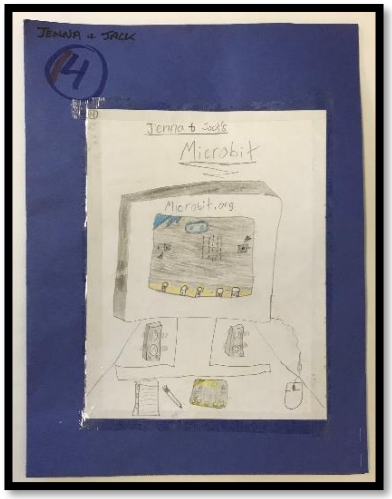
Micro:bits

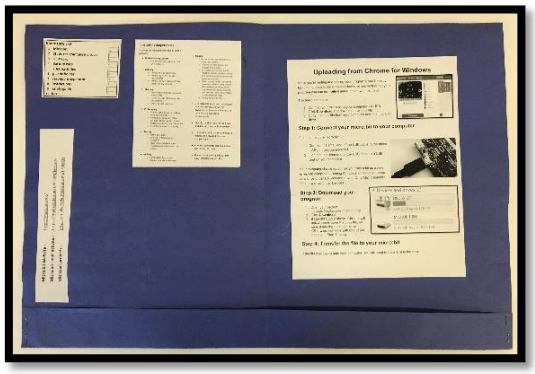
I was given a brand new kit to use with my class. I told my class that they were guinea pigs and we would be the test subjects working with the Micro:bits. They were given the challenge to product test them to see what worked, what didn't and how they could be used with a group of intermediate students. There were no lesson plans. I knew just as much as the students did, so if there was a problem, we would all need to figure it out. This is what we did as a class….



1. The students were put into pairs (someone they knew that they could work with) and the students made a folder to keep all their work in. This was just a large piece of construction paper, folded in half and then the bottom was folded upwards to create holding flaps. The students wanted to decorate it as well (optional).

2. The students were given **paperwork to glue into their folders** so that they could access it all the time. The pages that were included *(all attached)* were the…

    - A front cover page to decorate (optional)
    - Curricular Competencies
    - A list of items that are in the Micro:bit box.
    - A list of links that students need to access.
    - Instructions/Trouble Shooting Tips when plugging the Micro:bit into the computer to download the students' programmed .hex file.



3. The students were then given some other pages that they didn't glue into their folder.

    - Student activity list – basic challenge checklist, build project and paper project.
    - Student Reflection on the ADST Micro:bit Challenges

4. They were also asked to include some lined paper to write on when planning and brainstorming ideas.

5. The student pairs were given a number which matched the number on the Micro:bit box. That way the students were given the same Micro:bit each time.

Extra: One thing that we worked on as a class was taking the items out of the Micro:bit box and then putting them back into the box so that it all fits nicely. There is a way to pack it up so that the lid fits down and the flaps fold back in. This activity took some of my students quite awhile to figure out.

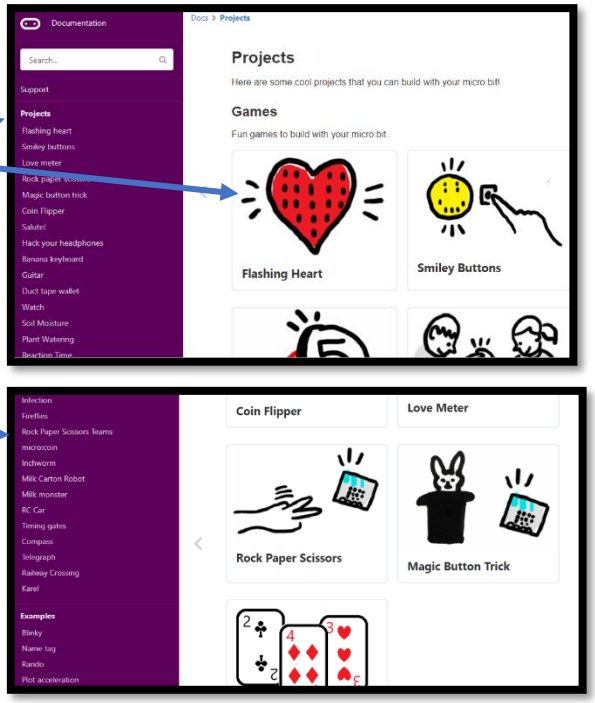
Micro:bits

The students were given three challenges using this link:

https://makecode.microbit.org/projects scroll down a little bit to see the bigger projects

a) **Mini Programming** – program the Micro:bit to do at least three of the programming activities that don't require building. The list of single (non-building) activities can be found here.
Students were more than welcome to develop more than three programming challenges.

