

Chapter One Robot Introduction and Dobot Experience

1. Know Teaching & Playback

2. Dobot Writing & Drawing

3. Play with Laser Engraving

4. Step into 3D Printing World

Chapter Two Dobot Blockly Chapter Three Integrated Application - Assembly Line Operations

1. Know Program

2. Loop Control

3. Conditional Statement

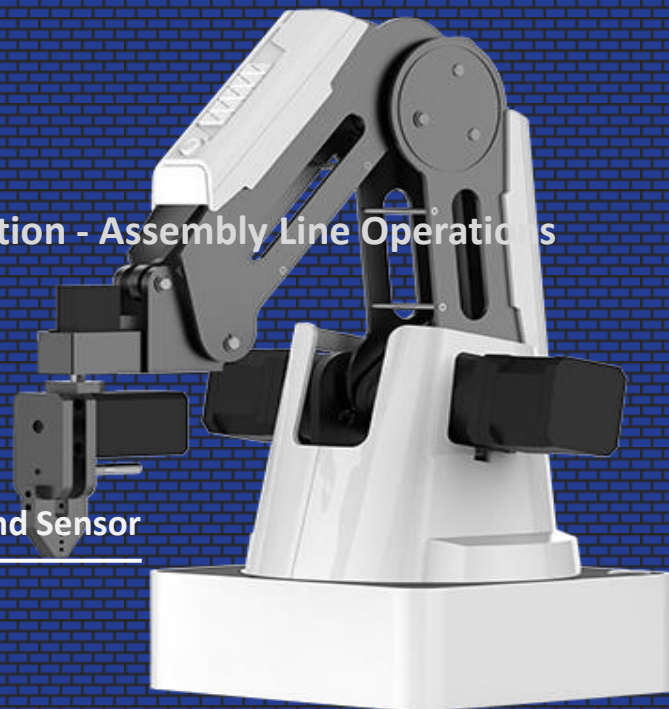
4. Using Functions

1. Robot Using in the Assembly Line

2. Blockly EIO Multiplex

3. Blockly and Assembly Line

4. Comprehensive Application of EIO and Sensor



Robot Introduction

A robotic arm is an automatic control device that mimics human arm functions and performs a variety of tasks. The robot system has multiple articulated links and allows movement in a planar or three-dimensional space or linear displacement movement. Structure by the mechanical body, the controller, servo and sensor components, and set a series of certain movement by the program, input the job sequence by the skilled operator, robot can learn it and operate correct movement repeatedly.



Chapter One Robot Introduction and Dobot Experience

1. Know Teaching & Playback

2. Dobot Writing & Drawing

3. Play with Laser Engraving

4. Step into 3D Printing World



Course Objectives

- Learn the function of robot writing & drawing;
- Master the operational method of robot writing & drawing.

