



# Flexx CNC - Fanuc User Manual

Flexxbotics Corporation Product

Version 1.0.0

March 2022



## Revisions

| Revision | Date     | Description     |
|----------|----------|-----------------|
| 0.0.0    | 2/8/2022 | Initial Release |

# Table of Contents

|  |           |
|--|-----------|
| <b>Revisions</b>                       | <b>2</b>  |
| <b>Table of Contents</b>               | <b>3</b>  |
| <b>Product Overview</b>                | <b>5</b>  |
| Product Description                    | 5         |
| Key Terms and Definitions              | 5         |
| <b>Installation and Setup</b>          | <b>6</b>  |
| Fanuc Hardware Setup                   | 6         |
| Flexx Machine Interface Hardware Setup | 9         |
| FANUC Software Setup                   | 10        |
| FANUC G-Code Programs                  | 18        |
| <b>Flexx CNC URCap</b>                 | <b>20</b> |
| Setup                                  | 20        |
| URCap Installation                     | 20        |
| License Activation                     | 21        |
| Robot Network Configuration            | 22        |
| Installation Node                      | 23        |
| Overview of Program Nodes              | 24        |
| <b>Step by Step Programming Guide</b>  | <b>28</b> |

# 1. Product Overview

## 1.1. Product Description

The Flexx CNC provides a full interface to allow a Universal Robot to bi-directionally communicate with a CNC machine. It utilizes pre-defined connections from the Flexx Machine Interface (hardware) to connect with the machine's ethernet port, cycle start button, and peripherals such as a vise, chuck, or door. Flexx CNC also provides out-of-the-box Polyscope Program Nodes to assist in programming your robot via a URCap interface.

## 1.2. Key Terms and Definitions

| Term                    | Definition / Description   |
|-------------------------|--|
| Flexx Machine Interface | A hardware interface between the robot and the machine.                        |
| Flexx CNC URCap         | Software that runs on the robot to communicate to the Flexx Machine Interface. |

## 2. Installation and Setup

### 2.1. Fanuc Hardware Setup

2.1.1. Open the front panel of the machine behind the display



2.1.2. Plug in the **ETHERNET CABLE** from the Flexx Machine Interface MACHINE into the ATA card mounted in the yellow housing.

- 2.1.3. Wire the **CYCLE START** digital output from the Flexx Machine Interface M12 connector into the relay on the rear of the cycle start button.
- 2.1.4. **Mill:** Wire the VISE digital output from the Flexx Machine Interface M12 connector into the digital input on the vise actuation control  
  
NOTE: For setups with more than one vise, utilize the additional general purpose OUTPUTS on the Flexx Machine Interface.
- 2.1.5. **Lathe:** Wire the CHUCK digital output from the Flexx Machine Interface M12 connector into the clamp foot pedal connected to the machine to close the chuck. Wire a general purpose OUTPUT on the Flexx Machine Interface to the unclamp chuck foot pedal to open the chuck.

## **2.2. Flexx Machine Interface Hardware Setup**

- 2.2.1. Connect the Machine Ethernet cable from the RJ45 connector labeled MACHINE to the RJ45 connector on the machine.
- 2.2.2. Connect the Cycle Start cable M12 connector to the machine interface CYCLE START Output.
- 2.2.3. **Mill:** Connect the Vise cable M12 connector to the machine interface VISE Output.
- 2.2.4. **Lathe:** Connect the Lathe cable M12 connector to the machine interface CHUCK Output.
- 2.2.5. Connect other general digital output and inputs to the labeled M12 connectors on the Flexx Machine Interface.
- 2.2.6. **If using Ethernet to the Robot -**
  - 2.2.6.1. Connect the Robot Ethernet cable from the RJ45 connector labeled ROBOT to the RJ45 connector inside the robot controller cabinet.
  - 2.2.6.2. Toggle the ETHERNET / DIO switch to the ETHERNET side.
- 2.2.7. **If using Digital I/O to the Robot -**
  - 2.2.7.1. Connect the robot side Digital Output connectors to the Digital Outputs inside the robot controller cabinet. Connect the Flexx Machine Interface side M12 connector to the 12-pin ROBOT IN connector.
  - 2.2.7.2. Connect the robot side Digital Input connectors to the Digital Inputs inside the robot controller cabinet. Connect the Flexx Machine Interface side M12 connector to the 4-pin ROBOT OUT connector.


- 2.2.7.3. Toggle the ETHERNET / DIO switch to the DIO side.
- 2.2.8. Plug in AC Power to the unit.
- 2.2.9. Verify that the power LED displays green.

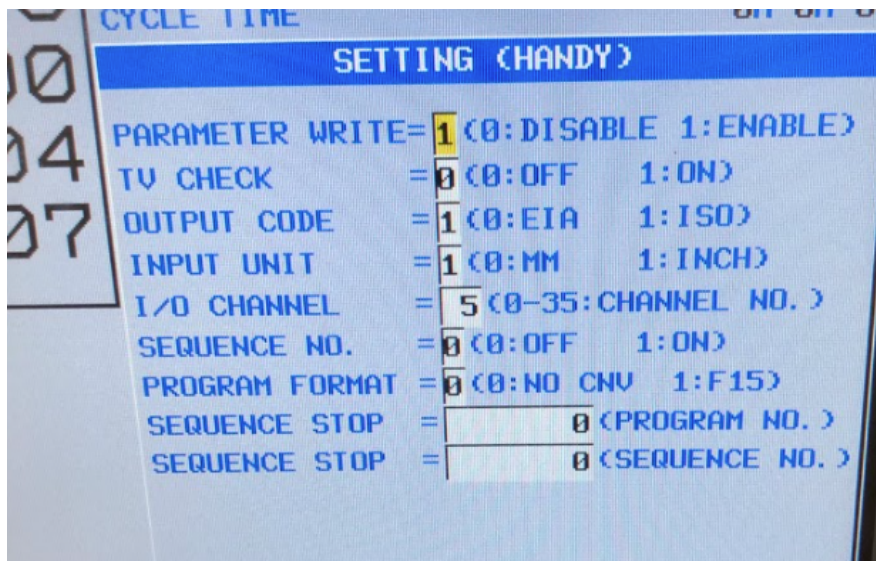


## 2.3. FANUC Software Setup

2.3.1. Switch to MDI mode using this key:



2.3.2. Hit  until **Setting (Handy)** is on screen, and change the Parameter Write Field from 0 to 1 by pressing “1” then “input” with the field highlighted.

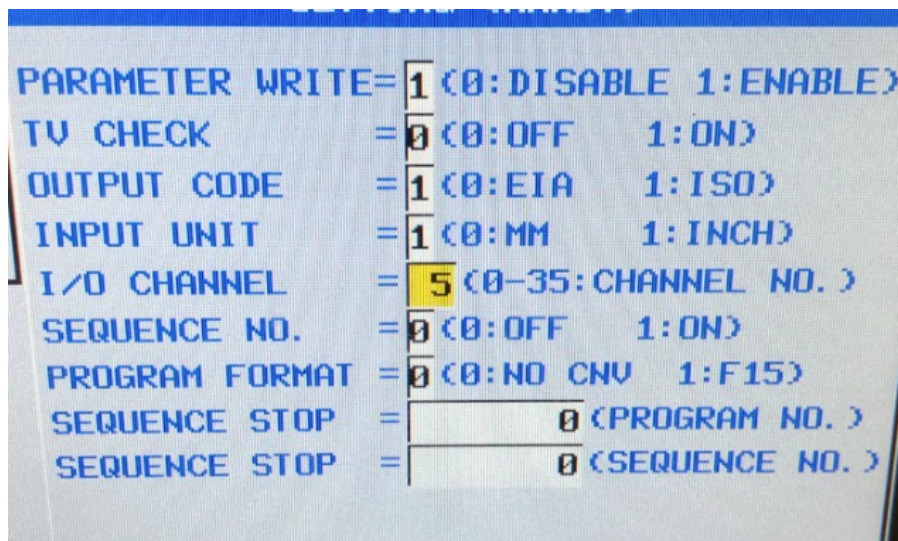


2.3.3. Plug in the USB provided by Flexxbotics to the port near the controller screen. Hit the **green OPRT key**. Hit the **Change Device** soft key. Hit the **USB** soft key.

