MEMS Soil Moniter

DESIGN DOCUMENT

Team #5

Client
Dr. Halil Ceylan

Advisers
Shuo Yang & Dr. Yang Zhang

Team Members
Nathan Coonrod (Report Manager and Hardware Lead)
Kyle Kehoe (Communications Manager and Testing Lead)
Jacob Verheyen (Meeting Facilitator and Hardware Designer)
David Severson (Web Master and Reliability Lead)
Sok-Yan Poon (Timeline Manager)

Team Email
sddec18-05@iastate.edu

Team Website
sddec18-05.sd.ece.iastate.edu

Revised: 4/22/18 (v 0.0.2)

Table of Contents

1 Introduction 3

1.1 Acknowledgement 3

1.2 Problem and Project Statement 3

1.3 Operational Environment 3

1.4 Intended Users and Uses 3

1.5 Assumptions and Limitations 3

1.6 Expected End Product and Deliverables 4

2. Specifications and Analysis 4

2.1 Proposed Design 4

Power 5

Digital Peripherals 6

Analog Peripherals 6

Software 6

2.2 Design Analysis 8

3. Testing and Implementation 9

3.1 Interface Specifications 9

3.2 Hardware and Software 9

Hardware Descriptions 10

Software Descriptions 13

3.3 Functional and Non-functional testing 14

3.4 Results and Challenges 19

4 Closing Material 21

4.1 Conclusion 21

4.2 References 22

List of Figures

Figure 1: Functional Prototype Architecture

Figure 2: Software Flow Diagram

Figure 3: DAQ Prototype

Figure 4: Temperature Chamber

Figure 5: Humidity Chamber

Figure 6: Digital Multimeter

Figure 7: Digital Oscilloscope

Figure 8: DC Power Supply

Figure 9: Signal Generator

Figure 10: LCR Meter

Figure 11: Arduino IDE

Figure 12: Excel Formatted Data

Figure 13: Difference in Resistance Between DMM and DAQ

Figure 14: CSV Data in .txt File

List of Tables

Table 1: Functional Requirements and Test Numbers

Table 2: Temperature Independence Test Data

List of Definitions

CCEE: Construction, Civil and Environmental Engineering

CSV: Comma Separated Values

ECpE: Electrical and Computer Engineering

DAQ: Data Acquisition System

MEMS: Micro-Electrical-Mechanical Systems

MRC: Microelectronics Research Center

UUT: Unit Under Test

1 Introduction

1.1 Acknowledgement

This project is possible because of the time donated (directly and indirectly) by the students and professors involved. Special thanks goes to Associate Professor Dr. Tuttle, who graciously volunteered his time to help us recreate MEMS sensors at Iowa State's MRC facilities.

1.2 Problem and Project Statement

The Civil, Construction and Environmental Engineering (CCEE) department at Iowa State wants to utilize MEMS sensors for soil monitoring underneath pavement. Currently, the CCEE department is using expensive, unreliable sensors for monitoring temperature and moisture content of soil underneath roadways. In addition, many of these sensors do not come with data acquisition modules. This leads to researchers spending hours in the field with their own measurement devices collecting data.

The CCEE department wants to utilize a MEMS sensor developed by the Electrical and Computer Engineering (ECpE) department at Iowa State. This sensor is inexpensive and precise, but does not come with a data acquisition system. This project will create a data acquisition system for the MEMS sensors designed by the ECpE department. The system will gather soil moisture and temperature throughout the course of a month, which will be interpreted by the CCEE researches to make decisions about the structural health of civil construction projects.

1.3 Operational Environment

The sensors and data acquisition module will be used outside in the harsh Iowa environment. The sensors will be underground and connected to the module through cables. The module will be at the surface to provide access to the SD card and the battery pack.

The sensor will need to have a low propensity to corrode since it will be in the ground for the entire life cycle. The data acquisition module must be able to last one month in an average climate; approximately one month in roughly 60 degrees Fahrenheit will be our "test climate."

1.4 Intended Users and Uses

The users of this product will be the researchers from CCEE who are monitoring the soil underneath roads or pavement. The researches will be gathering data such as temperature change and moisture content change and utilizing that data to make decisions for the Iowa Department of Transportation.

The product is intended to be used in the outdoors. It is to be used only for gathering temperature and moisture changes throughout the course of two to six weeks underneath pavement. This product could also be used to measure temperature or moisture of soil in other settings if needed.

1.5 Assumptions and Limitations

Assumptions

- There will be four independent channels for sensors
- The unit will have a minimum battery life of one month in average Iowa temperatures
- The unit will log data with an SD card

- The range of resistance measurements will be approximately 200-300 ohms
- Capacitance measurement will be in the range of 0-40 pF, with a resolution of 40 fF
- The cable connecting the sensor to the data acquisition module will be no longer than three feet
- 15-minute sampling period

Limitations

- Limitations (such as voltage input, max output, etc.) will be determined after we meet with the sensor developer.

1.6 Expected End Product and Deliverables

The end-product will be a complete DAQ system that can measure resistance and capacitance via a MEMS sensor and store that information in 15-minute intervals for an entire month.

