



# PoKeys plugin for Mach4

---

*PoKeys device setup*

Version: 5/2/2017

## SAFETY INFORMATION



This product is intended for integration by the user into a computer numerical control (CNC) machine. It is the user's responsibility to assess the overall system design and address all safety considerations that affect the users and equipment. The user assumes all responsibility for system design, including compliance with regulatory standards and codes issued by the applicable entities. PoLabs do not make any claims as to the suitability of this equipment for the user's application. Serious personal injury or equipment damage can occur from the improper integration, installation or operation of this product.

This product is not guaranteed to be fail-safe. The system that this equipment is used with shall be fitted with a separate means of fail-safe protection, emergency-stop capability and/or system power removal. This equipment may be connected to dangerous power sources, including electrical power sources. Dangerous voltage levels may be present at this equipment or at connected devices. Measures must be taken to prevent persons from contacting voltage sources which may be present. Equipment should be housed inside an enclosure suitable for the intended environment. Safety interlocks should be provided to prevent any and all dangers to personnel.

CNC machine tools are inherently dangerous, and can cause injury to operators and maintenance personnel. Operators and maintenance personnel shall be properly trained in the safe use, operation and maintenance of such machines. Automated machines that this equipment may be used with can move at any time. All persons exposed to such machines must understand the dangers that are present.

## PoKeys plugin for Mach4 - PoKeys device setup

### Installing PoKeys plugin for Mach4

PoKeys plugin for Mach4 requires the following resources to be installed:

- Mach4 - version 4.2.0.3017 or newer
- PoKeys USB driver (if USB connection with PoKeys device is used): to install it, use PoKeys setup package and select 'USB driver' in the installation options

PoKeys plugin for Mach4 is supported on Windows XP, Windows Vista, Windows 7/8/8.1/10 (32 and 64-bit).



Use PoKeys configuration software to update the firmware of the PoKeys device and configure the device-level settings (e.g. IP address).  
PoKeys configuration software should not be used to configure CNC-related settings if Mach4 is going to be used.

The plugin is installed by copying files Mach4PoKeysPlugin.m4pw and Mach4PoKeysPlugin.sig into Mach4's plugins folder. On first startup, enable the PoKeys plugin in Mach4 - open menu Configure > Mach, then check that the plugin is enabled.

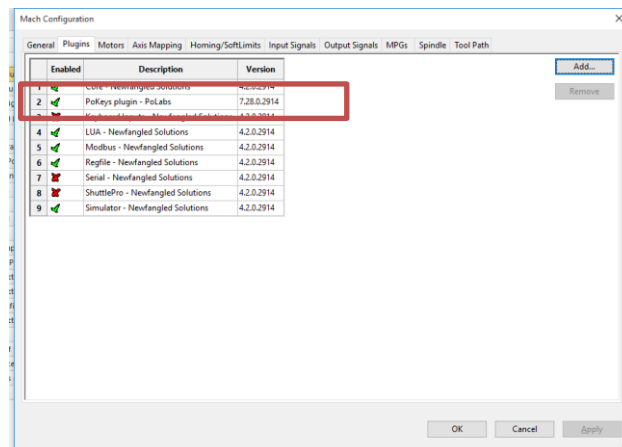
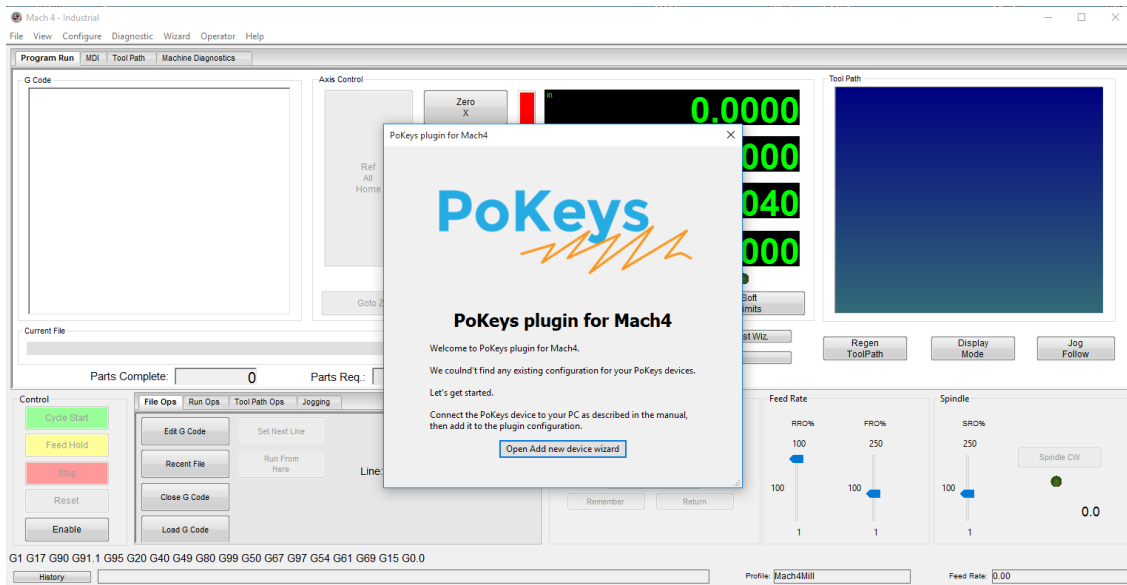


Figure 1: Enabling PoKeys plugin for Mach4

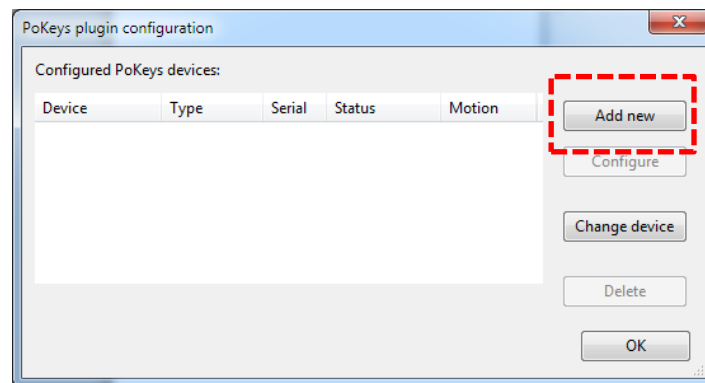
On the first setup, the plugin welcome screen will open and instruct you to configure new PoKeys device. Follow the instructions on the screen.

## PoKeys plugin for Mach4 - PoKeys device setup

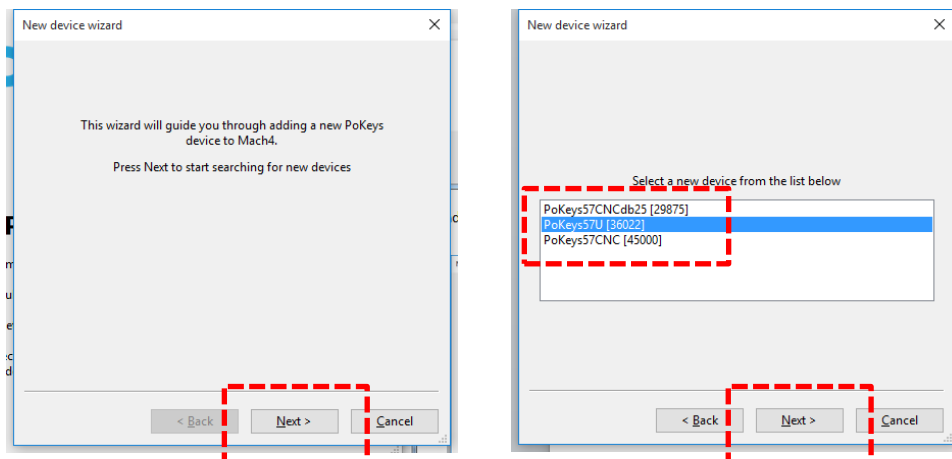


### Adding and configuring a new PoKeys device with Mach4

1. Start Mach4. If there is no configured PoKeys device, the welcome dialog will appear - click on 'Open Add new device wizard' and skip to step 4.
2. Open Configure > Plugins > PoKeys plugin - PoLabs
3. Main PoKeys plugin configuration screen will appear, where you can manage your PoKeys devices in Mach4



4. In order to add new device, click 'Add new' and follow the wizard



5. In the next step of the wizard, give a unique name can be assigned to your PoKeys device in Mach4 environment. By default, the name is constructed in the format PoKeys\_{serial number}.
6. PoKeys devices support different types of Pulse engines (unless the device is designed to operate in only one of them - e.g. PoKeys57CNC and PoKeys57CNCdb25). One can select from the following options:
  - a. Select Disabled if you won't be using the PoKeys Pulse engine as external motion controller in Mach4. PoKeys device can still be used as I/O device and/or pendant.
  - b. Select 'Integrated 3ch' if you will be using internal pulse generator with STEP/DIR signals available directly on PoKeys pins
  - c. Select 'External 4ch/8ch' without IO if you will be using simple external pulse generators that output STEP/DIR signals only (e.g. PoExtBusOC16CNC)

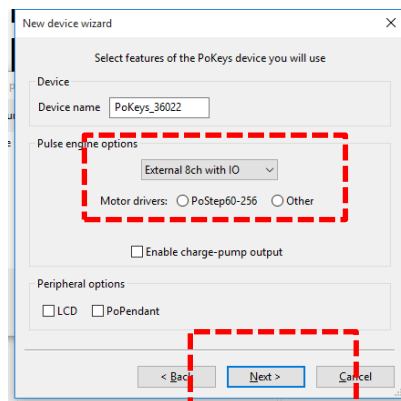
## PoKeys plugin for Mach4 - PoKeys device setup

- d. Select 'External 4ch/8ch with IO' if you will be using more advanced external pulse generators with additional IO capability (e.g. PoKeysCNCaddon)



In case of PoKeys57CNC or PoKeys57CNCdb25, the pulse engine is automatically configured by the plugin and the pulse engine option selection is disabled. Both devices use so called 'External pulse generator' (naming due to backwards compatibility to other PoKeys devices).

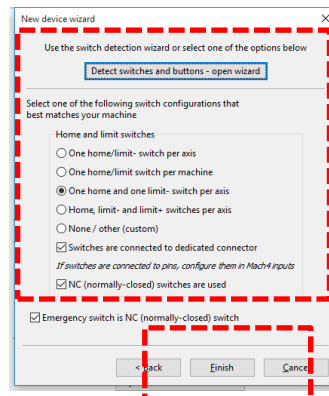
7. If you are using PoStep motor drivers, select the correct one. If using other motor drivers, use the 'Other' option. This setting configures the polarity of the motor enable signal - PoStep25-32 uses inverted (active low) enable signal, while PoStep60-256 and 'Other' option configures the device for normal (active high) enable signal.



8. If LCD or PoPendant1 will be used with this PoKeys device in Mach4, select LCD or/and PoPendant options to load the default configuration for those.
9. Next page of the dialog allows you to configure the machine switches (if Pulse engine has been enabled in previous step). The easiest way to configure the switches is to open the wizard that will guide you through the setup of switches (see chapter *Switches wizard*). To speed up the configuration, there are also common configurations available if switches are wired to default pins.

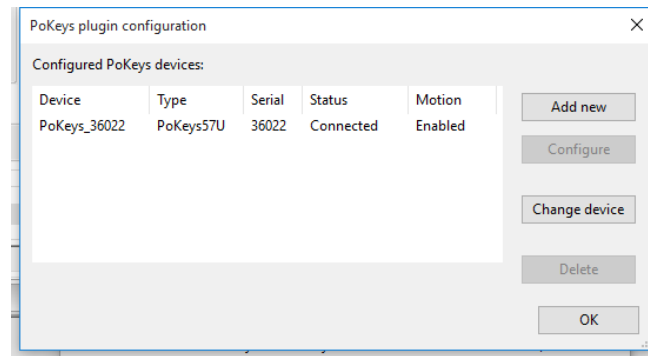


In case of doubt how switches are to be connected to PoKeys device, read the beginners guide on using PoKeys57CNC and Mach4.



## PoKeys plugin for Mach4 - PoKeys device setup

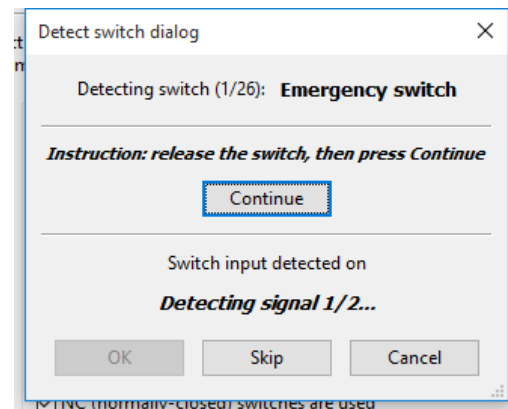
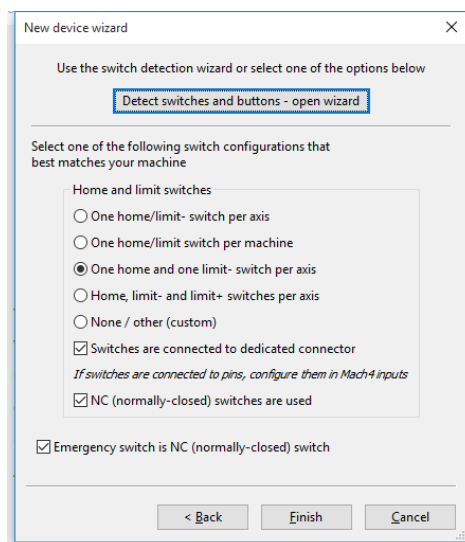
10. After clicking on Finish, the device will now appear listed in the list of configured PoKeys devices. If Pulse engine was enabled, it will automatically be selected as default motion device in Mach4. **Mach4 must be restarted.**
11. After restart, the device in the list should appear with the status 'Connected' and motion 'Enabled' (if Pulse engine was activated).



