Tongs robot and Debugging

Debugging the tongs robot

Fix the program errors in the space to the right.

Pay attention to the angles of the servomotor. Find the problems in the script below!

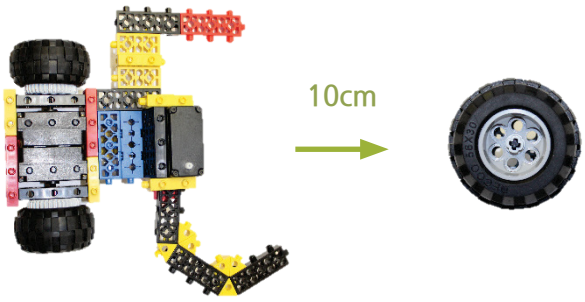
Program debugging journal

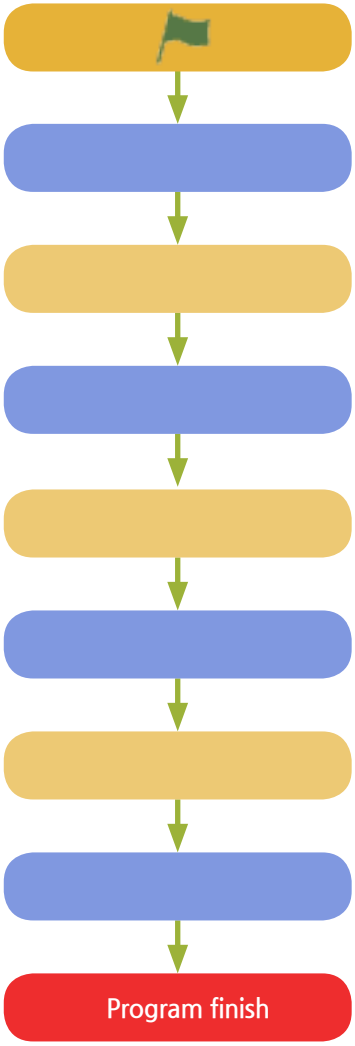
CLASS 2

Tongs robot Applications

Applying the uses of the tongs robot

Make a program that moves the tongs robot towards the tire and moves it back with the tire. Put the tire 10 cm away from the robot. Make the program pay attention to the distance between the robot and tire.
Using the flowchart, write down which movements it should have and code them in Scratch.





SCRATCH CODING KIT

Logic boost

CDS SENSOR

LESSON 9

CLASS 2

Introducing the sensor

What is the brightness sensor? (CDS sensor)

Now let's find out the brightness sensor which is one of the robotory input sensors.
The brightness sensor measures the amount of light. It's also known as a light sensor.

The value changes depending on the amount of light that goes into the sensor. We can produce different movements using the changing values on the light sensor.

Side

front

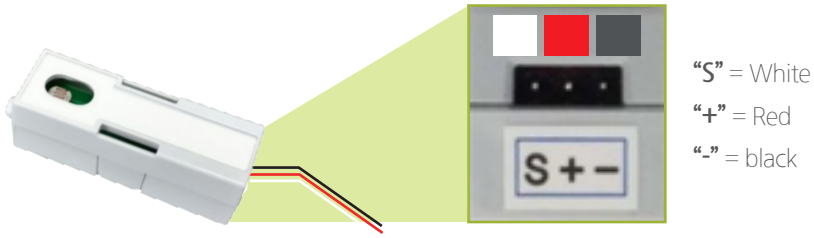
Above is the robotory brightness sensor. Look for one in the robotory materials.

Now, let's check for the values that the brightness sensor recognizes.

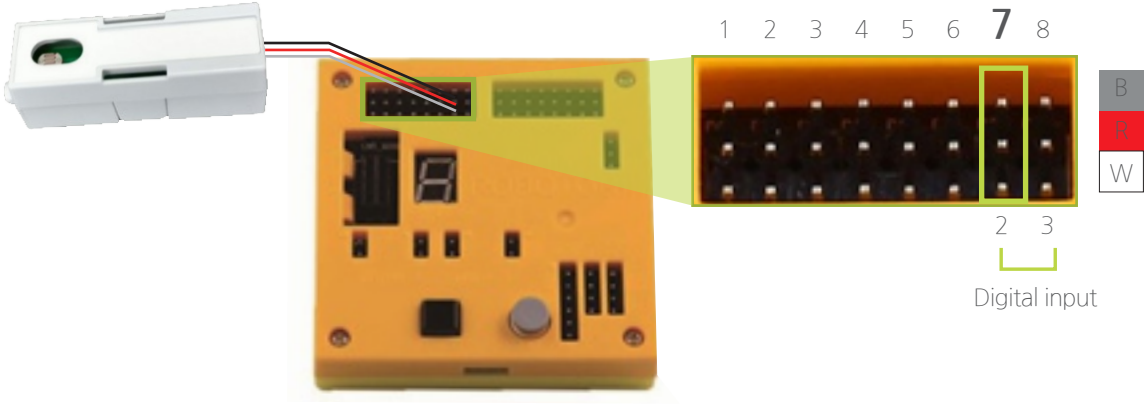
Connecting to the main cell

Connecting the brightness sensor (CDS sensor)

Now, we have learned about the brightness sensor. Let's learn how to connect it to the main cell. Use the 3-pin connector like you see below.



※ If you don't connect with the right cable colors, the sensor won't work, so be careful!
Connect the sensor to the main cell's slot 7.

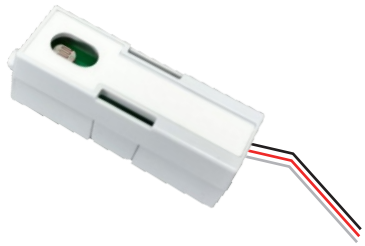


CLASS 2

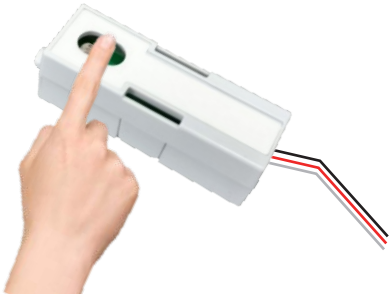
Experiencing the Brightness sensor coding

Using the brightness sensor

Now, let's see how Scratch shows the information that the sensor gives.
Cover and uncover the connected sensor.



If the brightness sensor is still, 'Digital2' will show 'true'



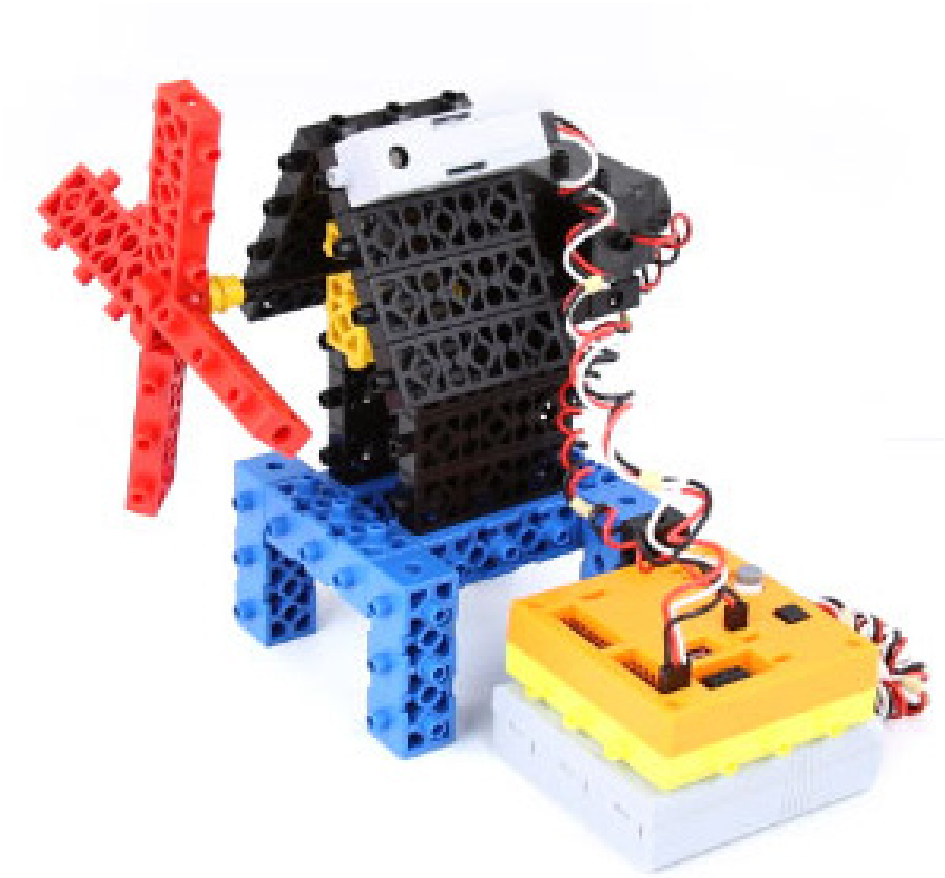
If the sensor was covered, 'digital 2' will show 'false'.

If digital 2 is true, it means the light is shining through. If it is false, it means it's dark with no light shining through.

How to Make a Robot

Using the brightness sensor

Now, we are going to find out how to apply the brightness sensor to the program.
First, let's make our old friend, the windmill. You must be really quick in making the windmill now.



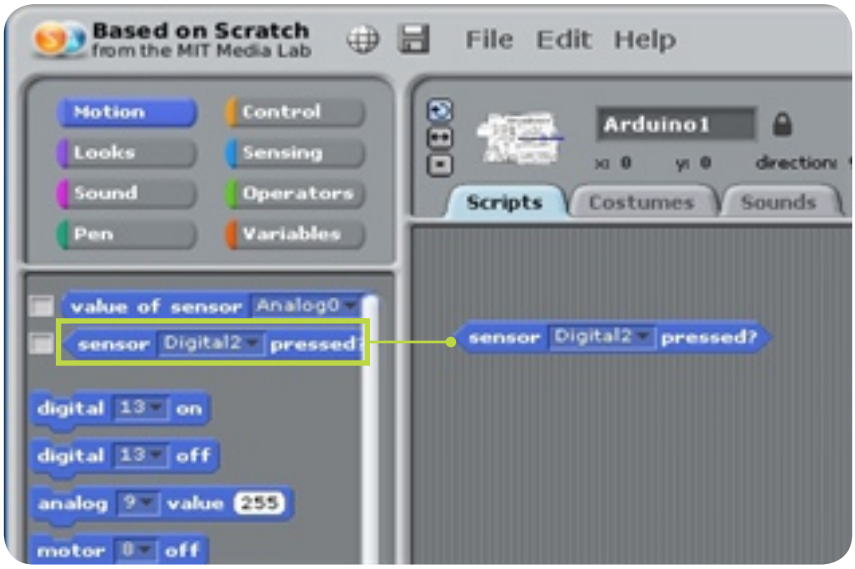
CLASS 2

Experiencing the Brightness sensor coding

Usinig the brightness sensor with the windmill

We are going to use the brightness sensor to operate the windmill.

First, we need a way to bring 'true' and 'false' of the brightness sensor to Scratch. To do this, go to the 'movement' tab and drag in 'is the sensor digital 2 pressed?' to the script.

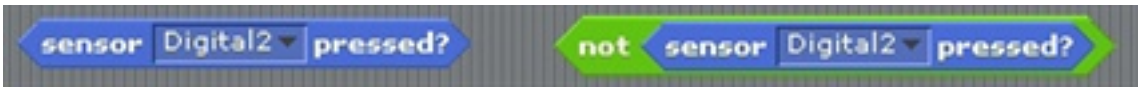


block shows if the value of the sensor is 'true' or 'false'

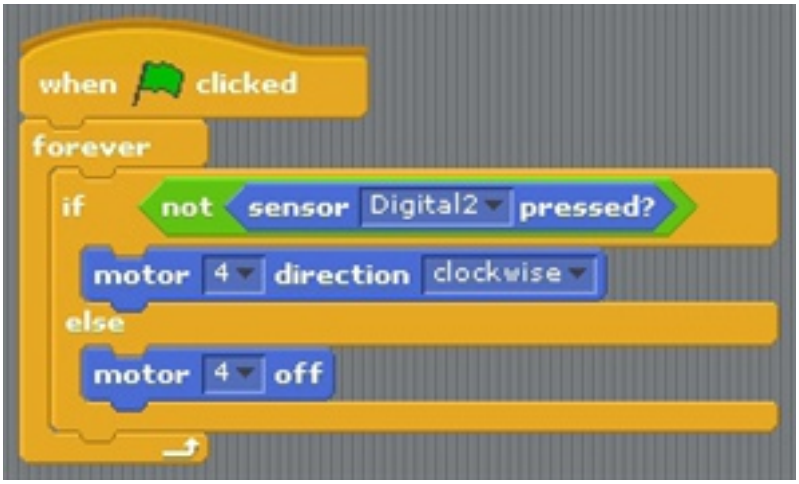
Experiencing the Brightness sensor coding

Coding the windmill with the brightness sensor

The button sensor 'is the sensor digital 2 pressed?' was used earlier as the condition to operate the windmill. For the button sensor, the first value was 'false' and it turned 'true' when the button was pressed.
But, the brightness sensor shows 'true' from the beginning. Therefore, we need an opposite condition to correctly operate the windmill.



