If There’s a Will, There’s a Wei

RRC table, system diagram, and functional decomposition

# RRC Table

## Use Case 1: Device Setup

|  |  |  |
| --- | --- | --- |
| **Responsibilities** | **Roles** | **HW or SW** |
| Device power on | Microcontroller | HW: small switch |
| Device power LED | Microcontroller | HW: LED |
| Device and base station connection | Communications | Both: SW connection & data, HW transmission |
| LED light blink on connection | Microcontroller | Both: SW power control, HW LED |
| Device to base station transmission | Communications | Both: SW connection & data, HW transmission |
| Base station to device transmission | Communications | Both: SW connection & data. HW transmission |
| Computer can read data from base station | Base station | Both: SW UART component, HW FTDI chip |

## Use Case 2: Device Disconnected

|  |  |  |
| --- | --- | --- |
| **Responsibilities** | **Roles** | **HW or SW** |
| Visualization of received data | Computer | SW visualization component |
| The device shuts down upon removal | Sensor array | Both: HW to read the abnormal signal, SW to safely power off the device |
| Device reconnects to base station on power on | Microcontroller | Both: SW connection manager and HW to facilitate the connection |
| User can force the device to reconnect via a setup button | Microcontroller | Both: SW connection manager, HW facilitates |

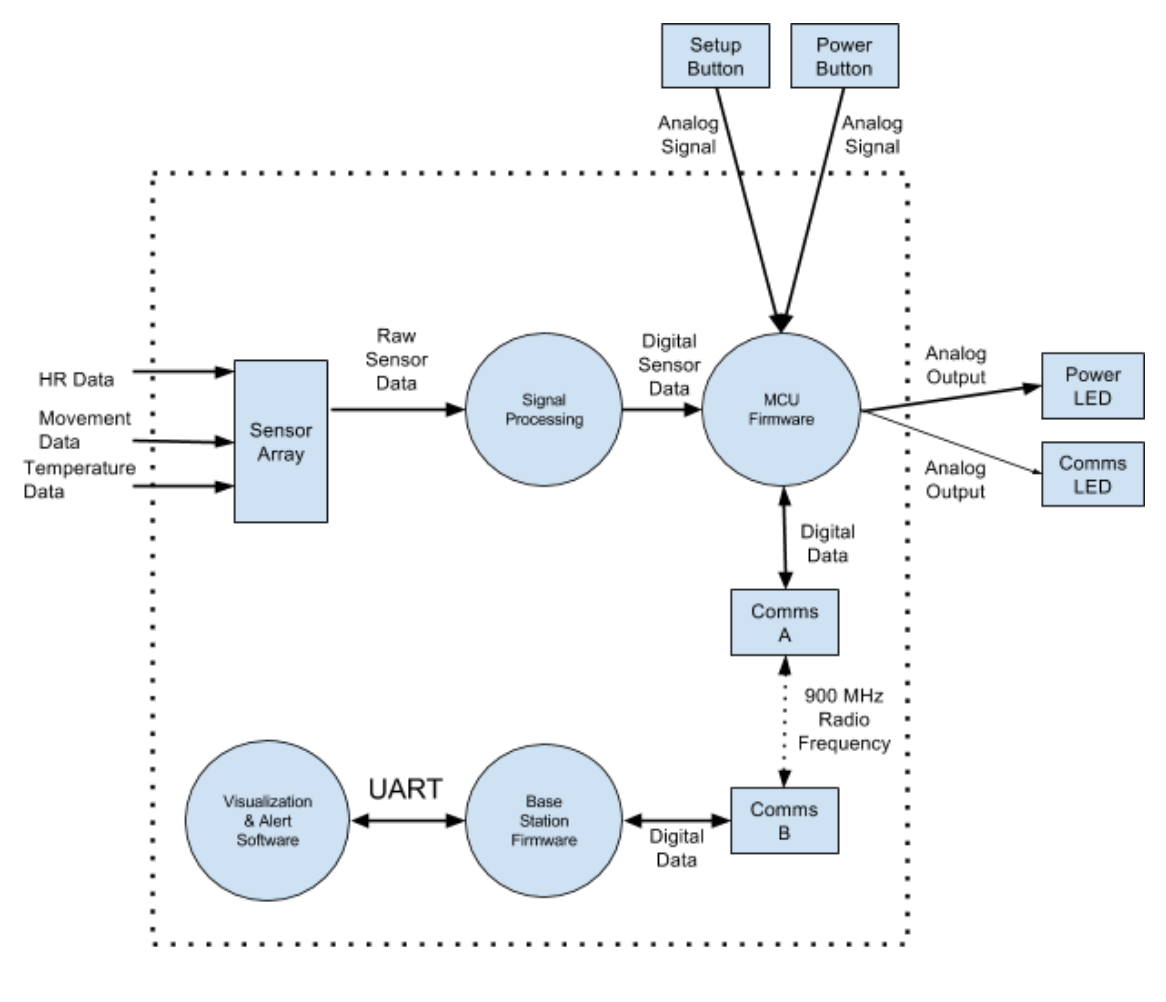
## Use Case 3: Regular Use

|  |  |  |
| --- | --- | --- |
| **Responsibilities** | **Roles** | **HW or SW** |
| Device can collect heart rate data | Sensor array | Both: HW sensors, SW signal processing |
| Device can collect temperature data | Sensor array | Both: HW sensors, SW signal processing |
| Device can collect movement data | Sensor array | Both: HW sensors, SW signal processing |
| The computer can set off an alarm if there are irregular values of any of the collected sensor readings | Computer | SW: service sends alert message |
| A user can deactivate the alarm | Computer | SW: button on the computer display |
| The warning LED blinks when the alarm is triggered | Microcontroller | Both: SW alert component, HW LED |
| The warning LED stops blinking when the alarm is deactivated | Microcontroller | Both: SW alert, HW LED |

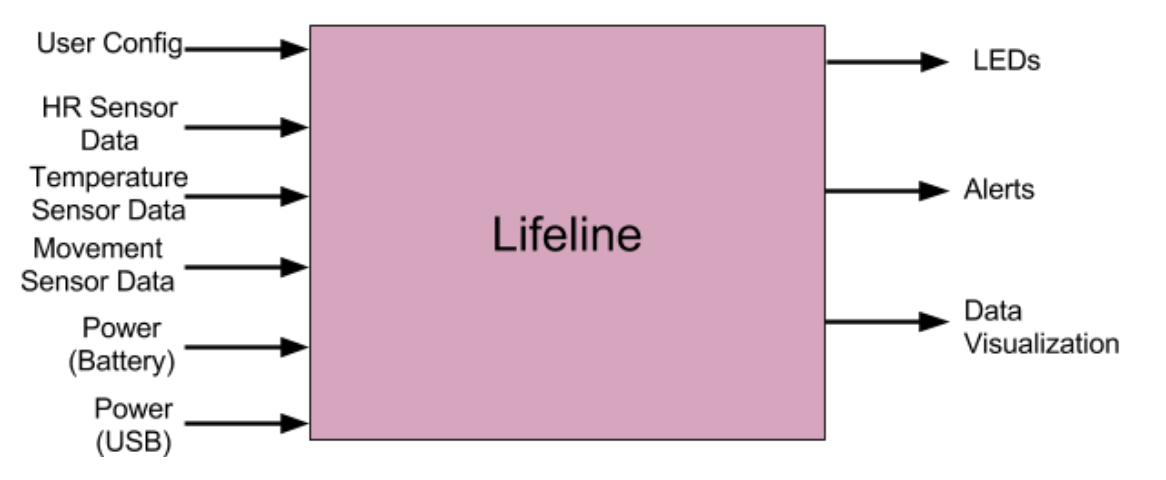
## Sponsor Requirements

|  |  |  |
| --- | --- | --- |
| **Responsibilities** | **Roles** | **HW or SW** |
| The device is battery powered. | Power system | HW: Battery & power management IC |
| The device can be charged via an external source. | Power system | HW: Power management IC |
| Device battery status can be relayed to the base station. | Power system | Both: HW fuel gauge, SW power management |

