

## On Standby



Xingcheng is not a metropolis located in the plains, but a huge mountain city. Mark and Rex will have to go across various steep “maker” mountains in order to arrive at IDA (Interstellar Development Authorities). There are hundreds of steep slopes and dangerous bends in the way, even seasoned racing drivers will still have to reverse and be careful about this journey.

Haha, but it is just as easy as driving on the highway for Mark, the shoo-in in car racing field. Let me tell you the secret that Mark is not driving by himself...

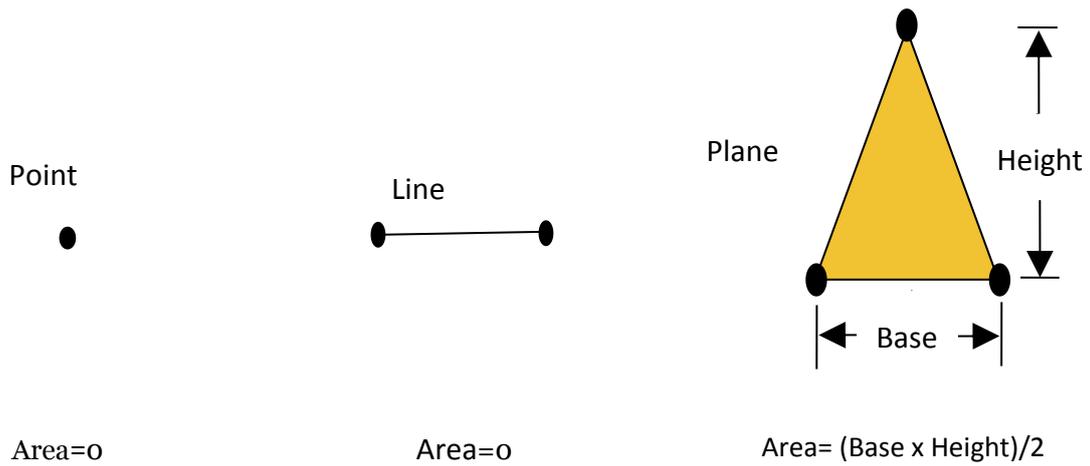
### Learning Objectives

If we want the vehicle with only two wheels to maintain balance, we'll need to add high-precision sensors and set complex algorithm for the vehicle. But when we add an additional wheel on this vehicle, then it will be able to stand upright on the floor easily.

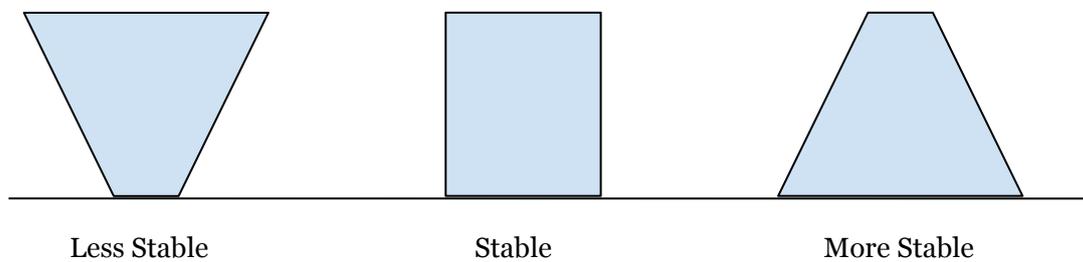
We will introduce Points, Lines, and Planes through assembling a three-wheeled robot. We then will observe how this robot works and learn how to write programs to control the robot remotely.

### Scientific knowledge

As we all know, to maintain an object not to fall down is to keep the plumb line that passes through the center of gravity inside the bottom face area (refer to Chapter 3). But if an object stands on the floor at only one point or two points, the plumb line will have nowhere to stay, and it will be very difficult for the robot to keep balance without adding control system. But why the robot can stand upright on the floor easily after adding a third wheel?

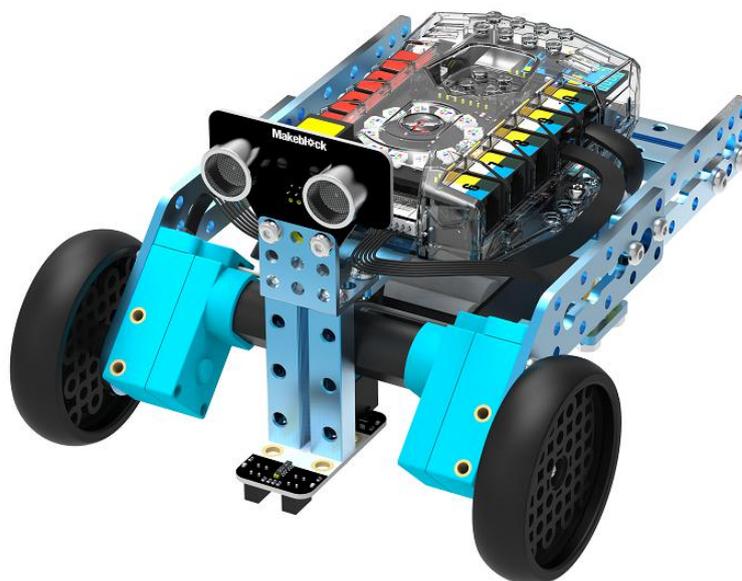


So when the object has area, then it will be able to stand on the floor. The larger the area is, the more stable the object.

