1.3 Block- Pick and Place

NAME: _____ *Date:*_____ *Section:* _____

INTRODUCTION

Robotic arms are excellent for performing pick and place operations such as placing small electronic components on circuit boards, as well as large boxes on pallets. A pick and place operation will require at least 5 points:

1. A home or safe location
2. A position above the object
3. A position at the object
4. A position above the drop off
5. A position at the drop off

In this activity you will learn how to make a basic ***Pick and Place*** operation in blockly. Through this activity you will learn how to program the robot to move and turn on its suction cup in blockly

Caution: NEVER wire anything to the Dobot Magician while it has power on. ALWAYS shutdown the Dobot before making connections or damage to the robot could occur. Be sure to ask your instructor if you have any questions.

KEY VOCABULARY

- MoveTo
- Placeholder
- Delay time
- Suction Cup
- Blockly Programming
- Pick and Place

All blockly commands and robotics terms have been put into separate *Appendices* and can be referred to at any time throughout these activities.

EQUIPMENT & SUPPLIES

- Robot Magician
- Dobot Blockly - Pick and Place Field Diagram
- 1” cubes or cylinders
- DobotLab Software or
- Suction Cup Gripper

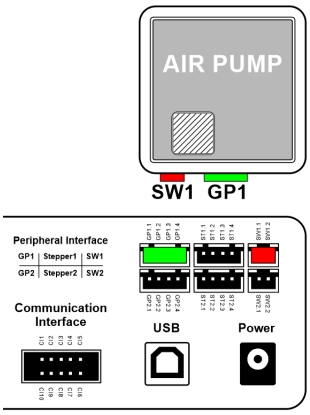
KEY CONCEPTS

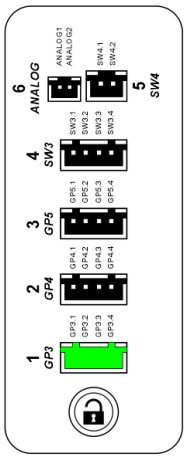
- Differentiate ways that robot arms can move using blockly: **Straight Line** and **Joint.**
- Differentiate between **absolute** and **relative** coordinates.
- What **Delays** are used for in robotics and how to create them.
- . What are some of the basic Dobot configurations?
- How **suction** is used to move objects and how to turn it on and off using blockly.
- Starting up and connecting the Dobot Magician.
- How to utilize a robot arm to move through a group of points by using an **end effector**.