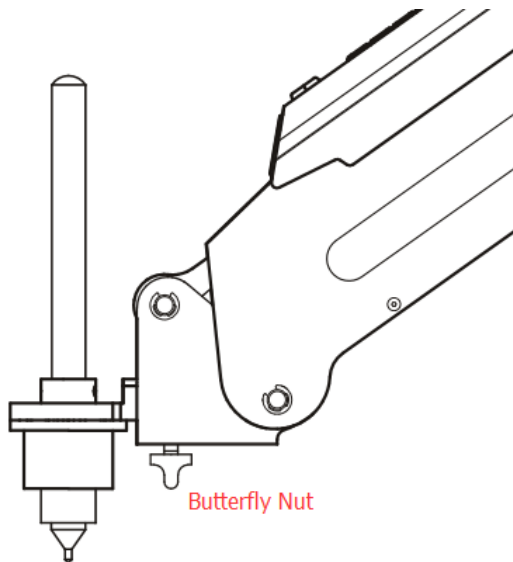
Equipment Needed

- A set of Dobot Magician
- A computer
- A pen holder
- 5 pieces of white paper

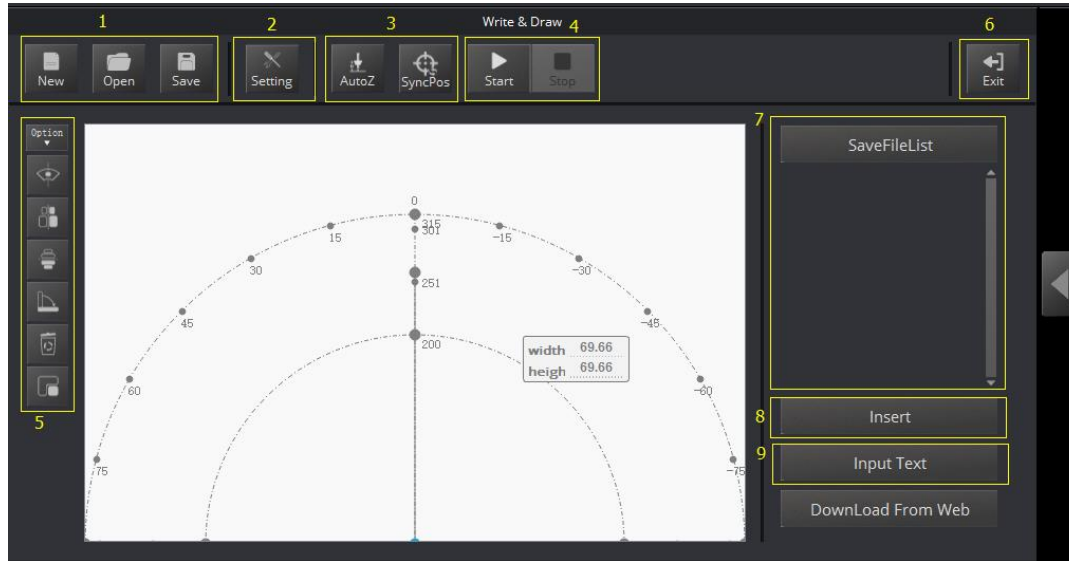
Robotic arm not only can follow the point location (PT) of the movement, but also follow the track, point into a line, playback the path accurately.

Installation of writing accessories

- Install the pen into the fixture;
- Lock the fixture into the end effector with butterfly nut.



The software interface introduction of writing & drawing



1. Files Operation: New/open/save

2. Tools Selection

Configuration of current tools as pen or laser

3. Position Synchronization

Move Dobot to a starting position to observe the drawing area

4. Start Stop

5. Tools bar of picture operation

Center, upside down, left-right flip, turning 90° , delete, zooming

6. Return to application main interface

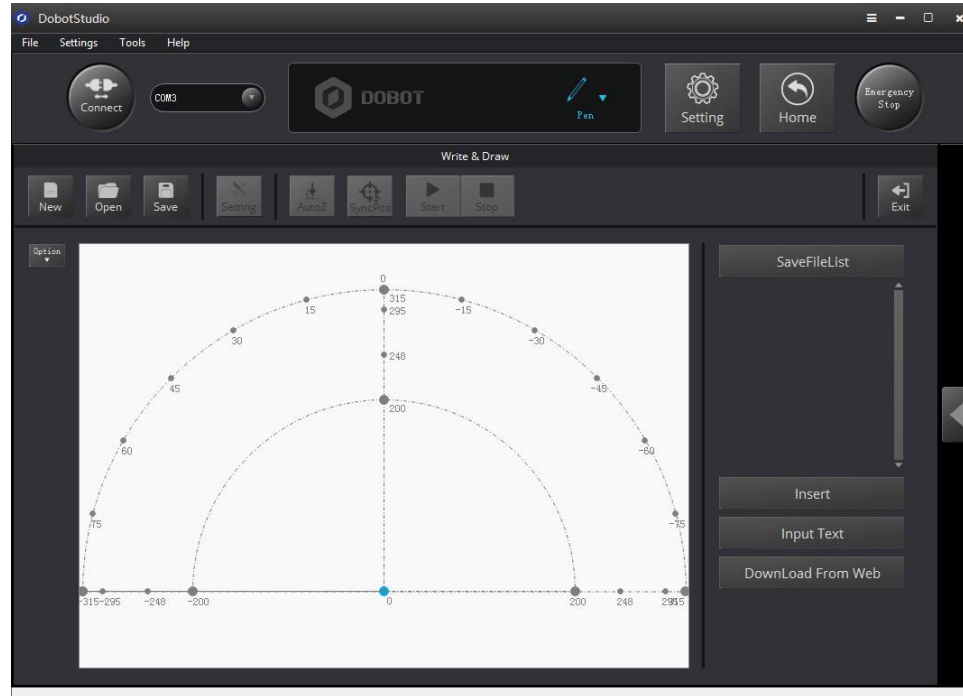
7. Files operation area: Open Recent

8. Insert shape

9. Input text

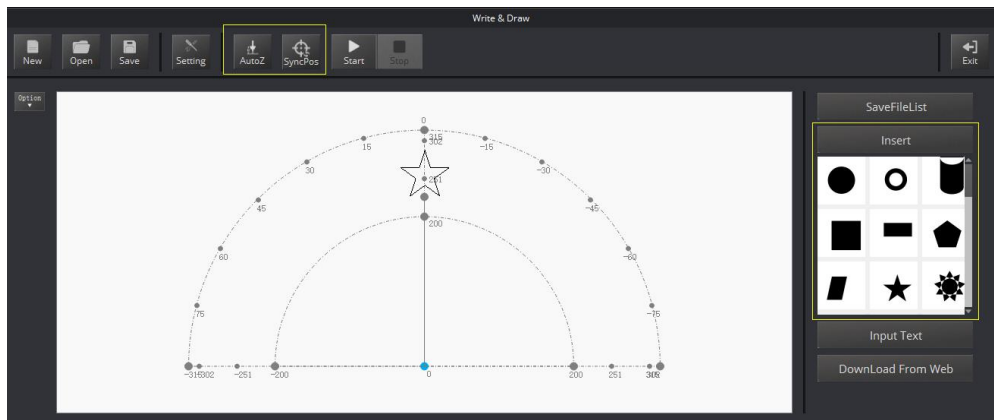
Connect Dobot Studio

Open the software, click writing & drawing module, here comes the interface of writing & drawing as follows:



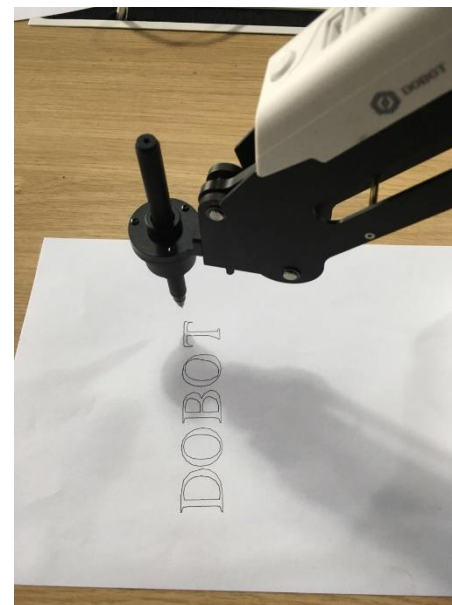
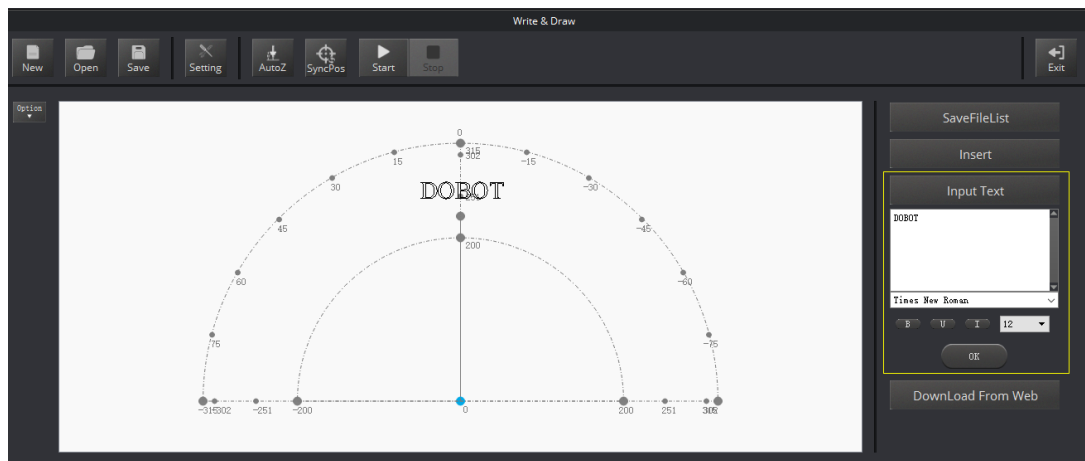
Import included pictures

- Insert simple pictures of five-pointed star;
- Adjust the nib position: press and hold down Unlock key on the forearm and drag the forearm to make the nib contact to paper, also, you can control Z axis moving down to paper gradually;
- Click AutoZ and get the current Z value, thus, you need not adjust the nib position when you write next time;
- Click SyncPos, Dobot will move into the top position of writing starting point automatically;
- Click Start, Pause and Stop to control Dobot.



