



# LTDEV-BF53x 网络 音视频开发板手册

2008

杭州来同科技有限公司

2008年8月6日

版本 V1.2

## 修改日志

序号	日期	修改记录	版本	修改人
1	2008-7-31	创建文档	V1.0	黄辉
2	2008-8-1	修改文档名, 修改文档总体结构	V1.1	黄辉
3	2008-8-5	增加硬件接口说明	V1.2	黄辉
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

# 目 录

1	总体介绍.....	4
2	硬件及接口描述.....	5
2.1	硬件配置.....	5
2.2	主要接口.....	7
2.3	内存及中断分配.....	9
3	开发环境搭建.....	12
3.1	硬件环境搭建.....	12
3.2	软件环境搭建.....	12
3.2.1	建立交叉编译环境.....	12
3.2.2	建立 TFTP 服务器.....	13
4	实验及测试流程.....	13
4.1	U-BOOT 启动.....	13
4.2	在 U-Boot 下测试网络.....	15
4.3	烧写内核.....	16
4.4	在 uClinux 下测试外设.....	18
5	软件及编程方法.....	25
6	光盘内容.....	26

# 1 总体介绍

Blackfin®处理器是一类专为满足当今嵌入式音频、视频和通信应用的计算要求和功耗约束条件而设计的新型 16~32 位嵌入式处理器。Blackfin 处理器基于由 ADI 和 Intel 公司联合开发的微信号架构 (MSA)，它将一个 32 位 RISC 型指令集和双 16 位乘法累加 (MAC) 信号处理功能与通用型微控制器所具有的易用性组合在了一起。这种处理特征的组合使得 Blackfin 处理器能够在信号处理和控制处理应用中均发挥上佳的作用——在许多场合中免除了增设单独的异类处理器的需要。该能力极大地简化了硬件和软件设计实现任务。

目前，Blackfin 处理器在单内核产品中可提供高达 756MHz 的性能。Blackfin 处理器系列中的新型对称多处理器成员在相同的频率条件下实现了性能的翻番。Blackfin 处理器系列还提供了低至 0.8V 的业界领先功耗性能。对于满足当今及未来的信号处理应用（包括宽带无线、具有音频/视频功能的因特网工具和移动通信）而言，这种高性能与低功耗的组合是必不可少的。

所有的 Blackfin 处理器都为系统设计师提供了十分重要的好处，包括：

- 可实现各种新型市场和应用的高性能信号处理和高效控制处理能力
- 可令系统设计师使器件功耗模式与终端系统要求相适应的动态电源管理 (DPM) 能力
- 可确保产品开发时间最小化的易用型混合 16/32 位指令集架构和开发工具套件。

LTDEV-BF53X 开发板基于 ADSP-BF531 嵌入式处理器，开发板提供了两套视频采集方案：一套使用 OV7640 彩色 CMOS 图像传感器采集视频，适用于对图像质量要求不高的应用环境。另一套使用 SAA7113 视频解码芯片，外接模拟摄像头采集视频，适用于对图像质量要求较高的应用环境。

另外，开发板提供了音视频采集、处理、播放功能，丰富的存储介质和传输接口，可根据实际系统灵活设计。

LTDEV-BF53X 开发板适用于安防监控产品，便携式音视频设备、网络音视频设备、生物特征识别以及消费类音视频产品。使用 LTDEV-BF53X 开发板进行产品开发，可大大降低软硬件开发工作量，加速产品的研发周期。

以下是 LTDEV-BF53X 开发板接口示意图：

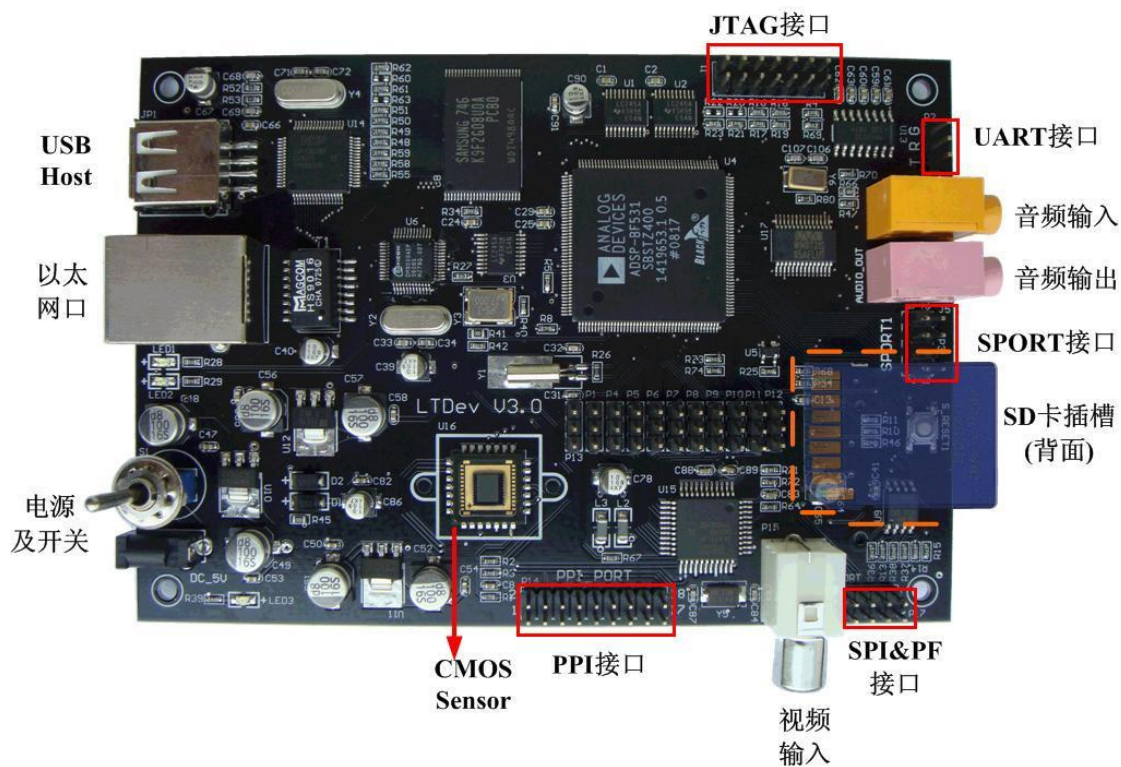


图 1 LTDEV-BF53X 网络音视频开发板接口示意图

## 2 硬件及接口描述

### 2.1 硬件配置

LTDEV-BF53X 网络音视频开发板硬件配置表如下：

硬件配置	
CPU	ADSP-BF531（主频 400MHz）
EEPROM	M25P80 1M Bytes
SDRAM	SAMSUNG K4S561632H 32M Bytes
Nand Flash	SAMSUNG K9F2G08U0A 256M Bytes
以太网	DM9000 10/100M 自适应快速以太网控制器
音频	WM8731 立体声音频编解码芯片
USB Host	飞利浦 ISP1362 USB2.0 接口
SD 接口	1 个 SD 卡插槽

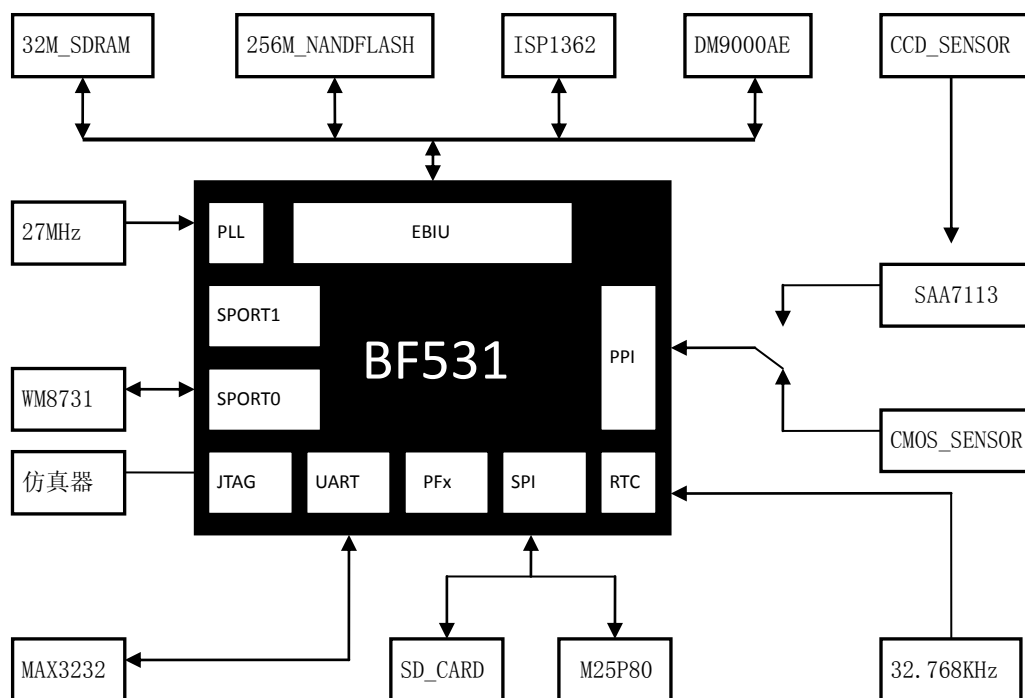
串口	1 个 UART 接口
视频接口	1 个 30 万像素 CMOS 数字视频输入接口 1 个 CVBS 模拟视频输入接口
实时时钟	1 个 RTC 时钟, 使用 CR2012 纽扣电池
电源	1 个 5V 电源插口, 1 个电源开关
其它扩展接口	JTAG 调试接口 UART 通信接口 SPORT1 音频扩展接口 SPI 接口, 可扩展 SPI 总线设备 PPI 接口, 可扩展支持 LCD、视频输入/输出接口

