GigaDevice Semiconductor Inc.

Arm® Cortex®- M3/M4/M23/M33 32-bit MCU

应用笔记

AN020

目录

图索引

表索引

1. 简介

CmBacktrace（Cortex Microcontroller Backtrace）是一款针对 ARM Cortex-M 系列 MCU 的错误代码自动追踪、定位，错误原因自动分析的开源库。主要特性如下：

- 支持的错误包括：
  - 断言（assert）
  - 故障（Hard Fault, Memory Management Fault, Bus Fault, Usage Fault, Debug Fault）
- 故障原因 自动诊断 ：可在故障发生时，自动分析出故障的原因，定位发生故障的代码位置，而无需再手动分析繁杂的故障寄存器；
- 输出错误现场的 函数调用栈（需配合 addr2line 工具进行精确定位），还原发生错误时的现场信息，定位问题代码位置、逻辑更加快捷、精准。也可以在正常状态下使用该库，获取当前的函数调用栈；
- 支持 裸机 及以下操作系统平台：
  - RT-Thread
  - uCOS
  - FreeRTOS（需修改源码）
- 根据错误现场状态，输出对应的 线程栈 或 C 主栈；
- 故障诊断信息支持多国语言（目前：简体中文、英文）；
- 适配 Cortex-M0/M3/M4/M7/M33 MCU；
- 支持 IAR、KEIL、GCC 编译器；

本文介绍了如何将 CmBacktrace 移植到 GD32 工程下的方法。

2. CmBacktrace 移植

2.1. CmBacktrace 下载

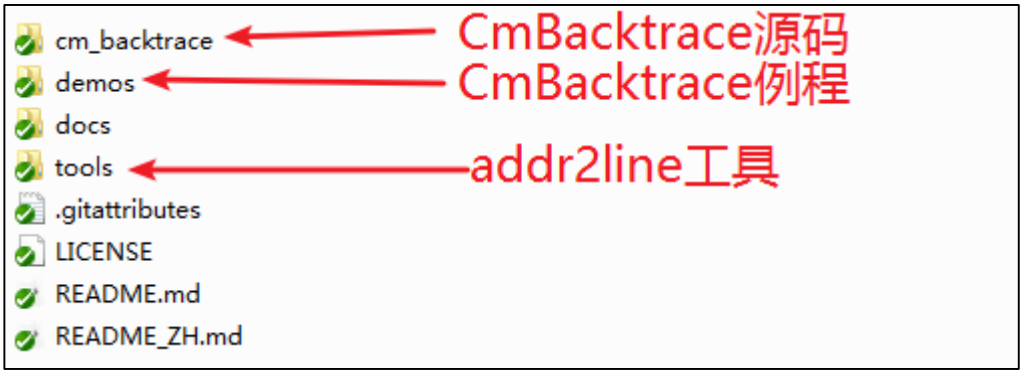
本文介绍的 CmBacktrace 移植平台为 GD32E507Z-EVAL 开发板。CmBacktrace 移植的 IDE 平台为 KEIL5 和 IAR。

CmBacktrace 源码可由 https://github.com/armink/CmBacktrace 下载。目前测试的 CmBacktrace 软件版本为 1.4.0，具体如下图所示。

图 2-1. CmBacktrace 的版本信息

```
/* library software version number */
#define CMB_SW_VERSION              "1.4.0"
```

