Chapter one

Basic Programming

Lesson one

Little Elephant Fruit Piano

Classmates, we know that elephants are herbivores, and they like to eat most are bananas and other fruits except for grass and leaves, of course, elephant living in the forest is no exception. One day, a magician came to the city, she has a magic which can make the fruit sound.The elephant was amazed, and he also secretly learned to gather a lot of fruit to play, the friends in the forest were stunned.

Do you want to finish the beautiful piano music with elephant?

Let's have a try!

Study Task

Task 1: Learn how to use programming software.

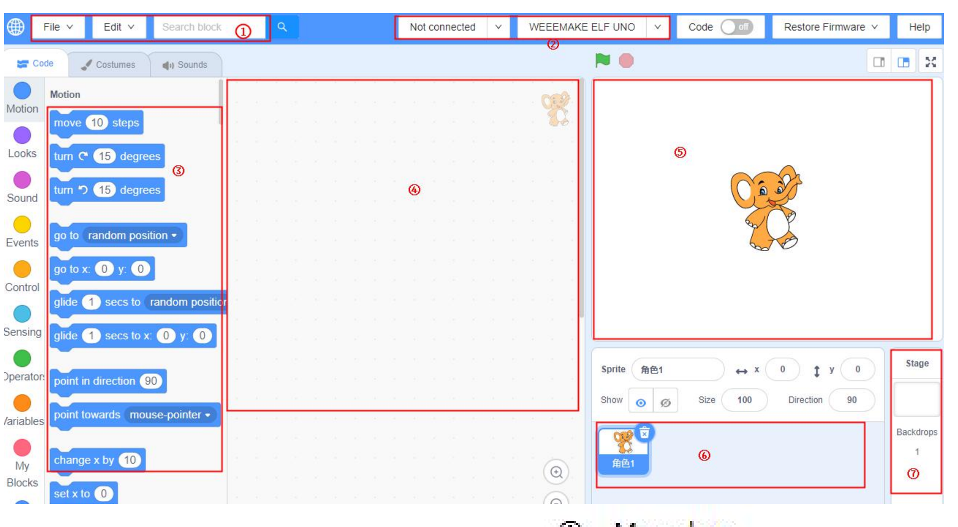
Task 2: Drive hardware in online mode.

Knowledge Point

1. Learn and be familiar with WeeeCode programming software

Click the desktop icon 

to enter the interface, you can see that "software interface" is divided into seven functional areas. As shown below.

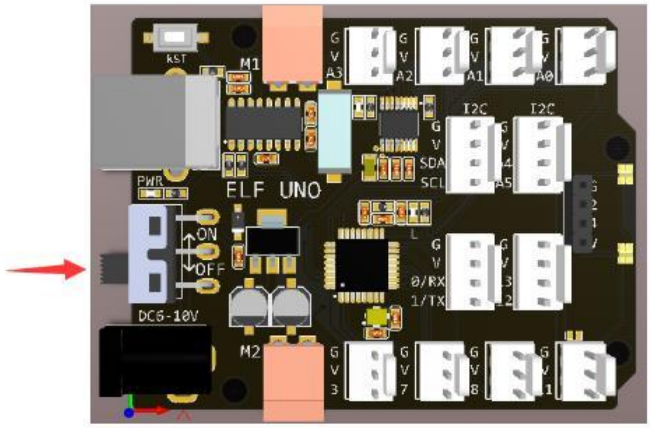


①、Menu bar
②、Hardware connection Settings area
③、Scripts menu
④、Programming area
⑤、Stage area
⑥、Role editing area
⑦、Stage editing area

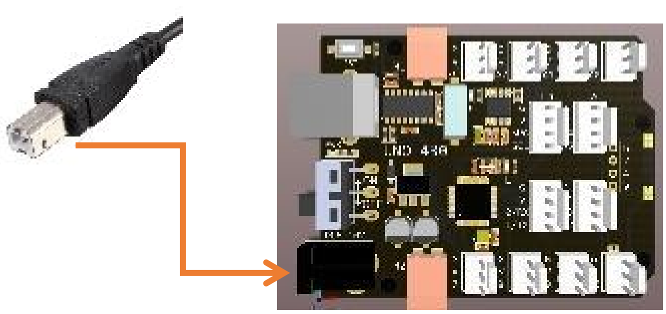
2. In the use of the interface, using scripting instructions and programming skills to design and execute tasks.

3. Online debugging

Turn on the Switch of the "Arduino" controller with battery installed, and be aware that there may be control program downloaded in the controller.



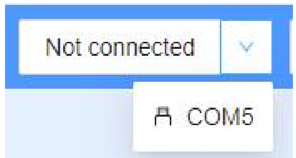
Insert the data line Type-A port into the USB port of the computer end, then insert the data line Type-B square port on the Arduino controller, be careful not to damage the USB interface.



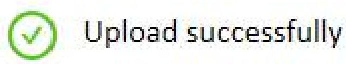
After confirming that the controller power is turned on and the data cable is connected successfully, open the WeeeCode software 

on the PC, select the

WEEEMAKE ELF UNO motherboard, 

and click the serial port, 

select online firmware debugging, 

when the upload is successful. 

You can get thecontroller to communicate with the computer in real time.

Practice

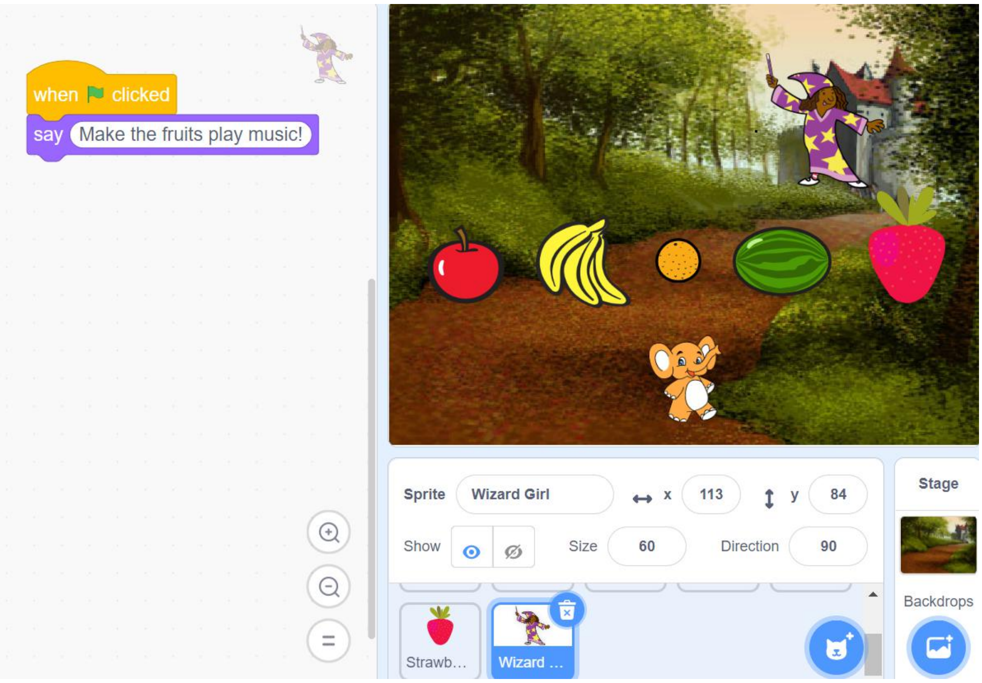
Classmates, through the above learning, now start programming.

(1) Making cartoon scenes

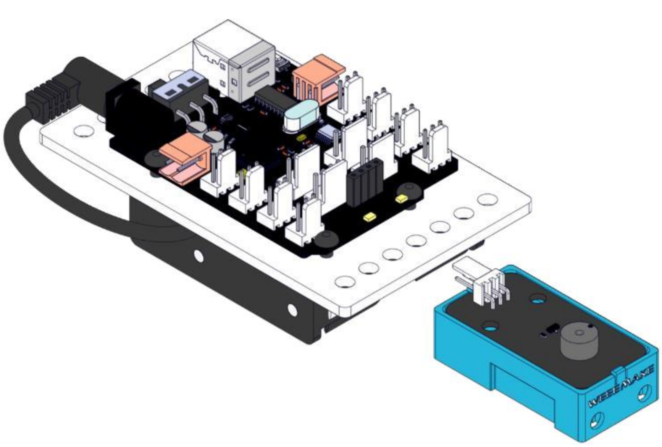
In WeeeCode software, create apple, banana, orange, watermelon and strawberry characters and put them in a suitable place; Modify the role of the elephant to a reasonable size. Create a female magician character and place it in the right place and adjust the role to the right size.

(2) Writing character programs

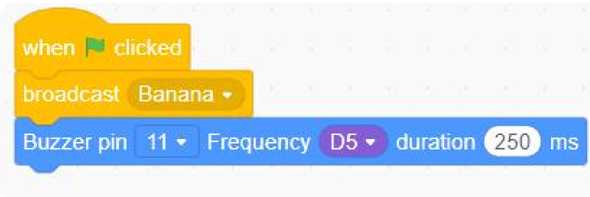
1. Female magician says the words "Make the fruits play music!" at the beginning, such as the program:

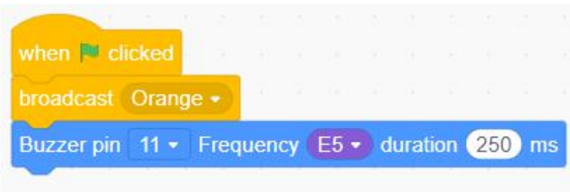


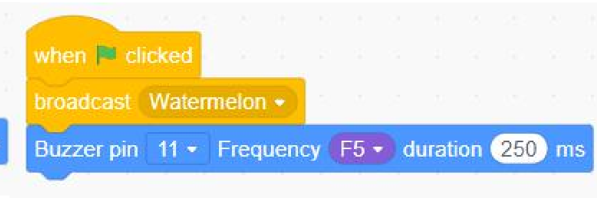
New electronic components: The buzzer is an integrated structure of the electronic signal, which belongs to an electronic component. With DC voltage or AC voltage, widely used in a variety of fields, such as: computer industry (mainboard buzzer, computer buzzer) printer (control board beaken), copier, alarm industry (alarm buzzer), electronic toy (music buzzer), agriculture, automotive electronic equipment industry (car buzzer, reversing buzzer, car buzzer, motorcycle buzzer) telephone (environmental buzzer), timer, air conditioning, medical equipment and other electroacoustic industries and environmental monitoring.

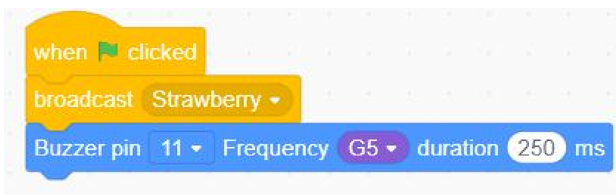


Connect the buzzer to digital port 11 using KF2510 3P terminal wire.

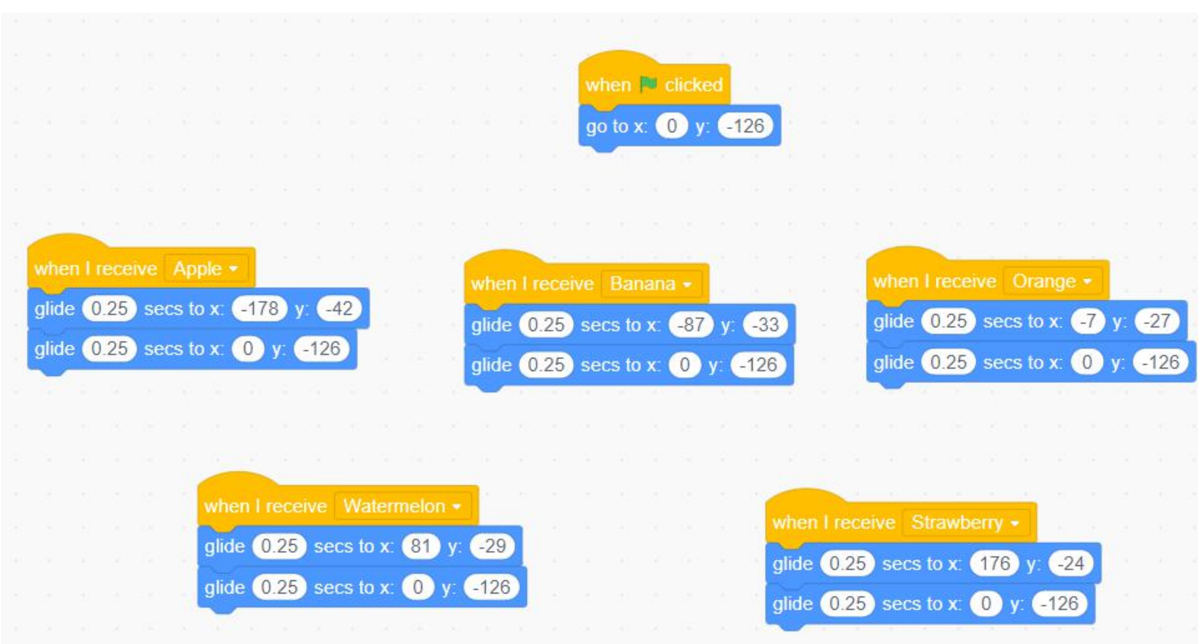




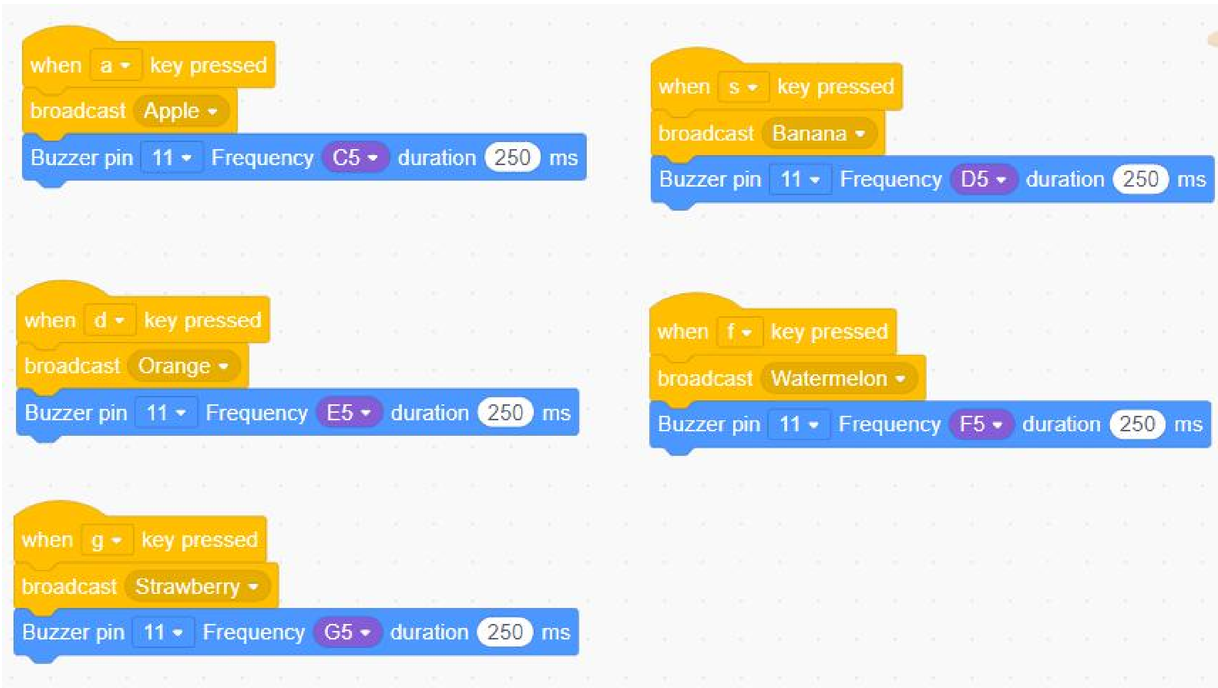




(3). Write elephant program, every elephant have to run to the location of the fruit, so must know the fruit coordinates, so the procedure of tinkling elephant is as follows:



(4). When clicking on the role with the mouse, some students find it difficult to keep up with the rhythm of the music. If not use the computer keyboard to play music, in fact very simple, as long as the fruit of the program slightly changed a bit.The procedure is as follows:

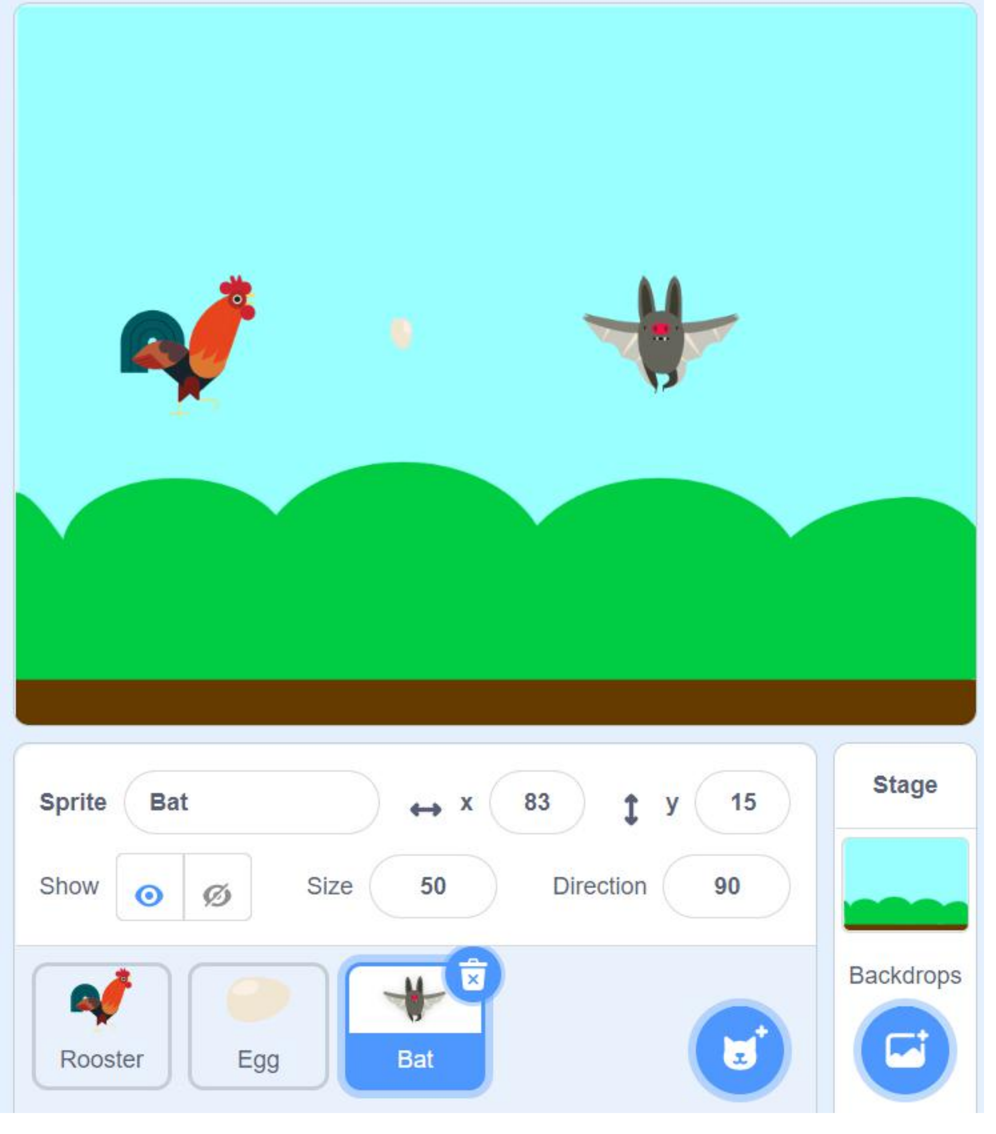


Classmates, after learning, make your own small elephant fruit piano now.

Can a fruit piano play any other music?

Lesson Two
Bats Wars Game

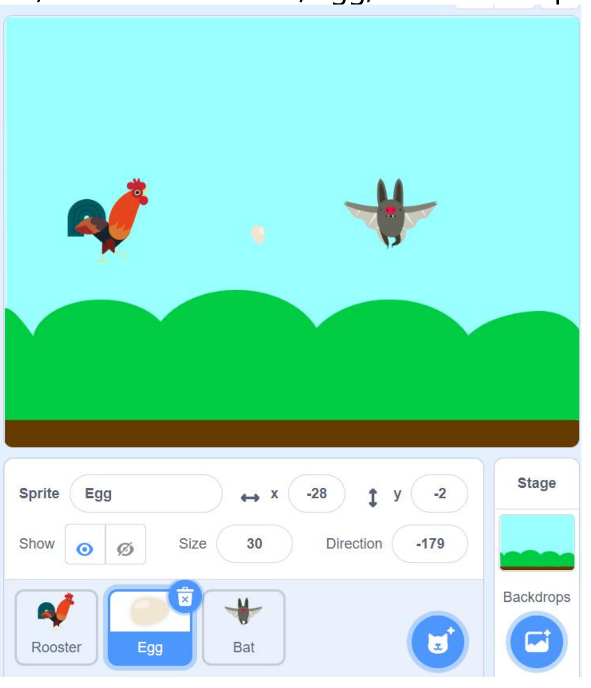
The elephant has a good friend in the forest. He is different from other similarities, then what’s his ability? In fact, he is a rooster. Because he can fly, everyone calls him "Aircraft". Recently, a number of uninvited guests come to the forest, they always steal the eggs of hen. They are bats. In order to protect the eggs of the hen, the rooster started fighting with the bat. However, the bat also has a specific skill, which is duplicate itself repeatly. Can the rooster defeat them?



Practice

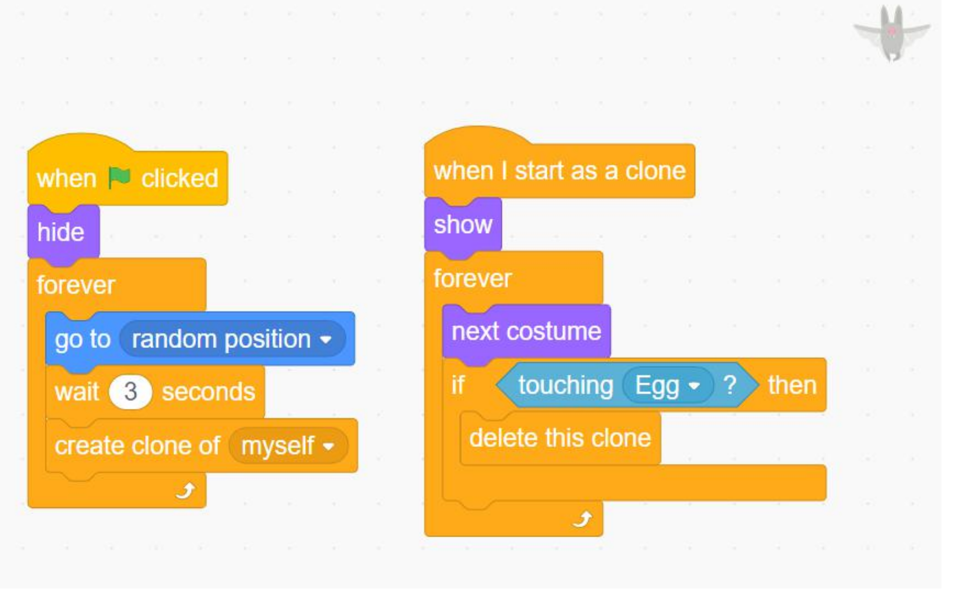
(1) Making cartoon scenes

In the WeeeCode software, create the rooster, egg, and bat and place them in right place.



(2) Writing character programs

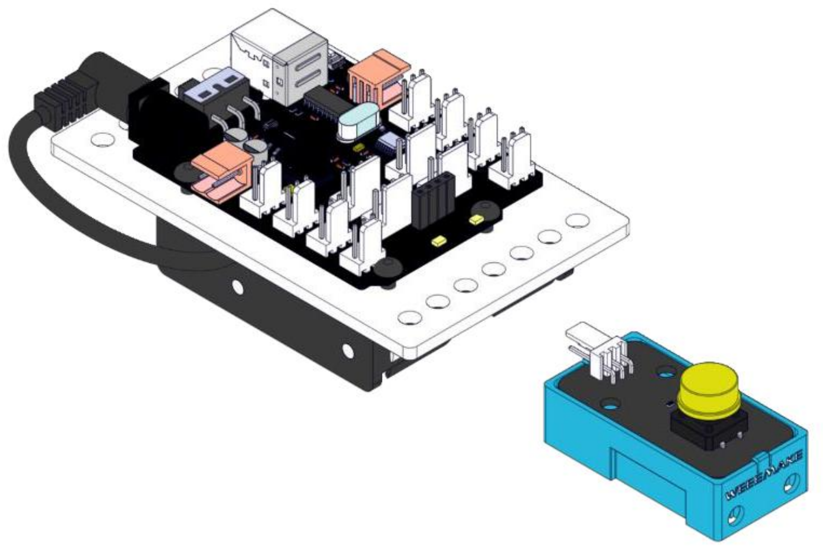
1. First of all, write the program of the bat, the bat hides the ontology every 3 seconds to randomly change places, and then copy a display of the bat, and flapping



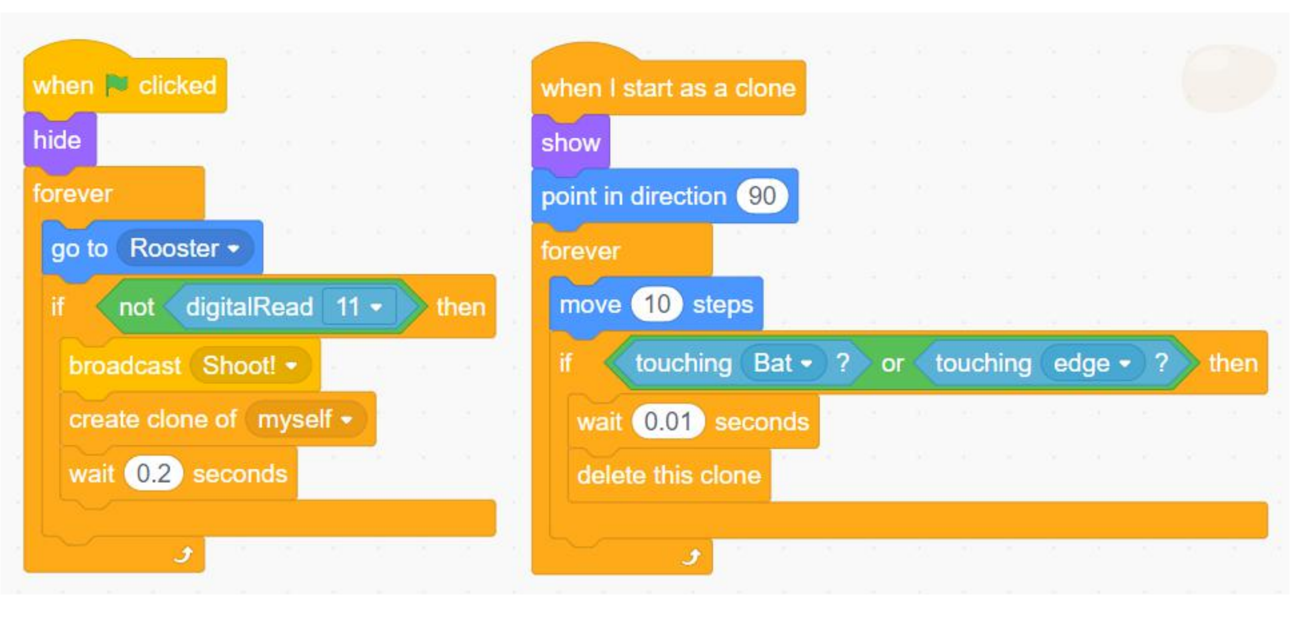
2. Write the program of egg. Egg is the bullet of rooster, and you can fire a bullet through pressing the button sensor connected to the mainboard.
How to write the program?

Add electronic components: buttons, is a dip (or called switch) to control certain functions of mechanical or program. In general, the red button is used to stop a certain function, while the green button can start a certain function. The shape of the button is usually circular or square.

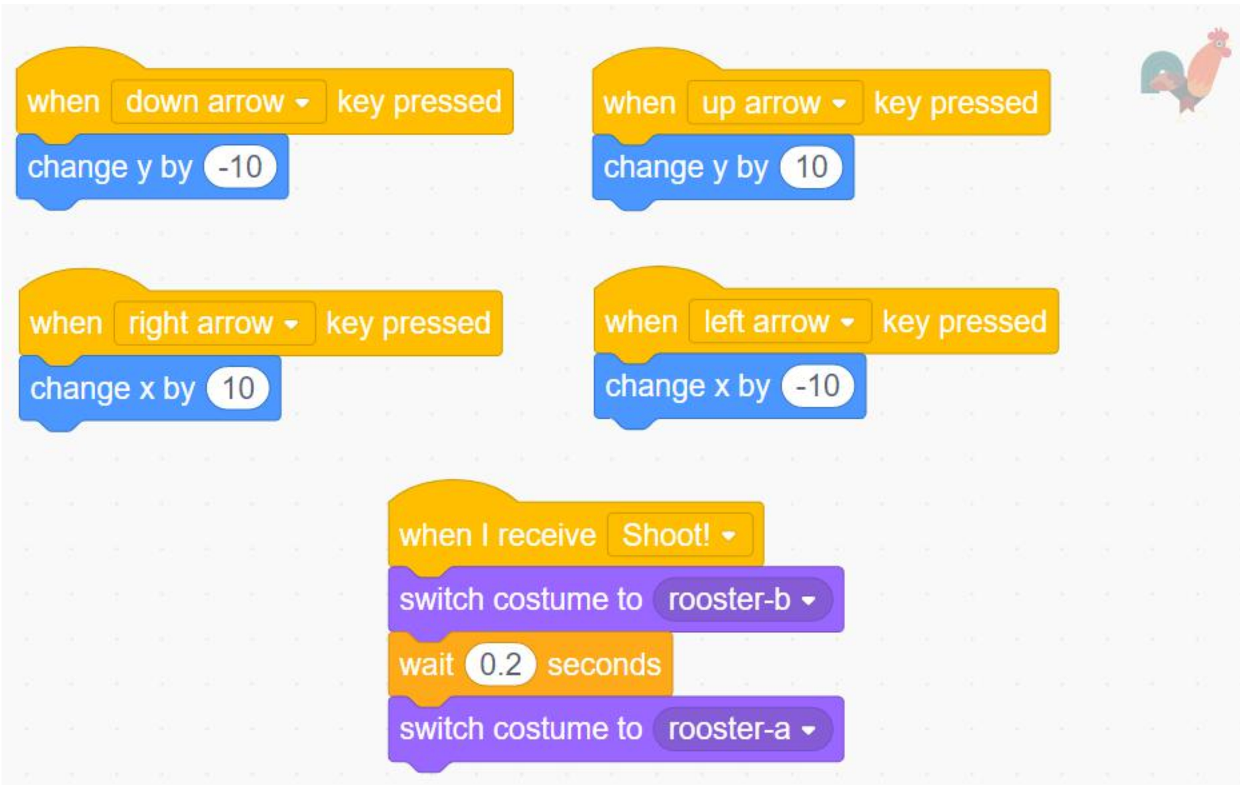
Most of the electronic products are useful to use this most basic human interface tool. With the improvement and innovation of the industrial level, but also the increasingly varied and rich visual effects of the key appearance.



Connect the button to digital port 11 using KF2510 3P terminal wire.



3. Write the program of the rooster. It can move by pressing the up and down keys, the program is as follows:



Classmates,have a try, make your own bat battle.

Can bat battles keep score?

Lesson Three
Ultrasonic Piano

Elephant wants to have a concert in the forest, but is worried about no piano. Little monkey had an idea for elephant, try to use ultrasound to make a piano.

Can you help elephant make a piano? Let's have a try.

Study Task

Task 1: Get to know the sensor.
Task 2: Write online game program.

Practice

1.Make cartoon scenes

In WeeeCode, create a monkey character and place them in the right places.

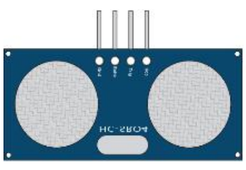
2. Write character programs

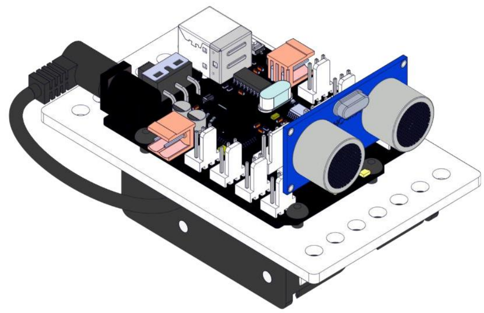
First write the little monkey program, the little monkey will dance at the beginning, the program is as follows:

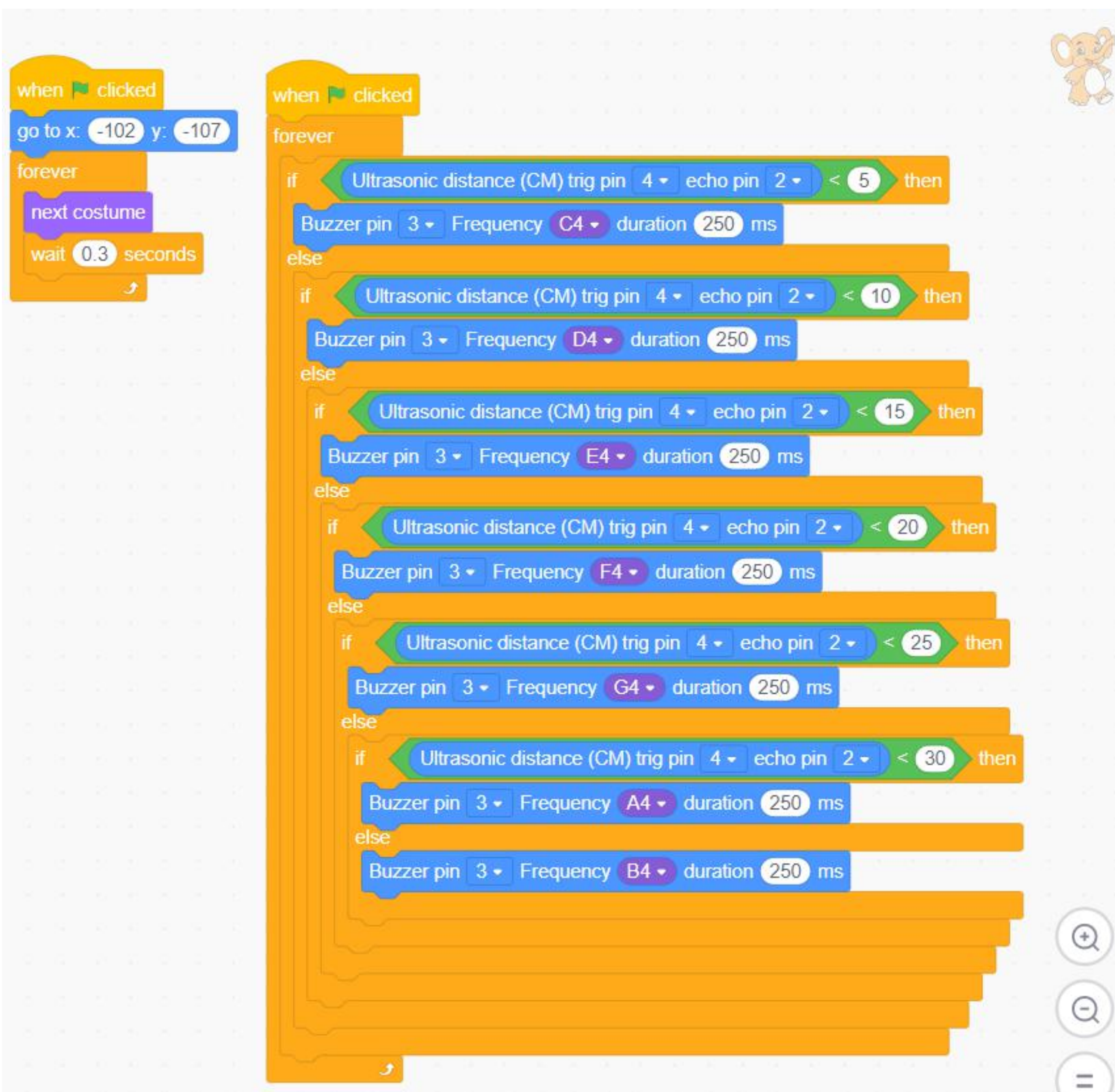
Second: Program for elephant, elephant can dance when begin.

Add electronic components: ultrasonic, ultrasonic wave is a frequency higher than 20,000 Hz (Hz) sound waves, its direction is good, strong reflection ability, easy to obtain a more concentrated sound energy, the propagation distance in the water is much farther than air, can be used for testing Attachment, speed, cleaning, welding, gravel, sterilization, etc. There are many applications in medicine, military, industry, and agriculture. Ultrasound is named by the lower limit of the human auditory limit.

Connect the ultrasound to the mainboard.







Classmates, have a try, make your own ultrasonic piano.

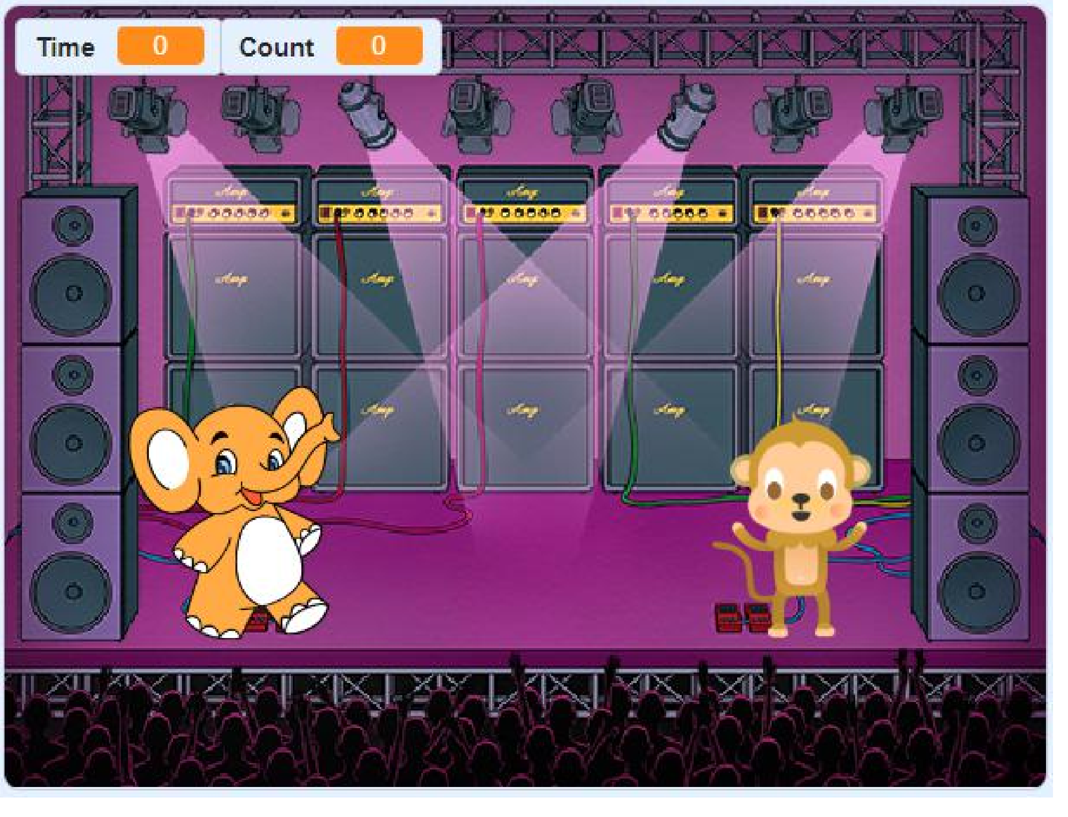
What if the ultrasonic piano can't stop? Do you have a way to optimize the program?

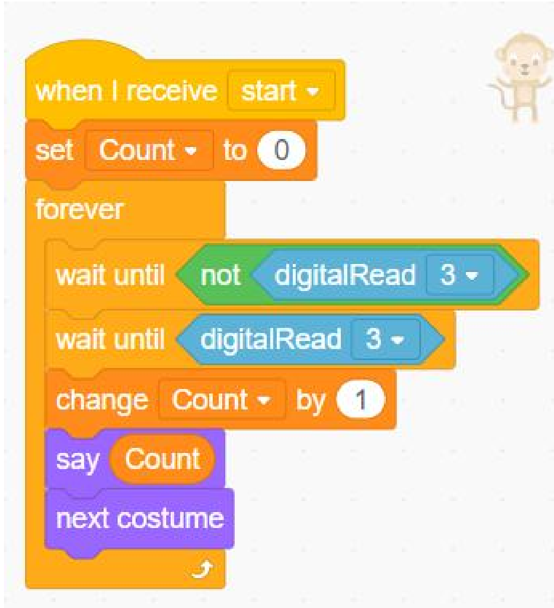
Lesson Four
Speed Challenge

Elephant held a game of hand speed in the forest, and many small animals took part in it. There are rooster, pigeons, dinosaurs, monkeys, and you know who gets the fastest hand title?

You guessed right, the answer is the monkey, because the little monkey is very lively and active, often climb tree with hands and fingers are flexible.

Do you know how elephant hand speed measuring device was made? Let's have a try.