12. Continue by configuring the device. Select the device in the list and click Configure (or double-click the device in the list). The device configuration dialog will appear with configuration option divided into the multiple sections (tabs).

## Switches wizard

A switch detection wizard is available as a part of new device setup wizard. Open it by clicking on 'Detect switches and buttons - open wizard'.



The wizard will guide you through the setup of main switches that are required for machine operation. Follow the instructions in the wizard - you will be instructed to release and press (for approximately 1 second) individual switches. If certain switch is not present on your machine, click on 'Skip' button. Once all switches are configured, the wizard will close automatically.

## PoKeys plugin for Mach4 - PoKeys device setup

### Configuring axis mapping in Mach4

If your Mach4 installation has not been pre-configured with the motor-axis mapping, open Configure>Mach menu and switch to Axis mapping. A clean installation will have no axis mapping, as shown below, which will make Mach4 unable to move any motor.

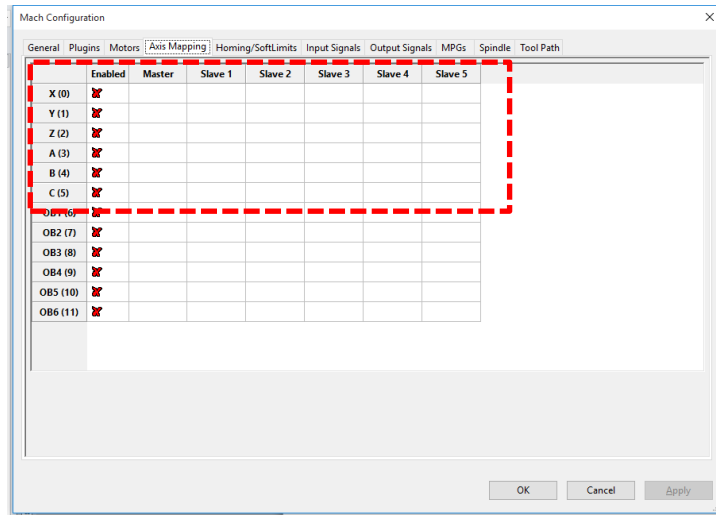


Figure 2: No axis mapping setup

Configure the 'Enabled' and 'Motor' columns with available motors. First motor output connector on the PoKeys device equals to 'Motor0', second one to 'Motor1' etc.

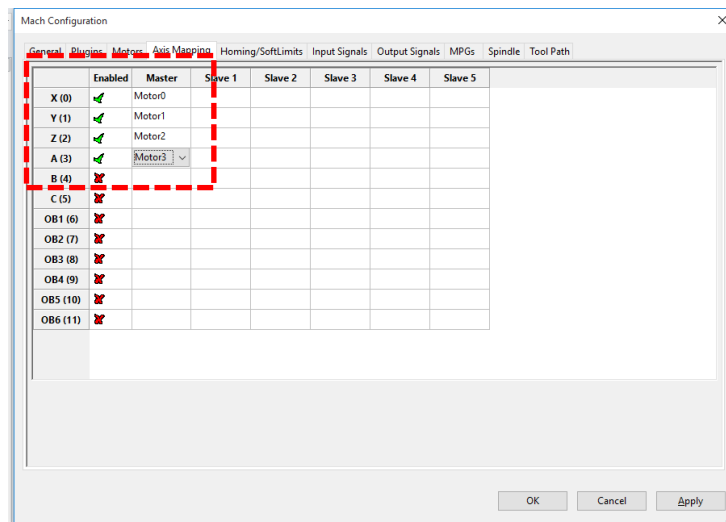


Figure 3: Standard axis mapping configuration



## PoKeys plugin for Mach4 - PoKeys device setup

### PoKeys Device configuration

Most of the PoKeys device configuration is done using Mach4's native configuration screens. Each configured PoKeys device in Mach4 is shown as a set of different devices, grouping the settings accordingly. All devices start with the common name (settable in the PoKeys configuration, PoKeys\_<serial number> by default) and end with descriptive tails:

- PoKeys\_<serial number>: Base PoKeys device object that holds the input/output pins, analog inputs, counters, LCD contents, PWM outputs, PoIL shared data slots etc.
- PoKeys\_<serial number> PoExtBus: This device holds PoExtBus outputs
- PoKeys\_<serial number> Matrix KB: This device holds statuses of matrix keyboard inputs (up to 128 inputs)
- PoKeys\_<serial number> Matrix LED 1/2: These two devices hold matrix LED outputs
- PoKeys\_<serial number> PE: Pulse engine related device. This device indicates statuses of dedicated home/limit switches, actual global state of the home/limit inputs (according to the configuration), dedicated axis enabled outputs etc.

### Configuring inputs

Use the Configure>Mach to setup inputs under 'Input Signals' page. Select the appropriate PoKeys device in the list of devices and select the desired input name.

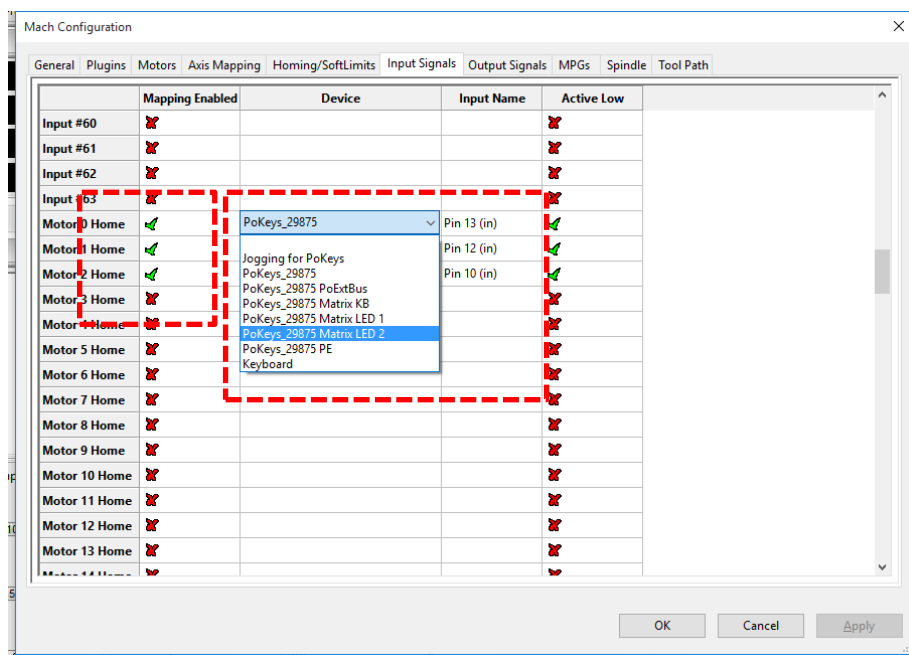


Figure 4: Inputs configuration



New device wizard should automatically configure the inputs for homing and limit switches. Use the described approach to change the settings if required.

## PoKeys plugin for Mach4 - PoKeys device setup

### Configuring outputs

Use the Configure>Mach to setup inputs under 'Output Signals' page. Select the appropriate PoKeys device in the list of devices and select the desired output name.

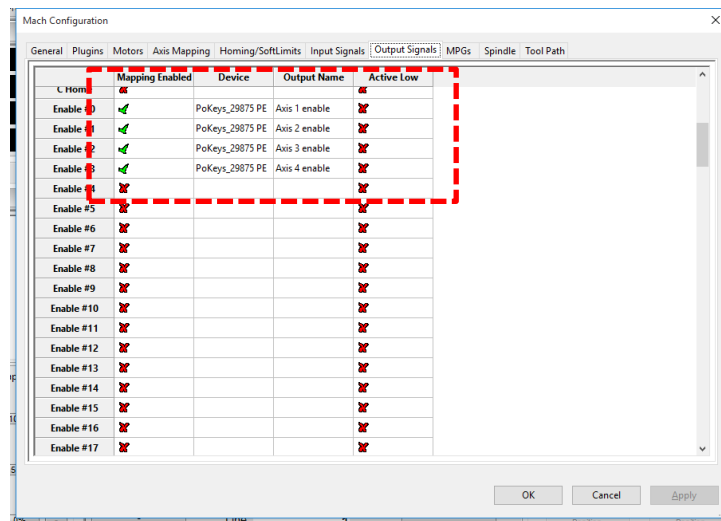


Figure 5: Outputs configuration



As with inputs for switches, outputs for motor driver enable signals should have been automatically preconfigured by the wizard.



Relays and open-collector outputs on PoKeys57CNC device are available under 'PoKeys\_<serial number> PE' device.

### Configure Homing and limits

PoKeys devices fully integrate with Mach4 and use Mach4's homing and limits setup. In case of any doubt, refer to the Mach4 user manual for more information.

The homing and limits configuration is accessible via 'Homing/SoftLimits' configuration page of Mach4 configuration dialog. The configuration is available for each axis separately.



Note a distinction between a motor and axis - motors are related to the physical outputs on the PoKeys device, while axes are related to machine motion direction. Mach4 uses Axis mapping settings to bind the two together. Homing is therefore configured per axis (not per motor).

Mach4 offers the following options:

- Home Dir: define the homing direction. Pos(itive) homing direction is in positive direction of the axis.

## PoKeys plugin for Mach4 - PoKeys device setup

- Home Order: this setting defined the sequence number for homing. Mach4 first homes axis 1, then continues with axis 2 etc. If multiple axes are configured with the same number, all axes are homed at once. A value of 0 will prevent Mach4 from homing the axis.
- Home Offset: defines the position of the home switch in regards to the origin of your machine.
- Home Speed%: this setting tells PoKeys device at which speed (percentage of the full speed) the homing is to be done.
- Home In Place: by enabling this setting, the home switch will be bypassed and current position will be marked as home.
- Soft Enable: enable software limits for the selected axis. Mach4 will prevent jogging outside the limits. You can use this option as an alternative to physical limit switches - note that this option cannot protect the device in case of stalled motion and as result, the misplaced limits. Values for the minimum and maximum limit positions are configured in last two columns.

Mach Configuration

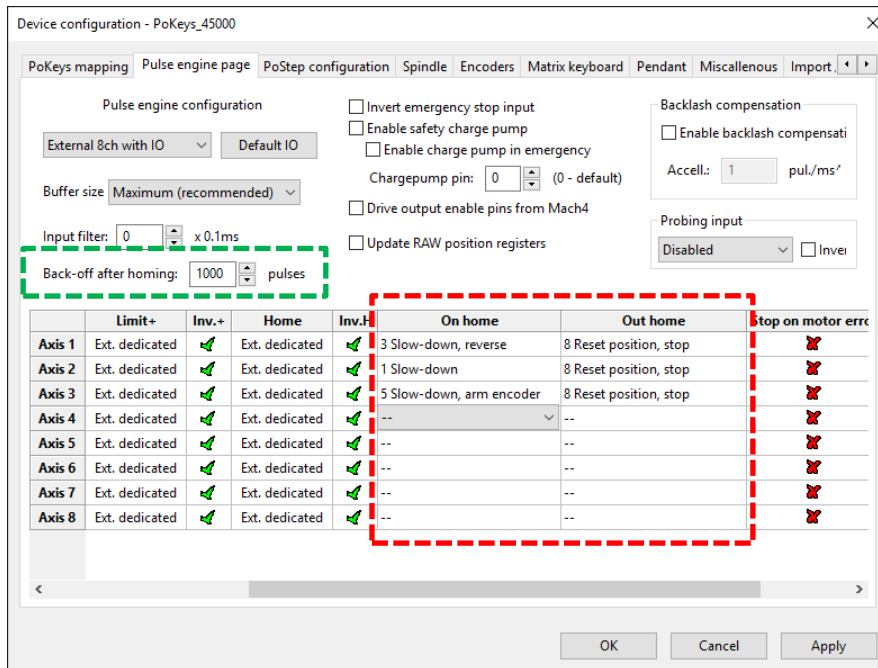
General Plugins Motors Axis Mapping **Homing/SoftLimits** Input Signals Output Signals MPGs Tools Spindle Tool Path

	Home Dir	Home Order	Home Offset	Home Speed%	Home In Place	Soft Enable	Soft Min	Soft Max
X (0)	Pos	2	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-24,0000	0,0000
Y (1)	Pos	2	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-14,0000	0,0000
Z (2)	Pos	1	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
A (3)	Pos	3	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
B (4)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
C (5)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB1 (6)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB2 (7)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB3 (8)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB4 (9)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB5 (10)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000
OB6 (11)	Pos	0	0,0000	20,00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	0,0000	0,0000

OK Cancel Apply

## PoKeys plugin for Mach4 - PoKeys device setup

In addition to standard homing configuration, PoKeys devices offer additional configuration of the homing behavior. Open PoKeys plugin configuration, select the device and switch to 'Pulse engine' tab.



