

## RZ/A1

R01AN3127CC0100

Rev. 1.00

### 基于 RSK+RZA1 平台的 Linux XIP Kernel 开发环境搭建

2015.10.09

#### 概要

此文档描述的是基于 RZ/A1H 芯片的 RSK+RZA1 开发板，在做 XIP 版 Linux Kernel 开发时怎么去搭建开发环境。下面是在搭建开发环境过程中的实际操作过程，记录下来作为自己的经验积累，也可以作为他人的一个入门参考。

#### 对象主板

RSK+RZA1, YR0K77210C000BE

#### 对象 MCU

RZ/A1H, R7S721001VCBG。

在把此应用说明应用在其它 MCU 上时，需要再评估是否适合。

## 目录

|                                       |           |
|---------------------------------------|-----------|
| <b>1. 概述 .....</b>                    | <b>3</b>  |
| <b>2. 虚拟机的介绍与设置 .....</b>             | <b>4</b>  |
| 2.1 虚拟机简介 .....                       | 4         |
| 2.2 虚拟机搭建 .....                       | 4         |
| 2.3 网络设置 .....                        | 8         |
| 2.4 文件共享 .....                        | 9         |
| 2.5 VirtualBox 的增强功能 .....            | 10        |
| <b>3. Ubuntu 主机设置 .....</b>           | <b>11</b> |
| 3.1 Ubuntu 下载 .....                   | 11        |
| 3.2 Ubuntu 的安装 .....                  | 11        |
| 3.3 Linux 开发必备工具包安装 .....             | 12        |
| 3.4 其它工具 .....                        | 12        |
| <b>4. RZ/A1 Linux BSP 下载及编译 .....</b> | <b>13</b> |
| 4.1 XIP 版 Linux 介绍 .....              | 13        |
| 4.2 RZ/A1 Linux BSP 下载 .....          | 13        |
| 4.3 Linux 编译 .....                    | 13        |
| 4.4 目标文件烧录 .....                      | 14        |
| 4.4.1. Segger J-Link 工具安装: .....      | 14        |
| 4.4.2. J-Link 烧录脚本: .....             | 14        |
| 4.4.3. 烧录操作: .....                    | 15        |
| 4.5 USB 转串口驱动安装 .....                 | 15        |
| 4.6 Linux 启动 .....                    | 16        |
| <b>5. 应用程序开发 .....</b>                | <b>16</b> |
| 5.1 代码工具 Eclipse 安装 .....             | 16        |
| 5.2 使用 Eclipse 建立工程 .....             | 17        |
| 5.3 调试方法介绍: .....                     | 19        |
| 5.4 示例工程 .....                        | 19        |

## 1. 概述

我们在搭建 RZ/A1 的 Linux 开发环境的时候，需要在 Linux OS 环境下进行。这里选用的是 Ubuntu。一般工程师都只有一台计算机，为了方便，我们需要借助虚拟机软件来在原有 Windows 系统的下安装 Ubuntu 系统。

在下面的章节中，我会先介绍一下怎么使用虚拟机，并且会介绍一下虚拟机使用的技巧，便于虚拟机里面的 Ubuntu 系统的使用，如网络设置，文件共享等。然后介绍一下做嵌入式开发时，需要用到的一些 Ubuntu 系统工具和设置。最后介绍一下 RSK+RZ/A1 平台上的 Linux 编译和怎么进行应用开发。

## 2. 虚拟机的介绍与设置

### 2.1 虚拟机简介

虚拟机软件可以在一台计算机(就是我们正在使用的真实的计算机)上模拟出来若干台虚拟计算机, 每台虚拟的计算机有自己的 CPU, 内存, 硬盘, 网卡等资源, 这些资源也是软件虚拟出来的, 但是在每一个虚拟的环境里, 你可以像使用一台真正的计算机一样使用它。每台虚拟计算机可以运行单独的操作系统而互不干扰, 可以实现一台电脑“同时”运行几个操作系统, 还可以将这几个操作系统连成一个网络。而且可以支持很多主流非主流的操作系统。

虚拟机软件有很多种, 一般常用的是 VMware Workstation 和 Oracle VM VirtualBox。这里我推荐使用 Oracle VM VirtualBox。因为是完全免费的, 支持 Linux、Unix、windows 等操作系统, 而且轻便简洁, 好用, 且对计算机的配置要求也不高。当然功能与收费软件比就相对少一些, 不过针对一般用户功能就已经相当强大了, 安装 Ubuntu 系统也足够用了。

VirtualBox 是一款开源虚拟机软件。VirtualBox 是由德国 Innotek 公司开发, 由 Sun Microsystems 公司出品的软件, 使用 Qt 编写, 在 Sun 被 Oracle 收购后正式更名成 Oracle VM VirtualBox。Innotek 以 GNU General Public License (GPL) 释出 VirtualBox, 并提供二进制版本及 OSE 版本的代码。使用者可以在 VirtualBox 上安装并且执行 Solaris、Windows、DOS、Linux、OS/2 Warp、BSD 等系统作为客户端操作系统。

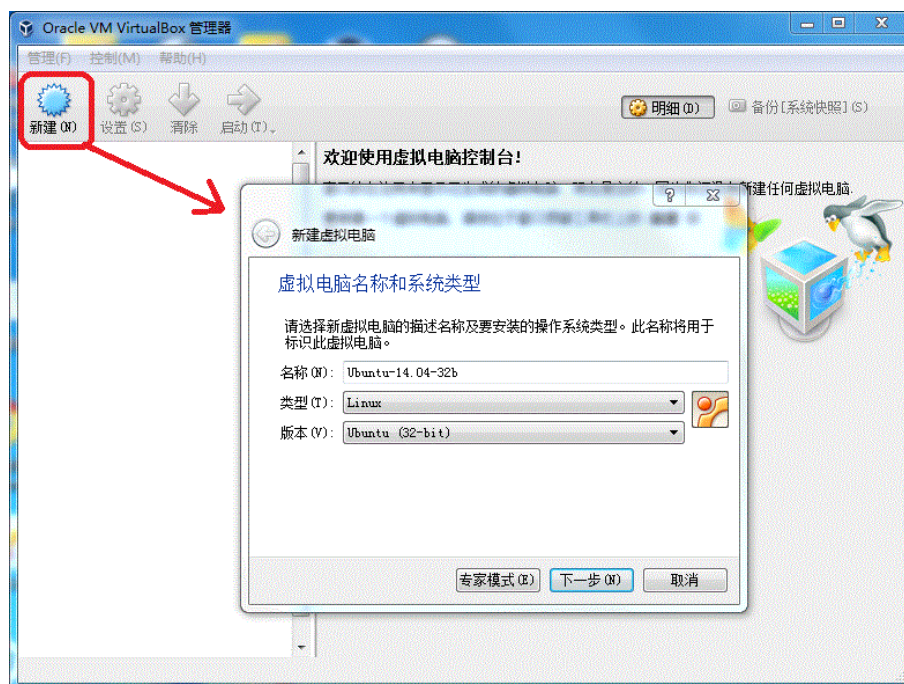
VirtualBox 号称是最强的免费虚拟机软件, 它不仅具有丰富的特色, 而且性能也很优异! 它简单易用, 可虚拟的系统包括 Windows (从 Windows 3.1 到 Windows10、Windows Server 2012, 所有的 Windows 系统都支持)、Mac OS X、Linux、OpenBSD、Solaris、IBM OS2 甚至 Android 等操作系统! 使用者可以在 VirtualBox 上安装并且运行上述的这些操作系统!

你可以从 VirtualBox 官网: <https://www.virtualbox.org/> 下载到最新版本的软件。对于虚拟机软件的安装步骤, 这里就不赘述了, 按照安装提示一步步操作即可。下面具体介绍一些 VirtualBox 软件中的一些使用技巧。