Bring the 'not' block from the calculation tab and combine it with the condition block. This denies the condition. Basically, it makes the condition opposite. Now, the condition that operates the motor is 'when the brightness sensor's digital value is false'!

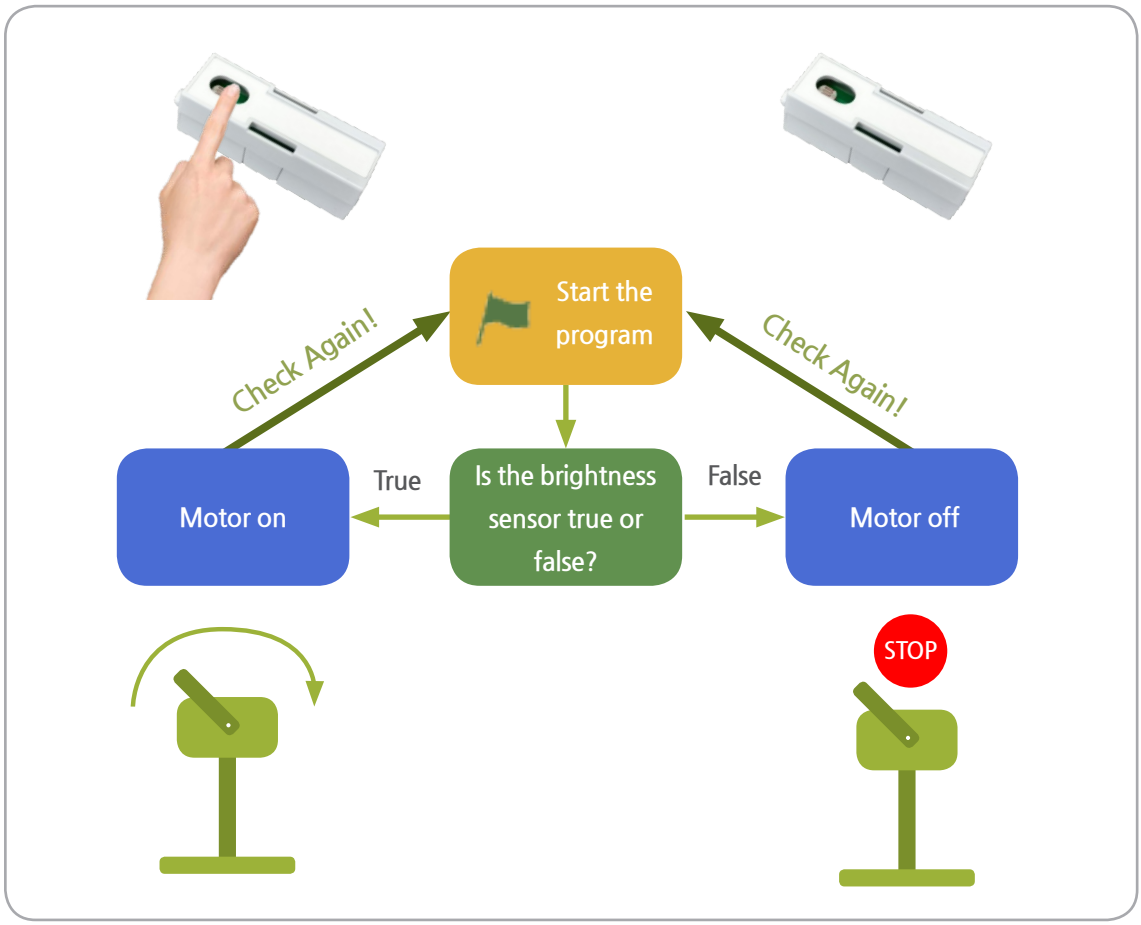


Make the script above. If the script above is activated, the DC motor will operate when the brightness sensor is covered.

CLASS 2

Experiencing the Brightness sensor coding

The flowchart looks the same except the condition

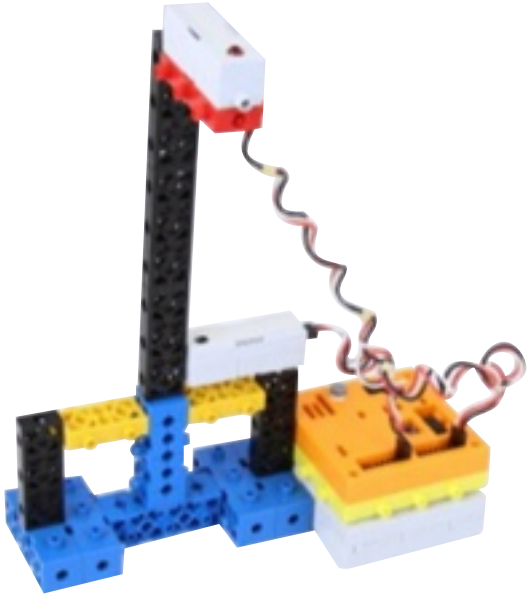


If you press the brightness sensor, the motor will turn on.
If you don't, the motor will turn off!

How to Make a Robot

Streetlight

Make a streetlight model using the brightness sensor.



Materials needed for the streetlight model

					
Diamond H8 x 1	Diamond H6 x 1	Diamond V6 x 4	Rubi 7 x 1	Rubi 2 x 1	Rubi 0 x 4
					
Mini 2 x 2	Mainboard 128 x 1	Battery case x 1	LED SENSOR x 1	CDS SENSOR x 1	connector x 3

CLASS 2

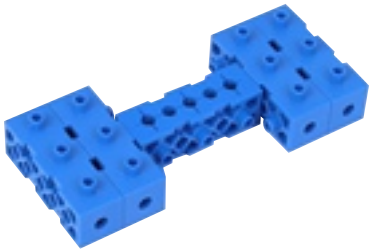
How to Make a Robot

Streetlight

1

x 4

x 1

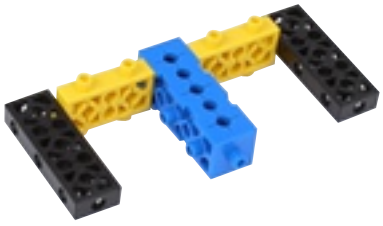


2

x 1

x 2

x 2



3

Combine figure 1 and figure 2

4

x 1

x 2

x 1

5

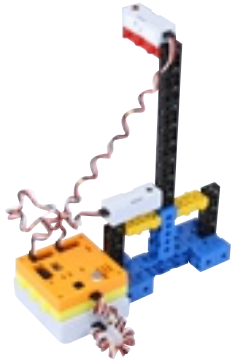
x 1

x 3

x 1

x 1

x 1



Experiencing the Brightness sensor coding

Turning on the streetlight using the brightness sensor

Make a script so that the streetlight turns on when the brightness sensor is covered.
Draw an outline below.

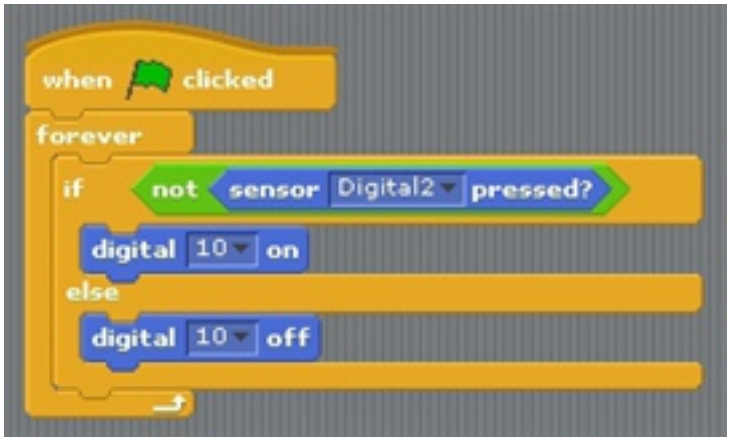
CLASS 2

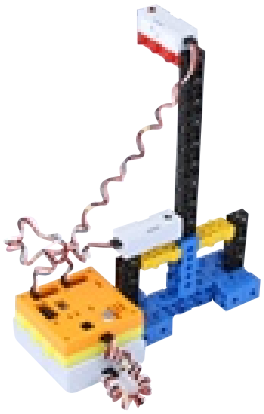
Experiencing the Brightness sensor coding

Turning on the streetlight using the brightness sensor

Do you remember how we activated LED sensor using the button sensor?
It's easy if you think about it as if we were using the brightness sensor instead of the button sensor.

Also, keep in mind that the digital values of the button sensor and brightness sensor are opposite. If you are trying to turn on the LED using the brightness sensor, you should make a script that looks like the one below.



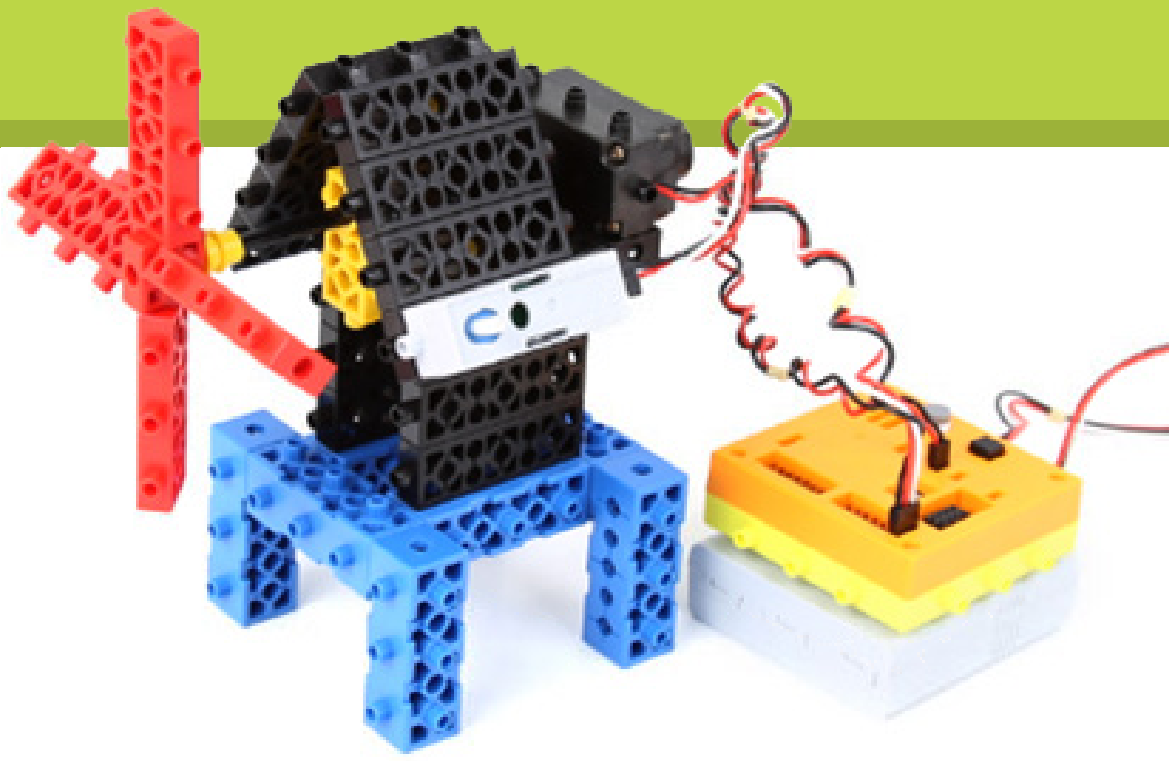


SCRATCH CODING KIT

Logic boost

Analog

LESSON 10

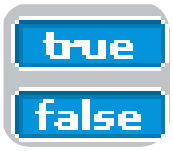


CLASS 2

Intro to Analog

What is an analog?

