Introduction to Robotics: Smart Cars

● Team Member "Green" ● Team Member "Blue" ◐ Shared Task

A

1 ◐ For this activity, you'll use an already-made "smart" car that uses a simple cardboard box, the MakerBit, along with some motors and wheels.

As an added bonus, this project uses the BBC micro:bit as the microcontroller, rather than the Arduino.

In fact, Activities 1 and 2 can be done exactly the same using the micro:bit, and Activity 3 (coding) can be done using the Microsoft MakeCode block programming language for the micro:bit instead of iForge.

2 ◐ If you have not already done so, put away the MakerBit, cables, touchpoints and the volcano model in the tray and on the table as you found them when you first started.

3 ◐ Locate the MakerBit smart car on the activity table, or ask the makerspace facilitator for one, and then proceed to the next page.

Sign in to MakeCode & Name Your Program

B

🟢 Team Member "Green" 🔵 Team Member "Blue" ◐ Shared Task

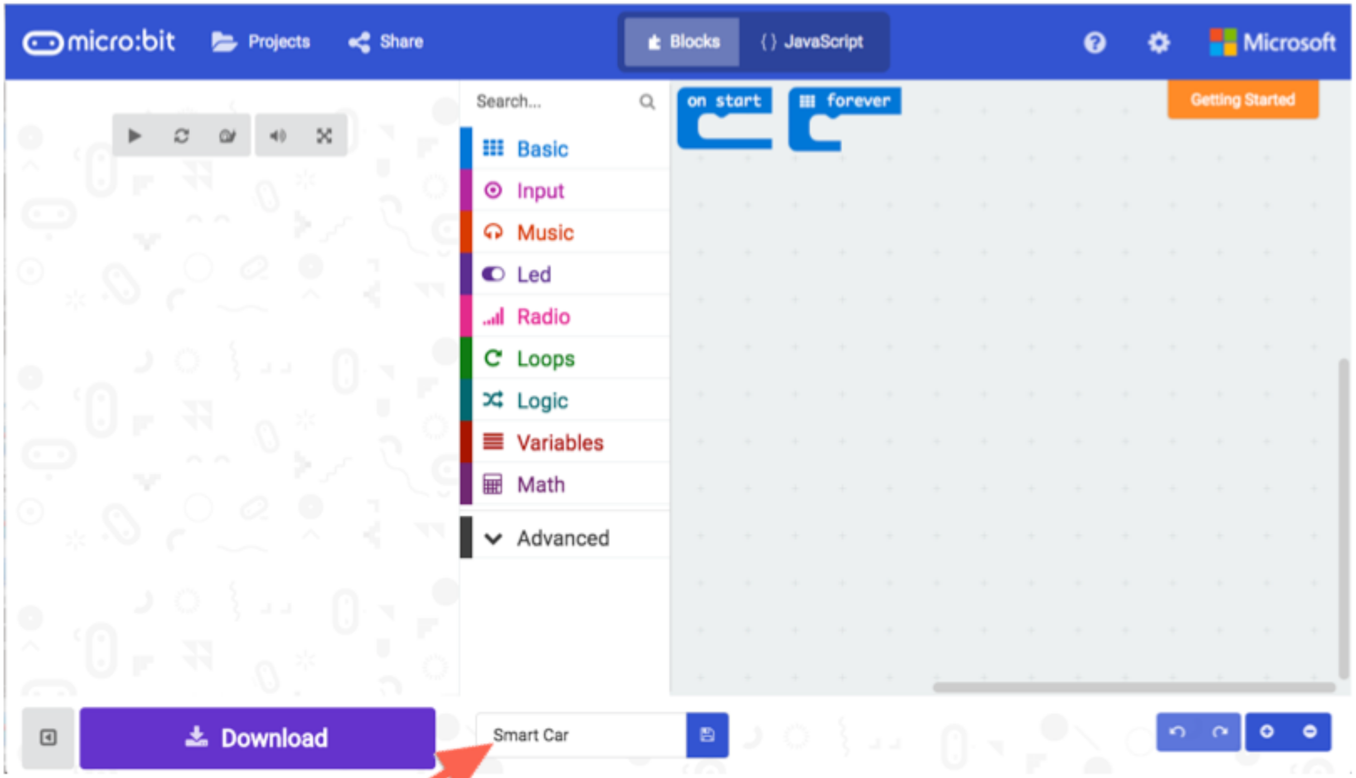
🟢 **1**

Type in this URL:

http://bit.ly/2P158Sp

to open the MakeCode graphic programming window on the computer you will be using to upload code to the smart car MakerBit.