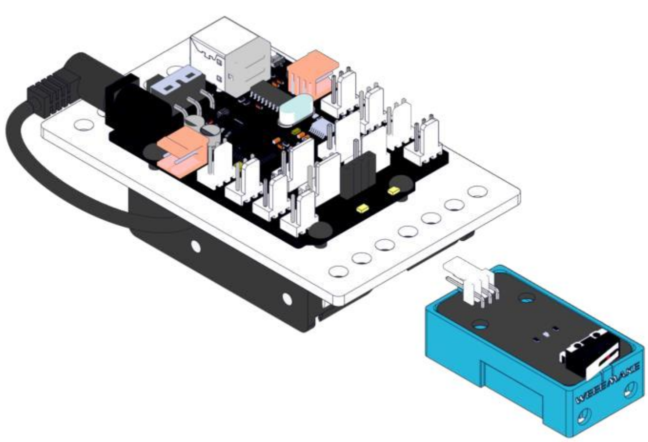


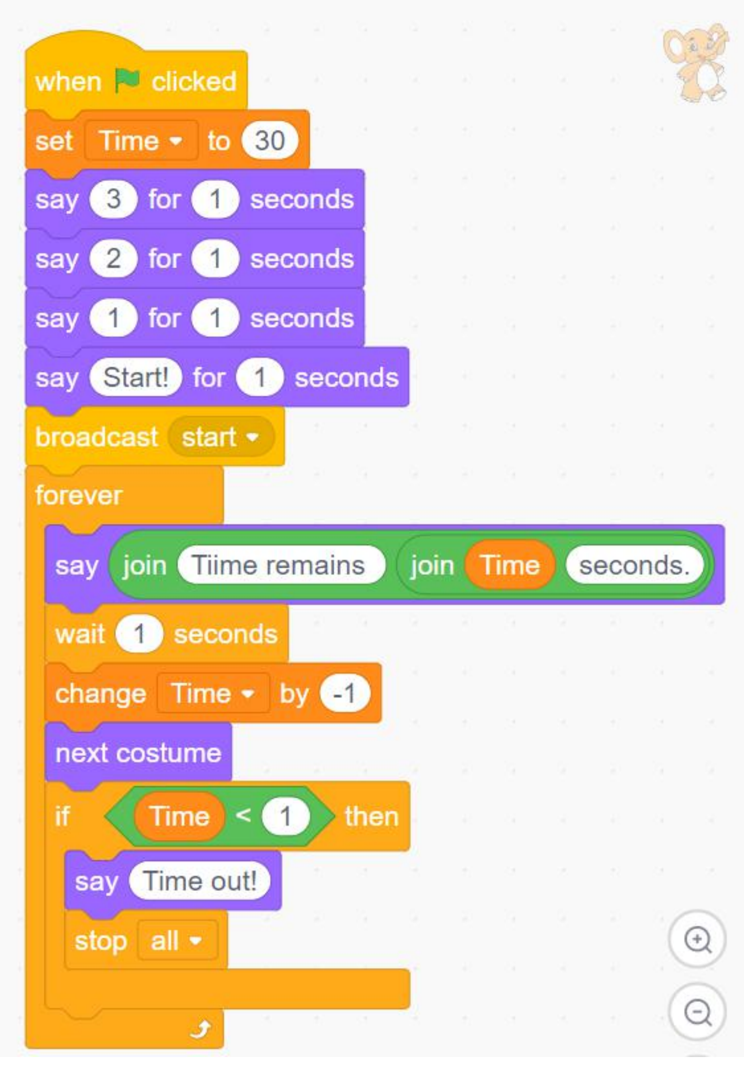


Second, write elephant program. Elephant will countdown when begin, and broadcast the start, so let's start.

New electronic components: limit switch (collision sensor), limit switch module is a physical switch, can play the role of limit anti collision, when triggered, it will output a level signal to the control end. Limit switch is digital sensor.

Wiring description: Use KF2510 3P terminal wire to connect the limit switch sensor to digital port 3.





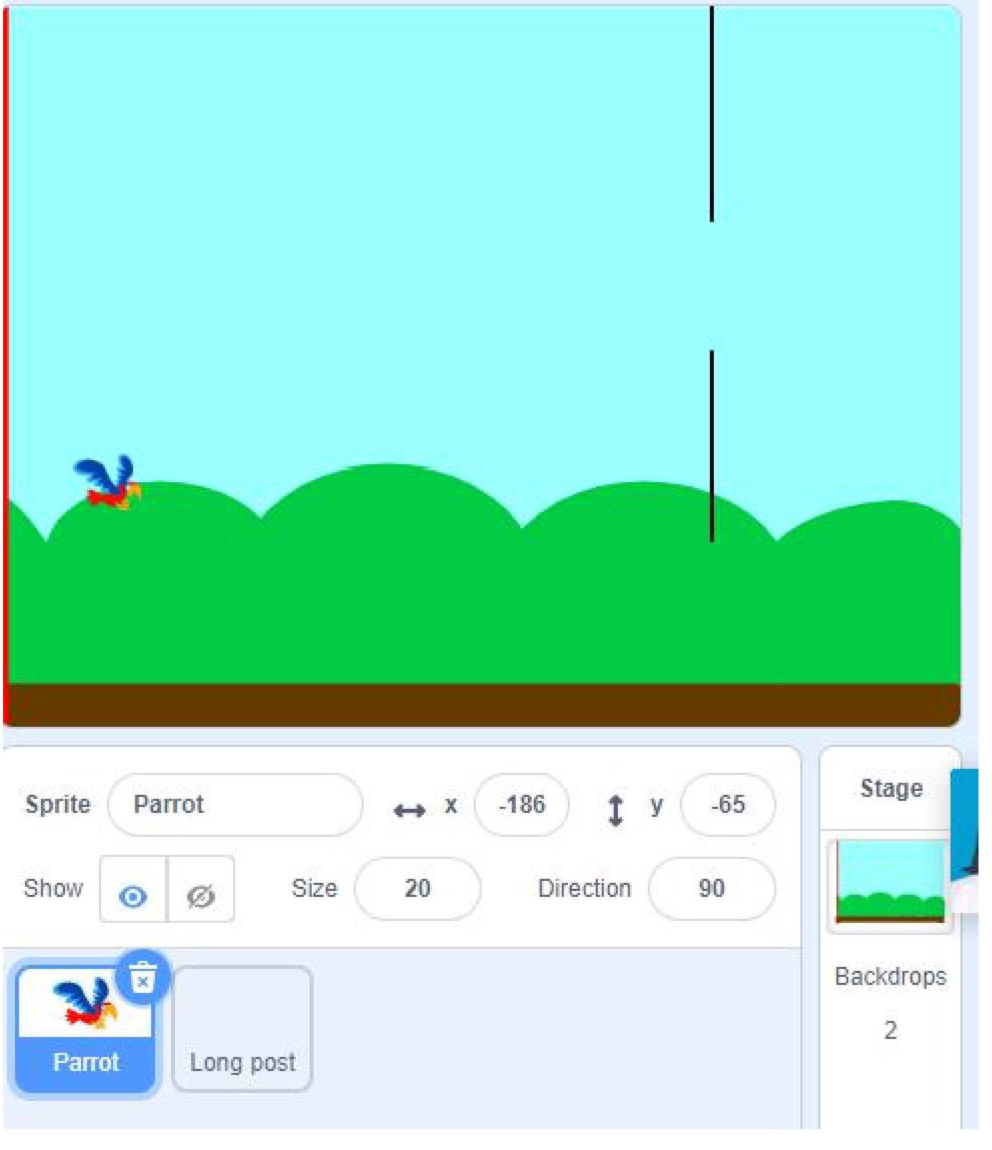
Okay, after learning, let’s have a try, make your own speed challenge.

How do you make a leaderboard? Students can try to make it.

Lesson Five
Flying Bird

Elephant's best friend, the parrot, has been getting sleepy lately. It is very dangerous for the parrot to fall asleep when it is flying. We have to remind the little parrot to prevent the little parrot fell asleep from hitting the pole.

Can you help the little parrot? Let's have a try.



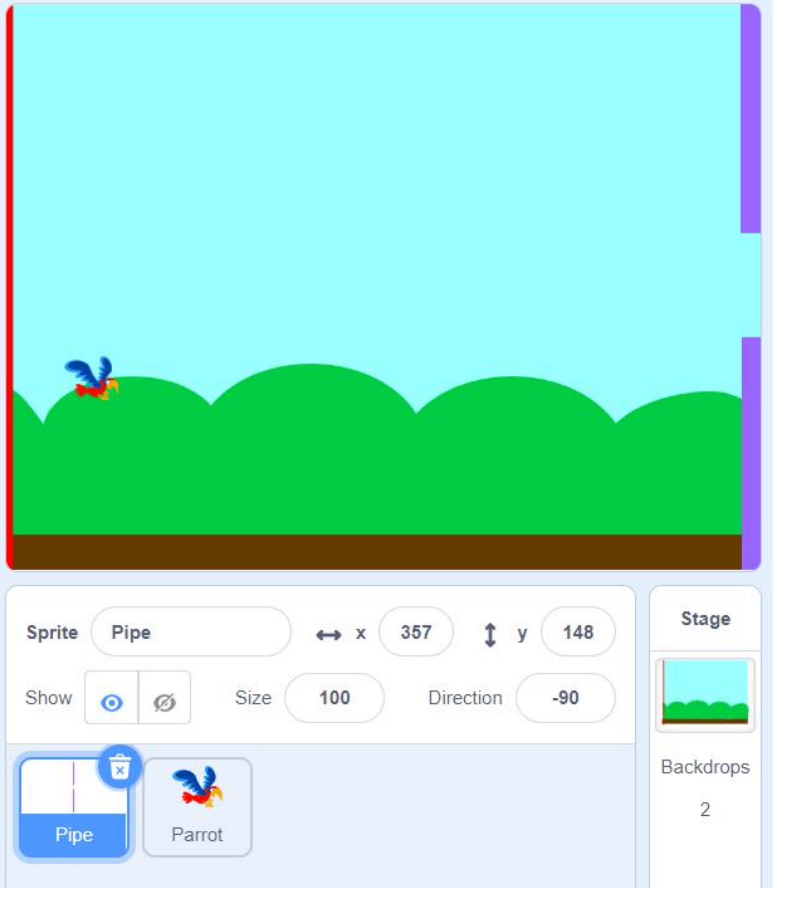
Study Task

Task 1: Get to know the sensor
Task 2: Write online game program

Practice

Making cartoon scenes

In WeeeCode, create a character of parrot and place it in the appropriate position. Draw a new role, draw a long pipe, and erase a gap in the middle of the pipe. The gap can not be too large or too small to affecting the bird fly through. The pipe should be very long, neither the gap is at the top or bottom should not reveal the end of the pipe; Choose the right stage background and draw a red line on the far left of the background.

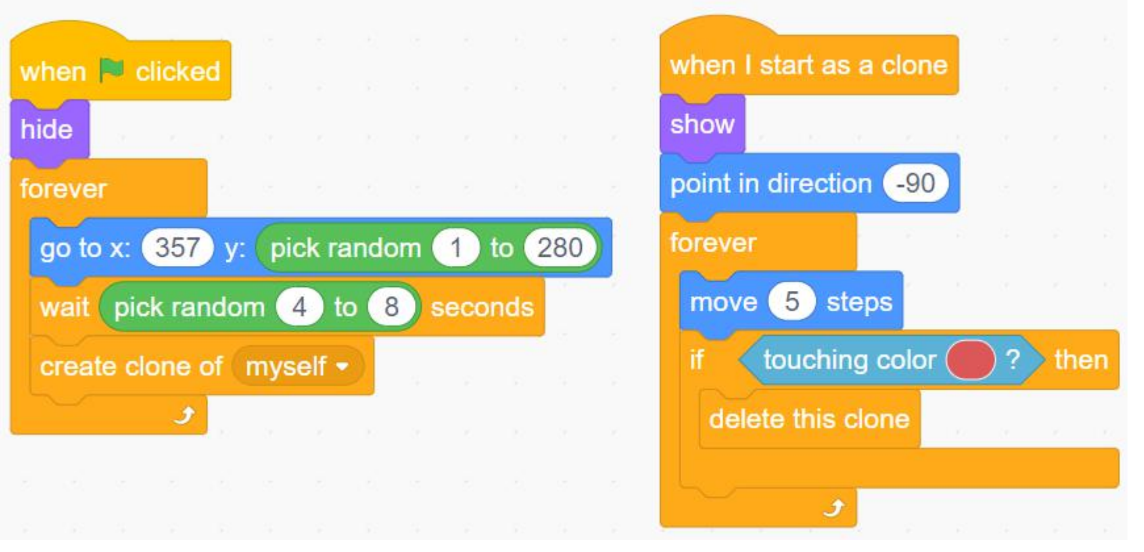


Writing character programs

1. Write the program of the draw pipe role.

Before writing the program, the column should be placed on the far left of the stage, and know the Y-axis coordinates of the top and bottom notch, and use random numbers to make the post move up and down randomly.

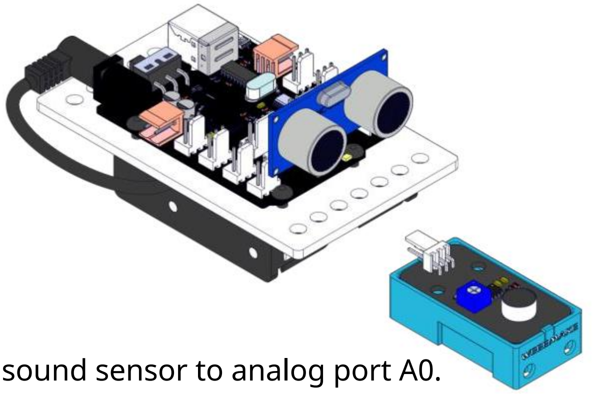
Procedure is as follows



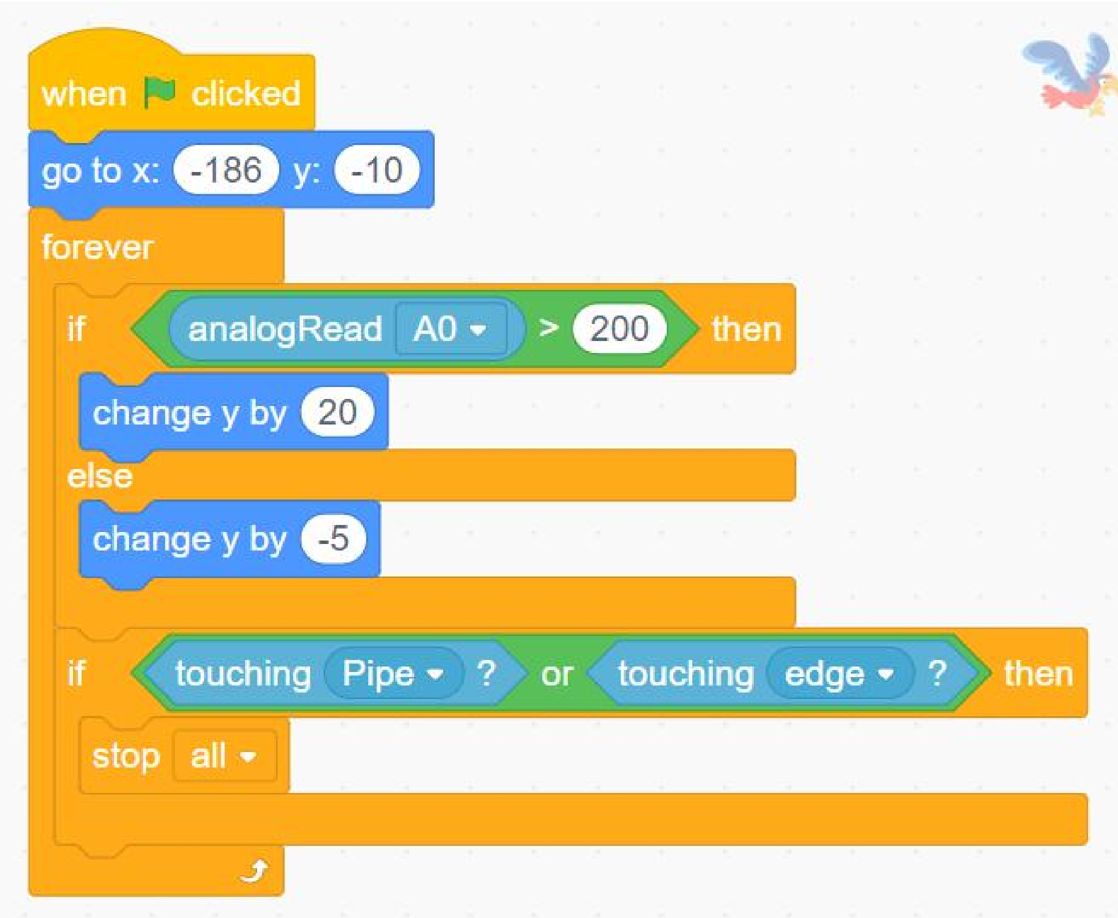
2. Write the program of the parrot. The parrot will keep flapping its wings at the beginning, but it keeps falling because it is too sleepy. When we call it with the sound sensor, it flies upwards, and if not, it will fall, and when it hits the post or falls to the ground, the game is over,

New electronic component: sound sensor, it's regarded as a microphone . It is used to receive sound waves and display vibration images of sound, but it cannot measure the intensity of noise.

The sensor has a capacitive electret microphone that is sensitive to sound. Sound waves cause the electret film inside the microphone to vibrate, causing a change in capacitance that generates a tiny voltage corresponding to that change. This voltage is then converted into voltage of 0-5V, which is received by the data collector after A/D conversion and transmitted to the computer.



Use KF2510 3P terminal wire to connect the sound sensor to analog port A0.



Okay, after learning, let's have a try, make your own flying bird.

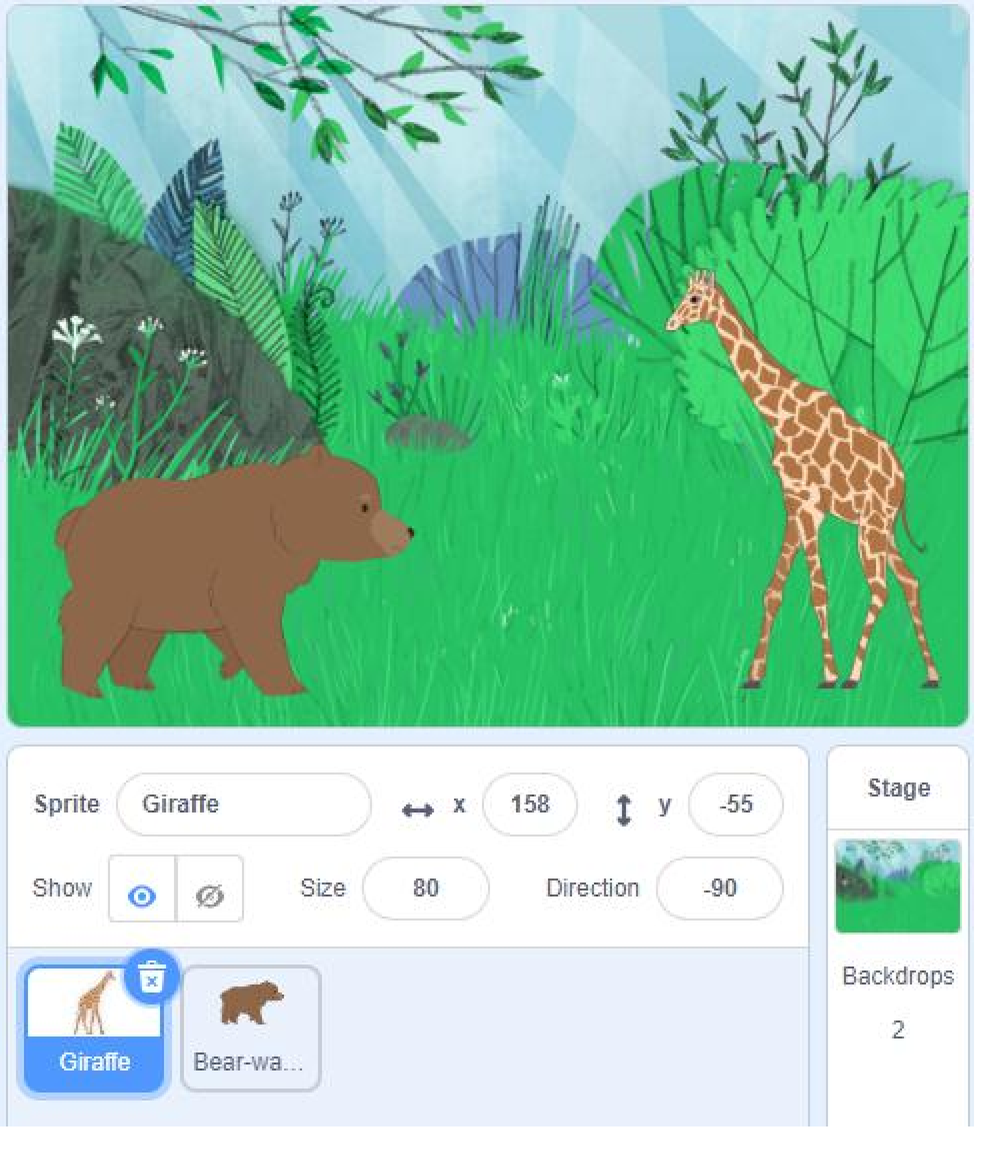
Can you try programming adding score system? Have a try!

Lesson Six

Day and Night

Elephant and many animals in the forest are good friends, they are busy in the daytime every day, and at night will go back to their own small room to sleep, can you help elephant make such a wonderful room?

Let's have a try.



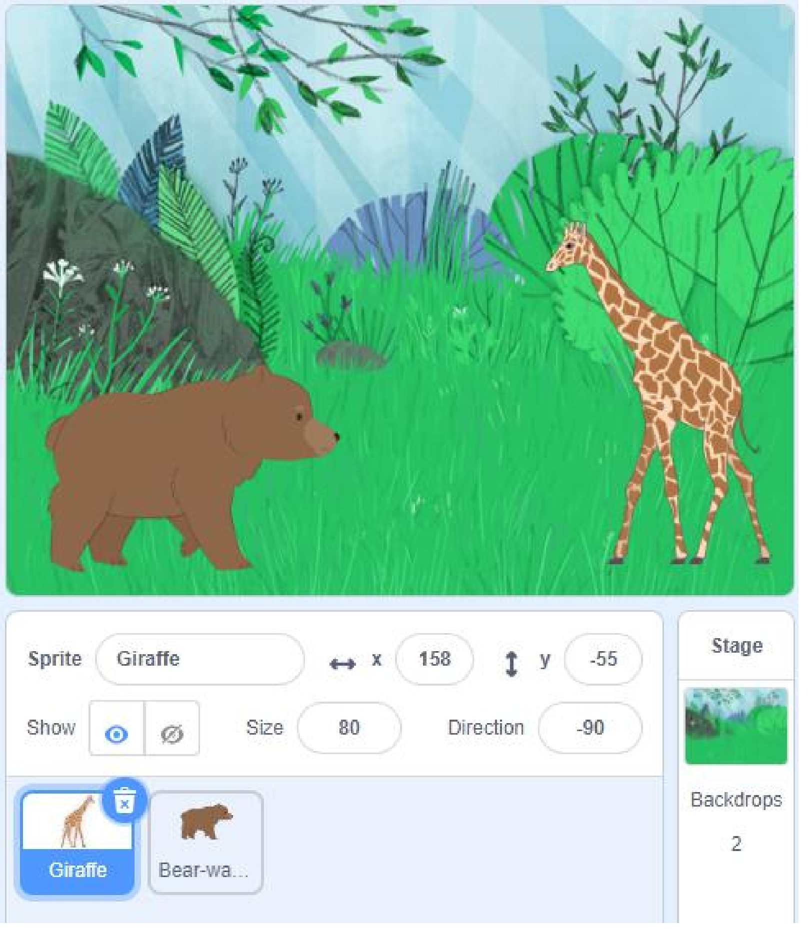
Study Task

Task 1: Get to know the sensor.
Task 2: Write online game program.

Practice

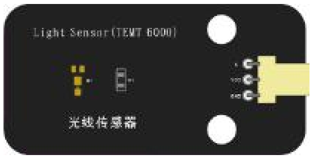
1. Making cartoon scenes

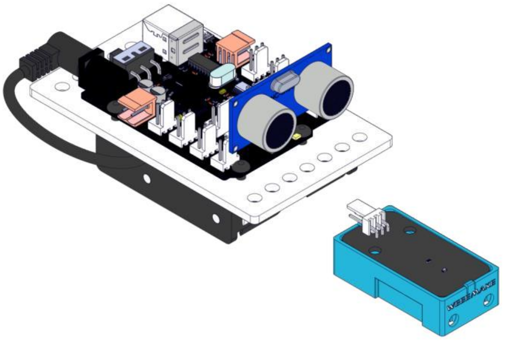
In WeeeCode, create sprite of giraffe and bear and place them in the right places, reduce the size and make it face down.



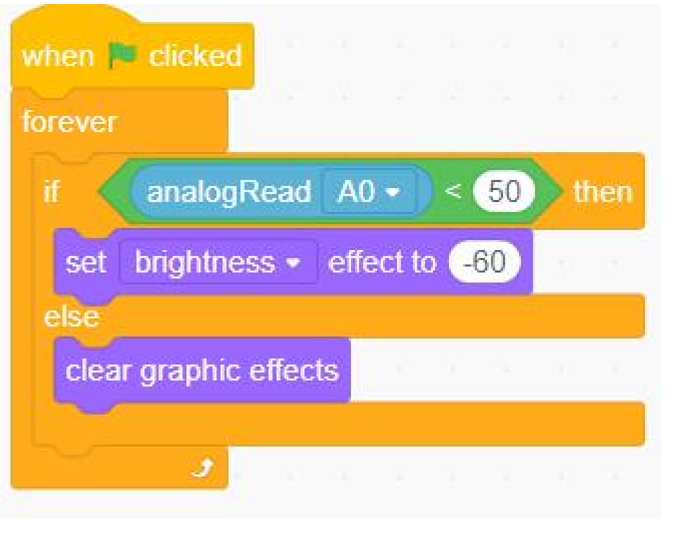
Second, Write stage programs, we need to adjust stage brightness , play background sound music, and program game stop control .

New electronics: Light sensors, also known as brightness sensors, are used on tablets and phones. Generally located above the screen of the handheld device, it can automatically adjust the screen brightness of the handheld device according to the light brightness of the handheld device, to bring the best visual effect to the user.





Wiring description: Use KF2510 3P terminal wire to connect the light sensor to analog port A0.



Program for animals, they go out for a walk during the day and trot home at night. So at night, let the animals walk and hide when they touch the edge of the stage. Can you write a program?

Okay, Classmates, aftering learning lesson, can you write a program game for animails back home when at night? Hav a try!

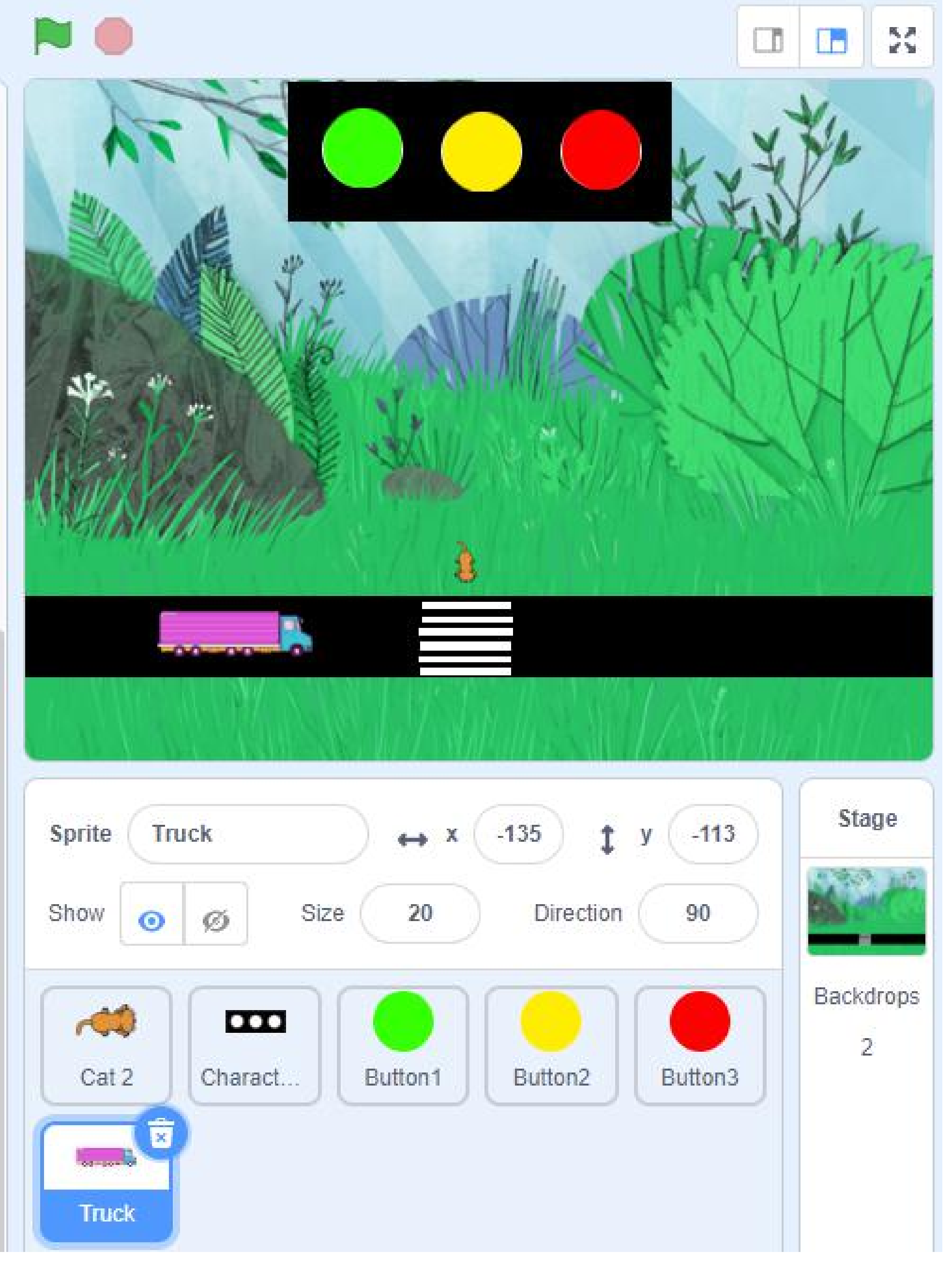
Can you try adding more animals to play together? Have a try.

Lesson Seven

Follow Traffic Rules

A motor-vehicle road recently opened in the forest where Jingle elephant lived. Many small animals had some traffic accidents because they did not understand the traffic rules. In order to prevent more traffic accidents, elephant decided to teach the forest animals about traffic. Do you know when it is okay to cross the road?

Let's have a try.

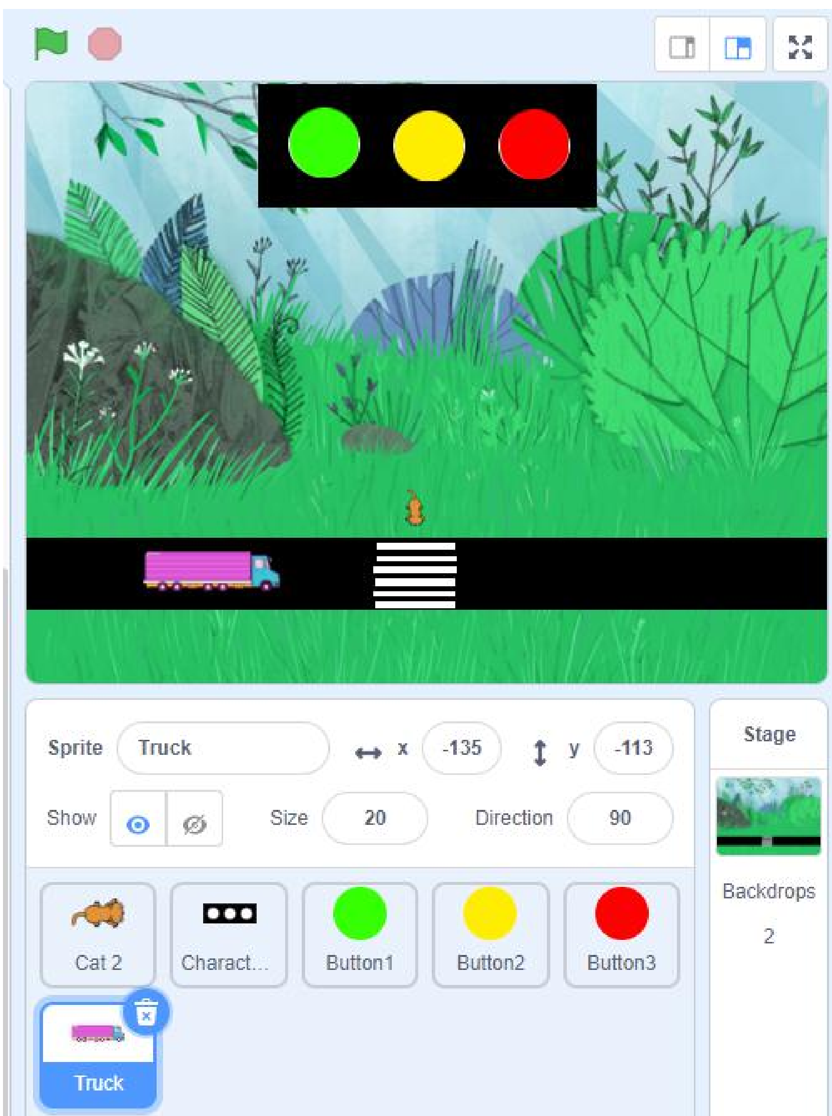


Study Task

Task 1: Get to know the sensor.
Task 2: Write online game program.

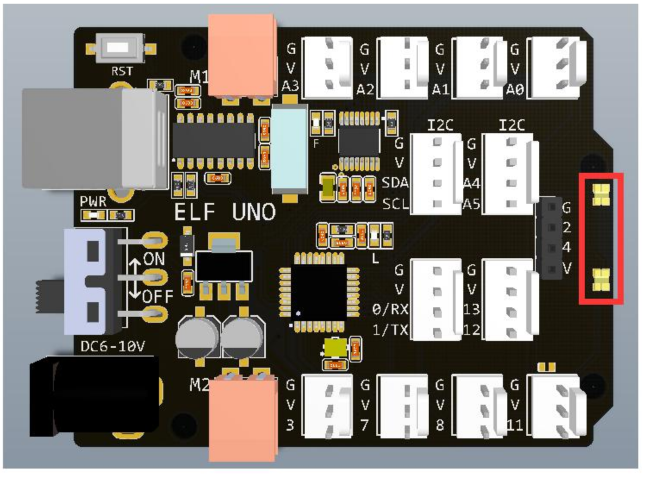
Practice

In WeeeCode software, new build character of cat, bus, green light, yellow light, red light and traffic light frame; New build forest stage, draw a road on the stage, draw a zebra crossing on the road; Reduce each character to the appropriate size and place them in the appropriate position on the stage.

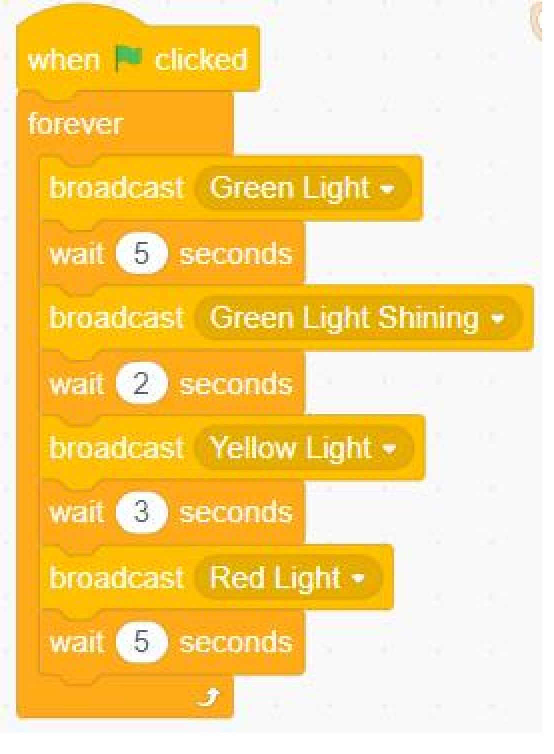


New electronic components: onboard RGB light, RGB light is the common intersection of three primary colors imaging. In addition, there are also blue light LED with yellow, and ULTRAVIOLET LED with RGB phosphor,

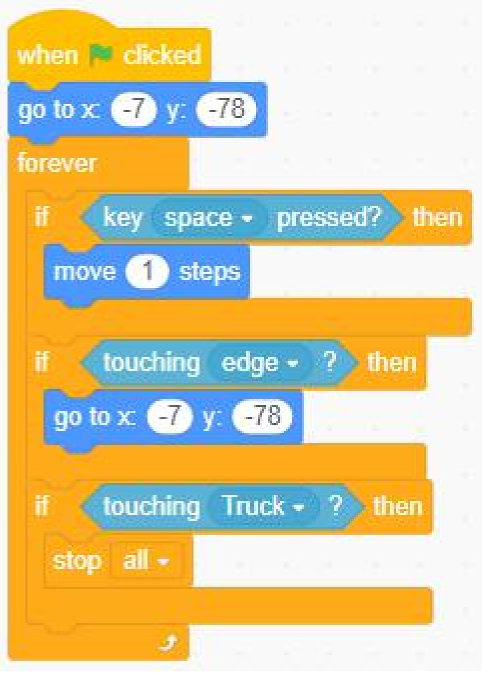
In general,they have their imaging principle. Some LED backlight board color is particularly clear and bright, and even with high quality degree of TV clearance, it is the RGB characteristics, advertised red is red, green is green, blue is blue ,in the light of the mixed color, with more diverse characteristics.



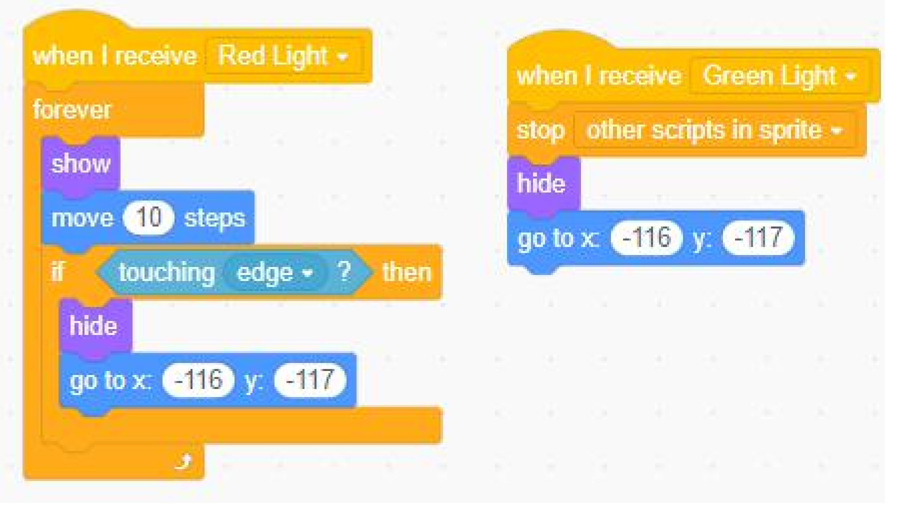
Program as follow:



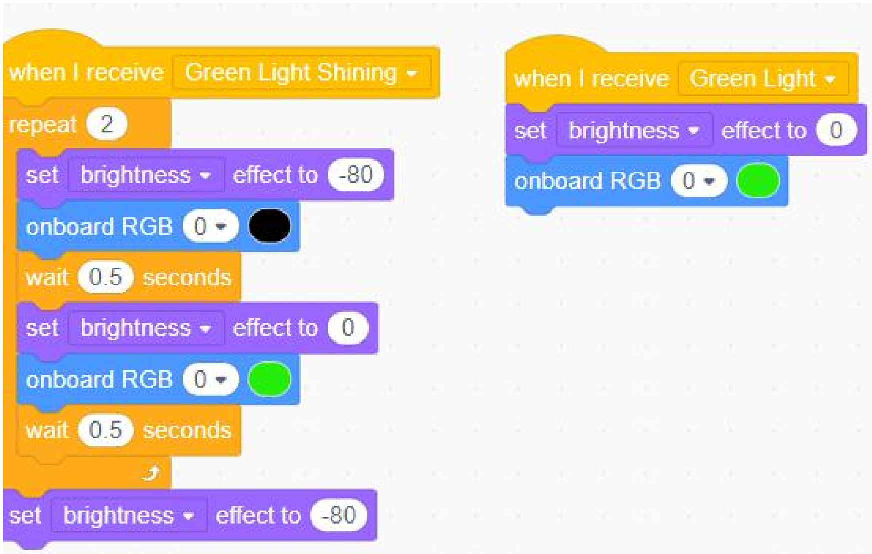
Write the cat program, the cat move by pressing the space bar , if hit the car, the game is over, if tough the edge, back to the starting point to continue to move, the program is as follows:

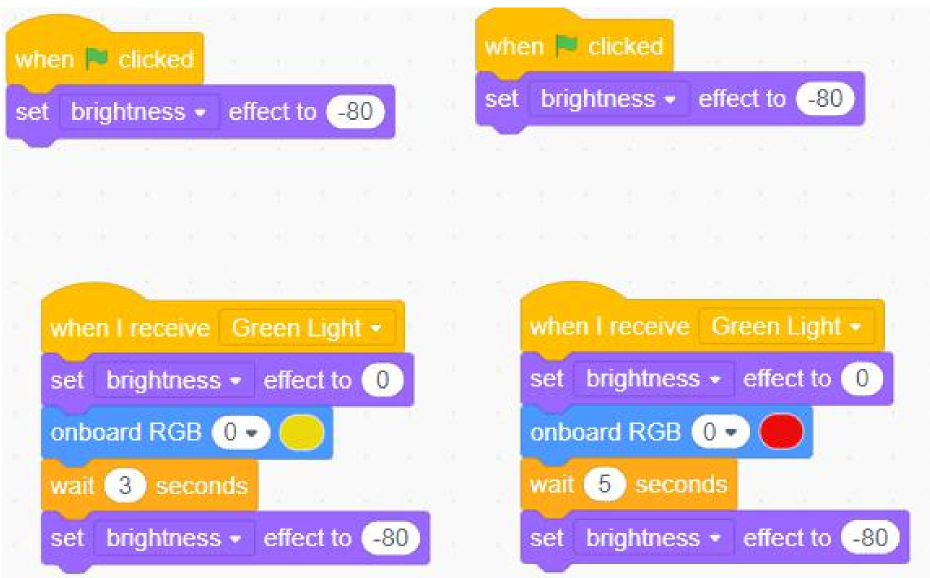


Write the program for car, the car start to move in the zebra when red light, if tough the edge, then back to the starting point to continue to move; If the light is green, hide and return to the starting point, as follows:



Create three new roles of button or draw three circles and color them red, green and yellow. Write a program according to the traffic lights rules in reality. The program is as follows:





Classmates, aftering learing the lessons, can you make a game of follwing traffice rules? Have a try.

Can you add points for crossing the road? How do you make it fun? Have a try!

