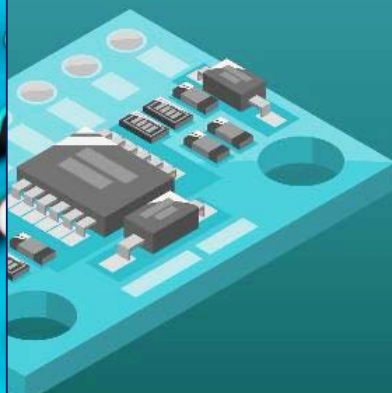


```
int x;  
float y;  
void setup(){
```

```
}  
void loop(){  
  if (5 > 2)  
    x = 5;  
  else  
    x = 2;  
}
```



ARDUINO WEEK



SEX 31/03 às 09h
Arduino, ser ou não ser?
Eis a questão!
Jorge Sebastião Canova

ROBÓTICA
Paula Souza

Cetec
Unidade do Ensino
Médio e Técnico

CPS
Centro
Paula Souza


**GOVERNO DO ESTADO
DE SÃO PAULO**

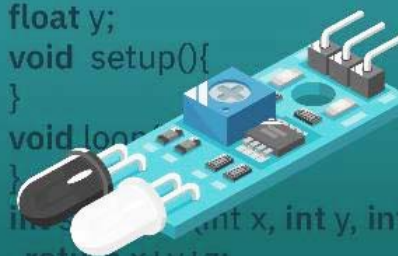
**Cetec
Capacitações**
Conta comercial
do WhatsApp



Escaneie esse
código para
iniciar uma
conversa



```
int x;  
float y;  
void setup(){  
}  
void loop(){  
  int x, int y, int z){  
    return x+y+z;  
  }  
}
```





Jorge Sebastião Canova (Apresentação)

• Professor

- Etec Profa. Anna de Oliveira Ferraz – Araraquara/SP (Cursos de técnico Mecatrônica)

• Formação

- Graduação/Mestrado em Eng. Elétrica – EESC/USP
- Especialização/Licenciatura – Fatec/CPS

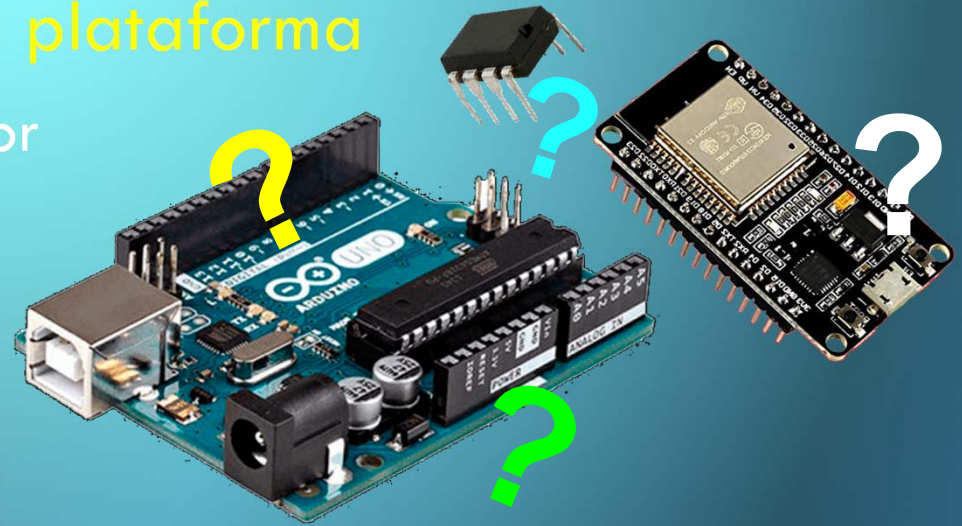
• Trajetória

- Professor/coordenador no ensino superior (2002 a 2021 – Professor ensino técnico (2008...)
- Projetos microcontrolados (2011...)

Conteúdo (Proposta)

- A partir de um estudo de caso, apresentar soluções “diferentes” quanto a plataforma

- CI de microcontrolador
- Arduino
- ESP32



- Considerações sobre as soluções apresentadas



Breve histórico...

- 1835 - Telegrafia por meio de fios - Samuel Morse
- 1854 - Álgebra de Boole para um sistema binário (0 ou 1) - George Boole
- 1876 - Surge o Telefone - Graham Bell
- 1888 - Ondas Eletromagnéticas – Hertz
- 1906 - Surge a Válvula Triodo - Lee de Forest
- 1922 - Transmissão de Rádio no Brasil
- 1926 - Surge a Televisão



Breve histórico...

- 1938 - Aplicou a Álgebra de Boole em Telefonia - Claude E. Shannon
- 1946 - Primeiro Computador Eletrônico - Eniac
- 1947 - Surge o Transistor - Bell Labs
- 1957 - Primeiro Satélite – Sputnik
- 1958 - Surge o Laser
- 1960 - Homem no Espaço - Yuri Gagarin
- 1966 - Surge a Fibra Óptica



Breve histórico...

- 1969 - Homem na Lua - Apollo XI
- 1971 - Primeiro microprocessador – Intel 4004
- 1972 - Televisão em cores no Brasil
- 1974 - Primeiro Microcomputador – Altair
- 1977 - Primeiro Microcontrolador – Intel 8048
- 1985 – Microcontrolador PIC – Arizona Microchip Technology
- 1989 - Telefonia Celular no Brasil



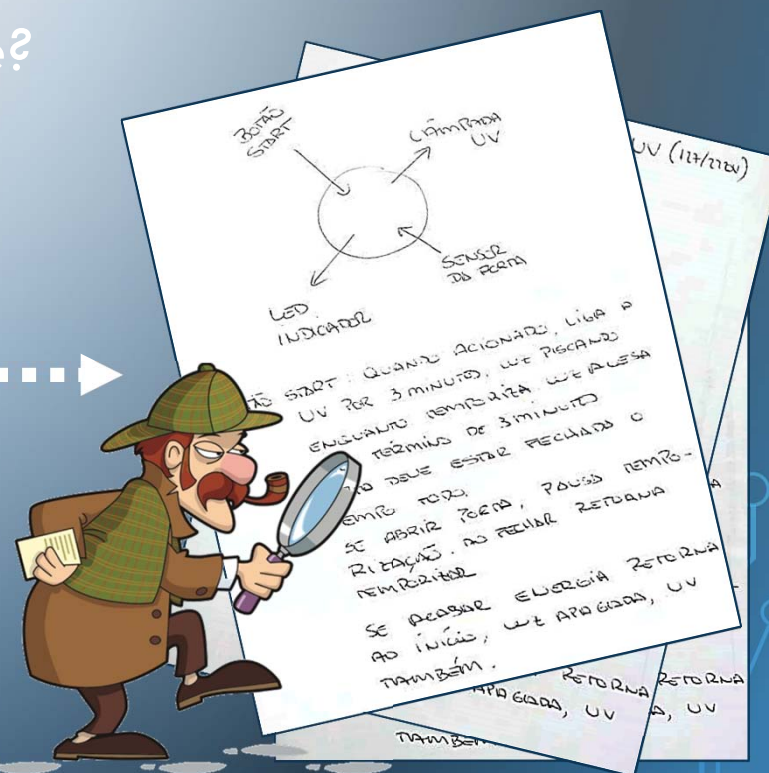
Breve histórico...