Close the Media Linker app if it is open.



Smart Car

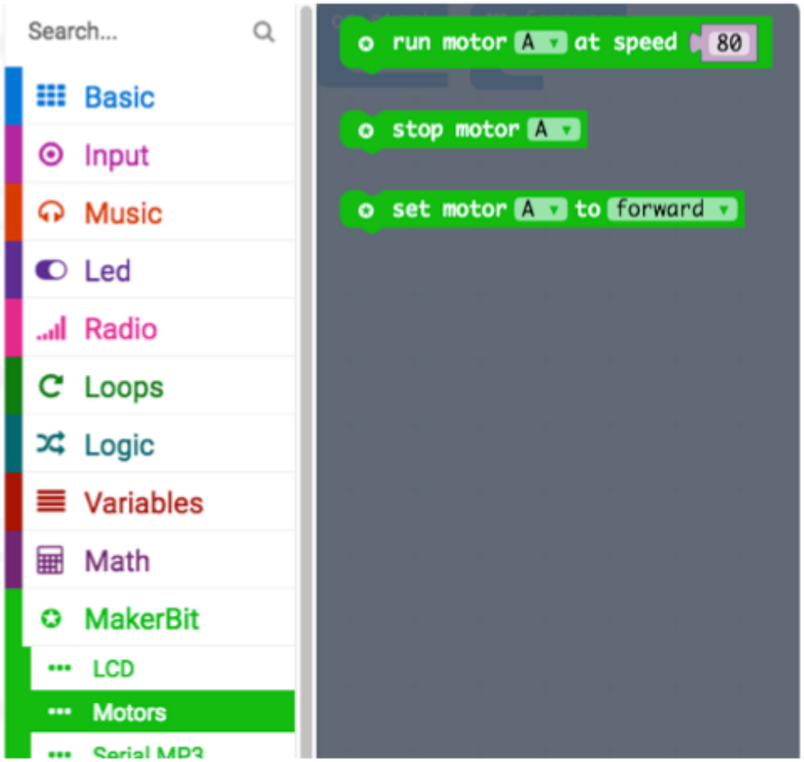
🟢 **2**

Enter a new name for your program, for example, "Smart Car".

Add Motor Control Blocks

C

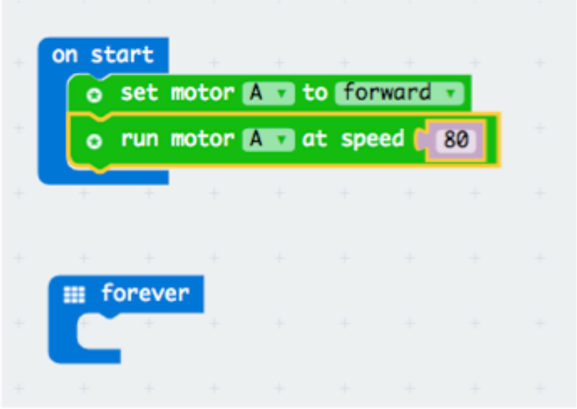
1. Click on “MakerBit” and then Motors to display the motor control blocks:



```
run motor A at speed 80
stop motor A
set motor A to forward
```

2. Click on “set motor A to forward" to place that block in the "start" block/

Repeat to drag “run motor A at speed 80" to the start block just below the other block.



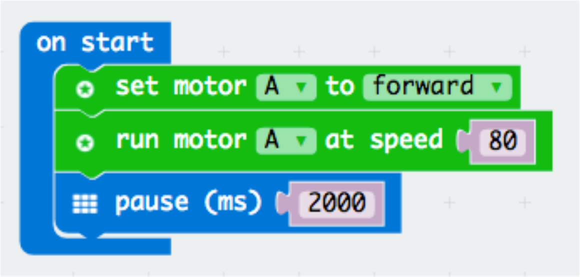
```
on start
    set motor A to forward
    run motor A at speed 80

forever
```

Note: If you add the wrong block, you can delete it by clicking on it and pressing the Delete key.

