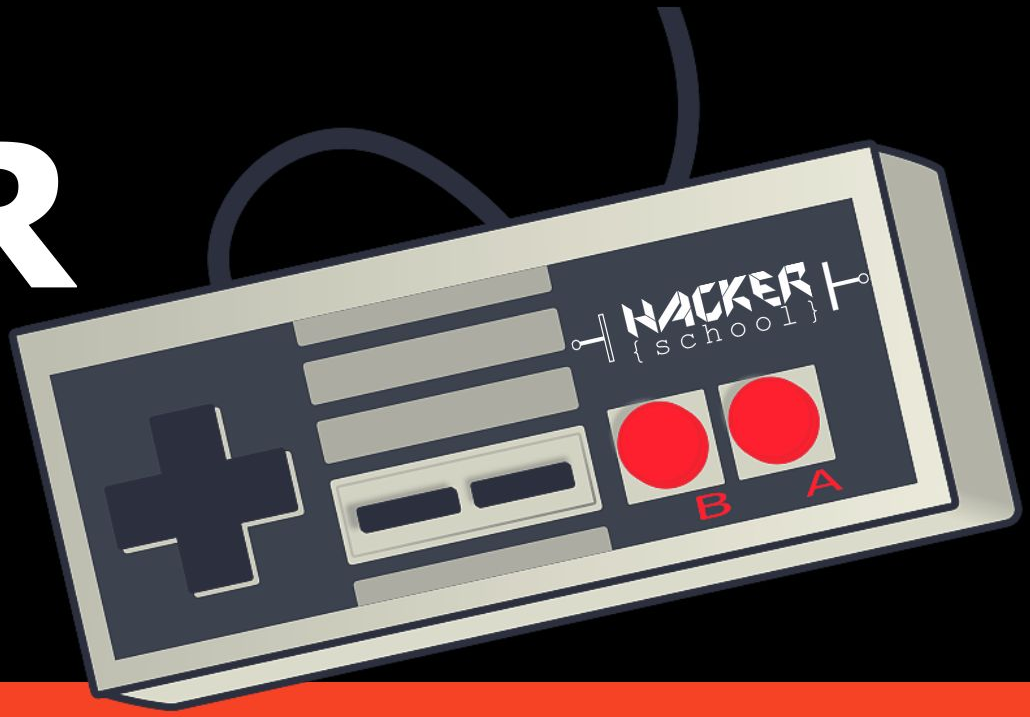
HACKER

entertainment system

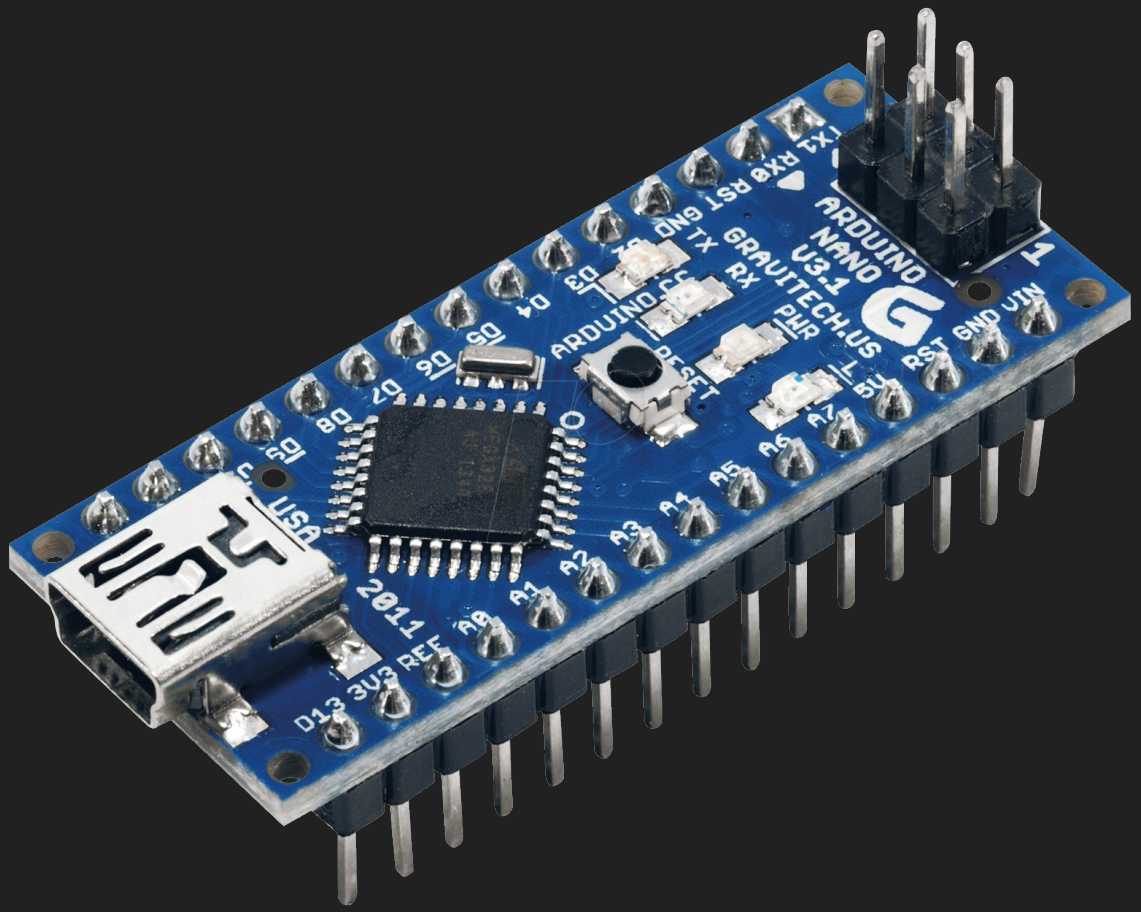


ARDUINO

João Borrego

Arduino

Arduino Nano



C,C++

Conceitos elementares

- Variáveis
- Testes de condição
- Ciclos
- Bibliotecas

Estrutura usual

- Inclusão de bibliotecas
- Declaração dos pinos como MACROs (`#define`)
- `void setup()`
- `void loop()`

Exercício 1 - Hello World!

Estrutura básica de um programa de Arduino

Serial Monitor

- BaudRate

