SUNLINK

8888 University Drive, Burnaby, BC V5A 1S6
Phone: 778-235-4757 Email: ensc440-sunlink@sfu.ca

November 12th, 2015

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, British Columbia
V5A 1S6

**Re: ENSC 305W/440W design specification for a solar real time transit display system**

Dear Dr. Rawicz,

Enclosed is our *Design Specification for a Solar Real Time Transit Display System,* proposed by Sunlink. Sunlink's goal is to design a solar powered device to be mounted at any existing bus stop that will provide pertinent transit information such as accurate next bus arrival times. We aim for our device to be inexpensive, reliable, and easy to install.

The purpose of the Design Specification is to explain in technical detail the design of Solarity. This includes system specifications, design of our system, and a functional test plan for our product.

The founding partners of our company, Zachary Kaarvik, Rohan Thomas, Dejan Jovasevic, Karen Ly-Ma, Rob Cornall and Tim Nguyen would like to personally thank you for your interest in our proposal. For any reason, feel free to contact us at ensc440-sunlink@sfu.ca.

Sincerely,

**Zachary Kaarvik**
CEO
Sunlink

***Enclosure: Design Specification for a Solar Real Time Transit Display System***

# DESIGN SPECIFICATION

## SOLAR REAL TIME TRANSIT DISPLAY SYSTEM

SUNLINK INC.

Issue Date: November 12, 2015
Revision: 1.7

**Contact Person:** Zachary Kaarvik
zkaarvik@sfu.ca

**Submitted To :** Dr. Andrew Rawicz-ENSC 440W
Steve Whitmore -ENSC 305W
School of Engineering Science
Simon Fraser University

**Date Issued:** November 12, 2015

# ABSTRACT

This document details the design specifications for Sunlink's product Solarity. The document details each individual component that will be used in the prototype. Our document includes hardware, software, and mechanical sections which step through our design process and justify the choices made for our designs.

Sunlink strives to implement an energy efficient and cost-effective solution for easier and more reliable commuting. Our product will serve transit users at all bus stops by providing real-time information on busses servicing any given stop. Solarity is entirely solar powered, requiring no external power source or additional infrastructure. Solarity receives the real time bus data wirelessly and presents the information on an e-paper display.

The design specification for Solarity shows the processes for the design of our proof-of-concept model, and focuses on the technical details of the hardware, software, and mechanical designs. A comprehensive test plan is also included to help ensure the components are fully functional as well as guarantee the functionality of the overall system.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# GLOSSARY

| | |
|---|---|
| 2G | Second-Generation Wireless Telephone Technology |
| A | Amperes |
| API | Application Program Interface |
| AT | Attention |
| AVL | Automatic Vehicle Location |
| CC | Charge Capacity |
| CSA | Canadian Standards Association |
| DCS1800 | Digital Cellular System, an frequency band from 1710 to 1784 MHz |
| EGSM900 | Extended Global System for Mobile Communications, an extended frequency band of GSM from 880 to 960 MHz |
| GPRS | General Packet Radio Service |
| GSM | Global System for Mobile Communications |
| GSM850 | Global System for Mobile Communications, an frequency band from 824 to849 MHz |
| HTTP | Hyper Text Transfer Protocol |
| I/O | Input/Output |
| IEEE | Institute of Electrical and Electronics Engineers |
| JSON | JavaScript Object Notation |
| LAMP stack | Linux, Apache, MySQL, and PHP web stack |
| LED | Light-emitting diode |
| MEAN stack | MongoDB, Express.js, Angular.js, and Node.js web stack |
| PCB | Printed Circuit Board |
| PCS1900 | Personal Communications Service, an frequency band from 1850 to1909 MHz |
| PV | Photovoltaic |
| RAM | Random Access Memory |
| RoHS | Restriction of Hazardous Substances |
| RS232 | Standard for serial communication |
| RTTI | Real Time Transit Information |
| SIM | Subscriber Identification Module |
| SPI | Serial Peripheral Interface |
| TransLink | Corporation Responsible for Metro Vancouver's Transportation Network |
| TTL | Time To Live |

# 1 INTRODUCTION

The Solarity Real Time Transit Display System is a relatively low cost system that easily mounts onto transit stop poles for the use of public transit commuters. The display system provides a convenient and precise method for checking the real time waiting durations of a given transit vehicle passing through that respective transit stop. It is compatible with information and data provided by TransLink. This document contains extensive design details that Sunlink will implement in the Solarity prototype.

## 1.1 SCOPE

This document supports the previous *Functional requirements* document for the Real Time Transit Display System. It provides a detailed description of design specifications for microcontroller and GSM module software, hardware and housing designs, and the functional test plan. This document will outline design decisions and discuss in detail our design process.

## 1.2 INTENDED AUDIENCE

The design specification document is intended to be used by the members of the Sunlink team throughout the design and development stages to ensure that the design adheres to the required specifications. This document will also be used to measure project progress and to aid in user documentation and test plan creation.

# 2 SYSTEM SPECIFICATION

## 2.1 USAGE

The majority of user interaction with Solarity will be by transit customers who are waiting at a transit stop with Solarity installed. These customers can view the information presented on Solarity's display regarding next bus schedules and other pertinent information about the transit system. Our aim is to have our device as simple as possible to understand for a transit user, and as such we do not require input from an end user. The user will solely need to observe the information from the display which is updated automatically.

Solarity has a proximity sensor to tell when people are nearby. When the transit customer is near the device their presence is detected and if the device is in sleep mode it will awaken and retrieve fresh data. When no users have been detected by the system for an appropriate length of time it will re-enter power saving sleep mode.

The other main, but less common, usage scenario is that of the person configuring Solarity. The person configuring the device must be able to input the number or ID of the transit stop that Solarity is installed at in order for the device to display correct stop data.

## 2.2 HIGH LEVEL DESIGN

Solarity consists of four subsystems.

The first subsystem involves the interface of bus numbers and times onto a low power e-paper display. The bus numbers and times are retrieved from a backend server which obtains the data through Translink's RTTI Open API system. Solarity's backend server sends the data to the device through a cellular data line. This cellular connection is provided by a GSM/GPRS modem which the microcontroller can use to access the internet. The microcontroller will be interfaced with a display where the information will be presented in a user friendly format.

The second subsystem is the power module. The system will require a battery large enough to power the microcontroller, the GSM module and the display throughout the daytime and nighttime. The battery capacity will have a high dependency on the active time of the device and will be charged continuously with a solar panel mounted on top of the bus pole. A battery charge controller will be wired between the batteries and the solar panel to prevent overcharging and pose as a safety measure.

The third subsystem utilizes sensors. In order to minimize the power consumption, proximity sensors will be added to the system to help detect when transit users are nearby. Solarity will generally remain in sleep mode during the night when busses are no longer in service and traffic is low. In addition, Solarity requires a light sensor. The e-paper display does not contain a backlight and will be difficult to see with low ambient light. The light sensor will be used to track when to turn on the backlight for the screen to aid with visibility.

The fourth subsystem involves Solarity manager backend server. Each Solarity device will be assigned a unique ID and a bus stop location for installation. The backend server will store the locations of all installed Solarity devices and the corresponding device ID. When a request is sent, the server will look up the ID and location and relay the appropriate bus numbers and times back to the device.

Figure 1 shows a block diagram of the overall architecture of Solarity.

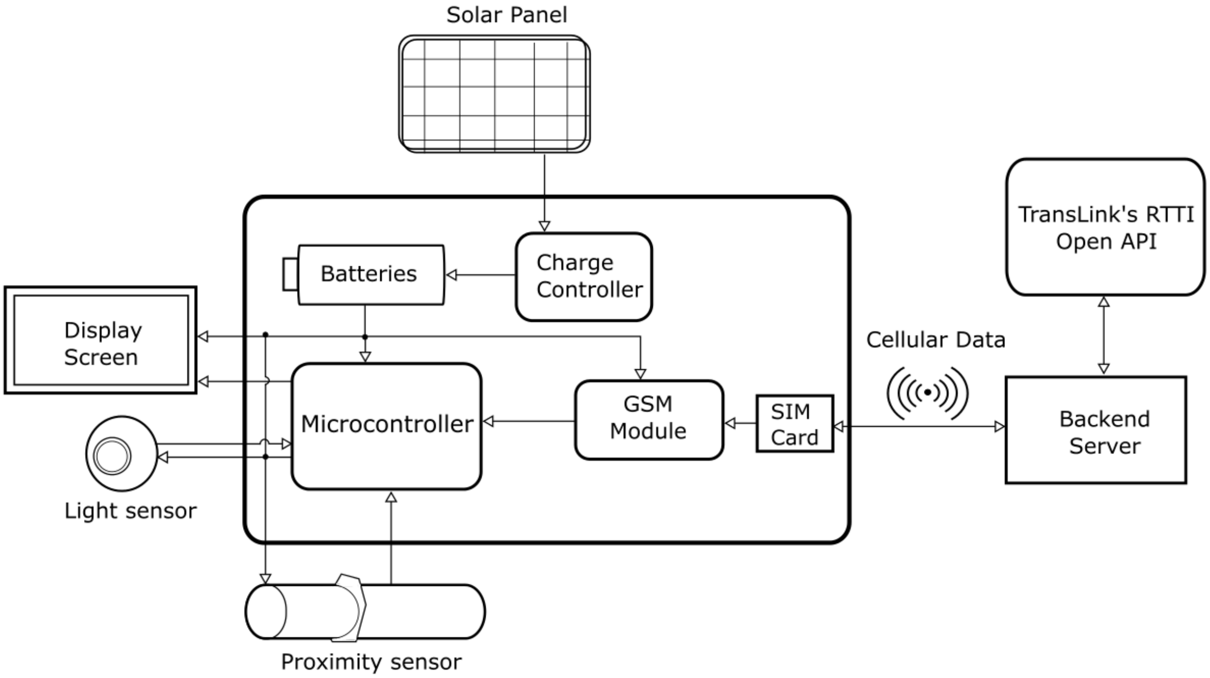


Figure 1: The block diagram of the overall architecture of Solarity.

# 3 DESIGN REQUIREMENTS

## 3.1 HARDWARE DESIGN

This section will identify the different hardware components that will be used in our product. These components have been manufactured by third party sources and will be integrated into our system. Our project utilizes the following hardware components:

- MSP432P401R Launchpad
- Sim800c GSM shield
- MPico Pervasive Display
- SparkFun Ambient Light Sensor (TEMT6000)
- LV-MaxSonar-EZ0 Proximity Sensor
- Battery Unit
- Solar Panel

The following sections will cover the use of each component, a technical description, and how they integrate into Solarity.

