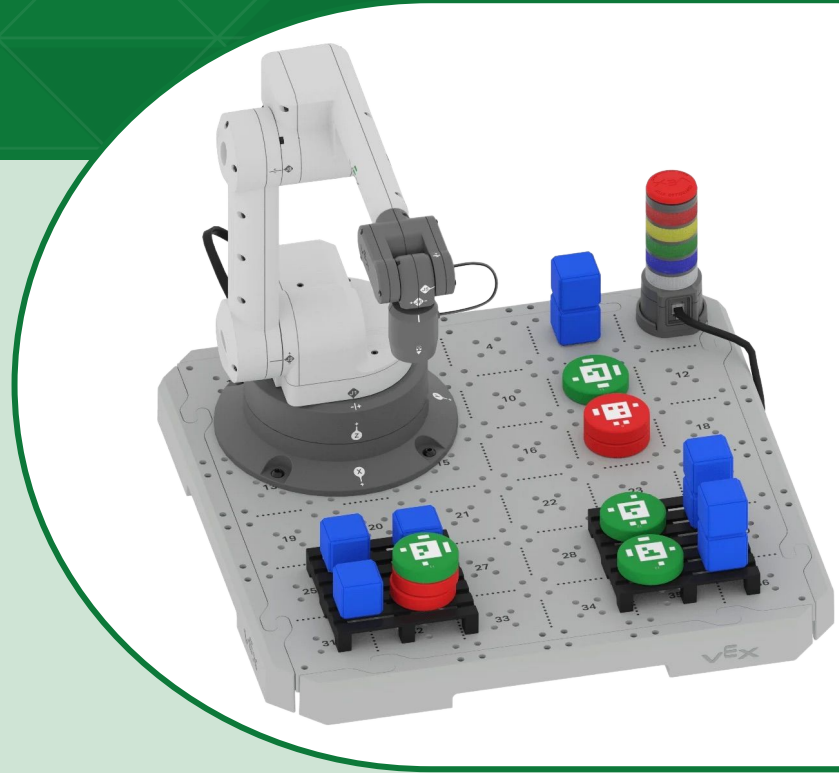


# Using the AI Vision Sensor with VEX CTE

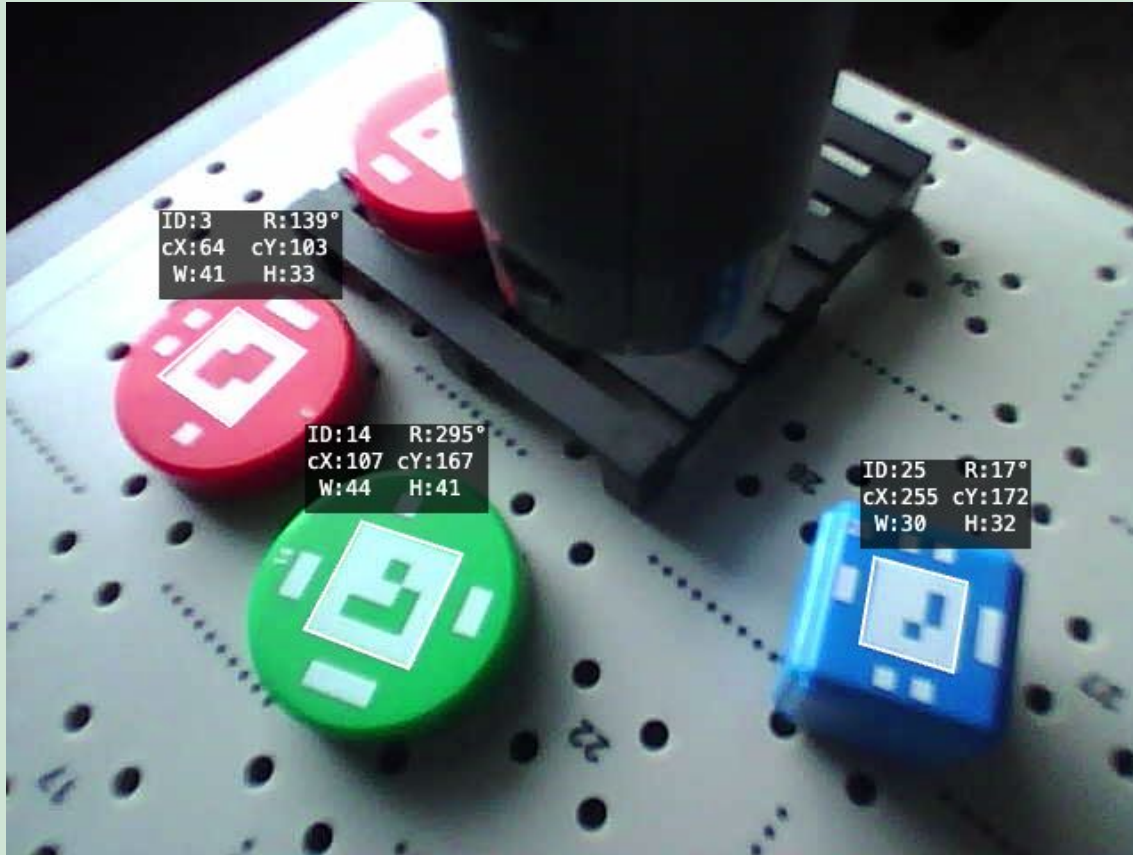
Lauren Oberst, Ed.D  
Educational Technology Consultant, VEX Robotics



