

# 001 – INTRODUÇÃO AOS MICROCONTROLADORES

Programação em seu nível mais alto e voltada para a prática.

# CONCEITOS:

## Automação e Controle

- As experimentações efetuadas com um projeto eletromecânico devem levar a uma configuração final totalmente automatizada
- As leituras dos parâmetros de entrada e saída e seu armazenamento também devem ser automatizadas, para descarregamento em ferramenta de análise operacional e gerencial

# CONCEITOS:

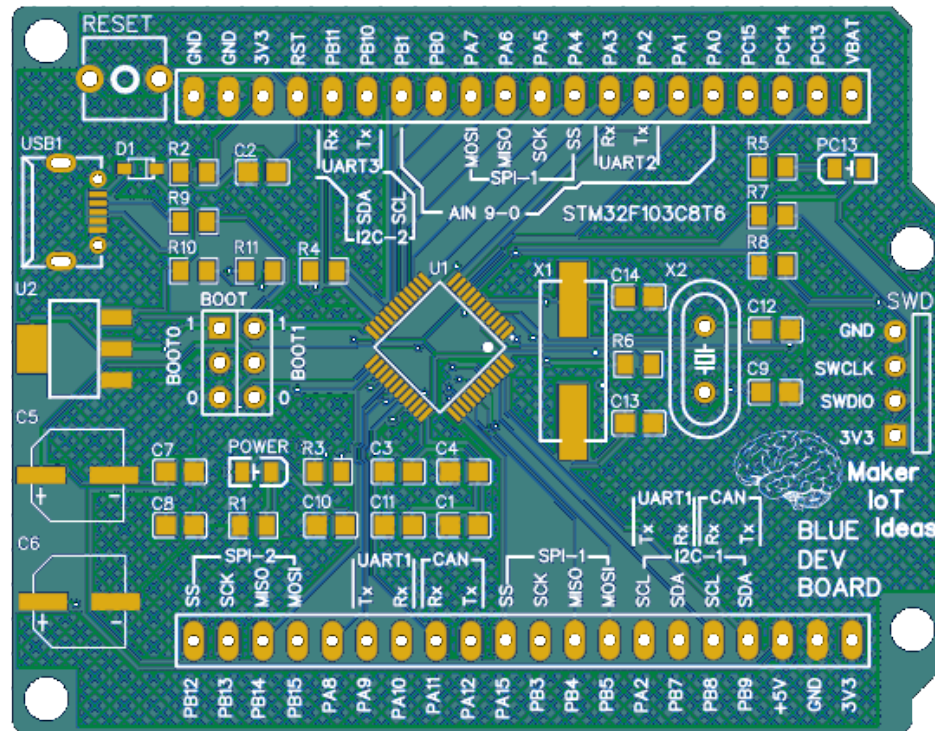
## O que automatizar ?

- **Processos perigosos para o ser humano**
- **Processos repetitivos**
- **Processos que exigem precisão**
- **Processos com muitas etapas**
- **Processos que envolvem produtos personalizados para os clientes**

# CONCEITOS:



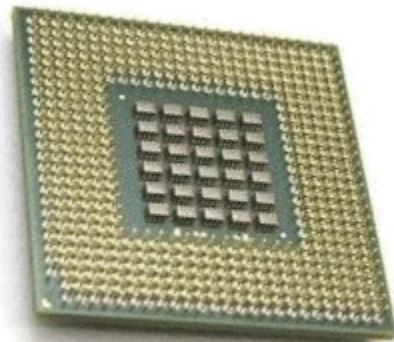
## PCB – Printed Circuit Board



# CONCEITOS:



## CPU – Central Processing Unit

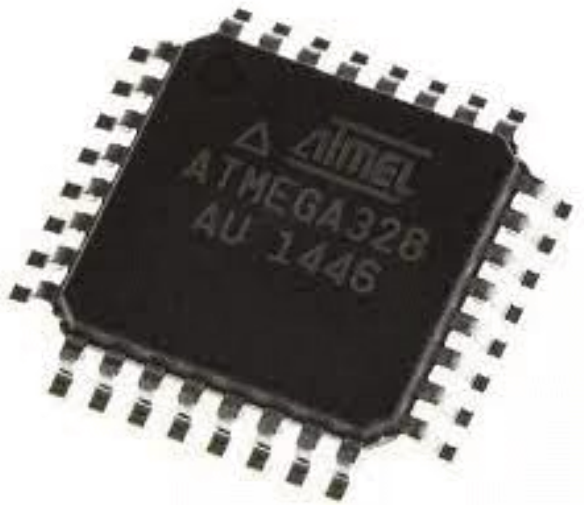


- Velocidades no patamar de GHz
- Necessitam de SO para operar
- Consomem muita energia e precisam de resfriamento
- Conectam-se a vários tipos de dispositivos, incluindo os de alta velocidade



# CONCEITOS:

## Microcontrolador



- Velocidades no patamar de MHz
- Não necessitam de SO para operar
- Consomem pouca energia, assim não precisam de resfriamento
- Requer menos componentes adicionais
- Seu uso é restrito a aplicações específicas



# CONCEITOS:

## Oscilador



- **Controla o passo temporal da sucessão de ações provocadas pelas instruções do microcontrolador ou microprocessador**
- **Possui uma frequência nominal**
- **Constituído de um cristal de quartzo e de um circuito divisor de frequência**

# CONCEITOS:



## LEDs



- Sinaliza um estado dos dispositivos
- Pode ser observado à distância
- Demonstra estados estáticos e dinâmicos
- Varia a natureza da informação de acordo com a cor

# CONCEITOS:

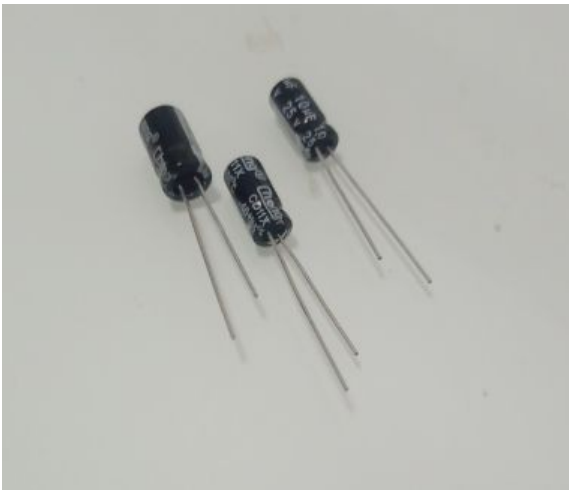
## Potenciômetro



- Equivale a uma resistência variável
- Possibilita a experimentação dinâmica do potencial e propriedades de um circuito
- Provê o conceito de regulagem, ajuste e equilíbrio das características de um projeto, para uma etapa posterior de automação

# CONCEITOS:

## Capacitor Eletrolítico



- **Seleciona o tipo de corrente alternada nos nós adjacentes do circuito**
- **Provê o acúmulo e o descarregamento harmônico de cargas no circuito**
- **Suaviza os impactos de variações de corrente**

# CONCEITOS:

## Push button



- Provê o corte absoluto de corrente em um trecho do circuito
- Possibilita a experimentação de parâmetros críticos como correntes e temperatura dos componentes
- Possibilita a experimentação do balanço do circuito, quando ainda não se fez a completa automação

# CONCEITOS:



## Servo motor



- Provê o controle de ajuste fino para o sincronismo entre os passos de um processo mecânico
- Fornece a precisão observada nos projetos de automação
- Possibilita aquilo que compreendemos como robótica
- Característica mais marcante da automação industrial

# CONCEITOS:

## Resistor



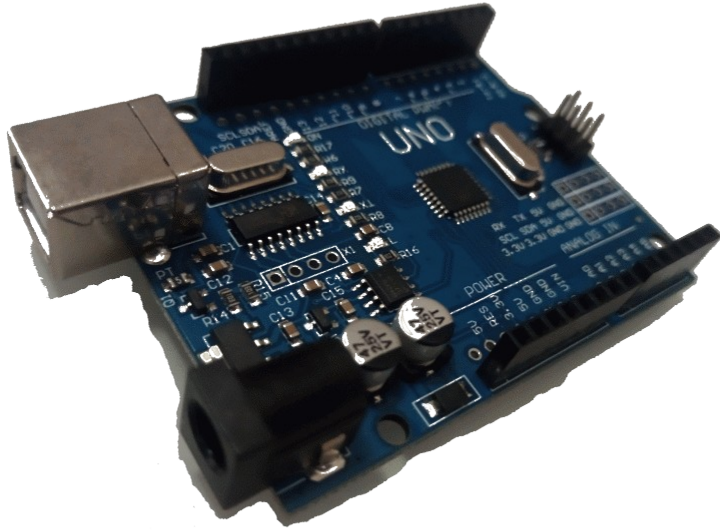
- **Balanceia potencial e corrente nos circuitos eletrônicos**
- **Atenua a corrente sobre um componente, reduzindo o seu aquecimento e, conseqüentemente, preservando-o do desgaste e dos conseqüentes defeitos**