- 1993 - Explosão da Internet
- 2005 - Criado o Arduino na Itália
- 2014 – Começa a popularização do ESP8266 (NodeMCU) e ESP32 – Expressif Systems
- Chega de história!!!!

Estudo de caso (cura de material com luz UV)

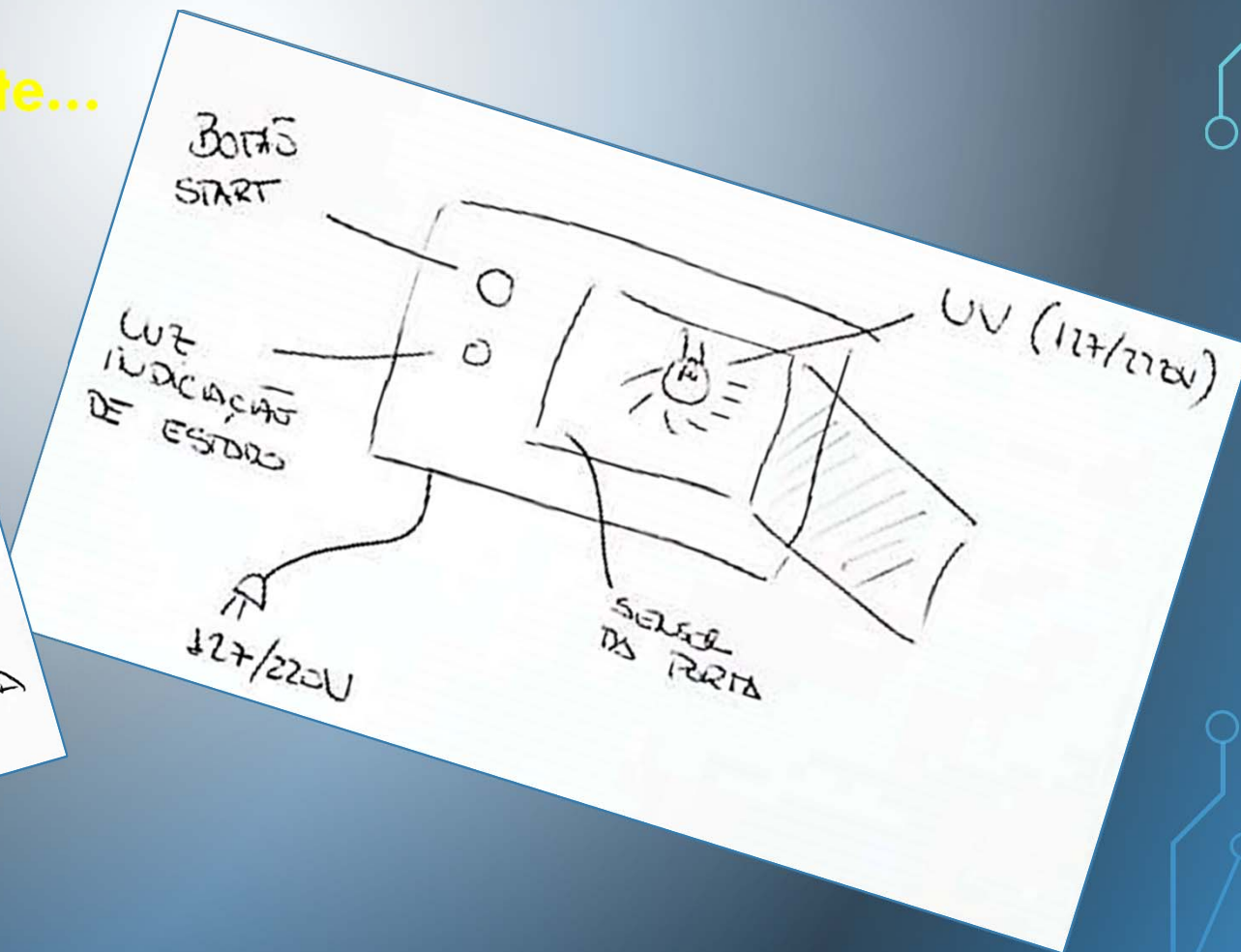
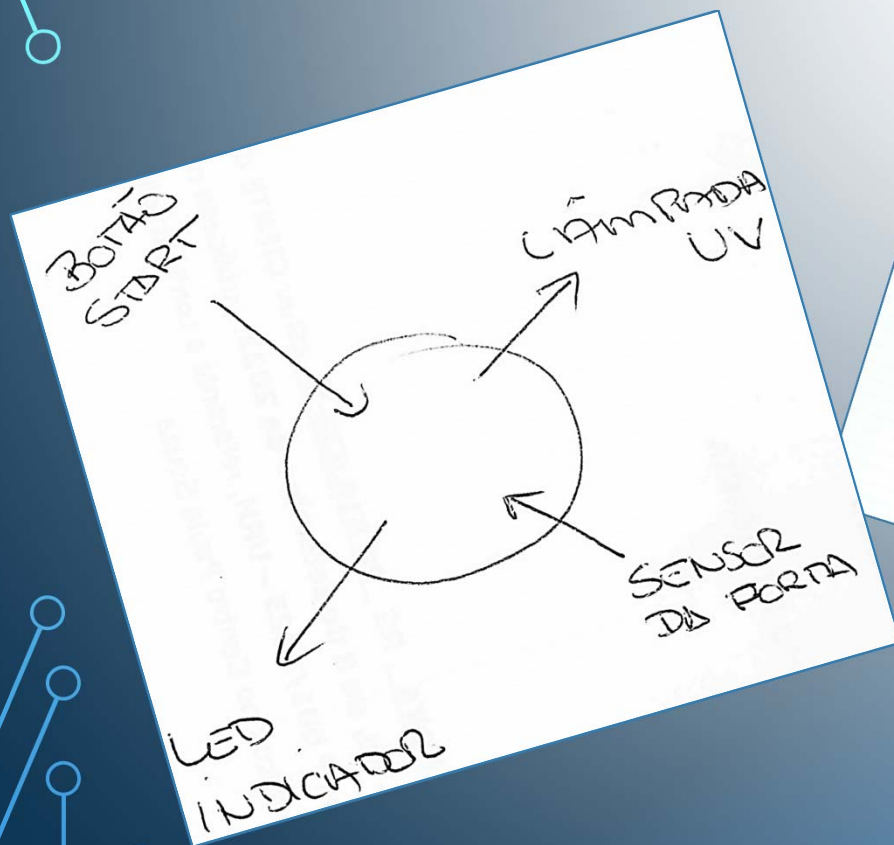
• Qual é o problema? Na conversa com o cliente...

