

# Table of Contents

Introduction	1.1
1. Overview	1.2
2. Introduction of Commands	1.3
Control Commands	1.3.1
2.1 Pause program command	1.3.1.1
2.2 Set the delay time command	1.3.1.2
2.3 Wait command	1.3.1.3
2.4 Custom Script	1.3.1.4
2.5 Get the current time	1.3.1.5
Variables Commands	1.3.2
2.6 Make a variable	1.3.2.1
2.7 Variable assignment	1.3.2.2
2.8 Modify variables	1.3.2.3
2.9 Make a list	1.3.2.4
2.10 Add a variable to the list	1.3.2.5
2.11 Deletes an item	1.3.2.6
2.12 Delete all items	1.3.2.7
2.13 Insert an item	1.3.2.8
2.14 Replace an item	1.3.2.9
2.15 Gets an item	1.3.2.10
2.16 Gets the total number of items	1.3.2.11
Motion Commands	1.3.3
2.17 Advanced configuration	1.3.3.1
2.18 Move command	1.3.3.2
2.19 Joint move command	1.3.3.3
2.20 Joint data command	1.3.3.4
2.21 Offset move command	1.3.3.5
2.22 Joint offset move command	1.3.3.6
2.23 Arc command	1.3.3.7
2.24 Circle command	1.3.3.8
2.25 Sync command	1.3.3.9
2.26 Set the continuous path rate	1.3.3.10
2.27 Set the joint velocity rate	1.3.3.11
2.28 Set the joint acceleration rate	1.3.3.12
2.29 Set the line motion velocity rate	1.3.3.13

2.30 Set the line motion acceleration rate	1.3.3.14
2.31 Modify the coordinate	1.3.3.15
2.32 Get the coordinate value	1.3.3.16
I/O Commands	1.3.4
2.33 Set digital output command	1.3.4.1
2.34 Set digital output immediately command	1.3.4.2
2.35 Wait digital input command	1.3.4.3
2.36 Set analog output command	1.3.4.4
2.37 Digital input command	1.3.4.5
2.38 Get analog input command	1.3.4.6
Modbus Commands	1.3.5
2.39 Creates a Modbus master command	1.3.5.1
2.40 Get the connection result command	1.3.5.2
2.41 Get input register command	1.3.5.3
2.42 Get holding register command	1.3.5.4
2.43 Get discrete input register command	1.3.5.5
2.44 Get coil register command	1.3.5.6
2.45 Get multiple coil register command	1.3.5.7
2.46 Set coil register command	1.3.5.8
2.47 Set multiple coil register values command	1.3.5.9
2.48 Set holding register command	1.3.5.10
2.49 Close Modbus master command	1.3.5.11
TCP Commands	1.3.6
2.50 Open socket command	1.3.6.1
2.51 Get open socket command	1.3.6.2
2.52 Create socket command	1.3.6.3
2.53 Get create socket command	1.3.6.4
2.54 Close socket command	1.3.6.5
2.55 Get variable command	1.3.6.6
2.56 Socket send variable command	1.3.6.7
2.57 Get socket send result command	1.3.6.8
Vision Command	1.3.7
2.58 Connect camera command	1.3.7.1
2.59 Get connect result command	1.3.7.2
2.60 Trigger the camera command	1.3.7.3
2.61 Send data to the camera command	1.3.7.4
2.62 Receive data from camera command	1.3.7.5
2.63 Get the number of data groups command	1.3.7.6
2.64 Gets a data command	1.3.7.7

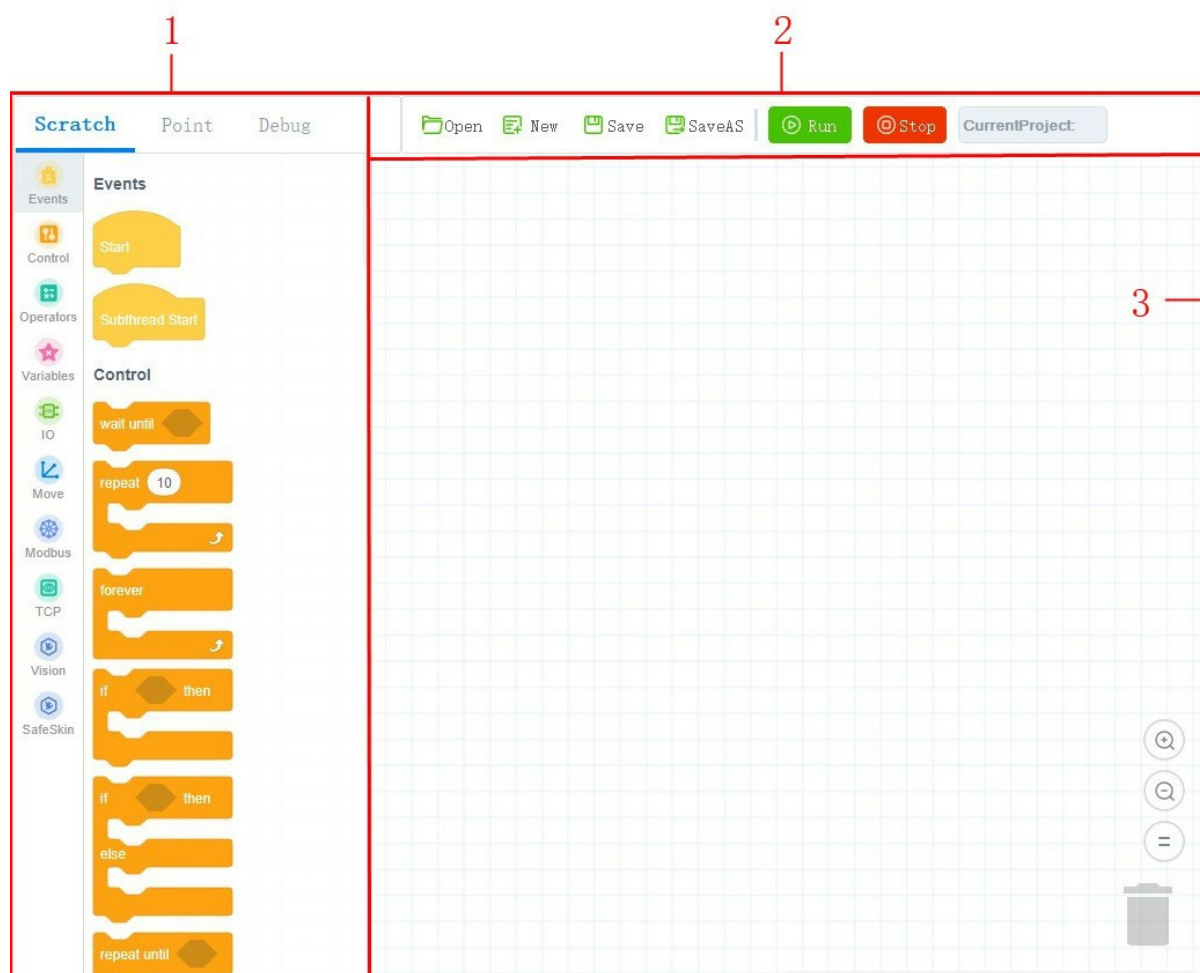
2.65 Close camera command	1.3.7.8
Pallet Command	1.3.8
2.66 Instantiate matrix pallet	1.3.8.1
2.67 Set the next stack index which is to be operated	1.3.8.2
2.68 Get the current operated stack index	1.3.8.3
2.69 Set the next pallet layer index which is to be operated	1.3.8.4
2.70 Get the current pallet layer index	1.3.8.5
2.71 Reset pallet	1.3.8.6
2.72 Check whether the stack assembly or dismantling is complete	1.3.8.7
2.73 Release palletizing instance	1.3.8.8
Safe Skin Command	1.3.9
2.74 Set the state of safe skin command	1.3.9.1
2.75 Set avoidance mode command	1.3.9.2
3. Description of Programming	1.4
3.1 Basic operation	1.4.1
3.2 Teaching points	1.4.2
3.3 Quick Start	1.4.3
3.3.1 Robot movement	1.4.3.1
3.3.2 I/O Setting	1.4.3.2
3.3.3 Register setting and reading	1.4.3.3
3.3.4 Create TCP Client	1.4.3.4
3.3.5 Create TCP Server	1.4.3.5
3.3.6 Vision Interaction	1.4.3.6





# Introduction






Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

# 1. Overview

Blockly is a kind of building block programming. You can write programs by block to quickly and conveniently control the robot. The blockly panel is shown in the following figure, and the description of blockly panel is listed in the following table.



