Author: ballaswag
Date: 9/14/2023

K1/Ingenic Cloner Dumping/Flashing

USE AT YOUR OWN RISK

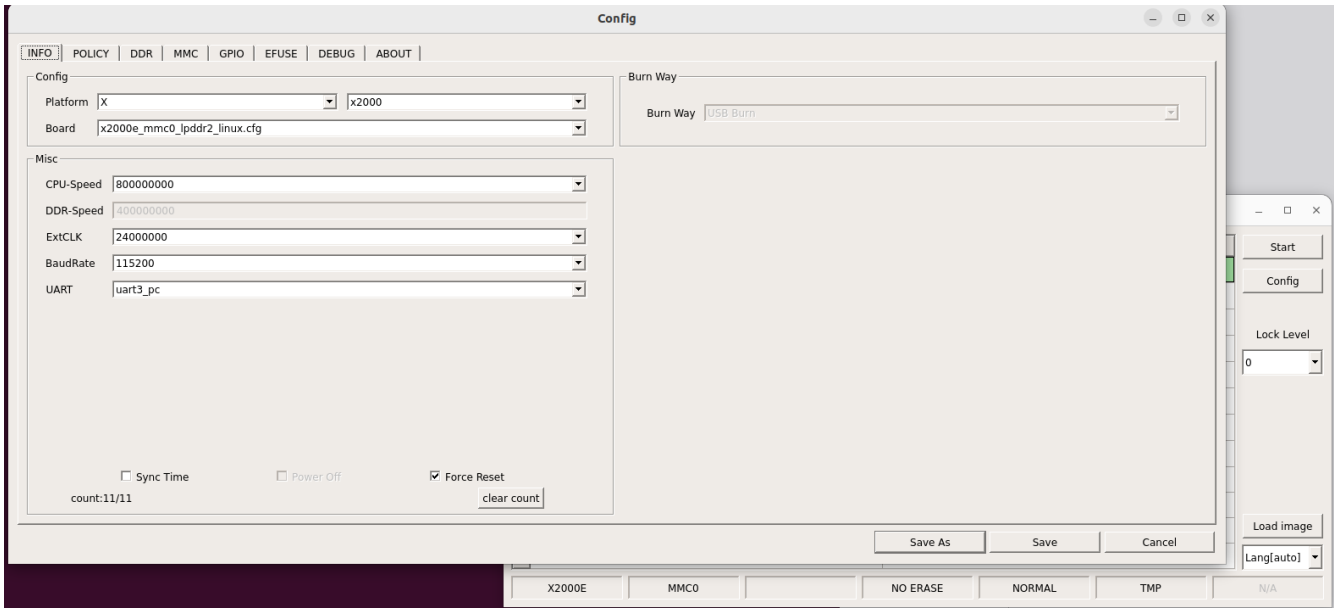
Table of Contents

1. Setting up

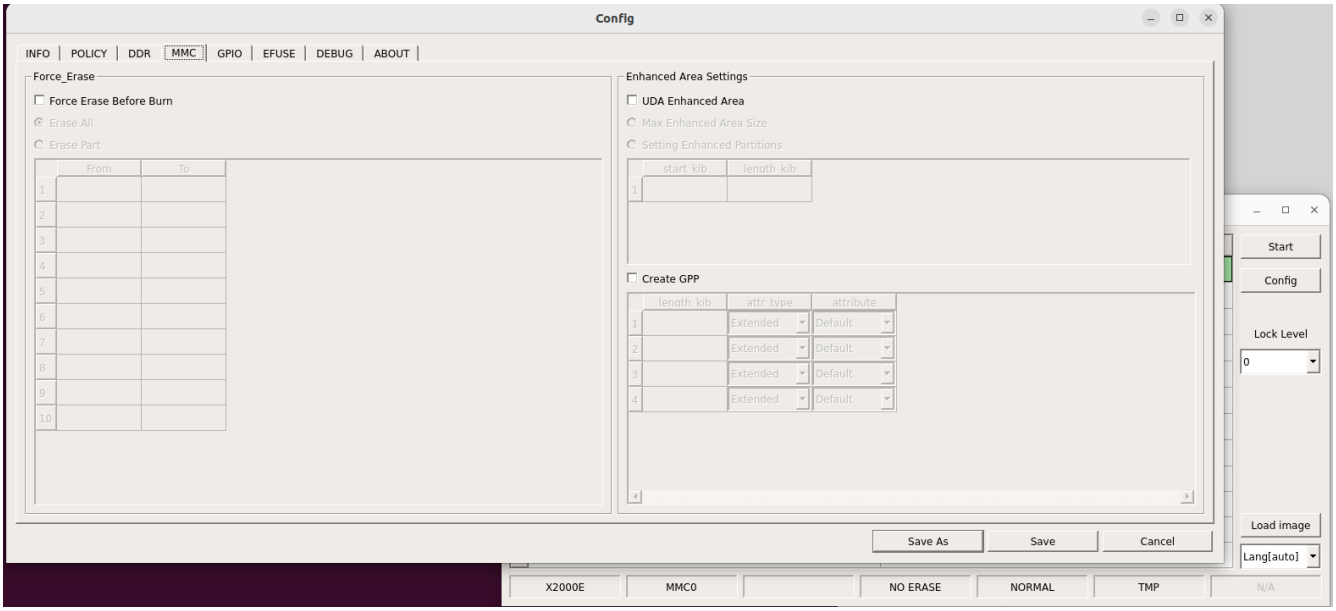
Follow Creality's guide to install the cloner tool (firmware-recovery-tool) via the github. Windows requires drives, linux doesn't. Follow their guide to get your board into USB Mode (hold boot, hold reset while still holding boot, release reset, release boot). In USB boot mode the printer shouldn't boot into the OS, the second blue LED is also dim and not blinking. Once in USB Boot mode USB connected to your printer and devices you can dump and write from the internal storage. Tip: You don't need to power via the printer power, the micoUSB is enough to power the board.

