

目录

Tina SDK 构建指南	1
环境配置	1
WSL 配置	1
安装构建工具	1
安装 REPO 工具	1
配置保存身份认证	2
SDK 拉取	2
新建文件夹保存 SDK	2
初始化 REPO 仓库	2
拉取 SDK	3
创建开发环境	3
SDK 配置	3
解决启动脚本报错	3
合入 100ASK 的 T113-S3 补丁	3
初始化编译环境	4
Tina SDK 配置	4
从基础配置开始	4
从 t113_pi 开始	4
修改 U-Boot 打印串口	5
适配 86 盒屏幕	6
适配 WIFI	10
解决编译报错 configure: error: XML::Parser perl module is required for intltool	13
解决 OpenSSH 编译问题	13
编译, 打包	14
烧录	14
FAQ	15
Tina-Linux 存储介质切换	15
Tina-Linux 存储开发指南	15
Tina-Linux 屏幕适配	15
Tina-Linux 修改日志串口	15

T113-S3 温度传感器.....	15
解决开机 overlay 分区挂不上的问题.....	15
解决 OpenSSH 编译问题.....	15
解决编译报错 configure: error: XML::Parser perl module is required for intltool	16
解决分区过小问题.....	16
解决服务无法自动启动问题	16
解决找不到 lo 网卡的问题.....	16
WIFI Manage2.0.....	16
解决 WIFI Manage 1.0 服务无法启动问题	16
如何连接 WIFI	16
解决 RGB 频率过高花屏问题.....	17
启动优化	17
修复 Warning: Using internal RC 16M clock source. Time may be inaccurate!	17
升级 libcurl 版本.....	18
升级 freetype 版本.....	19
升级 htop 版本.....	20
解决 tplayer 1080P 解码问题.....	20
升级 LVGL 版本.....	20
RTL8723BS 驱动	21
解决 ADB Type-C 空指针报错问题	21
添加 fftw3	21
添加 libjpeg-turbo.....	21
添加 tmux	21
解决 SPI Nand 无法启动烧录问题.....	22
Tina 无法重启.....	22
RGB 屏幕缺几行.....	23
U-Boot 背光没法用.....	23
开机 LOGO.....	23
解决 ADB 烧录问题.....	23

Tina SDK 构建指南

环境配置

WSL 配置

按照要求配置 WSL 环境，具体过程可参考百度，建议系统为 Ubuntu 20。

安装构建工具

进入系统后，输入下方命令安装需要的工具

```
sudo apt update -y
sudo apt full-upgrade -y
sudo apt install -y ack antlr3 asciidoc autoconf automake autopoint binutils bison
build-essential \
bzip2 ccache cmake cpio curl device-tree-compiler fastjar flex gawk gettext gcc-
multilib g++-multilib \
git gperf haveged help2man intltool libc6-dev-i386 libelf-dev libfuse-dev
libglib2.0-dev libgmp3-dev \
libltdl-dev libmpc-dev libmpfr-dev libncurses5-dev libncursesw5-dev libpython3-
dev libreadline-dev \
libssl-dev libtool lrzsz mkisofs msmtmp ninja-build p7zip p7zip-full patch pkgconf
python2.7 python3 \
python3-pyelftools python3-setuptools qemu-utils rsync scons squashfs-tools
subversion swig texinfo \
uglifyjs upx-ucl unzip vim wget xmlto xxd zlib1g-dev
```

安装 REPO 工具

repo 是 Google 开发的用于管理 Android 版本库的一个工具，repo 是使用 Python 对 git 进行了一定的封装，并不是用于取代 git，它简化了对多个 Git 版本库的管理。用 repo 管理的版本库都需要使用 git 命令来进行操作。因此，使用 repo 工具之前，请先确保已经安装 git。为什么要用 repo？项目模块化/组件化之后各模块也作为独立的 Git 仓库从主项目里剥离了出去，各模块各自管理自己的版本。Android 源码引用了很多开源项目，每一个子项目都是一个 Git 仓

库, 每个 Git 仓库都有很多分支版本, 为了方便统一管理各个子项目的 Git 仓库, 需要一个上层工具批量进行处理, 因此 repo 诞生。repo 也会建立一个 Git 仓库, 用来记录当前 Android 版本下各个子项目的 Git 仓库分别处于哪一个分支, 这个仓库通常叫做: manifest 仓库(清单库)。

```
sudo -i # 需要提权到 root 用户
curl https://mirrors.bfsu.edu.cn/git/git-repo > /usr/bin/repo
chmod +x /usr/bin/repo
```

安装完成后, 切换到普通用户, 然后更换为国内镜像源:

```
echo export REPO_URL='https://mirrors.bfsu.edu.cn/git/git-repo' >> ~/.bashrc
source ~/.bashrc
```

配置保存身份认证

新版本 git 默认加强了安全性, 身份认证不会保存, 导致拉取 repo 需要多次输入密码, 可以用下列命令配置:

```
git config --global credential.helper store
```

SDK 拉取

参考链接:

<https://bbs.aw-ol.com/topic/3947/%E6%96%B0-sdk-%E5%B9%B3%E5%8F%B0%E4%B8%8B%E8%BD%BD-d1-h-d1s-sdk>

d1s 芯片与 T113-S3 芯片除内核不同以外, 其他外设基本相同, 此处直接拉取 d1 内核, 然后合并 T113-S3 芯片方案配置即可。

新建文件夹保存 SDK

使用 mkdir 命令新建文件夹, 保存之后需要拉取的 SDK, 然后 cd 进入到刚才新建的文件夹中。

```
mkdir tina-sdk
cd tina-sdk
```

初始化 REPO 仓库

使用 repo init 命令初始化仓库, tina-d1-h 的仓库地址是:

https://sdk.aw-ol.com/git_repo/D1_Tina_Open/manifest.git

执行以下命令, 初始化 repo 仓库:

```
repo init -u https://sdk.aw-ol.com/git_repo/D1_Tina_Open/manifest.git -b master -m tina-d1-h.xml
```

如果提示 Username for 'https://sdk.aw-ol.com': 请输入 全志在线开发者论坛 的用户名和密码。(注: 需要全志在线开发者论坛 LV2 等级以上用户才有权限拉取 SDK)

