24

THE CLEANEST DUSTMAN

STEMJAM Teaching Guide

Developing make spaces to promote creativity around STEM in schools
Acronym: STEMJAM
Project no. 2016-1-ES01-KA201-025470

www.stemjam.eu

Co-funded by the
Erasmus+ Programme
of the European Union

THE CLEANEST DUSTMAN

ABSTRACT

The idea is to promote the cleaning in the classrooms, through a game.

It would be a competition between 2 mBots. The contest consists of collecting garbage from the ground of the classroom.

The mBot that collects the most, wins.

The fact of winning invites us to reflect on the large number of elements that the students throw to the ground and not to the appropriate containers. It also invites to modify those incorrect habits.

DIDACTIC OBJECTIVES

ENGINEERING and SCIENCE:

- Concepts of mechanics: degrees of freedom, choice of movements to be made, choice of mechanical components necessary to carry out those movements, rotation speed, reduction of speed, engines to carry out this reduction, etc.
- Design of structures:
  - Stability: concepts of inertia and center of gravity.
  - Choice of the most suitable structure for the assembly of the chosen components and sensors.
  - Assembly of the structure.
- Reference systems: positioning in a reference system.

TECHNOLOGY:

- Introduction to computational thinking.
- Assimilation, creation and programming of algorithms, to decompose complex problems into ordered sequences of simple instructions, which solve it.

STEM Subject: Science☒ Technology ☒ Engineering☐ Mathematics☒

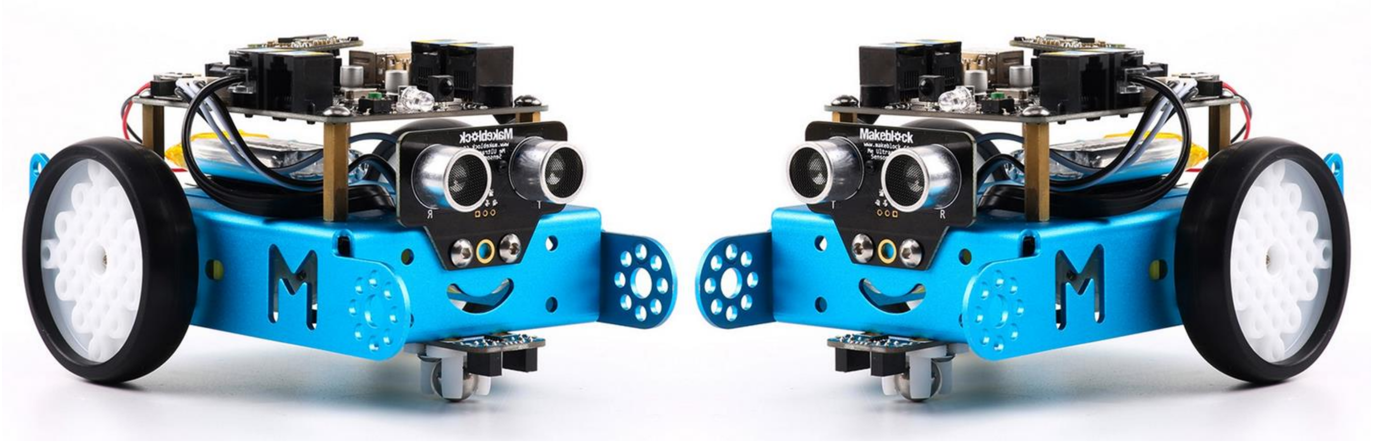
Education Level: 12-14 years☒ 14-16 years☐

PROBLEM STATEMENT

The mBot robot has to be programmed so that it can pick up small bags from the ground. The robot will also have to take the bags to their container. The code must also have a counter part, where the quantity of bags collected is played. All this must be done with a determined time, by means of a chronometer.

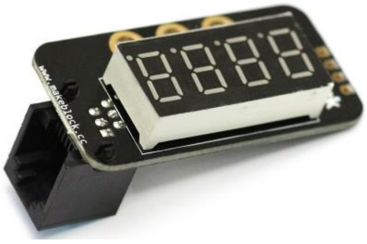
BOM (Bill of Materials Needed)

- (x2) mBot => Ref. 90054

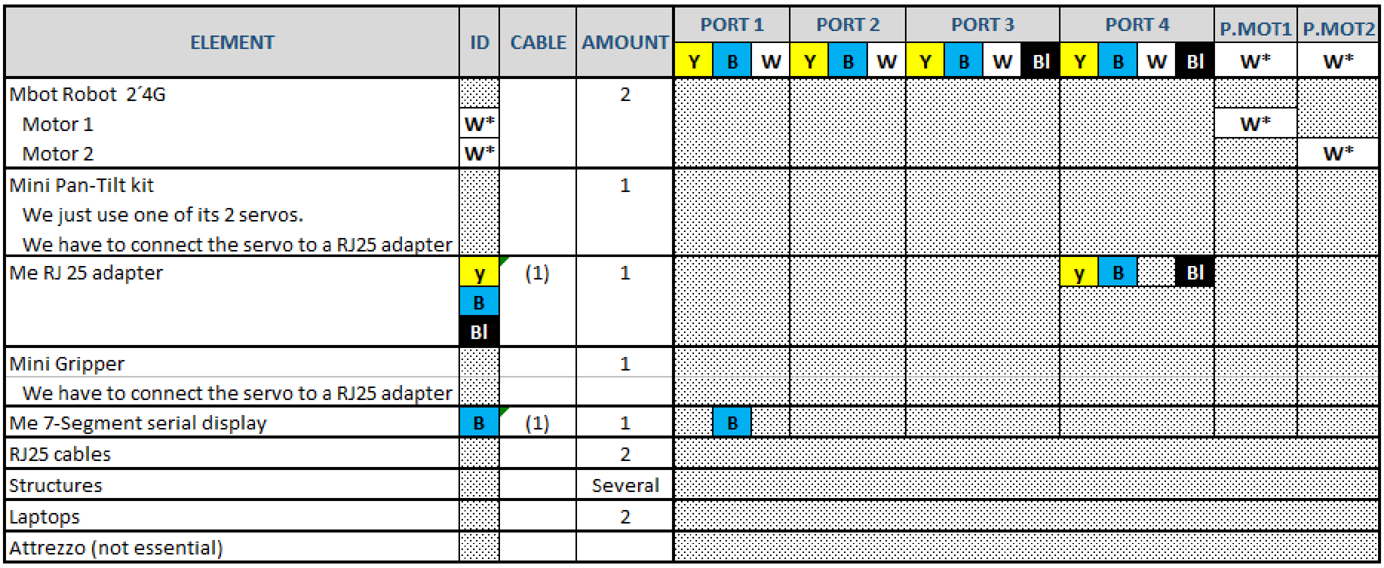


- Different beams and structures:

- Me **7-Segment Serial Display - Red**:



- Mini Pan-Tilt Kit.
- Mini gripper.
- Me adapter RJ25.
  (The Mini Pan-Tilt Kit and the Mini gripper, will be connected to this adapter).
- 2 Cables RJ25.
- 10 Little trash bags.
- 2 Containers for the trash bags.
- Rest of Attrezzo (not essential).



ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT1	P.MOT2
				Y	B	W	Y	B	W	Y	B	W	Bl	Y	B	W	Bl	W*	W*
Mbot Robot 2´4G			2																
Motor 1	W*																	W*	
Motor 2	W*																		W*
Mini Pan-Tilt kit			1																
We just use one of its 2 servos.																			
We have to connect the servo to a RJ25 adapter																			
Me RJ 25 adapter	y	(1)	1											y	B		Bl		
	B																		
	Bl																		
Mini Gripper			1																
We have to connect the servo to a RJ25 adapter																			
Me 7-Segment serial display	B	(1)	1		B														
RJ25 cables			2																
Structures			Several																
Laptops			2																
Attrezzo (not essential)																			

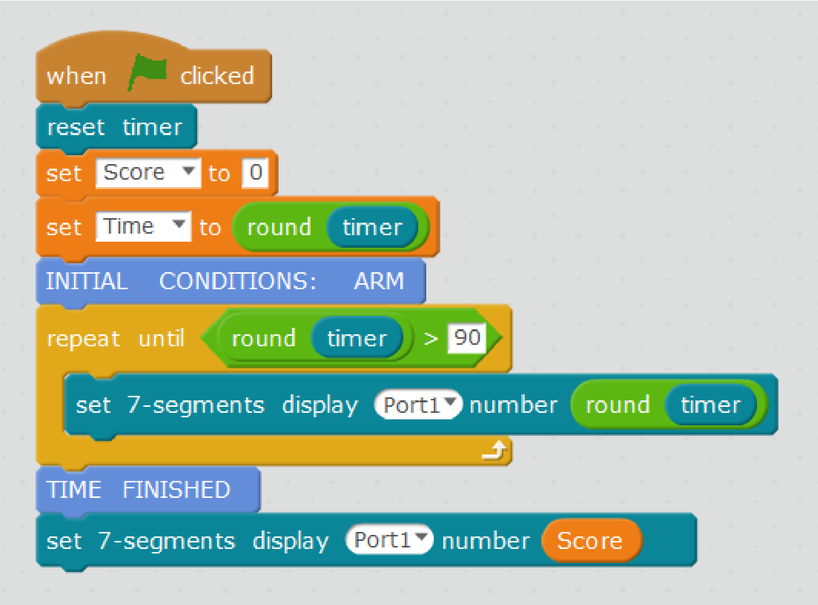
ACTIVITY DESCRIPTION

This activity consists of collecting the largest possible number of trash bags, in a certain time.

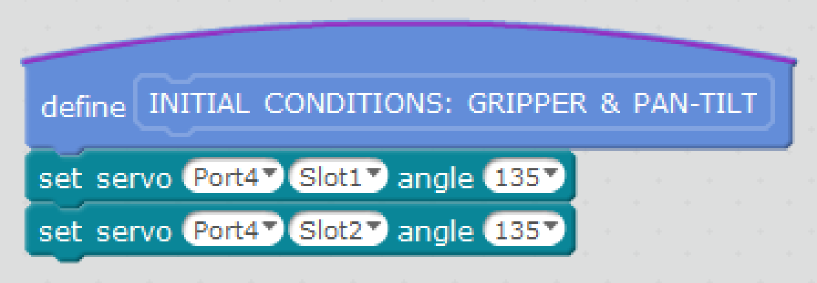
To achieve this goal, students will have to design everything related to the programming of movement (both, the mBot and the gripper & pan-tilt movements). In addition, they will have to program different sound and dance effects for the activity.

After all these technical tasks, we start with the PROGRAMMING.

1. INTRODUCTION PART OF THE PROGRAM:



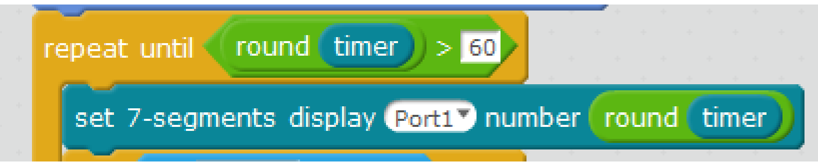
```
when ⚑ clicked
reset timer
set Score to 0
set Time to round timer
INITIAL CONDITIONS: ARM
repeat until round timer > 90
    set 7-segments display Port1 number round timer
TIME FINISHED
set 7-segments display Port1 number Score
```



```
define INITIAL CONDITIONS: GRIPPER & PAN-TILT
set servo Port4 Slot1 angle 135
set servo Port4 Slot2 angle 135
```

