

Program Template EXPLAINED

CNC Machine Tending Application



ROBOTIQ
eLearning

PROGRAM TEMPLATE CNC Machine Tending Application

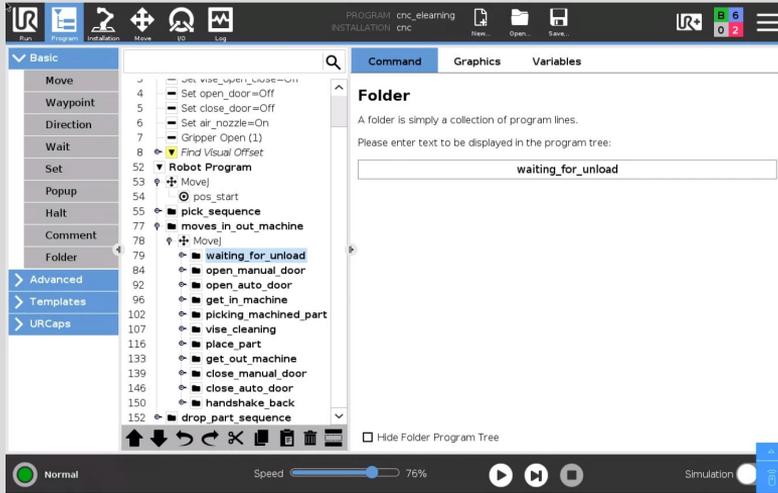




Description

Use this document to help you program your application using the template for the CNC Machine Tending Application. You can learn more about the steps on how to program your CNC machine tending application using the available videos for this course.

Visit elearning.robotiq.com for more details.



What you will need

- Robotiq CNC Machine Tending Kit
- Latest URcap - Gripper URcapLatest URcaps
 - Gripper URcap UCG-X.X.X.X
 - Copilot URcap UCS-X.X.X
- Universal Robot UR3e, UR5e, UR10e or UR16e
- Program Template: [cnc_elearning.urp](https://elearning.robotiq.com)

Folders in UR Programs

To insert a folder in your program, go to **Basic** → **Folder**. Once the folder is in the program, you can add nodes in it to build your program. In order to run only a certain part of the program, you can suppress the folders you do not want to run by selecting the folder and pressing on .

Folders in UR Programs

In the UR program tree, it is possible to insert a **folder** node in the program. This allows to separate your program into smaller sections and is very useful for:

- Making the program clearer and easier to understand
- Testing smaller sections of the program
- Reusing the same code throughout your program

We highly recommend you use this tool since it will make your life easier when building your application.

How to use the template

This template contains all the steps and more to automate your own CNC Machine Tending application. Since your setup will most likely be very different from ours, you will need to make sure to **re-teach all the waypoints** in the program. You should also change the names and select the right I/Os to best reflect your setup. This template is mostly a guide so that you do not forget any steps.

Feel free to suppress or even delete folders or lines if you really do not need them. However, if you delete anything, really make sure you will not need it. You can also add lines if required. This is first and foremost a learning tool designed to help you automate your first application, or practice automating a CNC machine. It contains everything you need for a basic integration.



Before Start Sequence

1

Before Start

This sequence will execute before the main program.

- 1 Set load_complete=Off
- 2 Set vise_open_close=Off
- 3 Set open_door=Off
- 4 Set close_door=Off
- 5 Set air_nozzle=On
- 6 Gripper Open (1)
- 8 Find Visual Offset
 - 9 If tag found
 - 10 Apply Visual Offset
 - 11 'Insert actions here'
 - 12 open_auto_door
 - 16 get_in_machine
 - 22 Find Contact Offset
 - 49 If tag not found
 - 50 'Optional'
 - 51 Popup: tag not found
- 52 Robot Program
 - 53 Move
 - 55 pick_sequence
 - 77 moves_in_out_machine
 - 78 Move
 - 79 Waypoint_2
 - 80 Waypoint_2

1 **Before Start** is used to execute commands before starting the main loop of the program. This is where all the variables are first set and where you will find the **visual** and/or **contact offset**. This section will only be done once.



