

PY32F030系列应用指南

-中断向量重新映射



Puya Semiconductor (Shanghai) Co., Ltd

目录

| | | |
|-------|---------------------------|---|
| 1 | 简介 | 3 |
| 1.1 | 实现原理 | 3 |
| 1.2 | Bootloader跳转APP注意事项 | 4 |
| 1.2.1 | 代码描述 | 4 |
| 1.3 | IAP代码 | 4 |
| 1.3.1 | 代码描述 | 4 |
| 1.3.2 | MDK配置 | 5 |
| 1.4 | APP代码 | 5 |
| 1.4.1 | 代码描述 | 5 |
| 1.4.2 | MDK配置 | 6 |
| 2 | 版本历史 | 7 |

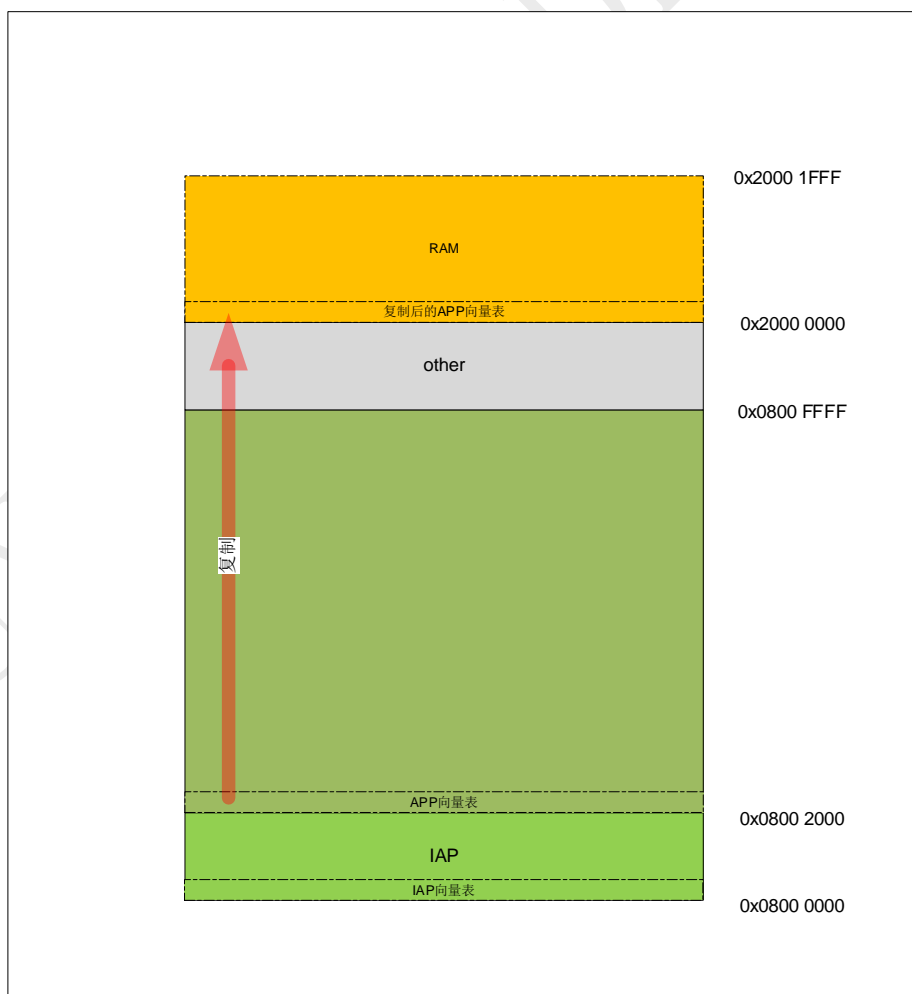
1 简介

PY32F030 系列微控制器采用高性能的 32 位 ARM®Cortex®-M0+内核，在系统复位后的第 4 个系统时钟进行锁存启动模式，常规使用的启动模式是从Main flash启动，即 0x800 0000 跟启动存储空间的0x0000 0000 对齐。从地址 0x0000 0000 取堆栈顶的值，然后从启动存储器的 0x0000 0004 地址开始执行指令。

对于部分应用场景，客户的代码并不都在Main flash的起始地址(0x800 0000)运行。例如用户自定义的引导程序（IAP）实现用户应用程序（APP）的升级，就需要用到中断向量表的重新映射。此应用笔记的目的是讲述如何实现中断向量表的重新映射。

