

Entradas Analógicas no Arduino

Prof. Dr. Sérgio Carlos
Portari Júnior

Revisão rápida



- Até agora aprendemos:
 - LED piscando (saída digital)
 - Controlar LED (entrada digital)

Entradas analógicas



- Diferente da entrada digital (0 ou 1), a entrada analógica permite:
 - Ler valores contínuos (0 a 1023 no Arduino)
 - Trabalhar com sensores que variam intensidade (luz, temperatura, etc.)

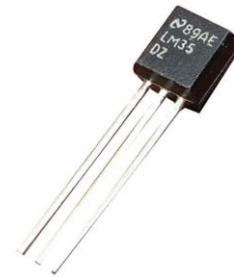
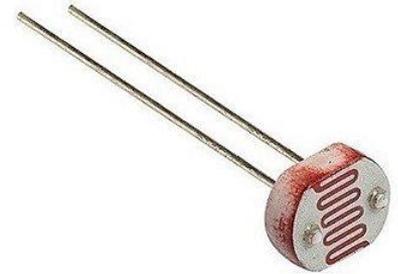
Conversão Analógico-Digital (ADC)

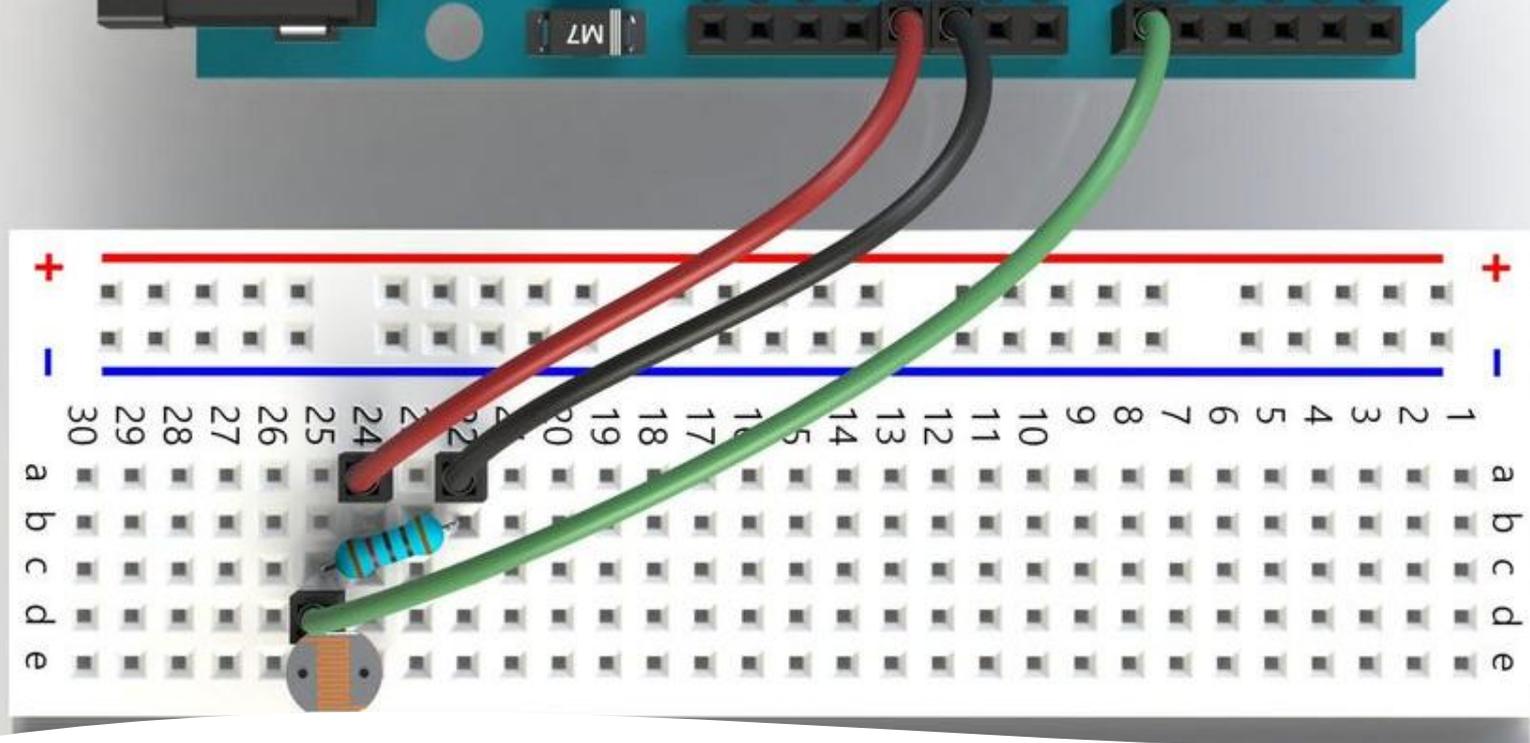


- O Arduino Uno possui um conversor A/D de 10 bits:
 - Converte tensões de 0V a 5V em valores de 0 a 1023
 - Exemplo: 2,5V \approx 512
 - Usado pela função **analogRead()**