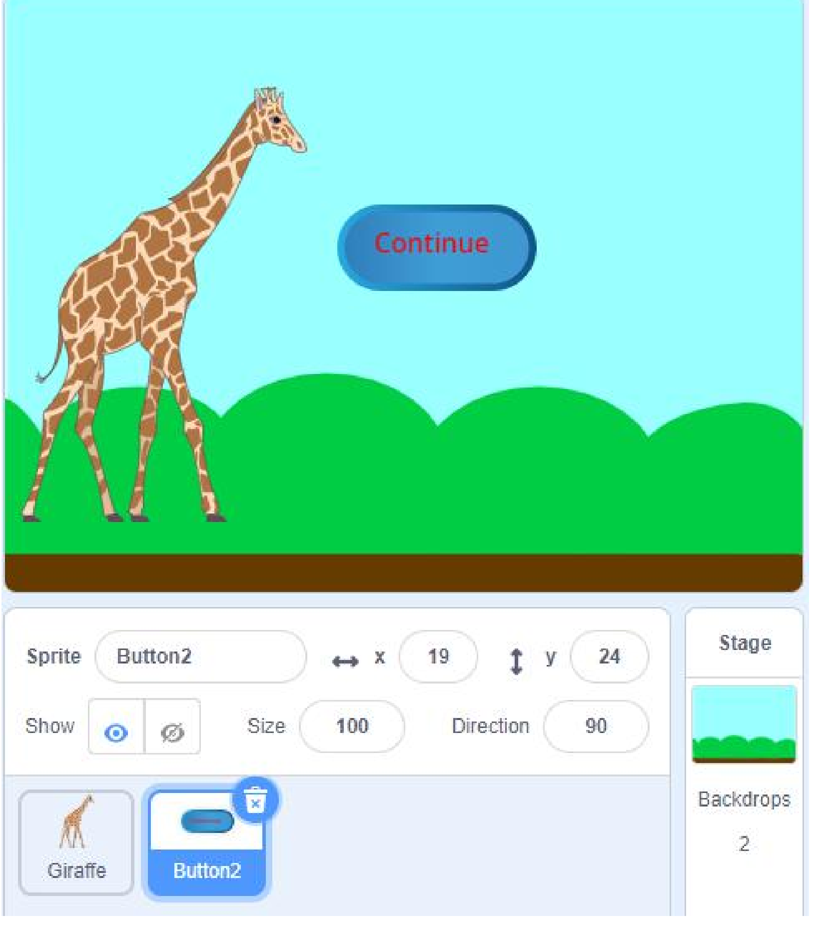
Lesson Eight

Height Measurement

Elephant's best friend giraffe is always confident of his height, so he wants to organize the small animals in the forest to measure height, but jingle elephant disagrees. He thinks that a tall height is not represent health, as long as its BMI is within the healthy range, it is healthy. Can you help a giraffe figure out its BMI?

A measure of thinness and health.
BMI= weight divided by height squared. (Unit of weight: kg; Height unit: meter.)

Let's have a try.



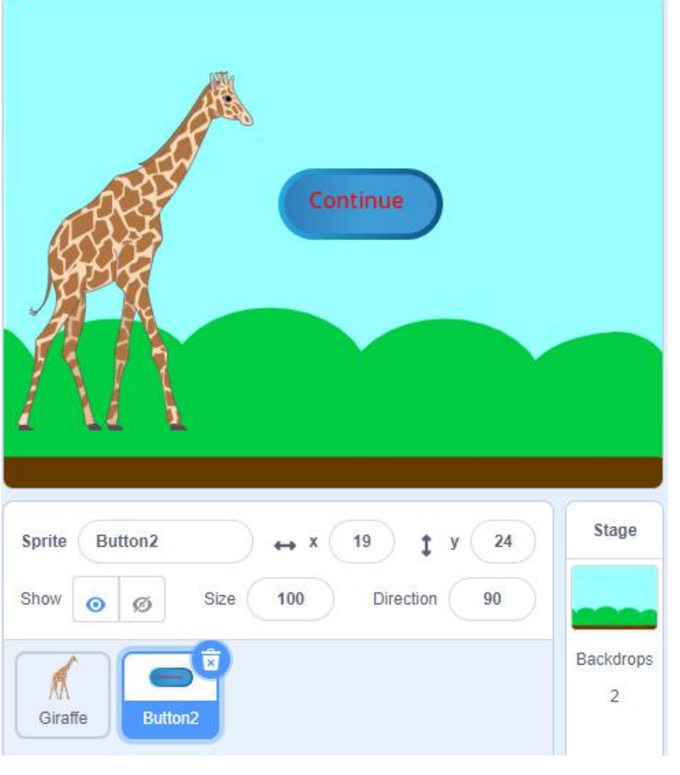
Study Task

Task 1: Get to know the sensor.
Task 2: Write online game program.

Practice

1. Making carton scenes

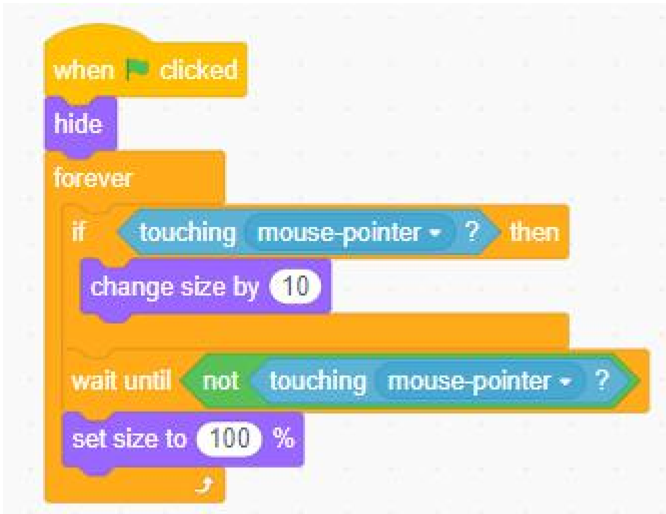
In WeeeCode software, create roles of giraffe and button , button is added the text "continue" in the modeling interface.



2. Write programs

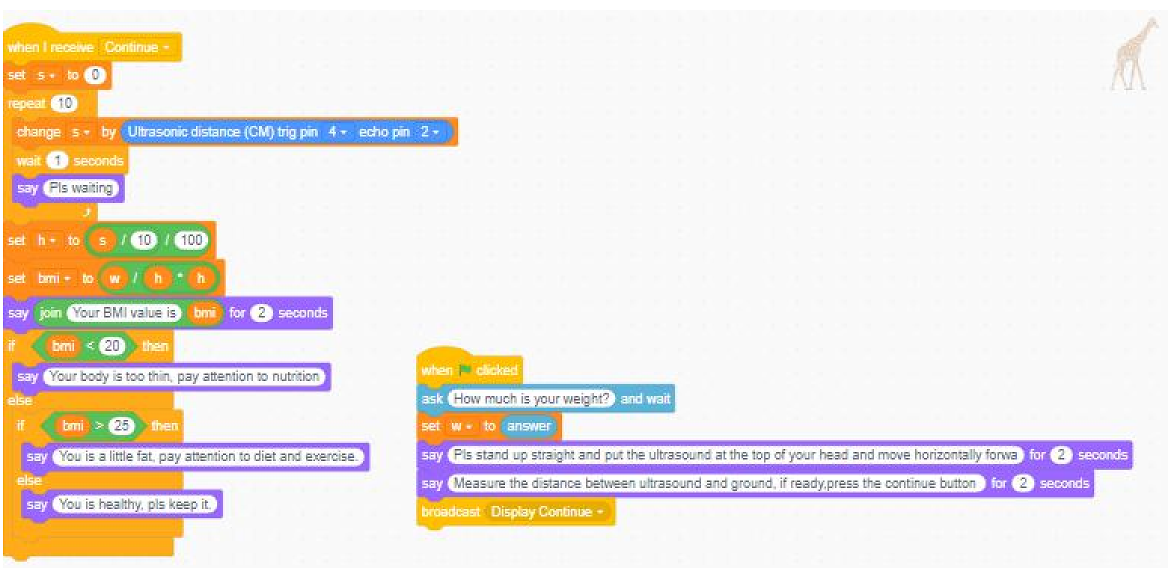
First write the button program, the button is a intermediate link which is responsible for the program to continue , hidden at the beginning, display when the broadcast is received, hide and send the broadcast after being clicked. It is appropriate to make some special effects.

Program as follows:



Write the program for giraffe, click Start button first and then input the user's weight; Then the measured user is prompted to stand in a normal posture so that height can be measured;

Then broadcast the display button. When the button is clicked, measure the value of ten ultrasonic waves and take the average value.Since the value obtained by ultrasonic waves is cm, use formula change its value into meters, calculate BMI and output the body condition according to the range.



Okay, after learning the lessons,you can program to test your healthy conditions, have a try!

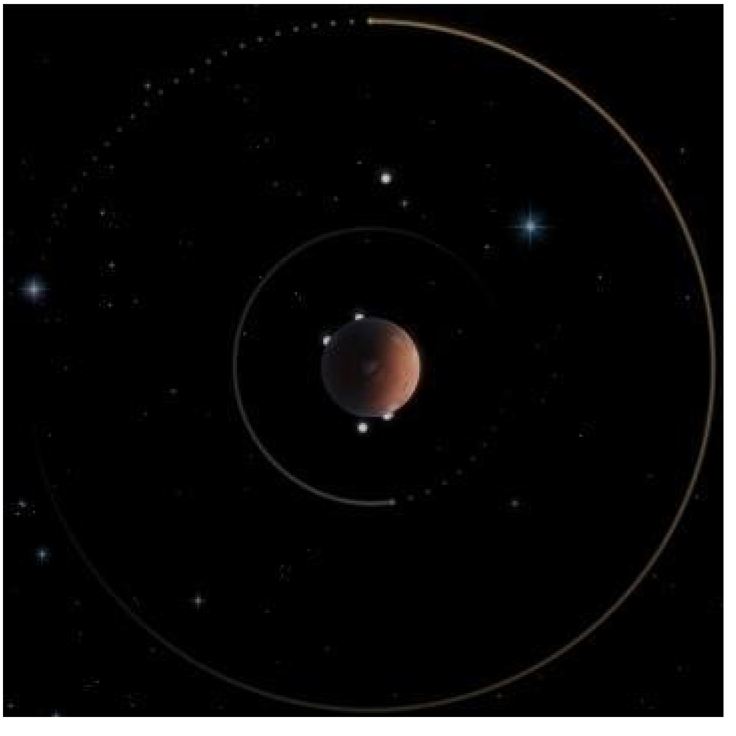
How to use ultrasound to measure height more accurately? Have a try!

Lesson Nine

Learn about Mars - Make a Rover

Speaking of "Mars" , we are strange and curious, Mars rover - only can see the amazing robot in science fiction movies, but what is a rover? What is a robot? How do we make a rover? In this lesson,

Let's learn about Mars and rover, see how mysterious it is.



Study Task

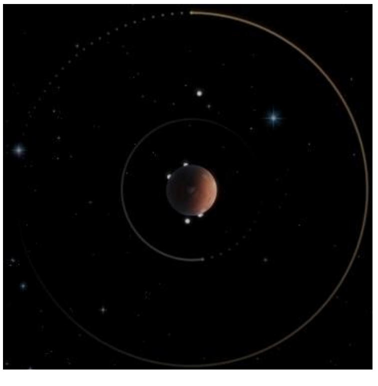
Task 1: Learn about Mars and the difference between Mars and Earth.
Task 2: Get to know the components, electronics and specialized programming software of rovers.
Task 3: learn machine component assembly connection .

Knowledge Point

1.Know about Mars.

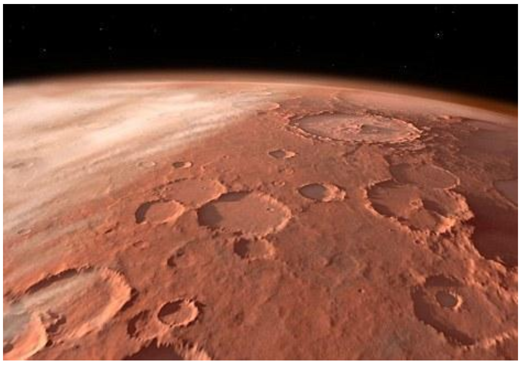
Mars is the fourth closest planet to the sun and the second smallest planet in the solar system after Mercury. It is one of the four Earth-like planets in the solar system.

Mars's reddish orange appearance comes from the fact that its surface is covered with hematite (iron oxide),
Mars is about half the diameter of Earth, with a similar inclination on its axis and rotation period, but twice as long as Earth.



Mars's carbon-dominated atmosphere is thin and cold, with impact craters, canyons, dunes and gravel on the surface and no stable liquid water, southern hemisphere is with ancient crater-filled highlands while younger lowland plains in the northern.

Mars has the largest known mountain in the solar system "Olympus Mons",and the largest canyon "Mariner canyon". Mars has two satellite , Phobos and Deimos, irregularly shaped asteroids that Mars may have captured. Based on observational evidence, Mars has been observed to resemble groundwater gusher, the Antarctic ice cap has partially retreated, and radar data shows large amounts of water ice beneath the surface at the poles and mid-latitudes.




Mars has four probes in orbit from NASA and Europe,"Odyssey", "Mars Express", "Mars Global Surveyor" and "MAVEN" ,and muitiple American rovers on the surface of the Mars,"curiosity"," Insight", as well as the mission-ending "Mars Pathfinder", "Phoenix", "Spirit" and "Opportunity" etc.

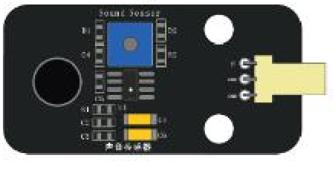


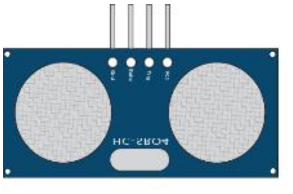
On May 15, 2021,China's space research team confirmed that the "Tianyun-1" landing rover landed in the pre-selected landing zone based on telemetry signals sent back by the "Zhurong rover", marking the country's first Mars exploration mission a complete success.

2.How are rovers made?

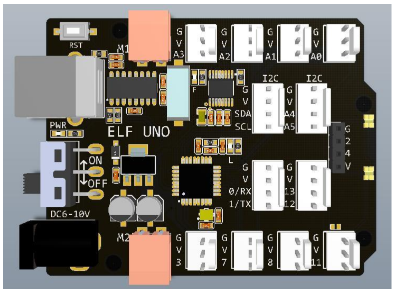
Supporting parts: structures used to support vehicles, such as the trunk of the human body
and the bottom plate of an automobile.

Sensing part： They are equivalent to human eyes, ears, skin and other sensory organs, which can detect the situation of the rover in real time and adjust the mechanical part to ensure that the rover's movements meet the requirements.



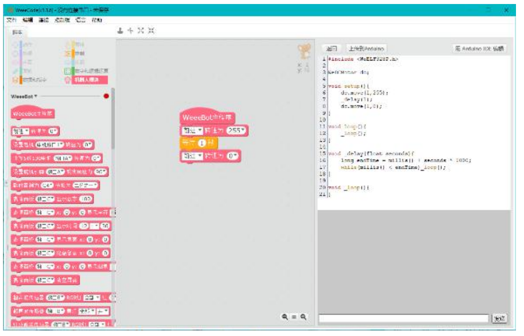


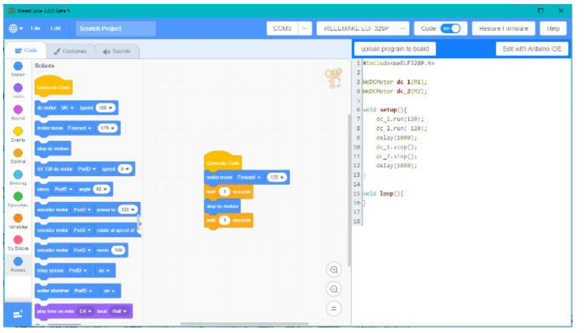
Control part: equivalent to the human brain, all the control of the robot is completed by one or more microcomputers, divided into centralized control and decentralized (level) control.



3. Program software

WeeeCode is a graphic programming software independently developed by Shenzhen TECH CO. LTD, which is suitable for primary and middle school robot learning. It is based on MIT's Scratch and Arduino platforms for open source graphic programming. WeeeCode 2.0 development is based on Scratch2.0, and WeeeCode 3.0 development is based on the latest Scratch3.0. There is no essential difference in basic functionality between the two versions and learning from 2.0 will not affect 3.0 application .

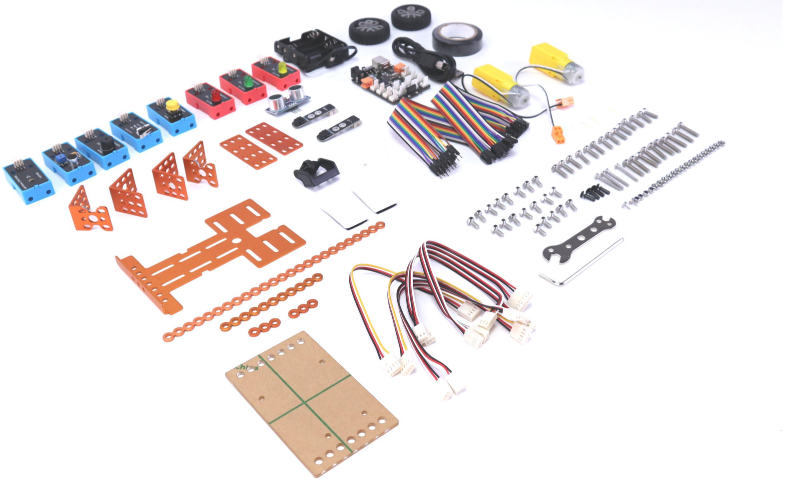




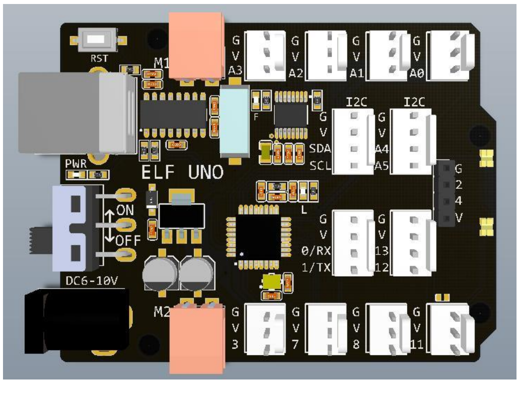
Practice

1.Know the components of the "rover".

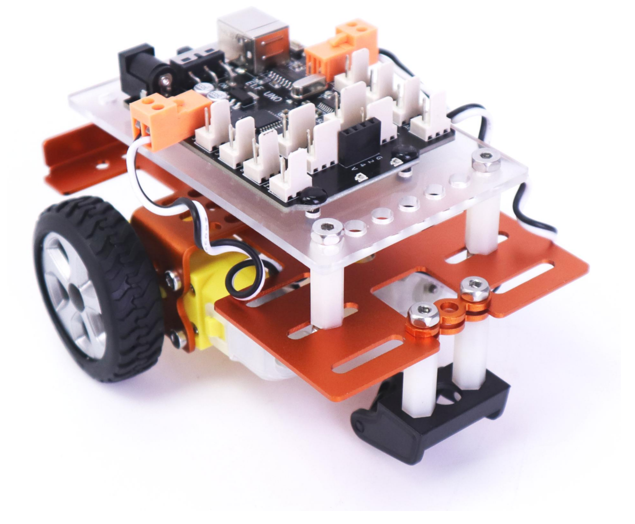
Get the "Arduino Educational Robot Kit" hardware kit, classmates will see various metal structures in different shapes, including the base plate, bracket 3x3, link 0208-24, injection molded universal wheel, motor, wheel and battery case, as shown below.



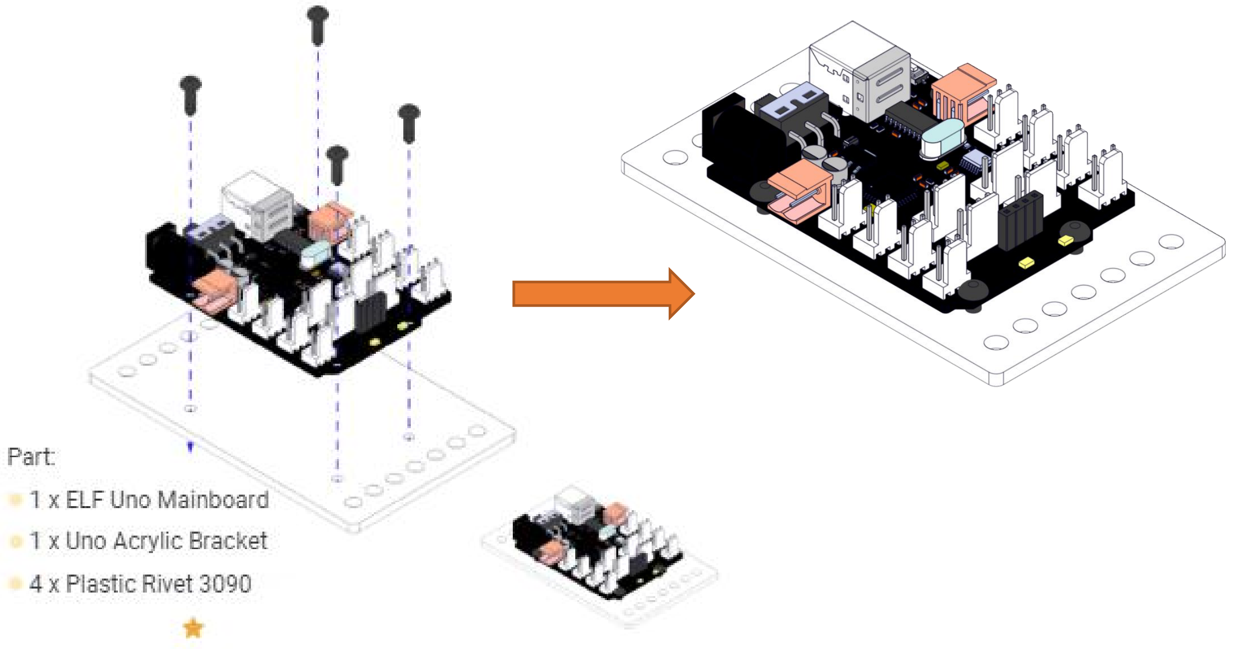
The Mars rover kit contains a variety of electronic components, but this lesson uses only the mainboard, as shown below.



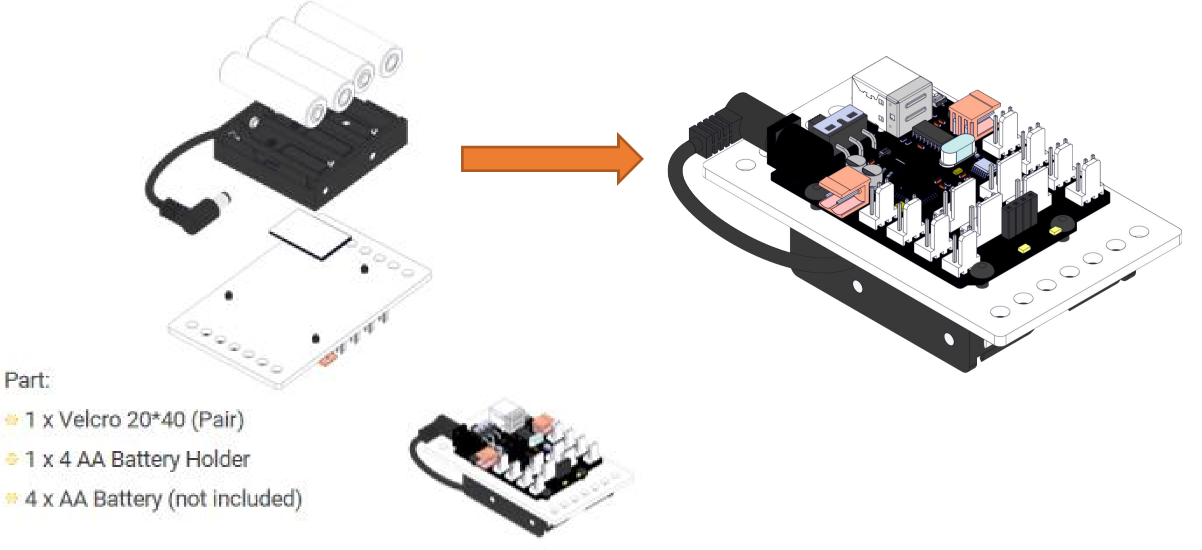
Assemble the structural parts and electronic components, and finally assemble the Mars Rover, as shown below.



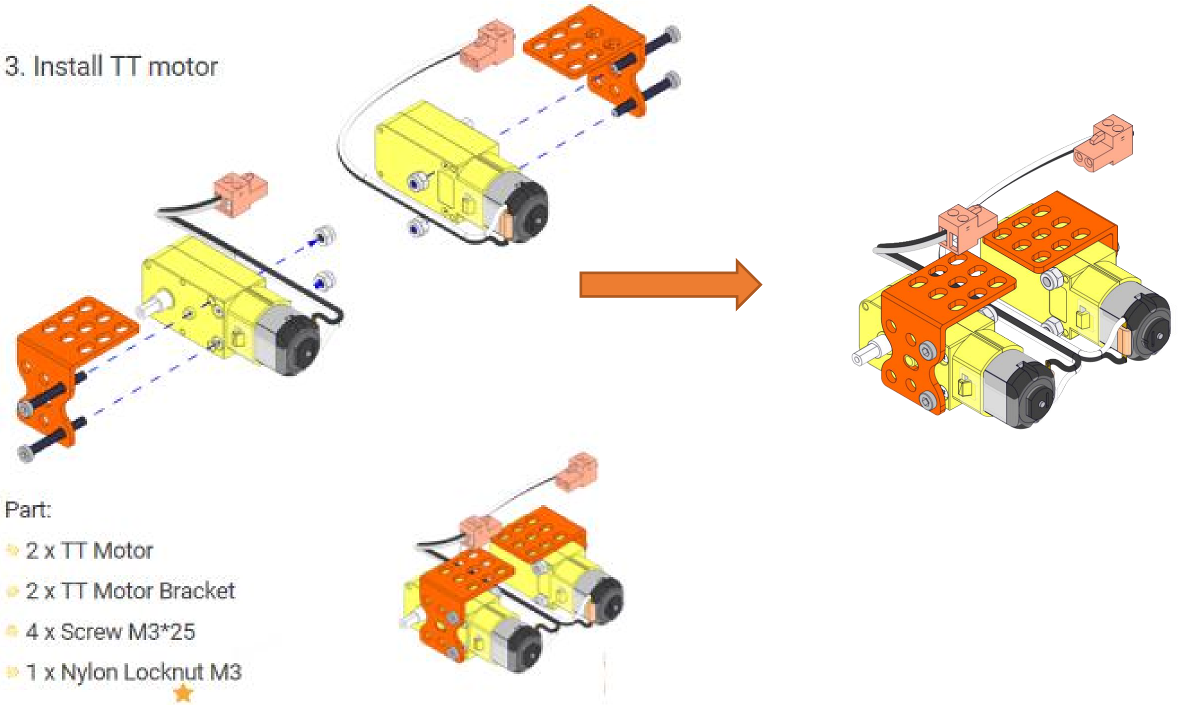
2.Assembly and connection of structural parts.
First, install ELF Uno mainboard.



Second, install the battery case and batteries.

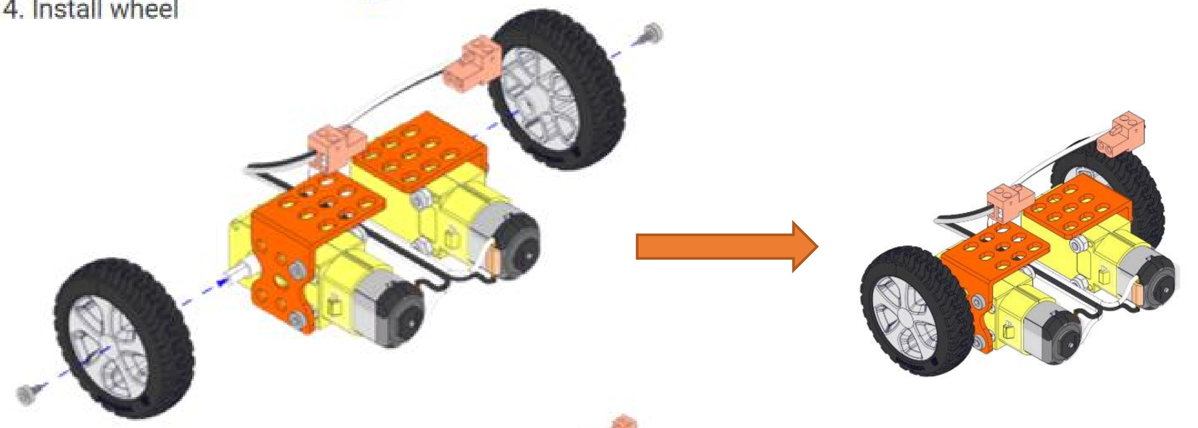


Third.Install the TT motor bracket.



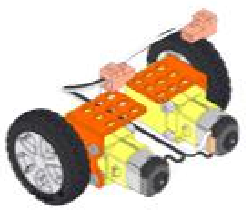
Fourth,install the wheel.

4. Install wheel



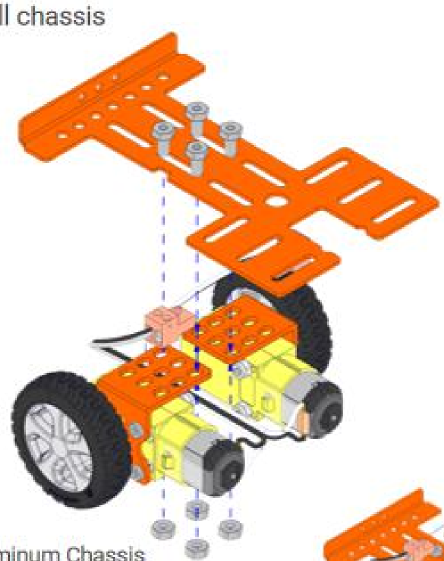
Part:

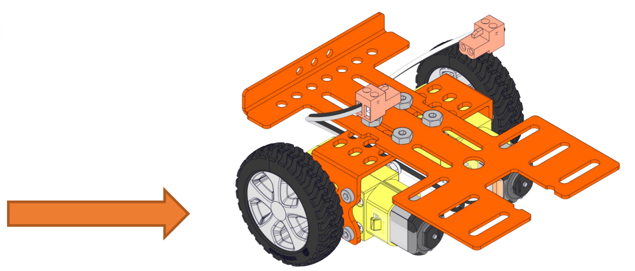
- 2 x Wheel
- 2 x Screw M2*6



Fifth,fix the motor assembly on the bottom plate.

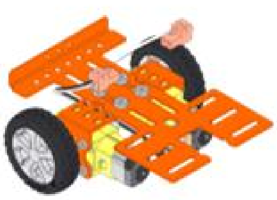
5. Install chassis



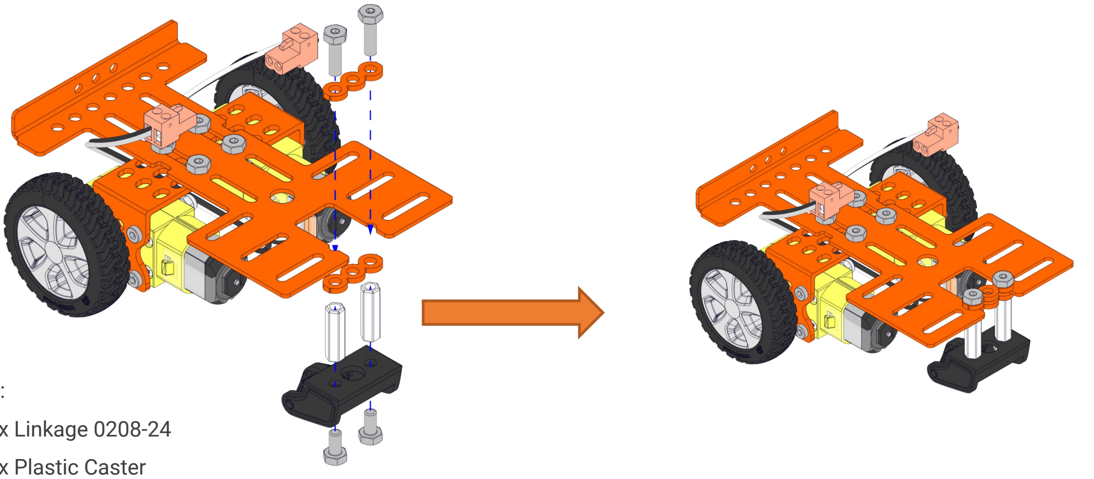


Part:

- 1 x Aluminum Chassis
- 4 x Screw M4*8
- 4 x Nut M4



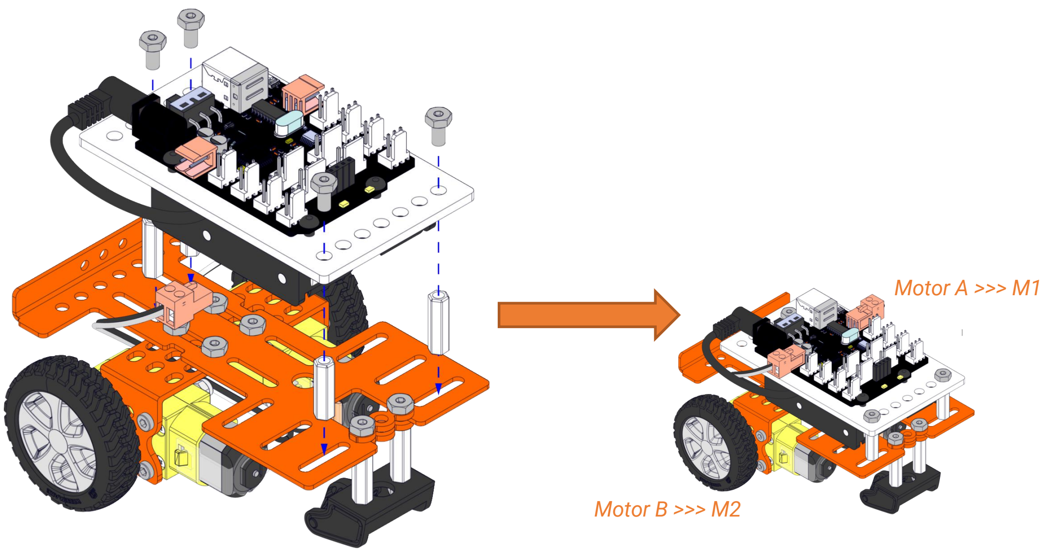
Sixth,install caster wheel.



Part:

- 2 x Linkage 0208-24
- 1 x Plastic Caster
- 2 x Nylon Stud M4*20
- 2 x Screw M4*14
- 2 x Screw M4*8

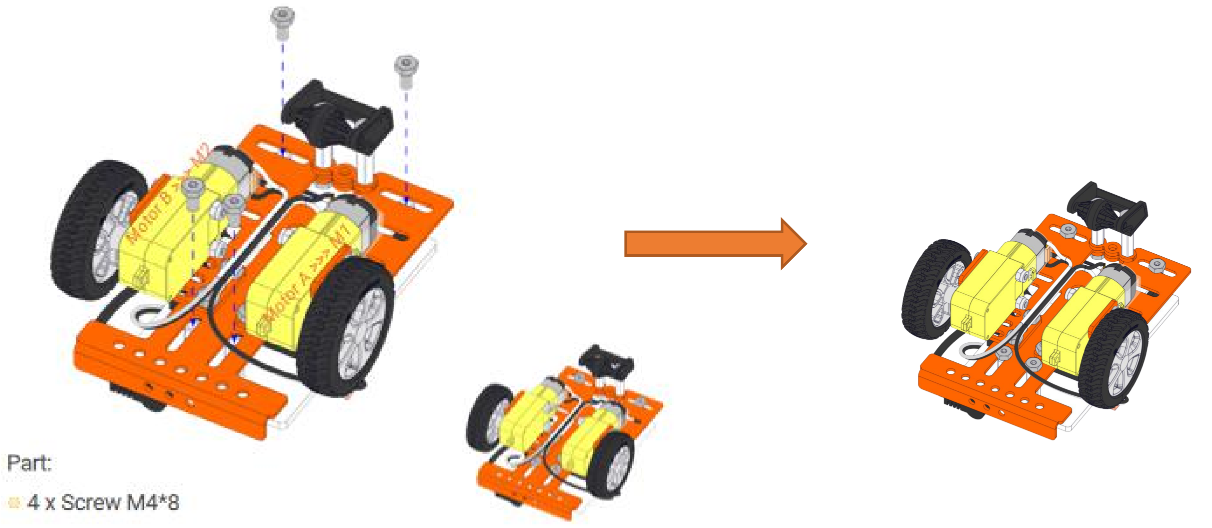
Seventh,install ELF Uno assembly - A



Part:

- 4 x Nylon Stud M4*20
- 4 x Screw M4*8

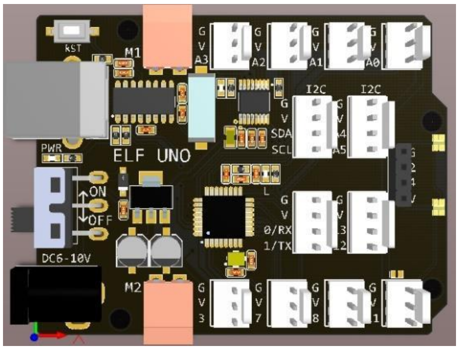
Eighth, install ELF Uno assembly - B



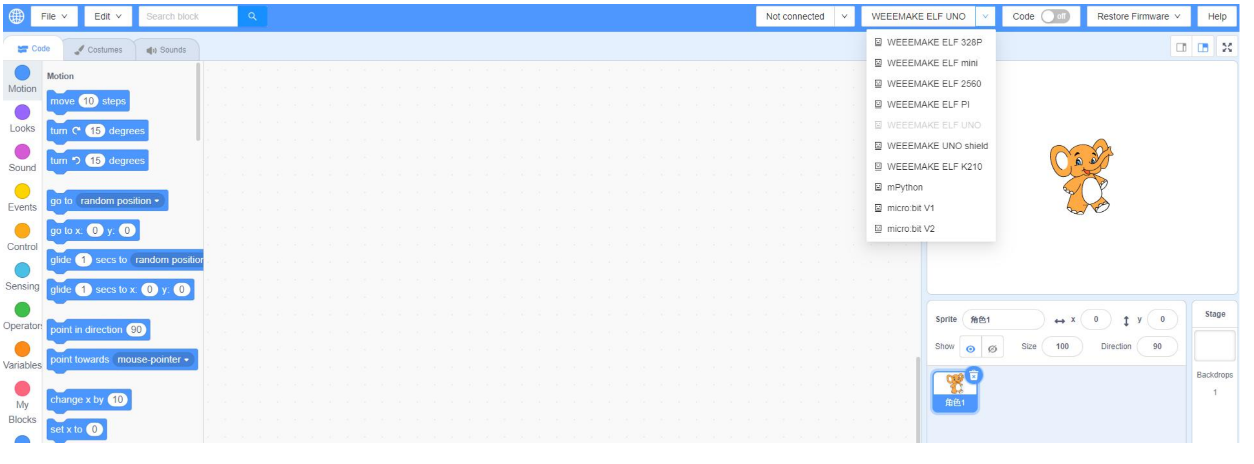
3.How does the robot connect with the computer?

Once the WeeeCode software is downloaded and installed on the computer, how do you connect the rover's mainboard to the computer software? Basically there are three steps:

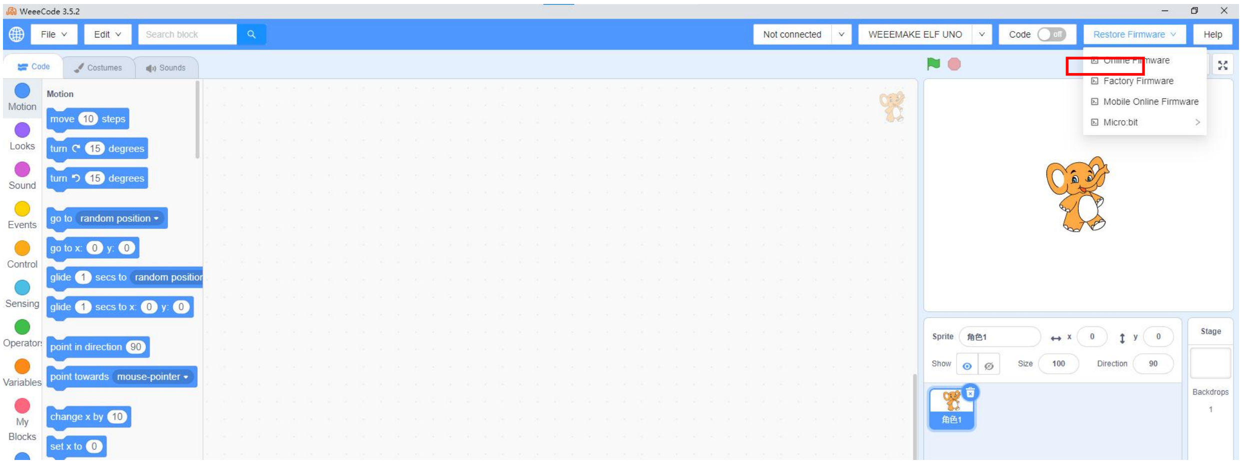
Step 1:Connect one end of the USB to the computer, and connect the other end to the master controller.



Step 2:Power on the mainboard, open the software, select WEEEMAKE ELF UNO mainboard, connect the serial port.



Step 3: Select the online firmware in the restore firmware, and when the message is uploaded successfully, the mainboard can communicate with the software in real time.



Now we know what a rover robot is and how it connects to a computer to program and "transmit" language, is it too much trouble for the rover robot and computer to run "online" programs? Is there a more convenient way?

Lesson Ten

Follow Me

Classmates, do you know how a robot works? What is the device that controls the robot to do simple thinking and judgment? How do rovers work on Mars when they are so far away from Earth when it takes a dozen minutes for each command to travel from Earth to Mars.To want to control the robot,we need to learn how to control the Robot's interrupt controller--Arduino.

In this lesson, we are going to get to know the Arduino microcontroller. Do you want to try it? "ZhuRong" is the Tianwen first mission rover. It is 1.85 meters high and weighs about 240 kilograms. The design life is three Martian months, equivalent to about 92 earth days. It was launched at 12:41 a.m. on July 23, 2020 in China carried by long March-5 from WenChang Space Launch Site .

The opening ceremony of the 2021 China Space Day is held on April 24, 2021 in Nanjing,China's Jiangsu province,China's first rover was named "ZhuRong". On May 17, the "ZhuRong Rover" sent back telemetry data from its orbiter the first time. At 10:40 am on May 22, the "ZhuRong Rover " left the landing platform and arrived at the surface of Mars.

Study Task

Task 1: Know the Brain of the Rover robot -- the Arduino Controller.
Task 2: Know, learn, and test the operate procedures of the Arduino Controller.
Task 3: Learn ultrasonic sensors and program them to follow the rover.

