NISE tutorial on Dynamixel servo motor control and CPG

29.10.2024

Dr.Ing.habil. John Nassour

[Goal]

Today's session aims to familiarize you with programming the Dynamixel servo motor.

With the provided example code, you will learn how to reset the EEPROM, how to set and change the ID of the servo motor, and then how to move the motor between two positions.

You will also learn how to connect your CPG code from the previous session to the motor output motion.

[Preparing libraries for Dynamixel servo motors]

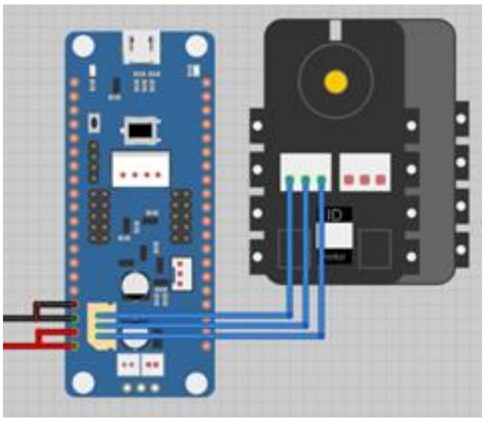
This tutorial uses “***OpenCM9.04***” to control the Dynamixel servo motors. To install the libraries, please use the link below:

https://emanual.robotis.com/docs/en/software/arduino_ide/

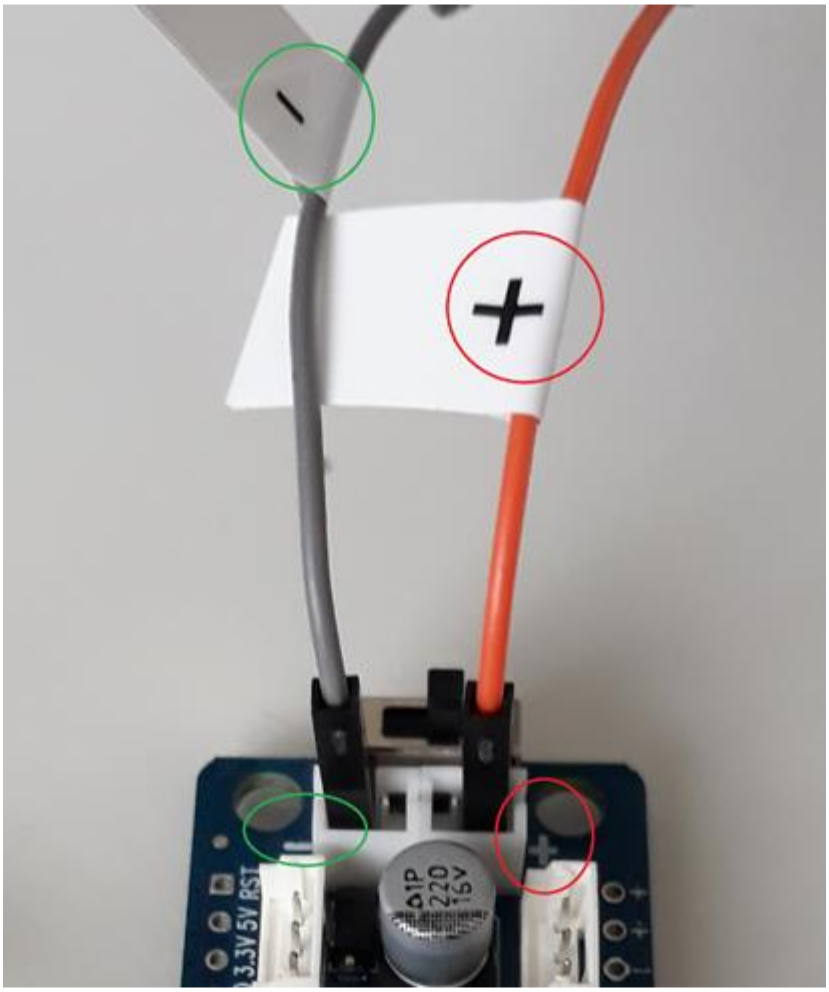
Upon successful installation, please run the examples listed in the same link: “***Test Example for LED***”, and “***Test Example for Button***”.

