

英利 Linux 工控主板使用必读（EM928x）

感谢您选择英利 EM9280 系列工控主板。

英利 EM928X 工控主板包括多个型号：EM9280 和 EM9287。为便于读者了解和使用英利产品，本手册中以 EM9280 为例进行讲解；示例程序也以 EM9280 命名。然而，本手册和相关的示例程序也同样适用于其他这个系列的产品。

为了让您能够尽快地使用好我们的产品，英利公司编写了这篇《使用必读》，我们建议每一位使用英利产品的用户都浏览一遍。我们本着通俗易懂的原则，按照由浅入深的顺序，采用了大量图片和浅显的文字，以便于用户能边了解、边动手，轻松愉快地完成产品的开发。

在使用英利产品进行应用开发的过程中，如果您遇到任何困难需要帮助，都可以通过以下三种方式寻求英利工程师的技术支持：

- 1、直接致电 **028-86180660 85329360**
- 2、发送邮件到技术支持邮箱 support@emlinux.com
- 3、登录英利网站 www.emlinux.com，在技术论坛上直接提问

另，本手册以及其它相关技术文档、资料均可以通过英利网站下载。

注：英利公司将会不断完善本手册的相关技术内容，请客户适时从公司网站下载最新版本的手册，恕不另行通知。

再次感谢您的支持！

目 录

1	搭建硬件开发平台	3
1.1	EM9280 开发评估套件说明.....	3
1.2	必要的准备	3
1.3	开发环境的硬件连接和安装	4
2	配置软件开发环境	7
2.1	配置超级终端	7
2.2	编辑 userinfo.txt 文件	10
2.3	安装 eclipse.....	12
2.4	设置文件系统挂载	16
3	开发自己的应用程序.....	19
3.1	创建工程文件 hello.....	19
3.2	打开已有的工程文件 wr.....	25

1 搭建硬件开发平台

1.1 EM9280 开发评估套件说明

用户第一次使用 EM9280 往往是购买开发评估套件，开发评估套件包括如下几部分：

- **EM9280 工控主板一块：**Freescale iMX283 处理器，主频 454MHz，预装嵌入式 Linux-2.6 实时多任务操作系统，接口资源丰富
- **EM9280 开发评估底板一块：**搭载 EM9280 并引出其板载资源。底板上提供了 EM9280 所有板载资源的标准接口，既方便用户对 EM9280 进行评估和开发，又为用户的外围硬件开发提供一定的参考
- **串口连接线一条：**3 线制串口连接线，用于输出调试信息
- **以太网连接线一条：**直连方式，用于进行目标机系统的管理维护以及开发网络方面的应用功能
- **USB 连接线一条：**A-B 连接线，用于进行系统内核烧写
- **直流电源线一条：**红黑双色，+5V，用于为系统供电
- **开发资料光盘一张：**为用户的开发提供丰富翔实的软硬件资料

1.2 必要的准备

用户要利用 EM9280 进行开发，需要作如下一些必要准备：

- 准备一台具有+5V 电压输出的普通直流稳压电源或开关直流电源（+5V±5%），将英利提供的直流电源线正确地连接到该电源的+5V 输出上（注意极性）。

注：如用户要进行 GPRS/CDMA 应用开发，由于 GPRS/CDMA 通讯模块发射的瞬间电流可达 2A，建议在选择开关电源时选输出功率在 10W 以上的。

- 准备一台带以太网接口、USB 接口的 PC 机作为开发主机，该 PC 机安装 Windows XP、Windows 7 操作系统或 Linux 操作系统。

注：1、调试串口可以使用 usb 转串口模块进行转接，然而，我们建议客户尽量使用带有物理串口的 PC 机作为开发主机。

2、如果用户在主机上使用 Linux 操作系统进行开发，由于 Linux 的开源和自

由性，以及市面和互联网上已具备的丰富翔实的参考资料和各种 Linux 社区资源，英利公司不再对 Linux 环境下的开发过程进行技术支持，特此说明。

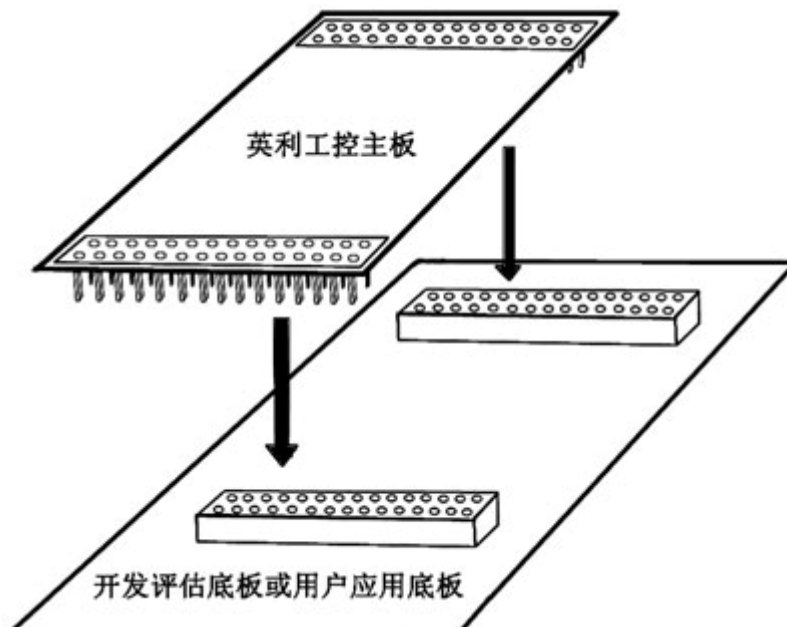
- 准备一台网络连接设备（集线器/交换机/路由器）。
- 准备一只可供临时存储数据的 U 盘。

1.3 开发环境的硬件连接和安装

在以上条件准备好以后，就可以按照如下顺序进行开发环境的硬件连接了。

1、EM9280 两侧有双排插针（CN1 和 CN2），这两排插针将 EM9280 的板载接口资源引出，而开发评估底板上安装有相对应的两个插座（EM9280_CN1 和 EM9280_CN2），EM9280 就象一个大芯片一样对插在开发评估底板上，从而构成一套较完整的开发系统，如下图所示。

注：在用户收到的开发评估套件中，EM9280 往往已经插在底板上，开发过程中用户如需进行插拔，请注意插针和插座的序号对应。



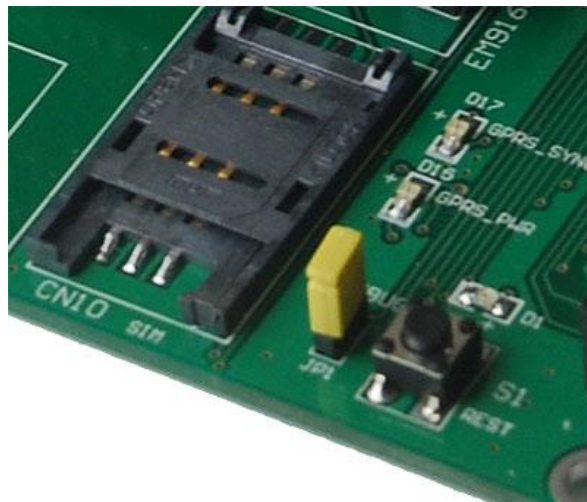
英利工控主板与开发评估底板的连接关系

2、EM9280 有两种工作模式：调试模式和运行模式。

调试模式是指开机以后系统处于调试状态，此时用户可以通过超级终端来操作 EM9280，实现应用程序下载调试、文件管理等功能。在开发阶段，系统总是处于这种状态。

运行模式是指开机以后系统自动开始执行用户指定的程序。开发完成，进入实际应用时系统总是处于这种状态。

EM9280 工作于上述的哪一种模式，是通过开发评估底板上的跳线器 JP1 来选择的(JP1 在开发评估底板上的具体位置见下图)。JP1 短接，则工作于调试模式；JP1 断开，则工作于运行模式。



