

TREGRESS 用户手册

by Tencent Penglai Lab

一、文档说明

本文档由腾讯蓬莱实验室开发用以说明回归工具 TREGRESS 的使用方法。

二、回归脚本使用说明

回归脚本是用来帮助验证进行回归测试的脚本代码，因为回归脚本和 makefile 保持一致，所以读者可以在每个项目的 sim 文件夹下面使用特定 makefile 命令就可以直接操作。如果需要开启特定的功能，需要读者自行配置自己所需要的 ini 配置文件（ini 文件例子在回归脚本所在文件夹下，用于拷贝修改）。

脚本包含 3 个文件：DoRegress.sh，DoRegress.py，regress.ini。DoRegress.py 是回归脚本主体，里面包含了回归脚本所有代码；DoRegress.sh shell 脚本，用来封装 DoRegress.py 以及所需要使用的额外命令；regress.ini 脚本配置文件（文档后称为 ini 文件），用来配置回归脚本。

每一次回归结束之后，会在 terminal 中显示回归结果，并且生成 log。

三、回归脚本使用要求

- 使用者只能修改自己所配置的 ini 文件，如果需要修改回归脚本，请联系对应项目脚本负责人进行修改；
- 使用者尽可能不要修改脚本所产生的 file 文件（cmd.file 等），在回归中删除或者修改该文件可能会导致脚本出现问题；
- 在使用脚本之前，需要确保 python 版本必须是在 3.7.5，如果不是，可能会导致库 import 错误；
- Python 的解释器存放位置必须为 /usr/bin/python3，如果没有的话，请使用 ln 将自己的 python3 解释器链接到这个位置；
- 读者需要安装如下库：paramiko、configparser、xlwt、xlrd、xlutils、requests、matplotlib、paramiko、hashlib、json；

四、使用回归脚本进行本地回归的基本步骤

1. 在与 sim 平级的 regress 文件夹下找到 ini 文件；
2. 拷贝 ini 文件并重命名为 xxx.ini，如 regress_sanity.ini；
3. 通过 vim 或者其他文件编辑器打开新的配置文件；
4. 在 tools 后面配置为

```
;url = {"http://git.code.oa.com/sharonsyu/V265_";
=====
tools = none
operation =
url =
none
```

5. 在 module_name 处配置验证模块的实例化名字，如：u_aaa；注：这里必须是和自己模块在 sv 里面配置的一样，否则会影响覆盖率收集；
6. 在 excel_name 处配置保存回归结果所使用 excel 名字；
7. 在 excel_path 处配置 excel 保存路径，该部分支持相对路径或者绝对路径；

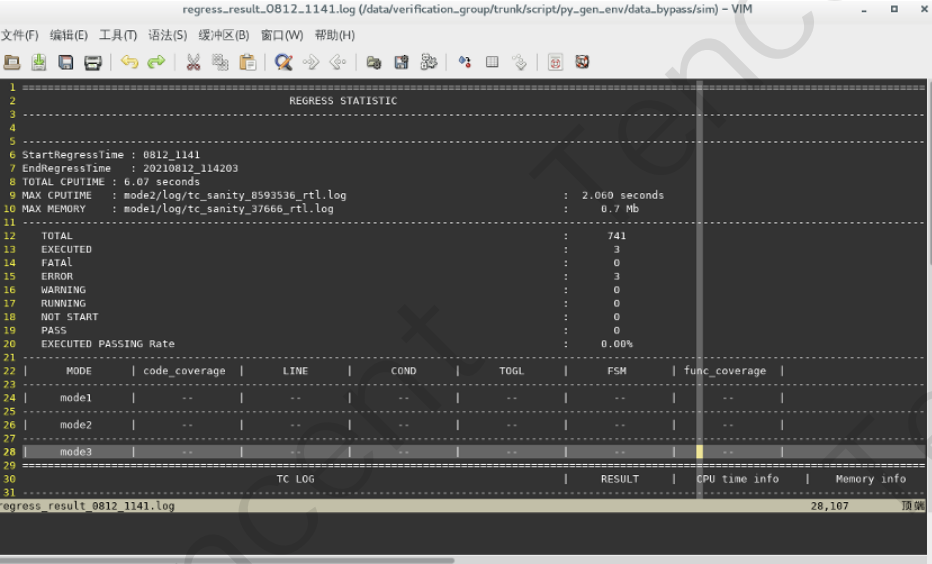
```
=====
; 回归报告选项包括:
; --- module_name : 指定当前模块在环境中的instance名称, 用于检索覆盖率, 注意, 必须是instance名, 例如: U_MD_TOP
; --- excel_name : 指定当前回归最终报告存放的excel名称, 例如: regress_result
; --- excel_path : 指定当前回归最终excel报告存放路径, 例如: /data/lhome/zifama/project/v265/regress
; =====e.g.=====
module_name = U_MD_TOP
excel_name = md_regress_result
excel_path = /data/lhome/zifama/project/v265/regress
=====
module_name = u_aaa
excel_name = aaa_regress
excel_path = ../script
```

