

简介

本文档介绍了 HC89S 系列中存储器模块的基本功能，以及使用该模块时的注意事项。在实际的开发过程中，如需更深一步了解该模块的基本功能以及操作事项，可以参考芯片手册中存储器模块的详细介绍。芯片手册中的例程为用户进一步的学习芯片提供参考，该例程也可以应用到实际的开发中。

- 本文档为 HC89S 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。
- 相关数据手册、工具及技术文档下载网址：<http://www.holychip.cn/>。

目录

1	功能介绍.....	3
2	FLASH 数据安全.....	3
3	OPTION.....	4
4	FLASH IAP 操作	4
5	FLASH ICP 操作.....	5
6	第二复位向量操作.....	6
7	数据存储器（RAM）	7
8	特殊功能寄存器（SFR）	8
	8.1 特殊功能寄存器列表.....	8
9	性能介绍.....	12
10	参考例程.....	12
11	其他信息.....	12

1 功能介绍

- 在工作电压范围内都能进行擦除和编程操作
- 在线编程（ICP）操作支持写入、读取和擦除操作
- ICP 操作可设置 32 位密码进行保护
- 在应用编程（IAP）支持用户自定义启动代码和类 EEPROM 区
- 灵活的代码保护模式
- 擦写次数至少 10 万次
- 数据保存年限至少 10 年
- 128 字节为 1 个扇区，8 个扇区为 1 页（1KB）

2 FLASH 数据安全

FLASH 的操作可以分成两种：第一种是通过工具（仿真器、烧录器）对 FLASH 进行读、擦、写操作，这种方式被称为在电路编程模式（ICP）；第二种是用户程序代码对 FLASH 其他扇区进行读、擦、写操作，这种方式被称为在应用编程模式（IAP）。

2.1 ICP 操作密码保护

用户可以通过上位机软件对 ICP 操作进行密码保护，密码长度为 4 个字节（32 位），一旦用户设置了密码，那么只有输入正确的密码才能进行 ICP 操作，否则就不能对 FLASH 进行任何操作，这样可以有效保护用户的程序代码。

2.2 ICP 读擦写 FLASH 保护

ICP 的读保护以 1K 字节为保护单位，当一个 1K 字节空间的读保护使能时，ICP 读这个 1K 字节空间，读出来的数据为全 0，但是仍然可以通过 ICP 操作进行仿真。

ICP 的擦写保护也是以 1K 字节为保护单位，当对应 1K 字节的擦写保护使能时，ICP 将不能擦除和编程这个 1K 字节空间，强写也不允许。

若对应 1K 字节空间读保护被使能，但被允许擦除与写入，则可先擦除后获得该 1K 字节空间的读允许，直至复位或掉电。

ICP 的读擦写保护通过上位机软件来配置，详细情况请参见 HC-LINK 用户手册。

2.3 IAP 读擦写 FLASH 保护

IAP 通过 MOV C 指令来读 FLASH，IAP 读保护以 4K 字节为单位，如果一个 4K 字节空间设置了读保护，其他 4K 字节空间的 MOV C 指令读这个 4K 字节空间，读出来的数据为零，但这个 4K 字节空间的 MOV C 指令可以读取自身的数据。

IAP 擦写 FLASH 的步骤详见 FLASH IAP 操作，IAP 的擦写保护以 1K 字节为单位，IAP 擦写之前需要先看相应扇区的擦写保护是否使能，没有使能擦写保护才能进行 IAP 的擦写。

若对应 1K 字节空间读保护被使能，但被允许擦除与写入，则可先擦除后获得该 1K 字节空间的读允许，直至复位或掉电。

IAP 的读擦写保护通过上位机软件来配置，详细情况请参见 HC-LINK 用户手册。

3 OPTION

在 64K 的 ROM 之外有一个只读的 OPTION 区域，存放的内容包括：用户定义的一些数据、用户设置的密码、芯片的一些配置、第二复位向量相关的内容。具体地址分配如下表。

