Interactive Punching Bag

Written by Lior
Wednesday, 23 February 2011 03:15 - Last Updated Monday, 07 March 2011 02:09

{youtube}ADbubYBdMqQ{/youtube}

For a while now I have been studying Karate and wanted to have a sparring buddy, without needing an actual body. Originally I was thinking of some robot arm that will punch and using either the Chronos watch from Texas Instruments or the Kinect as sensors. However, after messing around with the watch and the Kinect, I realized that they will not be sufficiently precise to sense some of the fast movement. After searching on the web for a while, I came a cross this interactive punching bag
, or the
Slam Man
. After watching some videos of the them, they both seemed kind of fragile, and I did not know where I could get one from the USA. Additionally, it will be nice to be able to modify the games as well. And how hard is it to place some LEDs inside a punching bag and sense some force using an Arduino?

Here is the the final result. This took me about half a day to accomplish using parts that I already have, and going to radio shack for the bright LED's (they only had 3 LED's, but I made room in the board for 6 sensors and 6 leds. I will pick some more in the next order to digikey). I also did not have a spare LCD, so for now I am using my Laptop for feedback. I will also pick up an LCD as some point, and remove the Laptop.

The bag has 3 modes, which you select in the beginning (all lights go on, and you punch on the light to select the mode). For now the LEDs are placed one in the left top, right top, and sternum. The first mode is to see how fast and how strong you punch the sternum. So you are given 30 seconds, and you are suppose to punch the sternum as fast as you can. The next mode is a sequence (that's what you see in the video). You can program a sequence of moves by punching the order that you want, and during the game, it will light an LED in the sequence and you are support to punch it out as soon as you can. The bag will measure how fast you reacted on average and how strongly you punched the bag. In the last game, a random LED will light and you are support to punch it. However, once in a while, either the top right or left LED will blink twice fast, which will indicate someone punching you. For this you need to react (move you head, block, get hit, whatever you want). I am not sensing anything at this point for this (maybe the Kinect could be used for this or I was going to try the Shard IR sensors for simplicity).

Prototype Theory
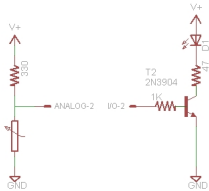
Sensor: Since the sensor is resistive, we need to convert it to voltage as a signal. This is done with a voltage divider. To figure out the resistor needed, I first measured the resistance of the sensor in its resting position, and then measured the resistance again by placing as much pressure as I could put on it. The values you get will probably depend on the sensor size, but for me I got about 4 M ohm at rest to 0.4K pressed. I wanted a voltage range between 5V and 2.5V (to account for stronger punches which push the sensor further then what I could simulate). Using the voltage divider formula, I cam up with a 300 ohm which will give it a voltage swing between 4.99 V and 2.86 V.

Here is the math

Vout = V1*R2/(R1 + R2) where R2 is the sensor and V is 5V.

Sensor not pressed: 5*4000000/(4000000+300) = 4.99V

Sensor Pressed: 5*400(400+300) = 2.86V

Lighting the LEDS: I got 3 (that the only quantity they had at that store) high brightness Red power LED from radio shack (Model: 276-020) . The LED should be red since the bag I was using is red, which means that it will filter all wavelengths of light except red. If you use a black bag, you might need a very powerful LED to show through the bag. Since the LED are rated from 2.6-3.3V at 70mA, the Arduino will not be able to draw that much current. Therefore, a transistor is needed. We will saturate the transistor, so it will be fully on with a 1K resistor. The second resistor is calculated using ohm's law (since the transistor can just be treated as a switch).

$V = IR$

However, since we want the voltage across the resistor, V will be Voltage of the source - Voltage drop of the LED.

$V_s - V_{led} = IR$

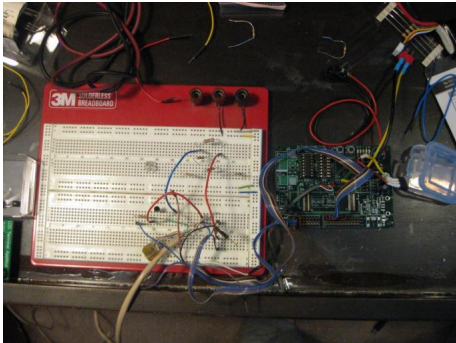$5V - 2.6V = 0.07A * R$

$R = (5-2.6)/0.07 = 34.3$ ohm

Since the smallest resistor I had in my possession is 47 ohm, I used that instead. This will give me a 51 mA draw at the LED, which should be sufficiently bright.

Written by Lior
Wednesday, 23 February 2011 03:15 - Last Updated Monday, 07 March 2011 02:09

I first prototyped everything using a proto-board to insure that the math is right.

Building The Prototype

BOM:

For Sensor: 6 X Resistors 330 +- 5%

For LED: Resistors 6 X 1K +- 5% and 6 X 47 +- 5% Transistor: 6 X 2N3904 (or any PNP with an Ic > 100 mA)

The prototype was built by using the components wires as traces. You can probably just follow the images to see how it is built.

{gallery rows=4}PunchingBag/ProtoBoard{/gallery}

Building The Sensors

The sensors are built using aluminum foil and conductive foam (I learned this technique from the Robot Builder Bonanza . I have the first version of this book, but I hope they kept it in this version as well). Cut two rectangles of aluminum (the size of the sensor), and solder two connectors to this. The soldering requires a very hot iron (I has mine on 850 degrees) and you need to move it back and forth scoring the aluminum a bit (be careful not to rip it) so you can get a good connection. It takes about 30 seconds for the aluminum to be sufficiently hot to accept the solder. You then tape two pieces of aluminum to each size of the conductive foam (I use the packaging tape with the string inside for added strength). Lastly, cut a square hole in the center of the sensor for the led. For the wires, I used a cat 5 cable that were striped about 3 feet. For the middle sensors, I don't cut them, but just strip in middle and should that directly.

{gallery rows=4}PunchingBag/Sensors{/gallery}

Software

Since I don't use the java gui with the Arduino (I use a make file, and a vi editor), you will need to take out modify the main function to be loop or something (I don't know since I have never used that gui).

PunchingBag.tar.gz