

Miyoo 掌機 移植 Linux 系統和模擬器

2018-12-17 20:15

好久不見囉！目前短暫回歸一下，鍵盤俠可能又準備要出動了，對吧！哈哈～～建議大家：應該心存正能量，凡事以感恩的心看待，人生會不一樣～～

RG 掌機由於拿不到程式碼，因此，我私底下做過很多實驗，花了相當多的時間，只是進度多數是倒退，因此，我也沒有發布新的消息，而剛好楊老闆、周哥、文濤大合體，終於讓 Tony 取得源碼(為何一開始不給我呢?)，我想老外會繼續維護相關軟體模擬器，而以 Tony 的能力，Kernel 鐵定會大躍進，因此，大家可以好好期待，只是我個人時間、能力有限，因此我就不插花了！

繼續為開源掌機尋找春天應該是我的本職，哈哈，因此，就讓我當一個領頭羊，再衝一波吧！目前多虧謝工贈我兩台 Miyoo 掌機，稍微研究一下，發現潛力蠻大的，因此，在這即將屆滿一週年(RG 掌機開發)的時候，另一波開源掌機應該也是時候發酵了！說實話，我不確定是否可以真正完成 Linux 系統移植以及模擬器教調，但是，我會盡力～～

Miyoo 掌機：

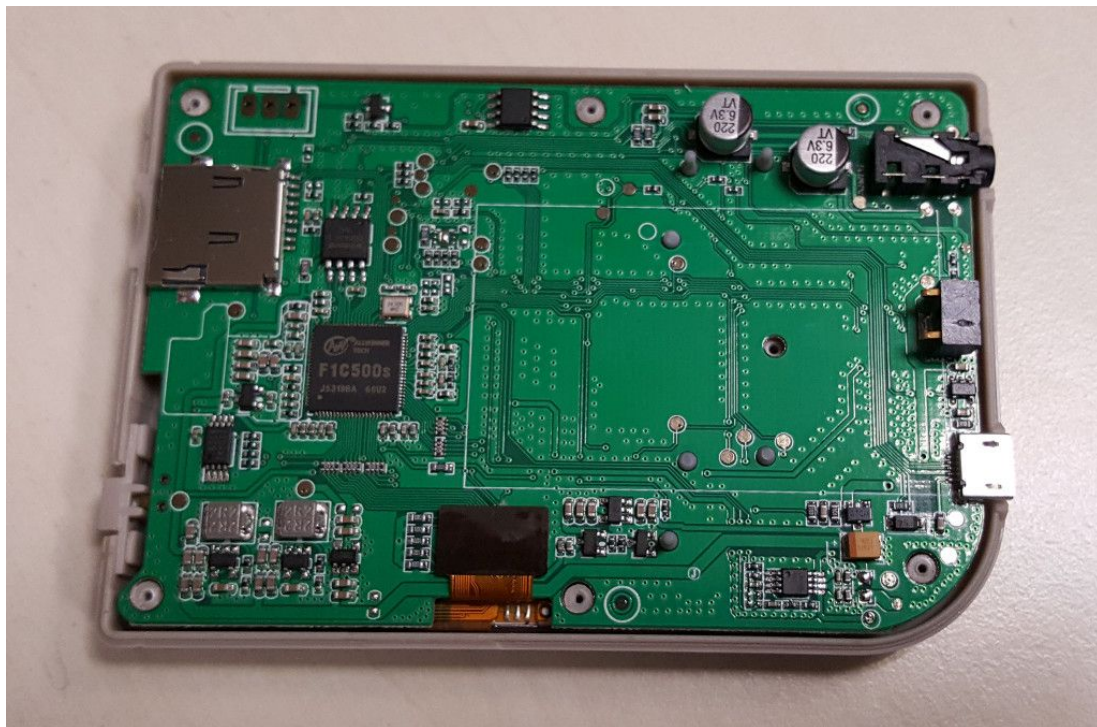
CPU: F1C500s 408MHz~900MHz (荔枝派網站說的...)

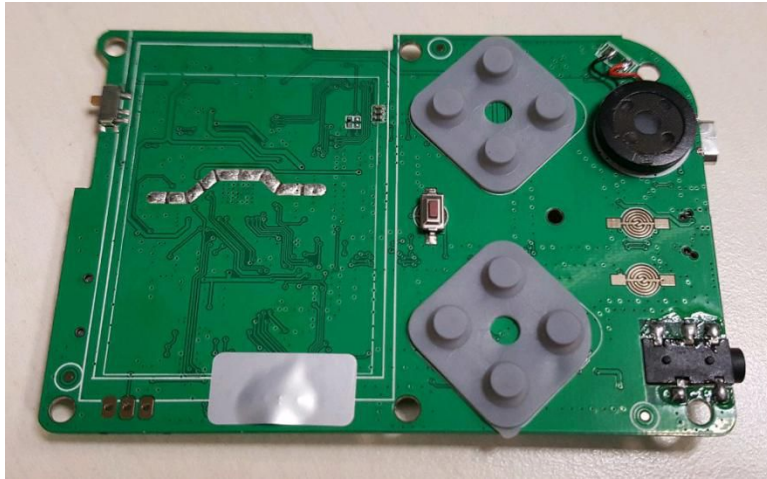
RAM: 32MB

Flash: 16MB SPI Flash

屏幕: LG IPS 2.4" 320x240

基本上，以這個規格來看，頗有 Wiz 掌機的味道～～





2018-12-17 20:45

可視角

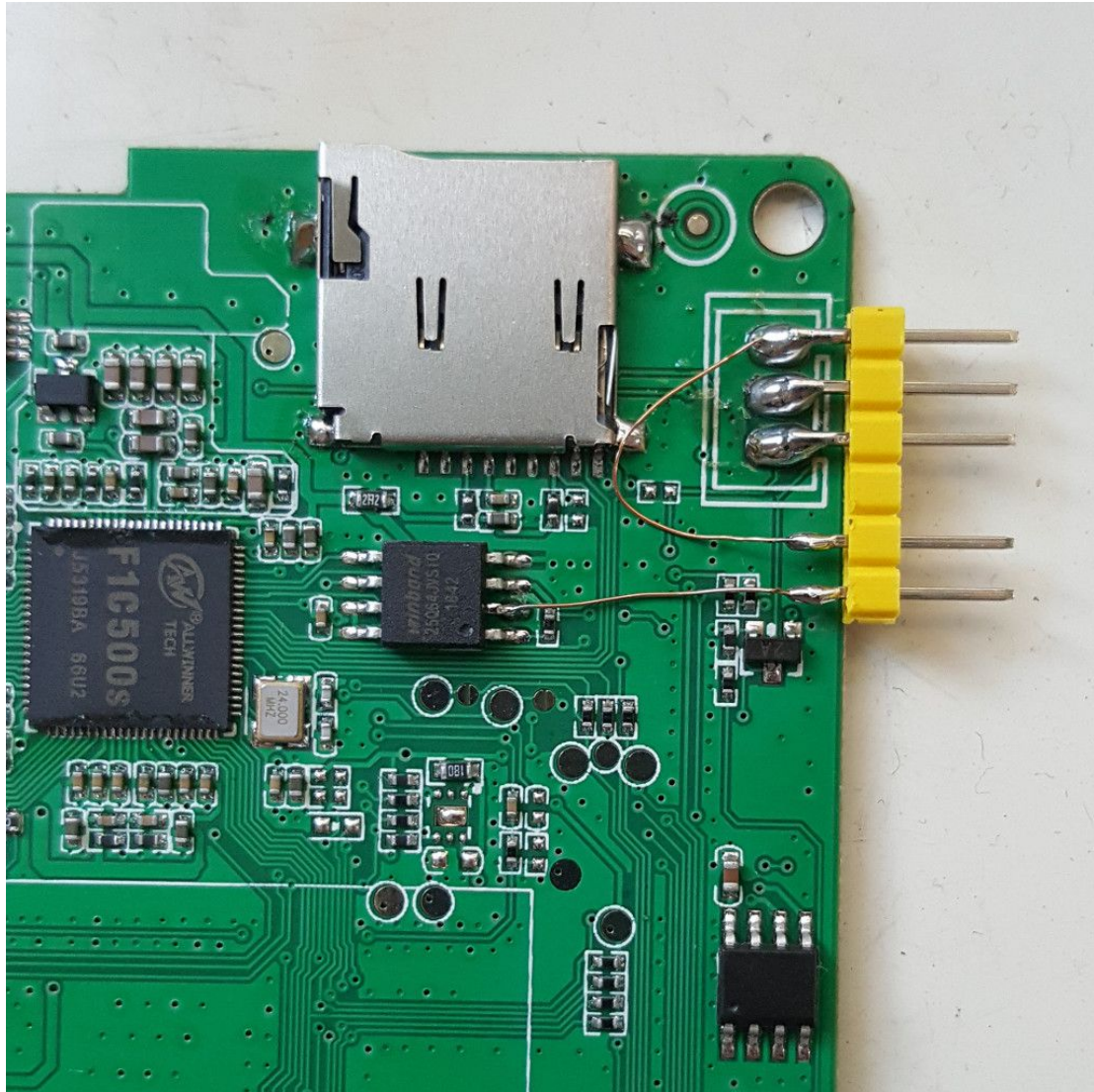






2018-12-18 11:15

UART 除錯腳位(由上而下)GND、UART1 RX、UART1 TX



2018-12-18 11:34

基本上，Miyoo 掌機可以看成是荔枝派 Nano，因此，熱愛嵌入式設計的同好，可以自己焊接 UART，一起研究～

依據目前得到的資料，USB 連接的是 UART0 TX、DBG_MS、DBG_CK 腳位，所以當開機進入 sunxi-fel 模式時，連接 USB 到 PC 時，使用者會發現 PC 並沒有偵測到任何 USB 裝置，解決的是則是焊接跳線，如下方式。

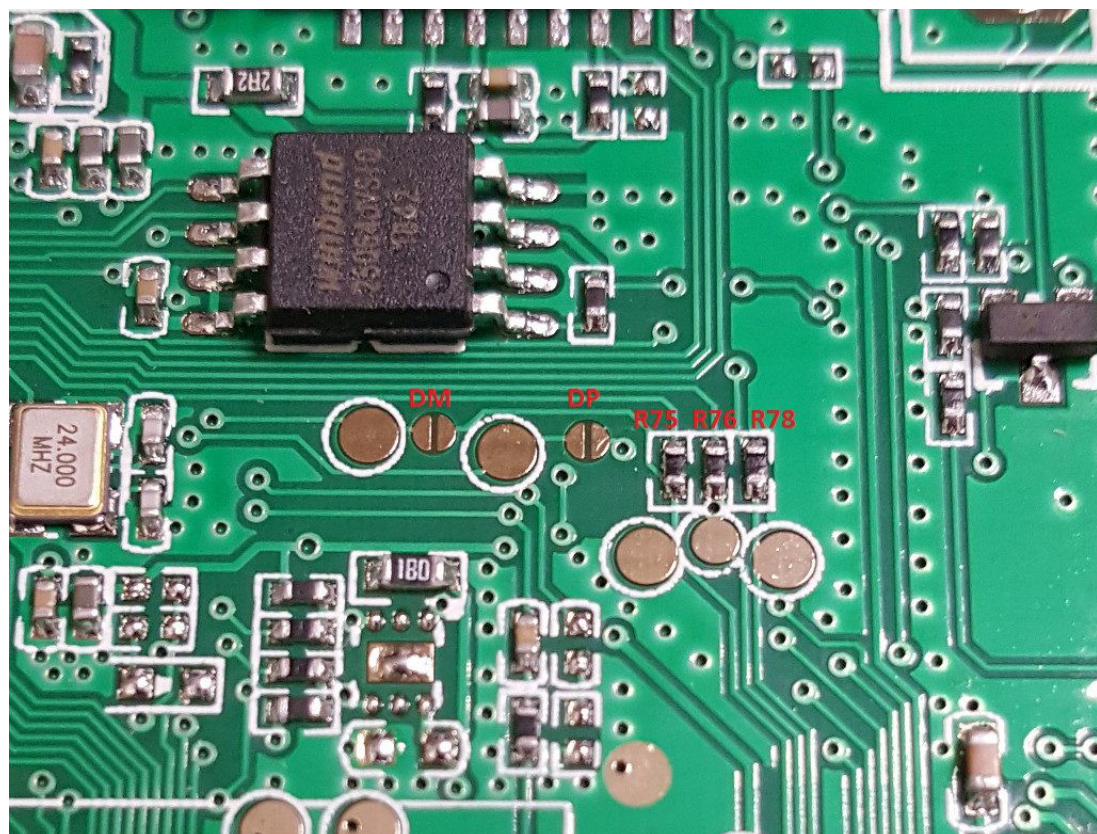
1. 短路 DM、DP 焊點
2. 移除 R75、R76、R78 電阻



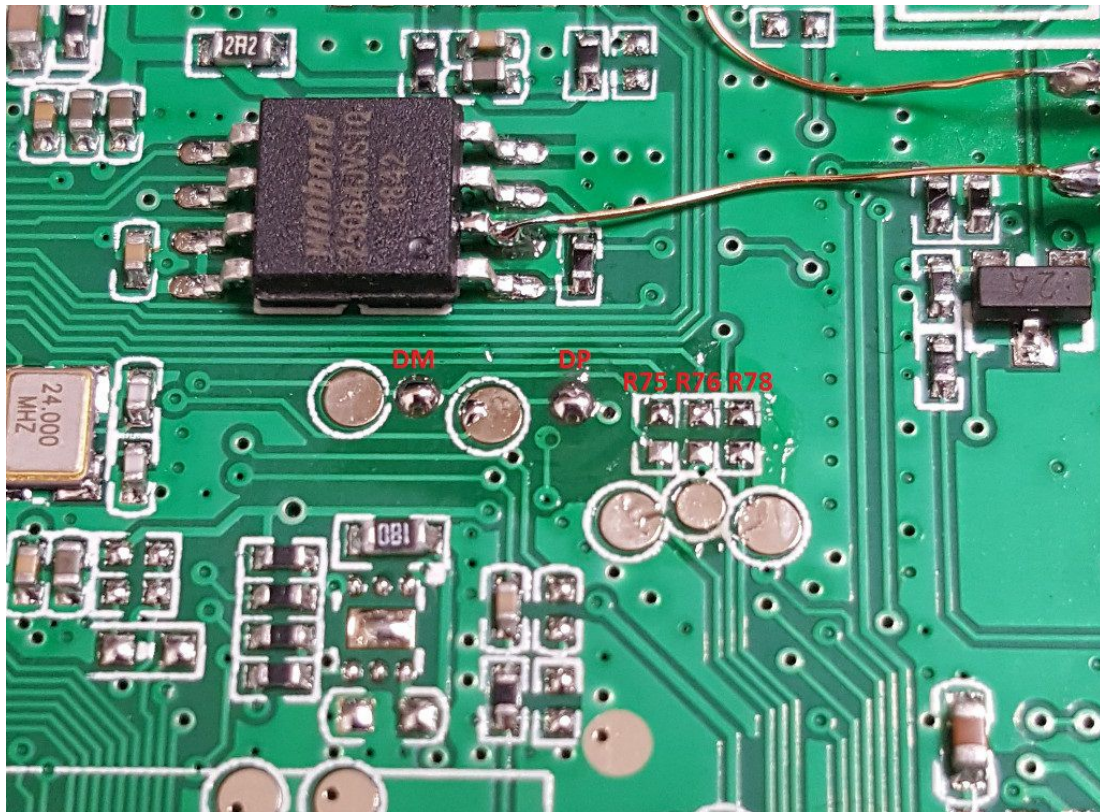
diai008: USB 連接的是 UART0 TX、DBG_MS、DBG_CK 腳位，所以當開機進入 sunxi-fel 模式時，連接 USB 到 PC 時，使用者會發現 PC 並沒有偵測到任何 USB 裝置，解決的是則是焊接跳線，如下方式。
这话是说 USB 还是要改动 不改读不出来 那么原机卡刷没速度 usb 读不出 不硬改刷机没戏吧 那么第一批吃螃蟹的个个都会硬改！

2019-01-01 17:30 回复

原本



修改後



2018-12-18 11:38

已經可以偵測到 sunxi-fel 模式

```
$ sudo dmesg -c
```

```
[520592.048623] usb 4-1.2.4.4: new full-speed USB device number 64 using ehci-pci
```

```
[520592.177876] usb 4-1.2.4.4: New USB device found, idVendor=1f3a, idProduct=efe8
```

```
[520592.177882] usb 4-1.2.4.4: New USB device strings: Mfr=0, Product=0, SerialNumber=0
```

```
$ lsusb
```

```
Bus 004 Device 064: ID 1f3a:efe8 Onda (unverified) V972 tablet in flashing mode
```

2018-12-18 18:08

UBoot 已經移植完成

https://steward-fu.github.io/website/handheld/miyoo/build_uboot.htm

Kernel 4.14.0 目前已經可以開機進入 Console

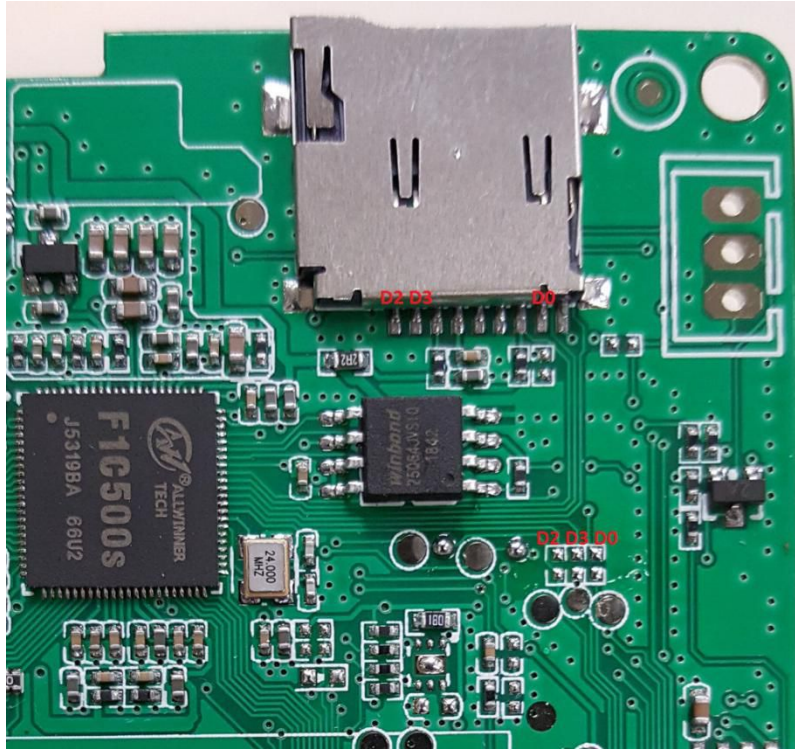
https://steward-fu.github.io/website/handheld/miyoo/build_kernel.htm

目前正在移植顯示驅動以及 framebuffer

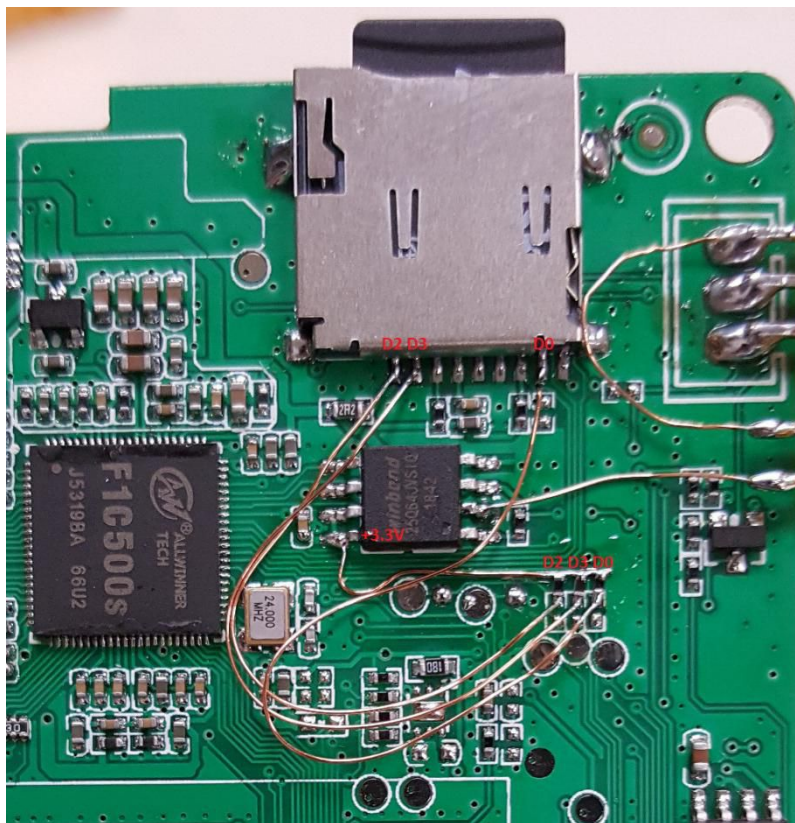
2018-12-18 19:51

依據目前得到的資料，MicroSD 使用的是 1bit 傳輸方式，使用者可以使用如下方式，將其改成 4bits 傳輸方式。

1. D2、D3、D0 拉提昇電阻(1K)
2. 焊接 D2、D3、D0 跳線



修復後



2018-12-19 10:29

依據資料手冊，這顆 CPU 最大的極限應該是 2.6GHz，不過，如果可以超頻到 800MHz，已經可以媲美 Wiz 掌機

The PLL output= $(24\text{MHz} * N * K) / (M * P)$.