- É necessário conhecer/identificar o problema!
- Quais são as reais necessidades do cliente?
- Faz parte de um projeto maior?
- Quais são os requisitos mínimos?
- Fazer/obter esboços!>
- Quanto mais informação, melhor!!!



Estudo de caso (cura de material com luz UV)

- Os esboços do cliente...



Estudo de caso (cura de material com luz UV)

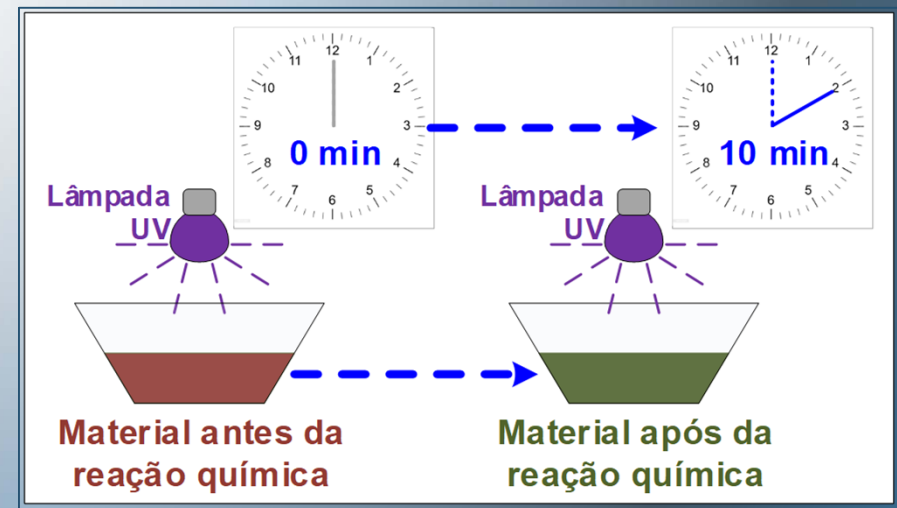
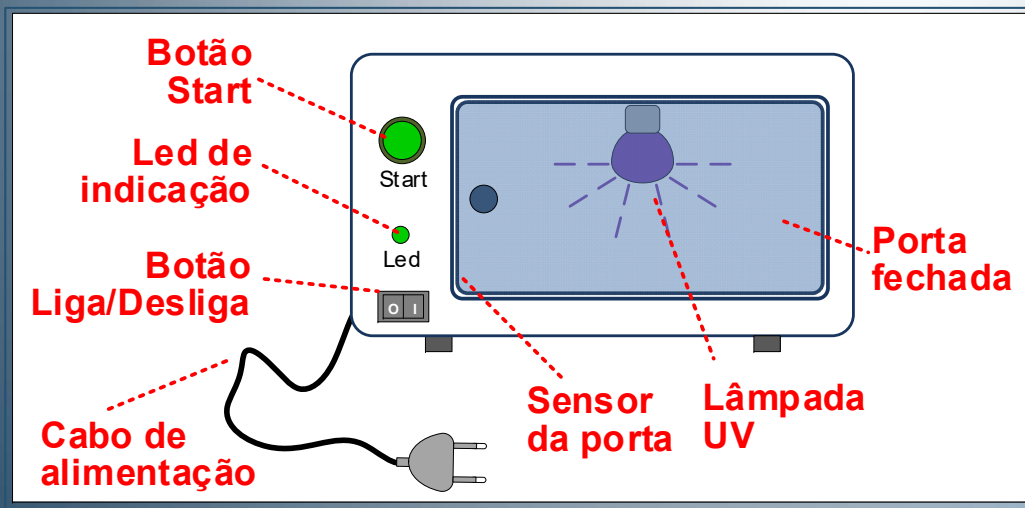
• Nas palavras do cliente...

BOTÃO START : QUANDO ACIONADO, LIGA A UV POR 3 MINUTOS. LIT PISCANDO ENQUANTO TEMPORIZA. LIT ALESA APÓS TÉRMINO DE 3 MINUTOS. PORTO DEVE ESTAR FECHADO O TEMPO TODO. SE ABRIR PORTA, PAUSA TEMPORIZAÇÃO. AO FECHAR RETORNA TEMPORIZAR. SE PERDAR ENERGIA RETORNA AO INÍCIO, LIT APAGADA, UV TAMBÉM.

**Avisou depois
via Zap...
Tempo: 10 min**

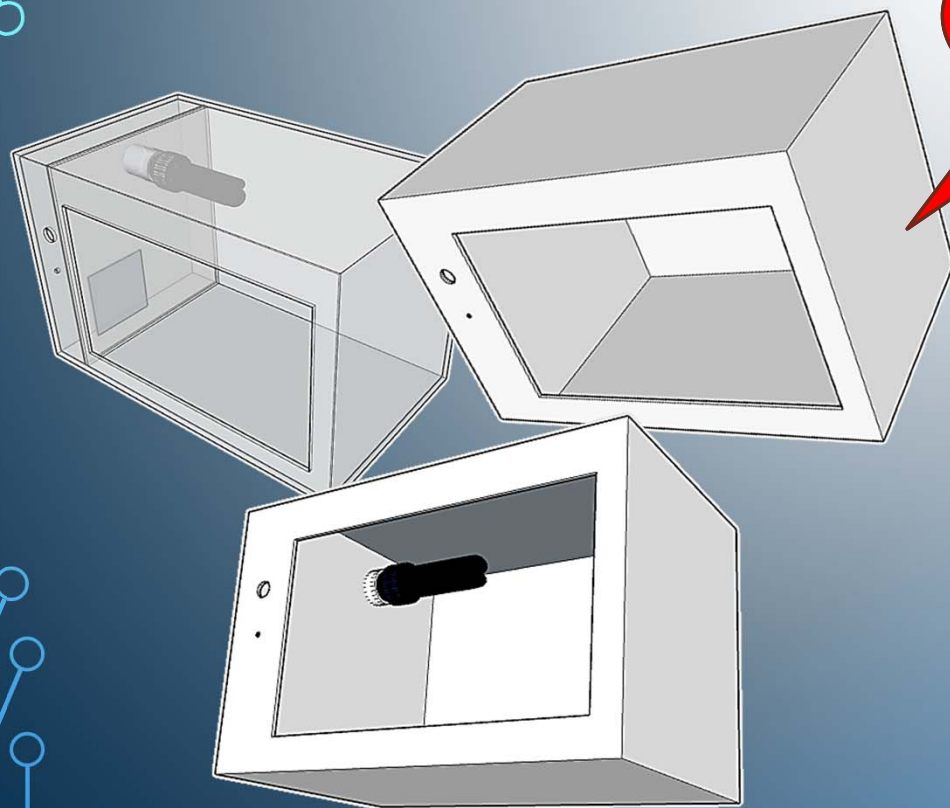
Estudo de caso (cura de material com luz UV)

Filtrando os esboços...