Sensores analógicos comuns

- Exemplos de sensores que fornecem valores analógicos:
 - LDR – mede intensidade de luz
 - Potenciômetro – mede posição de um eixo
 - LM35 – mede temperatura



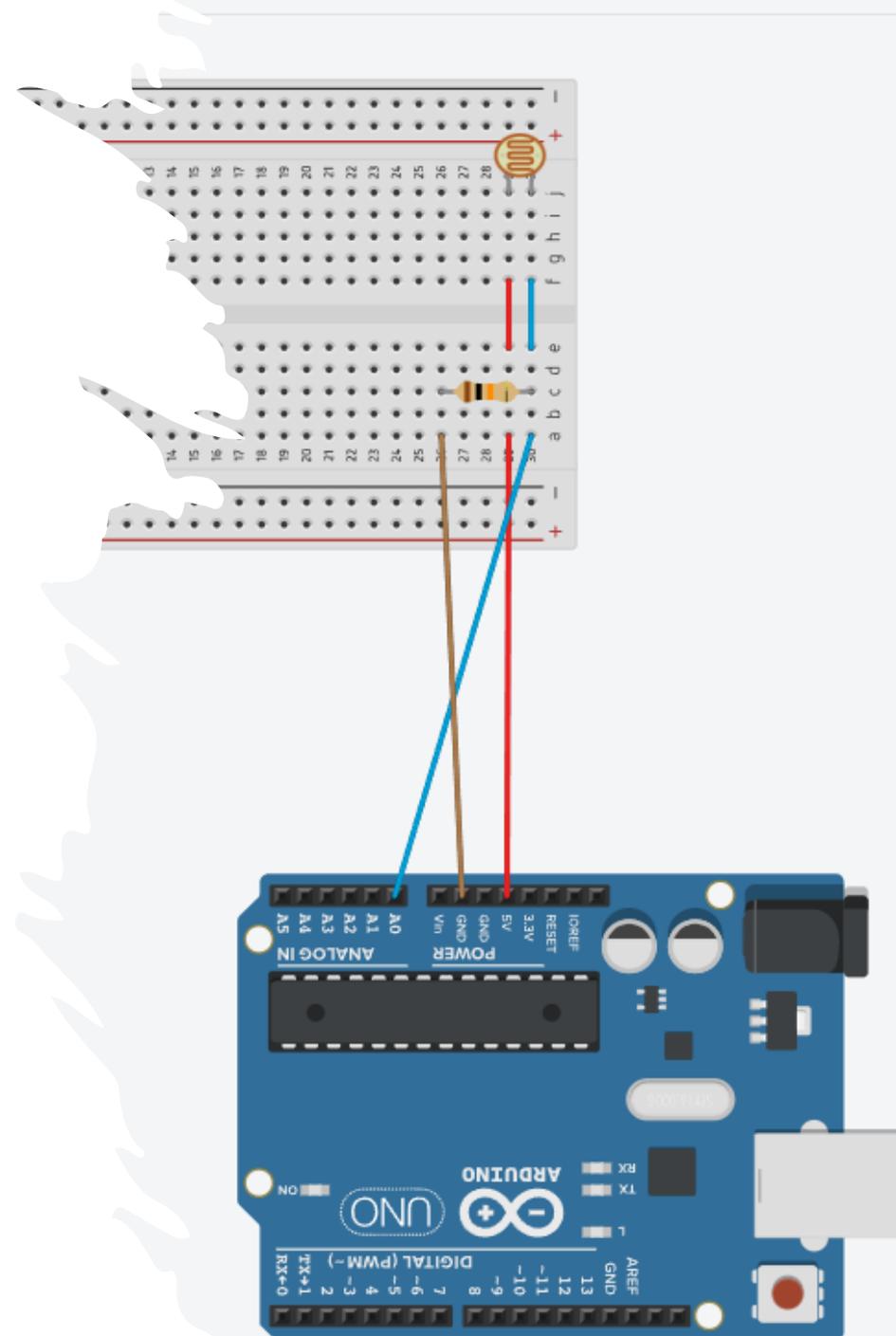


Montagem do LDR

- Para ler um LDR (Light Dependent Resistor) no Arduino, normalmente se monta um **divisor de tensão**. O esquema básico é o seguinte:
- Esquema de Montagem
 - Conecte um terminal do LDR ao 5V do Arduino.
 - Conecte o outro terminal do LDR a:
 - Um resistor fixo (ex.: 500 Ω), e
 - À entrada analógica do Arduino (ex.: A0).
 - O outro terminal do resistor fixo vai ao GND.
- Ou seja, o LDR e o resistor formam um divisor de tensão entre 5V e GND, e o ponto central vai para a entrada analógica.