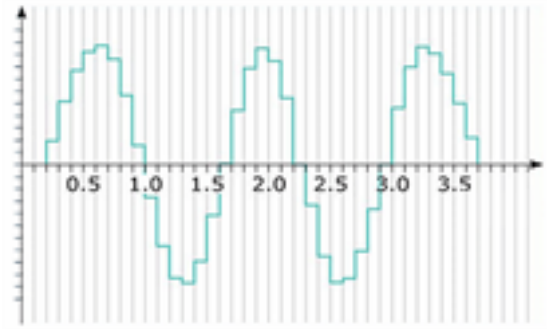
We know that 'true' and 'false' show whether the button/brightness sensor is on/dark or off/bright. This tells us only if it is on or off, so we called it digital output.



But in real world, the object is not just 'bright' or 'dark'. There are different amounts of brightness. Now, the brightness sensor can show the 'analog output' value. This shows a number to indicate how bright it is instead of just saying 'true' or 'false'.

Analog vs digital

Analogs were used mostly when computers were more expensive and not as popular. Now, for example, the music we listen to now with smartphones are available due to digital technology. Many '0's and '1's show that it's digital.

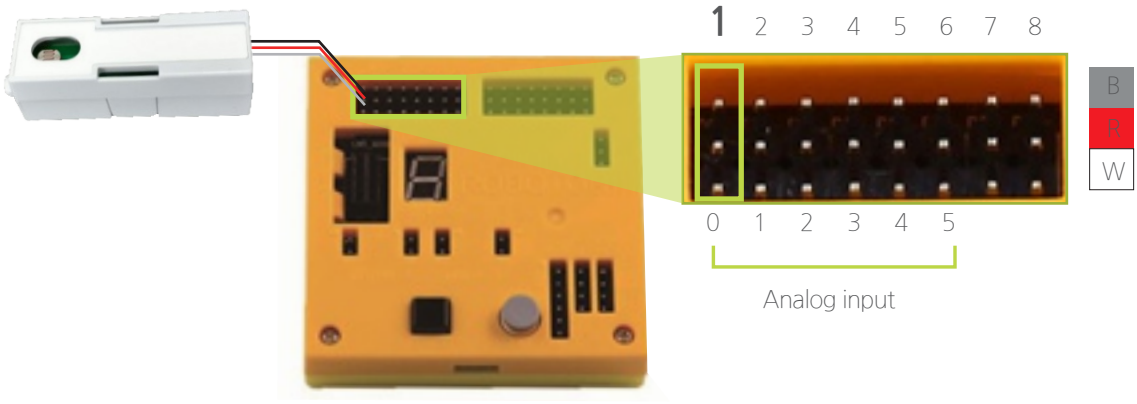


In this lesson, we will learn about analog values using the brightness sensor and the volume sensor.

Connecting to the main cell

Connecting the brightness sensor to the analog port

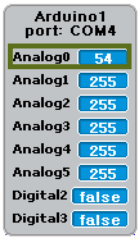
Connect the brightness sensor to the analog port of the main cell.



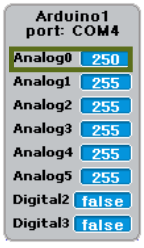
Check out the value after connecting the brightness sensor!

You can see that the value of Analog0 port changes. This is the analog value of the brightness sensor. The number will increase if you cover the brightness sensor, and the number will decrease if you lift up your finger(uncover the sensor).

You can see a small value at 'analog0' when the sensor is not covered



When the sensor is covered, you can see 'analog0' value going up



The analog value of the brightness sensor will be using the value of the amount of light received.

CLASS 2

Intro to the sensor

What is the volume sensor?

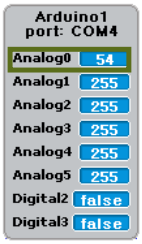
Let's learn about analog values more!
Not only the brightness sensor shows the analog value, the other sensors also show the analog values.
We are going to find out about one of the robotory sensors, the volume sensor.
The volume sensor literally controls the analog value like a remote controls the volume of music.

A volume sensor can be used with the servo cell. Put the servo cell on the volume sensor. The analog value will decrease when the servo cell is rotated clockwise and vice versa. It can change analog values in more detail than other input sensors.

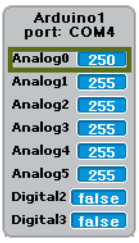
Side

front

When the volume sensor is rotated clockwise, the 'analog0' will decrease.



When the volume sensor is rotated counter clockwise, the 'analog0' wil increase.



Connecting to the main cell

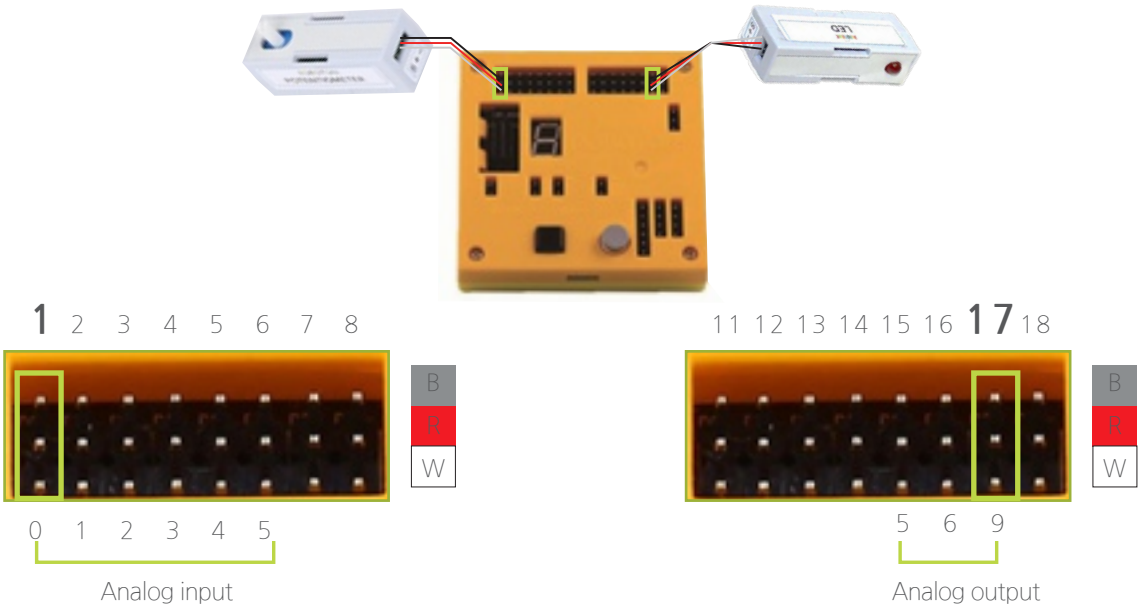
Coding the streetlight using the volume sensor

Do you remember how you coded the streetlight using the brightness sensor?
Now, we are going to use the volume sensor to code the streetlight. Since we learned how to turn on and off the streetlight, now we are going to make a streetlight that changes its brightness depending on the analog value.

We used the 'digital output/off' block to turn the LED sensor on and off. For the analog value ouput, let's see which block we should use. Drag in the block below to the script.

Did you drag in the block from the movement tab? Ompared to the 'digital output/off' block this block shows the output value in numbers. Basically, it can control the brightness by typing in the output value.

Let's use the streetlight here. We should connect the volume sensor instead of the brightness sensor. Connect the brightness sensor to analog input port 0 (port 1) and the LED sensor to analog output port 9 (port 17).



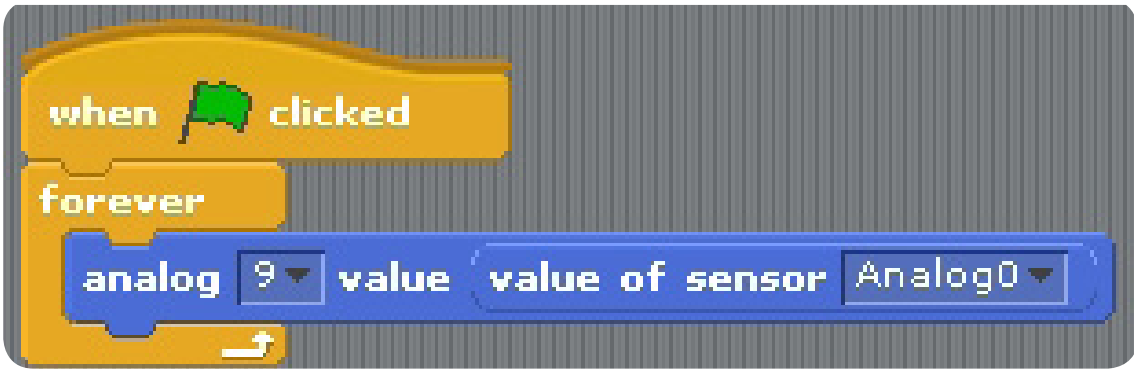
CLASS 2

Experiencing the volume sensor coding

Turning on the streetlight using the volume sensor

Let's start coding the streetlight using what we learned before (first, make the model).
The simple part is to make the volume sensor's analog output value the same as the volume sensor's analog input value. We just need to combine two coding blocks like below!

Do you understand this? Now, let's code the streetlight using the start block and adding the 'infinite repetition' block. Just make the analog output value equal to the volume sensor's analog input value. Then, just combine the two coding blocks. Check out the picture below.



Now, control the brightness of the streetlight using the volume sensor! The streetlight should be brighter as it gets darker outside.

Operator

What is an operator?

An operator is used to compare sizes.
Go to the calculation tab and find these blocks.

Is smaller than

same

Is bigger than

These blocks show '__is smaller than ___', 'equal to', and '___is bigger than___'.
The computer will decide if the operater is 'true' or 'false' looking at the values you typed into blocks. Look at the examples below.

TRUE	80 < 180	100 = 100	213 > 50
FALSE	120 < 80	30 = 40	42 > 60

For true examples, 80 is smaller than 180, so the computer says it's true. But, 120 is not smaller than 80, so the computer says it's false. Do the exercises below.

true

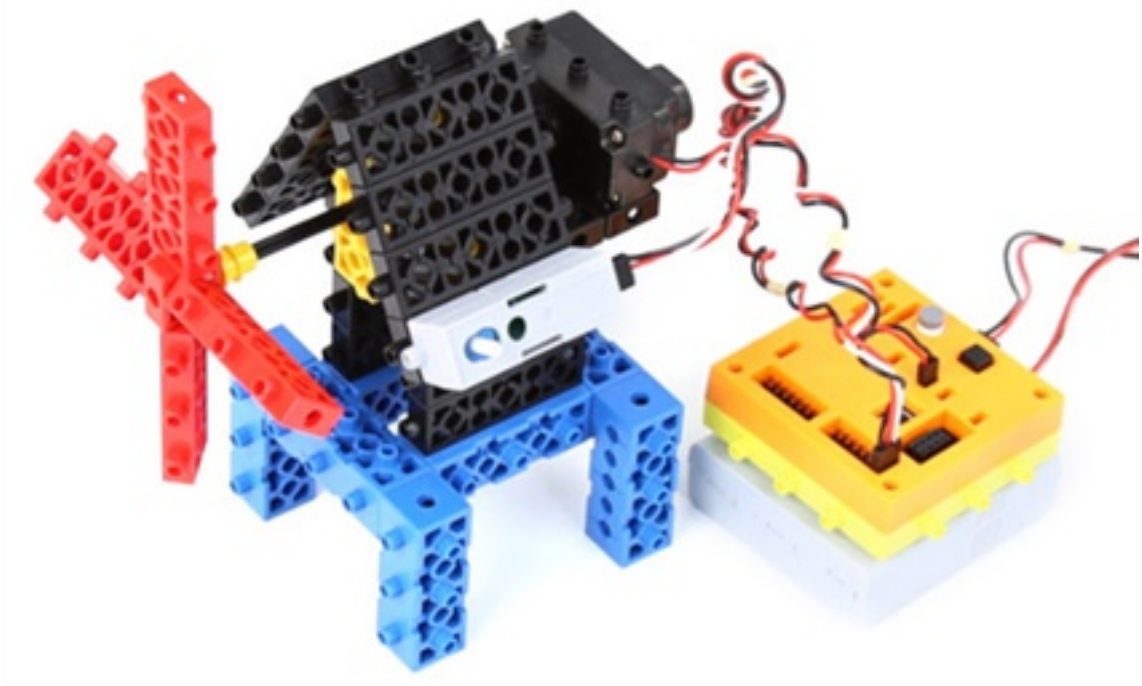
CLASS 2

Operator

Using the operator

Using the operator is like the start of artificial intelligence (A.I.) that uses sensors.
We are going to make the computer judge analog sensors using the operator.