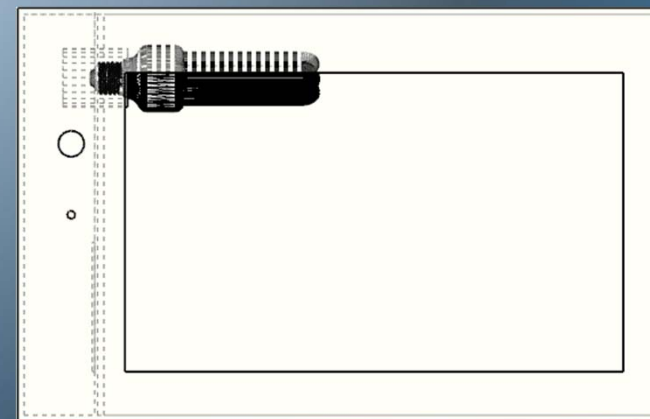
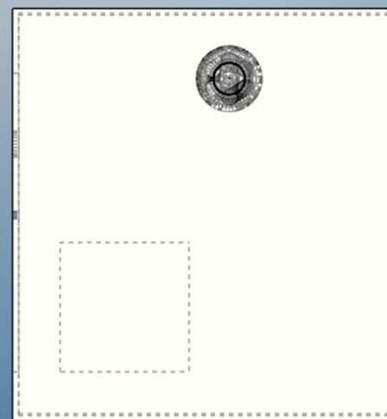
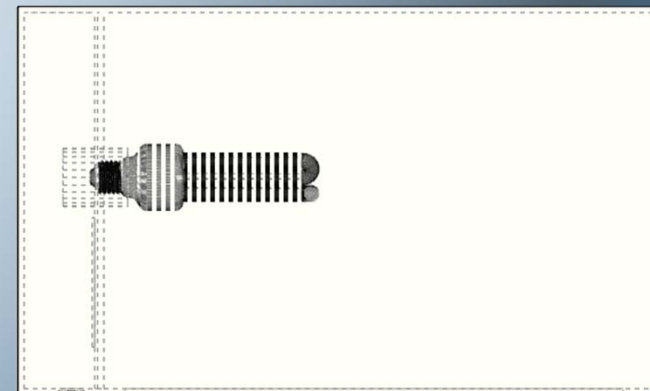


Estudo de caso (*cura de material com luz UV*)

- Filtrando os esboços...



Um TCC talvez?



• Estudo de caso (*cura de material com luz UV*)

• Filtrando e reescrevendo as palavras ...

- Quando energizado via cabo de alimentação e botão liga/desliga na posição *I*, o equipamento entrará em modo repouso (lâmpada UV e luz indicadora apagadas), independente do estado da porta (aberta ou fechada).
- Quando em modo repouso, se acionar o botão *Start* e se a porta estiver fechada, será iniciado o modo temporizando (lâmpada UV acesa e a luz indicadora piscante continuamente) e assim permanecerá por 10 minutos. Ao final, entrará em modo finalizado (lâmpada UV apagada e o luz indicadora acesa continuamente).
- Quando em modo temporizando, se for aberta a porta, entrará em modo pausado (lâmpada UV apaga, a luz indicadora piscante com duas piscadas curtas a cada 1s e contagem de tempo pausada).

• Estudo de caso (*cura de material com luz UV*)

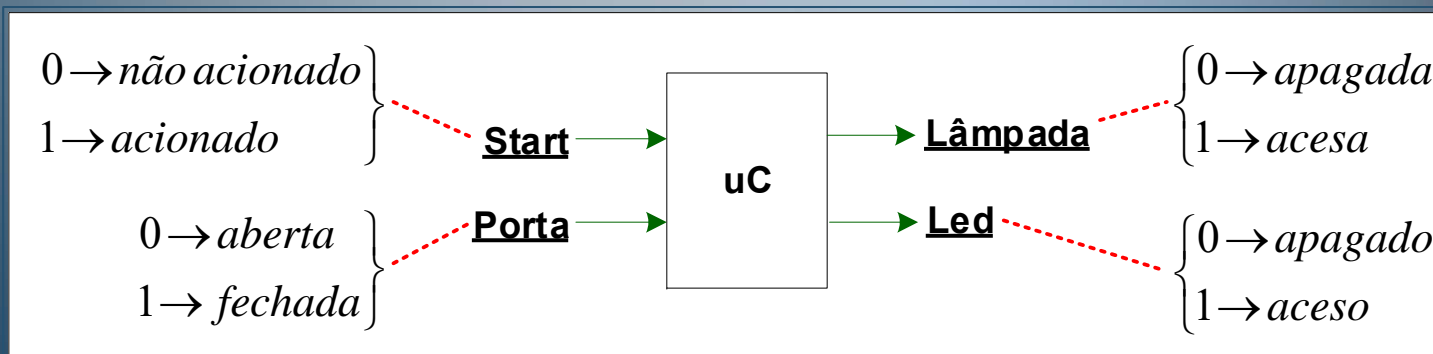
• Filtrando e reescrevendo as palavras ...

- Quando em modo pausado, se for fechada a porta, retornará ao modo temporizando até concluir-se o tempo de 3 minutos, quando entrará em modo finalizado.
- Quando em modo finalizado, se acionado o botão Start, nada acontecerá, se for aberta a porta, entrará em modo repouso.
- Quando em modo temporizado ou modo pausado, se necessário interromper o processo e retornar ao modo repouso, é necessário desligar (botão liga/desliga na posição **O**) e ligar novamente o equipamento (botão liga/desliga na posição **I**).
- Para um acionamento válido do botão **Start**, o mesmo deve ser acionado por pelo menos 3s.
- O acionamento do botão **Start** só é válido apenas no modo repouso.

Estudo de caso (*cura de material com luz UV*)

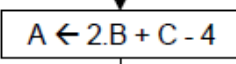
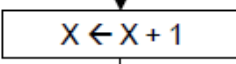

• Uma solução lógica (dentre várias)...

- Representada por um Fluxograma
- Tradução para o código em linguagem C/C++
 - Independente do microcontrador
 - Nesse momento interessam apenas os “estados das coisas”



Estudo de caso (cura de material com luz UV)

