

General_Purpose_MCU_sdk 介绍

Rev 1.0 —— 2010 年 06 月 16 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 2021 杰理科技有限公司未经许可，禁止转载

目录

Chapter 1	SDK 说明.....	4
1.1	编写目的.....	4
1.2	开发前准备工作.....	4
1.3	sdk 结构.....	4
1.4	main 函数流程.....	7
1.5	时钟设置及部分使用注意事项.....	8

修改日志

版本	日期	描述
1.0	2021/06/16	SDK 简要介绍。
更新:	● 建立初始版本	

Chapter 1 SDK 说明

1.1 编写目的

该文档主要描述 SDK 整体框架，方便进行开发。

1.2 开发前准备工作

使用本 SDK 前，需向相关人员申请相关的编译环境、工具链安装包等，安装后获取 license 文件方能正常进行开发。

1.3 sdk 结构

如图 1 所示，打开工程后，sdk 分为三大区域：源码、头文件、汇编，用户主要使用到的区域为前两个。

源码区结构如图 2 所示，apps 文件夹为用户代码存放位置，board 为一些 demo 配置区，src 为各个驱动接口或测试代码，其中 boot.c 为 main 函数的存放位置(后面篇幅会讲到 main 函数流程)。

头文件区如图 3 所示，芯片与接口相关的头文件存放在 bsp/ACxxxx/include/asm 下面，其他一些公共的头文件存放在 bsp/include，部分头文件没有对应的源码是因为存放在库里。

