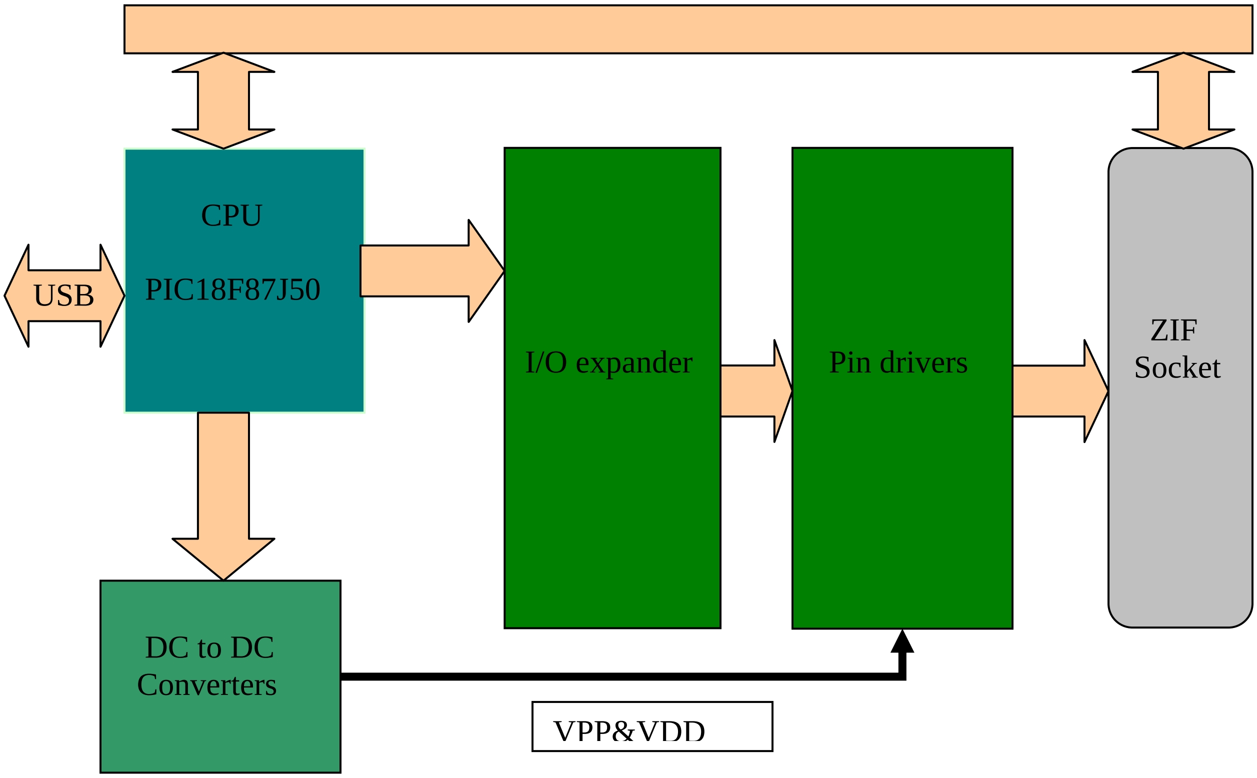
TL866 Programmer

TL866 programmer consists of the following functional blocks:

- **CPU block** based on the Microchip microcontroller Pic18F87J50.
- **I/O Expander** using eight 74HC373 latches and one 74HC164 shift register.
- **Pin drivers** realized with discrete components. Thus we have 16 VPP voltage switches, 24 VDD voltage switches and 25 GND switches.
- **DC to DC converters** wich generate the VPP programming voltage and VDD supply voltage.It is made with two MC34063 circuits.



CPU block

It is based on the PIC18F87J50 microcontroller and is the core of entire device.

The internal firmware is composed of two main modules:

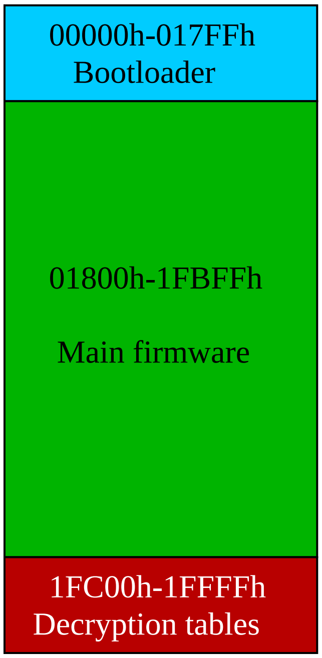
- **Bootloader:** for later firmware upgrade.
- **Main software:** implements all programming algoritms.

The internal controller flash memory has 128Kbytes and is divided as follows:

1-0x00000-0x017FF bootloader

2-0x01800-0x1FBFF main firmware

3-0x1FC00-0x1FFFF decryption tables



Of the three memory areas ,only the two can be updated. The entire 128Kbytes of flash memory is divided in blocks of 1kbyte each, so we have 128 blocks numbered from 0 to 127 as follows:

- 0 To 5 = bootloader (6Kbytes)
- 6 To 126 = main software (121Kbytes)
- 127 = decryption tables (1kbyte)

Updating software is mainly done as follows:

- Erasing blocks 6-126
- Writing this memory area with the new content.

This operation is provided by the bootloader, which controlled by the pc software will perform erase memory, decrypt received data blocks and writing flash memory.

