27

PETANQUE

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

PETANQUE

ABSTRACT

The activity is about to program the mBots so they can play petanque among themselves.

This activity will be a competition between two or more teams. Each team will control the robot with the remote control.

The ball will be thrown without exceeding a throwing line (placed on the floor). The direction and speed of the ball can be chosen.

The idea is that the ball must be left as close as possible to the small ball. The big ball may be also used to push away the balls from the opposite team. The team which brings its balls the closest to the small ball, wins.

In order to let each robot to carry its point count, other keys will be programmed.

The code will be recorded on the Arduino board, so that the development of the game will not depend on the use of a laptop.

DIDACTIC OBJECTIVES

- Learning to program the remote control.
- Learning to program the Arduino board.
- To have the first contact with the Arduino programming language.
- Learning how to use a 7 segments display, the led lights and buzzer.

STEM Subject: Science☐ Technology ☒ Engineering☒ Mathematics☐

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

PROBLEM STATEMENT

We will program different mBot remote control keys so that the direction and speed of the mBot can be graduated with them. It will act as a "launcher" of the balls.

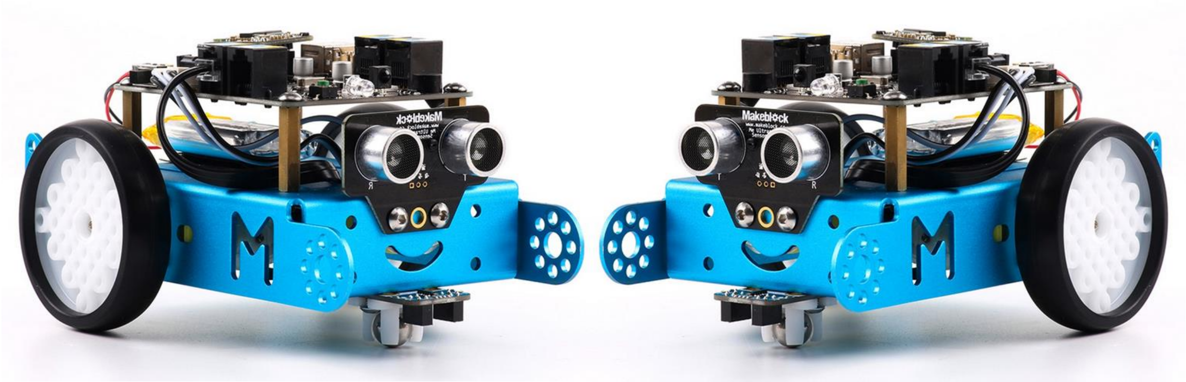
The programs will be recorded on the Arduino board of mBot.

There are several method to design mechanism of pushing the ball. We will show you two of them:

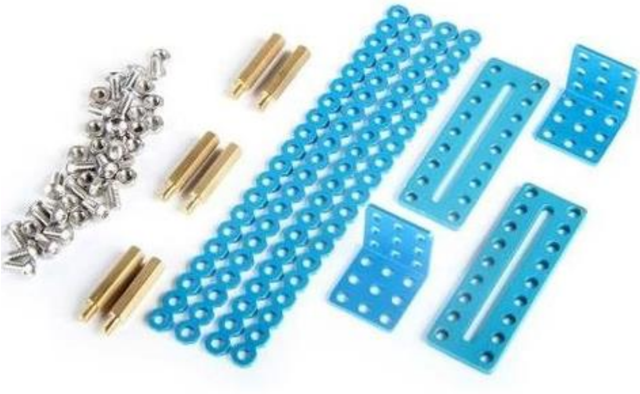
1. You control the speed of turning the mBot.
2. You control the movement of servomotor.

BOM (Bill of Materials Needed)

- (x2) mBots => Ref. 90054

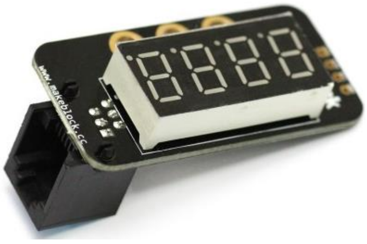


- Different beams and structures:



- (x2) **IR Controller**:

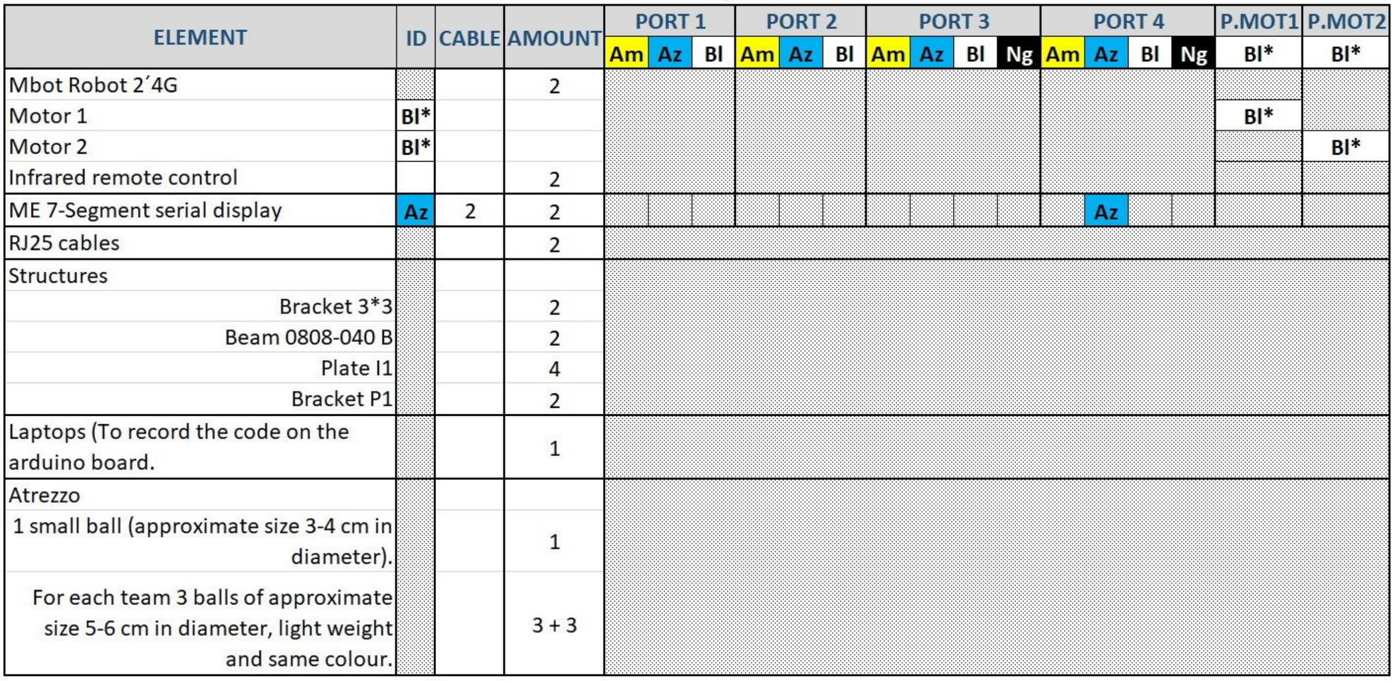
- (x2) Me **7-Segment Serial Display - Red**:



- For each team 3 balls of approximate size 5-6 cm in diameter, light weight and same colour.

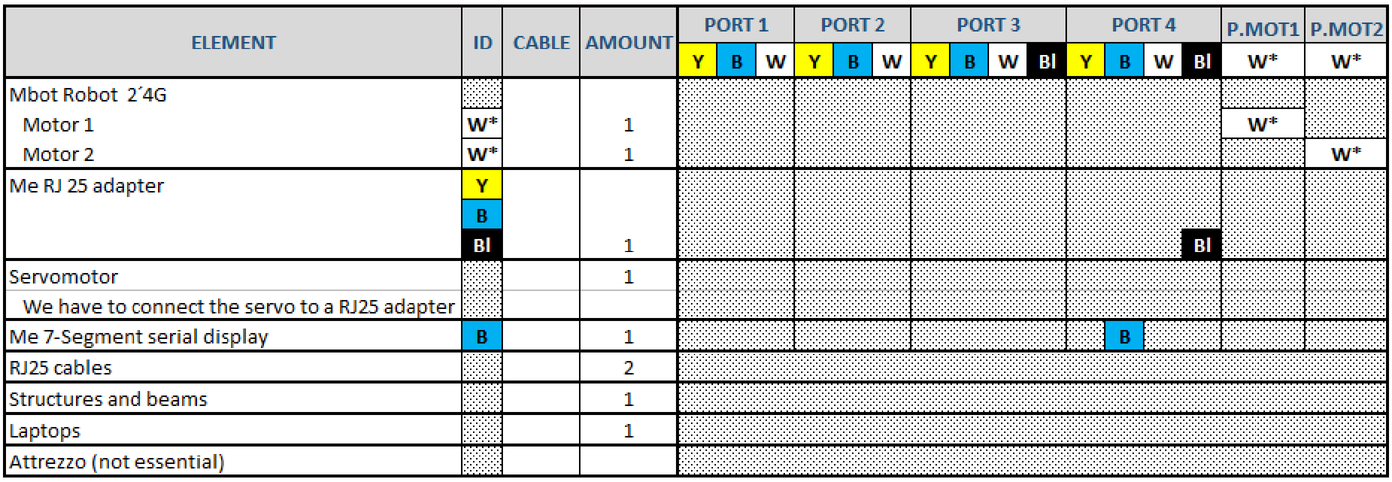
- 1 small ball (approximate size 3-4 cm in diameter).

First version



ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT1	P.MOT2
				Am	Az	Bl	Am	Az	Bl	Am	Az	Bl	Ng	Am	Az	Bl	Ng	Bl*	Bl*
Mbot Robot 2´4G			2																
Motor 1	Bl*																	Bl*	
Motor 2	Bl*																		Bl*
Infrared remote control			2																
ME 7-Segment serial display	Az	2	2												Az				
RJ25 cables			2																
Structures																			
Bracket 3*3			2																
Beam 0808-040 B			2																
Plate I1			4																
Bracket P1			2																
Laptops (To record the code on the arduino board.			1																
Atrezzo																			
1 small ball (approximate size 3-4 cm in diameter).			1																
For each team 3 balls of approximate size 5-6 cm in diameter, light weight and same colour.			3 + 3																

Second version



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																			
Motor 1	W*		1															W*	
Motor 2	W*		1																W*
Me RJ 25 adapter	Y B Bl		1														Bl		
Servomotor			1																
We have to connect the servo to a RJ25 adapter																			
Me 7-Segment serial display	B		1												B				
RJ25 cables			2																
Structures and beams			1																
Laptops			1																
Attrezzo (not essential)																			

