

32 位微控制器

HC32L110 / HC32F003 / HC32F005 系列的 Flash 串行编程

适用对象

系列	产品型号
HC32L110	HC32L110C6UA
	HC32L110C6PA
	HC32L110C4UA
	HC32L110C4PA
	HC32L110B6PA
	HC32L110B4PA
HC32F003	HC32F003C4UA
	HC32F003C4PA
HC32F005	HC32F005C6UA
	HC32F005C6PA
	HC32F005D6UA

目 录

1	摘要	3
2	Bootloader 简介	3
3	串行编程模式	4
3.1	通信标准	4
3.2	硬件连接	4
4	Flash 编程流程	5
4.1	下载并运行 RAMCODE	6
4.1.1	连接命令	7
4.1.2	下载 RAMCODE	7
4.1.3	执行命令	8
4.1.4	全擦除命令	8
4.2	更改波特率设置	9
4.3	Flash 操作	10
4.3.1	Flash 空白检测	11
4.3.2	Flash 写	12
4.3.3	读 Flash 校验和	13
4.3.4	Flash 加密设置	14
4.4	错误处理	15
	附录 1 : RAMCODE API 接口	16
	命令帧	16
	ACK 帧	17
	1) 串口波特率设置	18
	2) Flash 片擦除	19
	3) Flash 写	19
	4) 读 Flash 校验和	20
	5) Flash 空白检查	21
	6) Flash 加密设置	21
5	总结	22
6	版本信息 & 联系方式	23

1 摘要

本文主要介绍使用串行编程模式对 HC32L110 / HC32F003 / HC32F005 系列内部 Flash 进行编程的步骤和注意事项。

2 Bootloader 简介

为了方便客户对 HC32L110 / HC32F003 / HC32F005 系列产品的内部 Flash 进行串行编程，Bootloader 的设计实现了各种接口供客户进行开发。

3 串行编程模式

3.1 通信标准

串行编程模式采用标准 UART（Universal Asynchronous Receiver / Transmitter）通信，1 个起始位，8 个数据位，1 个停止位。

3.2 硬件连接

对 HC32L110 / HC32F003 / HC32F005 系列 MCU 的内部 Flash 进行串行编程的硬件条件是：

- 1) 在可运行的最小系统环境内。
- 2) 编程器端的信号线需要连接 VCC、GND、TX、RX 四条信号线如图 1 所示（以 HC32L110 系列为例）。

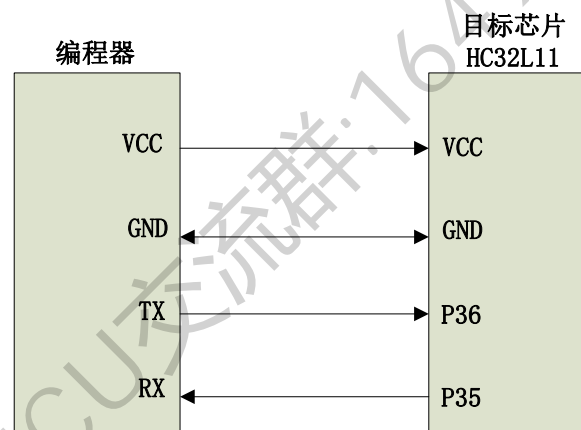


图 1 串行编程连接

4 Flash 编程流程

对 Flash 进行串行编程的总流程如图 2 所示。编程器采用波特率 9600 对目标芯片进行 RAMCODE 下载和波特率重设，然后使用重设的波特率进行 Flash 操作。