Add the blocks to motor B

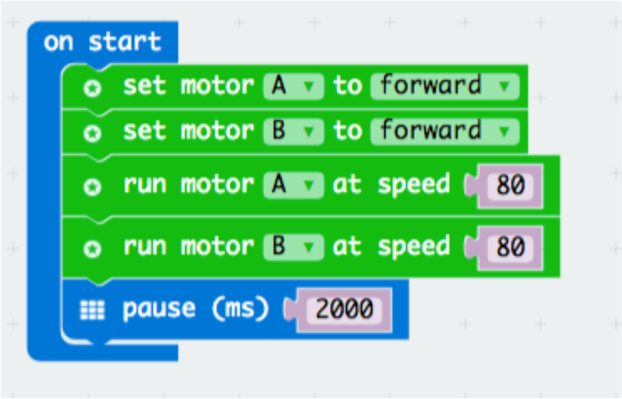
D

4 Click on the Basic category, and add a block for a pause. "2000" will be 2000 ms (milliseconds), which is equal to a pause of 2 seconds before going to the next set of blocks.



```
on start
    set motor A to forward
    run motor A at speed 80
    pause (ms) 2000
```

5 Repeat the process of dragging the set motor and run motor blocks, but edit the "A" to "B" and insert the "B" blocks after their matching "A" blocks.



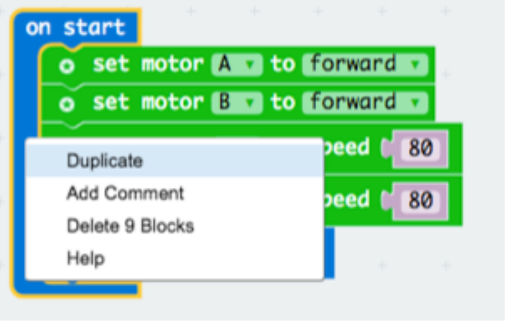
```
on start
    set motor A to forward
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
```

Adding Functions for Forward & Reverse

E

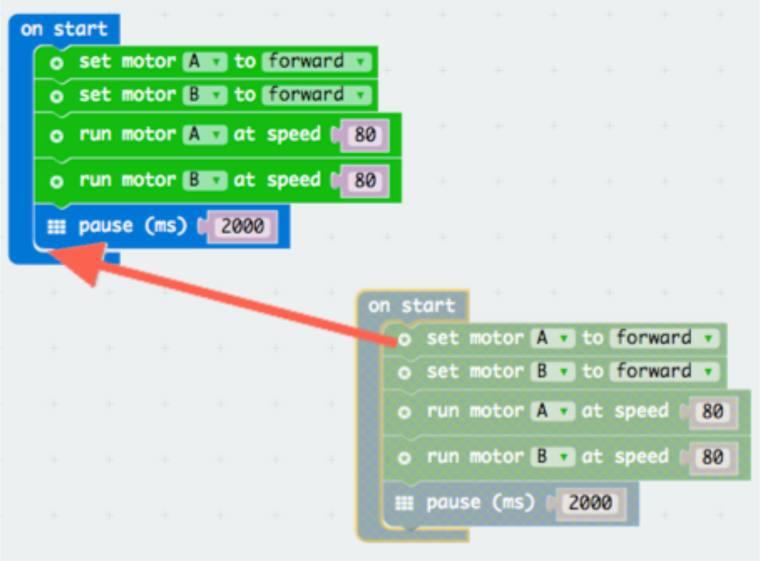
8 To create the set of Reverse blocks, let's learn a MakeCode trick for duplicating a group of blocks.

Control-click on the "start" block, and choose "duplicate".



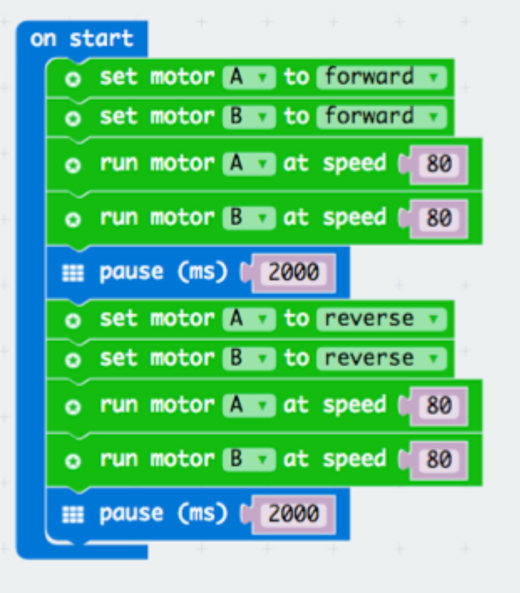
```
on start
    set motor A to forward
    set motor B to forward
    ...peed 80
    ...peed 80
```

9 Drag the entire on start group down and to the right. Then click and drag the first block INSIDE "on start" so that you can add all of them just below the "pause" block. Then drag the left-over duplicate "on start" block back to the categories to delete it.



```
on start
    set motor A to forward
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000

on start
    set motor A to forward
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
```

10 Edit "forward" to "reverse" for both A and B.

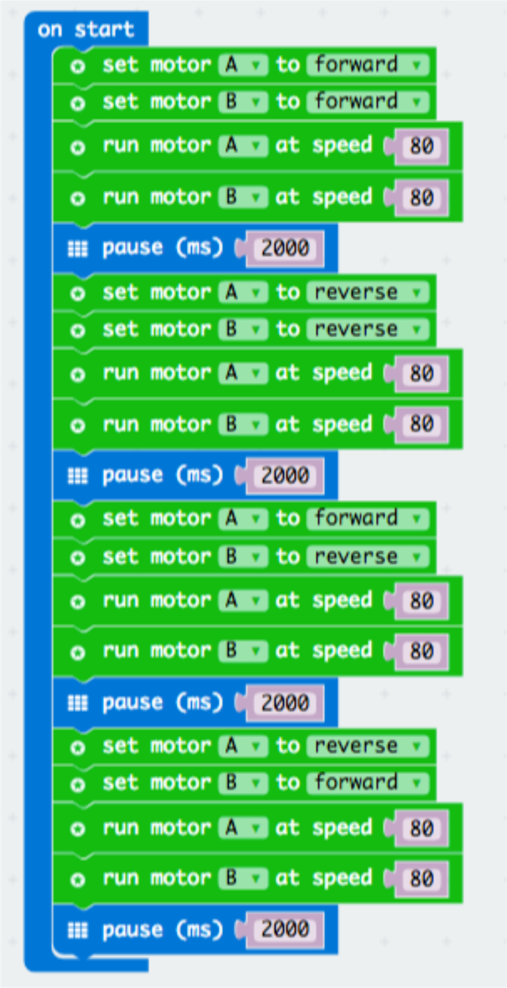


```
on start
    set motor A to forward
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
    set motor A to reverse
    set motor B to reverse
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
```

Combine Forward and Reverse for a Turn

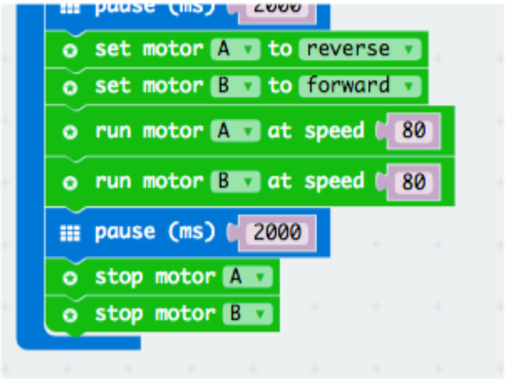
F

8 To make a turn, move one motor forward while the other is reverse:



```
on start
    set motor A to forward
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
    set motor A to reverse
    set motor B to reverse
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
    set motor A to forward
    set motor B to reverse
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
    set motor A to reverse
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
```

9 A full stop is easy. Just add the stop blocks.

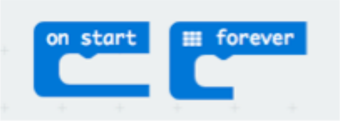


```
    pause (ms) 2000
    set motor A to reverse
    set motor B to forward
    run motor A at speed 80
    run motor B at speed 80
    pause (ms) 2000
    stop motor A
    stop motor B
```

On Start and Forever Loops

E

1 At the top of the MakeCode workspace were blocks named "on start" and "forever". The difference is consistent with the names. "On start" is only done once when the micro:bit is first powered on (or reset) and "forever" loops as long as the device is running. If you want something to happen only once, put those instructions in the "on start" block. If you want it to happen over and over, put that in the "forever" block.

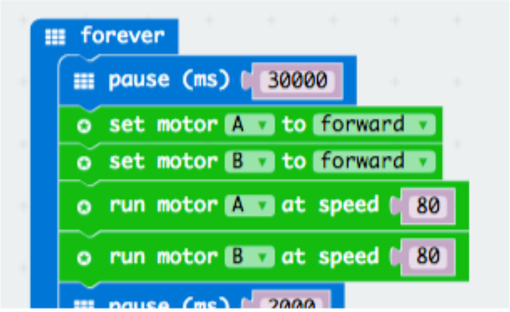


```
on start
```

```
forever
```

2 If you test your program now, the problem you'll encounter is that the motors will start as soon as you download the program on to the micro:bit, and then only run once. You won't have time to disconnect the usb cable and set the car on the ground.

One solution is to add a sufficiently long pause at the beginning of your program. Another solution is to move the blocks along with the pause to the forever block. That would look like this:



```
forever
  pause (ms) 30000
  set motor A to forward
  set motor B to forward
  run motor A at speed 80
  run motor B at speed 80
  pause (ms) 2000
```

"30000" ms (milliseconds) is 30 seconds. That should be enough time after downloading the program to disconnect the USB cable and set the smart car on the ground.

Download & Transfer Program

F

1 Putting your program on to the micro:bit is a 2-step process. First download from the MakeCode webpage, and second, move the file to the micro:bit.

To download, just click on the "Download" button. The program "smart car.hex" will be put into the downloads folder on your computer.

2 Next, connect the micro:bit to the computer using the USB cable. Then use the "files" icon in the applications folder.

You'll have about 30 seconds after you download to disconnect the USB cable and set the smart car on the ground before the motors start.

Test the Program & Iterate

G

1. Try changing the order of forward, reverse and turns to make the car move in a specific patch.

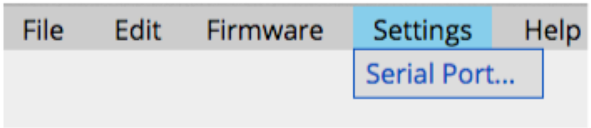
2. You can find further instructions about the MakerBit Smart Car, including how to use a second micro:bit as a remote control here: https://goo.gl/So2CfE

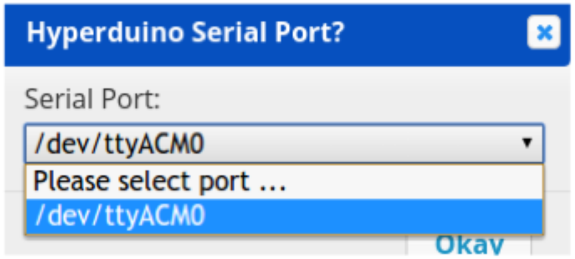
Restore MakerBit Program & Activity Area

R

1. Close the MakeCode browser window, and then open the MakerBit Media Linker.

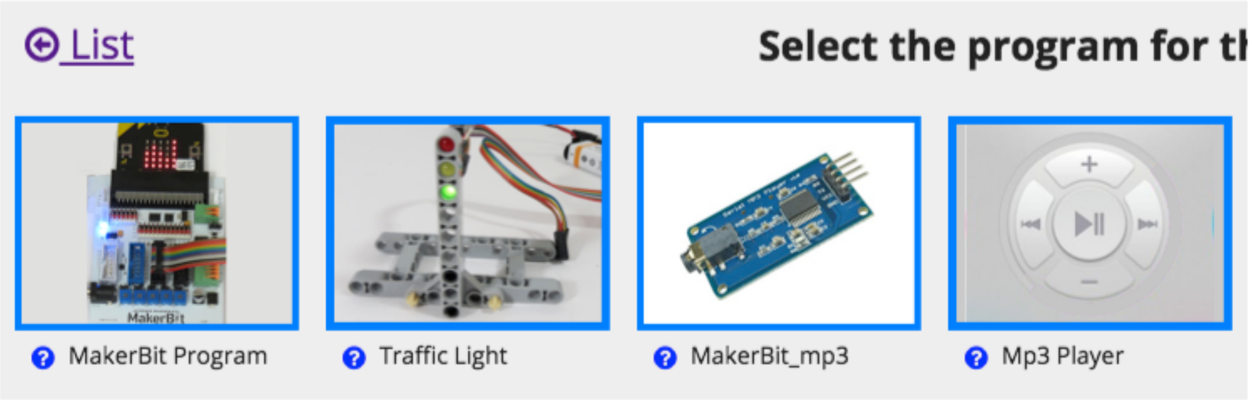
2. Choose Serial in Settings menu.




Hyperduino Serial Port?

Serial Port: /dev/ttyACM0

3. Choose MakerBit in the Programs gallery.



4. Clean up the activity area to put the materials back the way they were when you started.