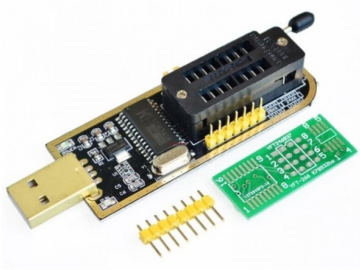


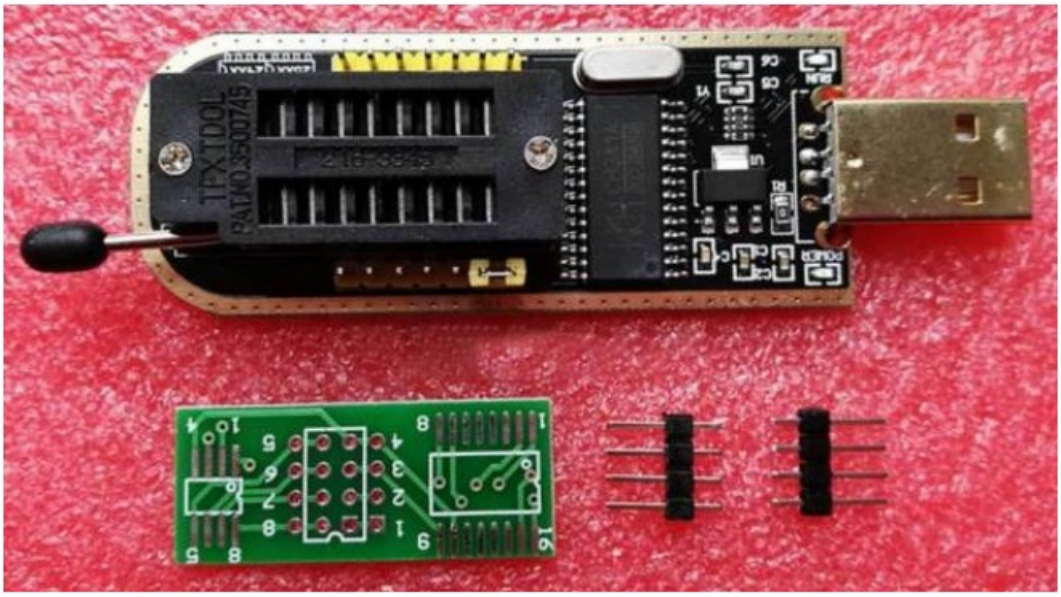
saturn CH341A Mini Flash Programmer Instructions

SATURN

CH341A Mini Flash Programmer Instructions

Contents

CH341A Mini Flash Programmer



Contents: [Dobrica PavlinuÅ¡iÄ ‘s random unstructured stuff]

This is 5V device without modification! It works for me(tm) but you have been warned http://www.eevblog.com/forum/repair/ch341a-serial-memory-programmer-power-supply-fix/Just lift pin 28 and put some tape under it to prevent any short circuit. Then solder one wire from this pin to pin 2 of AMS1117 and also to C4. The side of the C4 to connect is the one connected to CH341A pin 9.
CHM341A-3V3-fix.jpg

jumper on pins

1-2 SPI flash programmer mode
2-3 TTL serial mode
supported out-of-box by flashrom, probably better option, but anyway:

https://github.com/setarcos/ch341prog

dpavlin@nuc:/nuc/ch341a$ git clone https://github.com/setarcos/ch341prog
Cloning into ‘ch341prog’…
remote: Counting objects: 104, done.
remote: Total 104 (delta 0), reused 0 (delta 0), pack-reused 104
Receiving objects: 100% (104/104), 34.79 KiB | 0 bytes/s, done.
Resolving deltas: 100% (61/61), done.
Checking connectivity… done.
dpavlin@nuc:/nuc/ch341a$ cd ch341prog/
dpavlin@nuc:/nuc/ch341a/ch341prog$ make
gcc -std=gnu99 -Wall ch341a.c main.c -o ch341prog -lusb-1.0
dpavlin@nuc:/nuc/ch341a/ch341prog$ ./ch341prog

