

SAILBOAT RELAY REGATTA



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

SAILBOAT RELAY REGATTA

ABSTRACT

The activity consists in running a "relay race" in the sea, with sailboats, based on the ability of the students to answer science questions.

The race will be between two teams, which will sail at the same time. Each team of students have three sailboats: MB01, MB02, MB03 (three mBots). As the race advances, the program prompts the students for STEM questions on the computer screen: if the team answers correctly, their next boat is allowed to start sailing and continue the race. The team who first arrive at the finish line, will win.

DIDACTIC OBJECTIVES

GENERAL:

- ❖ Learning to develop the algorithm of an activity.
- ❖ Acquiring skills in problem solving.
- ❖ Acquiring motivation for the study and understanding of the contents of the different STEM areas.
- ❖ To learn the operation of different sensors and components.
- ❖ To develop computational thinking.
- ❖ Learning to work and cooperate in a group.

TECHNOLOGY:

- ❖ Learning to program communication between robots, using infrared messages.
- ❖ Learning to program the Arduino board.
- ❖ Having the first contacts with the Arduino programming language.
- ❖ To make a program that combines the operation of a robot linked to a computer program, with other autonomous robots, all working as a team.
- ❖ Learning how to use a 7-segment display, led lights, fan component and chronometer function.
- ❖ Calculating the current flowing through a lightbulb, when it is connected to a battery.
- ❖ Solving lever problems.

MATHEMATICS:

- ❖ Solving percentages and proportions.

SCIENCE:

- ❖ Calculating velocity as a function of space and time.
- ❖ Calculating density as a function of mass and volume.

STEM Subject: Science Technology Engineering Mathematics

Education Level: 12-14 years 14-16 years



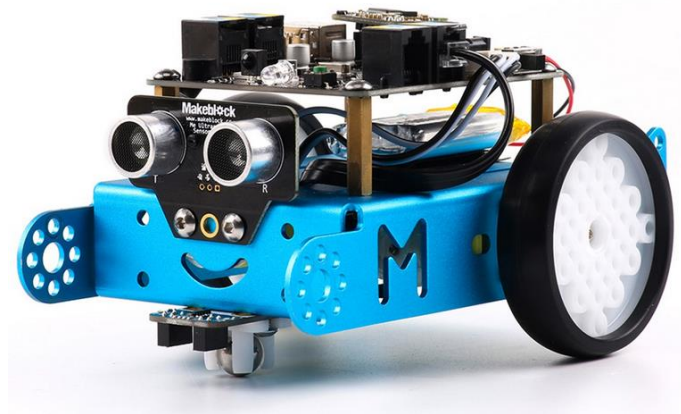
PROBLEM STATEMENT

We will combine different programs and routines in one. Therefore, the flow chart is very important for the success of the activity:

- ❖ Random approach to questions from different subjects.
- ❖ Actions conditioned on events.
- ❖ Movement, lighting effects, chronometer, fan ...
- ❖ Autonomous robots with their program recorded on the Arduino board working as a team with robots linked to a PC program.
- ❖ Communication between robots by sending infrared messages.

BOM (Bill of Materials Needed)

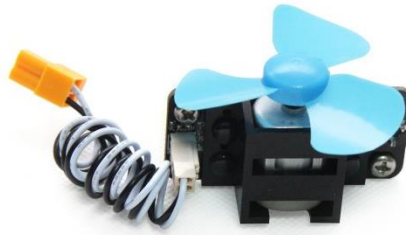
- (x6) mBots => Ref. 90054



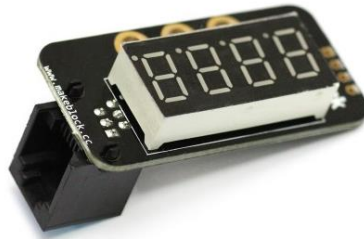
- ❖ Different beams and structures:



❖ 6 Me 130 DC Motor Pack:



❖ 2 Me 7-Segment Serial Display - Red:



❖ Rest of Attrezzo: buoys to signal the start and end of the stage. Flags for boats.

