

JPL Tactile Rover Wheel

David Choy, Andrew Kessler, Riley Shear

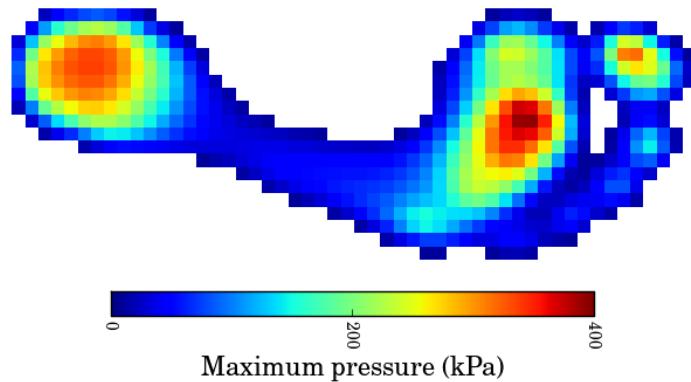
ME 218D, Fall 2013

The Mars Exploration Rover *Spirit* landed on Mars with its twin *Opportunity* in 2004. For the next five years, it performed a variety of scientific tests and measurements far in excess of its original planned 90-Martian-day duration. However, on May 1, 2009, 1892 Martian days into the mission, *Spirit* encountered a large patch of loose material; the rover lost all traction and was unable to move. Redesignated as a stationary research platform, *Spirit* sent its final transmission in March of the following year. In response to this event, which became known as "Free Spirit" the Jet Propulsion Laboratory (JPL) has funded the development of countermeasures to prevent similar occurrences in future extraterrestrial rover missions.

One potential solution proposes real-time measurement and mapping of the forces felt around the outer surface of each wheel. Such data would provide more detailed information on traction and surface consistency, allowing for improved navigation decisions. For example, a contributing factor to *Spirit's* immobilization was the deceptive nature of the terrain over which it was traveling: a thin hard crust concealing a layer of soft sand below. A sensor-enabled wheel might have detected the danger before it was too late to avoid.

The tactile wheel system, once completed, was to have the following requirements:

1. Shall provide a "heat map" of the pressure distribution of the wheel's contact with the substrate



- a. Similar in concept to
2. Shall report pressure for discrete square areas with side lengths less than 12.5% of the maximum contact patch width
3. Shall report pressure to an absolute accuracy of 10% of the minimum pressure possible for a fully loaded wheel

4. Shall use sensing technologies and components compatible with rigid “tires”
5. Shall allow continuous rotation of the wheel
6. Shall provide telemetry while the wheel is in motion
7. Shall provide updated maps at 1Hz

Additionally, should time and resources allow, the following specifications were preferred:

1. The system design not incorporate batteries
2. The system be compatible with rigid “grousers” as part of the “tire”
3. The system resolve shear at the surface of the tire and pressures (or forces) normal to the active face of the grousers

Hardware

Several different concepts for sensing method and overall structure were considered at the outset of the project. Each had its own measurable benefits and drawbacks.

The first concept was to place an array of strain gauges on the interior of a monolithic aluminum wheel. Through a precise model of the material and shape of the wheel's construction, the minute deflections in the metal of the wheel's surface could be translated into a rough map of force distribution. The system allowed for unibody construction of the wheel, the current preferred production process at JPL. However, this approach was not ultimately pursued for several reasons, first among them the cost. Strain gauges are a relatively expensive sensor and to cover the wheel with the specified resolution would exceed the given resources. Secondly, obtaining a reliable model of the wheel's deformation would require several lengthy construction and testing iterations, along with precision machining processes that were not feasible with the time and finances available.

The second concept was to employ a partially pre-fabricated solution. Several companies already build and sell systems that measure pressure distributions across a large area and output the data as a “heat map”, very similar to the intended application of this project. The advantages are clear: Employing an established solution would allow for minimal prototyping and testing, as well as economies of scale afforded by commercial products. However, these systems come in two forms: either as one-time-use sheets or as fully instrumented sensor pads, neither of which is appropriate for our application

The single-use sheets obviously would not work for a long-term Mars mission, while the reusable pressure pads would also have to blanket the entire exterior of the wheel, possibly interfering with available traction. Additionally, with the inclusion of grousers on the wheel, the pad would have to deform around these features, possibly affecting output quality. The other major consideration was the cost of purchasing such products. As mentioned previously, several companies sell such systems and a survey of these companies and their products revealed a minimum cost of \$15,000 per sensor array. Given the limited resources of the project, this concept was abandoned.

The third possibility considered was using force sensitive resistors all around the wheel's exterior surface. The sensors could be placed in any desired grid pattern, and could provide normal force readings at any desired location. If the sensors were distributed evenly, the force output could be easily converted to a pressure. These resistors are easily obtained, sized small enough to give the desired pixel resolution in the map, and, when bought in bulk, are relatively inexpensive. The downside to this system

is the introduction of many more possible points of failure. However, in the event of a sensor failure, the rest of the system would still be able to function, with only a single dead spot in the map.

Given the resources available, the decision was made to implement the third concept.

The first iteration was built as a proof of concept prototype. Using a small cardboard tube segment as the makeshift wheel, holes were cut to allow the tips of the force sensitive resistors to sit on the outside of the wheel while their tails terminated on the inside, with individual wire pairs for each sensor leading to voltage dividers driving the analog inputs of a microcontroller. The system also employed an available HC9S12E128 board to quickly write a version of the microcontroller code necessary to read all of the resistors and output the values over the serial port to a connected PC. Additionally, the first version of a PC interface was developed to map the transmitted measurements to a visual display of the pressure. Integration of all the components strongly suggested that further development would result in a system meeting the desired specifications and manufacturable with given resources.

Following the successful proof of concept, the system was scaled up to more closely align with the dimensions of an actual rover wheel. Plans were made to use a 6.5 inch diameter aluminum tube to provide the wheel's primary structure. While slightly smaller than *Spirit*'s wheels, the size was on par with those of its predecessor, *Sojourner*.

In order to meet the specifications, the sensors were placed in an 8x28 grid. Using a 6.5 inch long segment of the 6.5 inch diameter steel aluminum tube, this size array generates approximately square "pixels" in the pressure heat map. Each pixel covers a 0.722 inch (axial) by 0.704 inch (circumferential) area on the exterior of the wheel.

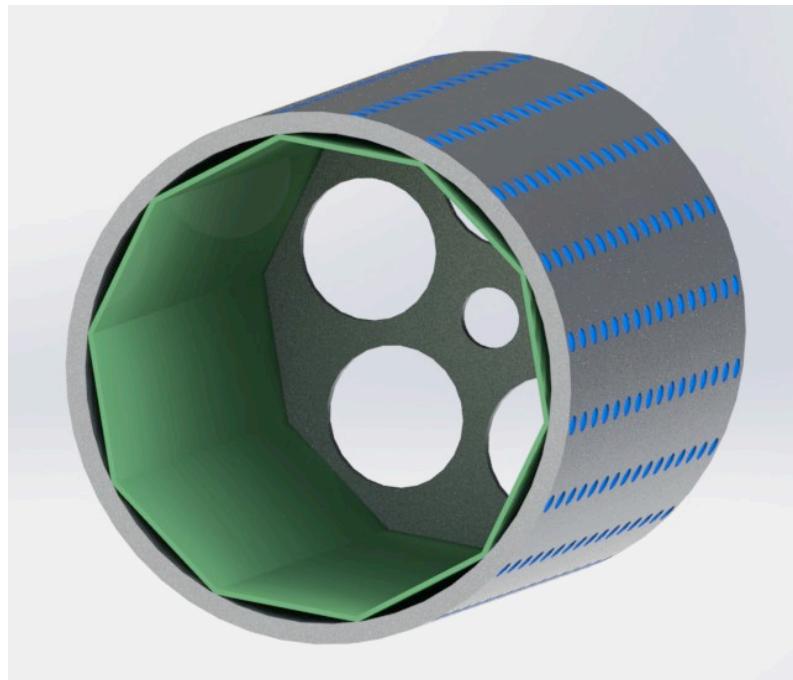


Figure 1 - Monolithic FSR reading concept.

It was at this point new problems with this design emerged. Because of the tube's small radius of curvature, the sensors did not sit flat on the wheel, complicating sensor mounting. Also, similar to the

large holes cut into the cardboard in the previous test, relatively large holes needed to be drilled through the tube to allow for the sensor tails to lead back to the interior of the wheel, which may interfere with the wheel's ability to operate. Finally, Following careful consideration about the assembly process, it became apparent that wiring 224 sensors to seven PCB's inside a six- inch opening was not feasible.

After careful consideration of a variety of options, design of the wheel was shifted to a modular system, where each section was a hybrid of machined and 3D printed parts. Each section would carry one sensor interface circuit board, four rows of seven sensors, a structural aluminum insert, and a 3D printed surface section carrying the curvature and sensor attachment pads.

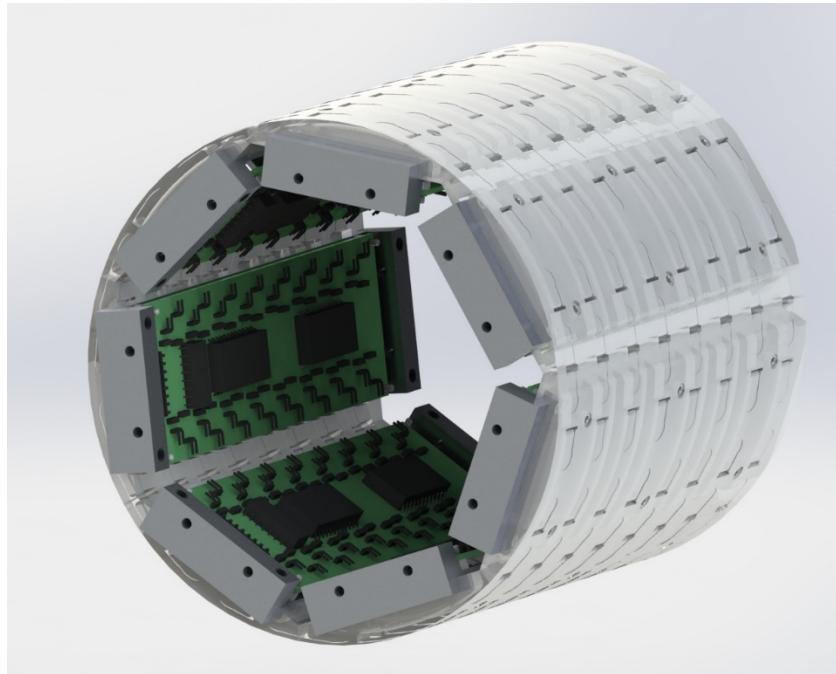


Figure 2 - Section wheel concept rendering.

