

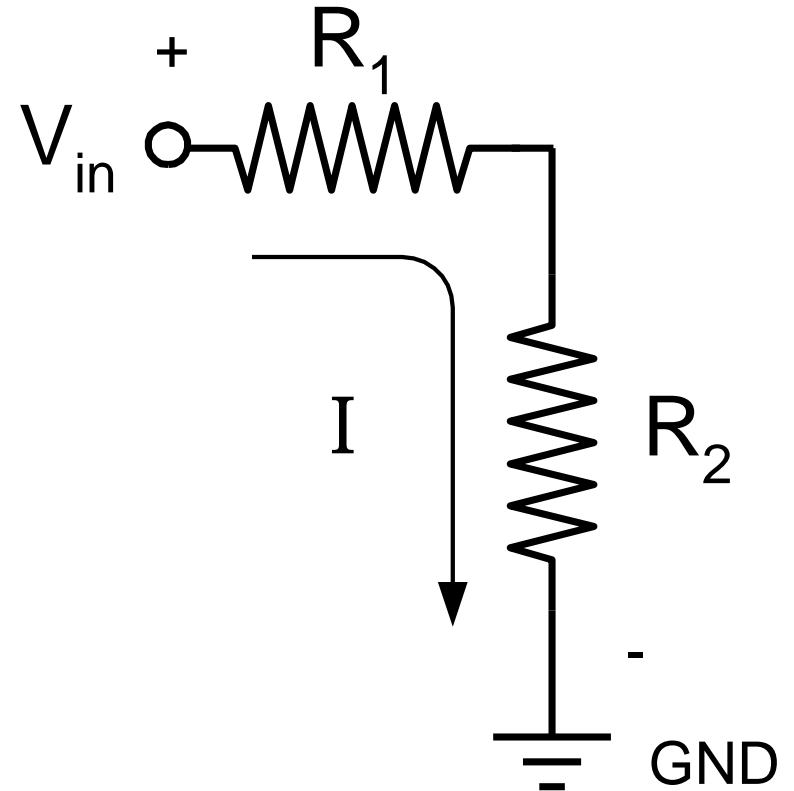
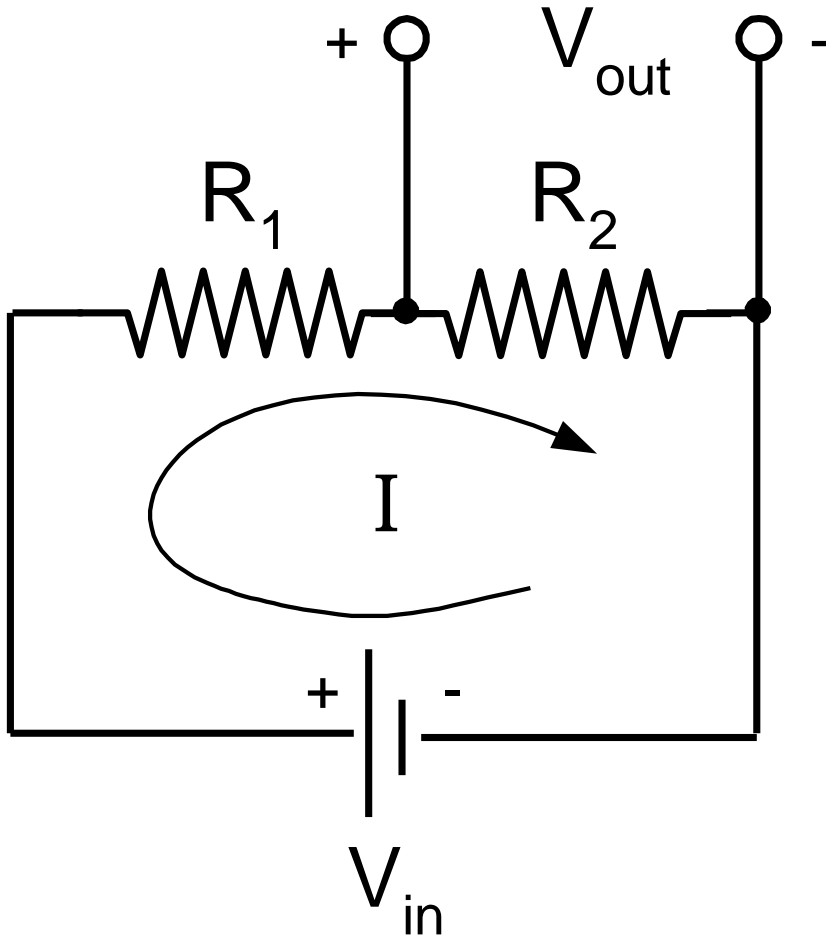
Arduino

