MachineMotion Programming Options

Contents

Overview

There are a number of different ways to program a MachineMotion controller, ranging from code-free MachineApps and MachineLogic, to Low Level Socket commands using TCP/IP communication. This document is meant to give guidance on which programing option is right for a given application.
There are 6 programming options, generally split into 3 categories:

Code Free

- MachineApps
- MachineLogic

Python Based

- Python
- MachineApp Template

Other

- URCap
- Low Level Socket

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.
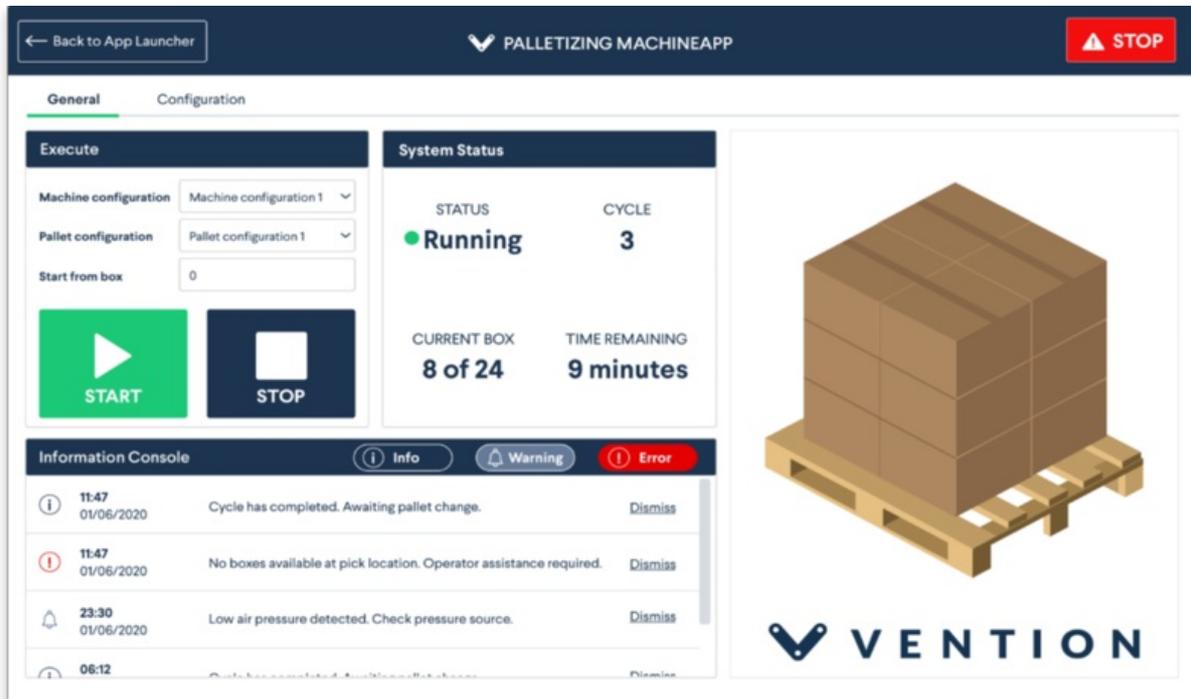
MachineApp - Palletizer