On each outer 3D printed section are 28 grooves and holes for the sensors. Each sensor is slid through its respective hole and glued down with epoxy on to the pad to hold it to the wheel (Figure 3).

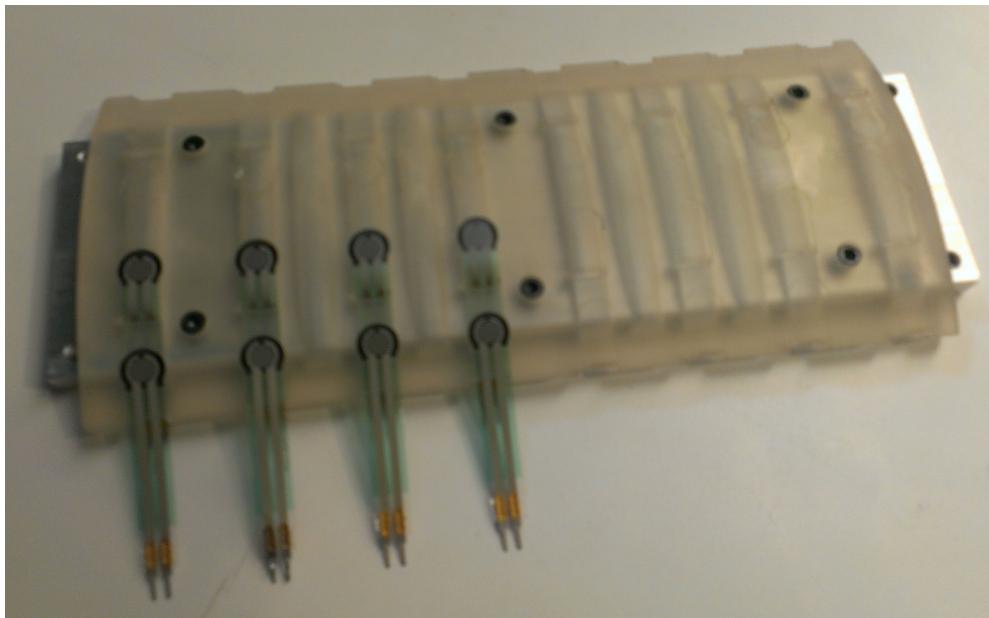


Figure 3 - Sample section during assembly, showing trace routing.

An axle running through a bearing mounted to each side plate supports the wheel itself. Each section of the wheel is attached to this side plate with machine screws. The bearing is held in place with a laser-cut acrylic cap that is bolted to the side plate.

Electronics

The Microchip PIC16F690 was chosen to provide on-board sensor reading and communication. This inexpensive model has the prerequisite number of analog channels, combined with a EUART line, and a fast analog-to-digital (A/D) converter. Also, the availability of this processor in both surface-mount and through-hole packages allows for quick and simple prototyping as well as future size and weight optimization without significant design changes.

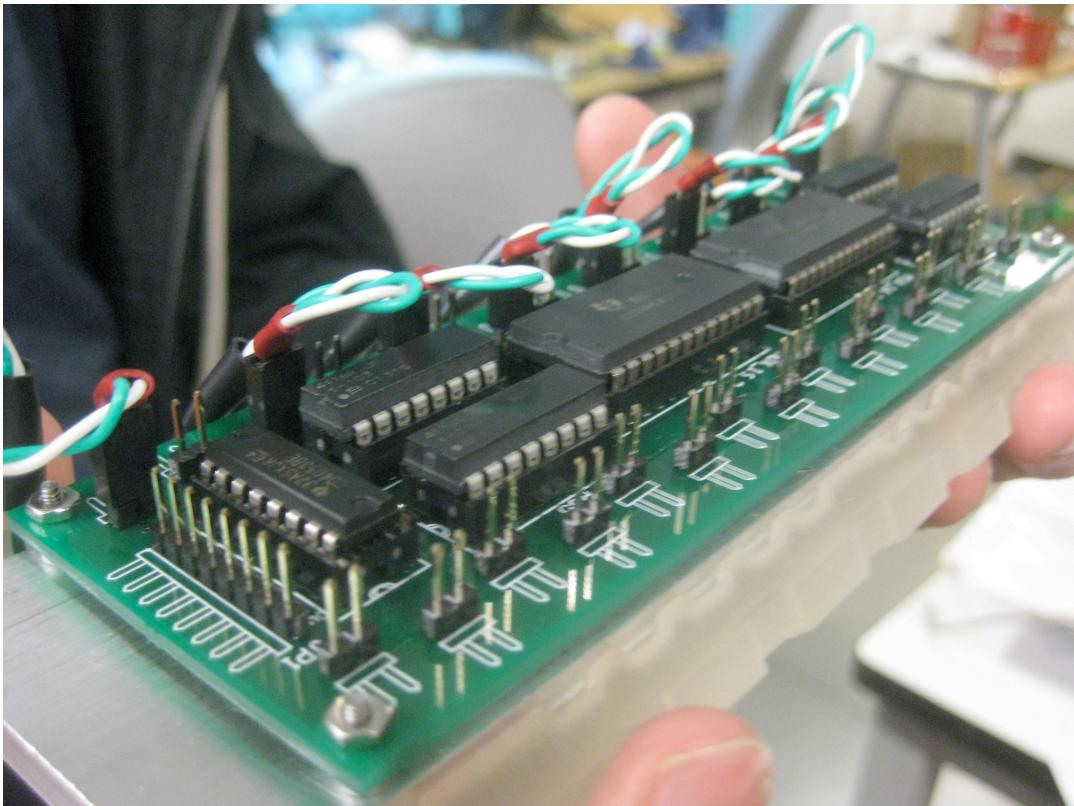
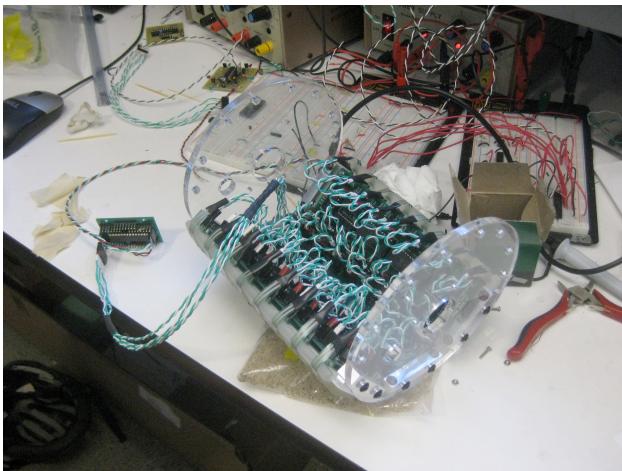
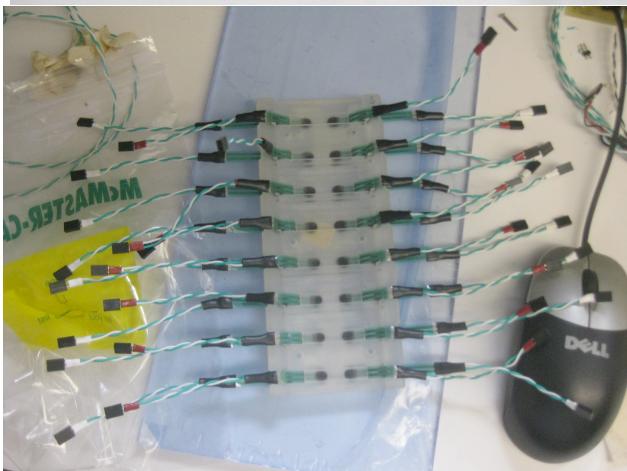
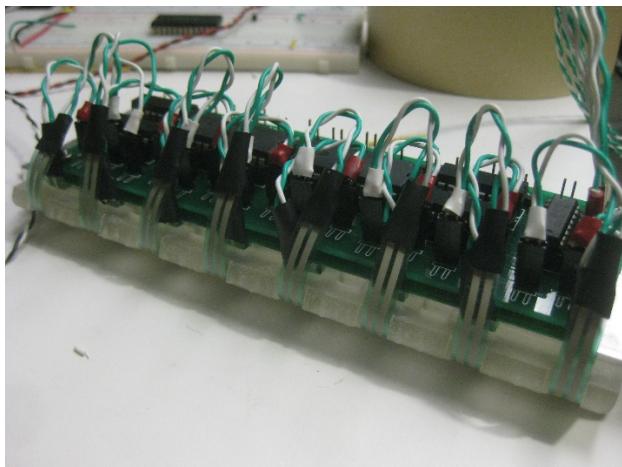


Figure 4 - Fully assembled PCB, showing (from left) interface connector, 2-channel MUX block, two resistor blocks, the two 16-channel MUX's, and two more resistor blocks.

In each of the seven sections, the 32 Interlink FSR-400 sensors are arranged into four rows of eight. To convert their force-varying resistance into an analog voltage reading, these are arranged as part of a voltage divider, each connecting on one side to +5v and on the other to a $10k\Omega$ resistor that was connected to Ground. The intermediate voltage of the divider was then fed into one of the inputs of either of two Texas Instruments HC4067E 16-channel multiplexers. The outputs of the multiplexers were then routed into a CD4053BE two-channel multiplexer. Therefore, five digital control lines were needed from the microcontroller to select any given sensor on a board. All sections were wired to the same five control lines, so that the same corresponding sensor was selected on each section at any time. The resulting analog output from each wheel section was connected to one of seven dedicated analog input pins on the microcontroller.