工作模式选择跳线器 JP1

3、将套件中串口连接线的两端分别接入开发主机的串口和 EM9280 开发评估底板的控制台串口，如下图所示。



连接调试串口

4、用户可以用交换机/路由器/集线器将主机和 EM9280 接入同一个网络，如下图所示。这样开发主机和 EM9280 就能够建立起网络连接。

注：EM9280 的 ip 地址一定要与开发主机的 ip 地址设置在同一网段内。



将开发主机和 EM9280 接入以太网

至此，EM9280 运行的基本硬件环境已搭建完成。

注：USB 连接线仅仅是在英利更新了操作系统内核，需要用户自行烧写的时候才使用。

具体的内核烧写方法请参阅光盘中的内核烧写说明文档。

现在可以给 EM9280 通电，即将+5V 直流电源线接头插在底板上的电源插头（注意正负方向）里，此时，EM9280 上的红色电源 LED 指示灯亮。

2 配置软件开发环境

EM9280 板载嵌入式 Linux-2.6 实时多任务操作系统，用户可以在主机使用 Windows 或者 Linux 操作系统进行应用程序的开发。鉴于 Windows 操作系统的广泛使用，为使用户快速、便捷地开发出自己的应用程序，减少学习 Linux 所需的时间和精力，英利公司进行了大量富有成效的工作，最终选取了一系列可以在 Windows 操作系统中开发 Linux 应用程序，并将程序下载到 EM9280 中运行测试的工具，下面将逐一介绍这些工具的安装、使用方法以及相关事宜，用户跟随本章的步骤即可快速搭建起 EM9280 的软件开发平台。

2.1 配置超级终端

EM9280 的运行信息会通过超级终端显示在开发主机的显示屏上；用户想要对 EM9280 的文件系统进行操作也需通过超级终端以命令行方式进行。

1、通过开始菜单打开超级终端，如下图所示。



启动超级终端

2、如下图所示新建连接。本文中以 **emlinux** 为例，用户可自行设置。



新建连接

3、新建一个串口连接，使用 **COM1**，如下图所示。



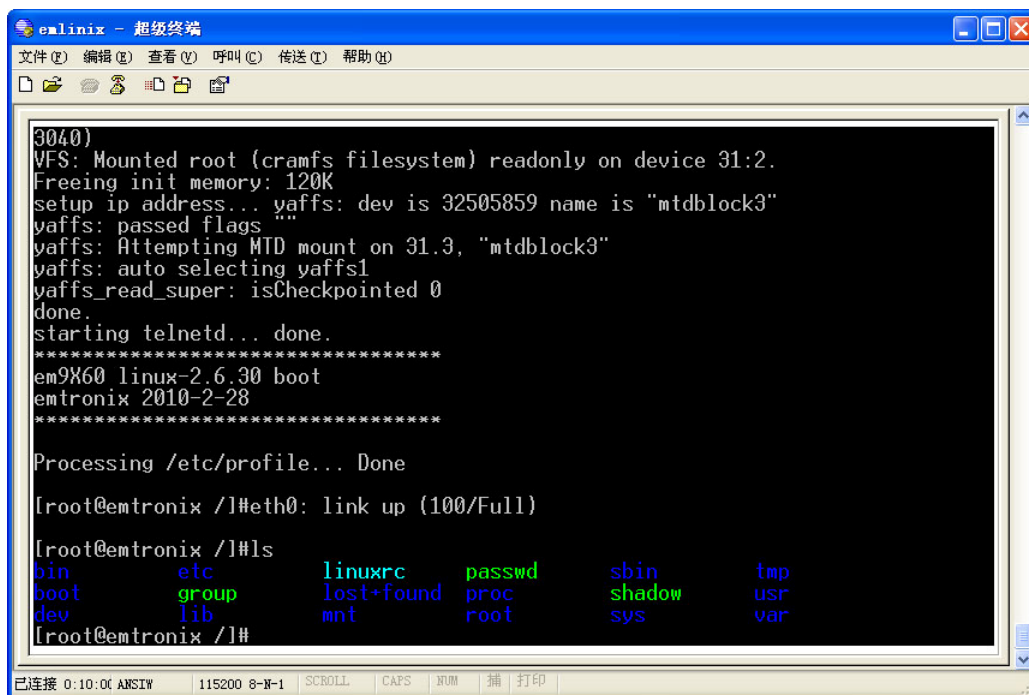
新建连接 COM1

4、如下图所示，设置每秒位数为 **115200**，数据流控制为**无**。



COM1 参数配置

5、完成以后给 EM9280 上电，超级终端将显示出 EM9280 的开机启动信息。启动成功以后回车进入命令行，此时可以通过超级终端使用 Linux 的命令对 EM9280 进行操作，如下图所示。图中显示的是使用 **ls** 命令查看 EM9280 中的文件。



通过超级终端使用 Linux 的命令操作 EM9280

2.2 编辑 userinfo.txt 文件

userinfo.txt 文件有三个作用：

- 1、配置 EM9280 的网络参数，让 EM9280 与开发主机处于同一网段
- 2、配置 NFS 挂载参数，让开发主机的指定目录能挂载到 EM9280 的指定目录下
- 3、配置应用程序参数，这样开发完成以后 EM9280 将自动根据该参数执行应用程序

userinfo.txt 文件的内容及格式如下（双斜线后不同字体和颜色的文字为加注的说明文字，并不包括在 userinfo.txt 文件中）：

```
[LOCAL_MACHINE]           // EM9280 信息

DHCP="0"                   // 配置 DHCP 客户端信息。设为“0”则 DHCP 关
                           // 闭，用户需手动设置网关、IP 地址、子网掩码；
                           // 设为“1”则 DHCP 开启，EM9280 将自行获取
                           // 上述网络参数

DefaultGateway="192.168.201.19" // 默认网关，根据用户所在的实际运行网络设置

IPAddress="192.168.201.175"   // EM9280 的 IP 地址，由用户自行设置

SubnetMask="255.255.255.0"   // 子网掩码，根据用户所在的实际运行网络填写

[NFS_SERVER]               // NFS 挂载信息

IPAddress="192.168.201.170"  // 开发主机 IP 地址，根据用户所在的实际运行
                           // 网络设置

Mountpath="/d/public"       // 开发主机上被挂载的文件夹名，本文中以
                           // “public”为例，用户可自行选择任意文件夹，
                           // 需注意的是必须带上文件夹路径

[USER_EXE]                 // 用户程序信息

Name="/mnt/nandflash/hello" // 系统开机自动执行的程序及其存储路径。开发
                           // 完成以后用户将自己的应用程序文件名填在
                           // 双引号之间取代目前的默认文件名，开机即可
```

```
Parameters=""
// 自动运行（注意，用户也可以在
// /mnt/nandflash/下建立子目录存放应用程序，
// 配置此项参数的时候一定要带上绝对路径）
// 系统开机自动执行的程序的参数配置。开发完
// 成以后在此处填入实际应用程序的参数，如果
// 没有则不填，但必须保留双引号
```

根据用户的实际网络参数编辑好 `userinfo.txt`，存入 U 盘，将 U 盘接入 EM9280 开发评估底板的 USB 接口，短接 JP1 使 EM9280 处于调试模式，然后上电。系统将自动搜索 USB 接口，将读到的 `userinfo.txt` 文件存放到 `/mnt/nandflash` 目录中，并按照其内容配置 EM9280 的网络参数。启动完成以后，可以通过超级终端使用 `ifconfig` 命令查看是否配置完成。

`userinfo.txt` 写入 EM9280 以后，系统每次开机都会自动读取该文件并按照文件内容进行配置。如果其中任何参数需要重新配置，可编辑好 `userinfo.txt` 并重复执行上述步骤。