地址	名称	地址偏移量	名称	地址	名称	地址偏移量	名称
0x0000	SN_DATA0	0x0020	FLASH_SC0	0x0030	ERST_SEL	0x0100	CHIP_ID0
0x0001	SN_DATA1	0x0021	FLASH_SC1	0x0031	ERST_ENB	0x0101	CHIP_ID1
0x0002	SN_DATA2	0x0022	FLASH_SC2	0x0032	XTAL_CFG	0x0102	CHIP_ID2
0x0003	SN_DATA3	0x0023	FLASH_SC3	0x0038	WAIT_TS	0x0103	CHIP_ID3
0x0004	SN_DATA4	-	-	0x0039	BORVS	0x0104	CHIP_ID4
0x0005	SN_DATA5	-	-	0x003E	RVCFG	0x0105	CHIP_ID5
0x0006	SN_DATA6	-	-	0x003F	nRVCFG	0x0106	CHIP_ID6
0x0007	SN_DATA7	-	-	-	-	0x0107	CHIP_ID7
0x0008	ID_DATA0	-	-	-	-	-	-
0x0009	ID_DATA1	-	-	-	-	-	-
0x000A	ID_DATA2	-	-	-	-	-	-
0x000B	ID_DATA3	-	-	-	-	0x0128	rc32m_trim
0x000C	ID_DATA4	-	-	-	-	0x012C	rc24m_trim
0x000D	ID_DATA5	-	-	-	-	-	-
0x000E	ID_DATA6	-	-	-	-	-	-
0x000F	ID_DATA7	-	-	-	-	-	-

HC89S105A 在出厂时都会固化一个 CHIP_ID，一共 8 个字节，一颗芯片一个 ID，不会重复，用户可以在程序中通过 MOVC 来读出，也可以通过工具读出。

SN_DATA 和 ID_DATA 是用户自定义数据，FLASH_SC 为客户密码，通过工具软件进行设置，如同设置代码选项一样，它们是可以被擦除和修改的，用户也可以在程序中通过 MOVC 来读出。

注意：1、用户在进行读 OPTION 操作前，需要将寄存器 INSCON[IAPS]位置 1。

2、首字母为“n”位对应数据的反码。

4 FLASH IAP 操作

HC89S105A 的 FLASH 一共有 512 个扇区，128 个字节为一个扇区， $8 \times 128 \text{ Bytes} = 1\text{K Bytes}$ 为一页， $1\text{K Bytes} \times 4 = 4\text{K Bytes}$ 为一块。

IAP 写之前必须先进行一次擦除操作，IAP 一次擦除一个扇区（128 字节），IAP 擦除时的地址寄存器可以是扇区里的任意地址。IAP 写是单字节写，每次写一个字节。

IAP 擦除一个扇区的时间是 5ms，在 CPU_CLK 为 16MHz 时，IAP 写一个字节的时间是 23μs，在 CPU_CLK 为 2MHz 时，IAP 写一个字节的时间是 37μs。

4.1 IAP 操作注意事项

HC89S105A 的用户程序代码可对 FLASH 进行读、擦、写操作，作为用户更新代码或存储数据使用，为保证用户对 FLASH 操作的安全性，使用过程中请注意：

1、在进行 FLASH 的 IAP 擦写之前，需要配置扩展 SFR 里 FREQ_CLK 寄存器，指明目前 CPU

时钟的频率，FREQ_CLK 寄存器配置的值等于 CPU 时钟的频率值，最小为 1MHz，假如目前 CPU 的运行频率为 16MHz，那就配置寄存器 FREQ_CLK=0x10。建议在 IAP 擦写之前，将 CPU 时钟频率分频为整数。当 CPU 时钟频率低于 1MHz 时，不能进行 FLASH 的 IAP 擦写操作。

- 2、系统在进行 IAP 操作时，不响应任何中断。
- 3、在 Option 中设置相关的 IAP 擦写保护，使用户程序所在扇区保护位，可以有效保证程序区不会被改写或误擦除。
- 4、IAP 擦写操作前，建议关闭中断（EA=0），确保在 IAP 操作期间不会被中断影响，待 IAP 擦写操作完成后，再将中断恢复。
- 5、在执行 IAP 操作时，不可避免的会遇到数据擦除结束后，尚未写数据就掉电的情况，所以建议采用双区域保存数据的方式，即使一个区域的数据被擦除，也可以保证另一个区域的数据被正常读取。