# Workshop Goals

- Explain what the AI Vision Sensor sees and how the CTE arm moves
- Use AI Vision Sensor data to guide arm movement
- Build and run a working CTE project using the AI Vision Sensor

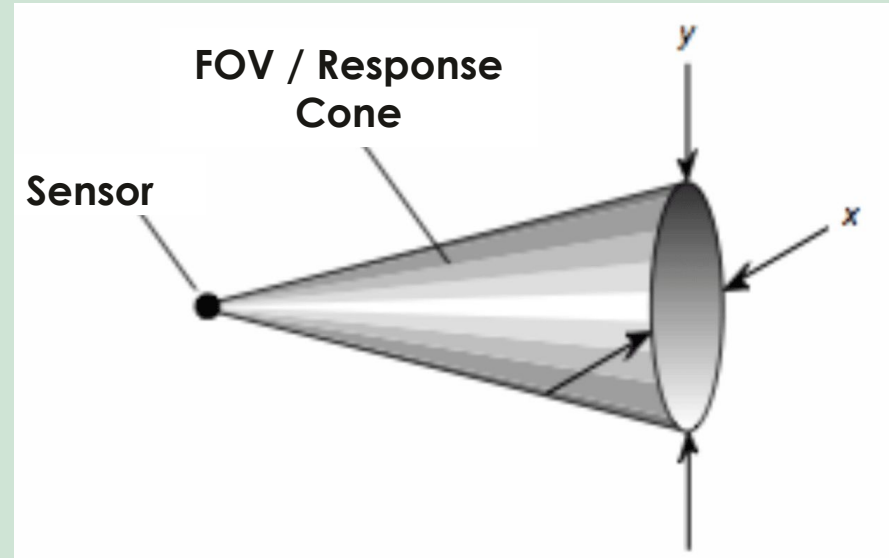
# AI Vision Sensor



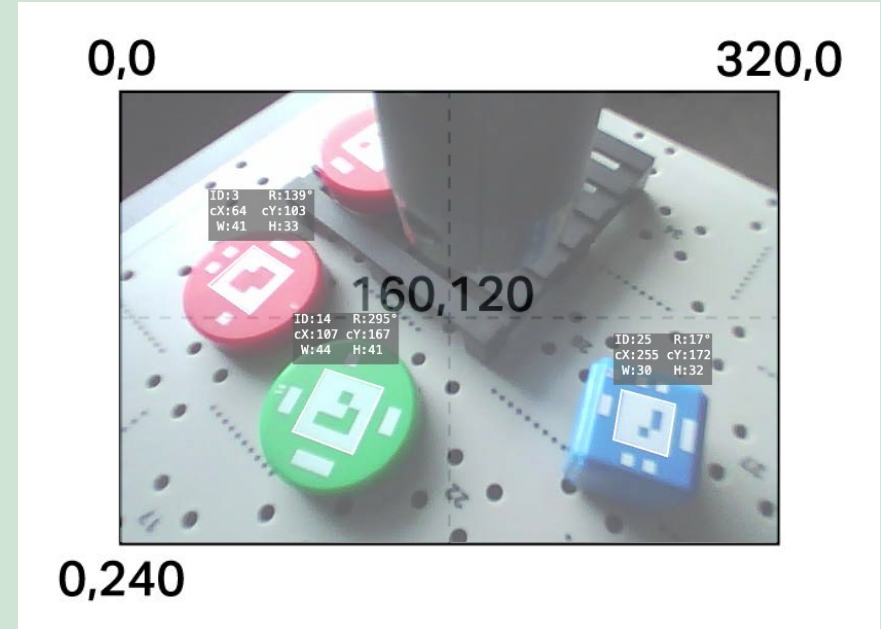
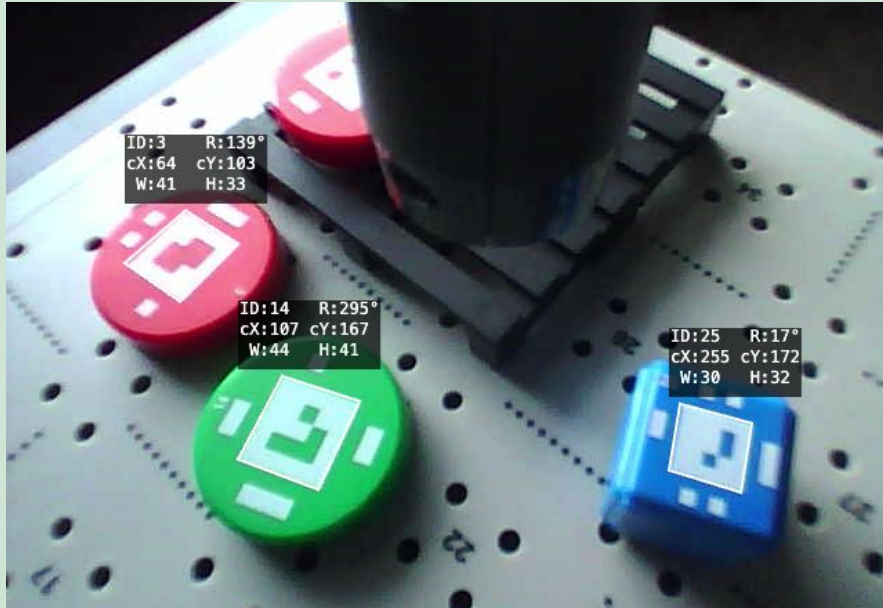
# AI Vision Sensor & Privacy

- The model runs entirely on the sensor
- No data is sent or stored externally
- Supports student data privacy (FERPA, GDPR)