The PLL output is for the CPU Clock.

Note: The PLL output clock must be in the range of 200MHz~2.6GHz.

Its default is 408MHz.

另外，Display Engine 支援硬體拉伸，這是一個相當不錯的加速功能

2.4.1. Display Engine

- Support four layers overlay, each layer size up to 2048x2048 pixels
- Support Alpha blending/color key
- Support multi-format input formats
 - 1/2/4/8/16/32 bpp color
 - YUV444/YUV422/YUV420/YUV411
- Support hardware cursor
- Ultra-Scaling engine
 - 4-tap scale filter in horizontal and vertical direction
 - Input and output size up to 1280x720 pixels
 - Support write-back to memory function
- Support de-interlacing function for Standard-definition video

2018-12-19 17:42

雖然我有點懶的理"鍵盤俠"，不過如果可以讓他們了解更多外面的世界，那也算是功德一件！

終於等到多位鍵盤俠出動，也終究暴露自己的缺點，RAM 以及 Flash 大小，對於系統的影響是有限的，尤其是這種小系統，除非 RAM 小到一定程度，如：1KB，至於 Flash，看看 RG 掌機就知道，系統是放在 MicroSD，而且讀取速度更是快於 SPI Flash，最後，24MHz 是外部石英振盪器，你有看過 1GHz 的外部石英振盪器嗎？想也知道，一定是使用 PLL 相鎖迴路達成。

人要學會正向看事情，凡事以正向思考，對你有益無害～



maxzhou88: 哈哈 😄，司徒莫怪有些人，記得在网上看過講 J20 飛機氣動和隱身設計的，被噴子噴得頭頭是道，好像他們比設計飛機的專家都懂設計

2018-12-19 23:01 回复

maxzhou88: 對於一個不那么複雜的系統來說，我倒認為 16MB 的 SPI flash 還大了點，它只是用來存放 u-boot 的鏡像和某些配置參數的讀寫保存，由於是多重 boot，說不定 SPI 中的資源數據還是壓縮的呢，16 MB 夠啦

2018-12-19 23:05 回复

maxzhou88: u-boot 被從 SPI 加載到片內 sram 後，SPI 就基本沒用了，頂多再用來存放一些配置參數，如屏的亮度啊，等等。

2018-12-19 23:10 回复

maxzhou88: u-boot 再把 OS 從 SD 卡（也可能是 SPI）中加載到 32MB 的 SDRAM，再啟動內核運行，32 MB 也不算小。

2018-12-19 23:13 回复

maxzhou88: 至於模擬器和遊戲 ROM 也在这 SDRAM 中運行，不要以為一個 GBA 的 ROM 就可能是 32MB 了，它那是採用虛擬內存方式來加載 ROM 的，把 SD 卡的一部分當交換區使用，又不是一次性把 ROM 全部加載進來。如果系統精簡的話，我看 16MB 的 SDRAM 也夠用

2018-12-19 23:19 回复

maxzhou88: 至於 24MHz 的晶振，那是可以用 PLL 倍頻/分頻的，1GHz 的晶振就算有，那也貴死人啊



貼吧用戶_5VU48Z1: 感謝大家的支持，也感謝精神股東周哥的說明，另外，後來我才發現目前 Miyoo 掌機的 Flash 是 8MB，不是我一開始說的 16MB，哈～

2018-12-20 10:13 回复

2018-12-20 16:50

感謝 開源掌機群：春華秋實 的 P 圖



2018-12-21 15:20

曝光的包裝盒



2018-12-23 16:46

說實話，這個移植真的蠻難的，因為全志的晶片資料，對於 Register 的描述相當不足，基本上，細節幾乎都是不說明的(君正的說明文件也差不多是這樣)，這個也是為何我比較偏愛 TI(德儀)晶片的另一個原因～

移植不成功，或許比較正常，移植成功的話，只能說老天保佑！目前使用 GPIO 做 LCD 顯示控制，已經可以點亮這個 IPS 屏，不過，顯示效能相當不好！總之，這是一個好的開始，只能繼續努力了！



2018-12-26 19:34

更新一下目前狀況:

1. 研究了一個禮拜後(參考很多資料以及程式碼), 還是不知道如何使用 TCON 驅動 i80 介面的屏, 如果有 Register 初始化程式碼(Allwinner 系列都可以)的人, 麻煩提供給我一下
2. 目前使用 GPIO 刷屏(沒有使用 DMA)時, 60FPS 時, CPU 使用率高達 85%
3. 幸好, 所有屏的排線都是掛在 Port-D 底下, 因此, 目前打算使用 DMA 做刷屏動作, 缺點則是需要兩倍 Buffer(WR=0, WR=1)

2018-12-27 20:07

剛剛測試了一下 Miyoo 超頻:

768MHz (可以穩定跑 PCE 模擬器), 816MHz (執行 PCE 模擬器後, 馬上掛點), 1GHz (開機就掛點), 所以基本上應該是很接近 Wiz 的速度, .Wiz 我可以穩定跑 800MHz, 不過目前預設出廠的我猜是預設值: 408MHz



2019-01-01 10:34

感謝開源掌機群(群友: win2next)協助設計的 logo







NxHope

A retrogame system by Situ



2019-01-02 07:49

老外製作的網站:

<https://jutleys.wixsite.com/bittboynew4u>

2019-01-02 19:00

是時候宣佈壞消息的時候了!

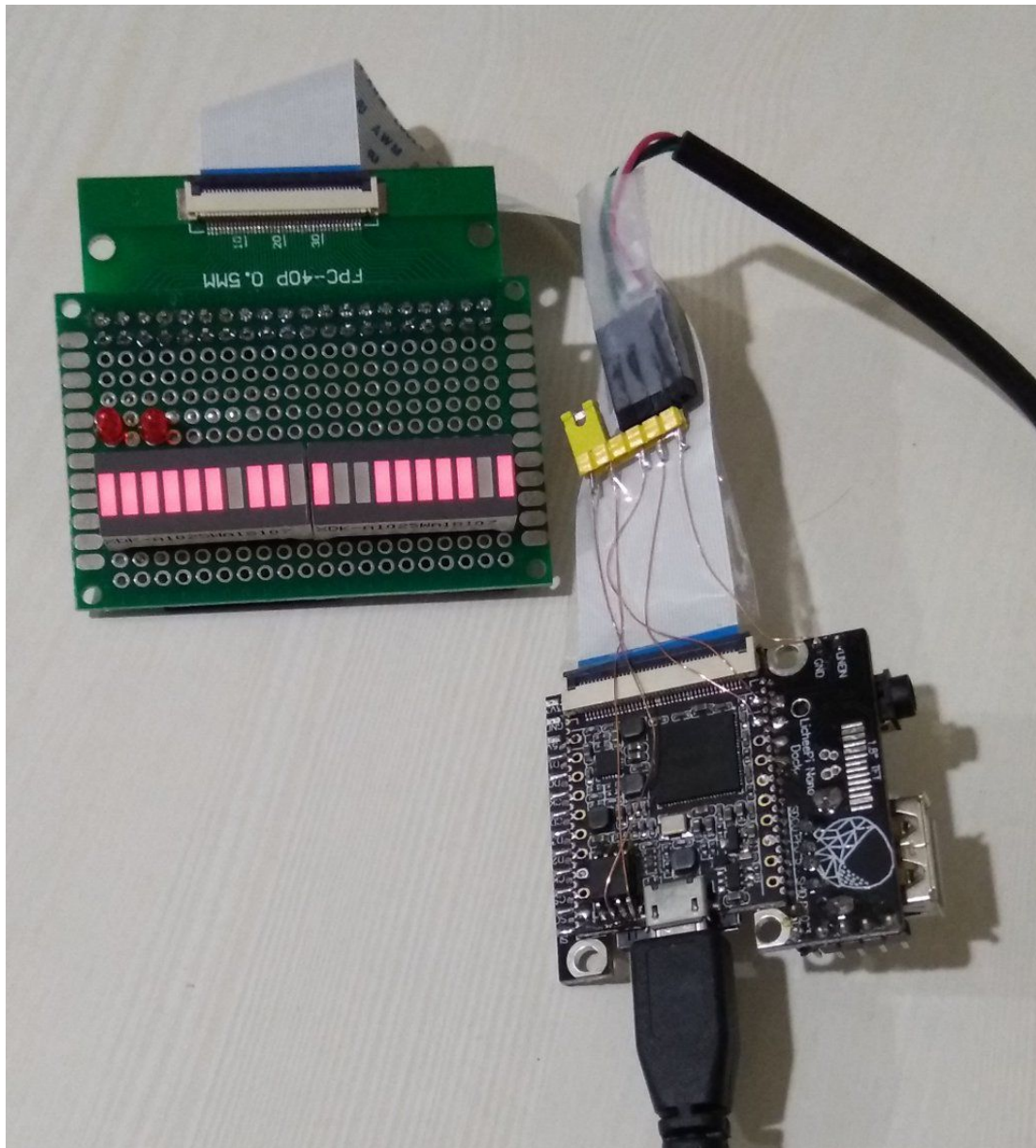
經過漫長的測試開發，目前 F1C100s/F1C500s 的 LCD i80(CPU 介面)應該是有問題的，包含 Serial 和 Parallel 兩個都是有問題，這個可能是 CPU 內部的 LCD 模組問題。

我花了相當多的時間閱讀相關代碼和資料，我甚至使用逆向工程逆向全志 Melis 系統的 Library(bsp_de.a, bsp_lcd.a)，盡我所能測試所有組合，包含 PLL 全部重新初始化，順序組合測試，無奈天不從人願，我還是無法控制 CPU 介面的輸出，即時我移植 XBOOT 系統，利用它的相關初始化，CPU 屏依舊不給面子，因此，我合理懷疑 F1C100s/F1C500s 的 LCD i80(CPU 介面)應該是有問題的！難怪市面上，看不到相關應用～



2019-01-02 19:03

因此，可以使用的硬體加速就只剩下 DMA + GPIO，可惜的是，還是無法成功使用 DMA 控制 GPIO 輸出，即使使用荔枝派 F1C100s 測試，還是一樣失敗



2019-01-02 19:32

接著如何繼續呢？目前只能換 RGB 介面的屏，剛好可以使用黑妹的 2.4" IPS 320*240 屏，期待改造可以早日完成，這段時間就先晒圖一下好了，哈哈





2019-01-02 20:12

另外如果沒有超頻，預設 408MHz 的話，GPIO 刷屏會浪費 85% 的 CPU 資源，而如果超頻到 768MHz 的話，GPIO 刷屏會浪費 60% 的 CPU 資源

2019-01-03 18:55

首先感謝 XBOOT 作者以及 XBOOT 群友(PowerAVR..等)的協助，給予司徒一些 DMA 參考程式(程式作者：dianguai14, 2013-06-28)~，目前可以使用 DMA 驅動 GPIO 並且點亮這個 CPU 屏，這種方式也是目前看來可以使用硬體加速的方式，缺點就是需要編碼(WR=0, WR=1)並且需要 8 倍的緩衝區，不過使用空間換取時間的方式如果可行，這也是一個好的方式！

不過目前 F1C100S Linux Kernel 的 DMA 中斷無法使用，我正在 debug 中！目前下列清單的工作做完後，你們可以不須硬體改機就可以刷機(但是 MicroSD 僅支援 1bit 模式，硬改後是 4bits)：

1. 移植 UBoot McroSD 開機載入 zImage
2. 移植 Linux Kernel DMA Interrupt
3. 移植 Linux Kernel 聲音驅動

CPU 使用率方面，因為 DMA 中斷尚未完成，因此，還無法知道，有新消息再跟你們說~

2019-01-03 19:23

另外，分享一下 DMA 驅動 GPIO(PD DATA)的程式，以後或許有人需要：

```
setbits(iomm.ccm + BUS_CLK_GATING_REG0, (1 << 6));
setbits(iomm.ccm + BUS_SOFT_RST_REG0, (1 << 6));
writel(0x00000000, iomm.dma + DMA_INT_CTRL_REG);
writel(0xffffffff, iomm.dma + DMA_INT_STA_REG);
writel(0x02310211, iomm.dma + NDMA0_CFG_REG);
writel(0x00000002, iomm.dma + DMA_INT_CTRL_REG);
writel(par->lram_phys, iomm.dma + NDMA0_SRC_ADR_REG);
writel(SUNXI_GPIO_BASE + PD_DATA, iomm.dma + NDMA0_DES_ADR_REG);
// testing gpio
uint32_t i;
uint32_t *s = par->lram_virt;
for(i=0; i<22; i++){
s[0] = ~(1 << i);
writel(4, iomm.dma + NDMA0_BYTE_CNT_REG);
writel(2, iomm.dma + DMA_INT_STA_REG);
setbits(iomm.dma + NDMA0_CFG_REG, (1 << 31));
mdelay(3000);
}
```

重點：

1. DRQ 都要設定成 DRQ_SDRAM (Kernel Mode)
2. Destination Type: I/O Mode
3. 指標都必須使用 Physical Address

2019-01-04 12:26

感謝"剃頭"協助製作 LR 按鍵



2019-01-04 14:48

重大改進!!

修復 DMA IRQ 問題後，目前使用 DDMA + GPIO 改進效能，掃屏時，CPU 使用率已經降至 0%(原本是 60%)，因此，整顆 CPU 的效能可以完全發揮了～

順道分享一下 DDMA 設置：相較於 NDMA，DDMA 需要設定 DRQ 成 AHB_MEMORY(For I/O Mode)

目前剩下如下工作：

1. uboot 從 microsd 載入 zImage 開機
2. Linux Kernel 聲音驅動

Miyoo 刷機參考網址：https://debugdump.com/t_444.html

2019-01-04 16:20

目前會不定期釋出測試刷機包，但是，僅能給硬改過的人刷(因為 UBoot 從 MicroSD 開機目前有點難搞)，因為需要修復 USB 以及 SPI 短路電路，如果不想要改機，大約要等到快過年前，才可以使用 MicroSD 刷機(預計過年前會修復 uboot 問題)，目前硬體改機，我教導"明暗"以及"剃頭"，手藝至少是我覺得不錯的，他們聯繫方式如下：

剃頭 QQ: 78534131

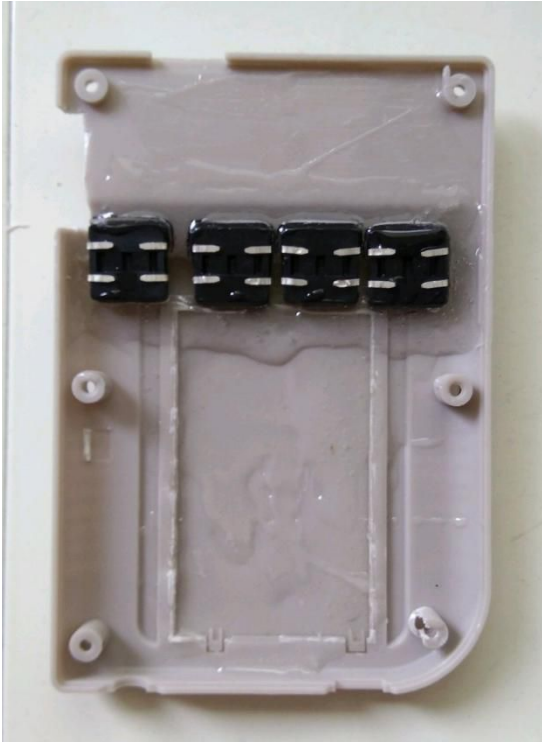
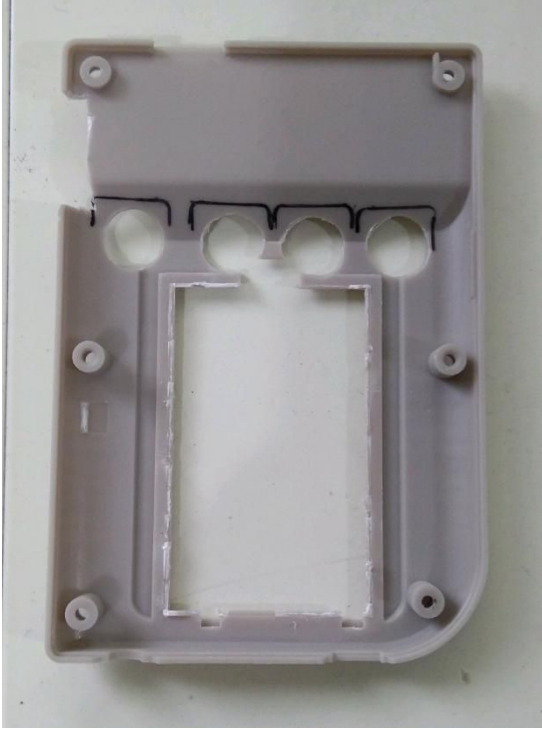
明暗 QQ: 626468274

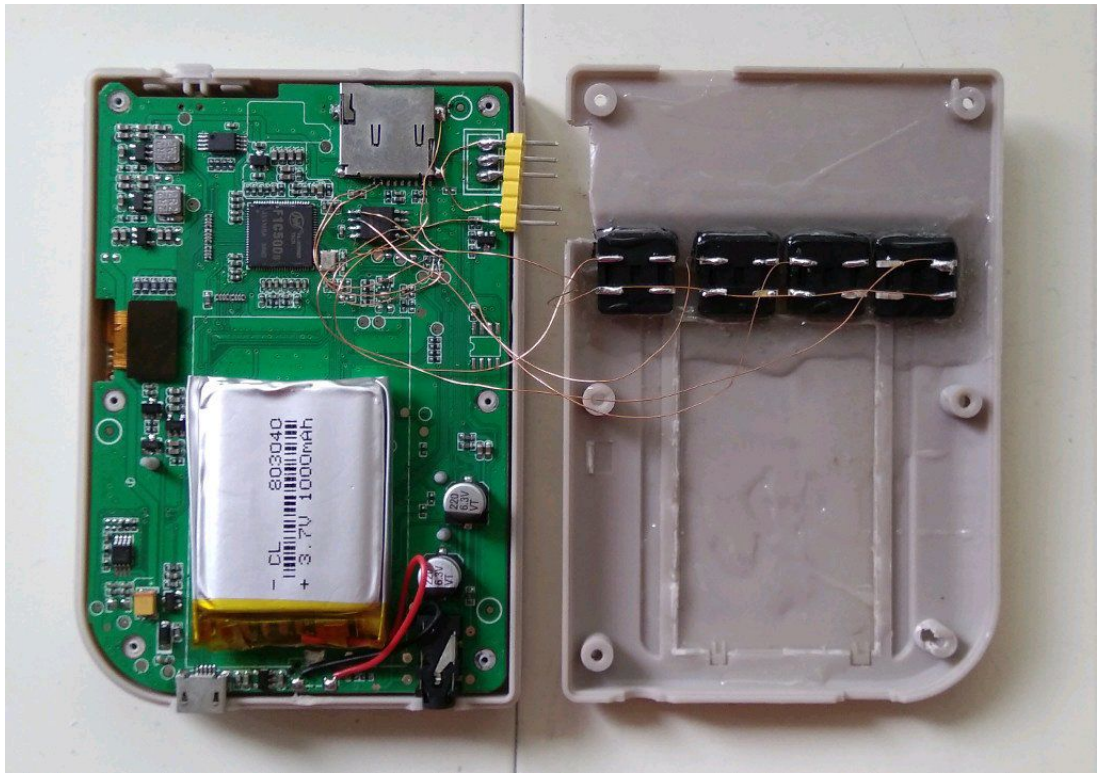
新增 LR 按鍵: 30 元(不包郵)

新增 LR 按鍵 + 固定電池(1500mA~1800mA): 50 元(不包郵)

至於 LR 形狀就看自己喜愛哪個，自己往上翻圖！我個人很建議自己硬改，因為難度不大，不用換屏！另外，這一次我還是一樣沒有拿廠商任何費用，義務免費開發開源系統給大家玩樂，希望大家熱愛開源掌機～







2019-01-05 15:53

最後確定電路為(按鍵都不須提昇電阻，使用 IC 內部的提昇電阻):

