TRINITY 创提信息

DT10 —嵌入式系统动态测试及调试工具

30天
不间断测试

分 析

掌握软件程序的实时执行过程

现在有越来越多、复杂且规模大的嵌入式软件，但开发的时间却相对越来越短。在这样的背景之下，为达成高质量及高效益的目标，建立在各种嵌入式平台基础上的二次开发，或重用之前代码模块中的源码来完成开发，便成为理所当然的开发方式。但是，开发者改动的部分会让系统整体变成什么样子？整个处理时间又会花上多久？这些问题也因此变得很难掌控。DT10会利用对先期开发基础源码的分析，事先做到掌握系统整体运行是否正确。

验 证

通过灰盒测试实现完整且有效的测试

DT10是为数不多能实现灰盒测试的动态测试工具。不只软件内部的测试，连同CPU周边的传感器、端口等硬件状态也都会和执行路径一起被验证，所以可以在同一个时间测试软件和硬件的整个运行过程。测试结果会自动产生报告，开发者和QA的工作人员都能通过使用该工具提升产品质量，使得工作更为高效。

计 划

可输入运行时间、周期时间和变量的设计值，做成一个“测试计划”

在DT10软件的界面上，可以输入运行时间、周期时间、参数、变量的设计值。除了输入的设计值可以在同一个页面（一览表）做管理之外，还可以当作设计书被导出。这些设计值，将在测试过程中被使用，可以和目标板上实际测量值做对照，如出现与设计值有出入的状况，系统会有警告的讯息弹出，便可通过这样的警告掌握问题的状况。

