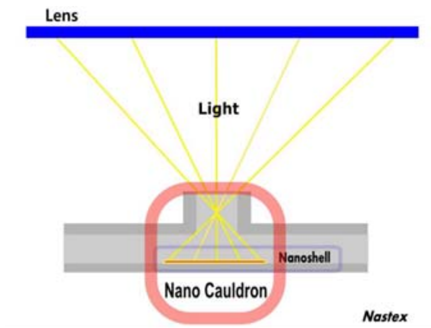
# AquaSol/AquaTrol Solar Water Heating System

**Tim Erickson and Dylan Pfeifer**
**EE385J Final Project**
**Submitted to Dr. Valvano**
**May 10, 2009**

**Overview**

We have invented, engineered and constructed the AquaSol/AquaTrol (ASAT) system, a functional, fully automated, solar water heating system, which can be installed on a residential or commercial rooftops to supplement traditional gas/electric water heating systems. The system employs a large Fresnel Lens (0.7 sq. meters) to focus light into an innovative heating chamber, referred to as the Nano-cauldron. The inside of the Nano-cauldron is coated with unique selective absorbing agent (Solkote + Gold Nanoshells), which boosts heating efficiency. The unique geometry of the cauldron allows much of the light to enter through a glass window on the heating pipes at steep angles (due to lens focusing), and then become "trapped" inside the chamber in order to efficiently heat water.

The ASAT control system features three-channel analog acquisition (water temperature, ambient temperature, and solar intensity), automated control, remote monitoring, and a lens positioning system (AquaTrol). The remote data logging capabilities of the control system will facilitate data collection over long time periods, as we seek to determine the system's long run efficiency and expected annual energy output.




**Sections**

# I. Functional Requirement Specification

Purpose

The purpose of this specification is to collect functional requirements of the system. The identification and enumeration of requirements will allow a method by which to evaluate the effectiveness of the implementation and bound system development time and feature sets. The functional specification fills a role similar to a Marketing Requirements Specification or Product Requirements Specification in a formal engineering flow.

Enumerated Requirements

I. AquaSol Collector and Heating Unit

a. The device should concentrate sunlight through lensing, spread out into the heating chamber, and heat water to 55° C, in a batch cycle heating process.
b. The device should test the effectiveness of a novel Nanoshell-enhanced collector which may be more efficient than a standard black paint collector.
c. The lens should be aligned normal to the sun for maximum solar energy harvesting.
d. The system should heat water to levels comparable to a conventional home water heater.

II. AquaTrol DAQ/Control System

a. Data Measurement Requirements
   i. In-pipe water temperature
      Senses the temperature of water in the heating chamber.
   ii. Pipe housing surface temperature
      Measures the external surface temperature of the water heating element
   iii. Ambient temperature
      Atmospheric temperature around the device
   iv. Ambient sunlight intensity
      Solar intensity normal to lens surface
   v. Date and Time
      Device will know and store wall-clock time and date
   vi. Millisecond time
      Device will express distance between control events to Millisecond resolution (240 Hz).

b. Control Requirements
   i. Manual Control: The system shall present manual, on-site, panel based control through an HMI interface
      a. An operator shall be able to manually open and close

water control valves through panel switches

b. An operator shall be able to manually position the lens through motor control switches
c. An operator shall be able to manually enable or disable automation control through panel switches.

ii. Automatic Control: The system shall be able to automatically control lens positioning and valve opening to deliver fully autonomous operation

a. The system shall open valves for water throughput at configurable maximum temperature setpoint
b. The system shall position the lens in response to time of day and position of the sun.

c. I/O Requirements

i. On-site panel I/O

a. The system shall have a panel readout for monitoring system condition at the device
b. The system shall have power and control indicator LED's

ii. Remote I/O

a. The system shall support remote I/O through a command line interface and network TCP/IP interface
b. The system shall support remotely managed data logging through TCP/IP

iii. Data Logging

a. The system shall support data logging and event logging for study of Nanoshell collector efficiency

iv. RS-232 CLI

a. The system shall have a local RS-232 command line interface exporting all control and monitoring options independent of the panel I/O

d. Power Requirements

i. The system shall operate independent of a land-line power supply
ii. The system shall exercise electrical isolation between the signal level DAQ and control circuits and the power-level motor and valve drive circuits

e. Safety

i. The system shall employ safety mechanisms on all power circuits

## II. Implementation Specification

The AquaTrol system uses the following modular solution to implement design requirements:

Sensors and Sensor Interface Module (SIM)

> The SSIM employs three conventional sensors, a RTD, a thermistor, and a light-sensitive photocell to measure temperatures and sunlight intensity. The SIM implements signal preamplification and active analog low-pass filtering and exports a direct signal interface to the Control Module ADC. The SIM shall be determinable in figures of merit such as frequency response, time contant, SNR, accuracy, range, resolution, and repeatablility.

Automation Control Module (ACM)

> The CM employs a 9S12 microcontroller, AGM LCD display, and I/O switches. The 9S12 (uC) will perform 60 HZ digital noise rejection, sensor calibration, control logic, sampling, I/O handling, and all other software functions required for the system. The DAQ shall be classifiable in real-time figures of merit, namely, maximum sampling jitter. The DAQ shall sample at a multiple of 60 HZ on 3 channels of the ADC unit, 960 Hz.

Valve Control Module (VCM)

> The VCM provides a panel switch control for the device water valves, and provides a logic-level optically isolated control input port by which the unit can be driven from the ACM.

Motor Control Module (MCM)

> The MCM provides a panel switch control for the lens positioning motor and a logic-level optically isolated control input port by which the unit can be driven from the ACM

Network Interface Module (NIM)

> The NIM provides an RS-232 to Ethernet 10BaseT bridge and embedded web-server by which the system can be monitored and controlled by TCP/IP (hence the Internet). The NIM supports DHCP, StaticIP, email notification, and SNMP notification for device data-logging. The NIM supports NTP or Time-Of-Day protocols for setting the device time for motor positioning.

Power Regulation Module (PCM)

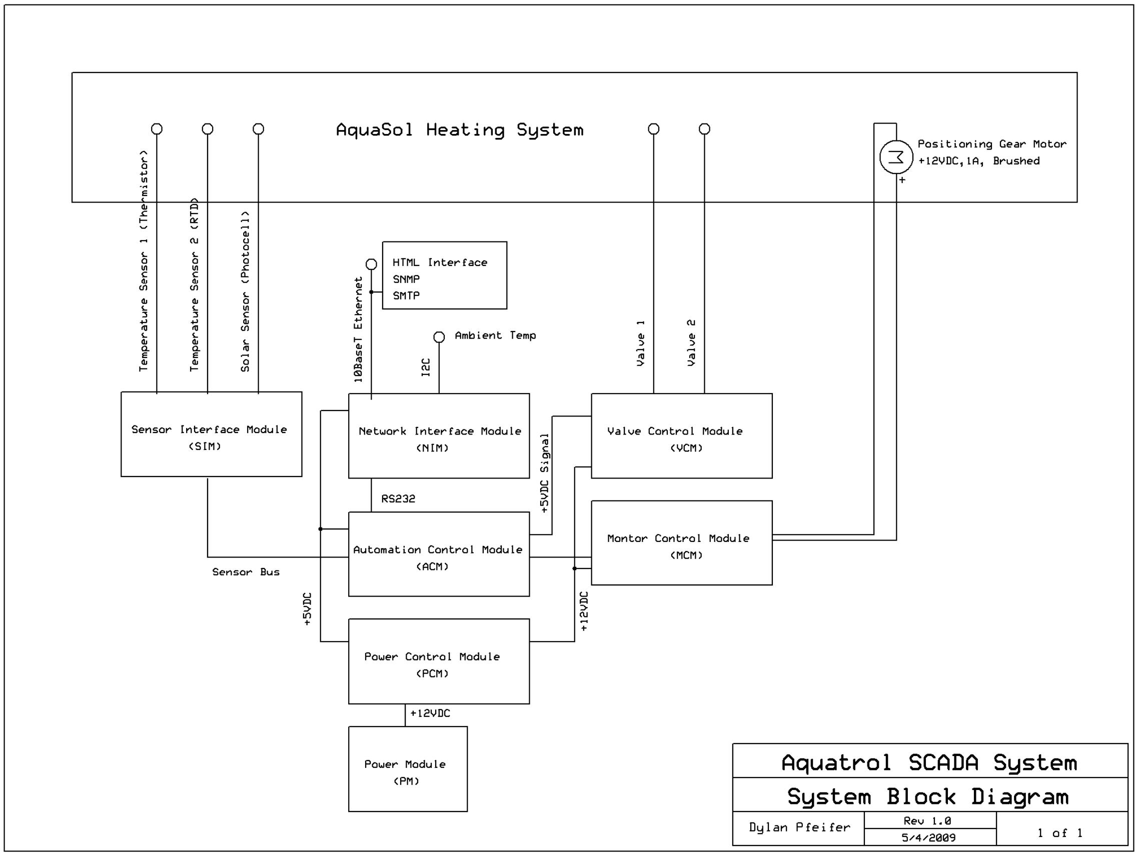
> The PCM inputs unregulated +12VDC battery power and exports regulated +7VDC and +5VDC power for the digital NIM and ACM. The VCM, MCM, and

PCM are all powered by unregulated +12VDC battery as the power drivers. The SSIM is powered by the regulated +5VDC from the ACM.

