

# User manual

## JoyPad Plugin for Mach3



## Index

1	Disclaimer .....	1
2	Background .....	2
3	What can it do?.....	2
4	Installing.....	3
4.1	Install JoyPad or game device in Windows .....	3
4.2	Install JoyPad plugin in Mach3.....	5
5	Buttons.....	12
5.1	Button layout.....	12
5.2	Assigning function to buttons .....	13
5.3	Some VBScript code for button macros .....	17
6	Use and copying .....	18
7	Contact and support .....	18

## 1 Disclaimer

Use this plugin AT YOUR OWN RISK!

This software is provided *as is*, free of charge, without any warranty of any type or nature - any use is completely at your own risk.

Limited support only at the Mach support forum:

<http://www.machsupport.com/forum/index.php>

## 2 Background

Mach3 from ArtSoft is very flexible software solution, offering a lot of options for controlling a desktop mill. After some use I found that jogging, probing and general operations was a problem in my configuration – how was it possible to see what the mill was doing and at the same time operating the keyboard, mouse and monitoring the screen. Physically it was possible but not practical.

A Joystick could solve the problem, but it was limited to only controlling two axes (X and Y), missing the third axe.

What I was looking for was:

- Jogging of at least 3 axes (X, Y and Z).
- Precision jog at low speeds not missing high speeds.
- Lots of programmable buttons.
- Some kind of safety mechanism preventing accidentally operating the machine.
- A low budget solution.

I found that the cheapest solution would be some kind of USB game pad. Logitech Dual Action JoyPad had the features I was looking for and at a low prize. Great – now I only have to find the Mach3 plugin for the JoyPad... No luck...

As a software developer I decided to write my own plugin for the JoyPad. C++ is not my preferred programming language, but it could be fun and an opportunity to brush up my C++ programming skills. As said so done...

The JoyPad plugin is written with using Mach3 SDK version 2.62 in native C++, Visual Studio .NET 2008 and relies only on Windows build in support for game devices (MS DirectX DirectInput).

## 3 What can it do?

The JoyPad plugin supports most DirectInput (Microsoft DirectX) game devices with joysticks and buttons. The game device can be connected via the game port, USB or wireless. I have only tested with the Dual Action JoyPad from Logitech but I know other game devices will work as well.

In short the JoyPad plugin supports:

- Control of up till 6 analog/digital axes.
- Control of up till 16 buttons.
- Control of one Point of View (POV or HAT).
- Button function is programmed as VBScript (Marc3 macros).
- Each axe can be scaled, inverted and mapped to a Mach3 axe.
- Precision jogging is obtained with third order scaling of the absolute stick position.
- Single or simultaneous multi axe jogging.
- Besides Mach3 and Windows there is no need for extra software.
- Works with Mach3 Parallel Port.
- Works with SmoothStepper version 1.2.

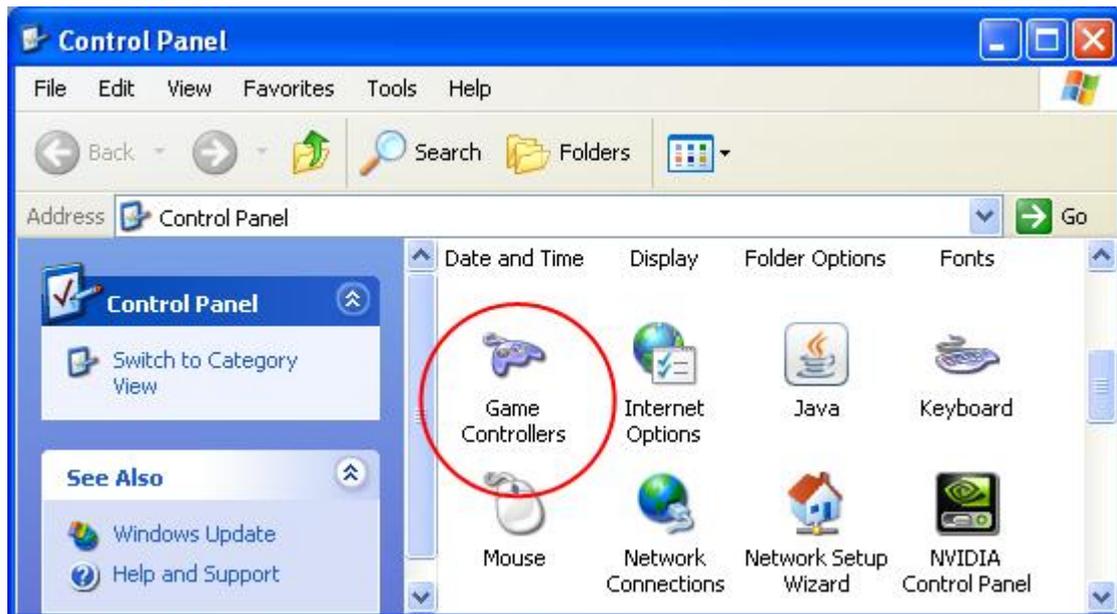
Tested on Windows XP SP3 but I assume that all newer Windows versions will work as long as Mach3 supports them.

