1 Blockly - Pick and Place

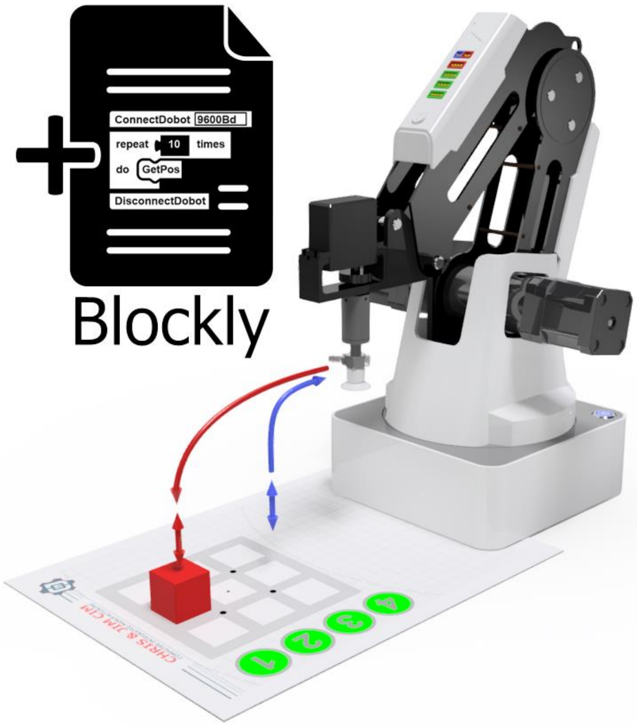
NAME: ____________________________ *Date:* ______________ *Section:* ____________

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 it's 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.

KEY VOCABULARY

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

All blockly commands have been put into a separate document called ***Blockly Vocabulary***, and can be referred to at any time throughout all of these activities.

EQUIPMENT & SUPPLIES

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

ESSENTIAL QUESTIONS

Essential questions answered in this activity include:

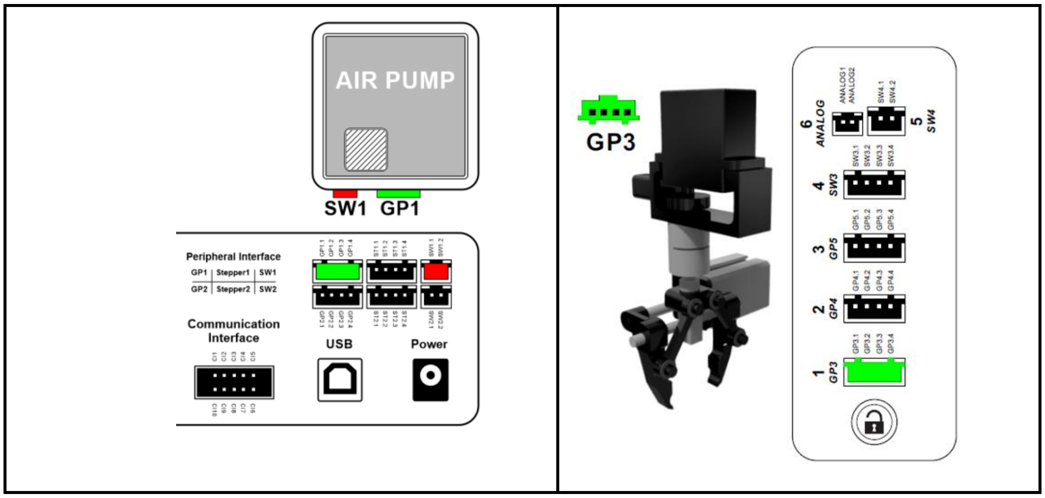
- How do I use blockly to move the robot?
- How can I create a *delay*?
- What method is used to turn the *suction on* and *off*?
- How do I use other end effectors?
- What are some of the basic Dobot configurations?