### 3.1.1 Electrical Design

The minimum battery capacity required for Solarity is equal to the summation of charge capacity of each individual subsystem. The subsystems are the GSM module, microcontroller, pervasive display, sensors, and LED. To calculate the charge capacity of each of these units in milliamp hours, the following equation is used,

$$Min\ Subsytem\ CC = I_{MAX}(A) \times Active\ time\ (h) \times (\#of\ times\ on\ in\ 1\ hour \times active\ hours\ per\ day) \quad (1)$$

The first subsystem is the GSM module which will utilize three modes of operation: data mode, power down mode and idle mode. The maximum current consumption, number of times active in an hour and active hours per day for these modes are summarized in Table 6.

Table 1: Max Current Consumptions of GSM Module

| Conditions | Max Current Consumption | # of Times Active per hour | Active hours per day |
|---|---|---|---|
| Idle mode | 18.7 mA | 30 | 18 |
| Data mode | 453.57 mA | | |
| Power down mode | 60 µA | | |

Solarity will send a request to Translink’s Open API to obtain updated transit information every two minutes. This refresh rate is frequent enough to keep transit users aware of real time bus information but may be adjusted to a slightly slower rate for further power efficiency. The microcontroller sends AT commands to request data, the GSM module is in idle mode while waiting for the requested data from the server. Once the data is received the GSM goes into data mode while sending the information back to the microcontroller. To conserve energy, the module is then powered down once the transaction is complete. Utilizing Table 6 and equation (1), the minimum charge capacity per day of the GSM is calculated as follows,

$$\begin{aligned} &Minimum\ CC\ of\ GSM = Idle\ mode_{CC} + Data\ mode_{CC} + Power\ Down\ Mode_{CC} \\ &= 453.57\ (mA) \times \left(\frac{6}{3600}(h) \times 30 \times 18\right) + (0.060\ (mA) \times \left(\frac{3584}{3600}(h) \times 30 \times 18\right) \\ &\qquad + 18.7(mA))(\frac{10}{3600}(h) \times 30 \times 18) \\ &= 468.52mAh \end{aligned} \tag{2}$$

The second subsystem is the MPico Pervasive Display. The display operates in two modes: refresh mode and receiving data through the SPI bus. The maximum current consumption, number of times active in an hour and active hours per day for these modes are summarized in Table 2.

Table 2: Max Current Consumptions of Display

| Conditions | Max Current Consumption | # of Times Active per hour | Active hours per day |
|---|---|---|---|
| Display refresh | 108 mA | 30 | 18 |
| Data reception with SPI | 24 mA | | |

The display will refresh to a new image with updated bus times every two minutes. With each update, the display needs to go into receiving data mode and then refresh mode. Utilizing Table 2 and equation (1), the minimum charge capacity of the display per day is calculated as follows,

$$\begin{aligned} Minimum\ CC\ of\ Display &= Display\ Update_{CC} + Data\ Reception\ with\ SPI_{CC} \\ &= \big(108\ (mA) + 24\ (mA)\big) \times \left(\frac{10}{3600}(h) \times 30 \times 18\right) \\ &= 198mAh \end{aligned} \tag{3}$$

The third subsystem is MSP432P401R microcontroller. The microcontroller operates in two modes, active and standby mode.

Table 3: Max Current Consumptions of Microcontroller

| Conditions | Max Current Consumption | # of Times Active per hour | Active hours per day |
|---|---|---|---|
| Active mode | 90 µA/MHz | 30 | 18 |
| Standby mode | 0.85 µA/MHz | | |

Utilizing *Table 6* and equation (1), the minimum charge capacity of the display per day is calculated as follows,

$$\begin{aligned} &\mathit{Minimum\ CC\ of\ Microcontroller} = \mathit{Active\ mode}_{CC} + \mathit{Standby\ mode}_{CC} \\ &= \left( (0.090 \left(\frac{mA}{MHz}\right) \times 48(MHz)) + 0.00085\ (mA) \right) \times \left(\frac{10}{3600}(h) \times 30 \times 18\right) \qquad (4) \\ &= 6.48\ mAh \end{aligned}$$

The fourth subsystem is the sensors. The first sensor the system will use is a LV-MaxSonar–EZ0 ultrasonic sensor that will be on continuously throughout the day to wake the system up when a transit user is nearby. The light sensor will sample the outdoor conditions every two minutes.

Table 4: Max Current Consumption of Proximity and Light Sensor

| Sensor | Conditions | Max Current Consumption | # of Times Active per hour | Active hours per day |
|---|---|---|---|---|
| Proximity | Active mode | 2 mA | Continuous | Continuous |
| Light | Active mode | 2.5 mA | 30 | 18 |

Utilizing Table 4 and equation (1), the minimum charge capacity of the display per day is calculated as follows,

$$\begin{aligned} \mathit{Minimum\ CC\ of\ Proximity\ sensor} &= \mathit{Active\ mode}_{CC} \\ &= 2\ (mA) \times 24h \qquad (5) \\ &= 48\ mAh \end{aligned}$$

$$\begin{aligned} \mathit{Minimum\ CC\ of\ Light\ sensor} &= \mathit{Active\ mode}_{CC} \\ &= 2.5\ (mA) \times \left(\frac{10}{3600}(h) \times 30 \times 18\right) \qquad (6) \\ &= 3.75\ mAh \end{aligned}$$

Table 5: Max Current Consumptions of LED

| Conditions | Max Current Consumption | # of Times Active per hour | Active hours per day |
|---|---|---|---|
| LED | 20mA | 30 | 18 |

$$\begin{aligned} &\mathit{Minimum\ CC\ of\ LED} = \mathit{Active\ mode}_{CC} \\ &= 20\ (mA) \times \left(\frac{10}{3600}(h) \times 30 \times 18\right) \qquad (7) \\ &= 30\ mAh \end{aligned}$$

The total charge capacity of the overall system is the sum of the charge capacity of the GSM, display, microcontroller, sensors and LED as seen in equation (4).

$$\begin{aligned} \mathit{CC\ of\ System} &= GSM_{CC} + Display_{CC} + Microcontroller_{CC} + Sensor_{CC} + LED_{CC} \\ &= 468.52mAh + 198mAh + 48mAh + 3.75 + 30 \qquad (8) \\ &= 748.27mAh \end{aligned}$$

The overall system will utilize a lithium ion battery with a capacity of 6600 mAh. Without any charging, the system can be powered on for 8.82 days with the battery at full capacity.

During the winter, since sunlight is limited, the solar panel will be tilted to maximize the most amount of energy using the equation below [1]

$$\begin{aligned} Optimum\ Angle\ of\ Tilt\ for\ the\ winter &= (Latitude\ of\ Vancouver \times 0.9) + 29^\circ \\ &= (49^\circ \times 0.9) + 29^\circ \\ &= 74^\circ \end{aligned} \tag{9}$$

To minimize maintenance costs, the solar panel will be at a fixed tilt. Since winter has the least sunlight, setting the tilt angle to be ideal for this season would make it compatible for all other seasons [1].

Solarity will use a 12V 20W solar panel. Field tests have shown that a loss of a factor of 2.5 occurs with charging of the battery with solar panels. The drop in voltage of the panel when connected to a load and heat dissipation are contributions to the loss factor. This loss factor increases even more with other non-ideal conditions [2]. For the estimation of time to charge Solarity's battery, the loss factor will be doubled to 5 in order to take into account cloudy days. The amount of time the solar panel will take to charge the 6600 mAh is follows,

$$\begin{aligned} &Estimated\ Time\ to\ Charge\ Battery \\ &\quad = \frac{Battery\ Voltage\ (V) \times Battery\ Charge\ Capacity\ (Ah)}{Power\ of\ Solar\ Panel\ (W)} \times 5 \\ &\quad = \frac{3.7\ (V) \times 6.6(Ah)}{20W} \times 5 \\ &\quad = 6.10\ hours \end{aligned} \tag{9}$$

### 3.1.2 Charging Design

The main goal of the charging module is to maximize the battery capacity by continually restoring energy back into the batteries. For a solar application, the most effective method to charge lithium ion batteries is through switch mode charging with the low power dissipation and higher charge current compared to a linear solution. Switch mode charging is useful for applications with varying input voltage levels [3]. Since the voltage levels from the solar panels will change drastically throughout the day depending on the amount of sunlight available, switch mode charging would be most effective.

To further increase the efficiency of charging a MPPT method will be employed in the system. MPPTs are commonly used for power optimization. MPPT compares the output voltage of the solar panel to the voltage of the battery in order to determine the most ideal voltage level to maximize the input current going into the battery [4].

Solarity will use the Sunny Buddy MPPT Solar Charger that utilizes Linear Technology LT3652 Power Tracking 2A Battery Charger for Solar Power. The LT3652 supports switching mode and uses a MPPT algorithm to maximize the battery charging. The solar charger is defaulted to have a charging rate of 450 mAh [5]. To increase the charging rate, the sense resistor will be modified using equation (1).

$$R_{sense} = \frac{0.01}{I_{CHG(MAX)}}(\Omega) = \frac{0.01}{2A} = 0.05\ \Omega \qquad (1)$$

Changing the current charging rate requires a few modifications to the charge controller. The inductor needs to be able to handle a saturated current of 2A as seen in equation (2).

$$L_1 = \frac{10R_{sense}}{delta\ I_{MAX}} \times V_{bat(FLT)} \times \left[1 - \left(\frac{V_{BAT(FLT)}}{V_{IN(MAX)}}\right)\right](\mu H) \qquad (1)$$

In addition the diode, D1, needs to be replaced to support a rating of 2A.

During the 2A bursts from the GSM module, the charge controller will try to directly provide the current to the load and the rest will be from the battery. Since the system uses a battery with a capacity greater than 2A then these infrequent high current bursts from the GSM module are already handled regardless of the charge controller's charging current rating.

### 3.1.3 Sensor Design

#### *3.1.3.1 Proximity Sensor*

An ultrasonic sensor will be used to trigger the system when users are nearby. The ultrasonic sensor is chosen over the more inexpensive option of using an infrared sensor since the sensor will be used outdoors. Since infrared sensors detect light radiation, the sensor is susceptible to indirect and direct sunlight. Ultrasonic relies on sound for detection at a certain range and is more ideal for outdoors. Unlike infrared sensors, ultrasonic sensors also have a wider beam which allows for a wider area of detection around the device [6].