## 4 Installing

### 4.1 Install JoyPad or game device in Windows

First make sure the game device you intend to use is recognized by Windows. Install the game device in Windows (may or may not need drivers from supplier).

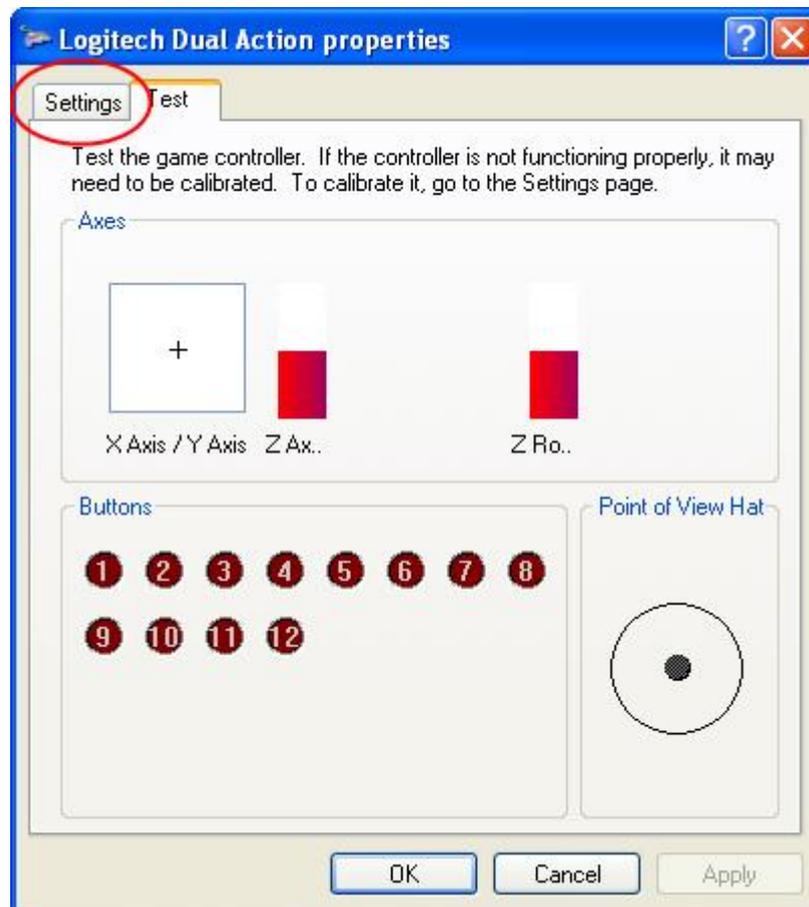
In Windows open the Control Panel:



Select *Game Controllers*:

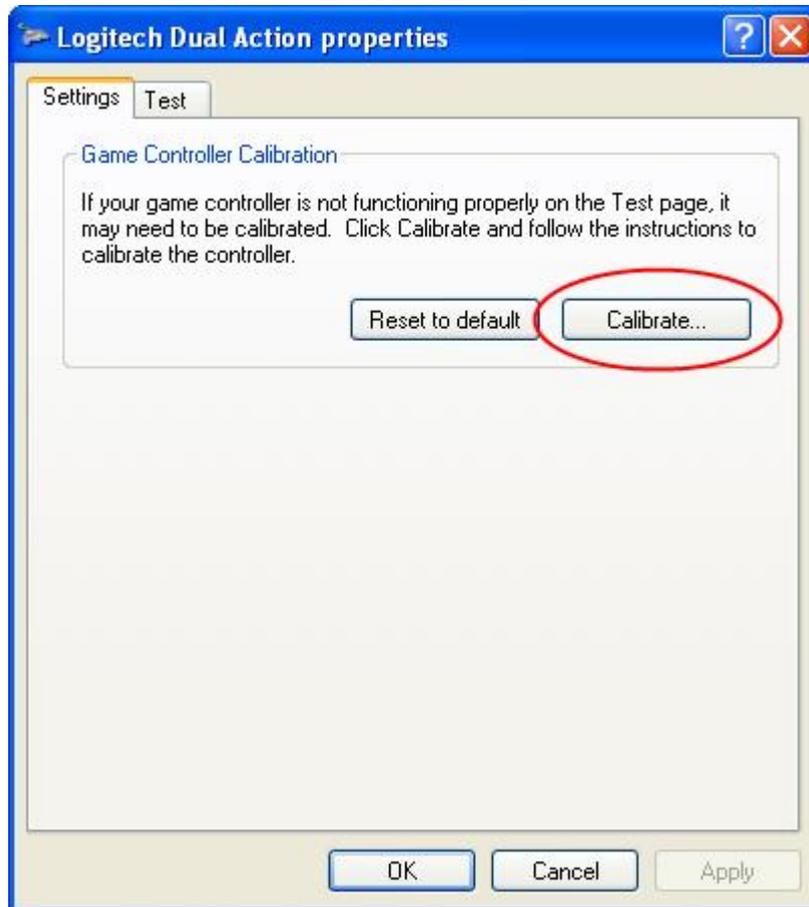


Select *properties* for your game device (in this case the Logitech Dual Action):



Verify that the game device is operating as expected.

Most game devices need calibrating before usage, select the *Settings* tab:



Select *Calibrate...* and follow the directions to calibrate your game device.

The calibration process finds and sets the zero, min and max point for the joysticks, avoiding *floating* when the joysticks are centered.

We are now ready for installing the JoyPad plugin in Mach3.

## **4.2 Install JoyPad plugin in Mach3**

Download latest JoyPad plugin (version 1.0 or later). It is a single file with file extension .m3p containing all needed code.

Double click the file and it will self-install in Mach3 plugin directory.

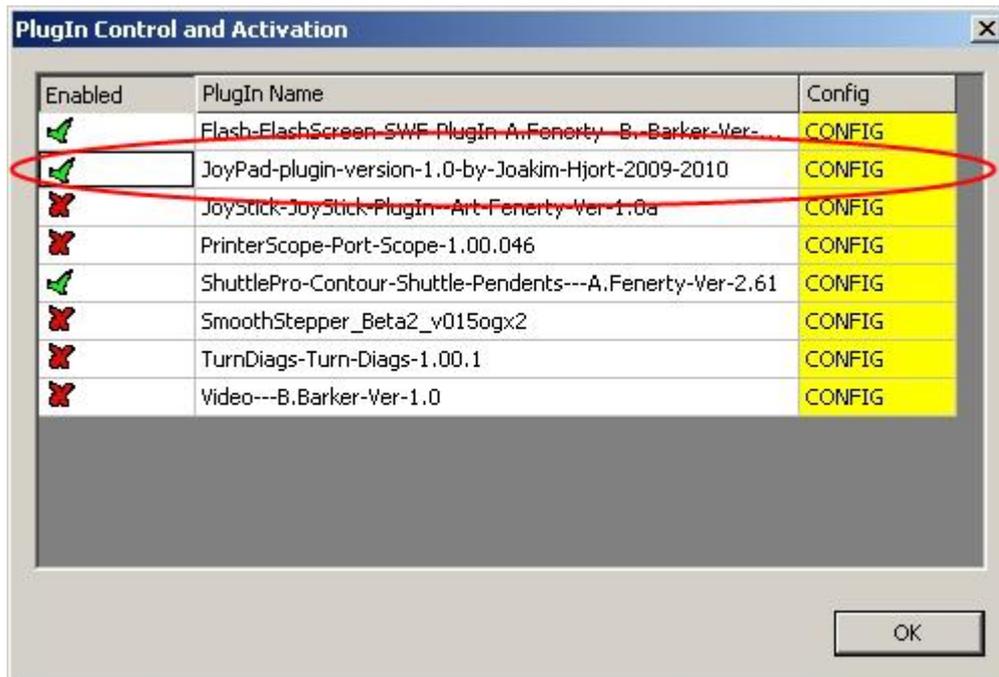
Start Mach3 program and select the *Config* menu:



Select *Config Plugins* submenu:



This will bring up a list of installed plugins. Locate the JoyPad plugin and enable it (marked as green V):

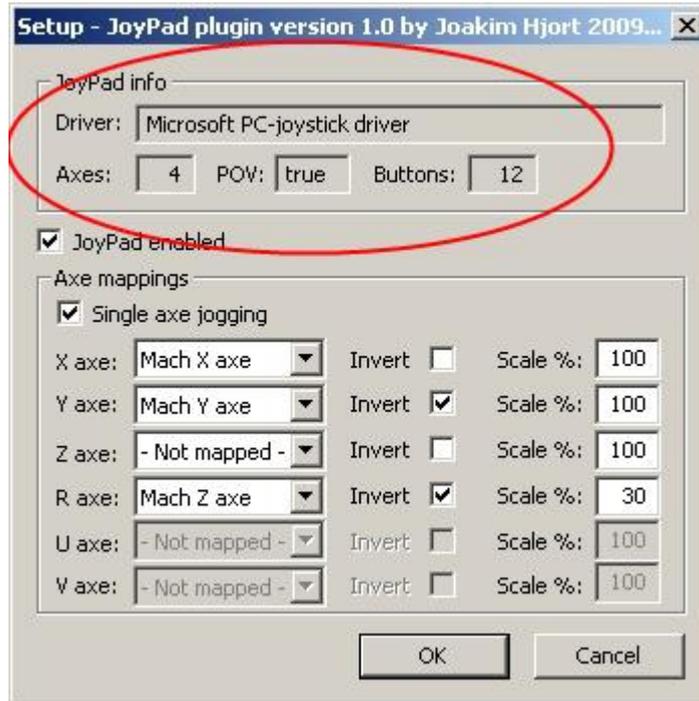


Select *CONFIG* to open JoyPad configuration:

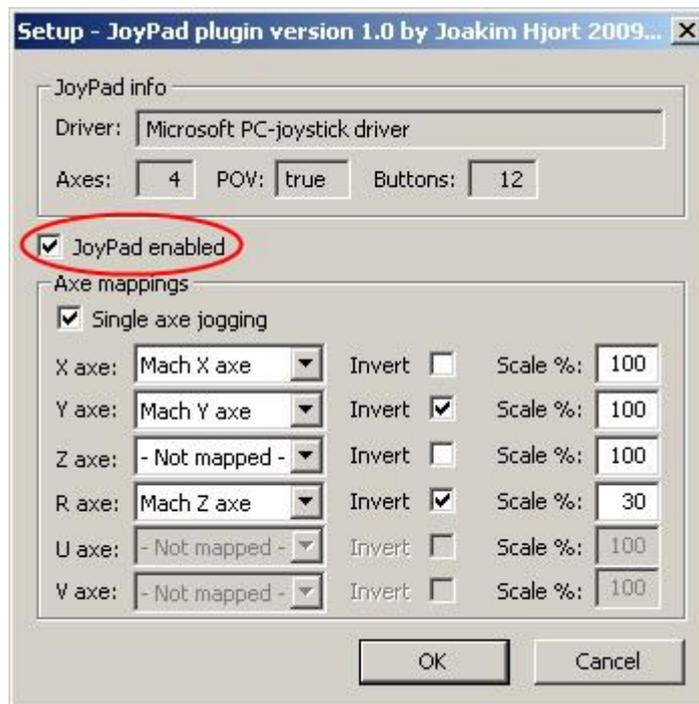


If you get this dialog the game device is not recognized. Please follow directions in previous chapter *Install JoyPad or game device in Windows* or you forgot to plug the device in.

