

STM32™ 微控制器系统存储器自举模式

前言

自举程序存储在 STM32 器件的内部自举 ROM 存储器（系统存储器）中。在生产期间由 ST 编程。其主要任务是通过一种可用的串行外设（USART、CAN、USB、I²C 等）将应用程序下载到内部 Flash 中。每种串行接口都定义了相应的通信协议，其中包含兼容的命令集和序列。

本文档适用于表 1 中所列产品。这些产品在整个文档中称为 STM32。

表 1. 适用的产品

类型	型号和产品系列
微控制器	所有 STM32F1（主流）产品 所有 STM32F2（高性能）产品 STM32F050（入门级）产品： – STM32F050x4、STM32F050x6、STM32F051x4、STM32F051x6 和 STM32F051x8 STM32L1（超低功耗）产品： – STM32L151xx、STM32L152xx 和 STM32L162xx STM32F3（模拟与 DSP）产品： – STM32F302xx、STM32F303xx、STM32F313xx、STM32F372xx、STM32F373xx 和 STM32F383xx STM32F4（高性能与 DSP）产品： – STM32F405xx、STM32F407xx、STM32F415xx、STM32F417xx、STM32F427xx 和 STM32F437xx

自举程序的主要功能如下：

- 使用嵌入式串行接口按照预定义的通信协议下载代码
- 可传送并更新 Flash 代码、数据和向量表部分

本应用笔记介绍了自举程序的一般概念。说明了使用表 1 中所列 STM32 器件的自举程序时支持的外设以及需要考虑的硬件要求。不过，支持的每种串行外设的低层级通信协议规范在单独的文档中进行了介绍。有关自举程序中使用的 USART 协议的规范，请参见 AN3155。有关自举程序中使用的 CAN 协议的规范，请参见 AN3154。有关自举程序中使用的 DFU（USB 设备）协议的规范，请参见 AN3156。有关自举程序中使用的 I2C 协议的规范，请参见 AN4221。

目录

1	相关文档	9
2	词汇表	9
3	自举程序一般说明	11
3.1	自举程序激活	11
3.2	退出系统存储器自举模式	11
3.3	自举程序标识	12
4	STM32F100xx、STM32F101xx、STM32F102xx、STM32F103xx 中容量和大容量超值型自举程序	14
4.1	自举程序配置	14
4.2	自举程序硬件要求	14
4.3	自举程序选择	15
4.4	自举程序版本	16
5	STM32F105xx 和 STM32F107xx 器件自举程序	17
5.1	自举程序配置	17
5.2	自举程序硬件要求	19
5.3	自举程序选择	19
5.4	自举程序版本	22
5.4.1	如何识别 STM32F105xx/107xx 自举程序版本	22
5.4.2	日期代码小于 937 的 STM32F105xx/STM32F107xx 器件上自举程序不可用	23
5.4.3	USART 自举程序 Get-Version 命令返回 0x20 而不是 0x22	23
5.4.4	自举程序 V2.0 中插入 USB 电缆时 PA9 功耗过大	23
6	STM32F101xx 和 STM32F103xx 超大容量器件自举程序	24
6.1	双存储区自举功能	24
6.2	自举程序配置	25
6.3	自举程序硬件要求	26
6.4	自举程序选择	27
6.5	自举程序版本	29

7	STM32L151xx 和 STM32L152xx 中容量超低功耗器件自举程序	30
7.1	自举程序配置	30
7.2	自举程序硬件要求	31
7.3	自举程序选择	31
7.4	重要事项	32
7.5	自举程序版本	33
8	STM32L151xx、STM32L152xx 和 STM32L162xx 大容量超低功耗器件自举程序	34
8.1	双存储区自举功能	34
8.2	自举程序配置	36
8.3	自举程序硬件要求	37
8.4	自举程序选择	38
8.5	重要事项	40
8.6	自举程序版本	41
9	STM32F205/215xx 和 STM32F207/217xx 自举程序	42
9.1	自举程序 V2.x	42
9.1.1	自举程序配置	42
9.1.2	自举程序硬件要求	43
9.1.3	自举程序选择	44
9.1.4	重要事项	45
9.1.5	自举程序 V2.x 版本	46
9.2	自举程序 V3.x	46
9.2.1	自举程序配置	46
9.2.2	自举程序硬件要求	48
9.2.3	自举程序选择	49
9.2.4	重要事项	51
9.2.5	自举程序版本 V3.x	52
10	STM32F405/415xx 和 STM32F407/417xx 自举程序	53
10.1	自举程序配置	53
10.2	自举程序硬件要求	55
10.3	自举程序选择	56
10.4	重要事项	58
10.5	自举程序版本	59