[Understanding by an example]

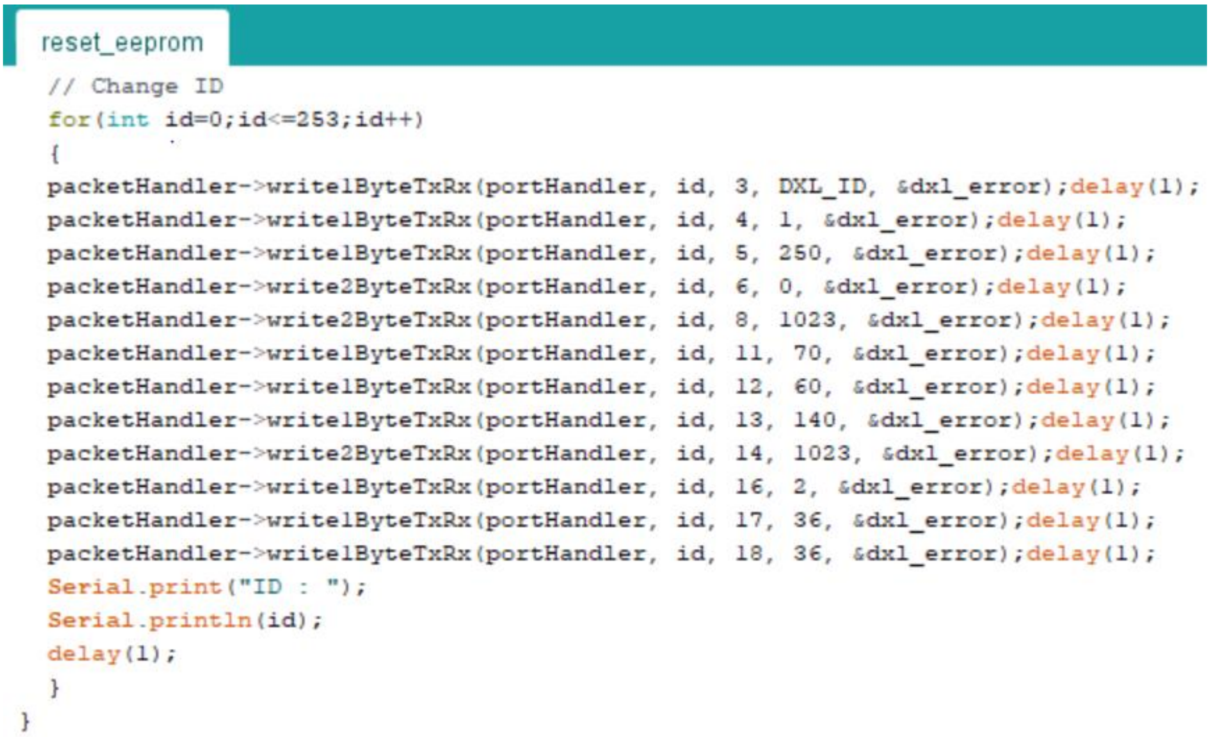
1. Connect the Dynamixel servo motor to the board.



2. Connect the 12 Volt power supply to the board; please make attention to the polarity (see **+** and **-** in the picture below)!



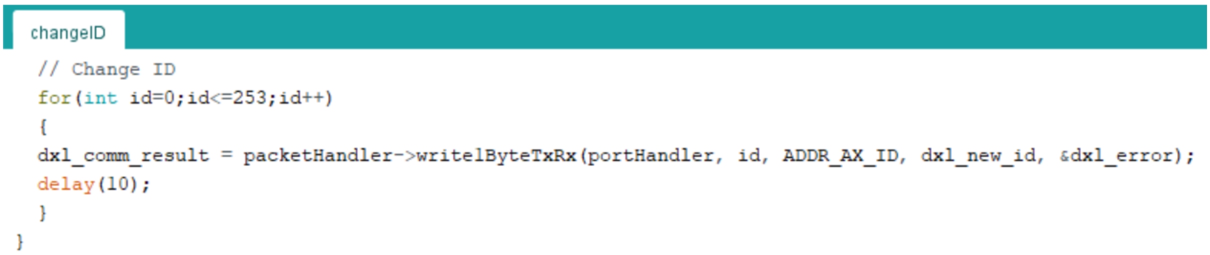
3. Run the provided code “reset_eeprom”.



```
  // Change ID
  for(int id=0;id<=253;id++)
  {
  packetHandler->write1ByteTxRx(portHandler, id, 3, DXL_ID, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 4, 1, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 5, 250, &dxl_error);delay(1);
  packetHandler->write2ByteTxRx(portHandler, id, 6, 0, &dxl_error);delay(1);
  packetHandler->write2ByteTxRx(portHandler, id, 8, 1023, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 11, 70, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 12, 60, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 13, 140, &dxl_error);delay(1);
  packetHandler->write2ByteTxRx(portHandler, id, 14, 1023, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 16, 2, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 17, 36, &dxl_error);delay(1);
  packetHandler->write1ByteTxRx(portHandler, id, 18, 36, &dxl_error);delay(1);
  Serial.print("ID : ");
  Serial.println(id);
  delay(1);
  }
}
```

This part will set all IDs (0-253) with all their eeprom addresses (3-18) to initial values. Therefore running ***reset_eeprom*** will reset all motor parameters.