Exercício 1 - Hello World!

```
void setup(){
   Serial.begin(9600);
}
void loop(){
   Serial.println(“Hello World!”);
   delay(1000);          // Valor em milissegundos
}
```

Exercício 2 - LED

Agora vamos modificar o programa anterior, para piscar um LED

```
#define LED 13
```

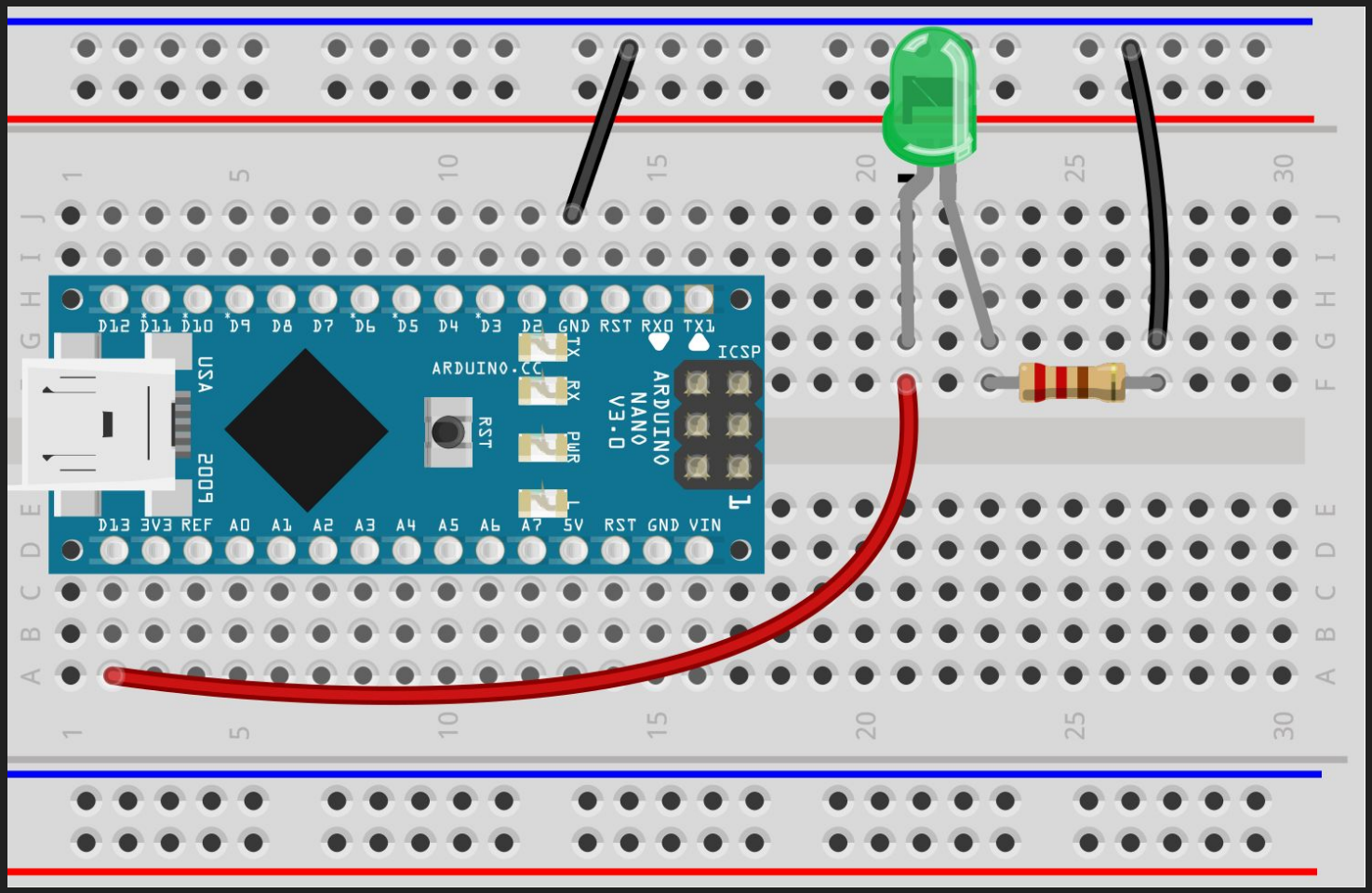
Utilizar macros para fazer o mapeamento dos pinos. Alternativamente, podem ser declarados como variáveis globais. (const int led = 13)

```
pinMode(LED, OUTPUT);
```