拉取 SDK

使用命令 `repo sync` 拉取 SDK

```
repo sync
```

由于 SDK 普遍较大, 拉取可能需要一定的时间。

创建开发环境

使用命令 `repo start` 创建开发环境分支

```
repo start product-smartx-d1-h-tina-stable-v2.0 --all
```

至此, SDK 获取完毕。

SDK 配置

解决启动脚本报错

当前克隆项目运行 `source build/envsetup.sh` 会存在以下报错:

```
-bash: build/envsetup.sh: line 1: syntax error near unexpected token `${r}'  
'bash: build/envsetup.sh: line 1: `function hmm() {
```

解决方法: 使用 `dos2unix` 批量将文件转换为 unix 格式

```
sudo apt install dos2unix  
find -type f | xargs dos2unix
```

合入 100ASK 的 T113-S3 补丁

参考链接: https://gitee.com/weidongshan/100ASK_T113-Pro_TinaSDK

此处需要在根目录操作, 否则会污染下载到的 tina-sdk

```
book@ubuntu1804:~$ git clone https://github.com/DongshanPI/100ASK_T113-Pro_TinaSDK.git  
book@ubuntu1804:~$ cd 100ASK_T113-Pro_TinaSDK  
book@ubuntu1804:~/100ASK_T113-Pro_TinaSDK$ git submodule update --init  
book@ubuntu1804:~/100ASK_T113-Pro_TinaSDK$ cp ./ * -rfvd ~/tina-sdk
```

合入完成后, 进入 tina-sdk 目录, 使用下列命令可以查看到更新:

```
repo forall -vc "git status"
```

常用 repo 命令参考：
撤销当前所有更改

```
repo forall -vc "git reset --hard"
```

清理所有未追踪文件

```
repo forall -vc "git clean -fd"
```

初始化编译环境

```
cd tina-sdk  
source build/envsetup.sh  
lunch
```

You're building on Linux

Lunch menu... pick a combo:

1. d1-h_nezha-tina
2. d1-h_nezha_min-tina
3. d1s_nezha-tina
4. t113_100ask-tina

Which would you like? [Default d1s_nezha]: 4

Tina SDK 配置

从基础配置开始

按照上面步骤进入编译环境后，输入下列命令，可以进入配置界面

```
make menuconfig
```

输入下列命令，可以进入 linux 内核配置

```
make kernel_menuconfig
```

从 t113_pi 开始

下面将从已配置好的设备树和配置开始配置，其中问题可参考下一章 FAQ
设备树文件：

<https://gist.github.com/FanhuaCloud/08a26e460167d9473286186cfbea2a5>

b

替换: tina-sdk/device/config/chips/t113/configs/100ask/board.dts

内核配置:

<https://gist.github.com/FanhuaCloud/b6a6312dc18abdebb5b1c74addb1fb6>

f

替换: tina-sdk/device/config/chips/t113/configs/100ask/linux-5.4/config-5.4

uboot 环境变量:

<https://gist.github.com/FanhuaCloud/f76f9ab87eec0edf380f621225124929>

替换: tina-sdk/device/config/chips/t113/configs/100ask/env.cfg

系统配置:

<https://gist.github.com/FanhuaCloud/da2ca3f66320055f83594338c88587e7>

替换: tina-sdk/device/config/chips/t113/configs/100ask/sys_config.fex

系统分区配置 (默认的不够用):

<https://gist.github.com/FanhuaCloud/c7c03fe4ca26b6776a5cc9ba257bf369>

替换: tina-sdk/device/config/chips/t113/configs/100ask/sys_partition.fex

修改 U-Boot 打印串口

参考文档:

<https://bbs.awol.com/topic/1673/%E5%85%A8%E5%BF%97%E8%8A%AF%E7%89%87tina-linux-%E4%BF%AE%E6%94%B9-uart-%E5%BC%95%E8%84%9A-uart%E7%AB%AF%E5%8F%A3>

从 device/config/chips/t113/configs/100ask/BoardConfig.mk 中找到使用的 U-Boot 配置, 将其打印串口改为 UART0

文件内容可参考下方:

```
LICHEE_CHIP:=sun8iw20p1
LICHEE_ARCH:=arm
LICHEE_BRANDY_VER:=2.0
LICHEE_BRANDY_DEFCONF:=sun8iw20p1_uart3_defconfig
LICHEE_KERN_VER:=5.4
LICHEE_KERN_DEFCONF:=sun8iw20p1smp_defconfig
LICHEE_REDUNDANT_ENV_SIZE:=0x20000
```

可以发现使用的 U-Boot 配置为 sun8iw20p1_uart3_defconfig, 找到文件并修改

tina-sdk/lichee/brandy-2.0/u-boot-2018/configs/sun8iw20p1_uart3_defconfig

其中有一段：

```
# Serial drivers
CONFIG_BAUDRATE=115200
CONFIG_SPECIFY_CONSOLE_INDEX=y
CONFIG_CONS_INDEX=1
CONFIG_SYS_NS16550=y
```

CONFIG_CONS_INDEX 就是 U-Boot 日志打印的串口，此处从 1 开始，我们的串口为 UART0，所以将该参数修改为 1 即可。

适配 86 盒屏幕

86 屏使用的是 ST7701 芯片，RGB 接口，需要 SPI 初始化才可以使用，所以需要增加初始化代码。

参考文档：

https://blog.csdn.net/weixin_43094346/article/details/127082843

根据文档，首先添加一个新的屏幕，代码直接参考如下：

创建 st7701_86.h st7701_86.c，下方直接给出代码，此处使用了模拟 IO 来实现 9 bit spi:

<https://gist.github.com/FanhuaCloud/b0f72ceb4ec5e8cefe6c0d2d65fb12cb>

<https://gist.github.com/FanhuaCloud/cbb54819bd40cab498134bc73074208>

8

将文件放入下方文件中：

tina-sdk/lichee/linux-5.4/drivers/video/fbdev/sunxi/disp2/disp/lcd

修 改 tina-sdk/lichee/linux-5.4/drivers/video/fbdev/sunxi/disp2/disp/lcd/Kconfig

在最后增加刚刚创建的屏幕：

```
config LCD_SUPPORT_ST7701_86
    bool "LCD support ST7701_86 panel"
    default y
    ---help---
        If you want to support ST7701_86 panel for display driver,
select it.
```