Before Start Sequence

The screenshot displays a software interface for a robot program. The main window shows a sequence of actions under the heading "BeforeStart". The actions are listed as follows:

- 1 BeforeStart
- 2 Set load_complete=Off
- 3 Set vise_open_close=Off
- 4 Set open_door=Off
- 5 Set close_door=Off
- 6 Set air_nozzle=On
- 7 Gripper Open (1)
- 8 Find Visual Offset
- 9 If tag found
- 10 Apply Visual Offset
- 11 'Insert actions here'
- 12 open_auto_door
- 16 get_in_machine
- 22 Find Contact Offset
- 49 If tag not found
- 50 'Optional'
- 51 Popup: tag not found
- 52 Robot Program
- 53 Move
- 55 pick_sequence
- 77 moves_in_out_machine
- 78 Move
- 79 Waypoint_2
- 80 Waypoint_2

A red circle with the number "2" is positioned over the "BeforeStart" sequence name in the left-hand pane. The right-hand pane shows the text: "This sequence will execute before the main program."

2 In this section we can see all the variables related to the handshake, as well as door management and air nozzle. These **Set** functions will have to be modified to fit with your own signal exchange configuration and wiring.



Before Start Sequence

The screenshot displays a robot programming software interface. The main window shows a sequence of actions under the heading "BeforeStart". The sequence is as follows:

1. BeforeStart
2. Set load_complete=Off
3. Set vise_open_close=Off
4. Set open_door=Off
5. Set close_door=Off
6. Set air_nozzle=On
7. Gripper Open (1)
8. Find Visual Offset (highlighted with a circled '3')
9. If tag found
 - 10. Apply Visual Offset
 - 11. 'Insert actions here'
 - 12. open_auto_door
 - 16. get_in_machine
 - 22. Find Contact Offset
 - 49. If tag not found
 - 50. 'Optional'
 - 51. Popup: tag not found
52. Robot Program
 - 53. Move
 - 55. pick_sequence
 - 77. moves_in_out_machine
 - 78. Move
 - 79. Waypoint_2
 - 80. Waypoint_2

The interface also features a top toolbar with icons for Run, Program, Installation, Move, I/O, and Log. A search bar and tabs for Command, Graphics, and Variables are visible. The bottom status bar shows "Normal" mode, a speed slider at 100%, and simulation controls.

3 The first gripper is then opened and a **Find Visual Offset** node is added. This is used to position the robot with respect to the machine. A contact offset node is then used for an even more precise positioning.

The visual and contact offset can be removed if your robot is always precisely positioned relative to the rest of the cell.



Before Start Sequence

The screenshot shows a software interface for programming a robot. The main window displays a sequence of actions under the heading "BeforeStart". The actions are listed as follows:

- 1. BeforeStart
- 2. Set load_complete=Off
- 3. Set vise_open_close=Off
- 4. Set open_door=Off
- 5. Set close_door=Off
- 6. Set air_nozzle=On
- 7. Gripper Open (1)
- 8. Find Visual Offset
- 9. If tag found
- 10. Apply Visual Offsets (highlighted with a callout box containing the number 4)
- 11. Insert actions
- 12. open_auto_door (folder)
- 16. get_in_machine (folder)
- 22. Find Contact Offset
- 49. If tag not found
- 50. Optional
- 51. Popup: tag not found
- 52. Robot Program
- 53. Move
- 55. pick_sequence (folder)
- 77. moves_in_out_machine (folder)
- 78. Move
- 79. Waypoint_2
- 80. Waypoint_2

The callout box with the number 4 points to the "Apply Visual Offsets" action, which is a sub-action of the "If tag found" condition. The "open_auto_door" folder is also visible in the sequence.

4 There are also **folder** used for opening the door and getting in the machine. These will be covered later in the main program.