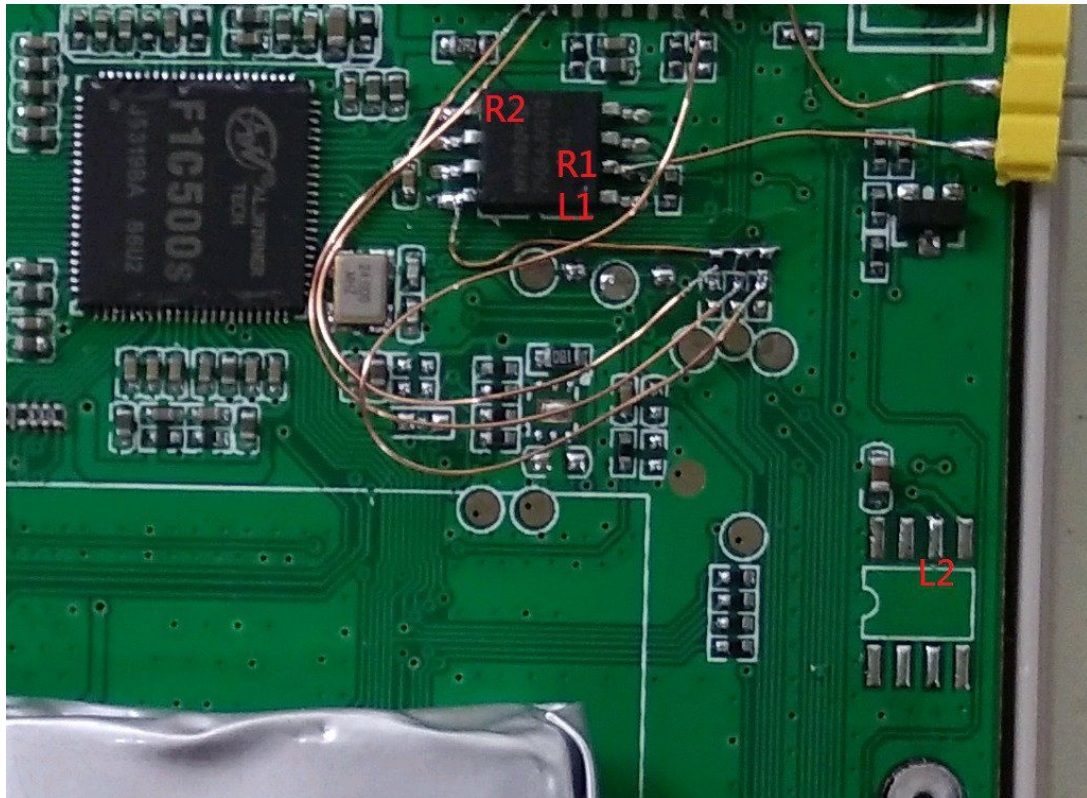
L1 接到 SPI IC 第 1 腳

R1 接到 SPI IC 第 2 腳

R2 接到 SPI IC 第 5 腳

L2 接到加密 IC 第 6 腳





2019-01-05 15:53

刷機步驟:

1. 關機
2. 取出 MicroSD
3. 按住 R1 按鍵
4. 插入 USB 到 PC
5. 開機
6. 放開 R1 按鍵
7. 執行如下命令燒錄

2019-01-06 07:02

jutleys 提供的比較圖



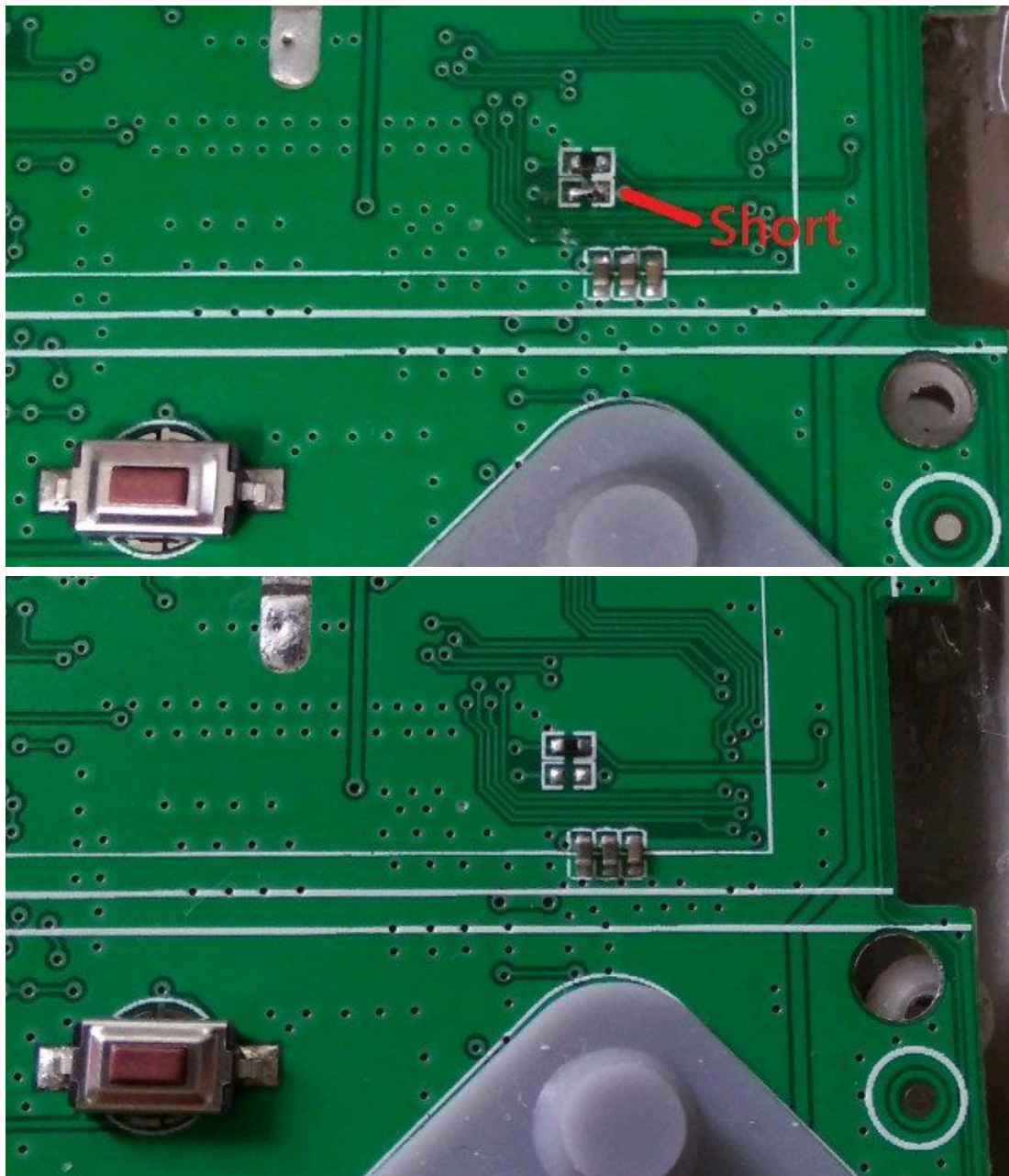


2019-01-06 22:34

請有修改硬體的人，再做一個動作，修復 ScreenTearing 硬體，因為目前屏的 TE 腳位尚未拉到 CPU，步驟如下說明（只要短路即可）。

目前 Miyoo 掌機如果你有感覺閃屏的話，那你的眼睛可以察覺 ScreenTearing 問題，如果沒有，那真的不錯，你的容錯能力還不錯！

目前我使用 DDMA+GPIO 掃屏，掃 60 個畫面，總共需要 1.02 秒（超頻到 768MHz），因此，DMA 是剛好符合 60fps 的需求，但是，意思就是沒有多餘能力處理其它問題，而針對 8080 介面的屏，最大問題就是 ScreenTearing(如下圖)，因此，目前打算使用 GPIO 中斷處理，只是尚未完成，但是需要先準備硬體，因此，請先做焊接動作！





2019-01-06 22:40

GBC 模擬器測試版 釋出測試:

https://www.github.com/steward-fu/miyoo_dev

刷機步驟:

1. 先改機 USB 以及 SDCard
2. 拔出 SDCard(8GB)
3. 透過 USB 刷入 miyoo_hwmod_spi_20190106_1.bin
4. 解壓縮 cfw_testing_20190106.rar 並透過電腦燒錄到 SDCard(8GB)
5. 開機後三秒再插入 SDCard

2019-01-06 22:43

目前尚未解決問題：

1. UBoot 透過 SDCard 開機
2. 聲音問題
3. 背光亮度調整
4. 電池資訊測量
5. 閃屏問題

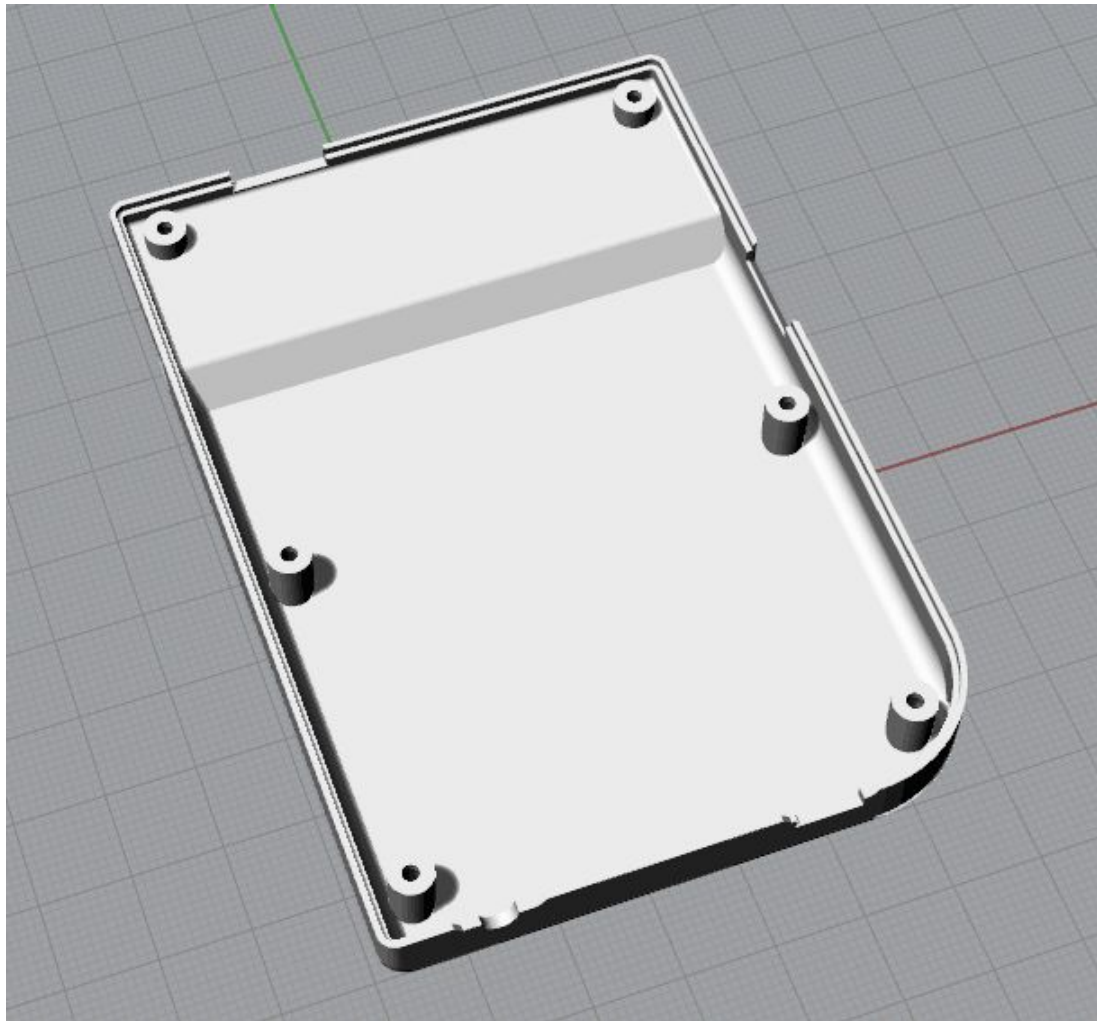
2019-01-07 12:13

目前 GBC 尚未完全優化完，畫面 60FPS，CPU 768MHz 使用率 95%，大家可以測試一下

2019-01-09 20:48



恶化之心 发个图片看看



贴吧用户_5VU48Z1: 太好了，如果這個外殼可以厚一點點，除了手感會更好，也可以加大電池，當然可以畫上 LR 會更好，相當期待這個背蓋：)

2019-01-10 12:27

目前使用 TE 可以讓 Screen Tearing 問題減緩，因為使用 DMA+GPIO 的方式本身就有些限制，加上 DMA 速度沒預期中快速，因此，這個方式預計是最佳的 DMA 方式！

或許很多人在意聲音移植以及模擬器速度，但是，個人比較在意基本功，尤其是顯示問題，因為懷舊老饕可以察覺這些細節，也這個原因，多數機器並無法滿足需求，當然這也是工欲善其事，必先利其器的道理！

下兩張圖是沒有 TE 協助的結果，斜紋是 CPU 跟屏的同步問題，上下紋是應用程式跟驅動程式的同步問題，斜紋部分，因為不是固定位置，所以畫面會抖動，上下紋倒是可以利用 double buffer 解決



2019-01-10 12:29

如下圖是使用 TE 的結果，可以看出斜紋位置固定了，因此，只有這兩個斜紋位置會不舒服，而不是原本的整個畫面



2019-01-11 08:36

被吞了....

FCEUX 測試版已經釋出，SPI 和 SDCard 都要更新，在原 [GitHub](#)，日期請用 20190110





2019-01-12 11:25

會有純軟刷 因為老外也在問我

聲音驅動如果用完 就會開始用那部分

2019-01-17 18:29

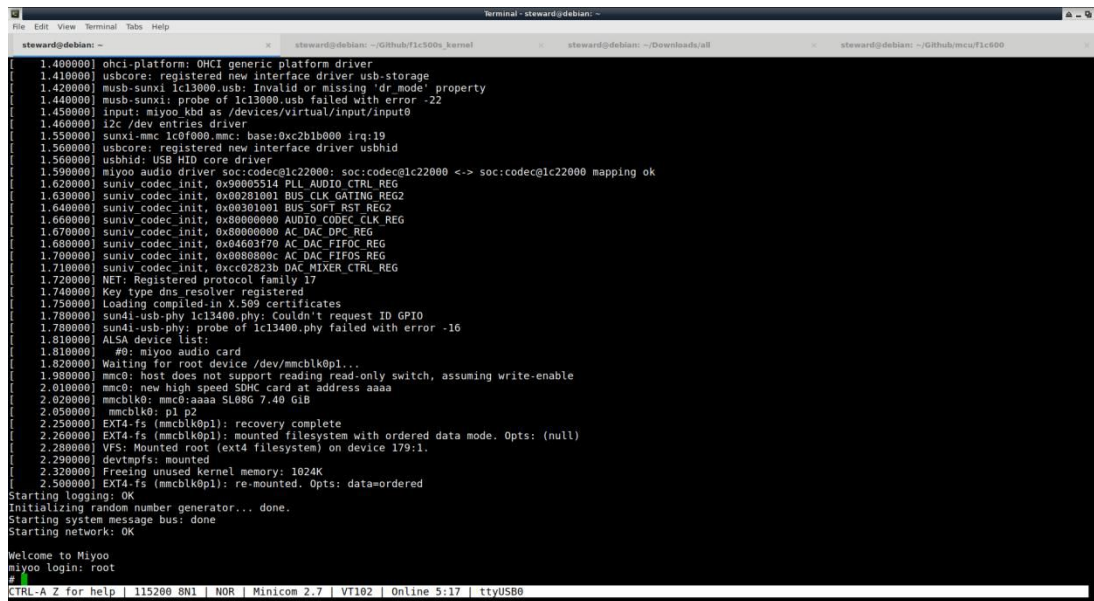
好久沒有更新了，剛好趁一個空檔說明一下目前進度！

相較於 framebuffer 顯示驅動，ALSA audio codec 音效驅動規劃的比較複雜一點點，加上之前都沒有好好研究過，因此，花了一些時間 study 一下，寫了一張 Miyoo 音效卡，這部分問題不大，畢竟夠多資訊可以參考！

接著就是對接 f1C500s audio codec 的實作，很遺憾的是，目前 audio codec 初始化後，只能聽見喇叭的嗡嗡聲音，DAC 的部分就是無法輸出，即使使用 Register 寫值或者 DMA 操作都是一樣！

這部分需要再研究看看，畢竟 Datasheet 裡面的資訊相當有限，參考的程式也是相當缺乏，只是很好奇的是，逆向一下原廠燒錄檔案，並沒有發現操作 audio codec 寫的動作，這部分可能需要再詳細看一下，因為那部分是最具參考的 Binary！

最後要感謝一些協助司徒的朋友、代理商以及廠商，提供一些參考的程式，如果你手上也有 F1C100s audio codec 程式，也歡迎提供給司徒，因為我有設定一個日期，在這個時間前沒有把聲音搞出來的話，會確定放生 Miyoo 掌機，畢竟我已經盡我最大努力來讓這台掌機變更好，既然天不從人願，那也只能如願！



```
steward@debian: ~
[ 1.400000] ohci-platform: OHCI generic platform driver
[ 1.410000] usbcore: registered new interface driver usb-storage
[ 1.420000] musb-sunxi: ic13000.usb: Invalid or missing 'dr_mode' property
[ 1.440000] musb-sunxi: probe of ic13000.usb failed with error -22
[ 1.450000] input: miyoo_kbd as /devices/virtual/input/input0
[ 1.460000] I2C /dev entries driver
[ 1.550000] sunxi-mmc: ic109000.mmc: base=0xc2b1b000 irq:19
[ 1.560000] usbcore: registered new interface driver usbhid
[ 1.560000] usbhid: USB HID core driver
[ 1.590000] miyoo audio driver soc:codecc@1c22000: soc:codecc@1c22000 mapping ok
[ 1.620000] suniv_codec_init, 0x9000514 PLL_AUDIO_CTRL_REG
[ 1.630000] suniv_codec_init, 0x00281001 BUS_CLK_GATING_REG2
[ 1.640000] suniv_codec_init, 0x00301001 BUS_SOFT_RST_REG2
[ 1.660000] suniv_codec_init, 0x00800000 AUDIO_CODEC_CLK_REG
[ 1.670000] suniv_codec_init, 0x80000000 AC_DAC_DPC_REG
[ 1.680000] suniv_codec_init, 0x04603f70 AC_DAC_FIFOS_REG
[ 1.700000] suniv_codec_init, 0x0080000c AC_DAC_FIFOS_REG
[ 1.710000] suniv_codec_init, 0xccc2b230 DAC_MIXER_CTRL_REG
[ 1.720000] NET: Registered protocol family 17
[ 1.740000] Key type dns_resolver registered
[ 1.750000] Loading compiled-in X.509 certificates
[ 1.780000] sun4i-usb-phy: ic13400.phy: couldn't request ID GPIO
[ 1.780000] sun4i-usb-phy: probe of ic13400.phy failed with error -16
[ 1.810000] ALSA device list:
[ 1.810000]  #0: miyoo audio card
[ 1.820000] Waiting for root device /dev/mmcblk0p1...
[ 1.900000] mmc0: host does not support reading read-only switch, assuming write-enable
[ 2.010000] mmc0: new high speed SDHC card at address aaaa
[ 2.020000] mmcblk0: mmc0:aaaa SL08G 7.40 GiB
[ 2.050000] mmcblk0: p1 p2
[ 2.250000] EXT4-fs (mmcblk0p1): recovery complete
[ 2.260000] EXT4-fs (mmcblk0p1): mounted filesystem with ordered data mode. Opts: (null)
[ 2.280000] VFS: Mounted root (ext4 filesystem) on device 179:1.
[ 2.290000] devtmpfs: mounted
[ 2.320000] Freeing unused kernel memory: 1024K
[ 2.500000] EXT4-fs (mmcblk0p1): re-mounted. Opts: data=ordered
Starting logging: OK
Initializing random number generator... done.
Starting system message bus: done
Starting network: OK

Welcome to Miyoo
miyoo login: root
#
CTRL-A Z for help | 115200 8N1 | NOR | Microm 2.7 | VT102 | Online 5:17 | ttyUSB0
```

2019-01-17 23:13

天底下最恐怖的事情就是，叫天，天回應了！
還是機器聽到要被放生，趕緊動作了???

最後發現是自己的測試問題...

