Building Your Custom Measurement System

By Randy A. Normann for OnMeasurement.com

Introduction

This report outlines one example for building a custom set of data collection electronics. It is the author hopes the example provided here directly benefits the reader and generates useful ideas for building the system meeting your own needs. Note, there are a few names of suggested suppliers within this document, the author has no agreement with any of them; they are provided as a general aid for getting started.

The Example Circuit

I'm building a moderately complex data collection system based around the Arduino Mega. I will use this system to test my Heat Flow Sensor, HFS. The system will run from a 9V (6XAA) battery pack during actual use. For testing, I will use a 9V DC wall plug in converter. A basic block diagram for the circuit is shown below. Besides the Arduino Mega, I'm planning on 6 shields and a small hand soldered interface board for connecting to the HFS and other sensors as needed. I'm planning on at least two high precision signal measurements using the AD22B and Analog Mux shields. There are also two instrumentation amplifiers with two RC filters used as anti-aliasing filters. Other less sensitive signals can go directly to the analog inputs on the Mega. All in all, this system has the Mega 16 analog inputs at 10Bit resolution and 16 analog inputs at 22Bit resolution. There are 2 fully differential inputs. That's a lot of analog signals. To save all that data, a micro-SD card shield is needed.

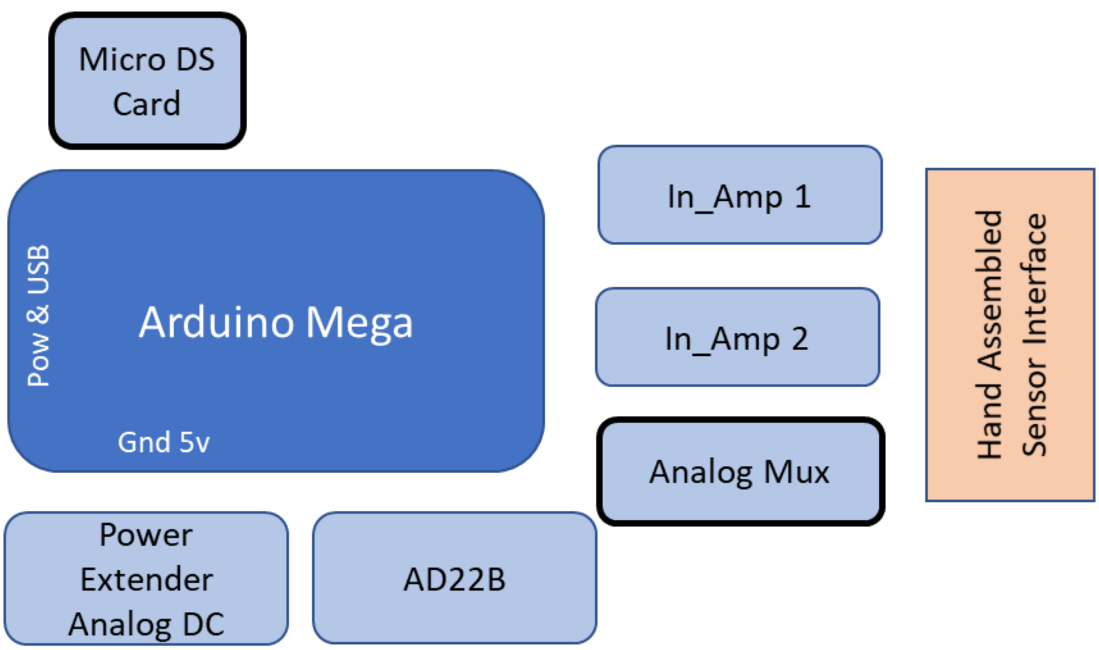

Figure 1. The proposed circuit based on the Arduino Mega.

