W3B CIM [C. DoBot Color Sensor] Dobot Magician

CHRIS & JIM CIM
COMPUTER INTEGRATED MANUFACTURING

5 Blockly - Using the Color Sensor

Portfolio link
https://m-jeide.github.io/eng-portfolio/CIM/Dobot%20Magician

WARNING:

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.

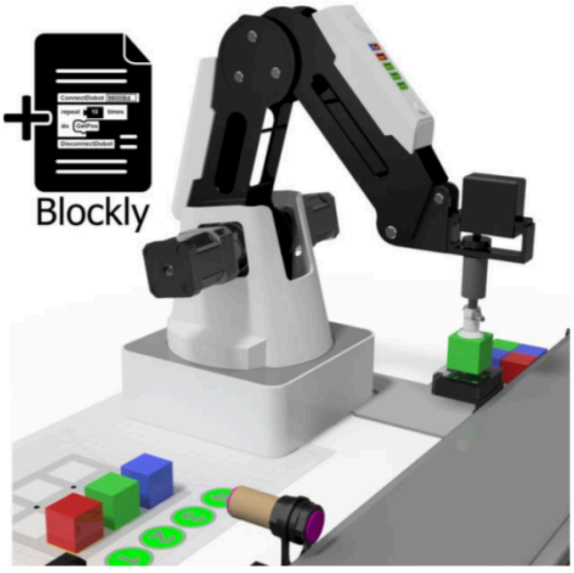
INTRODUCTION

INTRODUCTION

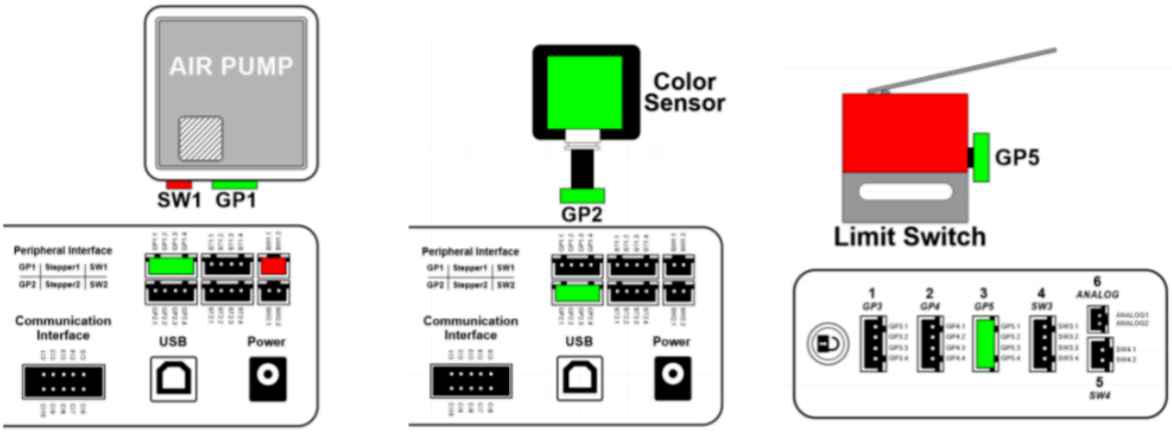
Sensors are often added to industrial robots in order for them to perform specific tasks. These sensors can be as simple as a color detecting sensor, or as complex as a full vision system that will allow a robot to be aware of its surroundings, or find a part and determine its location and orientation.

In this activity you will learn how to use and program the color sensor in blocky.

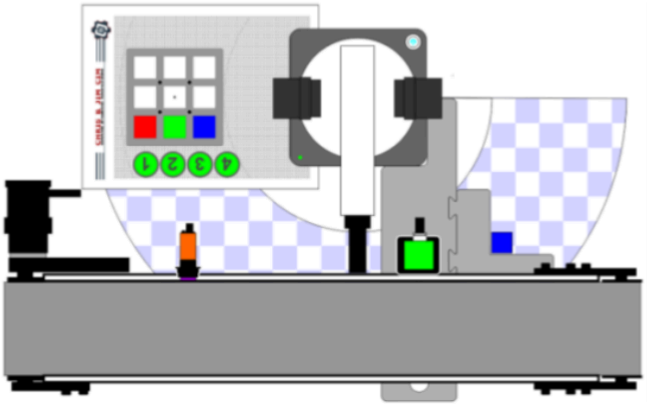
The dobot will pick up a part and move it above the color sensor. The dobot will then check the part's color and place it in a specific location for that specific color part. The robot will repeat the process each time a limit switch is pressed.