No.	Description
1	<p>Block area</p> <p><b>Scratch</b> : Provide all blocks</p> <p><b>Point</b> : Save teaching point that can be called when writing a program</p> <p><b>Debug</b> : View the Lua instruction corresponding to the blocks</p>
2	<p>Menu bar</p> <p> <b>Open</b> : Open a project</p> <p> <b>New</b> : Create a new project</p> <p> <b>Save</b> : Save the project</p> <p> <b>SaveAS</b> : Save the current project with a new name</p>

	 : Start running the program in the current code area.  : Stop the running program
3	<p>Code area</p> <p>Drag block to this page and edit it. Click the icon   in the code area to zoom in, zoom out and restore the blocks,  can be used to delete the selected block.</p>

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 18:07:02

## 2. Introduction of Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

# Control Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-17 14:39:09



## 2.1 Pause program command

- Function:



- Description: When the program runs to this command, pause the running program.
- Parameter: None
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:00:35

## 2.2 Set the delay time command

- Function:

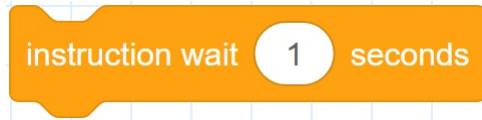


- Description: Set the delay time for all commands.
- Parameter: Time: Delay time; Unit: second
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-28 17:12:36

## 2.3 Wait command

- Function:

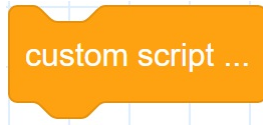


- Description: Set the delay time for robot motion commands.
- Parameter: time: Delay time; Unit: second
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-28 17:12:42

## 2.4 Custom Script

- Function:



- Description: Double-click the block, a script editing window will pop up. You can write Lua programs as required. After writing, click "Save".
- Parameter: None
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-28 17:12:50

## 2.5 Get the current time

- Function:



- Description: Get the current time.
- Parameter: None
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:14:51

# Variables Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.6 Make a variable

- Function:

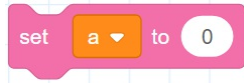


- Description: Make a variable.
- Parameter: Set variable name. The variable name must start with a letter and cannot contain special characters such as Spaces.
- Return: a variable

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:18:08

## 2.7 Set the value of a variable

- Function:



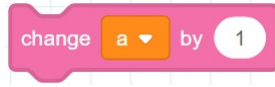
- Description: Set the value of a variable.
- Parameter:
  - Name: name of a variable
  - parameter: value of a variable
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:18:27



## 2.8 Modify the value of a variable

- Function:

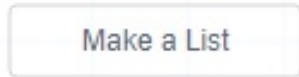


- Description: Modify the value of a variable.
- Parameter:
  - Name: name of a variable.
  - parameter: The value of an increase or decrease.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:19:09

## 2.9 Make a list

- Function:

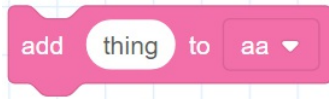


- Description: Make a list. When there are many variables, you can use lists to manage variables.
- Parameter: Set list name. The list name must start with a letter and cannot contain special characters such as Spaces.
- Return: a list

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:19:21

## 2.10 Add a variable to the list

- Function:

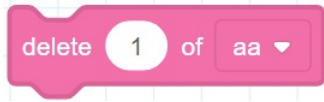


- Description: Add a variable to the list.
- Parameter:
  - variable: Set a variable.
  - list: Select a list.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:20:05

## 2.11 Deletes an item

- Function:



- Description: Delete an item from the list.
- Parameter:
  - Index: Set specified item.
  - list: Select a list.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:20:25

## 2.12 Delete all items

- Function:



- Description: Delete all items.
- Parameter: Select a list.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:21:35

## 2.13 Insert an item

- Function:

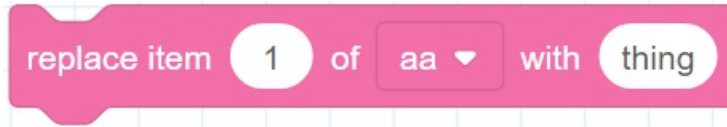


- Description: Inserts an item before an item in the list.
- Parameter:
  - Content: Set the content to be inserted.
  - Index: Set the specified item.
  - List: Select a list.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:22:18

## 2.14 Replace an item

- Function:



- Description: Replace the contents of an item in the list.
- Parameter:
  - Index: Set the specified item.
  - List: Select a list.
  - Content: Set the content to be inserted.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:22:50

## 2.15 Gets an item

- Function:



- Description: Gets a variable in the list.
- Parameter:
  - Index: Set the specified item.
  - List: Select a list.
- Return: a variable

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:23:23



## 2.16 Gets the total number of items

- Function:



- Description: Gets the total number of items.
- Parameter: Select a list.
- Return: Total number of items.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:00

# Motion Commands

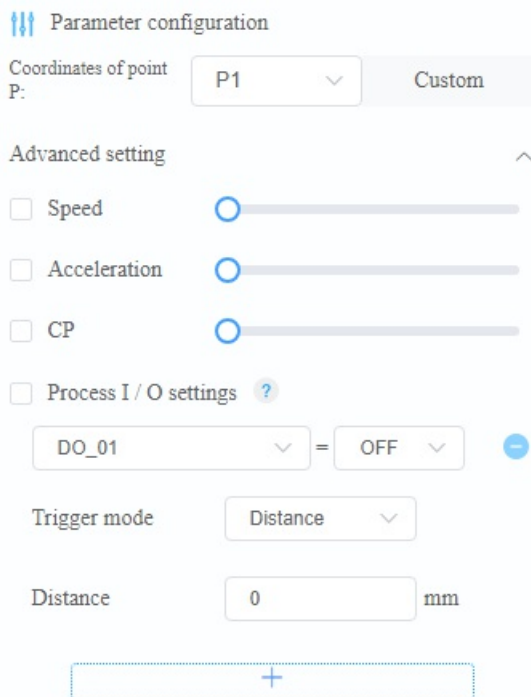
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.17 Advanced configuration

- Function:



- Description: Click Advanced configuration, on the displayed **Settings panel**, select a motion type and set basic and advanced parameters (optional).
- Type 1: MovJ: Move from the current position to a target position in a point-to-point mode under the Cartesian coordinate system.



- Parameter configuration
  - Coordinate of point P: Indicate target point, which is user-defined or obtained from the **Point** page. Only Cartesian coordinate points are supported.
- Advanced setting
  - Speed: Velocity rate. Value range: 1 - 100
  - Acceleration: Acceleration rate. Value range: 1 - 100
  - CP: Whether to set continuous path function. Value range: 0 - 100
  - Process I/O settings
    - DO: Output port of CR
    - Trigger mode: Distance or Percentage. Distance refers to the distance away from the starting point or target point. Percentage refers to the distance percentage between the starting point and the target point. range: 0~100
    - Distance: If the Distance value is positive, it refers to the distance away from the starting point; If the Distance value is negative, it refers to the distance away from the target point.
- Type 2: MovL: Move from the current position to a target position in a straight line under the Cartesian coordinate system.

Parameter configuration

Coordinates of point P:

Advanced setting ^

Speed

Acceleration

CP

Process I / O settings ?

=

Trigger mode

Distance  mm

- o Parameter configuration
  - Coordinate of point P: Indicate target point, which is user-defined or obtained from the **Point** page. Only Cartesian coordinate points are supported.
- o Advanced setting
  - Speed: Velocity rate. Value range: 1 - 100
  - Acceleration: Acceleration rate. Value range: 1 - 100
  - CP: Whether to set continuous path function. Value range: 0 - 100
  - Process I/O settings
    - DO: Output port of CR
    - Trigger mode: Distance or Percentage. Distance refers to the distance away from the starting point or target point. Percentage refers to the distance percentage. between the starting point and the target point. range: 0~100.
    - Distance: If the Distance value is positive, it refers to the distance away from the starting point; If the Distance value is negative, it refers to the distance away from the target point.
- Type 3: Jump: The robot moves from the current position to a target position in the **Move** mode. The trajectory looks like a door.

Parameter configuration

Coordinates of point P:

Raise height h1  mm

Descent height h2  mm

Max height z\_limit  mm

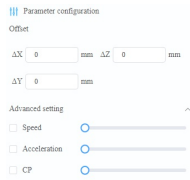
Advanced setting ^

Speed

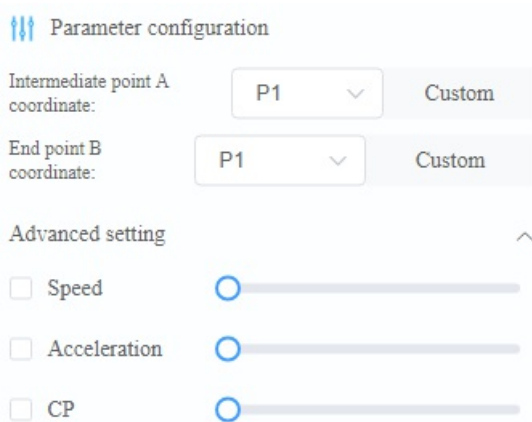
Acceleration

- o Parameter configuration





- Parameter configuration
    - Offset: X, Y, Z axes offset in the Cartesian coordinate system. Unit: mm
  - Advanced setting
    - Speed: Velocity rate. Value range: 1 - 100
    - Acceleration: Acceleration rate. Value range: 1 - 100
    - CP: Whether to set continuous path function. Value range: 0- 100
- Type 7: Arc: Move from the current position to a target position in an arc interpolated mode under the Cartesian coordinate system. This command needs to combine with other motion commands, to obtain the starting point of an arc trajectory.



- Parameter configuration
    - Intermediate point A coordinate: Middle point, which is user-defined or obtained from the **Point** page.
    - End point B coordinate: End point, which is user-defined or obtained from the **Point** page.
  - Advanced setting
    - Speed: Velocity rate. Value range: 1 - 100
    - Acceleration: Acceleration rate. Value range: 1 - 100
    - CP: Whether to set continuous path function. Value range: 0 - 100
- Type 8: Circle: Move from the current position to a target position in a circular interpolated mode under the Cartesian coordinate system. This command needs to combine with other motion commands, to obtain the starting point of an arc trajectory.

