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 Get an item 1.3.2.10
  - 2.16 Get 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 Offset move command 1.3.3.3
  - 2.20 Jump Movement, Jump parameters can be set in this command 1.3.3.4
  - 2.21 Jump Movement, Jump parameters are called by Arch index 1.3.3.5
  - 2.22 Circle command 1.3.3.6
  - 2.23 Arc command 1.3.3.7
- Move Arguments Commands 1.3.4
  - 2.24 Set the joint acceleration rate 1.3.4.1
  - 2.25 Set the joint velocity rate 1.3.4.2
  - 2.26 Set the Cartesian acceleration rate 1.3.4.3
  - 2.27 Set the Cartesian velocity rate 1.3.4.4
  - 2.28 Set the continuous path rate 1.3.4.5

2.29 Stop at this point 1.3.4.6
2.30 Set Load Parameters 1.3.4.7
Posture Commands 1.3.5
2.31 Gets the value of the current Cartesian position 1.3.5.1
2.32 Gets a value of the current Cartesian position 1.3.5.2
2.33 Gets the value of the current Joint position 1.3.5.3
2.34 Gets a value of the current Joint position 1.3.5.4
2.35 Cartesian position offset 1.3.5.5
2.36 Angle position offset 1.3.5.6
2.37 Custom Cartesian point 1.3.5.7
2.38 Custom joint point 1.3.5.8
2.39 Get coordinates 1.3.5.9
2.40 Gets a coordinate value of a point 1.3.5.10
I/O Commands 1.3.6
2.41 Get digital input 1.3.6.1
2.42 Set digital output 1.3.6.2
2.43 Set digital output immediately command 1.3.6.3
Modbus Commands 1.3.7
2.44 Creates a Modbus master command 1.3.7.1
2.45 Get the connection result command 1.3.7.2
2.46 Get input register command 1.3.7.3
2.47 Get holding register command 1.3.7.4
2.48 Get discrete input register command 1.3.7.5
2.49 Get coil register command 1.3.7.6
2.50 Get multiple coil register command 1.3.7.7
2.51 Set coil register command 1.3.7.8
2.52 Set multiple coil register values command 1.3.7.9
2.53 Set holding register command 1.3.7.10
2.54 Close Modbus master command 1.3.7.11
TCP Commands 1.3.8
2.55 Open socket command 1.3.8.1
2.56 Get open socket command 1.3.8.2
2.57 Create socket command 1.3.8.3
2.58 Get create socket command 1.3.8.4
2.59 Close socket command 1.3.8.5
2.60 Get variable command 1.3.8.6
2.61 Socket send variable command 1.3.8.7
2.62 Get socket send result command 1.3.8.8
Vision Command 1.3.9

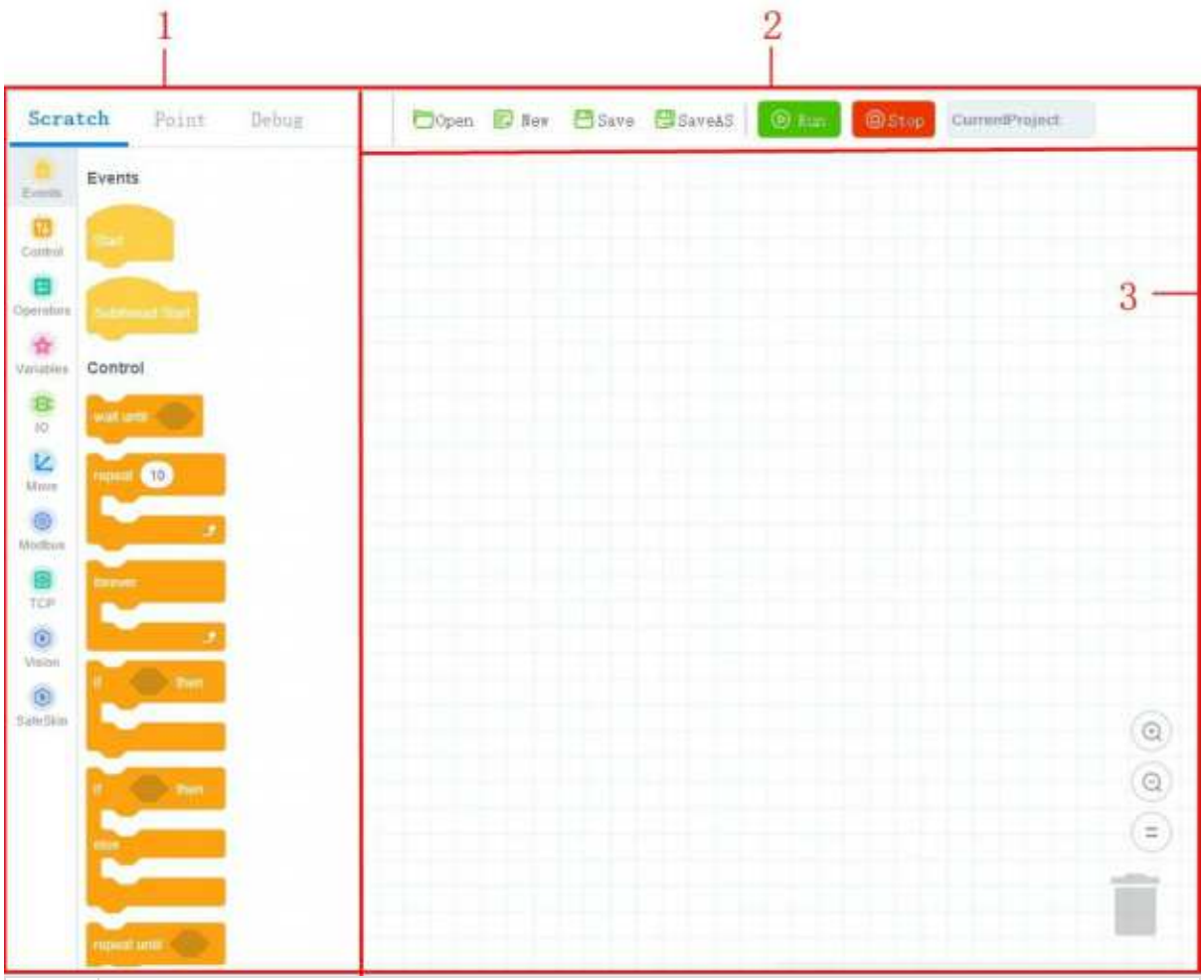
2.63 Connect camera command 1.3.9.1
2.64 Get connect result command 1.3.9.2
2.65 Trigger the camera command 1.3.9.3
2.66 Send data to the camera command 1.3.9.4
2.67 Receive data from camera command 1.3.9.5
2.68 Get the number of data groups command 1.3.9.6
2.69 Gets a data command 1.3.9.7
2.70 Close camera command 1.3.9.8
Pallet Command 1.3.10
2.71 Instantiate matrix pallet 1.3.10.1
2.72 Set the next stack index which is to be operated 1.3.10.2
2.73 Get the current operated stack index 1.3.10.3
2.74 Set the next pallet layer index which is to be operated 1.3.10.4
2.75 Get the current pallet layer index 1.3.10.5
2.76 Reset pallet 1.3.10.6
2.77 Check whether the stack assembly or dismantling is complete 1.3.10.7
2.78 Release palletizing instance 1.3.10.8
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: 2021-09-24 18:21:51

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	Block area Scratch :Provide all blocks Point : Save teaching point that can be called when writing a program Debug : View the Lua instruction corresponding to the blocks
2	Menu bar Open : Open a project New : Create a new project Save : Save the project SaveAS : Save the current project with a new name

	 : Start running the program in the current code area.  : Stop the running program
3	Code area 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.

