



Luat_IOT_SDK_C 语言下载调试手册

一、 工具介绍

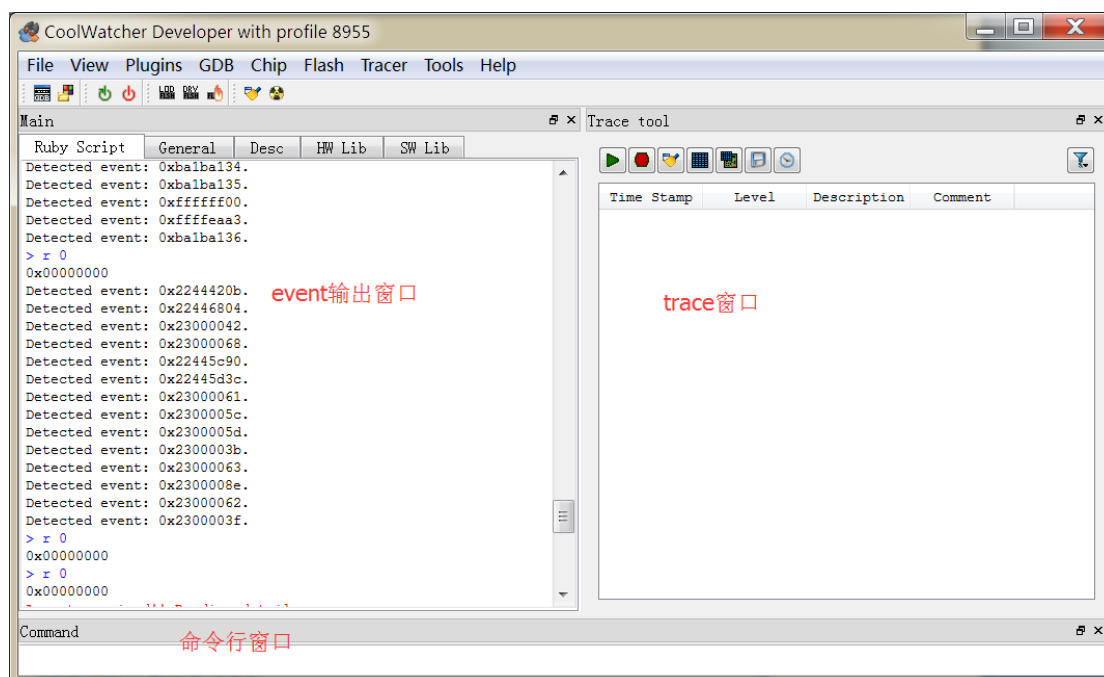
CSDK 安装后，桌面上会有一个 **coolwatcher** 工具，下载和调试都会用到该工具。



Coolwatcher 工具是通过串口和设备的 host 口进行通信，通信波特率为 921600，由于速率较高，所以建议使用 FTDI 等高速 USB 转串口线，其他 USB 转串口线可能会导致通信中断，而影响下载和调试。