Common setup for the cloner tool as follow. This assume we're not overriding the existing partition layout. The reflash is simply override write the existing partitions. I haven't tried change the partition layout. Feel free to try and report back.

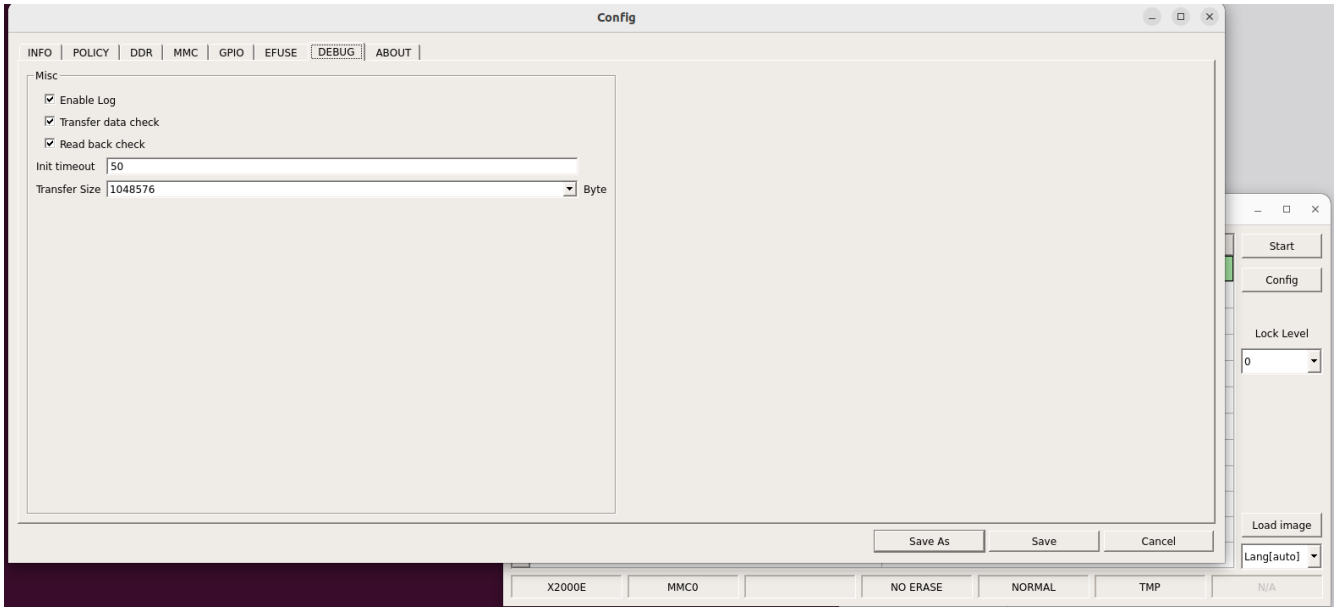
Click Config → INFO → Platform: X (x2000) → Board: x2000e_mmc0_lpddr2_linux.cfg



Click MMC → uncheck Force Erase Before Burn (No entirely sure if this does a full format, I disabled just in case)



DEBUG → Set Transfer Size → highest value (this should help dumping speed).