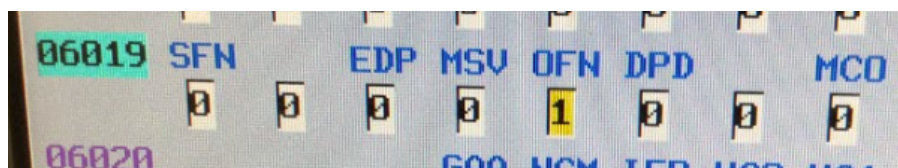
2.3.4. Use the arrow keys to highlight **O8000**. Hit the **green OPRT key**.

Advance with the **right arrow soft key** until **F INPUT** is present. Hit **F INPUT**, then **F GET**, the **F SET**, then **EXEC**.

2.3.5. Return to **Settings (handy)** and change the I/O channel to 4 by pressing “4” and then input with the field highlighted.



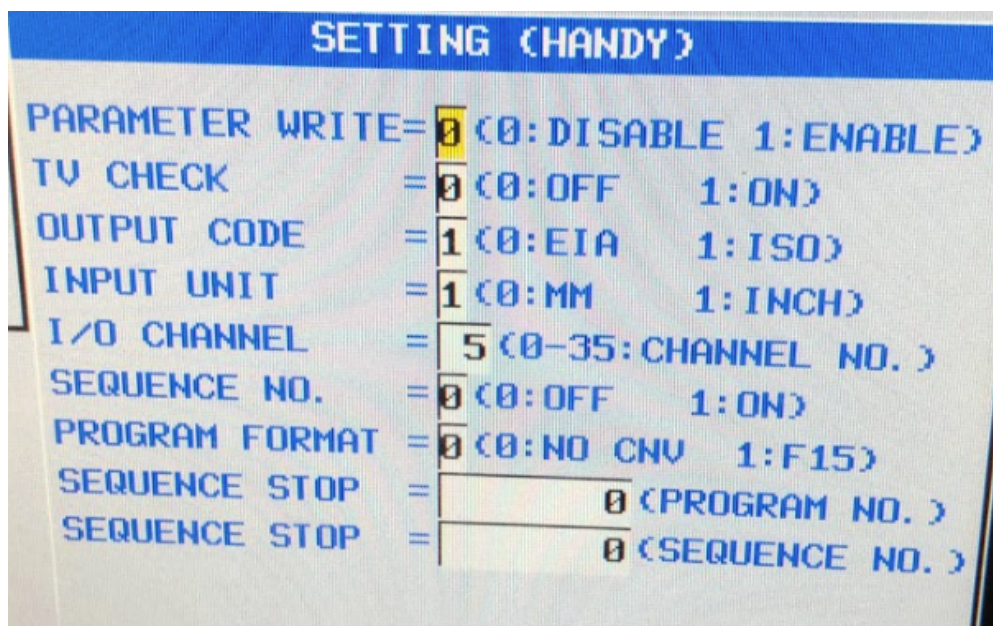
2.3.6. Hit the **Parameter** key, and enter “6019” then the No. SRH softkey in the bottom left corner. The below field should be highlighted. Hit the right arrow key until the OFN field is highlighted. Press “1” then “input”.



2.3.7. Return to the **Setting (Handy)** screen using the **key**. Highlight the Parameter Write field, type “0” and then “input”. Hit the Reset key



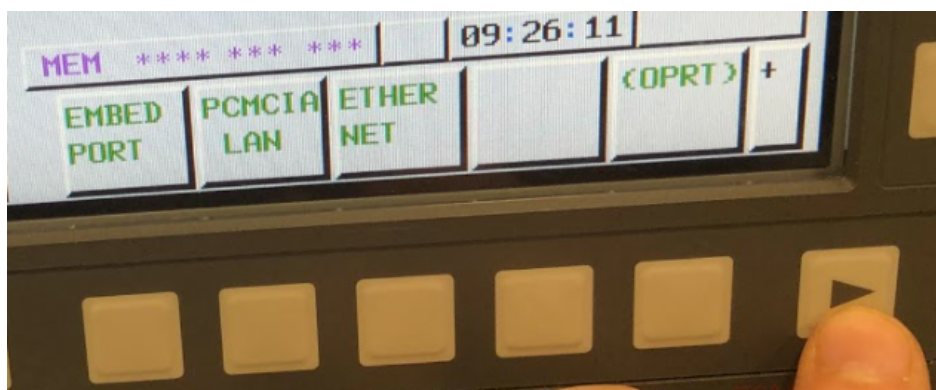
to clear the alarm before continuing.



- 2.3.8. Hit **green OPRT key**, then **Change Device** to **CNC Mem**. Use the arrow keys to highlight **O8000**. Hit the **green OPRT key**. Advance with the **right arrow soft key** until **F OUTPUT** is present. Hit **F OUTPUT**, then **F GET**, the **F SET**, then **EXEC**.

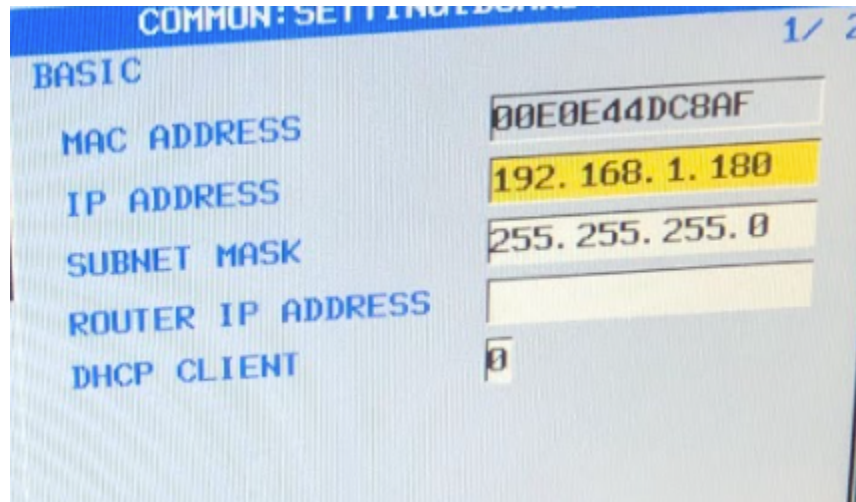


- 2.3.9. Hit the System Key then advance with the **right soft key** as shown until **EMBED PORT** appears on the softkey options.

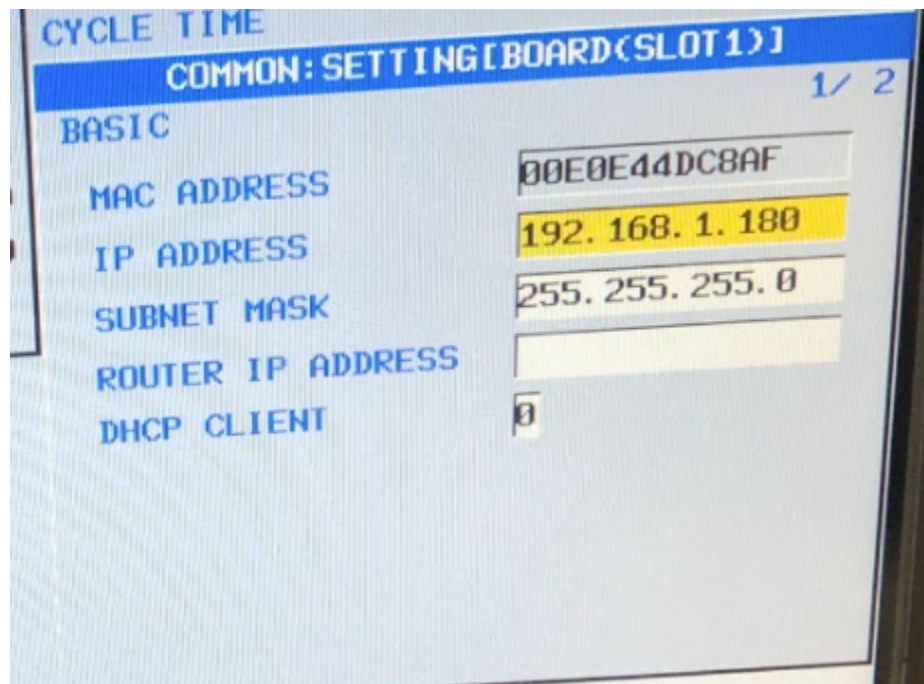




- 2.3.10. Hit the Embed Port Soft Key and then Common. Input the IP to **192.168.1.180** as shown.