Mounting the Arduino and Shields

I now have my circuit, see Figure 1. Wiring this large a system with header pins and wire is doable on any flat table top. Keeping this circuit together while moving it around the lab or further is a problem. I will build this system on to a metal plate using standoffs for now. I may not want an enclosure for this first prototype build. However, planning for a box could prove beneficial in the future. Meaning, the size of the metal plate should be large enough to mount my circuits and the correct size to fit a possible future enclosure.

In most cases, I will normally use the metal plate as my system ground. This provides something of a signal return plane under my circuits. Using batteries doesn't require an earth ground. Running from a non-isolated (normally inexpensive) DC plug in converter results in a connection back to earth ground via the power cable and wall socket. In short, using a metal conductor as a ground can lead to electrical shorting. Mounting circuits on wood or plastic does create a potential fire hazard.

Mounting your circuit directly inside an enclosure (plastic or metal) is normally difficult when working with so many interconnections. Having the circuit on a metal plate allows for it to be mounted inside most enclosures by using heavy standoffs between the enclosure floor and metal plate or by sliding the metal plate into matching groves built into the enclosure. The latter is the case shown in this example.

If I have a sensor which is sensitive to my system ground or earth ground, I can use the instrumentation amplifier circuit to isolate the sensor signal using the differential amplifier inputs. Normally the 2 small RF capacitors on the differential amplifier inputs (OM: In_Amp) will not cause an issue but you can remove them if needed.

What type of metal sheet? I have used heavy (1/4in) aluminum when wanting something to look really nice. It's possible to drill the mounting holes and tap them. This makes removing shields easier in the future and the aluminum is stiff, making mounting into an enclosure easier and more secure. However, if you're going to tap holes, you need mounting holes which match the board holes within about 0.005". Not possible with a hand drill.

For this example, I will be using a 0.04 inch thick sheet of 304 stainless steel. Thinner sheet steel is too flimsy while a 0.06 inch is OK, but harder to cut. The 40mil material is strong enough for the moderate size of my circuit. You will need to cut the sheet to match your circuit and enclosure. The fabrication of metal is outside the scope of this document. I will mention to always have safety equipment, training, and the right equipment when working with metal. Freshly cut steel has very sharp edges and contains burrs. Care is needed for handling. Deburring with a file or sander is required. Also, there are some metals that look like and feel like aluminum, but they are not aluminum. The dust particles from these metals are very hazardous to your health. Know what you are working with.

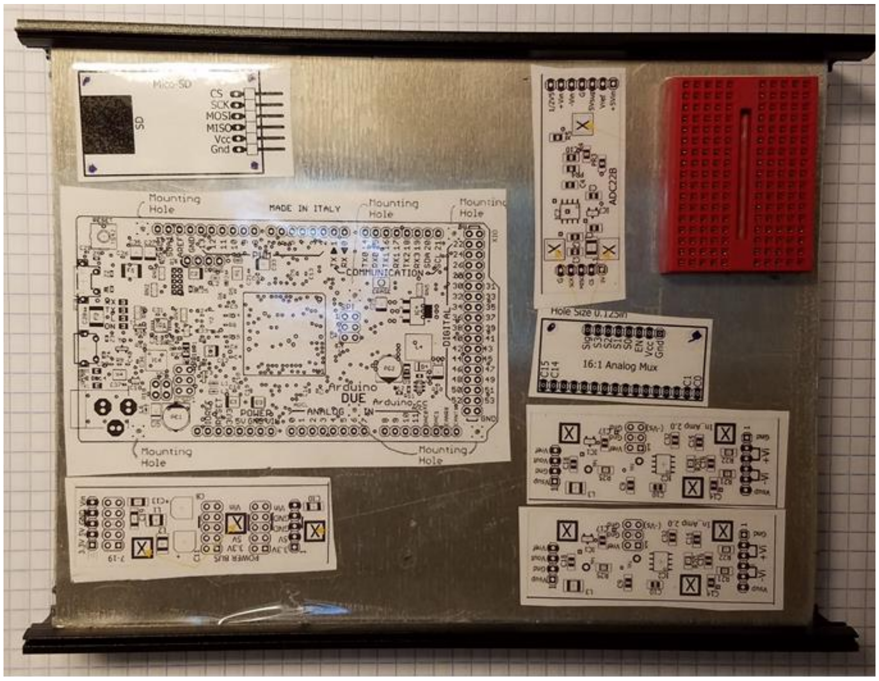

Figure 2. The layout using dolls taped on to a sheet of 304 stainless steel, 0.040 inch thick. The dolls include mounting hole locations.