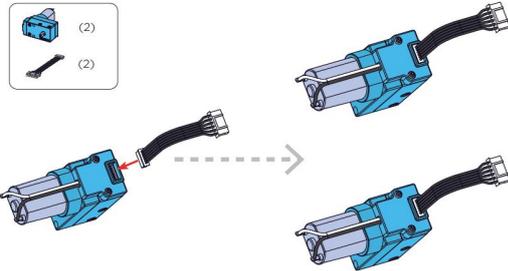
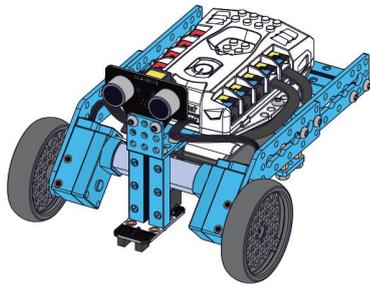


## Assembly Preparation

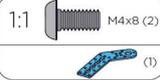
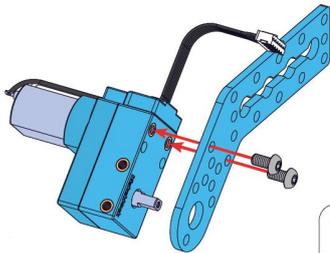
Build a three-wheeled racing robot with mBot Ranger.



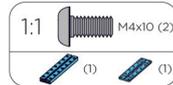
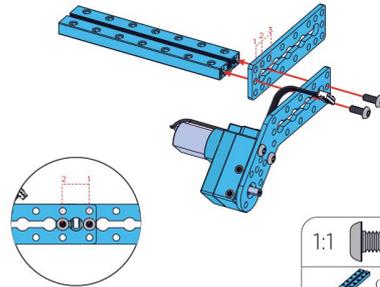
# Assemble Dashing Raptor



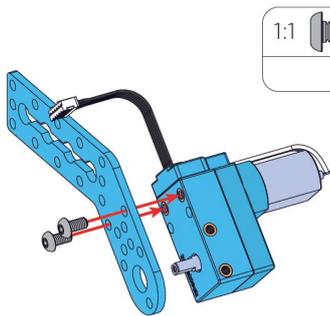
1



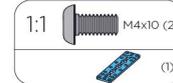
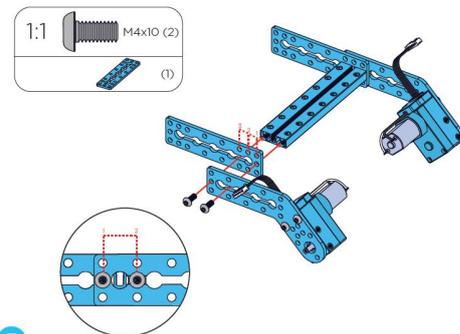
2



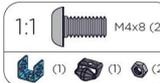
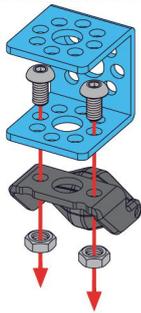
3



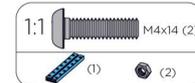
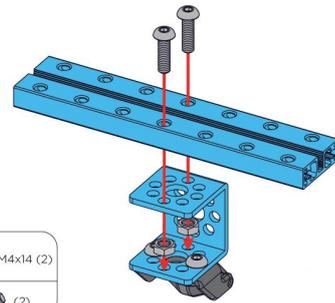
4



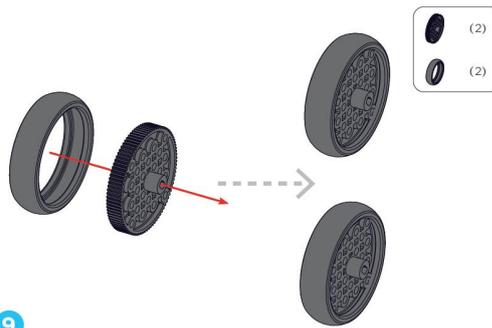
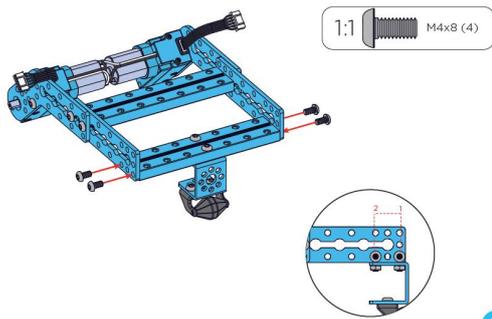
5