修 正

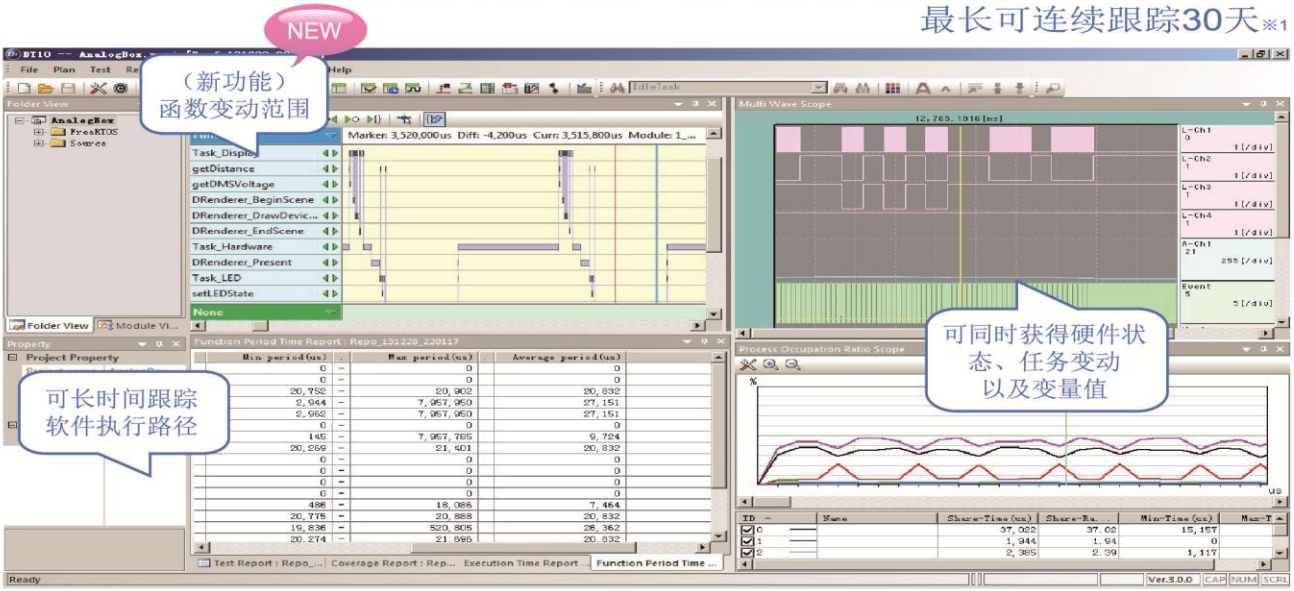
通过长时间不间断地跟踪调试，可以让目标板上执行的过程透明化

最适用于在不停止软件执行的情况下进行跟踪调试。因为可以取得并分析：任务的变化、函数的周期时间、运行时间、覆盖率、变量值的变化，所以整个调试的效率会大幅提升。就连不容易重现的问题，只要能重现1次，也可以通过跟踪数据分析原因。所以通过长时间运行测试及修正是非常高效的方式。

长时间跟踪分析软件执行过程

监视软件执行状态，回溯定位复杂Bug

最长可连续跟踪30天※1

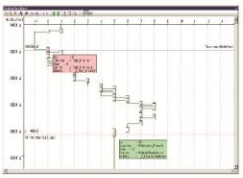


通过函数变动范围可进行快速分析！

根据时间轴，可以看到各个函数到底是在哪个时间点被执行的。再加上测试点和multi wave scope也都是同步输出，可以大幅提升调试的效率。

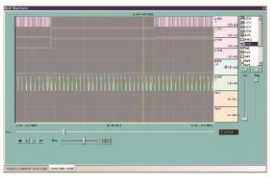
通过函数跟踪功能，可透视执行路径

函数内部处理和中断状况发生时的相关详细执行路径，都可以借此图像显示。同时也可以比对源码，是一个既便利又具有直观性操作的功能。



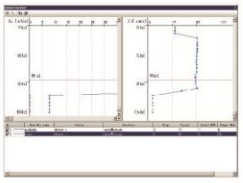
获得硬件的状态（如：电压、电流）

可以把DT10当作示波器来使用，这样可以检测电压和逻辑。另外，变量值和任务变动也会在同一页面显示，因此可以通过软硬件的跟踪结果，来确认软硬件配合是否有问题。对采用传感器设备的开发人员来说，这是一个相当有帮助的功能。



长时间追踪变量值的变化！

因为插入了用于导出变量值的测试点，所以会随着时间轴的变化，观察到变量值产生变化。可将多数的变量图像合并查看，也可以在同一个页面（一览表）显示各个变量的最大值和最小值。



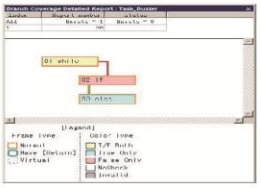
※1：还是取决于连接DT10的计算机的存储容量。

实时获取测试覆盖率

度量和分析目标板上测试的完整性

获取集成/系统测试的覆盖率

DT10获取覆盖率的方法很简单。只需要在目标板上执行即可；可以得到语句覆盖率、分支和MC/DC覆盖率。对于集成测试和系统测试，都可以从用户角度进行常规测试即可获取覆盖率数据。

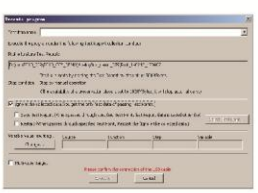


得到实时覆盖率数据

目标板上执行和获取报告时，不需要刻意停下来，可以一边在目标板上照常执行，一边收集覆盖率结果；覆盖率数据可以按照百分比不同，显示出不同的颜色进行区分。所以可以很快掌握未测试的程序代码，让测试更完整。

长时间收集覆盖率数据

如果要针对系统测试进行跟踪，很容易会有跟踪资料（trace data）太多而造成后续分析很花时间的问题。DT10有“一次跟踪”[one time trace]的功能，这个功能会让跟踪数据限制在一定的大小之内(不让跟踪数据无限扩张/膨胀)。就算是在执行系统测试时，也能有效地获取覆盖率资料。