# System Diagram

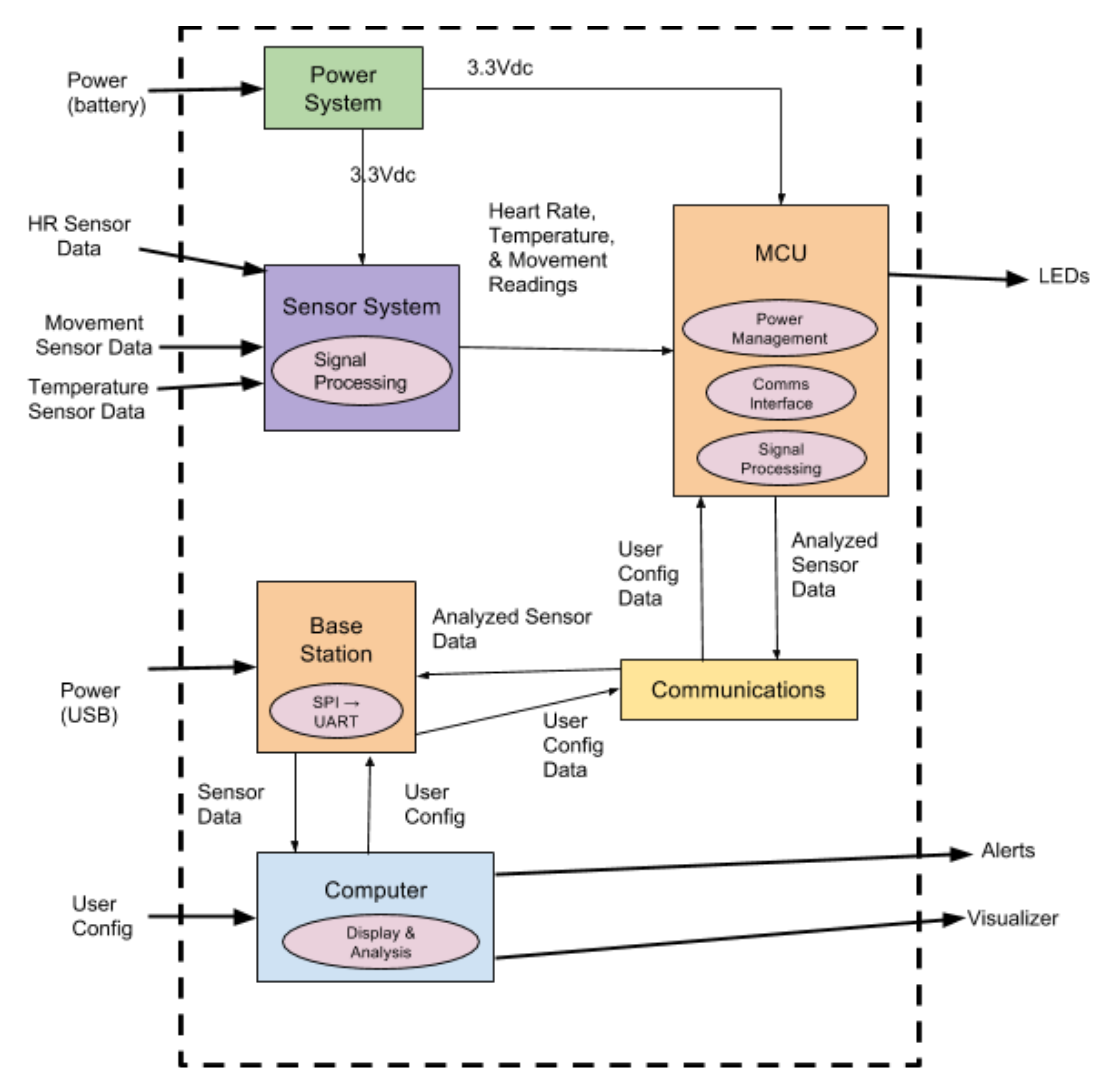


# L0 Functional Decomposition



|  |  |
| --- | --- |
| **Module** | Lifeline |
| **Inputs** | Battery power, USB power, temperature sensor data, movement sensor data, heart rate sensor data, and user config |
| **Outputs** | LEDs, alerts, and data visualization |
| **Functionality** | Takes data from the sensors, communicates them to a base station, and outputs alerts and data visualizations. The device is powered via a battery and has various LEDs to alert the user of certain events. |