The MEMS sensors were provided to us by the CCEE department and were developed by the ECpE department. When the sensors were initially given to our team, they were found to be non-functional due to excessive scratching since they had been used on a prior project in the past. Our team has worked with the previous developers of the sensors and will be working with faculty in the ECpE department to recreate the sensors that were able to measure temperature and moisture content using photolithographic mask and process steps used in a previous project. Although we are recreating the sensors, we are not responsible for the sensors' durability performance in the lab or in the field and needing to recreate them was not in the original scope of this project. Sensor handling will be of utmost importance for our project team and proper handling methods will be followed.

The output of the sensors will be collected by the team's data acquisition unit. The unit will store that information onto an SD card and be readily available by researchers when they need to access the data.

2 Specifications and Analysis

2.1 Proposed Design

The team has discussed with the customer and agreed upon the following requirements:

Requirement	Threshold / Description	Test Number
Independent Sensor Channels	Minimum 4	N/A
Operating Environment	-10 to 100 degrees Fahrenheit	1D
Battery Life	1 month in fair temperature, 2 weeks minimum	1B
Data Storage / Format	Onboard SD card / CSV datalogging	1A
Resistance accuracy	0.5 Ohms	2A
Resistance sensitivity	0.5 Ohms	2B
Capacitance accuracy	40fF	2C

Capacitance sensitivity	40fF	2D
Max sensor cable length	3ft	N/A
Sample rate	Nominal 1 sample / 15 mins	1C
Environmental Test	1 month in Iowa conditions	3B
Lab Test	1 month in lab conditions	3A
Cost	No requirement	N/A

Table 1: Functional Requirements and Test Numbers

To fulfil the above requirements, a prototype embedded hardware system has been developed which will meet all requirements in a lab environment. That is to say, the prototype is not required to meet any environmental or battery life requirements although the design is intended to show that the requirements are attainable with good mechanical design and better understanding of the battery requirements once the working prototype has been fully measured and characterized.

The functional prototype hardware will consist of the functional blocks shown in Figure 1. A detailed description of each block is described below:

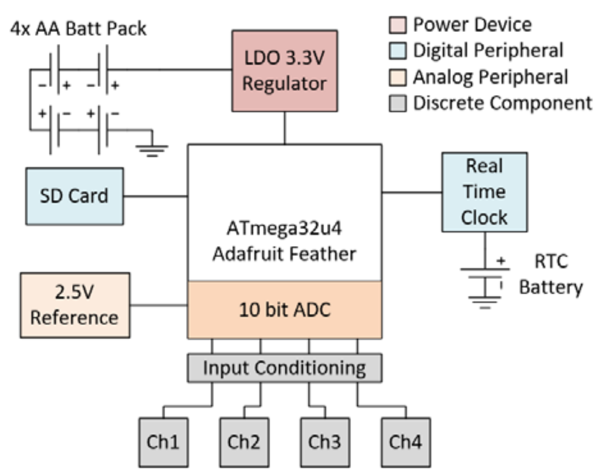


Figure 1: Functional Prototype Architecture

Power

The prototype will be powered by four AA cells in series for a total estimated capacity of 16.8Wh. This setup is not ideal given that a 3.3V LDO regulator is being used. At nominal battery voltage, regulation efficiency will be nearly 50%. This leaves great room for improvement on the final solution which can use a battery technology better suited to the application. The initial prototype requires an average of 20mA to operate the processor, real time clock, SD card, and ADC. Data acquisition of all four channels is estimated to take 10mS to complete. The microcontroller datasheet does not provide estimates of sleep current, so an estimated 100uA is used in the following calculation which includes regulator quiescent current, sleep current, and leakage. So with one sample every 15 minutes, the average draw is:

$$Average\ Draw = 100uA * \left(1 - \frac{4 * 10mS}{1hr}\right) + 20mA * \left(\frac{4 * 10mS}{1hr}\right) = \mathbf{100.2uA}$$

Clearly the battery life will be almost completely determined by the sleep current. In fact, increasing the time the processor runs from 10mS to 100mS on each sample only adds an additional 2uA average draw on the battery. Assuming a regulator efficiency of 50% and nominal battery capacity, the prototype should be able to attain a battery life of:

$$Optimal\ Battery\ Life = \frac{2.5Ah}{100.2uA} = 1039\ days\ of\ battery\ life$$

This assumes optimal software power reduction, negligible battery impedance and does not include any environmental effects. In real conditions, AA batteries are typically derated to as little as 25% of their nominal capacity. Even so, the above calculation shows that a one-month battery life across temperature is possible. The final product will also use a more efficient regulator which allows for a smaller battery.

Digital Peripherals

The functional prototype will make use of an Adafruit Feather shield which hosts a micro SD card slot as well as real time clock chip. The final product will implement these on the custom designed PCB but the Adafruit shield offers a convenient platform with which the team can obtain the same functionality in a larger, more expensive form factor. An 8 GB micro SD card will be onboard which will be able to store many months of data before being transferred to a PC. The real time clock will allow samples to be time stamped to do detailed analysis of collected data.

