Pietro Baima (PietroBaima)

PIERIN PIC18: AN OPEN SOURCE BOARD TO LEARN AND TO EXPERIMENT WITH MICROCONTROLLERS

7 June 2013

Nowadays it is plenty of microcontrollers with their development systems. There are tiny microcontrollers, in a little 6 pin case, and there are hulking ones, with 32 bits of stunning capabilities. This huge offer on the market is good, but on the other hand it can confuse those who would like to get experience in this field. For a beginner it's difficult, actually, to choose the right starting development system. In this paper I shall describe a good development system, tailored for people who would like to start off on the right foot. It means to do not waste time puzzling over things that are incomprehensible and useless.

Features

A good educational or fast-prototyping development system is a compromise solution among various elements. It should have these characteristics, at least:

- Easy to handle;
- Powerful enough to have no need to move up toward others microcontrollers;
- Suitable to be used in a multitude of applications, from the most simple to the trickiest;
- Free;
- Easy to be programmed either in assembly or in C.

There is plenty of them on the market, but some compiles for outdated microcontrollers, or run-of-the-mill ones, or compile non standard programming languages and so they are useless and pointless for acquire a good knowledge on microcontrollers, others are expensive or with lite versions showing really limited capacity, others are intricate, and so on...

Choosing the right microcontroller is critical!

The PIERIN

It's a little board I like to name "sassy", because, actually, it uses a microcontroller that is a piece of jewellery: **the PIC18F47J53**.

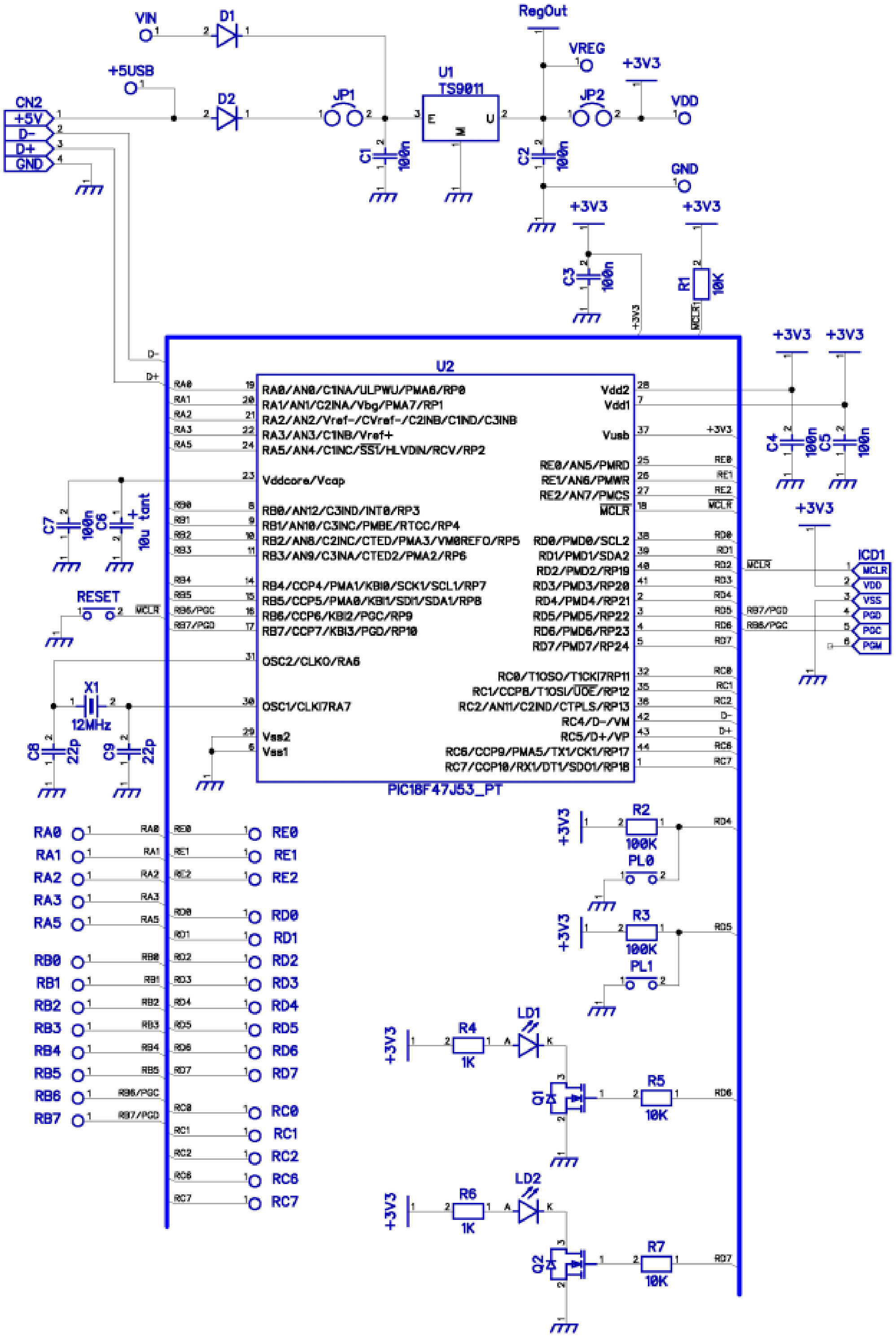
Here there are some characteristic:

- It's a 8 bit microcontroller, so it easy to use and to program in assembly, as well (good to learn);
- It has 128kB FLASH that allows you to do so many things. It has 4 times the FLASH of a PIC18F4550 or a ATmega328. Let me say that with 128KB FLASH you can do amazing things. When you get to fill the microcontroller memory you will have no problems to use any other microcontroller;
- It carries a really big RAM: 4kB;
- ADC with 12 bits of resolution;
- Timers with 8 and 16 bits, PWM, capture etc...
- SPI, I2C, USART and so on and so forth;
- I/O mapping, a really "tasty" characteristic;
- USB on board. With the integrated USB is possible to code really interesting applications, starting with a simple keyboard/mouse, continuing with SD-CARDs as far as audio applications;
- The development system is supplied by Microchip and the C compiler (lite version) works really well (even the optimization is limited). All the Microchip application projects are perfectly working with the lite version, too;
- MPLAB-X is a multi platform system, so the Linux and Mac enthusiasts are stood pat;
- In case you would like to run wild there are third party development systems, as the SDCC;
- REALLY IMPORTANT THING: the Microchip supplies a free huge library of code and applications from which you can learn a lot!
- Development systems are not tricky to use;
- The board carries a preprogrammed USB bootloader with no need of a driver;
- Even the PC bootloader is multiplatform;
- The board can work stand-alone simply connecting it to the USB or in a circuit. There are no microcontroller pin used for something specific but the USB ones;
- It really can replace the raw microcontroller during prototyping, growing into a development tool for custom products;

I think I'm allowed to say it can be considered a good compromise and a good starting point for those who want to learn seriously to use these beautiful objects.

Wiring diagram

Down here is shown the **PIERIN PIC18** wiring diagram.