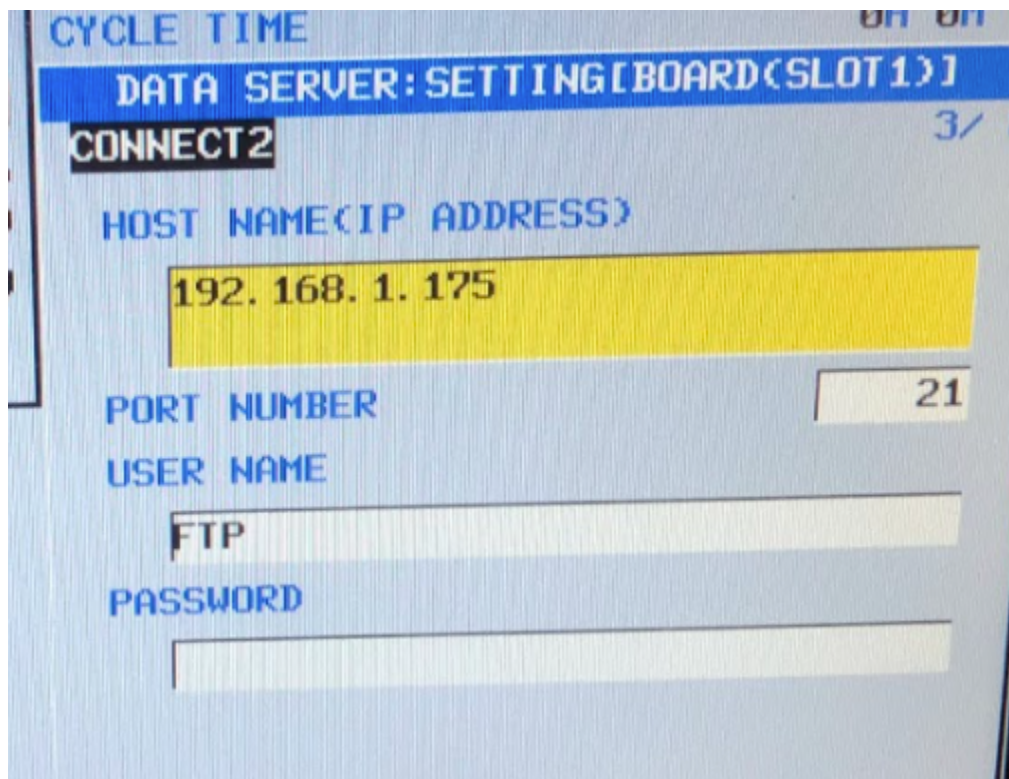


- 2.3.11. Return to the previous menu using the **left soft key**. Now select the **Ethernet** Soft Key and the Common. Verify that this IP address is also **192.168.1.180**

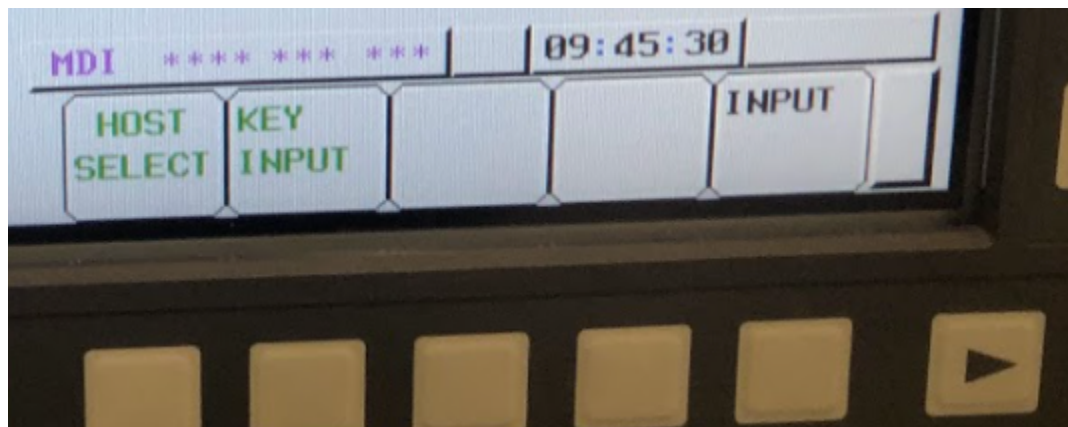


- 2.3.12. Now hit the **Data Server** soft key, hit down error until CONNECT 2 is on screen, make the host name **192.168.1.175**, and set the user name as

FTP as shown below.



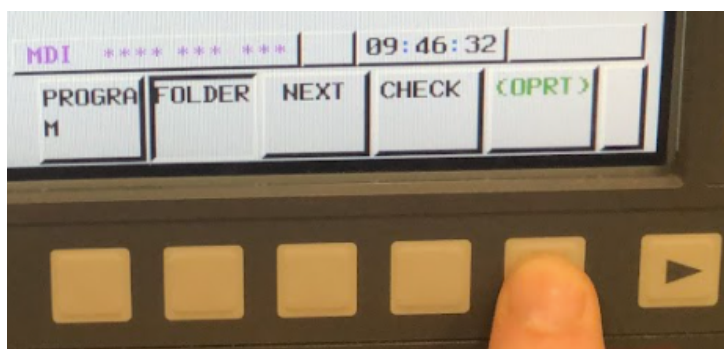
2.3.13. Hit **Host Select** soft key and verify the **CONNECT 2** is highlighted.