如果要让系统开机自动挂载，则 EM9280 上电启动之前必须先打开 `WinNFSD.exe`。

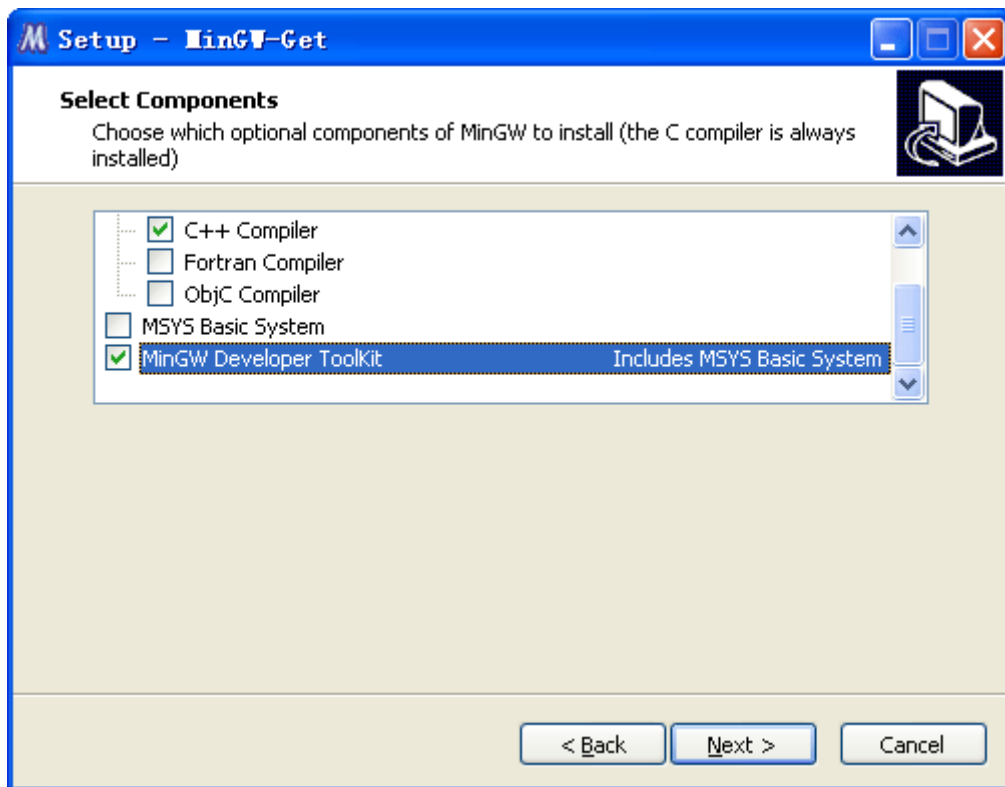
如果 EM9280 处于运行模式，则开机以后会自动执行 `Name="/mnt/nandflash/"` 中设置的应用程序。英利为用户分配的存储地址固定在 `/mnt/nandflash` 文件夹下，用户可以将应用程序直接存在这个目录中，也可以在此目录下建立子目录存放应用程序。用户配置该项参数的时候要带上绝对路径，否则系统无法找到执行文件。

注：Linux 操作系统严格区分大小写，因此此处的用户应用程序名称必须与实际的程序名称完全一样，包括大小写字母。

2.3 安装 eclipse

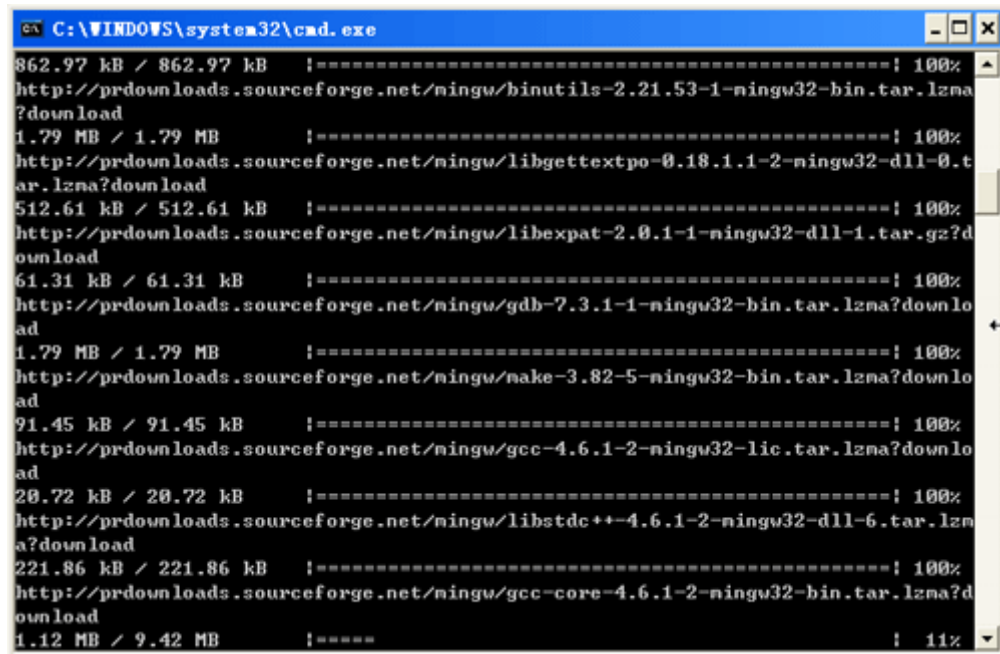
eclipse 是一款开源的免费开发工具，可以直接在 Windows 操作系统下生成 Linux 应用程序，省却用户学习使用 Linux 开发工具所需的大量精力和时间。经英利公司的努力工作，eclipse 编译的程序已经可以在英利主板上直接运行。此外，eclipse 的开发环境和 Visual Studio 等常用开发工具很相似，用户可以通过市面上很常见的 C 语言书籍以及 Linux 书籍来进行编程方面的学习。

1、在开发光盘的“工具软件”中找到“EclipseOnWindows”文件夹，安装 mingw-get-inst-20111118。如下图所示，在 **Select Components** 对话框中选择 **C Compiler**（该项默认已选中）、**C++ Compiler** 和 **MinGW Developer ToolKit** 三项，其他所有配置均使用系统默认设置。



安装 MinGW

MinGW 安装完成以后，会自动从网络下载一些库文件，如下图所示。等下载完毕以后，再继续后续软件的安装。



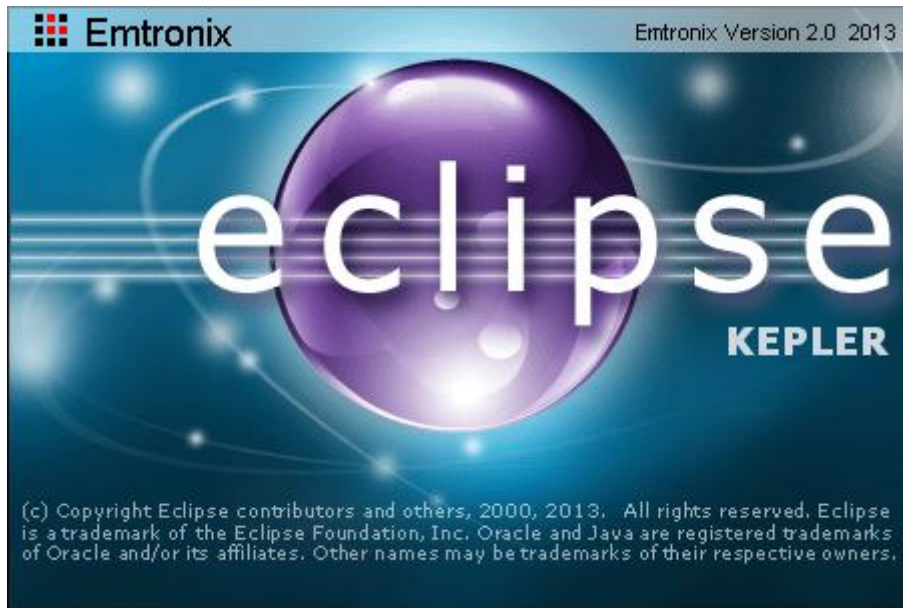
MinGW 自动下载文件

2、MinGW 安装完成以后，安装 jre-7u2-windows-i586，所有配置均使用系统默认设置。

3、jre 安装完成以后，安装 arm-2009q3-67-arm-none-linux-gnueabi，所有设置全部采用系统默认设置。

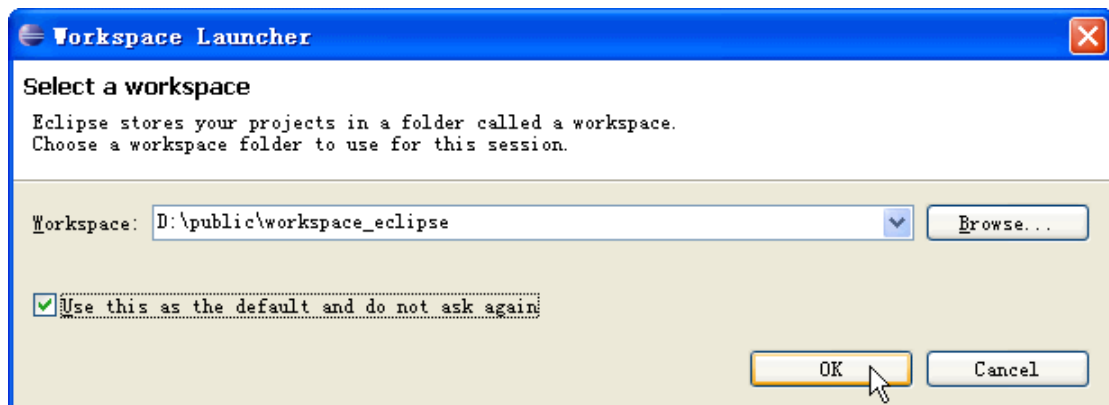
4、G++安装完成以后，将“eclipse”文件夹复制到开发主机任意目录（路径中不能有中文，建议放到 C 盘根目录下）。