# Aplicação dos Microcontroladores:

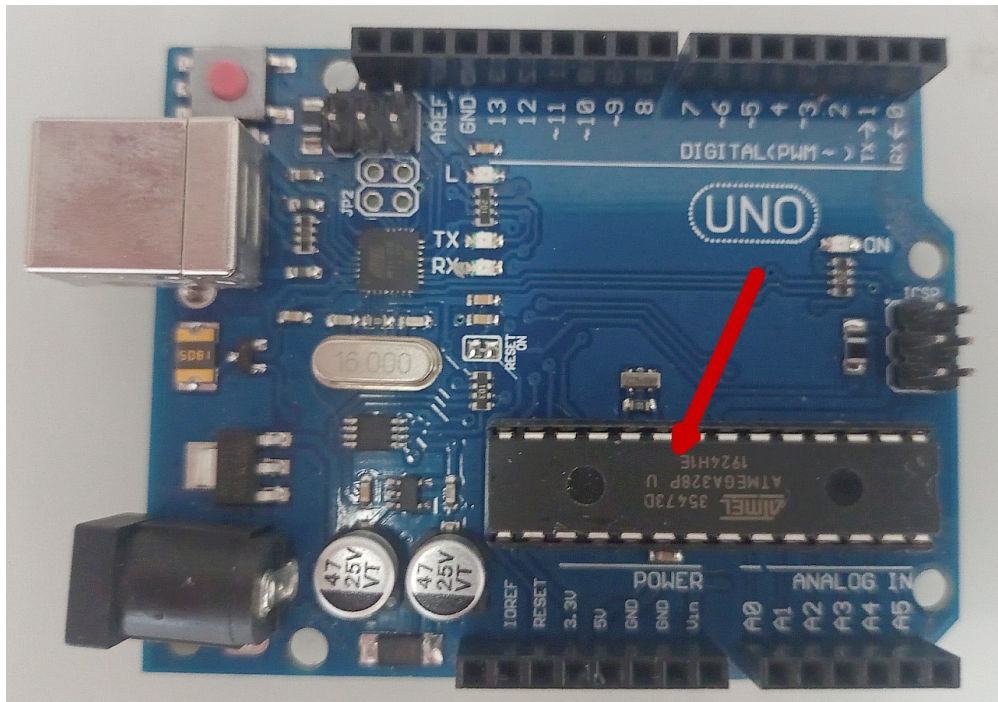


- Controle de acesso à empresa e departamentos
- Controle de temperatura (INDÚSTRIA)
- Controle de dispositivos com servomotores
- Controle de drones (CIVIL e MILITAR)
- Controle remoto
- Controle de eletrônicos automotivos
- Controle dos HDs de computadores
- Controle de câmeras de vigilância
- Robótica

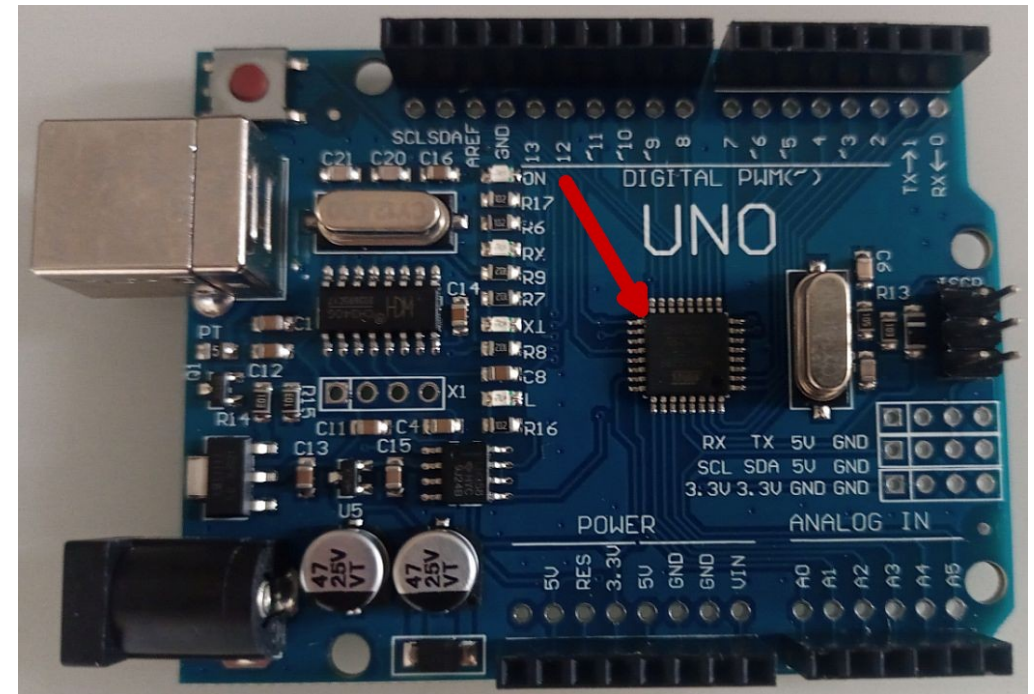


## 002 – PCBs Arduino no Mercado e componentes

O Arduino é um só, mas são muitos.



**Italiano legítimo(DIP-28)**



**Cópia chinesa (TQFP-32)**

O legítimo já é reconhecido automaticamente por PnP.  
A cópia necessita da instalação do driver CH340.

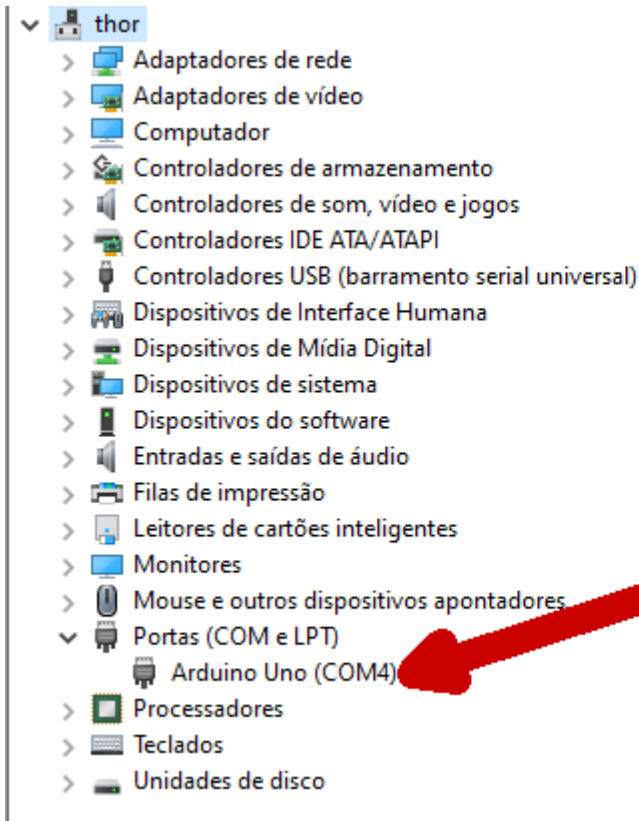
# Gerenciador de dispositivos



- ▼ thor
  - > Adaptadores de rede
  - > Adaptadores de vídeo
  - > Computador
  - > Controladores de armazenamento
  - > Controladores de som, vídeo e jogos
  - > Controladores IDE ATA/ATAPI
  - > Controladores USB (barramento serial universal)
  - > Dispositivos de Interface Humana
  - > Dispositivos de Mídia Digital
  - > Dispositivos de sistema
  - > Dispositivos do software
  - > Entradas e saídas de áudio
  - > Filas de impressão
  - > Leitores de cartões inteligentes
  - > Monitores
  - > Mouse e outros dispositivos apontadores
  - ▼ Portas (COM e LPT)
    - USB-SERIAL CH340 (COM3)
  - > Processadores
  - > Teclados
  - > Unidades de disco

Gerenciador de dispositivos do Windows mostrando a cópia chinesa com driver USB instalado.