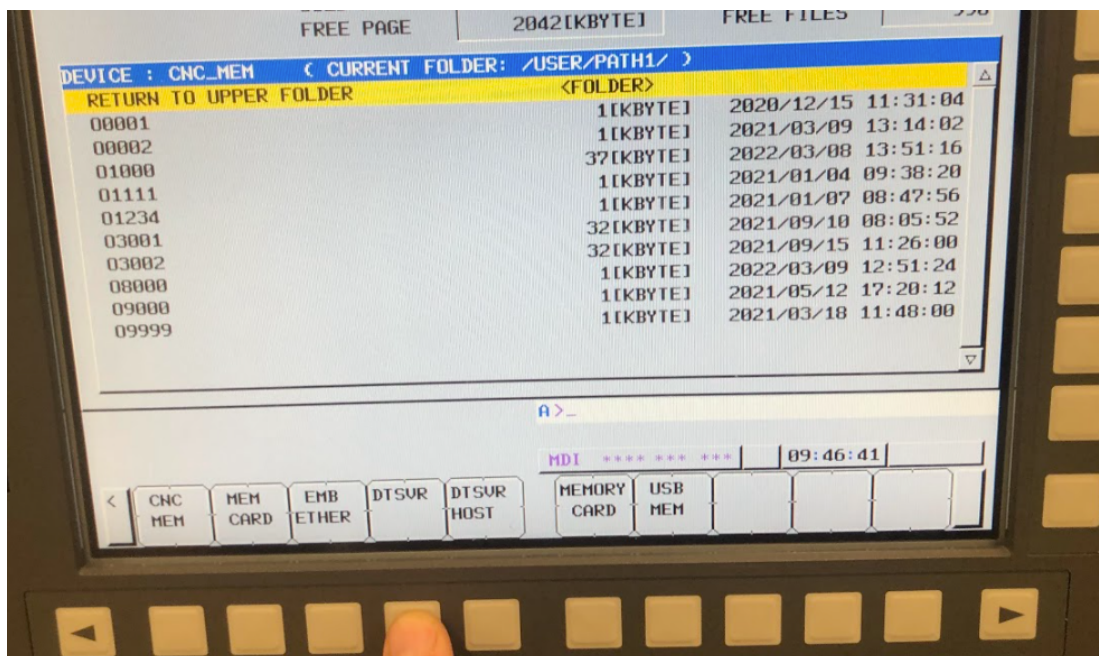


2.3.14. Switch to Edit Mode.



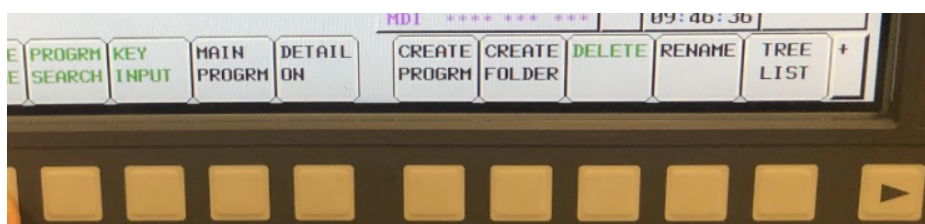
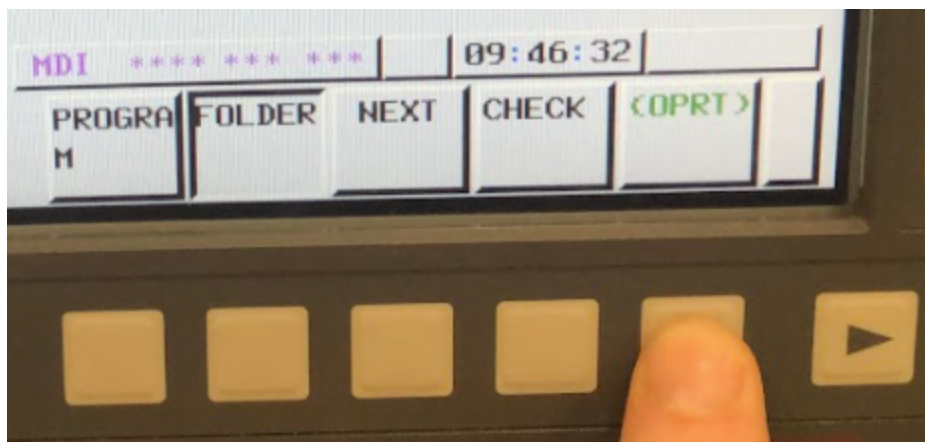
2.3.15. Hit the **Program Key**, then **green OPRT key**. Hit the **Change Device** soft key. Hit the **DTSVR** soft key.





- 2.3.16. All future programming should take place within this folder location, in the **DTSVR** tab so that the Flexx CNC can access your programs for loading.
- 2.3.17. Hit the **green OPRT key** and set **O8000** as your Main Program using the **Main Program** soft key. A small ampersand should appear next to the program.





- 2.3.18. At this point, the user can set the CNC into Memory mode, and run your program unattended.



## 2.4. FANUC G-Code Programs

### 2.4.1. Cycle End Detection

The Flexxbotics software automatically updates the selected g-code with snippets that are used to signal back to the UR. The software looks for the following codes in a selected g-code program to signal back to the robot that it is complete with an operation.

- M00 - a defined place in the g-code program to notify the UR that a break in the program is needed to perform a robot action.
- M30 - a defined place in the program to notify the UR that cycle is completed and to perform a robot action.
- #XX - where XX is a macro - a defined place in the g-code program where the user can define a break in the program is needed to perform a robot action. It can also be used in g-code calculations or IF statements.
- M99 - Used to signal the end of a subprogram call. When a program is called by a main program or sub program using M98, the program continues until reading M99. Implemented in a Main program, it will loop the program back to its start.

### 2.4.2. Writing Macros

The value of a macro can be written from the Flexx CNC UR Cap Write Macro Program Node. This will add a macro value assignment to the top of the g-code program prior to a cycle starting. The macro value can then be referenced in the g-code.

### 2.4.3. Dispatcher Program (Alternative to Multi-Program Loading)

An alternative to the program loading method above for multi-operation is to use a single dispatcher program. The single dispatcher program would contain IF

statements based on the value of a “selector” macro to perform a given g-code operation. The robot would then write the value of the “selector” macro in between g-code operations. In order to achieve this, at the end of each operations IF statement, an #XX macro would need to be defined for Cycle End Detection (see *Cycle End Detection* above).

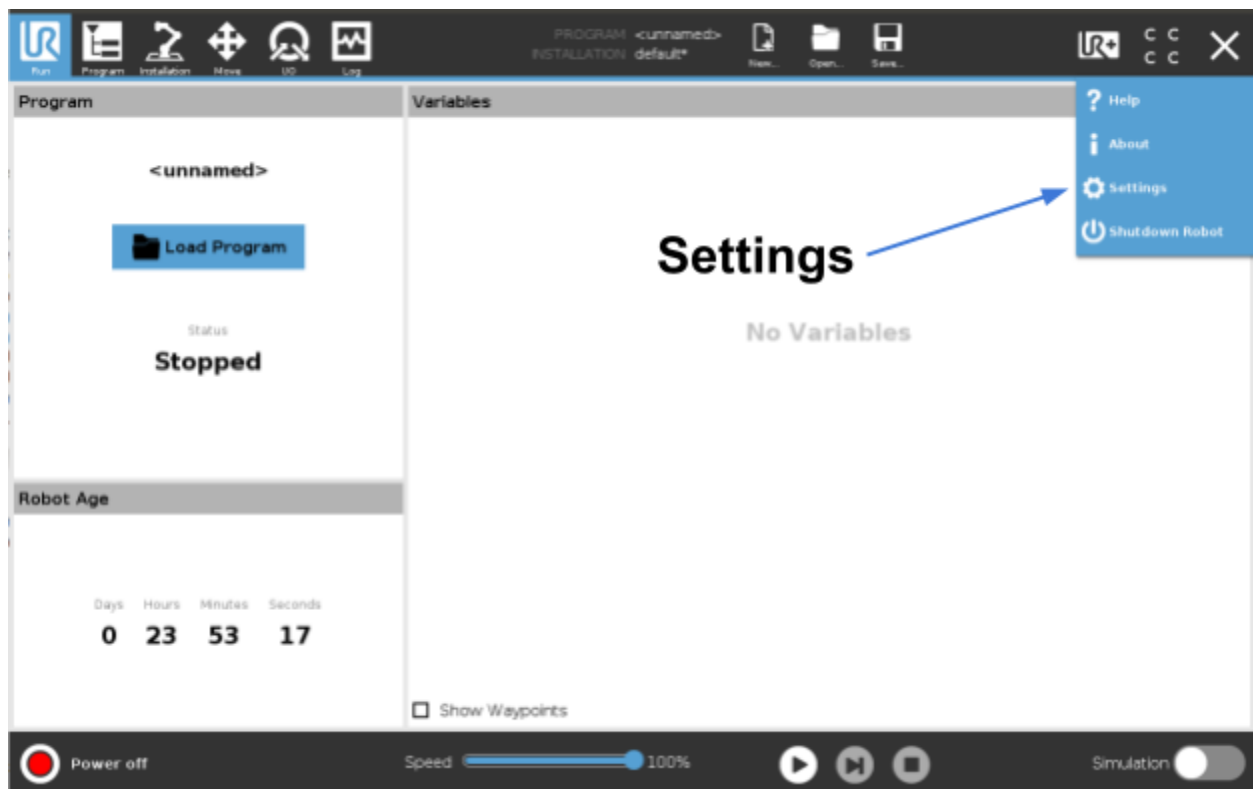
## 3. Flexx CNC URCap

### 3.1. Setup

#### 3.1.1. URCap Installation

3.1.1.1. Plug the USB into the Universal Robots teach pendant

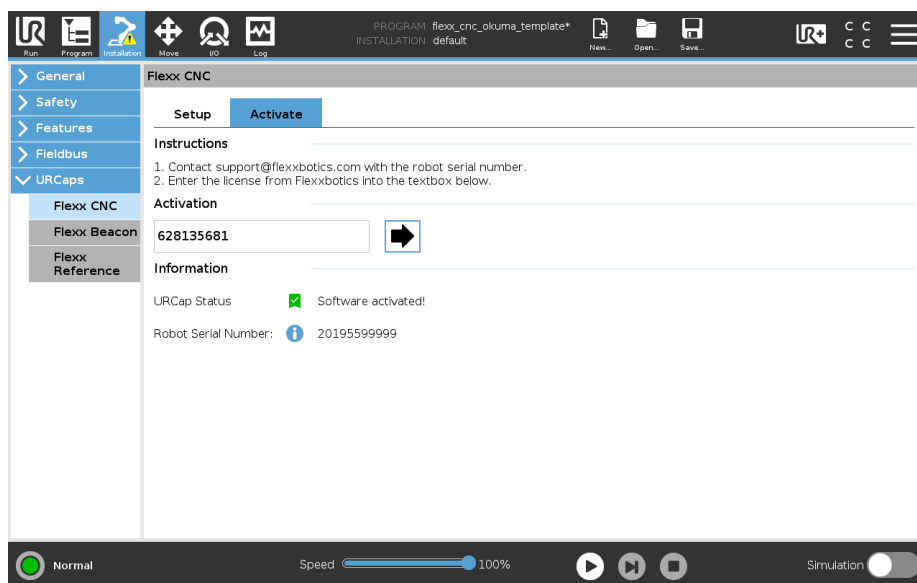
3.1.1.2. Navigate to the **Settings** page within Polyscope.