11	STM32F051x4、STM32F051x6 和 STM32F051x8 器件自举程序	60
11.1	自举程序配置	60
11.2	自举程序硬件要求	61
11.3	自举程序选择	61
11.4	重要事项	63
11.5	自举程序版本	63
12	STM32L151xx 和 STM32L152xx 中容量增强型超低功耗器件自举程序 ..	64
12.1	自举程序配置	64
12.2	自举程序硬件要求	66
12.3	自举程序选择	66
12.4	重要事项	68
12.5	自举程序版本	68
13	STM32F050x4 和 STM32F050x6 器件自举程序	69
13.1	自举程序配置	69
13.2	自举程序硬件要求	70
13.3	自举程序选择	70
13.4	重要事项	72
13.5	自举程序版本	72
14	STM32F372xx 和 STM32F373xx 器件自举程序	73
14.1	自举程序配置	73
14.2	自举程序硬件要求	74
14.3	自举程序选择	75
14.4	重要事项	77
14.5	自举程序版本	77
15	STM32F302xx 和 STM32F303xx 器件自举程序	78
15.1	自举程序配置	78
15.2	自举程序硬件要求	79
15.3	自举程序选择	80
15.4	重要事项	82
15.5	自举程序版本	82

16	STM32F383xx 器件自举程序	83
16.1	自举程序配置	83
16.2	自举程序硬件要求	84
16.3	自举程序选择	84
16.4	重要事项	86
16.5	自举程序版本	86
17	STM32F313xx 器件自举程序	87
17.1	自举程序配置	87
17.2	自举程序硬件要求	88
17.3	自举程序选择	89
17.4	重要事项	91
17.5	自举程序版本	91
18	STM32F427xx 和 STM32F437xx 器件自举程序	92
18.1	自举程序配置	92
18.2	自举程序硬件要求	94
18.3	自举程序选择	95
18.4	重要事项	97
18.5	自举程序版本	98
19	器件相关的自举程序参数	99
20	自举程序时序特性	101
20.1	USART 自举程序时序特性	101
20.2	USB 自举程序时序特性	108
20.3	I2C 自举程序时序特性	113
21	版本历史	115

表格索引

表 1.	适用的产品	1
表 2.	自举引脚配置	11
表 3.	嵌入式自举程序	12
表 4.	系统存储器自举模式下 STM32F10xxx 器件的配置	14
表 5.	STM32F10xxx 自举程序版本	16
表 6.	系统存储器自举模式下的 STM32F105xx/107xx 配置	17
表 7.	STM32F105xx 和 STM32F107xx 自举程序版本	22
表 8.	自举引脚和 BFB2 位配置	24
表 9.	系统存储器自举模式下 STM32F10xxx 超大容量器件的配置	25
表 10.	超大容量器件自举程序版本	29
表 11.	系统存储器自举模式下 STM32L15xxx 器件的配置	30
表 12.	STM32L15xxx 中容量器件自举程序版本	33
表 13.	自举引脚和 BFB2 位配置	34
表 14.	系统存储器自举模式下 STM32L1xxxx 大容量器件的配置	36
表 15.	STM32L1xxxx 大容量器件自举程序版本	41
表 16.	系统存储器自举模式下 STM32F2xxxx 的配置	42
表 17.	使用自举程序 V2.x 配置 STM32F2xxxx 电压范围	45
表 18.	STM32F2xxxx 自举程序 V2.x 版本	46
表 19.	系统存储器自举模式下 STM32F2xxxx 的配置	46
表 20.	使用自举程序 V3.x 配置 STM32F2xxxx 电压范围	51
表 21.	STM32F2xxxx 自举程序 V3.x 版本	52
表 22.	系统存储器自举模式下 STM32F40xxx/41xxx 的配置	53
表 23.	使用自举程序配置 STM32F40xxx/41xxx 电压范围	58
表 24.	STM32F40xxx/41xxx 自举程序版本	59
表 25.	系统存储器自举模式下 STM32F051xx 的配置	60
表 26.	STM32F051xx 自举程序版本	63
表 27.	系统存储器自举模式下 STM32L15xxx 中容量增强型器件的配置	64
表 28.	STM32L15xxx 中容量增强型器件自举程序版本	68
表 29.	系统存储器自举模式下 STM32F050xx 的配置	69
表 30.	STM32F050xx 自举程序版本	72
表 31.	系统存储器自举模式下 STM32F37xxx 的配置	73
表 32.	STM32F37xxx 自举程序版本	77
表 33.	系统存储器自举模式下 STM32F30xxx 的配置	78
表 34.	STM32F30xxx 自举程序版本	82
表 35.	系统存储器自举模式下 STM32F38xxx 的配置	83
表 36.	STM32F38xxx 自举程序版本	86
表 37.	系统存储器自举模式下 STM32F31xxx 的配置	87
表 38.	STM32F31xxx 自举程序版本	91
表 39.	系统存储器自举模式下 STM32F427xx/437xx 的配置	92
表 40.	使用自举程序配置 STM32F427xx/437xx 电压范围	97
表 41.	STM32F427xx/437xx 自举程序版本	98
表 42.	器件相关的自举程序参数	99
表 43.	小 / 中 / 大容量超值型器件的 USART 自举程序时序	102
表 44.	超大容量器件的 USART 自举程序时序	102
表 45.	互连型器件的 USART 自举程序时序 (PA9 引脚为低电平)	103
表 46.	互连型器件的 USART 自举程序时序 (PA9 引脚为高电平)	103
表 47.	中容量超低功耗器件的 USART 自举程序时序	104
表 48.	大容量超低功耗器件的 USART 自举程序时序	104

