

ELETIVA: ARDUINO UNO - ATMEGA 328

VISÃO GERAL

- IoT - Internet of Things
- Microcontrolador
- Arquitetura 2 - microprocessador 8086 - Todo recurso tínhamos que acoplar através do barramento de dados (controladoras) - circuito grande
- Hoje um processador diminuiu muito
 - Microcontrolador - todas as controladoras estão embarcadas, não precisa acoplar
- No 8086 tinha a vantagem de colocar o programa dentro do chip, na prática tem a memória de programa e dados externo
- Microcontrolador tem memória de dados dentro do chip
- ESP 32, STM 32, ATMEGA (Para construção de um ambiente chamado Arduino)
- Foco em Arduino

- Shields espetados na placa arduino
- Arduíno sozinho não consegue acessar internet → shield que tem cabo ethernet → Espeta uma na outra → Vai permitir que o arduino acesse a internet à cabo de rede
- http ftp, ntp, https...
- SIM 900 - SHIELD - "Celular sem interface gráfica"
- Arduino é software e hardware

QUANTIDADE DE TRABALHOS

- 3 trabalhos
- 2 durante o concentrado - 3 tempo maior

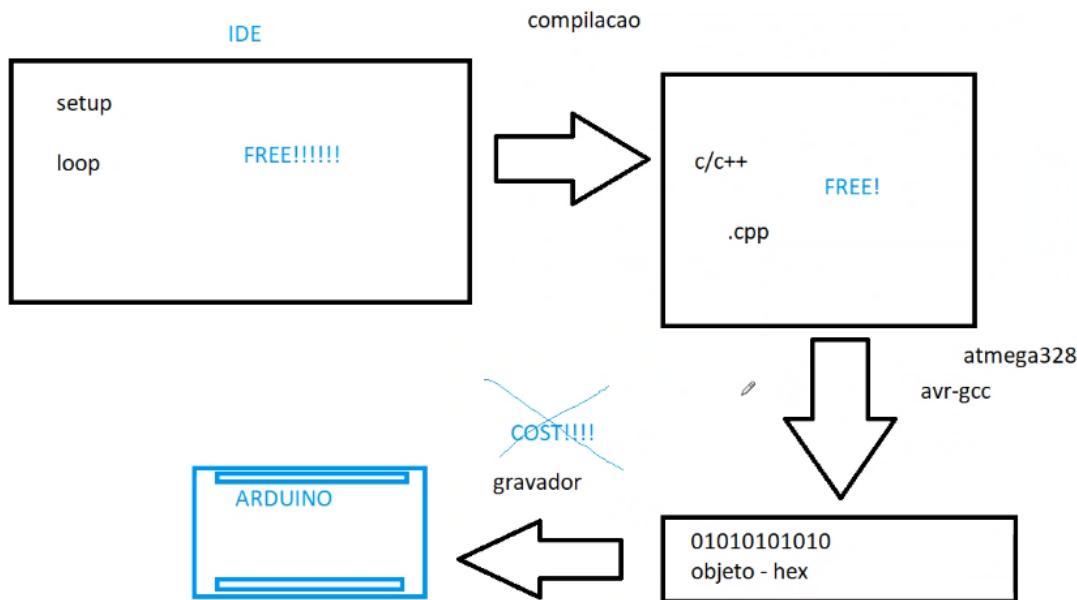
O QUE SERÁ ESTUDADO

- ATMEGA 328
 - Microcontrolador da ATMEL
- Antigamente:
 - Compilador C
 - ISP PROGRAMMER para gravação
 - E vários outros programas tinham que ser instalados separadamente
- Alimentado com 5v
- Clock de 16 mghz
- Compilador AVR GCC
- Engenheiros italianos tiveram a ideia de pegar todos os softwares acima e juntar tudo