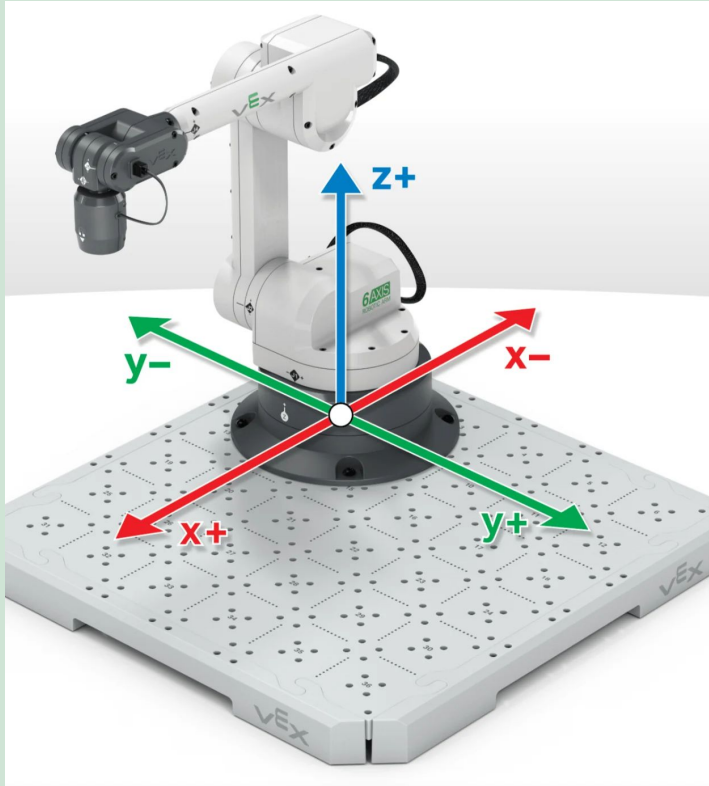
# Field of View



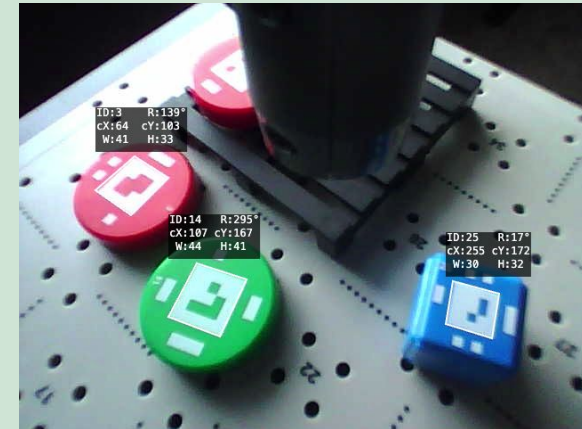