目前確定 Miyoo 可以發出聲音了！

目前需要解決的問題如下：

1. 從 SDCard 卡開機
2. 音效持續完善
3. 屏背光驅動
4. 電池驅動

由於需要幾個大咖(tony, sauce, pingflood, jutley)幫忙協助移植以及測試，因此，從 SDCard 卡開機會是接下來的首要任務，這樣分工會比較有效率，完成後，不改機的也可以一起協助測試，有改機的人，優勢之後就會看出來的！

[hockeypeng](#): 请司徒大大专门开一个改机说明，1bit 升 4bits 的电阻是要 1K 的还是 4.7K 的有定论吗？

2019-1-18 10:29 回复

[贴吧用户_5VU48Z1](#): 因為 I/O 的 PCB 拉線，改機後就不等長，因此，1K 或 4.7k 應該是沒有太大差別，如果有示波器可以量一下，看一下波形

2019-1-18 10:46 回复

2019-01-18 08:31

這個應該就是洗三溫暖的概念！

2019-01-18 20:31

重大進展！CPU 屏介面已經可以運作顯示！

首先感謝 XBOOT 群（木青）的 CPU 屏程式、XBOOT 作者、汗血小黑馬以及台灣高手的協助！

目前確定 CPU 屏介面可以在 F1C100s 以及 F1C500s 上面跑，這個進展幫助相當大，因為原本需要的編碼步驟已經可以省略，速度可以不需要白白浪費掉，記憶體也不需要浪費在編碼上，模擬器可以專心在模擬事情上！因此，有機會把 Screen Tearing 解決掉！

F1C500s TCON 支援 DMA 以及 DE(Backend)硬體加速，但是 DMA 確定無法使用，最後使用 DE(Backend) 加速，直接 Mapping Display Buffer 到 DE Register 使用，如果結合 DE(Frontend)就可以做硬體拉伸，速度到時候可以再次提升，就像 Tony 幫 RetroGame Plus 完成 IPU 的意思一樣，GCW0 則是原本驅動就支援，所以速度表現都不錯！

人旺的時候就是要趕快去買彩卷，如果之後好幾天沒消息的話，我應該是去享受榮華富貴了！

2019-01-21 15:24

沒錯，我沒有中彩卷，因為那天太晚去買了，店家已經關門了，所以只好繼續研究 Miyoo！

壞消息！

經測試後發現，如果沒有改 MicroSD 4bits 模式，Miyoo 目前無法從 MicroSD 開機！

因此，假如不改 MicroSD 4bits 模式，最少需求是改 USB 焊點，這樣至少可以從 USB 刷機！

這也是今天釋出測試版刷機包(給不改機的機器)才發現的問題。



SD-娇娘内核: 司徒，除了改 USB 焊点还要改一个 R1 按键才能最小刷机吧，或者用两根线拖出来，手动短路才能进入刷机模式吧

2019-1-22 01:03 回复

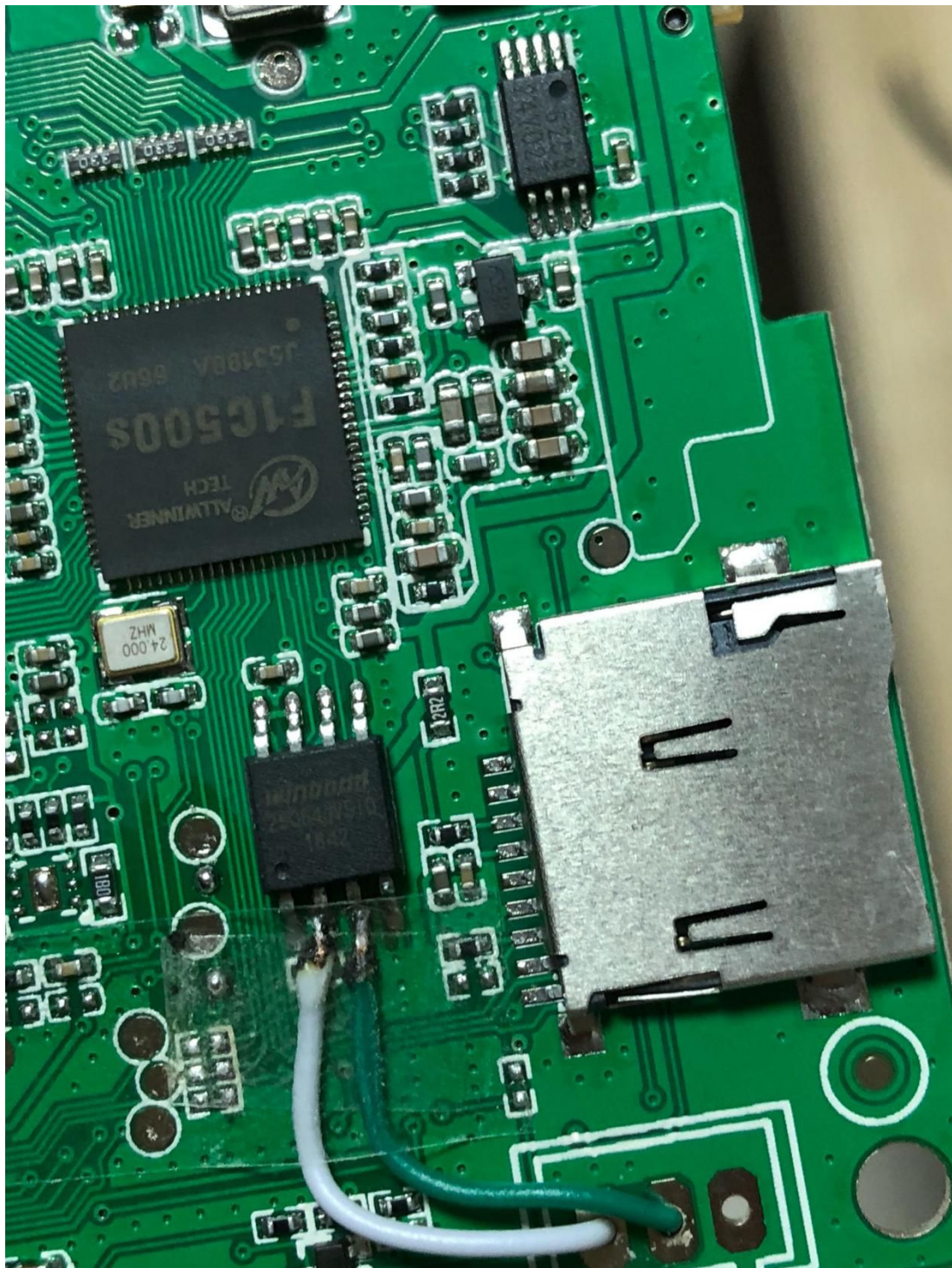
贴吧用户_5VU48Z1: 回复 华敏乐 :嗯嗯，感謝提醒，刷機時，需要先短路 SPI Flash 的 2 和 3 腳位，因此，拉個開關也可以，或者拉到 R 按鍵，方便燒錄使用也可以

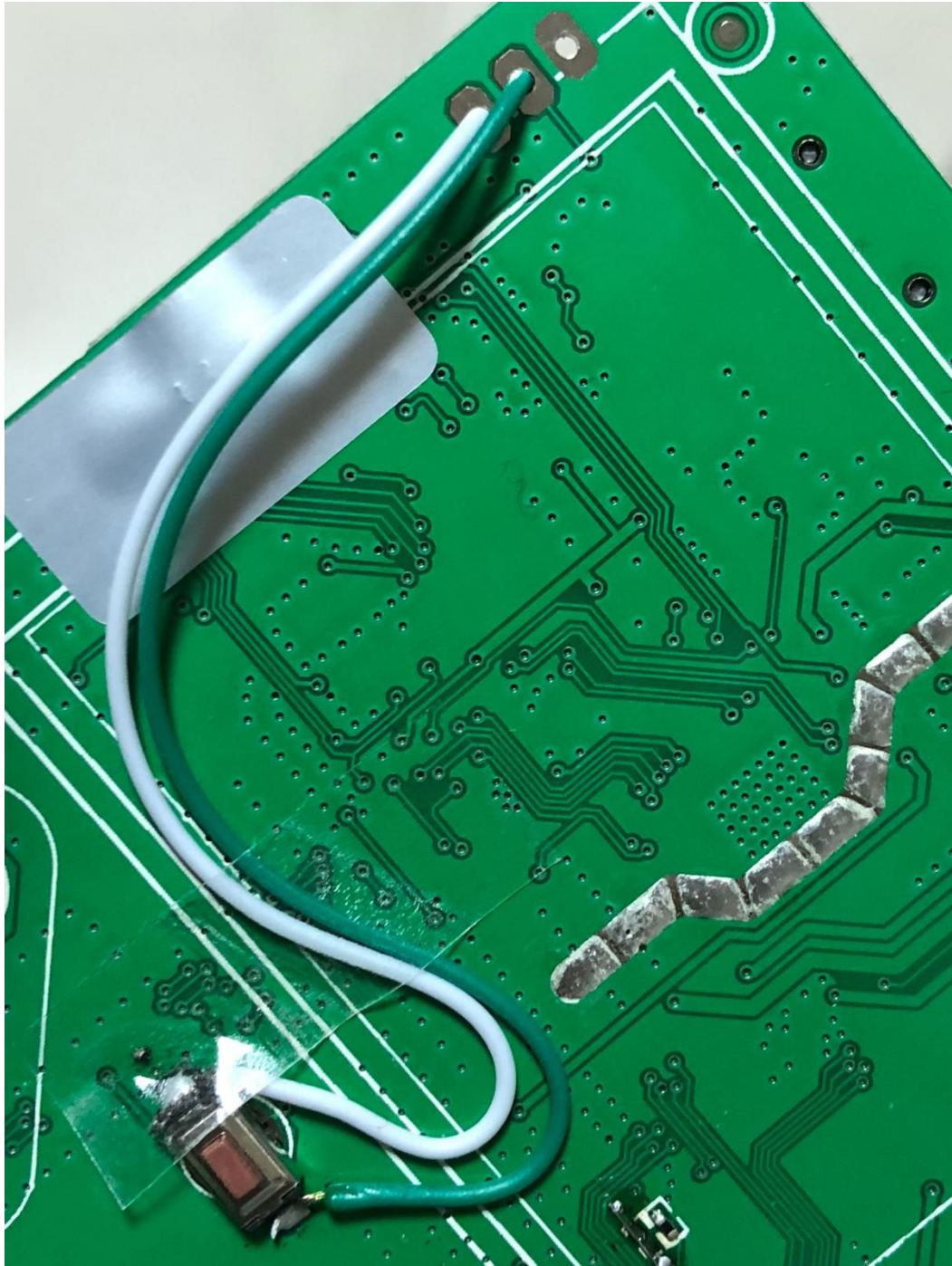
2019-1-22 07:40 回复

2019-01-22 20:21



SD-娇娘内核: 我直接这么飞的, 也能用





贴吧用户_5VU48Z1: 你這樣菜單鍵可以在模擬器裡面用？因為這個按鍵是矩陣掃描其中一顆，如果固定拉 High，應該有一排按鍵無法使用才是

2019-1-22 20:40 回复

SD-娇娘内核: 临时刷机用，然后断开。设备识别了，驱动打不上

2019-1-22 20:45 回复

SD-娇娘内核: 刷机后断开哦

2019-1-22 21:00 回复

贴吧用户_5VU48Z1: 焊個 LR 按鍵，然後接到 R 按鍵，這樣刷機就很方便

2019-1-23 08:57 回复

2019-01-23 20:10

剛好趁一個空檔來更新一下進度！

目前測試 **framebuffer** 驅動，發現一樣的燒錄檔案，在 F1C100s 跟 F1C500s 跑出來的結果不一樣，F1C500s 無法觸發 TCON 中斷，然後 TCON DEBE 會 hang up(離開 framebuffer probe 後)，root cause 目前暫時查不出來，不過經過測試，發現可以使用 **delay loaded module** 方式解決，掃屏速度已經算是目前最好的方式了！這個 **workaround** 方式還算是可以接受，因此，**framebuffer** 驅動短期確定不會在修改了。

驅動問題必須先解決，不然移植模擬器意義不大，相關問題如下：

1. 從 MicroSD 開機：目前沒有改 4bitS 確定無法作到，因此這個項目目前先擱置。
2. 音效驅動：目前正在執行中...
3. 顯示驅動：目前已經使用 **workaround** 方式解決。
4. 閃屏問題：暫時解不掉，先擱置。
5. 背光驅動：XBOOT 已經完成，將會參考其代碼。
6. 電池驅動：XBOOT 已經完成，將會參考其代碼。

聲音驅動完成後，預計釋出 Beta OS 供測試，但是，必須先做 USB 修復的動作，因為要更新 SPI Flash

贴吧用户_5VU48Z1: 可以，預計釋出 1bit 和 4bit 刷機包

2019-1-23 21:13 回复



sss888888sss: 回复 司徒_Steward :可以用编程器将刷机包刷入 SPI Flash 吗?

2019-1-24 01:30 回复

贴吧用户_5VU48Z1: 编程器是指 spi 燒錄機嗎?

2019-1-24 07:33 回复

sss888888sss: 应该是的

2019-1-24 09:30 回复

贴吧用户_5VU48Z1: 如果你打算量產，用編程器可以的，如果不是，那你需要飛六根線到編程器，應該不會比 usb 燒錄方便，當然，只要可以把系統燒錄到 spi flash，任何作法都可以

2019-1-24 09:39 回复

sss888888sss: 有免焊接拆裝的夾具，直接夾住焊在电路板上的芯片引出飞线到编程器

2019-1-24 09:49 回复

fac5438: 还是 usb 方便

2019-1-25 10:10 回复

2019-01-25 19:37

每天更新一下進度看似一件好事，但是常常更新遇到的問題，就有點悲歌了～
不過這些資訊至少可以留給後人參考，還是值得！

由於目前荔枝派 Nano 的 Kernel 尚未支援 drivers/DMA，所以，無法在 audio codec 裡面使用 pcm dma engine，於是，只好硬幹，在 audio codec 裡面實做 DMA 中斷以及初始化，但是，悲劇又來了，DMA 中斷無法觸發，真是...

最後，老天爺再度出手，睡夢中夢到，在 device tree 把 codec interrupt 換到 TCNO0 測試，竟然可以運作了！因此，開機時，利用 DMA 中斷可以實做出一個類似電話的聲響～

一切看似一帆風順，最終還是敗在 ALSA library，模擬器一跑，出現如下 underrun 問題：

```
ALSA lib pcm.c:8323:(snd_pcm_recover) underrun occurred
```

說實話，真的有點搞不懂，pcm_writel 觸發 codec 的 pointer()，我也已經實做，DMA 中斷會更新 ring buffer pointer，pointer() 會再度更新給 alsa library 中間層，就是不知道問題出在哪？

因此，只好使用最簡單的 speaker-test 測試，發現 framerate 好像有點問題，但是，F1C100s 可以調的參數跟這個 framerate 似乎沒有關聯，因此，目前似乎又卡在 audio codec driver....

```
# speaker-test -c 2
speaker-test 1.1.5
Playback device is default
Stream parameters are 48000Hz, S16_LE, 2 channels
Using 16 octaves of pink noise
Rate set to 48000Hz (requested 48000Hz)
Buffer size range from 2048 to 262144
Period size range from 1024 to 2048
Using max buffer size 262144
Periods = 4
was set period_size = 2048
was set buffer_size = 262144
0 - Front Left
1 - Front Right
Time per period = 0.292000
```

天阿～好多問題要解決喔...

很抱歉，司徒已經盡最大努力讓這台掌機變更好，無奈技能不足，因此，在聲音沒有出來之前，無法釋出測試包給各位測試，請知悉～

2019-01-26 00:11

Miyoo 掌機是我給自己的一個目標，雖然高遠，但是，總是有希望...



2019-01-27 12:51

目前已經把 alsa underrun 問題解決了，原因在於 DMA DRQ type 設定，之前並非設定成 codec DRQ，雖然君正 datasheet 沒有提 DRQtype 的用途，不過目前看來是 DMA 硬體運作速度設定。

測試模擬器發現另外一個問題，模擬器會卡住，好像整個 hang up，但是系統沒有掛掉，因此，可能是 ALSA codec format 設定問題，目前朝這個方向 debug 中。

目前看來人為因素居多，看來移植不成功，可能真的是司徒的問題，哈哈。

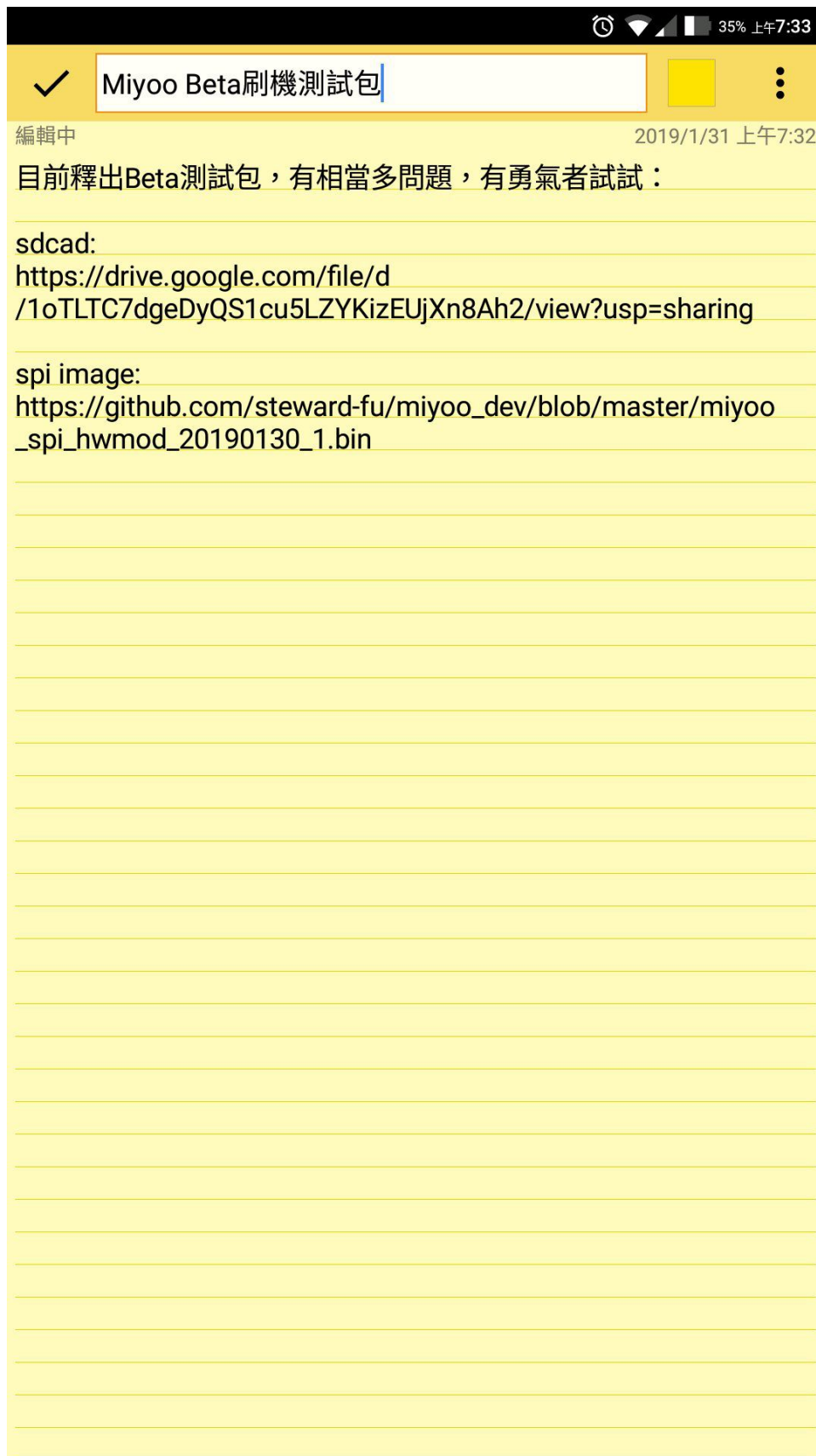
2019-01-30 22:22

目前釋出的測試包



2019-01-31 07:34

被百度吞了，只好用圖片



2019-02-01 19:17

1. SELECT + A/B => 調整音量
2. SELECT + TA/TB => 調整背光
3. 電池驅動已經完成
4. 僅測試以上 3 種功能，其餘不要玩，感謝

```
μfḡb2:\qι.τλε·dooδfε·com\ττfε\q\JKF·xυHSEλβεSxEWfqcK!τ2Sb8ητ9oττ \λτ6M3n2b=2μ9LτJ0  
2DC9Lq Iω9δε:  
ωτλoo zbt μmωoq Σ0Iθ0S0I τ'pτU  
2bI Efgμ2μ:
```

贴吧用户_5VU48Z1: 因為目前改機有的改一半，有的全改，因此，有點混亂，沒有改 sdcard 4bit 請不要刷這版

1.1: 修復USB模式

參照此帖: https://tieba.baidu.com/p/5981212479?see_lz=1, 根據23L, 短路DM、DP焊點, 移除R75、R76、R78電阻;

1.2: 修復TF卡4Bit模式

此步可以不做, 只是讀取TF卡會略慢, 預計未來對街機和PS1模擬器的流暢會有影響

參照1.1中帖子的34L, 將D2、D3、D0拉升4.7K (原文是1K, 實際有4.7K更好, 但實際使用並無差別), 並將卡槽的D2、D3、D0跳線至主板相應位置。

1.3: 制作刷机开关

其實只要在開機時候把兩個點短路連接一次就行, 最簡單的就是不裝外殼, 用一根鐵絲連接, 開啟刷机模式以後, 移除即可; 如果打算改LR按鍵, 這個就不用單獨去做了, 就是R1鍵

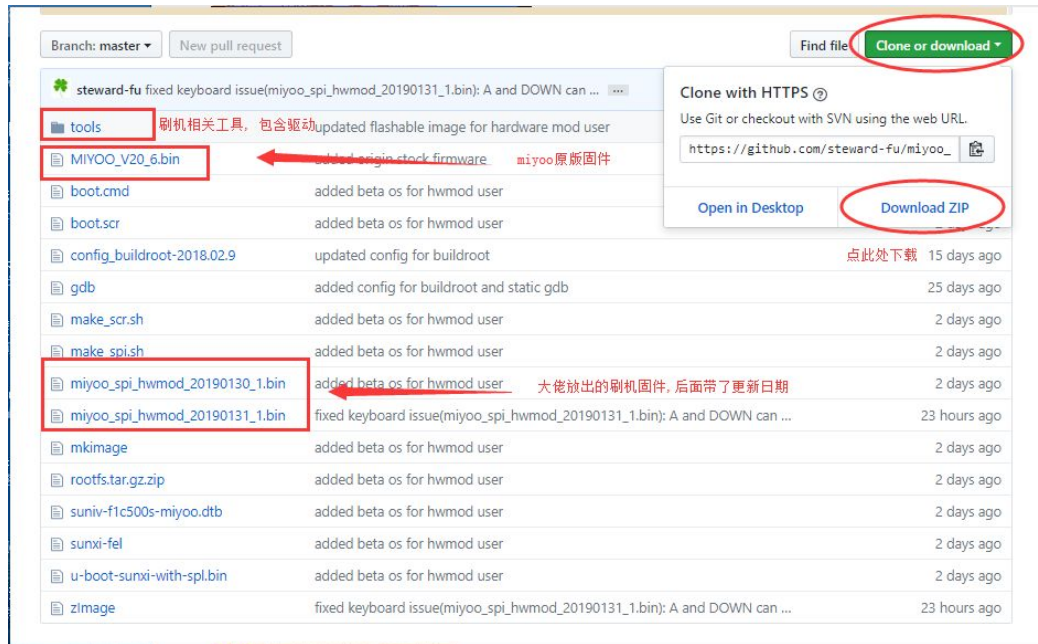
參照1.1中帖子的21L, 將spi flash 2, 3 (原圖右側最下面兩針對應的焊點) 飛出來做成跳線座或者開關。刷机時需要短路這兩點, 下簡稱“短路”。

