# 程序移植到SRAM中

### 程序变动:

打开宏定义: #define VECT\_TAB\_SRAM , 目的:

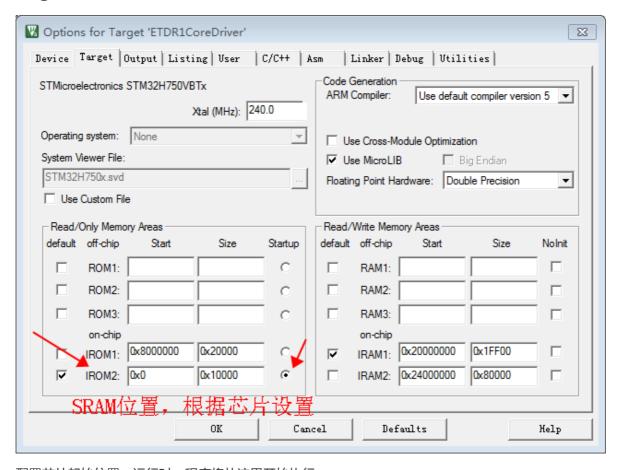
```
/* Configure the Vector Table location add offset address for cortex-M7 -----
----*/
#ifdef VECT_TAB_SRAM

SCB->VTOR = D1_ITCMRAM_BASE | VECT_TAB_OFFSET; /* Vector Table Relocation in
Internal AXI-RAM */ //D1_AXISRAM_BASE
#else
SCB->VTOR = FLASH_BANK1_BASE | VECT_TAB_OFFSET; /* Vector Table Relocation in
Internal FLASH */
#endif
```

设置中断向量表的位置,定义在SRAM中,这里的位置是可以变动的,但是要和编译设置里的startup位置保持一致。

## 项目设置:

#### target:



配置芯片起始位置,运行时,程序将从这里开始执行

#### Load:

更改RO数据存储位置

| Cortex-M Target Driver Setup  |             |               |                       | ×    |
|---|-------------|---------------|-----------------------|------|
| Debug   Trace Flash Download  | Pack        |               |                       |      |
| Download Function  LOAD  C Erase Full C Program  C Erase Sector Verify  Do not Erase Reset and Run  RAM for Algorithm  :art: 0x20000000 ize: 0x00001000 |             |               |                       |      |
| Programming Algorithm  Description  | Device Size | Device Type   | Address Range         |      |
| STM32H750xx   | 128k        | On-chip Flash | 00000000H - 0000FFFFH |      |
| <b>(</b>  | п           | :art:         | ize:                  |      |
| Add Remove  |             |               |                       |      |
| 设置程序download地址  |             |               |                       |      |
|   |             |               |                       |      |
|   |             |               | 确定 取消 应               | 用(A) |

更改程序存储位置

#### Startup:



```
FUNC void Setup (unsigned int region) {
   region &= 0xFFFF0000;
   SP = _RDWORD(region);
   PC = _RDWORD(region + 4);
   _wDWORD(0xE000ED08, region);
}

LOAD ".\\ETCR1CoreDriver\\ETCR1CoreDriver.axf" INCREMENTAL

Setup(0x00000000);
g, main
```

确保程序准确加载

## 注意:

1. 部分芯片 SRAM 和 FLASH 的时钟主频并不一致 ,程序中如果有含以计数器为约束条件的函数和功能需要谨慎对待。