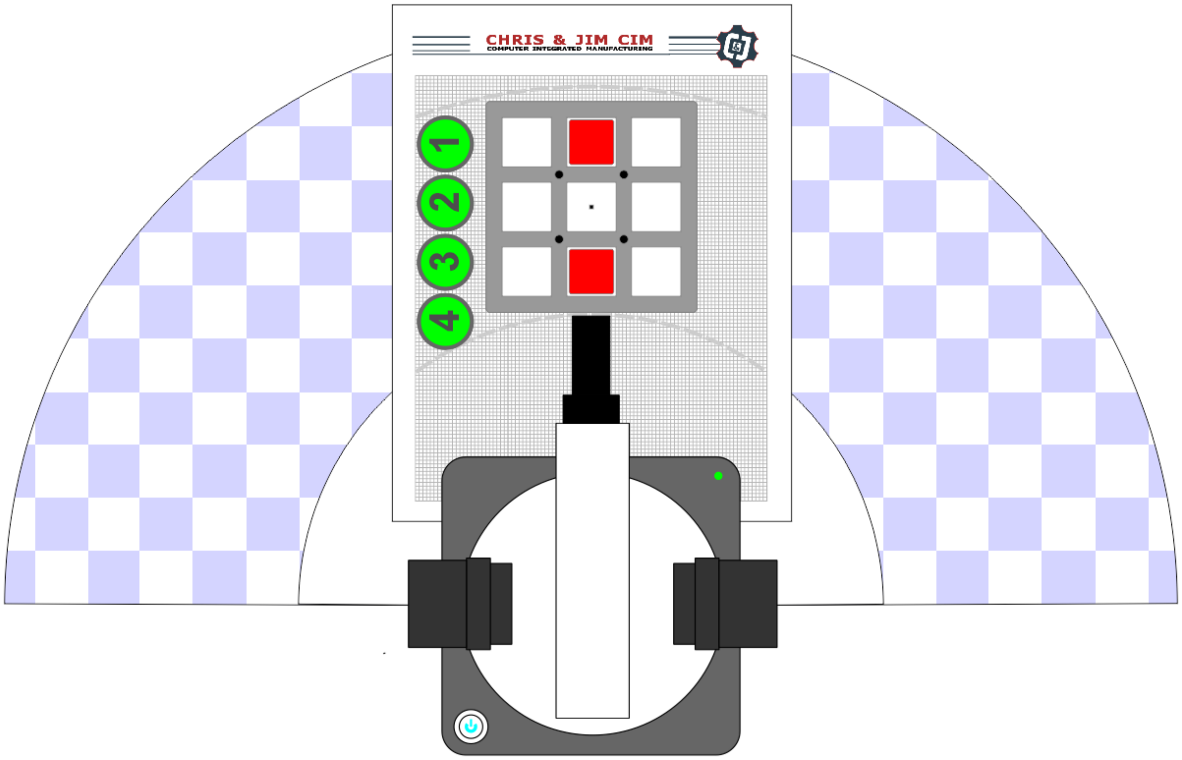
PROCEDURE

WARNING Power OFF

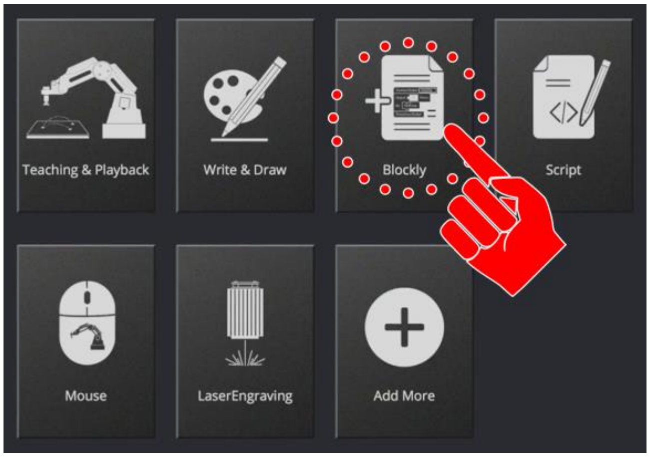
Caution: NEVER wire anything to the Dobot Magician while it has power on. ALWAYS turn it off before making connections or damage to the robot could occur. Be sure to ask your instructor if you have any questions.