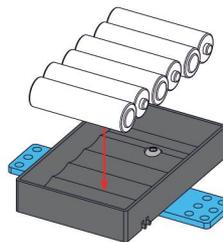
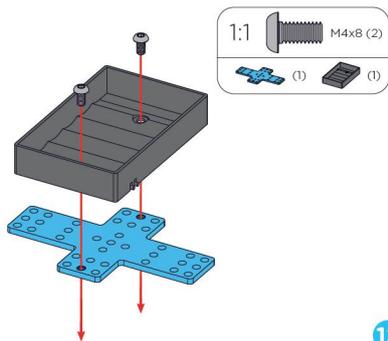
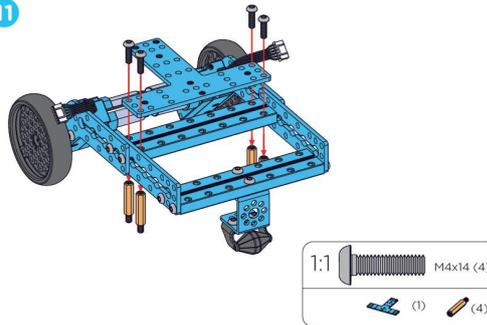
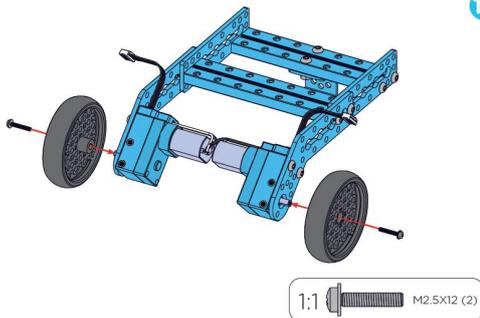
6



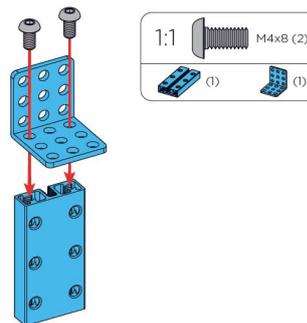
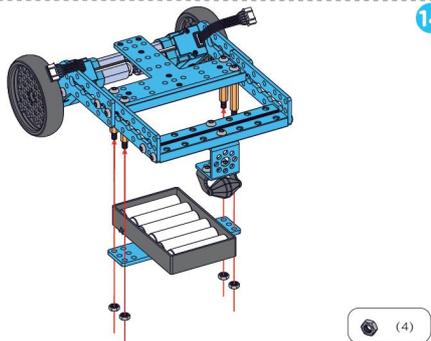
7

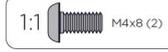
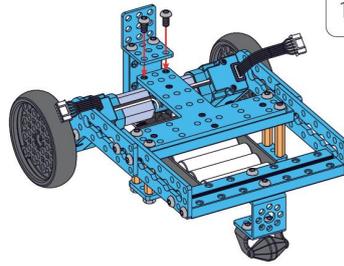
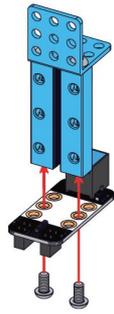


8 9  
10 11

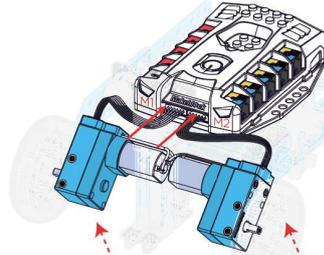
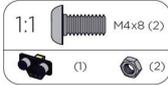
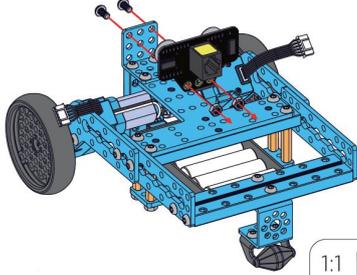


12 13  
14 15





16 17  
18 19

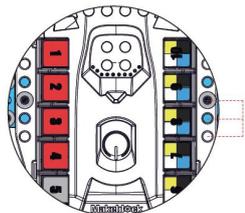
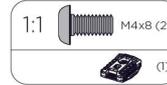
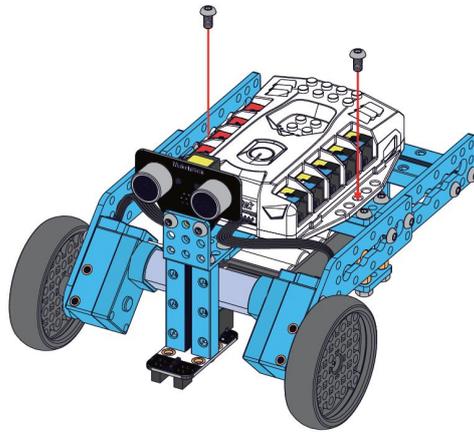


Connect the left motor and right motor to the corresponding port of Me Auriga.