- CPU:  
ADSP-BF531  
主频: 400MHz  
16 位嵌入式处理器  
带有动态电源管理, 可根据功耗需要动态的改变内核的供电电压。
- 存储器:  
EEPROM: 1 片 M25P80, 1M Byte  
SDRAM: 1 片 SAMSUNG K4S561632H 32M Byte  
Nand Flash: 1 片 SAMSUNG K9F2G08U0A 256M Byte
- 视频:  
视频解码: SAA7113 通过 I2C 总线控制, 支持 ITU656 数字视频标准  
CMOS Sensor: OV7640 通过 I2C 总线进行控制
- 音频:  
WM8731 低功耗立体声音频 CODEC  
支持立体声回放, 单声道音频输入, 采样率最高达 96kHz  
音频输入: MIC  
音频输出: Head Phone
- USB HOST:  
飞利浦 ISP1362 USB 控制器, 符合 USB2.0 标准, 支持全速数据传输 (12Mbit/S) 和低速数据传输 (1.5Mbit/S)
- 以太网控制器:  
DM9000 快速以太网控制处理器, 10/100 M bit/S 以太网接口

- 串行接口：
  - 一个 UART 通信接口
- SD 接口：
  - 可外接 SD 存储卡
- 外部扩展接口：
  - JTAG 调试接口
  - SPORT1 音频扩展接口
  - SPI 接口，可扩展 SPI 总线设备
  - PPI 接口，可扩展支持 LCD、视频输入/输出接口
- 其它：
  - 1 个 RTC，一个电源开关，一个 Reset 按钮。

## 2.2 主要接口

LTDEV-BF53X 开发板系统框图：



主要接口介绍：

### 一、EBIU: EXTERNAL BUS INTERFACE UNIT (外部总线接口单元)

EBIU为外部存储器地址和外设地址提供无缝连接，并且提供PC100/PC133速率的SDRAM标准接口，同时能够很好的兼容SRAM, ROM, FIFOs, FLASH MEMORY, ASIC/FPGA等其他外部设备。

本开发板中SDRAM、NAND\_FLASH、USB(ISP1362)、网卡模块(DM9000AE)，都属于总线设备，都是连在EBIU上。他们占用同样的数据总线，其原理是通过访问不同的地址来分时操作。

## 二、PPI: PARALLEL PERIPHERAL INTERFACE (并行外设接口)

ADSP-BF53x处理器提供可直接与并行A/D和D/A转换器、视频编码和解码器以及其它通用外设连接的并行接口(PPI)。PPI包括一个专用时钟引脚，多达3个帧同步引脚和多达16个数据引脚。输入时钟支持fSCLK/2 MHz的并行数据传输率，同步信号可以被配置为输入或输出。

PPI 支持各种通用模式和ITU-R656 模式操作。在通用模式下，PPI 提供多达16 位数据的半双工、双向数据传输，并且提供了多达3 个帧同步信号。在ITU-R656 模式下，PPI 提供8 或10 位视频数据的半双工、双向传输。此外，片内还支持行启动和场启动同步包的解码。

本开发板中采用的双同步信号、8位PPI输入或输出的模式。输入可以由跳线来选择由CMOS SENSOR (OV7640) 或者CCD SENSOR(SAA7113+CCD视频信号)的输入。并且开发板上引出了相应的P14(PPI PORT)做为用户灵活使用。

## 三、SPI: SPI COMPATIBLE PORT (串行外设接口)

ADSP-BF53x 处理器有1 个SPI 兼容的端口，能够使控制器与多个SPI 兼容的设备通信。

SPI 接口使用3 个引脚传输数据：2 个数据引脚(主输出-从输入MOSI 和主输入-从输出MISO)和1 个时钟引脚(串行时钟SCK)；1 个SPI 片选输入引脚(SPISS)可使其它SPI 设备选择处理器；7 个SPI 片选输出引脚(SPISEL7-1)使处理器能够选择其它SPI 设备。这些SPI引脚也可以被重新配置为可编程标志引脚。通过这些引脚，SPI 端口提供了全双工的同步串行接口，支持主从模式和多主环境。

在开发板中，一共接了2个设备，一个是M25P80(SPI\_LASH)，另一个是SD\_CARD，他们通过不同的片选来区分。

## 四、SPORT: SERIAL PORT (串行口)

ADSP-BF53x 处理器提供2 个双通道同步串行端口(SPORT0 和SPORT1)来完成串行和多处理器的通信工作。支持I2S 功能。支持双向操作：每个SPORT 都有2 套独立的发送和接收引脚，支持I2S 立体声8通道。

开发板上SPORT0接到了WM8731(CODEC芯片)，实现了音频的输入和输出功能。SPORT1接到P3插座上，供用户使用。



## 五、UART: THE UNIVERSAL ASYNCHRONOUS RECEIVER/TRANSMITTER (串口)

ADSP-BF53x 处理器提供1 个全双工的通用异步接收/发送(UART)端口,它与PC 标准的UART 完全兼容。UART端口为其它外设或主机提供了一个简化的UART 接口,支持全双工、有DMA 能力的异步串行数据传输。UART 端口支持5 至8 个数据位、1 或2 个停止位以及无校验、奇校验、偶校验位。

开发板中的UART接口经过MAX3232转换芯片接出至P2,以供用户调试使用。

## 六、PFx: PROGRAMMABLE FLAGS (可编程通用I/O端口)

ADSP-BF53x 处理器有16 个双向的通用可编程I/O 引脚(PF15-0)。每一个可编程引脚都能通过操作相应的标志控制寄存器、标志状态寄存器和标志中断寄存器被独立控制,可以当作普通的I/O口来用,也可以灵活的当作中断引脚来用,通过设置能够实现中断的高低电平触发和上下边沿触发。

