

ARDUINO DSL



GRUPO 10

GABRIEL BENEVIDES (GBMM@CIN.UFPE.BR)
MARCUS VINÍCIUS (MVFS@CIN.UFPE.BR)

GIOVANNA TRIGUEIRO (GTAA@CIN.UFPE.BR)

25, NOV, 2025

INTRODUÇÃO

01 PROBLEMA

Apesar de muito utilizado para ensinar programação, é pouco intuitivo para não fluentes em inglês.

02 SOLUÇÃO PROPOSTA

Desenvolvimento de uma DSL voltada para facilitar a criação de projetos com Arduino.

03 OBJETIVOS

Criar uma linguagem intuitiva para programar o Arduino

1. Definir uma gramática
2. Implementar um tradutor que converta código DSL em código C compatível com o Arduino.
3. Permitir testes no simulador ThinkerCAD, com operações como:
 - o Ler e escrever em portas digitais.
 - o Ligar e desligar LEDs.
 - o Ler e escrever valores analógicos.

FUNDAMENTOS

04 O QUE É ARDUINO?

- Plataforma de desenvolvimento de projetos eletrônicos.
- Open-Source e de baixo custo.

05 PARA QUE SERVE?

- Prototipagem rápida e barata de diversos projetos.
- Compatibilidade com centenas de sensores e atuadores.

06 COMO SE PROGRAMA?

- Em geral, utiliza-se uma IDE própria, o Arduino IDE.
- Linguagem: C/C++



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

FUNDAMENTOS

07 DSL - DOMAIN SPECIFIC LANGUAGE

Linguagem Específica de Domínio

Uma linguagem específica de domínio (DSL) é um sistema que fornece ao usuário um meio expressivo de resolver um problema. Ele permite a um usuário interagir com o sistema em seus termos - não apenas na linguagem do programador, voltada para um domínio específico.

Foco no domínio? uma linguagem limitada é útil se tiver um foco claro em um domínio pequeno(FOWLER, 2019)

