

## 簡介

HolyChip 部分Flash MCU採用了Flash作類EEPROM的設計方式，MCU的ROM空間均可以用作類EEPROM資料存儲 ( IAP操作 )，使用者在資料存儲時的靈活性大大提升。

在IAP操作時，如果不做好相關的保護措施，可能會造成程式運行異常。本手冊主要介紹IAP的使用注意事項。

- 本技術手冊適用晶片：HC89S003F4、HC89S105C6、HC89S105C8、HC89S105K6、HC89S105K8、HC89F0411P、HC89F0421、HC89F0431、HC89F0531、HC89F0541、HC89F301、HC89F302、HC89F303、HC89F3421、HC89F3531、HC89F3541。
- 相關資料手冊、工具及技術文檔下載網址：<http://www.holychip.cn/>。

# 目錄

<b>1、 BOR 電壓以及 BOR 消抖時間設置.....</b>	<b>3</b>
1.1 原理說明 .....	3
1.2 BOR 電壓設置方式 .....	3
<b>2、 Option 設置 IAP 擦防寫設置.....</b>	<b>5</b>
2.1 原理說明 .....	5
2.2 操作說明 .....	5
<b>3、 IAP 操作前判斷 MCU 電壓 .....</b>	<b>8</b>
<b>4、 中斷.....</b>	<b>8</b>
<b>5、 雙區域保存存儲資料.....</b>	<b>9</b>
5.1 實現原理 .....	9
5.2 操作說明 .....	9
<b>6、 IAP 函數絕對位址編譯 函數絕對位址編譯及指標調用 .....</b>	<b>10</b>
6.1 實現原理 .....	10
6.2 操作說明 .....	10
<b>7、 版本說明.....</b>	<b>12</b>

## 1、BOR 電壓以及 BOR 消抖時間設置

### 1.1 原理說明

MCU 工作電壓為 2.0V-5.5V，如果關閉 BOR 時，MCU 在 1.5V-2.0V 可能會工作異常，程式可能會跑飛從而導致誤擦除。如果將 BOR 電壓設置為 2.0V 及 2.0V 以上，可以保證 MCU 工作在一個穩定的電壓範圍內。

### 1.2 BOR 電壓設置方式

#### BOR 電壓檢測控制寄存器 BORC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W
復位值	1	0	0	0	0	0	0	0
位元符號	BOREN	BOR_DBC_EN	-			BORVS[2:0]		
號								

位編號	位元符號	說明
7	BOREN	BOR 使能位 0：禁止 BOR 1：允許 BOR
6	BOR_DBC_EN	BOR 消抖使能位 0：不使能 1：使能
5-3	-	保留位（讀為 0，寫無效）
2-0	BORVS[2:0]	BOR 檢測電壓點選擇位元

		000 : 1.8V
		001 : 2.0V
		010 : 2.4V
		011 : 2.6V
		100 : 3.0V
		101 : 3.6V
		110 : 3.9V
		111 : 4.2V

### BOR 電壓檢測控制寄存器 BORC

位編號	7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
復位值	0	0	0	0	0	0	0	0
位元符號	BORDBC[7:0]							

位編號	位元符號	說明
7-0	BORDBC[7:0]	<p>BOR 消抖控制位</p> <p>消抖時間 = BORDBC[7:0] * 8T<sub>CPU</sub> + 2 T<sub>CPU</sub></p> <p>注：需要使能 BOR_DBC_EN，否則 BOR 不消抖。</p>

在設置 BOR 電壓以及 BOR 消抖時間時，BOR 電壓至少設置到 2.0V，在系統允許的情況下，BOR 電壓設置的越高越好並且儘量將消抖時間設置較短從而保證 MCU 工作在一個可靠的電壓環境內。