Analog Peripherals

An ATmega32u4 microcontroller has an onboard 10 bit ADC which will be used for the prototype. When paired with an external 0.1% 2.5V reference, resistor measurements with resolutions of less than 1 ohm are possible.

Software

The software needed for our ATmega32u4 microcontroller to properly timestamp and collect temperature and moisture values will be coded in the Arduino programming environment. Specifically, standard Arduino libraries such as Wire.h, SD.h, and SPI.h will be used to easily communicate with external hardware including the RTC and SD card. The desired program flow is shown below in Figure 2.

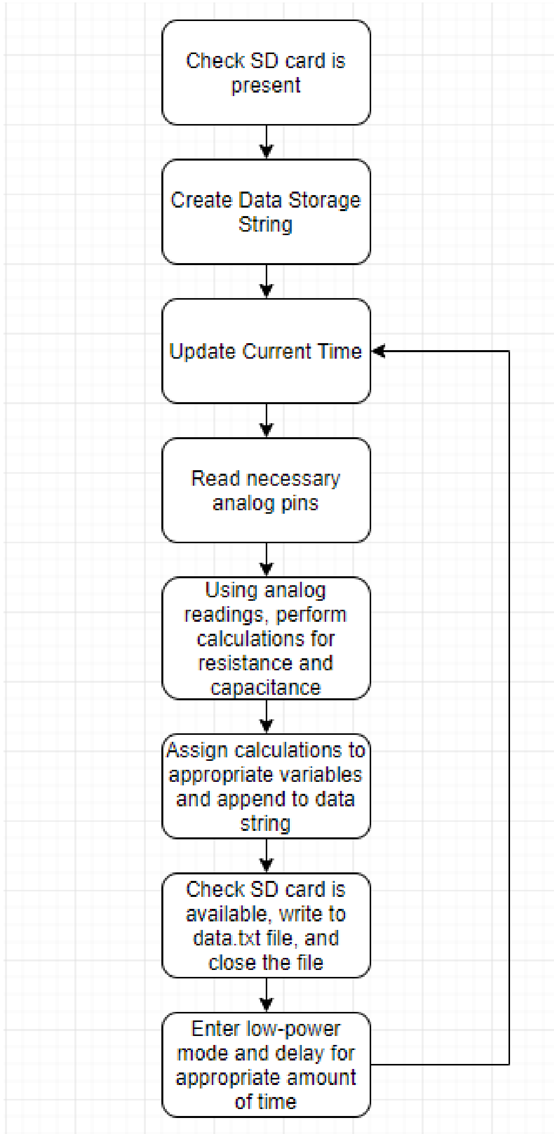


Figure 2: Software Flow Diagram

The flow diagram above follows a sequence of steps that can be used for successful data-logging to occur. First, all external hardware must be appropriately setup, e.g., the SD card must be in place for data to be written to it. A string for CSV data needs to be created and appended to for proper data-logging to take place. The RTC is then updated for the time, necessary analog pins read, calculations performed to get resistance and capacitance readings (resistance and capacitance can be converted to temperature and moisture content once sensors are characterized), data stored to the SD card, and the Arduino is put into low-power mode and delayed for an appropriate amount of time until the steps are ready to be repeated to log another data value. To properly extend battery life, the Arduino needs to be in low-power mode and triggered by an external signal to wake up and repeat the data-logging process. We plan to implement this functionality by using a dedicated square wave pin generator on our RTC as an interrupt signal. Assuming a sampling time of 15 minutes and using the basic $f = 1/T$ equation where f is the frequency in Hz and T is the period in seconds, our desired square wave would need to have a frequency of 0.00111 Hz or 1.11 mHz.

2.2 Design Analysis

As of writing, the design in Figure 1 has been demonstrated on a breadboard with the exception of the following requirements:

1. Battery life requirements, no power reduction implemented
2. All environmental requirements, implemented on breadboard
3. Capacitance (moisture) measurement

The prototype (shown in Figure 3) has been very successful in proving that the data collection and storage is feasible. Capacitance measurement is actively being worked on by the team and we are confident that we will be successful. In the event that ADC performance limits capacitance measurement the team has ordered an external 16 bit ADC which if needed can interface with the microcontroller and provide more resolution.

Each sensor channel will require some sort of conditioning to be measured by the ADC. Currently the prototype implements two temperature channels and two moisture channels. Temperature channels require a precision current source implemented with the reference and a high tolerance resistor. Moisture channels will require one high tolerance and high value resistor. The team is concerned that although precise measurements may be obtained in a lab environment, parasitic effects of sensor cables in moist soil will likely have detrimental effects on the moisture sensor. This will be examined closely once the functional prototype is fully assembled. In the event this is an issue, it's likely a small microcontroller will have to be placed near the sensor to eliminate parasitic effects. This microcontroller could also handle temperature measurements and digitally communicate the data to the microcontroller on the surface via a serial protocol such as I2C.