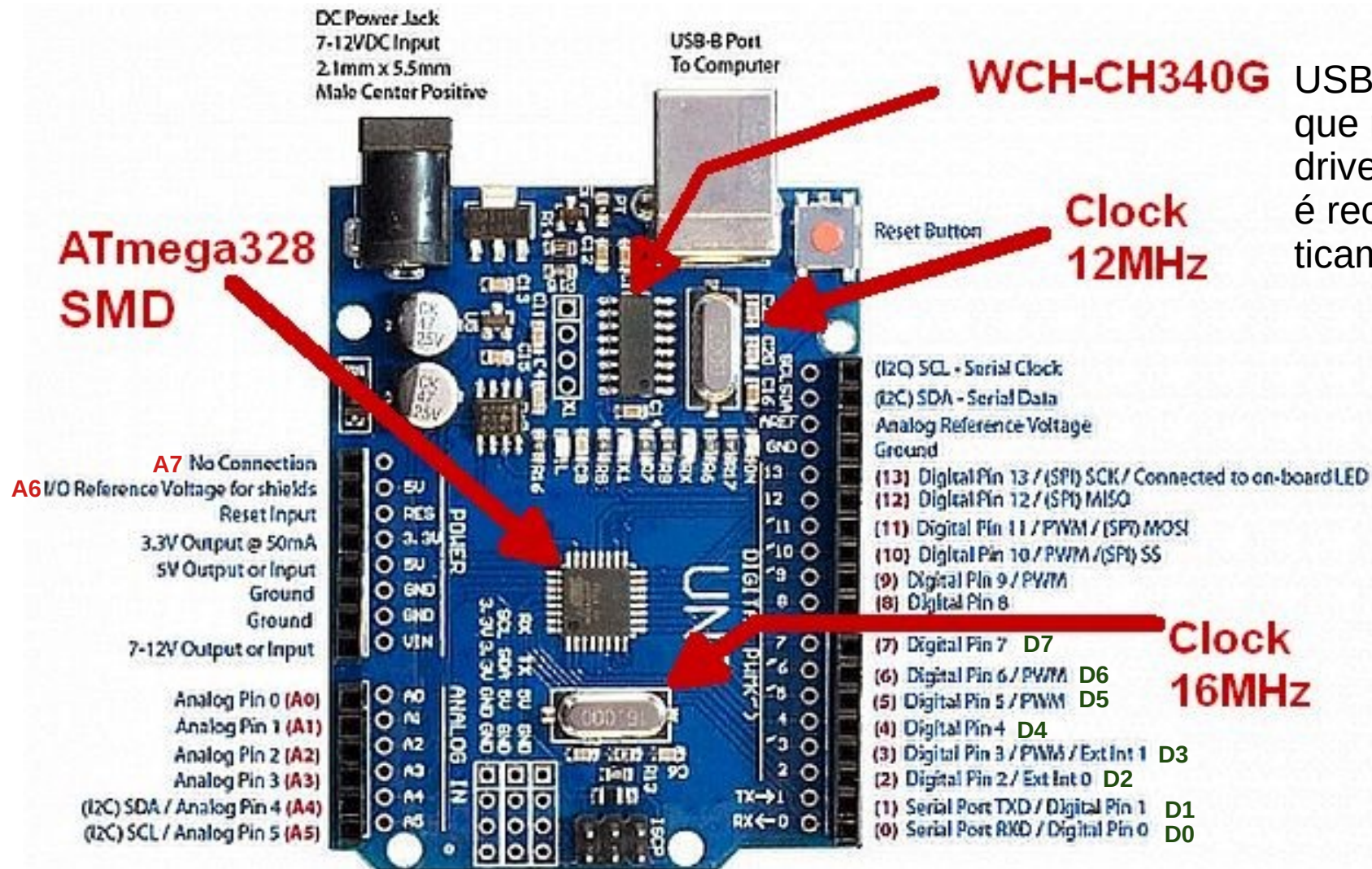
# Gerenciador de dispositivos



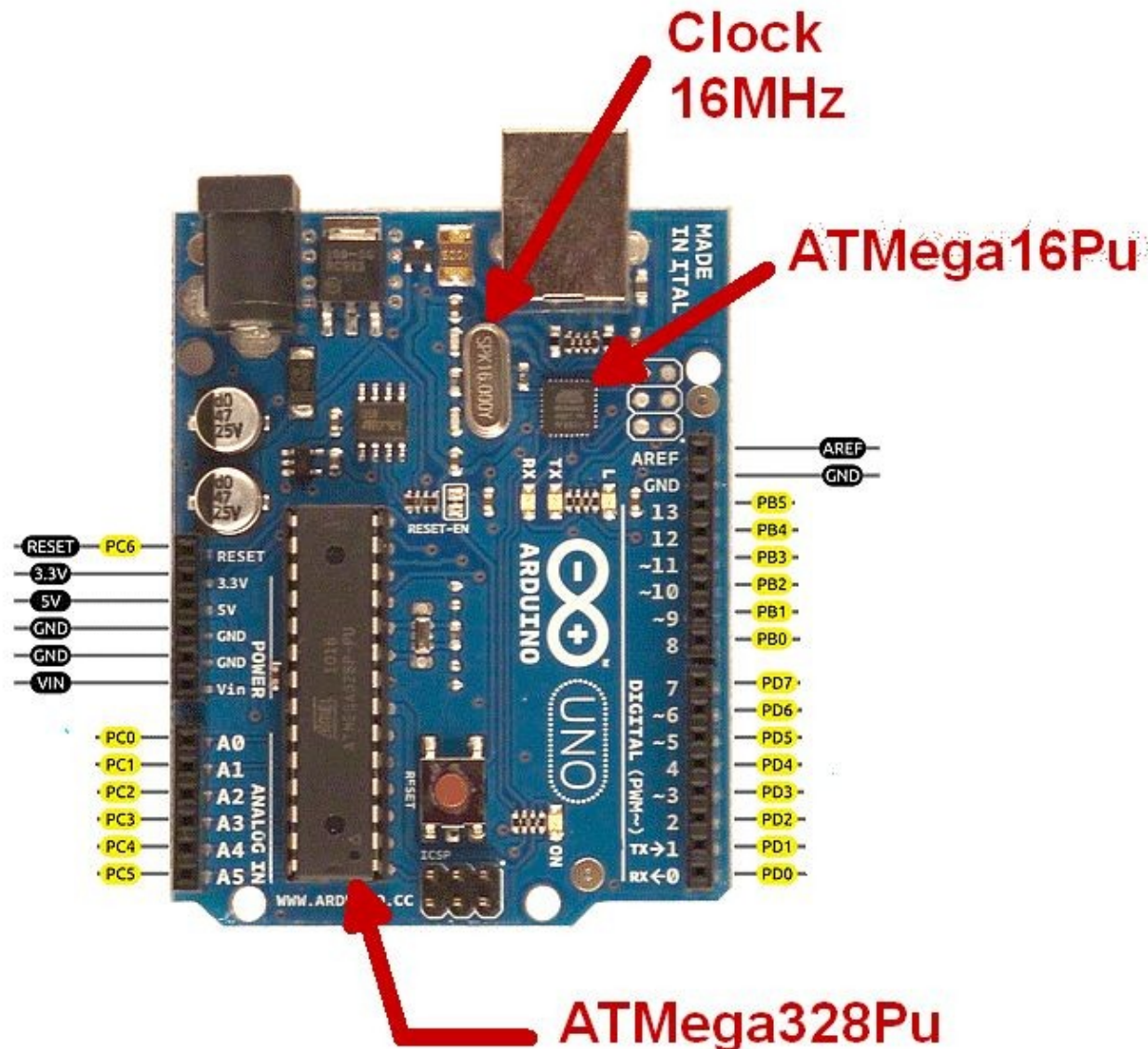
**Gerenciador de dispositivos do Windows mostrando o Arduino legítimo com driver USB PnP reconhecido automaticamente.**



# Placa com chip SMD (chinesa)



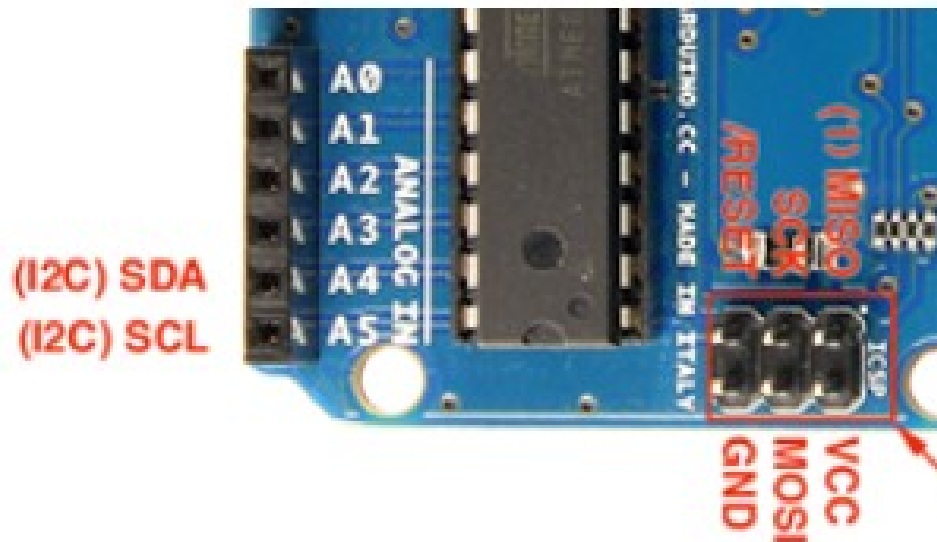
# Placa com chip DIP (italiana)



USB to serial interface,  
automaticamente reco-  
nhecido pelo Windows



# Pinos SDA e SCL (italiana)



Esse esquema se aplica aos Arduinos italianos sem pinos separados SDA e SCL.

