Documentation of tests

The purpose of this document is to record tests done the individual components of the Project Nerf Alpha.

The following components were tested, with performance and errors noted:

- Muzzle sensor emitter (IR emitter: LIR034)
- Muzzle sensor receiver (IR receiver: TRCT5000)
- Magazine sensor (Hall sensor: A3144)
- Display counter (7-segment display: 3461BS)
- Battery converter (Voltage converter: LTM8067)
- Buzzer (Buzzer: HYDZ)
- Jamdoor switch (Contact wires)

Muzzle sensor pair (IR emitter: LIR034, IR receiver: TRCT5000)

Several different sensor pairs were tried and tested here, including sensors from the “37-in-1-box-sensor-kit”. It was however discovered that those kinds of sensor pairs were used controlling televisions, while we simply needed to know when an object passed between the pair and interrupted the signal. The solution was to grab an IR LED, and a phototransistor covered in a film which blocks out all light except those from the infrared spectrum. After hooking these the pair up and writing a bit code to go with them, the result was satisfactory. A few minor adjustments were made to detect when a dart passed between the sensor was pair made and tested again. Performing to expectations now

Magazine sensor (Hall sensor: A3144)

Originally the magazine sensor was supposed to be created through avoidance sensors. These were however very unwieldy and difficult to put into a small package. These problems resulted in the usage of an alternative: a hall sensor. The sensor used here will activate when a magnet gets within 12-15 mm of it. This was tested with a variety of commonly available magnets.

The first few tests were unsuccessful. It simply did not work. It was then discovered that only the face if the sensor would register when an electronic magnetic field was present. The module in which the sensor was placed upon had the face of the sensor pointing downwards. After bending the sensor 90 degrees, all further attempts were successful.

Only minimal code is needed to make it work, since the sensor will send either a digital HIGH or LOW.

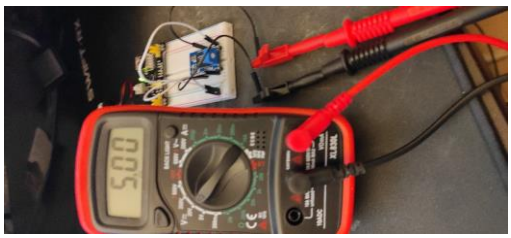
Display counter (4-serial 7-segment display: 3461BS)

The main challenges of trying to make a display counter was trying to figure out how the component worked. First, it was assumed that the component needed an integrated controller. It was however quickly discovered that this display already came with one.
Next, the difference between how common anode and common cathode works was discovered through trial and error. The original assumption was the component used a common anode, but when the component didn't behave as expected, the code used for it had to be rewritten to account for this mistake.
The next step was to figure out how bitmaps work, and how to include them in the code. At first, dictionaries were used, but turned out to be too unwieldy. Lists were used instead, as it was easier to simply use the index positions instead.

Another interesting point was sending command bits to the controller. With a sliver of trial and error, the command bits was successfully sent to the controller, but it was still a learning experience.

Voltage converter: LTM8067

This component was quite easy to work with. The only kind of testing done was after attaching the input wires with a 9V battery, then measuring the output while turning the variable resistor on the board until 5V and at least 200ma came out on the other side. No further tests needed.



Buzzer: HYDZ

When attempting to figure out what kind of buzzer to use, the two ones present in the freenove kit was chosen. One was a passive, the other an active. Simple tests were conducted: connect power (9V battery) and see what happens. The active buzzer immediately let out a sound, but nothing from the passive one. It was then discovered that passive buzzers can only provide a sound when powered by AC, while active buzzers have a built-in internal oscillator.

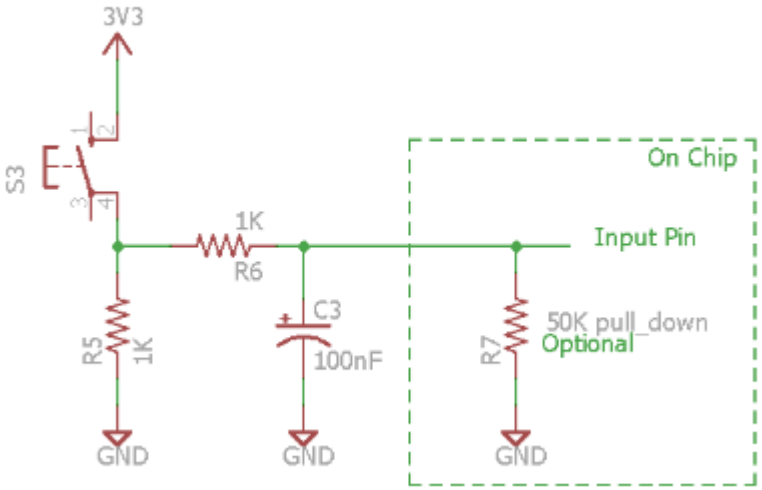
In the end, the active buzzer was chosen as that was much easier to work with. A few further tests using 3.3V and 5V supplied by the Raspberry Pi was also done, which very successful as well.

Jamdoor switch (Contact wires)

The nerfgun has a trapdoor on top of it to clear out jams. Opening the trapdoor prevents the gun from firing. A few wires has been attached to the door frame, with a conductive material attached to the corresponding area on the door itself. Fitting the wires properly was a little bit challenging, took a few tries, but finding the conductive material and checking it was easy.

Testing the functionality of the door itself was successfully, as it has simply been programmed as if it was a button or switch. This does however necessitate de-bouncing it. Two solutions were whipped up: one based on hardware, another based on software.

The hardware solution took both time and effort. It is a simply circuit involving two resistors and a capacitor to smoothen out the bounces of the jamdoor opening and closing using a small capacitor:



The solution worked if one didn't press and release the button too fast. It was an okay solution

The software solution works by simply adding "bouncetime = 250" to the event detection GPIO function. This works just as well.

In the end, the software solution, while just as good as the hardware one, saves money in the long run since no extra components are needed, which resulted in that being chosen.