# L1 Functional Decomposition



|  |  |
| --- | --- |
| **Module** | Power system |
| **Inputs** | Power (battery) |
| **Outputs** | 3.3Vdc |
| **Functionality** | Provides a steady output voltage no matter the inputs. |

|  |  |
| --- | --- |
| **Module** | Sensor system |
| **Inputs** | Heart rate sensor data, temperature sensor data, movement sensor data, 3.3Vdc |
| **Outputs** | Heart rate, temperature, and movement readings |
| **Functionality** | Collects sensor data for easy analysis by MCU. |

|  |  |
| --- | --- |
| **Module** | MCU |
| **Inputs** | Heart Rate, temperature, and movement readings, user configuration data, 3.3Vdc |
| **Outputs** | Analyzed sensor data, LEDs |
| **Functionality** | Analyzes sensor data, sends analyzed data to the communications system, and manages low power modes for the communications chip and sensors. |

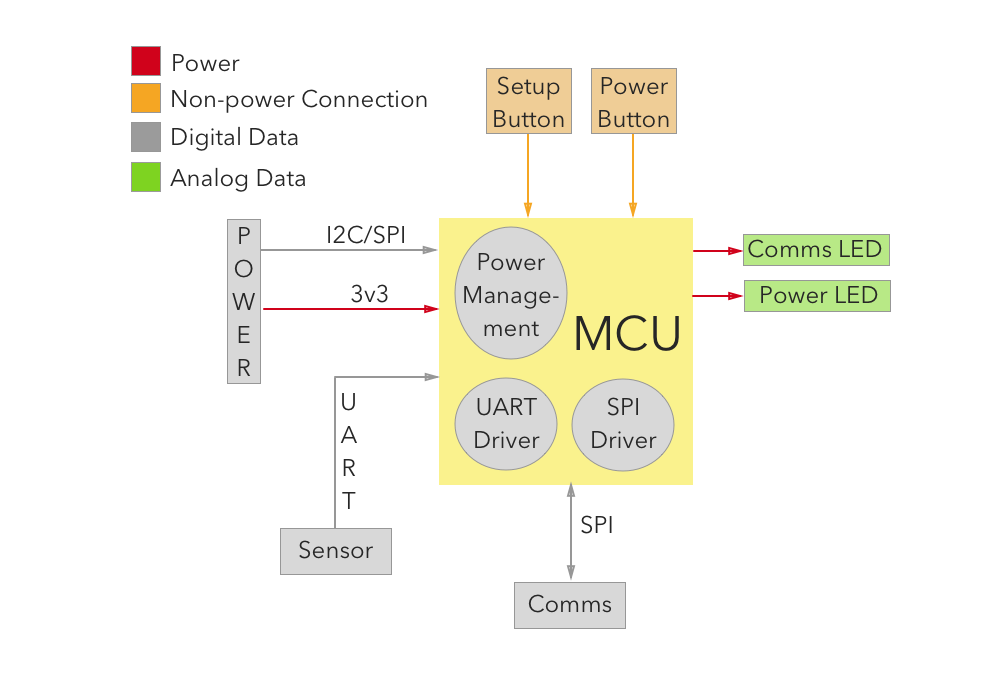
|  |  |
| --- | --- |
| **Module** | Communications |
| **Inputs** | Analyzed sensor data (device), user config data (base station) |
| **Outputs** | Analyzed sensor data (base station), user config data (device) |
| **Functionality** | Sends information between the device and the base station through high-frequency wireless communication. Data inputted should be outputted without modification. |

|  |  |
| --- | --- |
| **Module** | Base station |
| **Inputs** | Sensor data, User config data, USB power |
| **Outputs** | User config signal, Sensor data |
| **Functionality** | Interface between from the communications system and the computer. |

|  |  |
| --- | --- |
| **Module** | Computer |
| **Inputs** | User config, Sensor data |
| **Outputs** | Alerts, Visualization, User config data |
| **Functionality** | Gathers the sensor data and monitors for abnormalities. Will also visualize the data and allow the user to change configuration settings. |