- 3.1.1.3. On the left side of the screen, click on the **System** tab.
- 3.1.1.4. Click the **plus (+)** button in the bottom left corner of the screen.
- 3.1.1.5. Open the USB device folder from the file explorer.
- 3.1.1.6. Select the Flexx CNC **URCap file** and click **Open**.
- 3.1.1.7. Verify that Flexx CNC now appears in the **Active URCaps** list with a green check mark.

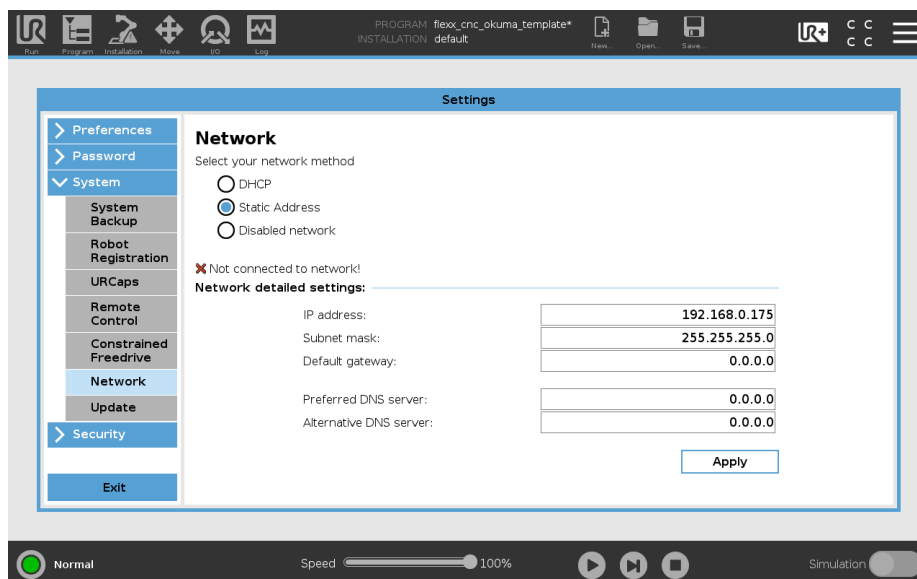
## 3.1.2. License Activation

- 3.1.2.1. Navigate to the **Installation Tab** within Polyscope.
- 3.1.2.2. On the left side of the screen, click on the **URCaps** tab.
- 3.1.2.3. Select the **Flexx CNC** urcap. Open the **Activate** tab.
- 3.1.2.4. Click the **“Enter a License”** text box to launch the keyboard.
- 3.1.2.5. Enter the license code provided by Flexxbotics and click **Submit**.



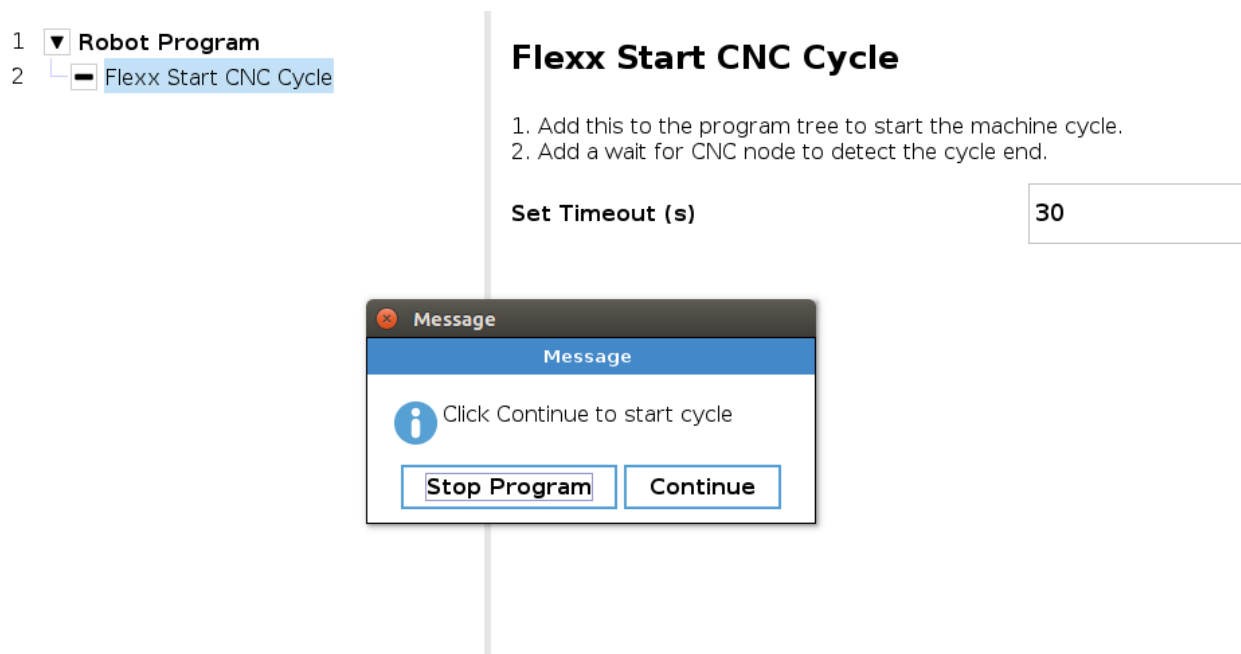
### 3.1.3. Robot Network Configuration

- 3.1.3.1. Navigate to the **Installation Tab** within Polyscope.
- 3.1.3.2. On the left side of the screen, click on the **System** tab. In the menu click on **Network**.
- 3.1.3.3. Configure the network as follows:
  - 3.1.3.3.1. Select Static Address
  - 3.1.3.3.2. IP address: 192.168.0.175
  - 3.1.3.3.3. Subnet Mask: 255.255.255.0
  - 3.1.3.3.4. Leave all other settings as 0.0.0.0
- 3.1.3.4. Click **Apply**.
- 3.1.3.5. Restart the controller.



## 3.2. Installation Node

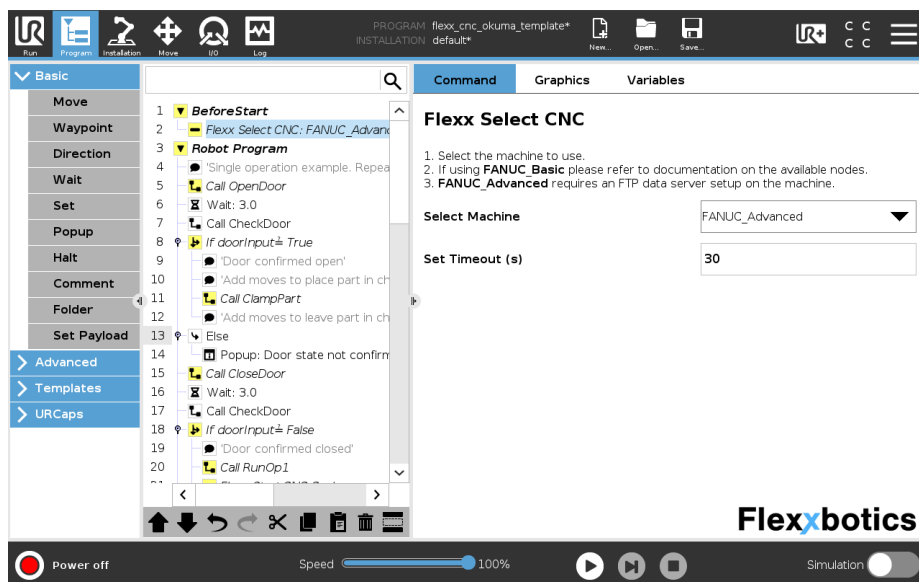
- 3.2.1. Select the type of interface being used as Ethernet or Digital I/O.
- 3.2.2. Ensure that Test Mode is check marked. This will provide the ability to step through the UR program to test that the interface is working correctly.  
  