1.1 实现原理

PY32F030系列需要将中断向量表从内部FLASH复制到RAM后，再设置SCB->VTOR寄存器，这样操作后APP才能正常运行。示意图如下，将APP向量表复制到RAM的起始地址，并将SCB->VTOR寄存器同样设置为RAM的起始地址



具体可参考demo例程，IAP和APP代码。

1.2 Bootloader跳转APP注意事项

在Boot跳转APP时，如果Bootloader系统时钟为PLL,跳转到APP前，必须把系统时钟切换到HSI,否则会导致程序工作不正常。

1.2.1 代码描述

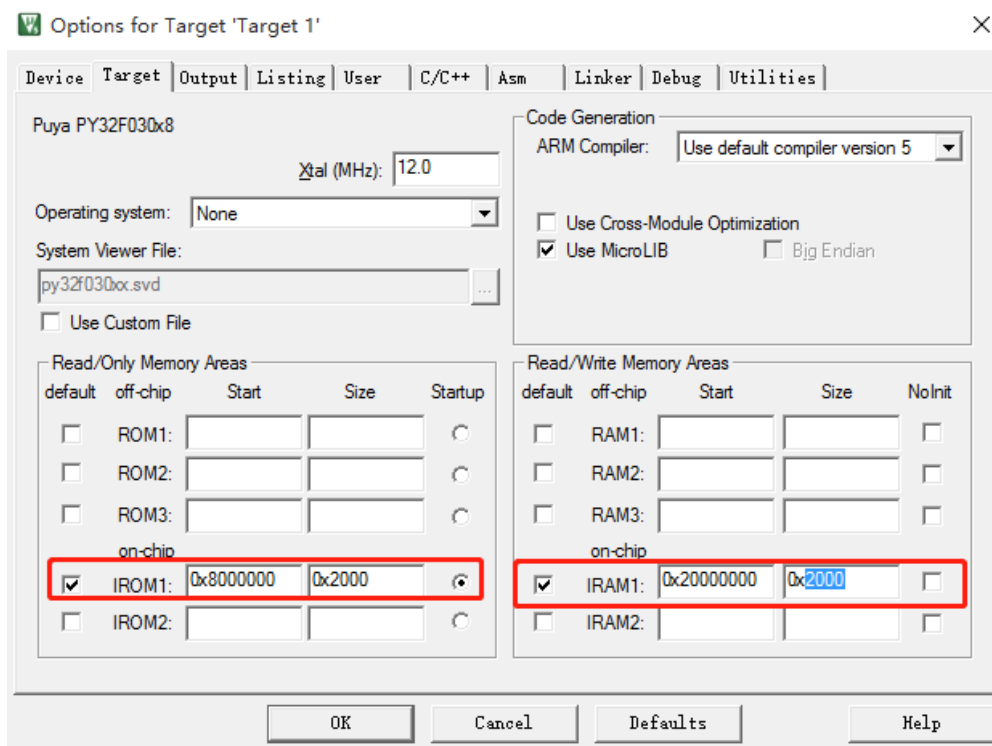
```
int main(void)
{
    /* 配置系统时钟为PLL */
    SystemClock_Config();
    ...
    update_flag = *((unsigned int *)ApplicationUpdateFlagAddr);
    if(update_flag != 0x78563412)
    {
        /* 重新配置系统时钟为HSI */
        LL_RCC_SetSysClkSource(LL_RCC_SYS_CLKSOURCE_HSI);
        while(LL_RCC_GetSysClkSource() != LL_RCC_SYS_CLKSOURCE_STATUS_HSI)
        {
            ...
        }
        /* 程序跳转至APP */
        Jump_App();
    }
}
```

1.3 IAP代码

1.3.1 代码描述

```
30 int main(void)
31 {
32     BSP_PB_Init(BUTTON_KEY, BUTTON_MODE_GPIO);
33     /* Test if Key push-button on PY32F030 Start_kit Board is pressed */
34     if (BSP_PB_GetState(BUTTON_KEY) == GPIO_PIN_RESET)
35     {
36         /* If Key is pressed, execute user code */
37         IAP_Init();
38     }
39     else
40     { /* Keep the user application running */
41
42         /* Test if user code is programmed starting from address "APPLICATION_ADDRESS" */
43         if (((__IO uint32_t*)APPLICATION_ADDRESS) & 0x2FFE0000) == 0x20000000)
44         {
45             /* Jump to user application */
46             JumpAddress = (__IO uint32_t*) (APPLICATION_ADDRESS + 4);
47             Jump_To_Application = (pFunction) JumpAddress;
48
49             /* Initialize user application's Stack Pointer */
50             __set_MSP((__IO uint32_t*) APPLICATION_ADDRESS);
51
52             /* Jump to application */
53             Jump_To_Application();
54         }
55     }
56
57     /* Infinite loop */
58     while (1)
59     {
60     }
61 }
62 }
```