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.
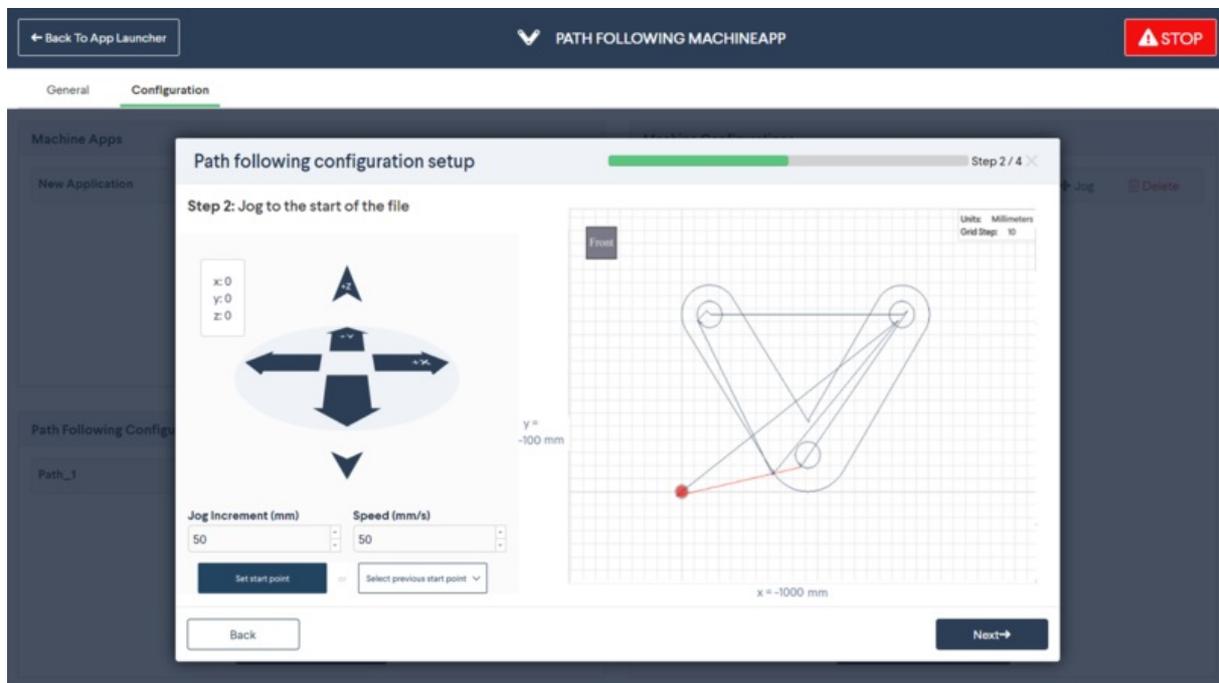

MachineApp - Path Following

MachineLogic

MachineLogic is the code-free programming tool built into MachineMotion. It allows for easy configuration and control of the MachineMotion components. Program flow control elements, such as if statements and loops, are available for more complex tasks.

All that is required to start programming in MachineLogic is a Vention Pendant or a computer with Google Chrome installed.

MachineLogic is ideal for stand alone, single MachineMotion systems with relatively simple procedural operations.

For more information about the MachineLogic check the MachineLogic Guide.
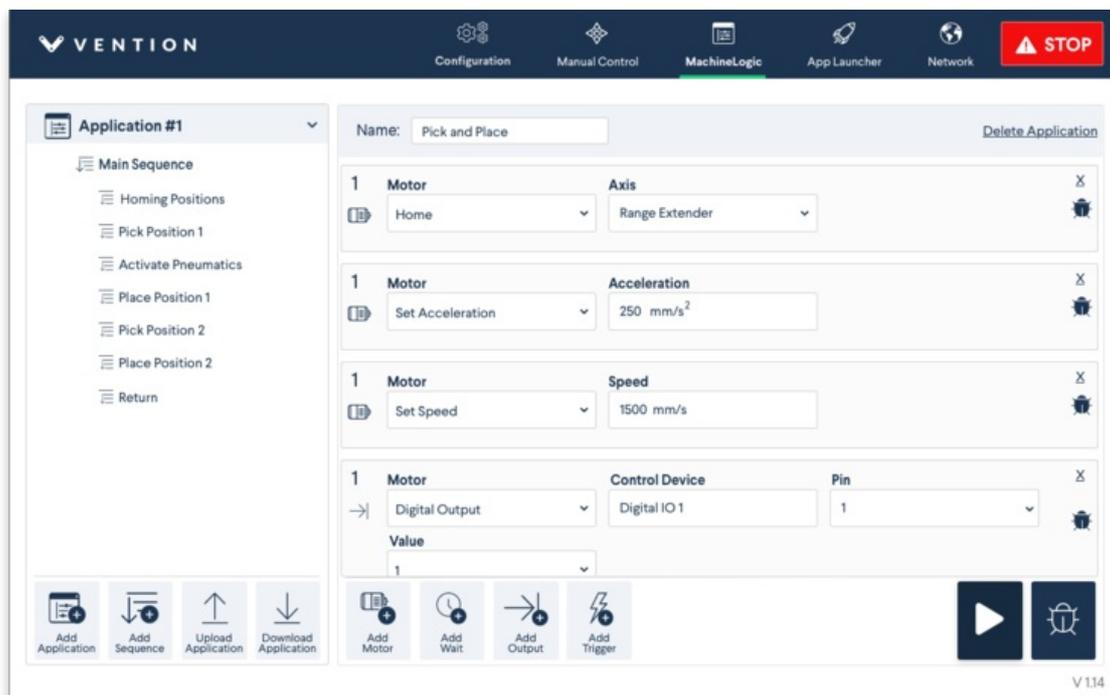

