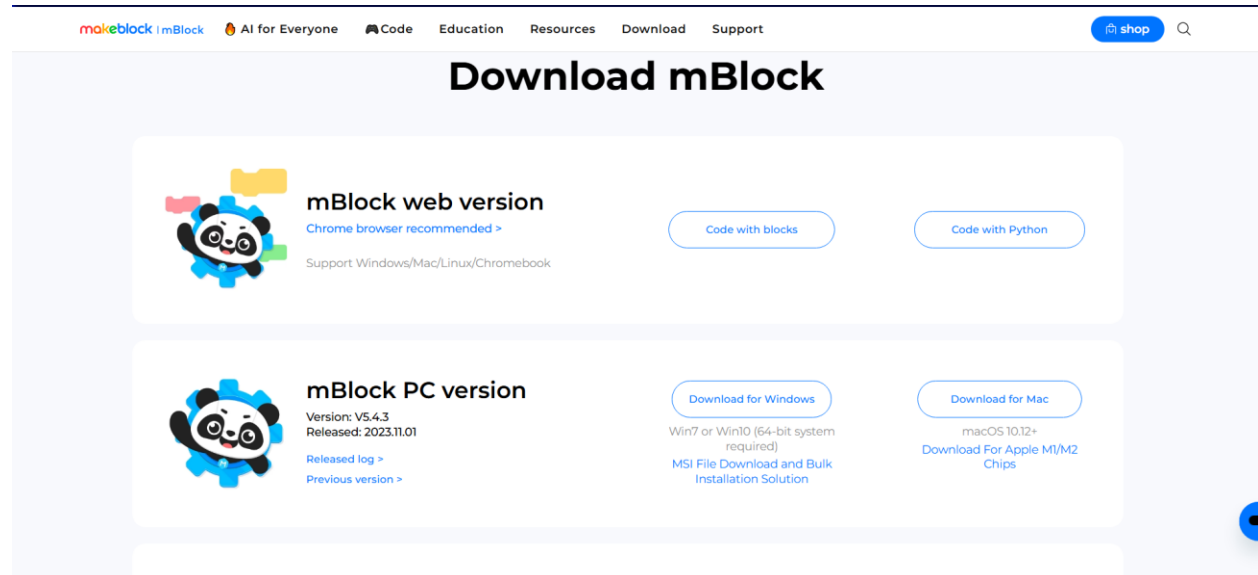


To install the program to use the mBot mega you must go to the website:

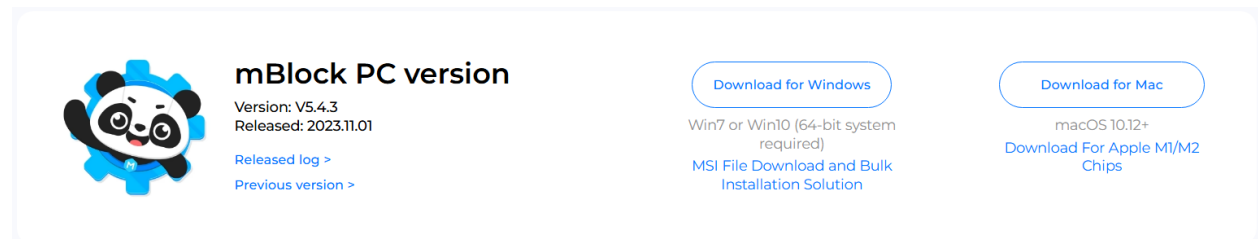
<https://mblock.cc/pages/downloads>

Next, you will scroll down to where it says download mBlock.



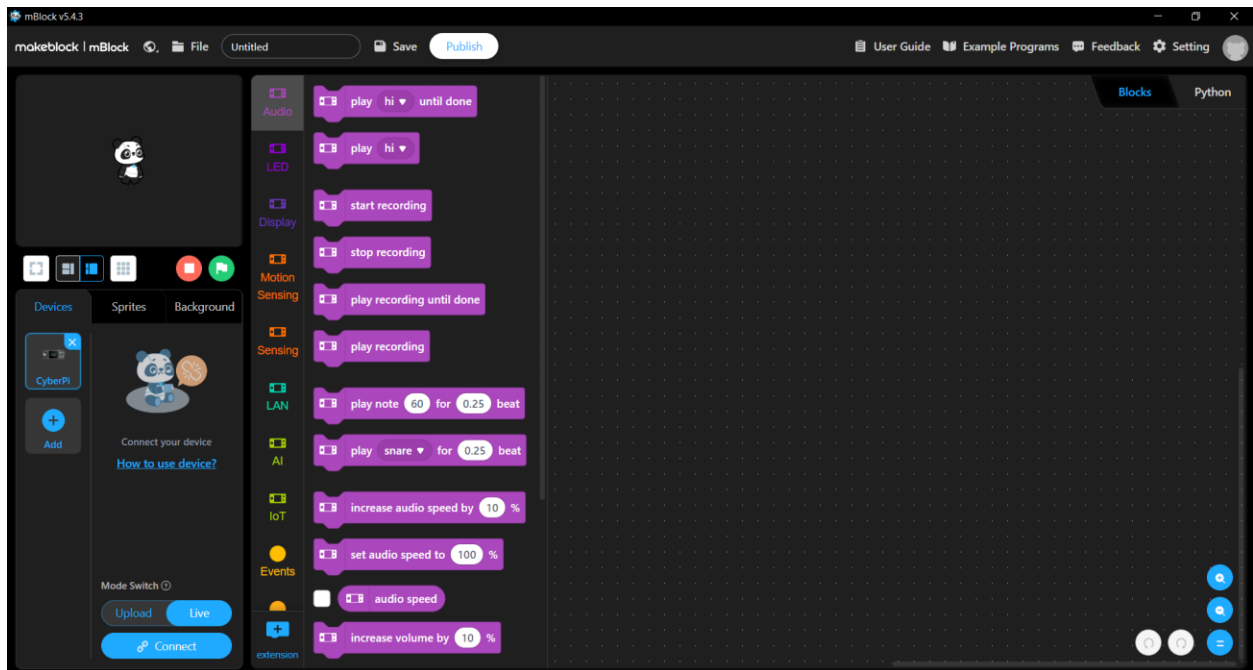
The screenshot shows the 'Download mBlock' page on the mBlock website. The page has a navigation bar at the top with links for 'makeblock | mBlock', 'AI for Everyone', 'Code', 'Education', 'Resources', 'Download', and 'Support'. There is a 'shop' button and a search icon on the right. The main heading is 'Download mBlock'. Below this, there are two main sections. The first section is for 'mBlock web version', which includes a note 'Chrome browser recommended >' and 'Support Windows/Mac/Linux/Chromebook'. It has two buttons: 'Code with blocks' and 'Code with Python'. The second section is for 'mBlock PC version', which includes the version 'V5.4.3' and release date 'Released: 2023.11.01'. It has links for 'Released log >' and 'Previous version >'. There are two buttons: 'Download for Windows' and 'Download for Mac'. The 'Download for Windows' button has a note 'Win7 or Win10 (64-bit system required)' and a link 'MSI File Download and Bulk Installation Solution'. The 'Download for Mac' button has a note 'macOS 10.12+' and a link 'Download For Apple M1/M2 Chips'.

Look at where it says PC version, and download the one that corresponds to the device being used (Either Windows or Mac)

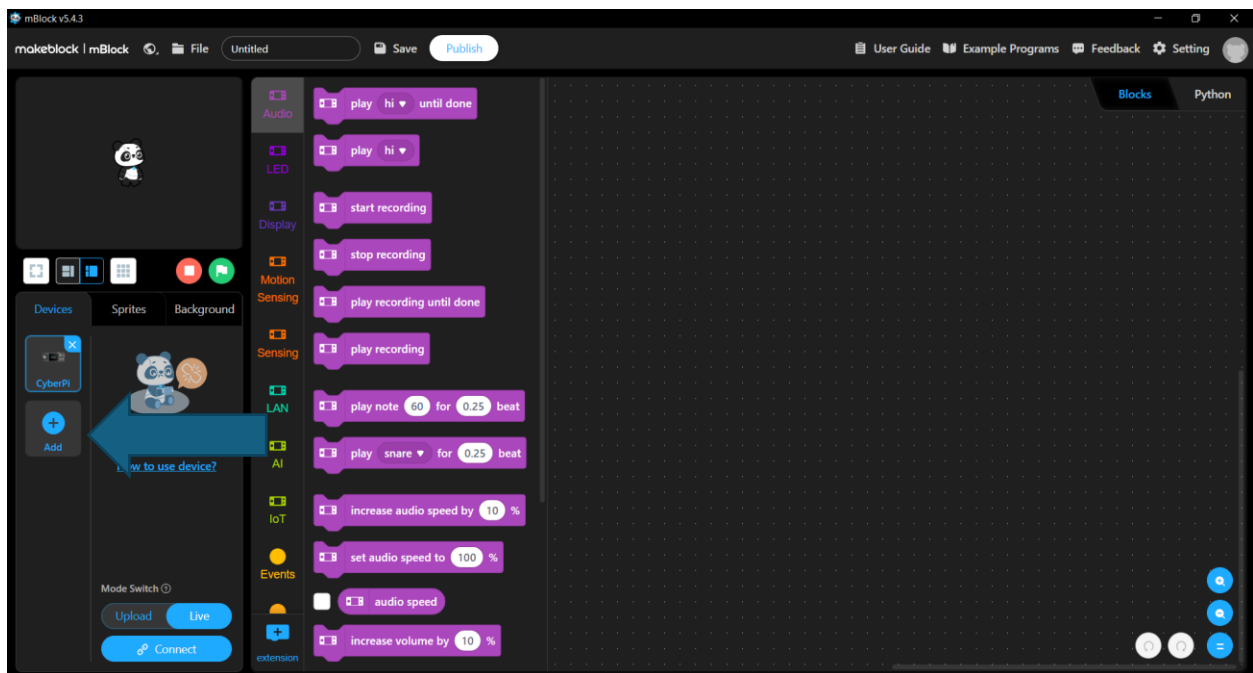


This is a close-up of the 'mBlock PC version' section from the previous screenshot. It features the mBlock logo (a panda head with a gear) on the left. To the right of the logo, it says 'mBlock PC version', 'Version: V5.4.3', and 'Released: 2023.11.01'. Below this are links for 'Released log >' and 'Previous version >'. To the right of the text are two buttons: 'Download for Windows' and 'Download for Mac'. The 'Download for Windows' button has a note 'Win7 or Win10 (64-bit system required)' and a link 'MSI File Download and Bulk Installation Solution'. The 'Download for Mac' button has a note 'macOS 10.12+' and a link 'Download For Apple M1/M2 Chips'.

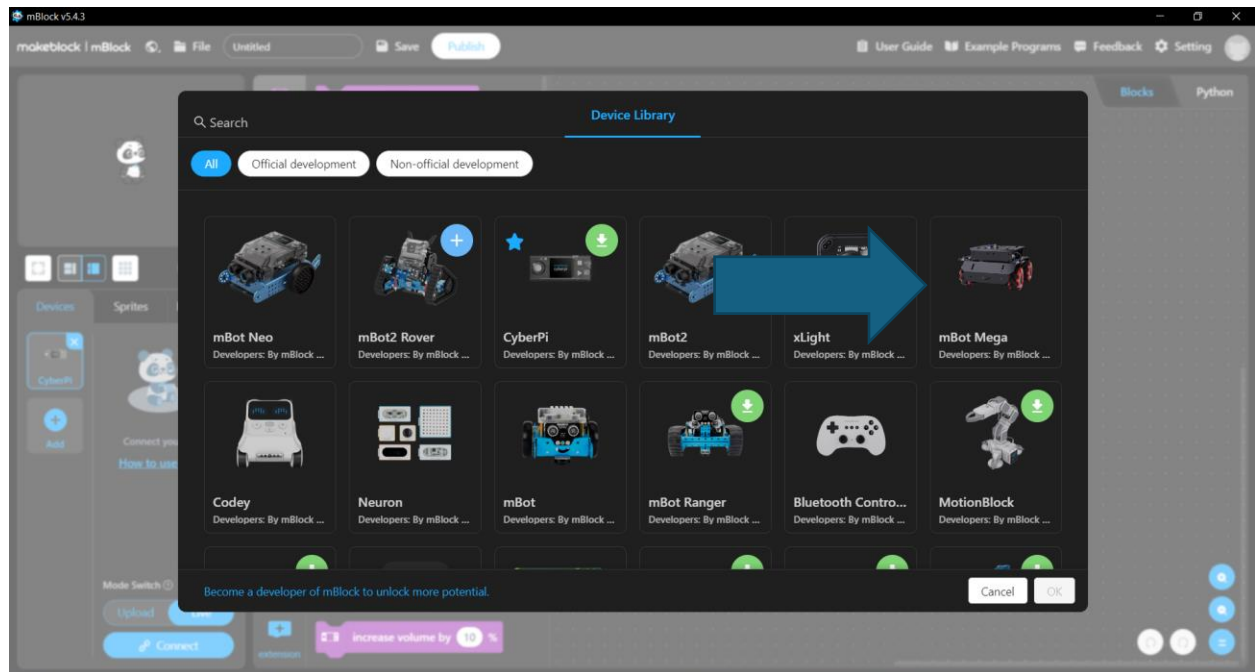
After downloading, your IDE should look like this when opened (yours may be white instead of black when first opened):



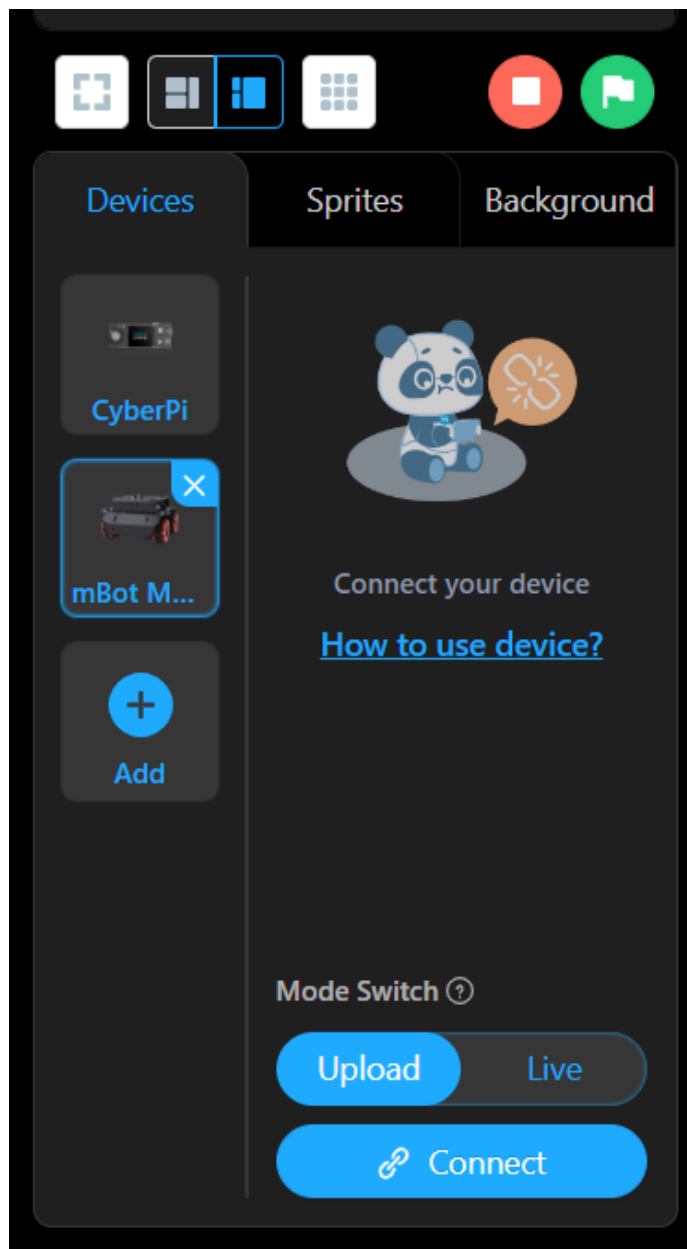
To add the mBot mega click the add button on the left side of the screen



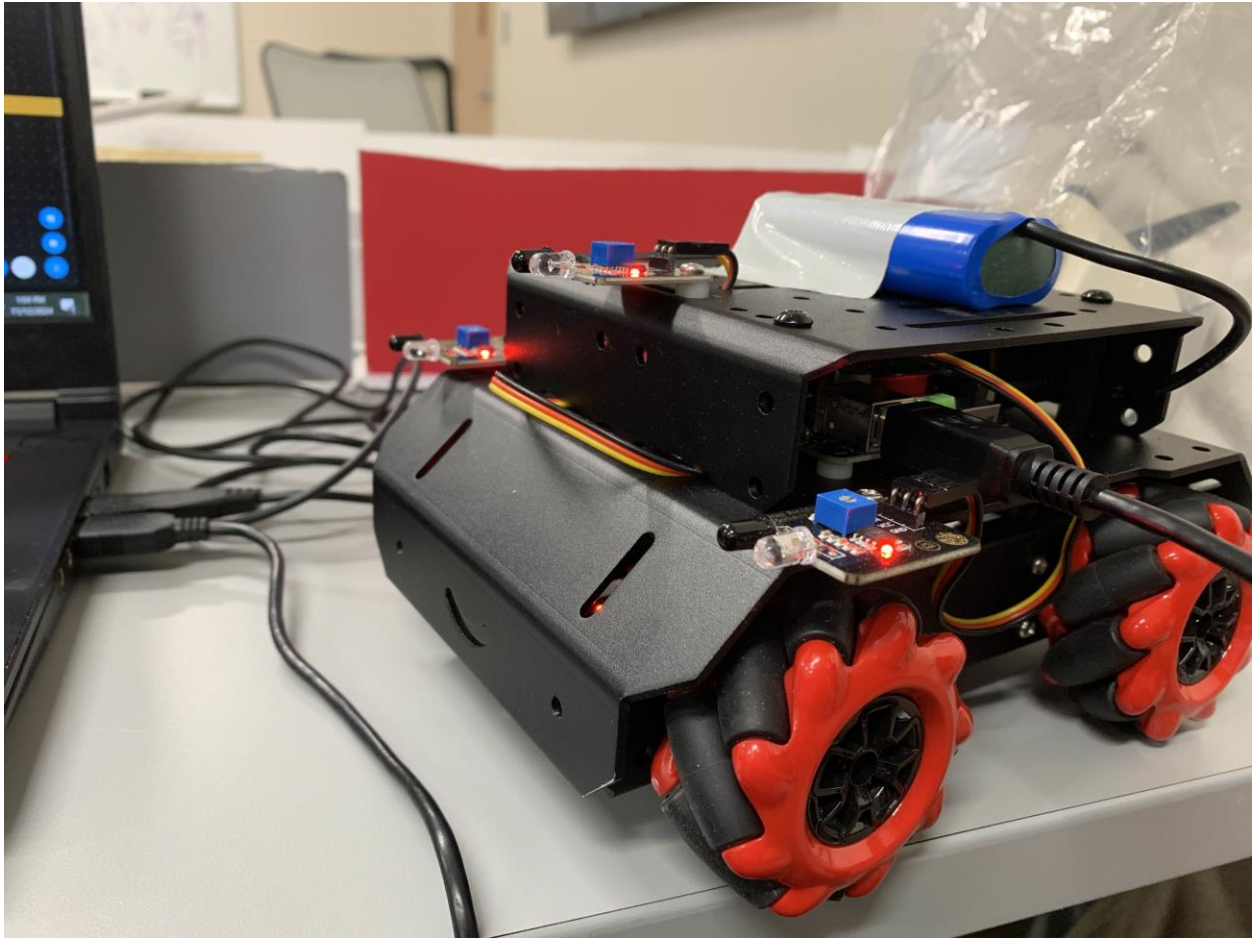
Next, click the mBot mega to add it to your devices, and click OK after clicking mBot mega



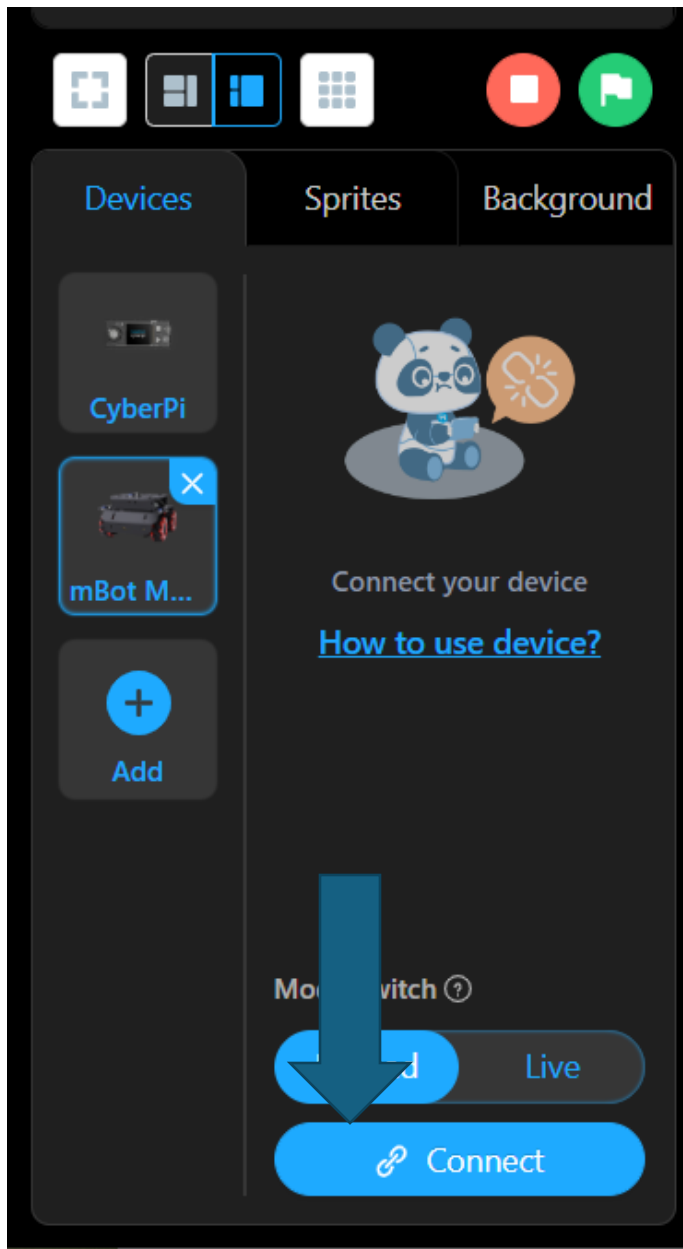
Make sure that the mode is switched to upload and not live, the blue should be highlighted for Upload



Once your mBot is connected like this,



Make sure the wire is connected to the raspberry pi on the bot and the USB is connected to your device, click the connect button and connect your bot.