表 49.	STM32F2xxxx 器件的 USART 自举程序时序	105
表 50.	STM32F40xxx/41xxx 器件的 USART 自举程序时序	105
表 51.	STM32F051xx 器件的 USART 自举程序时序	105
表 52.	中容量增强型器件的 USART 自举程序时序	106
表 53.	STM32F050xx 器件的 USART 自举程序时序	106
表 54.	STM32F37xxx 器件的 USART 自举程序时序	106
表 55.	STM32F30xxx 器件的 USART 自举程序时序	107
表 56.	STM32F38xxx 器件的 USART 自举程序时序	107
表 57.	STM32F31xxx 器件的 USART 自举程序时序	107
表 58.	STM32F427xx 和 STM32F437xx 器件的 USART 自举程序时序	108
表 59.	互连型器件的 USB 最短时序	109
表 60.	大容量超低功耗器件的 USB 最短时序	110
表 61.	STM32F2xxxx 器件的 USB 最短时序	110
表 62.	STM32F40xxx/41xxx 器件的 USB 最短时序	110
表 63.	中容量增强型器件的 USB 最短时序	111
表 64.	STM32F37xxx 器件的 USB 最短时序	111
表 65.	STM32F30xxx 器件的 USB 最短时序	111
表 66.	STM32F427xx/437xx 器件的 USB 最短时序	112
表 67.	STM32F38xxx 器件的 I2C 最短时序	114
表 68.	STM32F31xxx 器件的 I2C 最短时序	114
表 69.	文档版本历史	115

图片索引

图 1. STM32F10xxx USART1 自举程序 15

图 2. STM32F105xx 和 STM32F107xx 器件的自举程序选择 21

图 3. STM32F10xxx 超大容量器件的自举程序选择 28

图 4. STM32L15xxx 中容量器件的自举程序选择 32

图 5. STM32L1xxxx 大容量器件的自举程序选择 39

图 6. STM32F2xxxx 器件的自举程序 V2.x 选择 44

图 7. STM32F2xxxx 器件的自举程序 V3.x 选择 50

图 8. STM32F40xxx/41xxx 器件的自举程序选择 57

图 9. STM32F051xx 器件的自举程序选择 62

图 10. STM32L15xxx 中容量增强型器件的自举程序选择 67

图 11. STM32F050xx 器件的自举程序选择 71

图 12. STM32F37xxx 器件的自举程序选择 76

图 13. STM32F30xxx 器件的自举程序选择 81

图 14. STM32F38xxx 器件的自举程序选择 85

图 15. STM32F31xxx 器件的自举程序选择 90

图 16. STM32F427xx/437xx 器件的自举程序选择 96

图 17. USART 自举程序时序波形 101

图 18. USB 自举程序时序波形 109

图 19. I2C 自举程序时序波形 113



1 相关文档

有关所支持的每个产品（如表 1 中所列）的信息，请参见 <http://www.st.com> 提供的以下文档：

- 数据手册或产品简介
- 参考手册和/或 flash 编程手册

2 词汇表

小容量器件包括 Flash 容量介于 16 KB 到 32 KB 之间的 STM32F101xx、STM32F102xx 和 STM32F103xx 微控制器。

中容量器件包括 Flash 容量介于 64 KB 到 128 KB 之间的 STM32F101xx、STM32F102xx 和 STM32F103xx 微控制器。

