使用 DT10 进行覆盖率测试

CASIO　计算机 QV 事业部　　开发部

单元测试

以前

- 各个模块的特性相结合，然后实施独立的单元测试。

  驱动层（Driver Layer）：单一功能测试等

  中间层（Middle Layer）：整合功能测试、数据流测试等

  应用层（Application Layer）：状态迁移测试（state transition testing）等
- 每个模块测试内容各异，各自被测试到何种程度，不完全一样。
- 交给测试者评价的部分，应该是有的，但并不确定。

没有被执行到的部分被遗留下来，也没有做验证，所以没办法上市。

希望单元测试可以成为一个必要项目，然后获得覆盖率，并执行测试

- 希望不要造成开发人员太多负担，从导入到运用都能很简单地进行覆盖率获取和测试。
- 希望可以进行非常接近产品实际动作状态的测试，并获取覆盖率。

DT-10

ET 2010 因为对"灰盒测试"的关键词很感兴趣

2011/04 免费借用，并考虑导入

2011/07 租用半年，室内试用

TRINITY TECHNOLOGIES
创提信息科技（上海）有限公司
www.trinitytec.com.cn

地址：上海市浦东新区张江高科张衡路 200 号 3 栋 3207E 室
成都市高新区盛和一路 88 号康普雷斯 B 座 1914 室
电话：021-51328530（上海办公室）
028-83397325 （成都办公室）
邮箱：info@trinitytec.net

DT10 覆盖率测试方法

使用 DT-10 进行测试并获取覆盖率

＜对象＞

将在黑盒测试里面不容易确认的模块作为测试对象

＜方式和时期＞

新的内容：例外或异常的状况下，全部的路径的覆盖率做测量，希望能达到的目标为 C0 100%

＜更新＞

源代码的有限差分（finite difference）例外或异常以外的全部的路径的覆盖率做测量，希望能达到的目标为 C0 100%。

至少，到产品完成之前希望能够完成。

＜Output＞

希望可以将覆盖率数据和报告可以透过 CSV 输出，然后提交。

＜期待的效果＞

被要求应有动作的地方，则判断为正常。

没有动作的话，则被判断为异常。也有可能可以发现完全不动作的源代码。

＜结果＞

确认了覆盖率取得、确认了测试效果（对于作业流程负荷的），也可以通过覆盖率的信息来决定要不要继续进行测试。

地址：上海市浦东新区张江高科张衡路 200 号 3 栋 3207E 室
成都市高新区盛和一路 88 号康普雷斯 B 座 1914 室
电话：021-51328530（上海办公室）
028-83397325 （成都办公室）
邮箱：info@trinitytec.net

覆盖率的获取和测试流程

1. 在 DT10(+DTDiff)，在源代码里放测试点
2. DT10 中建一个 profile
3. 在放了测试点的源代码中建立

   加上这个模块，并换上以下 2 个模块

   - 保存测试的和日志资料，文件输出模块
   - 保存要用的内存映射模块

   (★透过使用内部存储器，来算出覆盖率数据。没有使用动态追踪程序★)
4. 使用在步骤 3 中所做的程序，并操作相机
   - 第一次的电源键中，开始进行测试点，日志的输出
   - 第二次的电源键中，被保存的测试的日志，文件输出到 SD 卡里
5. 通过 DT 转换器，对 SD 上的日志文件进行更换
6. 更换过的数据，通过 DT10 来读取，并显示覆盖率报表的数据
7. 确认没有被执行到的测试点。如果有需要的话，从第 4 步开始重来
   - 明明有测试点的地方但却没有办法进行操作。是异常情况吗？
   - 是因为那个测试的没有在使用吗？或是其实是问题点？
8. 输出覆盖率报表，然后生成 CSV 文件

地址：上海市浦东新区张江高科张衡路 200 号 3 栋 3207E 室
成都市高新区盛和一路 88 号康普雷斯 B 座 1914 室
电话：021-51328530（上海办公室）
028-83397325 （成都办公室）
邮箱：info@trinitytec.net