Power Module (PM)

The PM is a fused, switched +12VDC rechargeable battery supply.

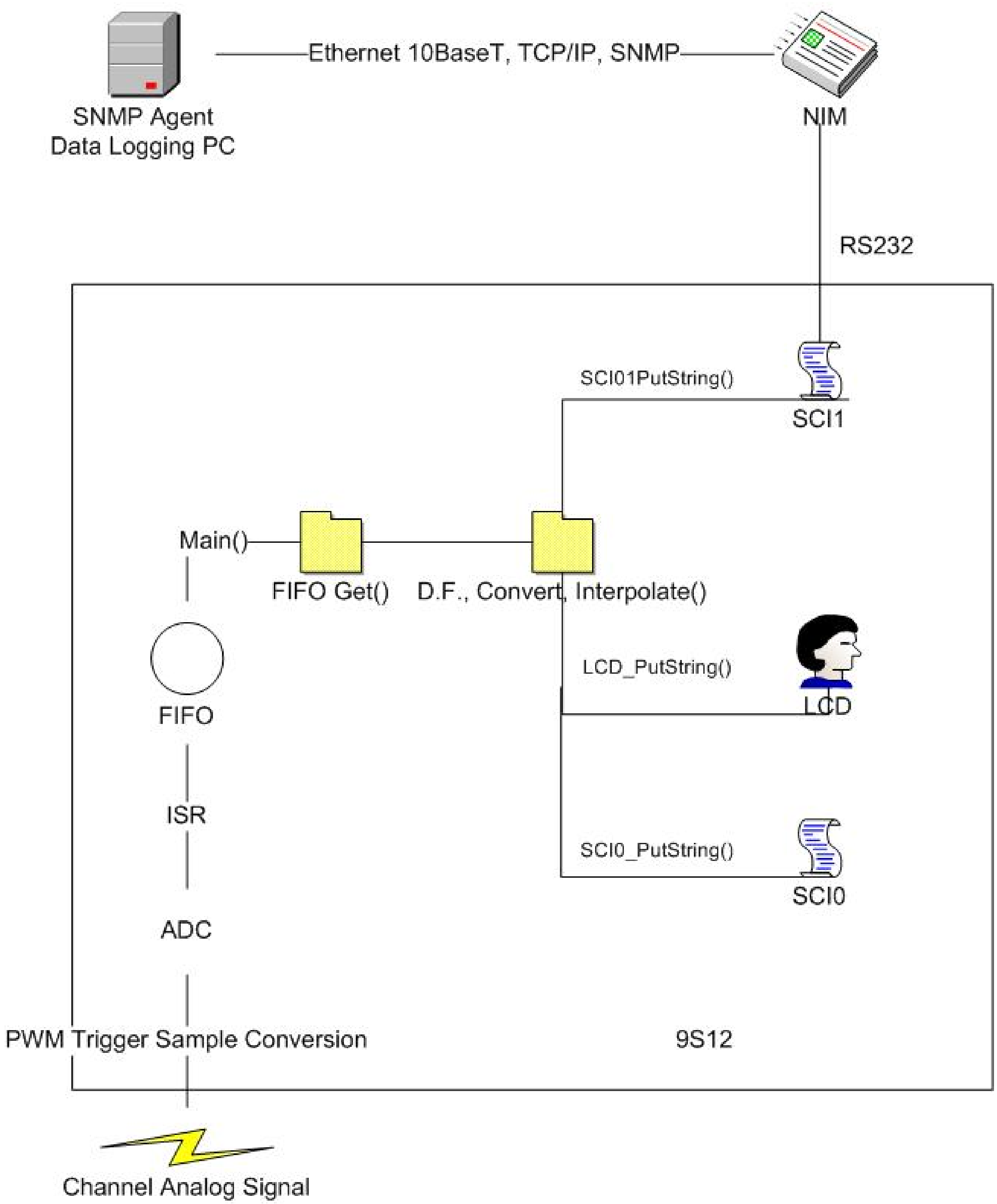
**Figure 1. System Block Diagram**




The system implementation interconnect is illustrated in Figure 1. Each module corresponds to an individual board and package (literal black box) with connector I/O.

**III. Software Dataflow**

DAS Dataflow Diagram



The datastream begins with an ADC conversion scheduled by a PWM hardware trigger. The conversion generates an interrupt which streams the data to a global FIFO through the ADC isr() handler. The handler takes the data from the ADCDR0-2 registers and puts 3 values into the fifo (for 3 channels).

In the foreground, main() takes the data from the FIFO, calls DigitalFilter(), then converts the sample to a voltage. The voltage is translated into a data value (temperature) by interpolation over a voltage-to-temperature lookup table. The converted value is then streamed to the LCD and SCI0 for immediate consumption, and to SCI1 for network consumption. SCI1 is connected to the NIM RS232 port. The PIC uC on the NIM takes the RS232 data and constructs SNMP UDP packets for broadcasting the data to SNMP agents. The agents then distribute the data to whatever applications are desired. This eliminates the need for on-device high storeage, and provides network-remote monitoring and control.

## IV. Protocols

Aquatrol uses a custom command line protocol for device control, and RS232 ascii protocol for communication with the NIM.

### CLI commands

Auto On/Off
    Turns on our off automatic control. Higher priority than panel automation on/off switch.
    Syntax: Auto 1/0
    Returns <Automation Off><Automation On> confirm
Dump On/Off
    Syntax: Dump 1/0
    Turns on or off RS-232 1-second data dumping
Valve1 On/Off
    Syntax: V1 1/0
    Turns on or off Valve 1
Valve2 On/Off
    Syntax: V2 1/0
    Turns on or off Valve2
Motor F/R
    Syntax: M F/R
    Motor forward/reverse

Motor Duration

Syntax: MT {seconds}

Duration motor is enabled during a forward/reverse command

Temp Limit

Syntax: T {limit}

Limit temperature for water to reach before it is replaced with cooler water after valve opening.

Hour Min Second

Syntax: H {0-23} M {0-60} S {0-60}

Set system wall-clock time

Sensor Read

Syntax: S1, S2 or S3

Read sensor value for sensor 1, 2 or 3

**EventLink protocol**

The following message protocol allows the 9S12 to send data to the NIM over RS232 for bridging to TCP/IP networks

<Timestamp:Event:Value>

The timestamp is the 9S12 clock time, for measuring distance between real-time events when recorded by the 9S12. The Event is the event source, a string ("Temp1, Temp2, PhotoC, Valve 1, Valve 2, MotorF, MotorR). For binary signals (on,off) the Value field contains 1/0, for continuous data the Value field is a string representation of the value. The NIM takes the values as strings and converts them to SNMP packets for a network high-storage data logger.

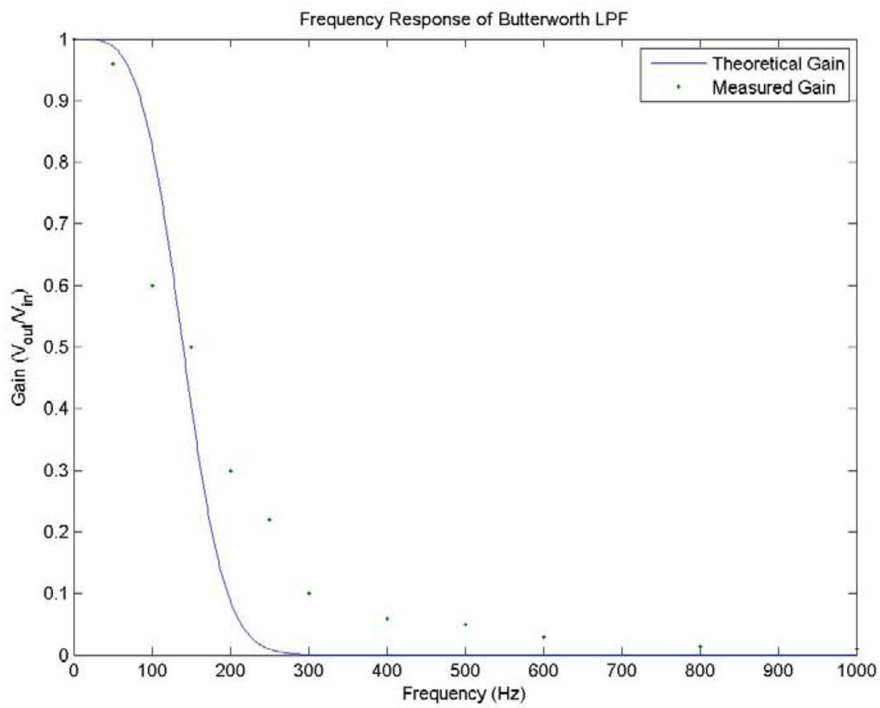
For network-to-device control commands, EventLink is also parsed by the 9S12 and translated to actuator events. For example <CMD:Valve1:0> will turn off valve 1, <CMD:Valve1:1> will turn it on. These are originated with SNMP write commands to the NIM, translated to EventLink commands over RS-232, translated to port write commands in the 9S12.