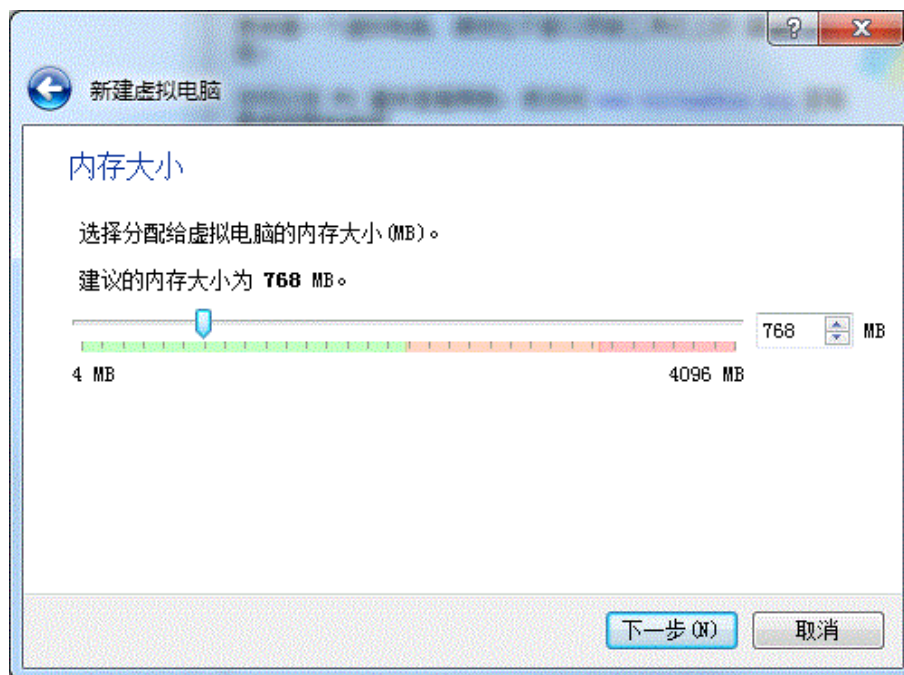
### 2.2 虚拟机搭建

下面以在 VirtualBox 5.0.6 上安装 Ubuntu 为例, 描述虚拟机的搭建过程。

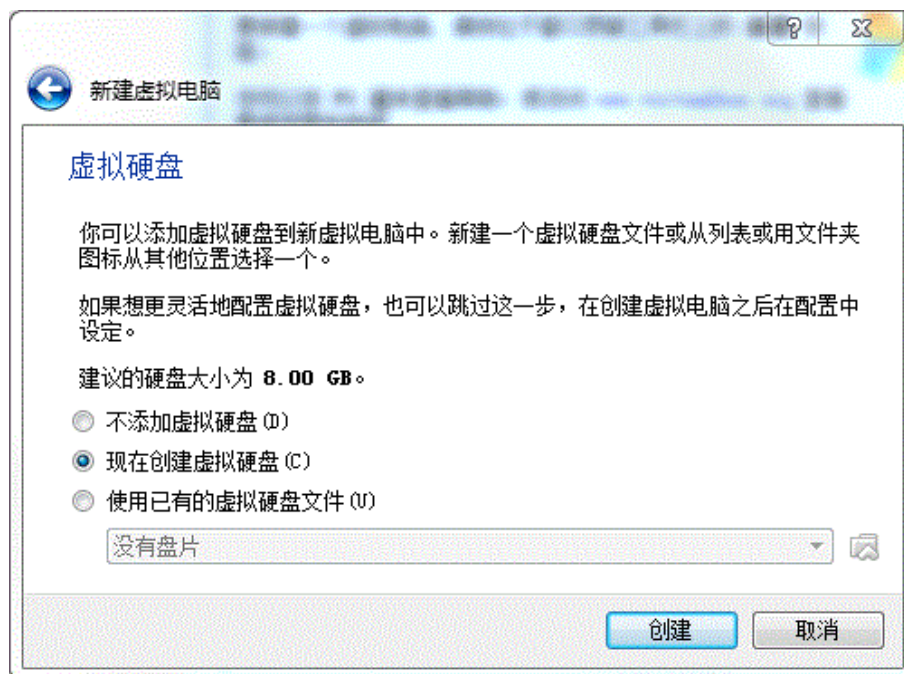
打开安装好的 VirtualBox, 点击新建按钮, 弹出创建新建虚拟电脑的向导对话框, 在对话框里选择 [下一步]后, 进入如下页面:



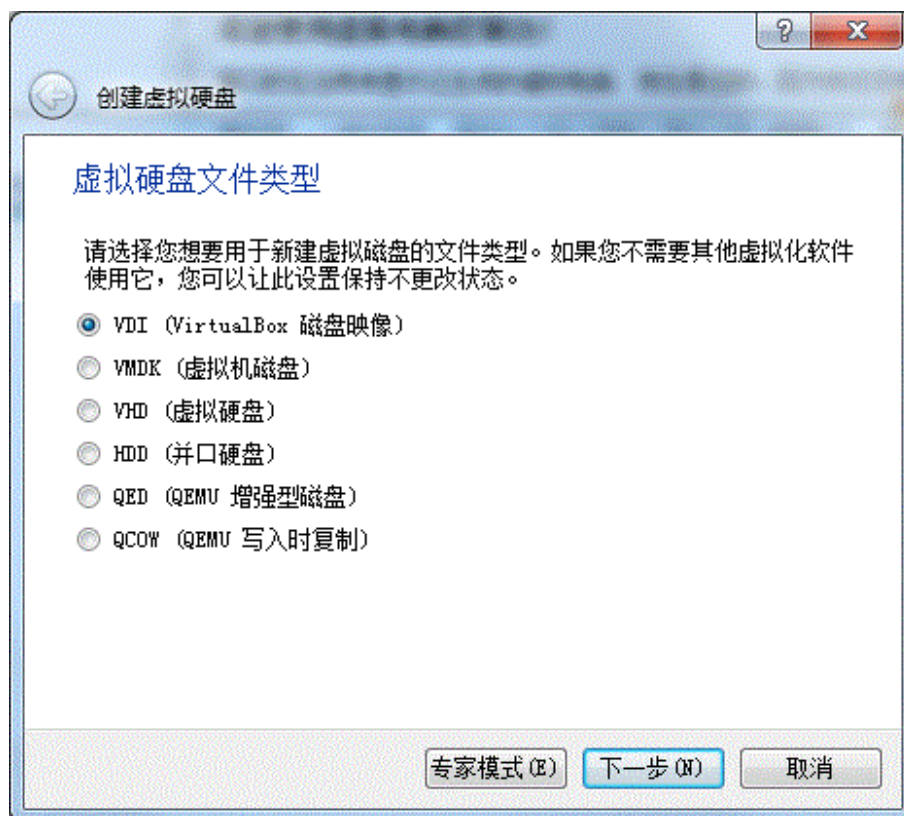
输入虚拟电脑的[名称]，可以随意设置，如 Ubuntu-14.04，在[操作系统]栏选择 Linux，在[版本]栏选择 Ubuntu。当前是向导模式，在后面每一项设置页内会有详细的说明以及设置的推荐值。如果点击【专家模式】按钮，该对话框就会改变，进入专家模式，所有的设置会显示在一页内，每一项设置没有说明信息。如果是新手，还是不要点击【专家模式】了。在当前的向导模式下点击[下一步]，出现如下对话框。



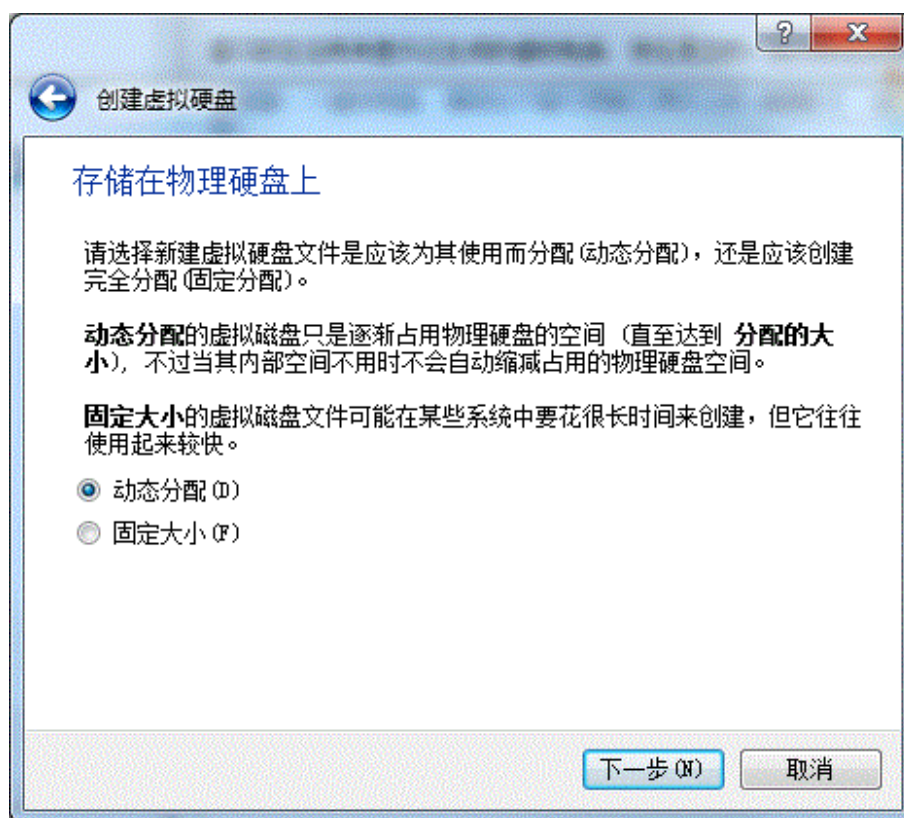
在该对话框里，需要设置的是虚拟电脑的内存大小，最好使用推荐值。如果设置太大会给主机带来比较大的负荷，太小的话虚拟机里面的操作系统跑起来不流畅。然后点击[下一步]，出现如下对话框。



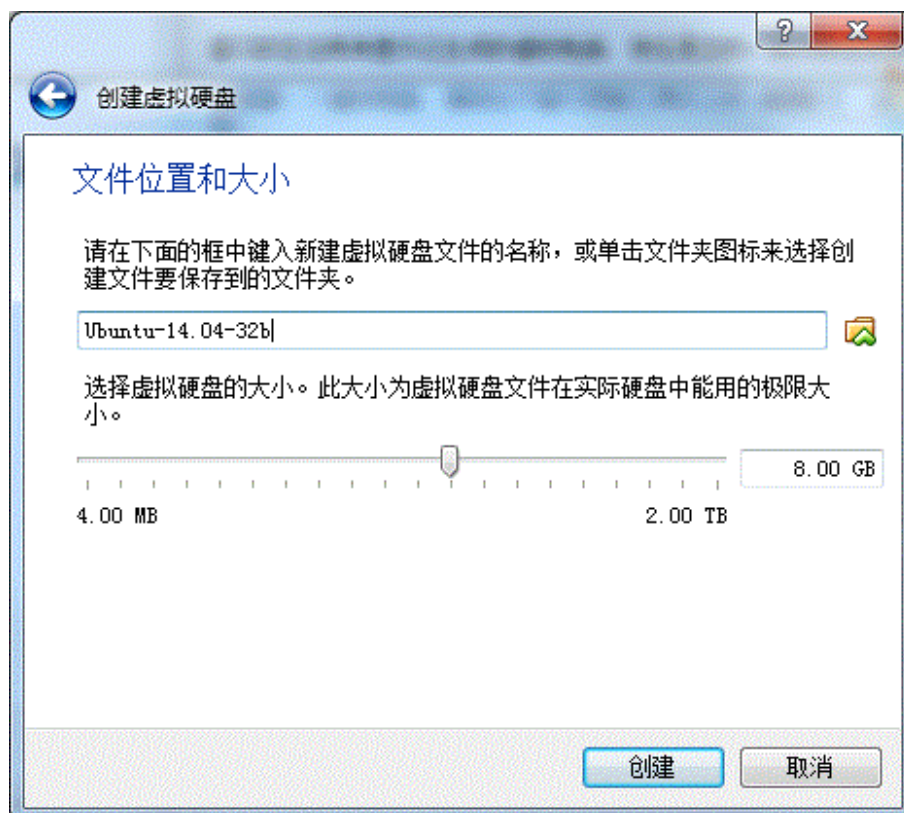
这个对话框是要为虚拟电脑设置硬盘。因为是新安装系统，我们需要创建一个新的虚拟硬盘。点击[下一步]后出现如下窗口来选择虚拟磁盘的文件格式。