Parte 2

Circuitos com resistores variáveis

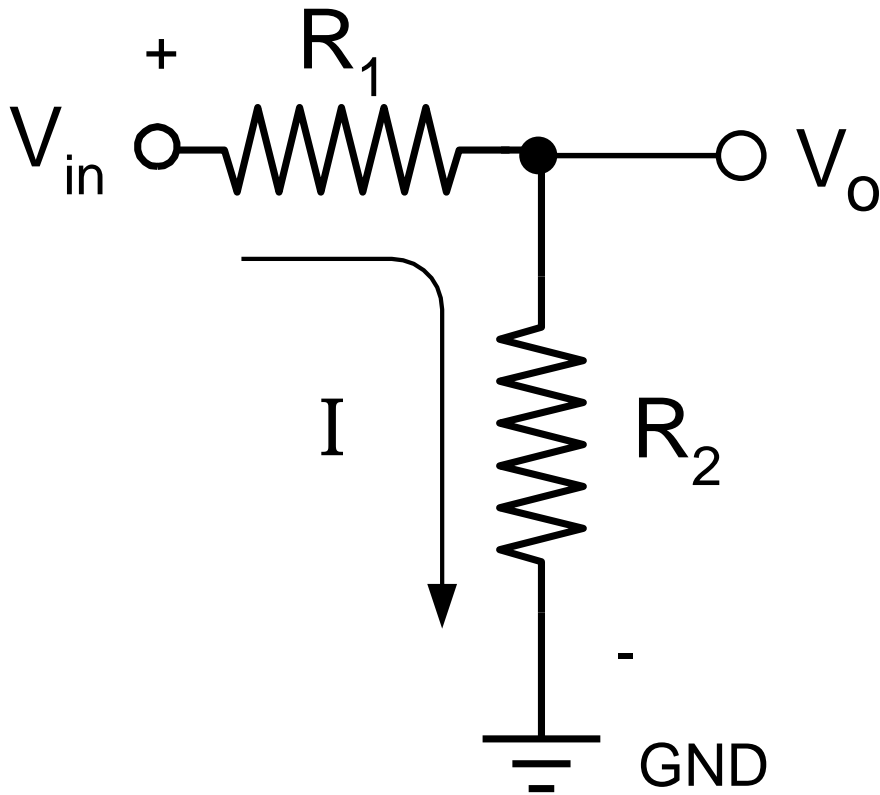
Circuitos com resistência elétrica

Circuito divisor de tensão resistivo

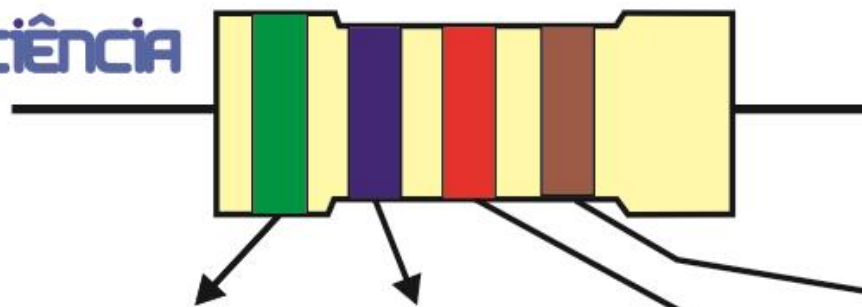


Circuito equivalente

Circuito divisor de tensão resistivo



$$\frac{V_o}{V_i} = \frac{R_2}{R_1 + R_2}$$



$R = 5600 \Omega, \pm 1\%$
 $R = 5,6 \cdot 10^3 \Omega$
 $R = 5,6 \text{ K}\Omega$