In Test Mode, the program will prompt the user to Continue on every Flexx CNC node as following:



- 3.2.3. Once a program has been verified to be working, uncheck the Test Mode.  
  
This will remove the pop ups and allow the program to run continuously without user interaction.

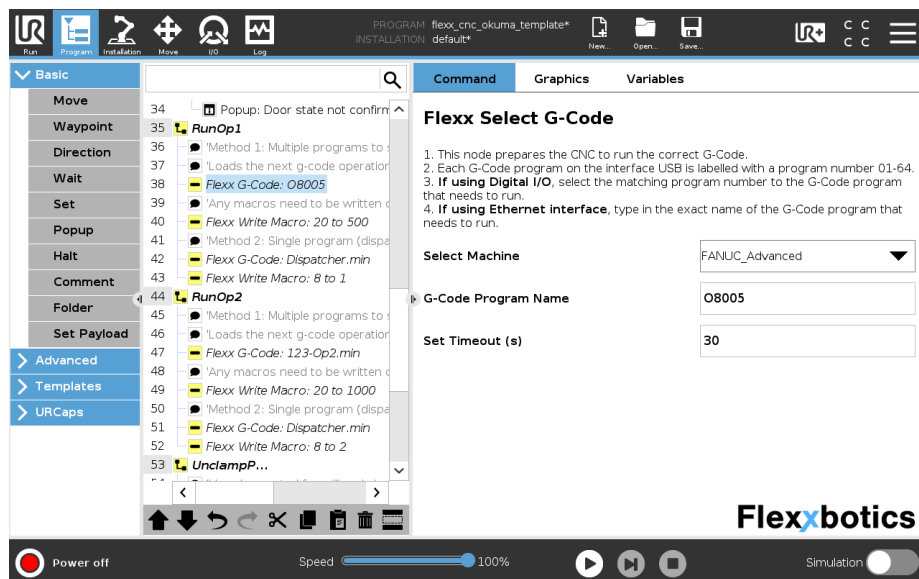
### 3.3. Overview of Program Nodes

3.3.1. **Flexx CNC Select Machine:** In the Before Start Sequence of the Robot Program to select the type of machine that the robot program is tending.

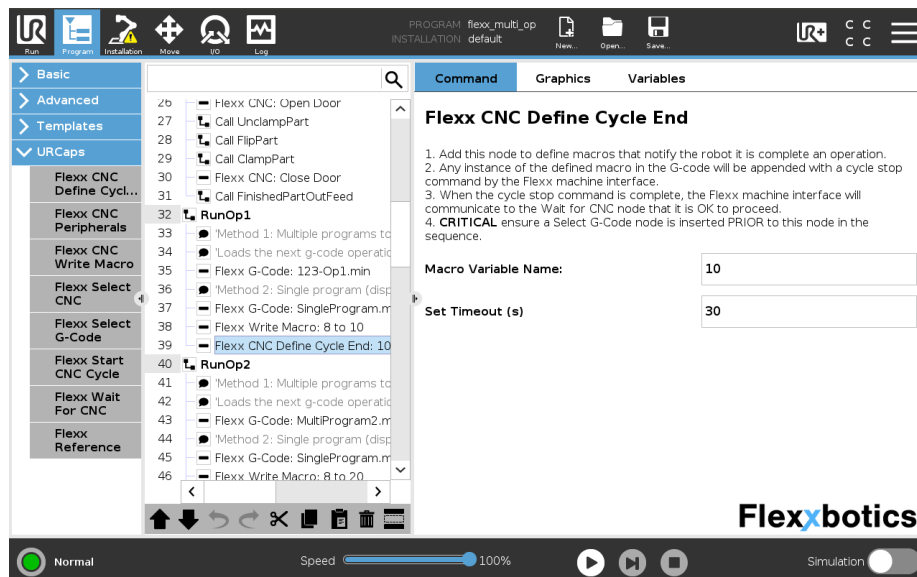


3.3.2. **Flexx Select G-Code:** Run the specified g-code program on the CNC on the next cycle start. For Fanuc, specify the g-code program name so that it matches EXACTLY the name of the program in the Data Server Folder.

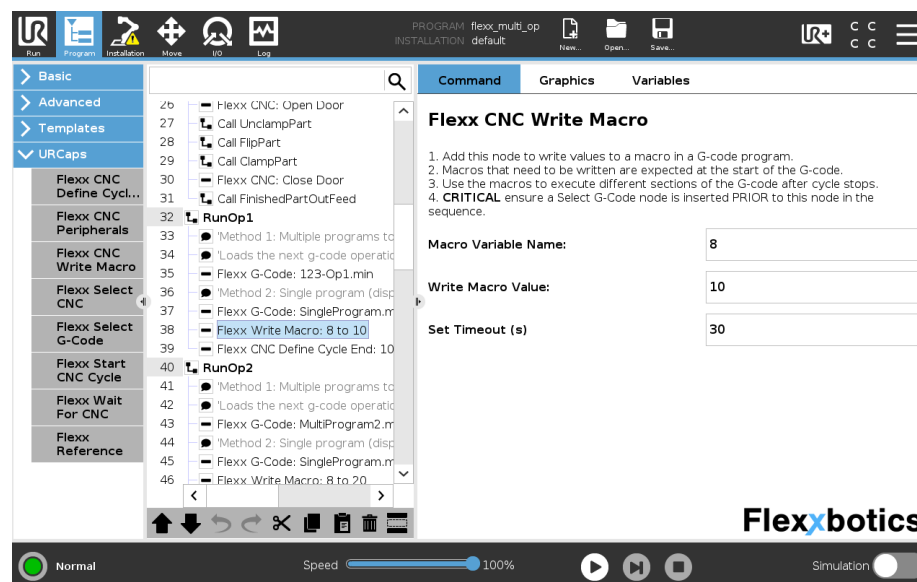
*Example if the program name is O8005:*



3.3.3. **Flexx CNC Define Cycle End:** Specify a macro that should be used for cycle end detection. This would likely only be used if you are using a dispatcher program to switch between operations.

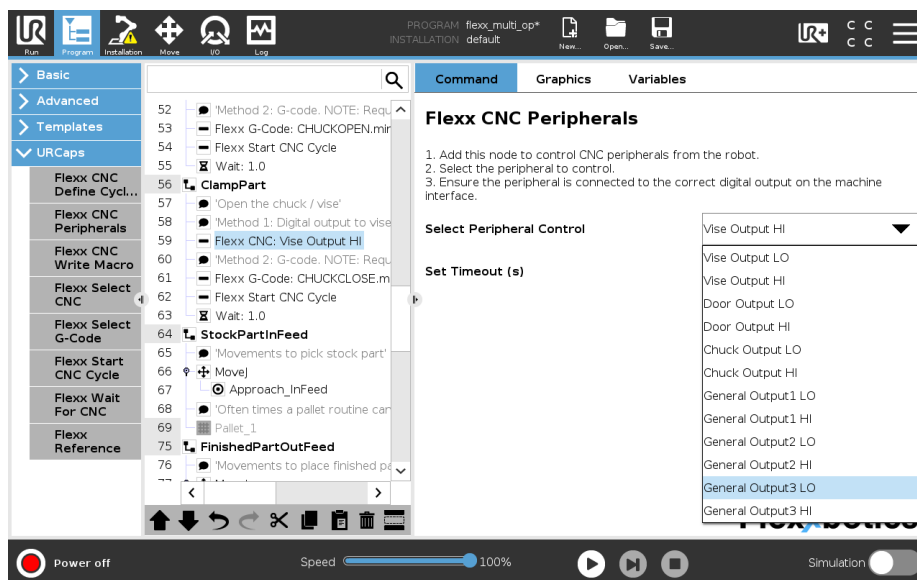


3.3.4. **Flexx CNC Write Macro:** Specify a value for a macro prior to the cycle start running. This is used to change variable parameters as well as initiate operation switching when using a dispatcher program. The Write Macro Value can either be specified as a numeric or a UR Script Variable.