The settings above configure the normal homing behavior for Axis 1 - slow-down and reverse the motion when hitting the home switch, while on home switch release reset the position and stop the motion. Axis 2 is configured a bit differently - hitting the home switch will slow-down the motion, while releasing the switch will result in the same action.



By default, PoKeys uses standard homing behavior and this setting does not need to be changed if a different homing motion is not required.

Axis 3 uses a combination of home switch and axis encoder - when home switch is detected by the PoKeys device, the motion will be slowed-down and the encoder index input will be armed. On the next encoder index signal change, the position will be reset and motion stopped. This approach enables the use of inaccurate switches with large hysteresis and highly-accurate and repetitive index pulses (in case of servo motors). This option uses Fast encoders peripherals and is fixed to each motor - first motor uses Fast encoder input 1, second one the Fast encoder input 2 and the third one the Fast encoder input 3.

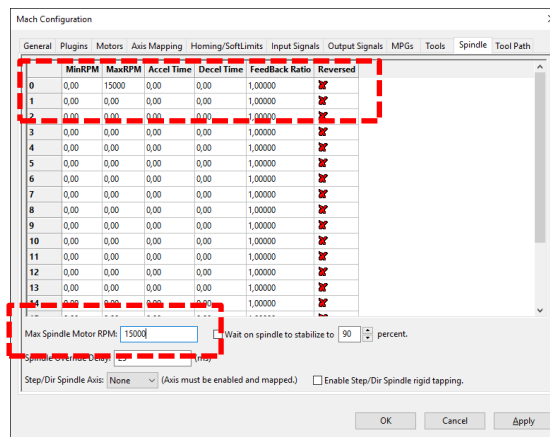


The green square in the picture above shows the back-off option configuration. The PoKeys device can be configured to back-off from a switch after homing procedure is complete in order to prevent accidental switch activation due to vibrations (these would result in E-stop if switch is used both for homing and limits). The value for the distance is specified in motor steps/pulses.

## PoKeys plugin for Mach4 - PoKeys device setup

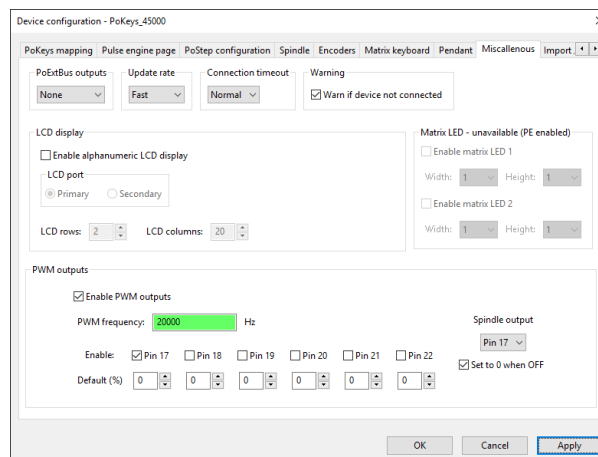
### Configuring spindle output

To use the spindle output with PoKeys and Mach4, first configure the Mach4's spindle settings. Open Mach4 configuration dialog and switch to Spindle tab. The table contains spindle settings for different spindle ratios. Set the MaxRPM in the table and Max Spindle Motor RPM in the setting below. These values equal to the spindle speed at maximum value of spindle speed output from the PoKeys device (e.g. in the example below, the spindle will rotate at 15.000 RPM when PoKeys outputs maximum PWM duty cycle signal of 100% or an analog output of 10 V). If lower spindle speed is commanded with the Sxxxx command in Mach4, the PoKeys device will output proportionally lower signal (e.g. if S3000 command is executed in Mach4, PoKeys will output PWM signal with 20% duty cycle or 2 V if spindle 0-10V analog output is used).



### Configuration of spindle output in PoKeys plugin

Besides the configuration in Mach4 settings, PWM output to drive the spindle must be configured in PoKeys plugin settings. Open the Miscellaneous tab and look at the PWM outputs configuration - Spindle output selection box is found on the right, while PWM signal configuration is available on the left. Appropriate PWM output must be enabled.



When using 0-10V analog output for spindle speed, set the PWM frequency to 20000 (20 kHz) and use the pin 17 in case of PoKeys57CNC.

## PoKeys plugin for Mach4 - PoKeys device setup

### Other PoKeys settings

Other PoKeys settings are accessed via PoKeys plugin configuration dialog.

PoKeys device configuration is divided into multiple sections (tabs) that can be accessed from the plugin configuration dialog. Open Configure > Plugins > PoKeys plugin - PoLabs and select the device from the list. Open the configuration dialog by double-clicking the device or click 'Configure' button.

### PoKeys mapping page

Here, PoKeys pins can be assigned different pin functions and their state can be inverted. If PoKeys pins are assigned to a specific IO in Mach4 configuration screen, this mapping will appear under 'Status'.

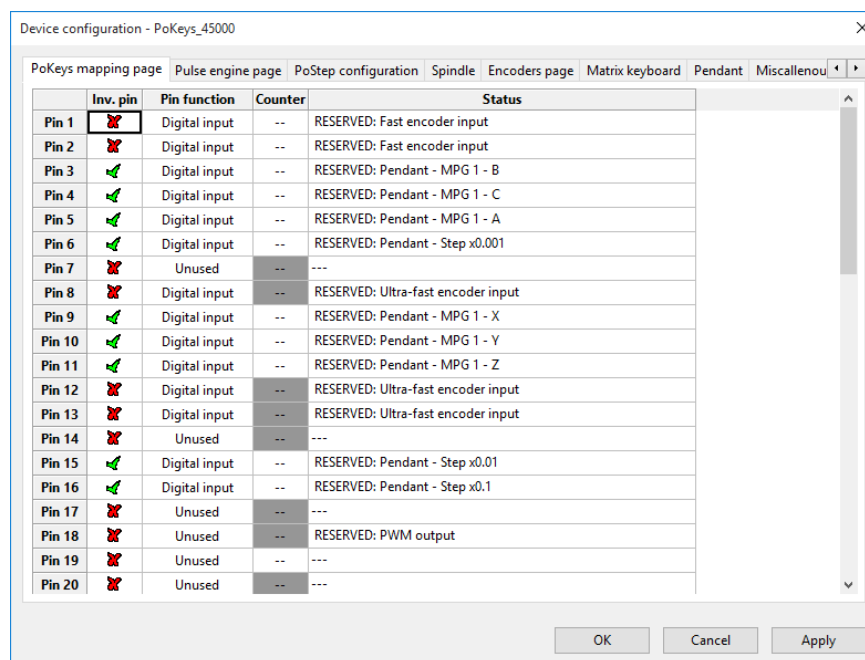


Figure 6: PoKeys mapping page

## PoKeys plugin for Mach4 - PoKeys device setup

### Pulse engine

The PoKeys Pulse engine is setup in this section.

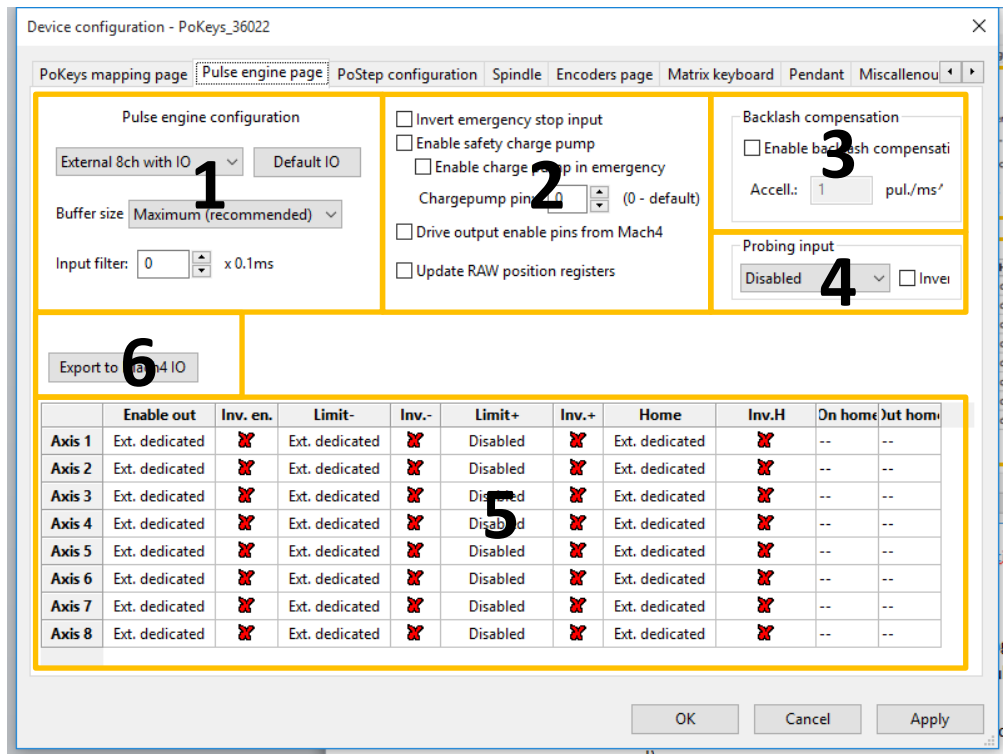


Figure 7: PoKeys Pulse engine configuration

The numbered areas correspond to the following settings:

1. Main Pulse engine settings - here, the type of the Pulse engine can be selected and buffer size/input filter adjusted. PoKeys devices support these pulse engine types:
  - a. Internal 3ch: available on general-purpose PoKeys IO devices (PoKeys56U, -56E, -57U, -57E). It uses predefined pins for step/direction outputs (see Pulse engine manual).
  - b. External 4ch/8ch without IO: available in combination of a general-purpose PoKeys IO device and CNC add-on board without IO capabilities (PoExtBusOC16CNC).
  - c. External 4ch/8ch with IO: available on PoKeys57CNC, PoKeys57CNCdb25 and in combination of general-purpose PoKeys IO device and CNC add-on board with IO capabilities (PoKeysCNCaddon or other 3<sup>rd</sup> party solutions).

In case of External pulse generator with IO capabilities, home and limit inputs can be connected to dedicated home/limit inputs connectors. Clicking 'Default IO' will map the Limit-, Limit+ and Home inputs to the external pulse generator IO.

Buffer size determines how much data is buffered by the device. It is recommended to leave the buffer size to a maximum value (up to 128 ms of data) to allow the device to gracefully handle short communication interrupts.

Inputs for home and limit switches support digital filtering. Setting the value of input filter to 0, disables the filter. Otherwise, the value determines the amount of time that device

## PoKeys plugin for Mach4 - PoKeys device setup

expects the signal to be active before the device responds to the signal. This option can be used to address the electrical noise on the switch input signals. One unit equals to 0.1 ms (0.0001 s) - the filter can be configured for 0 to 25.4 ms.

### 2. Additional Pulse engine options:

- Invert emergency stop input: By default, PoKeys expects a NC (normally-closed) switch to be used for emergency stop switch. If NO (normally-open) switch is used, check this option
- Enable chargepump output: PoKeys chargepump output generates a 5 kHz square signal in 'Running' state on the selected chargepump output pin.
- Drive output enable pins from Mach4: enable this option to allow Mach4 individually control enable outputs of each axis. By default, this option is disabled and enable signals depend on the 'Running' state. This option is suggested to be enabled to use the consecutive enabling of axis drivers (and thus avoiding high inrush currents due to all drivers enabling at the same time)
- Automatically map Mach4 signals: enable this option to automatically transfer PoKeys plugin settings into Mach4 configuration. PoKeys plugin has a priority over Mach4 dialog settings in this case.
- Update RAW positions registers (for debugging purposes): enable this option to export RAW axes positions to Mach4 registers.

### 3. Backlash compensation options: enable backlash compensation in the device by checking this option. Pokeys plugin reads the backlash width settings from Mach4 motor configuration. Motion acceleration for backlash compensation motion controller can be adjusted in steps of 1 pulse/ms<sup>2</sup>.

### 4. Probing input : select the pin you wish to use as a probing input. Based on your probe input type, you may need to select 'Invert polarity'

### 5. Axis options: here the standard signals for each axis can be configured. Since stepper drivers differ in the enable signal state, make sure that 'Inv. en.' (invert enable signal) option is properly configured for your drivers (PoStep25 requires the signals to be inverted). The 'On home' and 'Out home' columns define the functionality of the homing algorithm. See PoKeys57CNC user manual on more information on homing algorithm configuration.

Last two columns configure the behavior of the PoKeys device on motor driver error signals. If 'Stop on motor error' option is not enabled, checking of the error signal is skipped. If motor driver outputs active low error signal, check the 'Invert error' option.



## PoKeys plugin for Mach4 - PoKeys device setup

### PoStep configuration

PoStep60 series of PoStep drivers support PoExtension2 bus connection with PoKeys57CNC device, which can be used to configure the drivers via PoKeys57CNC device. First, configure a unique I2C address to a PoStep device using PoStep configuration tool (USB connection with the PoStep driver is required for this step).

