|  |
| --- |
|  |
| Final Project Report  Jackson Farley, Erik Leinen, Ellie Langley |
| |  |  |  | | --- | --- | --- | | Spring 2019 | 5/10/19 | ELC 4438 | |

# Project Summary

This device is implemented on a smart walker and determines whether the walker is being used and at what time. When not in use, the device gently reminders the user to use the walker, and when the gentle reminder fails, the harsh reminder is enacted. As more and more elderly use walking aids, and as the elderly population is more susceptible to injurious falls, it is difficult for an elderly person to achieve independence safely. This device will allow the elderly who require walking aids to live by themselves safely.

This device utilizes sensors to detect human presence on the upper and lower part of the walker. Depending on whether a human presence has been detected from the lower sensor, the upper sensor will then take a measurement. If the upper sensor has also taken a measurement with a proximal signal, then the device will output a “gentle reminder” by lighting the LED modules on the walker a solid yellow. If the walker has still not been used, then the device will output a “strong” reminder by flashing the LEDs red. When the buttons on the walker are pressed, the walker will go to an “in use state”. A time stamp will be taken using a clock module and output to the SD card module.

Most of the electrical parts for the smart walker have been placed within a plastic, red Sparkfun box, with holes drilled for wires which need to go outside the box. The two push buttons have been placed on the handlebars of the walker. The device is powered by 9V batteries.

# Project Objectives

1. Measure Human Presence: Use two sensors to detect the presence of the user effectively.
2. Time Accuracy: Maintain correct date and time for data logging.
3. Reminders: Send a reminder to the user if the walker has not been used but can sense the user’s presence.
4. Lightweight: The device will be lightweight as to not disrupt the balance of the smartwalker and will take up minimal space.

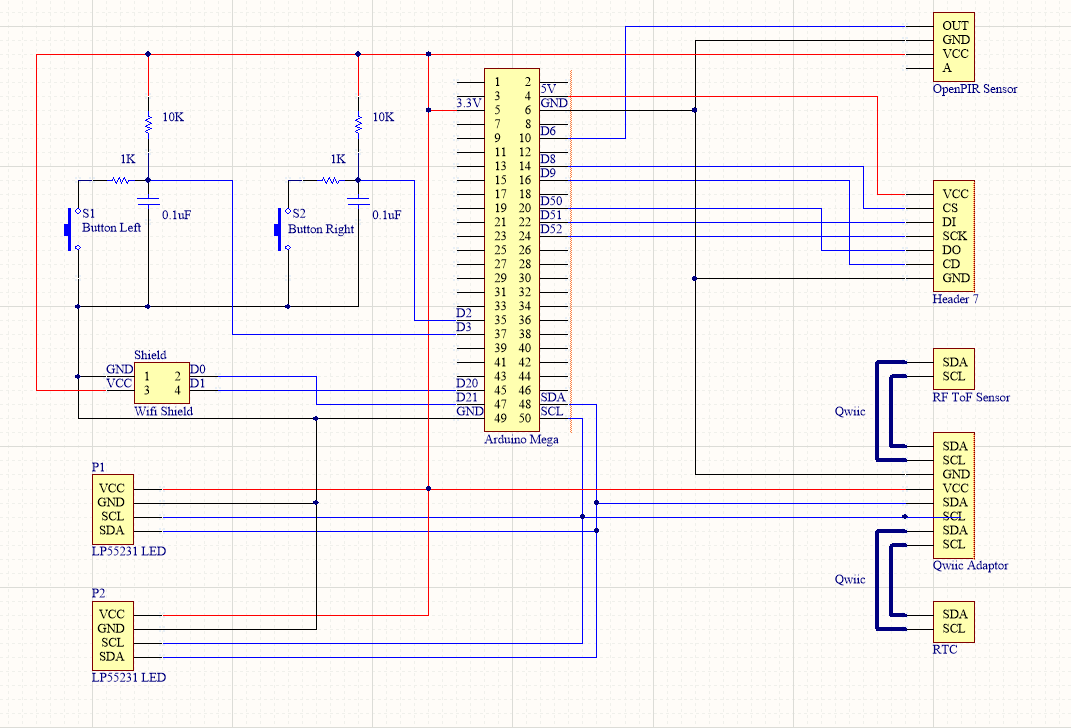
# Selected Hardware Platform

The Arduino Mega 2560 was selected as the hardware platform, as the sensors purchased for the project best interface with the Arduino IDE. The Arduino mega also has many digital pins that can be used as GPIOs, meaning that for the components which utilize GPIO pins there is plenty of space.