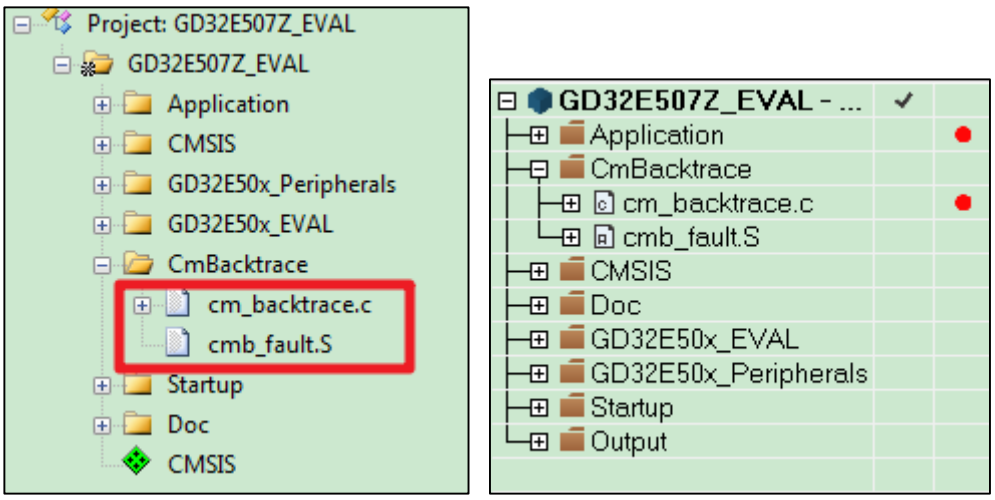
图 2-2. CmBacktrace 库文件简单介绍



2.2. 添加 CmBacktrace 源码文件

本文介绍的移植方式基于 GD32E507Z_EVAL_Demo_Suites 里的 01_GPIO_Running_LED 工程，首先将 CmBacktrace\cm_backtrace 库文件拷贝至 01_GPIO_Running_LED 文件下中。然后打开工程，在工程中添加 cm_backtrace.c 和 cmb_fault.S 两个文件。

图 2-3. Keil 和 IAR 工程配置



由于 cmb_fault.S 会使用到 HardFault_Handler，所以可以原有的 HardFault_Handler 注释掉。

图 2-4. 注释掉原有的 HardFault_Handler

```
/*!
    \brief      this function handles HardFault exception
    \param[in]  none
    \param[out] none
    \retval     none
*/
//void HardFault_Handler(void)
//{
//    /* if Hard Fault exception occurs, go to infinite loop */
//    while (1){
//    }
//}
```

2.3. IDE 工程配置

CmBacktrace 在使用 KEIL5 编译器时，必须配置支持 C99 标准。Keil 和 IAR 的工程配置如下图所示。

图 2-5. Keil 工程配置

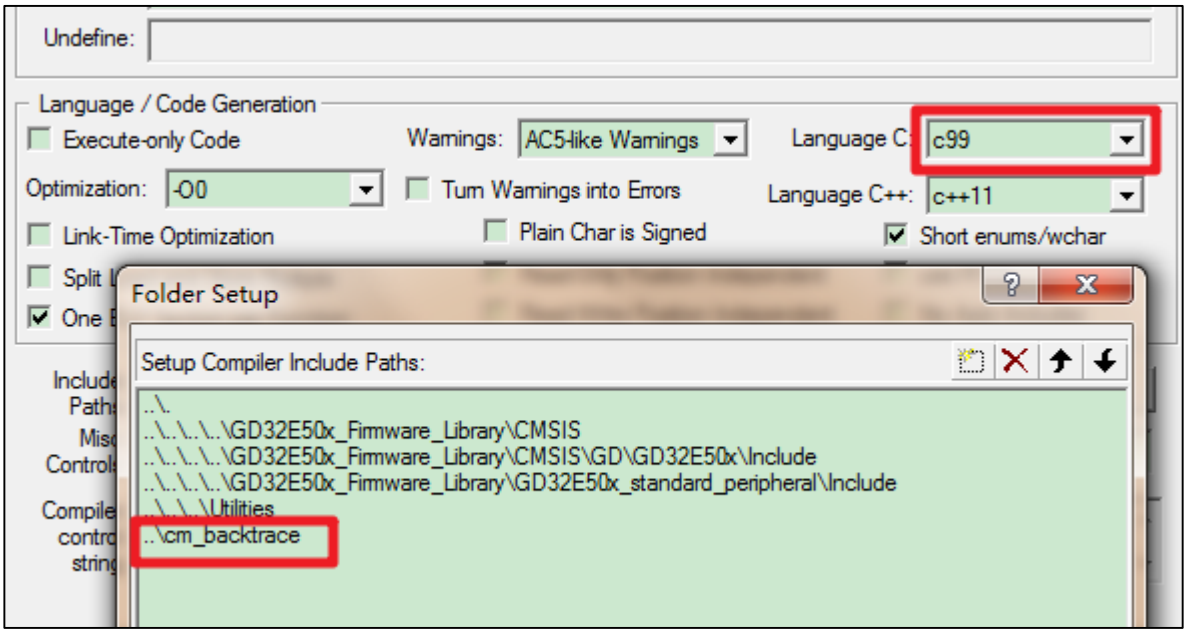
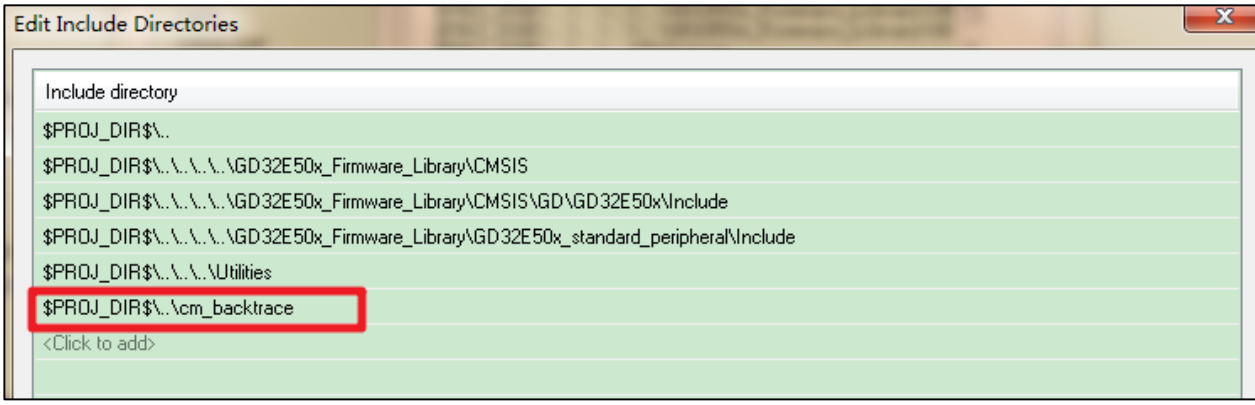