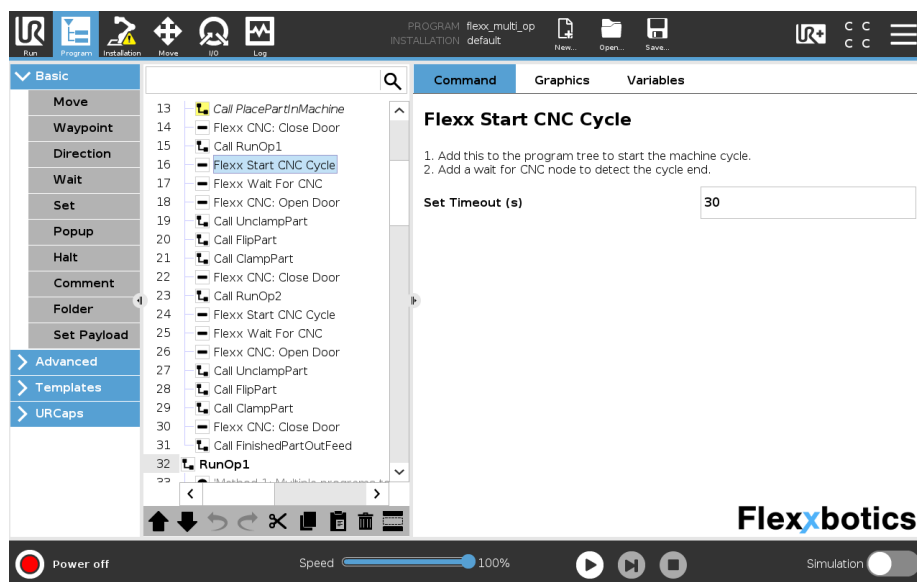




3.3.5. **Flexx CNC Peripherals:** Control peripheral devices from the Flexx Machine Interface. This node allows you to select which peripheral to control via a digital output. The list of peripherals coincide with the output labels on the Flexx Machine Interface enclosure.



3.3.6. **Flexx Start Cycle:** This will trigger the Cycle Start Output on the Flexx Machine Interface to trigger a cycle start on the machine.

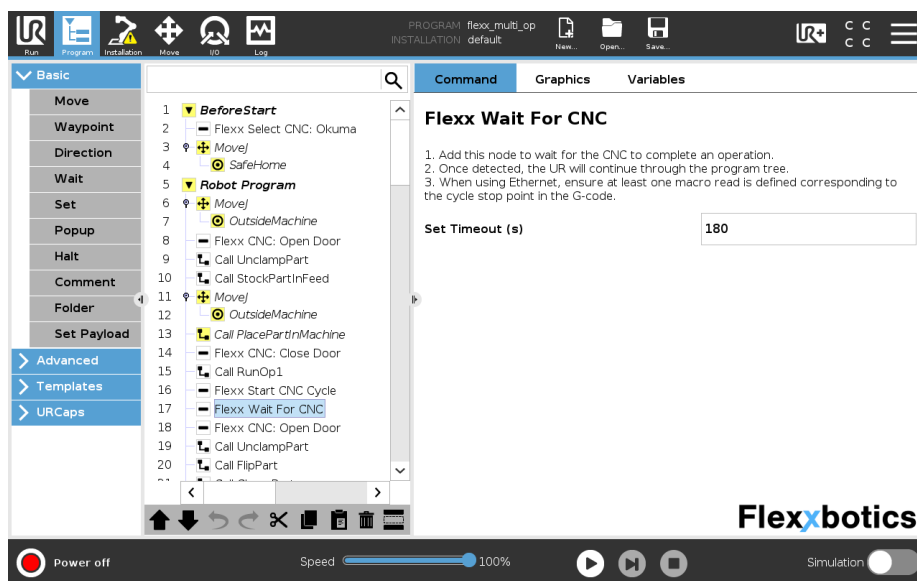


3.3.7. **Flexx Wait for CNC:** This node will wait for the CNC to complete running the g-code program or complete running an operation. It will wait at the node until a signal from the machine has been received.

For g-code completion, the M30 code is used to signal to the robot at this node that the program has completed (see *Cycle End Detection in the [FANUC G-Code Programs](#) section*).

For g-code operation completion (when the program has not completed, but the robot needs to perform a task), an #XX (defined macro) is used to signal to the robot that the machine has completed.

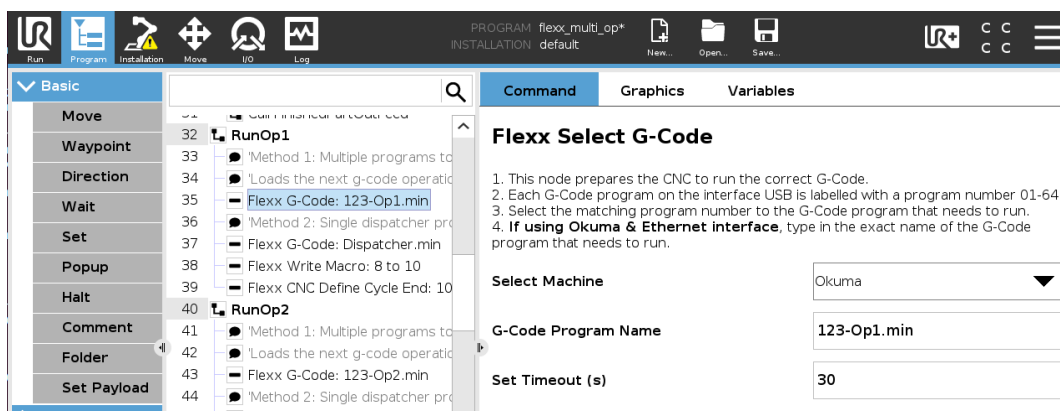
Once the signal is received, the robot program will continue. Depending on the cycle time of the program or operation, it is important to adjust the *Timeout* input to be greater than that cycle time.



## 4. Step by Step Programming Guide

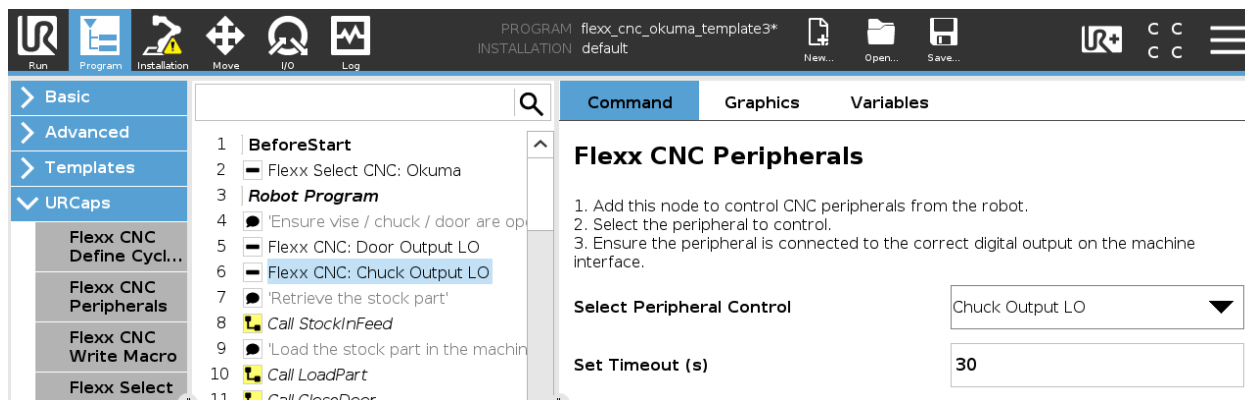
The following step by step explanation provides guidelines as to how to set up the robot to CNC communications. This example does not include how to program robot movements relative to the machine, but may provide some general insight for program architecture.