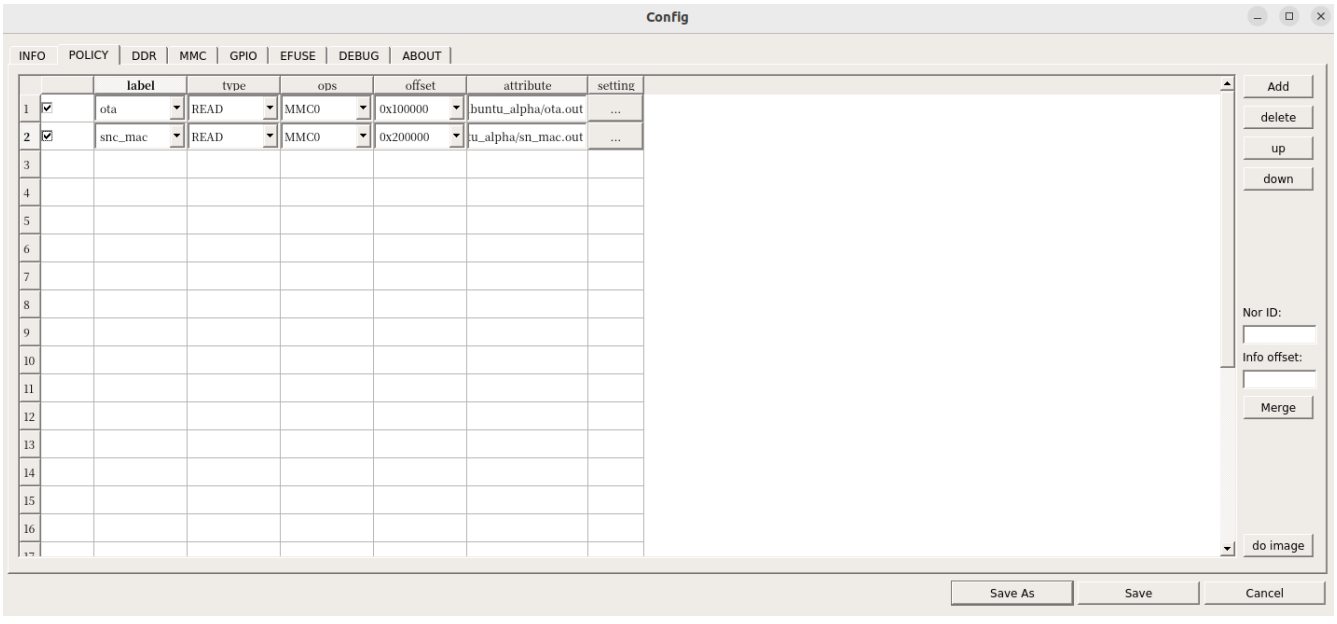
2. Dumping

Click POLICY. With the Board profile we picked, there should be 3 items by default, uboot, kernel, and rootfs. From my digging, the first 1MB of K1 storage should be MBR, EFI partitions. The SPL should be there as well. I was only interested in the OTA partition so the following only demonstrates that, you can extrapolate for other partitions. I deleted all the default items configured by the board configuration via the delete button on the right. Then added my own and named it ota and sn_mac.
I needed sn_mac to make sure all of that 1MB was dumped for ota. When the sn_mac offset was not provided, it didn't dump the whole 1MB of ota. Use the the following settings to dump

OTA
TYPE: READ
OPS: MMC0
OFFSET: 0x100000
ATTRIBUTE: click … and select a filename for this dump.

SN_MAC
TYPE: READ
OPS: MMC0
OFFSET: 0x200000
ATTRIBUTE: click … and select a filename for this dump.

Via fdisk you should see all the partitions' start and sizes. For OTA fdisk shows it starts at 2048 with a sector size of 512. So the offset is 2048 * 512 = 1048576 (hex 0x100000). Click Save, confirm, and click Start. You should get a full progress bar and the outfiles. I didn't not have to do boot/reset button combo to start the flash as mentioned in the Creality docs.



