



立宇泰电子

Liyutai Elec. CO., Ltd.

专注于做最好的嵌入式计算机系统供应商

NARI2410MU定制板

使用说明

(适用于 A09S2410MU V3.3)

Rev 0.1

2007年11月16日

杭州立宇泰电子有限公司

HangZhou LiYuTai Elec.Co.,Ltd

目 录

1 产品简介	3
2 硬件规格	3
2.1 核心板.....	3
2.2 扩展板.....	3
2.2.1 串行接口.....	4
2.2.2 以太网接口.....	8
2.2.3 LCD、电源指示灯、运行指示灯接口（J9）	8
2.2.4 电源、触摸屏、RS485 接口（J10）	10
2.2.5 SD卡接口.....	11
2.2.6 复位电路-RESET	11
2.2.7 蜂鸣器-BUZZER.....	11
3 软件支持	11
3.1 文件说明.....	11
3.2 准备开始.....	11
3.2.1 通过telnet登陆	11
3.2.2 启动脚本.....	12
3.2.3 通过telnet登陆	13
3.3 用户开发应用程序.....	13
3.3.1 安装交叉编译器.....	14
3.3.2 编写Hello代码.....	14
3.3.2 编译hello.....	14
3.3.3 采用FTP方式.....	14
3.4.应用程序实例.....	15
3.4.1 串口通讯实例.....	15
3.4.2 看门狗.....	16
3.4.3 蜂鸣器.....	16
3.4.4LED状态指示灯	17
4 合作方式和定制服务.....	17
4.1 合作方式.....	17
4.2 定制服务.....	18

1 产品简介

NARI2410MU 定制板具备 9 个可独立控制的串行口，2 路以太网接口，采用多任务 linux 操作系统管理这些接口；高速逻辑器件实现无延迟 RS485 半双工控制。系统能够在满负荷工作情况下实时响应每一个终端设备的数据请求，特别适合作为实现串口终端设备集中管理、实时数据采集的服务器主板。

NARI2410MU 定制板还提供了内置看门狗，数字真彩液晶屏接口，SD 卡接口，日历时钟，蜂鸣器，状态指示灯。结构上采用双层构架：ARMSYS2410-CORE 核心板加扩展板；它的出线、插座、板型结构都符合定制方的样板要求。

NARI2410MU 定制板特点是高性能、低功耗、低成本、使用方便、节省产品的研发时间和成本。该产品预装了 Linux2.4 系统，并随样机提供二次开发所需要各种资源和最专业的技术服务，用户不必关心硬件和驱动问题即可快速地将它应用到专业领域中去。

2 硬件规格

2.1 核心板

核心系统采用 6 层板工艺，具有良好的电气抗干扰性能，采用 SO-DIMM200 插座与扩展板连接。核心板的资源配置如下：

- **CPU:** 采用三星 S3C2410A 低功耗 32 位处理器，主频 200MHz;
- **SDRAM:** 64M Byte 内存;
- **NANDFlash:** 64M Byte 存储空间;