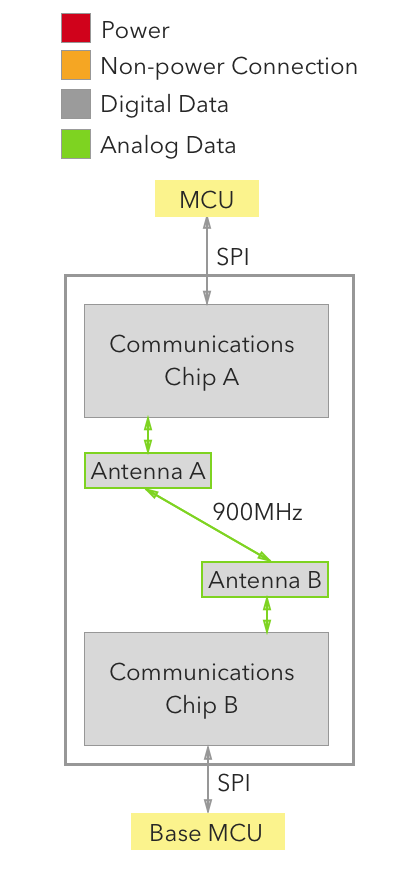
# L2 Functional Decomposition

## Main MCU



|  |  |
| --- | --- |
| **Component** | Main MCU |
| **Inputs** | Setup button, power button, UART from sensor MCU, 3v3 input from power circuit, I2C/SPI from fuel gauge, SPI from communications chip A |
| **Outputs** | Communications LED, power LED, SPI to communications chip A, |
| **Description** | This is the brains of our product. It will facilitate low power modes for our transceiver and sensor array, while also handling all data transmission and receiving. In a wake cycle, it should grab data from all of its inputs, wake the transceiver, send the data, and wait for a response from the base station. |

## Communications



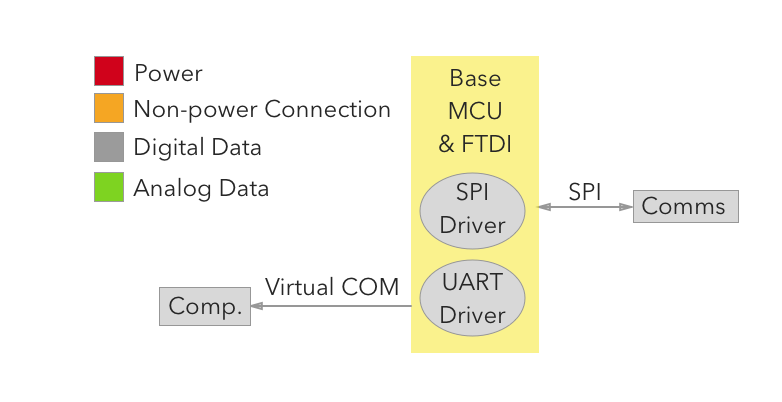
|  |  |
| --- | --- |
| **Component** | Communications Chip A |
| **Inputs** | SPI from main MCU, analog data from antenna |
| **Outputs** | SPI to main MCU, analog data to antenna |
| **Description** | The communications chip is responsible for taking digital data from the MCU and producing a high frequency communication signal that the antenna will transmit over our 900 MHz frequency band. This includes the bandwidth, transmission encryption, packetizing, and in some cases error correction. |

|  |  |
| --- | --- |
| **Component** | Communications Chip B |
| **Inputs** | SPI from base MCU, analog data from antenna |
| **Outputs** | SPI to base MCU, analog data to antenna |
| **Description** | Same as Chip A, this chip should facilitate wireless communication for the base station. |

|  |  |
| --- | --- |
| **Component** | Antenna A |
| **Inputs** | Analog data from Chip A, radio waves from Chip B |
| **Outputs** | Analog data to Chip A, radio waves to Chip B |
| **Description** | The antenna will do what an antenna does best, turn analog data into high powered radio waves. It will be tuned to work within the 902-928 MHz range. |

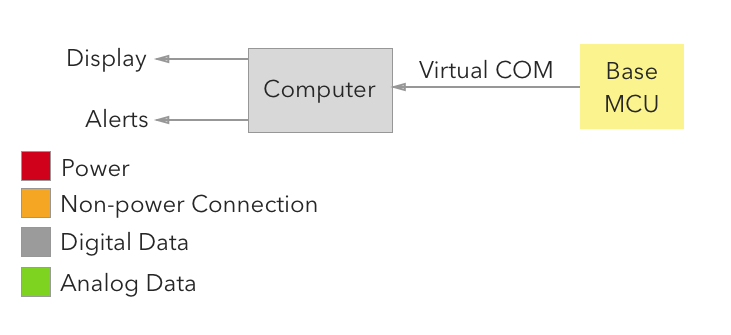
|  |  |
| --- | --- |
| **Component** | Antenna B |
| **Inputs** | Analog data from Chip B, radio waves to Chip A |
| **Outputs** | Analog data to Chip B, radio waves from Chip A |
| **Description** | Same as Antenna A |

