

公告

昵称: mfmdaoyou  
园龄: 3年2个月  
粉丝: 18  
关注: 0  
+加关注

< 2018年8月 >						
日	一	二	三	四	五	六
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

搜索

找找看

谷歌搜索

常用链接

[我的随笔](#)  
[我的评论](#)  
[我的参与](#)  
[最新评论](#)  
[我的标签](#)

友情链接

[test](#)

最新评论

1. Re:Could not connect to SMTP  
host: localhost, port: 25;  
String server =  
"smtp.163.com";String username  
= "";String password = "";  
Transport transport = ses.....  
--如花姐姐

2. Re:Cannot use ImageField  
because Pillow is not installed  
按照提示安装以后，还是出现同样的问  
题，提示我未安装pillow。重复安装几次  
还是不行  
--PaiPai\_hu

3. Re:Spring MVC新手教程（一）  
可不可以分享一下当前教程中使用的jar  
包？谢谢  
--奋斗ℓ

4. Re:《人工智能教程（张仰森）》  
（二）  
【决胜AI】人工智能与深度学习实战课  
程（深度学习 机器学习 人工智能  
python 数据分析 数据挖掘 Tensorflow  
Caffe）网盘下载: panbaiducom/s  
/1nuTn7fV 密码:.....  
--不为无学之

5. Re:最长递增子序列  
既然喜欢编程之美，不妨看看构建之法  
--SoftwareTeacher

随笔-1721 文章-1 评论-5

从零使用qemu模拟器搭建arm执行环境

为什么会有这篇文章

早在2011年的时候，跟当时同事一起讨论，做Linux系统开发正处于整个Linux开发中间层，没有上层的C/C++业务和数据库的开发经验。也没有底层的内核和驱动开发经验，究竟路该怎样走.....基于对Linux系统开发的理解和内核的兴趣。觉得选择Linux内核开发更适合自己。于是到淘宝上买了块三星s3c2440(arm 9)开发板，学起内核开发。没有过多久。机缘巧合，正式添加了公司的内核开发部。就这样跟内核和arm打上交道了。

没有想到这一做就是3年了，arm芯片仅仅有在公司才干使用，回到家里就不能访问了。去年（2014）开始觉得做内核久了，应该向内核社区提交patch，提升知名度和影响力。但在公司提交patch不方便。于是在家里通过qemu方式搭建于ARM A9的执行环境，进行开发和测试验证。一口气提交了好几个patch ([link1](#), [link2](#), [link3](#), [link4](#), [link5](#)) 并被社区接纳了。

近期在梳理Linux内存机制，不管是《深入理解Linux内核》还是《深入理解内核架构》这两本红宝书都无法告知你每一个细节的时候。就须要查看代码细节。改动代码。甚致做行为分析。

此时须要改动代码输出调试信息。以帮助更深入体会代码的逻辑。

这该是qemu派上场的时候。

事实上我前前后后搭建qemu+arm的执行环境已超过5次了。每次都要花上非常多时间。碰巧昨天有同事看到我再次搭建。他如搭珍宝，告诉我一定要将搭建qemu的方法告诉他。

所以，假设你想买个开发板来做arm + linux嵌入式开发，全然能够使用qemu进行开发。或者你像我一样，对内核机制关心，而不关心于详细的外设器件。最多是关心arm架构相关的功能，也能够使用qemu进行开发。

一句话：搭建qemu+arm环境，用于做内核开发和功能分析调试。

搭建好开发环境

我整个搭建过程都是在笔记本上进行的，Ubuntu 12.04系统；假设是Fodera环境。搭建过程可能略有不同。但关键步骤是不变的。

qemu模拟得最好的arm芯片，要数ARM公司的vexpress A9开发板了，本文的搭建过程都是环绕这个开发板进行的。

当然。假设你想搭其他开发板，也不难，仅仅要qemu和内核对它有成熟的支持就够了。

以下是step by step的搭建过程。建议没有特别诉求的朋友，依照下在面的步骤操作。或者先依据以下的步骤成功搭建vexpress执行环境之后。再依据自己的需求进行更改。

下载Linux内核

下载内核有两种方法，一种是用git直接下载内核代码树，方便后面的内核开发。还有一种是直接到内核社区下载相应版本号的源代码包。

我採用第一种方法，但后面发现主线上3.18版本号和后面版本号的代码，使用这样的搭建方法执行不起来。

眼下未查明问题的根因。

#### 阅读排行榜

1. 彻底明确怎样设置minSdkVersion和targetSdkVersion(14424)
2. 将iPhone投影到Mac上(11661)
3. Spring MVC新手教程 (一) (9138)
4. 关于HTML中文乱码问题(6739)
5. 最全Pycharm教程 (11) —— Pycharm调试器之断点篇(5832)