BOR 使能後的功耗約為 8uA，如使用者需要低功耗，建議在進入掉電模式之前關閉 BOR，掉電模式喚醒後再使能 BOR，從而達到節省功耗的目的。

## 2、Option 設置 IAP 擦防寫設置

### 2.1 原理說明

在 Option 中設置相關的 IAP 擦防寫，使能程式區資料保護位元，可以有效的保證程式區資料不會被改寫或誤擦除。

### 2.2 操作說明

#### 2.2.1 Keil 操作方式

以 HC89S003F4 的“Flash 做類 EEPROM 讀寫”範例為例。

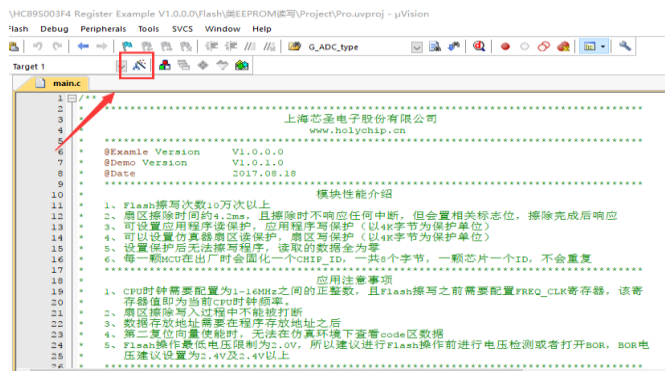


圖 2-1

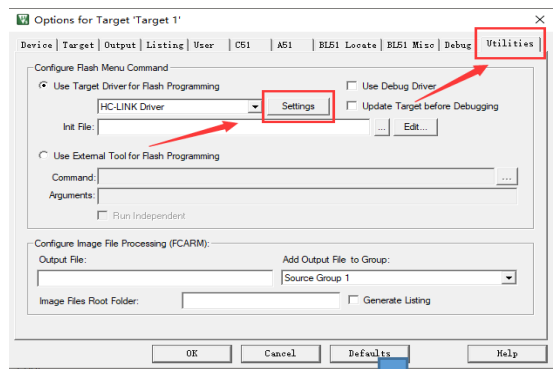


圖 2-2

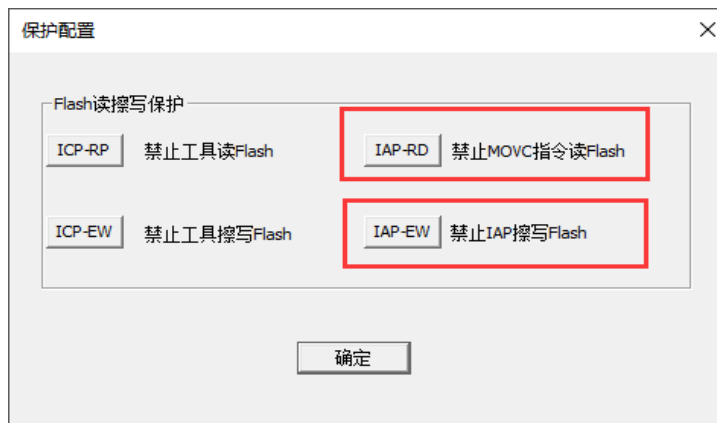


圖 2-4

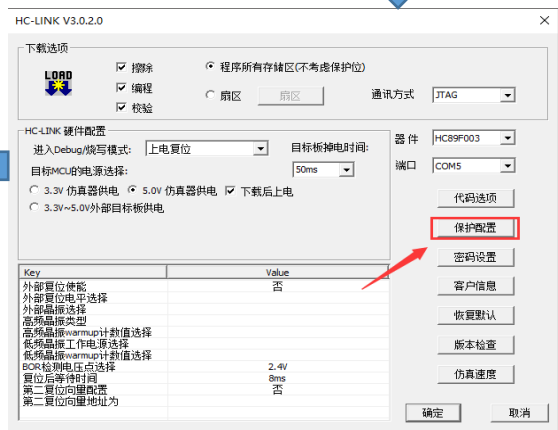


圖 2-3

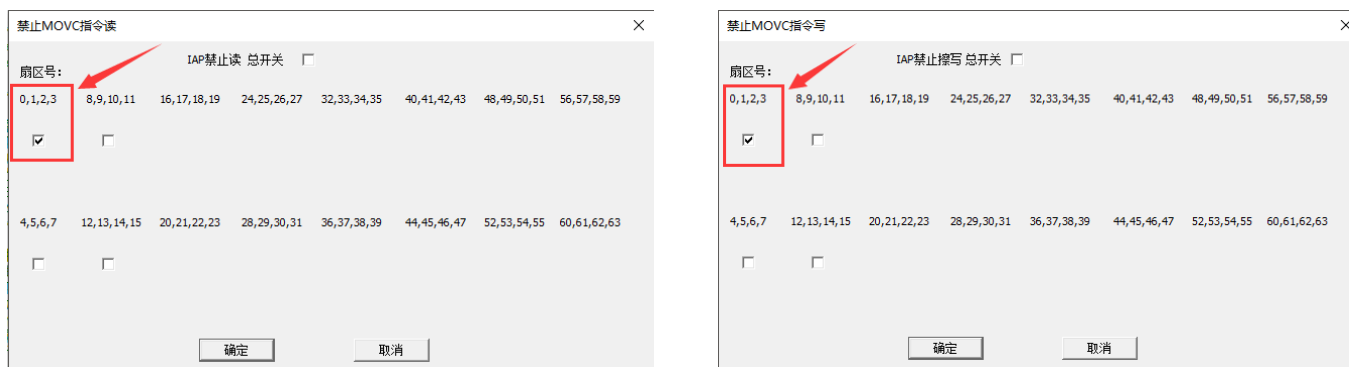


圖 2-5

### 操作步驟：

- 1、KEIL 中打開 Option。( 如圖 2-1 )
- 2、點擊 Utilities 選項卡中的 Settings 選項。( 如圖 2-2 )
- 3、點擊保護配置選項卡。( 如圖 2-3 )
- 4、分別打開 IAP-RD 以及 IAP-EW 選項卡。( 如圖 2-4 )
- 5、根據需求勾選相應的磁區後點擊確定。( 如圖 2-5 )

## 2.2.2 HC-PM51 操作方式



圖 2-6

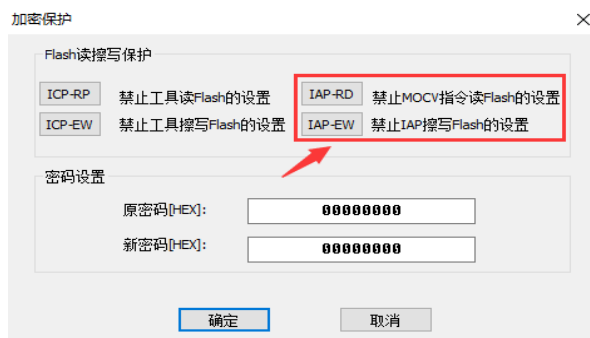


圖 2-7

