Electronic Badge for Career Fairs

ECE 445 Design Document - Spring 2020

Team 37:

Pragya Aneja - Communication RV, Schedule, Power RV

Ninad Godbole - High-Level, Control RV, Costs

Varad Khandelwal - Power RV, Block Diagram, Tolerance Analysis

TA: Shaoyu Meng

Table of Contents

1. Introduction

1.1 Objective

Career fairs are an excellent place for students to connect with recruiters from a variety of companies and pursue roles that interest them in the most direct way possible. However, most students spend the majority of their time waiting in long lines for the more popular and well-known companies, which causes them to miss out on other promising companies [1]. This aspect hinders a student's ability to get as much benefit from a career fair as he or she should be able to. Additionally, as lion share of time goes away in standing in line for big companies, several promising but small companies get discouraged due to lower student volume at their stall. Another time consuming aspect of career fairs is the requirement of companies for the students to fill out electronic forms regarding necessary contact information for employment. This process is something that is repeated every time the student stands in line for a different company. Another issue many people overlook with regards to career fairs is the massive amount of paper that is printed for resumes [2]. Students can carry up to twenty resumes and so career fairs spawn tens of thousands of papers a lot of which get unused and wasted. In general, career fairs are plagued with a variety of problems which stem from inefficiencies in different facets.

Our main goal is to reduce a major portion of these inefficiencies by tackling the biggest problem which we consider to be the long lines and wasted time. Our solution involves building an electronic badge that students can carry around in career fairs that allows them to be placed in a virtual queue for a given company. This will grant students tremendous flexibility and allow them to be more productive with their time.

1.2 Background

As traditional career fairs struggle with inefficiencies, there has been a rise of virtual career fairs. They reduce paper wastage, have virtual queue systems and easy to apply interface for the candidates. This makes them a great alternative to traditional career fair. However, they suffer from lack of physical interaction, which is considered a key element in networking. Body language describes several qualities about a candidate and recruiters use it as a strong metric for selection [3]. Overall, in person interactions facilitate higher information richness and makes traditional career an essential aspect for recruiting. Hence, our solution to implement virtual queues aims to aid the traditional career and make it par with virtual career fairs.

Overall, virtual queues are becoming much more popular as more people are beginning to experiment with the concept. An example of this is amusement parks, which have adopted this idea by allowing visitors to queue up to different rides through a wearable [4]. This provides visitors flexibility to enjoy more rides and improves overall experience. Although easily available in other industries, the virtual queue technology isn't widely adopted at career fairs which suffers from the same problem as amusement parks. If the virtual queue system is implemented in a traditional career fair, it could have several benefits for students and companies. Career fairs are sponsored by companies and if companies witness increased efficiencies, career fairs can charge higher prices and attract more companies. On the student side, it can increase the probability of getting through the company recruiting processes, which can result in higher employment/internship rate for university. Although seemingly simple, virtual queues can provide sustainable value to career fairs.

1.3 Visual Aid

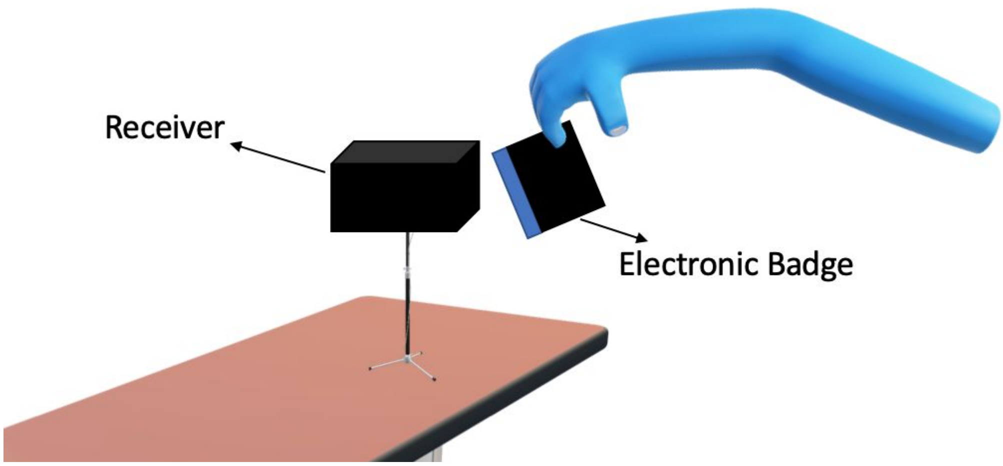

Figure 1.1: Tapping Electronic Badge on Receiver to add to virtual queue

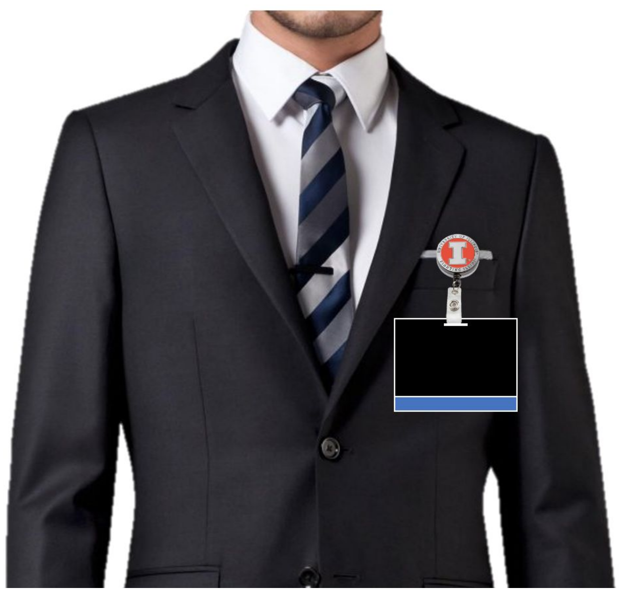

Figure 1.2: Electronic Badge with Retractable Holder

1.4 High-level Requirements

1. Electronic badge must be able to connect to the receiver with a tap (within 5 cm), and the student should be added to the virtual queue of the company.
2. Electronic badge must display the current position for the student on the virtual queue, alert them when their position is <10, and allow the student to remove himself/herself from the queue through the press of a button.
3. The receiver must maintain and process a virtual queue of upto 256 students and be able to broadcast the current positions of students in the queue.

1.5 Further Expansion for Business

Our project is a proof of concept to lower a student's physical wait time at a career fair. To demonstrate this, we decided to let the student add himself/herself to a singular virtual queue of a given company and be able to see his/her position as well as be alerted when he/she is at a position less than 10 of the queue. Ideally, if we were to expand on this for the full business purpose, we would make some changes. One of these changes is allowing the student to add himself/herself to 3 queues, with the positions of all three being shown on the badge. Another important change is increasing the queues storage from 256 to 999 positions. These changes would make it much more useful in a career fair and will definitely allow students to be more effective with their time.

2. Design

2.1 System Overview and Flowchart

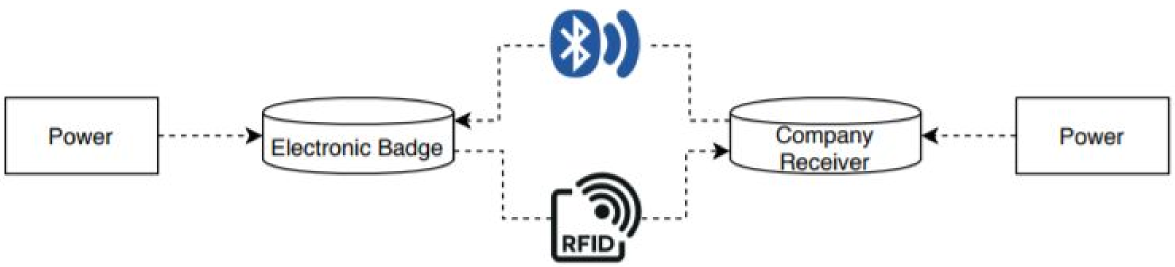

Figure 2.1: Top-level System Block Diagram

The virtual queue system will consist of 2 devices. The first is an electronic badge, which will have the student's name, year and major printed on top of an underlying PCB. It will have 1 LED screen to display the student's current position in a single virtual queue, a button to allow the student to remove her/himself from that queue (the button will only turn off badge from receiving updates about the queue. The entry on the virtual queue will be removed through a timeout method) and a buzzer to alert the student whenever their current position in the queue is < 10. It will also contain an RFID tag to send UIN information to the receiver through a tap, and a bluetooth module for long-range connection with the receiver, to receive position updates. Lastly, it requires a microcontroller to perform logic for initiating buzzing, communicating with bluetooth and RFID module.

The second device would be an electronic receiver, unique for each company. The receiver will be mounted on a stand or a table. When the student taps the badge on a receiver, he/she will be added to that company's queue. The receiver will also have a bluetooth module, to broadcast students' position in the queue, which will be received via the badge's bluetooth module. The updated positions for top 50 students in the virtual queue will be broadcasted every 2 minutes, and the updated positions for all students in the virtual queue will be broadcasted every 10 minutes. This is to ensure that students with positions < 50 get updates on their positions faster, but at the same time, the rest of the students also receive some updates while lowering the memory bandwidth consumption. After talking to the company, the student will have to re-tap on the receiver to let the virtual queue he/she is done, otherwise he will be automatically removed via a timeout method.

