

# Application Programming Interface: Low-Level Sockets for MachineMotion SW V2.3.1 and V2.3.0

## Contents

[Application Programming Interface: Low-Level Sockets for MachineMotion SW V2.3.0, 2.3.1](#)

[Overview](#)

[Configuring An IP Address](#)

[Opening a Connection](#)

[Commands Reference](#)



# Application Programming Interface: Low-Level Sockets for MachineMotion SW V2.3.0 , 2.3.1

## Overview

This document details the commands that can be sent to your MachineMotion V2 using Ethernet TCP/IP protocol. It is supported on MachineMotion software version v2.3.0 This API uses sockets to exchange data. All information is packaged in string format and sent over the socket communication channel to port 9999 of the ethernet IP Address.

## Configuring An IP Address

There are three physical ethernet connections available on the MachineMotion V2. The default ethernet port (labeled "To PC") will always have an IP address of 192.168.7.2. The second and third ethernet ports (labeled "LAN 1" and "LAN 2") may have a user configurable IP address. The configurable IP address can be set in one of three ways:

- Using a laptop and chrome web browser, navigate to 192.168.7.2, select the network configuration tab and enter the desired IP address.
- Using a laptop and the low level socket API, send the "ethernet configuration commands" to set the desired static IP commands.

- Use a pendant to connect to the Machine Motion control center, select the network configuration tab and enter the desired IP address.

## Opening a Connection

A connection must be opened before any commands can be sent or received from the MachineMotion V2 ethernet port. How to open a connection will depend on the programming language being used. Once the connection is open, the MachineMotion V2 will respond with “MachineMotion connection established”. It is recommended the first command sent is isReady to confirm a properly established connection.

User Agent	Direction	MachineMotion
Initialize TCP Connection	→	Accept
Receive	←	Send: MachineMotion connection established
Send: isReady;	→	Receive
Receive	←	Send: MachineMotion isReady=true;

## Commands Reference

### Continuous Move

#### Description

Moves a conveyor or rotary table axis continuously at a given speed and acceleration. This command is only available with MachineMotion software v1.16 and newer.

#### Format

SET im\_conv\_<axis> S<speed> A<acceleration>; Where <axis> is the axis (1, 2, 3 or 4) to move as a (String). Where <speed> is the speed to move the axis in mm/sec as a (String). Where <acceleration> is the acceleration to move the axis in mm/sec<sup>2</sup> as a (String).

#### Usage

SET im\_conv\_3 S100 A400; Starts moving axis 3 at 100 mm/sec with an acceleration of 400 mm/sec<sup>2</sup>. SET im\_conv\_3 S0 A400; Stops axis 3 using 400 mm/sec<sup>2</sup> deceleration.

#### Response on Receive

Ack

#### Response On Complete

None

### Define IP Address

#### Description

Define the IP address.

#### Format

de\_ethernet\_port\_ip\_config/<address>; Where <address> is the IPv4 address (###.###.###.###) as a string (String).

## Usage

de\_ethernet\_port\_ip\_config/192.168.7.3; Sets the IP address to 192.168.7.3.

## Response on Receive

Ack

## Response On Complete

None

## Delayed Absolute Move

---

### Description

Moves an axis to the specified absolute position after the de\_move\_abs\_exec command is sent.

### Format

SET de\_move\_abs\_<axis>/<value>; Where <axis> is the axis to move (1, 2 or 3) as a (String). Where <value> (String) is a position in mm.

## Usage

SET de\_move\_abs\_1/50/ Will move axis 1 to absolute position 50mm after the de\_move\_abs\_exec command is sent. SET de\_move\_abs\_2/150/ Will move axis 2 to absolute position 150mm after the de\_move\_abs\_exec command is sent. SET de\_move\_abs\_3/175/ Will move axis 3 to absolute position 175mm after the de\_move\_abs\_exec command is sent.

## Response on Receive

Ack

## Response On Complete

None

## Delayed Absolute Move Execute

---

### Description

Executes the latest de\_move\_abs\_1, de\_move\_abs\_2 and de\_move\_abs\_3 commands simultaneously.

### Format

de\_move\_abs\_exec;

## Usage

de\_move\_abs\_exec;

## Response on Receive

Ack

## Response On Complete

None

## Delayed Relative Move

---

### Description

Move an axis relative to its current position by the distance specified in the distance parameter after the command `de_move_abs_exec` is sent.

### Format

`SET de_move_rel_<axis>/<value>;` Where `<axis>` is the axis to move (1, 2 or 3) as a (String). Where `<value>` (String<!--<mark-->) is a position change in mm.

### Usage

`SET de_move_rel_1/-50/;` Will move axis 1, 50 mm in the negative direction after the `de_move_rel_exec` command is sent. `SET de_move_rel_2/100/;` Will move axis 2, 100 mm in the positive direction after the `de_move_rel_exec` command is sent. `SET de_move_rel_3/-75/;` Will move axis 3, 75 mm in the negative direction after the `de_move_rel_exec` command is sent.

