



软件应用指南  
ECB20-6Y28A5ME8G-I 单板机



## 目录

免责声明和版权公告 .....	1
1. 概述 .....	2
1.1. 软件资源 .....	2
2. 使用前准备 .....	2
2.1. 串口软件安装 .....	2
2.2. 设置拨码开关 .....	3
3. 快速开始 .....	3
3.1. 镜像刻录 .....	3
3.2. 上电开机 .....	4
4. 快速体验 .....	5
4.1. CPU .....	5
4.2. 内存 .....	8
4.3. eMMC 测试 .....	12
4.4. 看门狗 .....	13
4.5. 电源管理 .....	15
4.6. RTC .....	15
4.7. ETH .....	16
4.8. 显示 .....	17
4.9. 触摸 .....	19
4.10. TF 卡 .....	20
4.11. 4G 模块 .....	22
4.12. USB .....	27
4.13. WIFI 模块 .....	28
4.14. 音频 .....	33
4.15. LED .....	36
4.16. 按键 .....	37
5. 参考资料 .....	39
6. 修订说明 .....	39
7. 关于我们 .....	39

## 免责声明和版权公告

本文中的信息，如有变更，恕不另行通知。文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

文中所得测试数据均为亿佰特实验室测试所得，实际结果可能略有差异。

文中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

最终解释权归成都亿佰特电子科技有限公司所有。

### 注 意：

由于产品版本升级或其他原因，本手册内容有可能变更。亿佰特电子科技有限公司保留在没有任何通知或者提示的情况下对本手册的内容进行修改的权利。本手册仅作为使用指导，成都亿佰特电子科技有限公司尽全力在本手册中提供准确的信息，但是成都亿佰特电子科技有限公司并不确保手册内容完全没有错误，本手册中的所有陈述、信息和建议也不构成任何明示或暗示的担保。

## 1. 概述

本文着重介绍如何测试并使用亿佰特 ECB20-6Y28A5ME8G-I 单板机提供的功能，若用户还需在此基础上移植适配特定功能，也请参考《ECK20-6Y28A 软件开发指南》，后文以“软件开发指南”指代该文档。本文可作为前期评估指导使用，也可以作为通用系统开发的测试指导书使用。

### 1.1. 软件资源

ECB20-6Y28A5ME8G-I 搭载基于 Linux 5.10.9 版本内核构建的操作系统，开发资料附带嵌入式 Linux 系统开发所需要的交叉编译工具链，U-boot 2020.04 源代码，Linux 内核与各驱动模块的源代码，以及适用于 Windows 桌面环境和 Linux 桌面环境的各种开发调试工具，应用开发样例等。

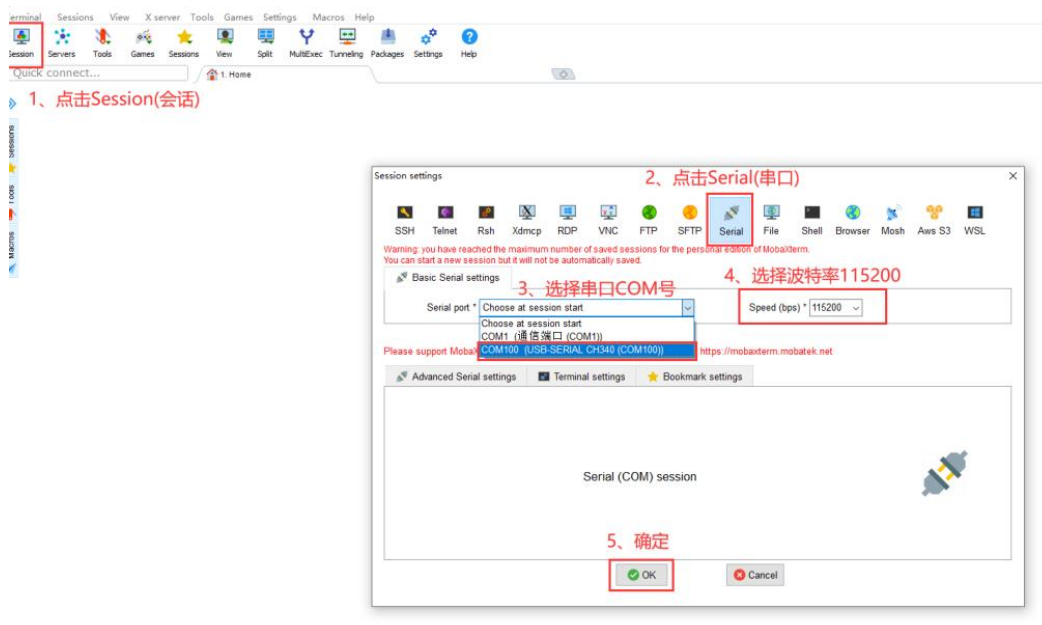
若用户想要从头编译 SDK，请先查看文档《buildroot\_custom\_kernel\_and\_uboot.pdf》拷贝内核与 uboot 到 buildroot 源码目录。

## 2. 使用前准备

### 2.1. 串口软件安装

使用串口前主要需要安装 CH340 串口驱动和 MobaXterm 串口终端。具体安装与使用方法可见软件开发指南。

连接好串口之后，安装 MobaXterm 后进行配置，按如下方式打开串口。



## 2.2. 设置拨码开关

单板机底板支持四种启动模式，分别是 USB（Download），TF/SD 卡，eMMC 与 NAND 启动模式。

SW								BOOT DEVICE
D1	D2	D3	D4	D5	D6	D7	D8	
OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF	USB
ON	OFF	OFF	OFF	OFF	OFF	ON	OFF	MicroSD
ON	OFF	ON	OFF	OFF	ON	ON	OFF	EMMC
ON	OFF	OFF	OFF	ON	OFF	OFF	ON	NAND

## 3. 快速开始

### 3.1. 镜像刻录

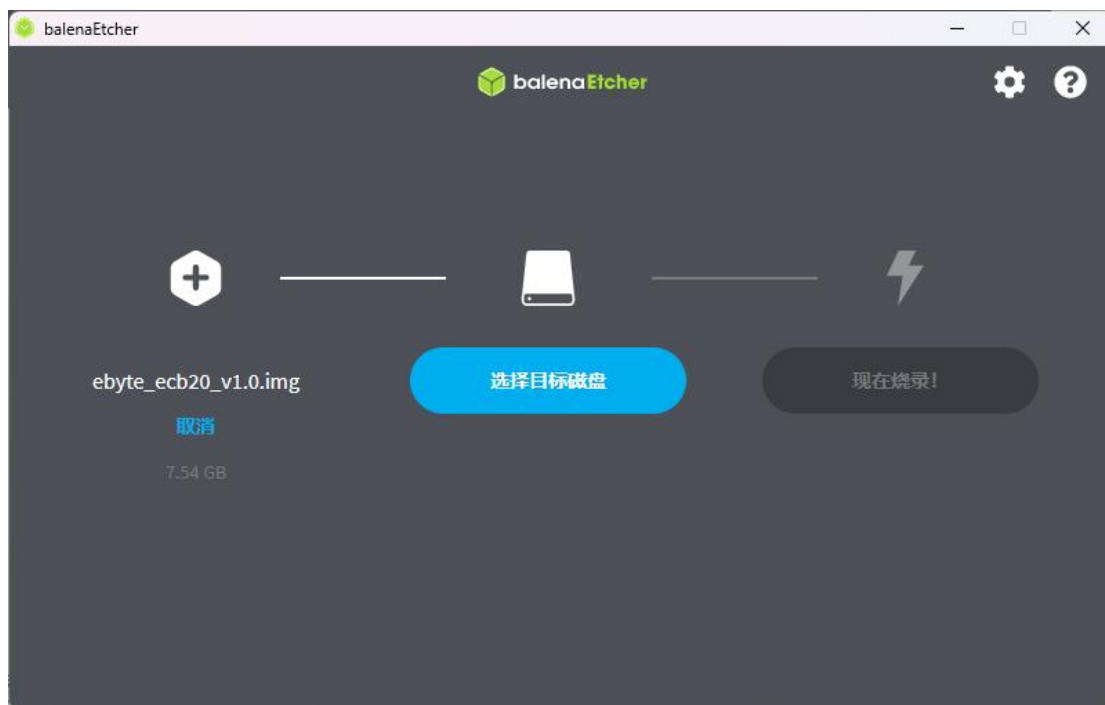
出厂开发板没有烧写镜像，请参考第二章内容配置拨码为从 SD 卡启动并制作相应 SD 启动卡，详细文档参考《Software\_Development\_Guide》。

