

Final Report

Joshua Black

Table of Contents

Introduction	1
Methods.....	2
Assumptions & Procedures.....	3
Results & Discussion	3
Conclusion.....	4
References	4
Appendix	4

Introduction

The Nerf Turret project's goal was to develop a pan-tilt turret from scratch that could be connected to household network technology such as Wi-Fi and be controlled remotely. As a payload, the turret would need carry a nerf dart magazine and CO2 firing system to propel nerf darts. Such a system could be used as deterrent for porch thieves on residential homes due to the turrets capability to use regular Wi-Fi and carry a payload such as a pepper spray canister.

The project was chosen for various reasons. Firstly, for my senior design project I was charged with designing a control scheme for an auto-targeting turret system that utilized IR camera data to detect, track, and extinguish fires by actuating 2 turrets at the same time. The Nerf Turret project, although different in form, still implements the pan-tilt scheme. Unfortunately, during my senior design project I was not directly involved in the mechanical design of the turrets; even if I was, the most important components were commercially bought. I have always wanted to develop an actuator such as a pan tilt turret. The Mechatronics term project was a perfect opportunity to design, from scratch, a turret and learn all the considerations, limitations, and capabilities of a gear-based pan-tilt.

Secondly, a gear-based pan-tilt is unique in that all its individual components are relatively simple. This is important because a constraint on the project was for it to be completely 3-D printable. This was not only because it enabled rapid prototyping, but also because I have never had a reason to use 3-D printing and wanted to learn the skills required for it.

Lastly, given the moderate part count, ease of manufacturing, and rapid prototyping capability, the Nerf Turret project seemed feasible to be accomplished by a single person while also dealing with a large Senior Design load.

Methods

As mentioned above, constraining the project to 3-D printing all main structural elements means that a rapid prototyping approach could be utilized along with the General Engineering Design Process (Malmquist, Frede, Wikander 742) outlined in Holistic Design Methodology For Mechatronic Systems. The main systems that have been designed include the pan tilt gear system, nerf payload design, CO2 firing system, electrical system, and software/network design.

The pan-tilt gear system was designed around two spur gear interfaces driven by 125:1 Pololu gear motors. Using the motor gear ratio as well as a large final gear ratio means that the system is stable when unpowered. This is important due to the moments created by the nerf payload when fully loaded. The base plate of the turret includes mounts for the various motors and structural elements and is limited by the available build volume of the Endeavor Lab's 3-D printers.

When designing the nerf payload it became evident that creating a magazine from scratch would be far outside the scope of this project due to the amount of design and testing work it would require. To simplify the design, it was decided to use an off-the-shelf nerf magazine. An attachable mount was designed to securely hold the nerf clip at the proper orientation and height for the CO2 nozzle. On the mount bracket that attaches to the tilt axis, two pin holes are put at a parametrically adjustable location to abstract the concept of a mount for the payload. This means that for any given turret, it simply needs to have pin holes at a particular spot, and it can be attached to this turret without fastening conflicts; similar to how the NEMA sizing standards work for stepper motors.

To eject the nerf darts from the magazine, a CO2 high pressure gas firing system was designed to directly feed gas into the cavity of a nerf dart from a needle nozzle mounted on the nerf payload. The CO2 gas canister is settled in a 3-D printed mount that is pinned to the base plate. The CO2 canister valve is opened via a push button on its back that is depressed by a rotating cam similarly mounted on the base plate.

To operate the motors, an Arduino Nano 33 IOT was chosen to be the brain of the controller. The Arduino Nano sets voltages on 2 L293D H-Bridge controllers to provide switching and polarity control with all the components mounted on a breadboard. To provide power a typical USB smartphone power supply, mounted with Velcro, is used with the Micro-USB port on the Nano.

The Arduino Nano is equipped with a Wi-Fi network transceiver that can connect to WPA2 networks. This is very convenient because it means that the Nano can connect to smart phone hotspots natively, providing a perfect test environment. Once connected to a network the Nano provides a TCP network server that controlling clients can connect to and issue commands to actuate the motors with certain polarities. On the control client side, a simple terminal application was developed to automatically search for, and connect to, an active Nano. Through the application you can issue

commands to the turret through event loop inputs occurring on the host machine. Essentially, this means the turret is controllable through common input devices such as a mouse or keyboard.