8. 配置回归所使用的最大并行数 paral_run_num；
9. 在 tc_list 处，填写自己所需要回归的 tc 信息，用 {} 包含，用 ; 区分两条 tc；配置方式如下图所示，ini 文件配置完毕
10. 到项目的 sim 文件夹下；
11. 执行 tregress xxx.ini 例如：tregress regress.ini；
12. 等待回归结果，可以在 terminal 中看到回归结果

```
-----
REGRESS STATISTIC
-----
StartRegressTime : 0812_1141
EndRegressTime   : 20210812_114203
TOTAL CPU TIME   : 6.07 seconds
MAX CPU TIME     : mode2/log/tc_sanit_8593536_rtl.log : 2.060 seconds
MAX MEMORY       : mode1/log/tc_sanit_37666_rtl.log   : 0.7 Mb
-----
TOTAL           : 741
EXECUTED        : 3
FATAL           : 0
ERROR           : 3
WARNING         : 0
RUNNING         : 0
NOT START       : 0
PASS            : 0
EXECUTED PASSING Rate : 0.00%
-----
MODE | code_coverage | LINE | COND | TOGL | FSM | func_coverage |
-----
mode1 | -- | -- | -- | -- | -- | -- |
mode2 | -- | -- | -- | -- | -- | -- |
mode3 | -- | -- | -- | -- | -- | -- |
-----
More information in /data/verification_group/trunk/script/py_gen_env/data_bypass/sim/regress_result_0812_1141.log
```

也可以在 log 里

面查看结果



五、回归脚本的其他相关可配置功能

所有的额外功能都是在基本使用步骤的基础上来添加的。以下介绍按照 regress.ini 文件的顺序，从上往下介绍。

1. 从 git 上更新（pull）代码后回归

在 ini 文件里面配置 tools 为 git，operation 为 update，url 为 git 仓库所对应的 url 与分支；

2. 从 git 上拉取（clone）新代码回归

在 ini 文件里面配置 tools 为 git，operation 为 clone，url 为 git 仓库所对应的 url 与分支，clone_workspace 为需要 clone 新的文件夹，sim_path 为本地 sim 目录的前一级；注：clone_workspace 和 sim_path 必须为绝对地址，clone 会在 clone_workspace 创建一个名为<年月日_时分秒>的文件夹

3. 从 svn 上更新（pull）代码后回归

在 ini 文件里面配置 tools 为 svn，operation 为 update，url 为 svn 仓库所对应的 url；

4. 从 svn 上拉取（co）新代码回归

在 ini 文件里面配置 tools 为 svn，operation 为 co，url 为 svn 仓库所对应的 url 与分支，clone_workspace 为需要 clone 新的文件夹，sim_path 为本地 sim 目录的前一级；注：clone_workspace 和 sim_path 必须为绝对地址，clone 会在 clone_workspace 创建一个名为<年月日_时分秒>的文件夹

