

异常调试介绍升级介绍

Rev 1.0 —— 2021 年 12 月 22 日

This translated version is for reference only, and the English version shall prevail in case of any discrepancy between the translated and English versions.

版权所有 2019 杰理科技有限公司未经许可，禁止转载

1、在 download.c 文件如下位置添加下面语句，在 tools 文件夹下面会生成 sdk.lst 文件

```
board_config.h  C board_ac695x_demo_cfg.h  C download.c 9+ X  sdk.elf.lst  C lib_system_config.c
u > br23 > tools > C download.c
97 @echo          SDK BR23
98 @echo *****
99 @echo %date%
00
01 cd %~dp0
02
03
04 set OBJDUMP=C:\JL\pi32\bin\llvm-objdump.exe
05 set OBJCOPY=C:\JL\pi32\bin\llvm-objcopy.exe
06 set ELFFILE=sdk.elf
07
08 %OBJDUMP% -D -address-mask=0x1fffffff -print-dbg %ELFFILE% > %ELFFILE%.lst
09 %OBJCOPY% -O binary -j .text %ELFFILE% text.bin
10 %OBJCOPY% -O binary -j .data %ELFFILE% data.bin
11 %OBJCOPY% -O binary -j .data code %ELFFILE% data code.bin
```

2、在 lib_system_config.c 文件开启异常打印，如下图

```
const int config_print_time = 0;

///异常中断，asser打印开启
#ifdef CONFIG_RELEASE_ENABLE
const int config_asser = 1;
#else
const int config_asser = 1;
#endif

const int config_system_info = 0;

//=====//
//                      SDFILE 精简使能                      //
//=====//
#ifdef CONFIG_SOUNDBOX_FLASH_256K
const int SDFILE_VFS_REDUCE_ENABLE = 1;
#else
NORMAL apps/soundbox/log_config/lib_system_config.c
```

3、当程序跑异常后会有下面打印

```
XX
0x5A5A5A5A 0x5A5A5A5A JL_LRCT->CON:0x00000000 0xFFFFFFFF
usp : 0x002026C4
ssp : 0x00005310
sp : 0x000052B0
Stack : 0x00005310
rets 0x01E5D1CC
reti 0x01E4BADE
retx : 0x00000000
rete : 0x00000000
psr : 0x00000000
icfg : 0x07010280
0x01E4BAB8 ->0x01E5D1C2 ->0x01E4BABC ->0x01E4BAC6

sem_post_rets_addr: 0x00000000
sem_rets_addr: 0x01E310EC
mutex_rets_addr: 0x00000000
```

rets : 异常位置返回地址

reti : 异常触发位置

蓝色框: 异常触发前 pc 调用地址

可以根据这些地址, 查看 lst 文件具体位置来定位异常触发在那句代码

但是异常触发有可能不是第一现场

Lst 文件如下

```
1
2 sdk.elf: file format ELF32-pi32v2
3
4 Disassembly of section .text:
5 text_begin:
6 1e00120: ee ff 40 13 00 00 sp = 4928 ## startup.S:27:0
7 1e00126: ed ff 40 13 00 00 ssp = 4928 ## startup.S:28:0
8 1e0012c: 81 ea d6 12 call 140716 <boot_info_init : 1e226dc > ##
9 1e00130: c3 ff e0 22 00 00 r3 = 8928 <report_bss_begin : 22e0 > ##
10 1e00136: 41 20 r1 = 0 ## startup.S:36:0
11 1e00138: c2 ff 9c 7f 00 00 r2 = 32668 <report_bss_size : 7f9c > ##
12 1e0013e: a2 a2 r2 = r2 >> 2 ## startup.S:38:0
13 1e00140: 02 03 rep 2 r2 { ## startup.S:40:0
14 1e00142: b1 05 [r3++=4] = r1
15 } ## startup.S:41:0
16 1e00144: f2 5d if (r2 != 0) goto -6 <text_begin+0x20 : 1e00140 >
17 1e00146: c4 ff 40 13 00 00 r4 = 4928 <report_irq_stack_end : 1340 >
18 1e0014c: c1 ff 74 19 e5 01 r1 = 31791476 <text_end : 1e51974 > ## sta
19 1e00152: c2 ff 80 0f 00 00 r2 = 3968 <report_data_size : f80 > ## sta
20 1e00158: a2 a2 r2 = r2 >> 2 ## startup.S:51:0
21 1e0015a: 12 03 rep 4 r2 { ## startup.S:53:0
22 1e0015c: 13 05 r3 = [r1++=4] ## startup.S:54:0
23 1e0015e: c3 05 [r4++=4] = r3
24 } ## startup.S:55:0
25 1e00160: f2 5c if (r2 != 0) goto -8 <text_begin+0x3A : 1e0015a >
26 1e00162: c4 ff 80 a2 00 00 r4 = 41600 <report_data_code_begin : a280 >
27 1e00168: c1 ff f4 28 e5 01 r1 = 31795444 <moveable_slot_begin : 1e528f4 >
28 1e0016e: c2 ff 84 30 00 00 r2 = 12420 <report_data_code_size : 3084 >
29 1e00174: a2 a2 r2 = r2 >> 2 ## startup.S:65:0
30 1e00176: 12 03 rep 4 r2 { ## startup.S:67:0
31 1e00178: 13 05 r3 = [r1++=4] ## startup.S:68:0
NORMAL cpu/br30/tools/sdk.lst unix < utf-8 < asm
```