# Comparação características



Clock Frequency (MHz)	20 max.
Flash size (KB)	32
SRAM size (Bytes)	2048
EEPROM SIZE (Bytes)	1024
UART	1
SPI	2
I <sup>2</sup> C	1
Timers	1 x 16-bit, 2 x 8-bit
ADC	8 x 10bit (TQFP Package)
GPIO	23 (shared with other peripherals)
Internal ADC reference	Yes

Table 1 – Major features of the ATmega328

Clock frequency (MHz)	48 max.
Flash size (KB)	32
SRAM size (Bytes)	4096
EEPROM SIZE (Bytes)	None.
USART	1
SPI	1
I <sup>2</sup> C	1
Timers	4 x 16-bit
ADC	10 x 12-bit
GPIO	26 (shared with other peripherals)
Internal ADC reference	Yes

Table 2 – Major features of the STM32F030K6T6

# UART e USART



## UART:

Universal Asynchronous Receiver/Transmitter

Circuito simples

Full duplex

Taxa de transmissão variável

## USART:

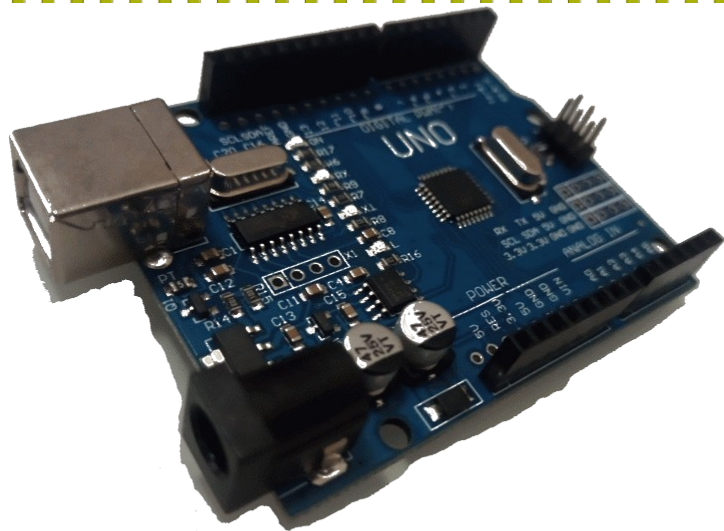
Universal Synchronous/Asynchronous Receiver/Transmitter

Circuito complexo

Half duplex

Taxa de transmissão definida

Mais veloz.