Coolwatcher 官方文档位置【[c:\CSDTK\cooltools\DOC\Coolwatcher_User_Guide_N.pdf](#)】

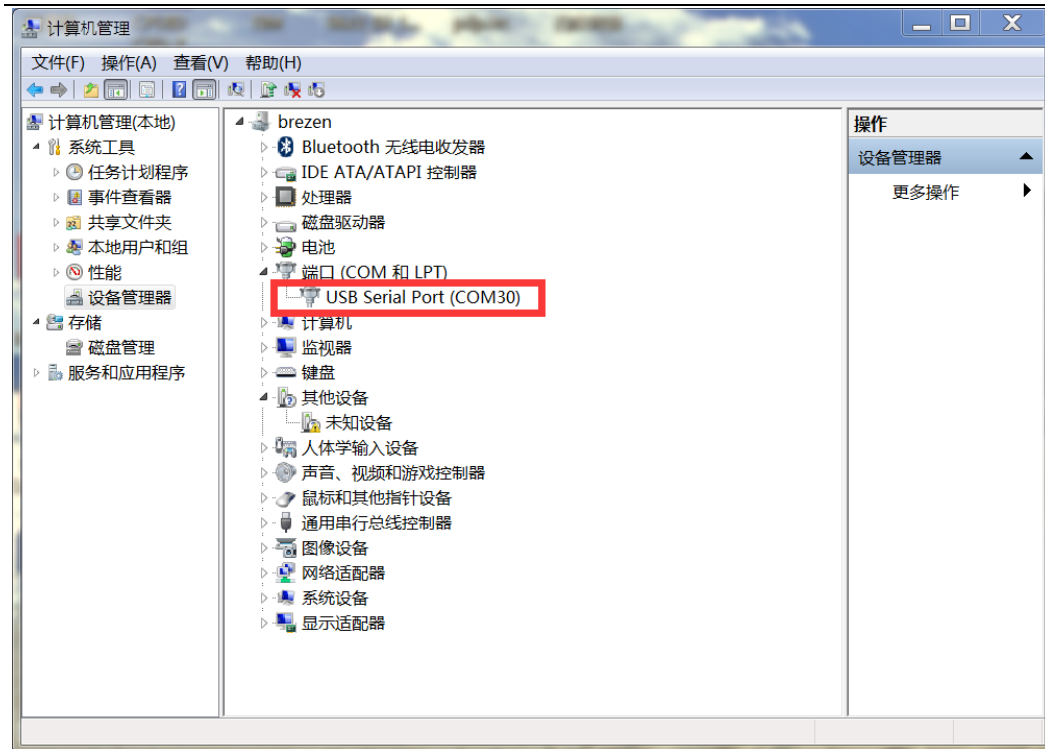
1. Coolwatcher 常用窗口



二、 打开工具

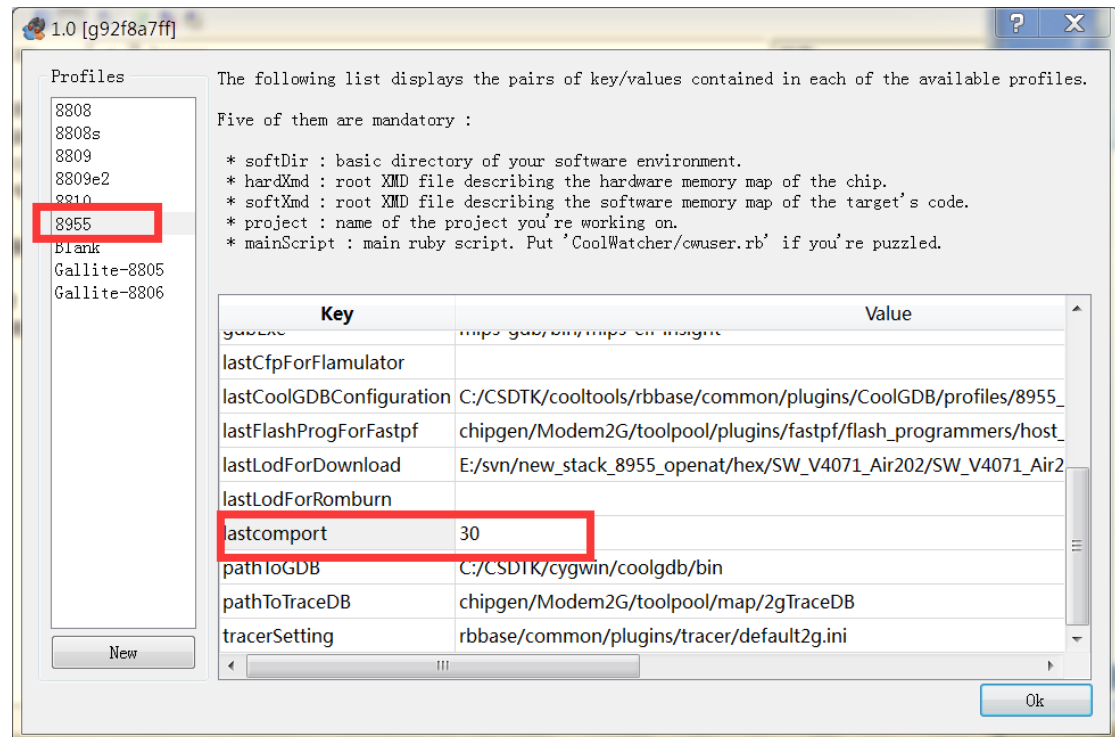
2.1 确认通信串口号

电脑设备管理器中查看串口号，如下图



2.2 配置工具

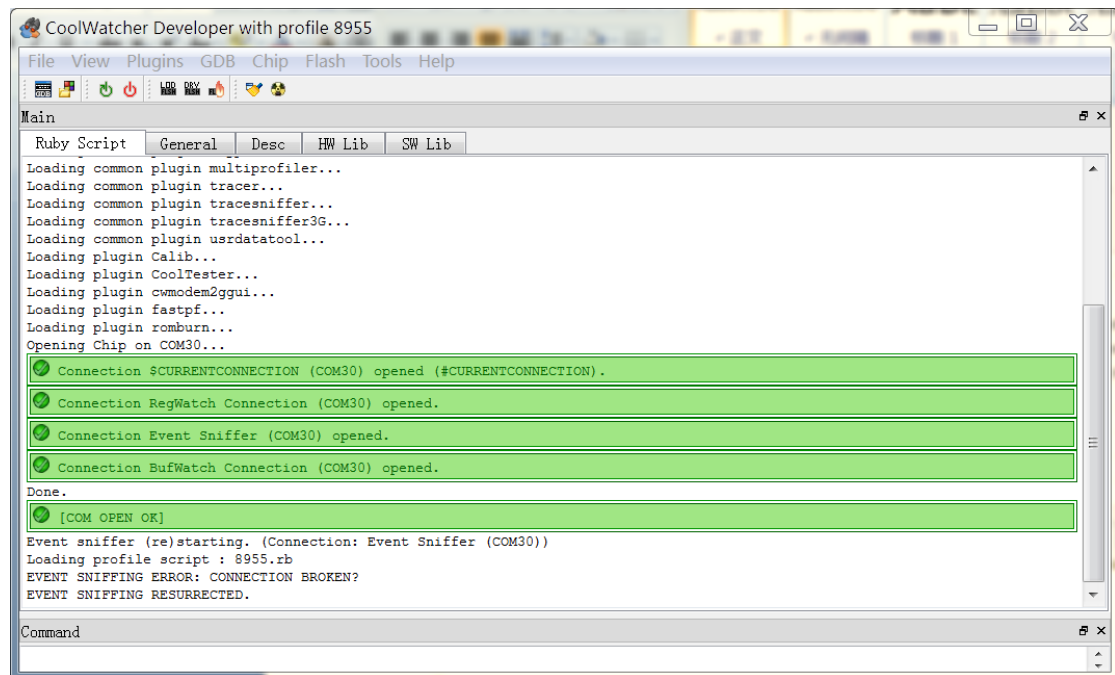
选择【8955】和 lastcomport，确认串口后点击【OK】



Lastcomport 和设备管理器中的串口号对应，例如【COM30】就填写 30。

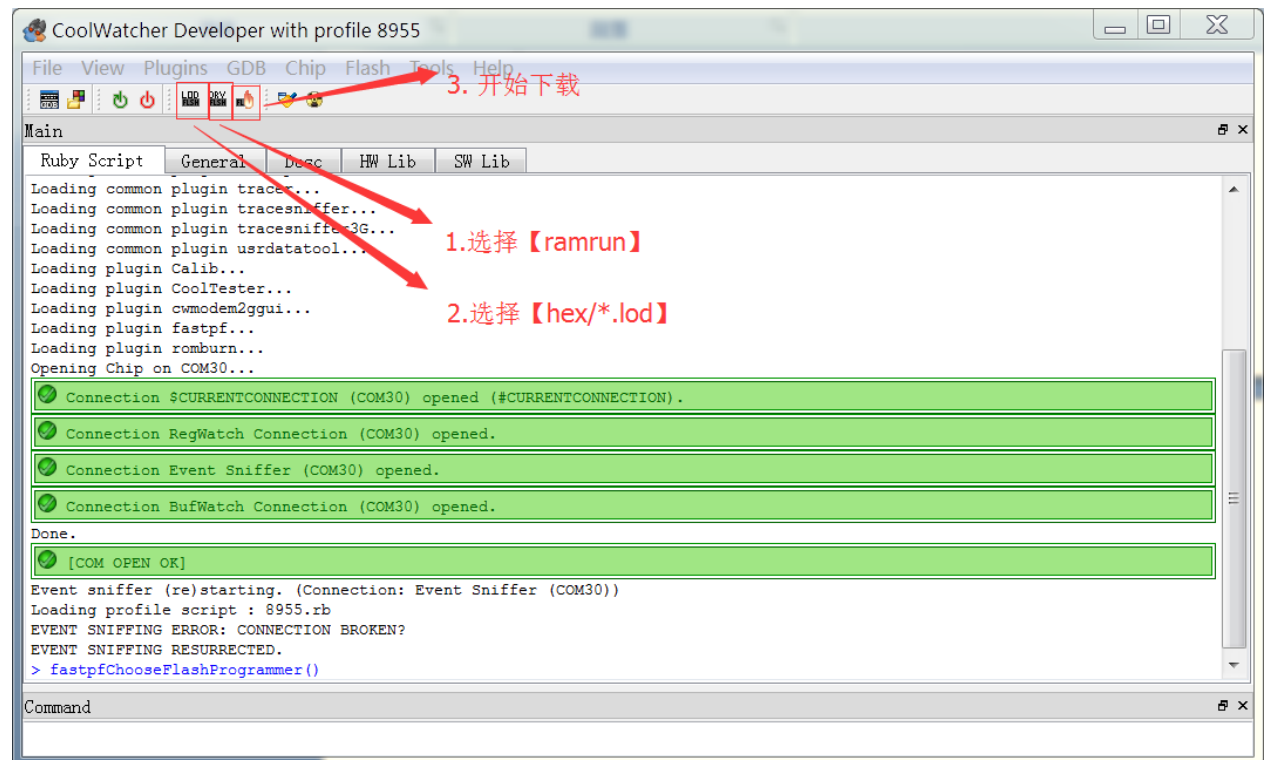
2.3 确认打开是否正常

如果串口正常打开，工具左侧会有如下提示




三、 下载

下载之前需要选择 sdk 中的【ramrun】和编译出来的【hex/xx/x.lod】文件。




3.1 选择 ramrun

Coolwatcher 有记忆功能，所以只需要在第一次使用工具时需要该步骤

