

# INTRODUCERE - SENZORUL DE CULOARE

BY SANJAY AND ARVIND SESHAN

This lesson uses SPIKE 3 software

# OBIECTIVELE LECȚIEI

- Învăță cum să folosești senzorul de culoare



# Ce este senzorul de culoare?

- Senzorul API poate citi fie culoarea, fie reflexivitatea luminii măsurate
- Spre deosebire de EV3, reflexivitatea este măsurată în timp ce pe planșă este proiectată lumină albă, nu lumină roșie.
- Distanța optimă de citire este de 16 mm (variind în funcție de culoare, dimensiunea obiectului și suprafață).
- Senzorul poate detecta 8 culori (expuse în dreapta) și nicio culoare (None).
- Senzorul poate raporta reflectanța Este un procentaj de la 0(nici o reflectanță) la 100%(reflexivitate totală).
- Senzorul poate raporta valori brute pentru roșu, verde, albastru, intensitatea (RGBI) ca tuple

**Notă:** În teste v3.4

- Culoarea Orange nu poate fi recunoscută.
- Intervalele pentru RGBI sunt neclare și nu sunt documentate în lecția Knowledge Base.

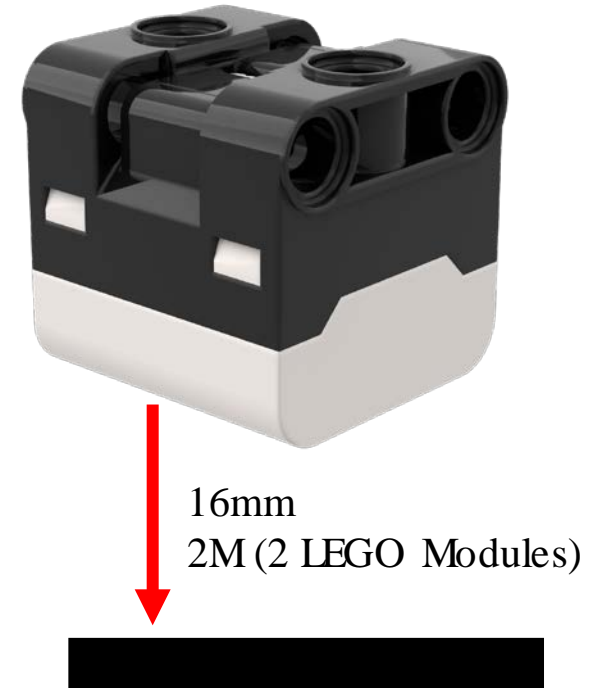
- ≡ AZURE
- ≡ BLACK
- ≡ BLUE
- ≡ GREEN
- ≡ MAGENTA
- ≡ ORANGE
- ≡ PURPLE
- ≡ RED
- ≡ TURQUOISE
- ≡ UNKNOWN
- ≡ WHITE
- ≡ YELLOW

# NOTĂ: ADB ȘI DETECTAREA DE CULOARE

*Senzorul de culoare de pe ADB (Advanced Driving Base in SPIKE Prime) este montat la aproximativ 8mm distanță de sol, dar distanța optimă pentru montare, conform specificațiilor, este de 16mm.*

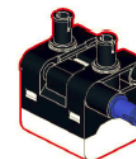
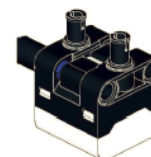
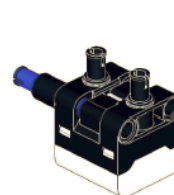
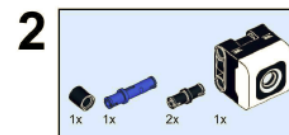
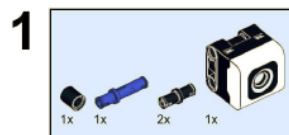
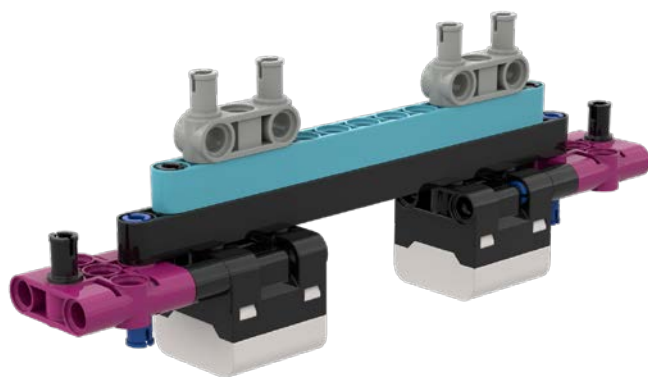
Când folosești acest design de robot, negru nu este citit corect în Color Mode, dacă utilizăm linii de bandă adezivă sau plașa FIRST LEGO League.