5 FLASH ICP 操作

5.1 JTAG 方式

用户可以通过 HC-LINK 仿真器对 MCU 进行编程，当 MCU 已经焊在用户板上后，如果用户采用上电复位的方式进入 JTAG，只需要链接 6 根线，用户系统必须断电，由仿真器提供电源。当用户系统不希望掉电时，可以采用 7 根线进入编程模式，多了一个复位引脚，仿真器更详细的使用说明请参见 HC-LINK 用户手册。

另外，因为编程信号非常敏感，用户需要用 5 个跳线将编程引脚（VDD、TDO、TDI、TMS、TCK）从应用电路中分离出来，如下图所示。

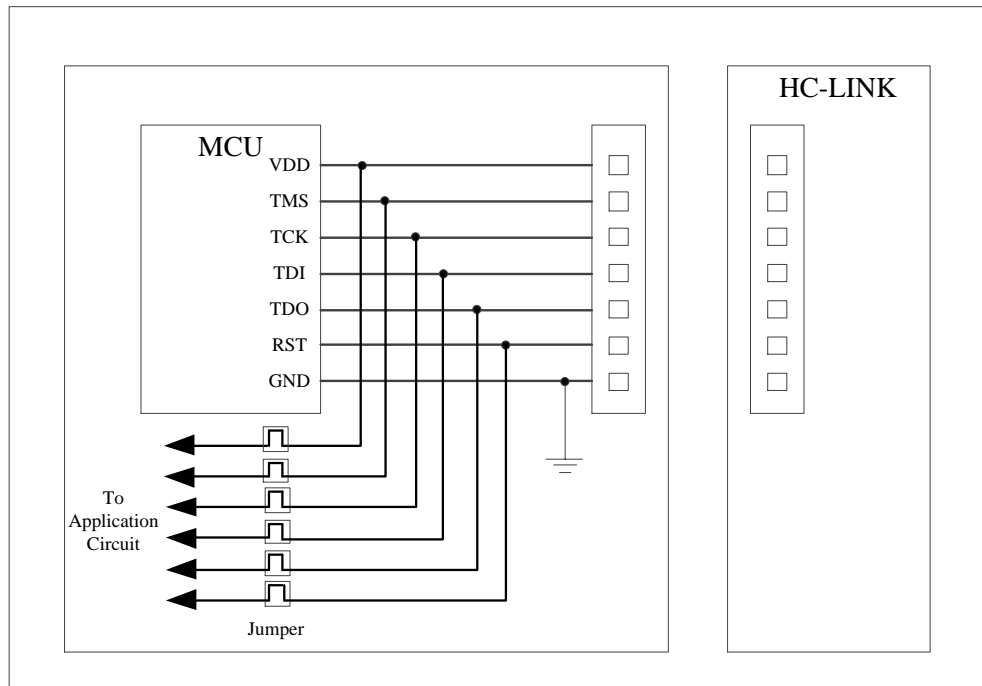


Figure 5-1 HC-LINK 编程硬件连接

5.2 双线方式

用户可以通过 HC-LINK 仿真器的双线方式对 MCU 进行仿真和编程，当 MCU 已经焊在用户板上后，如果用户采用上电复位的方式，只需要连接四根线（VDD、GND、SDA、SCK），用户系统必须断电，由仿真器提供电源。当用户系统不希望掉电时，可以采用五根线进入编程模式，多了一个复位引脚，仿真器更详细的使用说明请参见 HC-LINK 用户手册。

另外，因为编程信号非常敏感，用户需要用 4 个跳线将编程引脚（VDD、SDA、SCK、RST）从应用电路中分离出来，如下图所示。另外，如果使用外部复位引脚进入，也需要将外部复位引脚进行跳线分离。

