

制作ZCU102的PYNQ镜像



水生植物
一直在前进，偶尔会反思

6 人赞同了该文章

目录

收起

- 1. WSL的安装
 - 1.1 WSL的概念
 - 1.2 安装WSL命令
 - 1.3 设置Linux用户名和密码
 - 1.4 Linux系统的根目录在win...
 - 1.5 更新和升级包
 - 1.6 安装桌面环境
 - 1.7 安装Windows Terminal...
- 2. 下载安装vitis
- 3. 下载安装petalinux
 - 3.1 安装TFTP（可选）
 - 3.2 安装petalinux
- 4. 制作ZCU102的镜像
 - 4.1 下载ZCU102的BSP以及P...
 - 4.2 制作镜像
- 5. 上板测试
 - 5.1 烧录镜像
 - 5.2 连接开发板
 - 5.3 用samba服务传输文件

PYNQ 就是python+ZYNQ的意思，简单来说就是使用python在Xilinx 的ZYNQ平台上进行开发。是Xilinx开发的一个新的开源框架，使嵌入式编程人员能够在无需设计可编程逻辑电路的情况下即可充分发挥 Xilinx Zynq All Programmable SoC（APSoC）的功能。使用PYNQ能够方便设计者：1. 快速验证设计模型，加速开发；2. 提升软硬件协同设计；3. 高效整合系统资源。本文重点对使用petalinux制作PYNQ镜像的全过程做一下记录。

1. WSL的安装

由于笔者使用windows系统，而petalinux只能在linux下运行，因此笔者安装了WSL，这是windows下linux的子系统，已经有linux系统或者虚拟机的读者可跳过。

1.1 WSL的概念

什么是WSL？按照microsoft官方的描述，WSL是适用于Linux的Windows子系统，可以让开发人员按原样运行GNU/Linux环境-包括大多数命令行工具、实用工具和应用程序，且不会产生传统虚拟机或双启动设置开销。安装wsl以后，可以在windows下：

- 选择偏好的GUN/Linux分发版
- 运行常用的命令行软件工具（例如grep、sed、awk）或其他ELF-64二进制文件
- 运行Bash shell脚本和GNU/Linux命令行应用程序，包括：
- 工具：vim、emacs、tmux
- 语言：NodeJS、JavaScript、Python、Ruby、C/C++、C#与F#、Rust、Go等
- 服务：SSHD、MySQL、Apache、lighttpd、MangoDB、PostgreSQL
- 使用自己的GNU/Linux分发包管理器安装其他软件
- 使用类似于Unix的命令行shell调用Windows应用程序
- 在Windows上调用GNU/Linux应用程序
- 运行直接集成到Windows桌面的GNU/Linux图形应用程序
- 将GPU加速用于机器学习、数据科学场景等

目前WSL已经推出了新版本WSL 2，它支持适用于Linux的Windows子系统在Windows上运行ELF64 Li

赞同 6 5 条评论 分享 喜欢 收藏 申请转载 ...

查看windows版本以确定是否支持WSL

必须运行Windows 10 版本 2004 或更高版本 或windows 11。使用win + R, 然后输入winver, 选择确定, 可检查windows版本及内部版本号。

1.2 安装WSL命令

可以在管理员 PowerShell 或 Windows 命令提示符中输入此命令, 然后重启计算机来安装运行适用于 Linux 的 Windows 子系统 (WSL) 所需的全部内容。使用win + x以及shift + A来打开管理员PowerShell。

```
wsl --install
```

此命令将启用运行 WSL 并安装 Linux 的 Ubuntu 发行版所需的功能。 ([可以更改此默认发行版](#))。

这一步安装完成后需要重启计算机以生效。

使用wsl --list --online来查看可用的发行版列表并运行wsl --install -d <DistroName>来安装发行版

```
wsl --list --online  
wsl --install -d Ubuntu-20.04
```



1.3 设置Linux用户名和密码

安装结束后会弹出Linux终端界面，也可以在开始菜单中打开Linux终端。

在刚完成安装以后，系统要求你为Linux发行版创建用户名和密码。需要注意：

- 此**用户名**和**密码**特定于安装的每个单独的 Linux 分发版，与 Windows 用户名无关。
- 请注意，输入 **密码**时，屏幕上不会显示任何内容。这称为盲目键入。不会看到正在键入的内容，这完全正常。
- 创建**用户名**和**密码**后，该帐户将是分发版的默认用户，并将在启动时自动登录。
- 此帐户将被视为 Linux 管理员，能够运行 sudo (Super User Do) 管理命令。
- 在 WSL 上运行的每个 Linux 发行版都有其自己的 Linux 用户帐户和密码。每当添加分发版、重新安装或重置时，都必须配置一个 Linux 用户帐户。