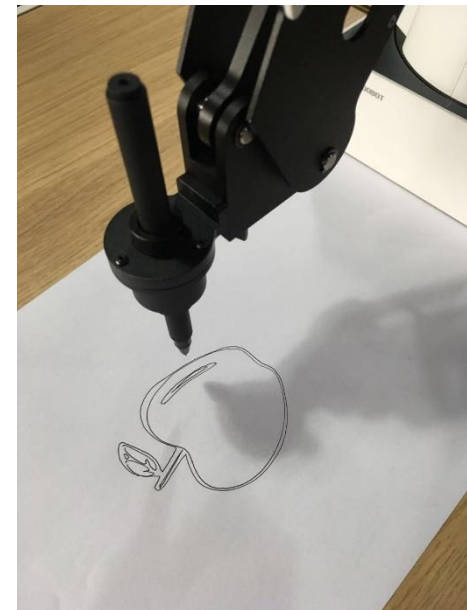
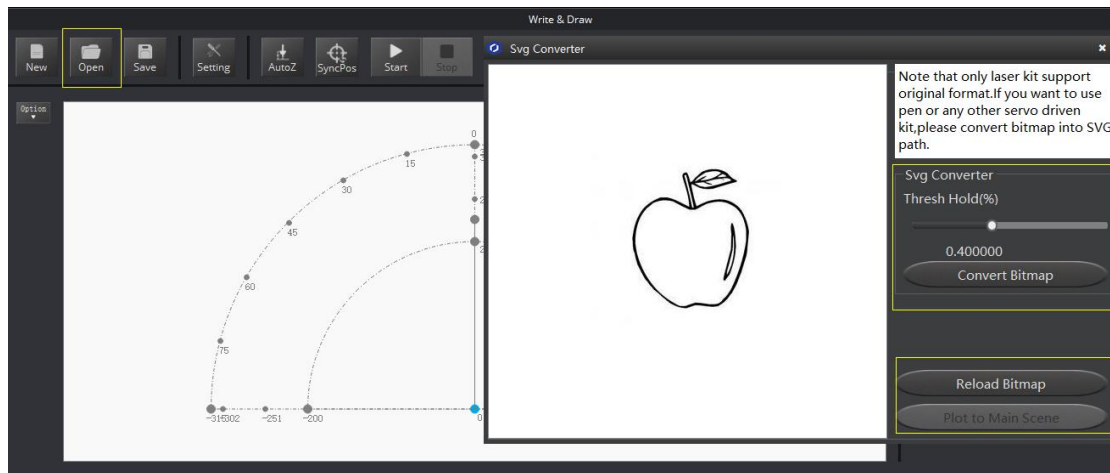
Input texts manually

Input Dobot Magician manually, and click Start.



Import the pictures directly

- Import pictures , convert these formats (BMP/JPEG/JPG/PNG and so on) into SVG recognized by Dobot, once took and once draw;
- After opening the picture, set proper thresh hold, click Convert Bitmap, then SVG path file will come out automatically, then click Plot to Main Scene, the ready trajectory can be loaded in writing main interface.



Use the computer drawing board to draw, import the arm software, let the robot arm draw. To see who painted fast and good, excellent works for the post-wall display.

Chapter Two Dobot Blockly

1. Know Programming

2. Loop Control

3. Conditional Statements

4. Using Functions

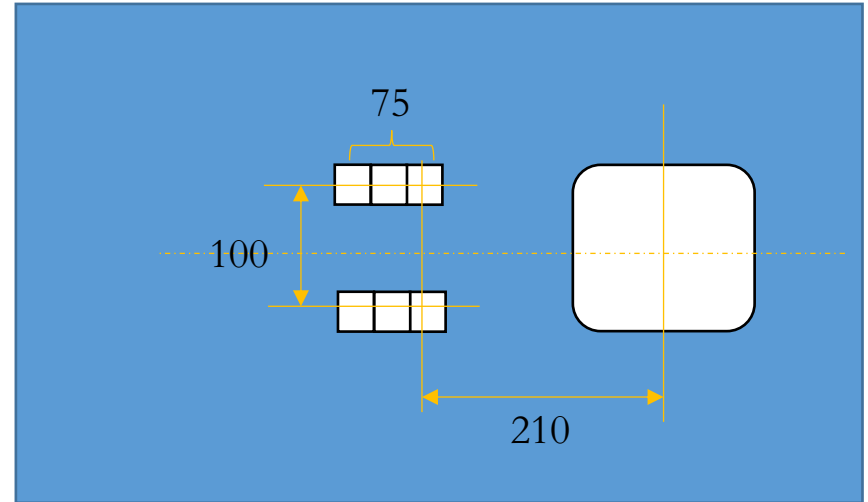


Course Objectives

- Learn programing
- Learn how to use Dobot by graphical program

Equipment Needed

- A set of basic robotic arm
- 5 building blocks of 25*25*25mm
- 4 pieces of A4 white paper



What is programming?

What is graphical programming?

The brief introduction of Blockly

How to control Dobot by Blockly?

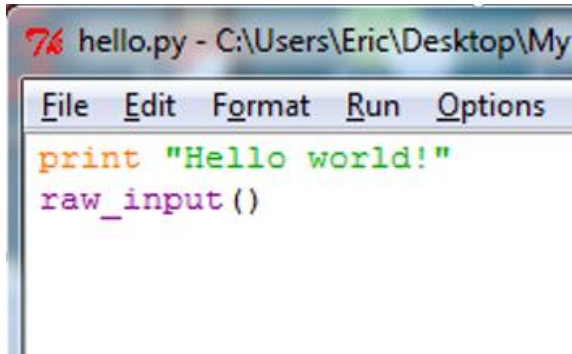
The first program to control Dobot

Divergent thinking: Write your program

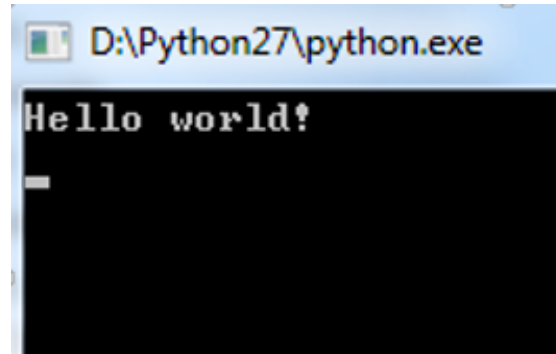
What is programming?

