

CHIPSEE_SOM3359/EVM3359 平台 Linux 系统开发指南

Version: 1.3

2013/02/25

注：文中所提到的商标及版权归各自商标持有人所有。

目 录

第一章	CHIPSEE_SOM3359 Linux 系统开发环境搭建.....	3
1.1	VMware workstation 安装.....	3
1.2	CHIPSEE_SOM3359/EVM3359 平台 Linux 系统的开发环境安装	6
第二章	CHIPSEE_SOM3359/EVM3359 平台 Linux 系统开发 IDE 一体化操作.....	7
2.1	CHIPSEE_SOM3359/EVM3359 开发环境启动.....	7
2.2	CHIPSEE_SOM3359/EVM3359 开发环境目录结构.....	10
2.3	CHIPSEE_SOM3359/EVM3359 开发环境一体化操作.....	11
附录:	18

第一章 CHIPSEE_SOM3359 Linux 系统 开发环境搭建

CHIPSEE_SOM3359/EVM3359 平台的 Linux 系统的开发、编译等需要基于 Linux 操作系统进行。CHIPSEE_SOM3359/EVM3359 平台的 Linux 系统的开发环境基于 Ubuntu 系统进行搭建，集成了编译器、SOM3359 的 Linux Kernel 源码、U-Boot 引导程序源码、编译相关依赖库、Eclipse IDE 及界面化编译操作环境。

CHIPSEE_SOM3359/EVM3359 平台的 Linux 系统的开发环境采用 VMware 虚拟机构建 Ubuntu 虚拟机系统，开发主机操作系统采用 Ubuntu 12.04 32 位版本。

1.1 VMware workstation 安装

VMware Workstation 是 VMware 公司开发的一款可以在 PC 机运行现有操作系统的同时虚拟其他客户机操作系统的软件。CHIPSEE_SOM3359/EVM3359 平台的 Linux 系统的开发环境利用 VMware workstation 软件在 Windows 7 系统下虚拟 Ubuntu 系统从而实现 Linux 开发环境。

VMware 下载和安装步骤如下：

1. VMware Workstation 软件下载：
访问 <http://www.vmware.com/products/workstation/overview.html> 网址，注册账户并申请评估版本，获取软件和评估注册码。
2. 下载完成后，双击 VMware-workstation-full-9.0.xxx.xx.exe 运行，启动安装，如下图所示。

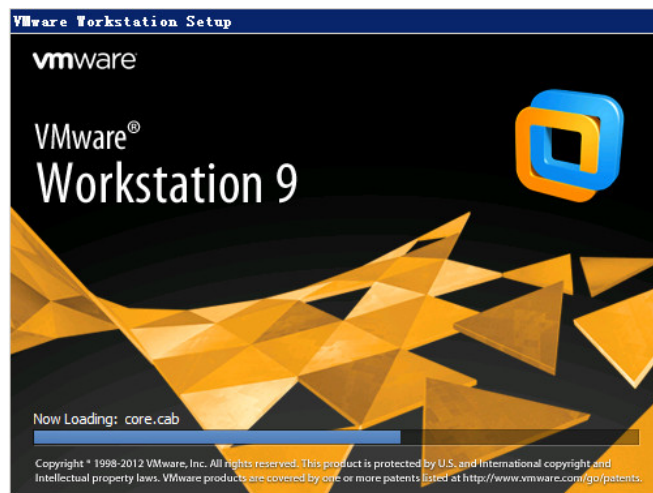
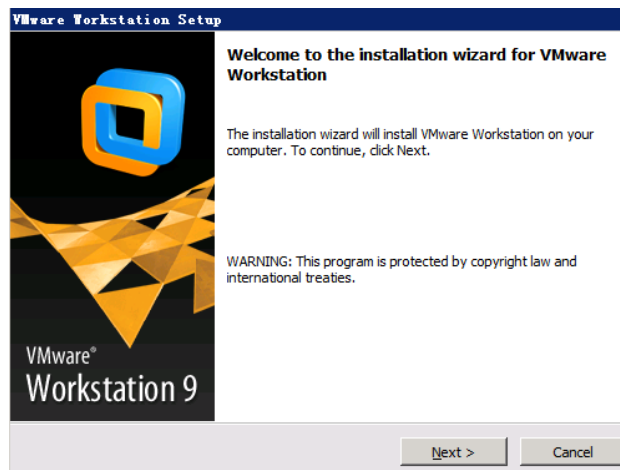
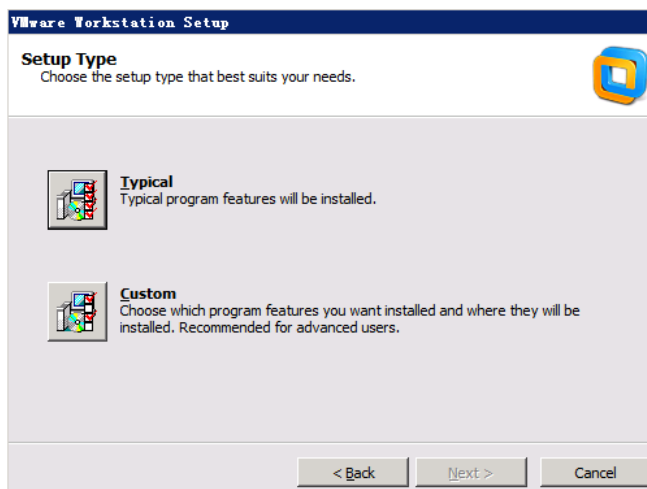


图 1-1

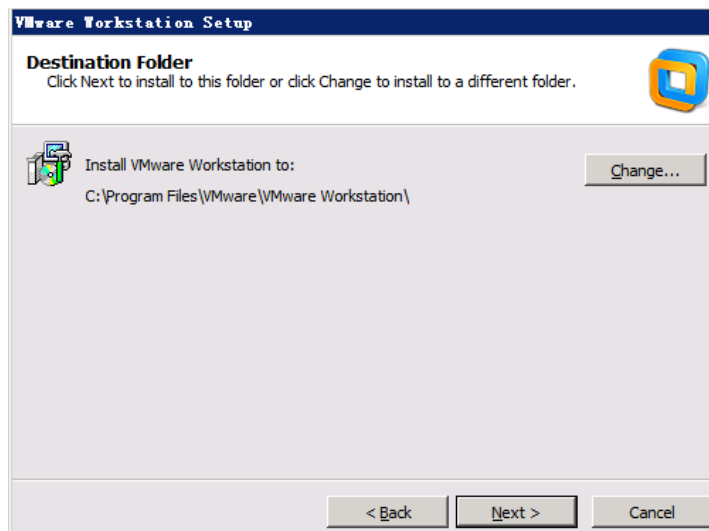
3. VMware 安装程序启动后，进入到 VMware 安装向导界面，弹出如下图所示。



