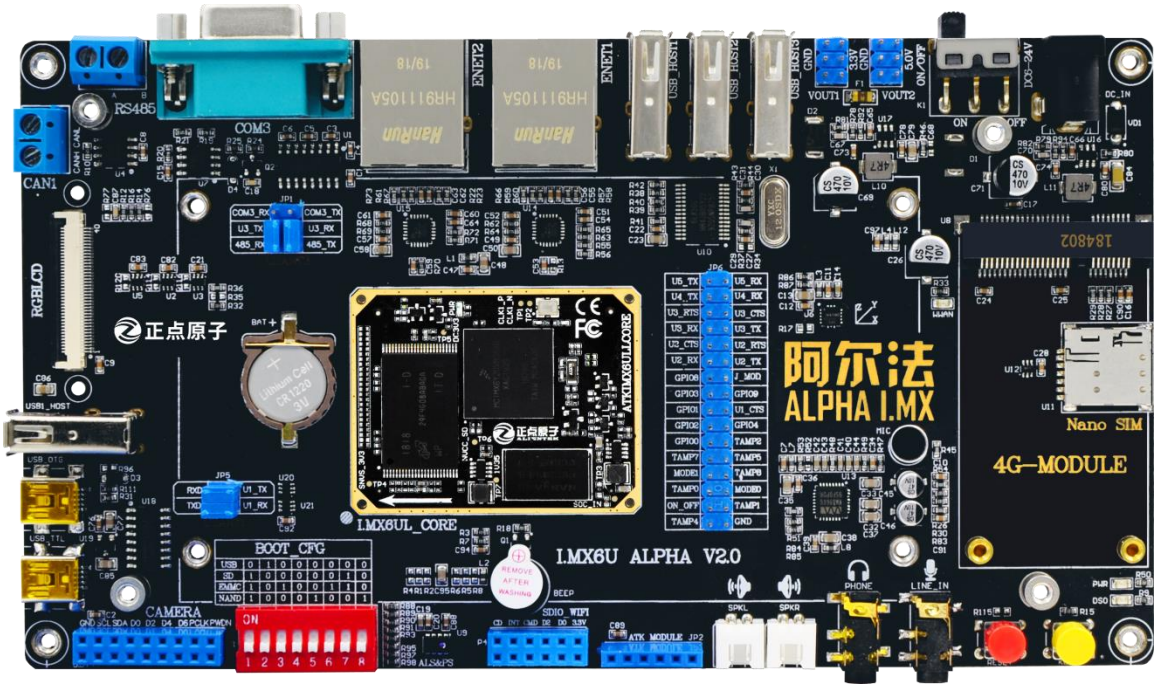
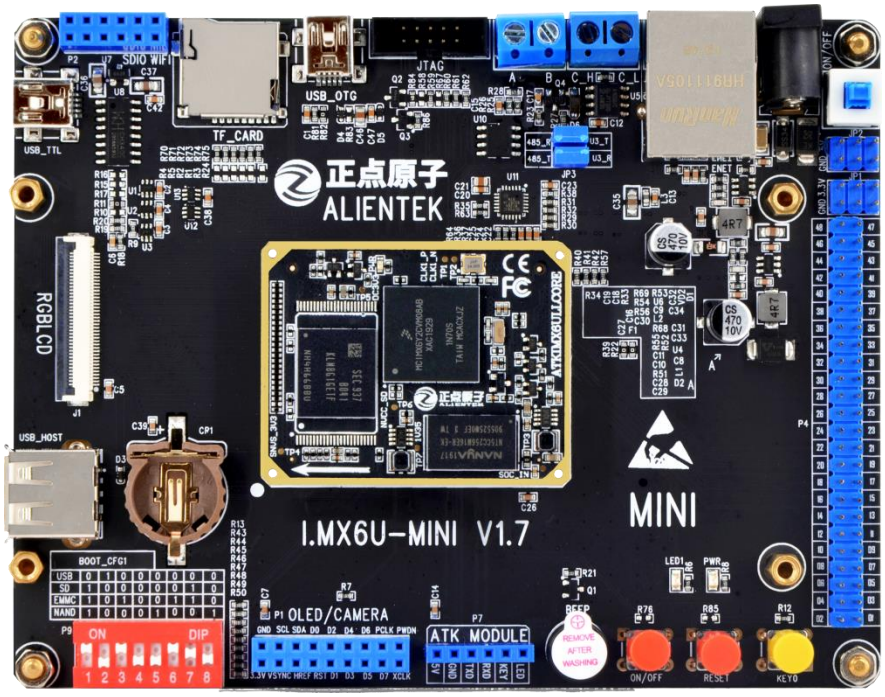
【正点原子】构建 Yocto 根文件系统 V1.3





