

JEDI 系列使用手册之 EasyARM2200 篇

(ADS 版本：1.2)

上海祥佑数码科技有限公司

2005—8—20

Revision History:

Revision	Date	Comment
1.0	June 18, 2004	Initial Version
1.1	September 30, 2004	Release09302004
1.2	Aug 20,2005	Add Easy ARM2200

目 录

JEDIPROBE 开发仿真器使用手册.....	1
第一章 JEDIPROBE开发仿真器介绍.....	2
开发仿真器套件外观.....	2
第二章 开发仿真器硬件说明.....	4
1. 核心板说明	4
2. 前板说明.....	4
3. 背板说明.....	5
4. 左侧板说明	5
5. 右侧板说明	5
6. 上下板说明	5
第三章 如何测试开发仿真器的好坏.....	5
1. 串口是通的吗？	5
2. 网口是通的吗？	11
3. 仿真器能工作吗？	11
4. JTAG 口是通的吗？	11
第四章 如何在 ADS 中安装 JEDIPROBE.....	17
第五章 如何连接 ZLG 的 EasyARM2200 开发板.....	21
第六章 常见连接问题集锦.....	28

第一章 JEDIPROBE 开发仿真器介绍

JEDIPROBE(JTAG Ethernet Debug Ice)



JEDIPROBE简介：

基于ARM的开发目前正成为嵌入式应用领域的一个热点，MICETEK 在前一代产品的基础上，又新推出具有更高性价比的JEDIPROBE 仿真器。其显著特点是快速应用程序下载，实现快速以太网口与主机通讯，极大方便了远程调试和团队共享的功能。

JEDIPROBE特点：

- CPU 实时控制

JEDIPROBE 与ARM通过高性能的JTAG口相连,实现对CPU的控制，不占任何用户资源，高速程序代码下载200-1000KB/S。

- TCP/IP网络连接

JEDIPROBE 与主机通讯采用10/100Base-T Ethernet 和串口通讯，网口既快速又可实现远程调试。

- Flash Memory 在线编程

JEDIPROBE 与HITool、JEDIVIEW 集成IDE结合，支持多种Flash Memory 算法库，并支持用户自定义Flash算法。还可以使用免费的Flash Utility应用程序。

- 用户寄存器自定义功能

用户可以方便地增加常用的寄存器和周边的寄存器,并可以按位方式显示,对于新CPU更加重要。

- Target 板自诊断功能

支持对目标Memory Fill, Search, Copy, Test, Compare Data Bus 和Address Bus TEST。这些功能对硬件调试非常有用。

- 硬件断点，支持条件断点和断点计数。

- 在线Firmware升级，支持最新的ARM或MIPS内核，保护用户投资。

开发仿真器套件外观



1. 开发套件构成

JEDIPROBE 开发仿真器套件包括:

- 1) 一个已测试好的JEDIPROBE 开发仿真器；
- 2) 产品使用说明书一本；
- 3) 一个JTAG 调试头,该调试头可用来烧写和调试程序；
- 4) 一条直连串口线和转接座；
- 5) 一条交叉线网线；
- 8) 一个5V 直流电源；
- 9) 一张JEDIPROBE光盘，包含有关驱动程序以及使用电子文档；
- 10) 一个包装盒。

光盘内所附软件：

1. JEDIPROBE 的驱动程序，用于添加到ADS、SDT、GreenHill等著名调试程序中，用于开发和调试ARM的程序；
 - 1) rdiJediProbe.dll
NAND FLASH 启动引导程序，通过bios 或jtag 烧入nand flash 0 地址,d9~d12 闪烁
 - 2) JediProbeCfg.exe
nor flash 启动,通过usb 下载,通过bios 或jtag 烧入nor flash 0 地址
 - 3) S3C2410_TEST
三星测试程序，包括各种测试程序（IIC，sd），仿真器usb 下运行载，
 - 4) 2410app
自编的测试程序，SD，按键，
2. JEDIProbe Flash Utility 烧写程序,这是一个专业水准的免费软件。而且，接口是开发的，自己可以任意添加新的FLASH 芯片到软件中，并且很方便同其他使用者交流。
3. JEDIView for ARM 调试环境使用版本及开发工具包，含各种流行的开发板的例子程序，如S3C44B0、S3C4510、AT91RM9200、S3C2410、W90N740等网口驱动，串口驱动，USB 驱动测试程序；
4. JEDIView for UC/OS II 免费测试版调试环境。可以更好的帮助你了解和和使用uC/OS II的嵌入式操作系统的调试方法和过程。

开发系统文档：

- 1) JEDIPROBE 使用手册
- 2) JEDIPROBE FAQ
- 3) ARM芯片的应用手册

2. JEDIProbe 开发仿真器资源

硬件技术参数：

- 16/32-bit CPU 内核
- 系统时钟：80MHz
- 一个通道UART/Console 口
- 一个10M以太网网络接口
- JTAG 可工作到40Mhz，自动适应并可以调整工作频率
- 下载速度可达 1M/S (JEDI-II) (300K ~400K JEDI probe)

第二章 开发仿真器硬件说明

1. 核心板说明

- 1) J1 (旁边写着JTAG) 的20 芯连接器座子是ARM 公司推荐的标准JTAG 接口；
- 2) S1 是复位按键；
- 3) D4 是电源指示灯，核心板正 常供电它就会亮；
- 4) D3 通过一个非门连接到CPU 的PWREN 引脚上，用作指示；
- 5) D1 和D2 分别连接到CPU 的GPB5 和GPB6 引脚上，用作系统指示；
- 6) JP1 用作系统启动模式选择。

2. 前板说明



- 1) DC output 为电源输出插座，输出直流电压的范围是3~6V（注意由于钽电解电容的耐压值只有16V，所以切记正极、负极的标识，否则仿真器内部将会损坏）；
- 2) Reset 为复位开关，将它按下2~3秒，系统就可以是目标板复位；
- 3) TARGET为标准的JTAG 20PIN接口，通过它仿真器可以对目标板控制或者通讯使用，使用20 芯排线与目标板上面的JTAG口进行一对一连接。用户也可以请根据自己需要，对照图纸来设置或更改了解排线，默认情况下这些排针都不接短路块；
- 4) Power、Active、COMM 为三个LED指示灯。

3. 背板说明



- 1) Power 为电源开关；
- 2) Ethernet 为以太网接口，由于支持有关以太网调试软件的连接。
- 3) Rx/Crs、Col/Link 为两个LED指示灯；
- 4) RS23 为串行接口，同标准的RJ15的连接线可以方便的同上位PC机器通讯，也可以按照双工的方式连接调试软件。

4. 左侧板说明



- 1) DC input 为电源输入插入口，内正外负；注意输入直流电压的范围是3~6V，一定不得超过9V，否则将损坏仿真器！另外电源插座的内芯为正极、外壳为负极，极性不符的电源插头也将损坏仿真器。
- 2) WARRANTY 易碎标签是产品的保修凭证，千万注意：个人不要自行打开，一般 JEDIPROBE 仿真器硬件上作了充分的保护，很不容易损坏；
- 3) Trigger 为硬件触发接口，（特殊用途保留，一般不使用）。

