Introduction

Following the selection, evaluation and validation of the sensors the device would be using, the team required a method by which to integrate them all in a compact manner and in such a way as to allow for a ‘plug-and-play’ user hardware interface. Additionally, since the above tests were conducted on prototyping test beds, such as breadboards and regulated benchtop power supplies, we needed this solution to incorporate a mobile regulated battery power supply. Additionally, the solution needed to incorporate the LEDs that would give a patient and caregiver information about the alarm state.

Subsequently, the team decided to use a custom-made printed circuit board (PCB) to achieve this. A custom PCB allows for a bespoke circuit design that is easier to debug than alternatives (like a solderable protoboard) in a compact design. This would also ultimately allow more flexibility for further amendment as well, given the design considerations specified below.

Component Selection and Schematic Capture

When making these considerations, we had to keep in mind that we had limited time to get a functioning prototype. As such we used as many off-the-shelf components for non-bespoke functionalities as we could. This would mean that we save time in testing, manufacturing and validating the PCB, as we would not need to assemble and test as many subsystems. Additionally, as we only had time for one PCB revision, we wanted to make the design as flexible as possible to allow us to integrate new functionalities as fast and seamlessly as possible.

Power Management

We decided to continue using the Adafruit batteries combined with their LiPo battery charger. This prevented us from having to integrate battery charge and safety circuity on the board. The charge controller’s output was connected to the Arduino’s built-in 3V3 regulator, further removing the need for a regulator circuit.

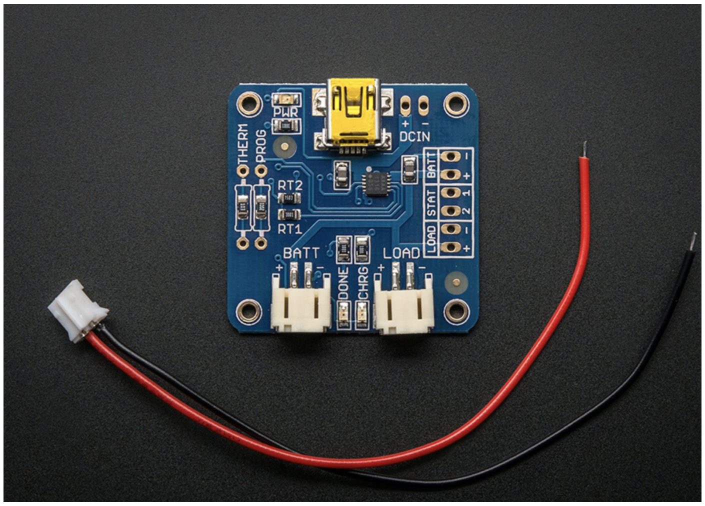


Figure XX: Adafruit's Battery USB charger https://www.adafruit.com/product/259

Safety and User Interfaces

In addition to the Adafruit charge controller for battery safety, we needed to keep in mind that we are interfacing with the human body. As such, we needed to have current and voltage limiters – in the form of a resistor and diodes respectively – on each lead that connects to the human body. As demonstrated below, a series 330kΩ resistor is a current limiter and the diodes will short current to ground if voltage exceeds their 0.6V forward voltages. The diodes are through-hole to allow for high currents to flow before burning out.

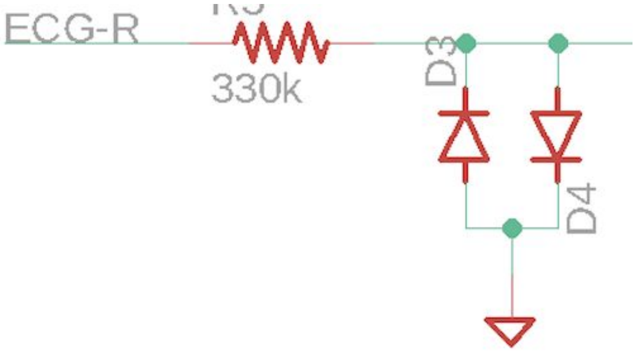


Figure XX: ECG Lead Safety Circuitry. Design courtesy of Evan Smith and ES152.

In order to indicate alarm states to the user and to allow us to determine the state of the program while debugging, we included three programmable LEDs and one power indicator LED. While the current draw is on the order of 10mA which reduces battery lifetime, the only LED on consistently is the power indicator LED, which is required for the user to establish correct operation. The others are programmable and can be turned off from the microcontroller. These components are surface mounted devices (SMD) to reduce footprint size.