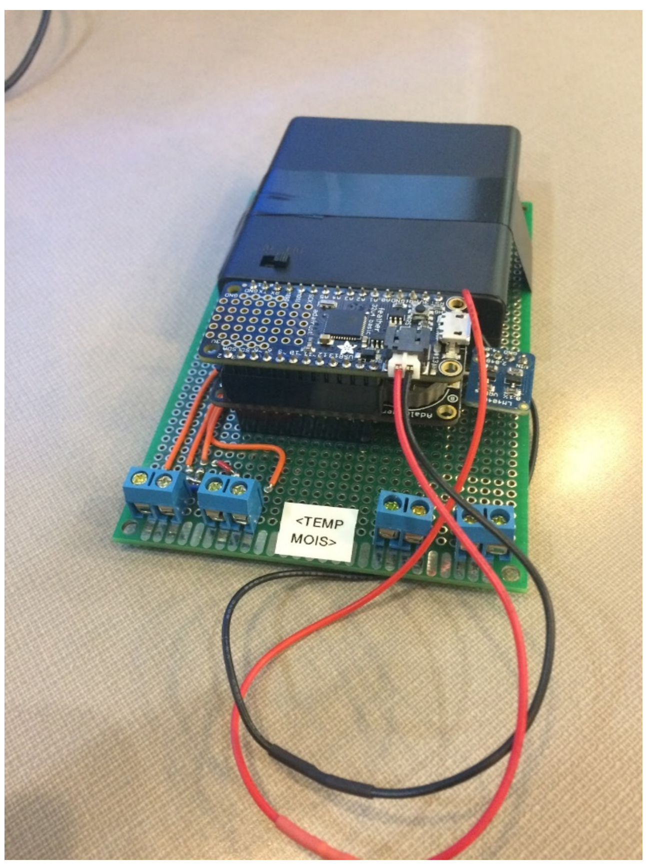


Figure 3: DAQ Protoytpe

Additional characterization of the analog inputs will be required upon prototype assembly to fully understand the sensor resolution and how it may be affected by the environment of the application. Full characterization and calibration over temperature will alert the team to any accuracy issues over temperature. Should any temperature issues arise, we will either increase resolution such that the requirements are still met at extremes of temperature requirements or will negotiate with customer to relax accuracy requirements at extreme temperatures.

Proposed design strengths:

- Simple and scalable to larger/smaller applications
- Low power consumption
- Highly accurate temperature measurement

Design Weaknesses and unknowns:

- Moisture measurement performance
- Enclosure and weatherproofing
- Ability to attain low standby current draw

3 Testing and Implementation

3.1 Interface Specifications

There are plenty of software/hardware interfacing that must be done in order to test our project effectively. The largest issue is going to be verifying that our measurement methods are correct. As will be explained in the next section, we will be using LCR meters and multimeters to verify that our testing methods are accurate.

Since we will be taking real-time measurements of resistance and capacitance, we need a way to visualize these readings. We have decided to store all of the data into a CSV file, then import it to a computer and view the data on excel. This will be the same interface used in our final project, but we will plot temperature and water content instead of resistance and capacitance.

We will be able to verify our project's temperature and resistance accuracy when we compare our plots completed on excel with the real temperature inside of the heat chamber. We will also compare our capacitance measurement accuracy a similar way.

3.2 Hardware and Software

Proper testing of our design prototype and solution will involve a variety of hardware and software. Most of the hardware is available through engineering labs in Coover Hall. We also have access to the necessary software as students at Iowa State through the workstations in the computer labs in Coover Hall. A humidity chamber could be used in the PCC lab in Town Engineering Building.

Hardware used for testing:

- Temperature Chamber
- Humidity Chamber
- Digital Multimeter

- Oscilloscope
- Power Supply
- Signal Generator
- LCR Meter

Software used for testing:

- Arduino IDE
- Excel
- National Instruments LabVIEW
- SPICE

Hardware Descriptions

Figure 4: Temperature Chamber

The temperature chamber provides a controlled temperature environment to test electronic circuits and devices of concern under different temperature conditions. There are openings to the exterior world that can be insulated with a special foam and still provide enough room for cables to be connected to a multimeter.

Figure 5: Humidity Chamber

(Source: https://www.associatedenvironmentalsystems.com/)

A humidity or environment chamber will enable us to be able to characterize our capacitive moisture sensors by controlling the relative humidity of the environment they are in and observing the changes in capacitance through external test equipment.

Figure 6: Digital Multimeter

The digital multimeter will be utilized to measure resistances, capacitances, voltages, and currents in our circuits. It will allow us to have a comparison for our data acquisition results and readings.

Figure 7: Digital Oscilloscope

An oscilloscope is a multifunction measurement tool that will allow us to monitor AC or DC signals and measure their frequency. This will allow us to confirm whether we are getting an expected output and can troubleshoot circuits in this way.

Figure 8: DC Power Supply

Even though our project will not be connected to a portable DC power supply, it will be utilized in the lab to conveniently provide a desired voltage where we need to in our circuits.

