

密级状态：绝密() 秘密() 内部(☒) 公开()

RK_Android 平台以太网调试说明

(技术部, MID 组)

文件状态： [] 正在修改 [<input checked="" type="checkbox"/>] 正式发布	当前版本：	V1.1
	作 者：	胡卫国
	完成日期：	2013-09-29
	审 核：	
	完成日期：	

福州瑞芯微电子有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有, 翻版必究)

版 本 历 史

版本号	作者	修改日期	修改说明	备注
V1.0	胡卫国	2013-09-29	初始版本	
V1.1	胡卫国	2014-3-3	增加 USB 以太网调试说明	
V1.2	胡卫国	2014-6-25	重新整理相关补丁	

目 录

1 以太网说明.....	3
2 VMAC 模块软件配置.....	4
2.1 KERNEL 配置.....	4
2.2 板级文件修改.....	5
2.3 PHY 电源控制.....	5
2.4 MAC 地址问题.....	6
3 USB 以太网模块软件配置.....	6
4 常见问题及排查.....	7
4.1 以太网与 WiFi 优先级问题.....	7
4.2 使用以太网时有些应用无法上网.....	8
4.3 VMAC 部分.....	8
4.3.1 机器不断重启.....	8
4.3.2 以太网无法连接上.....	9
4.3.3 开机概率性无法连接上以太网.....	10
4.3.4 组播 (multicast) 功能无法使用.....	11
4.3.5 RX 存在丢包现象.....	11
4.3.6 大量 64 字节 udp 包造成 VMAC RX 异常问题.....	12
4.4 USB 以太网部分.....	13
4.4.1 以太网无法使用问题排查.....	13
4.4.2 USB 以太网不稳定.....	14
4.4.3 RTL8152 由于无 MAC 地址导致异常进不了 Android 系统.....	14
5 补丁下载地址.....	14

1 以太网说明

目前 RK MID 平台支持以下两种以太网

1. 使用主控的 VMAC 模块加上外接的 PHY 实现；
2. 使用 USB 以太网卡实现；

目前支持的 PHY 芯片有：

LAN8720A

RTL8021F

支持的 USB 以太网卡有：

dm9620

ax8872b

sr9700

rtl8152b

2 VMAC 模块软件配置

2.1 Kernel 配置

make menuconfig 中需要打开 “RK29 VMAC ethernet support”

```
| Location:
|
| -> Device Drivers
|
|     -> Network device support
|
|         -> Ethernet (10 or 100Mbit)
```

Vmac 是 RK 主控集成的 IP，需要外部添加 phy 芯片才能工作，具体驱动代码在：

drivers/net/rk29_ymac.c

注意：在 RK3188 平台上，由于 RMII 接口与 SDMMC1 接口复用，所以需要关掉 SDMMC1 接口配置，防止干扰：

CONFIG_SDMMC1_RK29 = n

RMII 信号引脚是与其它功能脚复用的，如下 3188 芯片引脚复用情况。

```
GPIO3_C0/SDMMC1_CMD/RMII_TX_EN
GPIO3_C1/SDMMC1_D0/RMII_TX_D1
GPIO3_C2/SDMMC1_D1/RMII_TX_D0
GPIO3_C3/SDMMC1_D2/RMII_RX_D0
GPIO3_C4/SDMMC1_D3/RMII_RX_D1
GPIO3_C5/SDMMC1_CLKO/RMII_CLKO
GPIO3_C6/SDMMC1_DET/RMII_RX_ERR
GPIO3_C7/SDMMC1_WP/RMII_CSR_VALID
GPIO3_D0/SDMMC1_PWR/MII_MD
GPIO3_D1/SDMMC1_BACKEND/MII_MDCLK
```

可通过查看芯片中的寄存器来确认引脚是否工作在 RMII 格式：

RK3188:

使用 rmii_clockout(也就是 50M 的工作 clock 由主控供)

```
io -4 0x20008098
20008098: 0000aaaa // 低 16bit
io -4 0x2000809c
2000809c: 0000000a // 低 4bit
```