同时在 tina-sdk/lichee/linux-5.4/drivers/video/fbdev/sunxi/disp2/disp/lcd 文件夹内找到 panel.c 和 panel.h 两个文件，修改 panel.c 增加屏幕指针。并使用 ifdef 宏来确定只有启用这款屏幕的时候才会编译这个屏幕的驱动。指针的声明位于刚才修改的头文件中，需要确认声明与 panel.c 中的名称是否一致。


```
125     &K080_IM2HYL802R_800X1280_mipi_panel,
126 #endif
127 #ifdef CONFIG_LCD_SUPPORT_K101_IM2BYL02_L_800X1280
128     &K101_IM2BYL02_L_800X1280_mipi_panel,
129 #endif
130 #ifdef CONFIG_LCD_SUPPORT_NT35510_MIPI
131     &nt35510_panel,
132 #endif
133 #ifdef CONFIG_LCD_SUPPORT_ST7701S_3SPI
134     &st7701s_3spi_panel,
135 #endif
136 #ifdef CONFIG_LCD_SUPPORT_D395T9375V0
137     &d395t9375v0_panel,
138 #endif
139 #ifdef CONFIG_LCD_SUPPORT_ST7701_86
140     &st7701_86_panel,
141 #endif
142     &super_lcd_panel,
143     NULL,
144 };
145
146 void lcd_set_panel_funs(void)
147 {
148     int i;
149
150     for (i = 0; panel_array[i] != NULL; i++) {
151         sunxi_lcd_set_panel_funs(panel_array[i]->name,
152                                 &panel_array[i]->func);
153     }
154 }
155
156 #endif
157
158 :set number
```

```
#ifdef CONFIG_LCD_SUPPORT_ST7701_86
    &st7701_86_panel,
#endif
```

```
163 #endif
164 #ifdef CONFIG_LCD_SUPPORT_K101_IM2BYL02_L_800X1280
165 extern struct __lcd_panel K101_IM2BYL02_L_800X1280_mipi_panel;
166 #endif
167 #ifdef CONFIG_LCD_SUPPORT_K080_IM2HYL802R_800X1280
168 extern struct __lcd_panel K080_IM2HYL802R_800X1280_mipi_panel;
169 #endif
170 #ifdef CONFIG_LCD_SUPPORT_NT35510_MIPI
171 extern struct __lcd_panel nt35510_panel;
172 #endif
173 #ifdef CONFIG_LCD_SUPPORT_ST7701S_3SPI
174 extern struct __lcd_panel st7701s_3spi_panel;
175 #endif
176 #ifdef CONFIG_LCD_SUPPORT_D395T9375V0
177 extern struct __lcd_panel d395t9375v0_panel;
178 #endif
179 #ifdef CONFIG_LCD_SUPPORT_ST7701_86
180 extern struct __lcd_panel st7701_86_panel;
181 #endif
182
183 extern struct __lcd_panel super_lcd_panel;
184
185 #endif
186
187 ~
188 ~
189 ~
190 ~
191 ~
192
193 :set number
```

```
#ifdef CONFIG_LCD_SUPPORT_ST7701_86
extern struct __lcd_panel st7701_86_panel;
#endif
```

接着添加屏幕到上一级目录 tina-sdk/lichee/linux-5.4/drivers/video/fbdev/sunxi/disp2/disp 的 Makefile 中:

```

he@DESKTOP-DVJPBLU: ~/tin x + v
disp-$(CONFIG_LCD_SUPPORT_ST7701S) += lcd/st7701s.o
disp-$(CONFIG_LCD_SUPPORT_WTL096601G03) += lcd/wtl096601g03.o
disp-$(CONFIG_LCD_SUPPORT_T30P106) += lcd/t30p106.o
disp-$(CONFIG_LCD_SUPPORT_T020T20000) += lcd/to20t20000.o
disp-$(CONFIG_LCD_SUPPORT_S2003T46G) += lcd/s2003t46g.o
disp-$(CONFIG_LCD_SUPPORT_WILLIAMLCD) += lcd/williamlcd.o
disp-$(CONFIG_LCD_SUPPORT_LQ101R1SX03) += lcd/lq101r1sx03.o
disp-$(CONFIG_LCD_SUPPORT_INET_DSI_PANEL) += lcd/inet_dsi_panel.o
disp-$(CONFIG_LCD_SUPPORT_RT13QV005D) += lcd/rt13qv005d.o
disp-$(CONFIG_LCD_SUPPORT_ILI9881C) += lcd/ili9881c.o
disp-$(CONFIG_LCD_SUPPORT_TM_DSI_PANEL) += lcd/tm_dsi_panel.o
disp-$(CONFIG_LCD_SUPPORT_ST7789V_CPU) += lcd/st7789v_cpu.o
disp-$(CONFIG_LCD_SUPPORT_JD9366AB_3) += lcd/jd9366ab_3.o
disp-$(CONFIG_LCD_SUPPORT_TFT08006) += lcd/tft08006.o
disp-$(CONFIG_LCD_SUPPORT_BP101WX1_206) += lcd/bp101wx1-206.o
disp-$(CONFIG_LCD_SUPPORT_FX070) += lcd/fx070.o
disp-$(CONFIG_LCD_SUPPORT_K101IM2QA04) += lcd/k101im2qa04.o
disp-$(CONFIG_LCD_SUPPORT_CC08021801_310_800X1280) += lcd/CC08021801_310_800X1280.o
disp-$(CONFIG_LCD_SUPPORT_K080_IM2HYL802R_800X1280) += lcd/K080_IM2HYL802R_800X1280.o
disp-$(CONFIG_LCD_SUPPORT_K101_IM2BYL02_L_800X1280) += lcd/K101_IM2BYL02_L_800X1280.o
disp-$(CONFIG_LCD_SUPPORT_NT35510_MIPI) += lcd/nt35510.o
disp-$(CONFIG_LCD_SUPPORT_ST7701S_3SPI) += lcd/st7701s_3spi.o
disp-$(CONFIG_LCD_SUPPORT_D395T9375V0) += lcd/d395t9375v0.o
disp-$(CONFIG_LCD_SUPPORT_ST7701_86) += lcd/st7701_86.o
disp-y += lcd/super_lcd_driver.o

disp-$(CONFIG_EINK_PANEL_USED) += de/disp_eink_manager.o \
de/eink_buffer_manager.o de/eink_pipeline_manager.o \
de/disp_format_convert.o lcd/default_eink.o
70,55 95%

