SCIENCE NORTH SCIENCE NORD

Kinematics and The Micro:bit	Grade 11 and 12 - Physics	
Lesson Plan	Coding Tool	Micro:bit
	Cross-curricular	

Big Ideas	**Specific Expectations**
A1. demonstrate scientific investigation skills (related to both inquiry and research) in the four areas of skills (initiating and planning, performing and recording, analysing and interpreting, and communicating);	A1.5 conduct inquiries, controlling relevant variables, adapting or extending procedures as required, and using appropriate materials and equipment safely, accurately, and effectively, to collect observations and data B1.1 analyse, on the basis of research, a technology that applies concepts related to kinematics (e.g., devices used to measure speed in sports; rocket accelerators; motion-detecting sensors for security systems; speedometers in automobiles) [IP, PR, AI, C]

Description

In this hands-on STEM lesson, students will explore the principles of kinematics (displacement, velocity, and acceleration) through real-time data collection using the BBC micro:bit. Using the micro:bit's built-in accelerometer and the machine learning via platform micro:bit CreateAI, students will train models to recognize different types of motion.

Materials

- Micro:bit V2
- Materials to create a Microbit wrist mount
- Computer
- Graphing Tool
- https://createai.microbit.org/

Computational Thinking Skills

- Machine Learning
- Block Programming

Introduction

Kinematics is the branch of physics that describes how things move. It focuses on the motion of objects without worrying about what causes the motion (that's dynamics). In kinematics, we study:

- Displacement – how far an object has moved from its starting point and in what direction.
- Velocity – how fast an object is moving and in which direction.
- Acceleration – how quickly an object's velocity is changing.
- Time – how long the motion takes.

By measuring and analyzing these quantities, we can describe and predict how objects move in the real world. Like a car speeding up, a ball falling, or a student walking across the room.

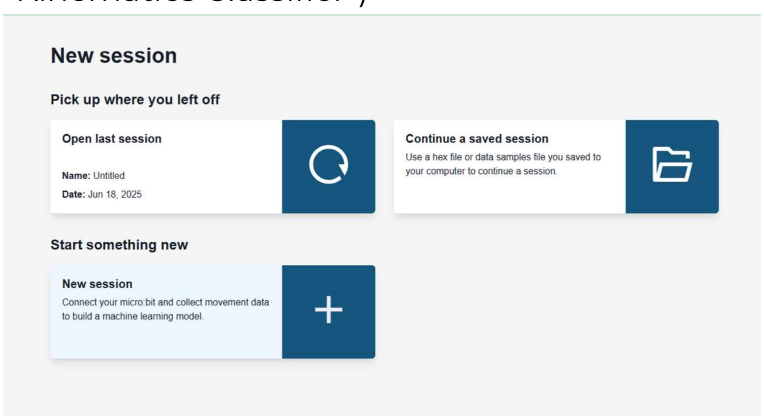
This lesson plan has students create models using machine learning of specific movements then track acceleration of those movements using the micro:bit V2's DataLogger extension.

Action

Step 1: Access CreateAI

Go to https://microbit.org/createai

Click "**Get Started**" then "**New Session**" and give it a name (e.g., "Kinematics Classifier")



Step 2: Connect Micro:bit

Click the connect button, make sure your micro:bit is attached via USB cable and follow the onscreen instructions on how to connect it to Bluetooth.

Step 3: Collect Motion Data

Create labels for each motion type you want to recognize (e.g., “walk,” “run,” “jump”).

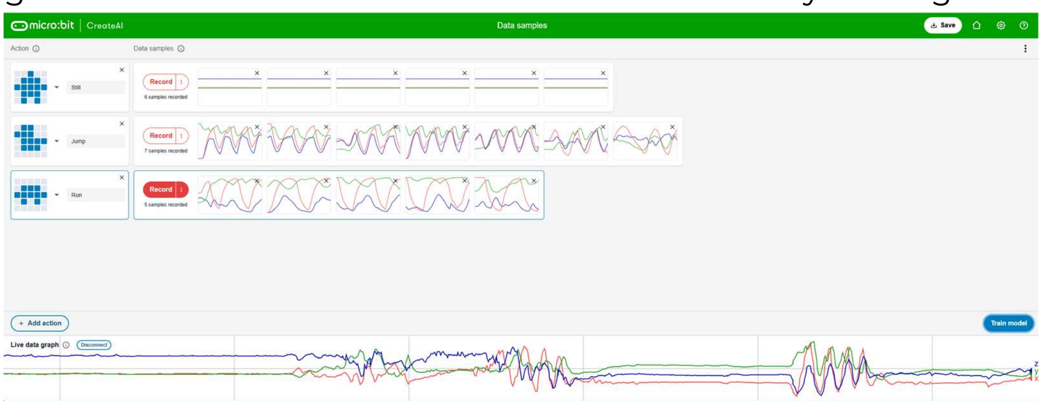
For each label:

Click “Record”

Perform the labelled motion while holding or wearing the micro:bit on a wrist strap.