下面简单描述如何制作 SD 卡：

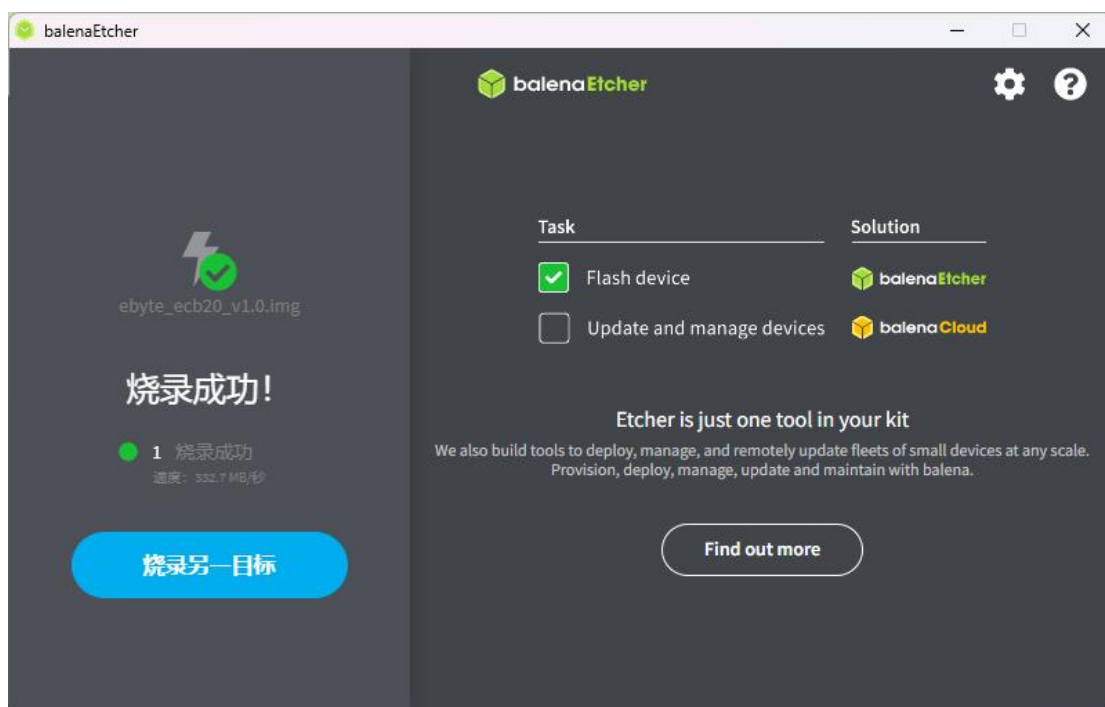
使用第三方的镜像烧录工具将位于 02\_Images/ 中的 img 镜像文件 ebyte\_ecb20.img 写入 SD 卡，在 04\_Tools/ 中已经包含了一个烧写工具：balenaEtcher-Portable-1.18.11.exe。



选择从文件烧录，文件选择之前制作的 wic 文件。



选择目标磁盘，然后点击现在烧录，开始烧录并等待烧录完成后移除设备即可使用 SD 卡作为启动设备。



SD 卡槽位于单板机背面，标注为 TF card。

### 3.2.上电开机

为了便于用户操作调试，本司 ECB20 系列单板机预留了调试串口，用户可在单板机右下角找到串口排针。

单板机启动后，串口终端打印 U-Boot 和内核的运行信息，如下所示。

```
U-Boot 2020.04-g93edef74 (Nov 07 2024 - 13:17:39 +0800)

CPU:   i.MX6ULL rev1.1 792 MHz (running at 396 MHz)
CPU:   Industrial temperature grade (-40C to 105C) at 38C
Reset cause: POR
Model: EBYTE i.MX6 ULL EVK Board
Board: ebyte I.MX6ULL
DRAM:  512 MiB
MMC:   FSL_SDHC: 0, FSL_SDHC: 1
Loading Environment from MMC... OK
[*]-Video Link 0 (1024 x 600)
      [0] lcdif@21c8000, video
In:    serial
Out:   serial
Err:   serial
switch to partitions #0, OK
mmc1(part 0) is current device
flash target is MMC:1
Net:
Warning: ethernet@20b4000 (eth1) using random MAC address - 8e:73:ad:0f:da:a0
eth1: ethernet@20b4000 [PRIME]
Fastboot: Normal
Normal Boot
Hit any key to stop autoboot: 0
WARNING: Could not determine tee to use
switch to partitions #0, OK
mmc1(part 0) is current device
switch to partitions #0, OK
mmc1(part 0) is current device
8467992 bytes read in 275 ms (29.4 MiB/s)
Booting from mmc ...
33591 bytes read in 18 ms (1.8 MiB/s)
Kernel image @ 0x80800000 [ 0x000000 - 0x813618 ]
## Flattened Device Tree blob at 83000000
   Booting using the fdt blob at 0x83000000
   Using Device Tree in place at 83000000, end 8300b336
Modify /soc/bus@22000000/epdc@228c0000:status disabled
ft_system_setup for mx6

Starting kernel ...

[    0.000000] Booting Linux on physical CPU 0x0
```

当温度超过限制范围，为保证自身安全，u-boot 将不会引导启动内核，请用户注意使用环境。

## 4. 快速体验

本章节将按照功能模块进行测试，提供一种初步的功能运用方式。

### 4.1.CPU

#### 4.1.1. 查看 CPU 信息

使用如下命令查看 CPU 信息。

```
[root@EBYTE-IMX6UL ~]# cat /sys/devices/soc0/soc_id

i.MX6ULL

[root@EBYTE-IMX6UL ~]# cat /proc/cpuinfo
```

```
processor      : 0

model name    : ARMv7 Processor rev 5 (v7l)

BogoMIPS     : 96.00

Features     : half thumb fastmult vfp edsp neon vfpv3 tls vfpv4 idiva idivt vfpd
32 lpae

CPU implementer : 0x41

CPU architecture: 7

CPU variant   : 0x0

CPU part      : 0xc07

CPU revision  : 5

Hardware     : Freescale i.MX6 Ultralite (Device Tree)

Revision     : 0000

Serial      : 162a49d768e2b9fc
```

soc\_id: 表明这是一颗 i.MX6ULL 芯片

processor: 系统中逻辑处理核的编号, 对于多核处理器则可以是物理核、或者使用超线程技术虚拟的逻辑核

model name: CPU 属于的名字及其编号

BogoMIPS: 在系统内核启动时粗略测算的 CPU 每秒运行百万条指令数 (MillionInstructions Per Second)

Features: CPU 支持的指令功能

### 4.1.2. 查看 CPU 使用率

使用如下命令查看 CPU 使用率

```
[root@EBYTE-IMX6UL ~]# top

top - 10:48:34 up 2 min,  1 user,  load average: 0.52, 0.45, 0.18

Tasks:  77 total,   1 running,  76 sleeping,   0 stopped,   0 zombie

%Cpu0  :  0.7/3.3   4[          ]

GiB Mem : 28.0/0.5  [          ]
```



GiB Swap: 0.0/0.0 [ ]										
PID	USER	PR	NI	VIRT	RES	%CPU	%MEM	TIME+	S	COM
1	root	20	0	3.2m	1.6m	0.0	0.3	0:05.81	S	init
115	root	20	0	3.2m	0.4m	0.0	0.1	0:00.06	S	`- /sbin/syslog+
119	root	20	0	3.2m	0.4m	0.0	0.1	0:00.05	S	`- /sbin/klogd +
130	root	20	0	14.1m	1.8m	0.0	0.4	0:01.10	S	`- /sbin/udev +
163	dbus	20	0	3.1m	0.1m	0.0	0.0	0:00.00	S	`- dbus-daemon
167	root	20	0	2.1m	1.2m	0.0	0.3	0:00.04	S	`- /usr/sbin/rp+
180	dhcpcd	20	0	2.3m	1.7m	0.0	0.3	0:00.03	S	`- dhcpcd: [man
181	root	20	0	2.6m	1.9m	0.0	0.4	0:00.08	S	`- dhcpcd:
198	dhcpcd	20	0	2.6m	0.3m	0.0	0.1	0:00.00	S	`- dhc
206	dhcpcd	20	0	2.6m	0.3m	0.0	0.1	0:00.00	S	`- dhc
182	dhcpcd	20	0	2.2m	0.2m	0.0	0.0	0:00.00	S	`- dhcpcd:
183	dhcpcd	20	0	2.2m	0.2m	0.0	0.0	0:00.00	S	`- dhcpcd:
226	root	20	0	7.4m	2.5m	0.0	0.5	0:00.11	S	`- /usr/sbin/nt+
231	root	20	0	2.5m	0.1m	0.0	0.0	0:00.00	S	`- /usr/sbin/dr+
238	root	20	0	2.3m	1.4m	0.0	0.3	0:00.00	S	`- rpc.statd
251	root	20	0	169.5m	55.7m	1.3	11.4	0:10.30	S	`- /usr/local/D+
254	root	20	0	2.7m	0.2m	0.0	0.0	0:00.00	S	`- /usr/sbin/co+

Tasks: 表示当前系统总共有 77 个进程, 有一个正在运行中, 76 个处于睡眠状态, 没有停止的进程也没有僵尸进程。

%Cpu\*: 表示 CPU\*的工作状态, 由于 IMX6ULL 为单核心 CPU, 此处只有一行 Cpu0。前面的 0.7 表示当前正在被进程使用的百分比, 而 3.3 是进程空闲时所占的百分比。