Figure 9: Signal Generator

A signal generator will be used to provide an excitation signal as an input to our circuits. We then will used the aforementioned oscilloscope to look at an output response and interpret data from the circuit.

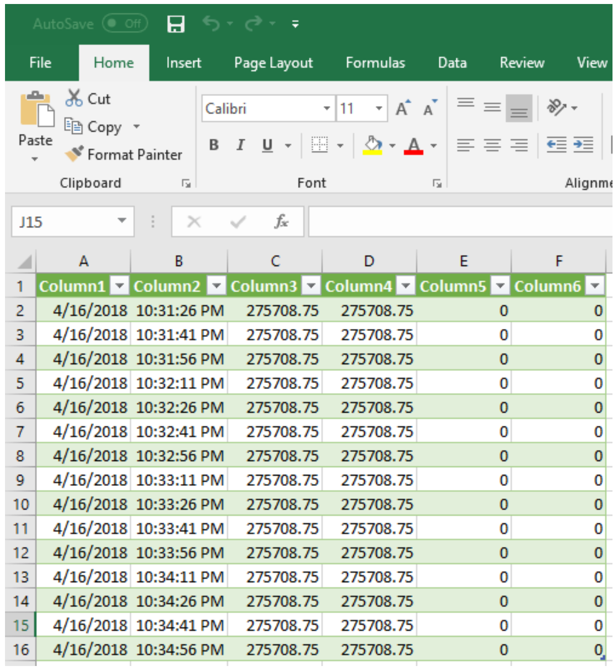
Figure 10: LCR Meter

An LCR meter will be used to get very accurate readings of inductors and capacitors when we are testing our data acquisition system with known inductance and capacitance values.

Software Descriptions

Figure 11: Arduino IDE

The Arduino IDE (Integrated Development Environment) is software that allows us to program and send a series of instructions to the Arduino board. Specifically, we are using an Arduino Feather Adalogger model that will be programmed via this programming environment.



	Column1	Column2	Column3	Column4	Column5	Column6
1	Column1	Column2	Column3	Column4	Column5	Column6
2	4/16/2018	10:31:26 PM	275708.75	275708.75	0	0
3	4/16/2018	10:31:41 PM	275708.75	275708.75	0	0
4	4/16/2018	10:31:56 PM	275708.75	275708.75	0	0
5	4/16/2018	10:32:11 PM	275708.75	275708.75	0	0
6	4/16/2018	10:32:26 PM	275708.75	275708.75	0	0
7	4/16/2018	10:32:41 PM	275708.75	275708.75	0	0
8	4/16/2018	10:32:56 PM	275708.75	275708.75	0	0
9	4/16/2018	10:33:11 PM	275708.75	275708.75	0	0
10	4/16/2018	10:33:26 PM	275708.75	275708.75	0	0
11	4/16/2018	10:33:41 PM	275708.75	275708.75	0	0
12	4/16/2018	10:33:56 PM	275708.75	275708.75	0	0
13	4/16/2018	10:34:11 PM	275708.75	275708.75	0	0
14	4/16/2018	10:34:26 PM	275708.75	275708.75	0	0
15	4/16/2018	10:34:41 PM	275708.75	275708.75	0	0
16	4/16/2018	10:34:56 PM	275708.75	275708.75	0	0

Figure 12: Excel Formatted Data

Excel is a very familiar datasheet program on the Windows operating system. We will be utilizing Excel to import CSV data stored on an SD card as shown in Figure 11 above.

LabVIEW/SignalExpress is a software package that can be utilized to automate and take a series of electronics test bench measurements very quickly. The data can then be transferred to a datasheet software program like Excel assuming the appropriate drivers have been installed to communicate with the electronic test equipment of concern.

Eagle is a software package that contains tools that can be used to simulate a proposed circuit that has been drawn in a schematic view. In addition, a printed circuit board and its traces can be routed in another window within Eagle.

3.3 Functional and Non-functional Testing

TEST TYPE #1

These tests verify the functionality of the product for all non-measurement requirements including data storage, battery life, and sample period.

Test	Required Result	Test Status
1A: Storage Requirement	1 month of data	Not Attempted
1B: Battery Life	1 month battery life in temperate conditions	Not Attempted
1C: Sample Period	15 minute sample period	Not Attempted
1D: Operating Environment	-10 to 100 degrees Fahrenheit	Temperature passed. Moisture not attempted

Test 1A Procedure: Storage

1. In a lab environment, attach two temperature and two moisture sensors to the UUT (unit under test)
   Note: test does not depend on sensor data
2. Configure the UUT for a sample period of 1 second by editing the CONFIG.txt file on the SD card
3. Configure the UUT for a sample length of 3000 samples.
   Note: 3000 derived from 15 minute sample period over 30 days with ~5% margin
4. Start datalogger
5. After completion, open SD card contents using PC and verify 3000 individual measurements exist for each channel with associated time recorded

Success Criteria: 3000 data points exist for each channel with associated timestamp