This dialog should appear:

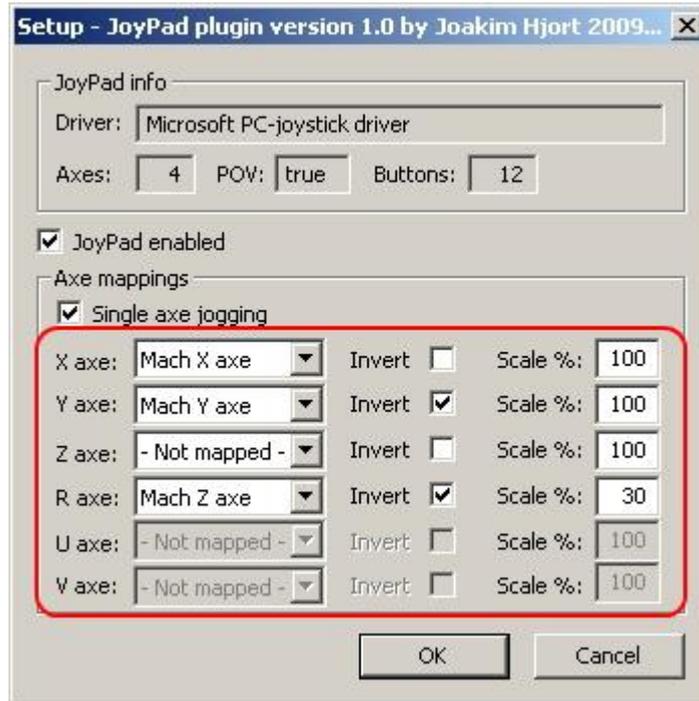


The top part of *JoyPad info* tells you the driver name, the number of axes, if a POV is present and last the number of buttons detected.



Next you have the opportunity to enable/disable the plugin. For the device to function this field must be checked as shown.

Up to 6 axes can be mapped to 6 Mach3 axes:



In this example the JoyPad X-axis is mapped to Mach3 X-axis, not inverted (direction not reversed) and max velocity is 100% of max speed for given Mach3 axis (configured in Mach3).

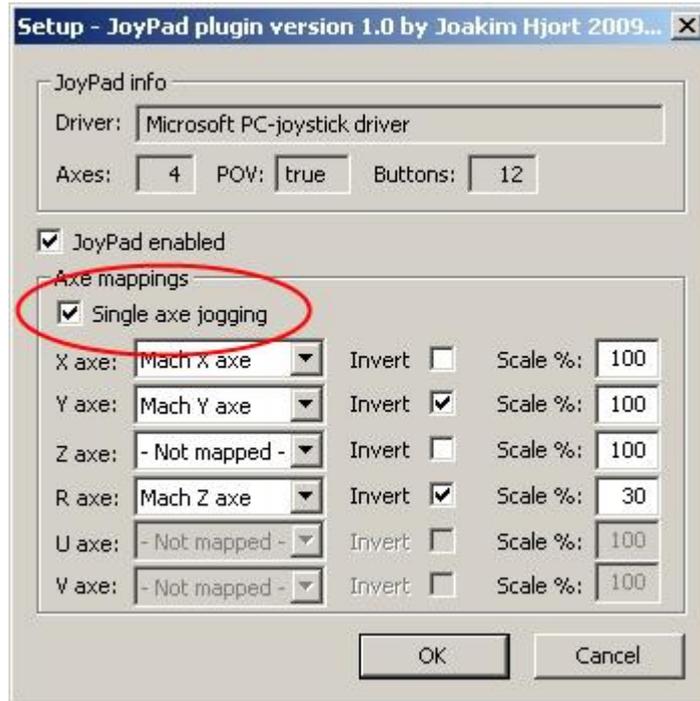
The same applies to JoyPad Y-axis, but here the direction is inverted (reversed).