GiB Mem 和 GiB Swap: 表示系统的内存占用与总大小, 单位为 G。

关于进程列表, 每一列的信息如下表:

列名	解释
PID	进程 ID。每个进程在系统中都有一个唯一的 PID。
USER	进程所属的用户。
PR	进程的优先级 (Priority)。较低的数值意味着更高的优先级。
NI	进程的 nice 值, 用于调整进程的优先级。
VIRT	进程使用的虚拟内存量。
RES	进程占用的实际物理内存 (常驻内存)。
%CPU	进程占用的 CPU 百分比。
%MEM	进程占用的物理内存的百分比。
TIME+	进程运行的总时间。
S	进程状态: 例如, S 表示 "睡眠" 状态, R 表示 "运行" 状态, Z 表示 "僵尸" 状态。
COMMAND	进程的启动命令或可执行程序名。

### 4.1.3. 查看 CPU 温度

使用如下命令查看 CPU 当前温度:

```
[root@EBYTE-IMX6UL ~ ]# cat /sys/class/thermal/thermal_zone0/temp
54725
```

对于返回值, 我们除以 1000 就是实际温度, 此处返回 54725, 则实际温度为 54.7°C。

## 4.2. 内存

### 4.2.1. 内存信息

使用如下命令查看内存信息:

```
[root@EBYTE-IMX6UL ~ ]# cat /proc/meminfo
MemTotal:          498920 kB
MemFree:           320260 kB
MemAvailable:     361232 kB
Buffers:           3764 kB
```

Cached:	39400 kB
SwapCached:	0 kB
Active:	11196 kB
Inactive:	71720 kB
Active(anon):	132 kB
Inactive(anon):	39816 kB
Active(file):	11064 kB
Inactive(file):	31904 kB
Unevictable:	0 kB
Mlocked:	0 kB
HighTotal:	0 kB
HighFree:	0 kB
LowTotal:	498920 kB
LowFree:	320260 kB
SwapTotal:	0 kB
SwapFree:	0 kB
Dirty:	0 kB
Writeback:	0 kB
AnonPages:	39804 kB
Mapped:	25148 kB
Shmem:	200 kB
KReclaimable:	9792 kB
Slab:	21164 kB
SReclaimable:	9792 kB
SUnreclaim:	11372 kB
KernelStack:	592 kB
PageTables:	576 kB
NFS_Unstable:	0 kB

Bounce:	0 kB
WritebackTmp:	0 kB
CommitLimit:	249460 kB
Committed_AS:	70092 kB
VmallocTotal:	1556480 kB
VmallocUsed:	2088 kB
VmallocChunk:	0 kB
Percpu:	104 kB
CmaTotal:	131072 kB
CmaFree:	64924 kB

**MemTotal:** 总内存量。这里显示的值为 498788 kB，表示系统总共有大约 487 MB 的物理内存可用。由于此处换算方式为 1024，而常规销售过程中一般使用 1000 换算，故此处大小也可看做 510.76MB，在加上部分内存留作它用，此处大小也算正常。

**MemFree:** 空闲内存量。这里显示的值为 369352 kB，表示当前系统中有大约 360 MB 的内存是空闲的，未被使用。

**MemAvailable:** 可用内存量。这里显示的值为 396952 kB，表示当前可供系统使用的内存总量，包括已缓存的内存和可用的内存。

**Buffers:** 缓冲区使用量。这里显示的值为 0 kB，表示系统当前没有使用任何内存作为缓冲区。

**Cached:** 缓存的内存量。这里显示的值为 39208 kB，表示系统当前用于缓存的内存量。

**SwapCached:** 交换缓存的内存量。这里显示的值为 0 kB，表示当前没有被缓存到交换空间中的内存。

**Active:** 活跃的内存量。这里显示的值为 8576 kB，表示当前正在使用的内存量。

#### 4.2.2. 内存测试

使用如下命令获取可用于测试的内存大小：

```
[root@EBYTE-IMX6UL ~]# free -m
```

	total	used	free	shared	buff/cache	available
Mem:	487	87	379	0	20	

388

Swap:                    0                    0                    0

此处可用于测试的内存大小最大为 388MB，但是考虑到后续其他程序的稳定运行，下面以 350MB 进行一次内存测试为例：

```
[root@EBYTE-IMX6UL ~ ]# memtester 350M 1

memtester version 4.5.0 (32-bit)

Copyright (C) 2001-2020 Charles Cazabon.

Licensed under the GNU General Public License version 2 (only).

pagesize is 4096
pagesizemask is 0xfffff000
want 350MB (367001600 bytes)
got 350MB (367001600 bytes), trying mlock ...locked.

Loop 1/1:

Stuck Address            : ok

Random Value             : ok

Compare XOR              : ok

Compare SUB              : ok

Compare MUL              : ok

Compare DIV              : ok

Compare OR               : ok

Compare AND              : ok

Sequential Increment: ok

Solid Bits                : ok

Block Sequential         : ok

Checkerboard             : ok

Bit Spread               : ok

Bit Flip                 : ok
```

```

Walking Ones      : ok

Walking Zeroes   : ok

8-bit Writes     : ok

16-bit Writes    : ok

Done.
    
```

可以看到此处全部测试项都是`ok`状态，表示测试通过，内存无明显错误。

## 4.3.eMMC 测试

### 4.3.1. 查看 eMMC 信息

使用如下命令获取 eMMC 设备容量大小与分区情况表：

```

[root@EBYTE-IMX6UL ~]# fdisk -l /dev/mmcblk1

Disk /dev/mmcblk1: 7.28 GiB, 7818182656 bytes, 15269888 sectors

Units: sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: dos

Disk identifier: 0x25951234

Device            Boot Start      End  Sectors  Size Id Type
/dev/mmcblk1p1 *    10240     81919    71680   35M ef EFI (FAT-12/16/32)
/dev/mmcblk1p2      81920 15269887 15187968  7.2G 83 Linux
    
```

此处 eMMC 分为了两个分区，p1 分区用于存放内核和设备树，为 EFI 分区。P2 分区用于存放 Linux 文件系统。可通过下面命令获取分区类型：

```

lsblk -f

NAME            FSTYPE FSVER LABEL UUID
FSAVAIL FSUSE% MOUNTPOINTS

mmcblk1p1      vfat   FAT32 boot  FCFE-9291                25.6
    
```

```
M    26% /boot

`-mmcblk1

mmcblk1p2    ext4    1.0    root    e39f8131-958c-466f-88c4-e5b1342aaba2    6.4G
5% /

`-mmcblk1

mmcblk1boot0

mmcblk1boot1
```

### 4.3.2. eMMC 读写

可以使用如下命令测试 eMMC 读写性能:

```
[root@EBYTE-IMX6UL ~ ]# dd if=/dev/zero of=test.bin bs=1M count=100 conv=fsync
100+0 records in
100+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 2.91373 s, 36.0 MB/s
[root@EBYTE-IMX6UL ~ ]# echo 3 > /proc/sys/vm/drop_caches
[root@EBYTE-IMX6UL ~ ]# dd if=test.bin of=/dev/null bs=1M
100+0 records in
100+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 1.80595 s, 58.1 MB/s
```

根据上面的内容, 我们可以得知, 当前单板机的写入速度为 36.0MB/s, 读取速度为 58.1MB/s。

同时上面的`echo 3 > /proc/sys/vm/drop\_caches`是为了消除系统缓存的影响, 获取实际的读取速度。

## 4.4.看门狗

使用如下命令查看片内看门狗配置信息:

```
[root@EBYTE-IMX6UL ~ ]# wdtctl
```

```

Device:          /dev/watchdog0

Identity:       imx2+ watchdog [version 0]

Timeout:       60 seconds

Pre-timeout:   0 seconds

FLAG           DESCRIPTION           STATUS BOOT-STATUS
KEEPALIVEPING  Keep alive ping reply       1       0
MAGICCLOSE     Supports magic close char   0       0
PRETIMEOUT     Pretimeout (in seconds)     0       0
SETTIMEOUT     Set timeout (in seconds)    0       0
  
```

使用如下命令配置看门狗超时时间为 5 秒并操作让单板机超时重启:

```

[root@EBYTE-IMX6UL ~ ]# wdctl -s 5

Timeout has been set to 5 seconds.

Device:          /dev/watchdog0

Identity:       imx2+ watchdog [version 0]

Timeout:       5 seconds

Pre-timeout:   0 seconds

FLAG           DESCRIPTION           STATUS BOOT-STATUS
KEEPALIVEPING  Keep alive ping reply       1       0
MAGICCLOSE     Supports magic close char   0       0
PRETIMEOUT     Pretimeout (in seconds)     0       0
SETTIMEOUT     Set timeout (in seconds)    0       0

[root@EBYTE-IMX6UL ~ ]# cat > /dev/watchdog

U-Boot 2020.04-g93edef74 (Nov 15 2024 - 10:23:36 +0800)

CPU:   i.MX6ULL rev1.1 792 MHz (running at 396 MHz)
CPU:   Industrial temperature grade (-40C to 105C) at 51C
  