We have three parts to our demo:

- I. Coding the bot to move back and forth
- II. Coding the bot to detect an obstacle
- III. The crash detection feature

1. Coding the bot to move back and forth

Looks

Motion


Sensing

Events

Control

Operators

when mBot Mega (MegaPi) starts up

when  clicked

when key pressed

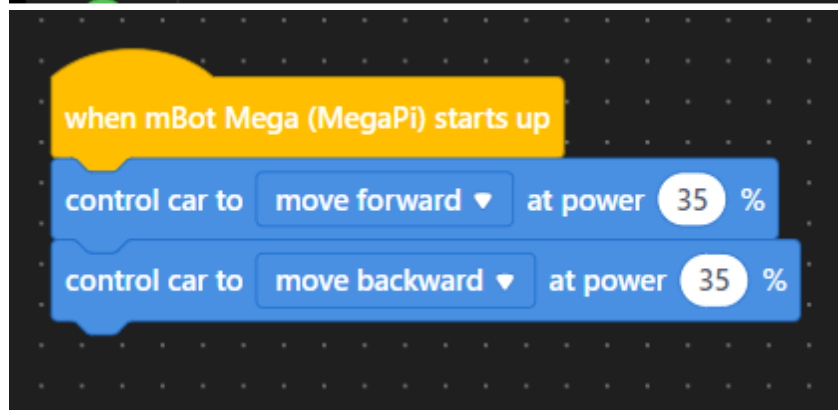
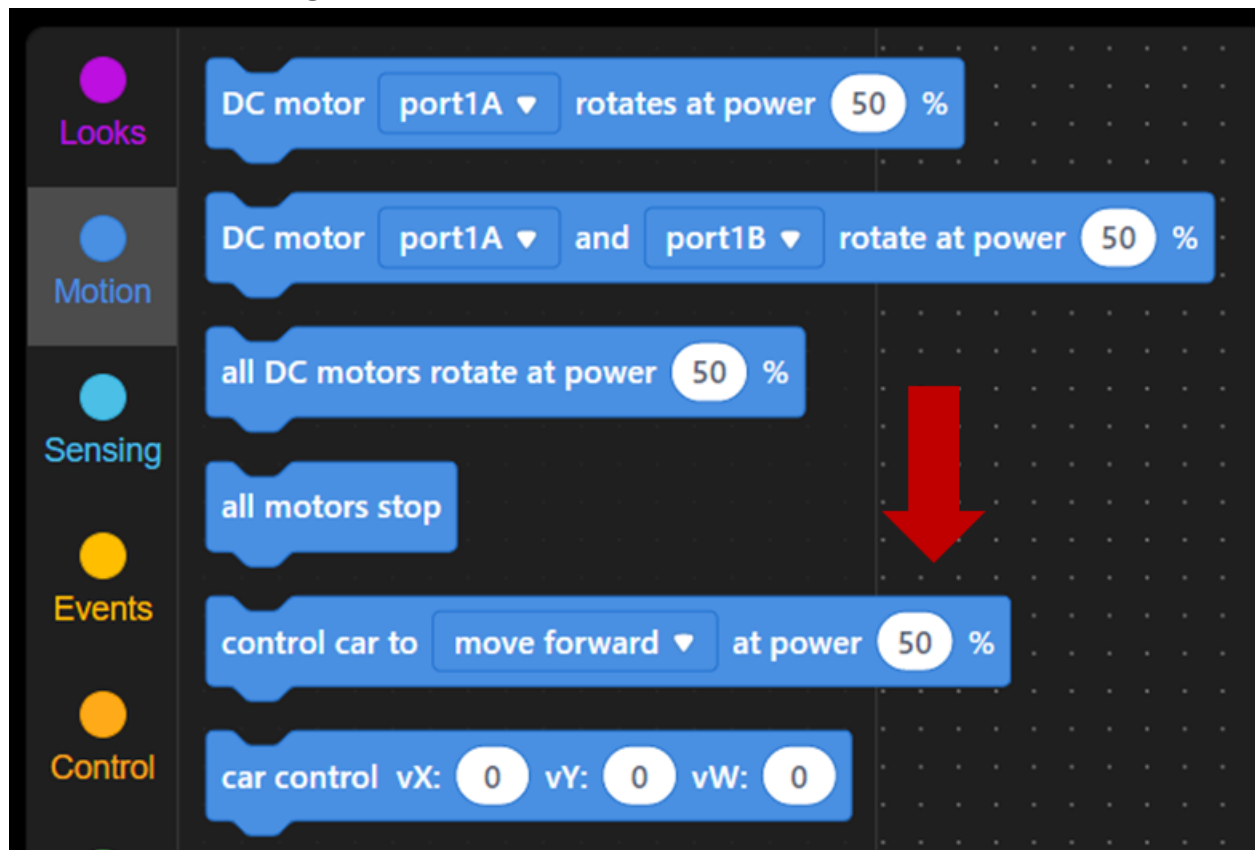
when I receive

broadcast

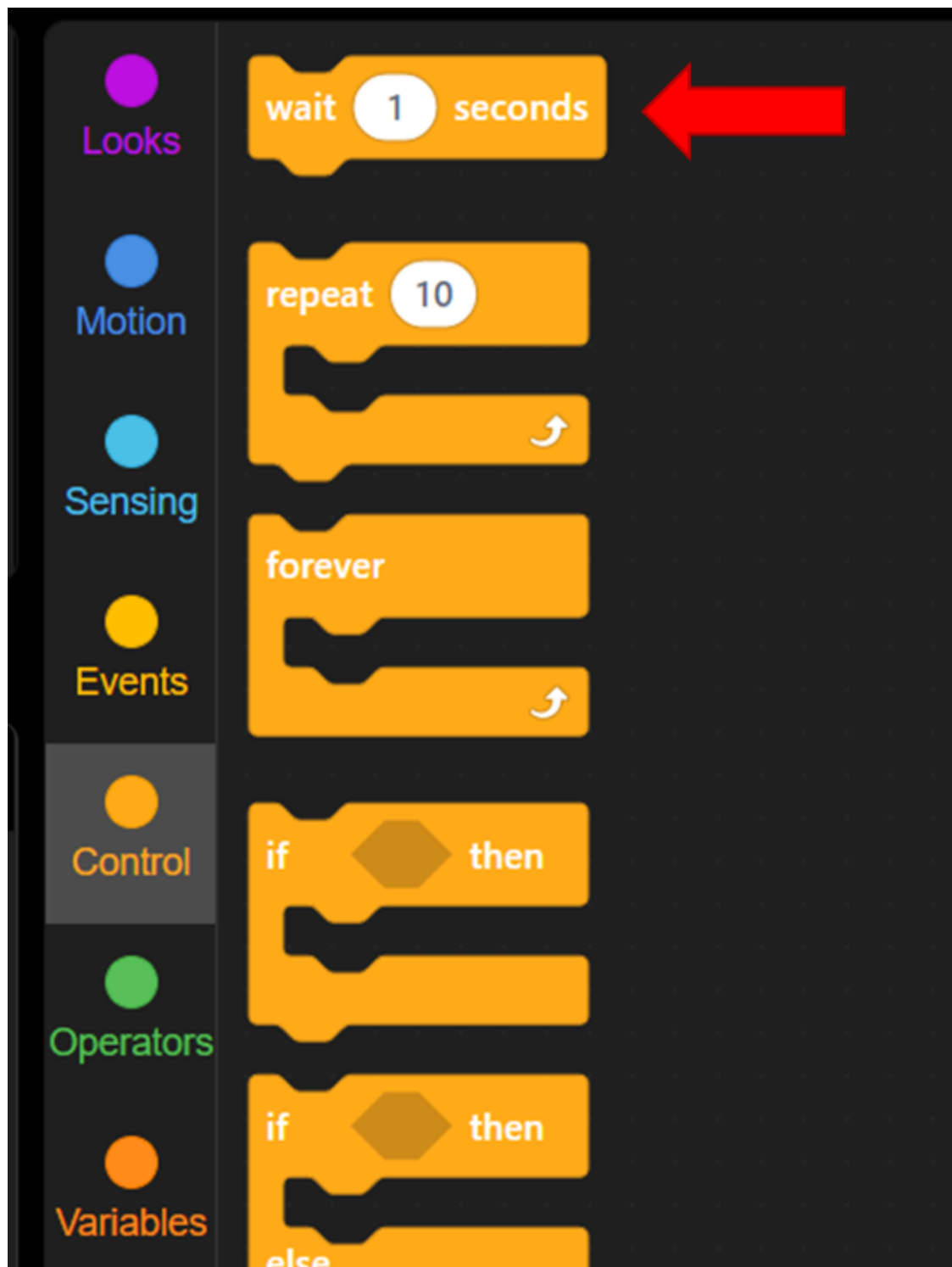
broadcast and wait

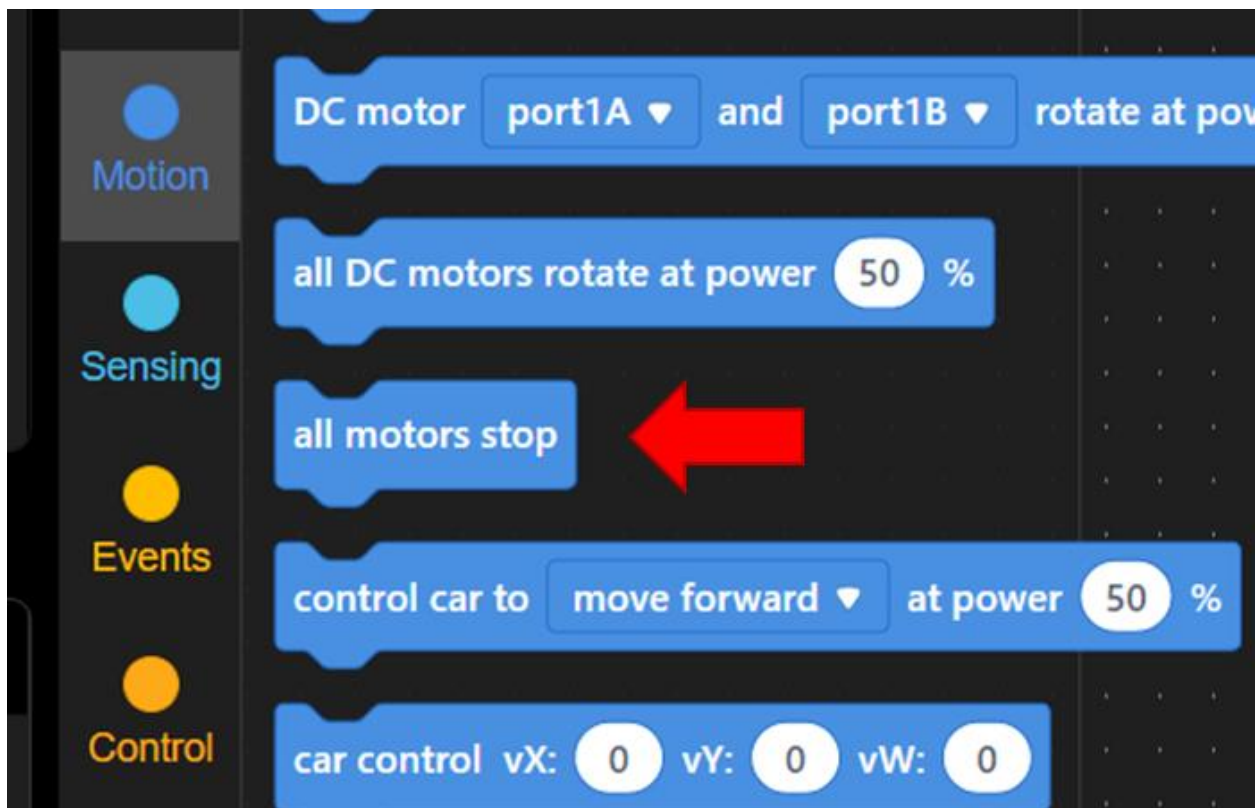
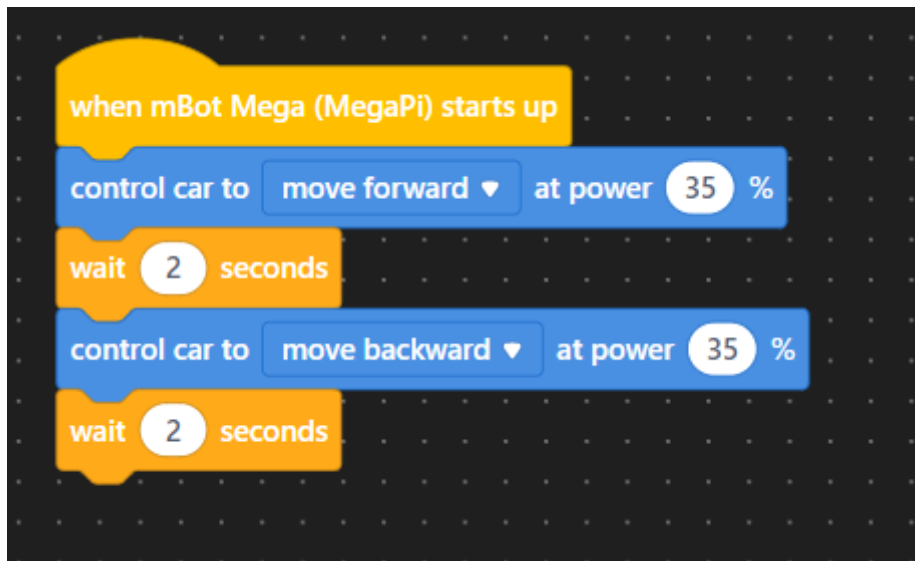


Make sure to grab two of the move blocks



Make sure to grab two wait second blocks





2. Coding the bot to detect an obstacle

Looks

when mBot Mega (MegaPi) starts up

Motion

when  clicked



Sensing

when key pressed

Events

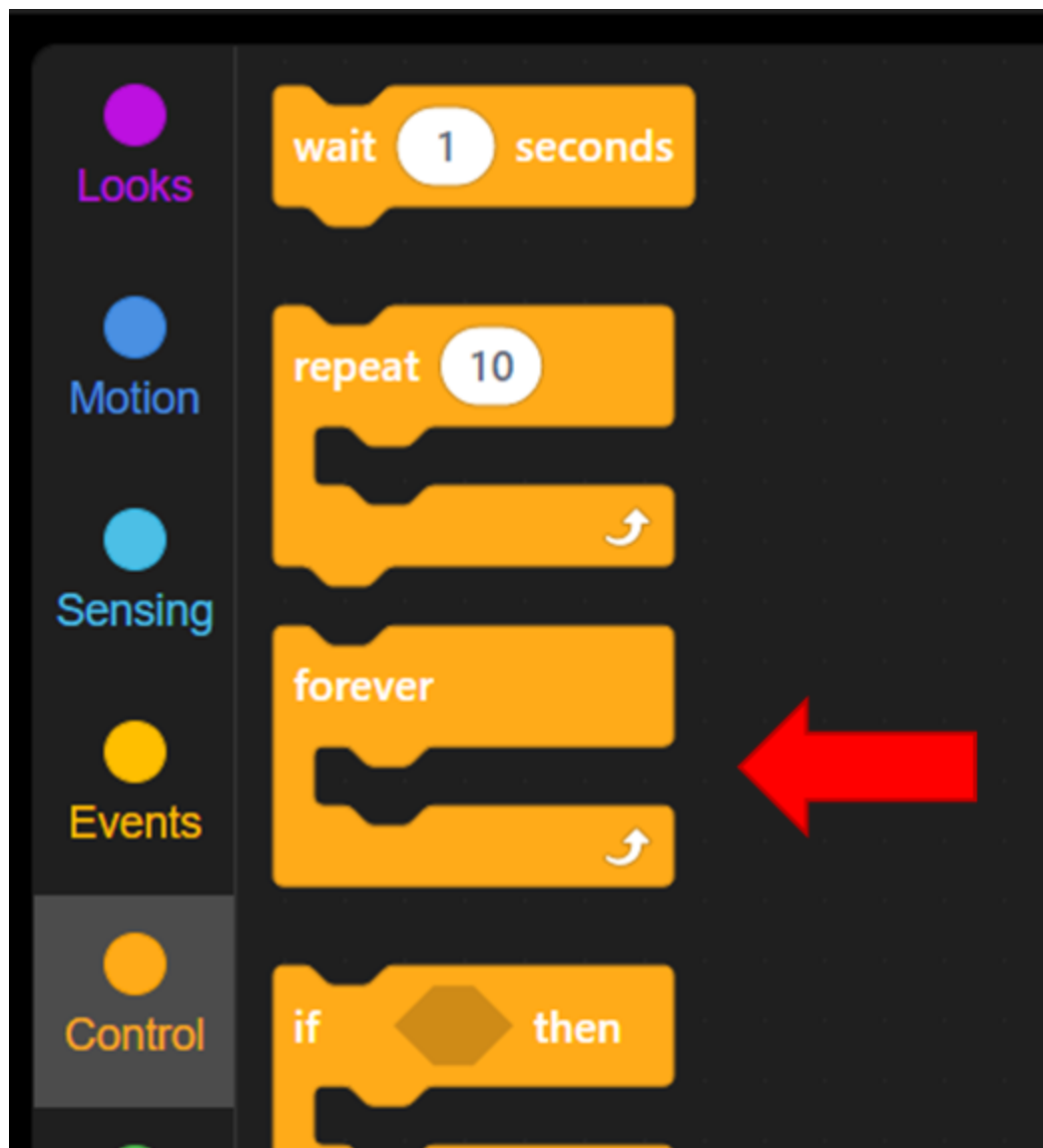
when I receive

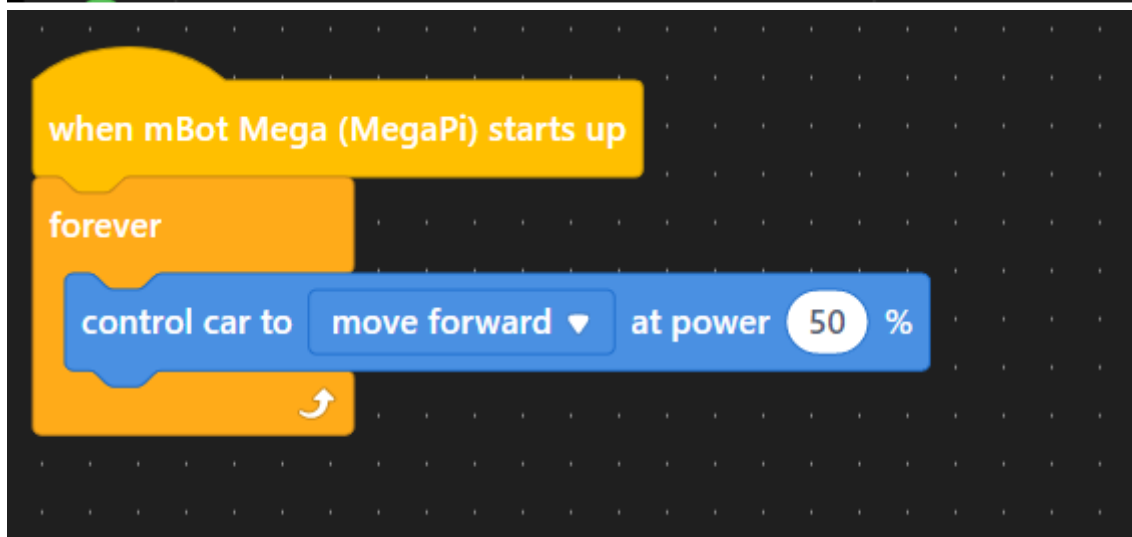
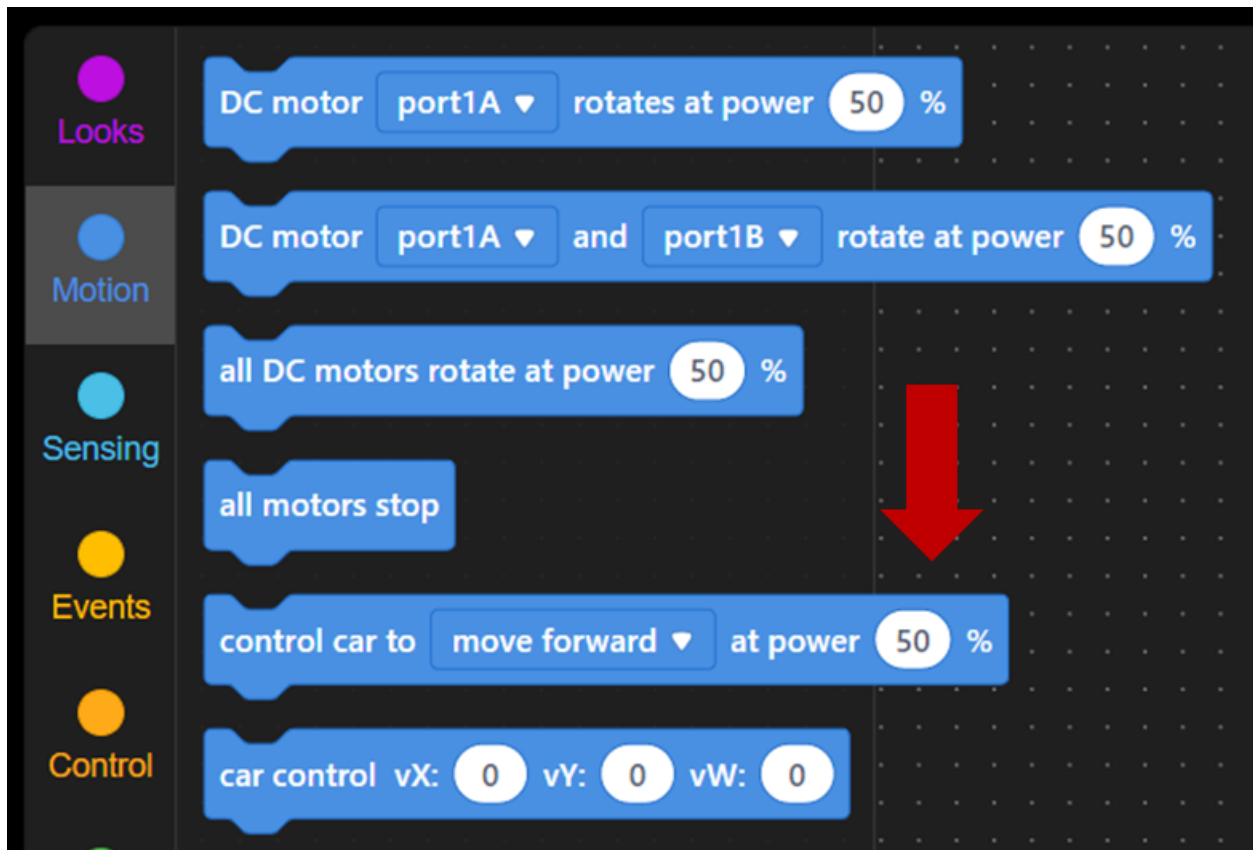
Control