5、完成以后进入 eclipse 文件夹，启动软件，用户可以看到加载了英创公司信息的启动画面，如下图所示。



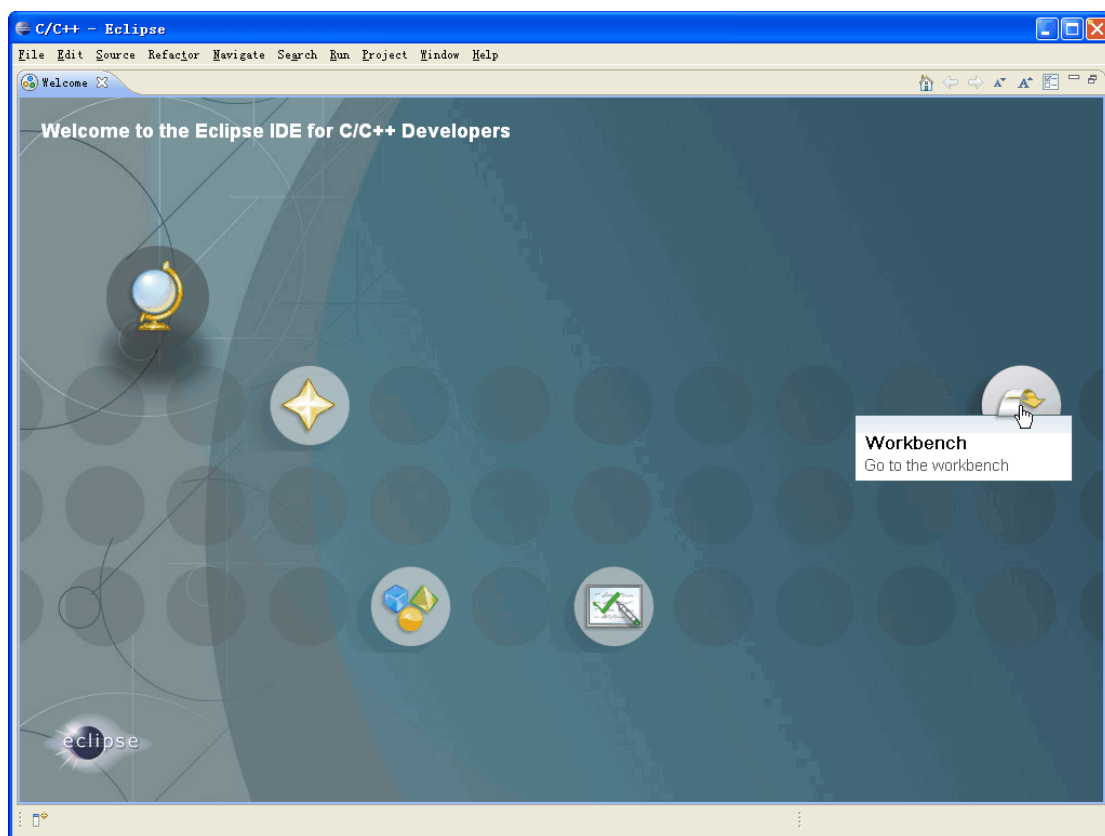
加载了英创公司信息的 eclipse 启动画面

6、出现如下图所示对话框，指定工程文件的默认保存路径，此后所有使用eclipse设计的文件将自动保存到该文件夹下。本文中以D:\public\workspace_eclipse为例，用户可自行指定，但是注意路径中不要带有中文名。



选择工程文件保存路径

7、启动以后，进入 Workbench，如下图所示。



进入 Workbench

至此，eclipse 开发工具安装完成。

2.4 设置文件系统挂载

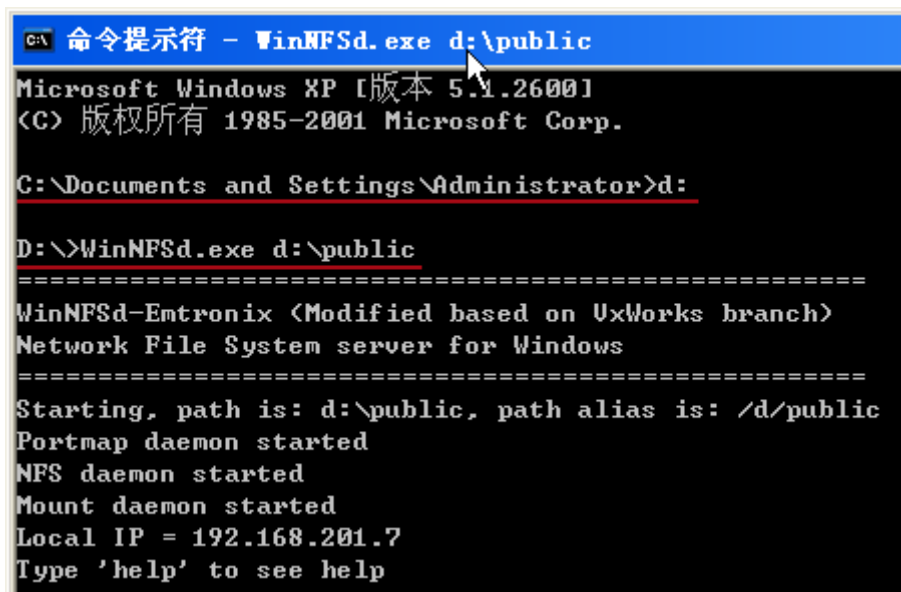
用户在开发主机中完成的应用程序必须通过一定的方法下载到 EM9280 的存储器中，才能进行运行测试。这种文件复制的方法有很多，英利公司建议使用文件系统挂载，此方法可以将开发主机中用户指定的某一个目录挂载到 EM9280 的 Linux 目录中，这样，用户在开发主机中完成的应用程序就可以直接放在该目录下，然后通过超级终端让其在 EM9280 上进行运行测试。

