

EETEC PAULINO BOTELHO

Abner Firmino Cerqueira
Nicolas Francisco Gonzales
Nicolas Rodrigues

**PROTOCOLO FIRMATA E SUAS APLICAÇÕES EM DESENVOLVIMENTO
DE PROJETOS**

**SÃO CARLOS-SP
2025**

Abner Firmino Cerqueira
Nicolas Francisco Gonzales
Nicolas Rodrigues

**ANÁLISE E APLICAÇÃO DO PROTOCOLO FIRMATA EM UM LANÇADOR
AUTOMÁTICO**

Trabalho de Conclusão de
Curso apresentado à ETEC
Paulino Botelho, como requisito
parcial para a obtenção do título
de Técnico em eletrônica.
Orientador: Prof. Valter Cesar
Govoni

SÃO CARLOS-SP
2025

AGRADECIMENTOS

Agradecemos primeiramente ao nosso professor e orientador Valter Cesar Govoni por sua excelente mentoria.

Também destacamos a importância de cada professor que com muita paciência se dedicou a nossa formação.

Dedicamos este trabalho aos nossos amigos e companheiros que estiveram conosco durante todos esses anos, sem eles não seria possível a realização deste projeto.

Cerqueira, Abner Firmino; Gonzales, Nicolas Francisco; Rodrigues, Nicolas; análise e aplicação do protocolo firmata em um lançador automático. 2025. Trabalho de Conclusão de Curso (Técnico em Eletrônica) – Etec Paulino Botelho, São Carlos , 2025.

RESUMO

Este trabalho tem como objetivo analisar o Protocolo Firmata, explicar seu funcionamento e exemplificar possíveis aplicações além de provar sua eficácia em desenvolvimento de projetos. Para tal, foi desenvolvido um protótipo funcional de uma lançador controlada diretamente pelo computador capaz de alterar a direção para qual está sendo apontada e atirar bolas de gel, para isso foi necessário conhecimento prévio de programação, mais precisamente, de conhecimento sobre a linguagem Python além de competência técnica na área da eletrônica. O resultado prova a eficácia do Protocolo Firmata em aplicações mais simples ou intermediárias, com isso, demonstrando seu potencial em projetos mais avançados.

Palavras-chave: Python. Firmata. Microcontrolador. Programação. Lançador.

ABSTRACT

This work aims to analyze the Firmata Protocol, explaining its operation, possible applications, and demonstrating its effectiveness in project development. To this end, a functional prototype of a launcher controlled directly by a computer was developed, capable of adjusting its direction and firing gel balls. The implementation required prior programming knowledge—specifically in Python—as well as technical skills in electronics. The results confirm the efficiency of the Firmata Protocol in simple to intermediate applications, also highlighting its potential for more advanced projects.

Keywords: Python. Firmata. Microcontroller. Programming. Launcher.

SUMÁRIO

1	INTRODUÇÃO.....	7
1.1	JUSTIFICATIVA.....	7
1.2	OBJETIVOS.....	7
1.3	METODOLOGIA.....	7
2	DESENVOLVIMENTO.....	8
2.1	MICROCONTROLADORES.....	9
2.2	PROGRAMAÇÃO.....	9
2.3	PROTOCOLO FIRMATA.....	9
2.3.1	HISTORIOGRAFIA E FUNCIONAMENTO.....	9
2.4	LANÇADOR AUTOMÁTICO.....	12
2.4.2	MONTAGEM, PARTE FÍSICA.....	13
2.4.3	MONTAGEM, PARTE DIGITAL.....	19
3	CONCLUSÃO.....	24

1 INTRODUÇÃO

Com o avanço da tecnologia dos softwares, e sua crescente influência em dispositivos de hardware, se tornou necessário um meio de estabelecer de forma prática e acessível uma comunicação entre estes dois meios. Nesse cenário, surge o Protocolo Firmata, um protocolo capaz de conectar microcontroladores em especial, Arduinos ou derivados, diretamente aos softwares do computador ampliando suas aplicações de forma exponencial.

O domínio dessa ferramenta permite que projetos amadores ou mesmo profissionais tenham seus potenciais explorados ao máximo de forma relativamente simples. É ideal para quem busca trabalhar de forma profissional ou apenas criar seus próprios projetos.

Com isso, será explicado seu funcionamento, possíveis aplicações e uma demonstração prática em um lançador controlado através do computador.

1.1 JUSTIFICATIVA

Originalmente, o projeto se resumia ao desenvolvimento de um lançador automático, posteriormente devido as dificuldades encontradas, essa ideia foi descartada, durante o projeto do lançador foi observado alguns problemas que embora simples se tornaram muito complicados de ser resolvidos pelo meio tradicional, com isso, foram direcionadas pesquisas a fim de encontrar uma solução prática. Essa solução foi o Protocolo Firmata e embora tenha sido motivo de facínio para os integrantes do grupo, é uma ferramenta praticamente desconhecida e pouco usada ou ao menos, pouco divulgada em redes sociais. Por esses motivos, o Protocolo Firmata se tornou o tema central deste trabalho.