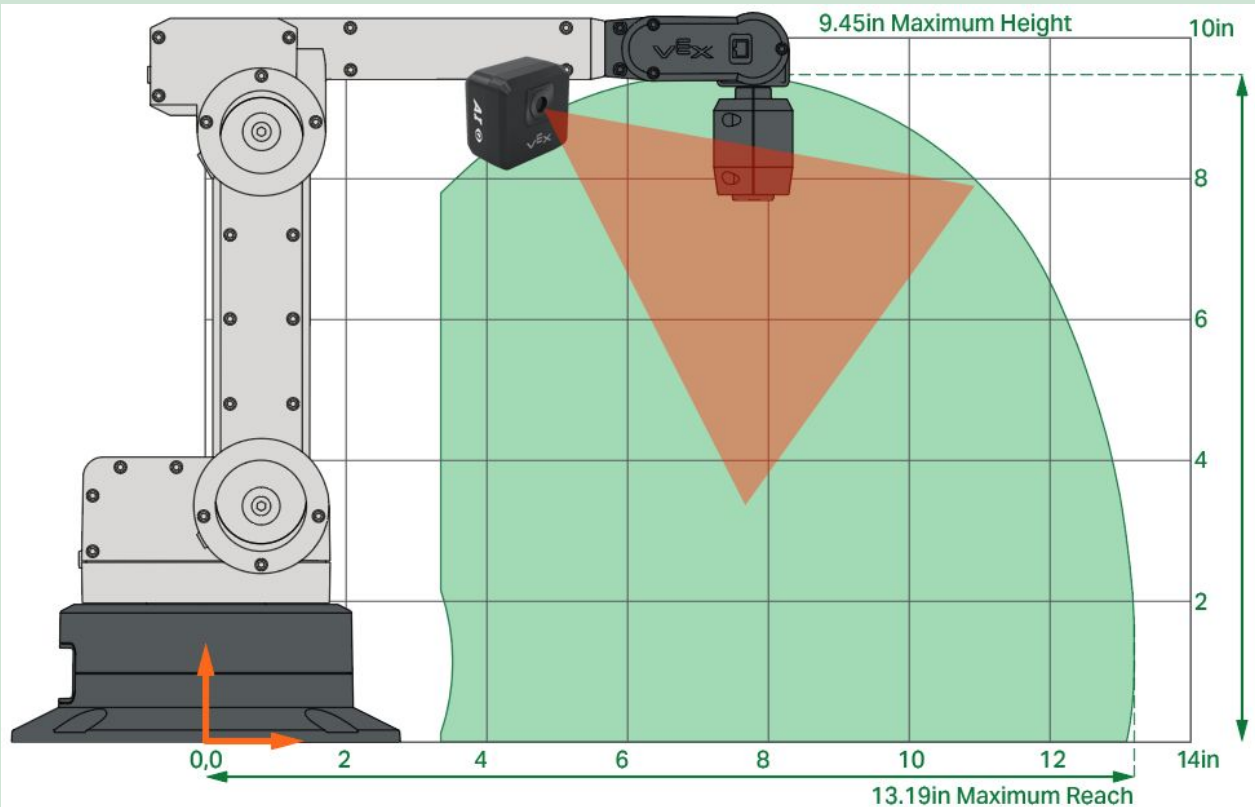
# Seeing in Pixels



