

DT10 功能介绍

DT10 多波示波器

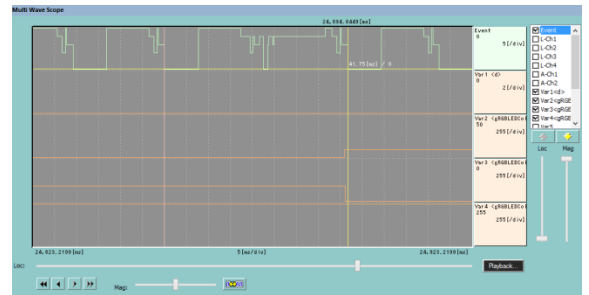
功能介绍

有些嵌入式软件方面的问题，利用传统的调试器可能无法解决，而通过逻辑分析器则能有效地解决。请仔细阅读本文，看我们如何一步一步地讲解在这种情况下所需的配置：

[Troubleshooting-real-time-software-issues-using-a-logic-analyzer](#)

但是，从传统意义上讲，逻辑分析器是硬件工程师用来调试数字电路的。从上面的参考文件中可以清楚地了解到，嵌入式软件工程师在使用分析器之前，要做很多准备工作，因为该工具不是一个专门用来解决问题的软件。

DT10 的**多波示波器+模拟盒**是专门为解决嵌入式软件的问题而研发的，它的功能与逻辑分析器或示波器的功能很相似。它可以捕获并显示运行时软件的特征，并在示波器上实时显示不断更新的精确时间图表。



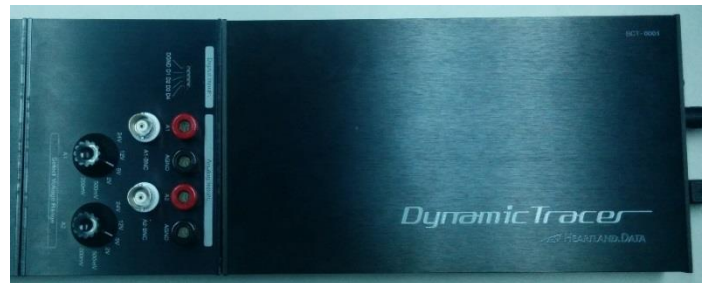
DT10 的一些功能能够帮助嵌入式工程师自动完成很多工作，包括：

- ✓ 可以在源代码中自动插入测试点（不需要复杂的代码或脚本）
- ✓ 能够与用户现有的主机通信机制（GPIO, UART, Ethernet 等）进行无缝结合
- ✓ 可获取并显示内核事件的踪迹（如：任务调度器，中断程序，等）
- ✓ 最多可同时获取并实时显示 20 个局部或全局变量
- ✓ 最长可进行长达 32 天的持续执行追踪（针对难以重现的问题）
- ✓ 有离线回溯调试的功能，可回顾实际执行中出现的变量转换情况
- ✓ 示波器数据点和代码的实时执行路径踪迹之间,可通过鼠标单击进行双向定位
- ✓ 可自动生成一系列分析报告，帮助发现并解决运行时出现的问题

DT10 信号模拟盒

除了获取实时的源代码特征之外，DT10 模拟盒组件还能够**从目标设备上获取逻辑和模拟信号**。动态追踪器会同时同步硬件和软件的数据，并将这些数据通过同一个时间图表实时显示在多波示波器上。

将 DT10 的动态追踪器和模拟盒结合起来，能非常有效地帮助嵌入式工程师来定位引发难以重现的软件问题的根源。



这个工具在排除嵌入式软件故障方面的功能的确非常强大！

功能显示

近程检测器设备使用传感器来计算正在靠近的目标的距离。在该设备上运行的软件是用来计算距离的，然后再利用所得到的结果，根据目标距离的远近来启动 1 个，2 个，或 3 个 LED 信号。

在这个例子中，我们利用 DT10 来监控嵌入式软件在运行时的特征，并将其与 LED 在这期间的行为关联起来，因而在任何情况下，我们都可以准确知道该软件在硬件上执行某个指定事件时所处的状态。同样地，我们还可以利用 DT10 监控正在执行的软件的特征，并确定此时硬件上的逻辑或模拟信号的状态。