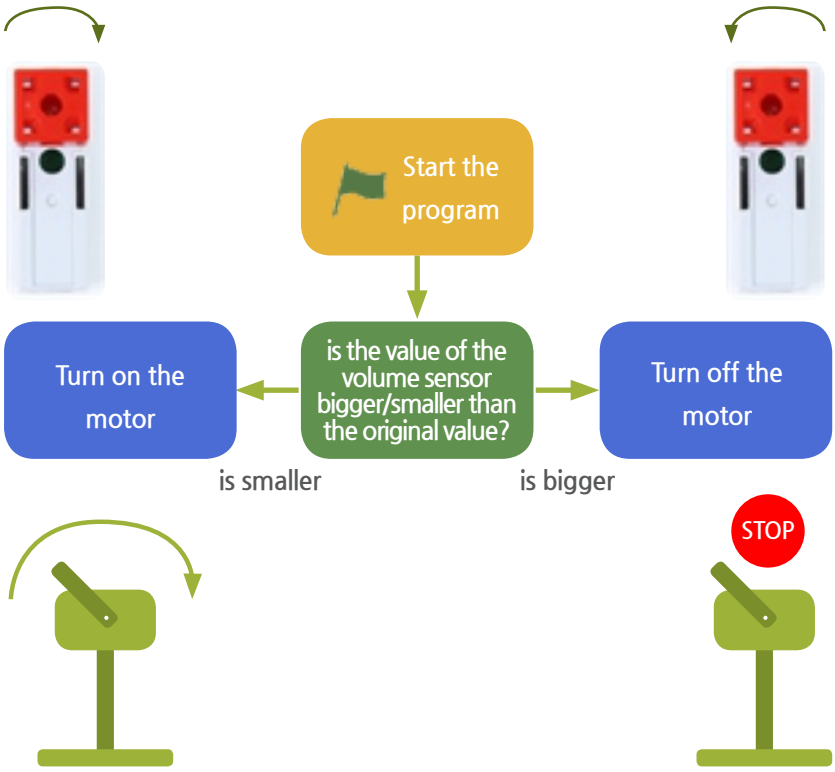
To understand this easily, we are going to make a windmill that is connected to the volume sensor (Refer to the manual before or the homepage-moretips online).



Operator

Using the operator

Now, we are going to use the volume sensor with the operator and analog values to activate the motor. Look at the flowchart below and say the program is going to keep repeating.



The program will keep checking if the value of the volume sensor is bigger/smaller than the original value. Remember that the value of the analog sensor goes up when the sensor is covered and it goes down when the sensor is uncovered.

You have to set a value that will stop the motor when you rotate the servocell on the windmill's volume sensor. This is called 'threshold'.

CLASS 2

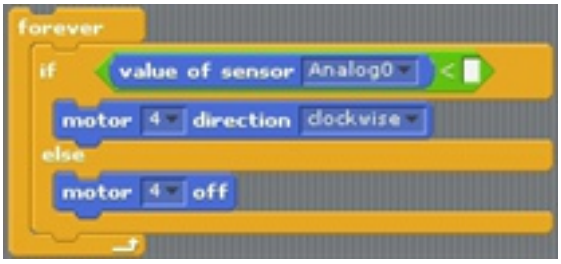
Operator

Using the operator

Out code will look like the example below. If the volume sensor value goes below a certain number, the motor will turn on. If it goes over, the motor will turn off. Ths is the same as the volume sensor's analog value going up when we rotate the sensor counter clockwise.

Add the calculation block to the code. Add  this block and combine it with the 'value of analog0 sensor' block.

To make this true, the operator is requiring the value of the volume sensor to be smaller than a certain number. The rest of the code will look like the one below.



Next, we have to choose our threshold value.

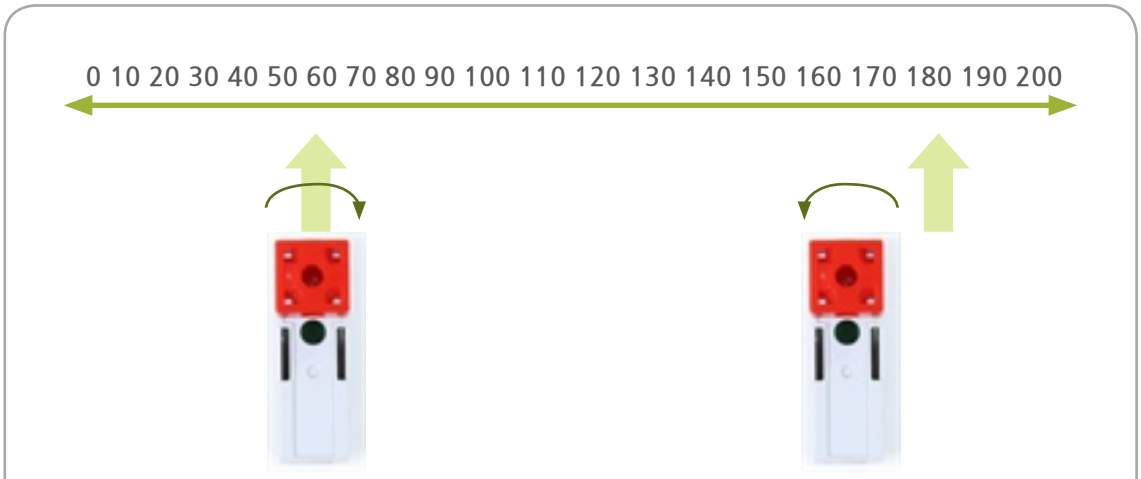
Threshold value

Threshold

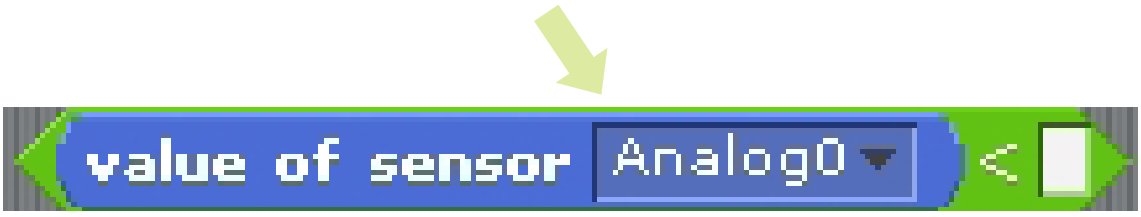
Let's see what threshold is. Say we are listing every possible number of the volume sensor's analog value.

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200

The numbers that we see, whether we rotate it clockwise or counter clockwise, will be between two different numbers.



The threshold value would be the value that goes into the operator block to start a program.

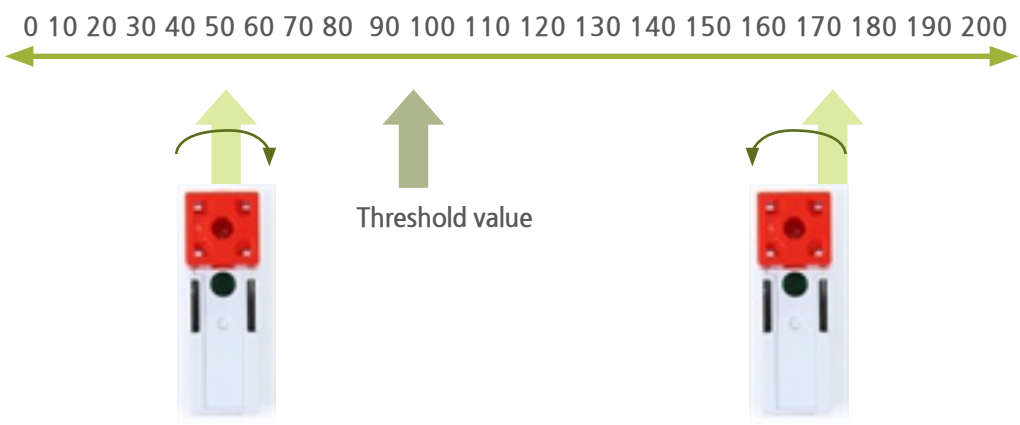


CLASS 2

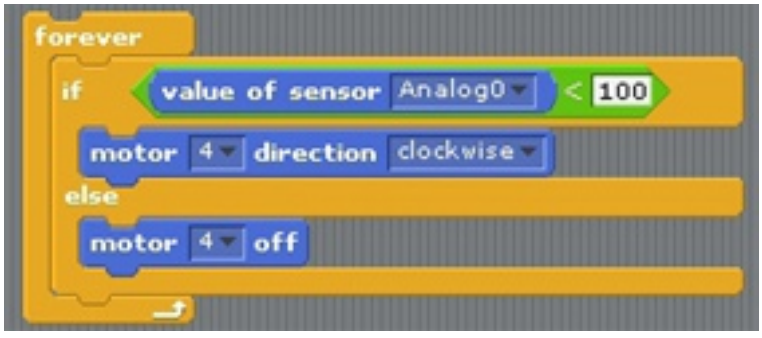
Threshold value

Threshold

To start a program, the value has to be at least near 50. To make it safe, we can set the value somewhere near 100.



Now your code will look like this

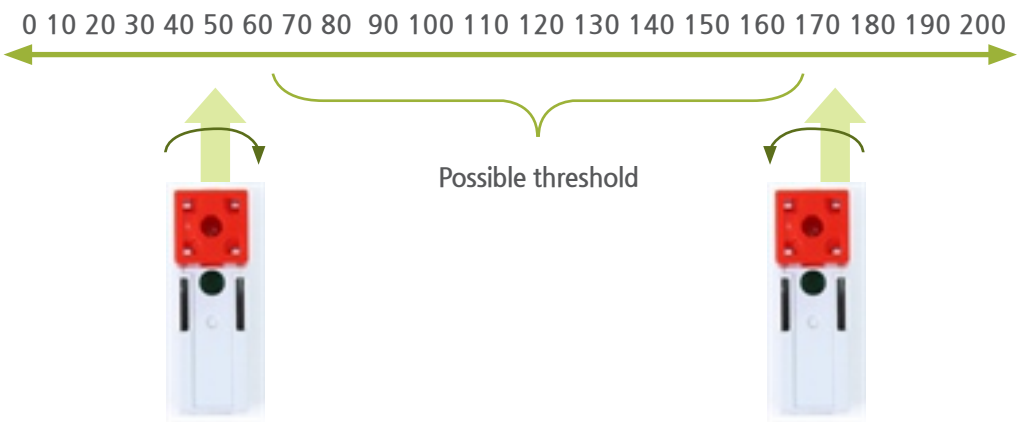


The motor will turn on if the volume sensor value goes below 100. If not, then the motor will turn off.

Threshold value

Threshold

Try with your motor! You will see the motor stopping when the value of the volume sensor goes below 100. You can set the value anywhere between 60 and 170 if it is smaller than the value you see when you rotate the sensor clockwise.



