CapSafe Necklace

By
Richa Meherwal
Kevin Horton
and
Joshua Graham

Final Report for ECE 445, Senior Design, Fall 2016

TA: Samuel Sagan
Contents

1. Introduction ........................................................................................................................................... 1
   1.1 Project Motivation ................................................................................................................................. 1
   1.2 Project Overview ................................................................................................................................. 1

2. Design Overview ..................................................................................................................................... 2
   2.1 Power Supply ...................................................................................................................................... 2
   2.2 Activation Module ............................................................................................................................. 3
   2.3 Capture Module .................................................................................................................................. 5
   2.4 Control Module .................................................................................................................................. 5
   2.5 Wireless Connectivity Module ......................................................................................................... 5
   2.6 Software Interface ............................................................................................................................. 6

3. Design Verification .................................................................................................................................. 7
   3.1 Voltage Regulator ............................................................................................................................. 7
   3.2 Activation Signal ............................................................................................................................... 7
   3.3 Low-Pass Filter .................................................................................................................................. 7
   3.4 Camera ............................................................................................................................................... 8
   3.5 Microcontroller and Bluetooth ........................................................................................................... 8
   3.6 Android Application ........................................................................................................................... 9

4. Costs ......................................................................................................................................................... 10
   4.1 Parts .................................................................................................................................................. 10
   4.2 Labor .................................................................................................................................................. 11

5. Conclusion ................................................................................................................................................. 12
   5.1 Accomplishments ............................................................................................................................... 12
   5.2 Limitations ....................................................................................................................................... 12
   5.3 Ethical Considerations ....................................................................................................................... 12
   5.4 Future Work ...................................................................................................................................... 12

References ..................................................................................................................................................... 13

Appendix A - R & V Table ........................................................................................................................... 14
Appendix B - Complete Schematic Diagram and Block Diagram ................................................................ 18
1. Introduction

1.1 Project Motivation

As one might suppose from the genesis of the It's On Us movement, universities' increased effort in training students in consent, and growing public awareness in general, sexual assault is a very real problem in the US. Such attacks are always highly damaging and stressful to the victims, and responding to them while also remaining safe is difficult at best--maybe the victim’s phone is in a pocket or purse, or maybe the assailant is threatening physical violence. But as awful as being the victim of an assault is, there are far too many stories that get shared around the internet about victims denied justice--and the impact that denial can have on their personal wellbeing. Of course, there may be myriad reasons a victim doesn’t get retribution, but among them is certainly insufficient evidence. CapSafe is a project targeted at helping protect users and, in the case that emergency assistance arrives too late, helping to close that evidence gap.

1.2 Project Overview

CapSafe is a personal safety wearable that serves its users in several ways. The first is that it takes photographs of the scene, hopefully capturing the attacker in at least of those photos. The second is that it sends an emergency text to contacts defined within the app, containing the user’s GPS location at the time of activation. The third is that it directly calls 911, potentially enabling the user to get emergency assistance. The device itself is very small, fitting on the back of a necklace, and operable in standby mode for eight hours. CapSafe also has a deactivation process accessible in the app in case of false positives.
2. Design Overview

2.1 Power Supply

2.1.1 Battery

Our first design choice was determining which type of battery to implement in our circuit. We wanted to find a rechargeable battery that was appropriately small, but still capable of powering our circuit for a time we deemed appropriate. We narrowed our battery choice to three options, whose area and current capacities are detailed in Table 2.1. Ultimately, we chose the 2 coin cells since the board area is drastically less than the other options. We knew our circuit would draw a small amount of current, so we deemed capacity an appropriate sacrifice for size.

2.1.2 Linear Regulators

Our design uses a 5-volt linear regulator in addition to the 3.3-volt regulator on the Bluetooth module. The full circuit remains functional while the microcontroller and camera are receiving at least 4.3 V; thus, we had to determine input voltage to the 5-volt regulator corresponding to an output of 4.3 V. Details of this problem are discussed in Section 2.1.3.