英创为用户提供的 NFS 服务器是一款名为 WinNFSd.exe 的免费 NFS 服务器，只需使用命令提示符打开和启动该服务器，即可以使用 NFS 功能。

1、打开开发资料光盘上的“工具软件”文件夹，将“WinNFSd.exe”复制到任意路径（路径不能带有中文名，本文以复制到 D 盘根目录为例）。

2、打开命令提示符，进入 D 盘，启动 WinNFSd.exe，如下图所示（此处的挂载文件夹以 d:\public 为例，用户可自行设置）。

挂载文件夹一定要与 userinfo.txt 中 NFS SERVER 项参数“Mountpath”中的配置一致。关于 userinfo 的编辑方法参见 2.2 节。



```
C:\> 命令提示符 - WinNFSd.exe d:\public
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

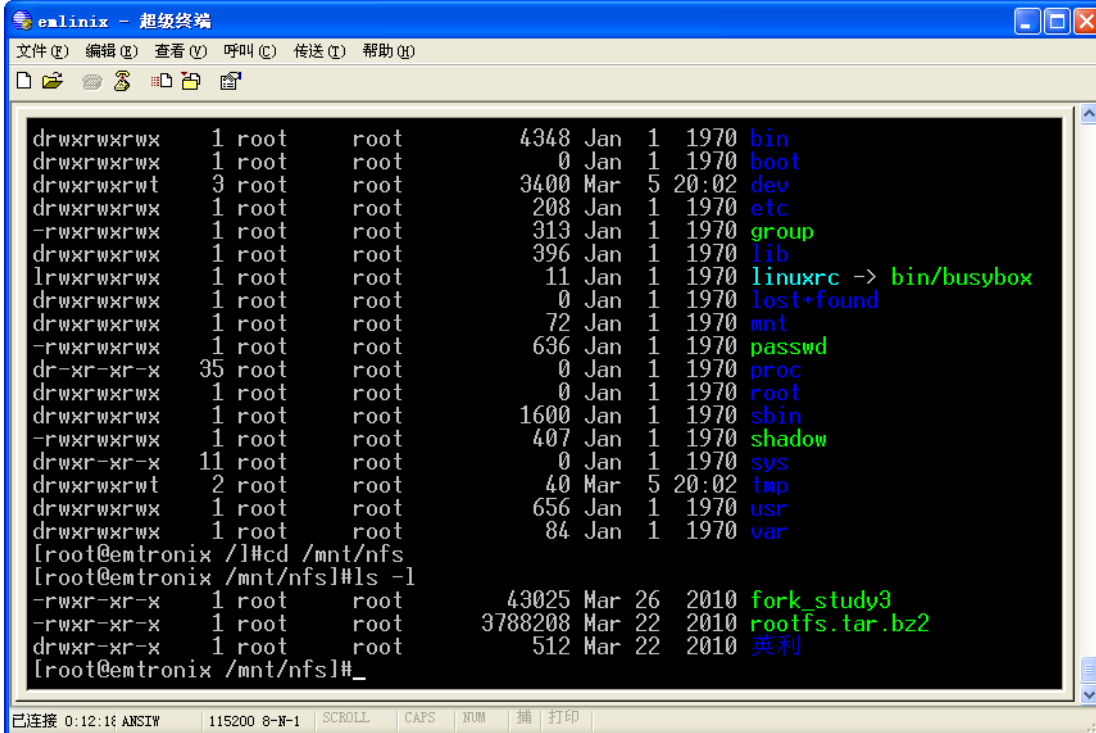
C:\Documents and Settings\Administrator>d:

D:\>WinNFSd.exe d:\public
=====
WinNFSd-Emtronix <Modified based on UxWorks branch>
Network File System server for Windows
=====
Starting, path is: d:\public, path alias is: /d/public
Portmap daemon started
NFS daemon started
Mount daemon started
Local IP = 192.168.201.7
Type 'help' to see help
```

启动 WinNFSd.exe

3、确认 `userinfo.txt` 文件已配置好并存入 U 盘，将 U 盘接在工控主板的 USB 接口上，然后为系统上电。英利在 EM9280 上为开发主机指定的挂载点是 `/mnt/nfs`，因此，在超级终端中使用命令 `cd /mnt/nfs` 进入 nfs 文件夹，使用命令 `ls -l` 查看，可以看到开发主机上 `public` 文件夹下的内容，如下图所示，表示挂载成功。

注：EM9280 上电启动之前必须先启动 WinNFSd.exe，才能自动挂载。



```

drwxrwxrwx 1 root root 4348 Jan 1 1970 bin
drwxrwxrwx 1 root root 0 Jan 1 1970 boot
drwxrwxrwt 3 root root 3400 Mar 5 20:02 dev
drwxrwxrwx 1 root root 208 Jan 1 1970 etc
-rwxrwxrwx 1 root root 313 Jan 1 1970 group
drwxrwxrwx 1 root root 396 Jan 1 1970 lib
lrwxrwxrwx 1 root root 11 Jan 1 1970 linuxrc -> bin/busybox
drwxrwxrwx 1 root root 0 Jan 1 1970 lost+found
drwxrwxrwx 1 root root 72 Jan 1 1970 mnt
-rwxrwxrwx 1 root root 636 Jan 1 1970 passwd
dr-xr-xr-x 35 root root 0 Jan 1 1970 proc
drwxrwxrwx 1 root root 0 Jan 1 1970 root
drwxrwxrwx 1 root root 1600 Jan 1 1970 sbin
-rwxrwxrwx 1 root root 407 Jan 1 1970 shadow
drwxr-xr-x 11 root root 0 Jan 1 1970 sys
drwxrwxrwt 2 root root 40 Mar 5 20:02 tmp
drwxrwxrwx 1 root root 656 Jan 1 1970 usr
drwxrwxrwx 1 root root 84 Jan 1 1970 var

[root@emtronix /]#cd /mnt/nfs
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#_

```

查看挂载到 Linux 目录下开发主机中的文件夹

4、如果开机挂载没有成功或者使用中连接中断，建议重启 WinNFSd.exe，然后键入命令进行挂载：

`mount -t nfs -o nolock 192.168.201.7:/d/public /mnt/nfs`

上述命令中的红字部分仅为示例，用户应填写自己实际的开发主机 IP 地址和挂载文件夹目录。

5、如果挂载仍然失败，建议同时重启工控主板和 WinNFSd.exe，然后再次测试。需注意的是，应确认该服务器没有被杀毒软件或者 Windows 自带的防火墙阻止。

