



Let me remember



Target

- Learn the application of variables.
- Complete an LED lighting scheme using multiple variable storage and applications.
- Understand how the robot performs memory storage and retrieval.



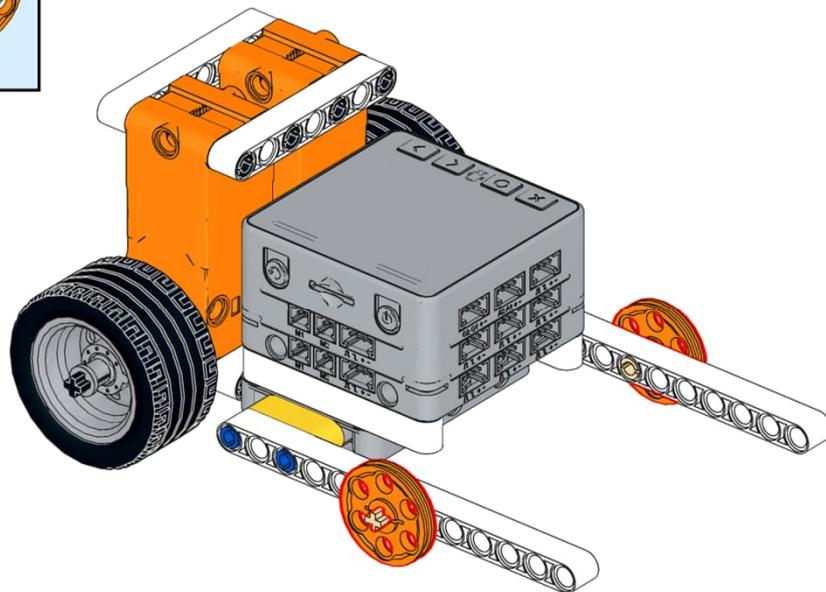
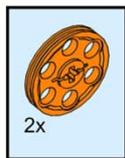