开发板上的PF口分配已经外部中断的使用在后面有具体的说明。

## 2.3 内存及中断分配

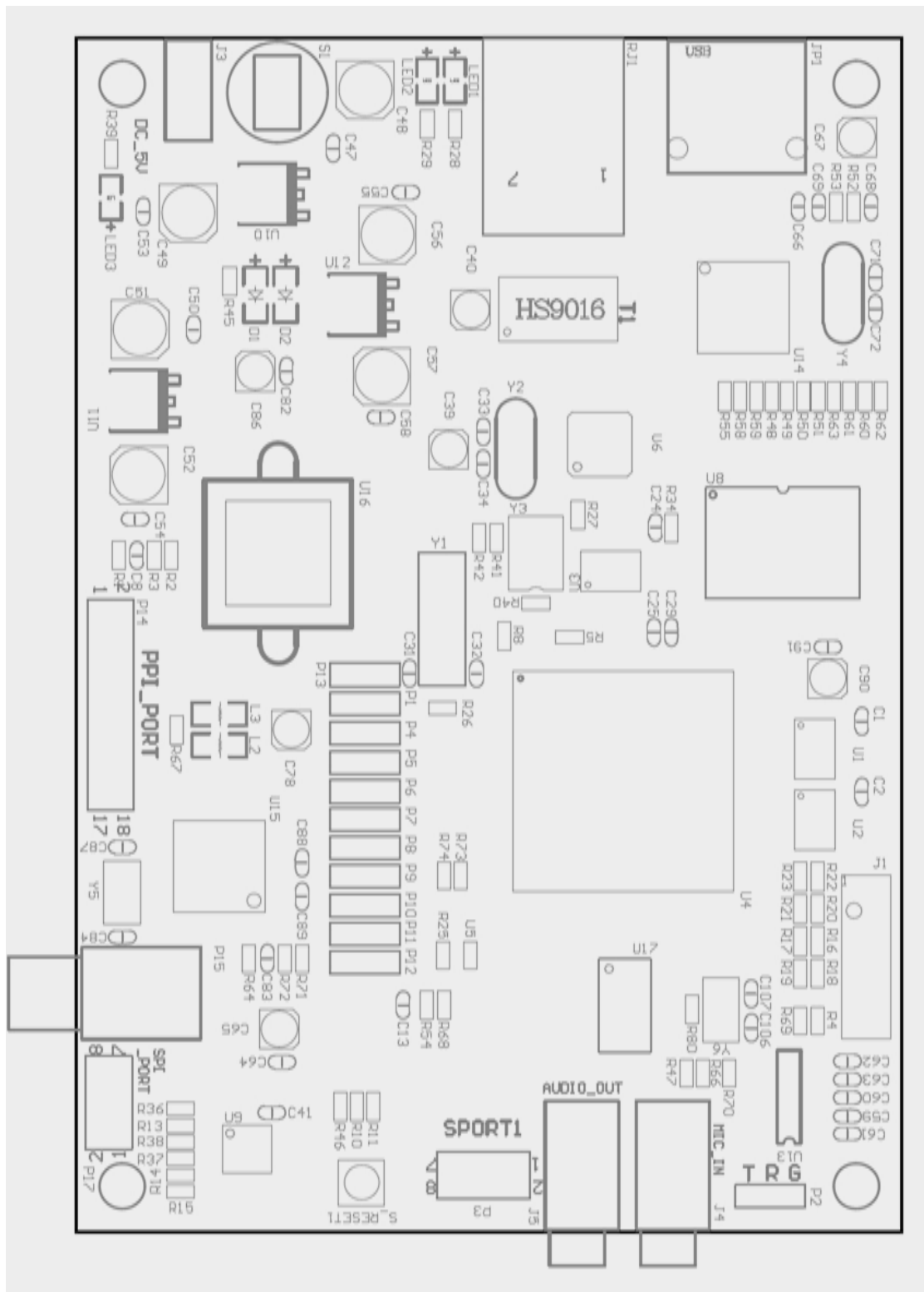
### 一、531 开发板内存映射表:

0x0000,0000 – 0x01FF,FFFF	<b>32M SDRAM</b>
0x0800,0000 – 0x1FFF,FFFF	RESERVED
0x2000,0000 – 0x200F,FFFF	<b>NANDFLASH</b>
0x2010,0000 – 0x201F,FFFF	RESERVED
0x2020,0000 – 0x202F,FFFF	<b>ISP1362(USB)</b>
0x2030,0000 – 0x203F,FFFF	<b>DM9000AE(LAN)</b>
0x2040,0000 – 0xFF7F,FFFF	RESERVED
0xFF80,0000 – 0xFF80,3FFF	RESERVED
0xFF80,4000 – 0xFF80,7FFF	DATA BANK A SRAM/CACHE
0xFF80,8000 – 0xFF90,3FFF	RESERVED
0xFF90,4000 – 0xFF90,7FFF	DATA BANK B SRAM/CACHE
0xFF90,8000 – 0xFF9F,FFFF	RESERVED
0xFFA0,0000 – 0xFFA0,7FFF	RESERVED
0xFFA0,8000 – 0xFFA0,FFFF	INSTRUCTION SRAM
0xFFA1,0000 – 0xFFA1,3FFF	INSTRUCTION SRAM/CACHE
0xFFA1,4000 – 0xFFAF,FFFF	RESERVED
0xFFB0,0000 – 0xFFB0,0FFF	SCRATCHPAD SRAM
0xFFB0,1000 – 0xFFBF,FFFF	RESERVED
0xFFC0,0000 – 0xFFDF,FFFF	SYSTEM MMR
0xFFE0,0000 – 0xFFFF,FFFF	CORE MMR

## 二、PF 以及外部中断分配

BF531 PROGRAMMABLE FLAGS	分配及用途
PF0	模拟 I2C 总线的 SDA 线
PF1	模拟 I2C 总线的 SCL 线
PF2	SPI_FLASH (M25P80) 的片选
PF3	PPI_FS3,PPI 在只使用行场同步模式下, 该脚接地
PF4	SD 卡插入中断
PF5	SD 卡写保护检测
PF6	NAND_FLASH 的 BUSY 信号
PF7	SD 卡片选
PF8	预留口, 未使用
PF9	预留口, 未使用
PF10	网络模块 DM9000AE 的中断信号
PF11	USB 模块 ISP1362 的中断信号
PF12	PPI 模式下的 PPI7
PF13	PPI 模式下的 PPI6
PF14	PPI 模式下的 PPI5
PF15	PPI 模式下的 PPI4

开发板 PCB 布局如下:



开发板 PCB 布局

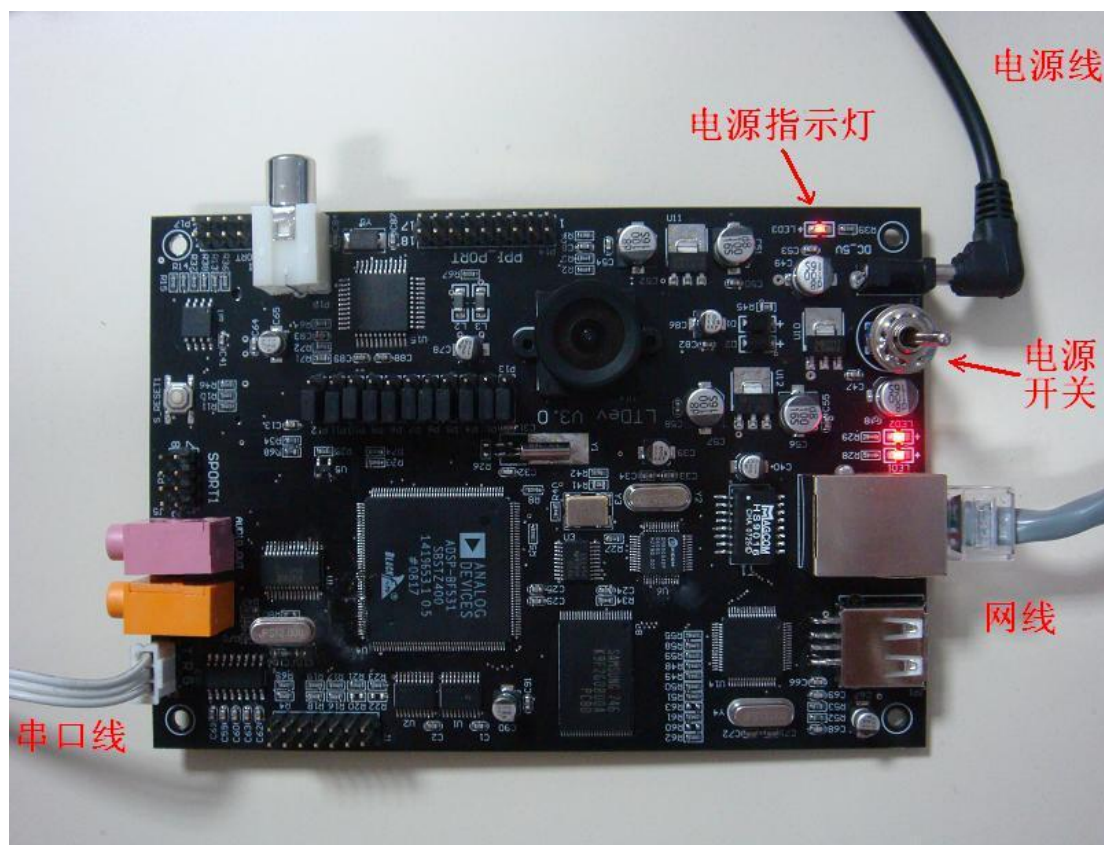
## 3 开发环境搭建

### 3.1 硬件环境搭建

开发板连接步骤：

1. 连接串口线，如果 PC 机没有串口，需要使用串口转 USB 接口线。注意，如果串口线接反，终端程序上将显示乱码。
2. 连接开发板网线。
3. 接通电源，打开电源开关，这时电源指示灯应该亮起。