The ultrasonic sensor that will be used for proximity detection for Solarity is the LV-MaxSonar–EZ0. The EZ0 is ideal for people detection having the most sensitive and widest beam range of the LV-MaxSonar EZ sensors. This sensor operates at 42 KHz which is quick enough for the application of detecting nearby users [7].

#### *3.1.3.2 Light Sensor*

The light sensor that will be used for this system is a TEMPT6000 Ambient Light sensor that utilizes a TEMPT6000 phototransistor. These light sensors are inexpensive and respond well to low and high light level transitions. The amount of voltage at $V_{A3}$ seen in Figure 2 is dependent on the amount of light seen at the base of the phototransistor. By altering the $V_{CC}$ rail and the resistor value, a wide range of voltage levels can be seen at $V_{A3}$ which in turn means a wide range of light intensities can be monitored. Since the sensor will be only be used to track the outdoor light conditions, quick response time is not necessary as natural ambient light changes gradually [8].

To reduce the power consumption, a bright white LED will be used to provide lighting to the screen.

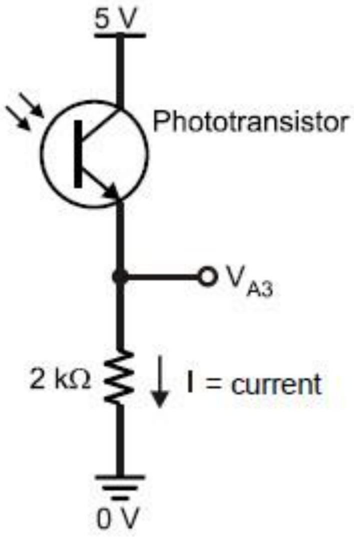


Figure 2: Schematic of the TEMPT6000 Ambient Light Sensor [8]

### 3.1.4 GSM Module Design

Solarity will use the SIM800c GSM module to communicate with the backend server mentioned above. Communication with the GSM will occur via buses connecting the receiving and transmitting pins on both the microcontroller and GSM module. The microcontroller will program the GSM module with specific AT commands that will tell the GSM to request data from the Solarity Server. The GSM will access the web address requesting for image data to be sent back. The GSM will transmit the image data from the transmit pin to the receive pin of the microcontroller. From here the image data is moved to the display.

This specific GSM module was chosen for a number of reasons. Firstly, the module has a low power mode that that will be used whenever not in action. When in this mode the module only requires 60μA and thus fits perfectly into our low power design. The SIM800c also supports 2G and 3G networks which is what Solarity needs to use the GPRS data services to be able to communicate with our server. This module has the ability to understand the specific AT commands Solarity needs to be able to transfer data from the server to the display using the GSM. The GSM chip and the GSM Shield which it is soldered on are relatively small in size so they do not restrict the components of Solarity from being compactly placed inside the enclosure.

### 3.1.5 Cellular Antenna Design

In order for the GSM module to maintain a strong network connection, the antenna must support the same frequencies band as the SIM800c, have a SMA connector and have a reasonable small length to be enclosure in the case. The quad-band cellular duck SMA antenna would be used with the GSM as it supports the same frequencies band as the SIM800c. As well as the antenna have a length of 58 mm, therefore the antenna can be stored inside the enclosure.

Figure 3: Quad-Band Cellular Duck SMA Antenna *[9]*

### 3.1.6 Microcontroller Design

Solarity requires a microcontroller that is capable of controlling the display, the GSM module, and the sensors. Specifically, the input/output requirements of our microcontroller are listed in Table 6.

Table 6: Microcontroller I/O Minimum Requirements

| Component | I/O Requirements |
|---|---|
| Display | Three pin SPI module<br>Digital output for enable pin<br>Digital output for chip select pin<br>Digital input for busy pin |
| GSM Module | Two pin UART module |
| Light Sensor | Analog input pin |
| Proximity Sensor | Digital input pin |

Further, we require a device capable of very low power operation in order to sustain long periods of battery operation.

The TI MSP432 Launchpad microcontroller accommodates our requirements and has additional features to allow for faster development and debugging. The Launchpad features a 48MHz 32-bit MSP432P401R processor with 64kB of RAM and 40 exposed input/output pins [10]. The MSP432P401R has eight SPI modules and four UART modules which exceeds our minimum requirements. With the Launchpad's integrated debugger we can reprogram the device on the fly through TI's Code Compiler Studio IDE. This is allowing us to rapidly implement our device's firmware. Consuming 4.56mA when running at full speed in active mode, the Launchpad has an advantage over comparable microcontrollers such as the Teensy (27.3mA) [11] and the Arduino Zero (20mA) [12].

Our microcontroller will interact with each of the components connected to it. It will control the data flow to and from the display and GSM modules, as well as monitor the input from the sensors. The microcontroller has available timers that enable us to enter a very low power sleep mode when the device is not performing any operations for a predetermined amount of time. For more detailed implementation information on each component please refer to the respective subsections.

*Figure 4* shows a high level overview of Solarity's firmware operation. In the figure we show an update time of every two minutes, this may change by prototype completion depending on how much power we find we must conserve for reliable operation in limited light conditions.

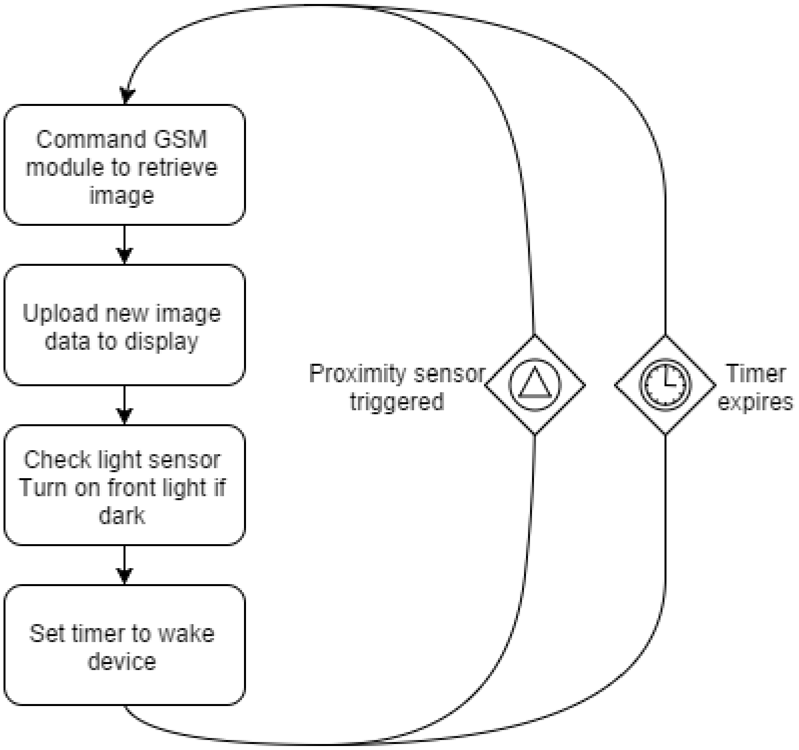


Figure 4: High Level Firmware Operation

### 3.1.7 Display Screen Design

In order to display the information to nearby users, it is necessary for our display system to satisfy a list of requirements. The Aurora Pervasive Display was chosen as our hardware display because it satisfied all these requirements as well as providing some other advantages.

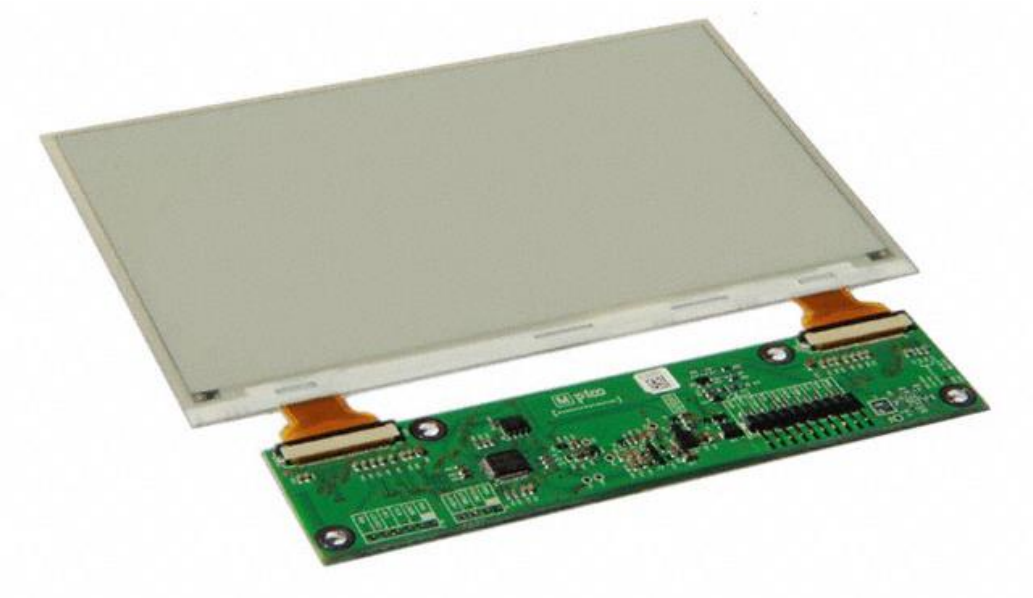


Figure 5: 7.4" MPico Pervasive Display [13]

First, it has all the image displaying capabilities that we require. It has the ability to display a stable image without constantly refreshing the screen when data input remains constant. This gives a lot of flexibility in terms of when the information is to be refreshed. The information provided to the display will come from the microcontroller through the SPI connection. As opposed to conventional screens that have problems with glare in the daytime and nighttime, it reflects ambient light rather than emitting its own light. Also, a low-power screen was chosen for the displaying hardware because we wish to demonstrate that our device does not require much power to operate, allowing for the system to be installed at multiple locations.

### 3.1.8 Wiring Design

Appendix C contains diagrams illustrating the wirings of the components in Solarity.

The four main modules of Solarity: The microcontroller, GSM module, sensors (proximity and light) and the charging circuit will be contained on a PCB board and wired appropriately. The wires and PCB board will be housed within the enclosure of the system and be well insulated (See Section 3.3.1).

## 3.2 SOFTWARE DESIGN

Solarity has two independent software components. The first component, the device firmware resides on Solarity's microcontroller and controls the device's interaction with the display, the GSM module, the charging circuit, and the sensors. The second software component, the Solarity manager server will store the registered devices and their assigned stop numbers in a database. The server will also have an API which each device will query for its image data. The database will be managed by way of a web interface which an authenticated manager can log in to and add, modify, or delete devices and their assigned stops. Figure  shows the interaction between the software components.