Right motor (M1)

Left motor (M2)

20



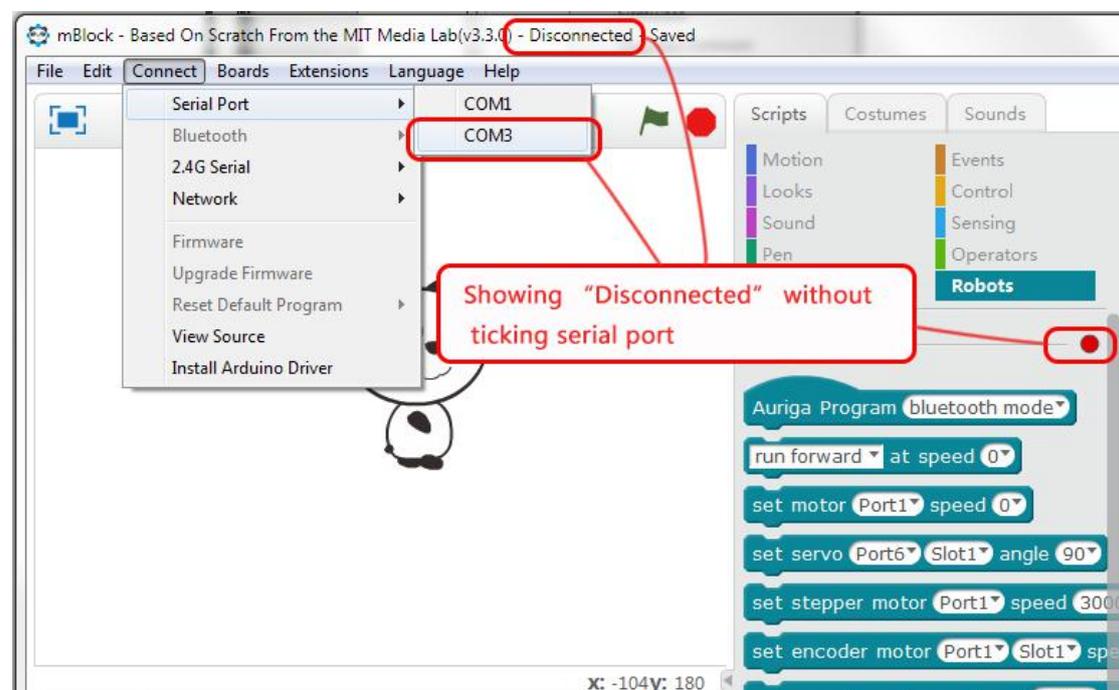
## Learning Task

### Learning task 1 - Upgrade Firmware & Connect the Robot with Computer via Bluetooth

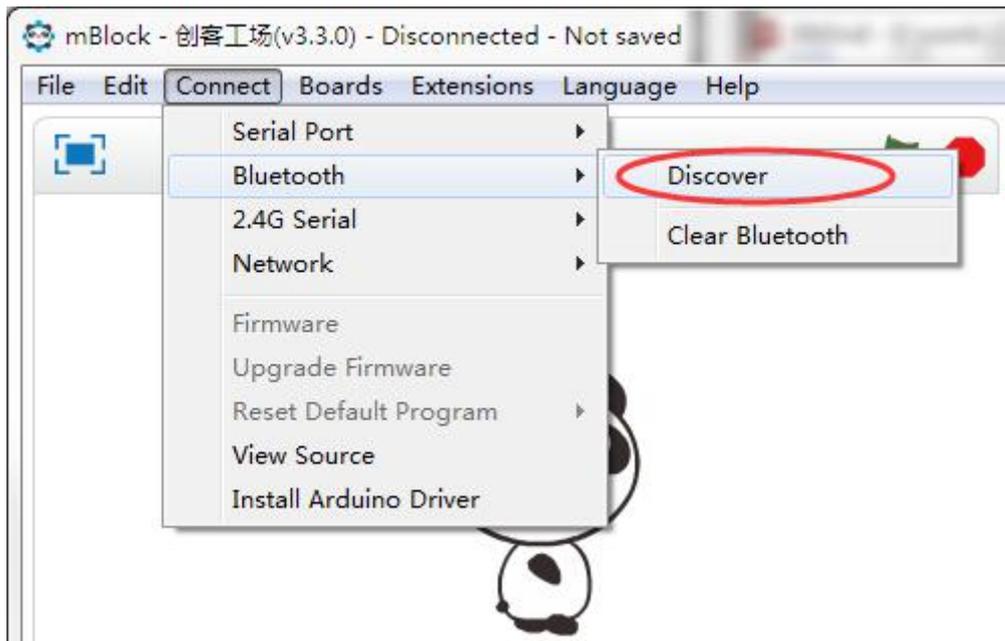
Prepare the three-wheeled racing robot, connect Me Auriga (mainboard) on Ranger with computer via USB cable, then click "Upgrade Firmware" (refer to Chapter 2).