使用 DT10 动态追踪器

1. 插入测试点

通过使用 DT10，可以将测试点自动插入源代码，以获取并输出软件在目标硬件设备上运行时的特征。这些特征包括：精确的时间信息，覆盖率信息，事件，变量监控数据，等等。

2. 配置测试点输出

在这个例子中，我们通过配置测试点，将执行软件的追踪数据通过 GPIO 口发送到动态追踪器装置上。目标设备上的 GPIO 针孔与 DT10 动态追踪器的 GPIO 口相连接。

3. 捕获 LED 数字信号

DT10 模拟盒也与目标设备相连接，通过设备上的 3 个 LED 来获取逻辑信号。为了响应附近的某个目标，设备上的 LED 会自动打开或关闭，这样 DT10 就可以准确追踪并监控**信号高低**转变的情况了。通过使用一个单独的时钟脉冲源，动态追踪器就能够自动同步 LED 在打开和关闭状态下所获取的数据。



这样配置工作就完成了。

如果有需要的话，可以让目标设备运行尽可能长的时间，以重现有问题的行为。通过 DT10 动态追踪器所获取的所有数据都会被同步并发送到安装在所连接的主机上的 DT10 软件中。多波示波器可以对这些数据进行实时监控，而且以后还可以回放。

测试一旦完成，DT10 分析引擎就会对所获得的数据进行扫描，并自动生成一系列分析报告，以帮助解决嵌入式软件中的问题。

DT10 生成报告这一步，有效地为工程师减少了很多负担！

分析

第一部分: 根据 LED 逻辑信号特征, 确定软件的执行状态

可以用多波示波器来检测逻辑或模拟信号的踪迹, 并通过运行传感器 LED 找出可能存在问题的阵列, 然后确定嵌入式软件执行过程中出现该阵列的具体时间。一旦发现了出现问题的具体位置, DT10 就会同步生成分析报告, 该报告有助于从不同视角检测软件在运行时的特征。

为了识别 LED 逻辑信号中可能存在问题的阵列, 用户可以在示波器中将其缩小, 并通过手动滚动来查看执行的踪迹; 然后再放大, 来对自己认为可能有问题的地方进行仔细检查。

或者, 用户可以利用 DT10 的回放功能和自动停止条件, 在示波器中有效地回放执行情况, 而且在符合自动停止条件的情况下 (如: LED1 继续, LED2 继续, LED3 断开), DT10 会自动停止回放。

既然我们已确定了可能存在的地点, 我们就可以在示波器上对其进行放大。在示波器上随便点击一下, DT10 就可以根据由其生成的一系列软件分析报告, 来同步那个时间点的执行情况, 包括软件当时所处的状态。

下面的截图显示了由 DT10 生成了几个视图, 这些视图能够帮助发现并解决问题:

Click on any point in the Multi-Wave Scope...

...DT10 updates the other windows to the same point in execution time

No.	Source	Function	Memory	Value of a variable	Elapsed time(ns)	Difference(ns)
10410 (10762)	Task_Hardware	getDistance			10,335,166,950	69,500
10411 (10763)	Task_LED.c	Task_LED	116	10,335,239,440	10,335,239,440	71,490
10412 (10765)	Task_LED.c	Task_LED		10,335,396,110	10,335,396,110	157,670
10413 (10766)	Task_LED.c	Task_LED		10,335,464,100	10,335,464,100	69,990
10414 (10767)	Task_Hardware	setLEDState		10,335,537,440	10,335,537,440	71,340
10415 (10768)	Task_Hardware	setLEDState		10,335,607,100	10,335,607,100	69,660
10416 (10769)						
10417 (10770)						

这样工程师就能够, 比如, 在某个信号状态出现之前、之时、之后检测代码变量的转变情况, 或者能够逐步跟踪执行路径, 并在硬件出现某些特殊行为之前, 发现是在执行了哪一行代码, 哪个分支时出现该问题的。