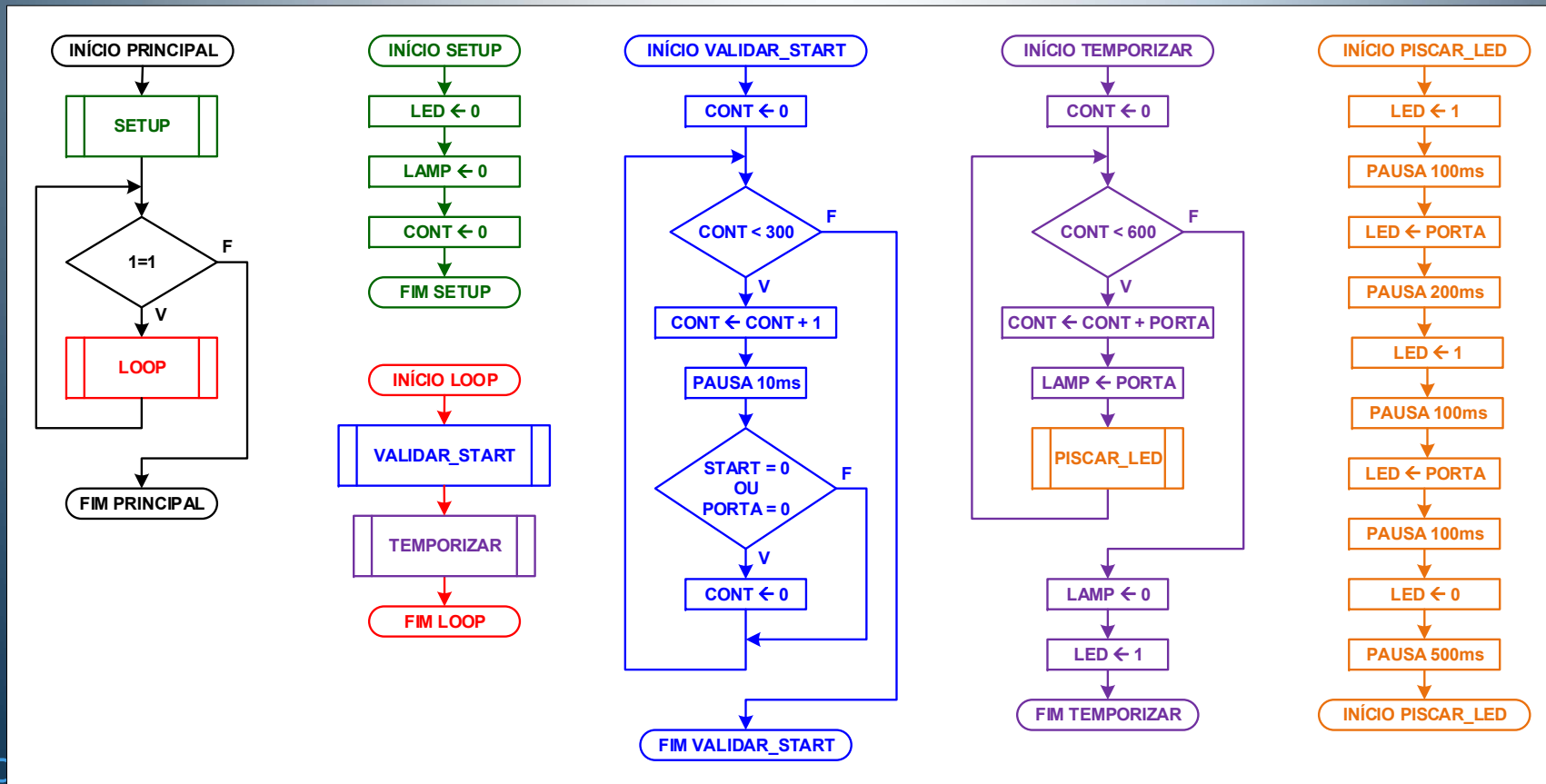
Tradução?

<div>  </div>	<div>  </div>	<div>  </div>
$a = 2*b + c - 4;$	$x = x + 1; \text{ OU } x++;$	
contador < 10 (menor que) contador < 10	$x \geq 5$ (maior ou igual a) $x \geq 5$	
$x \leq 5$ (menor ou igual a) $x \leq 5$	$a = 1$ (igual a) $a == 1$	
$a \leftarrow \bar{b}$ (atribui o complemento) $a = !b$	$a \neq 0$ (diferente de) $a != 0$	
$z \leftarrow x / y$ ou $z \leftarrow x \div y$ (divisão) $z = x / y$	$a = 1 \text{ E } b > 5$ (2 ou mais condições) $a == 1 \ \&\& \ b > 5$	
$a < 5 \text{ OU } a > 10$ (2 ou mais condições) $a < 5 \ \ a > 10$	$a < 5 \text{ OU } a > 10 \text{ E } b \neq c$ (2 ou mais condições) $a < 5 \ \ a > 10 \ \&\& \ b != c$	

$a \leftarrow \bar{b}$ (atribui o complemento)	$a = !b$	$a = 0$ (diferente de)	$a != 0$
$z \leftarrow x / y$ ou $z \leftarrow x \div y$ (divisão)	$z = x / y$	$a = 1 \text{ E } b > 5$ (2 ou mais condições)	$a == 1 \ \&\& \ b > 5$
$a < 5 \text{ OU } a > 10$ (2 ou mais condições)	$a < 5 \ \ a > 10$	$a < 5 \text{ OU } a > 10 \text{ E } b \neq c$ (2 ou mais condições)	$a < 5 \ \ a > 10 \ \&\& \ b != c$

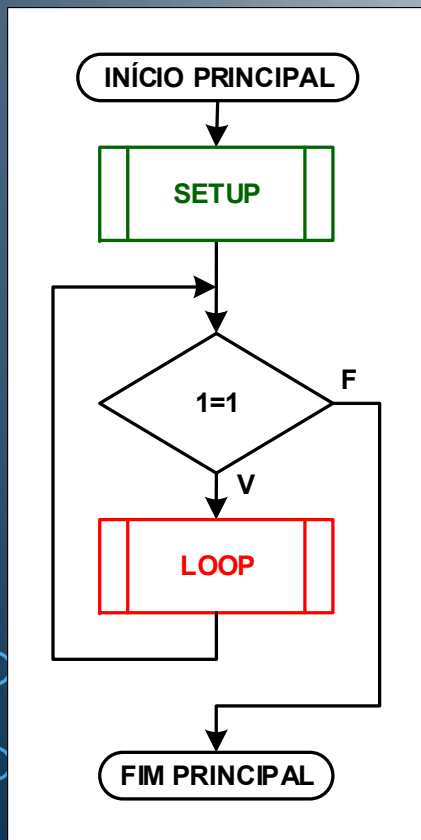
Estudo de caso (*cura de material com luz UV*)

• Fluxograma



Estudo de caso (*cura de material com luz UV*)

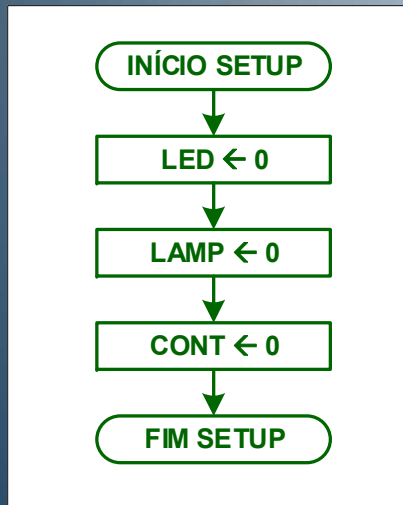
• Tradução do Fluxograma (por partes)



```
void main() {  
    setup(); // chama função setup()  
    while (1) {  
        loop(); // chama função loop()  
    }  
}
```

Estudo de caso (*cura de material com luz UV*)