使用 rmii_clockin(也就是 50M 的工作 clock 由外部供)

```
io -4 0x20008098
20008098: 0000aeaa // 低 16bit
io -4 0x2000809c
2000809c: 0000000a // 低 4bit
```

RK3066:

使用 rmii_clockout(也就是 50M 的工作 clock 由主控供)

```
io -4 0x200080c0
200080c0: 0000aaaa // 低 16bit
io -4 0x200080c4
200080c4: 0000000a // 低 4bit
```

2.2 板级文件修改

如果是 RK3168, RK3188 平板，需要在板级文件中加入以下 vmac 相应资源

```
#ifndef CONFIG_RK29_VMAC

#define PHY_PWR_EN_GPIO RK30_PIN0_PC0 // 根据实际硬件配置

#define PHY_PWR_EN_VALUE GPIO_HIGH

#include "../mach-rk30/board-rk31-sdk-vmac.c"

#endif
```

2.3 phy 电源控制

Phy 芯片工作时需要 power on 或 reset，这是由主控 gpio 在 vmac 驱动初始时控制的，具体在 2.2 节中的板级文件里定义。

2.4 MAC 地址问题

默认 MAC 地址是随机生成的，每次重新开机后 MAC 地址都会改变。如果想固定 MAC 地址，可通过工具将 MAC 地址写入到 flash 中，具体如下：

1. 使用 UpgradeDlTool 将工具写入到 flash 中；
2. 在 kernel 的 menuconfig 中选择 “ Ethernet mac from IDB ”

```
| Location:  
  
| -> Device Drivers  
  
| -> Network device support  
  
| -> Ethernet (10 or 100Mbit)  
  
| -> Ethernet mac source
```

3. VMAC 驱动会优先使用写入到 flash 中的 MAC 地址

3 USB 以太网模块软件配置

发布的 SDK 中默认支持，具体配置在：

```
| Location:  
  
| -> Device Drivers  
  
| -> Network device support (NETDEVICES [=y])  
  
| -> USB Network Adapters
```

```
<*> Multi-purpose USB Networking Framework
<*> ASIX AX88xxx Based USB 2.0 Ethernet Adapters
- * - CDC Ethernet support (smart devices such as cable modems)
< > CDC EEM support
<*> CDC NCM support
< > Davicom DM9601 based USB 1.1 10/100 ethernet devices
<*> Davicom DM9620 based USB 1.1 10/100 ethernet devices
<*> Willing Electronic SR9700 based USB 2.0
<*> Willing Electronic SR9800 based USB 2.0
< > SMSC LAN75XX based USB 2.0 gigabit ethernet devices
< > SMSC LAN95XX based USB 2.0 10/100 ethernet devices
< > Genesys GL620USB-A based cables
<*> NetChip 1080 based cables (Laplink, ...)
< > Prolific PL-2301/2302/25A1 based cables
< > MosChip MCS7830 based Ethernet adapters
< > Host for RNDIS and ActiveSync devices (EXPERIMENTAL)
<*> Simple USB Network Links (CDC Ethernet subset)
[ ] ALi M5632 based 'USB 2.0 Data Link' cables
[ ] AnchorChips 2720 based cables (Xircom PGUNET, ...)
[ * ] eTEK based host-to-host cables (Advance, Belkin, ...)
[ * ] Embedded ARM Linux links (iPaq, ...)
[ ] Epson 2888 based firmware (DEVELOPMENT)
[ ] KT Technology KC2190 based cables (Instanet)
[ * ]
```

4 常见问题及排查

4.1 以太网与 WiFi 优先级问题

WiFi 是 Android 的首选网络，在同时连接 WiFi 与以太网时，优化选择使用 WiFi。也就 WiFi 连接上的时候，再插入以太网是连接不上的，需要断开或关闭 WiFi，以太网才能连接上。