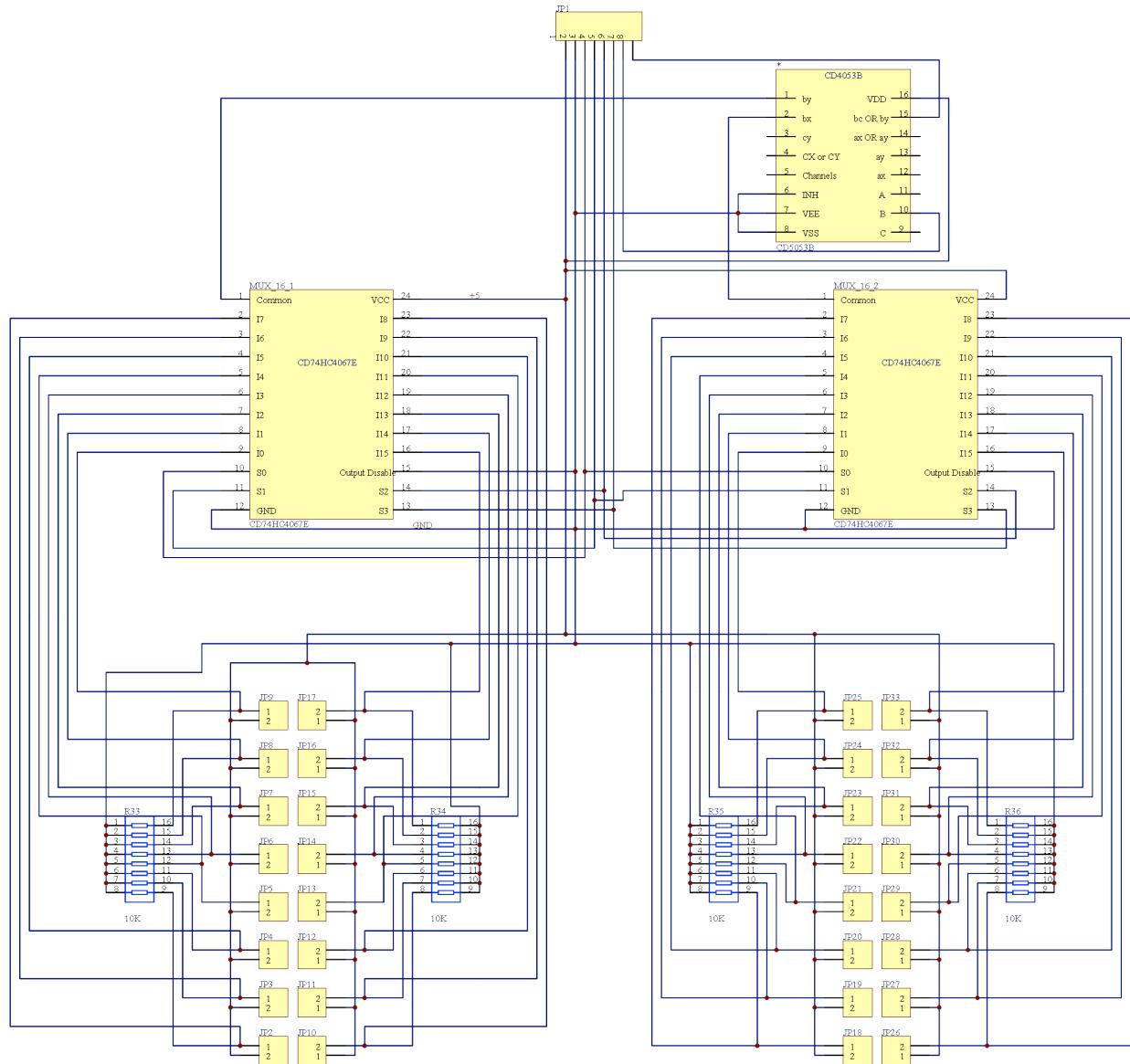


Figure 5 - Sensor board schematic

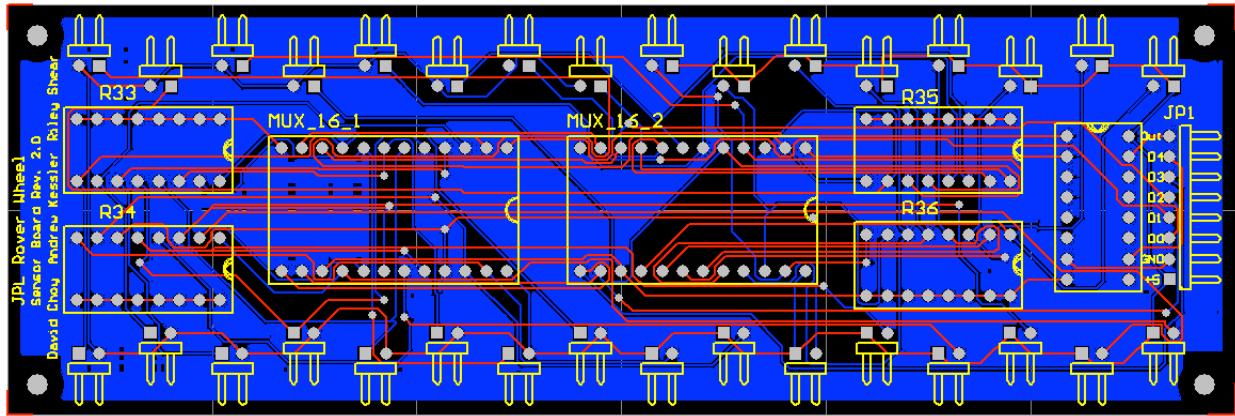


Figure 6 - sensor board PCB layout

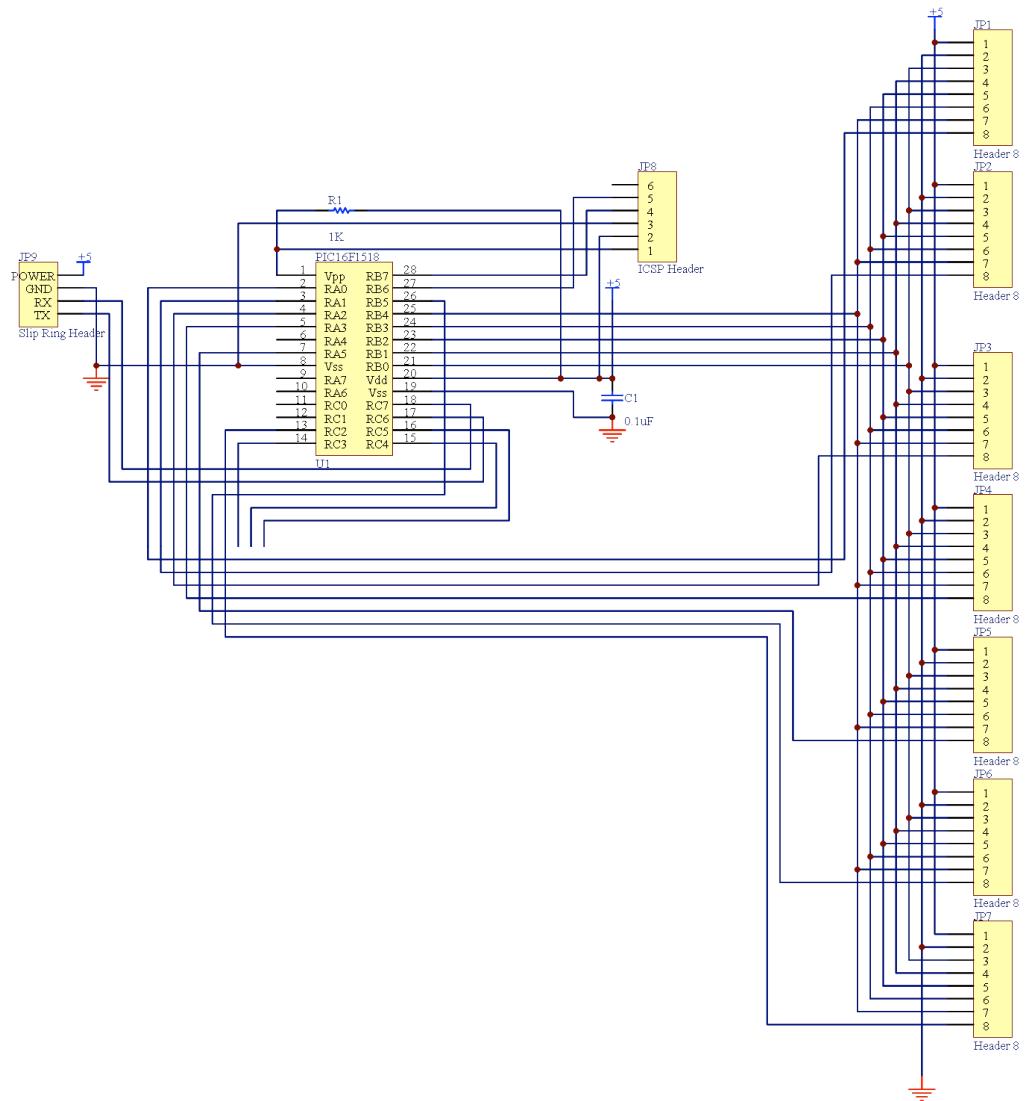


Figure 7 - Microcontroller board schematic

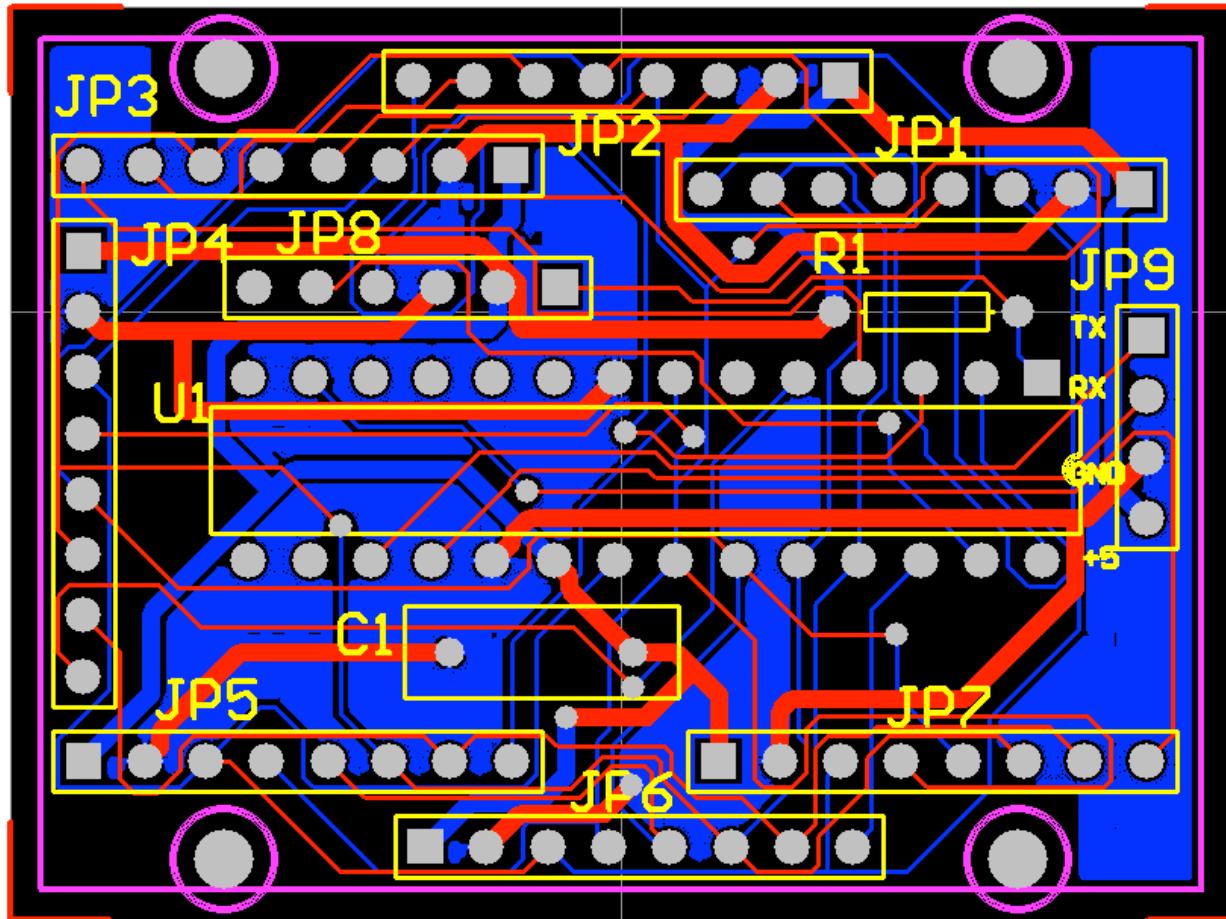


Figure 8 - Microcontroller PCB layout

To read all of the sensors, the five digital control lines were iterated through all 32 combinations of binary values. In each iteration, each of the seven analog input channels was captured and mapped to a digital value.

As the analog inputs are not read instantaneously, a finite delay time must be taken between successive measurements to ensure the proper sensor has been selected. Assuming an extreme 50 degrees Celsius temperature with $10\text{k}\Omega$ impedance, it is necessary to wait at least five microseconds for the analog acquisition time.

$$T_{acq} = 2\mu\text{s} - 10\text{pF} \left(1\text{k}\Omega + 7\text{k}\Omega + 10\text{k}\Omega \right) \ln \frac{1}{2047} + (50^\circ\text{C} - 25^\circ\text{C})0.05\mu\text{s}/^\circ\text{C} = 4.62\mu\text{s}$$

Assuming a worst-case scenario of 100 commands between analog readings, a 16MHz clock took 25 μs to get the next analog reading. The maximum rate for A/D conversion is therefore 40,000 samples per second. With only 224 sensors on the wheel, there is more than enough bandwidth to meet the 1Hz update specification. There is more overhead available to introduce more complicated calculations or increase the number of sensors placed on the wheel, if required.

If the transmission of each row is to be evenly spaced and still maintain a 1Hz update rate, each transmission needs to be sent every 35ms. Each transmission consists of the row number (0 – 27)

followed by the sensor values (two bytes each) sequentially in the same order as the sensors appear on the wheel. Each line is terminated with a new line character, which allows for easy parsing on the PC side.

The microcontroller outputs an asynchronous 5V signal over the EUART line. However, for simplicity, it was decided that the PC interface would be through USB. The interface is maintained by an FTDI FT232RL USB to Serial converter chip, available as a breakout board from SparkFun. Additionally, the chip was able to take power and ground from the USB connector and pass that through to the rest of the wheel circuitry, allowing our system to be powered completely off of USB. Communication and power lines are routed through the hollow axle and a concentrically-mounted slip ring, allowing continuous rotation while the sensor array is in operation.

PC Software

The PC graphical user interface (GUI) was written using the Java-based *Processing* IDE. The user interface contains a grid of squares, each mapping to a particular sensor on the wheel. The color of each box is linearly mapped from the force experienced on the associated sensor. The interface can also include overlays with the sensor number and the raw value being received at each sensor for debugging purposes. The lower portion of the interface features an averaging of each column of sensors corresponding to a row of sensors across the width of the wheel. This allows for a quick inspection of the data to see which part of the wheel is taking the force of the wheel. Finally, a drop-down menu was provided for selecting and connecting to a specific USB port. The ControlP5 library was used for buttons, drop-down menus, and other user-interface elements.