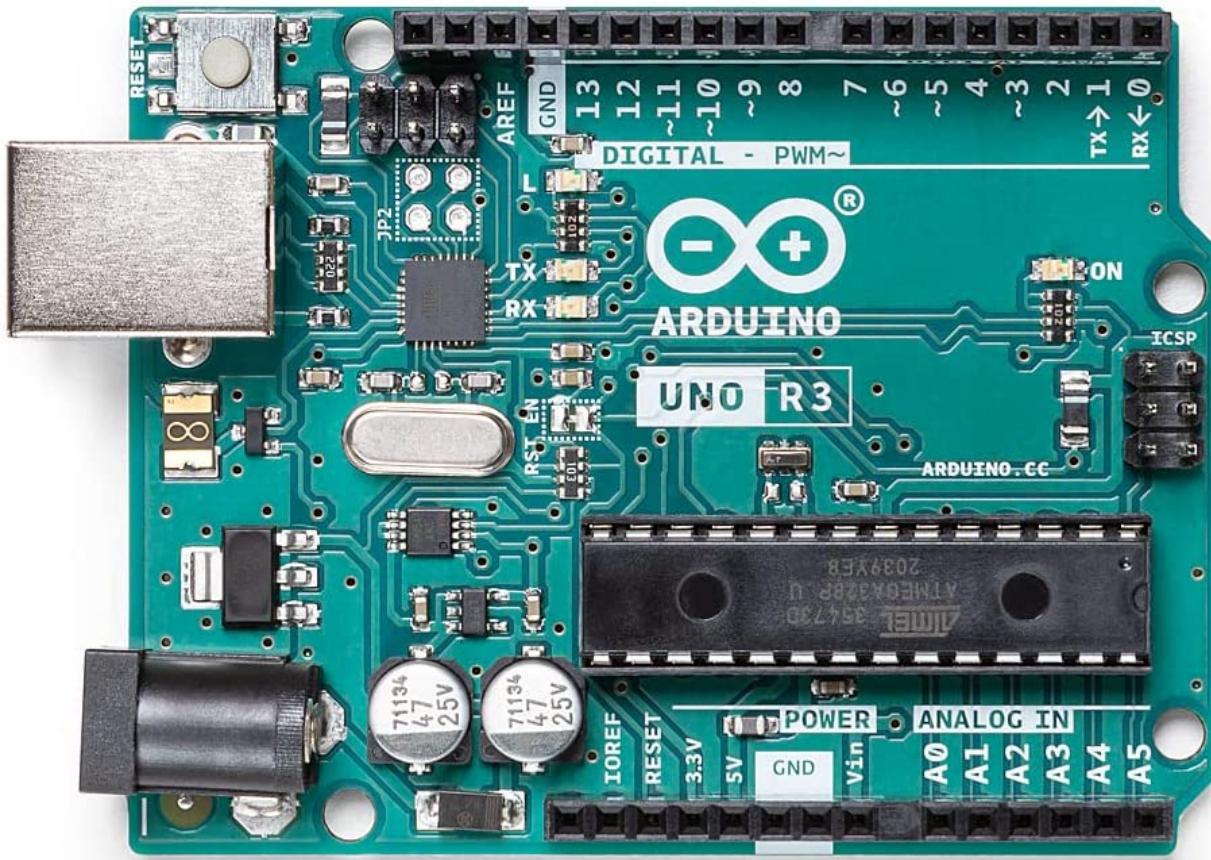
LINGUAGEM

- Criaram a IDE Arduino

- Função setup e loop → C do Arduino
- C do Arduino → C modificado
- Estas funções não existem no C clássico, exclusivas do C do Arduino
- Reservadas no compilador C do Arduino
- C++ Por baixo do pano → C
- Ao compilar, pega o código do arduíno e transforma ele em C/C++ → “Compilação temporária”
- Arquivo .cpp é compilado com o AVR - GCC (Compilador C do fabricante (ATMEGA328))
- Gera um arquivo binário (objeto - hex)
- Este objeto através de um gravador é colocado no arduíno
- Ao compilar o programa, ele gera o executável
- Programa → sketch
- 32k de memória e +2k de memória de dados