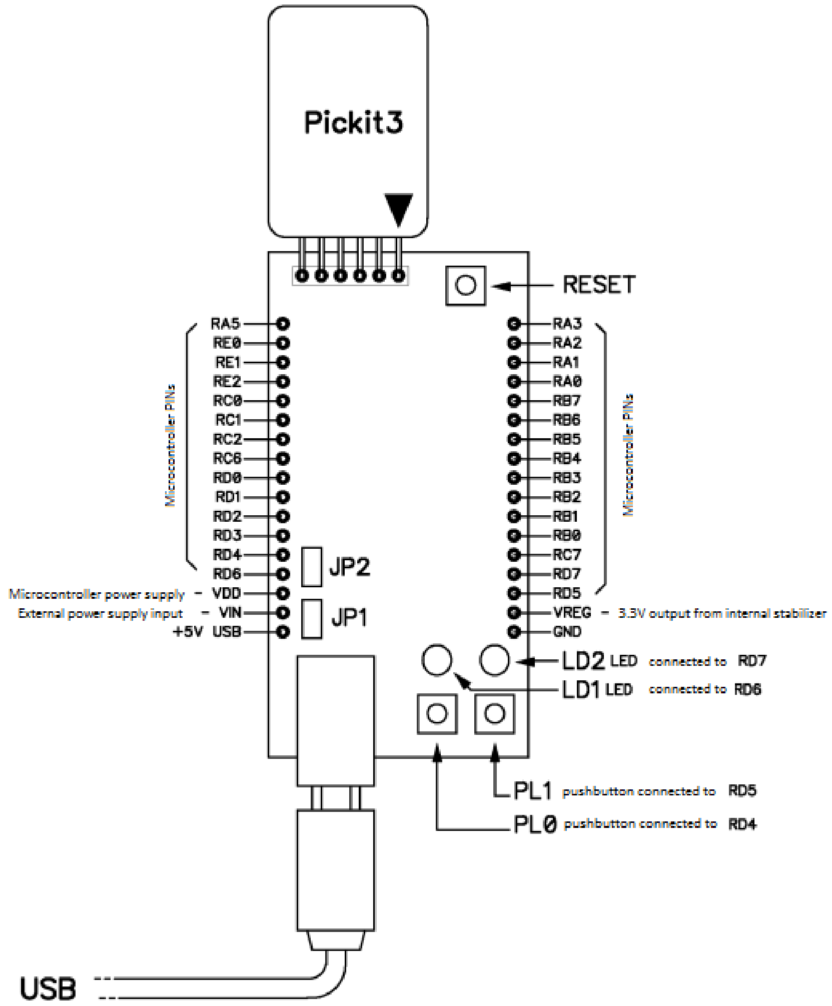
pierin_pic18_sch.png

On the top left there is the connector named USB CN2. From the pin 1 of CN2, we get the 5V supplied by the USB, displayed through the terminal +5USB. Across D2 and JP1, 5V reaches the voltage regulator U1. This regulator can be powered by an external source through the terminal VIN, too. The stabilized voltage at 3.3V is available on the terminal VREG (on the top left) and, through JP2, supplies whole the microcontroller circuit. If JP2 is unplugged, the circuit can be powered through the VDD terminal. The capacitors C3, C4 and C5 are bypass capacitors for the rail voltage and C6 and C7 (on the left) are used by internal regulator to generate the USB interface voltage.

Also on the left is the RESET button, the quartz X1 (12MHz) and two capacitors, C8 and C9, for the oscillator. On the right is the connector for the Pickit3. Going ahead downwards we reach the two pushbuttons PL0 and PL1 with their pull-up resistor and connected to pins RD4 and RD5 of the microcontroller, respectively. On the right bottom there are two leds. They are not driven directly by the microcontroller outputs but through the two mosfet Q1 and Q2. This way of doing things allows to save whole microcontroller outputs fan out.

How to connect it

Down here is shown the connection diagram.



Untitled.png

Starting from the top we reach the 6 pin connector to connect the emulator/debugger Pickit3. It is used to program the microcontroller and to debug it. Yet, via the bootloader, it is possible to do without the programmer because the microcontroller can be programmed through the USB (on the bottom).

The jumpers JP1 and JP2 are very important because they are used to configure the power line. Let me show how to configure them taking a look to the wiring diagram.

JP1=ON JP2=ON

Stand alone mode. The power supply will come by the USB connector. 5V (directly from the USB, on the terminal +5USB) and 3.3V (stabilized, on the terminals VDD and VREG) are available.

JP1=OFF JP2=ON

The board will need an external power supply, connected using the VIN terminal, because it doesn't take the power from the USB. The internal stabilizer is still working and supplies correctly the microcontroller and, eventually, could supply an external circuit. The 3.3V is available through VDD and VREG terminals.

JP1=ON JP2=OFF

The voltage stabilizer is powered by the USB, but it does not supply the microcontroller that needs an external power supply by the VDD terminal. The stabilized 3.3V voltage is available by the terminal VREG.

JP1=OFF JP2=OFF

The stabilizer is not supplied by the USB. It doesn't supply the microcontroller that needs an external power supply by the terminal VDD.

Conclusions

Well, the philosophy behind this kind of didactic and prototyping is explained, but I would like to highlight that **To have a board like this one it is not mandatory. Electronics-familiar people can easily assemble it by themselves** and, whether a pickit3 is not available, it's possible to ask a friend to pop in the bootloader.

Attached to this paper there is the file **pierinpic18.rar** with the wiring diagram pdf and the connection diagram inside. There is, as well, **the .hex file for the bootloader**, so you can download it, you can install it in the microcontroller, and from that time onwards you do not need the pickit3 anymore!

Have a **NICE EXPERIMENTATION!**

The file with the documentation and the bootload can be downloaded using **this link**

Estratto da "http://www.electroyou.it/mediawiki/index.php?title=UsersPages:Pietrobaima:pierin-pic18-an-open-source-board-to-learn-and-to-experiment-with-microcontrollers"