The interface design, along with the choice of *Processing* itself was done to give as much flexibility as possible to the user. For example, *Processing* is inherently cross-platform, the GUI can be run on any system that supports the Java runtime environment. The implementation of the GUI also provides for the ability to change the color mapping by swapping out a single configuration file.

In addition to the real-time interface, the software also includes a logging feature. The data received for each sensor are written to a file on the local PC in comma-separated values. The resulting file can be easily imported into data analysis software such as MATLAB or Excel for offline processing.

While the implementation of the GUI was relatively straightforward, some issues arose that are worth calling attention to. As mentioned in the previous section, the force readings were transmitted to the PC in the form of a single header byte indicating row number, sixteen data bytes representing the analog readings (two per each of eight sensors), then a single new-line byte. Because the analog values were transmitted as raw bytes, they could take on the value of numerical value corresponding to the ASCII newline character. Since the PC was scanning for this newline character, a check had to be added base on the amount of data received up to that point to determine whether the byte should be interpreted as a raw analog value or as the end of a line.

A second issue that arose involved transmission speeds from the microcontroller to the PC; if the microcontroller was sending readings over too quickly, the *Processing* serial buffer could overflow, leading to dropped bytes and invalid readings. Fortunately, the 35ms transmission period mentioned above for sufficiently long to avoid such issues.

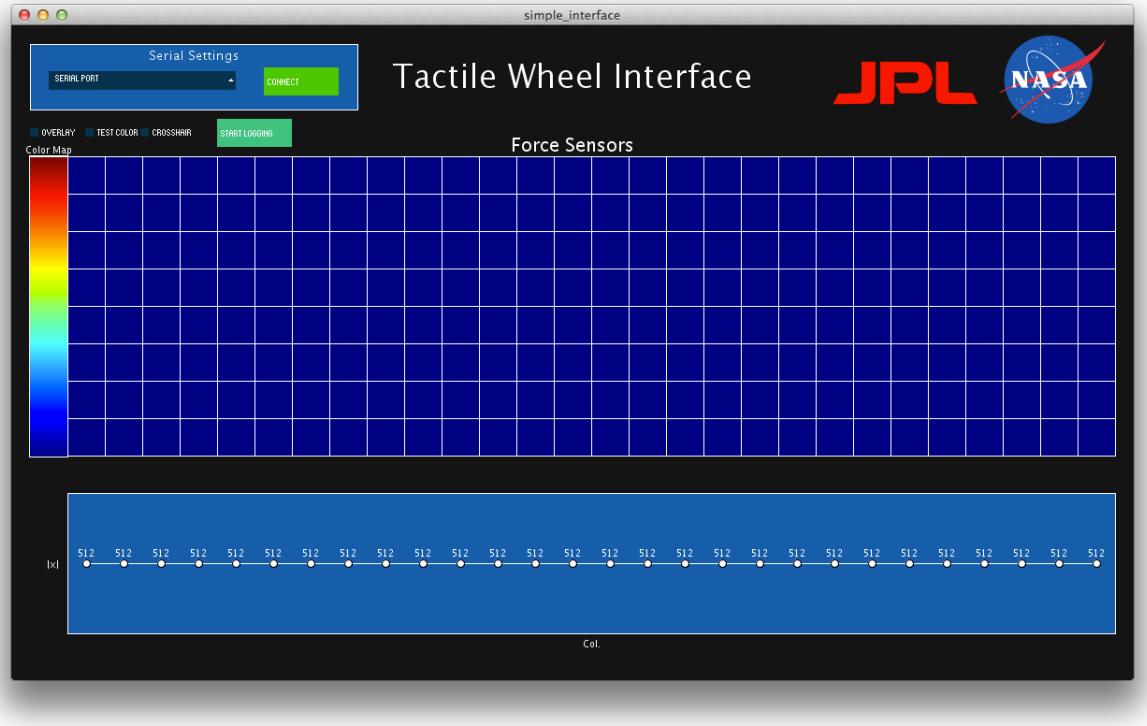


Figure 9: The GUI without the wheel connected.

Testing

Once the prototype system was fully assembled, we began preparations for testing its capabilities across multiple terrains. However, shortly into preparations, it was discovered that the adhesive holding the sensors together would not reliably withstand substantial tensile stress, such as the loading applied by the attachment method to the wheel's outer surface. The trace layer would physically delaminate from the pad, resulting in complete sensor failure. In future, custom-made sensors, a uniform external wrapping, or a redesign of the wheel geometry to reduce tensile stress might be able to eliminate this issue.

For finishing the proof of concept testing, the decision was made to use rubber bands individually wrapped around each circle of sensors to apply sufficient preloading force to prevent accidental delamination. An unwanted side effect of this approach is the introduction of unpredictable baseline forces applied non-uniformly to the entire sensor net. However, the effect is not catastrophic, and testing continued.