- Step 1: Add the **Select Machine Program Node** to the Before Start Sequence of the robot program. Select “FANUC” as the machine being used.
- Step 2: Create a **new SubFolder** called RunOp1. In the SubFolder:
  - Add a **Select G-Code Program Node** and select the g-code that needs to run.
  - Decision: Is the selected g-code a dispatcher program?
    - Add a **Write Macro Program Node** and write the value of the macro needed to perform Operation1 within the dispatcher program.
    - Decision: How are you performing cycle end detection?
      - M00 or M30? Do nothing.
      - Macro: Add a **Define Cycle End Program Node** and specify the macro being used for detection.
  - Decision: Do you need to write a macro value to the machine?
    - Add a **Write Macro Program Node** to write the value of the macro.
  - Decision: Are you performing multiple operations on the part?
    - Repeat Step 2 by creating a **new SubFolder** for the next operation.

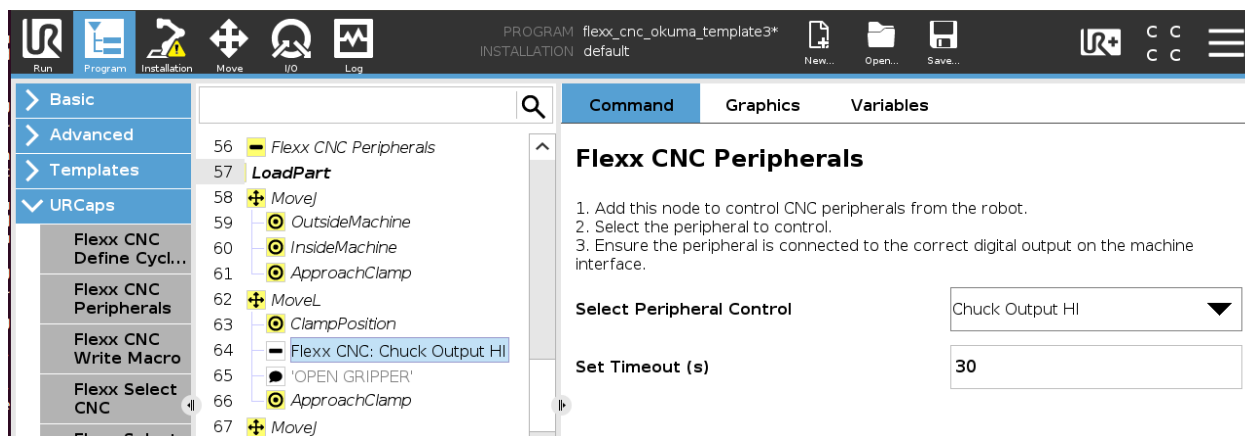


## Example of Step 2 (from template)

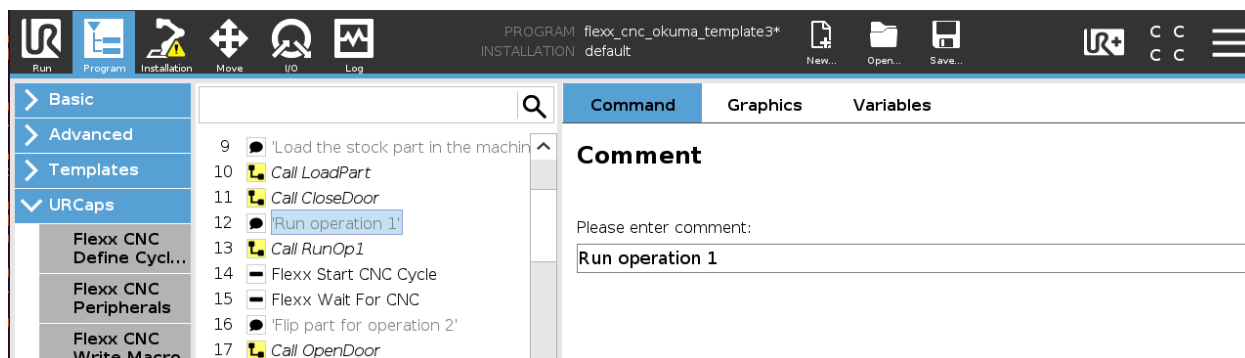
- Step 3: Ensure the door and vise / chuck are open in the Robot Program prior to starting any part load routines.
  - Add a **Flexx CNC Peripheral - Door Output LO** command (check the polarity of the AutoDoor being used. LO may indicate Door Open or Door Close; switch the Flexx CNC command to match the polarity.)
  - **Lathe:** Add a **Flexx CNC Peripheral - Chuck Output LO** command (check the polarity of the chuck input being used. LO may indicate Chuck Open or Chuck Close; switch the Flexx CNC command to match the polarity.).
  - **Mill:** Add a **Flexx CNC Peripheral - Vise Output LO** command (check the polarity of the vise input being used. LO may indicate Vise Open or Vise Close; switch the Flexx CNC command to match the polarity.).



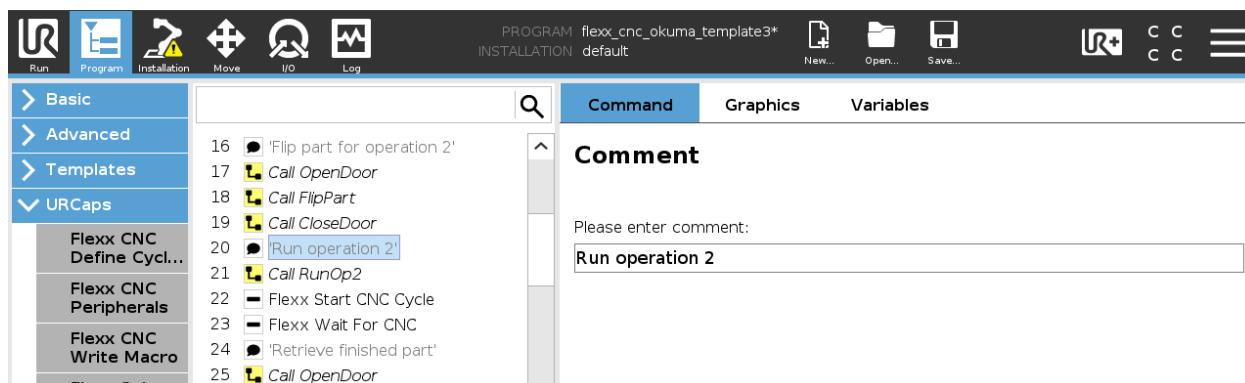
- Step 4: Program a routine to retrieve a stock part and load it into the machine.
  - **Lathe:** When loading the stock part in the chuck, utilize the **Flexx CNC Peripheral - Chuck Output HI** command to close the chuck (check the polarity of the chuck input being used. HI may indicate Chuck Open or Chuck Close; switch the Flexx CNC command to match the polarity.).
  - **Mill:** When loading the stock part in the vise, utilize the **Flexx CNC Peripheral - Vise Output HI** command to close the vise (check the polarity of the vise input being used. HI may indicate Vise Open or Vise Close; switch the Flexx CNC command to match the polarity.).



- Step 5: Call **RunOp1** in the Robot Program main loop.
- Step 6: Add a **Flexx Cycle Start** after Call RunOp1 in the Robot Program main loop.
- Step 7: Add a **Wait for CNC** after Cycle Start in the Robot Program main loop.



- Step 8: If running a multi-operation, it is likely that the part needs to be flipped after the first operation. Write a subprogram that flips the part. When flipping the part:
  - Use the **Flexx CNC Peripheral - Door Output** to control the Door
  - **Lathe:** Use the **Flexx CNC Peripheral - Chuck Output** to control the chuck to remove and place the part.
  - **Mill:** Use the **Flexx CNC Peripheral - Vise Output** to control the vise to remove and place the part.
- Once the part is flipped, repeat Steps 5, 6, and 7 in the Robot Program main loop to call the next operation and start the cycle. See example below:



- After the part has been completely machined, write a routine to Unload the part and place it into a finished part outfeed. During that process:
  - Use the **Flexx CNC Peripheral - Door Output** to control the Door
  - **Lathe:** Use the **Flexx CNC Peripheral - Chuck Output** to control the chuck to remove and place the part.
  - **Mill:** Use the **Flexx CNC Peripheral - Vise Output** to control the vise to remove and place the part.