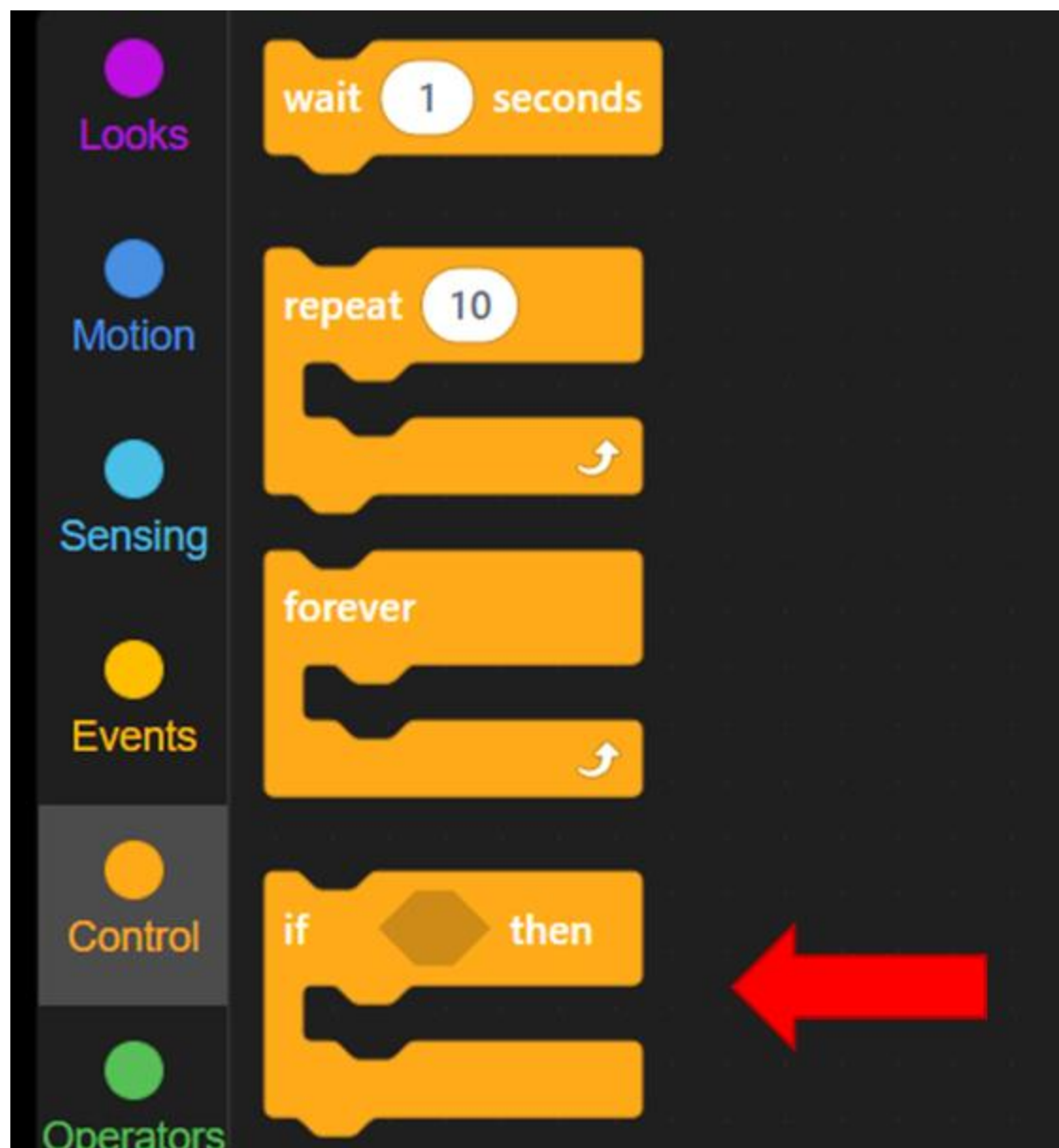
broadcast

Operators

broadcast and wait

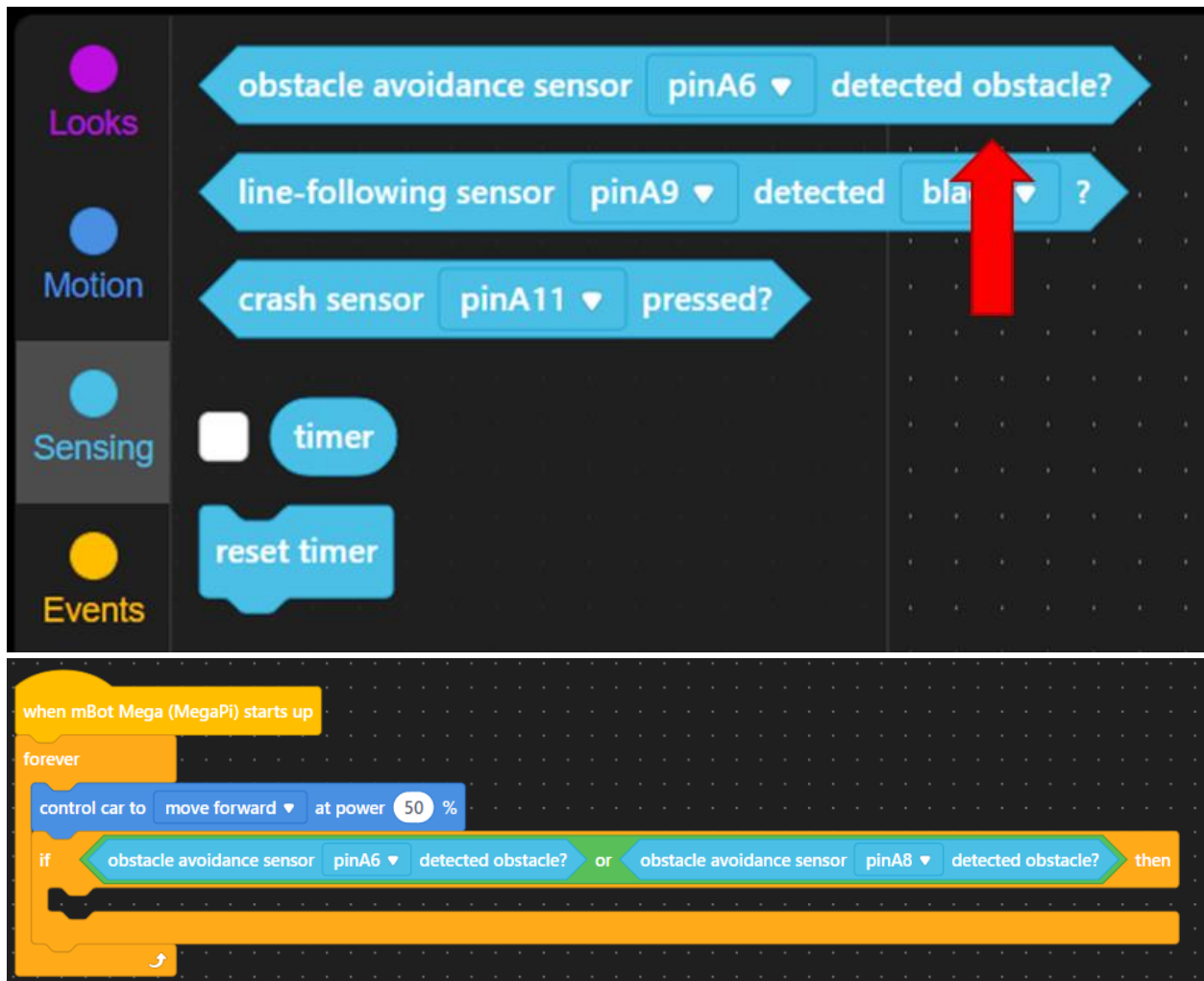








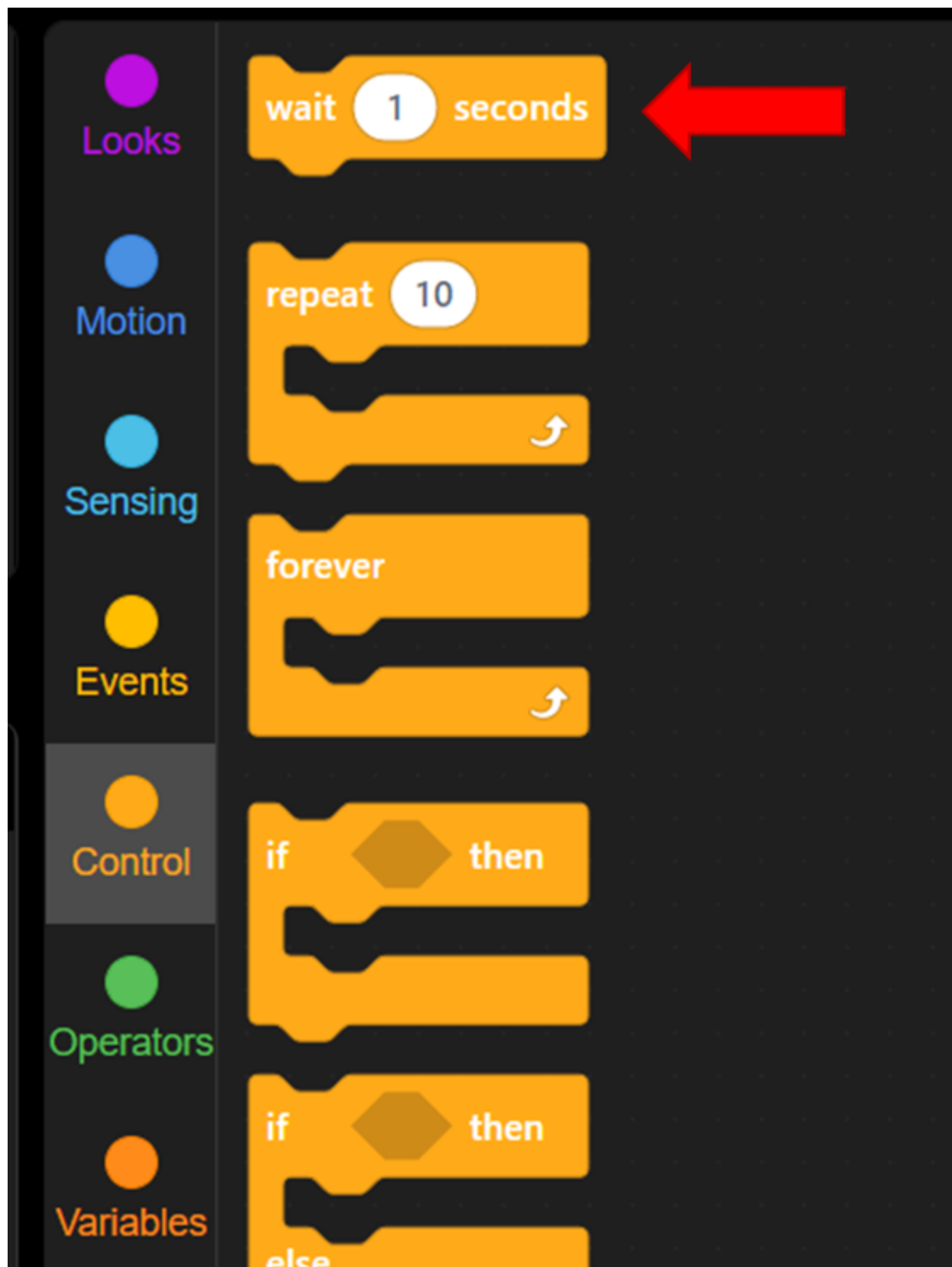
Grab two of the sensors block, and set them both to pin A6 and pin A8



The image shows a Scratch script for controlling a car. The script is organized into categories on the left: Looks, Motion, Sensing, Events, and Control. The script consists of the following blocks:

- Looks**: A blue block that says "DC motor port1A ▼ rotates at power 50 %".
- Motion**: A blue block that says "DC motor port1A ▼ and port1B ▼ rotate at power 50 %".
- Motion**: A blue block that says "all DC motors rotate at power 50 %".
- Motion**: A blue block that says "all motors stop". A red arrow points to this block from the right.
- Events**: A blue block that says "control car to move forward ▼ at power 50 %".
- Control**: A blue block that says "car control vX: 0 vY: 0 vW: 0".

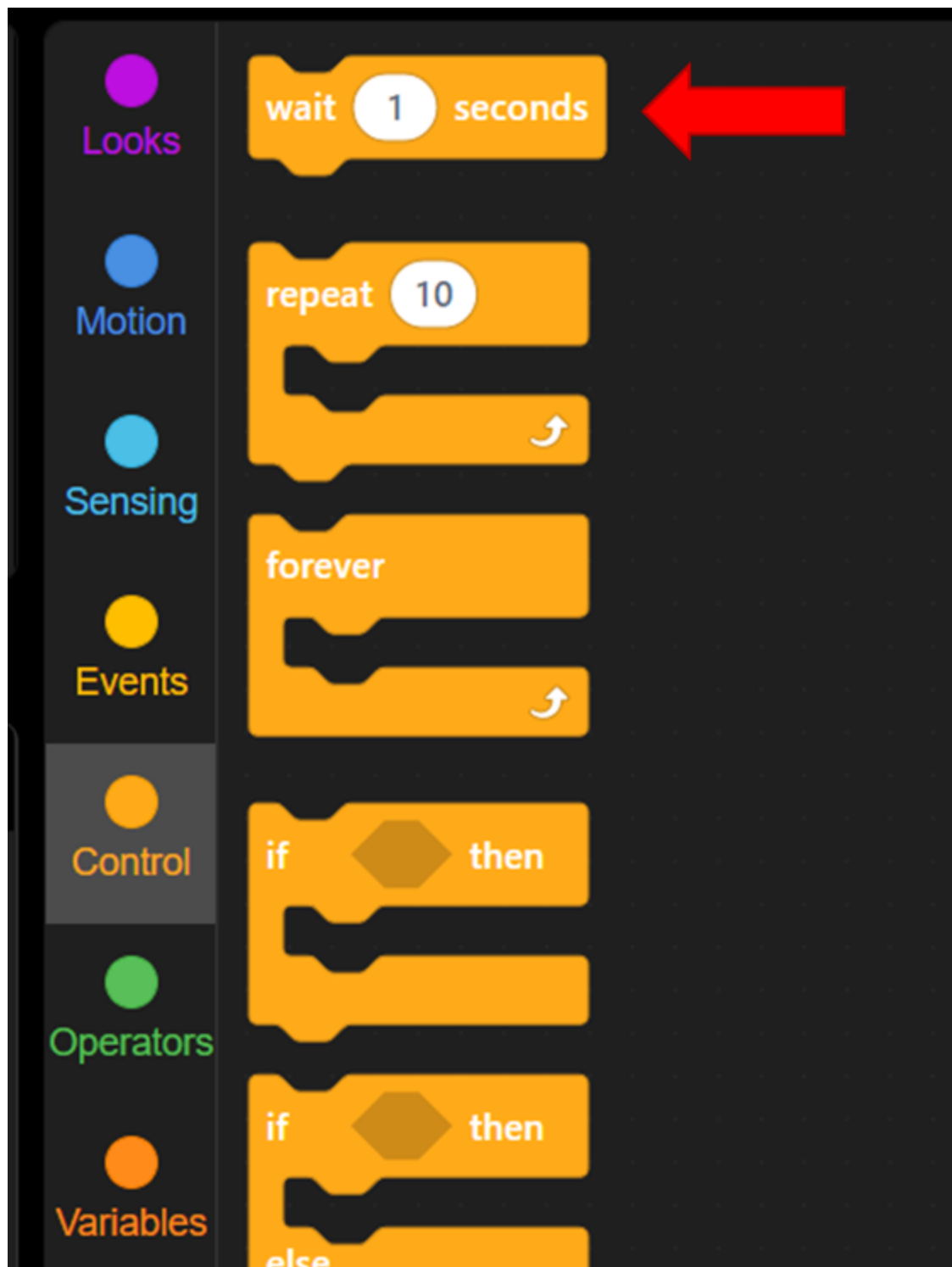
The script is set against a dark gray background with a light gray grid.

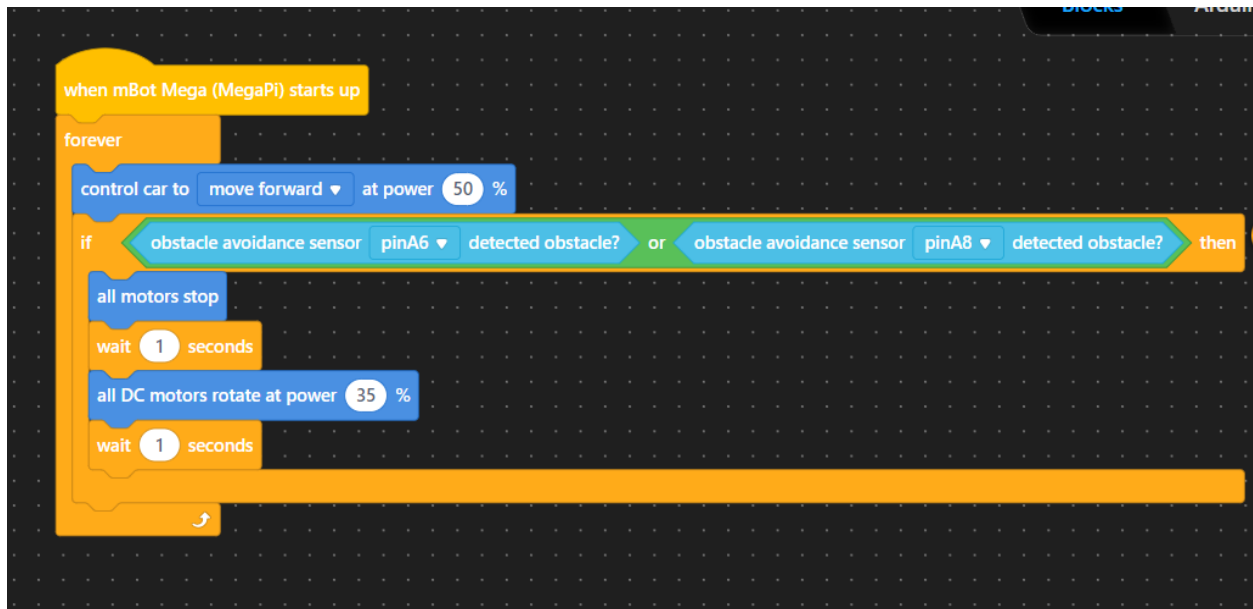


The image shows a Scratch script for controlling a car. The script is organized into categories on the left: Looks, Motion, Sensing, Events, and Control. The script consists of the following blocks:

- Looks**: A blue block that says "DC motor port1A ▾ rotates at power 50 %".
- Motion**: A blue block that says "DC motor port1A ▾ and port1B ▾ rotate at power 50 %".
- Motion**: A blue block that says "all DC motors rotate at power 50 %".
- Motion**: A blue block that says "all motors stop".
- Events**: A blue block that says "control car to move forward ▾ at power 50 %".
- Control**: A blue block that says "car control vX: 0 vY: 0 vW: 0".

A red arrow points from the "all motors stop" block to the "all DC motors rotate at power 50 %" block.