Below is a flowchart showing the process a student will go through:

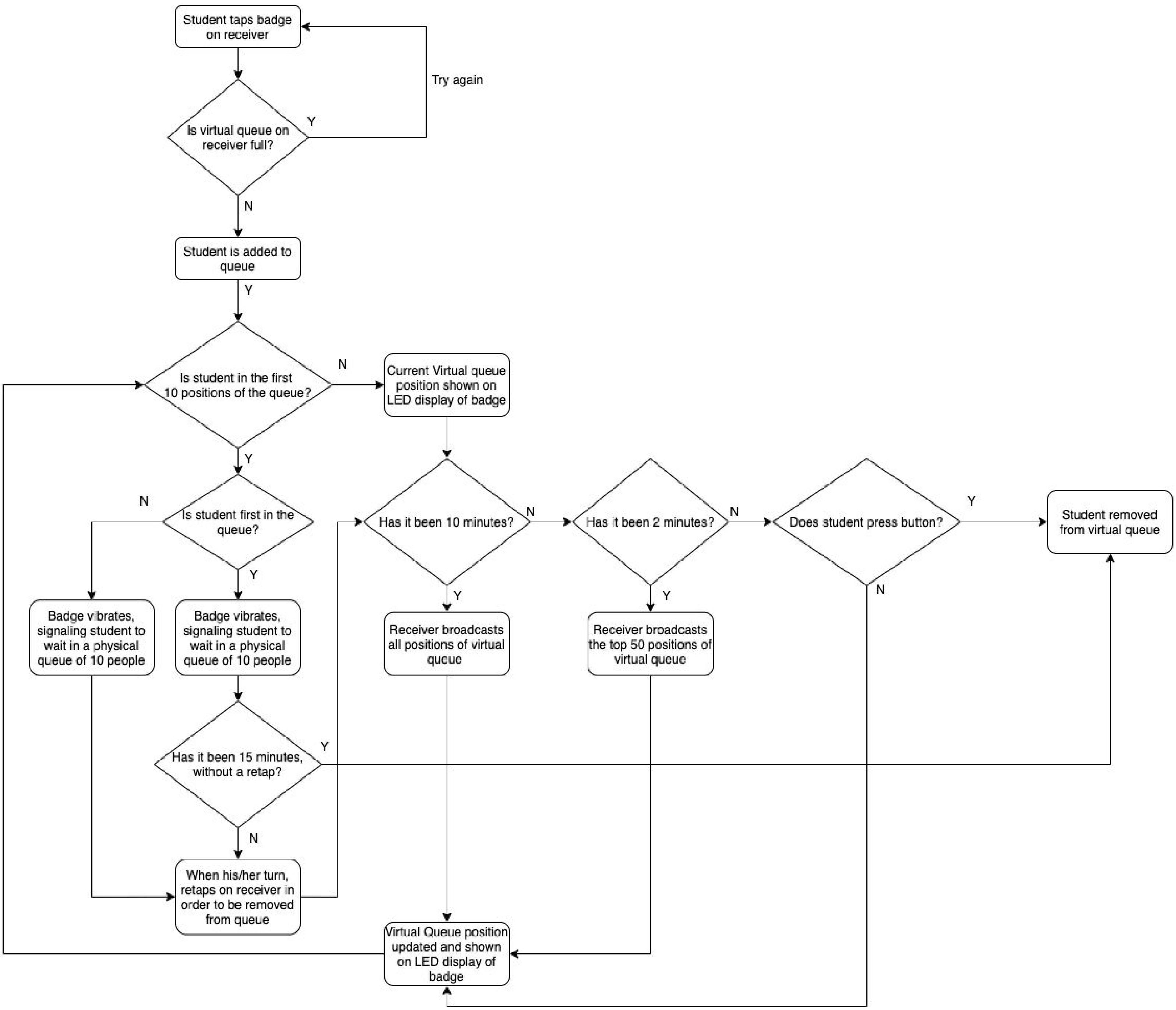

Figure 2.2: Flowchart for entire process

2.2 Electronic Badge

2.2.1 Block Diagram

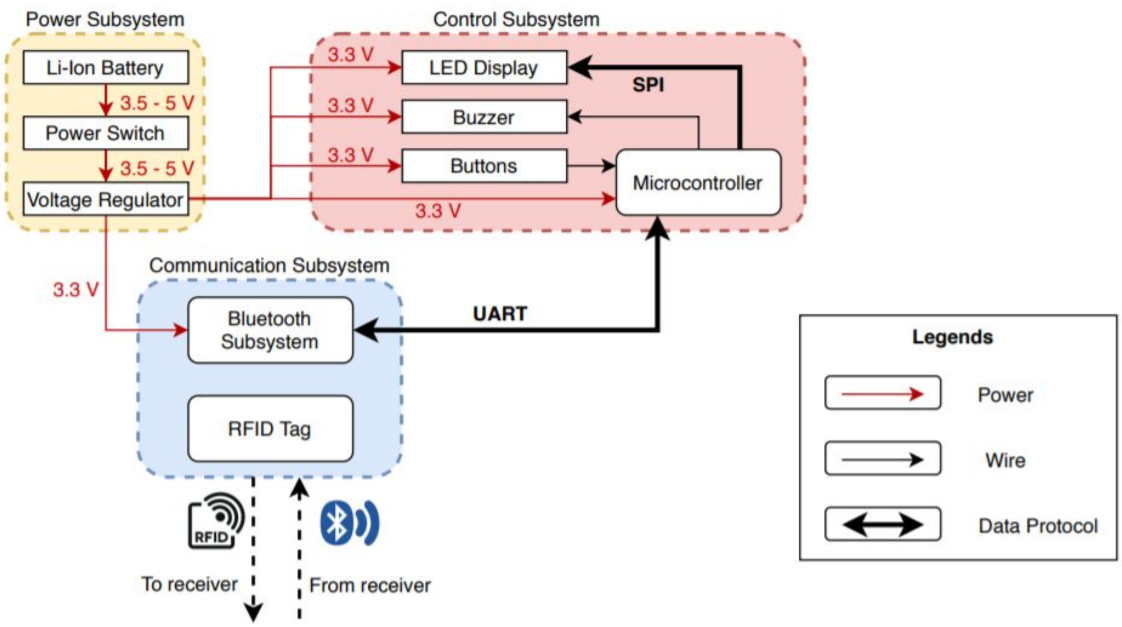

Figure 2.3: Electronic badge block diagram

2.2.2 Physical Diagram

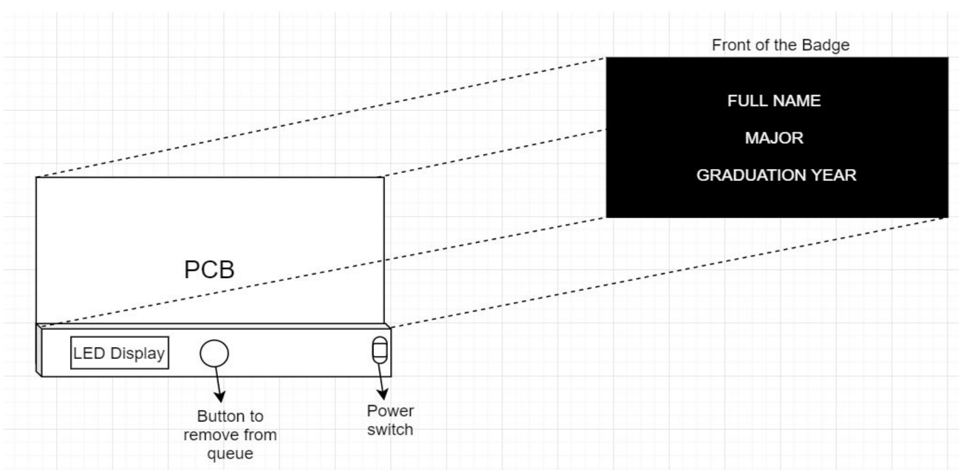

Figure 2.4: Electronic badge physical design

The black part will be hooked on the PCB and will cover the PCB board. It will be placed above the LED display and buttons.

2.2.3 Power supply

Li-ion battery

The lithium-ion battery must be able to keep the circuit continuously powered when switched on for 5-6 hours, and should be lightweight and small to fit on the electronic badge. The power supply has to supply around 3.5V-5V as all of the electrical components on the electronic badge (Microcontroller, LED display, Buttons, Buzzer and Bluetooth system) require an input voltage of 3.3V provided by connecting the power supply to a voltage regulator. Additionally the battery must be rechargeable.

Requirements	Verifications
***Requirement 1*:** Battery should be able to provide a voltage output of 3.5-5 V.	a. Connect the fully-charged li-ion battery with VDD and GND. b. Use a multimeter to check if the voltage output is in the specified range.
Requirement 2: Battery should be able to provide > 1500 mAh for 5-6 hrs.	a. Connect a fully-charged lithium-ion battery with VDD and GND. b. Draw 250 mA current from the battery for 5-6 hrs. c. Monitor output voltage battery and ensure it does not drop below 3.5V earlier.