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. Open up Blockly in the software

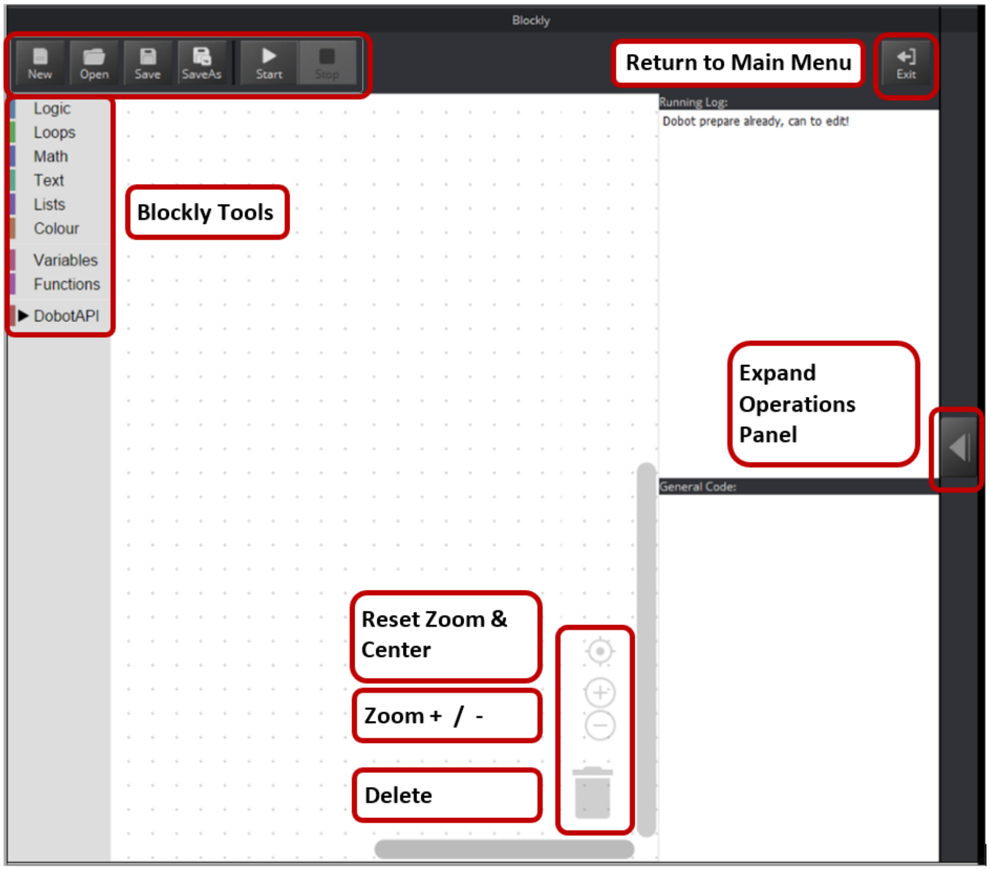


HELPFUL TIPS

When DobotStudio is closed with a file open, it will reopen with the last file used. Insure the file open is the one you want to edit, if it is not, you may end up overwriting another program

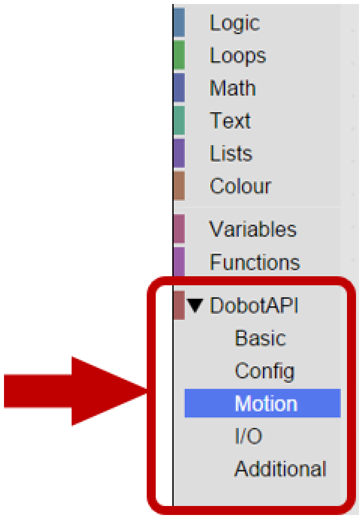
Shown below is the main menu for the Blockly programming section of DobotStudio. On the left are categories and tools that each block of code is sorted into to. On the right are controls for zooming around the program as you develop it as well as a trash can for any code element(s) you wish to delete. The scroll wheel on the mouse can also be used to zoom in and out of the program. In order to pan up and down or left and right you may either use the scroll bars shown or click and drag on an empty space in the program.