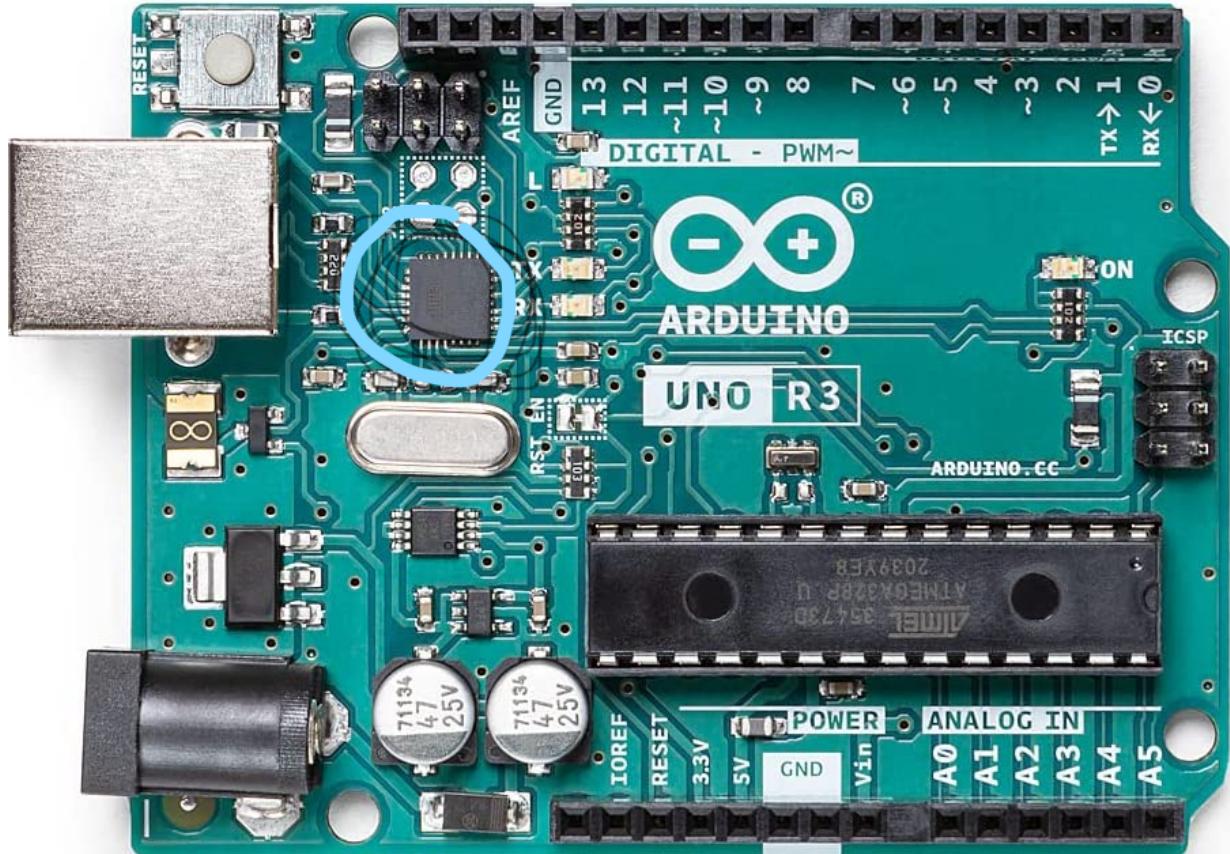
COMPONENTES DO ARDUÍNO UNO

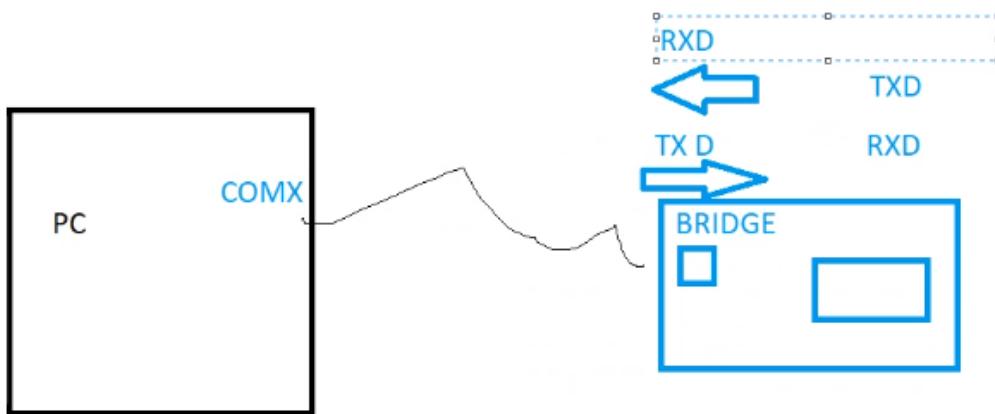


- Arduino é open hardware
- é possível clonar
- ARDUINO UNO R3

- Painel de controle → conecta arduino → Arduino UNO COM6
(CANAL DE COMUNICAÇÃO 6)