若需要设置以太网优先级更高，可打上以下补丁试试：

```
+++ b/services/java/com/android/server/ConnectivityService.java

@@ -1773,7 +1773,7 @@ private NetworkStateTracker makeWimaxStateTracker() {

    //log("mNetworkPreference = "+mNetworkPreference+" mActiveDefaultNetwork = "+
mActiveDefaultNetwork);

    if (mNetConfigs[type].isDefault()) {

        if (mActiveDefaultNetwork != -1 && mActiveDefaultNetwork != type) {

-            if ((type != mNetworkPreference &&

+            if ((*type != mNetworkPreference &&*/

                mNetConfigs[mActiveDefaultNetwork].priority >

                mNetConfigs[type].priority) /*||
```



```
mNetworkPreference == mActiveDefaultNetwork*/) {
```

4.2 使用以太网时有些应用无法上网

网络上很多应用，例如 QQ 等，在使用以太网时，会提示没有连接网络，导致无法上网。这是因为这些应用只认 WiFi 网络与移动网络，无法识别到以太网，这是应用本身的问题。

4.3 VMAC 部分

有异常时请打上以下补丁，保证 VMAC 部分代码是最新的，然后再排查：

Android4.1: Android_4.1_ethernet_update.rar

Android 4.2: Android_4.2_ethernet_update.rar

4.3.1 机器不断重启

从 kernel 的打印 log 看到：

```
net eth0: no PHY found
```

这是没有识别到 phy 设备，需要从以下几个方面来排查：

1) phy 是否有成功供上电？ power on 或 reset 是否正常使能？

2) rmii_clk 是否有异常？

a) rmii_clk 要求是 50M，如果不是 50M，首先确认主控是否有分出 50M 的 clock

可通过 `cat proc/clocks | busybox grep mac` 查看

```
mac_pll_div on 50 MHz usecount = 2 parent = general_pll
```

```
mac_ref 50 MHz usecount = 2 parent = mac_pll_div
```

```
mii_tx on 50 MHz usecount = 1 parent = mac_ref
```

```
hclk_mac on 150 MHz usecount = 1 parent = hclk_periph
```

如果是 RK3188T 的芯片，由于主控分不出 50M rmii_clk，需要使用外部晶体来供 clock，具体硬件及软件补丁见《关于 RK3188T 配 RTL8201F 注意点 20131226.rar》

如果是 RK3066 Android 4.1 的版本，尝试《rk3066_vmac_mac_ref_clock.patch》试试

b) 幅度需要达到 2V 以上

如果幅度达不到要求，从以下两点排查：

i) 如果是 RK31xx 芯片，有可能是主控 IO 驱动能力不够，先确认

arch/arm/mach-rk30/board-rk31-sdk-vmac.c 中函数 rk30_rmii_io_init 中是否有以下增加 gpio 驱动强度的操作：

```
grf_writel(0x0f<<16|0x0f,GRF_IO_CON3);
```

ii) 主控 AP0_VCC 需要供 3.3V 电压

iii) 将主控 rmii_clk 与 phy 芯片断开，测一下主控输出的 rmii_clk 幅度是否正常

c) 注意：主控 rmii 接口目前与其它模块复用，可能会有异常

i) 如果是 RK3066，rmii 接口与 cifl(camera 模块) 复用，需要打上《rk3066_vmac_cifl_compatible.patch》

ii) 如果是 RK31xx，rmii 接口与 sdio1(sdio wifi 模块)复用

以上两点，首先硬件上要排除相互干扰。

d) 如果 phy 是 RTL8201F，那么注意 phy 的第 15 脚需要断开，具体请参考《关于 RK3188T 配 RTL8201F 注意点 20131226.rar 中的 RK3188&3188T_以太网_使用 RTL8201F 更新以及注意点_V1.0.pdf》

4.3.2 以太网无法连接上

1) 先确认网线是否有接上

2) 查看以太网卡 eth0 接口状态信息

可通过 `busybox ifconfig eth0` 来查看

```
eth0      Link encap:Ethernet  HWaddr 02:C9:F3:B2:BA:81

          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)

Interrupt:51 Base address:0x8000

i) 如果发现 TX packets, RX packets 都为 0

那么可能是 RJ45 的座子有问题，座子跟主板上的管脚定义是否一样？

ii) 如果发现 TX packets 不为 0, RX packets 为 0

如果是 RK3066，那么打上补丁《rk3066_vmac_rx_fail.patch》

如果是 RK3188 平台，打上以下补丁试试：