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

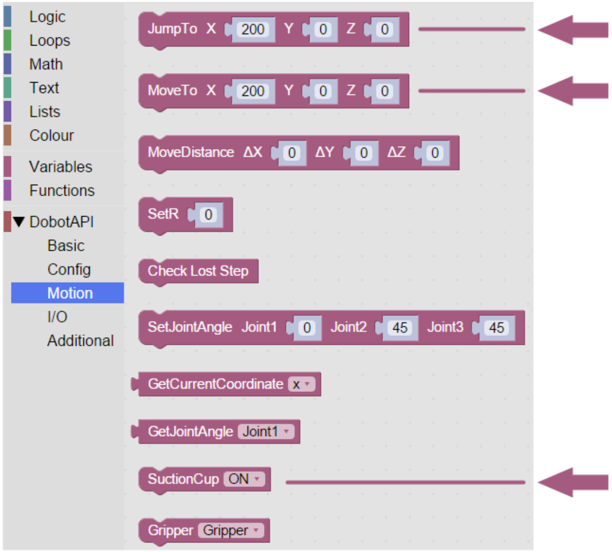


Dobot prepare already, can to edit!

In this program we will be using the **DobotAPI-Motion** sections in order to create a simple version of a Pick and Place program



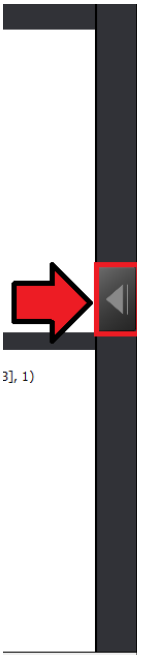
In this section there are blocks of code that allow you to move the Dobot ranging from the arm itself to the manipulator. For this program, we will need the ***MoveTo, JumpTo and SuctionCup[ON]*** blocks.

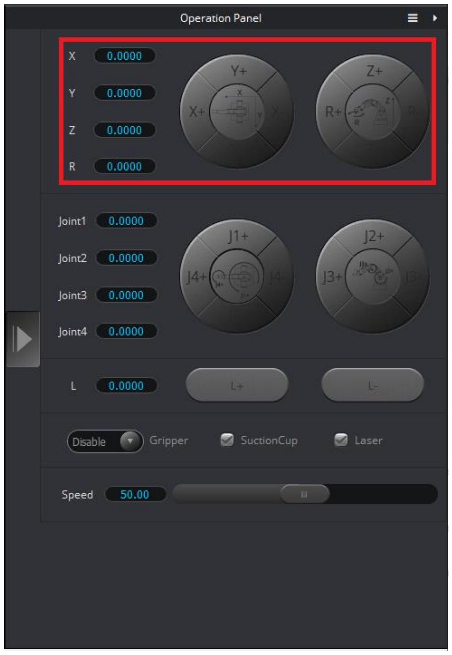


HELPFUL TIPS

For helpful definitions of all the important blockly commands, please see the glossary.

The first step in the Blockly Pick and Place process will be to define the X, Y, and Z coordinates for each step.





Expand the *Operation Panel*.

Use the X, Y, and Z Jog controls to locate the coordinates for each position and record them in the chart below. You can also use the Lock button to manually locate positions as we did in the Teach and Playback programming.