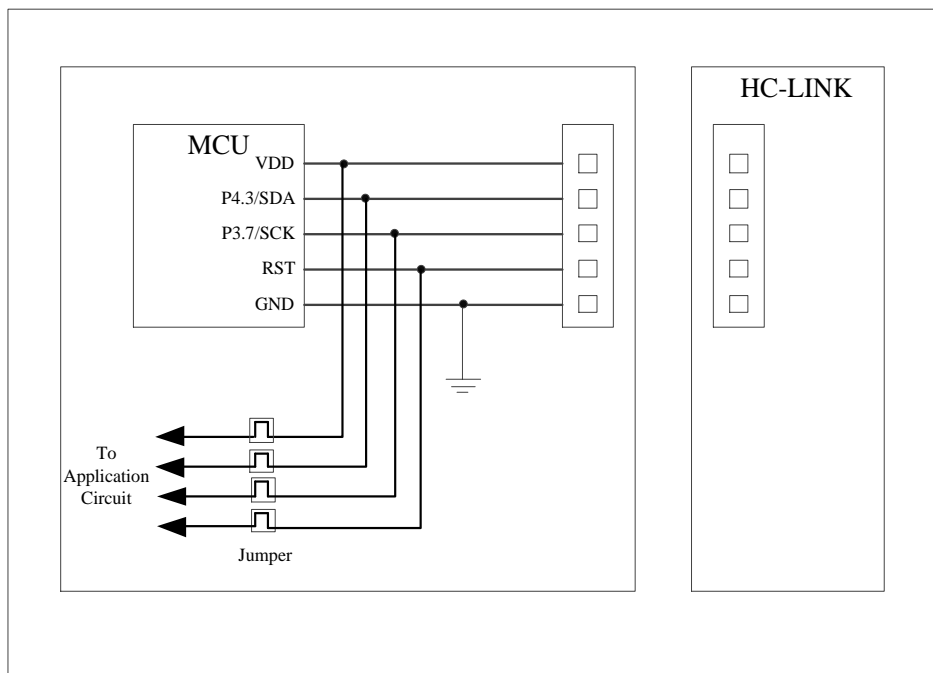


Figure 5-2 HC-LINK 编程硬件连接

当采用 ICP 模式进行操作时，建议按照如下步骤进行操作：

- 1、在开始编程前断开跳线（Jumper），从应用电路中分离编程引脚。
- 2、将芯片编程引脚连接至 Flash 编程器接口，开始编程。
- 3、编程结束后断开 Flash 编程器接口，连接跳线恢复应用电路。

6 第二复位向量操作

如果用户在代码选项中配置了第二复位向量使能和第二复位向量地址，那么芯片上电复位后，PC 会首先指向第二向量地址，开始执行用户的启动程序，用户启动程序的最后需要放置一条不重读代码选项的软件复位程序，那用户就会复位到 0x0000H 处，开始执行用户应用程序。

7 数据存储器（RAM）

HC89S105A 为用户提供了 256 Bytes 内部 RAM 和 4K Bytes 内部扩展 RAM 来作为数据存储器。下图为数据存储器空间分配。



Figure 7-1 数据存储器示意图

内部 RAM 的高 128 Bytes（0x80 ~ 0xFF）必须采用寄存器间接寻址方式。

内部扩展 RAM（XRAM）的地址范围是 0x000~0xFFFF，访问内部扩展 RAM 的方法与传统 8051 单片机访问外部扩展 RAM 的方法相同，但是不影响 I/O 口。在汇编语言中，内部扩展 RAM 通过 MOVX 指令访问，即 MOVX @DPTP 或者 MOVX @Ri。

8 特殊功能寄存器（SFR）

8.1 特殊功能寄存器列表

8.1.1 直接寻址读写 SFR

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
F8	RSTFR	IAP_ADDRL	IAP_ADDRH	IAP_DATA	IAP_CMDL	IAP_CMDH	-	-
F0	B	-	PWM2C	PWM2PL	PWM2PH	PWM2DL	PWM2DH	PWM2DTL
E8	-	PWMFLT	PWM1C	PWM1PL	PWM1PH	PWM1DL	PWM1DH	PWM1DTL
E0	ACC	PWMEN	PWM0C	PWM0PL	PWM0PH	PWM0DL	PWM0DH	PWM0DTL
D8	-	MCLDIVA0	MCLDIVA1	MCLDIVA2	MCLDIVA3	MCLDIVA4	MCLDIVA5	MCLDIVC
D0	PSW	LCDCON	T5CON	TL5	TH5	RCAP5L	RCAP5H	-
C8	P5	PCACLK	PCAMOD0	PCAMOD1	CCAPL0	CCAPH0	CCAPL1	CCAPH1
C0	P4	PCACON	PCACL	PCACH		PWM0DTH	PWM1DTH	PWM2DTH
B8	IE1	IP2	IP3	LVDC	T6CON	WDTC	CRCL	CRCH
B0	P3	IP4	IE2	LVDCMP	ADCC0	ADCC1	ADCRL	ADCRH
A8	IE	IP0	IP1	SPDAT	SPCTL	SPSTAT	IICDAT	IICADR
A0	P2	T4CON		INSCON	TL4	TH4	IICCON	IICSTA
98	SCON	SBUF	SADDR	SADEN	SBRTL	SBRTH	SCON2	PWM3C
90	P1	T3CON	TL3	TH3	-	-	PINTF0	PINTF1
88	TCON	TMOD	TL0	TL1	TH0	TH1	CLKSWR	CLKCON
80	P0	SP	DPL	DPH	PWM3P	PWM3D	-	PCON