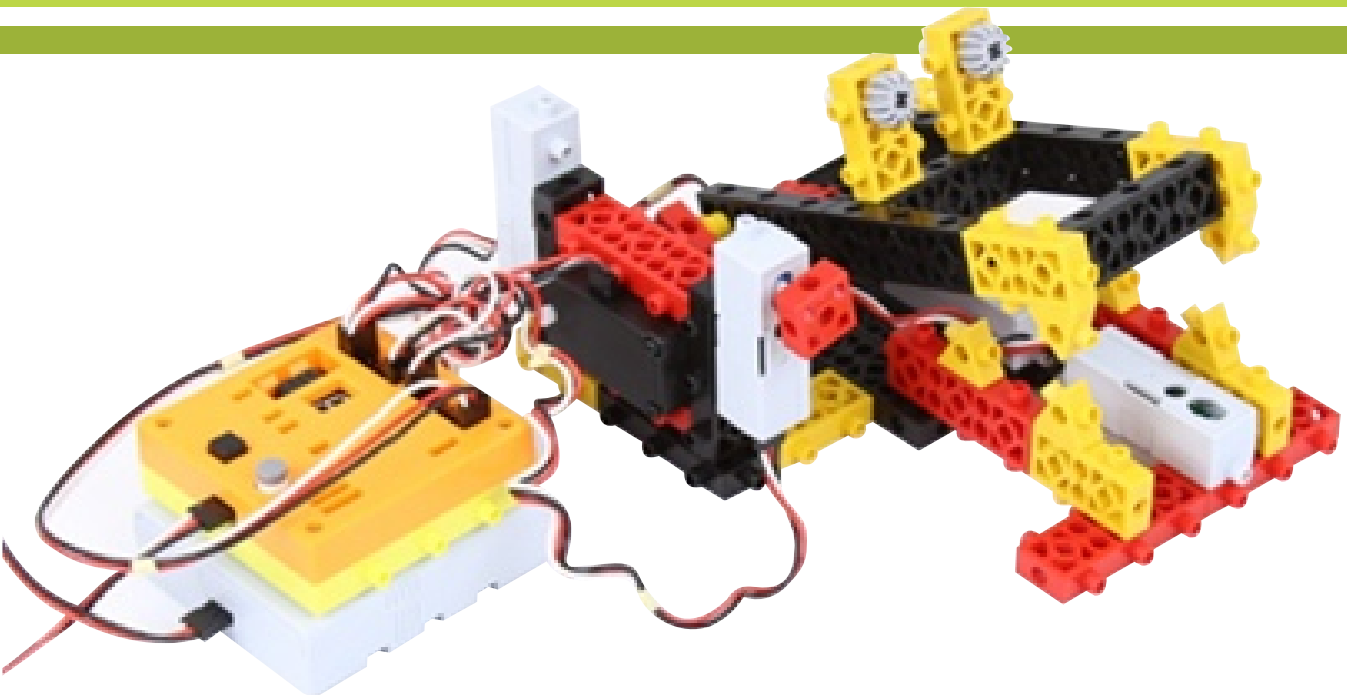
Try making different models and operate them using threshold values that you choose.

SCRATCH CODING KIT

Logic boost

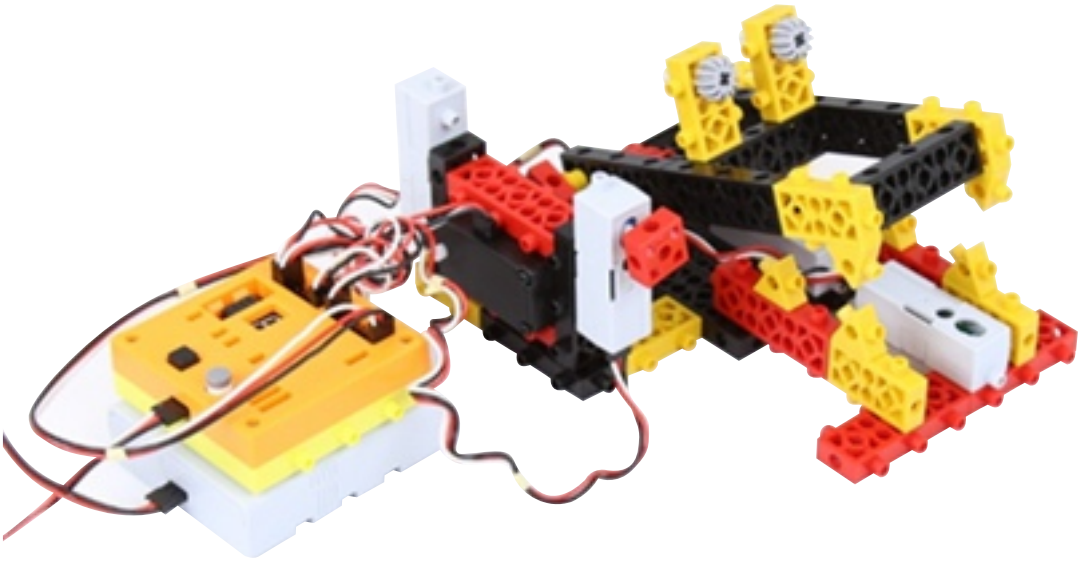
Alligator robot

LESSON 11



How to Make a Robot

Alligator robot



Let's make a scary alligator robot that opens and closes its mouth. We are going to use the servo motor to do this. **(We have to type in the angle for the servomotor when closing its mouth).**

The robot's mouth will be open in the beginning and close when the brightness sensor is covered. When the button sensor is pressed, it will open its mouth. Using the volume sensor, we are going to control the angle at which it opens its mouth. Using the sound sensor, we are going to produce effect sounds that can become a game.
We have learned how to operate a DC motor using the analog value of the volume sensor. Now, we are going to operate the servo motor using the brightness sensor. When the brightness sensor is covered, the analog value increases and when it is not covered, the analog value decreases. Remember this and start coding.

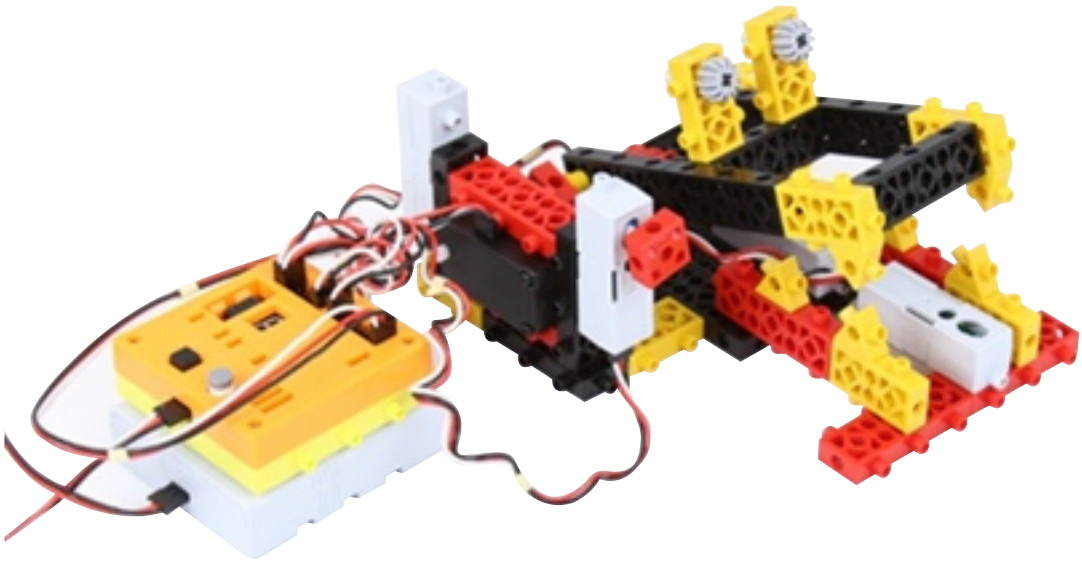
Let's start making the alligator robot.

CLASS 2

How to Make a Robot

Alligator robot

Let the brightness sensor's digital value be the input value to make the robot move with the servomotor.



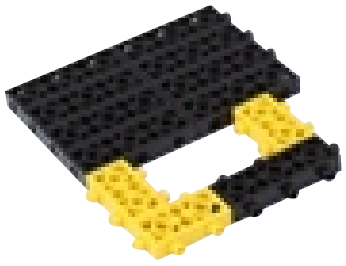
Materials needed for the alligator robot

											
Rubi 8 x 2	Rubi 4 x 6	Rubi 2 x 4	Rubi 0 x 5	Rubi 7 x 5	Rubi 6 x 1	Mini 2 x 9	Triangle x 8	Servo x 2	Vertical connector x 2	Middle connector x 4	Sawtooth 12 x 2
											
Motor connector x 1	A 23 x 4	A 96 x 1	Servo motor x 1	Main board 128 x 1	Battery case x 1	CDS SENSOR x 1	BUTTON SENSOR x 1	SONG GENERATOR x 1	POTENTIOMETER x1	3pin connector x 6	

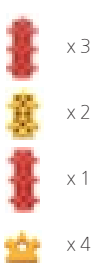
How to Make a Robot

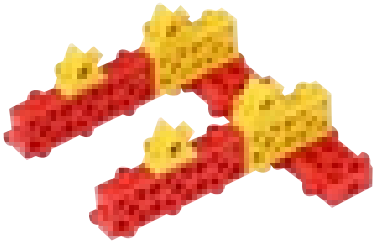
Alligator robot

1



2

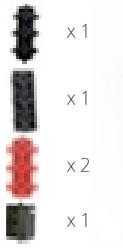




3

Combine figure 1 and figure 2

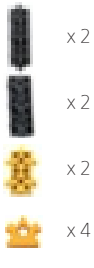
4



5

Combine figure 3 and figure 4

6



CLASS 2

How to Make a Robot

Alligator robot

7

8

9

Combine figure 6,7,8

10

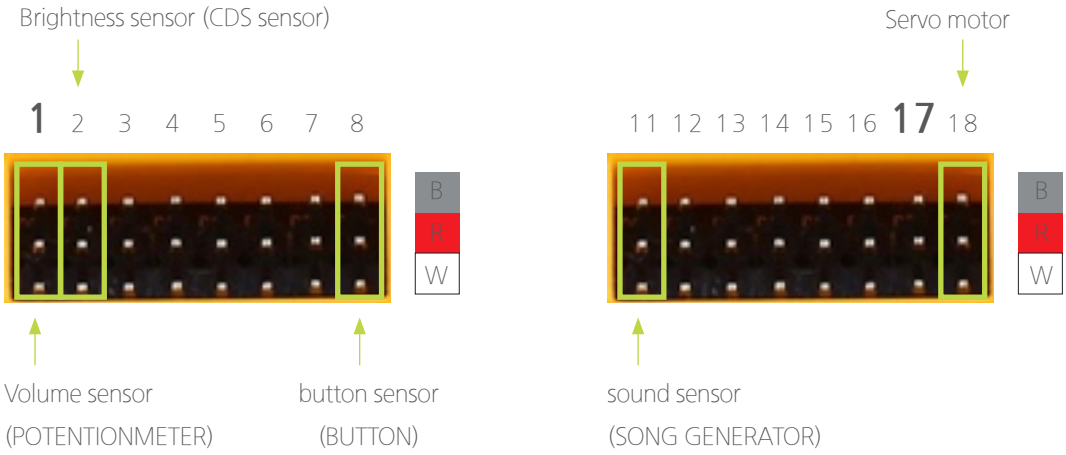
Combine figure 5 and figure 9

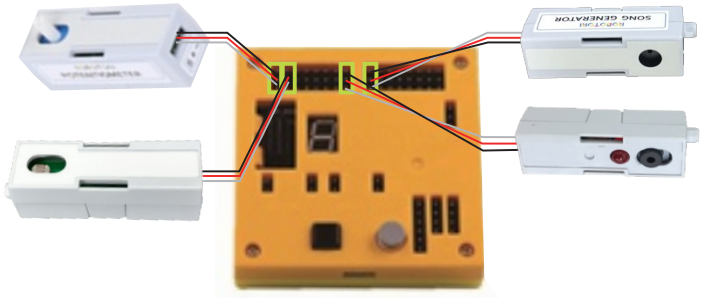
11

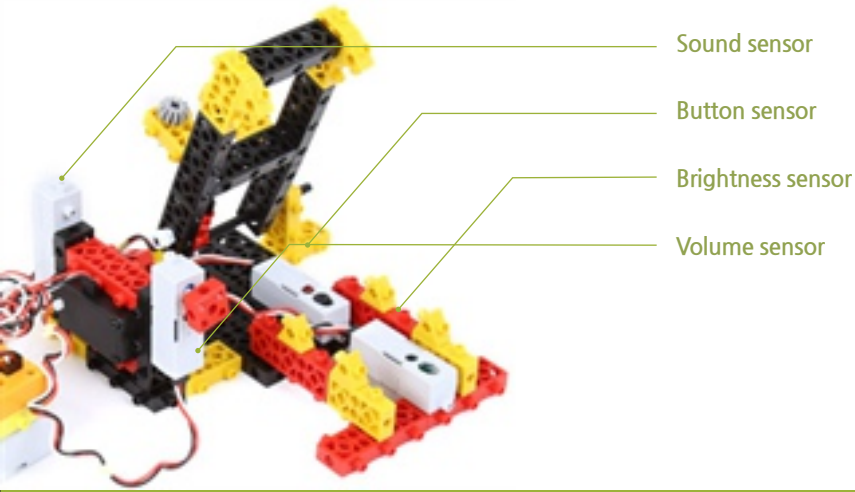
Complete

Connecting to the main cell

Alligator robot





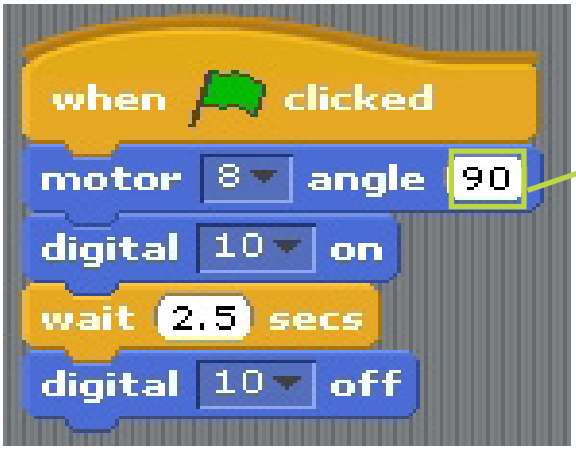


CLASS 2

Experiencing the alligator robot coding

Sound sensor and opening the alligator's mouth

First, code an explanation of the settings of the alligator in the beginning. When the script starts, the sound sensor should start playing music and the servo motor has to open the robot's mouth. Remember what we learned before and make a script.
Look at the script below.