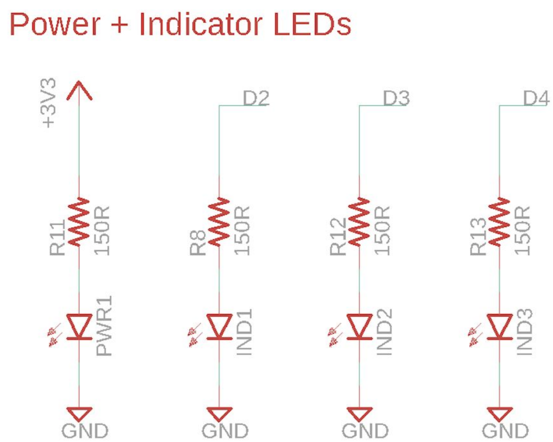


Figure XX: Indicator LEDs

Arduino Pro Mini

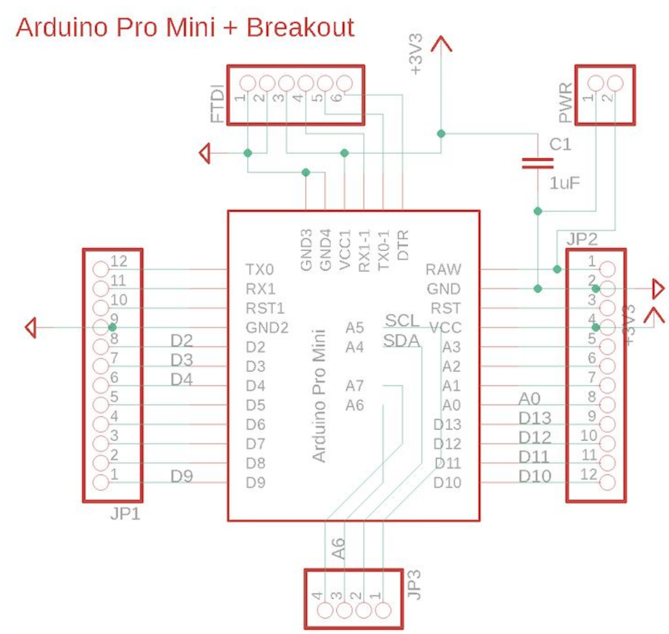


Figure XX: Arduino Pro Mini Schematic

We chose to use Sparkfun's Arduino Pro Mini due to its compact size and high performance specifications. This would be mounted onto the PCB by three sets of 2.54mm female header pins that connect to the 2.54mm male header pins soldered downwards from the Arduino. This way, we could easily swap out microcontrollers should one break or get damaged.

Keeping in line with the expandability specification, we broke out all of the arduino pins to 2.54mm male header pins that would allow us to easily connect the Arduino to extra components should the need arise. Likewise, we broke out the unregulated and 3V3 regulated power pins for debugging.

In order to program the Arduino Pro Mini, we need to use an off-board FTDI chip to translate the Rx/Tx serial port on the arduino to USB to interface with the Arduino IDE. These pins were likewise broken out and positioned to allow for easy connection to Sparkfun's FTDI connector.

All wires in the schematic that have labels are used on the PCB. The others are unused.

Radio Connector

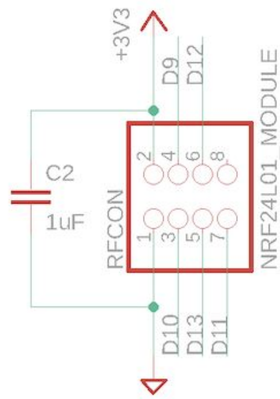


Figure XX: Radio Module Connector Schematic

In order to further ease prototyping, the RF radio connector was designed to match the pinout of the RF radio module, thus allowing for the module to sit neatly on top of the PCB. Again, this prevents the need for us to assemble a sensitive radio module on the PCB proper and allows us to swap the modules should one malfunction. C2 is bypassing the 3V3 supply bus to improve power stability. The connector is a 2.54mm female 2x8 header, thus also providing mechanical stability as well as a good electrical contact.

ECG Analog Processing

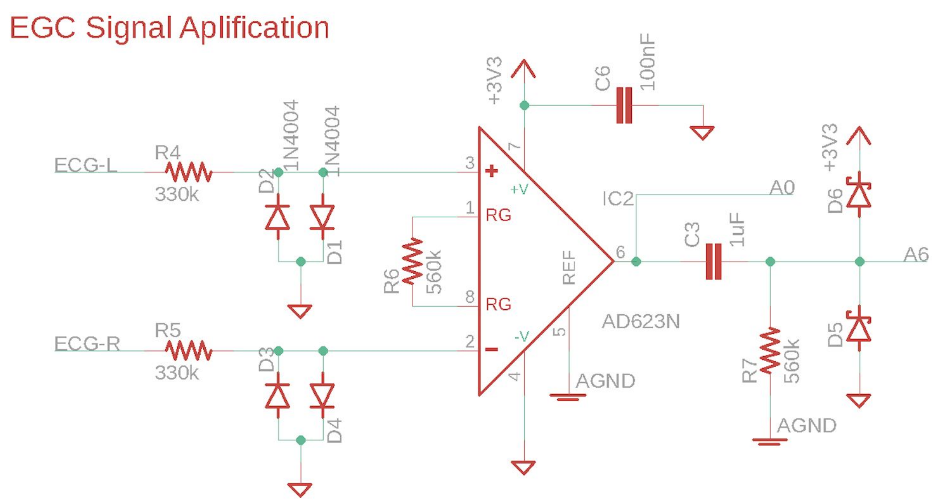


Figure XX: AD623N Schematic

The rationale behind using this topology has been covered earlier. The diodes and AD623N are all through-hole to allow for easier reworking of the component topology should the need arise. The capacitors and resistors are SMD to minimise the size of this circuit. C6 is bypassing the 3V3 supply bus to improve power stability. D5 and D6 are schottky diodes to protect the arduino pins from exceeding their specified 0V – 3V3 range.

AGND refers to the pseudo-ground reference supply that is required for single-supply operation. As we are using a battery to power the circuit, in order to process negative voltage analog signals, we use AGND as a zero-reference for all analog circuitry to ensure that the voltages at the microcontroller inputs stay within the 0V – 3V3 range.

