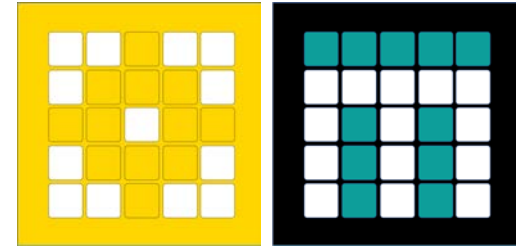


PRIME LESSONS

By the Makers of EV3Lessons



CONFIGURAREA MIȘCĂRII ROBOTULUI

BY SANJAY AND ARVIND SESHAN

This lesson uses SPIKE 3 software

OBIECTIVELE LECȚIEI

- Învățăm cum să facem ca robotul să meargă înainte și înapoi
- Învățăm să utilizăm metodele Motor Pair Move

CREAREA UNUI OBIECT MOTOR PAIR

- Mișcarea de bază este realizată utilizând obiectul MotorPair
 - Vezi lecția Configurarea Mișcării robotului pentru detalii despre crearea acestui obiect
- Slide-urile următoare vor acoperi diferite metode de utilizare a obiectelor pentru mișcare
 - Eg., `motor_pair.move_for_degrees(<parameters>)`
- Observați că spre deosebire de SP2, SP3 nu are obiecte. Vei utiliza funcții pentru a opera motor pair, și pot trece în slot-ul pereche de fiecare dată.

Nu inițializați mai mult de un motor pair cu aceleași porturi → acesta este redundant, doar va irosi memoria și va cauza conflicte indezirabile

FUNCȚIA Motor pair

- move
- move_for_degrees
- move_for_time
- move_tank
- move_tank_for_degrees
- move_tank_for_time
- pair
- stop
- unpair
- Vom acoperi funcțiile cu galben în această lecție

MOTOR_PAIR.PAIR()/UNPAIR()

Pentru a seta un motor pair, trebuie să împerechezi 2 motoare și să le desemnați unul din cele 2 nume disponibile.

Cum există doar 6 porturi pe HUB, nu poți mișca mai mult de 3 perechi de motoare.

Pentru Drive Base 1

```
motor_pair.pair(motor_pair.PAIR_1, port.C, port.D)
```

↑
Pair slot

↑
Left
motor

↑
Right
motor

Din momentul în care am setat o pereche, poți utiliza comenzi de mișcare cu numele perechii de motoare. Nu trebuie să utilizezi numele porturilor.

Pentru a reseta numele, utilizează unpair:

```
motor_pair.unpair(motor_pair.PAIR_1)
```

MOTOR_PAIR.MOVE()

Move este o funcție **synchronous** care mișcă perechea de motoare până când este primit un stop.

Diagram illustrating the parameters of the `MOTOR_PAIR.MOVE()` function:

Pair slot (indicated by a downward arrow pointing to `pair`)

Direcția și cantitatea prin care se întoarce robotul (indicated by a downward arrow pointing to `steering`)

The acceleration (deg/sec²) (indicated by a downward arrow pointing to `acceleration`)

Motor speed (deg/sec) (indicated by an upward arrow pointing to `velocity`)

Range for steering: -100 to 100

Range for acceleration: 0 to 10000

Range for motor speed:

- Small motor (essential): -660 to 660
- Medium motor: -1110 to 1110
- Large motor: -1050 to 1050

Example function call: `.move(pair, steering, velocity = 360, acceleration = 1000)`

`velocity=360` and `acceleration=1000` sunt valori predefinite dacă nu sunt setate altfel. Valorile pozitive întorc robotul spre dreapta, valorile negative spre stânga. Valorile mari întorc mai mult, e.g., 0 merge înainte, +/-50 mișcă doar o roată, +/-100 face robotul să se rotească

MOTOR_PAIR.STOP()

Stop este o funcție **synchronous** care oprește o comandă motor pair

```
.stop(pair, stop=motor.BRAKE)
```

Vezi lecția “Configuring Robot Movement” pentru o descriere a modurilor Stop. Lecția Knowledge Base are de asemenea informații bune.