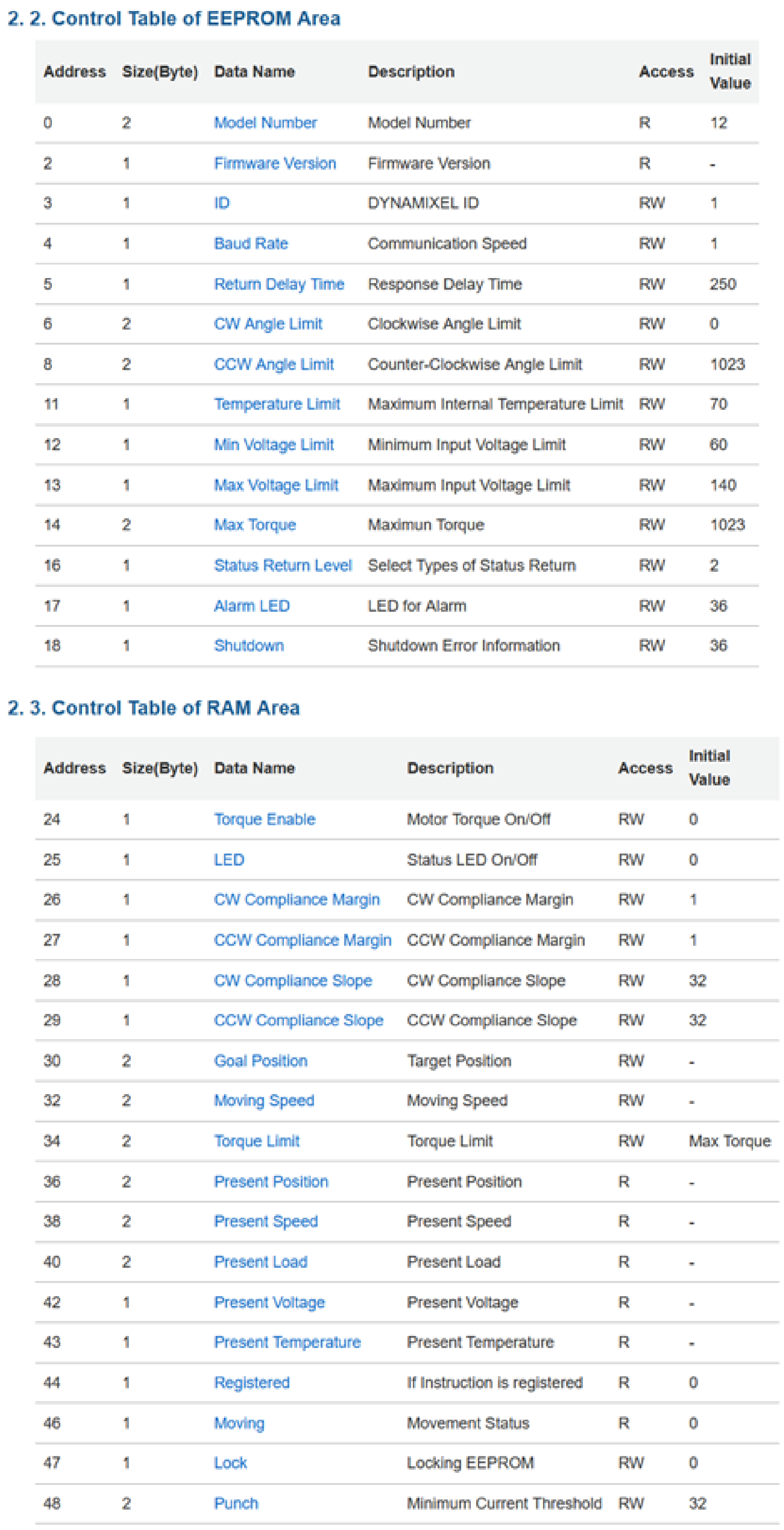
4. Run the provided code “***changeID***”. This code will ask you to provide over the serial port the desired ID of the motor. It will then scan the connected motors’ ID (in a range varying from 0 to 253) and give them the ID you have entered. Please make sure that only one motor is connected to the board to avoid having conflicting IDs.



```
  // Change ID
  for(int id=0;id<=253;id++)
  {
  dxl_comm_result = packetHandler->write1ByteTxRx(portHandler, id, ADDR_AX_ID, dxl_new_id, &dxl_error);
  delay(10);
  }
}
```

This part will set all IDs (0-253) to a new ID (dxl_new_id).

