

# MachineMotion Programming Options

## Contents

[Overview](#)

[MachineApps](#)

[MachineLogic](#)

[Code-Free Programming](#)

[Python Programming](#)

[MachineMotion for Universal](#)

[Robots \(URCap\)](#)

[Low Level Socket Commands](#)

## Overview

There are a number of different ways to program a MachineMotion controller, ranging from application-focused MachineApps, MachineLogic's Code-Free or Python Programming editors, to Low Level Socket commands using TCP/IP communication. This document is meant to give guidance on which programming option is right for a given application.

There are 6 programming options, generally split into 3 categories:

### Code Free

- MachineApps
- MachineLogic Code-Free Programming

### Python Based

- MachineLogic Python Programming

### Other

- URCap
- Low Level Socket API

## MachineApps

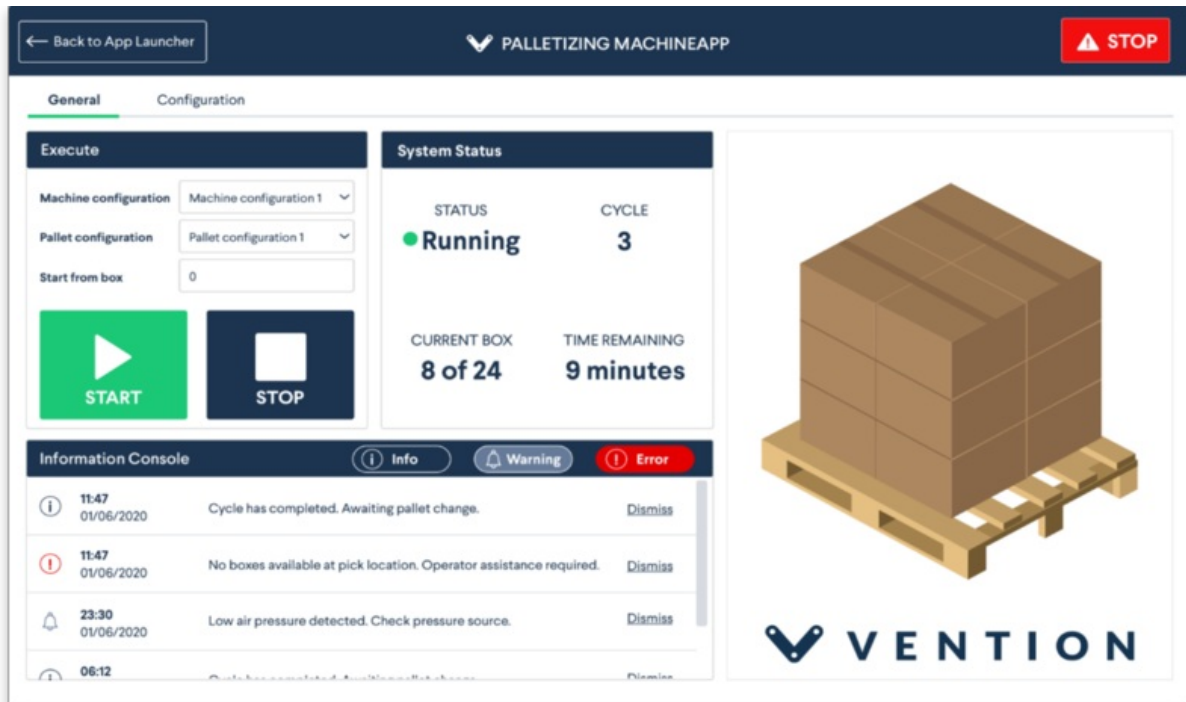
Machine Apps are Vention built applications for MachineMotion. They are code-free for the end user, only requiring software configuration to match the installed hardware.

Below are descriptions of the currently available MachineApps. All MachineApps are purchased separately from Vention and installed directly on the MachineMotion controller.

### Palletizer

The Palletizer MachineApp allows for the easy control of a cartesian palletizer. The intuitive interface allows users to create different layer patterns, skid configurations as well as pick multiple products from multiple intake points.

For more information about the Palletizer MachineApp check the [Palletizer User Guide](#).