## Base Station



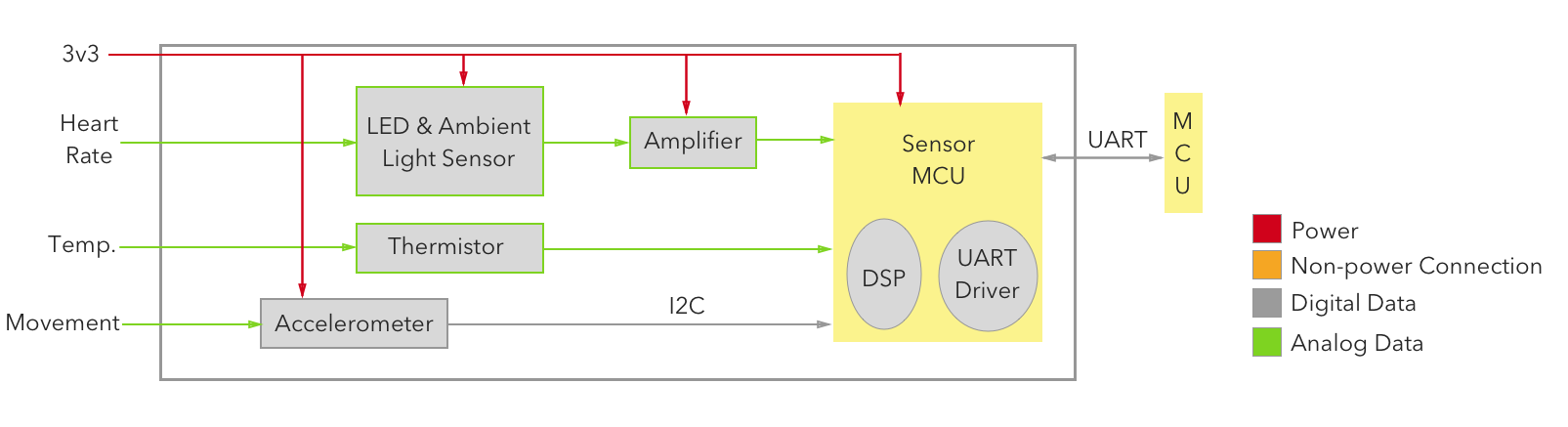
|  |  |
| --- | --- |
| **Component** | Base Station MCU |
| **Inputs** | SPI from Chip B, serial data from the Computer |
| **Outputs** | SPI to Chip B, serial data to the Computer |
| **Description** | The base station MCU facilitates taking digital signals from the communications chip and transmitting them over Serial/USB to the Computer for alerts and data visualization. It will also be able to send configuration data back to the main MCU via the communications system. |

## Computer



|  |  |
| --- | --- |
| **Component** | Computer |
| **Inputs** | Serial data from the base station MCU |
| **Outputs** | Serial data to the base station MCU, data visualizations, alerts |
| **Description** | The computer facilitates the data visualization & display, as well as the alerting system. It will also be able to change configuration settings by sending them back to the base station MCU |

## Sensor Array



|  |  |
| --- | --- |
| **Component** | LED & Ambient Light Sensor |
| **Inputs** | Heart rate. |
| **Outputs** | Analog signal |
| **Description** | This is the small system that will measure the blood flow in a user’s wrist and output an analog signal. |