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
Mbot Robot 2'4G			6																	BI*	BI*
Motor 1	BI*																			BI*	
Motor 2	BI*																				BI*
ME 130 motor pack 5V/10000rpm (FAN)	Az	6	6		Az																
ME 7-Segment serial display	Az	2	2																		
RJ25 cables			8																		
Structures																					
Beam 0808-040			6																		
Cutttable Linkage 080			6																		
Bracket P1			8																		
Laptops			2																		
Atrezzo (not essential)																					

ACTIVITY DESCRIPTION

In this activity, each team must answer correctly two STEM questions. If they do so, the first sailboat (MB1) will start to navigate. If they do not give the proper answer, the program will ask them another question. This proceeding will continue until two questions are being answered correctly.

Time will start to count since the beginning of the program. Time will be reflected in a display located on the third mBot (MB3). This third robot is the one which will cross the finish line.

The process is repeated two more times:

- ❖ After guessing another 2 questions, the MB2 will advance towards the MB3.
- ❖ Finally, by answering the last 2 questions, the MB3 will advance towards the finish line. When it arrives, the chronometer will stop.

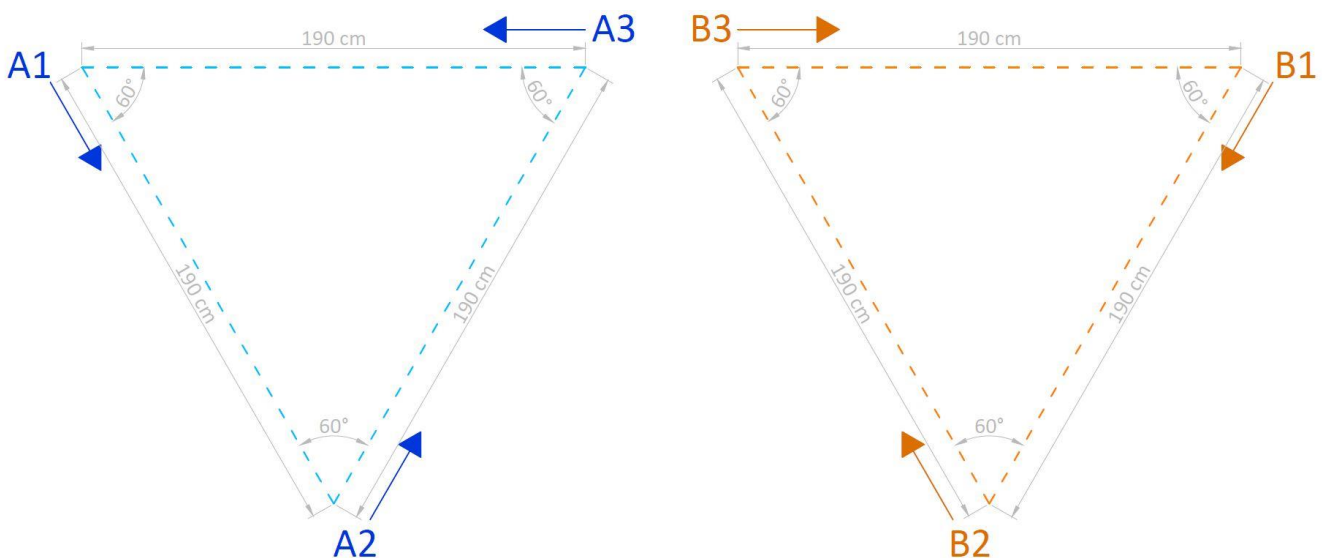
The main program will run from a PC and will be connected to the MB1 through the Wi-Fi or Bluetooth connection.

The MB2 and MB3 robots will work autonomously. Their own codes have to be recorded on their Arduino boards.

The MB1 will make a connection between the PC program and the autonomous robots (MB2 and MB3) by sending messages to them to execute their routines at the right time.

The configuration of movements and waiting times has been done according to the route for the regatta, which we can see in the sketch below.