Robot Program

PROGRAM: cnc_elearning
INSTALLATION: cnc

Run Program Installation Move I/O Log

Basic Command Graphics Variables

Move
Waypoint
Direction
Wait
Set
Popup
Halt
Comment
Folder

Advanced
Templates
URCaps

52 Robot Program 5
53 Move
54 pos_start
55 pick_sequence
77 moves_in_out_machine
78 Move
79 waiting_for_unload
84 open_manual_door
92 open_auto_door
96 get_in_machine
102 picking_machined_part
107 vise_cleaning
116 place_part
133 get_out_machine
139 close_manual_door
146 close_auto_door
150 handshake_back
152 drop_part_sequence

Folder
A folder is simply a collection of program lines.
Please enter text to be displayed in the program tree:
waiting_for_unload

Normal Speed 76% Simulation

- 5 The **Robot Program** is where all the main actions are programmed. It is divided in three main sequences:
- Pick a raw part
 - Move in and out of the machine
 - Drop the machined part

The second sequence is then further divided in different smaller sequences.



Robot Program

PROGRAM: cnc_learning
INSTALLATION: cnc

Run Program Installation Move I/O Log

Command Graphics Variables

Basic

- Move
- Waypoint
- Direction
- Wait
- Set
- Popup
- Halt
- Comment
- Folder
- Advanced
- Templates
- URCaps

52 Robot Program

- 53 MoveJ
- 54 pos_start
- 55 pick_sequence **6**
- 77 moves_in_out_machine
- 78 MoveJ
- 79 waiting_for_unload
- 84 open_manual_door
- 92 open_auto_door
- 96 get_in_machine
- 102 picking_machined_part
- 107 vise_cleaning
- 116 place_part
- 133 get_out_machine
- 139 close_manual_door
- 146 close_auto_door
- 150 handshake_back
- 152 drop_part_sequence

Folder

A folder is simply a collection of program lines.

Please enter text to be displayed in the program tree:

waiting_for_unload

Hide Folder Program Tree

Normal Speed 76% Simulation

6

The **pick sequence** picks object from the tray using the **Gripper 1**. This sequence uses a UR palletizing node to loop on each part in the tray.



Pick sequence

The screenshot displays the Robotiq software interface for configuring a pick sequence. The main workspace shows a tree view of the program structure. A node labeled 'Pallet_2' is highlighted with a circled '7'. The 'Palletizing' section is active, showing 'Pallet Properties' with fields for Name (Pallet_2), Feature (Base), Object Height (25.00 mm), and Item Counter (Pallet_2_cnt). The 'Actions' section has checkboxes for 'Add action before palletizing' and 'Add action after palletizing'. The interface includes a menu on the left, a toolbar at the top, and a status bar at the bottom showing 'Normal' and 'Speed 44%'.

7 The only node in the **folder** is a **Pallet** node. If you parts are on a tray and evenly separated, using a pallet allows to easily loop on each part without too much effort. Here we use a line pattern, but feel free to modify this node to fit your own tray.



Pick sequence

The screenshot shows the Robotiq software interface during a pick sequence configuration. The main window displays the 'At Each Item' wizard, which is completed (6 / 6). The wizard text states: 'The At Each Item sequence is now complete. Add the appropriate gripper action nodes in the Tool Action folder in the Program Tree.' The left sidebar shows a tree view of the program structure, with the 'At Each Item' folder highlighted and circled with the number 8. The bottom status bar indicates the robot is in 'Normal' mode, with a speed of 44% and simulation controls.

8 Once the **Patterns** are set for the pallet, it is time to teach how each item of the pallet will be treated.

First in the **Generated Movements** folder, we teach the approach and pick position with the gripper 1. Once at the pick point, a **Tool Action** folder is inserted where all the gripper actions are called. Here we close the **Gripper 1** and we do a **grip check** to make sure a part has been picked.



Pick sequence

The screenshot displays the Robotiq software interface. On the left, a tree view shows the program structure for 'pick_sequence', with the 'At Each Item' folder highlighted and a circled '9' next to it. The right pane shows a 'Wizard Finished (6 / 6)' dialog box with the text: 'The At Each Item sequence is now complete. Add the appropriate gripper action nodes in the Tool Action folder in the Program Tree.' Below the dialog are 'Previous', 'Next', 'Finish', and 'Cancel' buttons. The bottom status bar shows 'Normal' mode, a speed slider at 44%, and 'Simulation' mode.