ACTIVITY DESCRIPTION

First version

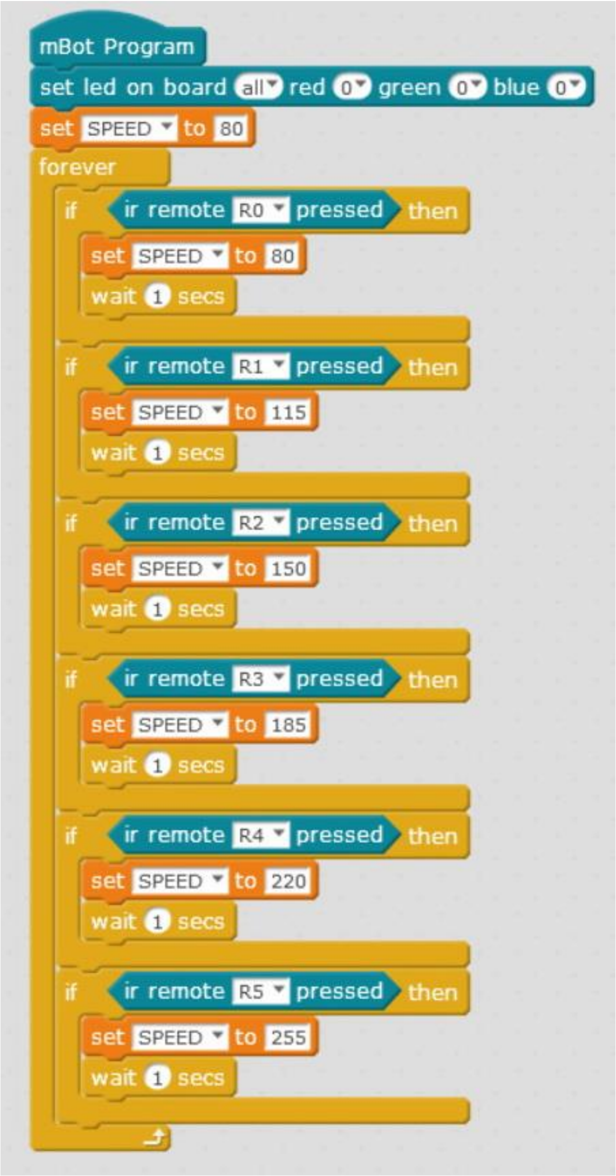
The work of the students consists of learning to program the remote control of the robot and record the programs that are developed on the Arduino board of the robot. To be able to play the game petanque with the mBot, they will have to plan which keys they want to program and with what function (movement, speed, counter, light and sound effects).

They will develop the flowchart and the code, with the help of the teacher.

The code is short and simple.

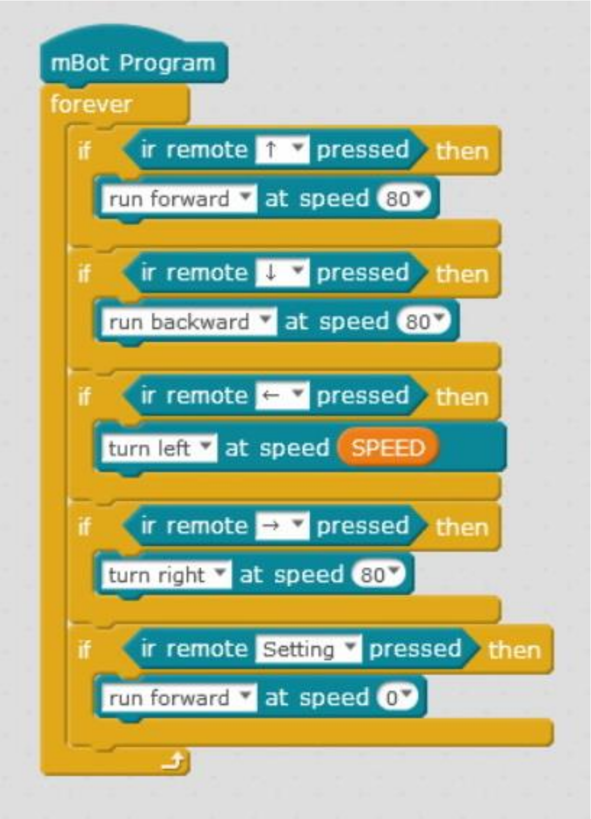
For better understanding by the students, the code has been subdivided into 4 subprograms.

ROUTINE 1: ALLOCATION OF FUNCTION TO THE KEYS OF THE REMOTE CONTROL



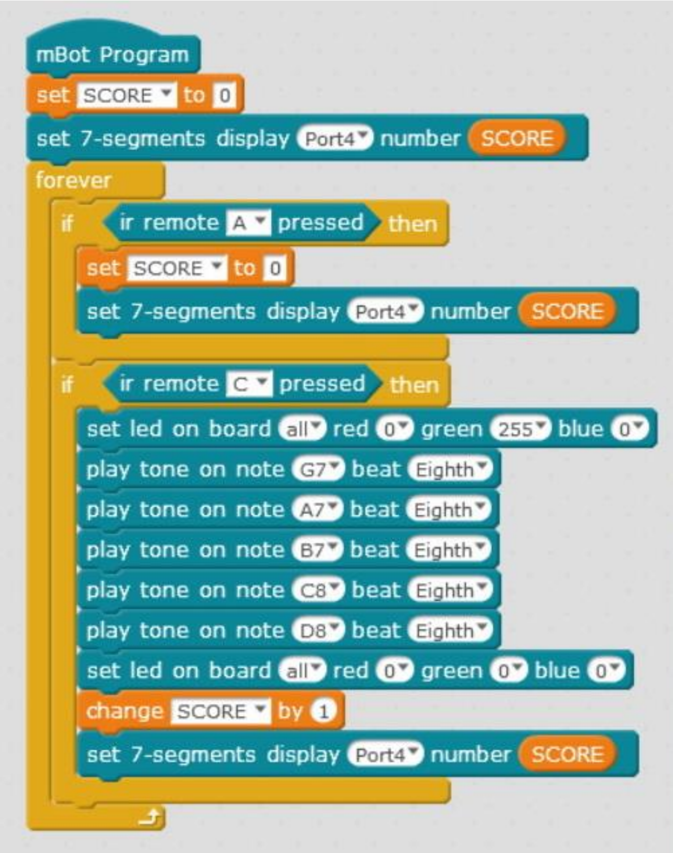
```
mBot Program
set led on board all red 0 green 0 blue 0
set SPEED to 80
forever
    if ir remote R0 pressed then
        set SPEED to 80
        wait 1 secs
    if ir remote R1 pressed then
        set SPEED to 115
        wait 1 secs
    if ir remote R2 pressed then
        set SPEED to 150
        wait 1 secs
    if ir remote R3 pressed then
        set SPEED to 185
        wait 1 secs
    if ir remote R4 pressed then
        set SPEED to 220
        wait 1 secs
    if ir remote R5 pressed then
        set SPEED to 255
        wait 1 secs
```