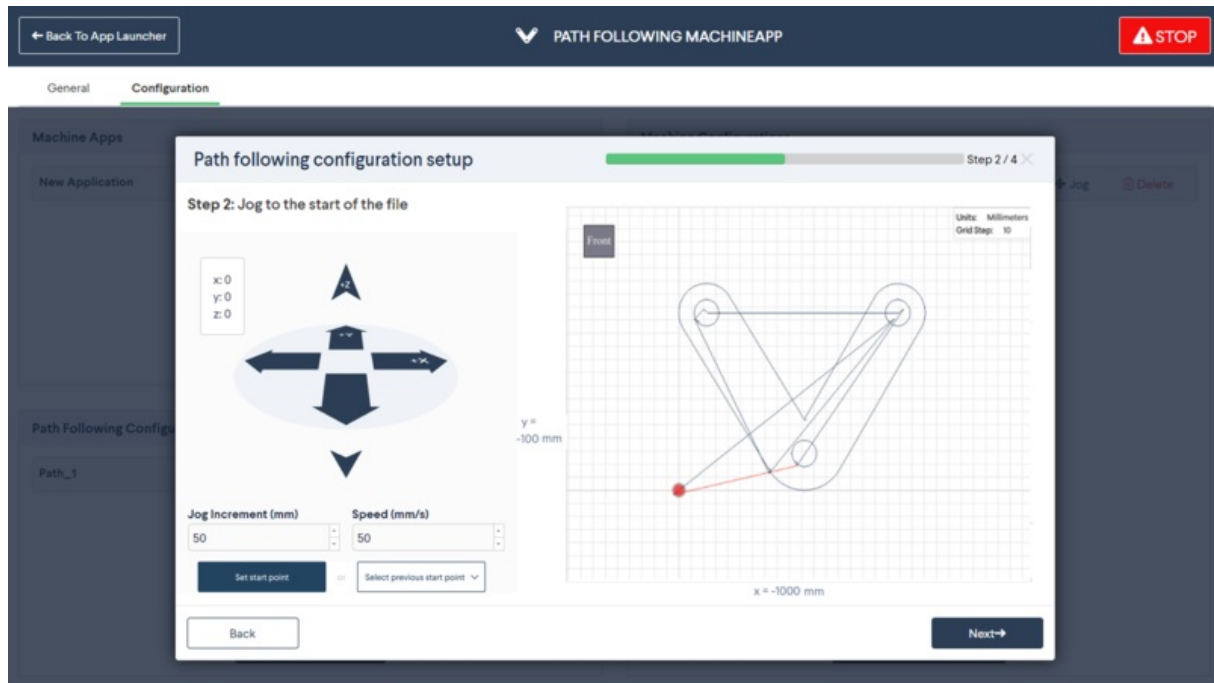


*MachineApp - Palletizer*

### Path Following

The Path Following MachineApp allows users to upload a DWG/DXF file that will be traced out by a multi-axis machine. It can be used for a number of applications, such as laser cutting, glue dispensing and CNC style machining.

For more information about the Path Following MachineApp check the [Path Following User Guide](#).