还有很多其它非常有用的、深入的报告, 能够帮助工程师分析运行时引发软件问题的根源。

第二部分：根据软件执行时的特征，确定 LED 的状态

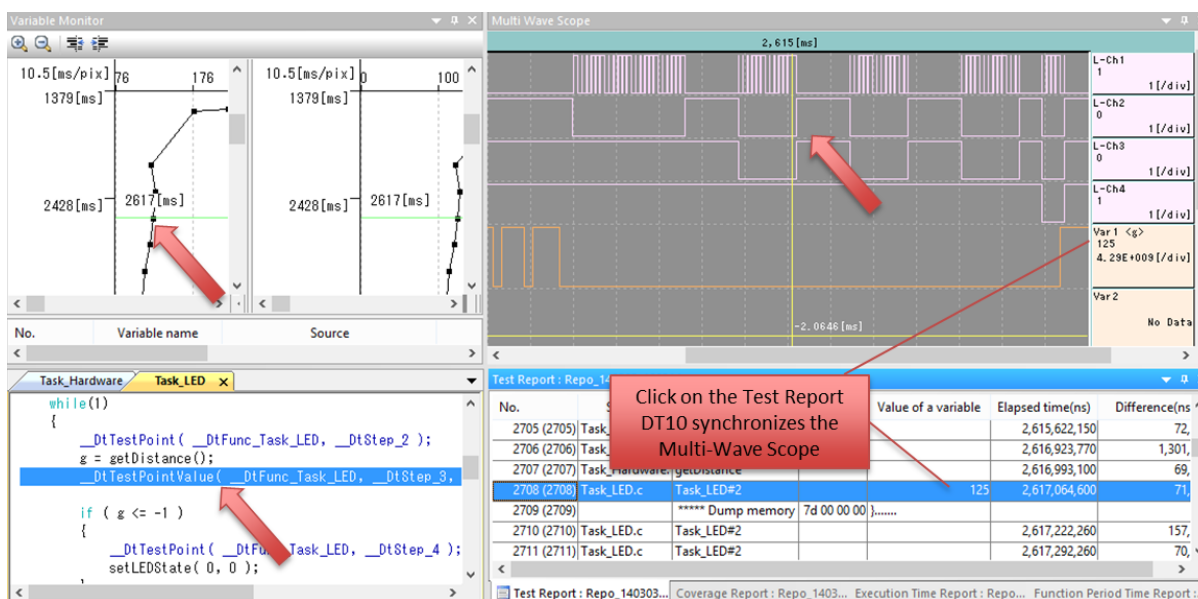
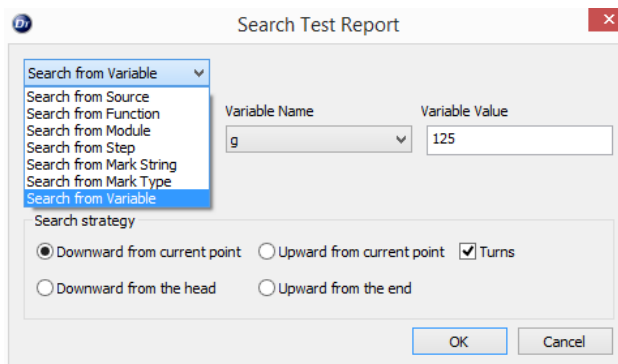
在前面的例子中，我们先使用多波示波器来确定 LED 逻辑信号踪迹所形成的阵列，再用 DT10 将其与软件当时的运行特征联系起来。

相反地，DT10 也能够很有效地查出软件在运行时的具体特征（如：变量的状态，出现的某个软件事件，执行代码的第几行）并将其与当时的 LED 信号状态联系起来。所以，我们也能够看到软件在特定的状态下，LED 逻辑信号的情况。

这个例子中，软件中的变量“g”存储的是传感器与接近目标之间的计算距离。软件会根据这个变量值将设备 LED1,2,3 设置为启动状态。

我们还能够搜索 DT10 测试报告，找出变量呈现异常值的情况。

然后在测试报告中点击有问题的那一步，DT10 就会更新其它活动窗口，让所有窗口都显示该运行时的情况。然后我们就可以通过多波示波器放大来查看逻辑和模拟信号的状态，以了解当时的 LED 状态。



总结

DT10 多波示波器和模拟盒在发现并解决嵌入式软件的问题方面有非常独特的能力。它的功能类似于逻辑分析器和示波器，可以让软件工程师看到代码的执行情况。但是 DT10 多波示波器是专门为嵌入式软件设计的，所以它的安装和配置更为简便。

嵌入式开发项目过程中经常会遇到一些非常难以重现的问题，DT10 的解决方案可以对这些问题进行追踪，帮助解决该问题。

版权声明：本文档版权归创提信息科技（上海）有限公司所有，并保留一切权利。