Record multiple samples (aim for 10 or more) The more samples for each motion the more accurate the micro:bit will be at identifying to movement.

Notice how we’re training the mirco:bit how to recognize its movement by recording its position on the X, Y and Z axis. By learning it’s positions during certain movements it can remember to recognize those coordinates as those specific movements. Make sure to record the micro:bit in a STILL state to allow you to code your micro:bits to not try and guess movements when the micro:bit isn't actually moving.



Step 4: Train the Model

Once you’ve collected enough data, click “Train Model”

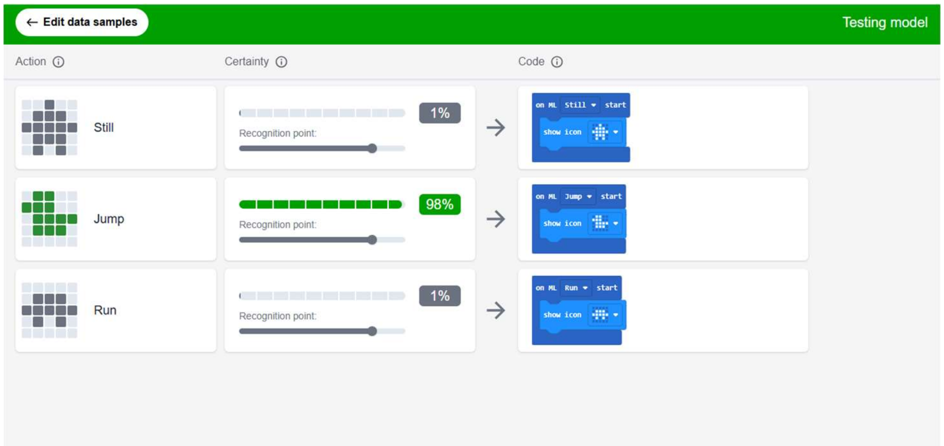
The system will analyze the data and build a classifier

Review the accuracy score and confusion matrix to see how well the model performs by reproducing those movements you trained on the previous screen.

Step 5: Test the Model

Click “Test” and try performing the motions again

Watch the live predictions to see if the model correctly identifies the motion by verifying the percentile accuracy of the movements while when repeated on this screen.



Step 6: Send to Microbit

Once you click on “Edit in Makecode” you’ll be brought to the coding page.