大容量器件包括 Flash 容量介于 256 KB 到 512 KB 之间的 STM32F101xx 和 STM32F103xx 微控制器。

互连型器件包括 STM32F105xx 和 STM32F107xx 微控制器。

小容量超值型器件包括 Flash 容量介于 16 KB 到 32 KB 之间的 STM32F100xx 微控制器。

中容量超值型器件包括 Flash 容量介于 64 KB 到 128 KB 之间的 STM32F100xx 微控制器。

大容量超值型器件包括 Flash 容量介于 256 KB 到 512 KB 之间的 STM32F100xx 微控制器。

超大容量器件包括 Flash 容量介于 768 KB 到 1 MB 之间的 STM32F101xx 和 STM32F103xx 微控制器。

中容量超低功耗器件包括程序存储器容量介于 64 KB 到 128 KB 之间的 STM32L151xx 和 STM32L152xx 微控制器。

中容量增强型超低功耗器件包括程序存储器容量为 256 KB 的 STM32L151xx 和 STM32L152xx 微控制器。

大容量超低功耗器件包括程序存储器容量为 384 KB 的 STM32L151xx、STM32L152xx 和 STM32L162xx 微控制器。

STM32F051xx 器件包括 Flash 容量介于 32 KB 到 64 KB 之间的 STM32F051x4、STM32F051x6 和 STM32F051x8 微控制器。

STM32F050xx 器件包括 Flash 容量介于 16 KB 到 32 KB 之间的 STM32F050x4 和 STM32F050x6 微控制器。

STM32F2xxxx 器件包括 Flash 容量介于 128 KB 到 1024 KB 之间的 STM32F215xx、STM32F205xx、STM32F207xx 和 STM32F217xx 微控制器。

STM32F31xxx 器件包括 Flash 容量介于 128 KB 到 256 KB 之间的 STM32F313xx 微控制器。

STM32F30xxx 器件包括 Flash 容量介于 128 KB 到 256 KB 之间的 STM32F302xx 和 STM32F303xx 微控制器。

STM32F37xxx 器件包括 Flash 容量介于 128 KB 到 256 KB 之间的 STM32F372xx 和 STM32F373xx 微控制器。

STM32F38xxx 器件包括 Flash 容量介于 128 KB 到 256 KB 之间的 STM32F383xx 微控制器。

STM32F40xxx/41xxx 器件包括 Flash 容量介于 512 KB 到 1024 KB 之间的 STM32F405xx、STM32F407xx、STM32F415xx 和 STM32F417xx 微控制器。

STM32F427xx/437xx 器件包括 Flash 容量介于 1024 KB 到 2048 KB 之间的 STM32F427xx 和 STM32F437xx 微控制器。

注: *BL_USART_Loop* 指 USART 自举程序执行循环。
BL_CAN_Loop 指 CAN 自举程序执行循环。
BL_I2C_Loop 指 I2C 自举程序执行循环。

3 自举程序一般说明

3.1 自举程序激活

通过配置特定“系统存储器”配置中的 **BOOT0** 和 **BOOT1** 引脚（请参见表 2），然后执行复位即可自动激活自举程序。

根据所用引脚配置的不同，可选择 **Flash**、系统存储器或 **SRAM** 作为自举空间，如以下表 2 所示。

在一些产品中，**BOOT1** 不表示 I/O 引脚，而是选项字节区域的一个位。**STM32F05x** 和 **STM32F3xx** 器件就属于这种情况，此时 **BOOT1** 通过选项字节中的 **nBoot1** 位进行配置。

- 当 **nBoot1** 位置 1 时，相当于表 2 中的 **BOOT1** 复位为 0。
- 当 **nBoot1** 位复位为 0 时，相当于表 2 中的 **BOOT1** 置 1。

表 2. 自举引脚配置

自举模式选择引脚		自举模式	别名使用
BOOT1	BOOT0		
X	0	用户 Flash	选择用户 Flash 作为自举空间
0	1	系统存储器	选择系统存储器作为自举空间
1	1	嵌入式 SRAM	选择嵌入式 SRAM 作为自举空间

表 2 表明 **BOOT** 引脚配置如下时，**STM32** 微控制器进入系统存储器自举模式：

- **BOOT0** = 1
- **BOOT1** = 0

复位后，在 **SYSCLK** 的第四个上升沿锁存 **BOOT** 引脚的值。

注： 在一些产品中，当 **STM32** 产品提供双存储区自举特性功能时，（**BOOT0** = 0 且 **BOOT1** = x）可进入自举程序。有关详细信息，请参见产品部分的双存储区自举特性部分。

