

# How to Remove Developer Lock on Chromebooks

Alexander Lukens

## 1. Materials Needed

- a. USB Flash Chip Programmer

[https://www.amazon.com/gp/product/B07T61DXFK/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o06\\_s00?ie=UTF8&psc=1](https://www.amazon.com/gp/product/B07T61DXFK/ref=ppx_yo_dt_b_asin_title_o06_s00?ie=UTF8&psc=1)

- b. 1.8V Adapter for programmer

[https://www.amazon.com/gp/product/B072KYK2DR/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o06\\_s00?ie=UTF8&psc=1](https://www.amazon.com/gp/product/B072KYK2DR/ref=ppx_yo_dt_b_asin_title_o06_s00?ie=UTF8&psc=1)

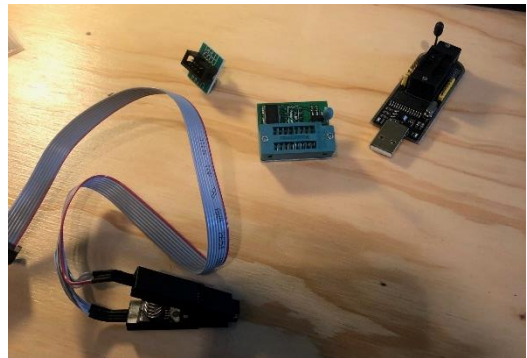
- c. Computer with Ubuntu Linux installed (tutorial to install Ubuntu

<https://tutorials.ubuntu.com/tutorial/tutorial-create-a-usb-stick-on-windows#0>)

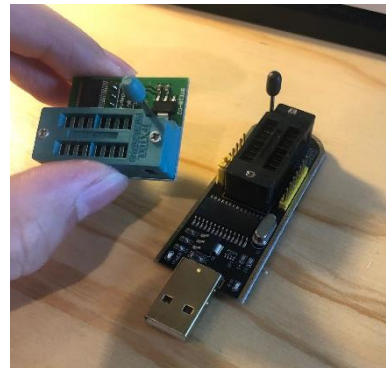
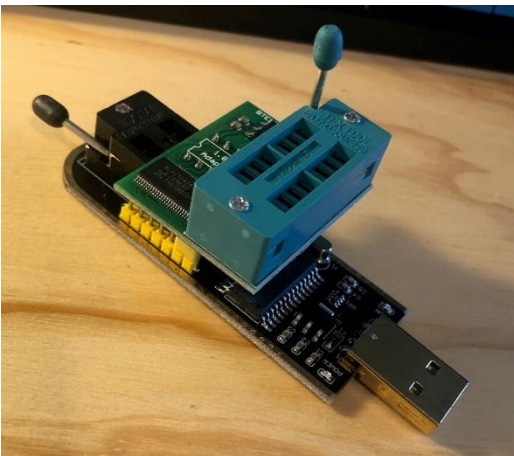
## 2. Follow Unbricking tutorial (<https://wiki.mrchromebox.tech/Unbricking>)

- a. Make sure USB Programmer is configured correctly

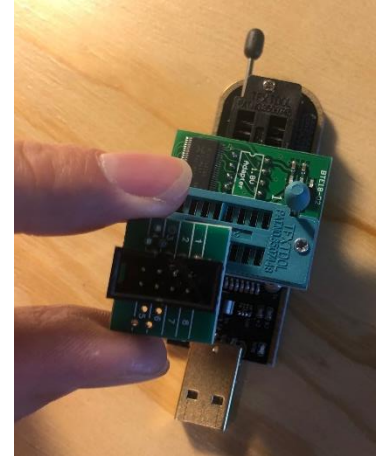
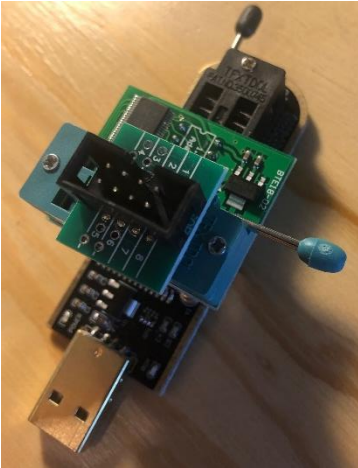
- i. Start with 4 parts from amazon links above (there will be additional parts, but you only need these 4)