b) **A Build Project** – programming the Micro:bit and building something out of "maker stuff" (cardboard, tinfoil, etc…). To find these projects just scroll down the page a bit.

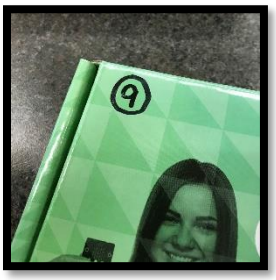


WARNING: Some projects (Milk Carton Monster, Inchworm) ask the students to cut wires or glue bits together which would damage or destroy the extra parts. In the case of the Guitar or the Watch, the students glued a small Zip-lock bag to the cardboard and placed the Micro:bit into the bag instead of directly gluing the Micro:bit to the cardboard. So, there are some ways to problem solve issues this could be another challenge for the students.

c) **An unplugged activity** – I chose a paper marble run that was made from nothing but paper and tape (attached). This unplugged activity was for the student teams when they were waiting their turn for some of the maker parts if the computers were down or the power was out.

I had them write a journal entry after each session – what went well, frustrations, thoughts about the days work…. this is great when the kid's computer doesn't work, Micro:bit doesn't load, they can explain what happened during the session. It was neat to see how their journal entries went from more frustrations in the beginning to writing about their successes at the end.

Planning Ahead



- The pairs of students were numbered and the Micro:bit with the same number is given to them. They would use this Micro:bit for each session.

- I encouraged the students to keep all the parts on their open folder so that the pieces don't get lost or fall between the desks in the lab.

- The students had the additional paperwork to fill in as they were planning and showing their thinking.

- As the students completed a simple task they would show me. The students would write the name of the Micro:bit activity. I would put my initials beside the activity when the Micro:bit did what it was supposed to. I would also talk to the students about the process and the programming of the Micro:bit.

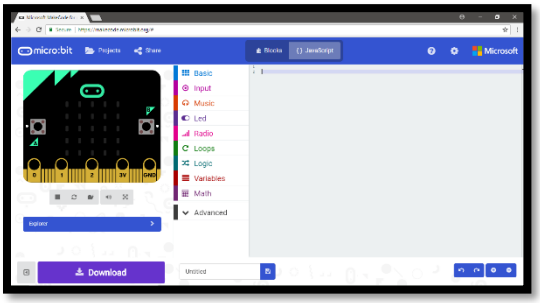
Into the Computer Lab

- When entering the lab, the students would sit in side by side with two computers between them. They can either share one computer, or they can each use a different one.

  - *I had many students share one computer and work in front of it.*
  - *I had other students working on one computer and using the other for the instructions. They followed the code and/or watched the videos on the second computer.*
  - *I had another set of students, when completing the simple, non-build programming activities, choose to each program a different activity. When they were finished, they loaded the hex file into their Micro:bit, showed me, and received the signature. Then they plugged the Micro:bit into the other computer to get a different hex file. They were programming twice as fast as other students.*

