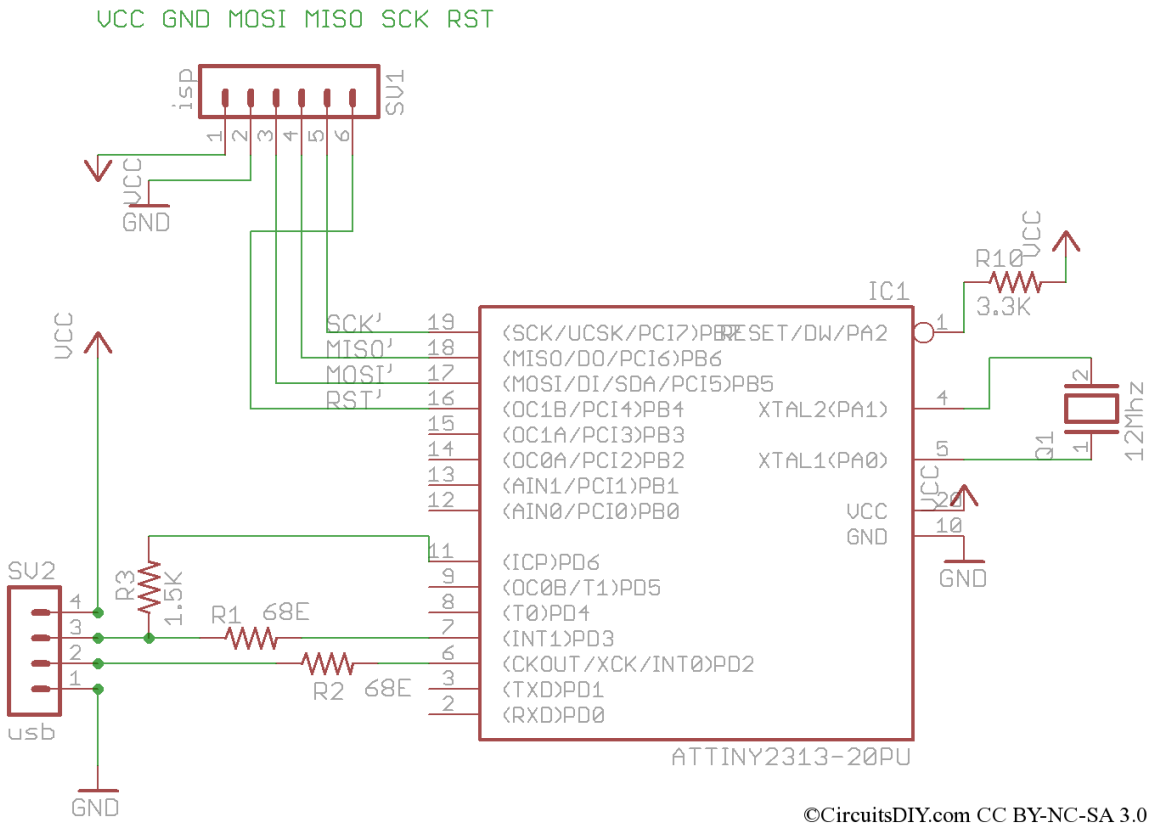
Simplest USB AVR programmer

Last Updated on September 20, 2017 by Arup — 39 Comments

I don't have desktop with legacy ports and don't like going to friend's home every time I need to download something to an AVR. I asked myself "how can i build a simple **diy avr USB programmer**?" .

Well, actually I asked Google, and the answer was Usbtiny. Below I'm describing my simplified Usbtiny programmer.



It's basically an extremely cut version of UsbTinyISP by Ladyada.When I went to build the v2.0 of the programmer, I've made it first on breadboard, and everything ran perfectly and it was able to do the job nicely. Next, I started removing *sidekicks*.

First, I've removed the 74HC125 buffer and gave direct
and it did worked. Next, I've removed few other resisto
essential ones.