Power switch

We don't want the device to run indefinitely, and would like to switch it off after use. Thus, we need a switch which disconnects the voltage regulator from the battery when switched off.

Requirement(s)	Verification
Requirement 1: A switch which disconnects the voltage regulator from battery when off and provides very less voltage (<= 0.2 Volts) drop when switched on.	a. Connect the switch across the battery and put a voltage meter right below the switch. b. Turn on the switch and note the voltage. c. Turn the switch off and verify that the value recorded for the voltage is less than 0.2 V.

Voltage regulator

As we are using Li-on batteries, the voltage across will degrade over time and the battery won't provide the voltage level needed for all circuits. So, we would need a voltage regulator to output smooth 3.3V, and this has to be maintained for a long time. This means we need to be efficient enough to provide 3.3V from 3.5-5 Volt battery. This can be achieved by a linear voltage regulator (low-drop off particularly) or a switch regulator. As, LDO regulator can effectively provide 3.3V from 3.5V-5V, we believe choosing switch regulator doesn't reason out; first it is more costly, and second it generates frequency interference in usage which can hamper with our RFID signals which exist in similar bandwidth region. So, we will choose a linear regulator of class LDO.

Requirement(s)	Verification
Requirement 1: Can output stable 3.2-3.3V from an input supply of 3.5-5V.	a. Connect battery with voltage regulator b. Use a multimeter across the voltage regulator to check if the voltage output is equal to 3.2-3.3V.
Requirement 2: Maintains thermal stability below 125°C.	a. Connect battery parallely with a resistor and connect the voltage regulator with GND and VDD across battery. b. Use an IR thermometer to ensure that temperature stays below 125°C.

Circuit schematic

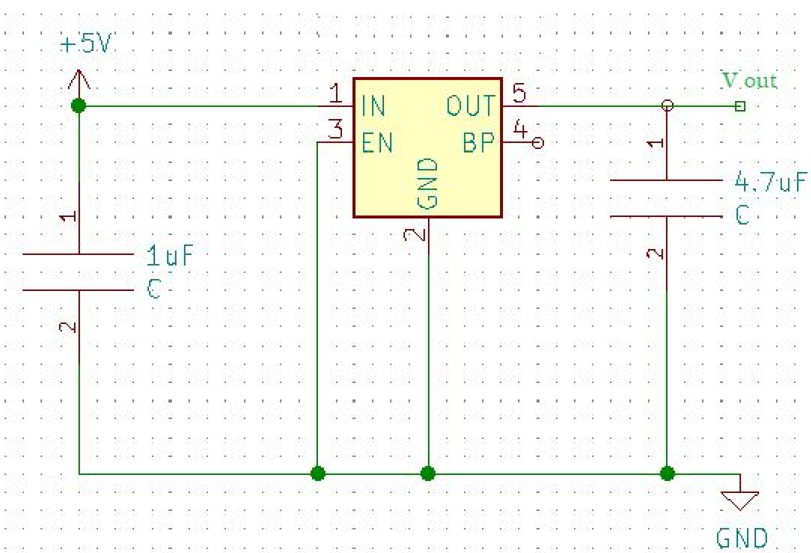

Figure 2.5: Power Supply unit

The Pin 6 comes from Micro-control, so that microcontroller can re setup the system if needed. The remaining circuit will be powered through pin 6, which contains an output of 3.3V.

2.2.4 Control Unit Subsystem

Micro-controller

Input(s): 3.3V power input, Updated position received by bluetooth module, Button press to remove from virtual queue.
Output(s): Command to update LED display, Command to disable buzzer and LED display on button press, Command to trigger vibration of the buzzer if position <= 10.

The microcontroller should be compatible with bluetooth (BLE) module, LED display, vibrating buzzer and button. It must have enough input and output pins to support all components, and must communicate with Bluetooth via UART and LED display via SPI. The RFID Tag ID would be physically stored in the user-programmable memory of the microcontroller, so when the updated positions on queue are being broadcasted using Bluetooth by the receiver, the microcontroller can match the tag ID with the current queue position of the specific user.
Main functionalities:

1. After it receives the broadcast regarding updated positions via bluetooth, it must filter to find the RFID tag's unique identifier (UID) in the message and thus match the updated position of the student on the queue.
2. After finding the updated position of the student on the queue, it must send the value to the LED display via SPI.
3. It must trigger vibration of the buzzer when the current position on the virtual queue is <= 10.
4. If the button is pressed (request to remove from virtual queue), it must disable updates on position (via bluetooth) and the LED display.

filtering to find the RFID tag's unique identifier (UID) in the message to locate the current position of the UID

Requirement(s)	Verification
Requirement 1: The microcontroller must be able to communicate over UART protocol with the bluetooth module.	a. Connect microcontroller to a USB-UART bridge, like CP2102. b. Using a terminal, send data via the UART bus and verify whether this is the same as the data echoed back.
Requirement 2: The microcontroller must be able to communicate over SPI protocol simultaneously with the LED display.	a. Connect microcontroller to a USB-SPI bridge, like MCP 2210. b. Using a terminal, send data via the SPI bus and verify whether this is the same as the data echoed back.

***Requirement 3*:** The microcontroller should be able to communicate over UART and SPI at speeds greater than 4.5 Mbps	a. Connect microcontroller to a USB-SPI or USB-UART bridge. b. Send a 0.45Mb block of data from the USB bridge. c. Echo data back over SPI or UART. d. Ensure that the data received matches the data sent, and time elapsed < 100ms.

High level Schematic connection

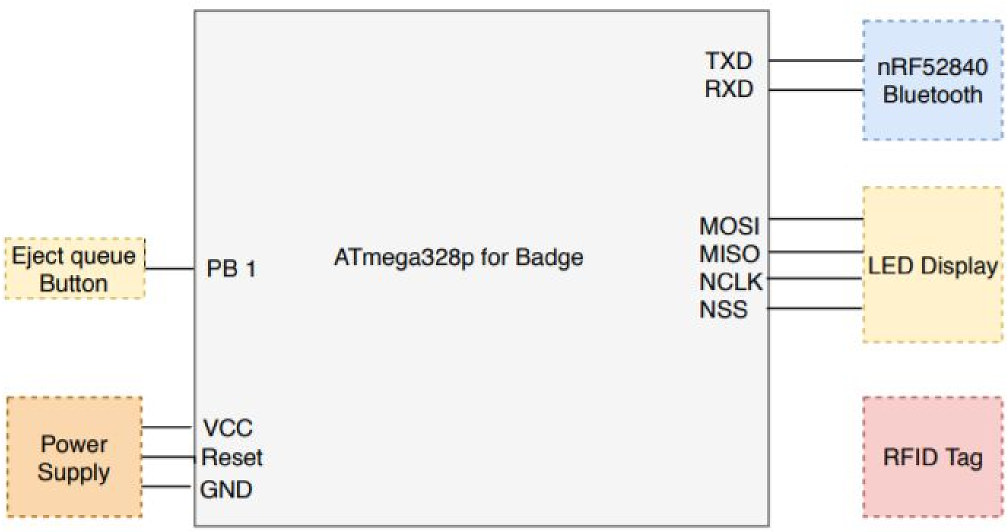

Figure 2.6: ATmega328P connection with other components

Pin layout

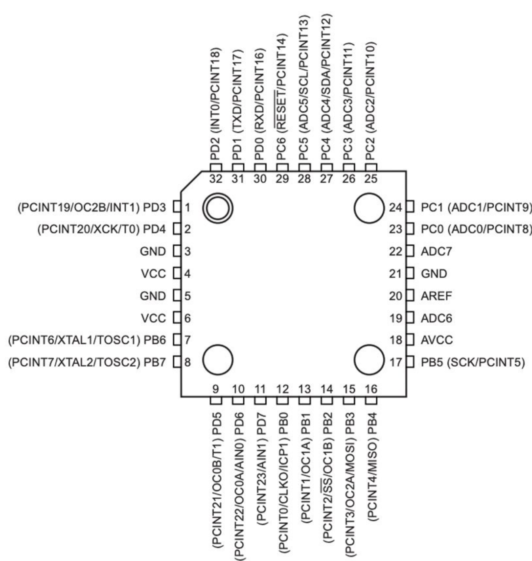

Figure 2.7: Pin layout for ATmega328P [5]

LED Display

The LED display should display the current position of the student on the virtual queue once he/she has been added onto it for only those students that are higher than the tenth position in the queue. It should take the input value from the microcontroller via SPI and display it.

Requirement(s)	Verification
Requirement 1: The LED must take input value via SPI from microcontroller and display it.	a. Connect the LED display to the microcontroller. b. Using the SPI communication procedure, seen above, send data and verify this data is correctly displayed on the LEDs.

Buzzer

The vibrating buzzer soldered on the underlying PCB should be able to alert the student. It should receive a signal from the microcontroller when the student's current position on the virtual queue is less than or equal to 11, and initiate the vibration of the motor disk to alert the student.