- Open the Micro:bit website

  https://makecode.microbit.org/#



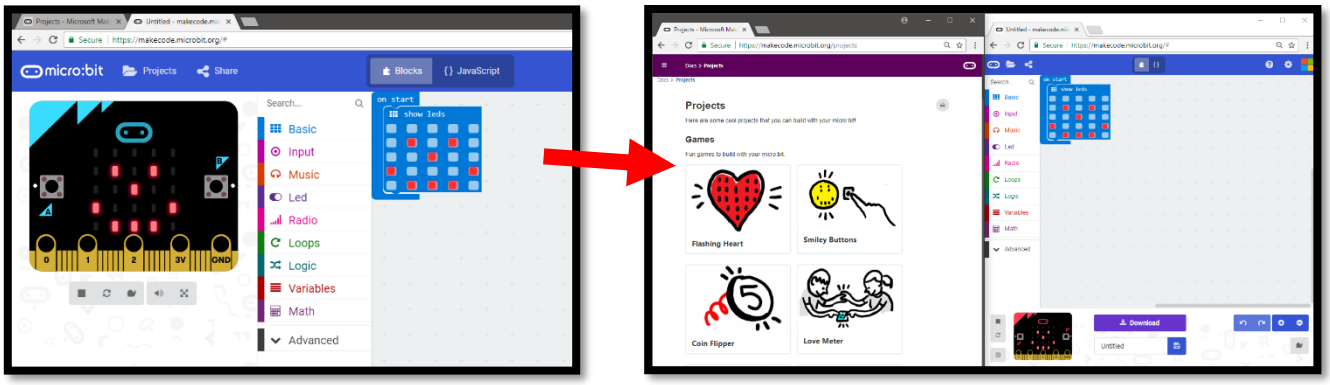
- This opens to the code making page where the students can start selecting their blocks to enter their code. GET CODING!!

Bonus Footage/Cheats

Split Screen

When coding a project, I showed the students how to do a split screen with the instructions on one side and the coding on the other.

- Open “Let’s Code” in one tab and the “Quick Links” in another tab.
- Pull one tab to the left and the other tab to the right to split the screen – instructions and coding

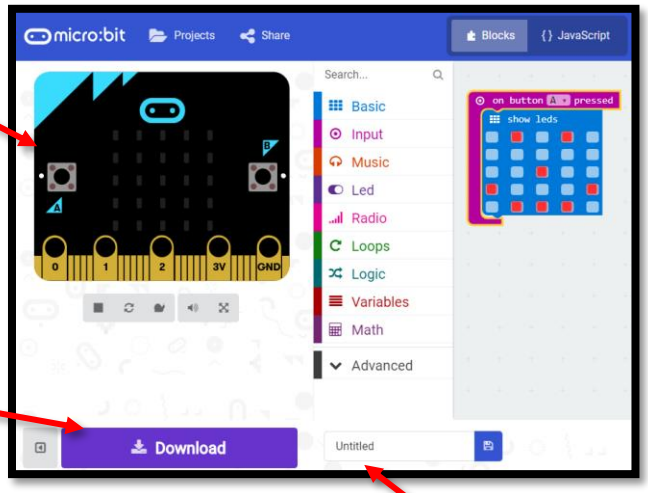


side by side.

- Students can then follow the coding instructions on the left and code on the right.

Testing the Code

- When the students have finished their coding, they can test it by hitting the buttons on the virtual Micro:bit.
- If there is an error in the coding, the Micro:bit will not respond correctly… coding corrections will need to be made.
- If the program tests correctly on the virtual Micro:bit, the students can download the .hex file to the Download folder.

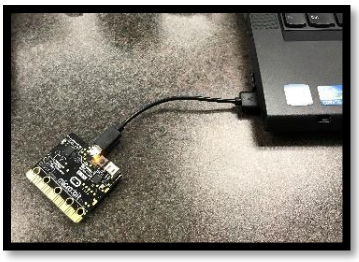


- Adding a name will allow the students to customize their work before downloading.
- When the download button is hit the .hex file will appear in the Download folder. It is ready to be dragged and dropped into the Micro:bit.

Plugging In

a) Students need to be reminded that they should plug the Micro:bit into the USB cord first, and then plug the USB cord into the computer.

b) Once it is plugged into the computer – **LEAVE IT**… it can take awhile for the Micro:bit to connect to the machine (20 seconds to 3 min).
   i. When the Micro:bit shows up as an extra drive on network called MICROBIT = success.
   ii. If it shows up as Unknown Drive = there is an error and you will need to unplug and try again. You may need to shut computer on and off. Then try again.



c) If it does not show up at all – after 5 minutes – unplug and shut computer on and off. Then try again.

d) When unplugging the Micro:bit from the computer – LEAVE the USB cord in the computer. If the USB cord is left in the computer, it will respond more quickly when you plug the Micro:bit back in.

e) Plugging and unplugging the Micro:bit from the USB port over and over will cause the computer to not recognize it. When this happens, the computer needs to be shut off and turned on again to clear it's cache.