Any change in the separation distance between mBots means that we must adjust these parameters in the affected code routines.



The complexity of the code decides in how to join the different routines without affecting the performance of the program.

There are **3 base programs for each team** (one for each of the three robots of a team). The programs of each team must be different, so that there is no interference between the messages of the two teams.

The only difference between the programs of the two teams are the messages that are sent. All the messages sent by the robot MB1 of team1 end in 1 ("chrono1", "a1", "b1", "end1"). The same for team 2, ("chrono2", "a2", "b2 ", " end2 ").

Program 1, called "starter", is the main program that the PC will run. This program manages the contest questions and the reactions of the MB1.

Programs 2 and 3 will be recorded on the Arduino boards of MB2 and MB3. The execution of the routines of these two programs, depend on the messages sent by the MB1, (it will communicate with them through the infrared port). Next, the different routines will be shown in the running order.

The codes shown are those of team 1. The codes of team 2 are identical and the only difference is that the messages end in "2" instead of in "1" and the orientation of the turn of Mb1.



ROUTINE 1: MAIN CODE (PC PROGRAM)

```
when clicked
  crono
  set led on board all red 0 green 0 blue 0
  repeat until answer = Percentage
    Math 01
  repeat until answer = Roundfractionwith numbers
    Math 02
  set minifan Port1 blow clockwise
  run forward at speed 200
  wait 5.8 secs
  set minifan Port1 blow stop
  run forward at speed 0
  repeat until answer = Velocity
    Science 01
  repeat until answer = density
    Science 02
  messageA
  wait 5 secs
  set motor M2 speed 100
  wait 1.5 secs
  set motor M2 speed 0
  repeat until answer = Round "I"
    Technology01
  repeat until answer = Round "R"
    Technology02
  messageB
  set motor M2 speed 100
  wait 1.5 secs
  set motor M2 speed 0
  wait 3 secs
  messageend
```

Adjust the waiting time.

Turn so that the message reaches mBot2

ROUTINE 2: MESSAGE "crono" SEND FROM MBOT1 TO MBOT3

```
define crono
  set led on board all red 60 green 0 blue 0
  repeat 10
    send mBot's message crono1
  set led on board all red 60 green 60 blue 0
  wait 0.5 secs
  set led on board all red 0 green 0 blue 0
```

ROUTINE 3: MBOT3 ARDUINO PROGRAM – PART ONE – MESSAGE "crono" RECEIVED

```
mBot Program
forever
  wait until mBot's message received = "crono1"
  reset timer
  set led on board all red 60 green 20 blue 0
  repeat until mBot's message received = "b1"
  set 7-segments display Port4 number timer
```

ROUTINE 4: MATH1 QUESTION

```
define Math 01
  set Number 1 to pick random 1 to 200
  set Number 2 to pick random 1 to 10
  set Percentage to Number 1 * Number 2 / 100
  ask join Calculate the join Number 1 join % of Number 2 and wait
```

ROUTINE 5: MATH2 QUESTION

```
define Math 02
set Numerator to pick random 1 to 10
set Denominator to pick random 1 to 10
set Number 3 to pick random 10 to 20
set Fraction with numbers to Numerator * Number 3 / Denominator
set Round fraction with numbers to round Fraction with numbers * 100 / 100
ask join If I have join Numerator join / join Denominator join of join Number 3 , How much do I have? and wait
```

ROUTINE 6: SCIENCE1 QUESTION

```
define Science 01
set Space to pick random 1 to 100
set Time to pick random 1 to 20
set Velocity to round Space / Time * 100 / 100
ask join If an object does join Space join m. every join Time join seconds, what is its velocity? and wait
```

ROUTINE 7: SCIENCE2 QUESTION

```
define Science 02
set mass to pick random 100 to 200
set volume to pick random 20 to 60
set density to round mass / volume * 100 / 100
ask join If an object has a join mass join kg mass and a join volume join m3 volume, what is the value of its density? and wait
```