Configurar o pino LED (13) para o modo output

```
digitalWrite(LED, HIGH);
```

Vamos montar o circuito!



Exercício 3 - LED e Serial

Serial é **Bidireccional!**

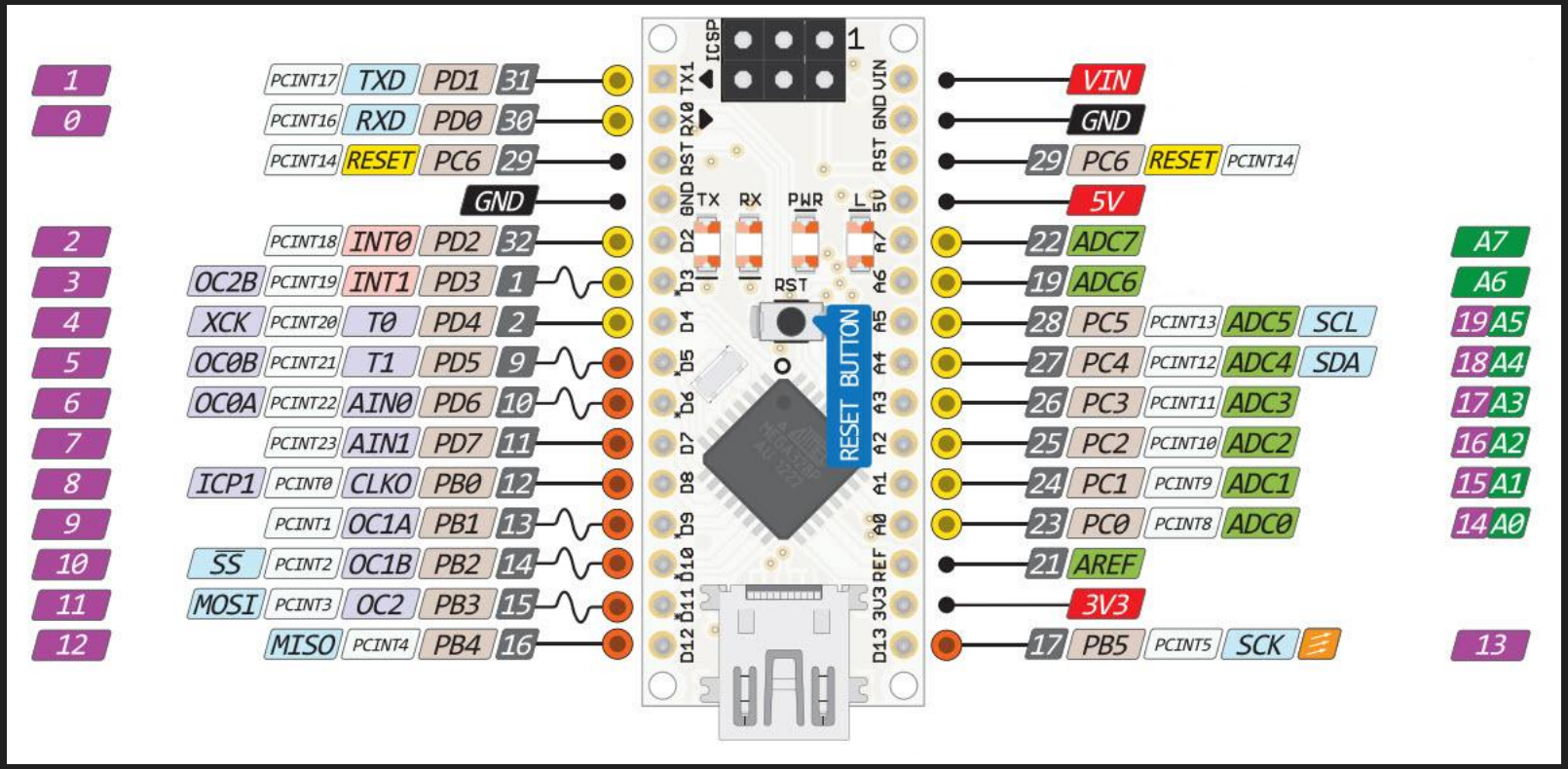
Vamos actualizar o programa anterior para acender o LED se for enviado um carácter via Serial.

```
char Serial.read()
```

