

**TÍTULO: Sensores de conexão**

CENÁRIO DE APRENDIZAGEM	
<b>Escola:</b>	<b>Duração (minutos):</b> 90
<b>Professor(a):</b>	<b>Idade dos alunos:</b> 14

<b>Ideia chave:</b>	<b>Sensores de conexão</b>
---------------------	----------------------------

**Tópicos:**

- Os alunos aprofundam a compreensão do significado, potencial e riscos da programação a nível da sociedade.
- Os alunos aprendem a usar a inteligência artificial.

**Objetivo:**

- Os alunos são capazes de projetar, criar, documentar e apresentar programas e robôs que resolvem um problema específico da vida real. Os programas criados incluem algoritmos de pesquisa, tabelas e funções automáticas. Vários eventos simultâneos acontecem nesses programas.

**Resultados:**

- Os alunos criam jogos, aplicativos ou aplicativos móveis mais complexos que simulam os assuntos.
- Os alunos aprendem sobre o potencial e os recursos de microcontroladores mais avançados.

**Formas de trabalho:**

- trabalho individual
- trabalho de pares
- trabalho de grupo

**Métodos:**

- apresentação
- discussão
- exercício interativo

## ARTICULAÇÃO

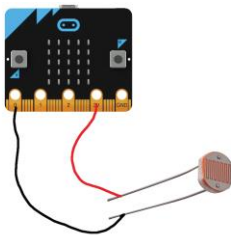
### Curso de ação (duração, minutos)

## INTRODUÇÃO

O professor inicia a discussão com os alunos:  
 Podemos adicionar vários sensores ao micro:bit.

## PARTE PRINCIPAL

Um dos sensores que é facilmente incorporado é o foto sensor GL5528. Trata-se basicamente de um resistor que muda a sua resistência dependendo da quantidade de luz que incidir sobre ele. Tem dois pinos e não precisamos de nos preocupar com a polaridade quando ligados. Um dos pinos está ligado ao pino 0 no micro:bit, enquanto o outro está ligado ao pino 3 V. (Também podes ligá-lo ao pino GND em vez do pino 3 V).



O próximo programa lê o valor do fotossensor e exibe-o no écran do micro: bit.  
 O comando `pin0.read_analog()` retorna um valor entre 0 e 1023.  
 Dependendo do tipo de sensor fotográfico que estamos a usar, como ele está ligado e quanta luz está a incidir sobre ele, os valores que obtemos estarão entre os números 0 e 1023.  
 Se colocarmos a mão sobre o sensor, verás como o valor no écran irá variar.

```

1 from microbit import *
2
3 while True:
4     ocitanje = pin0.read_analog()
5     display.scroll(str(ocitanje))
    
```

## EXERCÍCIO

De acordo com o exemplo anterior, os alunos podem projetar, criar e testar seus próprios programas.

### Exemplos:

<https://makecode.microbit.org/reference/input>  
[onButtonPressed](#)  
[onGesture](#)

[onPinPressed](#)  
[onPinReleased](#)  
[buttonIsPressed](#)  
[pinsPressedisgesture](#)  
[compassHeading](#)  
[temperature](#)  
[acceleration](#)  
[lightLevel](#)  
[rotation](#)  
[magneticForce](#)  
[runningTime](#)  
[setAccelerometerRange](#)  
[calibrate-compass](#)

### CONCLUSÃO

Alunos e professor discutem e avaliam as soluções apresentadas.

#### **Métodos**

entrevista de apresentação  
demonstração de discussão  
trabalhar na representação de papéis do texto  
trabalho gráfico  
exercício / simulação interativa no computador

#### **Formas de trabalho**

trabalho individual  
trabalho em em pares  
trabalho em equipa  
trabalho frontal

#### **Material:**

- micro:bit

#### **Bibliografia:**

- <https://makecode.microbit.org/reference/input>  
[onButtonPressed](#)  
[onGesture](#)  
[onPinPressed](#)  
[onPinReleased](#)

[buttonsPressed](#)  
[pinsPressedisgesture](#)  
[compassHeading](#)  
[temperature](#)  
[acceleration](#)  
[lightLevel](#)  
[rotation](#)  
[magneticForce](#)  
[runningTime](#)  
[setAccelerometerRange](#)  
[calibrate-compass](#)

OBSERVAÇÕES PESSOAIS, COMENTÁRIOS E NOTAS