Windows XP 防火墙中显示如下：



Windows 7 中防火墙显示如下：

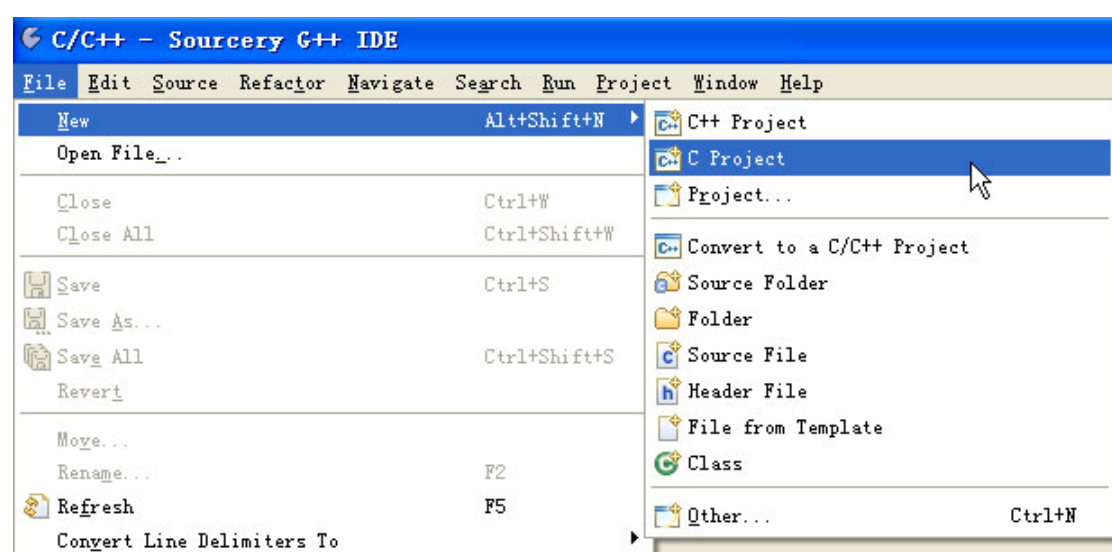


3 开发自己的应用程序

经过前两章，EM9280 的软硬件开发环境搭建均已完成，接下来用户可以进行应用程序的开发了。本章将通过两个实例介绍 EM9280 的软件应用开发步骤。

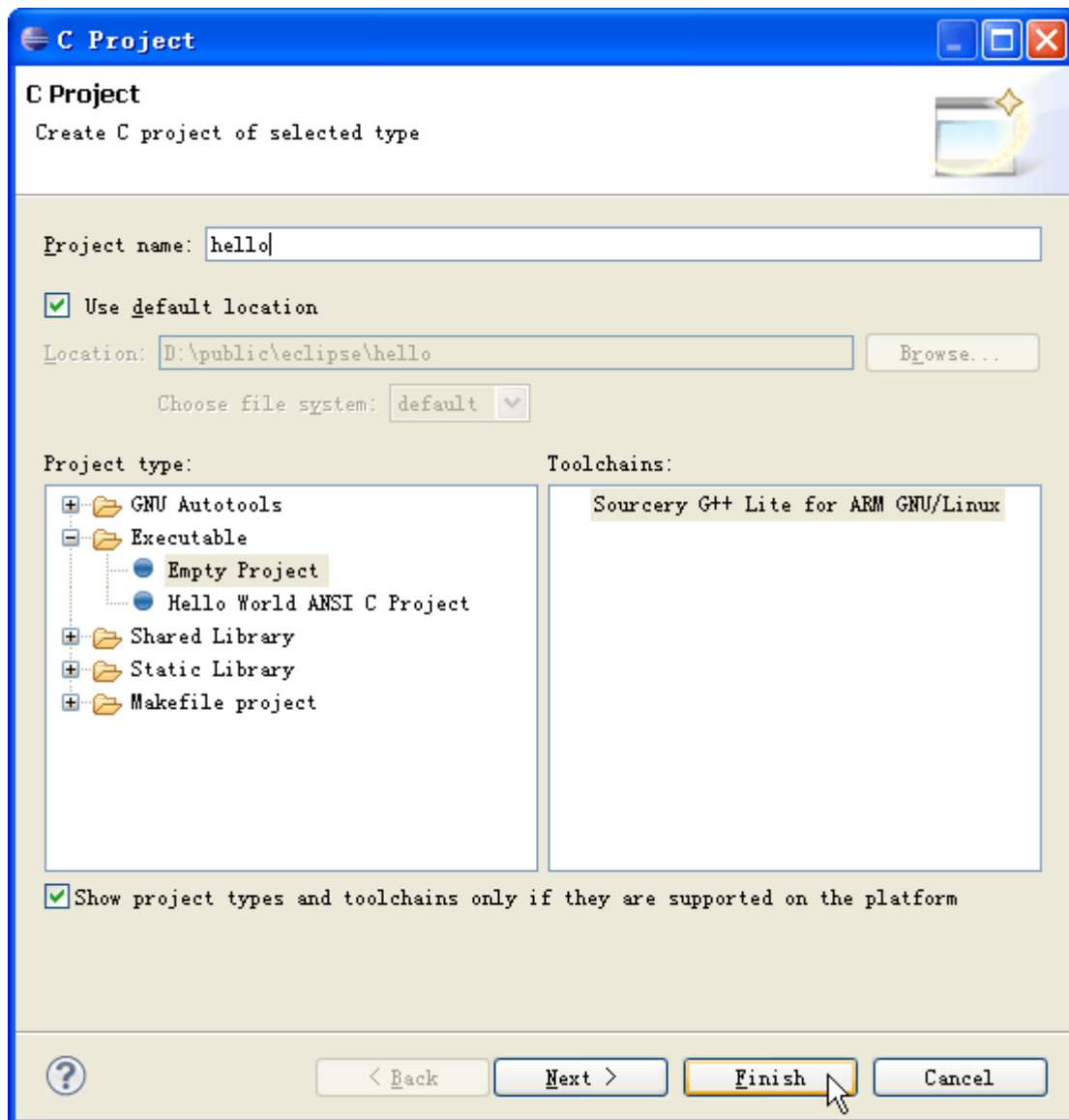
3.1 创建工程文件 hello

1、启动 eclipse / Sourcery G++，选择 **File -> New -> C Project**，如下图所示。



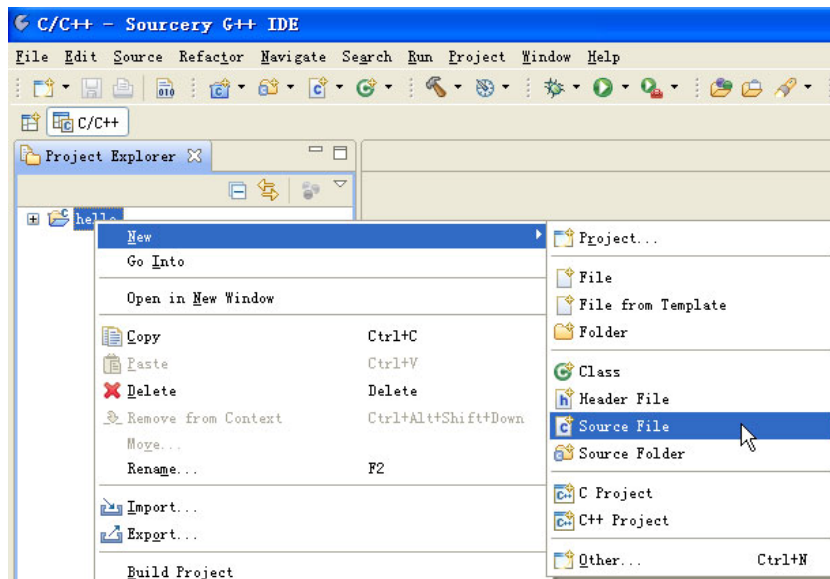
建立工程文件

2、在 **Project name:**中填入 **hello**，然后选择 **Finish** 完成工程建立，如下图所示。



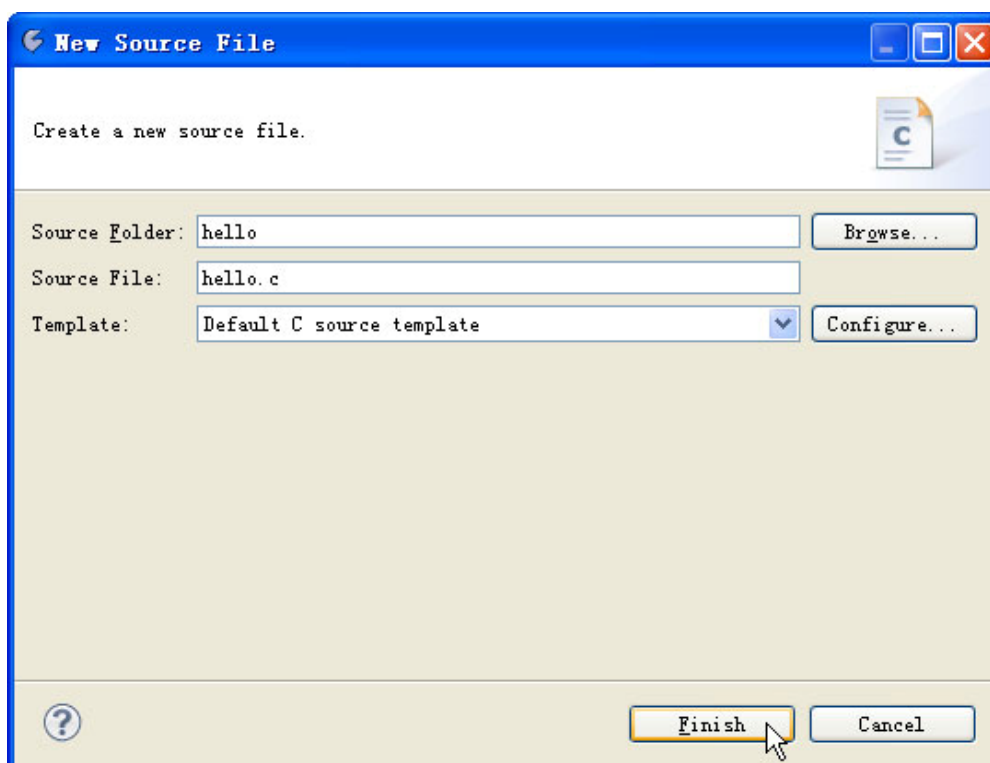
选择 Toolchains

3、在 Project Explorer 中建好的工程文件名 **hello** 上点右键选择 **New -> Source File**, 如下图所示。



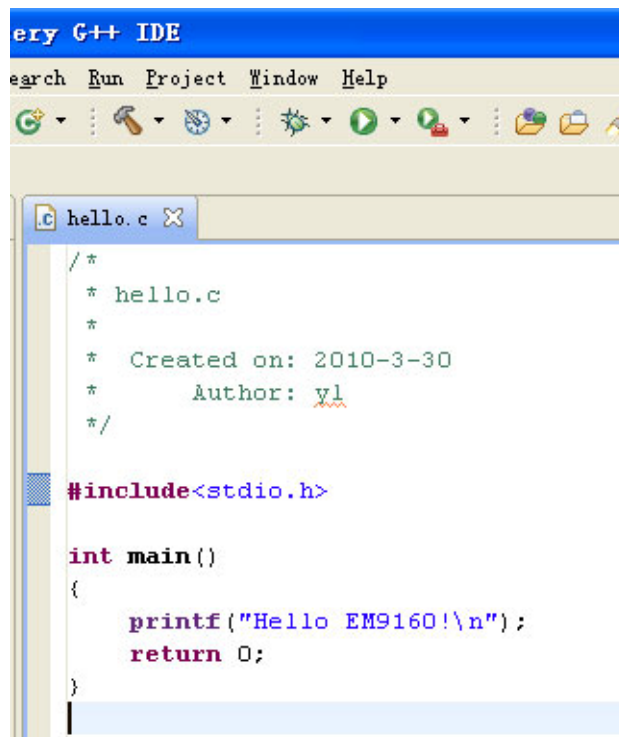
建立源程序文件

4、源程序文件命名为 **hello.c**, 如下图所示。



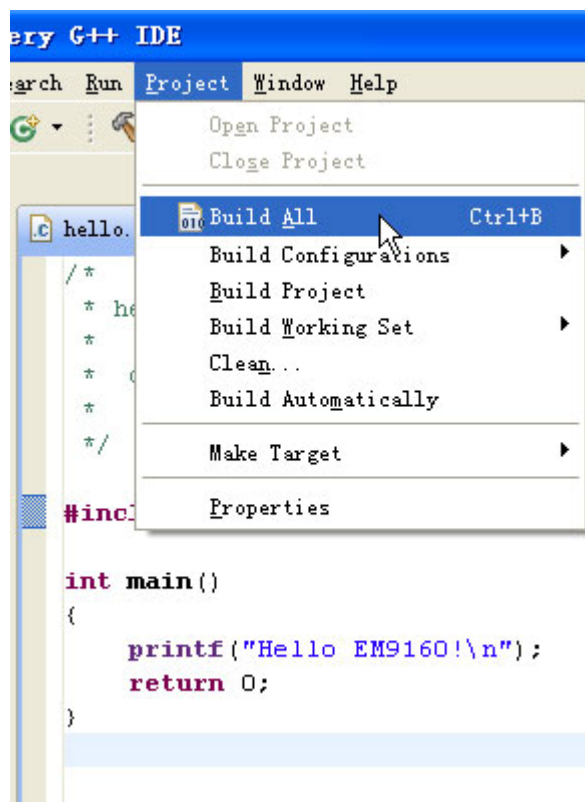
为源程序文件命名

5、系统自动建立源程序文件，输入程序源码并保存，如下图所示。



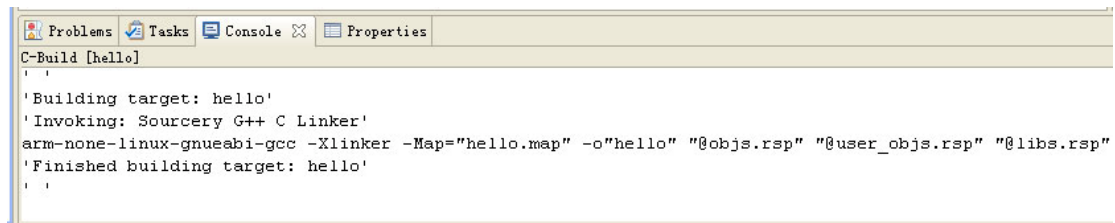
输入程序源码

6、选择 **Project -> Build All** 编译源码，如下图所示。



编译源码

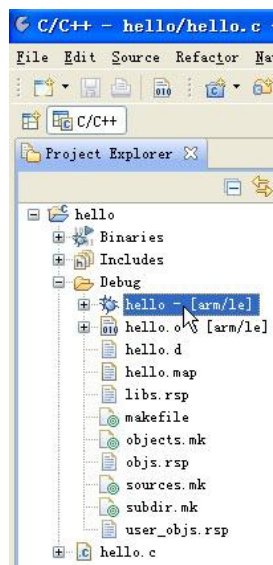
7、编译完成，底部区域会显示出编译信息，如下图所示。如果程序有错误，也会在底部区域提示。



```
Problems Tasks Console Properties
C-Build [hello]
'
'
'Building target: hello'
'Invoking: Sourcery G++ C Linker'
arm-none-linux-gnueabi-gcc -Xlinker -Map="hello.map" -o"hello" "@objs.rsp" "@user_objs.rsp" "@libs.rsp"
'Finished building target: hello'
'
'
```

编译信息

此时，通过左侧 Project Explorer 可以看到已经生成应用程序 hello，如下图所示。



生成应用程序

8、工程文件默认保存地址是 **D:\public\workspace_eclipse**；而挂载到 EM9280 下的开发主机文件夹为 **D:\public**，在超级终端通过如下命令进入工程文件夹：

```
cd /mnt/nfs/workspace_eclipse/hello/Debug
```

9、通过超级终端使用命令 **ls -l** 查看挂载文件夹，可以看到 **hello**，如下图所示。

```