Trouble Shooting

On Thin Clients

1. For best results, when using the Thin Clients, use the two USB outputs that are at the top of the image. The other two that are for the keyboard and mouse.

2. When plugging the Micro:bits into the computer with the USB cables, some Micro:bits take a very long time for the computer to recognize them (1 min up to 4min).

3. Repeatedly unplugging and re-plugging them back in makes the process even longer and will eventually cause the computer not to recognize it.

4. If the Micro:bit isn't recognized – **(M:Unknown HardDrive)** restarting the computer without the Micro:bit being plugged in did seem to help.

5. When the USB cord is plugged in, and the Micro:bit is plugged into the cord… Highly recommend leaving the USB cord plugged in and removing the Micro:bit from the USB cord. This will make the computer recognise the Micro:bit quickly.

Micro:bits

How to take the programming and the building to the next step – go from copying code to writing their own code

1. Video – spark fun electronics – working with Micro:bits - https://www.youtube.com/watch?v=kaNtg1HGXbY
2. Hackster.io – Intro for Teachers - https://www.youtube.com/watch?v=RkWDYTx_mg4

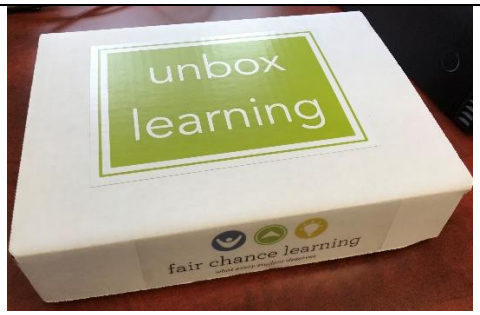
Coding – suggestions…

Sites

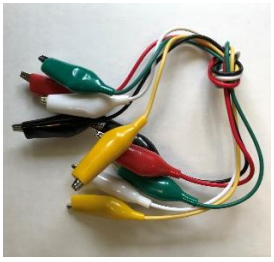
1.	**Micro:bits Site**	http://microbit.org/	Front page of the Micro:bit site.
2.	**Micro:bits Reference**	https://makecode.microbit.org/reference	Micro:bit block reference page. What do these blocks mean?
3.	**Discover JavaScript Blocks**	http://microbit.org/en/2017-03-07-javascript-block-resources	Quick step by step lessons.
4.	**Micro:bits Projects**	https://makecode.microbit.org/projects	Micro:bit projects page. Simpler project at the top and more complex as you scroll down.
5.	**Lets Code**	https://makecode.microbit.org/#	Direct to the Micro:bit coding platform.

Micro:bits

Along with the Micro:bits, an "Extras" box is included which is a collection of wires and parts that the Micro:bit can use to interact with it's environment.



Supply box has the following items in it (subject to change as Tech. dept. adds more)