Programming is the process of controlling a computer in a programming language. Programming languages have a fixed format and vocabulary constituting the source code in accordance with the specific syntax together.

For example, let the computer screen show: Hello world!



```
76 hello.py - C:\Users\Eric\Desktop\My
File Edit Format Run Options
print "Hello world!"
raw_input()
```



```
D:\Python27\python.exe
Hello world!
_
```


What is graphical programming?

Computer language programming is not only digital, loop, variables, but also to master a variety of programming language-specific functions and syntax, colorful and complex. Then is there a better and easier way to program it?

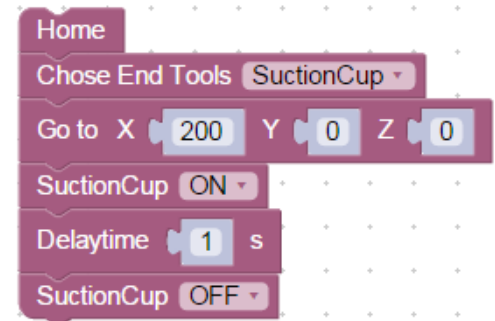
Yes, the answer is - graphical programming!

With it, you will do not know grammar, you can not use the keyboard, as long as the way by building blocks can be programmed!

Take a look at the following comparison of graphical programming and scripting:

```
posX = None
i = None

posX = dType.GetPoseEx(api, 1)
for i in range(1, 6):
    dType.SetPTPCmdEx(api, 2, (200 + i * 20), 0, 0, 0, 1)
    dType.SetWAITCmdEx(api, 1, 1)
dType.SetPTPCmdEx(api, 2, posX, 0, 0, 0, 1)
for i in range(1, 6):
    dType.SetPTPCmdEx(api, 0, posX, 0, 0, 0, 1)
    dType.SetWAITCmdEx(api, 1, 1)
```



Obviously, the blockly is easy to understand and clear at a glance.

Graphical programming is a platform created for zero-based people, fool-proof, focusing on cultivating logical thinking, training the problem-solving capability and lay a solid foundation for future programming learning !

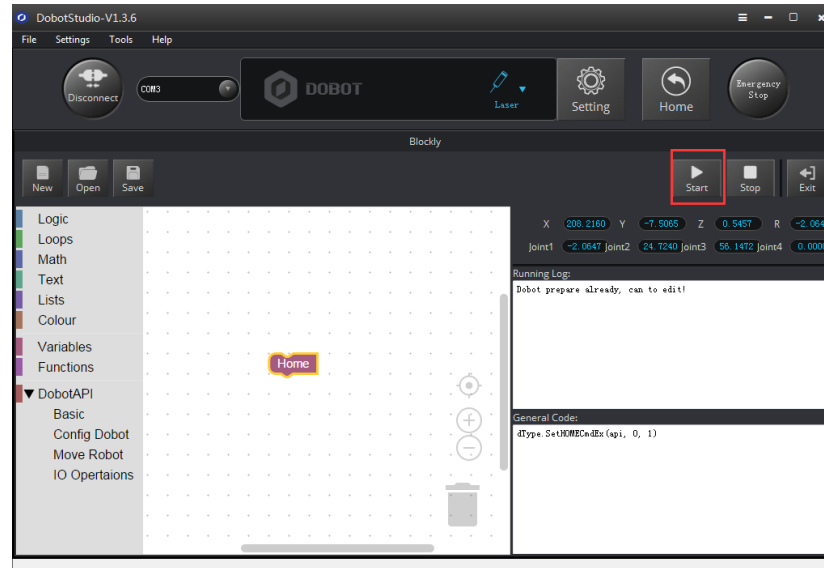
The brief introduction of DobotBlockly



Dobot Blockly is a platform for programming based on Google Blockly. In this process, users can program to for second development through puzzle, stright and easy to understand.

Now let's experience the usability of Blockly through a easy program;

Below the movement module, drag Home module into the main interface window, and then click Start, Dobot will reset to zero.



How about this? Is is very simple? This is the first major advantage of graphical programming!

The brief introduction of DobotBlockly



The advantage of Blockly mainly shown as follows:

- Simplify the preparation process, WYSIWYG; building block programming, the mouse dragging;
- Integrate basic logic, loop, mathematics, text, variables, functions and other modules, design, debugging, debugging steadily, effectively improve the programming speed.
- Integrate the exclusive API, whether it is the movement of the robot arm control, or the expansion of other sensors, such as temperature, sound sensors, all can be easily achieved, and effectively reduce the difficulty of secondary development;
- Control all the expansion of the robot arm IO interface, a powerful expansion of the function;
- Graphical modules and general code comparison, help to further understand the programming syntax and structure;
- Focusing on training logic programming thinking and troubleshooting capacity.