# 003 – Microcontrolador ATMega32 - Arquitetura

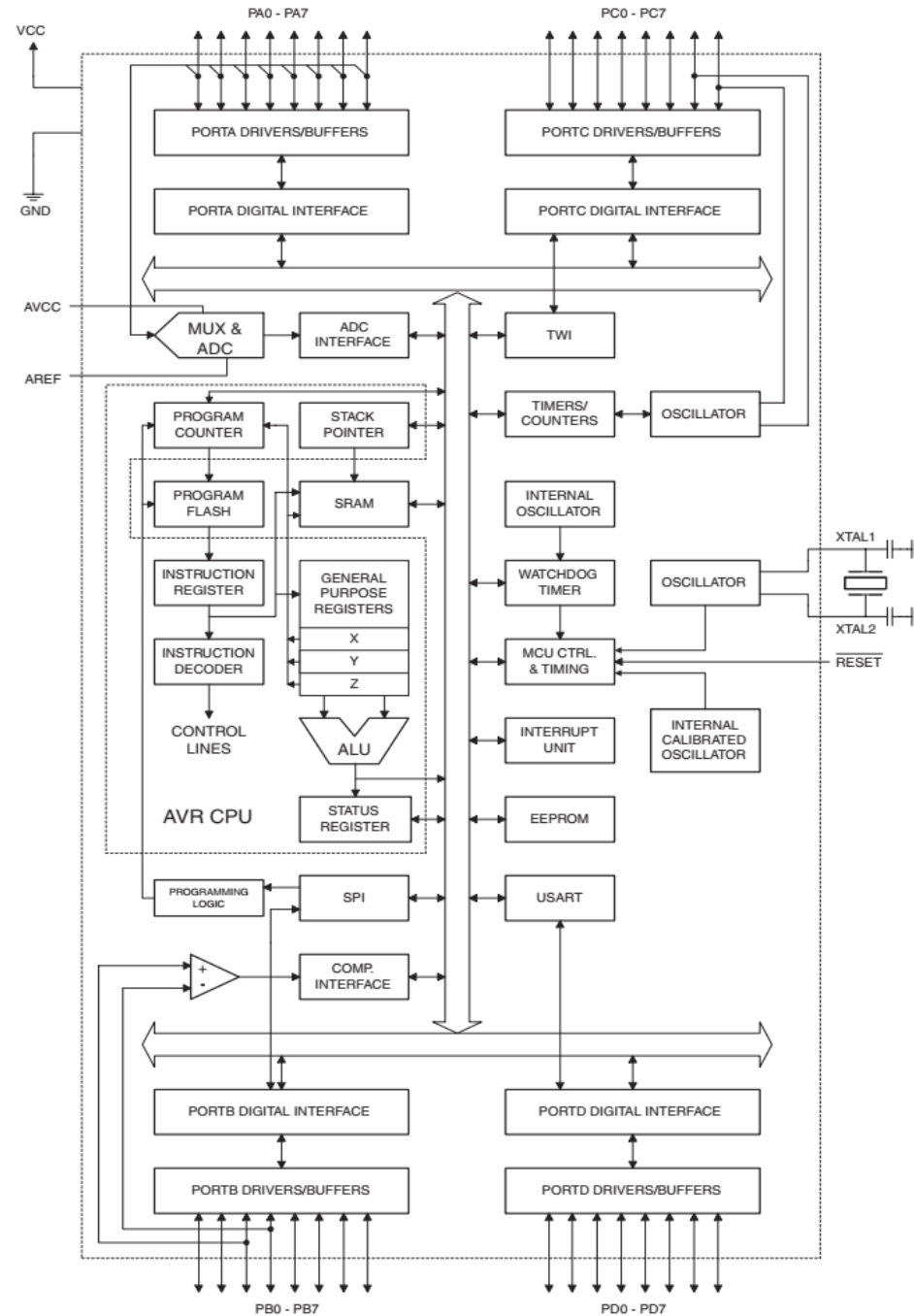
**Um microcontrolador versátil.**

# Características

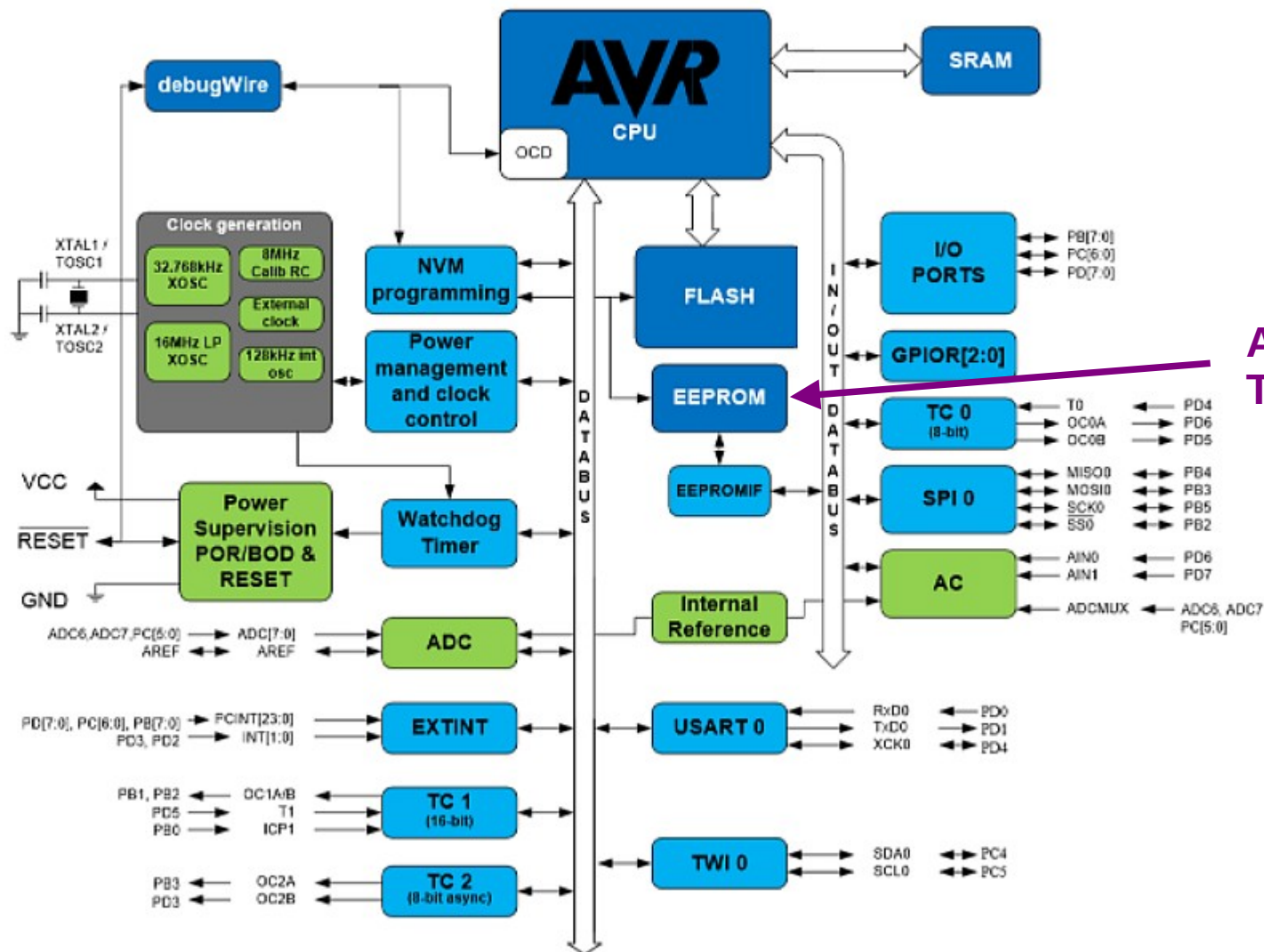


- Processador de 8 bits
- Arquitetura RISC
- SRAM 2Kb (DIP)/ 4Kb (SDM)
- Memória Flash 32Kb
- 32 x 8bits registros de propósito geral
- Arquitetura Harvard (Memórias de dados e de programa separadas)
- Todos os 32 registros conectados à Unidade Lógico-Aritmética
- 6 registros de 8 bits podem ser agrupados em 3 registros de 16 bits (X, Y e Z)
- EEPROM de 1Kb (DIP)/ 0Kb (SDM)
- 28 pinos (DIP)/ 32 pinos (SDM)

# Processador/Periféricos



# Arquitetura - DIP



Ausente no  
Tipo SMD



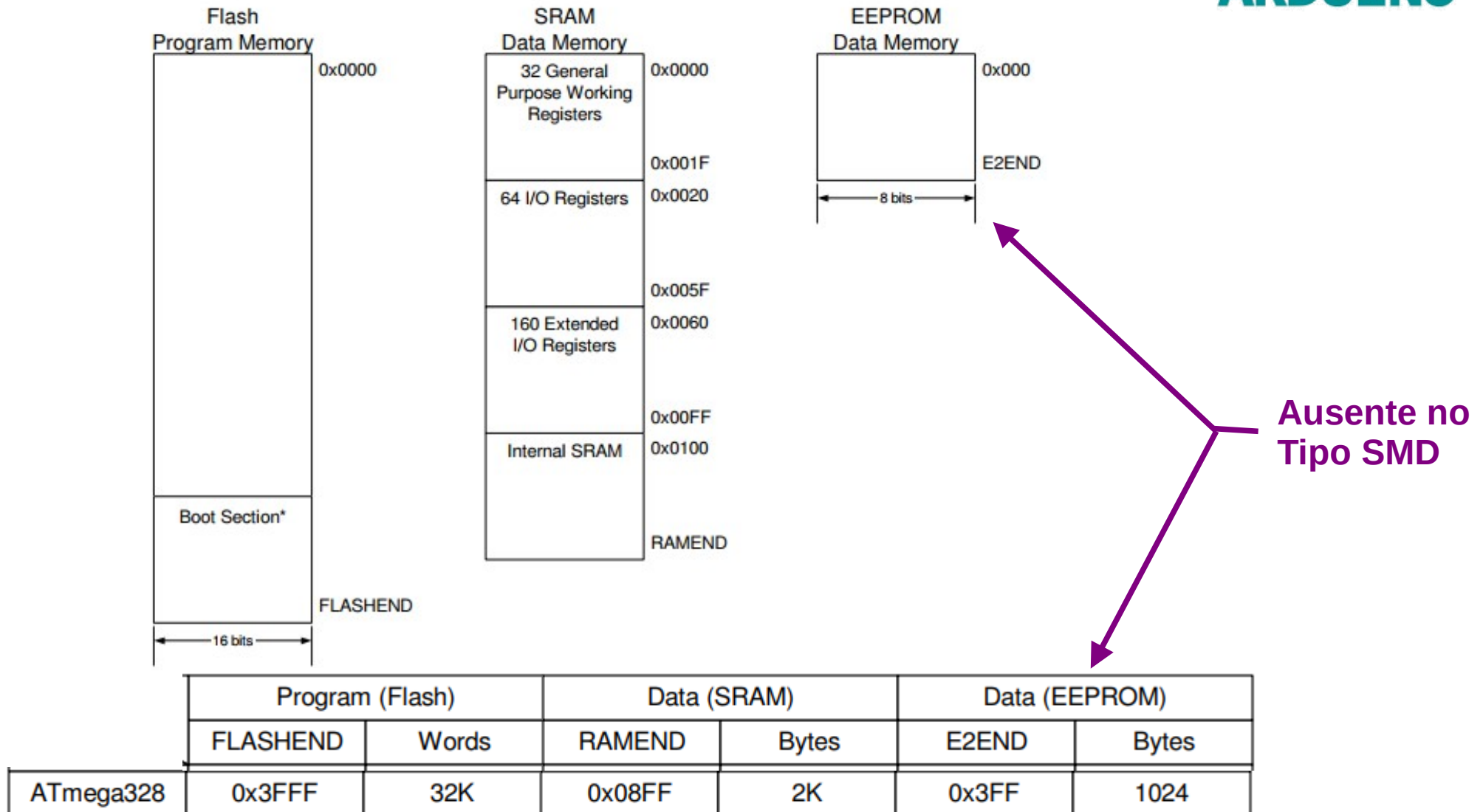
# Registros de propósito geral

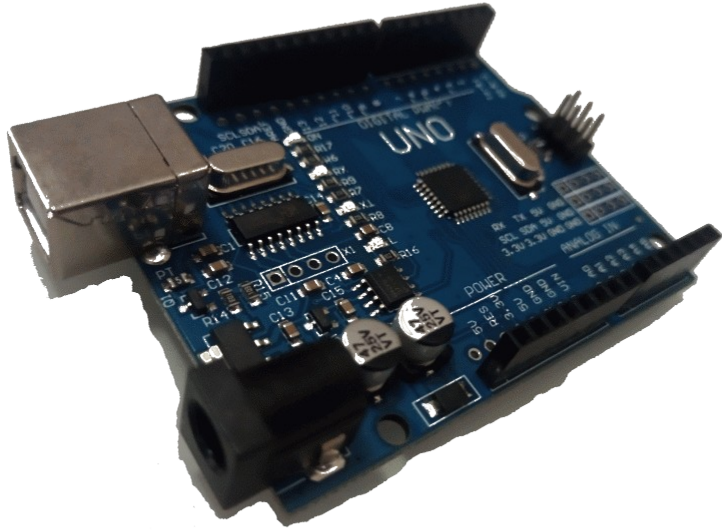


7	0	Addr.	
	R0	\$00	
	R1	\$01	
	R2	\$02	
	...		
	R13	\$0D	
	R14	\$0E	
	R15	\$0F	
	R16	\$10	
	R17	\$11	
	...		
	R26	\$1A	X-register Low Byte
	R27	\$1B	X-register High Byte
	R28	\$1C	Y-register Low Byte
	R29	\$1D	Y-register High Byte
	R30	\$1E	Z-register Low Byte
	R31	\$1F	Z-register High Byte

Agrupamento  
em 16 bits

# Mapa da Memória do ATmega328





# 004 – Preparando para programar o Arduino

Instalando e entendendo a IDE.

# Passos de instalação



Instalar a IDE a partir do site [docs.arduino](https://docs.arduino.cc)

Instalar o driver CH340SER para o modelo SMD

Conectar a placa no USB

Executar a IDE para escolha da placa

ATmega328	0x3FFF	32K	0x08FF	2K	0x3FF	1024
-----------	--------	-----	--------	----	-------	------

# Entrando no site do Arduino



<https://docs.arduino.cc/>

ATmega328	0x3FFF	32K	0x08FF	2K	0x3FF	1024
-----------	--------	-----	--------	----	-------	------

# Instalando o CH340 para SMD - 1



<https://www.blogdarobotica.com/2020/03/21/instalando-driver-serial-para-arduinios-com-chip-ch340/>

- Download Driver CH340 para WINDOWS 10, 8, 7 e abaixo: [link1](#)
- Download Driver CH340 para WINDOWS 11: [link1](#)

Após baixar o arquivo e clicar duas vezes logo se apresentará a tela abaixo:



# Instalando o CH340 para SMD - 2



<https://www.driverscape.com/download/usb-serial-ch340>



Driver Scape

Windows Driver Download Center

[Home](#) » USB-SERIAL CH340

Use the links on this page to download the latest version of USB-SERIAL CH340 drivers. All drivers available for download have been scanned by antivirus program. Please choose the relevant version according to your computer's operating system and click the download button.