## V. DAS Metrics

Calibration, Interfacing, and Filtering

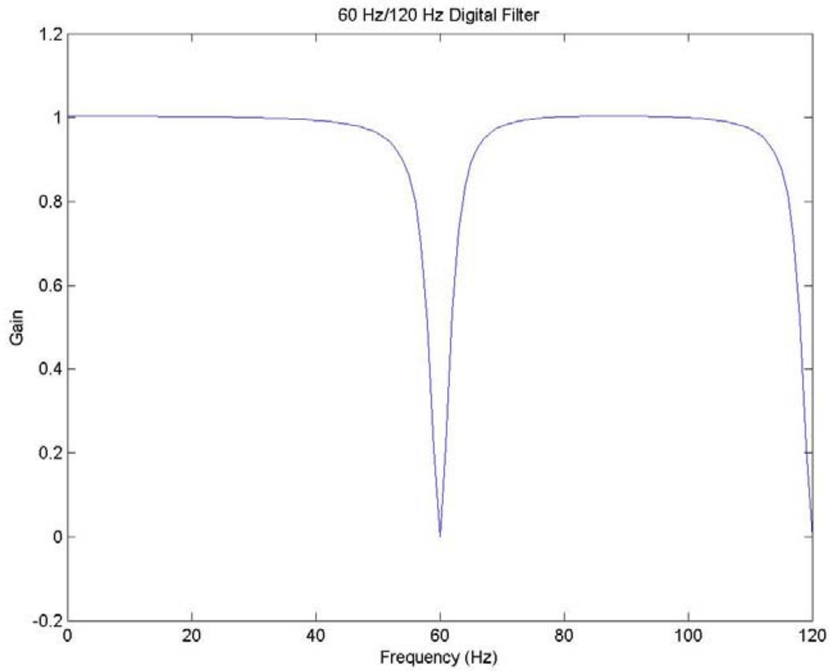
Low Pass Filter Testing

We tested our two-pole Butterworth 225 Hz Low Pass Filter and compared the measured gain to the theoretical gain. Our gain does not fall off as rapidly as predicted. Whereas the theoretical curve shows total rejection at 300 Hz, our filter passes about 10% at 300 Hz, and does not provide total rejection until roughly 900 Hz. Although this filter meets our present needs, it may be wise in the future to use a 3-pole filter if sharper rejection is needed.




Digital Filter

We designed a digital filter to reject 60Hz/120 Hz noise. The filter's gain was simulated in Matlab (next page). A median filter was also implemented to reject speckle noise.




Noise Analysis

Noise sources must be identified and quantified in the Sensor Interface Circuit. The sources of noise contribution are 60 Hz EMF noise, DC motor and valve inductor EMI, 1/f connector noise, resistor thermal noise, and opamp noise. Noise is compared by theoretical and empirical measure.

Theoretical measure:

1/f, 60 Hz noise, and motor noise do not have a theoretical model for this report. Interface noise from bridge resistors and opamps however can be included. The system bandwidth is 500 Hz. Using the highest-resistance resistor bridge in the SIM (1M ohm x2 and 4.7k), or the thermistor sensor interface, we calculate resistive thermal noise in the bridge:

V(rms, bridge) = sqrt[2*(4kt1M*500)^2 + (4kt4.7k*500)^2)], s.t.
T = 300 degrees K, k = 1.67 exp -23
= 4.47 uV

This noise is amplified through the INA122 instrumentation amp with a gain of 25. The INA122 datasheet input noise is 60nV/sqrt(HZ). At 500hz BW, the input noise is 1.31 uV. Assuming the INA122 input noise and bridge noise are uncorrelated, we can add them as independent noise sources through the sum of squares law.

Total INA122 input noise = sqrt (R bridge noise^2 + INA122 input noise^2)
= sqrt [ 4.47uV^2 + 1.31 uV^2)

= 4.65 uV

The output noise of the INA122 through a gain of 25 is then 116.45 uV.

Next, the signal is filtered for low-pass through an OPA2350 with unity gain. The input noise of the OPA2350 from datasheet is 5nV/sqrt(HZ). At system bandwidth of 500, the input noise is 111.8 nV.

Assuming the output referred noise of the INA122 and the input noise of the OPA2350 are uncorrelated, we find the total input noise of the OPA2350 by sum of squares.

Total OPA2350 input noise = sqrt[ 116.45 uV ^2 + 111.8 nV ^2]
= 116.45 uV

Since the OPA2350 is operated at unity gain, the total thermal noise from opamp and R-bridges is **116.45 uV/ADC channel.**

Empirical Noise Measurement

Empirical measurement will include 1/f, EMI, and other noise sources. Since the thermal noise is calculated 116.45 uV RMS at the ADC, and the ADC maximum resolution is 5 mV, significant variance in the ADC output per stable input should result from sources other than thermal.

We use the software method to calculate total RMS noise according to lab 1 procedures. We find a y(avg) measurement over N measures. We then calculate sqrt[1/n sum{y(i) – y(avg)] as the RMS noise.

We take n = 217 measurements using the SCI port of the 9S12 with the sensor input at a fixed resistance (4.7k) for the thermistor bridge. Since the thermistor interface circuit is candidate for the greatest noise contribution (by virtue of larger resistors, otherwise identical in other aspects to the other signal interfaces), we use it to represent the RMS noise/ADC channel of the instrument.

Collecting values and using Excel, we calculate the noise RMS as follows:

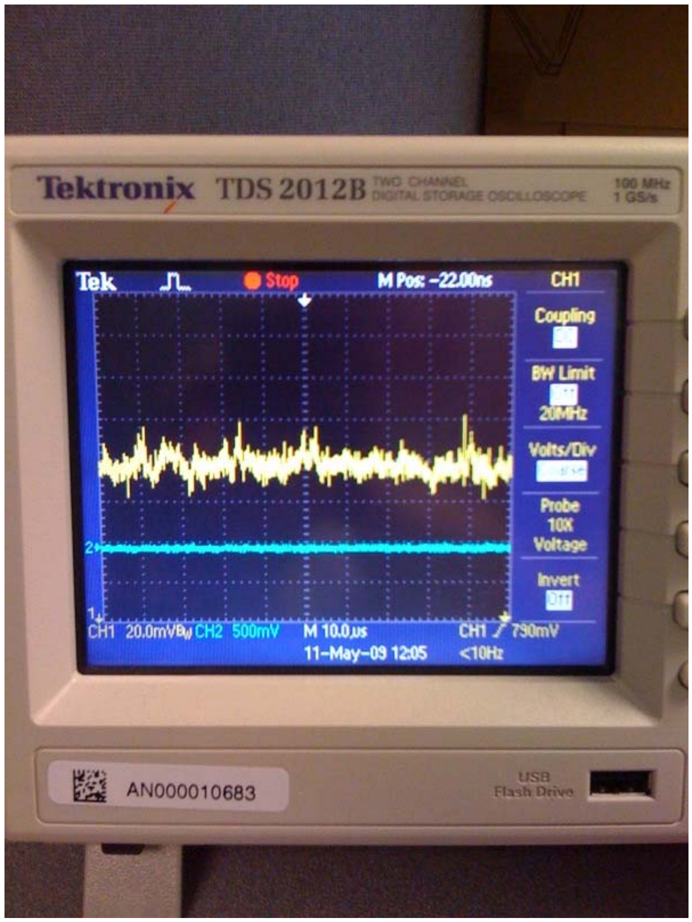
The thermistor is replaced with a 20k resistor, creating a fixed input .96V input to the ADC.