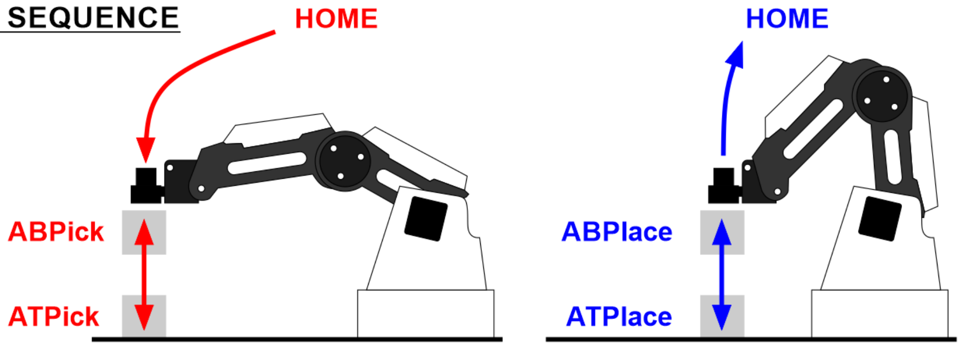
You may also use the lock button, and then look at the coordinates in this window to see what the XYZ values of the points are.

HELPFUL TIPS

Touch up the points so that all of the corresponding positions xyz values are aligned.

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



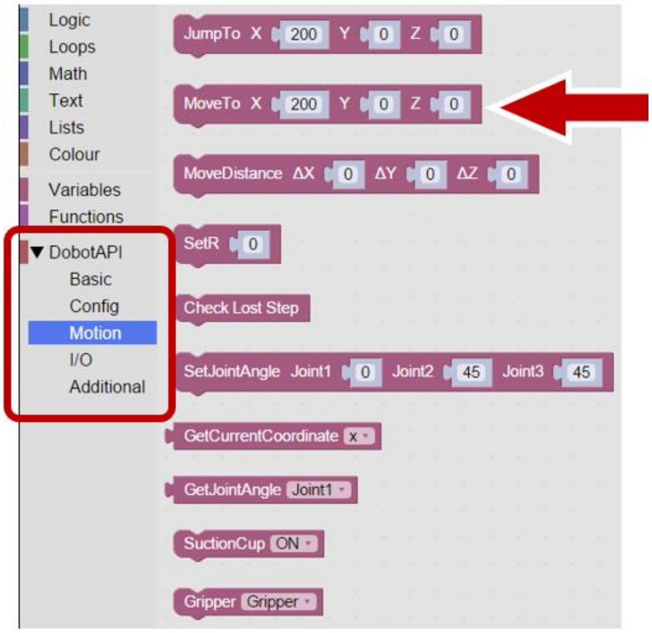
Complete the table below with all of the XYZ coordinates needed.

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

Now that all of the positions have been documented, we will transfer their locations in ***MoveTo*** blocks.

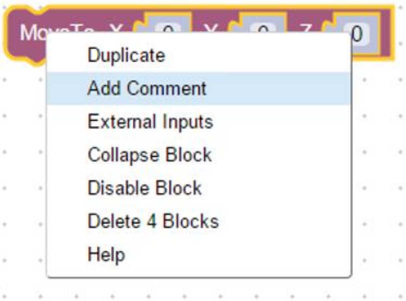
Drag over a ***MoveTo*** block from the *DobotAPI/Motion* Tool Box in the programming field.

This first block is going to be the HOME or start of the program. For this move, we want the robot to always go to a safe or reset position before continuing on with the program. This coordinate or point will be called home and it does not necessarily mean the position the robot is at when the home button is clicked.

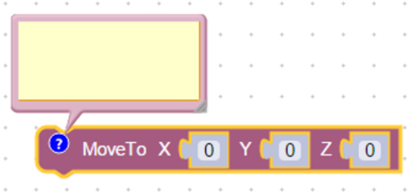


The Home position should be a position above the environment in which the robot is working, and safely away from any objects.

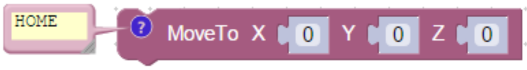
Once the coordinates for the home position have been recorded in the ***MoveTo*** command, right click on the command and select *Add Comment*.



> Please note the values in the images are placeholders, fake values to be changed later on, and that you will need to replace them with your own values that you get from your positions.