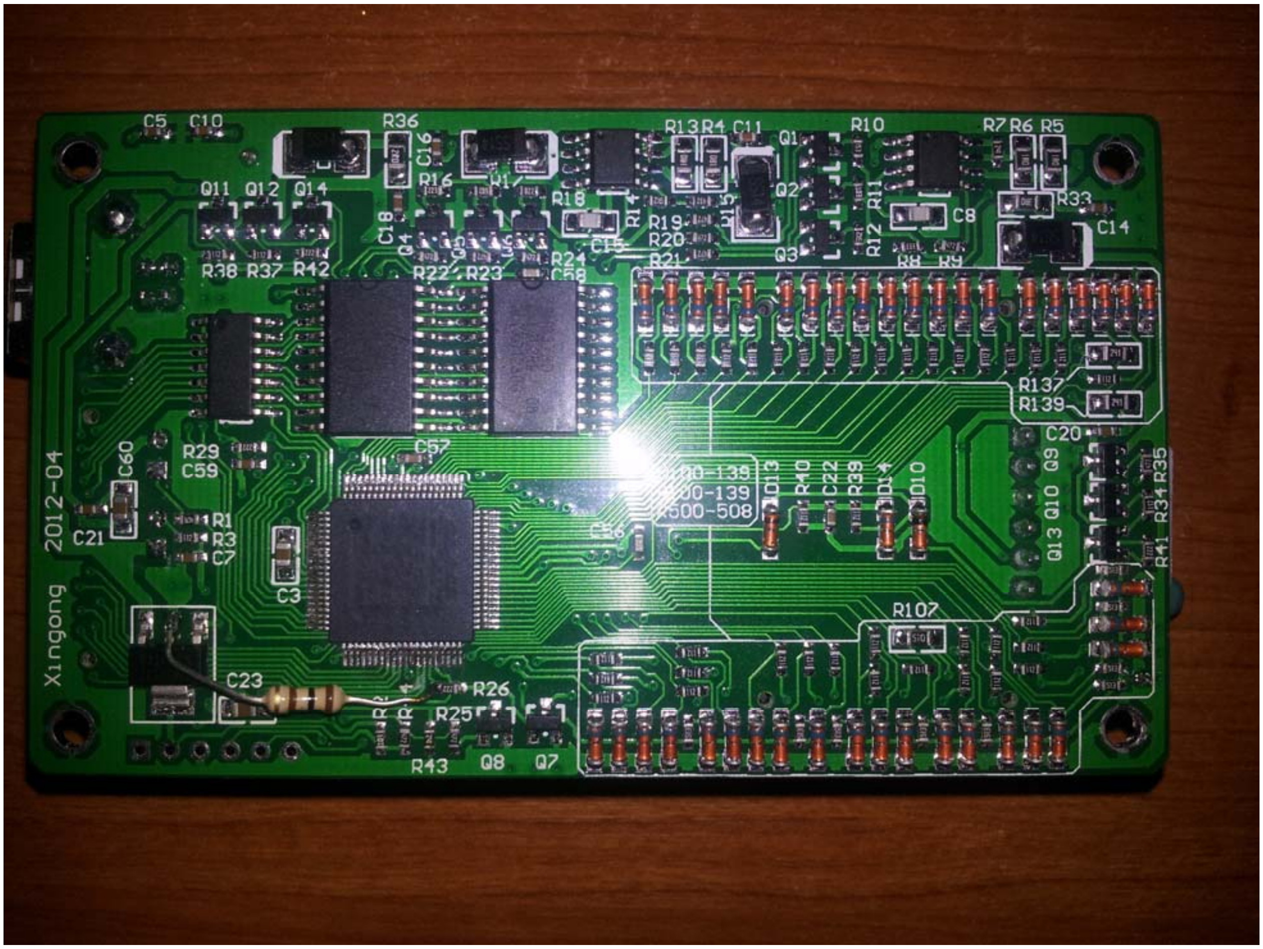
Bootloader

This module is the first that run at startup and is responsible for verifying the integrity of the main firmware and transfer execution to it.

At every startup it does the following operations:

1. initialize the controller ports and peripherals
2. check the state of pin RC1 ; if is logic one then jumps to step 5
3. check if bootloader mode was demanded by the main software; if yes then jumps to step 5.
4. verify the last 4 bytes of main software, if they are 55, AA, A5, 5A is considered that the main software is valid and transfers execution at adress 0x1800; bootloader cease functioning here.
5. USB module initialization
6. Waiting commands from PC
7. loop to step 6

As you can see, we can force the bootloader mode by holding the RC1 pin at logic one at startup, this can be done by soldering an resistor of approx. 100-200ohm between +3.3 V and pin 36 (R26) of the controller:



Although bootloader will check the software integrity , in some cases it may be a valid software signature but either main firmware is missing or corrupted, leading to invalid code execution and the cpu will stall. Out of this situation can only be done by forcing the bootloader mode.
Reviewing: entering bootloader mode can be done only in three cases:

-Forced by pin RC1, which has the highest priority.
-When was demanded by the main software.
-In case the main program is corrupted or missing.

