INDUSTRIAL REMOTE SENSING MODULE FOR MACHINERY AND SYSTEMS TROUBLEDSHOOTING

By
GREG DEPAUL

In Collaboration With
JOSE ESCOBEDO, CHENMU LI, XINRAN FANG and EZZULDDIN NAJI

A Thesis Submitted to The Honors College

In Partial Fulfillment of the Bachelors degree

With Honors in

Electrical Engineering

THE UNIVERSITY OF ARIZONA

M A Y 2 0 1 6

Approved by: ______________________________

Dr. Gerald Pine

College of Engineering

ABSTRACT

My team was given the task of design a data acquisition module that is both versatile in its ability to collect different sensory data by alternating sensor units, while also being portable, particularly capable of fitting within the average pocket. The final module is also cost effective, estimated to cost a mere twenty US dollars in order produce a single module.

MEMBERS INVOLVED

GREG DEPAUL – COMPUTER ENGINEERING – Programming Lead, working specifically on the data communications of the module and integrating the module into a graphical user interface.

JOSE ESCOBEDO – SYSTEMS ENGINEERING – Team Lead, responsible for the team intercommunications and keeping the project within the functional specifications.

CHENMU LI – MECHANICAL ENGINEERING – Mechanical lead, responsible for designing a durable case for the module that allows the user to interface with the module.

XINRAN FANG – COMPUTER ENGINEERING – GUI Lead, responsible for designing a graphical user interface that allows the user to give commands to the module in an easy and effective manner.

EZZULDDIN NAJI – ELECTRICAL ENGINEERING – Electrical Lead, responsible for designing printed circuit board that meets the demands for long lasting battery life and data acquisition.

Final Report

Universal Sensors: Remote Sensor Module

Spring 2016

Sponsor: Mark Irwin, Texas Instruments
Mentor: Clayton Grantham
Team 15055: Jose Escobedo, Chenmu Li, Xinran Fang, Ezzulddin Naji, Greg Depaul

Table of Contents

1.0 Project Scope

1.1 Project Overview

This project is the University of Arizona Engineering Capstone for Team 1555 sponsored by Texas Instruments. The team has collected and developed requirements, constraints, and risks to design and implement their Senior Capstone Project. The purpose of this project is to design a portable Remote Sensor Module (RSM) to aid technicians and professionals in troubleshooting by sensing real world signals and recording the information over a selectable time period. This recorded information will be downloaded onto a portable device, such as a laptop or notebook which can be viewed in an easy-to-understand GUI interface. The Remote Sensor Module Capstone will be completed for Senior Design Day, which is May 3rd, 2016.

Texas Instruments would also like to have the sensing modules designed with TI precision components to showcase their capabilities and affordability. This includes the MSP430 along with precision operational and instrumentation amplifiers. The initial RSM kit will include 5 remote sensors that can record electrical current, temperature, light interruption, motion, and the presence of fluid.

It is also a project goal to make this product marketable by keeping the cost of product creation to a minimum. The marketable price point for each RSM is $50.00 per unit. This includes only the module and not any respective sensors. The minimum acceptable profit margin for each RSM for this project to be successful will be 15%. It is important to note the $3,500 budget constraint on the project as well.

1.2 Concept of Operations

In order to achieve our goal for the RSM, we need to consider certain aspects about the system, such as stakeholders, system boundaries, system environment, and system constraint, and system use. The consideration of this and other aspects is called the Concept of Operations. This will allow the team to describe the Remote Sensor Module as an entire system.

After an analysis of the concept of operations, certain aspects of the project were noticed by the team. For example, the product is relatively new and has no prototypes to be based on. Texas Instruments has all the sensors, but not in any single device that compiles the data from all of their sensors. The RSM will be able to synchronize all of this data together in order for ease of operation by the user. In a sense, technicians will now only need one product in order to troubleshoot machinery instead of various devices.

In addition, the RSM’s operating environment requires the RSM to be attachable to any surface and be able to withstand certain industrial conditions like extreme temperatures, vibrations and liquid exposure. It was also determined that the RSM was able to be designed with the following design boundaries: TI TINA, PCB Express, Xcode, and Solidworks. Each of these boundaries pertain to a specific part of the RSM, be it the casing of the module or the coding required for data transmission. With these observations, the RSM’s expected output is for the RSM to collect data from its surroundings, transfer the sensor readings to memory, and through this data, display on a GUI the troubleshooting data.

The team must also consider the constraints the stakeholders have placed on the project. The entire project budget is limited to $3,500 and Texas Instruments wants the cost of making this product under $50, for the sake of marketability. Aside from financial constraints, the RSM must also have a battery life that allows the module to run for 24 hours and contain a battery that is rechargeable and replaceable.

For a more detailed list of the Concept of Operations, see the Appendix.

1.3 System Diagram

The following figure (Figure 1.3.1) is a diagram of the operation of the system:

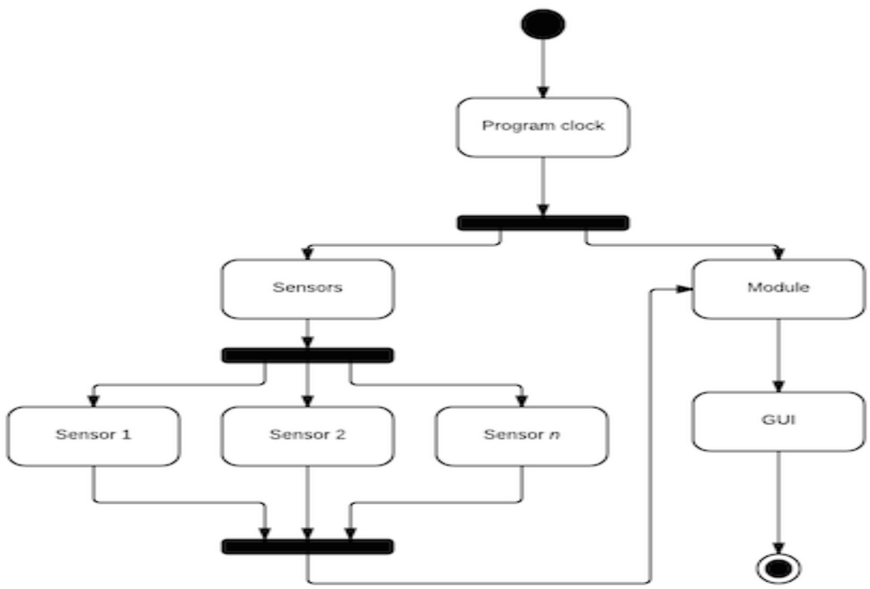


Fig. 1.3.1 RSM System Diagram

After the user programs a recording period, the user then attaches sensors to the object undergoing troubleshooting (up to 5). The sensors will transfer this data to the module (RSM) and the data in the module will be transferred to the device and displayed on the GUI.

1.4 Operational Scenarios

The system currently has 2 operational scenarios: Nominal and Off-Nominal. They are as follows:

Nominal scenario

1. Begins when user programs clock

2. User plugs in sensors
3. User attaches sensors
4. System completes recording
5. User detaches sensors
6. User transfers data
7. User starts up GUI
8. Ends as user saves data

Off-nominal scenario

1. Begins when user programs clock
2. User plugs in sensors
3. User attaches sensors
4. System completes recording
5. User detaches sensors
6. Transfer of data fails
7. User checks hardware
8. User checks software
9. Makes repairs
10. User reruns system
11. Ends when user is able to save data

2.0 System Requirements

Listed below are the requirements for the product and the team's aspirations for the design. It is divided into functional, interface, technology, performance, and mechanical requirements. Each requirement includes information about the type of requirement, its description, and status.

2.1 Functional Requirements

Number	Type	Description	Status
2.1	Functional	The RSM shall have certain dimensions and be portable	Must
2.2	Functional	The RSM must operate for a minimum of 24 hours in full record mode	Must
2.3	Functional	The RSM shall be able to transfer recorded data via a USB port to the portable device	Must
2.4	Functional	The RSM shall display information upon a graphical user interface	Must
2.5	Functional	The RSM shall provide steady and reliable data in certain industrial conditions.	Must

2.6	Functional	The RSM shall consist of a module interfacing with auxiliary sensors and a user interface	Must
2.7	Functional	The RSM shall be operated by a maximum of two batteries	Must
2.8	Functional	The RSM shall be attached to any kind of equipment surface via a common mechanical interface (CMI)	Must
2.9	Functional	The physical sensor will be encased in the housing or connected by cables no longer than 12”	Must
2.10	Functional	The RSM must have a real time clock and able to record the start and stop time of data collection	Must

Table 2.1.1 Functional Requirement

2.2 Interface Requirements

Number	Type	Description	Status
2.4.1	Interface	The GUI will display tested data vs time charts	Must
2.4.2	Interface	The RSM recorded data should have the ability to time synchronize with other RSM data (up to 5)	Must
2.6.1	Interface	The microcontroller in this module will be a MSP430 model	Must
2.6.2	Interface	The module will have a battery management chip	Must
2.6.3	Interface	The module will have an integrated Bluetooth module	Desired
2.6.4	Interface	Each sensor will have 1x1x1 dimensions for sensors	Must
2.8.2	Interface	The surfaces attachable include aluminum, glass, steel, and wood	Must
2.8.2	Interface	Additional CMI compatible surfaces include fiberglass, rubber, drywall	Desired

Table 2.2.1 Interface Requirement

2.3 Technology Requirements

Number	Type	Description	Status
2.3.1	Technology	The RSM shall have Bluetooth enabled data delivery	Desired
2.3.2	Technology	The RSM shall interact with an application to interpret data	Must
2.10.1	Technology	Auxiliary battery for the clock	Must

Table 2.3.1 Technology Requirement

2.4 Performance Requirements

Number	Type	Description	Status
2.2.1	Performance	The RSM shall have a programmable time and	Must

		recording resolution setting	
2.5.1	Performance	The temperature data will be specified in degrees Celsius	Must
2.5.2	Performance	Data such as voltage, current, motion, and fluid will be relative (i.e. yes or no)	Must
2.5.3	Performance	The RSM shall operate in the temperature range of -20C to 85C	Must
2.5.4	Performance	The RSM shall operate in the industrial temperature range of -40C to 125C	Desired
2.5.5	Performance	The RSM shall be submersible in its entirety in water	Desired
2.7.1	Performance	The RSM shall have a 3V to 5V battery	Must
2.7.2	Performance	The RSM battery shall be rechargeable	Must
2.7.3	Performance	The RSM battery will be replaceable	Must

Table 2.4.1 Performance Requirement

2.5 Mechanical Requirements

Number	Type	Description	Status
2.1.1	Mechanical	The dimensions will be 3x2x1 inches	Must
2.1.2	Mechanical	The dimensions will be .5x2x1 inches if plausible	Desired
2.1.3	Mechanical	The device should be less than 4 ounces in weight	Desired
2.9.1	Mechanical	The cables may be interchangeable between RSMs	Desired

Table 2.5.1 Mechanical Requirement

2.6 System Requirements Verification Matrix

The test matrix below shows how each requirement will be tested. The testing verification is done either by a form of a test, analysis, or inspection. Test-based verification is used when a measurement needs to be taken in order to substantiate the requirement. An analysis-based testing is used when the desired measure and a physical measure are linked through analysis. Finally, an inspection would entail a test that verifies, yes or no, if the requirement were accounted for. Refer to the Requirements section if needed for details on the requirements.