Before removing the USB cable from Me Auriga, we need to disconnect Ranger with mBlock in order to prevent the system from information misjudgment.

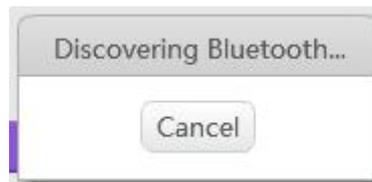
Click "Connect" tab, then click "Serial port" -> "COM x" to disconnect Ranger with mBlock. As shown in the picture below:



Get ready to connect Ranger with mBlock via Bluetooth: click the "Bluetooth" -> "Discover"



Don't forget to turn on the Bluetooth of your computer, then there will be a window showing "Discovering Bluetooth...", as shown:



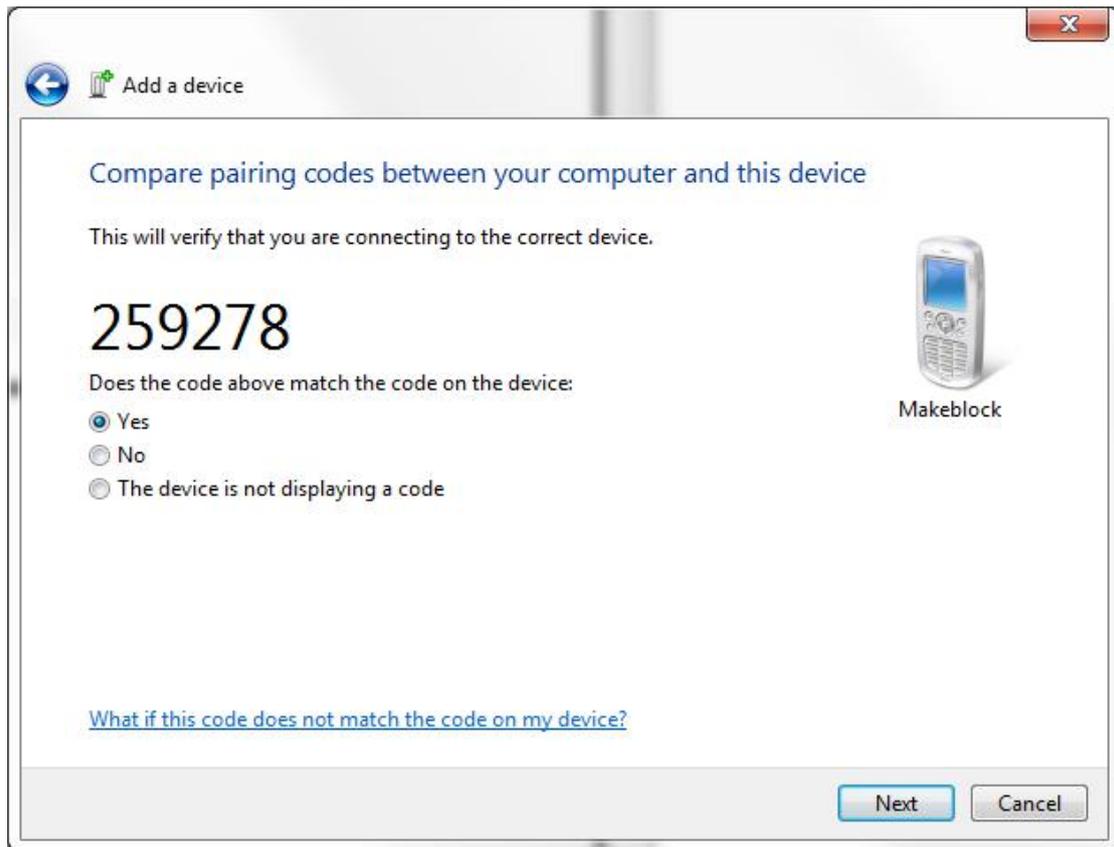
After searching the Bluetooth device, there will be a Makeblock Bluetooth MAC address showing in the list. Then click the address to start connecting.



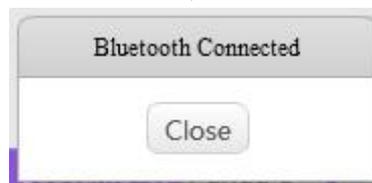
A small window showing "Connecting Bluetooth":



A window showing a password for devices that haven't been connected with the computer before:



Click "Yes" and there will be a reminder showing "Bluetooth Connected":



Click "Close" to close the window. There will be "Bluetooth Connected" in the mBlock title bar and the button (light) will turn green, as shown below:



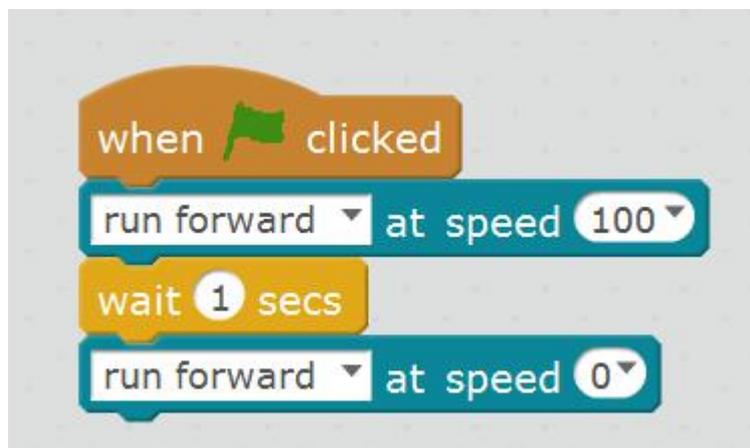
Congratulations, Ranger is connected with mBlock successfully! We can now control the robot with Bluetooth, which free us from controlling the robot with USB cable.

## Learning Task 2 - Command Ranger to Go Forward

Place Ranger on the floor and begin to control the robot remotely. We will use block "run forward at speed 0" under "Robots" Scripts.

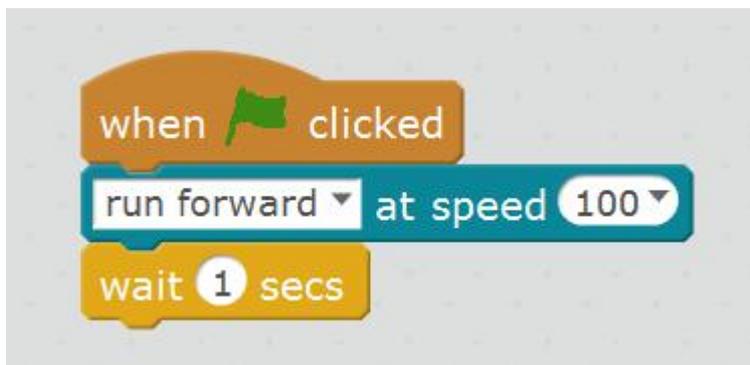
		<p>The speed of the robot moves forward is zero.</p>
<p><b>Function:</b> control the movement of Ranger</p> <p><b>[Forward]:</b> Set the moving direction of Ranger. The default value is (run forward), you can click on the inverted triangle to select (run backward), (turn left), (turn right) in the drop-down list.</p> <p><b>(Speed):</b> Set the speed of the motor. The default value is [0], it means that the motor doesn't rotate. The value ranges from -255 to 255, "255" means the motor is rotating forward at maximum speed.</p> <p>A negative value indicates the motor is rotating reversely, for example, "- 255" means the motor is rotating reversely at the maximum speed.</p>		

Implement the following program and see if Ranger will stop after going forward for 1 second.



What will happen if there is not a “run forward at speed 0” block at the end of the program (as the picture below)?

Note: There is the red button in the upper right corner of the Stage for you to stop the motor if Ranger sprints forward.



We know that the motor will not stop automatically after implementing the program, hence we need to use a command to stop it.

## Task

### Task 1 - Control the Robot Remotely for 5 Minutes

The blocks under “Events” Scripts can be used to implement programs. We have



introduced how to use “” block in previous chapters. Of course, there are other blocks for you to implement a series of blocks (program) according to different conditions.



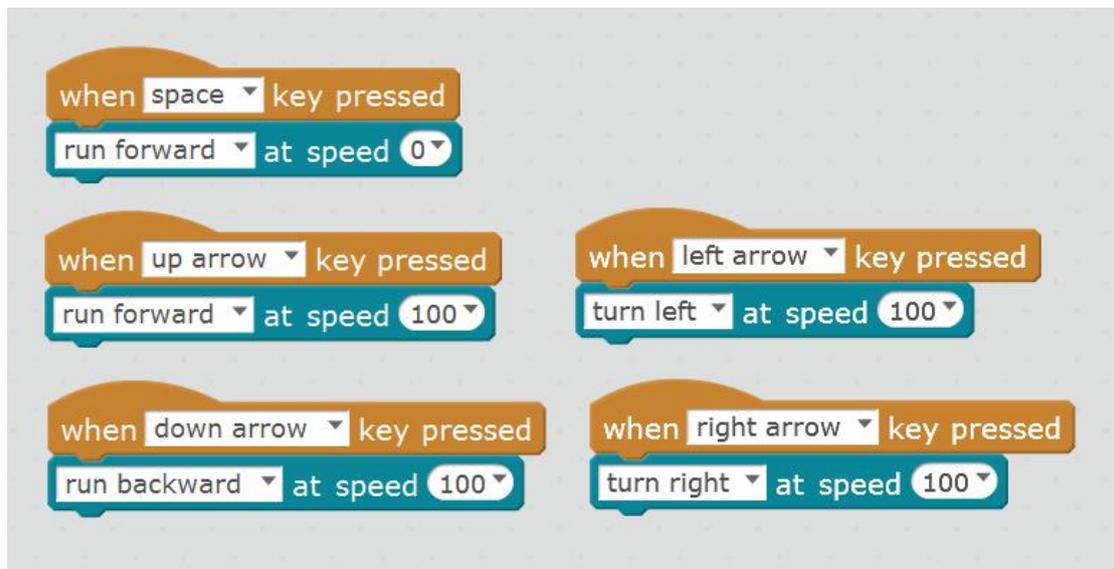
		when press the space key
<p>Press the space key to implement a series of blocks (program)</p> <p>Note: if pressing the space bar fails to implement the program, please check your input method and make sure you use the English input method.</p> <p>You can also click the inverted triangle to select different keys to implement the program.</p>		