01 Assembly



Assembly

23

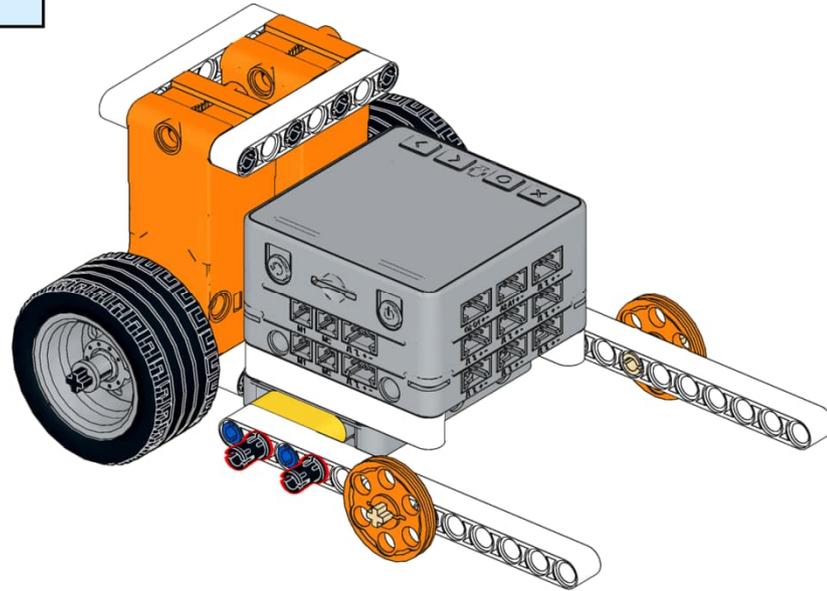


See Lesson 1
for complete
instructions



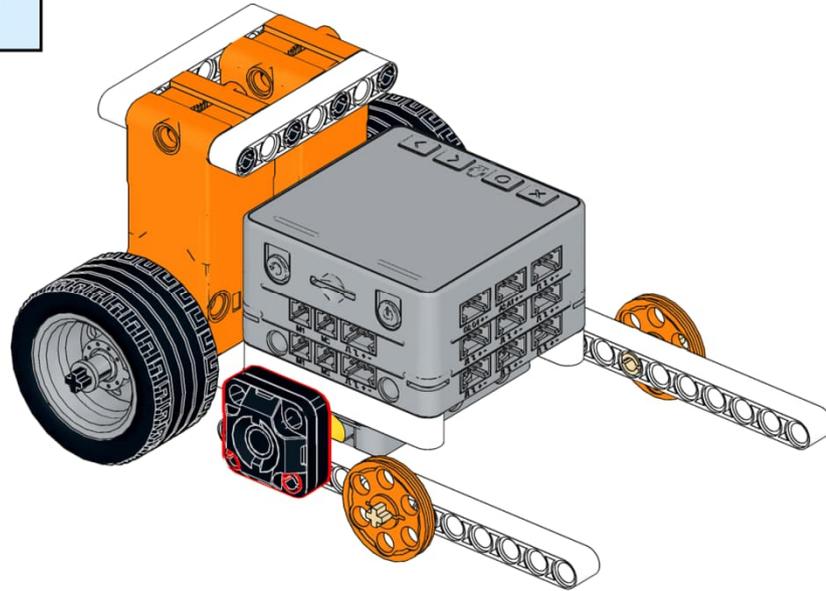
Assembly

24



Assembly

25



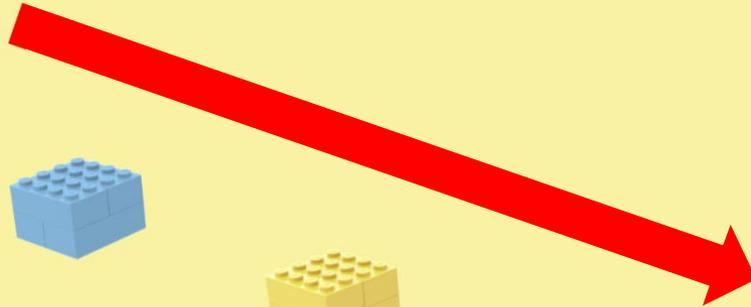


02 Task 1



Task 1

- Detect two colors, and then light up the color LEDs in **sequence**.



Task Segmentation

- First, complete the color detection.

Detect the color



Not detect
the color

Detect
the color

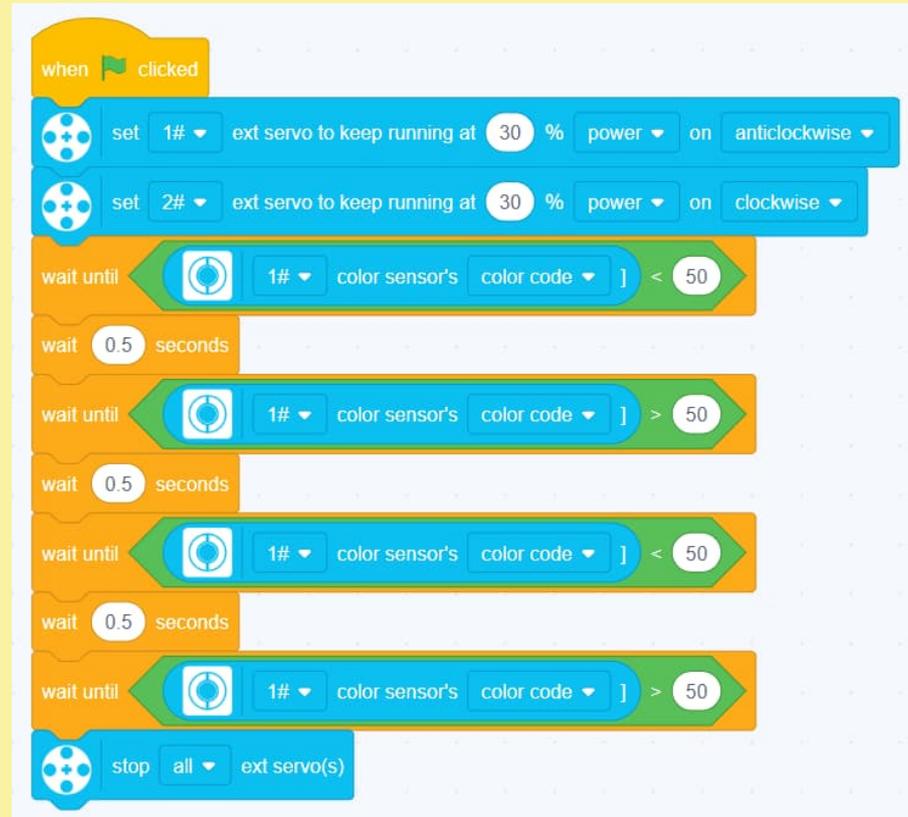


Not detect
the color



Task Segmentation

- Referrable program



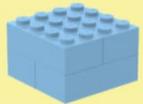
```
when clicked
  set 1# ext servo to keep running at 30% power on anticlockwise
  set 2# ext servo to keep running at 30% power on clockwise
  wait until [1# color sensor's color code] < 50
  wait 0.5 seconds
  wait until [1# color sensor's color code] > 50
  wait 0.5 seconds
  wait until [1# color sensor's color code] < 50
  wait 0.5 seconds
  wait until [1# color sensor's color code] > 50
  stop all ext servo(s)
```