COR **1ª Faixa (Número)** **2ª Faixa (Número)** **3ª Faixa (zeros ou Pot. de 10)** **4ª Faixa (Tolerância)**

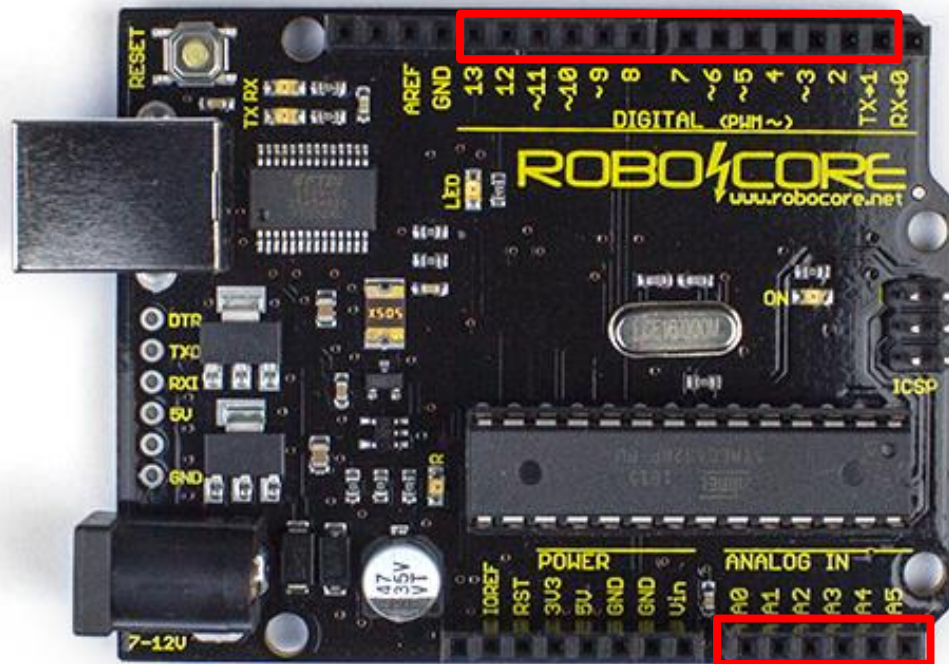
Preto	—	0	—	—
Marrom	1	1	0 (x 10 ¹)	1%
Vermelho	2	2	00 (x 10 ²)	2%
Laranja	3	3	000 (x 10 ³)	—
Amarelo	4	4	0000 (x 10 ⁴)	—
Verde	5	5	00000 (x 10 ⁵)	—
Azul	6	6	000000 (x 10 ⁶)	—
Violeta	7	7	—	—
Cinza	8	8	—	—
Branco	9	9	—	—
Ouro	—	—	x 0,1 (x 10 ⁻¹)	5%
Prata	—	—	x 0,01 (x 10 ⁻²)	10%

Código de cores dos resistores

Portas do Arduino

- Portas digitais: D1 a D13
- Portas digitais PWM: ~3, ~5, ~6, ~9, ~10 e ~11
- Portas analógicas: A0 a A5

Portas digitais



Portas analógicas

Saída digital

O sinal digital possui somente dois valores:

0 V (**LOW**)

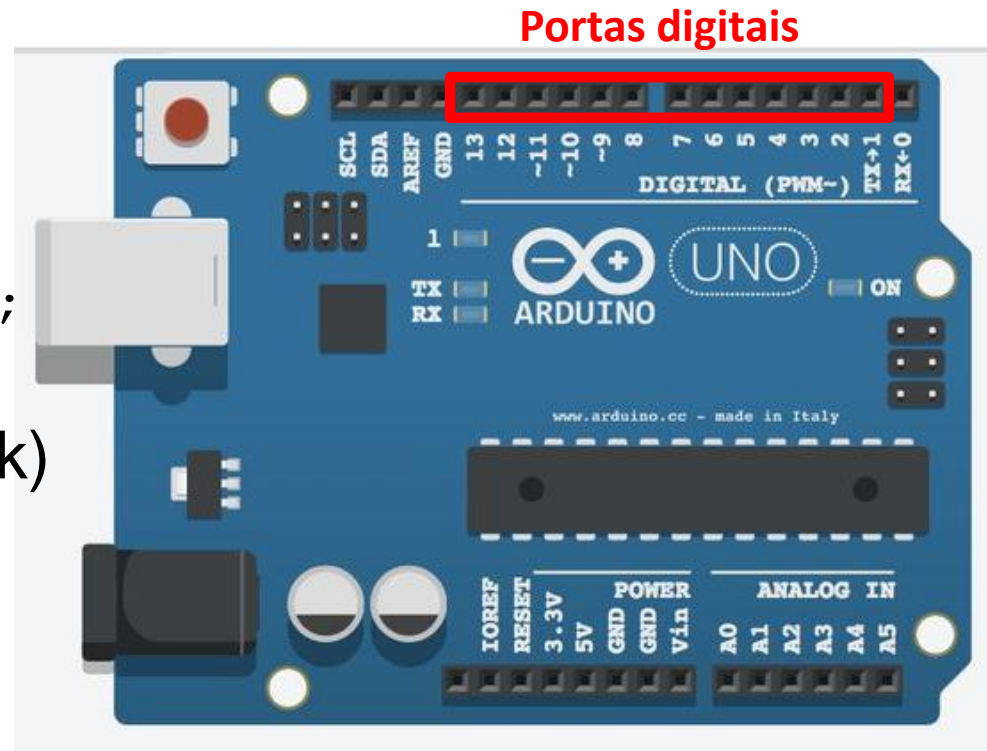
5 V (**HIGH**)

Portas digitais: 1 a 13

Sintaxe:

```
digitalWrite(Pino, valor);
```

Prática: LED piscante (blink)



Saída digital PWM (quasi-analógica)

PWM (*Pulse Width Modulation*): modulação por largura de pulso

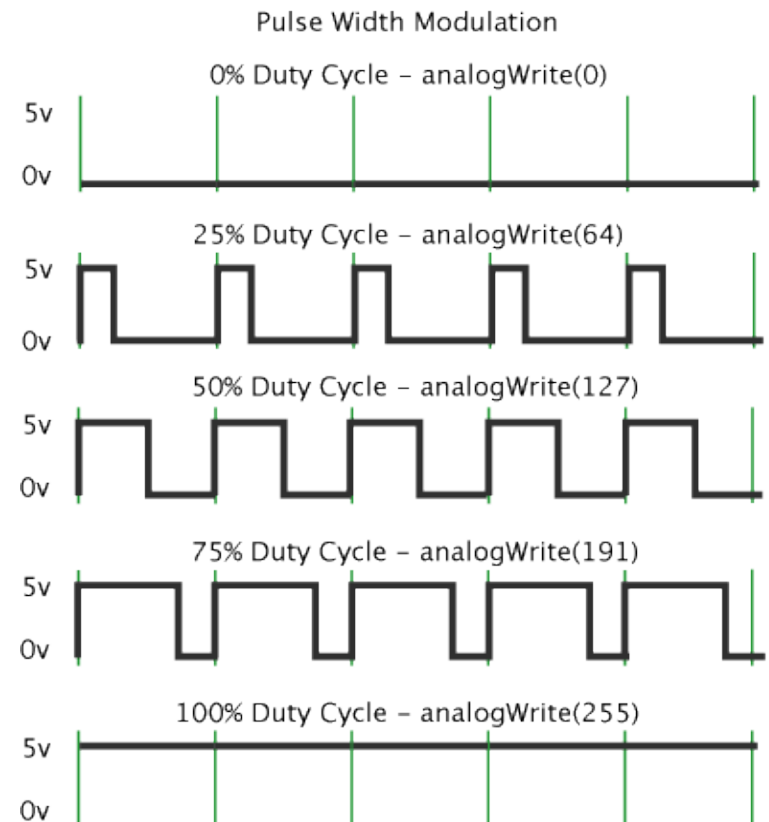
- ▣ É uma maneira de gerar saída analógica simples a partir de uma porta digital
- ▣ 8 bits de resolução
- ▣ 256 possibilidades (2^8): 0 a 255