Alligator Clips

Speaker

Motor w/attachments

Light Sensor

Ultrasonic Sensor

Jumpers

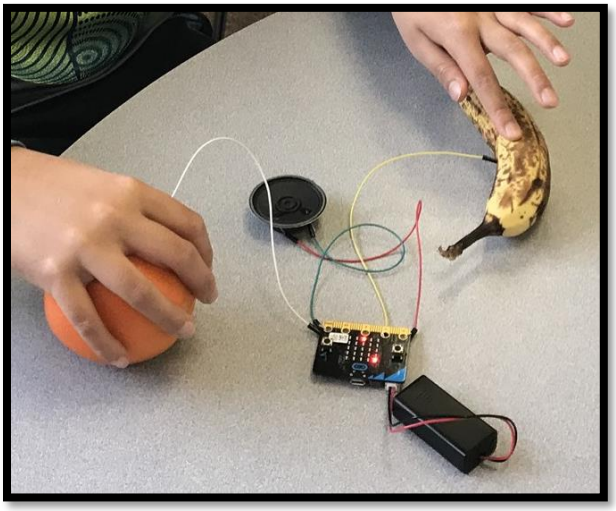
Diodes

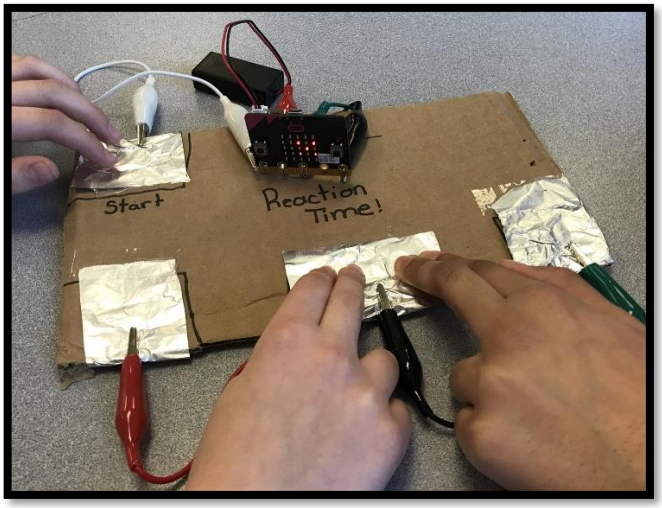
Breadboard

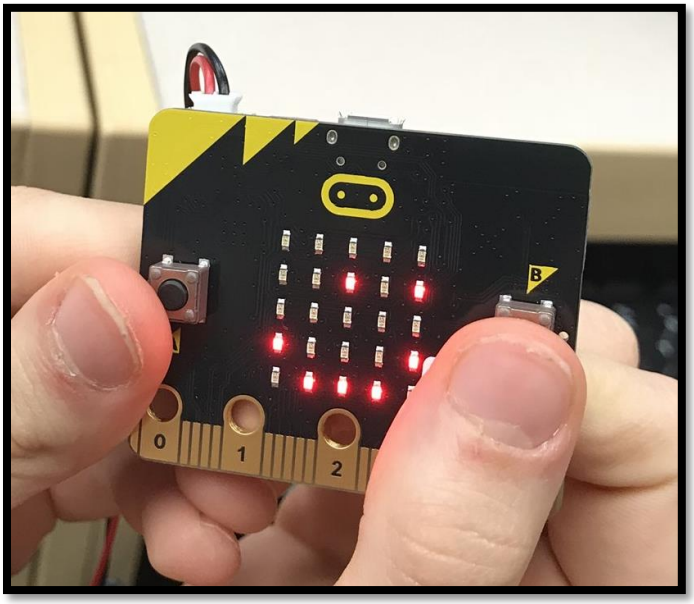
Alligator clips /audio jack

Battery pack w/batteries

Micro:bits





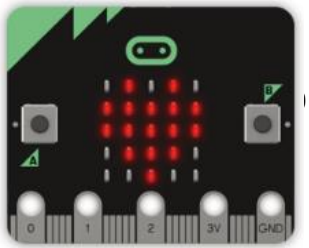


Micro:bits

Names: _____ _____ Number: _____

Simple Coding		“Maker Project”	Unplugged
Various Coding Challenges on the Micro:bit site		Build something that requires some of the extra components to attach to your Micro:bit.	Marble Run – build ramps out of paper to support a rolling marble length and height difference

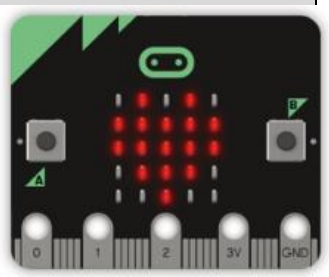
Student Reflection on the



ADST Micro:bit Challenges

Name: _____

	I could….	I needed help with this	I needed assistance occasionally	I could do this on my own
Defining	choose an activity to program and build.			
	outline a general plan, identifying tools and materials.			
Prototyping	code the necessary information correctly.			
	construct and program a design, changing plans, as needed.			
Testing	make changes and test again, repeating until satisfied.			
Making	construct the final product.			
Sharing	persist and not get frustrated things didn't quite work out.			
	demonstrate their product and describe their process.			
Teamwork	work effectively as an individual.			
	work effectively and collaboratively in a partnership.			