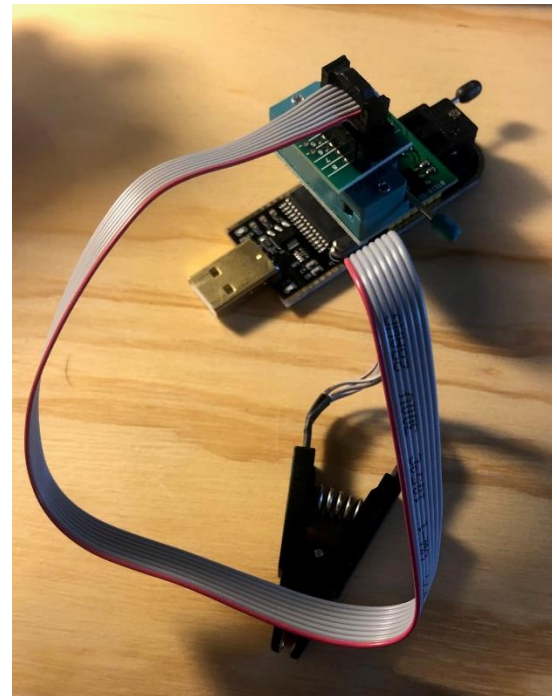
- ii. Start with the USB and the blue “1.8V adapter” (shown to the right). Make sure the black lever on the usb is flipped up, and then insert the pins on the blue adapter into the black slots closest to the USB connector as shown to the left. Then push the black lever down to lock it in place.



- iii. Next, grab the “chip clip” connector piece (shown on the right). Make sure the blue lever is lifted up, and insert the connector into the 1.8V adapter, making sure that pin 1 on the connector lines up with pin 1 on the 1.8V adapter. Then, push the blue lever down to lock the connector in place. It should look like the picture on the left.



- iv. Finally, grab the “Chip clip cable”, and connect it to the connector pins. It only inserts into the connector in one way.
- v. This is the final configuration of the “USB Programmer”. Please note that one of the wires on the chip clip cable is red. This shows you which pin is Pin 1 (this is important later.



- b. Download “Flashrom” software on Ubuntu Computer
  - i. The maker of the “Unbricking tutorial” has made a custom version of flashrom for this specific purpose.
  - ii. Open terminal on the Ubuntu computer
  - iii. Type in the following commands:
  - iv. **“cd ;”**
  - v. **“wget**  
<https://mrchromebox.tech/files/util/flashrom.0602.tar.gz>

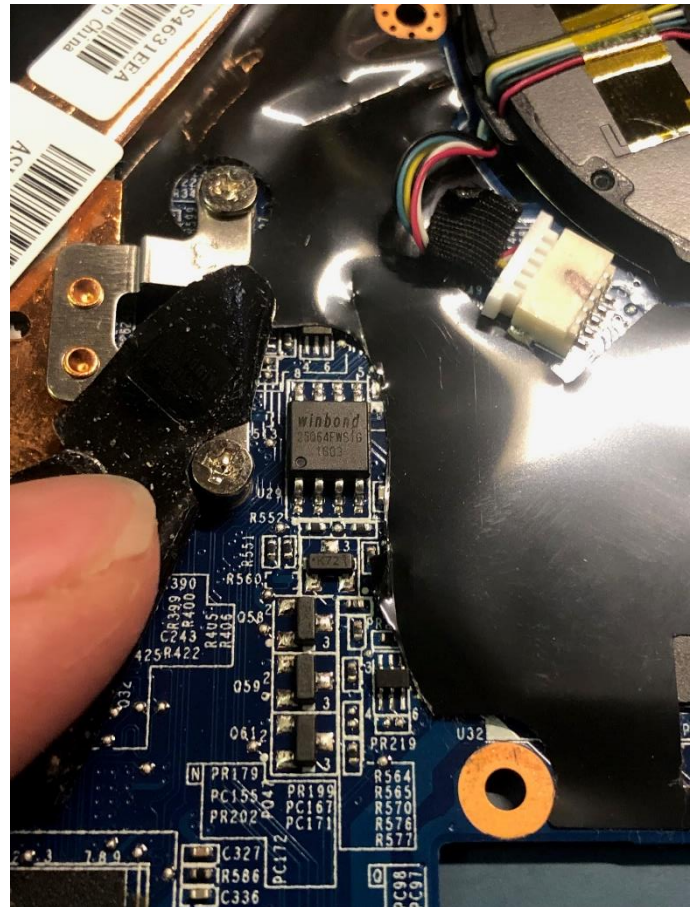


[&& tar -zxf flashrom.0602.tar.gz](#)” (this is all one command. take out the quotes)