emlinux - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

yaffs_read_super: isCheckpointed 0
done.
starting telnetd... done.
*****
em9X60 linux-2.6.30 boot
emtronix 2010-2-28
*****

Processing /etc/profile... Done

[root@emtronix /]#eth0: link up (100/Full)

[root@emtronix /]#mount -t nfs -o nolock 192.168.201.190:public /mnt/nfs
[root@emtronix /]#cd /mnt/nfs
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#
已连接 0:30:02 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

```

通过超级终端查看挂载文件夹

10、使用命令 **./hello** 执行应用程序，运行成功，终端显示出打印结果，如下图所示。

```

emlinux - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

drwxrwxrwx 1 root root 208 Jan 1 1970 etc
-rwxrwxrwx 1 root root 313 Jan 1 1970 group
drwxrwxrwx 1 root root 396 Jan 1 1970 lib
lrwxrwxrwx 1 root root 11 Jan 1 1970 linuxrc -> bin/busybox
drwxrwxrwx 1 root root 0 Jan 1 1970 lost+found
drwxrwxrwx 1 root root 72 Jan 1 1970 mnt
-rwxrwxrwx 1 root root 636 Jan 1 1970 passwd
dr-xr-xr-x 35 root root 0 Jan 1 1970 proc
drwxrwxrwx 1 root root 0 Jan 1 1970 root
drwxrwxrwx 1 root root 1600 Jan 1 1970 sbin
-rwxrwxrwx 1 root root 407 Jan 1 1970 shadow
drwxr-xr-x 11 root root 0 Jan 1 1970 sys
drwxrwxrwt 2 root root 40 Mar 6 01:28 tmp
drwxrwxrwx 1 root root 656 Jan 1 1970 usr
drwxrwxrwx 1 root root 84 Jan 1 1970 var
[root@emtronix /]#cd /mnt/nfs
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#./hello
Hello EM9160!
[root@emtronix /mnt/nfs]#
已连接 0:33:04 ANSIW 115200 8-N-1 SCROLL CAPS NUM 捕 打印