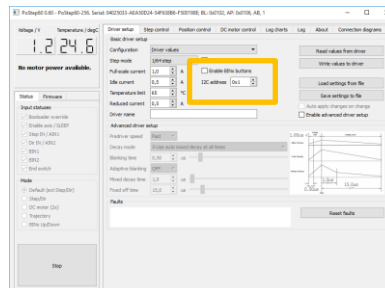


Figure 8: I2C address configuration in PoStep configuration tool

### Connecting PoStep drivers to PoKeys57CNC

Connect PoStep drivers (PoStep60-256) in parallel with PoKeys57CNC device using PoExtension2 bus. All connectors on the PoExtension2 bus must be crimped in parallel to each other (take care of the connector polarity) - see pictures below.

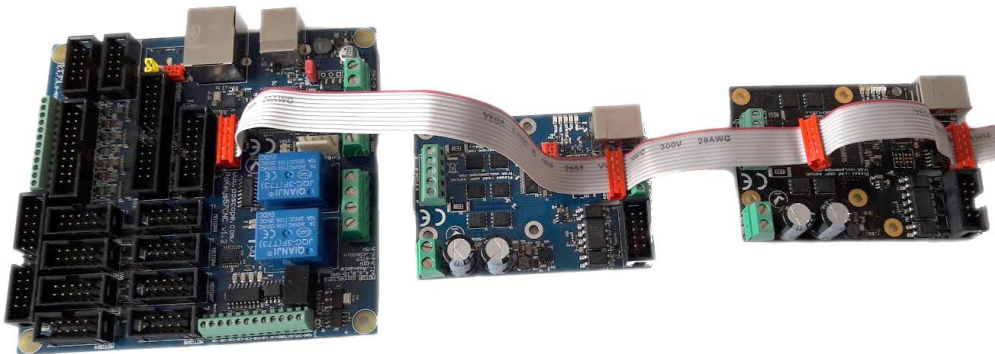


Figure 9: Two PoStep60-256 stepper motor drivers connected to PoKeys57CNC device



Figure 10: PoExtension2 cable for connecting PoStep drivers with PoKeys57CNC device

## PoKeys plugin for Mach4 - PoKeys device setup

### Configuration of PoStep driver in Mach4 PoKeys plugin configuration dialog

Select the PoStep driver I2C address for the selected PoStep driver in the list and click on 'Write changes'. PoKeys device will establish connection with PoStep driver and fetch its configuration. Step mode, temperature limit and currents settings can be accessed directly from this dialog.

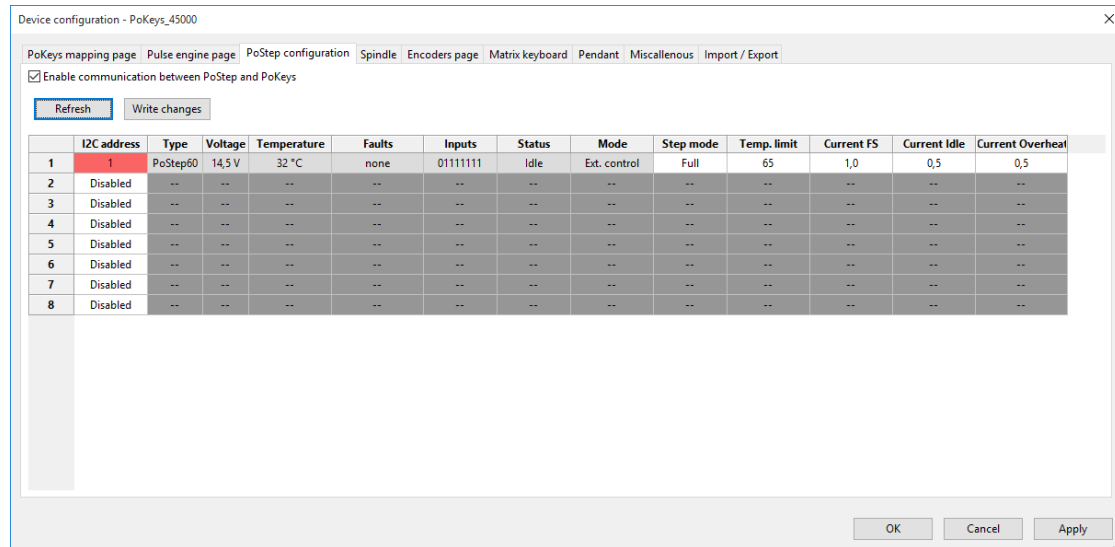


Figure 11: PoStep driver configuration in Mach4 PoKeys plugin settings



Unique I2C bus addresses must be assigned first to each PoStep60 driver and this same value entered in the I2C address column in the configuration.

### Spindle settings

Spindle settings can be found on a dedicated configuration section. Here, spindle speed measurements can be enabled. For mill operation (higher spindle speeds), first two options are available, while for lathe operation (lower spindle speeds and threading), the second two options are available. Either index or encoder (with index for threading) can be selected to measure the spindle speed. **If encoder option is selected, 'ultra-fast' encoder is used for this purpose.**



The encoder resolution (ticks per revolution) is the 1x resolution of the encoder - PoKeys will later decode the signals at 4x higher resolution automatically.

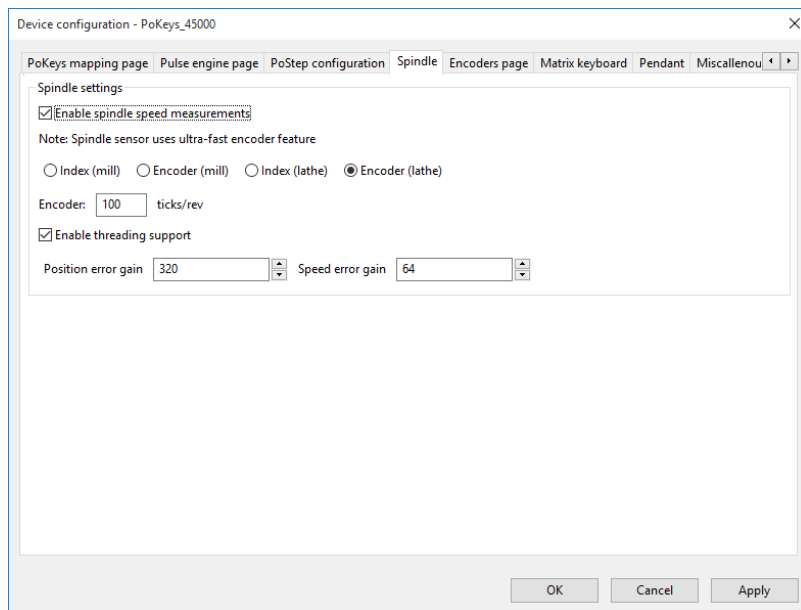


Figure 12: Spindle settings page

### Spindle settings - threading operation

PoKeys device support threading operation using spindle encoder with index. Enable the support for threading by enabling the 'Enable threading support' option.

Connect the spindle encoder (which must be mounted with 1:1 ratio of rotational speed to spindle itself) to dedicated encoder connector on the PoKeys57CNC device.

It is also possible to do the threading with index signal only - connect the index signal to index of the ultra-fast encoder input. In PoKeys57CNCdb25 connect index to pin 15.



PoKeys devices use state-of-the-art spindle speed estimation filters that not only follow the spindle speed variations, but also remove the long-term position drifting errors due to the spindle speed variations. The default values for the filters are 320 for position error gain and 64 for speed error gain.

## PoKeys plugin for Mach4 - PoKeys device setup

### Encoders page

Here the encoders (e.g. MPGs) are configured with standard settings available. We suggest using 'Fast encoders' option with MPGs (first three encoder rows when the option is enabled) and standard encoders for various other adjustment knobs (spindle or feed speed adjustment knob, etc.).

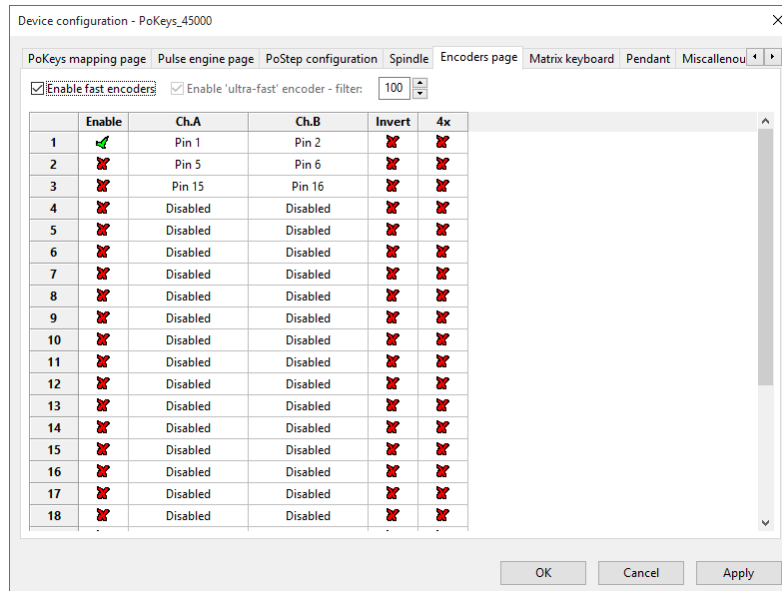


Figure 13: Encoders configuration dialog

Digital filtering for the 'ultra-fast' encoder can be adjusted in the setting field at the top.



PoKeys devices use three different approaches to decoding of the encoder signals. The following are available:

- Normal encoders: slowest decoding, appropriate for adjustment knobs
- Fast encoders: medium-speed decoding, appropriate for MPGs (manual pulse generators) or low-resolution axis encoders
- Ultra-fast encoder: single dedicated encoder input for high-speed encoders, appropriate for high-resolution encoder at higher speeds that are used to track spindle position

## PoKeys plugin for Mach4 - PoKeys device setup

### Matrix keyboard

In case of larger array of switches (not to be used as axis limit or home switches), matrix keyboard feature can be used to read them all.

Enable the matrix keyboard support first, then configure the pins that are used for rows and columns of the matrix keyboard.

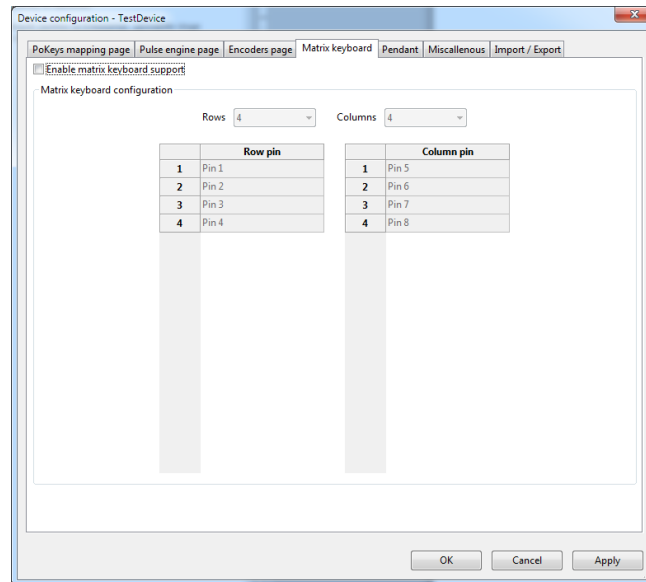


Figure 14: Matrix keyboard configuration dialog

The matrix keyboard statuses are then available via PoKeys Matrix KB device object in Mach4 configuration.

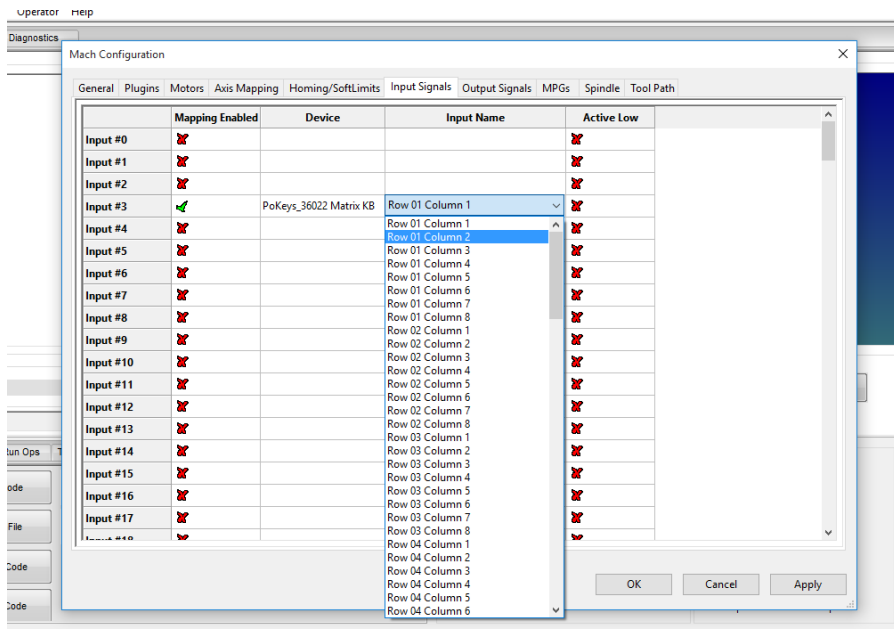


Figure 15: Matrix keyboard inputs

## PoKeys plugin for Mach4 - PoKeys device setup

### Pendant

PoKeys plugin for Mach4 integrates a very powerful and versatile support for various pendants. Key-based and MPG-based pendants are supported, as well as a combination of those.