当工作温度超出环境温度范围时，由于内部时钟 (**HSI**) 随着温度变化，致使串行通信协议时钟损坏，因而可能导致自举程序无法正常运行。

3.2 退出系统存储器自举模式

为了执行应用程序，必须先退出系统存储器自举模式。执行硬件复位即可实现此操作。复位期间，**BOOT** 引脚/位（**BOOT0** 和 **BOOT1**）必须设置为合适的电平，以选择所需的自举模式（请参见表 2）。复位后，CPU 将从自举存储器的存储器地址空间最底部（起始地址 0x0000 0000）开始执行代码。

3.3 自举程序标识

根据所用 STM32 器件的不同，自举程序可支持一个或多个嵌入式串行外设，用来将代码下载到内部 Flash 中。自举程序标识符 (ID) 可提供有关支持的串行外设的信息。

对于给定的 STM32 器件，自举程序通过以下各项标识：

1. **自举程序（协议）版本：**自举程序中使用的串行外设（USART、CAN、USB 等）通信协议的版本。可使用自举程序 **Get Version** 命令检索此版本。
2. **自举程序标识符 (ID)：**STM32 器件自举程序的版本，以 **0xXY** 形式的单字节代码表示，其中：

- **X** 指定器件自举程序所用的嵌入式串行外设：

X = 1：仅使用一个 USART

X = 2：使用两个 USART

X = 3：使用两个 USART，一个 CAN 和 DFU

X = 4：使用两个 USART 和 DFU

X = 5：使用两个 USART 和 I2C

- **Y** 指定器件的自举程序版本

下面以自举程序 ID **0x10** 为例。这表示仅使用一个 USART 的器件自举程序的第一个版本。

自举程序 ID 编程在器件系统存储器最后一个字节地址减 1 所对应的空间中，可通过自举程序“**Read memory**”命令来读取，或者通过使用 JTAG/SWD 直接访问系统存储器来读取。

下表提供了有关 STM32 器件嵌入式自举程序的标识信息。

表 3. 嵌入式自举程序

STM32 系列	器件	支持的串行外设	自举程序 ID		自举程序（协议）版本
			ID	存储单元	
F1	小容量	USART1	NA	NA	USART (V2.2)
	中容量	USART1	NA	NA	USART (V2.2)
	大容量	USART1	NA	NA	USART (V2.2)
	互连型	USART1/USART2（重映射）/ CAN2（重映射）/DFU（USB 设备）	NA	NA	USART (V2.2 ⁽¹⁾) CAN (V2.0) DFU(V2.2)
	中容量超值型	USART1	V1.0	0x1FFFF7D6	USART (V2.2)
	大容量超值型	USART1	V1.0	0x1FFFF7D6	USART (V2.2)
	超大容量	USART1/USART2（重映射）	V2.1	0x1FFFF7D6	USART (V3.0)
L1	中容量超低功耗	USART1/USART2	V2.0	0x1FF00FFE	USART (V3.0)
	大容量超低功耗	USART1/USART2/DFU （USB 设备 FS）	V4.5	0x1FF01FFE	USART (V3.1)/ DFU (V2.2)
	中容量增强型超低功耗	USART1/USART2/DFU （USB 设备 FS）	V4.0	0x1FF01FFE	USART (V3.1)/ DFU (V2.2)

表 3. 嵌入式自举程序（续）

STM32 系列	器件	支持的串行外设	自举程序 ID		自举程序 (协议) 版本
			ID	存储单元	
F2	STM32F2xxx	USART1/USART3	V2.0	0x1FFF77DE	USART (V3.0)
		USART1/USART3/CAN2/ DFU (USB 设备 FS)	V3.3	0x1FFF77DE	USART (V3.1)/CAN (V2.0)/DFU (V2.2)
F0	STM32F051xx	USART1/USART2	V2.1	0x1FFFF7A6	USART (V3.1)
	STM32F050xx	USART1	V1.0	0x1FFFF7A6	USART (V3.1)
F4	STM32F40xxx/ 41xxx	USART1/USART3/CAN2/ DFU (USB 设备 FS)	V3.1	0x1FFF77DE	USART (V3.1)/ CAN (V2.0)/ DFU (V2.2)
	STM32F427xx/ 437xx	USART1/USART3/CAN2/ DFU (USB 设备 FS)	V3.0	0x1FFF76DE	USART (V3.1)/ CAN (V2.0)/ DFU (V2.2)
F3	STM32F37xxx	USART1/USART2/DFU (USB 设备 FS)	V4.1	0x1FFFF7A6	USART (V3.1)/ DFU (V2.2)
	STM32F30xxx	USART1/USART2/DFU (USB 设备 FS)	V4.1	0x1FFFF796	USART (V3.1)/ DFU (V2.2)
	STM32F38xxx	USART1/USART2/I2C1	V5.0	0x1FFFF7A6	USART (V3.1)/ I2C (V1.0)
	STM32F31xxx	USART1/USART2/I2C1	V5.0	0x1FFFF796	USART (V3.1)/ I2C (V1.0)