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: 2021-09-28 11:16:07

Control Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

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: 2021-09-28 17:12:17

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: Gets the current time.
- Parameter: None
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:52:16

Variables Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

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: 2021-09-28 17:56:11

2.7 Set the value of a variable

- Function:

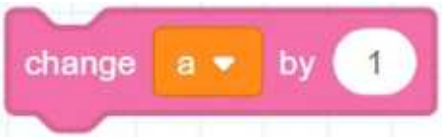


- Description: Set the value of a variable.
- Parameter:
  - Name: name of a variable
  - parameter: value of a variable
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 17:48:09

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: 2021-09-28 17:56:35

2.9 Make a list

- Function:

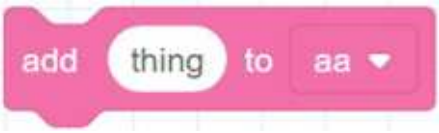


- Description: Make a list. When there are many variables, you can use lists to manage variables.
- Parameter: Set list name. The list name must start with a letter and cannot contain special characters such as Spaces.
- Return: a list

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 18:00:38

2.10 Add a variable to the list

- Function:

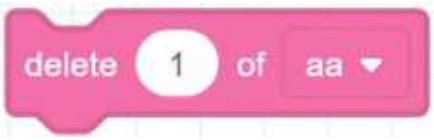


- Description: Add a variable to the list.
- Parameter:
  - variable: Set a variable
  - list: Select a list
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 18:10:01

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: 2021-09-28 18:14:16

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: 2021-09-28 18:15:18

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: 2021-09-28 18:22:43

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: 2021-09-28 18:22:35

2.15 Get an item

- Function:



- Description: Get a variable in the list.
- Parameter:
  - 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 14:53:04

2.16 Get the total number of items

- Function:



- Description: Get the total number of items.
- Parameter: Select a list.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:53:17

Motion Commands

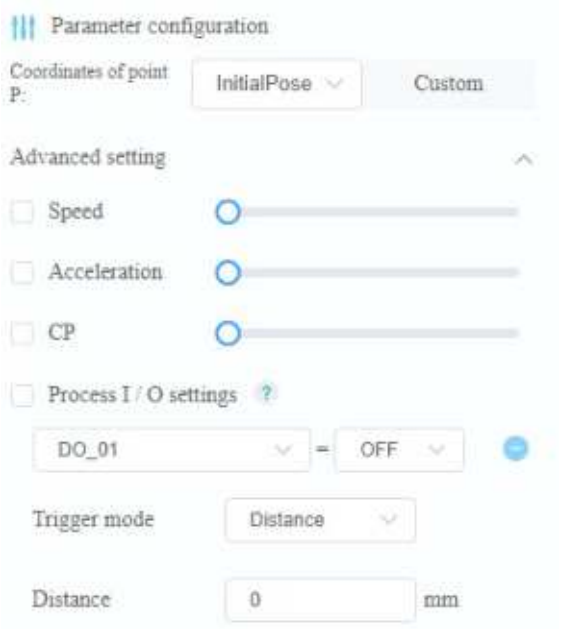
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.17 Advanced configuration

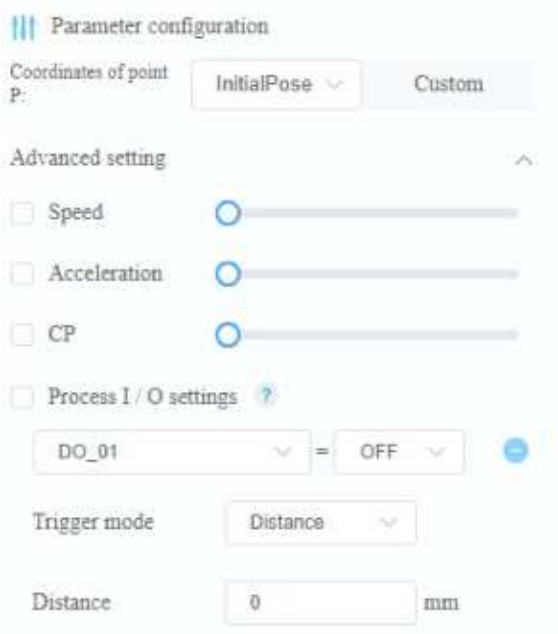
- Function:



- 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
  - 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 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
  - Coordinate of point P: Indicate target point, which is user-defined or obtained from the **Point** page. Only Cartesian coordinate points are supported.
  - Raise height h1: Lifting height
  - Descent height h2: Dropping height
  - Max height z_limit: Maximum lifting height
- Advanced setting
  - Speed: Velocity rate. Value range: 1 - 100
  - Acceleration: Acceleration rate. Value range: 1 - 100

- Type 4: JointMovJ: Move from the current position to a target position in a point-to-point motion under the Joint coordinate system.



- Parameter configuration
  - J1~J4: Indicate the joint angle of the target point
- 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 5: ReMovJ: Move from the current position to the offset position in a point-to-point mode under the Cartesian coordinate system.



- Parameter configuration
  - Offset: X, Y, Z, R 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 6: RelMovL: Move from the current position to the offset position in a straight line under the Cartesian coordinate system.



  - Parameter configuration
    - Offset: X, Y, Z, R 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.



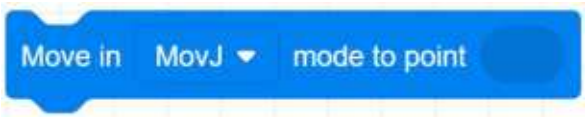
  - Parameter configuration
    - Intermediate point A coordinate: Middle point, which is user-defined or obtained from the **Point** page.

        - End point B coordinate: End point, which is user-defined or obtained from the **Point** page.
        - Number of cycles: Number of circles.
    - 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 15:11:14

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, the target point is the Cartesian point.
    - MovJ: point to point mode, the target point is the Cartesian point.
    - JointMovJ: point to point mode, the target point is the Cartesian point.
  - point: 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: 2021-09-28 18:41:23

2.19 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
  - △x: Indicate offset of X axis
  - △y: Indicate offset of Y axis
  - △z: Indicate offset of Z axis
  - △r: Indicate offset of R axis
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 18:45:37

2.20 Jump Movement, Jump parameters can be set in this command

- Function:



- Description: Jump Movement. The jump parameters can be set in this command.
- Parameter:
  - point: Set the target point. Only Cartesian point is supported.
  - h1: Lifting height form starting point.
  - h2: Dropping height.
  - z_limit: Lifting height h1 and dropping height h2 cannot be greater than z_limit, otherwise the alarm will be triggered.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 19:03:22

2.21 Jump Movement, Jump parameters are called by Arch index

- Function:



- Description: Jump Movement. The jump parameters are called by Arch index.
- Parameter:
  - point: Set the target point. Only Cartesian point is supported.
  - index: Arch index. Value range: 0 - 9. Please set Jump parameters on the System > Parameters > RobotParams > PlayBackArch page.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 19:08:39

2.22 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 14:53:49

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 14:53:57

Move Arguments Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.24 Set the joint acceleration rate

- Function:



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

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-29 14:54:05

2.25 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: 2021-09-28 19:15:42

2.26 Set the Cartesian acceleration rate

- Function:

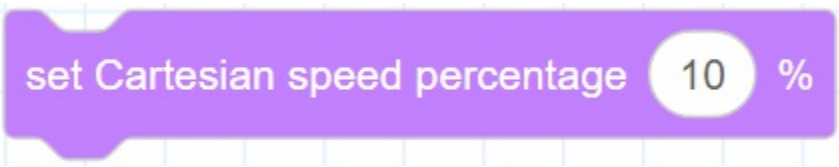


- Description: Set the Cartesian acceleration rate.
- Parameter: Acceleration rate, range: 0~100.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:54:38