Requirement(s)	Verification
Requirement 1: The vibration mini motor disk should be able to provide vibration frequency of >2000 RPM for the it to be recognized by the user.	a. Connect the voltage regulator with the li-ion battery as input to the vibrating mini motor. b. Use a piezoelectric sensor or accelerometer to measure the vibration frequency and ensure it is in the specified range.
Requirement 2: 3.3V should be within the vibrating mini motor's operating voltage.	a. Connect the voltage regulator with the li-ion battery as input to the vibrating mini motor. b. Use a multimeter to check if the voltage input is equal to the specified value and if the motor is vibrating.

Button

The button on the electronic badge should provide functionality for the student to remove himself/herself from the queue. When this button is pressed, the microcontroller should register the press, and disable reception of bluetooth broadcast for position updates and LED display. Our virtual queue system is such that when this student reaches on top of the virtual queue and does not show up, they would automatically be removed from the queue after a 15 minute timeout, and none of the other students would be affected.

Requirement(s)	Verification
Requirement 1: Check to make sure that the button press is registered by the microcontroller.	a. Connect the button to the microcontroller. b. Press the button and verify with the host program that the press was registered by the microcontroller.

2.2.5 Communication Subsystem

Bluetooth module

The bluetooth module must be BLE (Bluetooth low energy eg. RN4871). This means it would have low power consumption, few milliseconds of connection time and high data transmission rate. It must be able to receive information packets broadcasted from a 100m away, as that is the approximate length of the room career fairs are held in. This module must be able to receive the updated position of student (owner of the badge) by receiving the broadcast message and filtering to find the RFID tag's unique identifier (UID) in the message to locate the current position of the UID. Thus, the module must:

1. Receive any broadcast messages sent by the receiver's bluetooth.
2. Send the received message to microcontroller via UART to filter out the updated position of the student.

Requirement(s)	Verification
Requirement 1: Must communicate with the microcontroller using UART to send the received message.	a. Connect the bluetooth module to the UART input port of microcontroller. b. Send an information packet via another bluetooth module. c. Using a terminal, send data via the UART bus and verify whether data is received by microcontroller.
Requirement 2: Must receive the message broadcasted by the receiver's bluetooth module, and thus must be long-range (50-100 m).	a. Send an information packet via bluetooth stationed 100m away. b. Ensure that the packet is received by bluetooth module through microcontroller.

RFID Tag

The RFID tag should be able to activate when close enough to the receiver (and by extension, the RFID reader) and send the relevant information to allow students to be added to the virtual queue. It should have fast data transmission time and should only work within a close range as we would prefer the system to only work when the badge is in close proximity to the receiver.

Additionally, when tapped on the RFID reader again after the student has interacted with the recruiter, the user should be removed from the virtual queue.

Requirement(s)	Verification
Requirement 1: RFID tag's read range should be at most 5 cm from the receiver.	a. Connect RFID reader to a prototyping Arduino board. b. Move the tag increasingly far from the reader to determine the maximum range. c. Reduce power to the RFID reader accordingly.
Requirement 2: Should be able to send the Tag's UID to the RFID reader.	a. Connect RFID reader to a prototyping Arduino board. b. Upload code which will read in serial data transmitted by RFID reader via SPI. c. Place RFID tag within 5 cm of the reader. d. Check Arduino board's serial monitor for the tag's UID.
Requirement 3: The operating frequency should be ~13.56 MHz for appropriate data transmission.	The tag should operate at the same frequency as the reader for the reader to detect the tap.

2.3 Electronic Receiver

2.3.1 Block Diagram

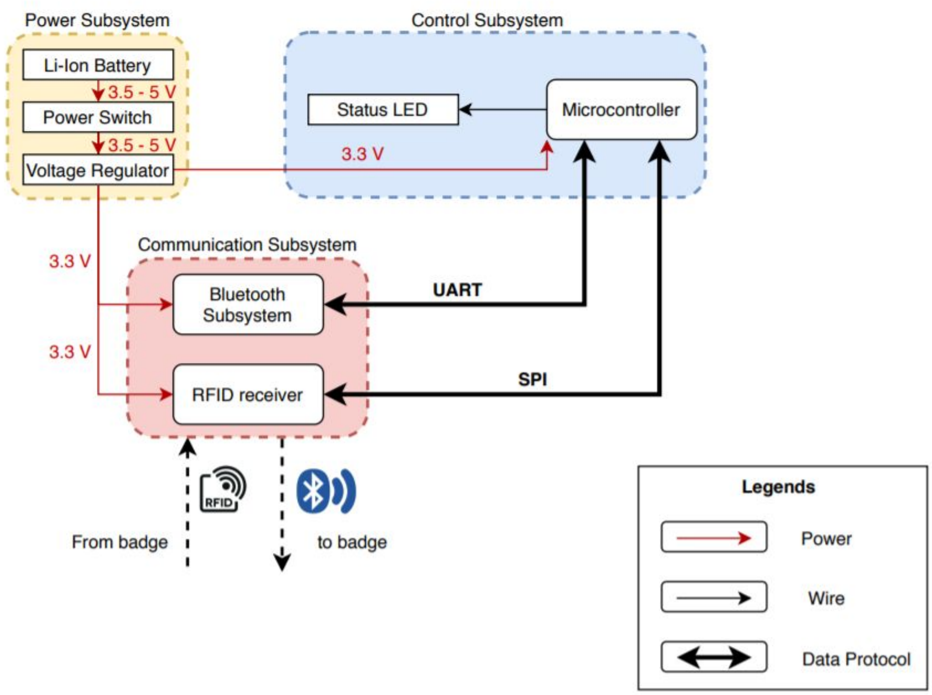

Figure 2.8: Block diagram for Electronic Receiver

2.3.2 Physical Diagram

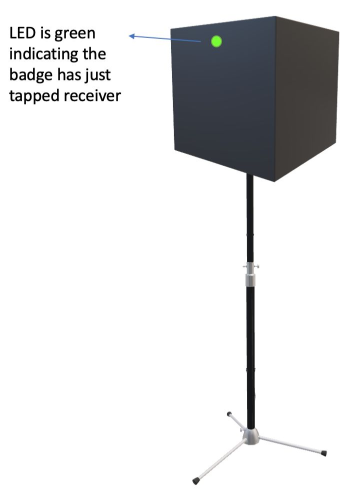

Figure 2.9: Physical diagram for Electronic Receiver

2.3.3 Power Supply

Li-ion battery

The lithium-ion battery must be able to keep the circuit continuously powered over a long period of time.

***Requirement 1*:** Battery should be able to provide a voltage output of 3.5-5 V.	a. Connect the fully-charged li-ion battery with VDD and GND. b. Use a multimeter to check if the voltage output is in the specified range.
Requirement 2: Battery should be able to provide > 1500 mAh for 5-6 hrs.	a. Connect a fully-charged lithium-ion battery with VDD and GND. b. Draw 250 mA current from the battery for 5-6 hrs. c. Monitor output voltage battery and ensure it does not drop below 3.5V earlier.

Power switch

We don't want the device to run indefinitely, and would like to switch it off after use. Thus, we need a switch which disconnects the voltage regulator from the battery when switched off.

Requirement(s)	Verification
Requirement 1: A switch which disconnects the voltage regulator from battery when off and provides very less voltage (<= 0.2 Volts) drop when switched on.	a. Connect the switch across the battery and put a voltage meter right below the switch. b. Turn on the switch and note the voltage. c. Turn the switch off and if the switch works, the circuit will get disconnected and the value should be close to 0 V.

Voltage regulator

As we are using Li-on batteries, the voltage across will degrade over time and the battery won't provide the voltage level needed for all circuits. So, we would need a voltage regulator to output smooth 3.3V, and this has to be maintained for a long time. This means we need to be efficient enough to provide 3.3V from 3.5-5 Volt battery. This can be achieved by a linear voltage regulator (low-drop off particularly) or a switch regulator. As, LDO regulator can effectively provide 3.3V from 3.5V-5V, we believe choosing switch regulator doesn't reason out; first it is more costly, and second it generates frequency interference in usage which can hamper with our RFID signals which exist in similar bandwidth region. So, we will choose a linear regulator of class LDO.

Requirement(s)	Verification
Requirement 1: Can output stable 3.2-3.3V from an input supply of 3.5-5V.	c. Connect battery with voltage regulator d. Use a multimeter across the voltage regulator to check if the voltage output is equal to 3.2-3.3V.
Requirement 2: Maintains thermal stability below 125°C.	c. Connect battery parallely with a resistor and connect the voltage regulator with GND and VDD across battery. d. Use an IR thermometer to ensure that temperature stays below 125°C.

