 2017.

[8]J. Mannheim, "Turning patients over in bed", *Medlineplus.gov*, 2017. [Online]. Available: https://medlineplus.gov/ency/patientinstructions/000426.htm. [Accessed: 24- Feb- 2017].

[9]C. Lyder and E. Ayello, "Pressure Ulcers: A Patient Safety Issue", *Ncbi.nlm.nih.gov*, 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK2650/. [Accessed: 24- Feb- 2017].

[10]*FSR Force Sensing Resistor Integration Guide and Evaluation Parts Catalog*, 1st ed. Camarillo, Ca: Interlink Electronics, 2017.

[11]*Guidance for Industry and FDA Staff*, 1st ed. Rockville, MD: CDRH, 2017.

[12]"Adafruit Learning System", *Learn.adafruit.com*, 2017. [Online]. Available: https://learn.adafruit.com/assets/17043. [Accessed: 25- Feb- 2017].

[13]*ATmega48/88/168 DataSheet*, 1st ed. Atmel, 2017, p. 55.

[14]L. Ada, *Adafruit HUZZAH ESP8266 breakout*, 1st ed. Adafruit, 2017.

[15]D. Gardner and J. Suresh, *Example Design Document*, 1st ed. 2017.