2.27 Set the Cartesian velocity rate

- Function:



- Description: Set the Cartesian velocity rate.
- Parameter: Velocity ratio, range: 0~100.
- Return: None

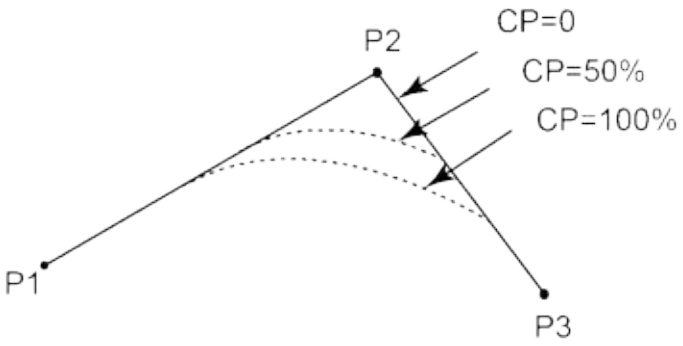
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 19:21:26

2.28 Set the smooth path rate

- Function:



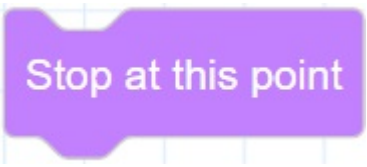
- Description: Set the smooth path rate. when reaching the end point from the starting point through the intermediate point, whether the transition through the intermediate point is in a right Angle mode or in a curve mode, as shown in the following figure.
- Parameter: Smooth path ratio, range: 0~100.
- Return: None



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:54:28

2.29 Stop at this point

- 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: 2021-09-28 19:42:46

2.30 Set Load Parameters

- Function:



- Description: Set payload, X-axis offset, Y-axis offset and servo index.
- Required parameter:
  - payload: Payload. Value range: 0- 750. Unit: g.
  - X-Offset: Offset in X-axis.
  - Y-Offset: Offset in Y-axis.
  - Optional parameter: index, servo parameter index. The default value range is 1 - 10.
- Return: None

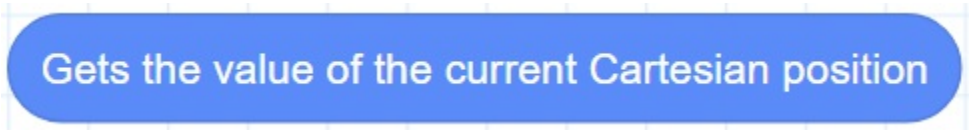
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 19:46:11

Posture Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.31 Gets the value of the current Cartesian position

- Function:



- Description: Get the current pose of the robot under the Cartesian coordinate system.
- Parameter: None
- Return: The cartesian coordinate of the current position

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:08:50

2.32 Gets a value of the current Cartesian position

- Function:

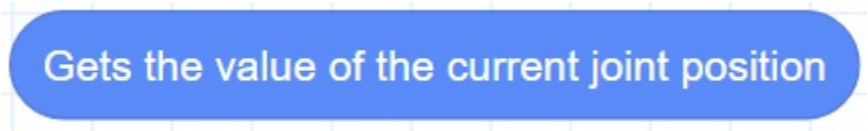


- Description: Get a coordinate value of the robot under the Cartesian coordinate system. If you have set the User or Tool coordinate system, get the index of User or Tool coordinate system.
- Parameter: Select an axis or coordinate system.
- Return: a coordinate value or index of coordinate system.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:55:06

2.33 Gets the value of the current Joint position

- Function:



- Description: Get the current pose of the robot under the Joint coordinate system.
- Parameter: None
- Return: Joint coordinates of the current pose.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:10:49

2.34 Gets a value of the current Joint position

- Function:



- Description: Get a coordinate value of the robot under the Joint coordinate system.
- Parameter: Select J1~J4.
- Return: The Joint coordinate of the current position.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:14:04

2.35 Cartesian position offset

- Function:



- Description: Set the X, Y, Z, R axes offset under the Cartesian coordinate system to return a new Cartesian coordinate point.
- Parameter:
  - △x: Indicate offset of X axis
  - △y: Indicate offset of Y axis
  - △z: Indicate offset of Z axis
  - △r: Indicate offset of R axis
  - position: Indicate the current Cartesian point.
- Return: Cartesian point.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-29 14:49:16

2.36 Angle position offset

- Function:



- Description: Set the joint offset in the Joint coordinate system to return a new joint point.
- Parameter:
  - △J1~△J4: Indicate J1 - J4 axes offset
  - position: Indicate the current joint point.
- Return: Joint point.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 20:19:41

2.37 Custom Cartesian point

- Function:



- Description: User-define a Cartesian point.
- Parameter:
  - {X, Y, Z, R}: Indicate X, Y, Z, R axes coordinates.
  - User: User coordinate system index. Value range: 0-9.
  - Tool: Tool coordinate system index. Value range: 0-9.
- Return: Cartesian point.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:55:28

2.38 Custom joint point

- Function:



- Description: User-define a joint point.
- Parameter: J1- J4 axes coordinates.
- Return: Joint point.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 20:24:13

2.39 Get coordinates

- Function:



- Description: Gets the coordinates of the target point.
- Parameter: Select the target point from the **Point** page.
- Return: Coordinates of the target point.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 20:27:08

2.40 Gets a coordinate value of a point

- Function:



- Description: Gets a coordinates of the target point. If you have set the User or Tool coordinate system, get the index of User or Tool coordinate system.
- Parameter:
  - axis: Select an axis or coordinate system
  - point: Select the target point from the **Point** page.
- Return: Coordinates of the target point or index of coordinate system.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-29 14:50:37

I/O Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:17:56

2.41 Get digital input

- Function:



- Description: Get the value of digital input port.
- Parameter: Select digital input port.
- Return: The value of digital input port.

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:30:45

2.42 Set digital output

- Function:



- Description: Set the status of digital output port.
- Parameter:
  - DO: Digital output index.
  - Status: set the DO to on or off.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:32:41

2.43 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.
  - Status: Set the DO to on or off.
- Return: None

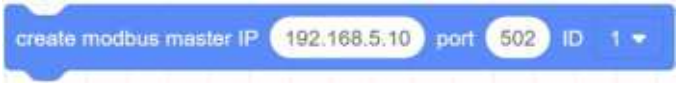
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 20:35:42

Modbus Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.44 Creates a Modbus master command

- Function:

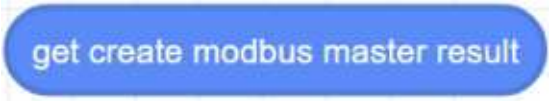


- Description: Creates a Modbus master to connect to a Modbus slave.
- Parameter:
  - IP: indicate IP address of the Modbus slave.
  - Port: indicate the port of the Modbus slave station.
  - ID: indicate ID of the Modbus slave, the value range is 1 to 4.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 14:56:08

2.45 Get the connection result command

- Function:



- 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: 2021-09-28 20:45:53

2.46 Get input register command

- Function:

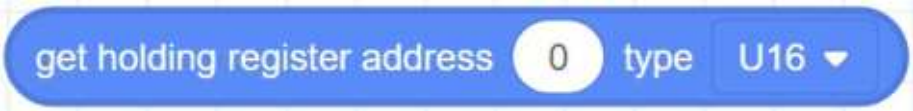


- Description: Read the input register value with the specified data type from the Modbus slave.
- Parameter:
  - Address: Starting address of the input registers. Value range: 0 – 4095.
  - type: Data type
    - “U16”: Read 16-bit unsigned integer (two bytes, occupy one register)
    - “U32”: Read 32-bit 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: 2021-09-28 20:46:01