ROUTINE 2: DEFINITION OF MOVEMENT



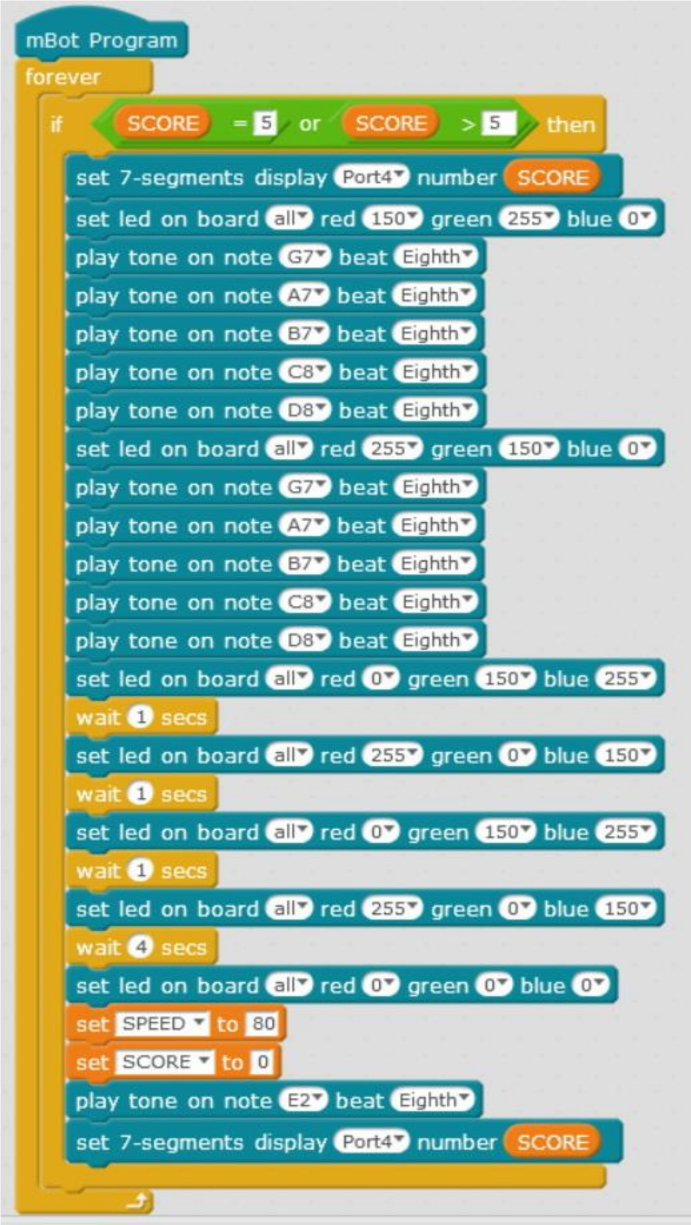
```
mBot Program
forever
    if  ir remote ↑ pressed  then
        run forward at speed 80
    if  ir remote ↓ pressed  then
        run backward at speed 80
    if  ir remote ← pressed  then
        turn left at speed SPEED
    if  ir remote → pressed  then
        turn right at speed 80
    if  ir remote Setting pressed  then
        run forward at speed 0
```

ROUTINE 3: SCORING AND EFFECTS



```
mBot Program
set SCORE to 0
set 7-segments display Port4 number SCORE
forever
    if  ir remote A pressed  then
        set SCORE to 0
        set 7-segments display Port4 number SCORE
    if  ir remote C pressed  then
        set led on board all red 0 green 255 blue 0
        play tone on note G7 beat Eighth
        play tone on note A7 beat Eighth
        play tone on note B7 beat Eighth
        play tone on note C8 beat Eighth
        play tone on note D8 beat Eighth
        set led on board all red 0 green 0 blue 0
        change SCORE by 1
        set 7-segments display Port4 number SCORE
```

ROUTINE 4: END OF THE MATCH, EFFECTS AND RESET OF COUNTERS AND PROGRAM



```
mBot Program
forever
  if < SCORE = 5 > or < SCORE > 5 > then
    set 7-segments display Port4 number SCORE
    set led on board all red 150 green 255 blue 0
    play tone on note G7 beat Eighth
    play tone on note A7 beat Eighth
    play tone on note B7 beat Eighth
    play tone on note C8 beat Eighth
    play tone on note D8 beat Eighth
    set led on board all red 255 green 150 blue 0
    play tone on note G7 beat Eighth
    play tone on note A7 beat Eighth
    play tone on note B7 beat Eighth
    play tone on note C8 beat Eighth
    play tone on note D8 beat Eighth
    set led on board all red 0 green 150 blue 255
    wait 1 secs
    set led on board all red 255 green 0 blue 150
    wait 1 secs
    set led on board all red 0 green 150 blue 255
    wait 1 secs
    set led on board all red 255 green 0 blue 150
    wait 4 secs
    set led on board all red 0 green 0 blue 0
    set SPEED to 80
    set SCORE to 0
    play tone on note E2 beat Eighth
    set 7-segments display Port4 number SCORE
```

We will record the code in the arduino board of the mBot. In this way, the mBot will work independently of the computer.