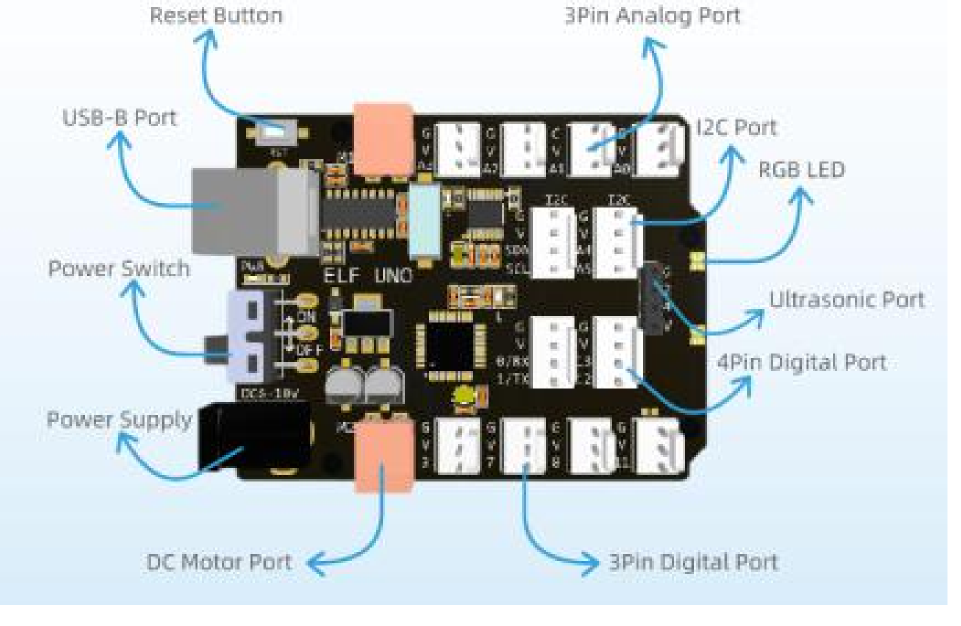
Knowledge Point

1. Know the Arduino Controller

Appearance and interfaces

Arduino controller is composed of "power supply interface", "data line interface", "motor interface", "sensor interface" and various chips.

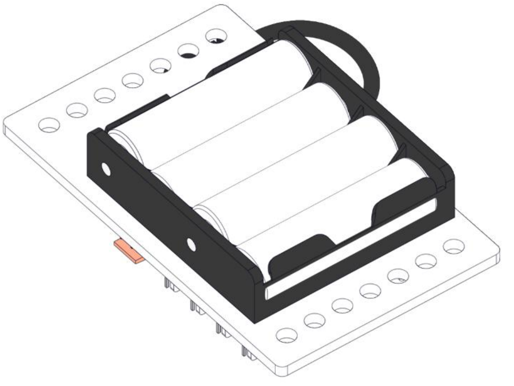
In the figure below, we can clearly see the specific name of each component.



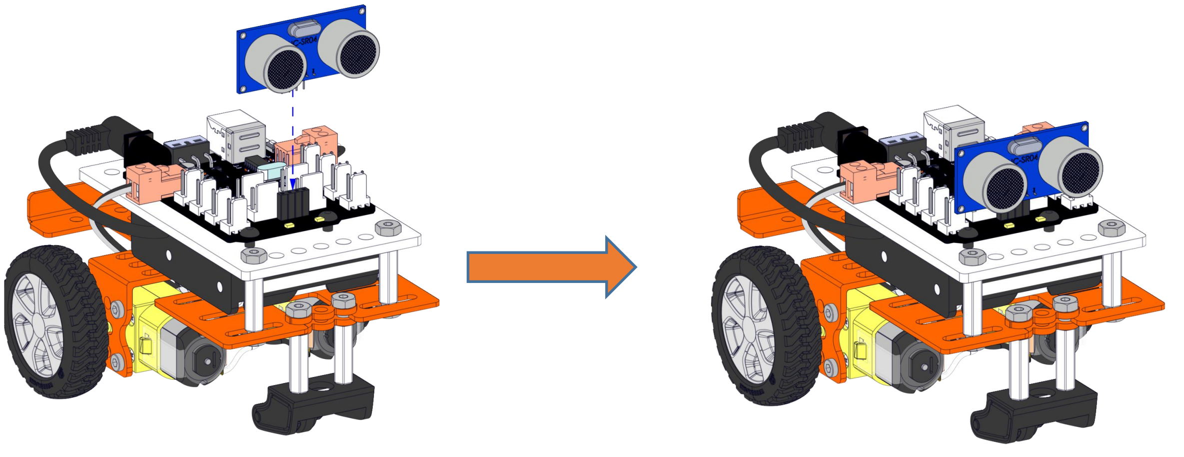
Arduino controllers have various multi-functional interfaces for easy access to external electronic devices and data transmission .

Precautions for installation

First，check the battery,the battery installation needs to distinguish the positive and negative poles, ensure that there is no short circuit when turning on the power, and put the battery in the correct way, as shown in the following figure.

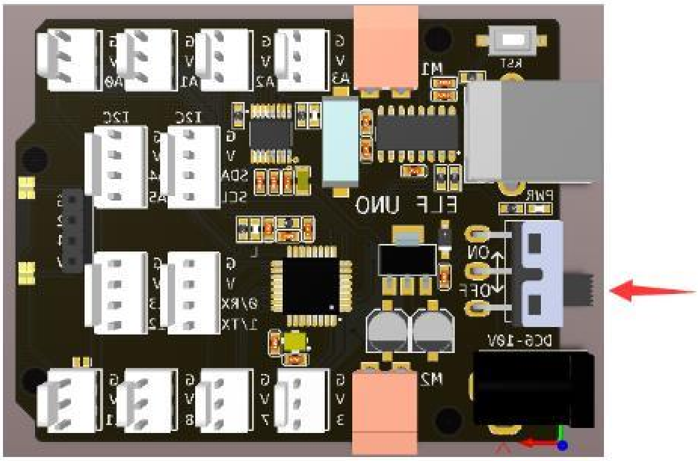


Second ,insert ultrasonic, when inserting ultrasonic, we must pay attention to the correctness of the interface. Once it is wrongly inserted, the ultrasonic is easily damaged after being energized, as shown in the following figure.



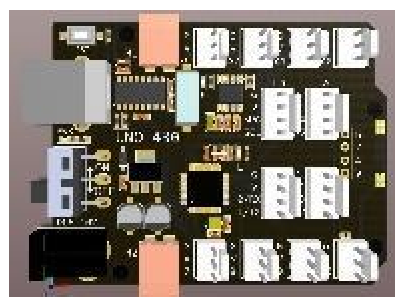
2. Operate procedures for "Arduino Controller"

Step 1: Turn on the switch of the "Arduino" controller with battery installed, and pay attention to the possibility that there is a control program downloaded in the controller to prevent the vehicle from falling due to sudden startup.



Turn on mainboard

Step 2: Connect the data cable, insert the type-A port of the data cable into the USB port of the computer, and insert the type-B square port of the data cable into the Arduino main controller. Be careful when inserting the cable, and do not damage the USB port vigorously.

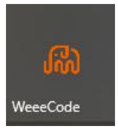


Connect data line

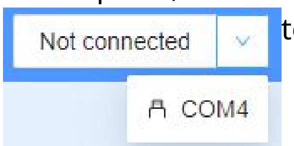
Practice

Programe for application of "ultrasound".

Step one: online debugging

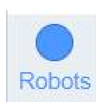
After confirming that controller power is turned on and the data line is connected successfully, open WeeeCode software 

on the computer, select WEEEMAKEELFUNO mainboard, 

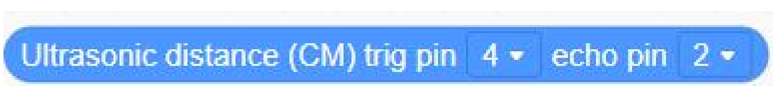
click 

to connect serial port, select online firmware 

when the upload is successful, the controller can communicate with the computer in real time.

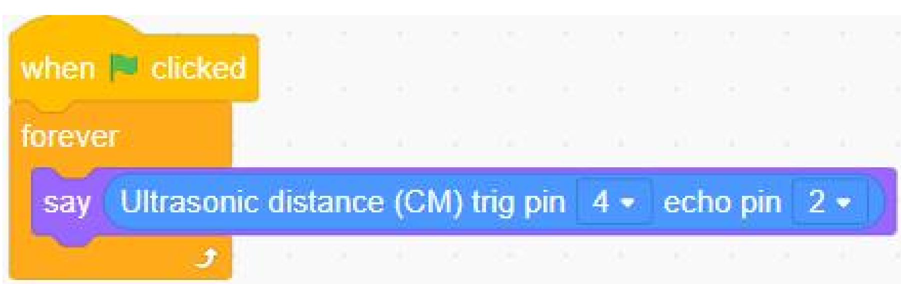
Note: In robot program category, 

find ultrasonic distance measuring block.



The block can select control pin *Trig* and value return pin *Echo. A*ccording to the port number tags and ultrasonic pin situation, we can learn that Trig pin is pin 4, Echo pin is pin 2. Double check the pin selection, do no select the wrong pin.

After finish writing program, click the green flag button  above the stage. The elephant then say the distance of the ultrasonic wave in centimetres.



Step 2: Program

After testing the ultrasonic distance, the following rover can be programed. The distance referenced here is as follows:

1. When the distance between the obstacle and the rover is less than 15cm, the rover will back and away from the obstacle;

2. When the distance between the obstacle and the rover is more than 25cm, the rover will move forward and approach the obstacle;

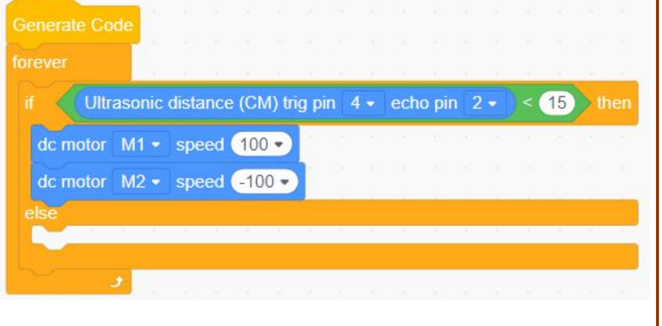
3. When the distance between the obstacle and the rover is 15-25 cm, the rover will stop moving.

Programming instructions

<table>
<tr><th>Programming steps</th><th>Programming instructions</th></tr>
<tr><td>

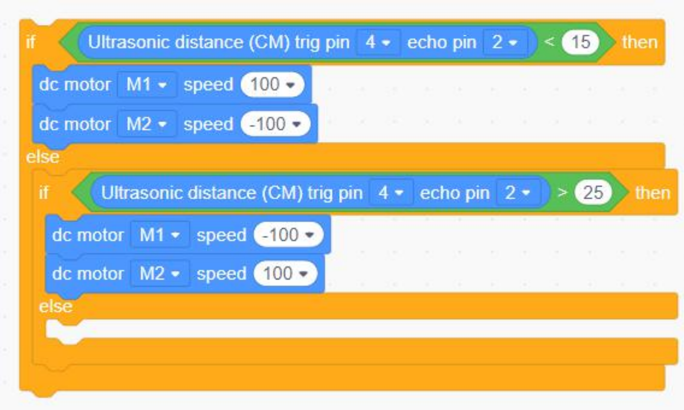

</td><td>
1. To run the program offline, you need to use the generated code program.

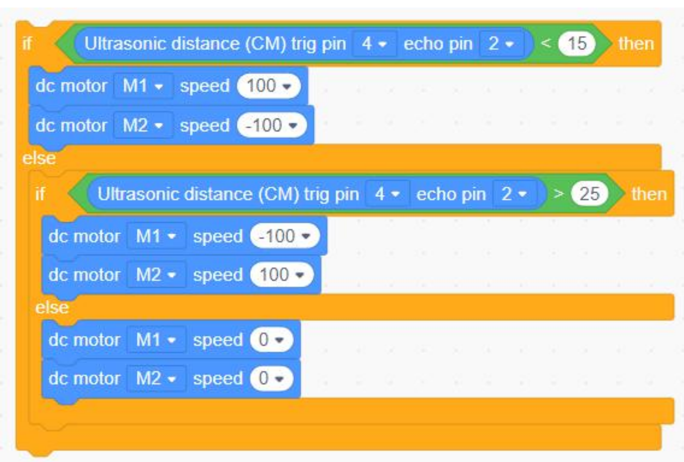

The program needs to run all the time, so you need to repeat the execution of the program block.


</td></tr>
<tr><td>


</td><td>
2. In the Control program bar, find "If... then...else"program block. According to the analysis, if the ultrasonic distance is less than 15 cm, then the vehicle move back, program is to control motor M1 to move clockwise and M2 motor to move counterclockwise.


</td></tr>
</table>

<table>
<tr>
<td>


</td>
<td>3. According to the program logic, when does not meet the contition that ‘distance between the ultrasonic and obstacles is less than 15 cm’, the program in else will be excuted, and then continue to detect whether distance between ultrasonic and obstacles is more than 25 cm or not. If more than 25 cm, then control the M1 and M2 motor to move forward.</td>
</tr>
<tr>
<td>


</td>
<td>4, when the program does not meet the conditions that the distance between ultrasonic and obstacle less than 15 cm, and does not meet the conditions that distance between ultrasonic and obstacle is greater than 25 cm, the program will execute the program content of else statement in inner condition judgment statement, that is, the program of the vehicle stop.</td>
</tr>
</table>

Step 3: Run offline

Once you've written your program, how do you upload it to the Arduino board?

1. Click open code switch. 

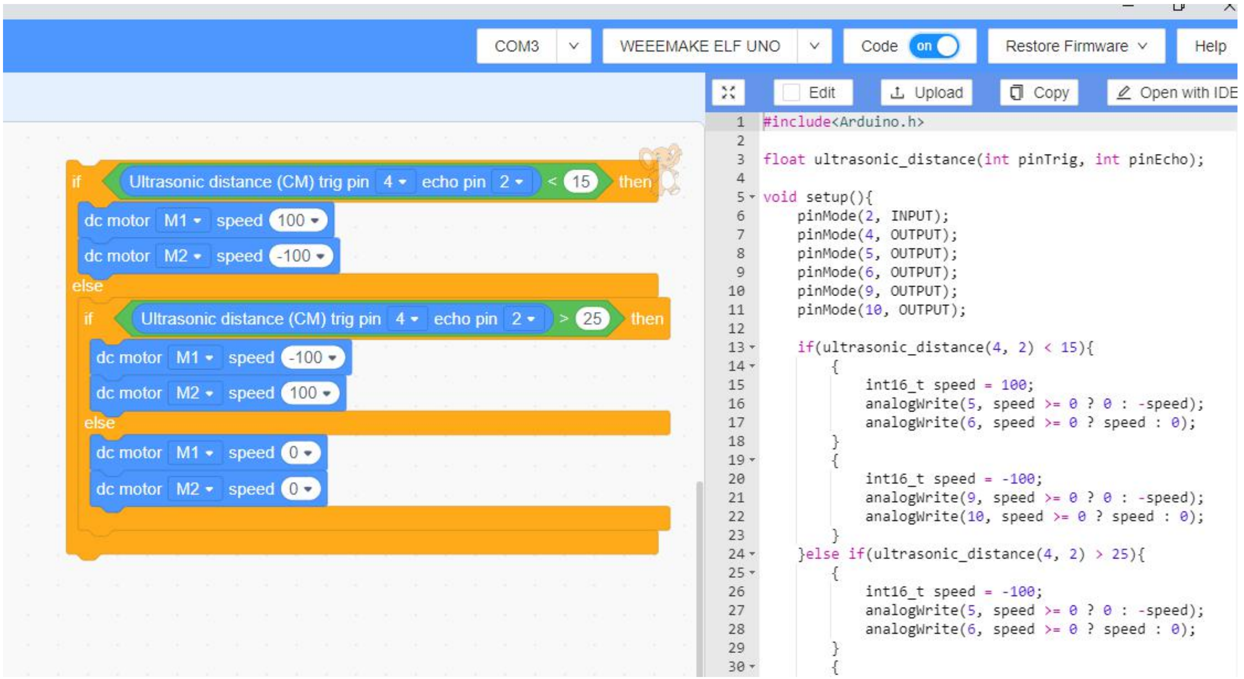
2. Click the upload button. 

3. When the upload success message is displayed  Upload successful , We can unplug the data and debug the rover.

If abnormal movement occurs, please carefully check the program and modify it.

After reconnecting the data line, upload the modified program to the Mars rover. If the fault persists, check whether the hardware is damaged. If the hardware is damaged, it is almost impossible to repair.

Therefore, connect the hardware properly to prevent short circuit and violent damage, and prevent conductors such as the screwdriver from contacting the pins.



Is it possible to test the return values of other types of sensors (sound, limit switches, light, etc.) in Arduino online mode?

Lesson Eleven

Obedient Mars rover

Classmates,do you want to communicate with the robot? Want it to do what we want it to do? We are going to learn the robot's special language - "programming language" to command the robot. Wouldn't it be cool to be able to control a robot through a programming language? In today's lesson, we are going to learn "programming language", to see if the robot can obey.

The Phoenix Mars lander successfully landed at the North Pole of Mars at 19:53 P.M. on May 25, 2008(Beijing Time). Phoenix launched from Cape Canaveral, Florida, in August 2007 and traveled 422 million miles to Mars. Phoenix is scheduled to spend 90 days on Mars to survey the red Planet's previously unexplored Arctic region, which is thought to have large amounts of ice hidden beneath the surface. Phoenix lacks the tools to detect signs of alien life, but it will study whether the ice has melted and look for traces of organic compounds in the Arctic permafrost to determine if life ever arose .

Study Task

Task 1: Comprehensively learn and get familiar with the user interface of programming software and common script instruction modules.

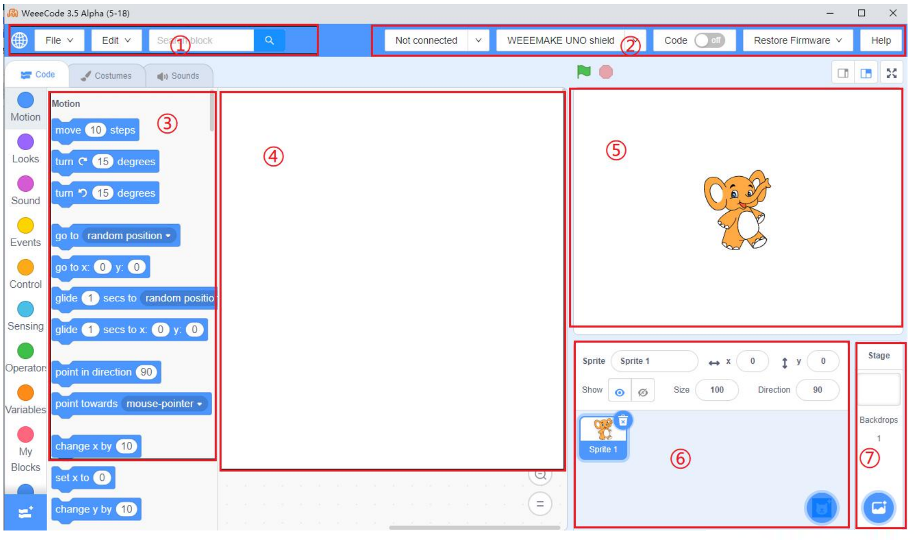
Task 2: Understand the three basic structures of "programming language" and the meaning of "task flow chart".

Task 3: Learn to use the "task flow chart" and the three basic structures of the programming language in programming.

Task 4: Learn the sound sensors, write an compliant rover program and debug the program.

Knowledge Point

1. Learn and be familiar with WeeeCode programming software.
Click the desktop icon to enter the interface as shown below. You can see that the "Software Interface" is divided into seven functional areas, such as



①、 Menu bar
②、 Hardware connection Settings area
③、 Scripts menu
④、 Programming area
⑤、 Stage area
⑥、 Role editing area
⑦、 Stage editing area

2. Three basic structures of a programming language: order, judgment and loop.

First,what is a "programming language"?

In order for a robot to perform various actions and tasks, we need to transmit information to the robot and send various instructions or statements that can be ccepted by robots. This is called "programming language", which has certain structure and needs to be written by human.

Second, three basic structure of " Programming Language"

Sequential structure: A linear, ordered structure that performs actions or tasks in sequence.

Judgment structure: a structure that executes according to conditions. It needs to be judged by conditions before performing different actions or tasks. It can be expressed like this: if the condition is satisfied, we execute A; If the conditions are not met, perform B.

Loop structure: a structure that allows a program to perform an action or task repeatedly. Conditions can be set at the entrance or end of the action to facilitate the program's progress.

Now let's use task flowcharts to help us learn and master programming languages,

3. Study and understand the task flowchart.

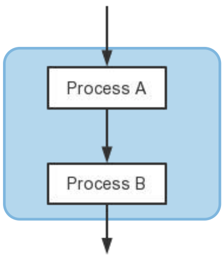
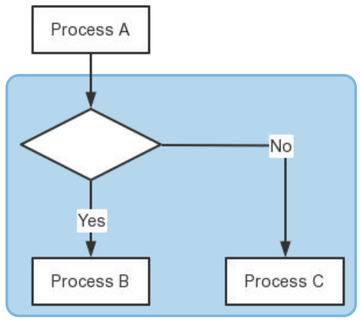
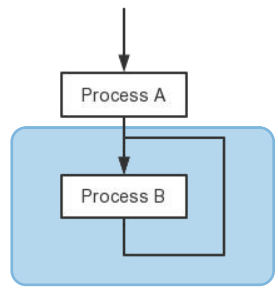
What is a task flowchart?
A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process. In flowchart, we use diagrams to think, analyze and judge, and use alternative routes, sequences and processes to achieve goals and results.

You'll notice that the flowchart has different shapes. These shapes are known as flowchart symbols.

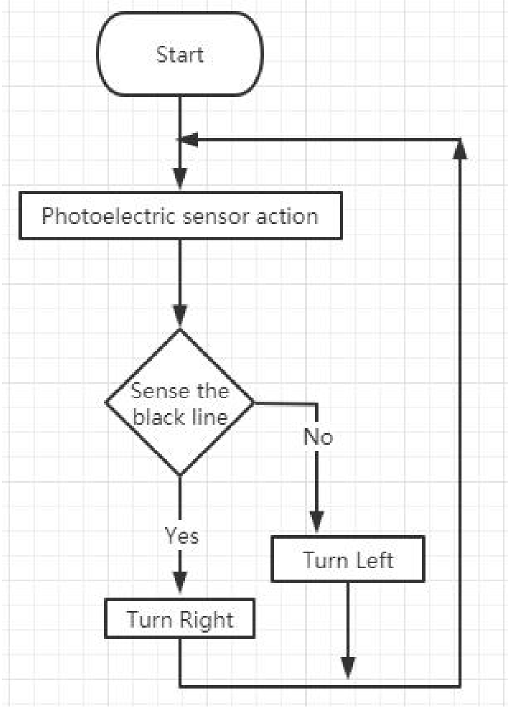
Symbol	Name	Function
	Start/End	The terminator symbol marks the starting or ending point of the system.
	Process	Represents a process such as a single step or and entire sub-process within a larger process.
	Decision	A decision or branching point. Lines representing different decisions emerge from different points of the diamond.
	Input/Output	Represents material or information entering or leaving the system.
	Arrows	A line is a connector that shows relationships between shapes.
	Connector	Indicates that the flow continues where a matching symbol has been placed.
	Data Storage	Indicates a step where data gets stored.

How to use task flowchart?

As shown below, it is easy to express the three basic constructs of a "programming language" in a "task flowchart."

<table>
<tr><th colspan="3">The three basic programming language structures</th></tr>
<tr><th>Sequences</th><th>Selections</th><th>Loops</th></tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>From process A enter into process B.</td>
<td>From process A to decision box, make selection “yes” to process B, or make selection “no” to process C.</td>
<td>From process A enter into process B, and then keep repeating process B.</td>
</tr>
</table>

A complete "task flow chart" representation.



Classmates can use to analyze the task and think about how the task will be performed.

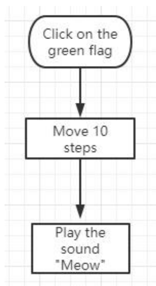
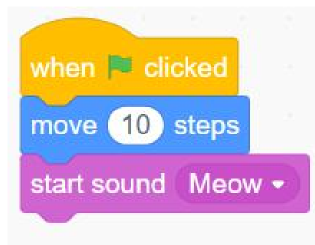
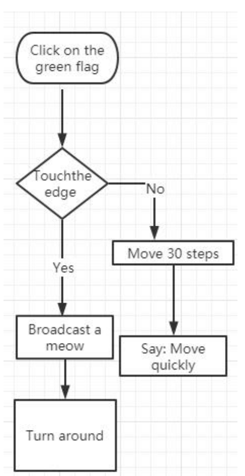
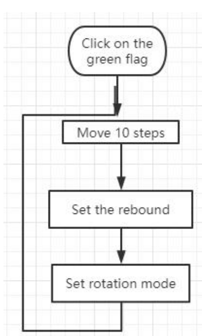
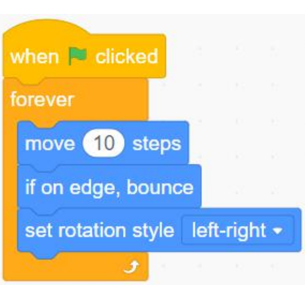
1. It analyzes tasks like this: First, "start" module, which is the starting point of program execution; Guided by the arrow, the next step is to enter the "photoelectric sensor action" module to ensure that the sensor works normally; Next step into the "sensing black line" module, let the sensor to judge whether the black line is detected, if detected, the robot turn right, otherwise (is not detected black line) the robot turn left.

2. Finally, no matter whether the robot turns right or left, it will return to the module of "photoelectric sensor action" to realize the circular action, so that the robot can continue this action.

Practice

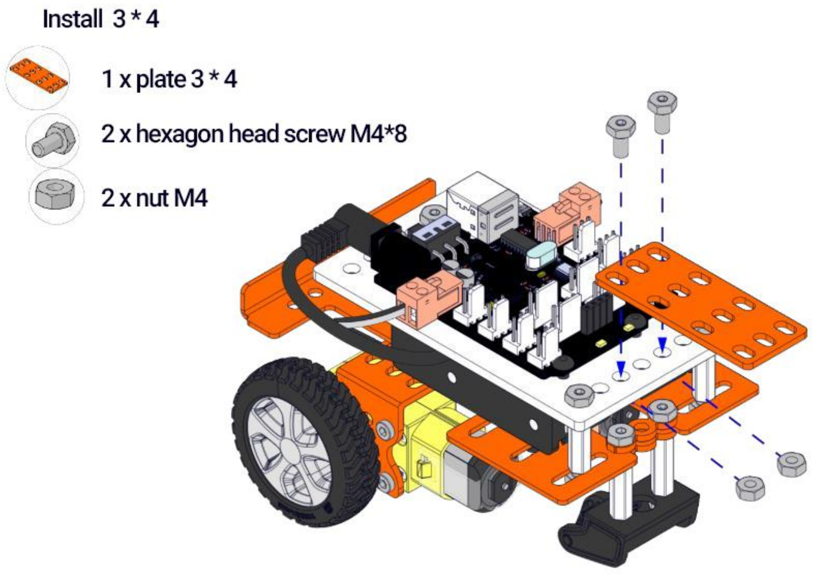
Classmates, we're going to use what we've learned to control programming from thinking to executing. You're going to do the three basic structures of a programming language.Let's begin!

Three basic structures of programming language

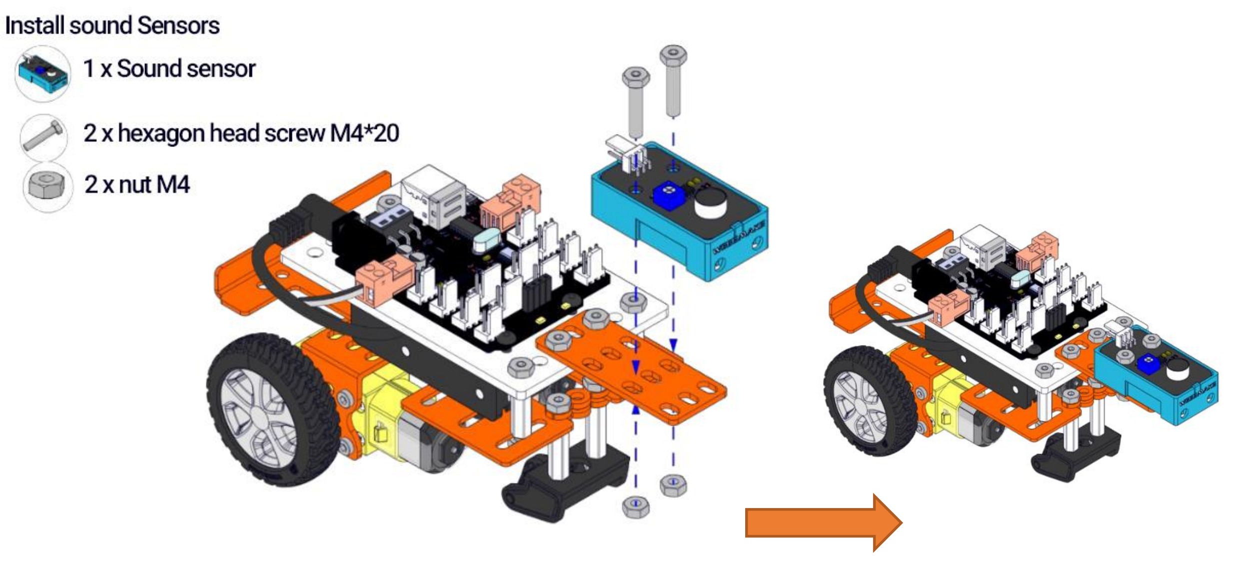
"Programming language" structure	Think "Task flow chart"	Use "structured" script instructions	Script instructions
Sequences structure:After the beginning of the program from top to bottom step to perform action one,action two, action three.			Press the green flag, execute in order.
Selections structure: the program starts after the execution of the judgment. If the conditions are met, perform action 2; otherwise, perform action 1.			Every time press the green flag, judge if touch the edge? If "yes" ,"meow" if touch the edge, turn around if "no" and If don't touch the edge, say "Go away , Go away" and move on.
Loops structure: perform action 1, action 2 and action 3 and then return to the loop point and repeat action 1, 2 and 3 again.			After the green flag is pressed, the actions are repeated.

New electronic components: sound sensor, sound sensor is connected with 3P connection wire and analog port , sound sensor function is equivalent to a microphone. It is used to receive sound waves and display vibration images of sound, but it cannot measure the intensity of noise.

1. Install Plate



2. Install sound sensor



Connect the sound sensor to digital port 3 with KF2510 3P terminal wire.

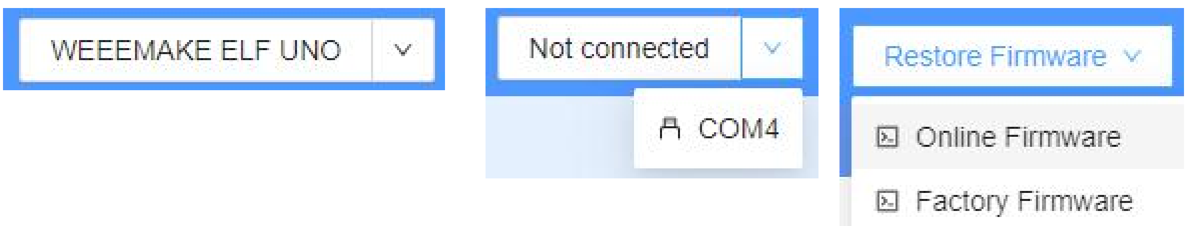
Application of "sound sensor"

In order to more intuitively know the "sound sensor" detection value,classnates can first use online operation, using WeeeCode small elephant to say the sound sensor value, so that you can better master the value of the sound sensor return.

Online debugging

After confirming that the power is turned on and the data cable is connected successfully, open the WeeeCode software of the computer.

Select WEEEMAKE ELF UNO mainboard, select serial port, select online firmware.



When upload successfully, we can get the controller to connect computer in real time. Upload successful

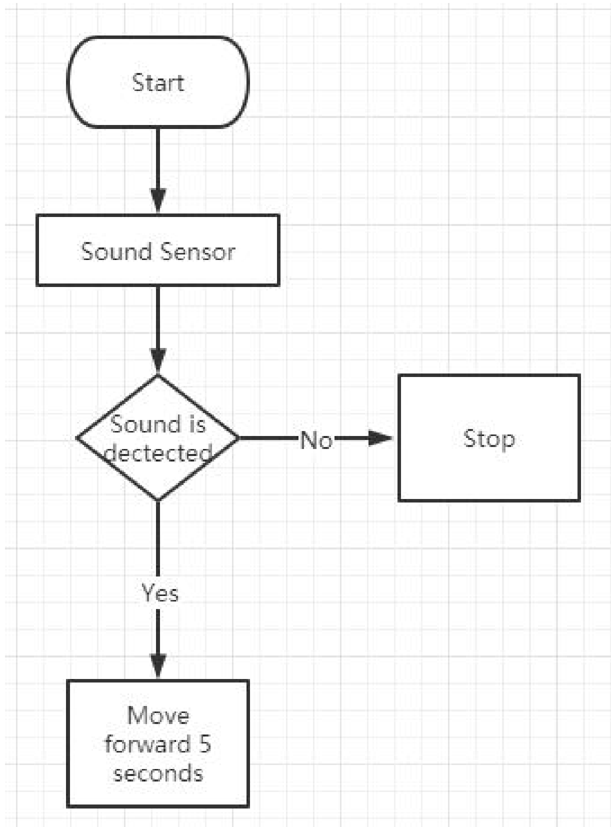
When finish writing program, click the green flag button  and the elephant will say the value that the sound sensor has detected.

After the sound sensor is tested, the rover car program can be written. The reference distance is as follows :

1. When the sound sensor detects the sound, the rover will move forward for 5 seconds and then stop.

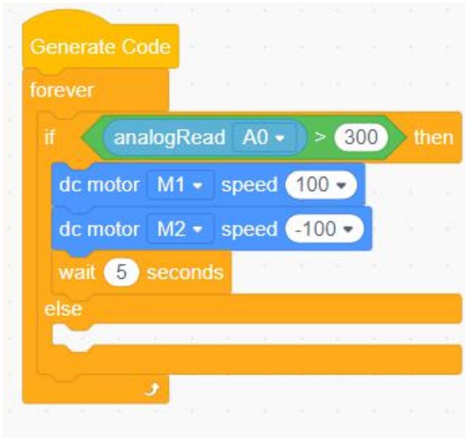
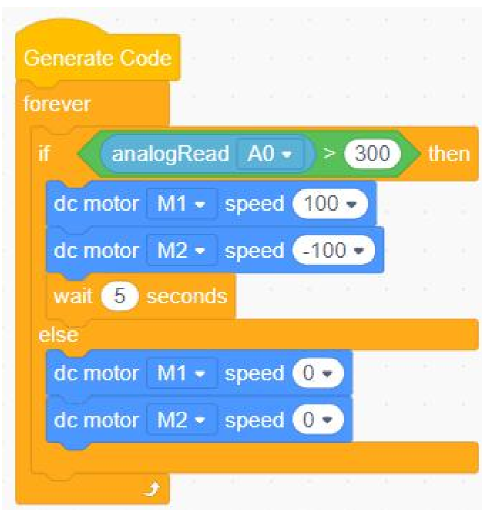
2. When no sound is detected, the vehicle stops.

The flow chart is as follows:



Program

Programming steps	Instructions
1. 	1.To run a program offline, you need to use generated code headers. The program needs to run all the time, so you need to execute the block repeatedly.

Programming steps	Instructions
2. 	2.In the control program bar, find the "*If... then... else*"program block. According to the analysis, if the sound is detected, then the vehicle forward for 5 seconds. To move forward, drive M1 motor clockwise and M2 motor counterclockwise. The robot stops when motor speed is 0.
3. 	3. According to the program logic, when the sound detection sensor does not detect the sound, it will execute the program in *else* statement, that is the motor stop program.

Offline debugging

Once you've written your program, how do you upload it to the Arduino board?

1. Click to open the code switch;

2. Click the compile and upload program button;

3, after the upload success tips, you can pull out data line, debug rovers, if appear abnormal movements, please check and modify the program, after the reconnect the wires, Mars will modify the program to upload the car, if still can't solve the problem, you can check whether the hardware is damaged, if yes, it is almost impossible to repair, Therefore, connect the hardware properly to prevent short circuit and violent damage, and prevent conductors such as the screwdriver from contacting the pins.

How do you make the rover turn using sound?

Lesson Twelve

Anti-Collision Mars Rover

Classmates, do you know how a real rover avoids obstacles? The rover is actually very "smart", it can avoid obstacles freely, but also can choose the appropriate research materials for collection, do you know how the rover is so smart?

Human exploration of Mars started in the 1960s. By the end of 2020, 45 Mars exploration activities had been carried out worldwide, including 21 by the United States, 19 by Soviet Russia, 1 by Japan, 2 by Europe, 1 by India and 1 by China. The history of human exploration of Mars can be divided into three stages according to technical level and mission content.
In the first stage, 1960-1970, relevant countries mainly carried out fly by exploration of Mars, sending pictures of Mars and exploring atmospheric parameters. However, the success rate of exploration missions was not high, and only one successful mission was completed.

In the second stage, 1970-1990, the United States and the Soviet Union resumed Mars exploration activities, focusing on orbiting and landing exploration. During this period, the success rate of exploration missions increased significantly. Many probes, including *Mariner 6*, successfully completed their missions. In particular, the Soviet Union's *Mars 3*, launched in 1971, successfully landed on Mars and became the first Mars lander. 1972 *Mariner 9* arrived on Mars as the planet's first satellite.

The 1990s is the third stage. The main exploration method is landing exploration, the main goal is to search for evidence of the existence of Martian water and signs of life. Among them are the well-known rovers Spirit, *Opportunity* and *Curiosity*, which have successfully found evidence of water on Mars.

There were only 24 successful or partially successful missions in 44 previous human exploration activities, with a success rate of less than 50%. The number of successful missions that land on Mars is in the single digits. Eight rovers have landed on Mars: *Viking ONE, Viking Two, Pathfinder, Opportunity, Spirit, Phoenix, Curiosity and Insight*, all of which were developed by the United States.

Up to now, there are six Mars exploration orbiters and two landing rovers still in operation. The orbiters include *Mars Odyssey, Mars Reconnaissance Orbiter, Mars Atmosphere and Volatile Evolution,* Europe's *Mars Express, Trace Gas Orbiter*, and India's *Mangalyan*.

The only active rovers are America's *Curiosity rover* and Its *Insight lander* on Mars.

Study Task

Task 1: Assemble a "rover" that can avoid obstacles.
Task 2: Get the computer to correctly read the information and data from the rover.
Task 3: Write the program and complete the control the rover to avoid obstacle.

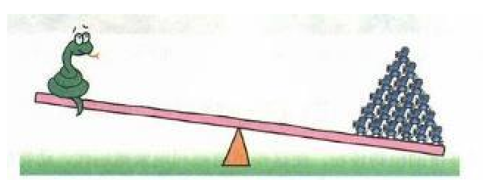
Knowledge Point

How do you assemble a "rover" that avoids obstacles?

Structural analysis of crashproof "rover"

New electronic components--limit switch (collision sensor), limit switch module is a physical switch, can play the role of limit anti-collision, when triggered, it will output a level signal to the control end. Limit switch is digital sensor.

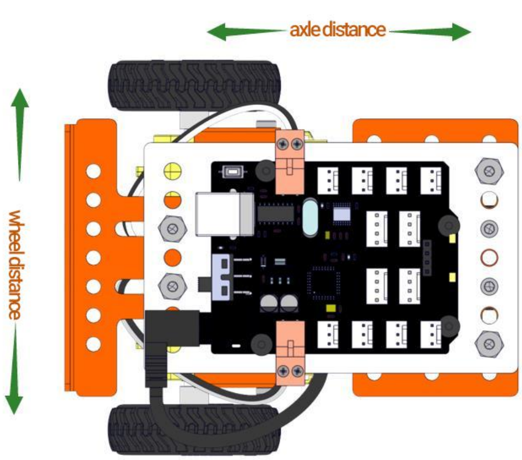
Key points to note when assembling



First : Keep balance and beauty.

1.Assembled 'Collision Mars Rover'
2.Balance back, front, left, right.
3.The weight should also be evenly distributed.

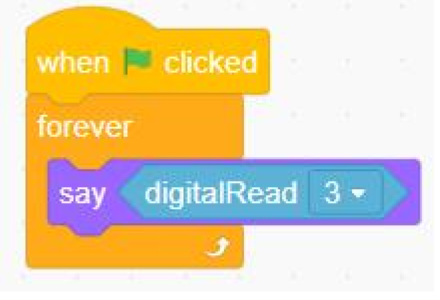
Second : control distance, as shown in the figure.



1. Control "wheel distance", refers to the control of left and right wheel space, which affects the rotation radius of the car.
2. Control "axle distance", refers to the control of the distance between front and rear wheels, which affects the sensitivity of the car steering.

Third : "Controller" reads test data on sensor

For the online control controller, we should learn to read the relevant information, data and information of the sensors on the computer, and control the car. When the rover is successfully connected to the computer, it can read the sensor information, data and control the output device. When the preparation is complete, it can be said to know the value of the "limit switch" sensor, as shown in the figure.



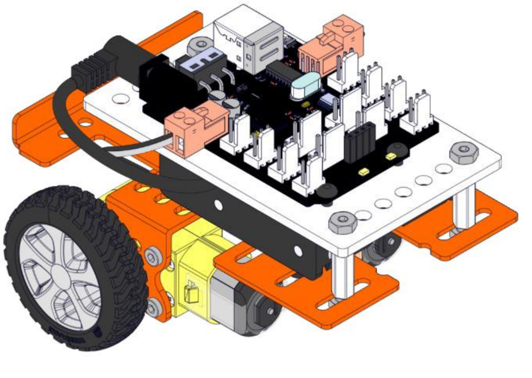
It is worth noting that when the limit switch does not collide with obstacle, the sensor returns a value of "1", which is the "true" said by the baby elephant; When the limit switch touches an obstacle, the sensor returns a value of "0", which is "false" said by the baby elephant. Only "0" and "1" two return values of the sensor, we call it a digital sensor;

The sensor whose return value is in a certain range is called an analog sensor. Common digital sensors: button, limit switch, digital sound sensor, ultrasonic, etc.Common analog sensors: light, potentiometer, slider, analog sound sensors, etc.