## System Information

Your machine is currently running: Windows 10 64bit (Detect)

## USB-SERIAL CH340 Drivers Download

**Driver Version:** 3.3.2011.11

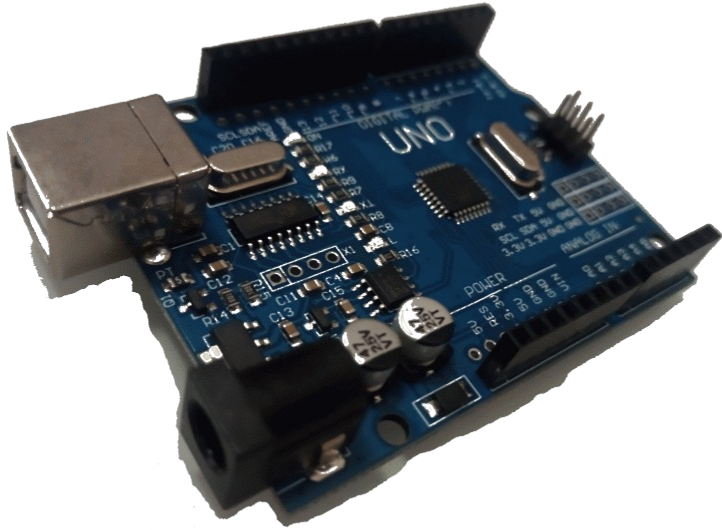
**Release Date:** 2011-11-04

**File Size:** 41.02K

**Supported OS:** Windows 10 64 bit, Windows 8.1 64bit, Windows 7 64bit, Windows Vista 64bit, Windows XP 64bit

Download





# 005 – Projeto Leds 3 portas e 3 Leds

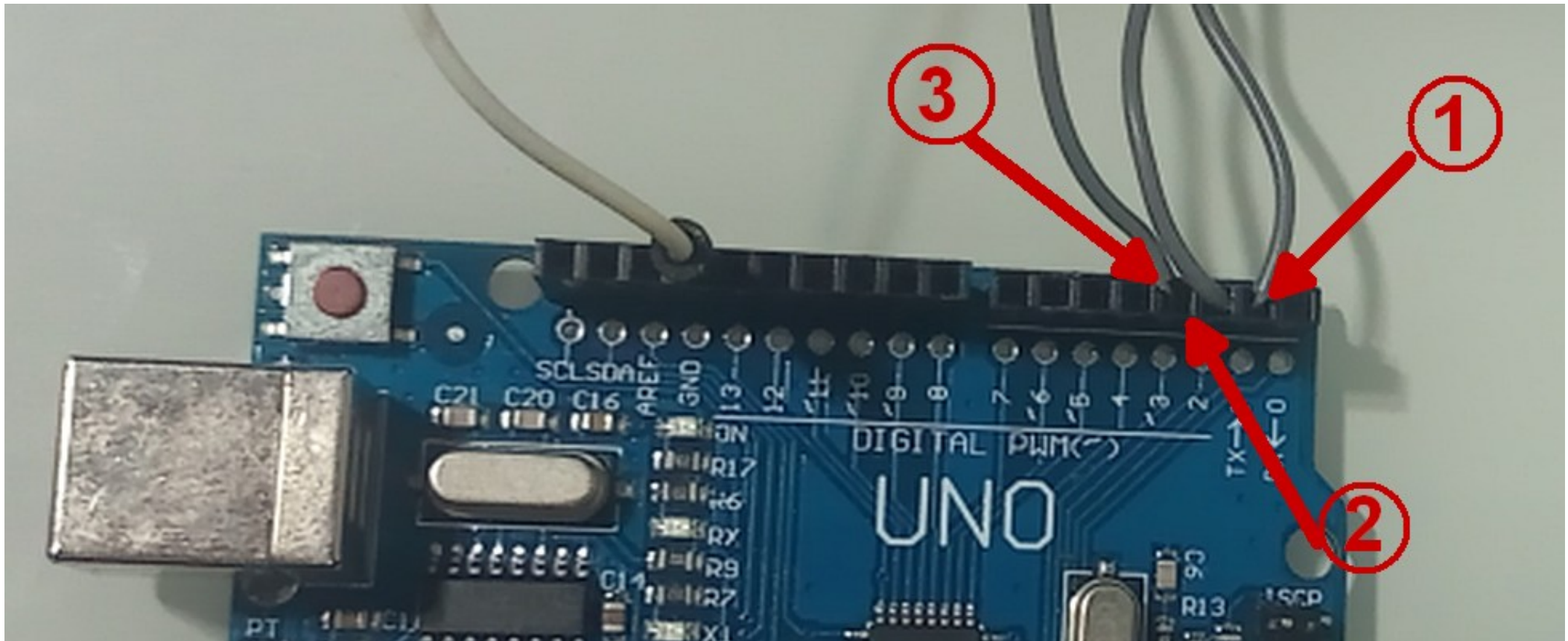
**Enviando sinais para Portas.**

# Operação do Arduino

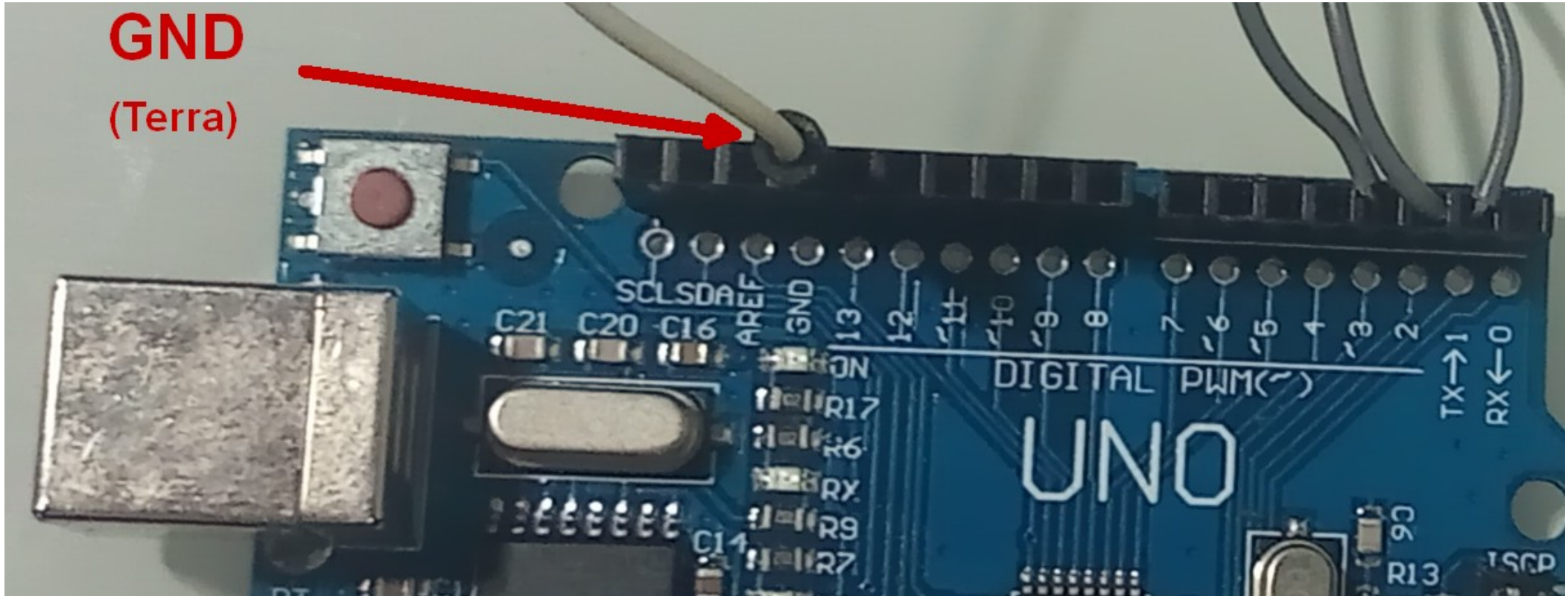


- Vamos operar com as **PORTAS** do Arduino
- De preferência com as **PORTAS** que servem tanto como **ENTRADA** como para **SAÍDA**
- Um microcontrolador gerencia processos chaveando as **PORTAS** com nível baixo (0-zero/desligado) ou alto (1-um/ligado), e desta forma desativando (nível baixo) ou ativando (nível alto) uma **PORTA**
- As operações de ativação e desativação são feitas por software, através da linguagem **Assembler** ou linguagem **C** por meio de programação
- A programação pode ser feita pela **IDE Arduino** ou pelo **ATMEL Studio** (IDE do fabricante, hoje Microchip)