- DSL INTERNA(EMBUTIDA)
- DSL EXTERNA()
 - REGEX
 - GIT CLI
 - MARKDOWN
 - CSS



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



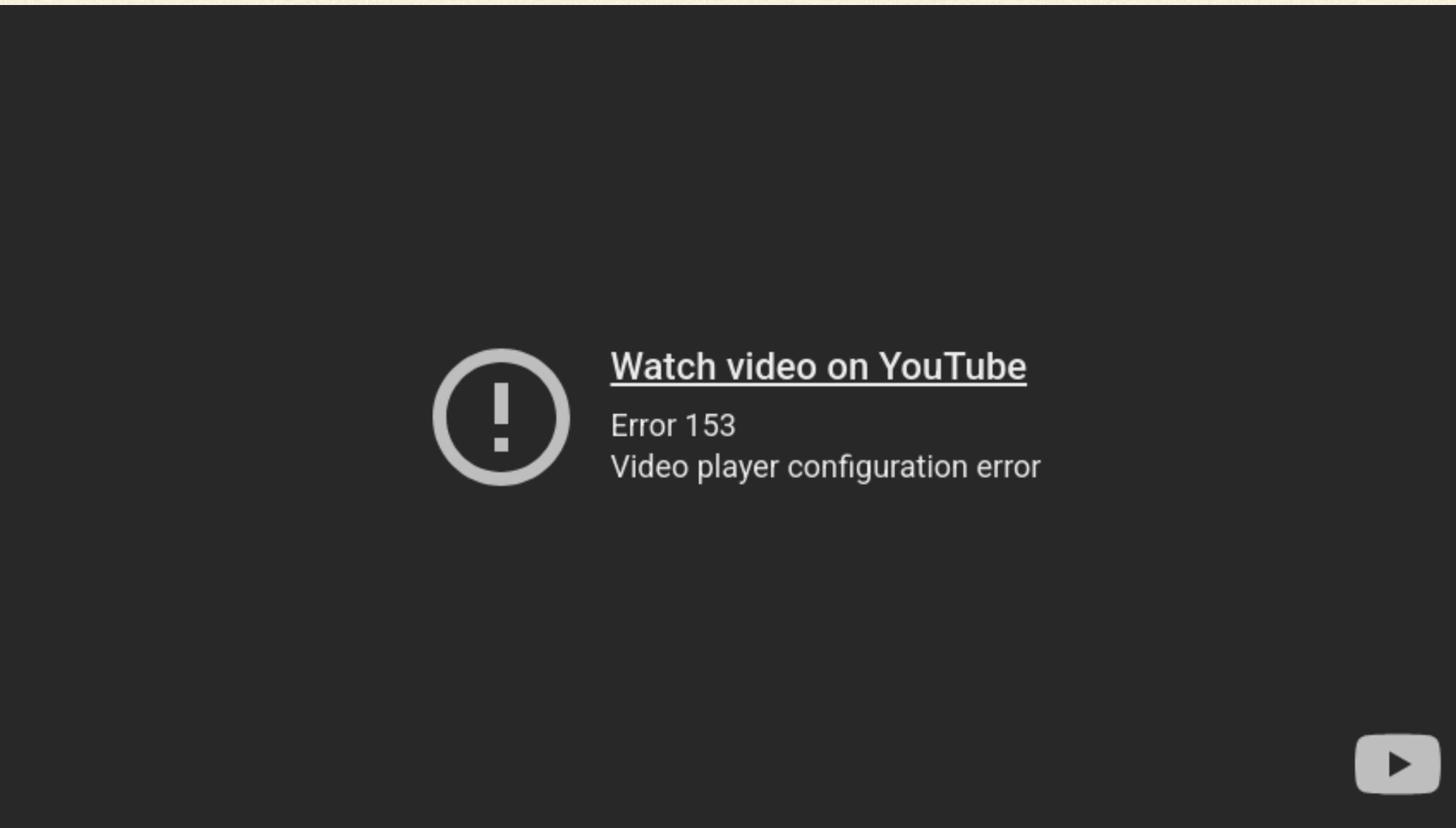
Centro de
Informática
UFPE

FUNDAMENTOS

08 THINKERCAD

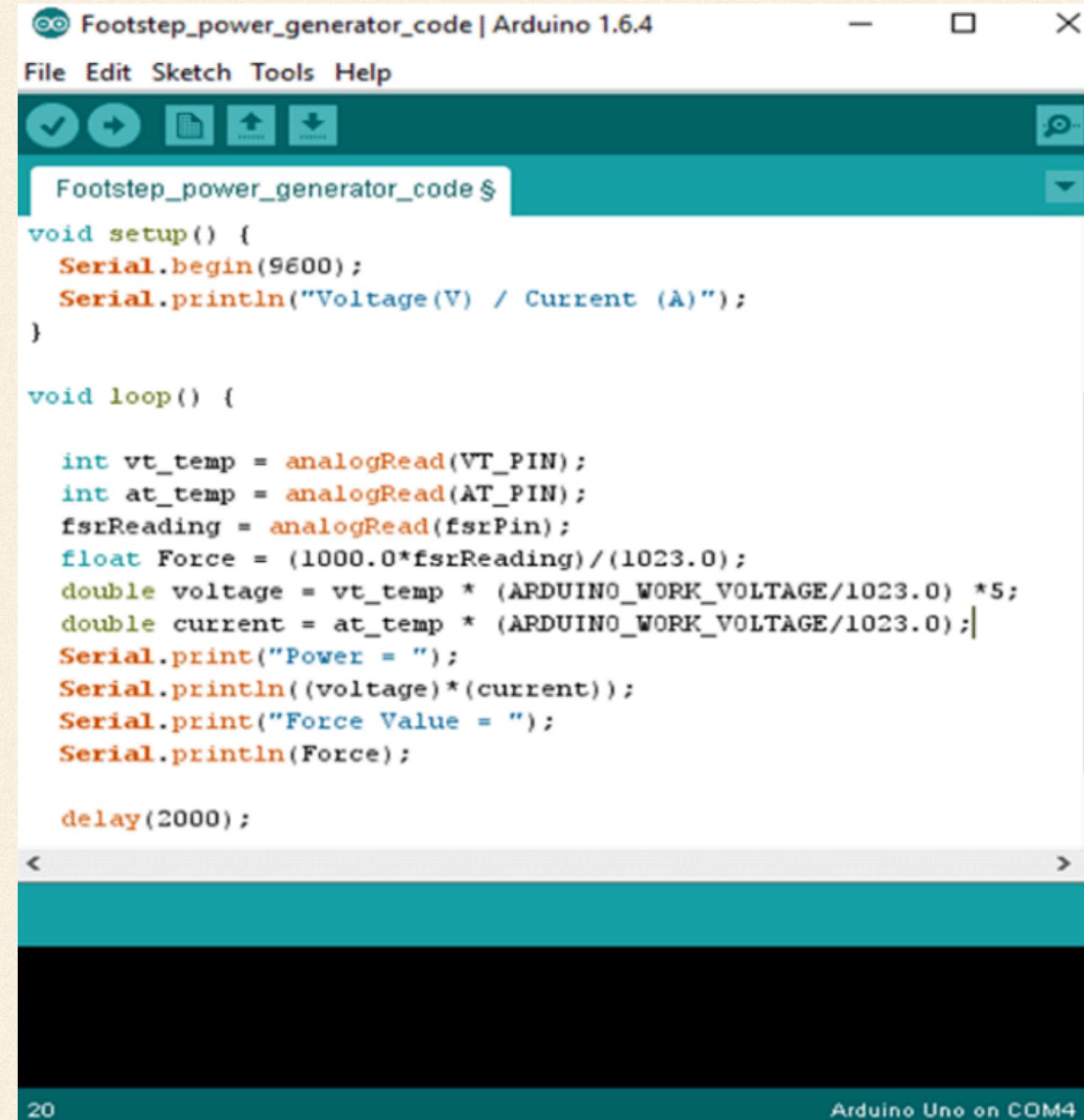
Linguagem Específica de Domínio

Autodesk Thinkercad é uma plataforma abrangente que oferece uma variedade de recursos para modelagem 3d e simulação de circuitos eletrônicos.



FUNDAMENTOS

09 ARDUINO X DSL



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** Footstep_power_generator_code | Arduino 1.6.4
- Menu Bar:** File Edit Sketch Tools Help
- Toolbar:** Includes icons for Save, Run, Upload, and Refresh.
- Sketch Editor:** Displays the C++ code for the sketch. The code reads analog inputs from pins VT_PIN and AT_PIN, calculates power and force, and prints them to the Serial monitor.

```

void setup() {
    Serial.begin(9600);
    Serial.println("Voltage(V) / Current (A)");
}

void loop() {
    int vt_temp = analogRead(VT_PIN);
    int at_temp = analogRead(AT_PIN);
    fsrReading = analogRead(fsrPin);
    float Force = (1000.0*fsrReading)/(1023.0);
    double voltage = vt_temp * (ARDUINO_WORK_VOLTAGE/1023.0) *5;
    double current = at_temp * (ARDUINO_WORK_VOLTAGE/1023.0);
    Serial.print("Power = ");
    Serial.println((voltage)*(current));
    Serial.print("Force Value = ");