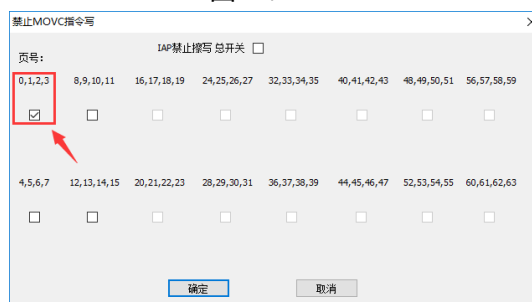
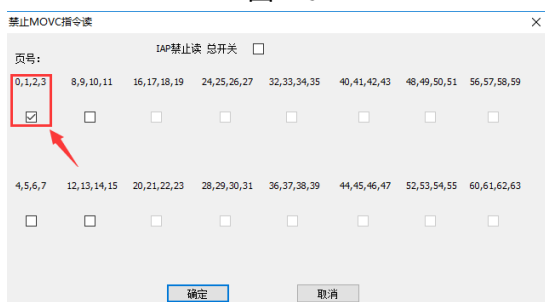


圖 2-8

操作步驟：

- 1、選擇晶片型號，配置代碼選項，載入代碼後勾選加密保護選項，點擊加密保護設置選項。( 如圖 2-6 )
- 2、分別打開 IAP-RD 以及 IAP-EW 選項卡。( 如圖 2-7 )
- 3、根據需求勾選相應的磁區後點擊確定。( 如圖 2-8 )

### 2.2.3、保護區域計算方法

```
Rebuild target 'Target 1'  
compiling main.c...  
..\user\main.c(106): warning C294: unreachable code  
linking...  
Program Size: data=28.0 xdata=0 code=585  
creating hex file from "Pro"...  
"Pro" - 0 Error(s), 1 Warning(s).  
Build Time Elapsed: 00:00:03
```

圖 2-9

HolyChip Flash 系列 MCU 均為 128 個位元組一個磁區， $8 \times 128 = 1K$  為 1 頁。

KEIL 編譯後會生成編譯位元組大小，以 HC89S003F4 的“Flash 做類 EEPROM 讀寫”範例為例，本常式編譯的位元組為 585 個 ( 如圖 2-9 )，由於 HC89S003F4 以 4K 位元組為保護為單位，所以在勾選時勾選 0、1、2、3 頁 ( 如圖 2-5 和圖 2-8 )。部分 MCU 以 1K 位元組為保護單位，使用者根據需求選擇即可。