```



```
Reset cause: POR
```

```
Model: EBYTE i.MX6 ULL EVK Board
```

在该流程中，会在比 5 秒多一点的时间后系统自动重启。

## 4.5.电源管理

本章节演示 Linux 电源管理的 Suspend 功能，让开发板睡眠，通过外部事件唤醒。

Linux 内核提供了多种 Suspend 方式，在 ECB20 系列单板机上，支持如下：

```
[root@EBYTE-IMX6UL ~]# cat /sys/power/state
```

```
freeze standby mem
```

在用户空间向"/sys/power/state"文件分别写入"freeze"、"standby"和"mem"，即可进入对于休眠模式。在当前配置当中，休眠后用户想要唤醒可以通过插入或拔出一张 SD 卡实现。

```
[root@EBYTE-IMX6UL ~]# echo freeze > /sys/power/state
```

```
[root@EBYTE-IMX6UL ~]# echo standby > /sys/power/state
```

```
[root@EBYTE-IMX6UL ~]# echo mem > /sys/power/state
```

## 4.6.RTC

Linux 系统中分系统时钟（软件时钟）和 RTC 时钟（硬件时钟），系统时钟掉电即会消失，RTC 时钟在安装电池的情况下会长期运行。如需使用外部 RTC 时钟，请将 ML2032（3V 可充）或 CR2032（3V 不可充）电池安装至 RTC 纽扣电池座，单板机上标识为'BAT'。

1. 设置当前时间：

```
[root@EBYTE-IMX6UL ~]# date -s "2010-10-10 10:10:10"
```

```
Sun Oct 10 10:10:10 CST 2010
```

2. 写入系统时间到 RTC，并查询是否写入：

```
[root@EBYTE-IMX6UL ~]# hwclock -w
```

```
[root@EBYTE-IMX6UL ~]# hwclock
```

```
Sun Oct 10 10:10:15 2010 0.000000 seconds
```

3. 关机拔出供电并等待一定时间后重新上电查看 RTC 时间：

```
[root@EBYTE-IMX6UL ~]# hwclock
```

Sun Oct 10 10:11:24 2010 0.000000 seconds

## 4.7.ETH

当前单板机存在两个网口，靠近 HDMI 的网口可用于在 uboot 阶段连接网络。

使用 net-tools 工具包中的 ifconfig 对网络进行手动配置，首先通过通过 ifconfig 命令查看网络设备信息如下：

```
[root@EBYTE-IMX6UL ~ ]# ifconfig

eth0      Link encap:Ethernet  HWaddr 4E:42:0B:5A:EE:7F
          inet addr:192.168.1.135  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::b5d8:8dc:ad0f:c06c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:66 errors:0 dropped:0 overruns:0 frame:0
          TX packets:63 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8089 (7.8 KiB)  TX bytes:5782 (5.6 KiB)

eth1      Link encap:Ethernet  HWaddr 8E:78:EB:6D:85:D7
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:134 errors:0 dropped:0 overruns:0 frame:0
          TX packets:134 errors:0 dropped:0 overruns:0 carrier:0
```

```
collisions:0 txqueuelen:1000  
  
RX bytes:9936 (9.7 KiB) TX bytes:9936 (9.7 KiB)
```

当前的根文件系统配置了自动 DHCP 功能, 但用户也可以通过使用`udhcp`手动获取一个 ip 地址, 又或者手动配置一个静态 ip。

#### 4.7.1. 自动获取

```
[root@EBYTE-IMX6UL ~ ]# udhcp -i eth0  
  
udhcp: started, v1.35.0  
  
udhcp: broadcasting discover  
  
udhcp: broadcasting select for 192.168.1.135, server 192.168.1.1  
  
udhcp: lease of 192.168.1.135 obtained from 192.168.1.1, lease time 7200  
  
deleting routers  
  
adding dns 114.114.114.114  
  
adding dns 61.139.2.69
```

对于非使用 eth0 网口的用户, 请务必不要遗漏此处的`-i eth0`参数。

#### 4.7.2. 手动配置

下面介绍给 eth0 手动配置 IP 地址 192.168.1.253 的方法, 命令如下:

```
[root@EBYTE-IMX6UL ~ ]# ifconfig eth0 192.168.0.253 netmask 255.255.255.0 up  
  
[root@EBYTE-IMX6UL ~ ]# ifconfig  
  
eth0      Link encap:Ethernet  HWaddr 4E:42:0B:5A:EE:7F  
  
          inet addr:192.168.0.253  Bcast:192.168.0.255  Mask:255.255.255.0  
  
          UP BROADCAST MULTICAST  MTU:1500  Metric:1  
  
          RX packets:118 errors:0 dropped:0 overruns:0 frame:0  
  
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0  
  
          collisions:0 txqueuelen:1000  
  
          RX bytes:13846 (13.5 KiB) TX bytes:8610 (8.4 KiB)
```

#### 4.8. 显示

ECB20-6Y28A5ME8G-I 支持 HDMI 和 LCD 两种显示方案:

HDMI 显示: 由于 i.MX6ULL 芯片本身没有 HDMI 相关的控制器, 所以 HDMI 是通过显示转换芯片 SII9022A 将 RGB 信号转换为 HDMI 信号进行输出, 支持最大分辨率 1280x720@60fps。

LCD 显示: LCD 使用的是本司的 ECA10-7LCD24R1006CT-C 触摸屏, 最大支持 RGB888 模式。

### 4.8.1. 切换显示

由于 HDMI 与 LCD 屏幕共用同样的引脚信号时钟, 且支持的分辨率各不相同, 所以无法同时使用, 用户在使用之前需要修改设备树文件, 切换使用 HDMI 或 LCD 屏。

用户可以通过设备设备树中的 `#define USE\_HDMI\_PHY 0` 值为 1 或 0 选择使用 HDMI 还是 LCD 屏, 下面展示的是使用 LCD 是的配置:

```
$ head arch/arm/boot/dts/ebyte-imx6ull-base.dtsi
// SPDX-License-Identifier: (GPL-2.0 OR MIT)
//
// Copyright (C) 2016 Freescale Semiconductor, Inc.

/dts-v1/;

#include "imx6ull.dtsi"
#include <dt-bindings/phy/jlsemi-dt-phy.h>

#define USE_HDMI_PHY 0
```

用户编译完毕 HDMI 支持的对应设备树文件后, 替换原来的 `/boot/ebyte-imx6ull-emmc.dtb` 文件并重启系统即可。

### 4.8.2. 分辨率查看

用户可以通过如下命令查看当前支持的分辨率与时序:

```
[root@EBYTE-IMX6UL ~]# fbset

mode "1024x600-60"
```

```
# D: 51.201 MHz, H: 38.096 kHz, V: 59.993 Hz  
  
geometry 1024 600 1024 600 32  
  
timings 19531 136 160 21 12 24 2  
  
accel false  
  
rgba 8/16,8/8,8/0,0/0  
  
endmode
```

此处列举的是对应 LCD 屏幕的参数。

### 4.8.3. 界面显示

目前 SDK 中可以支持使用 Qt 显示界面，其后端使用 linux-framebuffer，具体使用请参考文档《imx6ull\_buildroot\_QT\_Development.pdf》。