The screenshot displays the DobotBlockly interface. On the left is a sidebar with a category list: Logic, Loops, Math, Text, Lists, Colour, Variables, and Functions. The 'Logic' category is selected, showing blocks for 'if', 'do', and 'and'. A 'DobotAPI' folder is expanded, revealing sub-categories: Basic, Config Dobot, Move Robot (highlighted), and IO Operations. The main workspace contains a sequence of blocks: a 'Home' block, a 'Chose End Tools' block set to 'SuctionCup', a 'Set Linear Speed' block with 'velocityRatio' set to 20 and 'Set Linear Speed' set to 50, a 'Go to' block with X=200, Y=0, and Z=0, and a 'SuctionCup' block set to 'ON'. On the right, a 'General Code:' panel shows the corresponding JavaScript code:

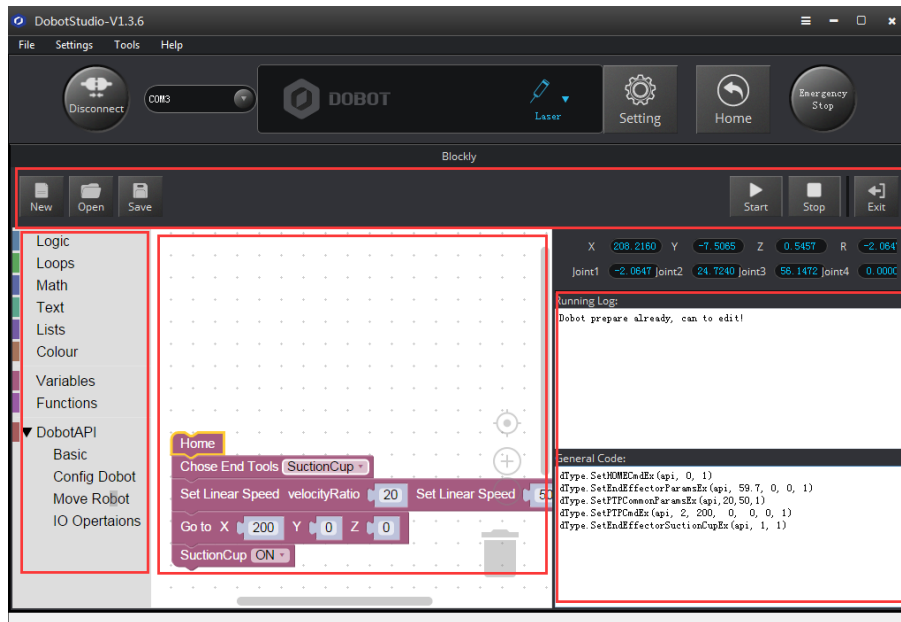
```
dType.SetHOMECmdEx(api, 0, 1)
dType.SetEndEffectorParamsEx(api, 59.7, 0, 0, 1)
dType.SetPTPCCommonParamsEx(api, 20, 50, 1)
dType.SetPTPCmdEx(api, 2, 200, 0, 0, 0, 1)
dType.SetEndEffectorSuctionCupEx(api, 1, 1)
```

The brief introduction of DobotBlockly



The main interface of Blockly, divided into four areas:

- Menu bar: new/open/start /stop and other operations;
- Module list: including basic logic, loops, variables and Dobot exclusive API;
- Program edit area: Main program edit area;
- Run log and general code display area: Show the corresponding code of each module.



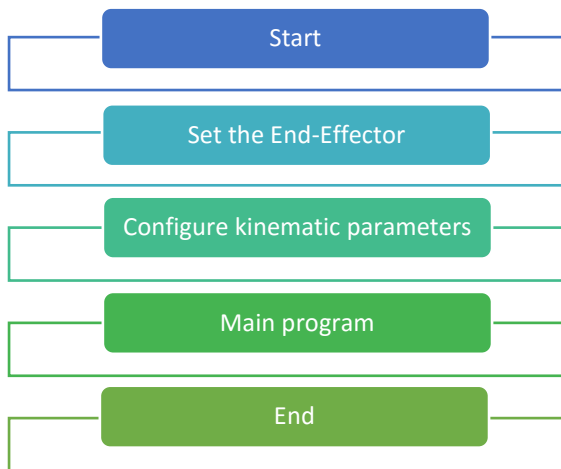
How to control Dobot by DobotBlockly?