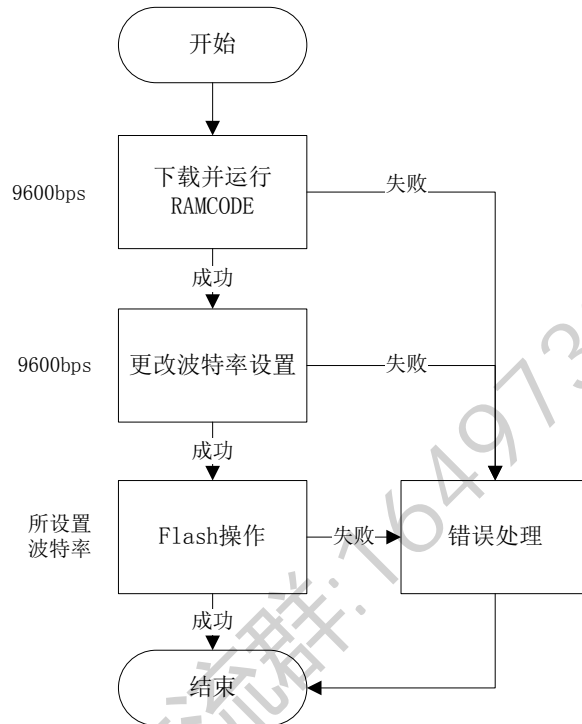


图 2 串行编程总流程

4.1 下载并运行 RAMCODE

下载并运行 RAMCODE 流程如图 3 所示。

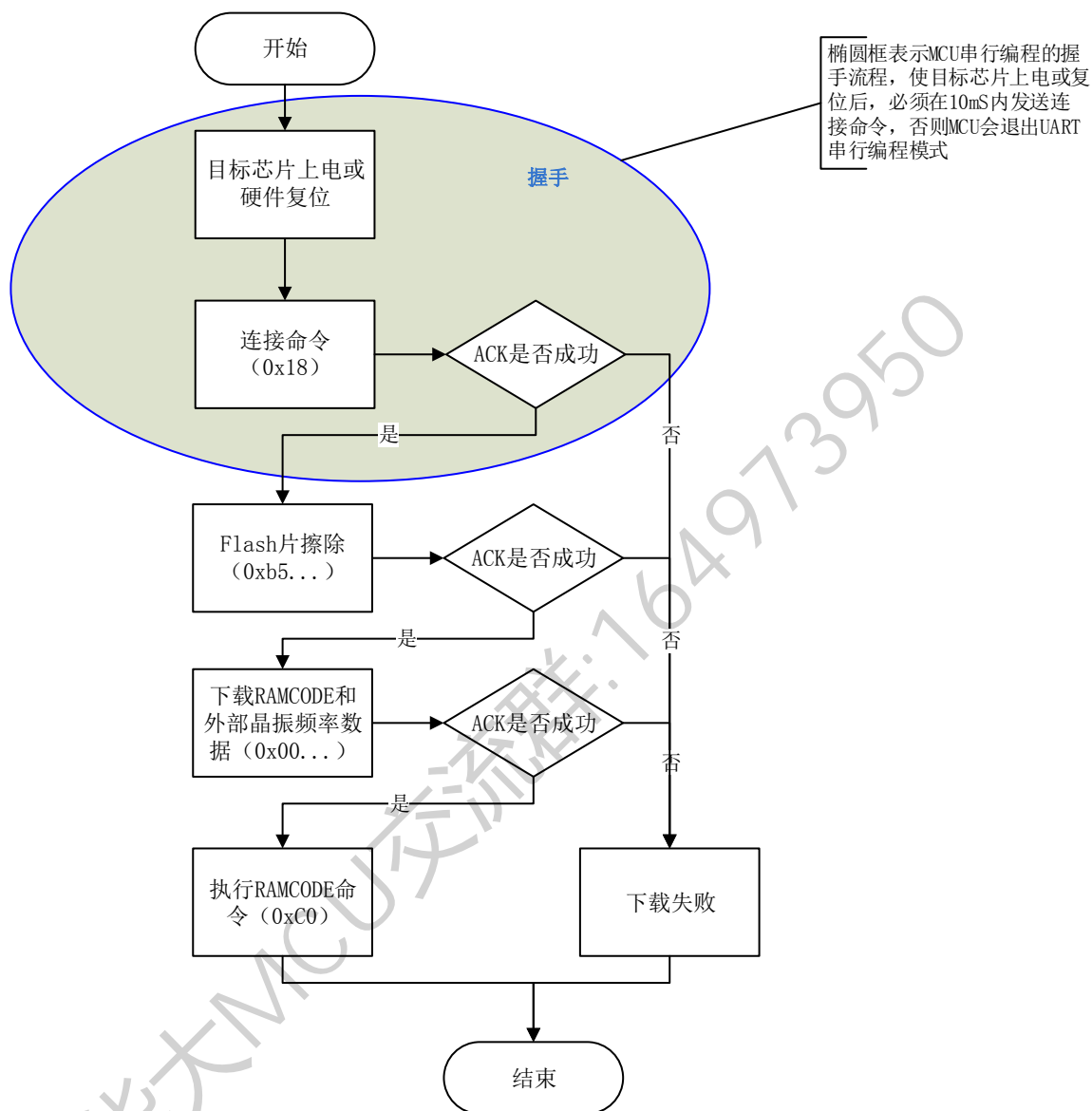


图 3 下载并运行 RAMCODE 流程

下载并运行 RAMCODE 使用到 4 条串行命令：连接命令、下载命令、执行命令、全擦除命令。

4.1.1 连接命令

连接命令 0x18 与目标芯片建立通信连接，为之后的数据下载做准备。建立连接过程须要在一定时间内多次发送，直到收到正确的 ACK 0x11，表示目标芯片已经准备好。

连接命令数据帧及对应 ACK 结构如下所示：

	编程器→目标芯片	目标芯片→编程器	
1 byte	连接命令 (0x18)		
1 byte		ACK:	
		成功	: 0x11

4.1.2 下载 RAMCODE

下载命令 0x00 将 RAMCODE 数据与外部晶振频率数据下载到 MCU 内部 RAM 相应位置。下载命令数据帧及对应 ACK 结构如下所示：

	编程器→目标芯片	目标芯片→编程器	
1 byte	下载命令 (0x00)		
4 byte	RAM 起始地址 (0x20000000)		
4 byte	RAMCODE 文件长度 (0x00000XXX)		
1 byte	命令校验和 (SUM1)		
1 byte		ACK:	
		成功	: 0x01
		校验和错误	: 0x02
N byte	下载数据		
1 byte	数据校验和 (SUM2)		
1 byte		ACK:	
		成功	: 0x01
