Robotics with Lego EV3 + Scratch

Andrea Sterbini – sterbini@di.uniroma1.it

Robotics: a very compelling problem setting

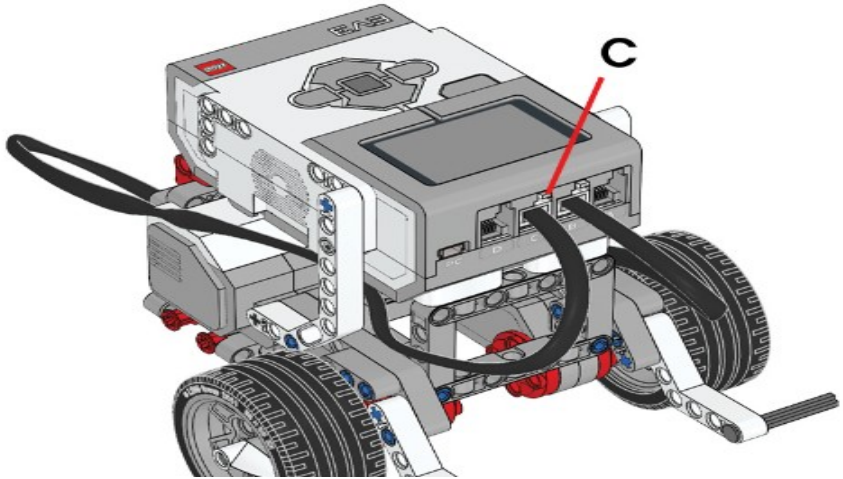
Using robots with kids allows you to:

- enhance motivation
- show concrete evidence of the program's actions
- force students to tackle CONCURRENT problems, e.g.:
  - read sensors WHILE moving
  - coordinate the movement of many motors (drive/arms)

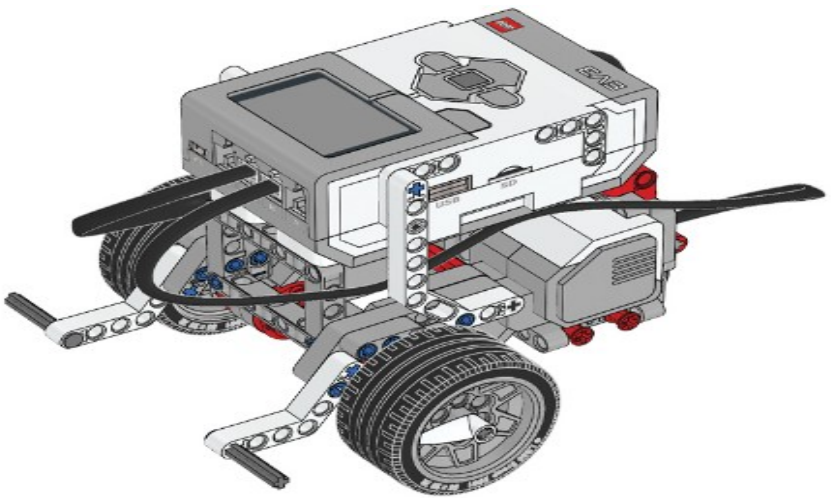
Scratch is VERY limited but, with Lego EV3, sufficient to build:

- a car moving in a labyrinth
- a robotic arm
- ...

Lego Mindstorms EV3 kit



To build a differential drive car you need:
- two independently controlled motors
- a distance sensor for front obstacles
- a light sensor to follow a line or to move towards a light