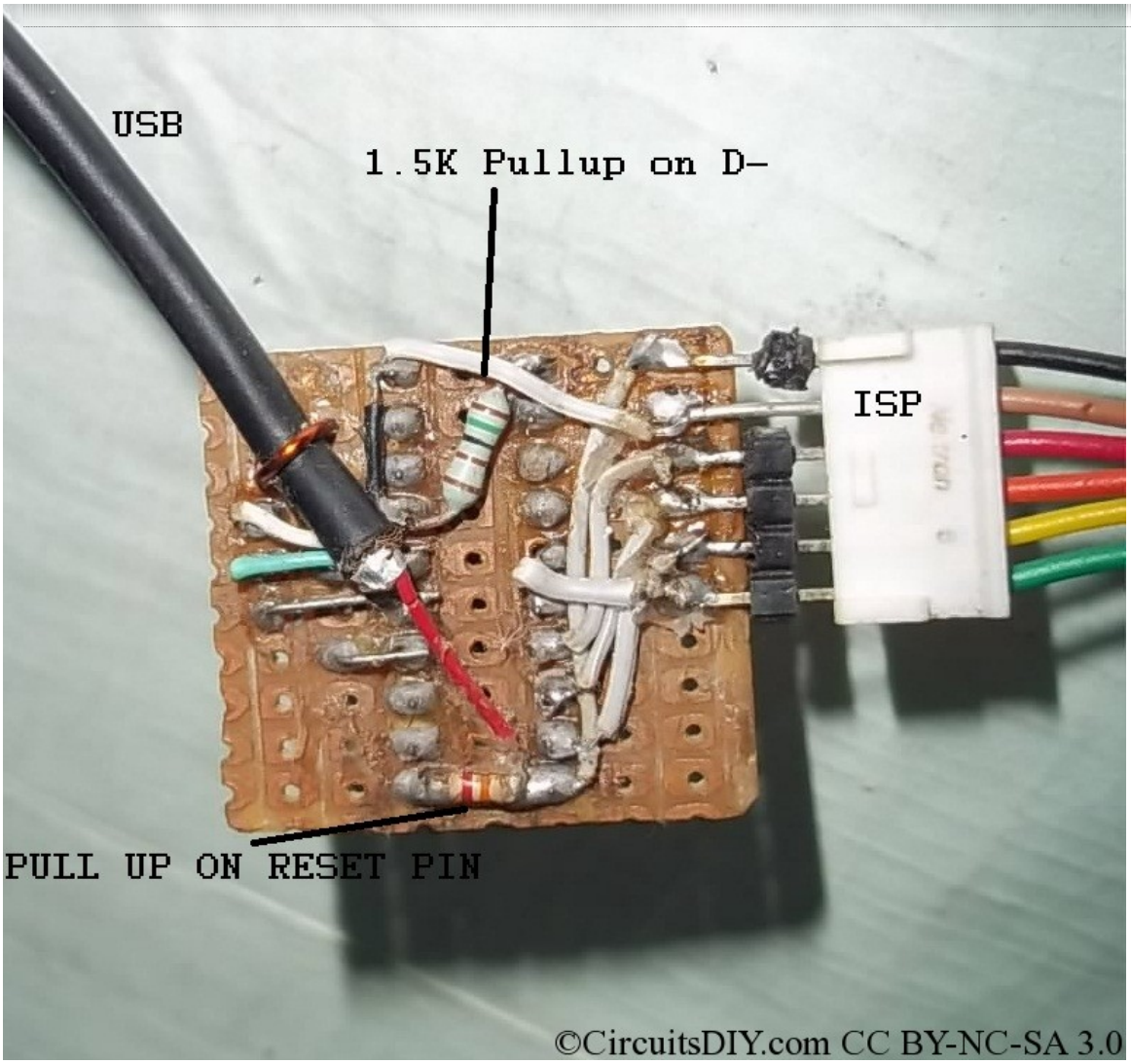
I've converted the ISP connector and USB connector to
job, it is left with the main Attiny2313 IC, one 12Mhz cr

modified version, there is 6 discrete parts (less than 40%),

1. Attiny2313 main IC
2. 12Mhz crystal
3. 3.3K reset pullup resistor
4. 1.5K D- pullup resistor
5. USB bus resistor 2x 68R.

With this minimum surrounding, everything worked correctly. I could have fabricated one small PCB for this but it's so small and tiny, that I made it on a small piece of veroboard.

I've customized the pin layout of the ISP header to m
VCC-GND-MISO-MOSI-SCK-RST.
I've used this one to program Attiny2313, Atmega8, Atm
There's no filtering capacitor in the circuit, nor there is



I stripped down the hardware much and removed the two indicator LEDs which were for
1. Programming : Surely, I know when I am programming, so no need of the LED, and
2. USB OK: If device didn't initialized properly and connected to USB, I'll get error from avrdude as "*Error: Could not find USBtiny device (0×1781/0xc9f)*" , what's the need of the LED then.