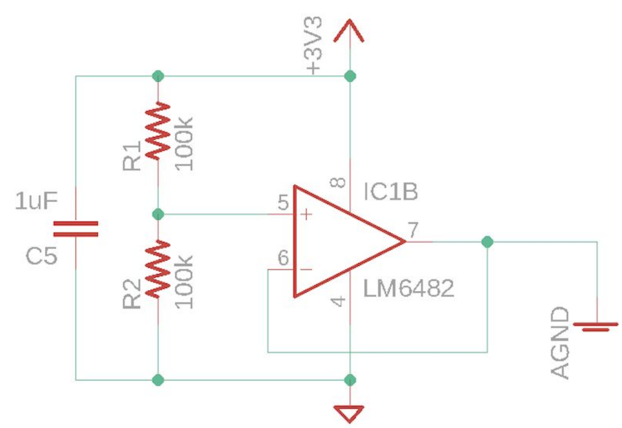


Figure XX: 1.15V Pseudo-Ground Schematic

Finally, to improve the user experience when interacting with the device, we decided to use a familiar 3.5mm jack ("headphone connector") to connect the ECG leads to the device.

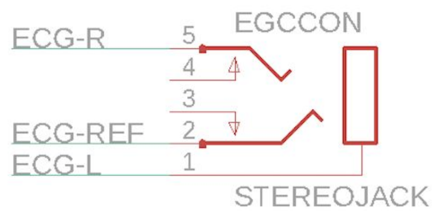


Figure XX: 3.5mm Jack Schematic

Ferromagnetic Random Access Memory Chip

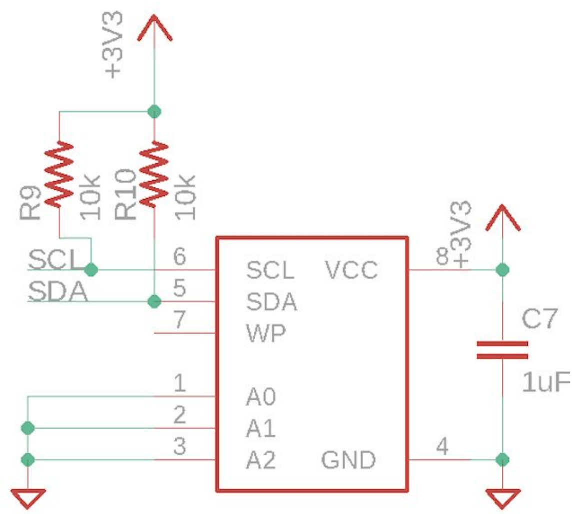


Figure XX: FRAM Schematic

Before the sensor algorithms were optimised, there was a need for extra off-board memory in the form of an I2C FRAM chip. Ultimately it was not used. It is worth noting that if we did indeed need to use it, in future revisions of the board we would simply choose to use a different microcontroller rather than adding this component. In our case, we were limited by size constraints on the PCB, and the Arduino Pro Micro fit within those constraints.

Temperature Sensor Buddy Board Connection

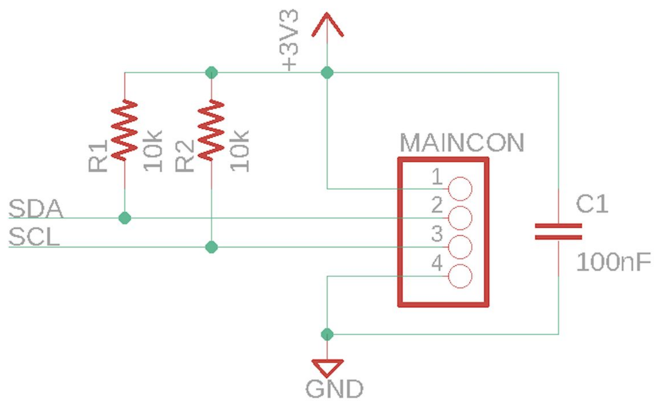


Figure XX: JST-PH Schematic

We decided to use a seperate board for the chosen temperature sensor. Due to its small size, the second board – called the Temperature Sensor Buddy Board – would break out the small pads on the temperature sensor and allow us to easily interface the sensor with the main board. We chose to use the smaller 2mm JST-PH connector family to interface the two boards as shown in the schematic. Capacitors on both ends bypass the 3V3 supplies. We chose the JST-PH because of its ubiquity, ability hand assemble and compact size that would not be too bulky on a patient.

Component Placement

Throughout this process, we used Eagle version 9.2.2 from Autodesk, as it provided a wide feature set that was easily available on the educational license. We decided to use a two-layer design to ease PCB design without exceeding the PCB budget. When placing components, we needed to keep several things in mind to minimise the size of the PCB while still maintaining flexibility in the design, ideal component placement for optimal functionality and ease of hand assembly. These are the things that we considered:

- All off-board connectors go on the edge of the PCB
- Analog components and traces are separated from digital ones to improve noise response
- Analog signal trace lengths are minimised to reduce noise
- Four 3mm diameter mounting holes are needed for mechanical mounting of the the main board. Four 2mm mounting holes are needed on the Temperature Sensor Buddy Board.

- Bypass capacitors are placed next to the power connections of their respective component. Additional bypass capacitors are added in sparsely populated areas of the board
- 25mil thick (1mil = 0.001 inches) traces for power to provide higher currents with minimal resistive losses
- 16mil thick analog and digital signal traces for reduced noise and reduced ground coupling.
- Arduino breakout headers are positioned close to and in line with the Arduino interface headers to ease pin number recognition on the prototype
- I2C 10kΩ pullup resistor pair located next to the Arduino I2C pins and on the Temperature Sensor Buddy Board to reduce noise
- RF radio module connector is located in such a way as to ensure that the radio module stays within the dimensional bounds of the PCB
- LEDs positioned on the edge to allow for good visibility.