8.1.2 外部扩展 XSFR

扩展 XSFR 采用和 XRAM 同样的访问方式，使用 MOVX A, @DPTR 和 MOVX @DPTR, A 来进行读写。

比如写一个地址为 0xFE88 的 XSFR，操作如下：

```
MOV    A, #wdata
MOV    DPTR, #0xFE88
MOVSX @DPTR, A
```

读地址为 0xFE89 的 XSFR，操作如下：

```
MOV    DPTR, #0xFE89
MOVSX A, @DPTR
```

使用 C 语言来编程时，只需要 #define ALLOCATE_EXTERN，并且 #include "HC89S105A.h"，就可以像操作直接寻址寄存器一样，直接赋值 XSFR，比如：

```
ADCC2 = 0x4D;
```


扩展XSFR（基地址0xFE80）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	TCON1	0x0010	-	0x0020	WDTCCR	0x0030	PITS0
0x0001	T4CON1	0x0011	CLKDIV	0x0021	-	0x0031	PITS1
0x0002	T5CON1	0x0012	FREQ_CLK	0x0022	CRCC	0x0032	PITS2
0x0003	T5CON2	0x0013	CLKOUT	0x0023	-	0x0033	PITS3
0x0004	PCA_PWM0	0x0014	UART_CLKS	0x0024	BORC	0x0034	-
0x0005	PCA_PWM1	0x0015	SPOV	0x0025	BORDBC	0x0035	-
0x0006	-	0x0016	-	0x0026	-	0x0036	-
0x0007	-	0x0017	PORB_IAPF	0x0027	LVDDBC	0x0037	-
0x0008	S2CON	0x0018		0x0028	-	0x0038	PINTE0
0x0009	S2CON2	0x0019	AWDCON	0x0029	-	0x0039	PINTE1
0x000A	S2BUF	0x001A	ADCC3	0x002A	RSTDBC	0x003A	
0x000B	S2ADDR	0x001B	ADCC2	0x002B		0x003B	
0x000C	S2ADEN	0x001C	PWM0DBC	0x002C	CLKPCKEN0	0x003C	
0x000D	S2BRTH	0x001D	PWM1DBC	0x002D	CLKPCKEN1	0x003D	TRMEN
0x000E	S2BRTL	0x001E	PWM2DBC	0x002E	ADCCONTV	0x003E	TRMV
0x000F	-	0x001F	-	0x002F	ADCGAPV	0x003F	-

扩展XSFR（基地址0xFEC0）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	SCRH0	0x0010	CAPH0	0x0020	PWM0PHASEH	0x0030	PWM0CMPH
0x0001	SCRL0	0x0011	CAPL0	0x0021	PWM0PHASEL	0x0031	PWM0CMPL
0x0002	SCRH1	0x0012	CAPH1	0x0022	PWM1PHASEH	0x0032	-
0x0003	SCRL1	0x0013	CAPL1	0x0023	PWM1PHASEL	0x0033	-
0x0004	SCRH2	0x0014	-	0x0024	PWM2PHASEH	0x0034	-
0x0005	SCRL2	0x0015	-	0x0025	PWM2PHASEL	0x0035	-
0x0006	SCRH3	0x0016	-	0x0026	-	0x0036	-
0x0007	SCRL3	0x0017	-	0x0027	-	0x0037	-
0x0008	SCRH4	0x0018	CAPCON0	0x0028	PWM0INTDIV	0x0038	-
0x0009	SCRL4	0x0019	CAPCON1	0x0029	PWM1INTDIV	0x0039	PWMCON0
0x000A	SCRH5	0x001A	ADCHTRH	0x002A	PWM2INTDIV	0x003A	PWMCON1
0x000B	SCRL5	0x001B	ADCHTRL	0x002B	-	0x003B	PWMCON2
0x000C	SCRH6	0x001C	ADCLTRH	0x002C	-	0x003C	PWMCON3
0x000D	SCRL6	0x001D	ADCLTRL	0x002D	-	0x003D	PWMCON4
0x000E	SCRH7	0x001E	ADCPLYH	0x002E	PWM3CLKS	0x003E	-
0x000F	SCRL7	0x001F	ADCPLYL	0x002F	-	0x003F	-