The Arduino Mega also has pins that can be configured as interrupts. This is important because the buttons implemented in this project must be configured as interrupts for the purpose of the device to be achieved.

Additionally, the Arduino Mega has SCL and SDA lines to allow for ease of I2C usage, as many components for this project use I2C capabilities.

# Hardware Design Description



Shifting Micro SD

Figure 1: Final Schematic for Smartwalker

The microcontroller used was the Arduino Mega 2560, and the Arduino IDE was used to implement the code onto the board. An Arduino Mega Protoshield was used to make the middle connections needed to reach the same circuit as shown in Figure 1. The proto board had the resistors and capacitors soldered onto it so that a hardware debounce was created for the buttons to give the most accurate readings of the button press. Above the protoshield, was another Wi-Fi shield that will be used in future developments of this product to help implement the ability to store data onto a website that can be accessed by another device. The Arduino is powered by a 9V battery, but in the future, it is desired that a rechargeable battery substitute should be used.

The LED modules used for the display of the states use 3.3V logic in order to operate and have four different pins. The voltage in, the ground, the SCL, and the SDA pins. They are powered by the 3.3V pin on the Arduino Mega, and connected to the SCL and SDA pins as well. The next connection is the Qwiic adapter module. This was used to connect multiple Qwiic compatible devices together without having to connect separate wires from the SDA and SCL pins. This reduced wires and made packaging easier. All Qwiic components use 3.3V logic and are therefore connected to the 3.3V pin on the Arduino. Connected to the Qwiic adaptor is the RF Time of Flight Sensor, and the Real Time Clock. Both devices were connected through the Qwiic wiring, and only have those connections used. The Real Time clock has a small battery that charges when power is applied to the board, and expends power from the battery when there is no power in. This allows the module to keep track of the current time even if the power is removed. The next module used was the Shifting Micro SD Breakout, which was used to write the data onto a micro SD card. For the Arduino Mega, the SCK, DI, and D0 pins must be connected exactly to digital pins 52, 51, and 50 respectively. The CS pin and CD pin can be connected to any other digital pin, but it was decided to use digital pins 8 and 9 for this project. This module is the only one that requires 5V logic and is therefore connected to the 5V pin to the Arduino. The last module is the Open PIR Sensor. This uses 3.3V logic and outputs a digital and analog signal, but only the digital output is needed for this project.

One other module that would be nice to include is a speaker, that would output an audible noise for different states to grab the attention of the user. This was not implemented due to time constraints but is a goal of the project sponsor.