Exercício 3 - LED e Serial

LIVE CODING!

Arduino Nano Pinout



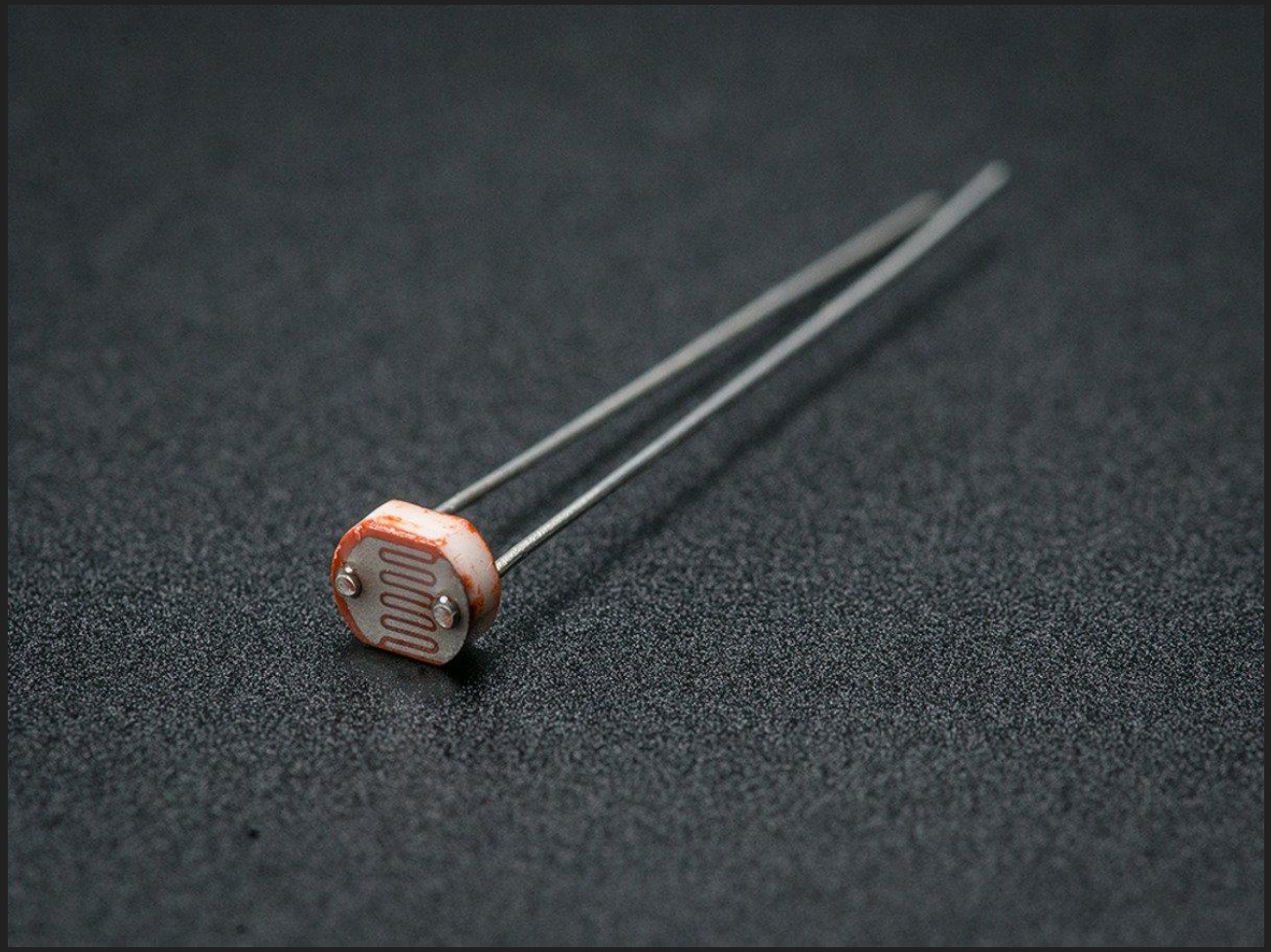
Exercício 4 - PWM LED

Agora vamos modificar o programa anterior, para mudar a **intensidade de brilho** de um LED.