ROUTINE 8: MESSAGE "a" SEND FROM MBOT1 TO MBOT2

```
define messageA
  set led on board all red 60 green 0 blue 0
  repeat 10
    send mBot's message a1
  set led on board all red 0 green 0 blue 20
  wait 0.5 secs
  set led on board all red 0 green 0 blue 0
```

ROUTINE 9: MBOT2 ARDUINO PROGRAM – MESSAGE "a" RECEIVED

```
mBot Program
  forever
    set led on board all red 0 green 0 blue 0
    wait until mBot's message received = "a1"
    set led on board all red 0 green 150 blue 0
    wait 0.5 secs
    set mini fan Port1 blow clockwise
    run forward at speed 200
    wait 6.5 secs
    set mini fan Port1 blow stop
    run forward at speed 0
```

ROUTINE 10: TECHNOLOGY1 QUESTION

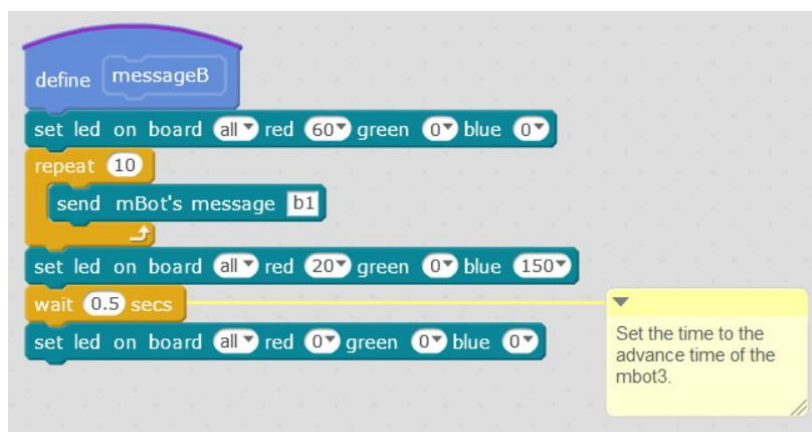
```
define Technology01
  set V to pick random 1 to 100
  set R to pick random 1 to 12
  set I to V / R
  set Round "I" to round I * 100 / 100
  ask join Calculate the current flowing through join R join Ω lightbulb, when it is connected to join V V battery. and wait
```

ROUTINE 11: TECHNOLOGY2 QUESTION



```
define Technology 02
set Force to pick random 1 to 200
set Effort Arm to pick random 1 to 12
set Resistance Arm to pick random 1 to 12
set Resistance to Force * Effort Arm / Resistance Arm
set Round "R" to round Resistance * 100 / 100
ask join How much load weight can an effort of join Force join N lift with a classone lever, if the effort arm is join Effort Arm join metres long and the load arm is join Resistance Arm metres long? and wait
```

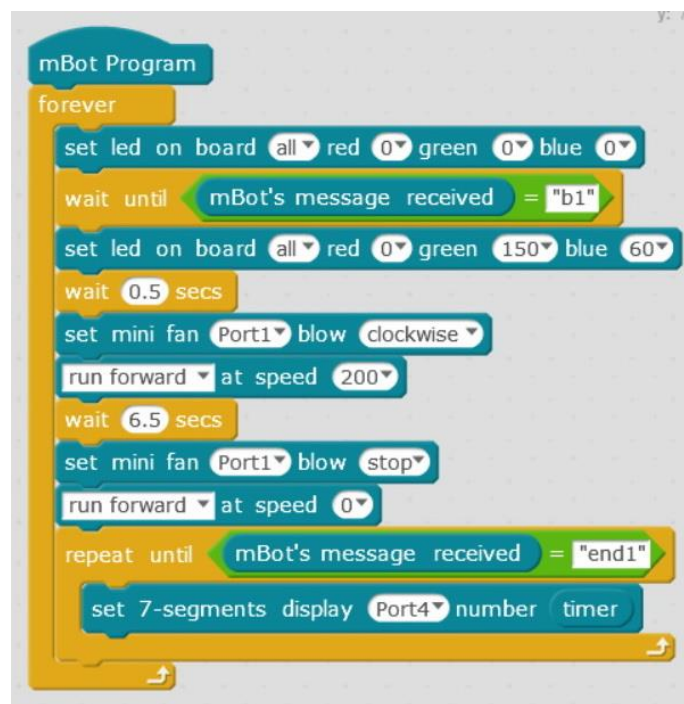
ROUTINE 12: MESSAGE "b" SEND FROM MBOT1 TO MBOT3



```
define messageB
set led on board all red 60 green 0 blue 0
repeat 10
  send mBot's message b1
set led on board all red 20 green 0 blue 150
wait 0.5 secs
set led on board all red 0 green 0 blue 0
```