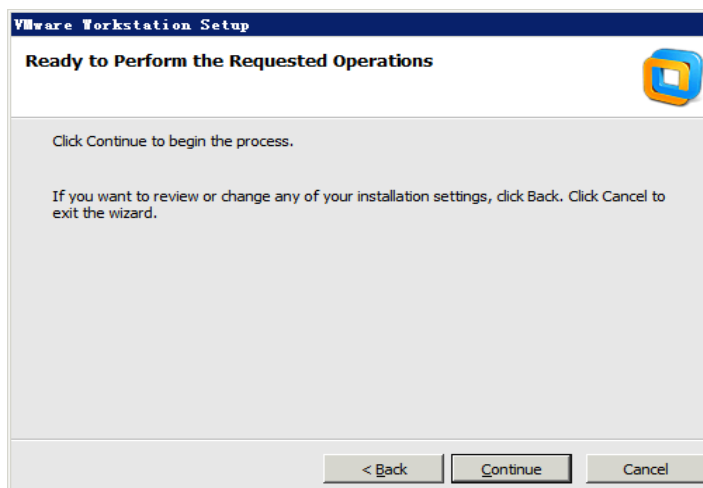
4. 点击安装向导界面中的 **Next** 按钮，进入安装类型选项界面，如下图所示。



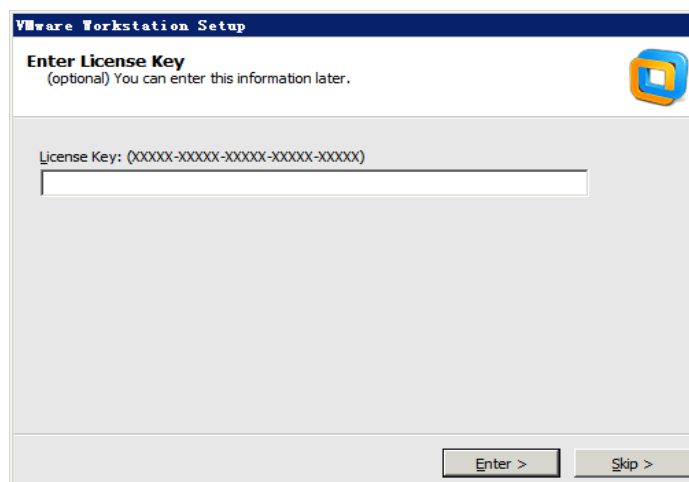
5. 在安装类型选项中，点击选择 **Typical** 类型处按钮，进入安装路径选择设置，如下图所示。



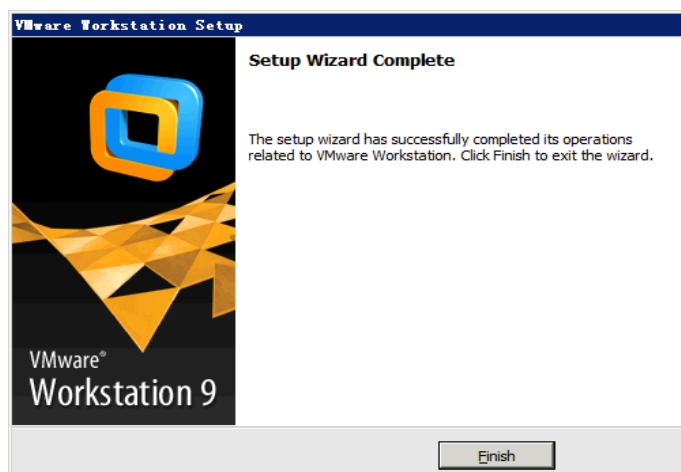
6. 点击 **Change** 按钮，选择安装的路径后，点击 **Next** 按钮，进入 **Software Updates**、**User Experience Improvement Program** 和 **Shortcut** 选项设置，依次点击 **Next** 按钮，进入到 **Ready to Perform the Requested Operations** 界面，如下图所示。



7. 点击 **Continue** 按钮，进行程序安装，安装过程中会要求输入 License Key，此处输入步骤 1 中获取的评估版注册码，点击 **Enter** 后继续安装，如下图所示。



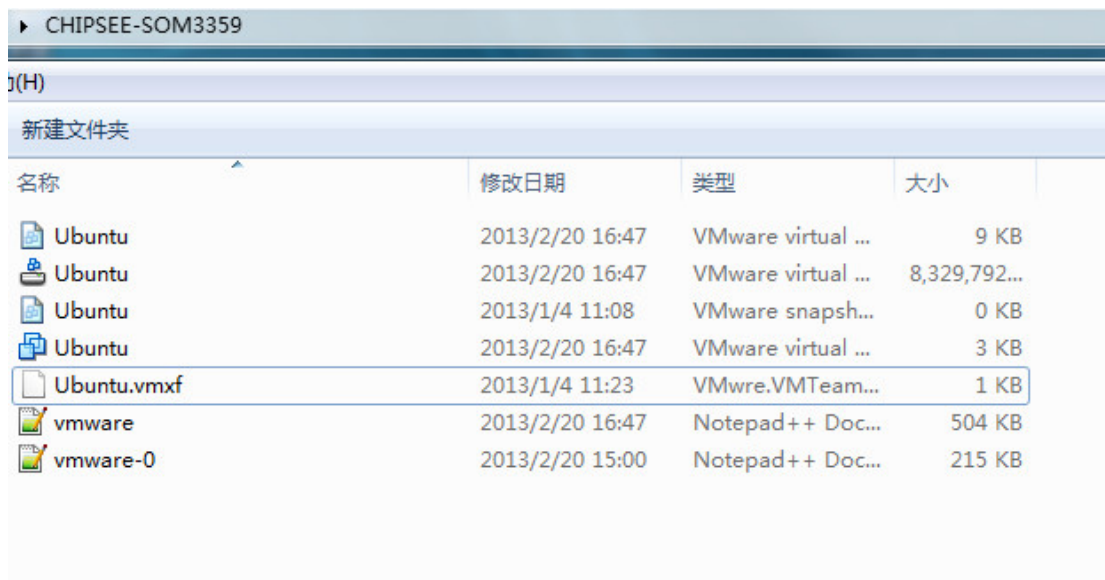
8. 等待完成安装，如下图所示，点击 **Finish** 按钮完成安装，并退出安装程序。