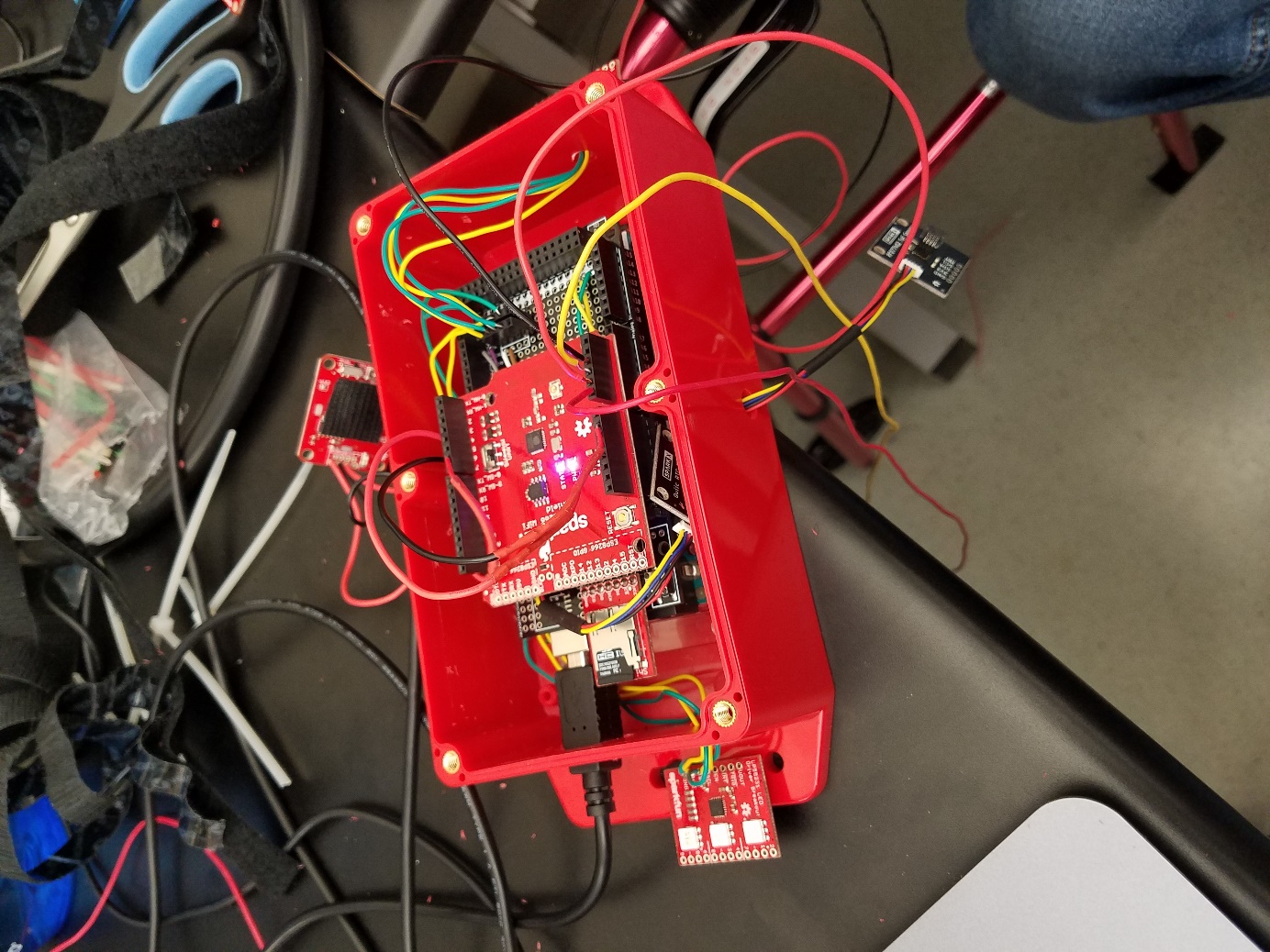


Figure 2: Hardware Implementation

1. **Arduino Mega 2650**

* Multiple digital pins, etc.

1. **Sensor 1**—Sparkfun Distance Sensor Breakout – RFD77402 (Qwiic)

* Detects human presence up to 2 meters with I2C

1. **Sensor 2**—Sparkfun OpenPIR Sensor

* Measures for human presence and returns a 1 or 0 depending on whether human presence is detected

1. **LEDs**—Sparkfun LP55231 LED Driver Breakout

* Interfaces with I2C and three RGB LEDs for different colors and flashing

1. **GPIO Buttons**—Momentary push button

* Simple GPIO buttons, values can be read by Arduino mega
* Debouncing methods required

1. **SD Card Module**—Sparkfun Level Shifting microSD Breakout

* Allows for mass storage of data on the Arduino Mega

1. **Clock**—Real Time Clock (Qwiic) – RV-1805

* Maintains accurate date and time for logging

1. **Adapter for I2C Sensors**—Sparkfun Qwiic Adapter
   * For easy use of multiple I2C Qwiic sensors on the Arduino mega

**Chosen software platform**

**Software Design**

***Overall***