1. Set up the robot with a suction cup - ***GP1 & SW1***
2. Wire the robot with the Color Sensor plugged into port ***GP2***
3. Wire the robot with the Limit Switch plugged into port ***GP5 - EI05***



Set up the robot, conveyor, and color sensor as shown in the diagram below:



Complete the table below with all of the XYZ coordinates needed for all four blocks placed on the four corners.				
	X	**Y**	**Z**	**R**

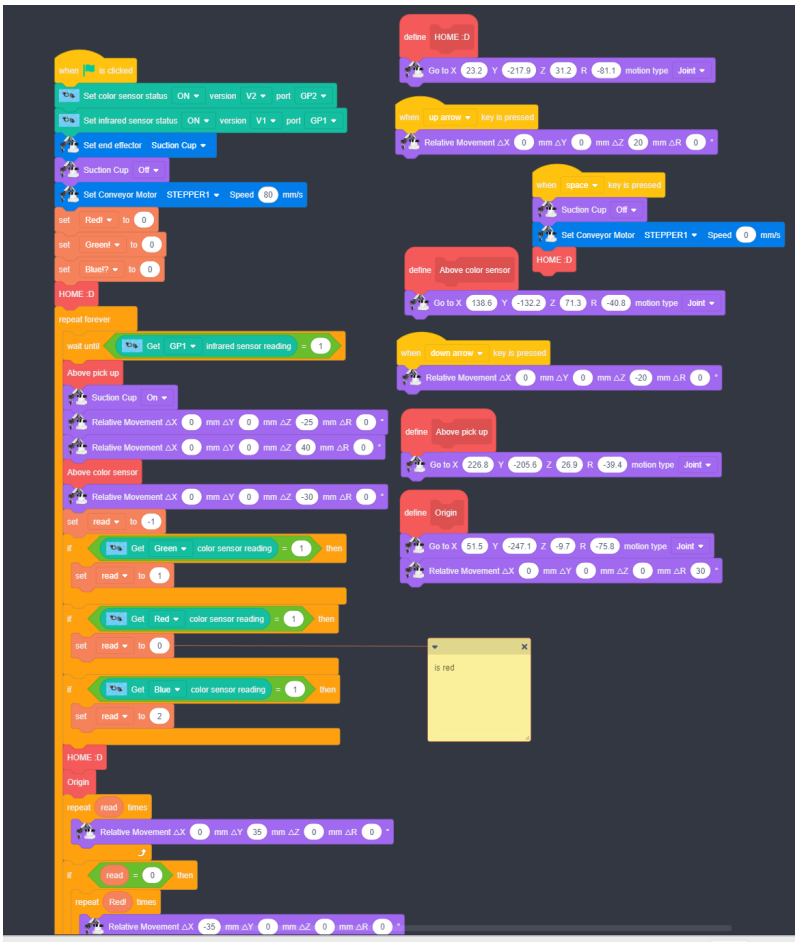
1. Home (center of grid but high above	23.2	-217.9	31.2	-81.1
2. Above Pick Up block (on conveyor belt)	226.8	-205.6	26.9	-39.4
3. Above color sensor (about 40mm above)	138.6	-132.2	71.3	-40.8
5. Origin (above 1st red cube position on cube)	51.5	-247.1	-9.7	-75.8
6. Relative movement - above pick up block (on conveyor belt) *Plunge*	0	0	-25	0
7. Relative movement - above pick up block (on conveyor belt) *moving up* - ensures block will not hit any of the sensors	0	0	40	0
9. Relative movement - above color sensor *plunge* - in order to have the block hover above sensor	0	0	-30	0
10. Relative movement - telling the Dobot to change columns on grid	0	35	0	0
11. Relative movement *used 3 times - for each color block but has the same job* - used to tell Dobot to move to next position in column if same color block has already been identifed	-35	0	0	0
12. Relative movement - placing block down in position that has already been identified prior to the *plunge*	0	0	-30	0