1.4: 修復屏幕撕裂問題

此步小心短路焊點周圍還有一個小的貼片電阻, 不要不小心焊掉了

參照1.1中帖子的176L, 短路圖中兩個焊點即可。

*注: 此焊點僅為解決自制系統中畫面撕裂問題而開的硬件開關, 對原系統無效。實現修復撕裂還需後續軟件層面的支持。總之請先將此焊點焊上。



2.2.2: 安装驱动

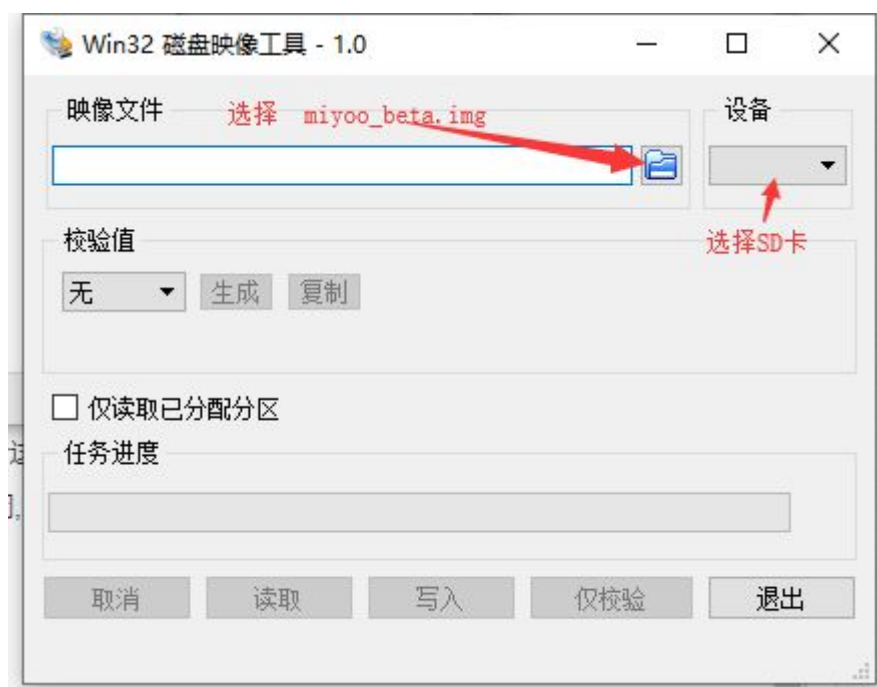
打开zadig-2.3.exe, 点击Options—List All Devices, 选中Unknown Device #1, 点击下方的Install Driver。

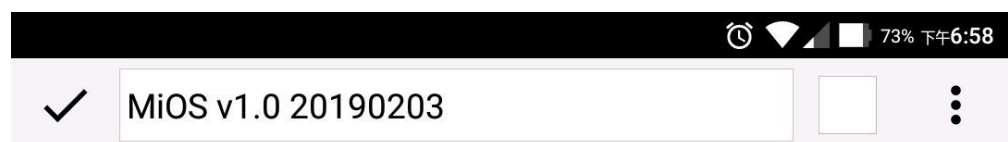
注：此步骤在同一台电脑上仅操作一次即可。

2.2.3: 刷入固件

将bin格式的编译好的固件放入上述工具同目录下, 空白处右键运行批处理, 输入”`sunxi-fel.exe -p spiflash-write 0 *.bin`” (无括号, *.bin是你的固件名)。

*注：也可在目录下新建文本文档, 输入”`sunxi-fel.exe -p spiflash-write 0 % 1`” (无括号), 保存为bat, 将bin格式固件拖到此bat文档上即可。





編輯中

2019/2/3 下午6:53

首先感謝眾多好友的協助，終於讓這個系統可以誕生！

而為了避免讓人以為OpenDingux就是司徒系統
因此這個系統就叫：MiOS，目前版號v1.0 (20190203)

在有限的時間以及能力支援下，這個系統也算及格
雖然距離流暢還有一段路要走，不過也算是一個里程碑

目前由於按鍵是掃描按鍵，因此，會有GhostKey問題
連續按下兩顆時，會有鬼鍵出現，之後只能硬解解決

目前有的機器可以超頻到800MHz，有的只能到700MHz
預設模擬器是跑702MHz，會當機，請改小

熱鍵：

select + A/B => 音量調整(10階)

select +TA/TB => 背光調整(10階)

必需使用GMenu2X的關機圖示關機，待屏變暗才可以關開關

已知問題：

1. ghost key問題
2. 仙劍進入後，必須連按兩下下鍵解決下鍵的問題
3. sms模擬器有更改設置是，5秒後才會回到60fps
4. gpsp無法設置選項

感謝太空淡藍色提供：

鏈接：https://pan.baidu.com/s/1XIS_BH1BCe66wcb-MbqIWA

提取碼：zetp

所以的檔案都可以從Github提取：

https://github.com/steward-fu/miyoo_rel

SORR可以跑的請分享下設置，感謝

最後，祝福大家新年快樂！







2019-02-03 20:37

另外，所以源碼(包含 Kernel 和模擬器)都會放到 Github，畢竟基於荔枝派 Nano 以及其它模擬器的基礎而改進，司徒還是會秉持感謝的心態開放修改後的源碼，也期望開源可以讓技術的發展越來越好，開放的時間為 5 月 1 日

超級 qioqio: 感谢司徒的新年礼物，只做了最简单的硬改，按键和 4bitS 还都没改。随便试了几个游戏，都很流畅。Miyoo 重生了



落魁战神: 凑个热闹 😊，谢谢司徒





2019-02-14 23:53

47% 下午11:53

✓ Patch (20190214)

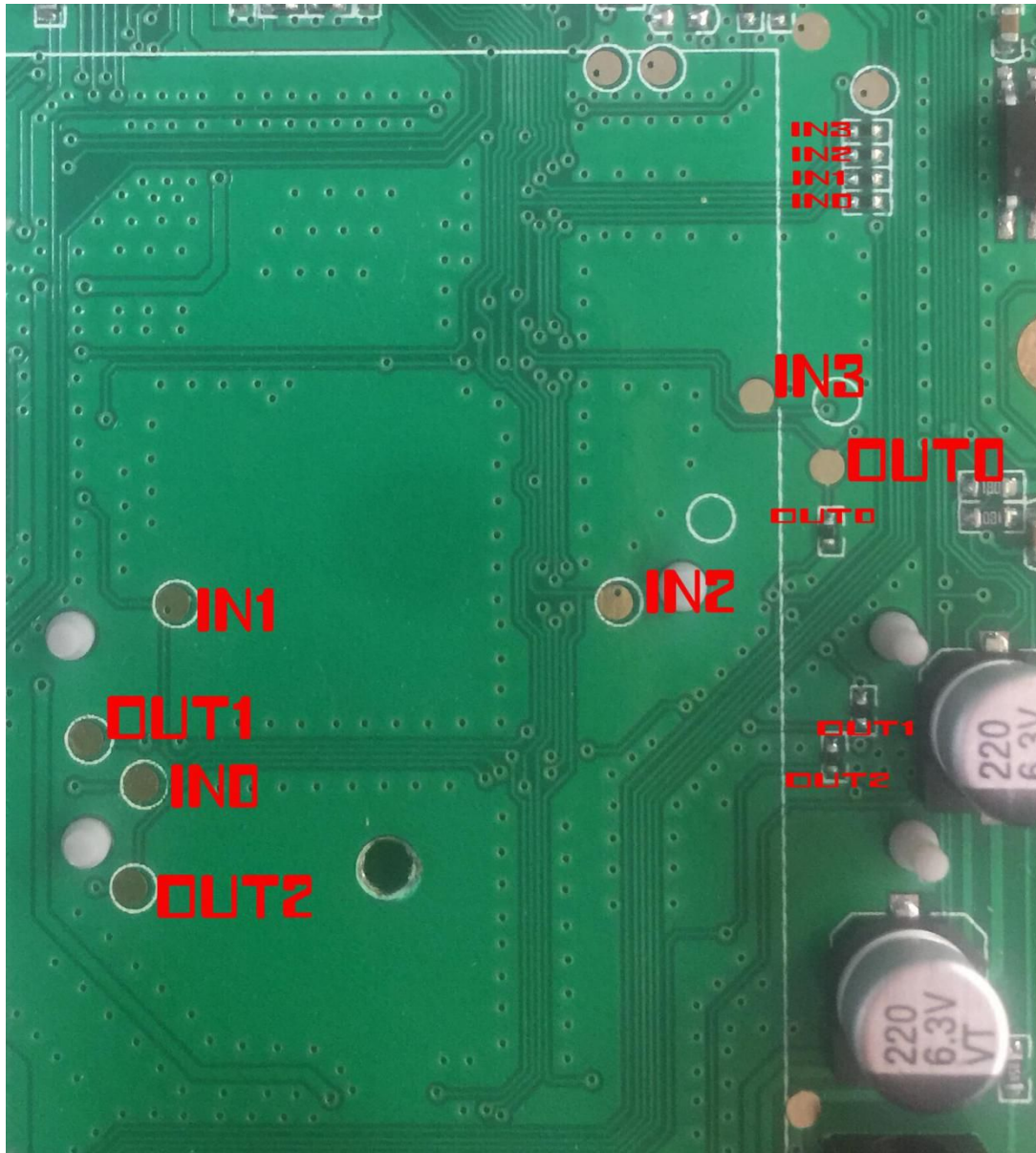
編輯中 2019/2/14 下午11:45

目前測試屏的TE功能，發現輸出只有30Hz，加上目前無法調整屏的Scanline位置去觸發TE腳位，然後奇怪的是，F1C500S的i80 Pixel Clock輸出也是少一半，因此，花了一些時間測試調整後，目前沒有使用TE腳位的功能，可以把閃屏解到一個算是相當不錯的狀態，值得測試一下，並加入支援SDL Double Buffer功能，目前釋出在Github上，檔名為：kernel_20190214.zip

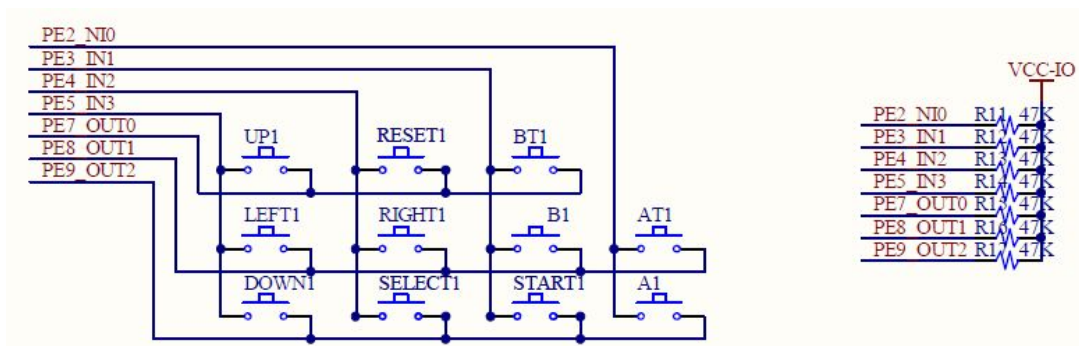
|

2019-02-15 15:41

硬改解決 Ghost Key 問題，我先把腳位貼上，之後改完在更新

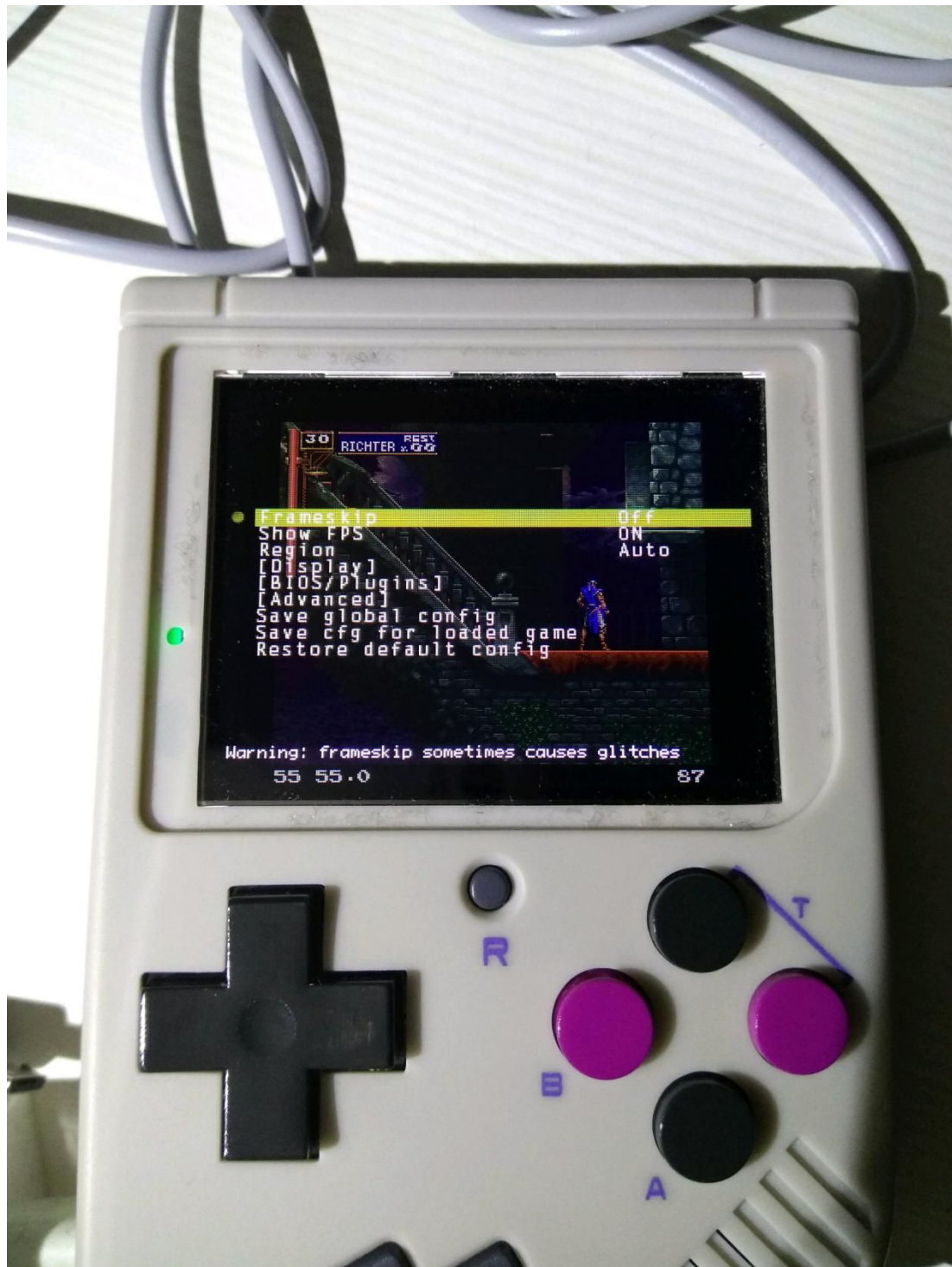


對應電路



2019-02-16 00:03

目前使用 SDL Mapping Buffer 操作且超頻到 702MHz，可以跑出 55FPS，CPU 使用率 85%，還有兩個加速方式可用，目前建議先改 4bits，讓讀取速度可以拉上來，我的測試已經是改 4bits 了！



2019-02-16 09:18

更正一下，右下角是 CPU 使用率 87%，目前測試是使用 1bit 的 SDCard，FPS 大約是 55，目前還在優化中，而測試 4bit 的 SDCard 則是大約增加了 5FPS 以上，因此，沒有要玩街機或者 PS1 遊戲的人，可以不用改

2019-02-16 22:43

先折磨你們一下，晚點再給你們玩...

2019-02-17 00:12

很高興達成第二階段設定的目標，目前 PS 模擬器已經釋出並且放在我的 Github miyoo_rel，由於 RAM 吃緊，所以相當依賴 Swap，建議改 4bit 且使用高速卡，下階段會開始優化這些模擬器，移植看運氣，優化看功力，期待可以優化到一個不錯的階段！



2019-02-17 11:02

gpsp 解掉無法設定拉伸問題，已上傳 miyoo_rel github



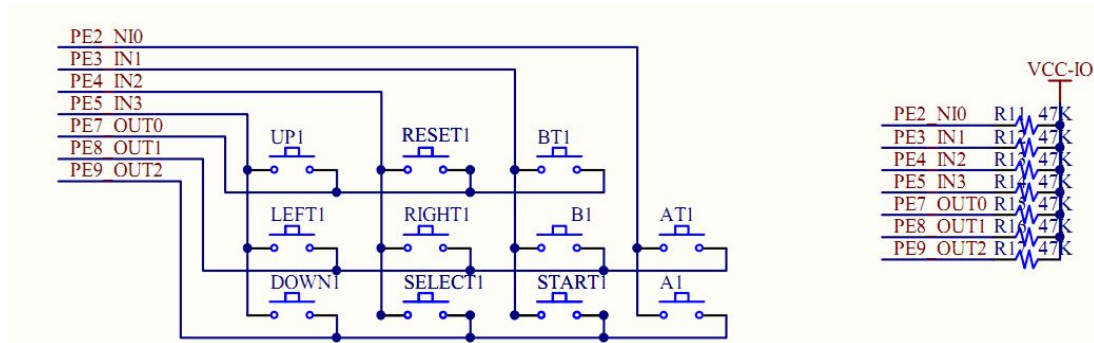
2019-02-18 17:07

目前司徒已經拿到第二版測試機器，雖然我原本沒有打算要支援，但是太多因素左右，最終還是讓第二版 Miyoo 可以刷機，第二版機器可以直接刷 SDCard 就可以開機，不需要任何硬體改造，並且保留原系統，所以可以兩個系統共存，目前還在善後中，有新消息在發布～

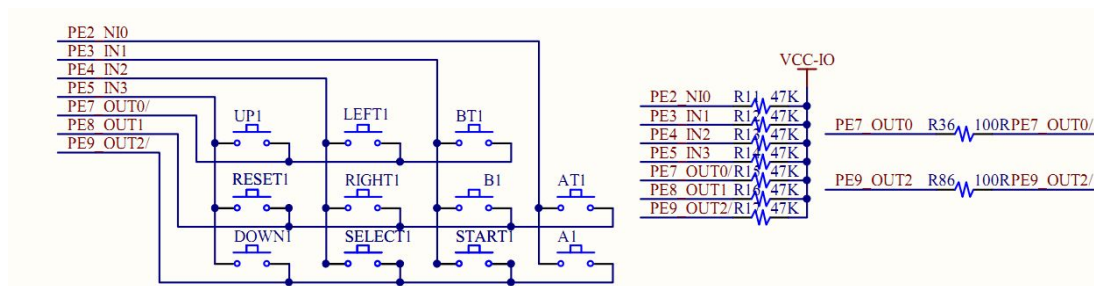


2019-02-19 11:52

目前第二版 Miyoo 的按鍵有做一個位置交換(Left and Menu)，意圖解決 Ghost Key 問題，如下：



第一版 Miyoo 的按鍵

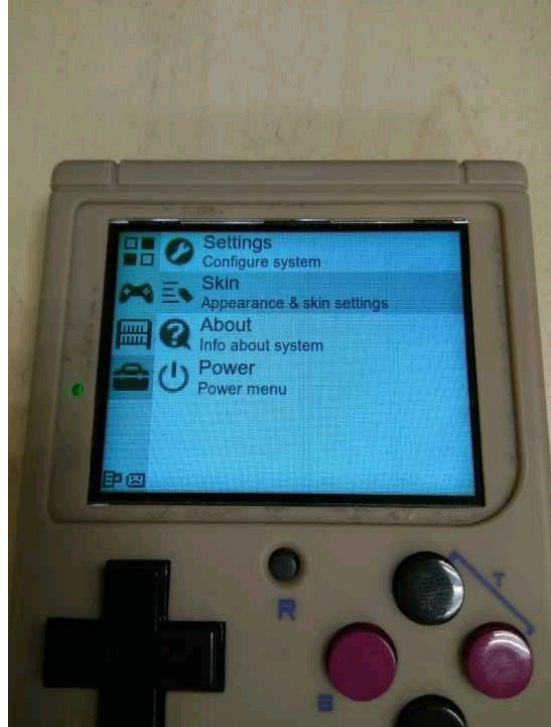


2019-02-19 17:23

GnGeo 模擬器已經更新到 miyoo_rel，村民要感謝跟我打賭 KOF97 八神連招的那位！

新主題，已經更新進 gmenu2x_20190219





2019-02-19 20:14

添加 ohboy 邊框以及調色板, ohboy_20190219



2019-02-20 00:43

gpsp 模擬器新加邊框功能, gpsp_20190220



2019-02-20 17:42

目前第一版以及第二版都確定有鬼鍵問題，老外建議換掃描方向，把鬼鍵移到反方向，目前看來效果還不錯，這版也沒有限制最多兩鍵同時輸入，建議可以刷新試試！

2019-02-20 18:07

目前加入兩張邊框圖片，已更新到 [github](#)



2019-02-21 12:31

除了搖桿，PS 模擬器可以支援震動功能，硬體有很多好玩的...