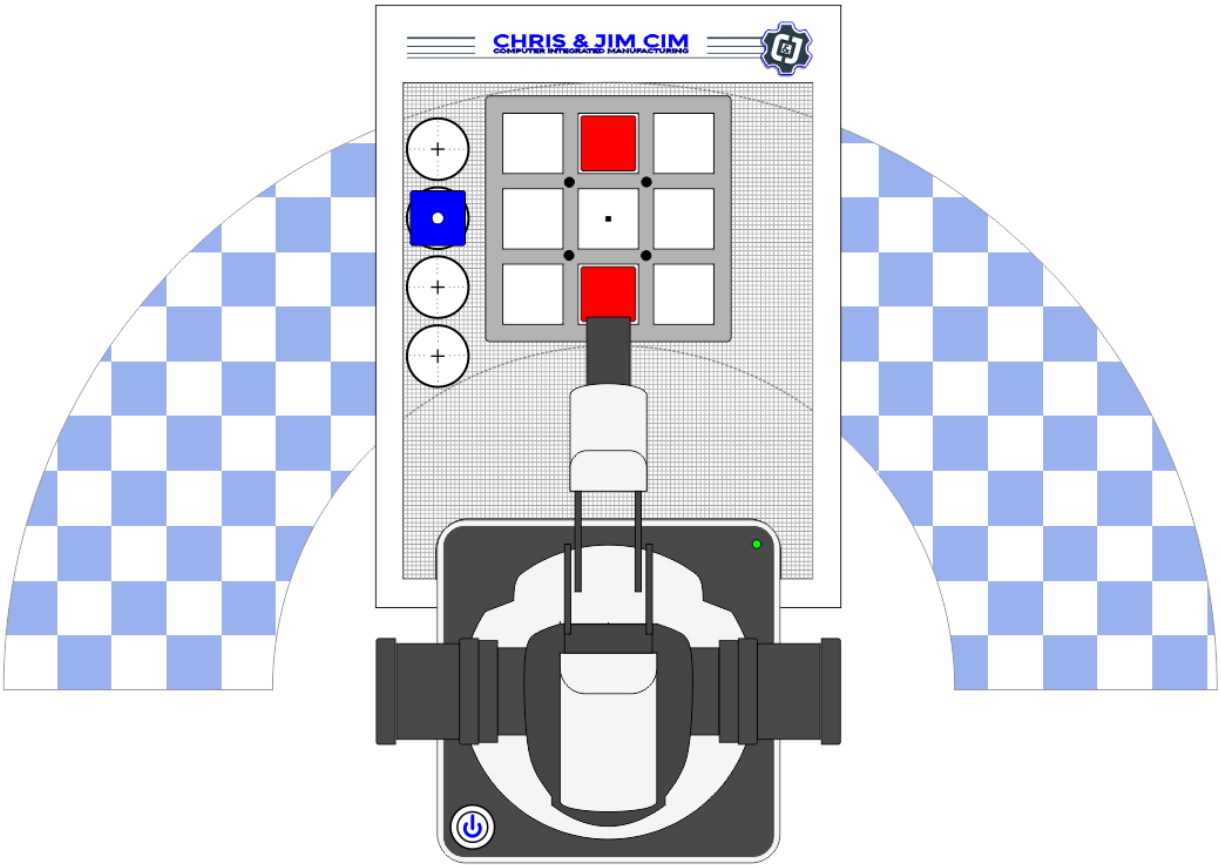
PROCEDURE

Caution: NEVER wire anything to the Dobot Magician while it has power on. ALWAYS turn it off before making connections or damage to the robot could occur. Be sure to ask your instructor if you have any questions.

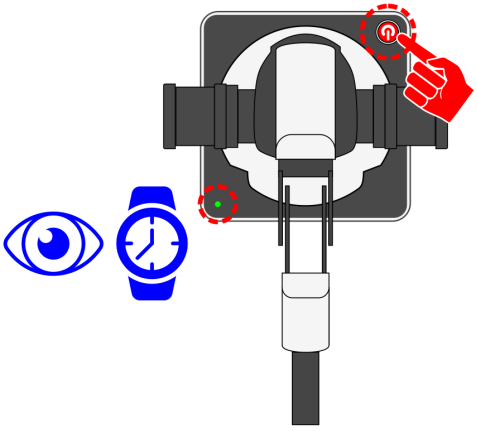
1. Print the Blockly - Pick and Place Field Diagram
2. Set up the robot with a **SUCTION CUP** and Air pump and place a cube in one of the red squares on the field diagram provided.





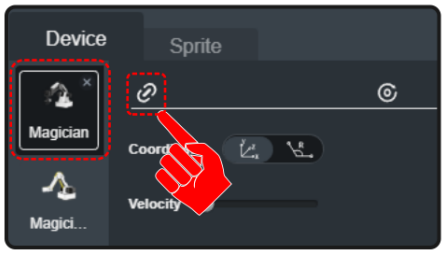


3. Once **ALL** of the wiring is done (sensors connected and robot arm connected), power ON the Magic Box and Robot. **WATCH** and **WAIT** for the robot to completely power on (GREEN INDICATOR)