Req. #	Requirement Type	T	A	I
	Interface			
2.4	GUI Display	**X**		**X**
2.6	Module-Sensor interface			**X**

2.8	CMI	**X**		
	Technology			
2.2	Wireless info transmission		**X**	
	Performance			
2.2	24 hour operation	**X**		
2.5	Reliable data transmission	**X**	**X**	
2.7	Maximum 2 batteries		**X**	**X**
	Mechanical			
2.1	Dimensions			**X**

Table 2.6.1 System Requirement Verification Matrix

3.0 Preliminary Design Review

Our Preliminary Design Review (PDR) consisted of solidifying much of what our sponsored desired into several concrete methodologies that could satisfy our sponsor's needs. Specifically, to design a remote sensing module within the dimension constraints. There were many tradeoffs, the most explicit one being the sensors capabilities against the resource constraint due to the limited size of the device.

3.1 System Architecture

The RSM has two major design components, the hardware side as well as the software side. We have taken the time to detail each of these components, specifically any implications that come from these designs.

3.1.1 Hardware View

Starting with the hardware view, you'll notice an expanse of functional units that we believe are necessary to satisfy the sponsor's needs.

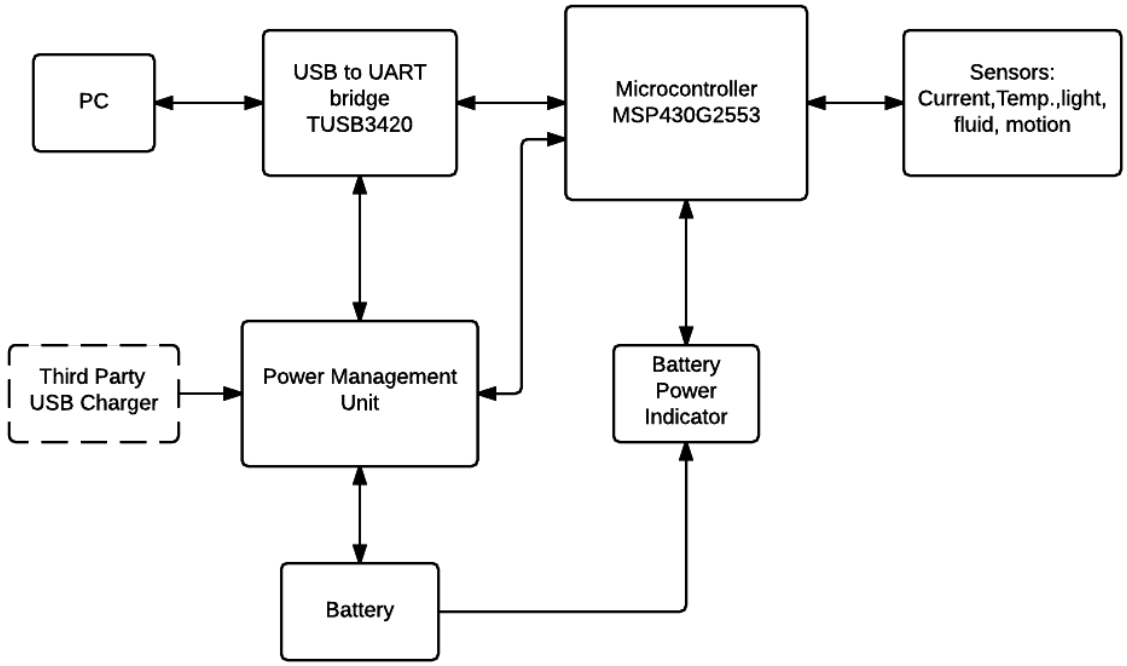


Figure 3.1.1 Hardware View

At the top of the hardware view, you see a rather typical paradigm for computing. The sensors serve as *input*, the microcontroller *processes*, the USB to UART bridge as well as the PC serve as both mechanisms to *store* and *output*. Thus, our system does a complete cycle of input, process, store and output. Further, the bottom of the hardware view diagram details how we plan to tackle the issue of powering the device.

3.1.2 Software View

The software view is a linear process. As the hardware view entails, we perform a cycle of input, processing, storing, and outputting results, those result being specifically sensory data.

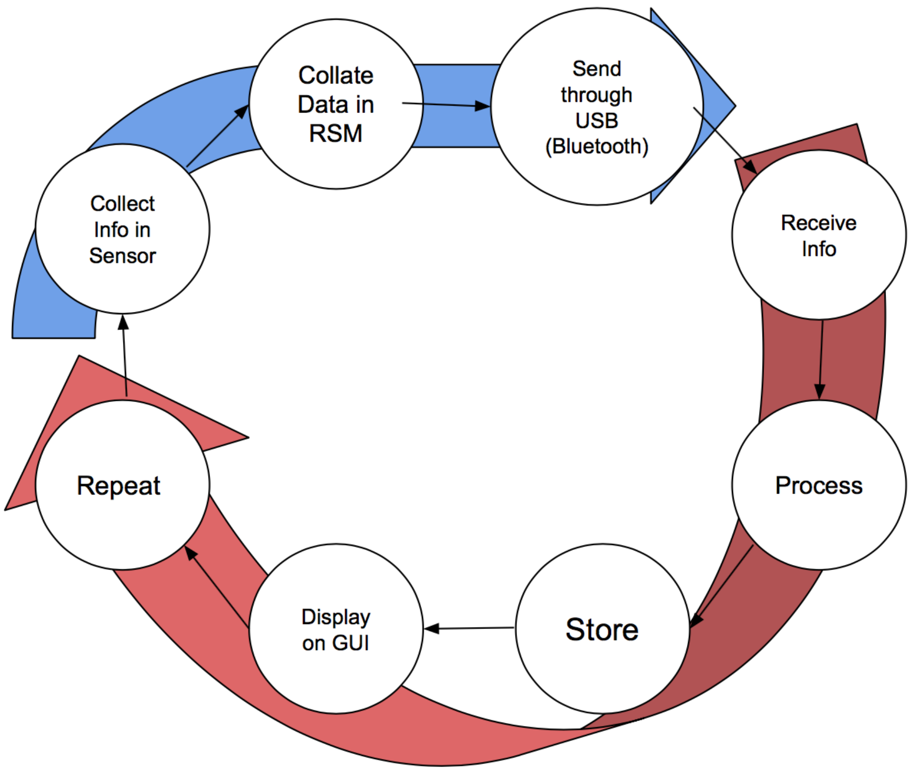


Figure 3.1.2 Software View

Observe that the blue section details the tasks to be performed by our microcontroller, while the red reflects the client's responsibilities. For the blue section, the RSM is simply to collect the information, repackage and then send that sensory information to the user client. The user client, which will consist of a single GUI application will then process that sensory information and visualize that information as a graph.

3.1.3 Functional Grouping

The RSM also has to be considered within its functionality. I order for the RSM to be completely functional, we must consider all interactions it has with other systems and/or other components The functional grouping for the RSM is in *Figure 3.1.3.*

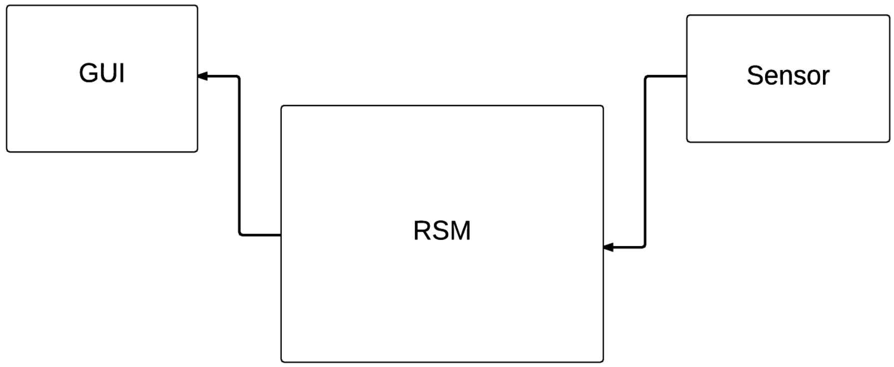


Figure 3.1.3 Functional Grouping

This functional grouping shows that the RSM is receiving data from the sensor that is currently connected to it. Once the data enters the RSM, it is processed and transferred into the GUI. Here the data will be displayed to the user.

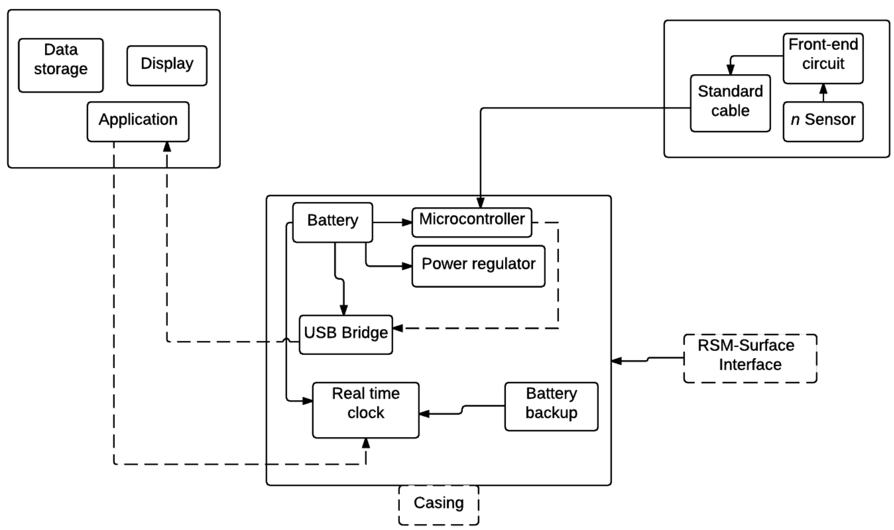


Figure 3.1.4 Detailed Functional Grouping

Figure 3.1.4 depicts a more detailed functional grouping specifies the exact process of each component: sensor, RSM, and GUI. The front-end sensor will have its necessary circuitry to interpret the data gathered by the sensor (see Section 5 for information on sensor subsystems). This data will be transmitted through the cable connected to the RSM. The Module's microcontroller will be able to process the data and synchronize with the other components.