MachineLogic

Python

Python Programming is the most flexible option, but requires some programming knowledge, so it is more difficult than MachineLogic. It is the preferred choice for complex custom applications.

There are a few use cases that require Python instead of MachineLogic, this includes:

- Communicating with custom hardware
- Complex logic including encoders and brake control
- Integrating third party software and tools
- Controlling multiple MachineMotions in unison
- Complex calculations

For more information about Python Programming check the Python Programming Guide.
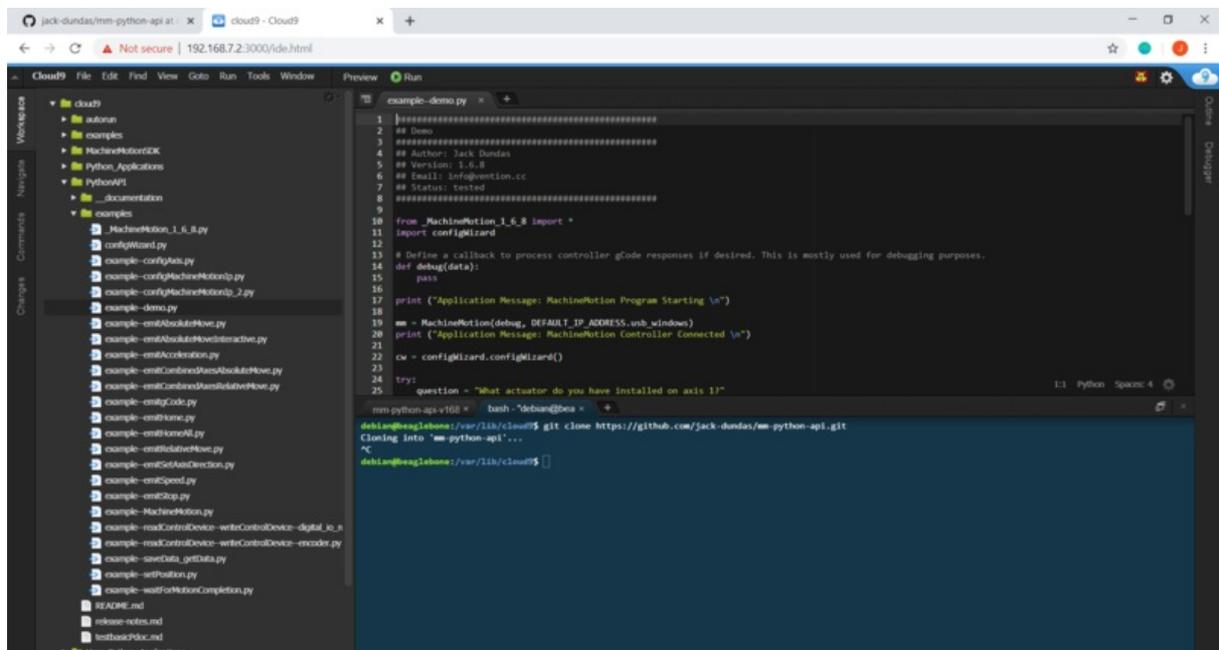
Cloud9 Python Interface

MachineApp Template

The MachineApp template is installed on every MachineMotion. It is a template that allows users to program their machine in Python, using a built in state machine. It also comes with UI template that can be built using JavaScript.

