Spring 2021

Crosswalk Sensor System

Jacob Schandel
jhs51@zips.uakron.edu

Mohammed Alsubhi
mma138@zips.uakron.edu

Aaron Gloeckler
amg281@zips.uakron.edu

Samuel Turner
set48@zips.uakron.edu

Crosswalk Sensor System

Senior Design Project Final Report

Design Team 16

Mohammed Alsubhi

Aaron Gloeckler

Jacob Schandel

Samuel Turner

Dr. Yilmaz Sozer

April 23, 2021

i. Table of Contents

ii. List of Figures

iii. List of Tables

Abstract

A number of safety concerns persist with modern crosswalks. People are distracted by things in their car, or their surroundings can make it hard to see oncoming pedestrians. With today's self-sufficient power technologies, available sensors, and cheap low-power wireless communication protocols, it is possible to create digital signage to warn of oncoming pedestrians. By using distance sensors such as ultrasonic or infrared sensors, systems can detect where objects, or in this case people, are with respect to a sensor on a fixed pole. With low-powered illumination technologies like LEDs and low-powered IoT communication protocols like ZigBee, a system can communicate to other hardware and human drivers without needing a great deal of power to do so in a world where there is an increasing emphasis on being eco-friendly. Additionally, by using energy harvesting equipment such as solar panels, it is possible to power such a system without having to connect it to an external supply of power requiring fossil fuels or other costs associated with working with an energy provider. Therefore, such a system is feasible and could save people from extraordinary pain as they do the ordinary task of crossing the street.

Key features of this system include…

- Wireless communication between 2 sign posts at either end of a crosswalk
- LED signage to indicate an oncoming pedestrian
- Sensors to detect pedestrians
- Self-sufficiency through energy harvesting and storage via solar panels and batteries

1. Problem Statement

1.1. Need (MA, AG, JS, ST)

There is a need for safer crosswalks in our communities. As of right now, visual impairments can make it less safe for pedestrians to cross the road. It is quite important to provide such a signal for drivers in low-visibility conditions such as snow, especially at the rate which pedestrian-involved crashes occur. Per NPR, "Across the U.S., 6,227 pedestrians died in traffic accidents in 2018, the highest number in nearly 30 years [9]."

1.2. Objective (MA, AG, JS, ST)

The proposed solution is to utilize a sensor safety system to detect when a pedestrian is near a crosswalk and alert nearby drivers of the pedestrian's presence. The system would detect if people are located at a crosswalk or crossing the road and trigger an external warning to oncoming drivers. This warning will come as an LED illuminated sign, indicating that a pedestrian is in the road. The system would utilize 2 sensors per pole, where each pole is located on each end of the crosswalk. These 4 sensors consist of 2 ultrasonic and 2 PIR distance sensors which will cover a minimum of 20-foot-long crosswalk. The lumen output will be optimized depending on current daylight levels. Finally, the sensor poles will be solar powered with a battery pack in order to be wirelessly energy self-sufficient.

1.3. Background (MA, AG, JS, ST)

(AG) The need for a pedestrian safety crosswalk system is clear. Statistics every year point to distracted drivers as the number one cause of pedestrian injuries and fatalities. The obvious number one distraction to drivers in today's world is their phones. *Inside Science* states, "there were 2,300% increase in crash risk in a driver who's texting versus just driving alone." [8] When drivers are distracted by their phone, they will not see pedestrians in the crosswalk. This is a very critical problem to address because countless lives are being lost every year due to a problem that can be solved. The National Highway Traffic Safety Administration specifically says, “40% of all American teens say they have been in a car when the driver used a cell phone in a way that put people in danger.” [4] These numbers are far too high to go unnoticed, especially when pedestrian safety is concerned. There has to be a system that helps pedestrians cross the road safely.

The goal of this senior design project is to build a crosswalk safety system that will alert drivers when a pedestrian in the crosswalk from either side of the crosswalk. This will be done by having the people crossing into a designated crosswalk step into a sensor-monitored region which sends a signal to a flashing strobe that intermittently flashes to warn drivers that there is someone crossing. The strobe will be bright enough to alert the driver by catching their attention, as well as include a sign around the strobe which warns of a pedestrian in the road. The system may also have a time limit to how long the strobe will flash so it will not block traffic. The sensor will also need to be made for a minimum amount of time so that it will not be falsely activated due to the weather or other instances. This crosswalk system will include a wireless design which will be powered through solar power and be implemented on both sides of the road. This compact design will allow the closed system to be easily installed at any crosswalk. With this system in place pedestrians will have the peace of mind when crossing the street. All incoming drivers will be alerted with bright lights on a sign warning them to slow down, which will protect the people in the crosswalk.

(ST) The basic theory of the idea is to help pedestrians of all kinds to be able to cross the street safely and help drivers recognize in advance of pedestrians crossing the street. According to Fitzpatrick, Kay (Safety Effectiveness of the Hawk) The Federal Highway Administration

(FHWA) pedestrian safety research program goal was to increase pedestrian safety and mobility from safer crosswalks and sidewalks. The city of Tucson, AZ developed the high intensity activated crosswalk (HAWK) pedestrian crossing beacon in the 1990's to help assist with pedestrian walking across the street. [7] The Hawk's purpose was to stop vehicles to allow pedestrians to cross the roadway safely and alert drivers to proceed forward after the lights have stopped blinking and the pedestrians have made it across. They tested the theory at 21 minor intersections to start out to see how many accidents and/or pedestrian crashes were going to happen within the trial period both at day and at night. The results were okay as the hawk did reduce the pedestrian accidents by 10%, but it was almost like the statistics as if the Hawk was never there to begin with. There were other ways that could have improved the Hawk significantly in numbers. Using flashing lights that are not blinding whether it be day or night that won't cause seizures to alert current drivers of today that pedestrians are crossing so they won't be distracted by other obligations. Having some electronic sign up with the flashing lights saying pedestrians is crossing an extra safety precaution to have with current drivers. Another idea to add to the cross walk could have been putting something on the crosswalk path to make it more noticeable whether it be day or night. This experiment was done by McLean, VA: U.S. Federal Highway Administration, Research; they tried the method of changing the pedestrian crosswalk lines in the road from transverse lines to either bar lines or continental lines. The concept and idea seemed great and if successful this would change the markings of the pedestrian crosswalk from state to state for whichever one is more successful in decreasing pedestrian accidents. Of course, the location matters as well as far as where the markings are being placed, the type of car, driver speed etc. The results for the experiment showed that using bar/continental lines were impactful compared to using transverse lines in the pedestrian crosswalk. The drivers were able to see two different types of line from a farther distance to realize they were approaching a crosswalk with no signals to alarm them. This is a similar idea to use when making a device for pedestrian crosswalk, but also suggesting using a certain paint material that makes it more noticeable or shine brighter so it could be seen more. The concept is similar as far as to get drivers' attention or notice from a far distance that they are approaching a crosswalk using bigger marks on the ground and potentially using flashing lights on a sign to alert current drivers as they tend to text and drive.

(JS) A number of solutions have been designed to overcome the challenges associated with pedestrian safety, whether they are the fault of a pedestrian or the fault of a motor vehicle operator. One of these solutions involves the utilization of signals coming from pedestrians' handheld devices, such as a smartphone, in order to trigger the transition of traffic signals commanding the flow of both pedestrian and vehicular traffic. [3] With regards to this solution, this ties back to a central controller than controls multiple traffic signals for multiple intersections in an effort to help manage traffic across a number of areas at once to maximize traffic efficiency. Such a system also allows for the notification of an impaired pedestrian that they are walking into a potentially dangerous situation via the device's user interface. Another solution involves the utilization of a sensor at the end of a crosswalk, rather than a signal from a pedestrian's handheld device, as mounted on a pole to determine when a pedestrian is ready to cross. [6] These sensors then allow for the transition of traffic signals for vehicles and pedestrians, much like the previous solution did.

(MA) Both of these designs have their limitations. When it comes to designing crosswalks systems, some certain limitations become apparent. Those limitations include the light level, crosswalk marking pattern, and street characteristics. Such limitations impose a challenge for designers as they could affect the driver's ability of detecting a crosswalk, which could potentially affect the overall success of the system. A study conducted by the Federal Highway Administration comparing the effects of different crosswalk marking patterns showed that "The detection distances to continental and bar pairs are statistically similar." [2] However, the detection distance to transverse lines was shorter than continental and bar pairs. While at the same time, the light level had an effect on the detection distances of each marking pattern. Another limiting factor the study was concerned about was the street characteristics, which includes traffic, speed limits, and street types. As the study found that higher speed limits and the existence of traffic could decrease the detection distance of a crosswalk. Also, "marking type had a significant effect on detection distance during the daytime at midblock crosswalks and at nighttime", the detection distance for midblock streets were significantly longer for the continental marking pattern than the transverse marking pattern during nighttime. [2] However, the difference was negligible for intersection street type. Characteristics such as age and gender were found to be insignificant in detecting a crosswalk. However, higher sight level and vehicles, i.e. SUVs and trucks made it easier for the drivers to detect a crosswalk from a longer distance than those with lower sight level and smaller vehicles.

Besides improving pedestrian safety, another benefit of implementing a crosswalk system is reducing vehicle crashes. According to a study conducted by the Federal Highway Administration, "A 29 percent reduction in total crashes, which is statistically significant at the 95 percent confidence level. A 69 percent reduction in pedestrian crashes, which is statistically significant at the 95 percent confidence level. A 15 percent reduction in severe crashes, which is not statistically significant at the 95 percent confidence level," which proves the effectiveness of the crosswalks systems at improving the safety of pedestrians and drivers. [7]

(JS) In conclusion, research shows a dire need for higher safety for the modern pedestrian. As both drivers and pedestrians are faced with a slew of new distractions as they travel, there is a need for a solution that saves lives and prevents injuries. When this is paired with current street conditions, it is apparent that a new solution is needed to protect pedestrians and motorists alike. This is a problem which shall be resolved with this project.

1.4 Marketing Requirements (ST, AG, JS)

1. The system provides visible warning indicators for drivers approaching the crosswalk.
2. The system senses pedestrian locations near and in the roadway.
3. The system can be implemented at any two-lane road.
4. The system optimizes power depending on daylight conditions.
5. The system should be energy self-sufficient.

2. Engineering Analysis (AG, ST, JS, MA)

2.1. Circuits (AG, MA, ST)

A power regulator circuit is used to adjust the input solar power to a chargeable voltage source for the battery. This voltage regulator is a crucial component to keep the 12V battery charged, but not overcharged. The circuit steps down the voltage from the solar panel in order to keep a constant voltage input of 14.6V-14.8V to charge the battery. This input voltage range is chosen for a 12V sealed lead acid battery because it allows the cells to charge quickly and efficiently without getting overcharged. The 12V solar panel has a max voltage output of 22V on the sunniest days, so this power regulator keeps the battery safe. The adjustable circuit utilizes the LM317 [20] voltage regulator component which allows for the resistors to be interchanged in order to get the desired output voltage. To keep a constant 14.6V-14.8V input into the 12V battery, the resistors need to be close to a 10-1 difference. 5k and 470 Ohm resistor combinations were chosen in order to keep a consistent output into the battery from the 16V-22V output range at the solar panel.

The sensor chosen to cover the long distance of the crosswalk is an PIR sensor: PIR (motion) Sensor [10]. The PIR sensor itself has two slots in it, each slot is made of a special material that is sensitive to IR. The lens used here is not really doing much and so we see that the two slots can 'see' out past some distance (basically the sensitivity of the sensor). When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors. When a warm body like a human or animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected. Each sensor will reach 20ft, per pole, at a height of 3ft up on the pole. A block diagram of the distance measuring IC can be seen in Figure 7 attached.

The ultrasonic sensor chosen to cover the area surrounding the side of the crosswalk and edge of the road is the LV-MaxSonar-EZ0 MB1000 [11]. This sensor will work by bouncing waves off pedestrians and receiving those waves back. The sensor will output an 1ft-6ft cone of coverage which is ideal for the sidewalk and crosswalk itself. This cone output can be seen in figure 9 attached below. The circuit will output a signal and depending on the distance of the pedestrians, receive back an input as the reflected signal. The difference between the time will tell if there is a human in the area in or near the crosswalk, and how far away they are. When either sensor detects a pedestrian, a signal will be sent to enable the warning LEDs.

The LM555-astable mode circuit, figure 6, was built in order to generate a continuous square DC pulse that will flash the LEDs that are attached to the crosswalk sign. The output-high and the output-low times are determined by the charging and discharging times of the external capacitor (C2), respectively. Those times are governed by the values of (R1+R2) and (R2), respectively. The period of the output signal can be calculated by the addition of the charging and discharging times, the frequency is calculated as the inverse of the period of the output signal, and the duty ratio is determined by the values of (R1) and (R2) [15]. A current-limiting resistor was implemented to avoid exceeding the rated current of the LEDs due to variations in supply voltage. The power dissipation range of the LM-astable mode circuit, including four LED strips connected in a parallel combination is as follows:

- LM555: 613 ~ 1180 mW [15]
- Four parallel LED strips, each consisting of three parallel LEDs and supplied by 3 V and 30 mA, at 0.5 duty ratio: 180 mW.
- Overall power consumption range: 793 ~ 1370 mW.