2.1.3 Power Protection

Because we are using lithium-ion batteries, it is important to prevent the battery from overdischarging, which could damage the batteries or the circuitry. We used a voltage divider circuit and comparator to generate and send a low-battery alert to the phone, notifying the user when it is time to turn off the circuit and recharge the batteries. After some tests, we found that the circuit remains functional until the camera is receiving less than 4.3 V. We found that this corresponds to a battery voltage of 5.1 V. Using this information, we created a voltage divider circuit as shown in Figure 2.2, choosing resistor values such that the comparator will output an active-low low-battery signal to the microcontroller when the battery approaches 5.1 V. Our 3.3 V source is from the 3.3-volt regulator on the Bluetooth module.

<table>
<thead>
<tr>
<th>Battery</th>
<th>Volume (mm²)</th>
<th>Capacity (mAh)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9V</td>
<td>21100</td>
<td>175</td>
</tr>
<tr>
<td>2x 3.7V Coin Cell</td>
<td>2198</td>
<td>80</td>
</tr>
<tr>
<td>2x 3.7V 18500</td>
<td>24925</td>
<td>1400</td>
</tr>
</tbody>
</table>

Table 2.1 – Potential batteries and their statistics
2.2 Activation Module

2.2.1 Force Sensing Circuit

Like the low-battery signal, we implemented a voltage divider circuit to generate the activation signal that indicates when to activate the Bluetooth module and camera. The first resistor in the network is a force-sensitive resistor (FSR). An FSR is a resistor whose resistance is ideally infinite when unpressed, and whose resistance decreases as increasing force is applied. We determined that a firm press on the FSR equates to about 4 N of force, or roughly 400 g, in Fig. 2.3. Choosing a smaller force for activation could result in false positives if the user were to accidentally drop the necklace. We found that the corresponding resistance of the FSR to 4 N is 2.4 kΩ. Using this information, we constructed the voltage divider network shown in the figure above. We had to choose the value of the second resistor such that the positive terminal of the comparator is equal to

![Figure 2.1 – A graph showing input(x-axis) vs output voltage(y-axis) for our 5-volt regulator](image1)

![Figure 2.2 – A schematic view of the low-battery detection circuit](image2)

![Figure 2.3 – The FSR manufacturer’s graph of resistance against force applied](image3)
3.3 V with the FSR resistance equal to 2.4 kΩ. Using the voltage divider equation, we found an appropriate resistor value of 4.7 kΩ. With this implemented, the comparator will output a high signal when the FSR is less than or equal to 2.4 kΩ. In other words, any press of 4 N or greater will activate the circuit.

2.2.2 Filter

To make our force-sensing circuit more accurate, we implemented a low-pass filter designed to attenuate noise from sources such as electromagnetic interference, thermal noise, and circuit defects. After some research, we found that most unwanted noise ranges from several kHz to a few MHz; thus, we decided the filter should have a cutoff frequency of 3 kHz [1]. We were then left with a choice among types of low-pass filters to use, as well as its order. We quickly narrowed the type to the one of the two most commonly used types of filters: Butterworth and Tschebyscheff. Butterworth filters have a flatter passband, while Tschebyscheff filters have a steeper rolloff [2]. We chose the Tschebyscheff filter because the impact of noise on the circuit was more important than the flatness of the passband response. Then came the decision of order; with higher order filters comes a steeper rolloff at the cost of a greater number of components. We want to balance low
component count with a steep rolloff, so we designed a 2nd-order Tschebyscheff filter, shown in Figure 2.5.

2.3 Capture Module

The capture module uses the VC0706, a 5-volt TTL serial camera that can operate at several different rates. It also has a power save mode—an important function given the low-power nature of this project—image size selection, and automatic JPEG compression. For optimal image transfer, we implemented the smallest image size (160 x120).

2.4 Control Module

The microcontroller used in this project is a standard low-power, low-cost ATmega328P AU TQFP. It has 32 kB of flash memory, occupies about 0.49 mm² on the PCB, and supports serial communication, the last of which was necessary for communicating with the chosen camera module. The microcontroller accepts input from the force-sensing circuit, controls the camera module, and makes the Bluetooth advertise itself for connection to the user’s phone. Following is pseudocode for the overall functioning of the microcontroller [y].

```plaintext
Function Loop
    sensorval = readsignal()
    If sensorval == HIGH
        read picture to VC0706
        writepicture()
        write picture to VC0706
        Buffer = get_bytes_to_read()
        read picture from VC0706 to microcontroller buffer
        Write Buffer to Bluetooth Buffer
        Send the bluetooth buffer
        Wait 10ms to ensure that the data is transferred completely
    End
    Write “Activate” to Bluetooth Buffer
End
```