2.47 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: 2021-09-28 20:46:07

2.48 Get discrete input register command

- Function:

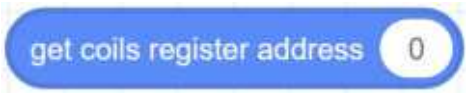


- Description: Read the discrete input register value from Modbus slave.
- Parameter: Address: starting address of the discrete inputs register. Value range: 0-4095.
- Return: The discrete input register value

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:46:15

2.49 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: 2021-09-28 20:46:22

2.50 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: 2021-09-28 20:51:57

2.51 Set coil register command

- Function:

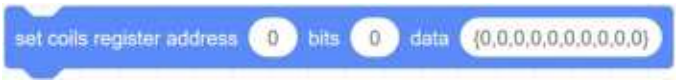


- Description: Set the coil register in the Modbus slave. This command is not supported when the coil register address is from 0 to 5.
- Parameter:
  - Address: Starting address of the coils register. Value range: 6 – 4095.
  - Data: the values written into the coil register.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:52:15

2.52 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 14:56:33

2.53 Set holding register command

- Function:

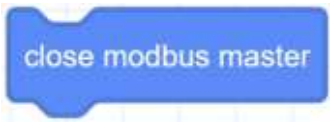


- Description: Set the holding register value in the Modbus slave.
- Parameter:
  - Address: Starting address of the holding registers to set. Value range: 0 – 4095.
  - Data: the values written into the holding register.
  - type: Data type
    - “U16”: Set 16-bit unsigned integer (two bytes, occupy 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 14:56:40

2.54 Close Modbus master command

- Function:



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

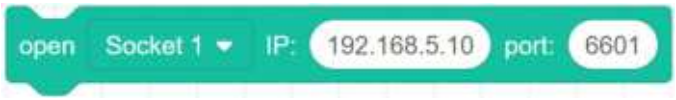
Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:52:36

TCP Commands

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.55 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 14:56:51

2.56 Get open socket command

- Function:

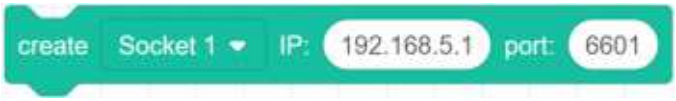


- Description: Get the connection result.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return:
  - 0: TCP connection is successful
  - 1: Input parameters are incorrect

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:57:33

2.57 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 14:57:03

2.58 Get create socket command

- Function:

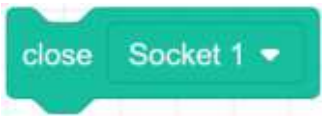


- Description: Get the connection result.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return:
  - 0: TCP network is created successfully
  - 1: TCP network is created failed
  - Socket: Socket object

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:57:47

2.59 Close socket command

- Function:

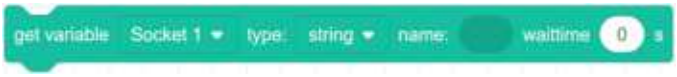


- Description: Release a TCP network.
- Parameter: Socket: indicate Socket index, value range: Socket 1 to Socket 4.
- Return: None

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-28 20:57:54

2.60 Get variable command

- Function:

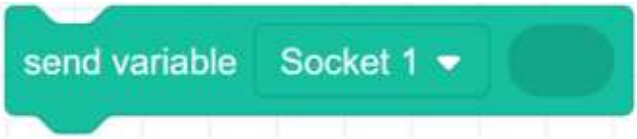


- Description: Obtain data through Socket communication.
- Parameter:
  - Socket: indicate Socket index, value range: Socket 1 to Socket 4.
  - Type: string or number
  - Name: Variable used to hold data
  - Waiting time: Set the waiting time, if the waiting time value is 0, it will wait until get data.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

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-29 14:57:21

2.61 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 14:57:27

2.62 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: 2021-09-28 20:58:16

Vision Command

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.63 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 05:41:01

2.64 Get connect result command

- Function:



- 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 05:40:35

2.65 Trigger the camera command

- Function:



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

s

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-29 05:43:25

2.66 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 05:44:44

2.67 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 05:44:30

2.68 Get the number of data groups command

- Function:



- 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 05:45:13

2.69 Gets a data command

- Function:



- 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 05:45:46

2.70 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 05:46:13

Pallet Command

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:16:07

2.71 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 05:57:50

2.72 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 05:59:23

2.73 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 14:59:31

2.74 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 06:00:58

2.75 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 06:02:59

2.76 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 15:36:45

2.77 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 06:08:19

2.78 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 06:10:48

3. Description of Programming

Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved，powered by GitbookRevision: 2021-09-28 11:31:18

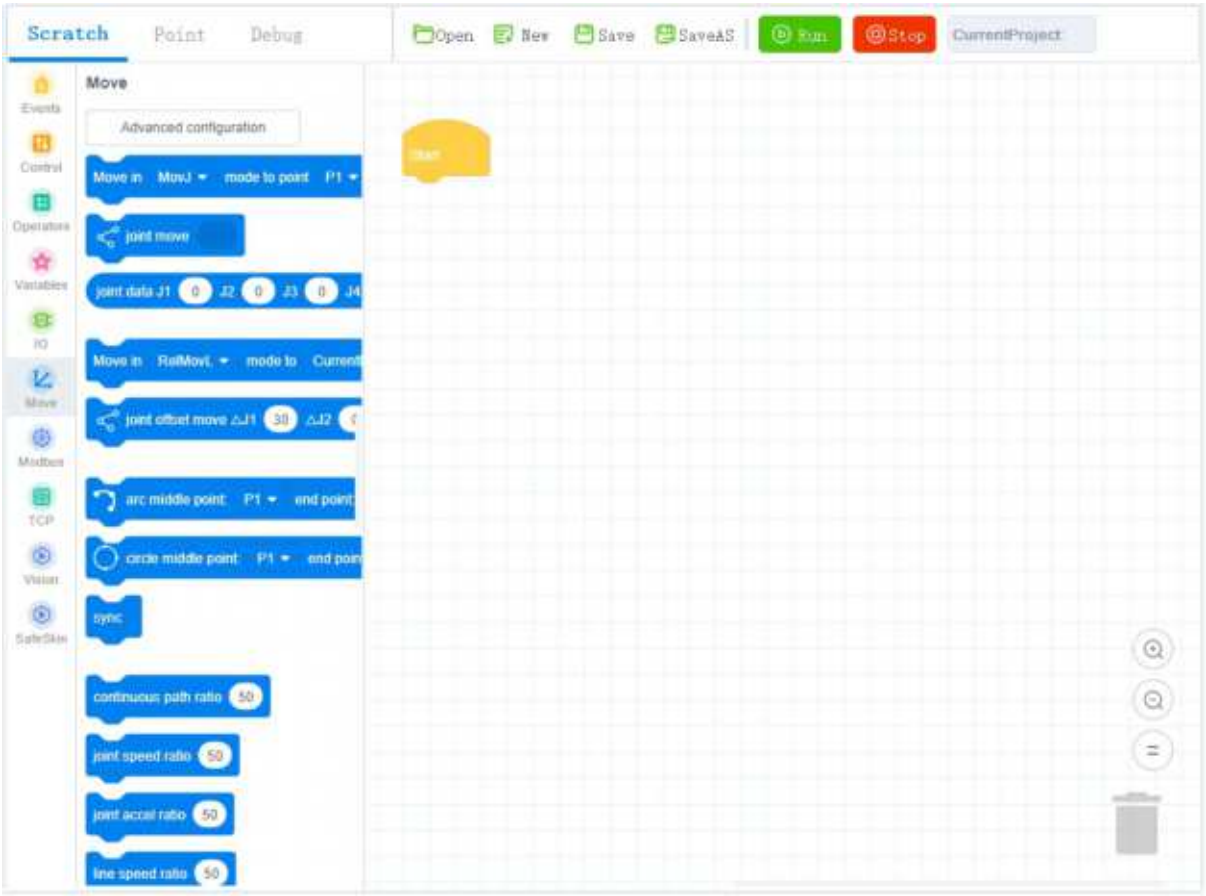
3.1 Basic operation

Prerequisites

The robot has been powered on.