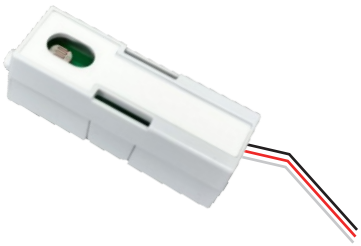
Carefully set the angle at which the alligator opens its mouth.

It might be different depending on the user, so check before putting in any number.

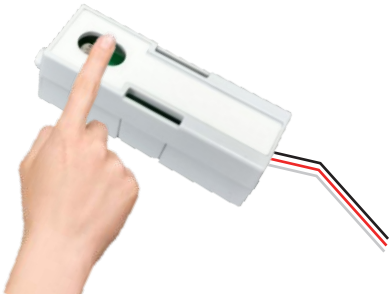
Check out the script above! When the script starts, the servo motor will open the robot's mouth and the sound sensor will play its music for 2.5 seconds. This was the code for the alligator's basic position before doing anything. It's really simple, but it's important to set a beginning position for robots when coding.

Experiencing the alligator robot coding

Brightness sensor and Opening/closing the alligator's mouth



If the brightness sensor is uncovered, the alligator should have an open mouth.



If the brightness sensor is covered, the alligator should have a closed mouth.

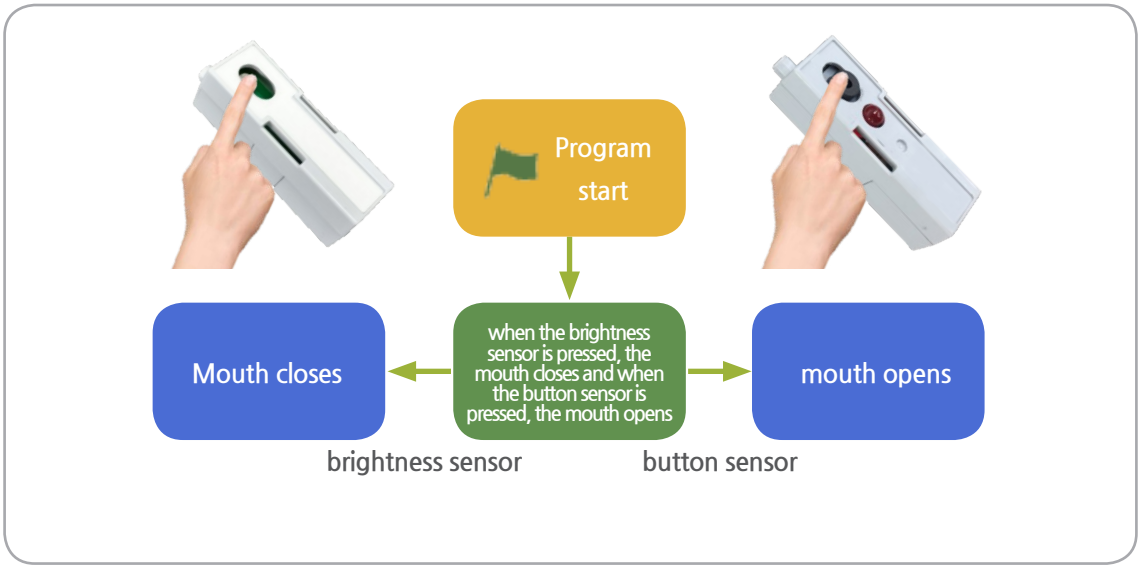
The brightness sensor should be used to close the the robot's mouth, and the button sensor should be used when to open the robot's mouth. Do you rememner how the analog value changes when the brightness value is covered? Keep in mind that the analog value increases when the brightness value is covered and start coding.

CLASS 2

Experiencing the alligator robot coding

Brightness sensor and Opening/closing the alligator's mouth

To show how the brightness sensor's input value and the button sensor's input value act on the servo motor in flowchart, it looks like below. Before, we checked how DC motor works with the windmill, but this time we have to check how the servo motor works.



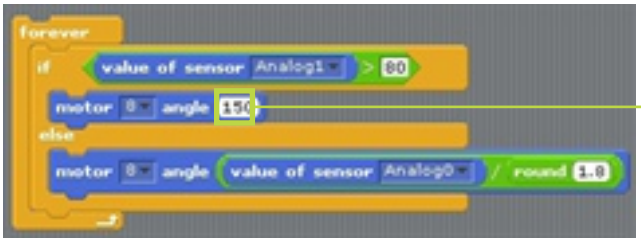
The program will keep checking the input value of the brightness and button sensor. The brightness sensor's value increases when it's covered and it decreases when it's not covered.

We have to set it so that when you cover the brightness sensor, the servo motor has to be at an angle in which the robot closes its mouth. When the button sensor is pressed, the mouth should open at an angle that is set by the volume sensor.

Experiencing the alligator robot coding

Brightness sensor, volume sensor, and servomotor

Now, we are going to make a code that will control the angle of the servomotor using the brightness and volume sensors. When the brightness sensor is covered, the mouth should close and when it's not covered, it has to set a max angle for the alligator to open its mouth using the volume sensor. Make a script like below!



Set an angle in which the alligator closes its mouth. It might differ between users.

If you see the script above, the servo motor's angle will be 150 degree when the value of analog1 sensor is larger than 80. Analog port 1 is connected with the brightness sensor. Therefore, when the brightness sensor is covered, it will operate the servo motor at a 150 angle degree.

If the analog1 sensor value is smaller than 80, it says the servomotor will be in at the 'analog0 sensor value/1.8 rounded' angle. When the brightness sensor is not covered, you should be able to use the volume sensor to control the mouth's angle. But, since we can't turn the volume sensor more than the angle where the robot can open its mouth, we have to divide the volume sensor's analog value with the right number to control the servo motor's angle.

The script above divided the volume sensor's analog value (analog0 sensor value) with 1.8 and rounded it. The 'rounded' operator can be found under the calculation tab and makes the calculated value into an integer. 255 is the largest number of the volume sensor's analog value, and if we divide it by 1.8, the angle of the closing mouth will not gog over the servo motor's angle. The important part is that you don't always divide it by 1.8. It depends on the angle you set for the alligator's closing mouth. It can be 1.7, 1.8, or 2.0. When dividing the volume sensor's analog value, make sure you check the angle in which the alligator closes its mouth.

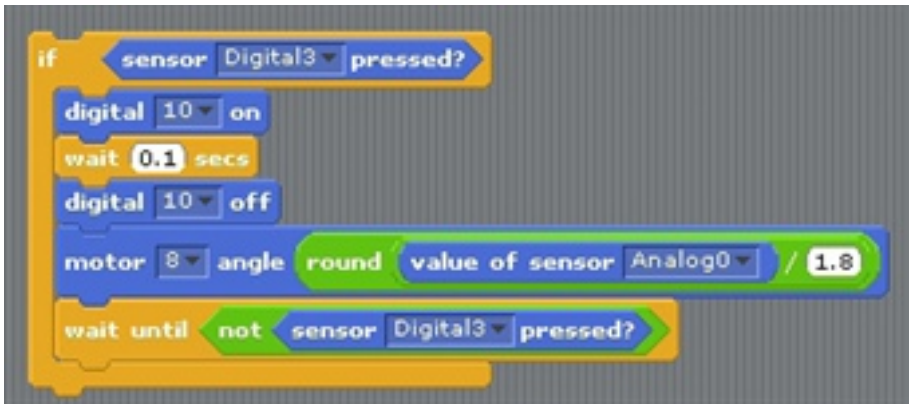
255 ÷ 1.8 = 141.666… → rounded → Smaller than the angle in which the alligator closes its mouth

CLASS 2

Experiencing the alligator robot coding

Button sensor and the servo motor

Now, code for the alligator's mouth movement when the button sensor is pressed.
First, we are going to make the sound sensor produce a short sound whenever the button sensor is pressed. And when the button sensor is pressed, the alligator's mouth will open until it reaches the angle that was set by the volume sensor. Make the script below!



For right now, the button sensor is connected to digital port 3, which is the right end port among all input ports. Make the sound sensor to play its effect sound for 0.1 sec whenever the button sensor is pressed. And, make the alligator open its mouth until it reaches the angle set by the volume sensor when the button sensor is pressed.

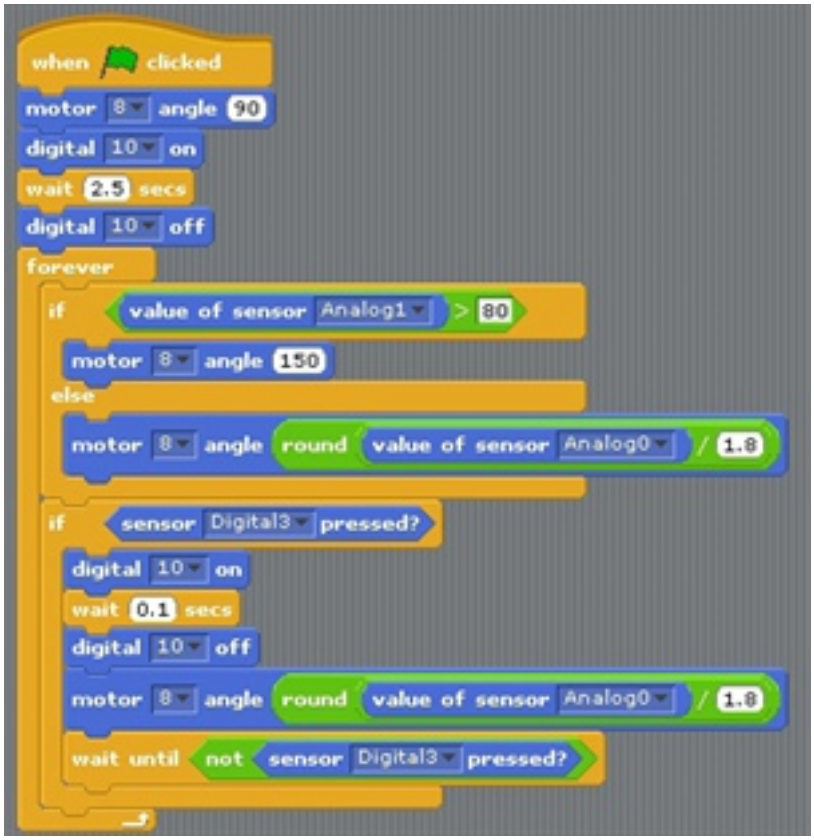
The last block 'wait until…' will be introduced later in the next level. It's simply a block that orders the alligator to wait until the volume sensor is not pressed. There is a block that keeps the sound playing and the mouth open when the button sensor is pressed. Though, if we put 'is the sensor digital 3 pressed? NOT' condition, the mouth will open once and the sound play once until the button sensor is unpressed. Let's learn more about the 'wait' block in level 4.

You just completed coding the alligator robot!
Combine all the scripts you have made into one script.

Experiencing the alligator robot coding

The alligator robot coding

If you combine all the scripts, it will look like this! Look at the whole script and see how each coding block carries out the different tasks.