When press the arrow keys (up arrow, down arrow, left arrow, right arrow) on the keyboard, Ranger will move forward, backward, turn left, and turn right respectively. The functions of these keys are as follows:

Keys	The movement of Ranger
up arrow	Ranger moves forward
down arrow	Ranger moves backward
left arrow	Ranger turns to the left

right arrow	Ranger turns to the right
space key	Ranger stops moving

We design the program as follows:



This example does not include “when green flag clicked” block, so we can see that there are many ways to implement the program.

In this program, you can press the space key or red button in the upper right corner of the Stage to stop Ranger (set the motor speed to zero).



Now it's time to control your Ranger and play with it in various games!

Let's go!

## Task 2 - The Improvement on Controlling Ranger Remotely

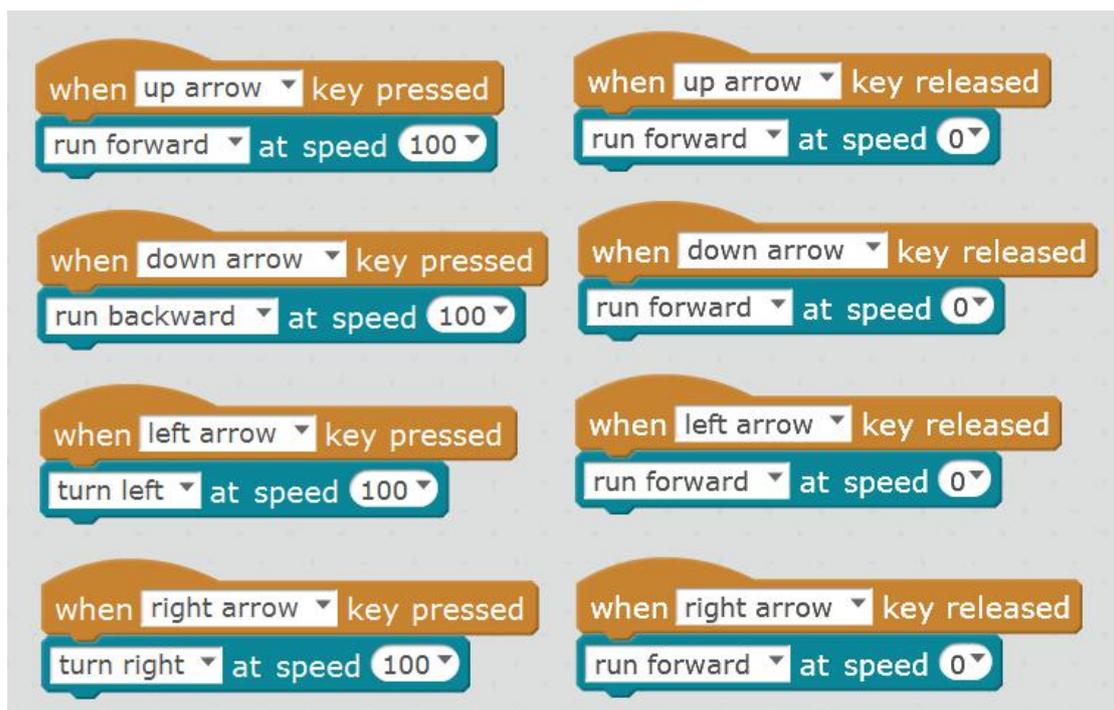
In previous task, Ranger will keep moving forward and will not stop after pressing the “up arrow” key. The motor speed is set as 100, hence we will need to press the space key to set the motor speed to 0.

But is there any other way to control the robot? Of course, we can use



<p>Motion</p> <p>Looks</p> <p>Sound</p> <p>Pen</p> <p>Data&amp;Blocks</p> <p><b>Events</b></p> <p>Control</p> <p>Sensing</p> <p>Operators</p> <p>Robots</p>		when release the space key
<p>Function: detect whether the keys has been released</p> <p>When the space bar is released, implement the command under this hat block.</p>		

Add a “when...key released” block to the program. When we release the key, the motor speed will turn to zero and Ranger will stop moving.



when up arrow key pressed run forward at speed 100	when up arrow key released run forward at speed 0
when down arrow key pressed run backward at speed 100	when down arrow key released run forward at speed 0
when left arrow key pressed turn left at speed 100	when left arrow key released run forward at speed 0
when right arrow key pressed turn right at speed 100	when right arrow key released run forward at speed 0

Have a try and see whether your Ranger is doing exactly as you set in the program. It is very convenient for using this software to control Ranger remotely.