# Arm Movement in 3D Space



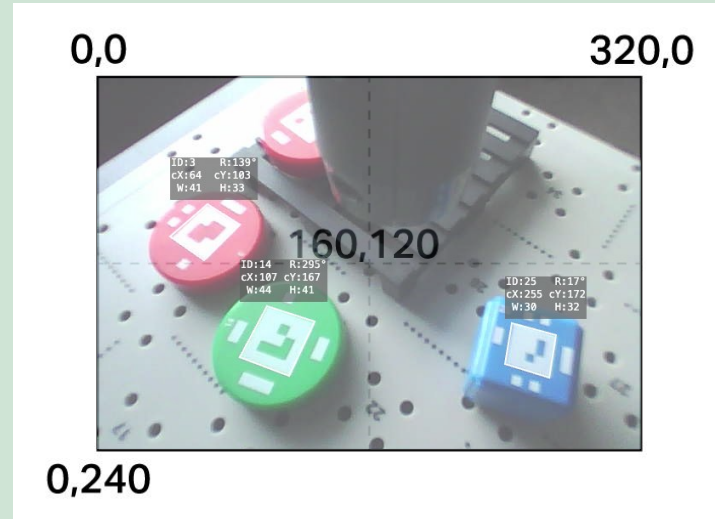
# Data in Different Spaces



# Bridging the Gap

If we can't directly convert pixels to millimeters, how do we use the data?

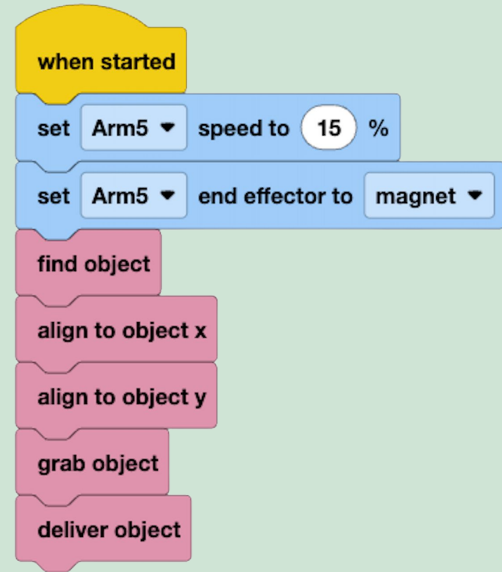
- Look at where the object is (CX, CY)
- Compare it to the center of the AI Vision Sensor's view
- Adjust the arm step-by-step until the object is centered



