

Using an STC Auto Programmer with STCGAL

Copying

This document is (c) 2022 Vincent DEFERT and is licensed under the Creative Commons Attribution 4.0 International License. Information about the license can be found at: <http://creativecommons.org/licenses/by/4.0/>

The STC Auto Programmer

A cool thing with STC's MCU is you only need a USB-to-serial adapter to program them. However, as the MCU programming procedure requires to power cycle it, you may find much more convenient to buy, in the same low price range, an "**STC Auto Programmer USB-TTL**". You can easily find it on AliExpress, for instance. Here's what it looks like:



Here are a few purchase links:

<https://www.aliexpress.com/item/4000027276498.html>

<https://www.aliexpress.com/item/1005004606266866.html>

<https://www.aliexpress.com/item/1005001572177861.html>

<https://www.aliexpress.com/item/1005004111347956.html>

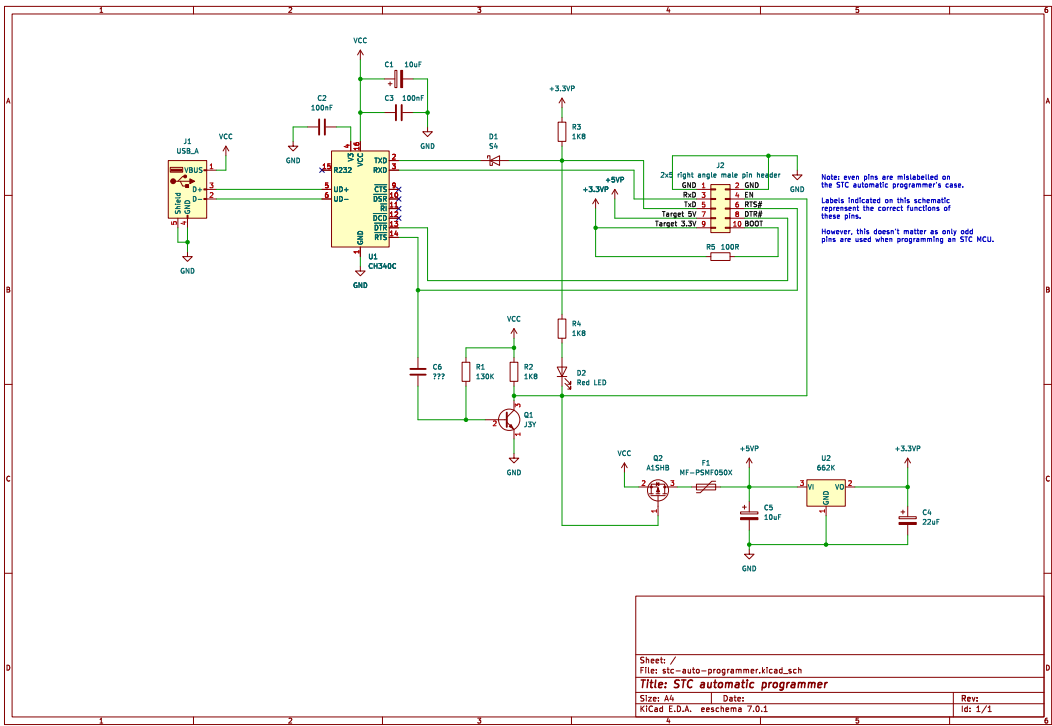
<https://www.aliexpress.com/item/1005003182799433.html>

However, this adapter was designed for use with STC-ISP and the way STCGAL power cycles the MCU is a little different, which will require a little modification.

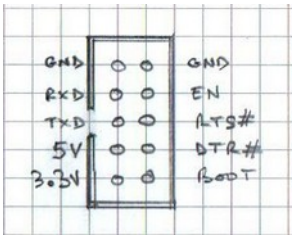
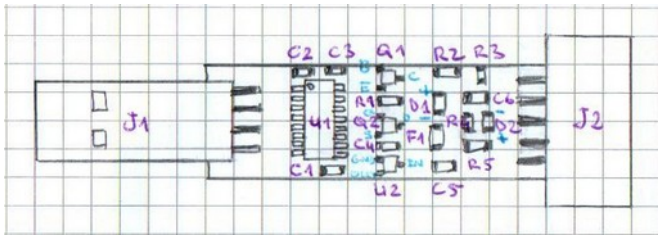
Apparently, STC-ISP wiggles \overline{RTS} to power cycle the MCU, whereas STCGAL just asserts \overline{DTR} (with -a or -a -A dtr) or \overline{RTS} (with -a -A rts) low for 0.5 second.