3.2 Design Concepts

The Remote Sensor Module can now be placed into different design concepts now that its major structure and components have been defined. The project team decided to implement two types of design for the RSM: wired or wireless. The reason for this decision was that this was the only area that could necessarily be changed as far as design. Most components, if changed, offered little to no change to the design of the RSM. The changes that would occur from the two design concepts would be the location of the sensors and the implementation of a Bluetooth module. It is important to note that Texas Instruments only requires for the RSM to be wired and if possible, to have it be wireless.

If the RSM is designed as a wired device, the sensors will be located outside of the module itself and connected through a standard cable. This cable feeds data into the module and another cable will be able to feed data to the GUI once connected.

It the RSM is designed as a wireless device, the sensors will be encased in the module and be able to receive the data transmitted to the Bluetooth module from the sensor. The module will then be able to connect to the GUI for data display.

3.3 Trade-off Studies

The trade-off studies will allow the project team to determine which components are the best suited for the RSM. The critical components have been determined by the project team and several options for these components have been determined as well.

In order for these options to be considered, the trade-off study will be conducted by using a weighted scoring method as follows:

Low score: - **Medium score: ~** **High score: +**

These scores will be placed onto the component's attributes. Should the component's attribute be a poor fit, it will be scored with a (-). If its does fit but has some small error or discrepancy it can be scored with a (~). Logically, if it is deemed to be a great fit, it will be scored with a (+). See the following trade off figures to view the RSM trade-off results.

3.3.1 Surface Interface Trade-Offs

The surface interface trade-offs shows as Table 3.3.1 was based on three different types of design for the RSM being attach to the common mechanical interface. The tape design is the one that would attach the module to the common industrial interface with sticky tapes or magic tapes. The limitation about this design is customer might have to replace it recently due to the stickiness of the tape might wore off throughout time. Second design is what the team called "Slidable Design." Just like the common TI calculators, there will be a slidable tray for the RSM module to be attach to the common industrial interface. The downside for this design is it might be a little time-consuming. Last possible design would be suction cups for the RSM to be attach to the common industrial interface. The limitation for this design is that the suction cup cannot be attach to most of the common industrial interface. After did the weight system, the team decided to use "Slidable Design." It will cost more time to make it but the team is put it into a long-term consideration.

Surface Interface	Tape	"Slidable" design	Suction cups

Reliability	Low (-)	High (+)	Low (-)
Installation	Normal (~)	Normal (~)	Easy (+)
Price	Normal (~)	High (-)	Normal(~)
Score	7	9	9

Table 3.3.1 Surface Interface Trade-Offs

3.3.2 Casing Trade-Offs

The Casing trade-offs shows as Table 3.3.2 only shows the material that the team will choose to make the casing out of. The team picked Aluminum, Stainless Steel, Fiberglass and Carbon Fiber. Team have taken withstand temperature of the material, price, manufacturability, interference with signals, weight and safety to produce into consideration to weight the materials. Aluminum would be ideal if it would not interference with the signals. Finally, team decided to use Carbon Fiber as the casing materials. It is the ideal material for this module because of it has a high withstand temperature, would not interference with signals and lightweight. It it more expensive and harder to manufacture, but it can be overcome with professional company when the product is being put into the market.

Casing	**Aluminum**	**Stainless Steel**	**Fiber-glass**	**Carbon Fiber**
Withstand Temperature	660^{o}C (+)	1510^{o}C (+)	846^{o}C (+)	3500^{o}C (+)
Price	$0.38/lb (~)	$0.25/lb (+)	$1.5/lb (~)	$5/lb (~)
Manufacturability	Easy (~)	Easy (+)	Hard (~)	Hard (~)
Interference	Yes (~)	Yes (-)	No (+)	No(+)
Weight	168.48lb/ft^3 (~)	494.21lb/ft^3 (-)	117lb/ft^3 (+)	117lb/ft^3 (+)

Safety	Safe (+)	Safe (+)	Not-safe (-)	Not-safe (~)
Score	21	22	20	23

Table 3.3.2 Casing Trade-Offs

3.3.3 Battery Trade-Offs

The trade-offs study (*Table 3.3.3*) for the batteries was based on three major points: power capacity, temperature range of operation, and availability. NiCd battery scored the lowest so it is not going to be considered as an option mainly because of it's size and energy capacity. The Tadiran battery has the best results on capacity, temperature range and size, but unfortunately it did not meet the sponsor's requirement on availability which states that the battery shall be easily obtained and replaced by the user. The options are now narrowed down to two batteries, Lithium-ion and Polymer Lithium-ion. Both batteries have the same total scores, however, the Lithium-ion battery has a better temperature range and therefore it the winner.

Battery	**Lithium-ion**	**Poly. Lithium-ion**	**Tadiran (Lithium)**	**NiCd**
Energy Density	High (~)	High (~)	High (+)	Low (-)
Voltage/Cell	3.7v (+)	3.7v (+)	3.6v (~)	1.2v (-)
Physical Shape	Cylindrical/ rectangular (~)	Cylindrical/ rectangular (+)	Cylindrical (-)	Cylindrical (-)
Availability	Available (+)	Available (+)	Unavailable/ Limited (-)	Available(+)
Charging Cycle	1000 (~)	1000 (~)	5000 (+)	<1000 (-)
Operation Temp. Range	-20C to 60C (~)	0C to 60C (-)	-55C to 85C (+)	-20C to 45C (-)
Score	22	22	20	10

Table 3.3.3 Battery Trade-Offs

3.3.4 Connection Trade-Offs

For the connection trade-offs study, we already had Bluetooth as the ideal method of connection. This was originally chosen because of its greater portability, a convenience in an industrial setting. However, for we looked at how Bluetooth would fare against the standard USB 2.0 connection in the categories of Data Transfer, Implementation, Portability, and Adaptation. It is no surprise the USB 2.0 exceeds Bluetooth in the majority of these categories, particularly in implementation. Because of this we have decided to create an initial device with USB 2.0, and then, if time permits, include Bluetooth capability.

Connection	**USB 2.0**	**Bluetooth**
Data Transfer	480 mb/second (+)	3 to 24 mb/second (-)
Implementation	Relatively Easy (+)	Needs particular calibration. (-)
Portability	Requires connection (-)	Wireless (+)
Adaptor	None (~)	ad-hoc IEEE 802.11 (~)
Score	14	10

Table 3.3.4 Connection Trade-Offs

3.3.5 Sensor Trade-Offs

The sensor trade offs were done to help us decide whether or not to go about creating our own sensors. The categories listed below would indicate that indeed we would not construct our own sensors. However, we later discovered after this study was done that we could in fact make sensors that would versatile in what sensory data was observed. Thus, we have chosen to do a combination of pre-made as well as self-made sensors for our module.

Sensor	**Pre Made**	**Self Made**
Difficulty to Implement	Easy (Already Implemented) (+)	Difficult and potentially time consuming (-)
Accuracy	Reliable (+)	Variable (~)
Likelihood of Failure	Unlikely (+)	Extremely likely (+)
Price	Cheap to Expensive (~)	Cheap (+)

Interface	API Provided (+)	Learning Curve (-)
Score	23	15

Table 3.3.1 Sensor Trade-Offs

3.3.6 Microcontroller Trade-Offs

Based on our trade-off study Table 3.3.6, MSP430 is the best choice for this project. MSP430 has a wider range for operating temperature, which is one of the requirements for this project. and it also has large flash memory for recording the data from the sensors. It has a bigger launchpad that could hardly fit in the casing, but if we only use the chip, more space can be saved for other parts. In addition to that, MSP430 also has an advantage on price. Both MSP430 and PIC24 have the same score. We choose MSP430 since our sponsor is Texas Instruments, he encouraged us to choose as many products from Texas Instruments as possible. So MSP430 is the winner.

Microcontroller	**MSP430**	**Arduino Uno**	**PIC24**
Supply-Voltage Range	1.8V - 3.6V	7V - 12V	2V - 3.6V
Digital I/O pins	24 (+)	14 (~)	53 (+)
Operating Temperature Range (C)	-40 - 105 (+)	-40 - 85 (~)	-40 - 85 (~)
Flash Memory	56 KB of FLASH and 4KB of RAM(+)	32 KB of Flash Memory(~)	64 KB of Flash Memory(+)
Size (WxL) (mm)	50 x 66(~)	53 x 69(~)	12 x 15(+)
Price	~$5(+)	~$23(-)	~$5(+)
Score	**23**	16	23

Table 3.3.6 Microcontroller Trade-Offs

3.3.7 Software Trade-Offs

For software trade-offs, we have three types of platform: Xcode, Visual Studio and Eclipse. Xcode requires less processing time and will run faster than the other two. Since software is mainly used for the graphical user interface (GUI), plugins, particularly for constructing the GUI, should be taken into consideration. After research, Xcode is found that, though it has fewer plugins, it is more reliable. This is especially important since the desired goal for this project is to implement our application on a smartphone. Therefore, we chose Xcode as software platform.

Software Platform	**Xcode**	**Visual Studio**	**Eclipse**
CPU Cost	Less space(+)	Plenty of RAM(-)	More CPU cost with Plugins(~)
Speed	Fast(+)	Debug Slow(-)	Moderate(~)
Plugins	Fewer plugins(+)	Some plugins, may or may not be reliable(-)	Some plugins are not reliable and lack of quality control. (-)
Language	C++, Objective C, Swift	C++, C#	C++, Java
Implementable on Mobile Device	Yes, iPhone(+)	No(~)	Yes, Android(+)
Library to Interface with Device	embedXCode (built specifically for MSP430)	VisualGDB	msp430gcc
Score	20	6	12

Table 3.3.7 Software Trade-Offs

3.3.8 Changes to PDR

There were a few changes made by the CDR date to the content presented by the PDR date. These can be seen in the following Figure 3.3.8.

Change made	**Description**
Pre-made sensors to self-made sensors	The sensors to be used in the RSM were determined to be pre-made. However, one of the team members pointed out that it was possible to create our own through an

	alternate way. After another trade-off, we decided to make our own sensors.
Removal battery backup	Since the RSM will now be able to receive the real time information from the software, the battery power needed to power the real time clock was minimized and the battery backup was removed from the system.