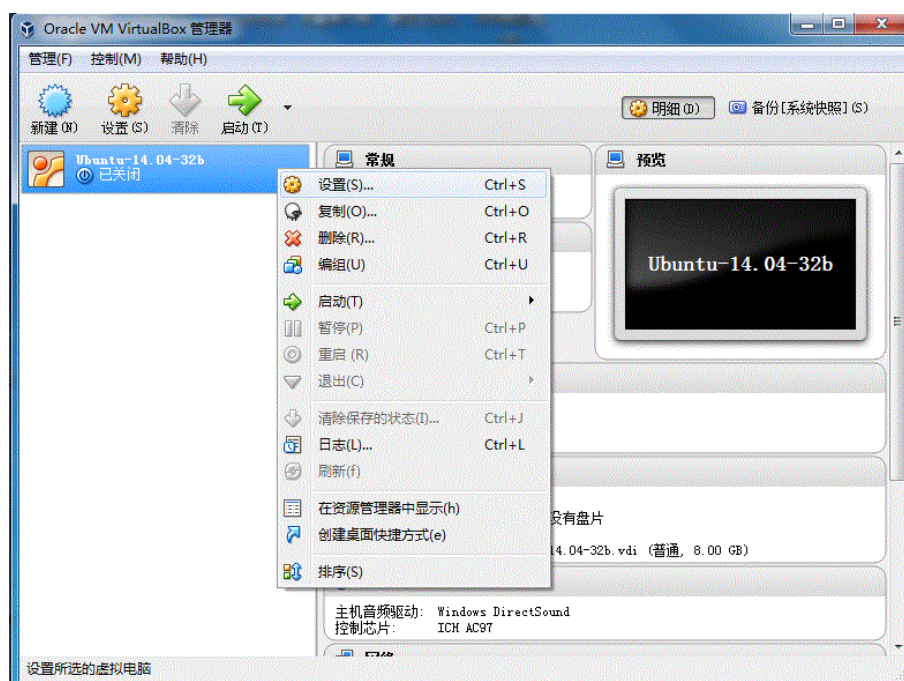
VMDK 是 VMWare 虚拟机软件创建的虚拟硬盘文件格式。我觉得 VDI 是 VirtualBox 自己的格式，应该支持得更好，所以我建议选 VDI 格式。然后再[下一步]出现如下对话框来选择硬盘的分配类型。



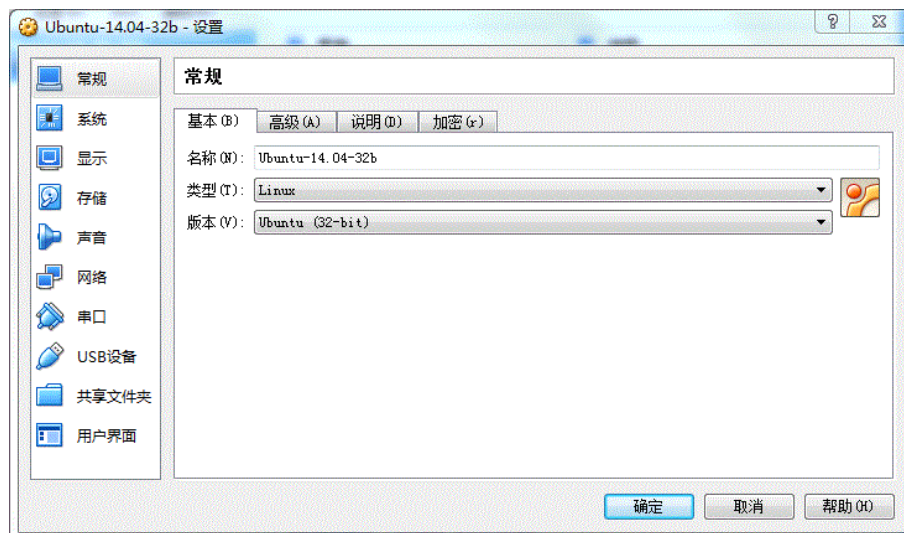
VirtualBox 会在主机硬盘上分配一个文件，作为虚拟硬盘。从上面图片的描述可以清楚的知道，动态分配模式下的虚拟硬盘对应的主机文件是动态分配大小的。固定大小模式，是在主机硬盘上分配一个固定大小的文件。



上面的文本框是输入虚拟硬盘占用的文件的文件名，点击右边的文件夹图标，可以更改虚拟硬盘路径。下面的滑块是设置虚拟硬盘的容量。如果上一步选择的是动态分配模式的话，这里可以随意设置硬盘的容量，尽可能的大都可以。如果选择的是固定大小模式，就需要稍微慎重，尤其是主机硬盘空间紧张的情况下。对于 Ubuntu 系统，一般容量 8G 可以了。如果做嵌入式开发的话，可以另外挂一个虚拟硬盘，10G 左右。然后点击[创建]，这样虚拟电脑就创建完成了。只是这台电脑还是空的，没有安装任何操作系统，还不能工作。虚拟电脑新建完成后，在 VirtualBox 主窗口的左边栏会列出所有的虚拟电脑。刚才新建的虚拟电脑也会出现在里面，如下图所示。

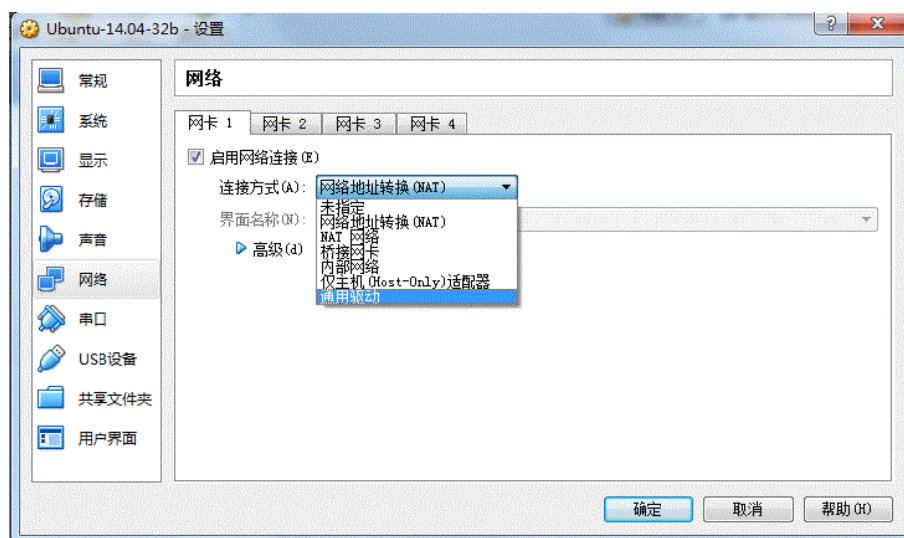


如果觉得之前的设置不正确，希望进行修改的话，还可以选中该虚拟电脑项目，在右键菜单里选中[设置]，会弹出如下对话框来进行详细设置。



## 2.3 网络设置

VirtualBox 主要有五种网络接入模式，如下图所示，它们分别是：NAT 网络地址转换模式(Network Address Translation)，NAT 网络模式，桥接网卡模式（Bridged Adapter），内部网络模式（Internal），和 Host-Only 适配器模式。



1) NAT 模式是最简单的实现虚拟电脑上网的方式，简单的讲，虚拟电脑是主机的一个程序，虚拟机访问网络的所有数据都是有主机提供，主机能上网，虚拟机也能上网。但是虚拟电脑并不真实存在于网络中，主机与网络中的其它机器都不能直接访问到虚拟电脑。也就是说网络是单向的，只有虚拟电脑可以通过网络访问到主机，而主机无法通过网络访问到虚拟电脑。另外虚拟电脑之间也是彼此完全独立的，相互间无法通过网络访问。

2) 网络地址转换(NAT)模式，实际上就是 NAT 模式，但是为了解决主机不能访问虚拟电脑的问题，该模式增加了端口转发功能。这样主机能够把通过访问主机指定的网络端口的数据包发给 VirtualBox，然后 VirtualBox 发给相应的虚拟电脑。

3) 网桥模式是通过把虚拟电脑的模拟网卡和主机的真实网卡桥接在一起，直接连入网络。类似于用一条虚拟的网线，把主机和虚拟电脑连接在一起了。这样虚拟电脑就跟主机在同一个网络中，虚拟电脑会被网络的服务器分配到独立的 IP 地址，所有网络功能完全和在网络中的真实机器一样。因此主机和虚拟电脑可以相互访问。

4) 内部网络模式, 虚拟机软件下的所有虚拟电脑组建的一个内部网络, 这个网络与主机网络完全断开, 相互不能访问。虚拟机内部的虚拟电脑可以相互访问, 前提是两台虚拟电脑的网络设置在同一个网段内。