Circuit schematic

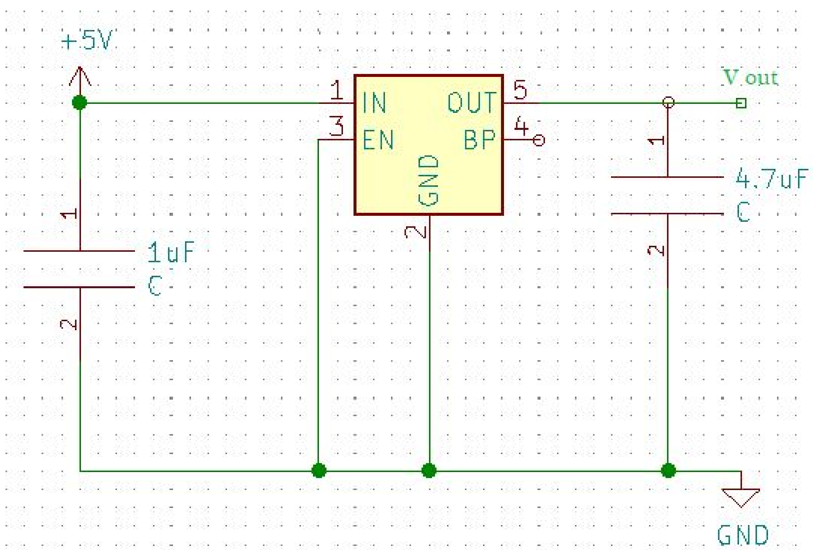

Figure 2.10: Power Supply unit schematic

2.3.4 Control Unit Subsystem

Micro-controller

Input: 3.3V power input, RFID Tag's UID registered when any student taps their badge to add to virtual queue, or after student is done interacting with the recruiter to remove them from virtual queue.
Output: Command to broadcast updated positions by bluetooth via UART, Command for status LED to flash green when any RFID tag's tap is registered.

The microcontroller should be compatible with bluetooth (BLE) module, status LED and RFID reader. It must have enough input and output pins to support all components, and must communicate with Bluetooth via UART and RFID reader via SPI. Whenever it receives information regarding a new tap from RFID reader, it must command the status LED to flash green to let the user know that the tap has been registered. Additionally:

1. If the tag UID is not already on the virtual queue, it must add it to the virtual queue.
2. If the tag UID is already on the virtual queue, this indicates that the student associated with this UID is done talking to the recruiter and now should be removed from the virtual queue.

It also must command the Bluetooth Low Energy (BLE) module to broadcast the current positions of the users on the virtual queue (refer to Bluetooth module description)

Requirement(s)	Verification
Requirement 1: The microcontroller must be able to communicate over UART protocol with the bluetooth module.	a. Connect microcontroller to a USB-UART bridge, like CP2102. b. Using a terminal, send data via the UART bus and verify whether this is the same as the data echoed back
Requirement 2: The microcontroller must be able to communicate over SPI protocol simultaneously with 2 systems (LED display and RFID tag)	a. Connect microcontroller to a USB-SPI bridge, like MCP 2210. b. Using a terminal, send data via the SPI bus and verify whether this is the same as the data echoed back
***Requirement 3*:** The microcontroller should be able to communicate over UART and SPI at speeds greater than 4.5 Mbps	a. Connect microcontroller to a USB-SPI or USB-UART bridge. b. Send a 0.45Mb block of data from the USB bridge. c. Echo data back over SPI or UART. d. Ensure that the data received matches the data sent, and time elapsed < 100ms.

High level Component connection

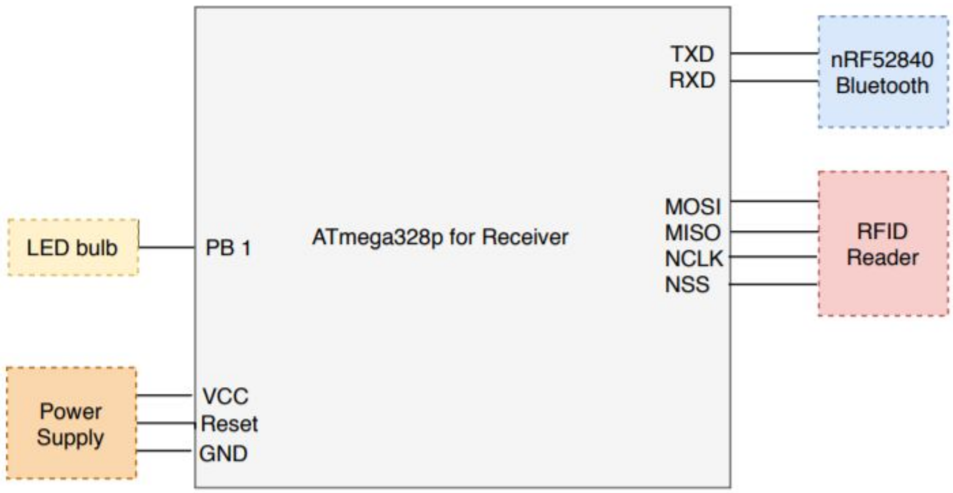

Figure 211: ATmega328P connection with other components

For pin layout, refer to **2.2.4**.

Status LED

The Status LED flashes green after the electronic badge is tapped on the receiver using the RFID reader. It should take the input value from the microcontroller and flash green based on whether the badge was successfully tapped on the receiver.

Requirement(s)	Verification
Requirement 1: The LED must take input value from microcontroller and flash green based on whether the badge was successfully tapped on the receiver.	a. Connect Microcontroller to status LED b. Send a signal from the microcontroller to the status LED and verify that it flashes green.

2.3.5 Communication Subsystem

Bluetooth Module

The BLE (Bluetooth low energy eg. RN4871) module should have low power consumption, few milliseconds of connection time and high data transfer rate. This module must broadcast:

1. the updated positions of the first 50 students on the queue identified by their RFID Tag's UID every 2 minutes.
2. the updated positions of all the students on the queue identified by their RFID Tag's UID every 10 minutes.

This broadcasting schedule is to ensure that students with positions < 50 get updates on their positions faster, but at the same time, the rest of the students also receive some updates while lowering the memory bandwidth consumption.

Calculation: Deciding how often updates need to be broadcasted.
As this device is meant for students who wish to stand in virtual queues for companies that tend to have long lines, i.e big companies such as Amazon, Microsoft, Google etc., we can assume that each company has at least 4 recruiters, and each recruiter interacts with each student for 5 minutes. As our physical queue is a maximum of 10 students,
average time for 10 students interacting with recruiters = 5 x 2.5 = 12.5 minutes.
Thus, average time for 50 students assuming no delay = 12.5 x 5 = 62.5 minutes ≈ 1 hour.
This calculation suggests that students whose turn to talk to the recruiter is going to come in an hour or less should be updated on their positions on the queue every 2 minutes, while the rest of students would be updated every 10 minutes, as currently they would have to wait for > 1 hr.

Calculation: Deciding the maximum number of students virtual queue should handle.
With the current virtual queue size of 256 students, the current position of any student can be represented using 1 byte. It is also known that the RFID Tag UID is ~4 bytes. Thus,
Size of information packet for one student's position = 1+4 = 5 bytes.
Size of information packet for 50 students = 5 x 50 = 250 bytes
Size of information packet for 256 students = 1280 bytes = 1.28 kB.
As the message with updated positions would be broadcasted often, we wanted the size of the message being broadcasted every 2 minutes to be < 1kB, thus for 50 students, it is 250 B. Additionally, the size of the information packet being broadcasted every 10 minutes (1.28 kB) is also reasonable.

Requirement(s)	Verification
Requirement 1: Must communicate with the microcontroller using UART to receive the information packet to broadcast regarding updated positions on virtual queue.	a. Connect the bluetooth module to the UART output port of microcontroller. b. Send an information packet via another bluetooth module. c. Using a terminal, send data via the UART bus and verify whether data is received by microcontroller.
Requirement 2: Must be able to broadcast the message such that	a. Broadcast an information packet via this bluetooth module. b. Ensure that the packet is received by another module stationed 100m away.

bluetooth receivers 50-100m away can receive the message.	

RFID Reader

The RFID reader should be able to activate the RFID tag when the badge is tapped on the receiver. It should have fast detection time and should only activate tag within a close range as we would prefer the system to only work when the badge is in close proximity to the receiver. Additionally, when the badge is tapped on the receiver, the reader must receive the RFID tag's UID and forward the information to the microcontroller via SPI. The microcontroller will then either add this UID to queue or remove if it already exists, and thus maintain the virtual queue. Additionally, when the microcontroller receives this information, it would command the status LED to flash green to inform the student that the tap has been detected.

Requirement(s)	Verification
Requirement 1: RFID receiver activates RFID tag within at most 5 cm range.	a. Connect RFID reader to a prototyping Arduino board. b. Move the tag increasingly far from the reader to determine the maximum range. c. Reduce power to the reader accordingly.
Requirement 2: Should be able to read the RFID tag's UID and must send the information to the microcontroller correctly.	a. Connect RFID reader to a prototyping Arduino board. b. Upload code which will read in serial data transmitted by RFID reader. c. Place RFID tag within 5 cm of the reader. d. Check Arduino board's serial monitor for serial number.
Requirement 3: The operating frequency should be ~13.56 MHz.	The reader should operate at the same frequency as the tag for it to detect the tap.