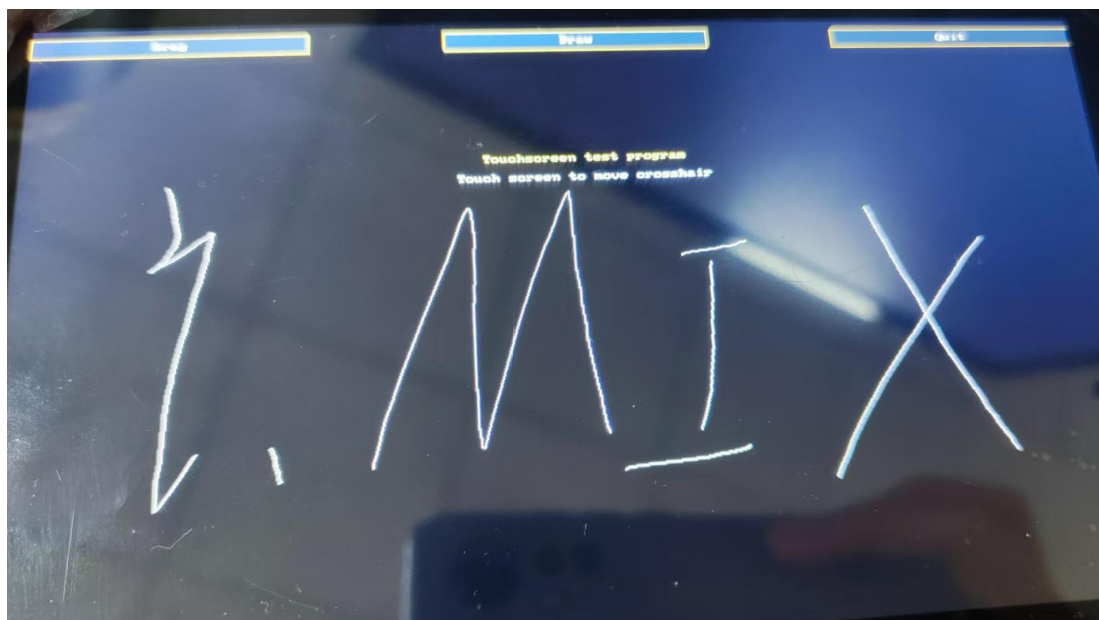
## 4.9. 触摸

当前单板机可与本司 ECA10-7LCD24R1006CT-C 触摸屏协同使用。用户可通过如下命令测试触摸屏功能：

```
[root@EBYTE-IMX6UL ~ ]# ts_test  
  
1731667110.548002: 537 18 255  
  
1731667110.590616: 537 18 0  
  
1731667111.345283: 134 171 255  
  
1731667111.461220: 134 171 255  
  
1731667111.472784: 135 173 255  
  
1731667111.484220: 137 177 255  
  
1731667111.495950: 139 184 255  
  
1731667111.506942: 141 193 255  
  
1731667111.514330: 142 200 255  
  
1731667111.525705: 143 206 255  
  
1731667111.537399: 144 210 255  
  
1731667111.548039: 144 214 255  
  
1731667111.558878: 145 215 255
```

```
1731667111.569677: 145 217 255
1731667111.581670: 145 219 255
1731667111.589180: 145 220 255
```

当在触摸屏上点击会实时从调试串口打印当前坐标位置。



## 4.10. TF 卡

SD 卡的卡座位于单板机背面，与调试串口处于同一侧，另一侧的卡座为 SIM 卡。

### 4.10.1. 查看 TF 卡信息

通过如下命令可以查看当前 TF 卡的状态信息：

```
[root@EBYTE-IMX6UL ~ ]# mount -t debugfs debugfs /sys/kernel/debug/
[root@EBYTE-IMX6UL ~ ]# cat /sys/kernel/debug/mmc0/ios

clock:          132000000 Hz
vdd:            21 (3.3 ~ 3.4 V)
bus mode:       2 (push-pull)
chip select:    0 (don't care)
power mode:     2 (on)
bus width:      2 (4 bits)
timing spec:    6 (sd uhs SDR104)
signal voltage: 1 (1.80 V)
```

```
driver type: 0 (driver type B)
```

可以发现, 当前 SD 卡工作在 SDR104 模式, 其工作电压为 1.8V, 工作频率为 132MHz, 使用 4 根数据线进行通信。

```
[root@EBYTE-IMX6UL ~ ]# fdisk -l /dev/mmcblk0

Disk /dev/mmcblk0: 29.72 GiB, 31914983424 bytes, 62333952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x303d75c4

Device          Boot Start      End  Sectors  Size Id Type
/dev/mmcblk0p1      2048 62333951 62331904 29.7G 83 Linux
```

可以发现, 当前 SD 卡只分配了一个分区, SD 卡总大小为 29.72 GiB。

#### 4.10.2. 测试 TF 卡速度

通过如下命令测试 SD 卡读写速度:

```
[root@EBYTE-IMX6UL ~ ]# mount /dev/mmcblk0p1 /mnt/

[root@EBYTE-IMX6UL ~ ]# dd if=/dev/zero of=/mnt/test.bin bs=10M count=10 conv=
fsync

10+0 records in
10+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 4.33329 s, 24.2 MB/s

[root@EBYTE-IMX6UL ~ ]# echo 3 > /proc/sys/vm/drop_caches

[root@EBYTE-IMX6UL ~ ]# dd if=/mnt/test.bin of=/dev/zero bs=10M

10+0 records in
10+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 1.94783 s, 53.8 MB/s
```

```
[root@EBYTE-IMX6UL ~ ]#
```

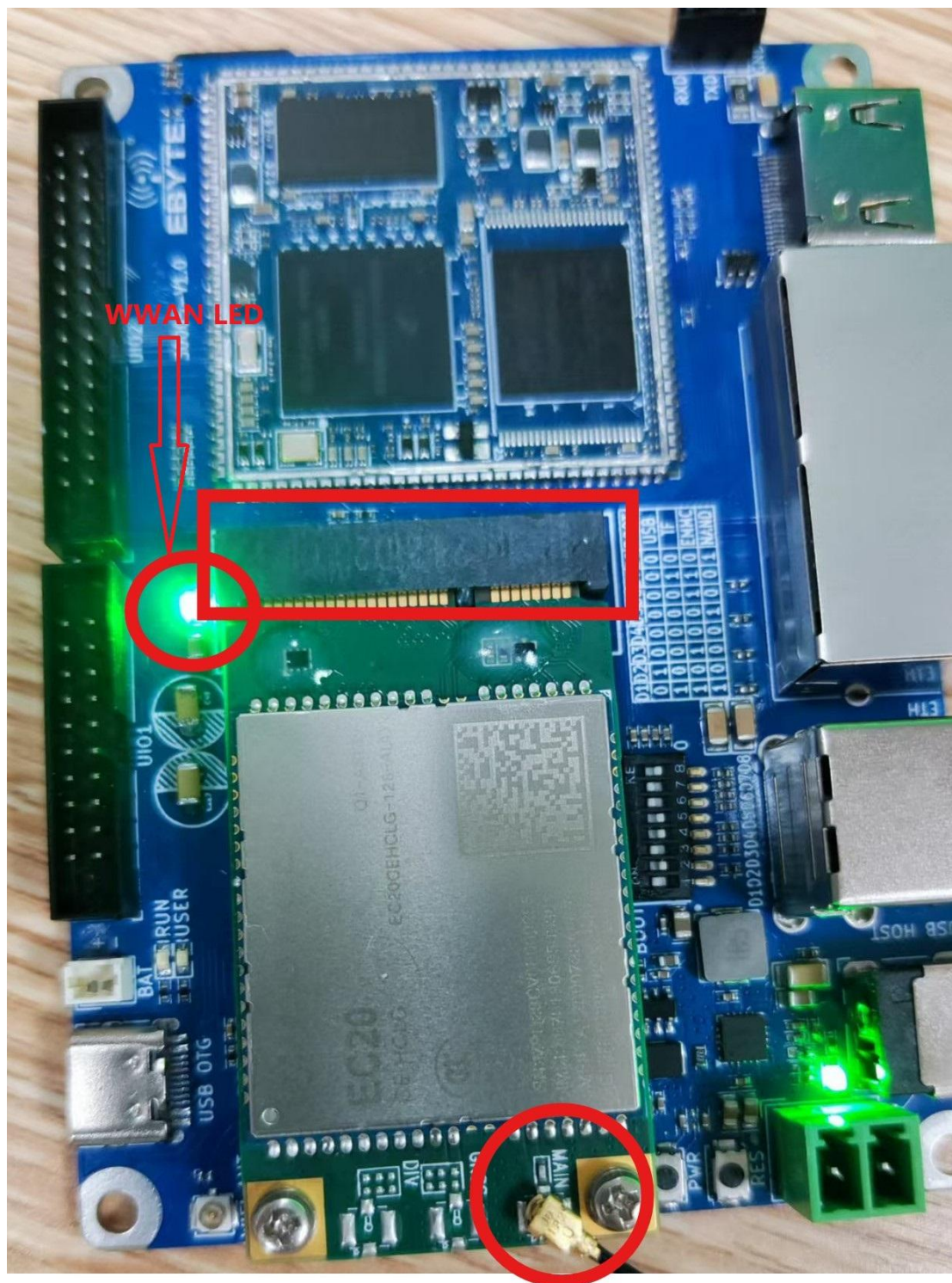
这里读取速度并没有达到理论的 108MB/s，是由于芯片本身的限制，但是只要超过了 50MB/s 都属于正常情况。