The image shows a Scratch script for controlling two servo motors. It starts with a 'when clicked' event. The first two blocks set servo 1 to 30% power anticlockwise and servo 2 to 30% power clockwise. The script then enters a loop of color sensing on servo 1. It waits until the color code is less than 50, then waits 0.5 seconds, then waits until the color code is greater than 50, then waits 0.5 seconds, and repeats this sequence. Finally, it stops all external servos.

Task Segmentation

- Use variables to store the two colors.
- After the robot stops, read, evaluate, and display the colors.

variable a



variable b



Read variable a
Read variable b



Task Segmentation

- Referrable program

Color Storage

```
when clicked
  set all lights color to closed
  set 1# ext servo to keep running at 30% power on anticlockwise
  set 2# ext servo to keep running at 30% power on clockwise
  wait until [1# color sensor's color code] < 50
  set a to [1# color sensor's color code]
  wait 0.5 seconds
  wait until [1# color sensor's color code] > 50
  wait 0.5 seconds
  wait until [1# color sensor's color code] < 50
  set b to [1# color sensor's color code]
  wait 0.5 seconds
  wait until [1# color sensor's color code] > 50
  stop all ext servo(s)
```

Color reading and judgment

```
if a = 5 then
  set all lights color to yellow
  wait 2 seconds
if a = 4 then
  set all lights color to blue
  wait 2 seconds
wait 0.5 seconds
if b = 5 then
  set all lights color to yellow
  wait 2 seconds
if b = 4 then
  set all lights color to blue
  wait 2 seconds
```

Precautions

```
if (a = 5) then
  set all lights color to yellow
  wait 2 seconds
if (a = 4) then
  set all lights color to blue
  wait 2 seconds
wait 0.5 seconds
if (b = 5) then
  set all lights color to yellow
  wait 2 seconds
if (b = 4) then
  set all lights color to blue
  wait 2 seconds
```

Note: Color code 5 is actually green. When the color sensor detects a yellow block, color code 5 appears when the distance is 1-3 cm, and the correct color code 7 only appears when the distance is less than 1 cm.

To accurately distinguish between yellow and green, let's make a slight structural adjustment.



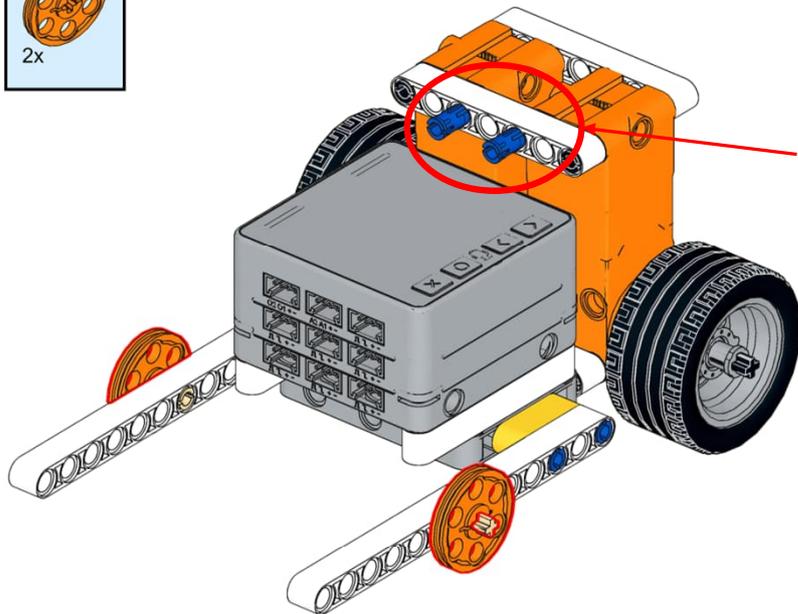
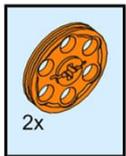


