

Apalis i.MX6

Linux 开发入门



Revision History

Date	Doc. Rev.	Apalis i.MX6 Version	Changes
2015-8-6	Rev. 0.1	V1.1A	初始版本
2015-8-21	Rev. 0.2	V1.1A	文档模板更新
2016-3-10	Rev. 0.2.1	V1.1A	2.3 和 2.8 小节增加关于显示的配置说明
2016-5-18	Rev. 0.3	V1.1A	添加 Toradex 社区
2016-6-27	Rev. 0.4	V1.1A	公司地址信息变更
2017-2-9	Rev 0.5	V1.1A	在 Eclipse 中使用 SDK 中的 toolchain 添加 FTP 下载地址 IT 版本模块 BSP 更新 BSP migrate
2017-08-17	Rev 0.6	Apalis iMX6D 512MB V1.0A V1.1A V1.1B Apalis iMX6Q 1GB V1.0A V1.0B V1.1A V1.1B Apalis iMX6Q 1GB IT V1.0A Apalis iMX6Q 2GB IT V1.1A V1.1B V1.1C	SPL 更新
2018-08-08	Rev 0.6	Apalis iMX6D 512MB V1.0A V1.1A V1.1B Apalis iMX6Q 1GB V1.0A V1.0B V1.1A V1.1B Apalis iMX6Q 1GB IT V1.0A Apalis iMX6Q 2GB IT V1.1A V1.1B V1.1C	更新 Linux BSP v2.8b3, git checkout

目录

1. 开发设备以及材料准备	5
1.1 硬件准备	5
1.1.1 电脑主机	5
1.1.2 开发平台	5
1.1.3 基本配件	5
1.2 软件准备	5
2. 开发环境硬件配置	6
2.1 硬件连接框图	6
2.2 电源连接	6
2.2.1 Apalis 开发板	6
2.2.2 Ixora 载板	7
2.3 显示输出连接	7
2.3.1 Apalis 开发板	8
2.3.2 Apalis 开发板	8
2.4 调试串口连接	8
2.4.1 Apalis 开发板	9
2.4.2 Ixora 载板	11
2.4.3 串口波特率设置	11
2.5 USB 键盘/鼠标连接	12
2.6 网络连接	12
2.7 上电启动计算机模块	12
2.7.1 Apalis 开发板	12
2.7.2 Ixora 载板	13
2.8 更改显示输出接口	14
3. 开发环境软件配置	15
3.1 开发软件安装	15
3.2 串口软件 Minicom 配置	15
3.3 Apalis i.MX6 Linux SDK	17
3.4 Linux BSP 下载	18
4. Eclipse 应用程序开发	19

4.1	创建 C 工程项目	19
4.2	添加工程文件	19
4.3	工程属性配置	20
4.4	Eclipse's Remote System Explorer 配置	23
5.	Eclipse 调试应用程序	29
5.1	Eclipse 设置	29
6.	U-Boot 和 Linux 源代码下载	33
6.1	U-Boot	33
6.2	Linux	33
7.	BSP 更新	34
8.	其他开发参考资料	36
8.1	OpenEmbedded	36
8.2	开发者中心	36
8.3	Toradex 社区	36
9.	Toradex 公司简介及本地支持联系方式	37
9.1	Toradex 公司简介	37
9.2	本地支持联系方式	37

1. 开发设备以及材料准备

1.1 硬件准备

1.1.1 电脑主机

- ✓ 64bit Linux PC

1.1.2 开发平台

- ✓ Apalis i.MX6Q 或者 Apalis i.MX6D 计算机模块
- ✓ Apalis Evaluation Board V1.0/V1.1 或者 Ixora V1.0 底板

1.1.3 基本配件

开发所需的基本配件，按需自行配置，不包含在标准载板购买包装内。

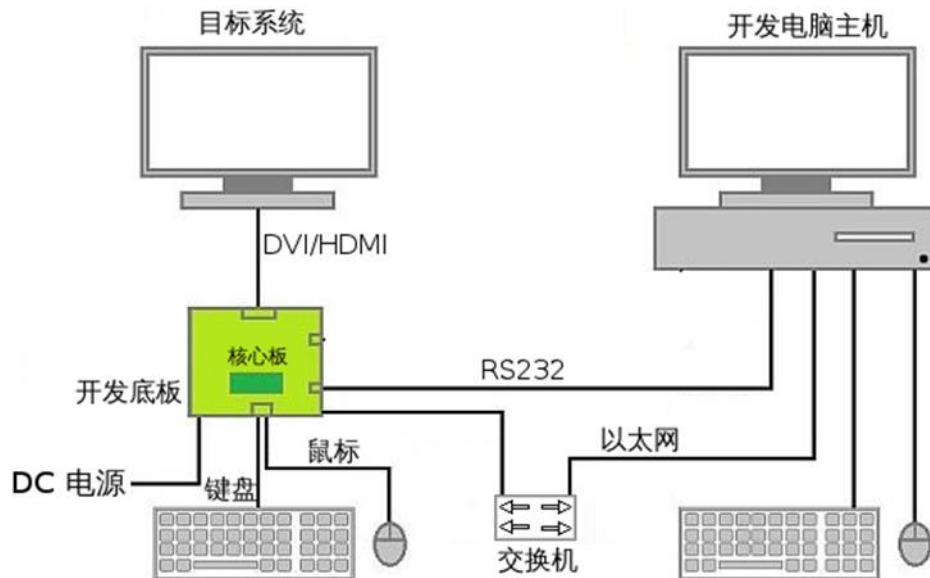
- ✓ 电源适配器/直流电源 (7 ~ 27V)
- ✓ DVI 或 HDMI 连接线
- ✓ DVI/HDMI 显示器
- ✓ 串口线
- ✓ USB 数据线

1.2 软件准备

- ✓ Ubuntu 14.04
- ✓ Eclipse CDT
- ✓ Minicom
- ✓ 交叉编译工具
- ✓ Apalis i.MX6 Linux BSP

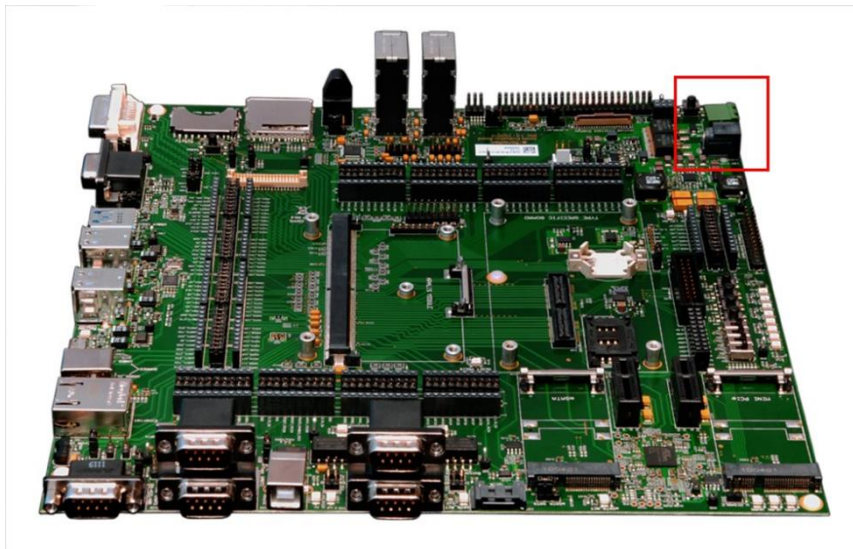
2. 开发环境硬件配置

2.1 硬件连接框图



2.2 电源连接

2.2.1 Apalis 开发板



- ✓ Apalis Evaluation Board 上的 X15 和 X17 接口，任意一路连接 DC 7~27V 电源即可，请注意 VCC 和 GND 的引脚位置。

2.2.2 Ixora 载板



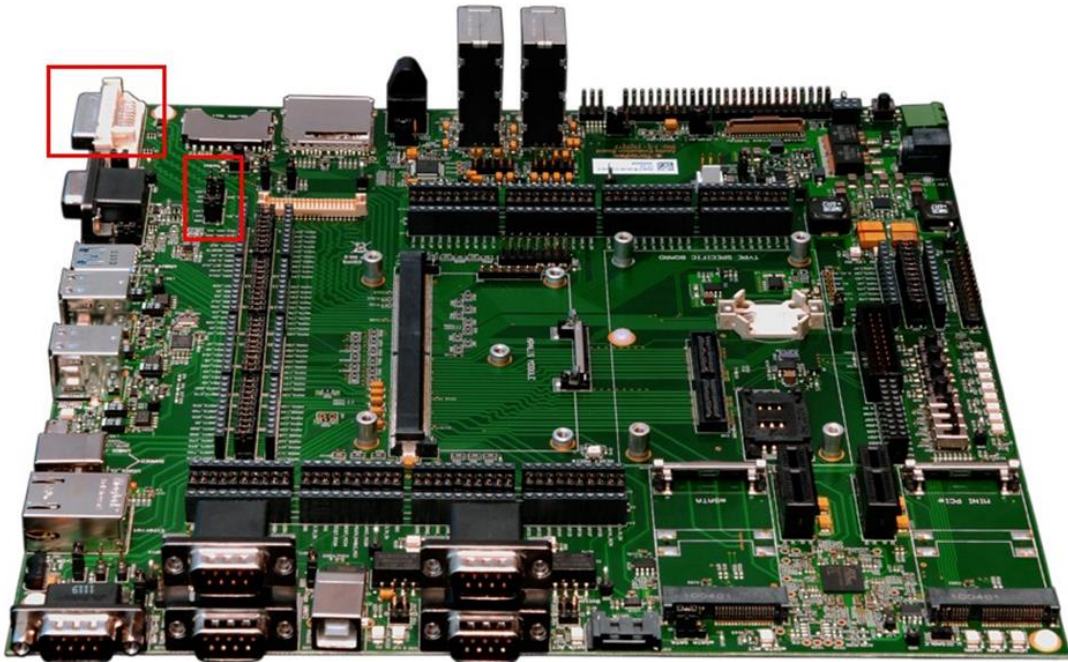
- ✓ Ixora 底板上的 X2 和 X3 接口，任意一路连接 DC 7~27V 电源即可，请注意 VCC 和 GND 的引脚位置