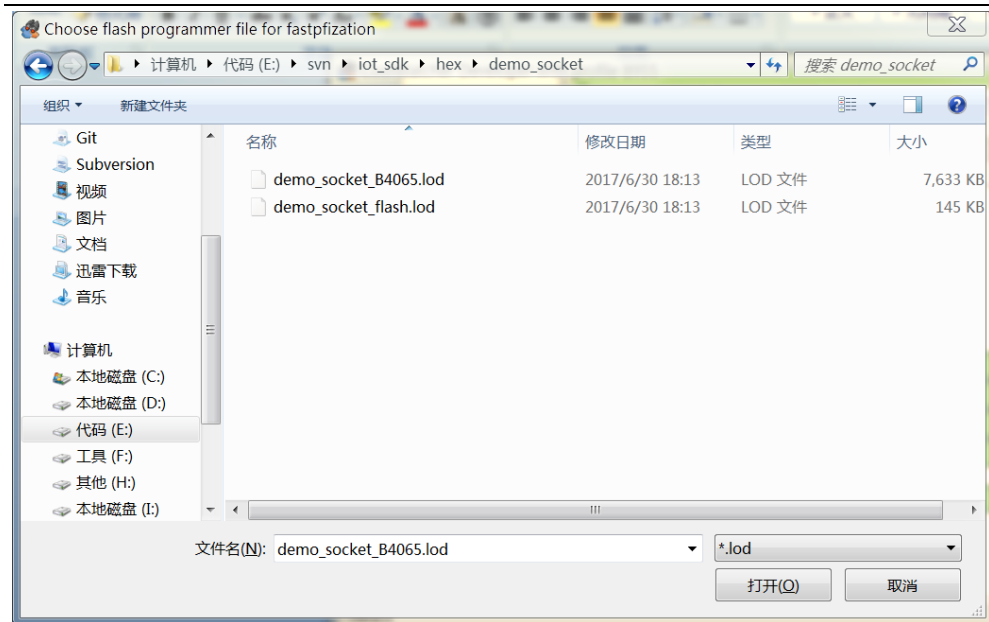
点击 ，选择【sdk 路径/ramrun/flsh_spi32m/ flsh_spi32m_CUSTOMER_host_ramrun.lod】

3.2 选择 lod 文件

点击 ，选择编译出来的【hex/xxx/xx.lod】文件有两个，例如编译【demo_socket】项目会生成下图所示的两个 lod 文件，其中【demo_socket_BXXX.lod】是包含应用层和底层软件的 lod 文件；【demo_socket_flash.lod】只包含应用层软件。

模块第一次升级或者 sdk 更新版本时需要选择【demo_socket_BXXX.lod】

其他情况下可以直接选择【demo_socket_flash.lod】，这样下载会快很多。

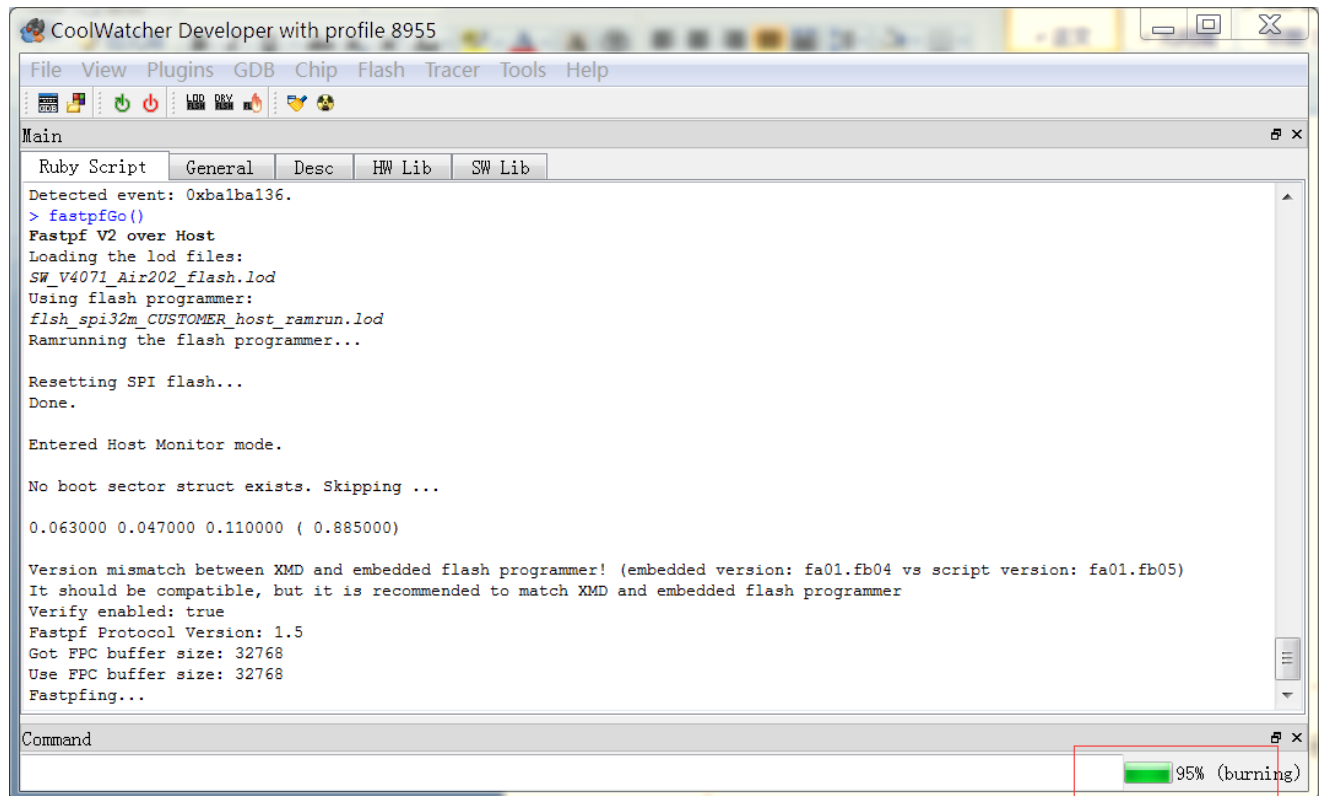


3.3 开始下载

下载之前，确认模块上电并开机，正常开机后工具左侧会有【event】输出。没有 event 需要检查模块供电是否正常，有【开机键】的板子需要长按该键，再没有输出就需要尝试关掉 coolwatcher 重新打开。

