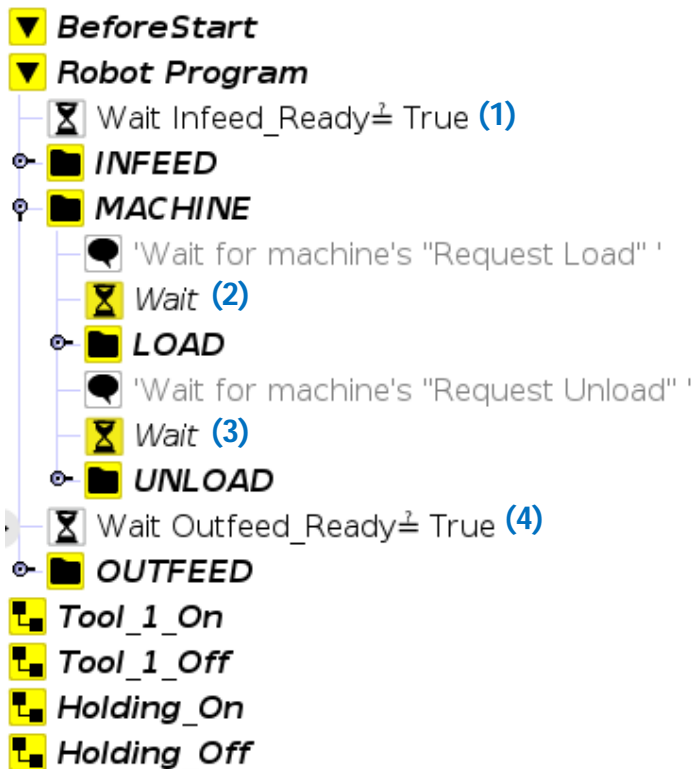


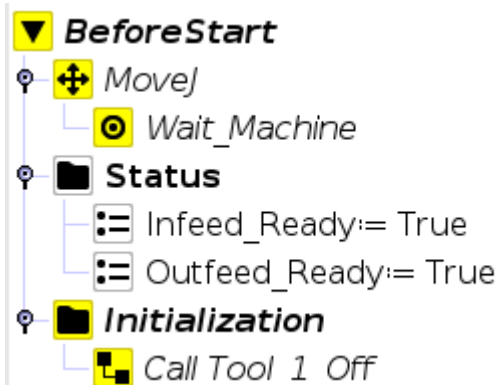
## Template



**ProTip!** Make your program more concise by creating SubPrograms to contain blocks of commands that need to be executed multiple times throughout the program. The sequences to open/close the Tool and make necessary payload adjustments are great examples of this.

## BeforeStart

This section executes once before entering the main program loop. Here you can move the robot to a starting position, reset the value of the status variables and initialize any necessary signals.



## Robot Program

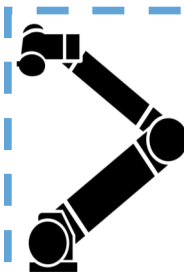
Use this program template to start building your machine tending application. The yellow nodes are incomplete, read below to understand their function then complete them.

The **Robot Program** loop consists of four main components: Infeed, Machine-Load, Machine-Unload and Outfeed. For this application, the robot will follow the commands in the same order every time.

When the program starts, the robot will wait until the **(1) Infeed\_Ready** indicates that there are available parts at the infeed location. This signal could be: initialized to True to do it immediately, assigned by an operator if a tray or pallet needs to be manually placed first, or triggered by a sensor that detects when there is a part in the infeed.

After picking a new part, the robot will wait outside the machine for the **(2) Load Request** signal before moving inside the machine to place the part. After exiting the machine, the robot will signal the machine and wait for the machine's **(3) Unload Request** to pick up the finished part.