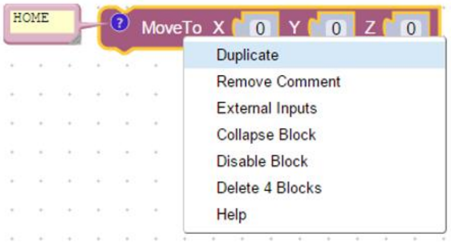
Click on the Question Mark that appears. This will allow you to name the position.



HELPFUL TIPS

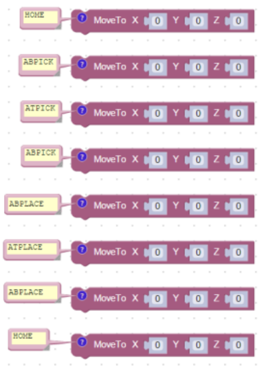
The Text Box that appears can be relocated and resized as needed. Clicking on the Question Mark again will collapse the call out box

Repeat the process for the remaining steps in the program. Ignore steps 4 and 8 (Vacuum On and Off) for now.



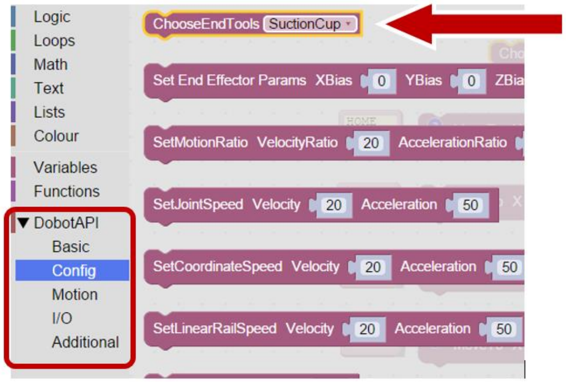
HELPFUL TIPS

You can also right click on a step and select Duplicate to create a copy of the step. This may speed up the process for steps that have similar coordinates

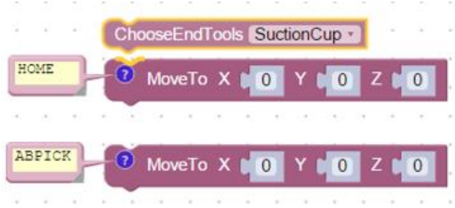


We will now assign the *End of Arm Tooling (EoAT)* as the Suction Cup

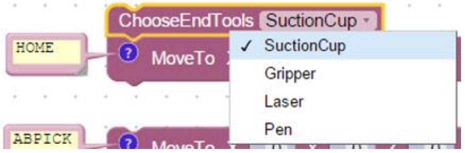
Drag the ***ChooseEndTools*** from the DobotAPI/Config Tool Box over to the programming field.


Logic
Loops
Math
Text
Lists
Colour
Variables
Functions
▼ DobotAPI
Basic
Config
Motion
I/O
Additional
ChooseEndTools SuctionCup
Set End Effector Params XBias 0 YBias 0 ZBia
SetMotionRatio VelocityRatio 20 AccelerationRatio
SetJointSpeed Velocity 20 Acceleration 50
SetCoordinateSpeed Velocity 20 Acceleration 50
SetLinearRailSpeed Velocity 20 Acceleration 50

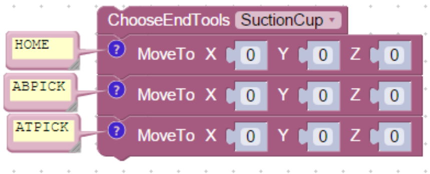
Drag the ***ChooseEndTools*** over to the 1st Home position until the links on both lines of programming turn orange and release the block code.



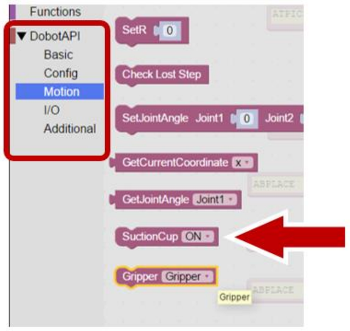
The ***SuctionCup*** should already be selected. If it is not, please select it from the drop down menu.