Main				
Ruby Script	General	Desc	HW Lib	
EVENT SNIFFING ERROR: CONNECTION BROKEN?				
EVENT SNIFFING RESURRECTED.				
> activateTracerGui()				
Loading Tracer GUI ...				
Tracer GUI loaded.				
TRACE SNIFFING ON.				
TRACE SNIFFING ALREADY RUNNING				
TRACE SNIFFING ALREADY RUNNING				
Detected hardware reset (0xffffffff).				
Detected event: 0x07778888.				
Detected event: 0x07778889.				
Detected event: 0x077788a0.				
Detected event: 0x077788b0.				
Detected event: 0x077788b1.				
Detected event: 0x88888880.				
Detected event: 0x88888881.				
Detected event: 0x88888882.				
Detected event: 0x88888883.				
Detected event: 0x88888884.				
Detected event: 0x88888885.				
Detected event: 0x88888886.				
Detected event: 0x88888888.				
Detected event: 0x88888889.				
Detected event: 0x88888890.				
Detected event: 0x88888891.				
Detected event: 0x88888892.				
Detected event: 0x09999092.				
Detected event: 0x09999093.				
Detected event: 0x09999098.				
Detected event: 0x09999099.				
Detected event: 0x0999909a.				
Detected event: 0x88888893.				
Detected event: 0x88888894.				
Detected event: 0x00000004.				
Detected event: 0x66660021.				
Detected event: 0x0000000f.				
Detected event: 0x00000000.				
Detected event: 0x88888895.				
Detected event: 0x88888896.				
Detected event: 0x88888897.				
Detected event: 0x88888898.				
Detected event: 0x88888899.				
Detected event: 0x8888889a.				
Detected event: 0x8888889b.				


 按钮开始下载，正常下载后工具右下角会提示下载进度



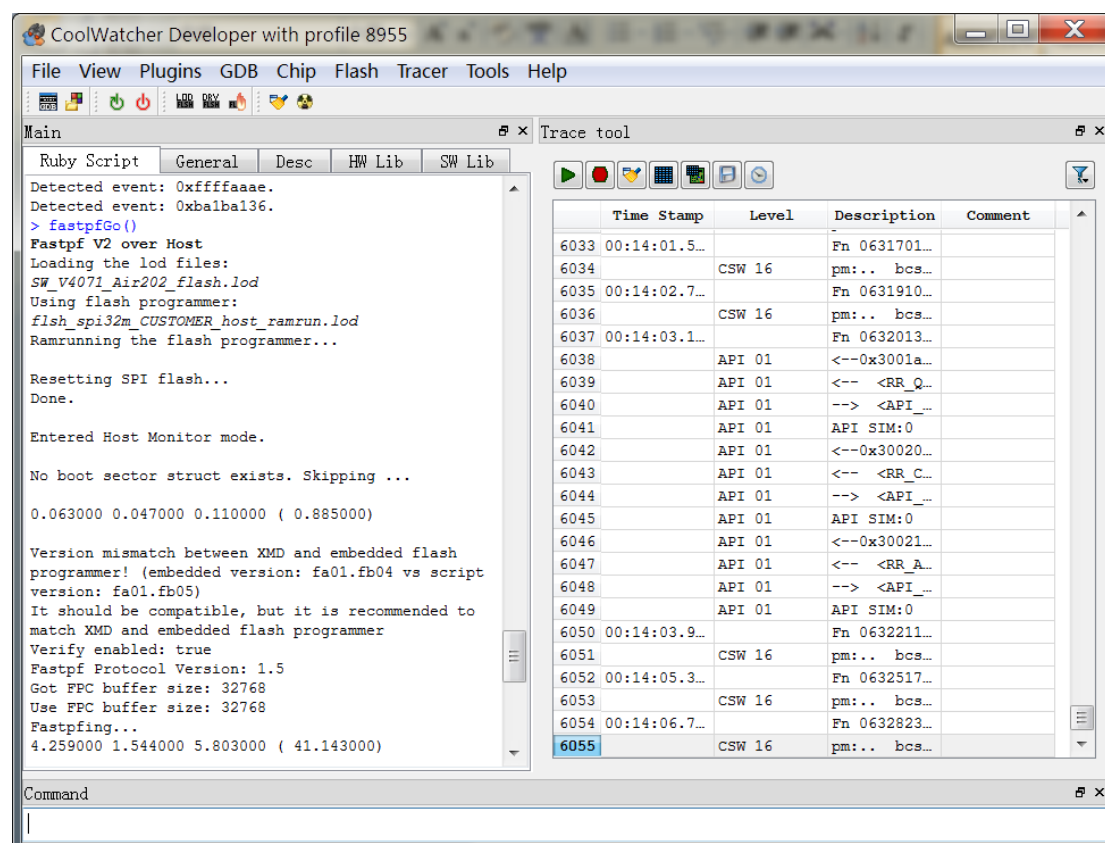
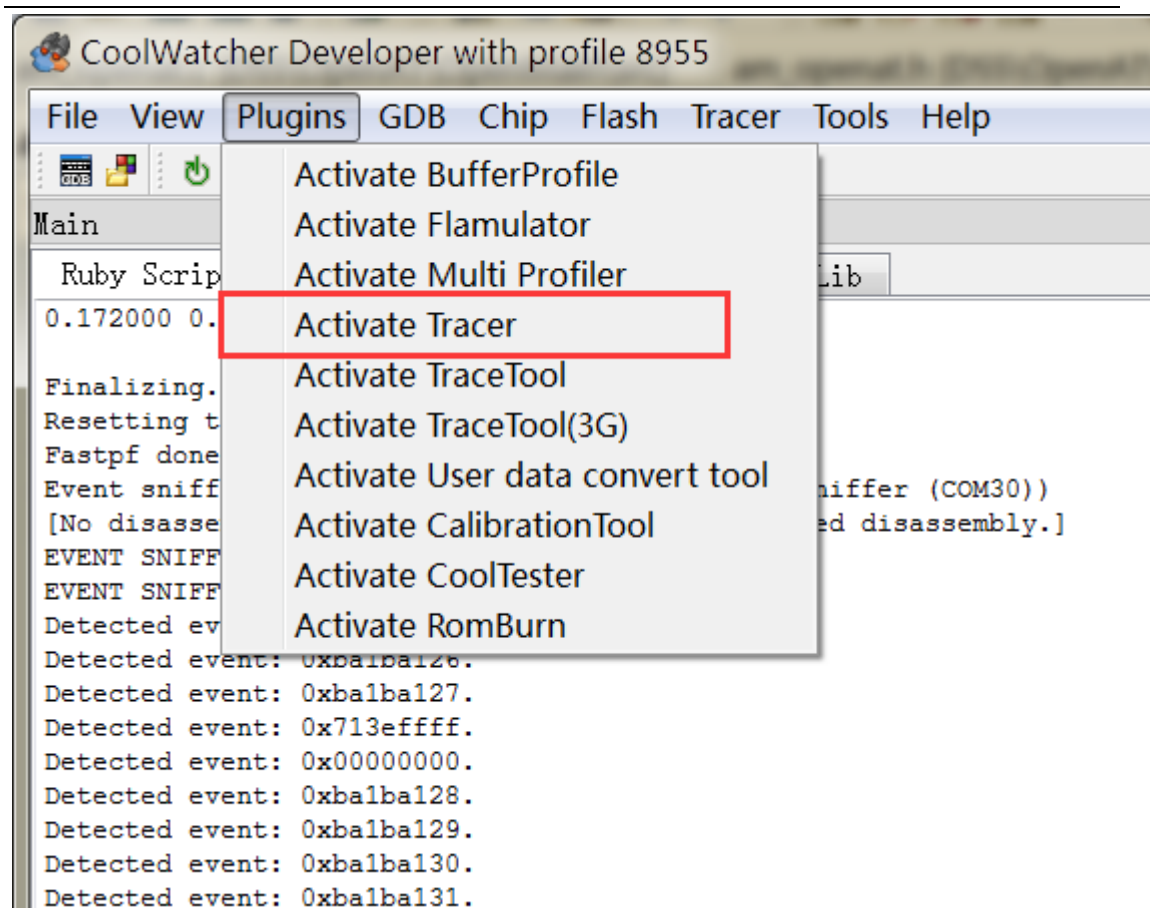
下载完成后，模块会自动重启，重启后依然会在工具左侧输出【event】数据。