The parameters for the output signal were chosen so that the signal's duty ratio is approximately 50%, with the output-high and low times being equal to 1 second. The following equations were used to determine the values of R1, R2, and C2:

$$Duty\ Ratio = \frac{R2}{R1 + 2(R2)}$$

$$t_{High} = 0.693(R1 + R2)C2\ seconds$$

$$t_{Low} = 0.693(R2)(C2)\ seconds$$

$$Period = t_{High} + t_{Low} = 0.693[R1 + 2(R2)](C2)\ seconds$$

$$Frequency = \frac{1}{Period}\ Hz$$

For $t_{High} = t_{Low} = 1\ second$ and $Duty\ Ratio = 0.5$:

è R1 = 1k W, R2 = 144k W, C2 = 10 mF

It should be noted that different combinations of R1, R2 and C2 values are possible for the same design parameter and can be chosen based on availability.

In order to control the operation, and modify the trigger level, of the LM555-astable mode circuit, an open-collector circuit, figure 6, using a 2N2222 transistor, is integrated so that the crosswalk sign can be turned on/off based on pedestrians detection and to override the default triggering level of the LM555 timer, the latter will ensure compatibility between the LM555 timer and PIC24FJ128GA010 microcontroller. The base resistance can be calculated by subtracting the supply voltage from the base-emitter voltage then dividing by the base current. Resistance values of 100 ~ 1k W were found to be operational.

2.2. Electronics (AG, MA, ST)

The LM317 is a 3-terminal adjustable positive-voltage regulator. It can supply more than 1.5A, over an input voltage range of 1.25V- 37V. The LM317 [20] has is adjustable by changing the resistances between its pins. To keep a constant 14.6V-14.8V, an approximate 10-1 ratio between R1 and R2 must be used. This allows the output of the power regulator circuit to supply a constant charge to the battery without overcharging and damaging it. An LTspice of the circuit can be seen in figure 14.

The PIR (motion) Sensor [10] is the chosen PIR sensor for the crosswalk safety system. This PIR sensor consists of a position sensitive detector using pyroelectric material. This allows for the distance of the pedestrian to be an output of a specific voltage level, however in the case of this system, the output will be in digital such as a proximity sensor.

The LV-MaxSonar-EZ0 MB1000 [11] is the chosen ultrasonic sensor for the sensor system. This sensor is critical to the sensor safety system. They allow for a cone of an area, off the sides and in a crosswalk, to be analyzed. Ultrasonic sensors measure the time of flight for sound that has been transmitted to and reflected from nearby objects. Based upon the time of flight, the sensor can output a voltage and tell how far away a human is. Any detection at all will allow for the system to read if a pedestrian is in the crosswalk and turn on the LED sign.

The LM555 is “a highly stable device for generating accurate time delays or oscillation. Additional terminals are provided for triggering or resetting if desired. In the time delay mode of operation, the time is precisely controlled by one external resistor and capacitor. For a stable operation as an oscillator, the free running frequency and duty cycle are accurately controlled with two external resistors and one capacitor. The circuit may be triggered and reset on falling waveforms, and drive TTL circuits” [15]. The LM555 can source/sink up to 200 mA and requires a small supply current, typically 10 mA at 15V supply, in order to operate. Supply voltage can vary between 4.5 ~ 16 V. [15]

2.3. Signal Processing (AG)

Signal processing consists of communication in terms of the various sensors sending information of human detection signals. When a human breaks the plane of the ultrasonic or infrared detection, the sensors will output a voltage level which the Explorer 16/32 [13] will take as an input and send a signal to the LEDs to turn on. This form of signal processing takes a human movement as input and outputs the LEDs on, to warn any oncoming drivers of a pedestrian.

2.4. Computer Networks and Communications (JS, ST)

A wide variety of wireless protocols exist for connecting devices to share data. While Wi-Fi is popular, it uses a large amount of power [18] compared to other communication protocols which exist. Bluetooth and ZigBee are known for being device communication protocols. Additionally, Bluetooth is commonly used for many applications such as wearable devices and audio [16] headsets. These devices are common for those walking down the street. Because it is designed [17] specifically for automation and utilizes a self-forming mesh network, ZigBee appears to be the best option. It also works with 300 meters of range [19] on average utilizing line-of-sight, and it can operate under low power, which makes it perfect for an outdoor sign running on a solar panel. Therefore, ZigBee appears to be the best communication option.

Using the Explorer 16/32 board [13] and PIC24FJ128GA010 [12] with the C programming language is needed in order to communicate with the sensors. Conditions can be set for the ranges of the sensors to do different operations if they fall within the range. For example, if the sensors have a minimum range of 2ft and a maximum range of 10ft, the code can be set to “do something” when a pedestrian is detected and within range that is inputted to the user's desire.

In order to communicate this information over ZigBee, we must use a ZigBee module like the XBee Pro S3B. The XBee uses UART serial communication to receive and transmit commands to the parent device. The XBee device [21] has two general modes of operation: a so-called transparent mode and an API mode. In API mode, the parent processor (in our case, the PIC24F) issues hexadecimal commands to the XBee over UART instead of just raw data. This allows for things such as addressing to specific devices in a mesh network. In transparent mode, the XBee modules behave almost like an invisible UART cable, where the TX and RX signals of the UART are sent wirelessly. In this mode, you simply send raw data over the UART connection. Since only two signs are being used in this system, while either implementation could be used, the transparent implementation should be sufficient for these purposes.

Regardless of which implementation is used, the free XCTU [22] software provided by Digi, the manufacturer of the XBee module, is used for initial out-of-the-box configuration of the XBee units. This software allows for not only the configuration of the XBee modules via a

USB-based programming tool, but also for the generation of sample hex commands for use with the XBee's API mode of operation. One major advantage of using transparent mode is, as is shown below, using MPLAB's Code Configurator tools in the MPLAB X IDE, all that is needed to transmit strings of information is a simple printf() statement, which can be remapped to transmit over the processor's UART quite easily. Also, it is simple to read blocks of data via the functionality set up via MPLAB Code Configurator's APIs.

```
#include "mcc_generated_files/system.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>


/*
                         Main application
 */
int main(void)
{
    // initialize the device
    SYSTEM_Initialize();

    while (1)
    {
        printf("Jake has it working!  Yay!\n");
    }

    return 1;
}
```

Figure 1: Sample Code for Printing Text Over UART Using System's Print Functionality

This communications code can be reapplied into simple-to-use functions for sending and receiving data. In the project, we used the following code. However, this code can be altered to monitor for specific characters to allow for multiple message types to be sent and received.

```
/**
 * @brief Sends a flag over UART wireless ZigBee communication
 * @param flag value specified in H file
 */
void sendFlag()
```

```
{
    int i = 0;
    for (i = 0; i < 1000; ++i)
        printf("DDDDDDDDDDD");
}

uint8_t readBuffer[20];

/**
 * @brief Read sent flag
 * @return flag value
 */
char receiveFlag(void)
{
    memset(readBuffer, 0x00, sizeof(readBuffer));

    UART2_ReadBuffer(readBuffer, 20);

    int i = 0;
    int count = 0;

    while (i < sizeof(readBuffer))
    {
        if (readBuffer[i] != '\0')
            return 'D';

        i++;
    }

    return 0;
}
```

Figure 2: Code to Send and Receive Flag Codes

2.5. Embedded System (ST)

The system will utilize a PIC24FJ128GA010 [12] microcontroller to control the signage. The PIC24FJ128GA010 is a 16-bit processor which can easily be programmed in C, has a selection of GPIO for connecting to sensors and LEDs, and supports a number of communication protocols such as UART and I2C which can be applied for use with a wireless communication antenna such as a Bluetooth or ZigBee antenna. Given the power consumption using the 5V rail of the explorer 16/32 board [13], it needs a minimum of 1 watt/hr.

3. Engineering Requirements Specification (AG, ST, MA, JS)

Based on the predefined marketing requirements, the following engineering requirements were derived.

Marketing Requirement	Engineering Requirement	Justification
2, 3	• The ZigBee communication must be able to communicate with each other in order to relay where the pedestrian is coming from and what to do.	This process will ensure accuracy and determine which steps are to be performed next. Additionally, this standard meets the required transmission distance and power consumption for this application.
2, 3	• PIR Distance sensors must cover a minimum of 20 feet, between both poles, for pedestrian detection.	The indicated coverage area is essential for human detection in a typical crosswalk. It will also clarify if a pedestrian is approaching from a sidewalk or a street
1, 3	• The sensor pole system must communicate wirelessly over the roadway to output the same LED signals, at the same time	This will alert drivers approaching from both ways of a road to ensure pedestrians safety.
5	• The pole will be solar powered with a battery pack so that the system will always be wirelessly powered.	This renewable energy source will be environmentally safer compared to other nonrenewable sources. The battery pack will ensure operation during nighttime or days with inadequate sunlight.
1, 4	• Depending on the light levels, the LEDs will optimize power and lumen output of the sign accordingly: LEDs	This will ensure clear visibility of warning message during different light levels. Additionally, this makes the system more efficient by limiting

	brighter during day, and dimmer at night.	the supplied power during times when less illumination is needed.
5	• Charge Controller needed to control the energy that is converted into power from the solar panel to the battery	The Charge Controller will prevent under/overcharge of the battery to maintain safe operations for the system
1, 2	• System will display LED lights to warn drivers for up to 10 seconds after a pedestrian leaves the zone.	This safety step will increase pedestrian's safety.
1	• The size of the LED crosswalk sign should be 12 x 18 in.	The size has been chosen to ensure clear visibility for approaching drivers within a an appropriate distance.
1-5	• Each pole must contain the same human detection capabilities.	Having a pole on each side of the street will allow the system to detect pedestrians approaching from either side. Additionally, the sensors on the second pole will confirm a triggered signal on the other side and ensure correct operation if one side fails.
1. The system provides visible warning indicators for drivers approaching the crosswalk. 2. The system senses pedestrian locations near and in the roadway. 3. The system can be implemented at any two-lane road. 4. The system optimizes power depending on daylight conditions. 5. The system should be energy self-sufficient.		

4. Engineering Standards Specification (JS, ST, MA)

Based on the needs of the project and the chosen hardware, the following standards were selected for use for this project.

Category of Standards	**Standard**	**Use**
Safety	ORC 4511.46 [14]	Yield to pedestrians in crosswalks and stop well back from the crosswalk to give other vehicles an opportunity to see the crossing pedestrians so they can stop too.
	ORC 4511.46 [14]	Never pass vehicles stopped at a crosswalk. There may be people crossing that you can't see.
	ORC 4511.46 [14]	Slow down and be prepared to stop when turning or otherwise entering a crosswalk.
Regulation	2A.08 [23]	All warning signs should be retroreflective to insure clear visibility
Communications	ZigBee	Sign units across the road from one another will communicate wirelessly.
Programming Languages	C	The C language will be used to program the PIC24FJ128GA010, whose instruction set architecture is based on the C language.

Table 1: Engineering Standards Specification

5. Accepted Technical Design (MA, AG, JS, ST)

5.1. Hardware Design (MA, AG, JS, ST)

Level 0 Block Diagram of Hardware System (JS, ST):

Given the system, this will have an input of some type of power coming the battery that will charge the sensors and LED's. The output will have some illuminating lights that will flash to get driver attention and will vary the lighting from light today, so that this doesn't blind the driver and this will save the use of unnecessary power.

Figure 3: Level 0 Hardware Block Diagram

Functional Requirements Table (AG, JS, ST):

Module	Crosswalk Signal
Designers	• Mohammed Alsubhi • Aaron Gloeckler • Jacob Schandel • Samuel Turner
Inputs	• Solar Power • Ultrasonic Sensor Detection • PIR Distance Sensor Detection
Outputs	• Flashing Strobe/Warning signal • LED sign illuminates • Wireless trigger to a paired signal unit with equivalent capabilities

Description	• Human detection system used for pedestrians, around and inside a crosswalk, to warn oncoming drivers from both sides of their presence through an LED sign

Table 2: Functional Requirements Table for Level 0 Hardware Block Diagram

Level 1 Hardware Diagram (ST)

The system will require a 20W battery that will be able to supply each individual sensor and once triggered it to sends a signal to the processor to light the LED warning lights. The solar panel will make the system a self-efficient, so it doesn't require to change the battery. The 12V solar panel has a max voltage output of 22V in the best conditions on the sunniest of days. That voltage coming from the solar panel varies, depending on whether it is cloudy or sunny and would overcharge or undercharge the battery, so a charge controller is required in order to regulate the voltage coming from the solar panel to the battery. The adjustable circuit utilizes the LM317 [20] voltage regulator component which allows for the resistors to be interchanged in order to get the desired output voltage. To keep a constant 13V-14V input into the 12V battery. The system will utilize a PIC24FJ128GA010 [12] microcontroller to control the signage. The PIC24FJ128GA010 is a 16-bit processor which can easily be programmed in C, has a selection of GPIO for connecting to sensors and LEDs, and supports a number of communication protocols such as UART and I2C which can be applied for use with a wireless communication antenna such as a Bluetooth or ZigBee antenna. This processor is used with explorer 16/32 board [13] in order to communicate with LV-MaxSonar-EZ0 MB1000 [11] and PIR Sensor: PIR (motion) Sensor [10] using the Analog-to-Digital convertor to measure the distance of the pedestrians in the crosswalk and the curb of the crosswalk. The PIC24FJ128GA010 can be coded to set the sensors to maximum and minimum range of detection and if detected within that range, it will communicate a signal to the Zigbee to communicate to the other Zigbee device on the other side of the crosswalk that a pedestrian is within range of the crosswalk and needs to cross. This will trigger the LED warning lights to flash to alert a driver that pedestrians are crossing the crosswalk and they need to slow down, which is required by law for safety measurements ORC 4511.46 [14].