# Portas de Entrada e Saída

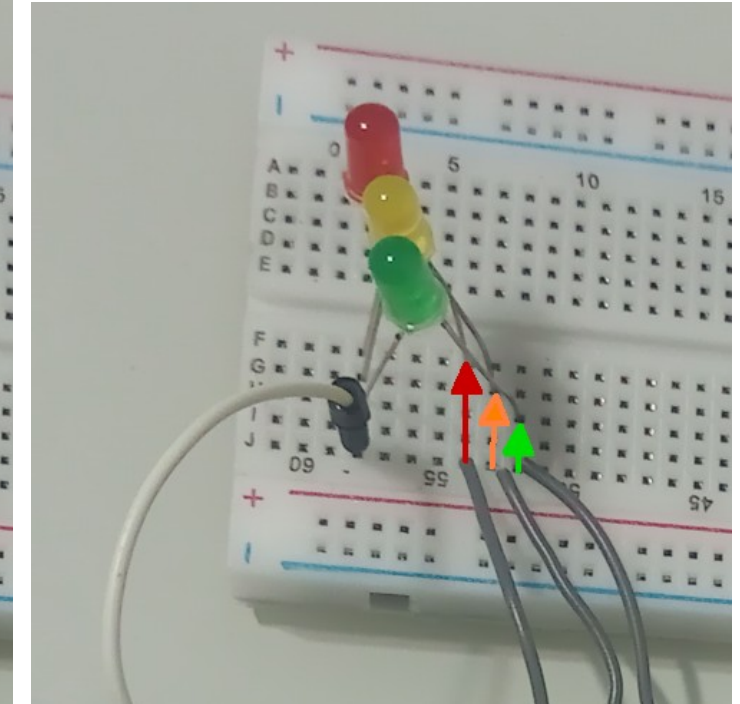
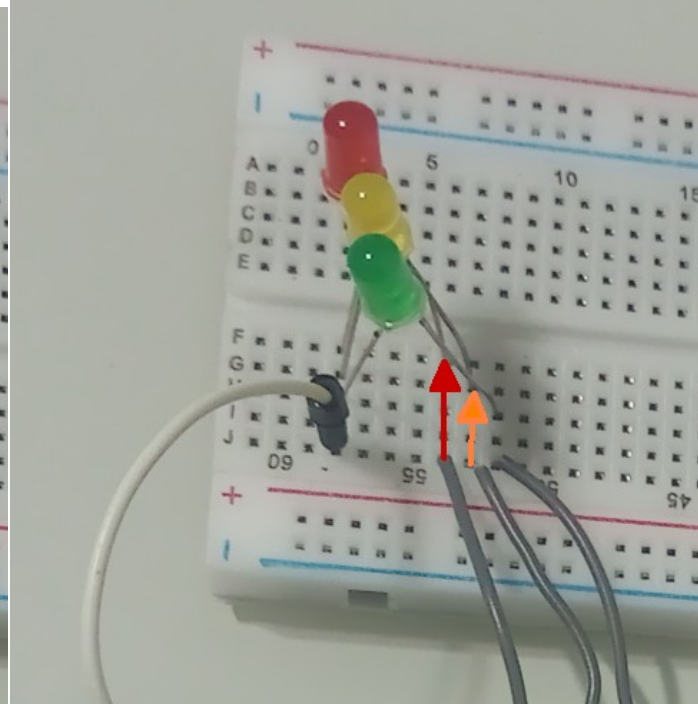
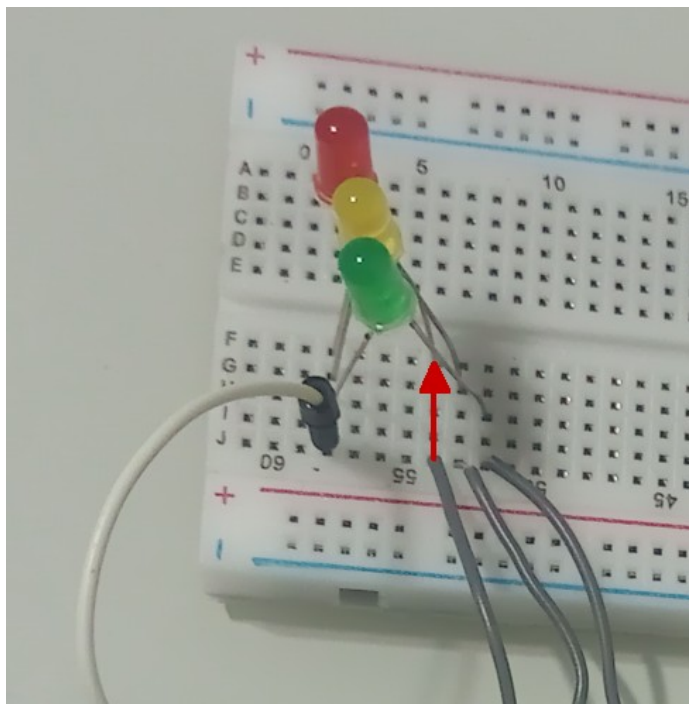


# Ground (GND) – Terra – Potencial Zero

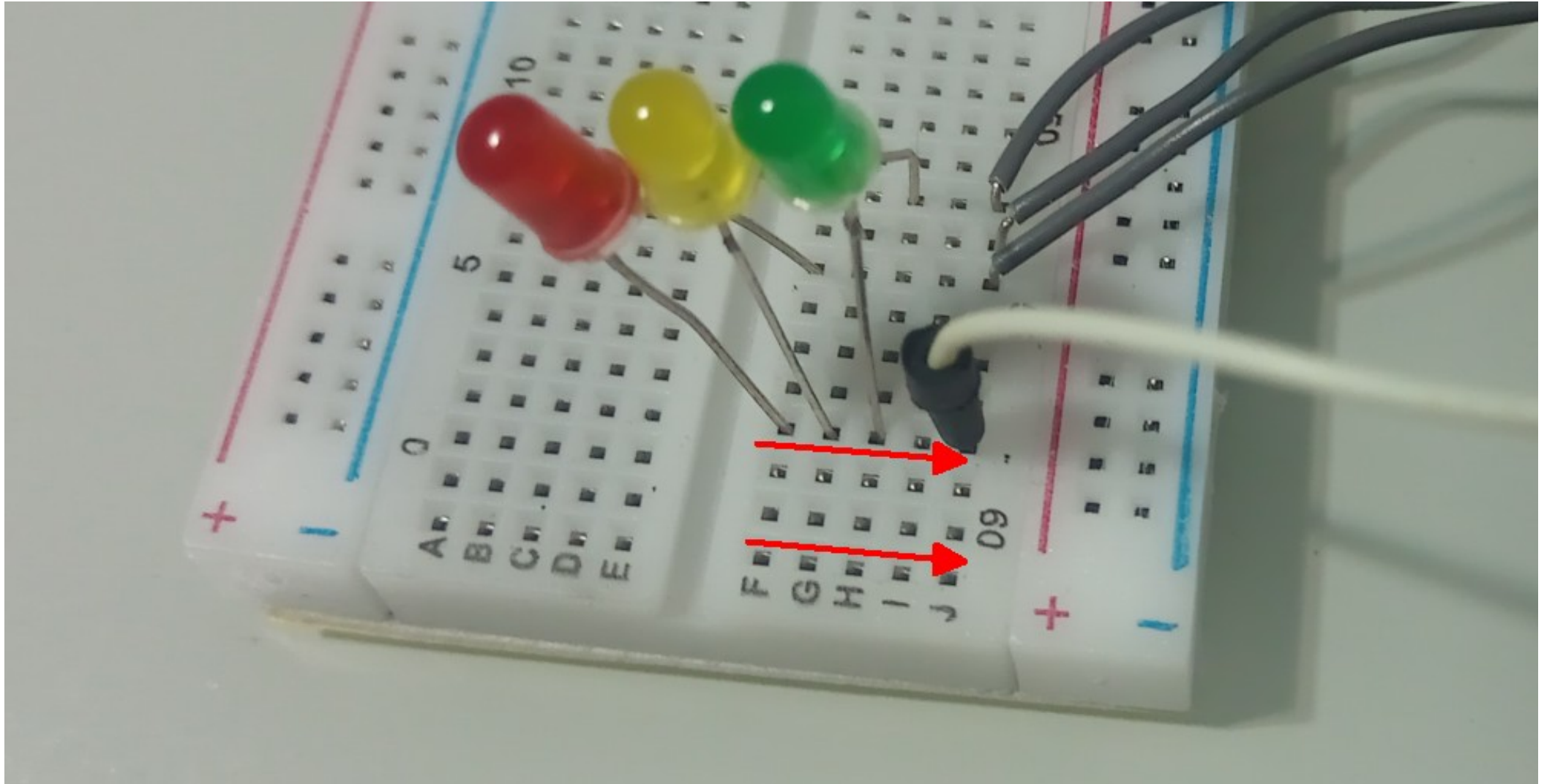




# Conexões aos Leds



# Conexões aos Leds



# Instruções de Portas em C



```
pinMode(1, OUTPUT); // Define porta de SAÍDA
```

```
pinMode(1, INPUT); // Define porta de ENTRADA
```

```
digitalWrite(1, HIGH); // Coloca a porta em estado alto/ativo
```