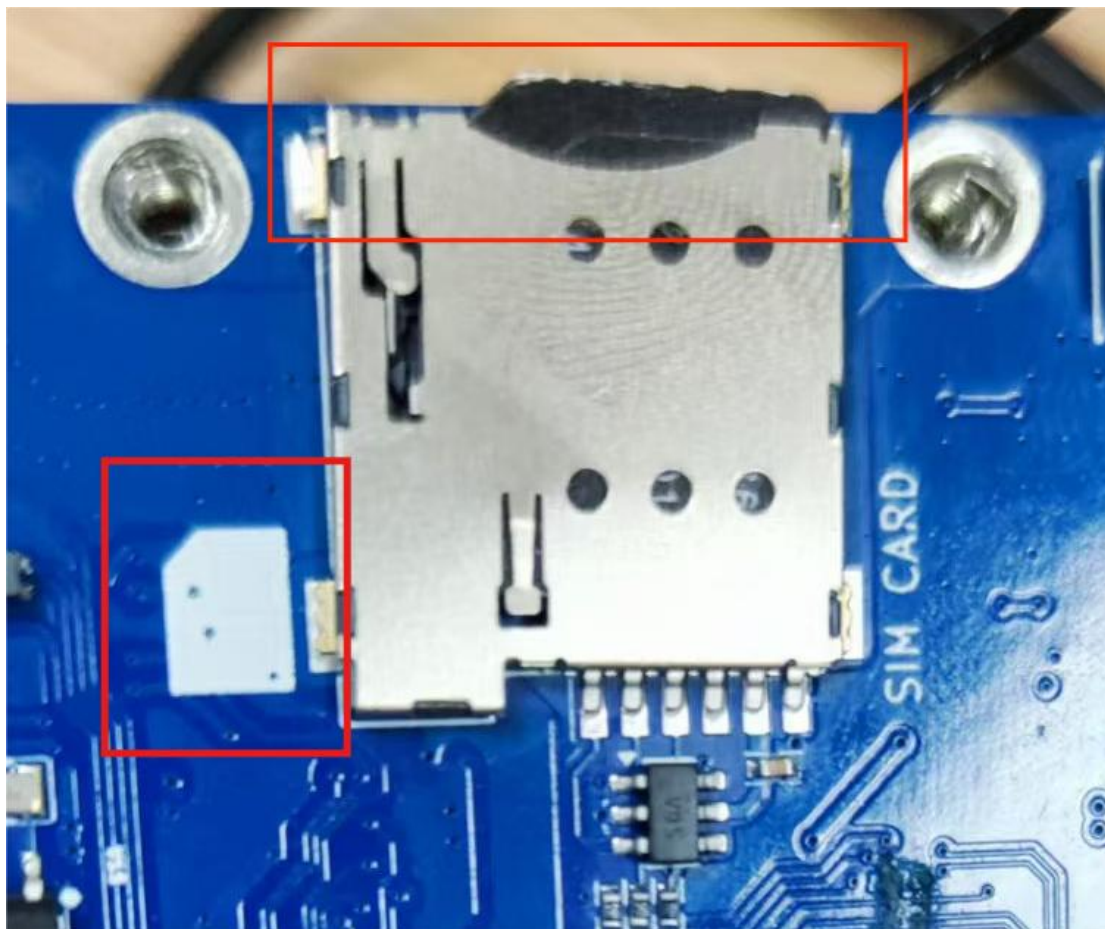
## 4.11.4G 模块

当前单板机预留 4G 模块接口，可支持部分移远 4G 模块的安装。相应模块请自行在网上购买，注意购买时需要买天线，单单模块是不能正常工作的！（备注：移远有许多型号的 4G 模块，本司测试过的是 EC20CEHCLGR06A08M1G\_AUD，不同的模块功能不一样，比如支持的运营商不一样，有一些模块不支持物联网卡等，详细请咨询卖家），其他型号模块请自行测试，理论上驱动一样，有需求找移远技术支持）。将 EC20 4G 模块插到 4G 模块接口处，拧上螺丝。保证 4G 模块与座子接口吻合连接。请使用原装天线，把天线连接到 4G 模块的 MAIN 接口处。

然后正确插入 4G 卡（**请仔细查看 SIM 卡座旁边的图示，按照图示方向插入 SIM 卡**），系统启动后 4G 模块接口旁的 WWAN LED 会闪烁发出绿光，插入模块情况如下图：







使用如下方式联网并测试网络通断:

```
[root@EBYTE-IMX6UL ~ ]# quectel-CM &
[1] 293
[root@EBYTE-IMX6UL ~ ]# [11-15_16:07:19:302] QConnectManager_Linux_V1.6.4
[11-15_16:07:19:312] Find /sys/bus/usb/devices/1-1.4 idVendor=0x2c7c idProduct=0x125,
bus=0x001, dev=0x004
[11-15_16:07:19:314] Auto find qmichannel = /dev/cdc-wdm0
[11-15_16:07:19:315] Auto find usbnet_adapter = wwan0
[11-15_16:07:19:315] netcard driver = qmi_wwan, driver version = 5.10.9-g90918d460
d17-dirty
[11-15_16:07:19:316] Modem works in QMI mode
[11-15_16:07:19:336] cdc_wdm_fd = 7
[11-15_16:07:20:338] QmiWwanInit message timeout
[11-15_16:07:21:437] Get clientWDS = 7
```

```
[11-15_16:07:21:469] Get clientDMS = 1
[11-15_16:07:21:501] Get clientNAS = 2
[11-15_16:07:21:533] Get clientUIM = 1
[11-15_16:07:21:565] Get clientWDA = 1
[11-15_16:07:21:597] requestBaseBandVersion EC20CEHCLGR06A08M1G_AUD
[11-15_16:07:21:725] requestGetSIMStatus SIMStatus: SIM_READY
[11-15_16:07:21:757] requestGetProfile[1] ///0/IPV4V6
[11-15_16:07:21:789] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[11-15_16:07:21:821] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[11-15_16:07:21:822] ifconfig wwan0 0.0.0.0
[11-15_16:07:21:864] ifconfig wwan0 down
[11-15_16:07:21:949] requestSetupDataCall WdsConnectionIPv4Handle: 0x86da2050
[11-15_16:07:22:077] ifconfig wwan0 up
[11-15_16:07:22:135] busybox udhcpd -f -n -q -t 5 -i wwan0
udhcpd: started, v1.35.0
udhcpd: broadcasting discover
udhcpd: broadcasting discover
udhcpd: broadcasting discover
udhcpd: broadcasting discover
udhcpd: broadcasting discover
udhcpd: broadcasting discover
udhcpd: no lease, failing
[11-15_16:07:37:737] File:ql_raw_ip_mode_check Line:137 udhcpd fail to get ip addresses, try next:
[11-15_16:07:37:737] ifconfig wwan0 down
[11-15_16:07:37:766] echo Y > /sys/class/net/wwan0/qmi/raw_ip
[11-15_16:07:37:766] ifconfig wwan0 up
[11-15_16:07:37:815] busybox udhcpd -f -n -q -t 5 -i wwan0
udhcpd: started, v1.35.0
```





## 4.12.USB

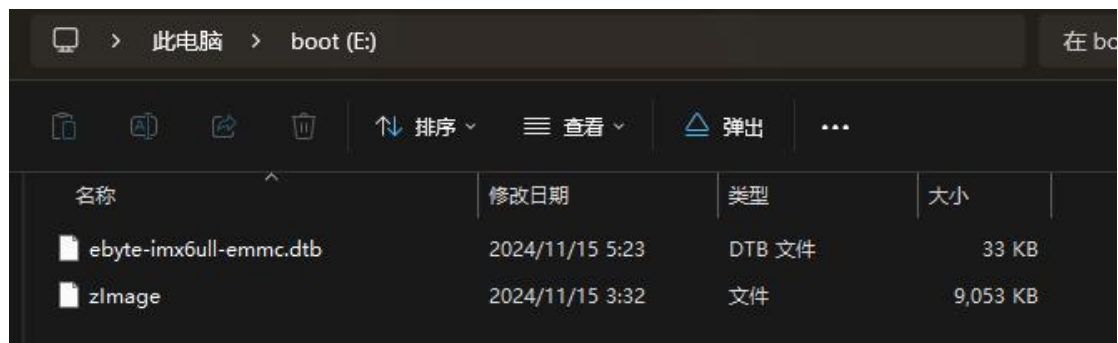
### 4.12.1. OTG

在单板机上存在一路 USB-OTG 接口，默认作为主机使用，通过 typec 转 usb 连接器连接 us 设备，将会自动挂载，比如鼠标，u 盘。同时用户也可以单独配置为从机模式。本章节主要描述如何测试 OTG 的从机功能，主机功能与 USB HOST 接口一致。

连接 PC 机与单板机的 OTG 接口，使用如下命令共享 EFI 分区给 PC 机：

```
modprobe g_mass_storage file=/dev/mmcbk1p1 removable=1
```

共享后，可通过电脑查看到如下文件：



### 4.12.2. HOST

本节通过相关命令或热插拔、USB HUB 验证 USB Host 驱动的可行性，实现读写 U 盘的功能、usb 枚举功能。

插入 U 盘后可通过如下命令查找存储设备。

```
[root@EBYTE-IMX6UL ~]# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sda	8:0	1	28.7G	0	disk	
`-sda1	8:1	1	28.7G	0	part	/media/usb0
mmcblk1	179:0	0	7.3G	0	disk	
-mmcblk1p1	179:1	0	35M	0	part	/boot
`-mmcblk1p2	179:2	0	7.2G	0	part	/
mmcblk1boot0	179:8	0	4M	1	disk	
mmcblk1boot1	179:16	0	4M	1	disk	

可以看到此处多出了一个 sda 设备,同时其第一个分区 sda1 挂载到了`/media/usb0`目录。

进入该目录可以发现 U 盘中存在的文件,且可以通过如下命令测试当前 U 盘的读写性能:

```
[root@EBYTE-IMX6UL ~ ]# dd if=/dev/zero of=/media/usb0/dd.bin bs=10M count=2
conv=fsync