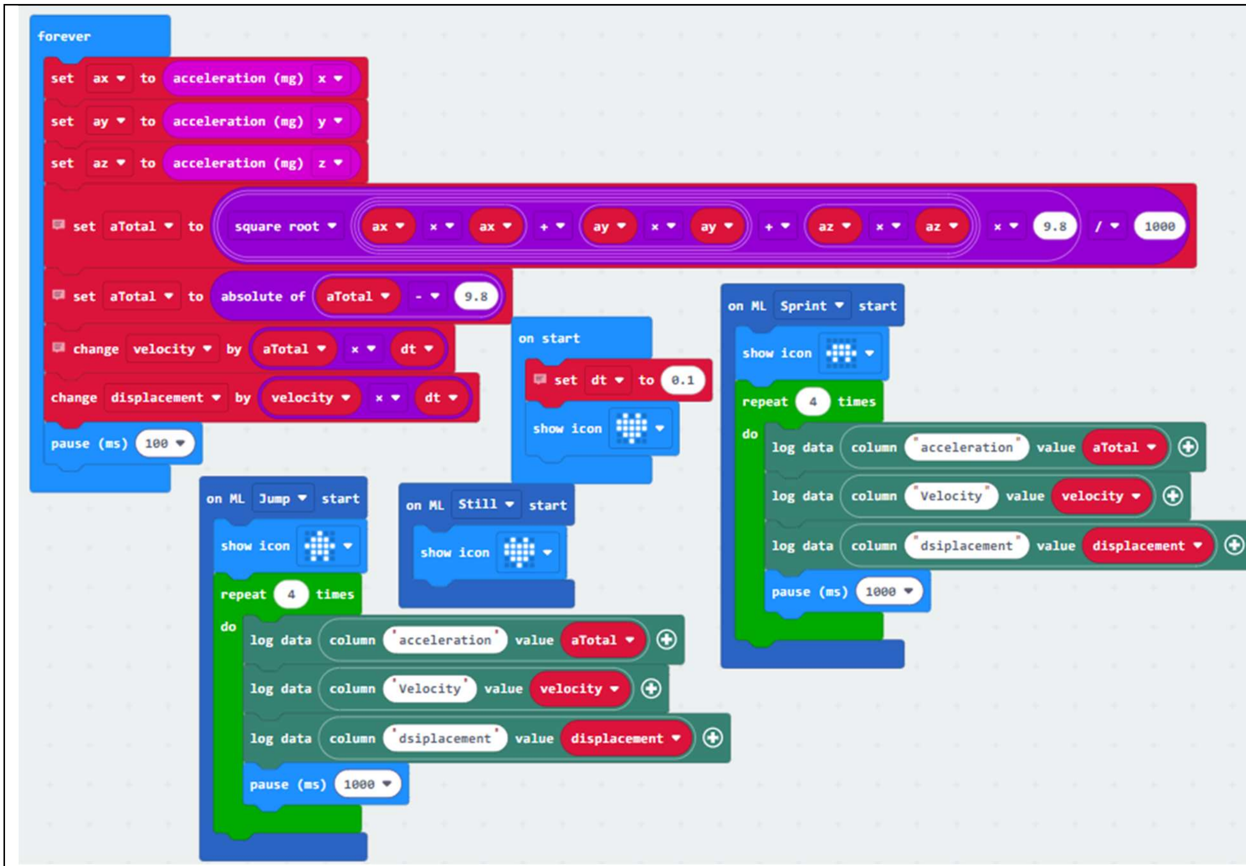
You can see blocks have been populated for your three movements. These blocks are found in then “Machine Learning” tab on the right. And code you add to these blocks will activate.

Step 7: Code your Microbit

Next we’ll add the data logger Extension by going to Extensions and searching “DataLogger”

This Extension allows you to save and record data from the sensors of the microbit.

We’re going to use the DataLogger blocks to keep track and calculate the displacement of the microbit while preforming the machine learned movements. **To do this we’ll use the code bellow:**



Or you can copy paste this code into the JavaScript:

```
ml.onStart(ml.event.Sprint, function () {
    basic.showIcon(IconNames.Tortoise)
    for (let index = 0; index < 4; index++) {
        datalogger.log(datalogger.createCV("acceleration", aTotal))
        datalogger.log(datalogger.createCV("Velocity", velocity))
        datalogger.log(datalogger.createCV("dsiplacement",
displacement))
        basic.pause(1000)
    }
})
ml.onStart(ml.event.Still, function () {
```

```
    basic.showIcon(IconNames.Heart)
})
ml.onStart(ml.event.Jump, function () {
    basic.showIcon(IconNames.House)
    for (let index = 0; index < 4; index++) {
        datalogger.log(datalogger.createCV("acceleration", aTotal))
        datalogger.log(datalogger.createCV("Velocity", velocity))
        datalogger.log(datalogger.createCV("dsiplacement",
displacement))
        basic.pause(1000)
    }
})
let az = 0
let ay = 0
let ax = 0
let displacement = 0
let velocity = 0
let aTotal = 0
// time step in seconds (1 second)
let dt = 0.1
basic.showIcon(IconNames.Heart)
basic.forever(function () {
    ax = input.acceleration(Dimension.X)
    ay = input.acceleration(Dimension.Y)
    az = input.acceleration(Dimension.Z)
    // Convert from milli-g to m/s² (1 g ≈ 9.8 m/s²)
```

```
    aTotal = Math.sqrt(ax * ax + ay * ay + az * az) * 9.8 / 1000
    // Subtract gravity (approximate)
    aTotal = Math.abs(aTotal - 9.8)
    // Estimate velocity and displacement
    velocity += aTotal * dt
    displacement += velocity * dt
    basic.pause(100)
})
```

What this code does:

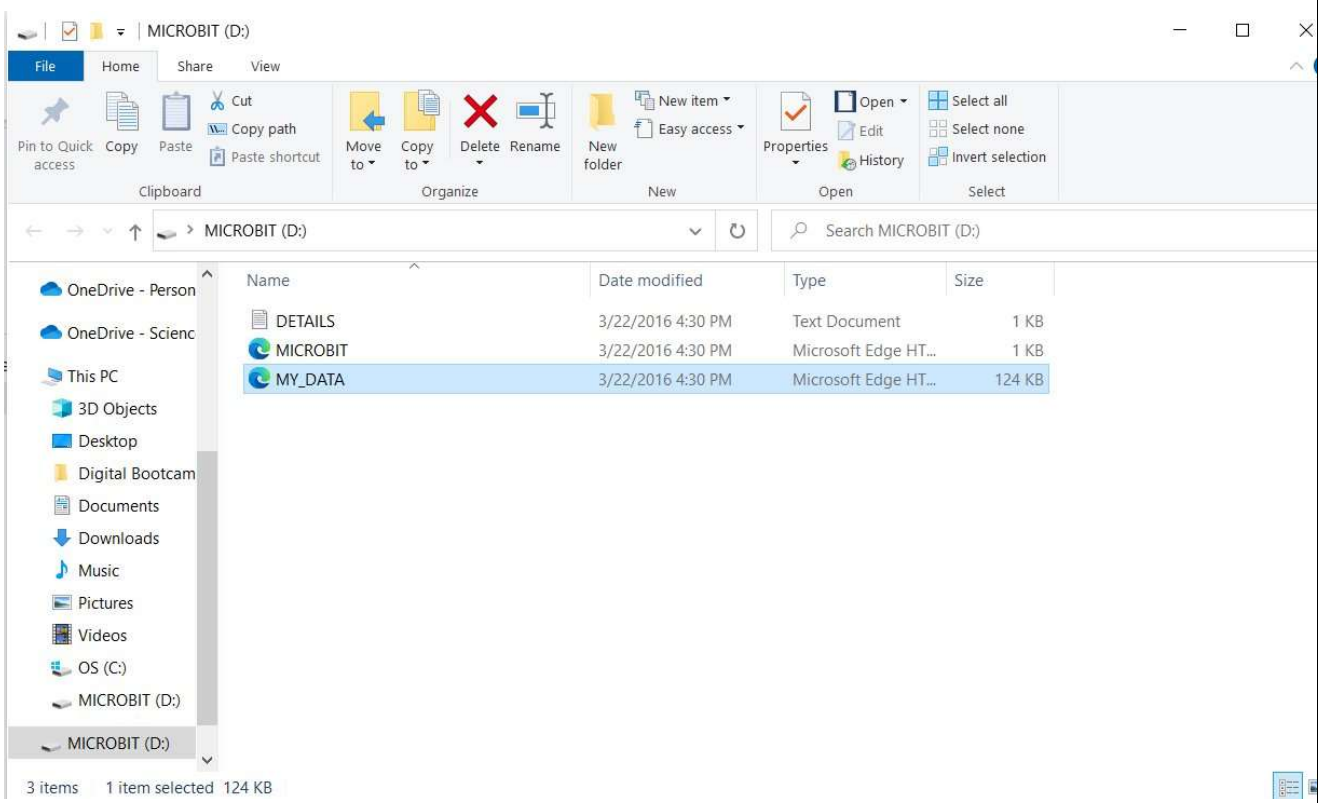
- Waits for machine learned action to occur
- Measures total acceleration from all 3 axes for 4 seconds
- Converts it to m/s^2 and removes gravity
- Estimates velocity and displacement using basic integration
- Saves data to the microbit for graphing or logging

Step 8:

Click "Download" to pair your Microbit to get the trained model as a .hex file

Plug in your micro:bit via USB
Drag and drop the .hex file onto the micro:bit drive
Once flashed, the micro:bit will start recognizing motions in real time!

Step 9:
You can download the dataLog from the microbits by reattaching them to the computer and opening them in your file explorer. Click on MY_DATA



Once you've clicking on my data the data log will open. That data log can be downloaded and opened in excel or sheets.

Your student can then fill in their worksheets, graph their findings and answer their reflection questions.

Consolidation/Extension

Students can now use this code and microbit to explore the kinematics of different objects in motion by attaching the microbit with this code and using machine learning to teach their microbits how the objects are expected to move.

Assessment The worksheet provided with reflection questions and screenshots of their working code would be excellent materials for assessment.
Additional Resources https://createai.microbit.org/ https://schools.sciencenorth.ca/educator-resources https://microbit.org/