03 Task 2



Assembly

23

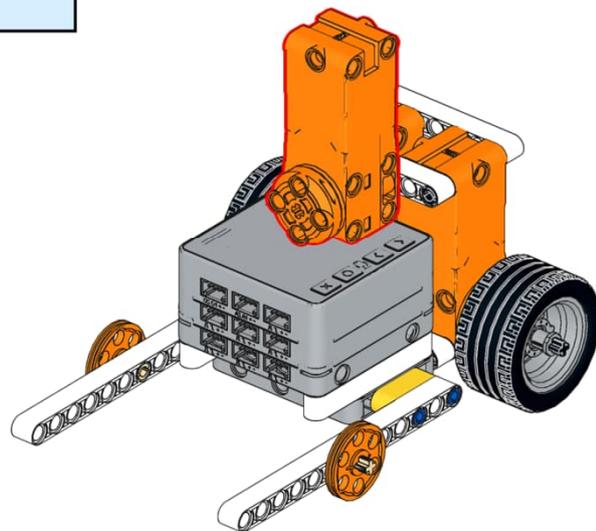
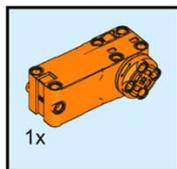


Replace the two-slot pin with a three-slot pin.



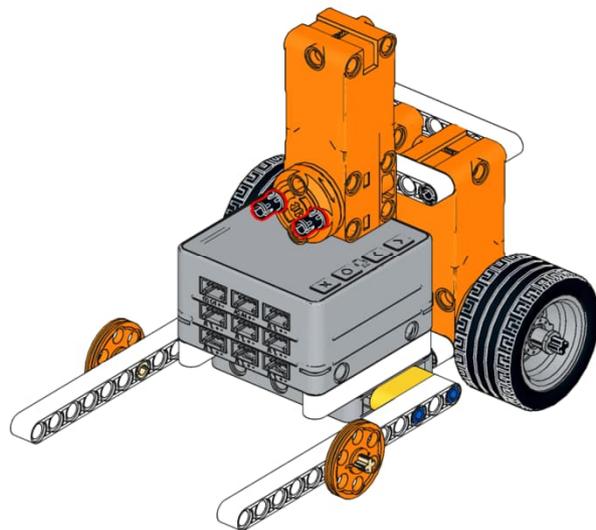
Assembly

24



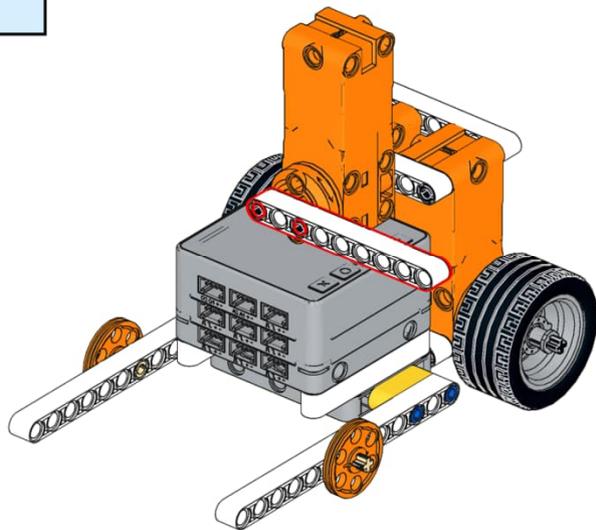
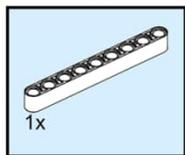
Assembly