2019-02-21 16:59

目前新加入組合鍵(必須同時按下)，已經更新到 Github miyoo_rel:

Start B: L1

Start A: R1

Start TB: L2

Start TA: R2


如果先按 **Start** 再按其它鍵，則會先發送 **Start**，**Start + A + B** 不會觸發，因為這個組合鍵的前提是單獨兩鍵才會運作

lan_vincent: 这个设定对玩非动作类 RPG 等游戏的玩家很友好。

2019-2-21 17:59 回复

sss88888sss: 可以不用改机增加按键了

2019-2-21 19:37 回复

流星划过瞬间  : 司徒大神，关于鬼键问题要按照图示标序的序号飞线硬改吗 

2019-2-22 00:21 回复

贴吧用户_5VU48Z1: 要用二極體防逆流，等我把腳位確認下，你們在修改


2019-2-22 09:16 回复

夜雨清塵: 回复 司徒_Steward :司徒大佬怎么又退群了 

2019-2-22 10:15 回复

贴吧用户_5VU48Z1: 來去自如、居無定所

2019-2-22 10:18 回复

幻星  : 嗯，扫描键防串就是加二極隔离

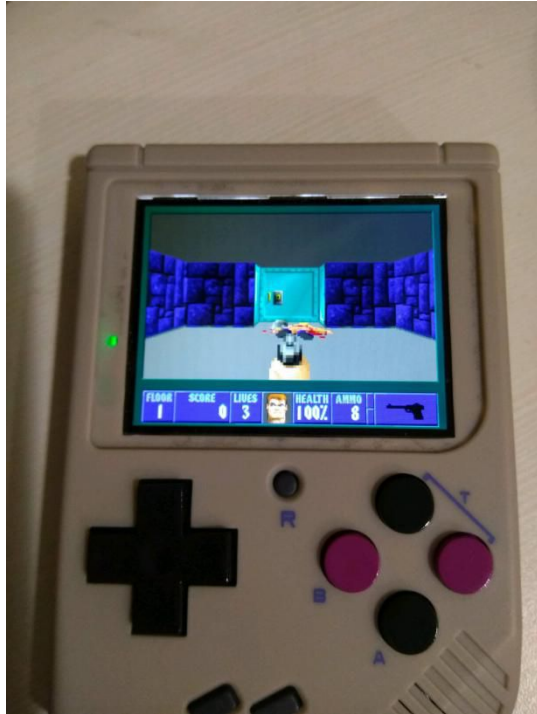
2019-2-22 11:36 回复

2019-02-22 22:26

ccdoom 更新成全屏

新增 wol3d 遊戲

都上傳到 miyoo_rel github



2019-02-23 00:32

cdogs 已經上傳到 miyoo_rel



2019-02-23 21:18

依據老外建議過濾雜訊，當按下 `down + left + b` 時，無視 `start`，已經更新到 `github`，請更新

2019-02-24 00:37

想當年使用 PS1 的震動手把時，那種激動的心情到現在都還是記憶猶新，因此，為了回憶一下過往的美好，司徒決定讓 Miyoo 可以支援 PS1 的震動手把功能，而加上 PCSX ReARMed 模擬器本身就有支援這個功能，因此，司徒決定先把硬體搞定，接著再把驅動以及模擬器完善，當然震動還可以應用在 Doom 和 Wold3D 上，可玩性十足，而怕電池不夠的人，司徒已經把電流降到一個最低限度，僅僅耗 18mA，相信改完電池，不太需要擔心。

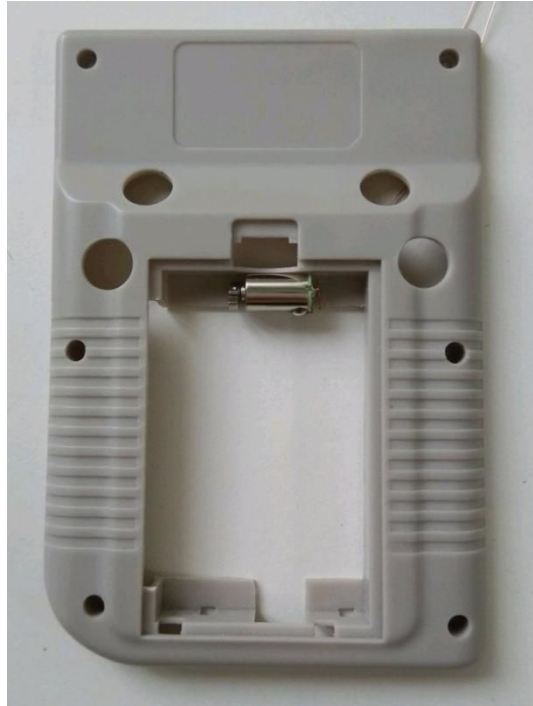
電路如下所示，因為司徒手上只有 NPN 和 NMOS 以及最低的 180 歐姆電阻，因此司徒利用兩個 NPN 充當 PNP 使用，這是因為 F1C500S 預設是高電位輸出，你應該不會想要在刷機時，那顆馬達一直跑吧？而 180 歐姆並聯成 90 歐姆使用，限制電流大於為 18mA，最後再使用二極體做反電動勢抵消，整體電路不難，只是司徒手上沒有適當的元件，因此看來比較複雜，手焊的部分，需要比較多的技巧，多多練習即可，畢竟可以擁有震動功能，值得！

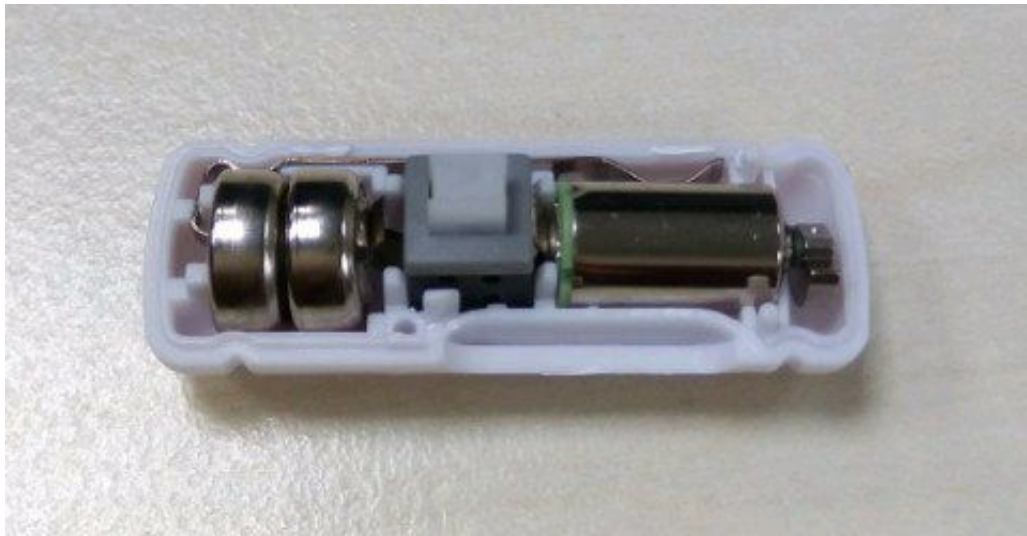
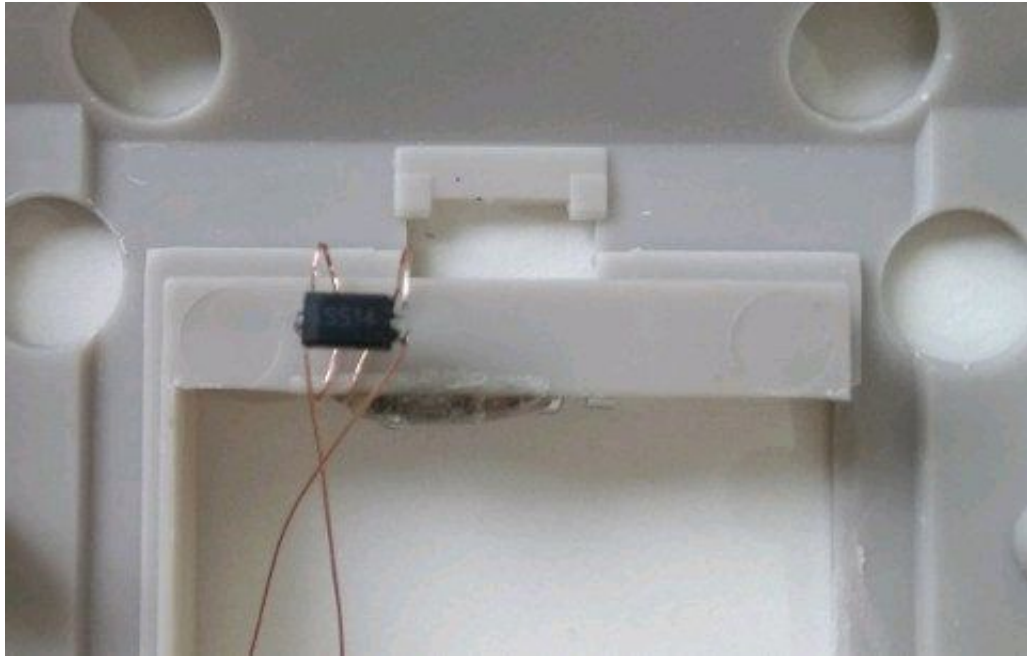
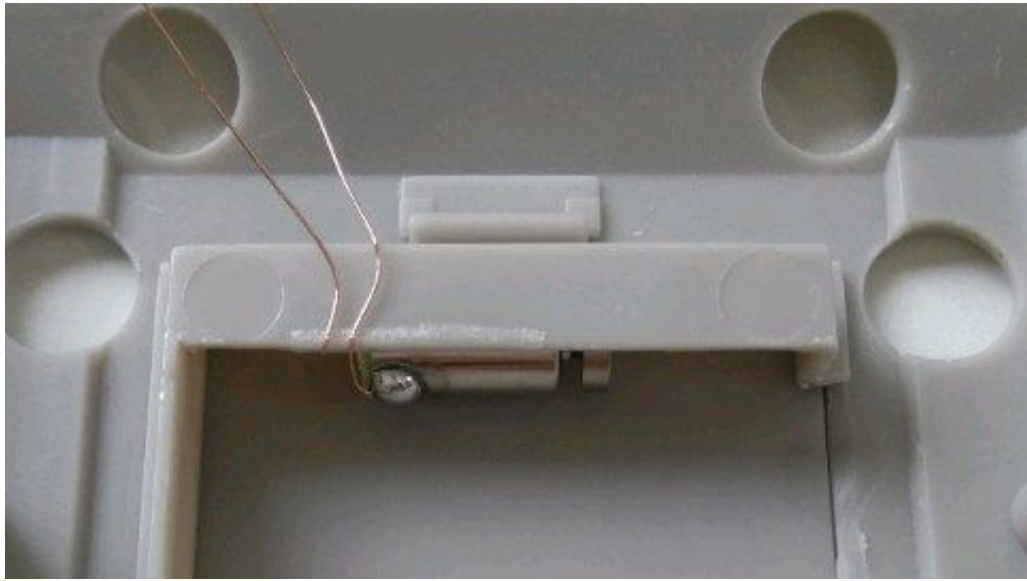
馬達的部分比較激情，選用強度比較好的 `durex play`，聽說也比較省電！

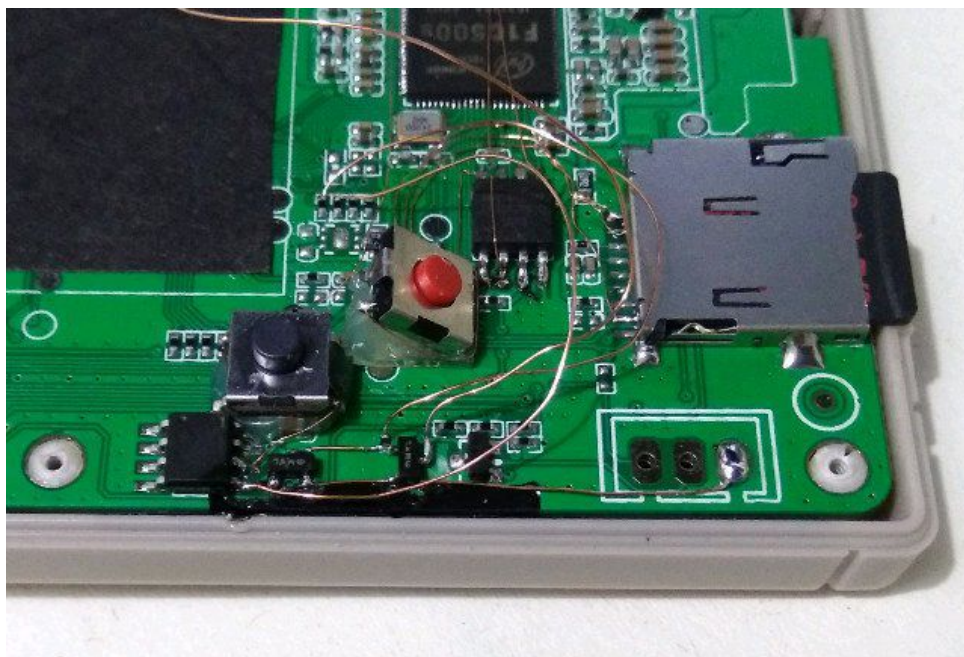
目前驅動已經完成，司徒正在移植 PCSX 的震動相關代碼！

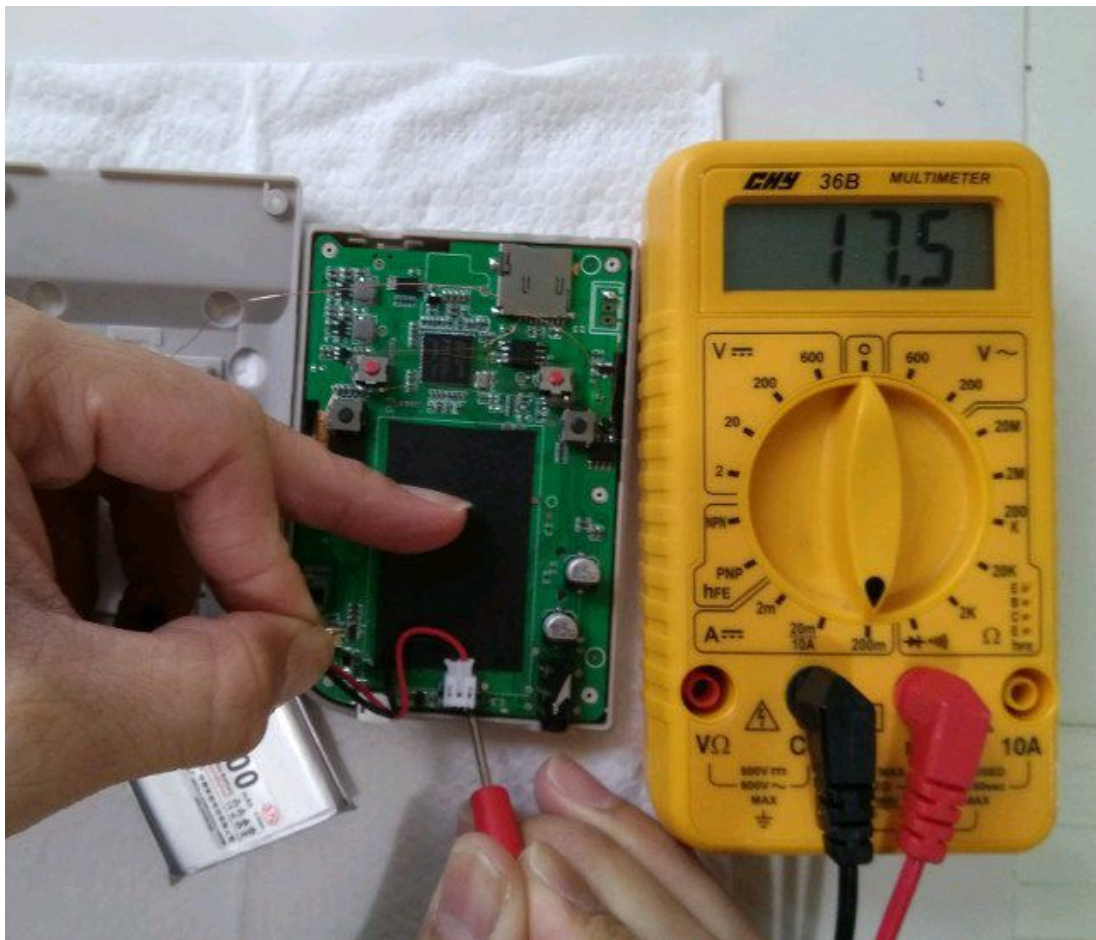
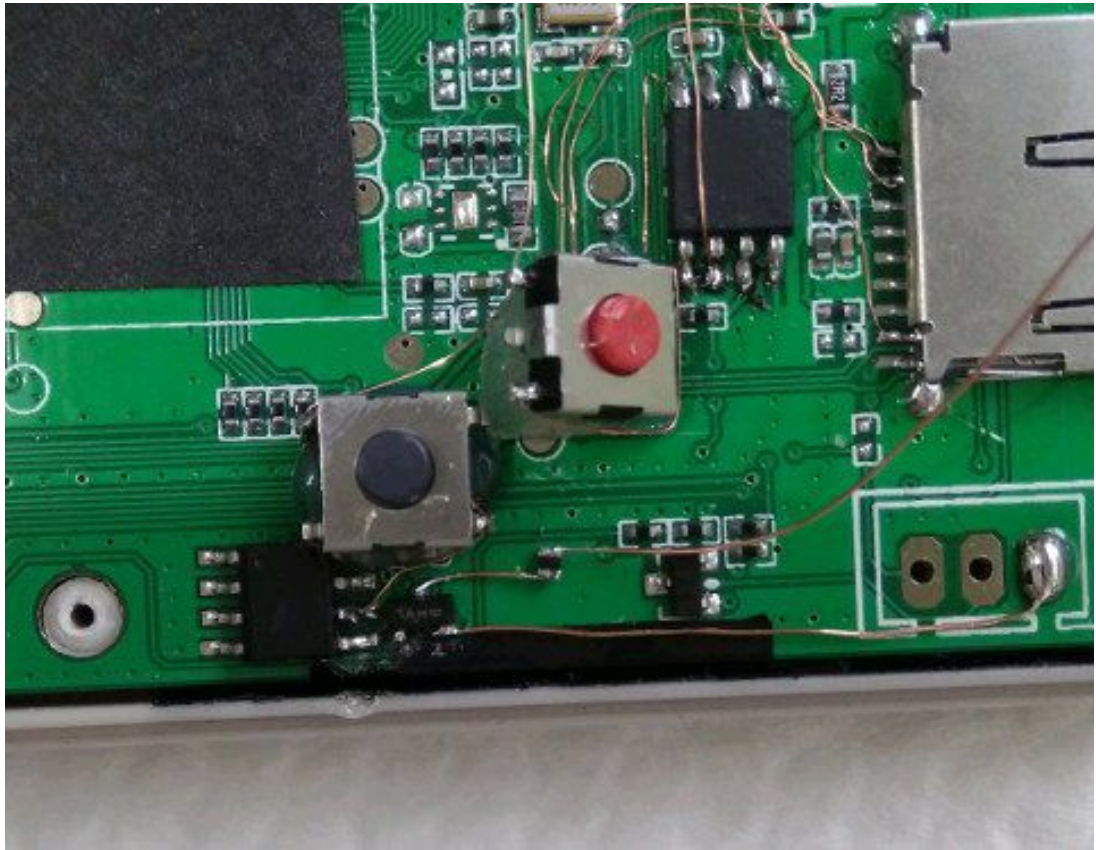