# Code Overview

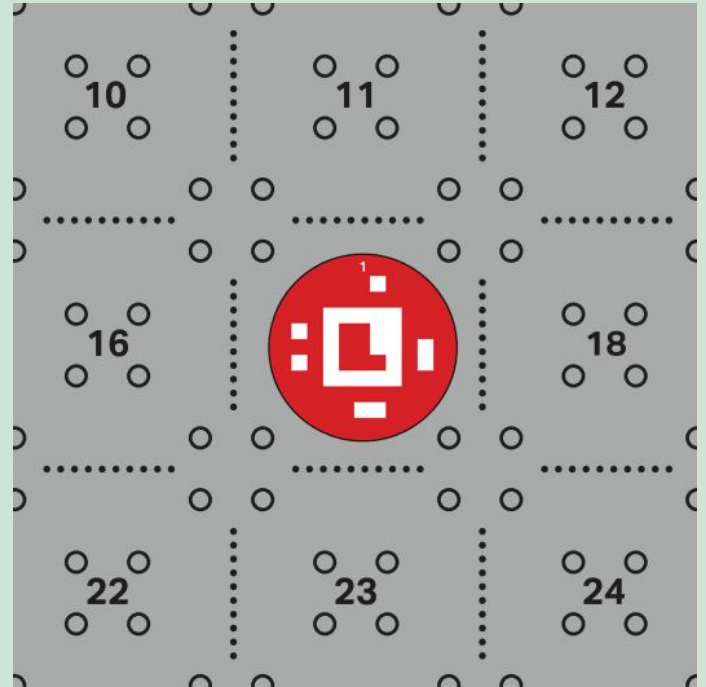
We're going to build the code in phases:

- **Phase 1:** Scan until we find an object
- **Phase 2:** Align the object along the x-axis
- **Phase 3:** Align the object along the y-axis
- **Phase 4:** Pick up the object
- **Phase 5:** Deliver the object



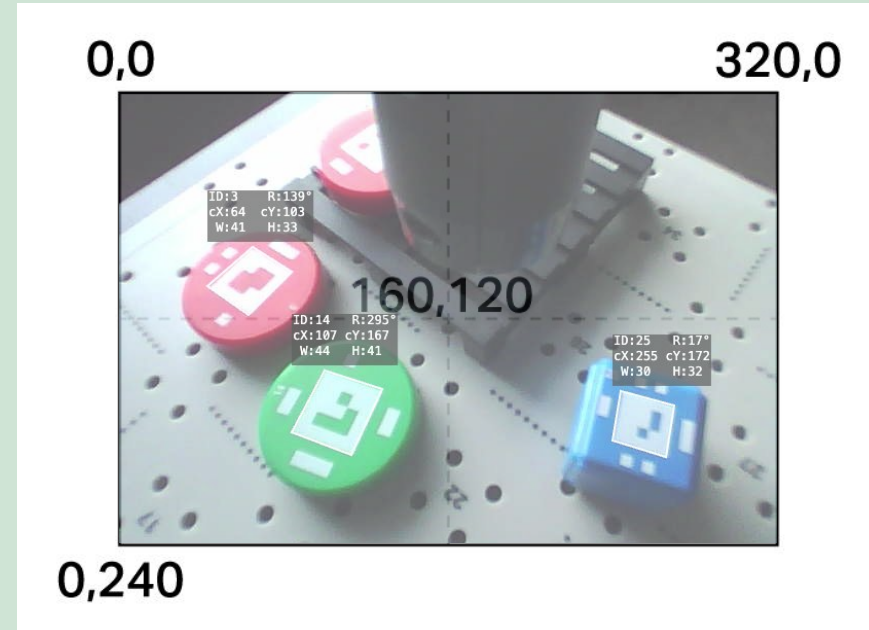
# Phase 1: Scan to Find Object

- Open the **Phase 1** example project.
- View the arm's (x, y, z) coordinates using the **Devices** screen.
- Place a disk on tile position 17 before testing.