```

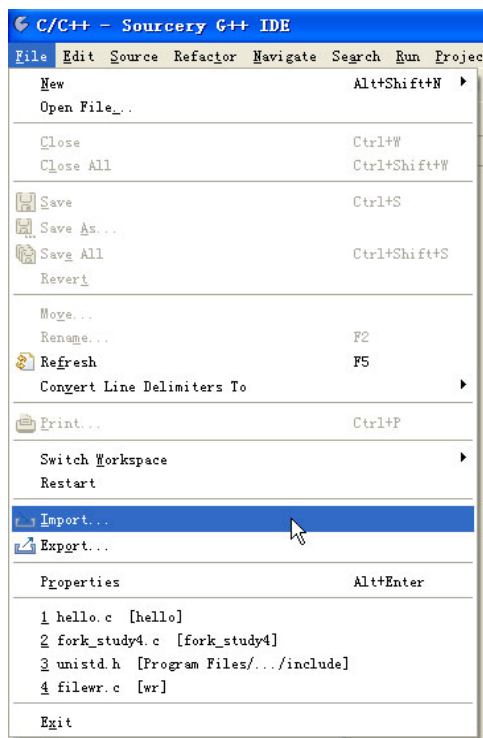
执行应用程序

这样，用户就完成了从工程文件建立到程序运行测试的一个完整过程。

3.2 打开已有的工程文件 wr

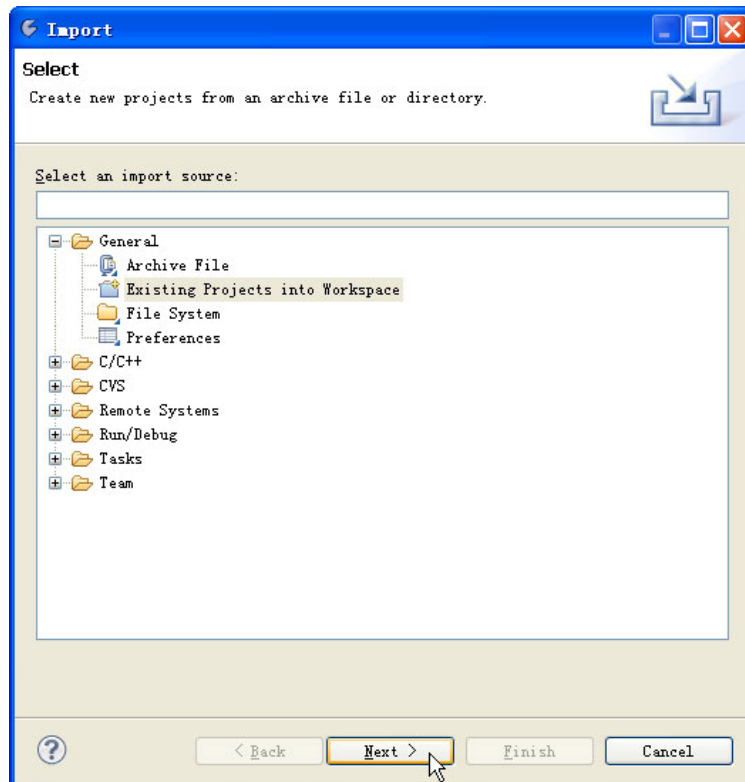
1、在英利开发光盘中找到 wr 文件夹，复制到开发主机任意地址。

2、启动 eclipse，选择 **File -> Import...**，如下图所示。



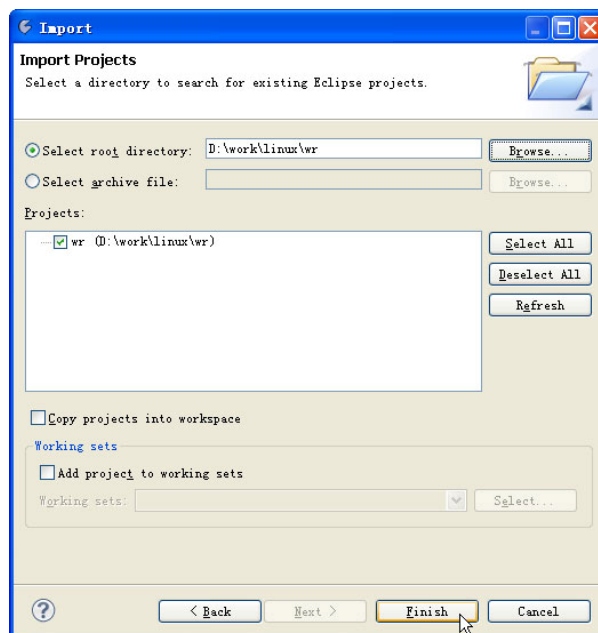
载入工程文件

3、选择 **General -> Existing Projects into Workspace**，如下图所示。



载入已存在的工程文件

4、通过 **Select root directory:** 右边的 **Browse...** 按钮找到之前复制到开发主机中的 **wr** 文件夹，其余选项保持默认设置，如下图所示。



找到wr文件夹

5、打开工程文件以后，如3.1节的步骤，进行编译，然后通过超级终端进入工程文件夹。这个程序是要打开一个名为“read.txt”的文件，将其内容读出然后复制到另一个名为“write.txt”的文件中，因此需将read.txt（这个文件放在wr\Debug中）也复制到D:\public下。之后通过超级终端查看该文件夹，可以看到wr和read.txt，如下图所示。

```

-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#_

```

通过超级终端查看复制进来的两个文件

6、使用命令./wr执行应用程序，运行成功，终端显示出打印结果，如下图所示。

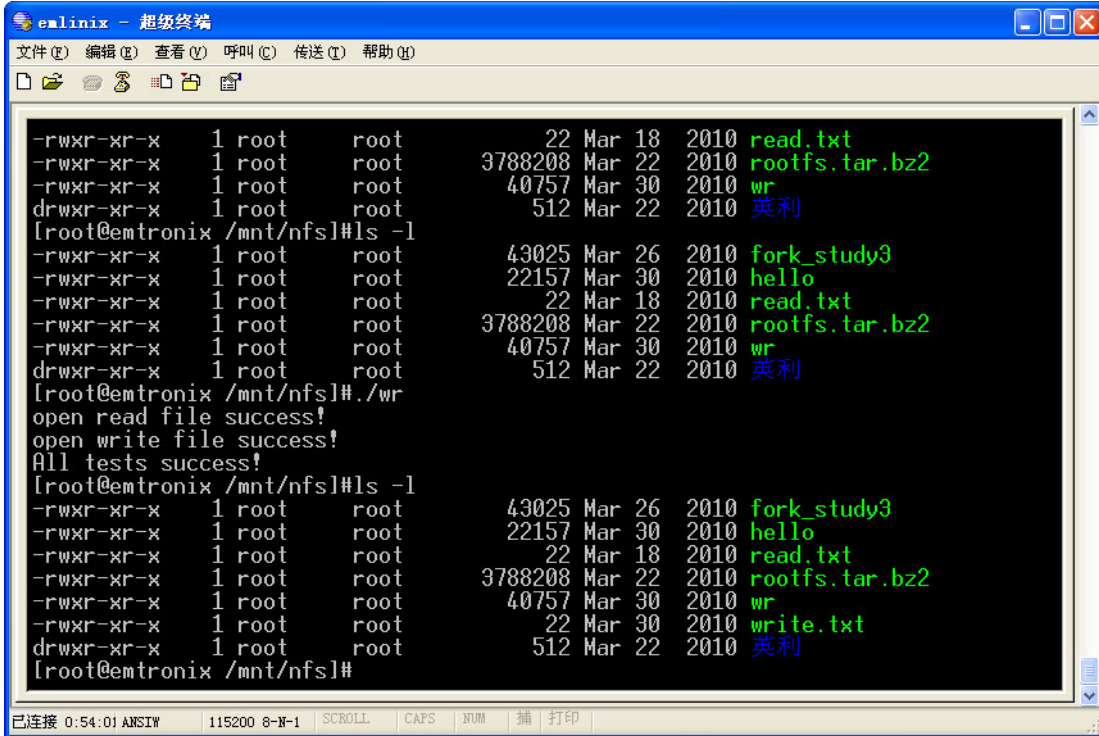
```

-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x 1 root root 43025 Mar 26 2010 fork_study3
-rwxr-xr-x 1 root root 22157 Mar 30 2010 hello
-rwxr-xr-x 1 root root 22 Mar 18 2010 read.txt
-rwxr-xr-x 1 root root 3788208 Mar 22 2010 rootfs.tar.bz2
-rwxr-xr-x 1 root root 40757 Mar 30 2010 wr
drwxr-xr-x 1 root root 512 Mar 22 2010 英利
[root@emtronix /mnt/nfs]#./wr
open read file success!
open write file success!
All tests success!
[root@emtronix /mnt/nfs]#_

```

执行应用程序

7、再次使用**ls -l**命令查看，程序已经新建了**write.txt**文件，如下图所示。在开发主机中打开该文件和**read.txt**对比，确认内容一致，表明程序运行成功。



```

-rwxr-xr-x  1 root  root      22 Mar 18  2010 read.txt
-rwxr-xr-x  1 root  root  3788208 Mar 22  2010 rootfs.tar.bz2
-rwxr-xr-x  1 root  root  40757 Mar 30  2010 wr
drwxr-xr-x  1 root  root    512 Mar 22  2010 英利
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x  1 root  root    43025 Mar 26  2010 fork_study3
-rwxr-xr-x  1 root  root   22157 Mar 30  2010 hello
-rwxr-xr-x  1 root  root     22 Mar 18  2010 read.txt
-rwxr-xr-x  1 root  root  3788208 Mar 22  2010 rootfs.tar.bz2
-rwxr-xr-x  1 root  root  40757 Mar 30  2010 wr
drwxr-xr-x  1 root  root    512 Mar 22  2010 英利
[root@emtronix /mnt/nfs]#./wr
open read file success!
open write file success!
All tests success!
[root@emtronix /mnt/nfs]#ls -l
-rwxr-xr-x  1 root  root    43025 Mar 26  2010 fork_study3
-rwxr-xr-x  1 root  root   22157 Mar 30  2010 hello
-rwxr-xr-x  1 root  root     22 Mar 18  2010 read.txt
-rwxr-xr-x  1 root  root  3788208 Mar 22  2010 rootfs.tar.bz2
-rwxr-xr-x  1 root  root  40757 Mar 30  2010 wr
-rwxr-xr-x  1 root  root     22 Mar 30  2010 write.txt
drwxr-xr-x  1 root  root    512 Mar 22  2010 英利
[root@emtronix /mnt/nfs]#

```

检查程序运行结果

这样，用户就可以在任何一台已经搭建好开发平台的 PC 上编辑自己的应用程序，并在 EM9280 上进行运行测试了。