### 3、IAP 操作前判斷 MCU 電壓

為防止 MCU 上電瞬間電壓不穩導致程式跑飛造成誤操作，所以建議每次 IAP 操作前採用 ADC 或者 LVD 檢測 MCU 當前電壓，如果電源低於 2.0V 則不進行 IAP 操作。具體的操作方式詳見各 MCU 資料手冊及常式。

### 4、中斷

IAP 操作前，建議關閉中斷 ( EA=0 )，確保在 IAP 操作期間不會被中斷影響，待 IAP 操作完成後，再將中斷恢復。



## 5、雙區域保存存儲資料

### 5.1 實現原理

在執行 IAP 操作時，不可避免的會遇到資料擦除結束後，尚未寫資料就掉電的情況，所以建議採用雙區域保存資料的方式，即使一個區域的資料被擦除，也可以保證另一個區域的資料被正常讀取。

### 5.2 操作說明

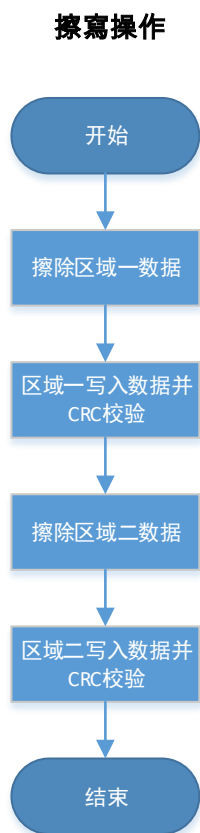


圖 5-1

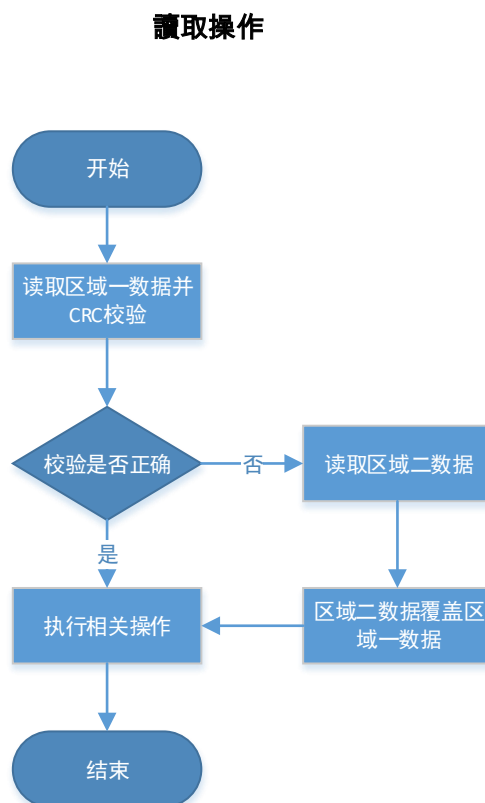


圖 5-2

**擦寫操作：**

更新資料時，先更新區域一內資料，更新完成後再更新區域二內資料。( 如圖 5-1 )

**讀取操作：**

讀取資料時，先讀取區域一內資料，並進行 CRC 校驗，如果校驗錯誤，則丟棄區域一的資料，讀取區域二內資料。( 如圖 5-2 )

## 6、 IAP 函數絕對位址編譯 函數絕對位址編譯及指標調用

### 6.1 實現原理

在 Option 中設置相關的 IAP 擦防寫，使能使用者程式所在磁區保護位元後，使用絕對位址編譯將 IAP 操作函數放到未設置 IAP 保護的區域，並使用函數指標的方式調用函數，保證 MCU 不會誤擦除用戶代碼。

### 6.2 操作說明

以 HC89S003F4 的“Flash 做類 EEPROM 讀寫”範例為例，

- 1、 打開 Flash 做類 EEPROM 讀寫，在主函數裡分別找到 FLASH 擦除(Flash\_EraseBlock)和寫 (FLASH\_WriteData)的函數，打開 Project -> Options for Target 功能表，選中 BL51 Locate 選項卡，在 Code: 中輸入: ?PR?\_Flash\_EraseBlock?MAIN(0X2000),?PR?\_FLASH\_WriteData?MAIN(0X2080)。格式為: ?PR?\_函數名?MAIN(要編譯的位址)，注意，輸入的文本 ( 包括標點符號 ) 必須是英文格式，否則編譯器會報錯，然後點擊 OK，程式編譯後，在 M51 檔中可以找到相關資訊，如 圖 6-3 所示程式已經被編譯在相關地址處。