Parameter configuration

Intermediate point A coordinate:

End point B coordinate:

Number of cycles:

Advanced setting ^

Speed

Acceleration

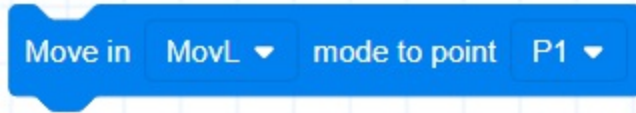
CP

- Parameter configuration
  - Intermediate point A coordinate: Middle point, which is user-defined or obtained from the **Point** page.
  - End point B coordinate: End point, which is user-defined or obtained from the **Point** page.
- Advanced setting
  - Speed: Velocity rate. Value range: 1 - 100
  - Acceleration: Acceleration rate. Value range: 1 - 100
  - CP: Whether to set continuous path function. Value range: 0 - 100

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:09

## 2.18 Move command

- Function:



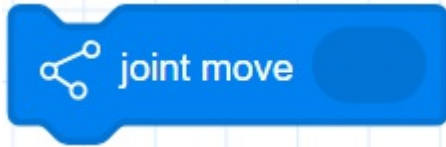
- Description: Move from the current position to a target position in straight line mode or point to point mode.
- Parameter:
  - Mode:
    - MovL: straight line mode
    - MovJ: point to point mode
  - P1: Indicate target point, which is obtained from the TeachPoint page.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:17



## 2.19 Joint move command

- Function:



- Description: Move from the current position to a target position in point to point mode.
- Parameter: Indicate the joint angle of the target position, the joint angles is set by Joint data command.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:27

## 2.20 Joint data command

- Function:

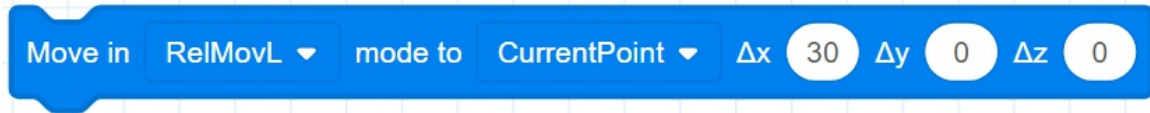
```
joint data J1 0 J2 0 J3 0 J4 0 J5 0 J6 0
```

- Description: Set the joint angles.
- Parameter: joint1~ joint 6: Indicate joint angle of J1 - J6 axes.
- Return: Joint angles.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:42

## 2.21 Offset move command

- Function:



- Description: Move the corresponding offset in X, Y and Z directions from the current position in straight line mode or point to point mode.
- Parameter:
  - Mode:
    - RelMovL: straight line mode
    - RelMovJ: point to point mode
  - $\Delta x$ : Indicate offset of X axis
  - $\Delta y$ : Indicate offset of Y axis
  - $\Delta z$ : Indicate offset of Z axis
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:24:51

## 2.22 Joint offset move command

- Function:

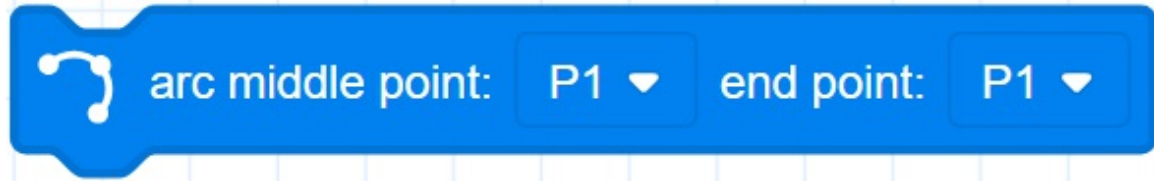


- Description: Move the joint offset in each axis from the current position in the Joint coordinate system.
- Parameter:  $\Delta J1 \sim \Delta J6$ : Indicate offset of J1 - J6 axes.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:04

## 2.23 Arc command

- Function:

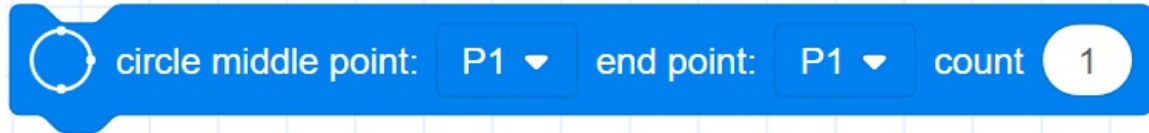


- Description: Move from the current position to a target position in an arc interpolated mode.
- Parameter:
  - middle point: Indicate middle point, which is obtained from the TeachPoint page.
  - end point: Indicate end point, which is obtained from the TeachPoint page.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:11

## 2.24 Circle command

- Function:



- Description: Move from the current position to a target position in a circular interpolated mode.
- Parameter:
  - middle point: Indicate middle point, which is obtained from the TeachPoint page.
  - end point: Indicate end point, which is obtained from the TeachPoint page.
  - count: number of whole circles, value range: 1 ~ 999.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:20

## 2.25 Sync command

- Function:

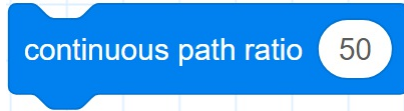


- Description: Block the program from executing queue instructions, and return after all the instructions are executed.
- Parameter: None
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:30

## 2.26 Set the continuous path rate

- Function:



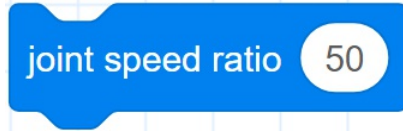
- Description: Set the continuous path rate.
- Parameter: Continuous path ratio, range: 0~100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:36



## 2.27 Set the joint velocity rate

- Function:



- Description: Set the joint velocity rate.
- Parameter: Joint speed ratio, range: 0 ~ 100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:47

## 2.28 Set the joint acceleration rate

- Function:

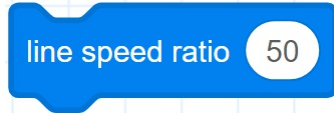


- Description: Set the joint acceleration rate.
- Parameter: Joint accel ratio, range: 0 ~ 100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:25:58

## 2.29 Set the line motion velocity rate

- Function:

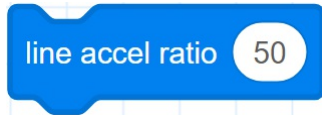


- Description: Set the line motion velocity rate.
- Parameter: Line speed ratio, range: 0~100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:26:10

## 2.30 Set the line motion acceleration rate

- Function:

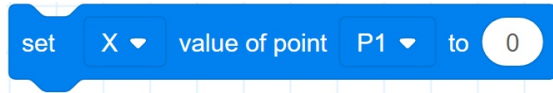


- Description: Set the line motion acceleration rate.
- Parameter: Line accel ratio, range: 0 ~ 100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:26:23

## 2.31 Modify the coordinate

- Function:

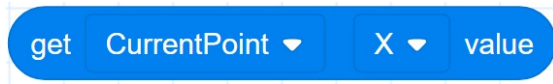


- Description: Modify the coordinate value of the teaching point.
- Parameter:
  - Coordinate axis: X, Y, Z, Rx, Ry, Rz.
  - P1: Indicate target point, which is obtained from the **Point** page.
  - Value: coordinate value.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:27:06

## 2.32 Get the coordinate value

- Function:



- Description: Get the coordinate value of the teaching point or current point.
- Parameter:
  - Point: teaching point or current point.
  - Coordinate axis: X, Y, Z, Rx, Ry, Rz.
- Return: None

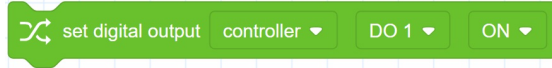
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:27:47

# I/O Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.33 Set digital output command

- Function:



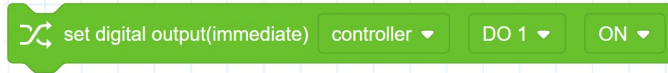
- Description: Set the status of digital output port.
- Parameter:
  - Control end: controller or tool.
  - DO: Digital output index. When you select controller, the value range is DO1~DO16; when you select tool, the value range is DO1~DO2.
  - Status: set the DO to on or off.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:28:14



## 2.34 Set digital output immediately command

- Function:

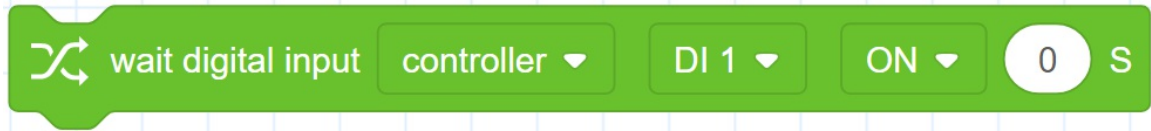


- Description: Without entering the queue command, set the status of digital output port immediately after pre-reading.
- Parameter:
  - Control end: controller or tool.
  - DO: Digital output index. When you select controller, the value range is DO1~DO16; when you select tool, the value range is DO1~DO2.
  - Status: set the DO to on or off.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:28:31

## 2.35 Wait digital input command

- Function:

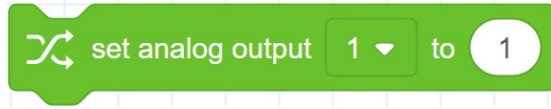


- Description: If any of the following conditions are met, the program continues to execute:
  - The status of DI is the same as the status set by the instruction.
  - The status of DI is different from the state set by the instruction, while the waiting time exceeds the preset time.
- Parameter:
  - Control end: controller or tool
  - DI: digital input index. When you select controller, the value range is DI1~DI32; when you select tool, the value range is DI1~DI2.
  - Status: Indicate status of DI.
  - Time: Set the waiting time, if the waiting time value is 0, it will wait until the condition is met.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:28:38

## 2.36 Set analog output command

- Function:

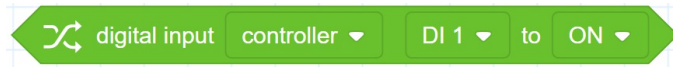


- Description: Set the value of analog output port.
- Parameter:
  - Port: analog output index.
  - Parameters: the value of the analog output.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:28:51

## 2.37 Digital input command

- Function:



- Description: Get the current I/O status and make judgments, which can be used as judgment conditions in a certain instruction.
- Parameter:
  - Control end: controller or tool.
  - DI: digital input index. When you select controller, the value range is DI1~DI32; when you select tool, the value range is DI1~DI2.
  - Status: indicate status of DI.
  - Time: set the waiting time, if the waiting time value is 0, it will wait until the condition is met.
- Return: The judgment result: the condition is true or not.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:29:06

## 2.38 Get analog input command

- Function:



- Description: Get the value of analog input port.
- Parameter:
  - Control end: controller or tool.
  - AI: analog input index.
- Return: The value of analog input port.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:29:35

# Modbus Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.39 Creates a Modbus master command

- Function:




- Description: Creates a Modbus master to connect to a Modbus slave.
- Parameter:
  - IP: indicate IP address of the Modbus slave.
  - Port: indicate the port of the Modbus slave station.
  - ID: indicate ID of the Modbus slave, the value range is 1 to 4.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:29:45

## 2.40 Get the connection result command

- Function:



get create modbus master result

- Description: Get the connection result.
- Parameter: None.
- Return:
  - 0: Modbus master is created successfully.
  - 1: Modbus master is created failed.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:29:53



## 2.41 Get input register command

- Function:

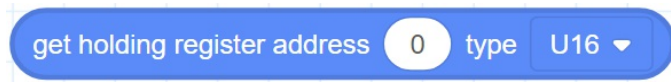


- Description: Read the input register value with the specified data type from the Modbus slave.
- Parameter:
  - Address: Starting address of the input registers. Value range: 0 – 4095.
  - type: Data type
    - “U16”: Read 16-bit unsigned integer ( two bytes, occupy one register)
    - “U32”: Read 32-bit unsigned integer (four bytes, occupy two registers)
    - “F32”: Read 32-bit single-precision floating-point number (four bytes, occupy two registers)
    - “F64”: Read 64-bit double-precision floating-point number (eight bytes, occupy four registers).
- Return: The input register value.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:30:11

## 2.42 Get holding register command

- Function:



- Description: Read the holding register value from the Modbus slave according to the specified data type.
- Parameter:
  - Address: starting address of the holding registers. Value range: 0 - 4095.
  - type: Data type
    - "U16": Read 16-bit unsigned integer ( two bytes, occupy one register)
    - "U32": Read 32-bit unsigned integer (four bytes, occupy two registers)
    - "F32": Read 32-bit single-precision floating-point number (four bytes, occupy two registers)
    - "F64": Read 64-bit double-precision floating-point number (eight bytes, occupy four registers)
- Return: The holding register value.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:30:29

## 2.43 Get discrete input register command

- Function:

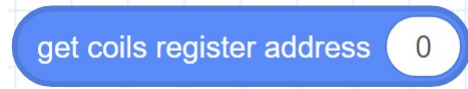
get discrete input register address 0

- Description: Read the discrete input register value from Modbus slave.
- Parameter: Address: starting address of the discrete inputs register. Value range: 0-4095.
- Return: The discrete input register value.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:30:45

## 2.44 Get coil register command

- Function:

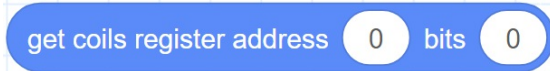


- Description: Read the coil register value from the Modbus slave.
- Parameter: Address: starting address of the coils register. Value range: 0 - 4095.
- Return: The coil register value.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:30:57

## 2.45 Get multiple coil register command

- Function:

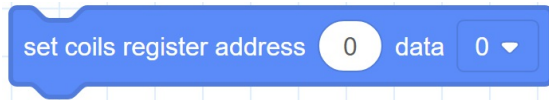


- Description: Read multiple coil register values from the Modbus slave.
- Parameter:
  - Address: starting address of the coils register. Value range: 0 – 4095.
  - Bits: Number of the coils to read. Value range: 0 to 4096- address.
- Return: The coil register values.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:31:08

## 2.46 Set coil register command

- Function:

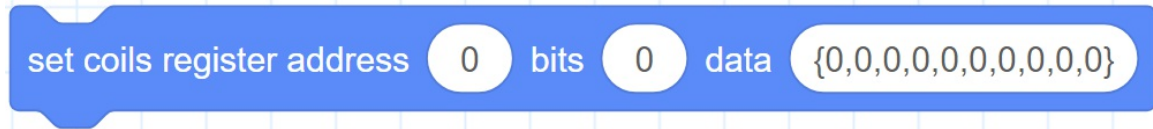


- Description: Set the coil register in the Modbus slave. This command is not supported when the coil register address is from 0 to 5.
- Parameter:
  - Address: Starting address of the coils register. Value range: 6 – 4095.
  - Data: the values written into the coil register.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:31:20

## 2.47 Set multiple coil register values command

- Function:

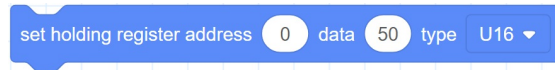


- Description: Set the multiple coil register values in the Modbus slave. This command is not supported when the coil register address is from 0 to 5.
- Parameter:
  - Address: Starting address of the coils register. Value range: 6 – 4095.
  - Bits: Number of the coils to read. Value range: 0 to 4096- address.
  - Data: the values written into the coil register.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:31:32

## 2.48 Set holding register command

- Function:



set holding register address 0 data 50 type U16 ▾

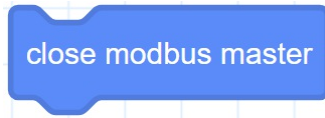
- Description: Set the holding register value in the Modbus slave.
- Parameter:
  - Address: Starting address of the holding registers to set. Value range: 0 – 4095.
  - Data: the values written into the holding register.
  - type: Data type
    - “U16”: Set 16-bit unsigned integer ( two bytes, occupy one register)
    - “U32”: Set 32-bit unsigned integer (four bytes, occupy two registers)
    - “F32”: Set 32-bit single-precision floating-point number (four bytes, occupy two registers)
    - “F64”: Set 64-bit double-precision floating-point number (eight bytes, occupy four registers).
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:31:45



## 2.49 Close Modbus master command

- Function:



- Description: Release a Modbus connection.
- Parameter: None
- Return: None

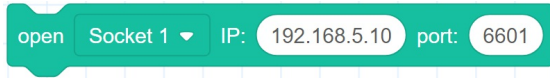
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:31:55

# TCP Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.50 Open socket command

- Function:



- Description: Create a TCP network, robot as a client.
- Parameter:
  - Socket: indicate Socket index, value range: Socket 1 to Socket 4.
  - IP: IP address of the server.
  - port: port of the server.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:32:08

## 2.51 Get open socket command

- Function:

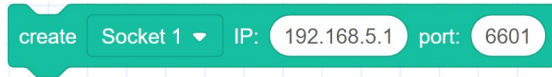


- Description: Get the connection result.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return:
  - 0: TCP connection is successful.
  - 1: Input parameters are incorrect.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:32:23

## 2.52 Create socket command

- Function:



- Description: Create a TCP network, robot as a server.
- Parameter:
  - Socket: indicate Socket index, value range: Socket 1 to Socket 4.
  - IP: address of the server.
  - port: Server port.

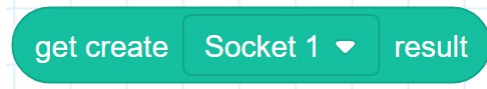
The port cannot be set to 502 and 8080. Otherwise, it will be in conflict with the Modbus default port or the port used in the conveyor tracking application, causing the creation to fail.

- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:32:42

## 2.53 Get create socket command

- Function:

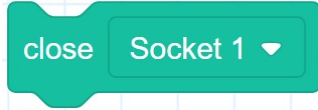


- Description: Get the connection result.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return:
  - 0: TCP network is created successfully.
  - 1: TCP network is created failed.
  - Socket: Socket object.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:32:56

## 2.54 Close socket command

- Function:

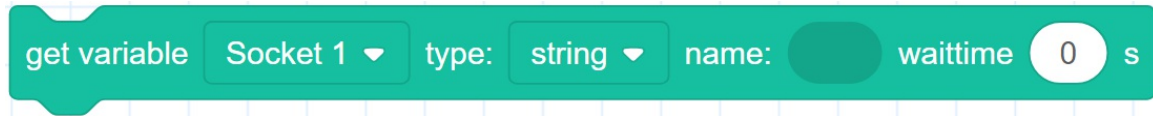


- Description: Release a TCP network.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:33:09

## 2.55 Get variable command

- Function:



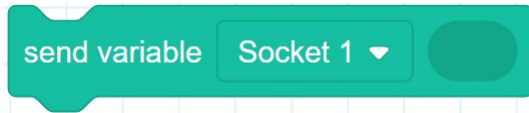
- Description: Obtain data through Socket communication.
- Parameter:
  - Socket: indicate Socket index, value range: Socket 1 to Socket 4.
  - Type: string or number.
  - Name: Variable used to hold data.
  - Waiting time: Set the waiting time, if the waiting time value is 0, it will wait until get data.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:33:23



## 2.56 Socket send variable command

- Function:

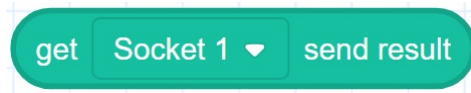


- Description: Send data through socket communication.
- Parameter:
  - Socket: indicate Socket index, value range: Socket 1 to Socket 4.
  - Variable: data to be sent.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:33:38

## 2.57 Get socket send result command

- Function:



- Description: Get the result of the data communication through the Socket.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return:
  - 0: Sending data is successful.
  - 1: Sending data is failed.

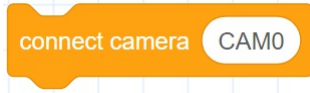
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:33:51

# Vision Command

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.58 Connect camera command

- Function:



- Description: Establish a communication connection with the camera.
- Parameter: Name of the camera.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:34:01

## 2.59 Get connect result command

- Function:

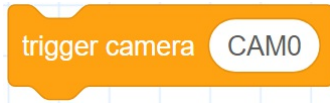
get connect camera result

- Description: Get the connection result.
- Parameter: None
- Return:
  - 0: Connect successfully
  - 1: Fail to Connect

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:34:09

## 2.60 Trigger the camera command

- Function:

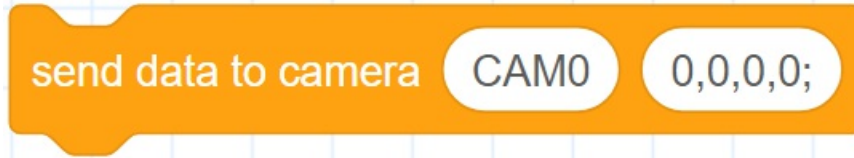


- Description: Trigger the camera to take a picture.
- Parameter: Name of the camera
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:34:20

## 2.61 Send data to the camera command

- Function:



- Description: Send data to the camera.
- Parameter:
  - Parameter 1: Camera name
  - Parameter 2: Data to be sent
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:34:28

## 2.62 Receive data from camera command

- Function:



- Description: Receive data from the camera.
- Parameter:
  - Parameter 1: Camera name.
  - Parameter 2: Data type, the value can be number or string.
- Return: The data sent by the camera.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:35:02



## 2.63 Get the number of data groups command

- Function:



camera data group

- Description: Gets the number of data groups sent by the camera.
- Parameter: None
- Return: The number of data groups.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:35:36

## 2.64 Gets a data command

- Function:

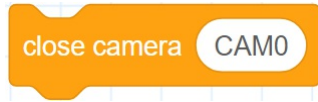
camera data at 1 group, at 1 item

- Description: Gets a data sent by the camera.
- Parameter:
  - Group: Specifies the group. Value range: 1 to the number of camera data groups.
  - Item: Select the item.
- Return: a data sent by the camera.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:35:53

## 2.65 Close camera command

- Function:



- Description: Release the connection.
- Parameter: None
- Return: None

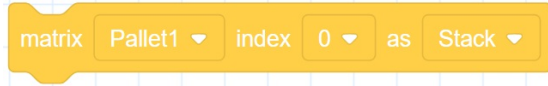
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:36:05

# Pallet Command

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 15:25:29

## 2.66 Instantiate matrix pallet

- Function:

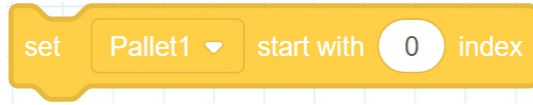


- Description: Instantiate matrix pallet.
- Parameter:
  - Pallet: Select a pallet from Pallet1 ~ Pallet4.
  - Index: Matrix pallet index.
  - IsUnstack: Stack mode. Value range: Stack, Assembly.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:38:19

## 2.67 Set the next stack index which is to be operated

- Function:

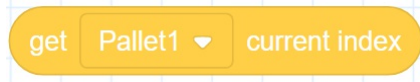


- Description: Set the next stack index which is to be operated.
- Parameter:
  - Pallet: Select a pallet from Pallet1 ~ Pallet4.
  - Index: The next stack index. Initial value: 0.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:38:53

## 2.68 Get the current operated stack index

- Function:

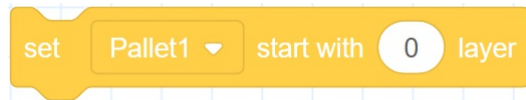


- Description: Get the current operated stack index.
- Parameter: Select a pallet from Pallet1 ~ Pallet4.
- Return: The current operated stack index.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:44:41

## 2.69 Set the next pallet layer index which is to be operated

- Function:



- Description: Set the next pallet layer index which is to be operated.
- Parameter:
  - Pallet: Select a pallet from Pallet1 ~ Pallet4.
  - Index: The next pallet layer index. Initial value: 0.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:45:51



## 2.70 Get the current pallet layer index

- Function:



- Description: Get the current pallet layer index.
- Parameter: Select a pallet from Pallet1 ~ Pallet4.
- Return: The current pallet layer index.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:45:45

## 2.71 Reset pallet

- Function:

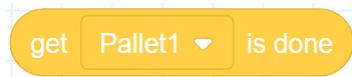


- Description: Reset pallet.
- Parameter: Select a pallet from Pallet1 ~ Pallet4.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 15:28:52

## 2.72 Check whether the stack assembly or dismantling is complete

- Function:

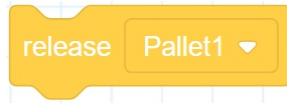


- Description: Check whether the stack assembly or dismantling is complete.
- Parameter: Select a pallet from Pallet1 ~ Pallet4.
- Return:
  - true: Finished.
  - false: Un-finished.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:45:32

## 2.73 Release palletizing instance

- Function:



- Description: Release palletizing instance.
- Parameter: Select a pallet from Pallet1 ~ Pallet4.
- Return: None

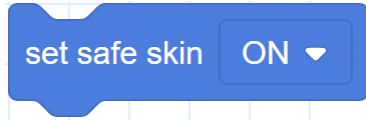
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:46:05

# Safe Skin Command

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45

## 2.74 Set the state of safe skin command

- Function:

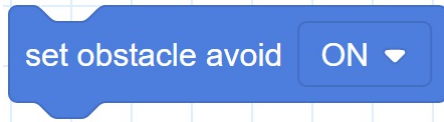


- Description: Set the state of safe skin.
- Parameter: safe skin status, 0: Turn off safe skin; 1: Open safe skin.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 12:46:16

## 2.75 Set avoidance mode command

- Function:



- Description: Set states of safe skin obstacle avoidance mode.

Tips: The obstacle avoidance function will make the robot move out of its original track, which may cause safety risks.

- Parameter: Status: states of safe skin obstacle avoidance mode, 0: Turn off safe skin obstacle avoidance mode; 1: Open safe skin obstacle avoidance mode.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-29 15:20:31

