

Puma for FreeEMS
Assembly, configuration, modification and installation
manual

Thursday 21st April, 2011

Contents

1	Overview	2
2	Getting the hardware	2
3	So you have a bare board, now what?	2
3.1	You now have a PUMA board with components, now what?	3
3.2	Testing your board with Jim Stim.	3
3.3	Uploading the initial firmware.	3
4	Uploading firmware via USB and boot loader	4
5	Trying out the board with a Tuning Software	5
6	Connecting the board to a vehicle	5
6.1	The basics	5
6.2	Connector boards	5
6.3	Puma's connections	6
6.4	Grounding	7
7	Troubleshooting	7
8	Spin1 specific notes	8

1 Overview

This document is to show how to choose components to order, properly build and connect your Spin 1 board to a vehicle.

Puma uses FreeEMS vanilla firmware, PC tuning software and Puma hardware to control fuel and ignition of large variety of engine configurations. See <http://puma.freeems.org> for a more detailed overview of what PUMA can do and is used for.

We recommend you purchase the hardware such that many of the below steps aren't required. This version of the manual includes instructions for the Spin 1 board, which was generally provided as just a bare board. You will need to purchase parts based on a BOM then install those parts. You will also need to program the board via BDM programming port to get the initial bootloader installed. After the initial build, you will be able to program the board with an USB cable and a standard PC.

Note these sections are written as modules, such that sections are useful on their own, but also arranged in a sequence that is handy for a first time installer.

2 Getting the hardware

The design and software is OpenSource and while you can technically build it by yourself, we recommend you purchase PUMA assembled.

There are two ways to get an assembled board: sign up on forum.diyefi.org and ask for one, or go to puma.freeems.org and place an order.

Spin 1 was offered as a bare board, semi-assembled board, and a couple were fully assembled, so there are notes about how to do some low level assembly, but future spins will likely remove these assembly notes.

3 So you have a bare board, now what?

Ok, you have the bare PCB, and it's going to need components installed on it to function as a EMS. This is what you need to do, in no particular order:

Get the BOM from https://github.com/nitrousnrg/puma/blob/master/BOMs/order_BOM.xls, select which modules you need to have populated and the get the components.

For digikey customers:

The third page is a digikey formatted file, ready to be saved as .csv to place the order.

For newark/mouser/farnell costumers:

Use the MFG number to place your order, available in the second sheet, named "MFG BOM".

A few things to note:

- EGT circuit won't work, it has a 500°C limit.
- The usb connector is plain wrong. In the BOM you can choose to buy a cable to use this wrong (female-A) connector, or buy another usb connector and hack things to install it.
- You don't want the shutdown circuit, so its FETs aren't in the BOM. Don't worry about that.

Once you are decided about the BOM, go to to digikey, mouser, newark or your favourite distributor and place the order.

IMPORTANT: If needed, modify the PCB as shown in “Spin 1 specific notes”

We recommend doing these steps sequentially.

1. Install MCU and critical components.
2. Test the MCU works by uploading the firmware
3. Install rpm input circuit choose hall input, or VR
4. Install misc outputs like Fuel, ect
5. Install injectors circuits
6. Install ignitions circuits

3.1 You now have a PUMA board with components, now what?

You will likely want to verify the board is good with a Jim Stim then it will need the initial firmware installed. If PUMA was purchased assembled, these steps should not be required, and you can proceed to the next step.

3.2 Testing your board with Jim Stim.

Blah blah blah, fill in this stuff.

3.3 Uploading the initial firmware.

You need:

- A BDM programmer like TBDML, PE micros, etc.
Buy, borrow or build a BDM pod compatible with Freescale S12
This is an interesting one <http://www.flashgenie.net/USBDM.html>
- CodeWarrior IDE or standalone app for code programming
Get the CodeWarrior IDE from (????), or download a small app from
http://sourceforge.net/projects/usbdm/files/Version%204.5/1.%20Installation_4_5/USBDM_Win32_4_5.zip/download, check "HCS12_FlashProgrammer".
- A Serial Monitor binary
You can get the Serial monitor binary from the folowing url:

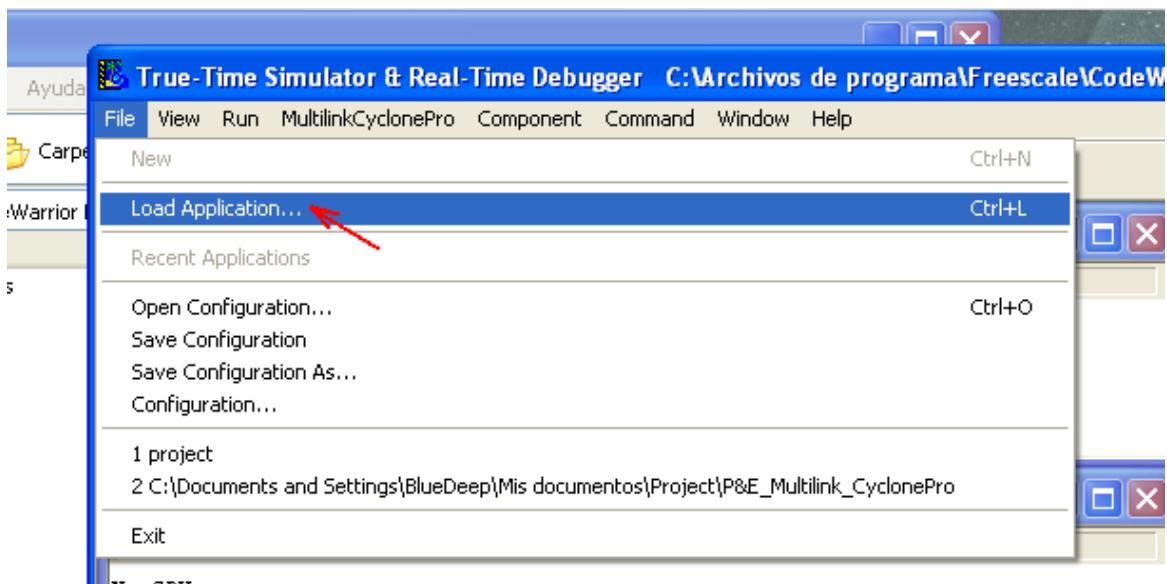
<https://github.com/fredcooke/freeems-vanilla/raw/master/lib/freeems.serial.monitor.s19>

Now proceed to the programming, the order is important:

1. Connect the BDM to the board
2. Connect the USB cable of the BDM to your PC
3. Supply Puma with 12v
4. Open Codewarrior's debugger, or execute directly
C:\Archivos de programa\Freescale\CodeWarrior for HCS12 V4.7\hiwave

The BDM pod should perform some recognition tasks

5. In the debugger, go to File->Load Application and select the .s19 binary.



6. Program the device. Verification should be successful.
7. Poweroff the board, then disconnect the USB, then disconnect the pod

Now you can load the latest FreeEMS firmware!

Install SeanK's loader from <https://github.com/seank/freeems-loader>

Its an easy to use application. Just connect to the USB and power up Puma (jumpering the load/run header), select the device /dev/ttyUSB0 in *nixes (COM?? in windows?), and push the load button. In the file dialog, select the .s19 firmware you want. Wait until it erases and program the device. Close the communications.

Remove the load/run jumper in the header and reset de board.

Congratulations! FreeEMS code is running in your board.

4 Uploading firmware via USB and boot loader

So you have a PUMA that may or may not be connected to your system.

This is how to upload a new firmware file to that board.

- Download and install

- Open
- Configure USB port
- Specify to upload firmware file.

5 Trying out the board with a Tuning Software

Install and open Megatunix (from github link?). Select FreeEMS as the ECU system, and hit "Find my ECU".

6 Connecting the board to a vehicle

6.1 The basics

Figure 1 shows how the PUMA board should be connected to the engine.

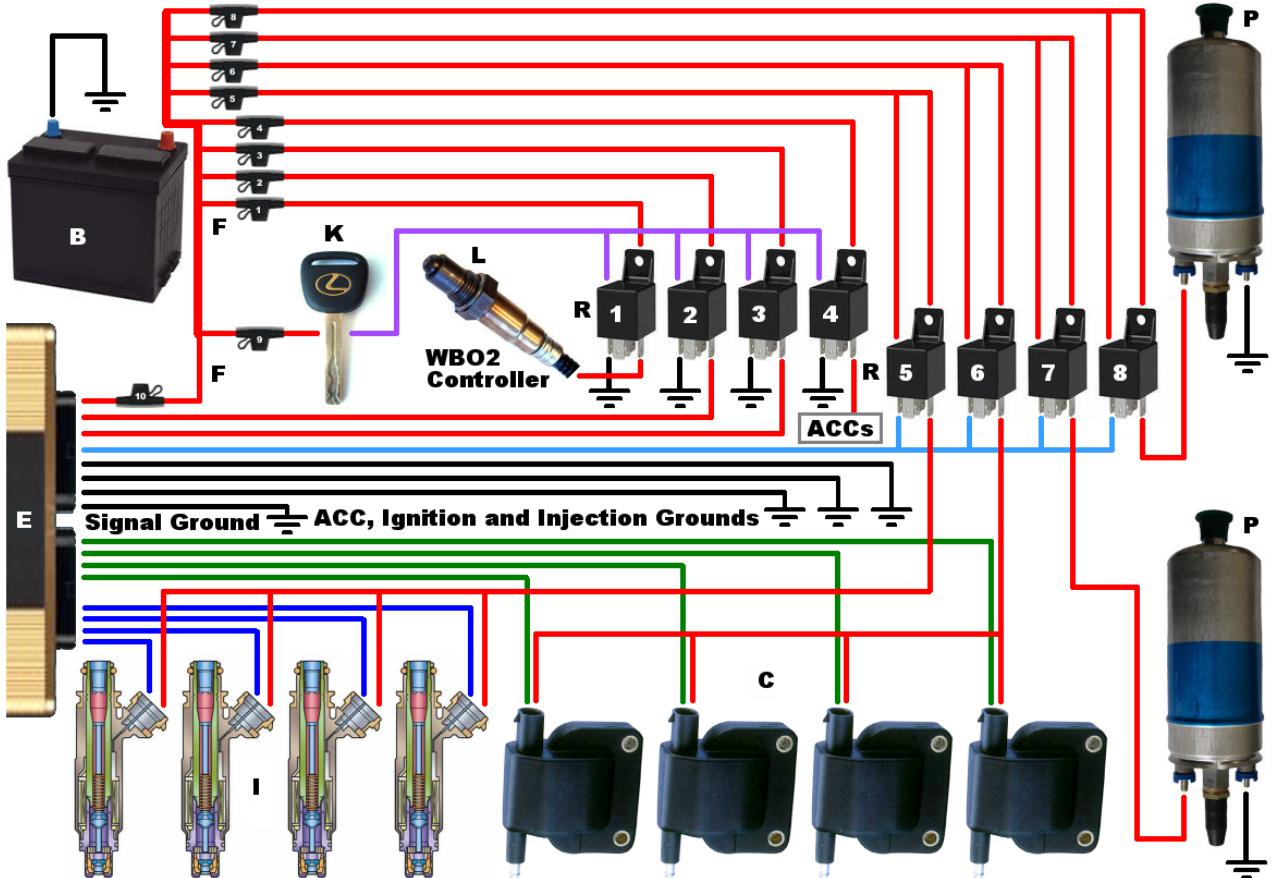
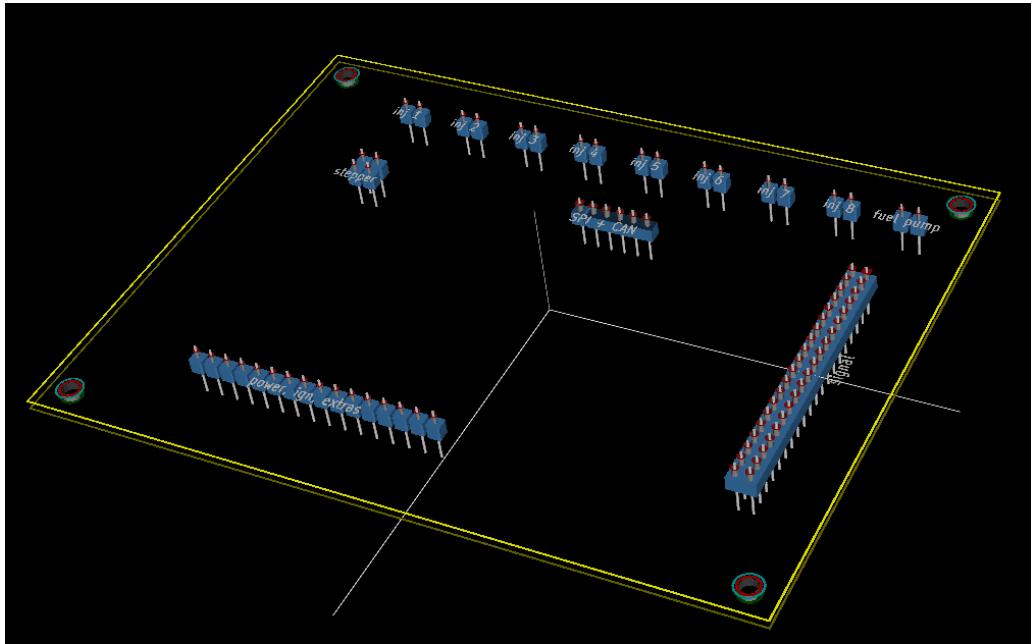


Figure 1: Wiring diagram

6.2 Connector boards

Puma is intended to have another board as an interface to the engine. This allows to have specific connectors and circuitry for some types of engines.

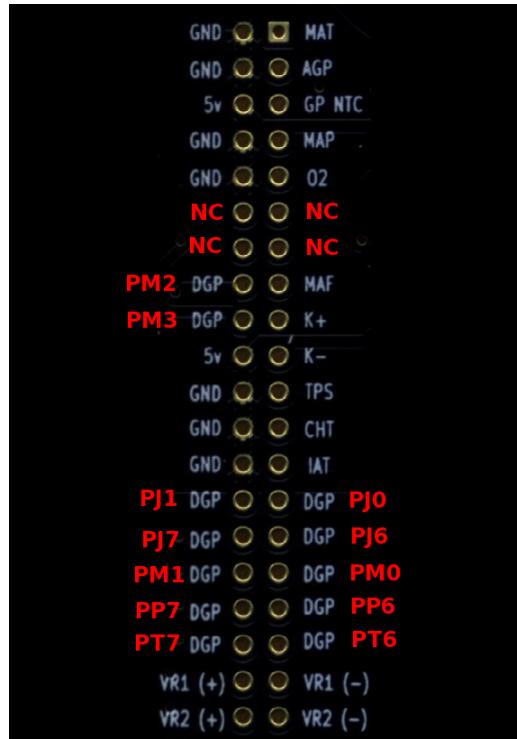
A connector board template for Spin1 would look like this one:



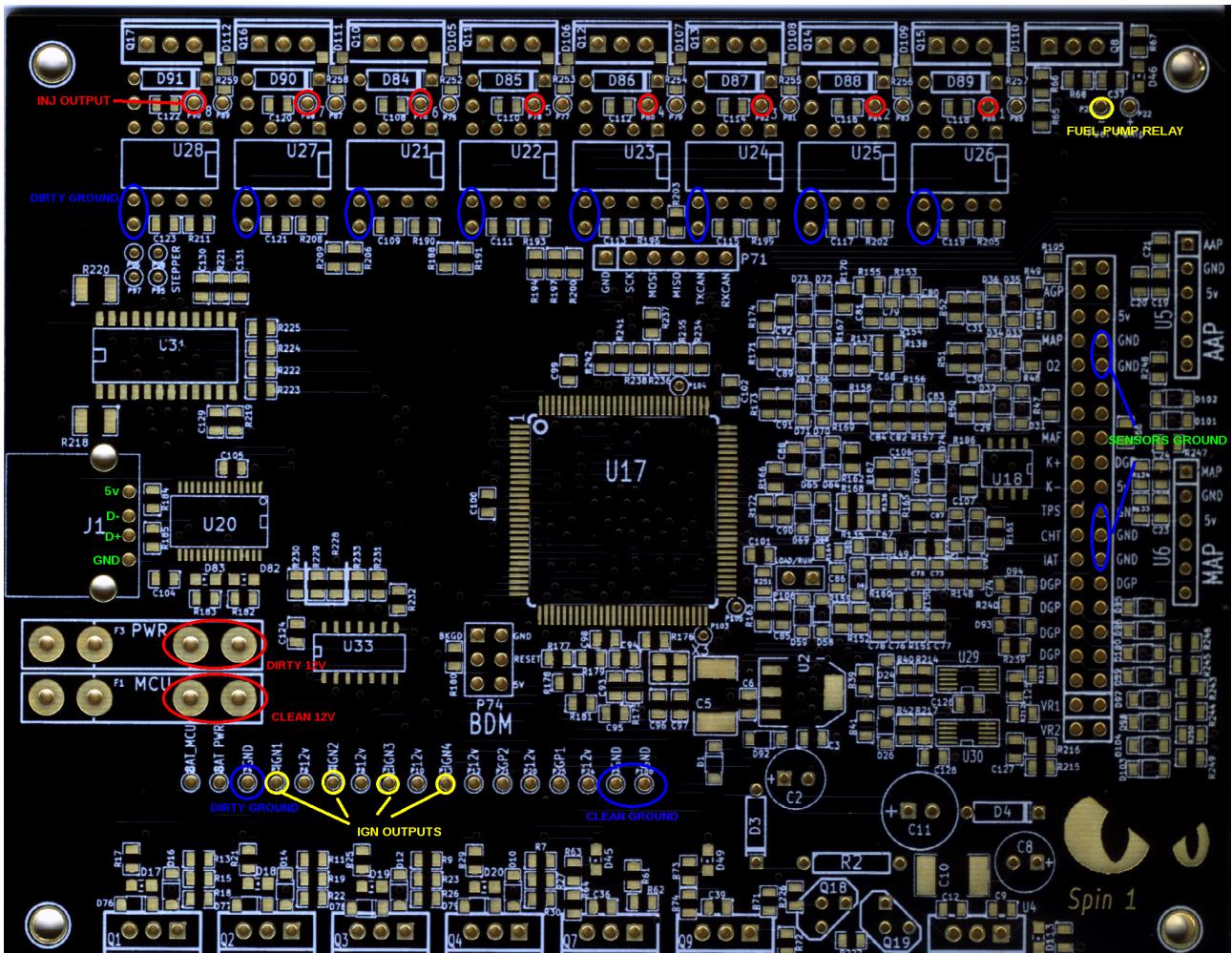
If you don't have a Connector Board, you can design/make/buy your own, or just skip it and connect Puma directly to the engine.

6.3 Puma's connections

In case you want to connect the engine directly, or want to design a connector board, these are the connection layout of a Spin1 board:



For digital crank/cam inputs, leave VR(-) floating and connect the signal wire to VR(+). It will set the threshold to 2.5v.



6.4 Grounding

Connect power grounds as noted in 6.3. This is the single most important part of the assembly

Don't make high currents go through sensors ground, use at least 5(?) ground wires to the battery in your installation.

7 Troubleshooting

You followed exactly what this document said, but things didn't work well. Ok, fear not, let's do some basic checking:

- FreeEMS code is loaded but it doesn't run.

Verify the firmware you loaded: Open Loader->Rip, and diff the .s19 file with the original. The only differences should be in the last part (that is the SM itself)

If it seems right, you are probably in Serial Monitor mode.

The LOAD/RUN header should be left open to allow for FreeEMS code to run. While in SM mode, the header pin is at 0v, and while running FreeEMS, the same pin has a 50hz square wave (with a multimeter you'll measure 2.5v).

The FT232 output (pin 1) must have a strong pullup so the MCU can leave the SM mode (point to USB hack), check it is working.

- Nothing works:

Check PLL and oscillator by measuring the voltage in the MCU-side of R175. It should be around 1.5v. If it is 2.5v (totally pulled up), you have a problem with the crystal or the PLL circuits. Check the solderings or continuity.

- BDM pod throws an error:

Check you have 5v or 2.5v in all the required VDD pins. Check continuity in the BKDM line. Check PLL and crystal.

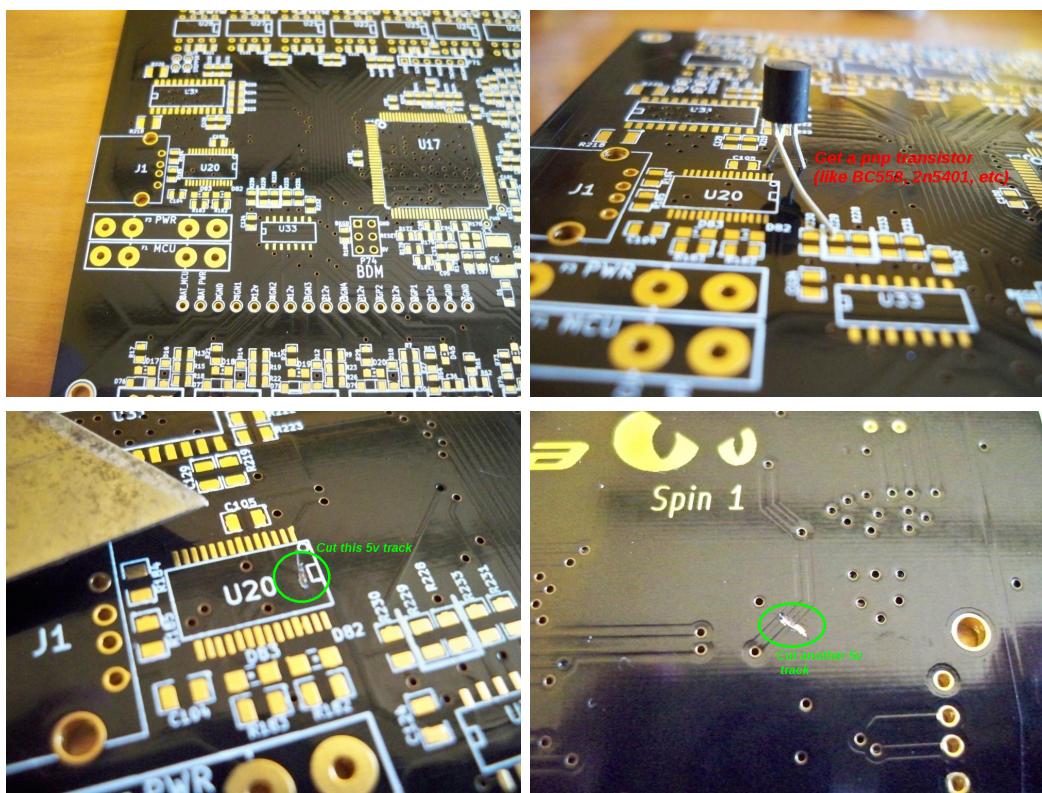
8 Spin1 specific notes

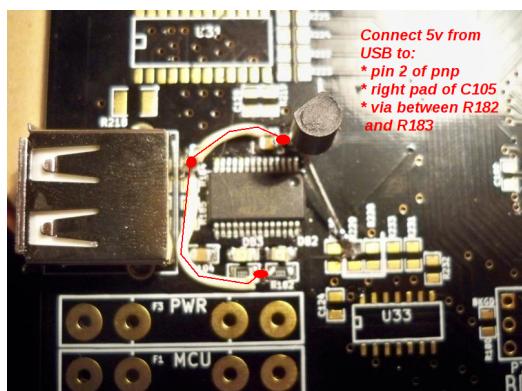
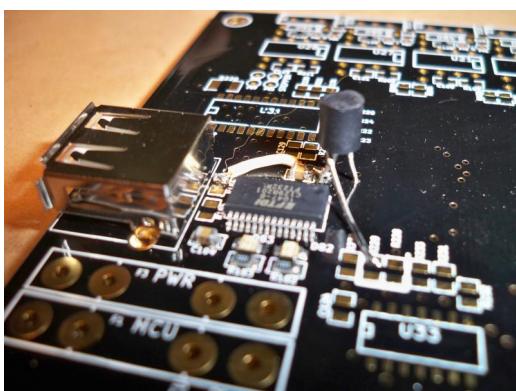
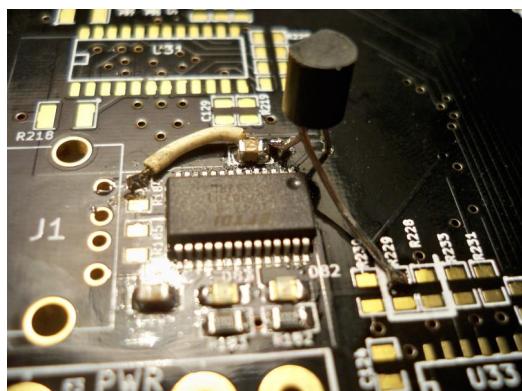
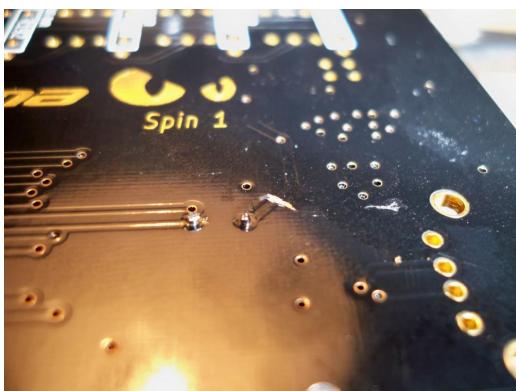
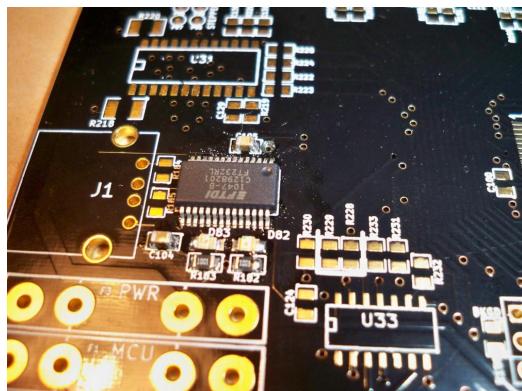
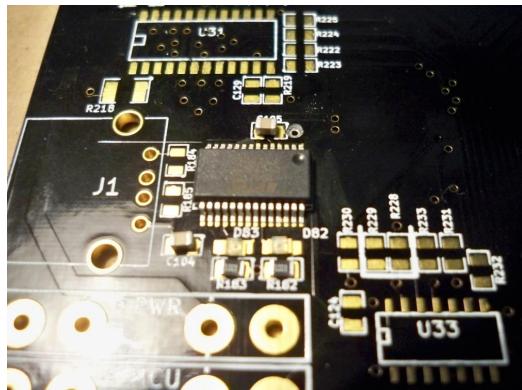
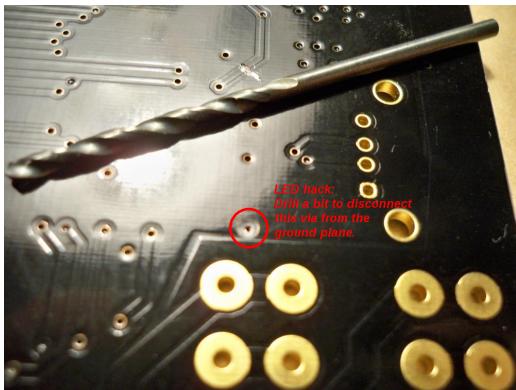
1. This spin has several modifications. Start these modifications by cutting the wrong traces, and adding jumpers as shown below.

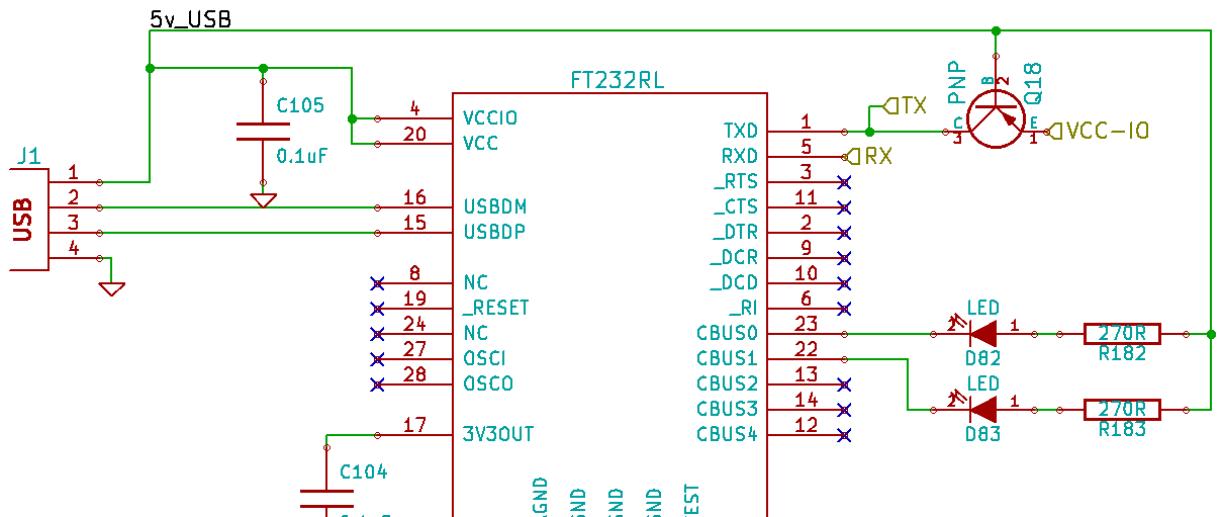
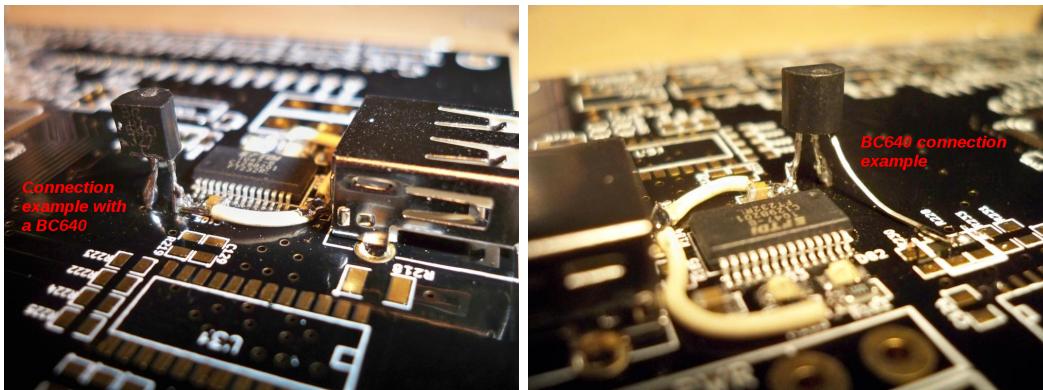
The FT232 IC needs also a pnp transistor or it will make the mcu start always in SM mode.

If you want blinking leds for the usb, there is a pulldown that should be a pullup.

To begin your USB modifications, take reference from the following pictures:







2. You have the PCB + components, so get KICAD and the files, or the pdf files from puma.freeems.org. PDFs aren't search-able so you may want to choose to install KICAD until we workout that.

3. Start the assembly!

Just solder the components. If you ordered a board, you should know how to do it. An oven is a fast way to get it done.

Don't put too much paste for the small regulator, or it will get misaligned.

Components that shouldn't be populated:

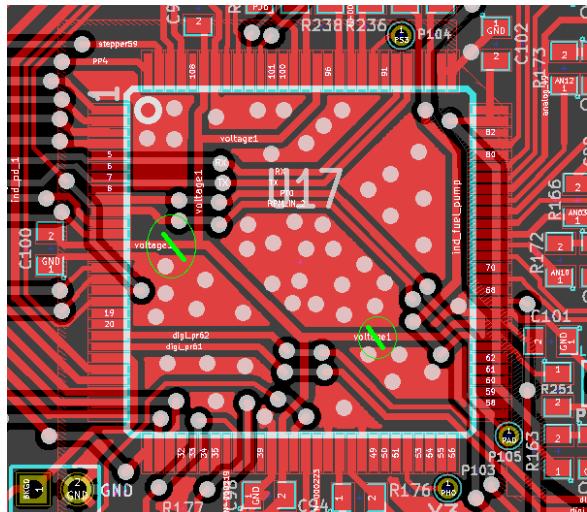
- F1, F3 (Fuses)
- R226, R227, Q18, Q19, bridge pin 1 and 3 of Q19 (this is the shutdown circuit)
- R133 (bad pullup)
- R228 OR R229, using one of them defines whether the XOR negates or not its outputs.
- If you use VR inputs, R212, R213, R215, and R216 should be bigger, like $\frac{1}{4}$ or $\frac{1}{2}$ W. 10k Ω to 20k Ω will be fine.
- U18, R186, R187, C107, D74, D75, C106 (thermocouple driver)