25 
2x



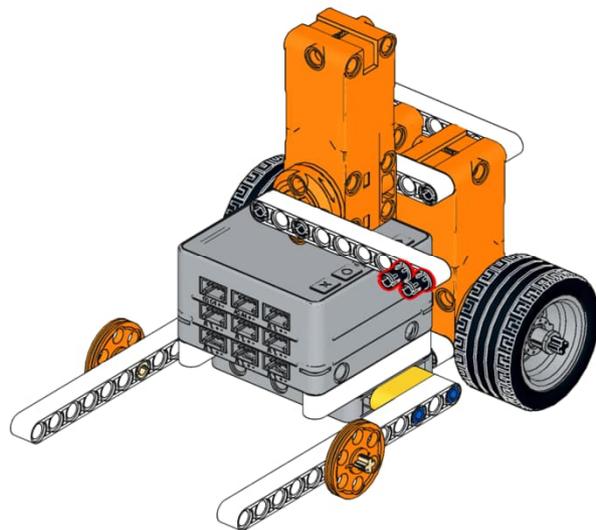
Assembly

26



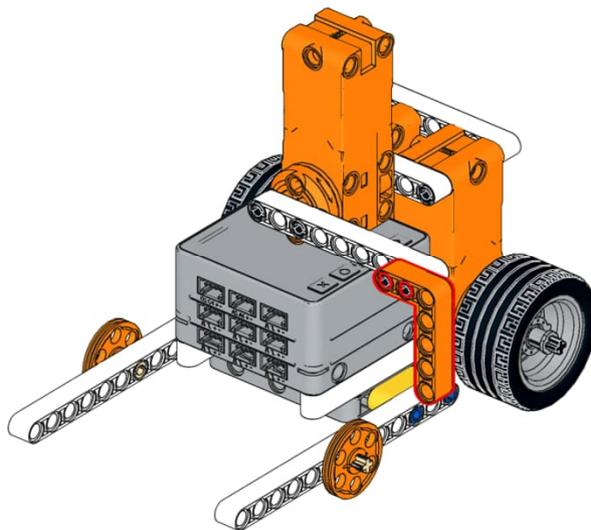
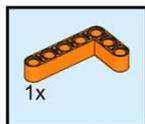
Assembly

27



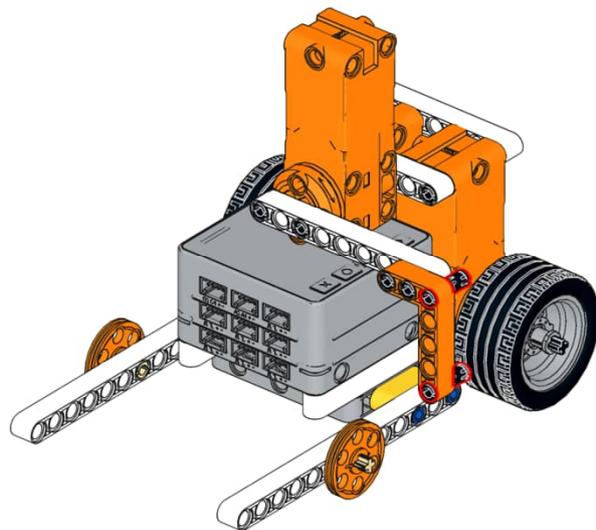
Assembly

28



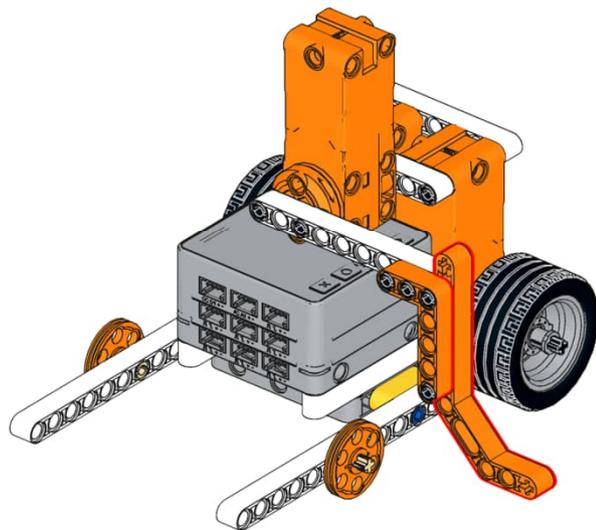
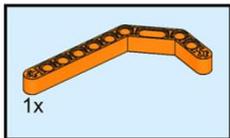
Assembly

29



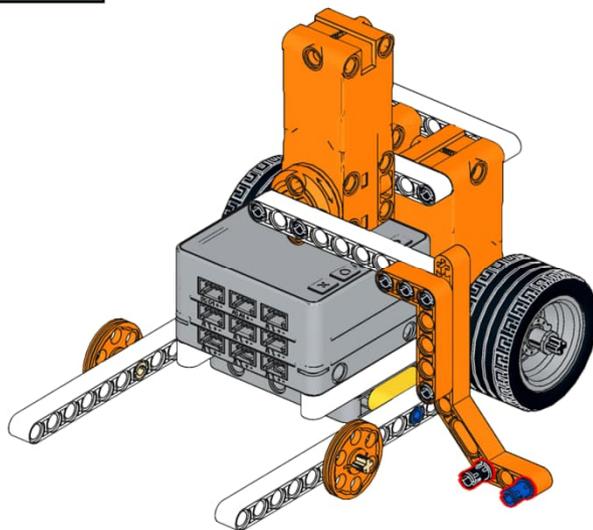
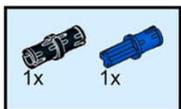
Assembly