2.2 扩展板

扩展板采用 4 层板工艺。尺寸结构、引出接口完全按照定制要求设计。下面分别描述这些接口。

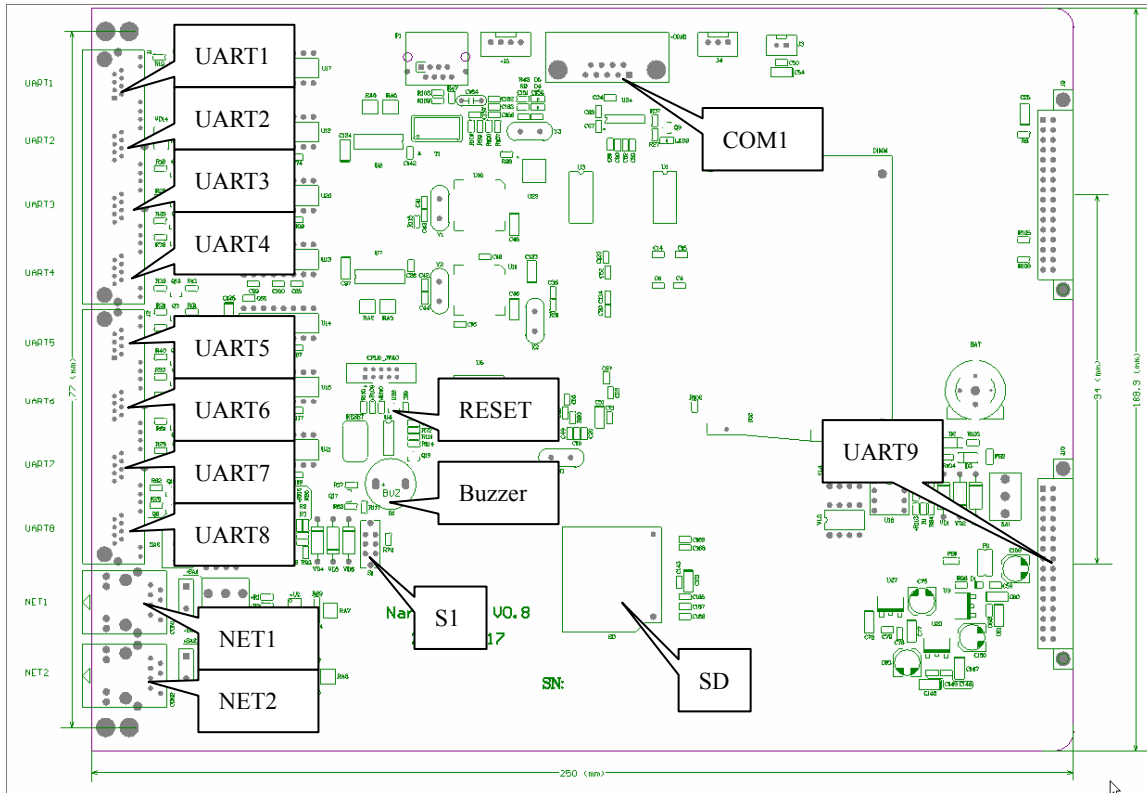


图 1: 扩展板上接口和电路布局

2.2.1 串行接口

Nari2410MU 定制板一共提供了 10 路串行口，它们包括：

- J1 和 J2 两个 4 联 RJ45 座引出了 8 个串行口 (UART1~UART8)。这些串口被分为 3 组：
 - **UART1, UART2**: 是提供 8 根信号线的 RS232 串行口；
 - **UART3, UART4, UART5, UART6, UART7**: 是提供 3 根信号线的 RS232 串行口；
 - **UART8**: 可以用拨动开关来切换 RS232 或 RS485 工作模式的串口，在 RS232 模式下 3 根信号线，在 RS485 模式下 2 根信号线，同时在对应的 RJ45 座上提供+5V 和 GND 电源线。
- J10 欧品插座上还引出了 1 路 RS485 串行口 **UART9**。
- COM1 口作为控制台串口，主要用于系统调试。

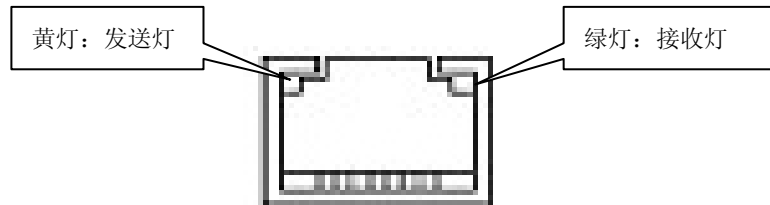
下面来描述这 10 路串口的特性参数：

2.2.1.1 UART1~UART8 共用特性参数

J1, J2上引出了互为独立的8路串行接口，它们都同时满足以下特性参数：

- 硬件：采用串行口控制器16C554；
- IRQ 号：1 (EINT1)；
- I/O物理地址：0x1000-0000~0x1000-003F；

- FIFO: 16 byte 硬件FIFO;
- 波特率: 目前支持4种波特率: 9600, 14400, 19200, 28800bps;
- 数据位: 5,6,7,8 ;
- 停止位: 1,2 ;
- 奇偶校验位: even, odd, none, mark, space;
- 保护功能: 15KVESD保护, 过压冲击保护;
- **Linux系统下设备名: ttyLT0~ttyLT7 (分别对应UART1~UART8);**
- 引脚排列及指示灯:



引脚排列: 8 7 6 5 4 3 2 1

当端口上数据有发送或接收时, 对应指示灯会点亮(闪烁), 当没有数据传输时两盏指示灯熄灭。

2.2.1.2 UART1, UART2 特性参数

UART1 和 UART2 是提供

- 流向控制: 支持无流控, 支持硬件流控;
- **设备名: ttyLT0, ttyLT1 (分别对应UART0, UART1);**
- 每一路信号线包括: DCD, RXD, TXD, DTR, GND, CTS, RTS, DSR共8路信号;
- 传输模式: RS-232全双工;
- 信号类型: 标准RS-232电平。
- 引脚定义:

UART1, UART2的引脚定义 (RJ45)	
1	DCD
2	RXD
3	TXD
4	DTR
5	GND
6	CTS
7	RTS
8	DSR

2.2.1.3 UART3~UART7 特性参数

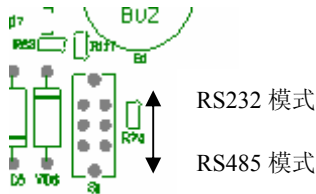
- 流向控制: 支持无流控;

- **设备名：ttyLT2~ttyLT6（分别对应UART3~UART7）；**
- 每一路信号线包括：RXD, TXD, GND共3路信号；
- 传输模式：RS-232全双工；
- 信号类型：标准RS-232电平。
- 引脚定义：

UART3~UART7的引脚定义（RJ45）	
1	--
2	--
3	GND
4	--
5	TXD
6	--
7	--
8	RXD

2.2.1.4 UART8 特性参数

该串行口具备RS232和RS485两种模式，通过S1来切换是工作在232模式下还是485模式下。



UART8在RS232模式下的技术参数：

- 流向控制：支持无流控；
- **设备名：ttyLT7；**
- 信号线包括：RXD, TXD, GND共3路信号；
- 传输模式：RS-232全双工；
- 信号类型：标准RS-232电平。
- 引脚定义：

UART8的引脚定义（RJ45）	
1	VDD50
2	GND
3	GND
4	B
5	TXD
6	A
7	VDD50
8	RXD

UART8在RS485模式下的技术参数:

- 流向控制: 支持无流控;
- **设备名: ttyLT7;**
- 信号线包括: A, B差分信号;
- 传输模式: RS-485半双工;
- 信号类型: 标准RS-485电平。

2.2.1.5 UART9 特性参数

UART9 信号线在 J10 上引出。

- 硬件: 处理器内置控制器;
 - IRQ 号: 55;
 - I/O物理地址: 0x5000-0000~0x5000-40000~0x5000-80000;
 - FIFO: 16 byte 硬件 FIFO
- **波特率: 目前支持4种波特率: 9600, 14400, 19200, 28800bps;**
- 流向控制: 支持无流控;
- **设备名: ttyS1;**
- 信号线包括: A, B差分信号;
- 传输模式: RS-485半双工;
- 信号类型: 标准RS-485电平。
- 引脚定义:

UART9的引脚定义 (J10上)	
17, 18	AB
19, 20	AA

2.2.1.6 COM1 特性参数

该串口用来连接 PC 机, 查看系统运行时的控制台、调试信息等。

- 硬件: 处理器内置控制器;
 - IRQ 号: 52;
 - I/O物理地址: 0x5000-0000~0x5000-40000~0x5000-80000;
 - FIFO: 16 byte 硬件 FIFO
- **波特率: 115200bps;**
- 流向控制: 支持无流控;
- **设备名: ttyS0;**
- 信号线包括: TXD, RXD, GND;
- 传输模式: RS-232全双工;
- 信号类型: 标准RS-232电平。
- 引脚定义:

COM1的引脚定义 (DB9)	
2	RXD
3	TXD
5	GND

2.2.2 以太网接口

图1中的NET1和NET2为2路带有Link和ACT指示灯的以太网接口。2路以太网接口互为独立，通过RJ45接口引出，技术参数如下：

2.2.2.1 技术参数

- 接口类型：10 Base-T
- 接口规程：符合IEEE-802.3
- 接口速率：10M
- IRQ号：37, 2;
- 设备名：eth0, eth1;

2.2.2.2 接口定义

如图1中的NET1和NET2，接口定义如下：

NET1,NET2接口定义	
1	TX+
2	TX-
3	RX+
6	RX-

2.2.3 LCD、电源指示灯、运行指示灯接口 (J9)

J9 接口上包含了 LCD 接口信号，电源指示灯和运行状态指示灯信号，下面分别介绍：

2.2.3.1 LCD 接口

可供使用的LCD屏：

- TFT型LCD屏：
 - 支持1, 2, 4 or 8-bpp (每像素点位数) 真彩TFT屏；

- 支持16-bpp真彩TFT屏;
- 支持24-bpp真彩TFT屏;
- 支持最大24bpp的16M色TFT屏;
- 支持多种屏幕大小:

典型的实际屏幕大小: 640×480, 320×240, 160×160, 及其它规格

最大虚拟屏幕大小: 4Mbytes

最大虚拟屏幕大小 (@64K色): 2048×1024 and others

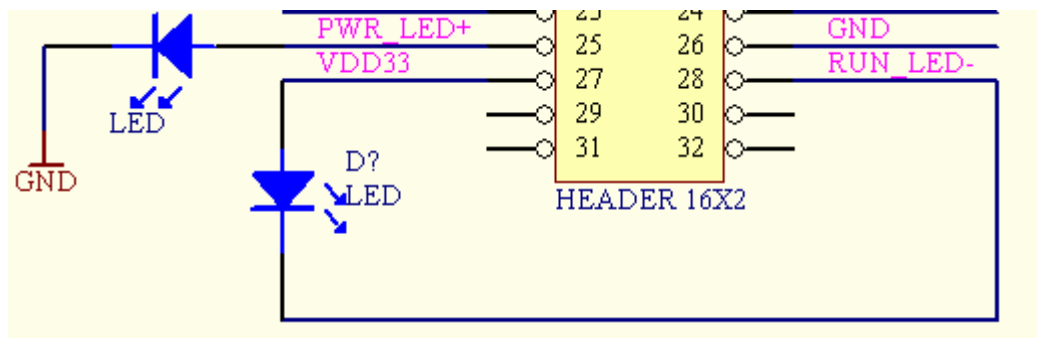
- 设备名: /dev/fb0
- 引脚排布: 在J9和J10上提供了信号。

2.2.3.2 电源指示灯，状态指示灯

PWR_LED+信号脚与GND信号之间连接一个发光二极管 (PWR_LED+连接正极), 可以作为电源指示灯使用, 在板子正确上电后, 该发光二极管就会点亮;

RUN_LED-信号脚与VDD33信号之间连接一个发光二极管 (RUN_LED-连接负极), 可以作为电源指示灯使用, 该板的Linux系统中提供了对该指示灯的驱动程序, 应用程序可以控制它的点亮 (具体方法参考软件支持章节)。

推荐的连接方法如下图所示:



2.2.3.3 接口定义

J9接口定义			
Pins	Net	Pins	Net
1	VDD50	2	nRESET
3	LCDVF0	4	LCD_PWREN
5	LCDVF2	6	LCDVF1
7	VD3	8	VD2
9	VD5	10	VD4
11	VD7	12	VD6
13	VD11	14	VD10
15	VD13	16	VD12
17	VD15	18	VD14

19	VD19	20	VD18
21	VD21	22	VD20
23	VD23	24	VD22
25	PWR_LED+	26	GND
27	VDD33	28	RUN_LED-
29	--	30	--
31	--	32	--

2.2.4 电源、触摸屏、RS485 接口 (J10)

2.2.4.1 电源接口

供电要求: +5VDC 输入;

整板功耗: 静态工作电流 300mA (@5V), 启动峰值电流 500mA (@5V);

2.2.4.2 触摸屏接口

可以支持 4 线电阻式触摸屏。

2.2.4.3 接口定义

J10接口定义			
Pins	Net	Pins	Net
1	--	2	--
3	--	4	--
5	--	6	--
7	--	8	--
9	VCC50	10	VCC50
11	VCC50	12	VCC50
13	GND	14	GND
15	GND	16	GND
17	AB	18	AB
19	AA	20	AA
21	VLINE	22	nXBACK
23	LEND	24	nXPON
25	AIN5	26	nYPON
27	AIN7	28	YMON
29	VFRAME	30	XMON
31	VM	32	VCLK