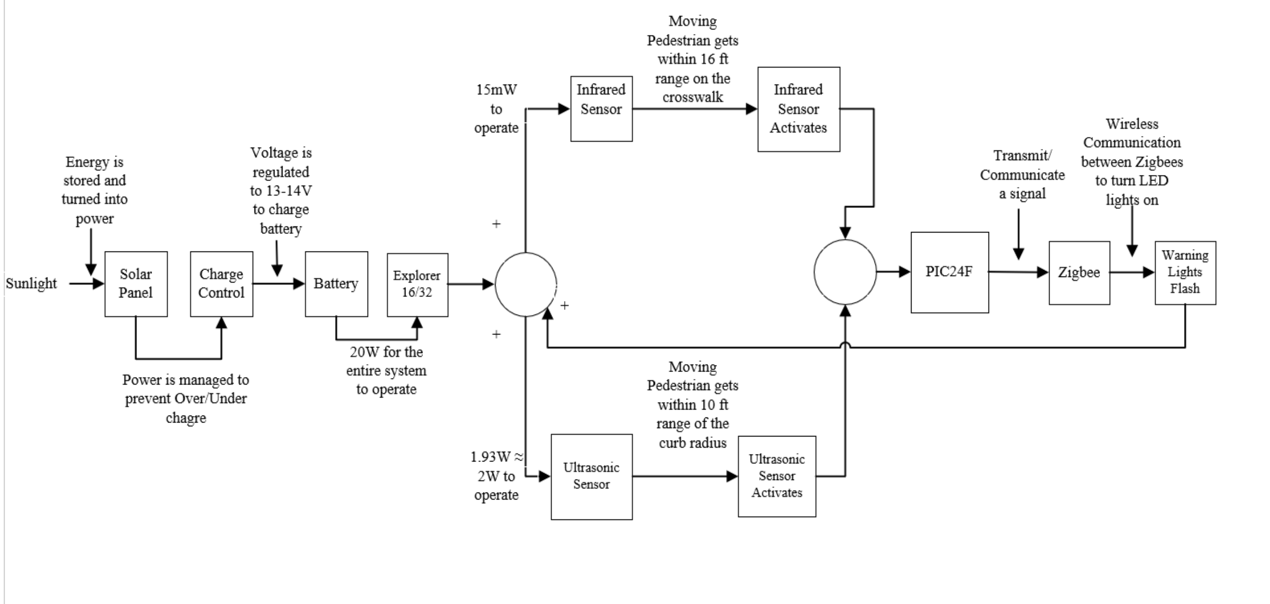

Figure 4: Level 1 Hardware Diagram

Module	Crosswalk Signal
Designers	• Mohammed Alsubhi • Aaron Gloeckler • Jacob Schandel • Samuel Turner
Inputs	• Solar Power • Ultrasonic Sensor Detection • PIR Sensor Detection
Outputs	• Flashing Strobe/Warning signal • LED sign illuminates • Wireless trigger to a paired signal unit with equivalent capabilities
Description	• Human detection system used for pedestrians, around and inside a crosswalk, to warn oncoming drivers from both sides of their presence through an LED sign

Table 3: Functional Requirements Table for Level 1 Hardware Diagram

Power/Voltage Regulator Circuit LM317 [20]

LTspice Circuit:

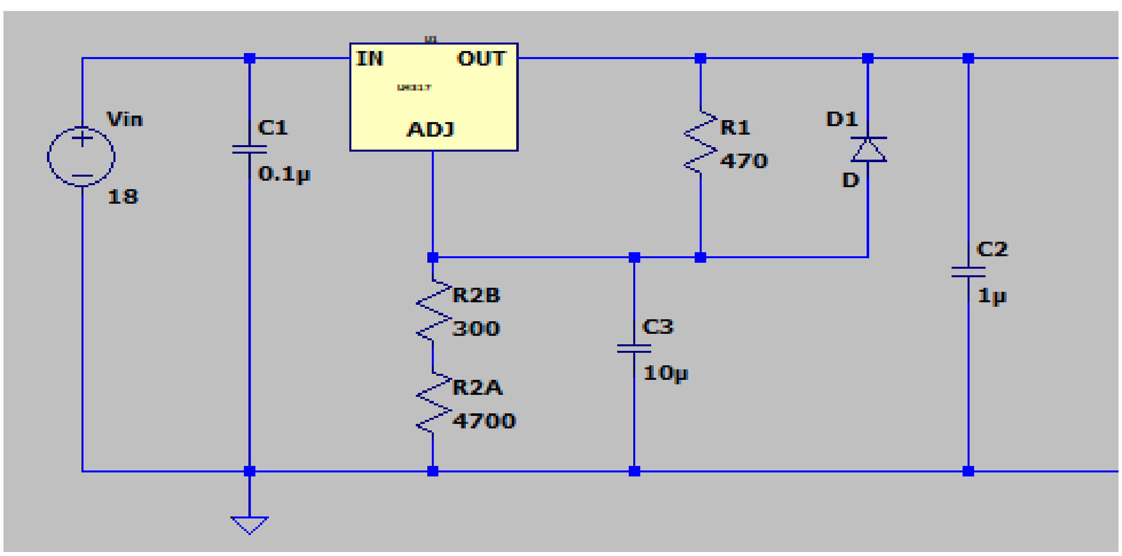

Figure 5: Voltage Regulator Schematic

Visual Warning Indicator Design (MA):

Ltspice Schematic:

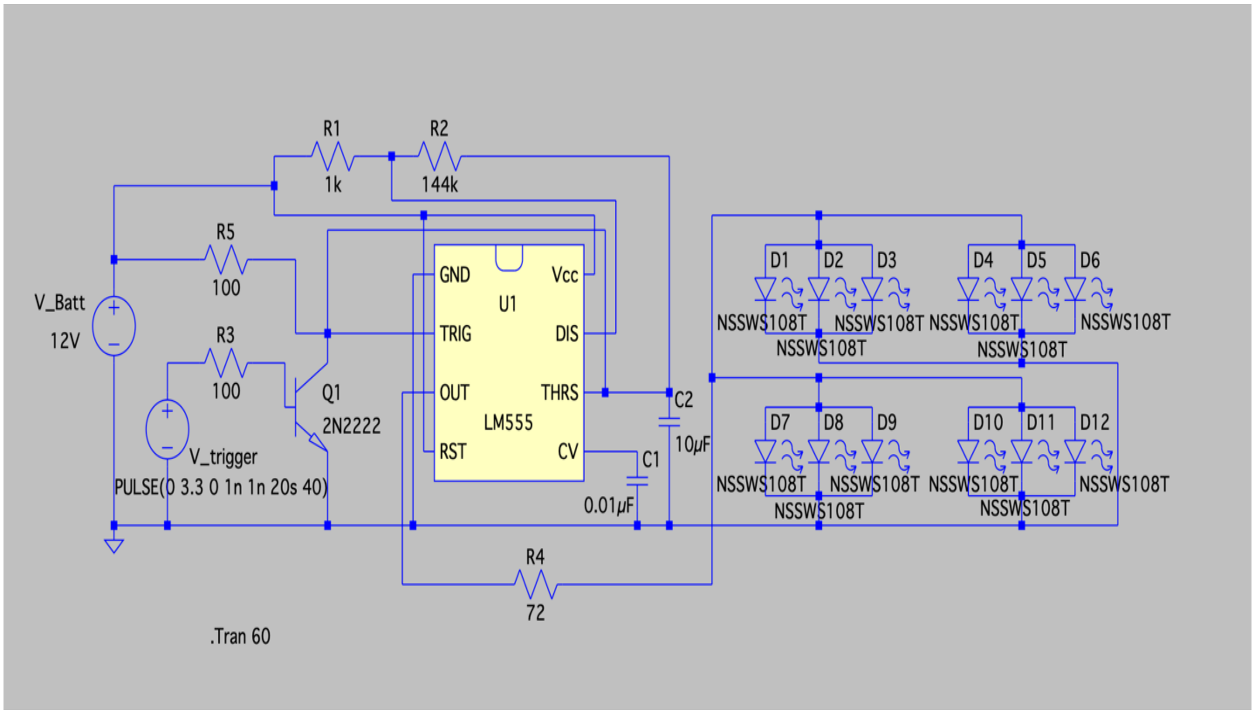

Figure 6: LM555-astable mode circuit including open-collector circuit

Ltspice Simulation:

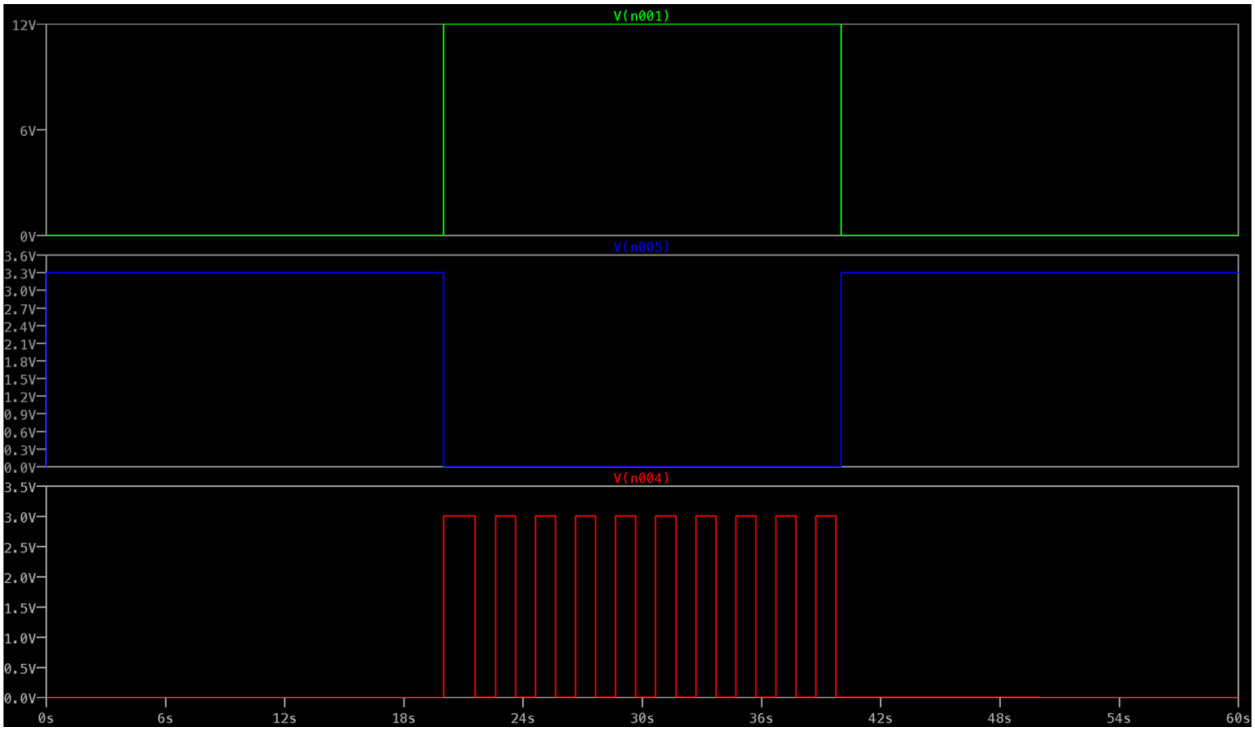

Figure 7: Top and bottom signals, LM555 input and output signals, respectively. Middle signal, Open-collector Signal.

Figure 7 shows the input signal of the 12V battery, top signal, initially as Low. Once a pedestrian is detected within the crosswalk range, the PIC24FJ128GA010 microcontroller signal, middle signal, is switched to Low. This activates the open-collector circuit and triggers the LM555, which "triggers from High to Low" [15]. In turn, the output of the LM555 is activated and flashes the LEDs on the crosswalk sign for approaching drivers. The triggering signal is kept Low for the duration in which the pedestrian is still within the range of the crosswalk sensors. The voltage level across each LED, bottom signal, was chosen based on the average input level of the 12V battery since it varies from 14.8 V, during daytime, to 10.5 V, during nighttime. As a result, LED brightness will be higher during the day and dimmer during the night.