30



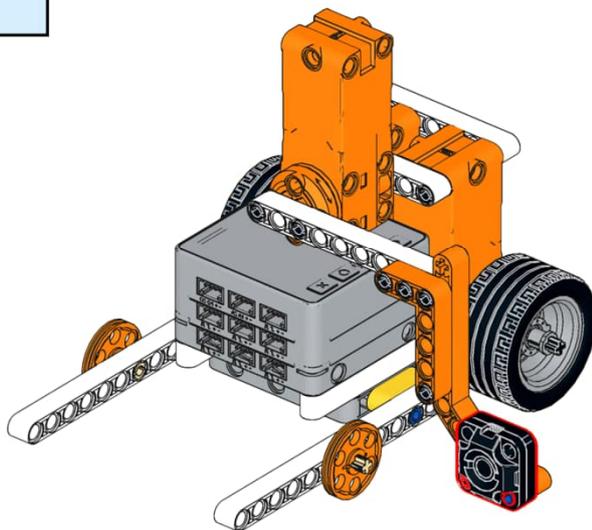
Assembly

31



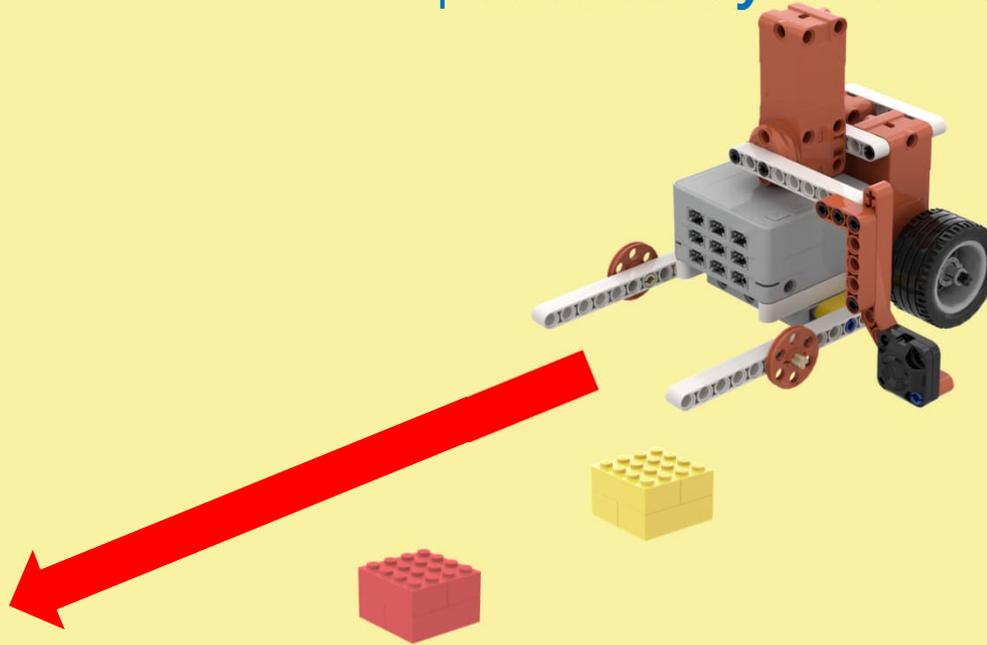
Assembly

32



Task 2

- Task 2: Detect red and yellow. After detection, light up the color LEDs in the order specified by **the teacher**.