5. Check the registers table (2.2 & 2.3) of your motor:



2. 2. Control Table of EEPROM Area

Address	Size(Byte)	Data Name	Description	Access	Initial Value
0	2	Model Number	Model Number	R	12
2	1	Firmware Version	Firmware Version	R	-
3	1	ID	DYNAMIXEL ID	RW	1
4	1	Baud Rate	Communication Speed	RW	1
5	1	Return Delay Time	Response Delay Time	RW	250
6	2	CW Angle Limit	Clockwise Angle Limit	RW	0
8	2	CCW Angle Limit	Counter-Clockwise Angle Limit	RW	1023
11	1	Temperature Limit	Maximum Internal Temperature Limit	RW	70
12	1	Min Voltage Limit	Minimum Input Voltage Limit	RW	60
13	1	Max Voltage Limit	Maximum Input Voltage Limit	RW	140
14	2	Max Torque	Maximun Torque	RW	1023
16	1	Status Return Level	Select Types of Status Return	RW	2
17	1	Alarm LED	LED for Alarm	RW	36
18	1	Shutdown	Shutdown Error Information	RW	36

2. 3. Control Table of RAM Area

Address	Size(Byte)	Data Name	Description	Access	Initial Value
24	1	Torque Enable	Motor Torque On/Off	RW	0
25	1	LED	Status LED On/Off	RW	0
26	1	CW Compliance Margin	CW Compliance Margin	RW	1
27	1	CCW Compliance Margin	CCW Compliance Margin	RW	1
28	1	CW Compliance Slope	CW Compliance Slope	RW	32
29	1	CCW Compliance Slope	CCW Compliance Slope	RW	32
30	2	Goal Position	Target Position	RW	-
32	2	Moving Speed	Moving Speed	RW	-
34	2	Torque Limit	Torque Limit	RW	Max Torque
36	2	Present Position	Present Position	R	-
38	2	Present Speed	Present Speed	R	-
40	2	Present Load	Present Load	R	-
42	1	Present Voltage	Present Voltage	R	-
43	1	Present Temperature	Present Temperature	R	-
44	1	Registered	If Instruction is registered	R	0
46	1	Moving	Movement Status	R	0
47	1	Lock	Locking EEPROM	RW	0
48	2	Punch	Minimum Current Threshold	RW	32

6. Please note that you can adjust the motor parameters in the motor EEPROM, such as the ID, max Torque, etc. To change any of these parameters, you need to write one or two bytes in the eeprom. Please check the “***Size(Byte)***” of your “***Data Name***” and then choose between “***write1ByteTxRx***” and “***write2ByteTxRx***” whether you want to write 1 or 2 Byte(s). The same is valid when reading from the EEPROM.

7. Note that the goal position is two bytes in size, it can be set between 0 and 1023. Create a sinus wave on the motor goal position with 0.5 Hz and an amplitude of 600, oscillating around the goal position 500. Set the maximum motor torque to be full range first and then half range. In each test apply manually a resistance to the moving part of the motor and at the same time illustrate the resulting position through SerialPort. Use the provided code “***changeID***” **[25% of the grade]**.

8. Note that the goal position is two bytes in size, it can be set between 0 and 1023. Create a sinus wave on the motor goal position with 0.5 Hz and an amplitude of 600, oscillating around the goal position 500. Set the moving speed to be full range first and then half range. In each test illustrate the resulting position through SerialPort. Use the provided code “***changeID***” **[25% of the grade]**.

9. Now, use the CPG code (RS_neuron.ino or Matsouka oscillator) from tutorial 1 to generate an oscillatory movement at the motor **[50% of the grade]**.

The deadline is **(see course page on moodle)**.

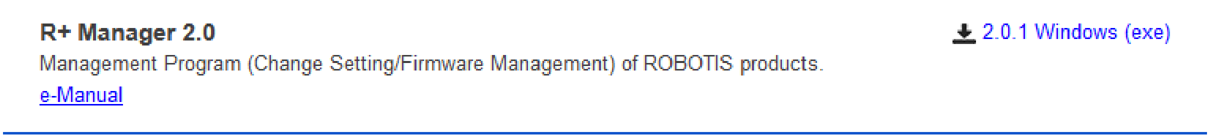
Supplementary Materials

R+ Manager 2.0 is needed when the OpenCM9.04 board crashes. Then a firmware recovery is necessary to rerun the microcontroller.