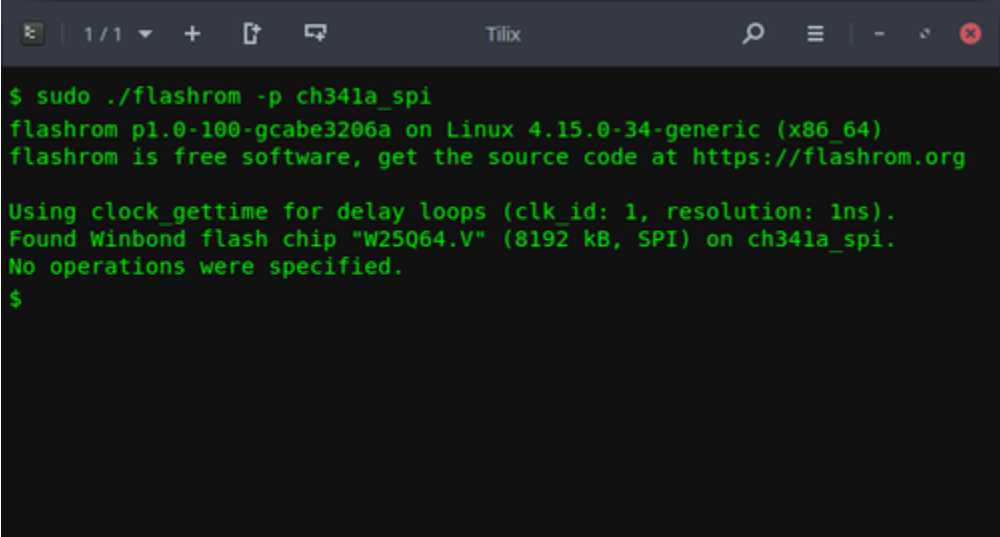
- c. Now you are capable of running the “flashrom” software
  - i. Insert the USB Programmer into the computer. If it was configured correctly, a RED LED will light up brightly to show it has power.
  - ii. You must locate the correct chip on the motherboard you wish to flash. On Lenovo 11e Chromebooks, it is right next to the heatsink+fan, under the black plastic cover. It will say

**“winbond”**. Do not mess with any other chip.

This may cause the device to not boot after re-programming the motherboard.



- iii. The dot in the corner of the “winbond” chip shows you which pin is pin 1.
- iv. Clip the chip clip to the Winbond chip, making sure that pin 1 on the chip is connected correctly to pin 1 of the connector
- d. Return to the computer with “flashrom” installed
  - i. With the connector connected to the chip, type the following command into terminal
    - 1. “sudo ./flashrom -p ch341a\_spi”
  - ii. This should test the connection between the computer and the flash chip on the motherboard. If everything is configured correctly, you will see the following information in the terminal:

A screenshot of a terminal window titled "Tilix". The terminal shows the execution of the command "sudo ./flashrom -p ch341a\_spi". The output is as follows:

```
$ sudo ./flashrom -p ch341a_spi
flashrom p1.0-100-gcabe3206a on Linux 4.15.0-34-generic (x86_64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q64.V" (8192 kB, SPI) on ch341a_spi.
No operations were specified.
$
```

- iii. If this does not show up, retry connecting the clip to the flash chip
- iv. Once the chip is detected, you can proceed with backing up and rewriting the chip
- e. To read/backup the contents of the flash chip, use the following command
  - i. “sudo ./flashrom -p ch341a\_spi -r \*filename\*”
  - ii. Replace \*filename\* with the name of the backup you want to create, just make it something easy to remember
  - iii. The command will take a minute to complete, and then the backup will be stored on the computer
- f. To write to the flash chip, first make sure you have a backup of a “good chip”. This should be from a completely working motherboard

of the same exact device, any other backup will cause this process to not work.

- i. Use the following command to write to the chip:
  - ii. “sudo ./flashrom -p ch341a\_spi -w \*filename\*”
  - iii. Where filename is the name of the “good backup” you created from a completely working motherboard
- g. If all goes well, the terminal will state that it is “writing and verifying” the flash chip
  - i. It is very possible to get an error that “write-protect” is enabled, meaning that it is impossible to write to the chip. Try making sure the screw pad for the RW-screw (Write Protect screw) is clean, and then try flashing again. This fixed my issues on several occasions
- h. Once the flashing process is complete, the device should be able to enter developer mode, and you will be able to assign the motherboard a new serial number as required by the school district.