如果忘记了 Linux 分发版的密码：

1. 请打开 PowerShell，并使用以下命令进入默认 WSL 分发版的根目录：wsl -u root 如果需要在非默认分发版中更新忘记的密码，请使用命令：wsl -d Debian -u root，并将 Debian 替换为目标分发版的名称。
2. 在 PowerShell 内的根级别打开 WSL 发行版后，可使用此命令更新密码：passwd <username>，其中 <username> 是发行版中帐户的用户名，而你忘记了它的密码。
3. 系统将提示你输入新的 UNIX 密码，然后确认该密码。在被告知密码已成功更新后，请使用以下命令在 PowerShell 内关闭 WSL：exit。

1.4 Linux系统的根目录在windows下的位置

在资源管理器当中输入如下命令可以到达Linux文件系统的根目录

```
\\wsl$
```

1.5 更新和升级包

执行以下命令来更新和升级包

```
sudo apt update && sudo apt upgrade
```

1.6 安装桌面环境

linux系统安装桌面程序

```
sudo apt-get install ubuntu-desktop unity compizconfig-settings-manager
```

下载并安装vcxsrv

[VcXsrv Windows X Server download | SourceForge.net](#)

打开及设置

按如下方式设置，其余默认即可，并保存



首发于
PYNQ和Vitis AI的学习与研究

切换模式



设置防火墙

打开xLai

然后我们可以在Linux下使用Gnome的文件管理器

```
nautilus
```

运行效果如下图

1.7 安装Windows Terminal改善体验（可选）

直接在微软商店中下载

在Windows上安装VSCode以后，可以在WSL的终端输入code，即可用VSCode打开当前文件夹，代替了原本丑陋的UI界面，显得更为清爽。

2. 下载安装vitis

注意：安装vitis需提前预留200G的磁盘空间，如果磁盘大小刚刚够，则在安装的时候不要向磁盘中下载其他东西

首先百度vitis

找到vitis的linux安装包下载，我这里选的是2022.1的版本



给文件可执行权限以后运行

一路next安装即可，修改目录到/home/zchliu/Xilinx，安装过程大约持续6个小时



首发于
PYNQ和Vitis AI的学习与研究

切换模式

安装完成后选择Manage Licenses



首发于
PYNQ和Vitis AI的学习与研究

切换模式

选择Load license, Copy License

把准备好的License文件中即可完成安装

可以在安装目录下看到安装好的vitis, vivado, HLS等工具

至此vitis的安装已经完成了

3. 下载安装petalinux

3.1 安装TFTP（可选）

TFTP作为一种简单的文件传输协议，在嵌入式开发中会经常使用到，而且后面我们在安装petalinux的工具时也会提示使用到TFTP服务，所以我们需要在Ubuntu上搭建TFTP服务器。需要安装tftp

TFTP需要一个文件夹来存放文件，我们在工作目录下新建一个~/tftpboot目录作为TFTP文件存储目录，后面使用petalinux工具的时候都默认使用该目录。创建~/tftpboot文件夹的命令如下：

```
sudo mkdir -p /tftpboot
sudo chmod 777 /tftpboot
```

这样，就在工作目录上创建了一个名为tftpboot的文件夹，需要注意的是，我们要给该目录权限，否则在后面运行的时候会遇到bug。

最后需要配置tftp。打开/etc/default/tftpd-hpa文件，将其内容修改为

```
# /etc/default/tftpd-hpa

TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/tftpboot"
TFTP_ADDRESS=":69"
TFTP_OPTIONS="-l -c -s"
```

TFTP_DIRECTORY就是上面我们创建的~/tftpboot目录，以后所有需要通过TFTP传输的文件都放到该文件夹里面。

最后，启动TFTP服务器

```
sudo service tftpd-hpa start
```

3.2 安装petalinux

首先百度petalinux

点击进去以后找到下载petalinux

直接下载安装包即可

下载下来以后是一个linux下的二进制可执行文件，我们将其放到linux下运行

```
petalinux-v2022.1-04191534-installer.run
```

可以通过在windows资源管理器中输入如下命令进入linux子系统的根目录，将文件复制进去

```
\\ws1$\Ubuntu-20.04\home\zchliu
```