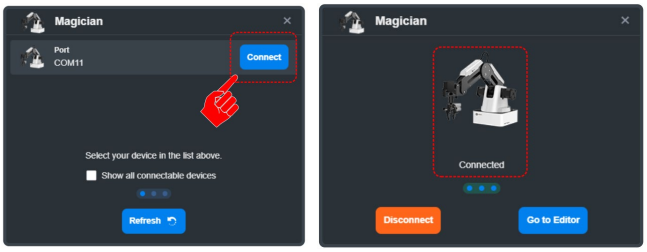


4. Open up DobotBlock Lab in the software and connect the robot.

5. Select the connect icon to establish control from the software to the robot. A connection window should pop up.

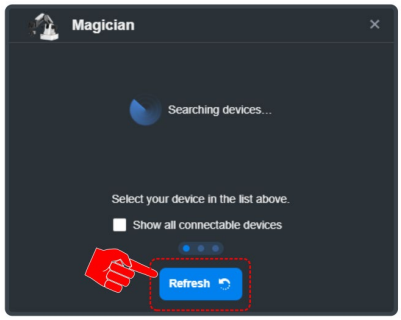


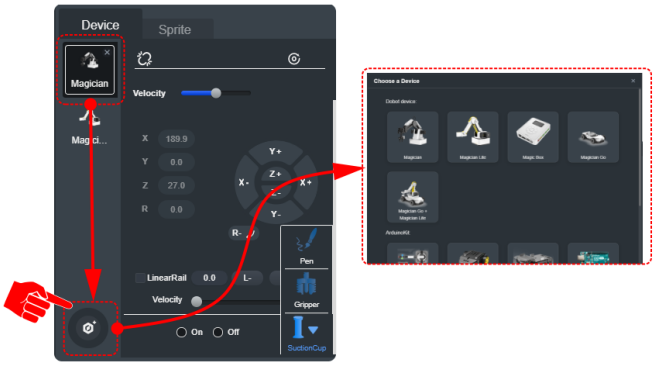
6. Select Connect from the pop up window. A new window should appear and show that the robot is Connected.



NOTE: if the Magician is not on the Device list, scroll down to select the "Choose a Device" option

NOTE: If no robot is found, the prompt will show as "Searching Devices". Reconnect the robot and select Refresh

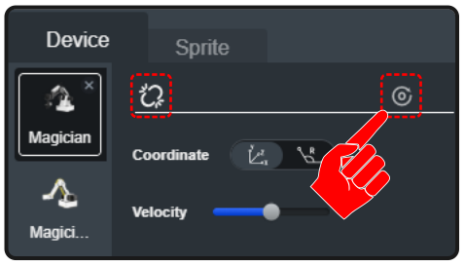




7. Once the robot is connected we need to home the robot to set its home position.

Before **HOMING** the robot, make sure the robot's **WORK ENVELOPE**, the area in which the robot can reach, is clear.

Select the **HOME** icon to start the robots homing process. **HOMING** the robot will return and set the robot to its initial **HOME** position.



8. Next, choose the SuctionCup as the Dobot's **END EFFECTOR** or **EoAT** (End of Arm Tooling).

   NOTE: You may need to scroll down to see the **EoAT** selections

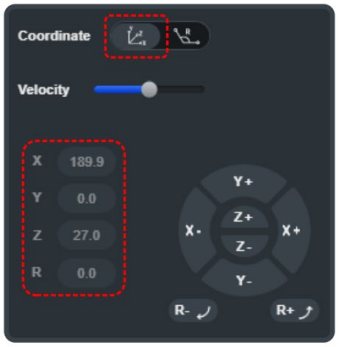
9. As with industrial robots, DobotBlock Lab provides the user with two methods for driving the robot manually. These options are XYZ MOTION and JOINT MOTION.

   JOINT MOTION - This method of movement allows you to configure one joint at a time by **DEGREES**. All of the other joints stay in the same relationship to the joint being driven. Practice with this type of movement.

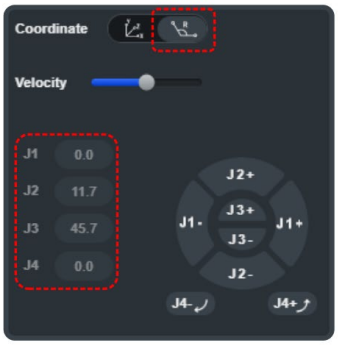
   XYZ MOTION – XYZ Motion will move on multiple axis at one time by **ABSOLUTE COORDINATES**. The robots motion is based on the keeping the EoAT in line with the axis chosen. Practice with this type of movement.

   The position data can be collected from either method depending on the coordinate motion style used when developing a program.