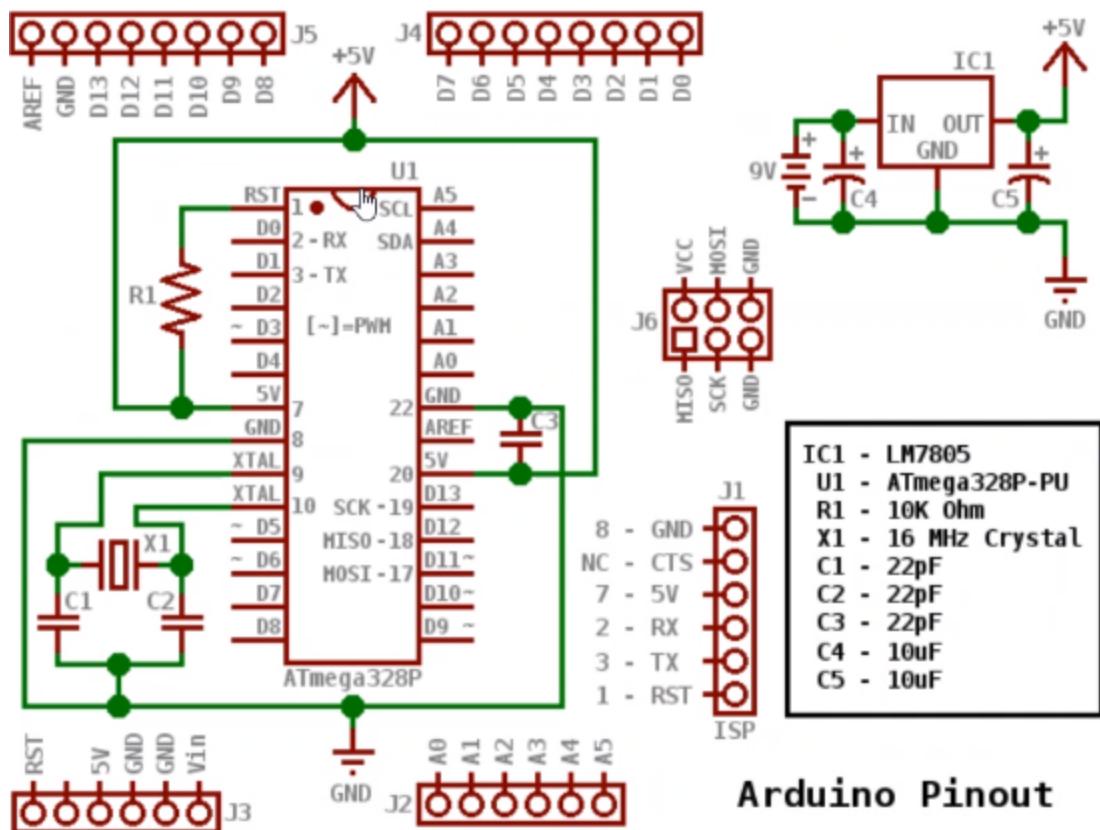
- Arduino uno é para não precisar comprar o gravador:
 - A placa dentro do processador tem um software de fábrica → bootloader → primeiro programa executado quando liga na alimentação ou quando reseta
 - Pula do bootloader para a aplicação
 - bootloader tenta conversar com a IDE do arduino mediante protocolo
 - Se enxargam através de um canal de comunicação (COM)
 - Na IDE escolhe a porta que o ARDUINO está
 - Bridge cria um canal de comunicação entre o PC e o ATMEGA328
 - PC (USB) Vai fazer um SCAN e o SO vai detectar um chip usb e verifica o que ele é
 - Reconhece que é um CHIP de comunicação
 - Cria um canal de comunicação
-
- BRIDGE REALÇADA:





- Transmite dados por essa ponte (Transmitted Data e Received Data)
- Nem sempre a placa arduino vai ter esse chip
- Arduino Leonardo por exemplo não tem bridge separada, ela está dentro do circuito integrado
- Se for rodar em outro arduíno tem que mudar na IDE (ferramentas) avisar o compilador que vc trocou o processador
- Selecionar “Carregar” na IDE para transferir para o arduíno
- avrdude abre o canal de comunicação (COM) e reseta o processador do arduíno através de um circuito eletrônico e executa o bootloader. Bootloader manda via TX um carácter de controle e o avrdude recebe. Ao reconhecer ele manda o bootloader se preparar para gravar dentro de uma memória instruções.