Curricular Competencies

Students are expected to be able to do the following:

- **Understanding context**
  - Gather information about or from potential users

- **Defining**
  - Choose a design opportunity
  - Identify key features or user requirements
  - Identify the main objective for the design and any constraints

- **Ideating**
  - Generate potential ideas and add to others' ideas
  - Screen ideas against the objective and constraints
  - Choose an idea to pursue

- **Prototyping**
  - Outline a general plan, identifying tools and materials
  - Construct a first version of the product, making changes to tools, materials, and procedures as needed
  - Record iterations of prototyping

- **Testing**
  - Test the product
  - Gather peer feedback and inspiration
  - Make changes and test again, repeating until satisfied with the product

- **Making**
  - Construct the final product, incorporating planned changes

- **Sharing**
  - Decide on how and with whom to share their product
  - Demonstrate their product and describe their process
  - Determine whether their product meets the objective and contributes to the individual, family, community, and/or environment
  - Reflect on their design thinking and processes, and their ability to work effectively both as individuals and collaboratively in a group, including their ability to share and maintain a co-operative work space
  - Identify new design issues

- Use materials, tools, and technologies in a safe manner, and with an awareness of the safety of others, in both physical and digital environments

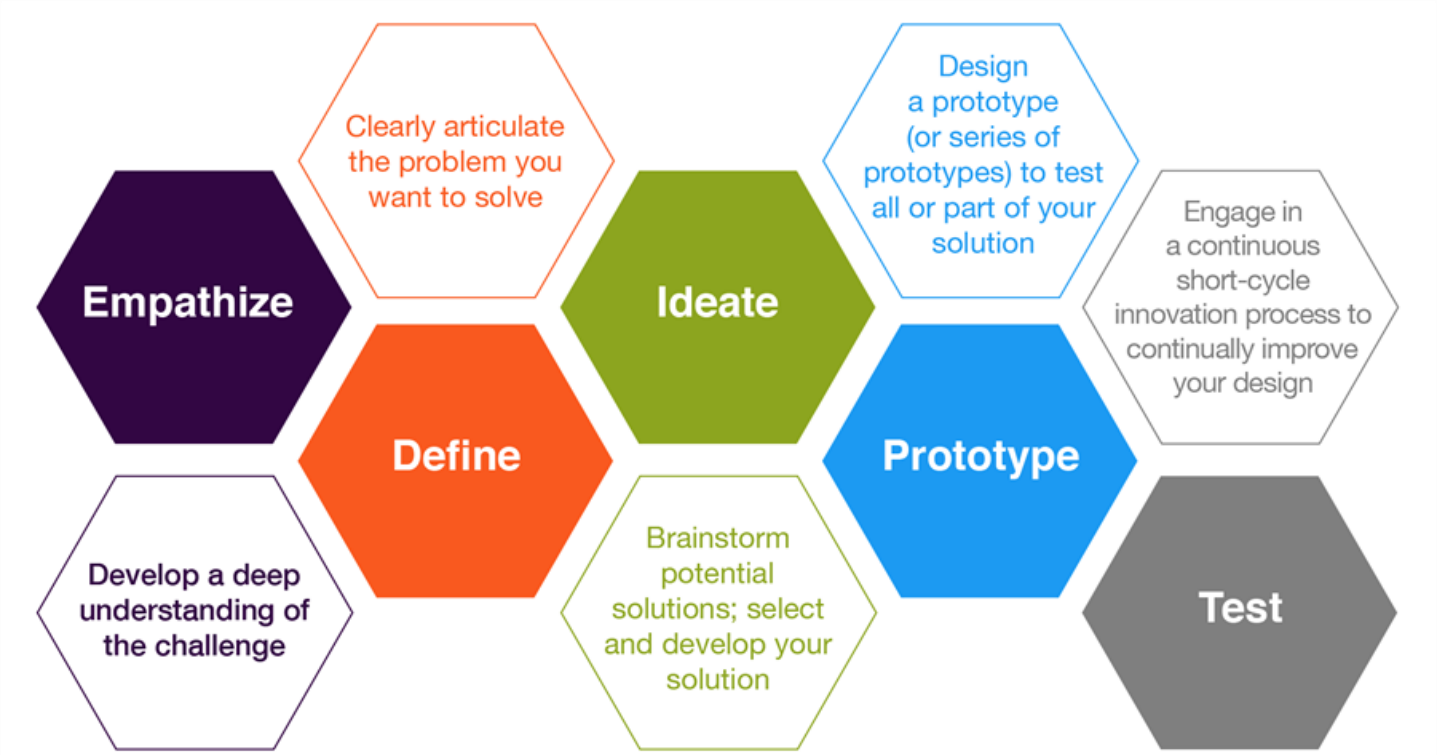
- Identify the skills required for a task and develop those skills as needed