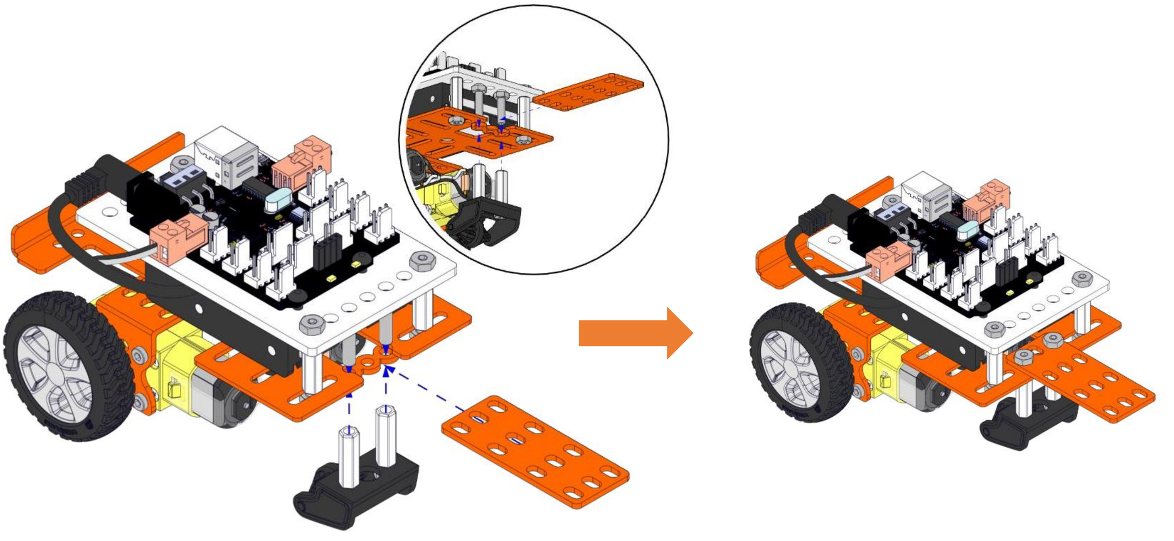
practice

Limit switch installation procedure

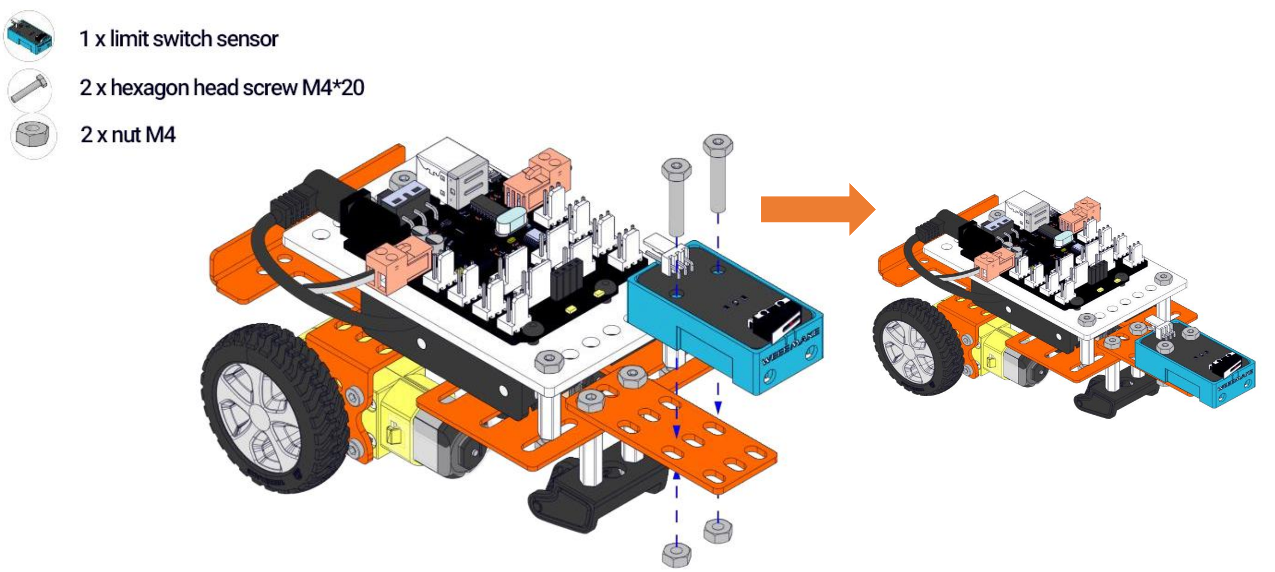
1. Remove the universal wheel



2. Install plate



3. Install the limit switch sensor



Use KF2510 3P terminal wire to connect limit switch sensor to digital port D3.

Attention:

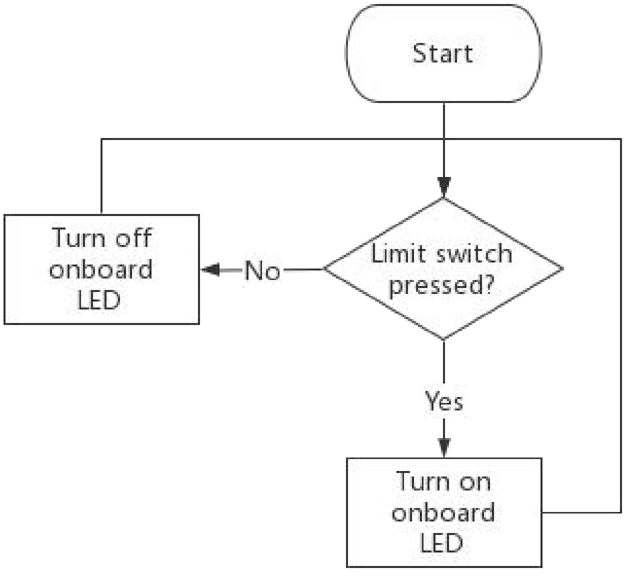
First point: when installing the sensor, use longer screws to fix it. Do not twist it too hard to prevent damage to the sensor or sensor base.

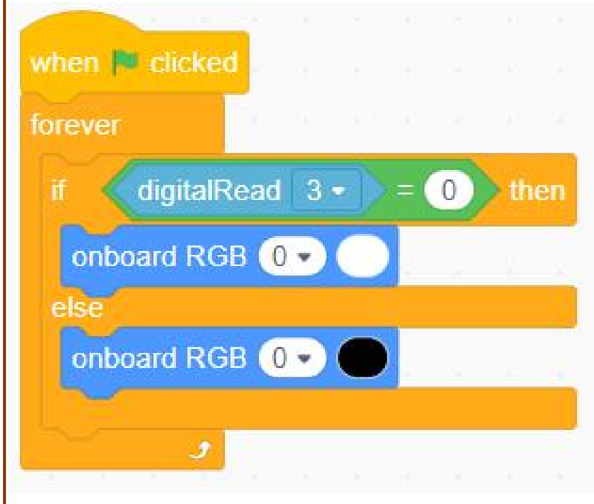
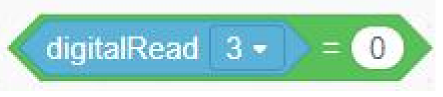
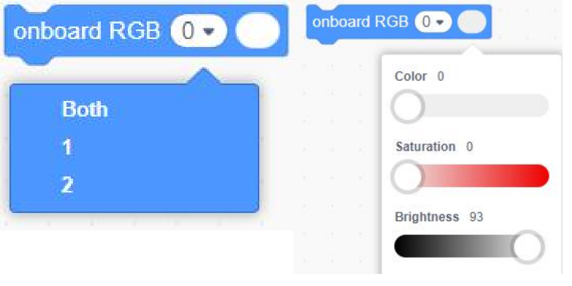
Second point: when assembling wiring, be sure to plug wire in accordance with the direction of wiring, remember not to reverse plug, and do not use metal devices to touch the plug pin.

Program

Program the limit switch to turn on the car LED

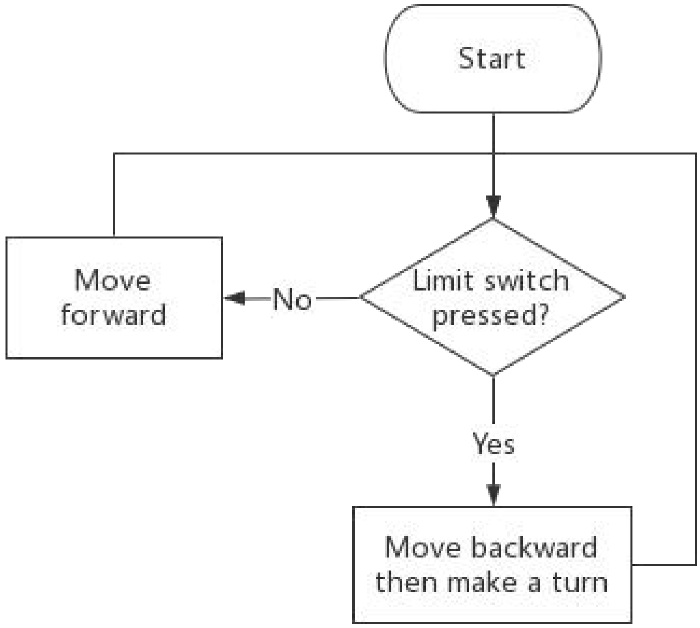
There are eight 3P interfaces on "Rover", among which 4 are analog sensor ports and 4 are digital sensor ports. In this case, limit switch is installed in digital port 3 port.

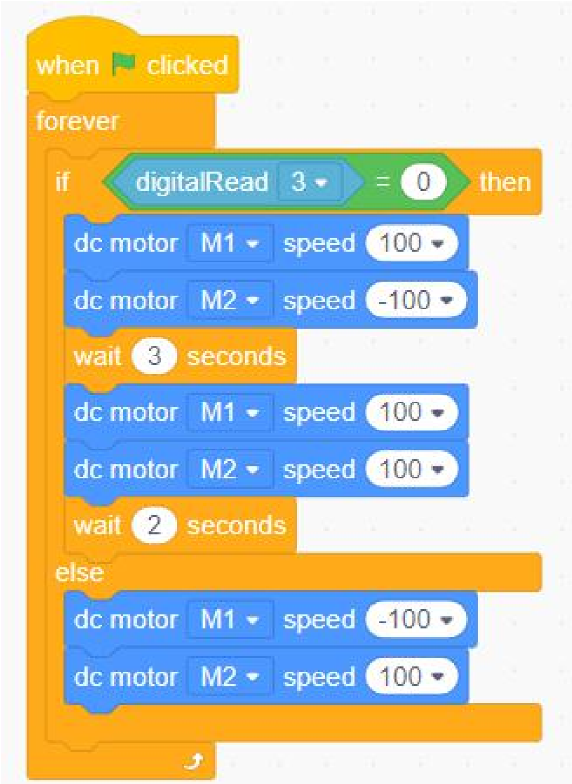


Programming steps	Instructions
	 When you press the touch sensor, the sensor returns a value of "0".  You can also select 1 or 2, the color can be set and selected by slider.

Program for Anti-Collision Mars Rover

The logic is the same with program to control the LED lights.



<table>
<tr><th>Programming steps</th><th>Instructions</th></tr>
<tr><td>
</td><td>

The motor on the left side of the rover is connected to M2, and the motor on the right side is connected to M1. When M2 turns forward and M1 reverses, the rover is in the forward state. When the vehicle hits a wall, it backs up for three seconds and then turns left for two seconds to avoid obstacles.</td></tr>
</table>

Okay, after learning lessons, have a try to make your Anti-Collision Mars Rover.

1. Can our rover connect other sensors?

2. What's the difference between front wheel drive and rear wheel drive?

Lesson Thirteen Follow Light

Classmates, do you know how the rover obtains energy for sports and scientific expeditions on far away Mars? In fact, many students know that it is the use of solar energy. Do you know what solar energy is? Why can solar panels generate electricity when exposed to sunlight?

Solar energy is a kind of renewable energy. It refers to the heat radiation energy of the sun, which is mainly represented by the sun's rays. In modern times, it is generally used to generate electricity or provide energy for water heaters.

Since the birth of life on earth, it has mainly relied on the heat radiation energy provided by the sun to survive. Since ancient times, humans have also known to use sunlight to dry objects and use them as a method of making food, such as making salt and drying salted fish. With the decline of fossil fuels, solar energy has become an important part of human energy use and has been continuously developed. There are two ways to use solar energy: light-to-heat conversion and photoelectric conversion. Solar power generation is an emerging renewable energy source.

There are two ways of solar power generation, one is the light-heat-electric conversion method, and the other is the light-electricity direct conversion method.

(1) The light-heat-electric conversion method uses solar radiation to generate electricity. Generally, a solar collector converts the absorbed heat into steam of the working fluid, and then drives a steam turbine to generate electricity. The former process is a light-heat conversion process; the latter process is a heat-electric conversion process.

(2) The light-electricity direct conversion method uses the photoelectric effect to directly convert solar radiation energy into electrical energy. The basic device of light-electricity conversion is the solar cell. A solar cell is a device that directly converts sunlight energy into electrical energy due to the photovoltaic effect.

It is a semiconductor photodiode. When the sun shines on the photodiode, the photodiode will turn the sun’s light energy into electrical energy to produce Current. When many batteries are connected in series or in parallel, a square array of solar cells with relatively large output power can be formed.
Generally, the solar panel on the rover is a device that directly converts light energy into electrical energy.

Study Task

Task 1: Assemble a "Mars Rover" that can recognize light
Task 2: Let the computer correctly read the light sensor data on the rover
Task 3: Write the program and complete the control of the "following light" Mars rover

Knowledge Point

How to assemble a "mars rover" that recognizes light?

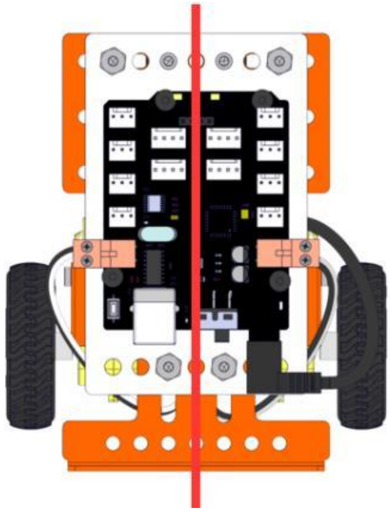
1. Structural analysis of the light-following "Mars Rover"

Classmates, after a few lessons, I believe you must be familiar with the rover. Do you remember the names of the connection structures and sensors of the rover? If you don’t know, think about the content of the previous lessons.

New electronic components: light sensors, also called brightness sensors, are mostly used in tablets and mobile phones. Generally located on the top of the handheld device screen, it can automatically adjust the brightness of the handheld device screen according to the brightness of the light where the handheld device is located, and bring the best visual effect to the user.

2. Important notes when assembling

The first point: keep centering and symmetry.
The assembled "following light rover" must maintain symmetry in the front, rear, left, and right, and the weight distribution must be even.



The second point: the direction is correct and the wiring is convenient.

When installing the sensor, pay attention to the direction of the sensor to prevent it from finding that it is too close to the other structures of the vehicle after the installation is complete, and the connection line cannot be connected.

The third point: the "controller" reads the test data of the "sensor".

Classmates, as usual, use the data cable to connect the controller and the computer. For online control of the main controller, learn to read the relevant information, data and data of each sensor on the computer, and control the car. When the "Mars Rover" is successfully connected to the computer, it can read the sensor information, data and information connected to the "Mars Rover" and control the output device to work. When the preparatory work is completed, we can read the value of the "light sensor" by "speaking", as shown:

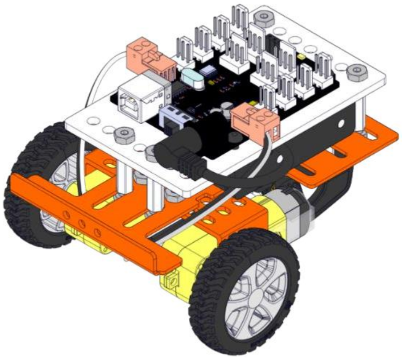
It is worth noting that the return value of the light sensor is very sensitive. When the light is completely shielded, the return value of the sensor can even reach 0; theoretically the maximum value of light is 1023, but it is difficult to reach the strongest light.
In the Arduino sensor we learned, the analog value returned by the analog sensor is 0-1023; if the digital sensor is connected to the analog sensor socket, the returned value is only two, 0 or 1023, which is determined by the characteristics of the sensor.
Common digital sensors: buttons, limit switches, digital sound sensors, ultrasonics, etc.;
Common analog sensors: light, potentiometer, slider, analog sound sensor, etc.

Practice

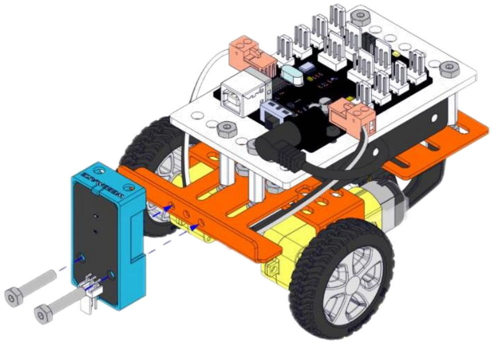
Students, through the above learning, now install the light sensor on the rover!

Light sensor installation steps.

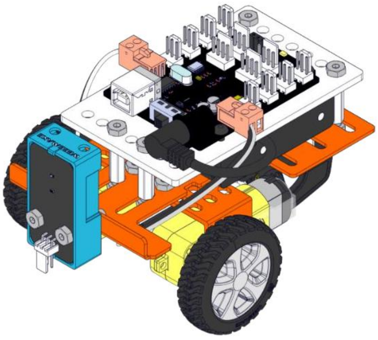
The assembling method of the chasing rover is divided into 3 steps. Students can refer to the picture below to complete the assembly. It takes about 5 minutes. Next, we will start to install it.



1. Locate the screw hole at the rear of the vehicle



2. Install the light sensor in the screw hole



3. Wiring instructions: Use KF2510 3P terminal wire to wire the light sensor to the A0 port of the mainboard.

Notice:

The first point: When installing the sensor, you should use a longer screw to fix it, and you should not screw it too hard to prevent damage to the sensor or the sensor base.

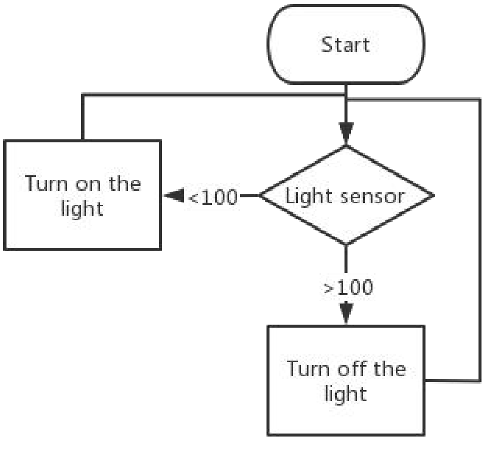
The second point: When assembling the wiring, be sure to insert the wiring according to the wiring direction, remember not to insert it backwards, and do not use metal devices to contact the wiring pins.

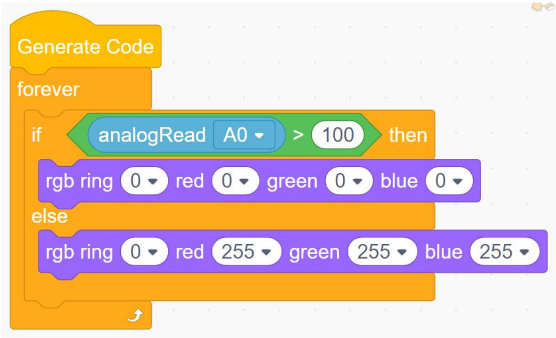
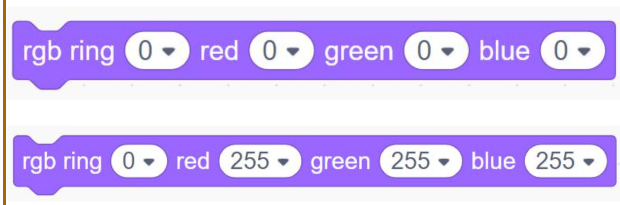
Check the connection status of the rover

1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted.
2. Data cable connection test, check whether the data cable is plugged in firmly.

Make a simple programming: use light to make car LED lights up.

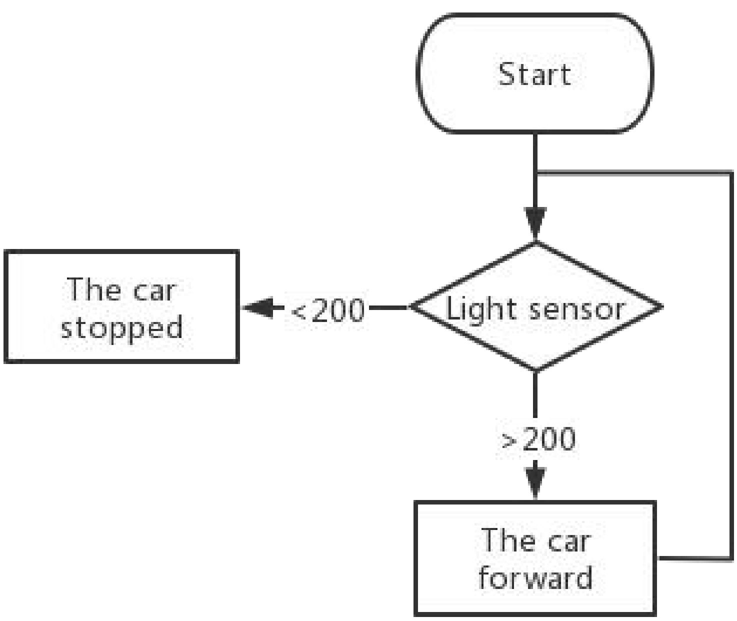
There are 8 3P line ports open on the "Mars Rover", 4 of which are analog sensor ports and 4 are digital sensor ports. In this case, we install the light sensor on the analog port A0 port.

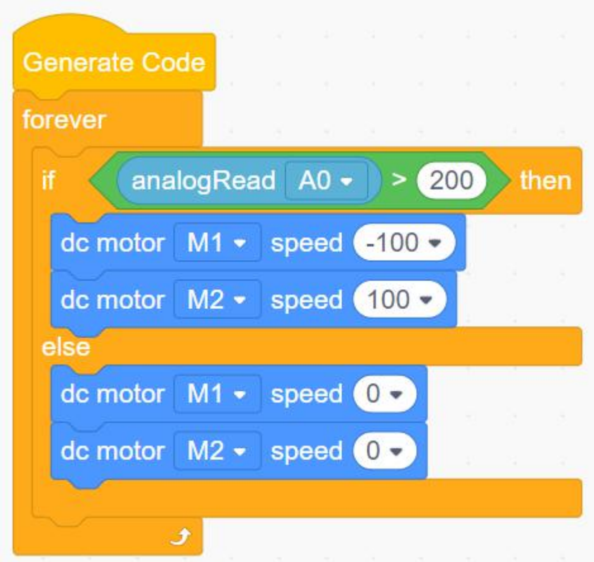


Program structure	Program description
	 Colorful lights are composed of three colors, red (R), green (G), and blue (B), which are commonly known as the three primary colors of RGB; Each color can be set, the maximum is 255, the minimum is 0, if all are 255, then it is white; if all are 0, then it is black.

Write a program for the light chasing rover

The program logic of the "following light rover" is the same as the on-board LED lights we wrote. You can complete the program as long as you know the movement state of the vehicle after it hits a nearby light source.



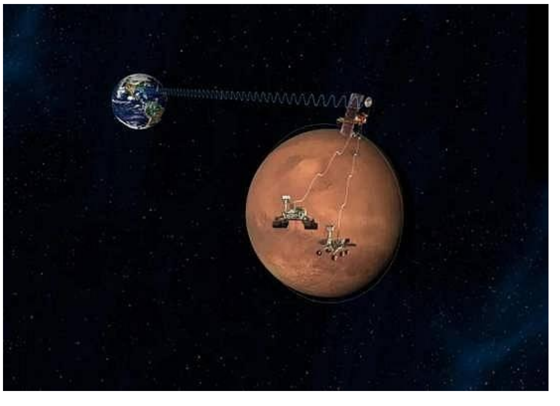
Program structure	Program description
	 Why the light value here is greater than 200, while the light value of the previous case only needs to be greater than 100? Because the previous case simulates a smart light system that turns off the lights during the day and turns on the lights at night, and this work imitates the energy movement of a Mars rover under strong sunlight, the thresholds of the two cases are different.

?

Do you know that there is a cloudy sky on Mars? On a cloudy day, what energy should the Mars rover use to move?

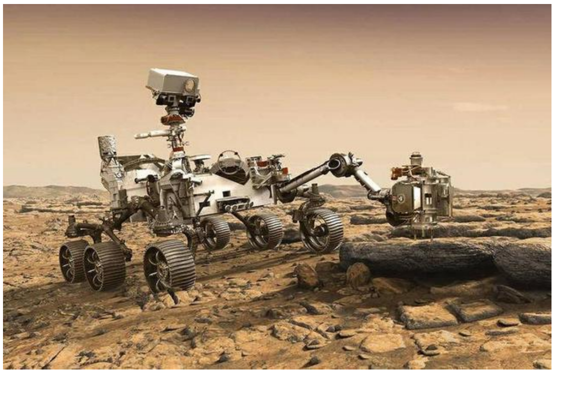
Lesson Fourteen If we can play music on Mars

Classmates, we all know that on the earth, sound is transmitted through the medium. Commonly used media are air, water and some solids. When we speak, the vibration of the vocal cords drives the surrounding air to vibrate, and a little bit is transmitted to the eardrum, and people can hear the sound. But on Mars, where the air is thin and people need to wear space suits, what kind of communication is needed between people? The Tianwen 1 Mars rover in my country uses electromagnetic waves to keep in touch with the earth. Humans have used radio communication for more than 100 years. Radio is a part of electromagnetic waves and belongs to low-frequency electromagnetic waves. Light is also an electromagnetic wave. The propagation speed of radio is naturally the same as the speed of light, which can propagate 300,000 kilometers per second in a vacuum. The speed of radio propagation is limited, and as the distance increases, the communication delay increases.



If sound can travel at the speed of light in a vacuum and you shout from the earth, it will take a long time for people on Mars to hear your voice. People on Mars responded, and when you received this response, a long time passed.

To control the probes on Mars from the earth, the commands can only be sent one by one, and sometimes they have to be sent repeatedly. This means that there is no real-time remote control of unmanned vehicles on Mars from Earth. To this end, the rover must be smart enough, have a high degree of automation, and be able to automatically avoid risks.



Study Task

Task 1: Assemble a "Mars Rover" player
Task 2: Write a program to complete the function of the Mars rover breathing light

Knowledge Point

How to assemble a "Mars Rover" player?

1. The principle of the "Mars Rover" player

The buzzer is an electronic buzzer with an integrated structure. It is powered by DC voltage. It is widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products as sound devices.

Electromagnetic buzzer: The electromagnetic buzzer is composed of oscillator, electromagnetic coil, magnet, vibrating diaphragm and shell.

After the power is turned on, the audio signal current generated by the oscillator passes through the electromagnetic coil, causing the electromagnetic coil to generate a magnetic field. The vibrating diaphragm periodically vibrates and produces sound under the interaction of the electromagnetic coil and the magnet.

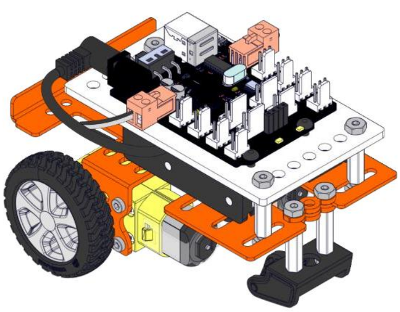
New electronic components: buzzer, which is an integrated structure of electronic buzzer, belongs to a kind of electronic components, adopts DC voltage or AC voltage to supply power, and is widely used in the computer industry (main board buzzer, chassis beeper) Buzzer, computer buzzer) printer (control panel buzzer), copier, alarm industry (alarm buzzer, alarm buzzer), electronic toys (music buzzer), agriculture, automotive electronic equipment industry (Car buzzer, reversing buzzer, car buzzer, motorcycle buzzer) telephone (environmental protection buzzer), timer, air conditioner, medical equipment and other electro-acoustic industries and environmental monitoring.

Practice

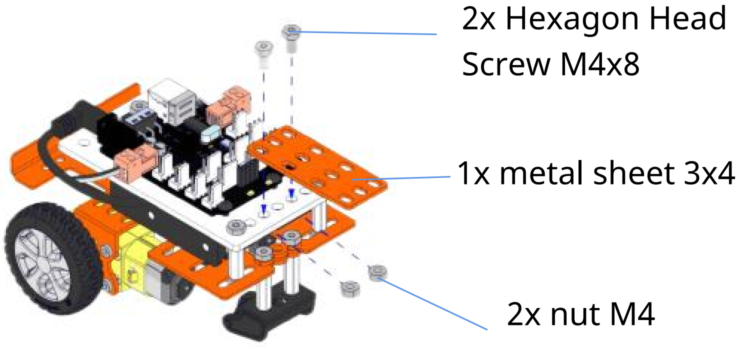
Students, through the above learning, now install the buzzer on the rover!

Buzzer installation steps

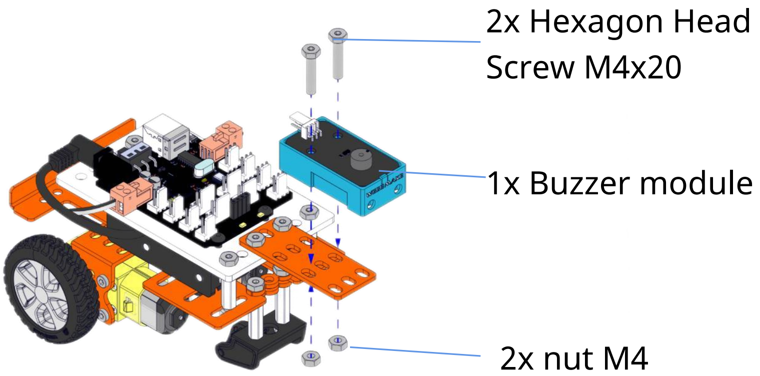
The assembly method of the rover player is divided into 4 steps. Students can refer to the figure below to complete the assembly. It takes about 5 minutes. Next, we will start to install it.



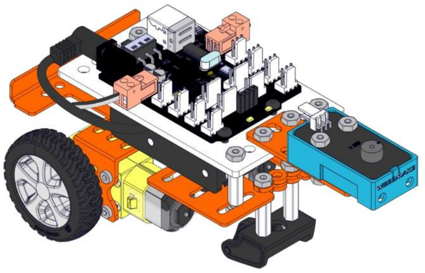
1. Locate the acrylic screw port on the motherboard



2. Mounting piece 3x4



3. Install the sensor

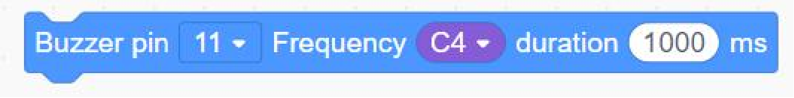


4. Wiring instructions: Use KF2510 3P terminal wire to wire the sound sensor to the digital port No. 11.

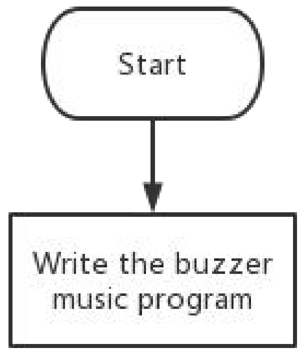
Notice:
When assembling the wiring, be sure to insert the wiring according to the wiring direction, remember not to insert it backwards, and do not use metal devices to touch the pins of the wiring.

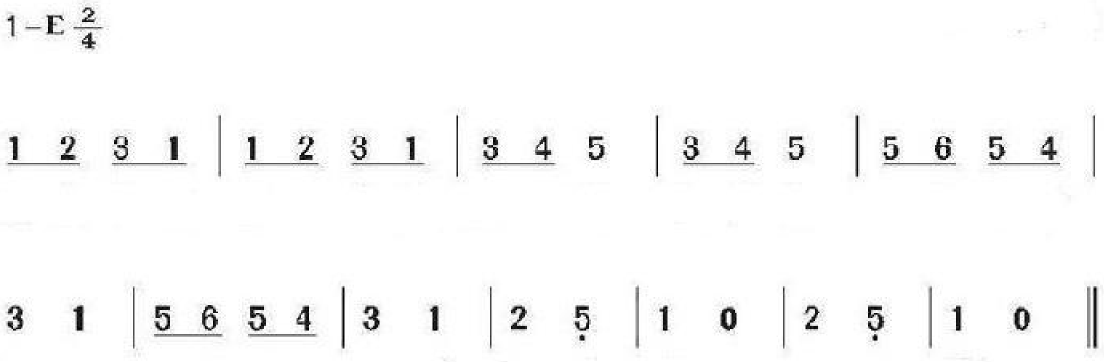
Check the connection status of the rover
1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted;
2. Data cable connection test, check whether the data cable is plugged in firmly.

Program the rover player
The program logic of the "Mars Rover" player is a piece of code that is executed sequentially. First of all, we need to know how to write a piece of music.
We can use the following program block to complete the buzzer sounding, where the pin is the interface, the frequency is the sounding frequency, and the duration is the beat.



C	D	E	F	G	A	B
1	2	3	4	5	6	7
Do	Re	Mi	Fa	Sol	La	Si





<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>

</td><td>According to the notation of a song, you can write a program, where the duration is the beat of the sound, and students who have studied music should know it.</td></tr>
</table>

```
#include<Arduino.h>

void buzzer(int pin, long frequency, long duration);

void setup(){
      pinMode(11, OUTPUT);

      buzzer(11, 262, 500);
      buzzer(11, 294, 500);
      buzzer(11, 330, 500);
      buzzer(11, 262, 500);
      buzzer(11, 262, 500);
      buzzer(11, 294, 500);
      buzzer(11, 330, 500);
      buzzer(11, 262, 500);
      buzzer(11, 330, 500);
      buzzer(11, 349, 500);
      buzzer(11, 392, 1000);
      buzzer(11, 330, 500);
      buzzer(11, 349, 500);
      buzzer(11, 392, 1000);
      buzzer(11, 392, 500);
      buzzer(11, 440, 500);
      buzzer(11, 392, 1000);
      buzzer(11, 349, 500);
      buzzer(11, 392, 1000);
      buzzer(11, 392, 250);
      buzzer(11, 440, 250);
      buzzer(11, 392, 250);
      buzzer(11, 349, 250);
      buzzer(11, 330, 500);
      buzzer(11, 262, 500);
      buzzer(11, 392, 250);
      buzzer(11, 440, 250);
      buzzer(11, 392, 250);
      buzzer(11, 349, 250);
      buzzer(11, 330, 500);
      buzzer(11, 262, 500);
      buzzer(11, 294, 500);
      buzzer(11, 196, 500);
      buzzer(11, 262, 500);
      delay(500);
      buzzer(11, 294, 500);
      buzzer(11, 196, 500);
      buzzer(11, 262, 500);
      delay(500);
}

void loop(){
}

void buzzer(int pin, long frequency, long duration){
       long pulse = 500000 / frequency;
       for(long i=frequency*duration/1000;i>0;--i){
           digitalWrite(pin, HIGH);
           delayMicroseconds(pulse);
           digitalWrite(pin, LOW);
           delayMicroseconds(pulse);
                          }
}
```

Classmates, let's get your hands dirty and make a player with a rover.

?

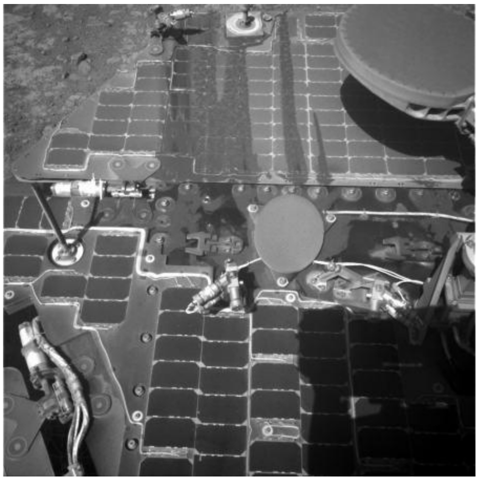
Classmates, do you know the numbered musical notation of any songs, try to compile it with a buzzer.

Lesson Fifteen Do you know breathing light?

Students, there are dangers everywhere on Mars. If you don't deal with it, the rover may be buried on Mars and will never be able to work. At this time, scientists can only declare the rover's "death".

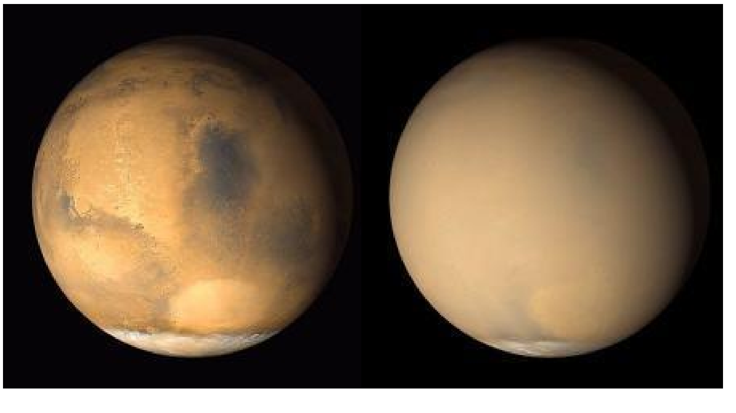
The US "Opportunity" has a fatal flaw in its design, which is its huge and horizontal layout.

NASA has previously known that there will be a strong sandstorm sweeping the world on Mars, but they still designed a large piece of solar cell for Opportunity that cannot adjust the angle by itself, and there is no necessary dust removal equipment designed to support it. For example, a simple mechanical brush (like a car wiper).

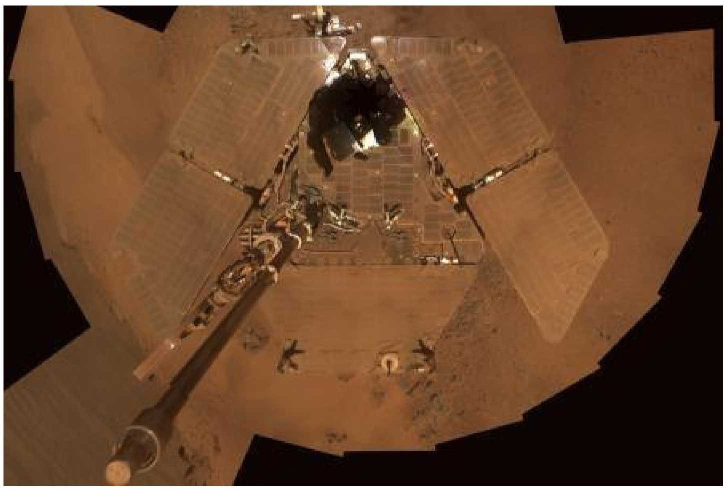


When a sandstorm arrives, the "Opportunity" will be covered by dust and lose its power supply. It can only wait for the high wind to blow away the dust on the battery. If the wind cannot blow away the sand and dust on the battery, and the lithium battery it carries is not enough to provide power for self-rescue, then the "Opportunity" rover can only wait to die on the spot.

The air on Mars is very thin, but even such thin air often blows up global sandstorms. The squally wind swept the entire planet with dense iron oxide dust, obscuring the sky, making this red planet not a single one. A clean place.



In fact, the brother of the American “Opportunity” Mars rover, the “Courage” is also due to lack of electricity.
He was "hibernating" when he was full, and finally waited three poles in the day and still couldn't charge up, and finally he was announced to give up treatment. From the previous self-portraits of the “Valor”, we can see that its solar cells were covered with Martian dust. Fortunately, it later encountered a Martian tornado. The strong wind blew the dust clean and immediately resurrected with blood. NASA calls this a "cleaning event."



During the period when the "Courage" and "Opportunity" Mars probes lost power, NASA of the United States sent signals to them many times in an attempt to wake them up. The final result was that the "Courage" had to give up because it drove into a pit and its terrain was relatively flat. The ashes on the solar cells could not be blown away, and it could not absorb enough sunlight. In the end, it had to give up. The experience of "Opportunity" is also roughly the same.

Study Task

Task 1: Assemble a "Mars Rover" breathing light
Task 2: Write a program to complete the function of the Mars rover breathing light

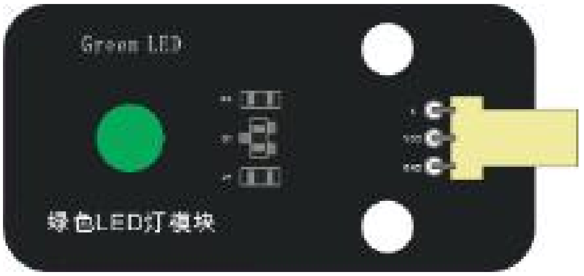
Knowledge Point

How to assemble a "Mars Rover" breathing light?

1. The principle of the "Mars Rover" breathing light

Breathing light means that the light changes gradually from bright to dark under the control of a microcomputer, and it feels like a person is breathing. It is widely used in mobile phones and has become one of the selling points of new mobile phones of major brands, acting as a notification reminder.
Pulse width modulation (PWM) is an analog control method that modulates the bias of the transistor base or the MOS transistor gate according to the change of the corresponding load to realize the change of the transistor or MOS transistor conduction time, thereby realizing the switching stabilized power supply Changes in output. This method can keep the output voltage of the power supply constant when the working conditions change, and it is a very effective technology to control the analog circuit with the digital signal of the microprocessor. Pulse width modulation is a very effective technology that uses the digital output of a microprocessor to control analog circuits. It is widely used in many fields from measurement and communication to power control and conversion.
middle.

New electronic components: LED lights, also known as light-emitting diodes, are solid-state semiconductor devices that can convert electrical energy into visible light, which can directly convert electricity into light. The heart of the LED is a semiconductor chip, one end of the chip is attached to a bracket, one end is the negative pole, and the other end is connected to the positive pole of the power supply, so that the entire chip is encapsulated by epoxy resin.