1.2 CHIPSEE_SOM3359/EVM3359 平台 Linux 系统的开发环境安装

将光盘中的 CHIPSEE_SOM3359_EVM3359.7z 压缩包，复制到 PC 主机目录下，以下以放置在 D:\SOM3359 目录下为例说明。

解压缩完成后，可以看到该目录下目录结构如下图所示。



CHIPSEE-SOM3359			
(H)			
新建文件夹			
名称	修改日期	类型	大小
Ubuntu	2013/2/20 16:47	VMware virtual ...	9 KB
Ubuntu	2013/2/20 16:47	VMware virtual ...	8,329,792...
Ubuntu	2013/1/4 11:08	VMware snapsh...	0 KB
Ubuntu	2013/2/20 16:47	VMware virtual ...	3 KB
Ubuntu.vmx	2013/1/4 11:23	VMwre.VMTeam...	1 KB
vmware	2013/2/20 16:47	Notepad++ Doc...	504 KB
vmware-0	2013/2/20 15:00	Notepad++ Doc...	215 KB

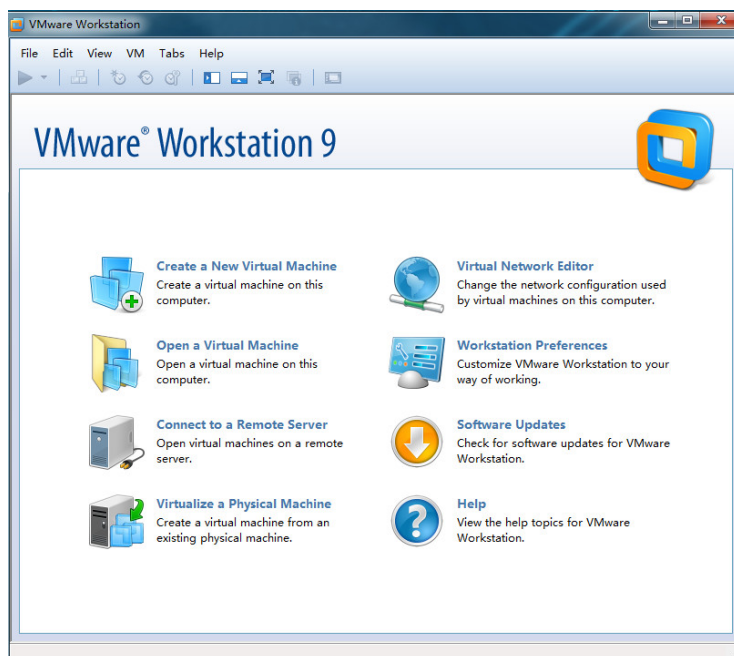
至此，CHIPSEE_SOM3359_EVM3359 平台的 Linux 系统开发所需要的所有资源已经安装完毕，下文将描述如何进行 Linux 和 U-Boot 的编辑、编译等操作，以及 SD Card 启动盘的制作和 NFS 的搭建一体化操作。

第二章 CHIPSEE_SOM3359/EVM3359 平台 Linux 系统开发 IDE 一体化操作

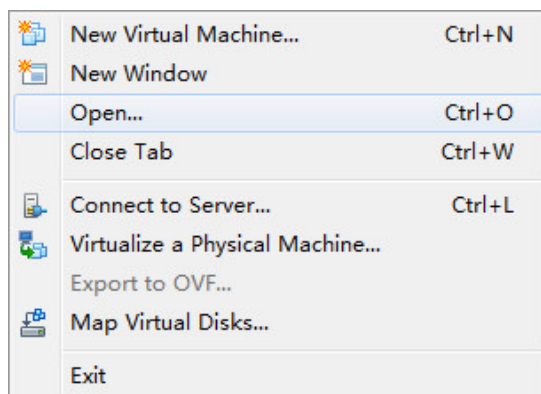
CHIPSEE_SOM3359_EVM3359 平台的 Linux 系统的开发环境中集成了 Eclipse IDE 用来简化代码的编辑、编译和工程管理工作，并集成了 Linux 内核编译、配置，U-Boot 的编译、配置，SDCard 制作、NFS 搭建一体化操作命令，用户仅需双击操作即可完成相应的操作，降低操作复杂度，最大程度的帮助不熟悉 Linux 命令操作的开发人员或者入门人员进行开发操作。

2.1 CHIPSEE_SOM3359/EVM3359 开发环境启动

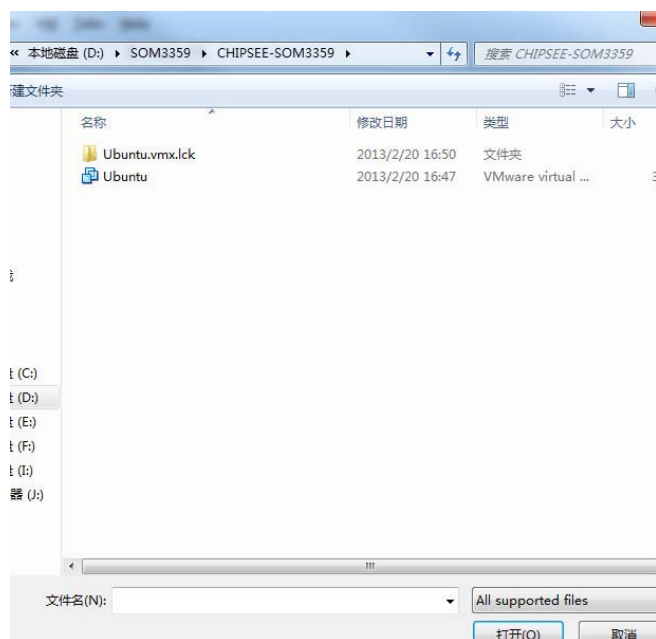
1. 双击桌面上的 VMware 图标，启动 VMware 程序。启动完成后，如下图所示。



2. 点击 VMware 程序的左上角的 File Menu，弹出的对话框如下图所示。



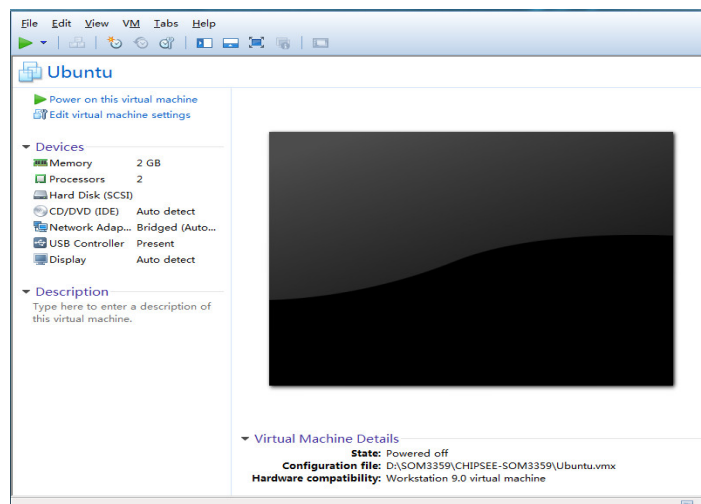
3. 依照对话框的提示，将地址栏指向虚拟机所在位置 D:\SOM3359\CHIPSEE_SOM3359，如下图所示。