正点原子广州市星翼电子科技有限公司

淘宝店铺 1：http://eboard.taobao.com

淘宝店铺 2：http://openedv.taobao.com

技术支持论坛 (开源电子网) ：www.openedv.com

原子哥在线教学：www.yuanzige.com

官方网站：www.alientek.com

最新资料下载链接：http://www.openedv.com/posts/list/13912.htm

E-mail: 1252699831@qq.com **QQ:** 1252699831

咨询电话：020-38271790

传真号码：020-36773971

团队：正点原子团队

正点原子，做最全面、最优秀的嵌入式开发平台软硬件供应商。

友 情 提 示

如果您想及时免费获取“正点原子”最新资料，敬请关注正点原子微信公众平台，我们将及时给您发布最新消息和重要资料。

关注方法：

(1)微信“扫一扫”，扫描右侧二维码，添加关注

(2)微信→添加朋友→公众号→输入“正点原子”→关注

(3)微信→添加朋友→输入“alientek_stm32”→关注

文档更新说明

版本	版本更新说明	负责人	校审	发布日期
V1.0	初稿：	正点原子 linux 团队	正点原子 linux 团队	2019.10.26
V1.1	1.更改 2.1 小节的目录结构 2.添加需要注意的问题，已经用红色字体标注。	正点原子 linux 团队	正点原子 linux 团队	2019.12.30
V1.2	1. 更改前言部分内容 2. 添加 repo 下载的备用链接 3. 添加 yocto 项目下载的备用链接 4. 添加第 2.1 小节，若离线编译报错 tslib.h，处理方法。	正点原子 linux 团队	正点原子 linux 团队	2020.3.7
V1.3	1. 删除离线构建，缩减文档文字。	正点原子 linux 团队	正点原子 linux 团队	2020.11.24

目录

前言

由正点原子用户反馈或者建议得出的经验，初学者没必要再构建 Yocto 了，建议把精力放在其他方面上。我们可以直接使用正点原子编译出来的 Yocto 文件系统（出厂根文件系统），没必要再去编译了，很耗时间。Yocto 项目庞大复杂，并非一本小小的教程能说的清楚，所以我们只写如何构建（可以说只是体验构建过程）。除了构建文件系统，基本很少用。如果有足够的时间和精力，电脑性能够的话，不介意编译过程中可能会出错（没固定解决方法），就可以按下文开始体验 Yocto 项目的构建。如果想学习更多，请参考开发板光盘 A-基础资料->4、参考资料->i.MX_Yocto_Project_User's_Guide.pdf。如有错漏，请到正点原子论坛指正，或者联系本文档编写作者 QQ1252699831 指正错误。

本文档所使用的环境：

- Windows 7 64bits，也适用于 Windows 8-10。不建议用 Windows 32 位来开发，Windows 32 位支持的内存大小有限，系统性能有限。
- 用正点原子驱动指南教程里使用的 Ubuntu16.04，Ubuntu 建议使用 16.04。否则安装及编译环境不一样导致出错，请自行解决！
- 要求读者会使用 FileZilla、WinSCP 及 Windows Git 进行 Ubuntu 与 Windows 间互传文件的方法。
- 虚拟机磁盘至少分配 120G 空间。计算机内存分配给虚拟机至少 8GB 以上。核心数分配建议 8 个或者 16 个。

第一章 构建 Yocto 根文件系统

1.1 安装 Git 与配置 Git 用户信息

Yocto 项目需要用到 Git（用 Git 作版本控制）。如果您已经有安装过 git 并且已经设置过本地用户 git 信息就跳过此小节。

```
sudo apt-get install git
git config --global user.name "alientek"    // 配置 Git 用户名，输入个人用户名
git config --global user.email 12345678@qq.com // 输入 email 地址
git config --list          // 查看个人配置的信息
```

1.2 获取 Yocto 项目

repo 是一个构建在 Git 之上的工具，它可以更容易地管理包含多个存储库的项目，而这些存储库不需要位于同一台服务器上。repo 很好地补充了 yocto 项目的分层特性，使用户可以更容易地将自己的层添加到 BSP 中。简单来说，repo 用来管理多个 git 工程。