In addition to the above considerations, we also made the top and bottom layers of copper ground planes with buried vias connecting the two at regular intervals to reduce noise and ground coupling. For the silkscreen – the white layer of text covering the solder mask – we placed component numbers to allow for easy component recognition when hand assembling the PCB. Additionally, we placed the PCB name and version number on the back solder mask, as well as the name of the designer to ensure accountability and help people know who to ask in case of questions. A white box is placed on top to allow for easy numbering of individual boards for our inventory and rework tracking system.

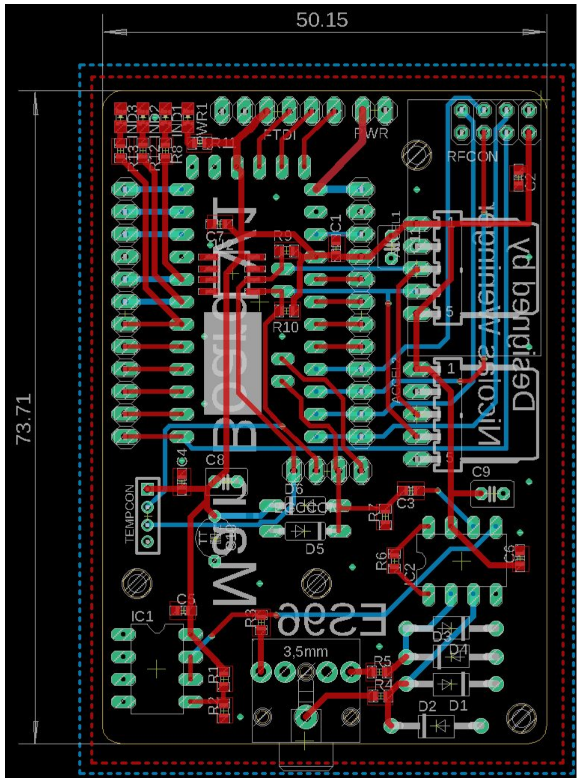


Figure XX: MainV1 Board Layout View. Units in mm.

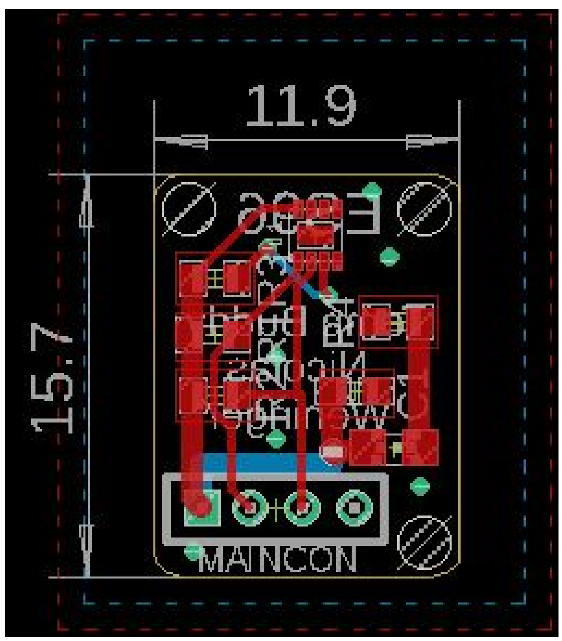


Figure XX: Temperature Sensor Buddy Board Layout View. Units in mm.

Manufacturing

Both PCBs were manufactured by a 3rd party manufacturer, as we did not have access to the requisite tools in-house. The order was placed just before thanksgiving break so that they would arrive the week following the break. Before the order was placed, we preformed design rule checks on the schematic and board files of both PCBs to ensure that we were in compliance with the tolerances stated by the manufacturer.

Assembly and population of the boards with components was done in-house using a combination of equipment available the SEAS EE lab in Maxwell-Dworkin and Paul Horowitz and Jim McArthur's electronics lab in Cruft. Leaded solder paste and a fine tipped soldering iron was used to solder the SMD components. Following each SMD component being placed, we verified that the connection was as intended by checking for the resistance of the component. All capacitors had infinite resistance and all resistors had a resistance within ±1% of the marked value, as expected.

Through-hole components on the main board were soldered using leaded solder as well, as this makes it easier to conduct reworks. On the Temperature Sensor Buddy Board, unleaded solder was used, due to safety concerns with skin contact.