After entering the bootloader mode and USB module initialization, it enters a loop of waiting for orders from the PC, which are only four in number, as follows:

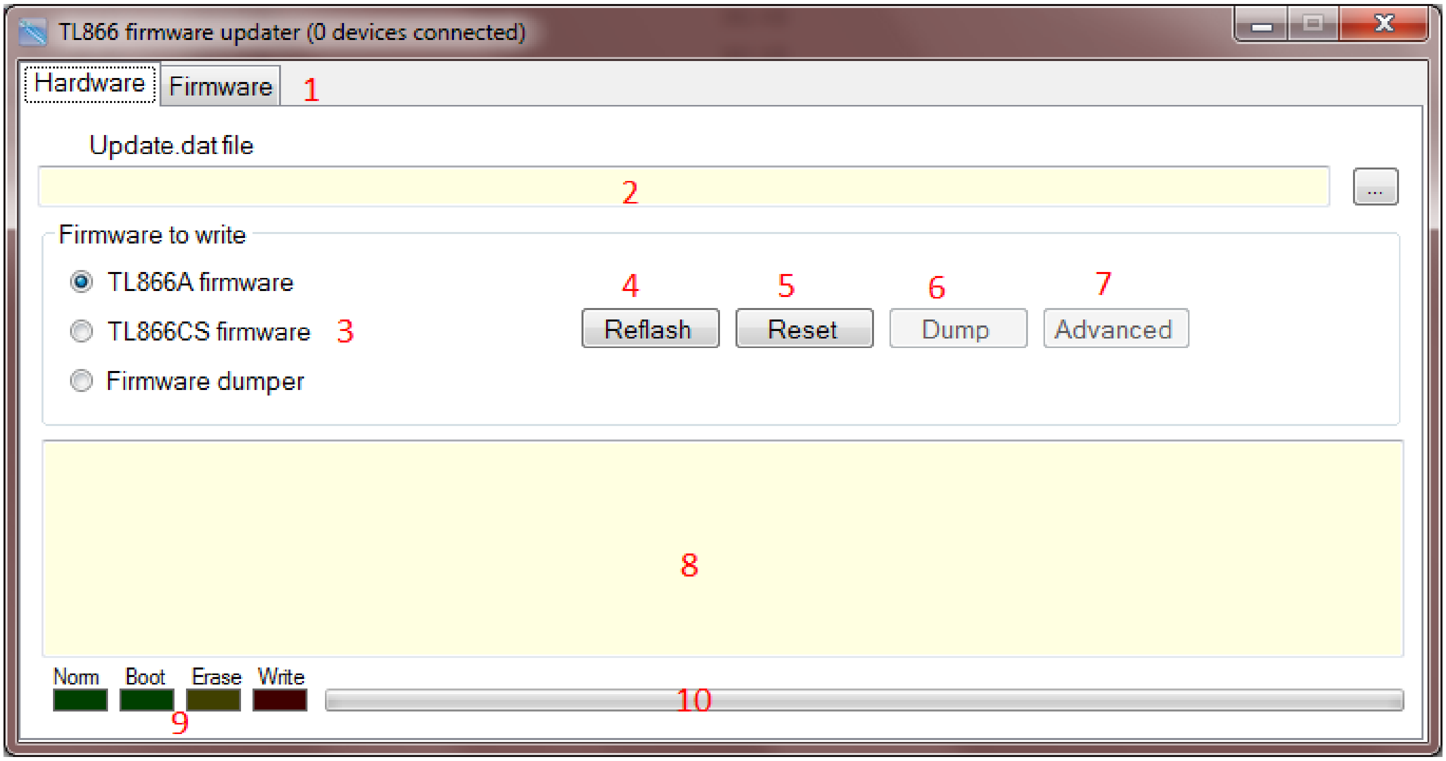
1 RESET resets the controller.
2 REPORT report status of the device, firmware version, serial number and device code.
3 ERASE erase the main program by initializing blocks 6-126 with FF.
4 WRITE decrypts and writes a block of data in the flash memory at specified address.

As a remark, although in step 4 is mandatory that the address specified in the command to be within the 0x01800-0x1FBFF range, actually the bootloader implemented in TL866 will write to any address ordered and has no kind of protection, if such a situation is reached by writing to addresses in blocks 0-5 or 127 then the bootloader will overwrite itself (ridiculous!) or table decryption block 127 will become corrupt and bootloader will not be able to do the right decryption. In both cases the device will become unusable, its restoration can only be done with an external programmer connected to J1 connector.
This is a nasty bug of the bootloader, so beware.
Although reprogramming device in the above case can be made very simple, we need the full firmware. For safety reasons the full firmware is not provided by the manufacturer (shame on you autolelectric).
For such cases i made a small utility software that provides the following functions:

-generates complete firmware using custom serial number and device specific code.
-updates main software by decrypting the update.dat file provided by the manufacturer.
-formatting the main area (blocks 6-126).
-switching between normal mode and bootloader.
-Displays the status of the programmer and identification info.

For safety reasons the program will not perform the formatting and rewriting without presence of the update.dat file, which will be checked for integrity.
The program will not execute any operation if is detected the presence of two or more programmers at the same time.