Figure 3.3.8 Table showing changes made from PDR presentation

4.0 Top-Level Design of Remote Sensor Module

4.1 RSM Solidworks Drawings

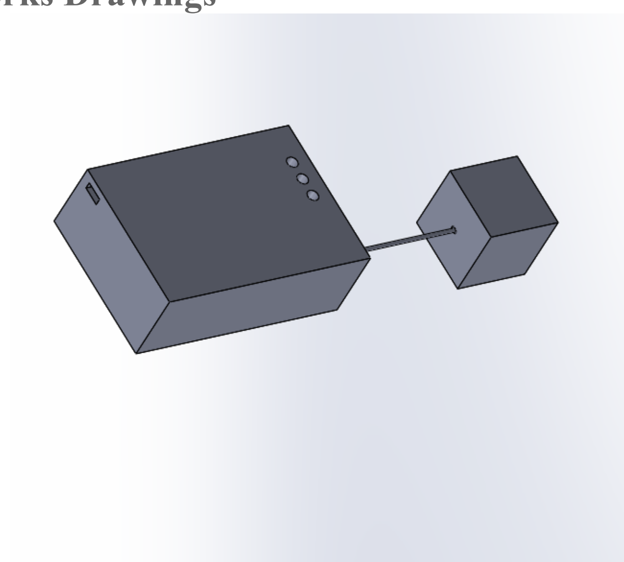


Figure 4.1.1 Troubleshooting Device SolidWorks Drawing

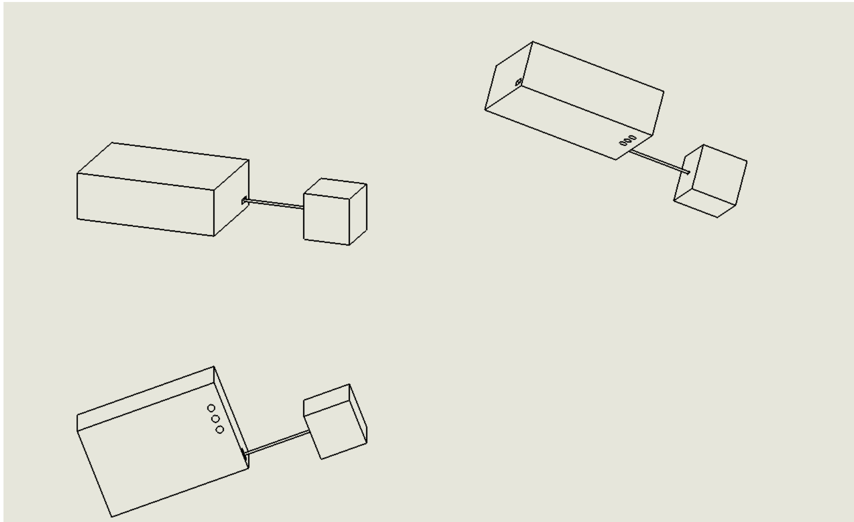


Figure 4.1.2 Three View Diagram of the Device (with an angle)

Both *Figure 4.1.1* and *4.1.2* are the schematics for the troubleshooting device system. In *Figure 4.1.1*, it is obvious that the system contains three major components: a RSM, a cable and a sensor box.

The biggest part in the figure is the RSM module which contains microcontroller and battery. There are three small holes on the module which indicate the battery level of the battery inside. The LED lights will be placed inside these three holes. The top one will be red, which indicate that the battery need to charge as soon as possible. The middle one will be yellow, which indicate that the battery is getting low. The lower one will be green, which indicate that the battery is good for at least 24 hours run time. The cable will have two standardized port at each end. The side to connect with the RSM module will be a standard USB port. The side that connect with the sensor box will be a micro-USB port. The cable will help the RSM module and the sensor box to transfer signals with each other. The smaller box in the figure is the sensor box, which include a circuit board and sensor inside. The circuit board will help to transfer signals into voltage data for the microcontroller to process.

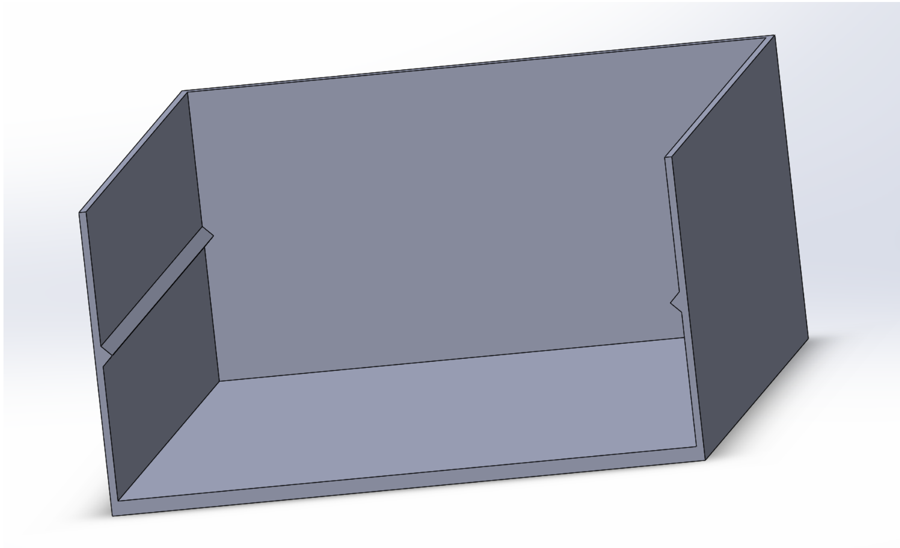


Figure 4.1.3 Slidable Design for Installation

Figure 4.1.3 is the rough slidable design for the installation purpose of the module. Slidable design will let the module be able to attach to a lot more common industrial surface than just a simple suction cups or tapes. The slidable case will be attach to the surface of machine that need to tested by using the screws. There will be four screw holes on the bottom of the slidable case at each corner. The final design will be determined in next semester after couple testing with the existing design.

Since the design contains drew four holes on the surface that need to attached onto. There are some materials that would not allow us to have this action. For example, if taking the glass into consideration, the fragile part of characteristic of glass will not allow the device install on it. The solution for that would be putting an additional adhesive material on the bottom surface of the slidable case so that it could stick on the glass surface. But as state in the last paragraph, the final design of the slidable case will be determined in semester.

4.2 RSM Schematic

The schematic shown in *Figure 4.2.1* is the complete design that will be implemented in the next phase of this project. The three circuits on the right of the figure are for three sensors. From the top: current sensor, Light sensor (can also be used for detecting motion and fluid level), and Temperature sensor. The MSP430G2553 (IC3) is the MCU that process and store data sent from the sensor. The CP2101 is the bridge communication chip between the PC and the MCU. The LP38512 is the voltage regulator that ensures a 3.3v to the rest of the circuitry. The LM3658is the Li-ion battery charger.

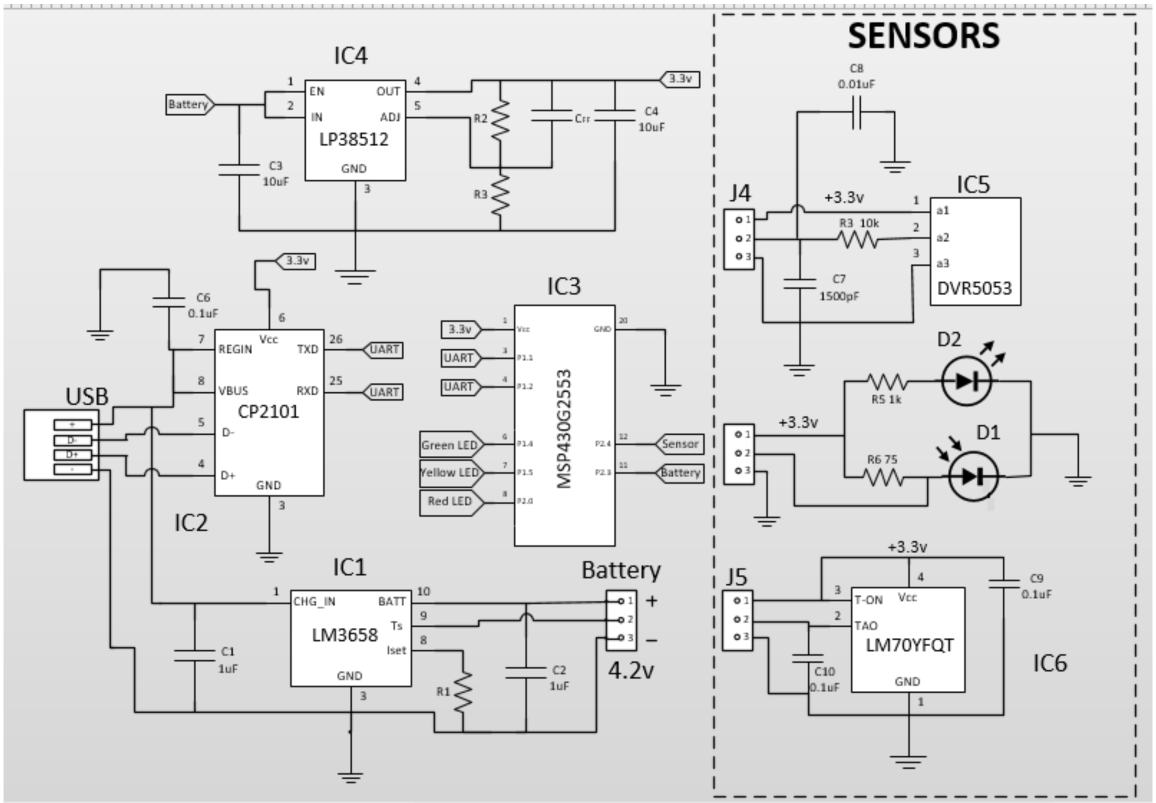


Figure 4.2.1 System Schematics

5.0 Subsystem Design

5.1 Module Subsystems

5.1.1 Battery Charger

The Lithium-ion battery powering the RSM requires a specific charger to ensure that the battery is safely charged to 4.2v in a wide temperature range. The circuit in *Figure 5.1.1* uses LM3658 chip which is equipped with a battery thermistor interface to continuously monitor the battery temperature by measuring the voltage between the TS pin and ground. This feature is crucial in maintaining battery's health because in environments where the temperature is under zero degrees celsius, it is not safe to charge the battery.

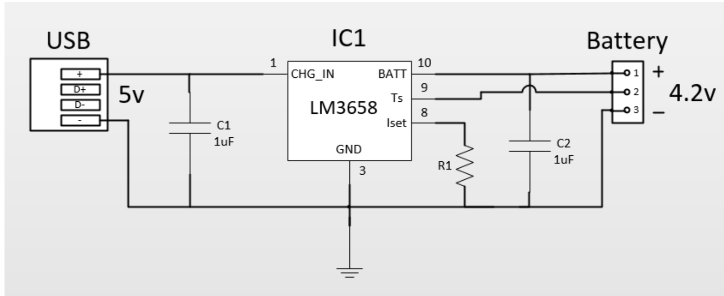


Figure 5.1.1 Battery charger

5.1.2 Voltage Regulator

The purpose of this circuit (*Figure 5.1.2*) is to supply the microcontroller and the sensors with a constant 3.3v dc. The LP38512 is capable of delivering a maximum of 1.5A output load current. Has a short-circuit protection. The output voltage can be calculated by the equation:

$$V_{OUT} = V_{ADJ} *(1+(R_2/R_3)) \text{ (Equation 5.1.1)}$$

V_{ADJ} is 0.5v if $2.5v < V_{IN} < 5.5v$

$R_2 = 2k$ ohm

$R_3 = 357$ ohm

TI recommends that C_{FF} is 2700pF to improve load transient response of the device.