The concept behind the pendant functionality is based on a set of pendant functions that can be enabled for each pendant (usually pendant will contain only a subset of all possible functions). These functions can be activated by a push-button switch (labelled as 'Button' later on) and/or toggle switch (labelled as 'Switch' later on). The difference between the two is in the way the specific pendant functions is activated - in case of push-buttons, the action of pressing the button triggers the command (e.g. toggling different settings, Cycle start, Stop, etc.), while in case of toggle switches, the current state of the switch is processed (e.g. enable jog, select axis, jog x in + direction, etc.). The state of most pendant functions can also be signaled on pendant's LEDs.

PoKeys plugin for Mach4 supports multiple pendants on multiple PoKeys devices. The plugin joins information from all pendants and executes the commanded jogging or control commands.

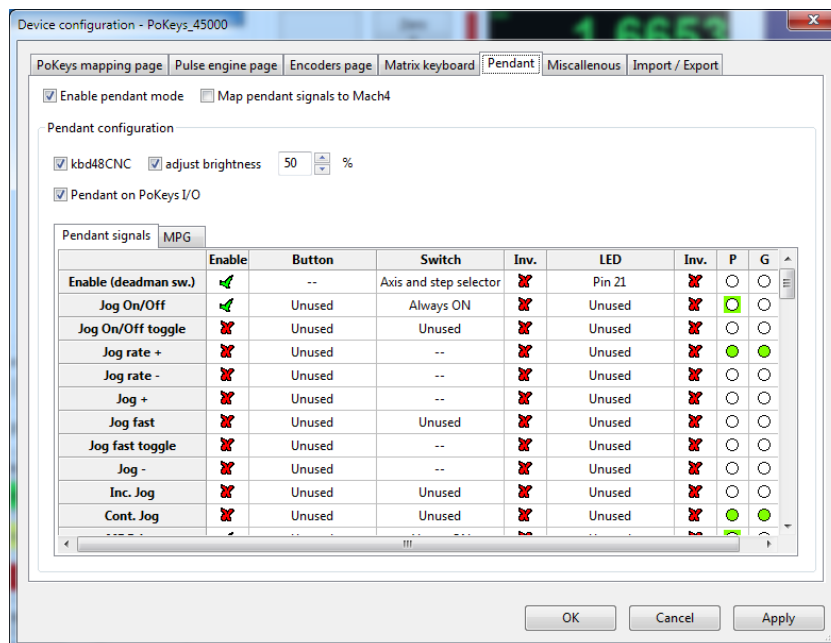
### Pendant setup

In order to setup the pendant, the following options are to be considered:

- Enable pendant mode: tells the plugin that the pendant is going to be used (this option must be activated for any pendant to work)
- kbd48CNC: enable the pre-configured support for kbd48CNC keyboard pendant
- Pendant on PoKeys I/O: enable the support for various pendants (both keyboard- and MPG-based), connected to PoKeys I/O pins

### Configuring Pendant on PoKeys I/O

The pendant configuration is divided into Pendant signals and MPG sections. The Pendant signals contains the list of all possible pendant functions and configuration section, where each pendant function can be assigned to a specific PoKeys I/O pin.



## PoKeys plugin for Mach4 - PoKeys device setup

### Pendant functions

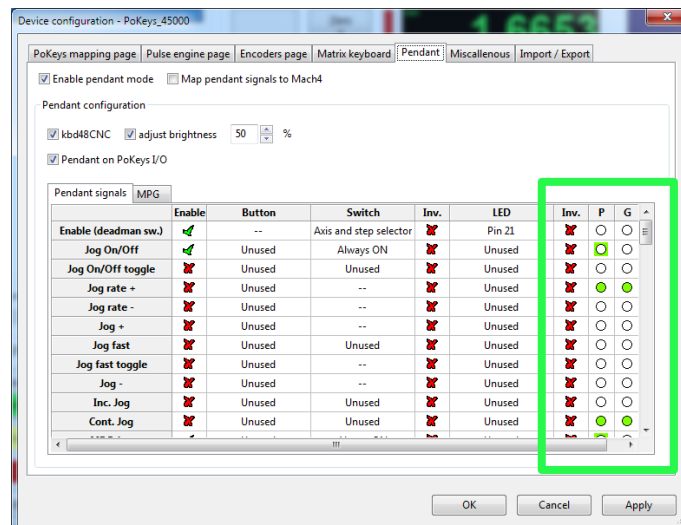
- **Enable (deadman sw.):** since PoKeys supports multiple pendants, each pendant can independently be enabled or disabled - this is done using this function. When disabled, the jogging motion is stopped and all LEDs are turned off. Pendants that contain the dead man's switch (also named Operator Presence Control, enabling switch) can use the signal from this switch to automatically enable or disable the pendant. This function should not be mixed with emergency (or Reset) switch. None of the pendant functions is taken into account while the pendant is disabled.
- **Jog On/Off:** pendants can support both the control (starting, stopping, pausing the job in execution, etc.) and jogging functionality. The jogging functionality must be enabled and this pendant function takes care of that. Since the keyboard-based pendants use push-button switches, the 'Jog On/Off toggle' function should be used instead since the jogging functionality is toggled on or off on each input signal activation.
- **Jog On/Off toggle:** similar as above, but is for use with pushbuttons (toggle switches). The jogging state is changed on each button press
- **Jog rate+/Jog rate-:** change jog rate (the value is displayed in Mach4 under jogging tab)
- **Jog fast:** toggle full-speed jogging
- **Jog +/-:** jog the current axis
- **Inc./Cont./MPG jog:** switching between incremental, continuous and MPG jogging.
- **Jog X/Y/Z/A/B/C +/-:** jog the corresponding axis
- **Axis X/Y/Z/A/B/C:** axis selection
- **MPG 1/2/3 - X/Y/Z/A/B/C:** assign the axis to the appropriate MPG
- **Step x0.001/x0.01/x0.1/x1:** select step size (0.001, 0.01, 0.1, 1.0 of the unit)
- **Cycle start/Feed hold/Stop/Reset:** start, hold, stop or reset job
- **Single step:** execute single step (*not supported*)
- **Reverse:** reverse jog execution (*not supported*)
- **Optional stop:** toggle optional stop
- **Edit:** open G-code editor (*not supported*)
- **Load:** open G-code open dialog (*not supported*)
- **Close:** close current G-code
- **Recent:** open list of recent G-code files (*not supported*)
- **Set next line:** set next line to execute (*not supported*)
- **Rewind:** rewind current G-code file
- **Run from here:** run G-code from current line (*not supported*)
- **Block delete:** Block delete option
- **Ref:** take reference measurement (*not supported*)
- **Goto 0's:** goto home command - the button must be pressed for more than 1 second for the function to activate
- **Spindle CW/Stop/CCW:** turn on or off the spindle and set the direction
- **Spindle speed +/-/reset:** set the spindle speed
- **Feedrate +/-/reset:** set the federate
- **Teach/Stop teach:** start or stop teach mode (*not supported*)
- **Shuttle mode:** shuttle mode (*not supported*)

## PoKeys plugin for Mach4 - PoKeys device setup

- **Emergency switch:** emergency switch that disables the Mach4 motion engine. Make sure to use proper hardwire-wiring of emergency switch to the machine power supply. This signal can only be used to notify Mach4 of the event that emergency switch has been activated.

### Pendant function status display

The two rightmost columns of the Pendant configuration screen display the actual pendant and global (aggregated) pendant states. The actual pendant (column P) displays the information on the state of the pendant that the configuration is being edited for. It will display the information even if the pendant is not enabled ('Enable (deadman sw.)' function active). The last column 'G' displays the aggregated function states of all pendants - this state is being used to trigger the jogging or command events.



Each cell displays one of the following states:

- inactive
- function is triggered by the input, but has inactive display (LED) state
- function is not triggered by the input, but has an active display (LED) state
- function is both triggered by the input and has an active display (LED) state



## PoKeys plugin for Mach4 - PoKeys device setup

### Example: configuring PoPendant1

**Note: the following table only gives an example on how to connect the PoPendant to PoKeys device. To ease the setup process, the configuration file for this example is provided on PoPendant homepage. Wiring can be rearranged by the user, but the plugin configuration must be adjusted accordingly.**



Wire	PoPendant "wire colour"	Function	PoKeys pin number	Mach4 Mapping
1	red	MPG +5V	5V	/
2	black	MPG GND	GND	/
5	green	MPG A	1	Encoder 1B
7	white	MPG B	2	Encoder 1A
	purple	N.C.	N.C.	/
	purple/black	N.C.	N.C.	/
9	green/black	Lamp +	+3.3V	/
11	white/black	Lamp -	14	Pendant enable
13	yellow	X axis	19	Axis X
15	yellow/black	Y axis	20	Axis Y
17	brown	Z axis	21	Axis Z
19	brown/black	A axis	22	Axis A
10	pink*	B axis	24	Axis B
12	pink/black*	C axis	27	Axis C
21	gray	x1	3	Step 0.001
23	gray/black	x10	4	Step 0.01
25	orange	x100	7	Step 0.1
	orange/black	Ctrl Switch	GND	/
4	Light blue	Estop	52	IO Estop
6	blue/black	Estop GND	GND	/
	red/black	N.C.	N.C.	/
	shield	shield	GND	/

**N.C.= not connected - wire it to GND**

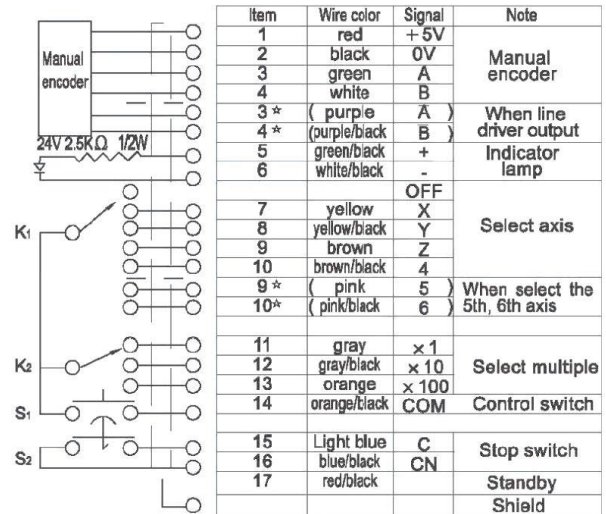


Figure 16: PoPendant internal wiring

PoPendant1 is an MPG-based pendant with MPG, axis and step selection switches, emergency stop button and dead-man switch on the side of the device.

In order to make it work in Mach4 plugin, the following settings must be selected:

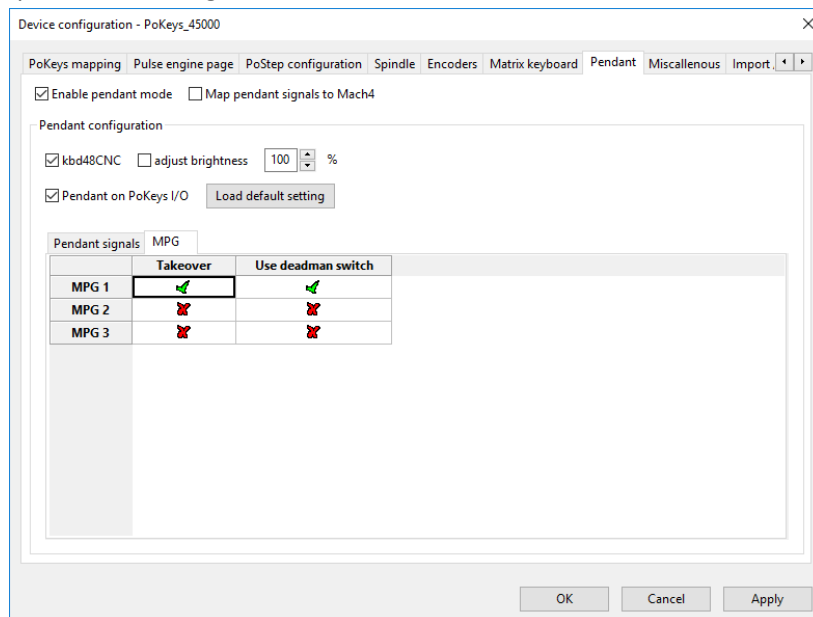
- Enable pendant mode: 'master' enable mode for pendants
- Pendant on PoKeys I/O: Pendant1 is connected to PoKeys I/O pins
- Enable (deadman sw.): select 'Axis and step selector' signal under 'Switch' column. Since dead-man switch in PoPendant1 is wired in series with axis and step selector, signal of those must be used to enable the pendant
- Jog On/Off: since no switch for turning the jogging functionality on or off is present on PoPendant1, select 'Always ON' under 'Switch' column.
- MPG jog: this pendant is using MPG to jog - select 'Always ON' under 'Switch' column
- MPG 1 - X,Y,Z,A,B,C: select which pins the axis selection switch on the PoPendant1 is wired to

## PoKeys plugin for Mach4 - PoKeys device setup

- Step x0.001, x0.01, x0.1: select which pins the step selection switch on the PoPendant1 is wired to