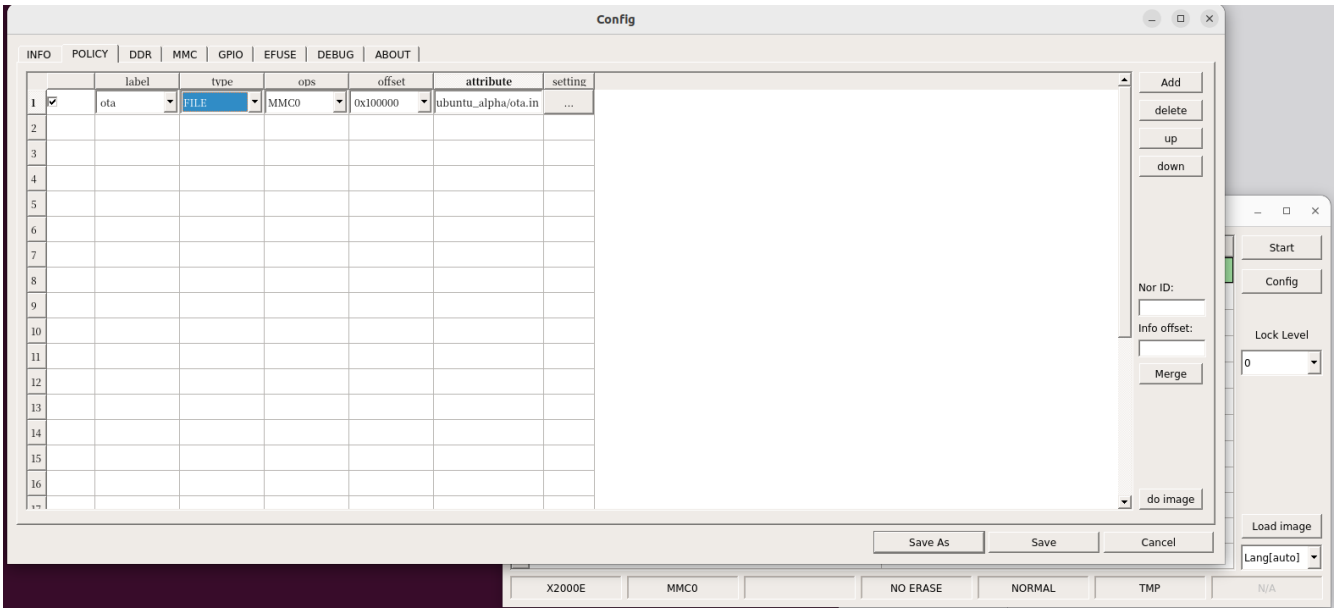
3. Writing

All configurations are the same except the POLICY. Open POLICY, change TYPE: FILE, offset is the same as were partition was read from. ATTRIBUTE is now the file you want to write onto K1. Note: writing as does not need a dummy item to tell where it should stop because it stops after writing all of input.

OTA
TYPE: FILE
OPS: MMC0
OFFSET: 0x100000
ATTRIBUTE: input filename

CAUTION: Make sure your input file is not bigger than the actual partition, if it is then you run the risk of overwriting to other partitions.

Save, confirm, make sure you're in USB Boot Mode, click Start. The progress bar should be green and you have successfully flashed a partition in the K1.



4. Recovering to backup kernel/rootfs

TLDR; Dump and see which kernel your ota is pointing at, write at ota offset with Step 2 of this guide to point at the other ota (this is 11 bytes of data if you really want to optimize).

The original partition layout has partitions: ota, sn_mac, rtos, rtos2, kernel, kernel2, rootfs, rootfs2, userdata. The OTA partition contains a string that tells rtos which kernel to boot, e.g.

ota:kernel
ota:kernel2

If you bricked your board by trashing one of rootfs or kernel, you can set ota to point at the other by dumping and reflashing ota. After you dumped ota, the file should be 1MB, and only 1 byte needs to edited.

Use a binary editor to REPLACE the corresponding byte. “2” here is 0x32. If I want to change this to ota:kernel then I simply edit 0x32 to 0x0a (newline).

If your OTA is pointing at ota:kernel then you want to replace the first 0x0a (newline) with 0x32.

Once you edited the string to point at the alternative kernel, save and verify ota.out is 1048576 bytes (1MB):

```
balla@swag:~/projects/cloner-2.5.37-ubuntu_alpha$ wc -c ota.out
1048576 ota.out
```

Follow Step 2 in this guide to flash the new ota. Once that’s done you should be able to boot if all you have broken were just the kernel/rootfs. Note: if you own kernel/rootfs is on an older firmware, you might see errors on the K1 display. Simply upgrade again to a newer firmware and you should be good to go.

5. Dumping all partitions

Don’t use blindly, check your K1 for its partition layout.

```
uboot(gpt/uboot):     Offset 0x000000000, Length 0x0000100000
ota:                  Offset 0x000100000, Length 0x0000100000
sn_mac:               Offset 0x000200000, Length 0x0000100000
rtos:                 Offset 0x000300000, Length 0x0000400000
rtos2:                Offset 0x000700000, Length 0x0000400000
kernel:               Offset 0x000b00000, Length 0x0000800000
kernel2:              Offset 0x001300000, Length 0x0000800000
rootfs:               Offset 0x001b00000, Length 0x0012c00000
rootfs2:              Offset 0x014700000, Length 0x0012c00000
rootfs_data:          Offset 0x027300000, Length 0x0006400000
userdata:             Offset 0x02d700000, Length 0x01a4a00000

Total disk size:0x00000001d2104200, sectors:0x0000000000e90821
```