5.2. Software Design (JS)

The PIC24FJ128GA010 [12] microcontroller can be programmed easily via the C programming language. *Note: while code is added in this section, to view the full program unabridged as it appears in C files, please see the Appendix.*

Level 0 Block Diagram of Software System (JS)

At its most fundamental, the software and hardware have the same inputs and outputs, with the exception that in order for a piece of software to run, having power to the hardware is implied and outside of the scope of the software. Essentially, utilizing the feedback of the sensors for pedestrian detection, the crosswalk signal

Figure 8: Level 0 Block Diagram of Software System

Module	Crosswalk Signal
Designers	• Mohammed Alsubhi • Aaron Gloeckler • Jacob Schandel • Samuel Turner
Inputs	• Ultrasonic Sensor Detection • PIR Sensor Detection
Outputs	• Warning signal • Wireless trigger to a paired signal unit with equivalent capabilities
Description	• Human detection system used for pedestrians, around and inside a crosswalk, to warn oncoming drivers from both sides of their presence through an LED sign

Table 4: Functional Requirements for Level 0 Block Diagram of Software System

Level 1 Flowchart of Software System (Main Process) (JS)

The main process of the system's software is, at its core, a loop which monitors for people and, when people are found, the person detected subsystem process drives the automated operation of the signage. This is simple to implement in C code, as shown below.

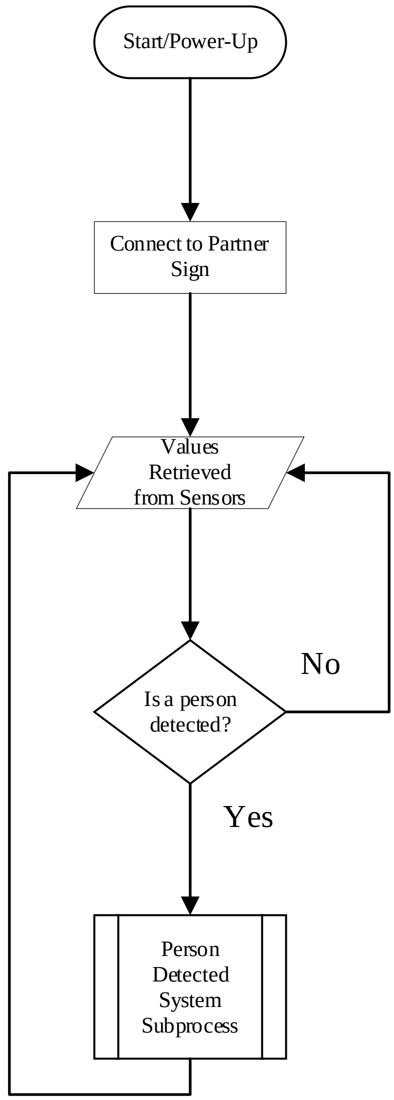

Figure 9: Level 1 Flowchart of Software System (Main Process)

Module	Crosswalk Signal Main Process
Inputs	• Information needed to communicate with partner unit across the crosswalk from this unit • Human position in crosswalk and adjacent entry/exit regions
Outputs	• Warning signal to roadside display on local unit • Wireless trigger to a paired signal unit with equivalent capabilities

Table 5: Functional Requirements Table for Level 1 Flowchart of Software System (Main Process)

```
/**
 * @brief This is how we look out for a person.
 */
int personFound(void);

/**
 * @brief Once a person is found, do this.
 */
void trackingPerson(int state);

/**
 * @brief MAIN PROGRAM
 */
int main(void)
{
    // initialize the device via MCC
    SYSTEM_Initialize();
    InitADC();

    // sign triggers on with a low signal from the PIC24
    SignLEDArray_SetHigh();

    // run special sensor initializations

    // initialize onboard LCD for debugging

    while (1)
    {
        // Add your application code
        int state = personFound();
```

```
        if (state != 0)
            trackingPerson(state);
    }

    return 1;
}
```
Figure 10: Main Process Code

Level 2 Flowchart of Software System (Person Detected System Subprocess) (JS):

The person detection subsystem details what occurs once a person has been detected. Essentially, the system outputs where a person is and where a person is going in the crosswalk. As in testing, the PIR sensors worked quite reliably, no timeout was needed for a person crossing the road.

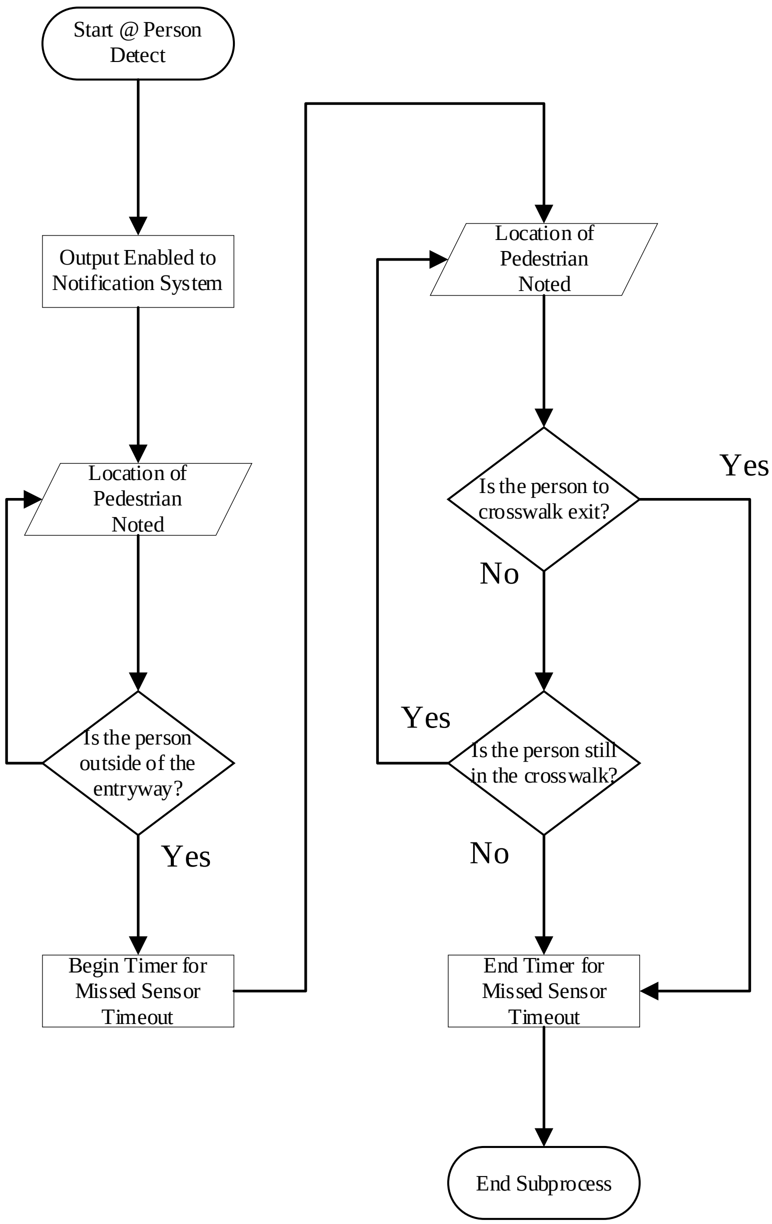

Figure 11: Level 2 Flowchart of Software System (Person Detected System Subprocess)

Module	Crosswalk Signal Main Process
Inputs	• Human position in crosswalk and adjacent entry/exit regions
Outputs	• Warning signal to roadside display on local unit • Wireless trigger to a paired signal unit with equivalent capabilities

Table 6: Functional Requirements Table for Level 2 Flowchart of Software System (Person Detected System Subprocess)

Level 3 Flowchart of Software System (Wireless Communications Subprocess) (JS):

The first level 3 system is our communication between signs. The communications system wirelessly beams information between signs. As we do not need to be communicating if nobody is in the crosswalk, we can program the first sign to detect a person to be the primary device, or “device A,” and the primary device will initiate communications. The other sign will be the secondary device, or “device B,” and will communicate in response to the primary. Assuming use of the transparent mode ZigBee devices, the code previously mentioned above for using the `printf()` function would be sufficient for sending strings of data between the signs quickly and easily. Also, such a method of sending information makes it easy to debug the system in a serial reading tool such as PuTTY.

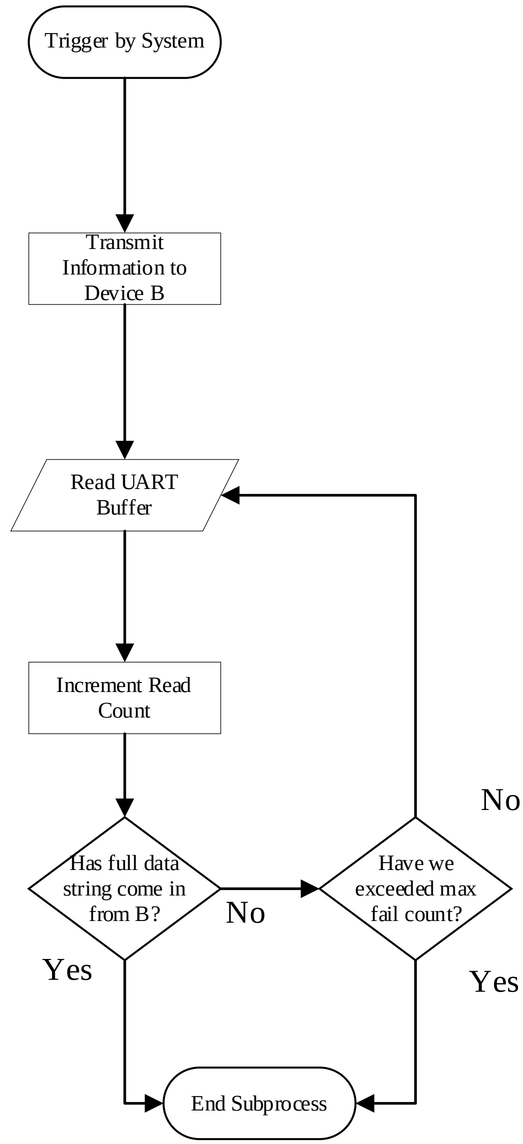

Figure 12: Level 3 Flowchart of Software System (Wireless Communications Subprocess)

Module	Crosswalk Signal Main Process
Inputs	• Human position in crosswalk and adjacent entry/exit regions
Outputs	• Wireless trigger to a paired signal unit with equivalent capabilities

Table 7: Functional Requirements Table for Level 3 Flowchart of Software System (Wireless Communications Subprocess)

Level 3 Flowchart of Software System (Sensor Subprocess) (AG, JS, ST):

The software for the system to read the analog sensors which detect pedestrians is quite simple. Simply put, it uses the A/D converter on the processor to read the current distance in. Below is a flowchart of this excessively simple process as well as a sample program where the analog sensor is analyzed and its current-read value is being displayed on the Explorer 16/32 board's LCD display. Note that depending on the sensor, the action in the dotted box may not be necessary.