3. The Crash Detection Code

Looks

Motion


Sensing

Events

Control

Operators

when mBot Mega (MegaPi) starts up

when  clicked

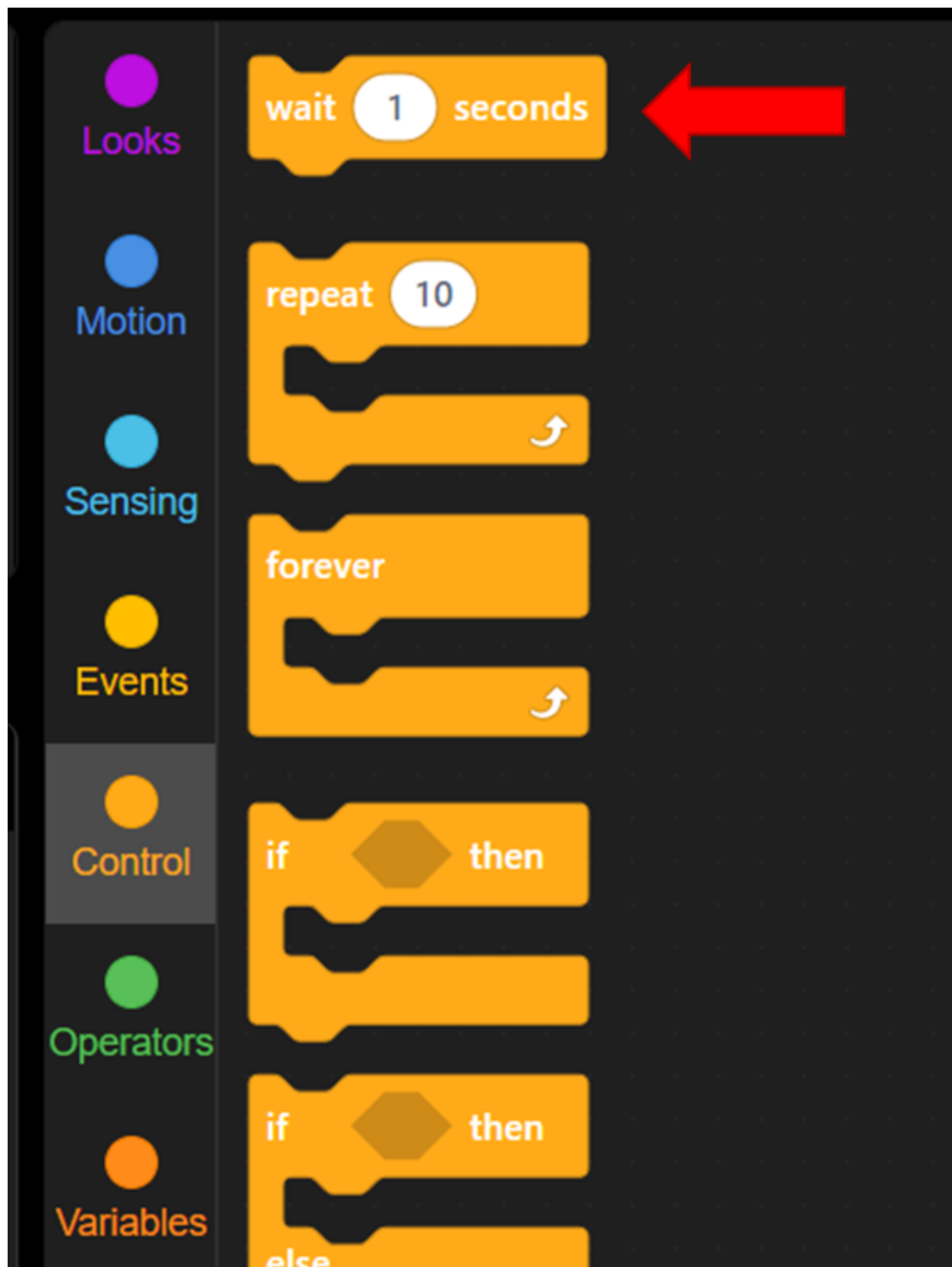
when key pressed

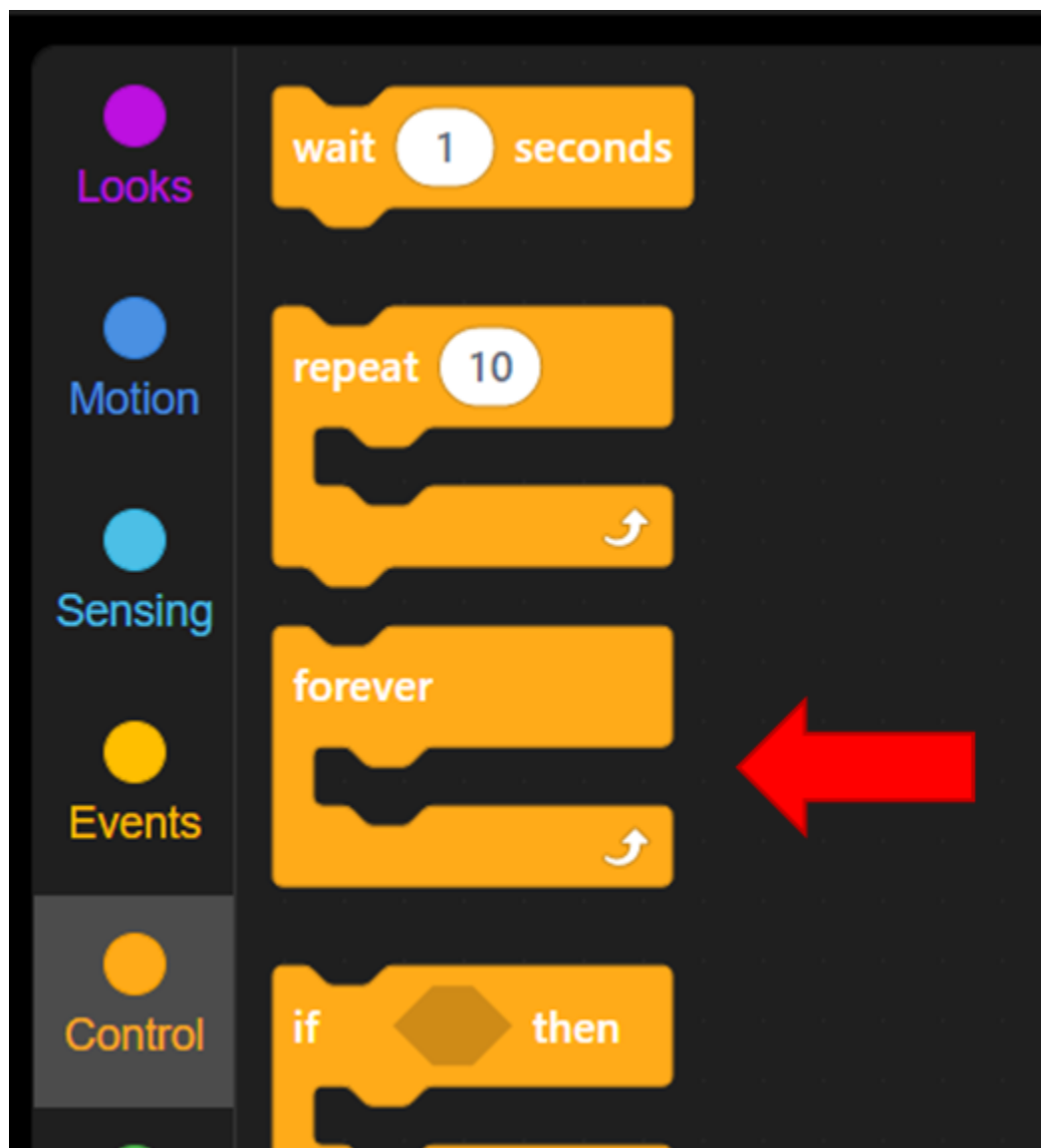
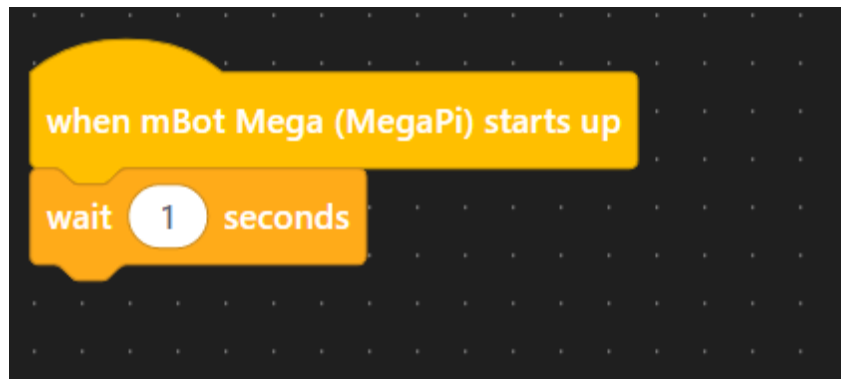
when I receive

broadcast

broadcast and wait



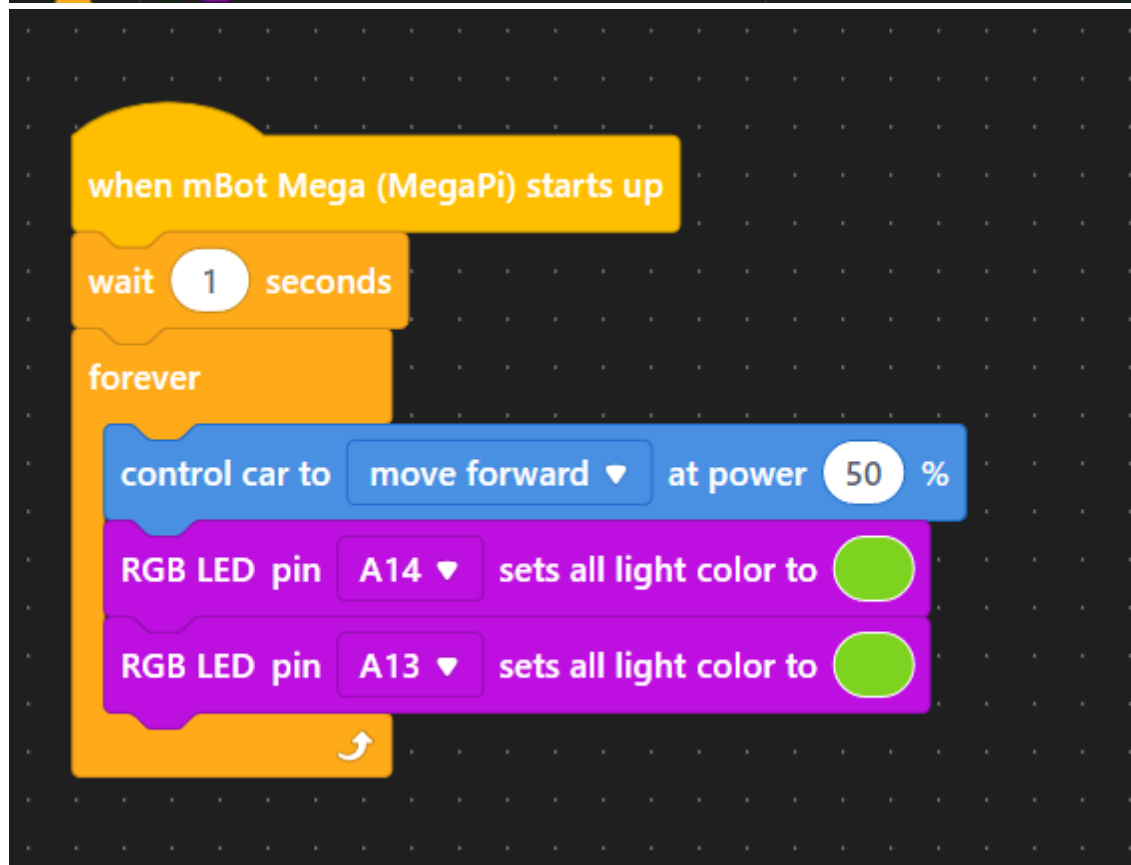
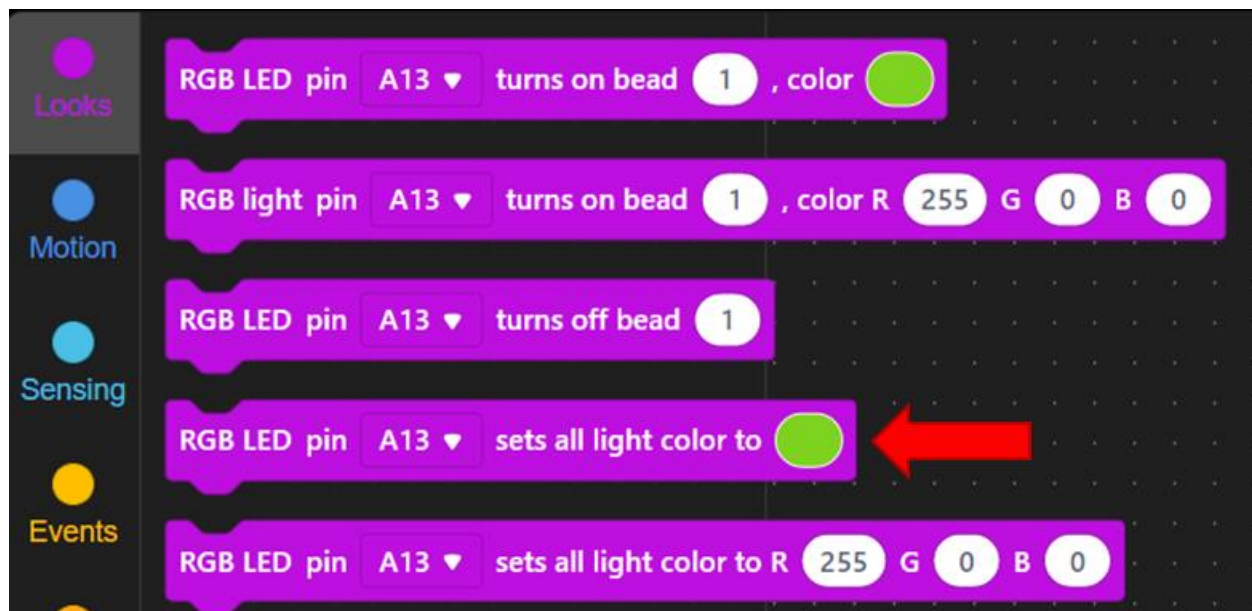


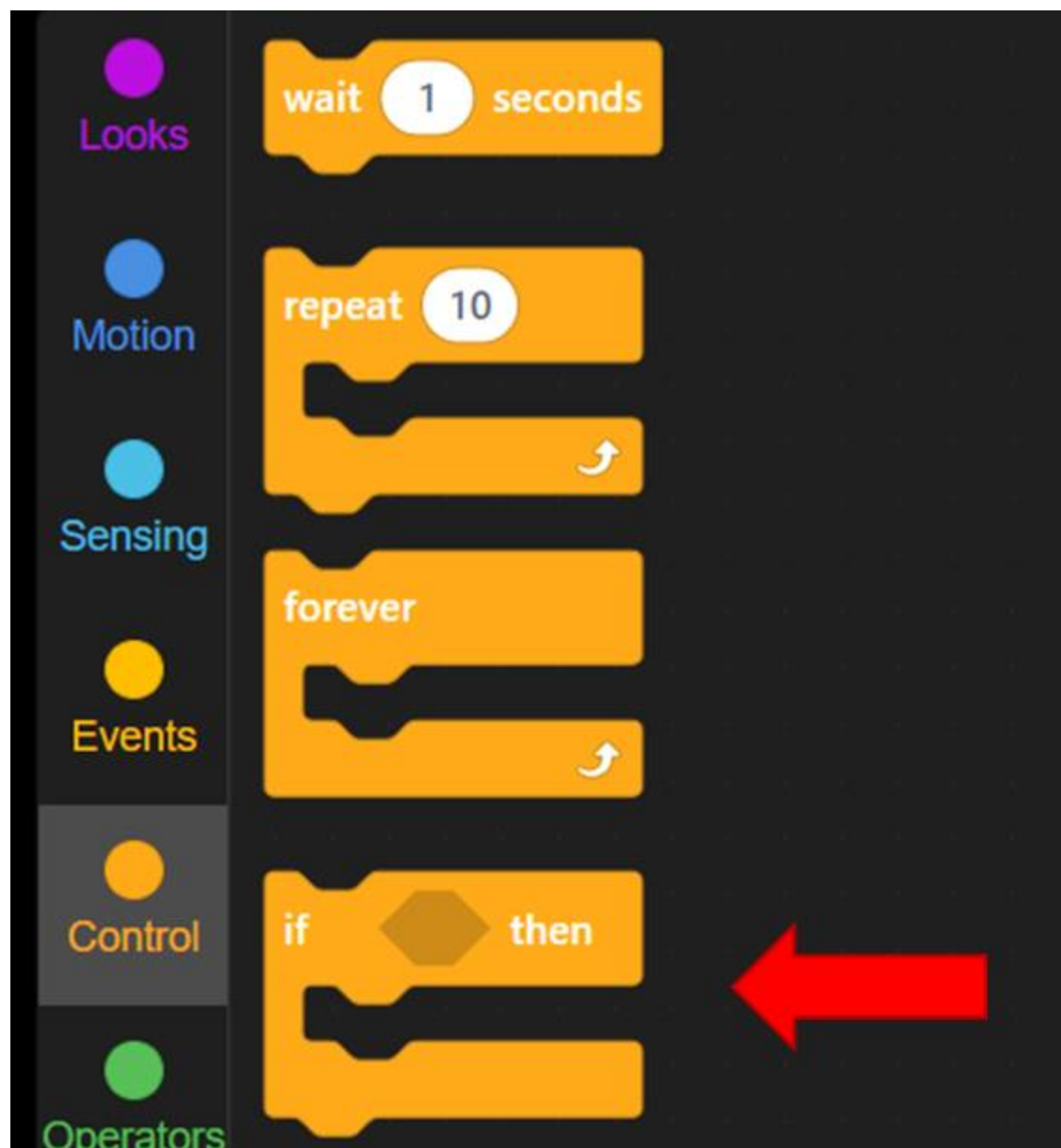


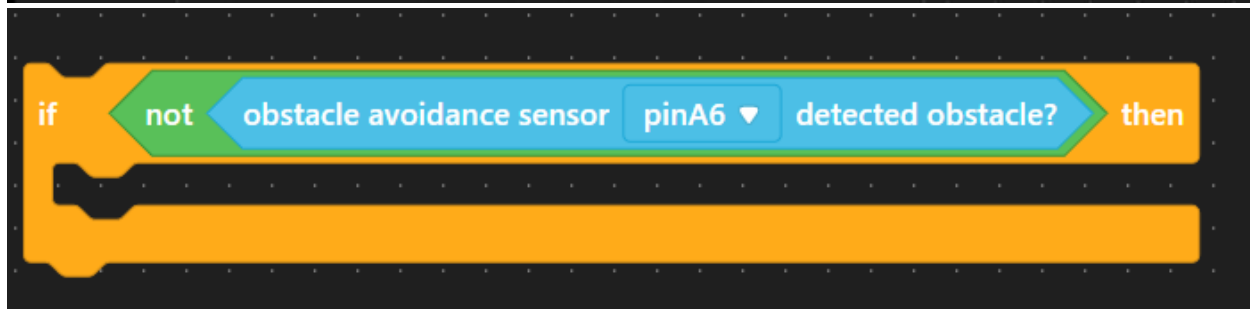
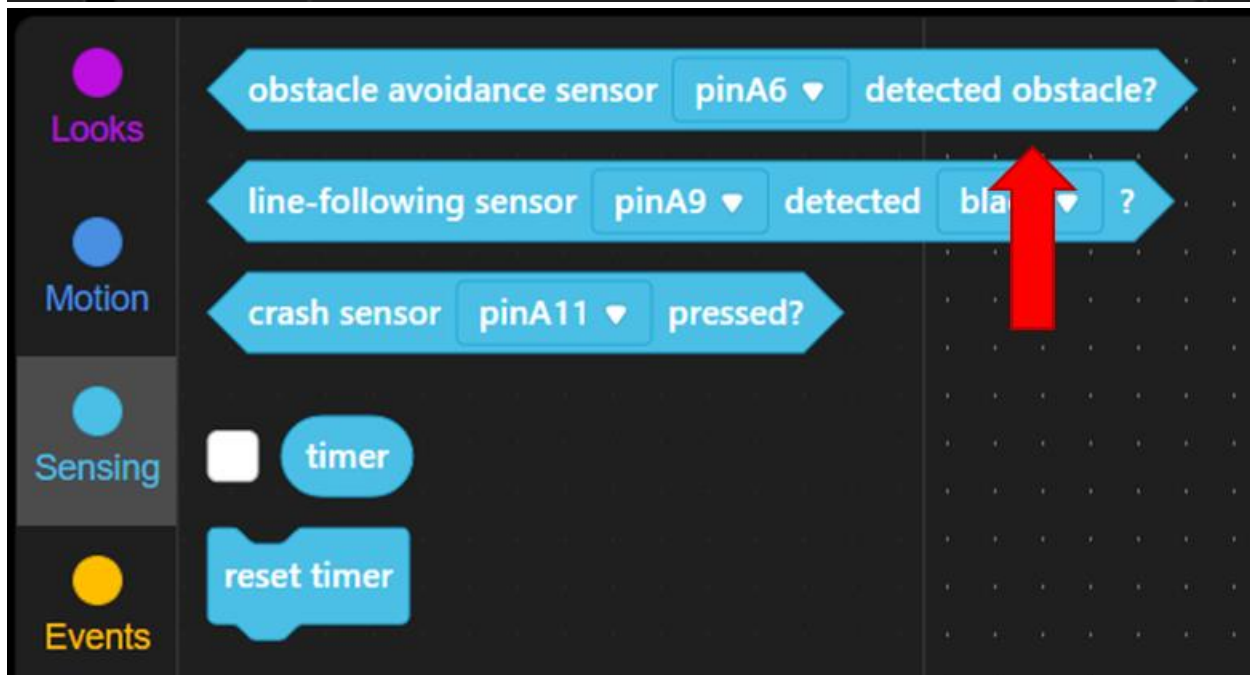
The image shows a Scratch script for controlling a car. The script is organized into five categories on the left: Looks, Motion, Sensing, Events, and Control. The script consists of the following blocks:

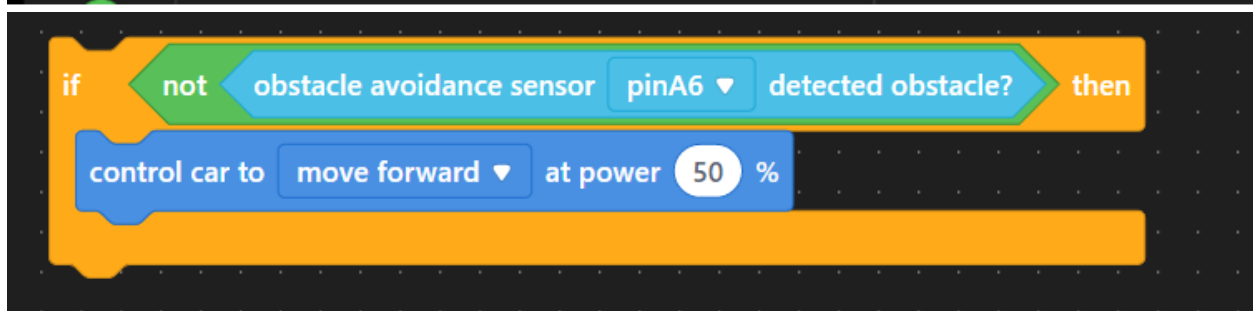
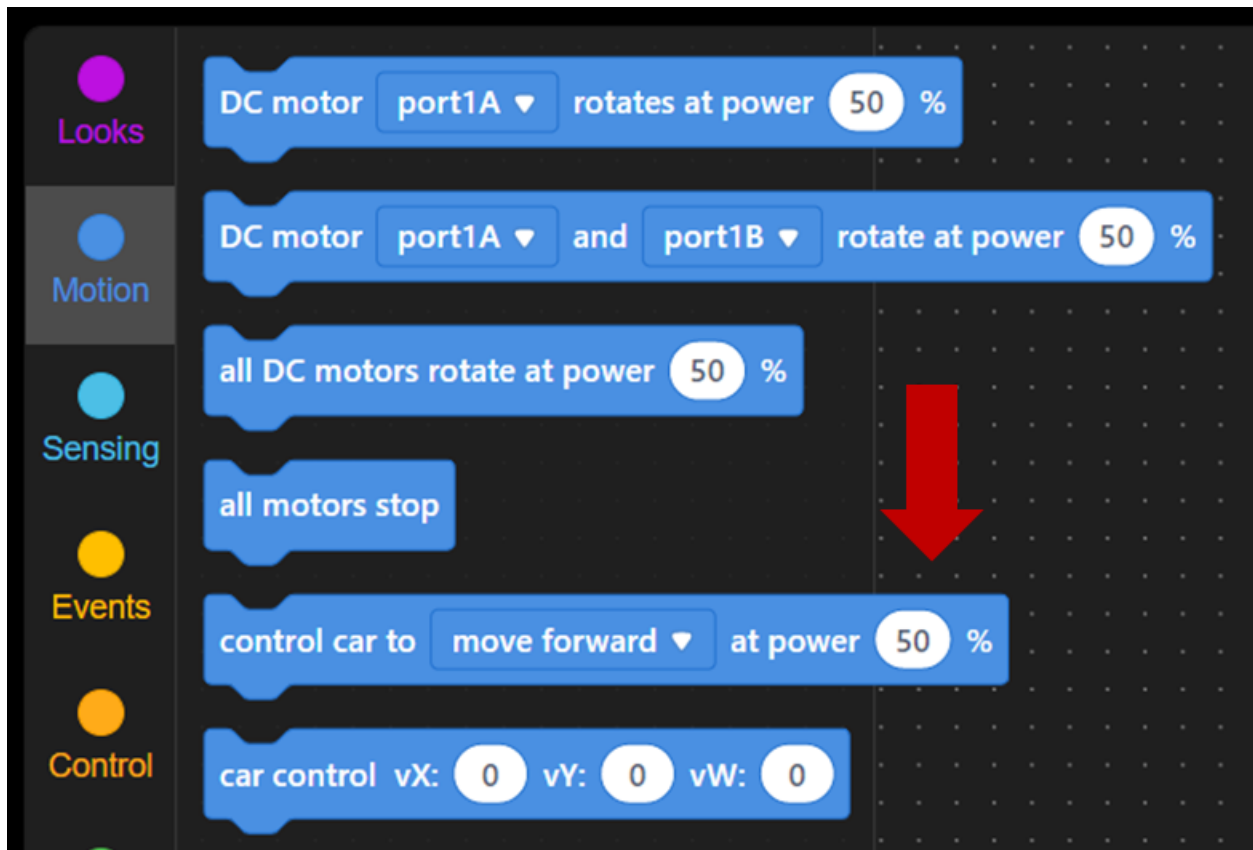
- Looks:** A blue block that says "DC motor" followed by a dropdown menu showing "port1A", then "rotates at power" followed by a numeric input field containing "50" and a percentage sign.
- Motion:** A blue block that says "DC motor" followed by a dropdown menu showing "port1A", then "and" followed by a dropdown menu showing "port1B", then "rotate at power" followed by a numeric input field containing "50" and a percentage sign.
- Motion:** A blue block that says "all DC motors rotate at power" followed by a numeric input field containing "50" and a percentage sign.
- Motion:** A blue block that says "all motors stop".
- Events:** A blue block that says "control car to" followed by a dropdown menu showing "move forward", then "at power" followed by a numeric input field containing "50" and a percentage sign.
- Control:** A blue block that says "car control" followed by "vX:" followed by a numeric input field containing "0", "vY:" followed by a numeric input field containing "0", and "vW:" followed by a numeric input field containing "0".