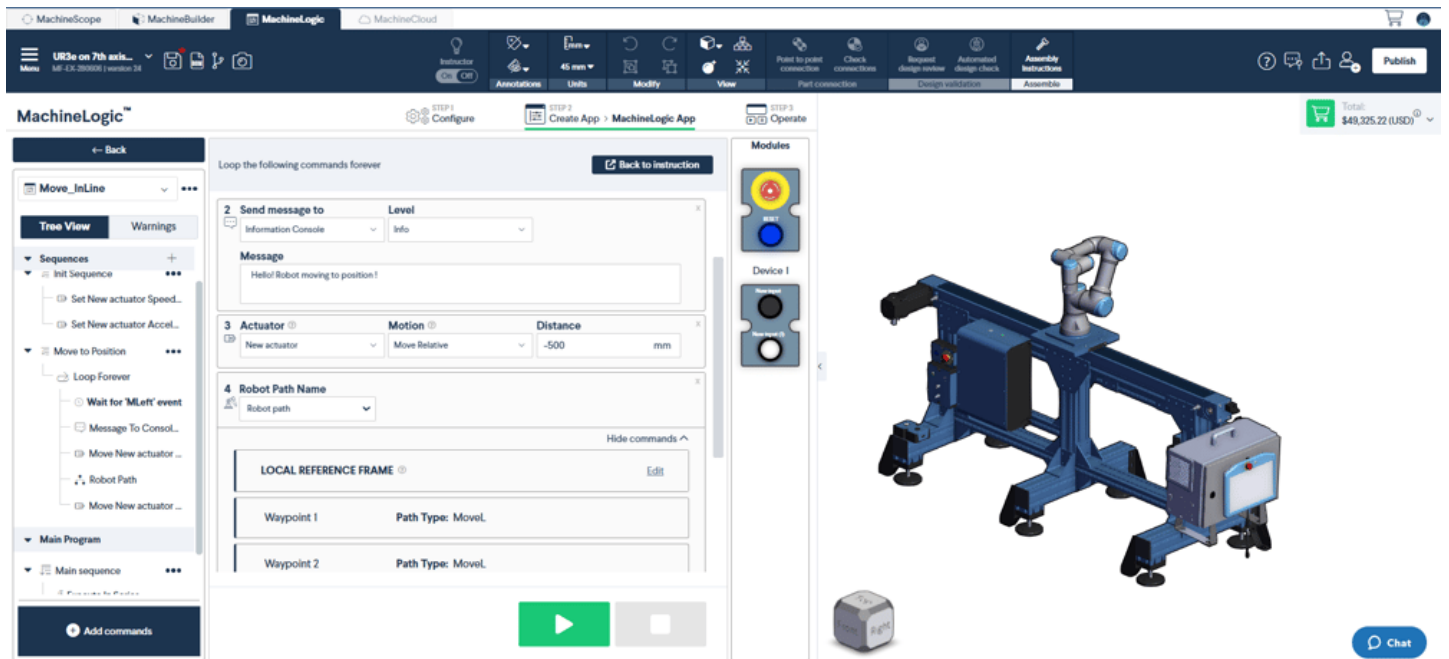
*MachineApp - Path Following*

## MachineLogic

## Code-Free Programming

MachineLogic Code-Free Programming is one of the standard tools built into the MachineMotion controller. It allows for easy configuration and control of the MachineMotion components and provides users with the means to program their machines using simple instructions cards in a drag and drop interface. Program flow control elements, such as conditional statements and loops, are available for more complex tasks. State machine programming is also possible.

For more information about MachineLogic Code-Free Programming, check the [MachineLogic Guide](#).

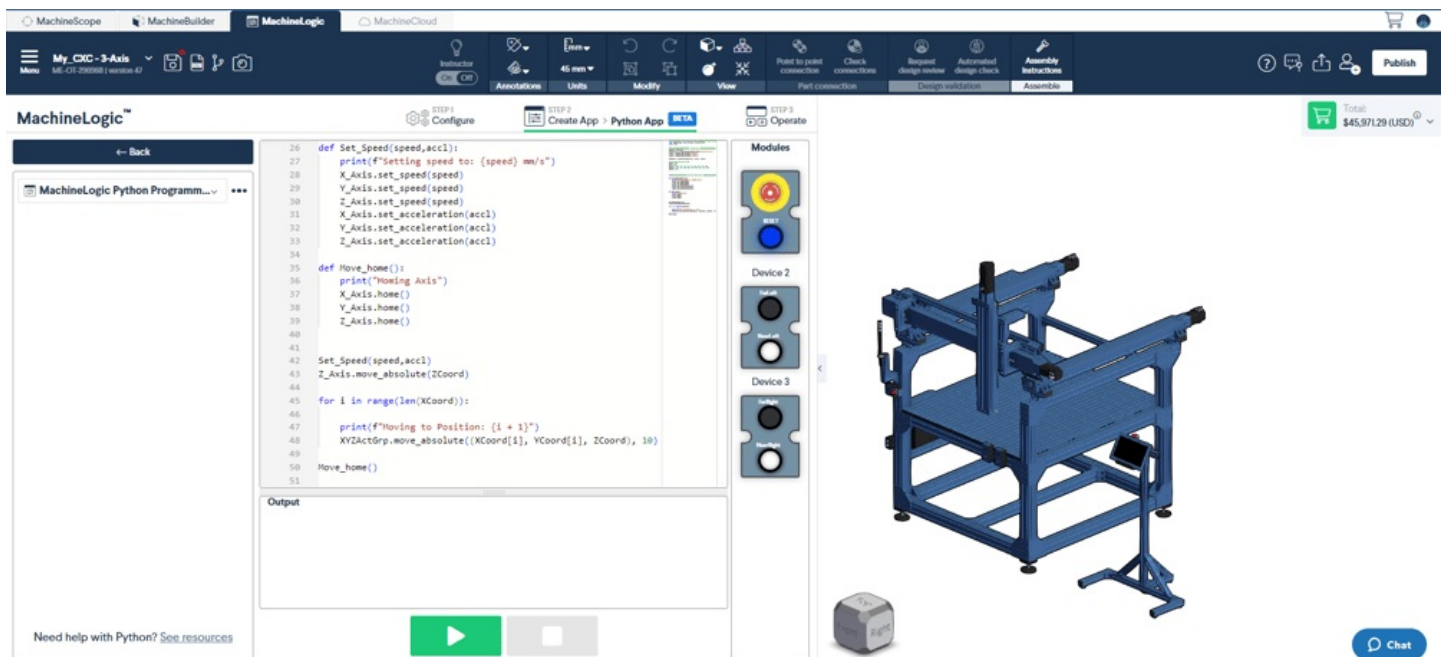


MachineLogic

## Python Programming

MachineLogic Python Programming is the second default option built into the MachineMotion controller. Just like Code-Free programming, Python Programming in MachineLogic allows for easy configuration and control of the MachineMotion components. It provides users with a Python IDE directly accessible from MachineLogic, in which users can program their machines using simple classes and methods encapsulating all functionalities required for automation programming.

