The License:

About the controller

Step2Mach is ethernet controller for Mach3 software.
It provides following features:
- 4 axis support
- 100 kHz step/dir frequency
- LPT DB25 connector (as it emulates PC parallel port)
- Easy to assemble hardware design (main components are DIP, all passives are SMD – 1206 or bigger)

The ethernet connection provides following features:
- Web interface for configuration
- DHCP client/server support
- NetBIOS support (for accessing device by its name)

Known limitations:
- No slaving support (not yet implemented)
- No threading support for lathes (not yet implemented)
- No spindle support (not yet implemented)

Building the device

The component list is short – you can get everything e.g. as single Digikey order (www.digikey.com)

The most expensive components are CPUs (33FJ64MC802) and SPI ethernet controller (ENC28J60SP). Most likely you can get them as samples from Microchip.

The PCB is dual layer, but you can also make it in simplified single layer version (top layer only). This comes with following limitations:
- The processors have to be programmed in external programmer (only once). This is because programming connectors use bottom layer.
- The leds on RJ45 ethernet connector will not work.

Please note that the PCB design was made in unusual way – with bottom component side (!).
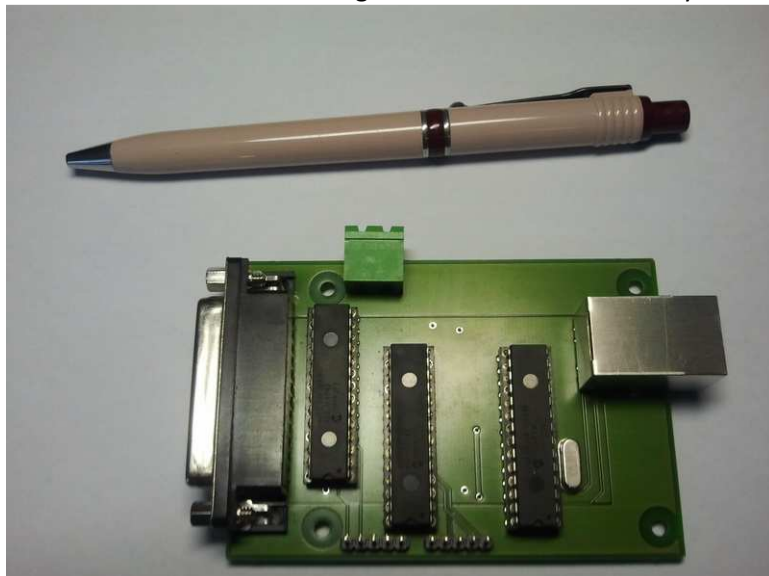
Once you assemble everything you need to program CPUs (note – here bottom view) using PIC
programmer (e.g. ICD2 or ICD3)
- JP1 is for ETH CPU – requires mach_eth\ethernet.hex
- JP2 is for CNC CPU – requires mach_firmware_bootloader\bootloader.hex
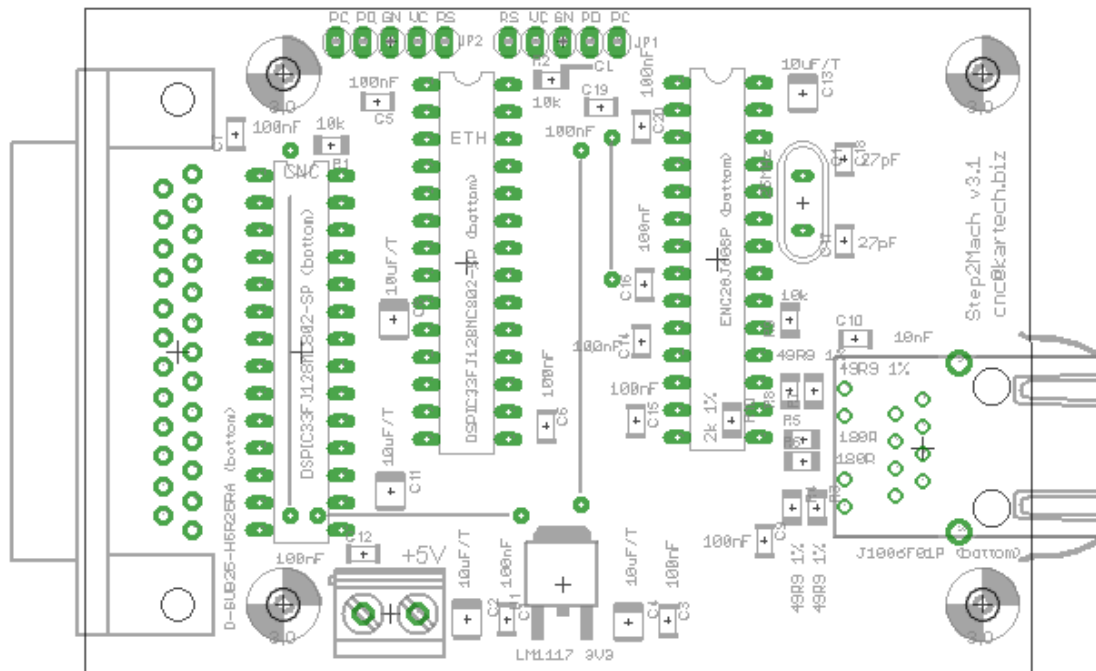
The Mach3 plugin will download target CNC firmware to the CNC CPU.
Note that ETH CPU can be update by programmer only