四、 调试

4.1 查看 trace

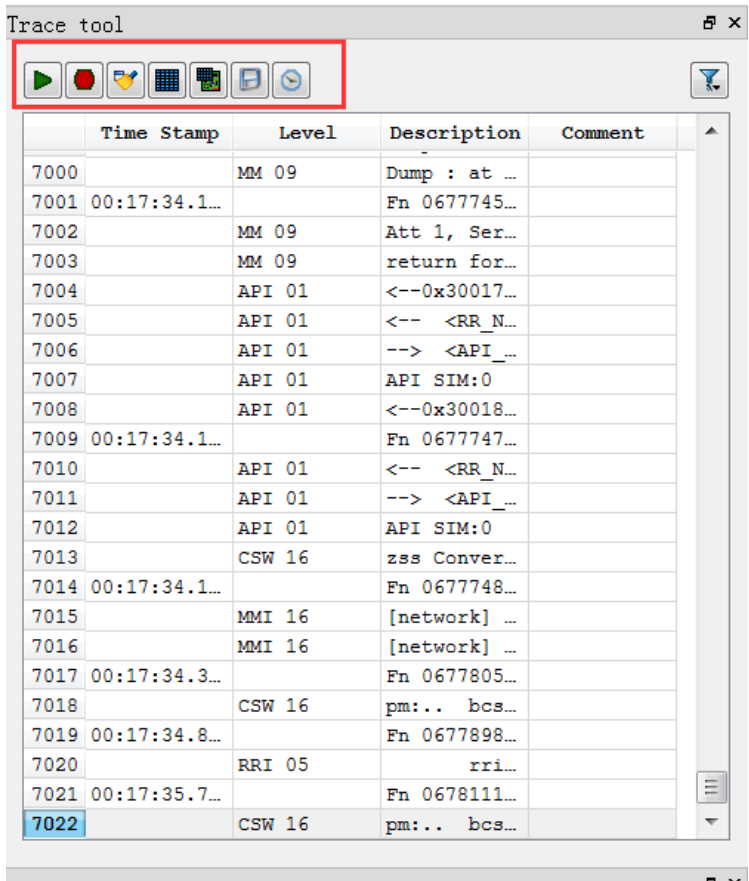
4.1.1 打开 trace tool

选择【plugins->Active Tracer】，工具右侧会出现【Trace tool】栏，如果没出现就右击一下工具空白处，选择【Trace tool】。




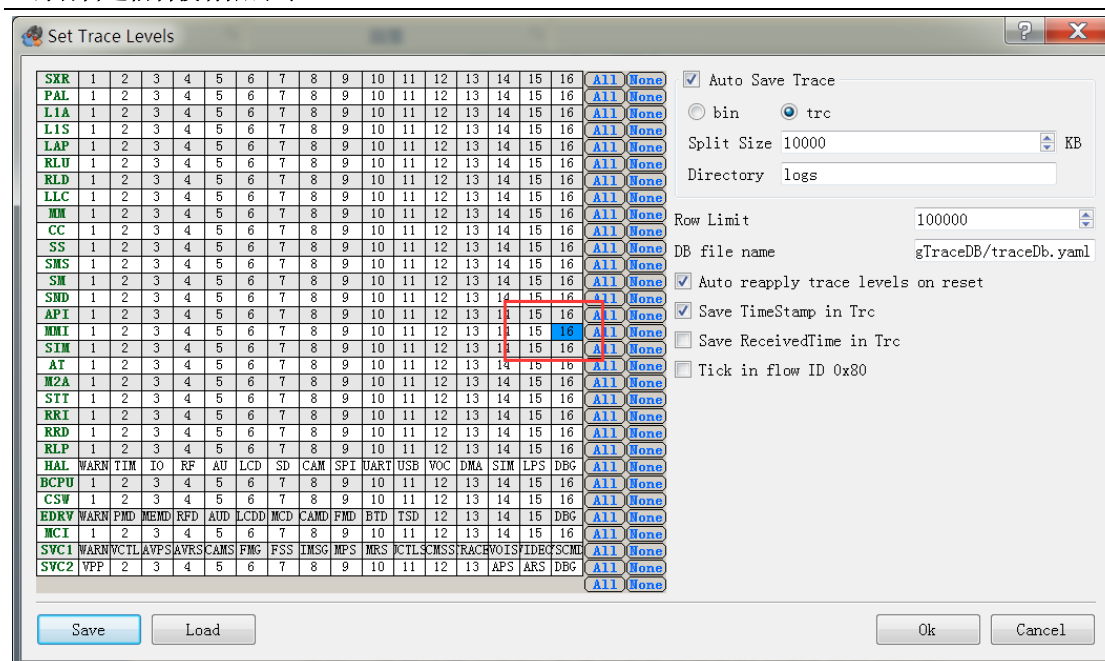
4.1.2 trace tool 工具说明

下图红框从左到右分别为【开始 trace】，【停止 trace】，【清空 trace】，【过滤 trace】，【应用过滤】，【保存 trace】，【使能 trace 时间】