Usage:

-h, –help display this message
-i, –info read the chip ID info
-e, –erase erase the entire chip
-l, –length <bytes> manually set length
-w, –write <filename> write chip with data from filename
-r, –read <filename> read chip and save data to filename
-t, –turbo increase the i2c bus speed (-tt to use much faster speed)

-d, –double double the spi bus speed

I2C in userspace

https://sourceforge.net/projects/ch341eepromtool/

dpavlin@nuc:/nuc/ch341a/ch341eepromtool_0.5$ gcc -o ch341eeprom ch341eeprom.c ch341funcs.c -lusb-1.0
dpavlin@nuc:/nuc/ch341a/ch341eepromtool_0.5$
dpavlin@nuc:/nuc/ch341a/ch341eepromtool_0.5$ ./ch341eeprom
ch341eeprom – an i2c EEPROM programming tool for the WCH CH341a IC
Version 0.5 copyright (c) 2011 asbokid <ballymunboy@gmail.com>
This program comes with asbolutely no warranty; This is free software,
and you are welcome to redistribute it under certain conditions:
GNU GPL v3 License: http://www.gnu.org/licenses/gpl.html

Usage:
-h, –help display this text
-v, –verbose verbose output
-d, –debug debug output
-s, –size size of EEPROM {24c32|24c64}
-e, –erase erase EEPROM (fill with 0xff)
-w, –write <filename> write EEPROM with image from filename
-r, –read <filename> read EEPROM and save image to filename
Example: ch341eeprom -v -s 24c64 -w bootrom.bin

flashrom SPI

Not needed anymore, included in mainline flashrom
git clone https://github.com/urjaman/flashrom/
git checkout -b origin/ch341a origin/ch341a
dpavlin@nuc:/nuc/flashrom$ sudo apt-get install pciutils-dev

linux kernel spi module

dpavlin@nuc:/nuc$ git clone https://github.com/gschorcht/spi-ch341-usb.git
Cloning into ‘spi-ch341-usb’…
remote: Counting objects: 63, done.
remote: Total 63 (delta 0), reused 0 (delta 0), pack-reused 63
Unpacking objects: 100% (63/63), done.
dpavlin@nuc:/nuc$ cd spi-ch341-usb
dpavlin@nuc:/nuc/spi-ch341-usb$ make
make -C /usr/src/linux-headers-4.14.0-3-amd64/ M=/nuc/spi-ch341-usb modules
make[1]: Entering directory ‘/usr/src/linux-headers-4.14.0-3-amd64’
CC [M] /nuc/spi-ch341-usb/spi-ch341-usb.o
Building modules, stage 2.
MODPOST 1 modules
CC /nuc/spi-ch341-usb/spi-ch341-usb.mod.o
LD [M] /nuc/spi-ch341-usb/spi-ch341-usb.ko
make[1]: Leaving directory ‘/usr/src/linux-headers-4.14.0-3-amd64’
dpavlin@nuc:/nuc/spi-ch341-usb$
dpavlin@nuc:/nuc/spi-ch341-usb$ sudo make install
[sudo] password for dpavlin:
dpavlin@nuc:/nuc/spi-ch341-usb$ sudo modprobe spi-ch341-usb
[525021.048281] spi-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output cs0 SPI slave with cs=0
[525021.048285] spi-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output cs1 SPI slave with cs=1
[525021.048287] spi-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output cs2 SPI slave with cs=2