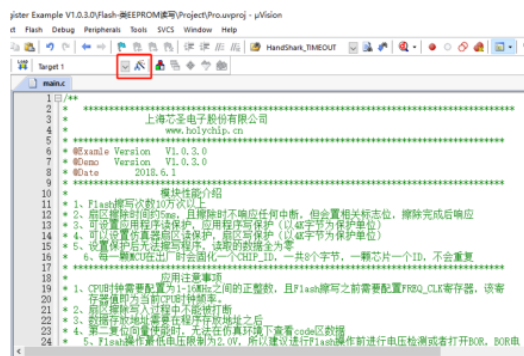


圖 6-1

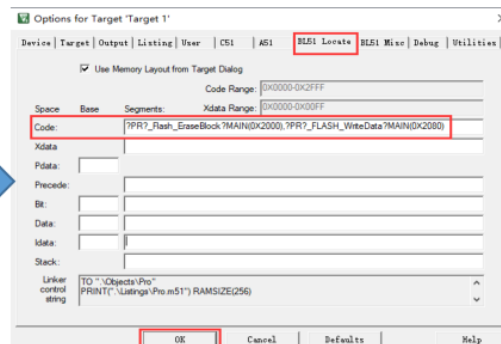


圖 6-2

```

*****  C O D E   M E M O R Y  *****
CODE  0000H  0003H  ABSOLUTE  ?C_INITSEG
CODE  0003H  0019H  UNIT      *** GAP ***
CODE  001CH  0007H  ABSOLUTE
CODE  0023H  0003H  ABSOLUTE  ?PR?MAIN?MAIN
CODE  0026H  00BAH  UNIT      ?C_C51STARTUP
CODE  00E0H  008CH  UNIT      ?PR?FLASH_WRITEARR?MAIN
CODE  016CH  0044H  UNIT      ?PR?UART1_FPT?MAIN
CODE  01B0H  0037H  UNIT      ?PR?FLASH_READARR?MAIN
CODE  01E7H  0037H  UNIT      ?C?LIB_CODE
CODE  021EH  0031H  UNIT      *** GAP ***
CODE  024FH  1DB1H  UNIT
CODE  2000H  0022H  UNIT      ?PR?FLASH_ERASEBLOCK?MAIN
CODE  2022H  005EH  UNIT      *** GAP ***
CODE  2080H  0024H  UNIT      ?PR?FLASH_WRITEDATA?MAIN

```

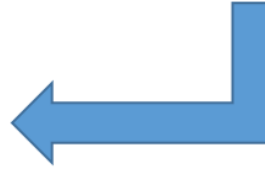


圖 6-3

## 2、使用函數指標調用絕對位址處的函數，詳情請參考常式

- (1) 定義函數原型：void Flash\_EraseBlock(unsigned int fui\_Address)；
- (2) 定義相應的函數指標變數：void (\*CALL\_FLASH\_ERASE)(unsigned int i)；
- (3) 函數指標變數賦值，指向我們定位的絕對位址的函數：CALL\_FLASH\_ERASE=Flash\_EraseBlock；
- (4) 函數指標的調用：(\*CALL\_FLASH\_ERASE)(0x2B00)；

## 7、版本說明

版本	日期	描述
V1.00	2018/1/25	初版
V1.01	2018/6/26	1、 增加相關型號 2、 修改部分錯誤

HOLYCHIP 公司保留對以下所有產品在可靠性、功能和設計方面的改進作進一步說明的權利。HOLYCHIP 不承擔由本手冊所涉及的产品或電路的運用和使用所引起的任何責任，HOLYCHIP 的產品不是專門設計來應用於外科植入、生命維持和任何 HOLYCHIP 產品產生的故障會對個體造成傷害甚至死亡的領域。如果將 HOLYCHIP 的產品用於上述領域，即使這些是由 HOLYCHIP 在产品設計和製造上的疏忽引起的，用戶應賠償所有費用、損失、合理的人身傷害或死亡所直接或間接所產生的律師費用，並且用戶保證 HOLYCHIP 及其雇員、子公司、分支機構和銷售商與上述事宜無關。

芯聖電子

2018 年 6 月