如图所示：



开发板串口、网线、电源连接示意图

### 3.2 软件环境搭建

#### 3.2.1 建立交叉编译环境

首先编译环境需要安装宿主机(PC机)操作系统，推荐使用 SUSE Linux 10.x。

交叉编译(Crossing Compile)是相对本机编译(Native Compile)而言,指在异种 CPU 上进行的编译。比如在 x86 机器上编译运行于 DSP 处理器上的程序就是交叉编译。LINUX 环境下使用 GNU 工具完成编译、链接等过程,包括编译器 GCC、运行库 GLIBC 和实用工具 BINUTILS。

光盘中 toolchains 目录下包括交叉编译工具,安装过程如下:

- 1) 拷贝 blackfin-toolchain-uclibc-full-08r1-8.i386.rpm 和 blackfin-toolchain-08r1-8.i386.rpm 两个文件到 linux 系统中
- 2) 执行:

```
rpm -Uvh blackfin-toolchain-08r1-8.i386.rpm  
rpm -Uvh blackfin-toolchain-uclibc-full-08r1-8.i386.rpm
```

执行以上命令后,交叉编译工具将被安装在以下目录

/opt/uClinux/bin-uclinux/bin/

/opt/uClinux/bfin-linux-uclibc/bin/

- 3) 设置编译器路径:

```
export PATH=/opt/uClinux/bfin-uclinux/bin:  
/opt/uClinux/bfin-linux-uclibc/bin:$PATH
```

这样系统会自动找到 gcc 工具 bfin-uclinux-gcc 等。用户也可以把设置路径的命令放在系统启动脚本中,这样就不需要每次运行都要设置路径。

执行完以上步骤后,就可以使用交叉编译器 bfin-uclinux-gcc、运行库和以“bfin-uclinux-”开头的以及以“bfin-linux-uclibc-”开头的各种工具。

工具链也可在 ADI 公司网站上下载: <http://blackfin.uclinux.org/>

### 3.2.2 建立 TFTP 服务器

为了能够使用网络在 U-Boot 环境中下载 uClinux 内核,需要在 PC 机上面建立 TFTP 服务器。可以使用 tftpd32 或者 turbo tftp 等 TFTP 工具。

## 4 实验及测试流程

### 4.1 U-BOOT 启动

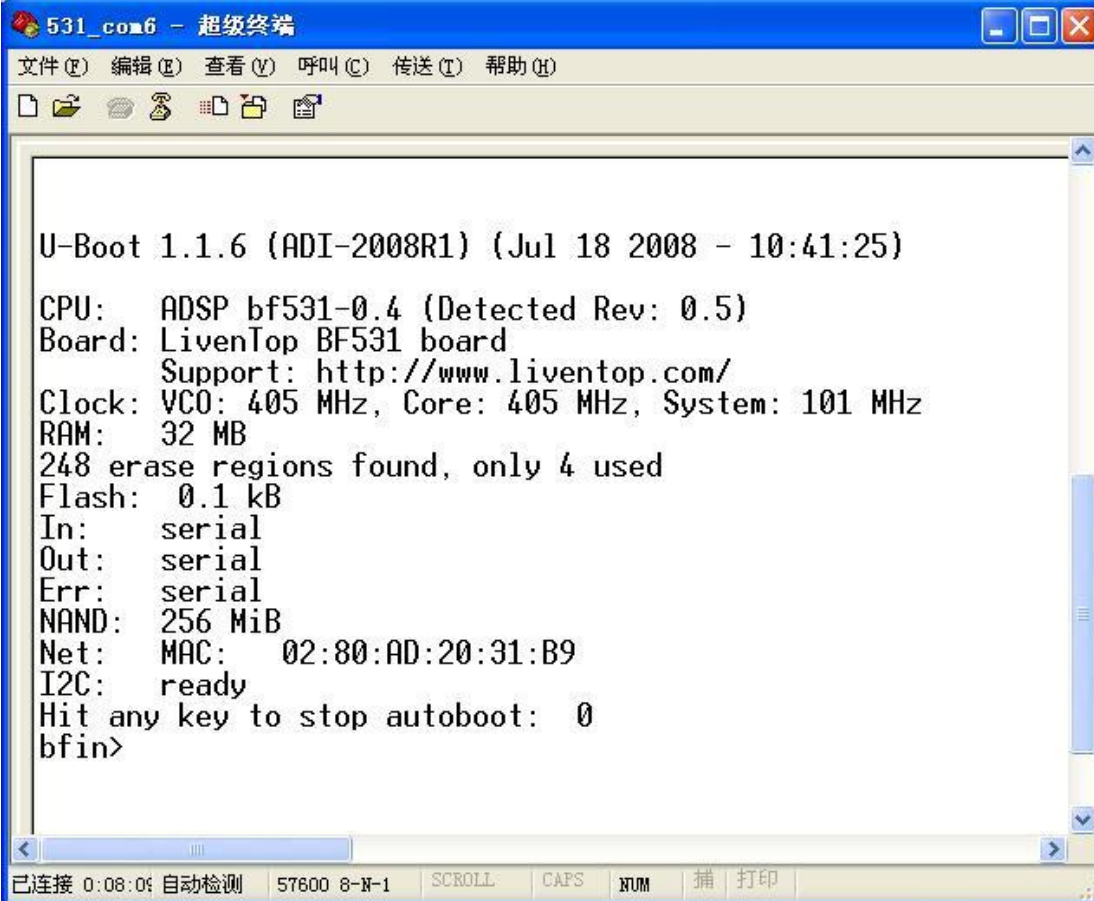
1. 配置 UART 串口控制终端

打开 PC 机超级终端程序,设置为 57600-8-1,即串口波特率为 57600,数据长度 8,停止位 1,无奇偶校验,无流控制。设置如下图所示:



#### COM 连接属性配置

连接好开发板串口和电源，打开电源开关，U-Boot 将运行，运行结果如图所示：



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
U-Boot 1.1.6 (ADI-2008R1) (Jul 18 2008 - 10:41:25)
CPU:  ADSP bf531-0.4 (Detected Rev: 0.5)
Board: LivenTop BF531 board
      Support: http://www.liventop.com/
Clock: VCO: 405 MHz, Core: 405 MHz, System: 101 MHz
RAM:   32 MB
248 erase regions found, only 4 used
Flash: 0.1 kB
In:    serial
Out:   serial
Err:   serial
NAND:  256 MiB
Net:   MAC: 02:80:AD:20:31:B9
I2C:   ready
Hit any key to stop autoboot: 0
bfin>
```

U-Boot 启动

U-Boot 检测系统信息并显示 CPU 型号、开发板型号、CPU 运行时钟、存储器大小、以太网控制器等信息，然后等待用户操作，如果用户在 3 秒内没有键盘按下，则自动启动 uClinux 操作系统。如果按下任意键，则进入 U-Boot 命令行。

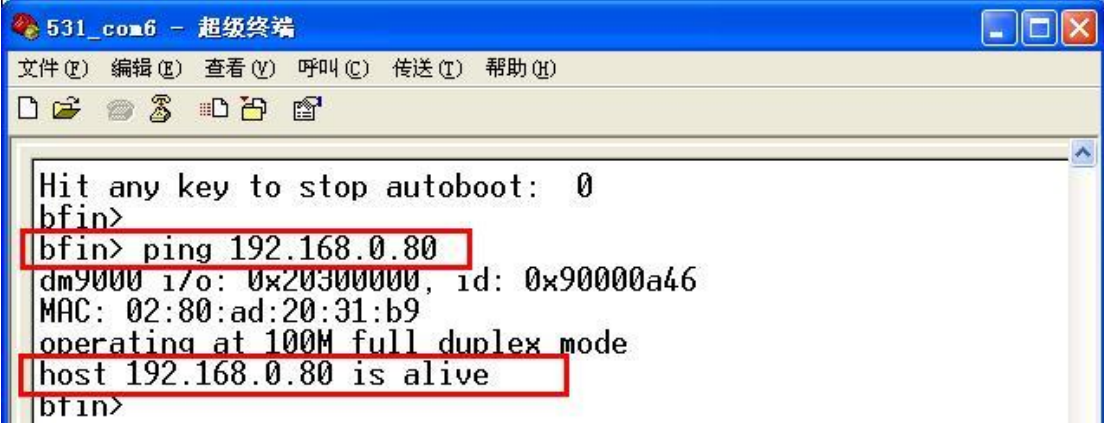
## 4.2 在 U-Boot 下测试网络

我们已经在 U-Boot 中移植了 DM9000 以太网控制器的驱动程序，因此可以通过以太网下载用户程序。在下载程序前最好测试一下网络接口是否连接好，测试方法如下：

- 使用随机配备的交叉网线连接开发板和 PC 机
- 设置 PC 机的 IP 地址 192.168.0.80，当然也可以是别的。
- 开发板上电，在五秒内按下任意键进入 U-Boot 菜单。
- 输入以下命令（注意，在 ping 之前请将 PC 机的防火墙关闭）：