图 2-6. IAR 工程配置



2.4. CmBacktrace 参数配置

在 cmb_def.h 中定义了针对不同的平台和场景的配置选项。

表 2-1. CmBacktrace 的配置参数

配置名称	功能	备注
cmb_println(...)	错误及诊断信息输出	必须配置
CMB_USING_BARE_METAL_PLATFORM	是否使用在裸机平台	使用则定义该宏
CMB_USING_OS_PLATFORM	是否使用在操作系统平台	操作系统与裸机必须二选一
CMB_OS_PLATFORM_TYPE	操作系统平台	RTT/uCOS/FREERTOS
CMB_CPU_PLATFORM_TYPE	CPU平台	M0/M3/M4/M7/M33

CMB_USING_DUMP_STACK_INFO	是否使用 Dump 堆栈的功能	使用则定义该宏
CMB_PRINT_LANGUAGE	输出信息时的语言	CHINESE/ENGLISH

CmBacktrace GD32E507Z 工程中的配置如下图所示。

图 2-7. cmb_def.h 的配置

```
#ifndef _CMB_CFG_H_
#define _CMB_CFG_H_

/* print line, must config by user */
#define cmb_println(...)              printf(__VA_ARGS__);printf("\r\n")
/* enable bare metal(no OS) platform */
#define CMB_USING_BARE_METAL_PLATFORM
/* cpu platform type, must config by user */
#define CMB_CPU_PLATFORM_TYPE         CMB_CPU_ARM_CORTEX_M33
/* enable dump stack information */
#define CMB_USING_DUMP_STACK_INFO
/* language of print information */
#define CMB_PRINT_LANGUAGE            CMB_PRINT_LANUUAGE_ENGLISH
#endif /* _CMB_CFG_H_ */
```

需要实现重定向函数

2.5. 其他说明

原代码中的 cmb_def.h 使用__CC_ARM 宏来区分是哪个 IDE 环境，对于 ARM Compiler Version 6 的编译器来说，使用的宏是__ARMCC_VERSION，具体如下图所示。