1.3.2 MDK配置



1.4 APP代码

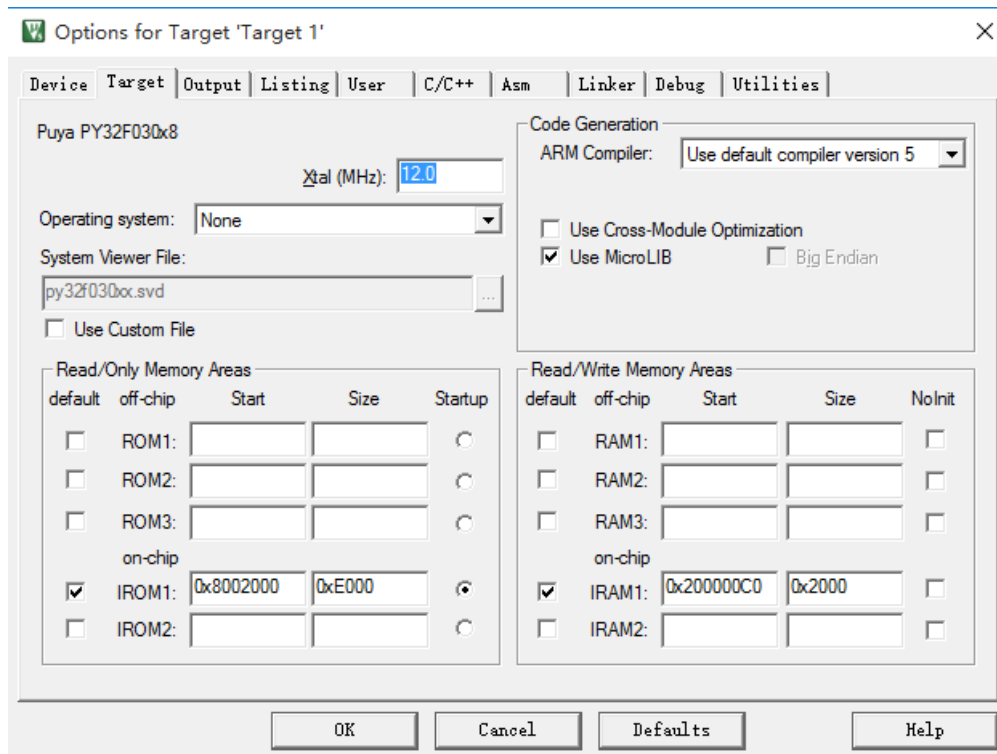
1.4.1 代码描述

```

31 int main(void)
32 {
33     uint32_t i;
34     //初始化所有外设, flash接口, systick
35     HAL_Init();
36
37     /* Copy the vector table from the Flash (mapped at the base of the application
38      load address 0x08002000) to the base address of the SRAM at 0x20000000. */
39     for(i = 0; i < 48; i++)
40     {
41         VectorTable[i] = *(__IO uint32_t*)(APPLICATION_ADDRESS + (i<<2));
42     }
43
44     SCB->VTOR = SRAM_BASE;
45     TIM14_init();
46
47     while (1)
48     {
49     }
50 }
51

```

1.4.2 MDK配置



2 版本历史

| 版本 | 日期 | 更新记录 | 作者 |
|------|------------|------|-----|
| V1.0 | 2022.03.04 | 初版 | LSQ |
| | | | |
| | | | |



Puya Semiconductor Co., Ltd.

IMPORTANT NOTICE

Puya Semiconductor reserves the right to make changes without further notice to any products or specifications herein. Puya Semiconductor does not assume any responsibility for use of any its products for any particular purpose, nor does Puya Semiconductor assume any liability arising out of the application or use of any its products or circuits. Puya Semiconductor does not convey any license under its patent rights or other rights nor the rights of others.