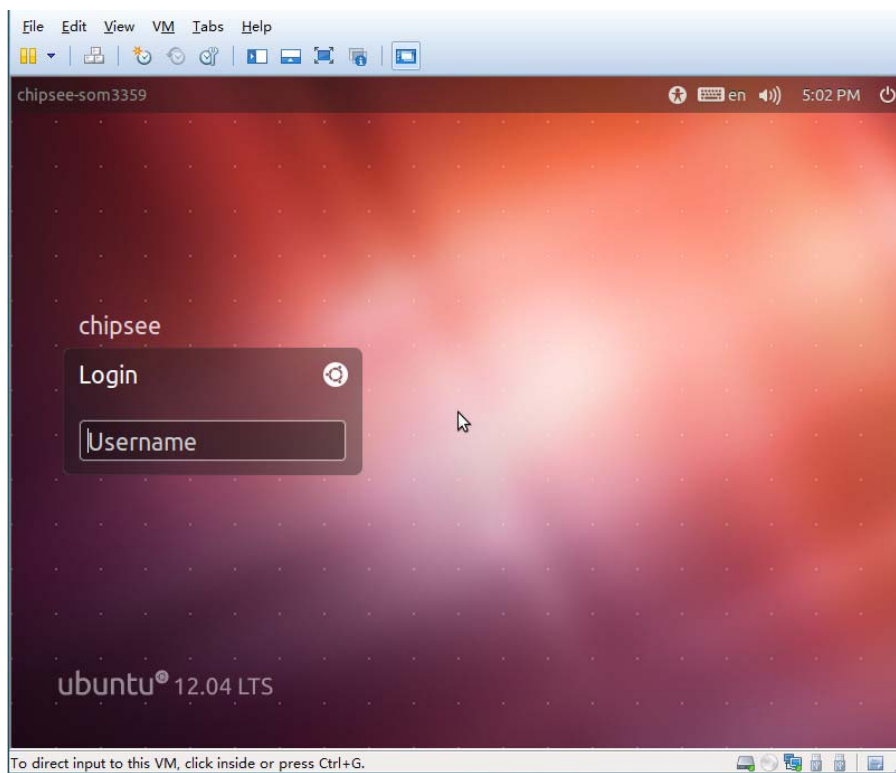
4. 选中 Ubuntu 文件并点击打开按钮，如下图所示。



5. 点击打开按钮后，如下图所示。点击 **Power on this virtual machine** 按钮，即可启动虚拟机。



6. 等待 Ubuntu 虚拟机系统启动完毕，如下图所示出现登陆界面。



7. 在登陆界面中输入账户 root，密码 som3359 即可登陆进入系统桌面，如下图所示。



8. 登陆完成后，可以点击 VMware 上的最大化操作菜单，将 Ubuntu 虚拟机界面最大化。



至此，CHIPSEE_SOM3359/EVM3359 开发环境启动已经完成，用户可以在 Ubuntu 环境下进行开发工作。

2.2 CHIPSEE_SOM3359/EVM3359 开发环境目录结构

进入系统后，可以通过打开左侧的 Panel 中的 Home Folder，通过 Computer->File System 查看各个目录下的文件。CHIPSEE_SOM3359/EVM3359 相关的代码和工具均在 opt 目录下，所有资源目录结构如下所示。




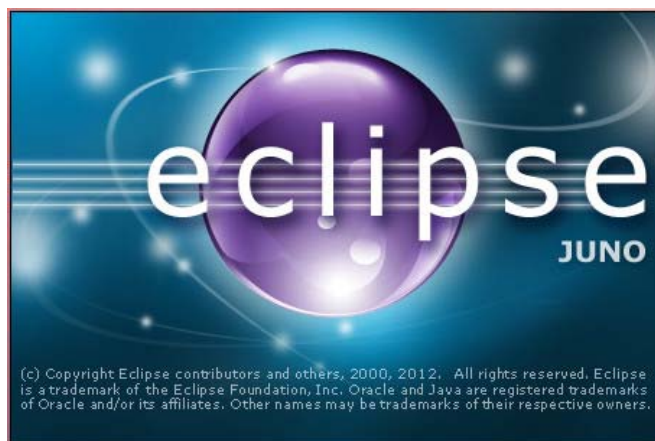
目录说明如下：

目录	说明
Eclipse	Eclipse IDE 环境
Projects	CHIPSEE_SOM3359/EVM3359 工程管理目录
SOM3359	针对 SOM3359/EVM3359 对应的工程文件目录
filesystem	文件系统压缩包和 nfs 文件系统搭建脚本
linux-devkit	编译工具
prebuilt-images	预编译的二进制启动文件和文件系统以及制作 SDCard 启动工具
SOM3359L	CHIPSEE_SOM3359/EVM3359 源代码目录
Linux-3.2.0-SOM3359	CHIPSEE_SOM3359/EVM3359 Linux 源代码
U-Boot-SOM3359	CHIPSEE_SOM3359/EVM3359 U-Boot 源代码