Set the time to the advance time of the mbot3.

ROUTINE 13: MBOT3 ARDUINO PROGRAM – PART TWO – MESSAGES "b" OR "end" RECEIVED



```
mBot Program
forever
  set led on board all red 0 green 0 blue 0
  wait until mBot's message received = "b1"
  set led on board all red 0 green 150 blue 60
  wait 0.5 secs
  set mini fan Port1 blow clockwise
  run forward at speed 200
  wait 6.5 secs
  set mini fan Port1 blow stop
  run forward at speed 0
  repeat until mBot's message received = "end1"
  set 7-segments display Port4 number timer
```


ROUTINE 14: MESSAGE "end" SEND FROM MBOT1 TO MBOT3

```
define message end
set led on board all red 60 green 0 blue 0
repeat 10
  send mBot's message end1
set led on board all red 0 green 20 blue 150
wait 0.5 secs
set led on board all red 0 green 0 blue 0
stop all
```

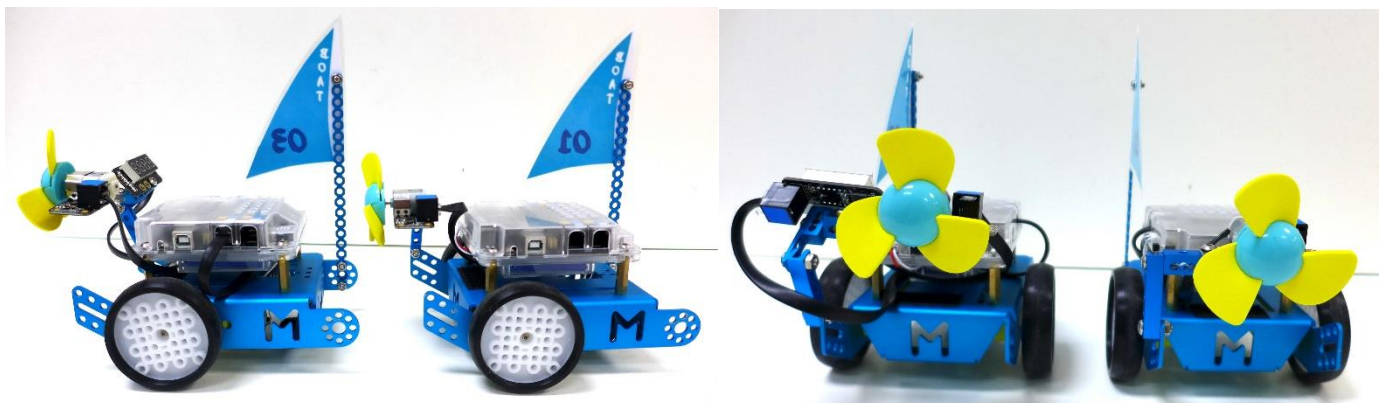
We will record the code from mbot2 and mbot3 in their Arduino boards. In this way, the mBots will work independently of the computer and there only will be need it two laptops to control six robots.