1. 对于互连型器件，USART 自举程序将返回 V2.0 协议版本而不是 V2.2 协议版本。有关详细信息，请参见 <http://www.st.com> 提供的“STM32F105xx 和 STM32F107xx 版本 Z”勘误表。

4 STM32F100xx、STM32F101xx、STM32F102xx、STM32F103xx 中容量和大容量超值型自举程序

本部分使用 STM32F10xxx 指代小容量、中容量及大容量 STM32F101xx 和 STM32F103xx 器件，低容量和中容量 STM32F102xx 器件，低容量、中容量及大容量 STM32F100xx 器件以及中容量和大容量超值型器件。

4.1 自举程序配置

STM32F10xxx 器件嵌入式自举程序仅支持 USART1 这一种接口。

下表介绍了系统存储器自举模式下自举程序需要使用的 STM32F10xxx 硬件资源。

表 4. 系统存储器自举模式下 STM32F10xxx 器件的配置

自举程序	功能/外设	状态	备注
USART1 自举程序	时钟源	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。
	RAM	-	自地址 0x20000000 起的 512 个字节空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFFF000 起的 2 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 用于接收。
	USART1_TX 引脚	推挽输出	PA9 引脚：USART1 用于发送。
	SysTick 定时器	使能	用于自动检测主机串口波特率。

系统时钟由嵌入式内部高速 RC 提供，自举程序代码不需要外部石英时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的外设寄存器（如上表所示）不会初始化为默认复位值。如果要使用这些寄存器，应在用户应用程序中对其重新配置。因此，如果应用程序正在使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

4.2 自举程序硬件要求

将 STM32 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

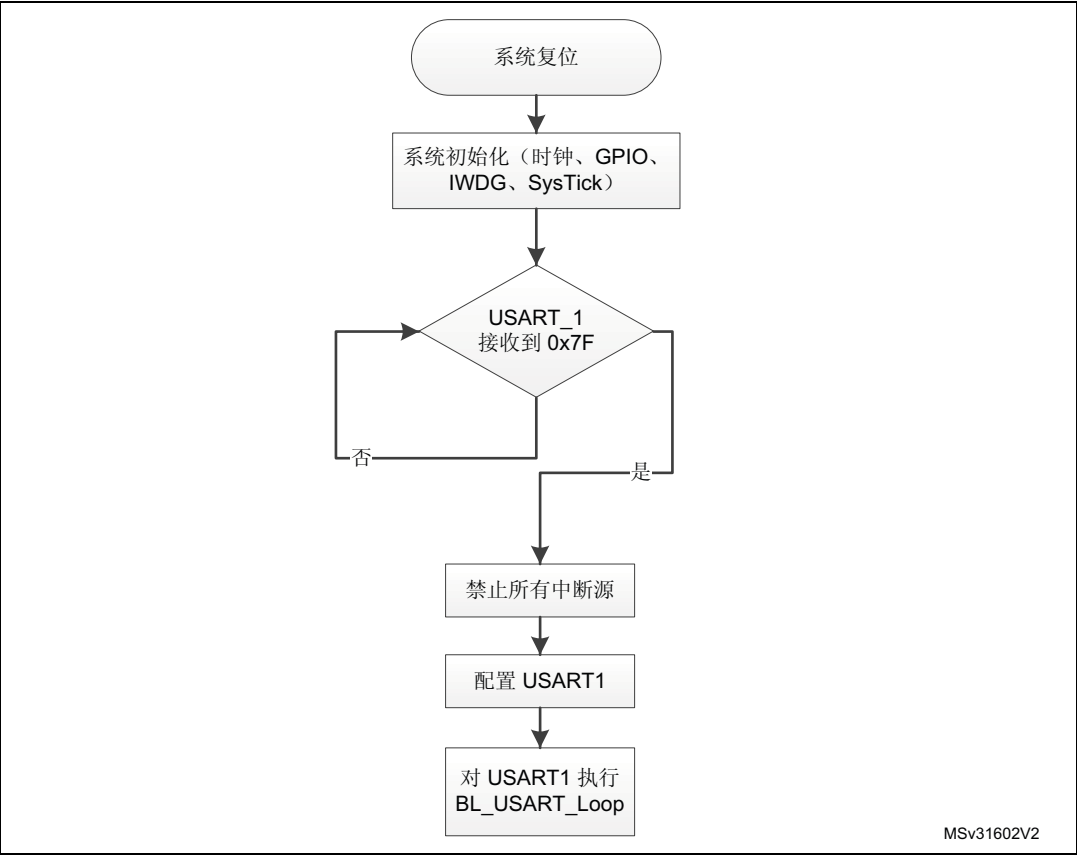
要在系统存储器自举模式下连接 STM32，必须将 RS232 串行接口（例如 ST3232 RS232 收发器）直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚。