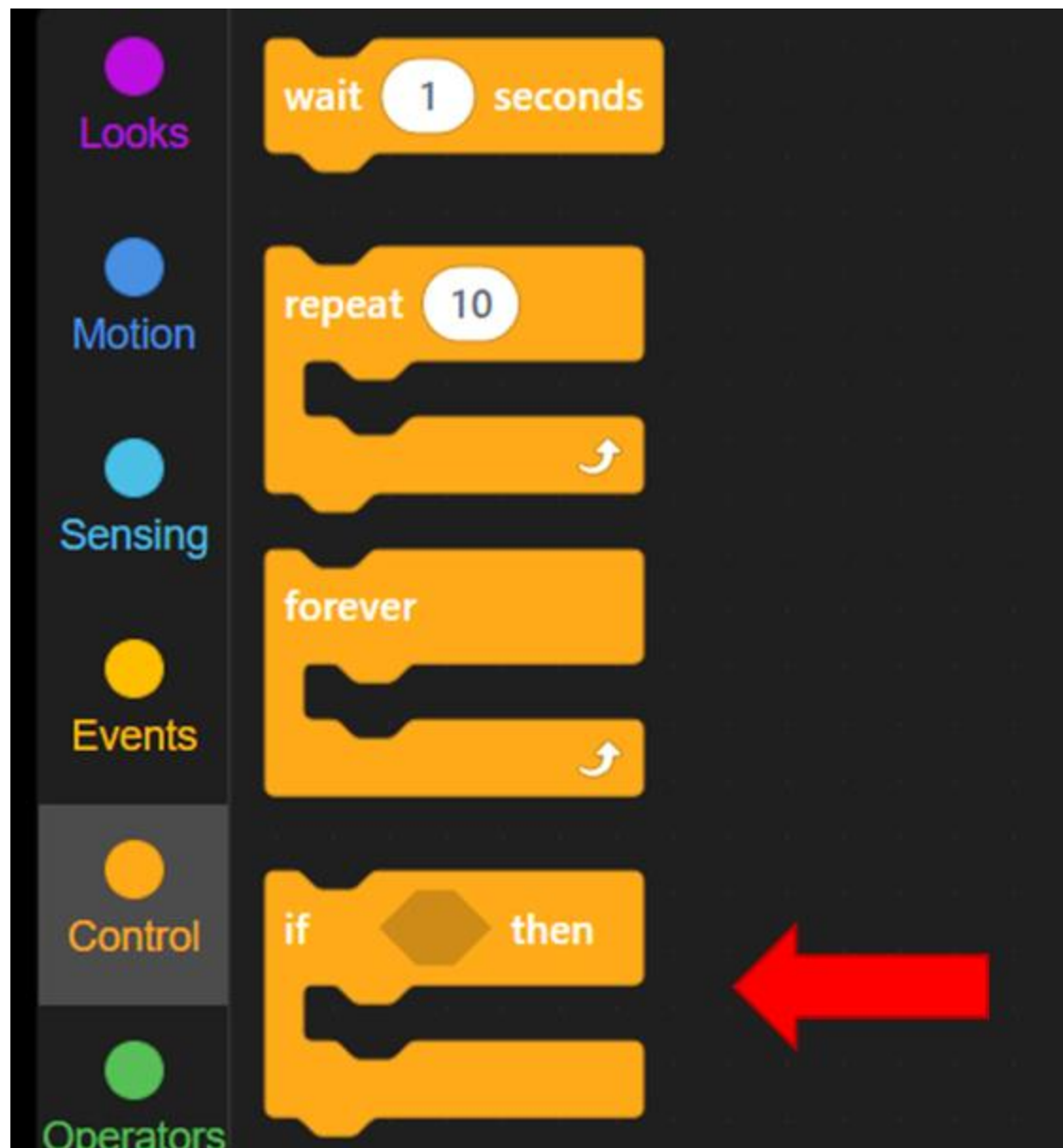
A large red arrow points downwards from the "all motors stop" block to the "control car to" block.

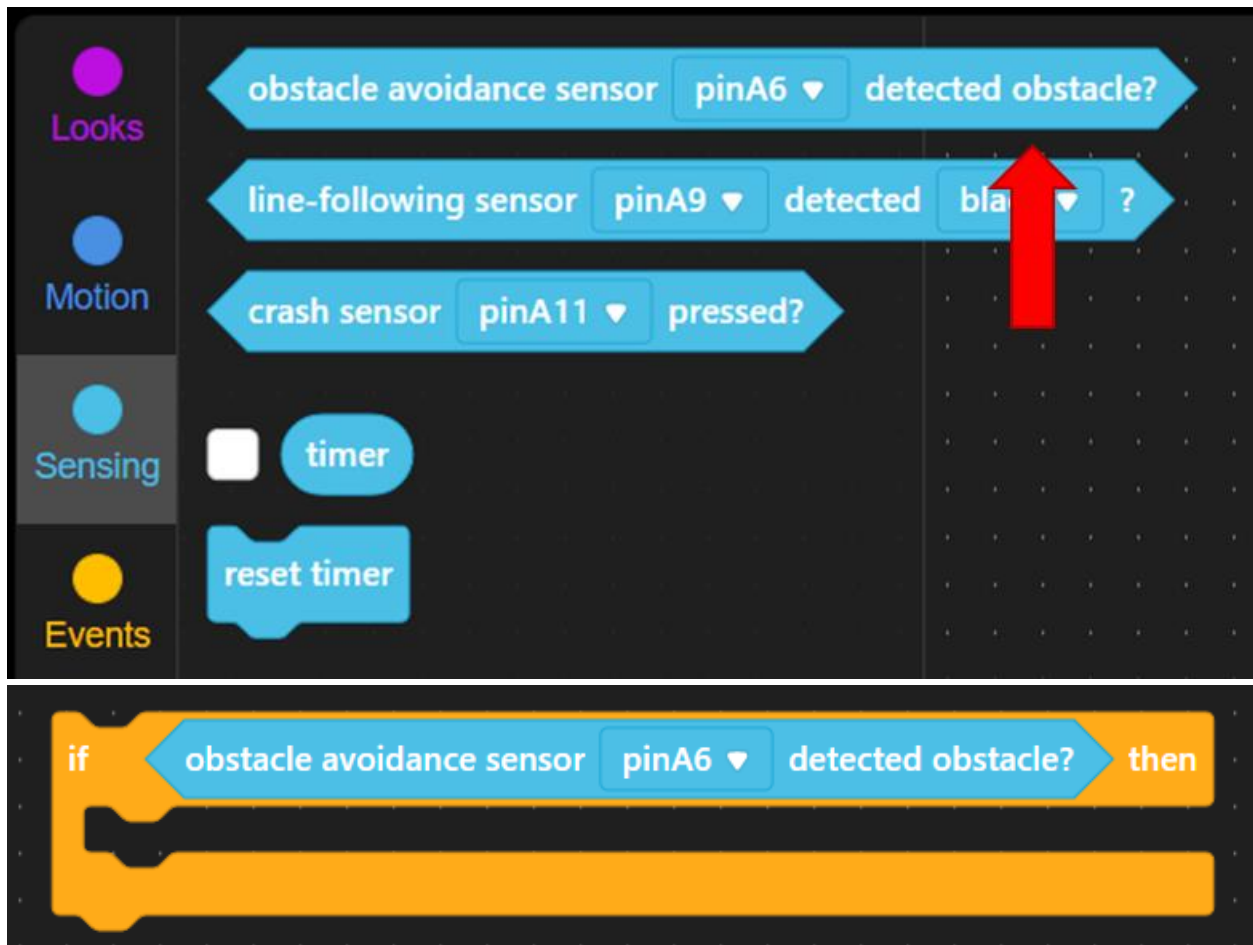


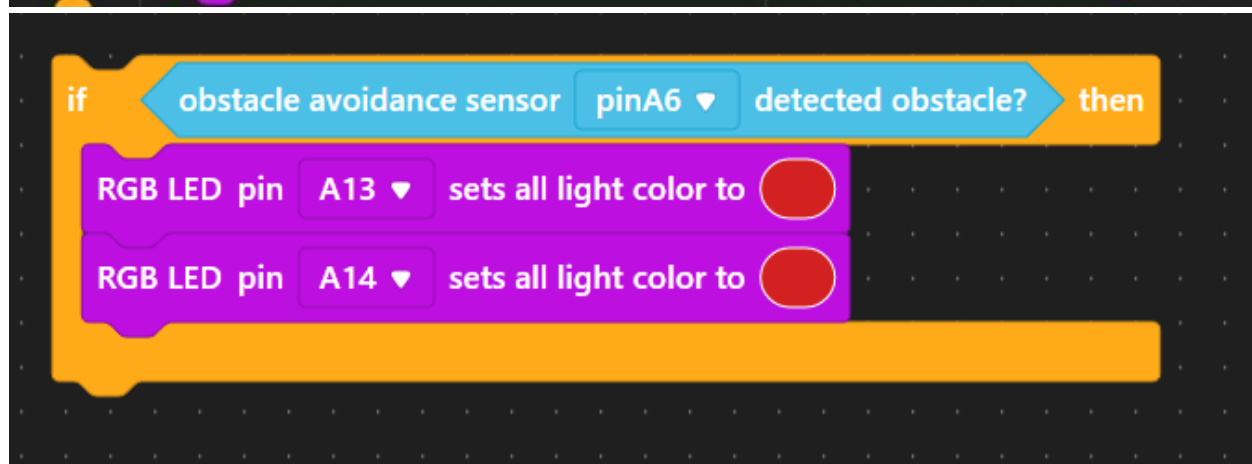
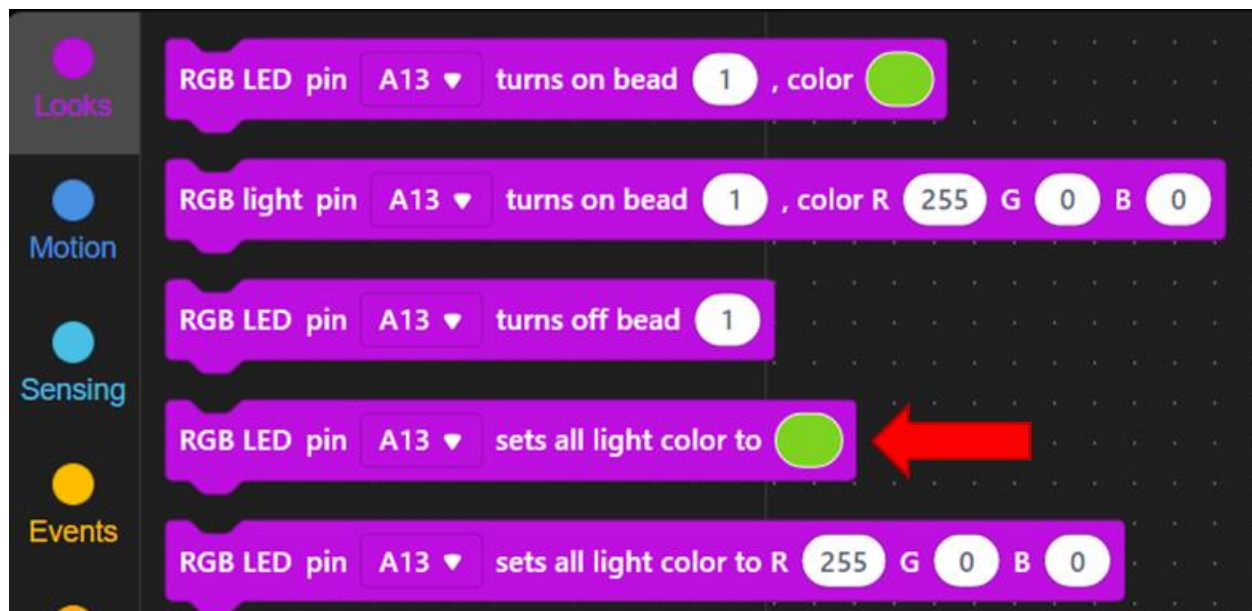








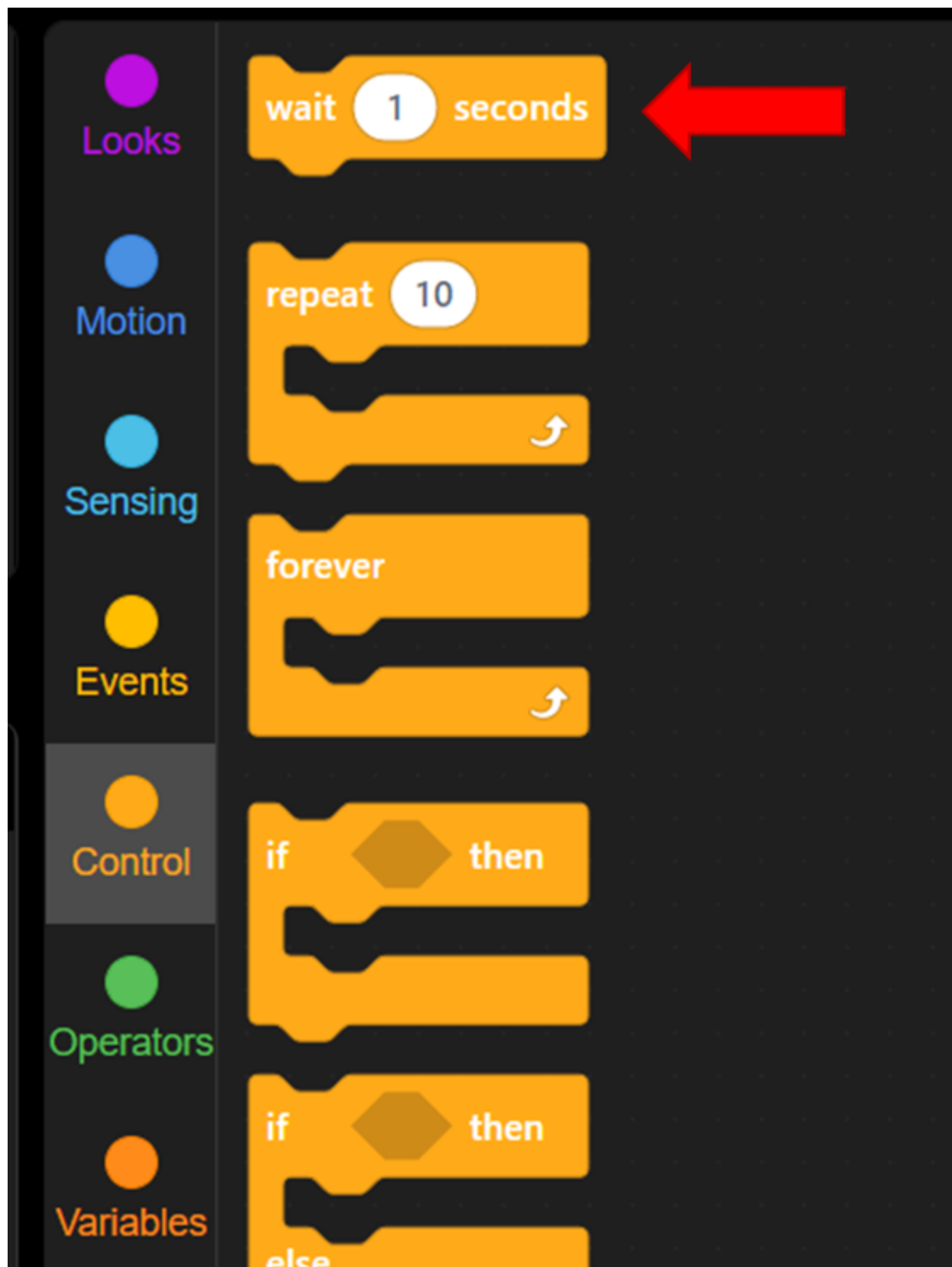


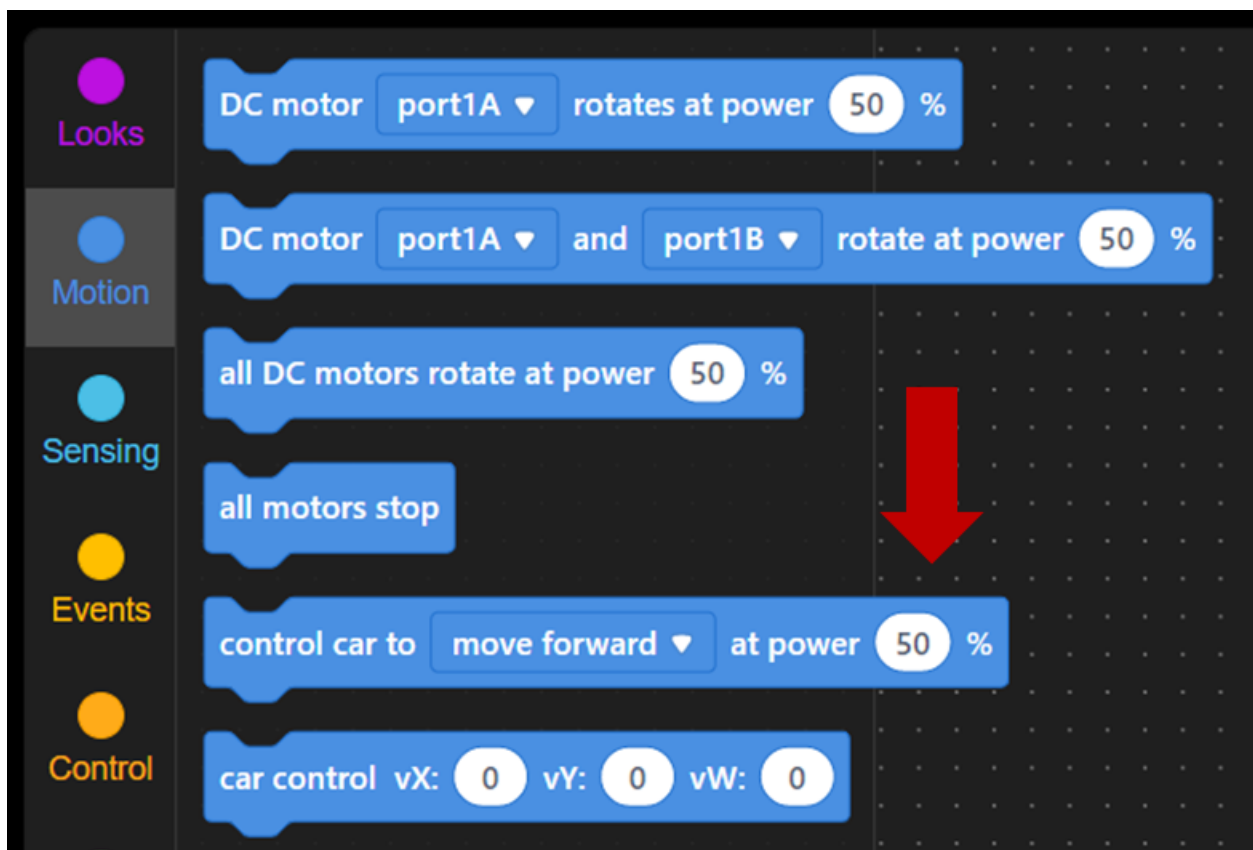
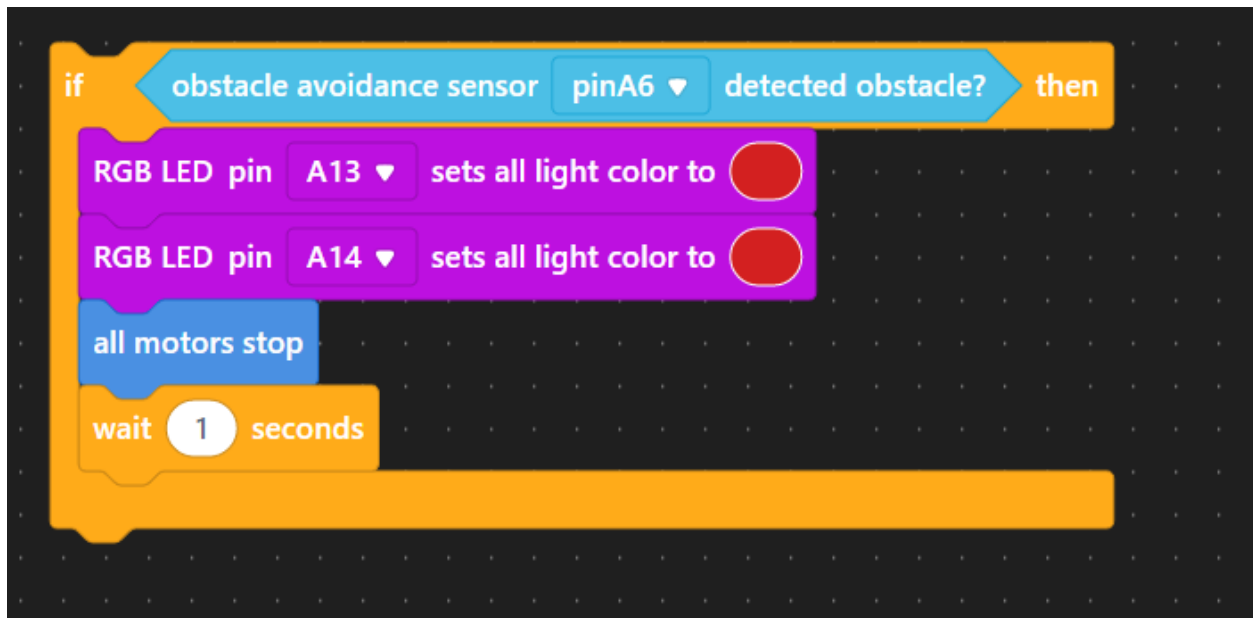