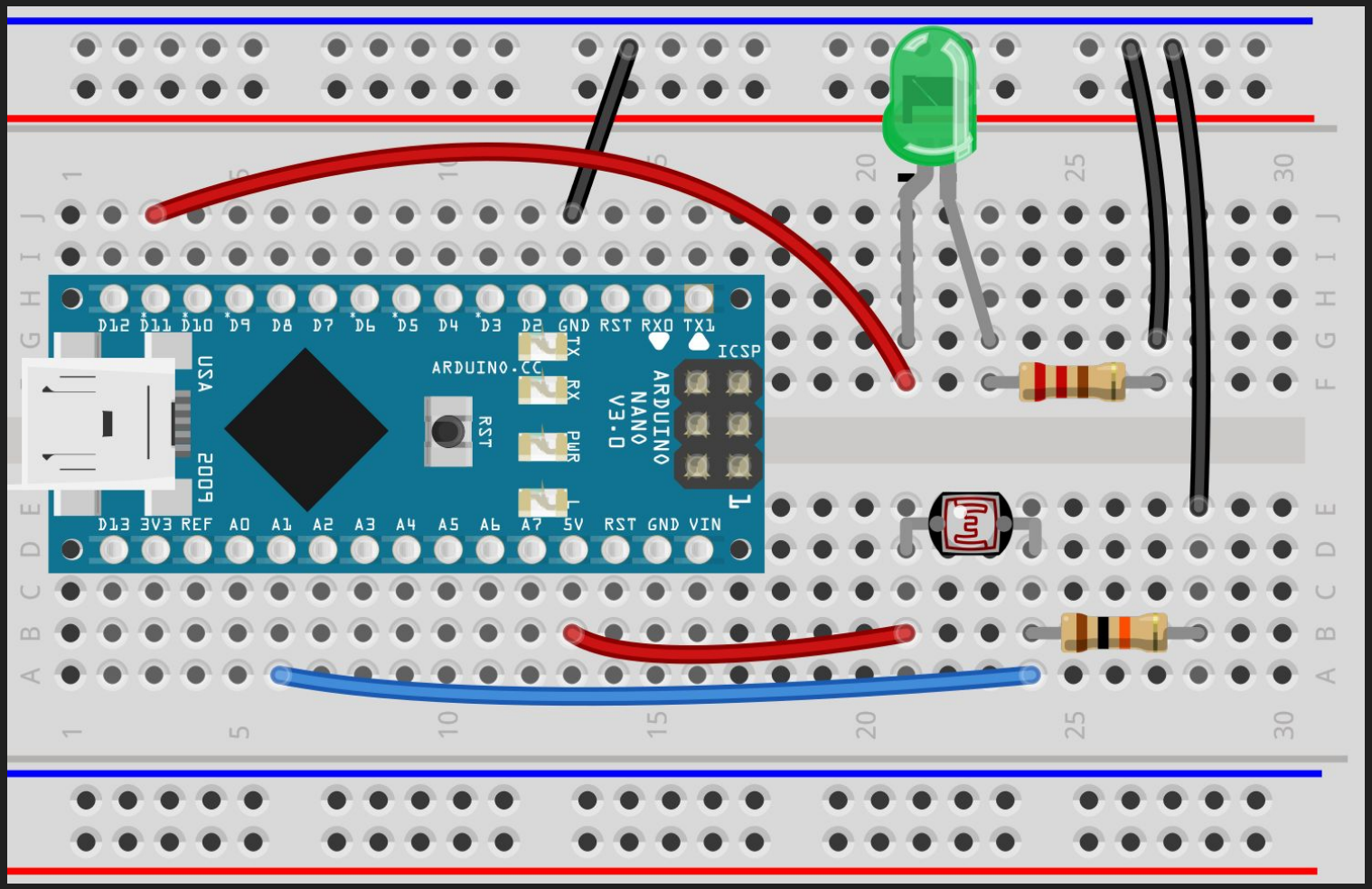
PWM - Pulse Width Modulation

Duty Cicle

Sensor LDR



Exercício 5 - Esquema