JoyPad Z-axis is not mapped and is therefore ignored by the plugin.

JoyPad R-axis is mapped to Mach3 Z-axis, inverted and the speed is scaled to 30% of max velocity. This makes it possible to control a short fast axis as by reducing max speed - in this case the Z-axis.

The Logitech Dual Action JoyPad only have 4 axes resulting in the last two mappings grayed out.

It is possible to select two types of jog action: Single- or multi-axis.



Checking *Single axe jogging* will limit the jog to only one axe at a time. This is the recommended setting.

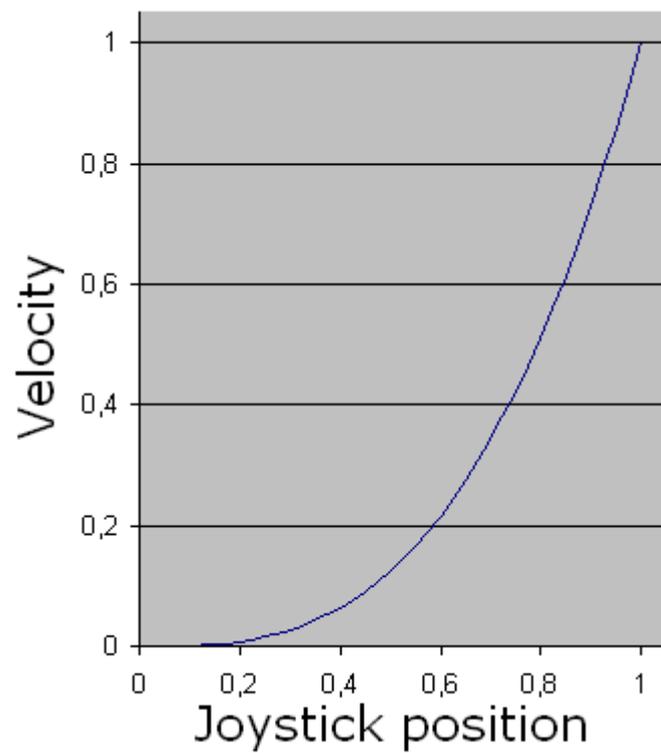
The selection of jog axe is done automatically and reselection can only be done when the stick is at centered. This may sound complicated but in real life it feels natural and intuitive.

Press *OK* to save the configuration to Mach3 xml file. Note Mach3 only physically saves its configuration on program exit – until then the configuration is only saved in Mach3 memory.

That is all! The JoyPad is configured for jogging!

**Note:** Before jogging around you need to know two things:

- **Motion will first occur when button 5 or 6 is pressed.**  
This is a safety percussion to avoid unintentional movement if the game device is accidentally activated or dropped. Only one of the buttons has to be pressed in conjunction with the joysticks. This allows for single handed operation of the game device.
- **Motion is scaled by a third order function to give precision and high speed moves.**



The graph shows joystick position and the resulting speed. Near the center the joystick gives slow speeds and high precision - At full stroke it will give full speed.

This makes it easy to make precision moves without the need of overrides - like the difference between driving a car with automatic gears versus manual gears.

## 5 Buttons

The JoyPad plugin support up to 16 buttons all of which (except button 5 and 6) can be programmed to any function supported by Mach3 built in macro scripting.

Button 5 and 6 is reserved as safety buttons but gives all other buttons two functions. One function when button 5 or 6 is pressed and one when not.

This gives up to 28 button functions on a game device with 16 buttons and 20 on a standard gamepad.

In addition to normal buttons the POV (Point Of View) has 8 buttons (N, NE, E, SE, S, SW, W and NW) with two functions each – a total of 16 button functions.