Next, I've thought to minify the program, but then thought that was unnecessary. The firmware being 2046bytes, fits tightly into the 2048byte
the code will make it fit in lesser space, but that space
user's project. Hence I've selected the v2.0 firmware as

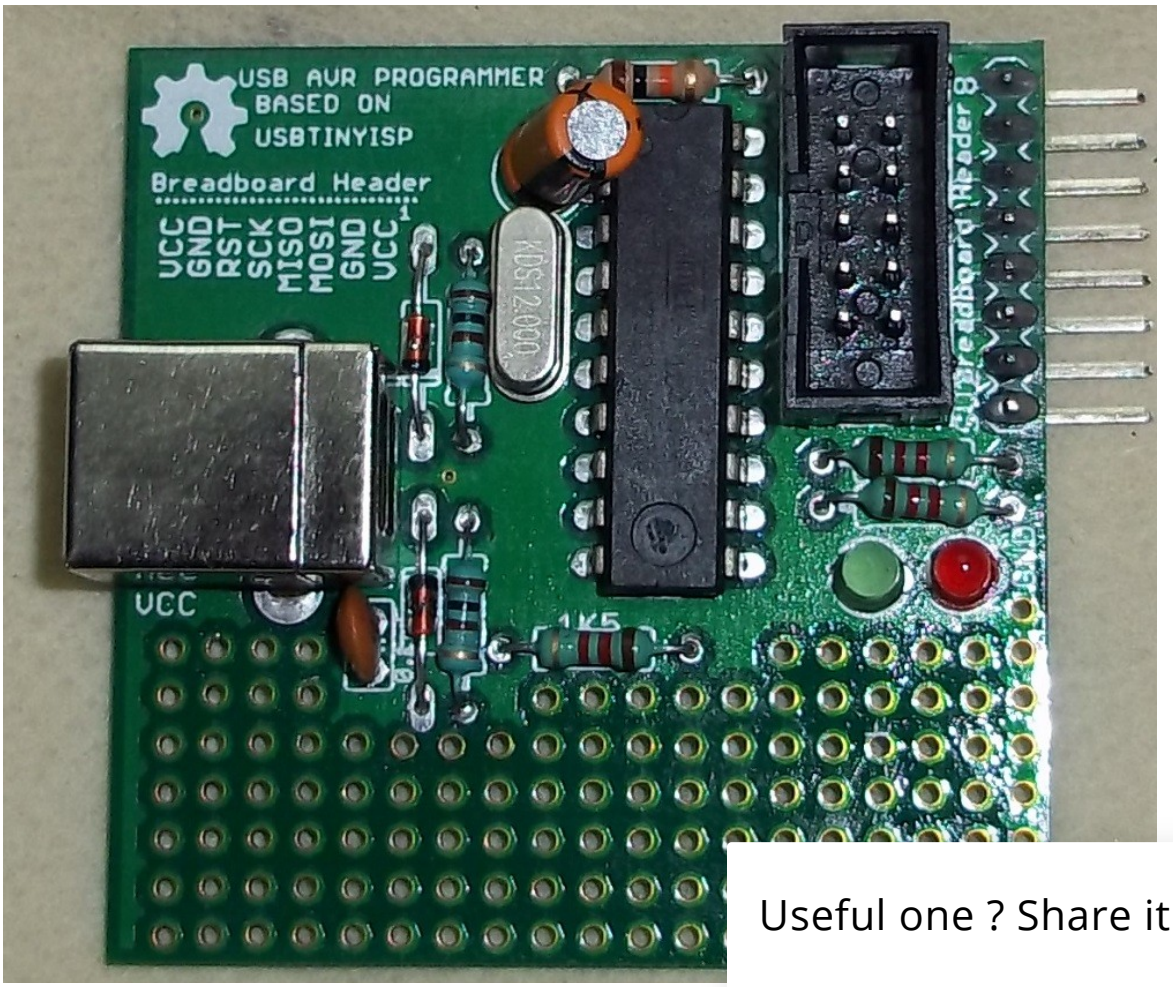
Note: To burn the firmware to Attiny2313, you'll nee
adviced to use any of these simple programmer that
Parallel port.

1. Download the USB driver and also firmware for v2.0 from Ladyada site.
2. Burn the firmware by AVRdude or by PonyProg and set fusebits as example

```
avrdude -c stk200 -p t2313 -U hfuse:w:0xdf:m -U lfuse:w:0xef:m
```

Reference Source: Ladyada and Instructables.

Update: PCB design and Final version



Comments on this article before 07-12-2011, here

- **K Scharf** on October 14, 2011 – 8:34 pm