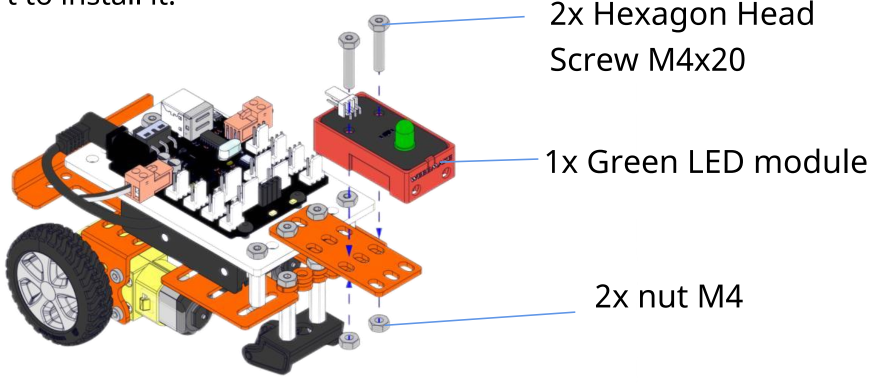


Practice

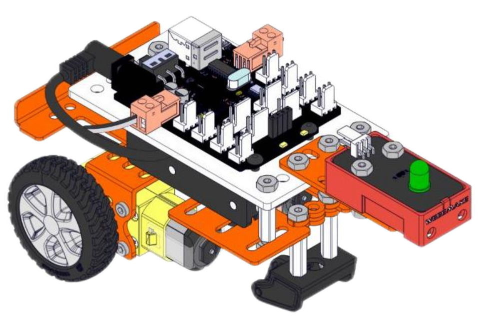
Students, through the above learning, Now install the green LED lights (yellow and red LEDs are also available) to the rover.

LED light installation steps

The assembly method of the Mars rover breathing light is divided into two steps. Students can refer to the figure below to complete the assembly. It takes about 5 minutes. Next, we will start to install it.



1. Install LED lights



2. Install the LED light, and connect the LED light to the digital port No. 3

Notice:

The first point: The wiring can only be connected to the pins corresponding to the 3rd and 11th digital ports (PWM pins).

The second point: When assembling the wiring, be sure to insert the wiring according to the wiring direction, remember not to insert it backwards, and do not use metal devices to contact the wiring pins.

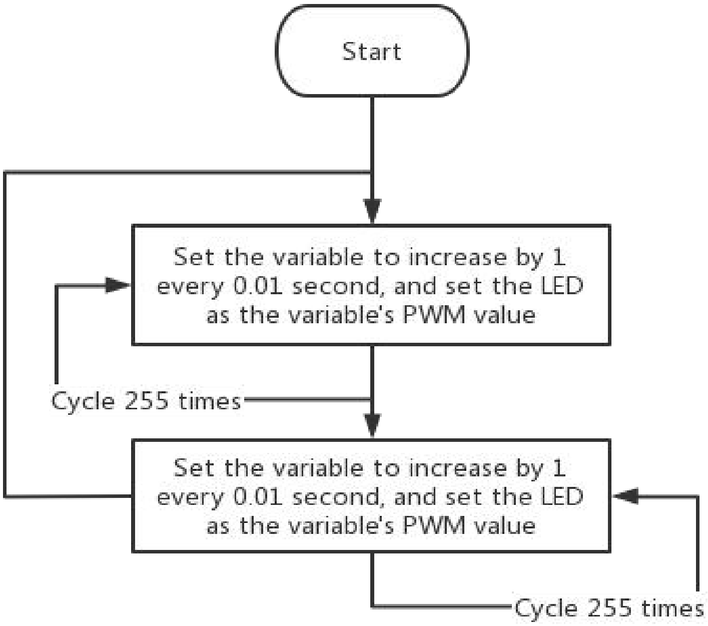
Check the connection status of the rover

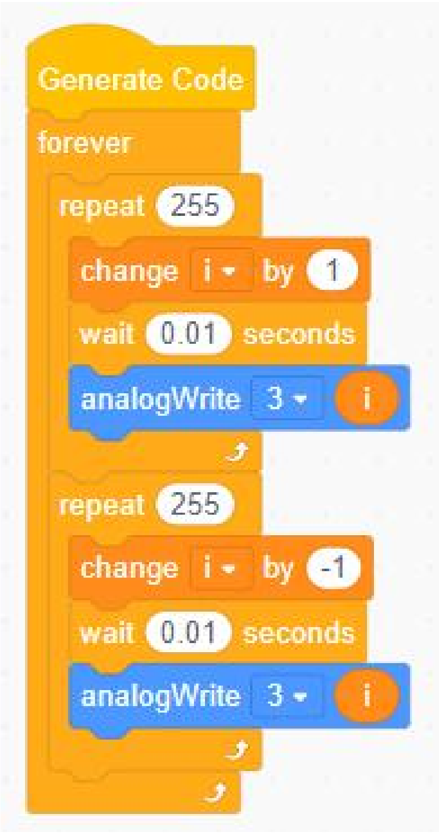
1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted;
2. Data cable connection test, check whether the data cable is plugged in firmly.

Program the breathing light of the Mars rover

The program logic of the "Mars Rover" breathing light is a piece of code that is executed sequentially. First of all, we need to know what kind of signal the PWM outputs.

The PWM output is a continuously changing level signal, the signal range is 0-255, the PWM signal output can be completed by the following program block, the higher the PWM signal, the brighter the light, and vice versa.



<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>
</td><td>

The variable decrease by 1 can be understood as the variable increase by -1, which runs every 0.01 second, so 2.55 seconds can realize the process from dark to light.</td></tr>
</table>

```
#include<Arduino.h>

double _i = 0;        //i;

void setup(){
                        pinMode(3, OUTPUT);
}

void loop(){
      for(int i=255; i>0; --i){
           _i += 1;
           delay(10);
           analogWrite(3, _i);
      }
       for(int i=255; i>0; --i){
           _i += -1;
           delay(10);
           analogWrite(3, _i);
      }
}
```

Classmates, let's do it quickly, make a Mars rover with breathing light.

?

Where have you seen breathing lights? Try to make a breathing traffic light, come on, you are the best!

Lesson Sixteen Traffic lights of Mars rover

Classmates, do you know how far is from the earth on Mars? The distance between the earth and Mars is 0.55~400 million kilometers. The short distance is about 55 million kilometers, and the longest distance is more than 400 million kilometers. Close contact between the two occurs approximately once every 15 years. In 1988, the distance between Mars and the Earth reached about 58.8 million kilometers, and in 2018 the distance between the two reached 57.6 million kilometers.

The distance between the Earth and Mars, even if the electromagnetic wave speed of light round-trip communication takes more than 45 minutes, which is enough for the speed of light to complete 20,000 circles around the earth.

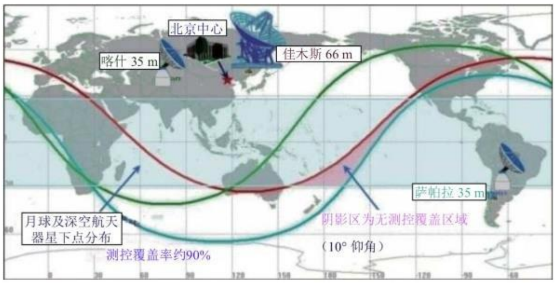
The speed of light is about 300,000 kilometers per second, which is the propagation speed of electromagnetic waves and the fastest speed that humans can touch and master. It takes only 0.13 seconds to go around the earth, and humans cannot even detect the weak time delay of electromagnetic waves when using communication equipment.

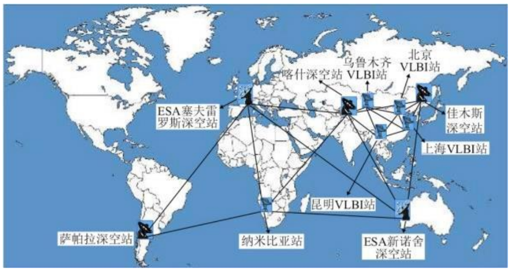
How to accurately get the signal from the Mars rover back to Earth?

1. Large-scale deep-space communication antenna

"Tianwen No. 1" weighs about 5 tons, but it needs the Long March 5, which weighs 870 tons, to be launched. It also includes three parts: orbiter, lander and patrol, and carries 13 scientific research instruments. The orbiter is the core of it. It carries important components such as the energy system, propulsion system, attitude control system, support structure and communication system, including the communication antenna;

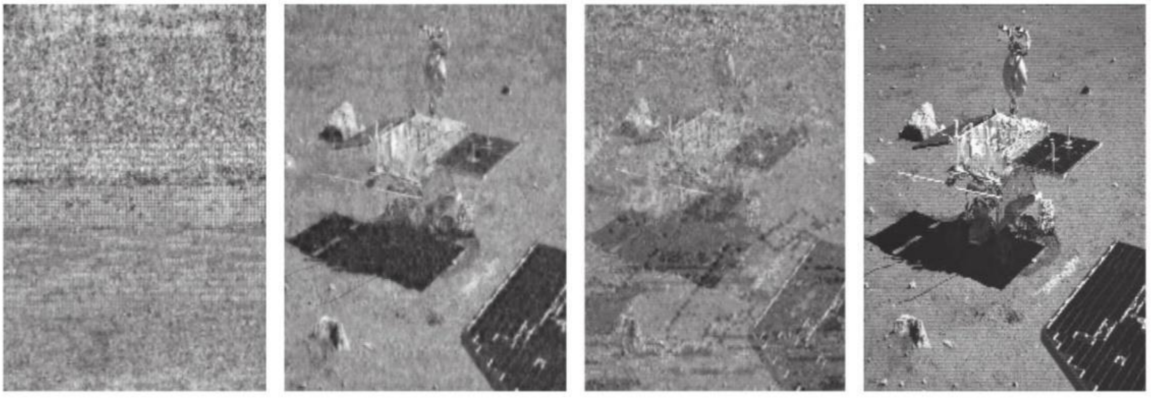
2. Global deep space communication network without dead ends





China’s plan is to deploy deep space stations in Jiamusi (the easternmost part of the country), Kashgar (the westernmost part of the country), and Zapala (western Argentina), which are sufficient to achieve effective communication with more than 92% of the sky area. Other stations can also participate in measurement and control communications and Payload scientific research data reception, etc. In addition, China's deep space communications network is also actively participating in international networking, using programs such as very long baseline measurements of a huge network to track the position of Tianwen No. 1 with high accuracy. China is currently the third country with a global deep-space communications network after the United States and Europe;

3. Signal transmission and processing technology upgrade



With more and more functions of deep space probes and more complex payloads, an intuitive manifestation is that the amount of data that needs to be sent back to the earth has increased sharply, but the signal strength has become weaker. Can not solve the problem fundamentally. Therefore, it is necessary to study multiple antennas to form a large array, and use complex data collaborative fusion analysis technology to achieve the effect of a huge size antenna and improve the overall receiving performance.

Study Task

Task 1: Assemble a "Mars Rover" traffic light
Task 2: Write a program to complete the functions of red light stop, green light line, and yellow light deceleration

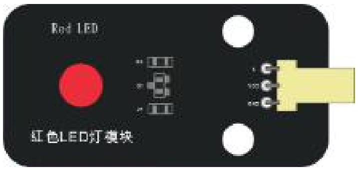
Knowledge Point

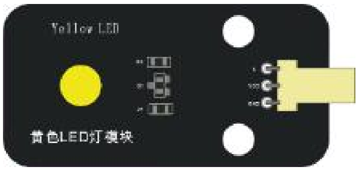
How to assemble the "Mars Rover" traffic light?

1. Structural analysis of "Mars Rover" traffic lights

Classmates, after a few lessons, I believe you must be familiar with the rover. Do you remember the names of the connection structures and sensors of the rover? Students who don't know should think about the content of the previous lessons.

New electronic components: LED lights, also known as light-emitting diodes, are solid-state semiconductor devices that can convert electrical energy into visible light, which can directly convert electricity into light. The heart of the LED is a semiconductor chip, one end of the chip is attached to a bracket, one end is the negative pole, and the other end is connected to the positive pole of the power supply, so that the entire chip is encapsulated by epoxy resin.



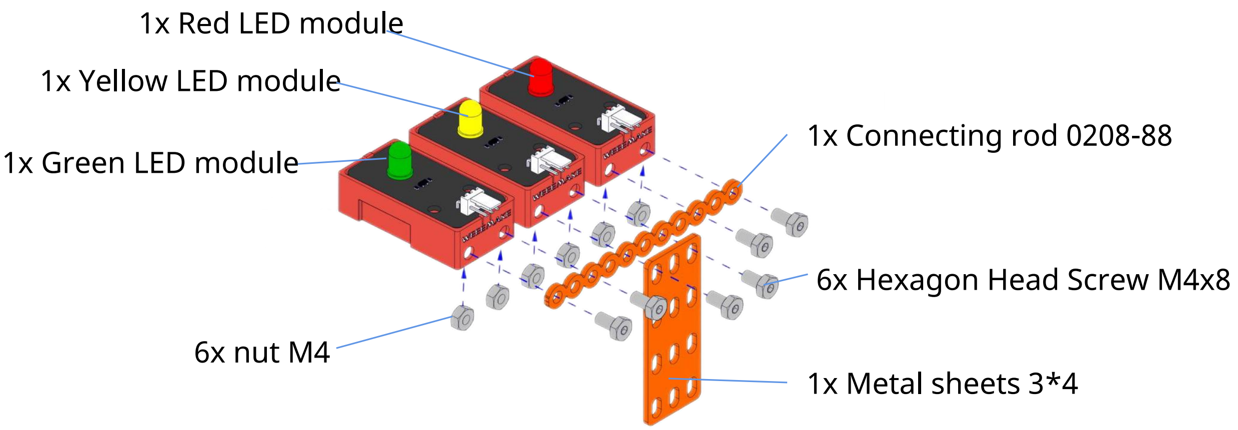


Practice

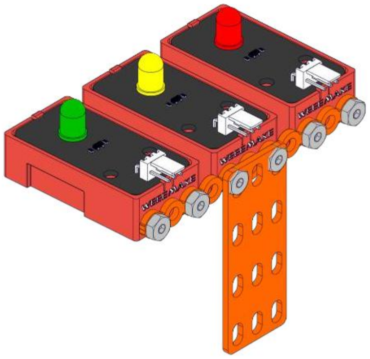
Students, through the above learning, now install the red, yellow, and green LED lights on the rover.

LED light installation steps

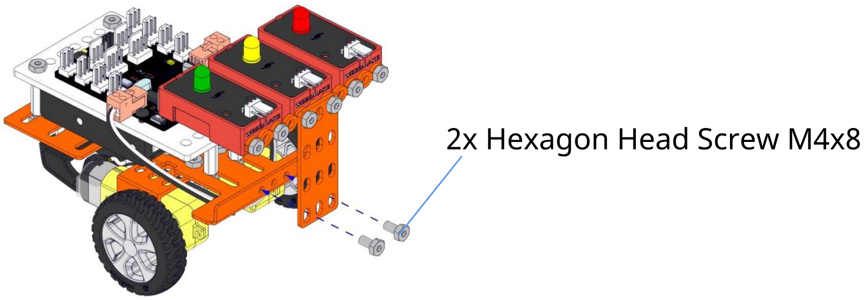
The assembly method of the Mars rover traffic light is divided into 4 steps. Students can refer to the figure below to complete the assembly. It takes about 5 minutes. Next, we will start to install it.



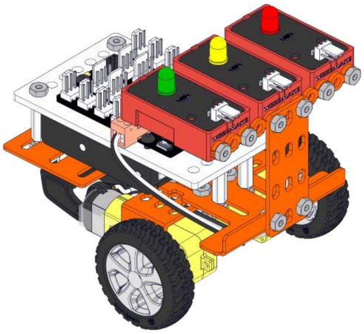
1. Install the 3 kinds of LED lights in order on the connecting rod 0209-88 and piece 3x4



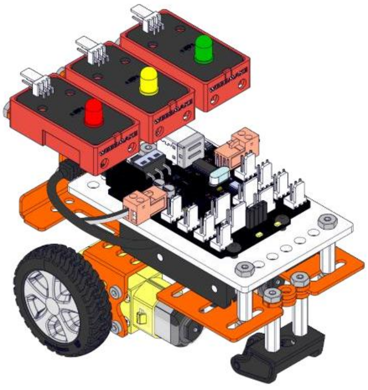
2. Install the sensor



3. Install the installed sensor in the screw hole at the rear of the car



4. Installation is complete



5. Use KF2510 3P terminal wire

①. Wire the red LED light module to the No. 3 digital port.

②. Wire the yellow LED light module to the No. 8 digital port.

③. Wire the green LED light module to the No. 7 digital port.

Notice:

The first point: When installing the structure, pay attention to the balance of the vehicle and do not obstruct the insertion of the data cable.

The second point: When assembling the wiring, be sure to insert the wiring according to the wiring direction, remember not to insert it backwards, and do not use metal devices to contact the wiring pins.

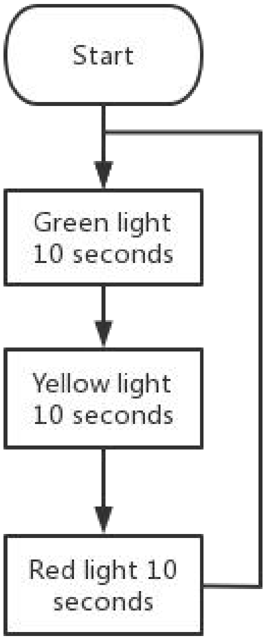
Check the connection status of the rover

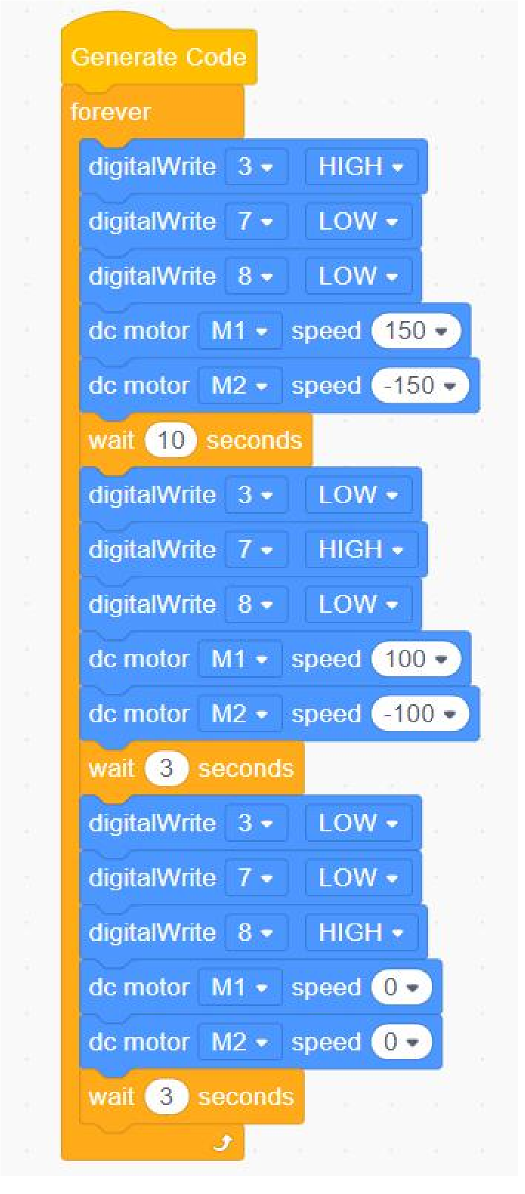
1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted.
2. Data cable connection test, check whether the data cable is plugged in firmly.

Programming the "Mars Rover" traffic light:

The program logic of the “Mars Rover” traffic lights is a piece of code that is executed sequentially. First, we determine how long the green, yellow, and red lights are on respectively. Here we assume that the green and red lights are on for 10 seconds, and the yellow light is on for 3 seconds.

Note: The green light is connected to the D7 digital port, the yellow light is connected to the D8 digital port, and the red light is connected to the D3 digital port.



<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>


</td><td>


Set the green light to high level, and the other lights to low level. Set the motor speed to 150. The yellow light behind can be slightly slower, and the duration is 10 seconds.
</td></tr>
</table>

```
#include<Arduino.h>

void setup(){
      pinMode(3, OUTPUT);
      pinMode(4, OUTPUT);
      pinMode(7, OUTPUT);
      pinMode(8, OUTPUT);
}

void loop(){
      digitalWrite(3, HIGH);
      digitalWrite(7, LOW);
      digitalWrite(8, LOW);
      {
           int16_t speed = 150;
          digitalWrite(4, speed >= 0 ? LOW : HIGH);
          analogWrite(5, speed >= 0 ? speed : -speed);
      }
      {
          int16_t speed = -150;
         digitalWrite(7, speed >= 0 ? LOW : HIGH);
         analogWrite(6, speed >= 0 ? speed : -speed);
      }
      delay(10000);
      digitalWrite(3, LOW);
      digitalWrite(7, HIGH);
      digitalWrite(8, LOW);
      {
           int16_t speed = 100;
          digitalWrite(4, speed >= 0 ? LOW : HIGH);
          analogWrite(5, speed >= 0 ? speed : -speed);
       }
       {
          int16_t speed = -100;
          digitalWrite(7, speed >= 0 ? LOW : HIGH);
          analogWrite(6, speed >= 0 ? speed : -speed);
        }
        delay(3000);
        digitalWrite(3, LOW);
        digitalWrite(7, LOW);
        digitalWrite(8, HIGH);
        {
             int16_t speed = 0;
            digitalWrite(4, speed >= 0 ? LOW : HIGH);
            analogWrite(5, speed >= 0 ? speed : -speed);
        }
        {
             int16_t speed = 0;
            digitalWrite(7, speed >= 0 ? LOW : HIGH);
            analogWrite(6, speed >= 0 ? speed : -speed);
         }
        delay(3000);
}
```

Classmates, let's get your hands dirty and make a Mars rover that can move according to traffic lights.

?

The traffic light system in reality is more complicated, so do you know how to realize the mutual cooperation of multiple traffic light systems?

Lesson Seventeen Obstacle Avoidance

Students, can ultrasonic sensors be used on Mars? Ultrasonic sensors are sensors that convert ultrasonic signals into other energy signals (usually electrical signals). Ultrasound is a mechanical wave with a vibration frequency higher than 20kHz. It has the characteristics of high frequency, short wavelength, small diffraction phenomenon, especially good directivity, which can become rays and propagate directionally. Ultrasound has a great ability to penetrate liquids and solids, especially in solids that are opaque to sunlight. When the ultrasonic wave hits the impurity or the interface, it will produce a significant reflection to form a reflected echo, and it can produce a Doppler effect when it hits a moving object. Ultrasonic sensors are widely used in industry, national defense, biomedicine, etc.

According to the working principle of ultrasonic, the air on the rover is thin, which is not conducive to ultrasonic work. Most real rover uses visual recognition to avoid obstacles.

There are 7 instruments on the orbiter of the Zhurong rover: medium-resolution camera, high-resolution camera, orbiter subsurface radar, Martian mineral spectrometer, Martian magnetometer, Martian ion and neutral particle analyzer, and Martian energy particles Analyzer. The six instruments on the rover: multispectral camera, terrain camera, rover surface radar, Mars surface composition detector, Mars magnetic field detector, and Mars meteorological instrument.

Study Task

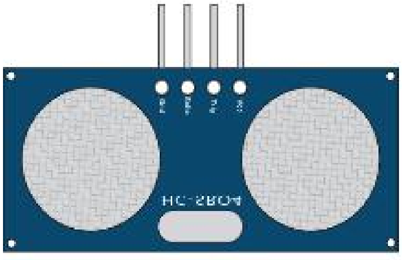
Task 1: Assemble a "Mars Rover" that can avoid obstacles ultrasonically
Task 2: Let the computer correctly read the ultrasonic data on the rover
Task 3: Compile the program and complete the control of the ultrasonic obstacle avoidance Mars rover

Knowledge Point

How to assemble an ultrasonic obstacle avoidance "Mars Rover"?

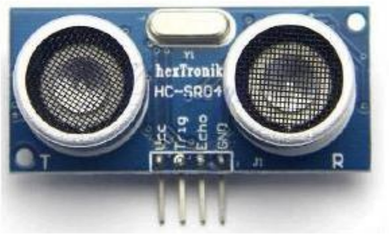
1. Structural Analysis of Ultrasonic Obstacle Avoidance Mars Rover

Learn about electronic components: Ultrasound is a kind of mechanical oscillation in elastic medium. There are two forms: lateral oscillation (transverse wave) and longitudinal oscillation (longitudinal wave). Longitudinal oscillation is mainly used in industrial applications. Ultrasonic waves can propagate in gases, liquids and solids, and their propagation speeds are different. In addition, it also has refraction and reflection phenomena, and attenuation during propagation. The frequency of ultrasonic waves propagating in the air is low, generally tens of KHz, while in solids and liquids, the frequency can be higher. It attenuates faster in the air, but spreads in liquids and solids, the attenuation is small, and the spread is far



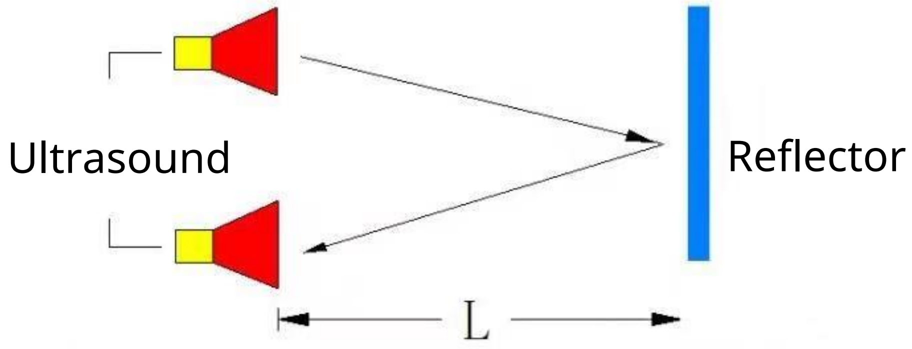
2. Important notes when assembling

The first point:: Accurate installation.
When installing the ultrasonic sensor, you must remember to pay attention to the correct pins when installing. If it is wrong, it is easy to burn the ultrasonic sensor and cause a short circuit.



The second point: Detection distance

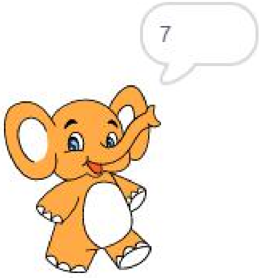
When the ultrasonic ranging accuracy is required to reach 1mm, the ambient temperature of ultrasonic propagation must be taken into consideration. For example, when the temperature is 0°C, the ultrasonic velocity is 332m/s, and at 30°C it is 350m/s, and the ultrasonic velocity change caused by the temperature change is 18m/s. If the ultrasonic measurement is performed at a sound speed of 0°C in an environment of 30°C, the measurement error caused by a distance of 100m will reach 5m, and the measurement error of 1m will reach 5cm.



The third point: The "controller" reads the test data of the "ultrasonic sensor"

For online control of the main controller, learn to read the relevant information, data and data of each sensor on the computer, and control the car. When the "Mars Rover" is successfully connected to the computer, it can read the sensor information, data and information connected to the "Mars Rover" and control the output device to work.

When the preparation work is completed, we can use the "speaking" method to read the value of the "ultrasonic" sensor, as shown in the figure.



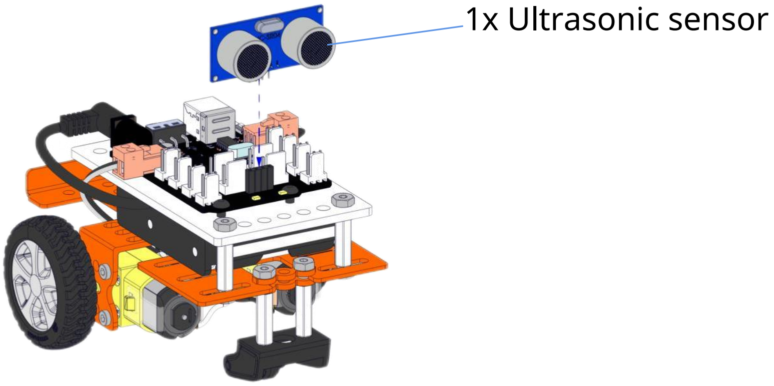
The value returned by the ultrasonic sensor is cm, but there is a slight error. The detection distance is between 5-500 cm. The farther the object is detected, the greater the error.
Due to the strong directivity of ultrasonic waves, slow energy consumption, and long distances in the medium, ultrasonic waves are often used for distance measurement. For example, rangefinders and level measuring instruments can be realized by ultrasonic waves. Ultrasonic detection is often quick, convenient, simple to calculate, easy to achieve real-time control, and can meet industrial practical requirements in terms of measurement accuracy, so it has also been widely used in the development of mobile robots.

Practice

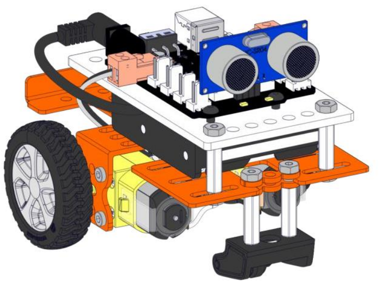
Students, after the above learning, let's install the ultrasonic sensor on the rover now.

Ultrasonic sensor installation steps

The standard assembly method of the ultrasonic obstacle avoidance rover is divided into two steps. Students can refer to the figure below to complete the assembly. It takes about 2 minutes. Next, we will start to install it.



1. Install the ultrasonic sensor



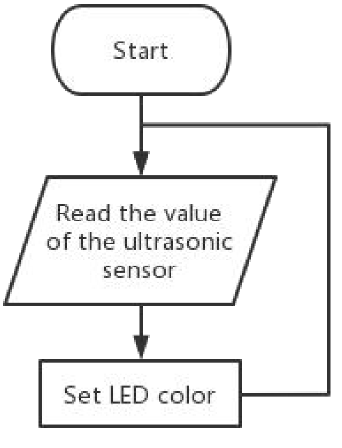
2. The installation of the ultrasonic sensor is complete

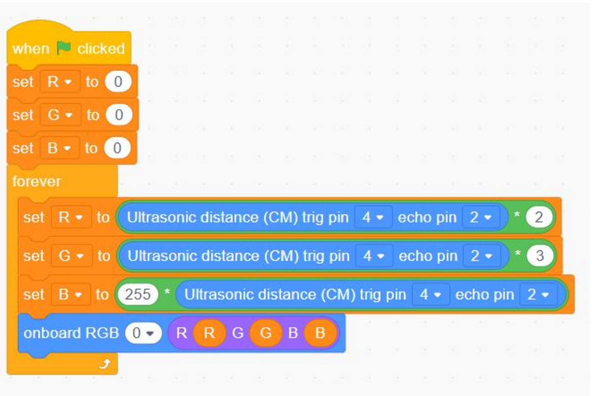
Notice:
When installing the ultrasonic sensor, do not press down hard, and do not shake the sensor back and forth to prevent the base from falling.

Check the connection status of the rover

1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted.
2. Data cable connection test, check whether the data cable is plugged in firmly.

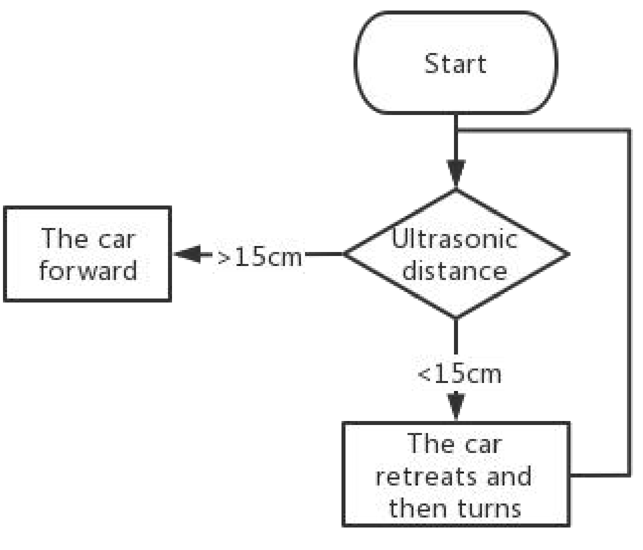
Make a simple programming: use ultrasonic to control car LED lights to change colors
Note that when writing programs, pay special attention to which digital port pins "trig" and "echo" are connected to.

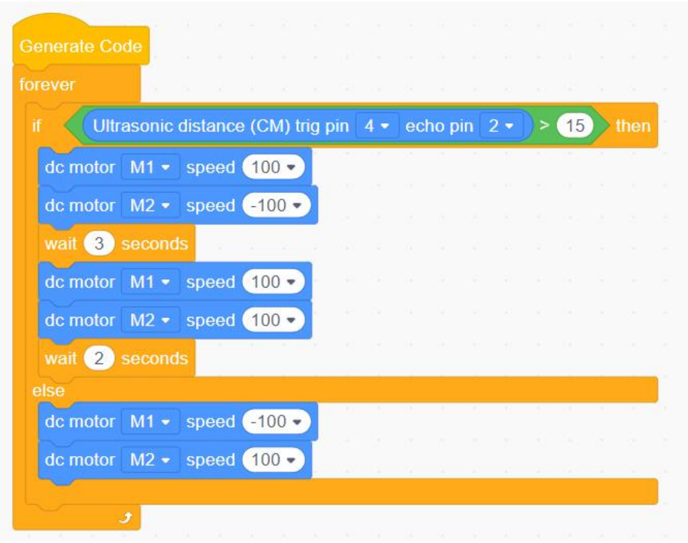
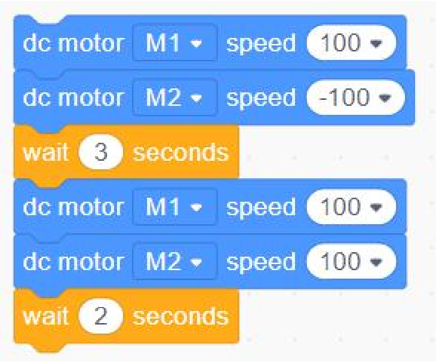


Program structure	Program description
	Set the three variables of R, G, B, and set the variables to 2 times, 3 times of the ultrasonic distance and the maximum value of the color minus the distance of the ultrasonic sensor. This setting is for each distance to be a different light color .

Programming the ultrasonic obstacle avoidance Mars rover

The program logic of "Ultrasonic Obstacle Avoidance Mars Rover" only needs to know the threshold value before the vehicle hits the obstacle, and then the program can be written by controlling the movement state of the vehicle.



Program structure	Program description
	 The motor on the left side of the rover is connected to M1, and the motor on the right is connected to M2. When M2 rotates forward and M1 reverses, the rover is in a forward state; when encountering obstacles, the vehicle will retreat for 3 seconds and then turn left for 2 seconds. Used to avoid obstacles.

```
#include<Arduino.h>

float ultrasonic_distance(int pinTrig, int pinEcho);

void setup(){
	pinMode(2, INPUT);
	pinMode(4, OUTPUT);
	pinMode(5, OUTPUT);
	pinMode(6, OUTPUT);
	pinMode(9, OUTPUT);
	pinMode(10, OUTPUT);
}
void loop(){
	if(ultrasonic_distance(4, 2) > 15){
	    {
	        int16_t speed = 100;
	        analogWrite(5, speed >= 0 ? 0 : -speed);
	        analogWrite(6, speed >= 0 ? speed : 0);
	    }
	    {
	        int16_t speed = -100;
	        analogWrite(9, speed >= 0 ? 0 : -speed);
	        analogWrite(10, speed >= 0 ? speed : 0);
	    }
	    delay(3000);
```

```
                {
                                int16_t speed = 100;
                                analogWrite(5, speed >= 0 ? 0 : -speed);
                                analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                                int16_t speed = 100;
                                analogWrite(9, speed >= 0 ? 0 : -speed);
                                analogWrite(10, speed >= 0 ? speed : 0);
                }
                delay(2000);
}else{
                {
                                int16_t speed = -100;
                                analogWrite(5, speed >= 0 ? 0 : -speed);
                                analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                                int16_t speed = 100;
                                analogWrite(9, speed >= 0 ? 0 : -speed);
                                analogWrite(10, speed >= 0 ? speed : 0);
                }
  }
}

float ultrasonic_distance(int pinTrig, int pinEcho){
    digitalWrite(pinTrig, LOW);
    delayMicroseconds(2);
    digitalWrite(pinTrig, HIGH);
    delayMicroseconds(10);
    digitalWrite(pinTrig, LOW);
    float distance = pulseIn(pinEcho, HIGH) / 58.0;
    delay(10);
    return distance;
}
```

Classmates, let's do it now, can you avoid obstacles? If not, how should the vehicle be adjusted?
Note: The waiting time in the case is only for reference, students can adjust it according to the actual situation.

?

Test under what circumstances the ultrasound cannot be used?

Lesson Eighteen Swing to preventing quicksand

Students, what is a "line-following sensor"? In the skills of controlling the movement of the car, it often makes
Using an electronic component-"line patrol sensor", it allows the car to drive along the black line, like a loyal patrolman, patrol every place in an orderly manner. In this lesson, we will learn the line patrol sensor and let the car patrol the line!
Quicksand, to put it simply, is sand that can flow like a liquid, that is, sand that can flow. Quicksand is a natural phenomenon. It often appears in deserts with unstable foundations. When a heavy object is placed on the sand body, it sinks to the bottom just like sinking to the bottom.

Study Task

Task 1: The working principle of the "line-following sensor"
Task 2: Line patrol programming task

Knowledge Point

The working principle of the "line-following sensor"

The line-following sensor is an electronic component that can identify the reflective intensity of the surface of an object (but cannot identify the color). It can be used to identify the strong reflective color and the weak reflective color when the indoor ambient light interference is not very serious.
The working principle of the line-following sensor is: the infrared light-emitting diode emits a beam of invisible infrared light and illuminates the objects with different reflective intensity. Because different colors have different absorption effects on light (in general, the darker the color, the absorption The more, the lighter the color, the less the absorption), resulting in different strengths of the reflected light, and the reflected light is received by the receiving end and converted into electrical signals through light wave signals, as shown in the figure：

The transmitting end   Receiving end

Points to note when using:
Normally, when the line-following sensor is 1-1.5 cm away from the reflector, the effect is good. In the case of strong outdoor light, when the photoelectric sensor is 0.5-0.8 cm away from the reflector, the effect can be better.

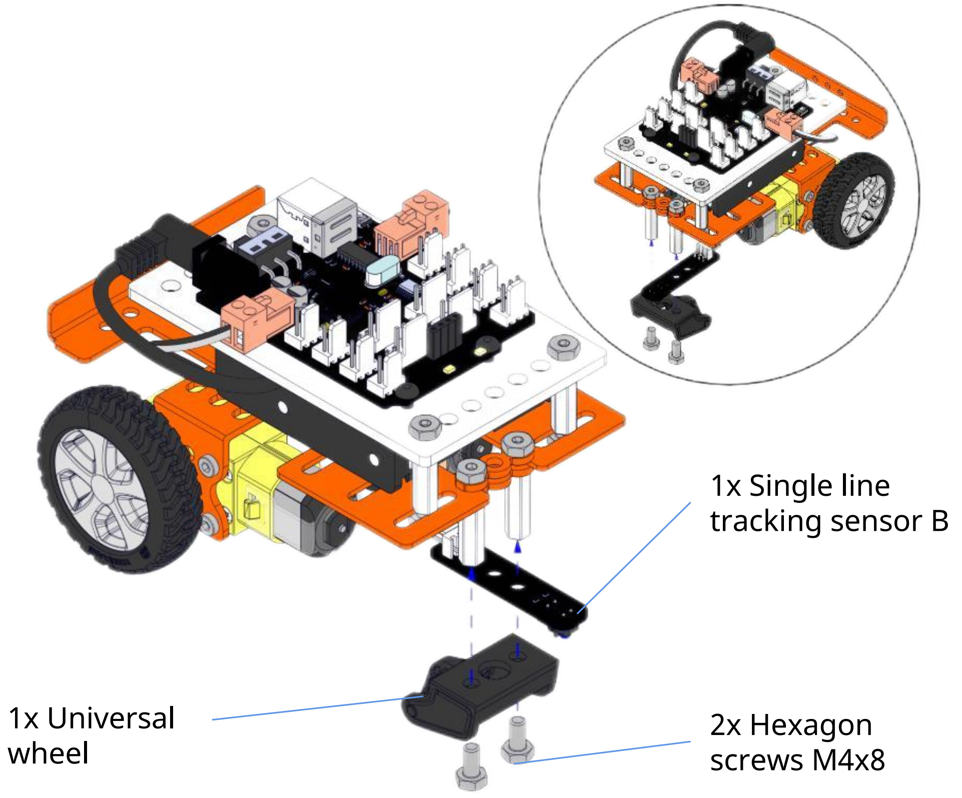
Practice

Preparation of the car before programming

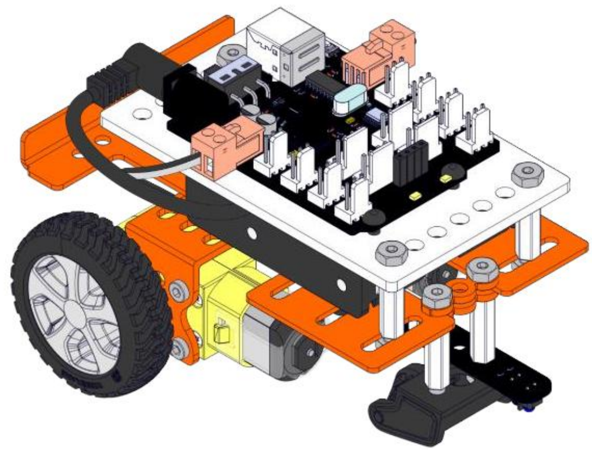
Before programming, check that the motor and photoelectric interface must be connected correctly and securely. There are two points to note:

1. The installation method of Mars rover

Please refer to Lesson 9 for the installation of the main part of the vehicle. When installing the single-channel tracking sensor, remember not to use too much force when screwing the screw to prevent damage to the sensor.



1. Remove the injection molding universal wheel,
and install the line-following sensor on one of the studs

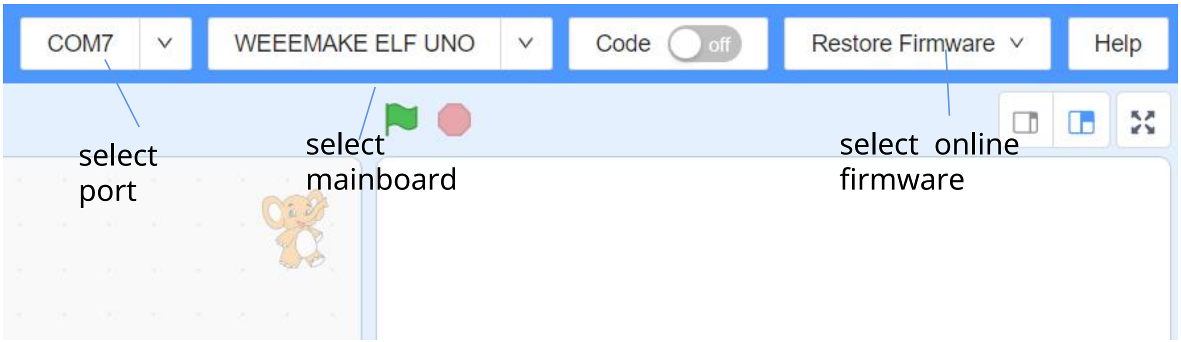


2. Use the KF2510 3P terminal wire
to wire the single-channel patrol sensor B to the A1 interface.

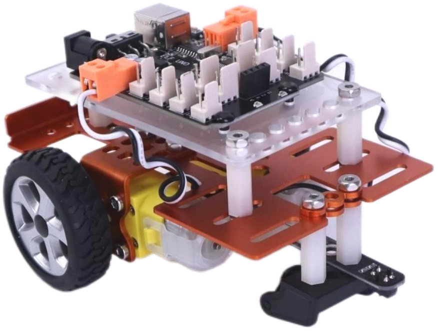
2. Check the field

connect the line-following sensor to port A1 of the analog sensor, and test the reflectance intensity of the field through the online mode. The steps are as follows.