### Response on Receive

Ack

### Response On Complete

None

## Delayed Relative Move Execute

---

### Description

Executes the latest `de_move_rel_1`, `de_move_rel_2` and `de_move_rel_3` commands simultaneously.

### Format

`de_move_rel_exec;`

### Usage

`de_move_rel_exec;`

### Response on Receive

Ack

### Response On Complete

None

## Get Position

---

## Description

Queries the current position of an axis.

## Format

GET im\_get\_controller\_pos\_axis\_<axis>; Where <axis> is the axis to get position from (1, 2 or 3) as a ( String)

## Usage

GET im\_get\_controller\_pos\_axis\_1 Returns position of axis 1. GET im\_get\_controller\_pos\_axis\_2 Returns position of axis 2. GET im\_get\_controller\_pos\_axis\_3 Returns position of axis 3.

## Response on Receive

None

## Response On Complete

(<position>) Where <position> is the position in mm units.

## Immediate Absolute Move

---

### Description

Moves an axis to the specified absolute position.

### Format

SET im\_move\_abs\_<axis>/<value>; Where <axis> is the axis to move (1, 2 or 3) as a ( String). Where <value> (String) is a position in mm.

### Usage

SET im\_move\_abs\_1/50/; Immediately Moves axis 1 to absolute position 50mm SET im\_move\_abs\_2/100/; Immediately Moves axis 2 to absolute position 100mm SET im\_move\_abs\_3/200/; Immediately Moves axis 3 to absolute position 200mm

### Response on Receive

Ack

### Response On Complete

None

## Immediate Home All

---

### Description

Returns all axes to their home locations sequentially: first axis 1, then axis 2, then axis 3.

### Format

im\_home\_axis\_all;

### Usage

im\_home\_axis\_all;

### Response on Receive

None

### Response On Complete

MachineMotion im\_home\_axis\_all = completed

## Immediate Home Axis

---

### Description

Returns an axis to its home location.

### Format

im\_home\_axis\_<axis>; Where <axis> is the axis to home (1, 2 or 3) as a ( String)

### Usage

im\_home\_axis\_1; Immediately Homes axis 1 im\_home\_axis\_2; Immediately Homes axis 2 im\_home\_axis\_3; Immediately Homes axis 3.

### Response on Receive

None

### Response On Complete

MachineMotion im\_home\_axis\_<axis> = completed Where <axis> is the axis requested to home.

## Immediate Relative Move

---

### Description

Immediately moves an axis by the distance specified relative to the current location of the axis. The distance parameter can be negative.

### Format

SET im\_move\_rel\_<axis>/<value>; Where <axis> is the axis to move (1, 2 or 3) as a ( String) Where <value> is a position value in mm as a ( String)

### Usage

SET im\_move\_rel\_1/50/; Immediately Moves axis 1, 50mm in the positive direction from its current position. SET im\_move\_rel\_2/-100/; Immediately Moves axis 2, 100mm in the negative direction from its current position. SET im\_move\_rel\_3/200/; Immediately Moves axis 3, 200mm in the positive direction from its current position.

### Response on Receive

Ack

### Response On Complete

None

## Immediate Stop

---

### Description

Immediately stops motion on all axes.

### Format

im\_stop;

### Usage

im\_stop;

### Response on Receive

Ack

### Response On Complete

MachineMotion im\_stop = complete

## is Motion Completed

---

### Description

Queries MachineMotion to determine if motion is currently being executed. This function is useful for waiting for the completion of a movement.

### Format

isMotionCompleted;

### Usage

isMotionCompleted;

### Response on Receive

None

### Response On Complete

MachineMotion isMotionCompleted = true or MachineMotion isMotionCompleted = false

## is Ready

---

### Description

Queries MachineMotion to know if it is ready to receive commands. This function indicates if the MachineMotion system is initialized and ready for operation after a connection event.

### Format

isReady;

## Usage

isReady;

## Response on Receive

None

## Response On Complete

MachineMotion is Ready = true

## Move to Closest Angle

---

### Description

Moves a rotary table axis to a given angle without making a full rotation. This makes it easier to move the axis after multiple full rotations. The axis will take the shortest path to the new position. This command is only available with MachineMotion software v1.16 and newer.

### Format

moveToClosestAngle/<axis>/<position>/request; Where <axis> is the axis (1, 2 or 3) to move as a (String). Where <position> is the angle command in degrees as a (String).

### Usage

moveToClosestAngle/2/45/request; Moves axis 2 to the 45 degree position. For example, if the axis was previously at 365 degrees, it would move to 405 degrees after this command. This differs from the absolute move command, which would move the axis from 365 degrees to 45 degrees.

### Response on Receive

Ack moveToClosestAngle/<axis>/<position>/request Where <axis> is the requested axis to move. Where <position> is the requested angle (in degrees).