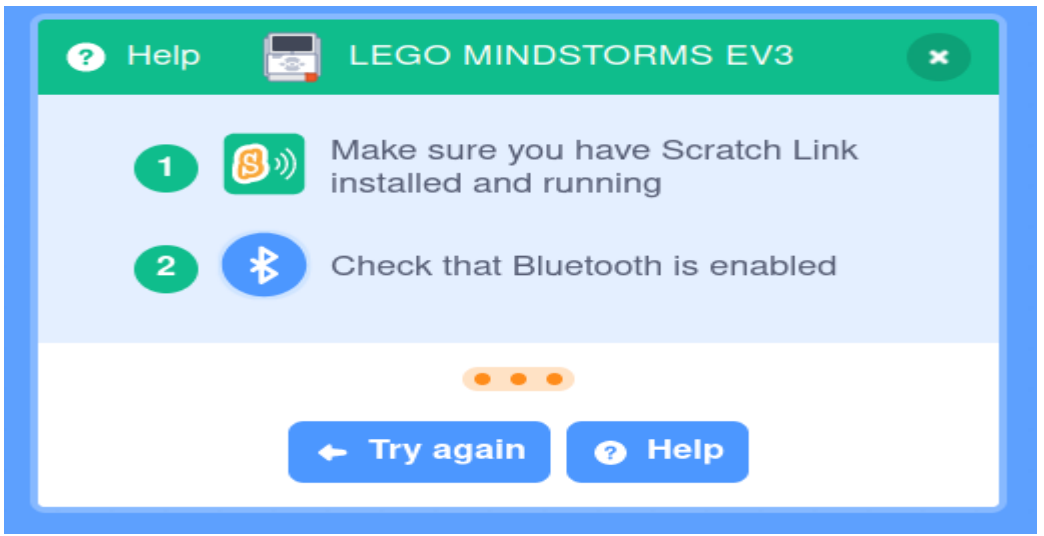


The “brick” has:

- 4 sensor ports (1, 2, 3, 4)
- 4 motor ports (A, B, C, D)
- 2 USB connections (slave + master)
- Bluetooth

Scratch for Lego Mindstorms EV3 or Lego Boost (Windows / MacOs / Chromebook / Android)

Install the Scratch Link driver (Windows)
Pair the EV3 Bluetooth with the PC
Add the EV3 extension to the Scratch project
NOTICE: It's added only if the Scratch Link connection is on
If all goes well you can use the EV3 blocks

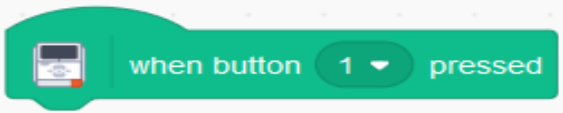


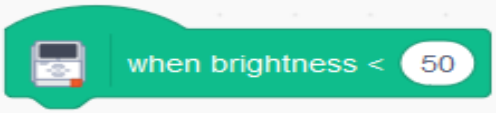
Mac/Chromebook/Android do not need the Scratch Link driver

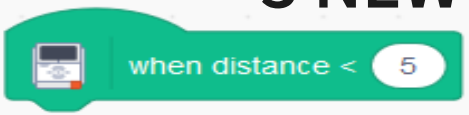
Linux: not available

Scratch Lego Mindstorms EV3/Boost extensions

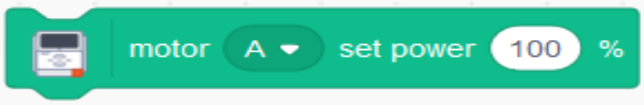
3 NEW EVENTS



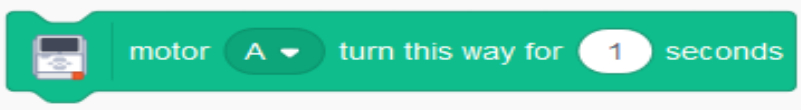


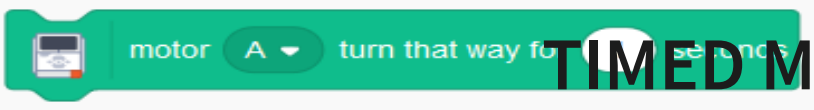


CONTROL MOTOR SPEED

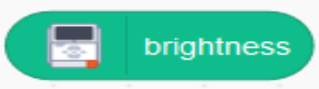


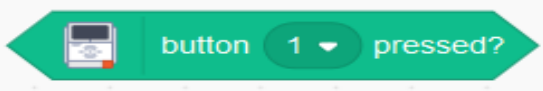
TIMED MOTORS



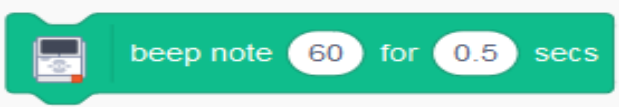


4 TYPES OF SENSORS





PLAY A NOTE FOR A GIVEN TIME



LIMITS of Lego EV3 Scratch extension: SENSORS

You can use ONLY ONE sensor for:

- DISTANCE (Ultrasound sensor)
  - DISTANCE IN INCHES??? (ask the student to convert to cm if needed)
- BRIGHTNESS (Light/colour sensor)
  - VERY LOW VALUES!!! (difficult to use)

The BUTTON-PRESSED event is somewhat erratic

The DISTANCE-LESS-THAN event works way better

The LIGHT-LESS-THAN event seems not to work!!! (!"$$"£!$!)

NOTICE:

the program actually runs in the browser and interacts with EV3 by Bluetooth

LIMITS of Lego EV3 Scratch extension: MOTORS

You can use UP TO 4

- motors on the A, B, C, D ports
- touch sensors on ports 1, 2, 3, 4

BUT: CANNOT rotate one motor for a given angle (ONLY TIMED run)

- Calibration is very important (e.g. time vs distance)

MOTOR POSITION SEEMS NOT TO WORK WELL (!$”$!”£|!$£|!)