It is very easy to Blockly programming, just two steps:

Step1: Develop the functions that the program need to be achieved;

Step2: Drag the module to organize the procedure and achieve the goal, shown as follows:



The general flowchart of Dobot programming design

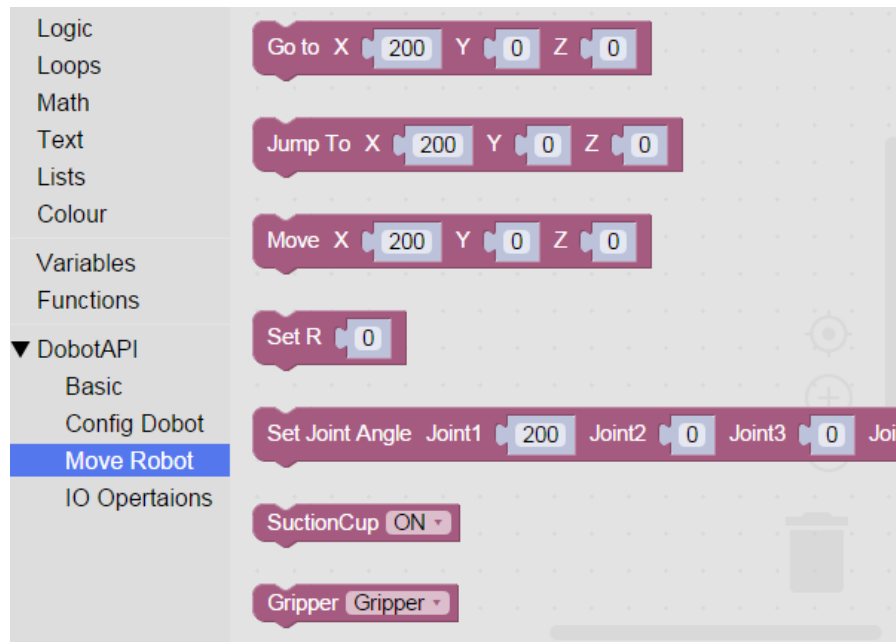
How to control Dobot by DobotBlockly?



Being familiar with the flowchart, here lets learn Dobot exclusive API.

What is the API? The API, called the application programming interface, is a pre-defined set of functions that provide developers with the ability to access a set of routines without accessing the source code or understanding the details of the internal working mechanism.

Take the human language to analogy, API is equivalent to idiom. The ancients have been summed up good idioms, such as "love me, love my dog", that is used, do not need more explanation. API is the same, is a good integration of a function, we do not know how the source code is written, how to achieve, as long as the call can output specific results, the robot is to achieve a specific action, a module an action, convenient and efficient.

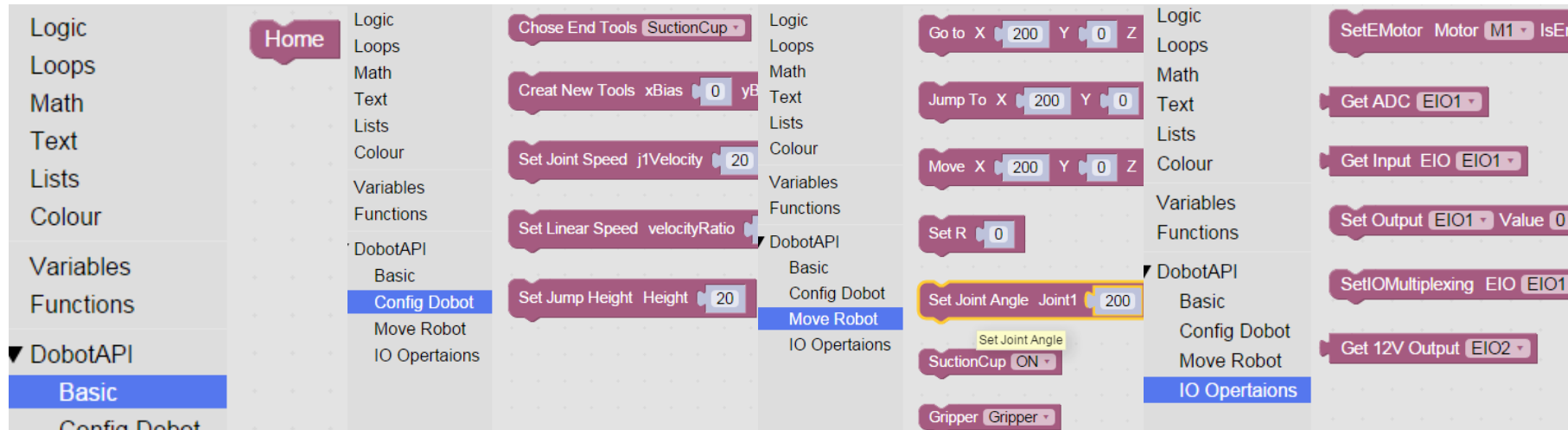


The first program to control Dobot



The exclusive API is divided into four areas:

- Basic: Homing module. The module can make the robot back to the origin preset;
- Config Dobot: The End-Effector of the robot arm, speed and acceleration settings;
- Move Robot : Controlled module, motion control of the robot arm, jog mode or coordinate system mode;
- IO Operations: IO operation module, advanced control of the expansion interface.