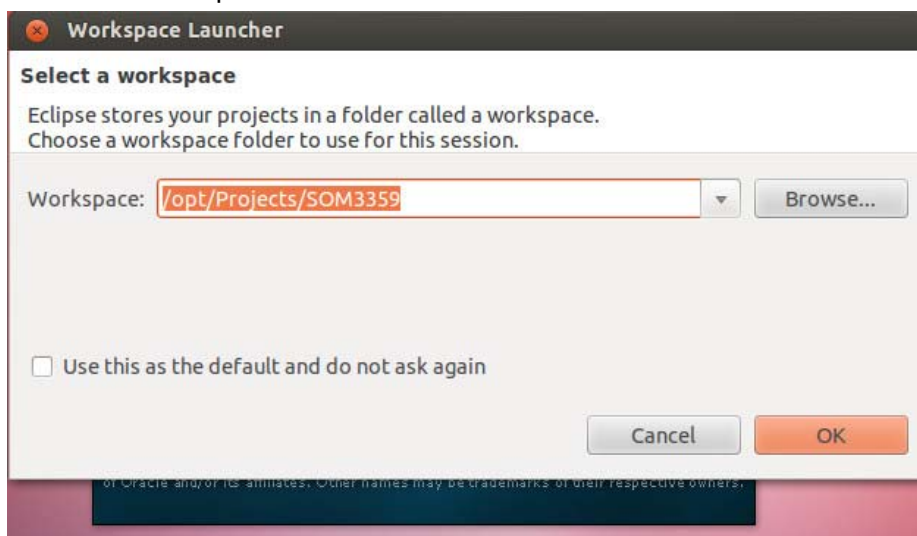
2.3 CHIPSEE_SOM3359/EVM3359 开发环境一体化操作

Ubuntu 虚拟系统启动完毕后，可以通过启动 Eclipse IDE 开发环境进行后续的编辑和编译等操作。详细步骤如下：

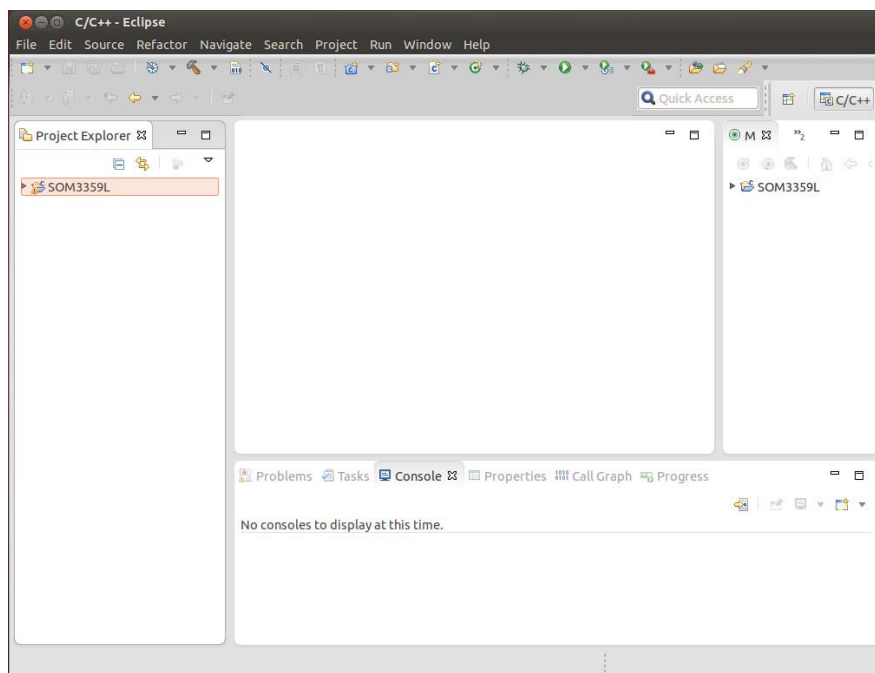
1. 启动 Eclipse IDE 开发环境，点击左侧 Panel 中 Chipsee 的 Logo 图标, Eclipse 启动界面如下图所示。



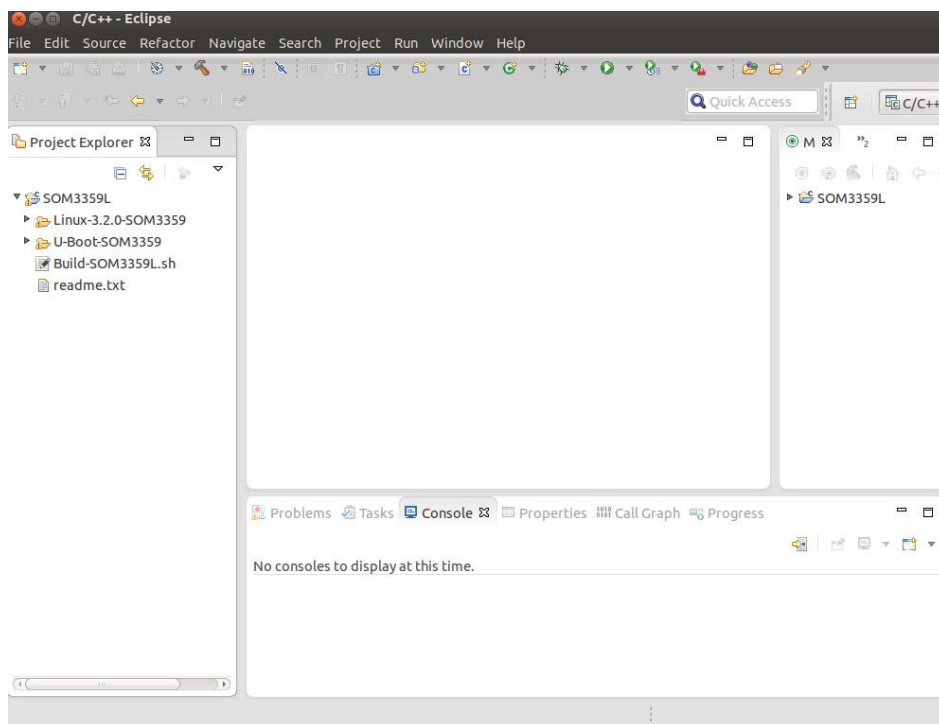
2. 在弹出的 Workspace 选择目录中，可以看到如下图所示的工程路径。



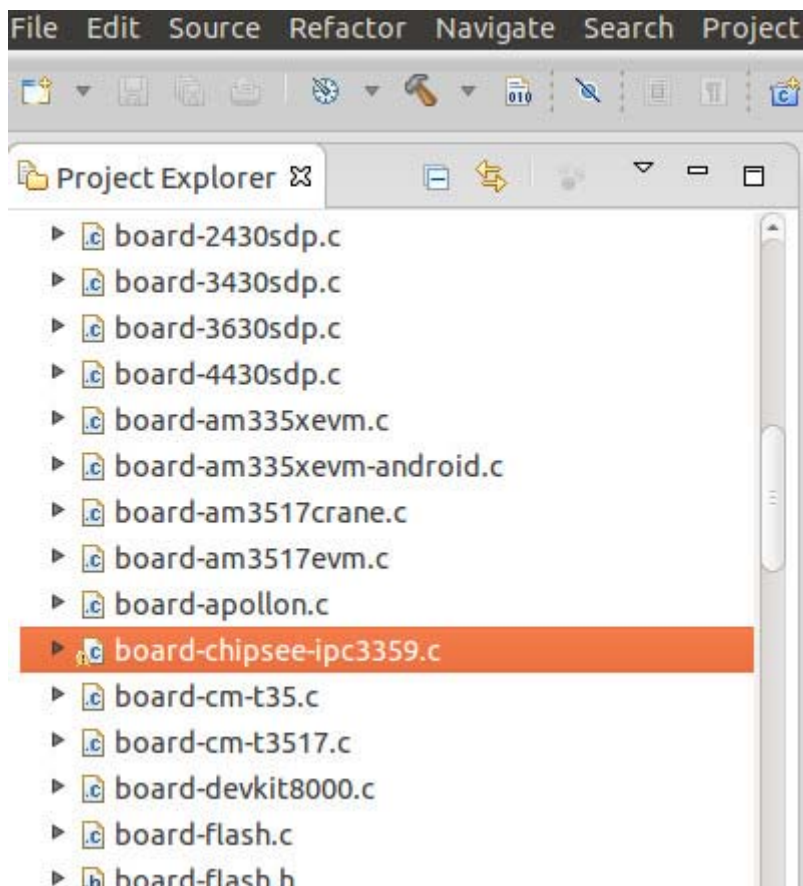
3. 点击 OK，继续 Eclipse 的启动进程，直至 Eclipse 启动完成，进入如下图所示的界面。

