

32 位微控制器

HC32L072 / HC32L073 / HC32F072 系列

USB 访问 SRAM 注意事项

本产品支持芯片系列如下

| | | |
|------|----------|----------|
| L 系列 | HC32L072 | HC32L073 |
| F 系列 | HC32F072 | |

注：部分型号不支持 USB 功能，详细信息请参阅数据手册。

目 录

| | | |
|---|-------------------------|---|
| 1 | 摘要..... | 3 |
| 2 | 功能介绍..... | 3 |
| 3 | USB 应用中用户自定义变量地址设置..... | 4 |
| 4 | 其他信息..... | 5 |
| 5 | 版本信息 & 联系方式..... | 6 |

1 摘要

本文件主要是描述 HC32L072 / HC32L073 / HC32F072 系列中 USB 访问 RAM 的地址范围、注意事项和如何充分利用 RAM 的空间来达到用户的开发要求。

本应用笔记主要包括：

- 功能介绍
- USB 应用中用户自定义变量地址设置

注意：

- 本应用笔记为 HC32L072 / HC32L073 / HC32F072 系列的应用补充材料，不能代替用户手册，具体功能及寄存器的操作等相关事项请以用户手册为准。

2 功能介绍

HC32L072 / HC32L073 / HC32F072 系列的 RAM 在系统映射中的地址范围如表 2-1 所示。

表 2-1 RAM 地址映射范围 and 对应容量

| 地址范围 | 容量 | Memory 类型 |
|-----------------------|----------|-----------|
| 0x20000000—0x20003FFF | 16 KByte | SRAM |

对于 HC32L072 / HC32L073 / HC32F072 系列的 USB，其访问 RAM 的地址范围为 RAM 的前 8K，其地址范围如表 2-2 所示。

表 2-2 USB 访问 RAM 的地址映射范围和容量

| 地址范围 | 容量 | Memory 类型 |
|-----------------------|---------|-----------|
| 0x20000000—0x20001FFF | 8 KByte | SRAM |

对于 HC32L072 / HC32L073 / HC32F072 系列的 RAM 的详细介绍可参考用户手册 RAM 控制器章节。

3 USB 应用中用户自定义变量地址设置

USB 访问 RAM 的范围为 0x20000000—0x20001FFF，如果超出该范围，USB 将无法访问，甚至可能无法正常运行 USB 相关的应用。因此，用户首先要确保与 USB 相关的变量在 RAM 中的地址范围内，对于收发的数组或者结构体，其起始地址和最后一个字节的地址也要求在该范围内。

目前针对用户程序中的变量，为了避免 USB 的访问范围超过其可访问范围，可以定义用户程序变量的起始地址，如：用户定义一个很大的数组，编译器自动为其分配地址的情况下，其所分配的地址可能靠前，并且当数组的大小足够大时，可能导致 USB 访问的变量在 0x20000000—0x20001FFF 之外，这种情况下，即使编译没问题，程序运行的时候 USB 也无法正常工作。对于这种情况，可以人为定义数组的起始地址，将其起始地址放在 RAM 的后面，则可以确保 USB 正常工作。

如果对于用户程序变量很多，并且各个变量大小不定的情况，则用户可以通过修改工程文件的分散加载文件，通过分散加载文件来定义用户定义的变量在 RAM 中的地址范围，确保整个工程能够正常工作。以下例子为通过分散加载文件的方式实现用户变量的范围限定。

对于 USB 样例中的 usb_customhid，如果要将在 main 中定义的所有变量放到 RAM 中地址范围为 0x20001500 开始的空间，则可以按照图 3-1 编写分散加载文件。图中除了 main.c 中的变量定义范围设置为在 0x20001500 开始的 9472 个字节范围内之外，其他文件定义的变量均在 0x20000000 开始的 8192 字节范围内。

```
*****  
; *** Scatter-Loading Description File generated by uVision ***  
*****  
  
LR_IROM1 0x00000000 0x00020000 { ; load region size_region  
ER_IROM1 0x00000000 0x00020000 { ; load address = execution address  
*.o (RESET, +First)  
*(InRoot$$Sections)  
.ANY (+RO)  
.ANY (+XO)  
}  
RW_IRAM1 0x20000000 0x00002000 { ; RW data  
.ANY (+RW +ZI)  
}  
RW_IRAM2 0x20001500 0x00002500 {  
main.o (+RW, +ZI)  
}
```

图 3-1 分散加载文件

使用分散加载文件之前，首先打开 keil 编译器中的 “Options for Target ‘usbd_customhid’” ->Linker 页面，如图 3-2 所示，将红色方框所圈的复选框取消勾选，然后在”Scatter File”编辑框选择对应的分散加载文件，本例的设置如图 3-2 所示。

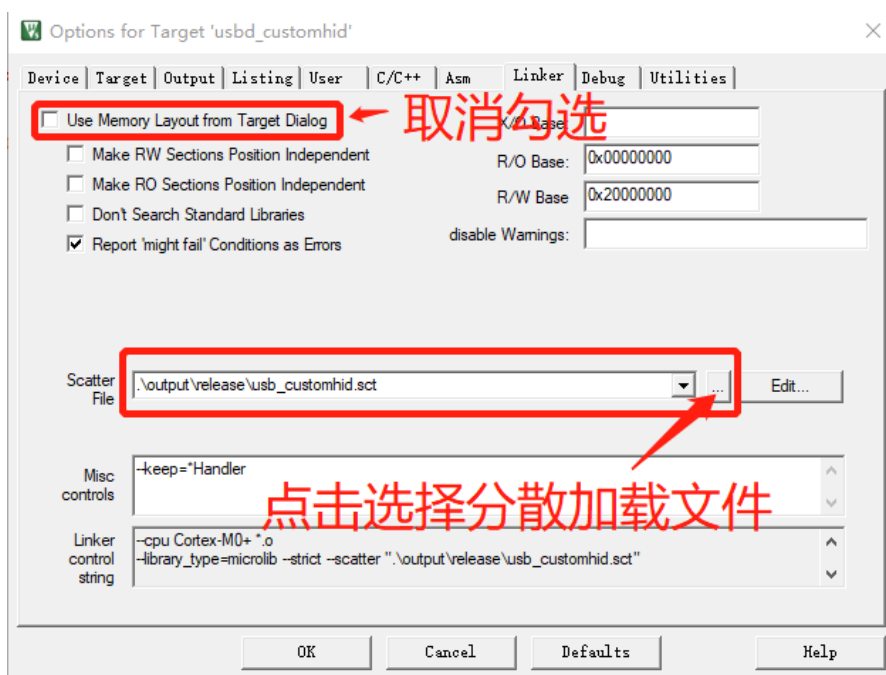


图 3-2 使用编辑后的分散加载文件的设置界面

4 其他信息

技术支持信息: <http://www.xhsc.com.cn>

5 版本信息 & 联系方式

| 日期 | 版本 | 修改记录 |
|------------|--------|-------------|
| 2020/10/16 | Rev1.0 | 初版发布。 |
| 2022/7/15 | Rev1.1 | 公司 Logo 更新。 |
| | | |



如果您在购买与使用过程中有任何意见或建议，请随时与我们联系。

Email: mcu@xhsc.com.cn

网址: <http://www.xhsc.com.cn>

通信地址: 上海市浦东新区中科路 1867 号 A 座 10 层

邮编: 201203