```

disp-\$(CONFIG_LCD_SUPPORT_ST7701_86) += lcd/st7701_86.o

make kernel_menuconfig 后，到 Device Drivers > Graphics support > Frame buffer Devices > Video support for sunxi > LCD panels select 找到屏幕，勾选即可。

```

he@DESKTOP-DVJPBLU: ~/tin x + v
.config - Linux/arm 5.4.61 Kernel Configuration
> Device Drivers > Graphics support > Frame buffer Devices > Video support for sunxi > LCD panels select
LCD panels select
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters
are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?>
for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

^(-)
[ ] LCD support inet_dsi_panel panel
[ ] LCD support WTL096601G03 panel
[ ] LCD support rt13qv005d panel
[ ] LCD support st7789v_cpu panel
[ ] LCD support CC08021801_310_800X1280 panel
[ ] LCD support JD9366AB_3 panel
[*] LCD support TFT08006 panel
[ ] LCD support bp101wx1-206 panel
[ ] LCD support fx070 panel
[ ] LCD support K101IM2QA04 panel
[ ] LCD support K101_IM2BYL02_L_800X1280 panel
[ ] LCD support K080_IM2HYL802R_800X1280 panel
[ ] LCD support nt35510_mipi panel (NEW)
[ ] LCD support st7701s_3spi panel (NEW)
[ ] LCD support D395T9375V0 panel (NEW)
[*] LCD support ST7701_86 panel

<Select> < Exit > < Help > < Save > < Load >

```

设备树已经配置完成，可参考下方，lcd_gpio 按照原理图配置即可：

```

&lcd0 {
    /* part 1 */
    lcd_used          = <1>;
    lcd_driver_name   = "st7701_86";
    lcd_backlight     = <255>;

    /* part 2 */
    lcd_if = <0>;
}

```

```

lcd_hv_if = <0>;

/* part 3 */
lcd_x          = <480>;
lcd_y          = <480>;
lcd_width     = <71>;
lcd_height    = <71>;
lcd_dclk_freq = <24>;

lcd_hbp       = <20>;
lcd_ht        = <526>;
lcd_hspw      = <6>;

lcd_vbp       = <10>;
lcd_vt        = <510>;
lcd_vspw      = <4>;

lcd_pwm_used  = <1>;
lcd_pwm_ch    = <7>;
lcd_pwm_freq  = <5000>;
lcd_pwm_pol   = <0>;

/* part 5 */
lcd_frm = <1>;
lcd_io_phase = <0x0000>;
lcd_gamma_en = <0>;
lcd_cmap_en = <0>;
lcd_hv_clk_phase = <0>;
lcd_hv_sync_polarity = <0>;

// reset
lcd_gpio_0 = <&pio PB 4 GPIO_ACTIVE_HIGH>;
// cs
lcd_gpio_1 = <&pio PB 7 GPIO_ACTIVE_HIGH>;
// sda
lcd_gpio_2 = <&pio PB 5 GPIO_ACTIVE_HIGH>;
// sck
lcd_gpio_3 = <&pio PB 6 GPIO_ACTIVE_HIGH>;

/* part 6 */
lcd_power = "vcc-ld";
lcd_pin_power = "vcc-pd";
pinctrl-0 = <&rgb18_pins_a>;
pinctrl-1 = <&rgb18_pins_b>;

```

```
};
```

至此屏幕适配完成,需要注意的是不同厂商的 86 屏初始化代码和时序不同,此处代码和时序仅供参考。

适配 WIFI

修改 tina-sdk/package/kernel/linux/modules/wireless.mk, 替换掉原来的 8723bs 配置, 代码参考下方:

```
he@DESKTOP-DVJPLBU: ~/tin × + v
18 # CONFIG_88EU_P2P=n
19 FILES:=$(LINUX_DIR)/drivers/net/wireless/rtl8188eu/8188eu.ko
20 AUTOLOAD:=$(call AutoProbe,8188eu)
21 endif
22
23 define KernelPackage/net-rtl8188eu/description
24 Kernel modules for RealTek RTL8188EU support
25 endif
26
27 $(eval $(call KernelPackage,net-rtl8188eu))
28
29 define KernelPackage/net-rtl8723bs
30 SUBMENU:=$(WIRELESS_MENU)
31 TITLE:=RTL8723BS support (staging)
32 DEPENDS:=@USB_SUPPORT +@DRIVER_WEXT_SUPPORTi # +r8723bs-firmware
33 # KCONFIG:=\
34 # CONFIG_STAGING=y \
35 # CONFIG_R8723BS \
36 # CONFIG_23BS_AP_MODE=y \
37 # CONFIG_23BS_P2P=n
38 FILES:=$(LINUX_DIR)/drivers/staging/rtl8723bs/r8723bs.ko
39 AUTOLOAD:=$(call AutoProbe,8723bs)
40 endif
41 |
42 define KernelPackage/net-rtl8723bs/description
43 Kernel modules for RealTek RTL8723BS support
44 endif
45
46 $(eval $(call KernelPackage,net-rtl8723bs))
:set number
```

```
define KernelPackage/net-rtl8723bs
SUBMENU:=$(WIRELESS_MENU)
TITLE:=RTL8723BS support (staging)
DEPENDS:=@USB_SUPPORT +@DRIVER_WEXT_SUPPORTi # +r8723bs-
firmware
# KCONFIG:=\
# CONFIG_STAGING=y \
# CONFIG_R8723BS \
# CONFIG_23BS_AP_MODE=y \
# CONFIG_23BS_P2P=n
FILES:=$(LINUX_DIR)/drivers/staging/rtl8723bs/r8723bs.ko
AUTOLOAD:=$(call AutoProbe,8723bs)
endif
```

修改内核中 RTL8723BS 的源码, 文件位于 tina-sdk/lichee/linux-5.4/drivers/staging/rtl8723bs/os_dep, 增加