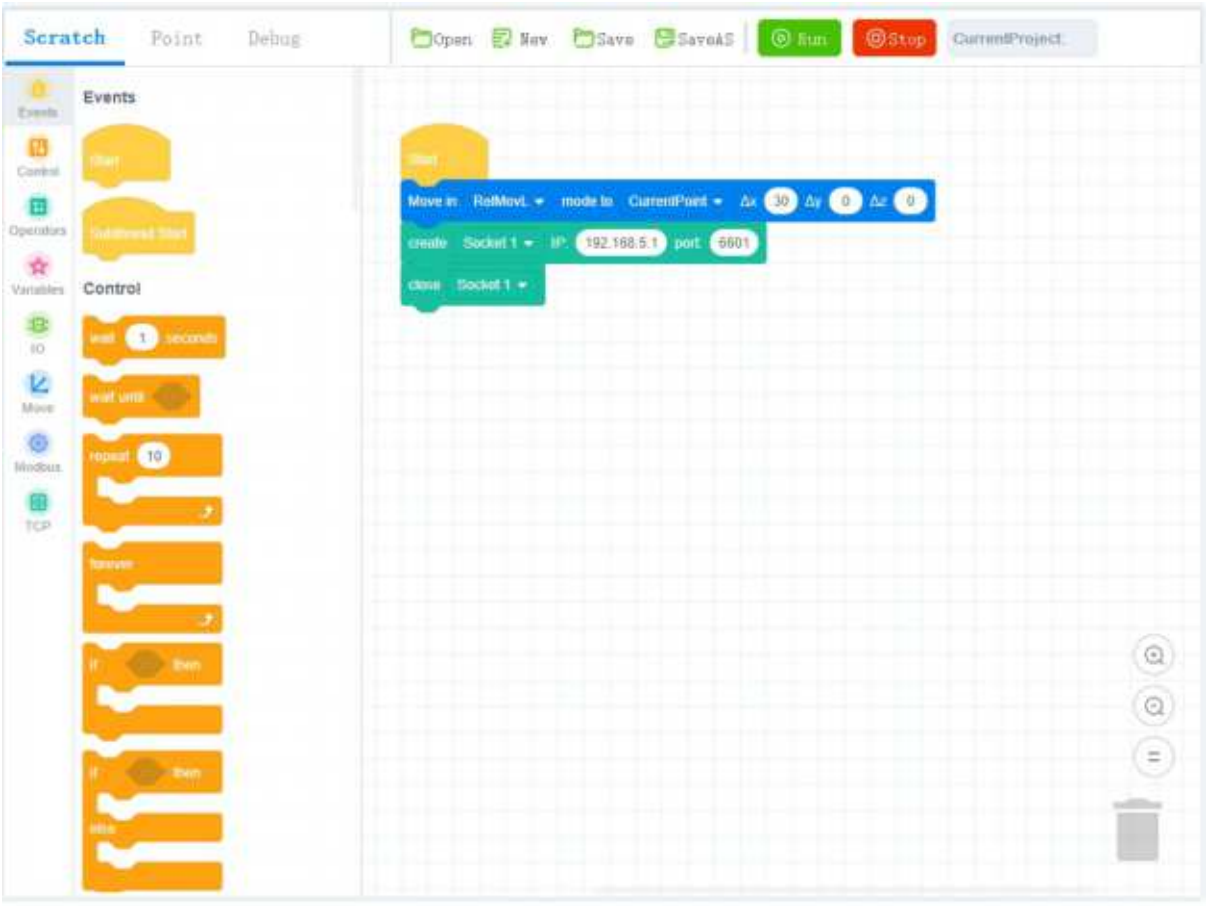
Procedure

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 Run 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

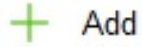
The project has been created or imported.

Procedure

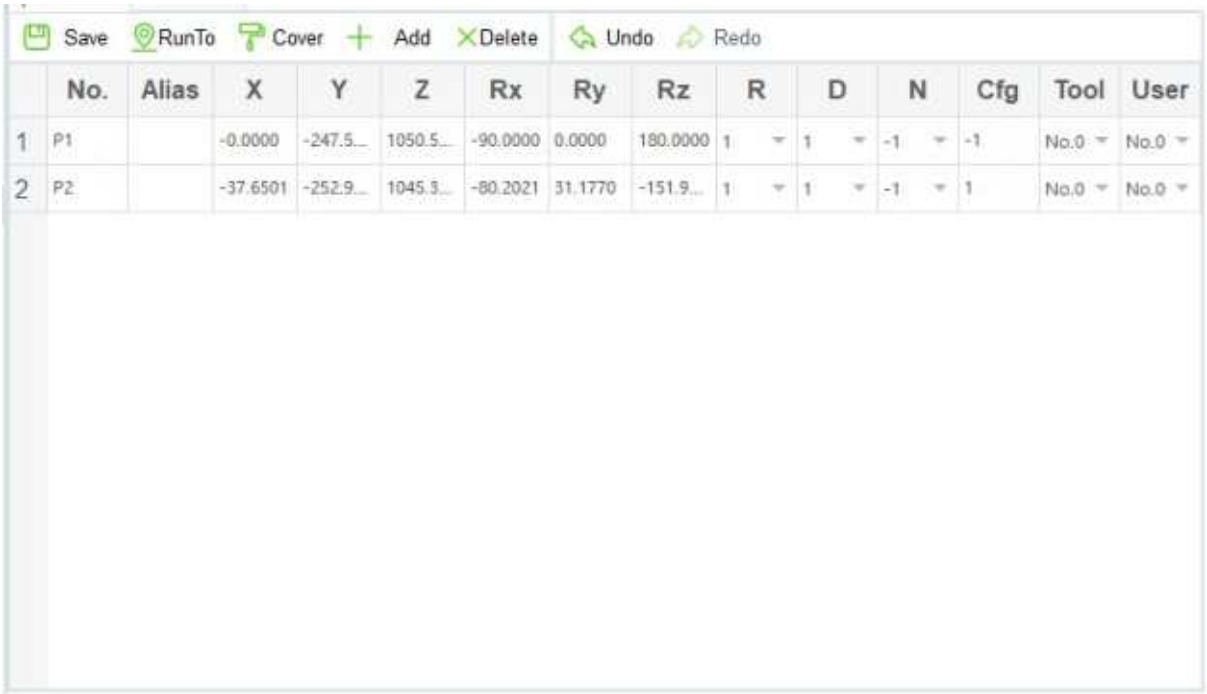
After creating a project, please teach positions on the **point** page for calling commands when programming a robot. If the existing taught positions list has been imported, this operation can be skipped.

Step 1 Click  to enable the robotic arm.

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

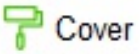
Step 3 Click **Point** to enter point page and click  to add a teaching point.