Assumptions & Procedures

To operate the turret a few conditions must be met. Most pertinently is that the turret can only be controlled through an active TCP Wi-Fi connection. There is no interface on the turret that would enable manual control without adding hardware and reprogramming the turret. Furthermore, it is assumed that whatever computer you use to run the control station software is a 64-bit windows machine with at least one of its network interfaces connected to the same network as the turret. The software will automatically scan all active network interfaces so even if the turret network is connected on a non-primary network interface, it will still be found. Once connected, an operator can issue commands using the arrow keys on a connected keyboard. The software could be configured to use other types of interfaces as well such as a mouse if needed, however, the arrow keys provide an intuitive control scheme. To power the turret, simply connect the Micro-USB cable from the battery to the Arduino Nano. Unfortunately, the Nano must be pre-programmed to connect to a specific network SSID with a specific passkey, so that network must be present.

Results & Discussion

The Pan-Tilt gear system performed better than expected given the material it is made from. However, to ensure proper rigidity was achieved, super glue was added to the slots on the base plate where all the vertical components would be mounted. This included the vertical columns for the tilt axis, the CO2 holder, and the motor mounts. After properly curing, the components are super rigid and able to withstand the force of the CO2 canister expelling.

The nerf payload can adequately hold the nerf clip with just enough force to keep it from moving while also being easily removed. To hold the needle nozzle a simple screw clamp provides an easy attachment mechanism. Even with the tight tolerancing on the nerf payload mounts, the CO2 has enough power to force the nerf dart about a dozen feet from the turret. It appears that the aerodynamic instability that arises directly at ejection can cause the tail of the nerf dart to gyrate and bleed a large portion of its kinetic energy.

Despite the high torque motor and cam rotor, the CO2 assembly does not have enough torque to properly depress the push button on the CO2 valve. This means that to fire a nerf dart, an operator must press the button. However, the use of the needle valve and standard extension tubing to connect the CO2 assembly to the needle mounted on the nerf payload worked flawlessly.

The electrical hardware was simple compared to the mechanical assembly of the turret. With a limited failure surface area, the electrical system performed perfectly. The

catch with the electrical assembly is that the L293D H-Bridges require 5V on for its logic input voltage. The Nano 33, however, is a 3.3V device that can only supply 3.3V. This means that the power and enable pins on the L293D must be tied to the input voltage from the USB battery bank and are always at maximum, removing the possibility of speed control. However, for the logic high levels for the input pins, a 2.3V signal is counted as high; this is vital because it enables us to use the regular digital output pins of the Nano.

The software system uses the official WIFI Nina library to operate the Nano's Nina Wi-Fi chip. This simplifies the development of the MCU code that runs on the Nano and increases reliability. The microprocessor of the Nano is fast enough that we can send a steady stream of data from the host machine without causing processing latency problems from on the Nano. Since a stream is possible, we can simply continuously relay the keyboard messages from the control station without extra timing limitations built in.

Final turret videos:

<https://drive.google.com/file/d/11LYkKGSudl8TtC4clvGSsLEwLnAlpMwB/view?usp=sharing>

[Functional turret.MOV](#)

Conclusion

Overall, the system performed to expectations. The main point of pain for the project is the CO2 firing system. If more time was available, a new system would be designed to hold the CO2 canister much more rigidly. Preferably, a valve opened and closed with a solenoid that is built in. This would enable rapid and short releases of CO2 that make for ideal nerf dart ejections.

It would also be preferable to design a custom PCB that is identical to the silhouette of the base plate. This would mean that almost all wiring and connections could be made with reliable PCB copper traces and pours; greatly reducing wire clutter and consistency with connections over the current breadboard solution.

References

Malmquist, Daniel, et al. "Holistic Design Methodology for Mechatronic Systems." *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 228, no. 10, 2014, pp. 741–757, <https://doi.org/10.1177/0959651814527936>.

Appendix

Arduino Nano 33 IOT datasheet:

<https://docs.arduino.cc/static/54ff45bdf0ca111f9c03f0996ea8c407/ABX00027-datasheet.pdf>

L293D datasheet:

https://www.ti.com/lit/ds/symlink/l293d.pdf?ts=1683652594758&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FL293D%253Futm_source%253Dgoogle%2526utm_medium%253Dcpc%2526utm_campaign%253Dasc-null-null-GPN_EN-cpc-pf-google-ww%2526utm_content%253DL293D%2526ds_k%253DL293D%2BDatasheet%2526DCM%253Dyes%2526gclid%253DCjwKCAjw3ueiBhBmEiwA4BhspJSGxsGL_kqceCUZXdoou_ZWSvoWoGsg0xkbBFJgejkVTpIQ8oq_JxoCbYIQAvD_BwE%2526gclsrc%253Daw.ds

Nerf Turret repository for cad files and code source files:

<https://github.com/JoshuaBlack0704/nerfturret>