首先立工程项目文件夹，文件夹名称可随意。

```
mkdir fsl-release-bsp
cd fsl-release-bsp
```

拷贝开发板光盘 A-基础资料->1、例程源码->7、第三方库源码->repo.tar.bz2 到虚拟机，拷贝到上面我们创建的 fsl-release-bsp 目录下。

```
tar xf repo.tar.bz2
```

```
alientek@ubuntu:~/fsl-release-bsp$ tar xf repo.tar.bz2
alientek@ubuntu:~/fsl-release-bsp$ ls
repo  repo.tar.bz2
alientek@ubuntu:~/fsl-release-bsp$
```

图 1.2 1 解压 repo

获取 Yocto 项目，克隆 imx-4.1-krogoth 分支（此分支就是 I.MX6U 4.1.15 Linux 分支）。https://source.codeaurora.org/external/imx/fsl-arm-yocto-bsp 为新的仓库地址，在 i.MX_Yocto_Project_User's_Guide.pdf 这本手册里，旧的地地址已经不能使用！

```
chmod u+x repo
./repo init -u https://source.codeaurora.org/external/imx/fsl-arm-yocto-bsp -b imx-4.1-krogoth
```

```
alientek@ubuntu:~/fsl-release-bsp$ ./repo init -u https://source.codeaurora.org/external/imx/fsl-arm-yocto-bsp -b imx-4.1-krogoth
gpg: 钥匙环‘/home/alientek/.repoconfig/gnupg/secring.gpg’已建立
gpg: 钥匙环‘/home/alientek/.repoconfig/gnupg/pubring.gpg’已建立
gpg: /home/alientek/.repoconfig/gnupg/trustdb.gpg：建立了信任度数据库
gpg: 密钥 920F5C65：公钥“Repo Maintainer <repo@android.kernel.org>”已导入
gpg: 合计被处理的数量：1
gpg:               已导入：1

Getting repo ...
   from git://codeaurora.org/tools/repo.git
remote: Enumerating objects: 174, done.
remote: Counting objects:  80% (140/174)
```

图 1.2 2 初始化 repo 仓库

如无法获取，请重新执行。在获取的过程中会要求确认第 1.1 小节里配置的 Git 信息。连续按下两次 Enter 后，再输入两次 y 即可！

获取完成后，如下图，可以看到目录下多了一个.repo 文件夹。

```
ls -a
```

```
alientek@ubuntu:~/fsl-release-bsp$ ls -a
.  ..  repo  .repo  repo.tar.bz2
alientek@ubuntu:~/fsl-release-bsp$
```

图 1.2 3 获取的 repo 仓库

执行 repo sync，同步 NXP 官方 fsl-arm-yocto-bsp.git 项目的 imx-4.1-krogoth 分支。此过程可能十分漫长，建议在早上执行，作者测试过早上比较快。可能早上网络使用的不多吧，所下同步的时间比较快。同步过程中可能因为网络问题出错，解决办法只有重新执行！

```
./repo sync
```

```
alientek@ubuntu:~/fsl-release-bsp$ ./repo sync

Initializing project fsl-community-bsp-base ...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 275 (delta 1), reused 2 (delta 0), pack-reused 269
接收对象中: 100% (275/275), 61.04 KiB | 0 bytes/s, 完成.
处理 delta 中: 100% (148/148), 完成.
来自 git://github.com/Freescale/fsl-community-bsp-base
 * [新分支]          daisy      -> freescale/daisy
 * [新分支]          danny      -> freescale/danny
 * [新分支]          denzil     -> freescale/denzil
 * [新分支]          dizzy      -> freescale/dizzy
 * [新分支]          dora       -> freescale/dora
 * [新分支]          dunfell    -> freescale/dunfell
 * [新分支]          dylan      -> freescale/dylan
 * [新分支]          fido       -> freescale/fido
 * [新分支]          gatesgarth -> freescale/gatesgarth
 * [新分支]          jethro     -> freescale/jethro
 * [新分支]          krogoth    -> freescale/krogoth
 * [新分支]          master     -> freescale/master
 * [新分支]          master-next -> freescale/master-next
 * [新分支]          morty      -> freescale/morty
 * [新分支]          pyro       -> freescale/pyro
 * [新分支]          rocko      -> freescale/rocko
 * [新分支]          sumo       -> freescale/sumo
 * [新分支]          thud       -> freescale/thud
 * [新分支]          warrior    -> freescale/warrior
 * [新分支]          zeus       -> freescale/zeus
 * [新标签]          1.2        -> 1.2
 * [新标签]          1.2-rc1    -> 1.2-rc1
 * [新标签]          1.3        -> 1.3
 * [新标签]          1.6        -> 1.6
 * [新标签]          1.7        -> 1.7
 * [新标签]          1.8        -> 1.8
 * [新标签]          2.0        -> 2.0
 * [新标签]          2.1        -> 2.1
 * [新标签]          2.2        -> 2.2
 * [新标签]          1.4        -> 1.4
 * [新标签]          1.5        -> 1.5
Fetching projects:  22% (2/9)
Initializing project meta-browser ...
remote: Enumerating objects: 66, done.
remote: Counting objects: 100% (66/66), done.
remote: Compressing objects: 100% (26/26), done.
接收对象中:   2% (132/6582), 44.01 KiB | 67.00 KiB/s
```