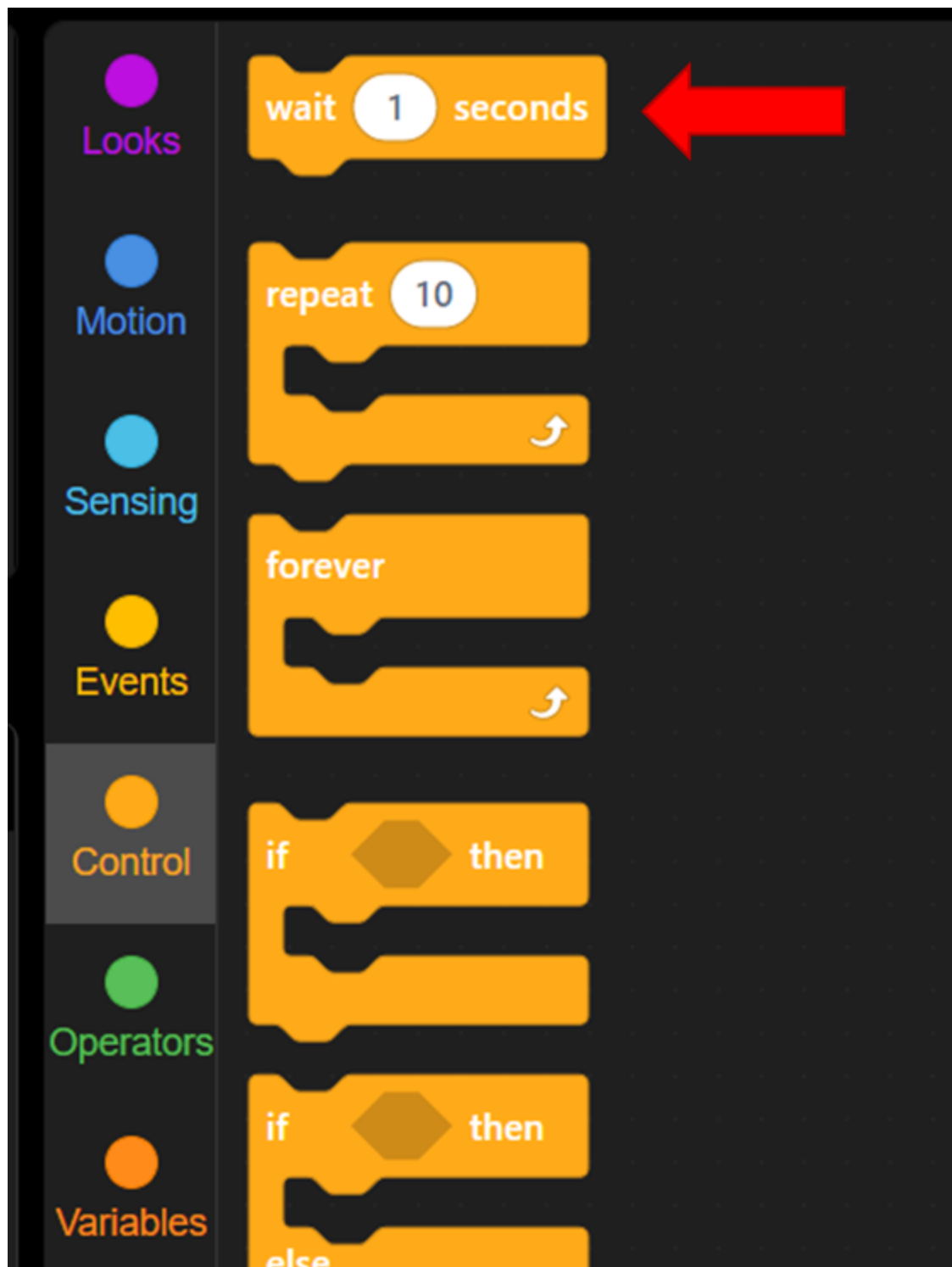
The image shows a Scratch script for controlling a car. The script is organized into categories on the left: Looks, Motion, Sensing, Events, and Control. The script consists of the following blocks:

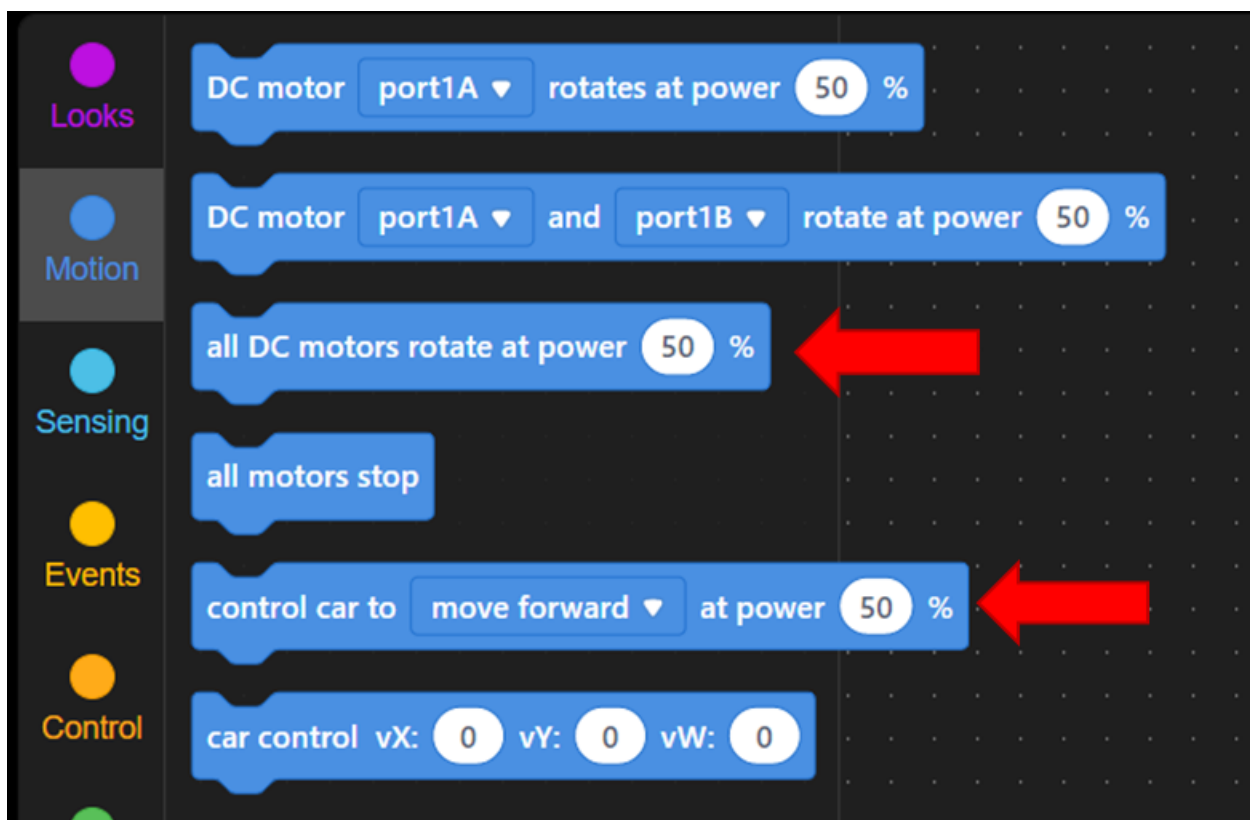
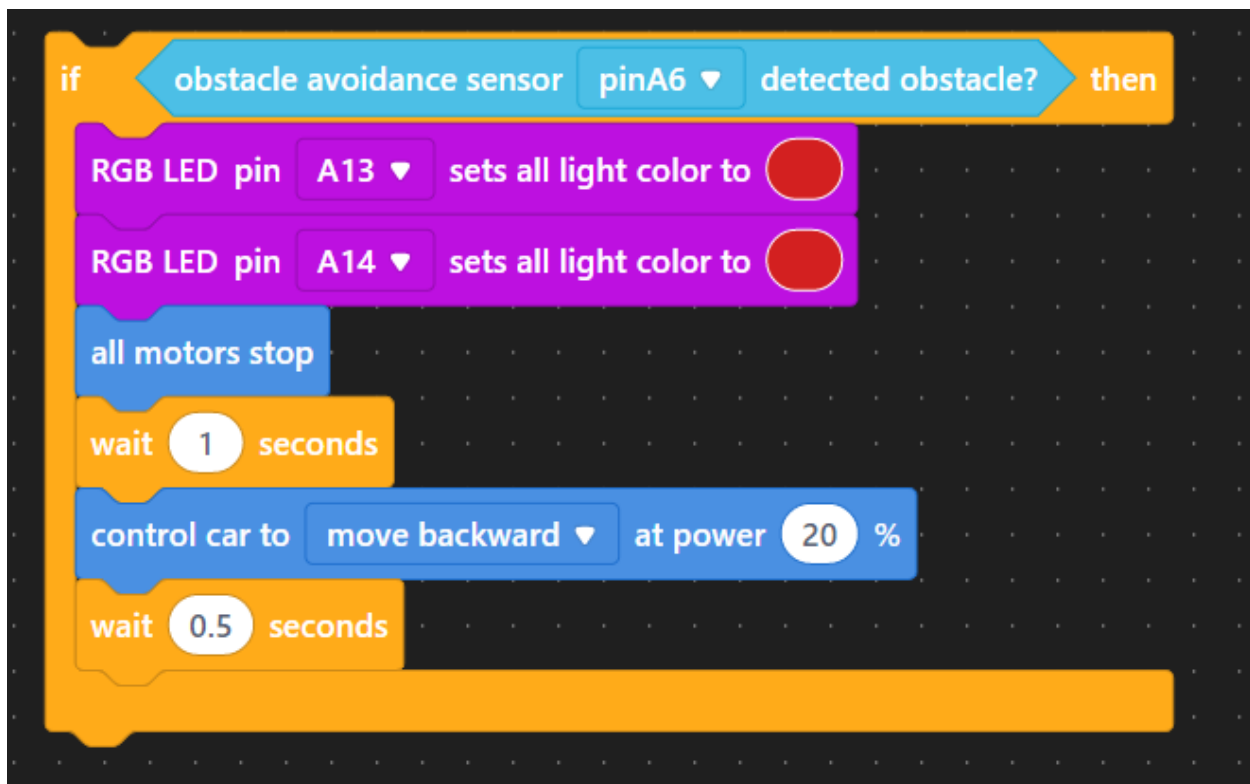
- Looks**: A blue block that says "DC motor port1A ▼ rotates at power 50 %".
- Motion**: A blue block that says "DC motor port1A ▼ and port1B ▼ rotate at power 50 %".
- Motion**: A blue block that says "all DC motors rotate at power 50 %".
- Motion**: A blue block that says "all motors stop". A red arrow points to this block from the right.
- Events**: A blue block that says "control car to move forward ▼ at power 50 %".
- Control**: A blue block that says "car control vX: 0 vY: 0 vW: 0".

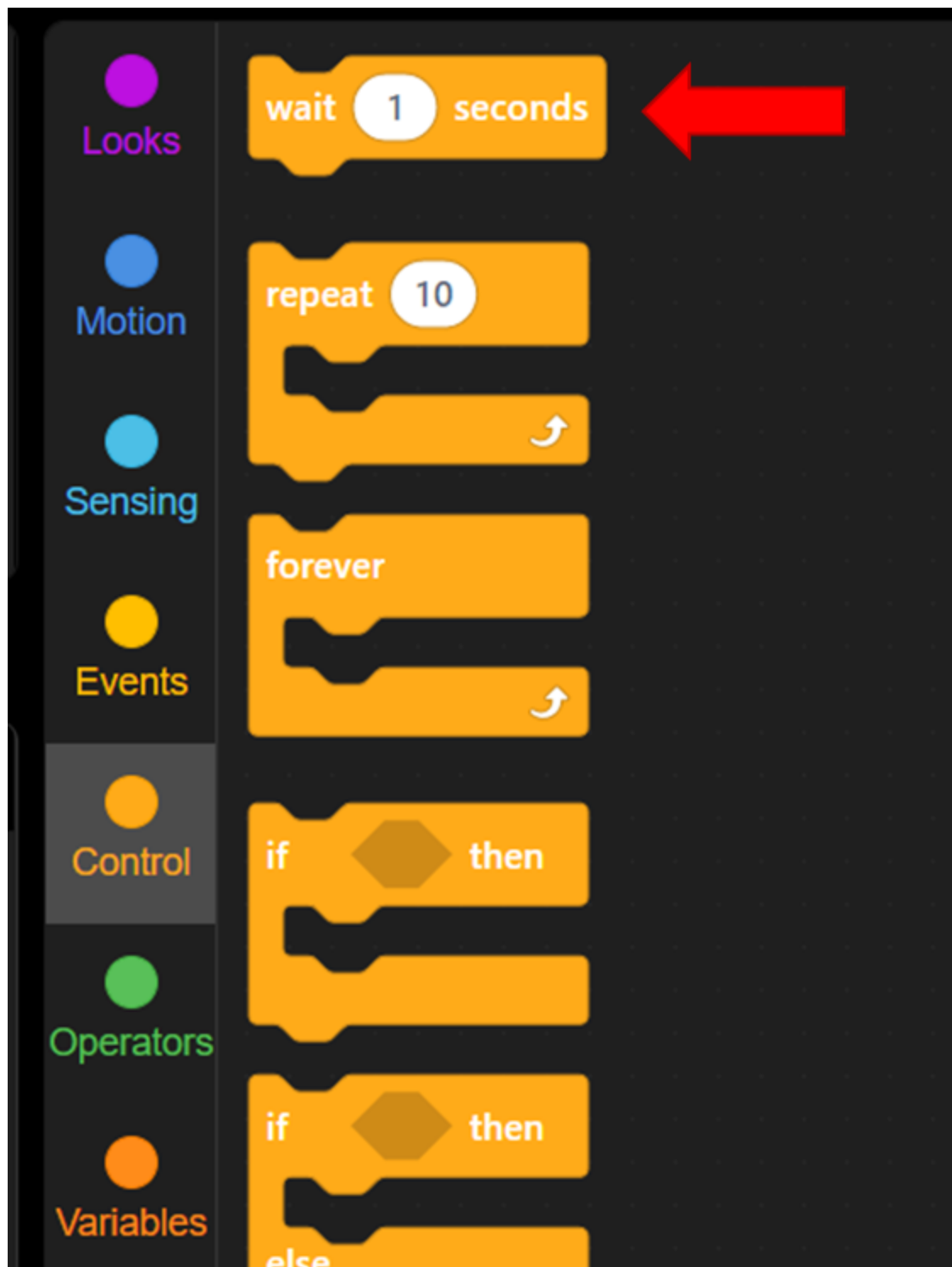
The script is set against a dark gray background with a light gray grid.