Software Installation

R+ Manager 2.0 produces a variety of software for use with DYNAMIXEL smart servos and ROBOTIS educational kits - available for free download at the Software Download page: https://en.robotis.com/service/downloadpage.php?ca_id=10

Select: **R+ Manager 2.0**

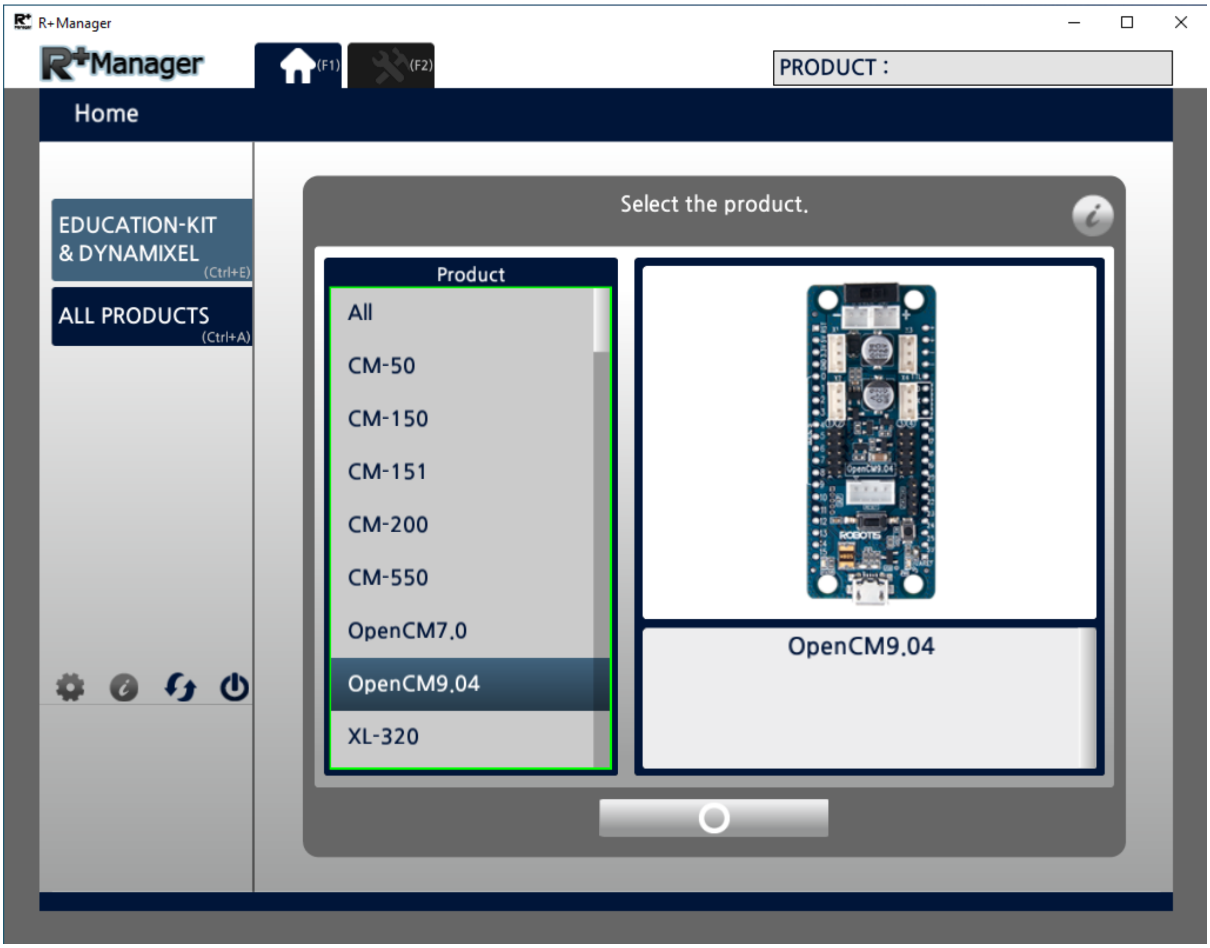


R+ Manager 2.0

Management Program (Change Setting/Firmware Management) of ROBOTIS products.