TL866 firmware generator/updater main interface



● TL866A firmware

○ TL866CS firmware

○ Firmware dumper

The main interface of utility consists of the folowing elements:

1. Switching betwen firmware updater and firmware generator
2. Path to the update.dat file (you will found this file in the same folder with minipro software)
3. Version of the firmware wich will be uploaded; If yo want to make a backup of your firmware please select firmware dumper
4. Reflash button; this will perform the main firmware update, you will need the update.dat file wich will be checked for integrity.
5. Reset button; will reset the programmer and toggling between normal mode and bootloader . If the programmer cannot enter to normal mode then it will stay in bootloader mode.
6. Dump button; this button will become active only if the firmware dumper has been uploaded.
7. Advanced button; this button will become active only if the firmware dumper has been uploaded. You can change device type, serial number and code protection bit on the fly, without having to use an external programmer.
8. Info area; for displaying device informations.
9. Device status; for displaying device status.
10. Write progress; for displaying firmware upgrade progress.

For regular upgrade of the main firmware you will need to:

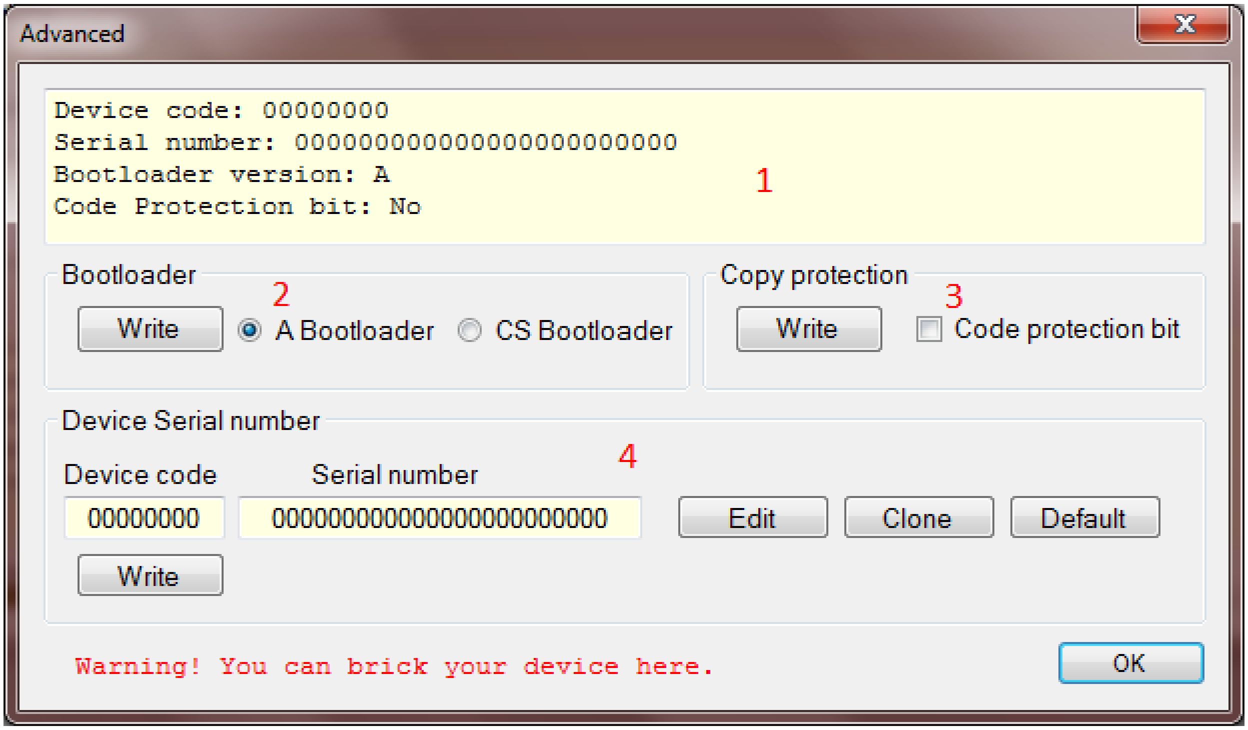
1- Browse for update.dat file
2- Click reflash button

This will update the main firmware and automaticaly will perform the folowing operations:

-switching to bootloader mode
-erase the main firmware area
-decrypt the file update.dat and write the main firmware area with the new content
-switching to normal mode (only if the right firmware was written).

No matter what version of programmer you have, the updater will upload the chosen firmware encrypted with the proper key, so you can have a TL866CS device programmed with TL866A firmware or vice versa.

Advanced window



```
Device code: 00000000
Serial number: 000000000000000000000000
Bootloader version: A
Code Protection bit: No
```

● A Bootloader ○ CS Bootloader

□ Code protection bit

Device code: 00000000

Serial number: 000000000000000000000000

Warning! You can brick your device here.

In this window you can reflash your TL866 without an external programmer but be very carefully!