```
ping 192.168.0.80
```

结果显示“host 192.168.0.80 is alive”表明网络接口工作正常，如下图所示：



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
Hit any key to stop autoboot: 0
bfin>
bfin> ping 192.168.0.80
dm9000 170: 0x20300000, id: 0x90000a46
MAC: 02:80:ad:20:31:b9
operating at 100M full duplex mode
host 192.168.0.80 is alive
bfin>
```

另外 U-Boot 支持很多命令操作，比如 TFTP 下载、烧写 Flash、设置环境变量等，用户可以键入“help”命令查看详细的使用方法。

## 4.3 烧写内核

要通过网络烧写内核，首先需要将内核镜像文件拷贝到 PC 端的 TFTP 服务器上，并在开发板上配置好服务器 IP 地址，如果已经配置过了，则这一步可以省略。

### 1. 配置网络

在开发板的 U-boot 命令行模式下输入：

```
setenv serverip 192.168.0.80
setenv ipaddr 192.168.0.50
setenv gatewayip 192.168.0.1
saveenv
```

其中，serverip 即 PC 机 IP 地址。ipaddr 即开发板 IP 地址，gatewayip 为网关 IP 地址。

### 2. 下载内核，并烧写内核到 nand flash 上

将 uImage 内核镜像放在 TFTP 服务器上，在超级终端里执行：

```
tftp 1000000 uImage
```

会把 TFTP 服务器上的 uClinux 内核读取到内存 0x1000000 地址，如下图所示：



```

531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
I2C: ready
Hit any key to stop autoboot: 0
bfin> tftp 1000000 uImage
dm9000 i/o: 0x20300000, id: 0x90000a46
MAC: 02:80:ad:20:31:b9
operating at 100M full duplex mode
TFTP from server 192.168.0.80; our IP address is 192.168.0.50
Filename 'uImage'.
Load address: 0x1000000
Loading: #####
done
Bytes transferred = 2905528 (2c55b8 hex)
bfin> _

```

接下来执行：

```
nand write.jffs2 1000000 0 400000
```

将把 0x1000000 地址的 uClinux 内核烧写到开发板的 nand flash 上。如图  
图所示：

```

done
Bytes transferred = 2905528 (2c55b8 hex)
bfin> nand write.jffs2 1000000 0 400000
NAND write: device 0 offset 0x0, size 0x400000
Writing data at 0x3ff800 -- 100% complete.
4194304 bytes written: OK
bfin>

```

### 3. 设置 U-Boot 启动参数

如果已经设置过启动参数，则这一步也可以省略。

执行：

```
setenv bootcmd nand read.jffs2 1000000 0 400000 \;bootm
save
```

至此 uClinux 系统已经烧写到开发板 Nand Flash 中，关闭开发板电源然后重新上电，将自动读取 nand flash 上的内核，并启动 uClinux。

## 4.4 在 uClinux 下测试外设

### 【设置系统时钟 RTC】

用 date 命令设置时间，用 hwclock 命令将时间写入 RTC。例如，要设置 RTC 时钟为 2008 年 8 月 6 日 18 点 47 分 34 秒，需要执行：

```
date 080618472008.34
```

```
hwclock -w
```

### 【USB 操作】

插入 U 盘，系统找到 U 盘并显示 U 盘信息，然后输入以下命令挂载 U 盘：

```
mkdir /mnt/udisk
```

```
mount -t vfat /dev/sda /mnt/udisk
```

如图所示：



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/> usb 1-2: new full speed USB device using isp1362-hcd a
usb 1-2: Product: USB MASS STORAGE CLASS
usb 1-2: SerialNumber: USB MASS STORAGE CLASS
usb 1-2: configuration #1 chosen from 1 choice
scsi4 : SCSI emulation for USB Mass Storage devices
scsi scan: INQUIRY result too short (5), using 36
scsi 4:0:0:0: Direct-Access    GENERIC  USB DISK DEVICE  1.00

sd 4:0:0:0: [sdal] 990705 512-byte hardware sectors (507 MB)
sd 4:0:0:0: [sdal] Write Protect is off
sd 4:0:0:0: [sdal] Assuming drive cache: write through
sd 4:0:0:0: [sdal] 990705 512-byte hardware sectors (507 MB)
sd 4:0:0:0: [sdal] Write Protect is off
sd 4:0:0:0: [sdal] Assuming drive cache: write through
sda:<7>usb-storage: queuecommand called
unknown partition table
sd 4:0:0:0: [sdal] Attached SCSI removable disk

root:/> mkdir /mnt/udisk
root:/> mount -t vfat /dev/sda /mnt/udisk
root:/> cd /mnt/udisk/
root:/mnt/udisk> ls
```

### 【Nand Flash 操作】

uClinux 内核启动后 nand flash 会自动挂载到/mnt/flash 目录下，操作/mnt/flash 目录即可。

使用 df 命令可以查看 nand flash 使用状况。

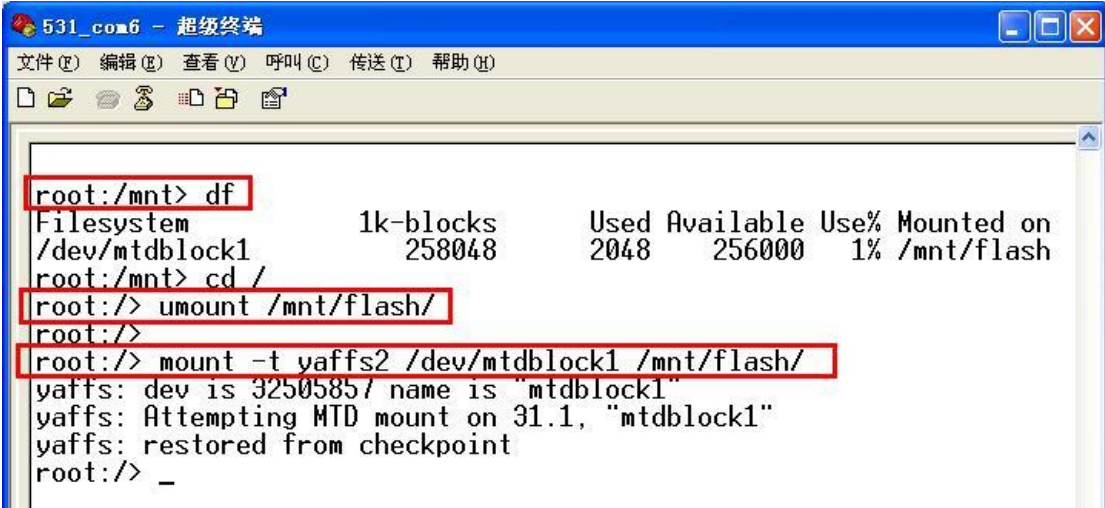
输入以下命令可以手动解除挂载 nand flash: (注意, 不能在/mnt/flash 目录下执行该命令)

```
umount /mnt/flash
```

输入以下命令可以手动挂载 nand flash:

```
mount -t yaffs2 /dev/mtdblock1 /mnt/flash
```

以上操作如下图所示:



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/mnt> df
Filesystem          1k-blocks      Used Available Use% Mounted on
/dev/mtdblock1      258048         2048    256000    1% /mnt/flash
root:/mnt> cd /
root:/> umount /mnt/flash/
root:/>
root:/> mount -t yaffs2 /dev/mtdblock1 /mnt/flash/
yaffs: dev is 32505857 name is "mtdblock1"
yaffs: Attempting MTD mount on 31.1, "mtdblock1"
yaffs: restored from checkpoint
root:/> _
```