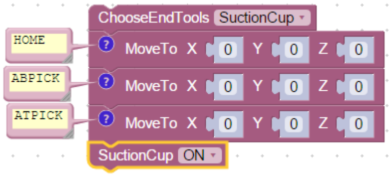


Link the next lines of code just before Step 4 - Turn on Vacuum



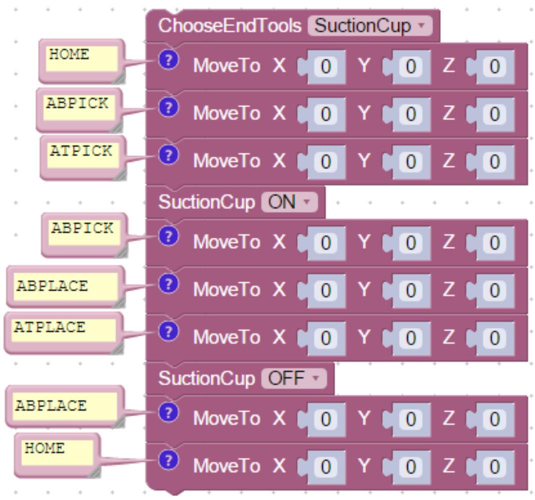
Drag the ***SuctionCup*** over and link it to the bottom of the Main block grouping.





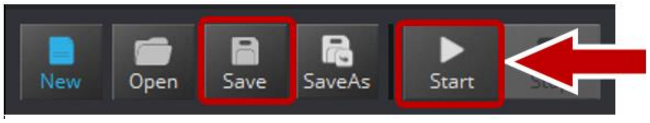
When dragging around blocks of code, if you select the top block, it will allow you to drag around that block and all of the blocks connected below it. If you selected a block of code from the middle of a string of blocks, it will disconnect that block and all of the blocks below it.

Complete the remaining portion of the program. Remember to add one additional ***SuctionCup*** command to turn off the Vacuum after placing the block down.

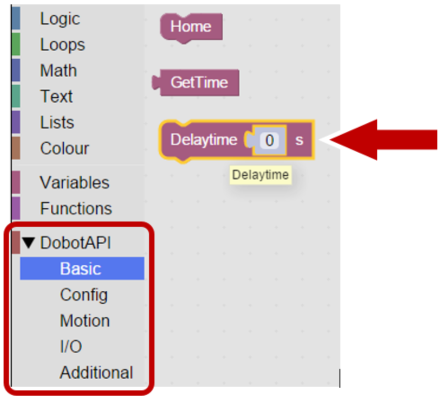


```
ChooseEndTools SuctionCup
HOME     MoveTo X 0 Y 0 Z 0
ABPICK   MoveTo X 0 Y 0 Z 0
ATPICK   MoveTo X 0 Y 0 Z 0
SuctionCup ON
ABPICK   MoveTo X 0 Y 0 Z 0
ABPLACE  MoveTo X 0 Y 0 Z 0
ATPLACE  MoveTo X 0 Y 0 Z 0
SuctionCup OFF
ABPLACE  MoveTo X 0 Y 0 Z 0
HOME     MoveTo X 0 Y 0 Z 0
```

Save your work and select *Start* to run your current program.



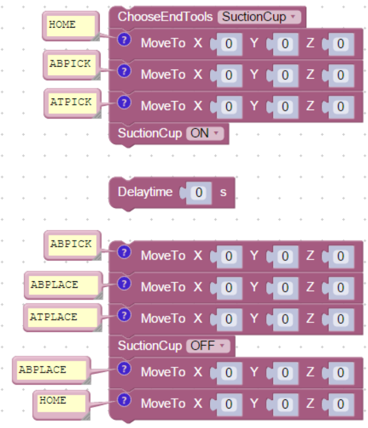
You should notice that we have the same issue in *Blockly* that we had in *Teach and Playback* where the suction cup has not fully engaged as the arm moves away from the ATPick location. We need to solve this problem in the same manner by adding a pause in the line of code to allow the vacuum to build up.