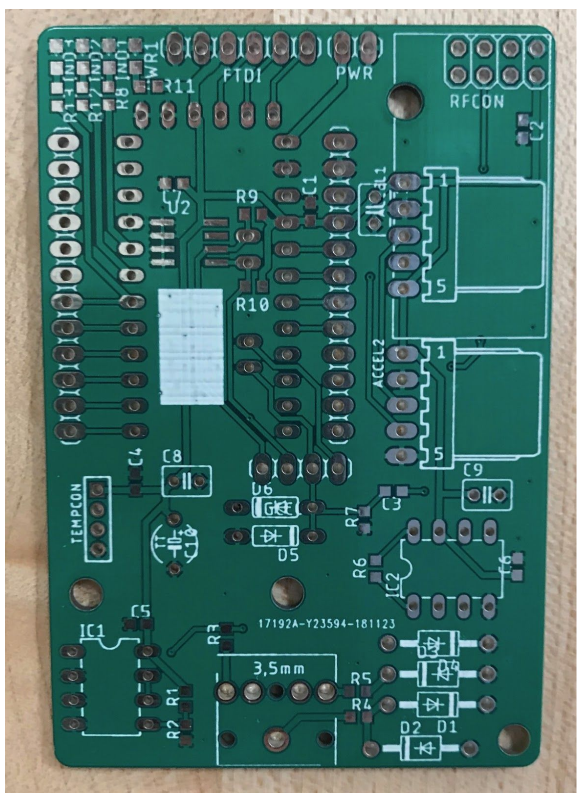


Figure XX: Unpopulated MainV1 PCB front

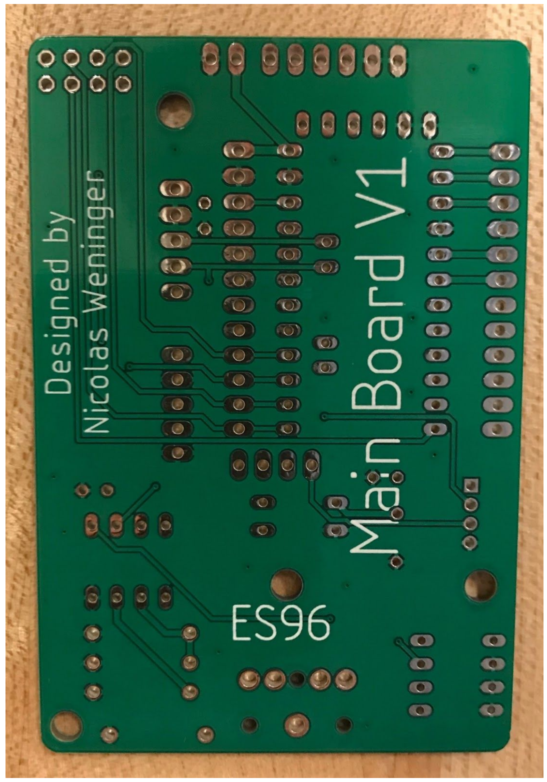


Figure XX: Unpopulated MainV1 PCB rear

The Temperature Sensor Buddy Board proved to be too challenging to assemble, as the footprint for the temperature sensor is designed for machine assembly rather than hand assembly. We were thus unfortunately unable to use the Temperature Sensor Buddy Board as planned. However, we did devise an alternate temporary solution, as detailed in the following section.

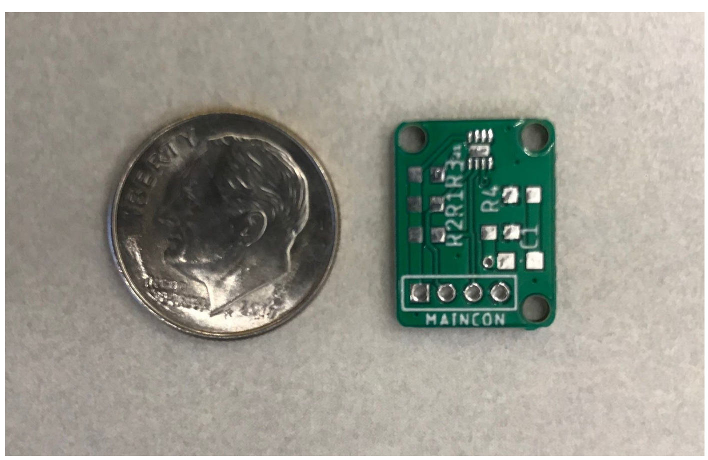


Figure XX: The unpopulated Temperature Sensor Buddy Board front, as compared to a nickel

Figure XX: The unpopulated Temperature Sensor Buddy Board rear

Required Reworks

Due to oversights in our design validation process and subsequent workarounds to these problems, a few things on the Main PCB ECG circuitry needed to be reworked, as discussed in an earlier section:

- Added a 10kΩ potentiometer to the former location of a now unused accelerometer connection. Connected 3V3 and GND
- Pin 3 of the op-amp was connected to the potentiometer wiper connection
- Pin 6 of the AD623N was connected to Arduino pin A0
- Pins 1 and 2 of the op-amp were shorted together
- R3 was not placed
- The ECG reference lead connection was connected to Pin 1 of the op-amp
- R7 of the high-pass filter was connected to pseudo-ground, not ground, via a different 560kΩ resistor.

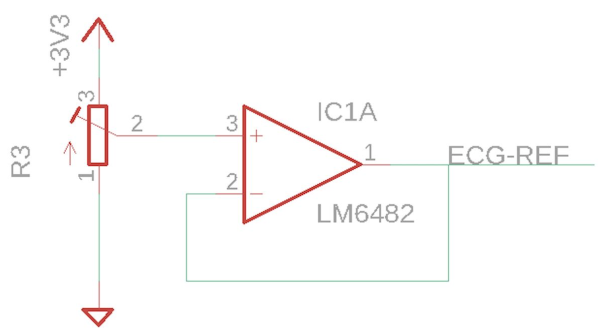


Figure XX: Reworked ECG Adjustment Schematic

In order to overcome the challenges faced with the Temperature Sensor Buddy Board in such a way to let us interface with the main PCB as desired, we decided to use a small protoboard combined with a breakout board for the temperature sensor that we had previously assembled and validated. This would allow us to wire the sensor to a 2.54mm connector that would be wired to the male JST-PH connector for the main PCB, as well as including the external circuitry required for proper operation of the sensor.