If you're on a strict budget, the 304 stainless might be too expensive. Many hobby stores sell sheets of polystyrene, 0.060 inch thick. About 1/10 the cost of stainless, but you lose your ground plane. However, polystyrene sheets are super easy to work with and can be directly written on with a permanent marker.

For physically laying out the circuit, I use paper dolls. I make a trace around the shield circuit board along with making where the mounting holes are located on paper. At www.onmeasurement.com, I have an MS Word file containing some paper dolls to cut out or to make digital layout in MS Word. Also, at OM are data sheets for some of the shields used in this assembly. The data sheets have 1:1 layouts. Print these in "actual size" mode and you should have a matching doll. Figure 2 is a photo of my layout.

Figure 3. This is the 4-40 standoff hardware used in this example.

The clear glossy shine (Figure 2) is from the tape holding the dolls to the metal sheet. I used heavy clear packaging tape. It comes in a wide 2-inch roll and it's thick enough that I can touch the drill bit tip to the tape verifying hole alignment before drilling when using a drill press. The red prototype plug-in board was used for testing my circuit and software. Later, it was replaced with a small perforated solder board. In general, I never like to build a prototype using a plug-in board. Over time and with usage, the plug-in circuits start having problems resulting in constant trouble shooting bad connections.

With the dolls in place, you're on to drilling the mounting holes. Arduino boards fit metric M2.5 or M3 or SAE 4-40 mounting hardware. OM board mounting holes are 0.112 inch hole size to fit metric M2.5 or SAE 4-40 mounting hardware. The OM pad is a large 0.2 inch square for a nut or screw head. Unfortunately, the Arduino's have some mounting holes with little clearance. So, the Arduino's work best with a female stand off and a screw. Figure 3 shows the mounting hardware for an SAE 4-40 setup. The stand off is steel. It's 0.2inch tall with a 0.3inch stud. Standoffs come in many sizes and materials. I have only worked with three types, steel, aluminum, and Nylon (for non-conductor reasons). In my experience, Nylon tend to strip threads and aluminum standoffs are normally mated with a stainless-

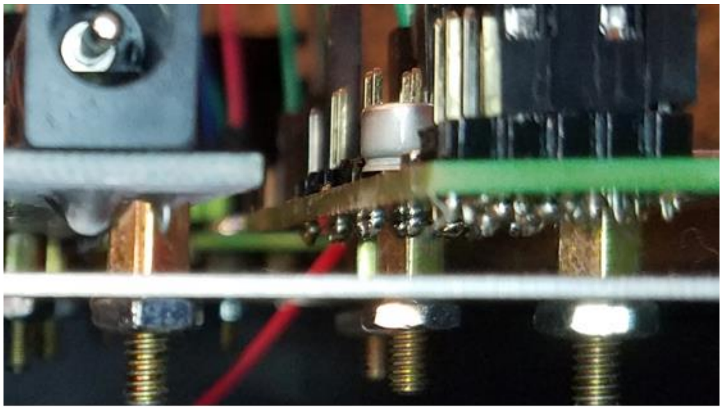

Figure 4. Photo showing board mounts.

steel screw and nut. A few temperature cycles and they can become loose. Steel mounting with a little blue Loctite® works well.

When using 4-40 mounting hardware, the drill size for mounting plate should be #31 (0.12in), 1/8 in (0.125in) or even #32 bit (0.1285in). The larger the hole, the more wiggle room for positioning the board mount. I always have a few off centers even when using a manual operated 3HP metal working drill press. When using a handheld drill, it's a good idea to use a punch to place a small dent in the location to be drilled. In any case, it's a good idea to have a wooden board backing your metal plate when drilling to prevent the plate from bending while drilling. Finally, after drilling your holes, manually deburr the hole using a deburring tool or a drill bit several times larger than used to drill the hole. Figure 4 is a photo showing boards mounted to the steel plate.