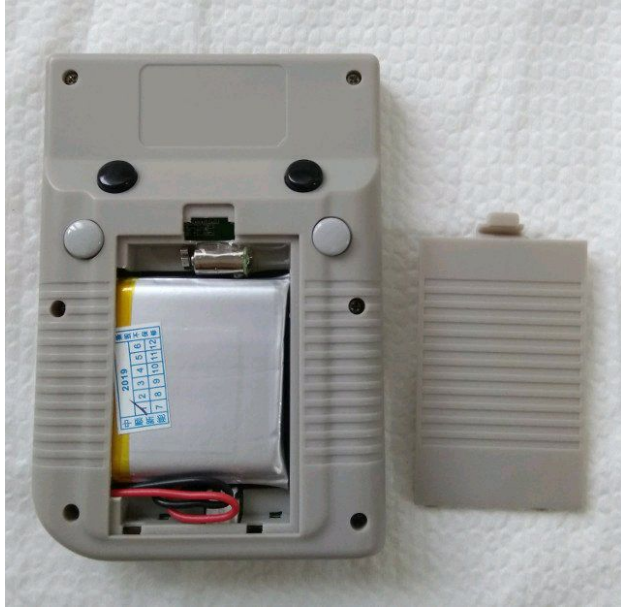


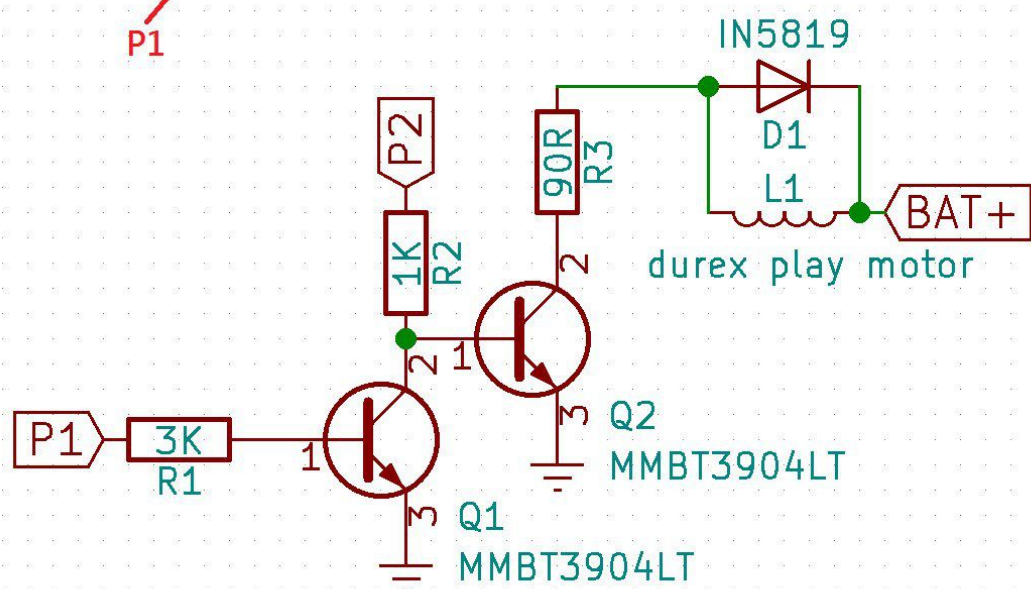
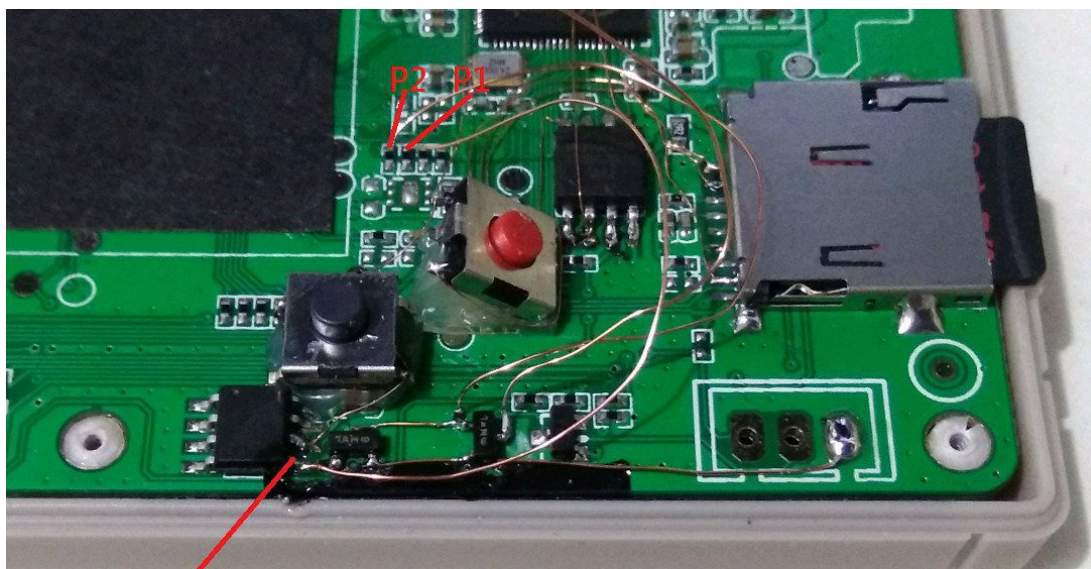












wikeylee: @司徒_Steward 请教司徒神，我电路基础比较差，可以简单图解一下这个电路接线方法么？

2019-2-24 18:25 回复

大小鸟啦: 你好司徒，是待机 18mA，还是工作的时候 18mA？谢谢你！

2019-2-24 19:51 回复

贴吧用户_5VU48Z1: 基本上，可以想像那两个电晶体就是一个开关，低电位时，开关让马达导通，而导通时，耗电流为 18mA

2019-2-24 20:11 回复

2019-02-24 14:29

目前全部都完成了，即將發布燒錄固件以及 PS 模擬器，有震動真的超級爽，強烈建議更改，震動輸出腳位電壓為，低電壓觸發，高電位停止。

1. 設定類別搖桿
2. 啟動震動選項
3. 遊戲也要開啟震動效果

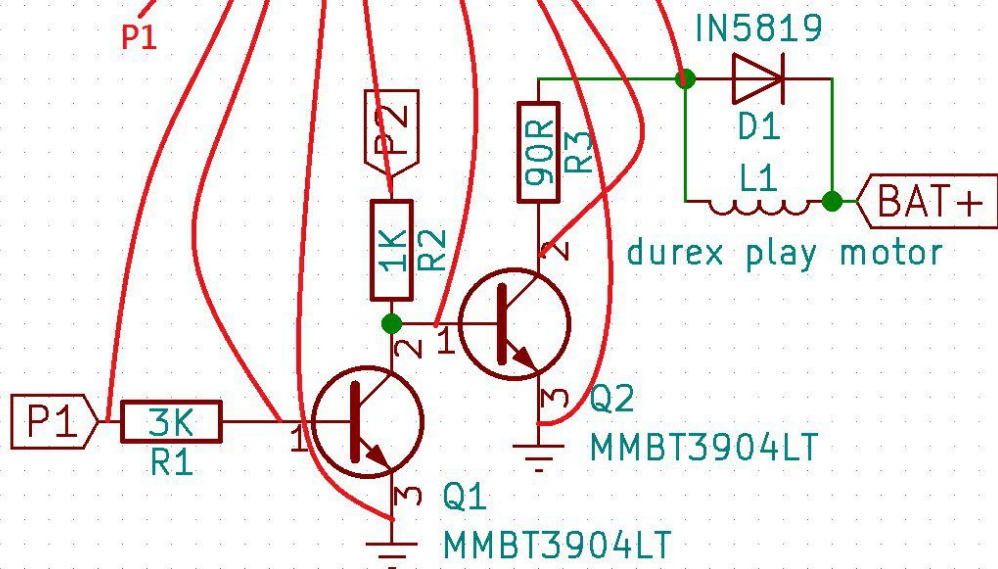
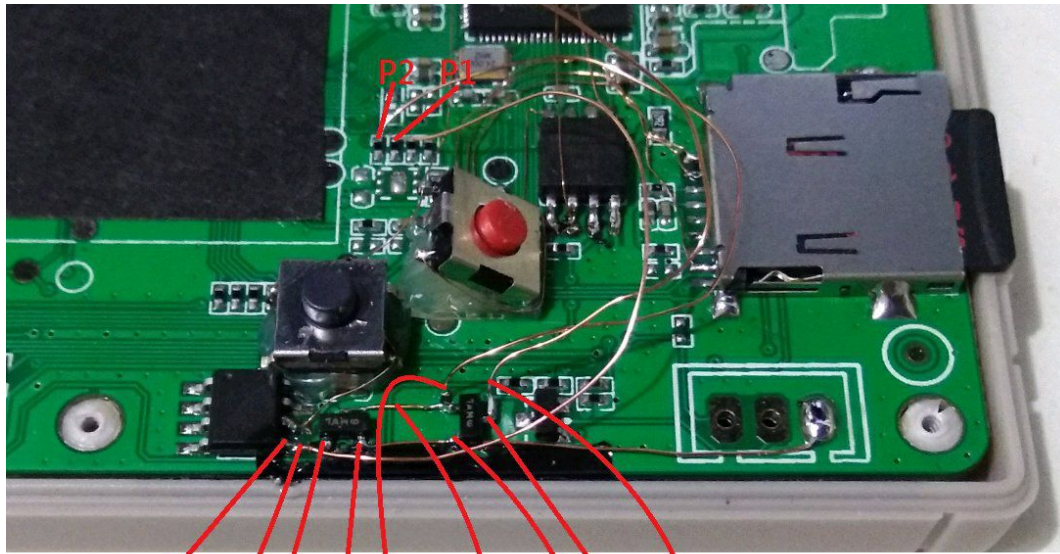




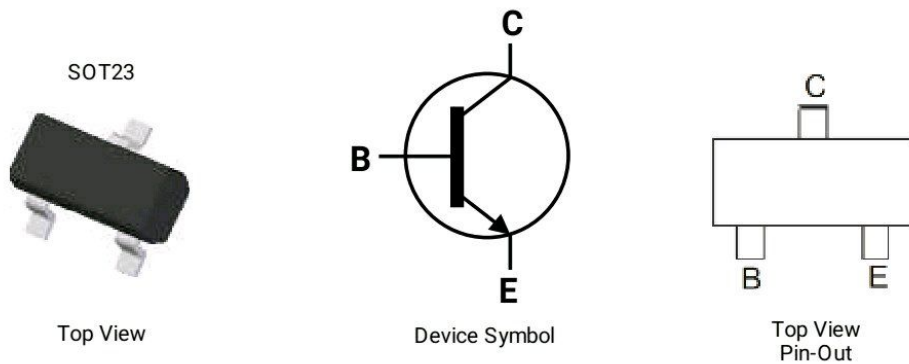
2019-02-25 09:38

不行，建議不要刷

拉線圖

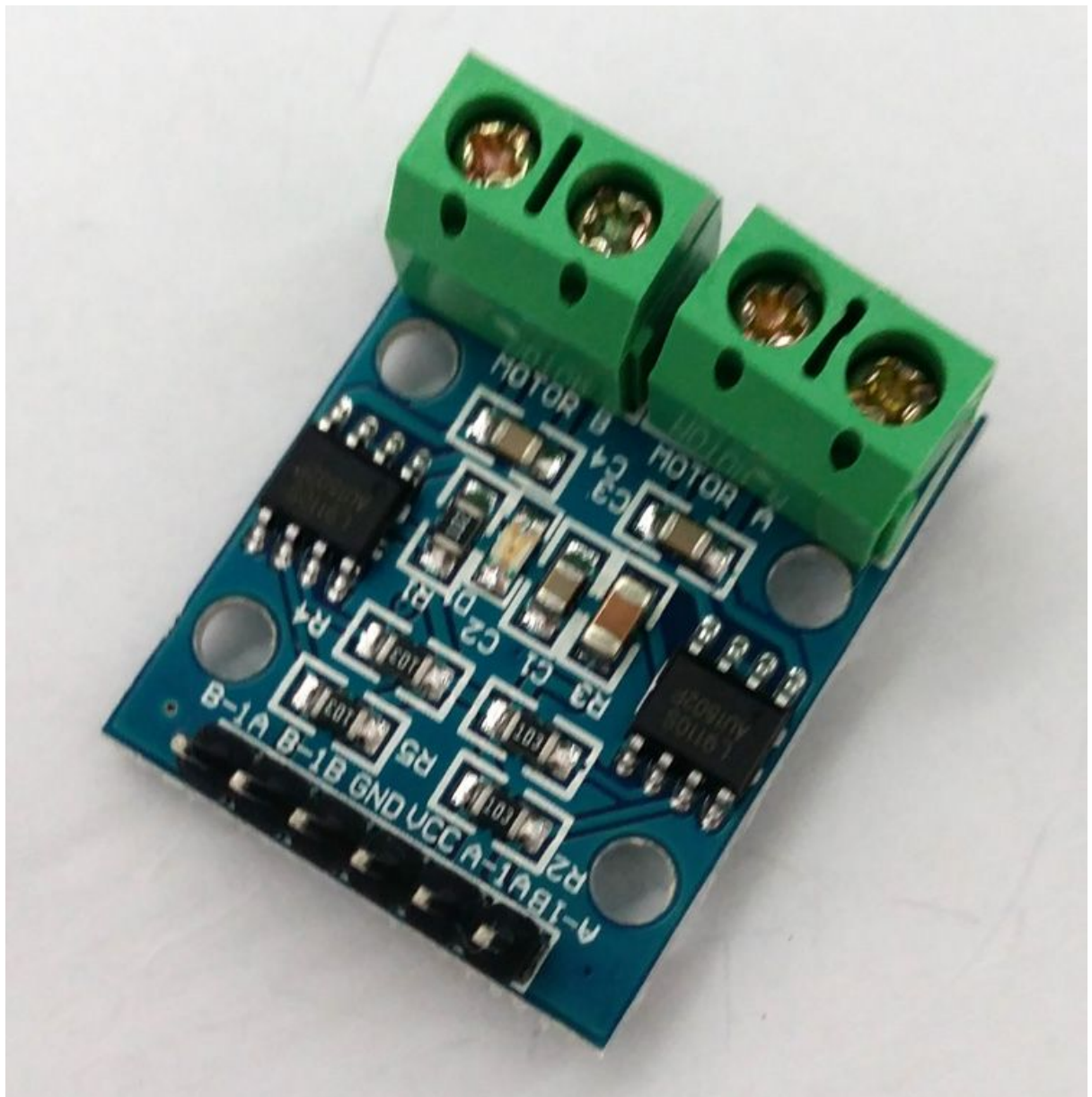


mmbt3904lt 接腳



2019-02-25 11:04

如果覺得自己焊接太麻煩，也可以買淘寶賣的電路板，當然也可以使用馬達驅動 IC，P1 腳位就是訊號線(低電位馬達運轉)，P2 是給提昇電阻的電壓



贴吧用户_5VU48Z1: 1. 把 p1 接在一起 2. 把 p2 接在一起 3. p1 接到 pcb a1a 4. p2 接到 pcb a1b 5. pcb vcc 接到電池+ 6. pcb gnd 接到電池-

2019-2-26 09:23 回复

2019-02-25 12:00

目前第二版 miyio 測試包已經提供測試
使用 win32 disk 燒錄並擴容

1. 直接插卡開機是 MiOS
2. 開機後 1 秒插入是原系統

Miyoo 第二版測試刷機包，請分享一下，感謝

https://drive.google.com/file/d/1WS7RGPg3V_JEDzXl5X19EsVYxt3zTnkl/view?usp=sharing

2019-02-25 12:24

因為目前 uboot 還在修，因此，開機前 10 秒時黑屏，不用擔心

2019-02-26 14:03

Jetluy基於測試版，幫忙打包一版很不錯的版本，想測試的可以刷刷（僅支援第二版Miyoo）

Changelog Bittboy_CFW_V2 Beta_20190225.7z

- *latest framebuffer fix thanks to steward
- *latest zImage fix thanks to steward
- *added cannonball thanks to sauce
- *added dosbox thanks to steward
- *added ggeo thanks to steward
- *updated gspsp thank to steward
- *added oswan thanks to sauce
- *updated ohboy with templates thanks to steward
- *updated pcsx_rearmed thanks to steward
- *ccdoom updated for full screen thanks to steward
- *added quake port thanks to sauce
- *added quake 2 port thanks to sauce
- *added sdlpal_v2 thanks to steward
- *added wolf3d port thanks to steward
- *updated gmenunix thanks to pingflood steward

<https://drive.google.com/file/d/1Y365b-eKczSrN4s9M3qddb7ESKmFNSpy/view?usp=sharing>

第二版 Miyoo 多數可以到 798MHz，不順的就先這樣頂頂



2019-03-01 21:52



← 歡迎加入discord



0 秒前

2019/3/1 下午9:52

Jutley(英國人)在 discord 線上討論版，建立懷舊掌機的討論版面，他希望可以看到更多亞洲人參與討論，因此，如果你有興趣看看老外在討論什麼，都歡迎安裝discord並加入，這也是訓練英語的好地方：)

以下是他的想法：

Hi Steward can you do me a favour can you share this link to the discord here to your forums on baidu and any other ones i would like more asians koreans etc to join this discord smiley retrogame handhelds below.