图 2-8. 原始 cmb_def.h 的条件编译

```
#if defined(__CC_ARM) || defined(__CLANG_ARM)
    /* C stack block name, default is STACK */
    #ifndef CMB_CSTACK_BLOCK_NAME
    #define CMB_CSTACK_BLOCK_NAME         STACK
    #endif
    /* code section name, default is ER_IROM1 */
    #ifndef CMB_CODE_SECTION_NAME
    #define CMB_CODE_SECTION_NAME         ER_IROM1
    #endif
#elif defined(__ICCARM__)
    /* C stack block name, default is 'CSTACK' */
    #ifndef CMB_CSTACK_BLOCK_NAME
    #define CMB_CSTACK_BLOCK_NAME         "CSTACK"
    #endif
    /* code section name, default is '.text' */
    #ifndef CMB_CODE_SECTION_NAME
    #define CMB_CODE_SECTION_NAME         ".text"
    #endif
#elif defined(__GNUC__)
```

图 2-9. ARM Compiler Version 6 关于__ARMCC_VERSION 的解释

Table 9-21 Predefined macros

Name	Value	When defined
__arm__	-	Always defined for the ARM compiler, even when you specify the --thumb option. See also __ARMCC_VERSION.
__ARMCC_VERSION	ver	Always defined. It is a decimal number, and is guaranteed to increase between releases. The format is PVVbbbb where: ▪ P is the major version ▪ VV is the minor version ▪ bbbb is the build number.

所以如果使用 ARM Compiler Version 6 的编译器，需要做一些修改，具体如下。

图 2-10. 使用 ARM Compiler Version 6 的编译器的修改

```
#if defined (__ARMCC_VERSION) && (__ARMCC_VERSION >= 6010050)
    /* C stack block name, default is STACK */
    #ifndef CMB_CSTACK_BLOCK_NAME
    #define CMB_CSTACK_BLOCK_NAME          STACK
    #endif
    /* code section name, default is ER_IROM1 */
    #ifndef CMB_CODE_SECTION_NAME
    #define CMB_CODE_SECTION_NAME          ER_IROM1
    #endif
#elif defined(__ICCARM__)
    /* C stack block name, default is 'CSTACK' */
    #ifndef CMB_CSTACK_BLOCK_NAME
    #define CMB_CSTACK_BLOCK_NAME          "CSTACK"
    #endif
    /* code section name, default is '.text' */
    #ifndef CMB_CODE_SECTION_NAME
    #define CMB_CODE_SECTION_NAME          ".text"
    #endif
#elif defined(__GNUC__)
```

3. CmBacktrace 功能测试

本章节介绍不对齐和除零两种错误导致 HardFault，并由 CmBacktrace 捕获并通过串口打印，具体如下。

表 3-1. fault_test_by_unalign 代码

```
void fault_test_by_unalign(void) {
    volatile int * SCB_CCR = (volatile int *) 0xE000ED14; // SCB->CCR
    volatile int * p;
    volatile int value;

    *SCB_CCR |= (1 << 3); /* bit3: UNALIGN_TRP. */

    p = (int *) 0x00;
    value = *p;
    printf("addr:0x%02X value:0x%08X\r\n", (int) p, value);

    p = (int *) 0x04;
    value = *p;
    printf("addr:0x%02X value:0x%08X\r\n", (int) p, value);

    p = (int *) 0x03;
    value = *p;
    printf("addr:0x%02X value:0x%08X\r\n", (int) p, value);
}
```

表 3-2. fault_test_by_div0 代码

```
void fault_test_by_div0(void) {
    volatile int * SCB_CCR = (volatile int *) 0xE000ED14; // SCB->CCR
    int x, y, z;

    *SCB_CCR |= (1 << 4); /* bit4: DIV_0_TRP. */

    x = 10;
    y = 0;
    z = x / y;
    printf("z:%d\n", z);
}
```

根据电脑具体使用是什么操作系统，将 CmBacktrace 的 tools 文件夹中已存放的 addr2line.exe 可以直接拷贝至 C:\Windows 下，或者将 CmBacktrace 仓库的 tools 文件夹路径添加至到环境变量 path 中，这样都能保证命令行工具能正常使用 addr2line 命令。