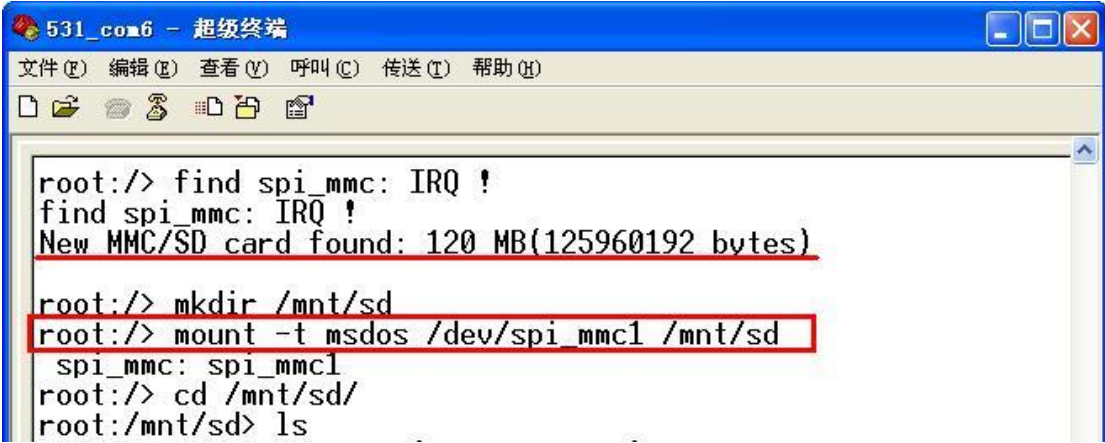
### 【SD 卡操作】

插入 SD 卡, 系统提示找到 SD 卡, 然后输入以下命令挂载 SD 卡:

```
mkdir /mnt/sd
```

```
mount -t msdos /dev/spi_mmc1 /mnt/sd
```

如下图所示:



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/> find spi_mmc: IRQ !
find spi_mmc: IRQ !
New MMC/SD card found: 120 MB(125960192 bytes)
root:/> mkdir /mnt/sd
root:/> mount -t msdos /dev/spi_mmc1 /mnt/sd
spi_mmc: spi_mmc1
root:/> cd /mnt/sd/
root:/mnt/sd> ls
```

### 【网络】

◆ 用 DHCP 配置网络:

```
dhcpcd eth0 -t 0&
```

◆ 手动配置网络:

设置网络 IP 地址及掩码:

```
ifconfig eth0 192.168.0.50 netmask 255.255.255.0 up
```

设置默认网关:

```
route add default gw 192.168.0.1
```

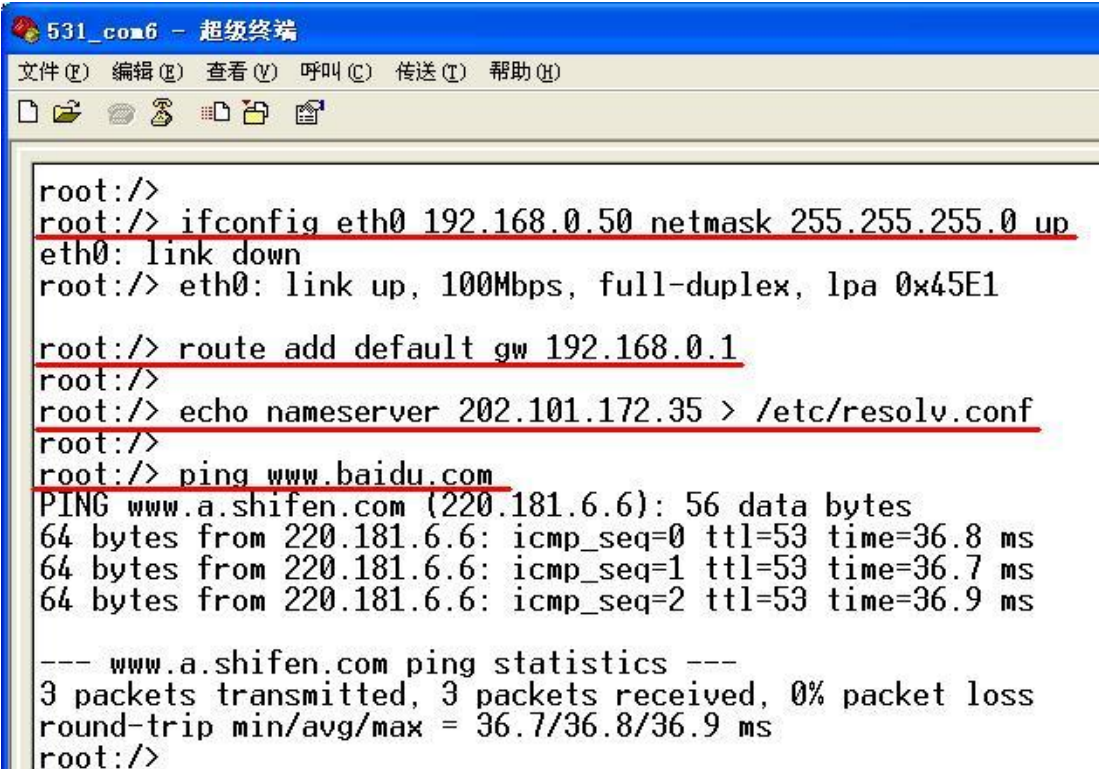
设置 DNS 服务器:

```
echo nameserver 202.101.172.35 > /etc/resolv.conf
```

使用 ping 命令测试 DNS:

```
ping www.baidu.com
```

如图所示:



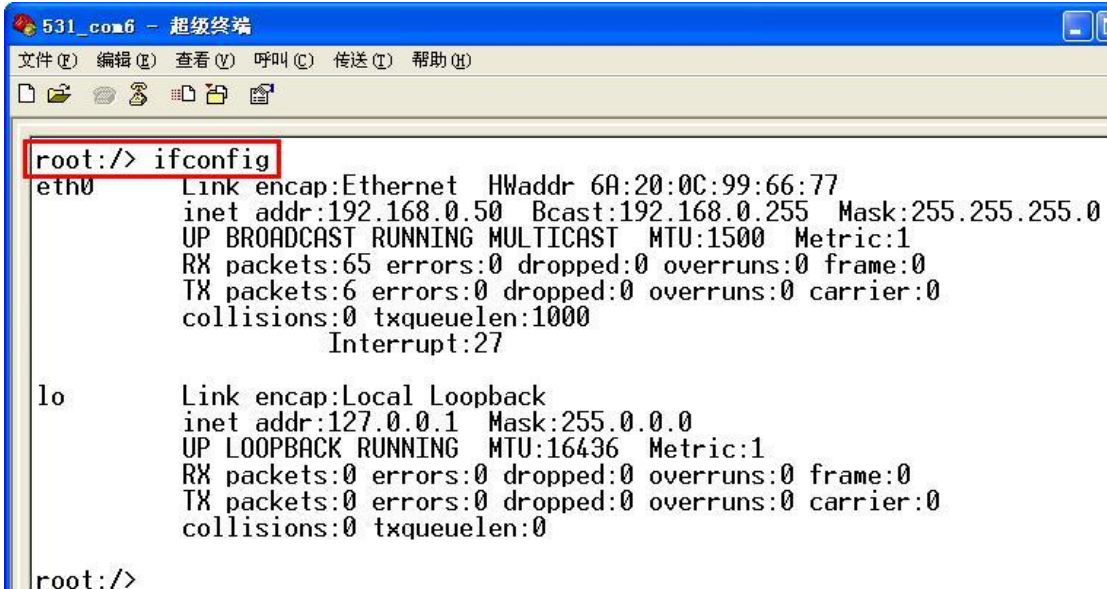
```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/>
root:/> ifconfig eth0 192.168.0.50 netmask 255.255.255.0 up
eth0: link down
root:/> eth0: link up, 100Mbps, full-duplex, lpa 0x45E1
root:/> route add default gw 192.168.0.1
root:/>
root:/> echo nameserver 202.101.172.35 > /etc/resolv.conf
root:/>
root:/> ping www.baidu.com
PING www.a.shifen.com (220.181.6.6): 56 data bytes
64 bytes from 220.181.6.6: icmp_seq=0 ttl=53 time=36.8 ms
64 bytes from 220.181.6.6: icmp_seq=1 ttl=53 time=36.7 ms
64 bytes from 220.181.6.6: icmp_seq=2 ttl=53 time=36.9 ms

--- www.a.shifen.com ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 36.7/36.8/36.9 ms
root:/>
```

可使用 ifconfig 命令查看网络信息:

```
ifconfig
```

如图所示:



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root: /> ifconfig
eth0      Link encap:Ethernet  HWaddr 6A:20:0C:99:66:77
          inet addr:192.168.0.50  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:65 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
             Interrupt:27

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  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:0

root: />
```

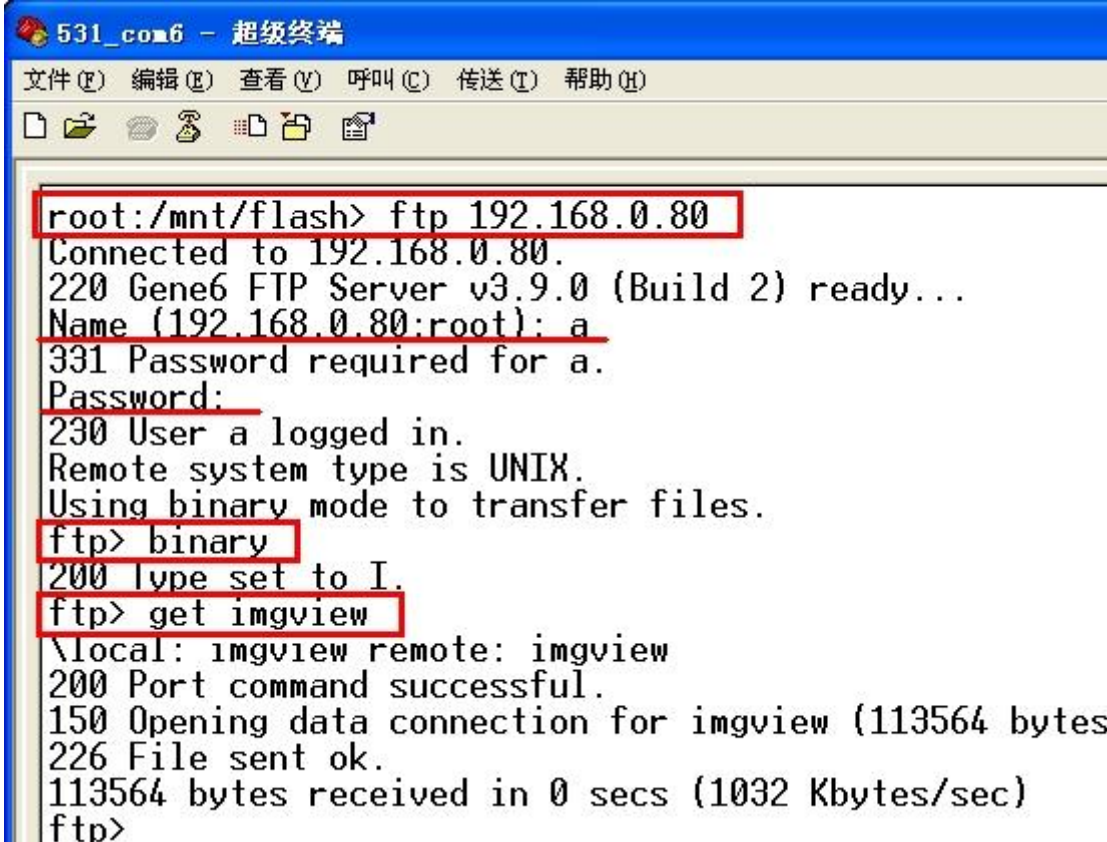
另外还可使用以下命令从 ftp 下载文件:

```
ftpget -u a -p a 192.168.0.80 local-file remote-file
```

或者:

```
ftp 192.168.0.80
```

并根据提示输入用户名和密码, 然后可以使用 get 命令下载文件, put 命令上传文件。如图所示:



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/mnt/flash> ftp 192.168.0.80
Connected to 192.168.0.80.
220 Gene6 FTP Server v3.9.0 (Build 2) ready...
Name (192.168.0.80:root): a
331 Password required for a.
Password:
230 User a logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> binary
200 lype set to I.
ftp> get imgview
\local: imgview remote: imgview
200 Port command successful.
150 Opening data connection for imgview (113564 bytes)
226 File sent ok.
113564 bytes received in 0 secs (1032 Kbytes/sec)
ftp>
```

注意, 如果下载二进制文件(非文本文件, 如可执行程序、mp3、jpeg 等), 需要用 binary 命令设置为二进制传输模式。

**【音频】**

播放 mp3 文件:

```
mp3play test.mp3
```

播放 wav 文件:

```
play -srw -f 48000 48k.wav
```

如图所示:



```
531_com6 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)
root:/mnt/flash> mp3play Sample.mp3
Sample.mp3: MPEG1-III (365824 ms)
root:/mnt/flash>
root:/mnt/flash> play -srw -f 48000 48k.wav
PLAYING at: freq=48000 chans=2 bitspersample=16
root:/mnt/flash>
```

开发板支持 8K,32k,48k,96k 音频格式,其它格式也可以播放,但播放速度可能与原始音频不同。

音量调节,执行:

```
echo 100 > /proc/asound/wm8731L/wm8731_vol
```

注意,音量最大为 100,最小为 0。

测试录音功能,执行:

```
audio_test
```

该程序将 mic 录音延时后播放出来。另外也可使用 vrec 程序进行录音:

```
vrec -rw -t 10 -s 48000 -b 16 test.wav
```

**【视频图像采集】**

本开发板可以使用 OV7640 的 CMOS 摄像头或者 SAA7113 作为视频输入,但是二者不能同时使用。

使用开发板自带的 CMOS 摄像头(默认):

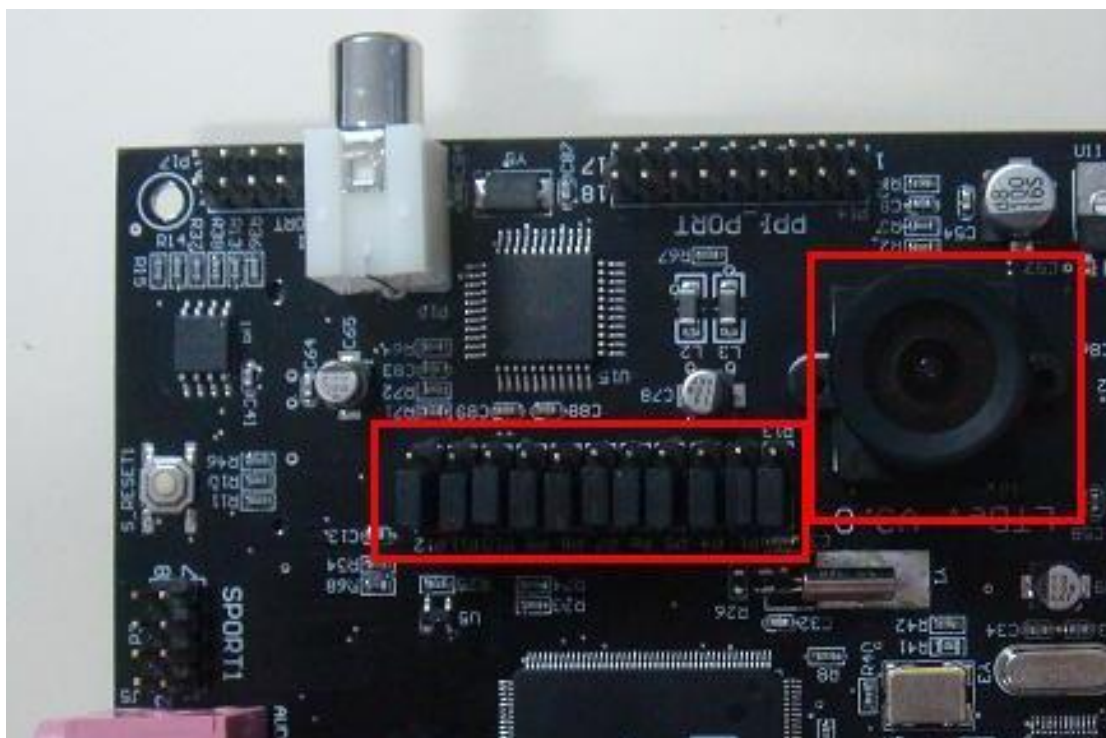
- 1) 将开发板上的跳线设置为“板载 CMOS 摄像头输入”(如下图所示)。
- 2) 将 PC 机的 IP 设置为 192.168.0.80,开发板 IP 设置为 192.168.0.50。  
注意:这两个 IP 可根据需要改动。
- 3) 在开发板上执行光盘 video/camera/目录下的测试程序 imgview,(后面的参数是 PC 机的 IP 地址):