5. 右侧板说明

仿真器右侧方的面板上，暂时无任何标识，以后的版本可能会有新的改动。

6. 上下板说明



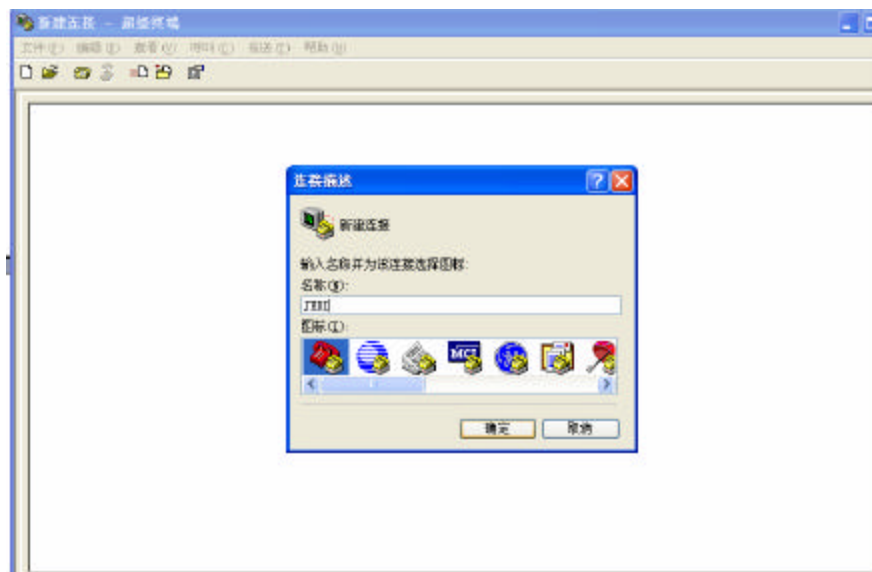
第三章 如何测试开发仿真器的好坏？

1. 串口能工作吗？

将自带的RS232线一端连接PC机RS232插座，另一端到JEDIPROBE后面板上面的RS232 Device接口（RS232），并将DC5V 电源插进电源插座，注意输入直流电压的范围是3~6V，一定不得超过9V，否则将损坏仿真器！另外电源插座的内芯为正极、外壳为负极，极性不符的电源插头也将

损坏仿真器。

1、在PC 机上运行超级终端程序，在菜单Configuration/option 里面作如下设置：



2、在serial port 菜单里面点击connect，让串口连接起来；



3、将电源开关拨向上，应该看到仿真器上面的Rx/Crs、Col/Link 亮了。仿真器上面的LED 闪烁几次之后停止闪烁，D2在D1 停止闪烁之后则是不断闪烁；



如果使用5V 稳压电源, 那么现在系统的总电流应该是200mA 左右（没有连接JTAG 仿真器）；



4、在超级终端里面，应该可以看到串口输出了如下信息：



出现上面的信息，说明仿真器串口是通的。

2 网口是通的吗？



将PC 机的网卡IP 设置为192.168.0.22，子网掩码为255.255.255.0，网关和DNS 不用填，如何输入ping 192.168.0.22 -t 来ping 自己的PC 机：
出现上面的信息，说明网口是通的。

3 仿真器的JTAG 口是好的吗？

一般仿真器的JTAG口界面简单，不易出错。如果你比较怀疑JTAG的接口有问题，可以按照JTAG接口标准，用仪器测量。

第四章 如何在ADS中安装JEDIPROBE

JEDIPROBE 开发仿真器可以通过添加自带的动态链接库文件，嵌入到ADS软件中，实现ADS所有的功能，这样就可以在ADS环境中借助JTAG 调试逻辑来控制CPU 的片选、地址、数据等信号实现的。