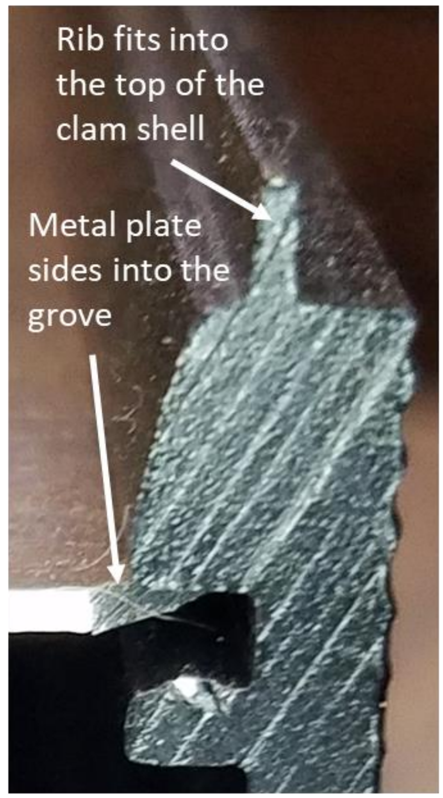

Figure 5. The circuits are mounted to the 0.04 inch 304 SS plate. The plate edges are smoothed and the corners removed. System slides in and out easily.

Mounting your circuit and metal plate into an enclosure is a good idea. It looks much better, and the enclosure protects your circuit both physically and electrically. I have worked with two basic types of enclosures, metal rack mount boxes and "extruded aluminum" metal boxes, some built as a clamshell.

The easiest means for getting an enclosure worthy of your efforts is to have one fabricated by a company like Protocase (https://www.protocase.com/). One of the hardest things to do is to manufacture mounting holes for connectors, LEDs, and displays. Protocase has free software for designing your enclosure with a good list of built-in options. This example will not take the easy route.

Going back to the example; The benefit of most extruded aluminum enclosures, they come with groves (slots) in the side walls to hold your circuit board or metal mounting plate. These groves are 0.06 to 0.08 inch wide. This allows your circuits to slide out of the enclosure by removing the end plates. Figure 5

Figure 6. Shows is the circuit mounted and wired inside ½ of the extruded aluminum “clam” shell enclosure. One of the end plates was mounted to the lower clamshell with the circuit to mount a 5 pin connector.

shows the 0.04 inch mounting plate slid into the enclosure circuit mounting grove. Figure XX shows the enclosure and circuits I’m using in this example. The enclosure is called a clam shell because the extruded aluminum box is split down the length of the box. The two halves are held together by mating slots-groves and by the two end plates. Each end plate has 4 small screws. When installed, the clam shells and end plates form a complete enclosure. These enclosures are not for use in the outdoor elements. However, as can be seen in Figure 6, 1 clamshell enclosure can be used to make 2 open face lab prototype systems.

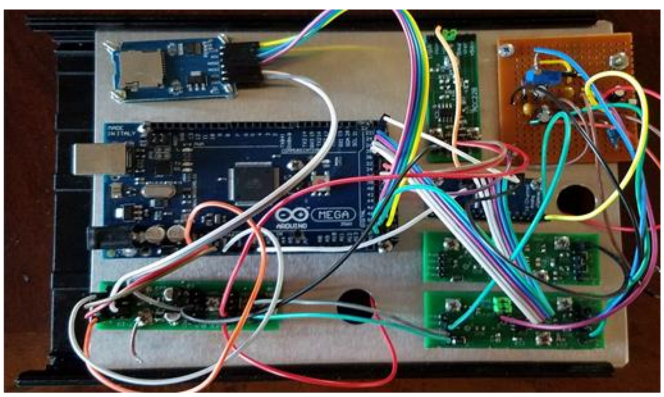

Figure 7. The finished product, a custom data logger. A lab prototype Arduino Mega with 7 shields and one hand assembled sensor interface board.