The first program to control Dobot



Now let's make the arms move!

Task: Let the robot suck up a block from point A to point B.

Open Blockly, drag the following procedure in the control module:

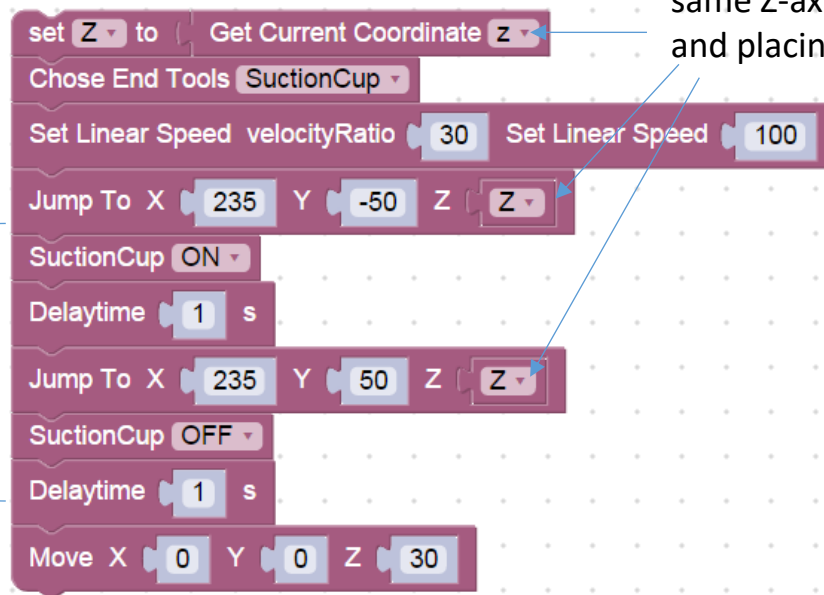
Get the Z-coordinate of the suction cup automatically as the suction height, and make sure that the same Z-axis height when picking and placing.

Set up endeffector

Config kinematic parameters

Main program: From point A to point B, suck up the block.

After execution, The Z axis rises 30mm back to the specified point



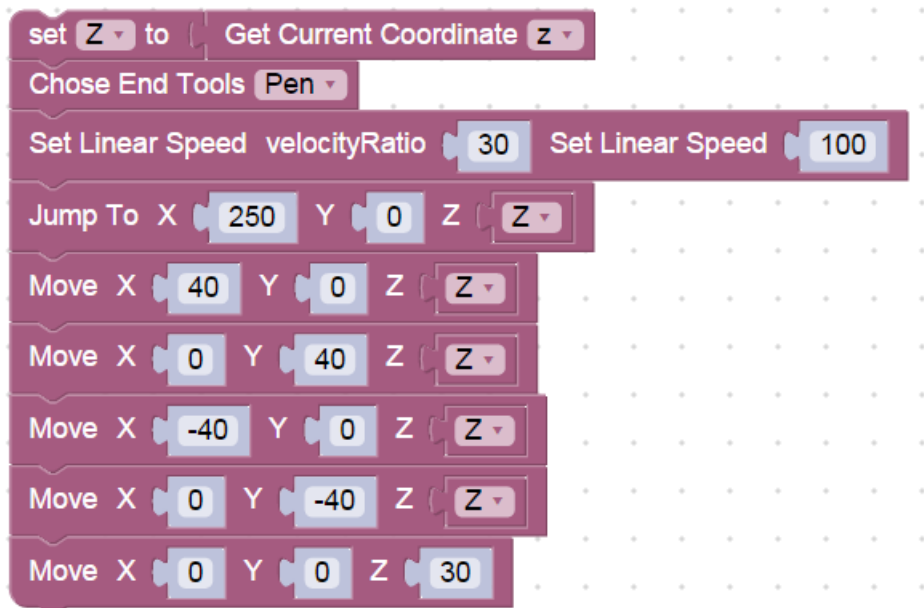
Demo 01

Notice: Before suctioning, place the suction cup at a suitable height above the building blocks so that the building blocks can be better sucked up. (The suction cup can be properly pressed with the building blocks)

Divergent thinking: Write your program

Task: Install the pen holder, and let Dobot draw a 40 * 40mm square.

Many ways to achieve, the reference procedure is as follows:



Demo 02

Note: Before writing, please place the pen on the appropriate writing height of the paper, so that it can write normally.

CONTACT US |



0755-33100907



sales@dobot.cc



www.dobot.cc



深圳市南山区
学苑大道1001号
南山智园 c2栋 18楼



THANK YOU

2116.11.11