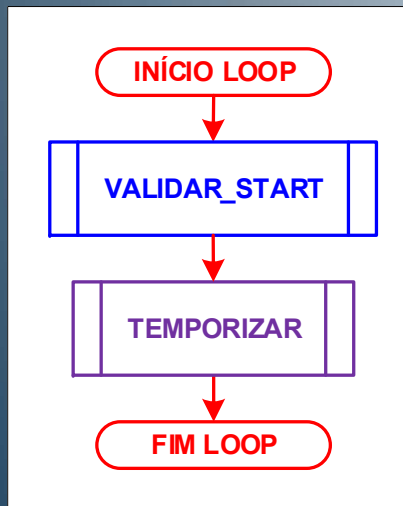
• Tradução do Fluxograma (por partes)



```
//  
// Função Setup =====  
//  
void setup() {  
    led = 0;  
    lamp = 0;  
    cont = 0;  
}
```

Estudo de caso (*cura de material com luz UV*)

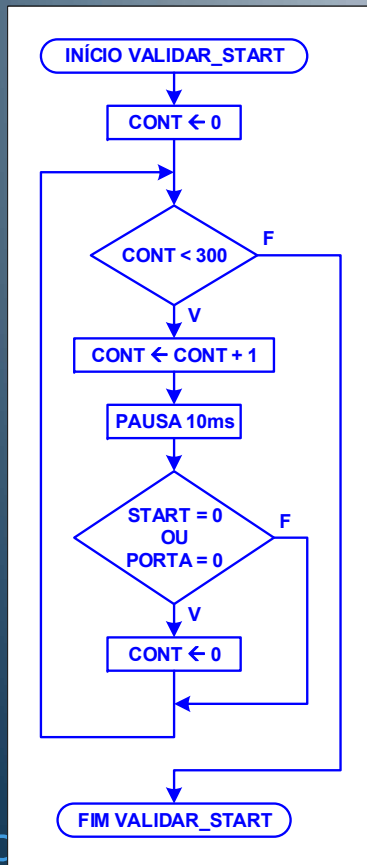
• Tradução do Fluxograma (por partes)



```
//  
// Função loop =====  
//  
void loop() {  
  validar_start(); // chama função validar_start()  
  temporizar(); // chama função temporizar()  
}
```


Estudo de caso (*cura de material com luz UV*)

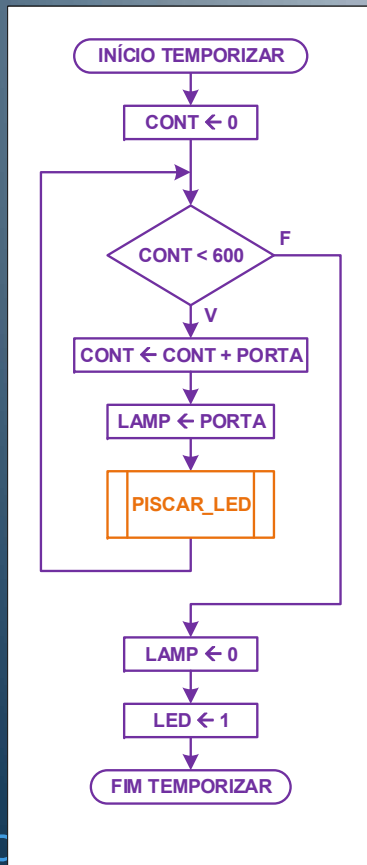
• Tradução do Fluxograma (por partes)



```
//  
// Função validar_start =====  
//  
void validar_start() {  
    cont = 0;  
    while (cont < 300) {  
        cont = cont + 1;  
        delay_ms(10);  
        if (!start || !porta) {  
            cont = 0;  
        }  
    }  
}
```

Estudo de caso (*cura de material com luz UV*)

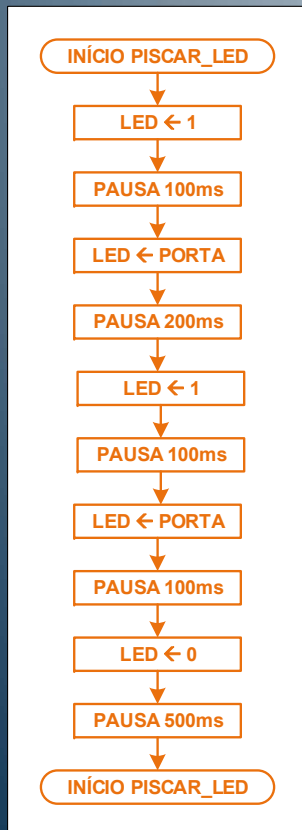
• Tradução do Fluxograma (por partes)



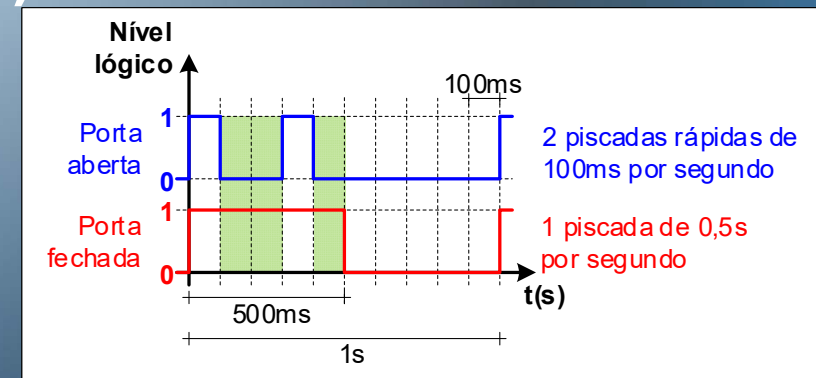
```
//  
// Função temporizar =====  
//  
void temporizar() {  
    cont = 0;  
    while (cont < 30) {  
        cont = cont + porta;  
        lamp = porta;  
        pisca_led(); // chama função pisca_led()  
    }  
    lamp = 0;  
    led = 1;  
}
```

Estudo de caso (*cura de material com luz UV*)

• Tradução do Fluxograma (por partes)

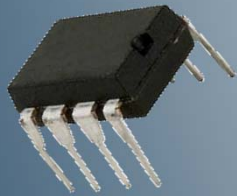


```
//  
// Função pisca_led =====  
//  
void pisca_led() {  
    led = 1; delay_ms(100);  
    led = porta; delay_ms(200);  
    led = 1; delay_ms(100);  
    led = porta; delay_ms(100);  
    led = 0; delay_ms(500);  
}
```