		校验和错误	: 0x02

注：

- 表格中 RAM 起始地址和 RAMCODE 长度数据均按小端方式发送。起始地址为 0x20000000 则依次发送 0x00、0x00、0x00、0x20
- RAMCODE 长度：为华大半导体提供 RAMCODE 文件的数据长度 (in byte)

- SUM1: 指下载命令、RAM 起始地址、RAMCODE 长度这 9 byte 数据以字节按字节累加结果
- SUM2: 指 N byte 的下载数据按字节累加结果

4.1.3 执行命令

执行命令 0xC0 发送后 RAMCODE 开始运行，如果命令发送且 RAMCODE 正确运行，ACK 会返回 11 个 byte 的随机数据，数据帧及对应 ACK 结构如下所示。

	编程器→目标芯片	目标芯片→编程器	
1 byte	执行命令 (0xC0)		
8 byte	虚拟数据 (全 0x00)		
1 byte	命令校验和 (0xC0)		
1 byte 或者 11byte		ACK:	
		执行成功 校验和错误	: 11 byte 的随机数据 : 0xC2

4.1.4 全擦除命令

全擦除命令用于在串行编程时擦除 MCU 内部用户区域的 Flash，数据帧及对应 ACK 结构如下所示。

	编程器→目标芯片	目标芯片→编程器	
1 byte	命令码 1 (0xB5)		
1 byte	命令码 2 (0x34)		
1 byte	命令码 3 (0x84)		
1 byte	命令码 4 (0x52)		
1 byte	校验码 (0xBF)		
1 byte		ACK:	
		执行成功 错误	: 0x01 : 0x02