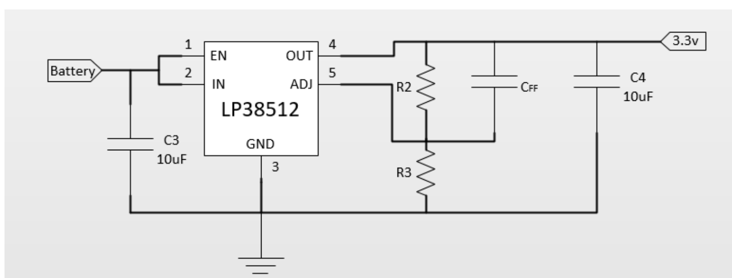


Figure 5.1.2 Voltage regulator

5.1.3 USB to UART Bridge

The CP2101 is USB transceiver chip that ensures a reliable data transfer between the user's PC and the microcontroller (MSP430G2553). The schematic of the circuit is illustrated in *Figure 5.1.3*.

- USB Specification 2.0 Compliant; Full Speed (12 Mbps)
- Baud Rates: 300 bps to 921.6 kbps

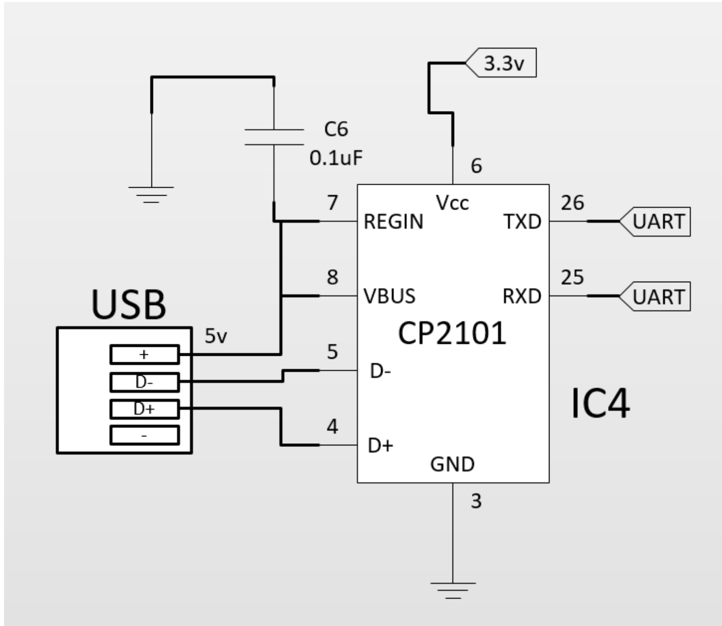


Figure 5.1.3 USB to UART bridge

5.1.4 Microcontroller

The MSP430G2553 (*Figure 5.2.4 and Figure 5.2.5)* is ultra-low-power mixed signal microcontroller with built-in 16- bit timer, up to 24 I/O capacitive-touch enabled pins, a versatile analog comparator, and built-in communication capability using the universal serial communication interface. In addition the MSP430G2x53 family members have a 10-bit analog-to-digital (A/D) converter. [**Reference**]

This microcontroller will be used to process and store data collected by the sensors over a period of 24 hours.

- 16kp flash memory
- 512b RAM
- 8 Channels 10-bit ADC
- Ultra-low power consumption
  - Active mode: 230uA at 2.2v, 1MHz
  - Standby mode: 0.5uA

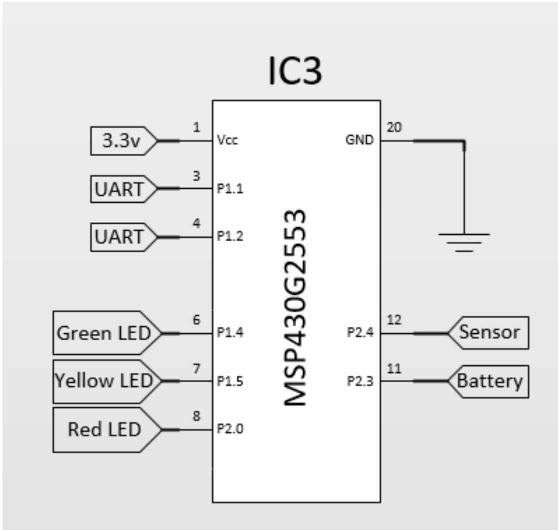


Figure 5.1.4 Microcontroller Circuit

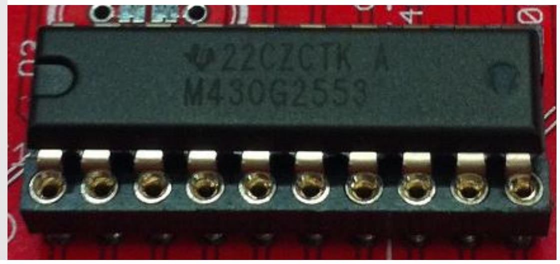
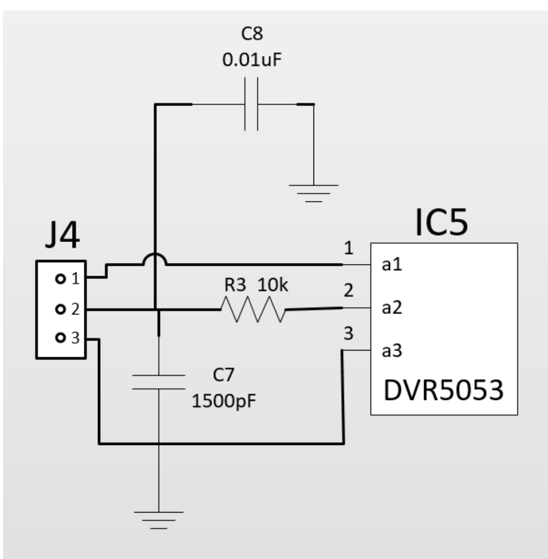


Figure 5.1.5 Microcontroller

5.2 Sensor Subsystems

5.2.1 Current Sensor

The DVR5053 used in the circuit in *Figure 5.2.1* is for an analog-bipolar hall-effect current sensor. It sense the magnetic field around it and output a certain voltage according to the strength of the magnetic field. The importance of using a hall-effect sensor is to avoid any interference or interruption to the under test running devices. Therefore, an iron core flux concentrator shown in *Figure 5.2.2* is used to concentrate the magnetic field around the cable that supplies power to the device under test. The magnetic field will be concentrated on the DVR5053 chip to achieve the best sensing results.



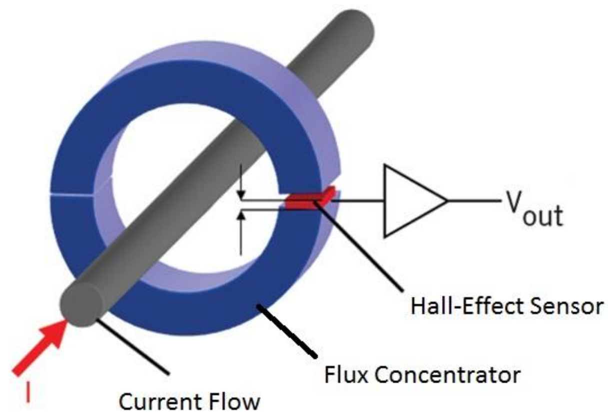


Figure 5.2.1 Current sensor Circuit *Figure 5.2.2 Iron Core Flux Concentrator*

The plot in *Figure 5.2.3* shows the sensor response to magnetic field magnitude (B). If no magnetic field applied to the device then VOUT will be 1v. Otherwise, if any magnetic field is applied near the chip then VOUT will change accordingly. This change in VOUT will be detected by the MSP430 through the ADC, processed, and then stored for later data analysis.

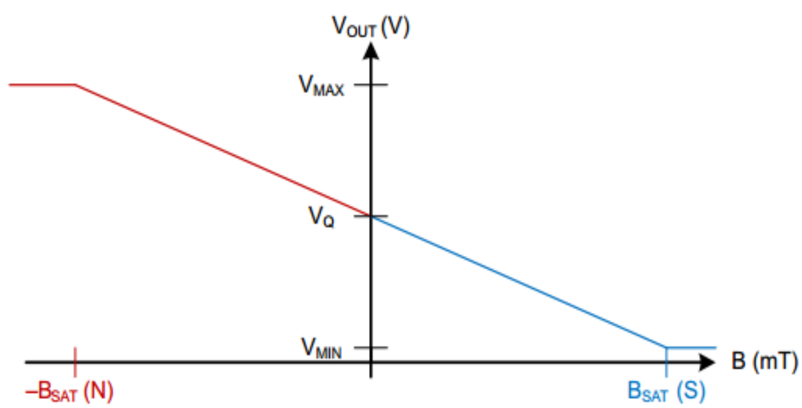


Figure 5.2.3 Sensor Response to magnetic field

5.2.2 Temperature Sensor

The circuit in *Figure 5.2.4* uses an ultra-small high precision temperature sensor chip LMT70.

- ultra-small chip (only 0.88x0.88mm)
- low power consumption: < 12uA at 3.3v
- high temperature reading accuracy: ± 0.3 °C.
- Output voltage resolution: 5.194mV/°C.

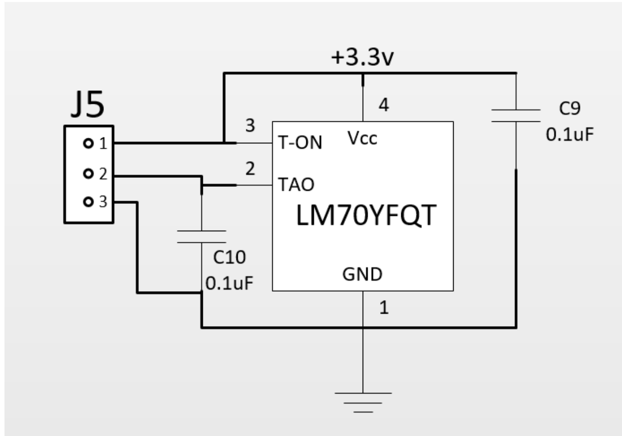
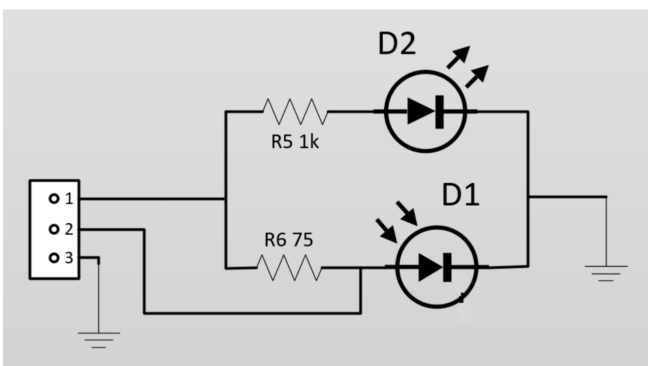


Figure 5.2.4 Temperature sensor

5.2.3 Motion Sensor

The circuit in *Figure 5.2.5* is a simple IR-transceiver. The circuit uses a two infrared LEDs, one is the transmitter and the other is the receiver. The circuit will output variable voltage at pin#2 according the strength of the reflected infrared light. This sensor can be utilized to detect moving objects, distance of objects, and fluid level detector. The circuit will be powered by a 3.3v at pin#1. *Figure 5.2.6* shows an example of possible design of this sensor.