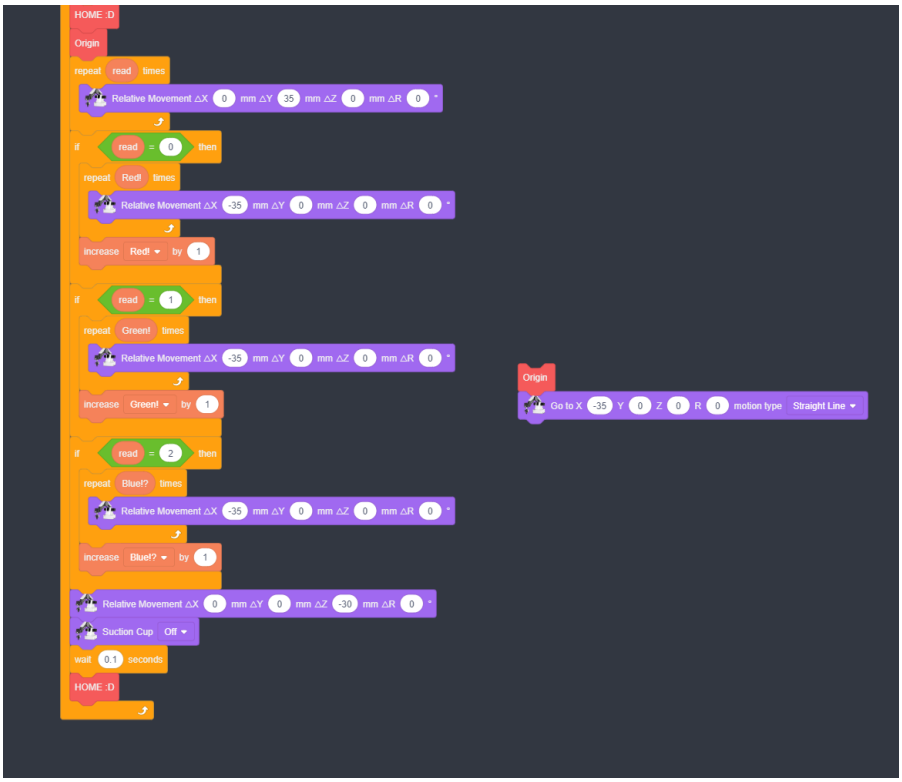
Video of Pick and Place Routine

https://www.youtube.com/watch?v=Lu_nDxDqv9Q

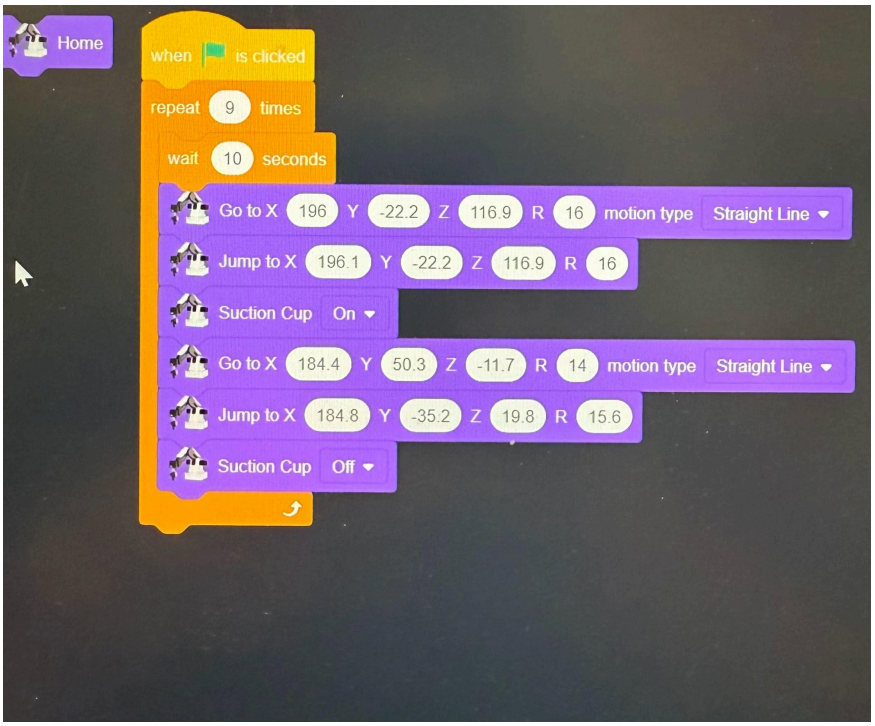
Screenshot of positions on the Dobot Software

Dobot 2 Code




HOME :D
Origin
repeat read times
Relative Movement ΔX 0 mm ΔY 35 mm ΔZ 0 mm ΔR 0
if read = 0 then
repeat Red! times
Relative Movement ΔX -35 mm ΔY 0 mm ΔZ 0 mm ΔR 0
increase Red! by 1
if read = 1 then
repeat Green! times
Relative Movement ΔX -35 mm ΔY 0 mm ΔZ 0 mm ΔR 0
increase Green! by 1
if read = 2 then
repeat Blue!? times
Relative Movement ΔX -35 mm ΔY 0 mm ΔZ 0 mm ΔR 0
increase Blue!? by 1
Relative Movement ΔX 0 mm ΔY 0 mm ΔZ -30 mm ΔR 0
Suction Cup Off
wait 0.1 seconds
HOME :D
Origin
Go to X -35 Y 0 Z 0 R 0 motion type Straight Line

DOBOT 1 CODE


Home
when is clicked
repeat 9 times
wait 10 seconds
Go to X 196 Y -22.2 Z 116.9 R 16 motion type Straight Line
Jump to X 196.1 Y -22.2 Z 116.9 R 16
Suction Cup On
Go to X 184.4 Y 50.3 Z -11.7 R 14 motion type Straight Line
Jump to X 184.8 Y -35.2 Z 19.8 R 15.6
Suction Cup Off

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

Since this assignment spanned over the course of several days, several of the positions had to be re-encoded or we risked the loss of precision due to the movements of the supplies. Another problem was that cubes would occasionally get stuck in the conveyor belt feeder, preventing the second dobot from collecting and sorting it.

Conclusion

1. In your own words, define a variable.

A variable is a piece of information that the computer stores for use at a later time, it can be modified as well.

2. In your own words, define a Function.

A function is a task or series of tasks that the computer remembers and executes when the function is called.

3. Explain what would have to be done to palletize two layers using bullet points or a step-by-step list below.

1. Go home
2. Set X, Y, Z to be the top-left corner of the grid for the first layer
3. Repeat two times
    a. Repeat three times
        i. Repeat three times
            1. Go home
            2. Grab block
            3. Go home
            4. Drop block at computed position
            5. Add +35mm to X
        ii. Remove -105mm from X
        iii. Add +35mm to Y
    b. Remove -105mm from Y
    c. Add +35mm to Z
4. Go home