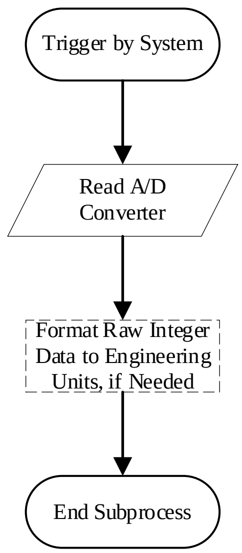

Figure 13: Level 3 Flowchart of Software System (Wireless Communications Subprocess)

Module	Crosswalk Signal Main Process
Inputs	• Human position in crosswalk and adjacent entry/exit regions
Outputs	• Distance from human detection sensors to rest of program to determine where person is in the crosswalk

Table 8: Functional Requirements Table for Level 3 Flowchart of Software System (Sensor Subprocess)

```
#include "mcc_generated_files/system.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define putLCD( d) WriteLCD( 1, (d))
#define CmdLCD( c) WriteLCD( 0, (c))
#define HomeLCD() WriteLCD( 0, 2) // See HD44780 instruction set in
#define ClrLCD() WriteLCD( 0, 1) // Table 9.1 of text book

void InitADC(int);
int ReadADC(int);
void ms_delay(int);
void InitPMP(void);
void InitLCD(void);
char ReadLCD(int);
void WriteLCD(int, char);
void SetCursorAtLine(int);
float cm; // initialize cm for the distance it is measured in

void ms_delay(int ms) {
    T2CON = 0x8030; // Timer 2 on, TCKPS<1,0> = 11 this 1:256
    TMR2 = 0;
    while (TMR2 < ms * 63); // 1/(16MHz/(256*63)) = 0.001008 close to 1ms
}

void InitADC(int amask) {
    AD1PCFG = 0xFFFE; // select AN0 as analog input
    AD1CON1 = 0x00E0; // auto convert @ end of sampling, Integer Data out.
    // see Text pg. 179 & Sec. 17 on AD1CON1.
    AD1CON2 = 0; // use MUXA, AVss and AVdd used as Vref
    AD1CON3 = 0x1F01; // Tad = 2xTcy = 125ns. 31*Tad for conversion time.
    AD1CSSL = 0; // no scanning required
    AD1CON1bits.ADON = 1; // Turn on the ADC
} // InitADC

int ReadADC(int ch) {
    AD1CHS = ch; // 1. select analog input channel
    // start sampling, automatic conversion will follow
    AD1CON1bits.SAMP = 1; // 2. Start sampling.
    while (!AD1CON1bits.DONE); //5. wait for conversion to complete
    AD1CON1bits.DONE = 0; // 6. clear flag. We are responsible see text.
    return ADC1BUF0; // 7. read the conversion results
} // ReadADC

void InitPMP(void) {
```

```
    // PMP initialization. See my notes in Sec 13 PMP of Fam. Ref. Manual
    PMCON = 0x8303; // Following Fig. 13-34. Text says 0x83BF (it works) *
    PMMODE = 0x03FF; // Master Mode 1. 8-bit data, long waits.
    PMAEN = 0x0001; // PMA0 enabled
}

void InitLCD(void) {
    // PMP is in Master Mode 1, simply by writing to PMDIN1 the PMP takes care
    // of the 3 control signals so as to write to the LCD.
    PMADDR = 0; // PMA0 physically connected to RS, 0 select Control register
    PMDIN1 = 0b00111000; // 8-
bit, 2 lines, 5X7. See Table 9.1 of text Function set
    ms_delay(1); // 1ms > 40us
    PMDIN1 = 0b00001100; // ON, cursor off, blink off
    ms_delay(1); // 1ms > 40us
    PMDIN1 = 0b00000001; // clear display
    ms_delay(2); // 2ms > 1.64ms
    PMDIN1 = 0b00000110; // increment cursor, no shift
    ms_delay(2); // 2ms > 1.64ms
} // InitLCD

char ReadLCD(int addr) {
    // As for dummy read, see 13.4.2, the first read has previous value in PMDIN1
    int dummy;
    while (PMMODEbits.BUSY); // wait for PMP to be available
    PMADDR = addr; // select the command address
    dummy = PMDIN1; // initiate a read cycle, dummy
    while (PMMODEbits.BUSY); // wait for PMP to be available
    return ( PMDIN1); // read the status register
} // ReadLCD
// In the following, addr = 0 -> access Control, addr = 1 -> access Data
#define BusyLCD() ReadLCD( 0) & 0x80 // D<7> = Busy Flag
#define AddrLCD() ReadLCD( 0) & 0x7F // Not actually used here
#define getLCD() ReadLCD( 1) // Not actually used here.

void WriteLCD(int addr, char c) {
    while (BusyLCD());
    while (PMMODEbits.BUSY); // wait for PMP to be available
    PMADDR = addr;
    PMDIN1 = c;
} // WriteLCD
// In the following, addr = 0 -> access Control, addr = 1 -> access Data

void putsLCD(char *s) {
    while (*s) putLCD(*s++); // See paragraph starting at bottom, pg. 87 text
```

```
} //putsLCD

void SetCursorAtLine(int i) {
    int k;
    if (i == 1)
        CmdLCD(0x80); // Set DDRAM (i.e., LCD) address to upper left (0x80 | 0x00
)
    else if (i == 2)
        CmdLCD(0xC0); // Set DDRAM (i.e., LCD) address to lower left (0x80 | 0x40
)
    else {
        TRISA = 0x00; // Set PORTA<7:0> for output.
        for (k = 1; k < 20; k++) // Flash all 7 LED's @ 5Hz. for 4 seconds.
        {
            PORTA = 0xFF;
            ms_delay(100); // 100 ms for ON then OFF yields 5Hz
            PORTA = 0x00;
            ms_delay(100);
        }
    }
}

main() {
    InitADC(0xFFEF); // initialize the ADC and analog inputs
    float centimeter = 2.54; // 1 in per  centimeter = 2.54
    int distance_IN; //distance measured in inches
    char distance_string [16];
    TRISB = 0; // all PORTB pins as outputs
    InitPMP(); // Initialize the Parallel Master Port
    InitLCD(); // Initialize the LCD

    while (1) // main loop
    {
        distance_IN = ReadADC(0); // read the sensor on pin 25, AN0, the value is
 read in inches based of data sheet
        cm = (distance_IN - 16) ; // inches to cm (*2.54) sensor goes from 10cm (
4") to roughly 80cm (32"), this is the conversion with a -
16 b/c it has that as a initial value with nothing plug in

        if ((10 >= cm) || (cm <= 80)) { // sensors go as high as 88.9cm or 35" so
 set a bound for no higher then 32" based of datasheet
            sprintf(distance_string, "%0.2f cm", cm); // taking a numerical cm va
lue to convert a formatted string
            ms_delay(400); // Just like in LCD lab, probably not needed.
            SetCursorAtLine(1); // Set to first line.
```

```
            putsLCD("Distance is :"); // Put message on first line of LCD
            SetCursorAtLine(2); // Set to start of second line
            putsLCD(distance_string); // Reading ADC I/O for sensor value
            ms_delay(400); // Just like in LCD lab, probably not needed.
        }
        else {

            SetCursorAtLine(1); // Set to first line.
            putsLCD("Out of range"); // Put message on first line of LCD
            SetCursorAtLine(2); // Set to start of second line
            putsLCD("Out of range"); // Put message on the second line of LCD

        }
    }
} // main
```

Figure 14: Sample Code of Second Level 3 Subsystem for Testing Purposes Only (Sensor Subprocess)

Additional to this code, as the PIC24FJ128GA010 only has one analog-to-digital converter onboard, the two sensors for the system must share the same A-D converter on the chip. As the sensors need not be read simultaneously, this is not a problem to do physically. To implement this in code, a changeover function was programmed, which lets the program set a channel to observe. Note that channel 1 was used for the PIR sensor and channel 2 was used for the ultrasonic sensor.

```
/**
 * @brief Selects channel to read from for a
 *
 * @param ch Analog channel to read from on processor
 */
void SelectPort(int ch)
{
    AD1CON1bits.ADON = 0; // Turn off the ADC to reconfigure

    switch(ch) // set values based on the channel to use
    {
        case 0: // select AN1 as analog input
            AD1PCFG = 0xFFFE;
            break;
```

```
        case 1:
            AD1PCFG = 0xFFFD; // select AN1 as analog input
            break;

        case 2:
            AD1PCFG = 0xFFFB; // select AN2 as analog input
            break;

        // there's only so many options here, so there's not really a default cas
e
    }

    AD1CON1bits.ADON = 1; // Turn on the ADC

    AD1CHS = ch; // 1. select analog input channel
}

/**
 * @brief Read value from ADC based on selected channel via SelectPort()
 *
 * @return Value from ADC
 */
int ReadADC(void)
{
    AD1CON1bits.SAMP = 1; // 2. Start sampling.
    while (!AD1CON1bits.DONE); //5. wait for conversion to complete
    AD1CON1bits.DONE = 0; // 6. clear flag. We are responsible see text.
    return ADC1BUF0; // 7. read the conversion results
}

/**
 * @brief This is how we look out for a person.
 *
 * @return integer value representing current detection state (0 = clear, 1 = per
son detected on this side of the sign, 2 = person detected at other sign)
 *
 */
int personFound(void)
{
    // has a person been found via the local sensor?
    if (ReadADChannel(2) <= 96)  // todo: specify #
    {
        // wait a moment to verify
        ms_delay(3000);
```

```
        // check again to make sure sensor wasn't a false flag
        if (ReadADChannel(2) <= 96)  // todo: specify #
        {
            sendFlag('D');
            return 1;
        }
    }

    // has a person been found via the remote sensor?
    char flag = receiveFlag();
    if (flag == 'D')
    {
        return 2;
    }

    // if nobody detected on either side, nobody is there
    return 0;
}
```
Figure 15: Analog Changeover Code Used in Project with Usage

Level 3 Flowchart of Software System (Digital Signage Subprocess) (MA, JS):

The signage will utilize two lighting circuits triggered by 2N2222 transistors. When a low signal (0V, with 3.3V defined as a high signal) is sent from the PIC microcontroller to the sign's illumination circuit, the lights will turn on. When a high signal is going to this circuit, the sign will turn off. This is a fairly basic system to implement in code.

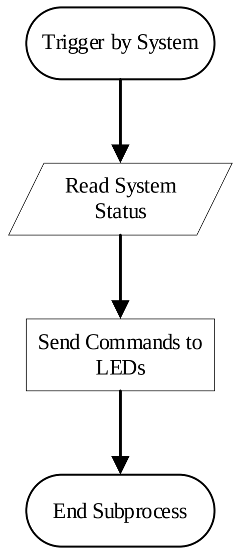

Figure 16: Level 3 Flowchart of Software System (Digital Signage Subprocess)

Module	Crosswalk Signal Main Process
Inputs	• Human position in crosswalk and adjacent entry/exit regions
Outputs	• Illuminated signal to oncoming traffic to stop for crossing pedestrians

Table 9: Functional Requirements Table for Level 3 Flowchart of Software System (Digital Signage Subprocess)

```c
/**
 * @brief Delay the program by a set number of milliseconds
 *
 * @param ms delay time in milliseconds
 */
void ms_delay(int ms)
{
    T2CON = 0x8030; // Timer 2 on, TCKPS<1,0> = 11 this 1:256
    TMR2 = 0;
    while (TMR2 < ms * 63); // 1/(16MHz/(256*63)) = 0.001008 close to 1ms
}

/**
 * @brief Once a person is found, do this.
 */
void trackingPerson(int stateSet)
{
    // sign on
    SignLEDArray_SetLow();

    // wait for someone to enter
    while (ReadADChannel(1) == 0)
    {
        ms_delay(500);
    }

    // wait for someone to exit range
    while (ReadADChannel(1) != 0)
    {
        ms_delay(500);
    }

    // sign off
    SignLEDArray_SetHigh();
}
```

Figure 17: Code for Sign Sequencer, Noting that Main Program Initialized with High Signal to Sign Circuitry

6. Mechanical Sketch (JS)

Mechanically, the system consists of two sign posts that communicate with each other wirelessly. The sign posts are on opposite sides of the road facing opposite directions in order to best signal to automotive traffic in both directions.

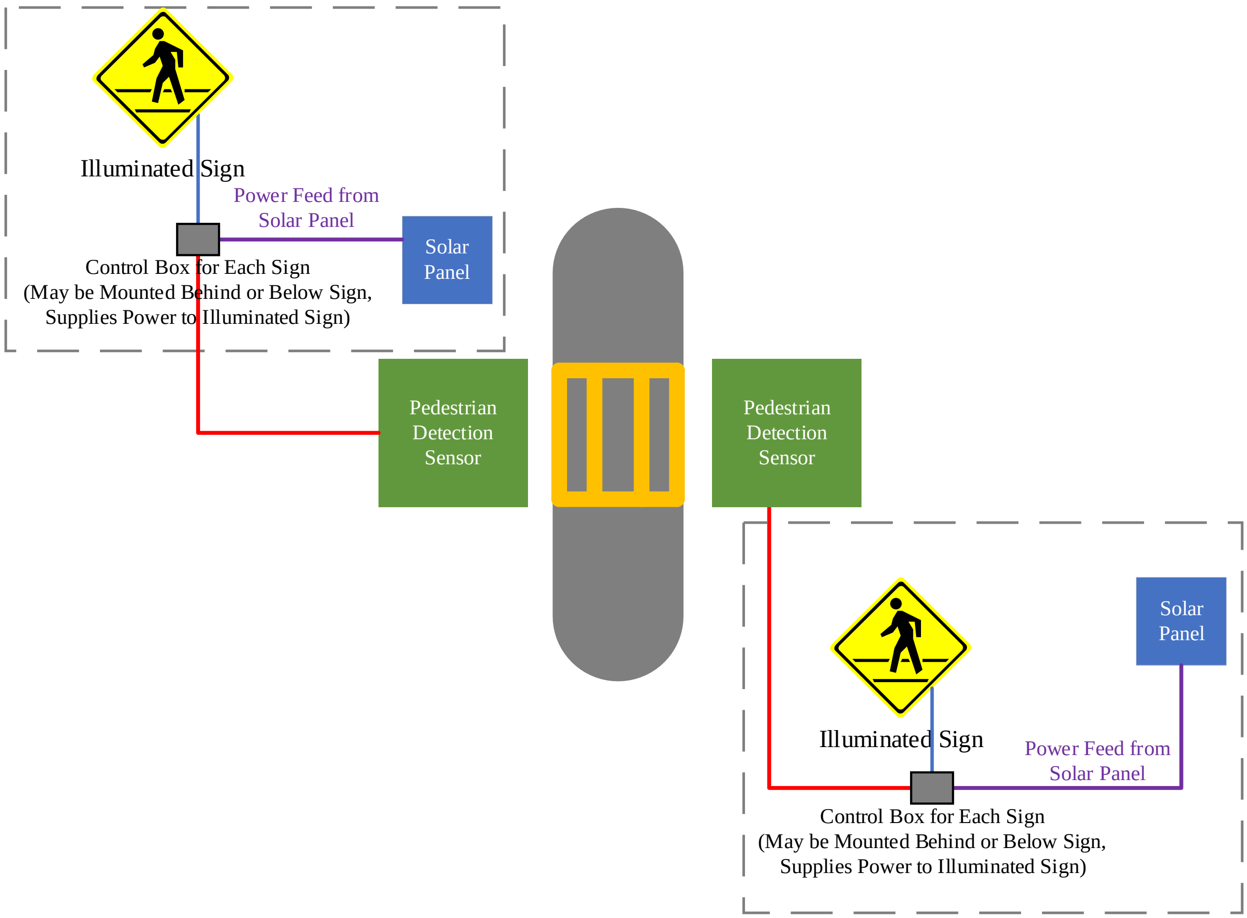

Figure 18: Mechanical Sketch of Crosswalk Sensor System

7. Design Team Information:

Mohammad Alsubhi, Electrical Engineering Major

Aaron Gloeckler, Electrical Engineering Major

Jacob Schandel, Computer Engineering Major

Samuel Turner, Electrical Engineering Major

8. Parts List

Below is a list of the parts projected to be used and selected as well as what has been spent or will be spent on these parts.