e-Manual

2.0.1 Windows (exe)



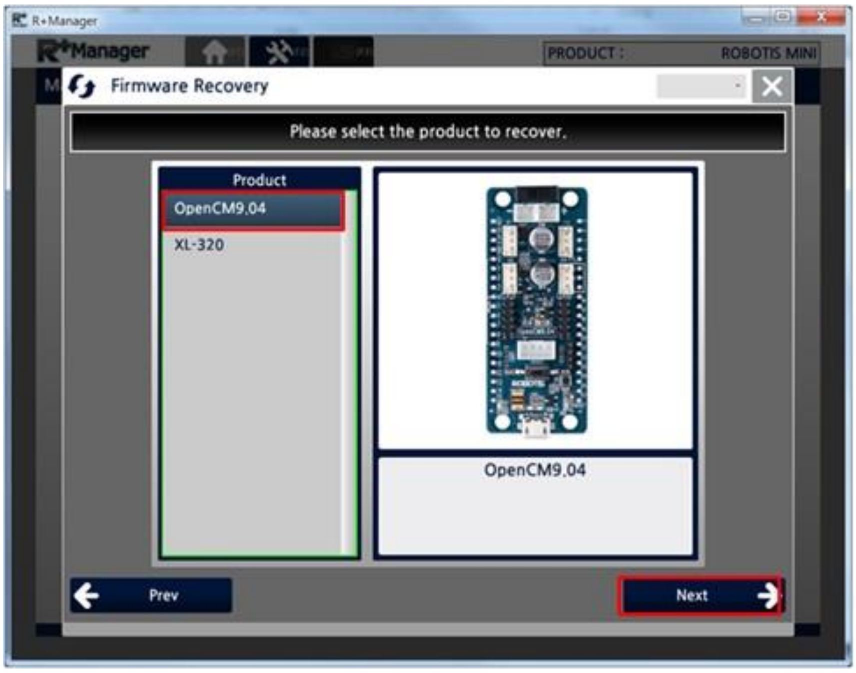
Select the product.

OpenCM9.04

1. Firmware Recovery

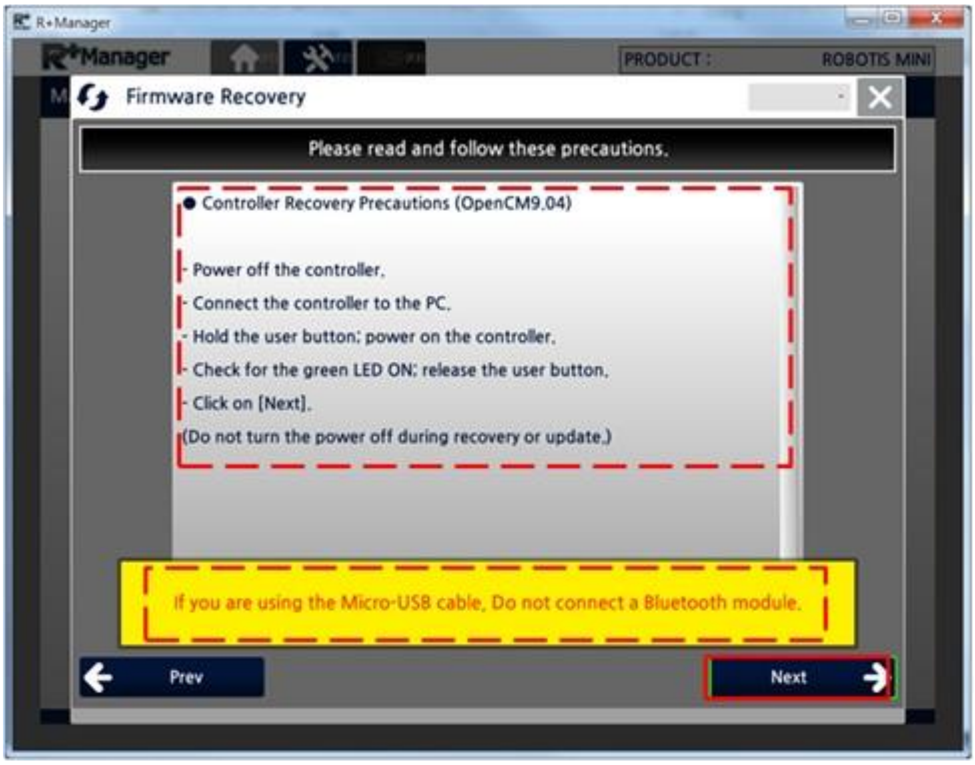
https://emanual.robotis.com/docs/en/software/rplus2/manager/#basic-features

1. In the Home Tab, select the product that you wish to recover, and click the “Next” button (**OpenCM9.04** was selected in the example).



Please select the product to recover.

2. Select the “Firmware Recovery” menu.
3. Follow the instructions that appear in order. Be sure to check the yellow warning message below.
4. After all preparations are finished, click the “Next” button.



Please read and follow these precautions.