#### 评论排行榜

1. Spring MVC新手教程 (一) (1)
2. 《人工智能教程（张仰森）》 (二) (1)
3. Cannot use ImageField because Pillow is not installed(1)
4. 最长递增子序列(1)
5. Could not connect to SMTP host: localhost, port: 25;(1)

#### 推荐排行榜

1. mysql数据库分库分表(Sharding) (2)
2. AngularJS实现跨域请求(1)

假设读者想高速搭建成功。建议选用3.16版本号的内核进行搭建。

方法一：使用git

```
git clone git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git
```

方法二：直接下载3.16源代码包

```
wget https://www.kernel.org/pub/linux/kernel/v3.x/linux-3.16.tar.xz
```

### 安装arm的交叉编译工具链

想必做嵌入式开发的朋友，对交叉编译工具链不陌生。

假设你订制一个交叉编译工具链，建议你使用crosstool-ng开源软件来构建。但在这里建议直接安装arm的交叉编译工具链：

```
sudo apt-get install gcc-arm-linux-gnueabi
```

### 编译Linux内核

生成vexpress开发板子的config文件：

```
make CROSS_COMPILE=arm-linux-gnueabi- ARCH=arm vexpress_defconfig
```

编译：

```
make CROSS_COMPILE=arm-linux-gnueabi- ARCH=arm
```

生成的内核镜像位于arch/arm/boot/zImage，兴许qemu启动时须要使用该镜像。

### 下载和安装qemu模拟器

事实上Ubuntu 12.04有qemu的安装包。但由于版本号较低。对vexpress开发板支持不友好，建议下载高版本号的qemu：

```
wget http://wiki.qemu-project.org/download/qemu-2.0.2.tar.bz2
```

配置qemu前。须要安装几个软件包：

```
sudo apt-get install zlib1g-dev
```

```
sudo apt-get install libglib2.0-0
```

```
1 | <span style="font-family:'Microsoft YaHei'"><span style="font-size:18px">sudo apt-get install libglib2.0-dev</span></span>
```

配置qemu，支持模拟arm架构下的全部单板：

```
./configure --target-list=arm-softmmu --audio-driv-list=
```

编译和安装：

```
make
```

```
make install
```

### 测试qemu和内核是否能执行成功

3. DockerHub基于Github自己自动化构建(1)  
4. HT全量量化的图形组件设计(1)  
5. 对Class.getResourceAsStream和ClassLoader.getResourceAsStream方法所使用的资源路径的解释(1)

qemu已经安装好了。内核也编译成功了，到这里最好是测试一下，编译出来的内核是否OK，或者qemu对vexpress单板支持是否够友好。

执行命令非常easy:

```
qemu-system-arm -M vexpress-a9 -m 512M -kernel /home/ivan/kernel_git/linux/arch/arm/boot/zImage  
-nographic -append "console=ttyAMA0"
```

假设看到内核启动过程中的打印，说明前的搭建是成功的。

这里简介下qemu命令的参数:

-M vexpress-a9 模拟vexpress-a9单板，你能够使用-M ?参数来获取该qemu版本号支持的全部单板

-m 512M 单板执行物理内存512M

-kernel /home/ivan/kernel\_git/linux/arch/arm/boot/zImage 告诉qemu单板执行内核镜像路径

-nographic 不使用图形化界面，仅仅使用串口

-append "console=ttyAMA0" 内核启动参数。这里告诉内核vexpress单板执行。串口设备是哪个tty。

注意:

我每次搭建，都忘了内核启动参数中的console=参数应该填上哪个tty，由于不同单板串口驱动类型不尽同样，创建的tty设备名当然也是不同的。

那vexpress单板的tty设备名是哪个呢？事实上这个值能够从生成的.config文件CONFIG\_CONSOLE宏找到。

假设搭建其他单板，须要注意内核启动参数的console=参数值，同样地，可从生成的.config文件里找到。

### 制作根文件系统

到这里是否大功告成了呢？事实上在上面的测试中，你会发现内核报panic，由于内核找不到根文件系统，无法启init进程。

根文件系统要考虑两个方面:

#### 1. 根文件系统的内容

假设你看过《Linux From Scratch》。相信你会对这一步产生恐惧感。但假设一直从事嵌入式开发。就能够放下心来。根文件系统就是简单得不能再简单的几个命令集和态动态而已。为什么Linux From Scratch会有那么复杂。是由于它要制作出一个Linux发生版。但在嵌入式领域，差点儿全部的东西，都是mini版本号，根文件系统也不例外。