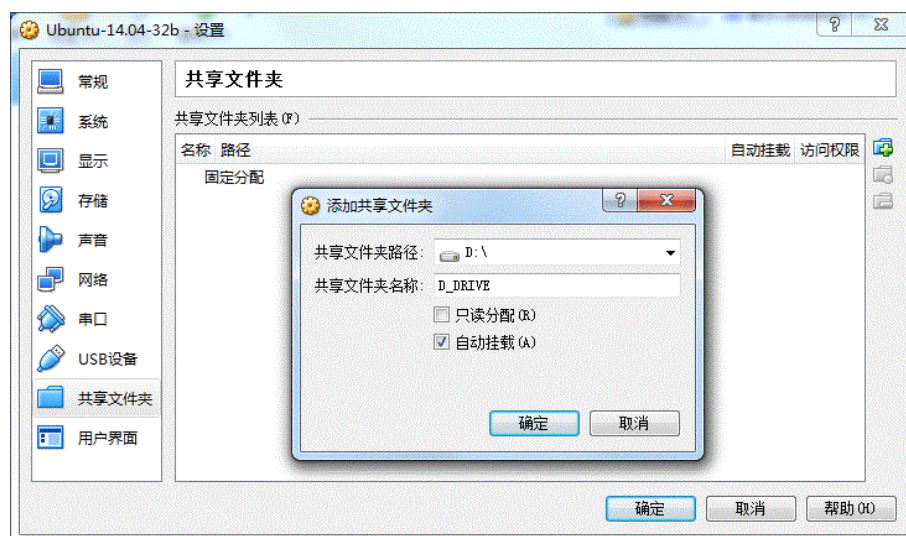
5) Host-Only 适配器模式, 大家可能已经注意到, 安装完 VirtualBox 后, 在主机的网卡连接管理窗口里多出来了一个网卡: VirtualBox Host-Only Network。这个网卡在之前的网络模式描述中都没有提到。它就是用于该模式的。在该模式中, 虚拟机内部是一个独立的内部局域网, 默认情况下虚拟电脑与主机间相互独立不能访问。这就很像内部网络模式了, 但是不同的是, 这个内部局域网是与主机的这个虚拟网卡接在一起的, 相当于它是主机下面的个子网络。虚拟主机能否访问到外部网络或者怎么访问外部网络, 就由这个虚拟网卡的设置决定。我们可以通过设置网卡共享或者桥接的方式来实现内部虚拟电脑。

对于虚拟机访问网络的设置, 上面提到的主机模式设置相对复杂, 需要了解 Windows 系统一些网络知识才能连上主机网络, 不推荐使用。如果只是为了实现虚拟电脑能访问互联网, 使用 NAT 模式最简单。要实现虚拟机与主机网络的相互访问, 还是网桥模式比较简单, 不需要特别的设置。如果开发板也接入了主机网络, 虚拟电脑和开发板之间也很容易实现网络连接。如果主机有多个网卡, 桥接模式可以把虚拟电脑网络桥接到任何一个主机网卡上, 成为该网卡所在局域网的一员。

## 2.4 文件共享

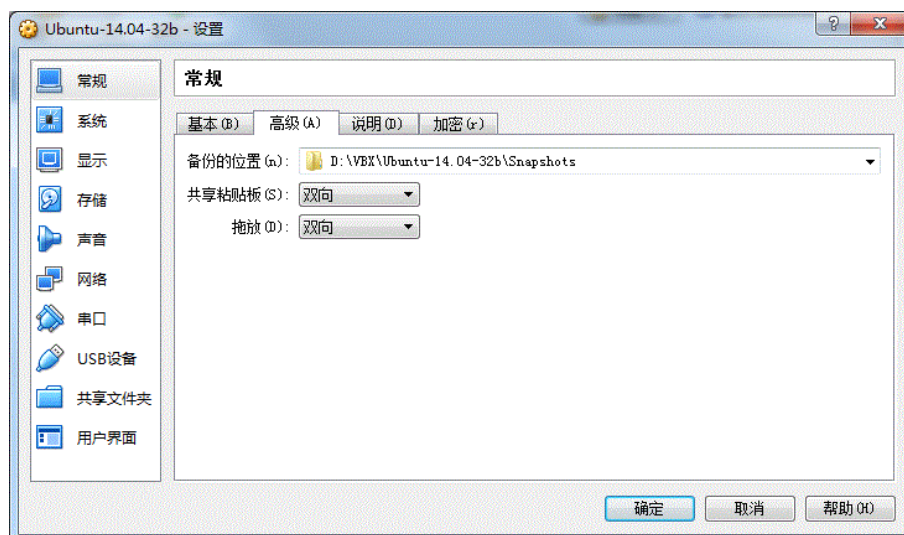
VirtualBox 为了实现虚拟电脑与主机的数据交互, 除了通过网络的方式外, 也增加了共享文件夹功能。在 VirtualBox 主界面左侧的虚拟电脑列表中选择相应的虚拟电脑, 在右键菜单里选择[设置], 在弹出的对话框中选择[共享文件夹], 如下图对话框所示。点击对话框最右边上“+”图标, 弹出如图小的对话框。在里面选择需要共享的目录和共享目录的名称。如果选择自动挂载的话, 再虚拟系统启动过程中, 可以自动挂载到系统中。如 Ubuntu 系统, 也可以在系统启动后, 通过如下命令, 手动挂载。

`sudo mount -t vboxsf D_DRIVE /mnt/d_disk` # D\_DRIVE 对话框里面输入的共享名称。



除了共享目录的方式外, VirtualBox 还提供了共享粘贴板和拖拽的方式。如下设置对话框所示, 可以设置粘贴板单向或者双向支持的功能。拖放的单向和双向支持。当然为了方便操作, 我们当然会选择双向支持了。

当然这些数据共享的功能, 需要虚拟机安装了操作系统, 并在在操作系统里面安装了 VirtualBox 的增强功能才能生效。下面一节就先介绍一下怎么安装增强功能。



## 2.5 VirtualBox 的增强功能

需要注意的是安装增强功能是安装在虚拟机里面的操作系统里。到目前为止，我们新建的虚拟机还是空的，没有安装系统。因此这一节的操作，需要按照后面的操作安装了系统后，再回来操作。

那么怎么安装增强功能呢。首先在虚拟电脑控制菜单里选择[设备] → [安装增强功能...]，如下图所示。这样就会自动给虚拟电脑的光驱中加载入增强功能的 ISO 文件。在虚拟电脑里打开光驱，就可以自动安装增强功能了。



在安装了[增强功能]后，为虚拟电脑增加了很多便捷的功能。如 U 盘访问。在 U 盘插入主机 USB 端口后，在虚拟系统的控制菜单里面选择[设备] → [USB]，在 USB 下一级菜单里面就会列出已经插入的 U 盘的名称，选中这个 U 盘名称后，该 U 盘就会自动挂载到虚拟系统里面了。对于其它 USB 设备，如 J-Link 等开发工具，也是相同的操作。

安装了[增强功能]后，可以让你的虚拟电脑里任意设置你的桌面分辨率大小。鼠标也不会被锁定在虚拟机里面。它会判断你的鼠标的位置，如果它移动到虚拟系统的窗口外面，会自动释放鼠标设备。

### 3. Ubuntu 主机设置

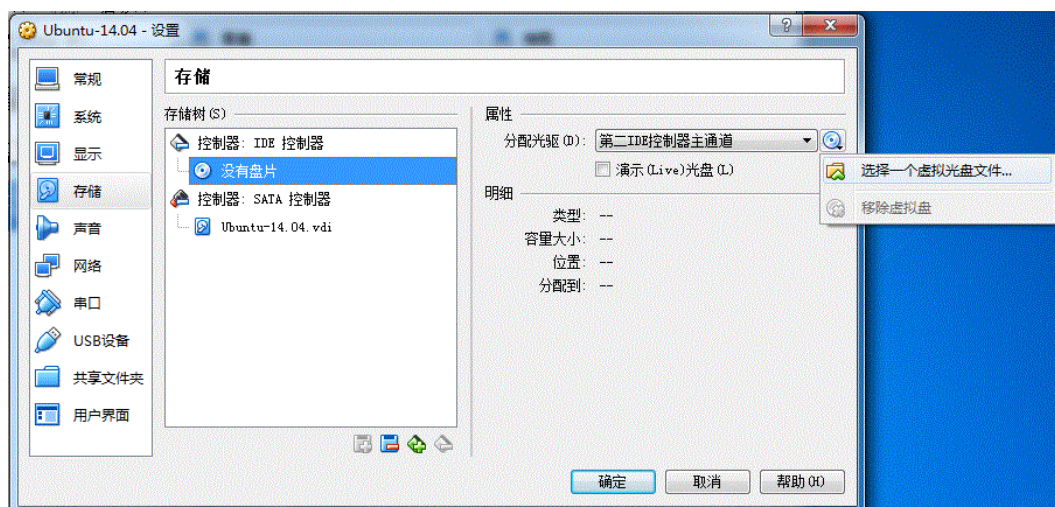
#### 3.1 Ubuntu 下载

进入如下 Ubuntu 官方网站：<http://www.ubuntu.com/download/desktop>，可以进入 Ubuntu 桌面版下载页面，为了兼容性更好，请选择 32-bit 的版本，如下图所示，点击[Download]按钮可以下载 Ubuntu 14.04.03 LTS 版本，文件名为：ubuntu-14.04.3-desktop-i386.iso。