1.2 OBJETIVOS

Explicar o Protocolo Firmata, exemplificar suas aplicações e possíveis formas de uso. Demonstrar sua capacidade através do funcionamento de um lançador feito com Arduino e controlada através de um computador.

1.3 METODOLOGIA

As informações sobre o Protocolo Firmata foram adquiradas através de uma análise do código do próprio protocolo. O código do lançador foi escrito em Python. Para a construção do projeto foi utilizada um Arduino Nano, dois servo-motores, uma estrutura de metal para servir como suporte para os motores e uma base construída pelos membros do grupo. O disparador de gel foi obtido de uma arma de brinquedo e é acionada por um relé conectado ao Arduino, ele funciona através de um sistema de engrenagens ligadas a um motor cc genérico, é baseado em um sistema de pressão para disparar as bolinhas de gel semelhante ao funcionamento de uma seringa. Também foi implementado um raio laser para auxiliar na precisão dos disparos. O circuito foi montado em uma placa perfurada soldado manualmente.

2.1 MICROCONTROLADORES

Em primeira instância, é necessário entender a base do projeto: os microcontroladores. Os microcontroladores são basicamente um mini computador. Possuindo processadores, memória RAM e outros periféricos de entrada e saída, eles servem basicamente como um cérebro para o equipamento podendo enviar ou receber sinais de acordo com as necessidades de cada usuário. Atualmente há diversos microcontroladores de vários modelos no mercado, mas o mais popular é o Arduino, esse do qual, é baseado na linguagem de programação C++.

Os microcontroladores são ferramentas úteis para o desenvolvimento de qualquer tipo de projeto, grandes ou pequenos, são baratos e relativamente fáceis de usar, o que os torna acessíveis para qualquer um que tenha interesse em se aprofundar na área.

2.2 PROGRAMAÇÃO

A programação é a outra frente fundamental desse projeto, é nela que se baseia o funcionamento dos Microcontroladores. De forma simplificada, programar significa descrever tarefas que o computador deverá realizar, no entanto, para isso, é utilizado uma forma de linguagem diferente da convencional, estas são as linguagens de programação. No passado, essas linguagens eram de extrema dificuldade de entendimento humano, no entanto, atualmente existem diversas linguagens mais simples para à compreensão como Python, Java Script, C, C+, C++ e outras. Essas linguagens facilitam o trabalho na hora de programar, mas não os deixam menos complexos, ainda dessa forma é necessário utilizar de lógica e conhecimento das normas da linguagem para o funcionamento adequado do código.

2.3 PROTOCOLO FIRMATA

2.3.1 HISTORIOGRAFIA E FUNCIONAMENTO

Durante o desenvolvimento de projetos eletrônicos com o uso de um Arduino, é possível que um desenvolvedor encontre alguns obstáculos na implementação de novas funções ao tentar conectar o microcontrolador à algum software do computador devido a limitações do próprio Arduino, nesse caso há duas possibilidades: o indivíduo pode desenvolver o próprio protocolo de comunicação ou utilizar o Protocolo Firmata, um protocolo padrão no Arduino.