Task Segmentation

- Before starting, raise the robotic arm to ensure the robot meets the size requirements. After starting, lower the robotic arm and align the color sensor with the block.
- When the color sensor detects the color, for more accurate detection, the robot should move forward a small distance, bringing the sensor closer to the block, **with a distance of less than 1 cm.**

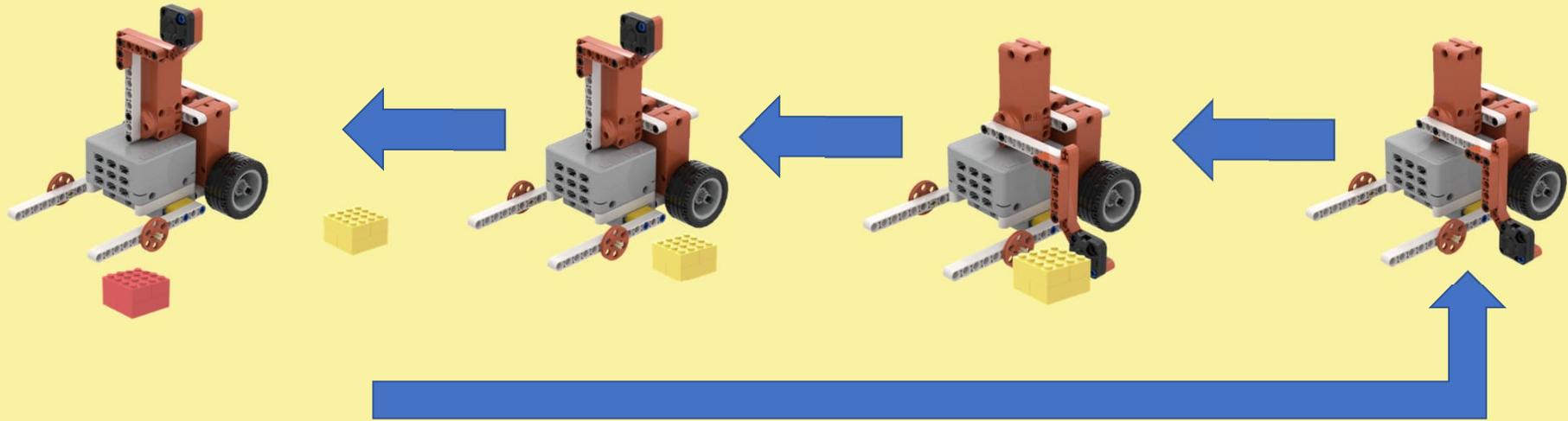


Task Segmentation

- Task Flow

Color detection process

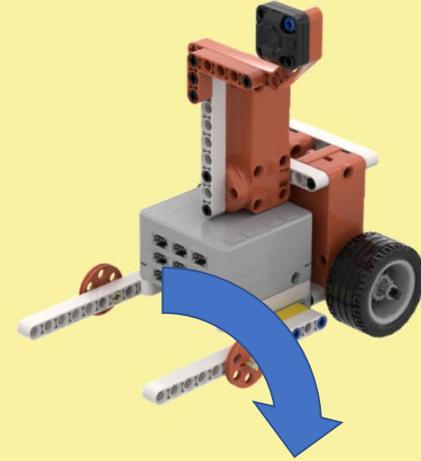
- 4、 Go a little further 3、 Raise the robot arm 2、 Detect the color 1、 Put down the robotic arm



Task Segmentation



It is recommended to use the absolute angle module for controlling the robotic arm. The absolute angle module can be used for operations up to 360 degrees.

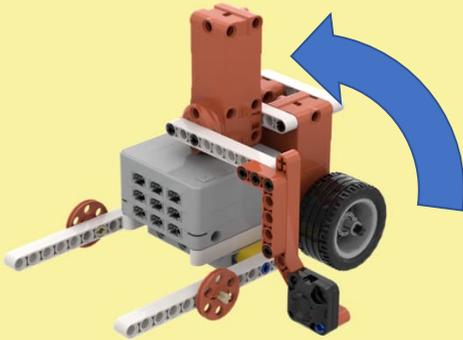


```
set 3# ext servo to rotate absolute angle 0 (0-359) degrees according to the shortest path at 30 (0-100)% speed
wait 0.5 seconds
wait until 3# ext servo done
wait 0.5 seconds
```

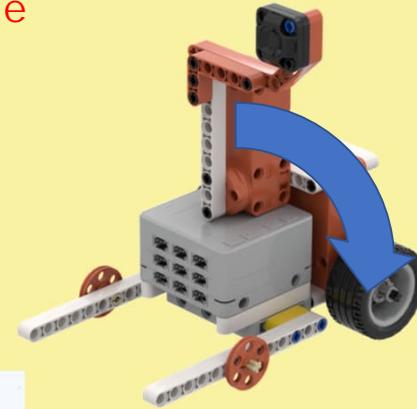
```
set 3# ext servo to rotate absolute angle 90 (0-359) degrees according to the shortest path at 30 (0-100)% speed
wait 0.5 seconds
wait until 3# ext servo done
wait 0.5 seconds
```

Task Segmentation

- Complete " My Blocks" for raising and lowering the robotic arm.