4.2 更改波特率设置

RAMCODE 下载成功并运行成功后，RAMCODE 实现的 API（应用接口）生效，其 API 接口格式和支持命令请查看附录 1：RAMCODE API 接口。

编程器可以通过 API 命令重新配置目标系统的波特率提高编程效率，建议配置波特率为 115200bps。（目前本系列 RAMCODE 支持的波特率为 9600、19200、38400、115200bps）

命令样例

例如将波特率设置为 115200，命令帧及波特率设置成功的 ACK 数据如下：

编程器→目标芯片		目标芯片→编程器	
Start code	0x49		
	0x53		
Frame length	0x00 0x09		
CMD	0x01		
Address	0x00		
	0x00		
	0x00		
	0x00		
Data length	0x00 0x01		
Baudrate	0x06		
Checksum	0x11		
		Start code	0x49 0x53
		Frame length	0x00 0x07
		CMD	0x01
		Status	0x00
		Address	0x00 0x00 0x00 0x00
		Checksum	0x08

4.3 Flash 操作

更改波特率设置成功后，Flash 操作流程所进行的串行通信都是用最新配置的波特率进行，Flash 操作流程如图 4 所示。

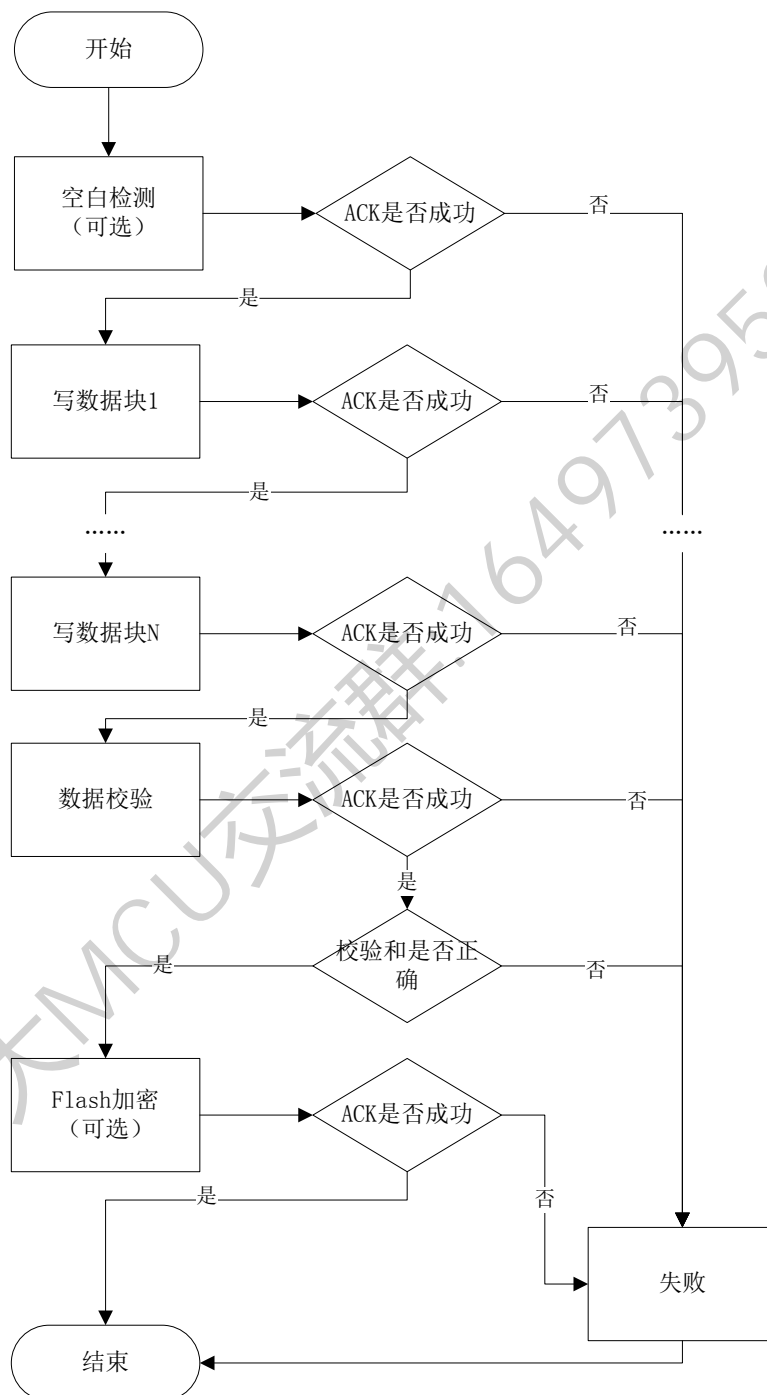


图 4 Flash 操作流程

4.3.1 Flash 空白检测

Flash 空白检测为可选步骤，检测结束后，ACK 会返回所检测区域是否是空白（全 FF）。

命令样例

片擦除成功后，对 Flash 整片数据空白检测及 ACK 数据如下：

编程器→目标芯片		目标芯片→编程器	
Start code	0x49 0x53		
Frame length	0x00 0x0C		
CMD	0x07		
Check address	0x00 0x00 0x00 0x00		
Data length	0x00 0x04		
Check Lengh	0x00 0x02 0x00 0x00		
Checksum	0x19		
		Start code	0x49 0x53
		Frame length	0x00 0x08
		CMD	0x07
		Status	0x00
		Check address	0x00 0x00 0x00 0x00
		Result	0x01
		Checksum	0x10

4.3.2 Flash 写

Flash 写将需要编程的数据写到 Flash 相应位置，可以分包多次用 Flash 写命令写入。

命令样例

写一包数据长度为 64Byte 的数据到 Flash 地址 0x0200 中及 ACK 数据如下：

编程器→目标芯片		目标芯片→编程器	
Start code	0x49 0x53		