Circuit schematic

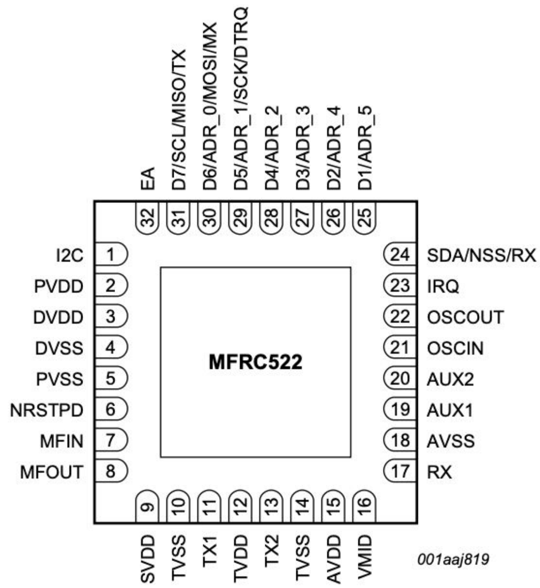

Figure 2.12: RFID Reader Circuit Schematic [6]

2.4 Tolerance Analysis

One important subsystem which should operate during the entire duration of career fair is the power-supply module. Every student would add themselves to virtual queues and wait accordingly; however, if the electronic badge system turns off during the career fair not only student lose will his/her spot in the career fair, he/she won't be able to add themselves in any further lines. This would result in the entire career fair day being wasted and is something which has to be absolutely avoided. Hence, we will model the power supply subsystem for tolerance analysis. To begin, we would need to calculate nominal mAH requirement for the badge to operate.

Our system works on 3.3V which is provided through a linear voltage regulator. As all components operate at 3.3V, we can use it as a benchmark to find power consumption per hour of each component. We have 4 main components on the badge which consume power:

a. **LED Display:** 10mAh for a single LED digit counter, and we will need 3 to represent 250 entries. Resulting in 30mAh
b. **Micro-controller**: 0.5 mAh for ATmega328p at 1 Mhz at 1.8V. However, we operate at moderate throughput of 5-6 Mhz at 3.3V. We can use the graph provided by TI to calculate the mA. In the graph, at 5-6 Mhz at 3.3 V, we consume 2.4 mA, resulting in 2.4 mAh consumption.

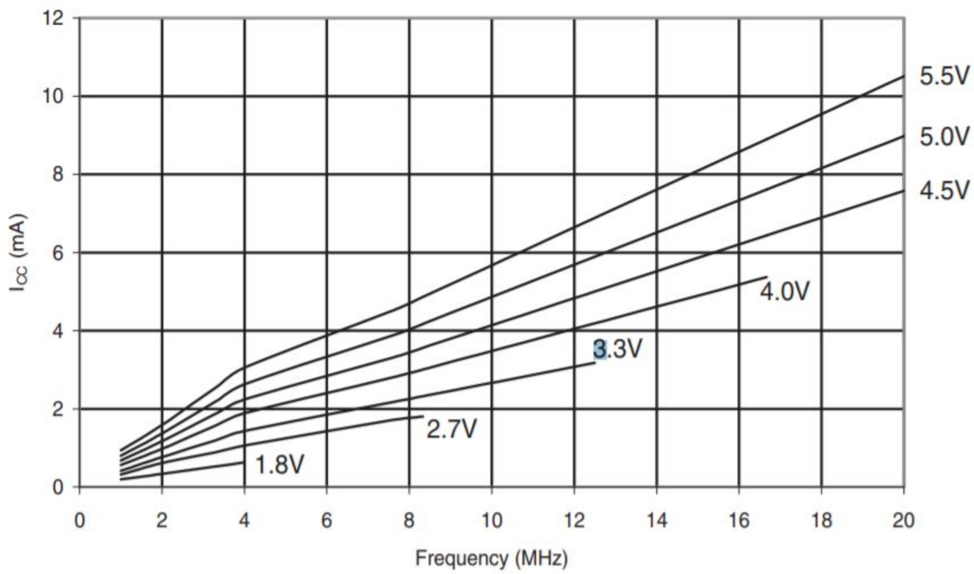

Figure 2.13: ATmega328: Active current vs y Current vs. Frequency (1-20MHz) [7]

c. **Bluetooth module:** Using the nRF52840 datasheet, power consumption graph we can see our typical power consumption will be 1.6 uA, resulting in 1.6uAH. This means, at rest state the bluetooth module will consume negligible power.

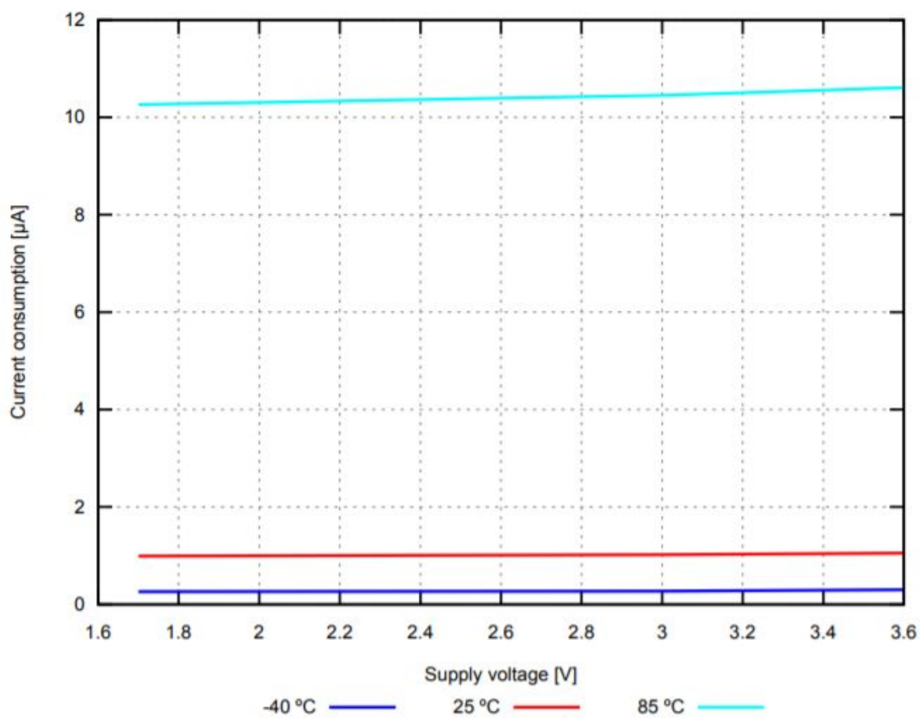

Figure 2.14: nRF52840 System ON, no RAM retention (typical values) [8]

d. **Buzzer:** 10 mAh for medium vibrations, which can be varied.

Nominal usage will result in 25mAh consumption every hour, which means for a typical career fair which lasts 6 hours, the badge consumption would result in 150 mAH consumption. Now we use a Li-on battery which provides approximately 2000 mAh and will be able to sustain the system at nominal usage for 80 hours. However, this is the best case scenario and in reality we would need to consider errors and extremes for devices. This would result in 2 changes; the battery can contain less charge than stated, it can provide reducing amount of voltages throughout the process, and second the components can run at peak capacity. To calculate the amount of charge we will actually receive from the battery, we would need to plot the graphs for a typical li-on battery and see after what charge dissipation level, does the voltage drop below 3.5V. As below this, our voltage regulator won't be able to maintain stable 3.3V. We were able to find a graph of 3500 mAh battery, rated at 3.7 V, and we used this graph to generate data points.

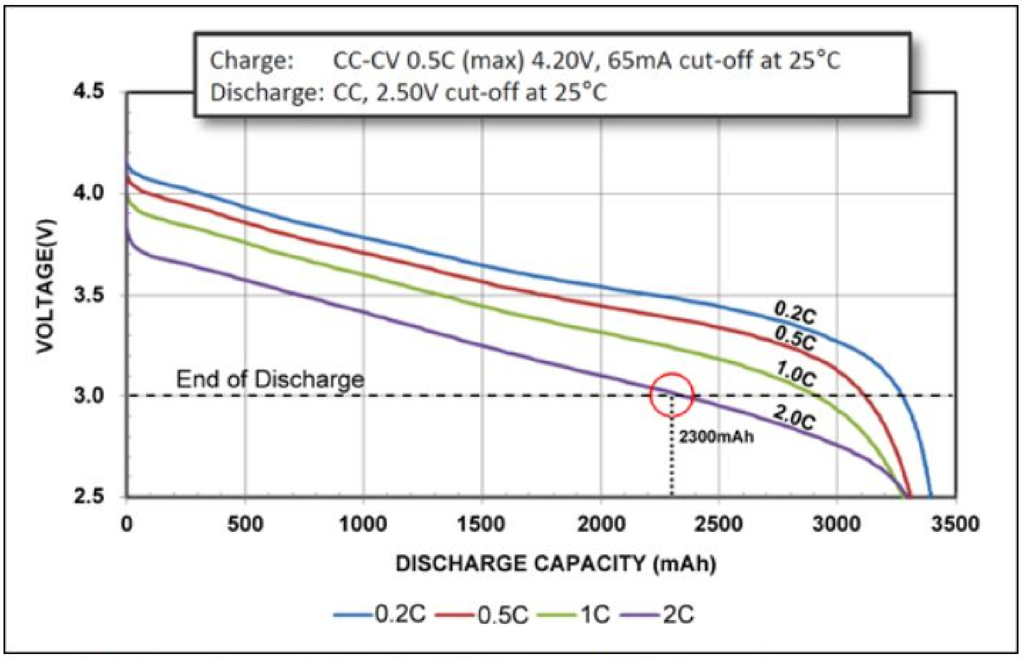

Figure 2.15: Li-on battery discharge over different charge values, Voltage vs Energy discharged [9]

Through interpreting this graph we derived these data points.