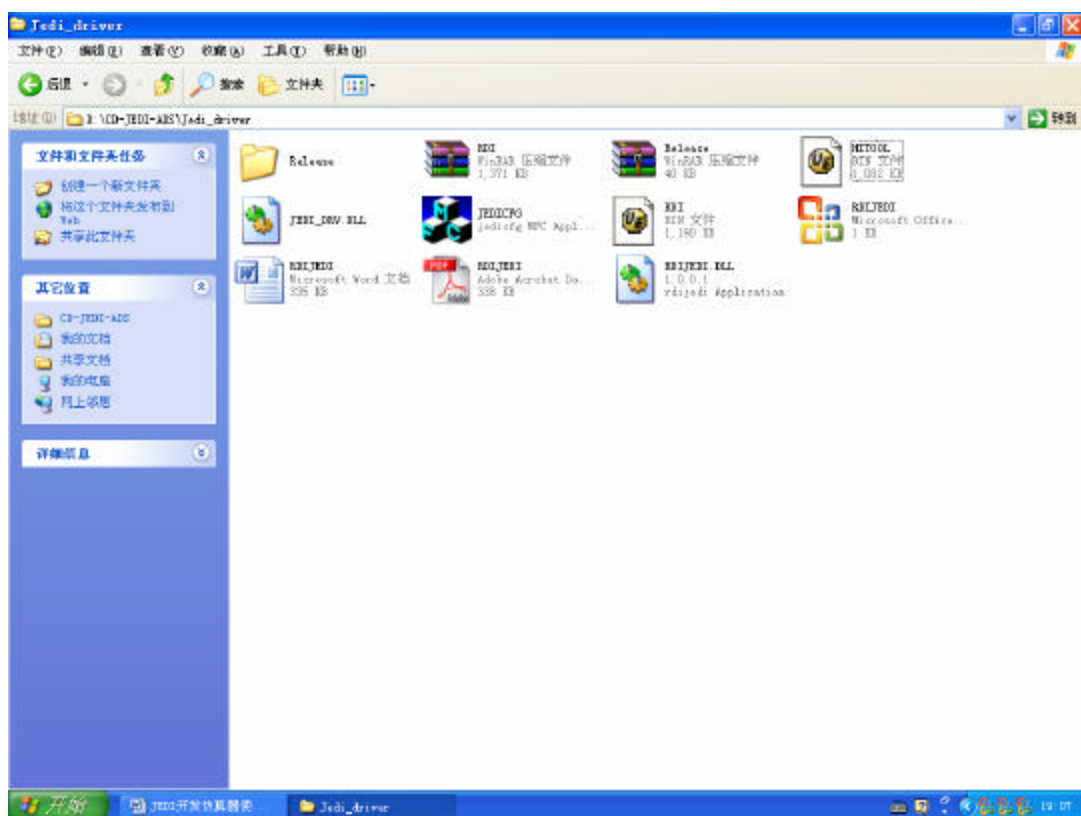
安装步骤：

一、拷贝文件

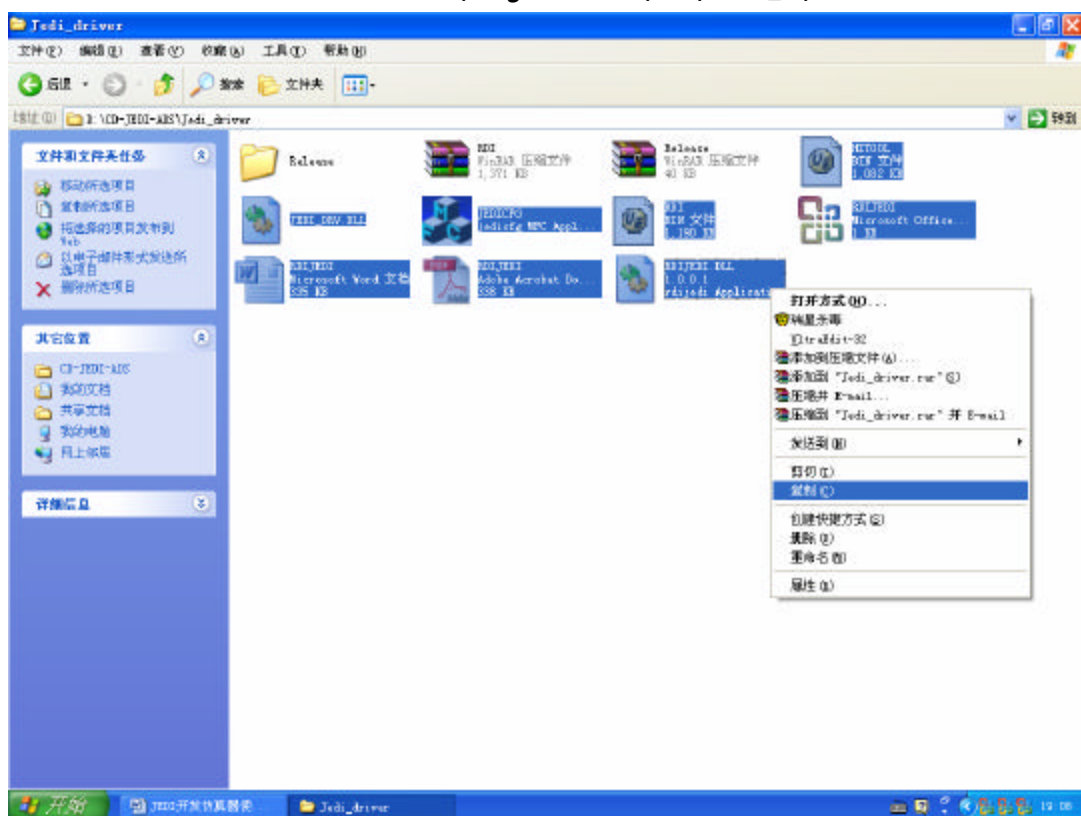
在自带的CDROM中，找到JEDIPROBE_Drive目录，将下面的所有文件复制到ADS安装目录下的BIN目录，如：C:\Program Files\ARM\ADSv1_2\BIN。（注明：Release.rar 文件是全部文件的压缩档案，以便拷贝传输）

下面我们用图示详解这个过程。

1. 进入 CDROM 中，找到 JEDIPROBE_Drive 所在目录；

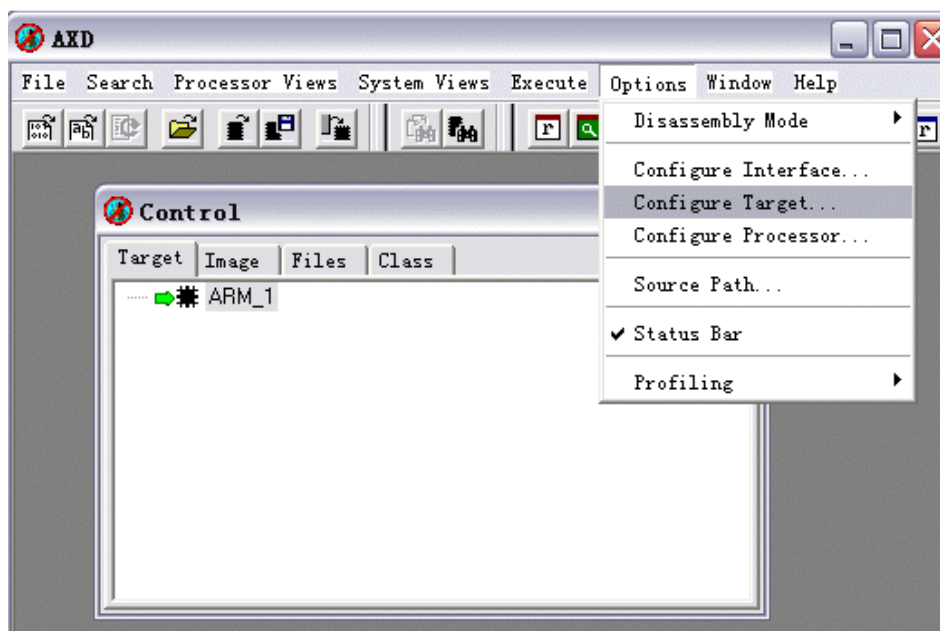


2. 复制拷贝到目标文件目录。如：C:\Program Files\ARM\ADSV1_2\BIN。

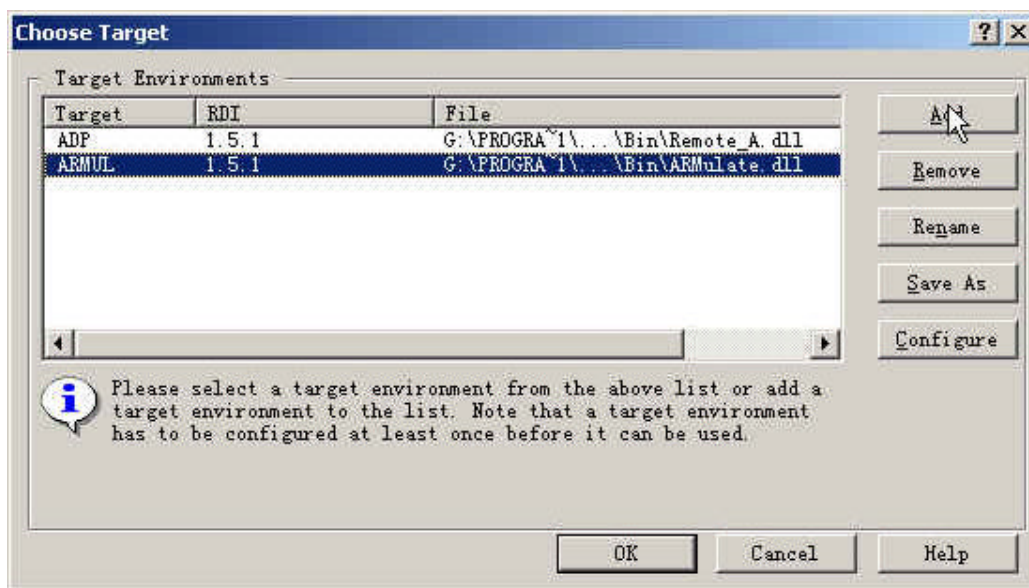


二、安装驱动程序

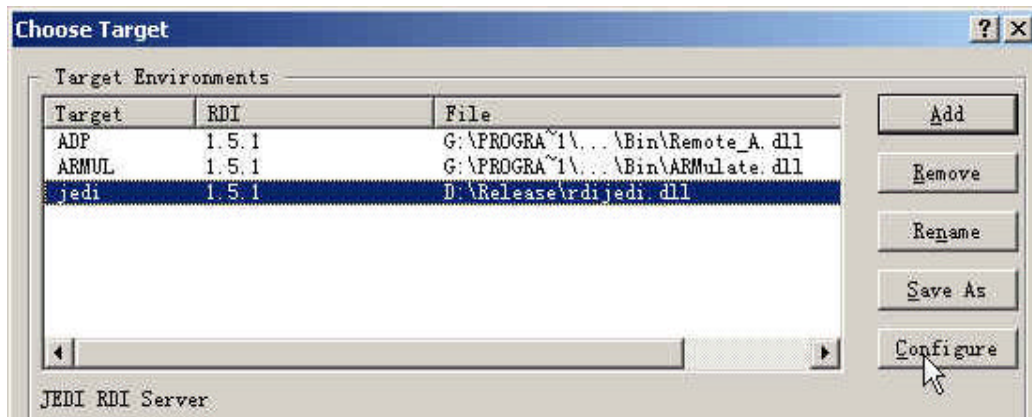
1. 运行ADS, 在AXD调试环境中, 打开『Options』->『Configure Target』, 弹出一个Choose Target 窗口;