# 3 Description of Programming

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-14 17:18:45



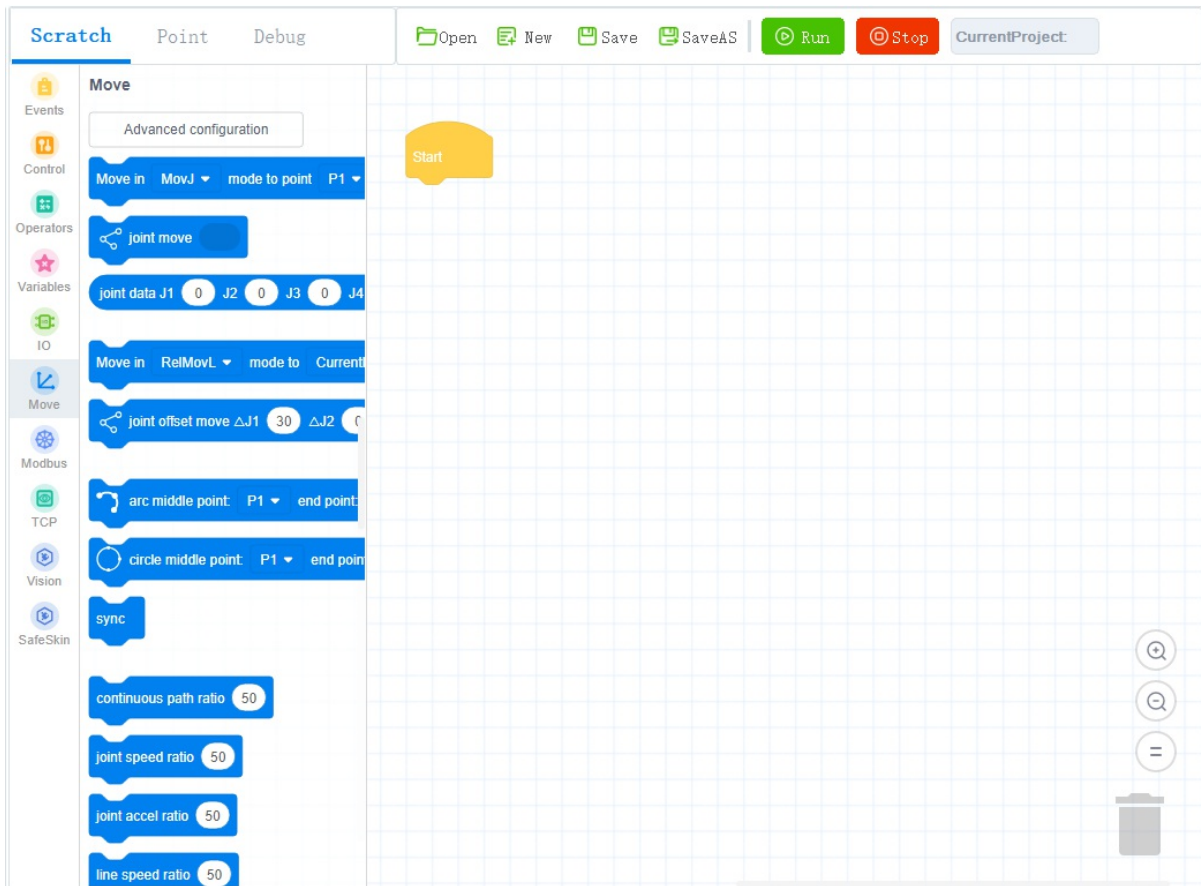
## 3.1 Basic operation

### Prerequisites

The robot has been powered on.

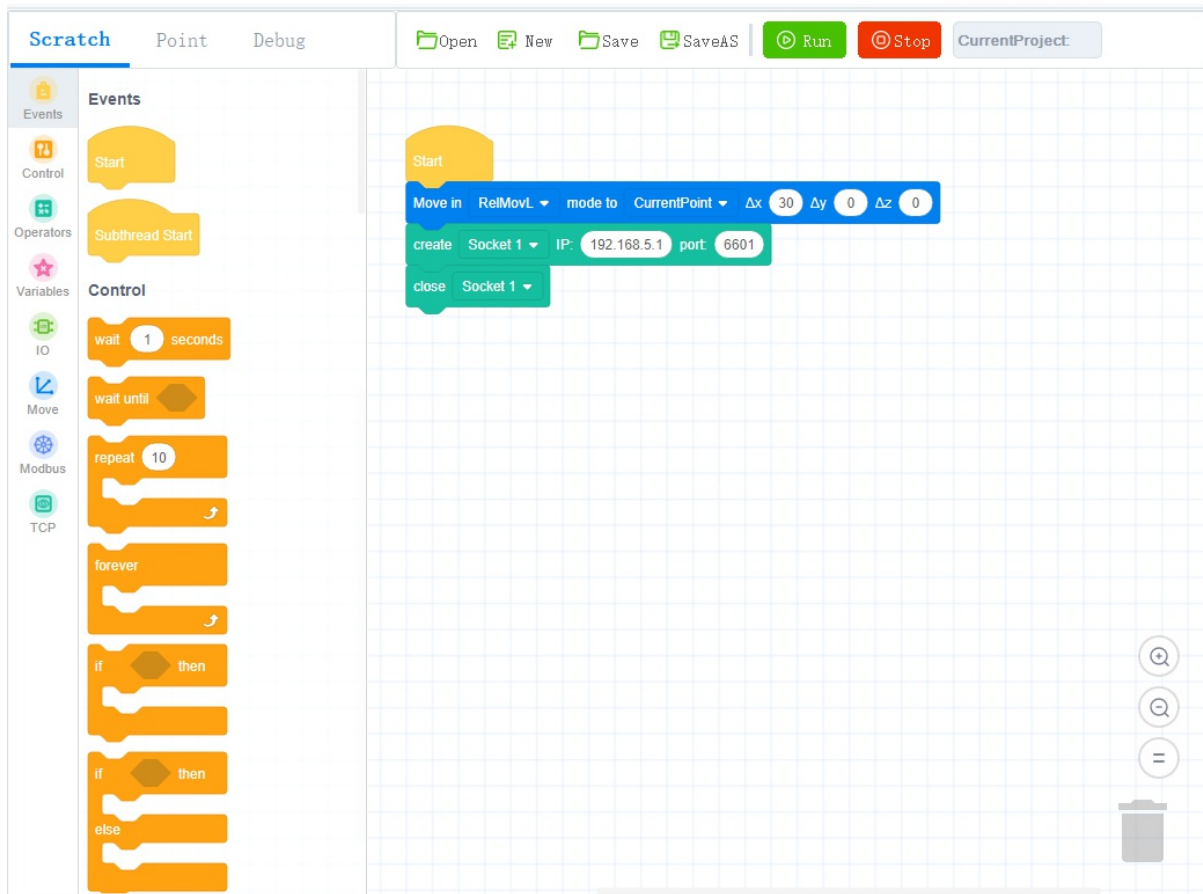
### Procedure


**Step 1** Enter the Blockly page. The system creates a new project by default.




**Step 2** Drag the blocks to the code area to start programming, as shown in following figure.


- Set the corresponding parameters of each block according to actual needs, for details see 2 Introduction.
- In the **point** page, you can save teaching point, when setting the parameters of the block, you can call the save point directly, for details see 3.2 Teaching points.



**Step 3** Click  Save to save the current project.

If it is the first time to save, you need to enter the project name.

**Step 4** Click  to enable the robotic arm.

**Step 5** Click  to run projects in the current code area.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-18 14:39:32

## 3.2 Teaching points


### Prerequisites

The project has been created or imported.

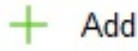
### Procedure


After creating a project, please teach positions on the **point** page for calling commands when programming a robot. If the existing taught positions list has been imported, this operation can be skipped.



**Step 1** Click  to enable the robotic arm.






**Step 2** Click **Jog** buttons to move the robot to a point.



**Step 3** Click **Point** to enter point page and click  **Add** to add a teaching point.

The teaching point information is displayed on the **point** page, as shown in the following figure.

No.	Alias	X	Y	Z	Rx	Ry	Rz	R	D	N	Cfg	Tool	User
1	P1	-0.0000	-247.5...	1050.5...	-90.0000	0.0000	180.0000	1	1	-1	-1	No.0	No.0
2	P2	-37.6501	-252.9...	1045.3...	-80.2021	31.1770	-151.9...	1	1	-1	1	No.0	No.0

Button	Description
 Add	Add a point
 Delete	Delete a point
 Cover	Cover a point. Select a teaching point, after jogging the robot to a point, click the icon to cover the selected teaching point
 RunTo	Run to a point, select a point, click the button to run the robot to this point
 Undo	Cancel

 Redo

Recover

- You can select a taught position and double-click the parameters on the line to modify the relevant information.

- Also, you can select a taught position and click  Cover to cover the current taught position.

**Step 4** Add points by referring to Step 2 and Step 3.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-17 14:38:07

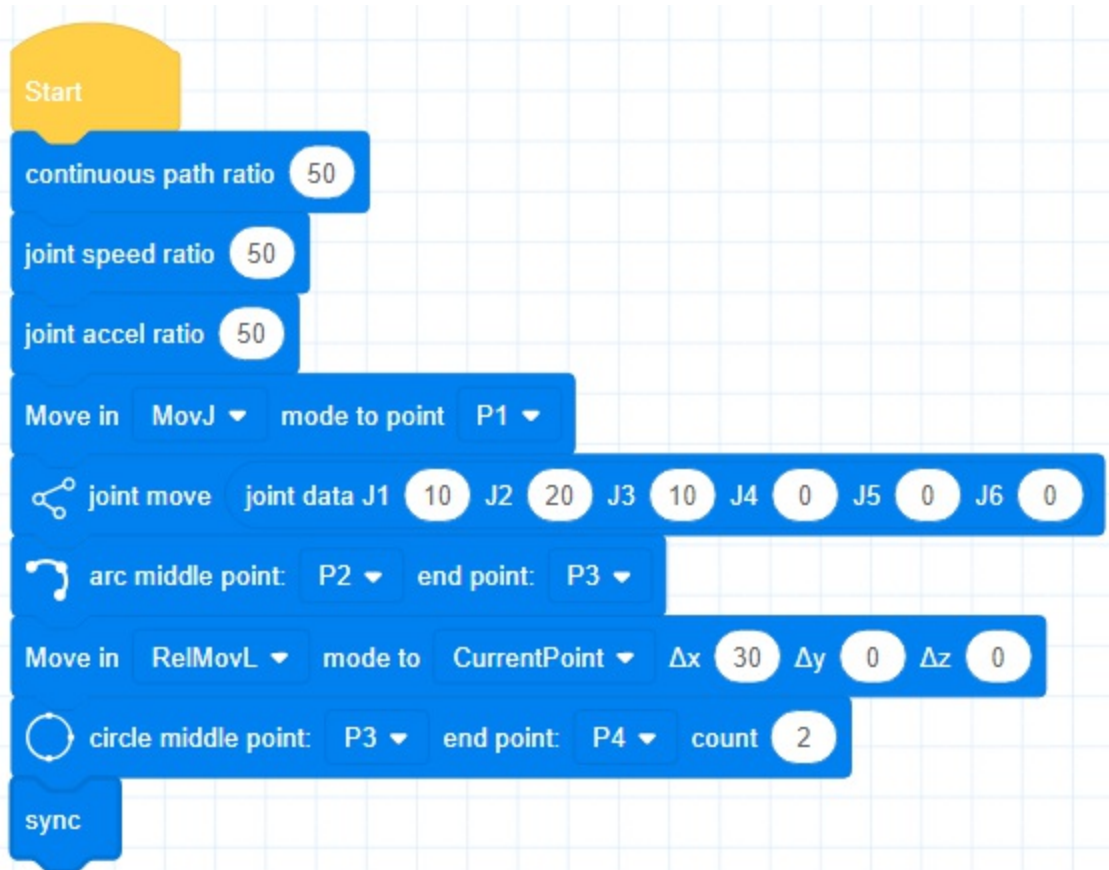
## 3.3 Quick Start

This section gives examples of Blockly for Motion commands, I/O commands, Modbus commands and TCP commands, for user reference only.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-15 15:19:33

