

# EV3 LEGO-prosjekt 1

## Rapport

<b>Gruppe 4 .....</b>
<b>IDATG1004 - Teambasert Samhandling .....</b>
<b>8. september - 21. september .....</b>

# **Innhold**

<b>1 Innledning</b>	<b>2</b>
<b>2 Problemstilling / Oppgavebeskrivelse</b>	<b>2</b>
<b>3 Planlegging</b>	<b>2</b>
<b>4 Bygging av roboten</b>	<b>2</b>
<b>5 Programmering</b>	<b>2</b>
<b>6 Testing og resultater</b>	<b>2</b>
<b>7 Diskusjon og feilkilder</b>	<b>3</b>
<b>8 Konklusjon</b>	<b>3</b>

# 1 Innledning

I dette prosjektet har vi jobbet med å programmere og kontrollere en LEGO EV3-robot ved hjelp av Python og biblioteket pybricks. Vi bygde en enkel robotbil med to motorer, og laget et program som viser meldingen «Hello World» på skjermen, kjører roboten i et rektangulært mønster, og avslutter med å spille av meldingen «Have a nice day». Prosjektet ga erfaring med bruk av for-løkker, motorstyring og grunnleggende kommunikasjon via lyd og LCD-skjerm på EV3-klossen.

## 2 Problemstilling / Oppgavebeskrivelse

I denne oppgaven skal vi bygge og programmere en robot. Når roboten er ferdig bygd og programmet starter, skal roboten vise teksten «Hello World» på LCD-skjermen. Deretter skal robotbilen bevege seg i et rektangulært mønster og fullføre en runde. Når runden er ferdig, skal roboten stoppe og gi en lydutgang ved å uttale meldingen «Have a nice day».

## 3 Planlegging

Vi starter med å fordele roller, slik at alle er klar over hva som er oppgaven til hver enkelt. Rollene fordeles som følger:

- Sadaf og Jørgen: Ansvar for å bygge roboten.
- Oliver: Ansvar for programmeringen.
- Helle og Sven: Ansvar for å opprette og vedlikeholde wiki-siden samt issue-boardet i GitLab.

## 4 Bygging av roboten

Lego-roboten ble bygget etter en oppskrift laget av LEGO Mindstorms EV3. Link til oppskriften vi benyttet: <https://assets.education.lego.com/v3/assets/blt293eea581807678a/b1tc8481dd2666822ff/5f8801e3f4f4cf0fa39d2fef/ev3-rem-driving-base.pdf>

## 5 Programmering

Programmet ble utviklet i Python med pybricks-biblioteket. Vi implementerte funksjoner for bevgelsene drive\_forward og turn\_right. Gjennom en for-løkke repeterte vi disse funksjonene slik at roboten utførte en rektangulær bevegelse.

## 6 Testing og resultater

Vi har gjennomført gjentatte tester av roboten og gjort nødvendige justeringer. Underveis har vi foretatt endringer av tidsintervaller i programmet, samt lagt til flere forklarende kommentarer for å gjøre koden mer tilgjengelig og lettere å forstå for utenforstående.

```

2   from pybricks.hubs import EV3Brick
3   from pybricks.ev3devices import Motor, TouchSensor, ColorSensor,
4   |           InfraredSensor, UltrasonicSensor, GyroSensor)
5   from pybricks.parameters import Port, Stop, Direction, Button, Color
6   from pybricks.tools import wait, Stopwatch, DataLog
7   from pybricks.robots import DriveBase
8   from pybricks.media.ev3dev import SoundFile, ImageFile
9
10  # Import necessary modules from the pybricks library
11
12
13  # Initialize EV3 and motors
14  ev3 = EV3Brick() # Create an instance of the EV3 brick
15  left_motor = Motor(Port.B) # Create an instance of the left motor
16  right_motor = Motor(Port.C) # Create an instance of the right motor
17
18
19  # Prints text on screen
20  ev3.screen.clear() # Clear the screen
21  ev3.screen.print("Hello World") # Print "Hello World" on the LCD screen
22  wait(2000) # Wait for 2 seconds
23  ev3.screen.clear() # Clears screen
24
25  # Function to drive straight
26  def drive_forward(speed, time_ms): # Parameters: speed and time in milliseconds
27      left_motor.run_time(speed, time_ms, Stop.BRAKE, False) # Left motor with parameters (speed, time, stop, wait)
28      right_motor.run_time(speed, time_ms, Stop.BRAKE, True) # Right motor with parameters (speed, time, stop, wait)
29
30  # Function to turn 90 degrees to the right
31  def turn_right(speed, time_ms): # Parameters: speed and time in milliseconds
32      left_motor.run_time(speed, time_ms, Stop.BRAKE, False) # Left motor with parameters (speed, time, stop, wait)
33      right_motor.run_time(-speed, time_ms, Stop.BRAKE, True) # Right motor with parameters (-speed, time, stop, wait)
34
35  # Drive a rectangle
36  for _ in range(4): # For-loop that repeats 4 times, with variable "_"
37      drive_forward(300, 2000) # Drive forward with speed 300, for 2 seconds
38      turn_right(300, 800) # Turn right with speed 300, for 0.8 seconds
39
40  # When finished, say "Have a nice day"
41  ev3.speaker.say("Have a nice day")

```

Figur 1: Utklipp av programmet

## 7 Diskusjon og feilkilder

Vi læerte underveis viktigheten av nøyaktig testing og små justeringer. Det var en del små variasjoner som kunne ført til at roboten ikke fulgte rektangelet helt presist. I tillegg, vi erfarte at samarbeid og ansvarfodeling var avgjørende for framgangen. Feilkider

## 8 Konklusjon

I dette prosjektet har vi lært mer om hvordan vi skal programmere roboten til å gjør det vi ønsker. For å programmere roboten ble vi mer kjent med for-løkker i python, samt å importere fra biblioteket pybricks.