CLASS 2

Experiencing the alligator robot coding

Using the 'If' phrase with the alligator robot

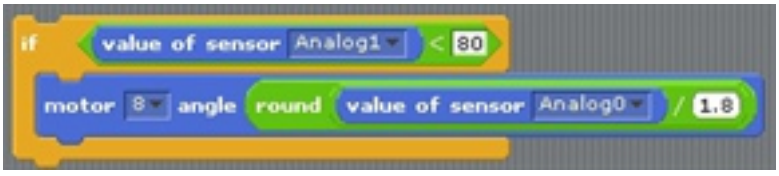
We just made the script using a 'if..or else..' phrase. Now, let's make the script using a 'If' phrase. It will be simple. Think about how you it would code the 'or else' part. Try it below!

Experiencing the alligator robot coding

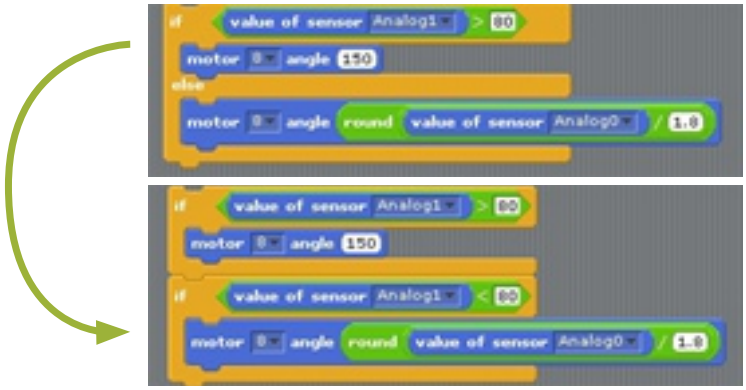
Coding 'If' with the alligator robot

If we controlled the brightness sensor's servo motor with the 'if..or else..' phrase before, we are going to use two 'if' phrases to control the servo motor. First, make a script that will close the alligator's mouth when the brightness sensor's value is big (when the sensor is covered).

The script above makes the alligator close its mouth when the brightness sensor's value is big. Let's see how the servo motor will work when the brightness sensor's value is small. Make the script below.



Did you make the script? The script above opens the mouth in an angle which was set by the volume sensor when the value of the brightness sensor is small. Now, combine these two scripts, and it will look like below! Start the script, and if it doesn't work well, try making it again slowly. It's easy if you think you are dividing 'If..or else..' into two parts.



The whole script can be found at http://www.robotori.com/web_eng -> Moretips -> Manual ->EDU ->Logic boost CODING CLASS 2 ->download whole script

CLASS 2

Experiencing the alligator robot coding

Alligator robot

Activate the program! Try opening its mouth with the button sensor before it closes due to the covered brightness sensor. A completely closed mouth is when the blocks touch each other. If the mouth closes too fast/slow, change the angle using the volume sensor.

Try playing this next game with your friends!

Alligator robot game rules

1. The players will cover the brightness sensor to make the robot close its mou
2. Before it completely closes its mouth, press the button sensor to open its mouth. When you press the button sensor, the brightness sensor should be covered.
3. Using the volume sensor, set how much you will open the robot's mouth by. It will be easier as if it opens wider.
4. Take turns in pressing the brightness and button sensors. You lose if you let the robot completely close its mouth!

SCRATCH CODING KIT

Logic boost

Scorpion root

LESSON 12

CLASS 2

exercise

Scorpion robot

Now, you are going to use different motors and sensors to make a model.
Try coding its movement using what we have learned about brightness sensors, LED sensors, DC motors, and servo motors. The name of this robot is Scorpion robot. It looks like a scorpion, and it can move its tongs up and down using the servo motor.

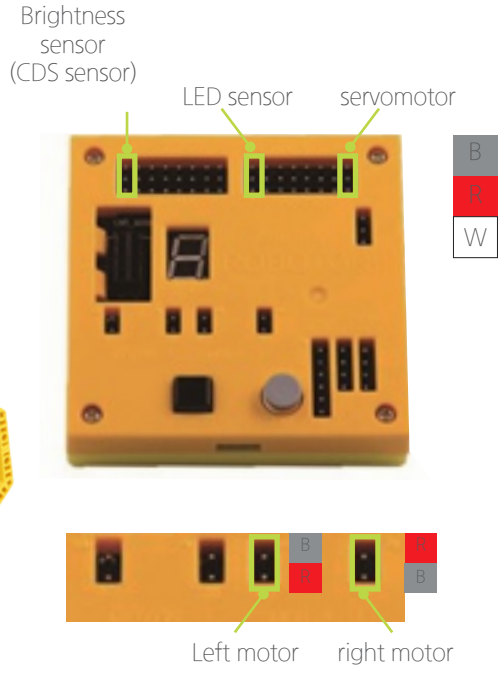
Combine one input sensor (brightness sensor), one output sensor (LED sensor), two DC motors, and one servomotor to make the robot.

Whenever the brightness sensor is covered, the servo motor on its tongs will operate and turn on the LED sensor. With keyboard arrow keys, you can control the robot's direction. Can you imagine how its movements will look like?

exercise

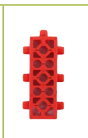
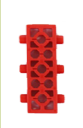
Scorpion robot

Make the scorpion robot.



※ Careful when plugging in the cables

Materials needed for Scorpion robot

												
Diamond H6 x 1	Diamond V8 x 6	Diamond V6 x 3	Rubi 8 x 9	Rubi 7 x 12	Rubi 4 x 7	Rubi 6 x 2	Rubi 2 x 4	Rubi 0 x 6	Mini 2 x 10	Mini 1 x 3	Curve x 8	Servo x 2
												
Link x 4	Sawtooth 12 x 4	Crown x 2	Sawtooth 36 x 6	A45 x 8	A64 x 3	Battery case x 1	DC motor x 2	connector x 4	Servo motor x 1	CDS ENSOR x 1	LED SENSOR x 1	

CLASS 2

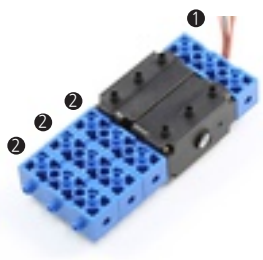
exercise

1 1-1

x 2

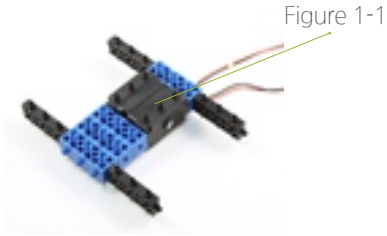
x 1

x 3



1-2

x 4



Assemble figure 1 with four of the cells with 8 bumps

2 2-1

x 2

x 1

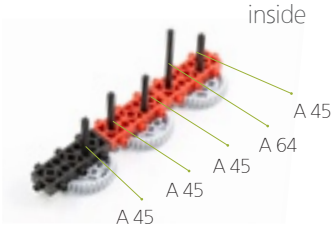
x 1
A 64

x 2

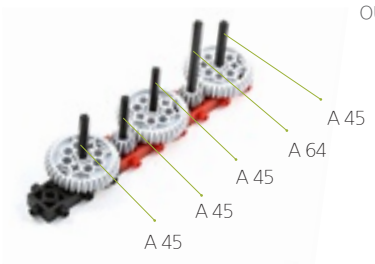
x 4
A 45

x 3

inside



outside



※ Careful with the locations of rods

2-2

x 2

x 1

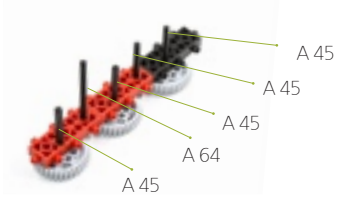
x 1
A 64

x 2

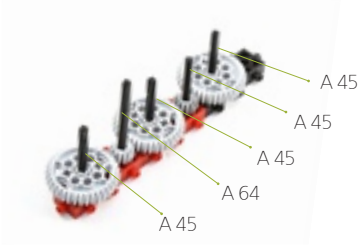
x 4
A 45

x 3

inside



outside



※ Careful with the locations of rods

3 3-1

Figure 2-1

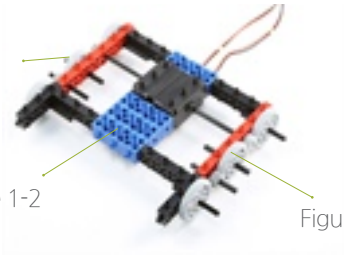


Figure 1-2

Figure 2-2

3-2

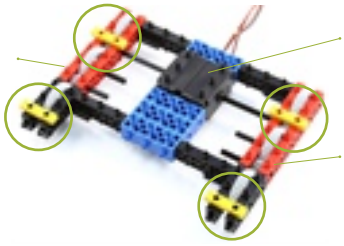
x 2

x 4

exercise

3 3-3

x 4

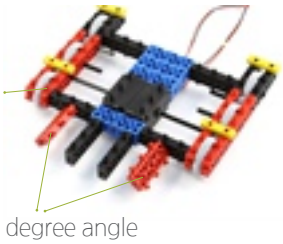


4 4-1

x 2

x 2

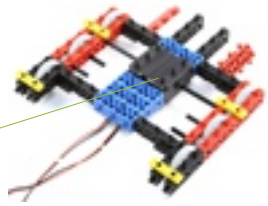
Figure 3-3



45 degree angle

4-2

Figure 4-1



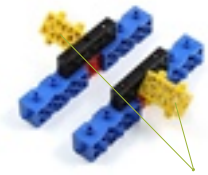
5 5-1

x 4

x 2

x 2 x 2

(A) (B)



Assemble in 45 degree angle

5-2

Figure 5-1 (A)