[525021.048290] spi-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio4 gpio=0 irq=0 (hwirq)
[525021.048292] spi-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio5 gpio=1 irq=1
[525021.048296] spi-ch341-usb 2-2.1.4:1.0: ch341_spi_probe: SPI master connected to SPI bus 0
[525021.048426] spi-ch341-usb 2-2.1.4:1.0: ch341_spi_probe: SPI device /dev/spidev0.0 created
[525021.048516] spi-ch341-usb 2-2.1.4:1.0: ch341_spi_probe: SPI device /dev/spidev0.1 created
[525021.048596] spi-ch341-usb 2-2.1.4:1.0: ch341_spi_probe: SPI device /dev/spidev0.2 created
[525021.049147] spi-ch341-usb 2-2.1.4:1.0: ch341_usb_probe: connected
[525021.049194] usbcore: registered new interface driver spi-ch341-usb

linux kernel i2c module

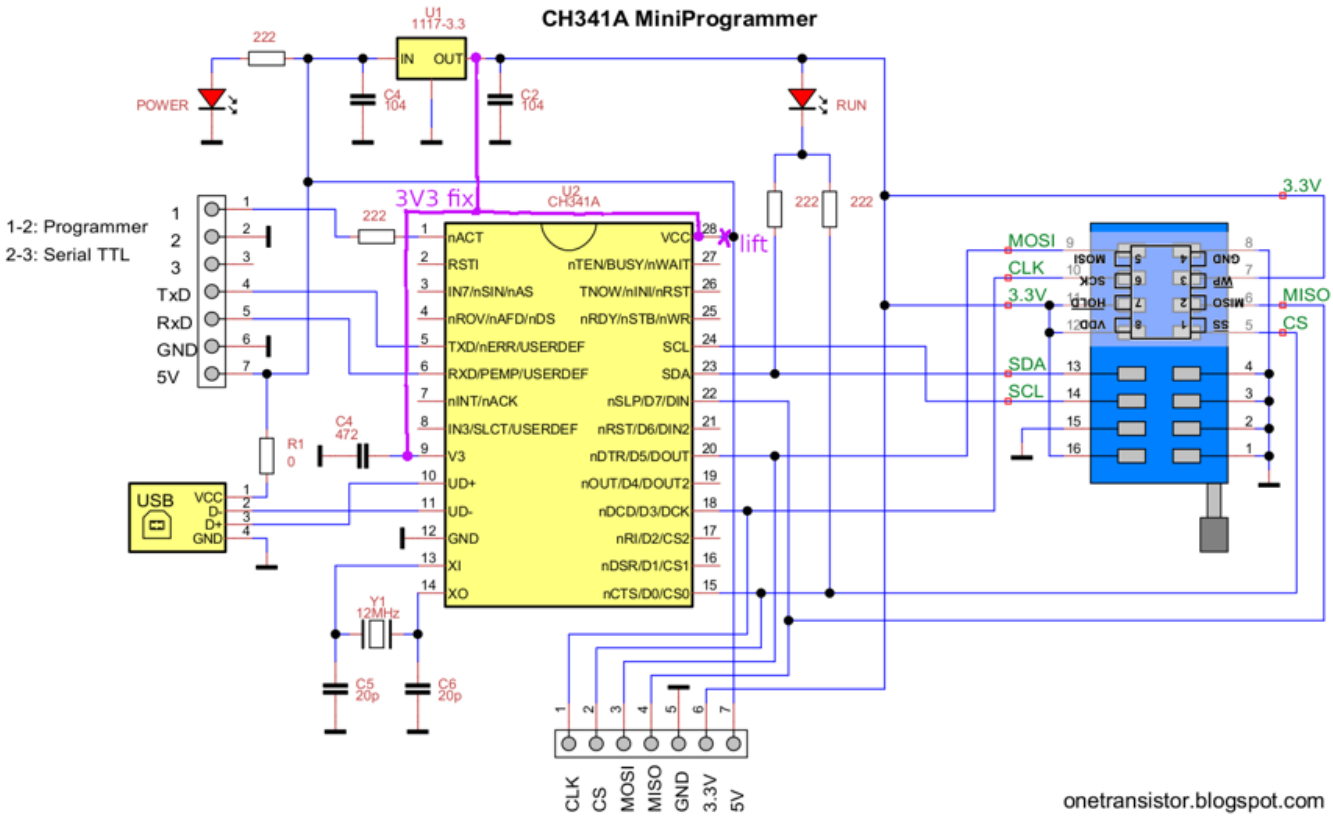
https://github.com/gschorcht/i2c-ch341-usb