The ethernet is configured for fixed IP during the start. Just connect it to directly to your network
card (configured for DHCP) and use http://step2mach in your browser
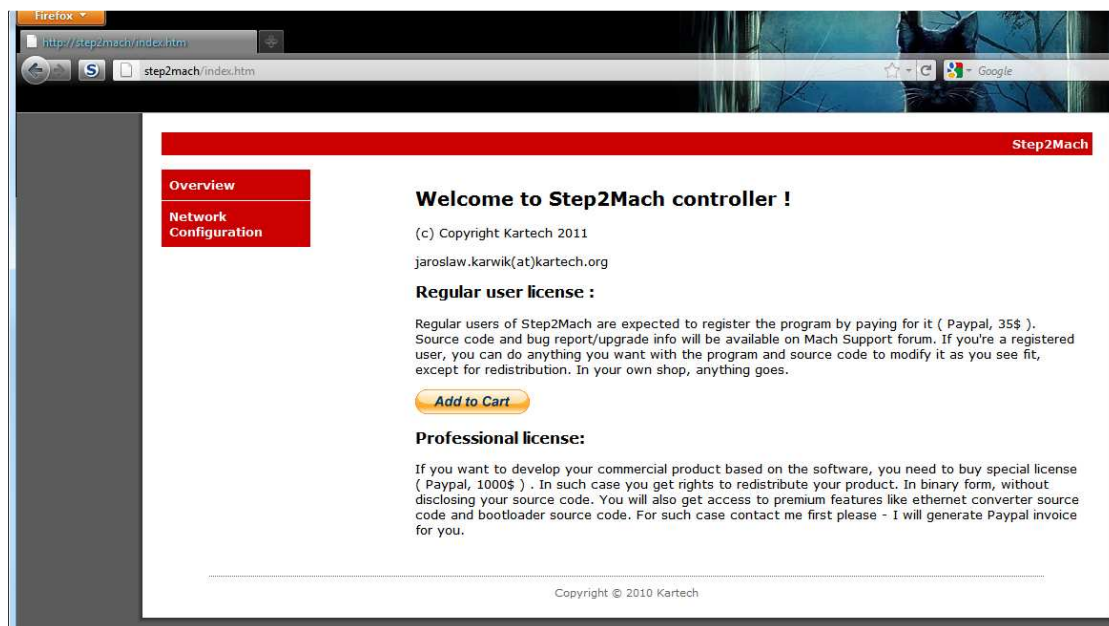
Go to network configuration and make final setup:

Please note:

- Boards do not have unique MAC addresses – should be connected to isolated networks only. If you need unique MAC address you have to buy it (or reuse one from your broken ethernet devices)
- If you mess the configuration you can reset it to default by shorting GN and PD on JP1 (not – not yet implemented – on TODO list)
- Do not use host name for plugin configuration – use IP address directly . The NetBIOS service works quite slowly and you will notice Mach3 being frozen during plugin start and each ESTOP activation (it forces device reconnection, the Windows name resolution service can take few seconds)

Plugin configuration

Copy Step2Mach.dll to Mach plugin directory (C:\Mach3\PlugIns\)