6 6-1

x 2 x 1

x 3 x 1

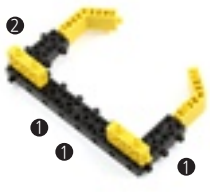
x 2

6-2

x 2 x 1

x 2

x 3

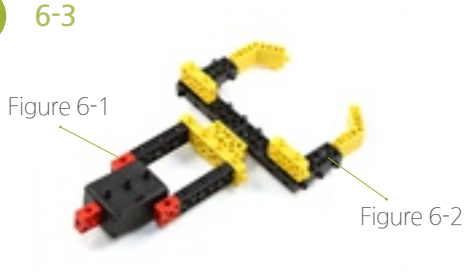


CLASS 2

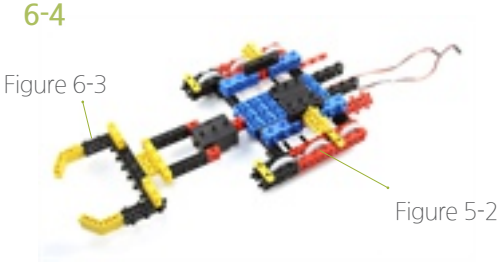
exercise

6

6-3



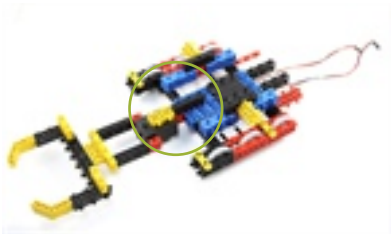
6-4



6-5

x 1

x 1



6-6

x 1 x 3

x 1

x 1

7

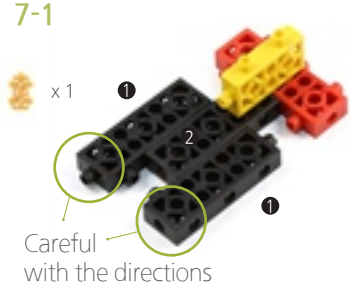
7-1

x 2

x 1

x 1

x 1



7-2

x 1
A 64

x 1

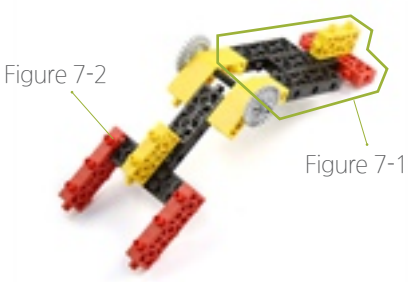
x 2

x 2

x 2

x 2

7-3



7-4

x 1

x 4

x 2

exercise

7 7-5

Figure 7-3

Figure 7-4

Assemble figure 7-3 and figure 7-4

7-6

Figure 6-5

Figure 7-5

Complete

CLASS 2

exercise

Coding the scorpion robot

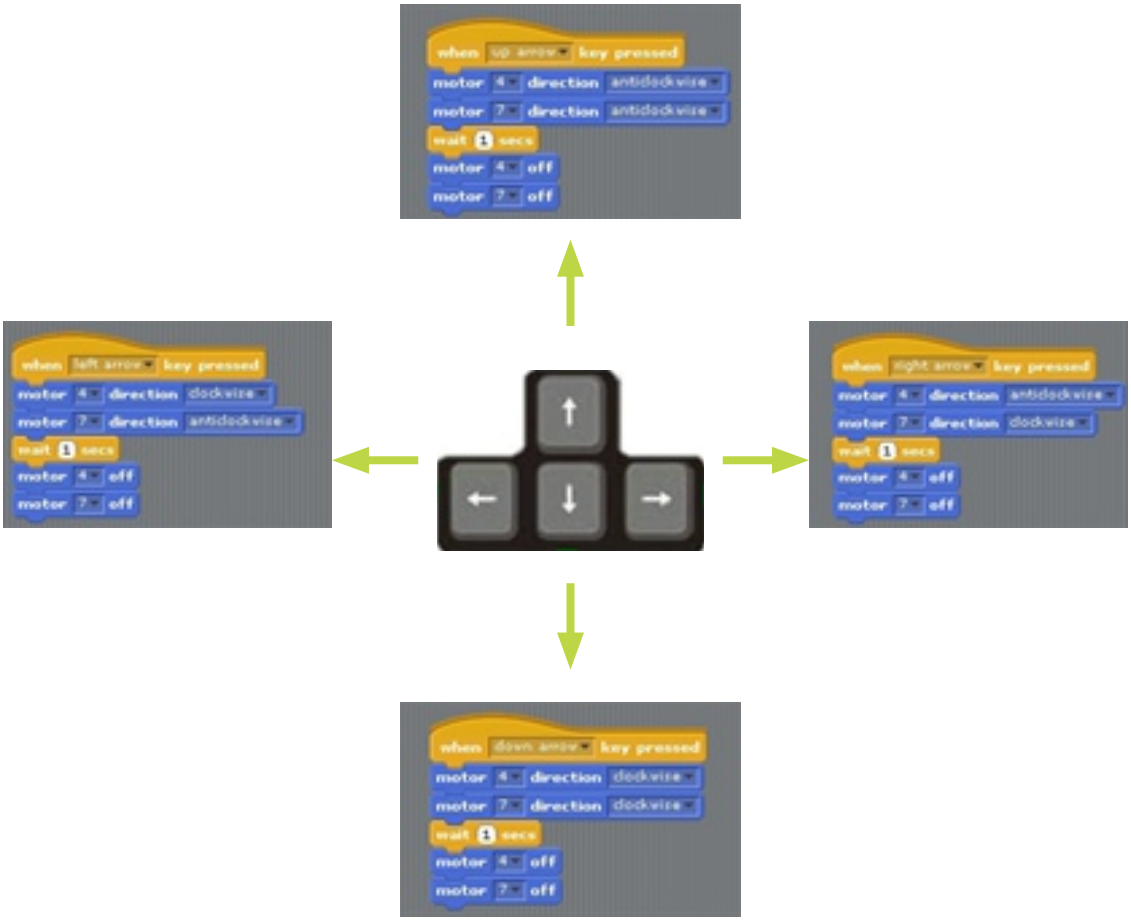
Try coding the robot by yourself.

When the brightness sensor is covered, the tongs should go up and the LED sensor should light up. DC motor has to be controlled with the keyboard arrow keys. When the brightness sensor is not covered, the tongs have to move down and hte LED sensor should turn off. Use an 'If..or else..' phase, sensor output, and motor activation blocks to make a script!

exercise

Using the keyboard to control the scorpion robot

Code so that the scorpion robot will move with the DC motor when arrow keys are pressed.
Do you remember the lizard robot from level 1? Consider the scorpion robot moving front/back and right/left to make a script. If you don't remember well, refer to the scripts below.

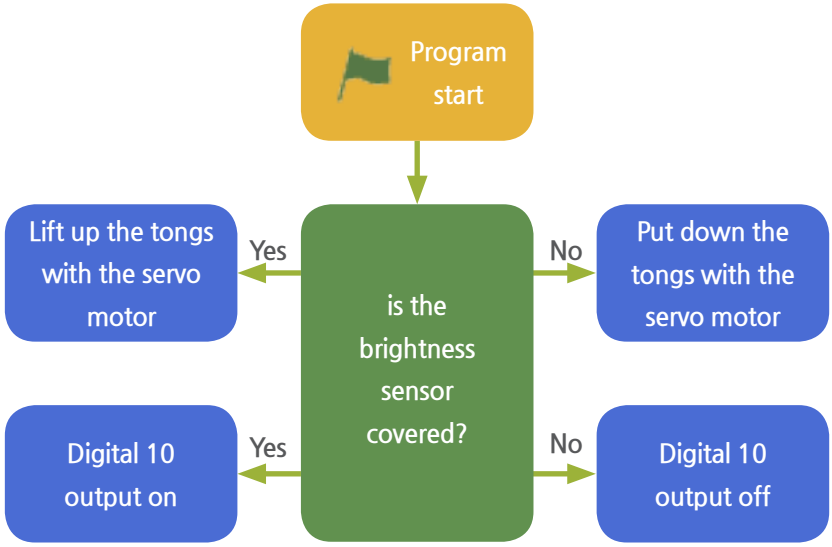


CLASS 2

exercise

Scorpion robot

Look at the flowchart below.
When the program is started, the servo motor should lift up the tongs and turn on the digital output (LED sensor) as the brightness sensor gets covered.



Do you remember what we learned regarding the brightness sensor?

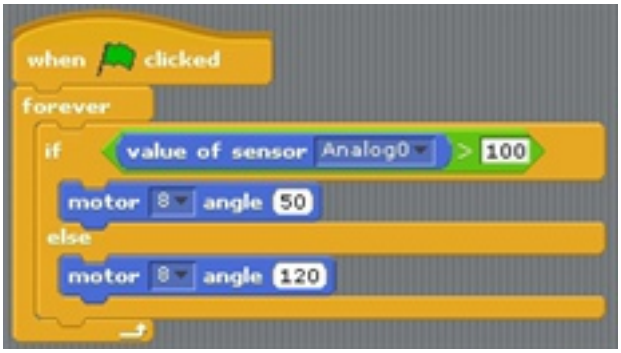
First, as the activation block, we will be using the 'if..or else..' block after the start of the script. The condition would be whether the brightness sensor is covered/uncovered. If it is satisfied (brightness sensor is covered), the servo motor lifts up the tongs and the LED sensor will turn on by turning on the digital output. If it is not satisfied (brightness sensor is uncovered), the servo motor will put the tongs down and turn off the LED sensor by turning off the digital output.

It's a simple script; you will be able to make it easily!

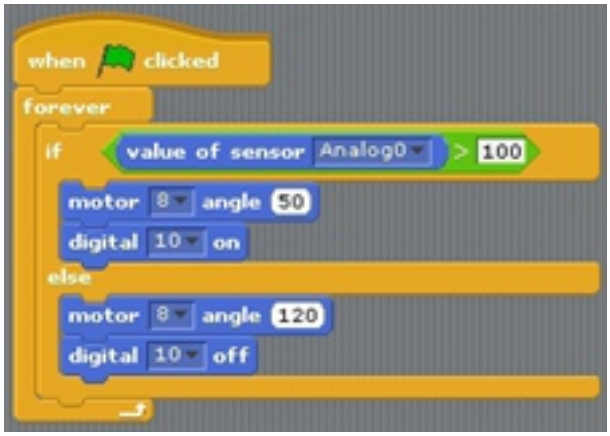
exercise

Scorpion root

Let's make the servo motor lift up the catchbot whenever the brightness sensor is pressed. Refer to the alligator robot script.



The script above operates the servo motor whenever the brightness sensor is pressed. Set an angle for the servo motor when lifting and setting down the catchbot. Let's add the digital output on/off now.



Now, the scorpion robot will lift the catchbot up and turn on the LED when the brightness sensor is pressed. Not only does it move the catchbot, it also produces different movements with the output sensor. Let's make the scorpion robot script using different coding blocks. Move on to the next page.

CLASS 2

exercise

Using the 'if' phrase with the scorpion robot

We just made a script using the 'if..or else..' phrase.

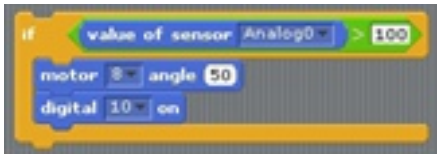
Now, make a script using the 'if' phrase only. It's going to be simple. Think about how you will make the 'Or else..' part.

Make a script below.

exercise

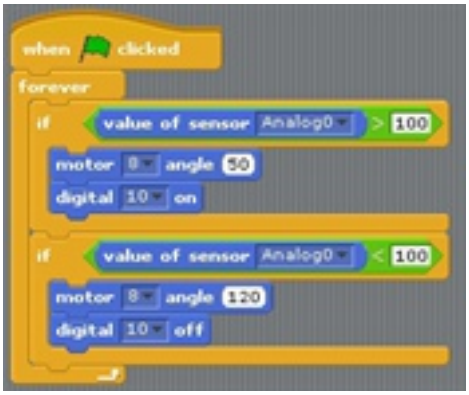
Coding the scorpion robot with an 'if' phrase

Since we controlled the servo motor and LED sensor with 'if ..or else..', we are now going to use two 'if' phrases to control the servo motor and LED sensor. First, make a script that will lift the tongs up and turn the LED sensor on when the brightness sensor value is big (when it's covered).



The script above will lift the tongs up and turn on the LED sensor when the value of the brightness sensor is big. Now, let's see how the servo motor and the LED sensor will work when the value of the brightness sensor is small. Make the script you see below.

Did you make the scipt? The script above sets the tongs down and turns off the LED sensor when the value of the brightness sensor is small. Now, combine the two scripts, and it will look like the one below! Start the script and try again slowly if it does not work well.



CLASS 2

exercise

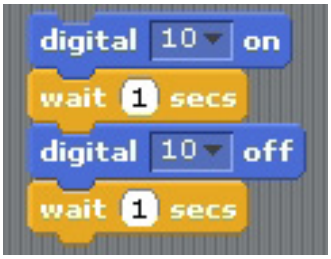
Changing the digital output of the scorpion robot

We just made the LED sensor turn on when the tongs are lifted and turn off when they're down.
Now, we are going to fix the script to make the LED sensor flash when the tongs are lifted and turn off when they are put down. Do you remember the script we made to make the LED sensor flash?
Make a script on the space below.

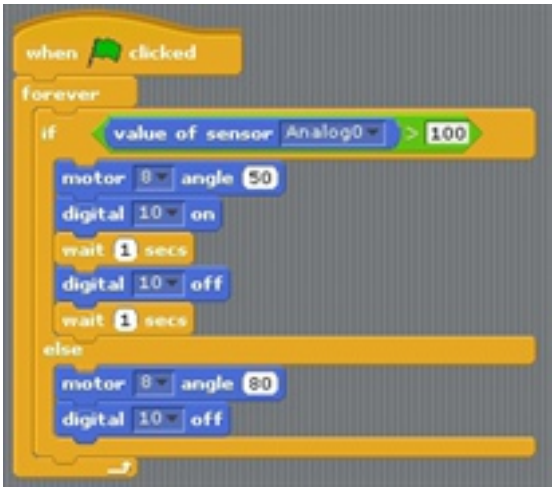
exercise

Changing the digital output of the scorpion robot

Do you remember how to make a script that makes the LED sensor flash?
The script that will make it flash every second looks like below. Compare this with yours.



Instead of keeping the LED sensor on when the tongs are lifted, make it flash every second. If you connect the whole script, it will look like the one below! Start the script and try again if it doesn't work well.

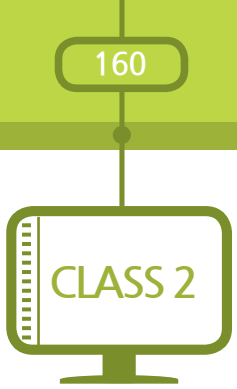


The whole script can be found at http://www.robotori.com/web_eng -> Moretips -> Manual ->EDU ->Logic boost CODING CLASS 2 ->download whole script

CLASS 2

MEMO

MEMO



MEMO

FCC Information to User

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Caution

Modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

FCC Compliance Information : This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment.