BITPLAYER RADIO CONTROLLER

LEARNING TOPICS:

Logic Statements

If Then, Else Statements

Radio communications

Digital signals

Loop

MATERIALS:

Micro:bit

Bitplayer controller

CuteBot

Computer

Internet Access

INTRODUCTION:

In this lesson we will be creating a remote control for our cutebot!

Before starting with the activity, have students answer the introduction questions below:

- What do you know about radio communication
- What types of blocks do you think we might use today?
- Do you know what an if statement is?

VOCAB:

Next, let's discuss some Micro:bit and computer science terminology:

Program – An algorithm that has been coded into something that can be run by a machine.

If, then, else Statements – Compares two or more sets of data and test them for a result. If results are true, then do (what you make the program do if results are true), else do (what you make the program do if results are false).

Bug – Part of a program that does not work correctly.

Debugging – Finding and fixing errors in a program.

Logic – Logic is a programmers best friend! Logic is used to dictate the flow of code/execution

INSTRUCTIONAL CONTENT:

Let's get started! First, click or type the following link "https://makecode.microbit.org/" which will take you to today's activity on the Micro:bit website.
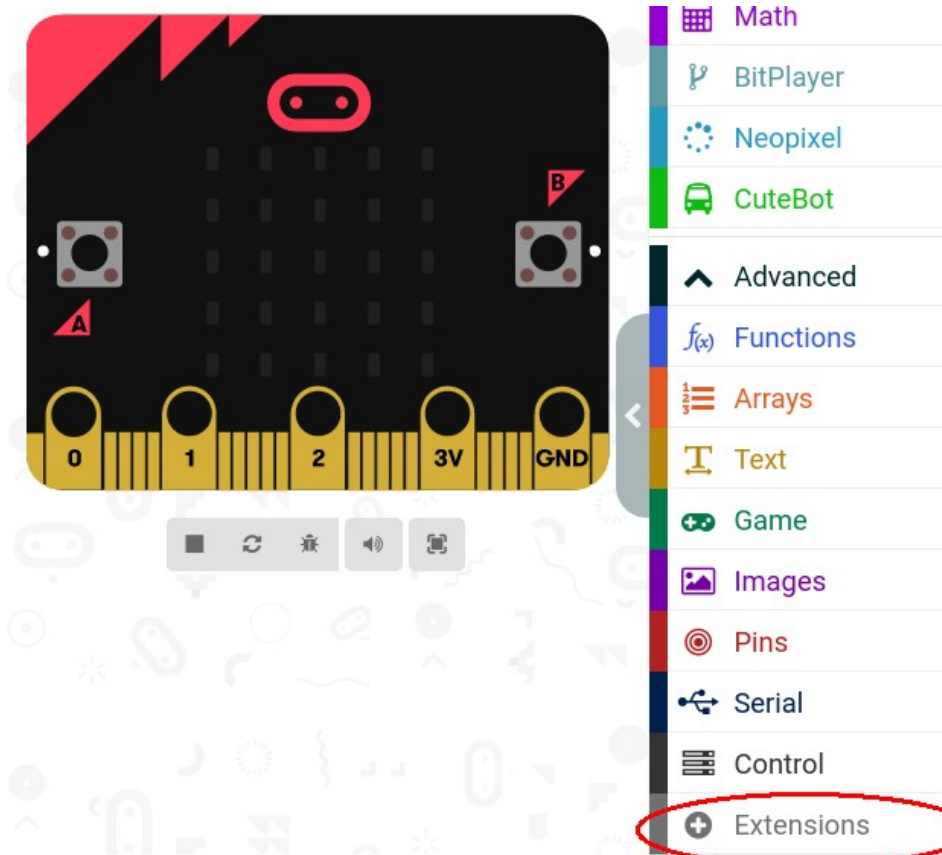
Review both the Micro:bits and the MakeCode tool with students if needed.

Once students have their MakeCode program loaded, have students click on "**New Project**". Name the project "BitPlayer Remote".
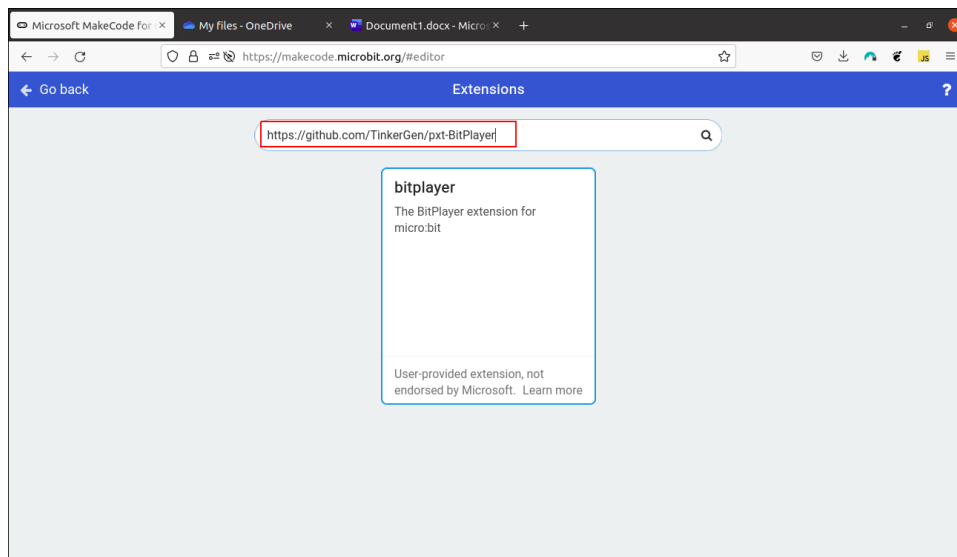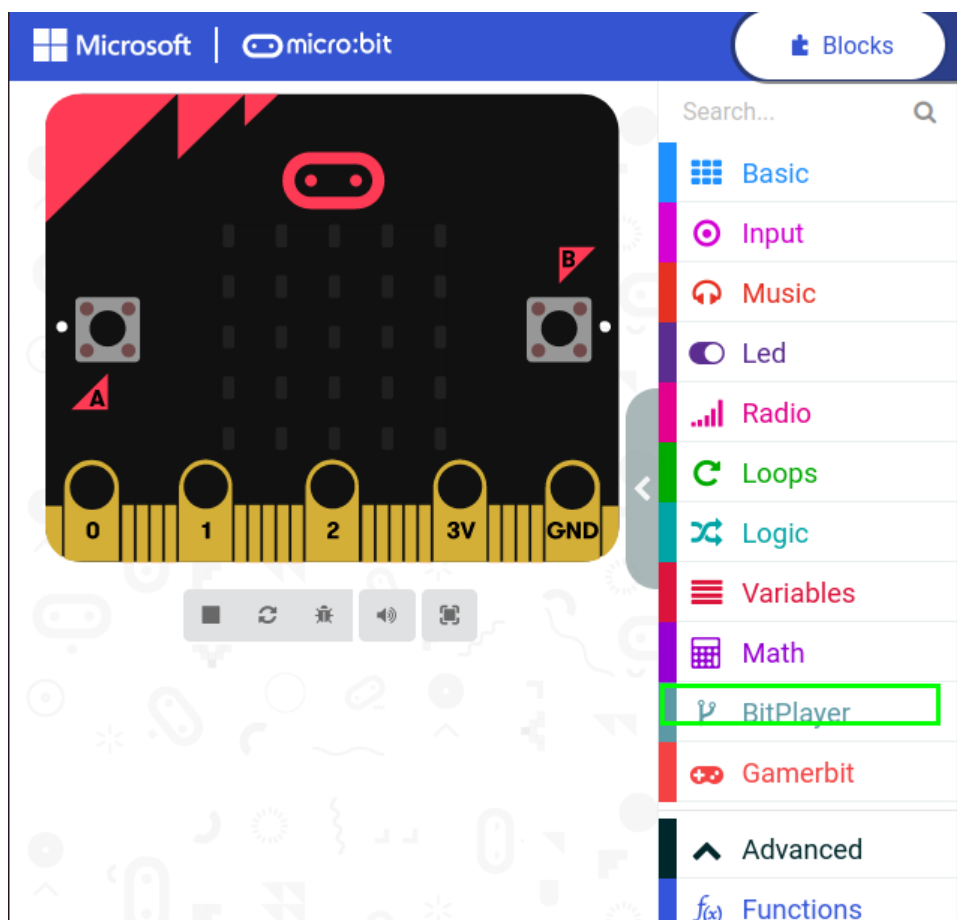
-

IMPORTANT STEP:

Navigate to the extensions section

Once in the navigation menu, paste the following URL in to the search bar

URL: https://github.com/TinkerGen/pxt-BitPlayer

After selecting this you will be returned to the makecode environment with a new type of code block available in the menu:

Now we can begin working on the code!

We're programming the commander for our cutebot! This means that our program will be somewhat simple/repetitive since we're simply sending 'keywords' from one microbit to the other in an effort to trigger certain actions!

Begin by setting your microbits radio group!
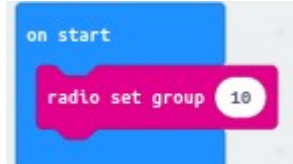
Make sure that this number matches in your car code and is <mark>different from your classmates!</mark>

Now that our microbit knows who it is talking to we can setup what command each button on our *bitplayer* will send. We will do this by using an 'if statement' which can be found in the logic tab
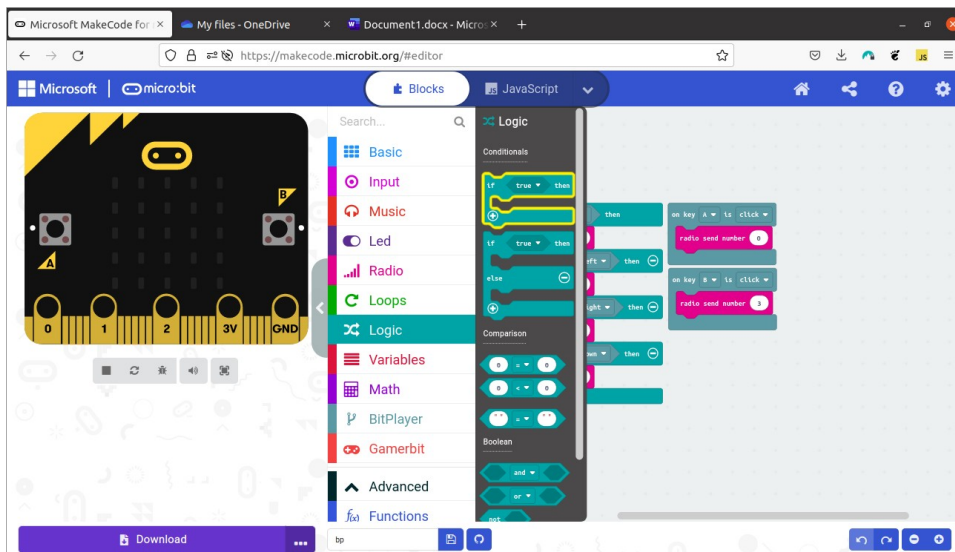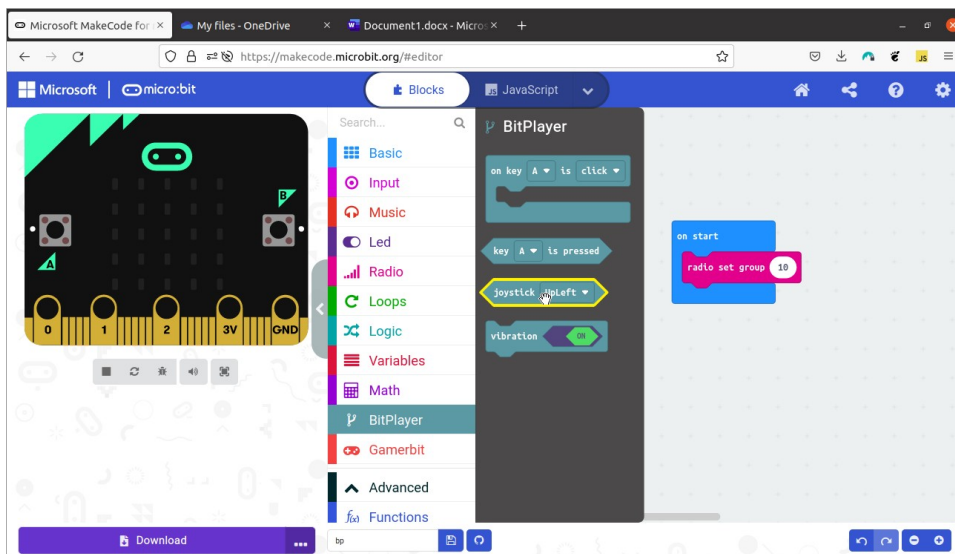

Once you have grabbed the if statement and placed it in the forever block your code should look as follows:
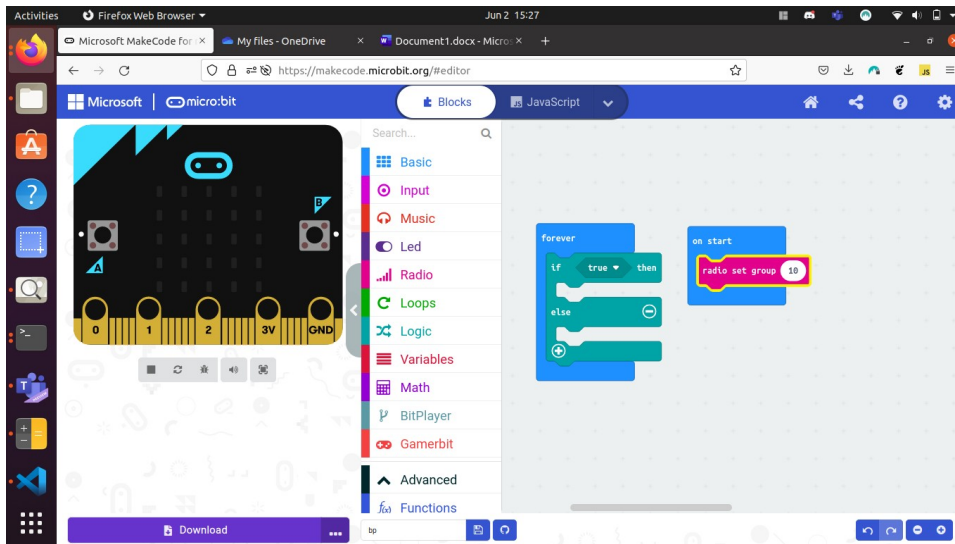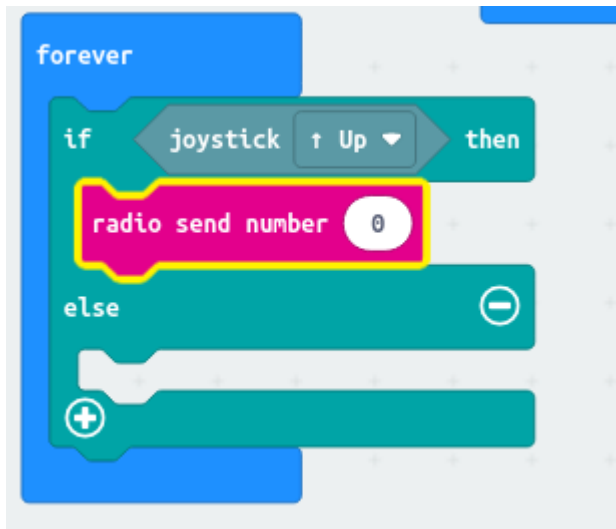
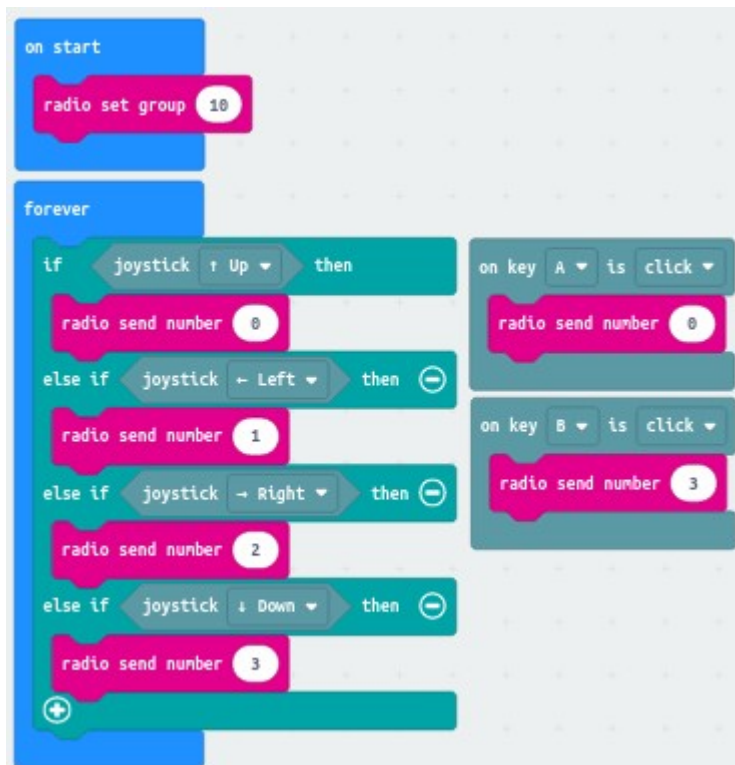We can now begin constructing logic around our bitplayer!

Open the bitplayer tab and grab a joystick block

Using these pieces you should now be able to construct the following code by snapping the joystick code into the if statement and grabbing a block from the radio tab!

Repeat this pattern until your code looks like this:

Note: <mark>Students may send different numbers but this must be mirrored in the receiving code in order to work properly also notice that buttons A and B send the same numbers as forward and back on the joystick, why do you think this might be?</mark>

LEARNING OUTCOMES:

By completing this activity, you will learn:

- Analog Values
- Logic Statements
- Loops

CLOSURE:

Reflect with students on the following questions:

- How could you modify the program to make it your own? What changes would you make?
- How do you think our program differs from real life car headlights?
- What other conditions might be useful to control the car/lights together.

REFERENCES:

Micro:bit Makecode. (n.d.). Retrieved September 16, 2021, from https://makecode.microbit.org/

ElekFreaks. (n.d.). Retrived May 18, 2022 from

https://www.elecfreaks.com