2.3 显示输出连接

韬睿提供的底板，大多数均支持 DVI/HDMI*、LVDS、并口 RGB、VGA 等常用显示器接口。Apalis i.MX6 系列模块的 Linux 默认采用 DVI/HDMI 输出。

* DVI 接口信号兼容 HDMI，可以使用 DVI 转 HDMI 接口，连接 HDMI 显示器。

2.3.1 Apalis 开发板



- ✓ Apalis Evaluation Board 上的 X11 通过 DVI 连接线连接显示器。确保 X39 和 X40 上选择 DVI 通道，这样可以使显示器使用 EDID，实现计算机模块自适应屏幕分辨率。

2.3.2 Apalis 开发板



- ✓ Ixora 底板上的 X17 接口，通过 HDMI 数据线连接显示。

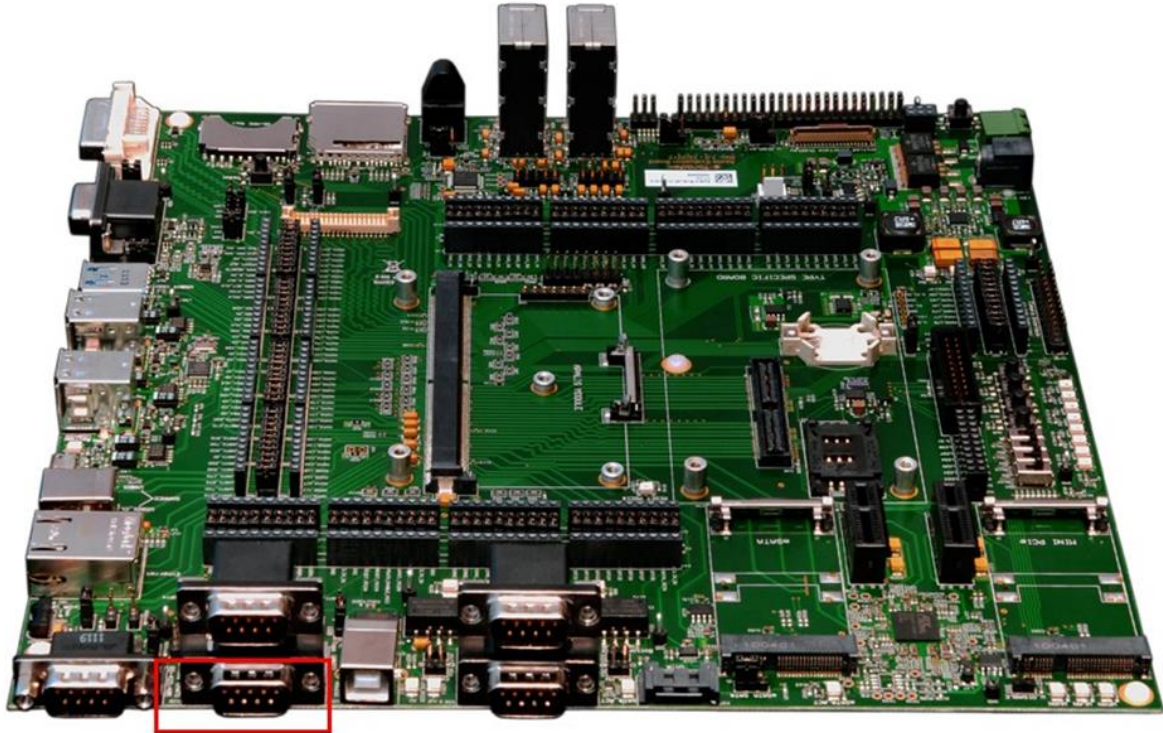
2.4 调试串口连接

Apalis 和 Colibri i.MX6 默认情况下将 UART_A 作为调试串口，该串口不仅可以用于应用开发时候，调试应用程序，还可以获取以及配置 Linux 和 U-Boot 系统。如果串口数量能够满足应用的前提下，一般建议保留调试串口的功能。

为了方便用户使用调试串口，Apalis Evaluation Board 提供两种方式连接串口。第一种，直接使用 RS232 串口连接。这个方法适用于台式机用户，一般台式机电脑均提供 RS232 接口。第二种，USB 串口连接。Apalis Evaluation Board 底板上的 FTDI FT232RL 芯片，将 UART 串口转为 USB 设备。这种适用于没有 RS232 接口的笔记本用户，当然，也适用于台式机。下面将介绍详细的连接方法。

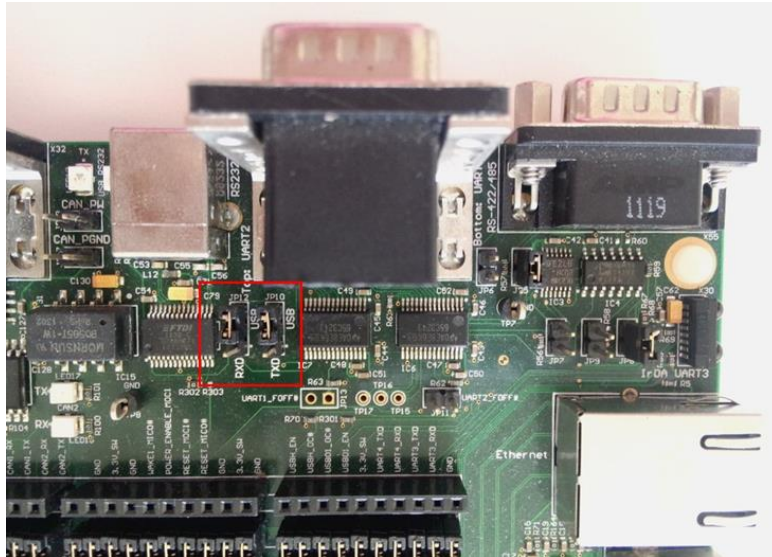
2.4.1 Apalis 开发板

RS232 串口接口

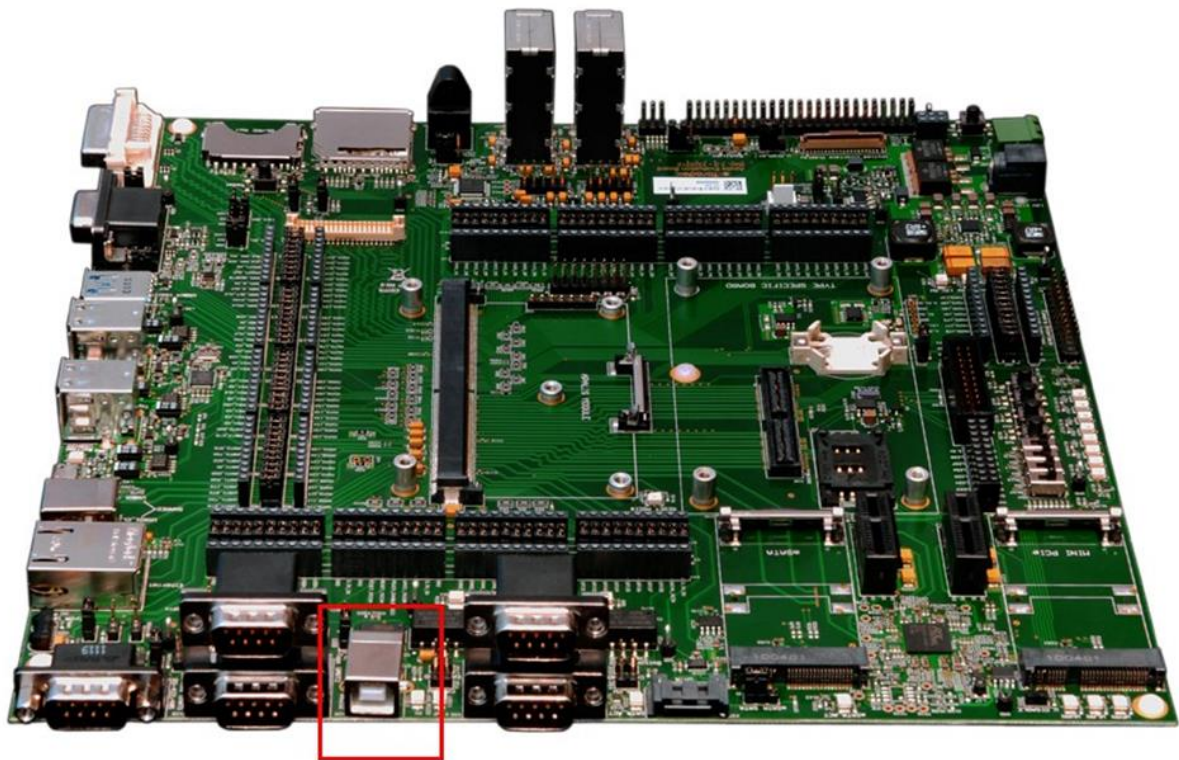


- ✓ 在 Apalis Evaluation Board 底板上 X28 的下面的串口为默认的调试串口，同时将 JP12 和 JP10 上的短路帽分别移至 RXD 和 TXD 一侧。

USB 转串口接口



- ✓ 在 Apalis Evaluation Board 底板上 X29 为 USB 调试串口，同时将 JP12 和 JP10 上的短路帽分别移至 USB 一侧。



- ✓ 使用 USB 数据线，将载板上的 X29 连接至电脑主机。

Ubuntu 中已经集成 FTDI FT232RL 驱动，使用 lsusb 命令即可发现该设备

```
ban@LinuxDev:~$ lsusb
Bus 003 Device 005: ID 0403:6001 Future Technology Devices International, Ltd FT232 USB-Serial (UART) IC
```

Windows 系统中需要先安装相应的驱动，才能访问串口。 FTDI FT232RL 可以从 FTDI 网站下载 <http://www.ftdichip.com/FTDrivers.htm>

2.4.2 Ixora 载板



- ✓ 在 Ixora 上 X22 为默认的调试串口。X22 为 IDC-10 接口，具体引脚定义请参考 [Ixora 技术手册](#)。X22 为 RS232 串口，用户可以制作对应的转接线连接至 DB9 接口，转接线线序如下：

DB-9	IDC-10
Pin 1	Pin 1
Pin 2	Pin 3
Pin 3	Pin 5
Pin 4	Pin 7
Pin 5	Pin 9
Pin 6	Pin 2
Pin 7	Pin 4
Pin 8	Pin 6
Pin 9	Pin 8