9 Once the tool action is complete, the next move is an exit point.

When building your application, simply re-teach the points corresponding to your own application.

The **ToolActionPoint** should be the point where the gripper picks the part. The **Approach** and **Exit** should be a few centimeters above the part to be picked.



Move in and out of the machine

The screenshot shows the Robotiq software interface. The top toolbar includes icons for Run, Program, Installation, Move, I/O, and Log. The main window is divided into several sections:

- Left Panel:** A sidebar with a 'Basic' tab and a list of actions: Move, Waypoint, Direction, Wait, Set, Popup, Halt, Comment, Folder, Advanced, Templates, and URCaps.
- Program Tree:** A list of program lines with a tree view. A folder named 'moves_in_out_machine' is highlighted, and a callout box with the number '10' points to it. The tree includes folders like 'waiting_for_unload', 'open_manual_door', 'open_auto_door', 'get_in_machine', 'picking_machined_part', 'vise_cleaning', 'place_part', 'get_out_machine', 'close_manual_door', 'close_auto_door', 'handshake_back', and 'drop_part_sequence'.
- Command Panel:** A panel with tabs for 'Command', 'Graphics', and 'Variables'. It contains a 'Folder' section with a text input field containing 'waiting_for_unload'.
- Bottom Panel:** A control bar with a 'Normal' status indicator, a 'Speed' slider set to 76%, and playback controls (play, stop, pause).

10 The **folder** called **move_in_out_machine** is the most complex part of the program. This involves all the communication with other items of the cell and a lot of movement. It is why it has been split in different folders.



Move in and out of the machine

PROGRAM: cnc_learning
INSTALLATION: cnc

Run Program Installation Move I/O Log

Command Graphics Variables

Basic

Move
Waypoint
Direction
Wait
Set
Popup
Halt
Comment
Folder

Advanced
Templates
URCaps

Folder

waiting_for_unload

11

Normal Speed 76% Simulation

11

The first **folder** contains the movements required to position the robot in front of the CNC.

Then a **Wait** command is used. This command wait for the CNC to send the request to unload the machine.

The **Gripper 2** is then closed. This part is only useful when opening and closing the door manually.



Move in and out of the machine

PROGRAM: cnc_learning
INSTALLATION: cnc

Basic | Command | Graphics | Variables

Move | Waypoint | Direction | Wait | Set | Popup | Halt | Comment | Folder

Advanced | Templates | URCaps

12

waiting_for_unload

Speed 76%

Simulation

12 Opening the door can be done two different ways: **manually** or with an **automatic door**.

In the template, both options are programmed, so make sure to **suppress** the one you will not be using.



Move in and out of the machine

The screenshot shows the RobotStudio software interface. The top menu bar includes 'Run', 'Program', 'Installation', 'Move', 'I/O', and 'Log'. The main window is divided into several panes. On the left, there is a 'Basic' pane with a list of commands: Move, Waypoint, Direction, Wait, Set, Popup, Halt, Comment, Folder, Advanced, Templates, and URCaps. The 'Move' command is selected. The central pane shows a program tree with the following structure:

- pos_start
- pick_sequence
- moves_in_out_machine
 - Move
 - waiting_for_unload
 - open_manual_door (highlighted with a callout '13')
 - Move
 - Waypoint_4
 - Waypoint_1
 - Waypoint_1
 - Waypoint_6
 - MoveL
 - Waypoint_6
 - open_auto_door
 - get_in_machine
 - picking_machined_part
 - vise_cleaning
 - place_part
 - get_out_machine
 - close_manual_door
 - close_auto_door
 - handshake_back
 - drop_part_sequence

The right pane is titled 'Folder' and contains the text 'open_manual_door'. The bottom status bar shows 'Normal' mode, a speed slider at 76%, and a 'Simulation' button.