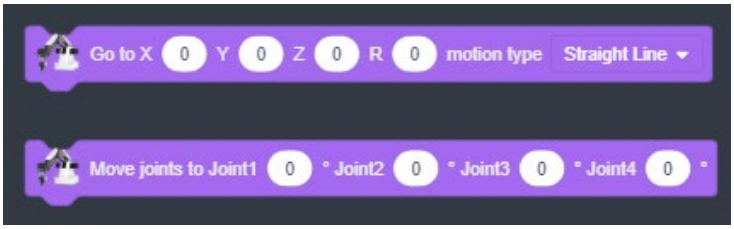
XYZ MOTION



JOINT MOTION

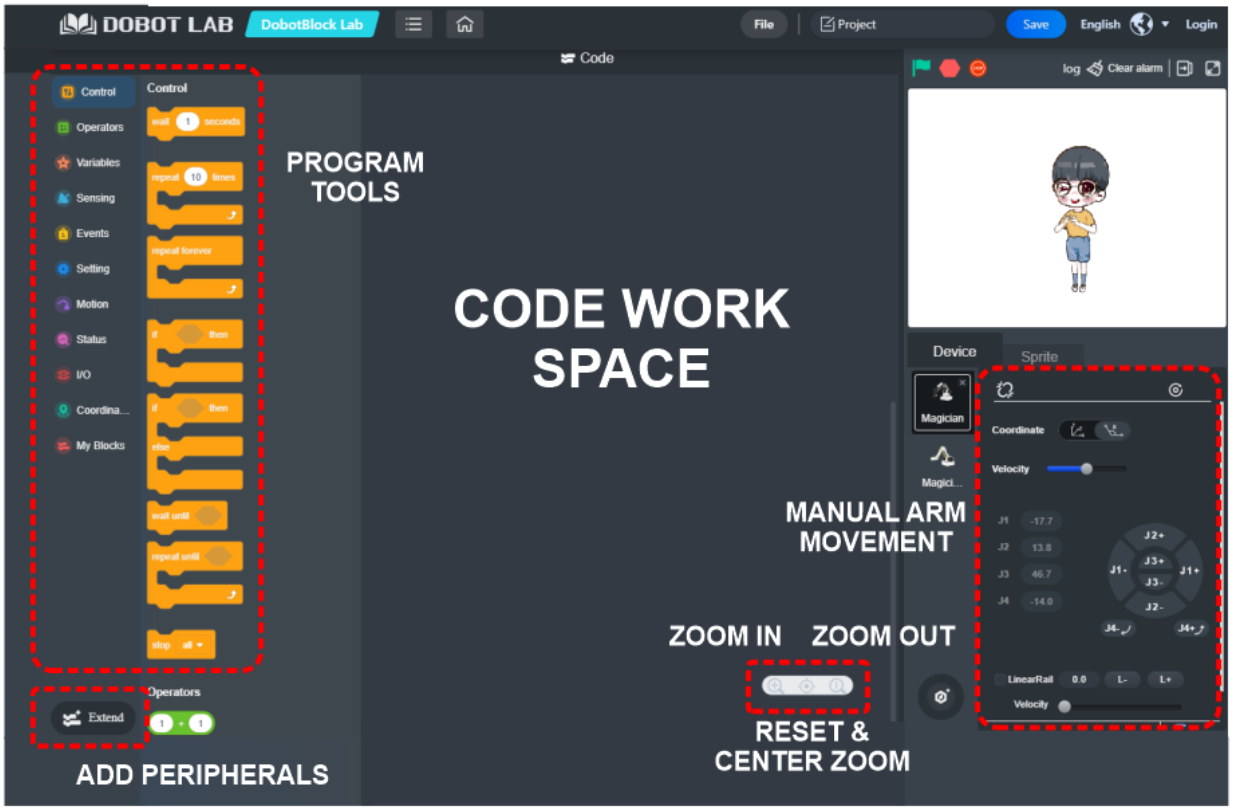


LINEAR MOTION	JOINT MOTION
X – In and Out	J1- Waist
Y – Side to Side	J2 - Shoulder
Z – Up and Down	J3 - Elbow
	J4 – Wrist *(Only aviable when the gripper or suction cup servo is attached)*



Shown below is the main menu for the Blockly programming section of DobotBlock Lab. On the left are categories and tools that each block of code is sorted into to. On the right are controls for zooming around the program as you develop it. The scroll wheel on the mouse can also be used to scroll up and down. The scroll wheel or the left mouse button can also be pressed and held to pan throughout the program workspace.

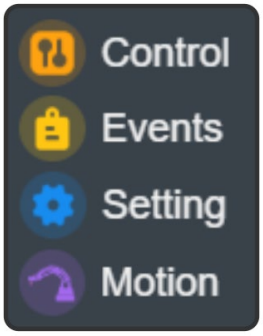
The Crosshair Icon can be used to reset the zoom level back to default and center your code in the middle of the field.