5. 输出结果保存为 excel

每次回归的结果会保存在 excel 表格中，如果是在同一天内产生的结果就会覆盖，如果是不同天的结果会加到后面，excel 的包含的信息。

如果打开了覆盖率，并且有多种模式，那么会把每种模式的覆盖率都会列出；

注：module_name 必须要和实例化名字相同，否则会找不到覆盖率文件，最后会无法显示覆盖率；

6. 并行执行

可以通过配置 paral_run_num 来配置并行进程数量，数量必须大于1；注：该进程需要读者根据自己的服务器内存和 license 数量进行配置；

7. 错误case 重新仿真

ini 配置 fail_rerun 为 enable，根据自己的需求添加 rerun_opts，如：在发现有错误之后，重新仿真时，读者需要打开波形，那么可以进行如下配置：

```

70 ;=====
71 fail_recmp = disable
72 recmp_opts =
73 fail_rerun = enable
74 rerun_opts = wave=fsdb
75
76

```

8. 错误case 重新编译后仿真

ini 配置 fail_recmp 为 enable，fail_rerun 为 enable，根据自己的需求添加 recmp_opts 和 rerun_opts，脚本会根据 tc 名称产生新的 simv 文件，文件名称为 tc 名称_simv，如：在发现有错误 log 之后，我想关掉对应覆盖率，然后进行仿真，那么可以进行如下配置：

注：fail_recmp 必须在 fail_rerun 打开后有效（默认不会进行只编译不仿真）

```

;rerun_opts = note=error wave=fsdb
;=====
fail_recmp = enable
recmp_opts = fcov=off ccov=off
fail_rerun = enable
rerun_opts = wave=fsdb

```

9. 结果不统计 Warning 数量

ini 配置 ignore_warning_en 为 enable，结果中只含有 warning 的 case 会认为通过；

```

103 ;=====
104 ignore_warning_en = enable
105 fatal_key_words = []

```

10. 添加新的 fatal/error/warning 关键词

添加新的关键词，如果 log 中出现 fatal/error/warning 关键词，也会认为该 case 为 fatal/error/warning；ini 中配置 fatal/error/warning_key_words list，用“,”区分关键词，关键词需要添加引号；注：key_words 的格式可以参考 python 的 list 格式。

```

105 ;=====
106 fatal_key_words = []
107 error_key_words = ['error', 'ERROR']
108 warning_key_words = []
109

```

11. 忽略 fatal/error/warning 关键词

当 log 中如果出现关键词，并且出现关键词的关联词，那么判断为 fatal/error/warning；ini 中配置 ignore_fatal/error/warning_key_words = {'关键词 a':['关联词 a1','关联词 a2',...,'关联词 an'], '关键词 b':['关联词 b1', '关联词 b2']}

例如：

```

109 ;=====
110 ignore_fatal_key_words = {}
111 ignore_error_key_words = {'tc_error':['UVM_INFO', 'Command']}
112 ignore_warning_key_words = {}
113

```

12. 忽略特定 log

配置不需要统计结果的 log。ini 中配置 total_ignore_log_list = ['log1 名称','log2 名称']；注：log 名称可以使用通配符进行设置；该功能使用于 make get_total（功能 20）

例如：

```

113 ;=====
114 total_ignore_log_list = ['vcs_compile_rtl.log', 'git.log', '*.log.swp', '*.log.swo']
115

```

13. 在脚本运行之前，进行操作

配置 pre_script/make_enable 为 enable，配置 pre_script/make 为所需要命令，例如：


```
57 ;=====
58 ;配置回归编译选项：
59 ;----docompile：回归开始时是否
60 ;----fcov：回归时是否打开功能覆
61 ;----ccov：回归时是否打开代码覆
62 ;----udf：回 归编译时需要添加的
63 ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>
64 ;docompile = disable
65 ;fcov = off
66 ;ccov = off
67 ;udf = "+define+AAA"
68 ;=====
69 docompile = enable
70 fcov = on
71 ccov = on
72 udf = ""
73 cmp_opts =
74
75 ;=====
76 ;----dorotation: run tc with ro
```