2+0 records in

2+0 records out

20971520 bytes (21 MB, 20 MiB) copied, 36.4012 s, 576 kB/s

[root@EBYTE-IMX6UL ~ ]# echo 3 > /proc/sys/vm/drop_caches

[root@EBYTE-IMX6UL ~ ]# dd of=/dev/zero if=/media/usb0/dd.bin bs=10M

2+0 records in

2+0 records out

20971520 bytes (21 MB, 20 MiB) copied, 0.799956 s, 26.2 MB/s
```

可以看到此时的 U 盘读取速度为 26.2MB/s,符合一个 USB 2.0 U 盘的正常读取速度。

## 4.13.WIFI 模块

单板机中通过本司的 E103-RTL8811CU 模块提供了基础的 WIFI 功能。

### 4.13.1. STA 模式

使用如下命令扫描附近可用 wifi 列表:

```
[root@EBYTE-IMX6UL ~ ]# iw dev wlan0 scan | grep SSID

SSID: TP-LINK_8BF6

SSID: zhendu

SSID: TEST_ZW

SSID: TP-LINK_\xef\xbc\x88\xe7\xa1\xac\xe4\xbb\xb6\xe7\xbb\x84)

SSID: B05_2.4G_\xe6\x89\x93\xe5\x8d\xb0\xe6\x9c\xba

SSID: B05_5G_1

SSID: \xe5\xa4\xa7\xe5\x8e\x85\xe7\xbd\x91\xe7\xbb\x9cD-5G

SSID: ChinaNet-F2FH_5G
```

这里以 TP-LINK\_8BF6 为例连接 WIFI 并使用:

1. 配置网络连接并查看配置信息:

```
[root@EBYTE-IMX6UL ~ ]# wpa_passphrase TP-LINK_8BF6 EC12345678 >> /etc/wpa_supplicant.conf

[root@EBYTE-IMX6UL ~ ]# cat /etc/wpa_supplicant.conf

ctrl_interface=/var/run/wpa_supplicant

ap_scan=1

network={
    key_mgmt=NONE
}

network={
    ssid="TP-LINK_8BF6"
    #psk="EC12345678"
    psk=9c26e18d1d12d9542f596ad7e4004d5c7c8cf1c157b90f06d10cc3fe2ba1cbbc
}
```

2. 连接网络:

```
[root@EBYTE-IMX6UL ~ ]# wpa_supplicant -B -i wlan0 -c /etc/wpa_supplicant.conf

Successfully initialized wpa_supplicant

rfkill: Cannot open RFKILL control device

[root@EBYTE-IMX6UL ~ ]# udhcpc -i wlan0

udhcpc: started, v1.35.0

udhcpc: broadcasting discover

udhcpc: broadcasting select for 192.168.1.156, server 192.168.1.1

udhcpc: lease of 192.168.1.156 obtained from 192.168.1.1, lease time 7200

deleting routers

adding dns 114.114.114.114

adding dns 61.139.2.69
```

```
[root@EBYTE-IMX6UL ~ ]# ifconfig wlan0

wlan0      Link encap:Ethernet  HWaddr E0:54:E9:60:57:39

            inet addr:192.168.1.156  Bcast:192.168.1.255  Mask:255.255.255.0

            inet6 addr: fe80::e17b:f9aa:351:b769/64  Scope:Link

            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

            RX packets:1111 errors:0 dropped:0 overruns:0 frame:0

            TX packets:45 errors:0 dropped:0 overruns:0 carrier:0

            collisions:0 txqueuelen:1000

            RX bytes:993561 (970.2 KiB)  TX bytes:6830 (6.6 KiB)
```

### 3. 测试网络连接:

```
[root@EBYTE-IMX6UL ~ ]# ping -I wlan0 120.27.154.119 -c 4

PING 120.27.154.119 (120.27.154.119): 56 data bytes

64 bytes from 120.27.154.119: seq=0 ttl=113 time=320.330 ms

64 bytes from 120.27.154.119: seq=1 ttl=113 time=351.853 ms

64 bytes from 120.27.154.119: seq=2 ttl=113 time=325.947 ms

64 bytes from 120.27.154.119: seq=3 ttl=113 time=316.699 ms

--- 120.27.154.119 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 316.699/328.707/351.853 ms
```

## 4.13.2. AP 模式

使用单板机作为 WiFi 热点，需要为每一个接入该热点的终端（例如手机）分配 IP，路由等网络参数。

在创建 wifi 热点功能过程中我们需要以下几个程序共同工作：hostapd，dnsmasq。同时为了可以访问外部网络，我们还需要使用 iptables 程序配置网络路由表。

### 4.13.2.1. 配置 DHCP 服务器

我们使用 dnsmasq 作为 DHCP 服务器，首先创建对应的配置文件 /etc/dnsmasq.conf：



```

interface=wlan0                # 设置为 wlan0, 指定在哪个接口上分配 IP 地址

dhcp-range=192.168.10.50,192.168.10.150,12h # DHCP 范围: 192.168.1.50 到 192.168.
1.150, 租约时间为 12 小时

dhcp-option=3,192.168.10.1      # 设置网关 (即路由器地址)

dhcp-option=6,192.168.10.1      # 设置 DNS 服务器地址 (通常设置为路由器地
址)

bind-interfaces                 # 防止 dnsmasq 在其他接口上启动
    
```

配置完成后启动 DHCP 服务器功能:

```

[root@EBYTE-IMX6UL ~ ]# ifconfig wlan0 192.168.10.1 up

[root@EBYTE-IMX6UL ~ ]# dnsmasq -C /etc/dnsmasq.conf
    
```

#### 4.13.2.2. 配置 AP 功能

使用 hostapd 配置并转换 wifi 模块作为 AP 功能, 用户需要注意: **不可同时工作在 AP 模块与 STA 模块, 在使用前需要关闭 STA 功能。**

修改 hostapd 的配置文件 /etc/hostapd.conf, 下面是一个简单的示例:

```

interface=wlan0

driver=nl80211

# mode Wi-Fi (a = IEEE 802.11a, b = IEEE 802.11b, g = IEEE 802.11g)

hw_mode=g

ssid=EBYTE

channel=7

wmm_enabled=0

macaddr_acl=0

# Wi-Fi closed, need an authentication

auth_algs=1

ignore_broadcast_ssid=0

wpa=2

wpa_passphrase=12345678

wpa_key_mgmt=WPA-PSK
    
```

```
wpa_pairwise=TKIP
```

```
rsn_pairwise=CCMP
```

此处配置了一个 SSID 为 EBYTE, PASSWD 为 12345678, 加密方式为 WPA2 的无线 wifi 热点。

修改完毕配置文件后, 使用如下命令打开 AP 功能:

```
[root@EBYTE-IMX6UL ~ ]# hostapd -B /etc/hostapd.conf  
rfkill: Cannot open RFKILL control device  
[ 906.106438] RTW: assoc success  
[ 906.109976] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready  
[ 906.129412] RTW: set group key camid:0, addr:e0:54:e9:60:57:39, kid:1, type:AES  
wlan0: interface state UNINITIALIZED->ENABLED  
wlan0: AP-ENABLED
```

此时可以在手机上找到对应的 wifi 网络, 但连接后会发现无法访问外部网络, 还需要用户执行下一步操作, 配置网络转发。

#### 4.13.2.3. 配置网络转发

使用如下命令配置网络转发, 使连接到 wlan0 的设备通过单板机的 eth0 进行网络访问。

```
[root@EBYTE-IMX6UL ~ ]# echo 1 > /proc/sys/net/ipv4/ip_forward  
[root@EBYTE-IMX6UL ~ ]# iptables -t nat -A POSTROUTING -s 192.168.10.1/24 -o eth0 -j MASQUERADE
```

#### 4.13.2.4. 设备管理查看

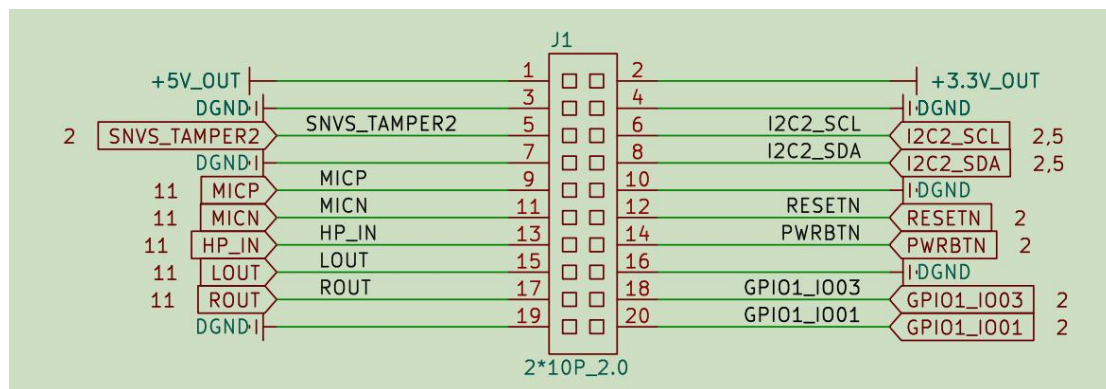
通过如下命令可以查看连接到 wlan0 的设备情况:

```
[root@EBYTE-IMX6UL ~ ]# arp -i wlan0 -a  
? (192.168.10.135) at 1e:92:47:fe:fc:99 [ether] on wlan0  
[root@EBYTE-IMX6UL ~ ]# iw dev wlan0 station dump  
Station 1e:92:47:fe:fc:99 (on wlan0)  
inactive time: 110 ms  
signal: -61 dBm
```

current time: 1731664035416 ms

## 4.14. 音频

单板机中通过 ES8316 提供了音频功能，但是默认并未引出耳机座，而是位于 UIO1 扩展排针处：



通过如下命令可以查看当前 codec 支持的功能情况：

```
[root@EBYTE-IMX6UL ~]# aplay -D hw:0,0 --dump-hw-params /dev/zero

Playing raw data '/dev/zero' : Unsigned 8 bit, Rate 8000 Hz, Mono

HW Params of device "hw:0,0":

-----

ACCESS:  MMAP_INTERLEAVED RW_INTERLEAVED

FORMAT:  S16_LE S24_LE

SUBFORMAT:  STD

SAMPLE_BITS: [16 32]

FRAME_BITS: [16 64]

CHANNELS: [1 2]

RATE: [16000 48000]

PERIOD_TIME: (333 1024000]

PERIOD_SIZE: [16 16384]

PERIOD_BYTES: [128 65532]

PERIODS: [2 255]

BUFFER_TIME: (666 2048000]
```

```
BUFFER_SIZE: [32 32768]
BUFFER_BYTES: [128 65536]
TICK_TIME: ALL
-----
aplay: set_params:1352: Sample format non available
Available formats:
- S16_LE
- S24_LE
```

用户通过自行连接扩展后，可以使用如下命令测试音频的输出功能：

```
speaker-test -c 2

speaker-test 1.2.6

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 64 to 16384
Period size range from 32 to 8192
Using max buffer size 16384
Periods = 4
was set period_size = 4096
was set buffer_size = 16384

0 - Front Left
1 - Front Right

Time per period = 5.636287

0 - Front Left
1 - Front Right
```

Time per period = 5.973515

0 - Front Left

1 - Front Right

使用如下命令录制一个 5 秒的音频:

```
[root@EBYTE-IMX6UL ~ ]# arecord -f S16_LE -r 48000 record.wav -c 2 -vv -d 5
```

```
Recording WAVE 'record.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

```
Plug PCM: Hardware PCM card 0 'ES8316' device 0 subdevice 0
```

Its setup is:

```
stream      : CAPTURE
```

```
access      : RW_INTERLEAVED
```

```
format      : S16_LE
```

```
subformat   : STD
```

```
channels    : 2
```

```
rate       : 48000
```

```
exact rate  : 48000 (48000/1)
```

```
msbits     : 16
```

```
buffer_size : 16384
```

```
period_size : 4096
```

```
period_time : 85333
```

```
tstamp_mode : NONE
```

```
tstamp_type : MONOTONIC
```

```
period_step : 1
```

```
avail_min   : 4096
```

```
period_event : 0
```

```
start_threshold : 1
```

```
stop_threshold : 16384
```

```
silence_threshold: 0
```

```
silence_size : 0
```

```
boundary      : 1073741824  
appl_ptr      : 0  
hw_ptr        : 0  
#####      + | 29%
```

## 4.15.LED

Linux 系统提供了一个独立的子系统以方便从用户空间操作 LED 设备，该子系统以文件的形式为 LED 设备提供操作接口。这些接口位于 `/sys/class/leds` 目录下。在硬件资源列表中，我们已经列出了开发板上所有的 LED。下面通过命令读写 `sysfs` 的方式对 LED 进行测试。

下述命令均为通用命令，也是操控 LED 的通用方法。

# 查看当前系统配置的所有 LED

```
ls /sys/class/leds/
```

# 查看用户 LED 当前状态

```
cat /sys/class/leds/user/brightness
```

# 控制用户 LED 点亮

```
echo 1 > /sys/class/leds/user/brightness
```

# 恢复 LED 为熄灭

```
echo 0 > /sys/class/leds/user/brightness
```

# 查看用户 LED 支持根据什么外设的工作状态呼吸闪烁

```
cat /sys/class/leds/user/trigger
```

# 控制用户 LED 按照 `mmc1` 工作状态呼吸闪烁

```
echo mmc1 > /sys/class/leds/user/trigger
```

# 测试 LED 是否按照 `mmc1` 工作状态闪烁。使用下面的方式当写入数据时 LED 会常亮一小段时间，直到写入完成。

```
dd if=/dev/zero of=test.bin bs=1M count=10 conv=fsync
```

## 4.16. 按键

在该单板机上，总共存在两个按键：RES 和 PWR，分别对应复位功能和开关机功能，在当前的配置当中，我们关闭了 PWR 按键的短按关机功能，但保留长按强制关机功能。

### 4.16.1. 复位

当按下 RES 按键，系统会立刻重启。

```
[root@EBYTE-IMX6UL ~ ]# memtester 100M 1

memtester version 4.5.0 (32-bit)

Copyright (C) 2001-2020 Charles Cazabon.

Licensed under the GNU General Public License version 2 (only).

pagesize is 4096

pagesizemask is 0xffff000

want 100MB (104857600 bytes)

got 100MB (104857600 bytes), trying mlock ...locked.

Loop 1/1:

  Stuck Address      : testing  3

U-Boot 2020.04-g93edef74 (Nov 15 2024 - 10:23:36 +0800)

CPU:  i.MX6ULL rev1.1 792 MHz (running at 396 MHz)
CPU:  Industrial temperature grade (-40C to 105C) at 54C

Reset cause: POR

Model: EBYTE i.MX6 ULL EVK Board

Board: ebyte I.MX6ULL

DRAM:  512 MiB

MMC:   FSL_SDHC: 0, FSL_SDHC: 1

Loading Environment from MMC... OK
```

从上面的 log 信息中可以发现, 系统还在执行 memtester, 但是立刻进入了 u-boot 信息打印。此时就是笔者按下 RES 按键的时机。

## 4.16.2. 电源

电源按键是用于当系统关机后, 开机使用, 或长按强制关机。当关机后, 系统无法通过 RES 复位启动, 必须使用 PWR 按键启动。

```
Sent SIGTERM to all processes
Sent SIGKILL to all processes
Requesting system poweroff
[ 23.793078] cfg80211: failed to load regulatory.db
[ 23.822778] imx-sdma 20ec000.sdma: loaded firmware 3.6
[ 23.830381] RTW: stop cmd thread during rtw_dev_shutdown
[ 23.860818] RTW: cmd thread is stopped during rtw_dev_shutdown
[ 23.877776] ci_hdrc ci_hdrc.1: remove, state 1
[ 23.882310] usb usb1: USB disconnect, device number 1
[ 23.887567] usb 1-1: USB disconnect, device number 2
[ 23.892604] usb 1-1.1: USB disconnect, device number 3
[ 23.974794] usb 1-1.3: USB disconnect, device number 4
[ 24.003973] RTW: rtw_ndev_uninit(wlan0) ifl
[ 24.093790] RTW: rtw_dev_unload: driver not in IPS
[ 24.105650] ci_hdrc ci_hdrc.1: USB bus 1 deregistered
[ 24.137017] reboot: Power down
```

```
U-Boot 2020.04-g93edef74 (Nov 15 2024 - 10:23:36 +0800)
```

```
CPU: i.MX6ULL rev1.1 792 MHz (running at 396 MHz)
```

```
CPU: Industrial temperature grade (-40C to 105C) at 49C
```

```
Reset cause: POR
```



```
Model: EBYTE i.MX6 ULL EVK Board  
  
Board: ebyte I.MX6ULL  
  
DRAM: 512 MiB  
  
MMC: FSL_SDHC: 0, FSL_SDHC: 1  
  
Loading Environment from MMC... OK
```

上面的 log 信息就是一次系统关机到按下 PWR 按键重新开机的过程。

## 5. 参考资料

- ❖Linux kernel 开源社区: <https://www.kernel.org/>
- ❖NXP 开发社区: <https://community.nxp.com>
- ❖IMX6ULL date-sheet
- ❖Buildroot-manual: <https://buildroot.org/downloads/manual/manual.html>
- ❖imx-yocto-LF5.10.9\_1.0.0

## 6. 修订说明

修订说明表

版本	修改内容	修改时间	编制	校对	审批
V1.0	初稿	24-11-11	WYQ	HSL	WFX
V1.1	增加章节快速开始, 修订错误拼写	24-12-12	WYQ	HSL	WFX

## 7. 关于我们



销售热线: 4000-330-990

技术支持: [support@cdebyte.com](mailto:support@cdebyte.com) 官方网站: <https://www.ebyte.com>

公司地址: 四川省成都市高新西区西区大道 199 号 B5 栋

((( )))<sup>®</sup>  
**EBYTE 成都亿佰特电子科技有限公司**  
Chengdu Ebyte Electronic Technology Co.,Ltd.