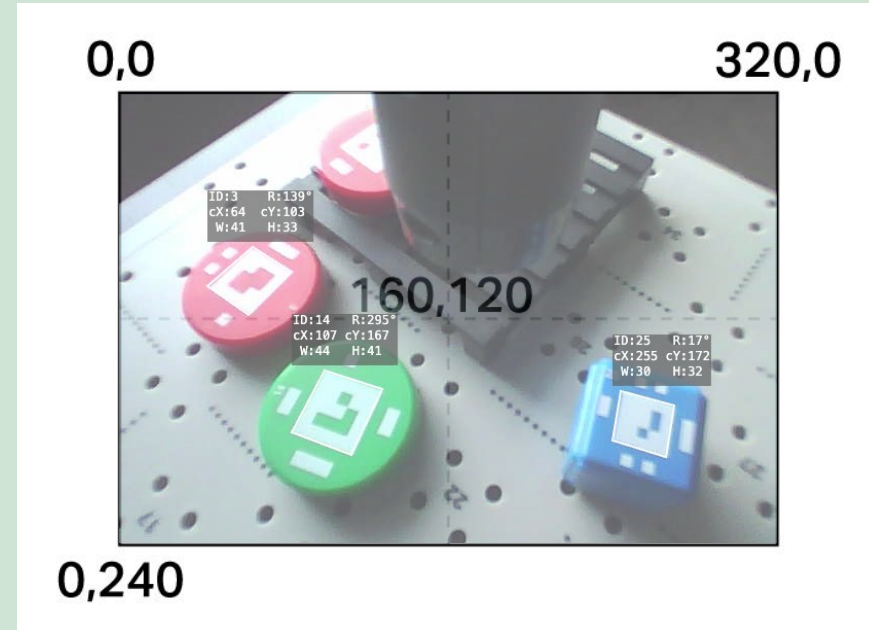
# Phase 2: Align to Object X

- Open the **Phase 2** example project.
- Fill in the values for centerX and how much the arm will increment while moving.
- Place a disk on tile position 17 before testing.



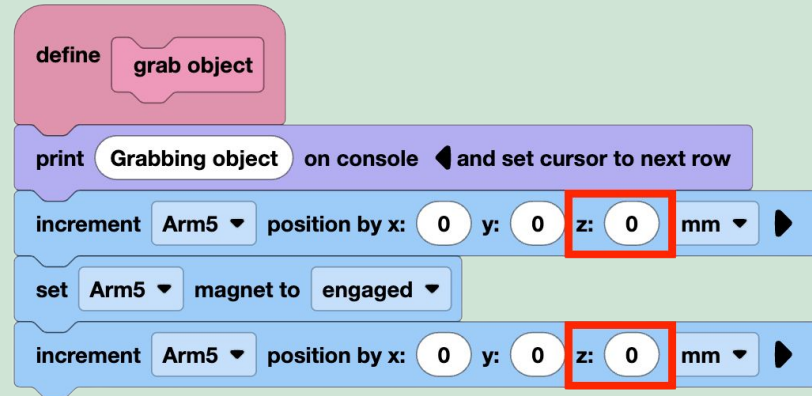
# Phase 3: Align to Object Y

- Open the **Phase 3** example project.
- Fill in the values for centerY and how much the arm will increment while moving.
- Place a disk on tile position 17 before testing.



# Phase 4: Grab Object

- Open the **Phase 4** example project.
- Fill in the values for how the z-coordinate should change to pick up the disk.
- Place a disk on tile position 17 before testing.