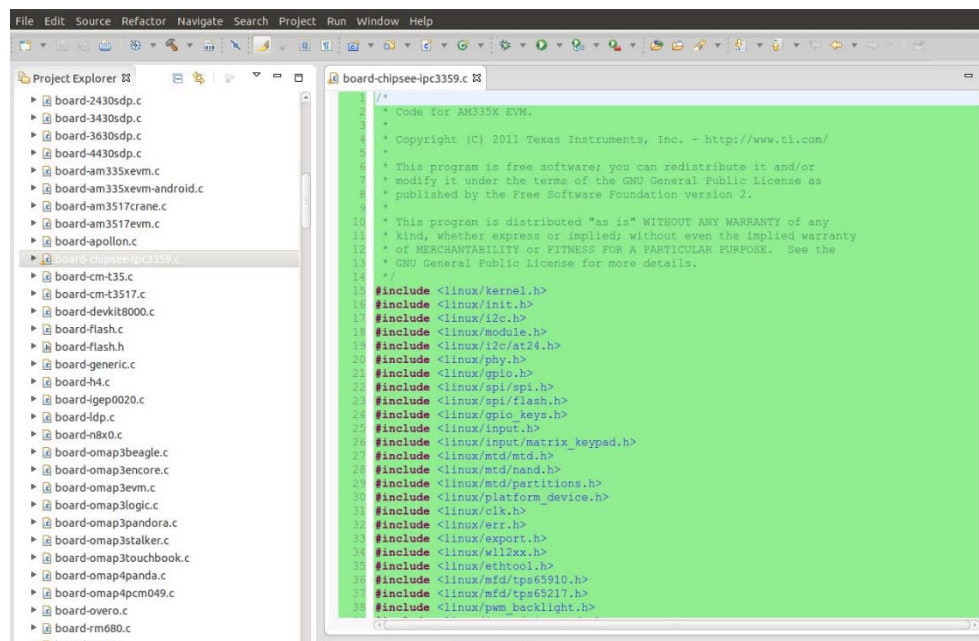


4. 从界面可以看到，左侧的 Project Explorer 中的工程为 SOM3359L，其下包含 Linux-3.2.0-SOM3359 和 U-Boot-SOM3359 工程及源码，点击 SOM3359L 即可查看其下的资源，如下图所示。

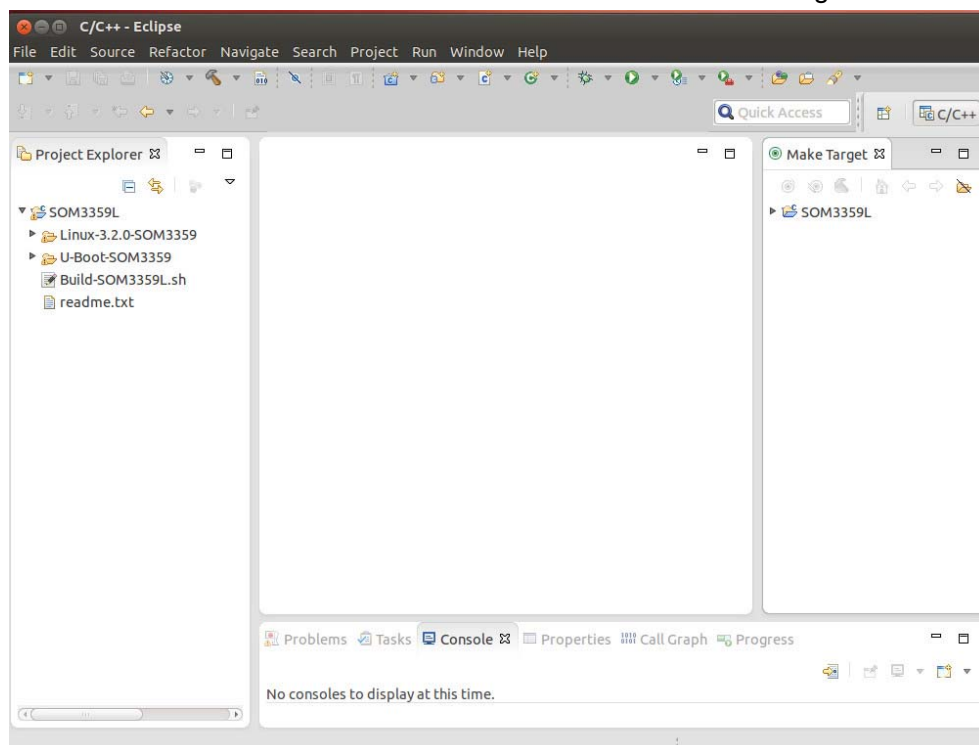


5. 可以依次点击查看 Linux-3.2.0-SOM3359 或者 U-Boot-SOM3359 下的目录内容，并可通过双击文件打开查看代码或编辑，如下图例所示。

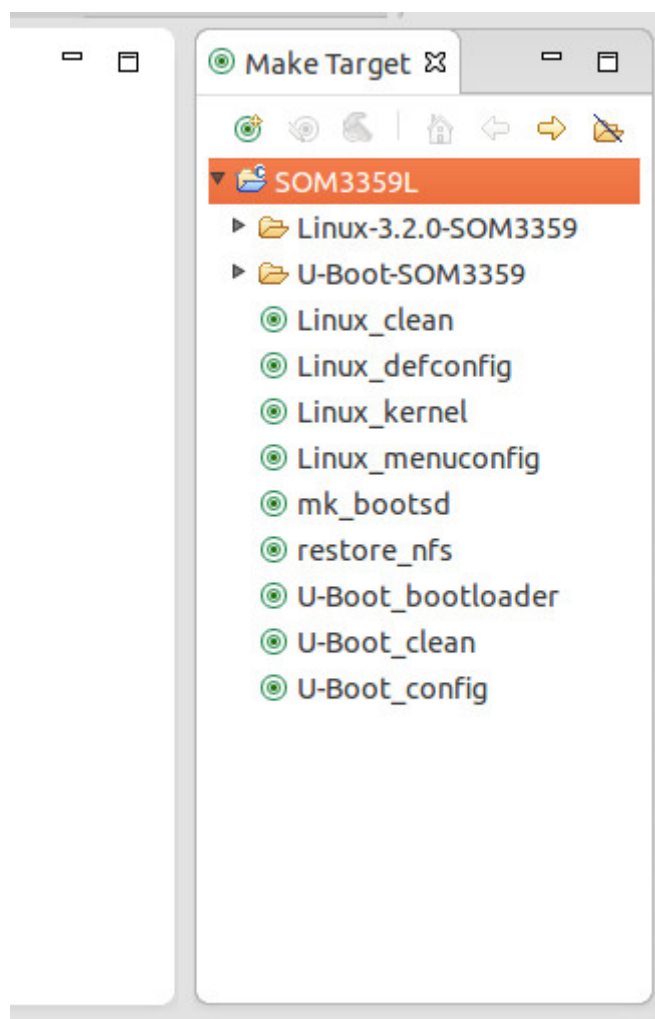




6. 在 Eclipse IDE 开发环境的左侧可以看到如下所示的 Make Target 菜单界面，其下是 SOM3359L，SOM3359L 中包含所有编译操作的一体化 Target 操作。