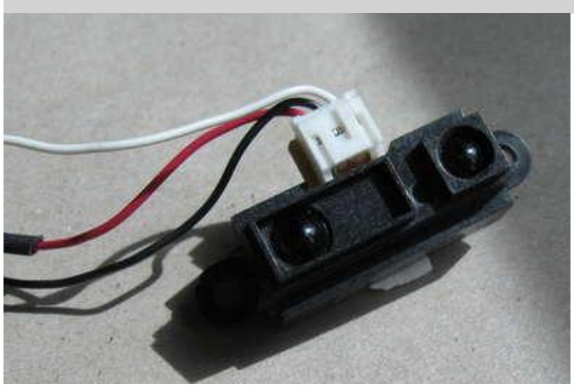
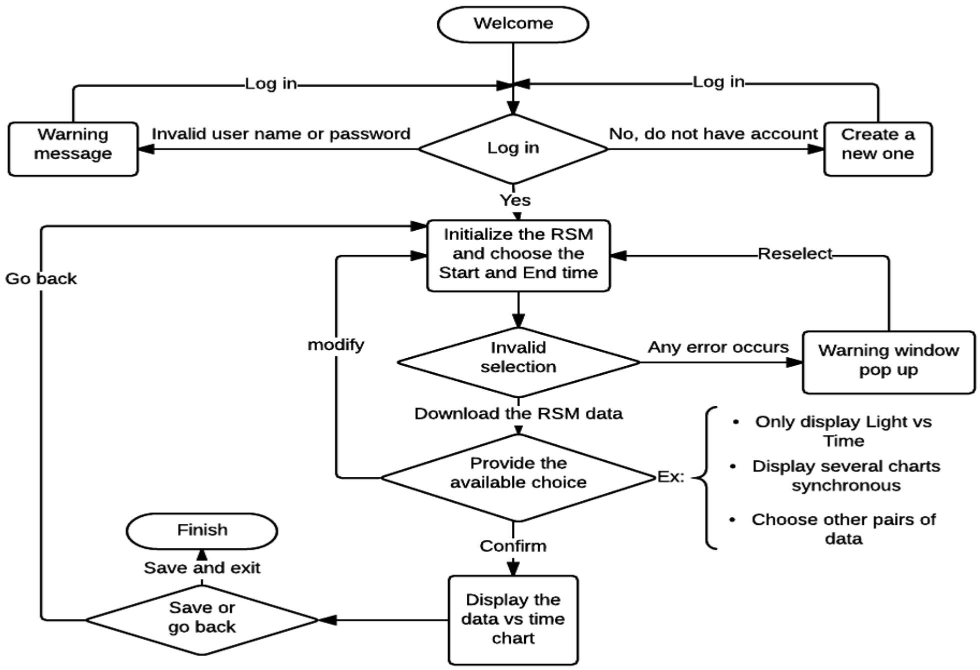


Figure 5.2.5 Motion sensor Circuit *Figure 5.2.6 Motion Sensor*

5.3 GUI Subsystems

Figure 5.3.1 GUI Design Flowchart



The Graphical User Interface (GUI) will provide user-friendly environment for RSM customers the gather and save the data for further use. Specifically, we have four major for this interface. Firstly, the user must be able to log in using very specific credentials. Upon logging in, that user then is given the ability to either import new sensory data or program their RSM. Lastly, the user can export the sensory data into a more convenient format, such as a csv (comma separated value) file.

Figure 5.3.2 is a flowchart for GUI processing. It the initial node the user is asked to enter credentials. The user also has the option of creating a new account. The user can then either select from previously gathered data to programming their RSM to collect new sensory data. The user can then terminate their session having the option to export their new found data.

5.3.1 Current Graphical Flow

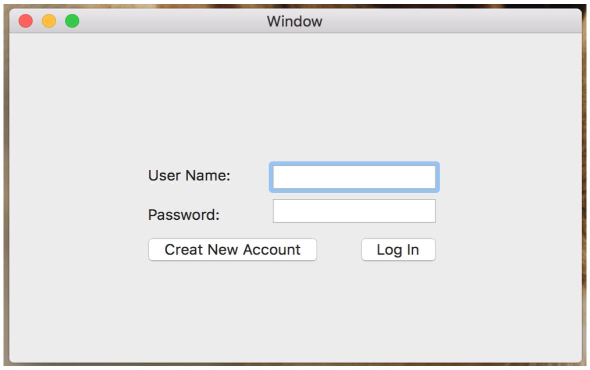


Figure 5.3.2 Log-In Window

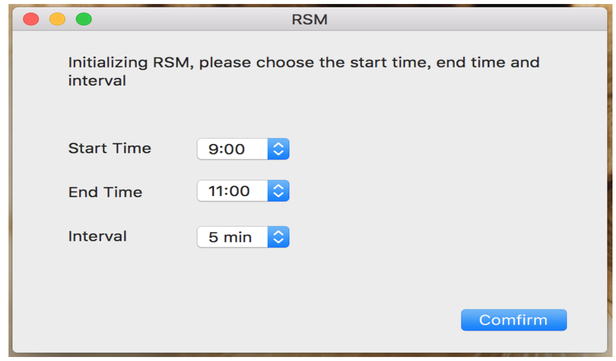


Figure 5.3.3 Initial RSM Window

The user will be asked to login an account credentials (*Figure 5.3.2 Log-In Window*). If they do not have one, they could create a new account. After that, the RSM will be initialized, and user could choose start time, end time and interval(*Figure 5.3.3 Initial RSM Window*).

Then the GUI will check whether the information that the user chose (*Figure 5.3.4 Selection Window*).

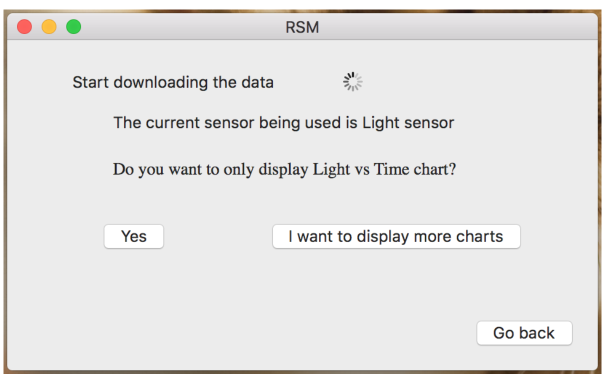


Figure 5.3.4 Selection Window

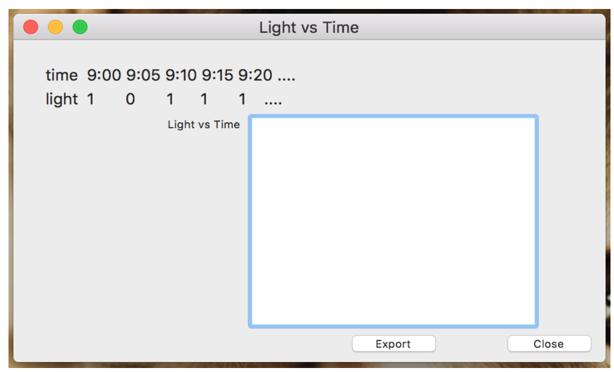


Figure 5.3.5 Display Chart Window

With the valid choice, a pop up window will show for user to choose the chart he wants to display(*Figure 5.3.5 Display Chart Window*).

Lastly the use can choose to export their data and close the application.

6.0 Development Design

Since there are no major deliverables for the second semester of senior design, the team has created their own build deadlines. The first deadline is to finish placing orders by January 25, 2016. The remaining deadlines are the dates for when the system development, system testing (acceptance test plan), testing environment development and posters for design day should be completed by.

6.1 System Development

The first category of building milestones was for the system development, the casing and the slidable design should have its first attempt module done on the CNC machine by February 15,2016. The electronic part should be complete with its first design done by March 1,2016. Software program should be done as the same day as the electronic part. At the same time, people who is in charge of their individual part should do some small testing, such as whether the GUI will work when there is a input data. On March 10, 2015 the system should come all together as a completed module. This would allow the team have a significant amount of time for testing and further improvement of the prototype.

6.2 System Testing

After the initial subassembly is created, then testing, from the Acceptance Test Plan, can be implemented to verify the device is working as intended. After testing, there will be mitigation plan for each possible circumstances that can come up during the testing. People who is in charge of the failed part will need to fix it within an acceptable time. The final version of the module should be completed by April 20, 2016. The final version of the module should work as expected with all requirement met. The final test time for the system testing will be April 25, 2016.

6.3 Testing Environment Development

The next category for the development plan is the testing environment development. Due to the limitation of the design day, team should start building a testing environment for the design day to showcase the design on March 21,2016. Before start building the testing environment, team should have a meeting to discuss about the possibility of building all five testing environment or not. If not, team should choose the one with most important feature of the module to build. The final testing environment module should be ready by April 25, 2016.

Finally the team would like to have a date for building the poster for the showcase on design day. The poster should be start building on April 20, 2016. It will give enough time for the team to think about what kind of information they would need to put on the poster to show to the audience. The poster should be finished by April 29 with the finalized module ready for showcase. Bothe the poster and the module should be ready to review by the sponsor and the mentor. After that, team should get ready to write the final report and the practice the presentation on the design day May 3, 2016. For further and more specific development plan, refer the A.3 Responsibility Matrix in the Appendix.

6.4 Implementation of development plan

In order to present the status of all milestones in this development plan, Figure 6.4.1 was created. This table shows whether the implementation plan was followed as scheduled and each milestones current status.

Milestone	**Target date**	**Status**
1st module case	02/15/16	DONE
1st PCB	03/01/16	DONE
1st complete module	03/10/16	DONE
Start testing environment	03/21/16	DONE
Start poster	04/20/16	DONE
Final complete module	04/20/16	DONE
Final testing day	04/25/16	DONE
Final testing environment	04/25/16	DONE
Final poster	04/29/16	DONE
Date updated: 05/04/16		

Figure 6.4.1 Development plan milestone table

It is important to note that this only mentions if the milestone dates were met. This does not mention which obstacles were presented in the implementation and why certain items were late. The following paragraphs specify both of these aspects.