Qty.	Refdes	Part Num.	Description	Suggested Vendor	Vendor Part Num.	Cost
8	R1, R2, …, R8	MFR-25FBF52-9K53	9.53k resistor	Digikey	9.53KXBK-ND	0.8
8	Dr1, Dr2, …, Dr8	MAX7219ENG+	LED Driver 24-PDIP	Digikey	MAX7219ENG+-ND	70.08
524	D1, …, D512	L513YD	Light Emitting Diodes	Digikey	2460-L513YD-ND	31.96
2	R9, R10	resistor	2k resistor	on campus	on campus	
4	VR1, VR2, VR3, VR4	LM317	Adjustable voltage regulator	On campus: D04	on campus	
8	C1, C2, …, C8	C320C104M5R5TA	0.1 uF Capacitor	Digikey	399-9776-ND	1.92
4	R11, R12, R13, R14	5k Ohm Resistor	5k Ohm Resistor	On campus	on campus	
4	R15, R16, R17, R18	240 Ohm Resistor	240 Ohm Resistor	On campus	on campus	
4	C9, C10, C11, C12	10 uF Capacitor	10 uF Capacitor	On campus	on campus	
2	C13, C14	1000 uF Capacitor	1000 uF Capacitor	On campus	on campus	
4	R19, R20, R21, R22	10k Ohm Resistor	10k Ohm Resistor	On campus	on campus	
4	R23, R24, R25, R26	120 Ohm Resistor	120 Ohm Resistor	On campus	on campus	
8	LED1, LED2, …	LED 7SEGA	LED 7 segment display	On campus: EE05	on campus	
2	B1, B2	ML3-12	12V, 40W, 3.4AH SLA battery Replacement (rechargeable)	Amazon	ML3-12	53.98
2	XBEE1, XBEE2	XB3-24Z8PT-J	Digi XBee Module	On campus	On campus	
2	SP1, SP2	B07WYZPN2B	50W mono Crystalline 12V solar panel	On campus	On campus	
2	US1, US2	LV-EZ4-982	Maxbotix Ultrasonic Rangefinder - LV-EZ4 - LV-EZ4	Adafruit	LV-EZ4-982	51.90
2	IRDS1, IRDS2	GP2Y0A21YK0F	IR distance sensor includes cable (10cm-80cm) - GP2Y0A21YK0F	Adafruit	GP2Y0A21YK0F	29.90
2	IRDS3, IDRS4	GP2Y0A710K0F	IR Distance Sensor - Includes Cable (100cm-500cm)	Adafruit	GP2Y0A710K0F	59.90
					Total Project Expenses for Fall 2020	***300.44***

Below are the final parts requested, used, and implemented in the final design of the project. New items were added as further analysis was done, and others were removed if not used in the design.

Qty.	Refdes	Part Num.	Description	Suggested Vendor	Vendor Part Num.	Cost
1	LED1, …	FLR-50T04-HW7	Super Bright White LEDs (25 pack)	adafruit	754	6.95
2	SIGN1, SIGN2	T1-1093	3M Reflective crosswalk sign 12" x 18"	Amazon	T1-1093-EG_12x18	38.00
2	PIR1, PIR2	PIR 189	Passive Infrared Motion Sensor (PIR)	Adafruit	189	19.90
1	12V DC Power connector	8541689448	12V DC Power Connector 5.5mm x 2.1mm	Amazon	8541689448	7.49
4	Cable for IR range sensor	DMC03-SC200	Cable for Sharp GP2Y0A710K 0F IR Range Sensor	RobotShop	DMC03-SC200	8.24
2	Batt1, Batt2	EXP1270	ExpertPower Standard 12V 7AH Rechargeable SLA Battery (2 PACK)	Amazon	EXP1270-2	34.00
4	Proto1,...,Proto4	Perma Proto	Adafruit Perma-Proto Half-sized Breadboard PCB - Single	Adafruit	Adafruit Perma Proto	18.00
5	Tr1, …, Tr5	2N2222	Npn transistor	On campus	On campus	

8	LEDS1, …, LEDS8	N/A	Individual LED strip	On campus	On campus	
4	R1, …, R4	N/A	100-ohm resistor	On campus	On Campus	
2	US1, US2	LV-EZ0-982	Maxbotix Ultrasonic Rangefinder - LV-EZ0 - LV-EZ0	Adafruit	LV-EZ0-982	49.98
4	N/A	11896	Header, 10-pin female	SparkFun	11896	2.00
1	N/A	BOB-08276	Xbee Adapter Board	SparkFun	BOB-08276	2.95
					Total Project Expenses for Spring 2021	187.51

9. Project Schedules (MA, AG, JS, ST):

The following is an updated table from the project Gantt chart showing task completion statuses as of the end of the fall 2020 semester.

Task Name	Start	Finish	Resource Names	% Complete
Project Design	Mon 8/24/20	Wed 11/25/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Midterm Report	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Cover Page	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
T of C, L of T, L of F	Mon 8/24/20	Mon 10/5/20	Jacob Schandel,Mohammed Alsubhi	100%
Problem Statement	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Need	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler	100%
Objective	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler	100%
Background	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler	100%
Marketing Requirements	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler	100%
Engineering Requirements Specification	Mon 8/24/20	Tue 9/29/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Decision Made: 1 vs. 2 Signs	Mon 9/21/20	Mon 9/21/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Engineering Analysis	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Circuits	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Mohammed Alsubhi,Samuel Turner	100%
Research Sensors	Mon 8/24/20	Sun 9/27/20	Aaron Gloeckler,Samuel Turner,Mohammed Alsubhi	100%

Electronics	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Mohammed Alsubhi,Samuel Turner	100%
Research Electronic Signage	Thu 9/24/20	Thu 10/1/20	Mohammed Alsubhi	100%
Signal Processing	Mon 8/24/20	Mon 10/5/20	Jacob Schandel,Samuel Turner	100%
Communications/Computer Networks	Mon 8/24/20	Mon 10/5/20	Jacob Schandel,Samuel Turner	100%
Research Communication Methods	Mon 9/21/20	Tue 9/29/20	Jacob Schandel	100%
Embedded Systems	Mon 8/24/20	Mon 10/5/20	Jacob Schandel,Samuel Turner	100%
Accepted Technical Design	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Hardware Design: Phase 1	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Mohammed Alsubhi,Samuel Turner	100%
Hardware Block Diagrams Level 0	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Hardware Block Diagrams Level 1	Mon 8/24/20	Tue 9/29/20	Samuel Turner	100%
Software Design: Phase 1	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Software Behavior Model Level 0	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Software Behavior Model Level 1	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Software Behavior Model Level 2	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Mechanical Sketch	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Team Information	Mon 8/24/20	Mon 10/5/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Slide Deck Due for Presentation (Based on Midterm Content)	Sat 8/29/20	Sat 8/29/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Project Schedules (Gantt Chart)	Mon 8/24/20	Mon 10/5/20	Jacob Schandel	100%
Paper Due	Mon 10/5/20	Mon 10/5/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%

Presentation to Class	Wed 10/7/20	Wed 10/7/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Project Poster	Wed 11/18/20	Wed 12/2/20		100%
Final Design Report	Tue 10/6/20	Wed 11/25/20		100%
Abstract	Tue 10/6/20	Mon 11/23/20		100%
Hardware Design: Phase 2	Tue 10/6/20	Mon 11/23/20	Aaron Gloeckler	100%
Modules 1…n	Tue 10/6/20	Mon 11/23/20		100%
Simulations	Tue 10/6/20	Mon 11/23/20	Aaron Gloeckler	100%
Schematics	Tue 10/6/20	Mon 11/23/20	Aaron Gloeckler	100%
Software Design: Phase 2	Tue 10/6/20	Mon 11/23/20		100%
Code (working subsystems)	Tue 10/6/20	Mon 11/23/20		100%
System integration Behavior Models	Tue 10/6/20	Mon 11/23/20		100%
Parts Lists	Tue 10/6/20	Mon 11/23/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Parts lists for schematics	Tue 10/6/20	Mon 11/23/20		100%
Materials Budget list	Tue 10/6/20	Mon 11/23/20	Aaron Gloeckler, Samuel Turner	100%
Proposed Implementation Gantt Chart	Tue 10/6/20	Mon 11/23/20	Jacob Schandel	100%
Conclusions and Recommendations	Tue 10/6/20	Mon 11/23/20		100%
Final Parts Request Form	Sun 10/11/20	Sat 10/24/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%
Subsystems Demonstrations	Thu 11/19/20	Thu 11/19/20	Aaron Gloeckler,Jacob Schandel,Mohammed Alsubhi,Samuel Turner	100%

Table 10: Fall 2020 Design Gantt Chart Dates, Completions, and Assignments

The following outlines what was the projected schedule for spring 2021 as we implement the final system for the end of the semester.

Task Name	Duration	Start	Finish	Predecessors	Resource Names
SDP2 Implementation 2020	**103 days**	**Mon 1/11/21**	**Fri 4/23/21**		
Revise Gantt Chart	14 days	**Mon 1/11/21**	Sun 1/24/21		
Implement Project Design	**89 days**	**Mon 1/11/21**	**Fri 4/9/21**		
Hardware Implementation	**47 days**	**Mon 1/11/21**	**Sat 2/27/21**		
Breadboard Components	**14 days**	**Mon 1/11/21**	**Sun 1/24/21**		
LED Sign	14 days	Mon 1/11/21	Sun 1/24/21		Mohammed Alsubhi
Solar/Battery Circuit	14 days	Mon 1/11/21	Sun 1/24/21		Aaron Gloeckler
ZigBee	14 days	Mon 1/11/21	Sun 1/24/21		Jacob Schandel
Sensors	14 days	Mon 1/11/21	Sun 1/24/21		Sam Turner
Assemble Hardware	**14 days**	**Mon 1/11/21**	**Sun 1/24/21**		
LED Sign	14 days	Mon 1/11/21	Sun 1/24/21		Mohammed Alsubhi
Solar/Battery Circuit	14 days	Mon 1/11/21	Sun 1/24/21		Aaron Gloeckler
ZigBee	14 days	Mon 1/11/21	Sun 1/24/21		Jacob Schandel
Sensors	14 days	Mon 1/11/21	Sun 1/24/21		Sam Turner
Test Hardware	**14 days**	**Mon 1/25/21**	**Sun 2/7/21**	**14**	
LED Sign	7 days	Mon 1/25/21	Sun 1/31/21	14	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Mon 1/25/21	Sun 1/31/21	14	Aaron Gloeckler
ZigBee	7 days	Mon 2/1/21	Sun 2/7/21	14	Jacob Schandel

