Nerf Turret Project

Produced Solo By

Joshua Black

Mechanical Engineering Senior

0

Project Outline

1. Goals
2. Constraints
3. Mechanical Design
4. Electrical Design
5. Software Design
6. Results

Goals

1

Pan-Tilt turret connectable to household network technologies such as Wi-Fi.

Remotely controllable from any laptop.

Propel Nerf darts.

Could be used to defer porch thieves.

Constraits

2

- Must connect to WPA2 encrypted network. (Smart phone hot spot)
- Main structure must be 3-D printable.
- Use pressurized gas. (Potential for pepper spray)
- 5V power supply.
- Must be controlled through network clients.

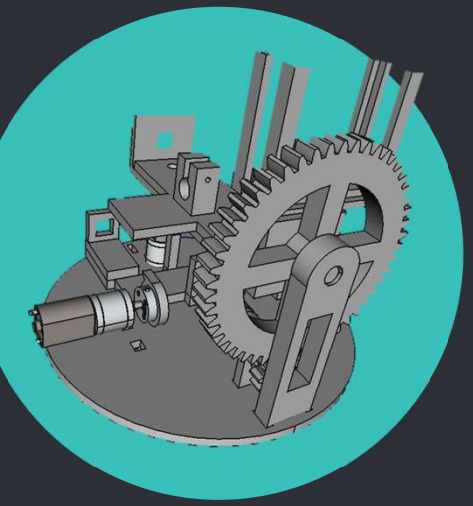
THIS IS A SLIDE TITLE

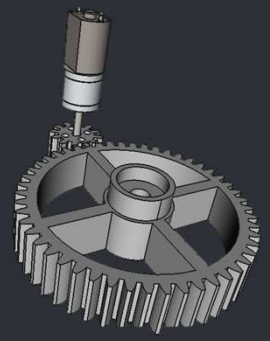
Here you have:

- A list of items
- And some text
- But remember not to overload your slides with content

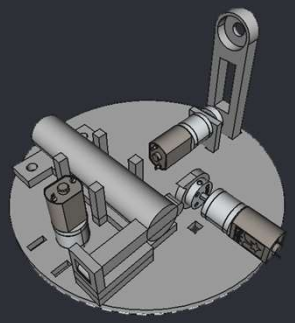
Your audience will listen to you or read the content, but won’t do both.

Mechanical Design

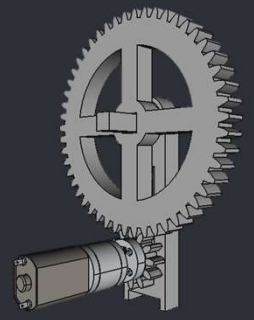




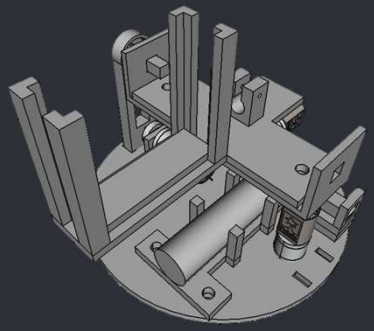
Pan Gears
- Spur
- 5:1 ratio
- Motor Driven