在linux系统下给安装文件给予可执行权限

```
sudo chmod a+x ./petalinux-v2022.1-04191534-installer.run
```

执行安装文件，--dir后面跟上要安装的目录地址

```
mkdir -p ~/Xilinx/petalinux/2022.1  
./petalinux-v2022.1-04191534-installer.run --dir ~/Xilinx/petalinux/2022.1
```

安装过程会检查环境并提示是否有依赖包未安装，根据提示安装即可

```
sudo dpkg --add-architecture i386  
sudo apt-get update  
sudo apt-get install gcc net-tools xterm autoconf libtool texinfo zlib1g zlib1g-dev gc
```



安装完成以后，我们还需要进行最后一个设置，petalinux工具需要主机系统的/bin/sh是bash，而Ubuntu默认的/bin/sh是dash，所以这里需要进行修改，如下：

```
sudo dpkg-reconfigure dash
```

选择NO|

在一般默认情况下，我们使用bash作为登录shell，所以source脚本文件settings.sh对petalinux所需的运行环境进行配置，命令如下：

```
source settings.sh
```

至此，petalinux安装完毕

4. 制作ZCU102的镜像

其实PYNQ为一部分板子提供了官方编译好的镜像，使用这些板子可以直接从pynq.io/board.html处下载。然而可惜的是，ZCU102不在此之列，因此，我们需要对其镜像进行手动编译。官方提供镜像的板子列表：

4.1 下载ZCU102的BSP以及PYNQ的文件系统

打开xilinx下载petalinux的界面往下划，能找到ZCU102的BSP下载链接

打开pynq的下载地址pynq.io/board.html，找到aarch64的文件系统下载，最好用迅雷会快一点（这个过程需要科学上网！不会的请自行百度）。这个预编译的文件系统和PYNQ二进制文件能够加速SD卡镜像的编译过程。

4.2 制作镜像

进入linux工作目录

下载pynq源代码

```
git clone --recursive https://github.com/Xilinx/PYNQ.git
cd PYNQ
git checkout origin/image_v3.0.1
```

使用git branch -a查看分支，分支选用和下载的vitis匹配的那一个

安装依赖环境，这个过程会安装qemu，以及交叉编译工具（该过程只需执行一次，再次执行会把qemu，crosstool删除以后重新安装）

```
./sdbuild/scripts/setup_host.sh
```

将ZCU104的文件复制为ZCU102的

```
cp -rf ./boards/ZCU104 ./boards/ZCU102
rm -rf ./boards/ZCU102/petalinux_bsp/
mv ./boards/ZCU102/ZCU104.spec ./boards/ZCU102/ZCU102.spec
```

修改ZCU102.spec的内容

```
sed -i -e "s/104/102/g" ./boards/ZCU102/ZCU102.spec
vim ./boards/ZCU102/ZCU102.spec
> BSP_ZCU102 := xilinx-zcu102-v2022.1-final.bsp
> STAGE4_PACKAGES_ZCU102 := xrt ethernet sensorconf boot_leds
```

修改好以后的内容如下

```
ARCH_ZCU102 := aarch64
BSP_ZCU102 := xilinx-zcu102-v2022.1-final.bsp
BITSTREAM_ZCU102 := base/base.bit
FPGA_MANAGER_ZCU102 := 1

STAGE4_PACKAGES_ZCU102 := xrt ethernet sensorconf boot_leds
```

拷贝ZCU102的bsp到./boards/ZCU102/下

将下载的根文件系统和预编译的二进制代码拷贝到./sdbuild/prebuilt/目录下并且如图进行重命名

配置环境变量

```
PATH=/opt/qemu/bin:/opt/crosstool-ng/bin:$PATH
source ~/Xilinx/Vitis/2022.1/settings64.sh
source ~/Xilinx/Vivado/2022.1/settings64.sh
source ~/Xilinx/petalinux/2022.1/settings.sh
```

进入sdbuild文件夹进行编译，过程中会提示缺少一些依赖包，根据提示完成安装

```
sudo apt-get install libtinfo5
cd ./sdbuild
make BOARDS=ZCU102
```

镜像编译的过程中也许会发生chroot: failed to run command 'cpio' : Exec format error的错误

从文件名可以看到，这是一个aarch64架构下编译的rpm文件，所以cpio会显示格式错误，大概从网络上搜索了一下相关的资料，可以通过启动binfmt服务来解决

[chroot: failed to run command 'dpkg-deb' : Exec format error · Issue #3779 · microsoft/WSL \(github.com\)](#)