2.4.3 串口波特率设置

默认串口波特率为 115200,8n1。

2.5 USB 键盘/鼠标连接

韬睿提供的 i.MX6 Linux BSP 中已经包含图形桌面系统 LXDM，用户只要连接 USB 鼠标和键盘，就可以方便地使用开发板上的 Linux 系统。底板均有多个 USB 接口，能够连接三个以上的 USB 设备，如需更多的接口，可以使用 USB Hub 进行 USB 接口扩展。i.MX6 上的 Linux 无需安装驱动，就能够支持 USB Hub。

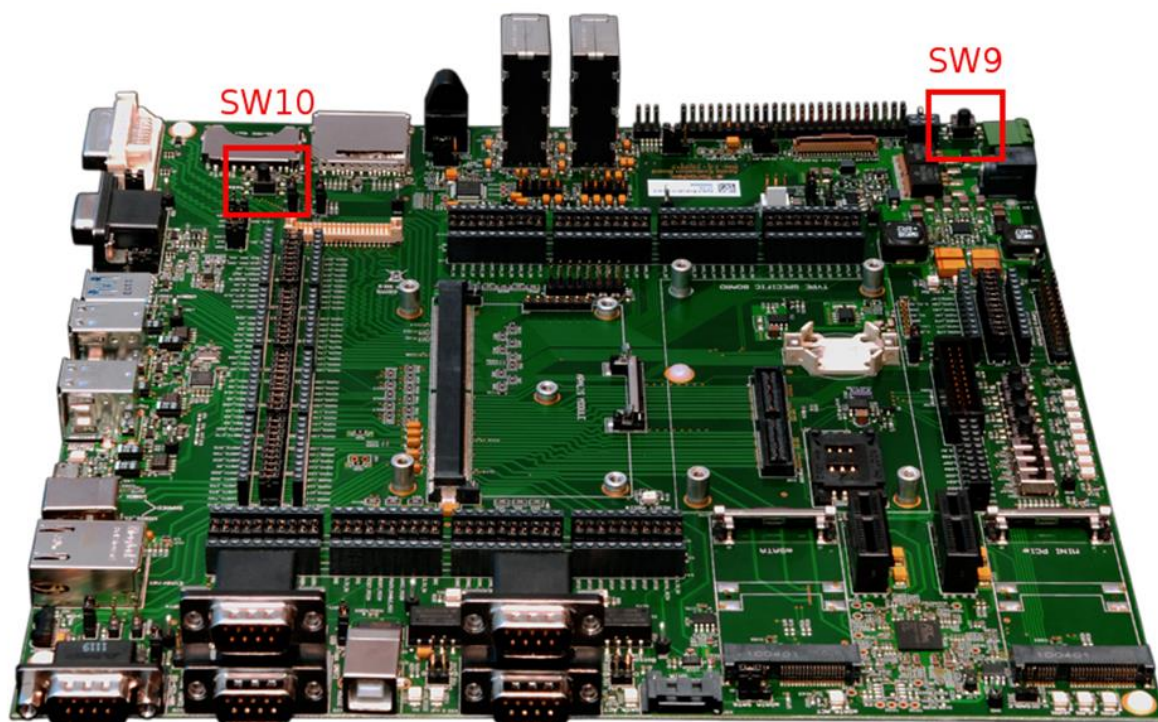
2.6 网络连接

在连接以太网的情况下，用户可以十分方便地以 SSH 连接远程登录到 i.MX6 上的 Linux 系统。同时用户也能用功 GDB 实现在线应用程序调试。i.MX6 上的 Linux 默认采用 DHCP 方式，自动获取 IP 地址。用户可以通过调试串口或者图形桌面系统查看 i.MX6 获取到的 IP 地址。

2.7 上电启动计算机模块

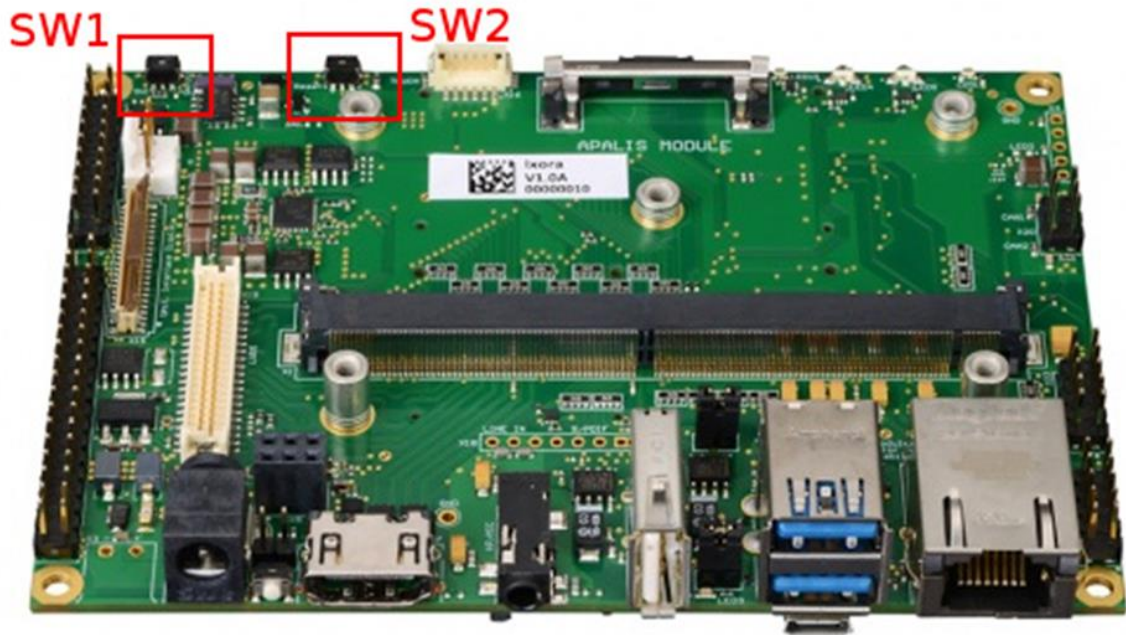
韬睿 Apalis 系列底板均有独立的上电和复位开关，因此，需要触发开关，核心模块才能启动。

2.7.1 Apalis 开发板



- ✓ SW9: 上电开关
- ✓ SW10: 复位

2.7.2 Ixora 载板



- ✓ SW1: 上电开关
- ✓ SW2: 复位

当 Apalis i.MX6 模块正确安装到底板上，并连接相应的电源和显示连线，上电启动后，就能够在显示器上看到 Linux 桌面系统。



2.8 更改显示输出接口

Apalis i.MX6 模块，除了支持 DVI/HDMI 外，还可以从 LCD、LVDS 和 VGA 输出显示，甚至是支持多屏同时显示。更改显示输出，需要在 UBoot 中启用相应的配置。例如使用 LVDS 作为输出。

```
-----  
| setenv vidargs_lvds 'video=mxcfb0:dev=ldb,1024x768M@60,if=RGB24 ldb=sin0 video=mxcfb1:off |  
| video=mxcfb2:off video=mxcfb3:off fbmem=32M' |  
|-----
```

更多关于显示的配置，请参考

http://developer.toradex.com/knowledge-base/framebuffer-%28linux%29#iMX_6_Framebuffer_Boot_Configuration

3. 开发环境软件配置

3.1 开发软件安装

✓ Ubuntu 14.04 软件安装:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install build-essential meld git gawk eclipse-cdt eclipse-cdt-launch-remote eclipse-rse minicom
sudo apt-get install gtk+-2.0-dev
```

如果使用的是 64 位系统, 请安装以下软件包

```
sudo apt-get install lib32z1 lib32ncurses5 lib32bz2-1.0
```

3.2 串口软件 Minicom 配置

✓ Apalis i.MX6 模块串口默认使用 115200, 8N1, no hardware flow control 格式协议。按照下面设置打开 minicom。

```
sudo minicom -s
```

```
+-----[configuration]-----+
```

```
| Filenames and paths   |
| File transfer protocols |
| Serial port setup     |
| Modem and dialing     |
| Screen and keyboard   |
| Save setup as dfl     |
| Save setup as..       |
| Exit                  |
| Exit from Minicom     |
+-----+
```

```
+-----+
| A - Serial Device   : /dev/ttyUSB0          |
| B - Lockfile Location : /var/lock           |
| C - Callin Program  :                      |
```

```

| D - Callout Program      :                |
| E - Bps/Par/Bits       : 115200 8N1    |
| F - Hardware Flow Control : No        |
| G - Software Flow Control : No        |
|                          |              |
| Change which setting?   |              |
+-----+
+-----[Modem and dialing parameter setup]-----+
|                          |              |
| A - Init string .....  |              |
| B - Reset string .....  |              |
| C - Dialing prefix #1.... ATDT                |
| D - Dialing suffix #1.... ^M                  |
| E - Dialing prefix #2.... ATDP                |
| F - Dialing suffix #2.... ^M                  |
| G - Dialing prefix #3.... ATX1DT              |
| H - Dialing suffix #3.... ;X4D^M              |
| I - Connect string ..... CONNECT              |
| J - No connect strings .. NO CARRIER   BUSY   |
|          NO DIALTONE   VOICE              |
| K - Hang-up string .....  |              |
| L - Dial cancel string .. ^M                |
|                          |              |
| M - Dial time ..... 45    Q - Auto bps detect ..... No    |
| N - Delay before redial . 2    R - Modem has DCD line .. No    |
| O - Number of tries ..... 10    S - Status line shows ... DTE speed |
| P - DTR drop time (0=no). 1    T - Multi-line untag .... No    |
|                          |              |
| Change which setting?   (Return or Esc to exit)   |
+-----+

```

上电启动后，可以在 minicom 中看到 Apalis i.MX6 模块输出的启动信。登录系统的默认用户名为 root，密码为空。


```
U-Boot 2015.04 (Jun 10 2015 - 09:35:43)
CPU: Freescale i.MX6SOLO rev1.1 at 792 MHz
Reset cause: POR
I2C: ready
DRAM: 256 MiB
MMC: FSL_SDHC: 0, FSL_SDHC: 1
auto-detected panel vga-rgb
Display: vga-rgb (640x480)
In: serial
Out: serial
Err: serial
Model: Toradex Colibri iMX6 Solo 256MB IT V1.0A, Serial# 04823113
Net: using phy at 0
FEC [PRIME]
Hit any key to stop autoboot: 0
.....