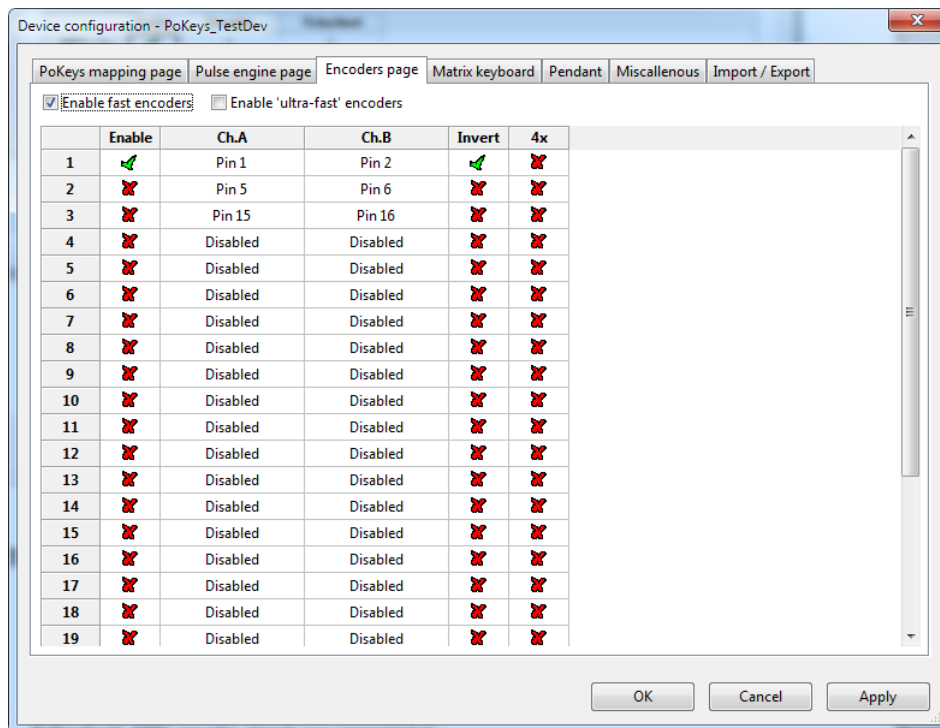
Enable (deadman sw.)	<input checked="" type="checkbox"/>	--	Axis and step selector	<input checked="" type="checkbox"/>	Pin 21	<input checked="" type="checkbox"/>
Jog On/Off	<input checked="" type="checkbox"/>	Unused	Always ON	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog On/Off toggle	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog rate +	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog rate -	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog +	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog fast	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog fast toggle	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Jog -	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Inc. Jog	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Cont. Jog	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG Jog	<input checked="" type="checkbox"/>	Unused	Always ON	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - X	<input checked="" type="checkbox"/>	Pin 9	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - Y	<input checked="" type="checkbox"/>	Pin 10	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - Z	<input checked="" type="checkbox"/>	Pin 11	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - A	<input checked="" type="checkbox"/>	Pin 5	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - B	<input checked="" type="checkbox"/>	Pin 3	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
MPG 1 - C	<input checked="" type="checkbox"/>	Pin 4	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Step x0.001	<input checked="" type="checkbox"/>	Pin 6	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Step x0.01	<input checked="" type="checkbox"/>	Pin 15	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Step x0.1	<input checked="" type="checkbox"/>	Pin 16	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Step x1	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Cycle start	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Feed Hold	<input checked="" type="checkbox"/>	Unused	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Stop	<input checked="" type="checkbox"/>	Unused	--	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>
Reset	<input checked="" type="checkbox"/>	--	Unused	<input checked="" type="checkbox"/>	Unused	<input checked="" type="checkbox"/>

- Switch to 'MPG' tab and check both tick boxes for MPG 1. The takeover option tells the plugin to take the control over MPG 1 in Mach4 (in order to change its mapping, step per tick etc.), while the deadman switch option enables the use of deadman switch input - if the switch is not pressed, the target MPG is disabled.

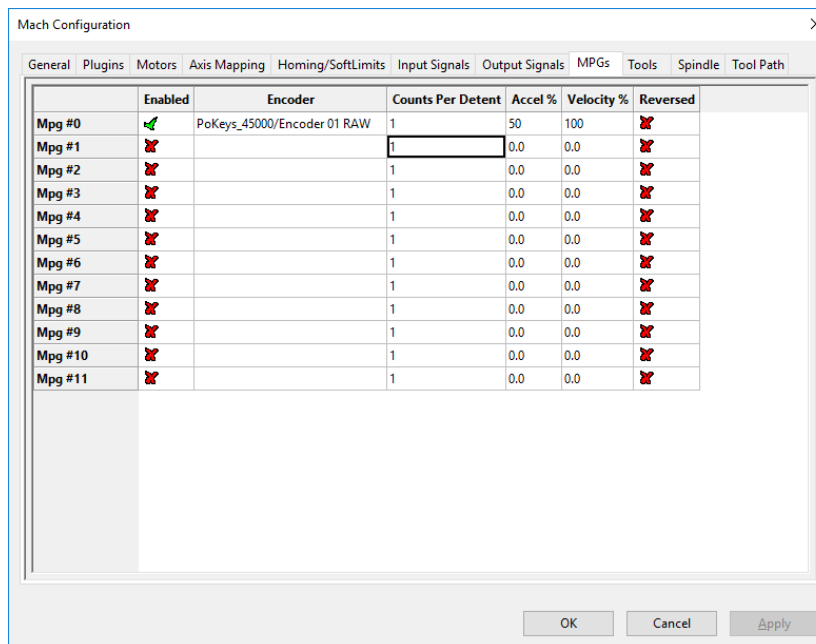


- Switch to Encoders configuration and configure encoder 1 to use the pins, where PoPendant1 MPG encoder signals are connected to.

## PoKeys plugin for Mach4 - PoKeys device setup



- Confirm the plugin configuration by clicking OK and open Mach4 configuration dialog (Configure > Mach...). Switch to 'MPGs' tab and configure the options as shown below.



The deadman switch can have its own dedicated input or can use the axis mapping and step selection knobs, which are usually wired in series with the deadman switch.

## PoKeys plugin for Mach4 - PoKeys device setup

### Pendant function configuration checklist

If pendant function is not reacting to the switch or button input, check that the following conditions are met:

- Is 'Enable pendant mode' option checked?
- Is 'Pendant on PoKeys I/O' option checked?
- Is 'Enable (deadman sw.)' function properly configured and is activated by the input signal?
- Is pendant function enabled (checkbox in the first column)?

## PoKeys plugin for Mach4 - PoKeys device setup

### Miscellaneous

This page contains additional settings for other peripherals. PoExtBus outputs can be enabled here, as well as the support for alphanumeric LCD display, PWM outputs and PoIL data interchange.

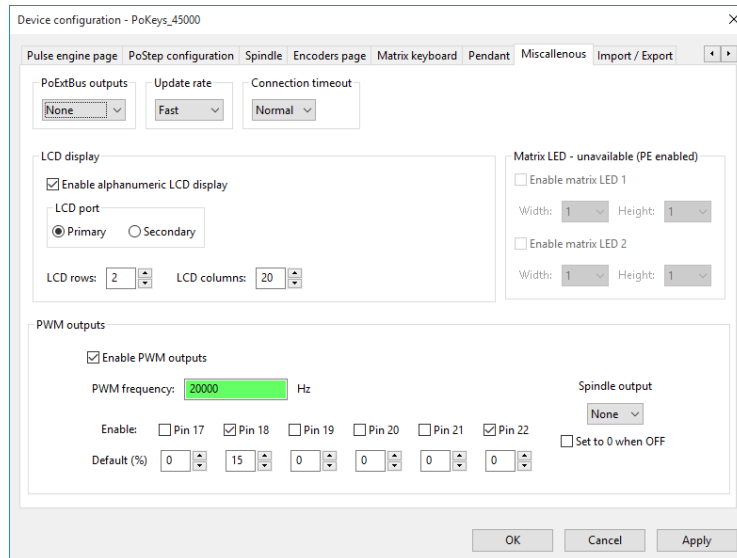


Figure 17: Miscellaneous settings

The update rate and connection timeout settings control the communication with the device. If Pulse engine functionality of PoKeys device is not used, the update rate can be lowered to reduce the load on the system or communication bus.

## PoKeys plugin for Mach4 - PoKeys device setup

### Import/Export of plugin settings

The import/export page is used to import or export PoKeys device configuration from or to a file. Select which settings you would like to import or export and click the appropriate Import or Export button.

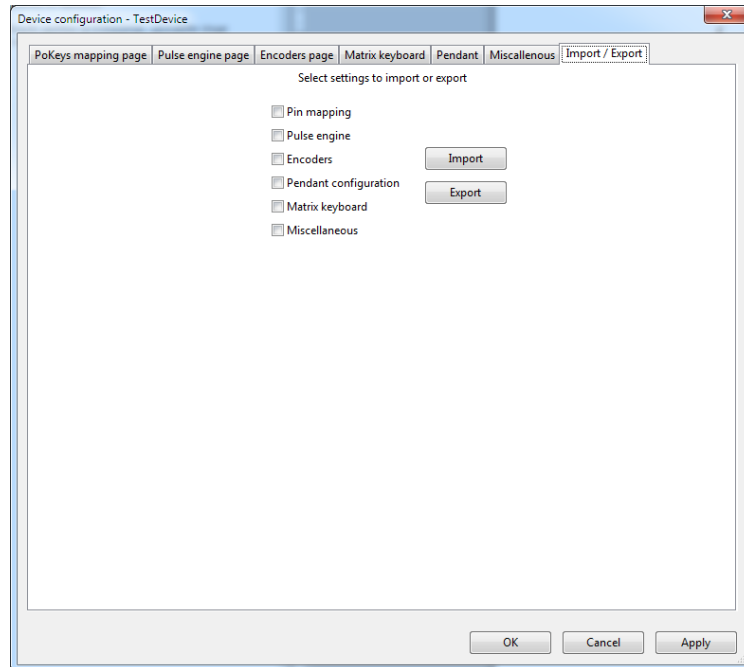


Figure 18: Import/export section

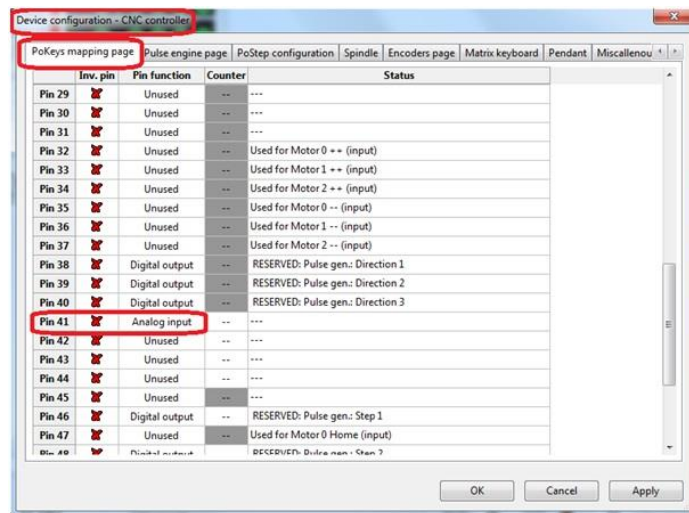
### Peripheral integration examples

Customization of peripheral operation in Mach4 with PoKeys plugin is done via built-in LUA scripting. Read the Mach4 LUA scripting manual on how to access the script editor and how to edit PLC scripts in Mach4.

### FRO (feedrate-override) setting using PoKeys analog input

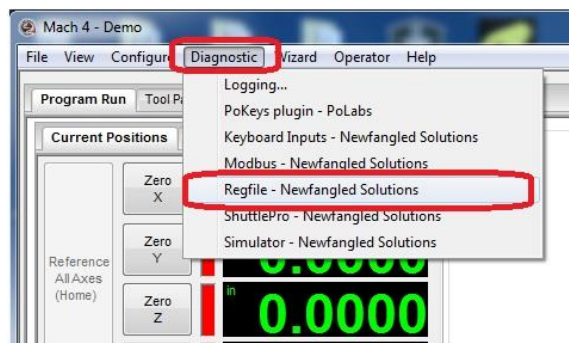
This example can be downloaded from [PoKeys analog\\_FRO\(526b\)](http://blog.poscope.com/wp-content/uploads/2016/04/PoKeys_analog_FRO.zip) ([http://blog.poscope.com/wp-content/uploads/2016/04/PoKeys\\_analog\\_FRO.zip](http://blog.poscope.com/wp-content/uploads/2016/04/PoKeys_analog_FRO.zip)). Move the file to a location easy to find, preferably in the Mach4 folder.

First, enable the analog input in PoKeys plugin. To do this, open the PoKeys plugin configuration in Mach4 and select your device. Then go to the PoKeys mapping page tab and select which pin should be analog input. Make sure to check the PoKeys device pinout schematics (under FAQ section on the device's homepage) for the correct pin number – [PoKeys57U](#), [PoKeys57E](#), [PoKeys57CNC](#).



If you haven't done it yet, connect a potentiometer to the selected analog input of the PoKeys device.