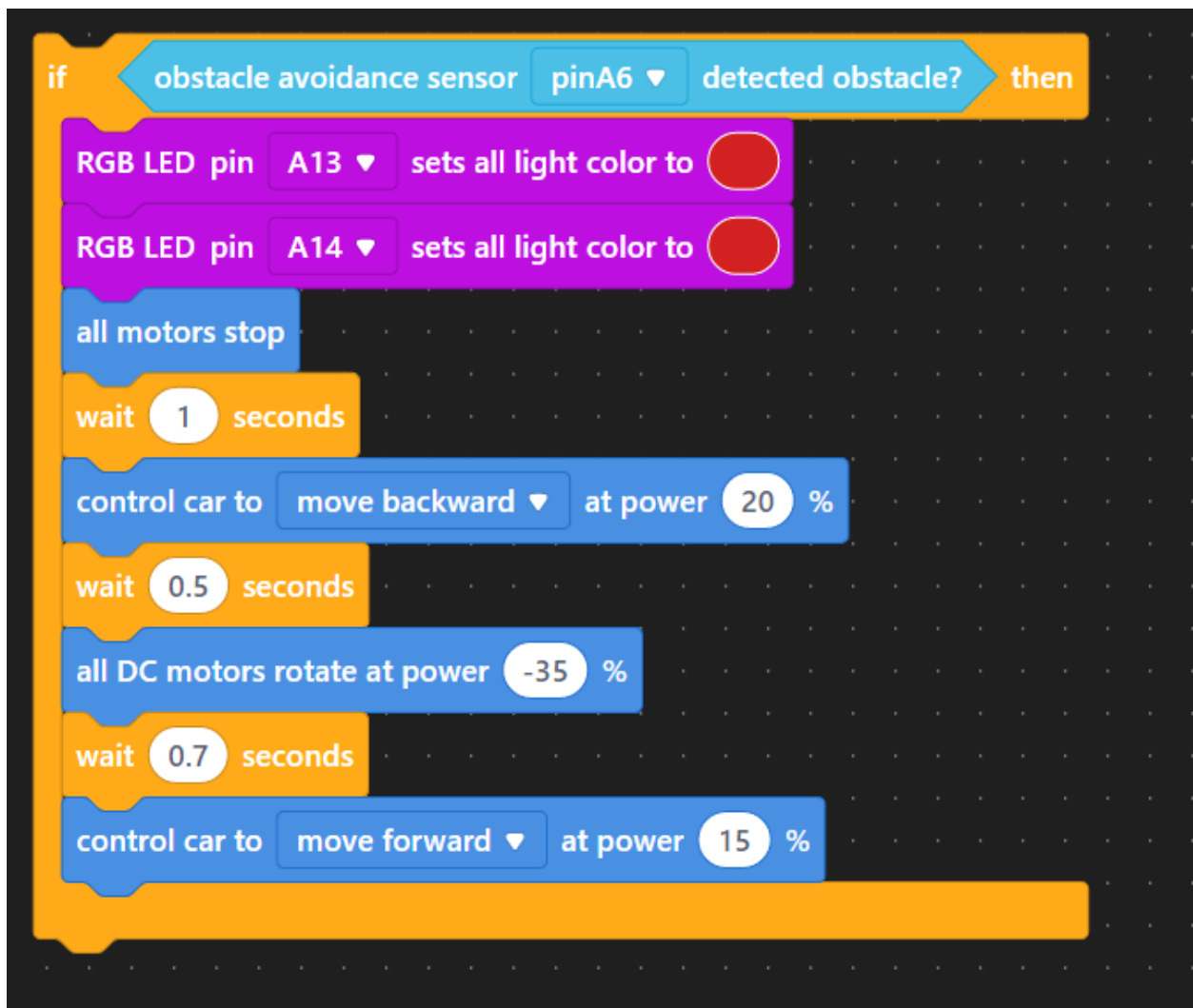




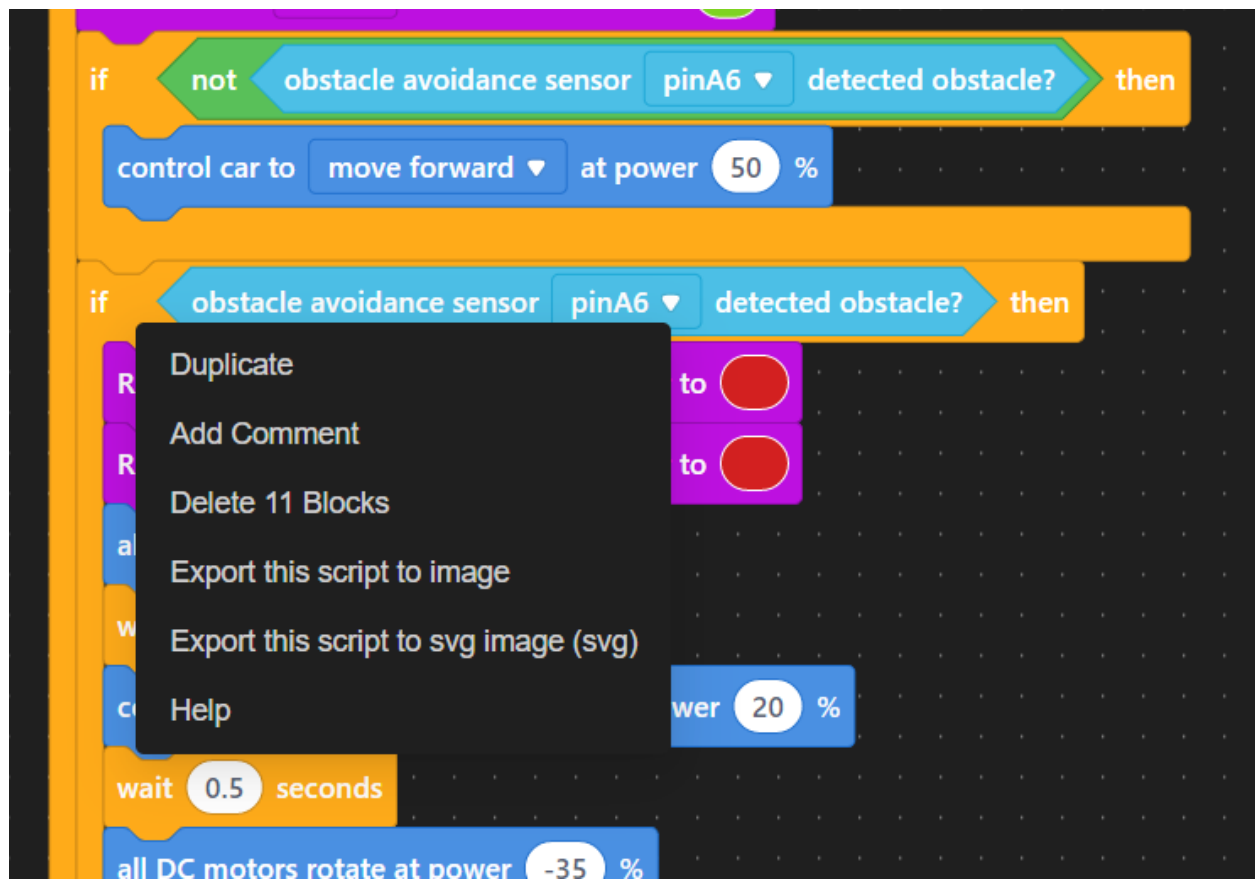


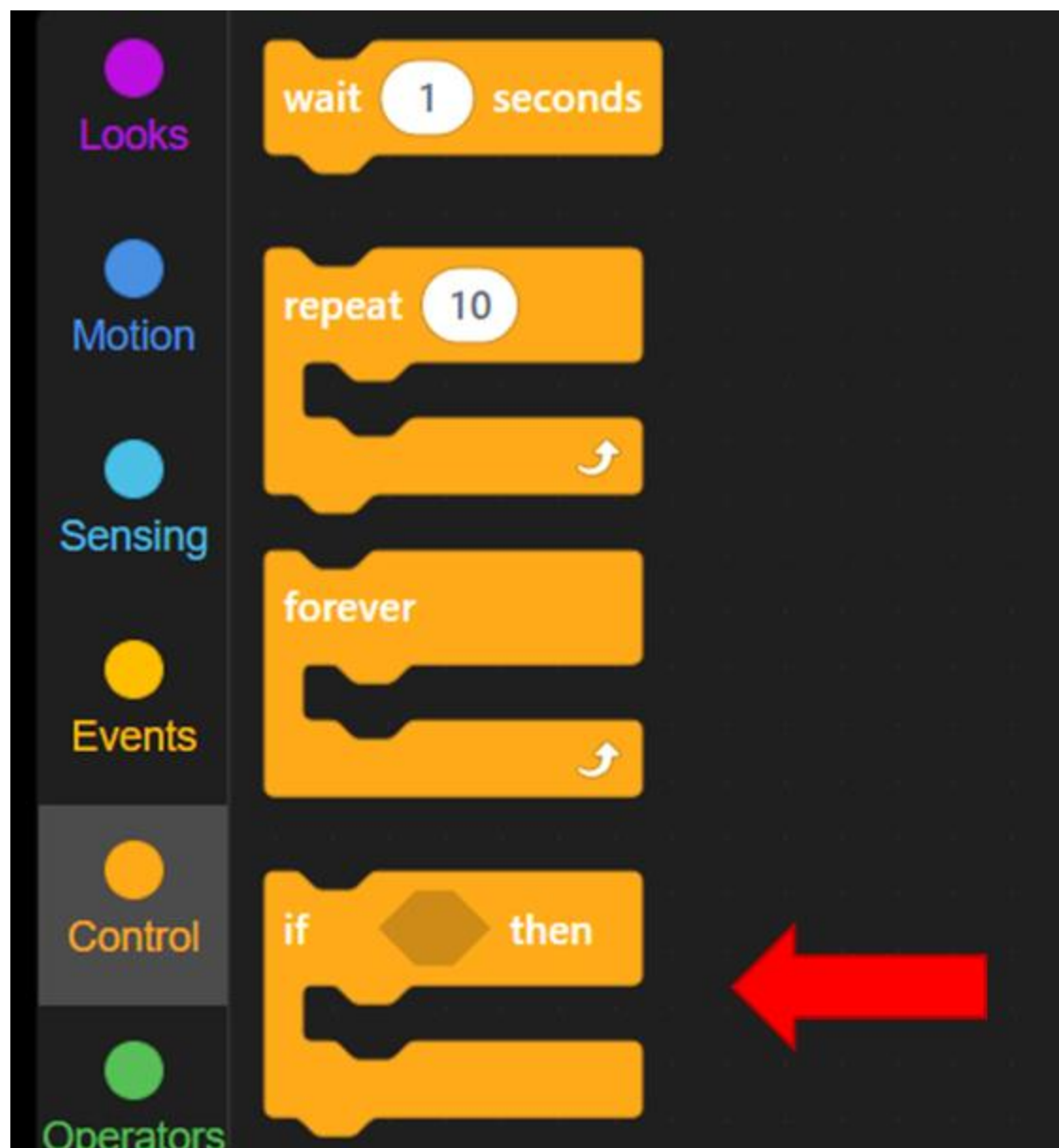


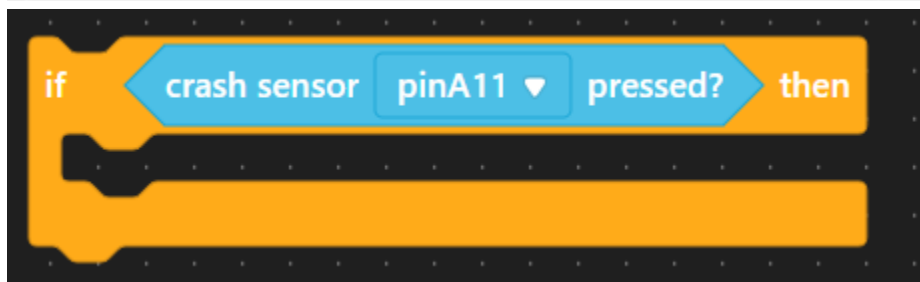
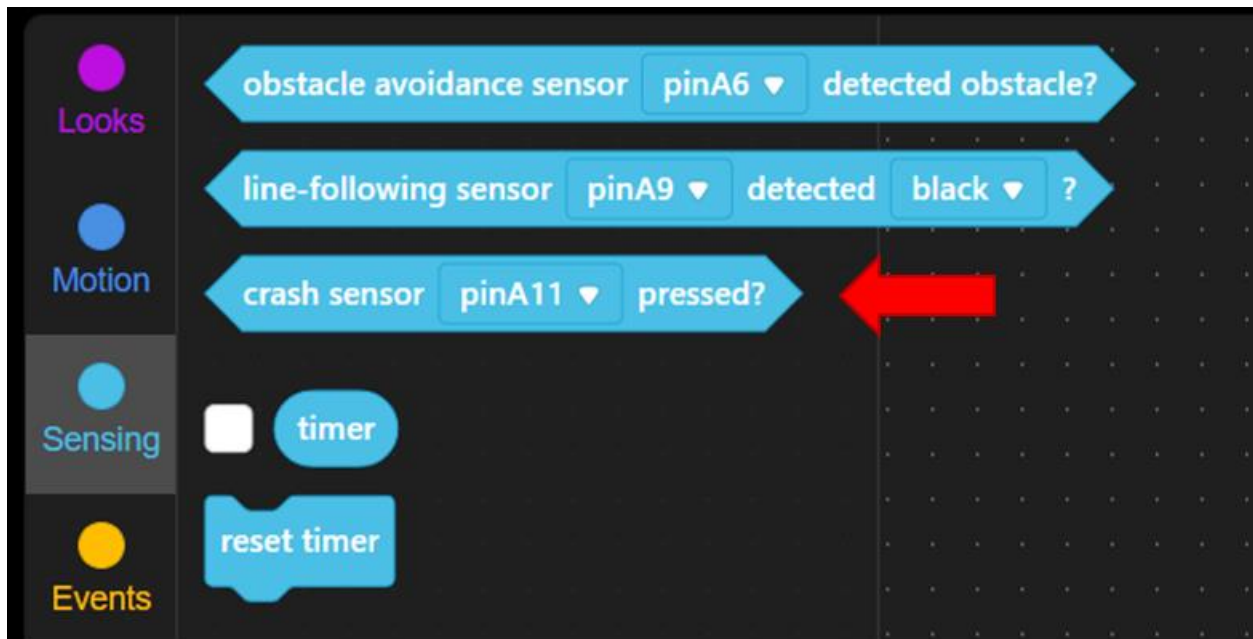


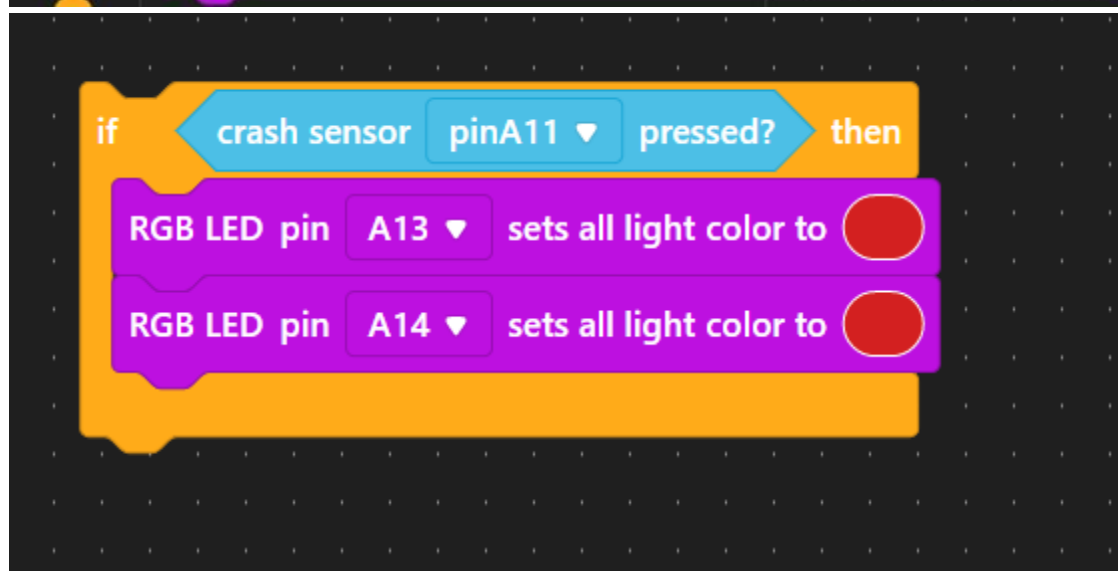
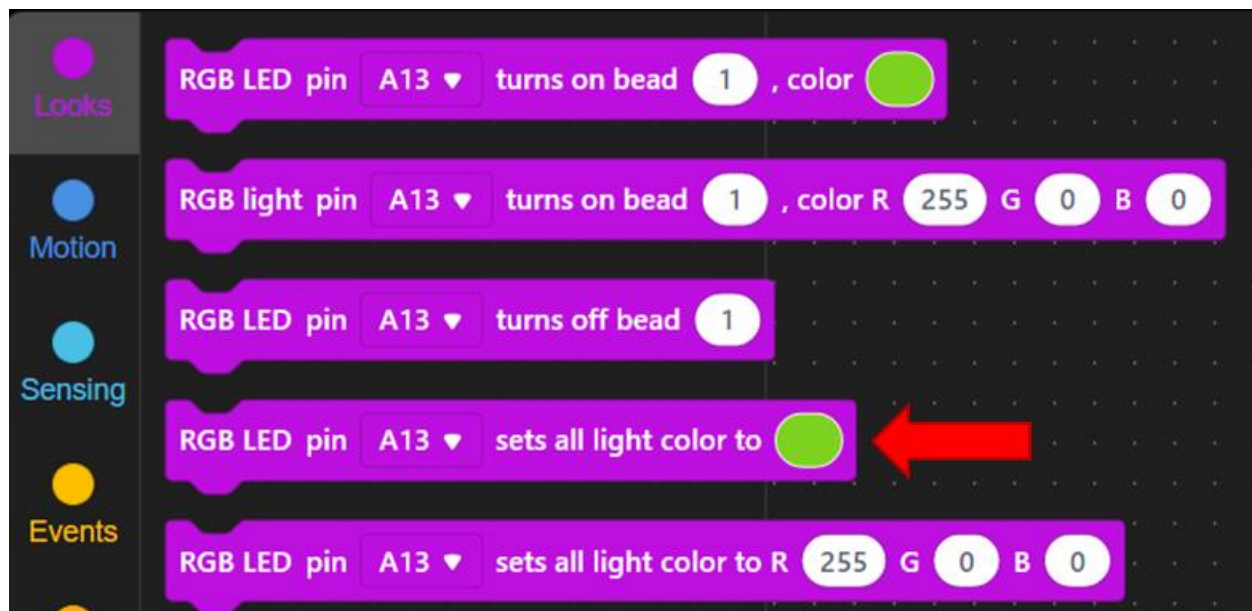


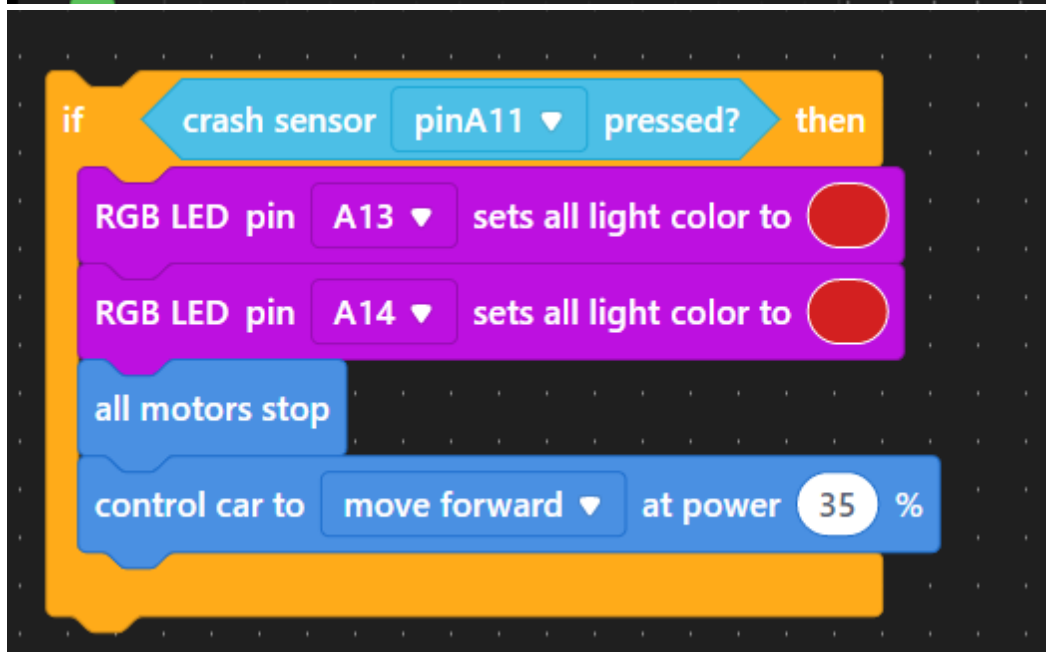
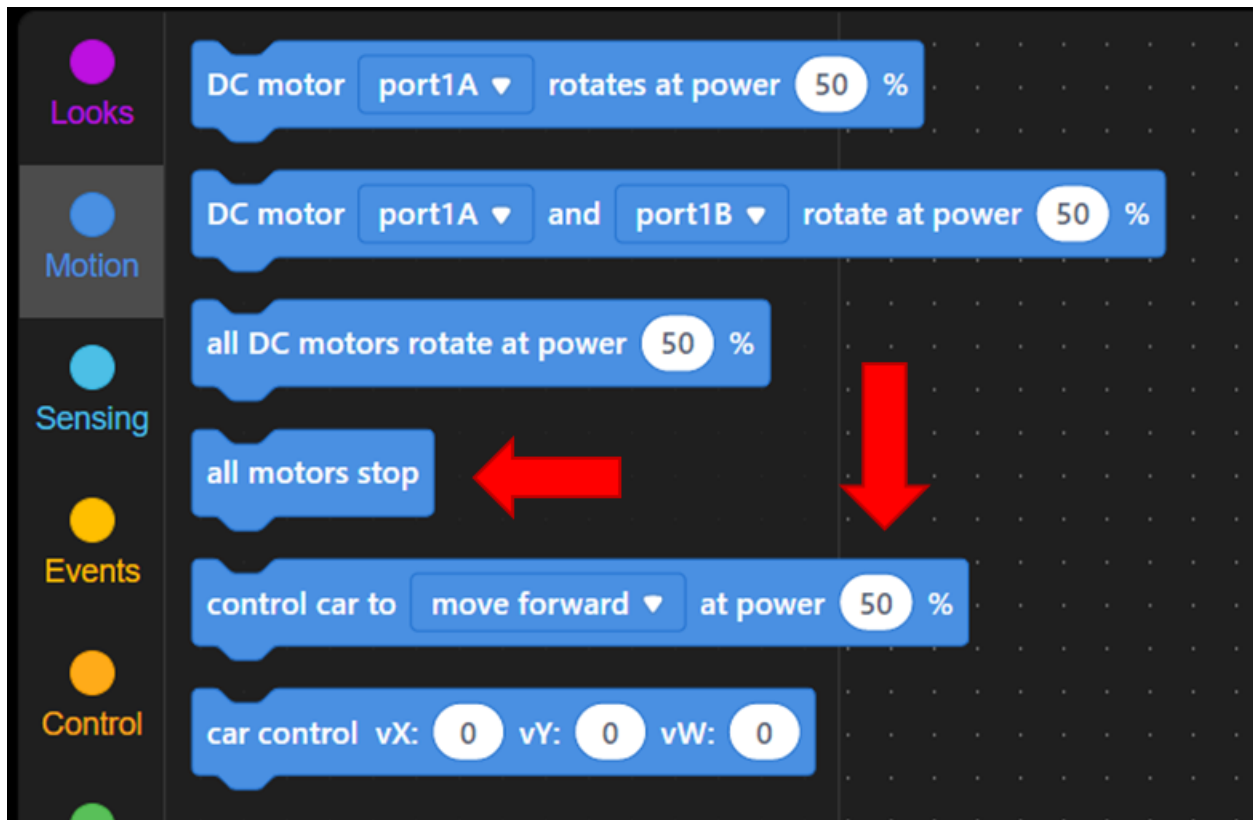
Duplicate both the if statements and change it to pinA8 by clicking right click and place it under the previously created if statements