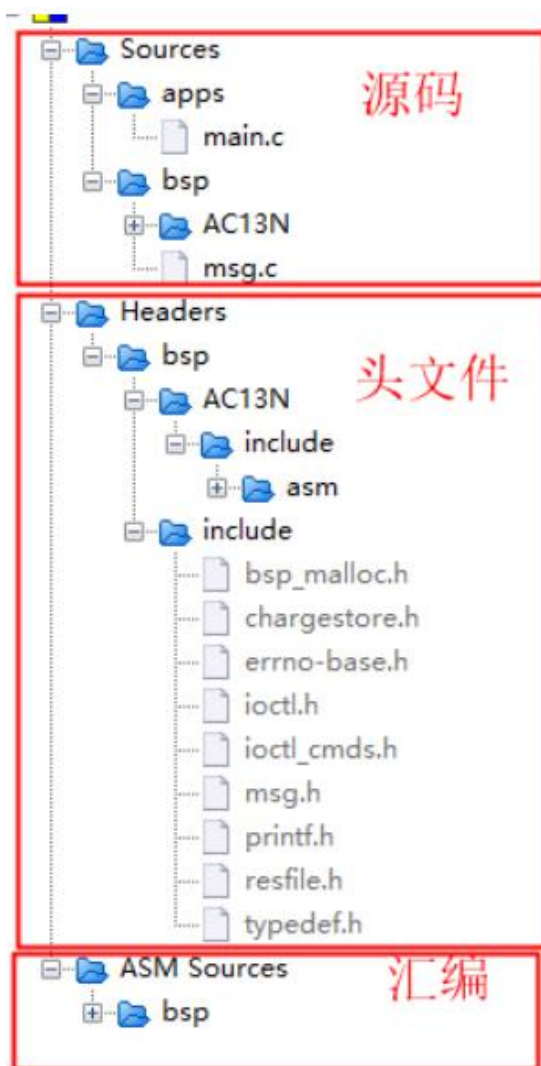


图 1

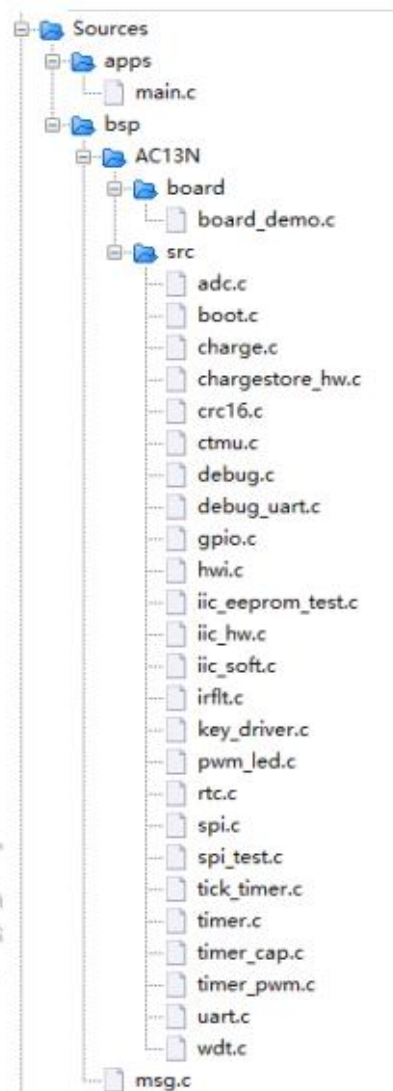


图 2

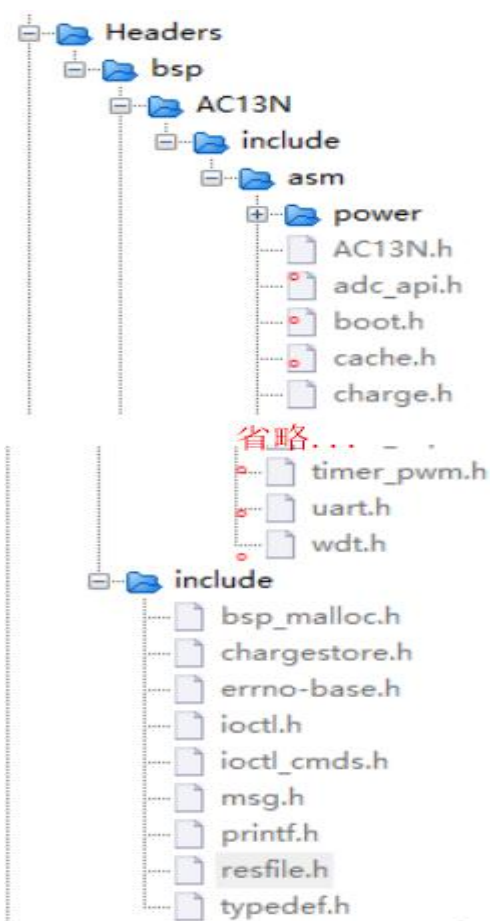


图 3

1.4 main 函数流程

main 函数存放的位置在 apps/bsp/ACxxx/src/boot.c，结构如图 4 所示，

```
int main()
{
    efuse_init();

    debug_uart_early_init();

    clk_voltage_mode(CLOCK_MODE_ADAPTIVE, SYSVDD_VOL_SEL_126V);

    clk_early_init(SYS_CLOCK_INPUT_PLL_RCL, 32768, 24000000);

    puts("\n\n");
    printf("boot %s %s\n", __DATE__, __TIME__);
    clk_dump();
    puts("\n\n");

    u32 my_reset = power_reset_source_dump();

    memory_init();

    //E11 系统必须提前打开
    pll_init();
    wdt_init(WDT_4S);
    /*wdt_close();*/

    resfile_init();

    board_init();

    request_irq(1, 2, exception_irq_handler, 0);

    debug_init();

    timer1_init();

    msg_init();

    printf("+++++++my_reset:0x%x\n", my_reset);

    user_main();
    return 0;
}
```

此部分请勿删减

用户主循环

图 4

1.5 时钟设置及部分使用注意事项

用户可以根据自己的需求去设置不同的时钟频率，用户跑起来后可以使用接口 `clk_set` 重新设置时钟，如设置 48m 为：

```
clk_set("sys",48*1000000);
```

用户可以通过 `clk_out` 接口输出来判断是否设置成功，如使用 PA3 输出系统时钟：

```
clk_out(IO_PORTA_03,HSB_CLK_OUT);
```

在使用外设模块（如 timer）选择时钟源时，省晶振的方案不能选择 OSC 作为时钟源，否则该模块将无法工作。