..-O-..
| |      .-  oo
| | |-----|| |-----
| | |__| --| '-| -'| | |
| | | | | | -- || --| | | | |
'-----' |-----"-----'
      -' |
      '---'

The Angstrom Distribution colibri-imx6 ttymxc0
Angstrom v2014.12 - Kernel
Colibri_iMX6_LinuxImageV2.4_20150731
colibri-imx6 login: root
```

3.3 Apalis i.MX6 Linux SDK

从 FTP 服务器以下目录中下载 SDK

CustomerFTP/Linux/OpenEmbedded/V2.6/SDK/Apalis_iMX6

ftp 地址: <ftp://ftp.toradex.cn>

将交 SDK 安装到用户指定的目录中 /home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/apalis_imx6

```
sudo chmod a+x ./angstrom-glibc-x86_64-armv7at2hf-vfp-neon-v2015.12-toolchain.sh  
./angstrom-glibc-x86_64-armv7at2hf-vfp-neon-v2015.12-toolchain.sh
```

3.4 Linux BSP 下载

你可以从下面链接下载韜睿最新发布的 BSP 文件，以及相应的发行说明。建议用户关注韜睿主页，我们会及时发布 BSP 更新信息，新版本 BSP 一般会解决之前发布版本中存在的一些问题，或者提供某些功能的性能改善。

<http://developer.toradex.com/files/toradex-dev/uploads/media/Colibri/Linux/Images/>

例如，下载 Apalis_iMX6_LinuxImageV2.6Beta2_20160630.tar.bz2 ， 需要以 root 权限解压。

```
sudo tar vxf Apalis_iMX6_LinuxImageV2.6Beta2_20160630.tar.bz2
```

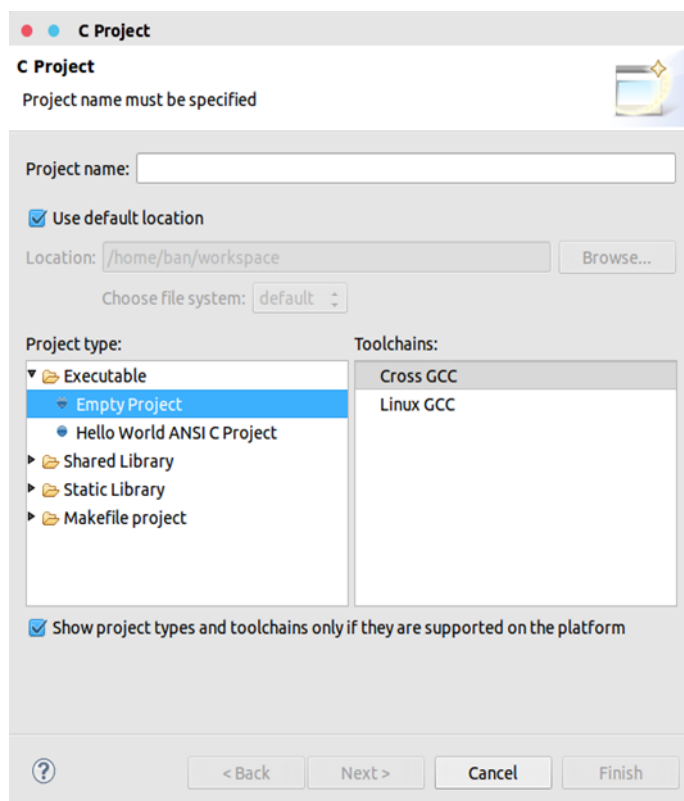
4. Eclipse 应用程序开发

在 Linux 系统中进行应用程序开发，可以有多种 IDE 供用户选择，其中 Eclipse 是较为常用的一种。Eclipse 能够自动生成 Makefiles 文件，这对于初学者来说，是一个极大的便利，可以免去学习 Makefiles 文件编写规则，方便用户快速上手 Linux 应用程序开发。下面将介绍如何在 Eclipse 中进行嵌入式应用程序开发。需要在终端先加载 SDK 的环境变量后，在打开 Eclipse，方法如下：