- X instrução em X endereço, Y instrução em Y endereço...
- Programa então é gravado e depois o bootloader da um salto para o seu programa.
- Depois que reseta, bootloader pula para a aplicação.
- Serial é mais lenta

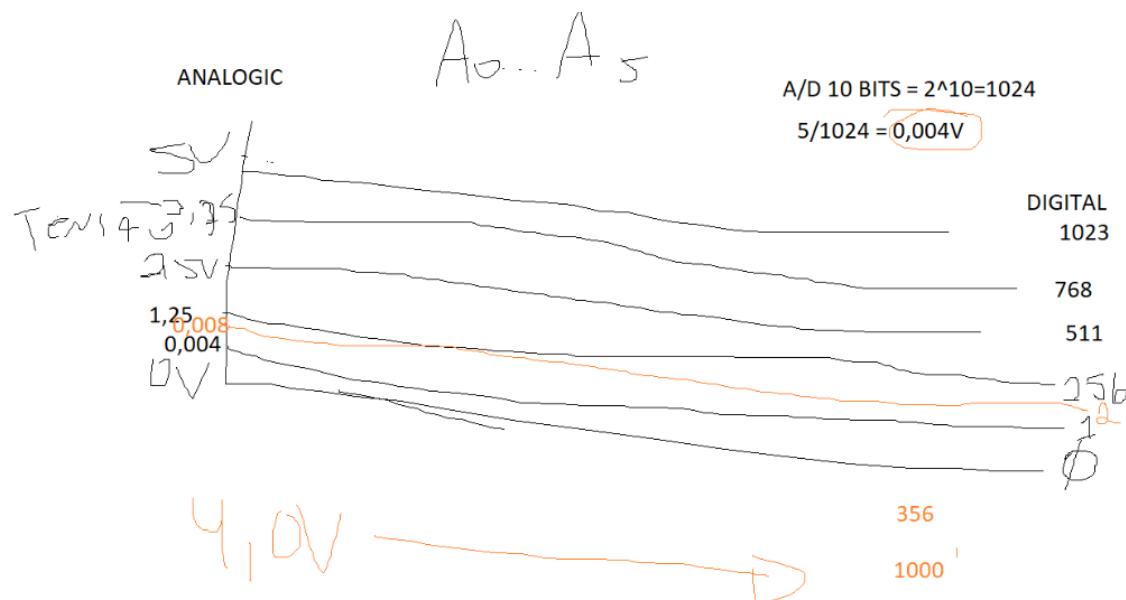


ATMEGA 328 → PINOS 2 E 3 SAO 0 TX E 0 RX

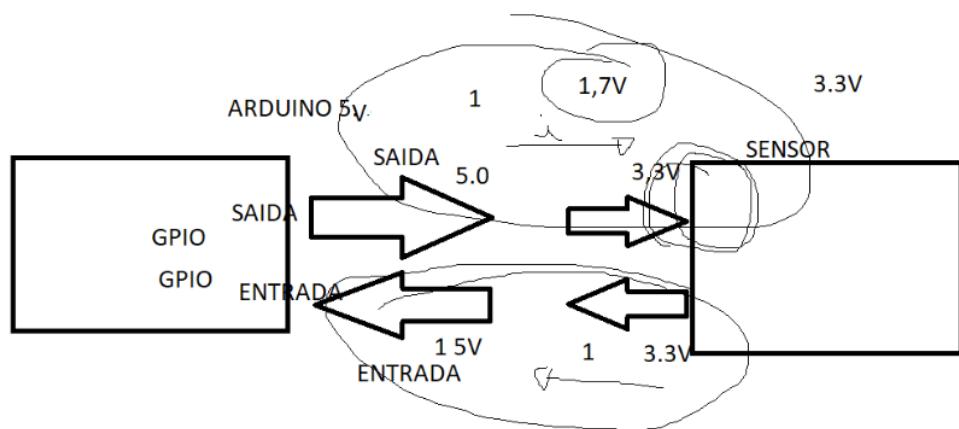
- Se apertar o reset sem querer, não vai perder a aplicação pois ela fica na memória flash (40 anos)
- IDE → FERRAMENTAS → GRAVAR BOOTLOADER