Frame length	0x00 0x48		
CMD	0x04		
Write address	0x00 0x00 0x02 0x00		
Write length	0x00 0x40		
Write data	略(64byte)		
Checksum	0xE3		
		Start code	0x49 0x53
		Frame length	0x00 0x07
		CMD	0x04
		Status	0x00
		Write address	0x00 0x00 0x02 0x00
		Checksum	0x0d

注：

- 命令帧中，Write data 为需要写入 Flash 中的 64 byte 数据。
- 命令帧中，Checksum 为此样例中的值，应用中请按照附录中的帧结构进行计算。

4.3.3 读 Flash 校验和

读 Flash 校验和命令起始地址和检测长度都可以设置。读取结束后，ACK 会返回所检测区域内的校验和（按字节累加）。

命令样例

Flash 从地址 0 开始长度 3610 字节校验和及 ACK 数据如下：

编程器→目标芯片		目标芯片→编程器	
Start code	0x49		
	0x53		
Frame length	0x00		
	0x0C		
CMD	0x06		
Check address	0x00		
	0x00		
	0x00		
	0x00		
Data length	0x00		
	0x04		
Check Lengh	0x00		
	0x00		
	0x0E		
	0x1A		
Checksum	0x3E		
		Start code	0x49
			0x53
		Frame length	0x00
			0x09
		CMD	0x06
		Status	0x00
		Check address	0x00
			0x00
0x00			
0x00			
Checksum result	0xEC		
	0xB2		
Checksum	0xAD		

注：Checksum result 为 Flash 校验区域内的校验和，此样例中校验和为 0xECB2。

4.3.4 Flash 加密设置

Flash 加密设置命令，将目标芯片的 flash 加密，成功后 RAMCODE 以及 SWD 接口无法读出 Flash 内部数据。

命令样例

编程器→目标芯片		目标芯片→编程器	
Start code	0x49 0x53		
Frame length	0x00 0x08		
CMD	0x09		
Address	0x00 0x00 0x00 0x00		
Data length	0x00 0x00		
Checksum	0x11		
		Start code	0x49 0x53
		Frame length	0x00 0x07
		CMD	0x09
		Status	0x00
		Address	0x00 0x00 0x00 0x00
		Checksum	0x10