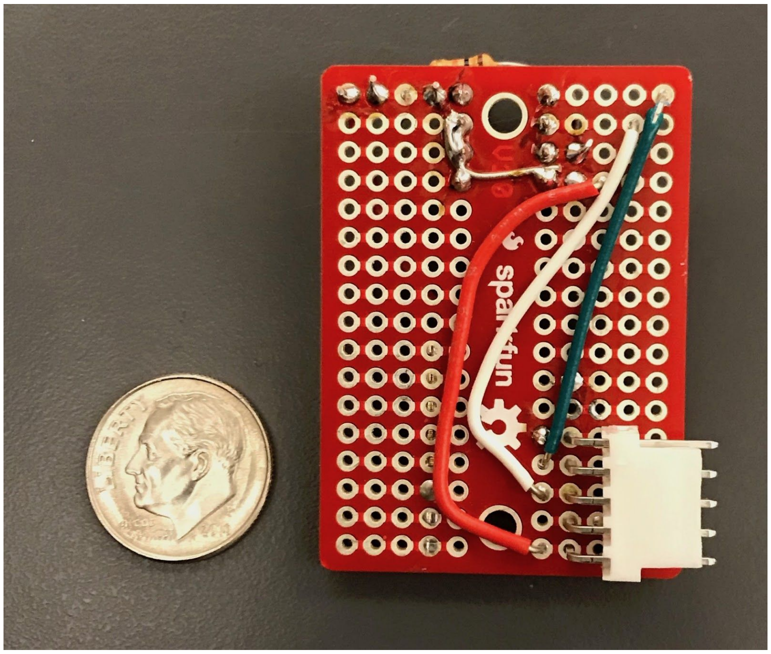


Figure XX: Temporary Temperature Sensor Buddy Board as compared to a nickel

Final MainV1 PCB Prototype

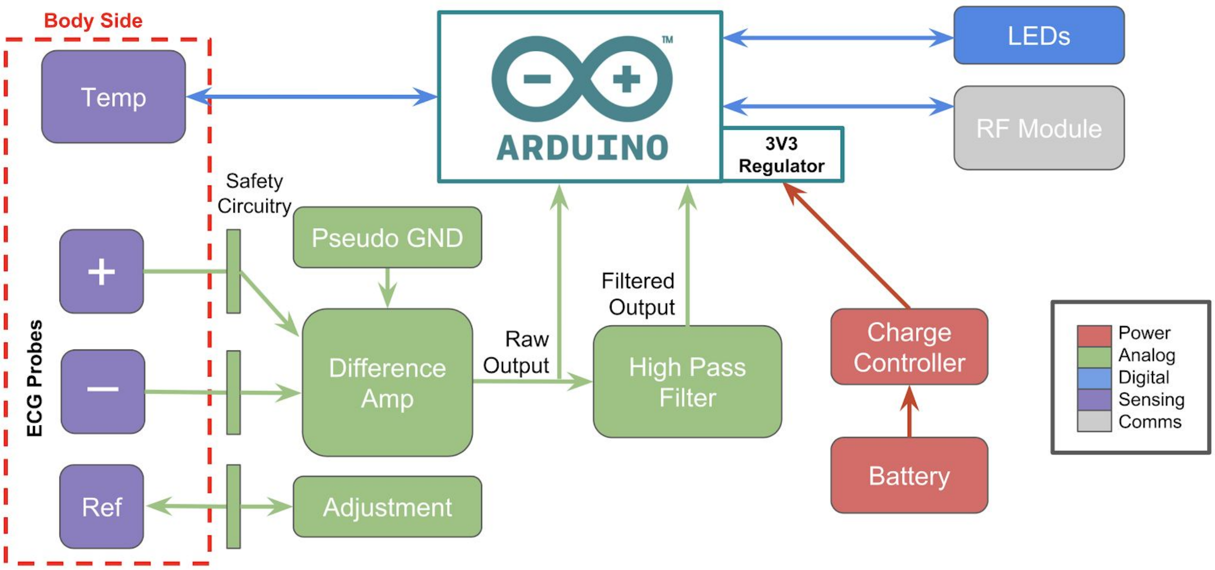


Figure XX: Final MainV1 PCB Block Diagram

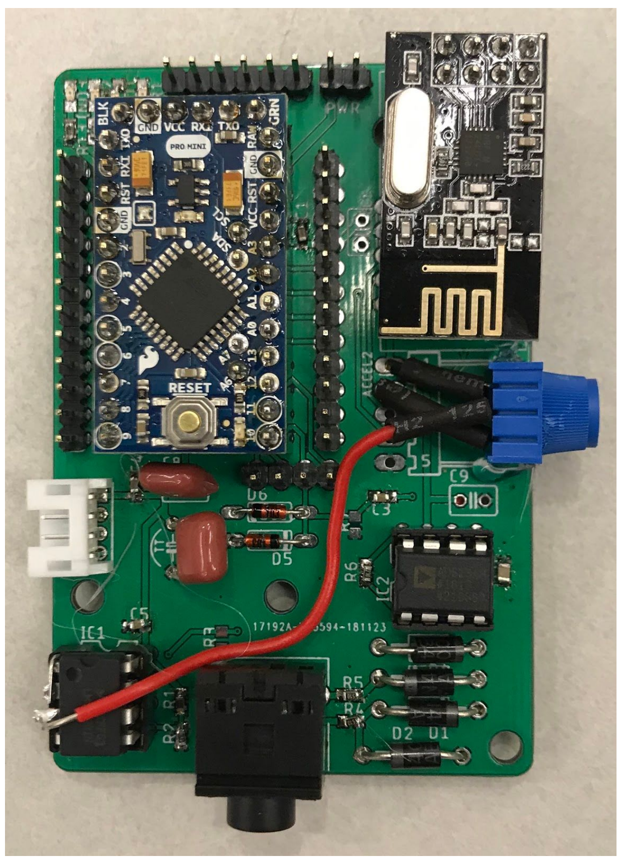


Figure XX: Populated MainV1 PCB front

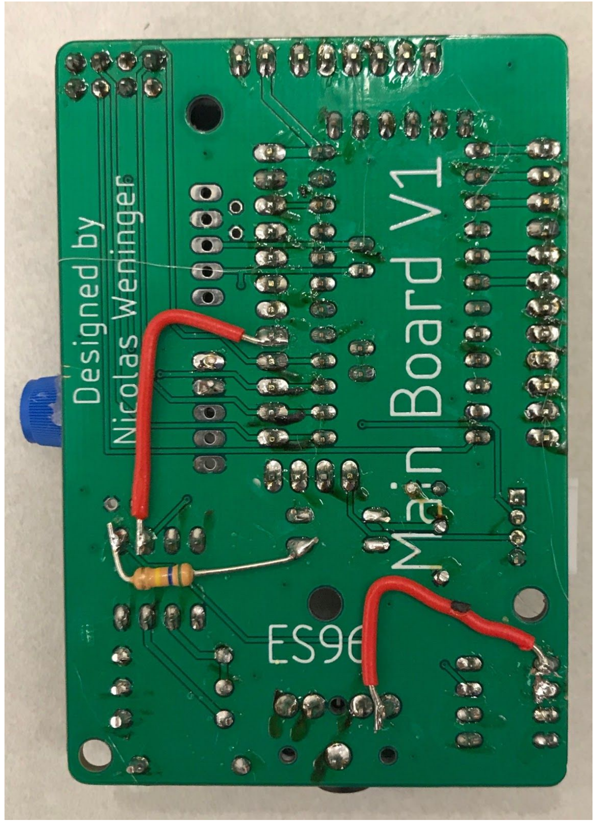


Figure XX: Populated MainV1 PCB rear

PCB Bringup and Validation

Once the PCBs were assembled, we conducted tests to see whether all subsystems were working correctly. The steps and results are outlined below. Between every step, power is turned off and on again to prevent issues that might arise with hotplugging. All but one subsystem worked – the ECG analog processing. Unfortunately, we did not have time to fix this. Discussion is continued in the next steps chapter.

Instructions	**Subsystem Under Test**	**Result**	**Observations**
With nothing plugged into PCB, power up.	PCB assembly		No shorts and issues with power delivery.
Plug in Arduino	Arduino		3V3 regulator and electrical pin connections work.
Plug in ECG leads.	ECG analog processing		Components work but output is saturated. Likely issue with high-pass filter.

Turn on pins D2-4	LED Indicator LEDs		All LEDs illuminate