2.5 Wireless Connectivity Module

CapSafe uses a low energy Bluetooth module (nRF8001) with an inbuilt, integrated BLE stack. When “Connected”, the Bluetooth device uses about 12 mA, and when “idle,” it uses about 100 nA. With an on-chip non-volatile memory for service configurations, it is easy to select and combine all necessary services required for our application, reducing the requirements on the application controller for handling all real-time operations related to the Bluetooth low energy communication protocol. The average current consumption of the link is significantly reduced compared to a regular continuous connection, and the
application latency of data both for the central and peripheral is significantly lower than for a connection using slave latency only [4].

2.6 Software Interface

The app integrates the functionality of capturing the Bluetooth signal and initiating activities only after detecting a valid packet. The functionality that we implemented was calling and messaging to numbers hard-coded in the software, and the activation and deactivation mechanism involves a heartbeating approach inspired by failure detectors in distributed systems.

We installed the user permissions in the Manifest file and tested different protocols for calling and messaging (specifically for US numbers). Activating the call and messaging immediately following would cause the call to end immediately. To get rid of this type of inconsistency we had to provide synchronization to make the call and messaging function atomic. We introduced a delay of about 10 secs and thread locks to prevent these two functions from executing simultaneously. A PhoneStateListener was later introduced to keep a track of whether the call is in progress or off-hook. If in progress, the background Bluetooth activity stopped; otherwise the looping continued.
3. Design Verification

3.1 Voltage Regulator

Our voltage regulator has an output voltage of at least 4.3 V when the input voltage draws near 5 V and outputs a steady 5 V when the input is at our fully-charged voltage of 7.4 V. Figure 2.1 depicts the measured input vs. output voltage of our regulator. As can be seen, we still achieve 4.3 V output when the input is at 5.1 V and the output settles at 5V well before our input voltage of 7.4 V is reached.

3.2 Activation Signal

Our circuit requires that any force of 4 N or more will generate an activation signal that goes high when the output voltage of the voltage divider network is greater than 3.3 V. Figure 3.1 shows the output voltage vs resistance graph of our FSR network. As mentioned before, a 4 N press yields a resistance of 2.4 kΩ, marked on the plot by a green dot. Any value greater than 4 N will lower the resistance, and the output voltage will further increase above 3.3 V, sending the activation signal.

3.3 Low Pass Filter

Our requirements included implementing a low-pass filter with a cutoff frequency of 3 kHz. Figure 3.2 shows the frequency response of the 2nd-order Tschebyscheff filter we implemented. As can be seen, the -3dB cutoff correlates almost exactly with a 3 kHZ frequency.

![Output voltage vs FSR resistance](image)

Figure 3.1 – A plot of the input to the comparator vs. the resistance of the FSR
3.4 Camera

A valid JPEG image starts with FFD8 and ends with FFD9. All of the image data transferred to software app via Bluetooth. Note that the JPEG image that we got had FFD8 FFFE as our header like seen in the figure 3.5. It would be easy to convert this hex data to an image if the header was xFFD8 xFFE0, where xFFE0 in the header indicates that the file is JFIF style. Unfortunately, we could not find resources that could help us understand how to convert the image of header of type xFFD8 xFFE0.

3.5 Microcontroller and Bluetooth

The microcontroller sends the Jpeg data to the software interface. After the image transfer is complete, the microcontroller sends a sequence of ASCII character that starts a call at the

---

Figure 3.2 – A plot of the filter’s frequency response

Figure 3.3 – Raw JPEG data received from the camera
software interface. The video for the microcontroller and Bluetooth working is uploaded on PACE.

### 3.6 Android Application

The app creates files, sends messages and calls 911, as clearly shown in the video. We tested it by calling 911 (scheduled beforehand). However, in the video, one can see that we have called an emergency contact instead. The message is supposed to contain the user’s current location, shown in Figure 3.6 (a). The most recent location corresponds to figure 3.6 (b), so we know the location is accurate. The JPEG file starting with xFFD8 and ending with xFFD9 is created in the device storage.

