NOTE: Obviously we won’t put all of this crap on the final poster, these are just some of the categories from past posters that are hanging in the hall.

**Problem/Opportunity:**

Innovation has increased exponentially over the past several years for the professional and the hobbyist alike. When 3D printing was first designed it was done by machines that cost hundreds of thousands of dollars. Today, 3D printers are cheap enough that almost anyone who wants one can afford one. However, these 3D printers deposit only a plastic filament. While still impressive, plastic isn’t acceptable for all designs. For this type of scenario we can turn to 3D printers that print in 3D.

Similar to plastic 3D printers of years past, metal 3D printers are currently very complex and expensive. While plastic printers melt a plastic filament and deposit it in a very thin layer, the most common form to 3D print in metal is via Direct Metal Laser Sintering, or DMLS. In DMLS, a thin layer of powder is deposited then a high power laser traces the desired path and fuses the powder into a solid by melting it locally. This process is extremely accurate, although also extremely expensive.

The purpose of our project is to create a cheaper 3D metal printer following the design of plastic 3D printers. We will combine a CNC machine with a MIG (metal inert gas) welder and use the welder to deposit wire. Building upon previous layers of deposition, we will be able to print 3D metal objects. The final products the printer will be producing will be internal components of industrial water pumps. Because of their use, all parts will be precision honed, so highly accurate prints are not a concern for this project.

**Objective:**

The key objective of this project is to create 3D metal printer by combining a CNC machine with a MIG welder. Because this method of 3D printing will not be very precise, the use of this printer will be aimed at a specific purpose: printing internal components of industrial water pumps. These components are currently made out of cast iron and wear out over time. Currently, to build a component, a wooden sample of the component must be created with great precision. A mold is then formed around the wooden version of the component. After the wooden version is removed from the mold, molten metal is poured into the mold, forming a new component. This freshly formed piece of metal then must by precision machined to within thousandths of an inch to properly fit. Including wait times, this process can take up to four months from start to finish and can cost several thousand dollars.

Our goal is to drastically cut both the lead time and the cost of manufacturing these components. Because the pieces are honed after completion, printing with extreme precision is not a concern for this project. Rather, the main concern here will be that there is no pitting throughout the print. In other words, we don’t want any air pockets in the metal. If this happens and one of these pockets is partially exposed after the honing process, the new component will fail quickly. By being able to avoid this possible flaw, our 3D metal printer will be able to print complete components with 2-3 days at around a fourth of the cost of the current method. Our focus on this project will be more a proof of concept rather than a final product.

**Key Observations and Learning:**

As there were many different aspects of this project, there was a lot to be learned. As is the case in all engineering projects, we had to make decisions on different tradeoffs, starting as early as deciding on what type of controller we wanted to use. There were also many things we learned that were unique to this project. For starters, we learned a lot about G-code, which is the language used to interface with the CNC. A lot was also learned around all the software involved with the project. Learning how to link to and use dlls was a new experience for most of us. We also learned a fair amount about welding; the difference between cold welds and hot welds and how different currents, wire speeds, and movement speeds affect the quality of the weld.

**Design:**

Although we weren’t concerned with an extremely accurate print, we still wanted the final product to be as accurate as the system would allow. This meant that we needed to monitor and control the quality of the weld in real time to be able to adjust settings during the deposition process. Our main source of feedback was the spacing of the droplets occurring during the weld. When using a MIG welder, a wire is fed out the end of the welding nozzle. When this wire comes into contact with a metal plate that is also connected to the ground terminal of the welder, the circuit is completed and the high current melts the wire. The current is so high, that the wire melts past the point of contact, which then creates an open circuit, until the wire feed (which is continuous as long) again reaches the ground plane and the circuit is again completed. Looking at this process closely, it can be seen that the deposition of the wire isn’t continuous, but rather a bunch of tiny droplets. We call this droplet spacing, and used it extensively throughout the remainder of our project. Through testing we determined that the average droplet spacing for a good weld was around 50 milliseconds (although this would change if the input current were to change drastically). Our control program looks at half a second time period, determines the average droplet spacing, and makes corrections to either increase or decrease wire speed accordingly. To know how to correct the weld, we ran a series of tests and created a lookup table of what we determined to be an ideal weld.

**Challenges:**

The main challenge faced in this project was consistently getting an ideal weld. There were four variables for each weld we created: surface temperature, current through the weld, wire feed speed, and CNC movement speed. An early decision we made was to try and keep the current and CNC movement speed fixed for each unique weld as much as possible. This meant that we only had to account for temperature and wire feed speed. Since we don’t have control over the temperature of the weld, we observe the temperature of the weld and adjust the wire feed speed accordingly.

To know how to adjust the wire feed speed, we ran a series of tests at various current settings. For each test, we wrote a G-code program to step through various CNC movement speeds as the welder deposits a one inch line. After running through all CNC movement speeds, we began a new test next to the previous and would only adjust the wire feed speed. After running these tests for a multitude of settings we were able to collect specific data for each weld, and by visual inspection, we decided which welds looked most ideal. We then combined our data with our ideal welds to be able to program the control board accordingly. The one variable out of our control was temperature. The temperature of the baseplate affects the weld in such a manner that we can use a lower current setting and still get an acceptable weld. To account for temperature, we simply polled the temperature sensor to ensure that the material being deposited on is above a predetermined temperature threshold. If, for any reason, the temperature drops below the threshold, the entire system is paused while the base is heated up using a torch. Currently, the torch is man-operated, although in the future the goal will be to have the system be automatically reheated when needed.

**Results:**

While this project was for all intents and purposes a proof of concept, as a group we would have liked to have implemented all the sensors and feedback controls discussed. We accomplished the proof of concept and implemented a basic feedback loop which controls the wire feed speed. Other options, such as the starting current for the welder as well as the torch routine to warm the baseplate to the proper temperature, are completed manually. The control program does include error checks, such as the welder running out of wire or the temperature falling below the desired threshold, but the whole system is not yet in place to correct these errors and guard against them.

**Conclusion:**

**Future Research:**