MOTOR_PAIR.MOVE_FOR_DEGREES()

- Comanda `move_for_degrees` mișcă perechea de motoare cu un anumit număr de grade.
- Este o funcție **asynchronous** – utilizează **await** dacă dorești să aștepti până când comanda finalizează task-ul.
- De obicei, ne dorim ca robotul să parcurgă o anumită distanță
- Vezi lecția “Configuring Robot Movement” pentru o descriere despre cum să scrii o funcție pentru parcurgerea unei distanțe și întoarcere un anumit număr de grade motor pair are nevoie să întoarcă robotul pe o anumită distanță. Depinde de caracteristicile fizice ale robotului tău.

```
.move_for_degrees(pair, degrees, steering=0, velocity = 360,  
                  stop = motor.BRAKE, acceleration = 1000)
```


MOTOR_PAIR.MOVE_TANK/MOVE_TANK_FOR_DEGREES()

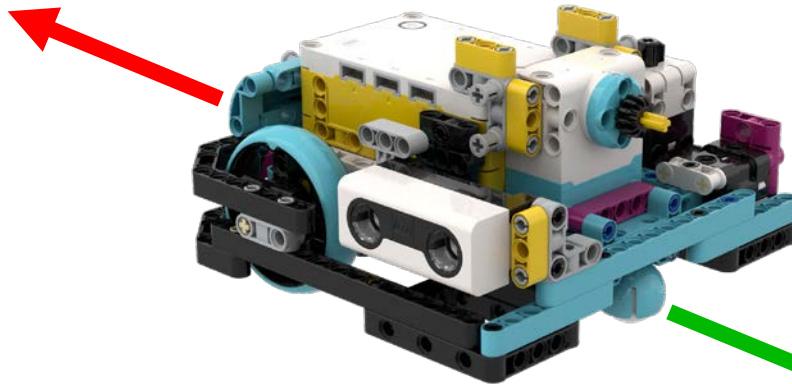
■ Aceste funcții sunt similare cu mișcare și mișcare pe grade, cu excepția cazului în care te lasă să setezi viteze individuale pentru motorul stânga și dreapta.

■ Poate fi util pentru a mișca fiecare roata independent, ex. Întoarcere sau aliniere la linie.

VALORI NEGATIVE

- Poți introduce valori pentru viteză sau grade
- Aceasta va face ca robotul să meargă înapoi
- Dacă ai dublă negație (e.g., viteză sau grade negative), robotul va merge înainte.

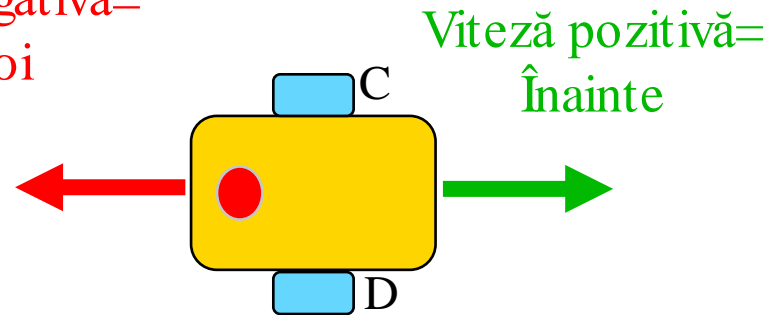
Viteză negativă=
Înapoi



Viteză negativă=
Înapoi

Viteză pozitivă=
Înainte

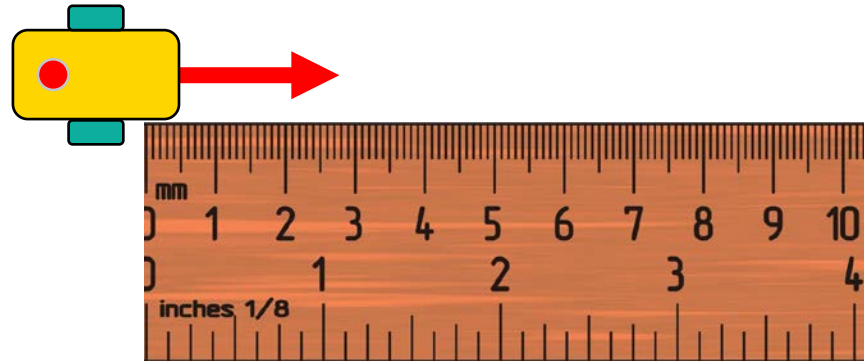
Drive Base 1



PROVOCAREA 1: MERS ÎNAINTE 10 CM

- Robotul merge înainte 10 centimetri
- Pași de bază:
 - Configurează robotul
 - Utilizează metoda MotorPairs (move_for_degrees() or move_tank_for_degrees()) pentru a mișca înainte 10cm