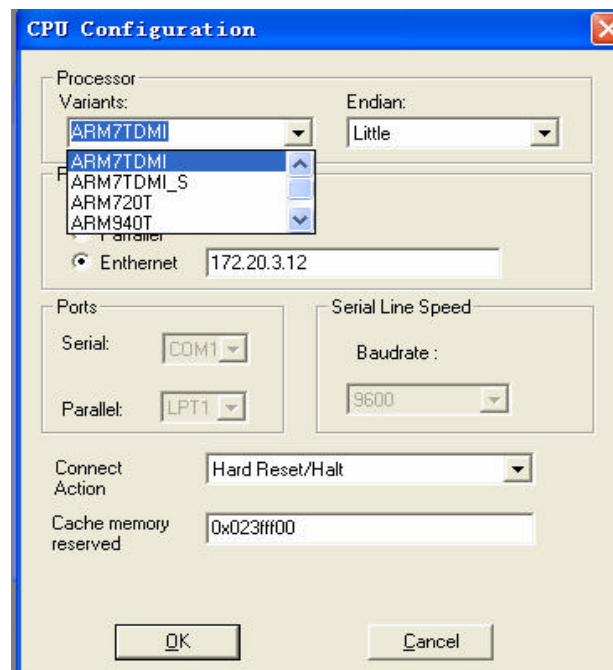
2、Add 添加JEDIPROBE的驱动程序,



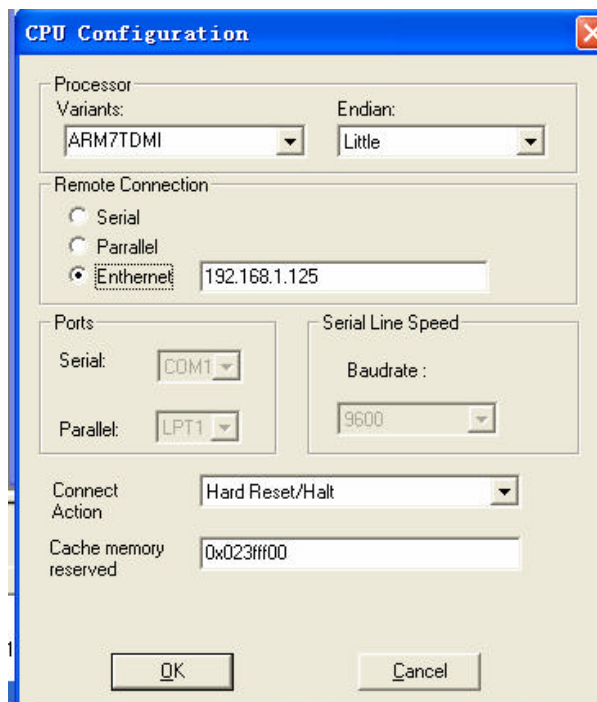
找到安装文件。



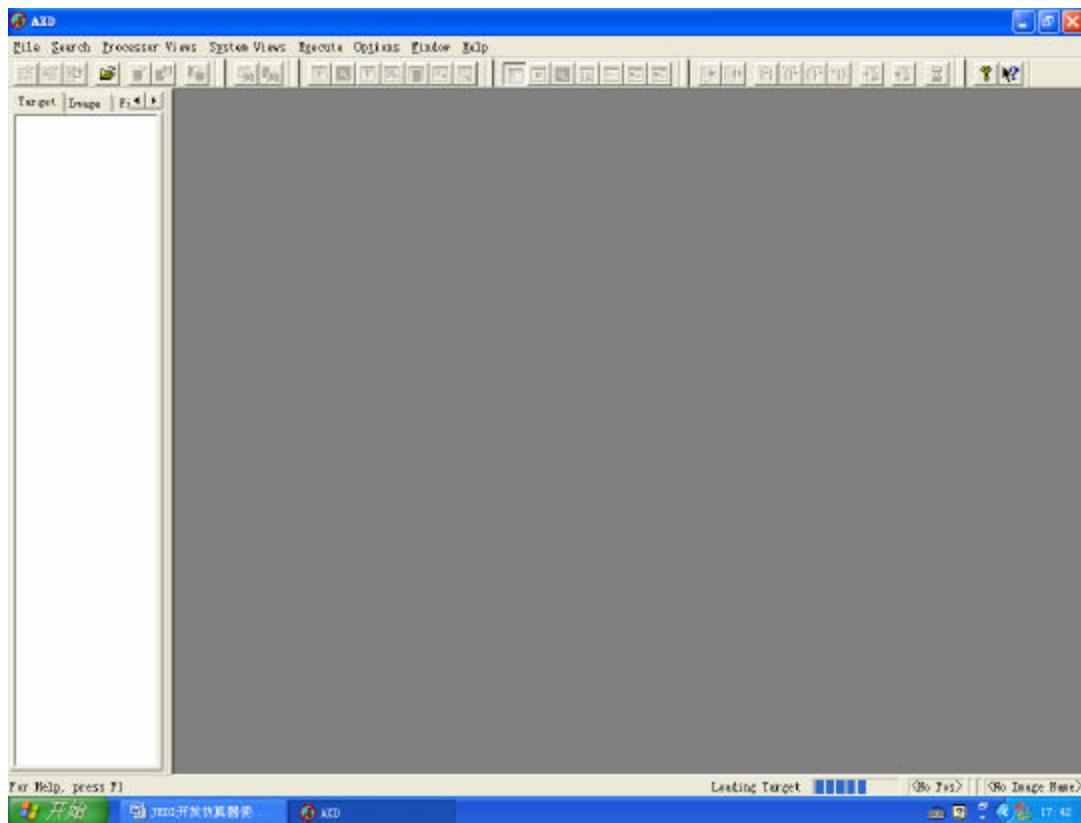
3、点击【Configure】所支持的ARM都列出来了，有ARM7TDMI、ARM720T、ARM920T、ARM940T等。