图 1.2 4 repo 同步仓库

同步完成后可以看到有如下文件。

```
alientek@ubuntu:~/fsl-release-bsp$ ls
fsl-setup-release.sh  repo  repo.tar.bz2  setup-environment  sources
alientek@ubuntu:~/fsl-release-bsp$
```

图 1.2 5 同步完成

1.3 开始构建 Yocto 文件系统

构建 Yocto 的文件系统镜像，freescale（NXP）提供了几种镜像供我们构建，文件系统镜像支持的功能越多，文件系统就越大，读者可自行选择一个来构建。

下面是发行版（DISTRO）的四种配置：

- fsl-imx-x11 -只有 X11 图形
- fsl-imx-wayland - Wayland weston graphics
- fsl-imx-xwayland - Wayland graphics and X11。不支持使用 EGL 的 X11 应用程序
- fsl-imx-fb -帧缓冲图形-没有 X11 或 Wayland

支持的 machine 有很多种，以下是按官方文档列出的几种。

- imx6qpsabreauto
- imx6qpsabresd
- imx6ulevk
- imx6ull14x14evk
- imx6ull9x9evk
- imx6dlsabreauto
- imx6dlsabresd
- imx6qsabreauto
- imx6qsabresd
- imx6slevk
- imx6solosabreauto

- imx6solosabresd
- imx6sxsabresd
- imx6sxsabreauto
- imx7dsabresd
- 如下表格简单介绍了几种镜像

core-image-minimal	仅支持设备启动的小镜像
core-image-base	目标设备硬件的只支持控制台的镜像
core-image-sato	镜像支持 X11 与 Sato 主题和使用 Pimlico 应用程序。它包含一个终端、一个编辑器和一个文件管理器。
fsl-image-machine-test	FSL 社区的核心镜像，支持控制台环境，没有 GUI 界面
fsl-image-gui	一个支持 GUI 的镜像
fsl-image-qt5	一个开源 Qt5 的镜像

- 官方提供了 fsl-setup-release.sh 脚本来构建项目，脚本语法如下：
- DISTRO=<distro name> MACHINE=<machine name> source fsl-setup-release.sh -b <build dir>
- 参数解释：
- DISTRO：发行版类型
- MACHINE：机器类型
- -b:指定工程构建输出的目录
- 发行版我们选择 fsl-imx-x11，正点原子的发布的系统需要运行 Qt5 程序，所以选择 fsl-image-qt5 这个文件系统镜像来构建。我们开发板是 IMX6ULL 芯片，所以 MACHINE 选择是 imx6ull14x14evk。fsl-setup-release.sh 这个脚本会解释用户配置的参数。

安装编译所必须的库。下面为一行指令，建议复两行复制。

```
sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib build-essential chrpath socat libsdl1.2-dev curl
```

执行下面的指令开始配置，build 是构建的目录。

```
DISTRO=fsl-imx-x11 MACHINE=imx6ull14x14evk source fsl-setup-release.sh -b build
```

配置过程中请你阅读 EULA 内容，输入 q 退出阅读内容后再按 y 确认已经阅读，再继续下一步。

构建的时长的因素与以下几点有关：计算机的性能（硬盘写入速度，CPU 的核心数），网络下载速度，还与您的 Ubuntu 虚拟机在安装时分配的核心数与分配的内存有关！构建过程中可能会出错(报错的位置请根据提示的信息自行解决，没固定的解决方法)，可按 Ctrl +c 终止，再次输入 bitbake fsl-image-qt5 构建。Yocto 支持从断点处重新构建。因为构建过程中已经生成缓存文件，重复构建时会跳过已经构建过的任务。注：编者 Ubuntu 分配 16GB 内存（用户虚拟机能尽量分大内存就分大内存，否则可能出现编译不通过的情况），16 个核心数，网速为 100M，大约构建了 9 个小时！