![App sending location](image)

(a) (b)

Figure 3.6 - App sending location
4. Costs

4.1 Parts

Table 4.1 shows the prices of each of the individual major components of CapSafe. The total cost of parts for this project was $121.22.

<table>
<thead>
<tr>
<th>Part (Part No. if applicable)</th>
<th>Supplier</th>
<th>Price</th>
<th>Part (with manufacturer no. if applicable)</th>
<th>Supplier</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery Holder</td>
<td>Digi-Key</td>
<td>$1.26</td>
<td>Batteries (2x 3.7V Rechargeable Coin Cell)</td>
<td>Digi-Key</td>
<td>$20.26</td>
</tr>
<tr>
<td>SPDT Switch</td>
<td>Adafruit</td>
<td>$0.95</td>
<td>TLC3702CD Comparator</td>
<td>Mouser Electronics</td>
<td>$1.09</td>
</tr>
<tr>
<td>LD1117#50 5V Regulator</td>
<td>Mouser Electronics</td>
<td>$0.44</td>
<td>Force-Sensitive Resistor</td>
<td>Adafruit</td>
<td>$7.00</td>
</tr>
<tr>
<td>LF351D Op-Amp</td>
<td>Mouser Electronics</td>
<td>$0.72</td>
<td>ATMega328P-AU Microcontroller</td>
<td>Mouser Electronics</td>
<td>$3.57</td>
</tr>
<tr>
<td>nRF8001 BLE Breakout</td>
<td>Adafruit</td>
<td>$19.95</td>
<td>Tactile Switch</td>
<td>Mouser Electronics</td>
<td>$0.64</td>
</tr>
<tr>
<td>PCB</td>
<td>PCBWay</td>
<td>$18.00</td>
<td>TTL Serial Camera VC0706</td>
<td>Adafruit</td>
<td>$39.95</td>
</tr>
<tr>
<td>TQFP-32 to DIP-32 SMT Adapter</td>
<td>Proto Advantage</td>
<td>$7.39</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 4.2 - Projected labor costs

<table>
<thead>
<tr>
<th>Name</th>
<th>Hours Invested</th>
<th>Hourly Rate</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joshua Graham</td>
<td>150</td>
<td>$35</td>
<td>$5250</td>
</tr>
<tr>
<td>Kevin Horton</td>
<td>150</td>
<td>$35</td>
<td>$5250</td>
</tr>
<tr>
<td>Richa Meherwal</td>
<td>150</td>
<td>$35</td>
<td>$5250</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>450</strong></td>
<td><strong>$105</strong></td>
<td><strong>15,750 x 2.5 = $39,375</strong></td>
</tr>
</tbody>
</table>

#### 4.2 Labor

The total cost of labor for this project, with three engineers working 150 hours at $35 per hour, is $39,375.
5. Conclusions

5.1 Accomplishments

In the end, we were able to construct a complete, working prototype on a breadboard and implement most of its functionality on a printed circuit board. The FSR reliably generated the desired activation signal, which the microcontroller reliably interpreted correctly, making the camera take the desired number of pictures with the desired interval, and the Bluetooth connection was established successfully. In addition, the app functioned exactly as desired, sending the appropriate text message and calling the correct phone number.

5.2. Limitations

We could have improved this project in many ways, given ample time. The biggest of those is including low-voltage cutoff circuitry, which would protect the user from any of the dangers of over-discharging the battery. Additionally, with a little more time, we could have constructed a complete, fully-functional necklace rather than just a board on which to demonstrate that our design works. Last, because of our choice of part, we were forced to use an unreliable method of programming the microcontroller which, ultimately, made a complete, working printed circuit board impossible.

5.3 Ethical Considerations

With a project called a “personal safety wearable," safety is clearly our number one concern--and also the largest ethical sticking point. For the safety of the user with regard to the project itself, we planned to incorporate insulative materials surrounding the circuitry to protect the user from any risk of electrical shock. The user’s safety in use case, however, is a much bigger issue. Because no two assaults are the same, we cannot make a definitive recommendation to the user whether they would be best served alerting their attacker to their activation of CapSafe or whether that would put the user in greater danger; we can only advise that the user trust their judgment of the situation.