Sintaxe:

```
analogWrite(ledPin, valor);
```

Prática: LED que pisca

- ▣ gradualmente



Entrada analógica

Entradas analógicas: convertem uma tensão de 0 a 5 volts para um valor proporcional de 0 a 1023 (10 bits)

- Sensor de temperatura, divisores de tensão, fotodiodo (sensor de luz)
- Questão: qual a menor variação de tensão detectável pelo Arduino?

Utilização das portas analógicas:

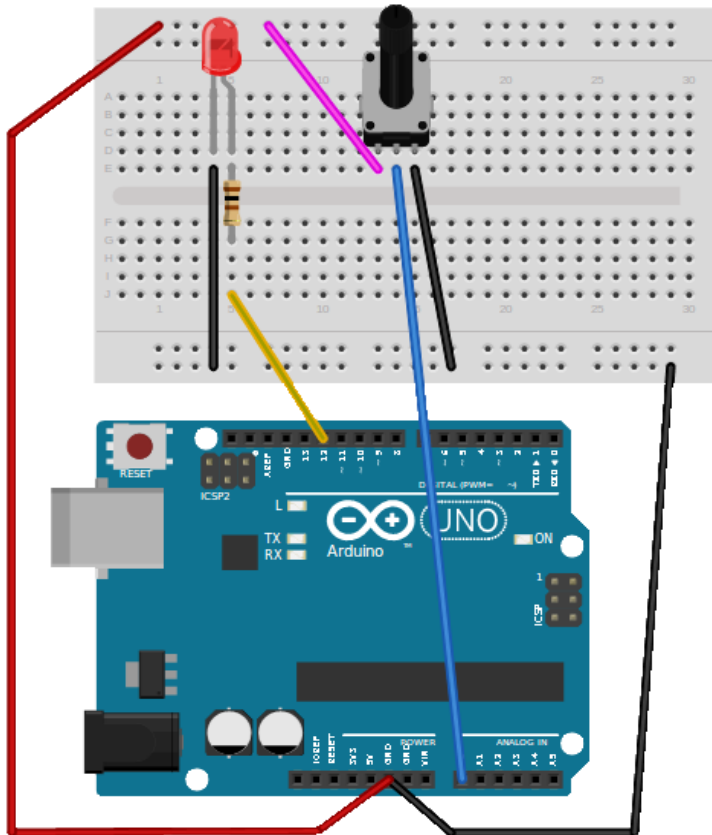
- A leitura pode ser gravada em uma variável:
- `int valor_sensor = analogRead(A0);`

Projeto: Controle pisca LED por PWM

❏ Materiais:

- ❏ Potenciômetro
- ❏ LED
- ❏ Resistor 300R

❏ Montagem:



❏ Código:

```
#define pinLED    12    // pino digital
#define pinPot    0    // pino analogico

int valor_pot;

void setup()
{
    Serial.begin(9600);
    pinMode(pinLED, OUTPUT);
}

void loop()
{
    // retorna um valor entre 0 e 1023
    valor_pot = analogRead(pinPot);
    // manda p/ USB (ver com Monitor Serial)
    Serial.println(valor_pot);

    digitalWrite(pinLED, HIGH);
    delay(valor_pot);
    digitalWrite(pinLED, LOW);
    delay(valor_pot);
}
```