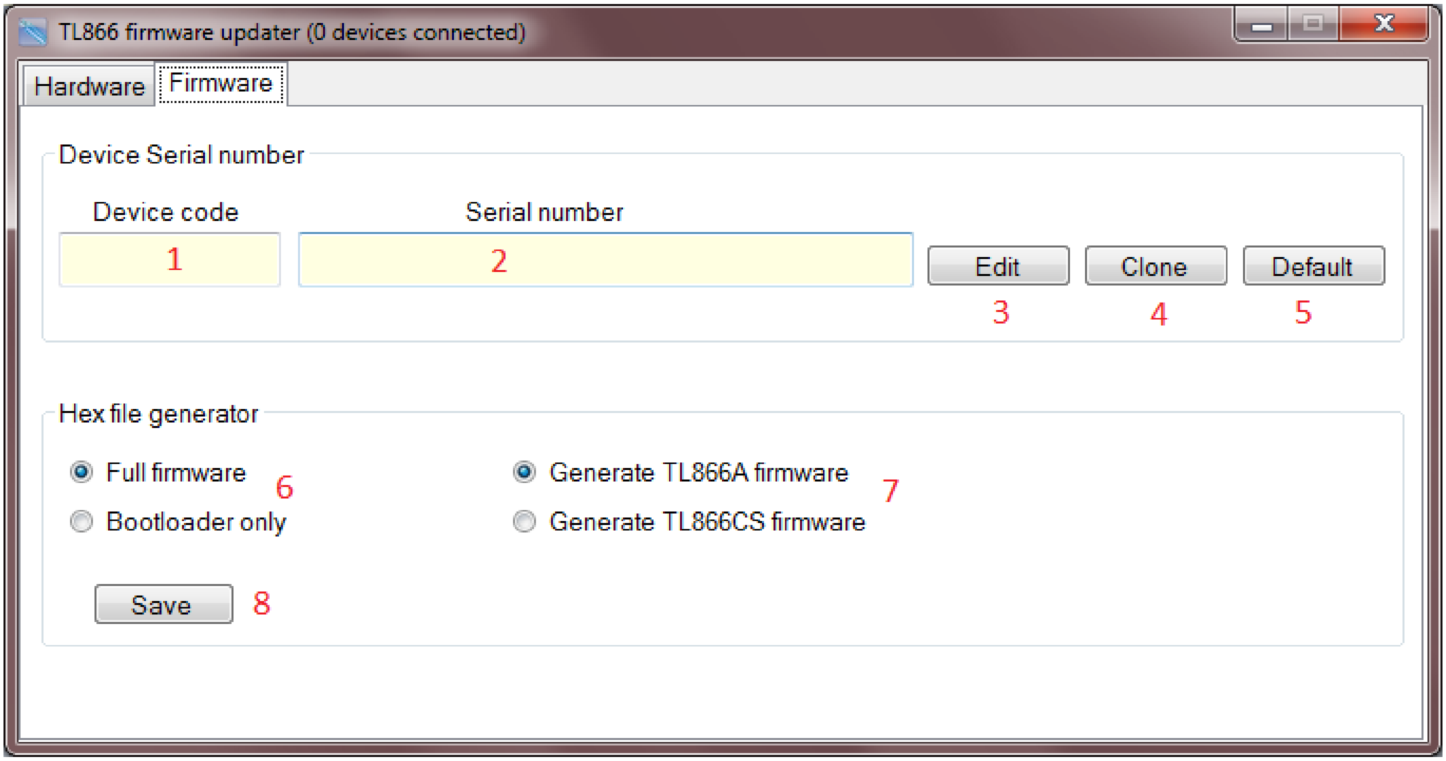
If something is going wrong you can brick your device (of course an external programmer will restore it).

So we have:

1. Device bootloader version,code protection bit status and device serial number info.
2. Bootloader section. Here you can rewrite your bootloader version at your choice, click write button to proceed; the new bootloader will be written very fast (1-2 sec).
3. Code protection bit section. Here you can change the PIC18F87J50 CP0 bit; click write button to proceed.
4. Device serial number section. Well you can change your device code/serial number here if you want!

When you finish with this advanced window please reflash the normal firmware by selecting the desired firmware version from main window or from minipro software.

The full firmware generator



● Full firmware
○ Bootloader only

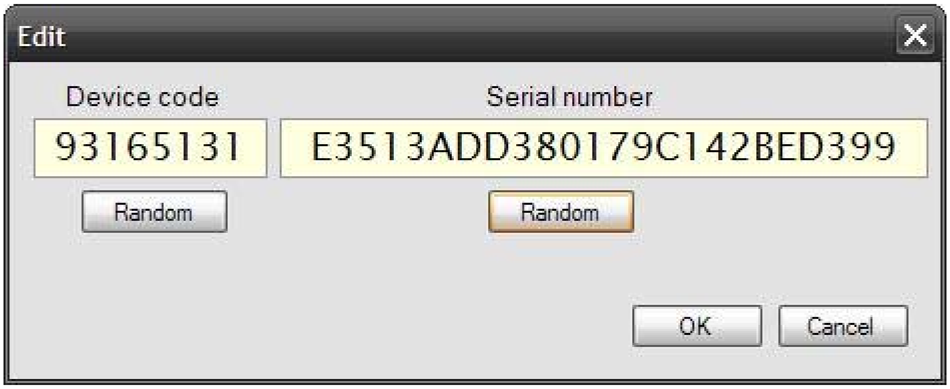
● Generate TL866A firmware
○ Generate TL866CS firmware

The firmware generator interface of this utility consists of the folowing elements:

1- Device code box; content of this box will be displayed in the about box of the Minipro software as DEV Code:xxxxxxxx (max. 8 characters)

2- Serial number box; content of this box will be displayed in the about box of the Minipro software as Serial:xxxxxxxxxxxxxxxxxxxxxxxx (max. 24 characters)

3- Edit buton; for editing device code and serial number as follows:



Device code: 93165131

Serial number: E3513ADD380179C142BED399

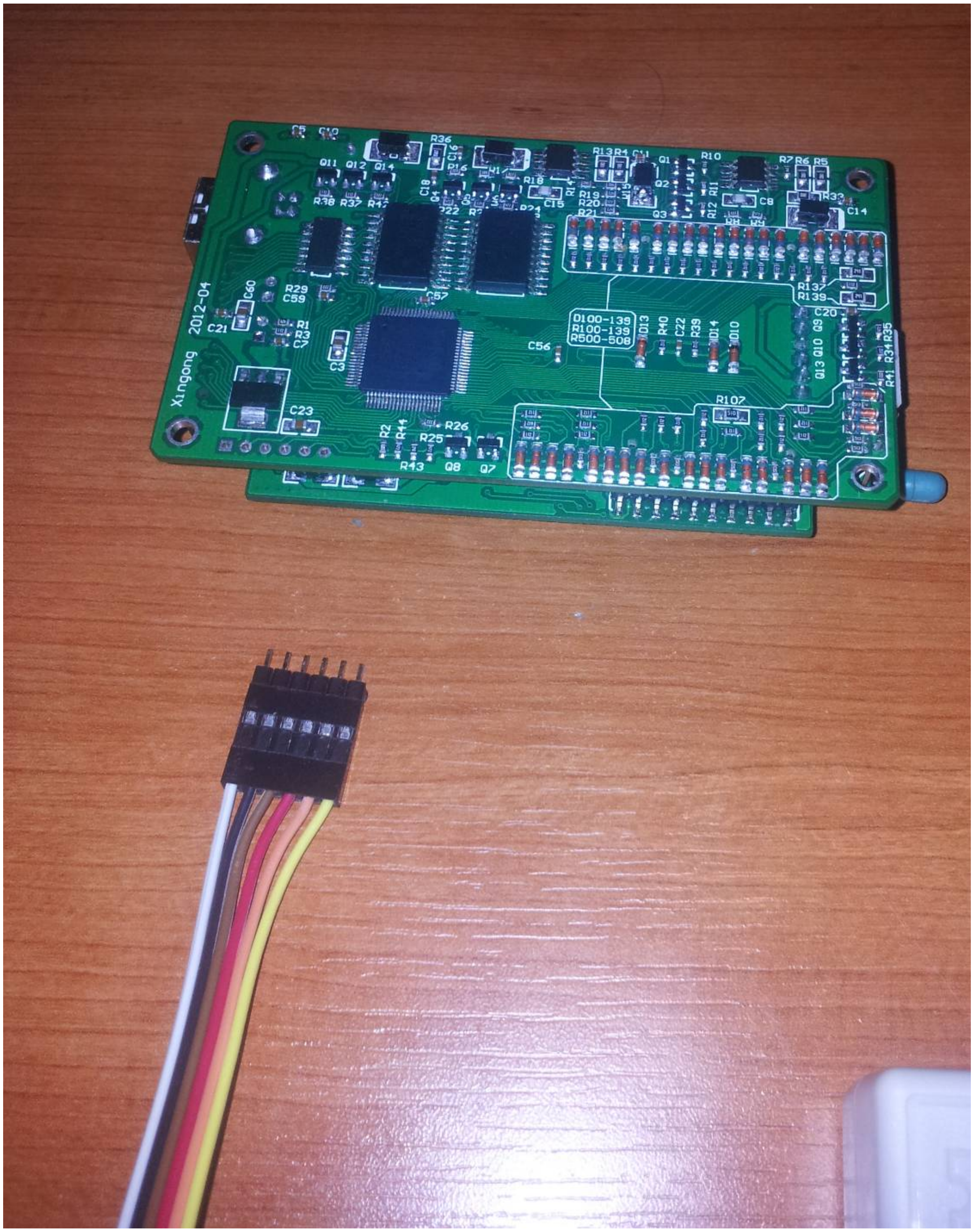
By clicking the two Random butons the device code or serial number will be updated with random numbers. Of course you can manualy enter what you want, the content of these two boxes is not restricted.

4- Clone button; this will copy the device code and serial number from active programmer plugged into the USB. If you intend to transform an CS version to A then clone device code and serial number of the working programmer first, for preserving original serial code and device code.
5- Default button; will update the device code and serial number with default ones.
6- Firmware version selector; The full firmware option will generate the hex file with main firmware writen in the 0x01800-0x1BFFF address range; for bootloader only option, this range will be filled with blank value (FF).
7- Firmware version to generate; you can choose between A and CS version.
8- Save button; by clicking this button you will be prompted to save the generated hex firmware file.