Keil 下生成的 fault_test_by_unalign 错误报告，以及使用 addr2line 打印的结果，具体如下所

示。

图 3-1. Keil 下生成的 fault_test_by_unalign 错误报告

```
Firmware name: CmBacktrace, hardware version: V1.0.0, software version: V0.1.0
Fault on interrupt or bare metal(no OS) environment
===== Thread stack information =====
  addr: 20000928    data: 200000b4
  addr: 2000092c    data: 0800305d
  addr: 20000930    data: 0800304f
  addr: 20000934    data: 0000001c
  addr: 20000938    data: 08003034
  addr: 2000093c    data: 080001c1
  addr: 20000940    data: 00000003
  addr: 20000944    data: e000ed14
  addr: 20000948    data: 00000000
  addr: 2000094c    data: 08001af5
  addr: 20000950    data: 00000000
  addr: 20000954    data: 00000000
  addr: 20000958    data: 00000000
  addr: 2000095c    data: 00000000
  addr: 20000960    data: 00000000
  addr: 20000964    data: 00000000
  addr: 20000968    data: 00000000
  addr: 2000096c    data: 08000679
========================================
=================== Registers information ====================
  R0 : 0000001c  R1 : 00000003  R2 : 00000007  R3 : 40013800
  R12: 00000001  LR : 08001811  PC : 080017c4  PSR: 29000000
==============================================================
Usage fault is caused by indicates that an unaligned access fault has taken place
Show more call stack info by run: addr2line -e CmBacktrace.axf -a -f 080017c4 08001810
080001c0 08001af4 08000678
```

图 3-2. 根据 Keil 生成的 axf 文件，使用 addr2line 工具获取函数调用栈信息

```
_Suites\Projects\GPIO_Running_LED\MDK-ARM\output>addr2line -e Project.axf -a -f
080017c4 08001810 080001c0 08001af4 08000678
0x080017c4
fault_test_by_unalign
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\01_GPIO_Running_LED\MDK-ARM/../main.c:119
0x08001810
fputc
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\01_GPIO_Running_LED\MDK-ARM/../main.c:139
0x080001c0
Reset_Handler
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\01_GPIO_Running_LED\MDK-ARM/..\..\..\..\GD32E50x_Firmware_Libra
ry\CMSIS\GD\GD32E50x\Source\ARM\startup_gd32e50x_cl.s:185
0x08001af4
main
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\01_GPIO_Running_LED\MDK-ARM/../main.c:65
0x08000678
$d
??:?
```

IAR 下生成的 fault_test_by_unalign 错误报告，以及使用 addr2line 打印的结果，具体如下所示。

图 3-3. IAR 下生成的 fault_test_by_unalign 错误报告

```
addr:0x00 value:0x20000A40
addr:0x04 value:0x08002E55

Firmware name: CmBacktrace, hardware version: V1.0.0, software version: V0.1.0
Fault on interrupt or bare metal(no OS) environment
===== Thread stack information =====
  addr: 20000a28    data: 08002e55
  addr: 20000a2c    data: 00000000
  addr: 20000a30    data: 00000000
  addr: 20000a34    data: 08002cb7
  addr: 20000a38    data: 00000000
  addr: 20000a3c    data: 08002ffb
====================================
================== Registers information ====================
  R0 : 0000001c  R1 : 00000003  R2 : 20000a24  R3 : 00000000
  R12: 00000008  LR : 08002d81  PC : 08002d5a  PSR: 09000000
=============================================================
Usage fault is caused by indicates that an unaligned access fault has taken place
Show more call stack info by run: addr2line -e CmBacktrace.out -a -f 08002d5a 08002d80
08002cb6 08002ffa
```

图 3-4. 根据 IAR 生成的 out 文件，使用 addr2line 工具获取函数调用栈信息

```
_Suites\Projects\GPIO_Running_LED\EWARM\GD32E50x\Exe>addr2line -e Project.out -a
 -f 08002d5a 08002d80 08002cb6 08002ffa
0x08002d5a
fault_test_by_unalign
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\GPIO_Running_LED/main.c:119
0x08002d80
fputc
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\GPIO_Running_LED/main.c:139
0x08002cb6
main
[redacted]GD32E50x_Demo_Suites\GD32E507Z_EVAL_Demo
_Suites\Projects\GPIO_Running_LED/main.c:65
0x08002ffa
_call_main
??:?
```

4. 版本历史

表 4-1. 版本历史

版本号.	说明	日期
1.0	首次发布	2021 年 11 月 30 日

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.