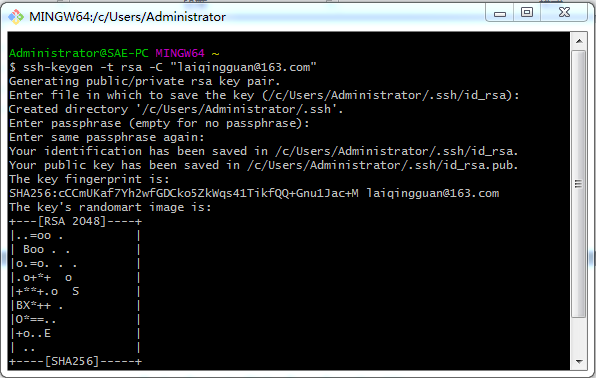
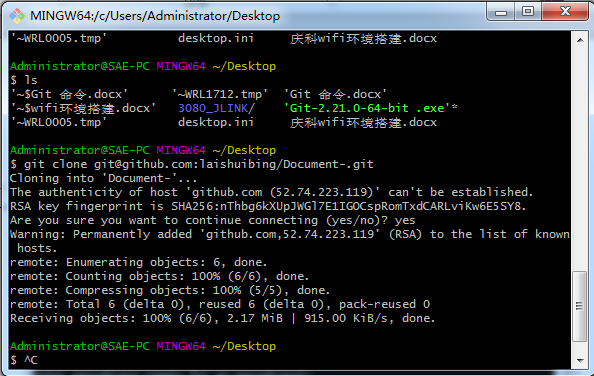
庆科wifi环境搭建步骤