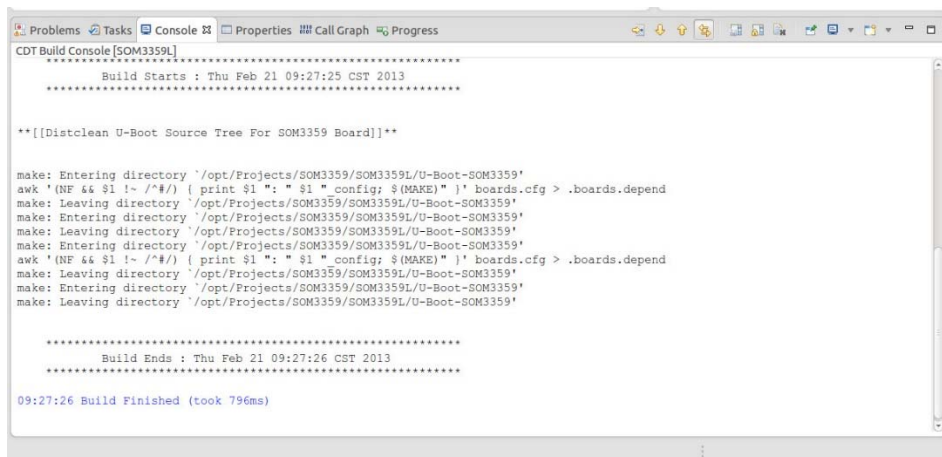


7. 点击 SOM3359L 可以看到其下的一体化操作命令如下图所示。



Target	说明
Linux_clean	执行 Linux 内核编译命令的 clean 操作，即清除 Linux 代码的所有编译产生的文件
Linux_defconfig	执行 Linux 内核编译命令的 defconfig 操作，即配置 Linux 内核为默认配置
Linux_kernel	执行 Linux 内核编译命令的 ulmage 和 modules 操作，即编译 Linux 内核产生 ulmage 文件和 modules 文件
Linux_menuconfig	执行 Linux 内核编译命令的 menuconfig 操作，即手动配置 Linux 内核
mk_bootsd	执行制作启动用 SDCard 的脚本命令
restorenfs	执行 NFS 文件系统搭建的脚本命令
U-Boot_bootloader	执行 U-Boot 的编译操作，产生 MLO 和 u-boot.bin/u-boot.img 文件
U-Boot_clean	执行 U-Boot 的 clean 操作，清除编译过程中产生的所有文件
U-Boot_config	执行 U-Boot 的配置操作，配置板卡操作

- 下面以 U-Boot 的一体化操作为例，说明如何操作使用。
- U-Boot_clean 操作，双击 Make Target 的 U-Boot_clean，此时执行 U-Boot_clean 操作，看在 console 处看到，如下 clean 操作的信息。



```

CDT Build Console [SOM3359L]
*****
Build Starts : Thu Feb 21 09:27:25 CST 2013
*****

**[[Distclean U-Boot Source Tree For SOM3359 Board]]**

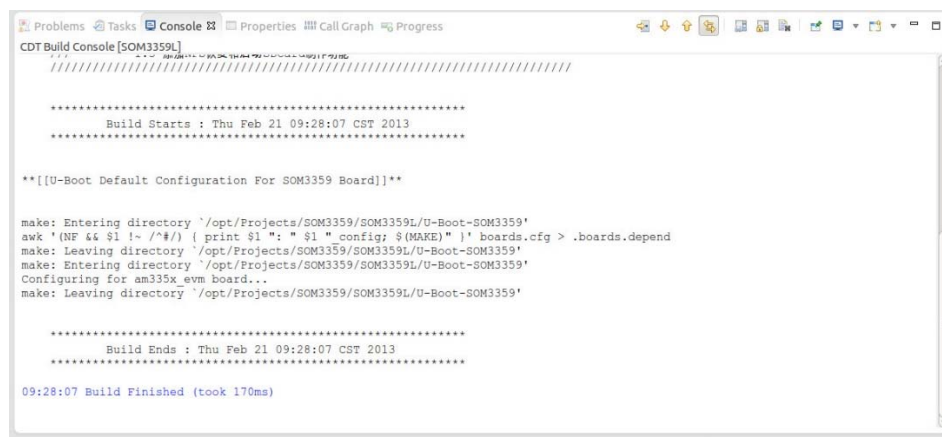
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
awk 'NF && $1 !~ /^#/ { print $1 " " $1 " config; $(MAKE)" }' boards.cfg > .boards.depend
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
awk 'NF && $1 !~ /^#/ { print $1 " " $1 " config; $(MAKE)" }' boards.cfg > .boards.depend
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'

*****
Build Ends : Thu Feb 21 09:27:26 CST 2013
*****

09:27:26 Build Finished (took 796ms)

```

10. U-Boot_config 操作, 双击 Make Target 的 U-Boot_config, 此时执行 U-Boot_config 操作, 看在 console 处看到, 如下 config 操作的信息。



```

CDT Build Console [SOM3359L]
*****
Build Starts : Thu Feb 21 09:28:07 CST 2013
*****

**[[U-Boot Default Configuration For SOM3359 Board]]**

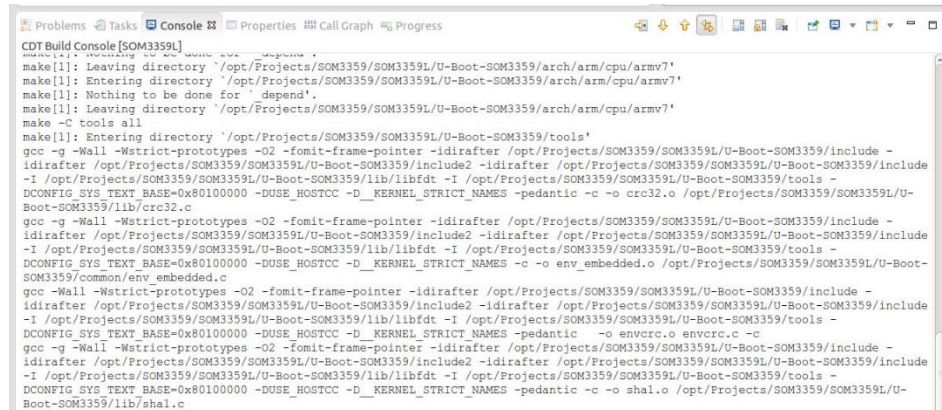
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
awk 'NF && $1 !~ /^#/ { print $1 " " $1 " config; $(MAKE)" }' boards.cfg > .boards.depend
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
Configuring for am335x_evm board...
make: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'

*****
Build Ends : Thu Feb 21 09:28:07 CST 2013
*****

09:28:07 Build Finished (took 170ms)