2.2.5 SD 卡接口

支持 SD 存储卡标准 (ver.1.0) / MMC 卡标准 (ver.2.11)。

2.2.6 复位电路-RESET

采用专用复位集成电路设计，确保系统上电复位的可靠性，低电压自动复位，同时支持手动复位。

如图1中的RESET即为手动复位键。

2.2.7 蜂鸣器-BUZZER

如图1中BUZZER用作声音提示，用户可以通过软件开启/关闭鸣响，或灵活设置鸣响时间。设备名为：/dev/buzz。

3 软件支持

3.1 文件说明

相关的软件工具文件，位于光盘目录下：

目录	文件	描述
应用程序实例源码	Server.tgz	串口通讯实例（主动发送方）；
	Client.tgz	串口通讯实例（被动接收方）；
	Busybox.tgz	Busybox源码包，包含看门狗的测试例子；
	buzz.tgz	蜂鸣器设备实例；
	state.tgz	状态指示灯实例；
交叉编译工具链	Cross-2.95.3.tar.bz2	交叉编译工具

3.2 准备开始

NARI2410MU定制板产品已经预装Linux2.4.18系统，开机即可进入系统控制台。

两种方式进入控制台：通过串口终端，或者通过网络telnet服务。

3.2.1 通过 telnet 登陆

telnet 是一个经常被使用的远程登录工具，它可以通过网线来登陆 Nari2410MU 的控制台，不需要连接串口线。


```
mkdir -p /var/lib
mkdir -p /var/run
mkdir -p /var/log

/etc/rc.d/init.d/netd start

insmod serial_16c554.o //加载了多串口的驱动
insmod secs8900a.o //加载了第二个网口的驱动
insmod s3c2410-buzz.o //加载了蜂鸣器的驱动
insmod s3c2410mu_state.o//加载了状态灯的驱动

#qtopia & //启动图形界面 QT，这里屏蔽了，不启动

ifconfig lo 127.0.0.1 //设置两个网口的 IP 地址，两个网口不要设置在同一个网段
ifconfig eth0 192.168.253.9 up
ifconfig eth1 192.168.250.9 up

/bin/hostname -F /etc/sysconfig/HOSTNAME

hwclock -s //从 RTC 读出系统时间
```

在这个脚本的最后加入应用程序的命令行，系统启动后就会自动运行这个应用程序。

3.2.3 通过 telnet 登陆

telnet 是一个经常被使用的远程登录工具，它可以通过网线来登陆 Nari2410MU 的控制台，不需要连接串口线。

当 NARI2410MU 定制板已经被设置了 IP 地址后，它就可以作为一台 telnet 服务器了，因为系统启动的时候已经在后台开启了 telnet 服务程序。用网线连接 NET1 跟 PC 机网口。

NARI2410MU 定制板的缺省 IP 地址被设置为 192.168.253.9，则在主机的终端中敲入以下命令：

```
telnet 192.168.253.9
```

输入用户名 **root**，不需要密码按回车，即可登录 NARI2410MU 定制板的控制台。

3.3 用户开发应用程序

下面说明如何基于 Nari2410MU 定制板开发应用程序。下面将会首先说明一个基本的应用程序（hello）如何编写、编译、下载到板子上运行；再对应用程序如何对板子上的串口、Watchdog、LED 灯、蜂鸣器操作进行说明。

3.3.1 安装交叉编译器

(1) 在编译linux之前, 要先安装交叉编译toolchain, 安装包位于光盘中的\交叉编译工具链\cross-2.95.3.tar.bz2。

在/usr/local 目录下建立名为arm的目录, 进入该目录, 执行解包:

```
mkdir /usr/local/arm
cd /usr/local/arm
tar xjvf cross-2.95.3.tar.bz2
```

(2) 编辑/etc/profile, 找到 pathmunge /usr/local/sbin 在其下面添加一行:

```
pathmunge /usr/local/arm/2.95.3/bin.
执行: export PATH=$PATH:/usr/local/arm/2.95.3/bin
```

这样, 应用程序交叉编译均可用 arm-linux- 来指定使用该交叉编译器。

3.3.2 编写 Hello 代码

下建立 hello 目录:

```
mkdir hello
```

编写以下代码, 将它保存为 hello/hello.c

```
#include <stdio.h>
int main(void)
{
printf("Hello, Nari2410MU!\n");
return 0;
}
```

3.3.2 编译 hello

输入以下命令交叉编译:

```
arm-linux-gcc -o hello hello.c
```

将在 hello 目录下生成 hello 可执行文件。下面我们就可以把 hello 下载到板子上执行, 推荐采用 ftp 方式传送到板子上。

3.3.3 采用 FTP 方式

采用 FTP 方式需要在 PC 机端建立一个 FTP 服务器。例如可以使用 Red Hat Linux9.0 中自带的 VSFTPD。关于 FTP 服务器的详细建立方法可参考网络相关资料。