20. 覆盖率开关

可以选择本次回归是否收集覆盖率，fcov 为功能覆盖率，ccov 为代码覆盖率，ini 配置 fcov/ccov=on/off;

21. Udf 配置选项

添加 vcs 编译选项、ini 配置 udf“配置1 配置2”，如：

```
6 ;=====
7 ;配置回归编译选项：
8 ;----docompile : 回归开始时是否执行vcs命令，此选项默认为enable。
9 ;----fcov : 回归时是否打开功能覆盖率，此选项默认为off。
0 ;----ccov : 回归时是否打开代码覆盖率，此选项默认为on。
1 ;----udf : 回归编译时需要添加的编译选项，如+define+AAA等。
2 ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
3 ;docompile = disable
4 ;fcov = off
5 ;ccov = off
6 ;udf = "+define+AAA"
7 ;=====
8 docompile = enable
9 fcov = on
0 ccov = on
1 udf = "+define+AAA"
2 cmp_opts =
```

22. 补充编译选项

补充编译时，除覆盖率、udf 之外的编译选项，ini 配置 `cmp_opts` = “配置1 配置2”如：

[illegible]

23. Tc 轮询测试

当配置多条 case 并且每一条回归的次数都大于1，可以配置dorotation 为 enable，修改回归顺序——从顺序回归，变成轮询每条 case 回归：

```
>=====
;tc_list = [
;    {'RUN_TIMES':10,'TC':'tc_sanity'
;;    {'RUN_TIMES':100,'TC':'tc_full',
;    {'RUN_TIMES':1,'TC':'tc_directed
;    ]
;=====
dorotation = enable
tc_list = [
]
```

将错误的回归结果汇总成新的 ini 文件，修改后可以直接进行简易回归，错误 ini 名称为 error_月日-时分.ini；如果需要使用该 ini 文件来进行回归，可以输入命令 `make do_regr regr_ini=error_月日-时分`；默认配置如下：

25. ini 文件缺失补充

26. 中断继续运行

当回归因为系统原因中断，会在 script 目录下产生 RemainTc.ini 文件，里面包含了之前 ini 文件的基础配置，并且将还没有运行的 tc 汇总在一起。当修复好系统之后，可以直接使用 make do_regr regr_ini=RemainTc.ini 继续仿真

27. python 库自动安装

如果当前版本 python 没有安装脚本所需要使用的库，脚本会先通过使用 pip 安装新的库，不过建议还是先安装所有的库会比较好。

28. 脚本在出现关键词之后再开始检查 warning、error、fatal

当初 log 里面出现关键词的时候，再开始检查 fatal\error\warning，如果没有填任何关键词，那么 log 会从头开始检查

```
receiver_list = []

#start_chk_fatal_key_word: log从该关键词之后，开始检查fatal，如果没有输入，则表示从头开始检查，默认为空
#start_chk_error_key_word: log从该关键词之后，开始检查error，如果没有输入，则表示从头开始检查，默认为空
#start_chk_warning_key_word: log从该关键词之后，开始检查warning，如果没有输入，则表示从头开始检查，默认为空
#####e.g.#####

start_chk_fatal_key_word = 
start_chk_error_key_word = start resetting
start_chk_warning_key_word = 

start_chk_fatal_key word = 
start_chk_error_key word = 
start_chk_warning_key word =
```

29. 回归结果不统计仿直错误的覆盖率

回归结果不统计仿真错误的覆盖率。配置开关 `del_err_cov`，默认为 `disable`

```
206 ;           {'RUN_TIMES':1, 'TC':tc}
207 ;       ]
208 ;=====
209 dorotation = disable
210 del_pass_log = enable
211 del_err_cov = disable
212 tc_list = [
213     1
```

30. 回归结束删除所有正确log

回归结束后删除所有正确的log。配置开关del pass log，默认为enable

```
206 ; { 'RUN_TIMES':1, 'TC':tc }
207 ; ]
208 ; =====
209 dorotation = disable
210 del_pass_log = enable
211 del_err_cov = disable
212 tc_list = [
213     1
```

31. 自动查找 ini 配置文件