For this activity, we will be using tools from multiple categories. The categories we will be using are:

- Control
- Events
- Settings
- Motion

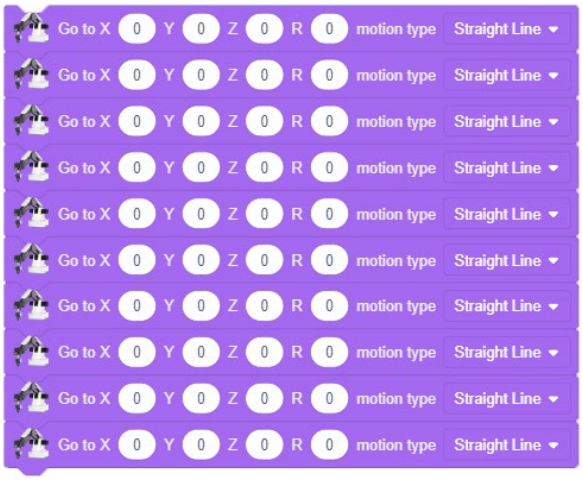
And can be found in the left menu as shown above.



10. Each program will typically start with some setup code (EoAT, Motion Velocity, Inputs, Outputs….)

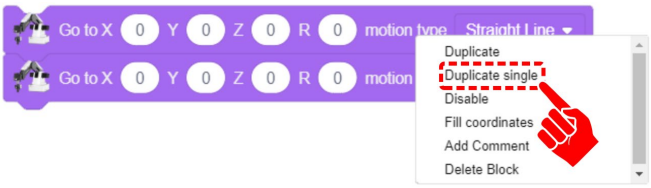
    For this program we will start by assigning the EoAT. Drag over the Set end effector tool from the **SETTINGS** category. Change the EoAT to the Suction Cup

11. Next, drag over multiple “Go to XYZR” from the **MOTION** category (8 steps, one for each movement of a pick and place routine).



HELPFUL TIPS

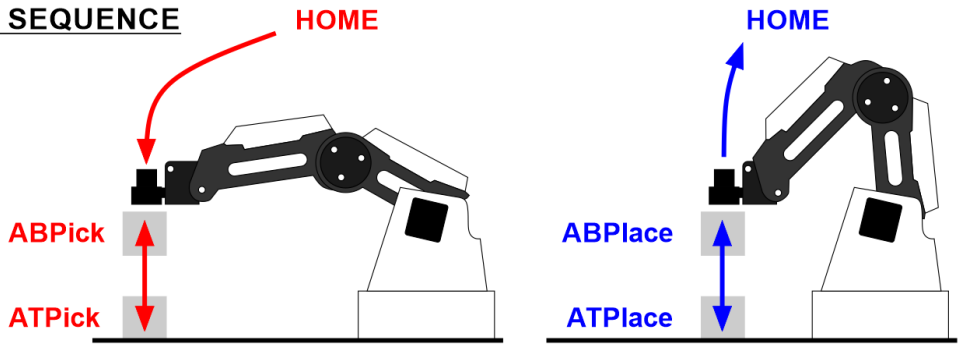
NOTE: you can also right click on an existing step in the program workspace and select DUPLICATE or DUPLICATE SINGLE.



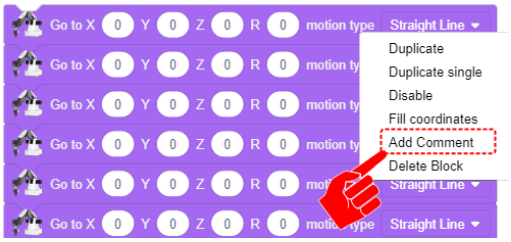
DUPLICATE will create a copy of the current step and all the steps that are below and attached to the current step.

PICK & PLACE SEQUENCE

1. HOME
2. ABPick
3. ATPick
4. Vacuum On
5. ABPick
6. ABPlace
7. ATPlace
8. Vacuum Off
9. ABPlace
10. HOME

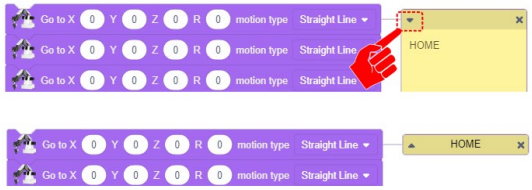


12. We will now add names/comments to each step. Right click on a step and select **ADD COMMENT**.



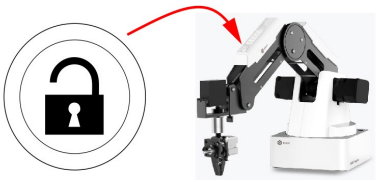
Type in the name of the step using the diagram above as a guide.