Index	Charge dissipated (mAh)	Voltage V	Index	Charge dissipated (mAh)	Voltage V
1	0	4.23	**19**	1800	3.626
2	100	4.15	**20**	1900	3.608
3	200	4.1	**21**	2000	3.59
4	300	4.05	**22**	2100	3.562

5	400	4	**23**	2200	3.534
6	500	3.95	**24**	2300	3.506
7	600	3.924	**25**	2400	3.478
9	700	3.898	**26**	2500	3.45
8	800	3.872	**27**	2600	3.41
10	900	3.846	**28**	2700	3.37
11	1000	3.82	**29**	2800	3.33
12	1100	3.792	**30**	2900	3.29
13	1200	3.764	**31**	3000	3.25
14	1300	3.736	**32**	3100	3.1765
15	1400	3.708	**33**	3200	3.103
16	1500	3.68	**34**	3300	3.0295
17	1600	3.662	**35**	3350	2.9
18	1700	3.644	**36**	3375	2.75
			37	3400	2.5

Table 1: Data points derived through charge dissipation from 3500 mAh battery discharge

And from this we can plot a graph on excel and devise a formula via fitting a cubic trendline on the data points.

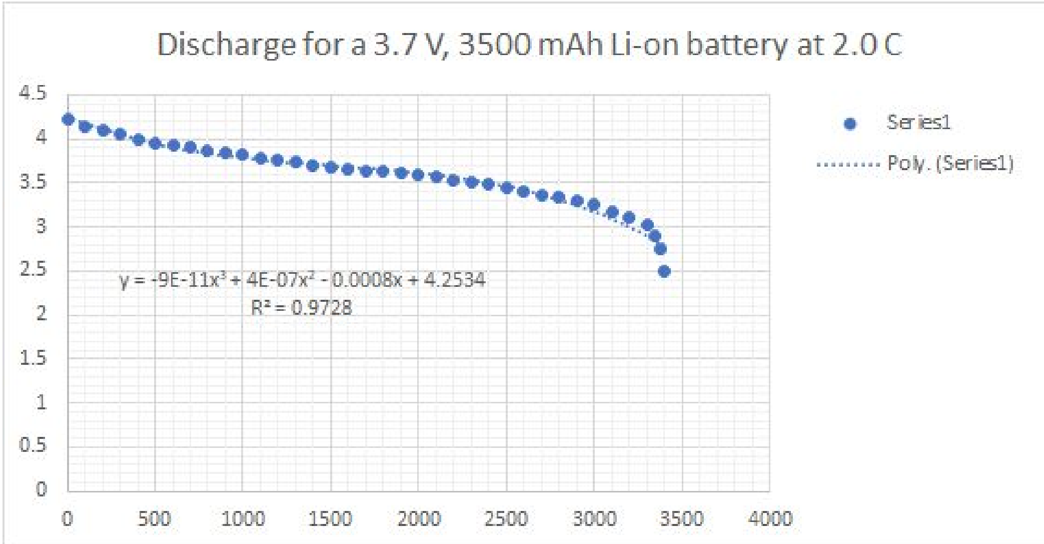

Figure 2.16: Fitting trendline through data points and deriving an equation for battery discharge

Assume this equation is f(x) and as our battery is around 2000 mAh, we would need to squeeze x-axis by 4/7, to create an equation for a battery with 2000 mAh. This results in an equation for our battery:

$$VoltageAcrossBattery(x) = -9 \cdot 10^{-11}(\tfrac{7}{4}x)^3 + 4 \cdot 10^{-7}(\tfrac{7}{4}x)^2 + 8 \cdot 10^{-4}(\tfrac{7}{4}x) + 4.2534$$

Now, we can only use battery until the voltage across the battery is greater than 3.5V. So we would need to find the discharge point after which our battery can't be used.

VoltageAcrossBattery(x) = 3.5, and we solve for x and we get **x = ~1205 mAh.**

Now battery in stricter analysis will only provide voltage levels above 3.5V till 1205 mAh. Working on the extreme, if we assume all our components will be working at max capacity then we will have following consumptions.

a. **LED Display:** 10mAh for a single LED digit counter, and we will need 3 to represent 250 entries. Resulting in 30mAh. It is still the maximum power used as LED will be continuously switched on.
b. **Micro-controller**: 0.5 mAh for ATmega328p at 1 Mhz at 1.8V. However, we operate at maximum throughput of 12.5 Mhz. We can use the graph provided by TI to calculate the mA. In the graph, at 12.5 Mhz at 3.3 V, we consume 3.5 mA, resulting in 3.5 mAh consumption.
c. **Bluetooth module:** Using the nRF52840 datasheet, power consumption for bluetooth module at peak (when continuously transmitting) will be 15 mA, resulting in 15 mAH.

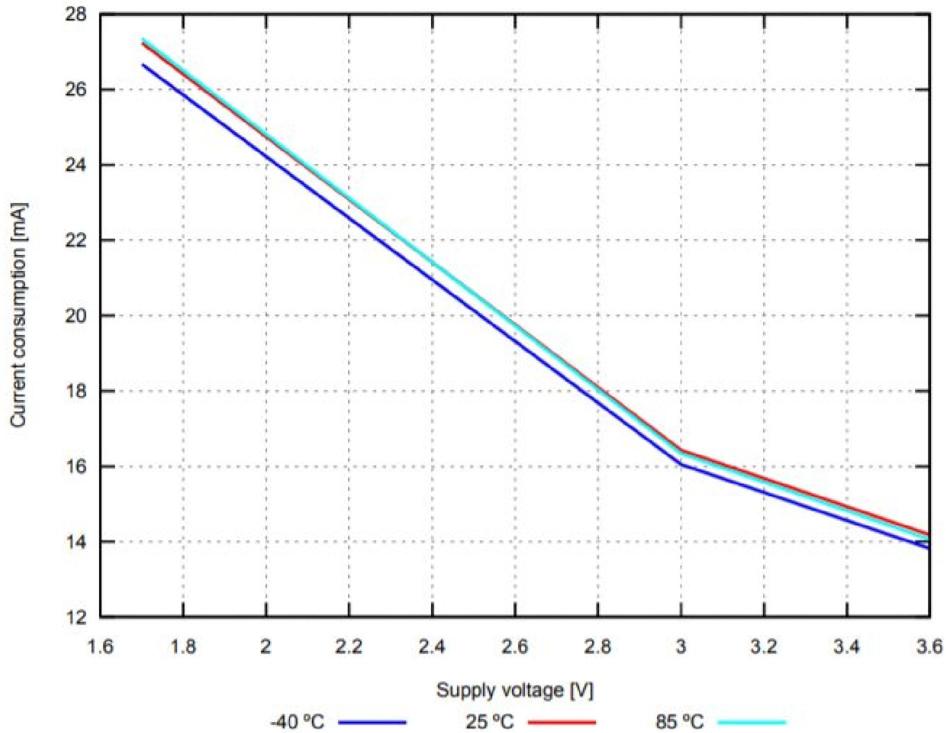

Figure 2.17: Active radio transmission for 8 dBm output power 1 Mbps BLE mode [8]

d. **Buzzer:** 10 mAh for medium vibrations, which can be varied.

This results per hour consumption of 58.5 mAh for the badge and the battery will provide correct voltage till ~1200 mAh. So, our badge will be able to function for 20.5 hrs (more than 6 hr career fair) even in extreme cases. To conclude the tolerance analysis, the power system will be able to hold up the badge and ensure the student has smooth experience during the entire time in career fair.

2.5 Software

The software aspect of our project revolves around the utilization of the queue data structure and so there is some functionality that we need to implement programmatically to the microcontroller that will be further elaborated upon.

First and foremost, the virtual queue will be stored as a list in memory. Each entry in the virtual queue will have a data structure containing UID, TAP-Count, Timer and BuzzSent.

Struct VirtualQueue {
    VirtualQueueEntry[256]
}

```
Struct VirtualQueueEntries {
	UID:		int
	TapCount:	int
	Timer:		int
	Buzzed:		bool
}
```

In our workflow, after the student taps the electronic badge on the company's receiver, we want our microcontroller in the receiver to handle the logic of adding somebody onto the virtual queue, and so this would involve the enqueue functionality of queue.

