First of all, this week started off well when we finally received the micro Adson 1x2 forceps that we had ordered. Unfortunately, only one of the two that we asked for had been ordered, but we decided to test them before ordering any more. On Friday, I practiced removing teeth from some dead mice and found that the new forceps do in fact work much better than the bigger forceps that we had ordered before, so we ordered two more pairs. This will give us enough tools to allow Matt, Sarah, and I to all remove teeth at the same time. This will be very helpful when it finally comes to starting our experiments.

In regards to the experiments, they will have to be pushed back once again, until we can have the training that we need for the Office of Animal Research to give us access to the facility. We contacted Jason Farnsworth, who will be giving us the class on animal handling, on Friday, and we will hopefully be taking the class this afternoon. Once we do finally have access, we will have to take a few days to remove the teeth from the test subjects and will not be able to start the experiment until after the mice have had a few days to heal from the surgery.

On Friday, I also looked into how to program a microprocessor in order to add it to our circuit. I did some research and looked into Pulse Wave Modulation (PWM), which is how we would be controlling the power of the signal sent to the ultrasound transducer. PWM works by using square waves to control the average voltage under a sine wave. By changing the duty cycle of the square wave, it can be used to allow only so much of the original signal through (Fig. 1). When the duty cycle is set to 50%, the output voltage is half of the input. When it is set
to 25%, the output is one quarter of the input. This allows the voltage going to the amplifier to be exactly what we want it to be.

Figure 1: Pulse Wave Modulation works by changing the duty cycle in order to obtain the desired output frequency.

After trying to figure out how microprocessors work for a few hours, I realized that I needed some help from someone who has worked with these before and would know a lot more about it than myself. I found out that Dr. Kotha had just hired Jim Macione, a graduate ECE student, so I went to him for some help. As it turned out, Sterling had already shown Jim our circuit and they had come to the conclusion that it was in fact not going to work. I sat down with Jim so that he could go over these problems with me and explain what I would need to look into if I wanted to design a better, working circuit.

The first problem was actually the protoboard that the circuit was built on. Jim told me that protoboards are not made to handle frequencies higher than around 100 kHz without error. Our design calls for at least 1MHz, meaning that we would need to find a board that we can solder our circuits too instead.

Another problem is that the circuit we were using was not designed correctly. The RLC parallel circuit design caused many different errors. Also, the MOSFET was not set up correctly.
to switch the circuits. The MOSFET gait voltage was equal to the source voltage, which results in the switch not functioning.

Jim told me that in order to make a proper circuit design I would have to look into half bridge MOSFET set-ups (Fig. 2) and Class D audio amplifiers (Fig. 3). The half-bridge allows the transducer to be fed with the correct input at the correct time and the amplifier increases the voltage correctly to allow for the accurate voltage.

Figure 2: A Half-Bridge MOSFET design that uses the switches to control the input to the transducer.

Figure 3: A Class D audio amplifier that takes an input sine wave and outputs an amplified signal.