### Response On Complete

None

## Read Digital Input

---

### Description

Reads a digital input from a Digital IO Module. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

### Format

io-expander/<module>/digital-input/<pin> Where <module> is the address (1, 2 or 3) of the Digital IO Module (String). Where <pin> is the terminal on the Digital IO Module (0, 1, 2 or 3) as a (String).

### Usage

io-expander/2/digital-input/0; Returns the current state of pin 0 on the Digital IO Module with address 2.

### Response on Receive



None

## Response On Complete

(<value>) Where <value> will be 0 or 1

## Read E-Stop Status

---

### Description

Reads the current status of the e-stop. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

### Format

estop/status;

### Usage

estop/status;

### Response on Receive

None

### Response On Complete

true When e-stop is engaged. false When e-stop is not engaged.

## Read Push Button

---

### Description

Read Push-Button for module [1-8]. Button 0 is the black button, button 1 is white

### Format

push-button/<module>/digital-input/<button>;

### Usage

push-button/2/digital-input/0;

### Response on Receive

None

### Response On Complete

push-button/<module>/digital-input/<button> <pushed|released>

## Release E-Stop

---

## Description

Releases software generated e-stop. It must be followed by a `estop/systemreset/request` command. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

## Format

`estop/release/request;`

## Usage

`estop/release/request;`

## Response on Receive

Ack `estop/release/request`

## Response On Complete

None

## Set All Axes Acceleration

---

### Description

Configures the travel acceleration of the system in mm/min<sup>2</sup>. For single-axis movements, the travel acceleration of the axis will be equal to this setting. For multi-axis synchronous moves, the overall motion acceleration will be equal to this setting.

### Format

`SET acceleration/<value>;` Where `<value>` is an acceleration value in mm/min<sup>2</sup> as a (String)

### Usage

`SET acceleration/100;` Sets acceleration for all axes to 100mm /min<sup>2</sup>

### Response on Receive

Ack

### Response On Complete

None

## Set All Axes Speed

---

### Description

Configures the travel speed of the system in mm/minute. For single-axis movements, the travel speed of the axis will be equal to this setting. For multi-axis synchronous moves, the overall motion speed will be equal to this setting.

### Format

`SET speed/<value>;` Where `<value>` is a position value in mm/minute as a (String)

## Usage

SET speed/300/; Sets speed for all axes to 300 mm/minute

## Response on Receive

Ack

## Response On Complete

None

## Set Position

---

### Description

Sets the current position of an axis to a new value (mm).

### Format

SET im\_set\_controller\_pos\_axis\_<axis>/<value>; Where <axis> is the axis to set position (1, 2 or 3) as a (String). Where <value> (String) is a position in mm.

### Usage

SET im\_set\_controller\_pos\_axis\_1/0/; Sets the current location of axis 1 as position 0 mm SET im\_set\_controller\_pos\_axis\_2/0/; Sets the current location of axis 2 as position 0 mm SET im\_set\_controller\_pos\_axis\_3/0/; Sets the current location of axis 3 as position 0 mm

### Response on Receive

Ack

### Response On Complete

None

## Set Power Switch

---

### Description

Write a state (on or off) to a power switch module

### Format

power-switch/<module>/digital-output/<"on"|"off">;

### Usage

power-switch/3/digital-output/"off"

### Response on Receive

None

### Response On Complete

## System Reset

---

### Description

Resets the system, this must be done after all e-stop event. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

### Format

estop/systemreset/request;

### Usage

estop/systemreset/request;

### Response on Receive

Ack estop/systemreset/request

### Response On Complete

None

## Trigger E-Stop

---

### Description

Sets the e-stop. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

### Format

estop/trigger/request;

### Usage

estop/trigger/request;

### Response on Receive

Ack estop/trigger/request

### Response On Complete

None

## Wait On Push Button

---

### Description

Wait for a button [0-1] on module [1-8] to achieve a certain state [pushed or released]. Button 0 is the black button, button 1 is white

## Format

```
push-button/<module>/wait-on/<button>/<pushed|released>;
```

## Usage

```
push-button/2/wait-on/0/pushed;
```

## Response on Receive

None

## Response On Complete

```
push-button/<module>/digital-input/<button> <pushed|released>
```

## Write Digital Output

---

### Description

Writes a digital output to a Digital IO Module. This command is only available with MachineMotion software v1.14\_hotfix\_200325 and newer.

### Format

io-expander/<module>/digital-output/<pin> <value>; Where <module> is the address (1, 2 or 3) of the Digital IO Module ( String). Where <pin> is the terminal on the Digital IO Module (0, 1, 2 or 3) as a (String). Where <value> is the desired state of the output (0 = 0V, 1 = 24V) as a ( String).

### Usage

```
io-expander/3/digital-output/2 1; Sets state of pin 2 to the value of 1 on the Digital IO Module with address 3.
```

### Response on Receive

None

### Response On Complete

None