```
ben@LinuxDev: /home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/apalis_imx6$ . environment-setup-  
armv7at2hf-vfp-neon-angstrom-linux-gnueabi  
ben@LinuxDev: /home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/ apalis_imx6$ eclipse
```

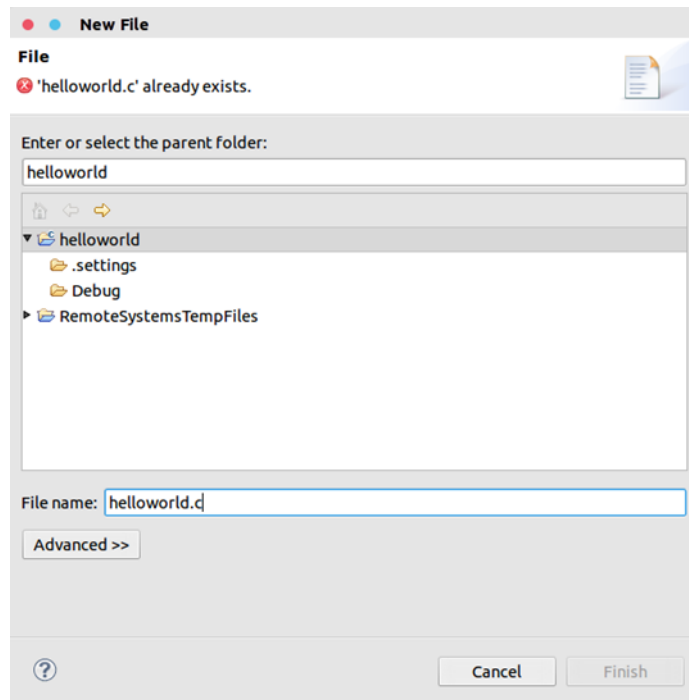
4.1 创建 C 工程项目

- ✓ 菜单选项 --> File --> New --> C Project
- ✓ Project type: Empty Project
- ✓ Toolchains: Cross GCC



4.2 添加工程文件

- ✓ 在 Project Explorer 中的工程文件夹上右击，New --> File
- ✓ file name : hello_word.c



在 helloworld.c 文件中输入 C 程序



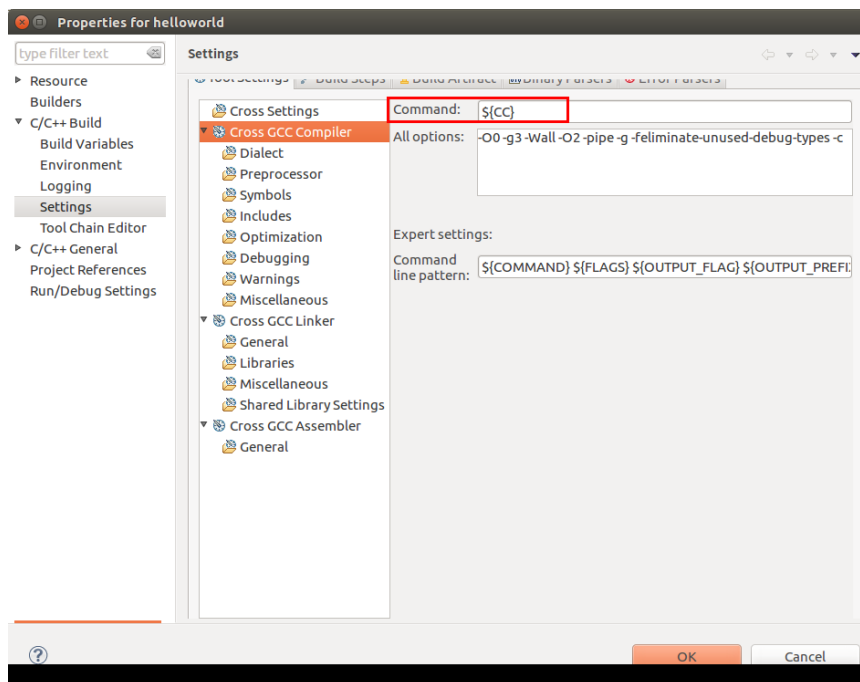
4.3 工程属性配置

默认的 Eclipse 中的编译器是针对 x86 平台，所以编译出来的程序无法运行在 ARM 嵌入式平台。因此，需要重新配置项目的属性。这里我们将使用 SDK 中的交叉编译工具。

- ✓ 在 Project Explorer 中的工程文件夹上右击，选择 Properties.
- ✓ 依次选择，C/C++ Build --> Settings

Cross GCC Compiler

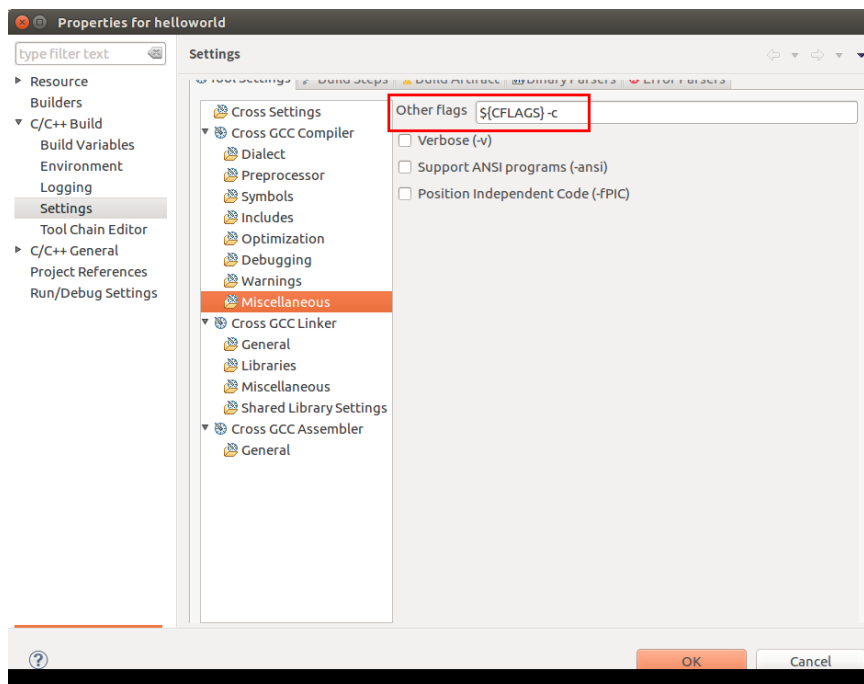
Command : \${CC}



该路径以实际电脑上安装为准，需要做相应的调整，请参考“交叉编译工具安装”章节。

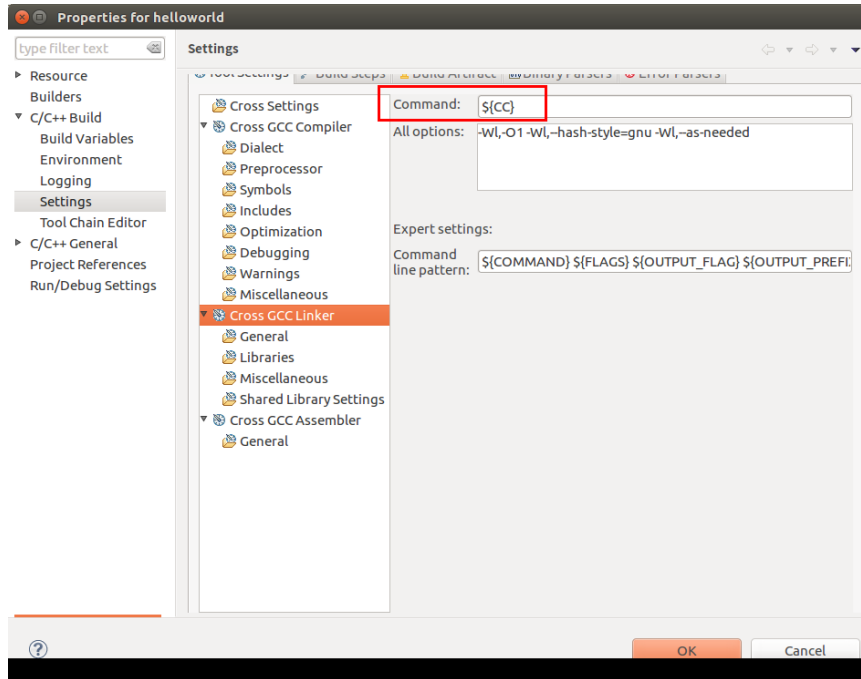
Miscellaneous

Other flags: `${CFLAGS} -c`



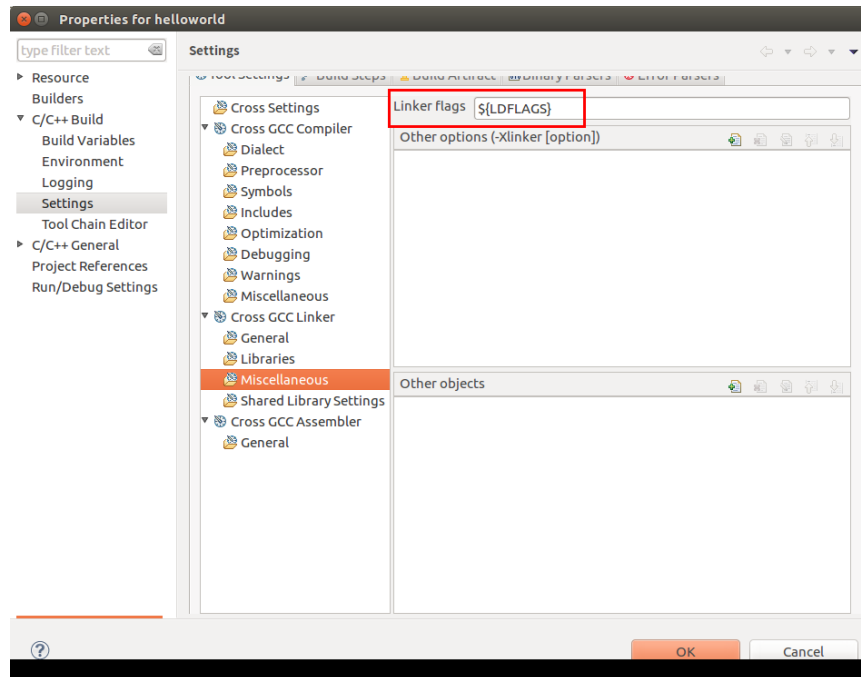
Cross GCC Linker

Command: `${CC}`



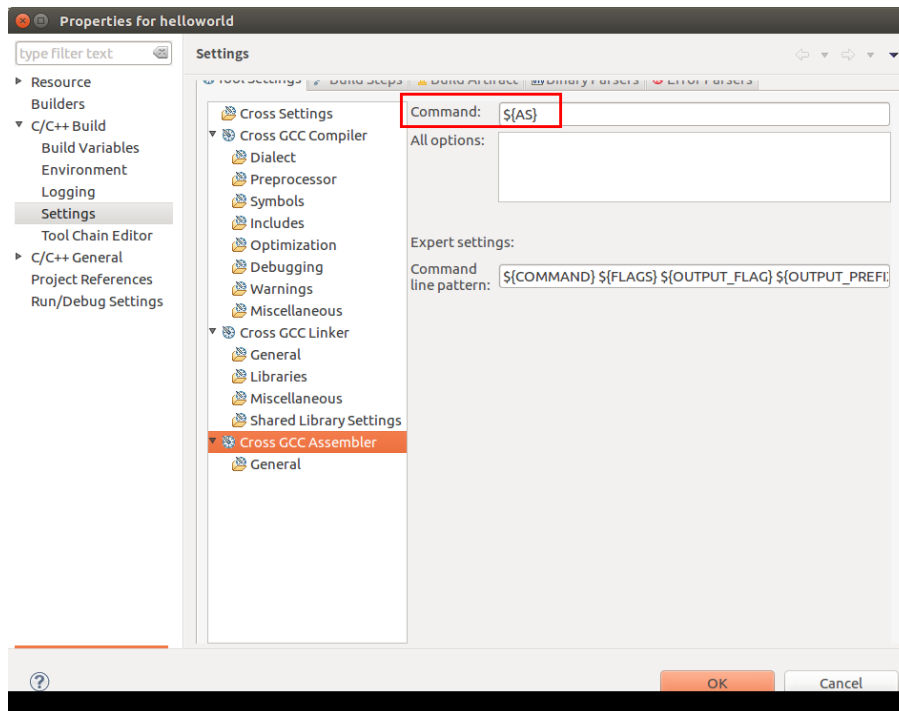
Miscellaneous

Linker flags: $\${LDLFLAGS}$



Cross GCC Assembler

Command : $\${AS}$

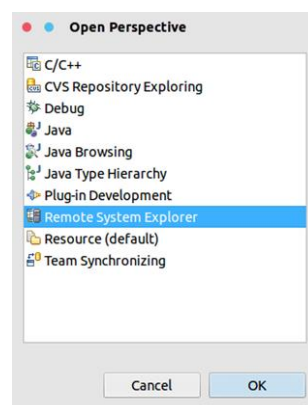


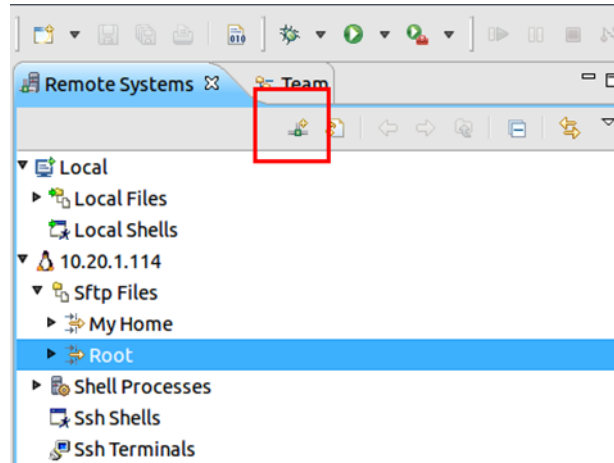
配置完毕后，就可以点击 Project --> Build All 编译项目。

4.4 Eclipse's Remote System Explorer 配置

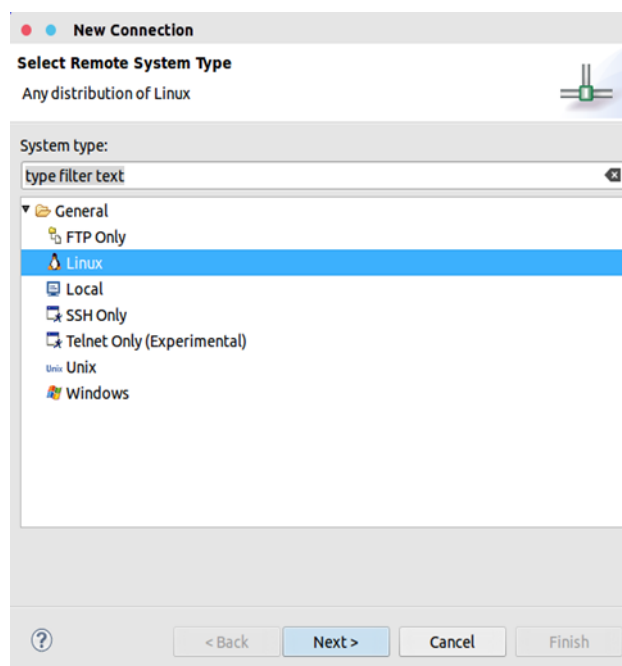
使用 Remote System Explorer 后，可以直接在 Eclipse 中访问目标板上的文件系统，方便将文件从电脑上复制到目标板，部署应用程序至开发板。

点击 菜单 > Windows --> Open Perspective --> Other..., 选择 Remote System Exploere





- ✓ 在 Remote System --> Define a connection to remote system 点击



- ✓ Host name: 目标板的 IP 地址
- ✓ 勾选 Verify host name

New Connection

Remote Linux System Connection

⊗ Please enter a valid connection name.

Parent profile: LinuxDev

Host name: 10.20.1.114

Connection name:

Description: Colibri i.MX6

Verify host name

[Configure proxy settings](#)