在 PC 机上把 hello 复制到 ftp 服务的共享目录 (如果是匿名用户则为/var/ftp 目录), 然后

在 Nari2410MU 板的控制台上执行：

```
ftp 192.168.253.1 ; 登录 ftp 服务器（这里假设您的 ftp 服务器 IP 为 192.168.253.1）。
>get hello ; 下载 hello
>bye ; 退出 ftp 登录
# chmod a+x hello ; 改变 hello 的可执行权限
# hello ; 执行 hello，观察执行结果。
```

3.4.应用程序实例

3.4.1 串口通讯实例

该实例的测试准备工作：需要PC机的一个COM口开启minicom作为控制台或者telnet登陆Nari2410MU的控制台。采用与串行口RJ45座配套的电缆线连接UART1和UART2。

这个例子使用到了两个程序：ttypst_server和ttypst_client。

- ttypst_server的功能是向指定的串口主动发送28个字符，然后等待接收到相同的字符返回；接收之后再次发送，如此循环。
- ttypst_client的功能是等待指定串口的数据，接收到28个字符后，校对是否正确，如果正确将28个字节发送回去，如此循环。

现在我们连接好了UART1和UART2，可以开始对通了，测试操作如下：

（1）在控制台上运行：

```
ttypst_client 0 9600 8 1 0 0 &
```

命令格式为：ttypst_Server或Client+端口号+波特率+数据位+停止位 + 有无流控+232还是485模式。具体取值，可以直接输入ttypst_client查看help信息。

从这个命令行看，就是使ttyLT0工作在9600波特率，1位停止位，无流控，在232模式下监听端口数据，&表示该程序在后台运行。

运行状况如下所示：

```
[@lyt /]# ttypst_client 0 9600 8 1 0 0 &
-----You chosed serial /dev/ttyLT0 -----
pen_port: Success to open /dev/ttyLT0
pen_port: Success to open port.
Begin Client ...
```

UART0开始等待。

（2）在控制台上运行：

```
ttypst_server 1 9600 8 1 0 0
```

运行状况如下所示：

```
[root@lyt /]# ttypst_server 1 9600 8 1 0 0

-----

Success to open port /dev/ttyLT1
pen_port: Success to open port.
```

UART1口开始发送28个字符一组的数据。

(3) 如果通讯顺利，UART1和UART2上的发送和接收指示灯将被点亮并闪烁，通讯一段时间后，在控制台上按下Ctrl+C来中止发送进程：

```
Start server, pls wait ... ..Ctrl+C has been pressed!  
Port1 Server Test OK:  
Send=839, Rev=839, Err=0, Time=62.976289S  
Success to close port /dev/ttyLT1  
[root@lyt /]#
```

这里打印出了通讯的情况，发送包的数量、接收包的数量、错误的字节数量、通讯的时间。

以上功能的实现，全部由光盘中server.tgz，client.tgz的源代码完成，用户在自行编写串口程序时可以参考。

3.4.2 看门狗

- 看门狗的设备名：/dev/misc/WDT
- 支持的API函数有：open，write，ioctl，close
- Open函数，用于打开看门狗设备。注意，系统在初始化时并没有开启看门狗功能，应用程序必须调用这个函数来打开看门狗功能。看门狗打开后默认为15秒钟超时，也就是说15秒之内必须喂狗来保证系统不复位。目前内核支持“打开后不能关闭看门狗”。
- Write函数，Write函数可以用来喂狗，也可以用来关闭看门狗，但是因为我们目前支持“打开后不能关闭看门狗”，所以关闭看门狗功能那不可用。
- Ioctl函数，WDIOC_GETSUPPORT命令，获得看门狗的功能支持信息；
 - i. WDIOC_KEEPLIVE命令，通过调用它也可以喂狗；
 - ii. WDIOC_SETTIMEOUT命令，用来设置新的延时参数；
 - iii. WDIOC_GETTIMEOUT命令，获得目前的延时参数