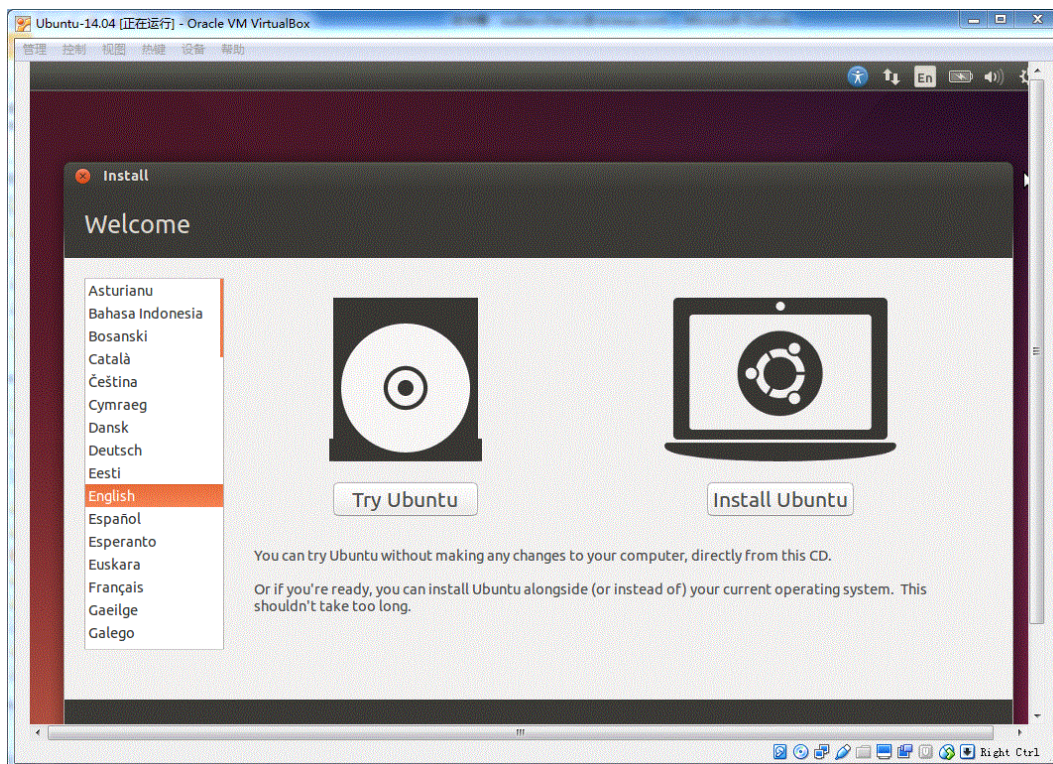


#### 3.2 Ubuntu 的安装

到这一步，目前我们新建的虚拟电脑如下图所示，在虚拟电脑的设置对话框里面选择[存储]，再选中中间的光驱控制器，然后点击最右边的[光盘]图标，弹出一个小菜单，点击[选择一个虚拟光盘文件...]后，就可以弹出文件对话框来选择 Ubuntu ISO 文件，这样就可以把 Ubuntu 安装光盘装载到虚拟电脑的光驱里面了。



在上面设置完成后，就可以点击 VirtualBox 主界面的[启动]图标来启动虚拟电脑。这时，该虚拟电脑的硬盘是空白，软驱也没有软盘，只有光驱 Ubuntu 安装光盘可以支持启动电脑。按照系统设置的启动顺序，也是由光驱来启动电脑。等待几分钟光盘数据加载的过程，就可以自动进入如下 Ubuntu 的安装界面。具体的安装过程这里就不赘述了，按照向导一步一步操作即可。



技巧：在进行嵌入式开发的时候，推荐在 Ubuntu 系统中使用两个虚拟硬盘。一个用于 Ubuntu 系统，另一个用于用户开发项目的文件数据。也就是还需要再新建一个虚拟硬盘，加载到虚拟电脑系统中，在 Ubuntu 系统中把它挂载到用户目录下。为什么呢，因为虚拟机跑了一段时间后，虚拟硬盘所占的文件会越来越大，虚拟机也会越跑越慢。当原有用户文件不再需要，或者需要再开发另外一个全新的项目，只需要另外再新建一个虚拟硬盘，把原来的用户目录所在的虚拟硬盘卸载掉，把新建的硬盘挂载上就可以了。这样的话，新的开发环境又会变得比较轻便，也不需要为了新项目重新再安装一个新的虚拟系统，也避免了新系统的重复设置和软件安装。

下面再描述一下在做嵌入式开发时，Ubuntu 系统需要做的一些设置。

### 3.3 Linux 开发必备工具包安装

在做 Linux 嵌入式开发时，需要使用如下命令安装两个必备的工具包：

```
user@user-v:/home/rza$ sudo apt-get install u-boot-tools
```

```
user@user-v:/home/rza$ sudo apt-get install ncurses-dev
```

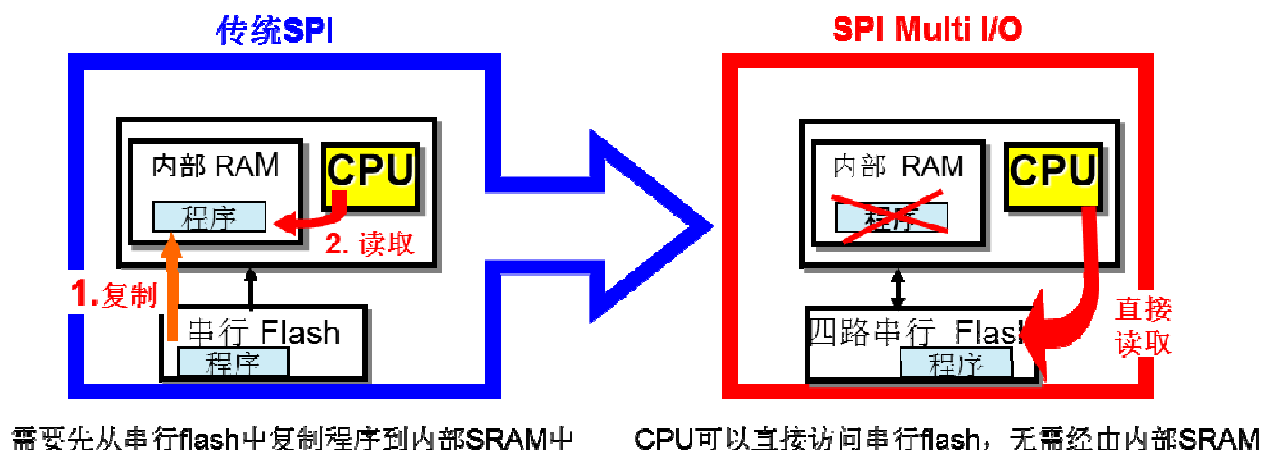
### 3.4 其它工具

在一般的 Linux 嵌入式项目开发中，为了便于应用开发，需要安装 FTP、NFS 等网络服务。FTP 主要为了方便直接下载 Linux Kernel、根文件系统等文件到开发板上。NFS 则可以把根文件系统直接放在主机上，不必烧录到开发板中，节省烧录时间，开发板可以直接同步使用主机上的根文件系统，可以大大地方便调试。但是在我们的 RZ/A1 XIP 版 Linux 上，安装这些网络服务不是很必要。这里就不详细介绍了，在网上可以搜索到很多安装攻略。

## 4. RZ/A1 Linux BSP 下载及编译

### 4.1 XIP 版 Linux 介绍

一般的嵌入式 Linux 开发，由于 Kernel 和根文件系统比较大，为了节省成本，代码被烧录进 NAND-Flash 或者 SPI-Flash 里面。而 Flash 的代码不能够直接执行，必须先由 u-boot 启动后，把 Kernel 和根文件系统装载到 RAM，然后再启动 Linux。而 RZ/A1 系列芯片有个非常有用的功能，可以支持 CPU 直接访问串行 Flash，而无需经由 RAM。这样代码就可以不必复制到 RAM，而在 SPI-Flash 里就可以直接运行，英语表达为：eXecute In place，简称为 XIP。如下图所示。



RZ/A1 系列芯片内置了大容量的 SRAM，但是对于运行普通的 Linux 还是远远不够的，为了节省内置的 SRAM，同时利用 RZ/A1 支持的 XIP 功能，在修改 Linux 的配置后，可以使 Linux 系统直接在 SPI-Flash 上运行，只需要消耗少量的数据 RAM。这样 RZ/A1 可以实现不扩展外置 SDRAM 情况下，也可以运行 Linux 系统，最大程度地发挥了 RZ/A1 的性价比。

### 4.2 RZ/A1 Linux BSP 下载

你可以从网址[https://github.com/renesas-rz/rskrza1\\_bsp\\_releases](https://github.com/renesas-rz/rskrza1_bsp_releases)下载最新的基于 RZ/A1 的 Linux BSP，然后解压缩到用户目录下。也可以直接在 Ubuntu 的终端用如下命令操作。

```
user@user-v:/home/rza$ wget https://github.com/renesas-rz/rskrza1_bsp_releases/blob/master/RSKRZA1-BSP-V1.1.1.tar.bz2
user@user-v:/home/rza$ tar x RSKRZA1-BSP-V1.1.1.tar.bz2
```

### 4.3 Linux 编译

进入上一步解压的目录：RSKRZA1-BSP-V1.1.1，里面有个 setup\_env.sh 和 build.sh 文件，setup\_env.sh 用于设置编译输出及 GCC 工具链的目录，当然也可以使用默认的，不修改。Build.sh 文件功能很强大，可以用于获取工具链，编译 u-boot，Kernel，和根文件系统。在 RSKRZA1-BSP-V1.1.1 下输入如下命令，就会显示出操作菜单，提示用户该如何使用该命令，如下所示。

```
user@user-v:/home/rza/RSKRZA1-BSP-V1.1.1$ ./build.sh
What do you want to build?
./build.sh get_toolchain      : Downloads pre-built Linaro toolchain for ARM
./build.sh kernel             : Builds Linux kernel. Default is to build uImage
./build.sh u-boot             : Builds u-boot
./build.sh buildroot          : Builds Root File System
./build.sh librzjpeg          : RZ/A1 JPEG HW decode example application
./build.sh axfs               : Builds an AXFS image from the last Buildroot output

You may also do things like:
./build.sh kernel menuconfig  : Open the kernel config GUI to enable options/drivers
./build.sh kernel rskrza1_xip_defconfig : Switch to XIP version of the kernel
./build.sh kernel xipImage    : Build the XIP kernel image
./build.sh buildroot menuconfig : Open the Buildroot config GUI to select additional apps to build
```

在命令行依次输入如下命令。

\$ ./build.sh get\_toolchain: 下载已经预编译好的 ARM 交叉编译工具链，并解压到设定的目录

\$ ./build.sh u-boot: 编译 u-boot，将会生成 u-boot.bin 文件。第一次运行该命令时，会自动下载 u-boot 源代码，解压缩到设定的目录，并把 RSK 板的修改补丁 patch 到解压的源代码里。

