

第 1 章	MT7620A 中新增串口.....	2
1.1	MT7620A 的串口资源.....	2
1.2	新增串口 2.....	2
1.3	简单测试.....	4

[www.hi-wooya.cc](http://www.hi-wooya.cc)

# 第 1 章 MT7620A 中新增串口

## 本章目标

- 了解 MT7620A 的串口资源
- 掌握如何给 MT7620A 新增串口

## 1.1 MT7620A 的串口资源

通过阅读 MT7620A 的芯片手册得知, MT7620A 一共有 2 个 UART, 分别如下。

串口 1, 即 UARTL, 也是我们常用的调试串口, 当然它也可以作为我们的普通的 GPIO 管脚使用。对应管脚如下。

UARTL	RXD2	GPIO#16
	TXD2	GPO#15

串口 2, 即 UARTF, 也就是我们常说的串口 2 了, 它是与我们的 IIS 等引脚复用的。对应管脚如下。

UARTF	RIN	GPIO#14
	DSR_N	GPIO#13
	DCD_N	GPIO#12
	DTR_N	GPIO#11
	RXD	GPIO#10
	CTS_N	GPIO#9
	TXD	GPIO#8
	RTS_N	GPIO#7

## 1.2 新增串口 2

从官网上下下载下来的 OpenWrt 默认是不支持 MT7620A 的串口 2 的, 即默认不支持扩展串口, 只支持调试串口, 也就是 uartlite。那么接下来我们就通过修改源码的方式, 让 OpenWrt 支持 MT7620A 的串口 2。

OpenWrt 中的芯片的资源管理现在有 dts 文件中, 对于 MT7620A 包括两个 dts 文件, 分别是 MT7620a.dts 和 mt7620a.dtsi, 位于 barrier\_breaker/target/linux/ramips/dts, 那我们扩展串口就需要修改这两个文件。串口在 linux 下的设备节点为/dev/ttyS\*, \*代表 0、1、2, 表示第一个串口, 第二个串口, 第三个串口。所以调试串口为/dev/ttyS0。他作为内核启动命令行。在 mt7620a.dtsi 文件中, 调试串口的设置内容如下。

```
chosen {  
    bootargs = "console=ttyS0,57600";  
};
```

如果要新增串口 2，那么新增的串口将使用/dev/ttyS0 这个设备节点，引出我们需要将调试串口的设备节点改为/dev/ttyS1。将上诉内容做如下修改即可。

```
chosen {  
    bootargs = "console=ttyS1,57600";  
};
```

在 OpenWrt 中的 mt7620a.dtsi 文件中，关于 UARTF 的初始设置内容如下。

```
uart@500 {  
    compatible = "ralink,mt7620a-uart", "ralink,rt2880-uart", "ns16550a";  
    reg = <0x500 0x100>;  
  
    resets = <&rstctrl 12>;  
    reset-names = "uart";  
  
    interrupt-parent = <&intc>;  
    interrupts = <5>;  
  
    reg-shift = <2>;  
  
    status = "disabled";  
};
```

其中 status 为它的默认状态，可以看到它默认是没有使能，所以需要把它修改为使能，将 status 状态值改为“okay”，并增加扩展串口的 GPIO 功能映射，最终修改如下。

```
uart@500 {  
    compatible = "ralink,mt7620a-uart", "ralink,rt2880-uart", "ns16550a";  
    reg = <0x500 0x100>;  
  
    resets = <&rstctrl 12>;  
    reset-names = "uart";  
  
    interrupt-parent = <&intc>;  
    interrupts = <5>;  
  
    reg-shift = <2>;  
  
    status = "okay";
```

```
pinctrl-names = "default";
pinctrl-0 = <&uartf_gpio_pins>;
};
```

检查 mt7620a.dtsi 文件中的 pinctrl 中是否存在 uartf\_gpio\_pins 的映射代码,如果没有要自行加上,新版本是有的,旧版本有可能没有,代码如下。

```
uartf_gpio_pins: uartf_gpio {
    uartf_gpio {
        ralink,group = "uartf";
        ralink,function = "gpio uartf";
    };
};
```

最后,我们还需要修改 MT7620a.dts 文件。

将

```
pinctrl {
    state_default: pinctrl0 {
        gpio {
            ralink,group = "i2c", "uartf";
            ralink,function = "gpio";
        };
    };
};
```

改为

```
pinctrl {
    state_default: pinctrl0 {
        gpio {
            ralink,group = "i2c";
            ralink,function = "gpio";
        };
    };
};
```

即将 `ralink,group = "i2c", "uartf";` 中的 `uartf` 去掉,否则 OpenWrt 会把 `uartf` 作为 GPIO 功能来使用。

### 1.3 简单测试

关于 MT7620A 的两个资源管理文件 MT7620a.dts 和 mt7620a.dtsi,我们已经修改好了,接下来重新编译系统。

```
# make V=99
```

编译完成以后,新生成的固件里面,就已经串口 2 了。

我们使用一个 USB 转 UART TTL 模块将开发板上面的 J403 引出的 UARTF 连接起来。然后

在开发板上面运行如下命令。

```
root@OpenWrt:~#  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~# echo wooya >> /dev/ttyS0  
root@OpenWrt:~#
```

串口二便会输出 wooya 字符串，效果如下。



```
Serial-COM15  
wooya  
wooya  
wooya  
wooya  
wooya  
wooya
```

注意：

- 1). 该教程为我司(<https://wy-wulian.taobao.com/>)原创教程，版权所有；
- 2). 该教程会不断更新、不断深入，详情请咨询我司客服；
- 3). 针对该教程，我们还有 QQ 群和论坛，专门负责技术答疑，详情请咨询我司客服。