扩展XSFR（基地址0xFF00）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	P0M0	0x0010	P2M0	0x0020	P4M0	0x0030	-
0x0001	P0M1	0x0011	P2M1	0x0021	P4M1	0x0031	-
0x0002	P0M2	0x0012	P2M2	0x0022	P4M2	0x0032	-
0x0003	P0M3	0x0013	P2M3	0x0023	P4M3	0x0033	-
0x0004	-	0x0014	-	0x0024	-	0x0034	-
0x0005	-	0x0015	-	0x0025	-	0x0035	-
0x0006	-	0x0016	-	0x0026	-	0x0036	-
0x0007	-	0x0017	-	0x0027	-	0x0037	-
0x0008	P1M0	0x0018	P3M0	0x0028	P5M0	0x0038	-
0x0009	P1M1	0x0019	P3M1	0x0029	P5M1	0x0039	-
0x000A	P1M2	0x001A	P3M2	0x002A	P5M2	0x003A	-
0x000B	P1M3	0x001B	P3M3	0x002B	-	0x003B	-
0x000C	-	0x001C	-	0x002C	-	0x003C	-
0x000D	-	0x001D	-	0x002D	-	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	-	0x001F	-	0x002F	-	0x003F	-

扩展XSFR（基地址0xFF40）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	P00DBC	0x0010	P0OUT	0x0020	COMP0EN	0x0030	P0DRENH
0x0001	P01DBC	0x0011	P1OUT	0x0021	COMP1EN	0x0031	P0DRENH
0x0002	P02DBC	0x0012	P2OUT	0x0022	COMP2EN	0x0032	P1DRENH
0x0003	-	0x0013	P3OUT	0x0023	COMP3EN	0x0033	P1DRENH
0x0004	-	0x0014	P4OUT	0x0024	COMP4EN	0x0034	P2DRENH
0x0005	-	0x0015	P5OUT	0x0025	COMP5EN	0x0035	P2DRENH
0x0006	-	0x0016	-	0x0026	-	0x0036	P3DRENH
0x0007	-	0x0017	-	0x0027	-	0x0037	P3DRENH
0x0008	-	0x0018	-	0x0028	SEGP0EN	0x0038	P4DRENH
0x0009	-	0x0019	-	0x0029	SEGP1EN	0x0039	P4DRENH
0x000A	-	0x001A	-	0x002A	SEGP2EN	0x003A	P5DRENH
0x000B	-	0x001B	-	0x002B	SEGP3EN	0x003B	P5DRENH
0x000C	-	0x001C	-	0x002C	SEGP4EN	0x003C	PDREN_SEL
0x000D	-	0x001D	-	0x002D	SEGP5EN	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	-	0x001F	-	0x002F	-	0x003F	-

扩展XSFR（基地址0xFF80）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	T0_MAP	0x0010	PWM0_MAP	0x0020	TXD_MAP	0x0030	INT0_MAP
0x0001	T1_MAP	0x0011	PWM01_MAP	0x0021	RXD_MAP	0x0031	INT1_MAP
0x0002	T3_MAP	0x0012	FLT0_MAP	0x0022	SCL_MAP	0x0032	-
0x0003	T4_MAP	0x0013	-	0x0023	SDA_MAP	0x0033	-
0x0004	T6_MAP	0x0014	PWM1_MAP	0x0024	\overline{SS} _MAP	0x0034	-
0x0005	BRTO_MAP	0x0015	PWM11_MAP	0x0025	SCK_MAP	0x0035	-
0x0006	T5_MAP	0x0016	FLT1_MAP	0x0026	MOSI_MAP	0x0036	-
0x0007	-	0x0017	-	0x0027	MISO_MAP	0x0037	-
0x0008	CAP0_MAP	0x0018	PWM2_MAP	0x0028	TXD2_MAP	0x0038	-
0x0009	CAP1_MAP	0x0019	PWM21_MAP	0x0029	RXD2_MAP	0x0039	-
0x000A	ECI_MAP	0x001A	FLT2_MAP	0x002A	-	0x003A	-
0x000B	PCA0_MAP	0x001B	-	0x002B	-	0x003B	-
0x000C	PCA1_MAP	0x001C	PWM3_MAP	0x002C	-	0x003C	-
0x000D	ADCST_MAP	0x001D	-	0x002D	-	0x003D	-
0x000E	-	0x001E	-	0x002E	-	0x003E	-
0x000F	CLKO_MAP	0x001F	-	0x002F	-	0x003F	-