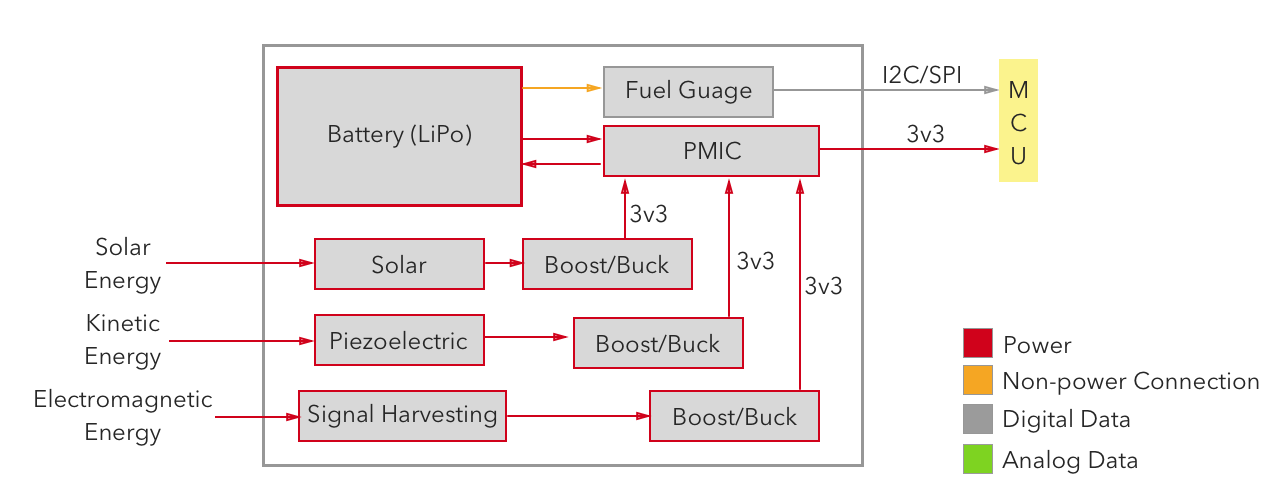
|  |  |
| --- | --- |
| **Component** | Amplifier |
| **Inputs** | Analog signal |
| **Outputs** | Amplified analog signal |
| **Description** | As the heart rate signal is very low in amplitude, we will need to amplify it in order to interpret it more accurately and reduce noise. |

|  |  |
| --- | --- |
| **Component** | Accelerometer |
| **Inputs** | Movement |
| **Outputs** | I2C/SPI data |
| **Description** | The accelerometer’s job is to retrieve acceleration data and send it on to the sensor MCU. |

|  |  |
| --- | --- |
| **Component** | Thermistor |
| **Inputs** | Temperature |
| **Outputs** | Analog signal |
| **Description** | The thermistor is here to give us a base reading of the person’s temperature. While it is not core body temperature, it should serve as a sufficient measurement for body temperature. |

|  |  |
| --- | --- |
| **Component** | Sensor MCU |
| **Inputs** | Analog signal from thermistor, analog signal from HR amplifier, digital data from accelerometer |
| **Outputs** | UART data to the main MCU |
| **Description** | The sensor MCU is a proposed way to reduce overall power consumption by using a higher powered MCU for main computations, paired with an extremely low power MCU for sampling data. As data sampling can be quick and doesn’t need much processing power, the MCU could quickly sample data and save it until it is requested (about every 30 seconds), opening up the main MCU for the more compute intensive tasks such as; toggling low power modes on the transceiver, managing the battery, and making sure the transceiver is on for as short of a time as possible. In essence, the sensor MCU acts as an analog to digital converter for our data, but goes a step further in computing things like heart rate to allow the main MCU to perform its tasks faster. |

## Power System



|  |  |
| --- | --- |
| **Component** | Solar, piezoelectric, and signal energy harvesting components |
| **Inputs** | Solar, kinetic, and electromagnetic energy |
| **Outputs** | Some power |
| **Description** | Energy sources have not been 100% decided on, and lots of research is being done and has to be done in this field. However, it is obvious through our huge requirements on battery length that some energy must be gained through outside sources, rather than just saving power via low-power modes or otherwise. |

|  |  |
| --- | --- |
| **Component** | Boost/Buck converters |
| **Inputs** | Power from energy sources |
| **Outputs** | 3.3V |
| **Description** | Often, charging circuits will need a very specific voltage in order to be 100% efficient. Adding a boost/buck converter in between the energy source and the charging circuit will allow us to reach peak efficiency with our charging circuit, something that will be much appreciated in the long run. |

|  |  |
| --- | --- |
| **Component** | Power Management IC (PMIC) |
| **Inputs** | Power from boost/buck converters, power from battery |
| **Outputs** | Charge to battery, power to device circuits |
| **Description** | Most batteries require some sort of circuit to provide management of battery charge. This is that circuit. |

|  |  |
| --- | --- |
| **Component** | Fuel Gauge |
| **Inputs** | Connection to battery |
| **Outputs** | I2C/SPI to main MCU |
| **Description** | A sponsor requirement is to be able to relay charge status to the base station. This circuit fills that requirement. |

|  |  |
| --- | --- |
| **Component** | Battery |
| **Inputs** | None |
| **Outputs** | Power, connection to fuel gauge |
| **Description** | We need a battery to store energy as we collect it and provide continuous power when no such energy exists. |