● Controller Recovery Precautions (OpenCM9.04)

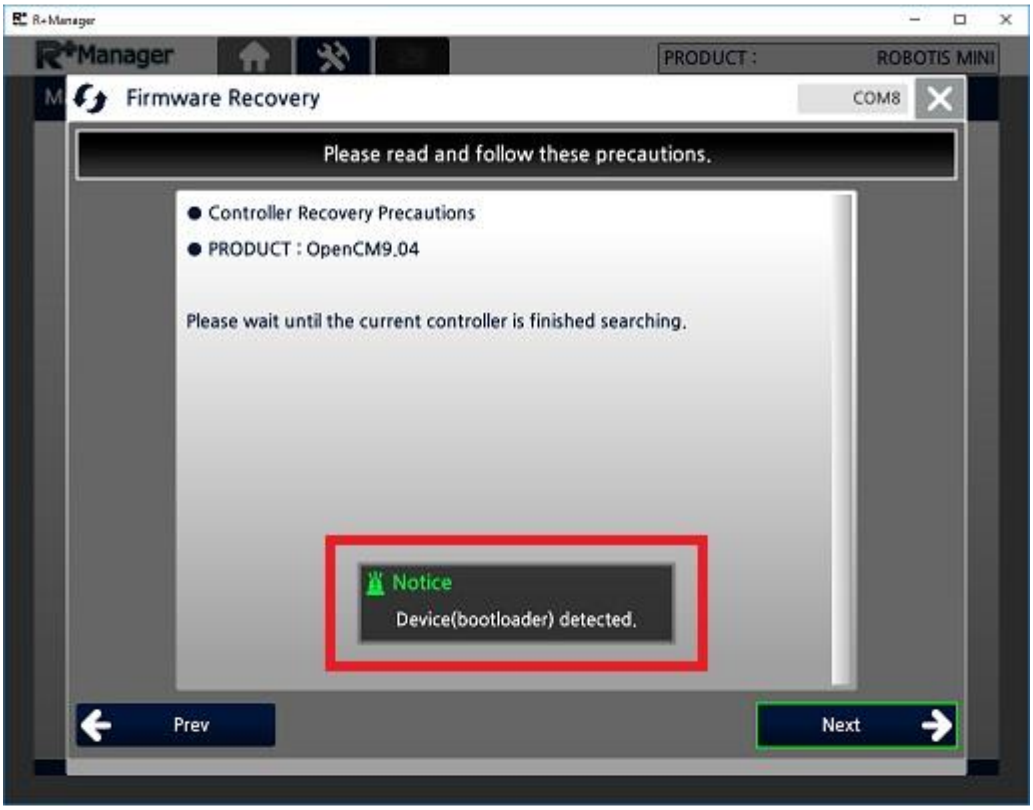
- Power off the controller.
- Connect the controller to the PC.
- Hold the user button; power on the controller.
- Check for the green LED ON; release the user button.
- Click on [Next].

(Do not turn the power off during recovery or update.)

If you are using the Micro-USB cable, Do not connect a Bluetooth module.

5. Select the connection port and click the “Next” button (Bluetooth serial port was used in the example below).

6. Once the device is connected, **bootloader** is detected for firmware installation. Click the “Next” button when it is activated.



