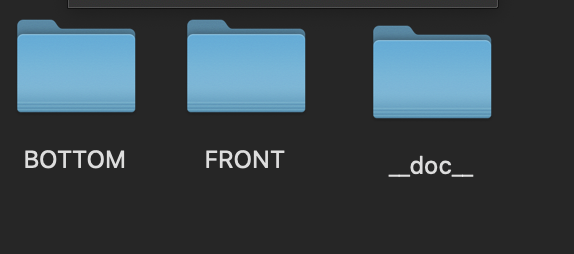
# 环境目录介绍

## 1-一级目录



BOTTOM文件夹中存放的是所有模块的代码。也就是所有上层环境调用的底层api，与具体某个环境无关，适用与所有环境。具体的每个环境都是根据调用模块。如test\_case\_manager ,work\_Info\_manager等。如果某个组件出现错误单独调试即可。

调用的api的组合根据环境需要选择即可。这种开发方式利于后期扩展，若有需求需要改到这一层代码，尽量考虑模块独立性和可配置性，否则不要改动这一层代码。

FRONT文件夹中存放的是环境上层的内容，测试时主要关注的是FRONT文件夹。

## 2-FRONT文件夹



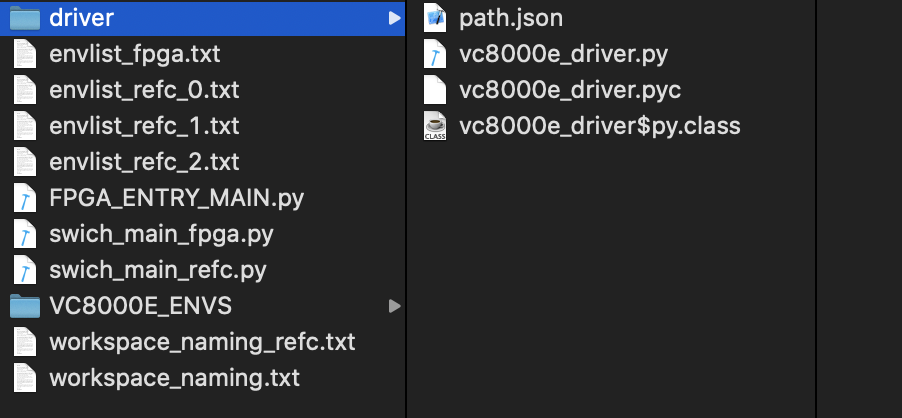
目前的FRONT文件夹中有这些内容。实际yuv和compare都是可移动的

yuv中的地址与环境中配置相关，因为目前testlist中（后期会介绍）配置的都是相对地址，若修改yuv目录在配置时需自行注意。

compare是对于结果解析的一个工具，该工具不是通用的，所以对自动化环境来说是一套完全独立的工具。（后续会介绍使用方法）

GROUP PKG就是整个环境的配置。

## 3-GROUP\_PKG



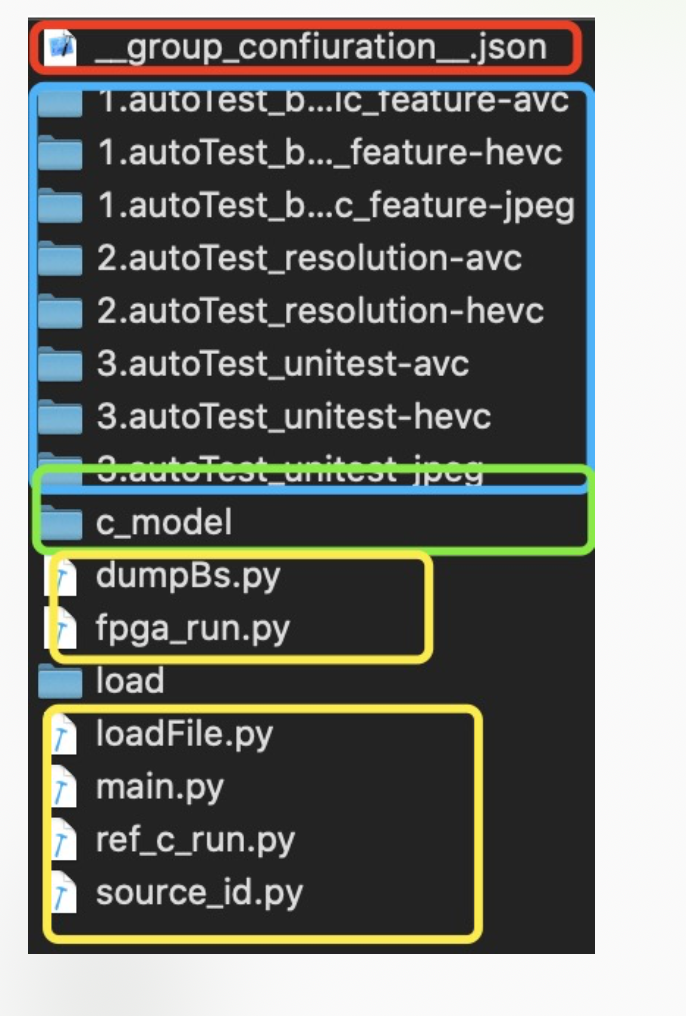
一个group中可以有多个ENVS（ENVS中可以包含多个env），每个ENVS需要有自己的driver。