在极小的目标环境上获取覆盖率

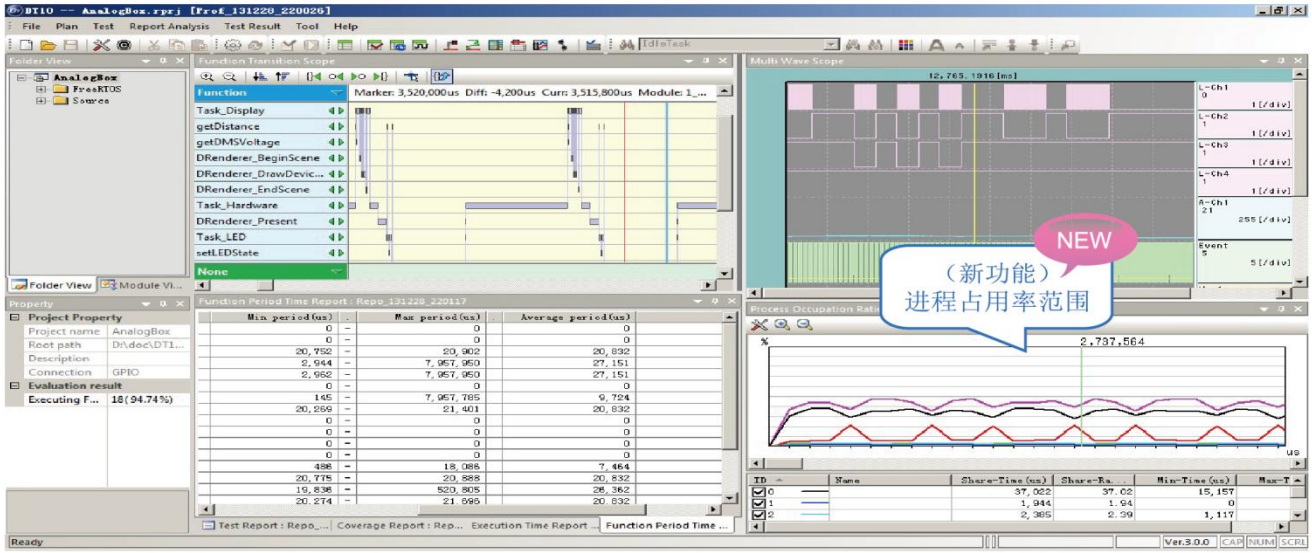
业界测试工具大多无法在极小资源的目标环境（如单片机）上进行测试并获取覆盖率，而DT10却完全可以做到！

多样的DT10使用案例

一提到获取覆盖率数据，最常想到的是单元测试。除了单元测试外，需要得到集成测试、系统测试覆盖率的开发团队越来越多。因为通过DT10确实能满足客户避免测试不完整的需求，许多开发部门都在使用。

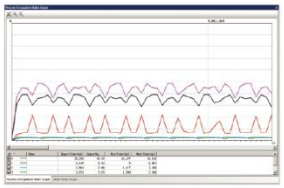
系统性能测试和优化

统计“运行时间”和“任务占用率”，优化性能瓶颈



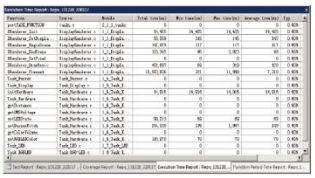
统计“进程占用率”

任务占用率可以通过图像显示，一目了然地看到任务中哪个部分负载较大。因为在那个时间点，可以直接定位到实际处理位置，所以可知道到底是哪个执行路径发生问题，从而很方便地了解问题状况。



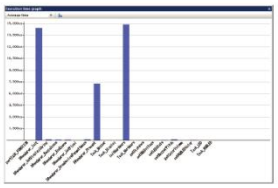
分析“运行时间”和“周期时间”

所有函数的运行时间（最大、最小、平均时间等）都可在统计后显示出来。另外，也可以得到任意两点之间的“处理时间”报告，非常方便地发现跟现有设计值存在差异的地方。



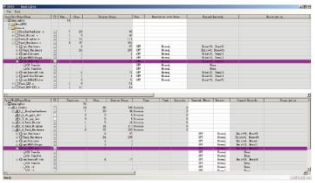
自动比较实际值和设计值

可预先设置期望的“运行时间”和“变量值”。在目标板上实际执行后，可将“测试值”和“设计值”做比较。不仅可以很方便地发现程序代码中发生异常的地方；如果后续该项目开发下一个版本，还可以重复使用这些设计值，从而提高软件开发的效率。