Verifică următorul slide pentru modificări. Instrucțiunile de construcție sunt disponibile într-un fișier separat pe site.

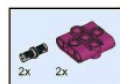


# MODIFICĂRI PENTRU ADB

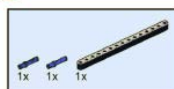
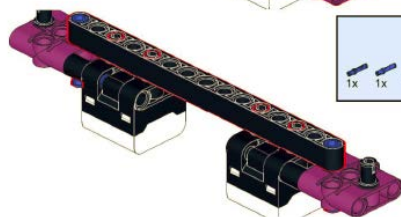
Instrucțiunile de modificare ale bumper-ului din față pentru ADB (astfel încât senzorul de culoare să fie ridicat cu o piesă LEGO mai sus) sunt incluse pe website.



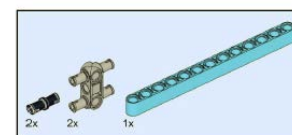
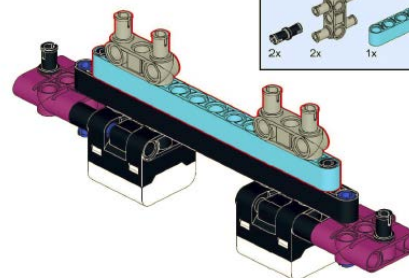
3



4



5



# CUM PROGRAMEZI CU UN SENZOR DE CULOARE?

- Sunt două moduri prin care puteți programa senzorul de culoare: Color Mode și Reflected light Mode

- Citește o culoare:

```
color_sensor.color(port.A)
```

↑  
Port

- Citește intensitatea:

```
color_sensor.reflection(port.A)
```

# Provocarea 1

■ Programează robotul să se miște înainte până când senzorul de culoare vede negru.

## ■ Pași de bază:

■ Scrie funcția `color_found` care să returneze „adevărat” dacă vede negru

■ Setează **mișcarea motorului** pentru robotul tău

■ Începe **mers drept înainte**

■ Așteaptă funcția `color_found`

■ Oprește **mișcarea**

■ Soluția de pe pagina următoare utilizează Drive Base 1 cu un senzor de culoare pe portul A. Ajustează programul în consecință.

# PROVOCAREA 1: SOLUȚIE UTILIZÂND CULOAREA

```
from hub import port
import runloop, motor_pair, sys, color_sensor, color

■
    # Funcția care să returneze „adevărat” dacă vede negru
def color_found():
    return color_sensor.color(port.A) == color.BLACK

async def main():
    # Setează pechea de motoare de tracțiune și începe mișcarea
    motor_pair.pair(motor_pair.PAIR_1, port.C, port.D)
    motor_pair.move(motor_pair.PAIR_1, 0)
    # așteaptă pînă când culoarea este găsită
    await runloop.until(color_found)
    # oprește și ieși
    motor_pair.stop(motor_pair.PAIR_1)
    sys.exit("Done")

runloop.run(main())
```



# PROVOCAREA 1: SOLUȚIE UTILIZÂND CULO AREA

```
from hub import port
import runloop, motor_pair, sys, color_sensor
```

```
# Funcția returnează „adevărat” când intensitatea luminii reflectate este mai mică decât 50.
# Această funcționează pentru negru-alb
# Planșa. Dacă planșa ta are mai multe culori, utilizează valori de referință mai mici pentru negru.
```

```
def intensity_found():
    return color_sensor.reflection(port.A) < 50
```

```
async def main():
    # Setează perechea de motoare pentru tracțiune și începe mișcarea
    motor_pair.pair(motor_pair.PAIR_1, port.C, port.D)
    motor_pair.move(motor_pair.PAIR_1, 0)
    # așteaptă până când valoarea intensității luminii reflectate este îndeplinită
    await runloop.until(intensity_found)
    # oprește și ieși
    motor_pair.stop(motor_pair.PAIR_1)
    sys.exit("Done")
runloop.run(main())
```

# CREDITS

Această lecție a fost creată de Sanjay Seshan și Arvind Seshan for SPIKE Prime Lessons

La această lecție au contribuit membrii comunității FLL Share & Learn.

Mai multe lecții sunt disponibile pe [www.primelessons.org](http://www.primelessons.org)

Această lecție a fost tradusă în limba română de echipa de robotică FTC – ROSOPHIA #21455 RO20



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).