VFS_internal_I_am_really_a_filesystem_and_am_NOT_a_driver，部分代码参考如下：

```
// SPDX-License-Identifier: GPL-2.0
/*****
 *
 * Copyright(c) 2007 - 2011 Realtek Corporation. All rights reserved.
 *
 *****/
#define _OS_INTFS_C_

#include <drv_types.h>
#include <rtw_debug.h>
#include <hal_data.h>

MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Realtek Wireless Lan Driver");
MODULE_AUTHOR("Realtek Semiconductor Corp.");
MODULE_VERSION(DRIVERVERSION);
MODULE_IMPORT_NS(VFS_internal_I_am_really_a_filesystem_and_am_NOT_a_driver);
```

增加了

```
MODULE_IMPORT_NS(VFS_internal_I_am_really_a_filesystem_and_am_NOT_a_driver);
```

删除 tina-sdk/target/allwinner/t113-100ask 下的 base-files，替换为附件中提供的 base-files，里面提供了 WIFI 初始化脚本以及相关的固件。

修改 tina-sdk/package/allwinner/wifimanager/files/wpa_supplicant.init，增加初加载驱动和 DHCP 命令，部分内容参考如下：

```
USE_PROCD=1
DEPEND=boot
PROG=/usr/sbin/wpa_supplicant
OOM_ADJ=-17

start_service() {
    procd_open_instance
```

```

#   wpa=`ps | grep wpa_supplicant | grep -v grep`
#   [ -n "$wpa" ] && {
#       killall  wpa_supplicant
#           sleep 1
#   }
/sbin/insmod /lib/modules/5.4.61/r8723bs.ko
ifconfig wlan0 up
for i in $(seq 1 5)
do
    wlan=`ifconfig | grep wlan0 | awk '{print $1}'`
    if [ x$wlan=="xwlan0" ]; then
        usleep 500000
        procd_set_param oom_adj $OOM_ADJ
        procd_set_param command $PROG -iwlan0 -Dnl80211 -
c/etc/wifi/wpa_supplicant.conf -l/etc/wifi/wpa_supplicant_overlay.conf -
O/etc/wifi/sockets
        procd_close_instance
    else
        echo "====try again up wlan0...===="
        usleep 500000
        ifconfig wlan0 up
    fi
done

wlan=`ifconfig | grep wlan0 | awk '{print $1}'`
if [ x$wlan=="xwlan0" ]; then
    echo "====wlan0 up succeeded!===="
else
    echo "====wlan0 up failed!===="
fi

brlan=`ifconfig | grep br-lan | awk '{print $1}'`
if [ x$brlan=="xbr-lan" ]; then
    echo "change the default ip address 192.168.1.1"

```

```
        ifconfig br-lan 192.168.100.100
    fi
    udhcpc -iwlan0 &
}
}
```

解决编译报错 configure: error: XML::Parser perl module is required for intltool

make menuconfig

关闭:

Libraries --->xkeyboard-config

Libraries --->libxkbcommon

解决 OpenSSH 编译问题

编译 OpenSSH 时，由于系统的 OpenSSL 版本问题，可能会导致编译失败，常见错误如下：

```
./libssh.a(ssh-pkcs11.o): In function `pkcs11_add_provider':
ssh-pkcs11.c:(.text+0x18d0): undefined reference to `EC_KEY_free'
./libssh.a(kexgen.o): In function `input_kex_gen_init':
kexgen.c:(.text+0x3f4): undefined reference to `kex_ecdh_enc'
./libssh.a(kexgen.o): In function `input_kex_gen_reply':
kexgen.c:(.text+0x6a4): undefined reference to `kex_ecdh_dec'
./libssh.a(kexgen.o): In function `kex_gen_client':
kexgen.c:(.text+0x7a8): undefined reference to `kex_ecdh_keypair'
```

或者：<https://forums.100ask.net/t/topic/4089>

我们忽略这些错误，直接升级 OpenSSH 的版本，就可以解决这些问题：

参考文档：<https://bbs.aw-ol.com/topic/5035/%E5%85%A8%E5%BF%97t113-s3%E8%A7%A3%E5%86%B3tina-sdk-openssh%E7%BC%96%E8%AF%91%E9%97%AE%E9%A2%98>