[I wonder if WSL will support the i386 \(32 bit\) program running later? · Issue #2468 · microsoft/WSL \(github.com\)](#)

参考上述回答的解决办法，我们执行如下命令

```
sudo apt-get update
sudo apt-get install qemu-user-static
```

使用update-binfmts --display查看对应的架构

```
qemu-aarch64 (enabled):
  package = qemu-user-static
    type = magic
    offset = 0
    magic = \x7f\x45\x4c\x46\x02\x01\x01\x00\x00\x00\x00\x00\x00\x00\x00\x02\x0
    mask = \xff\xff\xff\xff\xff\xff\xff\x00\xff\xff\xff\xff\xff\xff\xff\xff\xfe\x
  interpreter = /usr/bin/qemu-aarch64-static
  detector =
```



```
sudo service binfmt-support start
```

删除jammy.ZCU102文件夹，重新编译

```
cd sdbuild  
sudo rm -rf build/jammy.ZCU102  
make BOARDS=ZCU102
```

经过一段时间的等待，镜像就编译完成了

可以在./sdbuild/output/boot/ZCU102下看到编译好的镜像文件

5. 上板测试

5.1 烧录镜像

首先下载Win32 Disk Imager，这是一个烧写镜像的程序

[Download Win32 Disk Imager from SourceForge.net](#)

一路安装即可

打开以后长这样

找到一个没用的SD卡



首发于
PYNQ和Vitis AI的学习与研究

切换模式



连上电脑以后格式化并且删除卷

找到编译好的镜像烧录，建议先将wsl下的镜像拷贝到windows下再烧录

将写入镜像以后的sd卡插入ZCU102开发板USB



5.2 连接开发板

笔者这里上电的时候还出了一个问题，显示UART驱动没有装好

打开Silicon的网站[Silicon Labs \(silabs.com\)](https://silabs.com)，搜索CP2108，找到对应的驱动程序并下载

下载以后安装即可

打开mobaxterm，设置端口和波特率连接

打开开发板开关，能看到一系列系统启动命令，随后进入到PYNQ的目录下，用户名和密码都是xilinx

在浏览器输入pynq:9090能够进入到jupyter的目录下

5.3 用samba服务传输文件

在windows的资源管理器下面输入\\pynq\xilinx

用户名和密码都是xilinx，将其输入即可进入pynq的文件目录



至此，ZCU102的PYNQ镜像就制作好了，接下来我会用PYNQ平台试着开发一些有意思的东西

参考资料

pynq.io/community.html

pynq.io/board.html

[Xilinx/PYNQ: Python Productivity for ZYNQ \(github.com\)](#)

[UG1144: PetaLinux Tools Documentation Reference Guide](#)

[\(78条消息\) ZCU102制作PYNQ v2.7镜像_国庆打七折的博客-CSDN博客_pynq编译](#)

编辑于 2022-12-02 11:29 · IP 属地陕西

[PYNQ开发](#) [嵌入式开发](#) [FPGA开发](#)

写下你的评论...

5 条评论

默认 最新



没头脑

博主您好，方便分享一下镜像么，邮箱315045424@qq.com

2023-10-10 · IP 属地陕西

回复 喜欢



萝卜啊

大佬，请问可以分享最终编译得到的镜像么

z

2023-10-09 · IP 属地中国香港

回复 喜欢



微冷

请问有最终编译得到的镜像吗

2023-04-22 · IP 属地北京

回复 喜欢



萝卜啊

您好，请问您现在有编译得到的镜像么🙏求求了

2023-08-15 · IP 属地北京

回复 喜欢

文章被以下专栏收录



PYNQ和Vitis AI的学习与研究

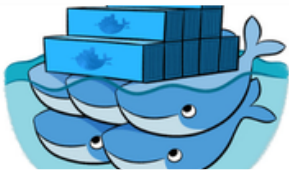
记录一些PYNQ和Vitis AI的学习过程

推荐阅读

Docker容器和镜像的区别

在学习使用docker过程中会遇到镜像和容器，两者之间的区别是什么？有什么关联？本文将用图文并茂的方式介绍容器、镜像的区别，能够很好的帮助各位深入理解Docker。一、什么是镜像？ 1.1...

小绿豆



Docker镜像配置深度学习环境 (Ubuntu 18.04)

XuanA...

发表于Pytor...



关于docker容器和镜像的区别

欣然

发表于JAVA高...



Docker系列-(2) 镜像制作与发布

码老思

发表于后端精进之...

