## 在Windows上安装Git配置选项

git config --global user.name "richkoala"

git config --global user.email "zhouh\_12@163.com"

git config --global core.autocrlf false  //禁用自动转换

## 进入目录创建git版本库

cd </G/zho/dma\_/zynq\_dma/zynq\_V4/dma\_comm.sdk >

git init //这个目录变成Git可以管理的仓库

## 版本库中添加需要进行版本管理的文件

git add < top\_ecos >

git status //查看文件状态 1) No commits yet 2) Untracked files

## 提交版本管理

git commit –m “comment”

git commit --amend -no-edit //修订处理覆盖到上一次的版本，不增加新的版本，注意版本ID发生变化

## 远程控制

ssh-keygen -t rsa -C [zhouh\_12@163.com](mailto:zhouh_12@163.com)

github setting下添加ssh公钥

ssh-rsa  [zhouh\_12@163.com](mailto:zhouh_12@163.com)

## 本地关联远程库

git remote add origin <https://github.com/richkoala/vc_ecos.git>

## 推送内容至远程库

git push -u origin master

避免用户名密码推送方式

git remote rm origin

git remote add origin https://richkoala:zho13844883449 @github.com/richkoala/arm\_ecos.git

git push origin master

## 文件的删除

git rm < test.txt >

git rm -r -n --cached "bin/" //-n：加上这个参数，执行命令时，是不会删除任何文件，而是展示此命令要删除的文件列表预览。

git commit -m "remove test.txt"

## 分支管理

git checkout -b dev //创建分支并跳转至新的分支'dev'

等效为一下两条指令

git branch dev //创建’dev’分支

git checkout dev //跳转至’dev’分支

git branch //命令会列出所有分支，当前分支前面会标一个\*号。

$ git branch

\* dev

Master

git merge dev //将dev分支的工作成果合并到master分支上：

Updating d46f35e..b17d20e

Fast-forward

readme.txt | 1 +

1 file changed, 1 insertion(+)

$ rm -rf .git  // 删除.git

## 版本回溯

git log --oneline //查看版本信息

git reset –hard HEAD~<num> //回到前100个

git reset –hard gitid //回到固定版本号码

git reflog //查看每一次的head指针变化

通过head指针指向的ID完成向未来回溯

git reset –hard gitid

git push origin <master> -f //**强制**推送至远程回到一个固定的版本

## 单个文件回溯

git checkout < commit\_id > -- <文件名称>

#### 创建标签

git checkout <master> //切换到需要打标签的分支上

git tag <name> //创建新标签

git tag < name > < commit\_id> //为历史版本创建标签

git tag //查看标签

git show <tagname> //查看标签信息

git tag -a < name > -m "<tag info>" < commit\_id>

//还可以创建带有说明的标签，用-a指定标签名，-m指定说明文字

git push origin <tagname> //推送某个标签到远程

git push origin –tags //一次性推送全部尚未推送到远程的本地标签

git tag -d < name > //本地标签删除

远程标签删除（需要执行一下两个操作）

1）本地删除git tag -d <tagname>

2）远程删除 git push origin :refs/tags/v0.9