Using \overline{RTS} is preferable: \overline{DTR} is asserted low when the port is opened, causing the target board to be powered off; this does not happen with \overline{RTS} , so the same STC Auto Programmer can be used both for firmware programming and UART debugging.

Schematic diagram

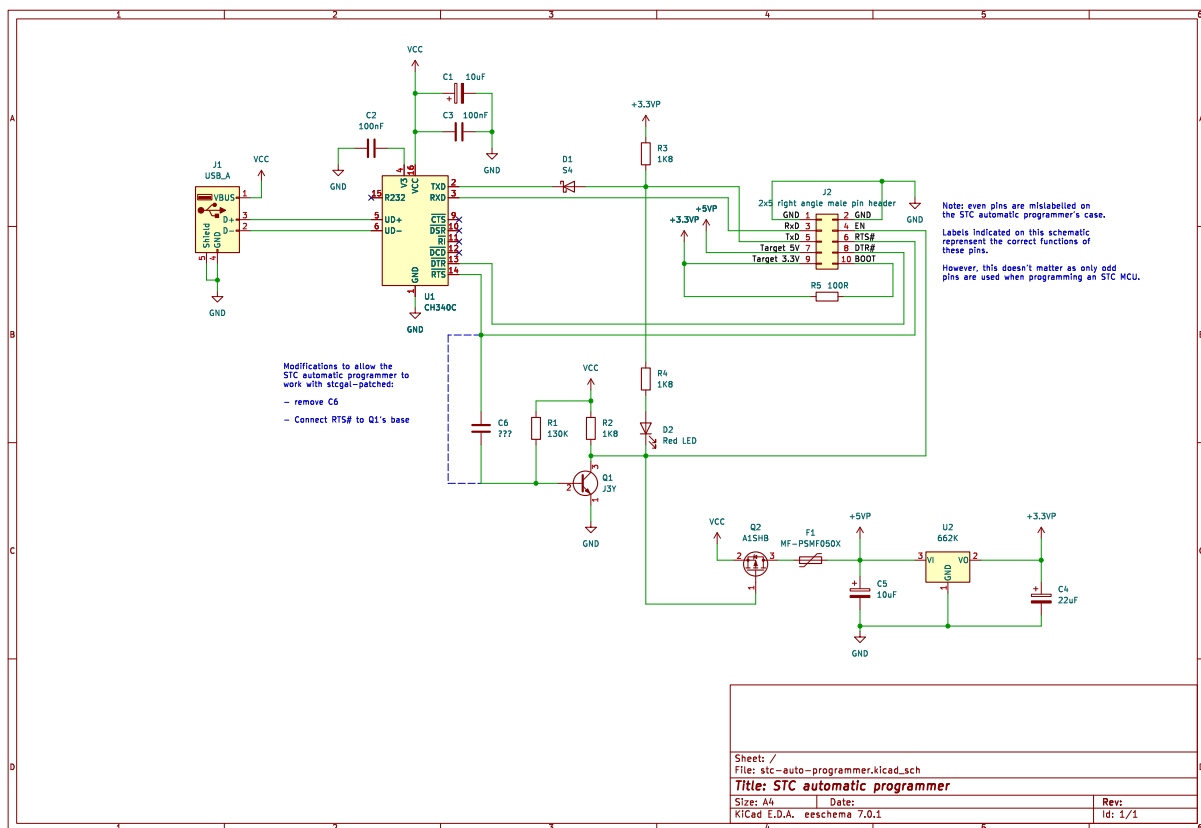


Board layout and connector pinout



Hardware modification for use with STCGAL

The required modification is to connect $\overline{\text{RTS}}$ directly to Q1's base, as shown on the schematic below:



Improved STC MCU Programmer

If you like DIY electronics, building an STC MCU programmer yourself is easy, and allows you to customise it to your taste. The **Improved_STC_MCU_Programmer/** directory of this repository contains an example of such a custom programmer, with the following improvements:

- RESET push button to allow for manual power cycling of the target MCU.
- Slide switch to select the target MCU's supply voltage.
- Uses an SOP-8 CH340N instead of an SOP-16 CH340C.
- USB B socket instead of a USB A plug so the programmer can be placed further away from the computer, and leaves room to connect other devices on adjacent USB ports of the computer.