## Challenging Task

### Challenging Task 1 - Using Different Keys to Control the Robot

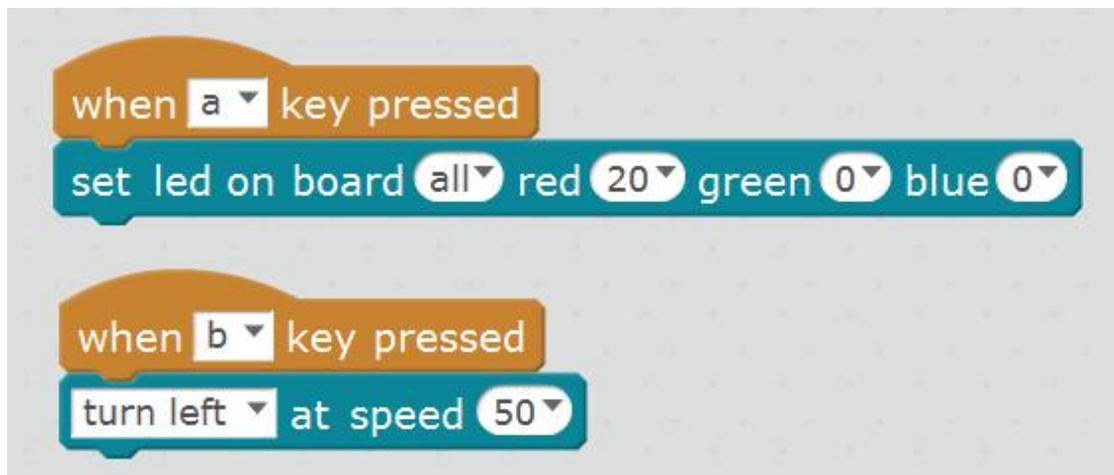
Try to use different keys to let Ranger move at different speed. For example:

1. When press the “a” key, Ranger moves to the left at a slow speed.
2. Add a light control function to the robot. When press the “b” key, all the RGB LEDs on Me Auriga turn red.

Please try to design the program on you own and fill in the table below:

Keys	The movement of Ranger
a	turns to the left at a slow speed
b	

Reference:



### Challenging Task 2 - Using More Keys to Control the Robot

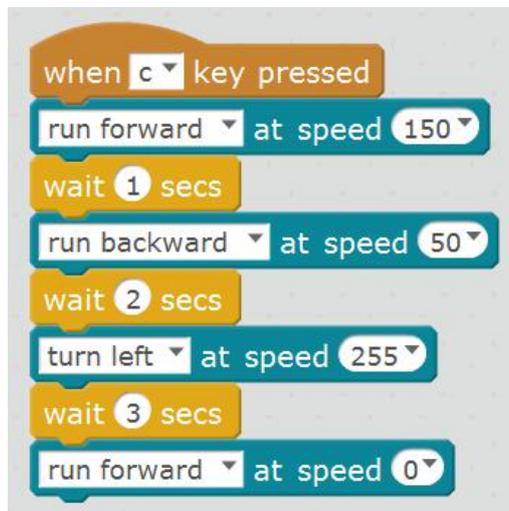
Use more keys and combine a series of blocks to make the robot dance. For example, when press "c" key, let Ranger do a series of actions:

move forward quickly→wait for 1 second→go backward slowly→wait for 2 seconds→turn to the left quickly→wait for 3 seconds→stop

Please fill corresponding blocks in the table below.

The movement of Ranger	Corresponding Blocks in mBlock
move forward quickly	
wait for 1 second	
go backward slowly	
wait for 2 seconds	
turn to the left quickly	
wait for 3 seconds	
stop	

Reference:



### Conclusion of this chapter

Most of remote control cars we played with when we were children are bought at department stores or shopping malls. We can play with these cars just by simply putting batteries in. When Ranger appears, the situation is totally different. We can assemble Ranger all by ourselves and write programs to add different functions to the robot. There are dozens of keys that can be used to control the robot, what would you do?



Mark entered the IDA with a super access card, and swipe it all the way to the director's office.

"Sit down and have a cup of tea, I know what you want to ask," said the director. It seemed that the director had expected he would come. The director then drew a square in the air and clapped his hands twice, soon a stereographic projection of the solar system appeared.

"How old are you, I can't believe you are still playing with this," Mark said with a derisive tone, playing with the director's vintage stereo imaging device, "you said you know what I want to ask, then answer my questions!"

Director then amplified the image of the solar system with his fingers and focused on a long, narrow shadow in the northern hemisphere of Mars. "We have observe this situation for over three weeks. The signal most likely came from here, this is all data we have received so far." said the director, taking a square transparent crystal out from the computer. He then put it on a business card and handed it to Mark.