修改 tina-sdk/package/network/openssh/Makefile，直接将版本修改为 8.9p1，即可解决编译问题。

编译，打包

输入下列命令，开始构建系统

```
make -j16
```

-j16 指的是使用 16 线程编译，实际编译时可按照电脑配置来选择使用多少线程编译。

输入下列命令，开始单独构建 U-Boot

```
mboot
```

输入下列命令，开始单独构建内核（构建系统会顺带构建内核）

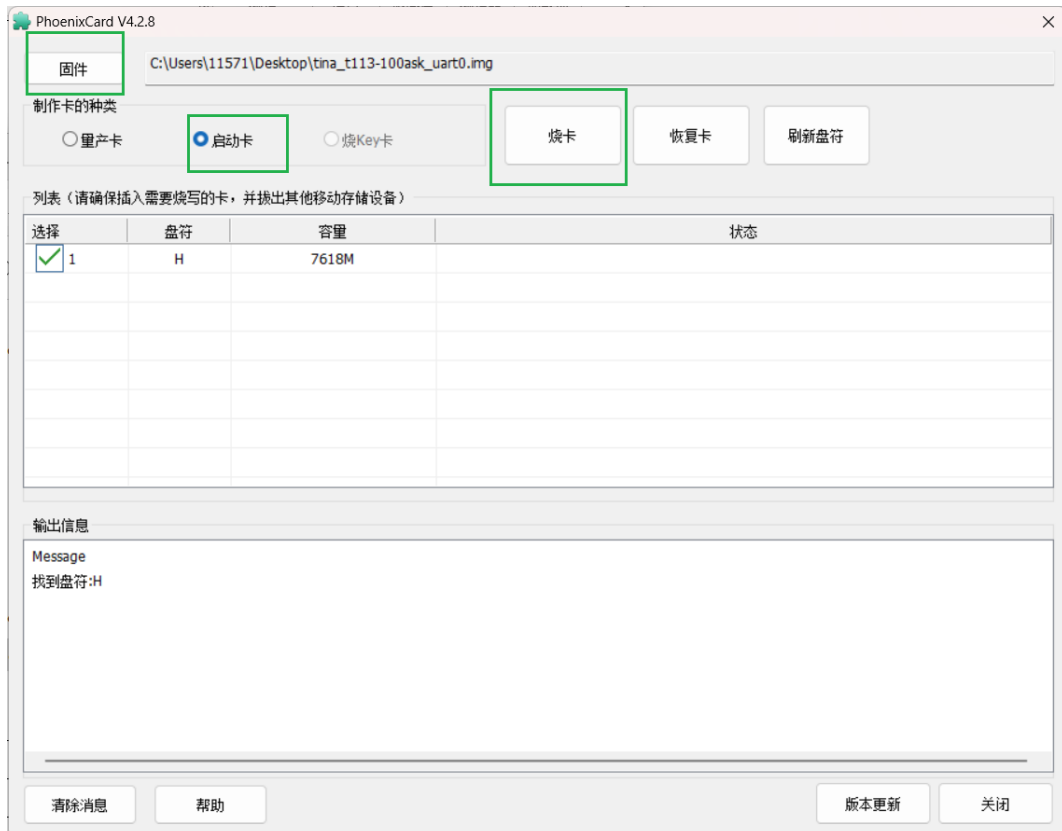
```
mkernel
```

输入下列命令，开始打包系统

```
pack
```

烧录

将 SD 卡插入电脑，打开 PhoenixCard.exe，点击固件，加载 img 固件，然后选择启动卡，再点击烧卡即可，需要注意的是，一定要选择启动卡，否则会启动失败！。



FAQ

Tina-Linux 存储介质切换

<https://bbs.aw-ol.com/topic/1701/tina-linux-%E5%AD%98%E5%82%A8%E4%BB%8B%E8%B4%A8%E5%88%87%E6%8D%A2-emmc-spi-nand-spi-nor-sd-card-sd-nand/2?lang=zh-CN>

Tina-Linux 存储开发指南

<https://zhuanlan.zhihu.com/p/610032464>

Tina-Linux 屏幕适配

https://blog.csdn.net/weixin_43094346/article/details/127082843

Tina-Linux 修改日志串口

<https://bbs.aw-ol.com/topic/1673/%E5%85%A8%E5%BF%97%E8%8A%AF%E7%89%87tina-linux-%E4%BF%AE%E6%94%B9-uart-%E5%BC%95%E8%84%9A-uart%E7%AB%AF%E5%8F%A3>

T113-S3 温度传感器

<https://bbs.aw-ol.com/assets/uploads/files/1648476032180-d1-tina-linux-%E6%B8%A9%E5%BA%A6%E6%8E%A7%E5%88%B6-%E4%BD%BF%E7%94%A8%E6%8C%87%E5%8D%97.pdf>

解决开机 overlay 分区挂不上的问题

默认 Tina-Linux 配置可能没有选中 e2fsprogs 包，导致第一次开机无法执行 ext4 格式化数据分区，我们只要在配置里面开启 e2fsprogs 即可。

解决 OpenSSH 编译问题

请参考上一章文档。

解决编译报错 configure: error: XML::Parser perl module is required for intltool

请参考上一章文档。

解决分区过小问题

https://blog.csdn.net/TQ_wangbin/article/details/133377483

解决服务无法自动启动问题

参考文档:

<https://bbs.aw-ol.com/topic/1889/tina-procd-init-%E4%B8%8E-busybox-init-%E5%88%87%E6%8D%A2>

解决方法为 busybox-init 修改为 procd-init

解决找不到 lo 网卡的问题

添加 tina-sdk/target/allwinner/t113-100ask/base-files/etc/config/network, 增加 lo 网卡即可, 内容如下:

```
config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'
```

WIFI Manage2.0

https://blog.csdn.net/weixin_43094346/article/details/127221494

解决 WIFI Manage 1.0 服务无法启动问题

参考“适配 WIFI”章节, 服务文件中, 改动 PROG 变量即可, 受影响的行参考如下:

```
PROG=/usr/sbin/wpa_supplicant
```

如何连接 WIFI

参考文档: <https://zhuanlan.zhihu.com/p/610031664>

```
wifi_connect_ap_test <ssid> <passwd>
```

解决 RGB 频率过高花屏问题

https://blog.csdn.net/weixin_43094346/article/details/127488702

按照文档修改该文件即可：

tina-sdk/lichee/linux-

5.4/drivers/video/fbdev/sunxi/disp2/disp/de/lowlevel_v2x/disp_al.c

启动优化

<https://cloud.tencent.com/developer/article/2386880>

或者参考：Tina_Linux_启动优化_开发指南.pdf

修改 tina-sdk/device/config/chips/t113/configs/100ask/u-boot-board.dts

修改 &target 中的 burn_key 设置为 0，&platform 中的 debug_mode 设置为

0

修改 tina-sdk/device/config/chips/t113/configs/100ask/sys_config.fex

修改 platform 下的 debug_mode 为 0 即可

修复 Warning: Using internal RC 16M clock source. Time may be inaccurate!

修改 tina-sdk/lichee/linux-5.4/drivers rtc/rtc-sunxi.c,将 val & LOSC_STATUS) == 0 修改为 == 1 即可。

参考数据手册，当寄存器值为 1 时，才说明有错误，此处为代码问题：

Offset:0x0004			Register Name: LOSC_AUTO_SWT_STA_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R	0x0	EXT_LOSC_STA Work only when the auto switch function is enabled. 0: External 32.768 kHz OSC work normally 1: External 32.768 kHz OSC work abnormally

修改后的代码如图：

```
he@DESKTOP-DVJPBLU: ~/tin x + v
321     dev_info(dev, "%04d-%02d-%02d %02d:%02d:%02d\n",
322             TM_YEAR_TO_REAL_YEAR(tm->tm_year),
323             TM_MON_TO_REAL_MON(tm->tm_mon), tm->tm_mday,
324             tm->tm_hour, tm->tm_min, tm->tm_sec);
325 }
326
327 static int sunxi_rtc_gettime(struct device *dev, struct rtc_time *tm)
328 {
329     struct sunxi_rtc_dev *chip = dev_get_drvdata(dev);
330     u32 hw_day, hw_hhmmss;
331     u32 val;
332     int err;
333
334     val = rtc_reg_read(chip, LOSC_AUTO_SWT_STA_REG);
335     if ((val & LOSC_STATUS) == 1)
336         dev_warn(dev, "Warning: Using internal RC 16M clock source. Time may be inaccurate!\n");
337
338     do { /* read again in case it changes */
339         hw_day = rtc_reg_read(chip, RTC_DAY_REG);
340         hw_hhmmss = rtc_reg_read(chip, RTC_HHMMSS_REG);
341     } while ((hw_day != rtc_reg_read(chip, RTC_DAY_REG)) ||
342            (hw_hhmmss != rtc_reg_read(chip, RTC_HHMMSS_REG)));
343     err = hw_day_hhmmss_to_tm(chip->data, hw_day, hw_hhmmss, tm);
344     if (err)
345         return err;
346
347     err = rtc_valid_tm(tm);
348     if (err) {
349         dev_err(dev, "sunxi_rtc_gettime(): Invalid rtc_time: %pR\n", tm);
350     }
351 }
:set number 335, 29-36 28%
```