Plug in RF module	RF Module		Rx/Tx with hub confirmed.
Plug in temperature sensor	Temperature Sensor Buddy Board		Data from Buddy Board successfully read via I2C.

Table YY: Results of PCB bringup and validation

We also needed to validate whether the V1 prototype indeed allowed for 1 full day of operation on the chosen 200mAh battery. We first conducted a preliminary theoretical lifetime analysis, looking at the current draw of the major components in the full circuit. This analysis indicated that the current battery would meet the specified 1 day lifetime.

Measurements of the “as built” hardware indicated that the current draw was higher than the preliminary analysis indicated. This is likely due to passive components such as voltage dividers and filters, and differences in specific component current draw within manufacturer stated tolerances. The measured current draw still comfortably meets the 1 day lifetime requirement.

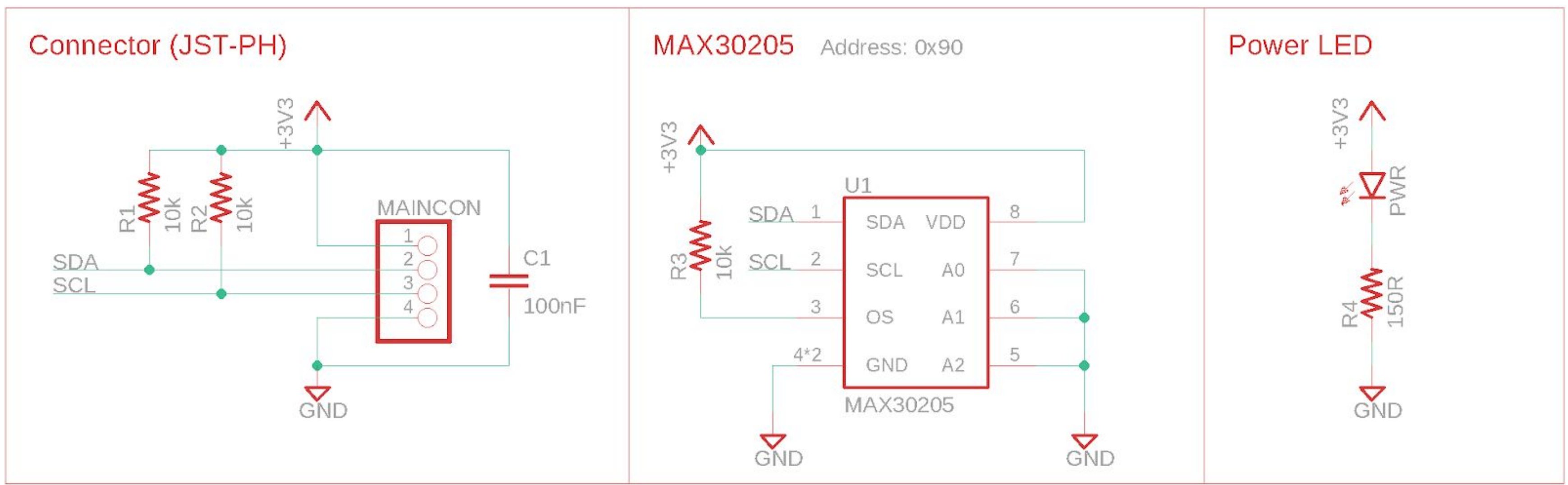
Component	**Average Current Draw (mA)**	**Comment**
LEDs	20.00	Only 2 power LEDs on continuously
AD623N	0.55	
Arduino Pro Mini	4.40	No modifications to the Arduino. In active mode.
RF Module	12.30	
LM6482	1.20	
MAX30205	0.93	
Adafruit Charge Controller	2.00	MCP73833
TOTAL	39.38	
Expected Lifetime (h)	**50.79**	