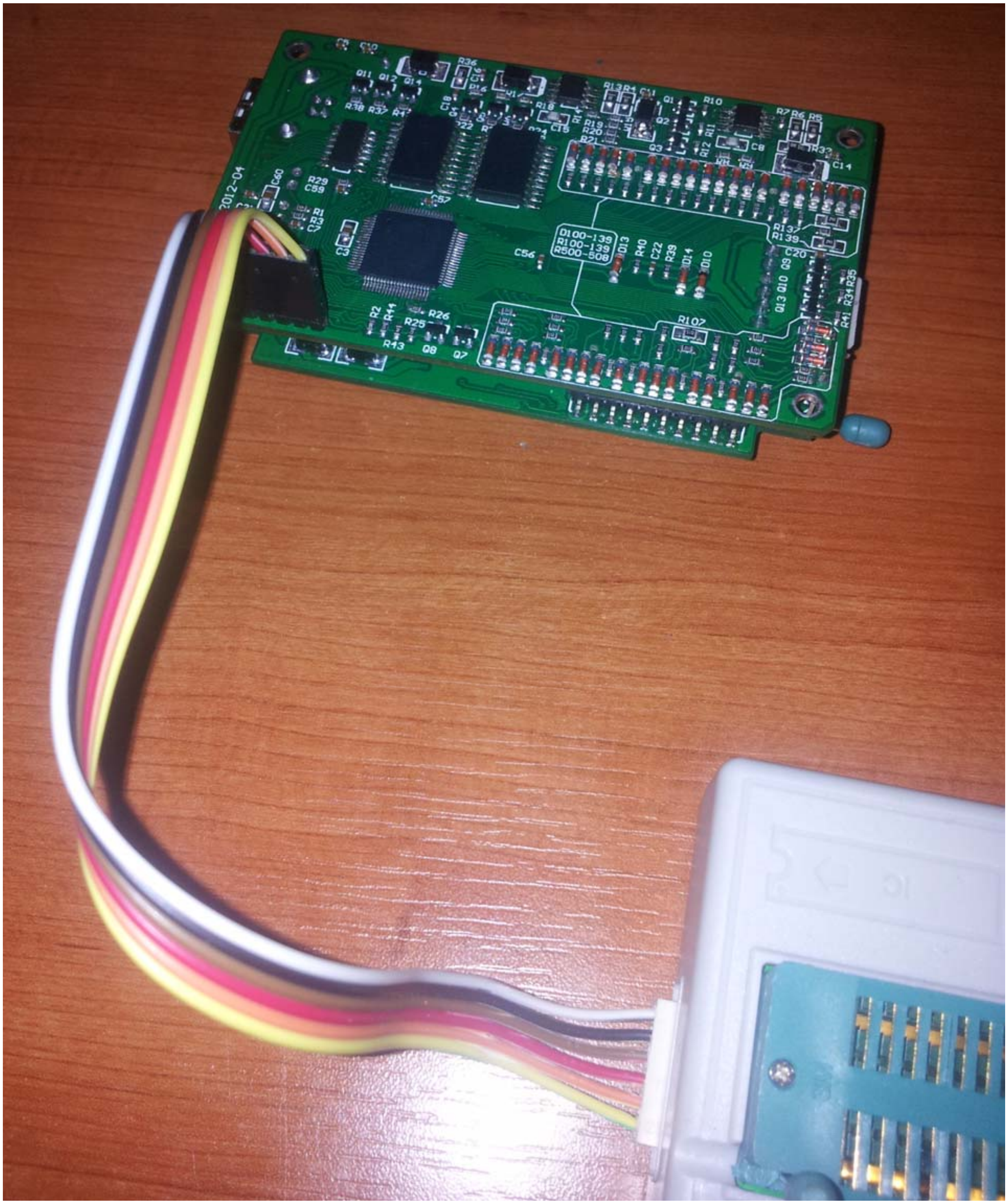
The encrypted device code and serial numbers is writen in the last area of flash memory (0x1FC00-0x1FFFF).
The generated firmware hex file contains the necesary config bytes for programming the PIC18F87J50;
the CP0 copy protect bit is not set, if you need to copy protect the programmer set this bit in the PIC programmer you use for this.

For reflashing the TL866 you will need an PIC programmer (Pickit2,Pickit3,Picpgm,another TL866A). The J1 socket is an standard ICSP connector, just plug your pic programmer cable here, load the hex file generated above and reflash.

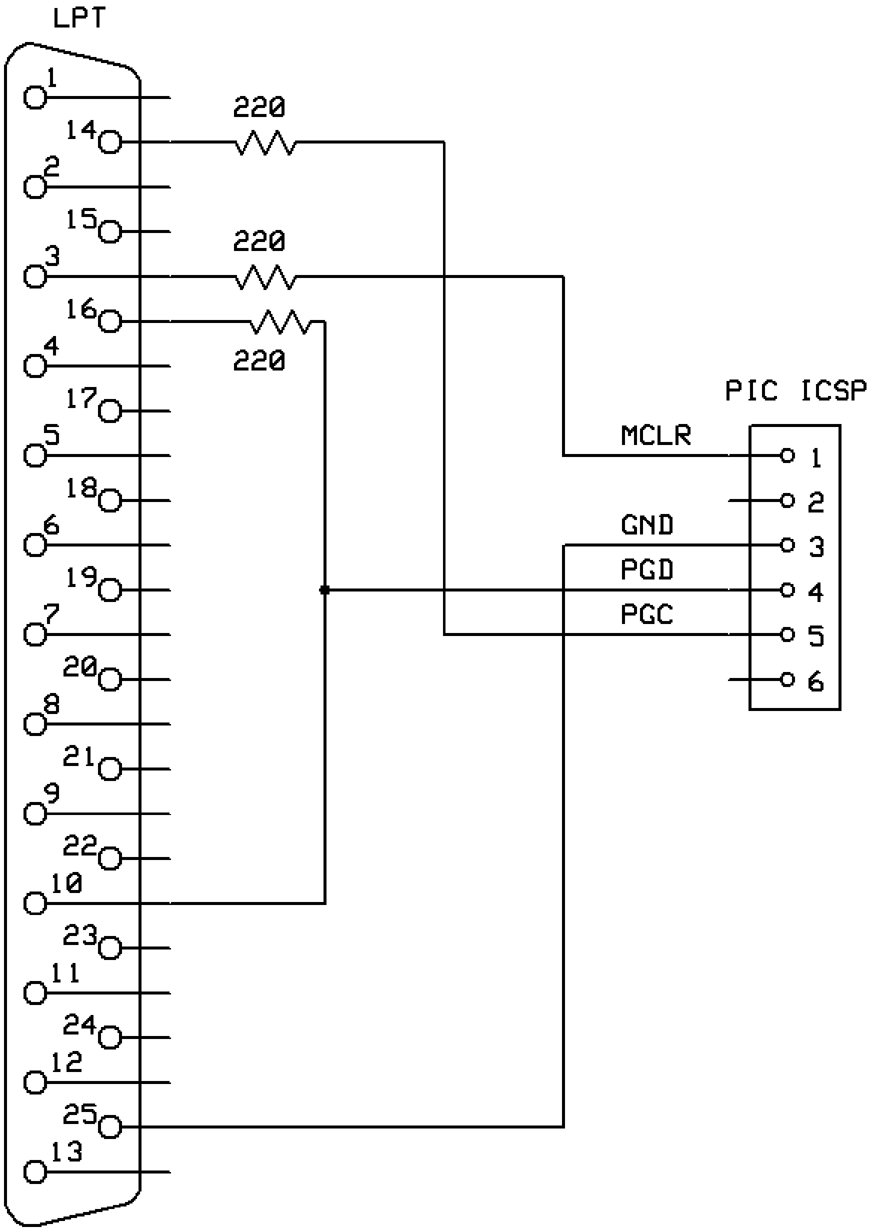
In this image the J1 connector is located down left to the PCB, the first square pad is the pin number 1
The programming cable has a mark to the pin 1, please be sure they match.