When the comment is complete, click the down triangle to collapse the comment



Repeat this process for all 8 movement steps

13. We now need to find the XYZR coordinates for each of the positions



Press and hold the lock button on the arm of the robot. This will release control of the arm to the operator and allow you to move the arm to a new location. Once at the new location, release the lock button to have the arm regain control and hold that position.

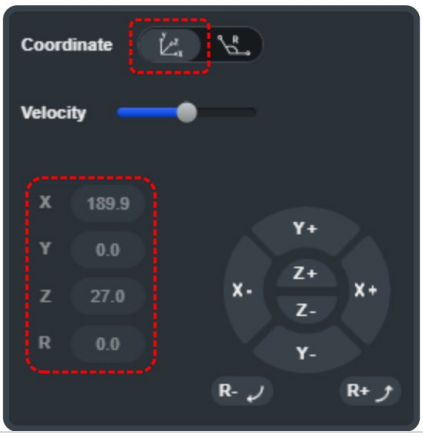
We will find the HOME position first. Using the lock button, move the arm to a new HOME position and release the button.

Ensure the COORDINATES shown are the XYZR

Right click on the 1st step (HOME) and select Fill Coordinates. This will use the robots current position to fill in the values for XYZR

It may be necessary to touch up the steps. For example, the X value of 189.9 can be set to 190.

Repeat this process for the remaining steps.



HELPFUL TIPS

Once the lock button is released, use the manual controls to fine tune the position or type in the desired values (round the numbers).

When dragging around blocks of code, if you select the top block, it will allow you to drag around that block and all of the blocks connected below it. If you selected a block of code from the middle of a string of blocks, it will disconnect that block and all of the blocks below it.

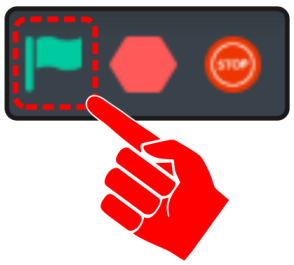
Complete the table below with all of the XYZ coordinates needed.

	X	Y	Z	R
1. Home				
2. Above Pick				
3. At Pick				
5. Above Pick				
6. Above Place				
7. At Place				
9. Above Place				
10. Home				

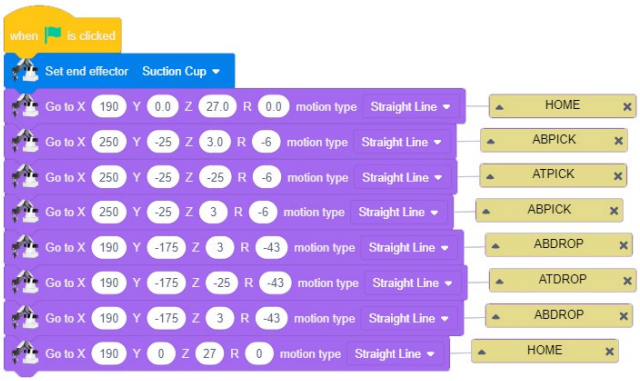
14. At this point we should test the positions we gathered. We will now perform a dry run (only perform movements) to see if our positions will work. To do this, the program needs a line of code that instructs it to run when the go button is selected.

15. Select the Go button (the GREEN flag) to run the program.

    From the EVENTS category, drag over the WHEN GO IS CLICKED tool and attach it to the top of your program.

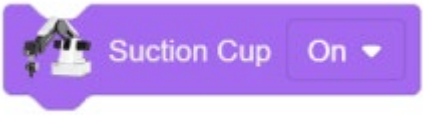


Make any adjustments to program as needed.