How to load a program on the arduino mBot board using mBlock:

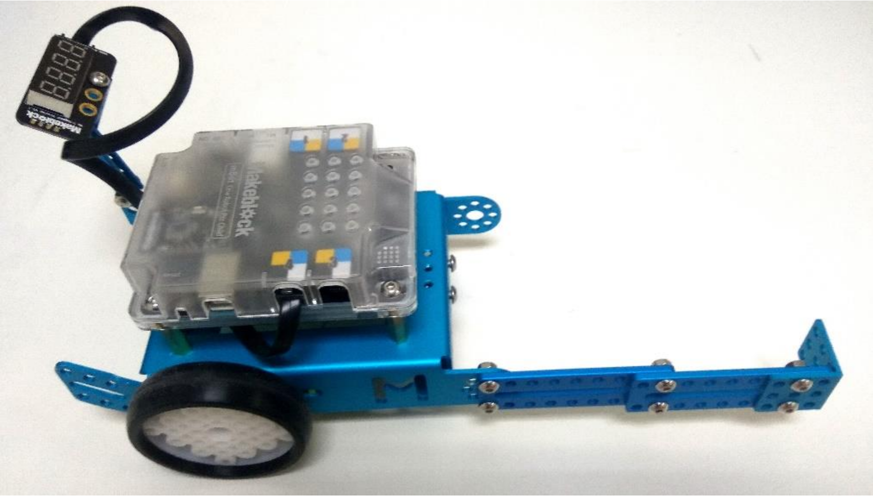
In order to load a program on the board using mBlock:

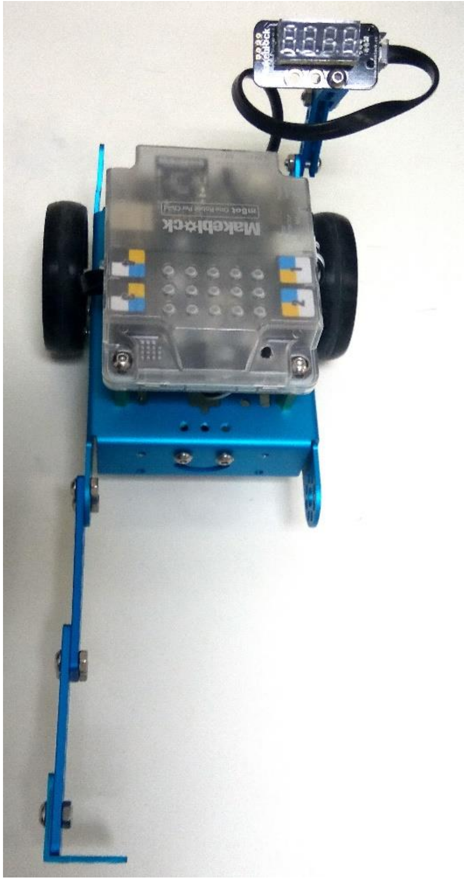
1. Choose mBot in the Board tab of the mBlock menu.
2. Connect the USB and choose "Serial Port" in the connect tab.
3. In the tab edit, choose "Arduino Mode" (In the program that we are going to load, instead of the green flag, we will put the blue command "mBot program")

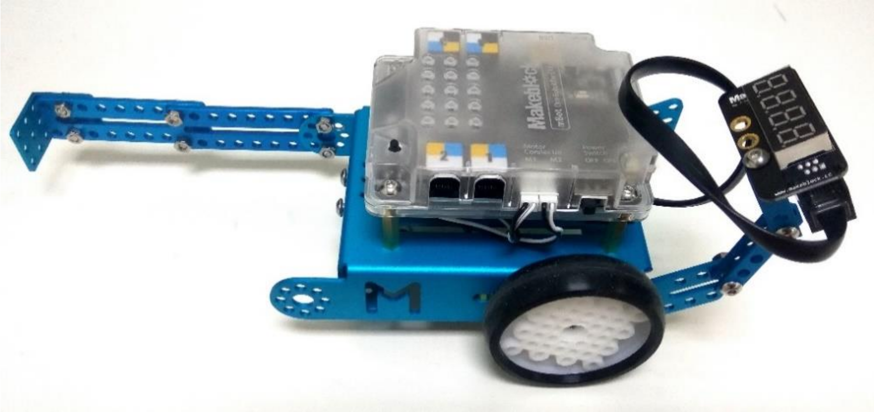
4. A window with the code will open to record it on the Arduino board of mBot. You can, if you want, modify your program. Finally, click on Upload to Arduino.

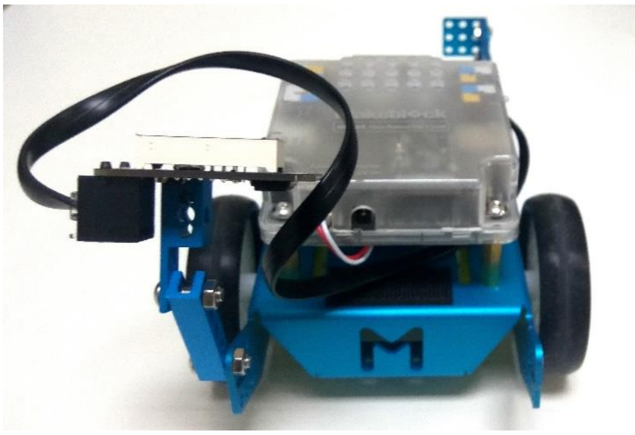
5. If there have been no errors, a message will be sent informing that the program has been recorded correctly. At this moment, you will be able to start enjoying the program introduced in the robot, without the computer turned on. For doing this, you must disconnect the USB cable and connect the batteries (or lithium battery) of the robot. You will see that your mBot works independently.

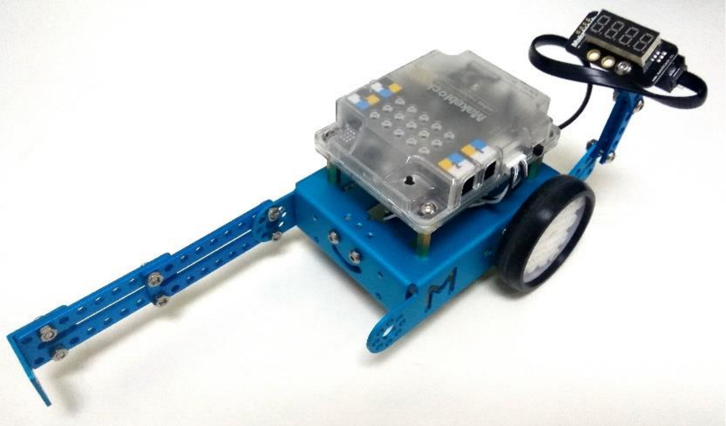
Structural composition:





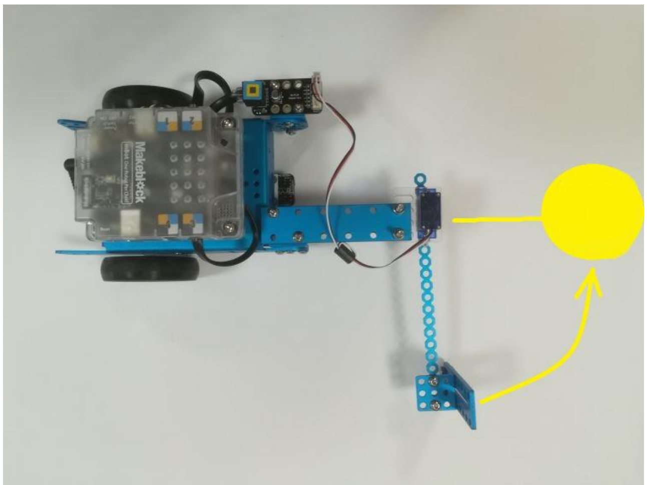




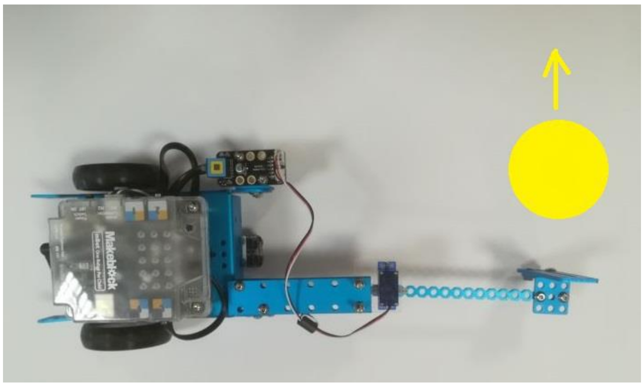


Second version

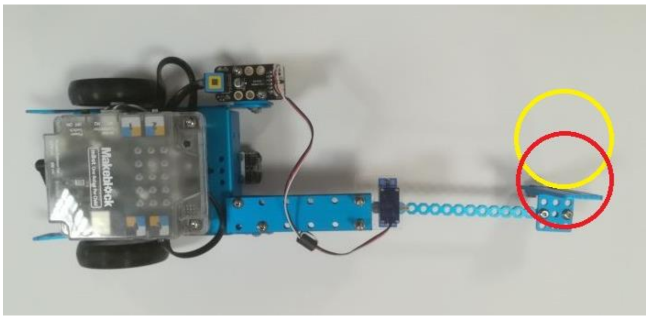
The second option of pushing the ball is to use servomotor. We can use remote control to run with mBot (arrows) and move servomotor.



Set mBot in front of the ball. The servomotor is ready to push the ball



When you set servomotor to position 2 the ball will be hit.

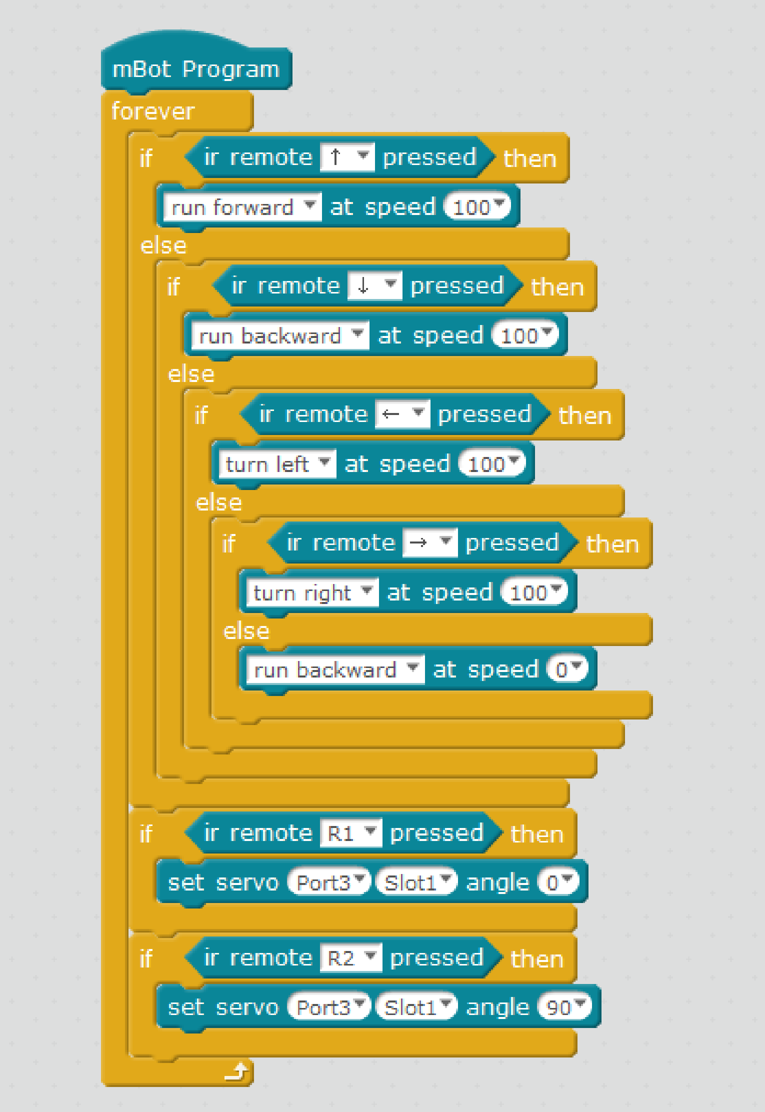


The servomotor pushes the ball with the same force – even if you set the angle to 45 or less. This is the disadvantage of this solution. To control the speed of the ball try to change the position of the mBot.

The yellow ball should have lower speed than the red ball.

This is the basic code, which let you run with mBot and move servomotor. The angle 0 and 90 should be change, because they depend on your construction. You must try values from 0 to 180

Note, that the *if* instruction is nested. Then, when you release the remote control button, the robot will stop.