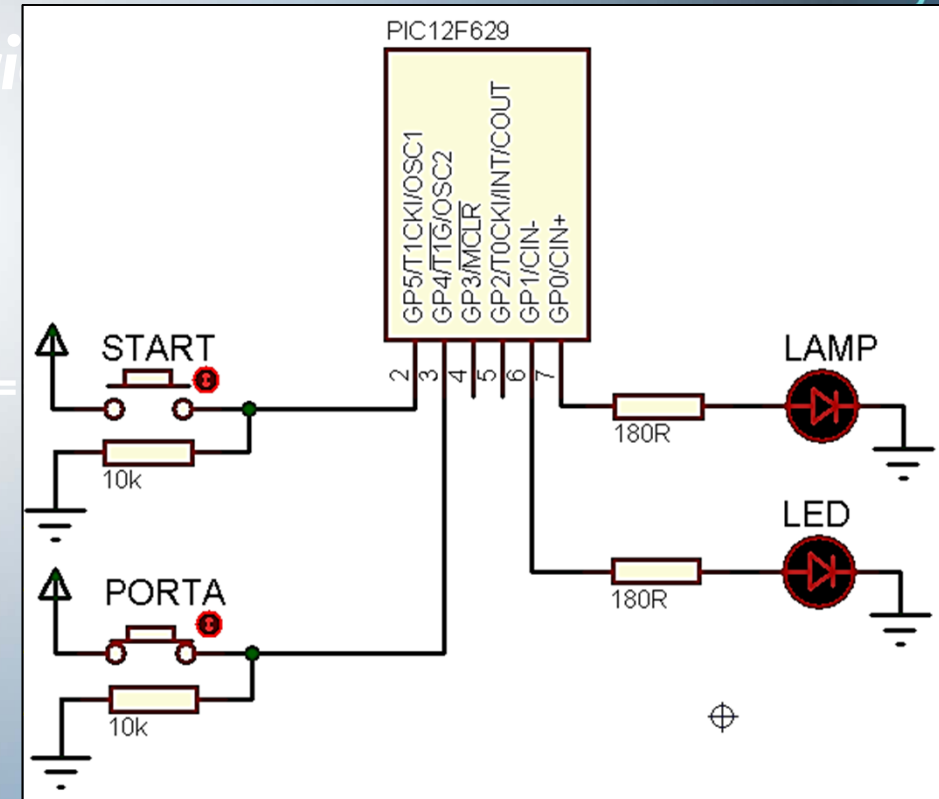


Estudo de caso (*cura de materi*)

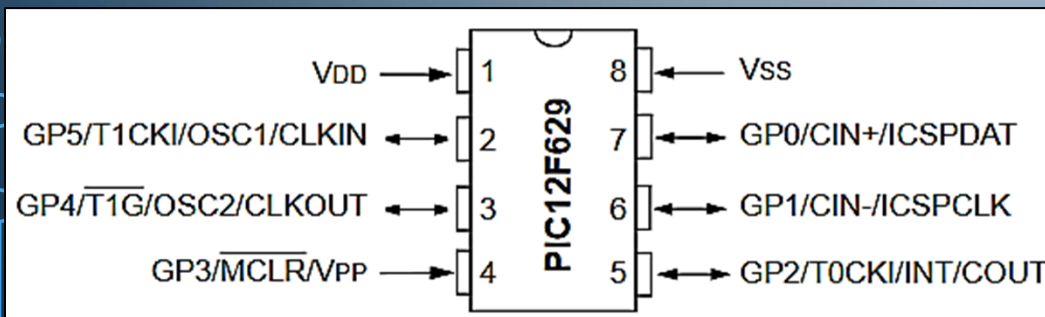
Ajuste para o PIC12F629



```
//  
// Apelidos dos pinos ==  
//  
sbit start at gp5_bit;  
sbit porta at gp4_bit;  
sbit lamp at gp0_bit;  
sbit led at gp1_bit;
```

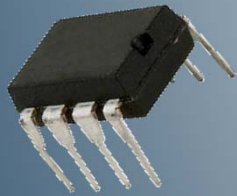


Conforme as
conexões
físicas com os
pinos do CI

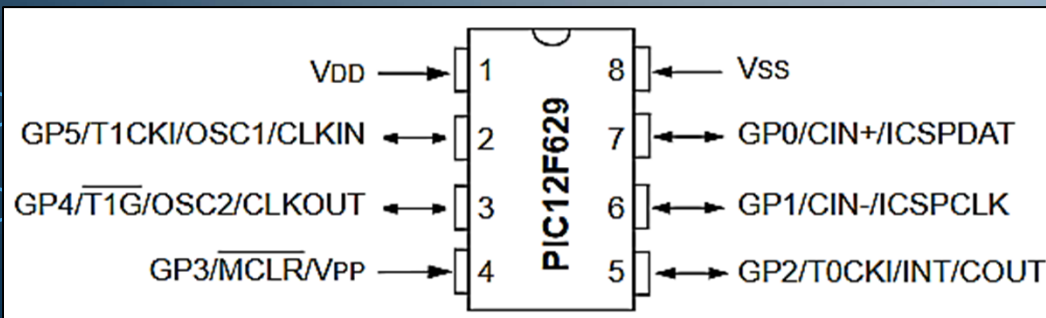


Estudo de caso (*cura de material com luz UV*)

Ajuste para o PIC12F629



```
//  
// Função Setup =====  
//  
void setup() {  
    intcon = 0x00;    // especifica  
    cmcon = 0x07;     // para o  
    trisio = 0b111100; // PIC 12F629  
  
    led = 0;
```



Tristate da porta GP

GP5	GP4	GP3	GP2	GP1	GP0
1	1	1	1	0	0

1 = entrada e 0 = saída

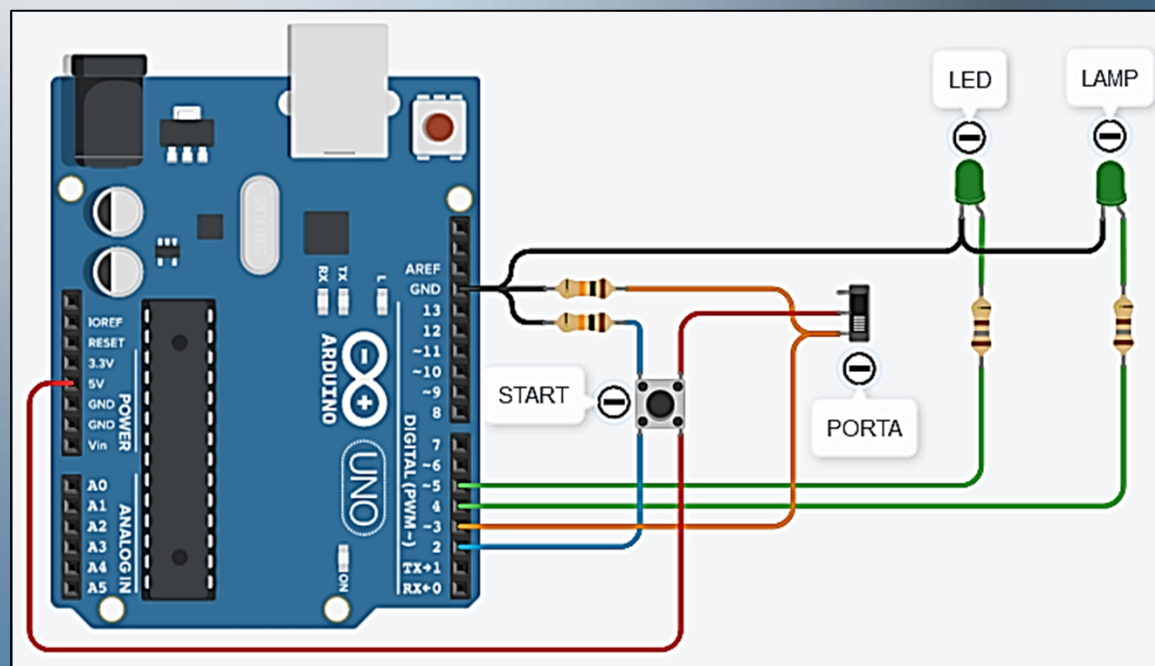
Estudo de caso (*cura de material com luz UV*)