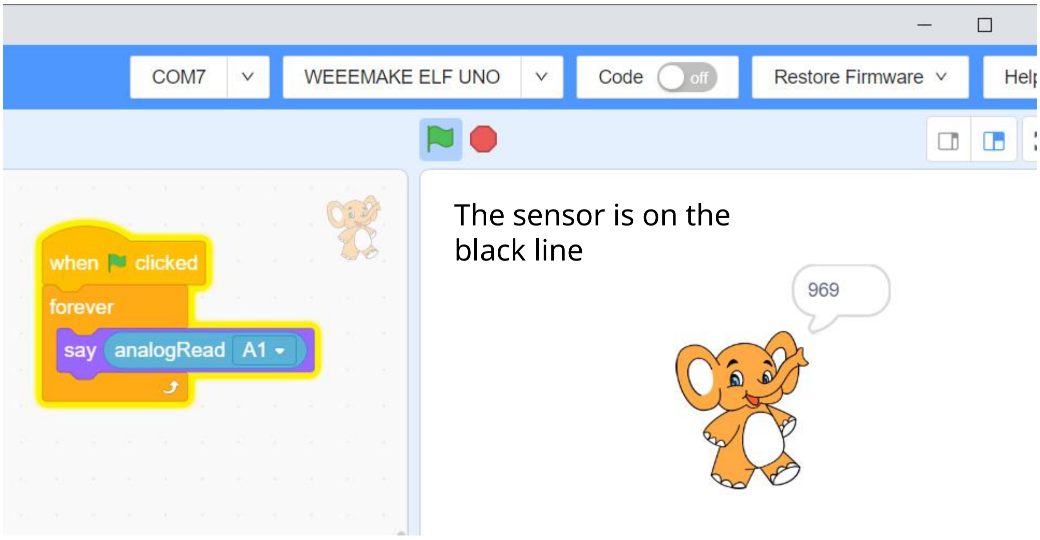
① Connect the computer and the Mars rover, after connecting the serial port, install the online mode firmware;

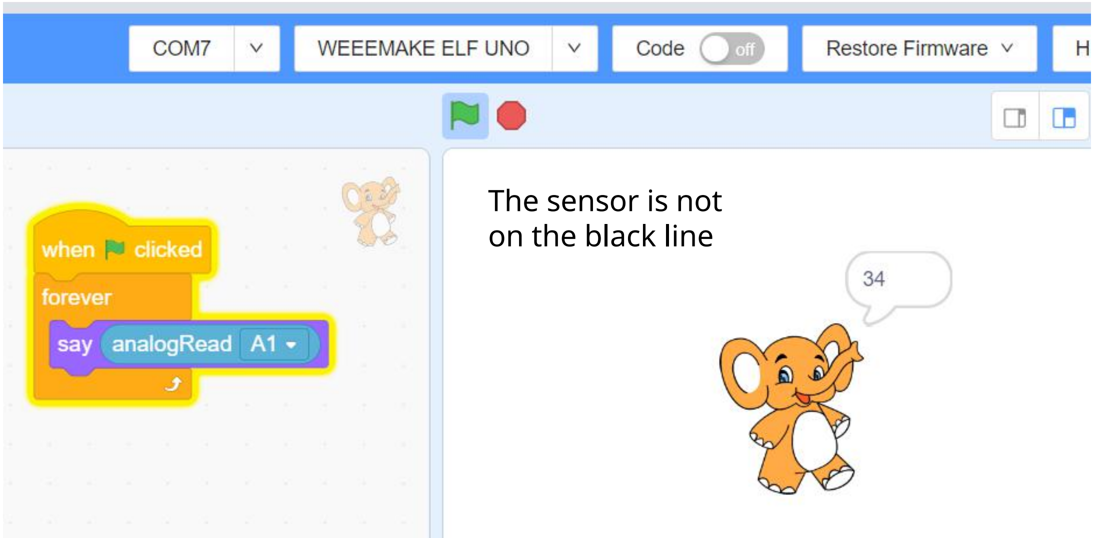


② Read the sensor value to obtain a stable and certain threshold.



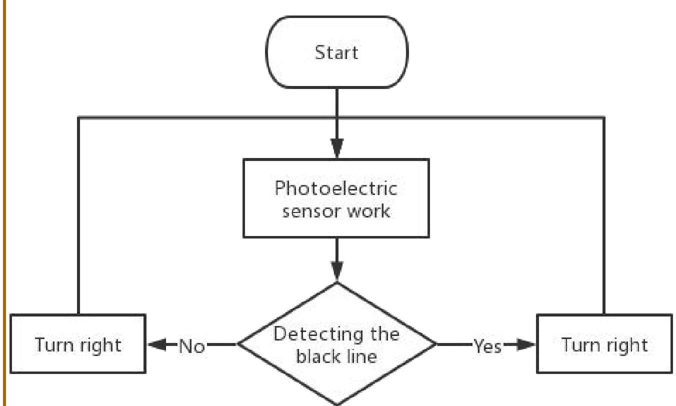
Read the sensor value online to determine the threshold.



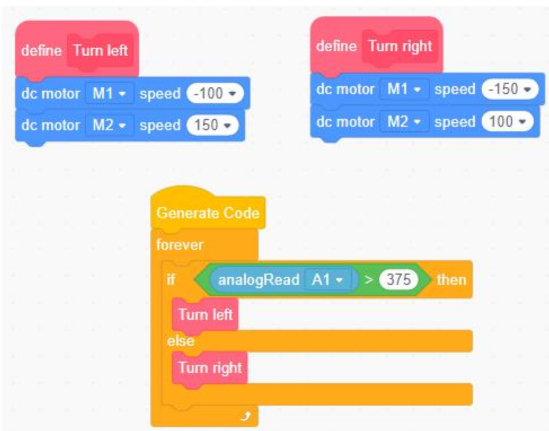


The value returned by the sensor on the black line is about 900, and the return value on the white desktop is about 50, the determination threshold is: (900+50)/2=475

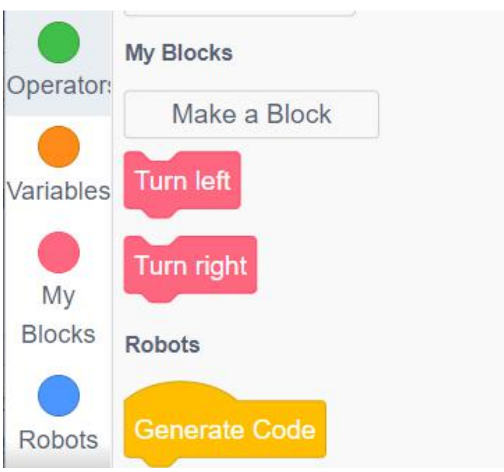
Programming ideas

Analysis task	flow chart
1. The car needs to install a line-following sensor; 2. It needs to be programmed and controlled to let the car go along the black line. 3. Principle: Use the line-following sensor to judge the black line on the road. Use the program to make the car hit the black line and turn right, otherwise it will turn left, so that the car will turn and go forward, but pay attention to when turning left and right. , The left and right motor speed settings, students can explore.	

Analyze programming tasks and determine script structure.

<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>
</td><td>Task 1: Use a photoelectric sensor to make the car patrol the route

Task 2: Set the rotation speed of the left and right motors

Task 3: Note that the speed should be adjusted appropriately according to the battery power</td></tr>
<tr><td>
</td><td>Using self-made building blocks To make the main program look clearer, we can add a self-made building block (same function as a sub-function), which means "packing" a function block program together to facilitate the call of the main program.</td></tr>
</table>

```
#include<Arduino.h>

//    Turn left()
void _Turn_left();
//    Turn right()
void _Turn_right();

void setup(){
      pinMode(5, OUTPUT);
      pinMode(6, OUTPUT);
      pinMode(9, OUTPUT);
      pinMode(10, OUTPUT);
      pinMode(A1, INPUT);
}

void loop(){
       if(analogRead(A1) > 375){
            _Turn_left();                         //Turn left
       }else{
            _Turn_right();                        //Turn right
       }
}
//    Turn left()
void _Turn_left(){
      {
             int16_t speed = -100;
            analogWrite(5, speed >= 0 ? 0 : -speed);
            analogWrite(6, speed >= 0 ? speed : 0);
       }
       {
             int16_t speed = 150;
             analogWrite(9, speed >= 0 ? 0 : -speed);
             analogWrite(10, speed >= 0 ? speed : 0);
        }
}

//    Turn right()
void _Turn_right(){
      {
             int16_t speed = -150;
            analogWrite(5, speed >= 0 ? 0 : -speed);
            analogWrite(6, speed >= 0 ? speed : 0);
      }
      {
            int16_t speed = 100;
           analogWrite(9, speed >= 0 ? 0 : -speed);
          analogWrite(10, speed >= 0 ? speed : 0);
       }
}
```

?

After studying in this lesson, I also want to know about the line patrol car:

1. Can the line-following sensor only be used for line-following?

Lesson Nineteen Line-following (Model A)

Students, what is an "orbital rover"? That is, two line-following sensors are used to control the skill of car line-following. When the line-following sensor is used to control the line-following movement of the car, you can choose to increase or decrease the number of line-following sensors to make a smarter car. In this lesson, we will learn the technique of "two-sensor patrol" to make the car drive more intelligently!
In a real Mars expedition, the rover needs to follow a set route and avoid rocks and cliffs, so it is called a rover with an orbit.

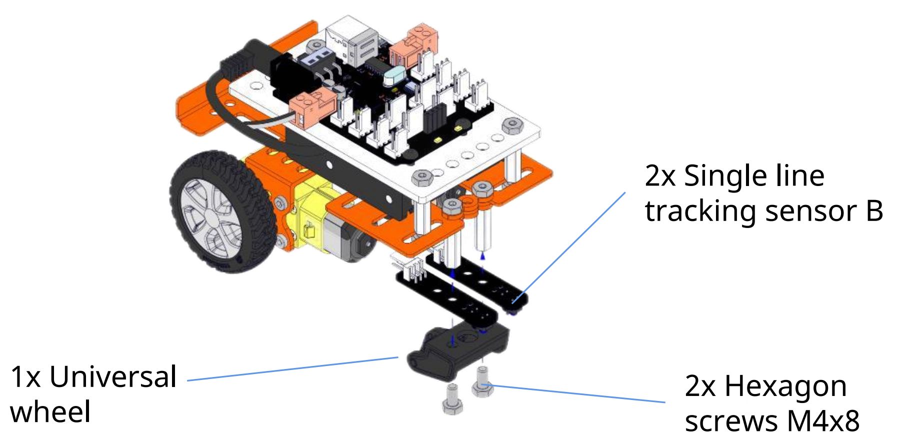
Study Task

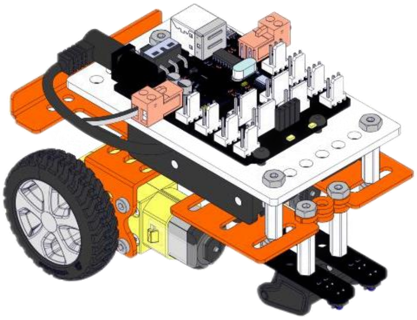
Task 1: Understand the basic actions in the two-sensor line patrol operation
Task 2: Build and use self-made blocks
Task 3: Write and execute: a program using self-made building blocks and two-sensor line tracking skills

Knowledge Point

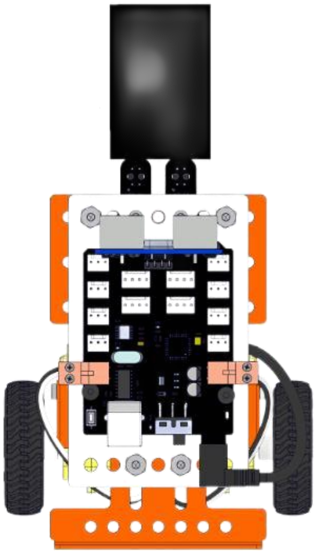
What is a two-sensor line patrol

Above the universal wheel of the "Mars Rover", two line-following sensors are placed side by side. Let them work at the same time through program control, and perform line inspection tasks together, which is a more efficient and accurate way to control the car line inspection. As shown below.



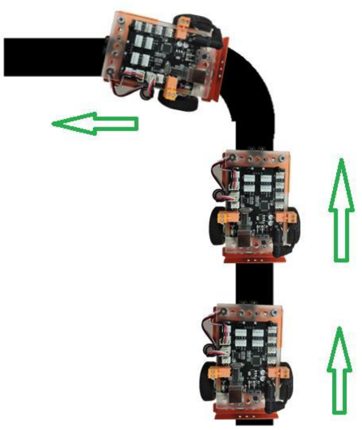
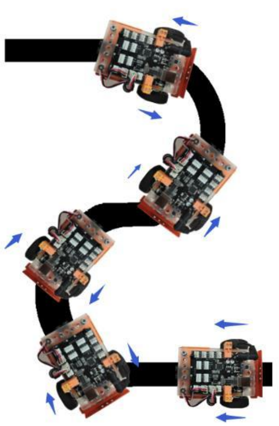


Successful installation

Two-sensor "Mars Rover"	Notice
 When viewed from above, the front of the car is facing forward, the single-channel patrol sensor B on the left is connected to the analog port A1, and the single-channel patrol sensor B on the right is connected to the analog port A2.	Students, when using the "Mars Rover" with two sensors, you need to pay attention to: 1. In order to distinguish and use these two line-following sensors well in program control, we use the names of the analog ports they are connected to number them as A1 and A2. 2. The two line-following sensors need to work at the same time, and use them to detect the deviation of the route when pre-set the line-following, and then adjust the direction by the speed difference between the left and right driving wheels.

Basic actions in line-following operations

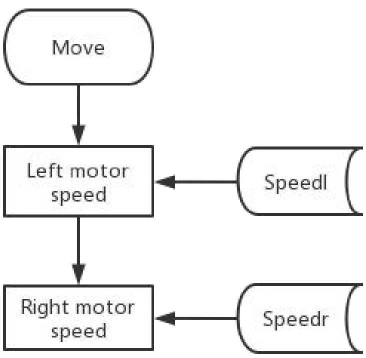
In line patrol operations, the trolley basically has two actions: going straight and turning, as shown in the figure below.
Straight travel: Use line-following sensor A1 and line-following sensor A2 to detect the route during line-following, and to monitor whether the car is deviating during line-following, and adjust the action through the differential speed of the left and right wheels.
Turning: Use the line-following sensor A1 and line-following sensor A2 to detect the route when following the line. When the car detects the turning black line, it will make a turning action and change the action through the different actions of the two driving wheels.

Motion path	Notice
1. Go straight: 	1. The movement path is straight or curved. 2. The purpose is to keep the car from deviating from the route during the action (as shown by the arrow on the left). 3. Two sensors are moving on the black line at the same time. This way of tracing is called two sensors moving on the black line at the same time. This way of patrolling is called Line-following (Model A).
2. Turn: 	1. The action path is to change direction. 2. The purpose is to make the car complete the turning action in action (as shown by the arrow on the left). 3. Need to change the action of the two driving wheels. As shown in the picture on the left, the car makes a turning action (right wheel forwards, left wheel backwards), which can prevent the car from deviating from the route a lot when turning.

Practice

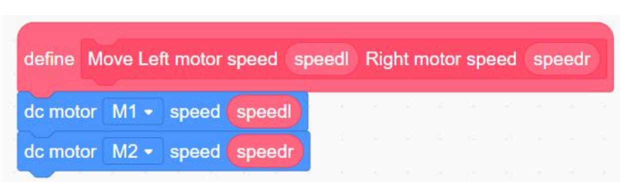
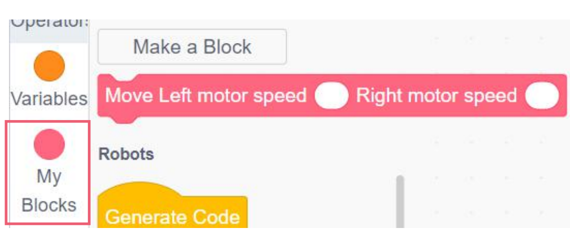
Create and use the new function

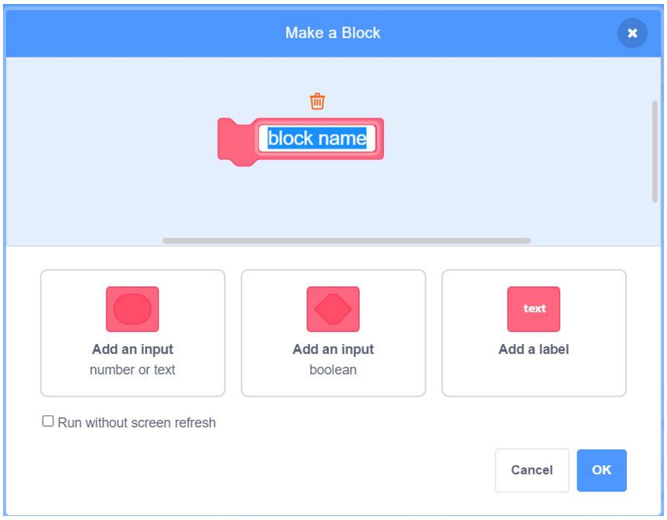
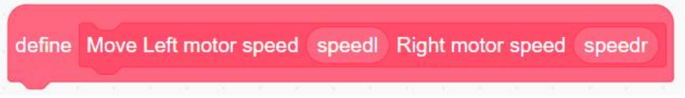
The new function is a self-built building block defined and constructed by yourself. In order to successfully master the two-sensor line tracking skills, here is a self-made building block that students need to use, as shown in the figure.

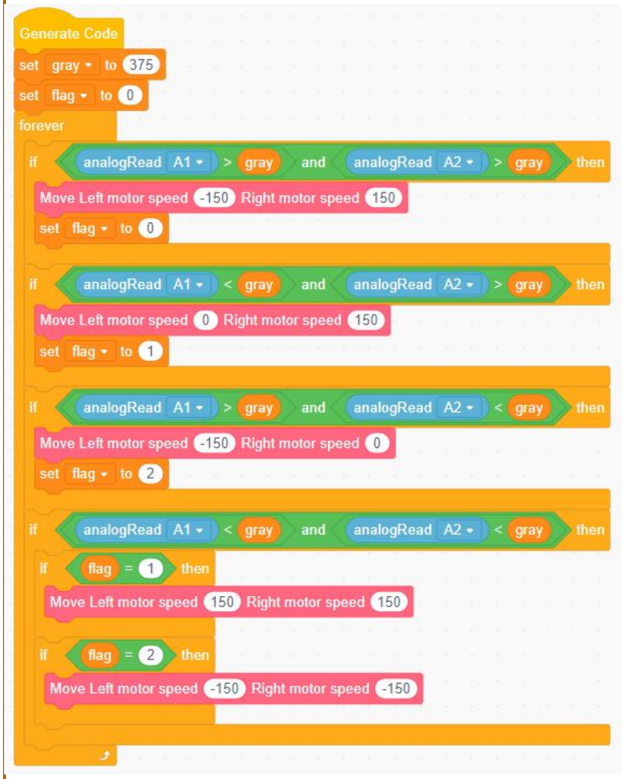
Task flow chart	Instructions for use
Self-made blocks - move: 	The speeds of the two motors M1 and M2 are defined as 'speedl' and 'speedr'. As long as the values of 'speedl' and 'speedr' are changed, the car can move forward and backward and turn left and right.

Write and execute the program.

Write the corresponding program according to the self-made building blocks, as shown in the figure.

Program structure	Program description
Move: 	**1. Create a new function** ① Choose self-made blocks to make new blocks 

Program structure	Program description
	② After the dialog box for creating a new self-made building block appears, enter the name.  ③ After entering the name of the self-made building block, start to set the variable parameters that will be used in the self-made building block, and add a text label to explain it. As shown in the figure below, create a new function called speed control. The two variables speedl and speedr are used in the function.  **2. Set the script of self-made building blocks** Let the left wheel travel at speedl and the right wheel at speedr.

Program structure	Program description
Data initialization:  	1. Create a new variable, as the gray threshold, name it "gray", and initialize it to 375; 2. Create a new variable, as the vehicle status flag, name it "flag", and initialize it to 0;
Programming:  	1. The A1 sensor is on the black line (greater than the gray threshold) and the A2 sensor is on the black line, the vehicle is moving forward, and the vehicle state variable is set to 0; 2. The A1 sensor is on the white background and the A2 sensor is on the black line, the vehicle makes a small right turn, and the vehicle state variable is set to 1; 3. The A1 sensor is on the black line and the A2 sensor is on the white background, the vehicle makes a small left turn, and the vehicle state variable is set to 2; 4. The A1 sensor is on the white background and the A2 sensor is on the white background, and then judge the state of the car at the previous moment. If the previous state is a small right turn, and this time it is a large right turn; if the previous state is a small left turn, At this time, it is a big left turn.

```
#include<Arduino.h>

//                    Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr);

double _gray = 0;  //gray;
double _flag = 0;  //flag;

void setup(){
                    pinMode(5, OUTPUT);
                    pinMode(6, OUTPUT);
                    pinMode(9, OUTPUT);
                    pinMode(10, OUTPUT);
                    pinMode(A1, INPUT);
                    pinMode(A2, INPUT);

                    _gray = 375;
                    _flag = 0;
}

void loop(){
                    if(analogRead(A1) > _gray && analogRead(A2) > _gray){
                                        _Move_Left_motor_speed(-150, 150);
                    //Move Left motor speed
                                        _flag = 0;
                    }
                    if(analogRead(A1) < _gray && analogRead(A2) > _gray){
                                        _Move_Left_motor_speed(0, 150);    //Move Left
motor speed
                                        _flag = 1;
                    }
                    if(analogRead(A1) > _gray && analogRead(A2) < _gray){
                                        _Move_Left_motor_speed(-150, 0);   //Move Left
motor speed
                                        _flag = 2;
                    }
                    if(analogRead(A1) < _gray && analogRead(A2) < _gray){
                                        if(_flag == 1){
                                                            _Move_Left_motor_speed(150,
150);               //Move Left motor speed
                                        }
                                        if(_flag == 2){
                                                            _Move_Left_motor_speed(-150, -
150);               //Move Left motor speed
                                        }
                    }
}

//                    Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr){
                    {
                                        int16_t speed = _speedl;
                                        analogWrite(5, speed >= 0 ? 0 : -speed);
                                        analogWrite(6, speed >= 0 ? speed : 0);
                    }
                    {
                                        int16_t speed = _speedr;
                                        analogWrite(9, speed >= 0 ? 0 : -speed);
                                        analogWrite(10, speed >= 0 ? speed : 0);
                    }
}
```

?

1. What kind of speed is suitable for the car to patrol the line? Are there any exceptions?

Lesson Twenty Line-following (Model B)

Classmates, we have learned about the orbital rover in the last class, and can walk along the black line, so why are we still the "orbital rover" in this class? In fact, there are many ways to patrol the line. The two-sensor patrol can be roughly divided into two types. The first one is "Line-following(Model A)", which is the knowledge point of the previous class, and the other one is studied in this class. Line-following(Model A), that is, the technique of using two line patrol sensors to "clamp" the black line to patrol the line by the trolley.
In the process of real industrial-grade robots traveling along the track, in fact, more is not to use the method of detecting black and white lines, but to use magnetic routes. The robot can perceive the changes in magnetism, so as to drive along the track.

Study Task

Task 1: Understand the second type of two-sensor line inspection——Line-following (Model B)
Task 2: Review and use self-made blocks
Task 3: Write and execute: a program using self-made building blocks and two-sensor line tracking skills

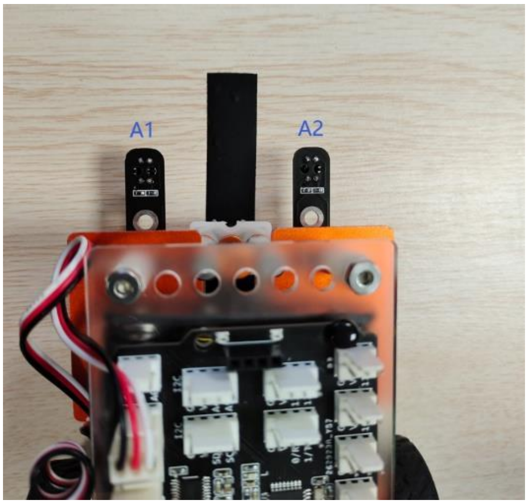
Knowledge Point

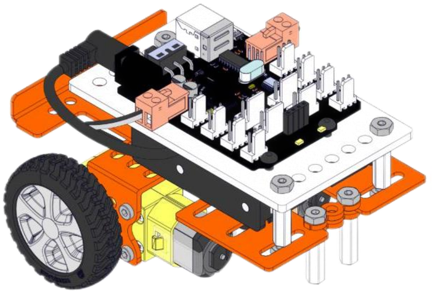
What is the Line-following (Model B)?

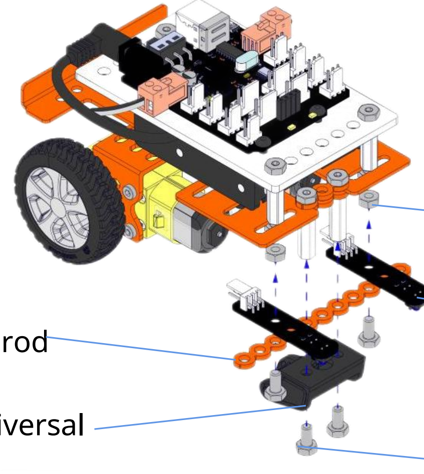
Above the universal wheel of the "Mars Rover", two line-following sensors are placed side by side. When the black line is relatively thin
At the same time, we can use Line-following (Model B) for line-following movement.
Some students may have questions, when to choose "Line-following (Model B)" and when to choose "Line-following (Model A)"?

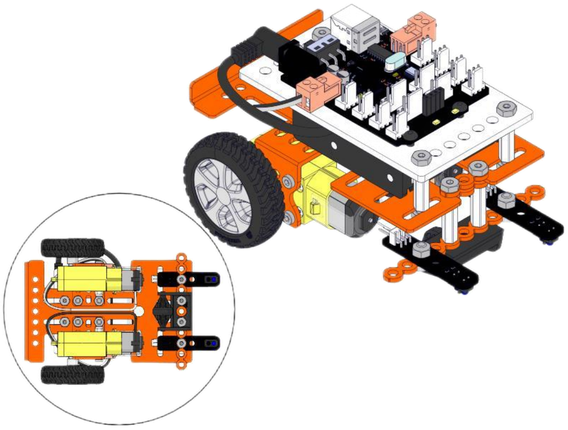
Model A is suitable for situations where the route is relatively curved and the black line is relatively thick;
Model B is suitable for the situation where the route is relatively flat and the black line is relatively thin;

In fact, most of these two line inspection methods can be exchanged with each other.

Picture	Notice
	Students, when using the "Mars Rover" with two sensors to patrol the line, you need to pay attention to: 1. In order to distinguish and use these two line-following sensors well in program control, we use the names of the analog ports they connect to number them as A1 and A2. 2. The two line-following sensors need to work at the same time, and use them to detect the deviation of the route when pre-set the line-following, and then adjust the direction by the speed difference between the left and right driving wheels.

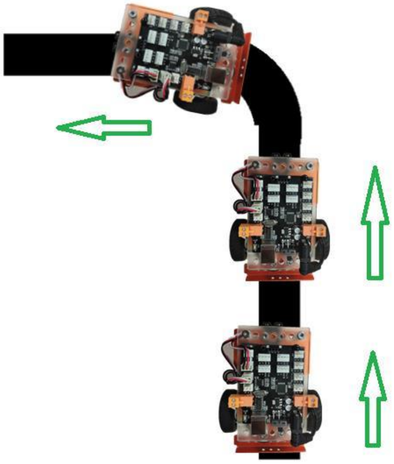


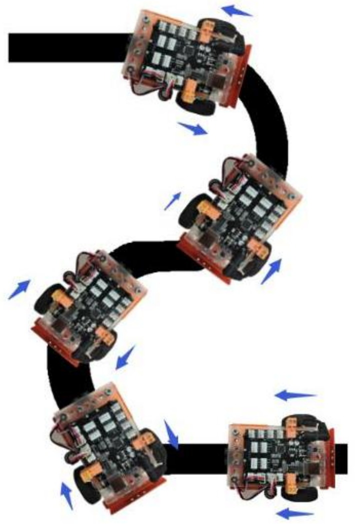




Basic actions in line-following operations.

In line patrol operations, the trolley basically has two actions: going straight and turning, as shown in the figure below.
Go straight: Use line-following sensor A1 and line-following sensor A2 to detect the route when patrolling the line, and to monitor whether the car deviates when patrolling the line, and adjust the action through the differential speed of the left and right wheels.
Turning: Use the line-following sensor A1 and line-following sensor A2 to detect the route when following the line. When the car detects the turning black line, it will make a turning action and change the action through the different actions of the two driving wheels.

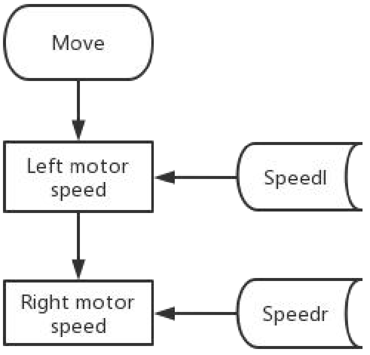
Motion path	Notice
1. Go straight: 	1. The movement path is straight or curved forward. 2. The purpose is to keep the car from deviating from the route during the action (as shown by the arrow on the left). 3. The two sensors are on both sides of the black line and the vehicle is not advancing above the black box. This line tracking method is called "clamping line tracking".

Motion path	Notice
2. Turn 	1. The action path is to change direction. 2. The purpose is to make the car complete the turning action in action (as shown by the arrow on the left). 3. Need to change the action of the two driving wheels. As shown in the picture on the left, the car makes a turning action (right wheel forwards, left wheel backwards), so that it can avoid a large number of deviations from the route when turning.

Practice

Create and use the new function

The new function is a self-built building block defined and constructed by yourself. In order to successfully master the two-sensor line tracking skills, here is a self-made building block that students need to use, as shown in the figure.

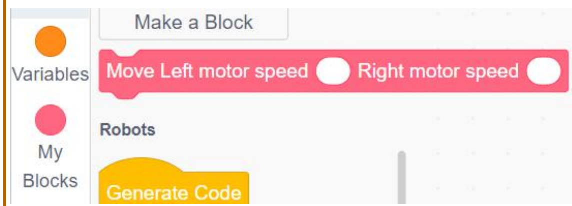
Task flow chart	Instructions for use
Self-made blocks - move: 	The speeds of the two motors M1 and M2 are defined as speedl and speedr. As long as the values of speedl and speedr are changed, the car can move forward and backward and turn left and right.

<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>

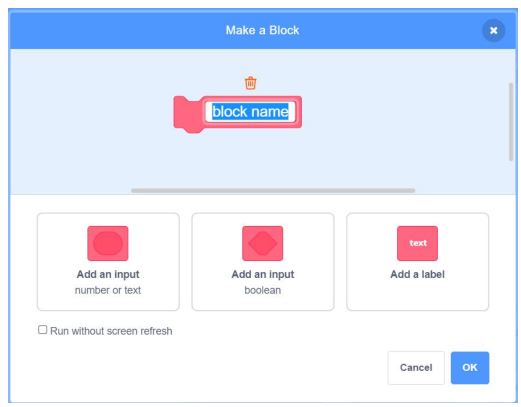
Move:

</td><td>

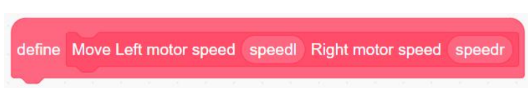
1. Create a new function
① Choose self-made blocks to make new blocks



② After the dialog box for creating a new self-made building block appears, enter the name.



③ After entering the name of the self-made building block, start to set the variable parameters that will be used in the self-made building block, and add a text label to explain it. As shown in the figure below, create a new function called speed control. The two variables speedl and speedr are used in the function.



2. Set the script of self-made building blocks
Let the left wheel travel at speedl and the right wheel at speedr.

</td></tr>
</table>

Program structure	Program description
Data initialization: 	1. Create a new variable, as the gray threshold, name it "gray" and initialize it to 375;
Programming: 	1. The A1 sensor is on a white background (less than the gray threshold) and the A2 sensor is on a white background, the vehicle moves forward; 2. A1 sensor is on the black line and A2 sensor is on a white background, the vehicle turns left; 3. The A1 sensor is on the white background and the A2 sensor is on the black line, the vehicle turns right.

```
#include<Arduino.h>

//              Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr);

double _gray = 0;  //gray;

void setup(){
                pinMode(5, OUTPUT);
                pinMode(6, OUTPUT);
                pinMode(9, OUTPUT);
                pinMode(10, OUTPUT);
                pinMode(A1, INPUT);
                pinMode(A2, INPUT);

                _gray = 375;
}

void loop(){
                if(analogRead(A1) < _gray && analogRead(A2) < _gray){
                                _Move_Left_motor_speed(-150, 150);
                //Move Left motor speed
                }
                if(analogRead(A1) > _gray && analogRead(A2) < _gray){
                                _Move_Left_motor_speed(150, 150);
                //Move Left motor speed
                }
                if(analogRead(A1) < _gray && analogRead(A2) > _gray){
                                _Move_Left_motor_speed(-150, -150);
                //Move Left motor speed
                }
}

//              Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr){
                {
                                int16_t speed = _speedl;
                                analogWrite(5, speed >= 0 ? 0 : -speed);
                                analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                                int16_t speed = _speedr;
                                analogWrite(9, speed >= 0 ? 0 : -speed);
                                analogWrite(10, speed >= 0 ? speed : 0);
                }
}
```

?

When does the rover choose the line-clamping line and when does it choose the line-clamping line?

Let's try it out by designing a map by yourself.

Lesson Twenty one Orbital obstacle avoidance

Students, there are actually many sensors working together during the real rover movement, such as infrared rays, but ultrasonic sensors are rarely used because the air on Mars is thin and ultrasonic waves cannot work properly. But the ultrasonic sensor can be used normally on the earth. In this lesson, we will use ultrasonic waves and line patrol to complete the simulation of Mars rover skills, making the Mars rover more and more powerful!
Ultrasound is a sound wave with a frequency higher than 20000Hz (Hertz). It has good directivity, strong reflection ability, and is easy to obtain concentrated sound energy. It travels farther in the water than in the air. It can be used for distance measurement, speed measurement, cleaning, and welding. , Gravel, sterilization, etc. There are many applications in medicine, military, industry, and agriculture. Ultrasound gets its name because the lower limit of its frequency exceeds the upper limit of human hearing.

Study Task

Task 1: Learn to use "ultrasound" to control the line-following action of the car
Task 2: Write and execute the ultrasonic control line patrol program

Knowledge Point

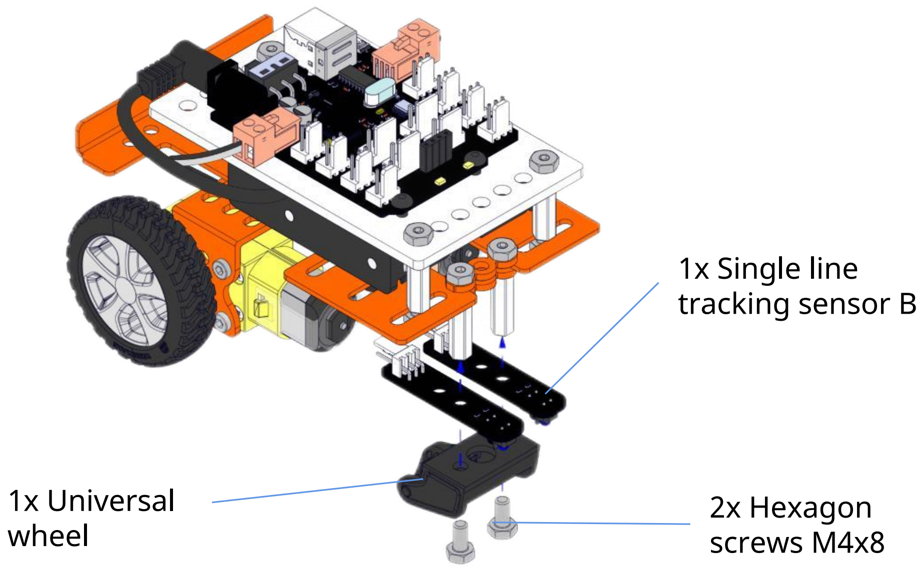
"Ultrasonic" control techniques

1. Structural analysis of the light-following "Mars Rover"

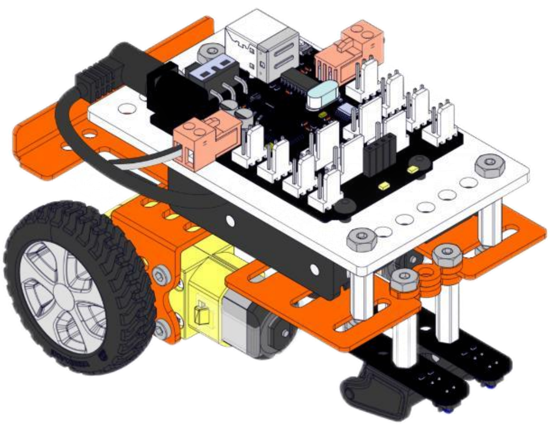
In the previous course, we learned how to use "ultrasound" and the matters needing attention, from which we learned that "ultrasound" can detect the distance of obstacles in front. In this lesson, we will use the value of obstacles to control the car to perform line-following actions.

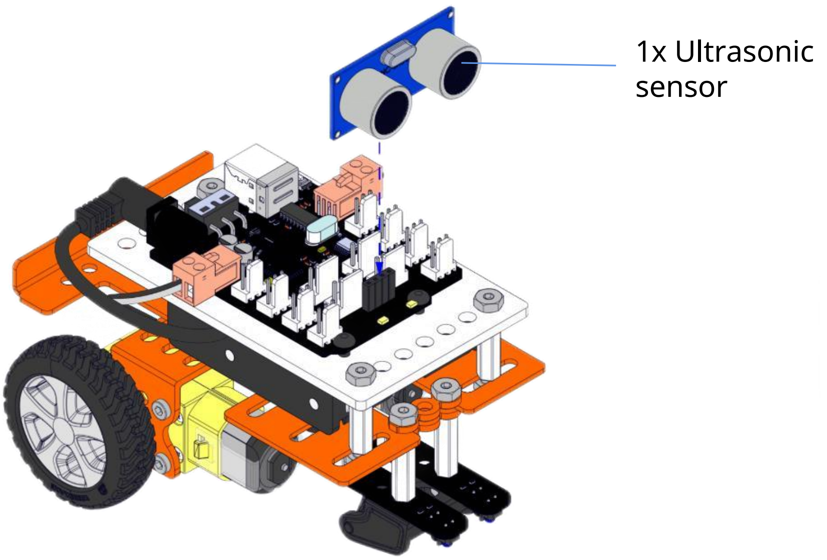
What needs to be done?

1. Measure "distance length"
In order to ensure the accurate travel distance of the "Mars Rover", the ultrasonic wave of the "Mars Rover" must be measured first.
The detection distance between the sensor and the obstacle on the patrol route.
2. Check the "Mars Rover"
Confirm that the motor direction is correct: connect the left motor to M1, connect the right motor to M2, the line-following sensor wiring from left to right are A1 and A2 respectively, and let the rover scan the field to determine the threshold.

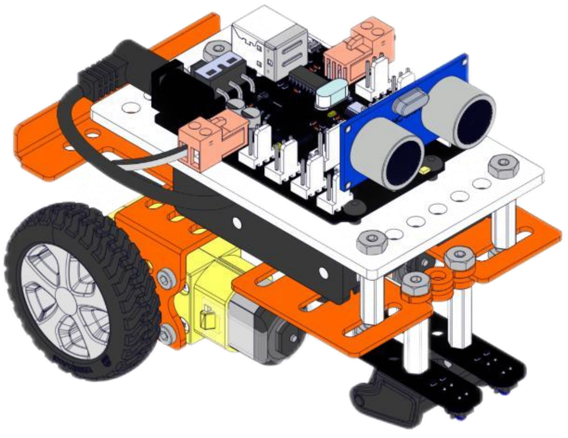


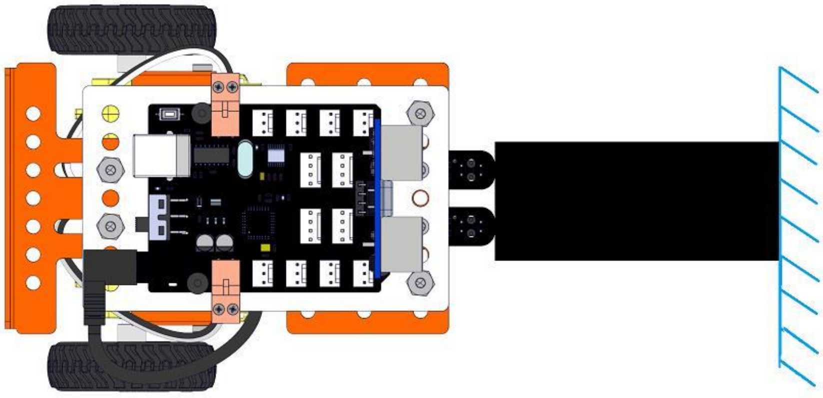
Install single-line tracking sensor B





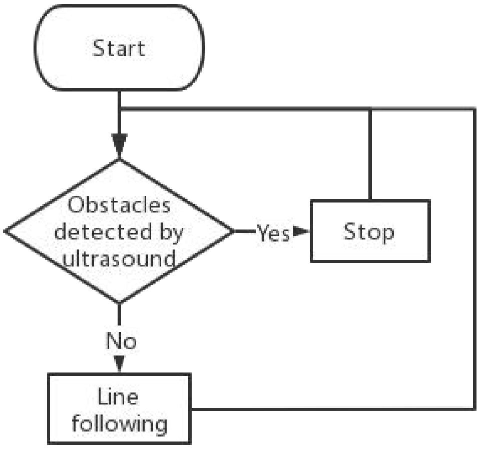
Install Ultrasonic sensor





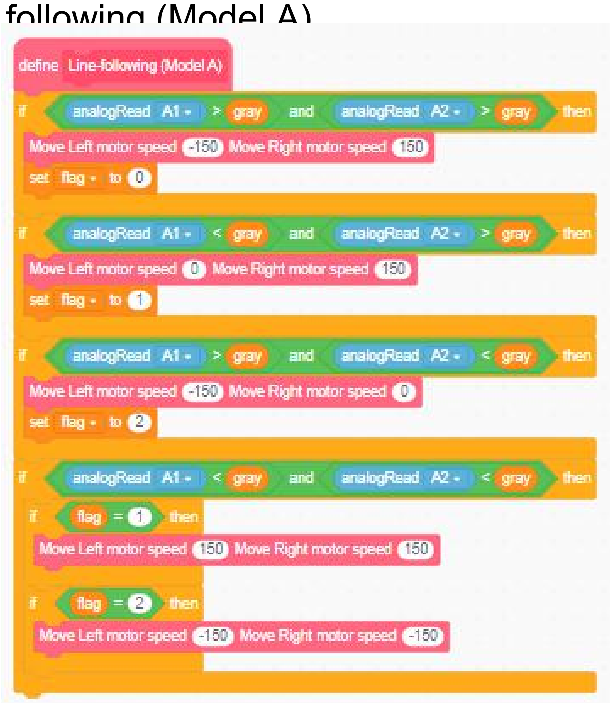
Practice

Programming ideas

Task flow chart	Instructions for use
	1. The rover has always used ultrasonic to detect whether there are obstacles in front of the vehicle; 2. If there is an obstacle, the vehicle will stop moving, if there is no obstacle, the vehicle will continue to follow the line.