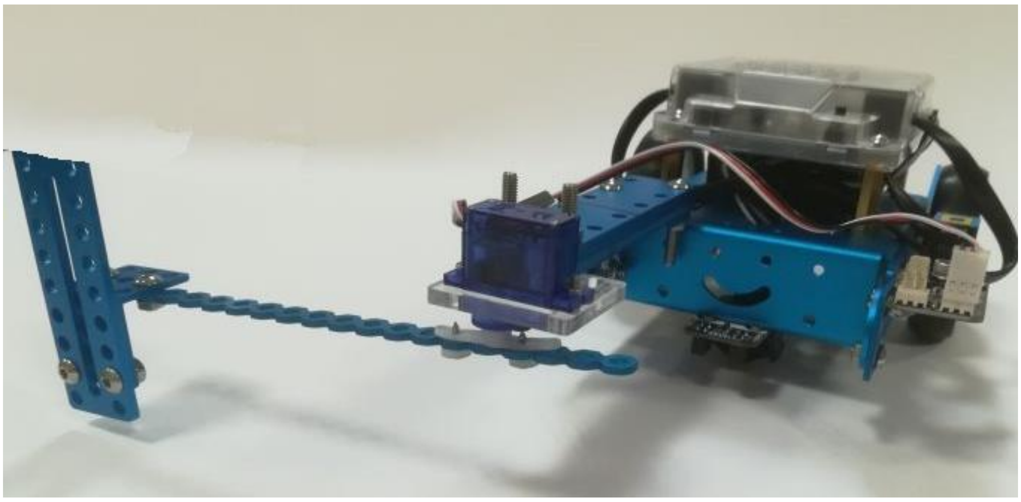


```
mBot Program
forever
    if ir remote ↑ pressed then
        run forward at speed 100
    else
        if ir remote ↓ pressed then
            run backward at speed 100
        else
            if ir remote ← pressed then
                turn left at speed 100
            else
                if ir remote → pressed then
                    turn right at speed 100
                else
                    run backward at speed 0
    if ir remote R1 pressed then
        set servo Port3 Slot1 angle 0
    if ir remote R2 pressed then
        set servo Port3 Slot1 angle 90
```

Of course you can add more instruction to count the point – you have learnt about it in version one. Below you can find flowchart of this part of code. It is very detailed.

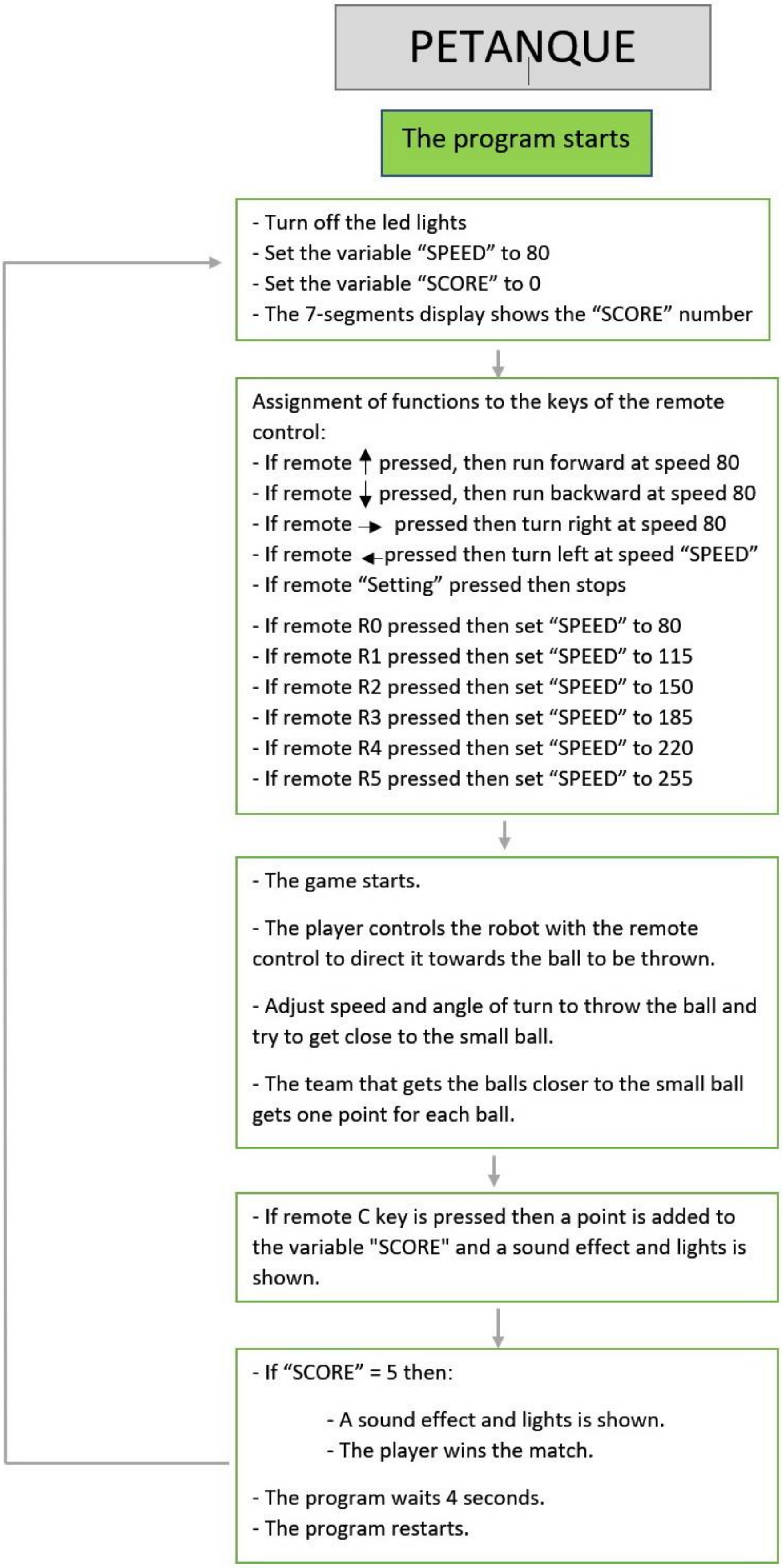
Structural composition:

In the following picture you can see how to mount the servomotor:



FLOW CHART

First version



PETANQUE

The program starts

- Turn off the led lights
- Set the variable “SPEED” to 80
- Set the variable “SCORE” to 0
- The 7-segments display shows the “SCORE” number

Assignment of functions to the keys of the remote control:

- If remote ↑ pressed, then run forward at speed 80
- If remote ↓ pressed, then run backward at speed 80
- If remote → pressed then turn right at speed 80
- If remote ← pressed then turn left at speed “SPEED”
- If remote “Setting” pressed then stops
- If remote R0 pressed then set “SPEED” to 80
- If remote R1 pressed then set “SPEED” to 115
- If remote R2 pressed then set “SPEED” to 150
- If remote R3 pressed then set “SPEED” to 185
- If remote R4 pressed then set “SPEED” to 220
- If remote R5 pressed then set “SPEED” to 255

- The game starts.

- The player controls the robot with the remote control to direct it towards the ball to be thrown.

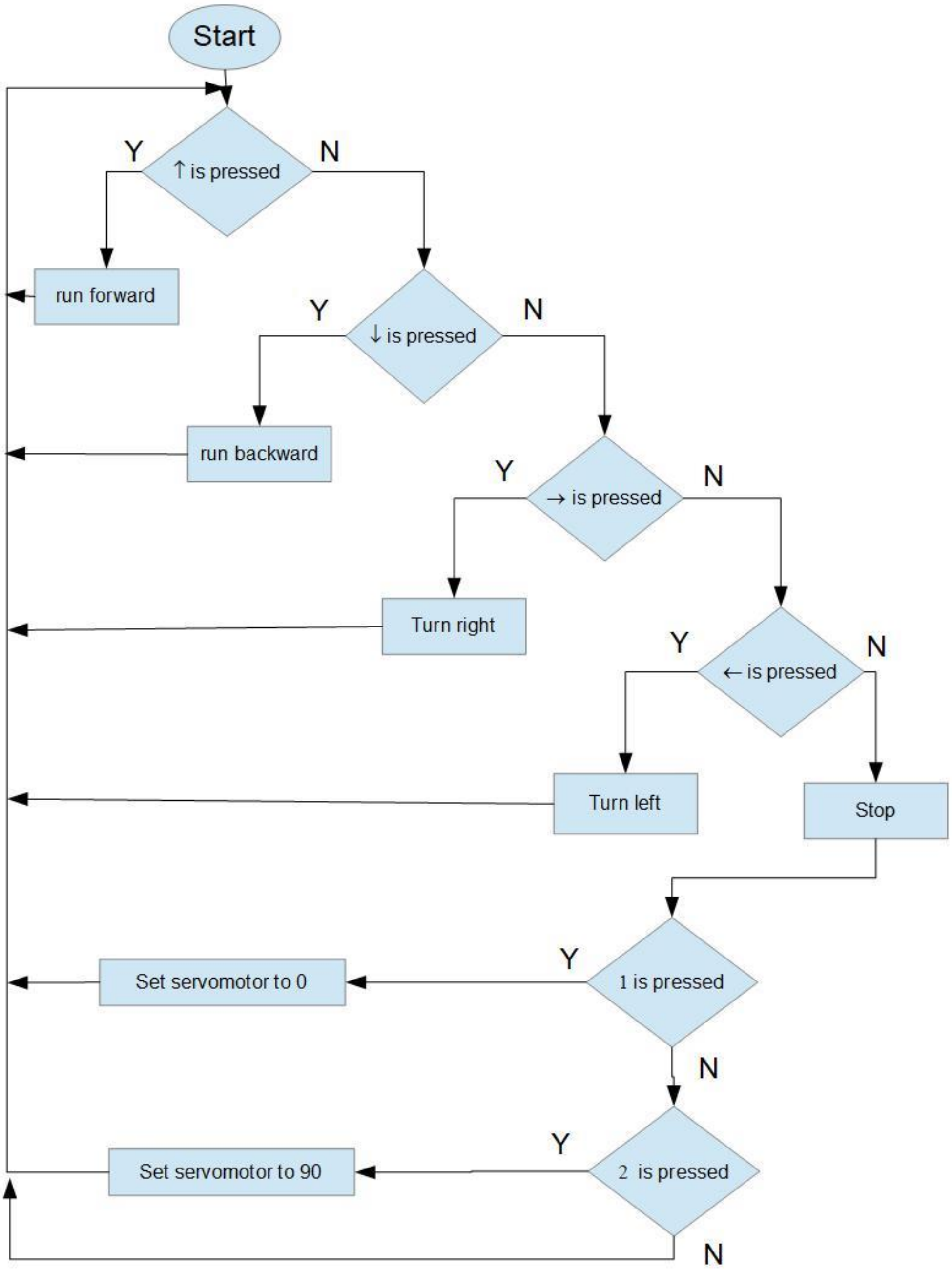
- Adjust speed and angle of turn to throw the ball and try to get close to the small ball.

- The team that gets the balls closer to the small ball gets one point for each ball.

- If remote C key is pressed then a point is added to the variable "SCORE" and a sound effect and lights is shown.

- If “SCORE” = 5 then:
    - A sound effect and lights is shown.
    - The player wins the match.
- The program waits 4 seconds.
- The program restarts.

Second version



MORE INFORMATION

DIFFICULTIES:

- Choice of the balls: they must be light so that the robot can throw them, but not too much that their movement is predictable.
- Making a good pitch is difficult. You could investigate the possibility of throwing the balls by direct push.
- For cancelling a command it is necessary to press a button on the remote control. This implies that to make a launch you have to press two buttons, one to start the launch and another to stop it.