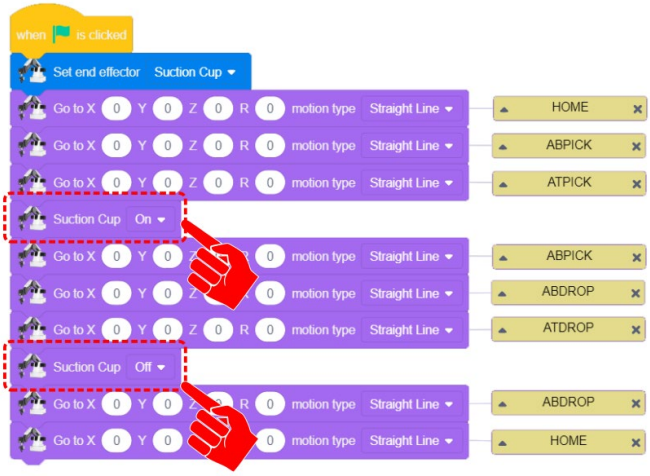


16. We will now add the steps needed to turn on and off the suction cup.

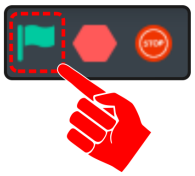
    From the MOTION category, drag over the GRIPPER tool.



    We will insert this step into the program. As you drag the step over to your program, the program will automatically separate where you drag the tool. We need the GRIPPER to close after we are at the position ATPICK and open after position ATDROP

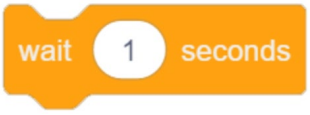


    Test the program and see if it works as expected.

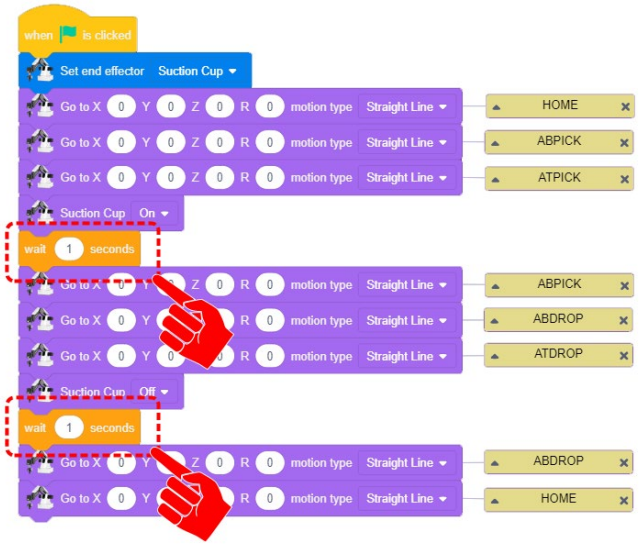


17. Finally, we will add a short pause after we turn on the suction cup and after we turn off the suction cup. This will ensure we have full pressure and have completely released the part before moving to the next step.

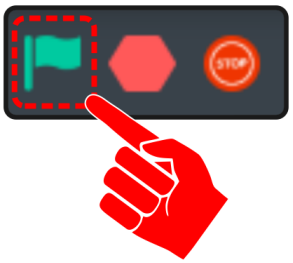
    From the **CONTROL** category, drag over the wait/seconds tool



Place a 1 second **WAIT** after each EoAT operation



Test the program and see if it works as expected.



HELPFUL TIPS

When dragging around blocks of code, if you select the top block, it will allow you to drag around that block and all of the blocks connected below it. If you selected a block of code from the middle of a string of blocks, it will disconnect that block and all of the blocks below it.

If your set up did not work correctly the first time, what did you have to do to make it work?

CONCLUSION

1. *What are the five needed positions for a pick and place operation?*

2. *Explain in your own words why it was necessary to add delay times into the program in the space below.*

3. *What is the purpose of the safe positions that are programmed above the object before it is picked up?*

GOING BEYOND

Finished early? Try some of the actions below. When finished, show your instructor and have them initial on the line.

_____ 1. Reverse the process so that at the end the robot puts the cube/cylinder back in its original position

_____ 2. Try picking and placing the object on locations that are not directly in front of the robot. What does this change? Can this be corrected in Blockly as it was in Teach and Playback? How?

_____ 3. For the sake of accuracy, change the speed of the arm as it moves from the ABPICK to the ATPICK, and ABDROP and ATDROP to slow it down by half (The codes needed to control speed are in the *SETTINGS* toolbox).