- Toda aplicação que use funções de tempo (pulso, delay)
 - Será baseado em 16mHz
 - 16 milhoes de instruções de assembly em 1 segundo
- é possível alimentar o arduino (processador) pela própria usb
 - Ou fonte de alimentação 9v
 - 9v queima o arduíno pq ele opera a 5v
 - 9v chegam num circuito integrado e baixa para 5v
 - ai vai para o atmega 328
- 0..13 são pinos digitais
 - só consegue fazer a leitura de 2 estados - 0 ou 5 → caso entrada
- a0 .. a5 são digitais/ou analogicos
 - consegue ler qualquer sinal de amplitude de 0 até 5
- por exemplo, o arduíno interage melhor com o mundo real, por meio de sensores ele consegue medir a temperatura durante o dia, menos frio mais frio, menos quente, mais quente, não fica limitado ao true e false
 - 0, 1.25, 2.5
 - converte analógico para digital
 - 0 até 1023
 - A/D 10 BITS = $2^{10} = 1024$
 - $5/1024 = 0,004V$
 - Consegue dividir a tensão em 1024 pedaços de 0,004v
 - 4v = 1000 digital

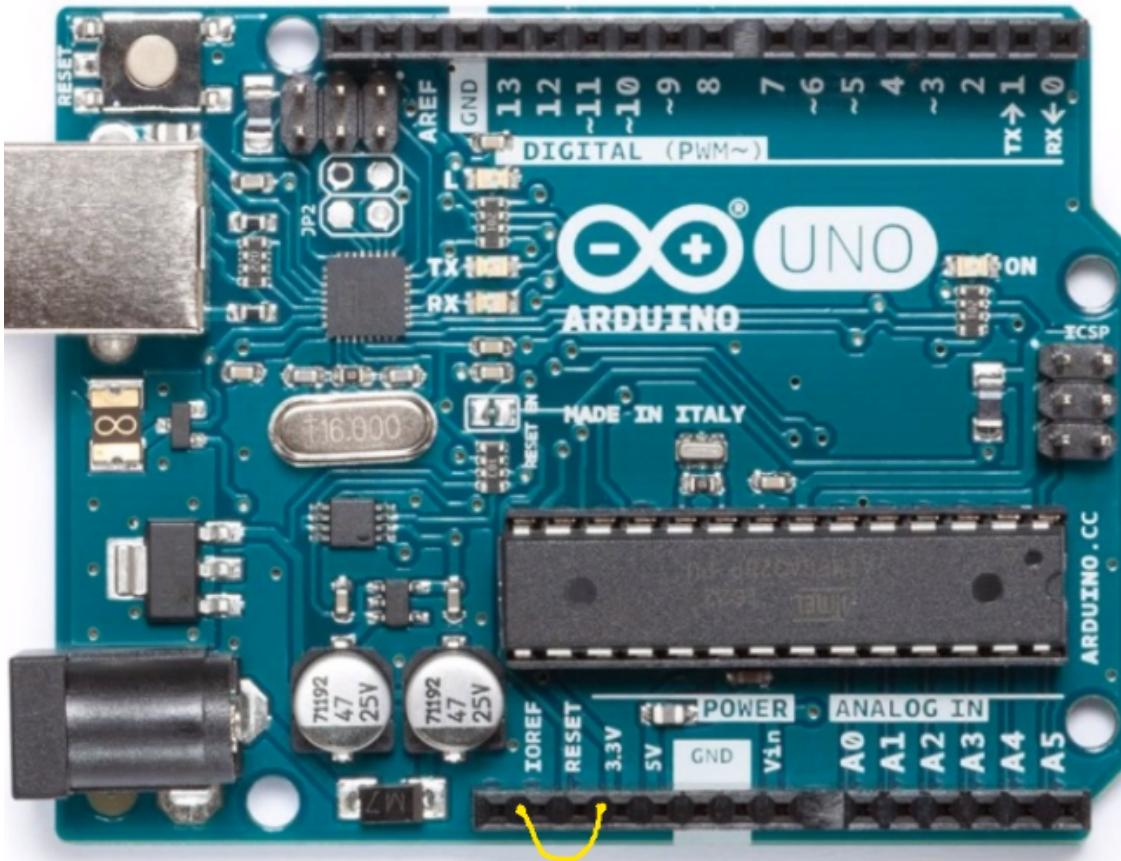
- 356 digital = 1,424V



exemplo: sensor tem 3.3v e arduino tem 5v



- circuito do arduino aguenta 3.3v
- mas o sensor não aguenta 5v que o arduino enviaria
 - ioref: pega um fio e liga no 3.3v



- Sendo assim, todas as IOs vão trabalhar com 0 - 3.3v

CÓDIGO

- SETUP é a primeira função executada. Inicializa o hardware.
- LOOP: Segunda função executada continuamente
- Fabricante lança novo processador → Tem o compilador do processador com C++ → diz que pode usar arduino para

programá-lo

- importar link de arquivo .json do github para a ide