driver目录下的path。json是为了配置当前driver如何找到Botton，在修改目录结构时需要修改，当在多平台使用时尽量使用相对地址，避免找不到出错。

该目录下driver和ENVS需要成套存在。目前只有vc8000e，对应与vc8000e\_driver和VC8000E\_ENVS

其余的txt都是配置文件，后面集中讲

## 4-ENVS



每个ENVS中的内容比较随意，都是根据自身ENVS中的配置决定。

比较关键的部分如上图框内所示：

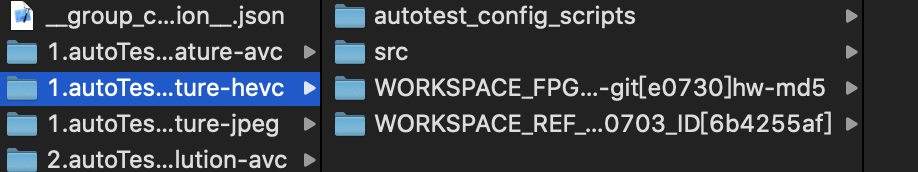
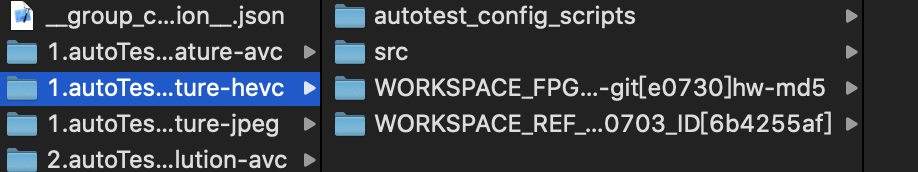
红框：configure json文件是整个ENVS的配置文件，每添加一个单独的env需要在里面注册该环境，否则运行时找不到该环境会报错。

蓝框：根据自身测试需要添加相应的环境，每个环境中执行的动作是相同的，对于不同测试需求可以开不同的env单独维护。（若只是对于一些基础配置不同，只需要改变环境内部的src，不需要添加环境）

绿框：需要在pc上运行时存放可执行文件的目录。

黄框：所有运行的py，核心是\*\_run.py

## 5-env



上图中为打开某个env中的内部结构。

Autotest config scripts中存放的是与当前env相关的配置。（环境的目标是只修改每个环境的配置达到自动化测试目的，这里的配置是整个env的核心，可以理解为是配置回调函数）

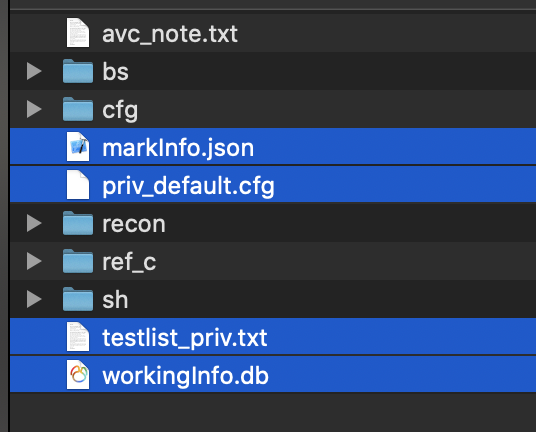
src中存放的是基础配置文件和testlsit，可以有多类。

## 6-workspace工作空间

每次开始运行某个env时，环境会根据配置打开相应的工作空间，整个运行结果全部在工作空间中存在。

工作空间的目的时方便调试，并且方便回归，多空间互不影响。

打开工作空间时，会check以下3个文件是否存在，若不存在会从src中拷贝testcase和default cfg。从groupconfiguration。json中把当前环境配置备份进来生成workinfo。json



