

О настройке buildroot для Allwinner V3S.

Как по порядку настроить buildroot с ядром 4.14 для того, чтобы работал дисплей с capacitive touch touch и WiFi.

Buildroot это достаточно удобная среда, для сборки embedded linux. Она объединяет вместе ядро линукса, uboot и busybox. Ядро 4.14 мне кажется всё ещё достаточно новым. Например Ubuntu, с которой я пишу текст имеет ядро 4.15.

Первоначальная установка buildroot.

По ссылке <https://github.com/Squonk42/buildroot-licheepi-zero> есть buildroot заточенный для Allwinner V3S и Lichee PI. Поэтому копируем оттуда основную ветку и в корневой директории пишем:

```
make licheepi_zero_defconfig
```

```
make
```

И долго долго ждем, когда соберётся первоначальная версия. Она будет без работающего дисплея, но уже с uart/i2c/ethernet(проводной)/LRADC работающими.

Дальше можно проверить, что оно как-то работает.

Записать на SD карту можно например так:

```
sudo dd if=output/images/sdcard.img of=/dev/sdc bs=4M oflag=dsync
```

/dev/sdc - это имя вашей sd карты, надо быть **аккуратнее**, чтобы случайно диск какой не отформатировать.

Подключать переходник USB<>UART надо к uart0, это пины IO-B09, IO-B08

После этого можно воткнуть SD карту, включить устройство и подсоединиться используя команды

```
cu -l /dev/ttyUSB0 -s 115200
```

```
screen /dev/ttySUSB0 115200
```

```
Login: root
```

```
Password: licheepi
```

Инициализация дисплея в uboot.

Запускаем

```
make uboot-nconfig
```

выставляем такие значения

ARM architecture->

```
LCD panel timing details
(x:800,y:480,depth:18,pclk_khz:33000,le:87,ri:40,up:31,lo:13,hs:1,
vs:1,sync:3,vmode:0)
```

```
(PB4) LCD Panel backlight PWM
```

```
() LCD panel backlight pwm is inverted <- тут не должно быть
галочки
```

Далее компилируем его

```
make uboot-rebuild
```

```
make
```

После записи на SD карту можно будет увидеть картинку пингвина (это uboot присует) и лог на экране, который уже linux пишет.

(Внимание! PB4 не используется для PWM в плате balmer_allwinner_v3s_v2)

Настраиваем Capacitive touch screen.

В нашем случае он на FT5446, и для него подходят драйвера ft5x06 которые и так есть в linux. Надо только скомпилировать драйвера и приписать соответствующие строчки описания железа в dts файл.

Нас интересуют следующие файлы:

```
sun8i-v3s-licheepi-zero.dts
```

```
sun8i-v3s.dtsi
```

В файле sun8i-v3s.dtsi в секции pio: надо определить секцию

```
edt_ft5x06_pins: edt_ft5x06_pins@0 {
    pins = "PB0", "PB1";
    function = "gpio_in";
};
```

Это наши пины для reset и interrupt для touch screen.

В файле sun8i-v3s-licheepi-zero.dts:

Добавить секцию (и убрать секцию со старым resistive touch в колторой ns2009 драйвера используются)

```
&i2c0 {
```

```
status = "okay";
```

```
polytouch: edt-ft5x06@38 {
```

```
    compatible = "edt,edt-ft5406";
```

```
    reg = <0x38>;
```

```
    interrupt-parent = <&pio>;
```

```
    interrupts = <1 3 IRQ_TYPE_EDGE_FALLING>; //PB3
```

```
    reset-gpios = <&pio 1 2 GPIO_ACTIVE_LOW>; //PB2
```

```
    pinctrl-names = "default";
```

```
    pinctrl-0 = <&edt_ft5x06_pins>;
```

```
};
```

```
};
```

Находятся они по таким путям

```
output/build/linux-zero-4.14.y/arch/arm/boot/dts/sun8i-v3s.dtsi
```

```
output/build/linux-zero-4.14.y/arch/arm/boot/dts/sun8i-v3s-licheepi-zero.dts
```

Так-же надо разрешить наши touch драйвера (и запретить от resistive touch screen)

```
make linux-nconfig
```

И ищем там настройки touchscreen.

Device Drivers->Input device support->Touchscreens

Nsiway NS2009 touchscreen - убрать звездочку

EDT FocalTech FT5x06 I2C Touchscreen support - поставить звездочку

после этого пересобираем

```
make linux-rebuild
```

```
make
```

Touch подключается так (можно по схеме посмотреть)

PB3 - PIN4 INT

PB2 - PIN3 RST

PB6 - PIN6 SCL

PB7 - PIN5 SDA

Ну и GND, VCC соответственно к земле и 3.3V

Само устройство будет по пути.

/dev/input/event1

Можно также запустить на устройстве.

i2cdetect -y 0

и увидеть

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00:				--	--	--	--	--	--	--	--	--	--	--	--	--
10:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
20:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30:	--	--	--	--	--	--	--	--	UU	--	--	--	--	--	--	--
40:	--	--	--	--	--	--	--	--	48	--	--	--	--	--	--	--
50:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
60:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
70:	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

UU на адресе 0x38 значит, что он используется драйвером Touch.

Настройка WiFi.

Собственно говоря тут всё работает практически «из коробки». Надо лишь загрузить драйвер r8723bs.ko (т. к. он не встроен в ядро) и настроить wpa_supplicant.

Для этого добавил файл /etc/init.d/S30wifi_driver в процесс загрузки.

Этот файл можно ручками скопировать после сборки. Но лично мне кажется, что лучше встроить это в процесс сборки, тем более buildroot предлагает такой вариант.

В файле .config изменяем на BR2_ROOTFS_OVERLAY="../buildroot_additional/rootfs_overlay" и всё, что в папке rootfs_overlay будет копироваться на наш диск с linux.

Далее, необходимо добавить и настроить wpa_supplicant.

Запускается он в файле rootfs_overlay/etc/network/interfaces

Файл /etc/balmer_SSID.conf - это имя нашей WiFi сети и пароль к ней.

Генерируется этот файл при помощи команды wpa_passphrase wifi_name wifi_password > balmer_SSID.conf

После этого делаем make и в нашем образе диска должны появиться файлы

/etc/balmer_SSID.conf

/etc/init.d/S30wifi_driver

/etc/network/interfaces

После этого WiFi будет запускаться и подключаться в процессе загрузки устройства.

После загрузки можно запустить ifconfig и увидеть, что появился wlan0

wlan0 Link encap:Ethernet HWaddr CC:D2:9B:4B:8C:AD

inet addr:192.168.1.46 Bcast:192.168.1.255 Mask:255.255.255.0

либо воспользоваться командами ping или wget которые есть в сборке по умолчанию.

Как отключить вывод консоли на экран.

В файле board/licheepi/boot.cmd заменяем первую строчку на такую

setenv bootargs console=ttyS0,115200 panic=5 rootwait consoleblank=0

vt.global_cursor_default=0 root=/dev/mmcblk0p2 earlyprintk rw

Это отключит как вывод консольных сообщений на экран, так и мигающий курсор.