    Serial.println(Force);

    delay(2000);
}

```

- Serial Monitor:** Shows the text "Arduino Uno on COM4".



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA

tokens

Comandos básicos

ACENDER: "acender"
 APAGAR: "apagar"
 ESPERAR: "esperar"
 PINO: "pino"
 LED: "led"
 SEGUNDOS: "segundos"

Controle

IF: "se"
 THEN: "entao" | "então"
 ELSE: "senao" | "senão"
 BOTAO: "botao" | "botão"
 PRESSIONADO: "pressionado"
 | "apertado"

Símbolos e Comentários

LBRACE: "{"
 RBRACE: "}"
 LPAREN: "("
 RPAREN: ")"
 SEMI: ";"
 NUMBER: sequência de dígitos (0-9)
 LINE_COMMENT: "//" seguido de qualquer caractere até o fim da linha
Tokens ignorados (SKIP)
 Espaços em branco: " ", "\t", "\r", "\n"
 Comentários de linha



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE

EXEMPLOS:
 ACENDER PINO 13
 ACENDER LED 13
 ACENDER PINO 13;



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE

EXEMPLOS:
APAGAR PINO 13
APAGAR LED 13



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE

EXEMPLOS:
ESPERAR 2 SEGUNDOS
ESPERAR 5 SEGUNDOS;



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL

GRAMÁTICA(LL1)

produções

PROGRAM → COMMAND* EOF

COMMAND → ACENDERCMD | APAGARCMD | ESPERARCMD | IFCMD

ACENDERCMD → ACENDER (PINO | LED) NUMBER [SEMI]

APAGARCMD → APAGAR (PINO | LED) NUMBER [SEMI]

ESPERARCMD → ESPERAR NUMBER SEGUNDOS [SEMI]