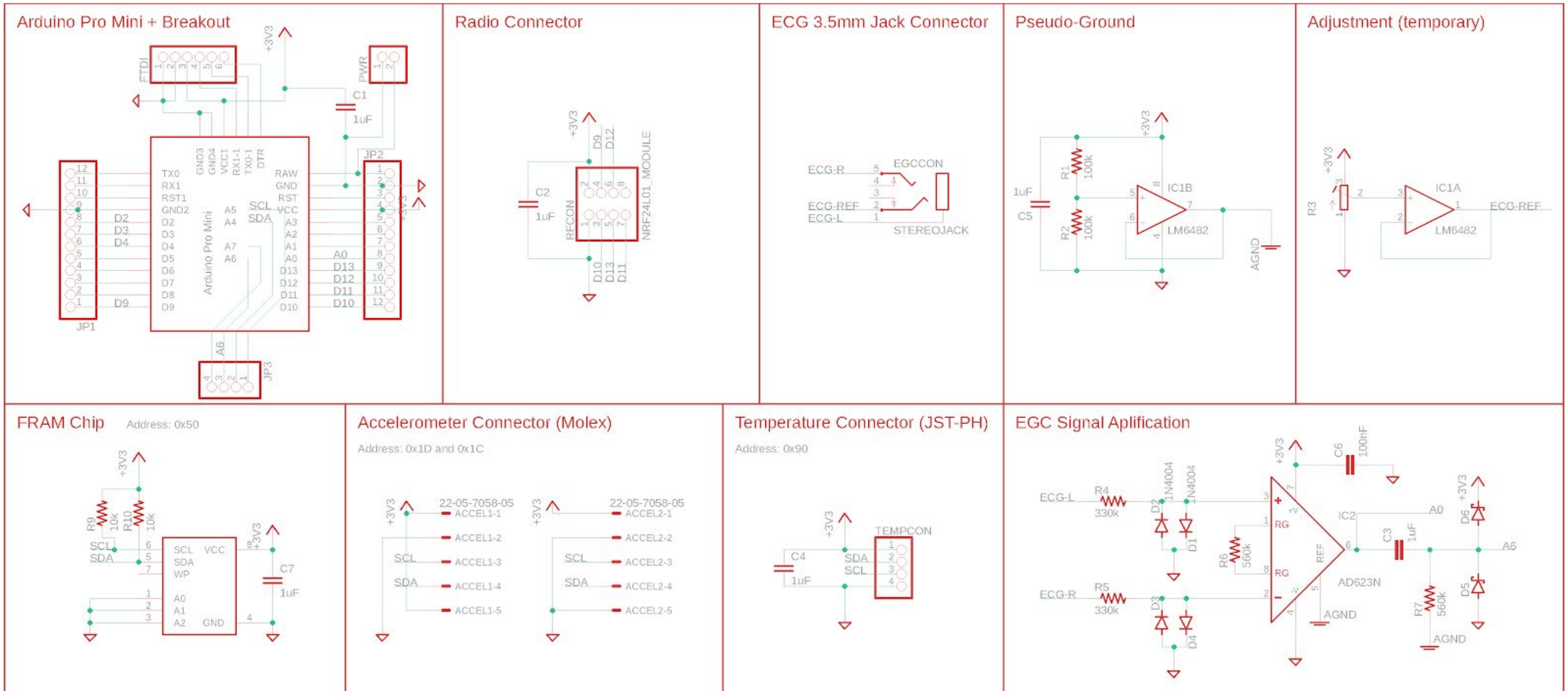
Table YY: Theoretical device lifetime analysis

Battery Capacity (mAh)	2000
Max. Current Draw (mA)	55
Average Current Draw (mA)	41.5
Minimum Lifetime (hours)	**36.4**
Average Lifetime (hours)	**48.2**

Table YY: Measured current draw lifetime analysis

Appendix I: Main Board and Temperature Buddy Board Full Schematics


Connector (JST-PH)
+3V3
R1
10k
R2
10k
MAINCON
1
2
3
4
SDA
SCL
C1
100nF
GND
MAX30205
Address: 0x90
+3V3
U1
R3
10k
SDA
SCL
1
2
3
4*2
SDA
SCL
OS
GND
VDD
A0
A1
A2
8
7
6
5
MAX30205
GND
GND
Power LED
+3V3
PWR
R4
150R
GND



V2 Notes:

Adjustment is a workaround to an issue with the critical freqiency of the high-pass filter C3/R7
Pin 6 of the AD623N is connected to A0 in order to determine whether there is saturation of the ECG signal