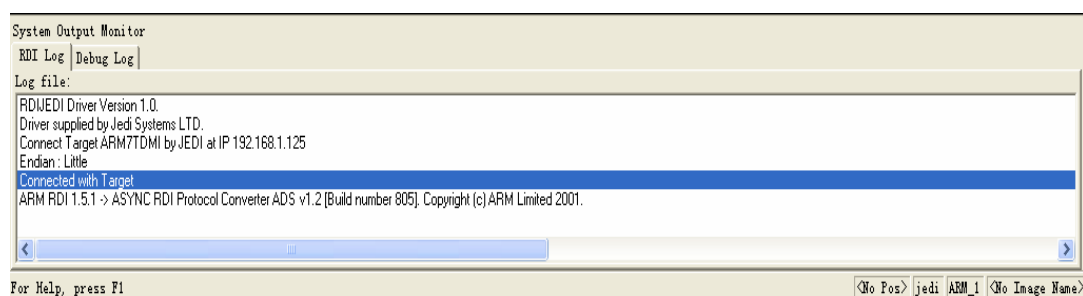
我们在这里选择 ‘ Ethernet ’ 来作为连接方式。



4. 选择 “ OK ” 并退出设置窗口，驱动程序开始被加载执行，



最后会出现如下信息在窗口中，由此可以看出连接已经正常。

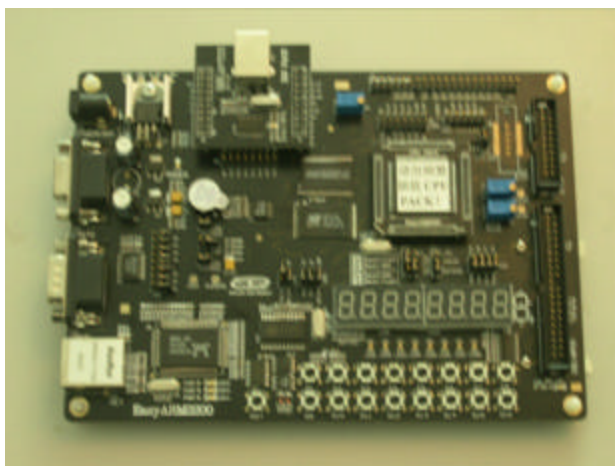


接下来，就可以调用ADS的其他各项功能，来调试你自己的各种程序了。

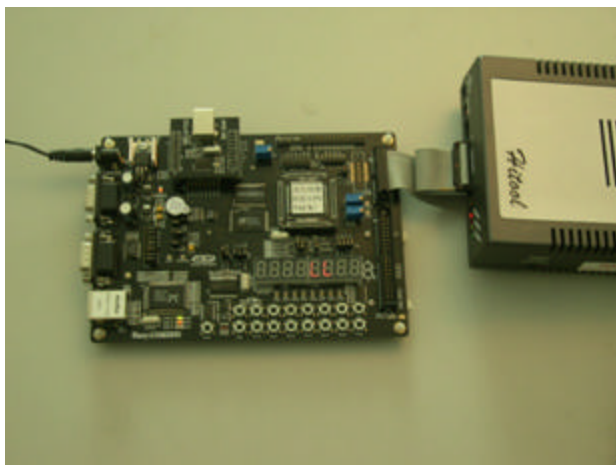


第五章 如何连接 EasyARM2200 开发板

一、参看 ZLG 的有关资料，我们首先认为，JP1 必须断开，JP10 必须连接上，使得开发板处于 JTAG 使能状态。JP6 为 BANK0—Flash BANK1—RAM. JP7 BOOTSEL — INSIDE。