覆盖率报告

カバレッジレポート：Prof_110729_171828

関数	ソース	モ..	TP設定数	未使用	通過禁..	非通過	無効TP	有効..	通過(正常)	通過(異常)	C0	C0(有効)
fIMCVMI_CloseFace	imcvm_face.c	1_..	3	0	0	0	0	3	2	0	66.67%	66.67%
fIMCVMI_ClearFaceInfoThru	imcvm_face.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_FaceDetectThru	imcvm_face.c	1_..	3	0	0	0	0	3	2	0	66.67%	66.67%
fIMCVMI_WriteFace	imcvm_face.c	1_..	5	0	0	0	0	5	2	0	40.00%	40.00%
fIMCVMI_MakeThru_Face	imcvm_face.c	1_..	11	0	0	0	0	11	10	0	90.91%	90.91%
fIMCVMI_MakeThru_MyseH	imcvm_face.c	1_..	16	0	0	0	0	16	0	0	0.00%	0.00%
fIMCVMI_FaceDetect	imcvm_face.c	1_..	5	0	0	0	0	5	0	0	0.00%	0.00%
fIMCVMI_GetAFInf	imcvm_moth.c	1_..	6	0	0	0	0	6	6	0	100.00%	100.00%
fIMCVMI_SetRoiDetecting	imcvm_moth.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_GetRoiDetecting	imcvm_moth.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_GetRoiExisting	imcvm_moth.c	1_..	3	0	0	0	0	3	3	0	100.00%	100.00%
fIMCVMI_Callback_MakeThruMoth	imcvm_moth.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_TaskMain_Moth	imcvm_moth.c	1_..	10	0	0	0	0	10	9	0	90.00%	90.00%
fIMCVMI_OpenRoi	imcvm_moth.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_CloseRoi	imcvm_moth.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_ExeRoi	imcvm_moth.c	1_..	4	0	0	0	0	4	3	0	75.00%	75.00%
fIMCVMI_SetSceneDetecting	imcvm_moth.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_GetSceneDetecting	imcvm_moth.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_TaskMain_SceneBst	imcvm_moth.c	1_..	5	0	0	0	0	5	5	0	100.00%	100.00%
fIMCVMI_ExeScene	imcvm_moth.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_MakeThru_IAF	imcvm_moth.c	1_..	23	0	0	0	0	23	0	0	0.00%	0.00%
fIMCVMI_MakeThru_PA	imcvm_moth.c	1_..	31	0	0	0	0	31	29	0	93.55%	93.55%
fIMCVMI_SetHdrExecuting	imcvm_moth_hdr.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_GetHdrExecuting	imcvm_moth_hdr.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_ExeHDR	imcvm_moth_hdr.c	1_..	4	0	0	0	0	4	3	0	75.00%	75.00%
fIMCVMI_MovieSetStreamEncode	imcvm_movie.c	1_..	2	0	0	0	0	2	1	0	50.00%	50.00%
fIMCVMI_MovieSetStreamDecode	imcvm_movie.c	1_..	2	0	0	0	0	2	1	0	50.00%	50.00%
fIMCVMI_MovieSetStream	imcvm_movie.c	1_..	9	0	0	0	0	9	8	0	88.89%	88.89%
fIMCVMI_MovieStopStream	imcvm_movie.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_MovieStopStream	imcvm_movie.c	1_..	4	0	0	0	0	4	3	0	75.00%	75.00%
fIMCVMI_MovieSetSkipMode	imcvm_movie.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_MovieSetSkipMode	imcvm_movie.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_SetDecodeBank	imcvm_movie.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_GetDecodeBank	imcvm_movie.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_Callback_MovieDecode	imcvm_movie.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_Callback_MovieResize	imcvm_movie.c	1_..	1	0	0	0	0	1	1	0	100.00%	100.00%
fIMCVMI_MovieYuvToH264	imcvm_movie.c	1_..	1	0	0	0	0	1	0	0	0.00%	0.00%
fIMCVMI_MovieYuvToH264	imcvm_movie.c	1_..	13	0	0	0	0	13	0	0	0.00%	0.00%
fIMCVMI_MovieH264ToYuv	imcvm_movie.c	1_..	13	0	0	0	0	13	8	0	61.54%	61.54%
fIMCVMI_MovieH264ToYuvPreview	imcvm_movie.c	1_..	6	0	0	0	0	6	2	0	33.33%	33.33%
fIMCVMI_MovieH264ToYuv	imcvm_movie.c	1_..	22	0	0	0	0	22	11	0	50.00%	50.00%
fIMCVMI_MovieH264ToYuv_PreDecode_Rot	imcvm_movie.c	1_..	24	0	0	0	0	24	18	0	75.00%	75.00%
fIMCVMI_MovieH264ToYuv_PreDecode	imcvm_movie.c	1_..	29	0	0	0	0	29	21	0	72.41%	72.41%
fIMCVMI_MovieYuvToH264_GetParam	imcvm_movie.c	1_..	21	0	0	0	0	21	12	0	57.14%	57.14%
fIMCVMI_MovieYuvToH264_SetParam	imcvm_movie.c	1_..	12	0	0	0	0	12	8	0	66.67%	66.67%
fIMCVMI_GetErrPara	imcvm_movie.c	1_..	2	0	0	0	0	2	2	0	100.00%	100.00%
fIMCVMI_SetIMAGEINFO_YUV_movie	imcvm_movie.c	1_..	3	0	0	0	0	3	3	0	100.00%	100.00%