### 5.1 Button layout

As an example the buttons on a Logitech JoyPad is organized as illustrated:





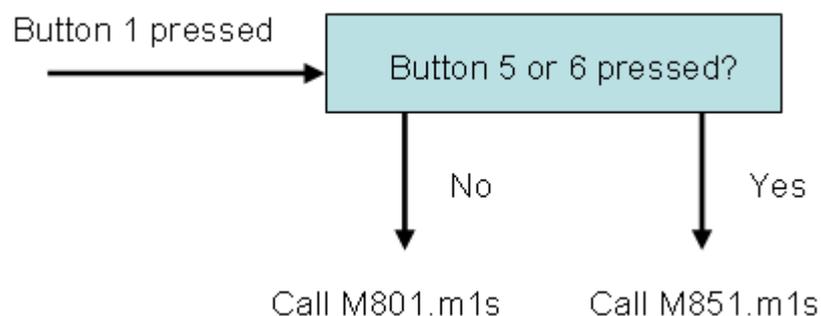
Button 5 and 6 is reserved as safety buttons and prevents the joysticks from operating when not pressed – this can prevent unintended movements of the machine in case the gamepad is dropped or operated by mistake.

In the case of button function assignment button 5 and 6 is used to double the functionality on the rest of buttons. This gives the opportunity to organize safety critical and more harmless functions to the two levels.

As an example button 1 could be assigned to zero the X-axis (relative harmless) and button 1 with button 5 or 6 pressed be assigned to move the machine X-axis to zero (potential harmful).

## 5.2 Assigning function to buttons

JoyPad plugin checks if a button is pressed approx. 10 times a second. When a button is pressed the plugin then checks if button 5 or 6 is pressed too and calls one of two macros as illustrated for button 1:



If only button 1 is pressed the macro named *M801.m1s* is executed. If button 1 is pressed at the same time as button 5 or 6 is pressed the macro named *M851.m1s* is called.

The macros are standard Mach3 macros containing VBScript and located where Mach3 places macros for the current active configuration (see Mach3 documentation).

As an example the path to macros for the configuration *Mach3Mill* is default placed in:

C:\Mach3\macros\Mach3Mill

The exact location (path) will depend on where Mach3 is installed and the name of the active configuration.

The following table is a complete mapping of buttons to macros:

<b>Button</b>	<b>Macro name</b>
1	M801.m1s
2	M802.m1s
3	M803.m1s
4	M804.m1s
7	M807.m1s
8	M808.m1s
9	M809.m1s
10	M810.m1s
11	M811.m1s
12	M812.m1s
13	M813.m1s
14	M814.m1s
15	M815.m1s
16	M816.m1s

<b>Button</b>	<b>Macro name, (Button 5 or 6 pressed)</b>
1	M851.m1s
2	M852.m1s
3	M853.m1s
4	M854.m1s
7	M857.m1s
8	M858.m1s
9	M859.m1s
10	M860.m1s
11	M861.m1s
12	M862.m1s
13	M863.m1s
14	M864.m1s
15	M865.m1s
16	M866.m1s

<b>POV:</b>	
<b>Button</b>	<b>Macro name</b>
1	M821.m1s
2	M822.m1s
3	M823.m1s
4	M824.m1s
5	M825.m1s
6	M826.m1s
7	M827.m1s
8	M828.m1s

<b>POV</b>	
<b>Button</b>	<b>Macro name, (Button 5 or 6 pressed)</b>
1	M871.m1s
2	M872.m1s
3	M873.m1s

- 4 M874.m1s
- 5 M875.m1s
- 6 M876.m1s
- 7 M877.m1s
- 8 M878.m1s

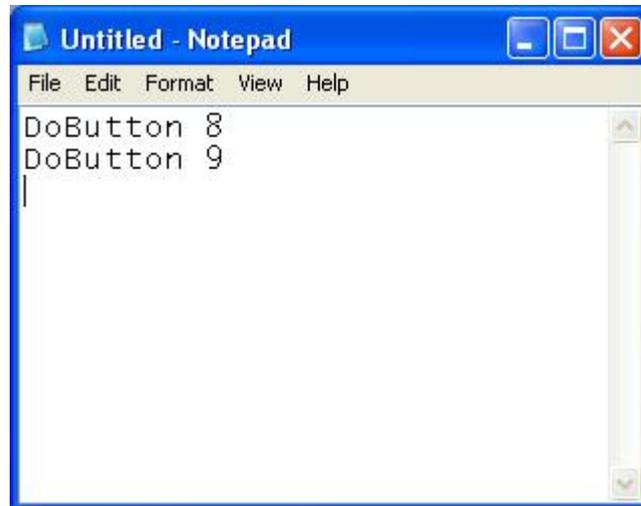


All macros can be executed in Mach3 as M-Codes, the macro M801.m1s will be executed if you type M801 in the command line or place the command in a G-Code file.