5.4 Future Work

Future work with this project will involve reselection of both the camera and the Bluetooth module with the intent of decreasing physical size and, especially for the camera, current consumption. In addition, the design as it exists is only feasibly marketable to women, and though women are more likely to be victims of assault, men are also potential victims and should not be ignored by this project. To that end, less gendered designs should be made available, such as on a watch or possibly even a ring. Last, future work on CapSafe should include greater software functionality in order to best aid the user; in particular, real-time GPS tracking, sending the captured images to emergency contacts, enabling a picture-only mode, and calling local law enforcement are highest among our priorities.
References


## Appendix A

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Testing &amp; Verification</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Battery and Safety</strong></td>
<td>1. Must not drop below 4.8V for 3 minutes when activated.</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2. Must not drop below 4.8V for 2 hours when on but not activated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Must not drop below 4.8V when for 24 hours when switch is off.</td>
<td></td>
</tr>
</tbody>
</table>
| **Checking battery life for activated circuit** | 1. Fully charge battery  
  2. Connect battery to complete circuit and turn on switch  
  3. Activate circuit by pressing FSR for 3+ seconds.  
  4. Probe battery and ensure battery remains above 4.8V for 3 minutes |        |
| **Checking battery life for standby circuit** | 1. Fully charge battery  
  2. Connect battery to complete circuit and turn on switch  
  3. Do not activate circuit, probe battery, and ensure battery remains above 4.8V for 2 hours |        |
| **Checking battery life for leakage current** | 1. Fully charge battery  
  2. Connect battery to complete circuit and turn off switch  
  3. Probe battery voltage and ensure battery remains above 4.8V for 6 hours |        |
| **Safety**            | 1. Low battery indicator signal sent when battery voltage <= 5.3V.                     | 5      |
|                       | **Checking low battery indicator**  
  1. Place battery with charge > 5.3V in circuit.  
  2. Probe low battery signal and ensure it goes low when battery reaches 5.3V or lower, until 4.8V. |        |
| **Activation System** | 1. Voltage outputted by Force Sensing circuit > 3 V  
  2. Comparator outputs a logic high when FSR is pressed | 5      |
|                       | **Voltage Swing Test**  
  1. Probe voltage across FSR  
  2. Measure voltage when FSR is unpressed  
  3. Measure voltage when FSR is pressed |        |
|                       | **Comparator Test**  
  1. Probe output of comparator  
  2. Verify logic low when FSR is unpressed (~120 mV)  
  3. Press FSR  
  4. Verify logic high (~3.3V) |        |
| **Filter**            | 1. Passband attenuation < 6-dB | 2      |
|                       | **Passband Attenuation**  
  1. Probe voltage across 10-kΩ resistor with oscilloscope at the output of the filter  
  2. Generate fixed-amplitude sine wave through the filter, checking 20, 200, and 2000 Hz |        |
<table>
<thead>
<tr>
<th>Feature</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rolloff &gt;6dB per decade</strong></td>
<td>3. Verify the voltage to be at most 6dB under input voltage</td>
</tr>
</tbody>
</table>
| **Stopband stays below -24 dB** | **Stopband Rolloff**  
1. Continue probing voltage across 10-kΩ resistor  
2. Generate fixed-amplitude sine wave through filter, checking 1, 2, 3, 4, and 5 times the cutoff frequency  
3. After measuring cutoff-frequency attenuation, compare other integer-multiples to verify rolloff is as specified |
| **Stopband Value** |  
1. Continue probing voltage across 10-kΩ resistor  
2. Generate fixed-amplitude sawtooth wave through the filter at 10 times cutoff frequency  
3. Verify that output voltage is less than -24 dB of input voltage |
| **Microcontroller** | **Operation Voltage Test**  
1. Connect microcontroller to adjustable power supply  
2. Set a pin output always high  
3. Sweep voltage supply 4.8-8.4V  
4. Probe pin and verify high signal |
| - Operates under 4.8-8.4V | **Activation Signal Test**  
1. Read and print the output of the digital pin that takes the input as the activation signal described before  
2. Verify if it is HIGH only when the FSR is pressed |
| - Receives and checks the activation signal from the Force sensing circuit | **Bluetooth Communication Test**  
1. Program the microcontroller to poll the values  
2. Implement pseudo code 2.1  
3. Print the values on the app console |
| - Should be able to receive and send commands to the Bluetooth module | **Camera Communication Test**  
1. Set to software Serial mode  
2. After initialization of SD card, send the PhotoTakeCommand() to the camera with an induced delay of 2-3 seconds for every picture.  
3. To read the bytes in the picture, initiate the ReadDataCmd() and read until the EOF |
| - Should be able to send commands to the camera | **Ensuring operation works under supply**  
1. Connect 5V power supply to module  
2. Probe 3.3V out and ensure outputs correctly |
| **Bluetooth Module** | **Verifying current consumption**  
1. Power on module  
2. Probe VIN with oscilloscope |
| - Operates under 5V supply | - Maximum current draw is 30mA |
| - Operates under 5V supply | - Ensure operation works under supply |
| - Operates under 5V supply | - Verifying current consumption |
- Minimum wireless range at least 0.05 m
- Max latency no more than 0.2s