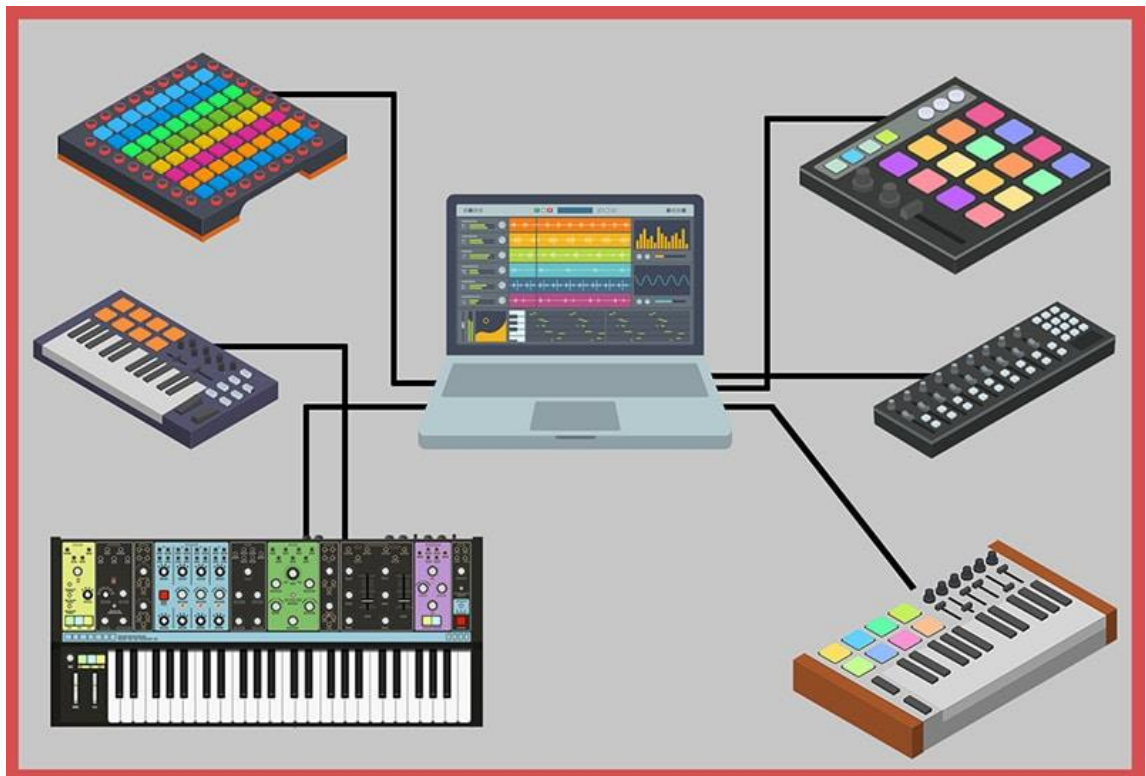
O Protocolo Firmata, é um protocolo capaz de estabelecer uma comunicação entre softwares de computadores com um microcontrolador, preferivelmente um Arduino ou derivado para um melhor desempenho. Esse protocolo é possível graças a um outro protocolo:

Protocolo Musical Instrument Digital Interface (MIDI) que originalmente foi projetado para conectar instrumentos musicais aos softwares de computador, mas que também serviu de base para a criação do Protocolo Firmata.

Felizmente não é preciso dominar o conhecimento sobre o funcionamento do MIDI ou como de fato o Protocolo Firmata trabalha para utilizar suas ferramentas, no entanto, a título de curiosidade segue a explicação: Musical Instrument Digital Interface como o próprio nome sugere é uma interface digital de instrumentos em um computador, foi inventado em meados de 1980 como uma forma universal de comunicação entre diferentes instrumentos, antes do MIDI, era necessário o uso de sequenciadores, nesse ponto, o grande problema dos sequenciadores era de que nem sempre eram compatíveis com outros instrumentos e equipamentos de um estúdio o que atrapalhava na produção e gerava gastos expressivos para a equipe. Graças ao MIDI esse problema foi contornado, mas como ele realmente funciona? Esse Protocolo possui basicamente três funções: MIDI-IN, MIDI-OUT e MIDI-THRU, elas servem respectivamente para receber, enviar e traduzir sinais para o sistema, por coincidência é basicamente assim que um microcontrolador funciona: recebendo, enviando e traduzindo essas informações.

Por essa semelhança com microcontroladores, o Protocolo MIDI foi usado para o desenvolvimento do Protocolo Firmata. A figura 1 demonstra o funcionamento do MIDI:

Figura 1



Fonte: <https://woodandfirestudio.com/pt/was-ist-midi/>

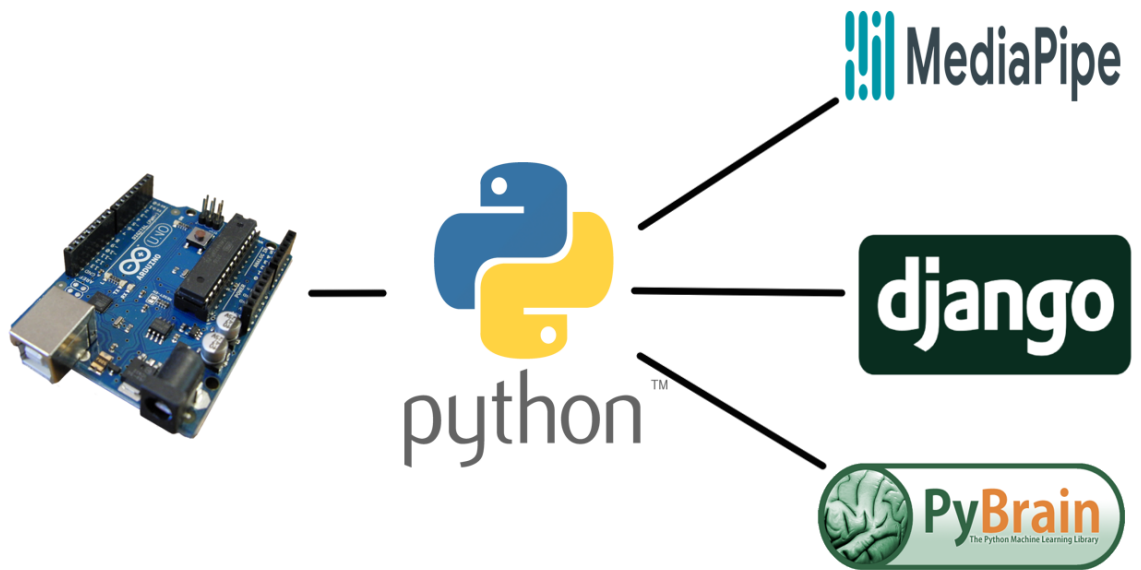
2.3.2 PRÁTICA

Na prática, o Protocolo Firmata oferece uma quantidade gigante de recursos para o usuário fazer o que quiser, afinal, este está conectado com todos os recursos que um computador pode oferecer, por tanto suas funções se limitam as competências e recursos do usuário. No entanto, cabe uma explicação como de fato o protocolo pode ser utilizado.

Em um primeiro contato, deverá ser escolhido alguma linguagem de programação, como Python, Java Script ou o próprio C++ do Arduino, de qualquer forma, estará disponível algumas funções nativas do protocolo, como poder configurar o pino do Arduino como: ANALÓGICO ou DIGITAL e OUTPUT, INPUT, PWM, SERVO ou PULL-UP, além das funções write para configurar um pino e read para lê-lo. Estas funções citadas anteriormente são as mais importantes e provavelmente as únicas que serão utilizadas em qualquer projeto convencional. É certo que essas funções não são o motivo que dão importância ao Protocolo Firmata, afinal, já são presentes no software original do Arduino, no entanto, seu destaque se dá na hora de implementar uma conexão com outros softwares que podem ser encontrados pela internet ou feitos pelo próprio usuário. Melhor seria dizer que o Protocolo Firmata é um intermediário entre microcontroladores e o computador.

Aqui segue alguns exemplos de softwares que podem ser combinados com Arduino através do Firmata: Media Pipe, Pybrain e Django. Segue a Figura 2, um esquema de conexão entre um microcontrolador uma linguagem de programação e os softwares citados.

Figura 2 – diagrama exemplo: Arduino e Python



Fonte: autoria própria

2.4 LANÇADOR AUTOMÁTICO

2.4.1 IDEIA

Como citado anteriormente, a ideia original era fazer apenas um lançador automático controlado pelo computador, no entanto, devido as dificuldades enfrentadas ao tentar fazer

pelos métodos tradicionais (utilizando o Arduino IDE) foi decidido que seria mais interessante dedicar este trabalho ao Protocolo Firmata por ser algo de aplicações gerais e não ser tão específico, porém, a ideia de um lançador ainda ficou como uma ótima forma de demonstração das possibilidades do que este protocolo tem a oferecer.

2.4.2 MONTAGEM, PARTE FÍSICA

Para um melhor entendimento do projeto, a explicação prosseguirá de forma fragmentada em duas áreas diferentes, a física e a digital, a começar pela física. A tabela 1 mostra os recursos utilizados na montagem.

Tabela 1 - Recursos utilizados

MATERIAIS	QUANTIDADE
SERVO-MOTOR	2
PLACA PCB	1
ARDUINO NANO	1
FONTE 7-7,5V	2
RELÉ	1
ESTRUTURA	1
LASER	1
MOTOR DO LANÇADOR	1
CAPACITOR 47 μ F	1

Fonte: autoria própria

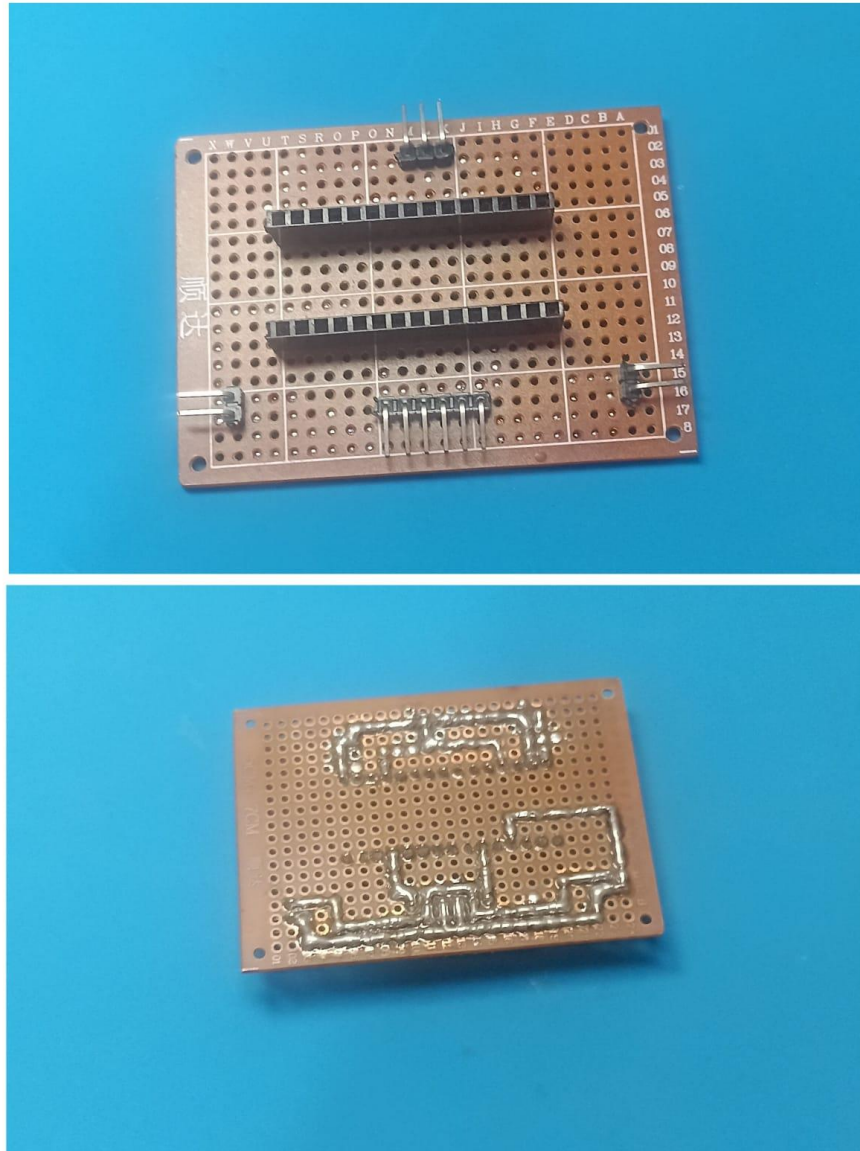
Em um primeiro momento foi soldado em uma placa perfurada pinos machos 90° para servir de entrada para os servo-motores, para uma fonte de alimentação e relé. Além disso foi soldado algumas conectores fêmeas para conectar o Arduino Nano e todas as trilhas foram soldadas na placa. As posições dos pinos e das trilhas foram escolhidas estrategicamente para

facilitar a solda. Foi construída uma base com sucata para servir de suporte para os servo-motores e para a placa.

Neste protótipo todos os componentes foram selecionados de forma meticulosa para atender as necessidades dos projetos, os servo-motores escolhidos foram Mg995 e Mg996, eles são fortes o suficiente para segurar o lançador gira-lo em sentido horizontal e vertical e aguentar o seu recuo, este por sua vez foi posicionado de modo a ter um torque mínimo na hora de executar disparos. O Arduino Nano se encaixou de forma perfeita ao projeto uma vez que ele possui um tamanho compacto e preço acessível. Ao receber o comando do computador, o Arduino enviará um pulso ao relé do qual abrirá passagem de uma corrente para o lançador. Dessa forma executando os disparos.

A seguir, a figura 3, figura 4, figura 5 e figura 6 correspondem respectivamente a placa utilizada, a estrutura construída, o disparador e o projeto montado.

Figura 3 – Placa PCB



Fonte: Autoria própria

Figura 4 – Estrutura



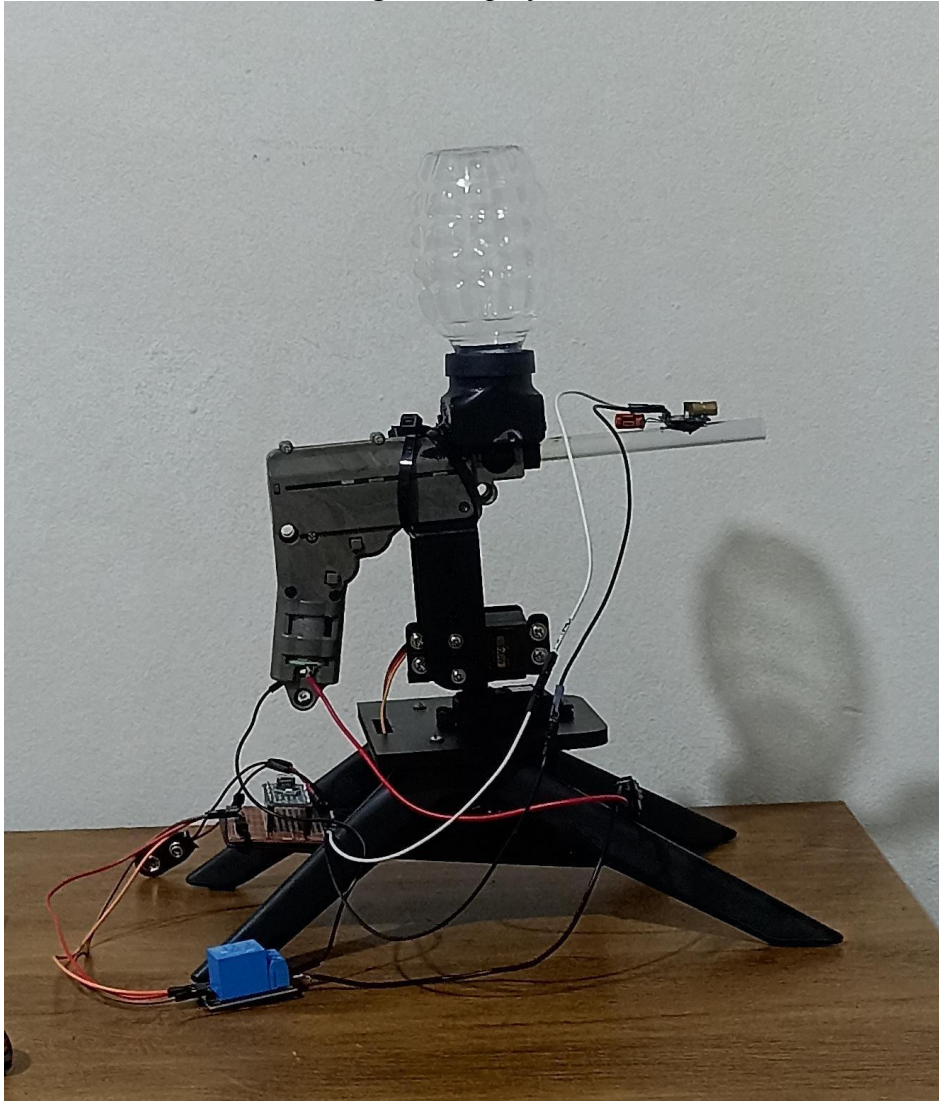
Fonte: Autoria Própria

Figura 5 - Lançador



Fonte: autoria própria

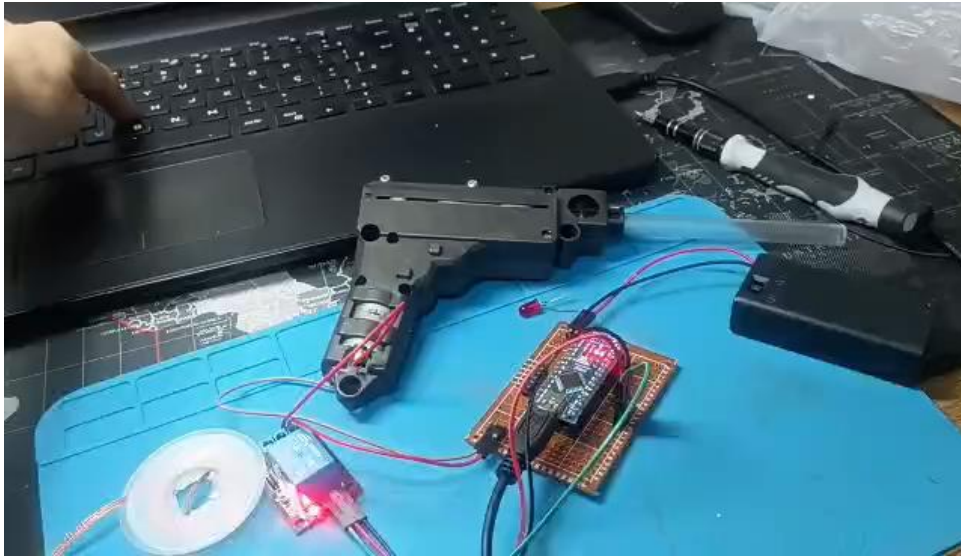
Figura 6 – projeto montado



Fonte: Autoria própria.

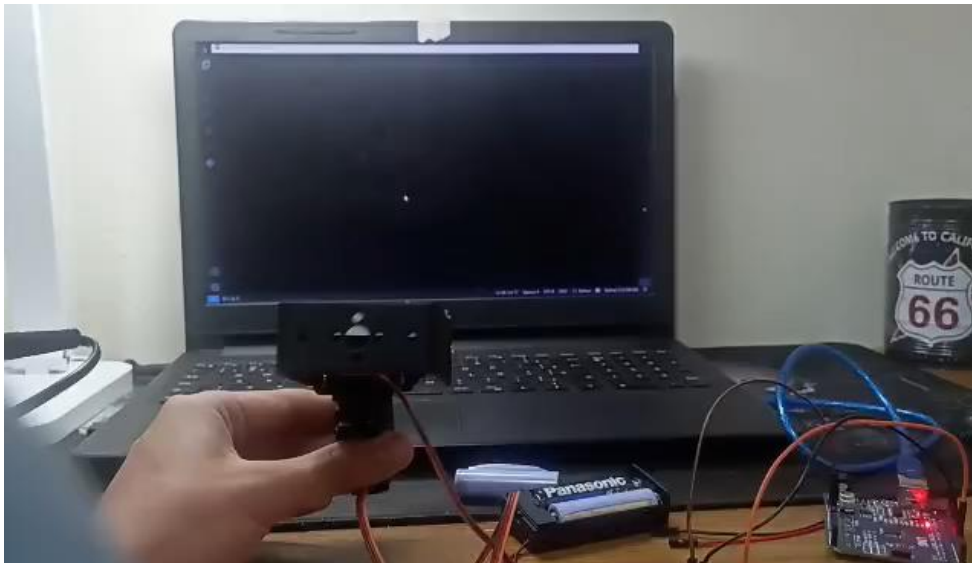
A montagem foi feita de forma fragmentada, foi feito testes primeiros com os motores e depois com o lançador para só então eles serem unidos em um só corpo. Abaixo, a figura 7 e a figura 8 demonstra o desenvolvimento:

Figura 7- teste do lançador



Fonte: Autoria própria

Figura 8 – teste dos motores



Fonte: Autoria própria

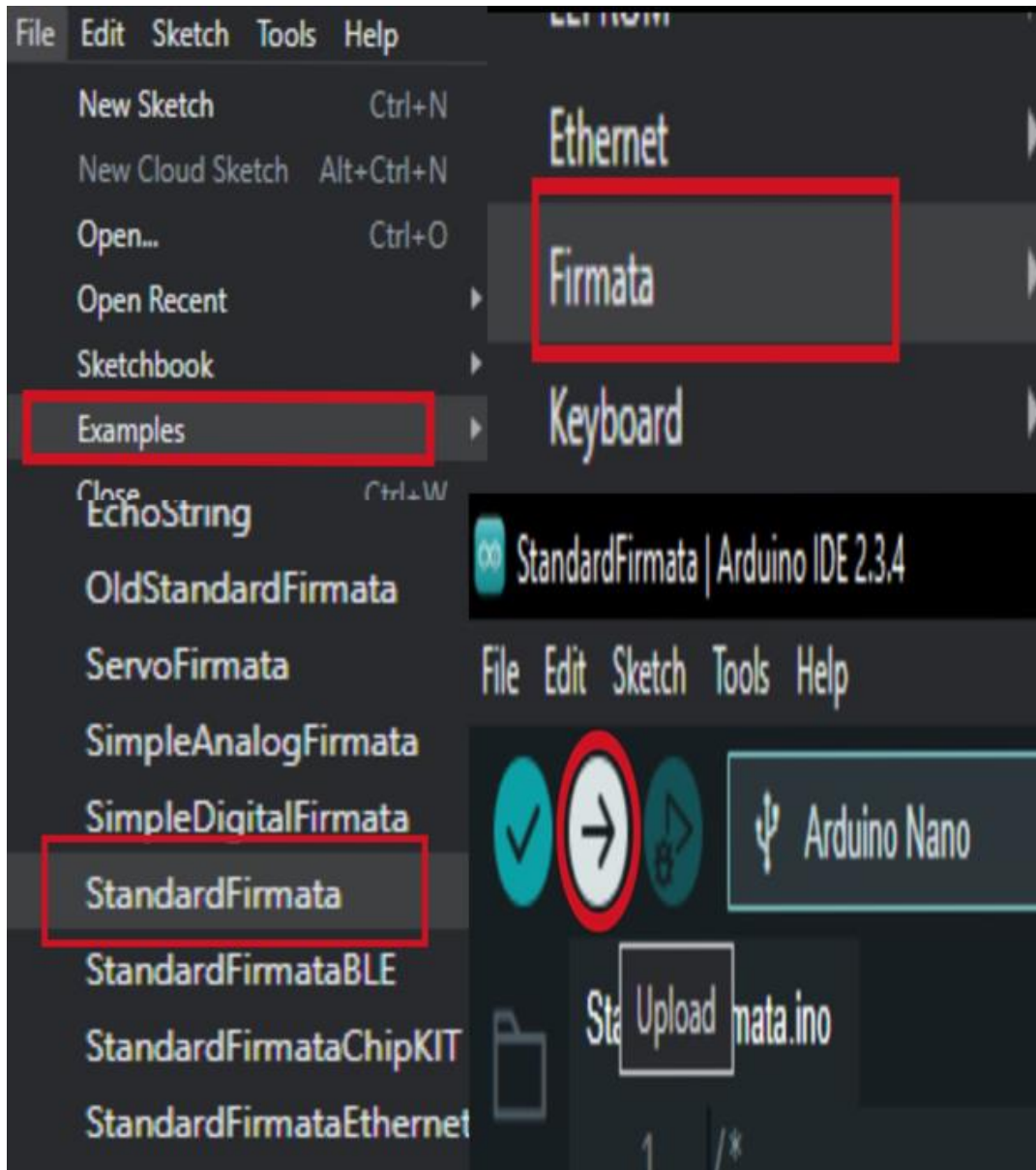
2.4.3 MONTAGEM, PARTE DIGITAL

Agora, cabe uma explicação acerca do desenvolvimento da parte digital do projeto.

O projeto foi iniciado instalando o Protocolo Firmata no Arduino, para isso foi necessário apenas conectar o Arduino no computador e executar o procedimento indicado na

figura 9. Em seguida foi necessário mudar de linguagem de programação, no caso a linguagem escolhida foi Python por sua facilidade de uso, também foram instaladas as seguintes bibliotecas para ser utilizadas no projeto: pyfirmata2, time e pygame, essas bibliotecas servem respectivamente para servir de ponte para o Protocolo Firmata, calcular o tempo durante a execução do projeto e por fim, para utilizar o mouse para controlar o lançador.

Figura 9



Fonte: autoria própria

Pois bem, como citado anteriormente, o Protocolo Firmata não é difícil de ser utilizado, até esse ponto ele já está sendo utilizado sem grandes dificuldades, no entanto, a partir daqui é necessário que o usuário tenha uma noção de programação.

O código trabalha da seguinte forma: o mouse se encontra em um plano cartesiano e conforme sua coordenada transmite um valor para o Arduino, esse do qual emite um sinal PWM para o servo motor assim controlando a sua rotação, os valores do eixo x corresponde à posição do motor que gira na horizontal, enquanto os do y corresponde ao motor que gira na vertical, esse movimento é limitado para que não ocorra acidentes, como o lançador trombar na base. Desse modo, é possível controlar a direção do lançador.

O lançador por sua vez, funciona em um circuito separado, ele é conectado a um relé que permite a passagem de corrente com um click no botão esquerdo do mouse dessa forma os disparos são efetuados.

A seguir, segue o código utilizado no projeto:

```
#importando todas as bibliotecas
from pyfirmata2 import Arduino, util
import time
import pygame as pg # Adicionado para mouse

#definindo todas as variaveis utilizadas
port = 'COM3'
board = Arduino(port)
m1 = 0
m2 = 0
vel = 3
lim = 0
xspd = 0
yspd = 0
posx = 90
posy = 90
limite = 0
gun = 3
shoot = 0
display_width = 1280
display_height = 720
```

```

def clamp(valor, minimo, maximo):
    return max(minimo, min(valor, maximo))

m1 = board.get_pin('d:3:s')
m2 = board.get_pin('d:9:s')
arma = board.get_pin('d:13:o')
Led = 0

# Inicializa Pygame para capturar mouse (adicionado)
pg.init()
screen = pg.display.set_mode((display_width, display_height)) # Janela simples para
mouse
pg.display.set_caption("Controle da Torreia via Mouse")
done = True # Para controle do loop

while done: # Mudado para done, para integrar com Pygame quit

    # Eventos do Pygame (para fechar janela)
    for event in pg.event.get():
        if event.type == pg.QUIT:
            done = False

    # Captura posição do mouse (adicionado)
    mouse_x, mouse_y = pg.mouse.get_pos()
    # Mapeia mouse (0-500) para posx/posy (0-255, como no seu clamp)
    posx = int( 180 * ((display_width - mouse_x) / display_width))
    posy = int( 180 * ((display_height - mouse_y) / display_height))

    print(posy)

    if event.type == pg.MOUSEBUTTONDOWN: shoot = 1
    else: shoot = 0

```

```
# Soma spd do teclado (se quiser hibrido; senão comente essas linhas)

posx = clamp(posx, 0, 255)
posy = clamp(posy, 0, 255)

arma.write(shoot)
m1.write(posx)
m2.write(posy)

time.sleep(0.02)

if keyboard.is_pressed("esc"): done = False # desligando o lançador

# Fecha Pygame (adicionado)
pg.quit()

#fim do código
```

3 CONCLUSÃO

Na área da eletrônica, existem diversas possibilidades de projetos que podem ser aplicados nas mais diversas funções. Na prática, o lançador apresentado é apenas uma pequena demonstração do que o domínio de diversos conhecimentos sobre a eletrônica pode criar e, embora não seja necessariamente prático no dia-a-dia, é um exemplo do que o conhecimento em ferramentas como o Protocolo Firmata pode proporcionar seja facilitando a vida de profissionais e amadores ou ampliando o potencial de diversos projetos, tudo isso de forma acessível.

Por fim, é certo afirmar que o Protocolo Firmata é uma ferramenta poderosa para ser usada em projeto dos mais variados tipos, sua simplicidade de aplicação e uso é prova disso e garante que qualquer um a utilize. De fato, não é apenas instalar e usar, ainda exige conhecimento em programação e eletrônica, mas consegue fazer boa parte do trabalho garantindo conforto no desenvolvimento de projetos.