13 To open the door manually, first move the closed gripper near the door handle. Then use a **MoveL** to open the door.

Make sure to move back in a position ready to enter the CNC.



Move in and out of the machine

The screenshot displays the RobotStudio software interface. On the left, a vertical menu lists various commands: Move, Waypoint, Direction, Wait, Set, Popup, Halt, Comment, Folder, Advanced, Templates, and URCaps. The main workspace shows a hierarchical tree of program elements. A red circle with the number 14 highlights the 'open_auto_door' folder. The 'Command' tab is active, showing a text input field with 'open_auto_door' entered. The status bar at the bottom shows 'Normal' mode and a speed of 76%.

14 To open the door with an automated system, you simply need to send the signal to open the door via the proper **digital output**.

Then a **Wait** command is inserted where the robot waits for the door to be fully open.

Finally, the **open_door** signal is set to off.



Move in and out of the machine

PROGRAM: cnc_elearning
INSTALLATION: cnc

Run Program Installation Move I/O Log

Command Graphics Variables

Basic

- Move
- Waypoint
- Direction
- Wait
- Set
- Popup
- Halt
- Comment
- Folder
- Advanced
- Templates
- URCaps

Robot Program

- Move
- pos_start
- pick_sequence
- moves_in_out_machine
 - Move
 - waiting_for_unload
 - open_manual_door
 - open_auto_door
 - get_in_machine (15)
 - Move
 - Waypoint_7
 - Waypoint_8
 - Gripper Open (2)
 - Waypoint_9
 - picking_machined_part
 - vise_cleaning
 - place_part
 - get_out_machine
 - close_manual_door
 - close_auto_door
 - handshake_back
 - drop_part_sequence

Folder

A folder is simply a collection of program lines.

Please enter text to be displayed in the program tree:

get_in_machine

Normal Speed 76% Simulation

15 The next **folder** is used to get the robot in position for picking the completed part. The first two points are used to get over the part with the second gripper.

Then the **Gripper 2** is opened and moved down to the part.



Move in and out of the machine

PROGRAM: cnc_elearning
INSTALLATION: cnc

Basic

Move
Waypoint
Direction
Wait
Set
Popup
Halt
Comment
Folder

Advanced
Templates
URCaps

8 Find Visual Offset
52 Robot Program
53 Move
54 pos_start
55 pick_sequence
77 moves_in_out_machine
78 Move
79 waiting_for_unload
84 open_manual_door
92 open_auto_door
96 get_in_machine
102 picking_machined_part
103 Set vise_open_close=Off
104 Wait: 0.1
105 Gripper Close (2)
106 Waypoint_15
107 vise_cleaning
116 place_part
133 get_out_machine
139 close_manual_door
146 close_auto_door
150 handshake_back

Folder
A folder is simply a collection of program lines.
Please enter text to be displayed in the program tree:
picking_machined_part

Normal Speed 76% Simulation

16 After moving the **Gripper 2** in position, the robot sends the signal to open the vise. After a **0.1 Sec Wait time**, the **Gripper 2** closes on the part to pick it and move away from the vise.

The signal sent to open or close the vise depends on your setup and should be adjusted accordingly.



Move in and out of the machine

PROGRAM cnc_elearning
INSTALLATION cnc

Run Program Installation Move I/O Log

Basic

Move
Waypoint
Direction
Wait
Set
Popup
Halt
Comment
Folder

Advanced
Templates
URCaps

Command Graphics Variables

Folder

A folder is simply a collection of program lines.
Please enter text to be displayed in the program tree:

picking_machined_part

Hide Folder Program Tree

Normal Speed 76% Simulation

17 After picking the machined part, it is important to clean the vise before putting a new blank part.

To do this, simply orient the **air nozzle** toward the vise and activate it. After that move the robot around to make sure it really cleans the vise properly. Then simply turn off the air nozzle.