Base Plate
- Packed
- Slot and glue mounting
- Rigid

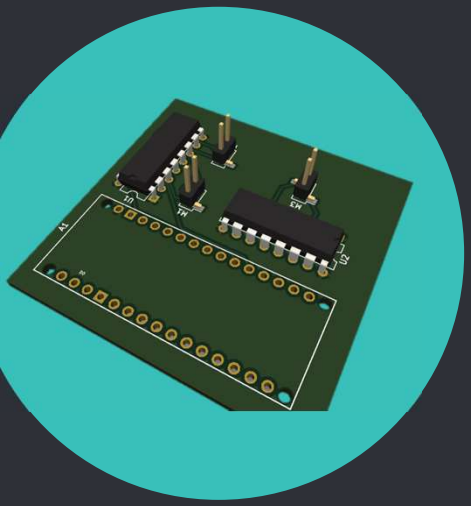


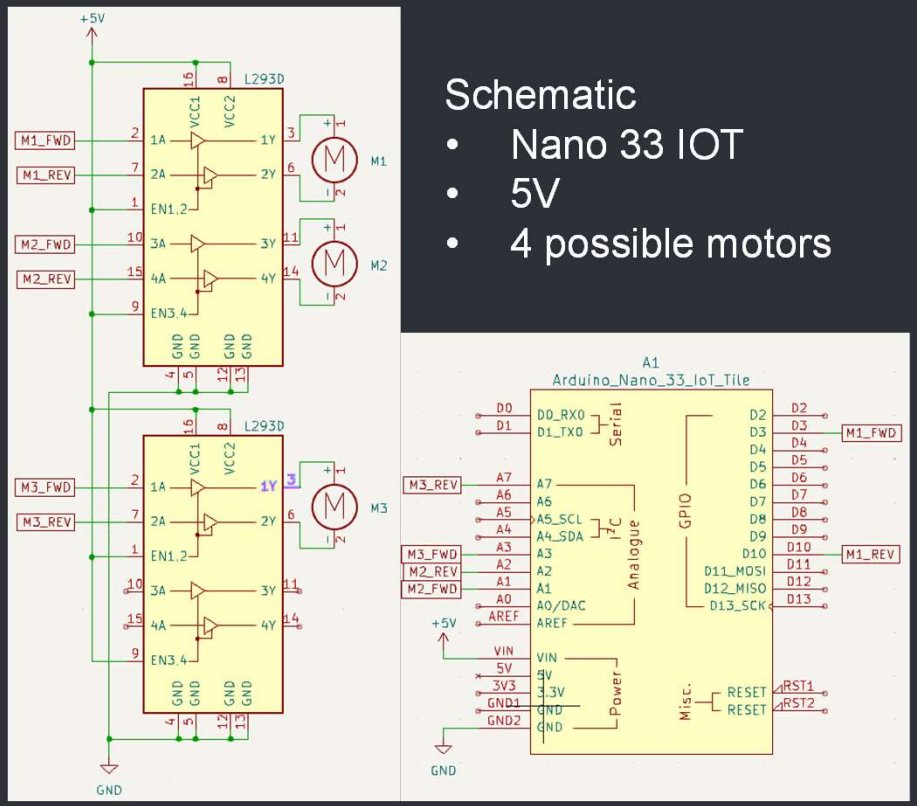
Tilt Gears
- Spur
- 5:1 ratio
- Motor Driven