\$ ./build.sh kernel rskrza1\_xip\_defconfig: 选择当前的 Kernel 配置为运行于 RSK 板的 XIP 版 Linux Kernel，第一次运行 kernel 参数时，会自动下载 Linux Kernel 源代码，解压缩到设定的目录，并把 RSK 板的修改补丁 patch 到解压的源代码里。

\$ ./build.sh kernel menuconfig: 根据需要修改 Kernel 配置，不修改的话，就不要执行该命令。

\$ ./build.sh kernel xipImage: 编译 Linux kernel，将会生成 xipImage 的二进制文件。

\$ ./build.sh buildroot menuconfig: 根据需要修改根文件系统配置和选择需要的工具。第一次运行 buildroot 参数时，会自动下载 buildroot 源代码，解压到设定的目录。

\$ ./build.sh buildroot: 编译 buildroot，将会生成目标板运行的根文件系统目录。

\$ ./build.sh axfs: 可以把上一步 buildroot 生成的根文件系统打包成 axfs 格式的镜像二进制文件。

## 4.4 目标文件烧录

到现在 Linux 系统的二进制文件全部编译完成了，那么怎么把生成的 bin 文件烧录到 Flash 中呢。这里推荐使用 Segger J-Link 工具。

### 4.4.1. Segger J-Link 工具安装:

首先进入 Segger 官网网址: <https://www.segger.com/jlink-software.html>，选择[DEB Installer 32-bit version]下载。直接在 Ubuntu 中运行安装。推荐安装 5.0 版以上的 J-Link 工具，因为它能支持 RZ/A1 全系列的芯片，也支持 RSK 板 8 兆以上的二进制文件烧录。

### 4.4.2. J-Link 烧录脚本:

在 4.2 Section 里解压的 RSKRZA1-BSP-V1.1.1/Extra/ J-Link\_QSPI\_Program 目录下，即包含了 J-Link 脚本工具，首先需要把上一步 4.3 Section 编译生成的二进制文件复制到该目录，可以参考如下命令复制。

```
# 当前目录在 RSKRZA1-BSP-V1.1.1
$ cp output/u-boot-2015.01/u-boot.bin Extra/J-Link_QSPI_Program
$ cp output/linux-3.14/arch/arm/boot/dts/r7s72100-rskrza1.dtb Extra/J-Link_QSPI_Program/dts/r7s72100-rskrza1.dtb.bin
$ cp output/linux-3.14/arch/arm/boot/xipImage Extra/J-Link_QSPI_Program/xipImage.bin
```

#### 4.4.3. 烧录操作:

首先需要更改 RSK 板的 SW6 设置 SW6-1=OFF, SW6-3=OFF, 其余保持 ON 即可。把 J-Link ICE 工具插到 RSK 板上, USB 线插入电脑主机的 USB 端口, 连通 RSK 开发板的电源。然后在 Ubuntu 虚拟机的控制菜单里面如下图, 选择 Segger J-Link 项, 这样就可以把 J-Link 的控制权交给 Ubuntu 系统了。如果是第一次这么操作, Ubuntu 系统会自动安装 J-Link 的驱动。



接着使用如下命令进行烧录:

```
user@user-v:/home/rza/RSKRZA1-BSP-V1.1.1$ cd Extra/J-Link_QSPI_Program
user@user-v:/home/rza/RSKRZA1-BSP-V1.1.1/Extra/J-Link_QSPI_Program$ ./Program_QSPI.sh
```

#### OPTIONS

- 1 = Program u-boot
- 2 = Program Device Tree Blob
- 3 = Program Kernel (uImage)
- 4 = Program Kernel (xipImage)
- 5 = Program Rootfs (squashfs)
- 6 = Program Rootfs (axfs)
- 9 = Exit

Choose option:

按照如上提示, 依次输入 1, 2, 4, 6, 9 后烧录完成, RSK 开发板断电, 移除 J-Link ICE 工具即可。

#### 4.5 USB 转串口驱动安装

在 Linux 嵌入式开发中, 通过串口来控制 and 显示输出信息是必不可少的。在 RSK+RZ/A1 板上, 在 RZ/A1 的 UART 上连接了串口转 USB 的转换芯片, 这样只需要一根 Mini USB 线插入 CN18 端口, 另一端连接 USB 电脑即可。另外关键是需要还在 PC 主机上安装该转换芯片对应的 USB 转串口驱动。该驱动

文件也包含在：RSKRZA1-BSP-V1.1.1/Extra/USB-Serial-Drivers(win)目录下。需要把它复制到主机的 Windows 系统中安装。该驱动的安装方法，与一般的驱动安装方法一样，这里不描述了。

## 4.6 Linux 启动

USB 转串口安装好后，开启一个串口调试工具，选择该 USB 转串口的串口端口号，设置波特率为：115200，数据位：8，奇偶校验：无，停止位：1，如下图所示。



RSK 板上电后，就可以在串口调试工具里显示 u-boot 启动信息，如下：

```
U-Boot 2015.01 (Aug 14 2015 - 13:24:53)
```

```
I2C: ready
```

```
DRAM: 10 MiB
```

```
Flash: 64 MiB
```

```
SF: Detected S25FL512S_256K with page size 512 Bytes, erase size 256 KiB, total 64 MiB
```

```
In: serial
```

```
Out: serial
```

```
Err: serial
```

```
i2c bus is busy.
```

```
i2c bus is busy.
```

```
      SPI Flash Memory Map
```

```
-----
      Start  Size  SPI
u-boot: 0x00000000 0x080000 0
env: 0x00080000 0x040000 0
DT: 0x000C0000 0x040000 0
Kernel: 0x00100000 0x280000 0+1 (size*=2)
rootfs: 0x00400000 0x1C00000 0+1 (size*=2)
```

```
Net: sh_eth
```

```
Hit any key to stop autoboot: 0
```

```
=>
```

在上面的提示符输入 `print` 命令可以显示很多环境变量，已经设置好了很多运行命令。对于这里 XIP 版 Linux，输入命令：`run xa_boot`，即可启动 Linux 系统。也可以输入 `set bootcmd "run xa_boot"`，然后输入 `saveenv`。这样每次上电重启后，就不需要再输入命令，上电后可以自动进入 Linux 系统。

## 5. 应用程序开发

### 5.1 代码工具 Eclipse 安装

为了便于 Linux 的应用开发，这里推荐安装 Eclipse 工具。Eclipse 是著名的跨平台的自由集成开发环境（IDE）。最初主要用来 Java 语言开发，通过安装不同的插件 Eclipse 也可以支持不同的计算机语言，比如 C，C++ 和 Python 等开发工具。Eclipse 有很多优点：方便建立交叉编译环境，自动生成 `makefile`；在 Eclipse 环境里阅读和写代码也非常便利；而且它是一个开源代码的工具，公司可以免费获得该软件的使用权。下面介绍一下安装方法。

大家知道 Eclipse 是基于 JAVA 开发的，因此需要先安装 Java 套件。安装 Java 可以选择直接去 java 官网下载 Linux 版的安装包，<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>。也可以使用如下命令方式安装。

```
user@user-v:/home/rza$ sudo apt-get purge openjdk*           # 删除已安装的旧版本
user@user-v:/home/rza$ sudo apt-get repository ppa:webupd8team/java # 添加 PPA 源
user@user-v:/home/rza$ sudo apt-get update                  # 更新下载源
user@user-v:/home/rza$ sudo apt-get install oracle-java7-installer # 也可以直接在 ubuntu 软件中心安装
```

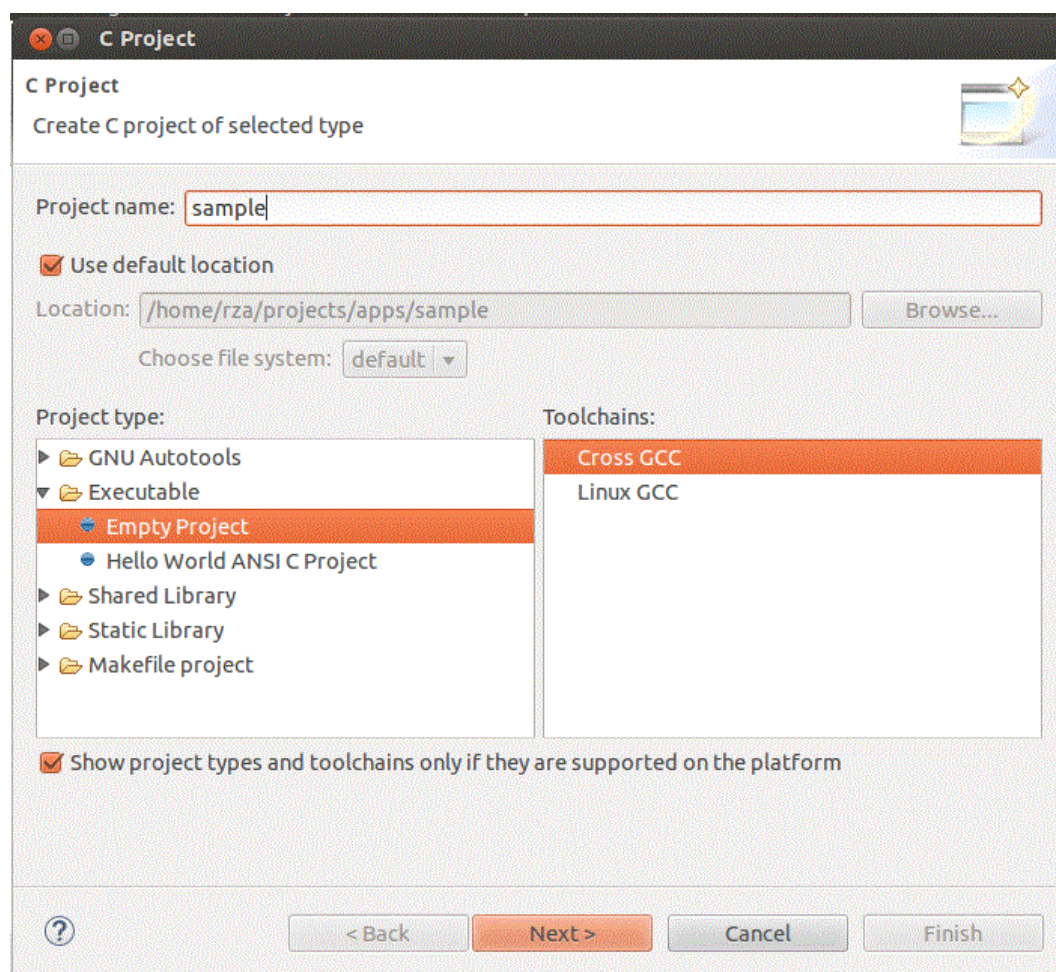
然后进入 eclipse 官网<http://www.eclipse.org/downloads/?osType=linux>，选择下载 Eclipse IDE for C/C++ Developers。解压缩到本地路径即可，如：/opt/eclipse。