```
+++ b/arch/arm/mach-rk30/board-rk31-sdk-vmac.c
@@ -3,10 +3,6 @@ static int rk30_vmac_register_set(void)
{
    //config rk30 vmac as rmii
    writel_relaxed(0x3 << 16 | 0x2, RK30_GRF_BASE + GRF_SOC_CON1);
+   int val = readl_relaxed(RK30_GRF_BASE + GRF_IO_CON3);
+   writel_relaxed(val | 0xf << 16 | 0xf, RK30_GRF_BASE + GRF_IO_CON3); // 提高驱动强度
+   val = readl(RK30_GRF_BASE + GRF_SOC_CON2);
+   writel(0x1 << 6 | 0x1 << 22 | val, RK30_GRF_BASE + GRF_SOC_CON2); // support the data
package without heade
    return 0;
}
```

4.3.3 开机概率性无法连接上以太网

Phy 芯片需要做一次复位工作，确认 drivers/net/rk29_vmac.c 中是否已经有以下修改：

```
@@@ -1070,8 +1070,11 @@ int vmac_open(struct net_device *dev)
    clk_enable(clk_get(NULL, "mac_ref"));

    //phy power on
-   if (pdata && pdata->rmii_power_control)
+   if (pdata && pdata->rmii_power_control) {
+       pdata->rmii_power_control(0);
+       msleep(100);
+       pdata->rmii_power_control(1);
+   }
```

```
msleep(1000);
```

4.3.4 组播（multicast）功能无法使用

确认 drivers/net/rk29_vmac.c 中是否已经有以下修改

```
@@@ -1432,10 +1436,10 @@@ static void vmac_set_multicast_list(struct net_device *dev)
    spin_lock_irqsave(&ap->lock, flags);

    promisc = !(dev->flags & IFF_PROMISC);
-   reg = vmac_readl(ap, ENABLE);
+   reg = vmac_readl(ap, CONTROL);
    if (promisc != !(reg & PROM_MASK)) {
        reg ^= PROM_MASK;
-       vmac_writel(ap, reg, ENABLE);
+       vmac_writel(ap, reg, CONTROL);
    }

    if (dev->flags & IFF_ALLMULTI)
```

4.3.5 RX 存在丢包现象

```
busybox ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 42:FF:66:4D:B9:60

          inet addr:192.168.0.24   Bcast:192.168.0.255   Mask:255.255.255.0

          inet6 addr: fe80::40ff:66ff:fe4d:b960/64 Scope:Link

          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

          RX packets:47 errors:19 dropped:0 overruns:0 frame:19

          TX packets:37 errors:0 dropped:0 overruns:0 carrier:0

          collisions:0 txqueuelen:1000

          RX bytes:4592 (4.4 KiB)  TX bytes:4582 (4.4 KiB)

          Interrupt:51 Base address:0x8000
```

如果出现 RX packets 存在丢包现象，那么可能与 mac 中 50M clock 精度有关，如果是 3188T 芯片，请使用外部的 clock，参考补丁《关于 RK3188T 配 RTL8201F 注意点 20131226.rar》

4.3.6 大量 64 字节 udp 包造成 VMAC RX 异常问题

使用 iperf 发送长度为 64 字节的 udp 包，会造成 VMAC 来不及接收，最终造成无法接收，一直无法恢复问题，可通过以下补丁解决：

```
+++ b/drivers/net/rk29_vmac.c

@@@ -1134,6 +1134,9 @@@ int vmac_open(struct net_device *dev)

    temp = (RX_BDT_LEN << 24) | (TX_BDT_LEN << 16) | TXRN_MASK | RXRN_MASK;

    vmac_writel(ap, temp, CONTROL);

+      /* lock up fix enable*/^M
+      vmac_writel(ap, vmac_readl(ap, EXT1) | (1<<23), EXT1);