If a custom UI is needed, and the machine's process can be written as a state machine, the MachineApp template is the best choice. It is generally used for more complex machines, with multiple MachineMotions and IO.
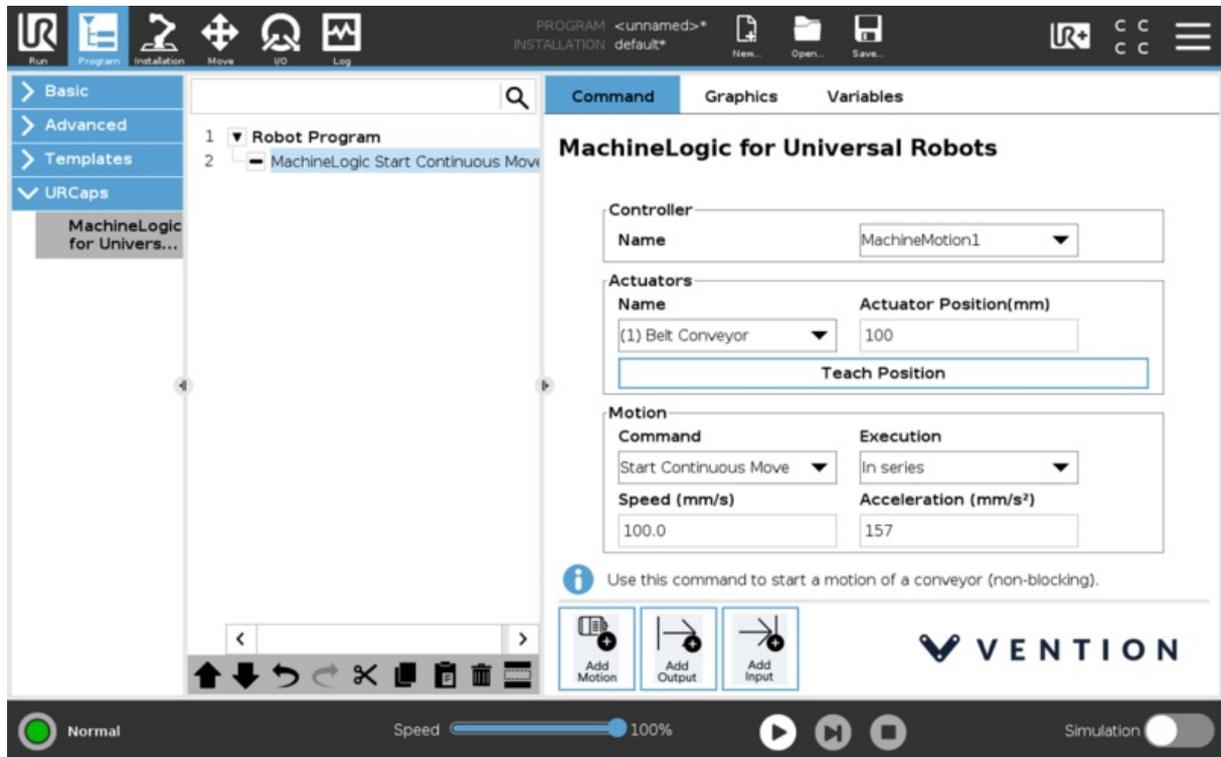
For more information about the MachineApp Template check the MachineApp Template Guide.

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.