```

11. U-Boot_bootloader 操作, 双击 Make Target 的 U-Boot_bootloader, 此时执行 U-Boot 的编译操作, 看在 console 处看到, 如下编译操作过程的信息及最终产生引导程序文件的信息。



```

CDT Build Console [SOM3359L]
*****
Build Starts : Thu Feb 21 09:28:07 CST 2013
*****

**[[U-Boot Default Configuration For SOM3359 Board]]**

make: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'
make[1]: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/arch/arm/cpu/armv7'
make[1]: Nothing to be done for `depend'.
make[1]: Leaving directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/arch/arm/cpu/armv7'
make -C tools all
make[1]: Entering directory `/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools'
gcc -g -Wall -Wstrict-prototypes -O2 -fomit-frame-pointer -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include2 -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/libfdt -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools -DCONFIG_SYS_TEXT_BASE=0x80100000 -DUSE_HOSTCC -D_KERNEL_STRICT_NAMES -pedantic -c -o crc32.o /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/crc32.c
gcc -g -Wall -Wstrict-prototypes -O2 -fomit-frame-pointer -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include2 -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/libfdt -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools -DCONFIG_SYS_TEXT_BASE=0x80100000 -DUSE_HOSTCC -D_KERNEL_STRICT_NAMES -c -o env_embedded.o /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/common/env_embedded.c
gcc -g -Wall -Wstrict-prototypes -O2 -fomit-frame-pointer -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include2 -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/libfdt -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools -DCONFIG_SYS_TEXT_BASE=0x80100000 -DUSE_HOSTCC -D_KERNEL_STRICT_NAMES -pedantic -o envrcrc.o envrcrc.c -c
gcc -g -Wall -Wstrict-prototypes -O2 -fomit-frame-pointer -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include2 -idirafter /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/include -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/libfdt -I /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools -DCONFIG_SYS_TEXT_BASE=0x80100000 -DUSE_HOSTCC -D_KERNEL_STRICT_NAMES -pedantic -c -o sha1.o /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/lib/sha1.c

```



```

CDT Build Console [SOM3359L]
glibc/4.5.3/include -pipe -D__CONFIG_ARM -D ARM -mabi=aapcs-linux -mno-thumb-interwork -march=armv5 -Wall -Wstrict-
prototypes -fno-stack-protector -Wno-format-nonliteral -Wno-format-security -o omap3_spi.o omap3_spi.c -c
/opt/Projects/SOM3359/SOM3359L/./linux-devkit/bin/arm-arago-linux-gnueabi-ld -r -o libspi.o omap3_spi.o
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/spi'
make -C drivers/twserial/
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/twserial'
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/twserial'
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/twserial'
rm -f libtws.o; /opt/Projects/SOM3359/SOM3359L/./linux-devkit/bin/arm-arago-linux-gnueabi-ar rcs libtws.o
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/twserial'
make -C drivers/usb/eth/
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/eth'
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/eth'
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/eth'
rm -f libusb_eth.o; /opt/Projects/SOM3359/SOM3359L/./linux-devkit/bin/arm-arago-linux-gnueabi-ar rcs libusb_eth.o
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/eth'
make -C drivers/usb/gadget/
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/gadget'
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/gadget'
make[1]: Entering directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/drivers/usb/gadget'
/opt/Projects/SOM3359/SOM3359L/./linux-devkit/bin/arm-arago-linux-gnueabi-gcc -g -Os -fno-common -ffixed-r8 -msoft-float -
D_KERNEL -D__CONFIG_SYS_TEXT_BASE=0x80100000 -D__CONFIG_SPL_TEXT_BASE=0x402F0400 -I/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/
include -fno-builtin -ffastmath -isystem /opt/Projects/SOM3359/SOM3359L/linux-devkit/bin/./lib/gcc/arm-arago-linux-
gnueabi/4.5.3/include -pipe -D__CONFIG_ARM -D ARM -mabi=aapcs-linux -mno-thumb-interwork -march=armv5 -Wall -Wstrict-
spl.map -o u-boot-spl
/opt/Projects/SOM3359/SOM3359L/./linux-devkit/bin/arm-arago-linux-gnueabi-objcopy --gap-fill=0xff -O binary /opt/Projects/
SOM3359/SOM3359L/U-Boot-SOM3359/spl/u-boot-spl /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/spl/u-boot-spl.bin
/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools/mkimage -T omapimage \
-a 0x402F0400 -d /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/spl/u-boot-spl.bin /opt/Projects/SOM3359/SOM3359L/U-Boot-
SOM3359/MLO
Section CHSETTINGS offset 40 length c
CHSETTINGS (c0c0c0c1) valid:0 version:1 reserved:0 flags:0
GP Header: Size 1325a LoadAddr 402f0400
/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/tools/mkimage -T omapimage -n byteswap-spl \
-a 0x402F0400 -d /opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/spl/u-boot-spl.bin /opt/Projects/SOM3359/SOM3359L/U-Boot-
SOM3359/MLO.spl
Section ESHCNITTT offset 40000000 length c000000
CHSETTINGS (c1c0c0c0) valid:0 version:0 reserved:1 flags:0
GP Header: Size 1325a LoadAddr 402f0400
make[1]: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359/spl'
make: Leaving directory '/opt/Projects/SOM3359/SOM3359L/U-Boot-SOM3359'

*****
Build Ends : Thu Feb 21 09:30:09 CST 2013
*****

09:30:16 Build Finished (took 1m:44s.904ms)

```

CHIPSEE_SOM3359/EVM3359 平台的 Linux 系统开发环境完全基于 Eclipse IDE 构建，所有操作均可通过 Make Target 中的相关操作完成，用户所需要仅仅是针对板卡进行源码的修改和分析，通过双击对应的 Make Target 操作完成编译等工作以及 SDCard 制作操作等等。

Note: 关于 Eclipse 相关操作此处不做详细介绍，如有需要请通过网络查询 Eclipse 的使用详细指南。

附录:

1. www.eclipse.org
2. http://software-dl.ti.com/dsps/dsps_public_sw/am_bu/sdk/AM335xSDK/latest/index_FDS.html
3. www.chipsees.com