? < Back Next > Cancel Finish

New Connection

Files

Define subsystem information

Configuration	Properties				
<input type="checkbox"/> dstore.files <input type="checkbox"/> ftp.files <input checked="" type="checkbox"/> ssh.files	<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> </tbody> </table>	Property	Value		
Property	Value				

Available Services

- Ssh / Sftp File Service
- SSH Connector Service
 - SSH Settings

Description

Work with files on remote systems using the Secure Shell (ssh) protocol.

? < Back Next > Cancel Finish

New Connection

Processes
Define subsystem information

Configuration	Properties				
<input type="checkbox"/> dstore.processes <input checked="" type="checkbox"/> processes.shell.linux	<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> </tbody> </table>	Property	Value		
Property	Value				

Available Services

- Shell Process Service

Description

This configuration allows you to work with processes on remote linux systems using any contributed Shell subsystem.

New Connection

Shells
Define subsystem information

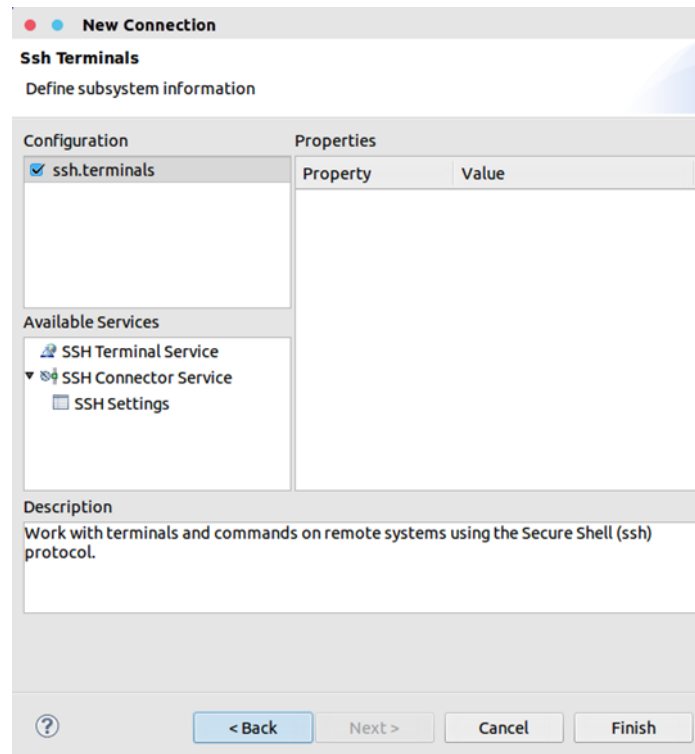
Configuration	Properties				
<input type="checkbox"/> dstore.shells <input checked="" type="checkbox"/> ssh.shells	<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> </tbody> </table>	Property	Value		
Property	Value				

Available Services

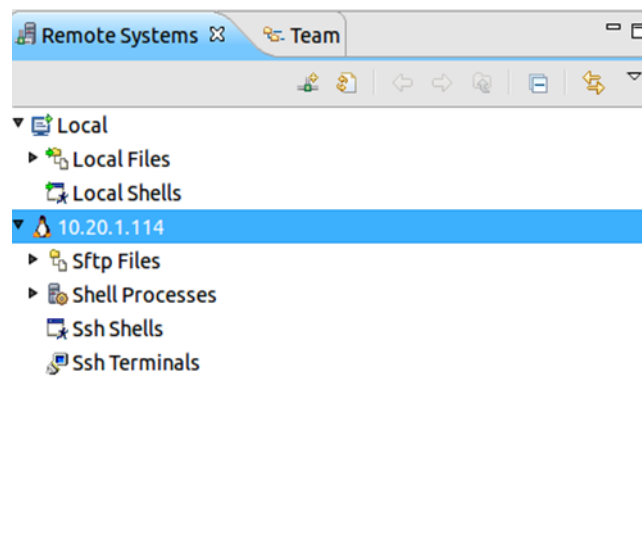
- Generic shell service
- SSH Connector Service
 - SSH Settings

Description

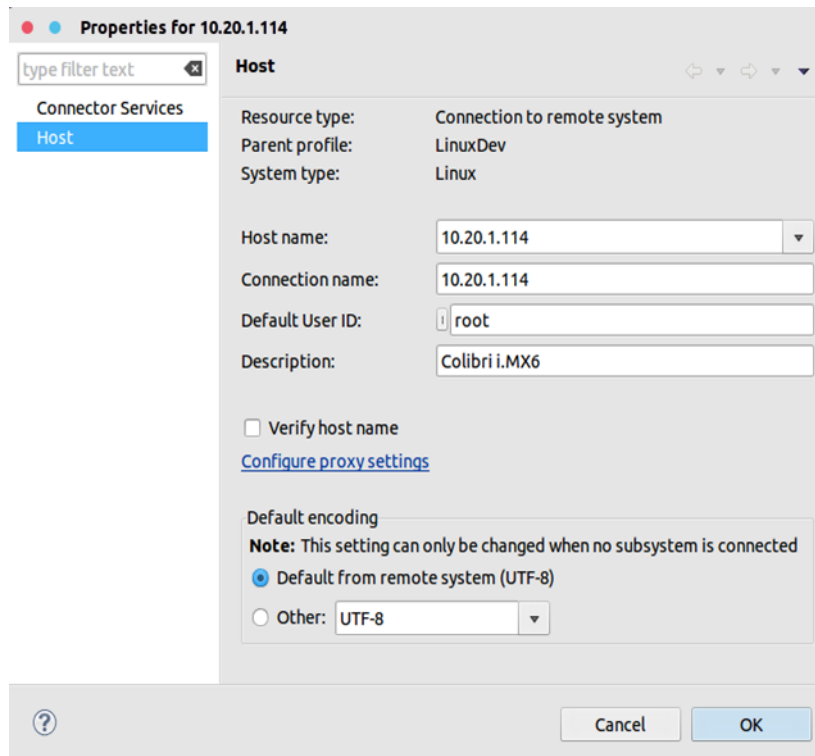
Work with shells and commands on remote systems using the Secure Shell (ssh) protocol.



- ✓ 配置完成后，在 Remote Systems 中可以看到目标板的信息



默认配置下，Eclipse 会使用电脑的用户名来登录目标板系统。这里需要更改登录默认用户名以及密码。



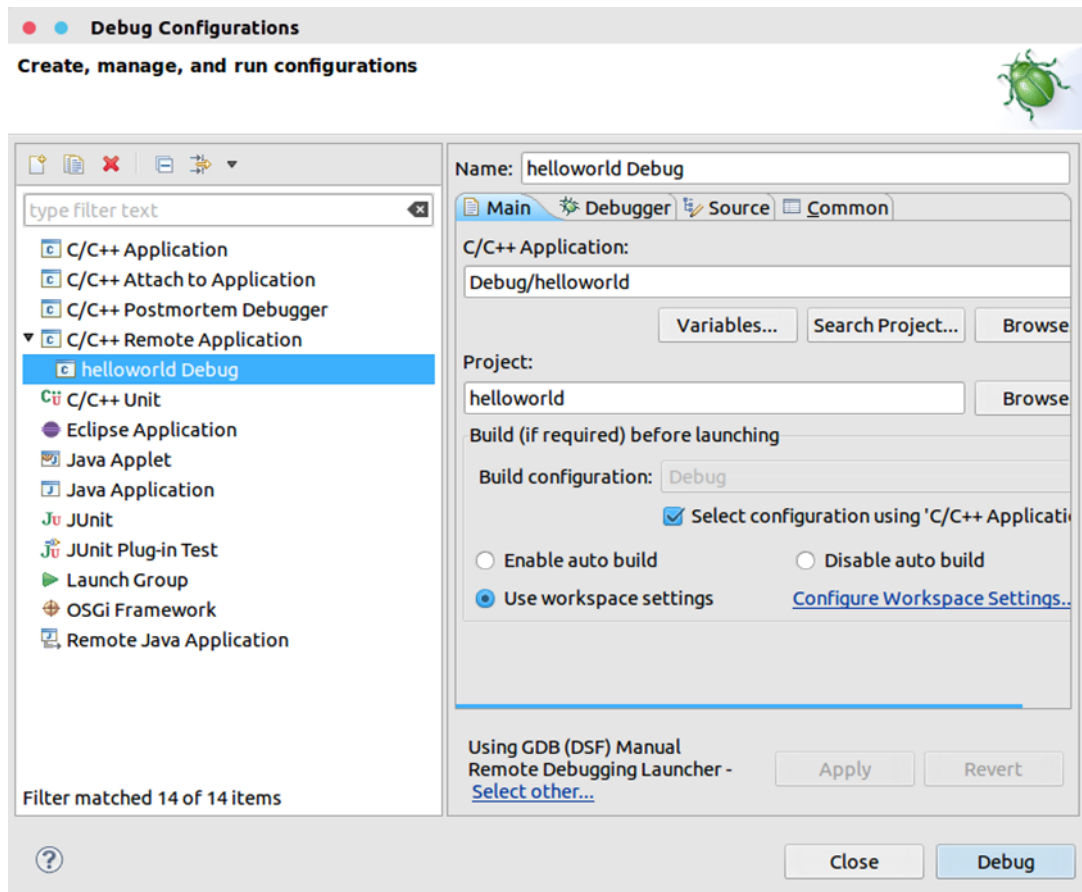
- ✓ Remote Systems 中右击目标板的 IP 地址，点击 Connect 即可连接。
- ✓ 在 Eclipse 中将编译好的程序复制到 Remote System --> Sftp files --> My home。使用 SSH 终端连接目标板，运行下面命令：