注：USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此用户可将这些引脚用于其它外设或 GPIO。

有关硬件建议的详细信息，请参见应用笔记 AN2586：可从意法半导体网站 <http://www.st.com> 获取“STM32 硬件开发：入门”。

4.3 自举程序选择

图 1. STM32F10xxx USART1 自举程序



进入系统存储器自举模式且微控制器已按上述要求配置后，自举程序代码立即开始扫描 USART1_RX 引脚，等待接收 0x7F 数据帧：一个起始位、0x7F 数据位、偶校验位和一个停止位。

此数据帧的持续时间由 SysTick 定时器测量。之后，该定时器的计数值用于计算关于当前系统时钟的相应波特率因子。

随后，代码将相应初始化串行接口。通过计算出的波特率，向主机返回确认字节 (0x79)，指示 STM32F10xxx 已准备好接收用户命令。

4.4 自举程序版本

表 5 列出了 STM32F10xxx 器件的自举程序版本。

表 5. STM32F10xxx 自举程序版本

自举程序版本号	说明
V2.0	初始自举程序版本。
V2.1	<ul style="list-style-type: none">- 更新 Go 命令以初始化主栈指针- 更新 Go 命令，以在跳转地址处于选项字节区域或系统存储区时返回 NACK- 更新 Get ID 命令以返回两个字节的器件 ID- 将自举程序版本更新为 V2.1
V2.2	<ul style="list-style-type: none">- 更新 Read Memory、Write Memory 和 Go 命令，以通过 NACK 响应拒绝对自举程序所用的 RAM 存储器前 0x200 字节的访问- 更新 Readout Unprotect 命令，以在禁止 ROP 之前将整个 RAM 内容初始化为 0x0



5 STM32F105xx 和 STM32F107xx 器件自举程序

5.1 自举程序配置

STM32F105xx 和 STM32F107xx 器件嵌入式自举程序支持四种串行外设：USART1、USART2、CAN2 和 DFU (USB)。这意味着支持四种串行外设：USART1、USART2、CAN2 和 DFU (USB)。

下表显示了在系统存储器自举模式下 STM32F105xx 和 STM32F107xx 器件的自举程序需要使用的硬件资源。

表 6. 系统存储器自举模式下的 STM32F105xx/107xx 配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。仅用于 USART1 和 USART2 自举程序，以及在 CAN2、USB 检测期间用于 CAN 和 DFU 自举程序（选择 CAN 或 DFU 自举程序后，时钟源将由外部石英时钟提供）。
		HSE 使能	只有 DFU 和 CAN 自举程序必须使用外部时钟，外部时钟必须提供以下其中一种频率：8 MHz、14.7456 MHz 或 25 MHz。 对于 CAN 自举程序，使用 14.7456 MHz 作为 HSE 时，PLL 仅用于生成 48 MHz 的频率。 对于 DFU 自举程序，PLL 则用于从所有支持的外部时钟频率生成 48 MHz 的系统时钟。
		-	使能时钟安全系统 (CSS) 中断以用于 CAN 和 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	系统存储器	-	自地址 0x1FFF B000 起的 18 KB 空间包含自举程序固件。
	RAM	-	自地址 0x2000 0000 起的 4 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 用于接收。
	USART1_TX 引脚	推挽输出	PA9 引脚：USART1 用于发送。
	检测阶段期间，USART2_RX (PD6)、CAN2_RX (PB5)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 6. 系统存储器自举模式下的 STM32F105xx/107xx 配置（续）