It is recommended to use the absolute angle module for controlling the robotic arm. The absolute angle module can be used for operations up to 360 degrees.



```
define Robot Arm x degree position
set 3# ext servo to rotate absolute angle x 0~359 degrees according to the shortest path at 30 (0~100)% speed
wait 0.5 seconds
wait until 3# ext servo done
wait 0.5 seconds
```

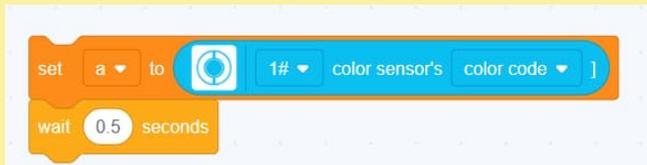
```
define Robot Arm x degree position
```

Task Segmentation

- To accurately detect colors, after detecting a color, move forward a short distance to get close to the color block, and then make the record.



```
set 1# ext servo to keep running at 30 % power on anticlockwise
set 2# ext servo to keep running at 30 % power on clockwise
wait until 1# color sensor's color code < 50
wait 0.3 seconds
stop all ext servo(s)
wait 0.5 seconds
```



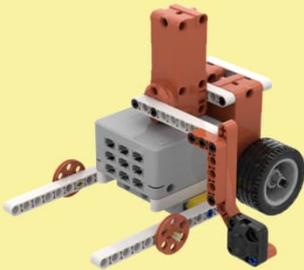
```
set a to 1# color sensor's color code
wait 0.5 seconds
```



Task Segmentation 1

- Task Flow

1、 Put down the robotic arm



```
when green flag clicked
  set a to 0
  set b to 0
  set c to 0
  set all lights color to closed
  Robot Arm 90 degree position
```



Task Segmentation 2

- Task Flow

2、Detecte the color



```
define go forward
  set 1# ext servo to keep running at 30 % power on anticlockwise
  set 2# ext servo to keep running at 30 % power on clockwise
  wait until [1# color sensor's color code] < 50
  wait 0.3 seconds
  stop all ext servo(s)
  wait 0.3 seconds
```

```
go forward
  set a to [1# color sensor's color code]
  wait 0.5 seconds
```

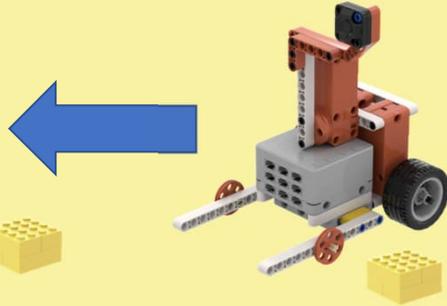
Task Segmentation 3

- Task Flow

4. Cross over the color block.



3. Raise the robot arm



```
define Cross over the color block
  Robot Arm 0 degree position
  set 1# ext servo to keep running at 30 % power on anticlockwise
  set 2# ext servo to keep running at 30 % power on clockwise
  wait 0.5 seconds
  stop all ext servo(s)
  Robot Arm 90 degree position
```

Task Segmentation 4

- Task Flow

```
define colors color
if color = 9 then
  set all lights color to red
if color = 7 then
  set all lights color to yellow
if color = 5 then
  set all lights color to green
if color = 4 then
  set all lights color to blue
wait 2 seconds
```

- After completing the detection and saving, finally display the color.



Task Integration

- " My Blocks" uses the modules from the first three pages. Connect the overall program to complete the detection of multiple color blocks.

```
when green flag clicked
  set a to 0
  set b to 0
  set c to 0
  set all lights color to closed
  Robot Arm 90 degree position
```

Connect the remaining program content with the right side.

```
go forward
set a to [1# color sensor's color code]
wait 0.5 seconds
Cross over the color block
go forward
set b to [1# color sensor's color code]
wait 0.5 seconds
Cross over the color block
go forward
set c to [1# color sensor's color code]
wait 0.5 seconds
colors a
colors b
colors c
```