217 values are collected. The raw ADC conversion is 202 (0.96V). The average ADC sample is 202.1843. The RMS (sqrt(1/217 sum( y(i) – y(avg))) is **5.55 mV.**

This is much larger than the theoretical RMS noise of the resistors and opamps. We attribute the extra empirical noise to quantization error in the ADC, which has 4.88 mV resolution, and 60Hz noise.

Using an oscilloscope FFT feature, we inspect for any other noise sources such as 1/f:

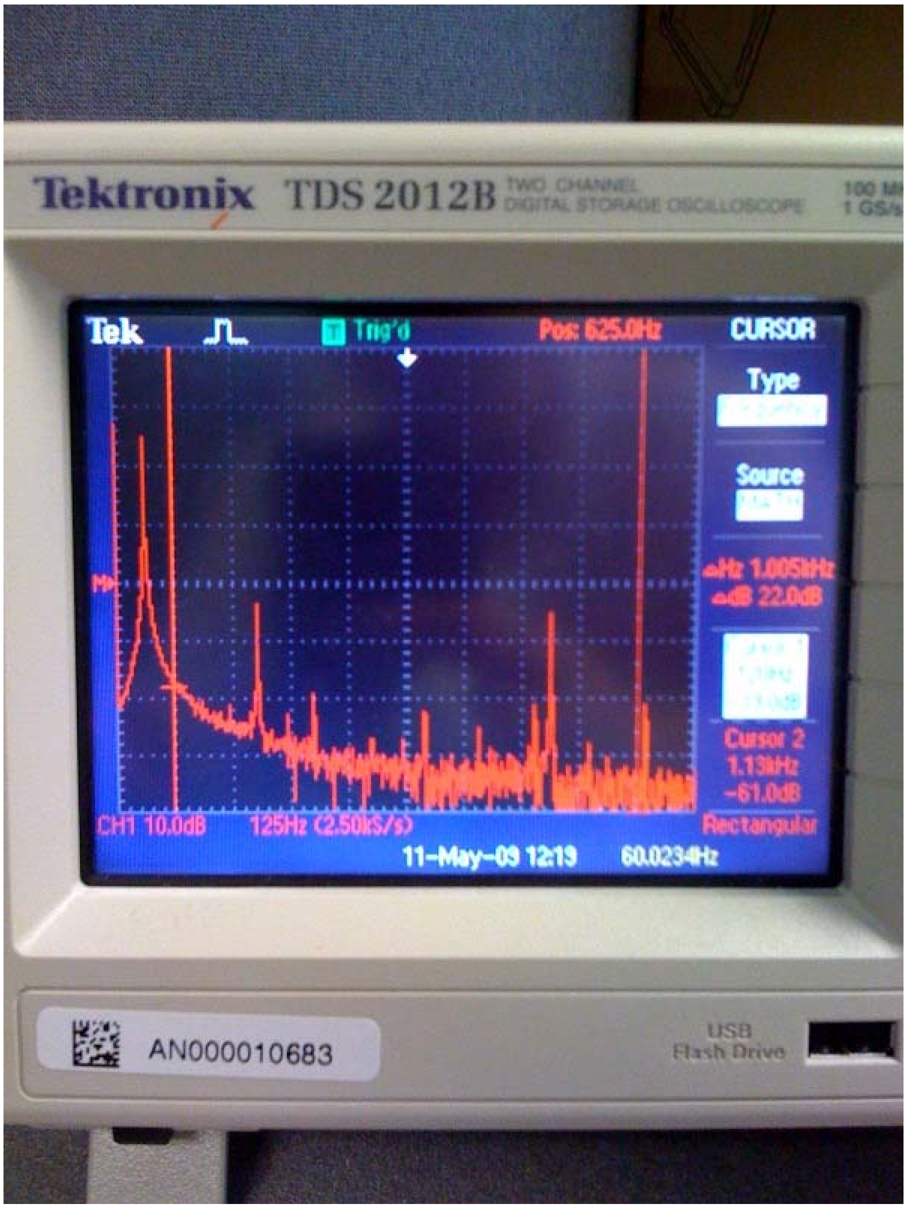
**Analog Interfacing Circuit Noise**




The average pk-to-pk amplitude of the signal centered 0.96V (20k thermistor) is 2mV. Using the 1/8 approximation, the signal RMS noise is **250uV**.

Using the FFT utility, in Figure 3, there is 1/f noise and roll-off as the system bandwidth (400 Hz) is approached. The 60 hz contribution and harmonics are clear by inspection.

**FFT of analog interface circuit**



Noise summary:

Sensor interface before ADC: **116.45 uV** theoretical, **250 uV** measured pk-to-pk/8
Software measured noise after ADC/uC: **5.55 mV**

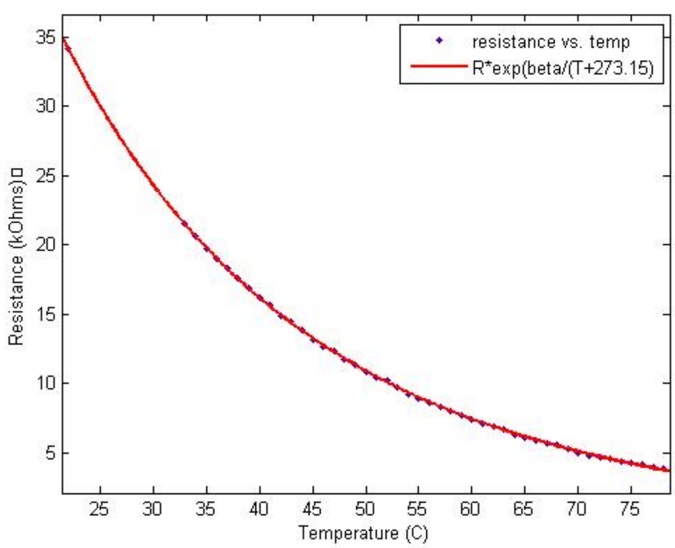
Using significant figures per the ADC resolution (5 mV), a system measurement can be no more accurate than +/- 2.5mV, per quantization error.

## Sensor Metrics

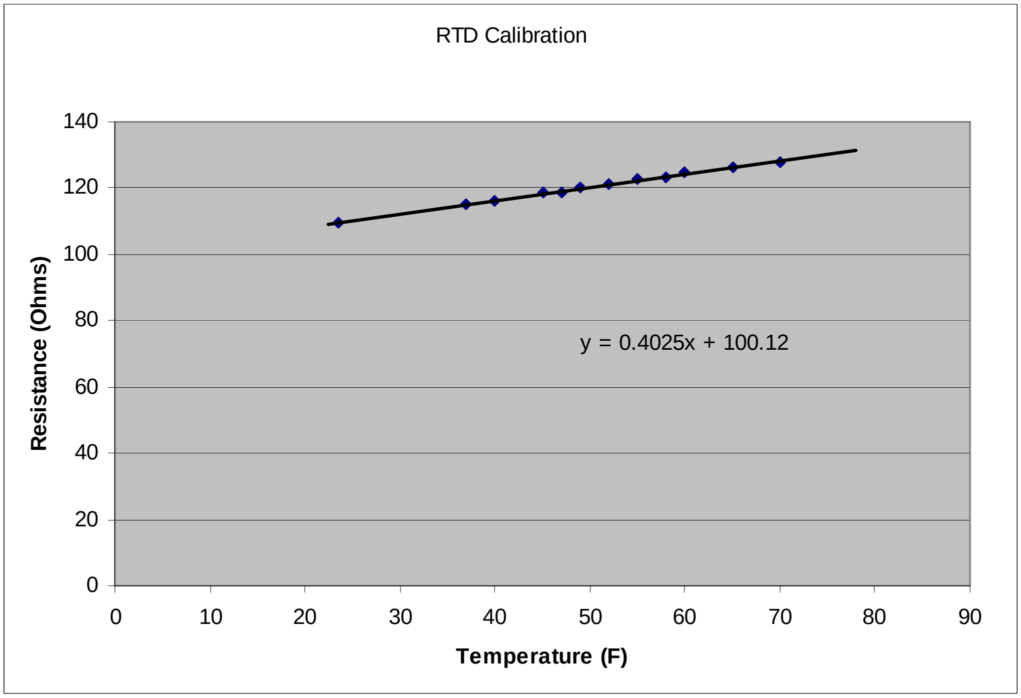
Aquatrol must accurately measure the instantaeous temperature of water, ambient air, and solar light intensity. System events (water heating, light intensity) happen on a time scale of tenths of seconds. Since the device must actuate water flow at temperatures within a safety margin, **time constant, accuracy, and resolution** are significant figures of merit for the device sensors and instrumentation.

### Thermistor Metrics

The thermistor was calibrated by measuring it's resistance at multiple values in a large heated water reservoir as the reservoir cooled. An equation of best fit was found (R=.0001371 * exp(3678/(273.15+T))-1.124) in order to produce an analytical equation for R(T), resistance as a function of temperature. A scientific mercury thermometer was used to establish "truth" for the measurement.

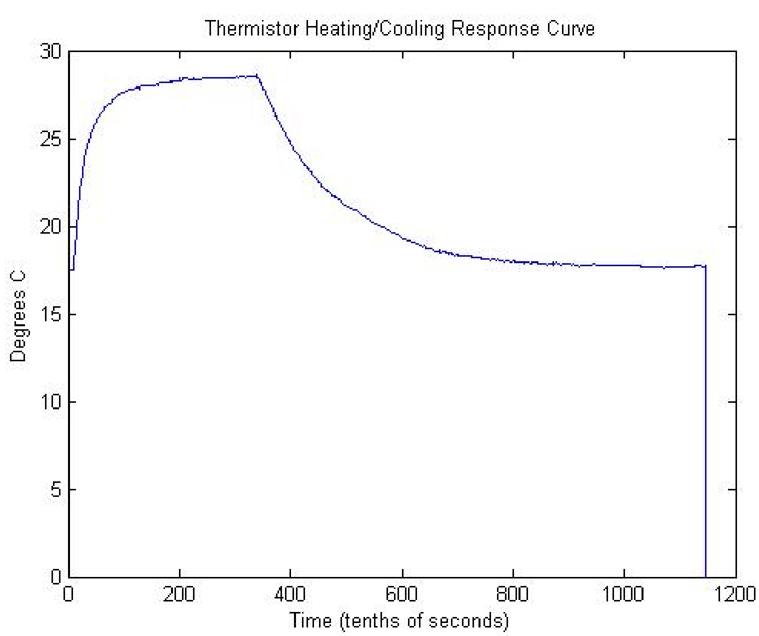


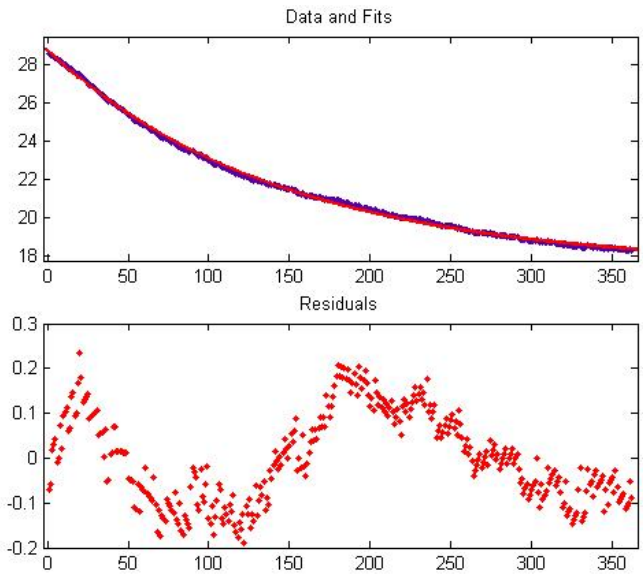

An identical procedure was used to determine R(T) for the RTD platinum sensor, which is **not** currently installed in our design. The RTD will be installed in the future to measure internal pipe temperatures or ambient temperature. Unlike the thermistor, the RTD has a linear temperature dependence.




## Thermistor Time Constant

The time constant for the thermistor was measured for both heating and cooling cycles. For the experiment, the thermistor was pinched in order to induce heating and allowed to cool in air.

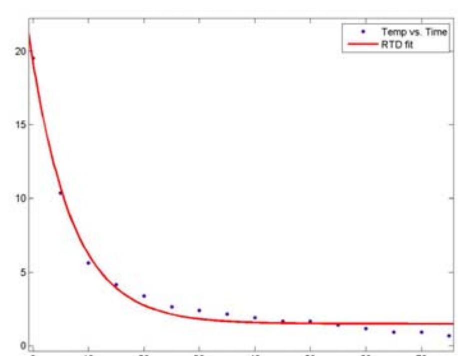




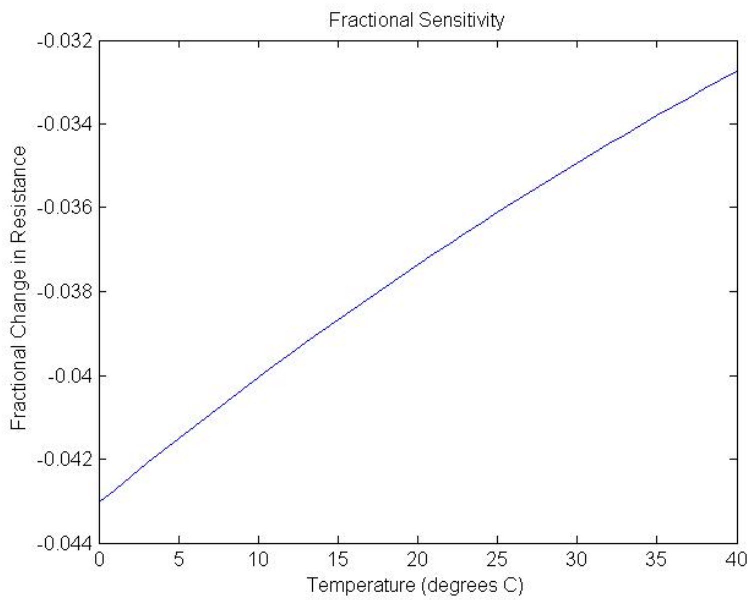

Data are shown above for the cooling of the thermistor. Matlab's curve fitting toolbox was used to determine the cooling time constant τ, which is shown above with corresponding residuals. X-axis units are tenths of seconds. The function of best fit was found to be: $T(t) = 11.23e^{-t/144.84} + 17.48.$ Using this method, the time constant τ was found to be **14.48s.** I all cases the residual never deviates from the best fit by my more than two-tenths of a degree. A similar procedure was used to measure the heating time constant, which has the form $T(t) = 1 - e^{\frac{-t}{\tau}}$. The heating time constant was found to be 0.6 seconds. This was much faster than cooling because of direct contact with the heat source (human fingers).

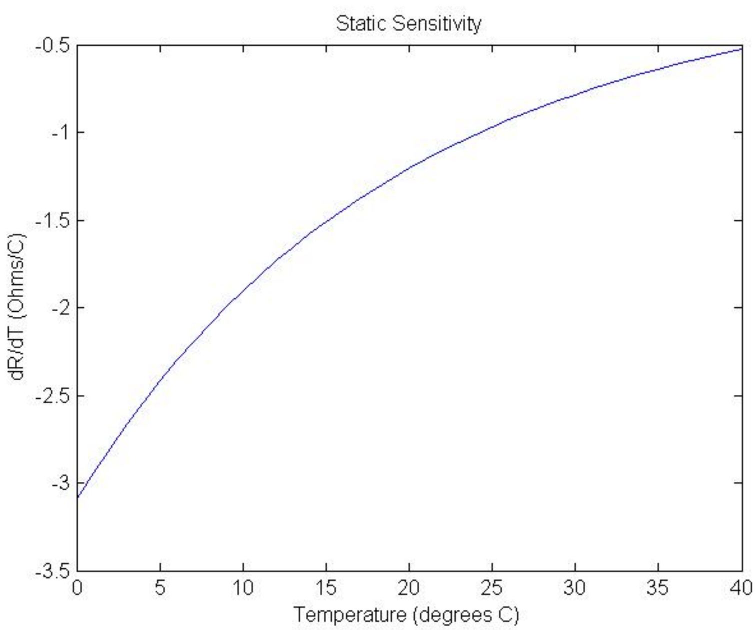
## <u>RTD Time Constant</u>

The RTD time constant was measured by dropping the RTD, initially at room temperature, into a bucket of ice water. The line of best fit is: Temperature(t) = 1 7.85*exp(-0.1356*t)+1.285 degrees C. The time constant is therefore: 1/.1356= 7.37 seconds. Since we have not yet installed the RTD, no other metrics have been measured.



# Thermistor Sensitivity



## Thermistor Accuracy

In order to measure accuracy, temperature measurements were made in a heated water bath. The water bath was heated above 40 degrees C and allowed to cool. The temperature in the water bath was carefully read with a digital scientific thermometer. This temperature was assumed to be the true temperature. Measurements were made at 40, 35, 30, 25, and 20 degrees C. Additionally, temperature was measured for an ice bath (0 degrees). The following table records the results. Accuracy of the system could have been improved by calibrating at many points, instead of just 0 and 40 degrees.

| True Temperature | Measured Temperature | Difference |
|---|---|---|
| 40 | 39.6 | 0.4 |
| 35 | 34.4 | 0.6 |
| 30 | 29.4 | 0.6 |
| 25 | 24.6 | 0.4 |
| 20 | 19.5 | 0.5 |
| 0 | 0.268 | -0.268 |

## Thermisotr Repeatability

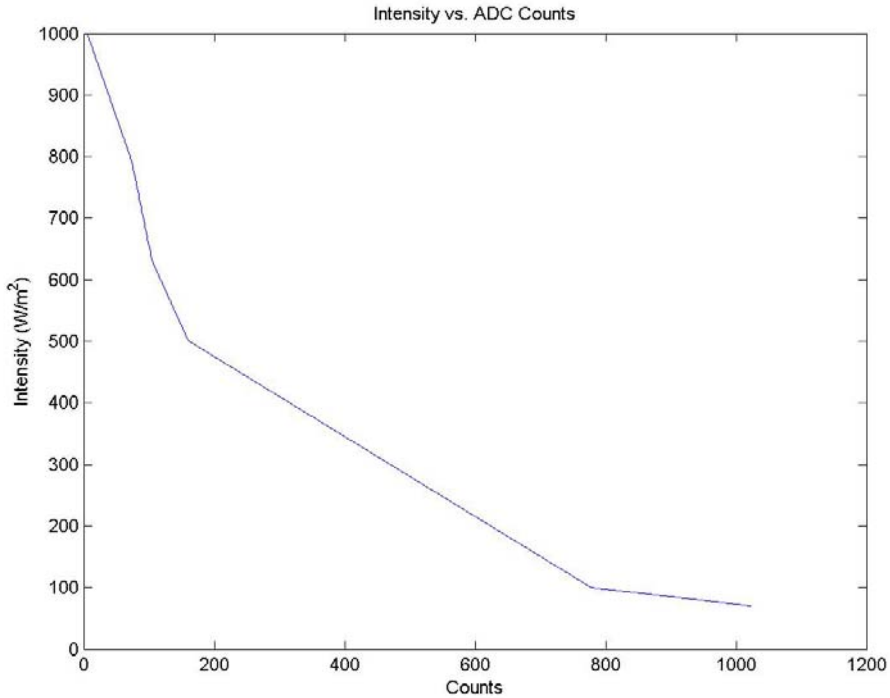
Within the program code (sampling rate = 240 Hz), the temperature was sampled once every 5 seconds or 1200 samples for 15 minutes. The script was saved in a text file using Hyperterminal, and the text file was read into Matlab, in order to pick out the minimum and maximum values. The max value was found to be 18.269, while the minimum value was found to be 17.962, which is a difference of 0.307 degrees. I performed this experiment in open air, and it is possible that the air conditioning system may have activated during this time. In the future it would be better to place the thermistor in a large reservoir equilibrated to room temperature.

## Thermistor Reproducibility

Three temperature measurements were taken with the thermistor in ice water in the morning, afternoon, and at night. They were found to be: 0.0527, 0.268, and 0.229, respectively. Given that the temperature of the ice water should have been the same, I cannot account for the system drift. It is possible that the resistance of the thermistor changed slightly due to the proximity of the lead wires. Resolution is the smallest change in temperature that can be measured. Repeatability is the change in the measurement over very short time scales. Difference over such short time scales are likely attributable to noise. Reproducibility is the change in the measurement over much longer time scales. Good reproducibility is characterized by low system drift. For instruments that are not self-calibrating, reproducibility typically tends to degrade over time.

### Photoresistor Metrics

The photoresistor circuit was calibrated as follows. At high noon (assumed solar intensity = 1000 $W/m^2$), the resistance of the CdS photoresistors was measured when covered with different neutral density filters. Measurements were made with filters of optical density = 0.1, 0.2, 0.3 1.0, and 0 (no filter). These data points were linearly interpolated to generate an intensity vs. resistance plot. Based on this data and our bridge circuit design, a 10-bit lookup table was generated mapping intensity into ADC count. Due to inability to adjust light intensity, we were unable to determine the devices accuracy resolution, or repeatability. The device's solar tracking capabilities will work, as long as the relative values produced by the solar sensing circuit our accurate, which is true for our design. However, these values must be known with precision in order to accurately determine the system's efficiency. Shown below is a plot of intensity vs. ADC counts.

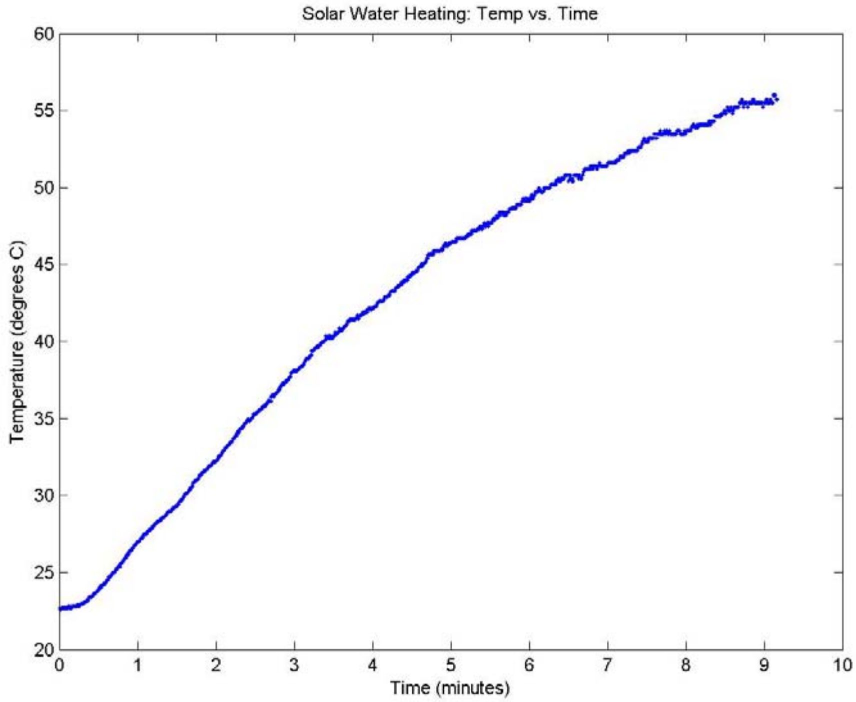



## <u>VI. Results</u>

### Efficiency

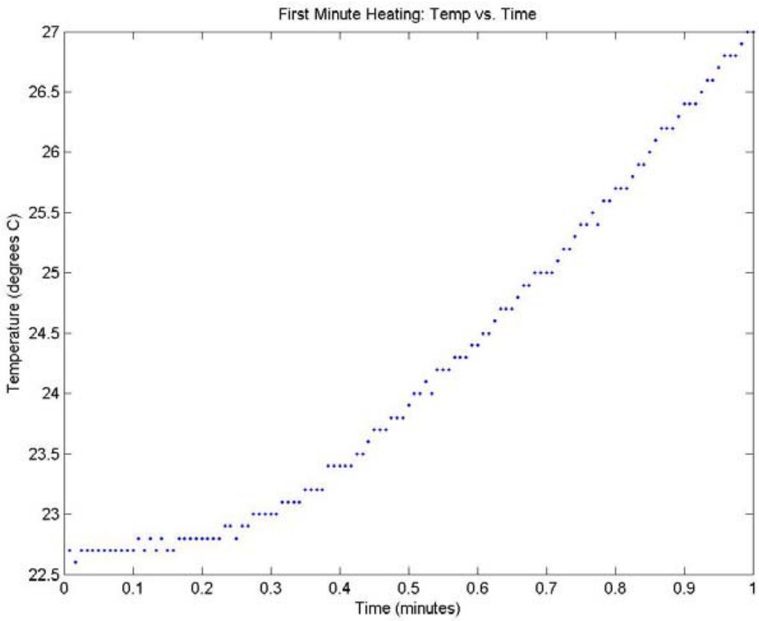
One key performance metric for our solar water heating system is efficiency. We define the efficiency $\eta$ as the percentage of incident solar energy directly converted into heat at the end of the batch heating process. In the batch heating process, water enters the heating chamber at initial temperature $T_i$, which is the temperature of the municipal water supply, and exits at final temperature $T_f$ (55 °C). Each batch consists of approximately one quart or more precisely, **(841 g)** of water.

The area of our lens is 28x38 inches or **0.686 sq. meters**. Since measurements were taken on a bright and sunny day around 2:00 PM, we assume a solar power of **1000 W/m$^2$**. This implies a total power input $\boldsymbol{Q_{in}}$ **of 686 W**. The following temperature vs. time plot illustrates the batch heating process taking place under these conditions. In approximately 520 s, the temperature rises by **22°C**.




Since $Q_{out} = mc_p \Delta T$, the total heat generation is 77400 J during the cycle, which equates to a power output of **149 W**. $Q_{in}$ = (686 W/s) (520 s) = 356,720 J. Therefore $\boldsymbol{\eta}$ **= 21.7%.** Although this efficiency figure is superior to thin film solar cell efficiency (10-15%), it less than that of a flat-plate solar collector (41%)[1]. Although the efficiency of our device is currently less than flat-plate technology, we believe that the efficiency could be dramatically improved if better materials and a smaller heating chamber geometry were employed. The current system features a stainless steel heating chamber insulated with silver bubble insulation. Heat loss would be minimized if heating chamber were made from a quality ceramic material. Additionally, the large batch size (almost one quart) and geometry (long cylinder) are problematic because they increase surface area. Heat loss could also be minimized if a smaller batch size or more spherical geometry were employed.

To gain insight into the potential efficiency of the system, we analyzed heating within the first minute, before the temperature gradient increased to sufficient levels to create significant heat loss. The plot below shows a temperature increase of 4.3 °C within the first 42 seconds.




.

**This corresponds to an "instantaneous" efficiency of 52%.** This figure is indicative potential efficiency of the current system with perfect insulation. The power output with perfect insulation would be **356W**. This figure does not include the energy required to move the lens, open the valves, and power the microcontroller. Further effiency gains could be realized if a glass window with an anti-reflective coating were used instead of generic borosilicate.

## Device Power Consumption

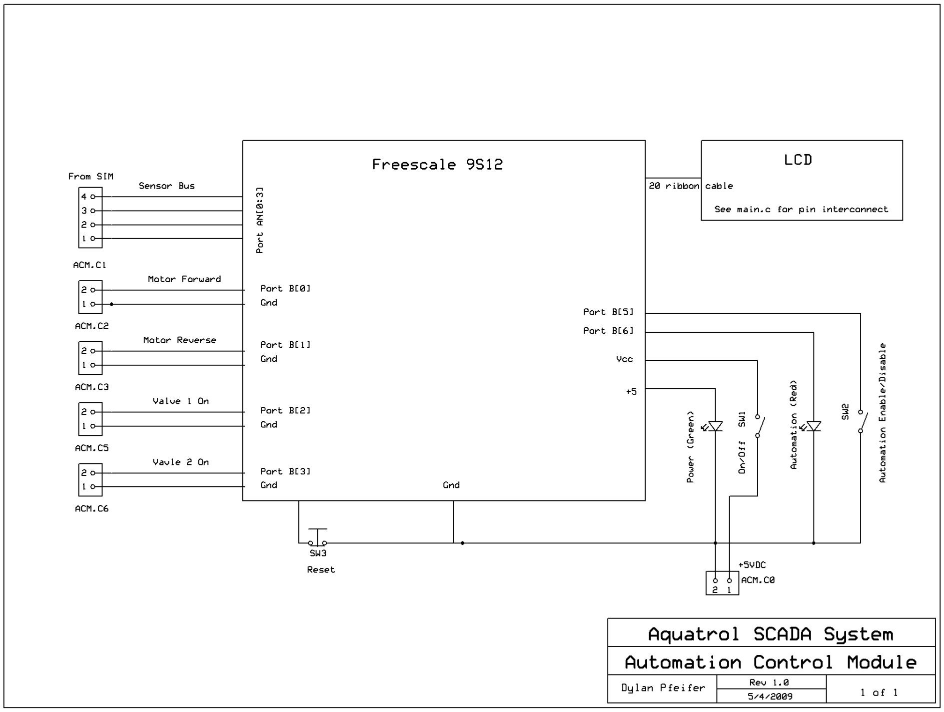
The control system for our device consists of a Freescale 9S12DP512 microcontroller, 9 integrated circuits, **two 12W valves, and a 24W motor**. Freescale lists the supply current of the **DP512 at 65 mA**. Although power consumption could be minimized by turning off ADC interrupts, while the device "sleeps" during the night, we take this value as a constant current draw. The 9 ICs (three 2.5 V voltage references) are each rated at 10 mA. The **total circuit supply current is thus 155 mA**. The table below outlines the expect power draw for all components during a typical day. The load factor is the percentage of time each device is on. The valve and motor load factors are based on the production of 18 gallons of hot water: motor movement for 30 seconds, every fifteen minutes, during 8 hours of high intensity sunlight with valves opening for 2.5 seconds 72 times a day. Although the motor is capable of drawing 2.1 A, in practice it was measured to only draw 800 mA.

| Device | Current (mA) | Voltage (V) | Load Factor (%) | Power (W) |
|---|---|---|---|---|
| DP512 | 65 | 9 | 100 | 0.585 |
| REF03 (3) | 10 | 5 | 100 | 0.05 |
| INA122P (3) | 10 | 5 | 100 | 0.05 |
| OPA2350(3) | 10 | 5 | 100 | 0.05 |
| 1/30 HP Motor | 800 | 12 | 1.06 | 0.10176 |
| Asco Solenoid Valves (2) | 1000 | 12 | 1.6 | 0.192 |
| | | | **Total Power (W) =** | 1.02876 |

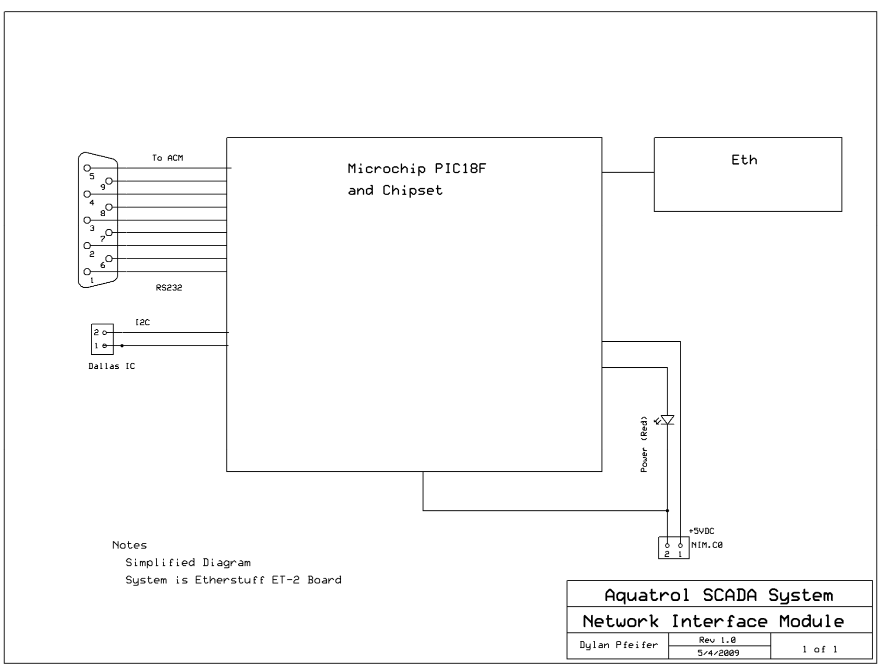
Since the total average power draw is 1.03 W, the device consumes 89 kJ of energy each day. It is currently powered by a 9 Amp-hour, 12 Volt battery, which stores 389 kJ of energy. Under these conditions, the device will run for 4.4 days without recharging the batteries. The control system uses a relatively small amount of energy compared to the energy generated by the system, and thus has little effect on the device's efficiency. During daylight hours, it would effectively reduce the devices power output by only one watt, dropping it from 149 W to 148 W. Based on these figures, it is likely that the system could be recharged with a small solar cell, and operate for years without maintenance, if properly weatherized.

## VII Schematics

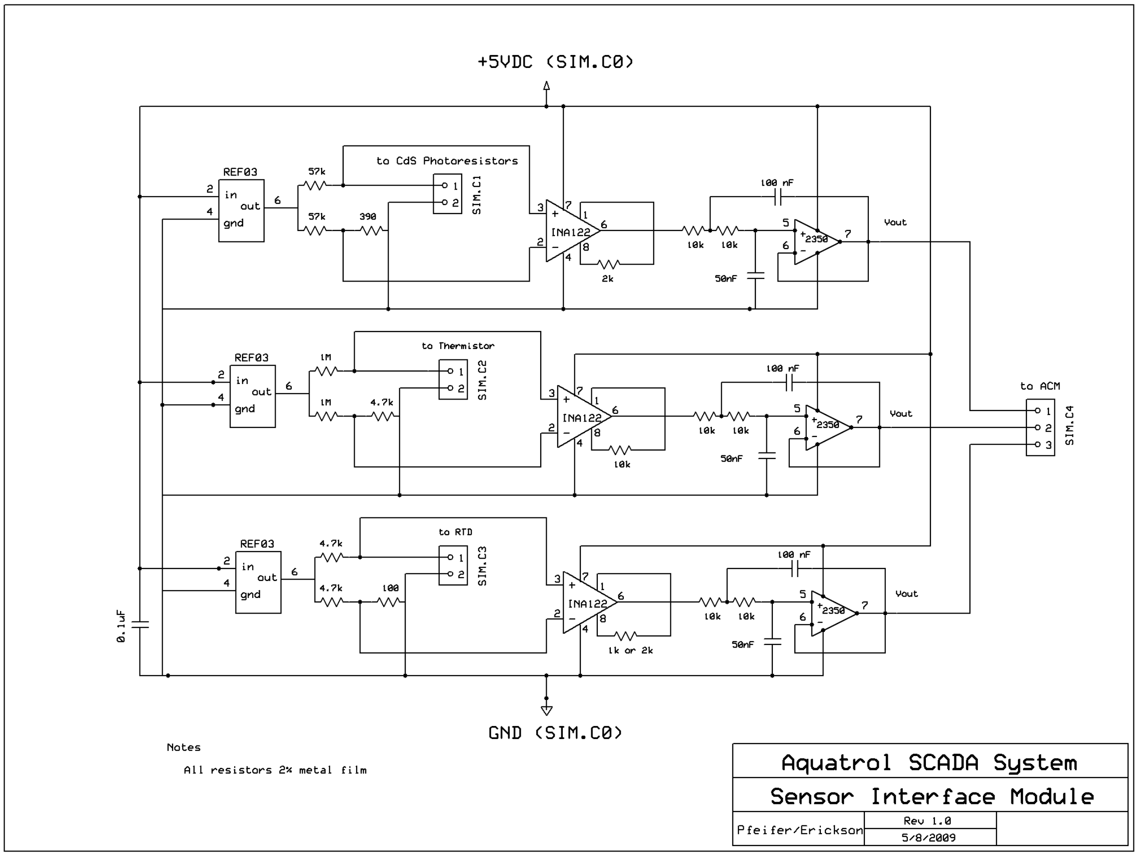
## Figure SCH.1 ACM



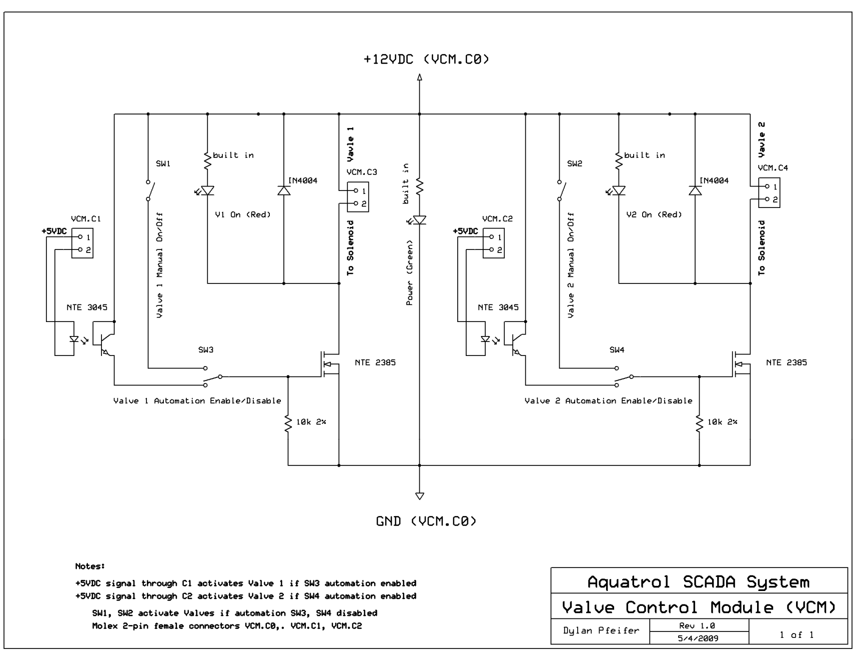

## Figure SCH.2 NIM (generalized)



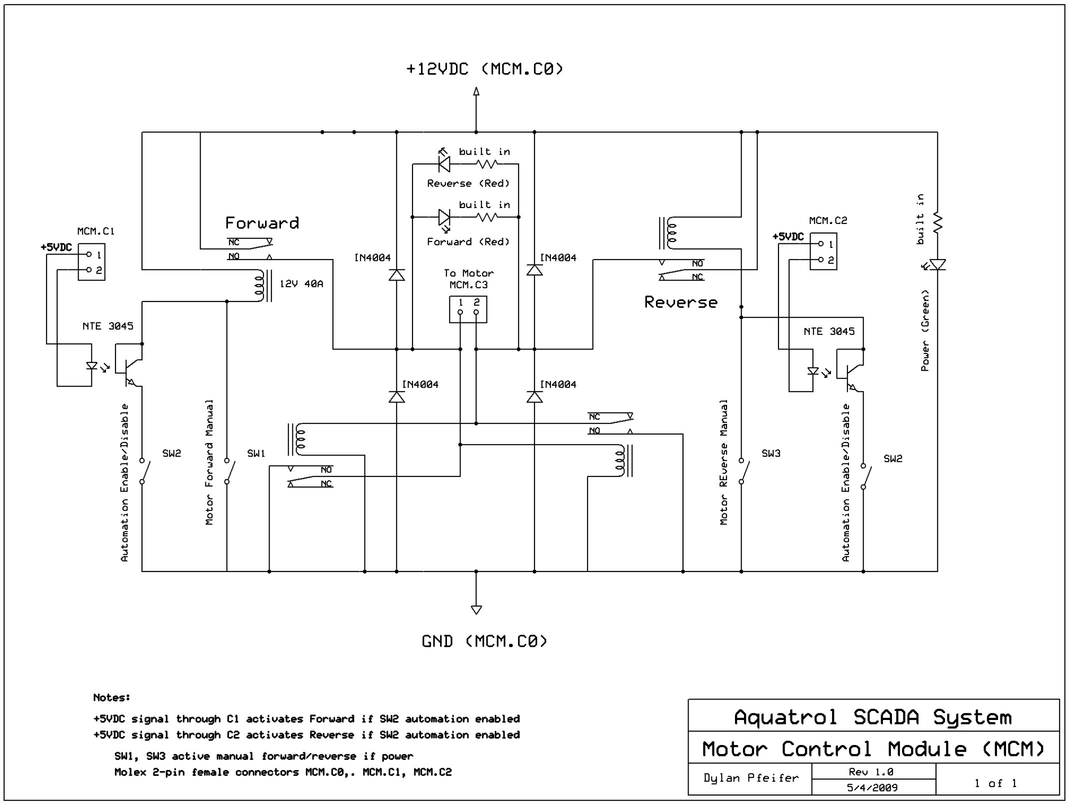
## Figure SCH.3  SIM



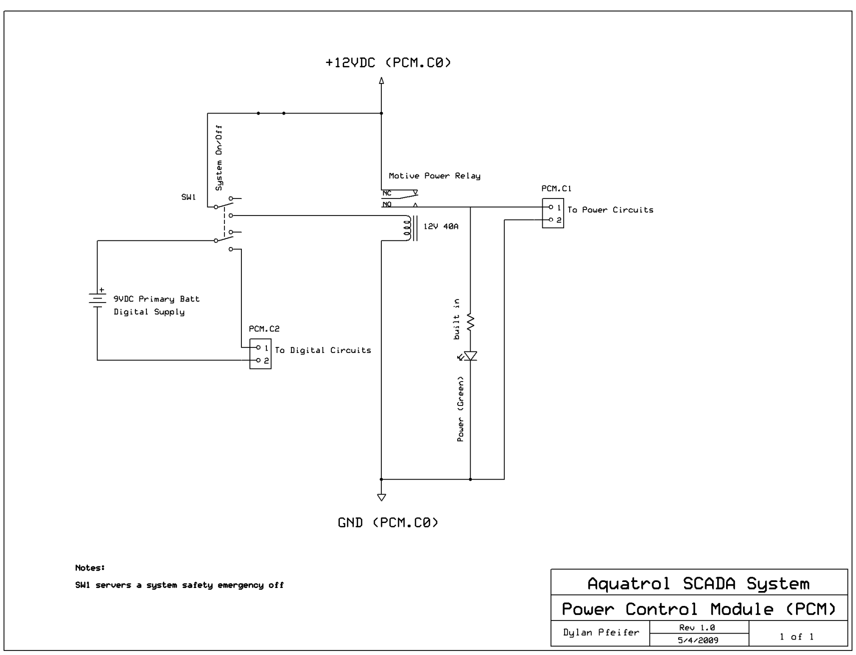

## Figure SCH.3  VCM




## Figure SCH.4  MCM




## Figure SCH.5 PCM



# Figure SCH.6 PM