4.4 错误处理

如图 2 所示，RAMCODE 下载等过程中均可能出现失败的情况，遇到出错的情况，如果再一次对目标芯片进行 Flash 编程需要使芯片断电或硬件复位并且重新进行握手及 RAMCODE 下载流程。

建议的错误处理流程如图 5 所示，方式 1 和方式 2 选择任意一种可以恢复目标芯片的状态。

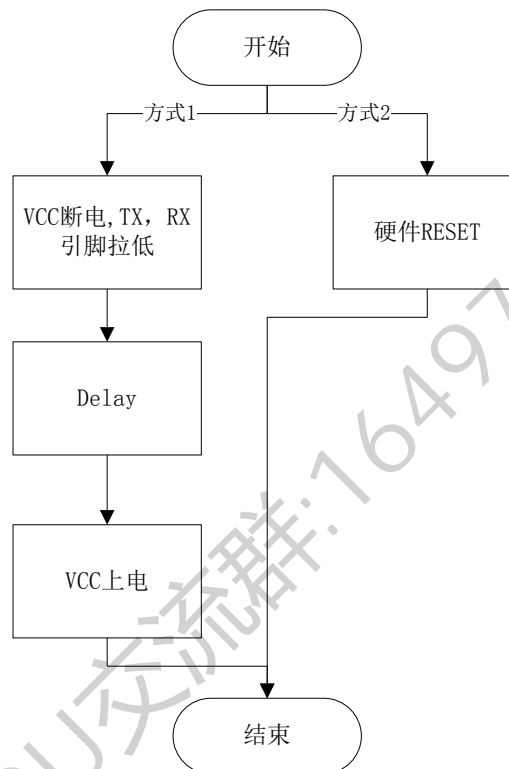


图 5 错误处理

注：

- Delay 时间长度与板子硬件环境有关，需要满足 Delay 时间内 MCU 所在系统的 VCC 完全掉电为止。
- 硬件 RESET 是指通过 MCU 的硬件复位引脚复位，复位时序要求请查看具体芯片的规格书。

附录 1 : RAMCODE API 接口

命令帧

命令帧是指由上位机（PC 或者编程器）发送给下位机（MCU）的数据帧。

D0D1	D2D3	D4	D5D6D7D8	D9D10	...	Dn
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	XXXX	XX	XXXXXXXX	XXXX	XX...XX	XX

Start code: 数据帧起始码 0x4953, D0 – 0x49, D1 – 0x53 (RAMCODE API 接口中提到的 word, long 数据都按此大端结构发送数据)

Frame Length: 数据帧长度, 从 CMD 到 Checksum 的字节数

CMD: 命令码, 定义如下:

CMD	命令	CMD	命令
0x01	串口波特率设置	0x06	读 Flash 校验和
0x02	Flash 片擦除	0x07	Flash 空白检查
0x04	Flash 写	0x09	Flash 加密设置

Address: Flash 地址, 有些命令此域无效, 为 0

Data Length: 数据长度, 有些命令此域无效, 为 0

Data: 数据, 某些命令可能缺省这一项

Check Sum: 数据帧校验和 (Frame Length, CMD, Address, Data Length, Data 按 byte 相加)。

ACK 帧

ACK 帧是指下位机（MCU）返回给上位机（PC）的数据帧。

D0D1	D2D3	D4	D5	D6D7D8D9	...	Dn
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	XXXX	XX	XX	XXXXXXXX	XX...XX	XX

Start code: 同命令帧

Frame Length: 同命令帧

CMD: 同命令帧

Status: ACK 状态

0 – 成功;

1 – 命令帧校验错误;