- Use familiar tools and technologies to extend their capabilities when completing a task

- Choose appropriate technologies to use for specific tasks

- Demonstrate a willingness to learn new technologies as needed

Micro:bits


Empathize
Develop a deep understanding of the challenge
Define
Clearly articulate the problem you want to solve
Ideate
Brainstorm potential solutions; select and develop your solution
Prototype
Design a prototype (or series of prototypes) to test all or part of your solution
Test
Engage in a continuous short-cycle innovation process to continually improve your design

Uploading from Chrome for Windows

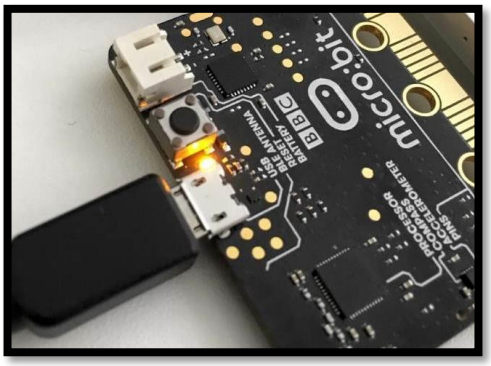
While you're writing and testing your programs, you'll mostly be running them in the simulator, but once you've finished your program you can **compile** it and run it on your micro:bit.

The basic steps are:

1. Connect your micro:bit to your computer via USB
2. Click **Download** and download the `.hex` file
3. Copy the `.hex` file from your computer onto the micro:bit drive

Step 1: Connect your micro:bit to your computer

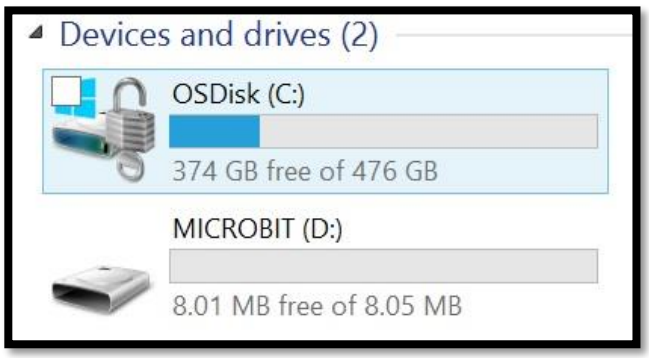
First, connect the micro:bit:



1. Connect the small end of the USB cable to the micro USB port on your micro:bit.
2. Connect the other end of the USB cable to a USB port on your computer.

Your computer should recognise your micro:bit as a new drive. On computers running Windows, `MICROBIT` appears as a drive under Devices and drives. On a Mac it appears as a new drive under Devices.

Step 3: Download your program



1. Open your project on https://makecode.microbit.org/
2. Click **Download**
3. If you did Step 2 above, Chrome will ask where to save the `.hex` file, so save it into the `MICROBIT` drive. Otherwise, continue with one of the options in Step 4 below.

Step 4: Transfer the file to your micro:bit

If the file was saved onto your computer, you will need to transfer it to the micro:bit.