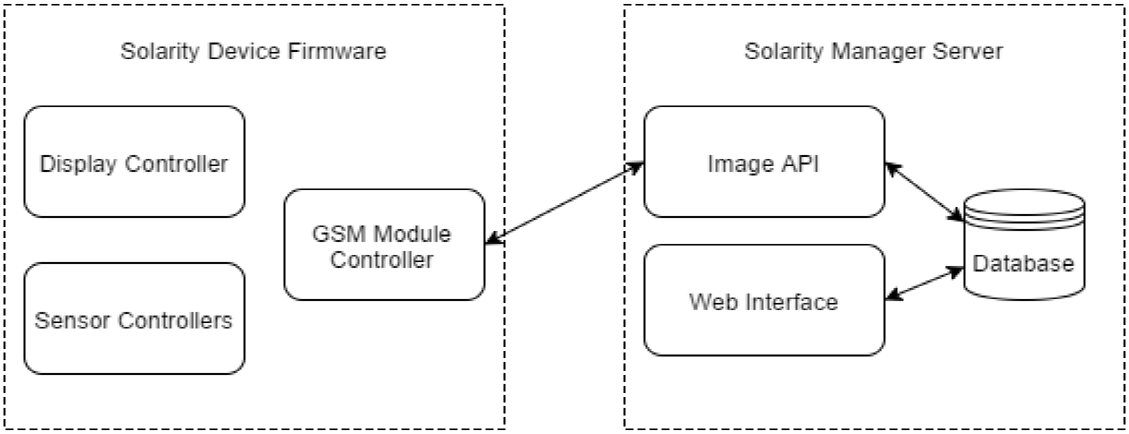


Figure 3: Solarity Software Interaction Diagram

### 3.2.1 Solarity Device (Client) Firmware Design

Solarity's firmware must accomplish three separate major functions in order to meet our requirements. First it must query the server over a cellular data connection to get image data. Once the device retrieves the image data it will send this data to the display controller. Solarity must also monitor input from its sensors, and enter a low power mode when it will be inactive for any length of time.

Solarity will connect to the cellular network by communicating with the SIM800c GSM module. Communication is performed over an UART interface running at a baud rate of 115200 bps. The microcontroller will control the GSM module by sending AT commands. When Solarity is turned on from an off state, the microcontroller will send an AT command to query the name of the network carrier. Once the name of the network carrier has been received, the microcontroller will send a list of AT commands to the GSM module to create a bearer service for that specific network operator.

To connect to the cellular networking when the device is querying the server, an AT command will be send to open the bearer service to get the GSM module to enter GPRS mode. The SIM800c has an embedded TCP/IP stack that is driven by AT commands and enables Solarity to access the internet HTTP service. Therefore, AT commands for HTTP application will be send to the GSM module to connect to the server and retrieve the information to be display. Relevant AT commands are listed in Table 7.

Table 7: AT Commands Description [14]

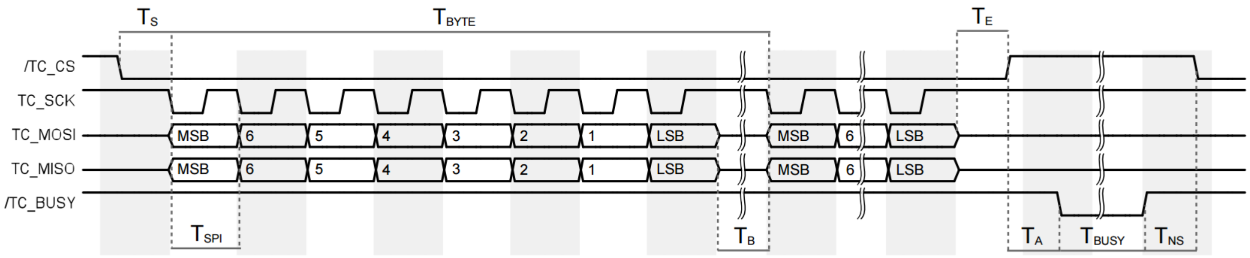
| Command | Description |
|---|---|
| AT+COPN | Query the operator Name |
| AT+CFUN=0 | GSM module enter to minimum functionality mode |
| AT+CFUN=1 | GSM module enter to full functionality mode |
| AT+CSQ | Query the strength of signal connection |
| AT+SAPBR=3,1,APN, “X” | Set up the APN of the bearer connection, where X is the name of apn of the operator |
| AT+SAPBR=3,1,USER,”X” | Set up the username of the bearer connection, where X is the username to connect to the APN of the operator |
| AT+SAPBR=3,1,PWD,”X” | Set up the username of the bearer connection, where X is the password to connect to the APN of the operator |
| AT+SAPBR =1,1 | Open bearer connection |
| AT+SAPBR =0,1 | Close the bearer connection |
| AT+HTTPINIT | Initialize HTTP service |
| AT+HTTPTERM | Terminate HTTP service |
| AT+HTTPPARA=URL,”X” | Setting the URL HTTP parameter, where X is the URL |
| AT+HTTPPARA=CID,1 | To use the bearer connection to connection to the request-URL |
| AT+HTTPACTION=0 | To retrieve whatever information is identified by the request-URL |
| AT+HTTPREAD | Read the HTTP server response |

Solarity will interact with its display by communicating with the display’s controller module. Communication is performed over an SPI interface running at 2MHz. After activating the controller enable pin and allowing the device up to 200ms to initialize, commands may be sent to the device. Image data is uploaded in chunks of 250 bytes, repeated until all image data has been uploaded. Relevant commands are listed in *Table 8*. Acceptable SPI byte transfer timing is shown in *Figure 6*.

Table 8: Display Command Description *[15]*

| Command | Description |
|---|---|
| 0x20<br>0x01<br>0x00<br>[packet size]<br>[image data] | Upload image data. Packet size may be a maximum of 250 (0xFA) bytes. Image data must be in EPD format. After each image upload command the internal data pointer is incremented by the packet size. This command should be repeated until the entire 48,016 byte image file is uploaded. |

| | |
|---|---|
| 0x24<br>0x01<br>0x00 | Display update. This command will update the display with the current contents of memory. If data was uploaded with the upload image data command then that image will appear on the display. If no new data was uploaded, the previous image will be refreshed. |



| Time | $T_S$ | $T_{BYTE}$ | $T_{SPI}$ | $T_B$ | $T_E$ | $T_A$ | $T_{BUSY}$ | $T_{NS}$ |
|---|---|---|---|---|---|---|---|---|
| Min. | 6.0 µs | | 82 ns | 0 | 11.0 µs | 5 µs | 21 µs | 2.0 µs |
| Typ. | | 2.67 µs | 333 ns | | | | | |
| Max. | | | 8 us | | | | | |

Figure 6: SPI Timing Diagram *[15]*

Finally, Solarity's firmware must obtain data from its light and proximity sensors and react accordingly. Light level is detected with a phototransistor, and an ultrasonic sensor is responsible for proximity detection.

The output voltage levels from the phototransistor will be monitored through an analog pin of the microcontroller. When the threshold voltage is reached due to low outdoor light levels, the LED will be powered on. Since the natural outdoor light transitions slowly, the microcontroller will only sample the voltage level of the phototransistor every two minutes. Sampling with intervals will help to limit the power consumption. By prototype completion we may reduce this interval further if we find the power savings outweigh the update frequency in practice.

The ultrasonic sensor has three different communication methods: analog voltage, RS232 or TTL serial, and pulse width. In our design we have chosen to read the data from the sensor through its analog voltage output. This method has lesser accuracy than reading the serial output, but the simplicity of implementation outweighs this minor inaccuracy. The microcontroller will convert the voltage on the analog pin to a value between 0 and 1023. This value is multiplied by 5mm to determine the distance of an object from the sensor [7]. When the distance to the nearest object changes we can detect the presence of a person and activate the device appropriately.

### 3.2.2 Solarity Manager (Server) Design

Solarity Manager will provide an interface to manage devices as well as generate images when individual Solarity devices request them over an HTTP API. The server will run on an Ubuntu virtual server hosted by DigitalOcean. Having our server hosted by a third party provides a reliable system to build and test our software at a minimal expense. The expense of the server is negated by education discounts provided by

DigitalOcean, and will not impact our budget for the duration of the prototype development.

The server will run a MEAN software stack instead of a more traditional LAMP stack, the components of which are detailed in Table 9. We have decided to use a MEAN stack for its increased flexibility, decreased development time, wide availability of software modules, and team member familiarity with JavaScript development.

Table 9: MEAN Stack Components *[16]*

| Component | Description |
|---|---|
| MongoDB | Schemaless database system. Data can be saved and retrieved in JSON format, allowing for easy integration in a JavaScript application. |
| Express.js | Framework to build web applications with Node.js. It provides many features for rapidly building both web applications and dynamic HTTP APIs. |
| Angular.js | JavaScript MVC framework for building web applications. Provides a complete solution for front end interface development. |
| Node.js | Server side JavaScript environment. Asynchronous and event driven. Provides a massive amount of open source libraries to extend functionality through its package manager, npm. |

The user interface will allow a manager to register, modify, and delete Solarity devices and their assigned transit stop IDs. In our prototype this interface will be implemented as a simple table with the device ID, assigned stop number, and last access time of the device.

The database will consist of a single table to retain Solarity device information. It will store the device ID, the assigned transit stop number, and the last device request time. In the prototype version, the database server will be running on the same virtual machine as the rest of the web stack.

The image API will be triggered by each Solarity device in order to retrieve the image which will be displayed on Solarity's screen. The logical flow of the image API is shown in Figure 7. In order to create the image we are using the Node.js library gm [17]. The image format required by the display controller is a proprietary format termed EPD, which represent each pixel with a single bit. The server will send the response in this format [15].