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



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 

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

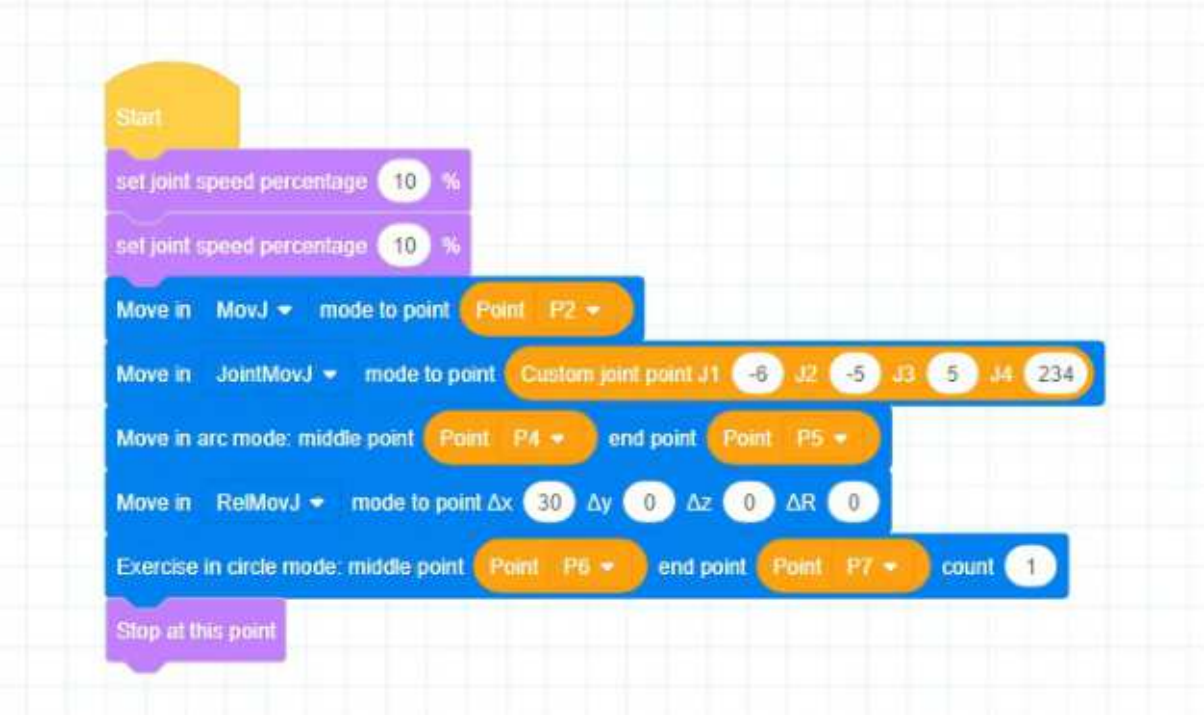
3.3 Quick Start

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

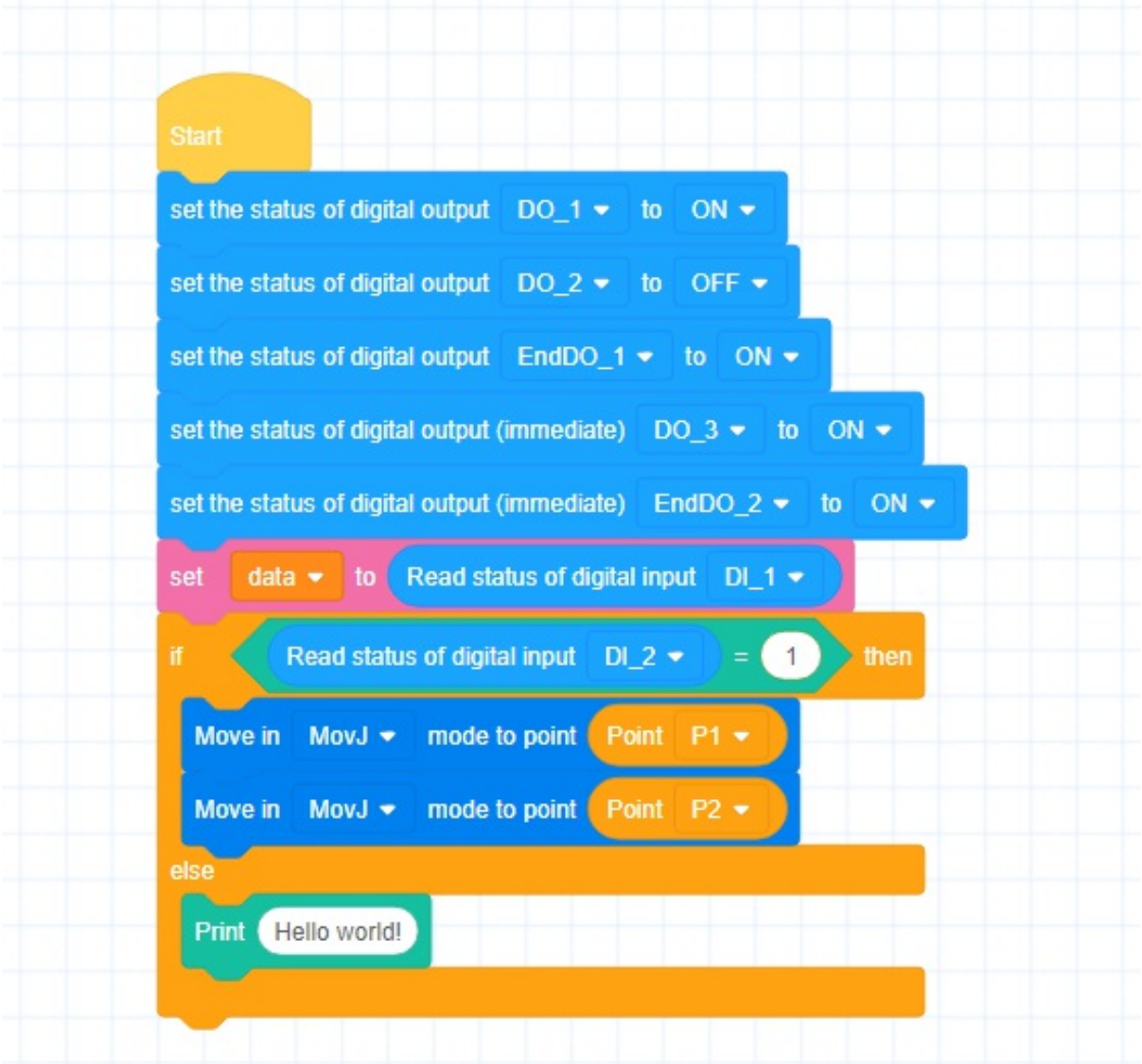
Running the Motion commands can control the movement of robot in the joint coordinate system and the Cartesian coordinate system. The detailed description of the Motion commands, please see **Motion Commands**. The following figure shows a programming program that includes Motion commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-29 15:16:37