IFCMD → IF BOTAO NUMBER PRESSIONADO THEN LBRACE COMMAND (COMMAND)* RBRACE

SE BOTAO 4 PRESSIONADO ENTAO { ACENDER LED 13 }

SE BOTAO 4 PRESSIONADO ENTAO { ACENDER LED 13 ESPERAR 1 SEGUNDOS APAGAR LED 13 }

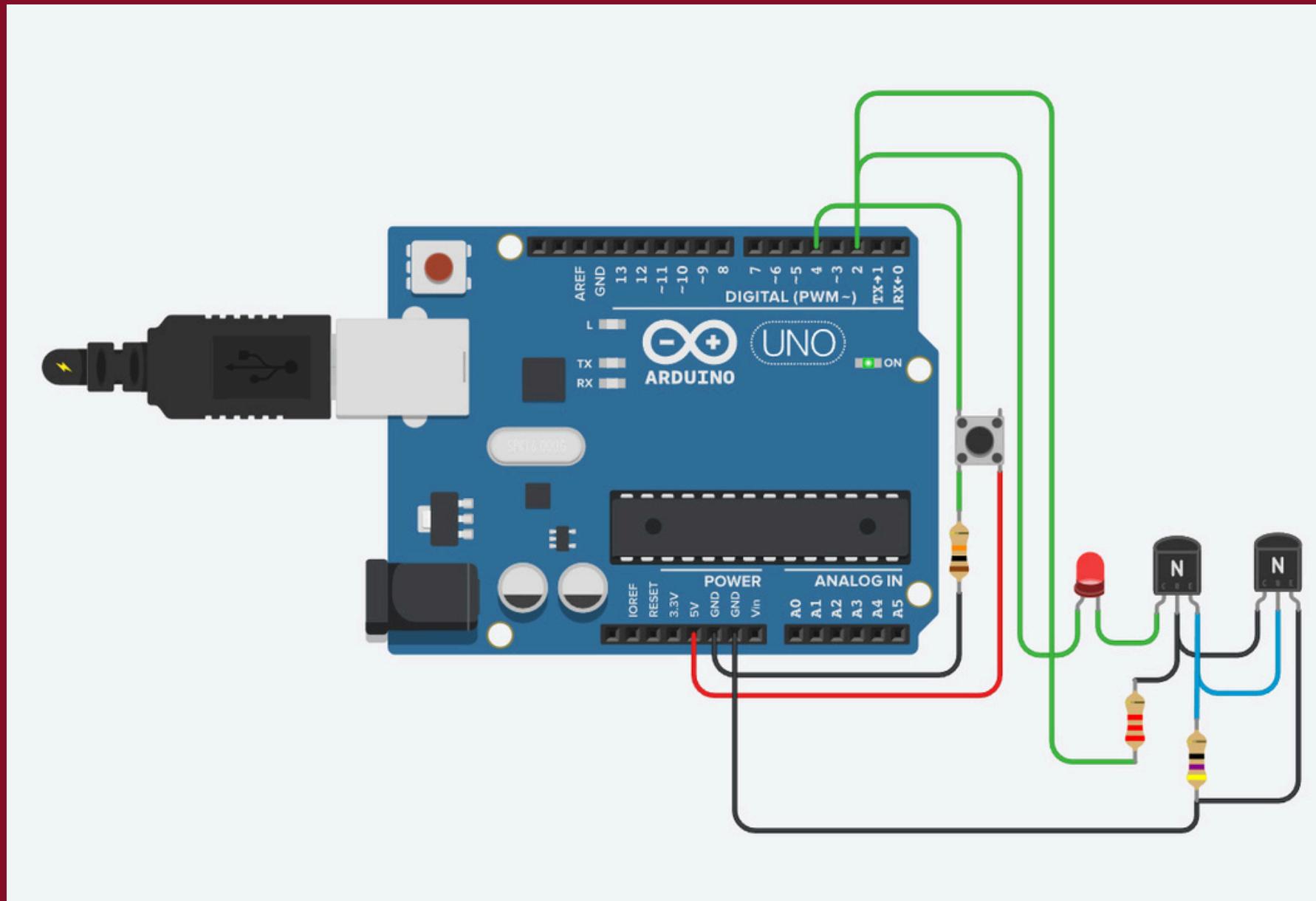


UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE

09 ARDUINO X DSL



COMANDO



```
1 se botao 4 pressionado entao:
2   acender led 2
3   esperar 1 segundos
4   fim
```

RESULTADO



```
1 void setup() {
2   pinMode(2, OUTPUT);
3   pinMode(4, INPUT_PULLUP);
4 }
5
6 void loop() {
7   if (digitalRead(4) == HIGH) {
8     digitalWrite(2, HIGH);
9     delay(1000);
10  }
11  // repetir com pequeno atraso
12  delay(100);
13 }
```

REFERÊNCIAS

01 DOMAIN-SPECIFIC LANGUAGES
GUIDE
MARTIN FOWLER, 2019

01 PROF AUGUSTO SAMPAIO



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



Centro de
Informática
UFPE