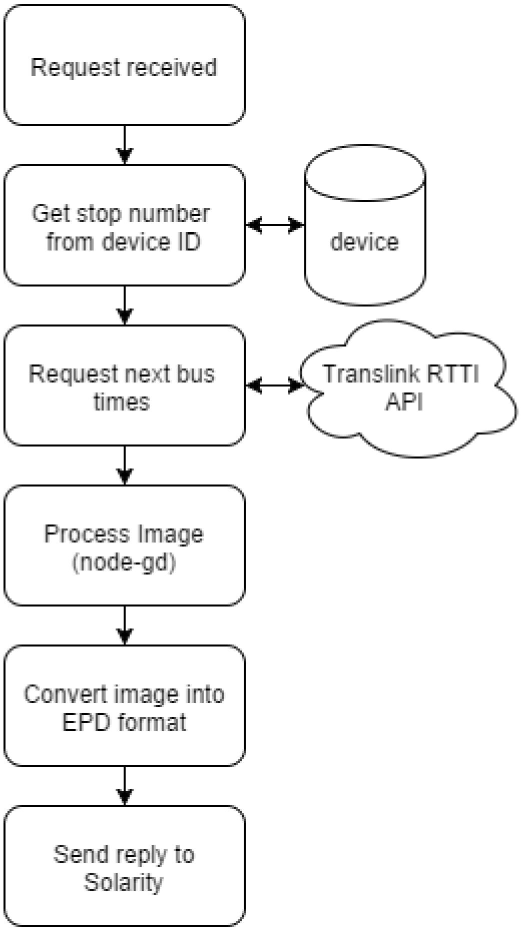


Figure 7: Image API Logic Flow

## 3.3 MECHANICAL DESIGN

This section of the document outlines the major mechanical components used in the Sunlink Solarity system. Various features and specifications are given for each component along with a rationale for why these specific components were selected for the design.

### 3.3.1 Enclosure Design

The enclosure will be designate as a type 3R enclosure as specified by C22.2 No.94-M91: Special Purpose Enclosure. See Appendix A for enclosure type definitions. The enclosure is designed to be protected against rain, snow and be will undamaged by external formation of ice on the enclosure. Figure 8 provides an isometric view of the intended enclosure. For more views and low level design please refer to Appendix B.

Figure 8: Isometric View of Intended Enclosure

The front of enclosure has openings for display screen, as well as for the light and proximity sensors and roof for the LED, as shown in *Figure 9* below.

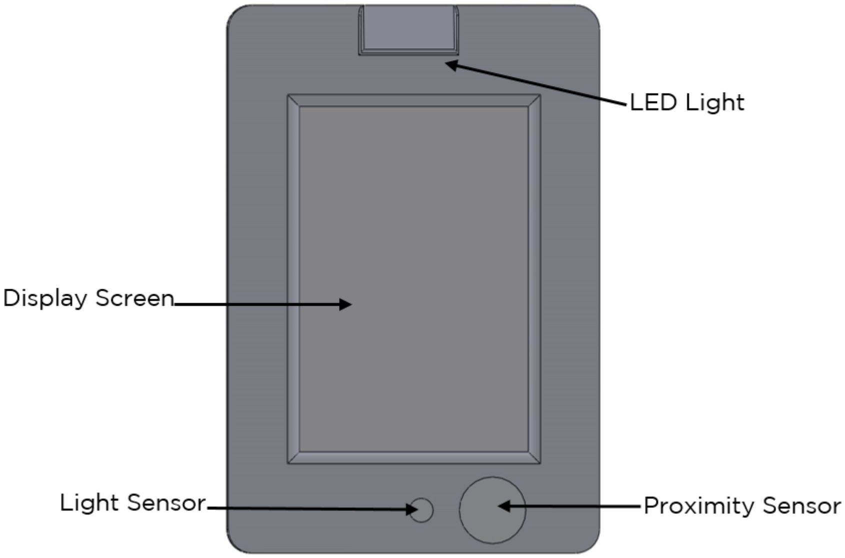


Figure 9: Front view of Intended Enclosure

The display screen and sensors will be behind a polycarbonate material. Polycarbonate was chosen as the material because it is a transparent, strong and stiff thermoplastic material with outstanding impact resistance. Polycarbonate's toughness and optical clarity make it ideal to protect the screen from outside damages, as well as it does not block visibility of the display and sensors. Polycarbonate has a low water absorption index of 0.16, which corresponds to a 0.35% weight increase when immersion in water for 24 hours [18]. This will protect the display screen and sensors from rain and snow weather.  Table 10Table 10: Typical Properties of Polycarbonate list the typical properties of polycarbonate.

Table 10: Typical Properties of Polycarbonate [18]

| | **Units** | **Polycarbonate** |
|---|---|---|
| Tensile Strength | Psi | 9500 |
| Flexural modulus | Psi | 345,000 |
| Izod impact (notched) | Ft-lbs/in of notch | 12.0-16.0 |
| Heat deflection temperature @ 264 psi | °F | 270 |
| Maximum continuous service temperature in air | °F | 240 |
| Water absorption (immersion 24 hours) | % | 0.16 |
| Coefficient of linear thermal expansion | in/in/°Fx$10^{-5}$ | 3.8 |

The enclosure will be made of Kydex® thermoplastic sheet, which is a proprietary thermoplastic sheet material that has excellent aesthetic qualities, superior impact resistance, and outstanding formability. Other key characteristics of the material includes good electrical insulating properties, consistent color throughout material to help hide scratches and wear, and a very low water absorption index to help water proofing the device [18]. Table 11 list Kydex® thermoplastic sheet typical properties.

Table 11: Typical Properties of Kydex® Thermoplastic Sheet [18]

| | **Units** | **Kydex 100** |
|---|---|---|
| Tensile Strength | Psi | 6100 |
| Flexural modulus | Psi | 335,000 |
| Izod impact (notched) | Ft-lbs/in of notch | 18.0 |
| Heat deflection temperature @ 264 psi | °F | 173 |
| Maximum continuous service temperature in air | °F | - |
| Water absorption (immersion 24 hours) | % | 0.05-0.08 |
| Coefficient of linear thermal expansion | in/in/°Fx10$^{-5}$ | 4.2 |

For the prototype, the enclosure will be made of ABS to provide proof-of-concept.

The enclosure is designed in a way to make it harder for the device to be removed and stolen by non-authorized personal. The enclosure has a hole diameter of 58 mm to wrap around a 60 mm diameter pole, as shown in Figure 10: Top view of Intended Enclosure.

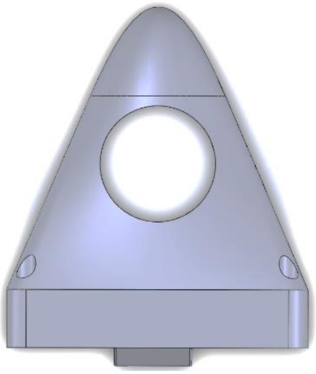


Figure 10: Top view of Intended Enclosure

The field technician will have to install Solarity first before installing the solar panel. The technician will first bring Solarity on top of the pole and align the hole of Solarity to the pole. Then the technician will slide Solarity down along the pole to a height for transit users to view the screen. Custom made security bolts, nuts, and screws will be used to tighten Solarity to the pole and prevent theft. Figure 11 shows the procedure of installing Solarity on a pole.

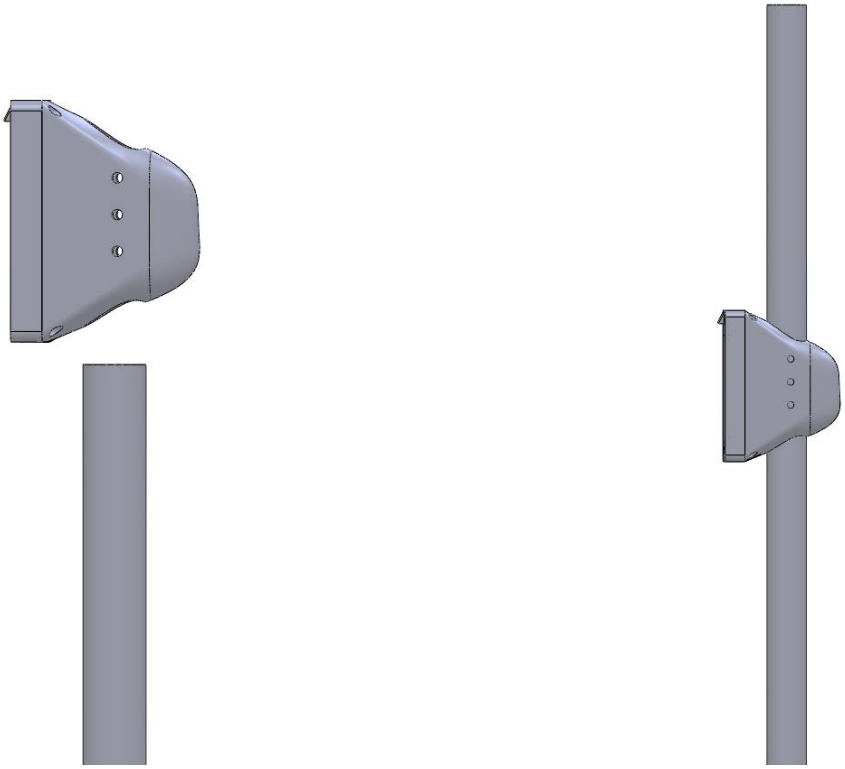


Figure 11: Installing the Device

Then the solar panel will be mounted on top of the pole, making it harder for non-authorized personal to steal the device.

### 3.3.2 Solar Panel Placement Design

The solar panel will be mounted on top of the pole with a universal top of the pole mount. The solar panel will be mounted on top of the pole with custom security nuts, bolts and screws to lower the chances of possible vandalism and theft.

The output of a solar panel is maximized when the rays of the sun are normally incident upon its surface. Because the sun is lower in the sky during the winter, the solar industry recommends orienting the solar panel due south, at a tilt angle (with respect to the ground) of latitude +15° to maximize output during the winter months [19]. The load must operate year round, so the solar panels will be mounted at latitude + 15° tilt angle as shown in Figure 12.

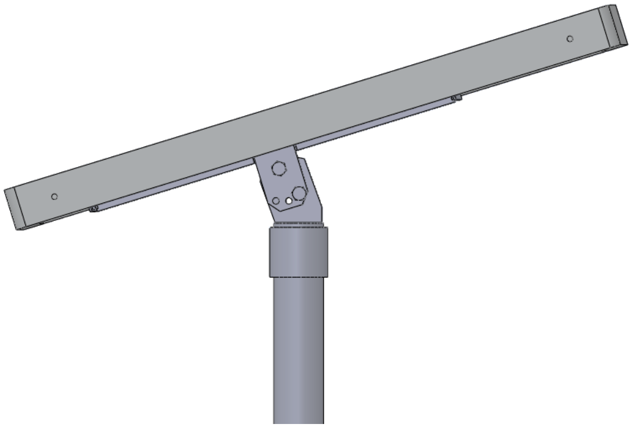


Figure 12: Side View of Solar Panel Orientation for Maximizing Winter Output (Northern Hemisphere)

# 4 STANDARDS

Solarity will follow CSA-C22.2 Canadian Electrical Code Part I and Part II standards. These standards establish safety standards for the installation and maintenance of electrical equipment in buildings, structures, and premises [20] [21]. Solarity will meet these standards by some of the following design:

- The enclosure is made up of Kydex® thermoplastic sheet , which is non-combustible
- The enclosure has no rough edges and burrs
- Solarity will have markings to indicate registered trademark, rated battery voltage, rated solar panel, and warning or caution markings near the battery and solar panel.
- Solarity has a unique enclosure that does not resemble a toy

Solarity will follow CSA-C22.2 No. 0.17-00 – Evaluation of Properties of Polymeric Materials and CSA-C22.2 No. 94-M91 – Special Purpose Enclosures standards. These standards provides procedures for evaluating materials and parts made from polymeric materials intended for application in electrical equipment, as well as design and safety requirements [22] [23]. These standards provides test cases to test flammability, mechanical properties, thermal properties, resistance to weathering and electrical properties of the material. As our enclosure will be made up of entirely polymeric material, we must follow these standards to ensure that the material we chosen meets those requirements. We will meet these standards by subjecting the enclosure to the tests outlined by these standards and adjust our design appropriately to pass those tests.