Exercício 5 - Sensor LDR e Serial Monitor

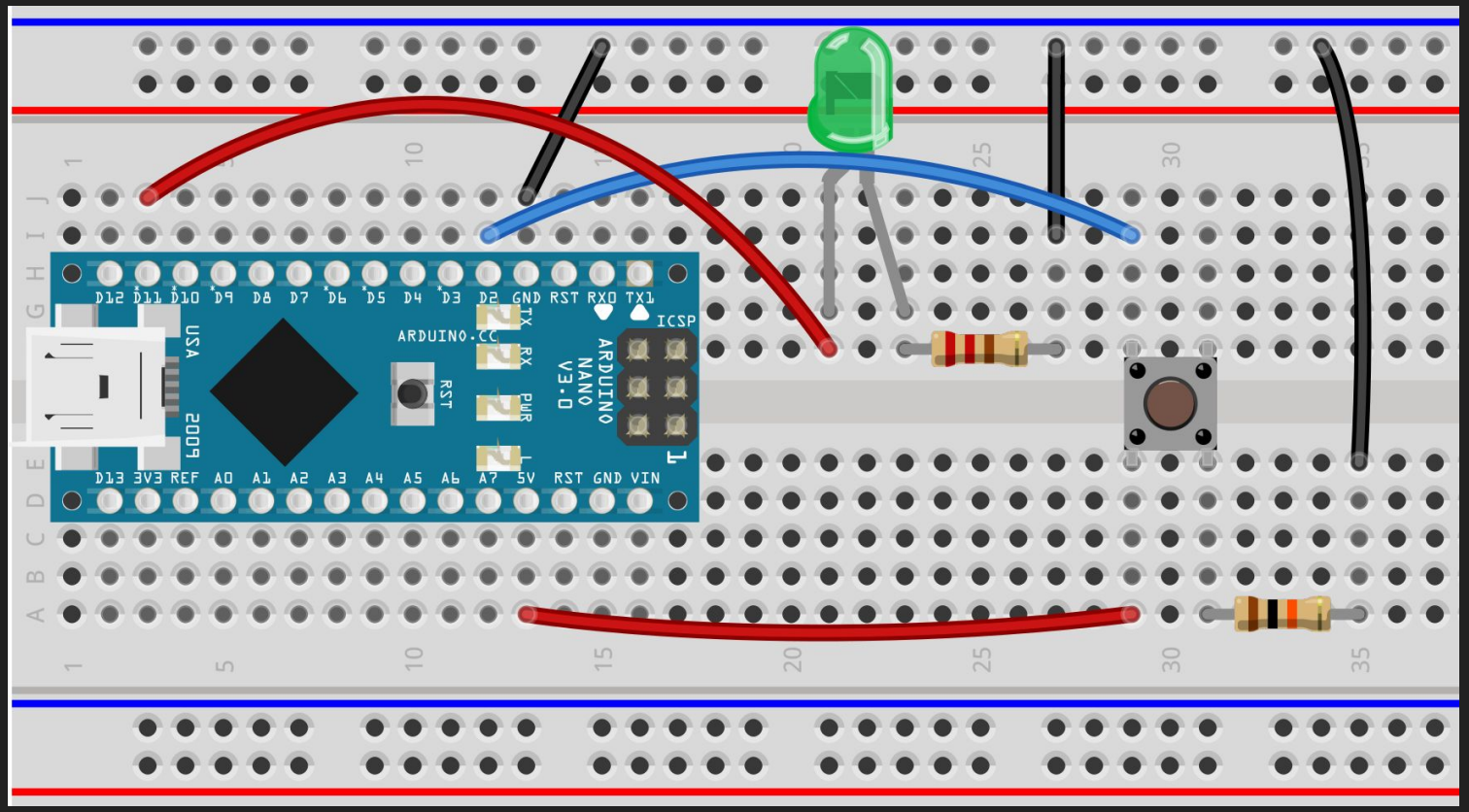
Agora, vamos tornar possível a leitura da entrada e da saída no Serial Monitor