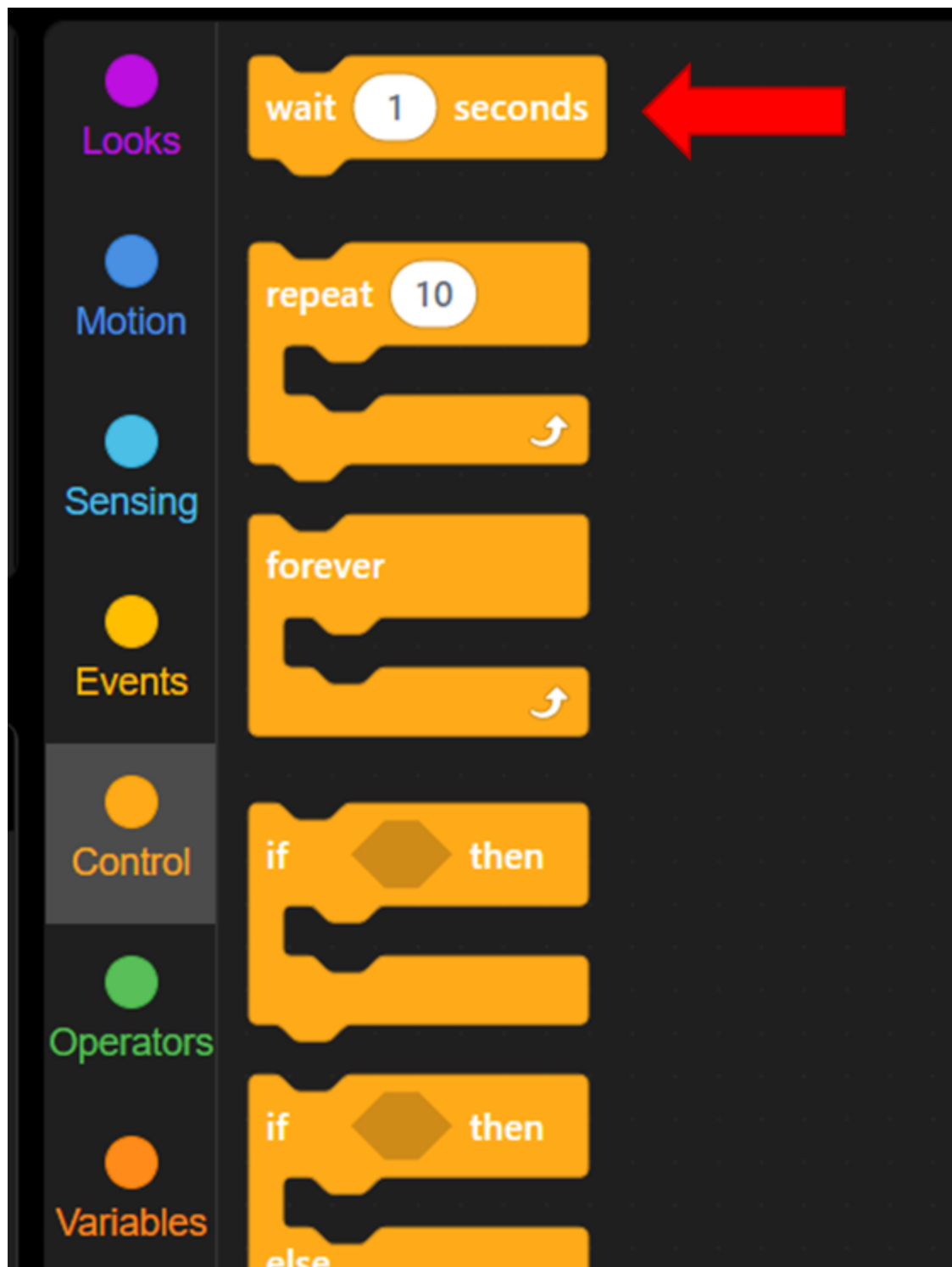


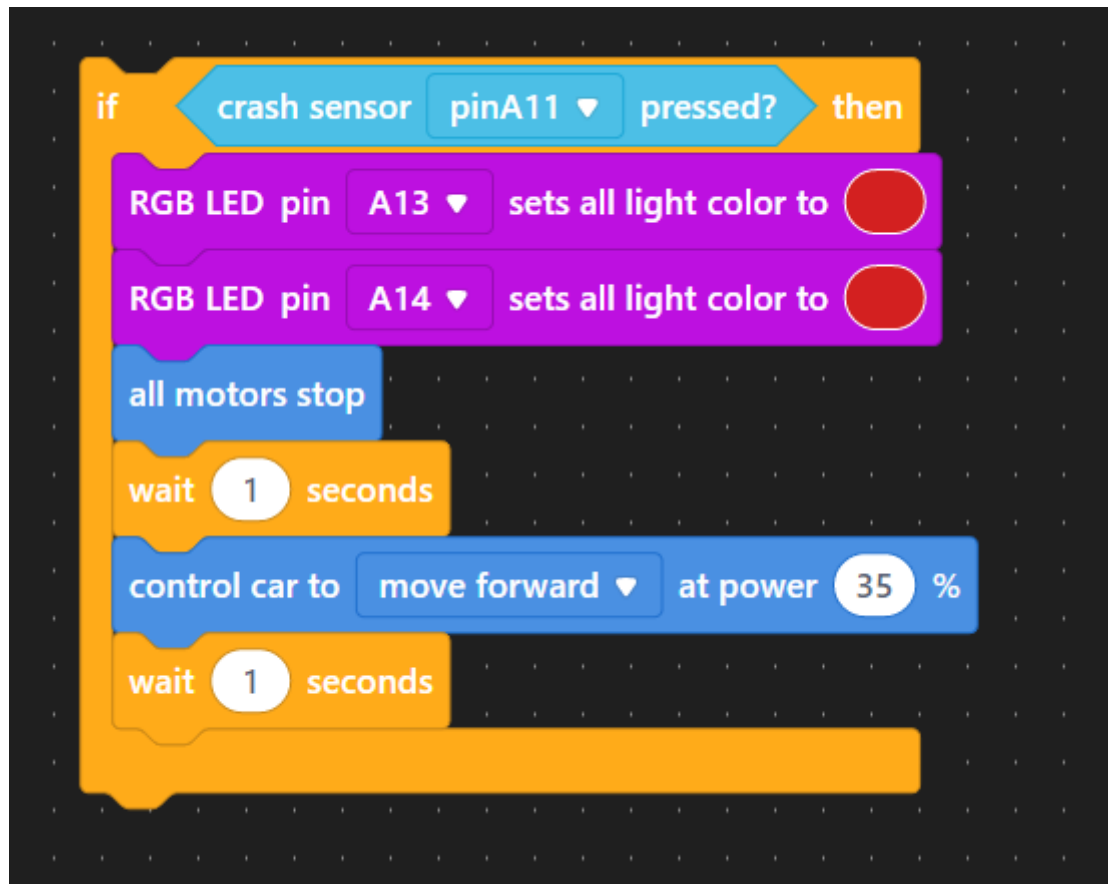




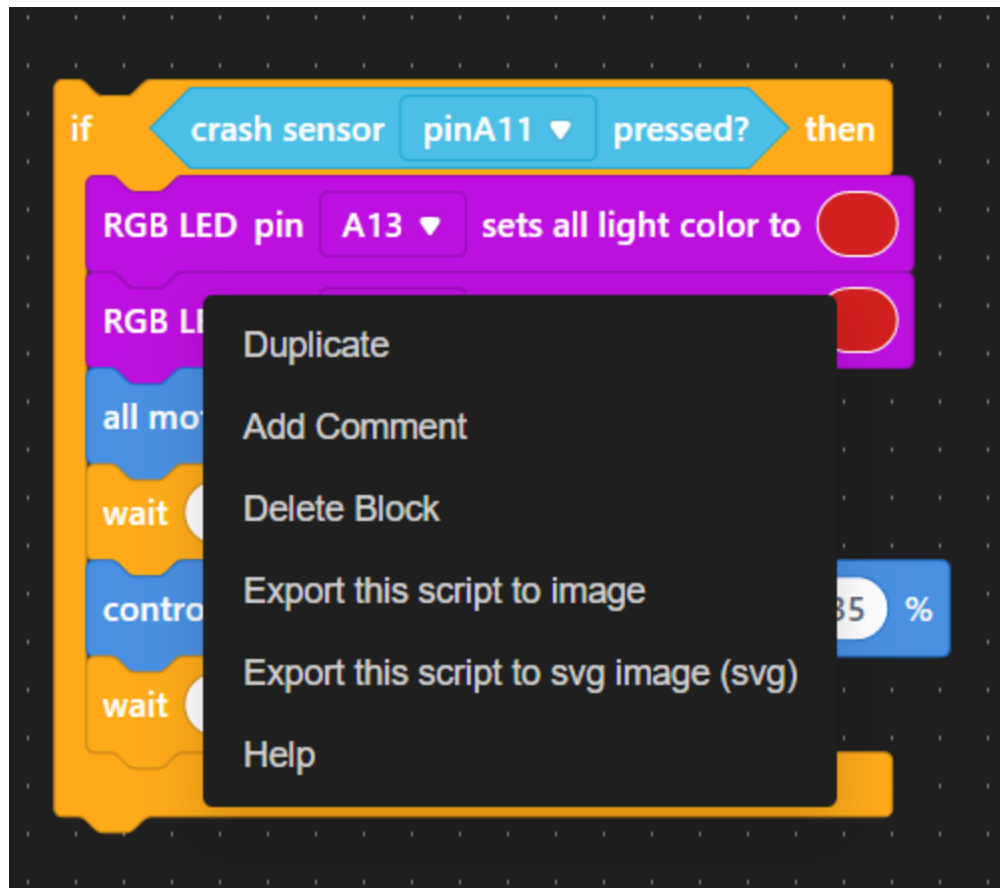




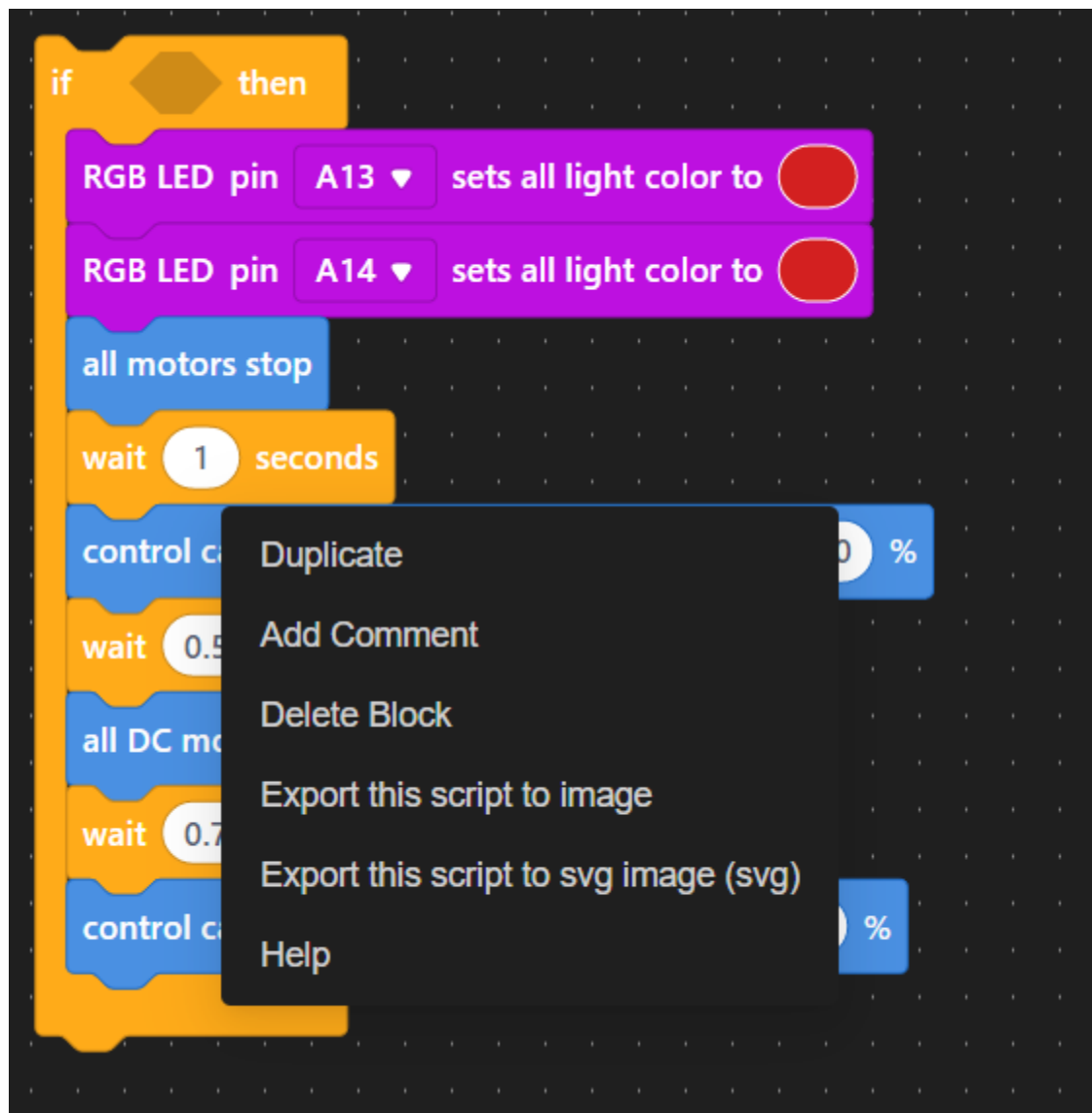


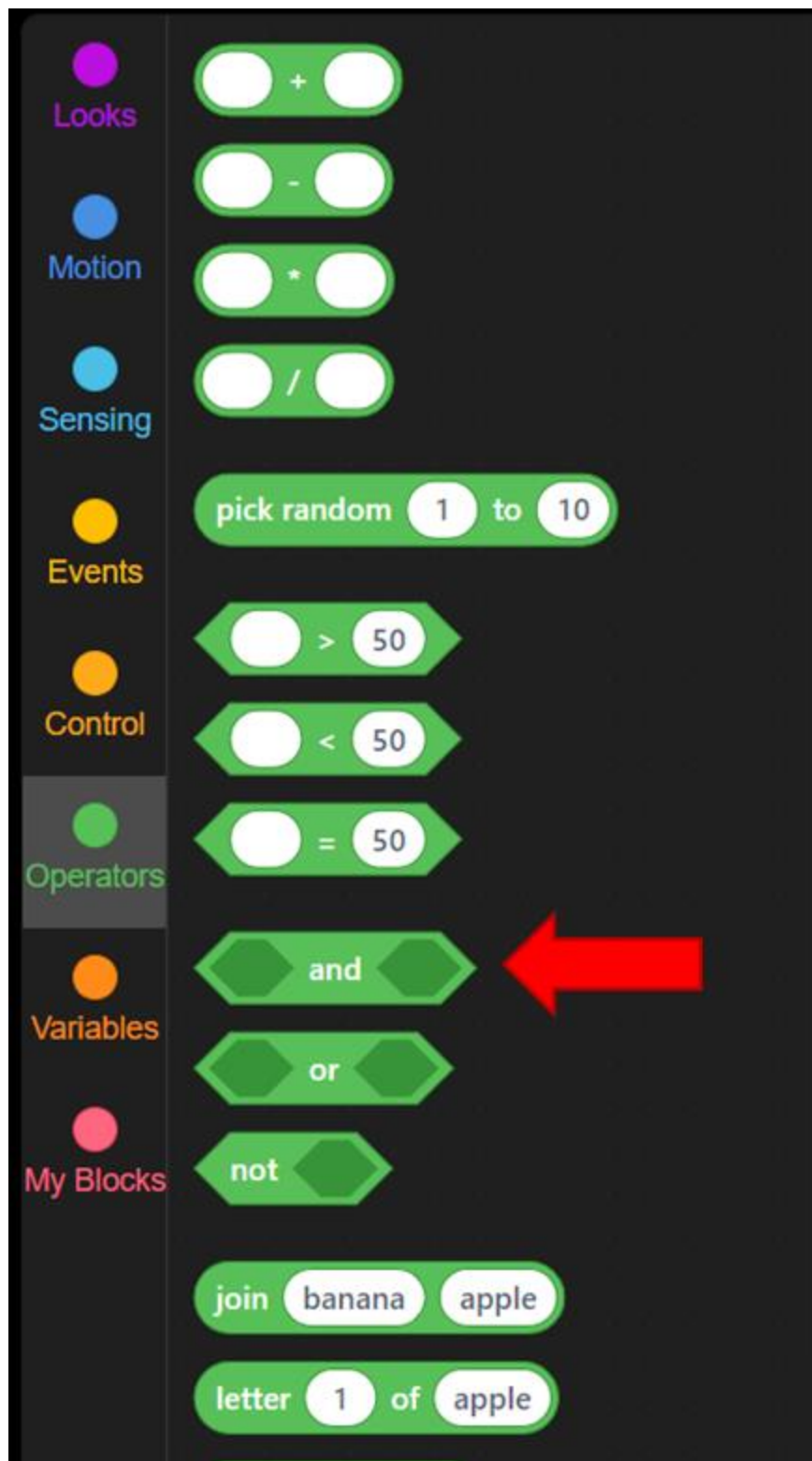


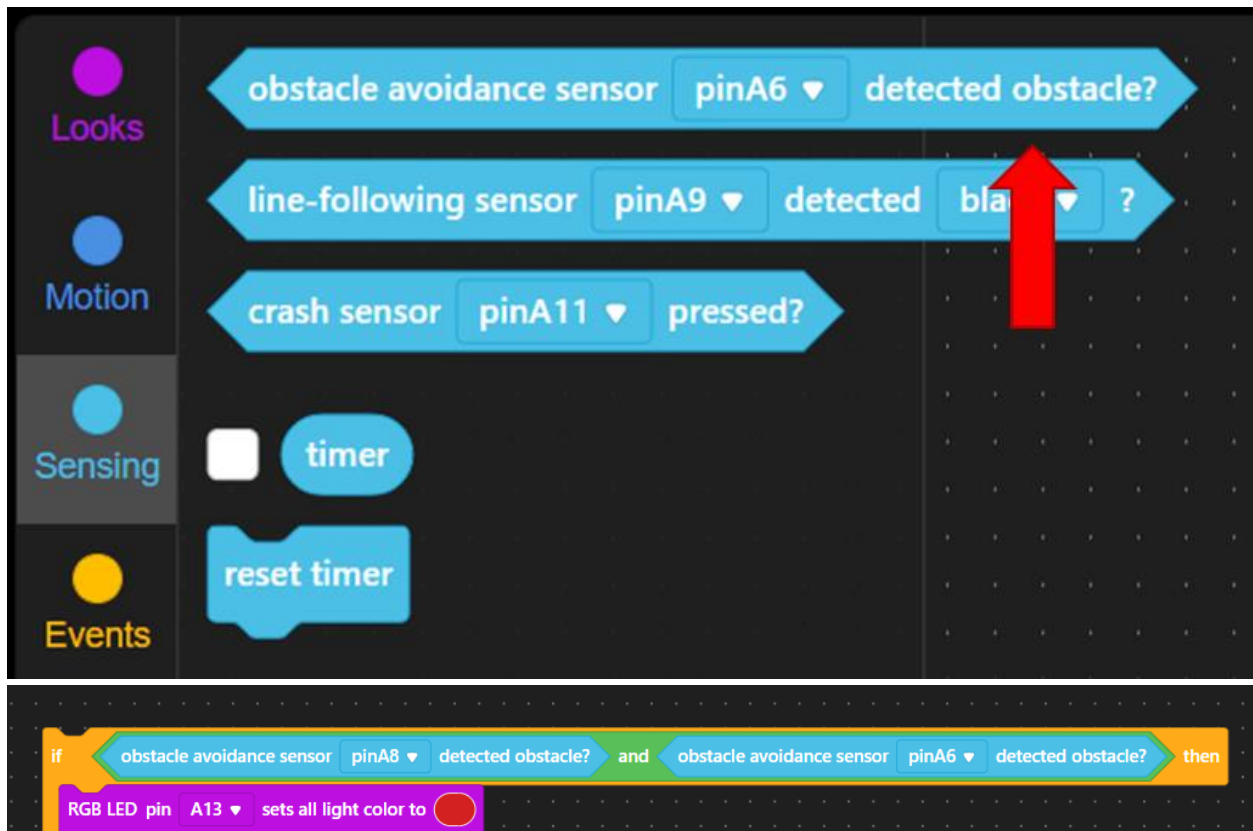
Duplicate it again by clicking right click. Set it for pinA12

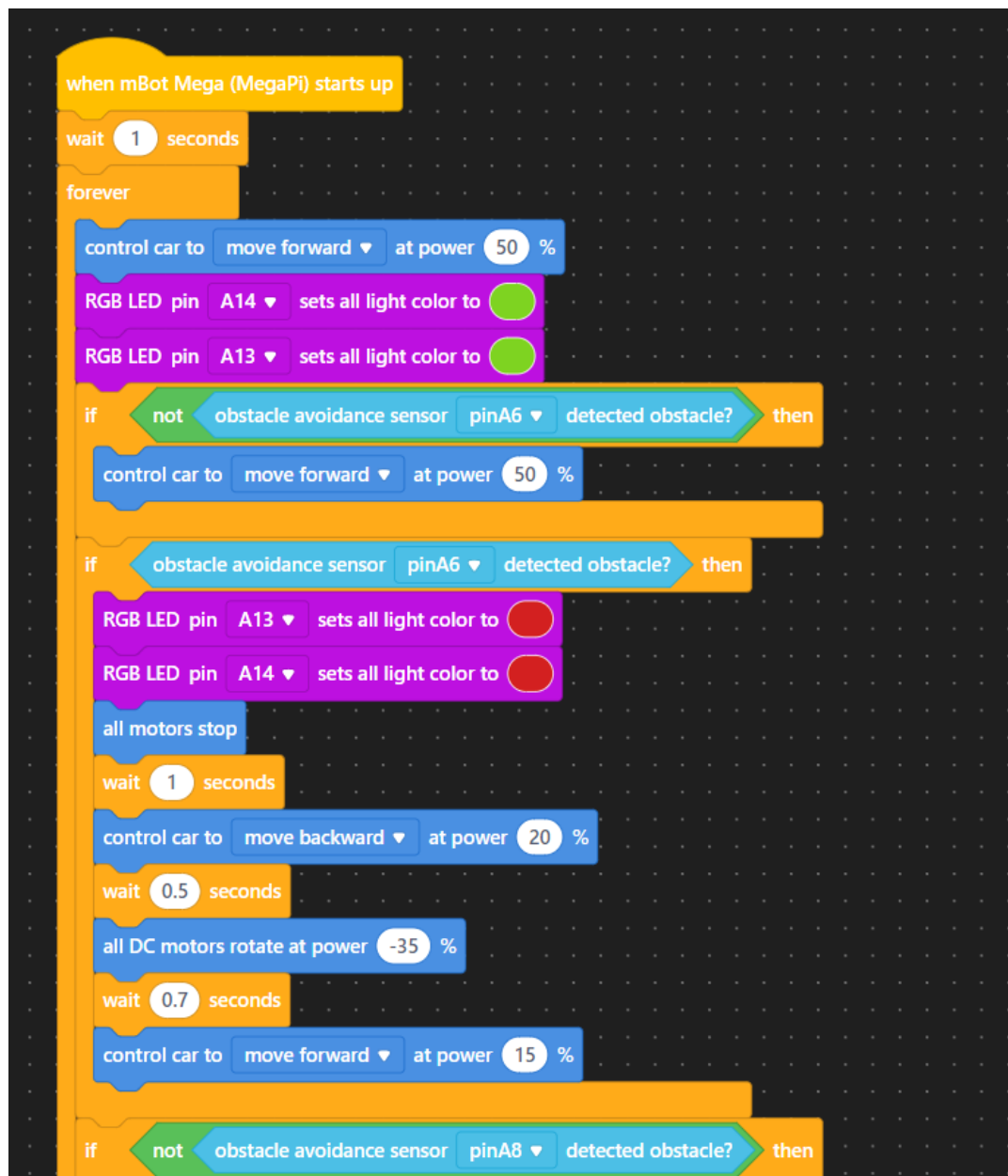


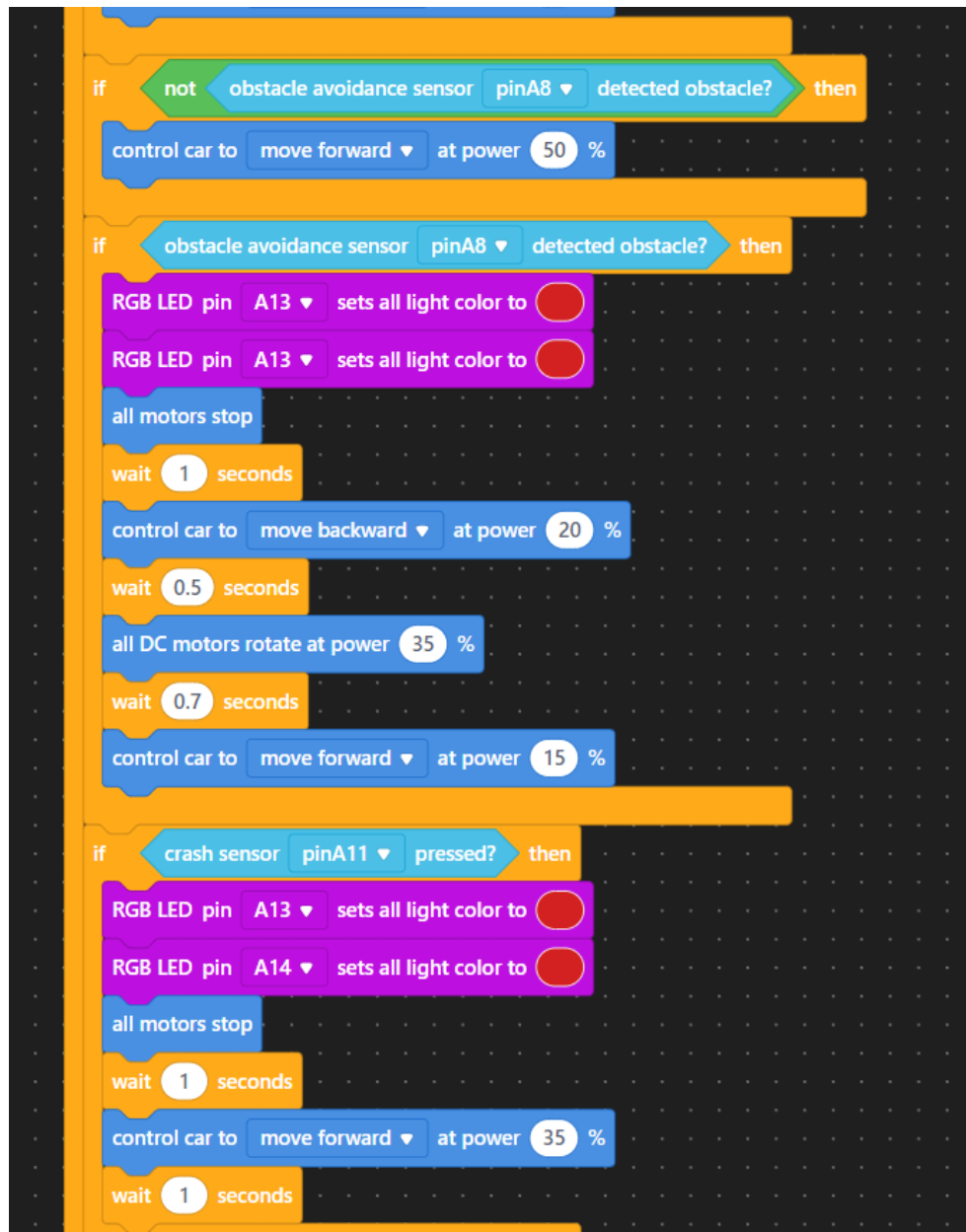
Duplicate the two first if statements we created by clicking right click and dragging them to the bottom

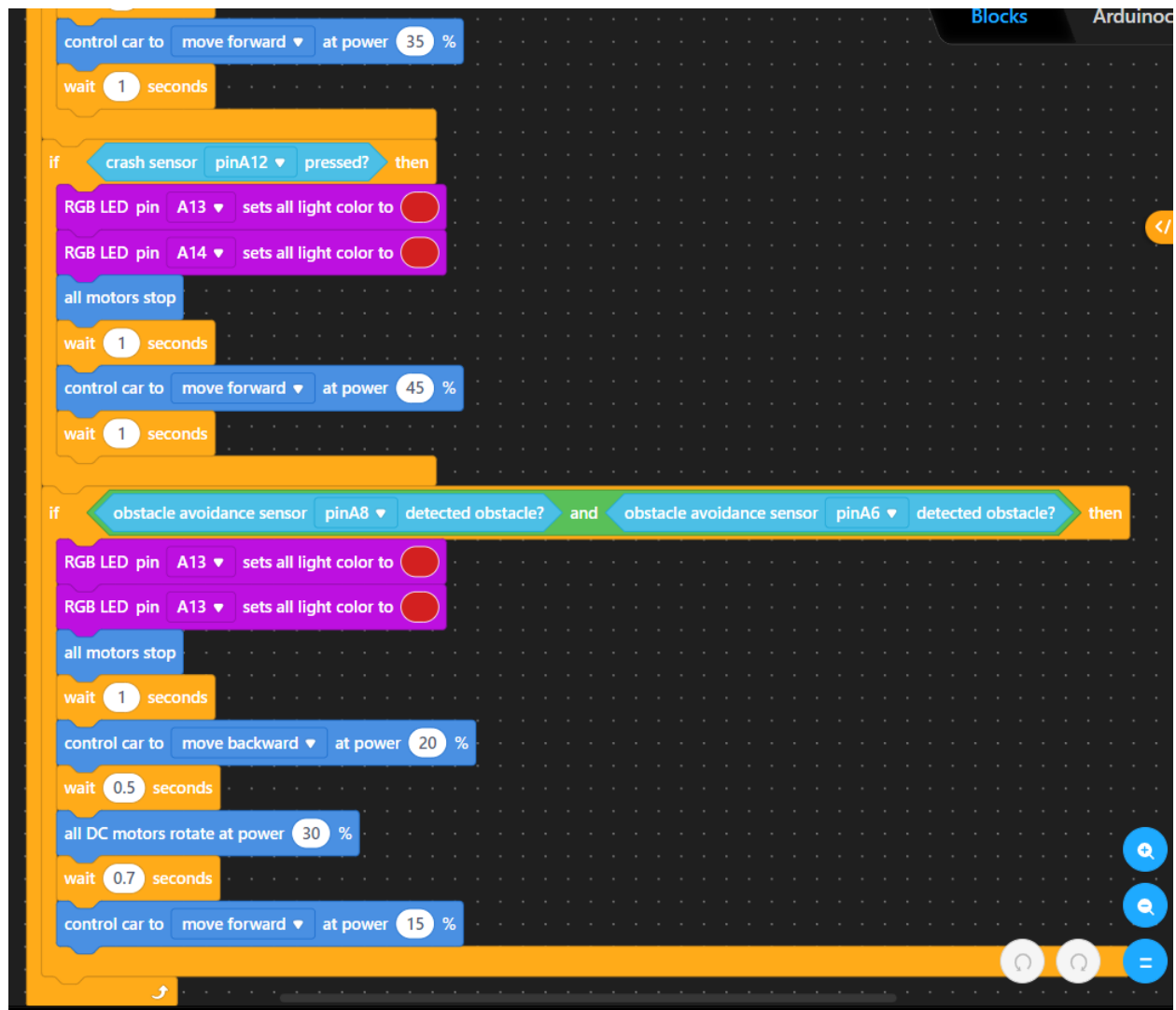












(All of this code can be found in our cheat sheet)

4. Between each part, the students will test run the bot using obstacles around them, or the created obstacle course