升级 libcurl 版本

直接修改 tina-sdk/package/network/utils/curl/Makefile，将版本号改为最新版本即可，版本号可参考 PKG_SOURCE_URL 中的版本。

```
# Copyright (C) 2007-2015 OpenWrt.org
#
# This is free software, licensed under the GNU General Public License v2.
# See /LICENSE for more information.
#
include $(TOPDIR)/rules.mk

PKG_NAME:=curl
# 7.54.1
PKG_VERSION:=7.88.1
PKG_RELEASE:=1

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.bz2
PKG_SOURCE_URL:=http://curl.haxx.se/download/ \
http://www.mirrorspace.org/curl/ \
ftp://ftp.sunet.se/pub/www/utilities/curl/ \
ftp://ftp.planetmirror.com/pub/curl/ \
http://www.mirrormonster.com/download/ \
http://curl.mirrors.cyberservers.net/download/
#PKG_MD5SUM:=6b6eb722f512e7a24855ff084f54fe55
PKG_MD5SUM:=4418e0d94f29d352afafdab445e37a3d

PKG_LICENSE:=MIT
PKG_LICENSE_FILES:=COPYING

PKG_FIXUP:=autoreconf
```

参考代码如下：

```
PKG_NAME:=curl
# 7.54.1
PKG_VERSION:=7.88.1
```

```

PKG_RELEASE:=1

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.bz2
PKG_SOURCE_URL:=http://curl.haxx.se/download/ \
    http://www.mirrorspace.org/curl/ \
    ftp://ftp.sunet.se/pub/www/utilities/curl/ \
    ftp://ftp.planetmirror.com/pub/curl/ \
    http://www.mirrormonster.com/curl/download/ \
    http://curl.mirrors.cyberservers.net/download/
#PKG_MD5SUM:=6b6eb722f512e7a24855ff084f54fe55
PKG_MD5SUM:=4418e0d94f29d352afafadab445e37a3d

```

同时，CONFIGURE_ARGS 需要删去 --without-libmetalink 以兼容新版，参考下图：

```

source "${SOURCE}/Config.in"
endif

TARGET_CFLAGS += $(FPIC)

CONFIGURE_ARGS += \
    --disable-debug \
    --disable-ares \
    --enable-shared \
    --enable-static \
    --disable-manual \
    --without-ca-bundle \
    --without-nss \
    --enable-curldebug \
    --without-librtmp \
    $(call autoconf_bool, CONFIG_IPV6, ipv6) \
    $(if $(CONFIG_LIBCURL_AXTLS), --with-axtls="$(STAGING_DI
    $(if $(CONFIG_LIBCURL_CYASSL), --with-cyassl="$(STAGING_
    $(if $(CONFIG_LIBCURL_GNUTLS), --with-gnutls="$(STAGING_
    $(if $(CONFIG_LIBCURL_OPENSSL), --with-ssl="$(STAGING_DI
    $(if $(CONFIG_LIBCURL_POLARSSL), --with-polarssl="$(STAG
arssl) \
    $(if $(CONFIG_LIBCURL_MBEDTLS), --with-mbedtls="$(STAGIN

```

升级 freetype 版本

同样的，freetype 也可以升级，修改 tina-sdk/package/libs/freetype/Makefile，直接修改版本号即可，参考代码如下：

```

PKG_NAME:=freetype
# 2.9.1

```

```
PKG_VERSION:=2.13.2
PKG_RELEASE:=2

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.gz
PKG_SOURCE_URL:=@SF/freetype
PKG_MD5SUM:=d99447cf00c5f2679918f66f2b5771f6
#PKG_MD5SUM:=3adb0e35d3c100c456357345ccfa8056
```

升级 htop 版本

修改 tina-sdk/package/utils/htop/Makefile, 参考代码如下, 修改方法如上:

```
PKG_NAME:=htop
# 2.0.2
PKG_VERSION:=2.2.0
PKG_RELEASE:=1

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.gz
PKG_SOURCE_URL:=http://hisham.hm/htop/releases/$(PKG_VERSION)/
PKG_HASH:=179be9dccb80cee0c5e1a1f58c8f72ce7b2328ede30fb71dcdcf3365
39be2f487
```

注意更新后需要删除 package/utils/htop/patches/ 100-fix-libtool-version-check.patch 补丁

解决 tplayer 1080P 解码问题

<https://bbs.aw-ol.com/topic/5087/t113-s3%E8%A7%A3%E7%A0%811080p%E8%A7%86%E9%A2%91%E8%8A%B1%E5%B1%8F-set-scaledown%E8%A7%86%E9%A2%91%E7%94%BB%E9%9D%A2%E4%B8%8D%E5%8A%A8>

package/allwinner/tina_multimedia/tplayer/configs/t113_linux_cedarx.conf

检查是否使用 video lbc mode, is_lossy and rc_en

如果配置了设置 0 0 0

```
# video lbc mode, is_lossy and rc_en
vd_lbc_mode = 2
vd_lbc_is_lossy = 1
vd_lbc_rc_en = 0
```

升级 LVGL 版本

<https://bbs.aw-ol.com/topic/4493/%E6%8A%8A-d1-h-t113-%E7%9A%84-lvgl->

[8-1-0-%E5%8D%87%E7%BA%A7%E5%88%B0-8-3-](#)