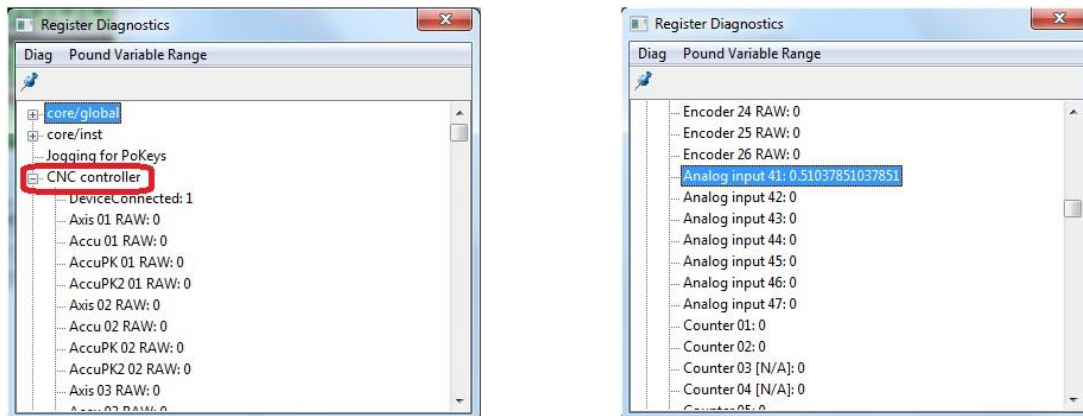
To test if the settings are correct, click on “Diagnostic” tab in Mach 4 and select “RegFile – Newfangled Solutions”.



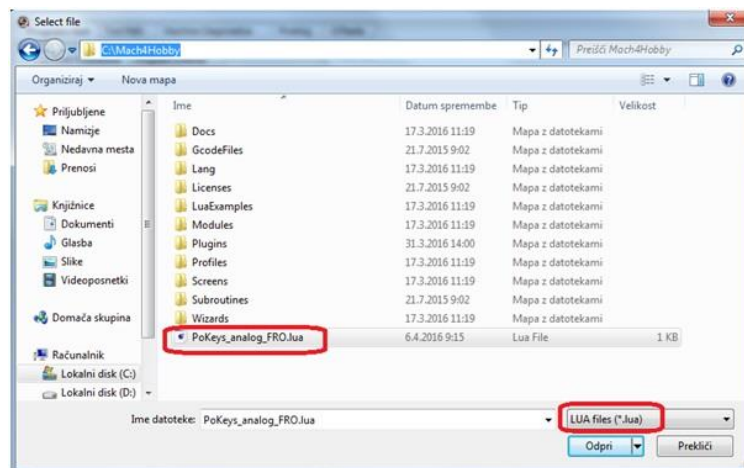
A new window should open. Here you expand your PoKeys device options (in our example, PoKeys device was named “CNC Controller”) and scroll down to analog input. Under desired analog input pin

## PoKeys plugin for Mach4 - PoKeys device setup

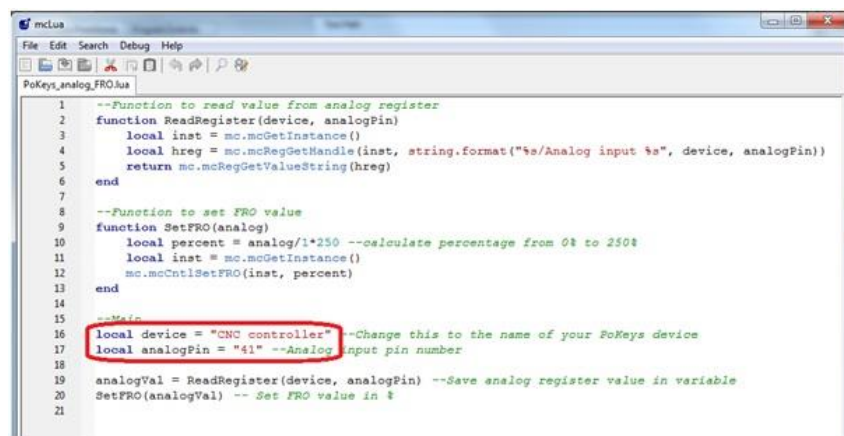
you should see the value change from 0 to 1 as you turn the potentiometer. Value of 0 equals to a voltage of 0 V, while the value of 1 equals to voltage of +3.3 V on the analog input pin.



Now open the "PoKeys\_analog\_FRO.lua" file from Mach4. To do this click on "Operator" menu inside Mach4 and select "Edit/Debug scripts". Go to the location of the .lua file and open it.



In this file, you must change the "device" and "analogPin" values inside double quote (do not remove the double quotes). The device name is the name of your device and the analogPin value is the analog input pin number.



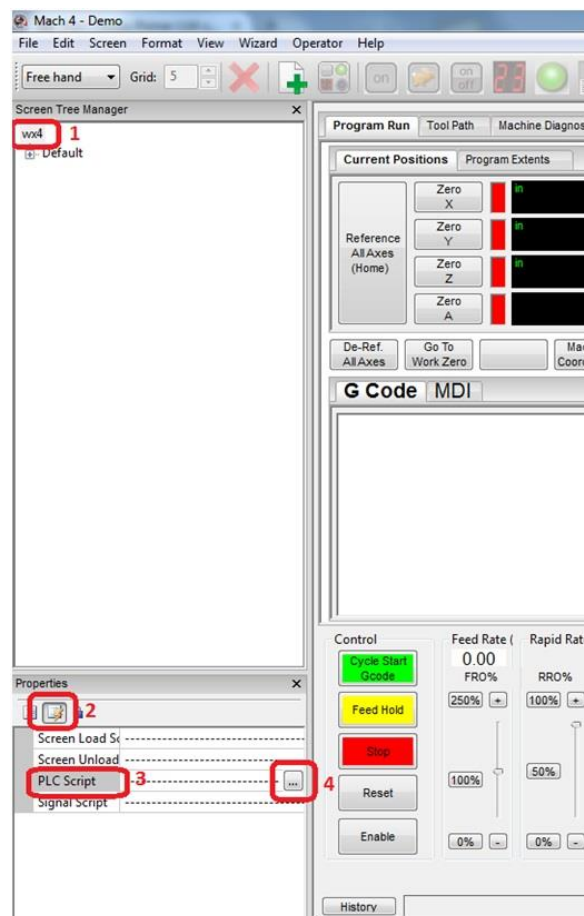


## PoKeys plugin for Mach4 - PoKeys device setup

In this example the FRO will change from 0 to 250%. If you want to have a different range you must edit this line:

*local percent = analog/1\*250 –calculate percentage from 0% to 250%*

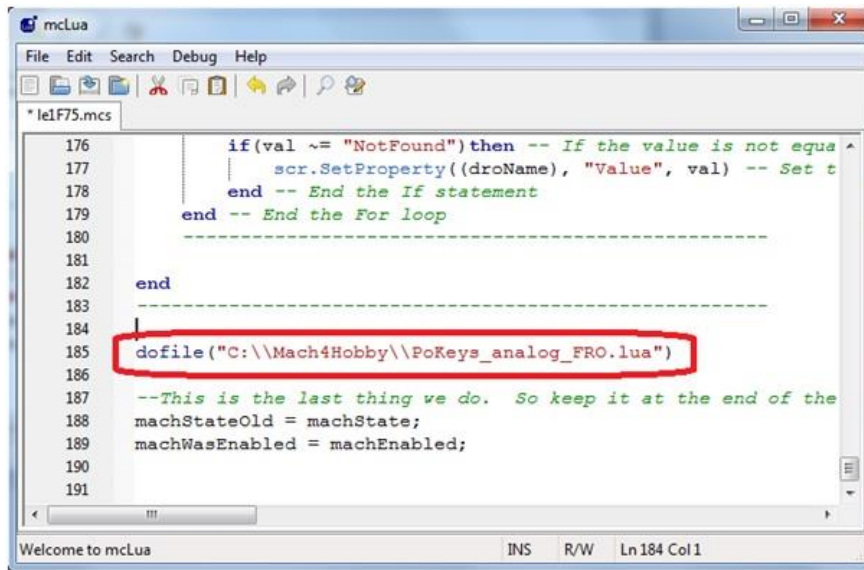
Once done save the file and close it. Now we must add one line to the Mach4 PLC script. Click on the “Operator” menu and select edit screen. Mach4 screen should change a bit. Now there will be 2 windows on the left side of the screen. The top one is the Screen Tree Manager. In this window, select the topmost line (1). In the bottom window named Properties, click on the second button as shown in the picture (2). Next find the PLC script option and click on it (3). This will add a button with 3 dots to the right of it. Click this button (4).



When you click this button, the LUA editor with the PLC script will open. Scroll to the bottom of the script and almost at the end add this line:

```
dofile("C:\\Mach4Hobby\\PoKeys_analog_FRO.lua")
```

## PoKeys plugin for Mach4 - PoKeys device setup



```
176         if(val ~= "NotFound") then -- If the value is not equal
177             scr.SetProperty((droName), "Value", val) -- Set t
178         end -- End the If statement
179     end -- End the For loop
180     -----
181
182 end
183 -----
184
185 dofile("C:\\Mach4Hobby\\PoKeys_analog_FRO.lua")
186
187 --This is the last thing we do. So keep it at the end of the
188 machStateOld = machState;
189 machWasEnabled = machEnabled;
190
191
```

If your path to the PoKeys\_analog\_FRO.lua file is different you must change the path between " ". Remember to use double \\. Now save the PLL script. You can either go to File->Save or click Ctrl+S. You can now close the LUA editor. Now in Mach 4 again click on the Operator and then the Edit screen. This will stop the editing. When asked save the current screen.

If you have followed this Mach4 tutorial correctly you should now see the FRO slider move as you turn your potentiometer. If you want to remove this option just remove the line we have just added in the PLC script and if you want to change the input pin just edit the PoKeys\_analog\_FRO.lua script as shown above.

### Using LCD display registers

If LCD display support is enabled in the plugin settings, the display can easily be accessed using the LUA registers. The registers are available as 'LCD Line x' (x is the LCD line number - 1 to 4) of the PoKeys device object. The following example sets a single line on the LCD display (PoKeys\_45000 is the device name used in this example).

```
local inst = mc.mcGetInstance()
local hreg = mc.mcRegGetHandle(inst, "PoKeys_45000/LCD Line 1")
mc.mcRegSetValueString(hreg, "line of text")
```

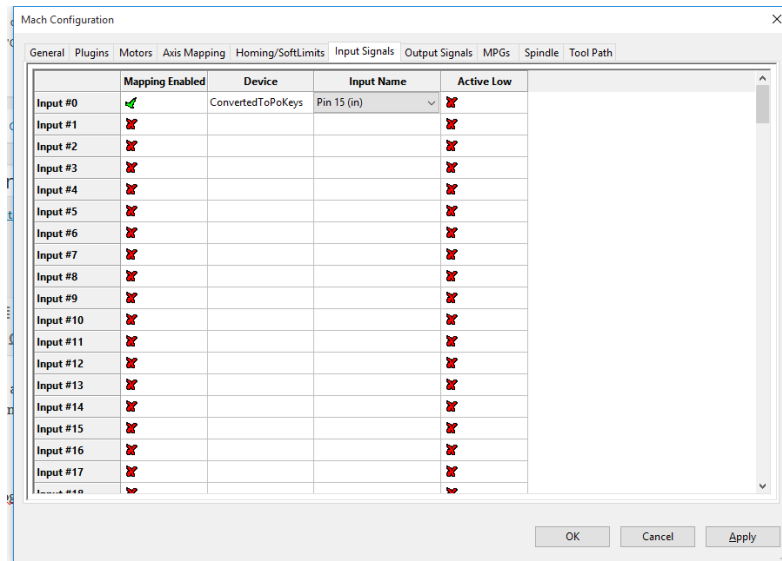
### Custom signal mapping

We will map the signal to one of the PoKeys inputs and configure it to trigger the 'Cycle start' function.

#### Mapping the signal

Open Config > Mach configuration dialog and switch to 'Input signals' tab. Select your PoKeys device and appropriate IO using the drop-down boxes. We selected pin 15 in the following screenshot

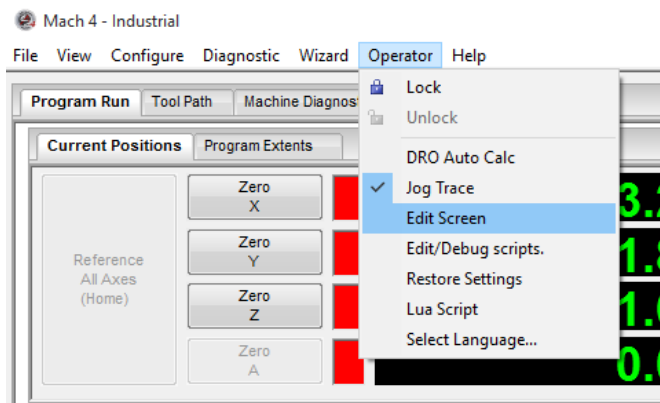
## PoKeys plugin for Mach4 - PoKeys device setup



Confirm the configuration and close the dialog.