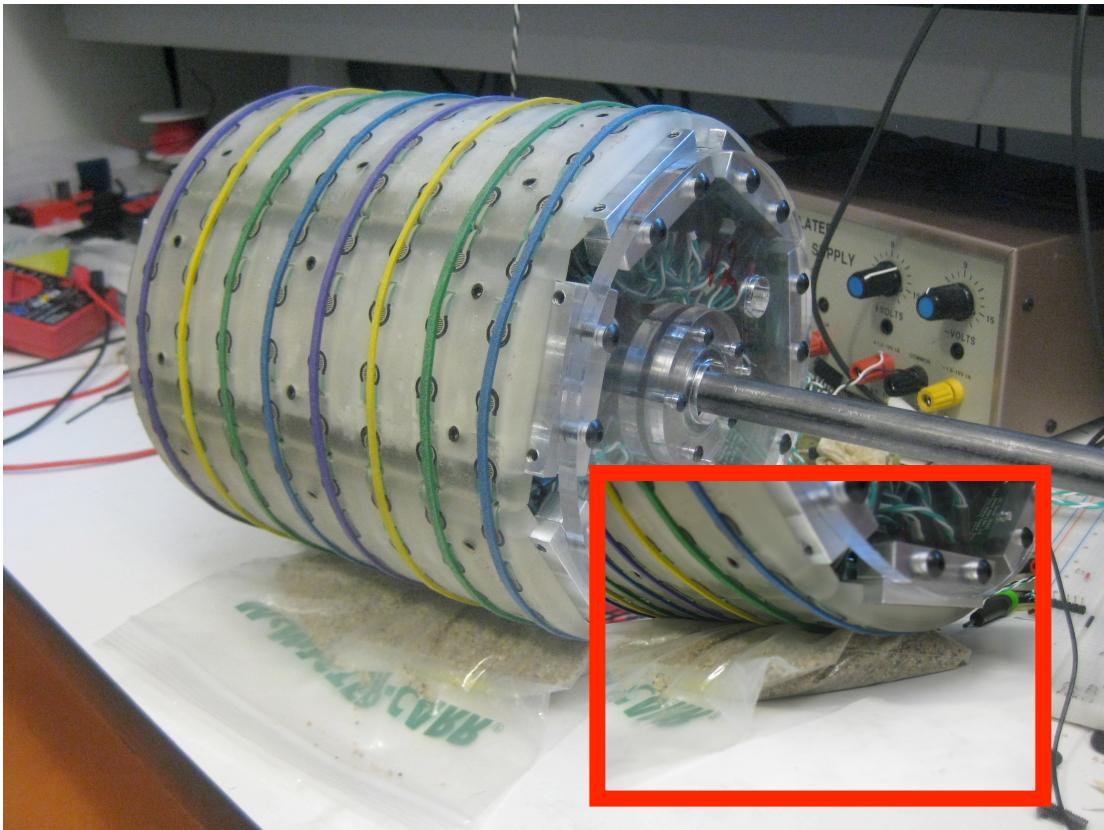


Figure 10 - Initial sand-bag test loading conditions

In order to test the wheel, a test rig was devised and built to allow for constant weight. A second “dumb” wheel composed of just an aluminum shell was created to balance the rig. Each wheel was then slid onto a hollow tube for the axle and snap rings were applied at the outer end of each tube to keep the wheels from sliding back off. These two tubes were then both welded to a steel plate to reduce the bending stress applied by the test rig on the tubes. Finally, this plate had a large steel handle welded to it for ease of movement. The high density of the steel used in the rig reduces the amount of additional weight required in order to reach specified testing levels.

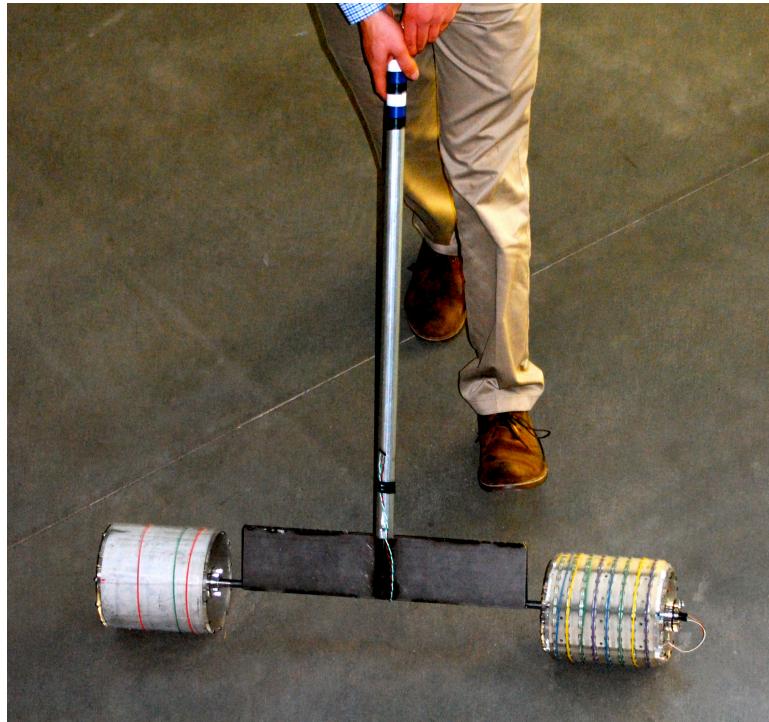
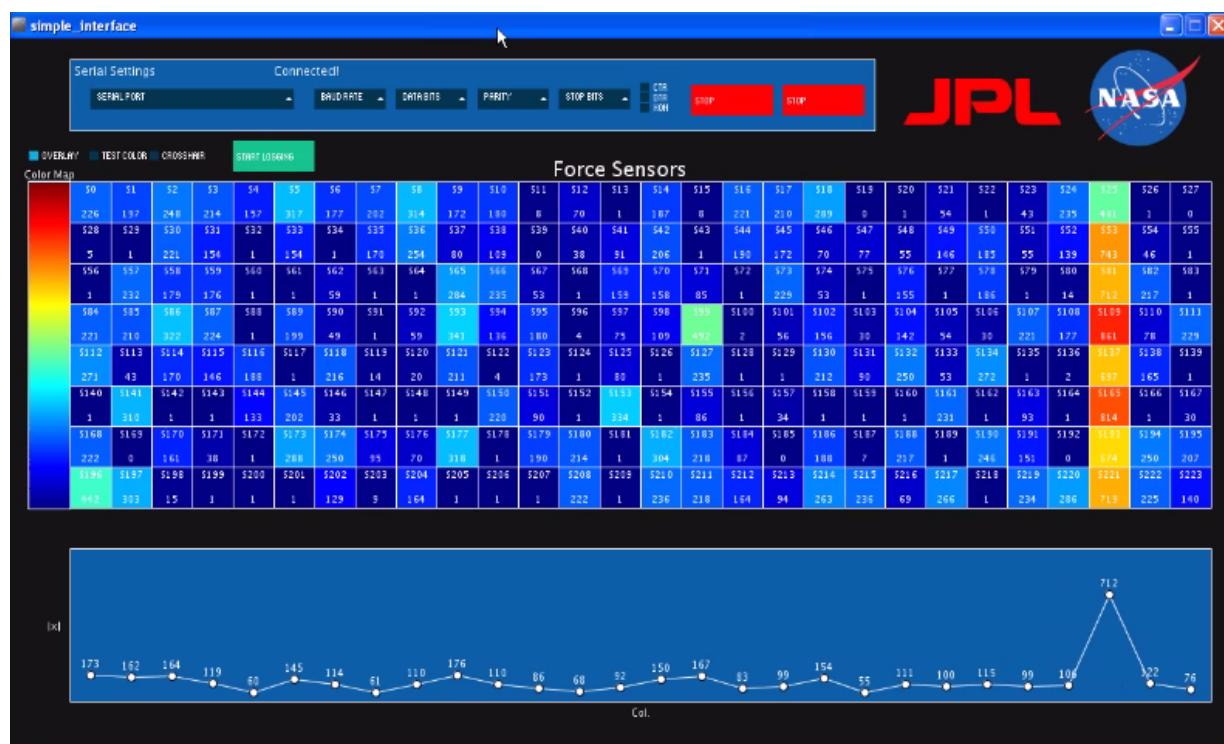
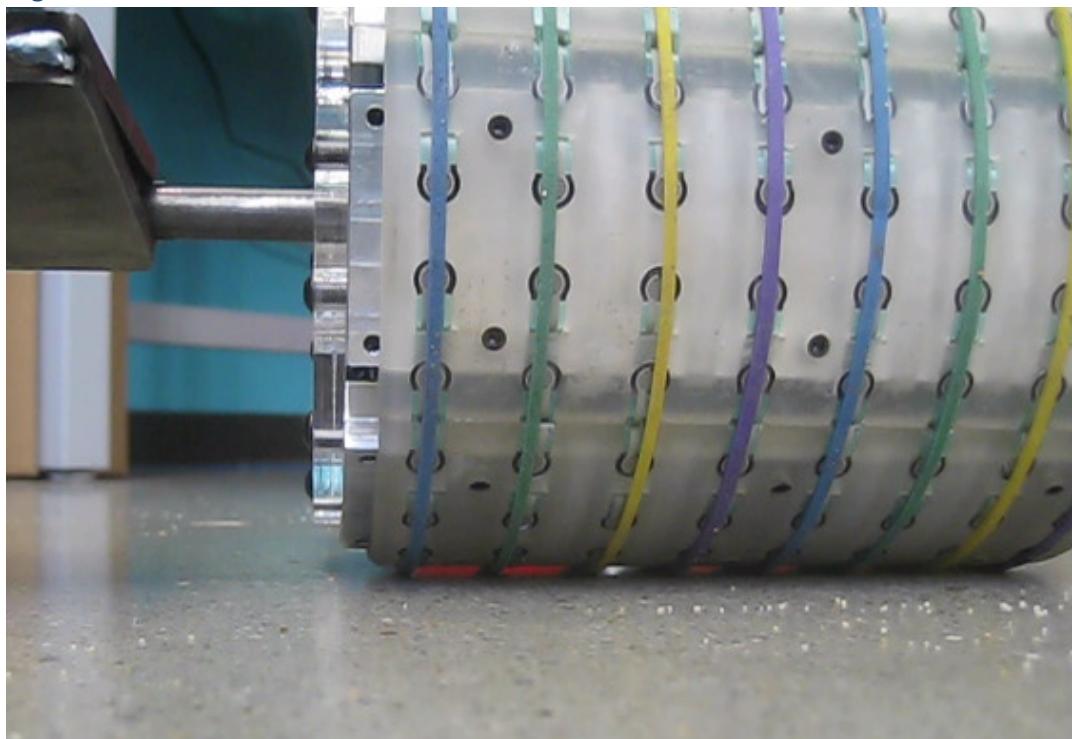


Figure 11: The assembled test rig; with “dummy” wheel on the left.