<https://discord.gg/5qbZKsm>

2019-03-03 13:46

SORR 已經更新到 github miyoo_rel, 目前已經正常可以玩了



2019-03-03 17:32

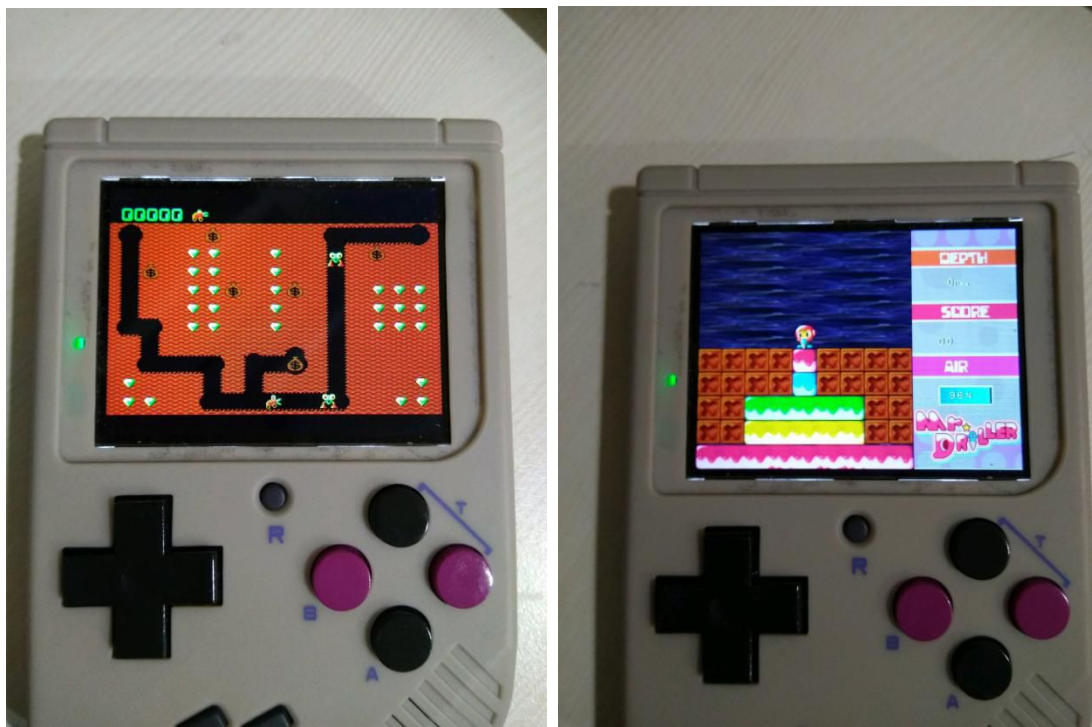
好的 晚點發布 硬改解決鬼鍵的包|

2019-03-03 17:33

asciportal 已經更新到 github



digger 和 mr.drillux 已更新



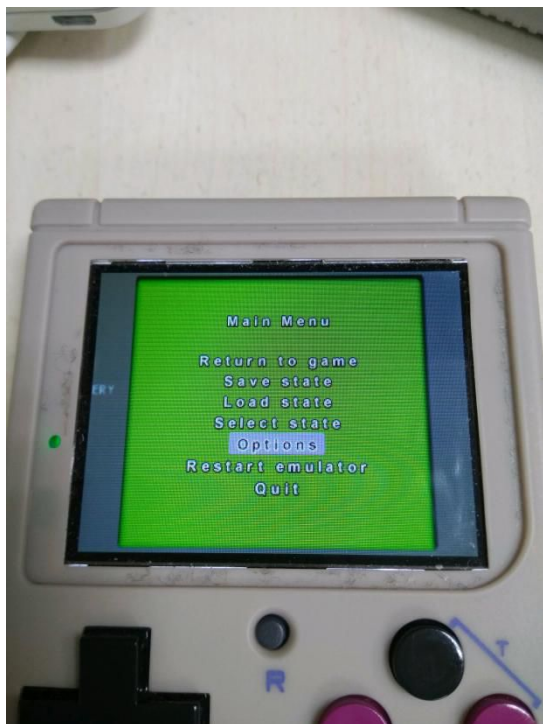
2019-03-07 23:19

目前司徒還是持續優化 PS 模擬器，跟 Wiz 的距離已經越來越近，目前只有給群裡以及老外測試，待穩定之後會放到 github



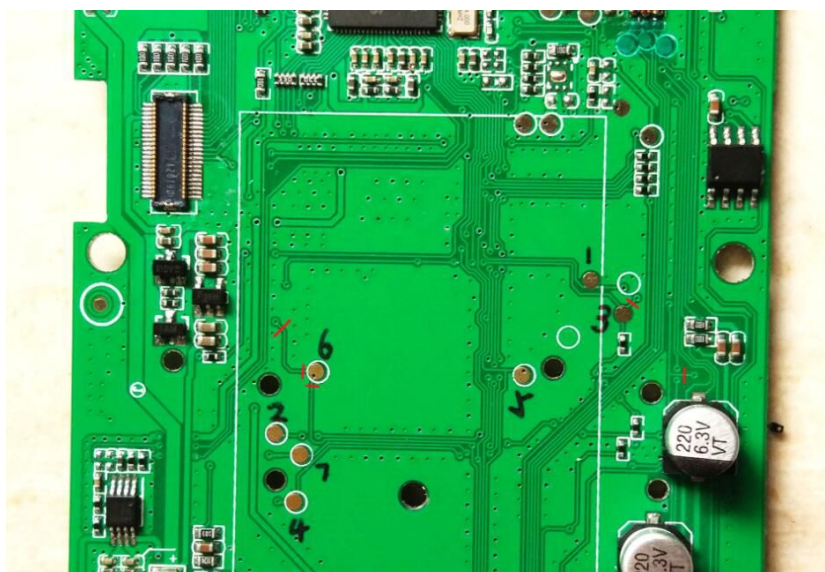
gambatte 模擬器已經上傳 github

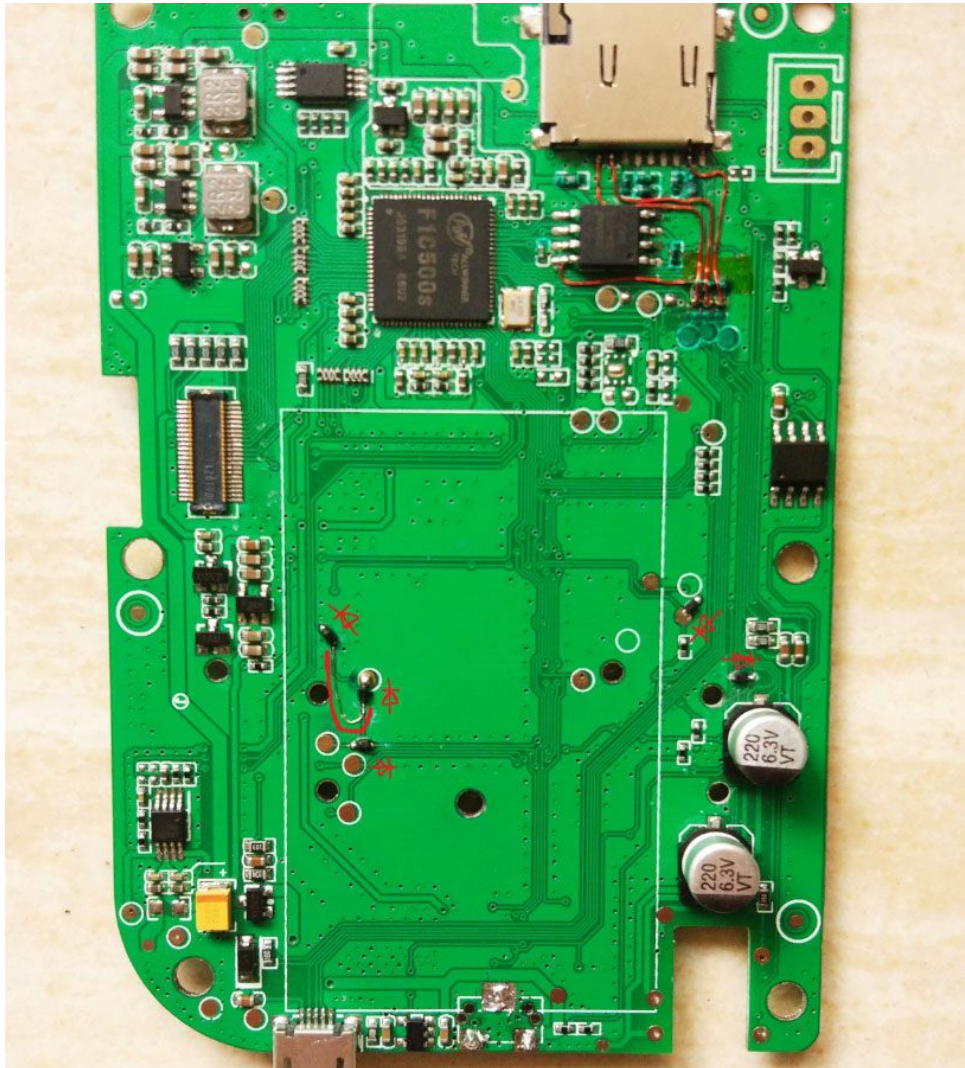
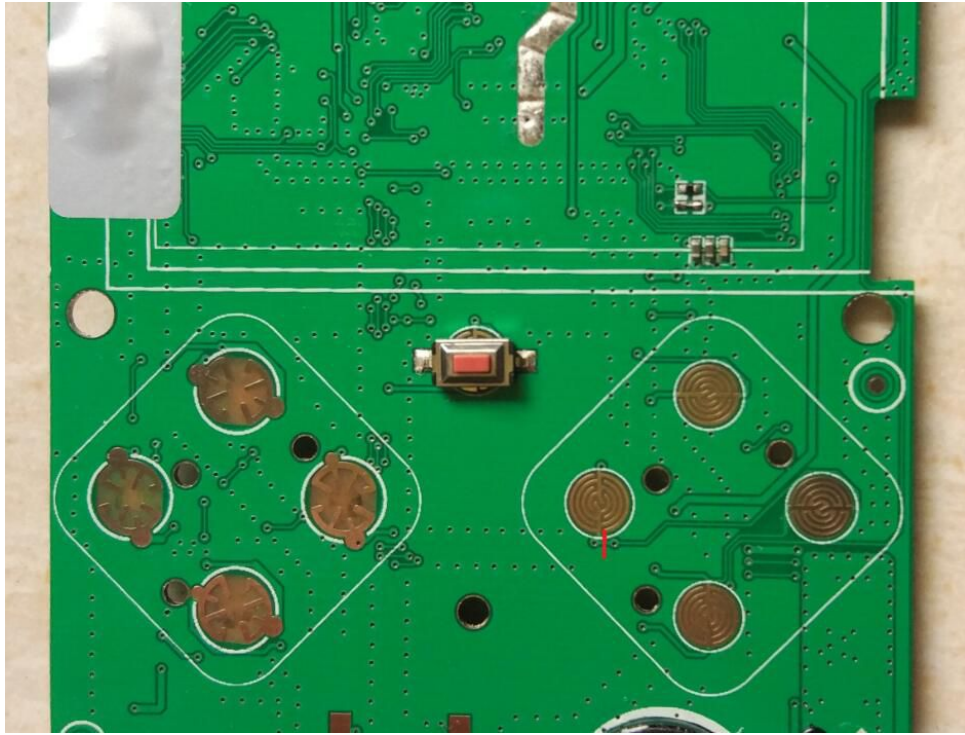




幻星：非常感激司徒赋予这机子更强生命力！分享本人采用的另一种解决 MIYOO 鬼键（串键）的方案：此为增加二极管解决方案，不想过多跳线的朋友可用此方法，完全解决游戏主键产生的鬼键问题。只需要一根很短的跳线+5 个 0603 规格的二极管，如下图通过切断和刮漆焊接完成。线路非常细，建议用尖头刮刀操作，切断和刮点不能太大，要恰到好处，过大了焊接时可能会与旁边的负极短路，刮漆时也要适当用力，过大力会把铜片去掉，因线路面积太小，也不能随便轻刮几下，要刚好磨花整个铜片表面才好上锡，用好一点焊锡，最后最好用风枪吹吹焊加固下。这些都是看个人的经验手法了。提示：此硬改法为第一版机器+刷司徒的 0220 及之后的 SPI 程序。第二版的机器厂家调整了 L 和菜单键，按原理图，只需更少的二极管就能解决鬼键，我暂时没有第二版的机器，未能分享硬改图。

2019-3-11 18:48 回复





2019-03-12 15:12

MP3 播放器晚點更新到 github

A: 加入播放清單

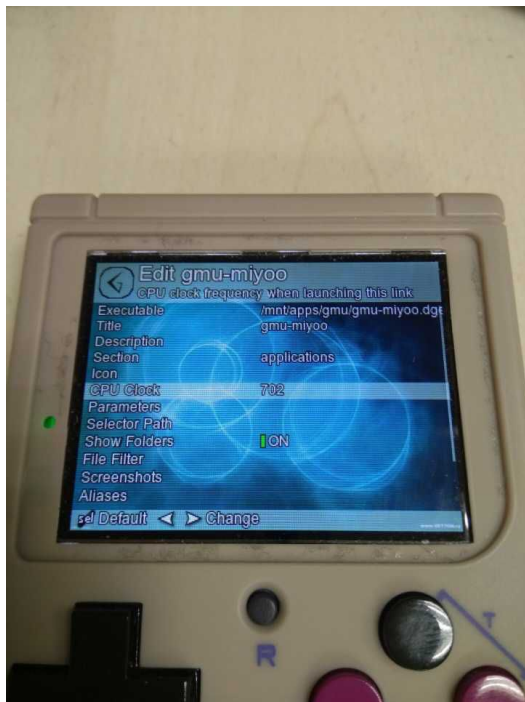
START: 切換播放頁面

SELECT -> START: 離開

R Menu: 關閉螢幕

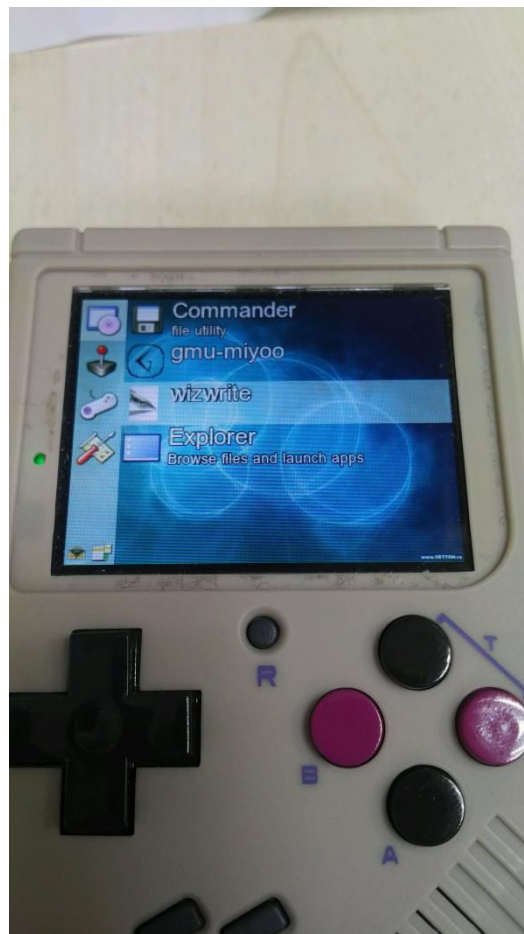
R(START + A): 開啟螢幕

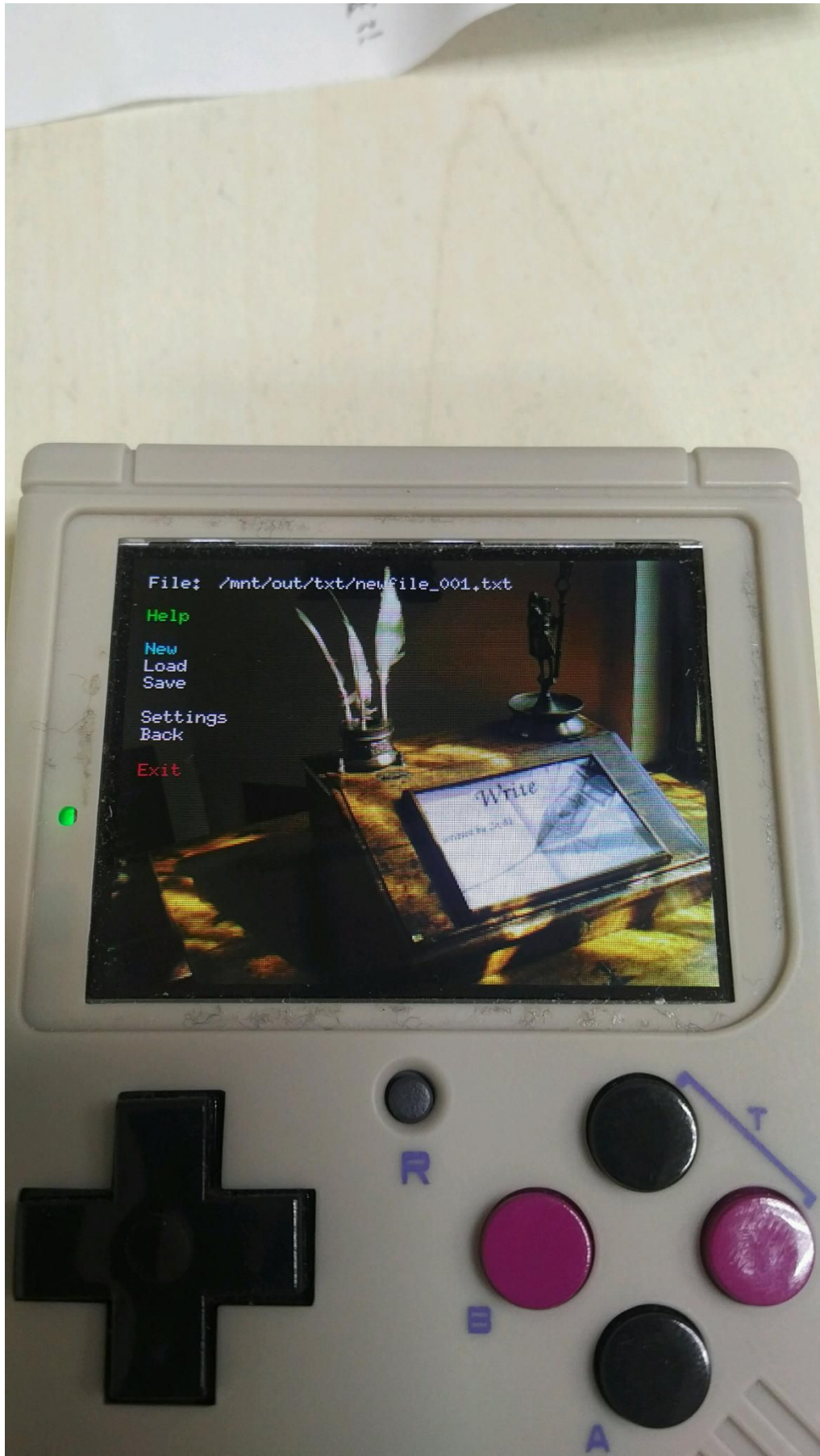




2019-03-13 15:28

由於之前移植的文字編輯器有點模糊，因此，改用這個 wizwrite，目前已經上傳到 github





2019-03-14 19:04

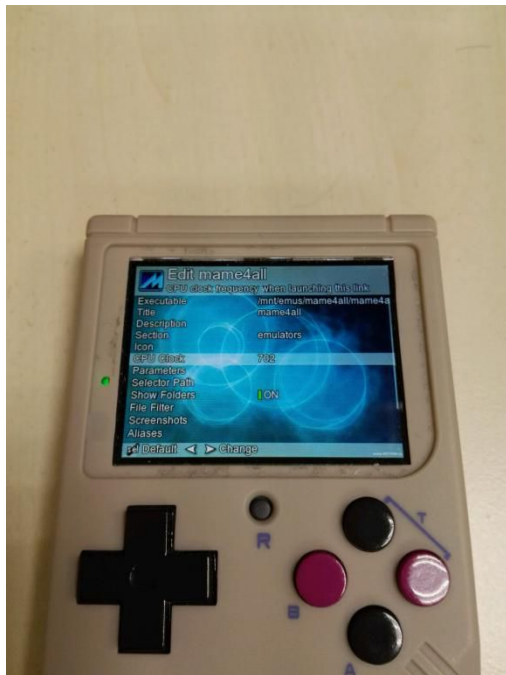
Changelog
Bittboy_CFW_V3.1_Beta_20190314

- *fixed quake not starting changed set cpu 702mhz
- *added wonderswan emulator back thanks sauce
- *added gamebattler gb emulator (B => quick save, TB => quick load) thanks steward
- *swapped to gp32 gameblabla to fix input lag problems thanks sauce
- *fixed gambattle emulator (change A and B button, remove hotkey for both quick save/load) thanks steward
- *music player gmuplayer (A: add into play list START: change tab page SELECT -> START: exit R Menu: blank screen R(START + A): turn screen on) thanks steward.
- *added wizwrite thanks steward
- *added new game tombstone thanks steward

https://drive.google.com/file/d/1eD1YnV3s3kIBWdAHb4U10_GYIqSgK-D/view?usp=sharing

2019-03-14 20:24

mame4all 已經上傳 github





2019-03-15 15:38

首先感謝各位的支持，目前 Miyoo 系統算是比較穩定了，待之後把小問題解決後，會出一版正式版，而模擬器優化就如司徒之前說的：Miyoo / V3s / 黑妹 系列都會持續優化，只是優化需要時間累積，短時間只能做移植動作！

司徒目前會先把博士 V3s 掌機的系統先建立起來，因此，miyoo 會取消置頂，改由 v3s 置頂，但是後續更新，司徒都會持續更新到這個貼子，請不用擔心！

也感謝大家對開源的支持！

2019-03-18 17:51

標注一下：

目前閃屏問題比較難解，因為經過測試，屏的 Screen Tearing 腳位輸出是超過每秒 60 次(raising edge)，但是 F1C500s/F1C100s 使用 GPIO_PE 做 Interrupt 觸發(raising edge trigger)，卻只能收到每秒 30 次的中斷觸發(司徒在中斷裡面只做 count 動作)，即使設定 Highest priority 也是一樣，因此，閃屏問題，目前看來 f1c500s/f1c100s 是無法解決，因為 GPIO 的中斷觸發每秒只能 30 次，所以，最終只能使用 RGB 屏解決閃屏問題！

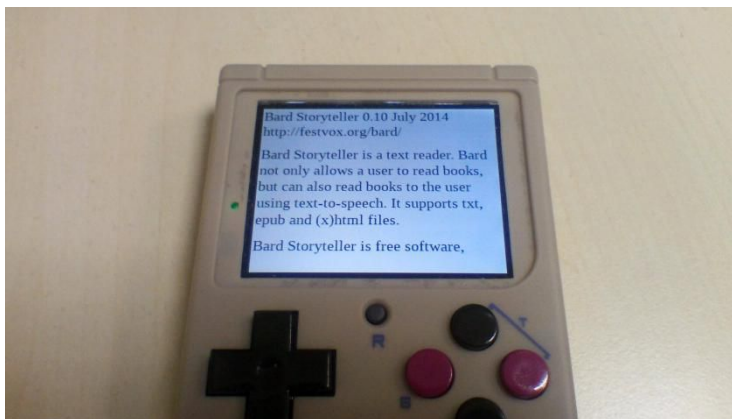
2019-03-19 17:34

金庸已經正在上傳 github



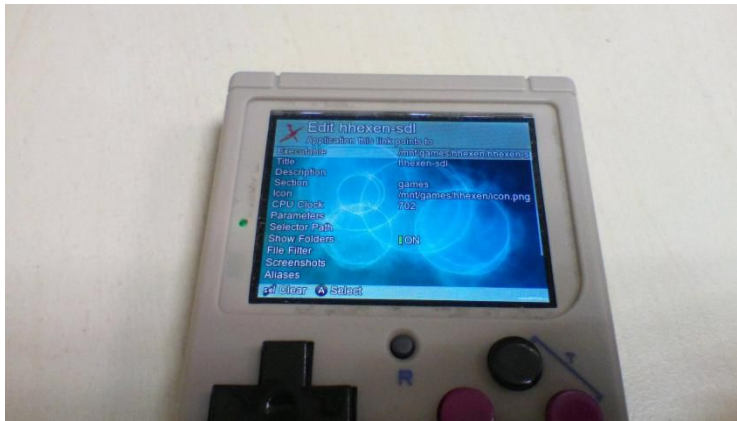
2019-03-20 20:19

liero, bard 已經是上傳 github



2019-03-21 16:41

hhexen, hherctc 正上傳 github





2019-03-21 17:32

rott 晚點更新到 github

uploaded my github already



2019-03-26 00:09

文曲星已經完成移植



2019-04-09 17:46

目前 GBA 模擬器(gpssp)已經支援振動功能, 已經上傳 github, 如下遊戲是有支援振動功能



2019-04-09 19:32

目前 GBC(gambatte)也已經支援振動了, 已經上傳 github



2019-04-12 11:32

對於喜愛的東西來說, 花費時間在其身上研究, 不應該說是浪費時間, 應該說是淫浸在興趣之中...
對於無法等待的人來說, 建議購買其它掌機替代!

2019-04-19 23:54

執行 pcsx 的 reset game 應該就可以震了，這個 bug 之後修

2019-04-22 18:57

這個是最後更新通知，目前我已經把所有源碼都放到我的 github (F1C500s, Miyoo)，對於程式碼有興趣的朋友，都可以持續更新優化，目前會先好好休息，致力於把自己的網站完善，也如同之前承諾，等之後再次出關後，會持續優化更新模擬器，也期許開源掌機可以越來越好，最後，感謝各位的支持！

原文地址：<https://tieba.baidu.com/p/5981212479>

原作者：司徒(<https://steward-fu.github.io/website/index.htm>)

整理人：三木同子

整理时间：2020年9月27日

注：原文有刪減，不是刪改