Sensors	7 days	Mon 1/25/21	Sun 1/31/21	14	Sam Turner
Revise Hardware	**7 days**	**Mon 2/1/21**	**Sun 2/7/21**	**19**	
LED Sign	7 days	Mon 2/1/21	Sun 2/7/21	19	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Mon 2/1/21	Sun 2/7/21	19	Aaron Gloeckler
ZigBee	7 days	Mon 2/1/21	Sun 2/7/21	19	Jacob Schandel
Sensors	7 days	Mon 2/1/21	Sun 2/7/21	19	Sam Turner
MIDTERM: Demonstrate Hardware Subsystems	5 days	Mon 2/8/21	Fri 2/12/21	24	
SDC & FA Hardware Approval	0 days	Sat 2/27/21	Sat 2/27/21	25	
Software Implementation	**47 days**	**Mon 1/11/21**	**Sat 2/27/21**		
Develop Software	**28 days**	**Mon 1/11/21**	**Sun 2/7/21**		
LED Sign	28 days	Mon 1/11/21	Sun 2/7/21		Mohammed Alsubhi
ZigBee	28 days	Mon 1/11/21	Sun 2/7/21		Jacob Schandel
Sensors	28 days	Mon 1/11/21	Sun 2/7/21		Sam Turner
Test Software	**28 days**	**Mon 1/11/21**	**Sun 2/7/21**		
LED Sign	28 days	Mon 1/11/21	Sun 2/7/21		Mohammed Alsubhi
ZigBee	28 days	Mon 1/11/21	Sun 2/7/21		Jacob Schandel
Sensors	28 days	Mon 1/11/21	Sun 2/7/21		Sam Turner
Revise Software	**14 days**	**Mon 2/8/21**	**Sun 2/21/21**	**28**	
LED Sign	14 days	Mon 2/8/21	Sun 2/21/21	28	Mohammed Alsubhi
ZigBee	14 days	Mon 2/8/21	Sun 2/21/21	28	Jacob Schandel
Sensors	14 days	Mon 2/8/21	Sun 2/21/21	28	Sam Turner

MIDTERM: Demonstrate Software Subsystems	5 days	Mon 2/22/21	Fri 2/26/21	42	
SDC & FA Software Approval	0 days	Sat 2/27/21	Sat 2/27/21	43	
System Integration	**42 days**	**Sat 2/27/21**	**Fri 4/9/21**		
Assemble Complete System Integration	**14 days**	**Sat 2/27/21**	**Fri 3/12/21**	**43**	
LED Sign	14 days	Sat 2/27/21	Fri 3/12/21	43	Mohammed Alsubhi
Solar/Battery Circuit	14 days	Sat 2/27/21	Fri 3/12/21	43	Aaron Gloeckler
ZigBee	14 days	Sat 2/27/21	Fri 3/12/21	43	Jacob Schandel
Sensors	14 days	Sat 2/27/21	Fri 3/12/21	43	Sam Turner
Test Complete System Integration	**7 days**	**Sat 3/13/21**	**Fri 3/19/21**	**46**	
LED Sign	7 days	Sat 3/13/21	Fri 3/19/21	46	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Sat 3/13/21	Fri 3/19/21	46	Aaron Gloeckler
ZigBee	7 days	Sat 3/13/21	Fri 3/19/21	46	Jacob Schandel
Sensors	7 days	Sat 3/13/21	Fri 3/19/21	46	Sam Turner
Revise Complete System Integration	**16 days**	**Sat 3/20/21**	**Sun 4/4/21**	**51**	
LED Sign	16 days	Sat 3/20/21	Sun 4/4/21	51	Mohammed Alsubhi
Solar/Battery Circuit	16 days	Sat 3/20/21	Sun 4/4/21	51	Aaron Gloeckler
ZigBee	16 days	Sat 3/20/21	Sun 4/4/21	51	Jacob Schandel
Sensors	16 days	Sat 3/20/21	Sun 4/4/21	51	Sam Turner
Demonstration of Complete System	5 days	Mon 4/5/21	Fri 4/9/21	56	
Develop Final Report	**103 days**	**Mon 1/11/21**	**Fri 4/23/21**		
Write Final Report	103 days	Mon 1/11/21	Fri 4/23/21		
Submit Final Report	0 days	Fri 4/23/21	Fri 4/23/21	63	
Project Demonstration and Presentation	0 days	Thu+ 4/9/21	Thu 4/9/21		

Table 11: Spring 2021 Projected Gantt Chart Dates, Completions, and Assignments

The following outlines what was the actual schedule for spring 2021 as we implement the final system for the end of the semester. During spring semester, we saw a few unforeseen setbacks, including but not limited to the following large issues:

- Issues with integration of the lighting system
- Issues with the original IR sensor for in the crosswalk

Task Name	Duration	Start	Finish	Predecessors	Resource Names
SDP2 Implementation 2020	**103 days**	**Mon 1/11/21**	**Fri 4/23/21**		
Revise Gantt Chart	14 days	**Mon 1/11/21**	Sun 1/24/21		
Implement Project Design	**89 days**	**Mon 1/11/21**	**Fri 4/9/21**		
Hardware Implementation	**47 days**	**Mon 1/11/21**	**Sat 2/27/21**		
Breadboard Components	**14 days**	**Mon 1/11/21**	**Sun 1/24/21**		
LED Sign	14 days	Mon 1/11/21	Sun 1/24/21		Mohammed Alsubhi
Solar/Battery Circuit	14 days	Mon 1/11/21	Sun 1/24/21		Aaron Gloeckler
ZigBee	14 days	Mon 1/11/21	Sun 1/24/21		Jacob Schandel
Sensors	14 days	Mon 1/11/21	Sun 1/24/21		Sam Turner
Assemble Hardware	**14 days**	**Mon 1/11/21**	**Sun 1/24/21**		
LED Sign	14 days	Mon 1/11/21	Sun 1/24/21		Mohammed Alsubhi
Solar/Battery Circuit	14 days	Mon 1/11/21	Sun 1/24/21		Aaron Gloeckler
ZigBee	14 days	Mon 1/11/21	Sun 1/24/21		Jacob Schandel
Sensors	14 days	Mon 1/11/21	Sun 1/24/21		Sam Turner
Test Hardware	**7 days**	**Mon 1/25/21**	**Sun 1/31/21**	**14**	

LED Sign	7 days	Mon 1/25/21	Sun 1/31/21	14	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Mon 1/25/21	Sun 1/31/21	14	Aaron Gloeckler
ZigBee	7 days	Mon 1/25/21	Sun 1/31/21	14	Jacob Schandel
Sensors	7 days	Mon 1/25/21	Sun 1/31/21	14	Sam Turner
Revise Hardware	**7 days**	**Mon 2/1/21**	**Sun 2/7/21**	**19**	
LED Sign	7 days	Mon 2/1/21	Sun 2/7/21	19	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Mon 2/1/21	Sun 2/7/21	19	Aaron Gloeckler
ZigBee	7 days	Mon 2/1/21	Sun 2/7/21	19	Jacob Schandel
Sensors	15 days	Thu 2/4/21	Thu 2/18/21	19	Sam Turner
MIDTERM: Demonstrate Hardware Subsystems	0 days	Wed 2/10/21	Wed 2/10/21		
SDC & FA Hardware Approval	0 days	Sat 2/27/21	Sat 2/27/21	25	
Software Implementation	**47 days**	**Mon 1/11/21**	**Sat 2/27/21**		
Develop Software	**28 days**	**Mon 1/11/21**	**Sun 2/7/21**		
LED Sign	28 days	Mon 1/11/21	Sun 2/7/21		Mohammed Alsubhi
ZigBee	28 days	Mon 1/11/21	Sun 2/7/21		Jacob Schandel
Sensors	28 days	Mon 1/11/21	Sun 2/7/21		Sam Turner
Test Software	**28 days**	**Mon 1/11/21**	**Sun 2/7/21**		
LED Sign	28 days	Mon 1/11/21	Sun 2/7/21		Mohammed Alsubhi
ZigBee	28 days	Mon 1/11/21	Sun 2/7/21		Jacob Schandel
Sensors	28 days	Mon 1/11/21	Sun 2/7/21		Sam Turner
Revise Software	**14 days**	**Mon 2/8/21**	**Sun 2/21/21**	**28**	

LED Sign	14 days	Mon 2/8/21	Sun 2/21/21	28	Mohammed Alsubhi
ZigBee	14 days	Mon 2/8/21	Sun 2/21/21	28	Jacob Schandel
Sensors	14 days	Mon 2/8/21	Sun 2/21/21	28	Sam Turner
MIDTERM: Demonstrate Software Subsystems	5 days	Mon 2/22/21	Fri 2/26/21	39	
SDC & FA Software Approval	0 days	Sat 2/27/21	Sat 2/27/21	40	
System Integration	**42 days**	**Sat 2/27/21**	**Fri 4/9/21**		
Assemble Complete System Integration	**14 days**	**Sat 2/27/21**	**Fri 3/12/21**	**40**	
LED Sign	14 days	Sat 2/27/21	Fri 3/12/21	40	Mohammed Alsubhi
Solar/Battery Circuit	14 days	Sat 2/27/21	Fri 3/12/21	40	Aaron Gloeckler
ZigBee	14 days	Sat 2/27/21	Fri 3/12/21	40	Jacob Schandel
Sensors	14 days	Sat 2/27/21	Fri 3/12/21	40	Sam Turner
Test Complete System Integration	**7 days**	**Sat 3/13/21**	**Fri 3/19/21**	**43**	
LED Sign	7 days	Sat 3/13/21	Fri 3/19/21	43	Mohammed Alsubhi
Solar/Battery Circuit	7 days	Sat 3/13/21	Fri 3/19/21	43	Aaron Gloeckler
ZigBee	7 days	Sat 3/13/21	Fri 3/19/21	43	Jacob Schandel
Sensors	7 days	Sat 3/13/21	Fri 3/19/21	43	Sam Turner
Revise Complete System Integration	**16 days**	**Sat 3/20/21**	**Sun 4/4/21**	**48**	
LED Sign	16 days	Sat 3/20/21	Sun 4/4/21	48	Mohammed Alsubhi
Solar/Battery Circuit	16 days	Sat 3/20/21	Sun 4/4/21	48	Aaron Gloeckler
ZigBee	16 days	Sat 3/20/21	Sun 4/4/21	48	Jacob Schandel
Sensors	16 days	Sat 3/20/21	Sun 4/4/21	48	Sam Turner
Demonstration of Complete System	5 days	Mon 4/5/21	Fri 4/9/21	53	
Develop Final Report	**103 days**	**Mon 1/11/21**	**Fri 4/23/21**		

Write Final Report	103 days	Mon 1/11/21	Fri 4/23/21		
Submit Final Report	0 days	Sat 4/24/21	Sat 4/24/21	60	
Spring Recess	7 days	Mon 4/12/21	Sun 4/18/21		
Project Demonstration and Presentation	0 days	Fri 4/9/21	Fri 4/9/21		

Table 12: Spring 2021 Actual Gantt Chart Dates, Completions, and Assignments

10. Conclusions and Recommendations (JS)

In conclusion, crosswalks can pose serious safety concerns in a world where distracted driving and racing to the finish of everything are both prominent. By providing motorists with more powerful notifications that individuals may be in the road ahead, the safety of both the motorist and the pedestrian can be greatly increased. Therefore, we are recommending the use of special signage incorporating human-detecting sensors to better alert oncoming traffic of pedestrians utilizing the crosswalk ahead.

11. References

[1] B. J. Campbell, *A review of pedestrian safety research in the United States and abroad*. McLean, VA: U.S. Department of Transportation, Federal Highway Administration, Research, Development, and Technology, Turner-Fairbank Highway Research Center, 2004.

[2] Turner-Fairbank Highway Research Center, Ed., *Crosswalk marking field visibility study*. McLean, VA: U.S. Federal Highway Administration, Research, Development, and Technology, Turner-Fairbank Highway Research Center, 2010.

[3] D. Sennett and B. Daly, "Devices and methods for controlling a change of status of traffic light at a crosswalk," US8462017B2, Jun. 11, 2013.

[4] "Distracted Driving | Facts and Stats | Texting and Driving." http://wayback.archive-it.org/3109/20130305131643/http://www.distraction.gov/content/get-the-facts/facts-and-statistics.html (accessed Apr. 04, 2020).

[5] John A. Volpe National Transportation Systems Center (U.S.) and United States, Eds., *Evaluation of the focused approach to pedestrian safety program: prepared for the Federal Highway Administration Office of Safety*. Washington, D.C.: U.S. Dept. of Transportation, Federal Highway Administration, Office of Safety, 2009.

[6] K. Brady, R. T. Calle, J. Sels, D. O'Cualain, and A. Bahri, "Pedestrian crossing and/or traffic light control method and apparatus," US10269239B2, Apr. 23, 2019.

[7] K. Fitzpatrick and E. S. Park, *Safety effectiveness of the HAWK pedestrian crossing treatment*. McLean, VA: Turner-Fairbank Highway Research Center, 2010.

[8] *Texting While Driving: Creating the Perfect Storm of Distraction*. New York, N.Y: Infobase, 2014.

[9] S. Stachura, “Why Pedestrian Deaths Are At A 30-Year High,” *NPR.org*, 28-Mar-2019. [Online]. Available: https://www.npr.org/2019/03/28/706481382/why-pedestrian-deaths-are-at-a-30-year-high. [Accessed: 16-Feb-2020].

[10] “PIR (motion) Sensor” Adafruit, Dec. 01, 2006, Accessed: Mar. 05, 2021. [Online]. Available: https://www.adafruit.com/product/189.

[11] “Maxbotix Ultrasonic Rangefinder - LV-EZ4 - LV-EZ4,” *Adafruit*. https://www.adafruit.com/product/982 (accessed Sep. 29, 2020).