### 3.3.1 Robot movement

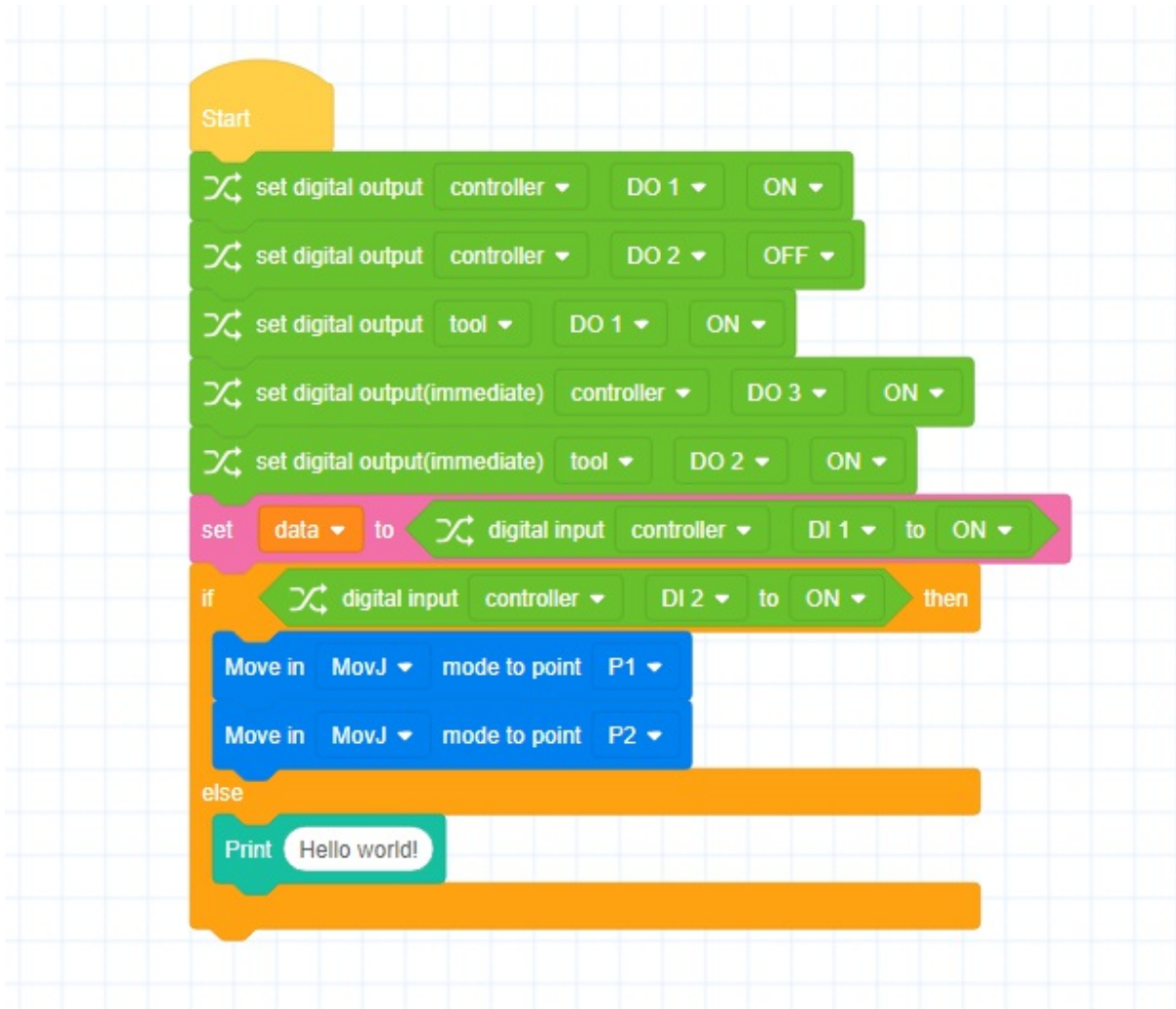
Running the Motion commands can control the movement of robot in the joint coordinate system and the Cartesian coordinate system. The detailed description of the Motion commands, please see **Motion Commands**. The following figure shows a programming program that includes Motion commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-15 15:23:55

### 3.3.2 I/O Setting

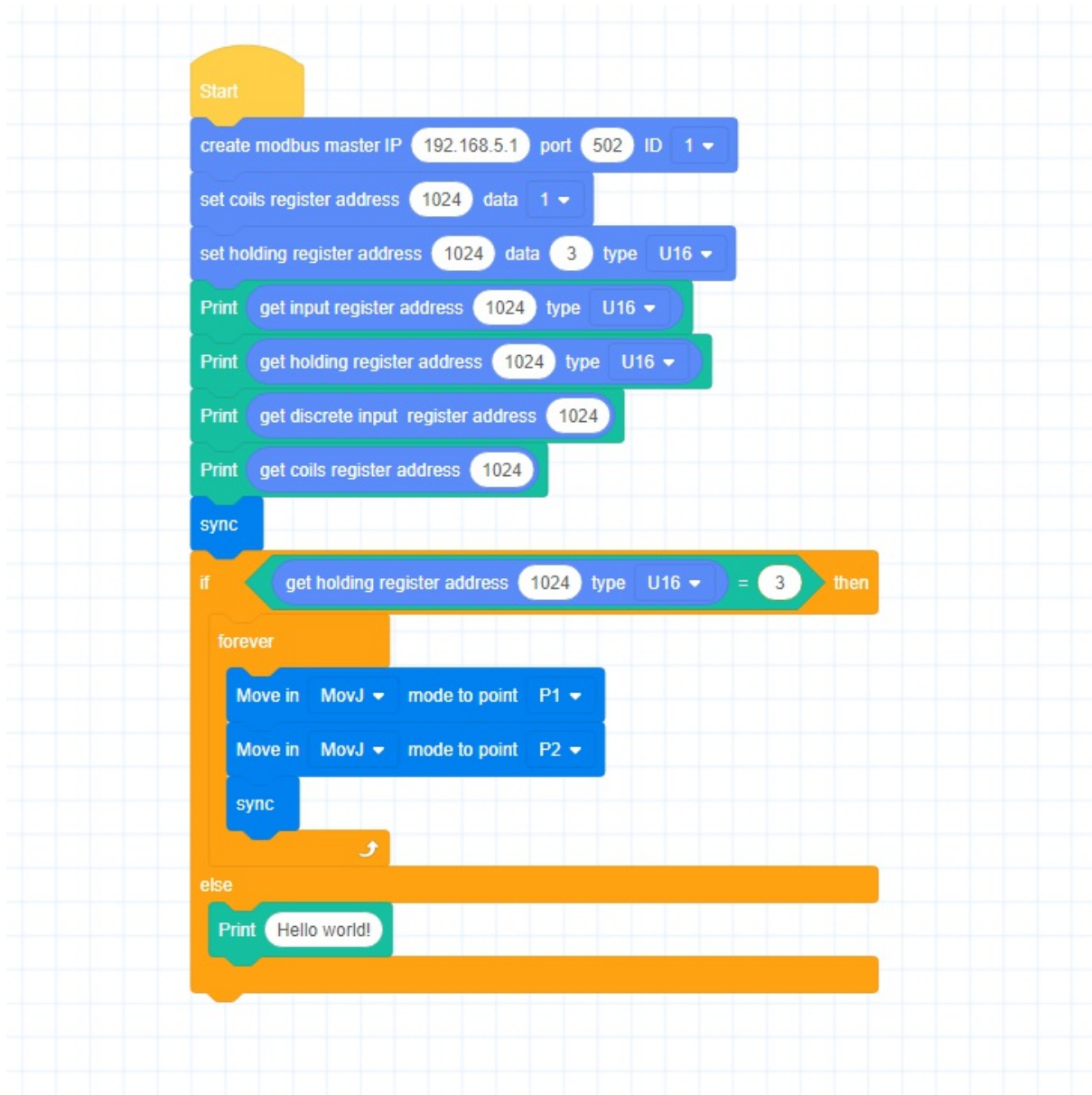
Running I/O commands to set or get each I/O statue. The detailed description of the I/O commands, please see **I/O Commands**. The following figure shows a programming program that includes I/O commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-17 15:03:40

### 3.3.3 Register setting and reading

By running the Modbus commands to set or read the value of each register address. The detailed description of the Modbus commands, please see **Modbus Commands**. The following figure shows a programming program that includes Modbus commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-15 15:29:10