4.1.3 过滤 trace

点击  按钮过滤 trace，如果只抓取应用层的 trace 可以只打开 MMI 16 部分，设置完成后点击【OK】即可。



针对底层的问题抓取 trace，需要和技术支持具体确认过滤哪些模块。

4.2 死机问题调试

死机常见的两种现象

4.2.1 软件 assert 死机

```
Detected event: 0x2300003f.
Assert received!! Reading detail...
ASSERT DETAIL :
Func:demo_socket_task @line153
Detected BCPU in the GDB loop (0x9db10000), connect GDB debugger for more info.
Detected CPU fatal error (0x9db00010), it went through GDB stub and will reboot.
```

4.2.2 cpu 异常

```
Detected CPU fatal error (0x9db00010), it went through GDB stub and will reboot.
```

4.2.3 Coolgdb 调试死机问题

点击菜单项 Tools->GDB Launcher，启动如下配置框：

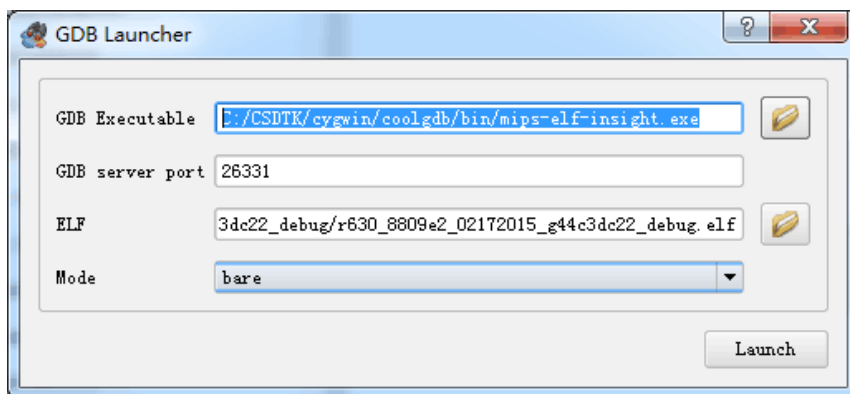
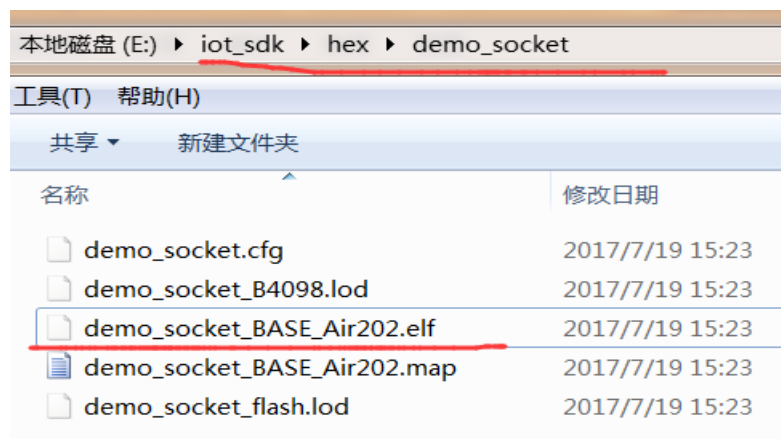


图 15

【GDB exe 文件】选择 cygwin 安装包中的

C:\CSDTK\cygwin\coolgdb\bin\mips-elf-insight.exe 文件；

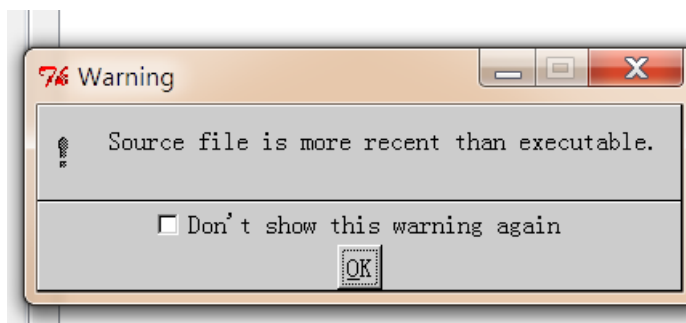
【elf 文件】选择 sdk 工程/hex 路径下的文件如下图



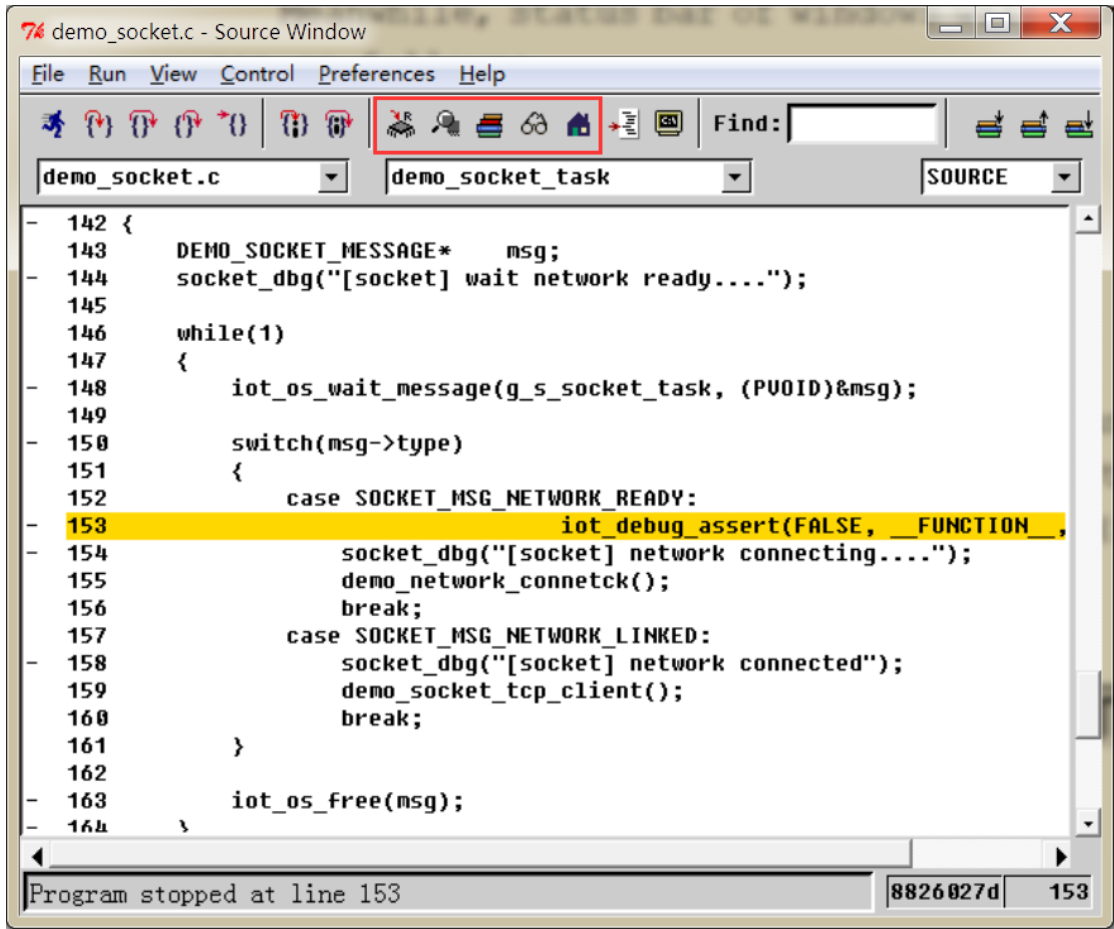
【Mode 模式】一般选择 bare

三种模式“bare”、“sx”、“live”；“sx”相比“bare”启动的 gdb，多了两个命令，thread info 和 thread；“live”模式一般用于严重的死机分析。