TRINITY TECHNOLOGIES
创提信息科技（上海）有限公司
www.trinitytec.com.cn

地址：上海市浦东新区张江高科张衡路 200 号 3 栋 3207E 室
成都市高新区盛和一路 88 号康普雷斯 B 座 1914 室
电话：021-51328530（上海办公室）
028-83397325 （成都办公室）
邮箱：info@trinitytec.net

DT10 覆盖率测试报告

模块		测试结果		
	是什么样的测试	测试时间	问题数量	是否有效
art_slide_read_bullet1	全面的	2 小时	0	整体来说有效
bimtd	全面的	8 小时	0	有效
dspd_xxxxx	部分的			
facem	全面的	2 小时	9	整体来说有效
file_drvxxxxx	全面的	40 小时	4	有效
flem_xxxxx	全面的			
fmvd	全面的	2 小时	0	整体来说有效
lcd_mng_xxxxx	部分的			
main_main_xxxxx.a	全面的	3 小时	0	有效
mem_mng_xxxxx	部分的	20 小时	3	有效
mltpm_sel	全面的	1 小时	1	整体来说有效
plym_xxxxx	部分的	21 小时	0	整体来说有效
pwc_xxxxx	全面的	1 小时	0	整体来说有效
pwc_rtc	全面的			
recm_xxxxx	部分的（新增部分）	6 小时	1	有效
second_loader_b1_l	全面的	40 小时	4	有效
tmrd_all_bullet1	全面的	8 小时	0	有效
trcd	全面的	2 小时	0	整体来说有效

DT10 覆盖率测试报告

- 对象为 61 个模块中的 43 个模块，大约是 70%
- 覆盖率测试有效果吗？可以看到报告里面显示大致为「有效」、「整体而言有效」
- 发现的问题点：27 个分支判断错误，跟顶层模块的 I/F 不匹配，完全没有动作的源代码
- 其他
  - 要测试到全部的路径很困难
  - 非正常系统点（Abnormal system point）的设定很困难
  - 通常运作下不能确认的部分，现在可以确认了
  - 可以发现很多不需要的源码，能明确哪些功能没有被使用到
  - 发现了藏在分支里面的问题
  - 发现单元测试中，不是所有的分支都能被检查到
  - 产生测试用例很辛苦，但覆盖率检查比较轻松
  - 条件编译（Conditional Compile）是有效的
  - 多报告合并分析（Combined analysis of multiple reports）中可以更动测试点，这点很好

地址：上海市浦东新区张江高科张衡路 200 号 3 栋 3207E 室
成都市高新区盛和一路 88 号康普雷斯 B 座 1914 室
电话：021-51328530（上海办公室）
028-83397325 （成都办公室）
邮箱：info@trinitytec.net