Drive Base 1



PROVOCAREA 1 SOLUȚIA

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

# cm, aceasta este o constantă a robotului tău
WHEEL_CIRCUMFERENCE = 17.5

# input-ul trebuie să fie aceeași cu WHEEL_CIRCUMFERENCE
def degreesForDistance(distance_cm):
    # Adaugă rația roților de transmisie dacă este nevoie
    return int((distance_cm/WHEEL_CIRCUMFERENCE) * 360)

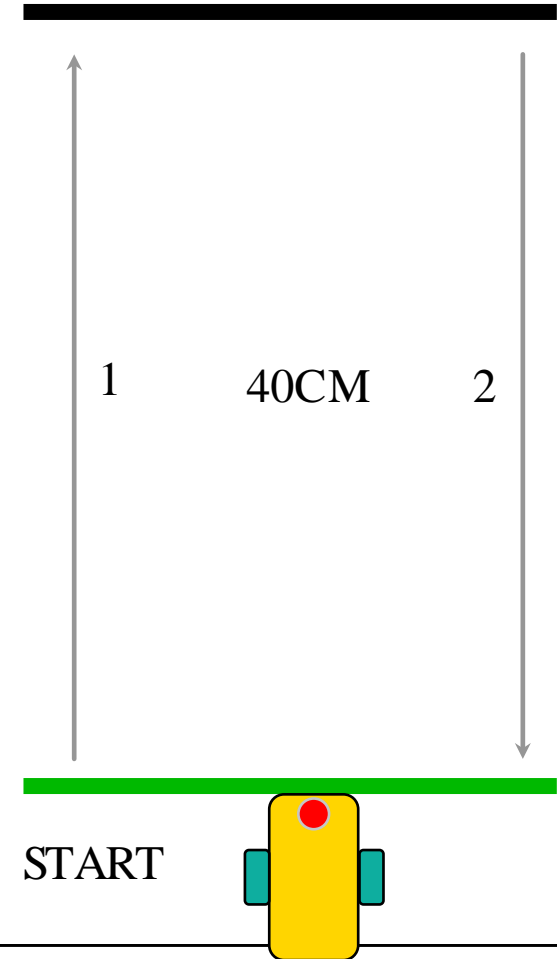
async def main():
    # Drive Base 1
    motor_pair.pair(motor_pair.PAIR_1, port.C, port.D)
    await motor_pair.move_for_degrees(motor_pair.PAIR_1, degreesForDistance(10), 0)
    sys.exit("Finished")

runloop.run(main())
```

PROVOCAREA II: MERS ÎNAINTE ȘI ÎN APOI

- Mișcați robotul înainte de la linia de start la linia de finish (1) și înapoi la start (2)
- Pași de bază:
 - Configurați robotul
 - Utilizați funcția MotorPair și mergeți înainte pentru distanța dorită (40cm)
 - Utilizați aceeași funcție MotorPair pentru mersul înapoi (40cm)

FINISH



PROVOCAREA 11 SOLUȚIA

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

# cm, aceasta este o constantă a robotului tău
WHEEL_CIRCUMFERENCE = 17.5

# input-ul trebuie să fie aceeași cu WHEEL_CIRCUMFERENCE
def degreesForDistance(distance_cm):
    # Adaugă rația roților de transmisie dacă este nevoie
    return int((distance_cm/WHEEL_CIRCUMFERENCE) * 360)

async def main():
    # Drive Base 1
    motor_pair.pair(motor_pair.PAIR_1, port.C, port.D)
    await motor_pair.move_for_degrees(motor_pair.PAIR_1, degreesForDistance(40), 0)
    await motor_pair.move_for_degrees(motor_pair.PAIR_1, degreesForDistance(-40), 0)
    sys.exit("Finished")

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

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/).