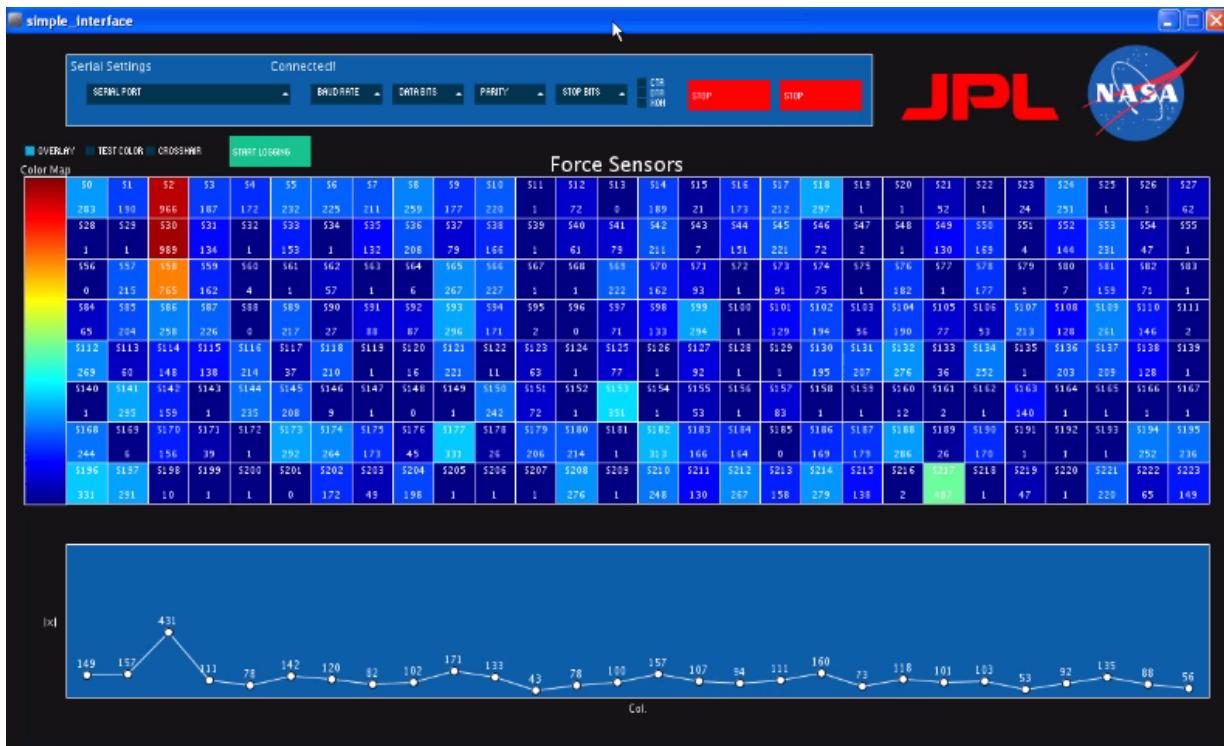
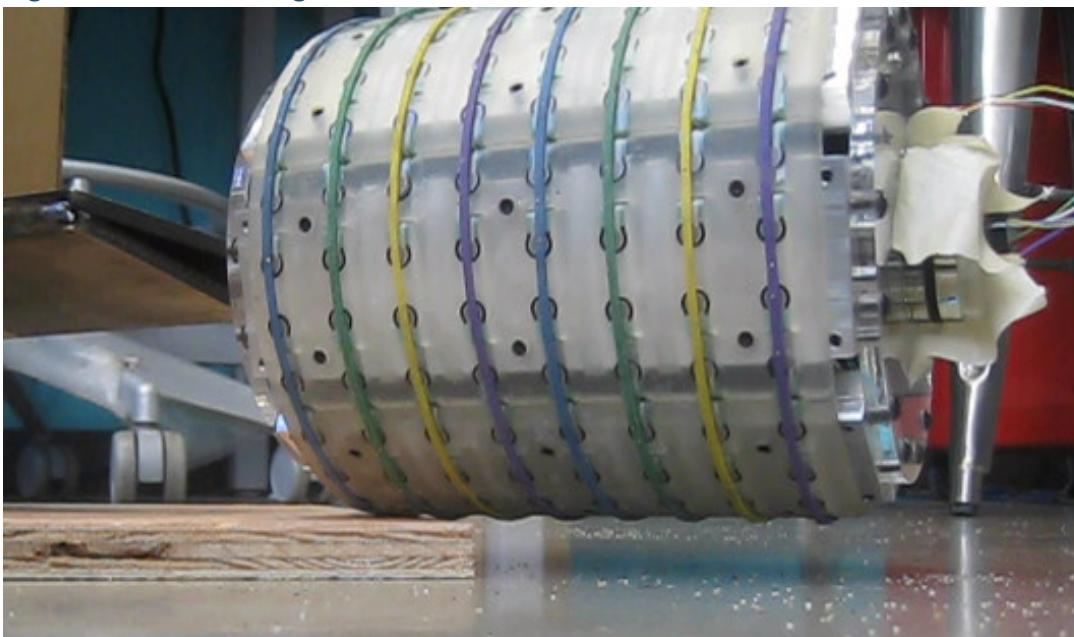
Ideally, the wheel sitting on a rigid surface should display equal values in the heat map for each row of sensors, with zero loading on all except for the row in contact with the surface. However, several factors affected the loading so that this was not seen in testing. First, the rubber bands are compressible so the amount of load taken by each sensor varies across the width of the wheel. Secondly, since imperfections in the welding operation resulted in the axle not being connected straight across to the second wheel, the expected load distribution was not uniform to begin with. Between these two major issues, as well as other smaller factors such as tolerances in the materials and irregularities in sensor readings based upon substrate geometry, the loading is not uniform and the exact force on each sensor could not be accurately determined. However, as demonstrated below, the relative and qualitative measurements provided by the sensor network allow for clear differentiation between different terrain types, fulfilling the original purpose of the project.

Rigid Surface



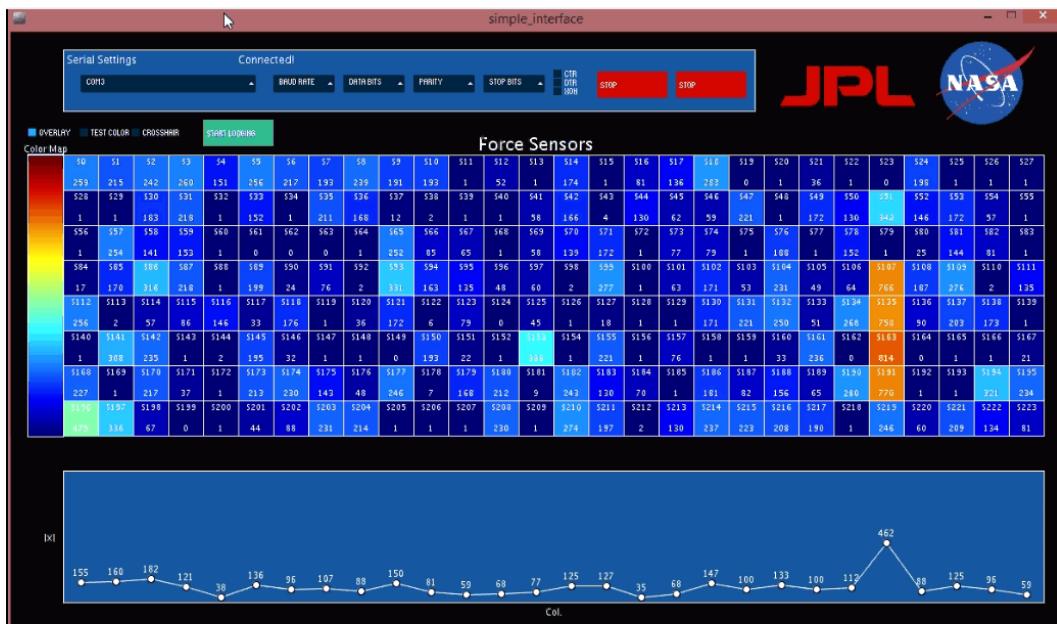
As expected, traversing a rigid surface produces a more or less uniform force distribution along a single column of sensors. The non-idealities of the system prevent a perfectly level voltage reading.

Rigid Surface Over a Rigid Obstacle



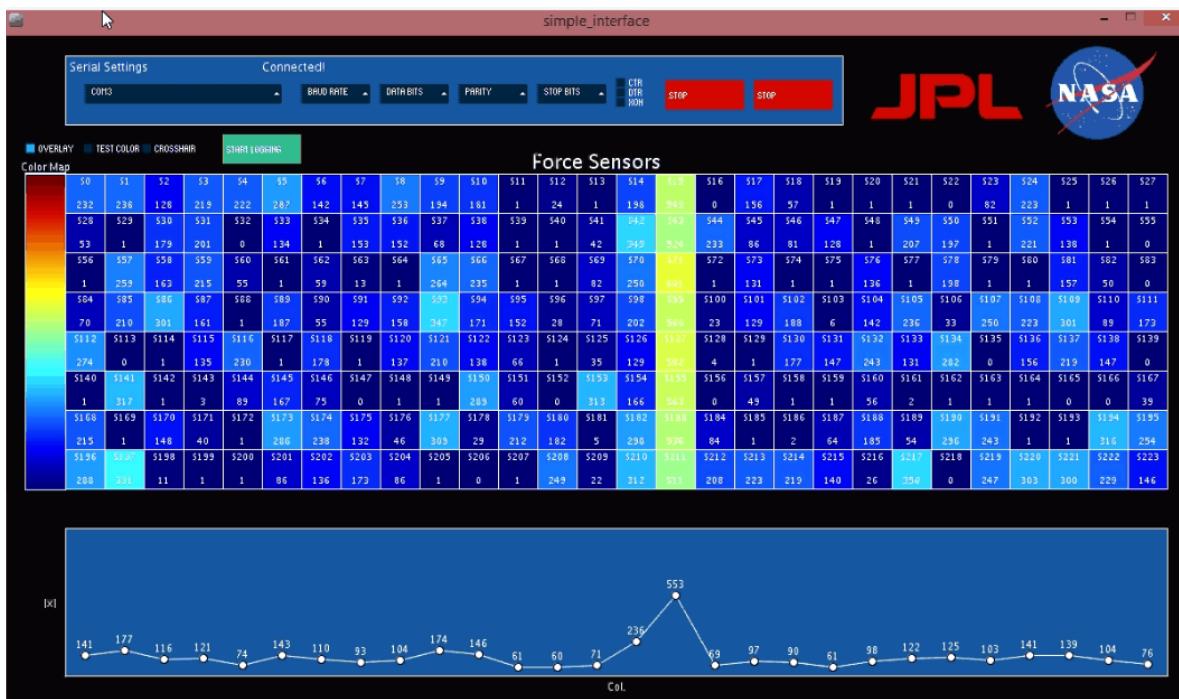
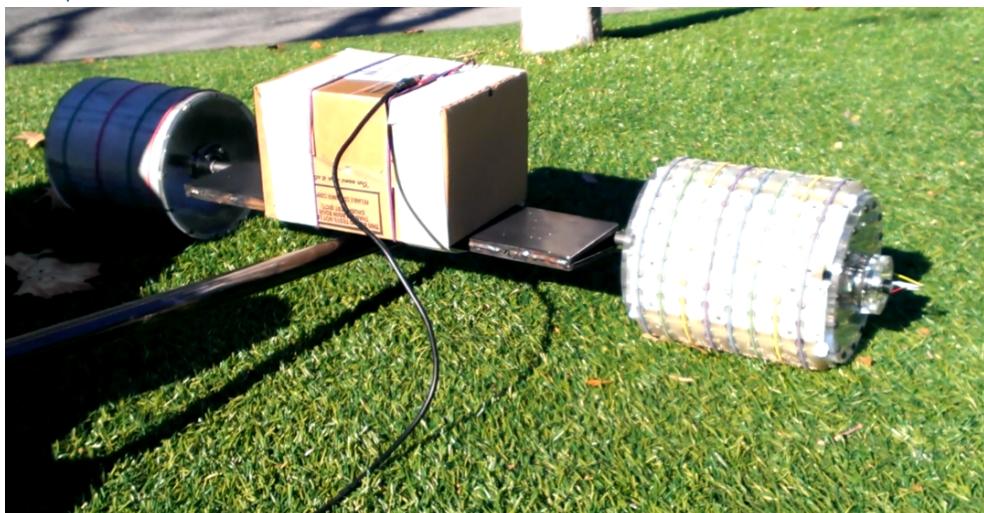
As the wheel traverses a flat, rigid obstacle, the full column changes into a smaller distribution with correspondingly higher force, as expected. The force values returned to the full column as soon as the wheel rolled off of the obstacle.