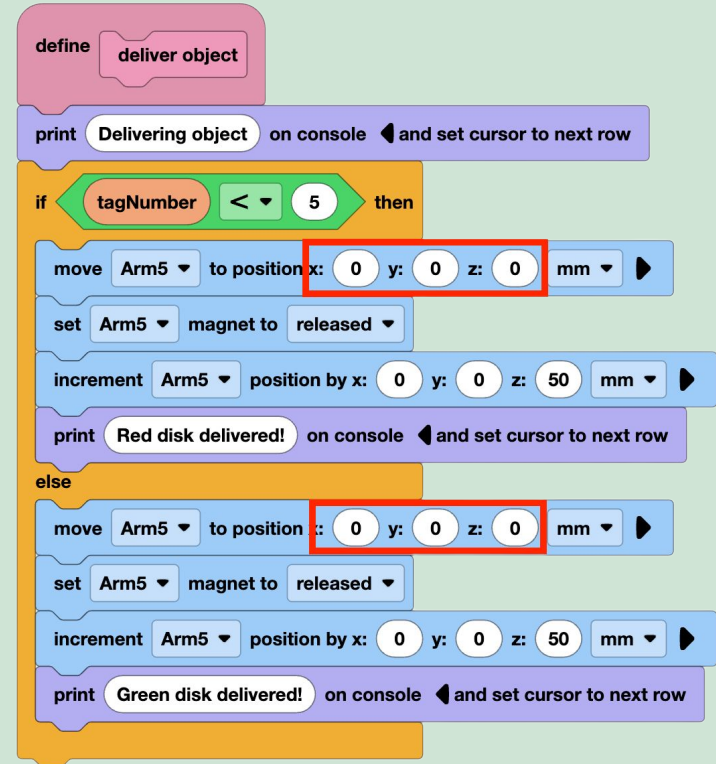


```
define grab object
  print Grabbing object on console and set cursor to next row
  increment Arm5 position by x: 0 y: 0 z: 0 mm
  set Arm5 magnet to engaged
  increment Arm5 position by x: 0 y: 0 z: 0 mm
```

The image shows a Scratch code editor snippet for a function named 'grab object'. The function consists of four code blocks: a 'print' block, an 'increment' block, a 'set' block, and another 'increment' block. The 'z' coordinate values in both 'increment' blocks are highlighted with red boxes.

# Phase 5: Deliver Object

- Open the **Phase 5** example project.
- Fill in the (x, y, z) values for where to deliver a red and green disk.
- Place a disk on tile position 17 before testing.



```
define deliver object
  print Delivering object on console and set cursor to next row
  if tagNumber < 5 then
    move Arm5 to position x: 0 y: 0 z: 0 mm
    set Arm5 magnet to released
    increment Arm5 position by x: 0 y: 0 z: 50 mm
    print Red disk delivered! on console and set cursor to next row
  else
    move Arm5 to position x: 0 y: 0 z: 0 mm
    set Arm5 magnet to released
    increment Arm5 position by x: 0 y: 0 z: 50 mm
    print Green disk delivered! on console and set cursor to next row
```

The image shows a Scratch script for a 'deliver object' function. The script is contained within a 'define' block. It starts with a 'print' block that says 'Delivering object' on the console and sets the cursor to the next row. This is followed by an 'if' block that checks if 'tagNumber' is less than 5. If true, it executes a sequence of blocks: 'move Arm5 to position x: 0 y: 0 z: 0 mm', 'set Arm5 magnet to released', 'increment Arm5 position by x: 0 y: 0 z: 50 mm', and 'print Red disk delivered!' on the console and set the cursor to the next row. If the 'if' condition is false, it goes to the 'else' block, which executes: 'move Arm5 to position x: 0 y: 0 z: 0 mm', 'set Arm5 magnet to released', 'increment Arm5 position by x: 0 y: 0 z: 50 mm', and 'print Green disk delivered!' on the console and set the cursor to the next row. The 'x: 0 y: 0 z: 0' values in both 'move' blocks are highlighted with a red box.

# Stay Connected

## Let's Connect!

- Tag me in the **VEX PD+ Community!** @Lauren\_Oberst
- Email: Lauren@vex.com

## Want to Learn More? Join VEX PD+ as an All-Access Member!

- Schedule a **1-on-1 Session** in VEX PD+