4. Program the MCU using a BDM pod.

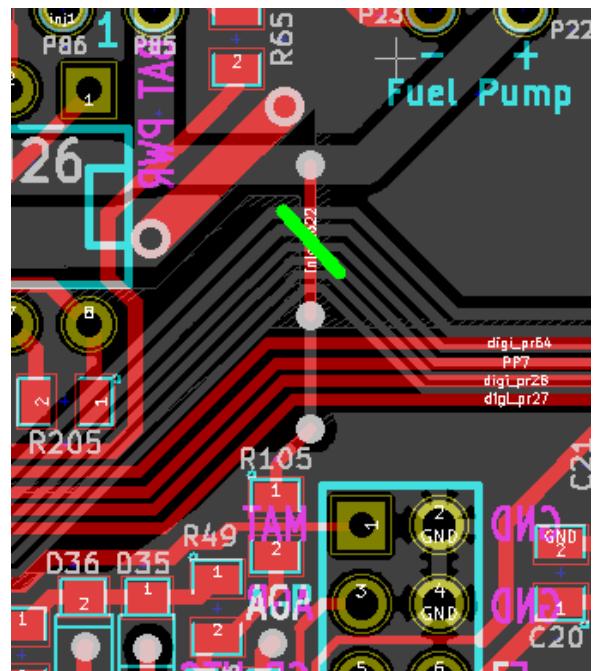
Install Codewarrior, open the programmer, go to File → Load application, and select the .s12 (FreeEMS serial monitor).

It should get connected, program it, verify, and never complain.

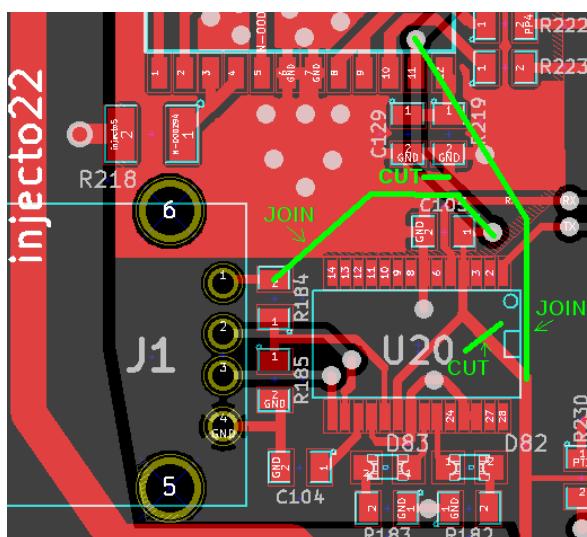
5. Load FreeEMS firmware, using seank's loader.
6. Install MTX and connect to the board to the PC to check that freeems is running.



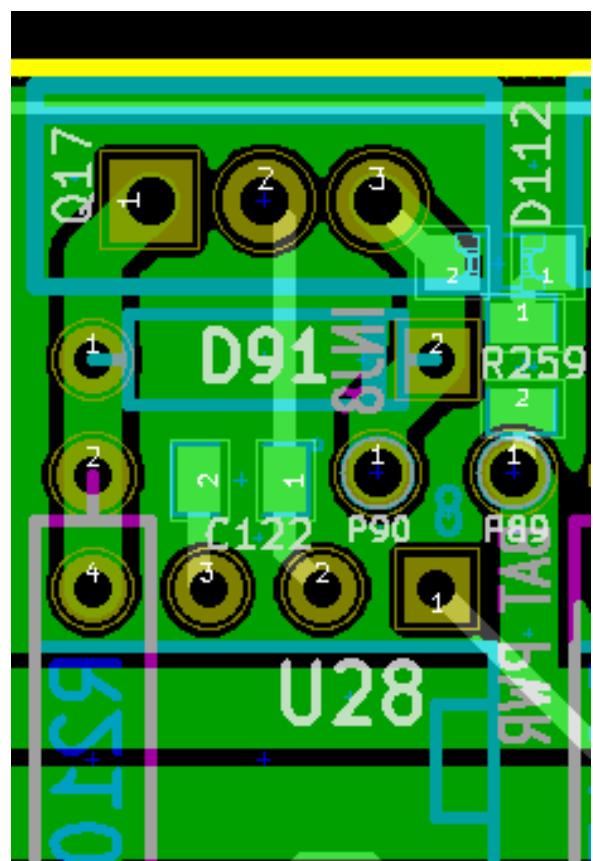
(a) Cut wrong VDD traces



(b) Correct BRV source



(c) FT232 circuit -lacks PNP-!!



(d) Correct transistors connection

Figure 2: Spin1 mods