```
imgview 192.168.0.80
```

- 4) 在 PC 上执行光盘 video/PCserver/目录下的 PCserver.exe。

然后就能在 PC 上看到视频图像。

跳线设置如下图所示:



跳线设置：板载 CMOS 摄像头输入

如果要使用外接摄像头，需要：

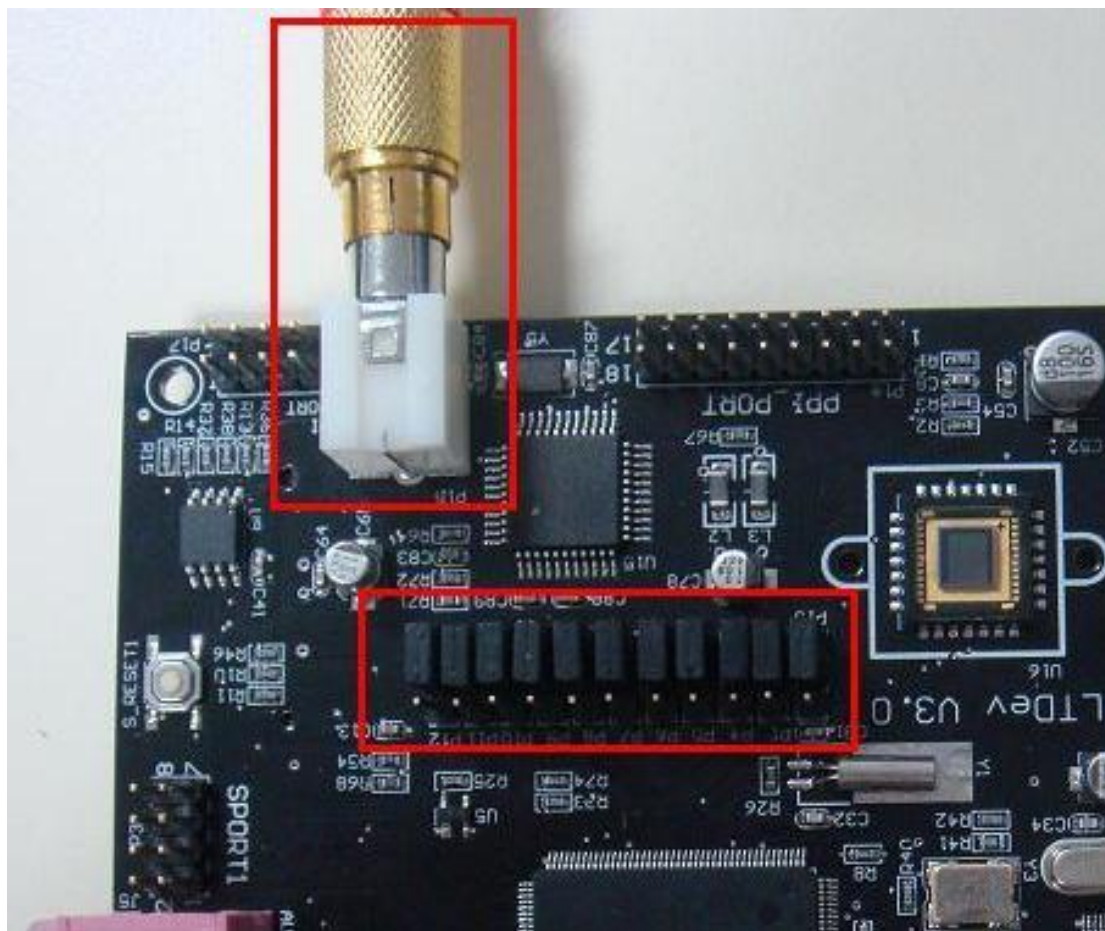
- 5) 将开发板上的跳线设置为“外接 CVBS 模拟视频输入”(如下图所示)。
- 6) 将摄像头的视频信号线连接到开发板上
- 7) 下载内核：uImage\_CVBS
- 8) 将 PC 机的 IP 设置为 192.168.0.80，开发板 IP 设置为 192.168.0.50。
1. 在开发板上执行光盘 video/目录下的测试程序：imgview（后面的参数是 PC 机的 IP 地址）：

```
imgview 192.168.0.80
```

- 9) 在 PC 上执行光盘 video/目录下的 PCserver.exe。

然后就能在 PC 上看到视频图像。

跳线设置如下图所示：

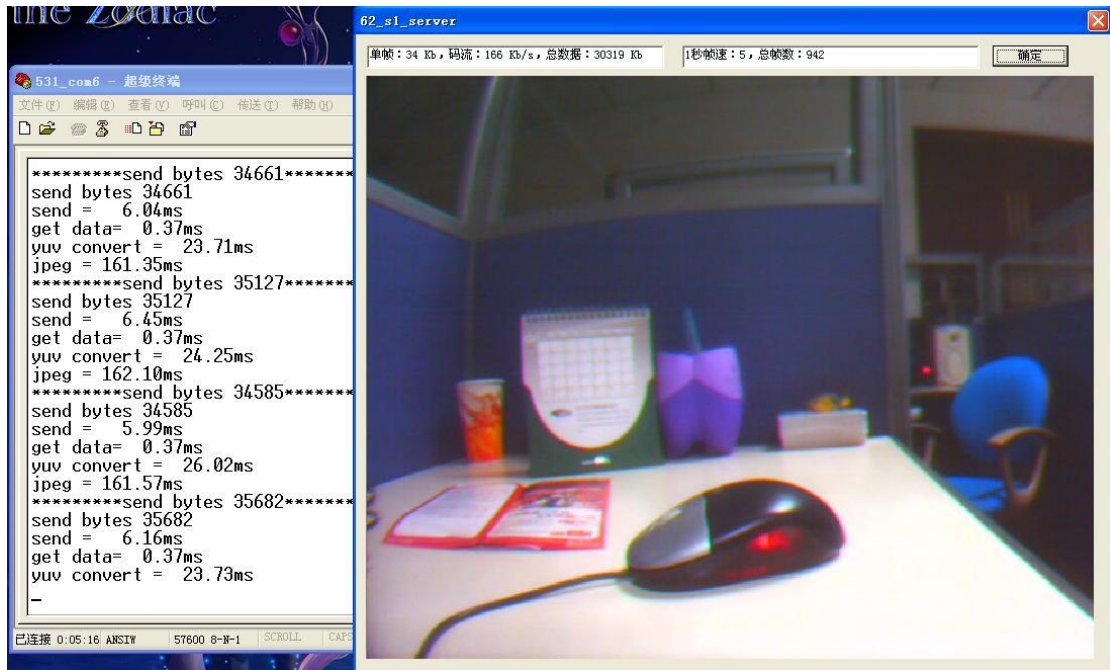


跳线设置：外接 CVBS 摄像头输入

说明：本程序目的是为用户演示如何采集图像数据，不是商业化程序。用户可以在该程序基础上开发自己的商业化应用。

视频获取效果如下图所示：





## 5 软件及编程方法

以下是一个 hello world 程序 hello.c 的源代码：

```
#include <stdio.h>

int main() {
    printf("Hello, World\n");
    return 0;
}
```

编译该例子：

```
bfm-uclinux-gcc -Wl,-elf2flt hello.c -o hello
```

生成 hello 可执行文件，在开发板上执行

```
chmod +x ./hello
```

```
./hello
```

将显示：

```
Hello, World
```

关于在 uClinux 下软件编程的详细帮助信息，请参考 ADI 网站：

<http://docs.blackfin.uclinux.org/doku.php>

## 6 光盘内容

光盘目录树:

LTDEV-BF53x

|---- bootloader

|---- documents

|---- uClinux

|---- toolchain

|---- video

|---- audio

各部分说明如下:

**bootloader** — U-Boot 二进制映像文件

**documents** — LTDEV-BF53x 模块的用户指南、PCB 布局、以及芯片 datasheet 资料

**uClinux** — uClinux 内核，包括源代码和一个编译好的映像文件：**uImage**

**toolchain** — 建立交叉编译环境所需要的开发工具链

**video** — 摄像头示例程序，包括开发板上的应用程序 **imgview**，PC 端的监控程序 **PCserver.exe** 及相应的源代码，可以实现 **motion JPEG** 压缩、传输和显示。

**audio** — 音频示例程序，包括 **mp3** 播放工具：**mp3play** 和录音测试程序：**audio\_test** 及相应的源代码。