Failure Criteria: Any other outcome results in test failure

Test 1B Procedure: Battery Life

1. Ensure battery is fully charged
2. Configure UUT for 15-minute sample period
3. Configure UUT for 3000 samples
4. Attach two temperature and two moisture sensors
   Note: test does not depend on recorded sensor data
5. Place datalogger in a secure location indoors, record start time, and start datalogger
6. One month later, recover datalogger and analyze data on SD card

Success Criteria: Minimum of 2,880 individual data points exist for each channel with timestamp

Failure Criteria: Fewer than 2,880 individual data points on any channel or timestamps fail at any point in the test

Test 1C Procedure: Sample Period

1. In a lab environment, configure UUT for 15-minute sample period
2. Configure UUT for 4 samples
3. Attach two temperature and two moisture sensors
   Note: test does not depend on recorded sensor data
4. Start datalogger
5. One hour later, retrieve and open SD card on PC

Success Criteria: Four samples exist for each channel and samples occur every 15 minutes with 5 second margin of error

Failure Criteria: Any number of samples less than or greater than four or recorded time stamp is not in 15 minute increments

Test 1D Procedure: Operating Environment

1. *In a lab environment, configure UUT for 1 minute sample period*
2. *Configure UUT for 120 samples*
3. *Attach the following reference components to sensor terminals:*
    a. *270 ohm resistor to Temp channel #1*
    b. *200 ohm resistor to Temp channel #2*
    c. *1pF capacitor to Moisture channel #1*
    d. *2.2pF capacitor to Moisture channel #2*
4. *Place UUT into temperature test chamber*
   Note: the objective of this test is NOT to measure how the reference components change with temperature. Place reference resistors outside test chamber, place reference capacitors outside test chamber using cable with known parasitic capacitance.
5. *Start UUT and temperature chamber with minimum 1 cycle from –10 to 100 degrees Fahrenheit over two hours. Record the temperature with time stamps to be able to compare to recorded data later*
6. *After test completes, retrieve SD card and open recorded data on PC*

Success Criteria: Less than 1% change in recorded sensor value over the duration of the test

Failure Criteria: Any sensor channel changes more than 1% over the duration of the test

TEST TYPE #2

These tests verify the accuracy and sensitivity of the DAQ satisfying or exceeding customer requirements.

Test	Required Result	Test Status
2A: Resistance Accuracy	Within 0.5 ohms	Failed
2B: Resistance Sensitivity	< 0.5 ohm resolution	Not Attempted
2C: Capacitance Accuracy	Within 10fF	Not Attempted
2D: Capacitance Sensitivity	< 10fF resolution	Not Attempted
2E: Temperature Independence	<= 1 bit change over temp	Passed for resistance

Test 2A Procedure: Resistance Accuracy

1. *Obtain a multimeter with 10mOhm accuracy +/- 1%*
   Note: HP34401A is commonly available and meets these requirements
2. *Obtain resistance standard capable of sweeping from 10 to 1k ohms*
3. Configure DAQ to continually stream measurements with sample period 1 second to PC serial monitor
4. Set resistance standard to desired value within range 100 – 400 ohms
5. Measure resistance standard using bench multimeter and record results

6. Disconnect bench multimeter from resistance standard, connect DAQ, and record reported value
7. Repeat steps 4-7 for all values in range 100-400 ohms

Success Criteria: DAQ reported value is within 0.5 ohms across range 100 -400 ohms

Failure Criteria: DAQ reported value is more than 0.5 ohms from meter reported value

Test 2B Procedure: Resistance Sensitivity

1. Configure DAQ to continually stream measurements with a sample period of 1 second to PC serial monitor

   Note: Both the output of the ADC (0 – 1023) and calculated resistance can be shown

2. Connect decade box to terminals of a resistance channel
3. Increment decade box by 1 Ohm to cause a 1-bit change in the ADC reading
4. Observe the calculated resistance value before and after decade box was incremented
5. Calculate the difference before and after decade box was incremented by 1 Ohm

Success Criteria: Calculated difference in step 5 is < 0.5 Ohm

Failure Criteria: Calculated difference in step 5 is > 0.5 Ohm

Test 2C Procedure: Capacitance Accuracy

1. Obtain LCR meter with minimum 10fF accuracy
2. Obtain capacitor standard or several capacitors in the range of several hundred fF to several pF
3. Configure DAQ to continuously stream measurements with sample period of 1s to PC serial monitor
4. Place reference capacitor on DAQ moisture sensor terminals
5. Record DAQ reported measurement
6. Disconnect capacitor from DAQ
7. Measure capacitor with LCR meter with test frequency as near 45kHz as possible and record results
8. Repeat steps 4-8 for each capacitor in the range of 100fF to 4pF

   Success Criteria: DAQ and LCR meter readings differ no more than 40pF for any test point

   Failure Criteria: Any two readings are more than 40pF apart between LCR meter and DAQ

Test 2D Procedure: Capacitance Sensitivity

1. Configure DAQ to continuously stream measurements with sample period of 1s to PC serial monitor

2. Place reference 1pF capacitor on moisture sensor terminals
3. Record DAQ reading of capacitor
4. Add 10fF capacitors until DAQ output changes by 1 bit
   Note: may need to place larger capacitors in series to get such small value
5. Measure and record LCR reading of same arrangement of capacitors while disconnected from DAQ