Controle intensidade do LED por PWM

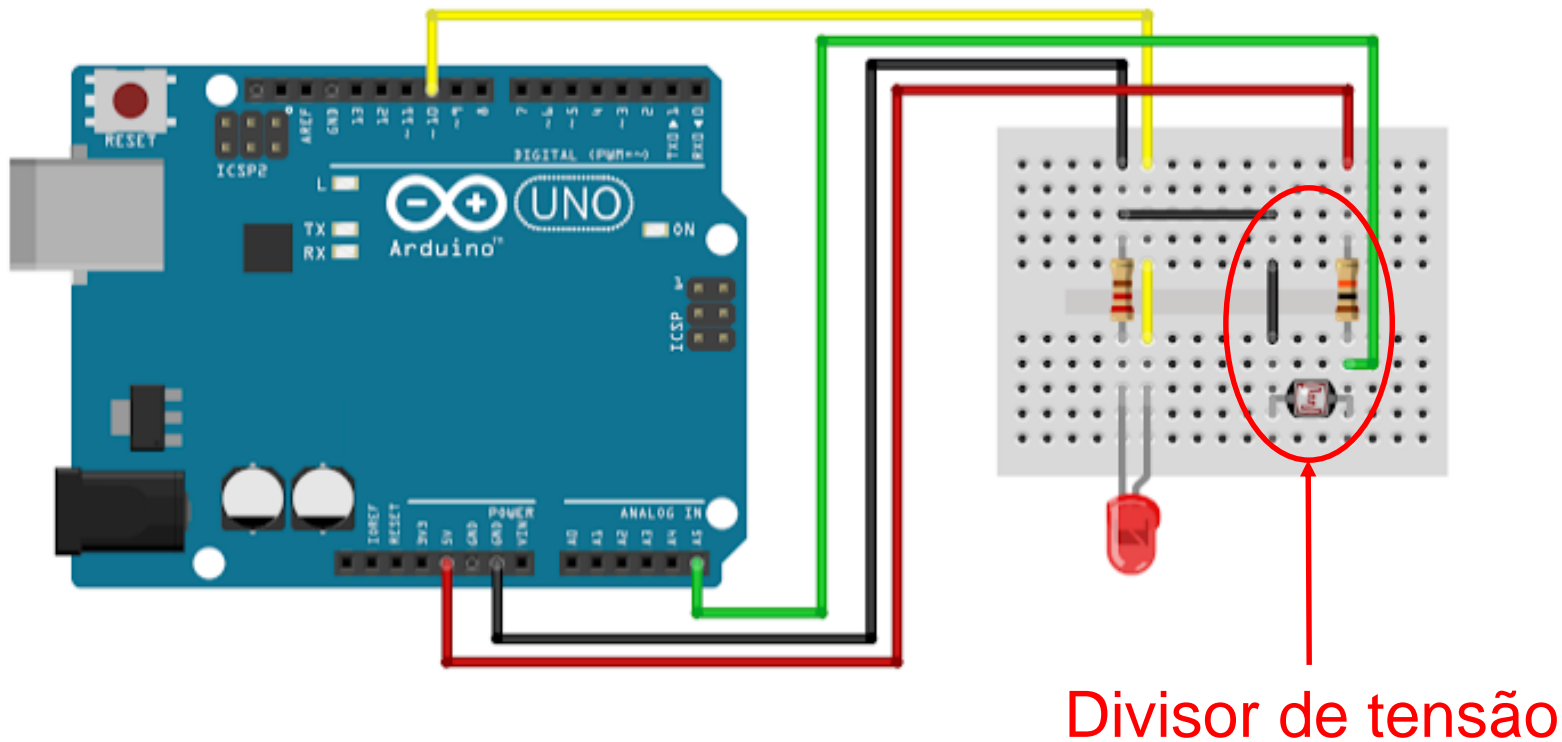
```
// ControlaLED

int pinoLED = 11;

void setup() {
  Serial.begin(9600);
  pinMode(pinoLED, OUTPUT);
}

void loop() {
  int valorPot = analogRead(A0);
  int pwm = map(valorPot, 0, 1023, 0, 255);
  analogWrite(pinoLED, pwm);
  Serial.print(valorPot); Serial.print(" ");
  Serial.println(pwm);
}
```

Controle do LED por LDR



Controle do LED por LDR

```
// Sensor_LDR
int pinoLED = 11;
int pinoLDR = A0;
void setup() {
    pinMode(pinoLED, OUTPUT);
}

void loop() {
    int valorPot = analogRead(pinoLDR);
    if (valorPot > 850) {
        digitalWrite(pinoLED, HIGH);
    } else {
        digitalWrite(pinoLED, LOW);
    }
}
```