Move in and out of the machine

The screenshot shows the Robotiq software interface. The top menu bar includes 'Run', 'Program', 'Installation', 'Move', 'I/O', and 'Log'. The main window is divided into a left sidebar with a 'Basic' tab, a central program tree, and a right-hand 'Command' panel. The program tree lists various steps from 84 to 132, including 'open_manual_door', 'open_auto_door', 'get_in_machine', 'picking_machined_part', 'vise_cleaning', 'place_part', and a 'Move' folder containing 'Waypoint_15', 'Waypoint_10', and 'Find Surface'. A red circle with the number '18' highlights the 'Find Surface' folder. The 'Command' panel shows a 'Folder' description and a text input field containing 'pick_sequence'. The bottom status bar shows 'Normal' speed, a 76% speed slider, and a 'Simulation' button.

18 Placing the part in the vise can be done using a few different methods. Here we use the **find surface** feature as well as the **force control** feature of the Robotiq copilot.

First the part is moved over the vise and the **find surface** command is used to find the bottom and sides of the vise.



Move in and out of the machine

The screenshot shows the UR+ software interface. The top bar includes icons for Run, Program, Installation, Move, I/O, and Log, along with program information (PROGRAM: cnc_learning, INSTALLATION: cnc) and file operations (New..., Open..., Save...). The left sidebar has a tree view with categories: Basic, Advanced, Templates, and URCaps. The 'Folder' node is selected. The main window displays the program tree for the 'Folder' node, showing a sequence of steps from 84 to 132. A red circle with the number 19 highlights the 'Force Control' node at step 128, which contains a 'Set vise_open_close=On' node, a 'Wait: 0.2' node, and a 'Gripper Open (1)' node. The bottom status bar shows 'Normal' mode, a speed slider at 76%, and simulation controls.

19 A **force control** node is then inserted to make the robot compliant when closing the vise. This allows the robot to move with the part when the vise is closing, preventing any protective stop due to a misalignment of the part in the vise.

After the vise is closed, the gripper releases the part and moves away from it.



Move in and out of the machine

The screenshot shows a CNC control software interface. The top toolbar includes icons for Run, Program, Installation, Move, I/O, and Log. The main window is divided into a left sidebar with a 'Basic' tab, a central program tree, and a right 'Folder' editor. The program tree shows a 'Robot Program' starting at line 52, with a 'moves_in_out_machine' folder at line 77. A 'Move' folder is expanded at line 133, containing waypoints 21, 22, 23, and 28. A red circle with the number '20' highlights the 'Move' folder. The 'Folder' editor on the right shows a text input field containing 'get_out_machine'. The bottom status bar shows 'Normal' mode, a speed slider at 76%, and a 'Simulation' button.

20 This **folder** simply contains the movements required to get out of the machine. It is important to be completely out and leave enough space for the door to close.



Move in and out of the machine

The screenshot shows the RobotStudio software interface. The top menu bar includes options like Run, Program, Installation, Move, I/O, and Log. The main window is divided into a left sidebar with a 'Basic' tab, a central program tree, and a right-hand 'Command' panel. The program tree on the left shows a hierarchy starting with 'Robot Program', followed by 'Move|', 'pos_start', 'pick_sequence', and a folder named 'moves_in_out_machine' which is highlighted. Below this folder are several sub-folders: 'waiting_for_unload', 'open_manual_door', 'open_auto_door', 'get_in_machine', 'picking_machined_part', 'vise_cleaning', 'place_part', 'get_out_machine', 'close_manual_door', 'close_auto_door', 'handshake_back', and 'drop_part_sequence'. A callout box with the number '21' is positioned over the 'moves_in_out_machine' folder. The right-hand panel shows a 'Folder' description and a text input field containing 'moves_in_out_machine'. The bottom status bar shows 'Normal' mode, a speed slider at 76%, and simulation controls.