透视“处理时间的波动”

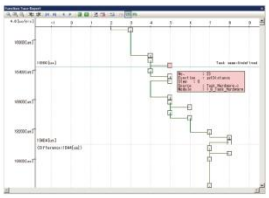
如果使用“运行时间的长条图”，则可发现实际的处理时间与设计值是否有差异。另外，也可以看到目标板上处理时间的变化（波动）；因为能知道异常的处理时间，所以可提前发现/修正“与时间相关的潜在问题”。



其它有用的功能

分析程序状态和逻辑

可显示程序的状态迁移和顺序变化的情况。

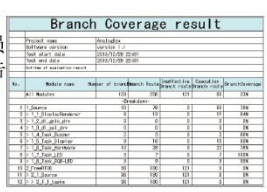


测量CPU的负荷

分析CPU的负荷测量数据，并且用图形化的方式，显示随时间轴变化的CPU负荷情况。因为目标板运行的情况下，可以显示实时的状况，所以可以检测任务的优先级，调度是否安排合理。

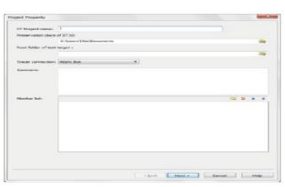
测试报告自动生成

运行时间和覆盖率测量的结果，可以自动生成报告。对开发人员来说，不用花时间在做测试报告上，而可以更专注于开发。



基于变更的插装方式

可以将测试点只插入在与前一个版本不同之处。如果和覆盖率测试相结合，效果将会倍增。有版本更新的状况下，可只测试更新的程序代码，使测试的程序代码量降到最低，这样会是最有效率的测试方式。

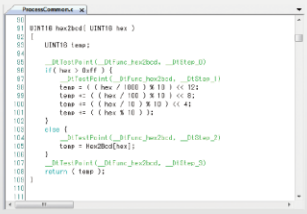


使用方法

DT10并不只是一个跟踪调试（trace debug）的工具。它可以测试目标板上实时运行的程序，可以连续跟踪记录长达30天，同时还提供很多简单实用的功能，用于分析这些跟踪资料（trace data）。让开发及测试人员有更多功能和数据来优化大规模、复杂的嵌入式软件，在提高开发效率的同时，也能完成质量提升的目标。

简单的三个步骤

插装

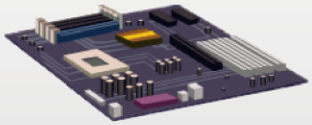


自动对源码插入测试点

编译构建

利用程序原开发编译环境

目标板上执行



捕获实时数据传送到PC端软件

支持C/C++/C#/Java代码

可针对函数的入口处/出口处/分支自动插入测试点，也可以手动插入代码到任何位置；不依赖CPU和OS类型。

连接方式

6种连接方式，多样化选择。

- 异步总线连接
  ※使用NOR型flash memory的总线来连接。
- SD card I/F 连接
  ※使用SD card/micro card slot来连接。
- GPIO/SPI连接
  ※根据空闲的埠数不同，可以自行选择4 bit parallel/3线式serial。
- Ethernet连接
  ※可以通过UDP、TCP/IP(Server, Client)来连接。
- CAN连接
- UART连接

连接方式

※如果无法按照上述方式做连接的话，不用通过dynamic tracer。
可直接从目标板如USB memory、硬盘等的跟踪数据存储器（trace storage）将执行路径的数据发送出去。

公司：创提信息科技（上海）有限公司
地址：上海市浦东新区张衡路200号3幢3207E室
电话：021-51328530 / 传真：021-51328526
Email：info@trinitytec.net
网址：www.trinitytec.com.cn

TRINITY 创提信息