如上图所示。

**workingInfo。db**

**testlist priv.txt**

**Workinfo。Json**

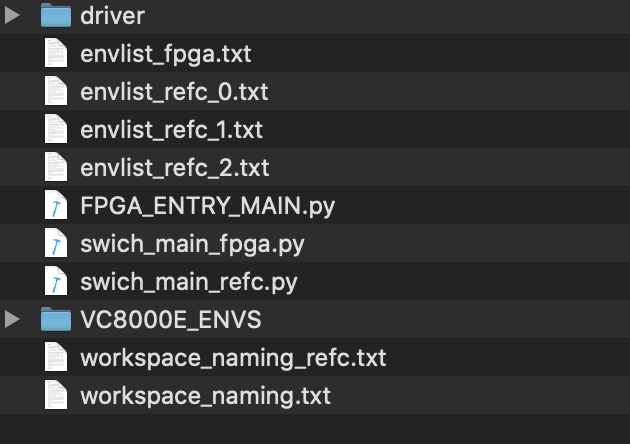
workingInfo。db每次更新环境会新生成一个，用于把当前测试进度持久化存储，类似与database的功能。通过修改workingInfo。db可以控制下个运行某种配置

# PC环境使用说明

具体使用的系统根据可执行文件决定。环境不区分系统，但是需要在运行的系统上安装python2.7

本章是当环境配置好之后，运行的步骤说明。对于不同环境，步骤可能有差异（差异的原因是某些功能可能不支持）

## VC8000E使用说明



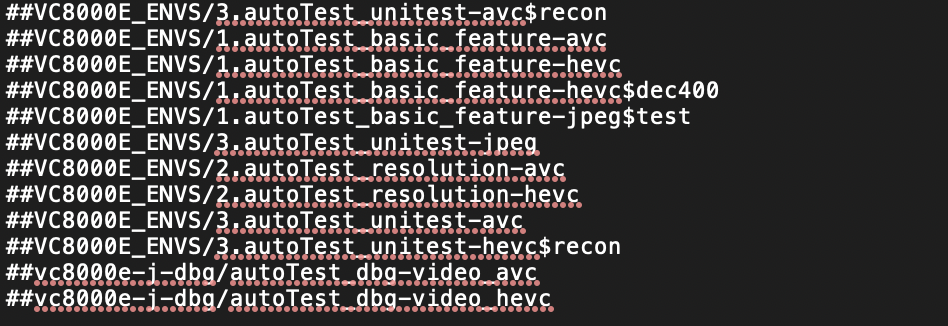
运行环境时只需要关注GROUP PKG这一层

1. 配置当前测试需要pass的所有环境

（目前vc8000e有fpga和pc-linux两个运行平台，当运行refc时需要配置c相关的txt）

**Envlist\_refc\_\*.txt**

如下图所示：



当环境的末尾添加$xxx时，表示使用的default配置文件和testlist用env名/src/xxx目录下的。（确保目录存在）

不添加时使用env名/src/default下的。

（testcase具体使用后续介绍）

1. 对当前运行环境的工作目录命名

**Worksapce\_naming\_refc.txt**

说明：

1）命名文件中只有第一行被识别

2）%Y 表示年

%m 表示月

%d 表示日

e.g.

%Y-%m-%d-wwx-test-new-version

真正生成的工作空间名字后面还有一串8字符长度的字符串，这串字符串为截取的md5字符串。Md5值来源于使用的可执行文件（包括可执行文件的固定配置，如vc8000e需要tb。cfg），该值用于区分c模型。避免更换c模型而未更换工作空间名导致的码流覆盖问题。也便于回归

1. 运行

在GROUP PKG目录下执行 switch\_main\_refc.py

默认为使用envlist\_refc\_0.txt作为当前执行的testcaselist。若要指定其他的，使用 -f，如 -f 1

指定运行线程数使用-p参数，如 -p 6，表示用6个线程运行，默认用4个

e.g.

python switch\_main\_refc.py -f 2 -p 6

# Testcase list说明

只要使用到test case manager模块的地方都需要配置Testcase list，该manager的作用就是维护当前testlist。

规则：

1. 无# 表示未开始运行
2. 1个# 表示正在运行
3. 2个# 表示运行完毕，不再运行
4. 3个# 表示case有错误，目前只检测size问题，不配置size的话默认1g

运行介绍：

manager会寻找下一个运行的case，首先打开正在运行的，若没有，则打开第一个未运行的case。

首次打开时，会在当前case前添加 #号，表示正在运行当前环境

执行结束时会在当前case前再添加 #号。之后不再打开该case