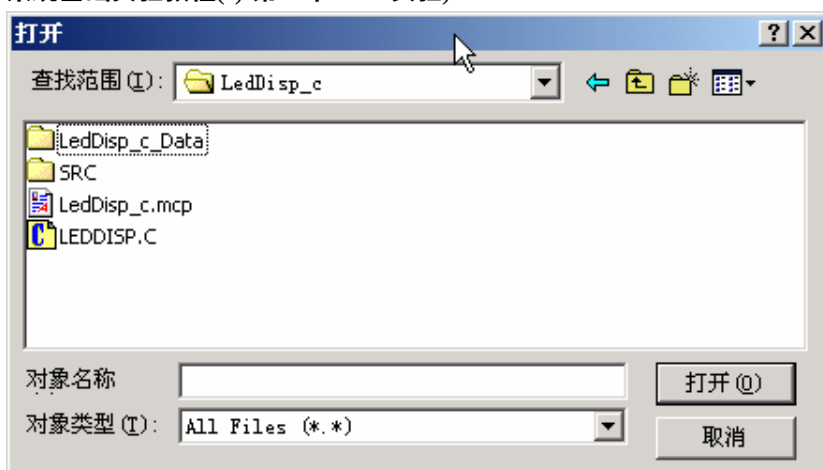


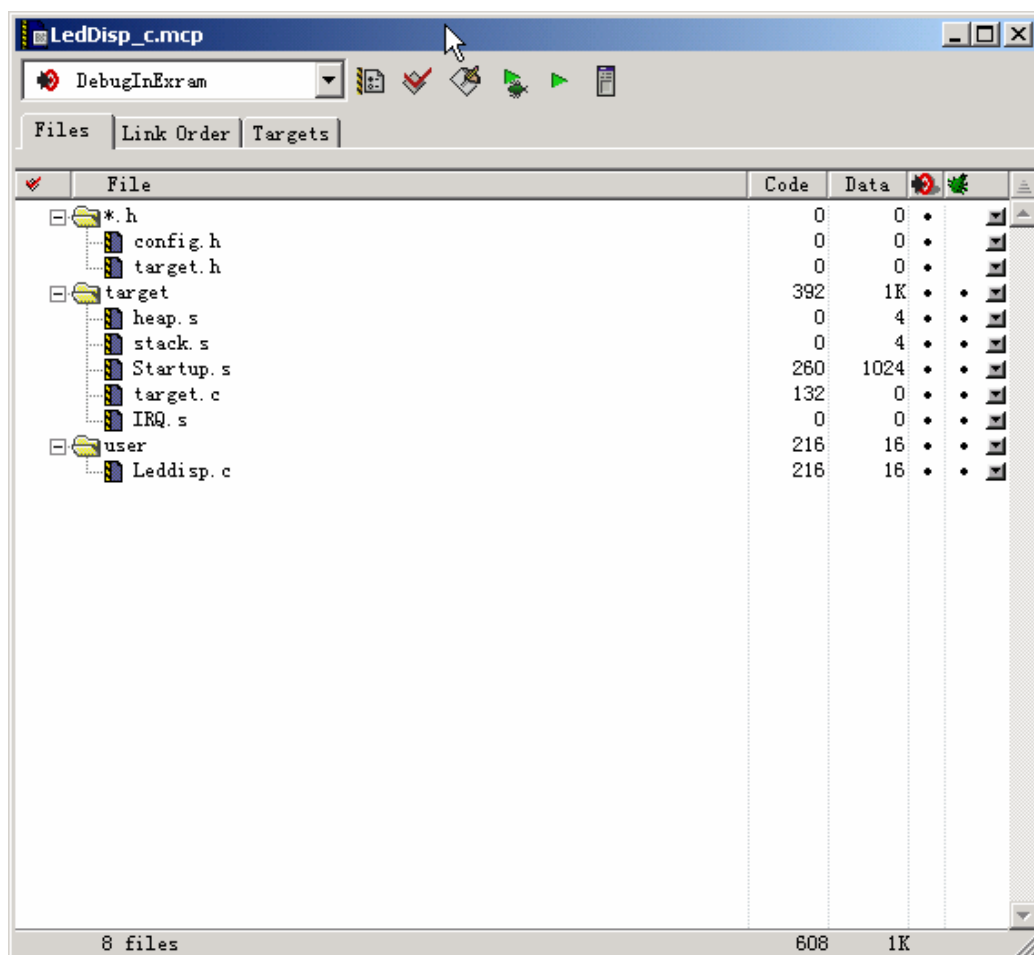
连接图示：



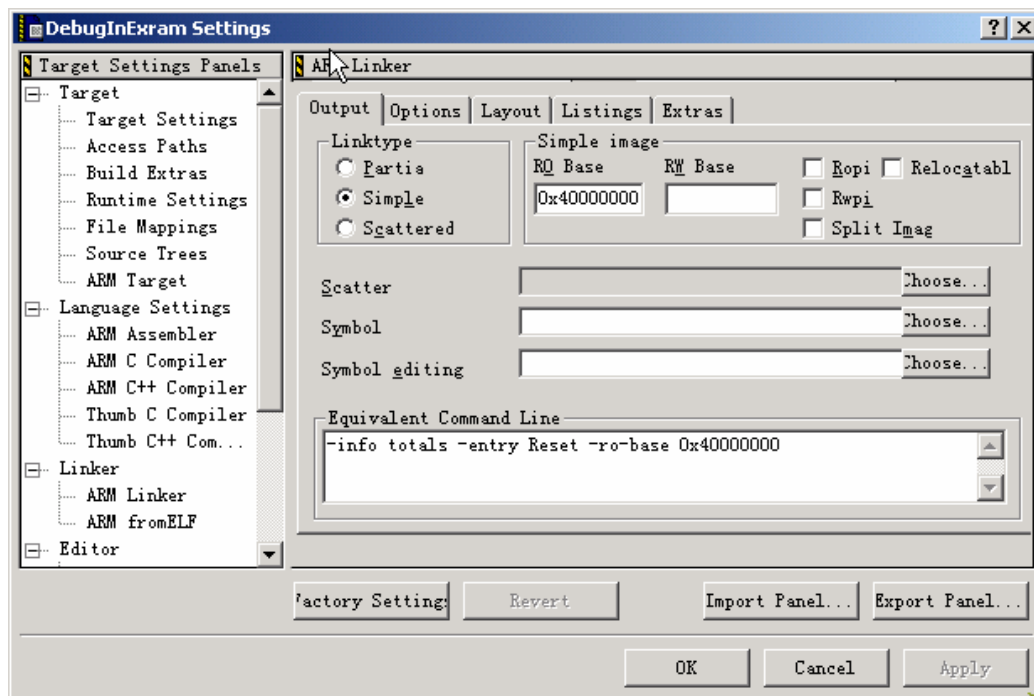
二、我们以 ZLG CDROM 中的例子程序为例来说明调试过程。

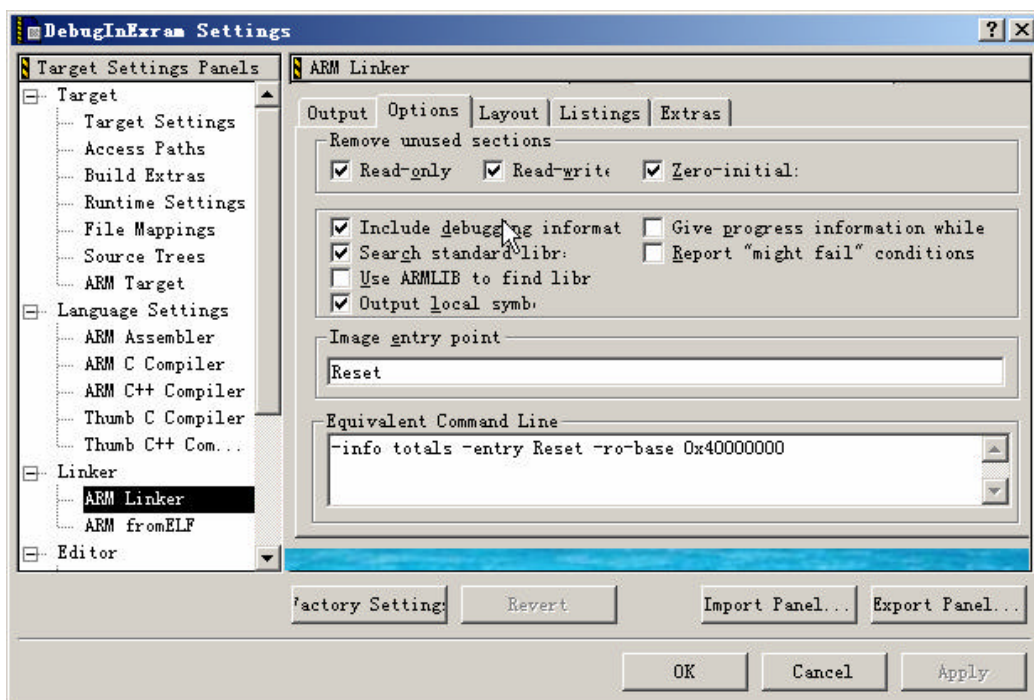
首先，用 ADS Version 1.2 调用 ZLG CDROM 中的一个 LED 点灯例子，(E\ARM 与嵌入式系统基础实验教程(1)\第 3 章\3.11 实验)





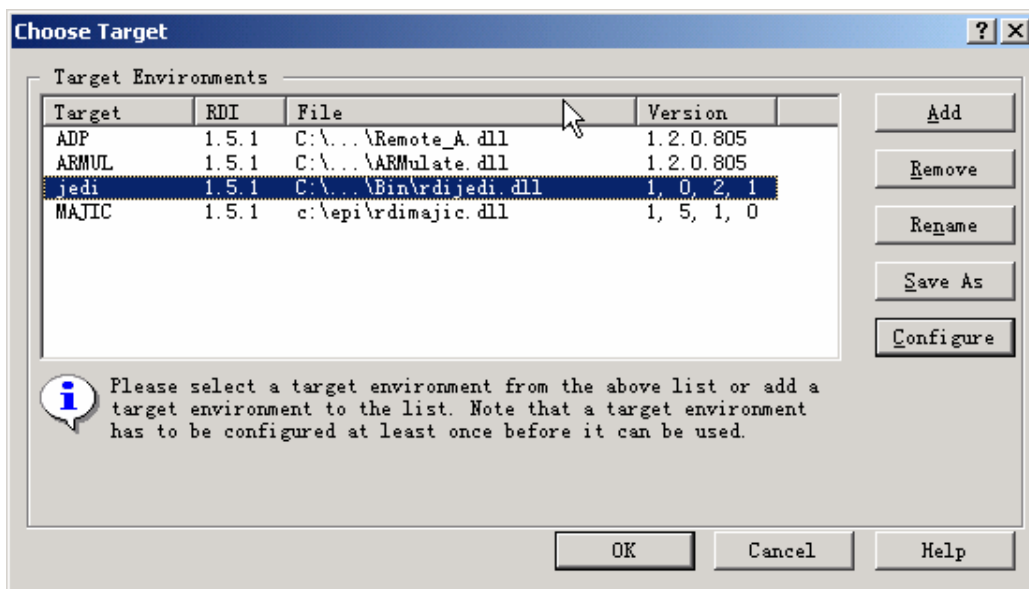
然后，改变其编译的设置参数，（注意：只需要改变一处，其余参看 ZLG 说明书自行更改）



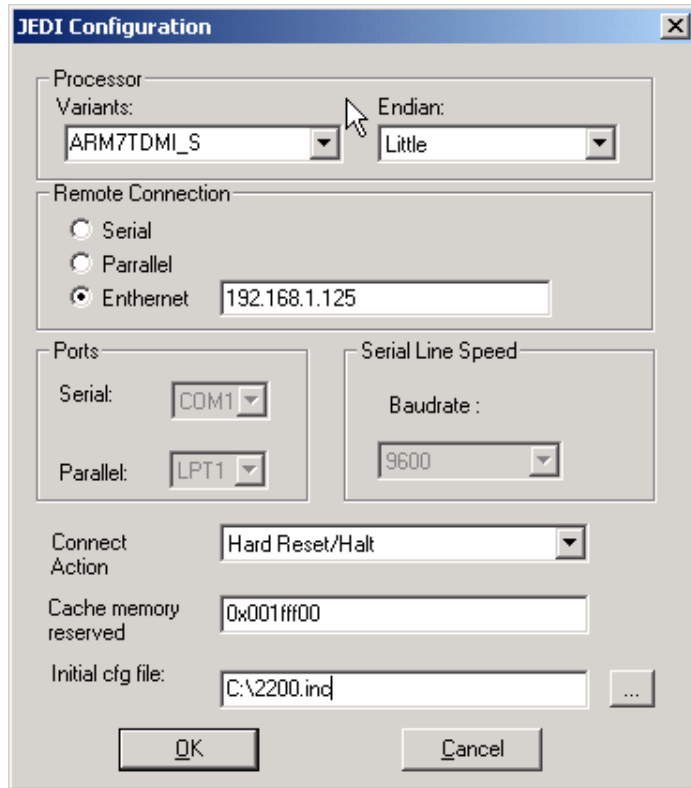


设置好后，重新编译。这里要注意的是，ZLG 的程序本身强调的是在 ISP 模式下烧录在 FLASH 中运行，而 FLASH 中是无法做在线调试的，原因很简单，FLASH 不能够在线实时做更改，设置断点等工作，这不能不是 ZLG 的一个缺陷。所以建议大家调试的时候，一定要重新编译过，再做调试。