3. Ensure current draw is less than or equal to 30mA

**Wireless Range Test**
1. Upload test code to microcontroller to send one signal to phone
2. Place phone 2m away from Bluetooth module
3. Use app and verify connection is made

**Minimum latency test**
1. Configure the Bluetooth Module to send at it’s default baud rate which is 19200
2. Set the connection interval to 100ms. Thus Effective Connection Interval = 500ms
3. Therefore, we can transmit the data within this interval by running a while loop in the code and check if anything is received on the software side within this interval and print it on the interface

<table>
<thead>
<tr>
<th><strong>5V Linear Regulator</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Can handle input voltage up to 8.4V</td>
</tr>
<tr>
<td>- Output Voltage = 5±0.3V when input=7V</td>
</tr>
<tr>
<td>- Current range &gt;300mA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Input Voltage Range Test</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Connect regulator to 8.4V supply</td>
</tr>
<tr>
<td>2. Probe output and verify 5±0.3V</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Output Voltage Test</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Connect regulator to variable power supply</td>
</tr>
<tr>
<td>2. Send 7V to regulator</td>
</tr>
<tr>
<td>3. Probe output and verify 5±0.3V</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Current Range Test</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Connect regulator to 7V from power supply</td>
</tr>
<tr>
<td>2. Attach 30 ohm power resistor in series to ground</td>
</tr>
<tr>
<td>3. Ensure power supply output current &gt;=300mA</td>
</tr>
<tr>
<td>4. Probe output of regulator</td>
</tr>
<tr>
<td>5. Ensure regulator output remains 5±0.3V</td>
</tr>
</tbody>
</table>

**Software Interface**
- Must be an android or IOS application
- Must be compatible with the Bluetooth module
- Must be able call 911, send location to emergency contacts
- Should be able to create hex files from the jpeg data
- Conversion of hex data into jpeg

**Verification:**

**Without Bluetooth module**
1. Make the application
2. Make buttons on the application such that each button executes one function.
3. Tap every button to check if the calls are generated or location data is sent to the contacts specified

**With Bluetooth module**
1. Poll the Bluetooth RX port till it is has data packets available
2. If a valid data packet (ASCII data) is detected, print it on the screen.
3. Later activate an in-built java timer of 15 seconds upon valid character detection
4. If >15 seconds the app places a call and sends location to contacts

**File Creation Verification:**
1. Go to /Files/Device Storage
2. Check if the corresponding files are created
3. For ensuring that the hex files are complete, check FFD8 as the start of the file and FFD9 as the File End.

| Camera | 1. Set the image size (chose 640 x 480) | 2. Execute takePicture() |
|        | 3. Create a filename after checking that it already does not exist in the SD card [5] | 4. SD.open that file |
|        | 5. Get the size of the image | 6. Till image is fully read, write 32 bytes to a buffer |
|        | 7. And write that buffer to the file that SD card opened | 8. After the entire picture is read, close the file |
|        | 9. Remove SD card and check for the pictures | 5 |

- Takes pictures to be saved in less than one minute
- Pictures must be less than 10 KB
Appendix B - Complete Schematic Diagram and Block Diagram

Figure B.1 – The full circuit schematic

Figure B.2 – The full project block diagram