Drag and drop the ***Delaytime*** block over the the programming field.

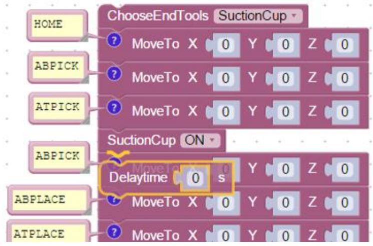
The block of code can be added in one of two different ways.

Option 1: Separate the lines of code selecting the ABPick, below the ***SuctionCup On*** command, and dragging it down. The ***Delaytime*** can now be added and the bottom of the code reassembled.

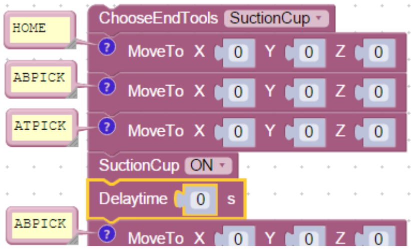


```
ChooseEndTools SuctionCup
HOME     MoveTo X 0 Y 0 Z 0
ABPICK   MoveTo X 0 Y 0 Z 0
ATPICK   MoveTo X 0 Y 0 Z 0
SuctionCup ON

Delaytime 0 s

ABPICK   MoveTo X 0 Y 0 Z 0
ABPLACE  MoveTo X 0 Y 0 Z 0
ATPLACE  MoveTo X 0 Y 0 Z 0
SuctionCup OFF
ABPLACE  MoveTo X 0 Y 0 Z 0
HOME     MoveTo X 0 Y 0 Z 0
```

Option 2: Drag the ***Delaytime*** over to its desired location. Wait for the connections to turn orange and release the code. This will automatically insert the line of code and shift the remaining code down.

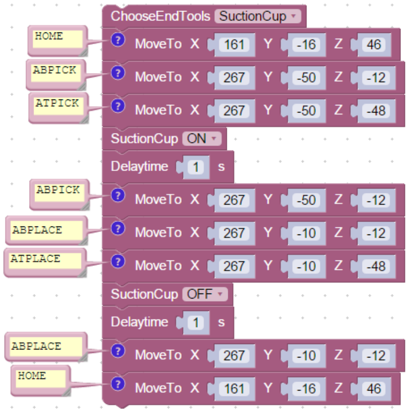


```
ChooseEndTools SuctionCup
HOME     MoveTo X 0 Y 0 Z 0
ABPICK   MoveTo X 0 Y 0 Z 0
ATPICK   MoveTo X 0 Y 0 Z 0
SuctionCup ON
ABPICK   Delaytime 0 s   Y 0 Z 0
ABPLACE  MoveTo X 0 Y 0 Z 0
ATPLACE  MoveTo X 0 Y 0 Z 0
```



```
ChooseEndTools SuctionCup
HOME     MoveTo X 0 Y 0 Z 0
ABPICK   MoveTo X 0 Y 0 Z 0
ATPICK   MoveTo X 0 Y 0 Z 0
SuctionCup ON
Delaytime 0 s
ABPICK   MoveTo X 0 Y 0 Z 0
```

Add an additional ***Delaytime*** after the vacuum is turned off. Again, Test your Code. The Delay time should be adjusted appropriately.



```
ChooseEndTools SuctionCup
HOME     MoveTo X 161 Y -16 Z 46
ABPICK   MoveTo X 267 Y -50 Z -12
ATPICK   MoveTo X 267 Y -50 Z -48
SuctionCup ON
Delaytime 1 s
ABPICK   MoveTo X 267 Y -50 Z -12
ABPLACE  MoveTo X 267 Y -10 Z -12
ATPLACE  MoveTo X 267 Y -10 Z -48
SuctionCup OFF
Delaytime 1 s
ABPLACE  MoveTo X 267 Y -10 Z -12
HOME     MoveTo X 161 Y -16 Z 46
```

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 wit 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?