2 – 无法支持的命令码;

3 – Flash 地址不在支持范围内

4 – 帧长度不在支持范围内

5 – 数据长度不在支持范围内

6 – 波特率不在支持范围内;

7 – 加密状态下, 禁止读、写或页擦除操作;

Address: 同命令帧

Data: 同命令帧

Check Sum: 同命令帧

1) 串口波特率设置

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10	D11	D12
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	0x0009	0x01	0x00000000	0x0001	Baud rate	XX

Baud rate: 上位机设置的波特率，详细定义如下：

Baud rate	波特率
0x01	9600
0x02	14400
0x03	19200
0x04	38400
0x05	57600
0x06	115200
0x07	128000
0x08	76800
0x09	256000

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9		D10
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0007	0x01	Std	0x00000000	N/A	XX

Std: 0 – 表示正确收到命令帧

非 0 – 失败（详情见 ACK 帧说明）

2) Flash片擦除

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10		D11
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	0x0008	0x02	0x00000000	0x0000	N/A	XX

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9		D10
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0007	0x02	Std	0x00000000	N/A	XX

Std: 0 – 片擦除成功

非 0 – 失败（详情见 ACK 帧说明）

3) Flash写

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10	D11~Dn-1	Dn
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	XXXX	0x04	Write Adr	Write Length	Write Data	XX

Write Adr: Flash 写的起始地址

Write Length: Flash 写的长度

Write Data: Flash 写的的数据

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9		D10
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0007	0x04	Std	Write Adr	N/A	XX

Std: 0 – Flash 写成功

非 0 – 失败（详情见 ACK 帧说明）

Write Adr: Flash 写的起始地址

4) 读Flash校验和

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10	D11D12D13D14	D15
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	0x000C	0x06	Check Adr	0x0004	Check Length	XX

Check Adr: 计算校验和的 flash 起始地址

Check Length: 计算校验和的长度 (in byte)

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9	D10~D11	D12
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0009	0x06	Std	Check Adr	Checksum	XX

Std: 0 – 计算校验和成功

非 0 – 失败 (详情见 ACK 帧说明)

Read Adr: 计算校验和的 flash 起始地址

Checksum: 计算出的校验和

5) Flash空白检查

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10	D11D12D13D14	D15
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	0x000C	0x07	Check Adr	0x0004	Check Length	XX

Check Blank Adr: 空白检查的 flash 起始地址

Check Blank Length: 空白检查的长度 (in byte)

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9	D10	D11
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0008	0x07	Std	Check Adr	Result	XX

Std: 0 – 空白检查成功

非 0 – 失败 (详情见 ACK 帧说明)

Check Adr: 空白检查的 flash 起始地址

Result: 空白检查结果

TRUE(1) – 检查范围空白

FALSE(0) – 检查范围非空白

6) Flash加密设置

命令帧

D0D1	D2D3	D4	D5D6D7D8	D9D10		D11
Start code	Frame Length	CMD	Address	Data Length	Data (optional)	Check Sum
0x4953	0x0008	0x09	0x00000000	0x0000	N/A	XX

ACK 帧

D0D1	D2D3	D4	D5	D6D7D8D9		D10
Start code	Frame Length	CMD	Status	Address	Data (optional)	Check Sum
0x4953	0x0007	0x09	Std	0x00000000	N/A	XX

Std: 0 – 加密设置成功

非 0 – 失败 (详情见 ACK 帧说明)

5 总结

以上章节简要介绍了 Bootloader 串行编程，并详细说明了 HC32L110 / HC32F003 / HC32F005 系列的 Bootloader 编程的接口，在开发中用户可以根据自己的实际需要使用该 Bootloader。

华大MCU交流群:164973950

6 版本信息 & 联系方式

日期	版本	修改记录
2018/6/4	Rev1.0	初版发布



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@hdsc.com.cn

网址 : www.hdsc.com.cn

通信地址：上海市张江高科园区碧波路 572 弄 39 号

邮编：201203