点击 Launch 按钮后，会弹出如下界面：

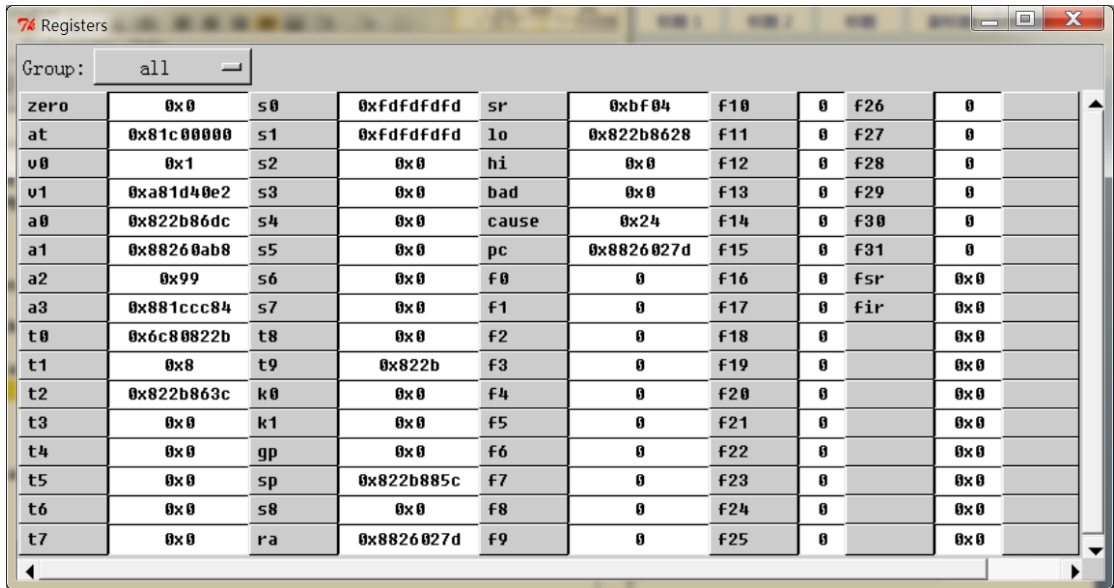


点击图中 ok 按钮，会弹出如下界面



其中常用的信息如上图中的红框部分，从左到右分别为：【寄存器信息】，【内存信息】，【调用栈】，【查看全局变量】，【局部变量】。

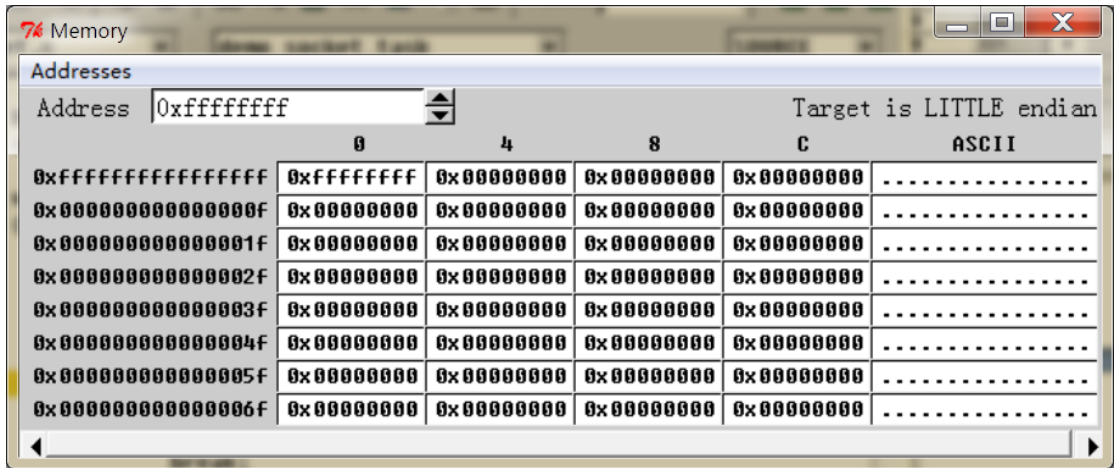
点击 ，显示如下




其中【cause】值/4 表示 cpu 异常原因值，如下图的【ExcCode】。

ExcCode	助记符	描述
0	Int	中断
1	Mod	存储操作时，该页在 TLB 中被标记为只读。
2	TLBL	没有 TLB 转换（读写分别）。也就是 TLB 中没有和程序地址匹配的有效入口。
3	TLBS	当根本没有匹配项（连无效的匹配项都没有）并且 CPU 尚未处于异常模式—SR(EXL) 置位—即 TLB 失效时，为高效平滑处理这种常见事件而采用的特殊异常入口点。
4	AdEL	（取数、取指或者存数时）地址错误：这要么是在用户态试图存取 kuseg 以外的空间，或者是试图从未对齐的地址读取一个双字、字或者半字。
5	AdES	
6	IBE	总线错误（取指或者读取数据）：外部硬件发出了某种出错信号，具体该怎么做与系统有关。因存储而导致的总线错，只能作为一个为了获取要写入的高速缓存行而执行的高速缓存读操作的结果间接出现。
7	DBE	
8	Syscall	执行了一条 syscall 指令。
9	Bp	执行了一条 break 断点指令，由调试程序使用。
10	RI	不认识的（或者非法的）指令码。
11	CpU	试图运行一条协处理器指令，但是在 SR(CU3-0) 中并没有使能相应的协处理器。 具体说，就是当 FPU 可用位 SR(CU1) 没有置位时从浮点操作得到的异常：因而是浮点仿真开始的地方。
12	Ov	自陷形式的整数算术指令（比如说 add 但 addu 不会）导致的溢出。C 语言程序不使用溢出-自陷指令。
13	TRAP	符合了 teq 等条件自陷指令的某一条。
14		目前未用。在有些拥有 L2 高速缓存的老式的 CPU 上，当硬件探测到可能的高速缓存重影时使用这位，在 4.12 节对此有解释。
15	FPE	浮点异常。（在某些很老的 CPU 上，浮点异常以中断形式出现。）
16-17	-	定制的异常类型，与具体实现相关。
18	C2E	来自协处理器 2 的异常（如果有的话，就是对指令集的一种定制的扩展）。
19-21	-	保留给未来扩展使用
22	MDMX	试图运行 MDMX 指令，但是 SR(MX) 位没有置位（很可能该 CPU 没有实现 MDMX）。
23	Watch	load/store 的物理地址匹配了使能的 WatchLo/WatchHi 寄存器。