只读：

扩展XSFR（基地址0xFFC0）

偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称	偏移地址	XSFR 名称
0x0000	SN_DATA0	0x0010	CHIP_ID0	0x0020	-	0x0030	-
0x0001	SN_DATA1	0x0011	CHIP_ID1	0x0021	-	0x0031	-
0x0002	SN_DATA2	0x0012	CHIP_ID2	0x0022	-	0x0032	-
0x0003	SN_DATA3	0x0013	CHIP_ID3	0x0023	-	0x0033	-
0x0004	SN_DATA4	0x0014	CHIP_ID4	0x0024	-	0x0034	-
0x0005	SN_DATA5	0x0015	CHIP_ID5	0x0025	-	0x0035	-
0x0006	SN_DATA6	0x0016	CHIP_ID6	0x0026	-	0x0036	-
0x0007	SN_DATA7	0x0017	CHIP_ID7	0x0027	-	0x0037	-
0x0008	ID_DATA0	0x0018	-	0x0028	-	0x0038	-
0x0009	ID_DATA1	0x0019	-	0x0029	-	0x0039	-
0x000A	ID_DATA2	0x001A	-	0x002A	-	0x003A	-
0x000B	ID_DATA3	0x001B	-	0x002B	-	0x003B	-
0x000C	ID_DATA4	0x001C	-	0x002C	-	0x003C	-
0x000D	ID_DATA5	0x001D	-	0x002D	-	0x003D	-
0x000E	ID_DATA6	0x001E	-	0x002E	-	0x003E	-
0x000F	ID_DATA7	0x001F	-	0x002F	-	0x003F	-

HC89S105A 在出厂时都会固化一个 CHIP_ID，一共 8 个字节，一颗芯片一个 ID，不会重复，用户可以在程序中像读 XSFR 一样可以直接读出。

SN_DATA 和 ID_DATA 是用户自定义数据，通过工具软件进行设置，如同设置代码选项一样，它们是可以被擦除和修改的，用户也可以在程序中像读 XSFR 一样可以直接读出。

9 性能介绍

参数	符号	条件	最小值	典型值	最大值	单位
读写测试	NENDUR	-	100000	-	-	Cycle
数据保存时间	TRET	T=25°C	-	10	-	year
扇区擦除时间	TERASE	1 个扇区 (128 字节)	-	5	-	ms
字节写入时间	T _{PROG}	1 个字节, F _{cpu} =16MHz	-	23	-	us
读取耗电流	I _{DD1}	F _{cpu} =16MHz	-	4	-	mA
写入耗电流	I _{DD2}	-	-	4	-	mA
擦除耗电流	I _{DD3}	-	-	2	-	mA

10 参考例程

芯圣（Holychip）官方提供了存储器模块的参考例程，用户可通过例程进一步学习和使用该模块，在实际的应用开发中也可以直接参考例程快速对该模块进行操作。

11 其他信息

技术支持信息： www.holychip.cn

HOLYCHIP 公司保留对以下所有产品在可靠性、功能和设计方面的改进作进一步说明的权利。HOLYCHIP 不承担由本手册所涉及的产品或电路的运用和使用所引起的任何责任，HOLYCHIP 的产品不是专门设计来应用于外科植入、生命维持和任何 HOLYCHIP 产品产生的故障会对个体造成伤害甚至死亡的领域。如果将 HOLYCHIP 的产品用于上述领域，即使这些是由 HOLYCHIP 在产品设计和制造上的疏忽引起的，用户应赔偿所有费用、损失、合理的人身伤害或死亡所直接或间接所产生的律师费用，并且用户保证 HOLYCHIP 及其雇员、子公司、分支机构和销售商与上述事宜无关。

芯圣电子

2022 年 6 月