Solarity will meet both CSA-C22.2 No. 107.2.01- Battery Chargers and CSA-C22.2 No. 0.23-15 – General Requirements for Battery-Powered Appliances standards. These standards applied to portable, mobile, and station battery chargers and battery-powered devices for indoor and outdoor use [24] [25]. We will meet those standards by the following design:

- Enclosure will enclose all electrical parts
- Enclosure will be made of Kydex® thermoplastic sheet, which has high strength and rigidity to withstand abuse to which Solarity may be subjected
- The space within the enclosure of Solarity is sufficient to provide ample room for wiring
- All of the wires will be insulated.
- Connectors from the solar panel and battery to the device will have a unique shape, such that there is only one way to connect them. This will prevent the technicians from installing the solar panel or battery in reverse polarity.
- Solarity will be grounded to the pole to prevent any electrical shock.

- The charge controller has over current and voltage protection to prevent any damage to the battery and microcontroller.

Solarity will meet the RoHS compliance, which restricts the use of six hazardous materials found in electrical and electronic products [26]. Solarity will meet this compliance by not using the six hazardous material, as outlined on the RoHS website.

# 5 SUSTAINABILITY & SAFETY

## 5.1 SUSTAINBILITY AND RELIABILITY

The reliability of Solarity is a critical aspect of the product. The device must be able to operate through months and years of servicing transit users.

The prototype device enclosure will be 3D printed with a strong, lightweight, and durable thermoplastic material. Individual parts of our product will be recyclable, lead-free, and toxic chemical free. Our system will be powered with environmentally friendly solar power.

The battery packs used by Solarity will be charged with solar power. The solar panels will provide enough energy throughout the year, even during days with limited sunlight.

The product will operate and remain reliable under a range of temperatures throughout the year and varying weather conditions.

## 5.2 SAFETY

Since Solarity will be installed in public areas, it will be subjected to many different environmental conditions as well as possible abuse from transit users. Solarity will be designed such that the risk of fire or electric shock as a result of abnormal operation, hazardous environment, and human factors are obviated as far as is practical.

In order to prevent the battery and solar panel from being installed in reverse polarity, they must connect to the charge controller with non-reversible polarity connectors. The battery will be plugged into the charge controller through a JST connector, these connectors only allow for insertion in a single orientation. The solar panel will be connected to the charge controller with a DC jack, which cannot be installed in reverse polarity.

The enclosure will contain all live parts of our system except for the solar panel. The solar panel will be mounted at the top of the transit pole, and the cable connecting the panel to the charge controller will be out of reach, installed in the hollow center of the transit pole. This will however require that a hole be drilled to allow the solar panel cable to connect to the enclosure, mounted around the exterior of the pole. This cable will be insulated to prevent an electrical short.

Solarity's enclosure features rounded edges without sharp corners to prevent injury to personnel installing the device, or to the users of the device. The enclosure is also secured with screws that can only be unscrewed by using a special tool. One example of such a screw, the Key-Rex is shown in Figure 13. These screws, produced by Bryce Fastener are only removable with the unique driver bit that is provided with the screws [27].

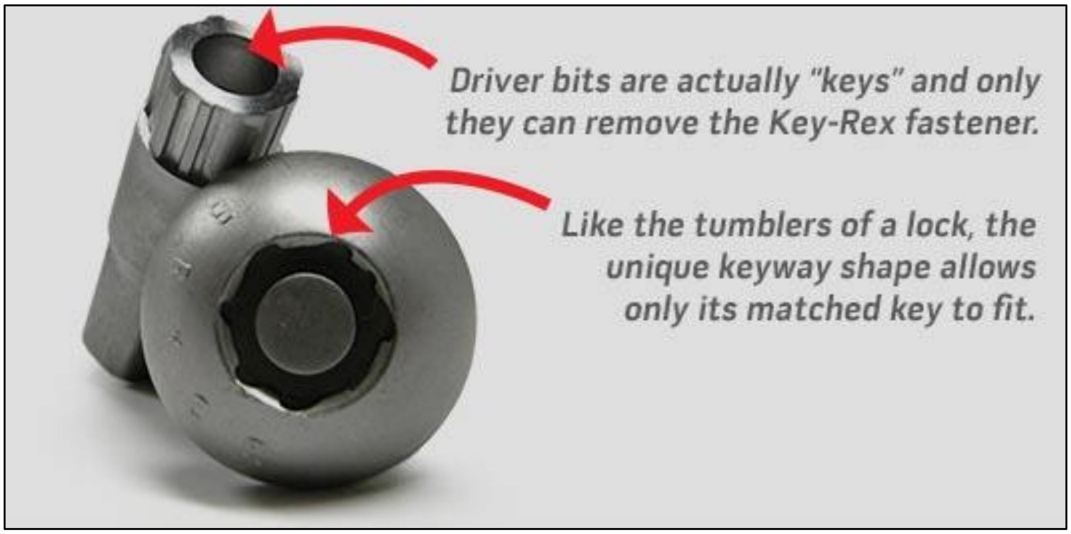


Figure 13: Key-Rex Fastener [27]

# 6 SYSTEM TEST PLAN

## 6.1 UNIT TESTING

### 6.1.1 Microcontroller

The MSP432 MCU needs to be able to receive transit data from the server (via the GSM), as well as send commands and data to our display. In order to confirm that sending and receiving works correctly we will perform separate tests.

MCU Test 1: Receive and store data
Run a program that, after receiving data from the server through the GSM module, will take the data from the UART and store into RAM. The program will print sample data from memory to screen to check functionality.
**Outcome:** Sample data from the server will be printed onto the console successfully from the microcontroller from RAM.

MCU Test 2: Send command and data to display
Run a program that will send an upload image command from our MCU to the display with data stored on RAM previously.
**Outcome:** The display will update with a new image correctly shown.

The Microcontroller will be integrated with the GSM Module, Display, Sensors, and battery unit to ensure a full functioning system.

### 6.1.2 GSM Module

The Sim 800C GSM Shield needs to be able to send requests (AT commands) to connect with the server and get data to the MCU. In order to confirm the functionality of the GSM we will perform a test.

GSM Test: Send AT Commands
Send commands through a serial console to connect to the server and read data back to the console.
**Outcome:** The serial console will show correct data from the server on the console.

### 6.1.3 Display

The MPico Pervasive Display needs to be able to display a stable image, and update when refreshed.

Display Test 1: Display image
Upload image to the display through the SPI port
**Outcome:** Display controller will return the correct response code (0x9000).

Display Test 2: Refresh display
Send a refresh command to the display, with a new image on the display.
**Outcome:** Display will correctly refresh the screen, showing the new image.

### 6.1.4 Proximity Sensor

Proximity Sensor Test 1: Sense nearby people within maximum radius
Detect a nearby person within 6 meters radius facing forward of the system.
**Outcome:** Sensor correctly detects nearby person

Proximity Sensor Test 2: Sense nearby people outside maximum radius
Detect a nearby person outside 6 meters radius facing forward of the system.
**Outcome:** Sensor will not detect the person.

### 6.1.5 Light Sensor

Light Sensor Test: Sense when daylight is not available
During the night time, detect when there is not enough daylight to view the display
**Outcome:** Sensor correctly detects the lack of light and switches on the LED.

### 6.1.6 Battery Unit and Solar Panel

Battery Unit and Solar Panel Test 1: Battery lifetime
Without recharging, run fully charged battery (connect to all components) until battery depletes
**Outcome:** Battery will die after 144 hours.

Battery Unit and Solar Panel Test 2: Charging capability
During discharge (or operation of system), charge battery from solar energy through the panels.
**Outcome:** Battery will charge until full (within 7 hours) or until the absence of solar energy.

## 6.2 SYSTEM INTEGRATION TESTING

In order to integrate the separate components of our device efficiently and effectively, we will integrate components in a specific manner. Since the testing will involve having to integrate certain components, we will integrate them as necessary to perform the tests. These integrations will be tested according tests explained below.

### 6.2.1 Integration Test 1

As explained in the functional specification the overall proof of concept of Solarity is to a stable transit information image displayed on the pervasive display. The first integration test starts with a request from the GSM module and after the whole process is complete the microcontroller will transmit the image data to the display, which will showcase the bus time arrival as an image. The required steps to satisfy the completion of this integration test are labeled in order as follows:

1. The Solarity unit will be properly plugged up and powered.
2. The Solarity unit (the unit will have a unique identification number) be placed at a designated transit station/stop in Metro Vancouver. The IT technician will input the unit's unique identification number to the server and match it with the designated transit station. The server will now be able to provide the Solarity unit with the applicable transit information.
3. The microcontroller will program the GSM with AT commands.

*The steps above only occur during the initial setup of Solarity. The following steps (4-13) get repeated in order every 2 minutes and continue as long as respective Solarity device is in operation:*

4. The GSM Module will send a request to the Solarity server via the 3G network in the form of GPRS data.
5. The server will send a request to the TransLink server to retrieve the bus information for the designated stop.
6. The server will convert the data received into an image that will be displayed.
7. The server will break the image into a byte-stream, where the bytes represent the ordered pixels that make up the image.
8. The bytes will be transformed into the EPD format, which is the format that the pervasive display controller can read.
9. The server will send the data to the GSM module.
10. The GSM module will transmit the bytes received to the microcontroller.
11. The microcontroller will transfer the bits to the pervasive display.
12. The pervasive display will produce the image on its screen.