编译完成后，打开 AXD 调试软件，并在 OPTION 中设置一些参数。



选择 JEDI 的配置选项，并点击“Configure”选项：



注意：用于仿真器连接开发板时候，需要做一个初始化的工作，否则 CPU 无法在调试模式下识别自身的硬件资源。我们建议大家手工写一个教本来给 AXD 调用，这里我们取名一个为 2200.INC 的教本范例文件，其具体源码如下（注意：因不同 EasyARM2200 的设计时间和硬件配置不同，请大家仔细参考 ZLG 的资料，我们只是在 LPC2210 CPU 上测试通过）

```
#2200.inc
#EasyARM2200 EVB Init File
#This file write by Tom Lang & Stephen Huang .ShenZhen

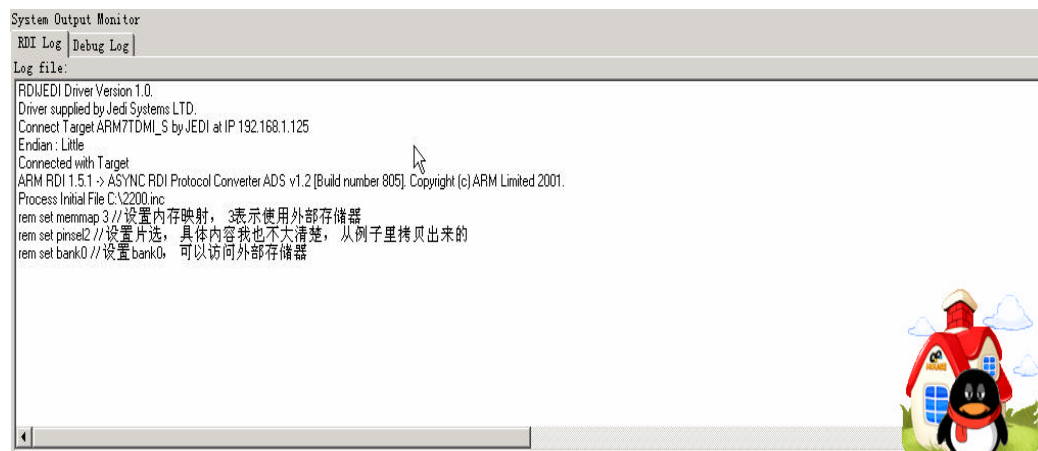
!
rem Set MEMMAP 3 // 设置内存映射， 3 表示使用外部存储器
long 0xE01FC040 0x3

rem Set PinSEL2 // 设置片选， 具体内容我也不大清楚， 从例子里拷贝出来的
long 0xe002c014 0x0f814914

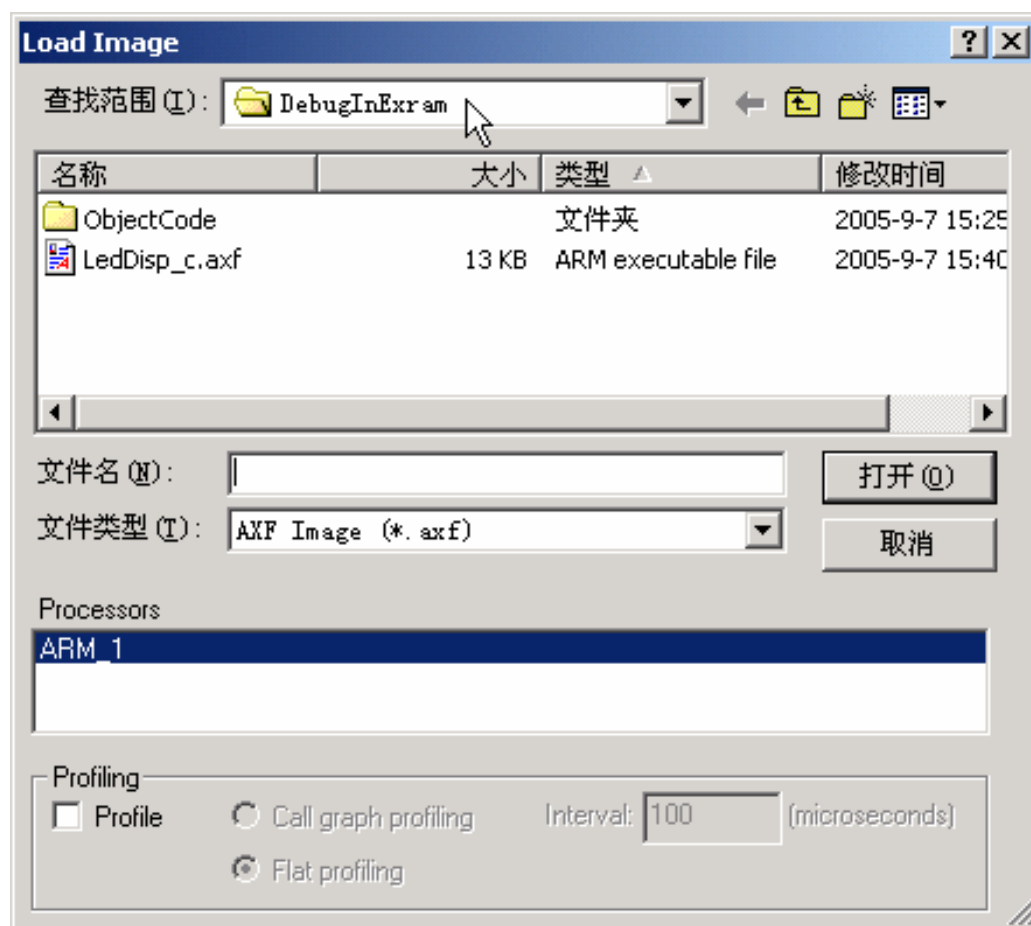
rem Set Bank0 // 设置 bank0， 可以访问外部存储器
long 0xFFE00000 0x1000ffef
!

#用 TXT 格式保存
```

点击‘OK’后，仿真器会在 ADS 的指令下对开发板进行初始化，并在指示窗口显示下列信息；



然后调用编译好的带.AXF 后缀格式的文件，

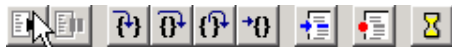


```

76     CODE32
77
78     AREA    vectors, CODE, READONLY
79     ENTRY
80     ;中断向量表
81     Reset
82     LDR     PC, ResetAddr
83     LDR     PC, UndefinedAddr
84     LDR     PC, SWI_Addr
85     LDR     PC, PrefetchAddr
86     LDR     PC, DataAbortAddr
87     DCD     0xb9205f80
88     LDR     PC, [PC, #-0xff0]
89     LDR     PC, FIQ_Addr
90
91     ResetAddr      DCD     ResetInit
92     UndefinedAddr  DCD     Undefined
93     SWI_Addr       DCD     SoftwareInterrupt
94     PrefetchAddr   DCD     PrefetchAbort
95     DataAbortAddr  DCD     DataAbort

```

这个时候我们看到程序停在 82 行的位置，这是一个 ARM 程序的起始代码，用的是 ASM 表示方式，如果要看到 C 源代码，则点击全速运行



则程序会停到 MAIN(VOID) 这个地方，ZLG 例子程序在这里设置了一个断点，从这里我们就可以正式看到 C 的源代码了。

```

52     }
53
54
55     const uint8 DISP_TAB[16] = { 0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,
56                                   0x81,0x42,0x24,0x18,0x24,0x42,0x81,0x00 };
57     /*****
58     * 名称: main()
59     * 功能: 根据表DISP_TAB来控制LED显示。
60     *****/
61     int main(void)
62     {
63         uint8 i;
64
65         PINSEL0 = 0x00000000;    // 设置管脚连接GPIO
66
67         IODIR = SPI_IODIR;      // 设置SPI控制口为输出
68         while(1)
69         {
70             for(i=0; i<16; i++)
71             {
72                 HCS95_SendDat(~DISP_TAB[i]); // 输出LED显示数据
73                 DelayNS(5);                // 延时
74             }
75         }
76     }