A macro is simply VBScript created and written in a text editor as Windows Notepad and saved with the macro name given in the table.

A handy macro could be to zero X- and Y-axis when you press button 11 (joystick button for X- and Y-axis).

To make that macro open Notepad and write the VBScript two commands:



and save it as *M811.m1s* in Mach3 macro directory for the current configuration:



Remember to set *Save as type* to *All Files* (if not Notepad will append .txt to the filename).

To test the macro you can try pressing joystick button 11 on the JoyPad in Mach3 or executes M-Code *M811* in the screen 2 command line.

### How does it work?

*DoButton 8* and *DoButton 9* calls the two buttons numbered 8 and 9 on Mach3 screen performing the same function as if you have pushed the buttons with the mouse. Button 8 and 9 zeros X- and Y-axis.

It is beyond the scope of this manual to cover macro writing in Mach3 but a lot of help can be found in Mach support forum and wiki:

<http://www.machsupport.com/forum/index.php>

[http://www.machsupport.com/MachCustomizeWiki/index.php?title=Main\\_Page](http://www.machsupport.com/MachCustomizeWiki/index.php?title=Main_Page)

Tutorial videos are found at:

<http://www.machsupport.com/videos/>

## 5.3 Some VBScript code for button macros

In the following table are some VBScript macro commands with description.

DoButton 8	Zero X-axis value
DoButton 9	Zero Y-axis value
DoButton 10	Zero Z-axis value
DoButton 11	Zero A-axis value
DoButton 12	Zero B-axis value
DoButton 13	Zero C-axis value
DoButton 17	Goto zero (all axes)
DoButton 22	Home X
DoButton 23	Home Y
DoButton 24	Home Z
DoButton 25	Home A
DoButton 26	Home B
DoButton 27	Home C
GoSafeZ	Goto safe Z
Code("G00 X0")	Rapid move X to 0
Code("M7")	Mist cool on
Code("G28")	Return machine home
Code("G31")	Straight probe
Code("M999")	Call macro M999.m1s
Code("G01 X0 Y0") Code("X10") Code("Y10") Code("X0") Code("Y0") Message "10 unit square"	Move in 10 unit square at federate and display message
Message "Hi"	Displays the message Hi
SetOEMDRO 818, (GetOEMDRO 818 * 1.1)	Increase the feed rate by 10%

DoOEMButton 110	Toggle spindle on/off
DoOEMButton 113	Toggle flood on/off
DoOEMButton 114	Toggle mist on/off

These are only examples – only your imagination sets the limit for functionality in macros. It is even possible to write macros performing complex operations as pocket milling or surface milling, including dialogs setting up parameters.

Follow the links given earlier to read more about macro programming in Mach3 with VBScript.

## 6 Use and copying

You may use and copy the JoyPad plugin as long the ReadMe and user manual file is included in the distribution and the software remains free of charge.

Please don't remove my name from the distributions - I have put a lot of time and effort in developing this plugin...

At the time being the source code is closed and not public – I had plans to release the source to the public under an open source license, but can't find the spare time to do it...

## 7 Contact and support

I can be contacted at the Mach3 forum ([www.machsupport.com](http://www.machsupport.com)), user area51.

As the software is provided "as is" there is no support, but try searching the Mach3 forum for support.

If you like the software, find errors or have suggestions to improvements, I would like to hear about it.

No guarantee that it will be implemented or the code corrected - this is a spare time project, but I feel pride in developing stable usable code ☺

**During the development of this plugin I got a lot of help from skilled people at the MachSupport forum – Thanks to you all!**

Hope you found this short user manual useful – happy CNC machining ;-)

Regards,

Joakim Hjort