## 5.2 使用 Eclipse 建立工程

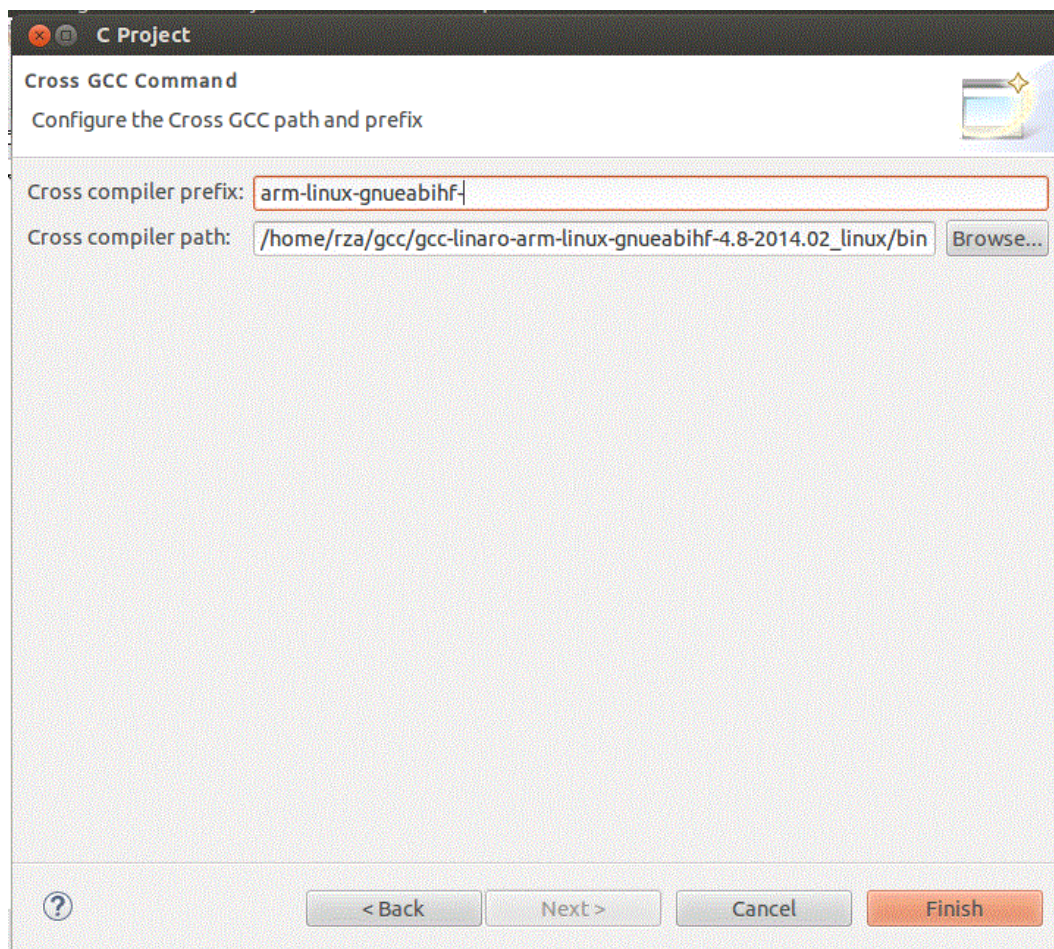
运行如下命令打开 eclipse。

```
user@user-v:/home/rza$ cd /opt/eclipse           # 假设 eclipse 安装路径为/opt/eclipse
user@user-v:/home/rza$ ./eclipse                 # 启动 eclipse
```

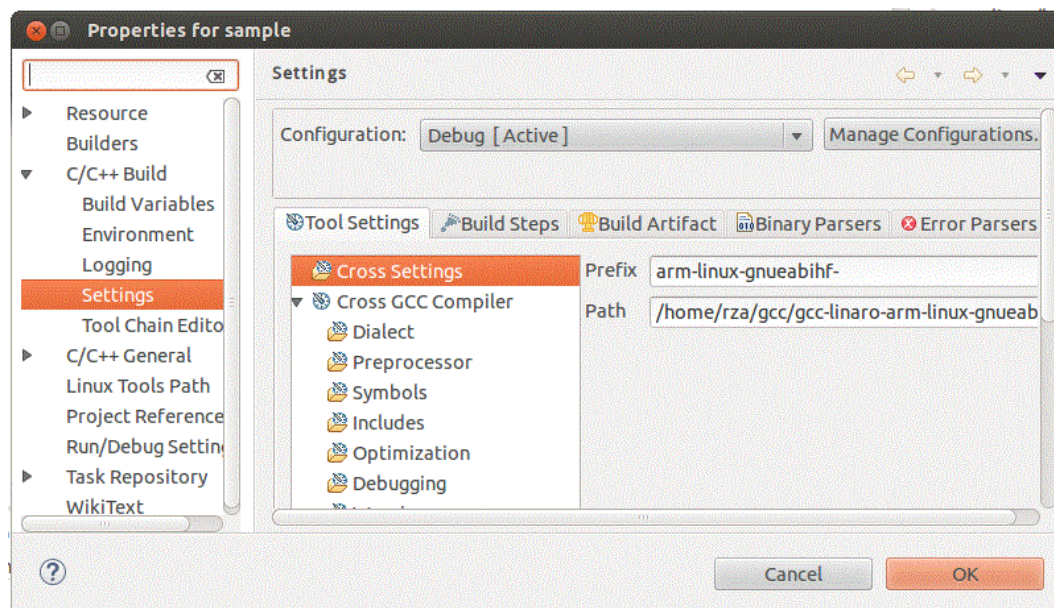
在 eclipse 菜单里选择[File] → [New] → [C Project]，弹出如下的 C Project 新工程对话框，需要输入工程的名称，在工程类型里面选[Empty Project]，在[Toolchains]里面选择[Cross GCC]。



在上面的对话框里面选择[Next >]，会进入编译配置对话框，这一步没有什么需要修改的，再选择[Next >]进入如下交叉编译工具链选择的对话框。



在[Cross compiler prefix]栏输入编译工具命令的前缀，如：arm-linux-gnueabi。在[Cross compiler path]栏输入前面 4.3 Section 设定的 GCC 编译工具的路径。然后点击[Finish]按钮，新建工程就完成了。如果还需要对该工程修改配置，也可以在主界面左边的工程浏览窗口选中该工程 Sample，在右键菜单中选择[Properties]，弹出如下工程属性对话框，进行需要的设置。



### 5.3 调试方法介绍:

写好编译完成的应用程序，怎么下载到 Linux 运行调试呢？这里简单介绍三种途径。

第一种，把生成的应用程序复制到根文件系统目录里，然后用 `./build.sh axfs` 命令重新打包根文件系统镜像文件。这种方法简单，但是每次要重新下载根文件系统的镜像文件到 Flash，比较花时间，不利于调试阶段，在生产阶段可以采用。

第二种，把应用程序复制到 U 盘里，然后在开发板的 Linux 系统里挂载该 U 盘，并执行里面的命令。这种方法调试阶段也比较方便，但是挂载 U 盘和在 U 盘目录下运行应用程序会额外占用一部分 RAM，而且与最终产品的运行环境有差异。

第三种，把应用程序全部放在一个目录下，用 `axfs` 工具把该目录也打包成 `axfs` 格式的镜像文件。再用 `J-Link` 工具把该镜像文件烧录到 Flash 指定的地址。在开发板的系统里使用如下命令把该区域挂载到 Linux 系统下，然后进入该挂载目录运行应用程序。

```
$ mount -t axfs -o physaddr=0x1A000000 none /mnt/axfs
```

也可以把下面的内容保存为文件 `S98_load_app`，使其具有可执行属性，然后放到根文件系统的 `/etc/init.d/` 目录下面，这样每次嵌入式 Linux 系统启动后，可以自动加载应用程序的 `axfs` 镜像文件。

```
#!/bin/sh
# Modify by Chen Xudian 20150918
# Only run at system boot, not at system shutdown
if [ "$1" != "start" ]; then
    exit
fi

# Mount AXFS if exists
# Lookfor "Adva", short for "Advanced XIP FS" starting at 1A000004
AXFS_EXIST=$(/root/bin/mem r 1 1A000004 1 | grep 61766441)

# You can pass "skipaxfsmnt" on the kernel command line to skip this
SKIP_AXFS_MNT=$(dmesg | grep skipaxfsmnt)
if [ "$SKIP_AXFS_MNT" != "" ]; then
    AXFS_EXIST=""
fi

if [ "$AXFS_EXIST" != "" ]; then
    echo "<> Mounting AXFS on /mnt/axfs" > $SERIAL_TTY
    mount -t axfs -o physaddr=0x1A000000 none /mnt/axfs
    if [ -e /mnt/axfs/autorun.sh ]; then
        echo "<> Running /mnt/axfs/autorun.sh..." > $SERIAL_TTY
        /mnt/axfs/autorun.sh
    fi
fi
```