点击 ，显示如下



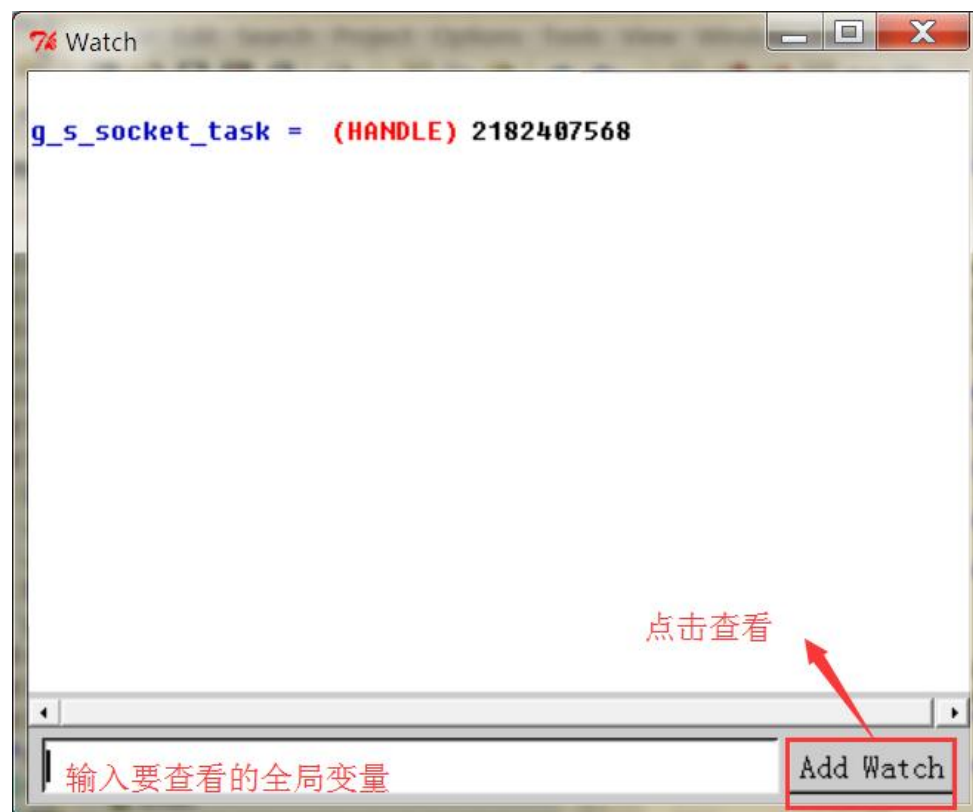
可以通过【hex/xx/xx.map】文件搜索全局变量对应的地址，保留地址的后面 4 个字节，从这里查看对应内存的信息。


点击 ，显示如下：

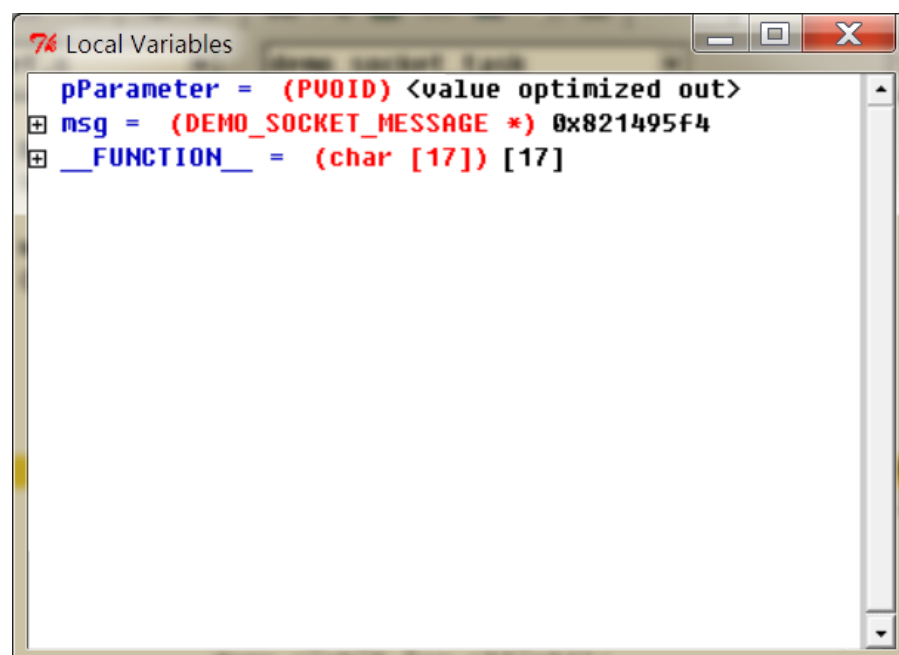


从上到下分别显示函数调用关系，配合  可以查看对应函数中没有被编译器优化的局部变量值。

点击 ，显示如下：




点击 , 显示如下



配合 stack 堆栈信息，可以查看对应函数中未被优化的局部变量

注意：使用 coolgdb 时注意 coolwatcher 中的脚本信息，正确信息如下：


```
Launching debugging system on:
E:/svn/iot_sdk/hex/demo_socket/demo_socket_BASE_Air202.elf
Launching CoolGDB with configuration file :
C:/CSDTK/cooltools/rbbase/common/plugins/CoolGDB/profiles/8955_xcpu.cgdb.
Generating :
C:/CSDTK/cygwin/coolgdb/bin/coolgdbinit and telling GDB to find CoolGDB on TCP port: 26331.
```

如果 elf 文件不正确或者不存在就无法正常观察调试信息。sdk 编译成功后，会在【hex/xxx/xxx.elf】文件，如果 elf 不正确，可以通过 ，重新选择 lod 文件，工具会默认选择 lod 文件同目录下的 elf 文件。

五、Coolwatcher 常见问题

5.1 无法正常通信

一旦串口无法正常通信，就无法进行下载、调试、输出 trace。
无法通信分为以下三种情况：

5.1.1 电脑串口异常

这种情况下，coolwatcher 工具都会无法正常打开对应串口，可以重新插拔串口尝试恢复

5.1.2 模块睡眠

在模块正常工作的情况下，开机后等一段时间就会进入睡眠，这时候 trace 可以正常输出，但是**无法进行下载和调试**，可以通过【coolwatcher 命令行窗口】中输入【r 0】来确认通信是否正常
通信正常

```
> r 0
0x00000000
^
```

通信异常

```
> r 0
⚠ `CHST_Read32': Request to coolhost timed out ! (CoolHostTimeoutError)
./rbbase/common/coolhost_connections.rb161 `CHST_Read32'
./rbbase/common/coolhost_connections.rb161 `read32'
./rbbase/common/base.rb514 `r'
```

5.1.3 工具异常

在以上信息都确认没有问题的情况下，如果依然无法正常下载和调试，可以通过【coolwatcher 命令行窗口】输入【reop】或者重启 coolwatcher 软件来恢复