Exercício 6 - PWM Led controlado por sensor

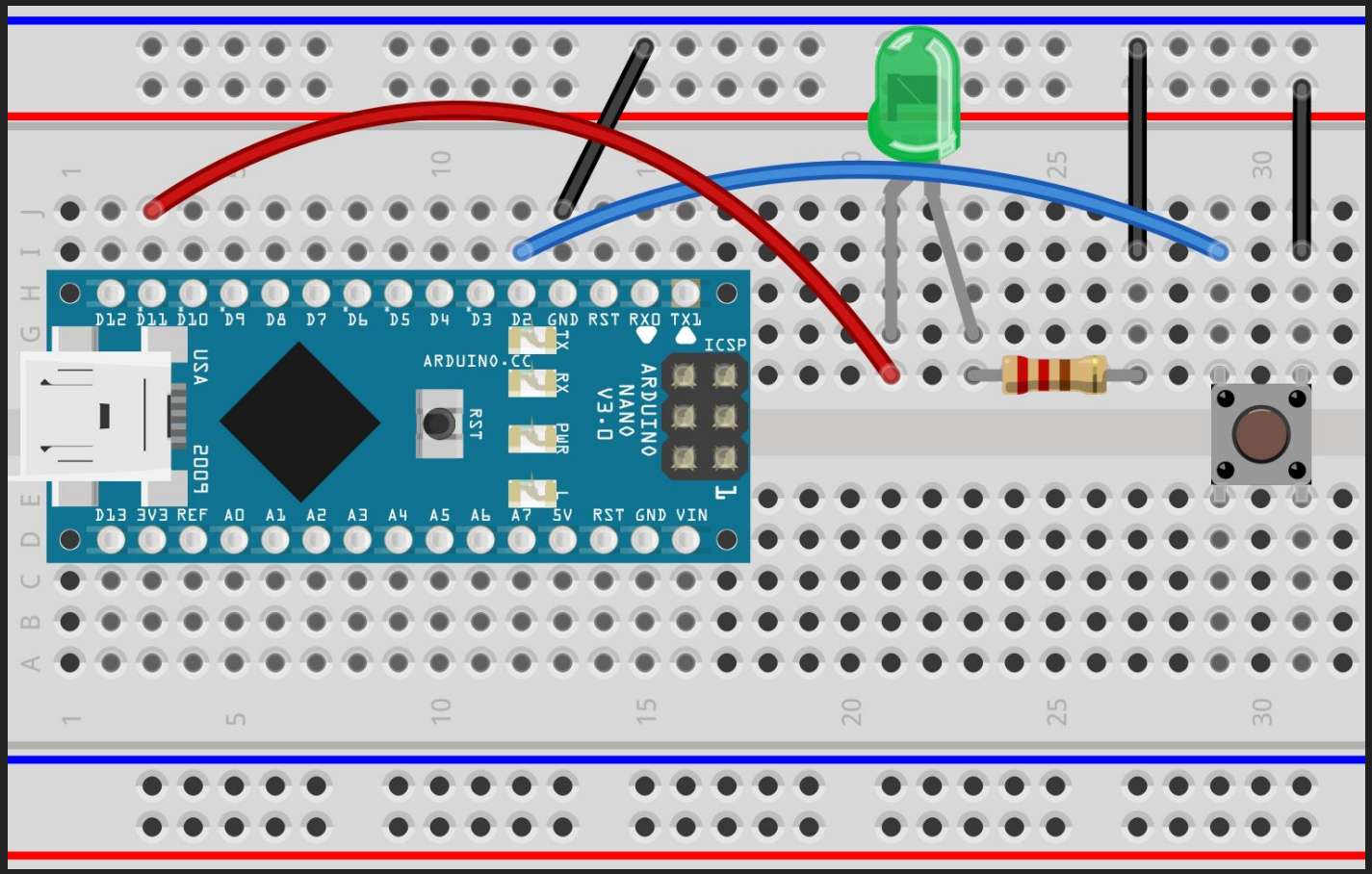
Por final, vamos controlar o LED com o valor lido pelo LDR.

```
output = map(input_var, in_min, in_max, out_min, out_max);
```