Rocky Surface



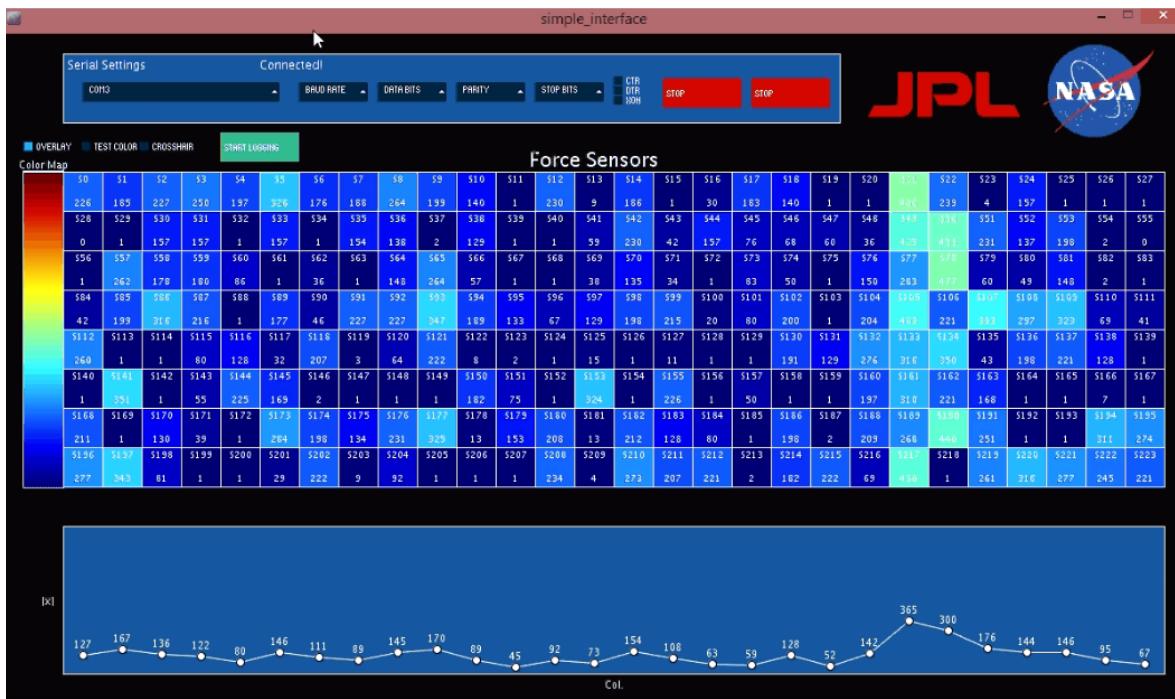
This test covered the traversal of an uneven, rocky surface, causing the measured weight distribution to vary widely, as would be expected. Most values are confined to a single column, but the distribution across the column is more akin to a series of point loads than full uniform planar surface contact.

Compliant Surface



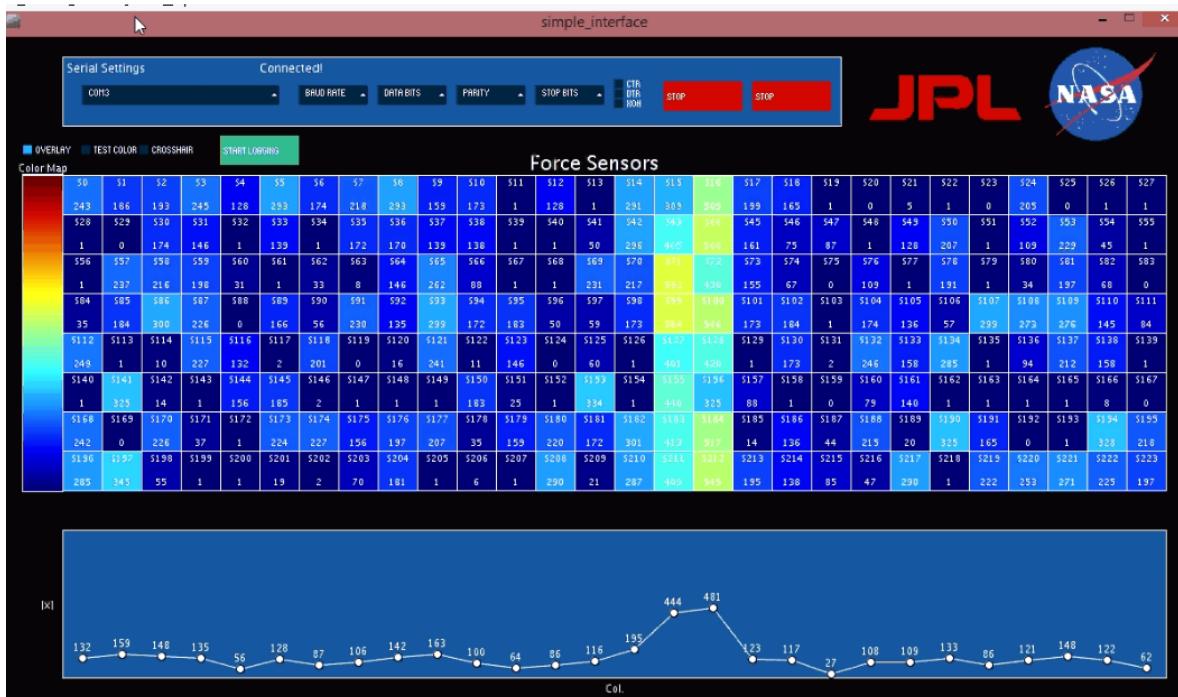
This surface is unlikely to be encountered on Mars, but recognition of similar properties could assist in later substrate identification. The uniformity of the sample indicates how the wheel responds to a compliant but contiguous material, such as the field turf shown here.

Dragging Through Sand



The wider contact patch results in substantially lower pressure measured when the same weight is applied. The rubber bands provided additional offset values, reducing the distinctiveness of the actual contact positions.

Forcing Weight Into Sand



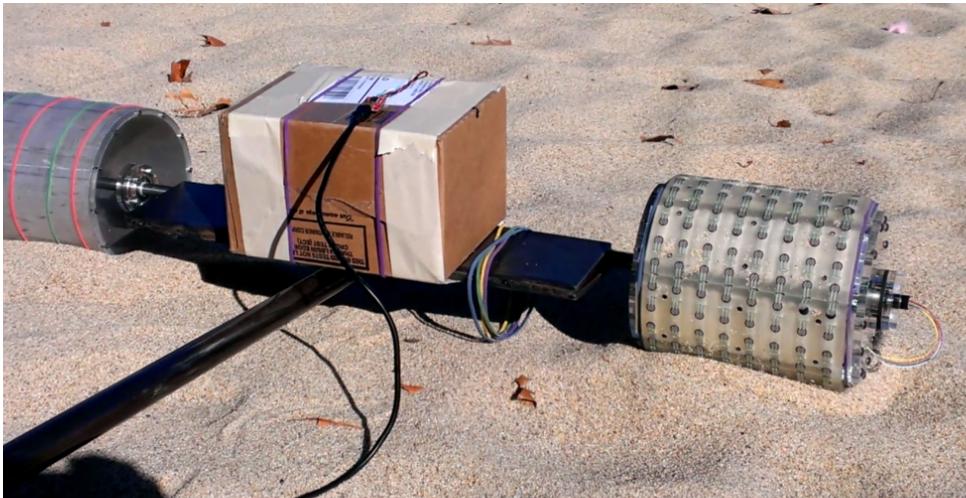
We attempted to increase the relative measurements on the contact patch by increasing the force manually. Results were as expected, with increased magnitudes in the contact area.