应用层实例代码请参考：busybox-1.00/miscutils/watchdog.c

如何测试看门狗？在控制台上输入：

```
watchdog -t 10 /dev/misc/WDT
```

后台程序watchdog开始对看门狗进行每10秒一次的喂狗。注意时间间隔不能超过15秒，否则系统就会复位。

3.4.3 蜂鸣器

在控制台上输入以下指令，测试蜂鸣器的驱动：

```
# buzz 1 500 □蜂鸣器响500ms
```

```
# buzz 1 0 □蜂鸣器开始鸣叫
```

```
# buzz 0 0 □蜂鸣器停止鸣叫
```

因该设备非标准设备，在此特别说明：

该设备文件为“/dev/buzz”，用户可以通过打开设备文件并调用“ioctl”函数来控制蜂鸣器的鸣叫。例如，下面的代码将打开蜂鸣器并令其鸣叫500ms。

```
int on=1;//if on=0, turn off the backlight
```

杭州立宇泰电子有限公司

第 16 页 共 18 页

电话：0571-56763523 56763526

网址：www.hzlitai.com.cn
www.armsystem.com.cn

传真：0571-89902166

```
int fd = open("/dev/buzz", WRONLY);  
ioctl(fd, on, 500); //turn on for 500ms  
close(fd);
```

3.4.4 LED 状态指示灯

在控制台上输入以下指令，测试LED状态指示灯的驱动：

```
[@lyt /]# state
```

Usage: state 0|1

Example: state 1 -->ON.

Example: state 0 -->OFF.

```
[@lyt /]# state 1 //状态指示灯点亮
```

```
[@lyt /]# state 0 //状态指示灯熄灭
```

```
[@lyt /]# state 1 //状态指示灯点亮
```

因该设备非标准设备，在此特别说明：

该设备文件为“/dev/state”，用户可以通过打开设备文件并调用“ioctl”函数来控制LED的点亮。例如，下面的代码将。

```
fd = open("/dev/state", 0);  
if (fd < 0) {  
    perror("open device state_led error.");  
    exit(1);  
}  
ioctl(fd, on, NULL);  
close(fd);
```

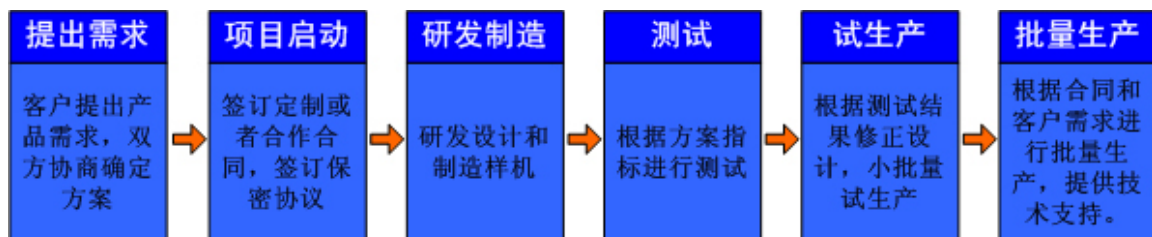
4 合作方式和定制服务

4.1 合作方式

NARI2410MU 定制板应用板为批量生产的成熟产品，立宇泰针对本产品提供的合作方式为：批量供板方式。

考虑到不同用户有特殊需求，立宇泰电子还可以提供进一步的定制服务。

定制流程如下：



欢迎联系我们：

杭州立宇泰电子有限公司

第 17 页 共 18 页

电话：0571-56763523 56763526

网址：www.hzlitai.com.cn
www.armssystem.com.cn

传真：0571-89902166

产品咨询: 0571-56763523/25/26-801

技术服务: 0571-56763523/25/26-802

Email: lyt_office@yahoo.com.cn

4.2 定制服务

立宇泰电子在嵌入式系统产品设计和ARM处理器应用方面累积了丰富经验。我们的解决方案建立在已有的成熟架构上,能够有效缩短用户产品向市场投放需要的时间,以及降低在软硬件移植方面的风险。

立宇泰电子将针对用户的需求,最终产品的应用场合和应用特点,提供专项设计开发和全面测试服务,我们的目标是为用户大大节省项目的研发时间和成本。

我们大量采用的处理器技术包括:低功耗三星ARM7, ARM9处理器,完全支持现有的各种操作系统。项目定制系统将主要以立宇泰的标准模块和开发平台为基础而搭建构成。

我们的定制服务根据与客户签订的合同启动,内容可以包括结构设计,开发,认证,生产和售后服务。