root@nuc:/nuc# git clone **https://github.com/gschorcht/i2c-ch341-usb.git**
Cloning into 'i2c-ch341-usb'…
remote: Counting objects: 39, done.
remote: Total 39 (delta 0), reused 0 (delta 0), pack-reused 39
Unpacking objects: 100% (39/39), done.
root@nuc:/nuc# cd i2c-ch341-usb
root@nuc:/nuc/i2c-ch341-usb# make
make -C /usr/src/linux-headers-4.14.0-3-amd64/ M=/nuc/i2c-ch341-usb modules
make[1]: Entering directory '/usr/src/linux-headers-4.14.0-3-amd64'
CC [M] /nuc/i2c-ch341-usb/i2c-ch341-usb.o
Building modules, stage 2.
MODPOST 1 modules
CC /nuc/i2c-ch341-usb/i2c-ch341-usb.mod.o
LD [M] /nuc/i2c-ch341-usb/i2c-ch341-usb.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.14.0-3-amd64'
root@nuc:/nuc/i2c-ch341-usb#
root@nuc:/nuc/i2c-ch341-usb# sudo make install
root@nuc:/nuc/i2c-ch341-usb# modprobe i2c-ch341-usb

[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output gpio0 gpio=0 irq=0
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output gpio1 gpio=1 irq=1
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output gpio2 gpio=2 irq=2
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: output gpio3 gpio=3 irq=3
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio4 gpio=4 irq=4 (hwirq)
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio5 gpio=5 irq=5
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio6 gpio=6 irq=6
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_cfg_probe: input gpio7 gpio=7 irq=7
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_i2c_probe: created i2c device /dev/i2c-8
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_i2c_set_speed: Change i2c bus speed to 100 kbps
[Wed Feb 7 16:37:00 2018] i2c-ch341-usb 2-2.1.4:1.0: ch341_usb_probe: connected
[Wed Feb 7 16:37:00 2018] usbcore: registered new interface driver i2c-ch341-usb

schematics and info

http://onetransistor.blogspot.hr/2017/08/ch341a-mini-programmer-schematic.html
http://www.zoobab.com/ch341-usb-spi-i2c-uart-isp-dongle
I added soic 8 pinout over zif socket, because position of pin 1 is not obvious (or clearly marked anywhere on top).
I suggest that you fix that with silver sharpy.



onetransistor.blogspot.com

alternative schematics

https://github.com/Upcycle-Electronics/CH341A-Pro
ch341Apro_schematicV01.pdf

Documents / Resources

	saturn CH341A Mini Flash Programmer [pdf] Instructions CH341A Mini Flash Programmer, CH341A, Mini Flash Programmer, Flash Programmer, Programmer

References

- [CH341A Mini Programmer Schematic and Drivers · One Transistor]
- [CH341A Serial Memory Programmer Power Supply Fix - Page 1]
- [The GNU General Public License v3.0 - GNU Project - Free Software Foundation]
- [CH341 USB SPI I2C UART ISP dongle - .[ZooBaB].]

- GitHub - gschorcht/i2c-ch341-usb: A Linux kernel driver for ch341 emulating the I2C bus
- GitHub - gschorcht/i2c-ch341-usb: A Linux kernel driver for ch341 emulating the I2C bus
- GitHub - gschorcht/spi-ch341-usb: Linux kernel driver for CH341A USB to SPI and GPIO adapters
- GitHub - setarcos/ch341prog: A simple SPI/IIC programer based on ch341a.
- GitHub - Upcycle-Electronics/CH341A-Pro: Retrace pics and schematic for the Cheap Black CH341A Serial Flash EEPROM BIOS Programmer
- GitHub - urjaman/flashrom: mirror of flashrom.org's svn + some of my own development branches
- User Manual