Run Mach3

Activate the plugin – choose OK
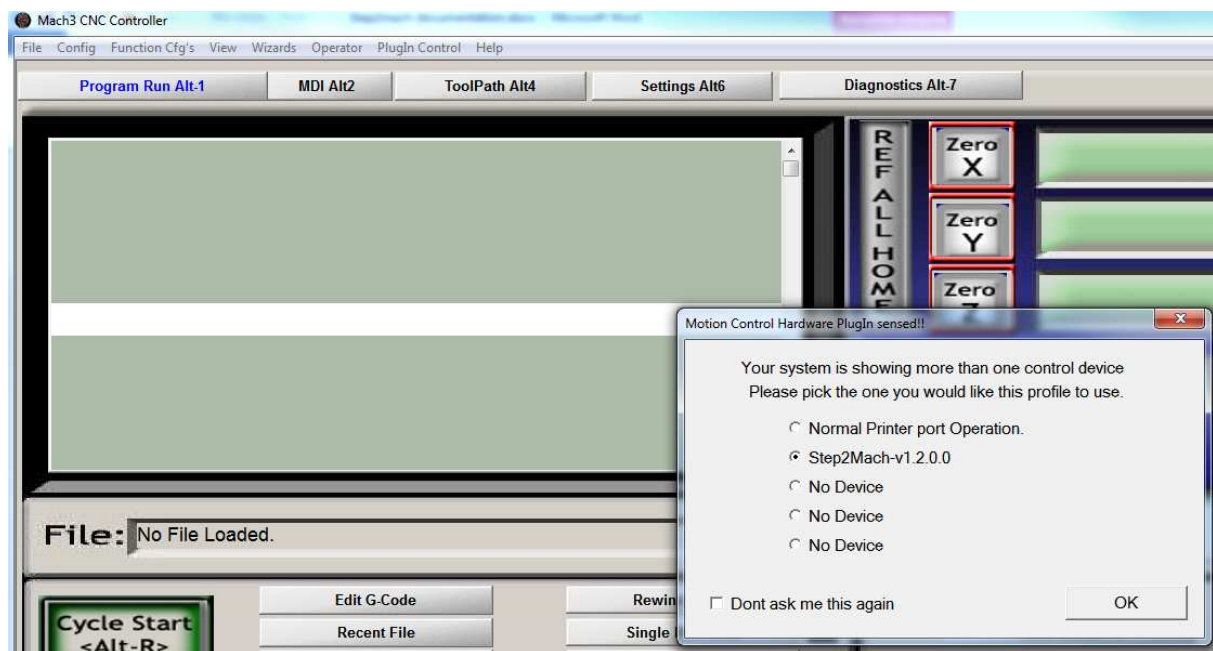
Configure plugin

Note:

- Use IP address, not NetBIOS name
- Debug mode enables additional console window with dump of internal Mach variables. You may have to resize the window to be shown correctly as default 80x25 characters is not big enough

- The licensing system was disabled in the code.

Firmware upgrade

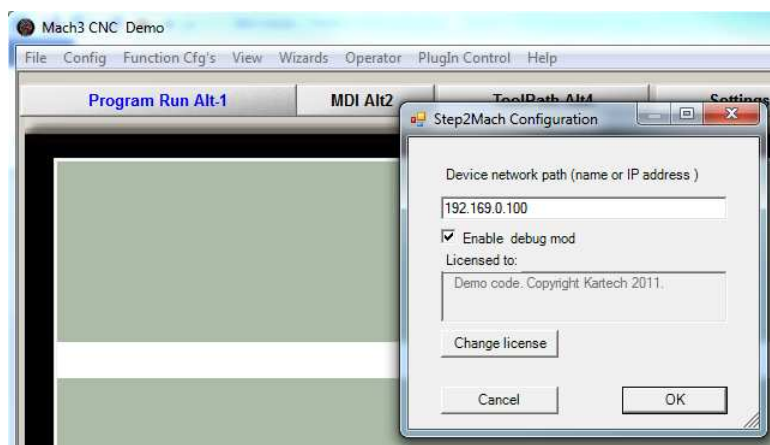Each time plugin connects to the device it checks for firmware version. In case of mismatch (or missing firmware) it will upgrade it. The firmware is embedded (compiled into) the plugin code.

Todo list / known limitations

- Upgrade process needs improving – right now after successful upgrade user needs to press ESTOP again (to force reconnection)
- Better visualization of connection/upgrade process
- Implement slaving support
- Implement threading support for lathes
- Implement spindle support

Source code organisation

Mach_common

Header files reusable between PC plugin and dsPIC firmware

Mach_eth

Firmware for ETH CPU. Modified Microchip TCP/IP stack with support for transferring data from UDP sockets to serial port. Source code only in premium version. Binary version available as ethernet.hex

Mach_firmware

dsPIC firmware for motion controller. Debug version to be used without bootloader (directly), release to be embedded into PC plugin (you need to copy it to mach_plugin\bootloader\fw\ and execute convert.bat to get fw.cpp file with encoded firmare. Change firmware version in convert.bat file if needed).

Mach_firmware_bootloader

Bootloader code for CNC CPU. . Source code only in premium version. Binary version available as bootloader.hex

Mach_plugin

PC plugin code. Note – contains also licensing system which was disabled in the code.

Tools needed for the project

- Visual Studio Express 2010

  http://www.microsoft.com/visualstudio/en-us/products/2010-editions/visual-cpp-express

- Microsoft DDK (for ATL support)

  http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=11800

  Note: You need to add DDK paths for include files and libraries to be able to use ATL

- Mach plugin wizzard

  http://www.j1sys.com/legacy/MachPluginWizardSetup1.1a.msi

  Note: The plugin was originally generated in the wizard. It references some of its header files which describe Mach interface

- Mplab IDE (IDE for developing dsPIC code)

  http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002

- C30 Compiler (for dsPIC)

  http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en536656

  Note: Microchip calls it student version, but it is just GNU compiler. Free also for commercial use – not be fooled by their naming.

- Hardware design - Eagle

  http://www.cadsoftusa.com/