    /* enable, after all other bits are set */

    vmac_writel(ap, temp | EN_MASK, CONTROL);

diff --git a/drivers/net/rk29_vmac.h b/drivers/net/rk29_vmac.h
index 1f2ecf2..496a32c 100755
--- a/drivers/net/rk29_vmac.h
+++ b/drivers/net/rk29_vmac.h

@@@ -67,6 +67,8 @@@

#define MDIO_DATA_OFFSET      (0x34/0x4)

#define MAC_TXRING_HEAD_OFFSET (0x38/0x4)

#define MAC_RXRING_HEAD_OFFSET (0x3C/0x4)

+ #define EXT1_OFFSET  (0x40/0x4)

+ #define EXT2_OFFSET  (0x44/0x4)

/* STATUS and ENABLE register bit masks
```

4.4 USB 以太网部分

4.4.1 以太网无法使用问题排查

第一步：

USB 以太网接到通过 USB HOST 口连接到主控，先确认是否有以下 USB 设备枚举到的打印：

```
[ 43.234402] DWC_OTG: dwc_otg_hcd_enable, enable host controller
[ 43.344720] DWC_OTG: ^^^^^^^^^^^^^^^^^Host Mode
[ 43.344806] DWC_OTG: Init: Port Power? op_state=1
[ 43.344843] DWC_OTG: Init: Power Port (0)
[ 43.704382] usb 2-1: new high speed USB device number 2 using usb20_host
[ 43.915473] usb 2-1: New USB device found, idVendor=0bda, idProduct=8152 //这里打印出枚举到的设备信息
[ 43.915525] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 43.915587] usb 2-1: Product: USB 10/100 LAN
[ 43.915618] usb 2-1: Manufacturer: Realtek
[ 43.915646] usb 2-1: SerialNumber: 00E04C360001
[ 43.917853] cdc_ether 2-1:2.0: eth0: register 'cdc_ether' at usb-usb20_host-1, CDC Ethernet Device, 00:e0:4c:36:00:01
```

如果没有打印以上信息，证明 USB 枚举失败，需要排查：

- 1) USB HOST 口工作是否正常，可以接鼠标等设备测试；
- 2) USB HOST 口供电是否足够；
- 3) 如果是接在 HUB，确认 HUB 工作是否正常；

第二步：

如果 USB 设备被正确枚举到了，那么可通过以下命令查看 eth0 接口状态：

```
busybox ifconfig eth0
```

```
eth0      Link encap:Ethernet  HWaddr 00:E0:4C:36:00:01
```

// 确认是否有获取到以下 IP 地址

```
inet addr:192.168.0.120 Bcast:192.168.0.255 Mask:255.255.255.0
```

```
inet6 addr: fe80::2e0:4cff:fe36:1/64 Scope:Link
```

```
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```

```
RX packets:577 errors:0 dropped:0 overruns:0 frame:0
```

```
TX packets:465 errors:0 dropped:0 overruns:0 carrier:0
```

```
collisions:0 txqueuelen:1000
```

```
RX bytes:751796 (734.1 KiB) TX bytes:39502 (38.5 KiB)
```

如果没有正确获取到 IP 地址，那么可能是所在局域网有问题，可以在设置中设置成静态 IP 地址试试。

4.4.2 USB 以太网不稳定

例如在播放网络视频时，容易断线，从 log 来看到有以下 USB 异常信息：

```
usb 2-1: USB disconnect, device number 2
```

这个可能是 USB HOST 口供电不足造成的，需要硬件上修改。

4.4.3 RTL8152 由于无 MAC 地址导致异常进不了 Android 系统

如果使用 RTL8152 USB 以太网芯片，由于默认没有 MAC 地址，驱动会报异常，可能导致进入到 Android 系统。请打上以下补丁《rtl8152 解决 MAC 地址为空补丁_4.14.rar》

5 补丁下载地址

补丁可从以下 FTP 地址下载：

FTP 地址： <ftp://www.rockchip.com.cn>

账号名： rkwifi

密码： Cng9280H8t

目录： 26-以太网相关补丁