Figure 7 is of the completed custom-built laboratory data logger with micro-SD card, 16-10bit and 16-22bit ADC channels. There are two instrumentation amplifier gain circuits with 2 pole, anti-aliasing filters.

Using the power extender shield (OM), I have a star grounding system for my sensors and DC filtered supply voltage for analog circuits. I wanted to test this data logger for high resolution measurements. So, I ran two tests, a noise floor bench test and a low noise outside test with an actual HFS cable.

Some Testing Results

The first test was done on the bench with fixed resistors replacing the sensor inputs. The resistors were 0.1%, 20ppm, 1.2K and 750K ohm resistors, see Figure 8. They were soldered directly to the mating (female) connector to provide a low noise input without a long cable. Not shown was a large value capacitor across the 1.25Vref. With this as our input, the noise in the system will be dominated by the data logger itself. This is our noise floor of the system just built. Figure 9 is of the sensor input circuit.

The instrumentation amplifier has a gain of 100 and is providing a differential input with an output range of 0V to 2.5V. That output range maps into -1.25V to 1.25V divided by 100 or an input voltage range -0.0125 to 0.0125. With our high-resolution ADC, we hope to see very small analog voltage signs coming from our sensor.

Not shown is the measurement of the ambient instrumentation temperature. The instrumentation temperature sensor was a 1000 ohm RTD sensor using a 2K precision series resistor. The temperature measurement also used the high-resolution AD22B DAC through the analog mux. This high-resolution temperature reading is needed to correct for temperature drift of the measurement system.

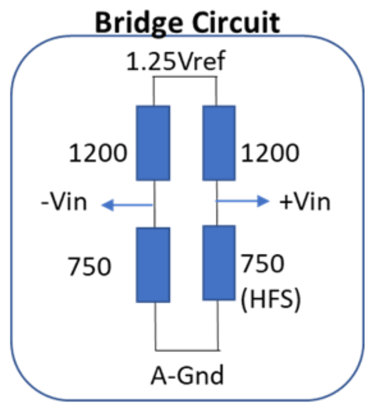

Figure 8. Low noise, passive fixed value sensor simulation for noise floor testing.

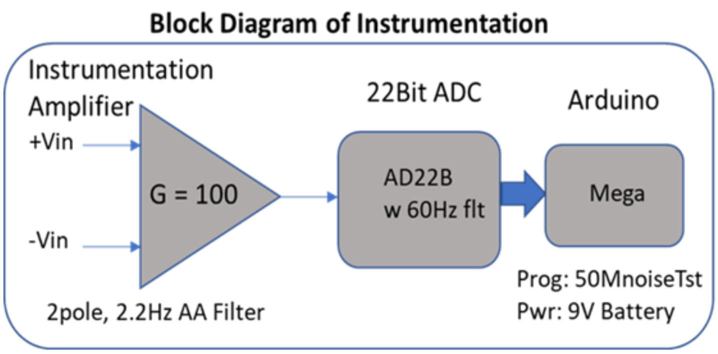

Figure 9. The system instrumentation amplifier and AD22B under test as built.

Although the test was run with the instrumentation open to the air and local 60Hz noise, the test results were good.