For the completion of the module, one of the biggest issues was the casing. Initially, the casing was supposed to be developed from carbon fiber. Given the difficulty of manufacturing with carbon fiber, the material was then switched to teflon (PTFE). There was also a delay in receiving the PTFE as the order was somehow lost. Once the material was obtained and an initial casing developed, a redesign in order to accommodate a stronger battery and easier accessibility delayed the casing even more. Due to the casing, our first module will be our final module and has therefore caused our first and final module milestones to not be met on the desired date.

The team also faced a communication error between the software and the PCB. This may be due to damage to the board or a slight error in wiring from the microprocessor to the board. The problem is in process of being fixed and if not finished by Design Day, it will be mentioned. NOTE: the RSM still works as planned, this only affects a small cosmetic issue with the module. While all systems in both the software and hardware areas are fully functioning, this has delayed the completion of the modules as desired.

6.5 Testing Results

This section will detail the results of the Acceptance Test Plan (see Appendix A.2) and the performance of the module in meeting the requirements set at the beginning of the project's life cycle.

Comparing the results of the final module with Systems Requirement Verification Matrix will allows us to see if the final RSM met the requirements set at the beginning of the project.

Req. #	Requirement Type	Final RSM Check
	Interface	
2.4	GUI Display	TBD
2.6	Module-Sensor interface	TBD
2.8	CMI	TBD
	Technology	
2.2	Wireless info transmission	TBD
	Performance	
2.2	24 hour operation	TBD
2.5	Reliable data transmission	TBD
2.7	Maximum 2 batteries	TBD
	Mechanical	
2.1	Dimensions	TBD

Table 6.5.1 SRVM Table with requirements check

The Acceptance Test Plan has an addition in the Appendices. See Appendix A.3 to see the complete results of the testing. In order to summarize these results, the following figure was developed.

FIGURE INSERTED AFTER FINAL TESTING

7.0 Risk Analysis

7.1 Risk Analysis Table

This risk analysis table, Figure 7.1.1, is an organized list of the potential risks for the Remote Sensor Module project. Also present in this table is the likelihood and consequence of each individual risk. They are split into categories as follows:

Likelihood: Rare, Unlikely, Moderate, Likely, Certain

Consequence: Insignificant, Minor, Moderate, Major, Catastrophic

This will allow the team to categorize the value of the risk and deem what is the necessary mitigation plan for that risk. The mitigation plan can also be found in this table.

Risk #	Risk	Likelihood	Consequence	Mitigation Plan
	Technical risks			
1	Battery life unable to fulfill minimum run time	Likely	Minor	Redesign system to allow battery to fulfill run time
2	RSM sensors provide inaccurate data	Unlikely	Major	Calibrate system. If not working after calibration, replace sensors
3	Data transfer from RSM to GUI fails	Rare	Major	Calibrate and troubleshoot system. Re-test module connections with PC
4	RSM unable to withstand extreme conditions	Likely	Minor	Make sure system is sealed. Improve insulation and sealant if needed
5	CMI unattachable to surface	Unlikely	Moderate	Determine better system for CMI
6	Dimensions not viable	Likely	Major	Determine new dimensions in requirements

	Project Management Risks			
7	Poor management results in deadline not being met	Rare	Catastrophic	Resolve management issue ASAP. Delegate management responsibilities
8	Change in requirements after SRR	Likely	Minor	Change requirement. Minimize delay.
9	Misallocation of funds will cause team to go over budget	Unlikely	Catastrophic	Budget research comparison. Check with Purchasing manager regularly
10	Software/Hardware needed for project become unavailable	Rare	Major	Search for alternatives and implement ASAP. Find mentor for education of software
11	Miscommunication with sponsor causes problems	Unlikely	Catastrophic	Establish new communication standards
	External Risk			
12	Testing system is not prepared by Design Day	Unlikely	Major	Alternative system or present data
13	Suppliers unable to send materials on time	Unlikely	Major	Research other vendors and check on shipping times
14	Low market demand for product	Moderate	Minor	Lower price and increase functionality
15	Manufacturability of product	Unlikely	Minor	Make sure materials can be manufactured. Communicate with manufacturing department
16	Exceed sponsor product cost	Moderate	Major	Establish new sponsor cost

Figure 7.1.1 Risk Analysis table

7.2 Risk Register

The risk register allows for the risk identified within the risk table to have a quantitative value that determines how much weight an individual risk has, therefore allowing the team to focus on the most prevalent risks and mitigating them. The categories mentioned for Likelihood and Consequence were given percentage values and inserted into the Probability and Impact sections, respectively. See Figure 7.2.1 for the risk register.

Analyzing the risk register allows the project team to see that the most important risks are the battery being unable to fulfill the run time, the dimensions of the RSM not being viable, and going over the cost of the product that Texas Instruments has set. These are risks that have large impact values and need to be mitigated and watched closely.

Risk #	Description	Category	Probability (P)	Impact (I)	Risk Factor
1	Battery life unable to fulfill minimum run time	Technical	0.5	0.85	0.425
2	RSM sensors provide inaccurate data	Technical	0.05	0.75	0.0375
3	Data transfer from RSM to GUI fails	Technical	0.01	0.85	0.0085
4	RSM unable to withstand extreme conditions	Technical	0.5	0.05	0.025
5	CMI unattachable to surface	Technical	0.05	0.35	0.0175
6	Dimensions not viable	Technical	0.65	0.9	0.585
7	Poor management results in deadline not being met	Project Management	0.01	0.95	0.0095
8	Change in requirements after SRR	Project Management	0.85	0.05	0.0425
9	Misallocation of funds will cause team to go over budget	Project Management	0.05	0.95	0.0475
10	Software/Hardware needed for project become unavailable	Project Management	0.01	0.75	0.0075
11	Miscommunication with sponsor causes problems	Project Management	0.05	0.95	0.0475
12	Testing system is not prepared by Design Day	External	0.05	0.85	0.0425
13	Suppliers unable to send materials	External	0.1	0.85	0.085

	on time				
14	Low market demand for product	External	0.35	0.05	0.0175
15	Manufacturability of product	External	0.05	0.05	0.0025
16	Go over sponsor product budget	External	0.5	0.55	0.275

Figure 7.2.1 Risk register

7.3 Risk Severity Matrix

Figure 7.3.1 is a Risk Severity Matrix and is an extension of the previous tables allowing the project team to visualize what risks are the most prevalent for the project. The risks that were previously mentioned are in the red zone along with other risks. These risks include miscommunication with the sponsor misallocation of funds. The team has to also keep these risks under scrutiny a need to be mitigated throughout the project life cycle.

	Insignificant	**Minor**	**Moderate**	**Major**	**Catastrophic**
Certain					
Likely		**4,8**		**1,6**	
Moderate		**14**		**16**	
Unlikely		**15**	**5**	**2,12,13**	**9,11**

Rare				**3,10**	**7**

Figure 7.3.1 Risk Severity Matrix

8.0 Budget and Bill of Materials

The following table shows the price for each relevant component of the RSM. In order to determine the actual cost of making the product, we considered adding the price per 1,000 orders of the item, as this will greatly decrease the cost of the product. Table 8.1 shows the information in more detail.

Part	Price /1 item	Price /1k
MSP430G2553	$2.60	$1.00
32KHz Crystal	$1.20	$0.65
Green LED	$1.70	$0.56
Red LED	$0.45	$0.29
Yellow LED	$1.12	$0.35
Battery	$5.00	$2.50
CP2101	$3.45	$2.95
Capacitors X 10	$1.00	$0.02
LM3658	$1.38	$0.50
Resistors X 6	$1.00	$0.20
LP38512	$2.05	$0.85
DVR5053	$1.27	$0.35
IR LED x 2	$2.00	$1.00
LMT70	$3.19	$0.85
RSM case	$7.00	$2.00

Sensor case	$4.00	$0.90
Third party USB Charger	$7.50	$3.50
Total	**$45.91**	**$18.47**

Table 8.1 Cost of product table

As is seen, we are greatly under the cost for making the RSM both with individual parts as well as per 1K orders. Since these are the major costs incurred, it is also a verification that our $3,500 budget for the project will not be exceeded.

9.0 Conclusion

9.1 Next Steps

The following Figure 9.1.1 shows the schedule of deliverables for the following semester. The team has completed all duties up to the Critical Design Review and has now scheduled all activities for the construction phase of the RSM. The Gantt chart Figure 9.1.2 visualizes the schedule showing the team which order to create the deliverables. Refer to the development plan in section 6 for a more detailed explanation of the next semester.

	i	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1			**Remote Sensor Module**	**148 days**	**Thu 9/3/15**	**Tue 5/3/16**	
2	✓		**Systems Requirement Review**	**21 days**	**Thu 9/3/15**	**Thu 10/1/15**	
9	✓		**Preliminary Design Review**	**19 days**	**Fri 10/2/15**	**Wed 10/28/15**	**2**
21	✓		**Critical Design Review**	**27 days**	**Thu 10/29/15**	**Fri 12/4/15**	**9**
30			**RSM Construction Phase**	**81 days**	**Mon 12/7/15**	**Tue 5/3/16**	**21**
31			Module construction	25 days	Mon 12/7/15	Mon 2/15/16	29
32			Sensor construction	20 days	Wed 2/3/16	Tue 3/1/16	29
33			GUI construction	20 days	Wed 2/3/16	Tue 3/1/16	29
34			Preliminary system testing	5 days	Wed 3/2/16	Tue 3/8/16	31,32,33
35			Final assembly	31 days	Wed 3/9/16	Wed 4/20/16	34
36			Final system testing	3 days	Thu 4/21/16	Mon 4/25/16	35
37			Environment construction and testing	30 days	Mon 3/21/16	Fri 4/29/16	34
38			Final poster	7 days	Thu 4/21/16	Fri 4/29/16	35
39			Design Day	1 day	Tue 5/3/16	Tue 5/3/16	38