```

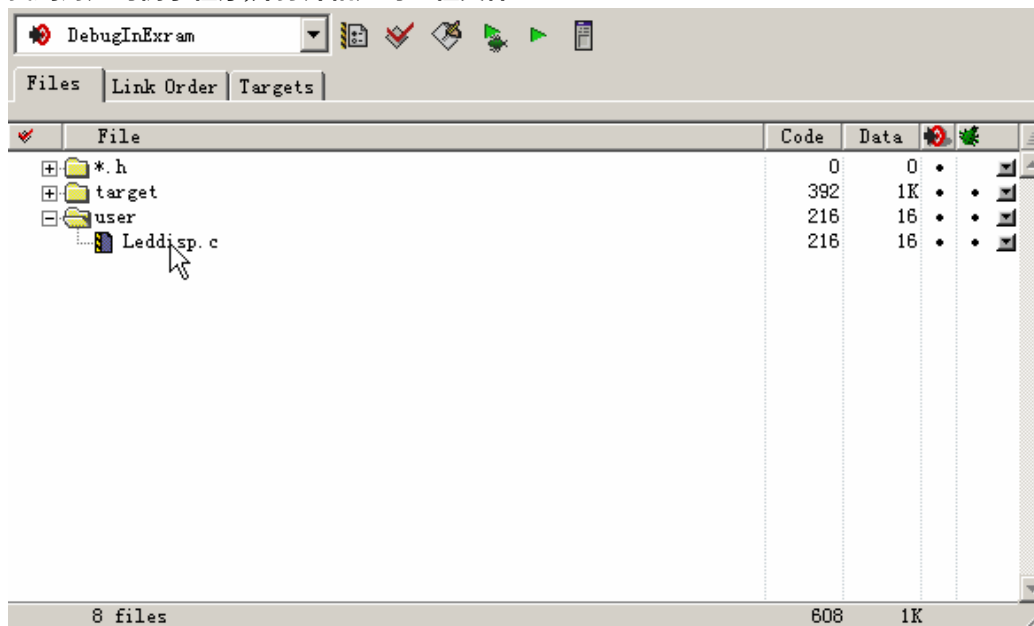


第六章 常见连接问题集锦

Q:为什么我调试全速运行的时候,LED 常亮而没有等顿啊?

A:这是因为程序设置的延长时间太短造成的.

找到对应的例子程序,并打开相应的工程文件.



```
Path: D:\ZKG_ADS\LedDisp_c\Leddisp.c

* 通过I/O模拟同步串口与74HC595进行连接，控制74HC595驱动LED显示。
* 说明：将跳线器JP8短接。
*****
#include "config.h"

#define SPI_CS 0x00000100 /* P0.8 */
#define SPI_DATA 0x00000040 /* P0.6 */
#define SPI_CLK 0x00000010 /* P0.4 */

#define SPI_IOCON 0x00000150 /* 定义SPI接口的I/O设置字 */

*****
* 名称: DelayNS()
* 功能: 长软件延时
* 入口参数: dly 延时参数，值越大，延时越久
* 出口参数: 无
*****
void DelayNS(uint32 dly)
{
    uint32 i;
    for(; dly>0; dly--)
        for(i=0; i<50000; i++);
}

*****
* 名称: HC595_SendDat()
* 功能: 向74HC595发送一字节数据
* 入口参数: dat 要发送的数据
* 出口参数: 无
* 说明: 发送数据时，高位先发送。
*****
void HC595_SendDat(uint8 dat)
{
    uint8 i;

    IO0CLR = SPI_CS; // SPI_CS = 0
    for(i=0; i<8; i++) // 发送8位数据
    {
        IO0CLR = SPI_CLK; // SPI_CLK = 0
        /* 设置SPI_DATA输出值 */
    }
}
```

Line 27 Col 18

修改参数

```
Path: D:\ZKG_ADS\LedDisp_c\Leddisp.c

* 通过I/O模拟同步串口与74HC595进行连接，控制74HC595驱动LED显示。
* 说明：将跳线器JP8短接。
*****
#include "config.h"

#define SPI_CS 0x00000100 /* P0.8 */
#define SPI_DATA 0x00000040 /* P0.6 */
#define SPI_CLK 0x00000010 /* P0.4 */

#define SPI_IOCON 0x00000150 /* 定义SPI接口的I/O设置字 */

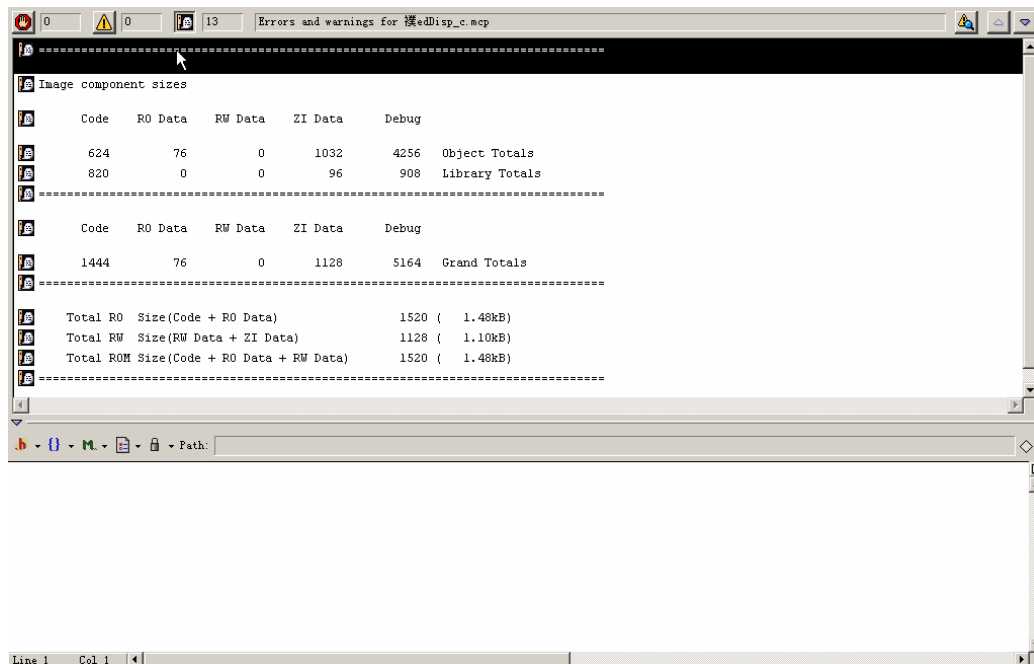
*****
* 名称: DelayNS()
* 功能: 长软件延时
* 入口参数: dly 延时参数，值越大，延时越久
* 出口参数: 无
*****
void DelayNS(uint32 dly)
{
    uint32 i;
    for(; dly>0; dly--)
        for(i=0; i<50000; i++);
}

*****
* 名称: HC595_SendDat()
* 功能: 向74HC595发送一字节数据
* 入口参数: dat 要发送的数据
* 出口参数: 无
* 说明: 发送数据时，高位先发送。
*****
void HC595_SendDat(uint8 dat)
{
    uint8 i;

    IO0CLR = SPI_CS; // SPI_CS = 0
    for(i=0; i<8; i++) // 发送8位数据
    {
        IO0CLR = SPI_CLK; // SPI_CLK = 0
        /* 设置SPI_DATA输出值 */
    }
}
```

Line 30 Col 1

重新编译一遍,并下载.



ObjectCode
LedDisp_c.axf

这个时候,你会发现程序会使得 LED 慢下来许多.

(待整理完善)