3.3.2 I/O Setting

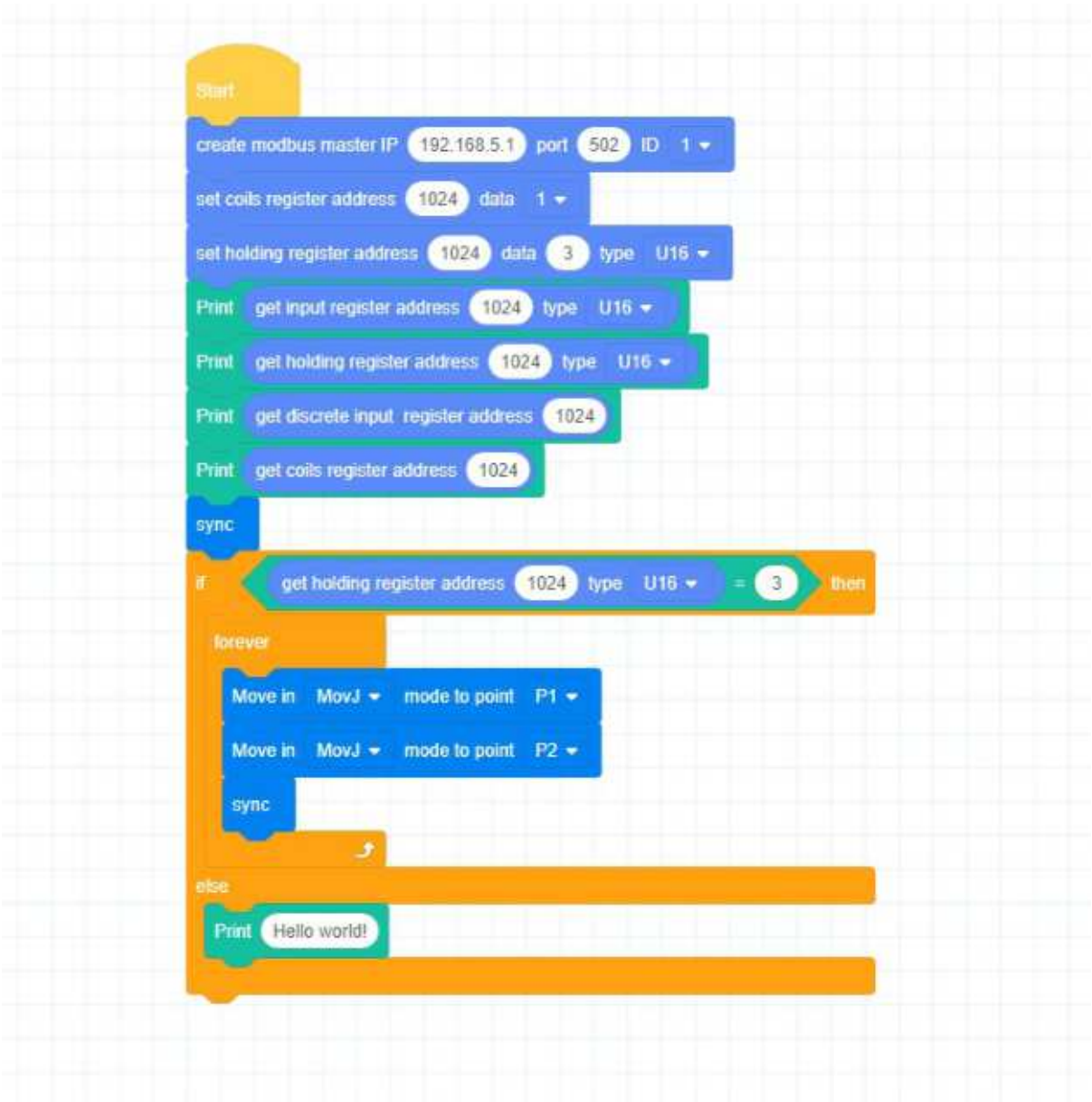
Running I/O commands to set or get each I/O statue. The detailed description of the I/O commands, please see **I/O Commands**. The following figure shows a programming program that includes I/O commands.



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

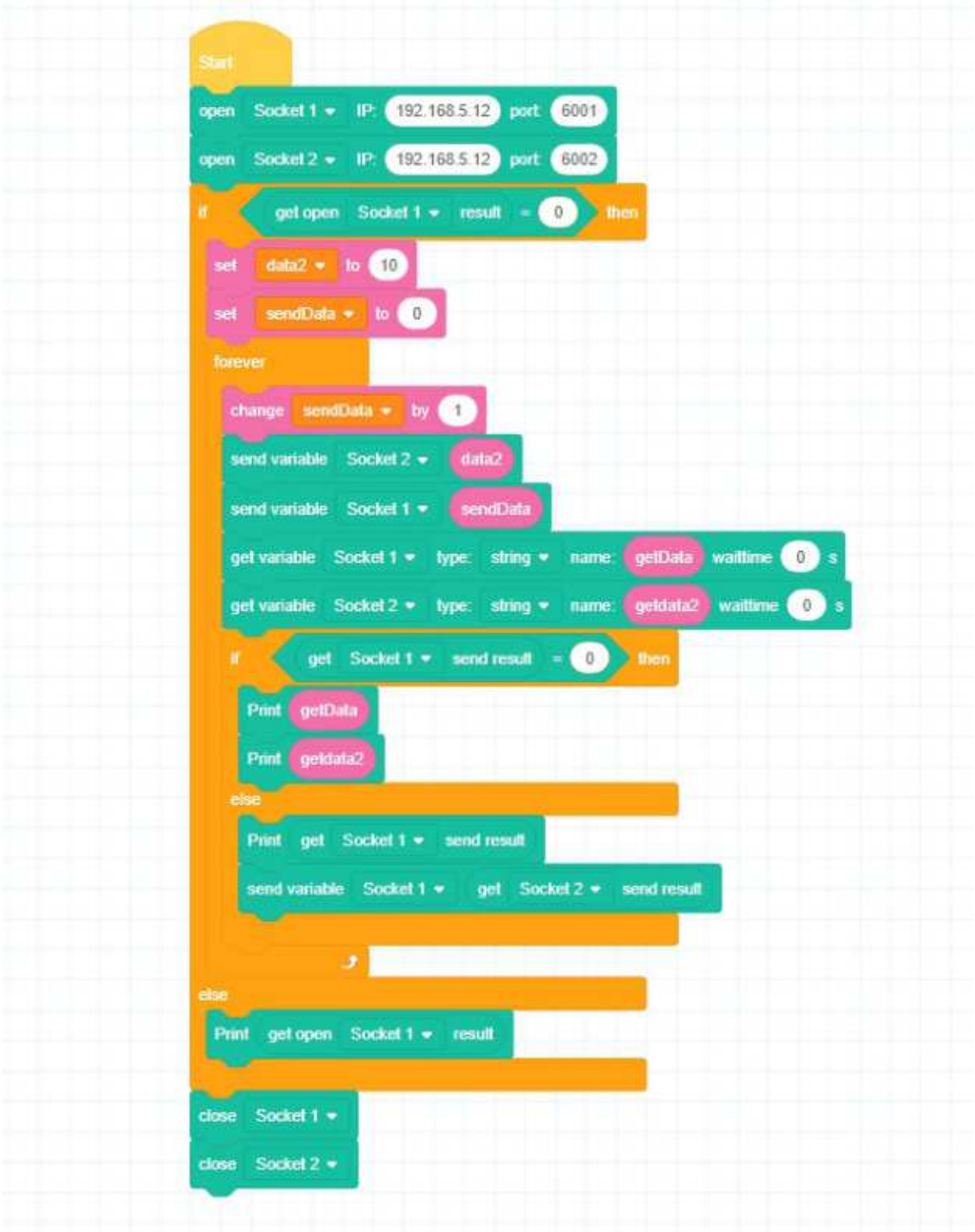
By running the Modbus commands to set or read the value of each register address. The detailed description of the Modbus commands, please see **Modbus Commands**. The following figure shows a programming program that includes Modbus commands.



