采用库的方式实现Arduino的终端解释运行框架

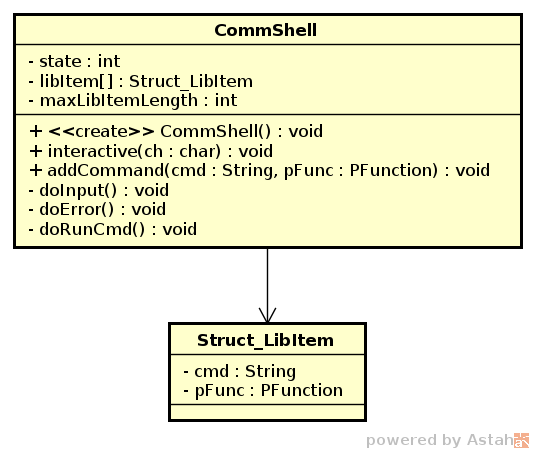
# Arduino库运行的原理

Arduino所谓的库运行，是将一系列的功能采用面向对象的方法进行封装，然后分割成头文件（\*\*\*.h）和源文件（\*\*\*.cpp）两个文件，将其置于Arduino的库目录中的方法。是的，不需要编译直接放到该目录中（单独建个目录）就行。然后在开发新的Arduino程序的时候，可以直接使用（通过菜单sketch-->include library）。

所以可以看出，Arduino的库其实只是源代码的复用。Arduino在编译用户程序的同时也会把其包含的库也编译，最后输出的是一个Arduino的映像文件。

# Arduino终端解释程序框架的库模式设计

所谓库模式设计，就是采用面向对象的方式对Arduino终端解释程序框架进行再设计。设计得到的模型如下图：



该设计得到的模型是最基本的模型，可以在后续软件开发的过程中进行修改，特别是参数方面没有考虑到具体的情况，还有错误处理和信息的打印等方面没有考虑，需要在实现的时候自己添加相应的内容。

# 测试

在Arduino程序中，在Setup()函数中执行：

CommShell shell = CommShell();

shell.addCommand(cmd, pCmdFun);

在loop（）函数中，执行：

Shell.interactive();

然后，打开Arduino IDE的串口应用程序，进行测试。