Write and execute the program.

Program structure	Program description
New self-made building blocks: Line-following (Model B) 	1. The A1 sensor is on a white background (less than the gray threshold) and the A2 sensor is on a white background, the vehicle moves forward; 2. A1 sensor is on the black line and A2 sensor is on a white background, the vehicle turns left; 3. The A1 sensor is on the white background and the A2 sensor is on the black line, the vehicle turns right.

Program structure	Program description
New self-made building blocks: Line-following (Model A)  	1. The A1 sensor is on the black line (greater than the gray threshold) and the A2 sensor is on the black line, the vehicle is moving forward, and the vehicle state variable is set to 0; 2. The A1 sensor is on the white background and the A2 sensor is on the black line, the vehicle makes a small right turn, and the vehicle state variable is set to 1; 3. The A1 sensor is on the black line and the A2 sensor is on the white background, the vehicle makes a small left turn, and the vehicle state variable is set to 2; 4. The A1 sensor is on the white background and the A2 sensor is on the white background, and then judge the state of the car at the previous moment. If the previous state is a small right turn, and this time it is a large right turn; if the previous state is a small left turn, At this time, it is a big left turn.
Main program:  	1. First initialize the variables; 2. Always check the ultrasonic distance at this time, if the distance is less than 15, let the vehicle stop, if the distance is greater than 15, the normal line inspection; 3. According to the driving trajectory, select the appropriate way of patrolling.

```
#include<Arduino.h>

float ultrasonic_distance(int pinTrig, int pinEcho);
//                  Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr);
//                  Line-following (Model B)()
void _Line19following_14Model_B15();
//                  Line-following (Model A)()
void _Line19following_14Model_A15();

double _gray = 0;  //gray;
double _flag = 0;  //flag;

void setup(){
                    pinMode(2, INPUT);
                    pinMode(4, OUTPUT);
                    pinMode(5, OUTPUT);
                    pinMode(6, OUTPUT);
                    pinMode(9, OUTPUT);
                    pinMode(10, OUTPUT);
                    pinMode(A1, INPUT);
                    pinMode(A2, INPUT);

                    _gray = 375;
                    _flag = 0;
}

void loop(){
                    if(ultrasonic_distance(4, 2) < 15){
                                        _Move_Left_motor_speed(0, 0);      //Move Left motor
speed
                    }else{
                                        _Line19following_14Model_A15();   //Line-following
(Model A)
                    }
}

float ultrasonic_distance(int pinTrig, int pinEcho){
                    digitalWrite(pinTrig, LOW);
                    delayMicroseconds(2);
                    digitalWrite(pinTrig, HIGH);
                    delayMicroseconds(10);
                    digitalWrite(pinTrig, LOW);
                    float distance = pulseIn(pinEcho, HIGH) / 58.0;
                    delay(10);
                    return distance;
}

//                  Move Left motor speed("speedl", "speedr")
void _Move_Left_motor_speed(double _speedl, double _speedr){
                    {
                                        int16_t speed = _speedl;
                                        analogWrite(5, speed >= 0 ? 0 : -speed);
                                        analogWrite(6, speed >= 0 ? speed : 0);
                    }
                    {
                                        int16_t speed = _speedr;
                                        analogWrite(9, speed >= 0 ? 0 : -speed);
                                        analogWrite(10, speed >= 0 ? speed : 0);
                    }
}
```

```
//                      Line-following (Model B)()
void _Line19following_14Model_B15(){
                        if(analogRead(A1) < _gray && analogRead(A2) < _gray){
                                                _Move_Left_motor_speed(-150, 150);
                        //Move Left motor speed
                        }
                        if(analogRead(A1) < _gray && analogRead(A2) > _gray){
                                                _Move_Left_motor_speed(-150, -150);
                        //Move Left motor speed
                        }
                        if(analogRead(A1) > _gray && analogRead(A2) < _gray){
                                                _Move_Left_motor_speed(150, 150);
                        //Move Left motor speed
                        }
}

//                      Line-following (Model A)()
void _Line19following_14Model_A15(){
                        if(analogRead(A1) > _gray && analogRead(A2) > _gray){
                                                _Move_Left_motor_speed(-150, 150);
                        //Move Left motor speed
                                                _flag = 0;
                        }
                        if(analogRead(A1) < _gray && analogRead(A2) > _gray){
                                                _Move_Left_motor_speed(0, 150);    //Move Left motor
speed
                                                _flag = 1;
                        }
                        if(analogRead(A1) > _gray && analogRead(A2) < _gray){
                                                _Move_Left_motor_speed(-150, 0);   //Move Left motor
speed
                                                _flag = 2;
                        }
                        if(analogRead(A1) < _gray && analogRead(A2) < _gray){
                                                if(_flag == 1){
                                                                        _Move_Left_motor_speed(150, 150);
                        //Move Left motor speed
                                                }
                                                if(_flag == 2){
                                                                        _Move_Left_motor_speed(-150, -
150);                   //Move Left motor speed
                                                }
                        }
}
```

?

After studying in this lesson, I also want to know about the line patrol car:

We have learned the ultrasonic detection distance patrol, what scene is it suitable for?

Lesson Twenty two Mars Rover transport work

The “Zhurong” rover has a height of 1.85 meters and a weight of about 240 kg. It has a pair of cute “big eyes” and a square device at the top. It is the rover’s panoramic camera, which can see the topography and landforms in front of it and can help Mars. When the car avoids obstacles to carry forward and carry out detection, its multi-spectrum has several spectral bands to identify the composition of the minerals in front.

There are 4 big wings next to the belly of the rover, which are 4 solar panels. Like the Yutu lunar rover, the rover also relies on solar energy to obtain energy. However, because Mars is farther from the sun and the atmosphere on the surface also reduces sunlight, the rover has more wings than Yutu.

The rear part of the rover is used to get in touch with the earth. Not only can it obtain action instructions, but it can also use it to return precious data found on Mars to home on the earth. The rover carries six types of payloads, including terrain cameras, multispectral cameras, subsurface detection radars, surface composition detectors, surface magnetic field detectors, and meteorological measuring instruments, which provide a guarantee for the completion of the scientific mission of patrolling and detecting the surface of Mars.

Among them, the subsurface detection radar can detect the composition of the soil 10 meters underground on Mars, and the detection depth of the ice layer can reach 100 meters. In order to resist the low temperature of minus 85 degrees Celsius at night, and to be stronger,

The rover uses a variety of high-tech materials. The speed of the rover on Mars is 200 meters per hour. The design life of the rover is three Martian months, which is equivalent to approximately 92 Earth days.

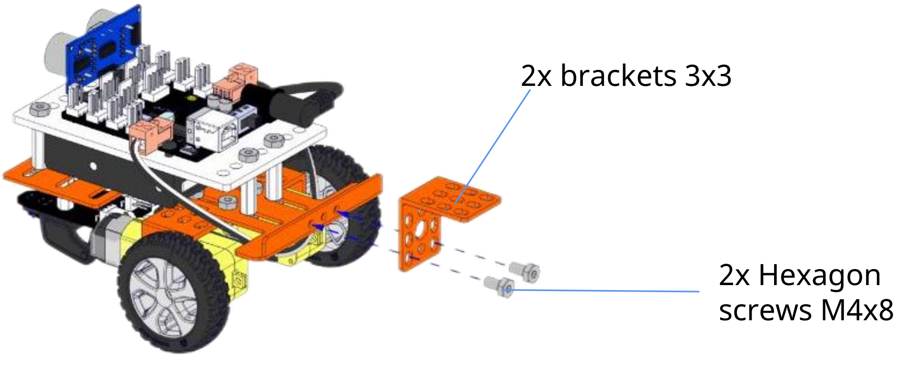
Study Task

Task 1: Assemble a "Mars Rover" that can transport materials
Task 2: Use ultrasound to control the stop position of the "Mars Rover"
Task 3: Write the program and complete the control of the rover to transport materials

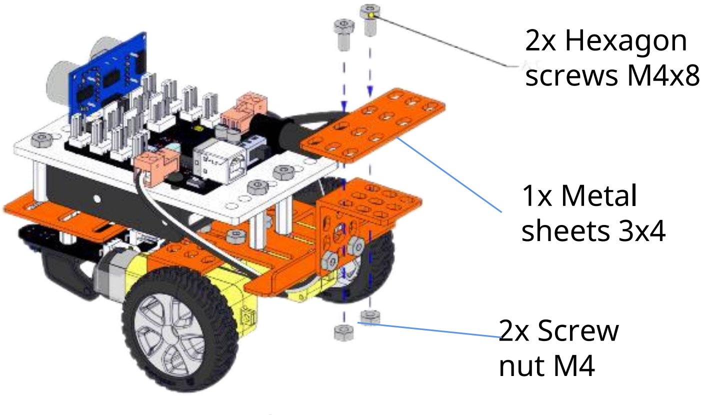
Knowledge Point

Rover transportation installation steps

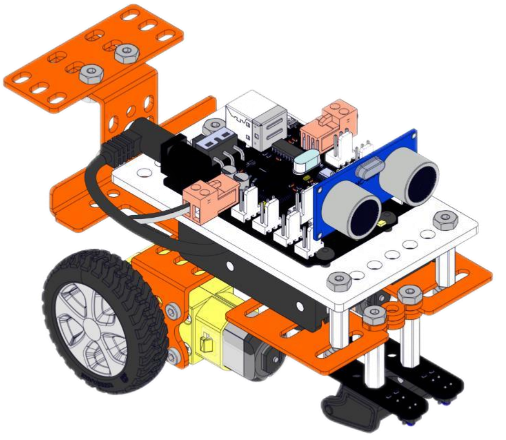
Please refer to Lesson 9 for the main body structure. The assembly method of the rover transportation is divided into 3 steps. Students can refer to the figure below to complete the assembly. It takes about 5 minutes. Next, we will start to install it.



1. Mounting brackets



2. Install metal sheet 3x4



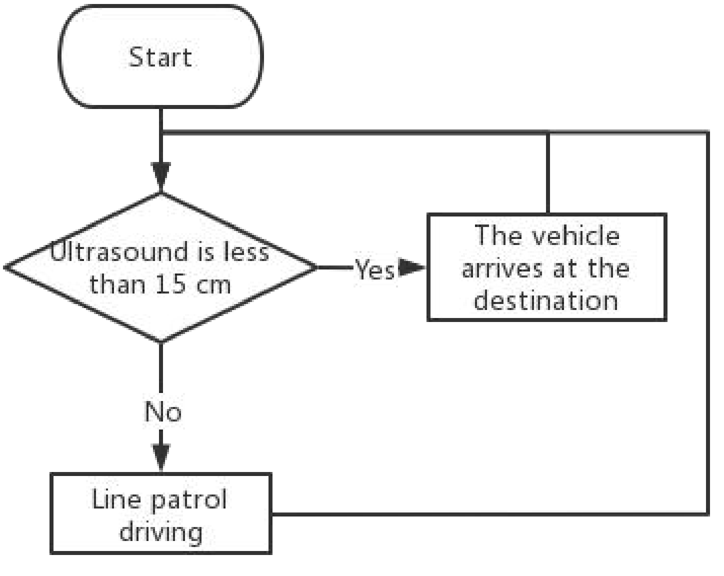
3. Place the goods on the bracket made at the back to complete the goods handling.

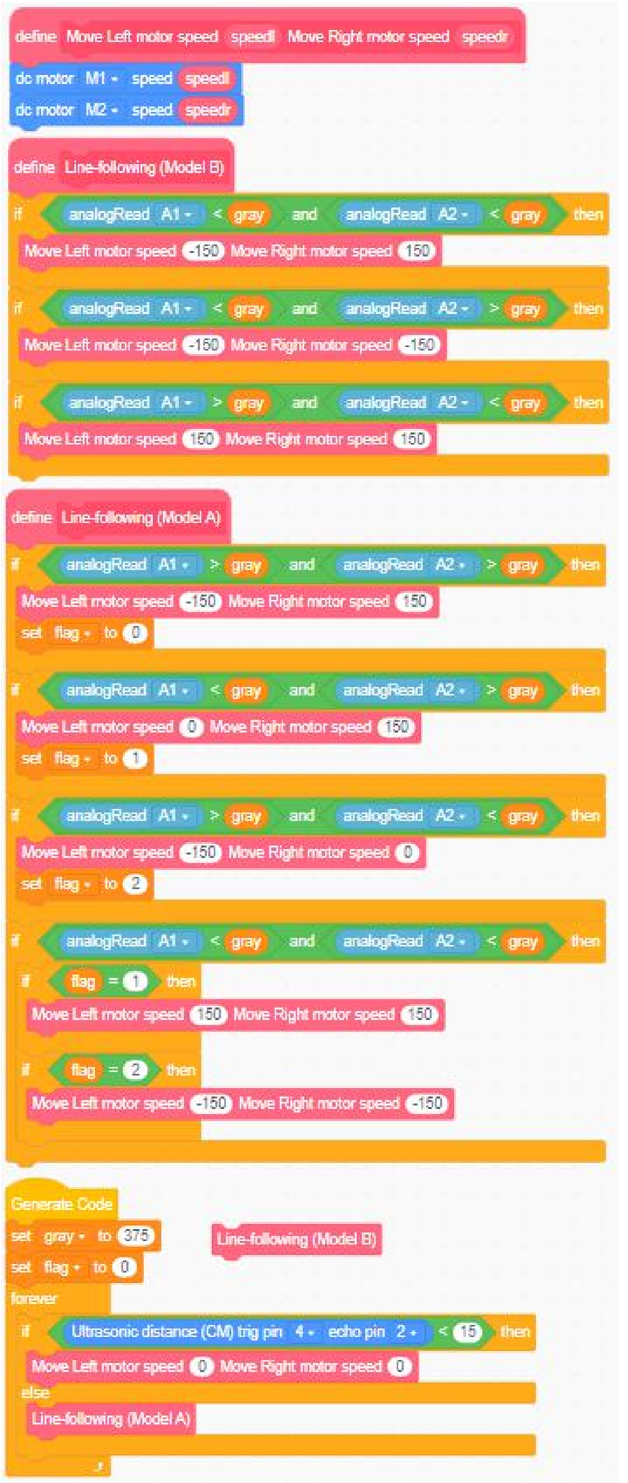
Check the connection status of the rover

1. Check the wiring of the motor and sensor, check whether the connecting wire is firmly inserted;
2. Data cable connection test, check whether the data cable is plugged in firmly.

Write the program for the rover transportation

The programming logic of "Mars Rover Programming" is the same as the on-board LED lights we wrote. As long as we know the movement state of the vehicle after hitting the wall, we can complete the programming.



<table>
<tr><th>Program structure</th><th>Program description</th></tr>
<tr><td>


</td><td>It is the same as the previous course, compile pressing line patrol and clamping line patrol, choose different patrol methods according to different routes.</td></tr>
</table>

Classmates, let's practice it quickly and see if the rover transport can deliver specific goods to the designated location? If not, how should the vehicle be adjusted?

?

If there is no robotic arm on Mars, how can we install and unload cargo?

Lesson Twenty Three
Multifuntional Mars Rover Car

Classmates have learned a lot about Mars exploration car. In fact, there are many sensors to work together in real world, which is a huge system, Now let’s make such a Mars rover car!

As the fifth Mars rover that the United States sent to mars, "Perseverance" is equipped with a series of new technology to identifying ancient Martian environments capable of supporting life, seeking out evidence of former microbial life existing in those environments.

With a set of cameras, X-rays, laser imaging, and new landing equipment, Mars car will complete the task of exploring Jejeo's mount, which is a lake site 35 billion years to find biological characteristics.

The perseverance login will rely on AI and computer vision to achieve "the most accurate landing to date".

There are two instruments at the end of the Mars car arm: One is the X-ray rocky chemical planetary instrument, using the X-ray bundle to determine the chemistry signs of the past. The second is Raman luminescent organic and chemicals that scan the living environment, then emit lasers to the target to assess the concentration of organic minerals and molecules and may exist in previous water environments.

The Mars rover also has an interested technology called Mars Oxygen In-Situ Resource Utilization Experiment(MOXIE), which is an instrument that converts carbon dioxide in the Mars into oxygen. At the same time, the roaming car is also equipped with a prototype, a laser microscopic, an ultraviolet spectrometer, and a multi-camera.

According to NASA, engineers will understand Mars atmosphere by measuring the heat and pressure experienced by Mars. Specifically, Mars ambient dynamic analyzer can accomplish this task by measuring wind speed and wind direction, atmospheric pressure, temperature, dust size and humidity sensors. The Mars will also include a "advanced sensor kit" including a temperature sensor, a pressure sensor, and a weather station.

Finally, the detective radar is a radar penetrating ground, and a detailed centimeter image can be created for the geological structure of the meter surface.

Study Task

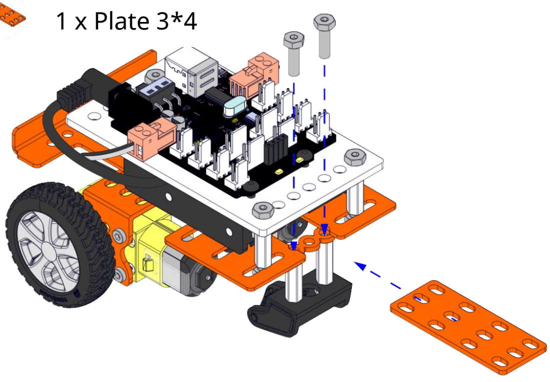
Task 1: Assemble sensors and mechanical parts.
Task 2: Write the program of by making blocks of “obstacle avoidance”, “chasing light”, “sound detection”, “buzzer”.
Task 3: Use “Button” to switch program, write the main program.

Practice

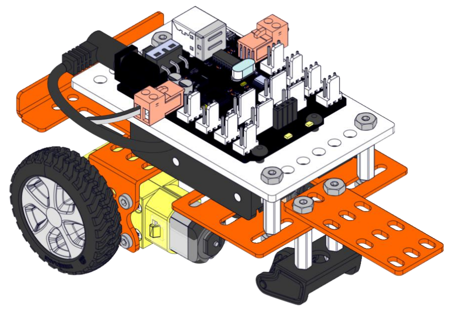
Assemble sensors and mechanical parts.

1. Install Plate 3*4

1 x Plate 3*4



2. Install Plate 3*4

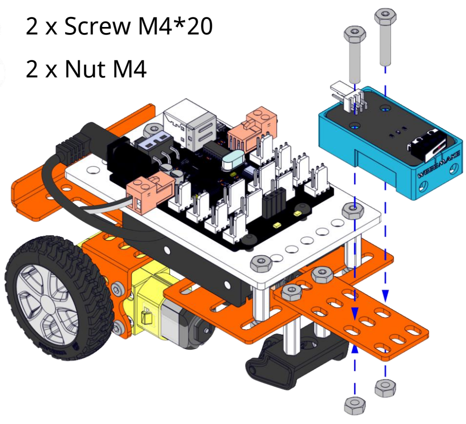


3. Install the limit switch module.

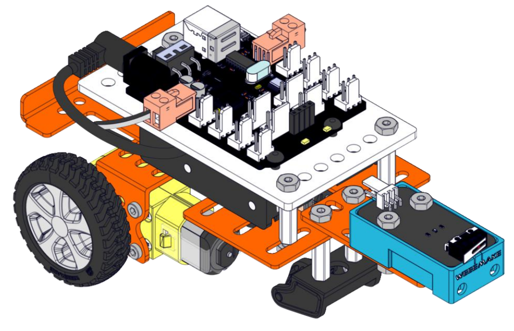
1 x Limit Switch

2 x Screw M4*20

2 x Nut M4



4. Install the limit switch module.

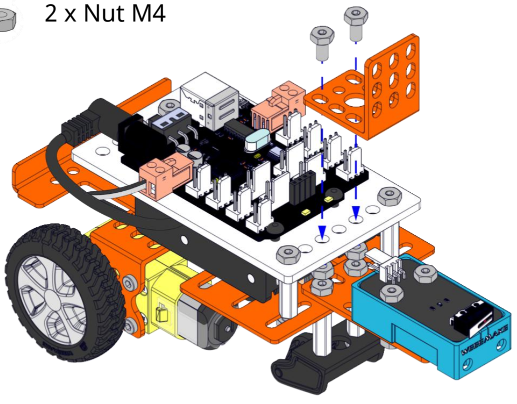


5. Install the Bracket 3*3

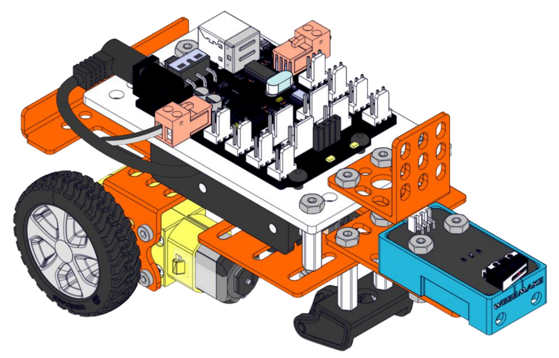
2 x Screw M4*8

2 x Bracket 3*3

2 x Nut M4

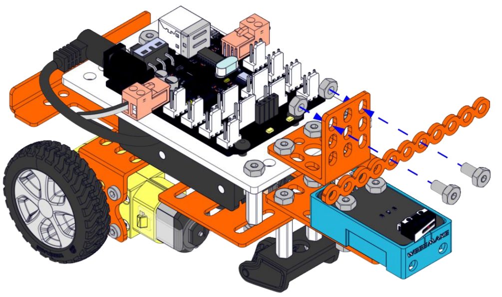


6. Install the Bracket 3*3

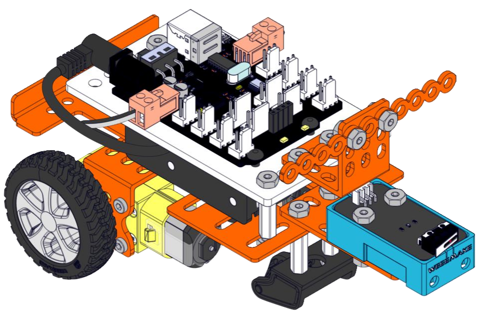


7. Install Linkage 0208-88

1 x Linkage 0208-88
2 x Screw M4*8
2 x Nut M4

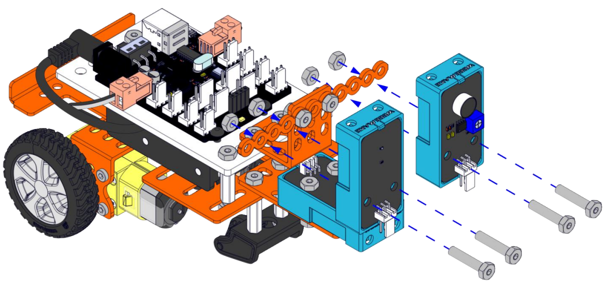


8. Install Linkage 0208-88

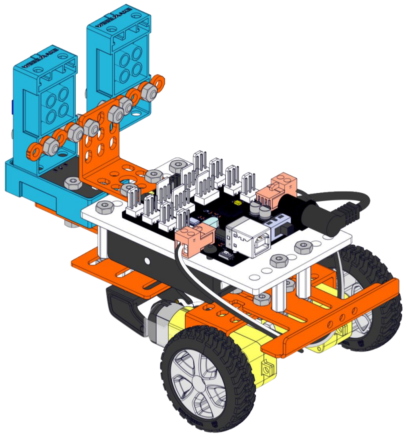


9. Install the light and sound sensor.

1 x Sound Sensor
1 x Light Sensor
4 x Screw M4*20
4 x Nut M4

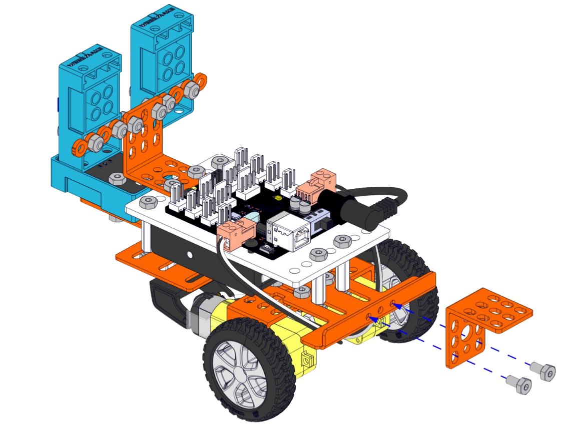


10. Install the light and sound sensor.

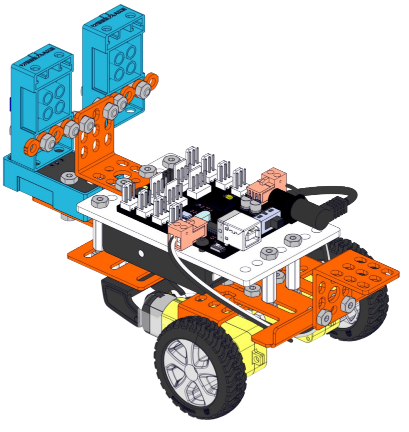


11. Install the Bracket 3*3

2 x Screw M4*8
2 x Bracket 3*3

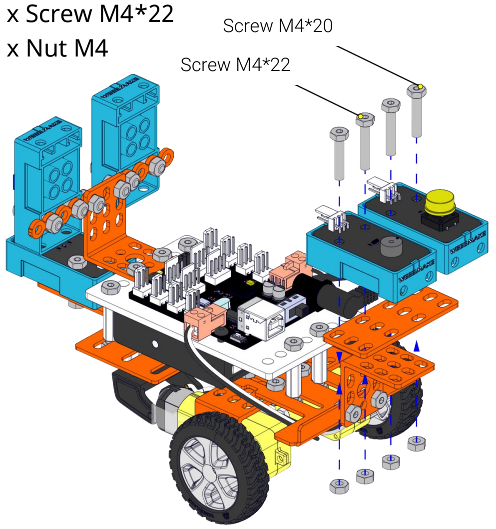


12. Install the Bracket 3*3

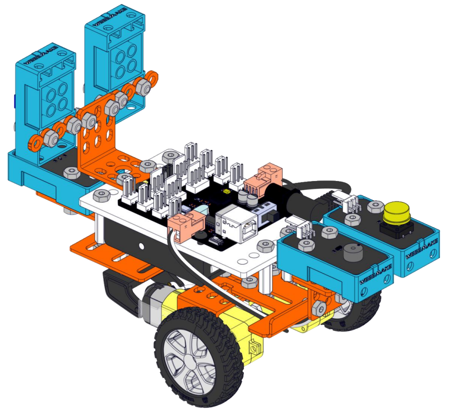


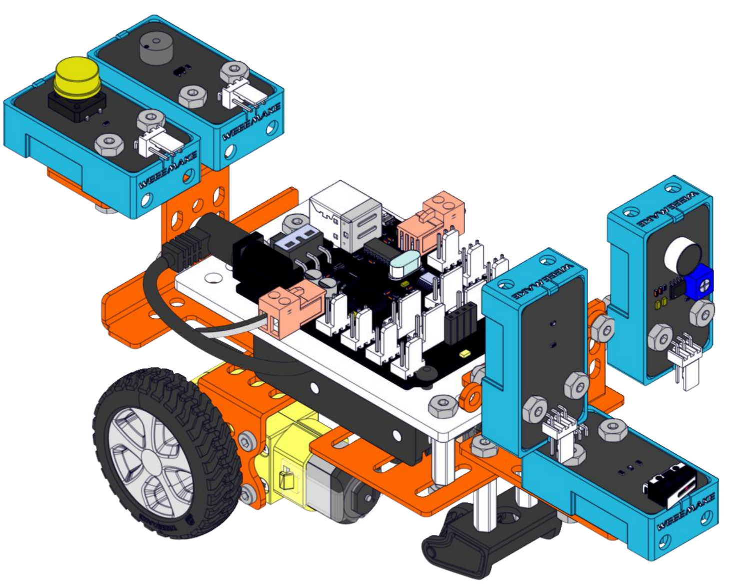
13. Install button and buzzer.

1 x Button
1 x Buzzer
1 x Plate 3*4
2 x Screw M4*20
2 x Screw M4*22
4 x Nut M4



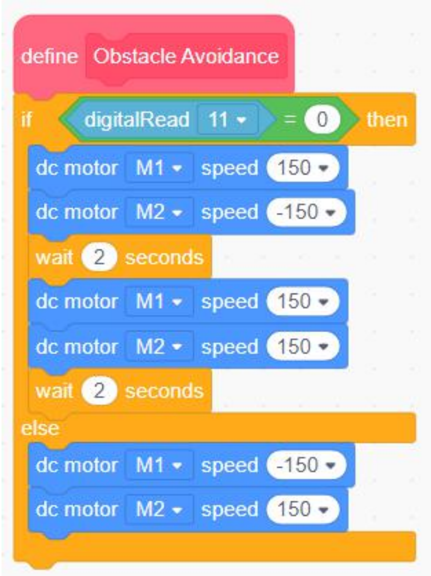
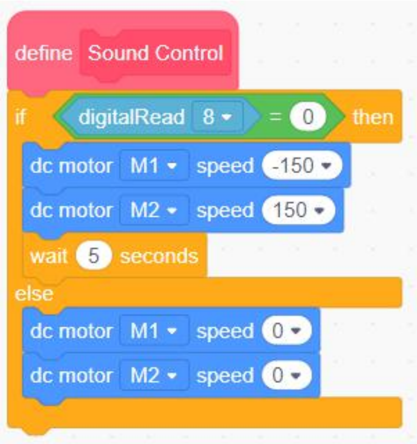
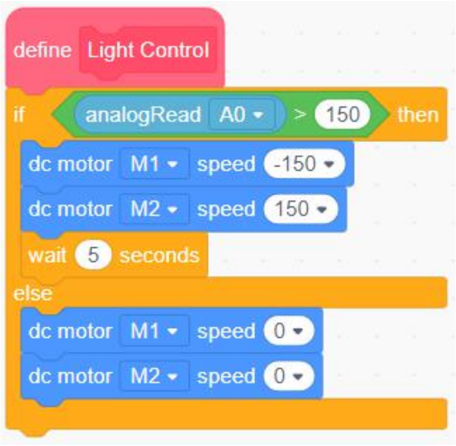
14. Install button and buzzer.

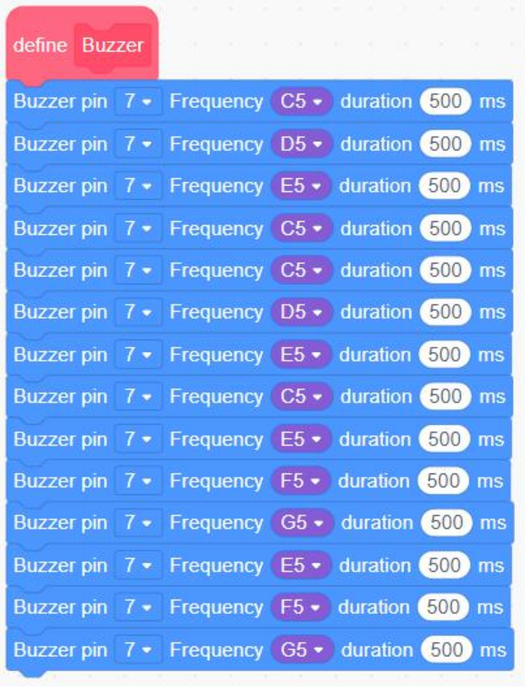




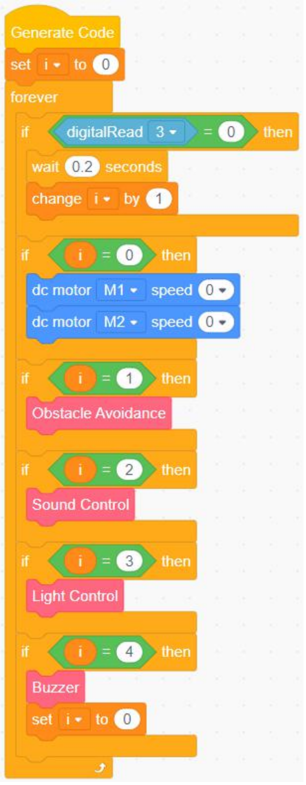
Write program for the multifunctional Mars Rover robot.

Use “make a block” to create program for each function.

Program structure	Program description
1. Obstacle avoidance program.  	Connect limit switch module to digital port 11; If the mars rover hit any obstacle (disital port 11 read value is 0), the robot car should move backward for 2 seconds, and then turn left (or right) for 2 seconds. If the mars rover hits nothing, it should drive forward.
2. Sound control program.  	Connect sound sensor to digital port 8; If the mars rover didn’t detect any sound (disital port 8 read value is 0), the robot car should move forward. Otherwise, the robot car should stop.
3. Light control program. (Chasing light)  	Connect light sensor to analog port A0; If the mars rover detects any light (analog port A0 read value is larger than 150), the robot car should move forward. Otherwise, the robot car should stop.

Program structure	Program description
4. Buzzer program. 	Connect buzzer to port 7; Search a sheet music on the internet, and set to the buzzer.

Write the main program.

Program structure	Program description
	Make a variable I, set I to 0; In repeated execution, detect if the button is pressed (connect the button to digital port 3, if pressed, port 3 read value 0) Once button is pressed, increase the variable I by 1, wait 0.2 seconds; If the variable is 0, the mars rover stops; If the variable is 1, run program “obstavle avoidance”. If the variable is 2, run program “sound control”. If the variable is 3, run program “light control”. If the variable is 4, run the buzzer program, and reset the variable I to 0.

```
#include<Arduino.h>

void buzzer(int pin, long frequency, long duration);
//                    Obstacle Avoidance()
void _Obstacle_Avoidance();
//                    Sound Control()
void _Sound_Control();
//                    Light Control()
void _Light_Control();
//                    Buzzer()
void _Buzzer();

double _i = 0;        //i;

void setup(){
                    pinMode(3, INPUT);
                    pinMode(5, OUTPUT);
                    pinMode(6, OUTPUT);
                    pinMode(7, OUTPUT);
                    pinMode(8, INPUT);
                    pinMode(9, OUTPUT);
                    pinMode(10, OUTPUT);
                    pinMode(11, INPUT);
                    pinMode(A0, INPUT);

                    _i = 0;
}

void loop(){
                    if(digitalRead(3) == 0){
                                        delay(200);
                                        _i += 1;
                    }
                    if(_i == 0){
                                        {
                                                            int16_t speed = 0;
                                                            analogWrite(5, speed >= 0 ? 0 : -speed);
                                                            analogWrite(6, speed >= 0 ? speed : 0);
                                        }
                                        {
                                                            int16_t speed = 0;
                                                            analogWrite(9, speed >= 0 ? 0 : -speed);
                                                            analogWrite(10, speed >= 0 ? speed : 0);
                                        }
                    }
                    if(_i == 1){
                                        _Obstacle_Avoidance();                    //Obstacle Avoidance
                    }
                    if(_i == 2){
                                        _Sound_Control();//Sound Control
                    }
                    if(_i == 3){
                                        _Light_Control();   //Light Control
                    }
                    if(_i == 4){
                                        _Buzzer();
                                        _i = 0;
                    }
}
```

```
void buzzer(int pin, long frequency, long duration){
        long pulse = 500000 / frequency;
        for(long i=frequency*duration/1000;i>0;--i){
                digitalWrite(pin, HIGH);
                delayMicroseconds(pulse);
                digitalWrite(pin, LOW);
                delayMicroseconds(pulse);
        }
}

//      Obstacle Avoidance()
void _Obstacle_Avoidance(){
        if(digitalRead(11) == 0){
                {
                        int16_t speed = 150;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = -150;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
                delay(2000);
                {
                        int16_t speed = 150;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = 150;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
                delay(2000);
        }else{
                {
                        int16_t speed = -150;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = 150;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
        }
}

//      Sound Control()
void _Sound_Control(){
        if(digitalRead(8) == 0){
                {
                        int16_t speed = -150;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
```

```
                {
                        int16_t speed = 150;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
                delay(5000);
        }else{
                {
                        int16_t speed = 0;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = 0;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
        }
}

//      Light Control()
void _Light_Control(){
        if(analogRead(A0) > 150){
                {
                        int16_t speed = -150;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = 150;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
                delay(5000);
        }else{
                {
                        int16_t speed = 0;
                        analogWrite(5, speed >= 0 ? 0 : -speed);
                        analogWrite(6, speed >= 0 ? speed : 0);
                }
                {
                        int16_t speed = 0;
                        analogWrite(9, speed >= 0 ? 0 : -speed);
                        analogWrite(10, speed >= 0 ? speed : 0);
                }
        }
}

//      Buzzer()
void _Buzzer(){
        buzzer(7, 523, 500);
        buzzer(7, 587, 500);
        buzzer(7, 659, 500);
        buzzer(7, 523, 500);
        buzzer(7, 523, 500);
        buzzer(7, 587, 500);
        buzzer(7, 659, 500);
        buzzer(7, 523, 500);
        buzzer(7, 659, 500);
        buzzer(7, 698, 500);
        buzzer(7, 784, 500);
        buzzer(7, 659, 500);
        buzzer(7, 698, 500);
        buzzer(7, 784, 500);
}
```

?

Can you personalize your mars rover robot by adding a sheel for it?

Lesson Twenty Four
Deveopment Challenge - Mars Rover

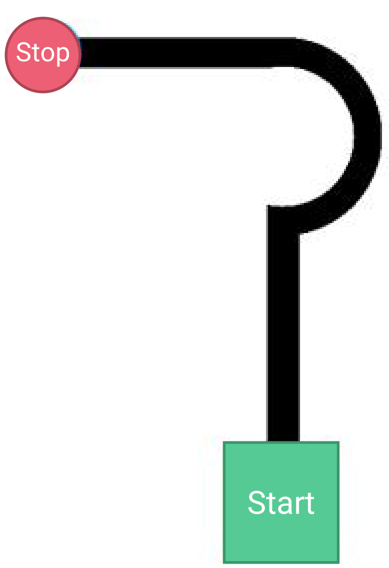
Through the study of all courses, students have been able to control the Mars rover to complete a variety of action and tasks, and now take a palm to the achievements! Congratulations on the last lesson of this book. We know that the route of the Mars rover will be changed according to the instructions from the earth, and there will be a new challenge. Being a small maker who can make a mars rover robot, we are willing to continue to accept challenges, then what challenges now?

Study Task

Task 1: Complete the assigned task.
Task 2: Speed challenge, be quicker.
Task 3: Challenge the super goal.

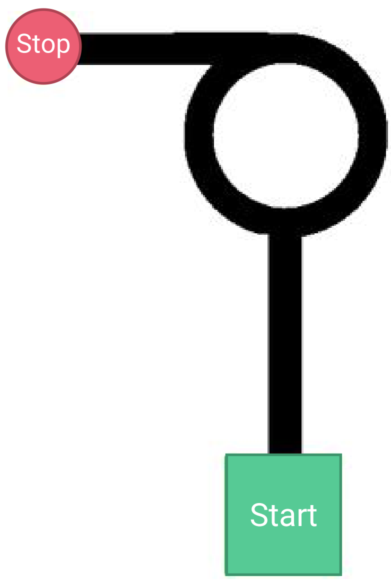
Knowledge Point

Complete the assigned task.

Task	Instruction
Map: 	1. The robot starts from the starting point, and stops at the end point. The end point is a cylindrical obstacle. 2. The robot should follow the black line to the end. 3. The line-following mode can be optional.

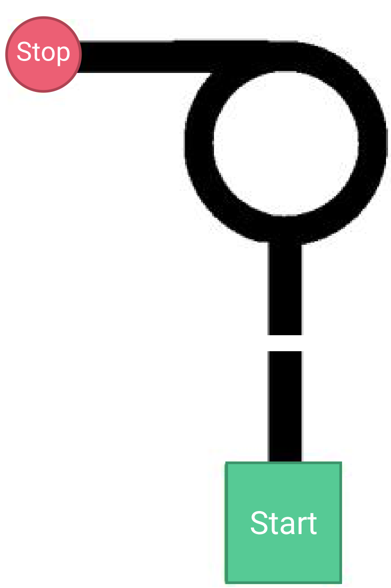
Speed challenge, be quicker.

Through the line-following mission, see who's mars rover is the fastest.

Task	Instruction
Map: 	1. The robot starts from the starting point, and stops at the end point. The end point is a cylindrical obstacle. 2. The robot should follow the black line to the end. The line-following mode can be optional. 3. The fastest robot has the highest score.

Challenge the super goal.

Through the line-following mission, see who's mars rover is the fastest.

Task	Instruction
Map: 	1. The robot starts from the starting point, and stops at the end point. The end point is a cylindrical obstacle. 2. The robot should follow the black line to the end. The line-following mode can be optional. 3. There is a gap in the black line, fine a way to pass it. 4. The fastest robot has the highest score.

?

In completing a difficult challenge task, you need to pay attention to the structure of the mars rover, and pay attention to the size. This is not only tests the programming capabilities of the students, but also reflects the mechanical structure and engineering understanding. So, in mechanical disciplines, what mechanical structures do you know?