Exemplo LDR

- Por que precisa ser feito assim?
 - O LDR sozinho não gera tensão
 - Ele só muda de resistência conforme a luminosidade:
 - baixa resistência na luz,
 - alta resistência no escuro.
- Como o Arduino lê tensão (0–5V) e não resistência, é necessário um circuito que converta essa variação de resistência em variação de tensão → daí o divisor de tensão.



Código para ler LDR

```
// Pino analógico onde o LDR está conectado  
const int LDR_PIN = A0;
```

```
void setup() {  
  Serial.begin(9600); // Inicia a comunicação serial  
}
```

```
void loop() {  
  // Faz a leitura analógica (0 a 1023)  
  int valorLDR = analogRead(LDR_PIN);
```

```
  // Mostra no Serial Monitor  
  Serial.print("Valor LDR: ");  
  Serial.println(valorLDR);
```

```
  delay(500); // Aguarda meio segundo  
}
```

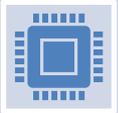
Comunicação Serial no Arduino

- **O que é:**
É a forma de comunicação entre o **Arduino** e o **computador** (ou outros dispositivos) enviando dados **bit a bit** através de um cabo USB ou pinos RX/TX.
- **Velocidade (baud rate):**
Define quantos bits são transmitidos por segundo.
Exemplo: `Serial.begin(9600);` → 9600 bits/s.
- **Uso principal em aula:**
 - Mostrar valores de sensores.
 - Depuração (debug).
 - Enviar comandos entre PC ↔ Arduino.

Serial.print e Serial.println

- `Serial.print(valor);`
→ Envia o valor para o computador, **na mesma linha**.
- `Serial.println(valor);`
→ Envia o valor e adiciona uma **quebra de linha** (pula para a próxima).

Funções novas



analogRead(A0) → lê intensidade de luz no LDR (0 a 1023).



Serial.println(ldrValue) → mostra o valor no Monitor Serial.

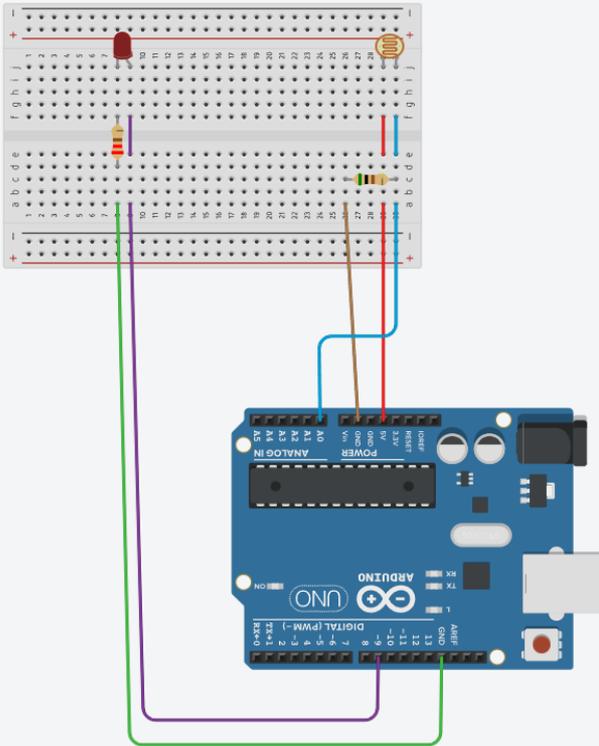


map() → converte a faixa de 0-1023 para 0-255 (escala de brilho do LED).



analogWrite(ledPin, brilho) → ajusta o brilho do LED com saída PWM.

Exemplo completo



```
// ===== Pinos =====
```

```
const int LDR_PIN = A0; // LDR no divisor de tensão -> ponto médio em A0
```

```
const int LED_PWM = 9; // LED no pino 9 (PWM)
```

```
// ===== Calibração (ajuste conforme sua montagem/ambiente) =====
```

```
const int LDR_MIN = 3; // leitura no "mínimo" de luz
```

```
const int LDR_MAX = 508; // leitura no "máximo" de luz
```

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

```
void loop() {  
  // 1) Ler o LDR  
  int leitura = analogRead(LDR_PIN);  
  
  // 2) Travar a leitura dentro da faixa calibrada  
  leitura = constrain(leitura, LDR_MIN, LDR_MAX);  
  
  // 3) Mapear para 0–255 (PWM). Quanto mais luz, maior brilho.  
  int brilho = map(leitura, LDR_MIN, LDR_MAX, 0, 255);  
  
  // 4) Aplicar PWM no LED  
  analogWrite(LED_PWM, brilho);  
  
  Serial.print("LDR: "); Serial.print(leitura);  
  Serial.print(" PWM: "); Serial.println(brilho);  
  delay(30); // pequeno debounce/estabilização  
}
```

Atividade prática

Inverter lógica (LED apaga quando claro)

Substituir LDR por potenciômetro

Adicionar dois leds, vermelho e verde; considerando 0-127 – acende o vermelho e considerando 128-255 – acende o verde.