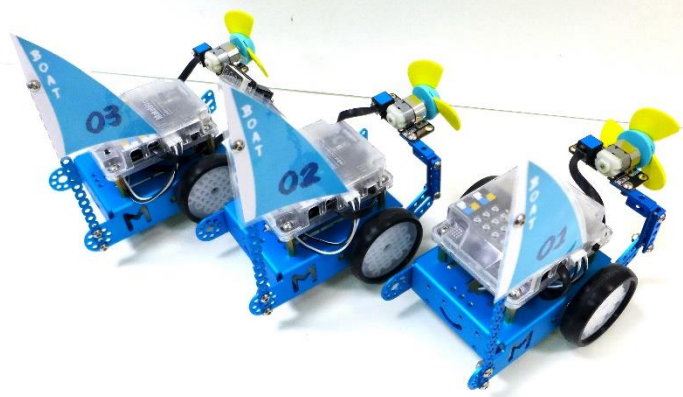
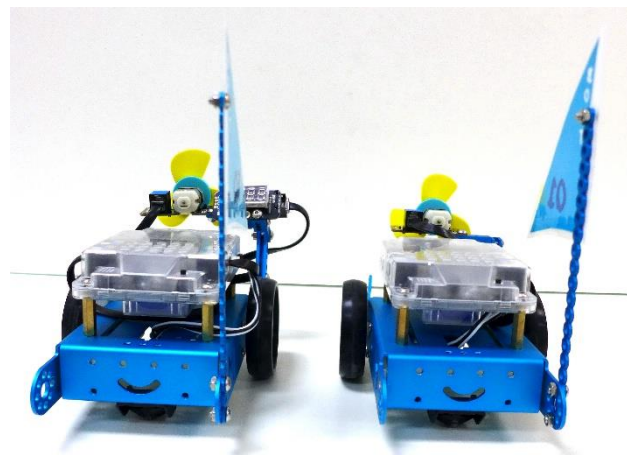
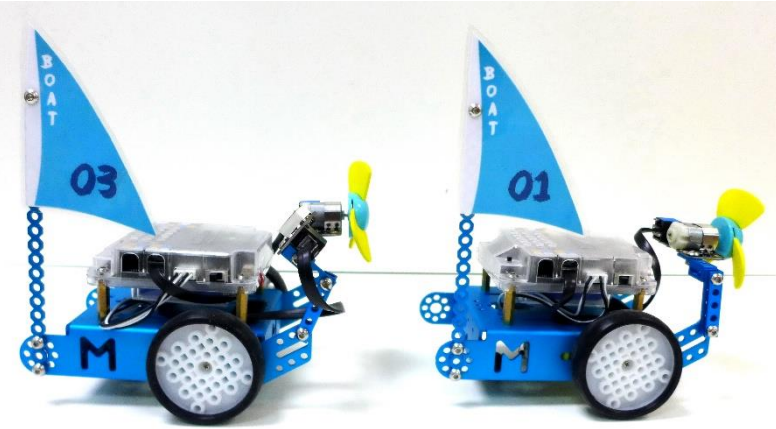
How to load a program on the arduino mBot board using mBlock:

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.