[12] “PIC24FJ128GA010 Family 64/80/100-Pin, General Purpose, 16-Bit Flash Microcontrollers.” Microchip Technology, Inc., 2012, Accessed: Sep. 01, 2020. [Online].

[13] “Explorer 16/32 Development Board User’s Guide - Developer Help.” https://microchipdeveloper.com/boards:explorer1632 (accessed Oct. 04, 2020).

[14] “LaWriter Ohio Laws and Rules, 4511.46 Right-of-way of pedestrian within crosswalk.” http://codes.ohio.gov/ (accessed Oct. 04, 2020).

[15] “LM555 datasheet.” LM555 datasheet | TI.com. https://www.ti.com/document-viewer/LM555/datasheet. (accessed Apr. 21, 2021)

[16] “Audio Streaming,” Bluetooth® Technology Website. https://www.bluetooth.com/learn-about-bluetooth/solutions/audio-streaming/ (accessed Oct. 04, 2020).

[17] “Benefits of Zigbee,” Zigbee Alliance. https://zigbeealliance.org/why-zigbee/ (accessed Oct. 04, 2020).

[18] E. Products, “Bluetooth vs Wi-Fi vs ZigBee,” Electronic Products, Aug. 23, 2017. http://www.electronicproducts.com/bluetooth-vs-wi-fi-vs-zigbee/ (accessed Oct. 04, 2020).

[19] “Zigbee,” Zigbee Alliance. https://zigbeealliance.org/solution/zigbee/ (accessed Oct. 04, 2020).

[20] "LM317," LM317 3-Terminal Adjustable Regulator. https://www.ti.com/lit/ds/symlink/lm317.pdf?ts=1606134433267&ref_url=https%253A%252F%252Fwww.google.com%252F (accessed Oct. 25, 2020).

[21] "Interfacing xbee with PIC microcontroller using MikroC," *PIC Microcontroller*, Mar. 29, 2016. https://pic-microcontroller.com/interfacing-xbee-with-pic-microcontroller-using-mikroc/ (accessed Nov. 25, 2020).

[22] "XCTU." https://www.digi.com/products/embedded-systems/digi-xbee/digi-xbee-tools/xctu (accessed Nov. 25, 2020).

[23] https://mutcd.fhwa.dot.gov/htm/2003r1/part2/part2a.htm#section2A08 (accessed Nov. 25, 2020)

12. Appendices (AG, ST, MA, JS)

PIR (motion) Sensor [10]:

Block Diagram/Circuit

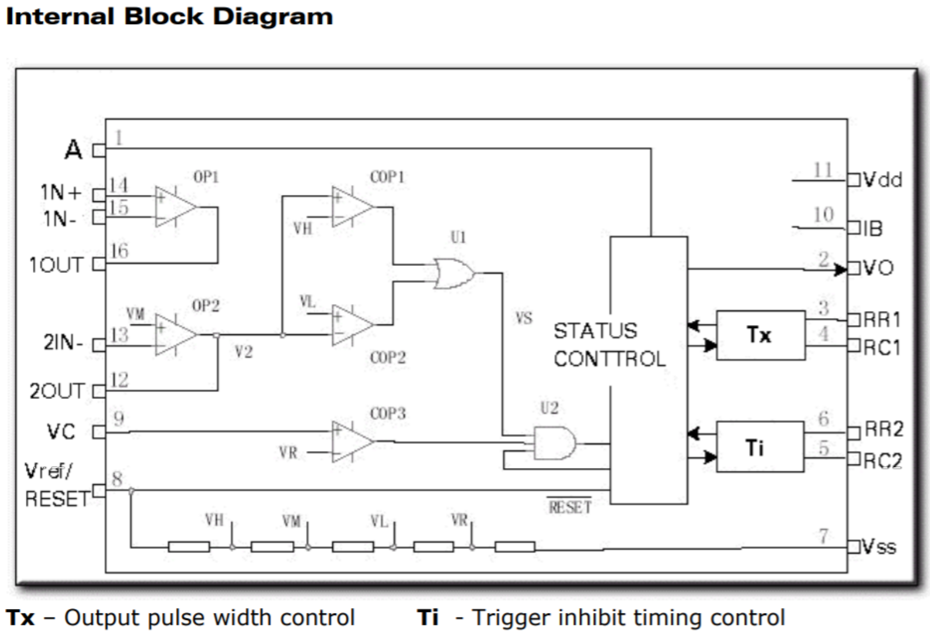

Figure 19: Block Diagram for PIR (motion) Sensor

PIR (motion) Sensor Circuit Configuration:

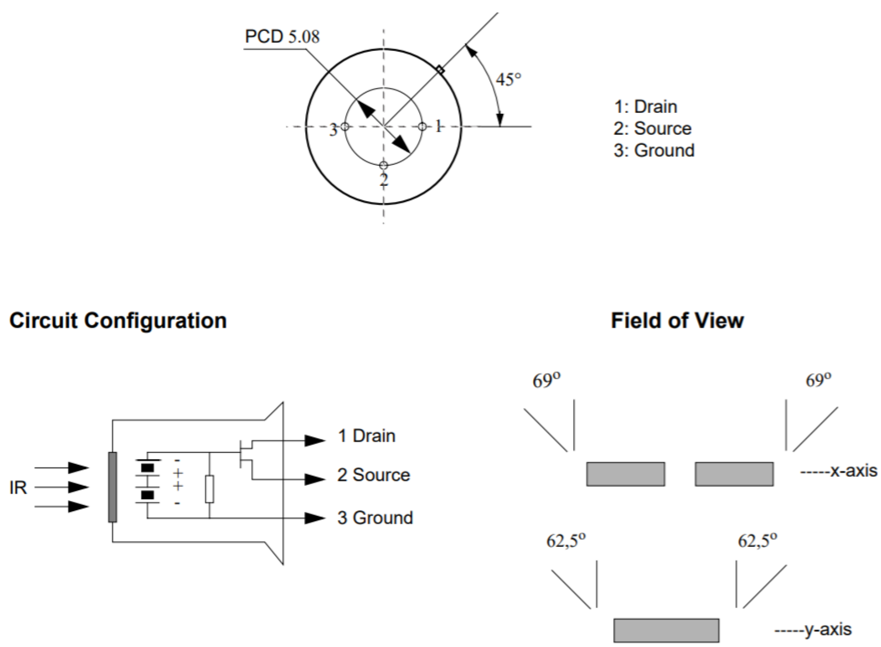

Figure 20: PIR (motion) Sensor Circuit Configuration

Maxbotix Ultrasonic Rangefinder - LV-EZ4 Series [11]:

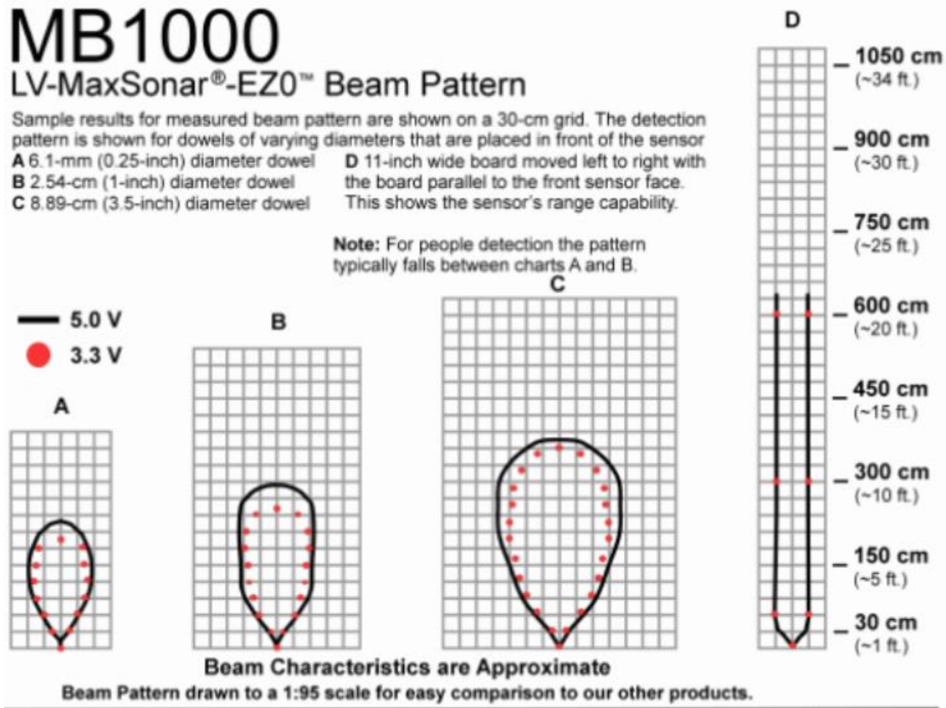

Figure 21: Ultrasonic Sensor Specification

PIC24FJ128GA010 100 Pin Layout [12]:

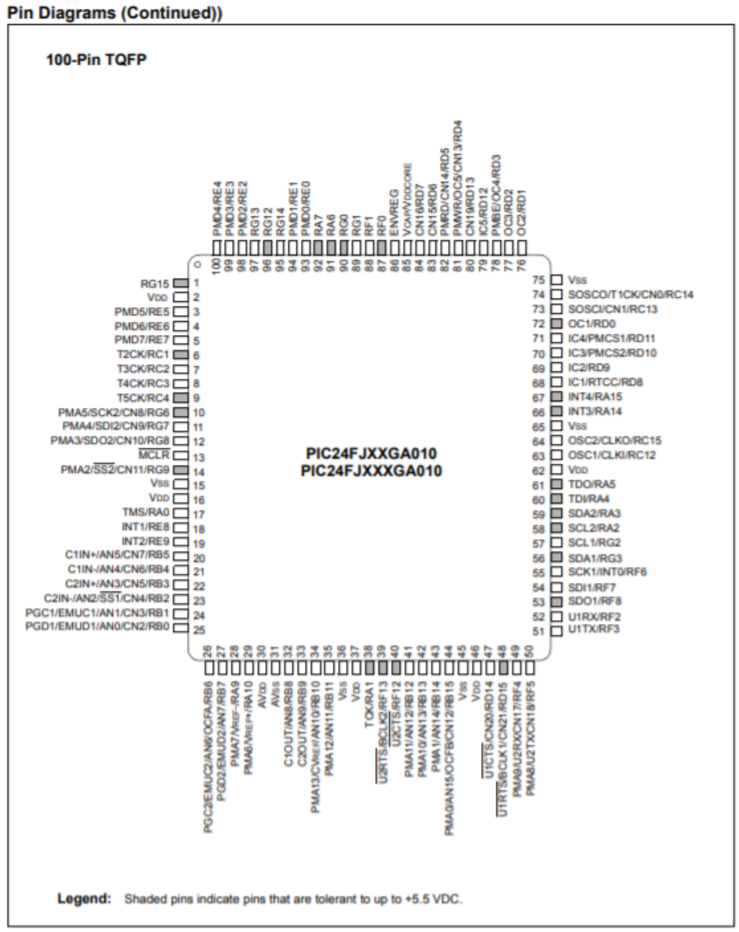

Figure 22: Microcontroller Pin Configuration

Explorer 16/32 Development Kit [13]:

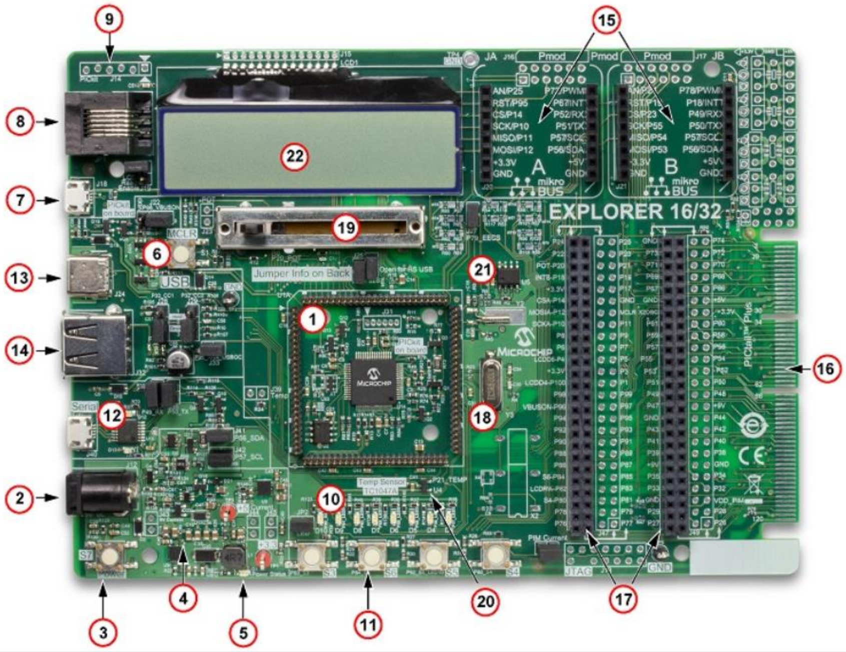

Figure 23: Explorer 16/32 Board