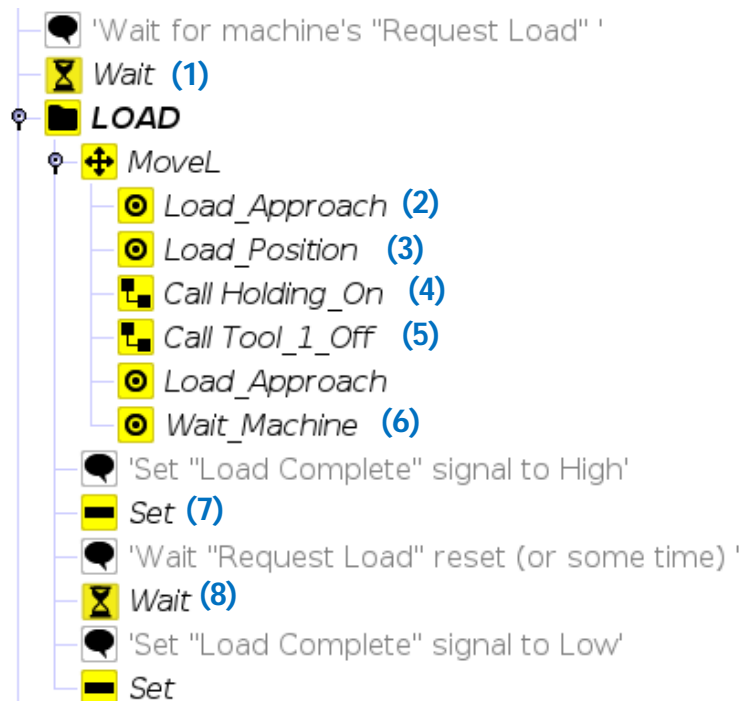
After picking the finished part, the robot will wait for **(4) Outfeed\_Ready** before placing the finished part. This signal could be: initialized to True to do it immediately, assigned by an operator if a tray or pallet needs to be manually placed first, or triggered by a sensor that detects when the placing position in the outfeeder is empty.



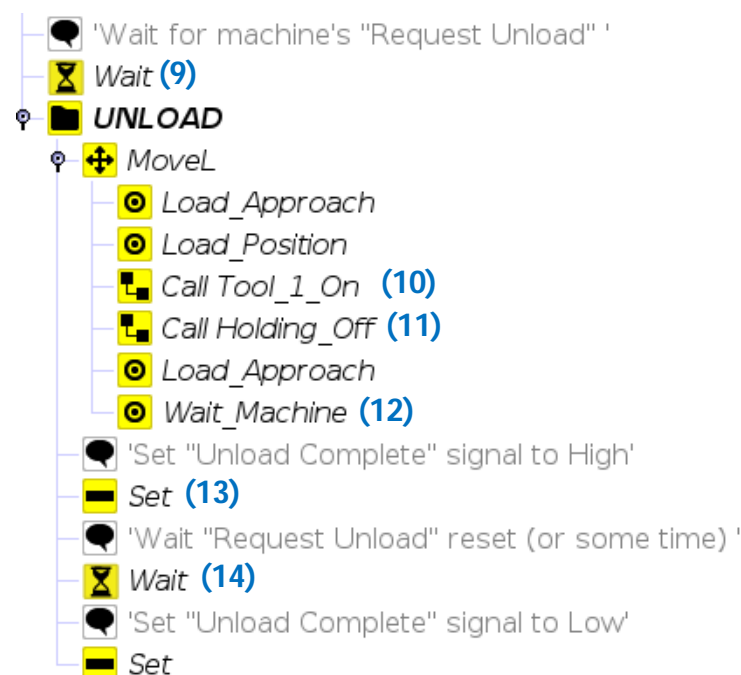
### UR Academy

If you're not familiar with UR programming head over to UR Academy to learn the basics:  
[www.universal-robots.com/academy](http://www.universal-robots.com/academy)

## Template



**ProTip!** If you're having trouble aligning the robot tool with the machine holding try this: With the workpiece clamped in the machine, put the robot into freedrive mode and then grip the other end of the workpiece with the robot tool. This will align the robot tool with the machine holding for the part handover before you save the **Load\_Position**



## LOAD

This **Folder** contains the sequence the robot will follow to load the machine. Before executing, the robot will **(1) Wait** for the load request from the Machine.