See the MachineLogic Python Programming documentation [here](#)



MachineLogic

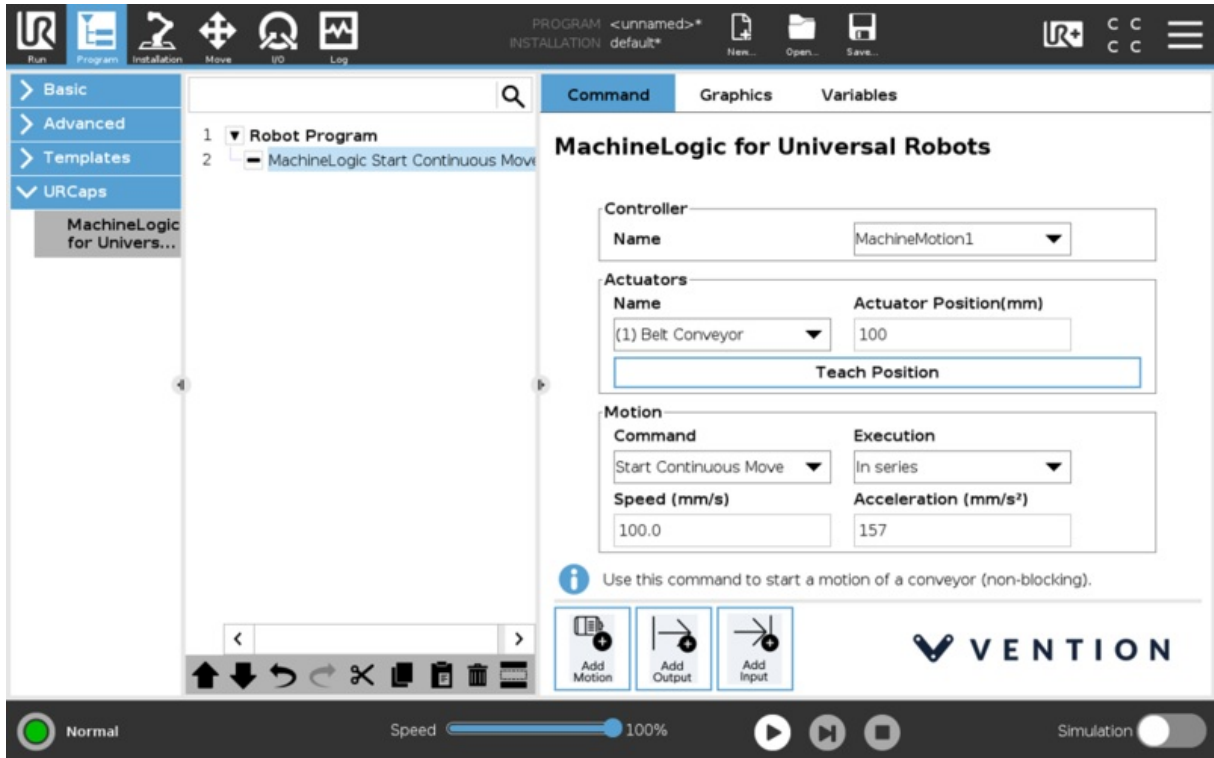
MachineLogic can be either accessed from Vention.io, in the associated MachineBuilder design session, or directly from the MachineMotion Controller itself using either the Vention pendant or by connecting a laptop in the designated ethernet port. For more information on getting started using the MachineMotion controller, [click here](#).

## MachineMotion for Universal Robots (URCap)

MachineMotion for Universal Robots URCap makes it easy to program applications where a Universal Robot will be interacting with a MachineMotion. The URCap is purchased from Vention and installed on the UR controller.

Once installed, the URCap provides a easy interface to control the MachineMotion from inside the UR programming interface. This means the entire robot station program will live in one place. This is especially useful for UR range extenders.

For more information about the MachineMotion for Universal Robots URCap check the [URCap User Guide](#)



*MachineMotion for Universal Robots (URCap)*

## Low Level Socket Commands

Low Level Socket commands allow an ethernet connected device to send commands to the MachineMotion using TCP/IP communication. This means the MachineMotion can be controlled by a PLC, robot controller or anything else that can send and receive TCP/IP messages.

These commands are only useful when the MachineMotion will not be main controller in the system, and the main controller cannot use the Python API.

For more information about Low Level Socket Programming check the [Low Level Socket API](#).