Exercício 7 - Push Button



Exercício 7a - Pullup interno



Interrupções

Interrupções

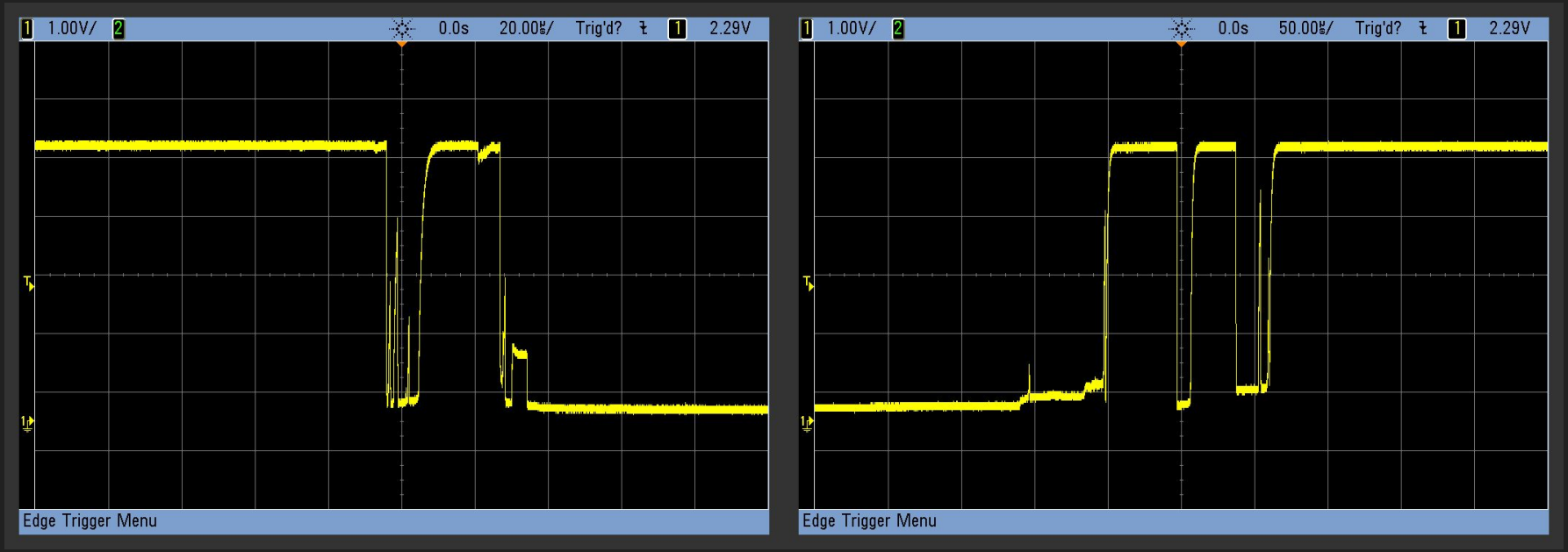
```
// Configuração do valor do tempo (15ms)
TCCR2A = 0;
TCCR2B = 1<<CS22 | 1<<CS21 | 1<<CS20;
// Máscara Int - Timer2 Overflow Interrupt
TIMSK2 |= 1<<TOIE2;
```

Interrupções

```
// Comportamento perante interrupção
SIGNAL(TIMER2_OVF_vect) {
    my_function();
}
```

Exercício 8 - Push Button com Interrupções

Exercício 9 - Debounce



Exercício 10 - Código Final

Por final, basta apenas acrescentar os botões restantes.

Isto será conseguido através do uso de vectores (arrays), e as alterações a serem feitas ao programa serão mínimas