*To test the sensors:*

13. The 2 sensors (proximity and light respectively) of the system have to be tested to ensure that the integrated system is working accurately:
    A. For the proximity sensor, we have to keep the system idle (in other words, have no one walk within 6m radius in front of it) for a given amount of time so that the display goes stops refreshing when not needed. Then let a person walk within the 6 meters radius facing forward of the system. The sensor will correctly detect the nearby person and switch the screen from idle mode and refresh the image.
    B. For the light sensor, we have to test the system during the night time to detect when there is not enough daylight to view the display. The sensor will correctly detect the lack of light and switches on the backlight LED on.

### 6.2.2 Integration Test 2

The first integration test covered the operation of Solarity without integrating the solar panel. The integration of the solar panel with the integrated system depicted in the previous integration test completes the Solarity unit. The system connected to solar panels will be tested according to steps 1 through 13 in the previous section. The solar panel in particular will be tested in parallel to steps 1 through 13. During discharge (or operation of system), the battery will charge from solar energy through the panels. The battery will charge until full (within 7 hours) or until the absence of solar energy. After this test, the fully integrated system will have been fully tested.

# 7 CONCLUSION

These functional specifications handle the requirements and capabilities of Sunlink's current design for Solarity.

The functional specifications were broken up into three major categories: Hardware, Software, and Mechanical, where each requirement has a specific priority. We are committed to completing critical requirements like easy configurability and basic power supply needs within our development phase, and if time permits, complete the lower priority requirements. Sustainability and safety is also a big concern to factor in for our product as reliability is key for success, and with the product in contact with the public 24/7, safety requirements must be followed closely.

The development of our proof-of-concept is well underway and we will be seeing a working prototype by December 2015.

# 8 REFERENCES

[1] GreenEnergy.ca, "Tilt and Angle Orientation of Solar Panels," [Online]. Available: http://greenerenergy.ca/PDFs/Tilt%20and%20Angle%20Orientation%20of%20Solar%20Panels.pdf. [Accessed 12 November 2015].

[2] Voltaic Systems, "Estimating Battery Charge Time from Solar," 14 May 2010. [Online]. Available: http://www.voltaicsystems.com/blog/estimating-battery-charge-time-from-solar/.

[3] Maxim Integrated Products, Inc., "Switch-Mode, Linear, and Pulse Charging Techniques for Li+ Battery in Mobile Phones and PDAs," 2 August 2002. [Online]. Available: https://www.maximintegrated.com/en/app-notes/index.mvp/id/913.

[4] Leonics, "Basics of MPPT Solar Charge Controller," 2013. [Online]. Available: http://www.leonics.com/support/article2_14j/articles2_14j_en.php.

[5] Linear Technology Corporation, "Power Tracking 2A Battery Charger for Solar Power," [Online]. Available: http://cdn.sparkfun.com/datasheets/Prototyping/LT3652.pdf. [Accessed 11 November 2015].

[6] Society of Robots, "Infrared vs. Ultrasonic - What You Should Know," 27 January 2008. [Online]. Available: https://www.societyofrobots.com/member_tutorials/node/71.

[7] MaxBotix Inc., "HRLV-MaxSonar-EZ Series," [Online]. Available: http://maxbotix.com/documents/HRLV-MaxSonar-EZ_Datasheet.pdf. [Accessed 10 November 2015].

[8] A. Lindsay, "How the Phototransistor Circuit Works," 3 March 2012. [Online]. Available: http://learn.parallax.com/node/258.

[9] RP Electronics , "Quad Band Cellular Duck Antenna," [Online]. Available: http://www.rpelectronics.com/cel-00675-quad-band-cellular-duck-antenna-sma.html. [Accessed 9 November 2015].

[10] TI Texas Instruments Incorporated, "MSP432P401R LaunchPad," [Online]. Available: http://www.ti.com/tool/msp-exp432p401r. [Accessed 8 November 2015].

[11] L. PJRC.COM, "Lowering Power & Extending Battery Life," [Online]. Available: https://www.pjrc.com/teensy/low_power.html. [Accessed 8 November 2015].

[12] Arduino, "Arduino Zero Low Power Overview," [Online]. Available: https://www.arduino.cc/en/Tutorial/ArduinoZeroPowerConsumption. [Accessed 8 November 2015].

[13] Pervasive Displays, "MpicoSys 7.4" Solutions," [Online]. Available: http://www.pervasivedisplays.com/kits/mpicosys740. [Accessed 10 November 2015].

[14] SIMCON, "SIM800 Series AT Command Manual," Adafruit, [Online]. Available: https://www.adafruit.com/datasheets/sim800_series_at_command_manual_

v1.01.pdf. [Accessed 9 Nov 2015].

[15] Pervasive Displays, "Developer's Guide," [Online]. Available: http://www.pervasivedisplays.com/LiteratureRetrieve.aspx?ID=212981.

[16] J. Raj, "An Introduction to the MEAN Stack," 17 April 2014. [Online]. Available: http://www.sitepoint.com/introduction-mean-stack/.

[17] "gm," [Online]. Available: https://www.npmjs.com/package/gm. [Accessed 5 November 2015].

[18] Curbell Plastics , "Curbell Plastics Engineering Materials Guide," Curbell Plastics, [Online]. Available: http://www.curbellplastics.com/technical-resources/pdf/engineering-materials-guide-curbell.pdf. [Accessed 9 November 2015].

[19] IEEE, "IEEE 1562 - Guide for Array and Battery Sizing in Stand-Alone PV Systems," New York, 2007.

[20] CSA Group, "CSA C22.1-15: Canadian Electrial Code, Part 1: Safety Standard for Electrial Installations," Toronto, 2015.

[21] CSA Group, "CSA C22.2 No.0-10: General requirments - Canadian Electrial Code, Part 2 (reaffirmed 2015)," Toronto , 2010.

[22] CSA, "CSA-C22.2 No. 0.17-00 – Evaluation of Properties of Polymeric Materials (reaffirmed 2013)," Toronto, 2000.

[23] CSA Group, "CSA C22.2 No.94-M91: Special Purpose Enclosures (reaffimed 2011)," Toronto, 1991.

[24] CSA Group , "CSA C22.2 No. 0.23-15 : General requirement for battery-powered appliances," Toronto , 2015.

[25] CSA Group , "CSA C22.2 No. 107.2-01 : Battery Chargers (reaffirmed 2011)," Toronto , 2008.

[26] RoHS, "RoHS Compliance," 13 October 2015. [Online]. Available: http://www.rohsguide.com/. [Accessed 15 October 2015].

[27] Bryce Fastener, "Key-Rex," [Online]. Available: http://www.tinyosshop.com/image/cache/data/sensors/Ultrasonic/MBseries-2-600x600.jpg. [Accessed 9 November 2015].

[28] K. S. Hāfez, "The Gift," Page 160, 1350. [Online]. Available: http://www.goodreads.com/author/show/859027._. [Accessed September 2015].

[29] GreenerEnergy.ca, "Tilt and Angle Orientation of Solar Panels," [Online]. Available: http://greenerenergy.ca/PDFs/Tilt%20and%20Angle%20Orientation%20of%20Solar%20Panels.pdf. [Accessed 12 November 2015].

[30] ElectronicProducts.com, "MSP432P401R is a Low-Power, High-Performance, 32-Bit ARM Cortex M4F," [Online]. Available: http://www.electronicproducts.com/Digital_ICs/Microprocessors_Microcontrollers_DSPs/MSP432P401R_is_a_Low_Power_High_Performance_32_Bit_ARM_Cortex_M4F.aspx. [Accessed 11 November 2015].

[31] Lee's Electronic, "SIM800C QUAD BAND GSM/GPRS WITH GPS SHIELD,"

[Online]. Available: http://leeselectronic.com/product/1163.html. [Accessed 12 November 2015].

[32] Sparkfun, "Ultrasonic Range Finder - LV-MaxSonar-EZ0," [Online]. Available: https://www.sparkfun.com/products/8502. [Accessed 12 November 2015].

[33] Sparkfun, "SparkFun Ambient Light Sensor Breakout - TEMT6000," [Online]. Available: https://www.sparkfun.com/products/8688. [Accessed 12 November 2015].

# APPENDIX A

Table 12: Enclosure Type Definitions *[23]*

| Type | Description |
|---|---|
| Type 2 | An enclosure for indoor use, constructed so as to provide a degree of protection against dripping and light splashing of noncorrosive liquids, and falling dirt. |
| Type 3 | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against rain, snow, and windblown dust; undamaged by the external formation of ice on the enclosure. |
| Type 3R | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against rain, and snow, undamaged by the external formation of ice on the enclosure. |
| Type 3S | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against rain, snow, and windblown dust; the external mechanism(s) remain operable while ice covered. |
| Type 4 | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against rain, snow, windblown dust, splashing and hose-directed water; undamaged by the external formation of ice on the enclosure. |
| Type 4X | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against rain, snow, windblown dust, splashing and hose-directed water; undamaged by the external formation of ice on the enclosure; resists corrosion. |
| Type 5 | An enclosure for indoor use, constructed so as to provide a degree of protection against dripping and light splashing of noncorrosive liquids, and settling dust, lint, fibres, and flyings. |
| Type 6 | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against the entry of water during temporary submersion at a limited depth; undamaged by the external formation of ice on the enclosure. |
| Type 6P | An enclosure for either indoor or outdoor use, constructed so as to provide a degree of protection against the entry of water during prolonged submersion at a limited depth; undamaged by the external formation of ice on the enclosure; resists extended corrosion. |
| Type 12 | An enclosure for indoor use, constructed so as to provide a degree of protection against circulating dust, lint, fibres, and flyings; dripping and light splashing of noncorrosive liquids; not provided with knockouts. |
| Type 12K | An enclosure for indoor use, constructed so as to provide a degree of protection against circulating dust, lint, fibres, and flyings; dripping and light splashing of noncorrosive liquids; and provided with knockouts. |
| Type 13 | An enclosure for indoor use, constructed so as to provide a degree of protection against circulating dust, lint, fibres, and flyings; seepage and spraying of noncorrosive liquids including oils and coolants. |

Table 13: Comparison of Specific Application of Enclosures for Nonhazardous Locations *[23]*

| **Provides a degrees of protection against the following environmental conditions** | **Type of Enclosure** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **2** | **3** | **3R** | **3S** | **4** | **4X** | **5** | **6** | **6P** | **12** | **12K** | **13** |
| Dripping and light splashing of noncorrosive liquids, falling dirt | x | x | x | x | x | x | x | x | x | x | x | x |
| Circulating dust, lint, fibres, and flyings | - | x | - | x | x | x | - | x | x | x | x | x |
| Setting dust,lint, fibres, and flyings | | x | | x | x | x | x | x | x | x | x | x |
| Hose down and splashing water | - | - | - | - | x | x | - | x | x | - | - | - |
| Corrosion | - | - | - | - | - | x | - | - | x | - | - | - |
| Occasional temporary submersion | - | - | - | - | - | - | - | x | x | - | - | - |
| Occasional prolonged submersion | - | - | - | - | - | - | - | - | x | - | - | - |
| Oil and coolant seepage, spraying and splashing | - | - | - | - | - | - | - | - | - | - | - | x |
| Rain, snow, and external formation of ice* | - | x | x | x | x | x | - | x | x | - | - | - |
| External formation of ice+ | - | - | - | x | - | - | - | - | - | - | - | - |
| Wind-blown dust | - | x | - | x | x | x | - | x | x | - | - | - |

*External operating mechanism(s) shall not be required to operate when the enclosure is ice covered.