There are also times when a student is taken off a queue either through the student's own choice or through a time limit of 15 minutes, which would require the dequeue functionality of queue.

```
enqueue(UID) {
	#The person spoke to the company and has tapped to be removed
	If ( UID in VirtualQueue)  {
		dequeue(UID)
	}

	Else {
		NewEntry VirtualQueueEntry(UID, 0, 0, False)
		VirtualQueue.add(NewEntry)
	}
}

Dequeue(UID) {
	Entry <- getEntry in VirtualQueue.search(UID)
	VirtualQueue.remove(Entry)
}
```

We will also need functionality for checking whether or not a queue is full as this will dictate a student's ability to add himself/herself to the queue. Similarly, we will need functionality for when the queue is empty. There are some other important functionalities below.

```
# There will be a function which starts a timer whenever bluetooth sends an updated position for an entry, and the position is less than 10.
Buzzed(UID) {
	Entry <- getEntry in VirtualQueue.search(UID)
	Entry.Buzzed <- True
	Entry.Timer <- Start
}
```

There will also be an automatic delete function, which will autorun every 5 minutes to delete entry from VirtualQueue whose timer in VirtualQueue has exceeded 15 minutes

```
DeleteTimer(UID) {
    For each Entry in VirtualQueue:
        # The entry hasn't responded within 15 minutes and needs to be deleted
        If (Entry.Buzzed == True and Entry.Timer > 900 seconds) {
            VirtualQueue.remove(Entry)
        }
}
```

3. Cost Analysis

3.1 Labor

The labor costs can be calculated by the formula seen below, using an hourly rate of 50$, 12 hours per week for 10 weeks:

$$Labor\ Costs\ =\ 2.5\ \times\ (group\ size)\ \times\ (hourly\ cost)\ \times (hours\ per\ weeks) \times\ \ (\#\ of\ weeks)$$

Inputting the values:

$$Labor\ Costs\ =\ 2.5\ \times\ 3\ \times\ 50\ \times\ 9\ \times\ 10$$

Therefore we have:

$$Labor\ Costs\ =\ \$45,000$$

3.2 Parts

Table 2: Cost of Parts

Name	Manufacturer	Product ID	Quantity	Price ($)	Total Price ($)
LDO Voltage Regulator	Mouser	595-TPS77033DBVRG4	2	1.13	2.26
3.7V Rechargeable Battery	Adafruit Industries LLC	1528-1857-ND	2	12.5	25
Battery Charger	Adafruit Industries LLC	1528-1833-ND	1	6.95	6.95
Power Switch	TE Connectivity	1825255-1	1	1.53	1.53

LED Display	Jameco	225963	1	1.75	1.75
Button	TE Connectivity	1825967-1	2	0.23	0.46
Buzzer	Adafruit Industries LLC	1201	1	1.95	1.95
Bluetooth Module	Nordic Semiconductor	nRF52840	2	12.95	25.9
Microcontroller	Microchip Technology	ATMEGA328P-AN	2	2.11	4.22
RFID Module	Mifare	RC522	1	6.99	6.99
Retractable Badge Holder	N/A	N/A	1	3.29	3.29
PCB Case	Hawk Electronics Inc	1455A802BK	1	6.9	6.9
Arduino Uno R3	Arduino	A000066	1	18	18
USB to TTL Serial Connector	Adafruit Industries LLC	1528-2128-ND	1	9.95	9.95

3.3 Total Costs

Table 3: Total Costs

Section	**Cost ($)**
Labor	45,000
Parts	105.2
Total	45,105.2

4. Schedule

Table 4: Schedule for Project

Week	**Pragya Aneja**	**Ninad Godbole**	**Varad Khandelwal**
03/02 - Design Review	Design Review and Incorporating Feedback and Critique	Design Review and Incorporating Feedback and Critique	Design Review and Incorporating Feedback and

			Critique
03/09 - Team Evaluation, Soldering Assignment	Ordering Parts	Ordering Parts	Ordering Parts, Begin Drafting PCB
03/23 - First Round PCB	Testing and Programming Bluetooth module	Testing and Programming RFID tag and Reader using Arduino	Design PCB for first round of PCBWay order
03/30 - Individual Progress Report	Implement Bluetooth module and RFID functionality of the Communication Subsystem	Construct Power Subsystem Circuit for Receiver and Badge	Construct Power Subsystem Circuit for Receiver and Badge
04/06 - Final PCB	Construct Control Subsystem Circuit for Badge with regards to LED display, buttons and buzzer	Work on Construction of Control Subsystems for badge and receiver	Construct Microcontroller Circuit for Receiver and begin programming logic for virtual queue Receiver, Finalize BCB
04/13	Finish leftover work, Integrate all components for the Badge and Receiver respectively and Perform Final Testing	Finish leftover work, Integrate all components for the Badge and Receiver respectively and Perform Final Testing	Finish leftover work, Integrate all components for the Badge and Receiver respectively and Perform Final Testing
04/20	Mock Demo and Final Corrections	Mock Demo and Final Corrections	Mock Demo and Final Corrections
04/27	Final Demo	Final Demo	Final Demo
05/04	Final Presentation, Final Paper	Final Presentation, Final Paper	Final Presentation, Final Paper

5. Safety and Ethics

When considering the ethics of our project we made sure to consult the IEEE code of ethics which consist of 10 different guidelines. Based on these guidelines, our project doesn't have any significant issues to breach the outlined ethical code, but there are some things that are worth talking about.

Some of these ethical issues relate to the eighth listed guideline of the IEEE code of ethics which says “to treat fairly all persons..” [10]. In certain cases, some of the parts which are used in our project, like the LED display or bulb could stop working due to overuse or other unknown reasons. This could result in some students not being able to use an electronic badge of their own to take advantage of the virtual queue system. We will therefore have to make sure we are aware of this as we build our badge and focus on using good quality parts. Another ethical issue which could unfairly impact some people is if they are talking to a certain recruiter and their position comes up for another virtual queue causing them to miss their place in the other queue. To counter this issue, we decided to have the badge alert them when they are tenth in the queue so that they have enough lag time to make their position. Another ethical issue could arise if the cost of the badge is placed on the students which could result in some students missing out on the product. Ideally this issue would not be present given the badges are handed free of cost from ECS itself.

The first and ninth listed guidelines of the IEEE code of ethics are also very important which say “to hold paramount the safety, health, and welfare of the public...” and “to avoid injuring others...” respectively [10]. These guidelines connect well with safety problems that could arise for our project which will be further elaborated upon. The two biggest safety issues for our project involve the Li-ion battery and the badge design. When it comes to the Li-ion battery, these can overheat with a lot of usage and when this goes to extreme levels, it has the potential to catch fire [11]. Although this is very rare, it is an important safety hazard and definitely something we have to be mindful with regards to the usage for both the electronic badge and receiver. Another safety issue is if a student drops the badge by accident, which could cause damage to the badge for future use and result in the parts being scattered across the floor. It is therefore very important that the badge is made to be as sturdy and durable as possible to avoid such a safety hazard.

6. References

[1] L. McQuerrey, “Problems With Job Fairs,” Chron.com, 21-Nov-2017. [Online]. Available: https://work.chron.com/problems-job-fairs-12144.html. [Accessed: 14-Feb-2020].

[2] A. Zubair, “Why Choose A Virtual Career Fair? (Benefits & Trends),” vFairs.com, 14-Jan-2020. [Online]. Available: https://www.vfairs.com/why-choose-a-virtual-career-fair-benefits-trends/. [Accessed: 14-Feb-2020].

[3] “Job Fair Tip #2 - Body language,” DiversityJobs.com, 13-Feb-2020. [Online]. Available: https://www.diversityjobs.com/2010/02/job-fair-tip-2/. [Accessed: 14-Feb-2020].

[4] B. MacDonald, “Disneyland to use virtual queue for new Rise of the Resistance ride,” Orange County Register, 24-Dec-2019. [Online]. Available: https://www.ocregister.com/2019/12/23/disneyland-to-use-virtual-queue-for-new-rise-of-the-resistance-ride/. [Accessed: 14-Feb-2020].

[5] “ATmega328P,” Microchip. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf?fbclid=IwAR1NmfLAtSnuxQ2ExBtGuWSFB522n5f5T6iS3rcBEIZ4TgZybzSZvs6PSNU. [Accessed: 27-Feb-2020].

[6] “MFRC522,” Vetco. [Online]. Available: https://vetco.net/datasheets/VUPN6326/MFRC522 datasheet.pdf?fbclid=IwAR2N5zrk0PRtgRUaM3Y90caslPW3B1xZMMLVTPUrEkCGeQusscqX0IcV8lU. [Accessed: 27-Feb-2020].

[7] “ATmega48A,” Microchip. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf?fbclid=IwAR2O3EeIhwFmI0DSgTVY4LopkJ9AoXhwZDEMjLCQhri9kQIdc17hQAgLms8. [Accessed: 27-Feb-2020].

[8] “BU-501a: Discharge Characteristics of Li-ion,” Calculating the Battery Runtime - Battery University. [Online]. Available: https://batteryuniversity.com/learn/article/discharge_characteristics_li?fbclid=IwAR1-jL3NHPB6w9HD3VaFrzmBa7Q_q6gX5HOq0UKEdImE_m9IX4rHWxZuBm8. [Accessed: 28-Feb-2020].

[9] “nRF52840,” Nordic Semiconductor. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.0.pdf?fbclid=IwAR0P9WGry1kf6JPsUg5QvKUAwXP1WdnYgQhjdyi8o701rYDRLoaYNmBpPiE. [Accessed: 27-Feb-2020].

[10] “IEEE Code of Ethics,” IEEE. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 14-Feb-2020].

[11] “Safety Concerns with Li-ion,” Safety Concerns with Li-ion Batteries – Battery University. [Online]. Available: https://batteryuniversity.com/learn/article/safety_concerns_with_li_ion. [Accessed: 14-Feb-2020].