Success Criteria: No more than 40fF was added to DAQ terminals before LSB changed

Failure Criteria: More than 40fF added to DAQ terminals before LSB changed

Test 2E Procedure: Temperature Independence

1. Place DAQ inside temperature chamber capable of minimum 0 to 100F
2. Connect reference resistor and capacitor to DAQ input terminals. Resistor should be approximately 250 ohms, capacitor should be approximately 1pF
   Note: reference resistor and capacitor should be outside temperature chamber
3. Configure DAQ to continually output measurements to PC serial monitor every 1s
4. Set temperature chamber to sweep over 0 to 100 degrees Fahrenheit.
5. Monitor serial monitor and record any change in measured value and the temperature at which the change occurred

Success Criteria: Output for both temperature and moisture changed by no more than 1 LSB over the entire temperature range

Failure Criteria: Output for either reference component changed by more than 1 LSB at any point in the test

TEST TYPE #3

These tests verify the DAQ system meets or exceeds requirements for its intended use in the field.

Test	Required Result	Test Status
3A: Lab Test	1-month data collection	Not Attempted
3B: Environment Test	1-month data collection	Not Attempted

Test 3A Procedure: Lab Test

1. Configure DAQ for 15-minute sample period
2. Attach one sensor each of moisture and temperature
3. Attach one reference resistor and one reference capacitor to the remaining channels
4. Charge DAQ battery to full
5. Place DAQ in secure indoor location and leave for one month
6. After 1 month, retrieve and open the contents of the SD on a PC

Success criteria: Data exists for entire 1-month period, reference values measure consistently throughout test

Failure criteria: Battery does not last 1 month, measurement of reference values differ by more than 1 LSB over test, or sensor measurements change drastically

Test 3B Procedure: Environmental Test

1. Configure DAQ for 15-minute sample period
2. Attach two sensors each of moisture and temperature
3. Charge DAQ battery to full
4. Place DAQ in secure outdoor location with sensors buried
5. Start DAQ
6. After 1 month, retrieve and open the contents of the SD on a PC

Success criteria: Data exists for entire 1-month period if temperature was moderate, two weeks if temperature was consistently under 30 degrees
Failure criteria: Battery does not last for required duration or sensor channels measure significantly different

3.4 Results and Challenges

The following are some preliminary data and tests that were performed to test the functionality requirements of our DAQ system along with interpretations of collected data and any challenges encountered.

TEST #2A: Resistance Accuracy Measurement Range

Test #2A was performed in Coover using an HP 34401A digital multimeter. We measured resistance values between a range of 10 ohms and 1000 ohms, with an emphasis on values between 100 and 400 ohms, as that is what our sensors will most likely behave as. We measured an average difference of 0.41 ohms between the DMM readings and our DAQs readings. A plot showing this difference between 100 and 400 ohms is shown below in Figure 13.

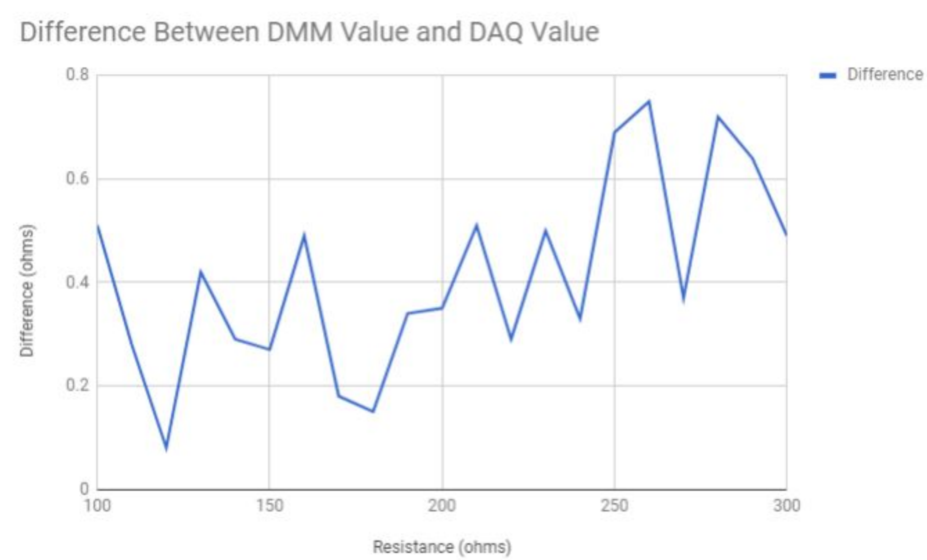


Figure 13: Difference in Resistance Between DMM and DAQ

TEST #2E: Temperature Independence

Test #2E was performed in Coover with a TestEquity Half Cube temperature chamber. We started by putting our perfboard prototype shown in Figure 2 inside the temperature chamber and connecting it to a laptop to read its measurements of a constant nominal resistance of 250 Ohms connected to a channel. We first started at 23 degrees Celsius and then swept the ambient temperature at various setpoints going from top-to-bottom in Table 2 below. The overall process involved first cooling the DAQ to −17.7 degrees Celsius, heating it to 49.5 degrees Celsius, and cooling it all the way down to −42 degrees Celsius. The mode of the measured data was recorded once the setpoint temperature "settled", i.e., we waited a couple of minutes after the setpoint was reached to allow the prototype to actual reach the ambient temperature in the chamber. As one can see, the reading never fluctuated by more than one bit which corresponds to 0.97 Ohms, so we deemed this test a success.