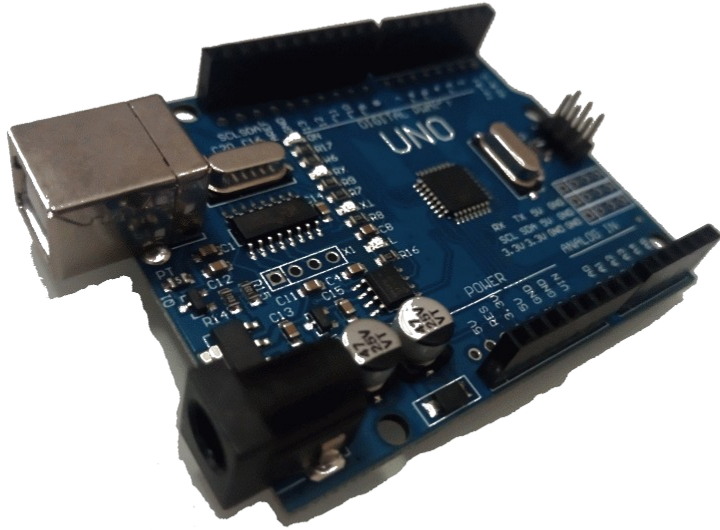
```
digitalWrite(1, LOW); // Coloca a porta em estado baixo/passivo
```

# Pausa de processamento em C



`delay(1000); // Define pausa em milissegundos`



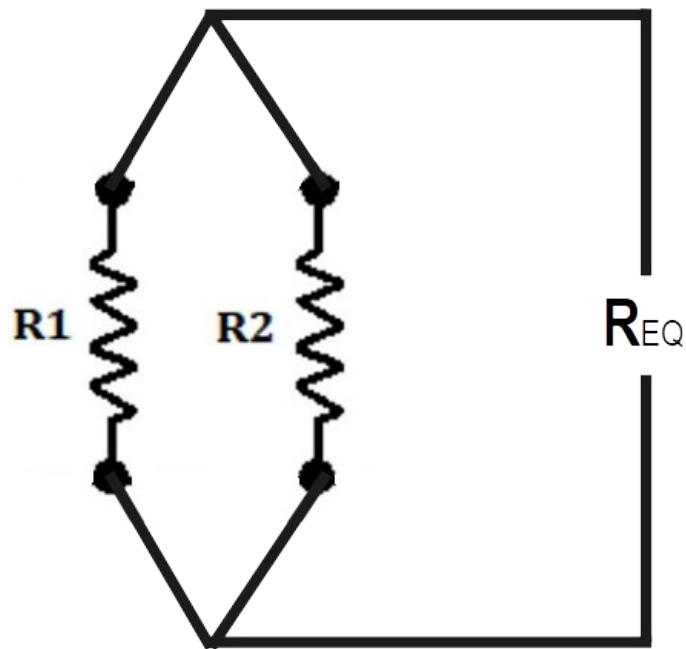


# 006 – Projeto Leds

## Como usar Funções

### Detalhes da linguagem C.

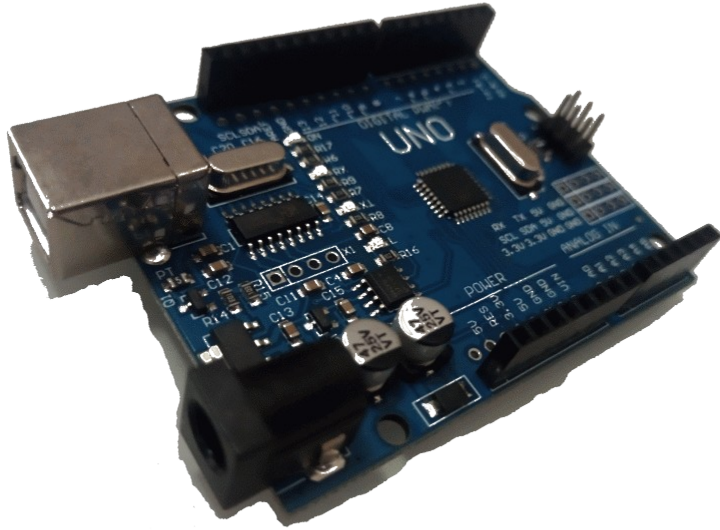
# Resistores em paralelo



$$\frac{1}{R_{EQ}} = \frac{1}{R_1} + \frac{1}{R_2}$$

$$\frac{1}{R_{EQ}} = \frac{1}{r} + \frac{1}{r}$$

$$R_{EQ} = \frac{r}{2}$$

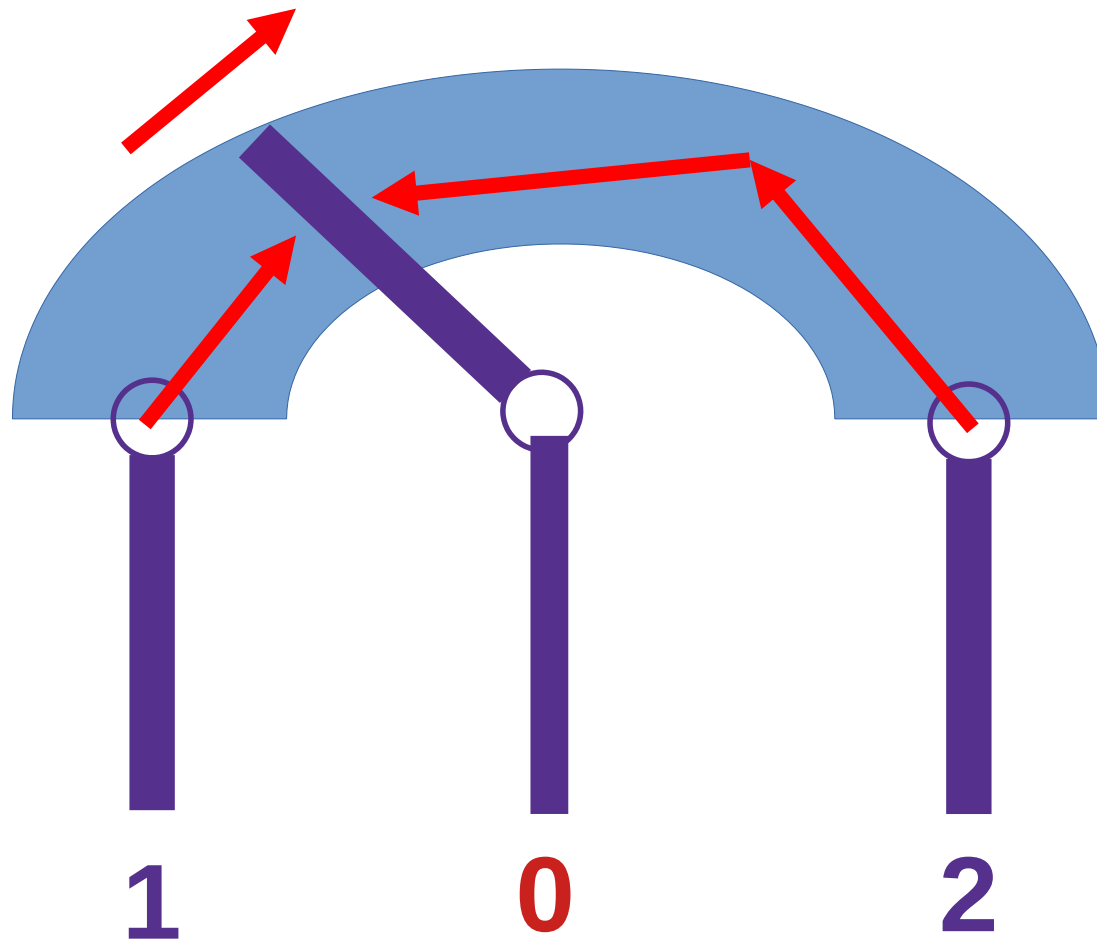


# 007 – Utilizando o potenciômetro no circuito

**Resistência variável.**

# Potenciômetro

## Resistor variável



# Circuito esquemático