**(2) Load\_Approach** ensures that the robot reaches the **(3) Load\_Position** to place a part into the machine without colliding with it. They are Fixed Waypoints that need to be taught to move the part with Gripper 1.

**(4) Call SubProgram Holding\_On** to close the chuck, vice or other fixture which holds the part, then **(5) Call SubProgram Tool\_1\_Off** to open Gripper 1 and leave the part in the machine: this will ensure that the part doesn't move during the transfer from the gripper to the machine.

Move through **Load\_Approach** to **(6) Wait\_Machine**, another Fixed Waypoint that needs to be taught in a position with the robot outside the machine.

**(7) Set** a signal to indicate the machine is loaded and **(8) Wait** for the machine request to reset before clearing this indication.

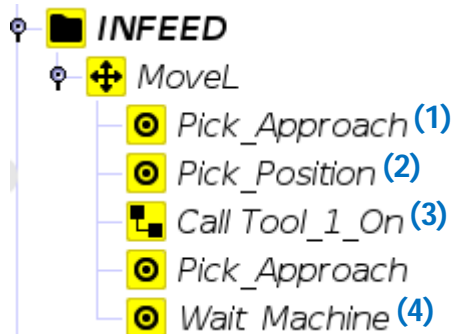
## UNLOAD

This **Folder** contains the sequence that the Robot will run to unload the machine. Before executing, the robot will **(9) Wait** for the unload request from the machine.

The sequence will be similar to **LOAD**, but this time the gripper is closed using **(10) Tool\_1\_On** before the part is release by **(11) Holding\_Off** SubProgram. Again this is to ensure that the part doesn't move during the transfer from the machine to the robot tool.

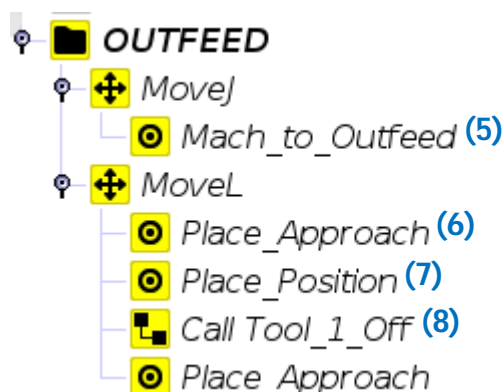
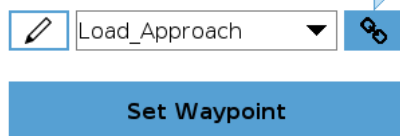
The robot will move out of the machine through the previously defined **Load\_Approach** to the **(12) Wait\_Machine** position before the signal is **(13) Set** to indicate that the Machine is loaded. Then, it will **(14) Wait** for the machine's load request to reset, before clearing this indication.

## Template



**ProTip!** Use the Link Waypoints functionality to make the **Pick\_Approach** positions before and after the **Pick\_Position** identical. This makes the program becomes easier to maintain as editing either of the **Pick\_Approach** Waypoints will update **all** of them.

### Waypoint



## INFEED

This **Folder** contains the sequence that the Robot will follow to pick new parts. The fixed infeed presents a single item to the robot at a fixed location.

(1) **Pick\_Approach** is a Fixed Waypoint that should be taught in a position close to the pick location from which the robot can approach the infeeder without colliding with it. Then the robot moves to the (2) **Pick\_Position**, which is another Fixed Waypoint that must be taught at the position where the gripper picks up the part.

(3) **Call SubProgram Tool\_1\_On** to close the gripper and pick up the new part.

Optionally move to (4) **Wait\_Machine** to be closer to the machine and reduce the time to exchange or load the part in the machine when it is ready.

## OUTFEED

This **Folder** contains the sequence that the robot will run to place finished parts into a slide, conveyor or any other outfeeder that receives the parts at the same location every time.

The robot moves through the (5) **Mach\_to\_Outfeed** position out of the machine to the (6) **Place\_Approach** position, which is taught so that the robot can reach the (7) **Place\_Position** without colliding with the outfeeder.

Then (8) **Call** the SubProgram **Tool\_1\_Off** to open the gripper and place the finished part.