● Controller Recovery Precautions

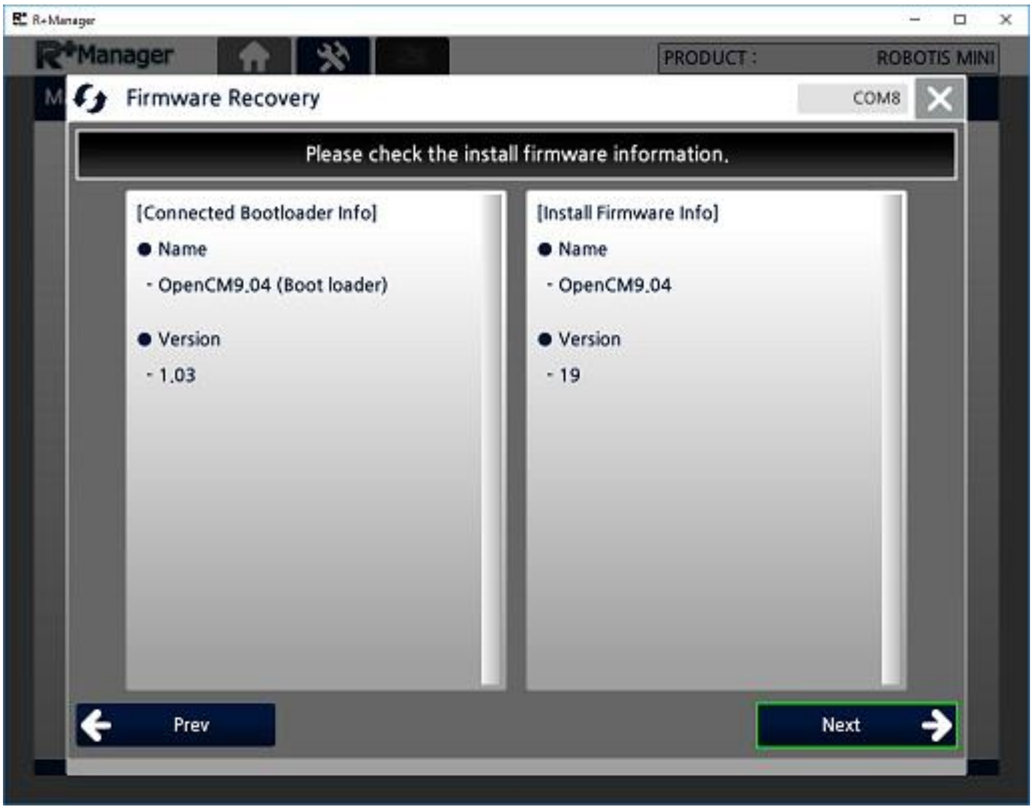
● PRODUCT : OpenCM9.04

Please wait until the current controller is finished searching.

Notice

Device(bootloader) detected.

7. Check the information of the firmware to be installed and click the “Next” button.

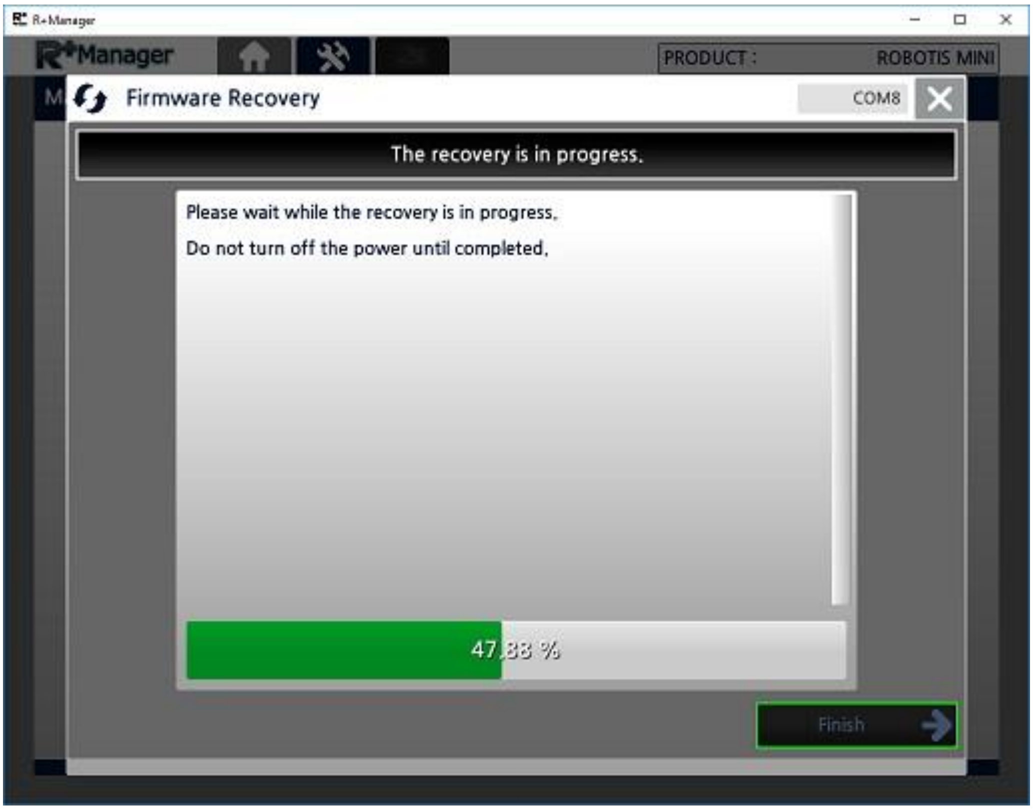


Please check the install firmware information.

[Connected Bootloader Info]
● Name
- OpenCM9.04 (Boot loader)
● Version
- 1.03

[Install Firmware Info]
● Name
- OpenCM9.04
● Version
- 19

8. When the recovery is completed, click the “Finish” button.



The recovery is in progress.

Please wait while the recovery is in progress.

Do not turn off the power until completed.