### 5.4 示例工程

我们在上面新建的 `eclipse` 工程里添加如下 `sample.c` 文件，在 `eclipse` 的菜单里选择 [Project] → [Build Project]，在 `Debug` 目录下可以生成可以运行的 `sample` 文件。

```
// File Name: sample.c
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    printf("!!!Hello World!!!\n"); /* prints !!!Hello World!!! */
    return EXIT_SUCCESS;
}
```

在该工程的 Debug 目录下新建 output 目录，并把 sample 文件复制到 output 目录下，然后用 mkfs.axfs 命令打包，（mkfs.axfs 命令在），并把生成 apps.axfs.bin 的文件文件用 J-Link 工具烧录到 0x1A000000 地址。

```
$ cd /home/rza/projects/RSKRZA1-BSP-V1.1.1/output/axfs/
$ ./mkfs.axfs -s -a /home/rza/projects/apps/sample/Debug/output apps.axfs.bin
```

Apps.axfs.bin 文件烧录完成，RSK 开发板重新上电后，因为 4.7.3 中可以自动挂载 0x1A000000 的应用程序到/mnt/axfs 目录，因此在开发板的 Linux 系统启动后输入如下命令，可以显示执行的结果如下：

```
$ cd /mnt/axfs/           # 进入应用程序目录
$ ls                      # 显示该目录的内容
sample
$ ./sample                # 运行应用程序
!!!Hello World!!!        # 应用程序执行结果
$
```

## 公司主页和咨询窗口

瑞萨电子主页

<http://cn.renesas.com/>

RZ/A1H LSI URL:

<http://cn.renesas.com/products/mpumcu/rz/rza/index.jsp>

咨询

<http://www.renesas.com/inquiry>

[contact.china@renesas.com](mailto:contact.china@renesas.com)

所有商标及注册商标均归其各自拥有者所有。

版本更新历史

| 版本   | 日期          | 描述 |      |
|------|-------------|----|------|
|      |             | 页码 | 内容   |
| 1.00 | Oct.09.2015 | —  | 初始版本 |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |
|      |             |    |      |

## 产品使用时的注意事项

本文对适用于单片机所有产品的“使用时的注意事项”进行说明。有关个别的使用时的注意事项请参照本文档以及Technical Update.

### 1. 未使用的引脚的处理

【注意】将未使用的引脚按照正文的“未使用引脚的处理”进行处理。

CMOS产品的输入引脚的阻抗一般为高阻抗。如果在开路的状态下运行未使用的引脚，由于感应现象，外加LSI周围的噪声，在LSI内部产生穿透电流，有可能被误认为是输入信号而引起误动作。

未使用的引脚，请按照正文的“未使用引脚的处理”中的指示进行处理。

### 2. 通电时的处理

【注意】通电时产品处于不定状态。

通电时，LSI内部电路处于不确定状态，寄存器的设定和各引脚的状态不定。通过外部复位引脚对产品进行复位时，从通电到复位有效之前的期间，不能保证引脚的状态。

同样，使用内部上电复位功能对产品进行复位时，从通电到达到复位产生的一定电压的期间，不能保证引脚的状态。

### 3. 禁止存取保留地址（保留区）

【注意】禁止存取保留地址（保留区）

在地址区域中，有被分配将来用作功能扩展的保留地址（保留区）。因为无法保证存取这些地址时的运行，所以不能对保留地址（保留区）进行存取。

### 4. 关于时钟

【注意】复位时，请在时钟稳定后解除复位。

在程序运行中切换时钟时，请在要切换成的时钟稳定之后进行。复位时，在通过使用外部振荡器（或者外部振荡电路）的时钟开始运行的系统中，必须在时钟充分稳定后解除复位。另外，在程序运行中，切换成使用外部振荡器（或者外部振荡电路）的时钟时，在要切换成的时钟充分稳定后再进行切换。

### 5. 关于产品间的差异

【注意】在变更不同型号的产品时，请对每一个产品型号进行系统评价测试。

即使是同一个群的单片机，如果产品型号不同，由于内部ROM、版本模式等不同，在电特性范围内有时特性值、动作容限、噪声耐量、噪声辐射量等不同。因此，在变更不认同型号的产品时，请对每一个型号的产品进行系统评价测试。

## Notice

- Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
- Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
- Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
- You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
- Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.  
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc. Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
- You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
- Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
- Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
- Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
- It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
- This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
- Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.  
(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.  
(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

以下"注意事项"为从英语原稿翻译的中文译文, 仅作参考译文, 英文版的"Notice"具有正式效力。

## 注意事项

- 本文件中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文件中的电路、软件 and 相关信息, 请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失, 瑞萨电子不承担任何责任。
- 在准备本文件所记载的信息的过程中, 瑞萨电子已尽量做到合理注意, 但是, 瑞萨电子并不保证这些信息都是准确无误的。用户因本文件中所记载的信息的错误或遗漏而遭受的任何损失, 瑞萨电子不承担任何责任。
- 对于因使用本文件中的瑞萨电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为, 瑞萨电子不承担任何责任。本文件所记载的内容不应视为对瑞萨电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
- 用户不得更改、修改、复制或者以其他方式部分或全部地非法使用瑞萨电子的任何产品。对于用户或第三方因上述更改、修改、复制或以其他方式非法使用瑞萨电子产品的行为而遭受的任何损失, 瑞萨电子不承担任何责任。
- 瑞萨电子产品根据其质量等级分为两个等级: "标准等级"和"高质量等级"。每种瑞萨电子产品的推荐用途均取决于产品的质量等级。如下所示:  
标准等级: 计算机、办公设备、通讯设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备以及工业机器人等。  
高质量等级: 运输设备 (汽车、火车、轮船等)、交通控制系统、防灾系统、预防犯罪系统以及安全设备等。  
瑞萨电子产品无意用于且未被授权用于可能对人类生命造成直接威胁的产品或系统及可能造成人身伤害的产品或系统 (人工生命维持装置或系统、植于体内的装置等) 中, 或者可能造成重大财产损失的产品或系统 (核反应堆控制系统、军用设备等) 中。在将每种瑞萨电子产品用于某种特定应用之前, 用户应先确认其质量等级。不得将瑞萨电子产品用于超出其设计用途之外的任何应用。对于用户或第三方因将瑞萨电子产品用于其设计用途之外而遭受的任何损害或损失, 瑞萨电子不承担任何责任。
- 使用本文件中记载的瑞萨电子产品时, 应在瑞萨电子指定的范围内, 特别是在最大额定值、电源工作电压范围、移动电源电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用瑞萨电子产品而产生的故障或损失, 瑞萨电子不承担任何责任。
- 虽然瑞萨电子一直致力于提高瑞萨电子产品的质量和可靠性, 但是, 半导体产品有其自身的具体特性, 如一定的故障发生率以及在某些使用条件下会发生故障等。此外, 瑞萨电子产品均未进行防辐射设计。所以请采取安全防护措施, 以避免当瑞萨电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计 (包括但不限于冗余设计、防火控制以及故障预防等)、适当的老化处理或其他适当的措施等。由于难于对微机软件单独进行评估, 所以请用户自行对是最终产品或系统进行安全评估。
- 关于环境保护方面的详细内容, 例如每种瑞萨电子产品的环境兼容性等, 请与瑞萨电子的营业部门联系。使用瑞萨电子产品时, 请遵守对管制物质的使用或含量进行管理的所有相应法律法规 (包括但不限于《欧盟ROHS指令》)。对于因用户未遵守相应法律法规而导致的损害或损失, 瑞萨电子不承担任何责任。
- 不可将瑞萨电子产品和技术用于或者嵌入日本国内或海外相应的法律法规所禁止生产、使用及销售的任何产品或系统中。也可将本文件中记载的瑞萨电子产品或技术用于与军事应用或者军事用途有关的任何目的 (如大规模杀伤性武器的开发等)。在将本文件中记载的瑞萨电子产品或技术进行出口时, 应当遵守相应的出口管制法律法规, 并按照上述法律法规所规定的程序进行。
- 向第三方分销或处分产品或者以其他方式将产品置于第三方控制之下的瑞萨电子产品买方或分销商, 有责任事先向上述第三方通知本文件规定的内容和条件, 对于用户或第三方因非法使用瑞萨电子产品而遭受的任何损失, 瑞萨电子不承担任何责任。
- 在事先未得到瑞萨电子的书面认可的情况下, 不得以任何形式部分或全部转载或复制本文件。
- 如果对本文件所记载的信息或瑞萨电子产品有任何疑问, 或者用户有任何其他疑问, 请向瑞萨电子的营业部门咨询。  
(注1) 瑞萨电子: 在本文件中指瑞萨电子株式会社及其控股子公司。  
(注2) 瑞萨电子产品: 指瑞萨电子开发或生产的任何产品。



### SALES OFFICES

### Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "http://www.renesas.com/" for the latest and detailed information.

#### Renesas Electronics America Inc.

2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.  
Tel: +1-408-588-6000, Fax: +1-408-588-6130

#### Renesas Electronics Canada Limited

1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada  
Tel: +1-905-898-5441, Fax: +1-905-898-3220

#### Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K  
Tel: +44-1628-585-100, Fax: +44-1628-585-900

#### Renesas Electronics Europe GmbH

Arcadisstrasse 10, 40472 Düsseldorf, Germany  
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

#### Renesas Electronics (China) Co., Ltd.

Room 1709, Quantum Plaza, No.27 ZhichunLu Haidian District, Beijing 100191, P.R.China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

#### Renesas Electronics (Shanghai) Co., Ltd.

Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333  
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

#### Renesas Electronics Hong Kong Limited

Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: +852-2265-6688, Fax: +852-2886-9022/9044

#### Renesas Electronics Taiwan Co., Ltd.

13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan  
Tel: +886-2-8175-9600, Fax: +886-2-8175-9670

#### Renesas Electronics Singapore Pte. Ltd.

80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949  
Tel: +65-6213-0200, Fax: +65-6213-0300

#### Renesas Electronics Malaysia Sdn.Bhd.

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jin Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia  
Tel: +60-3-7855-9390, Fax: +60-3-7855-9510

#### Renesas Electronics Korea Co., Ltd.

12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5141