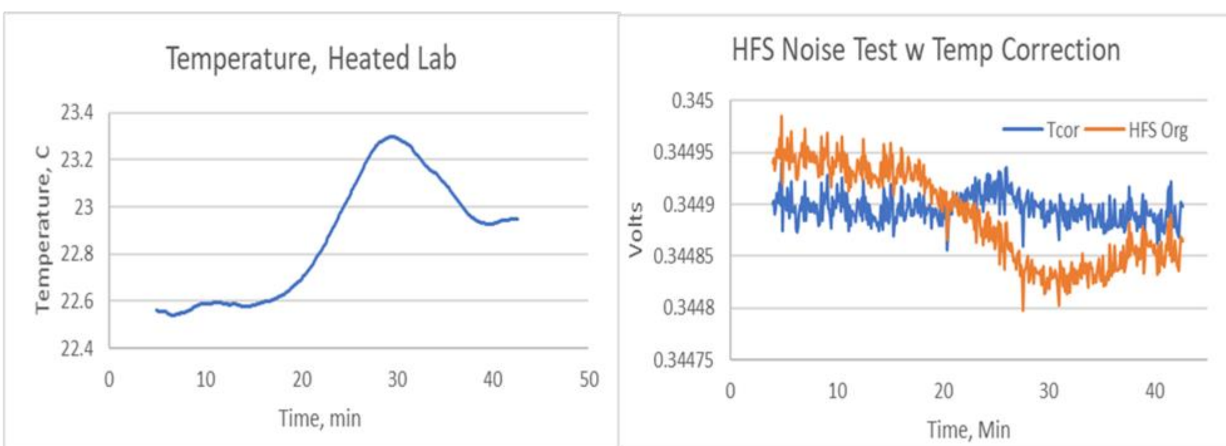

Figure 10. The plot to the left is from the instrumentation temperature sensor. The lab has central heat on a cold day. On the right the HFS voltage reading are from the simulated sensor. The Tcorr readings are the HFS original reading after temperature correct was applied.

The voltage readings for the precision resistor bridge are shown in Figure 10 as HFS Org. HFS stands for my Heat Flow Sensor. With a gain of 100X and a 22bit ADC, this system is looking at very small voltages and is temperature sensitive as all high-resolution data loggers are. Before going to a field test, the temperature sensitivity should be corrected for.

The data logger is also monitoring temperature using a 1000 ohm, RTD. This allows for tracking the temperature of the data logger. Figure 10 left side shows the local instrumentation temperature following the lab air temperature. Figure 10 right is providing the fixed resistor simulated HFS readings for both with and without temp compensation. The standard deviation of the non-temperature compensated readings is 44.8μV. After compensating for the changing temperature (a simple linear relationship delta V to delta C), the standard deviation is reduced to 13.6μV. Remember, the instrumentation amplifier has a gain of 100X. Our standard deviation from the sensor is only 0.136μV (or as ~150nV). These are very low-level sensor outputs.

Figure 11. The custom built data collection system is working on the dock. The 50 meter sensor cable is in about 10ft of water.

To test the noise level with 50 meters of the Heat Flow Sensor cable, I needed a very stable temperature. I did have a lab or test oven stable enough for testing this long sensor. So, I used a near-by pond. I placed the water tight sensor down in 10 ft of water and ran noise level tests, assuming the pond water at depth was static for the duration of my test. A photo of the data logger on the deck with a sensor cable connected is in Figure 11.

The good news, the noise floor measured about 4nC/m for 1 standard deviation. Wow, that's temperature sensitivity.

Conclusion

I hope this example aids the researcher/reader in building their own scientific data loggers. There are many ways of building such a system. There are many options of board mounting and making electrical connections. So many options can be confusing to someone just getting started.

Discussed in this example were the use of star grounding and analog supply voltage filters. These are key elements for getting low level measurements in any scientific electronic data logger. This example demonstrates a reasonable outcome. The use of a 22bit ADC allows the designer to get well below the sensitivity of the standard Arduino 10bit ADC. However, without the use of star grounding and filtered analog supply voltages, the results would have been marginally better than the Arduino by itself. More details on making low level scientific measurements is detailed in the author's book, *Arduino for Projects in Scientific Measurement*.

The next generation of this example would be to place it inside a complete metal enclosure and perhaps replace the header/jumper wire connection system with soldered wires. In any event, the system designed and developed here could be converted to a single PCB. All the circuits are in the public domain thanks to a robust Arduino community. Best of Luck, *Randy*.