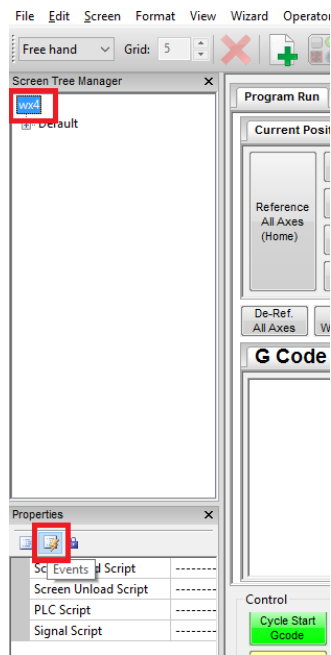
### Opening the signal mapping code

The signal mapping code is a part of the Mach4 screen (window) initialization code, accessible from the screen editor. Go to 'Operator > Edit screen

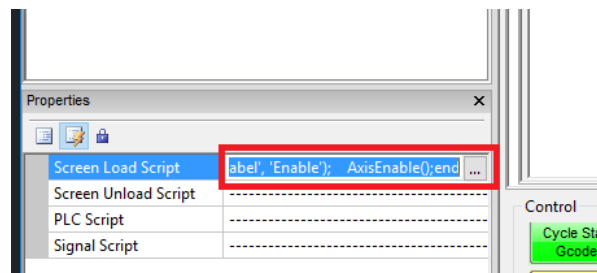


Once the screen editor is opened, one has the access to all parts of the Mach4 GUI parts. For mapping the signal, we will edit the Screen Load script. In the list on the right, locate the top element in the list (also the name of your screen – e.g. wx4, wx6 etc.) and click on it. In the lower left corner, locate the 'Properties' pane. There is a 'Events' button that will open the list of Mach4 scripts that are a part of the screen.

## PoKeys plugin for Mach4 - PoKeys device setup



Locate the 'Screen Load Script' and click in the field on the right. A button (with '...') will appear that will open the script editor when clicked.



### Editing the code

Locate the list of function declarations in the code that look similar to the code highlighted in green and red boxes on the image below.

```
20     scr.setProperty( btnRefAll1 , Enabled , tostring(st
21     scr.SetProperty('btnRefAll2', 'Enabled', tostring(st
22     scr.SetProperty('btnGotoZero', 'Enabled', tostring
23     scr.SetProperty('tabJogging', 'Enabled', tostring(st
24     if (state == 1) then
25         AxisEnable();
26     end
27 end.
28 [mc.ISIG_INPUT0] = function (state)
29 end,
30 [mc.ISIG_INPUT1] = function (state)
31     -- if (state == 1) then
32         CycleStart()
33     -- else
34         --mc.mcCntlFeedHold (0)
35     -- end
36 end,
37 end,
38 [mc.OSIG_JOG_CONT] = function (state)
39     if( state == 1) then
40         --scr.SetProperty('btnToModel', 'label', tostring
41         --scr.SetProperty('btnToModel', 'label', tostring
42         --scr.SetProperty('btnToModel', 'label', tostring
Welcome to mcLua
```

## PoKeys plugin for Mach4 - PoKeys device setup

This part of the code initializes the list of signals and their respective function handlers. In the default screen, ISIG\_INPUT0 signal has an empty function declared, while the ISIG\_INPUT1 signal has a function with some example code that shows how to trigger the function in Mach4.

Let's add the following code to the ISIG\_INPUT0

```
if (state == 1) then
    CycleStart()
end
```

The code now looks

```
25     AxisEnable();
26     end
27 end
28 [mc.ISIG_INPUT0] = function (state)
29     if (state == 1) then
30         CycleStart()
31     end
32 end,
33 [mc.ISIG_INPUT1] = function (state)
34     -- if (state == 1) then
35     --     CycleStart()
36     -- --else
37     --     --mc.mcCntlFeedHold (0)
38     -- end
```

Save the code, then exit the screen editor (again via Operator > Edit screen menu). Save the changes to the screen and test your new shortcut.

### Adding more signals

If you look at the code in the script, you will notice that you have somewhere in the beginning

```
-----
-- Signal Library
-----
SigLib = {
```

Then, one block as following for each signal

```
[mc.<signal name>] = function (state)
    < code >
end
```

So, SigLib is a Table name, which contains handler functions for all signals in the list. The SigLib ends somewhere later on (around row 70) with a '}'. To add a new signal handler, make sure that you insert it in the Table and that you put a comma at the end of it.

```
-----
-- Signal Library
-----
SigLib = {
[mc.SIGNAL1] = function (state)
    < code >
end,
[mc.SIGNAL2] = function (state)
    < code >
end,
```

## PoKeys plugin for Mach4 - PoKeys device setup

```
[mc.SIGNAL3] = function (state)
  < code >
end,
...

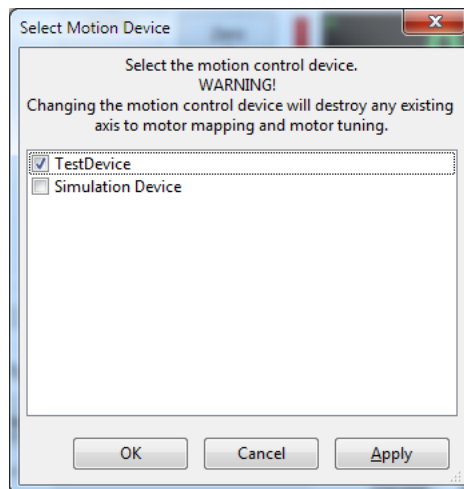
[mc.SIGNALn] = function (state)
  < code >
end
}
```

## Changing the device or moving configuration

The PoKeys plugin configuration allows you to transfer the configuration from one device to another (in the case if PoKeys device needs to be changed). Select the device to change and click on 'Change device'. Follow the wizard to select the new device. When finished, restart Mach4 for the changes to be applied. This operation can also be performed if the original device is not present.

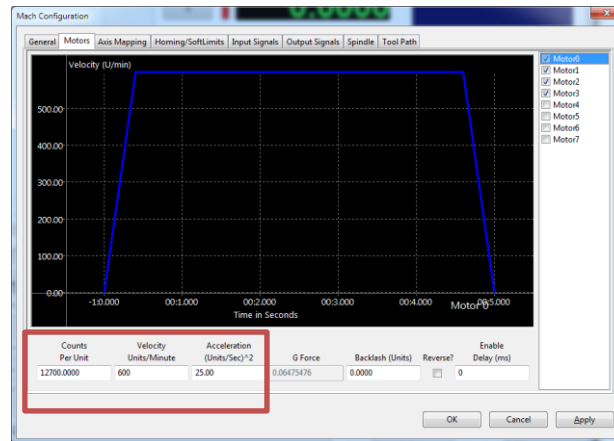
## Configuring PoKeys as motion device

In order to use PoKeys device as motion controller for Mach4, select your PoKeys device in Mach4's dialog: Configure > Select Motion dev. **Restart of Mach4 is advised after changing the motion device.**



Configure motor parameters - open Config > Mach dialog and switch to Motors tab.

## PoKeys plugin for Mach4 - PoKeys device setup



Configure the 'Counts Per Unit' field based on your machine configuration. The 'Velocity' field must obey the PoKeys pulse generator limitation on the maximum step frequency:

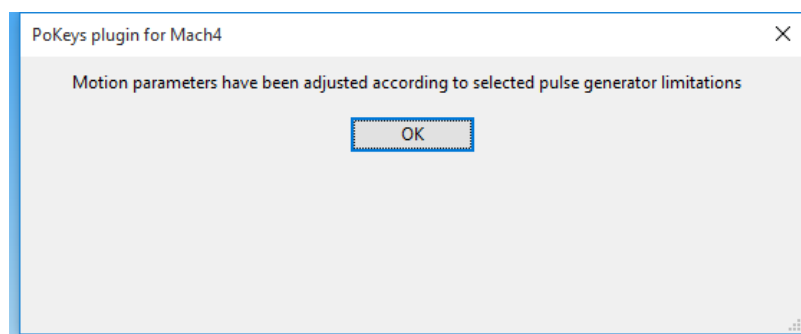
Generator type	Maximum step frequency
Internal 3 channel	25 kHz
External 4/8 channel	125 kHz

The maximum 'Velocity' value can be calculated using the following formula

$$V_{max} = \frac{60 * f_{max}}{N_{unit}}$$

where  $f_{max}$  is the maximum step frequency of the PoKeys pulse engine (see table above) and  $N_{unit}$  is the 'Counts per Unit' of your machine.

Example: let the machine's Counts per Unit be 24000 and external pulse generator is used (PoExtBusOC16CNC, PoKeysCNCaddon, etc.). The maximum velocity is then  $V_{max} = \frac{60 * 125000}{24000} = 312.5 \text{ Units/Minute}$ . **PoKeys plugin will automatically reduce the maximum velocity if it is configured above the limit. In this case, the following dialog will appear when enabling the machine**

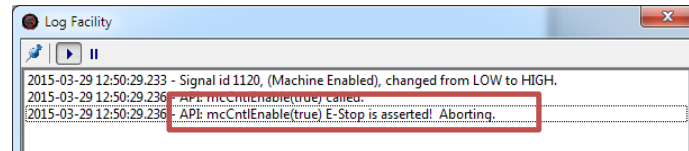


### Troubleshooting:

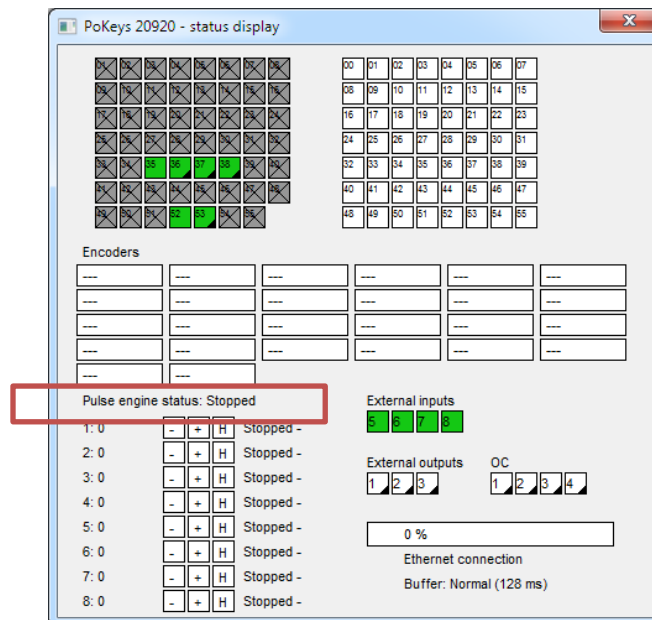
**Problem:** After clicking the 'Enable' button, Mach4 will show 'E-stop condition!' in the status bar and PoKeys device will not start the axis drivers.

**Reason:** E-stop switch is not properly configured.

**Indications:** Mach4 logging screen will show the following information



PoKeys plugin diagnostic screen (accessible via Diagnostics > PoKeys plugin) will list the Pulse engine as 'Stopped'.



**Solution:** Check that you have the emergency switch polarity properly configured. Open the PoKeys device configuration dialog, go to Pulse engine page and check the 'Invert emergency switch polarity'.



### Please read the following notes

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice.
2. PoLabs does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of PoLabs products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of PoLabs or others. PoLabs claims the copyright of, and retains the rights to, all material (software, documents, etc.) contained in this release. You may copy and distribute the entire release in its original state, but must not copy individual items within the release other than for backup purposes.
3. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of the products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. PoLabs assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
4. PoLabs has used reasonable care in preparing the information included in this document, but PoLabs does not warrant that such information is error free. PoLabs assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
5. PoLabs devices may be used in equipment that does not impose a threat to human life in case of the malfunctioning, such as: computer interfaces, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment, and industrial robots.
6. Measures such as fail-safe function and redundant design should be taken to ensure reliability and safety when PoLabs devices are used for or in connection with equipment that requires higher reliability, for example: traffic control systems, anti-disaster systems, anticrime systems, safety equipment, medical equipment not specifically designed for life support, and other similar applications.
7. PoLabs devices shall not be used for or in connection with equipment that requires an extremely high level of reliability and safety, as for example: aircraft systems, aerospace equipment, nuclear reactor control systems, medical equipment or systems for life support (e.g. artificial life support devices or systems), and any other applications or purposes that pose a direct threat to human life.
8. You should use the PoLabs products described in this document within the range specified by PoLabs, especially with respect to the maximum rating, operating supply voltage range and other product characteristics. PoLabs shall have no liability for malfunctions or damages arising out of the use of PoLabs products beyond such specified ranges.
9. Although PoLabs endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, PoLabs products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a PoLabs product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures.
10. Usage: the software in this release is for use only with PoLabs products or with data collected using PoLabs products.
11. Fitness for purpose: no two applications are the same, so PoLabs cannot guarantee that its equipment or software is suitable for a given application. It is therefore the user's responsibility to ensure that the product is suitable for the user's application.
12. Viruses: this software was continuously monitored for viruses during production, however the user is responsible for virus checking the software once it is installed.
13. Upgrades: we provide upgrades, free of charge, from our web site at [www.poscope.com](http://www.poscope.com). We reserve the right to charge for updates or replacements sent out on physical media.
14. Please contact a PoLabs support for details as to environmental matters such as the environmental compatibility of each PoLabs product. Please use PoLabs products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. PoLabs assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
15. Please contact a PoLabs support at [support@poscope.com](mailto:support@poscope.com) if you have any questions regarding the information contained in this document or PoLabs products, or if you have any other inquiries.
16. The licensee agrees to allow access to this software only to persons who have been informed of and agree to abide by these conditions.
17. Trademarks: Windows is a registered trademark of Microsoft Corporation. PoKeys, PoBlocks, PoKeys55, PoKeys56U, PoKeys56E, PoKeys57E, PoKeys57U, PoKeys57CNC, PoKeys57CNCdb25, PoScope, PoLabs and others are internationally registered trademarks.