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

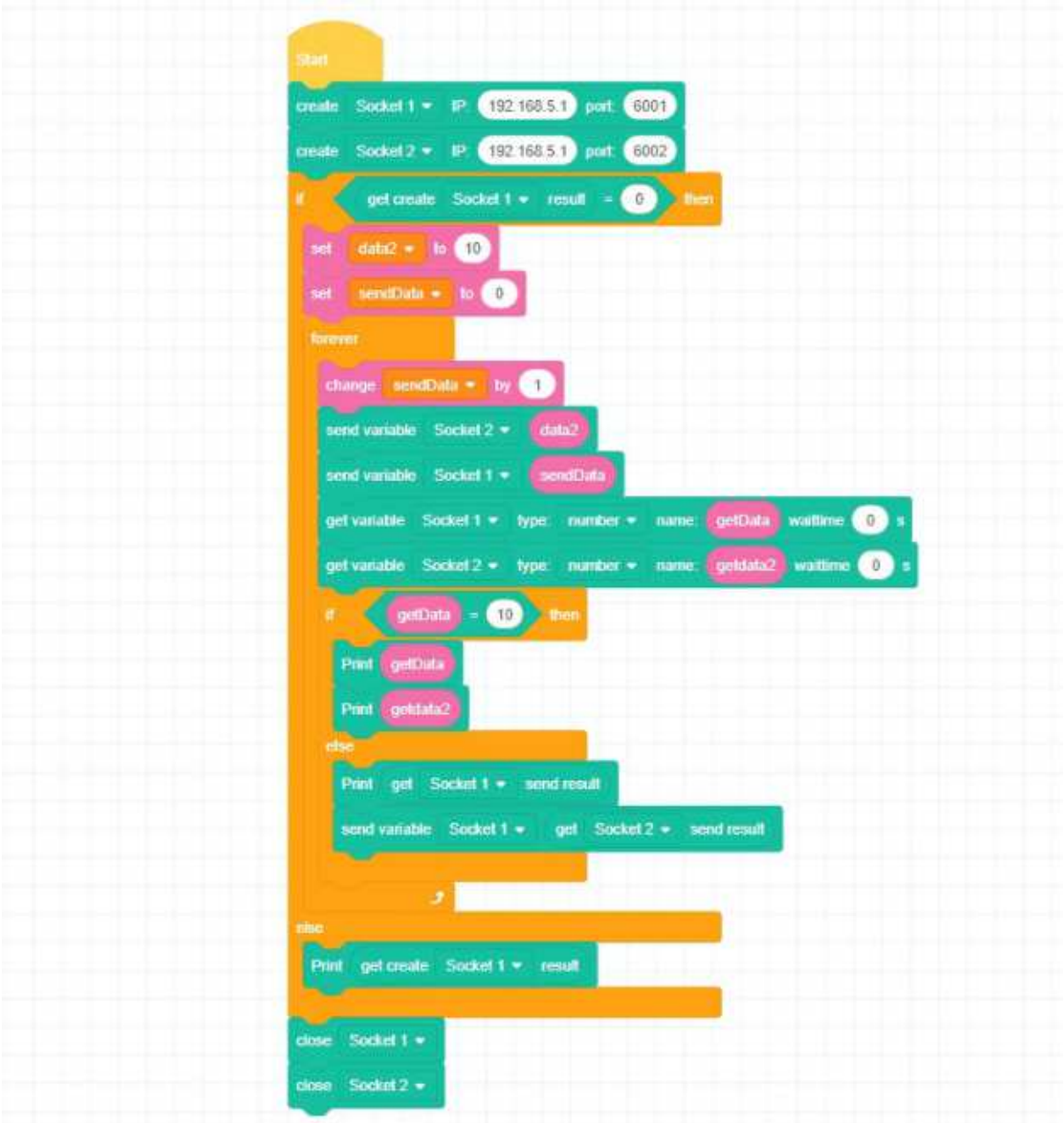
Run to establish communication with the TCP server, the robot as the TCP client. Running TCP commands can send and read communication data, the detailed description of TCP commands, please see **TCP Commands**. The following figure shows a programming program that includes TCP commands.



Copyright © ShenZhen Yuejiang Technology Co., Ltd 2021 all right reserved, powered by GitbookRevision: 2021-09-17 15:03:20

3.3.5 Create TCP Server

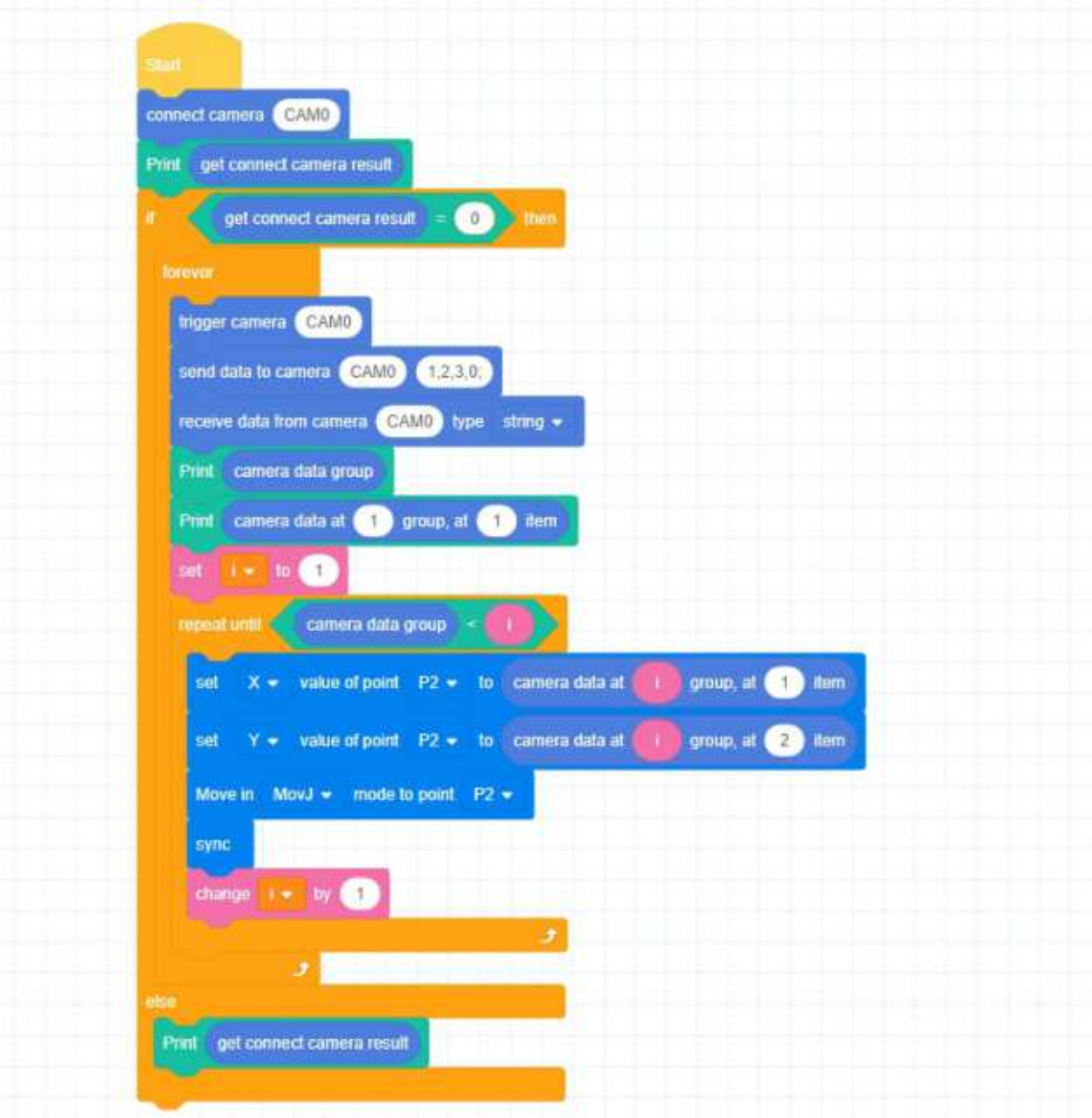
Run to set the robot as the server, waiting for the TCP client to connect. Running TCP instruction can send and read communication data, the detailed description of TCP commands, please see **TCP Commands**. The following figure shows a programming program that includes TCP commands.



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: 2021-09-29 15:16:57