Temperature (Celsius)	Temperature Farenheit	Measured Resistance (DAQ in Ohms)
23	73.4	249.25
15	59	249.25
0	32	249.25
-17.7	0.14	250.22
35	95	249.25
37.7	99.86	249.25
49.5	121.1	249.25
-42	-43.6	250.22

Table 2: Temperature Independence Test Data

TEST #1C: 15 Minute Sampling Period

Due to time considerations, a variation of test #1C was performed with a sampling period of 15 seconds as opposed to 15 minutes. We know this test will need to be performed again, however, we see that our software and hardware is capable of formatting data in a CSV format to a .txt file and that data can be formatted in Excel. No capacitors or resistors were attached to the DAQ system since not all functionality was implemented yet on the prototype. The CSV data shown below in Figure 14 is "dummy data" that corresponds to the Excel data shown in Figure 12 on page 14.

```
4/16/2018,22:31:26,275708.75,275708.75,0.00,0.00
4/16/2018,22:31:41,275708.75,275708.75,0.00,0.00
4/16/2018,22:31:56,275708.75,275708.75,0.00,0.00
4/16/2018,22:32:11,275708.75,275708.75,0.00,0.00
4/16/2018,22:32:26,275708.75,275708.75,0.00,0.00
4/16/2018,22:32:41,275708.75,275708.75,0.00,0.00
4/16/2018,22:32:56,275708.75,275708.75,0.00,0.00
4/16/2018,22:33:11,275708.75,275708.75,0.00,0.00
4/16/2018,22:33:26,275708.75,275708.75,0.00,0.00
4/16/2018,22:33:41,275708.75,275708.75,0.00,0.00
4/16/2018,22:33:56,275708.75,275708.75,0.00,0.00
4/16/2018,22:34:11,275708.75,275708.75,0.00,0.00
4/16/2018,22:34:26,275708.75,275708.75,0.00,0.00
4/16/2018,22:34:41,275708.75,275708.75,0.00,0.00
4/16/2018,22:34:56,275708.75,275708.75,0.00,0.00
```

Figure 14: CSV Data in .txt File

One challenge was figuring out how the RTClib.h library functions worked and understanding the functions' return types, specifically functions that operated on class objects. Once that aspect was understood by looking at example code provided in the RTClib.h library, we used SD data-logger example code from the SD.lib library and modified it for our purposes to show the date, time, resistor channel 1, resistor channel 2, capacitance channel 1, and capacitance channel 2 values on each line of the data file.

4 Closing Material

4.1 Conclusion

Our team has a working prototype for resistance/temperature measurements, and we have commenced modeling and simulating a capacitance measurement circuit. Furthermore, we have confirmed logistics in getting the MEMS sensors refabricated at the beginning of next semester, Fall 2018.

We have split up into three primary task groups – a group whose purpose is to continue developing our circuit measurement prototypes, a group that is working with professors to ensure we have functional sensors, and a group that is working on software development.

The end product will be a complete system that can measure resistance and capacitance via a MEMS sensor and store that information in 15-minute intervals for an entire month. The MEMS sensor was originally provided to us by the CCEE department and was developed by the ECpE department. These sensors (used on a previous project) were found to be not functional at the time we received them due to excessive scratching. It was decided that we should attempt to recreate the MEMS sensors and interface them with our DAQ system. We asked for ECpE Associate Professor Dr. Garry Tuttle to help guide and provide expertise so that we could recreate the MEMS sensors at the MRC. He kindly agreed to guide through the various process steps involved in fabricating MEMS sensors on a wafer.

The best plan of action will be to continue testing of our most recent prototype and printed circuit board, which combines both the capacitance and resistance measurement circuitry and channels. While one part of the team is working on testing and better improving upon our most recent prototype, the other team members will work collectively to continue software development. In addition, MEMS sensor fabrication and characterization tests will need to be performed to be able to convert capacitance and resistance readings into moisture and temperature values respectively.

4.2 References

1. Ceylan, Halil; Yavas, Seval; Dong, Liang; Jiao, Yueyi; Yang, Shuo; Kim, Sunghwan; Gopalakrishnan, Kasthurirangan; and Taylor, Peter, "Development of a Wireless MEMS Multifunction Sensor System and Field Demonstration of Embedded Sensors for Monitoring Concrete Pavements, Volume I" (2016). InTrans Project Reports. 219. http://lib.dr.iastate.edu/intrans_reports/219