Figure 9.1.1 Schedule of Deliverables Spring 2016 semester

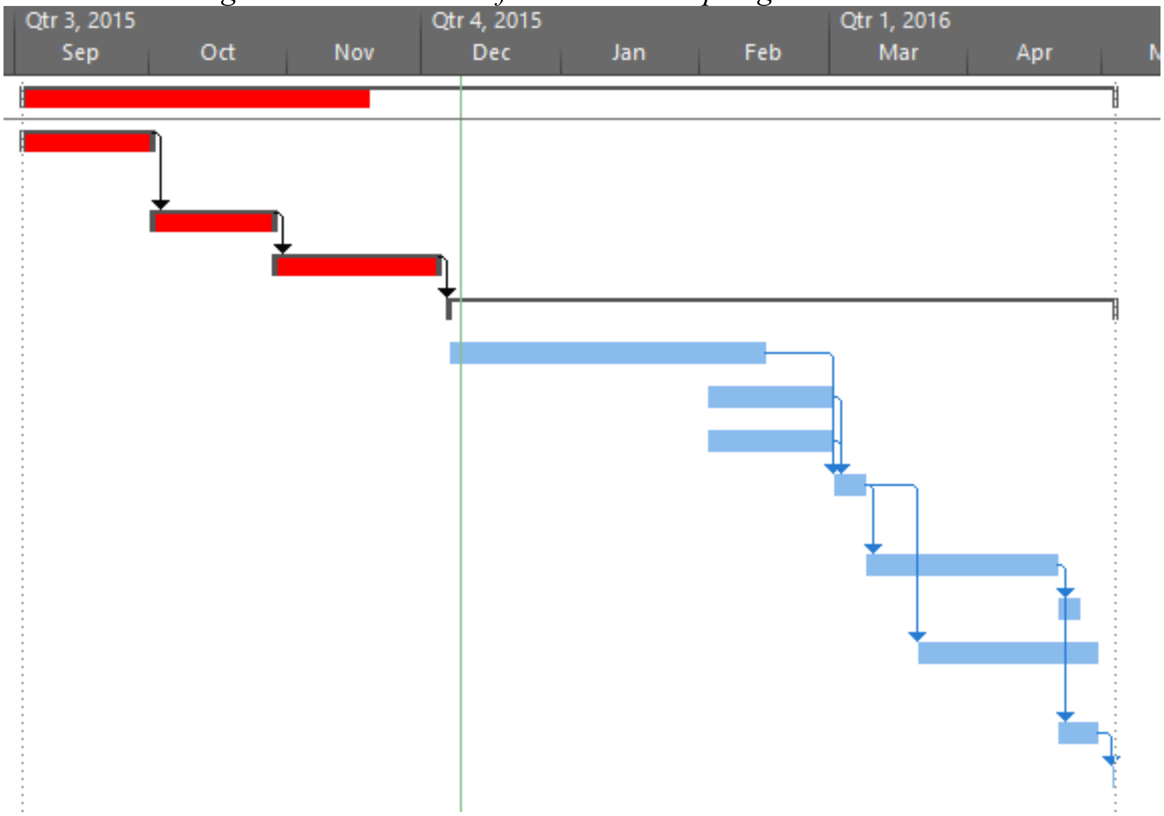


Figure 9.1.2 Gantt chart showing project plan

UPDATE: Now that the project has reached completion, we were able to determine future steps that would benefit the project and meet the desired vision of the project sponsor.

9.2 Summary

Team 1555, sponsored by Texas Instruments, is creating a Remote Sensor Module for their senior design project. At the present, the team has collected and developed requirements, risks, constraints, and tests to design and build the RSM. The system development and design phase are completed, and the remaining academic year will be dedicated to building and implementing the device. The RSM allows the user to observe data such as temperature, motion, current, light, and fluids while demonstrating the performance of Texas Instruments' MSP430 MCU and other TI precision components. The final design was presented on Design Day, May 3, 2016.

Appendices

A.1 Concept of Operations Table

Section	Title	Key Elements
1	Introduction	The objective will be to design and test small sensor modules with USB interface capability that can be externally attached to any type of industrial machinery or system to aid in troubleshooting. The sensor modules will be able to measure and record light, pressure, position, and temperature over time and at resolution levels permitted by on-board memory. The data will be down loaded to a PC or transmitted wirelessly to a handheld device (cell phone) for analysis by the field technician or engineer.
2	Stakeholders	Texas Instruments Technicians
3	Product History (Background)	TI has sensors for this reason. Sensors not together Sensors don't record data. this product will be able to sync all this data Integrates software with sensors in order to better understand data ● GUI will allow ease of data analysis Easier reading of data. Ex: "Technician only needs this one product to read various data" Portable Independently powered Time controlled Interchangeable sensors/combined sensors
4	System Boundaries	Xcode TI TINA Spice Solidworks Velcro/Suction Cup/Magnet/Tape PC App USB connections
5	System Environment	Attaches to metal, wood, plastic, etc Mostly indoor Sensor shall survive extreme machine conditions
6	System Constraints	$50 per module w/sensor. $200 for the kit (price dependent on interchangeability)

		Dimensions 3x2x1 3 to 5V battery ● Should last 24 hrs (minimum) ● Rechargeable ● Replaceable/Easily accessible ● Consider battery life (how often to buy new battery?) Don't want to take apart any other systems Common Mechanical Interface Power on indicator Interface to PC ● Read what's on the memory ● Specify units/types of measurements ● Application that sets time/schedule Memory ● Must have storage space for 24 hr run time Timer/Clock Cost ($3500)
7	System Use	Manual on/off switch Stand by Automatic setting/schedule ● Set time and check every interval Maintenance ● Calibrate sensor accuracy and sensitivity ● Battery ● Update software Explanation of how the system will be used including: ● Users ● Timelines
8	Expected Output	Expected to collect data from environment. Transfer to memory. Send to application in the correct units and time interval. ● GUI is easy to use ● Bandwidth Validation ● Will be validated through field testing on all relevant surfaces and machinery ● Sponsor will determine success of prototype

A.2 Acceptance Test Plan

The acceptance test plan is the the test plan the team have came up to test the characteristic of the module. Each test plan is designed to test each requirement that have been addressed in the requirement report. There will be three different types of test for the test plan: inspect, test and analyze. Table 2.6.1 provides a basic information about the which testing verification has been chosen to test the requirement. Table A.1 above shows the method of each test. All of the test plan will be taking action next semester when the team finished the first module. If the module not meet the requirement, there will be a mitigation plan for each single requirement.

Majority of the inspecting test can be tested by using basic measurement tools, such as caliper, scale or just simple plug-in plug-out action. There are some requirement that might need to have the team to find someone with no background knowledge of the project to test the probability or the easiness of the installation of the module. The team try to make the module as much user friendly as possible.

Testing part can be monitored by observe the existence of the data or signal. Since majority of the testing part is related with sensors, the team need to find a software or hardware that can read the signal from the sensors. If the sensor does not meet the requirement, team member should do adjustment based from the result. Same concept would apply to the GUI also. By the end of February 2016, team should have a test data file for the software to display on the GUI. If the GUI does not display the data from the file, team member should do adjustment in the code to make sure the software is capable to read the data file and display it on a GUI chart. And according to the requirement, the battery should last for at least 24 hours of operation. In order to test whether it can provide the power for the module for 24 hours of operation, there should be have some test run to see how long the battery can last.

Analyzing test is the most precise tests out of the three different types of tests. Most analyzing tests need to use control variable to determine whether the sensor would provide a reliable data or not. Team will come up with some steady or easy controlled environment to test the sensitivity of sensors. If it does not meet the requirement, team should come up with a different design for the sensors. As for GUI display, apart from observe whether the software is capable to display the data, the accuracy of the software need to take into consideration. If the GUI does not display the accurate data from the file, team member should do adjustment in the code to make sure the software is capable to read the data file and display the accurate data on a GUI chart.

Requirement	**Test plan**	**High-3**	**Medium-2**	**Low-1**	**Min Score**
All the parts have to fit into the casing which is 3x2x1 inches	Use a caliper to measure the dimension of the RSM	Smaller than 3x2x1 cubic inches	About 3x2x1 cubic inches	Larger than 3x2x1 cubic inches	2
The RSM should be less than 4ozs	Use a scale to measure the weight of the RSM module	Less than 4ozs	About 4ozs	Greater than 4ozs	2
The RSM should last for at least 24 hours operation	Do a test run for the RSM to run continuously	More than 24 hours	About 24 hours	Less than 24 hours	2

	without any interruption				
The sensors should collect data accurately	Do a test run of a known variables (eg. ice water mixture) to see whether it gives a correct data or not	Greater than 95% accuracy	Approximately 95% accuracy	Lower than 95% accuracy	3
GUI shall display the accurate data from the field	Do a test run of a known variables (eg. ice water mixture) to see whether the GUI will display the correct data or not	Display all accurate data	Display data, but inaccurate	No data displayed	3
The RSM should provided a reliable data transmission	Do couple control variable test to monitor whether the RSM will collect and display the correct data in GUI	GUI displayed accurate data	GUI displayed some accurate data points	GUI doesn't have data displayed	3
The RSM-sensor interface should be allow the different sensor to be attachable	Attach different types of sensors to the module using the standardized cable	All the sensors shares the common port	Some of them shares the same port	None of them is share the same port	3
The battery should be able to rechargeable	Do couple test run to see whether the batteries are rechargeable	100%	80-100%	<80%	3
The battery should be able to replaceable	Observe how difficult for people to replace the battery inside the module	1-2 min	3-5 mins	>5mins	2
The RSM should be able to attach to any types of common industrial surface	Do couple test run to see whether the RSM module can be attached to the common industrial	Attach to more than 8 types of the common industrial	Attach to 5-8 types of the common industrial surface	Attach to 3-5 types of the common industrial surface	2

	surface (such as: aluminum, steel, wood, glass,etc)	surface			
The sensor of the module should be easily attachable and detachable	Ask people with no background knowledge of this product to assemble the parts together to see how long it will take them to figure in/out	1-2 min	3-5mins	>5mins	3
The micro controller should be programmed to have a real time clock in order to synchronize the data with each other	Set the time with the laptop and make some random event happened during the measurement and record the time of the event occurred then check with the GUI later	The GUI shows the exact time of the random event happened	The GUI shows 1-2 minutes of delay	The GUI shows >2 mins of delay	3

Table A.1 Acceptance Test Plan

A.3 Responsibility Matrix

The Responsibility Matrix is the matrix shows the leadership role for each member in the team and how each single task has been assigned to the team. In this specific matrix I stands for Informed, R stands for Responsible, C stands for Consulted and A stands for accountable.

Tasks\Member	Jose Escobedo	Chenmu Li	Xinran Fang	Ezzulddin Naji	Gregory DePaul
Module Development	**I**	**R,C,A**	**I**	**I**	**I**
Electronic Development	**I**	**I**	**C**	**R,C,A**	**C**
Software Development	**I**	**I**	**R,C,A**	**C,I**	**R,C,A**
Meeting Minutes	**R,C,A**	**C,I**	**I**	**I**	**I**
System Testing	**R,C,A**	**R,C,A**	**R,C,A**	**R,C,A**	**R,C,A**
Environment Development	**R,A**	**R,A**	**C,I**	**C,I**	**C,I**

Poster	**R,C,A**	**C,I**	**C,I**	**C,I**	**R,C,A**

Table A.3 Responsibility Matrix showing the person in charge of the next semester's deliverables