OTHER:

- BLUETOOTH IS “BLOODY TRICKY” … (!|$!%£$!”%!”)

CONCURRENCY for Robotics in Scratch

Good reason to use Scratch

You must coordinate:

- many motors (at least 2 for differential movement)
- reading many sensors

You can define multiple threads for the same event/MESSAGE

- unfortunately, messages DO NOT carry arguments
  - (HACK: use global variables)

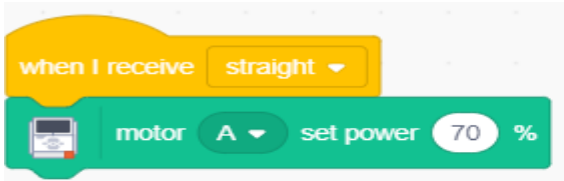
Choose a simple parametrization of single movements and JOIN

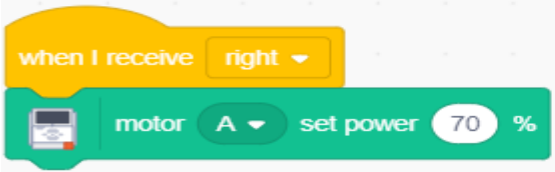
- time, speed, direction for each motor
- to get forward/backward, curves

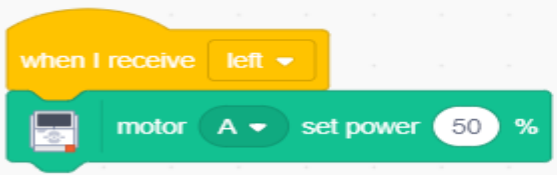
Example: Line Follower coordinating two motors with messages

Using messages to coordinate concurrent actions on both motors

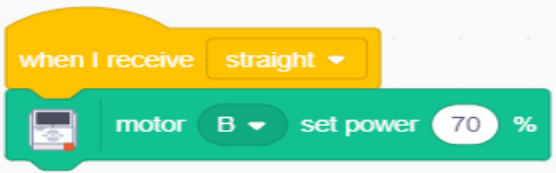
MOTOR A

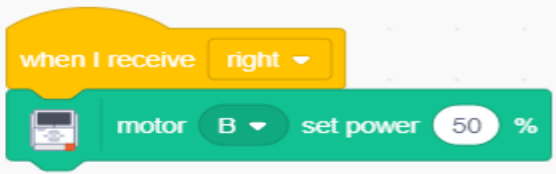


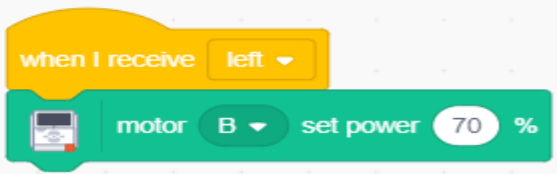




MOTOR B

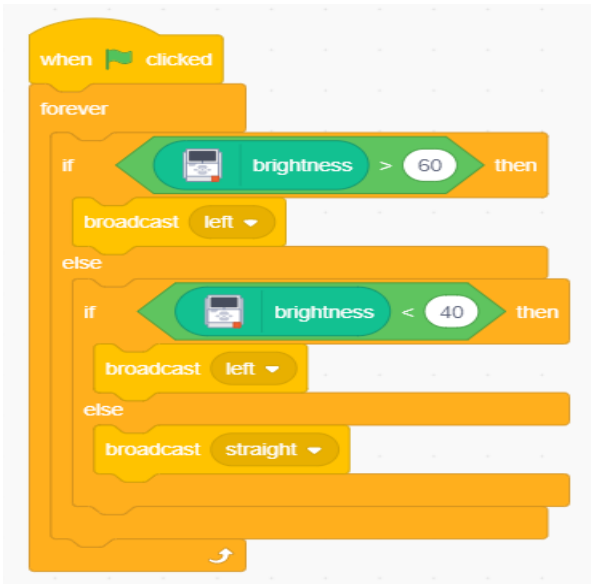






CONTROLLER

Stay on the line border where reflected light is circa 50%



Robots and Concurrency:
New factors to take into cosideration

You must calibrate movements and sensors values

Consider the possible compositions of concurrent actions
(and add semaphores to exclude incompatible actions)
(and remember to unlock semaphores)

Decide if an action should be Blocking/Non-blocking
i.e. if you must wait or not for message completion