21 Here are included the two methods to close the door. **Make sure to suppress the one you are not using.**



Move in and out of the machine

The screenshot displays the RobotStudio software interface. The top menu bar includes options like Run, Program, Installation, Move, I/O, and Log. The main window is divided into a left sidebar with a 'Basic' tab, a central program tree, and a right-hand 'Folder' dialog box. The program tree shows a sequence of steps: 53 Move, 54 pos_start, 55 pick_sequence, 77 moves_in_out_machine, 78 Move, 79 waiting_for_unload, 84 open_manual_door, 92 open_auto_door, 96 get_in_machine, 102 picking_machined_part, 107 vise_cleaning, 116 place_part, 133 get_out_machine, 139 close_manual_door (highlighted with a red circle and the number 22), 140 Move, 141 Waypoint_11, 142 Waypoint_12, 143 Waypoint_6, 144 MoveL, 145 Waypoint_1, 146 close_auto_door, 150 handshake_back, and 152 drop_part_sequence. The 'Folder' dialog box is open, showing a text input field containing 'close_manual_door'. The bottom status bar shows 'Normal' mode, a speed slider at 76%, and a 'Simulation' button.

22 The manual closing works the same way as the manual opening. The first movements position the gripper next to the handle and the last linear movement closes the door.



Move in and out of the machine

PROGRAM cnc_elearning
INSTALLATION cnc

Run Program Installation Move I/O Log

Basic Command Graphics Variables

Move
Waypoint
Direction
Wait
Set
Popup
Halt
Comment
Folder

Advanced
Templates
URCaps

7 Gripper Open (1)
8 Find Visual Offset
52 Robot Program
53 Move
54 pos_start
55 pick_sequence
77 moves_in_out_machine
78 Move
79 waiting_for_unload
84 open_manual_door
92 open_auto_door
96 get_in_machine
102 picking_machined_part
107 vise_cleaning
116 place_part
133 get_out_machine
139 close_manual_door
146 close_auto_door
147 Set close_door=On
148 Wait door_closed=Hi
149 Set close_door=Off
150 handshake_back
152 drop_part_sequence

Folder
A folder is simply a collection of program lines.
Please enter text to be displayed in the program tree:
close_auto_door

23

Normal Speed 76% Simulation

23 The automatic closing also works similarly to the automatic opening. We send a signal to close the door through the right digital output. We then wait for the sensor to tell us that the door has been closed. Then we turn of the closing signal.



Move in and out of the machine

PROGRAM: cnc_elearning*
INSTALLATION: cnc

Basic | Command | Graphics | Variables

Folder

A folder is simply a collection of program lines.
Please enter text to be displayed in the program tree:

handshake_back

24

Normal | Speed 76% | Simulation

24 When everything is done, the robot sends the signal to the machine to start its machining process.

After that a wait command is added to make sure the machine starts its process before resetting the signal.



Move in and out of the machine

The screenshot shows the Robotiq software interface. The top toolbar includes icons for Run, Program, Installation, Move, I/O, and Log. The main window displays a program tree on the left and a command window on the right. The program tree shows a folder named 'drop_part_sequence' at line 154, which is highlighted with a callout box containing the number '25'. The command window shows the text 'drop_part_sequence' entered. The bottom status bar includes a 'Normal' indicator, a speed slider set to 76%, and a 'Simulation' button.

25 The **drop_part_sequence** folder works similarly to the one used to pick the part in the tray.

A **Pallet** node is used to make sure the emplacement of the part is always correct.

You can put the machine part in the same tray as the blank or a different one, but the program will remain similar, only the points will change.



Other Subprograms Called

In the program template, other programs are called in the program tree. These programs are not presented in this template and not provided. They are used to pick and place the part after the process. These can easily be program based on **Gripper Product Lessons**.

For more details about the types of insertions used (linear, spiral), please refer to the **Insertion Feature lesson** in the **Copilot Product Lessons**.



More Templates Available!



ROBOTIQ
eLearning

elearning.robotiq.com



LEAN
ROBOTICS



Share with your Peers!



D o F

Join the DoF Room to discuss about this learning Module, ask questions and get answers from the Pros!

[Join Here!](#)

Robotiq's community where industrial **Automation Pros** share their **know-how** and **get answers**.

[Ask your Questions !](#)