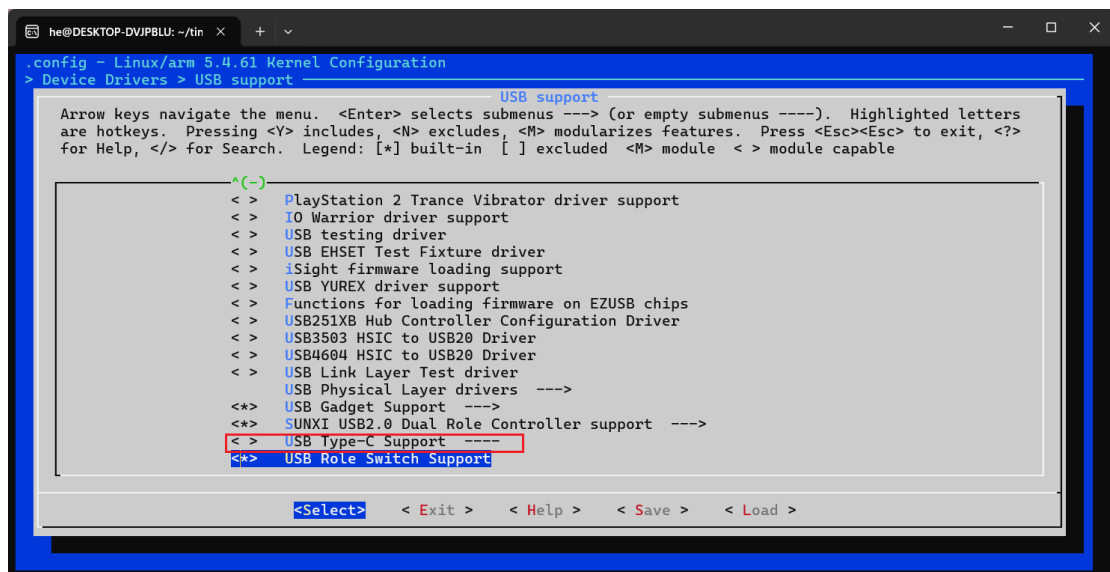
[2-%E5%B9%B6%E9%80%82%E9%85%8D-g2d-%E5%8A%A0%E9%80%9F](#)

RTL8723BS 驱动

<https://github.com/openlumi/rtl8723bs>

解决 ADB Type-C 空指针报错问题

make kernel_menuconfig,关闭 type-c 支持即可



添加 fftw3

文件位置: package/libs/fftw3

<https://gist.github.com/FanhuaCloud/1f83cf55d35042cae7e4e80bd8681000>

添加 libjpeg-turbo

文件位置: package/libs/libjpeg-turbo

<https://gist.github.com/FanhuaCloud/24d479a2adbe4af5dbe86f8326e1117>

9

添加 tmux

文件位置: package/utils/tmux

<https://gist.github.com/FanhuaCloud/179f10ff5161f58e1089535fffa56f34>

解决 SPI Nand 无法启动烧录问题

https://whykan.com/t_10074.html

1.uboot dts 开启 wp 和 hold 的内部上拉，默认的只开启了 CS 的内部上拉

```
&spi0_pins_b {
    allwinner,pins = "PC3", "PC7", "PC6";
    allwinner,pname = "spi0_cs0", "spi0_hold", "spi0_wp";
    allwinner,function = "spi0";
    allwinner,muxsel = <2>;
    allwinner,drive = <1>;
    allwinner,pull = <1>; /* cs, hold, wp should be pulled up */
};
```

2.linux dts 对应的 drive 需要按照默认的 10，是别的值可能会开机卡死在 linux

Tina 无法重启

<https://bbs.aw-ol.com/topic/1928/t113-tina-reboot-%E6%97%A0%E6%B3%95%E9%87%8D%E5%90%AF/15>

应该是 wifi 驱动中存在死锁，可以试试下面的补丁：
相应的修改模组驱动即可。

```
diff --git a/drivers/net/wireless/rtl8821cs/os_dep/osdep_service.c
b/drivers/net/wireless/rtl8821cs/os_dep/osdep_service.c
index 39b5345b9885..bb037d64add2 100755
--- a/drivers/net/wireless/rtl8821cs/os_dep/osdep_service.c
+++ b/drivers/net/wireless/rtl8821cs/os_dep/osdep_service.c
@@ -1279,12 +1279,15 @@ u32 _rtw_down_sema(_sema *sema)
 {

#ifdef PLATFORM_LINUX
-
+ #if 0
    if (down_interruptible(sema))
        return _FAIL;
    else
```



```
        return _SUCCESS;
-
+ #else
+     down(sema);
+     return _SUCCESS;
+ #endif
#endif
#ifdef PLATFORM_FREEBSD
    sema_wait(sema);
```

RGB 屏幕缺几行

修改 RGB 时序，加大 vbp 参数

U-Boot 背光没法用

<https://bbs.aw-ol.com/topic/949/%E8%AF%B7%E9%97%AE-uboot-%E4%B8%8D%E8%83%BD%E9%A9%B1%E5%8A%A8pwm7-pd22-%E6%98%AF%E4%BB%80%E4%B9%88%E5%8E%9F%E5%9B%A0%E5%91%A2/5>

解决方法：使用屏幕驱动里面的 IO 控制，直接点亮背光

开机 LOGO

tina-sdk/lichee/brandy-2.0/u-boot-2018/configs/sun8iw20p1_uart3_defconfig

增加 CONFIG_BOOT_GUI=y

修改 LOGO 图片：target/allwinner/generic/boot-resource/boot-resource

解决 ADB 烧录问题

根据错误找到 RTC 问题，按照提示修改 RTC 配置即可

make kernel_menuconfig

Device Drivers->RTC->Sunxi rtc reboot flag 打开即可

```

|| _ || | | | | | | | | | | | |
|| | | | | | | | | | | | | | |
|| | | | | | | | | | | | | | |
-----
Tina is Based on OpenWrt!
-----

```

```
Tina Linux (Neptune, 61CC0487)
-----
```

```
root@TinaLinux:/# reboot efex
write misc: efex
write rtc: efex
E:can not access /sys/devices/platform/soc/rtc/flag
E:can not access /sys/devices/platform/soc@3000000/7090000.rtc/flag
root@TinaLinux:/# [ 16.623658] printk: udevd: 1 output lines suppressed due to ratelimiting
█
```