本文制本的根文件系统 = busybox(包括基础的Linux命令) + 执行库 + 几个字符设备

#### 2. 根文件系统放在哪里

事实上依赖于每一个开发板支持的存储设备，能够放到Nor Flash上。也能够放到SD卡，甚至外部磁盘上。

最关键的一点是你要知道开发板有什么存储设备。

本文直接使用SD卡做为存储空间，文件格式为ext3格式

### 下载、编译和安装busybox

```
wget http://www.busybox.net/downloads/busybox-1.20.2.tar.bz2
```

```
make defconfig
```

```
make CROSS_COMPILE=arm-linux-gnueabi-
```

```
make install CROSS_COMPILE=arm-linux-gnueabi-
```

安装完毕后。会在busybox文件夹下生成\_install文件夹。该文件夹下的程序就是单板执行所须要的命令。

### 形成根文件夹结构

先在Ubuntu主机环境下，形成文件夹结构，里面存放的文件和文件夹与单板上执行所须要的文件夹结构全然一样。然后再打包成镜像（在开发板看来就是SD卡）。这个暂时的文件夹结构称为根文件夹

1. 创建rootfs文件夹（根文件夹），根文件系统内的文件全部放到这里

```
sudo mkdir rootfs
```

2. 拷贝busybox命令到根文件夹下

```
sudo cp busybox-1.20.2/_install/* -r rootfs/
```

3. 从工具链中拷贝执行库到lib文件夹下

```
sudo cp -P /usr/arm-linux-gnueabi/lib/* rootfs/lib/
```

4. 创建4个tty终端设备

```
sudo mknod rootfs/dev/tty1 c 4 1
```

```
sudo mknod rootfs/dev/tty2c 4 2
```

```
sudo mknod rootfs/dev/tty3c 4 3
```

```
sudo mknod rootfs/dev/tty4c 4 4
```

### 制作根文件系统镜像

1. 生成32M大小的镜像

```
dd if=/dev/zero of=a9rootfs.ext3 bs=1M count=32
```

2. 格式化ext3文件系统

```
mkfs.ext3 a9rootfs.ext3
```

3. 将文件复制到镜像中

```
sudo mkdir tmpfs
```

```
sudo mount -t ext3 a9rootfs.ext3 tmpfs/ -o loop
```

```
cp -r rootfs/* tmpfs/
```

```
sudo umount tmpfs
```

## 系统启动执行

完毕上述全部步骤之后。就能够启动qemu来模拟vexpress开发板了，命令参数例如以下：

```
qemu-system-arm -M vexpress-a9 -m 512M -kernel /home/ivan/qemu/linux/arch/arm/boot/zImage  
-nographic -append "root=/dev/mmcblk0 console=ttyAMA0" -sd a9rootfs.ext3
```

从内核启动打印。到命令行提示符出现，激动人心的时刻出现了.....

## 写在后面的话

通过上面的步骤。搭建出来一个最小的qemu+arm开发环境，你能够上面的基础上改动内核。或者添加一些测试程序在单板上执行，甚至使用单板的flash设备。

在此，你能够做纯arm架构的内核开发。或者与架构无关的内核开发，也能够做单板相关的驱动开发。



“扫码打赏作者”

好文要顶

关注我

收藏该文

mfmdayou  
关注 - 0  
粉丝 - 18

+加关注

« 上一篇: hdu 2255 奔小康赚大钱 (KM算法)  
» 下一篇: Uva 10081 Tight words (概率DP)

posted @ 2017-06-02 16:48 mfmdayou 阅读(3122) 评论(0) 编辑 收藏

努力加载评论中...

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】超50万VC++源码：大型组态工控、电力仿真CAD与GIS源码库！

【前端】SpreadJS表格控件，可嵌入应用开发的在线Excel

【免费】程序员21天搞定英文文档阅读

【推荐】如何快速搭建人工智能应用？



## 最新IT新闻:

- 红芯生死局：我道歉，我们改
- “机”动时刻| 演奏、踢球、冲咖啡.....这些事儿机器人都能干
- 微软在好莱坞开设“混合现实捕捉”工作室
- Debian Linux迎来25周岁生日

· 疯狂复制流水线“名师”，在线教育一对一陷入四重危机  
» 更多新闻...

**最新知识库文章:**

· 被踢出去的用户  
· 成为一个有目标的学习者  
· 历史转折中的“杭派工程师”  
· 如何提高代码质量?  
· 在腾讯的八年，我的职业思考  
» 更多知识库文章...

站长统计站长统计