Dragging In Sand Over Buried Object



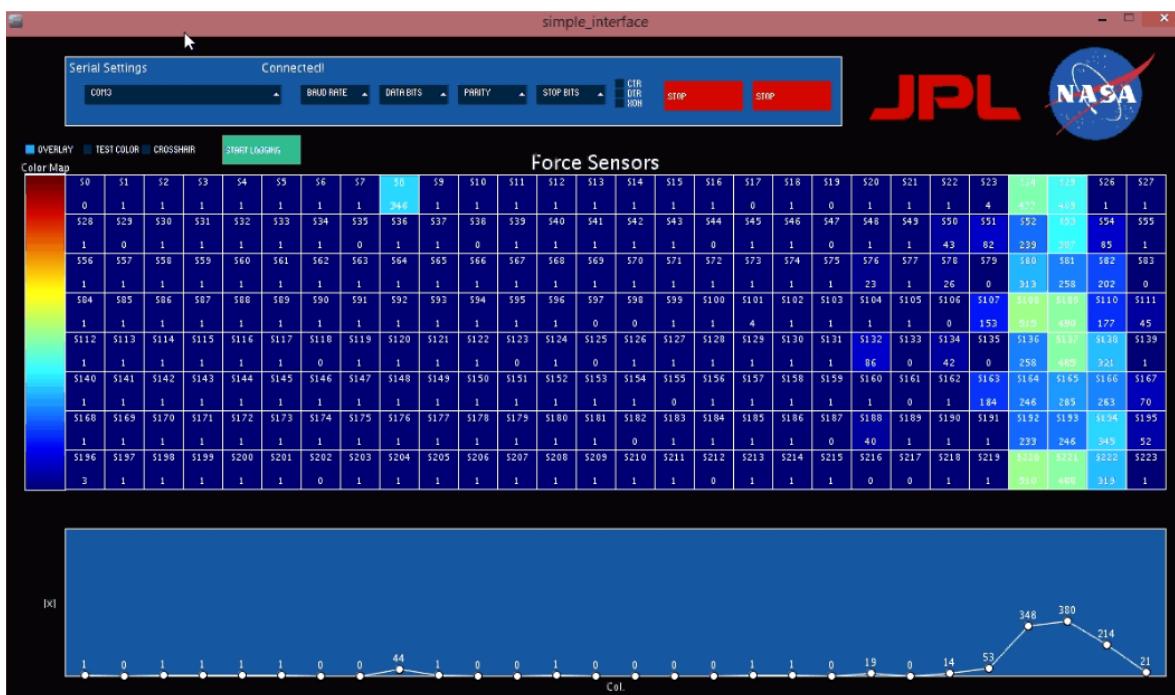
A (3.5 inch diameter) aluminum cylinder was buried in the test area, with only a small prominence above the surrounding area. Results were very promising, showing a sudden transition from the wide contact shown above to a small contact area similar to the earlier rigid-obstacle test. This would allow the detection of unseen sub-surface objects that might otherwise go undetected.

Dragging Through Sand with Rubber Bands Removed



The preloading from the rubber bands dramatically reduced the ability of the sensor net to detect the minimal pressure applied by sandy surfaces. Removing the rubber bands was risky, as it could have allowed rapid sensor delamination, but the possibility of improved data collection outweighed the detriments from losing several sensors after the other tests had been completed. Results shown above are improved dramatically from the earlier sample, demonstrating the potential of this technology if the delamination issue can be rectified.

Forcing Weight Into Sand With Rubber Bands Removed



The same manually increased force application was tested on the band-free model as had been done with the bands attached. Again, force magnitudes and area increased.

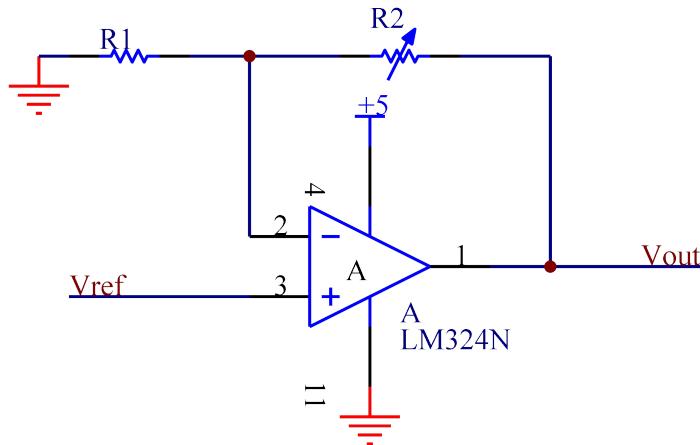
Dragging In Sand Over Buried Object With Rubber Bands Removed



The removal of the rubber bands also improved the detection of the submerged test object, with a clearer distinction on the GUI corresponding to the anomalous object.

Future Recommendations

Improve sensor output resolution: Using a voltage divider to measure the resistance output of the sensors resulted in a non-linear force measurement curve. This caused a more complicated translation from the analog output value into a force. In order to correct for the non-linear curve, the force sensitive resistor could have been connected to a constant current source circuit with an operational amplifier. R1 would be sized appropriately to give the desired range of forces sensed depending on the force/resistance range output of the force sensitive resistor.



Increase sensor coverage: Additionally, the force sensitive resistors only cover about 5% of the surface area of the outside of the wheel, resulting in the possibility of undetected contact measurements on non-deformable surfaces. This could be corrected by either increasing the density of the resistors, or increasing the size of the active area of the resistors. The force sensitive resistors used were chosen for their low cost and ease of use, but other commercial and custom options could be considered for another prototype iteration.

$$\text{Coverage} = \text{Active area of sensor} * \text{Number of sensors} / \text{Area of wheel contact surface} =$$

$$\pi \times 0.1^2 \times \frac{224}{\pi \times 6.5^2}$$

$$V_{\text{out}} = V_{\text{ref}} + V_{\text{ref}} \frac{R_2}{R_1}$$

$$V_{\text{out}} \in [V_{\text{ref}}, 5\text{v}]$$

Improve sensor attachment method: One of the major hardware problems encountered in this prototype was the propensity for the sensors to delaminate under the tensile stress to which the current design subjects them. An alternative could involve securing individual rigid pads to the outside of the wheel, so that each sensor is still independently actuated but is no longer in tension. This would solve both the problem of adhering the sensors to the outer surface of the wheel and the delamination of the sensor layers.

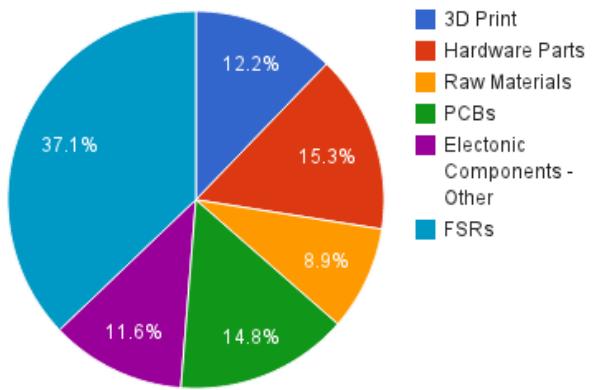
Bill of Materials

Part	Quantity	Units	Company	Part Number	Unit Cost	Total Cost
28-position DIP socket	1	ea	Digikey	A100210-ND	0.41	0.41
PIC16F1518	1	ea	Digikey	PIC16F1518-I/SP-N	1.94	1.94
Male header pins	574	ea	Digikey	A26509-40-ND	0.04	23.48
40-position female headers (x10)	1	pk	Fry's	7726788	11.99	11.99
1k resistor	1	ea	SPDL		0.01	0.01
.1uF capacitor	1	ea	SPDL		0.10	0.10
16-position DIP socket	35	ea	Digikey	A100206-ND	0.19	6.80
24-position DIP socket	14	ea	Digikey	A24798-ND	2.47	34.58
16-selection multiplexer	14	ea	Digikey	296-33087-5-ND	2.10	29.40
2-selection multiplexer	7	ea	Digikey	296-2059-5-ND	0.53	3.71
10k 8-resistor array	14	ea	Digikey	4116R-1-103LF-ND	0.47	6.55
2-position female headers	224	ea	Digikey	S7000-ND	0.19	43.46
White heatshrink	56	in	McMaster-Carr	7132K721	0.09	4.78
Red heatshrink	56	in	McMaster-Carr	7132K722	0.09	4.78
Black heatshrink	224	in	McMaster-Carr	7132K616	0.06	13.57
Thin white heatshrink	24	in	McMaster-Carr	7132K2	0.05	1.27
Force Sensitive Resistors	224	ea	Digikey	1027-1000-ND	3.40	761.60
Aluminum bar (3/8"x3/8")	14	in	McMaster-Carr	9008K21	0.11	1.58
Microcontroller PCB	1	ea	AP Circuits		22.13	22.13
Section PCB	7	ea	AP Circuits		36.58	256.06
Acrylic	316.172	sq in	TAP Plastics	Acrylic Sheet - Clea	0.06	19.80
Sealed bearings	4	ea	McMaster-Carr	6384K49	9.51	38.04
Duron	16	sq in	Room 36		0.01	0.11
2-56 screws	32	ea	McMaster-Carr	91735A019	0.12	3.68
2-56 nuts	32	ea	McMaster-Carr	91841A003	0.03	0.81
4-40 screws	10	ea	McMaster-Carr	91255A116	0.24	2.36
4-40 nuts	10	ea	McMaster-Carr	90480A005	0.01	0.08
8-32 screws	28	ea	McMaster-Carr	97763A178	0.10	2.77
Twisted pair wire	89	ft	SPDL	24 gauge	0.10	8.90
Slip ring	1	ea	Adafruit	1195	17.50	17.50
Steel tube handle	3	ft	McMaster-Carr	89955K801	16.04	48.13
Steel reinforcement bar (1/4"x4")	3	ft	McMaster-Carr	8910K584	11.50	34.51
Aluminum plate (1/4"x2")	49	lin in	McMaster-Carr	8975K26	0.31	15.43
Aluminum wheel tube	7	lin in	Alan Steel			10.00
3D printed sections	7	ea	Room 36		44.71	312.97
Rubber bands	8	ea	Staples		0.22	1.76
FTDI chip	1	ea	Sparkfun	DEV-09716	14.95	14.95
USB cable	1	ea	Fry's		5.99	5.99
Wheel axle	3	ft	McMaster-Carr	89955K779	8.97	26.90

Total \$1,792.89

Budget

Budget Allocation



Appendices

Appendix A: Code

All code is available as a Git repository at <https://github.com/apkessler/ME218D-JPL>.

The source code for the PC interface GUI is in the /pc_interface/simple_interface/ subdirectory. Precompiled versions of the GUI for Windows, Mac, and Linux are also available in this folder.

The source code for the microcontroller is in the /MicrocontrollerCode/ subdirectory.

Appendix B: Hardware CAD Files

CAD files are included as a separate ZIP file sent with this report.