1. 安装Git软件。
2. 一路回车，下一步，完成git软件安装。
3. 参考Git 命令文档完成创建远程仓库密钥。



1. Clone 远程仓库测试密钥是否成功。

$ git clone git@github.com:laishuibing/Document-.git

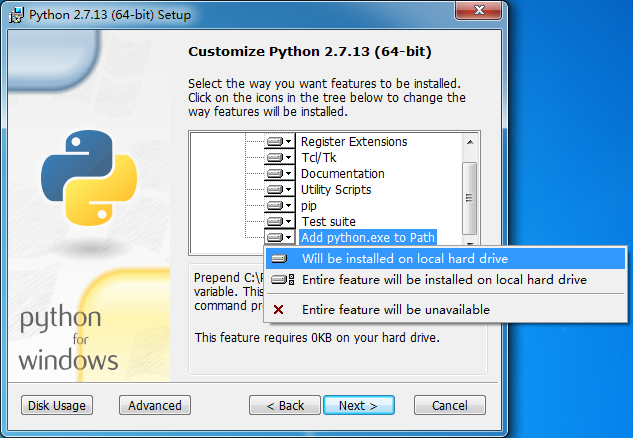


Git 设置完成。

1. 安装Python 2.7.13

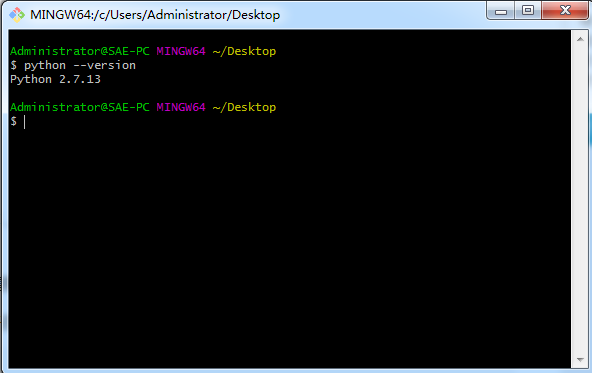
1.关闭Git bash。

2.双击运行Python2.7.13安装包。选择Add python.exe to path 选项，点击下一步完成即可。



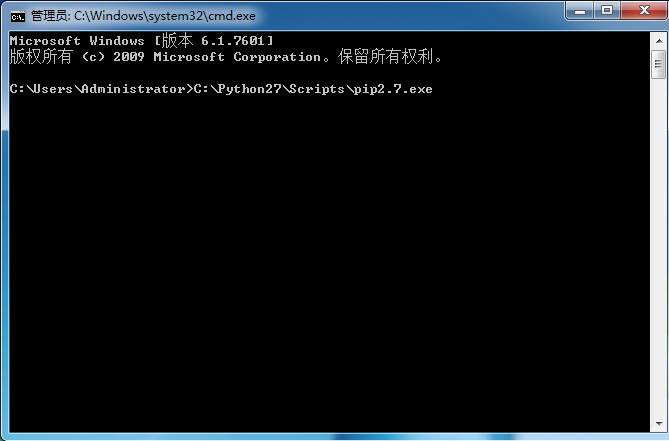
1. 验证Python是否安装成功。

打开Git bash，输入Python –version 命令。返回Python 2.7.13代表成功。



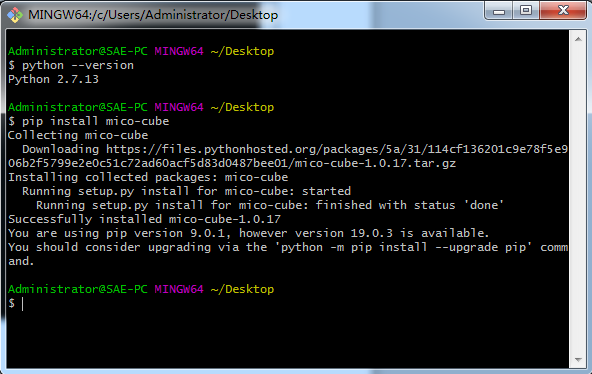
1. 设置python 环境变量

打开CMD 命令窗口，打开C:\Python27\Scripts目录，把pip2.7.exe直接拖入CMD命令窗口中。回车运行。

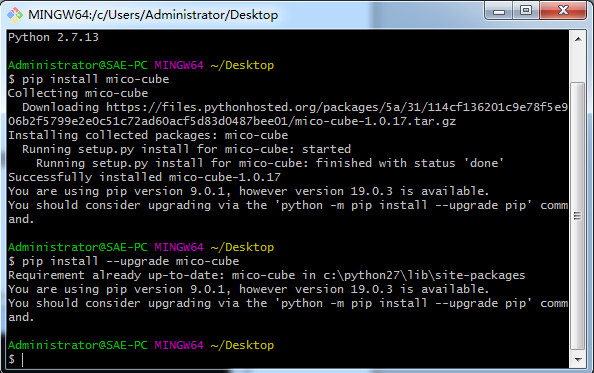


1. 安装Mico Cube

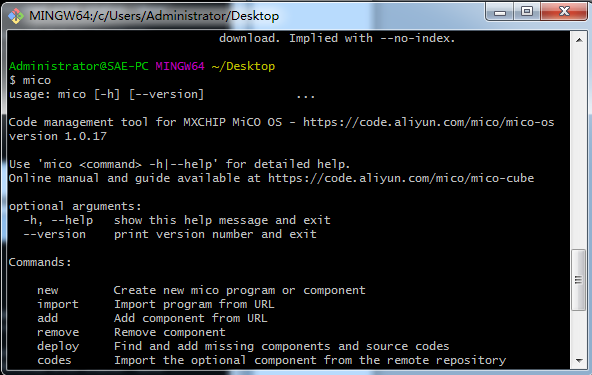
打开Git Bash 输入$ pip install mico-cube 命令。