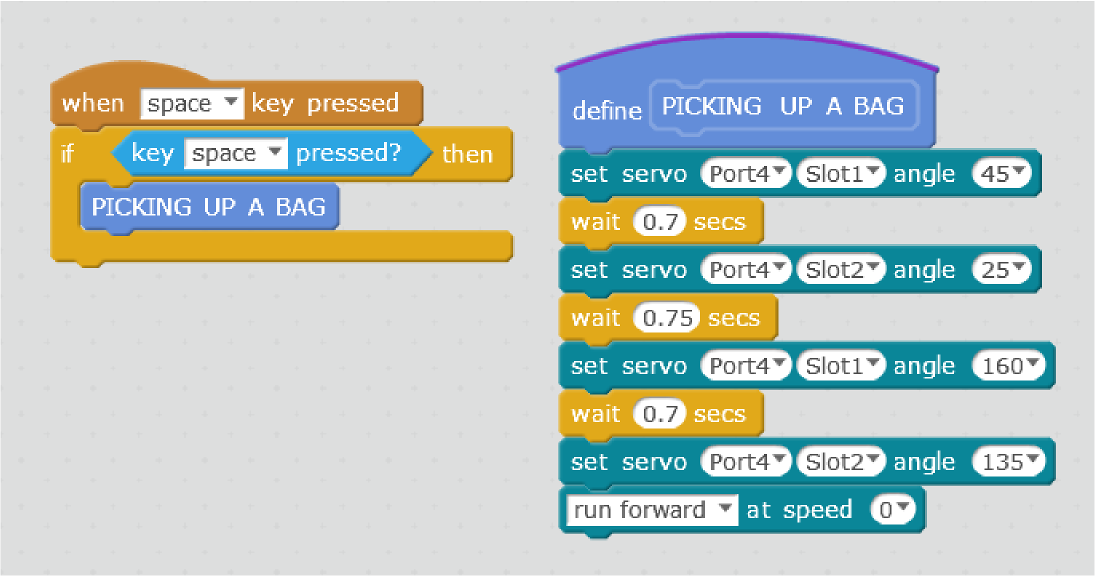
Right at the beginning, we will reset the timer, reset the score and place the “ARM” to the right place.

The time left will be shown in the display until time is finished.



```
repeat until round timer > 60
    set 7-segments display Port1 number round timer
```

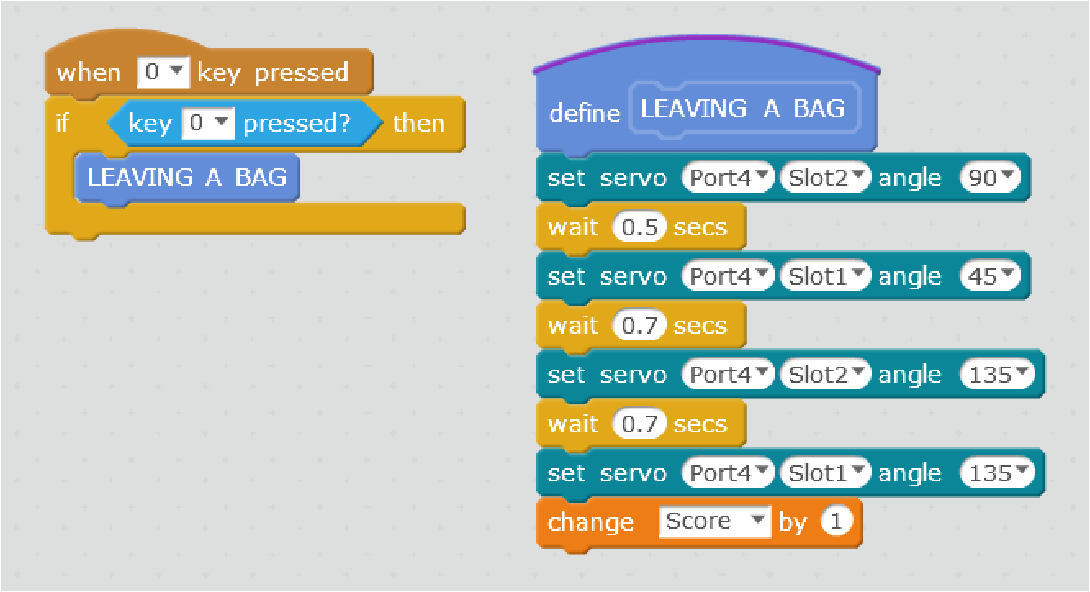
2. PICKING UP A BAG Routine. (Key “space”):



```
when space key pressed
if key space pressed? then
    PICKING UP A BAG
```

```
define PICKING UP A BAG
set servo Port4 Slot1 angle 45
wait 0.7 secs
set servo Port4 Slot2 angle 25
wait 0.75 secs
set servo Port4 Slot1 angle 160
wait 0.7 secs
set servo Port4 Slot2 angle 135
run forward at speed 0
```

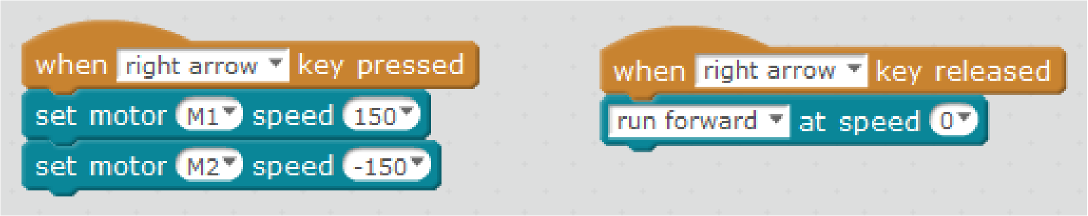
3. LEAVING A BAG Routine. SCORING. (Key “0”):



```
when 0 key pressed
if key 0 pressed? then
    LEAVING A BAG
```

```
define LEAVING A BAG
set servo Port4 Slot2 angle 90
wait 0.5 secs
set servo Port4 Slot1 angle 45
wait 0.7 secs
set servo Port4 Slot2 angle 135
wait 0.7 secs
set servo Port4 Slot1 angle 135
change Score by 1
```

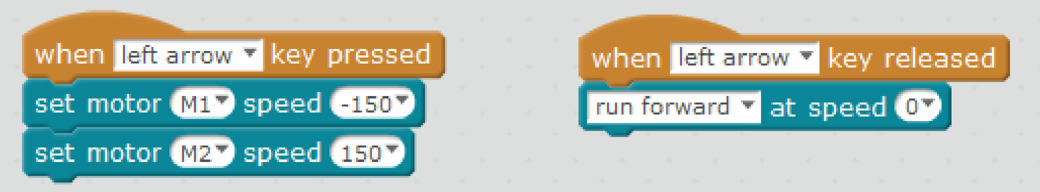
4. MOVING TO THE RIGHT Routine. (Right arrow key):



```
when right arrow key pressed
set motor M1 speed 150
set motor M2 speed -150
```

```
when right arrow key released
run forward at speed 0
```

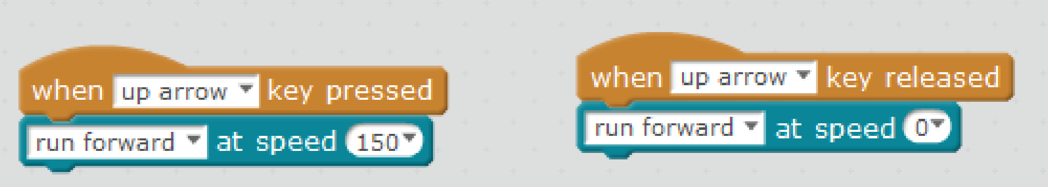
5. MOVING TO THE LEFT Routine. (Left arrow key):



```
when left arrow key pressed
set motor M1 speed -150
set motor M2 speed 150
```

```
when left arrow key released
run forward at speed 0
```

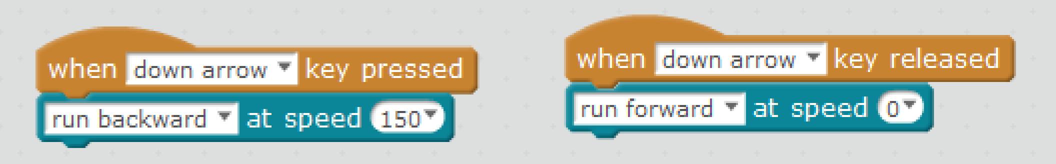
6. MOVING FORWARD Routine. (Up arrow key):



```
when up arrow key pressed
run forward at speed 150
```

```
when up arrow key released
run forward at speed 0
```

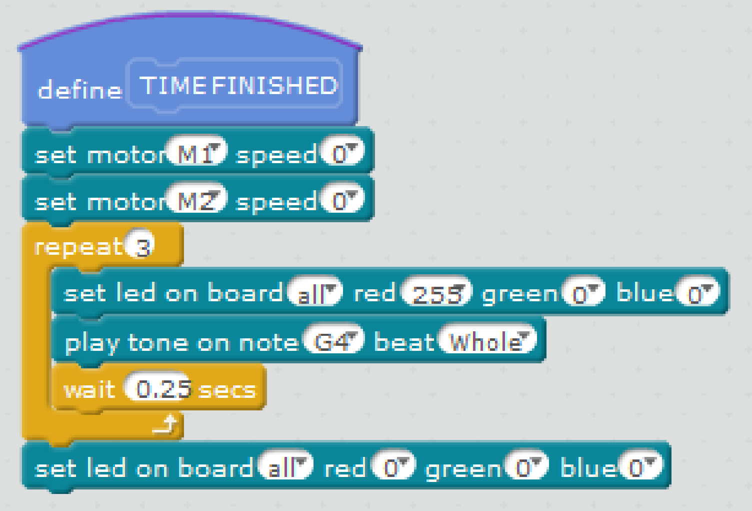
7. MOVING BACKWARDS Routine. (Down arrow key):



```
when down arrow key pressed
run backward at speed 150
```

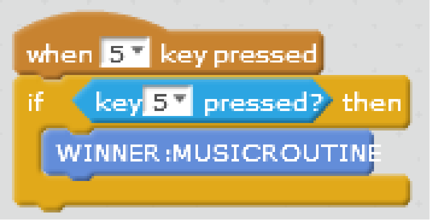
```
when down arrow key released
run forward at speed 0
```

8. TIME FINISHED ROUTINE:

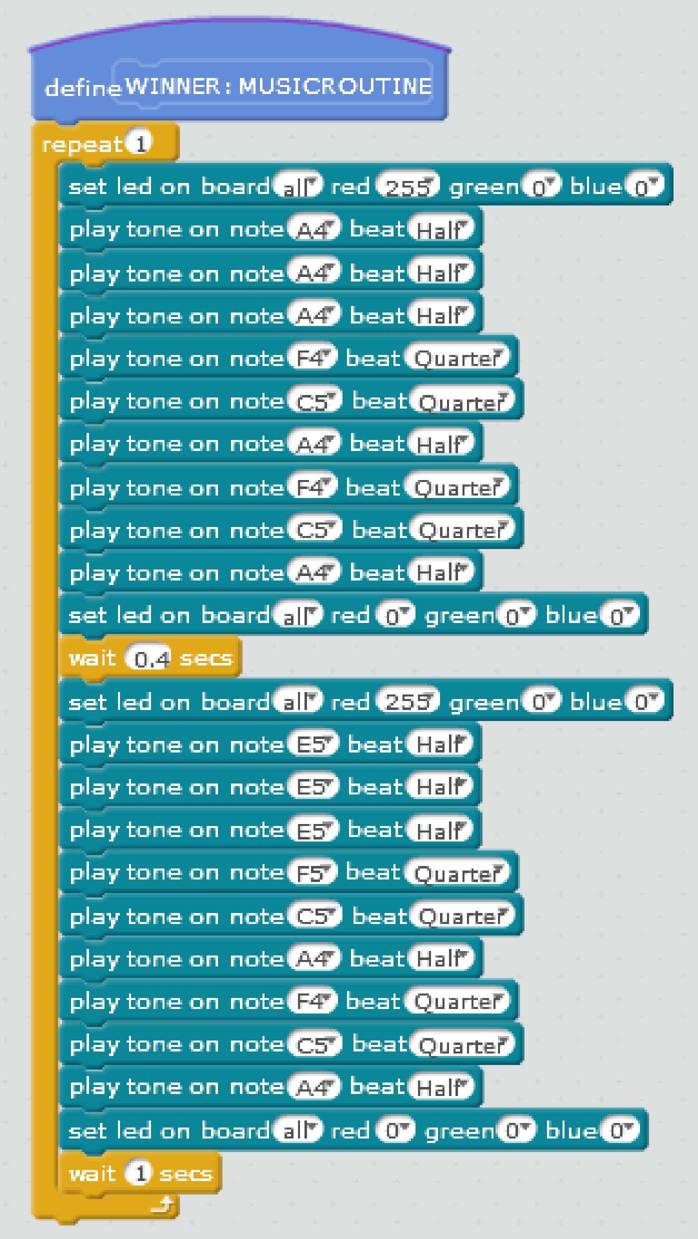


```
define TIME FINISHED
set motor M1 speed 0
set motor M2 speed 0
repeat 3
    set led on board all red 255 green 0 blue 0
    play tone on note G4 beat Whole
    wait 0.25 secs
set led on board all red 0 green 0 blue 0
```

9. WIINER: MUSIC Routine. (Key "5"). (Just for the winner):

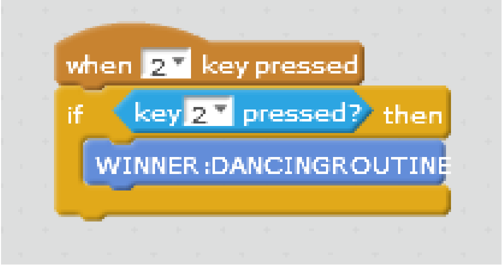


```
when 5 key pressed
if key 5 pressed? then
    WINNER :MUSICROUTINE
```

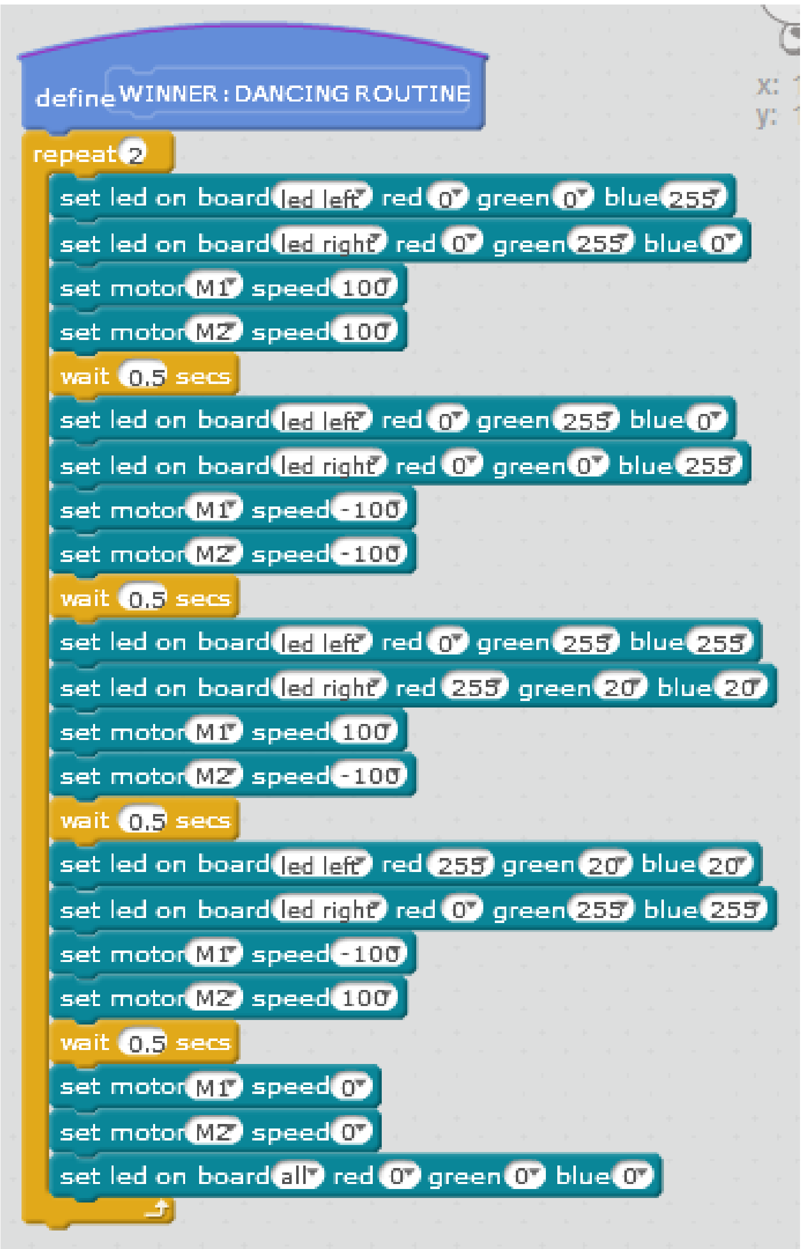


```
define WINNER : MUSICROUTINE
repeat 1
    set led on board all red 255 green 0 blue 0
    play tone on note A4 beat Half
    play tone on note A4 beat Half
    play tone on note A4 beat Half
    play tone on note F4 beat Quarter
    play tone on note C5 beat Quarter
    play tone on note A4 beat Half
    play tone on note F4 beat Quarter
    play tone on note C5 beat Quarter
    play tone on note A4 beat Half
    set led on board all red 0 green 0 blue 0
    wait 0.4 secs
    set led on board all red 255 green 0 blue 0
    play tone on note E5 beat Half
    play tone on note E5 beat Half
    play tone on note E5 beat Half
    play tone on note F5 beat Quarter
    play tone on note C5 beat Quarter
    play tone on note A4 beat Half
    play tone on note F4 beat Quarter
    play tone on note C5 beat Quarter
    play tone on note A4 beat Half
    set led on board all red 0 green 0 blue 0
    wait 1 secs
```

10. WINNER: DANCING Routine. (Key “2”). (Just for the winner):

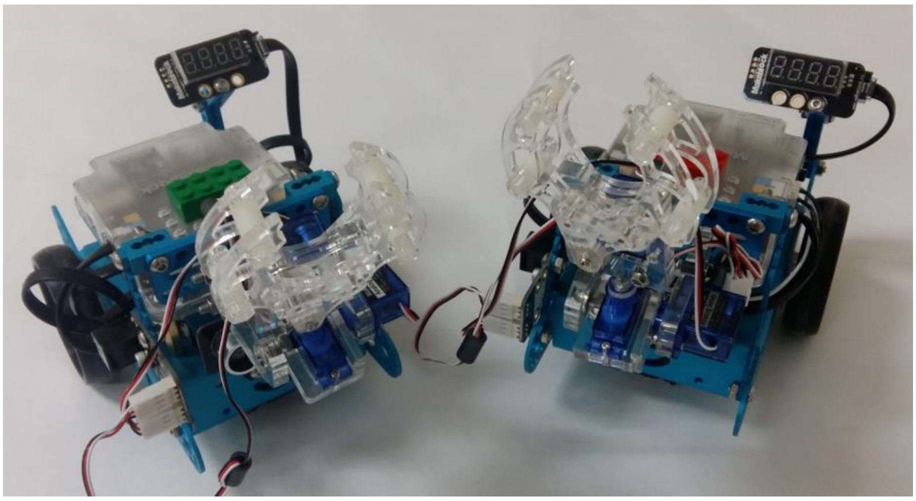


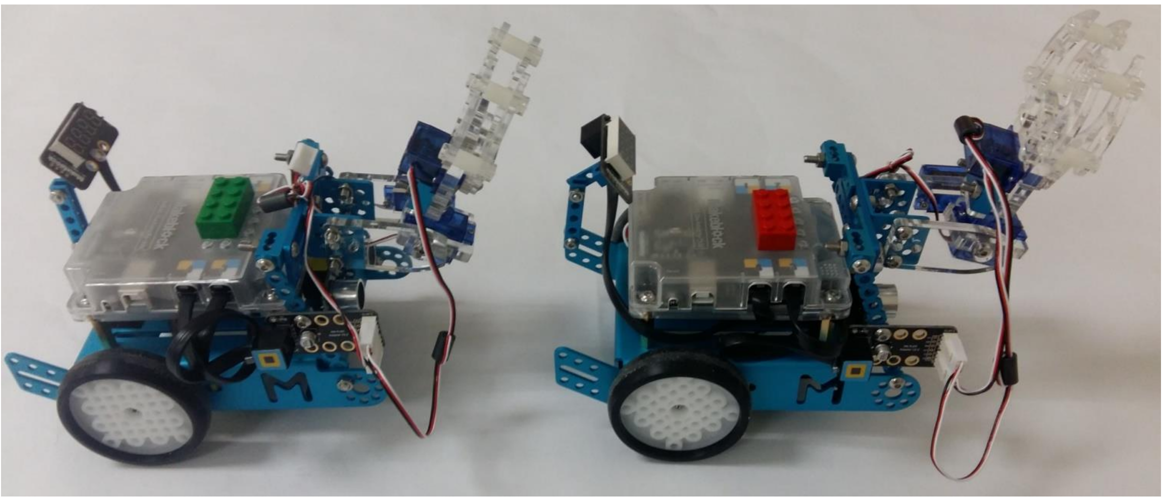
```
when 2 key pressed
if key 2 pressed? then
    WINNER :DANCINGROUTINE
```

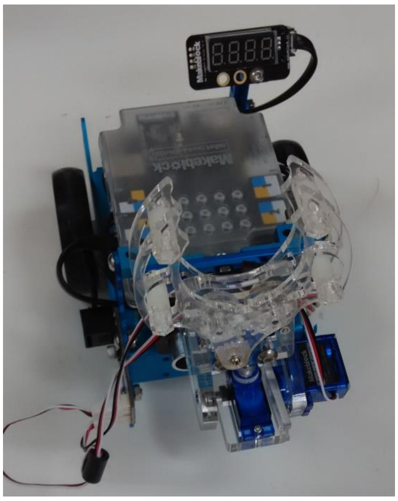


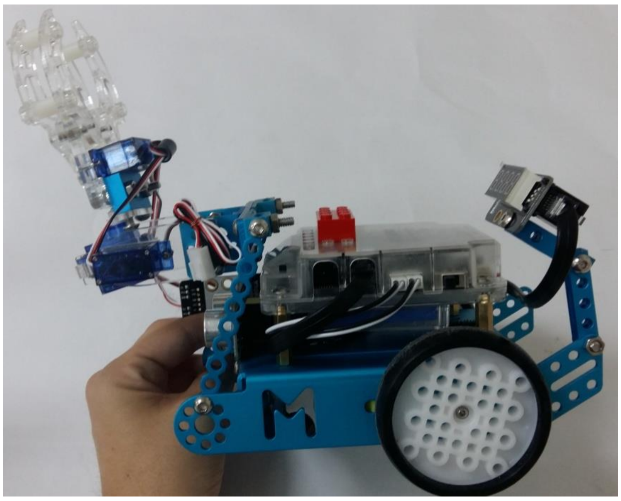
```
define WINNER : DANCING ROUTINE
repeat 2
    set led on board led left red 0 green 0 blue 255
    set led on board led right red 0 green 255 blue 0
    set motor M1 speed 100
    set motor M2 speed 100
    wait 0.5 secs
    set led on board led left red 0 green 255 blue 0
    set led on board led right red 0 green 0 blue 255
    set motor M1 speed -100
    set motor M2 speed -100
    wait 0.5 secs
    set led on board led left red 0 green 255 blue 255
    set led on board led right red 255 green 20 blue 20
    set motor M1 speed 100
    set motor M2 speed -100
    wait 0.5 secs
    set led on board led left red 255 green 20 blue 20
    set led on board led right red 0 green 255 blue 255
    set motor M1 speed -100
    set motor M2 speed 100
    wait 0.5 secs
    set motor M1 speed 0
    set motor M2 speed 0
    set led on board all red 0 green 0 blue 0
```

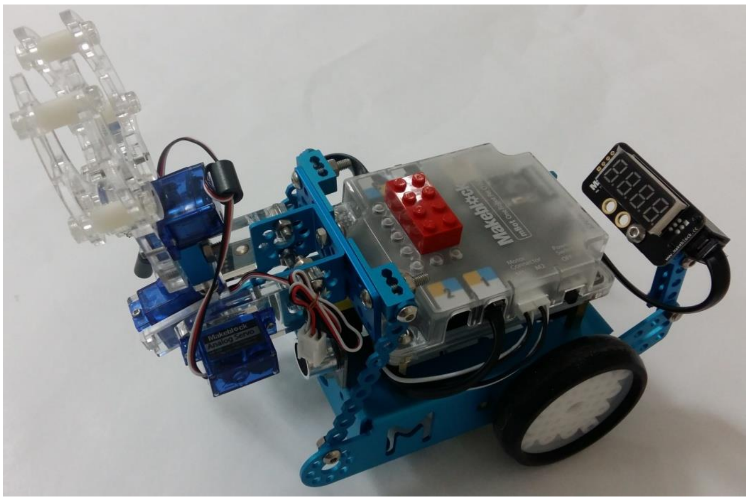
Once, the programming is finished, we start building up <u>THE STRUCTURE</u>, where all the mechanical elements will be set. Also the electronic elements.

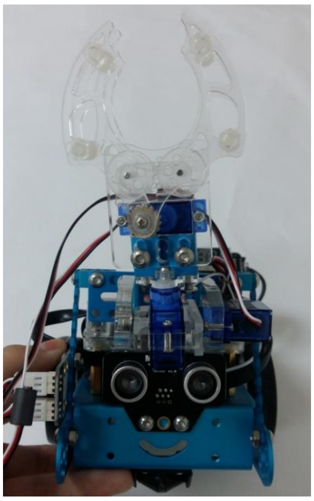


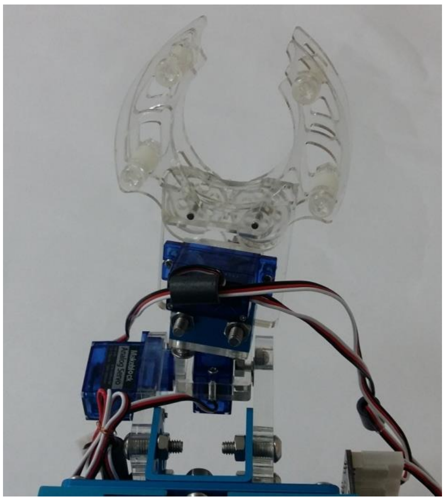


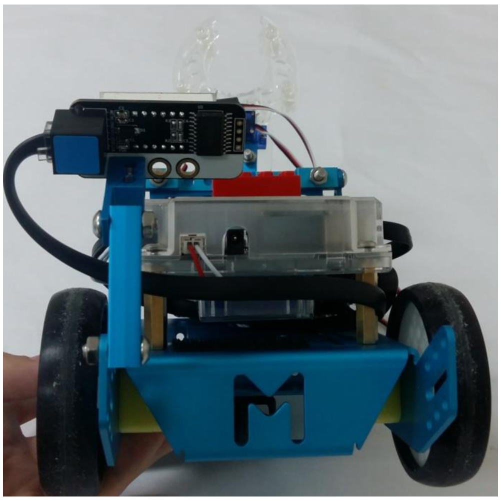


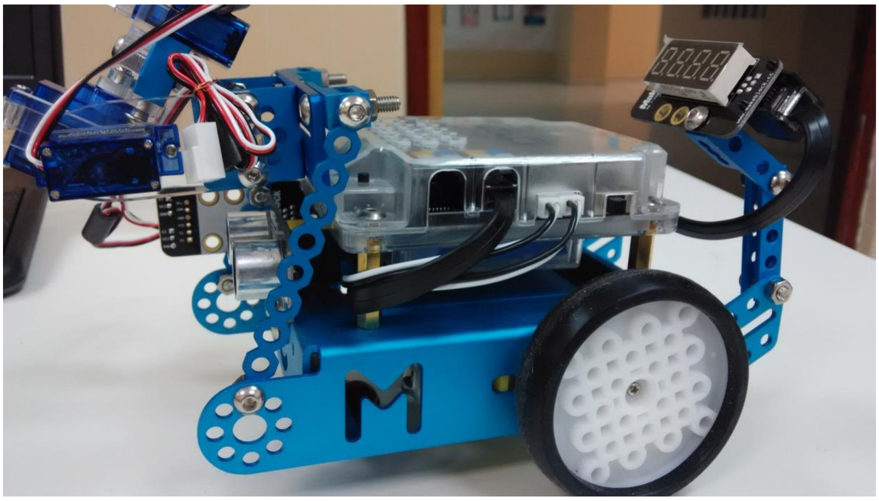


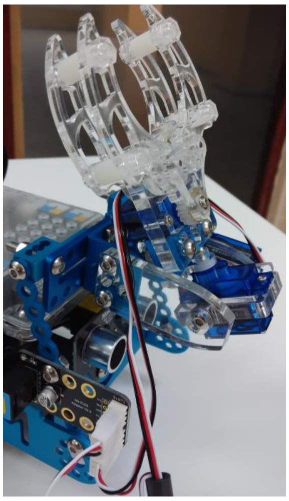


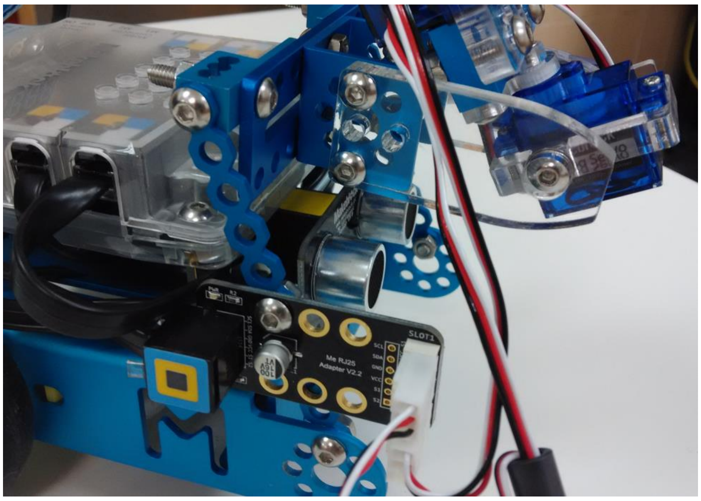


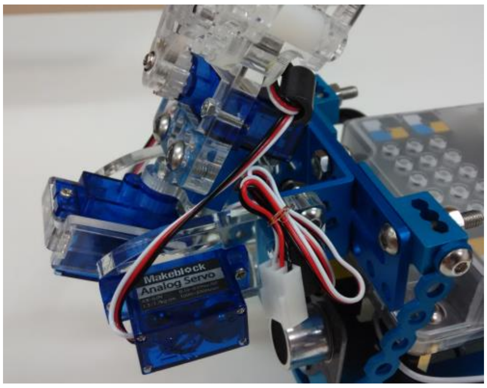


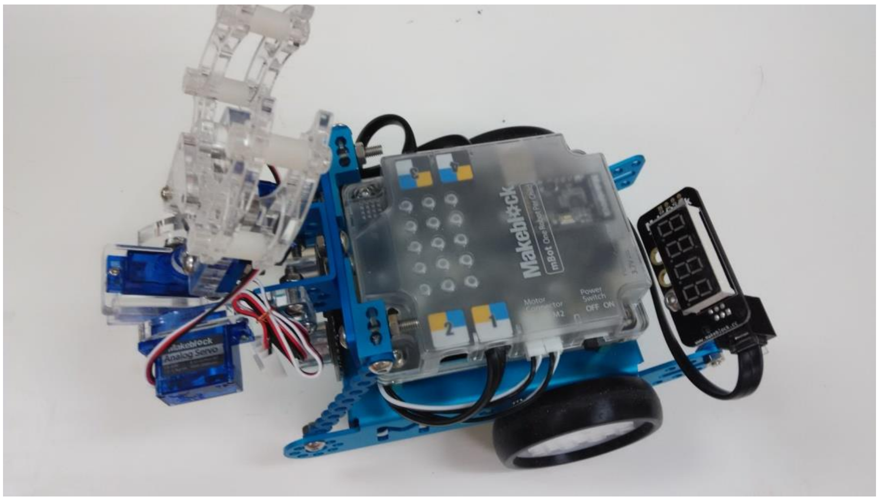




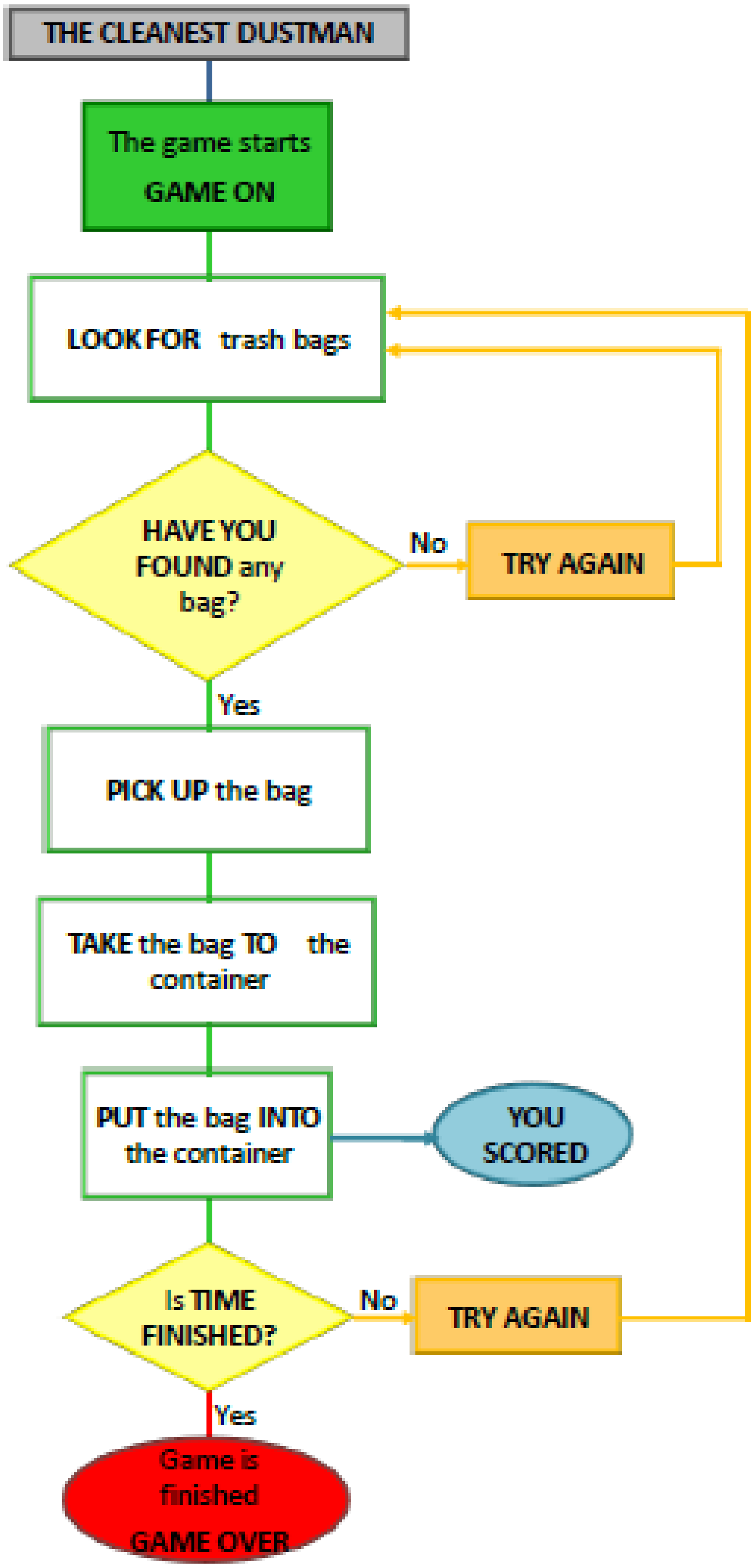



Me RJ25 Adapter V2.2
SLOT1


Makeblock
Analog Servo


Makeblock
mBot

FLOW CHART


THE CLEANEST DUSTMAN
The game starts
GAME ON
LOOK FOR trash bags
HAVE YOU FOUND any bag?
No
TRY AGAIN
Yes
PICK UP the bag
TAKE the bag TO the container
PUT the bag INTO the container
YOU SCORED
Is TIME FINISHED?
No
TRY AGAIN
Yes
Game is finished
GAME OVER

STUDENT'S EVALUATION

For the evaluation of the students in this activity, use the Evaluation Rubric designed for this project.

BIBLIOGRAPHY

"Guía de Scratch". CEIP de Cella, (Teruel). Pdf.

"Scratch. Guía didáctica para profesores". (Pdf). (www.isuriarte.com).

"Jugando con MBlock". Makeblock España.

"Divirtiéndome con MBot". Susana Oubiña.

Comunidad de Makeblock en español. (http://www.makeblock.es/foro/).

"Curso de Scratch + Arduino". J. Javier Esquiva Mira.

MORE INFORMATION

DIFFICULTIES:

- GRIPPER & PAN-TILT: when the servo motors look for a certain angle, sometimes they do not stop at all. This can hinder the operation of some sensors such as the ultrasonic sensor.

- GRIPPER & PAN-TILT: it´s very important to follow properly the instructions in order to build them up in the right way and direction.

- GRIPPER & PAN-TILT: do not force the servo motors with your hands.

- ULTRASONIC SENSOR: it can be used for avoiding obstacles, when they are right in front of the MBot. If the obstacle is in a certain angle with respect to the MBot, the ultrasonic does not work, and the robot can not avoid it.