Nerf Payload
- Pinned mount
- Pressure clip holder
- Nozzle clamp

Electrical Design



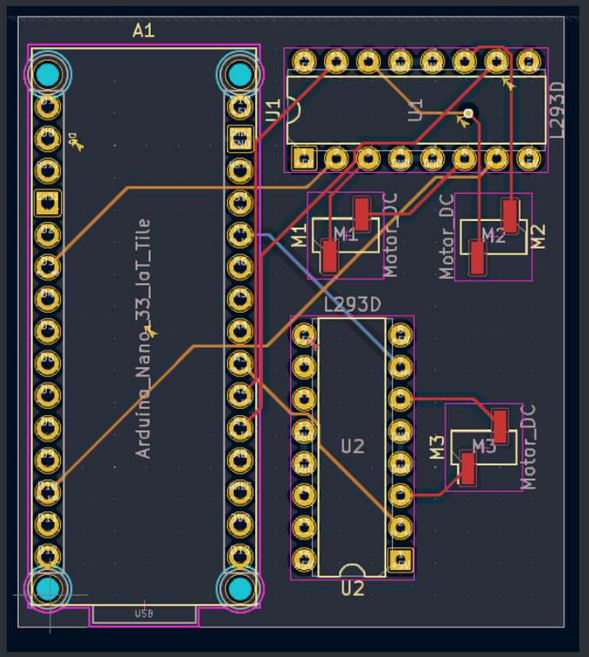


Schematic

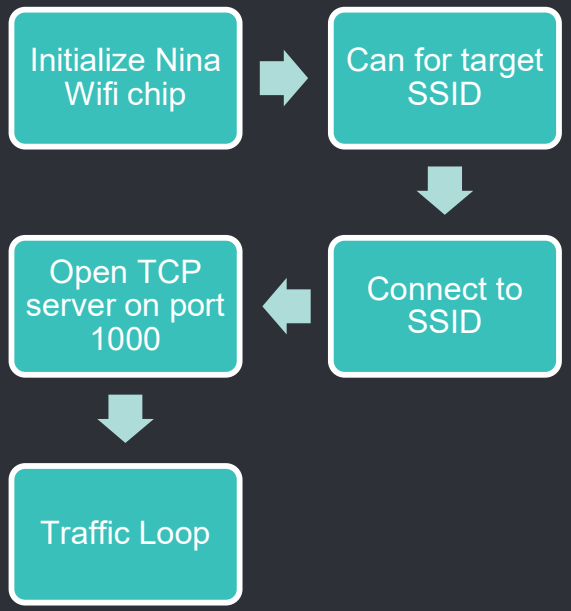
- Nano 33 IOT
- 5V
- 4 possible motors

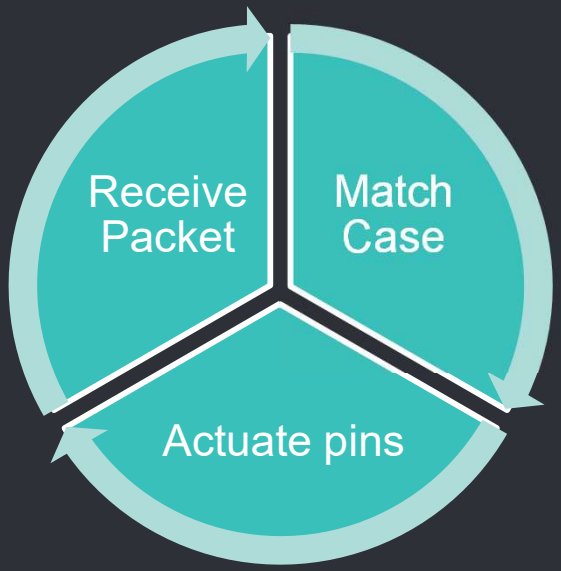
PCB

- 4 Layers
- Compact
- Not implemented



Software Design






```rust
match evt_rx.recv().await{
    Ok(KeyEvent { code: KeyCode::Up, modifiers: _, kind: KeyEventKind::Press, state: _ }) => {
        let _ = tgt.write_u8(Command::TILT_UP.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("Sent command".magenta()));
    }
    Ok(KeyEvent { code: KeyCode::Up, modifiers: _, kind: KeyEventKind::Release, state: _ }) => {
        let _ = tgt.write_u8(Command::TILT_OFF.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("Sent command".magenta()));
    }
    Ok(KeyEvent { code: KeyCode::Down, modifiers: _, kind: KeyEventKind::Press, state: _ }) => {
        let _ = tgt.write_u8(Command::TILT_DOWN.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("Sent command".magenta()));
    }
    Ok(KeyEvent { code: KeyCode::Down, modifiers: _, kind: KeyEventKind::Release, state: _ }) => {
        let _ = tgt.write_u8(Command::TILT_OFF.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("Sent command".magenta()));
    }
    Ok(KeyEvent { code: KeyCode::Right, modifiers: _, kind: KeyEventKind::Press, state: _ }) => {
        let _ = tgt.write_u8(Command::PAN_RIGHT.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("Sent command".magenta()));
    }
    Ok(KeyEvent { code: KeyCode::Right, modifiers: _, kind: KeyEventKind::Release, state: _ }) => {
        let _ = tgt.write_u8(Command::PAN_OFF.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("S
    }
    Ok(KeyEvent { code: KeyCode::Left, modifiers: _, kind: Ke
        let _ = tgt.write_u8(Command::PAN_LEFT.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("$
    }
    Ok(KeyEvent { code: KeyCode::Left, modifiers: _, kind: Ke
        let _ = tgt.write_u8(Command::PAN_OFF.into()).await;
        let _ = execute!(stdout, style::PrintStyledContent("S
    }
```

```
// listen for incoming clients
WiFiClient client = server.available();
if (client) {
  Serial.println("new client");
  while (client.connected()) {
    if (client.available()) {
      int8_t cmd = client.read();
      Serial.println(cmd);
      if (cmd == TILT_UP){
        digitalWrite(TILT_1, LOW);
      }
      if (cmd == TILT_DOWN){
        digitalWrite(TILT_2, LOW);
      }
      if (cmd == TILT_OFF){
        digitalWrite(TILT_1, HIGH);
        digitalWrite(TILT_2, HIGH);
      }
      if (cmd == PAN_RIGHT){
        digitalWrite(PAN_2, LOW);
      }
      if (cmd == PAN_LEFT){
        digitalWrite(PAN_1, LOW);
      }
      if (cmd == PAN_OFF){
        digitalWrite(PAN_1, HIGH);
        digitalWrite(PAN_2, HIGH);
      }
    }
```

Results

Mechanical

- Great base plate rigidity with super glue
- Compact
- Tilt motor at torque capacity
- Need different CO_2 firing assembly

Electrical

- Simple
- Can use higher voltage for more motor torque
- Implementing PCB can save space
- Battery maybe oversized

Software

- Success with network systems
- Easy network scan and connection
- Arrow key interface is fun
- Fault tolerant

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

Functional_turret.MOV