```
l chmod +x hello_world
```

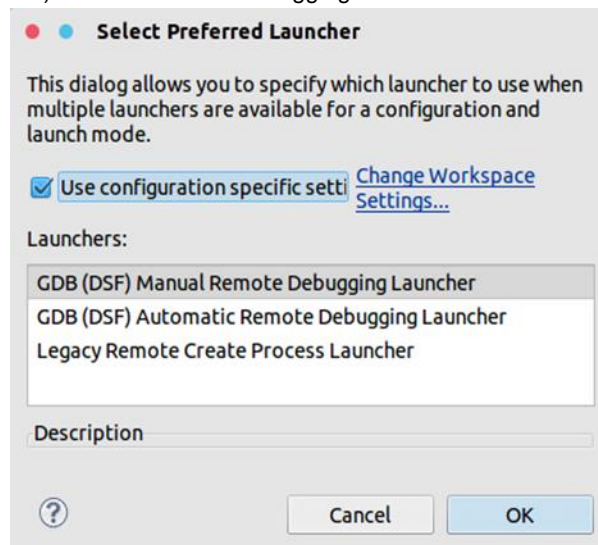
5. Eclipse 调试应用程序

5.1 Eclipse 设置

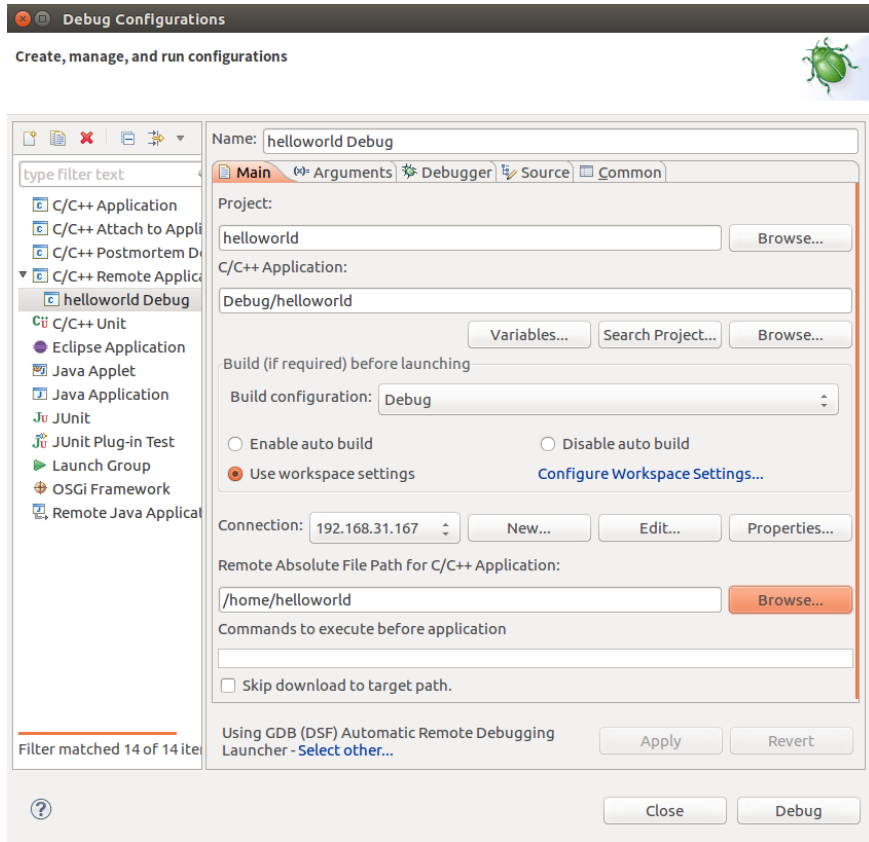
- ✓ 在 Eclipse 中, 点击 菜单 --> Run --> Debug Configurations...
- ✓ 创建一个 C/C++ Remote Application



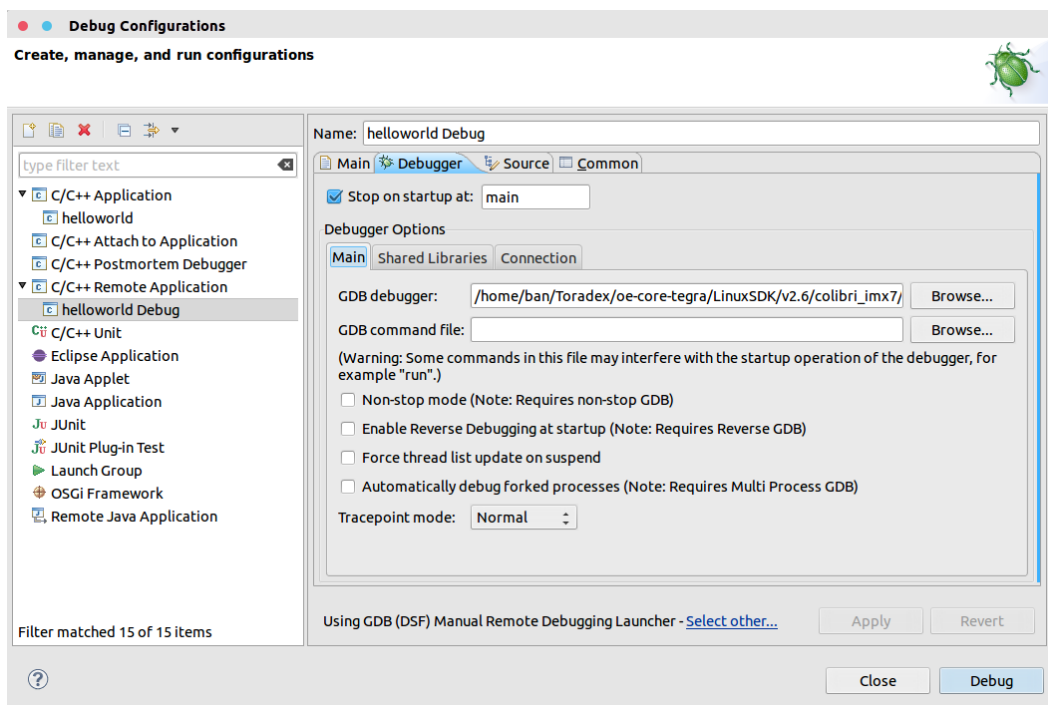
- ✓ 点击底部 Using GDB(DSF) Manual Remote Debugging Launcher – Select other...



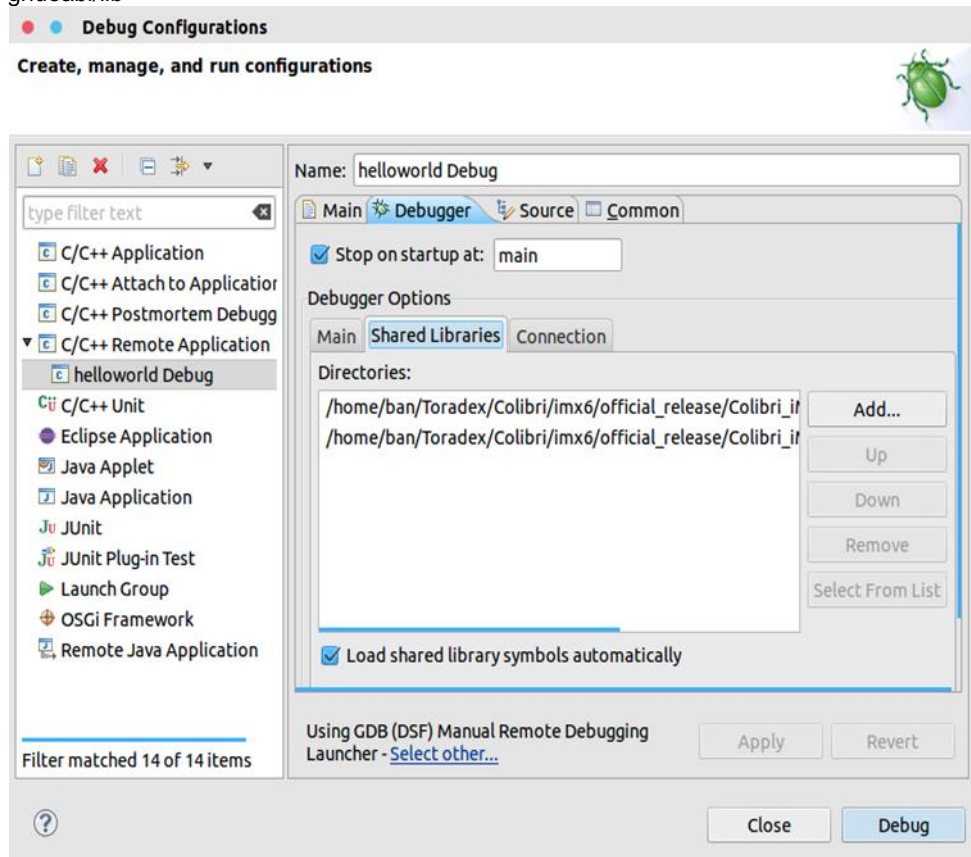
- ✓ 点击 Browser, 选择开发板上应用目录, 一般选择 /home/root 目录



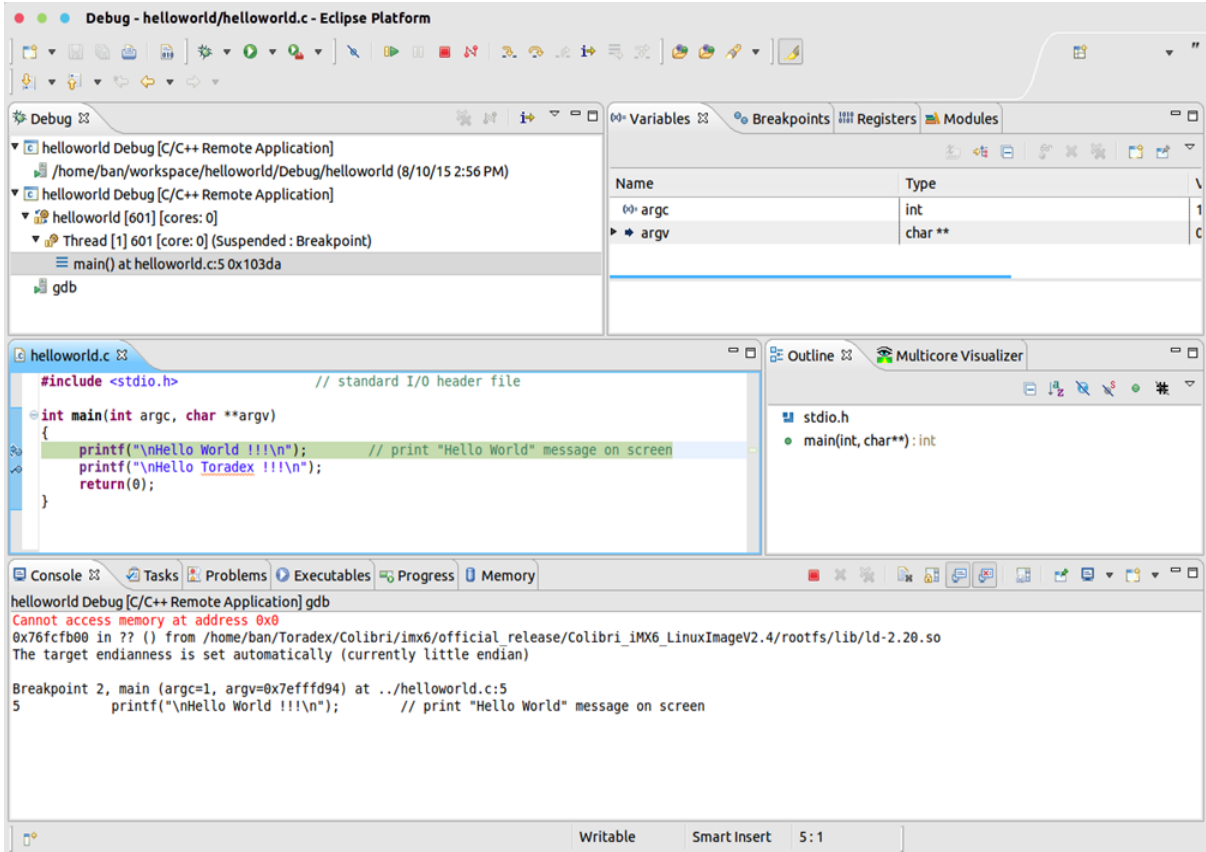
- ✓ 选择 GDB debugger
/home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/apalis_imx6/sysroots/x86_64-angstromsdk-linux/usr/bin/arm-angstrom-linux-gnueabi/arm-angstrom-linux-gnueabi-gdb



- ✓ 在 Debugger 标签中也，添加共享库文件目录，库文件位于 SDK 文件夹
/home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/apalis_imx6/sysroots/armv7at2hf-vfp-neon-angstrom-linux-gnueabi/usr/lib
/home/ban/Toradex/oe-core-tegra/LinuxSDK/v2.6/apalis_imx6/sysroots/armv7at2hf-vfp-neon-angstrom-linux-gnueabi/lib



- ✓ Connection 标签中设置，目标板的 IP 地址，以及 gdbserver 侦听端口。最后点击 Debug，进入调试界面。调试结果输出在运行 gdbserver 的 SSH 窗口中。



The screenshot displays the Eclipse IDE interface during a GDB debug session. The main editor shows the source code of `helloworld.c` with the following content:

```
#include <stdio.h> // standard I/O header file

int main(int argc, char **argv)
{
    printf("\nHello World !!!\n"); // print "Hello World" message on screen
    printf("\nHello Toradex !!!\n");
    return(0);
}
```

The console window shows the following output:

```
helloworld Debug [C/C++ Remote Application] gdb
Cannot access memory at address 0x0
0x76fcfb00 in ?? () from /home/ban/Toradex/Colibri/imx6/official release/Colibri_iMX6_LinuxImageV2.4/rootfs/lib/ld-2.20.so
The target endianness is set automatically (currently little endian)

Breakpoint 2, main (argc=1, argv=0x7efffd94) at ../helloworld.c:5
5     printf("\nHello World !!!\n"); // print "Hello World" message on screen
```

The Variables window shows the current state of the program's variables:

Name	Type
argc	int
argv	char **

6. U-Boot 和 Linux 源代码下载

韜睿将 Apalis i.MX6 的 U-Boot 和 Linux 源代码均公开在 韜睿的 git 服务器上，并持续维护更新。用户可以直接从服务器上下载源码，建议用户使用最新版本的 U-Boot 和 Linux，并下载最新的分支。

6.1 U-Boot

✓ `git clone -b 2016.11-toradex git://git.toradex.cn/u-boot-toradex.git`

6.2 Linux

✓ `git clone -b toradex_4.9-1.0.x-imx git://git.toradex.cn/linux-toradex.git`

U-Boot 和 Linux 具体的配置和编译方法，请参考 [Build U-Boot and Linux Kernel from Source Code](#)

<http://developer.toradex.cn/software-resources/arm-family/linux/board-support-package/build-u-boot-and-linux-kernel-from-source-code>

在下载 Linux 源码时，请根据对应的模块选择需要的分支版本

https://developer.toradex.cn/knowledge-base/build-u-boot-and-linux-kernel-from-source-code#Image_Versions

当前系统安装的 BSP 版本通过下面命令查看

```
~# cat /etc/issue
```

如果你需要使用特定的版本，除了制定分支外，还需要 checkout 至对应的 tag

<http://git.toradex.cn/cgit/linux-toradex.git/refs/tags>

注意：

1) 自 Linux BSP v2.8b2 开始，iMX6 采用 zImage 格式，替代之前的 ulmage，在编译内的时候请使用 make zImage 命令。

7. BSP 更新

韬睿会不定期地发布针对 Apalis i.MX6 模块的 BSP，这些 BSP 会解决历史版本中遗留的或者用户报告的问题，以及为了提高相应模块的性能而增加的补丁等。关于 BSP 的说明请参考以下网页 Linux BSP Release Details：

<http://developer.toradex.com/software-resources/arm-family/linux/linux/release-details?view=all>

- ✓ 下载 Apalis_iMX6_LinuxImageVx.yz.tar.bz2，以 root 权限解压
- ✓ 准备 1GB 以上 VFAT 格式的 SD 卡
- ✓ 执行脚本 `sudo update.sh -o /PATH_TO_SD_CARD`
- ✓ EvalBoard V1.0A/V1.1A:

将 SD 卡插入载板上 SD/MMC 8bit (X18) 或者 SD/MMC 4bit (X19)

- ✓ Ixora V1.0A:

将 SD 卡插入载板上 SD/MMC 8bit (X18) 或者 SD/MMC 4bit (X19)

如果您从 V2.2 BSP 版本升级或者使用 Ixora 载板，需要在 U-Boot 中配置以下命令：

```
Apalis iMX6 # setenv drive 2
Apalis iMX6 # setenv setupdate 'fatload mmc ${drive}:1 ${loadaddr} flash_mmc.img; source ${loadaddr}'
Apalis iMX6 # save
```

复位之后，重新进入 U-Boot 命令模式，执行下面命令，完成 BSP 更新

```
Apalis iMX6 # run setupdate
Apalis iMX6 # run update
```

如果您从 V2.4 版本升级到 V2.6 或者更新的 BSP 时会出现下面的提示

```
MMC write: dev # 0, block # 2, count 614 ... 614 blocks written: OK
done.
successfully updated U-Boot, power-cycle and enter "run setupdate; run migrate" to complete update
Apalis iMX6 #
```

然后需要重启上电启动开发板

```
Apalis iMX6 # run setupdate; run migrate
```

V2.6 Beta 1 以及后续版本的 BSP 会启用 eMMC Fast Boot，如果选择了 Yes，这个过程是不可逆的，模块将一直采用 eMMC Fast Boot 模式启动。

针对 IT 版本的 Apalis iMX6 模块，如果使用 V2.7 版本之前的 BSP，执行 “run update_it” 进行更新。如果使用 V2.7 以及后续版本，可以使用 “run update” 进行更新。

V2.7 以及后续版本采用了 SPL bootloader，通过下面命令更新

```
Apalis iMX6 # run setupdate  
Apalis iMX6 # run update_spl
```

详细的说明，请参考 [Flashing Embedded Linux to iMX6 Modules](#)

<http://developer.toradex.com/knowledge-base/flashing-linux-on-imx6-modules>

<http://developer.toradex.com/knowledge-base/linux-booting>

8. 其他开发参考资料

8.1 OpenEmbedded

韜睿的 BSP 通过 OpenEmbedded/Yocot 进行发布，用户可以下载、编译完整的 BSP，以及自行裁剪 BSP，或者添加第三方软件等，具体请参考 OpenEmbedded (core)

<http://developer.toradex.com/software-resources/arm-family/linux/board-support-package/openembedded-%28core%29>

8.2 开发者中心

韜睿的开发者中心提供了丰富的学习资料，内容涵盖驱动移植、应用开发、多媒体显示、外设操作、内核编译等嵌入式开发常见知识。用户可以从中学习简单的开发，到具体应用的深入研究。上面的文章均有韜睿位于瑞士总部的研发工程师编写和维护，直接保证文章的时效性和可用性。

<http://developer.toradex.com>

更多关于 Linux 的开发，请点击下面网页：

<http://developer.toradex.com/knowledge-base/getting-started-with-toradex-arm-modules#Linux>

8.3 Toradex 社区

在 Toradex 社区里，任何人都可以询问关于我们产品的问题，我们的研发人员将会解答您的问题。我们创建这个社区，是因为我们发现很多用户都会遇到同样或者类似的问题，公开这些解答，使得用户能够更快地找到答案。

<https://www.toradex.com/community>

9. Toradex 公司简介及本地支持联系方式

9.1 Toradex 公司简介

Toradex 是一个领先的 ARM®计算机模块 (CoM) / 系统模块 (SoM) 厂商, 产品被众多嵌入式应用所采用。使用先进的 ARM 处理器, 例如 Freescale® i.MX 6 & Vybrid™, NVIDIA® Tegra™, 和 Marvell® PXA, 韜睿的计算机模块系列在价格、性能、功耗和接口方面为用户提供丰富的选择。使其成为满足各种嵌入式计算市场需求的理想模块。在嵌入式市场中, Toradex 因其产品的稳定性、长生命周期、免费的高级技术支持和透明的价格体系而与众不同。Toradex 能够帮助用户降低其嵌入式产品的上市时间以及研发成本。

9.2 本地支持联系方式

韜睿 (上海) 计算机科技有限公司

公司网址 – www.toradex.cn

公司地址 – 上海市徐汇区虹桥路 188 号 1806 室

联系电话 - 021-54380582

以上所有的信息仅供您的参考, 其中并不包含任何承诺。Toradex 会不定期发布软件更新, 以上信息不保证能够适用于最新的软件。关于文中任何的错误、遗漏或者链接内容, 我们对此不承担责任。