• Ajuste para o Arduino UNO



```
//  
// Apelidos dos pinos =====  
//  
byte start = 2;  
byte porta = 3;  
byte lamp = 4;  
byte led = 5;
```

Conforme as
conexões
físicas com os
pinos da placa



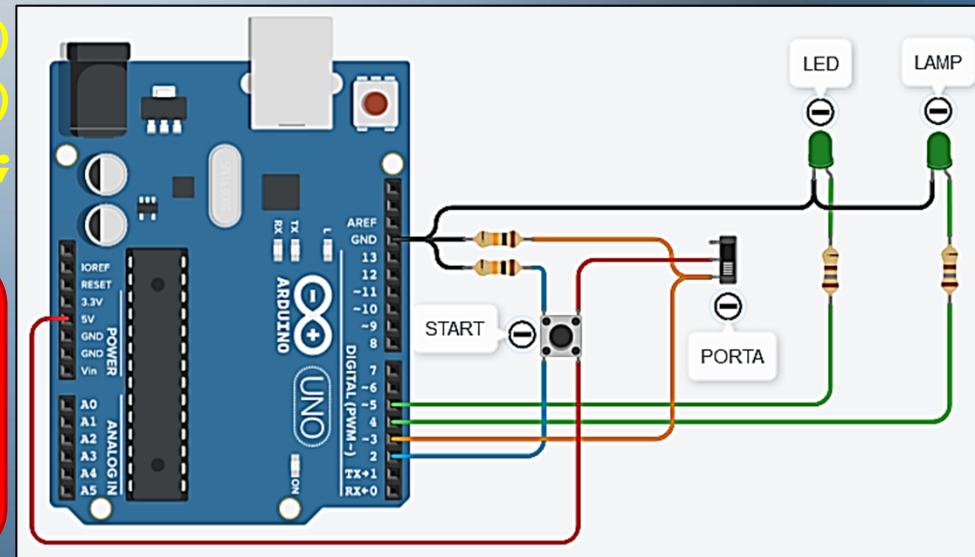
Estudo de caso (*cura de material com luz UV*)

Ajuste para o Arduino UNO



```
//  
// Função Setup =====  
//  
void setup() {  
  pinMode(start, INPUT); // específico  
  pinMode(porta, INPUT)  
  pinMode(lamp, OUTPUT)  
  pinMode(led, OUTPUT);  
  
  led = 0;  
  lamp = 0;  
  cont = 0;  
}
```

Conforme as
conexões
físicas com
os pinos da
placa



Estudo de caso (*cura de material com luz UV*)

• Ajuste para o Arduino UNO



```
delay_ms(100); → delay(100);
```

```
lamp = 0; → digitalWrite(lamp, LOW);
```

```
led = 1; → digitalWrite(led, HIGH);
```

```
cont = cont + porta;
```

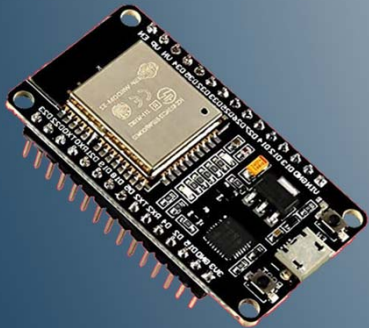
```
→ cont = cont + digitalRead(porta);
```

```
led = porta
```

```
→ digitalWrite(led, digitalRead(porta));
```

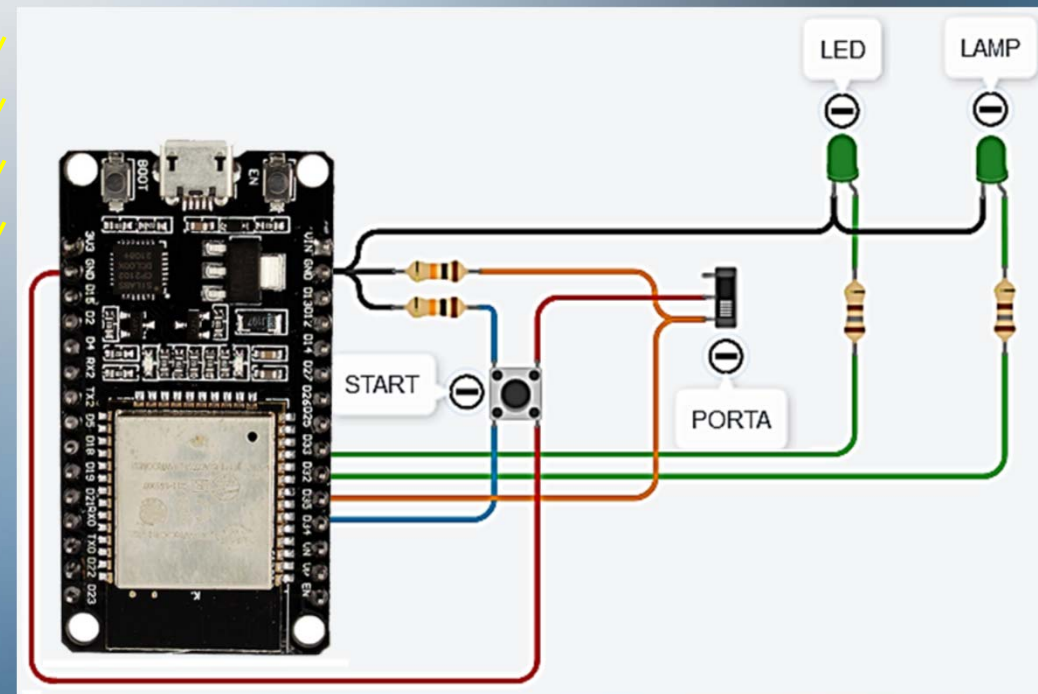
Estudo de caso (*cura de material com luz UV*)

- Ajuste para o ESP32 (tem vários modelos)



```
//  
// Apelidos dos pinos =====  
//  
byte start = D34; //  
byte porta = D35; //  
byte lamp = D32; //  
byte led = D33; //
```

Conforme as
conexões
físicas com os
pinos da placa



○ Obrigado pela sua atenção!!

Dúvidas?

Comentários?

Jorge Sebastião Canova

jorge.canova@etec.sp.gov.br

jorgecanova@gmail.com