输入$ pip install --upgrade mico-cube 命令升级Mico Cube

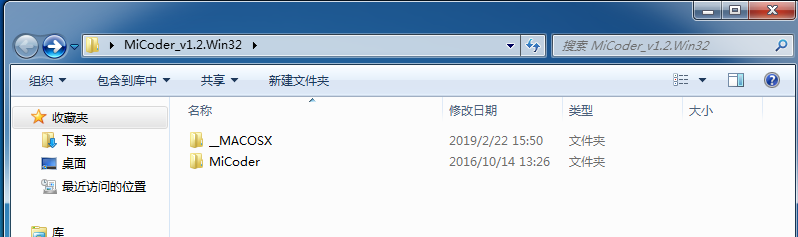


在Git Bash 输入mico 验证Mico Cube是否安装成功。

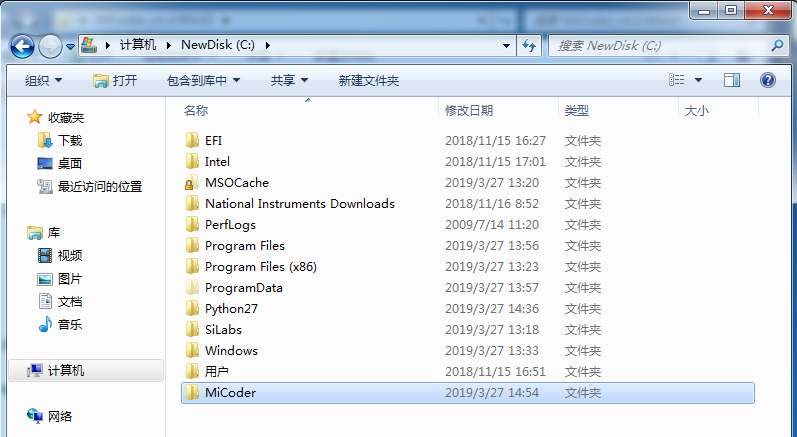


1. 下载MiCoder Tools 工具链。

点击<http://developer.mxchip.com/downloads/25>链接下载window版本的工具链。解压到指定文件夹。



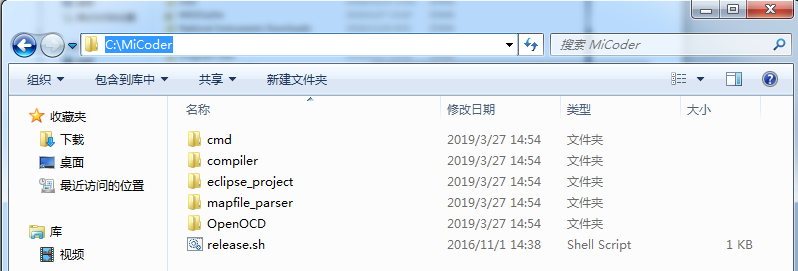
拷贝MiCoder 文件夹到C盘目录下。

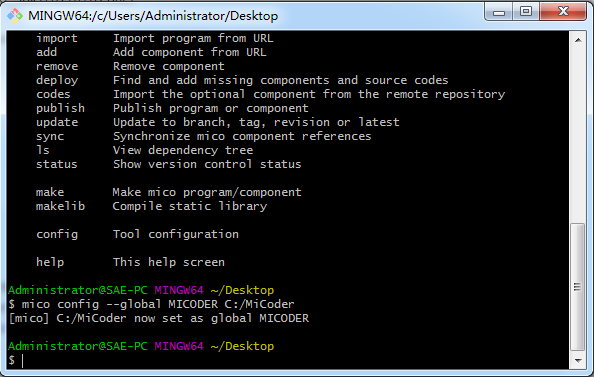


1. 设置设置MiCoder Tools的路径。

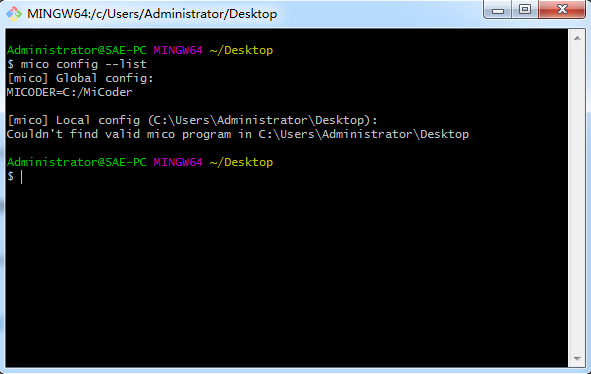
在Git Bash输入$ mico config --global MICODER C:/MiCoder 命令

（C:/MiCoder为Micoder路径，Git Bash需要把 \ 改成 /）





输入$ mico config –-list 命令确让是否成功



1. 创建一个新工程。
2. 新建工程：打开Git Bash进入到桌面中，输入$ mico new test -v –vv 命令创建一个test工程。
3. 导入工程：mico import mico-demos –v –vv 导入mico-demos例程。
4. 添加和移除组件

1.使用mico add 命令添加组件：$ mico add <https://code.aliyun.com/mico/Lib_aws.git>

2.使用 mico remove 命令移除组件 $ mico remove Lib\_aws

组件仓库地址：

<http://developer.mxchip.com/developer/md/bWljby1oYW5kYm9vay8xLk1pQ09faGFuZGJvb2tfZ3VpZGUvNC5taWNvX2NvbXBvbmVudHNfbGlzdC5tZA>

1. 代码编译

[target] [download] [run | debug] [JTAG=xxx] [VERBOSE=1] [JOBS=<jobs>]

编译并且下载基于 MiCOKit-3080B 的 Helloworld 应用程序和其他相关固件，使用命令： helloworld@MK3080B@moc total download JTAG=jlink\_swd

编译基于 MiCOKit-3165 的 Helloworld 应用程序，使用命令：helloworld@MK3165 JTAG=jlink\_swd

1. Git Bash进入到 mico文件夹中，执行

$ mico make test@MK3080B@MOC download JTAG=jlink\_swd run

命令即可。

