

IoT: Robótica, Sensores, Atuadores e Parametrização

Objetivos da Aula

Apresentar conceitos fundamentais de robótica no contexto de IoT.

Aprender a escolher, ler, condicionar e calibrar sensores.

Conhecer atuadores e como controlá-los via microcontrolador.

Parametrização de robôs (ex.: tuning PID).

Exemplo prático: nó de distância + motor + telemetria MQTT.

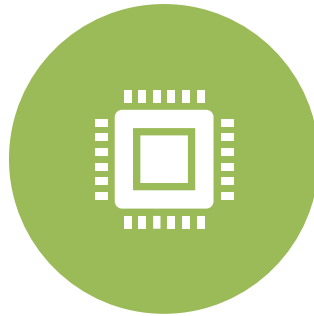
Agenda

1. Robótica em IoT — visão arquitetural
2. Sensores: definições, tipos, interfaces
3. Atuadores: definições, acionamento
4. Parametrização de robôs (PID, tuning)
5. Exemplo prático passo a passo
6. Exercícios e critérios de avaliação

Robótica dentro do ecossistema IoT



ARQUITETURA TÍPICA: NÓ-
DE-BORDA → GATEWAY →
CLOUD → APLICAÇÃO.



EXEMPLOS DE HARDWARE:
ESP32, STM32, ARDUINO.



PONTOS DE ATENÇÃO:
LATÊNCIA, CONSUMO,
SEGURANÇA, NORMAS.

Sensores — definição e cadeia de medição

Sensor: converte
grandeza física em
sinal elétrico.

Cadeia: transdutor →
condicionamento →
ADC/interface digital
→ processamento →
envio.

Tipos comuns de sensores e aplicações

- Proximidade/distância: Ultrassom (HC-SR04), LiDAR — evitar colisões.
- Temperatura/umidade: BME280, DHT22, PT100 — monitoramento térmico.
- IMU: MPU6050 — orientação e estabilização.
- Encoder: odometria/posição.
- 4-20mA, células de carga — aplicações industriais.

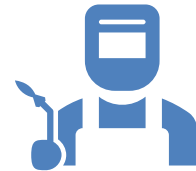
Interfaces elétricas e protocolos



Analógico (ADC): requisitos de condicionamento.



I2C / SPI / UART / 1-Wire: sensores digitais.



Industrial: 4-20mA, Modbus, CAN. Atenção a pull-ups, isolamento.

Exemplo prático — HC-SR04 (ultrassom)

- Ligação: Vcc 5V, GND, TRIG → pino digital, ECHO → pino digital.
- Conversão: tempo de ida-volta → distância = $(\text{tempo}/2) * \text{velocidade do som}$.

Código (HC-SR04) — Exemplo Arduino

- ```
// HC-SR04 Arduino (exemplo didático)
const int trigPin = 5;
const int echoPin = 6;
long duration;
float distance_cm;

void setup() {
 Serial.begin(9600);
 pinMode(trigPin, OUTPUT);
 pinMode(echoPin, INPUT);
}

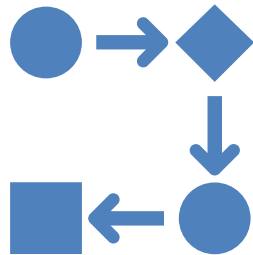
void loop() {
 digitalWrite(trigPin, LOW);
 delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin, LOW);
 duration = pulseIn(echoPin, HIGH);
 distance_cm = (duration / 2.0) *
0.0343;
 Serial.print("Distância: ");
 Serial.print(distance_cm);
 Serial.println(" cm");
 delay(200);
}
```

ADC:  
resolução e  
exemplo  
numérico

Fórmula:  $LSB = V_{ref} / (2^N - 1)$ .

Exemplo: ADC 10 bits,  
 $V_{ref}=5V \rightarrow LSB \approx 4.887$   
mV. Interpretar  
contagens.

# Calibração e condicionamento



Passos: verificar linearidade, ajustar offset e ganho, medir desvio padrão.



Condicionamento: filtros RC, amplificadores, proteção (TVS, diodos).

# Atuadores — definições e tipos

Motores DC, Stepper,  
Servomotor,  
Relés/Contatores,  
Atuadores lineares.

Controle: PWM, H-  
bridge, sinais  
analógicos/4-20mA.

# Exemplo prático — motor DC + H-bridge

Componentes: motor DC,  
driver (L298/DRV8833),  
fonte externa, MCU.

Proteções: diodos de  
flyback, desacoplamento,  
aterramento. Código  
PWM no slide.

## Parametrização de robôs — conceitos

Parâmetros cinemáticos:  
comprimentos, offsets,  
limites.

Parâmetros dinâmicos:  
massa, inércia, atrito.

Parâmetros de controle:  
ganhos PID, anti-windup,  
filtros.

# Tuning PID — método manual (passo a passo)

- 1)  $K_i=0$ ;  $K_d=0$ .
- 2) Aumentar  $K_p$  até oscilação leve.
- 3) Diminuir  $K_p$  para metade.
- 4) Ajustar  $K_i$  para eliminar erro em regime.
- 5) Ajustar  $K_d$  para reduzir overshoot.

Exemplo:  
controle de  
velocidade  
com encoder  
(pseudocódigo)

---

1. Ler encoder em intervalo  $T \rightarrow$  calcular velocidade.

---

2.  $\text{erro} = \text{setpoint} - \text{velocidade}$

---

3.  $\text{integral} += \text{erro} * T$ ;  $\text{derivativo} = (\text{erro} - \text{erro\_anterior})/T$

---

4.  $u = K_p * \text{erro} + K_i * \text{integral} + K_d * \text{derivativo} \rightarrow$  aplicar PWM.



Integração  
IoT:  
telemetria e  
controle  
remoto

---

Fluxo: MCU → process local → publicar MQTT → broker → dashboard → comandos de retorno.

---

Mensagens: timestamp UTC, ID do nó, leituras brutas e calibradas.  
Segurança: TLS, autenticação.

# Considerações industriais (coleta de dados)

4-20mA para longas distâncias e ambiente ruidoso.

Isolamento galvânico, gateways Modbus → MQTT, dimensionar ciclo de amostragem.

Boas  
práticas e  
critérios de  
qualidade

---

Documentar  
esquema, firmware,  
pinout.

---

---

Testes unitários e de  
integração,  
segurança,  
usabilidade.