开始构建含 Qt 库的根文件系统

```
bitbake fsl-image-qt5
```

构建完成后进入 build/tmp/deploy/images/imx6ull14x14evk 这个目录下。

```
alientek@ubuntu:~/fsl-release-bsp/build/tmp/deploy/images/imx6ull14x14evk$ ls
fsl-image-qt5-imx6ull14x14evk-20190904030729.rootfs.ext4      u-boot-sd-2016.03-r0.imx
fsl-image-qt5-imx6ull14x14evk-20190904030729.rootfs.manifest  zImage
fsl-image-qt5-imx6ull14x14evk-20190904030729.rootfs.sdcard    zImage--4.1.15-r0-imx6ull14x14evk-20190904030729.bin
fsl-image-qt5-imx6ull14x14evk-20190904030729.rootfs.tar.bz2   zImage--4.1.15-r0-imx6ull-14x14-evk-20190904030729.dtb
fsl-image-qt5-imx6ull14x14evk.ext4                            zImage--4.1.15-r0-imx6ull-14x14-evk-btwifi-20190904030729.dtb
fsl-image-qt5-imx6ull14x14evk.manifest                        zImage--4.1.15-r0-imx6ull-14x14-evk-emmc-20190904030729.dtb
fsl-image-qt5-imx6ull14x14evk.sdcard                          zImage--4.1.15-r0-imx6ull-14x14-evk-gpmi-weim-20190904030729.dtb
fsl-image-qt5-imx6ull14x14evk.tar.bz2                         zImage--4.1.15-r0-imx6ull-14x14-evk-usb-certi-20190904030729.dtb
modules--4.1.15-r0-imx6ull14x14evk-20190904030729.tgz         zImage-imx6ull14x14evk.bin
modules-imx6ull14x14evk.tgz                                   zImage-imx6ull-14x14-evk-btwifi.dtb
README_-_DO_NOT_DELETE_FILES_IN_THIS_DIRECTORY.txt            zImage-imx6ull-14x14-evk.dtb
u-boot.imx                                                    zImage-imx6ull-14x14-evk-emmc.dtb
u-boot-imx6ull14x14evk.imx                                    zImage-imx6ull-14x14-evk-gpmi-weim.dtb
u-boot-imx6ull14x14evk.imx-sd                                 zImage-imx6ull-14x14-evk-usb-certi.dtb
u-boot.imx-sd
alientek@ubuntu:~/fsl-release-bsp/build/tmp/deploy/images/imx6ull14x14evk$
```

图 1.3 1 查看编译的产物

生成的文件中，其中蓝色文字内容的是链接文件，这里就不列出了。

文件名	用途
*.rootfs.manifest	文件系统内的软件列表
*.rootfs.ext4	ext4 格式的文件系统压缩包
*.rootfs.sdcard	可直接写入 SD 卡，从 SD 卡启动的镜像
*.rootfs.tar.bz2	tar.bz2 格式的文件系统压缩包
modules*.tgz	ttgz 格式的内核模块压缩包
u-boot*.imx	官方多版本类型 u-boot 镜像
zImgae*/zImage*.bin	内核设备树与 bin 文件

我们需要的文件系统镜像压缩包是 fsl-image-qt5-imx6ull14x14evk-20190904030729.rootfs.tar.bz2，文件系统比较大，读者按需裁剪，或者按需修改 yocto 项目源文件再构建。

1.4 构建 SDK 工具

构建完成我们含 Qt5 的文件系统后，还需要构建 SDK 工具。SDK 工具里面含交叉编译 Qt 工具链。

同样地在 build 目录下接着执行，即可构建。

```
bitbake meta-toolchain-qt5
```

构建成功后，进入 build/tmp/deploy/sdk 目录，查看生成的文件。其中 fsl-imx-x11-glibc-x86_64-meta-toolchain-qt5-cortexa7hf-neon-toolchain-4.1.15-2.1.0.sh 是我们需要的文件。可参考【正点原子】I.MX6U 用户快速体验 V1.6.pdf 体验第四章使用此工具。

附录-A