+External operating mechanism(s) shall be operable when the enclosure is ice covered.

“X” indicates a degree of protection is provided.

# APPENDIX B

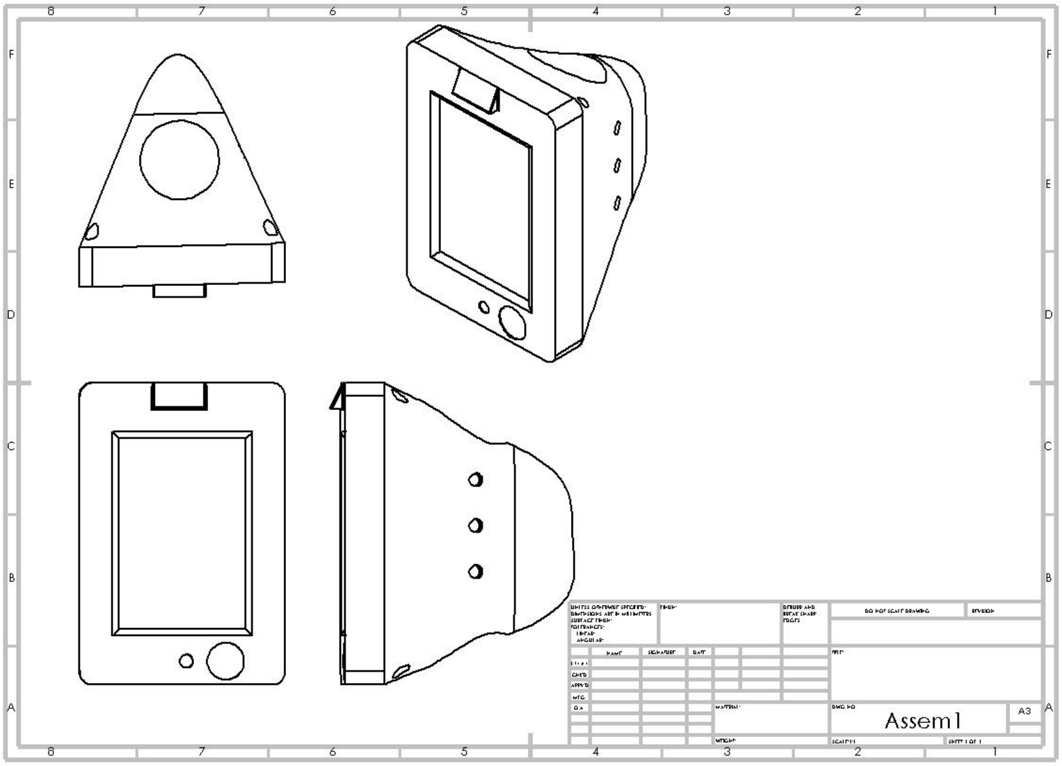


Figure 14: Isometric View of Solarity

# APPENDIX C

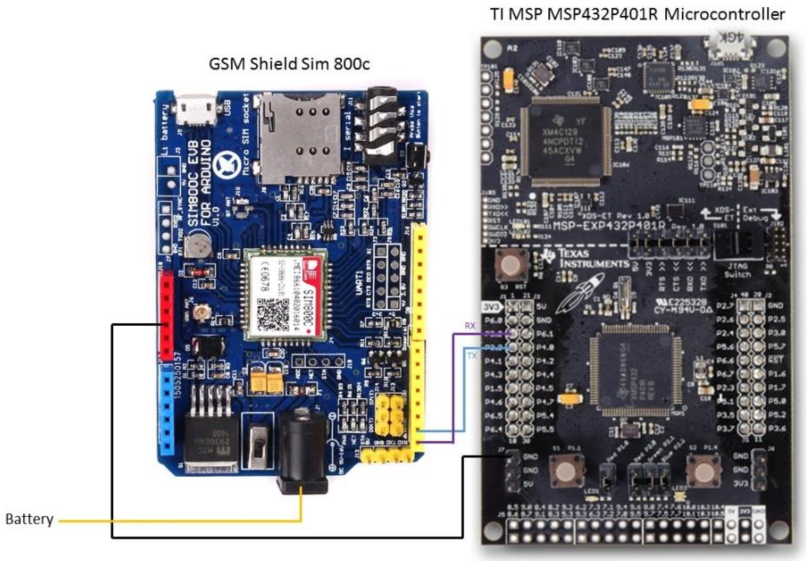


Figure 15: Wiring between Microcontroller [27] and GSM Module [28]

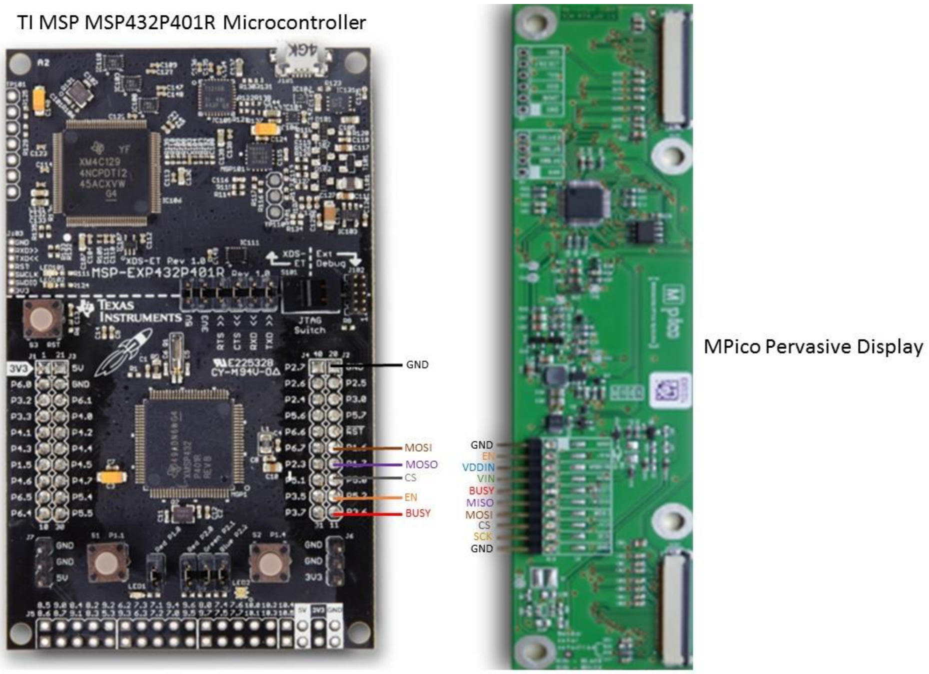


Figure 16: Wiring between Microcontroller [27] and Display Circuit [29]

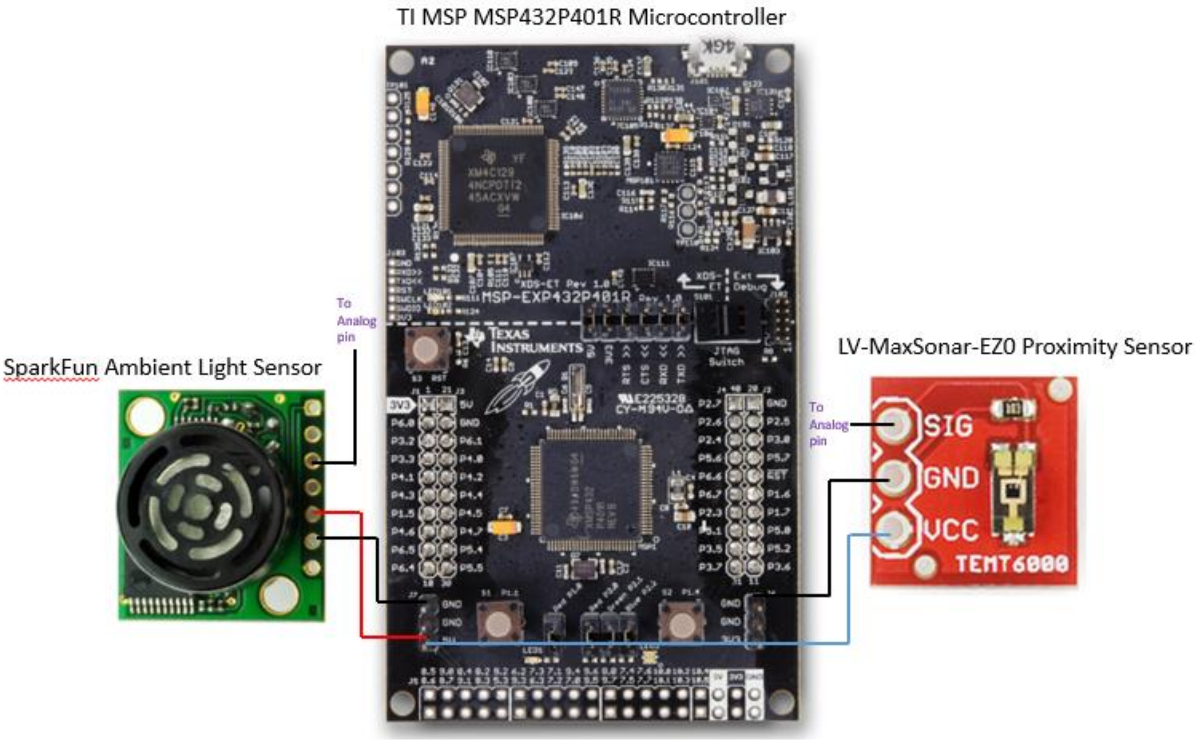


Figure 17: Wiring between Microcontroller [27] Proximity Sensor [30] and Light Sensor [31]

**SUNLINK**
Simon Fraser University,
8888 University Dr.
Burnaby, BC Canada
Email: ensc440-sunlink@sfu.ca