In this image the TL866 programmer is reflashed by another TL866, be sure to activate ICSP_VCC enable in the minipro software.

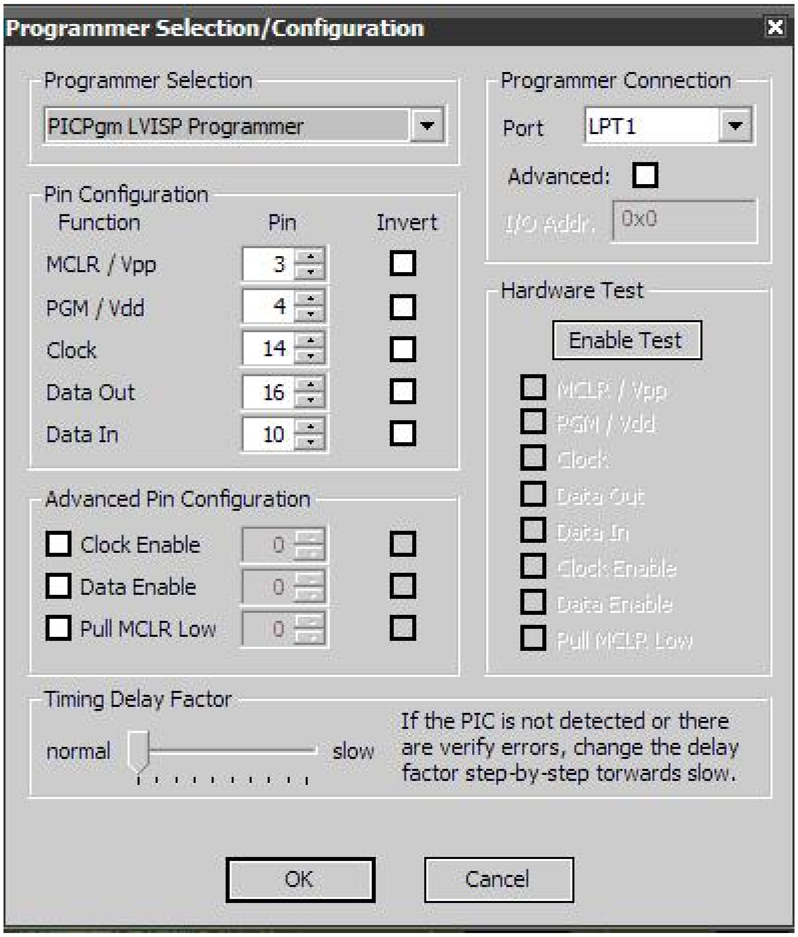


If you don’t have a dedicated PIC programmer then you can build a very simple one, but you will need the presence of an LPT port in your PC like this:



The programming software used is picpgm wich can be downloaded here:
http://picpgm.picprojects.net/download/winpicpgm_v1650.zip

Settings to make picpgm to work with this "programmer"



Programmer Selection/Configuration

Programmer Selection: PICPgm LVISP Programmer

Programmer Connection

Port: LPT1

Advanced: □

I/O Addr.: 0x0

Pin Configuration

Function	Pin	Invert
MCLR / Vpp	3	□
PGM / Vdd	4	□
Clock	14	□
Data Out	16	□
Data In	10	□

Advanced Pin Configuration

□ Clock Enable	0	□
□ Data Enable	0	□
□ Pull MCLR Low	0	□

Hardware Test

- □ MCLR / Vpp
- □ PGM / Vdd
- □ Clock
- □ Data Out
- □ Data In
- □ Clock Enable
- □ Data Enable
- □ Pull MCLR Low

Timing Delay Factor: normal ... slow

If the PIC is not detected or there are verify errors, change the delay factor step-by-step torwards slow.