### 3.3.4 Create TCP Client

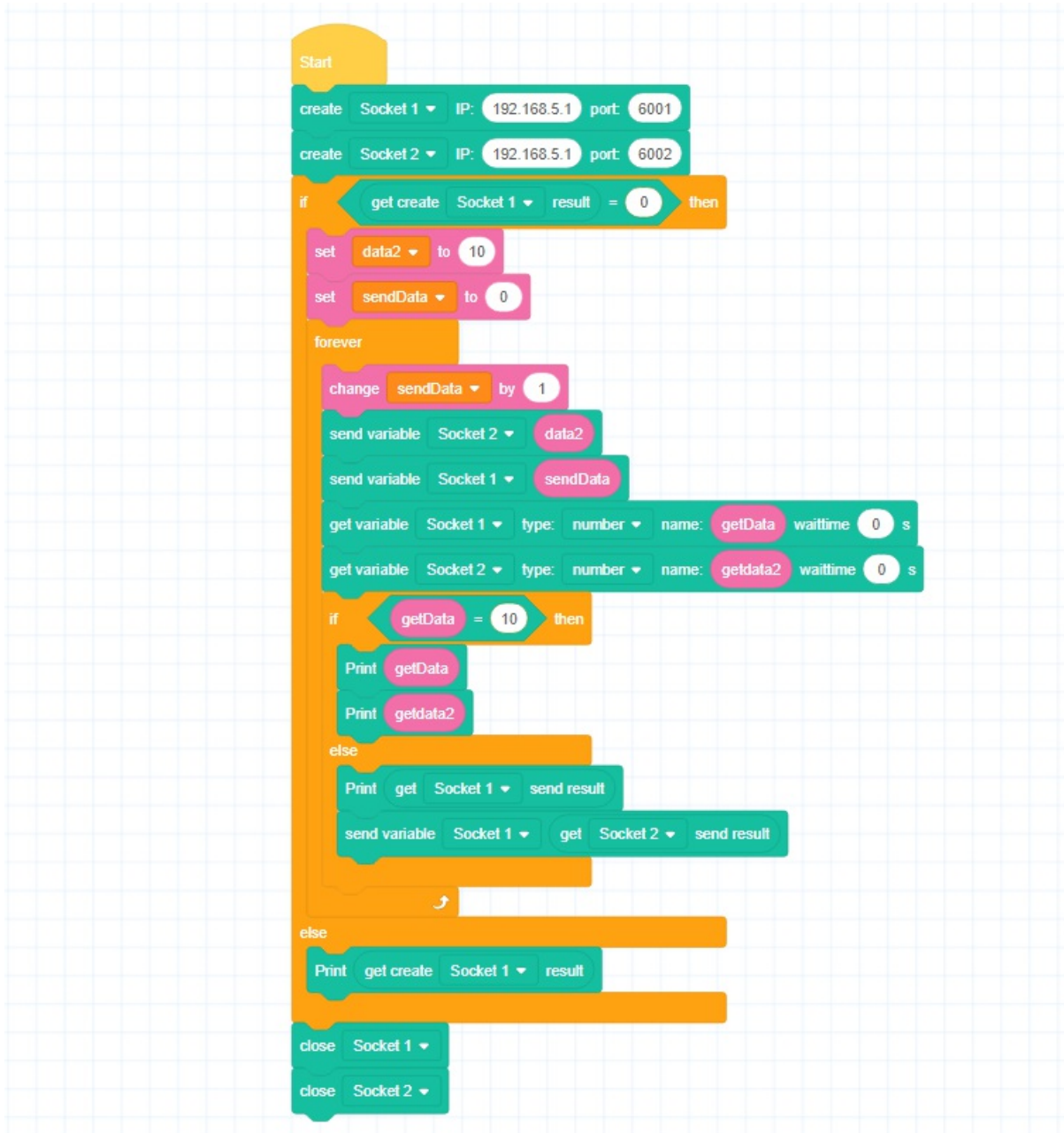
Run `open Socket 1 IP: 192.168.5.10 port: 6601` to establish communication with the TCP server, the robot as the TCP client. Running TCP commands can send and read communication data, the detailed description of TCP commands, please see **TCP Commands**. The following figure shows a programming program that includes TCP commands.

```
Start
open Socket 1 IP: 192.168.5.10 port: 6601
open Socket 2 IP: 192.168.5.12 port: 6002
if get open Socket 1 result = 0 then
  set data2 to 10
  set sendData to 0
  forever
    change sendData by 1
    send variable Socket 2 data2
    send variable Socket 1 sendData
    get variable Socket 1 type: string name: getData waittime 0 s
    get variable Socket 2 type: string name: getdata2 waittime 0 s
    if get Socket 1 send result = 0 then
      Print getData
      Print getdata2
    else
      Print get Socket 1 send result
      send variable Socket 1 get Socket 2 send result
  end
else
  Print get open Socket 1 result
close Socket 1
close Socket 2
```



### 3.3.5 Create TCP Server

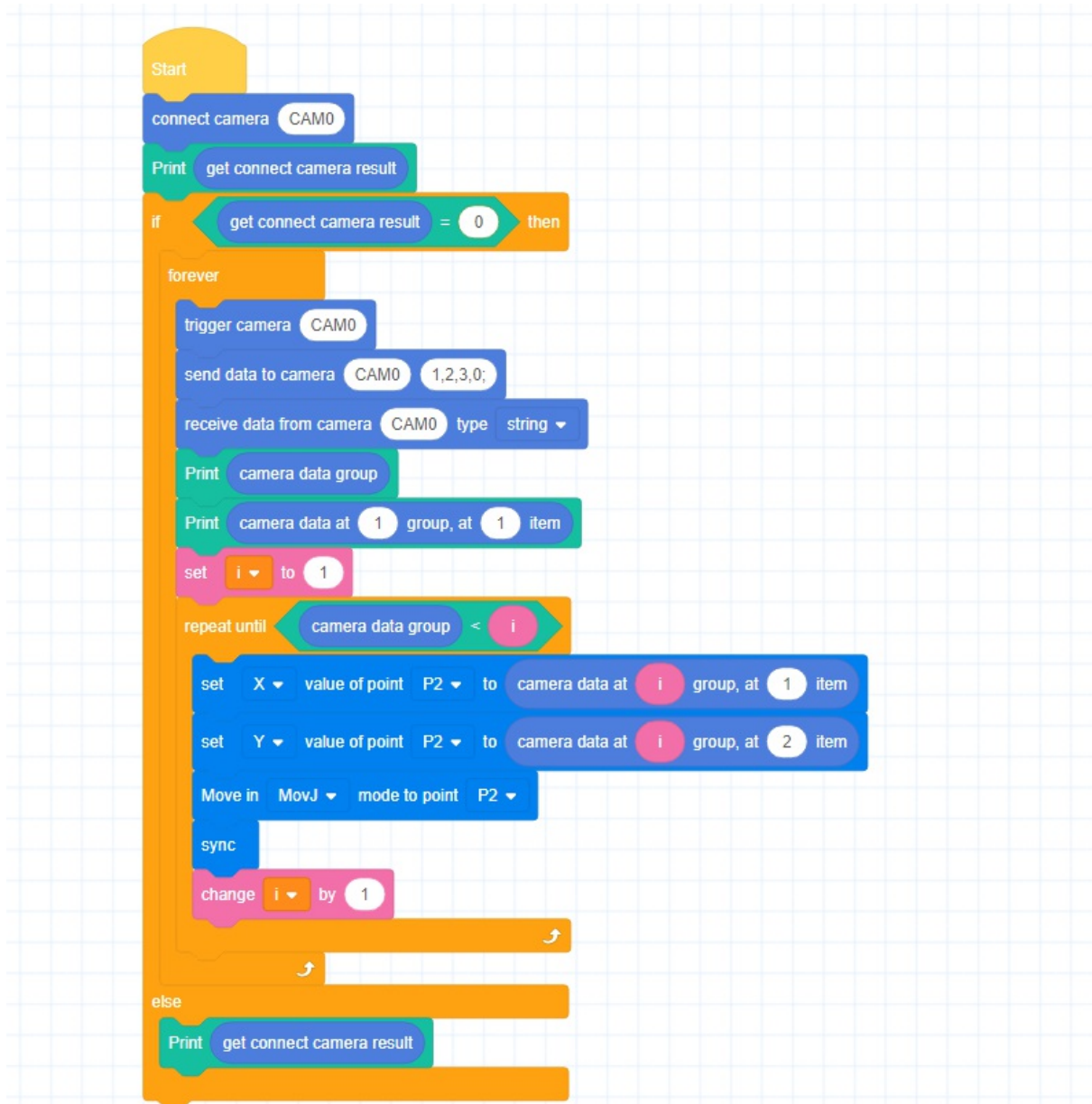
Run `create Socket 1 IP: 192.168.5.1 port: 6601` to set the robot as the server, waiting for the TCP client to connect. Running TCP instruction can send and read communication data, the detailed description of TCP commands, please see **TCP Commands**. The following figure shows a programming program that includes TCP commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-15 15:38:41

### 3.3.6 Vision Interaction

Running the vision instructions to establish communication with the camera, and to send and read the camera data. The detailed description of vision commands, please see **Vision Commands**. The following figure shows a programming program that includes vision commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision:  
2021-09-15 15:38:00