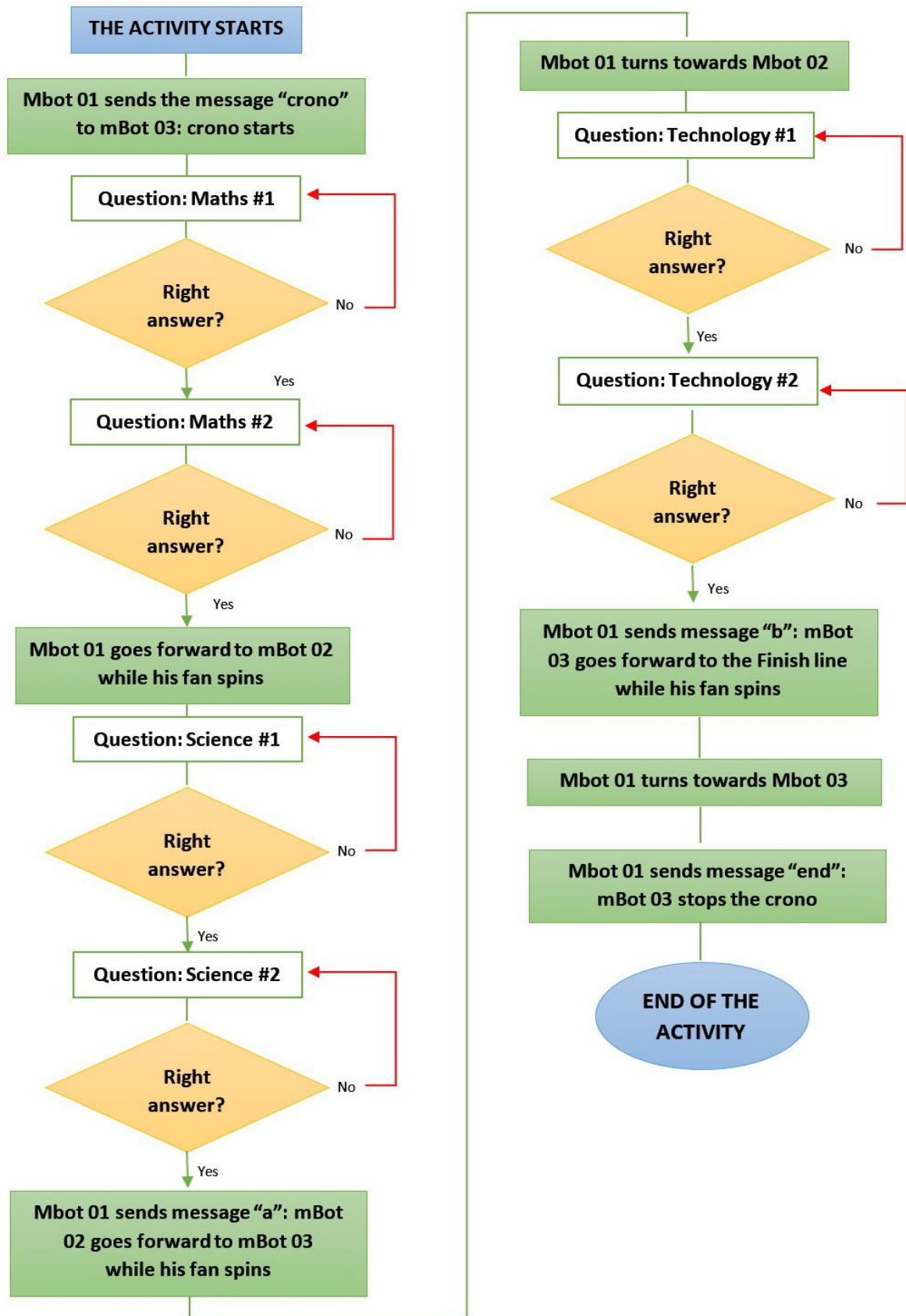
Once, the programming is finished, we start building up THE STRUCTURE, where all the mechanical elements will be set. Also the electronic elements.





FLOW CHART

SAILBOAT RELAY REGATTA



STUDENT'S EVALUATION

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

MORE INFORMATION

DIFFICULTIES:

1. Finding the way to combine the different routines without affecting the performance of the program.
2. To transform the initial idea into a flowchart and the flowchart into code that works well.
3. To get 3 robots to work as a team using only one pc.
4. Combining six random questions from different subjects in a single code.
5. On many occasions some mBots do not move in a straight line.
6. The speed of the mBot depends on the charge level of their batteries.
7. In order for infrared messages to be received, they must be sent repeatedly for a period of time (see the code). If an infrared message is sent in a timely manner it is normal that it is not received by the receiver.
8. The infrared signal is very sensitive to intense light. To enable the reception of infrared messages the mBot activity has to be carried out in a place without direct sunlight and with a moderate light intensity.
9. The infrared transmitter/ receiver is located on the front of the Mbot. To enable the reception of the messages, the robots must be placed in a position as close as possible (see the sketch). According to our tests, an angle of 60° and a distance of 1'90 meters are parameters that work.
10. During the activity, the transmitter mBot has to change its position to always comply with the previous conditions.