自举程序	功能/外设	状态	备注
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 用于接收（重映射引脚）
	USART2_TX 引脚	推挽输出	PD5 引脚：USART2 用于发送（重映射引脚）
	检测阶段期间，USART1_RX (PA10)、CAN2_RX (PB5)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
CAN2 自举程序	CAN2	使能	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注： 在 STM32F105xx 和 STM32F107xx 器件中，由于 CAN1 管理 CAN2 和 SRAM 之间的通信，因此执行 CAN 自举程序期间要为 CAN1 提供时钟。
	CAN2_RX 引脚	输入	PB5 引脚：CAN2 用于接收（重映射引脚）。
	CAN2_TX 引脚	推挽输出	PB6 引脚：CAN2 用于发送（重映射引脚）。
	检测阶段期间，USART1_RX (PA10)、USART2_RX (PD6)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
DFU 自举程序	USB OTG FS	使能	USB OTG FS 配置为强制器件模式
	OTG_FS_VBUS 引脚	输入或复用功能，通过 USB OTG FS 控制器自动控制	PA9：电源电压线
	OTG_FS_DM 引脚		PA11：USB 发送接收数据线
	OTG_FS_DP 引脚		PA12：USB 发送接收数据线
	中断	使能	使能 USB_OTG_FS 中断向量以用于 USB DFU 通信。
	检测阶段期间，USART1_RX (PA10)、USART2_RX (PD6) 和 CAN2_RX (PB5) 引脚必须保持高电平或低电平。		

对于 USARTx 自举程序，系统时钟由嵌入式内部高速 RC 提供。该内部时钟同样用于 DFU 和 CAN 自举程序，但仅限选择阶段。选择阶段结束后，执行 DFU 和 CAN 自举程序需要使用外部时钟（8 MHz、14.7456 MHz 或 25 MHz）。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。
如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

5.2 自举程序硬件要求

将 STM32F105xx 和 STM32F107xx 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 **BOOT0** 引脚保持高电平，使 **BOOT1** 引脚保持低电平。

要在系统存储器自举模式下连接 STM32F105xx 和 STM32F107xx，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 **USART1** 连接自举程序，则检测阶段期间，**USART2_RX** (PD6)、**CAN2_RX** (PB5)、**OTG_FS_DM** (PA11) 和 **OTG_FS_DP** (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 **USART2** 连接自举程序，则检测阶段期间，**USART1_RX** (PA10)、**CAN2_RX** (PB5)、**OTG_FS_DM** (PA11) 和 **OTG_FS_DP** (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 **CAN2** 连接自举程序，则检测阶段期间，**USART1_RX** (PA10)、**USART2_RX** (PD6)、**OTG_FS_DM** (PA11) 和 **OTG_FS_DP** (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 **DFU** 连接自举程序，则检测阶段期间，**USART1_RX** (PA10)、**USART2_RX** (PD6) 和 **CAN2_RX** (PB5) 引脚必须保持高电平或低电平，并且不得悬空。
- 通过以下硬件连接外设：
 - **RS232** 串行接口（例如 **ST3232 RS232** 收发器），使用 **USART1** 时必须将该接口直接连接到 **USART1_RX** (PA10) 和 **USART1_TX** (PA9) 引脚；使用 **USART2** 时则必须直接连接到 **USART2_RX** (PD6) 和 **USART2_TX** (PD5) 引脚
 - **CAN** 接口（**CAN** 收发器），必须将该接口直接连接到 **CAN2_RX** (PB5) 和 **CAN2_TX** (PB6) 引脚
 - 经认证的 **USB** 电缆，必须使用此类电缆到连接微控制器（也可使用 **ESD** 保护电路）

USART1_CK、**USART1_CTS** 和 **USART1_RTS** 引脚未使用，因此应用程序可将这些引脚用于其它外设或 **GPIO**。该注释同样适用于 **USART2**。

使能 **USB** 设备后，其所有相关引脚专用于 **USB** 通信，无法用于其它应用目的。

用户可借助 **PC** 串行小应用程序，使用 **RS232** 串行接口分别通过 **CTS** 线和 **DCD** 线来控制 **BOOT0** 和复位引脚。用户必须使用全零调制解调器电缆。**STM3210C-EVAL** 板提供了实现此控制功能所需的硬件。相关详细信息，请参见文档：“**STM3210C-EVAL** 板用户手册”，可从意法半导体网站获取该手册：<http://www.st.com>。

5.3 自举程序选择

STM32F105xx 和 STM32F107xx 嵌入式自举程序支持四种外设接口：**USART1**、**USART2**、**CAN2** 和 **DFU (USB)**。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 **Flash**。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

注：如下文所述，检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 5.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

要使用 CAN2 接口，应将 CAN 电缆连接到 CAN2。自举程序在 CAN2_RX 引脚 (PB5) 检测到帧后，自举程序固件将进入 CAN 循环并开始检查外部时钟频率值，如果 HSE 为 8 MHz、14.7456 MHz 或 25 MHz，CAN 自举程序固件将进入无限循环并等待，直到接收到消息，否则将产生系统复位。

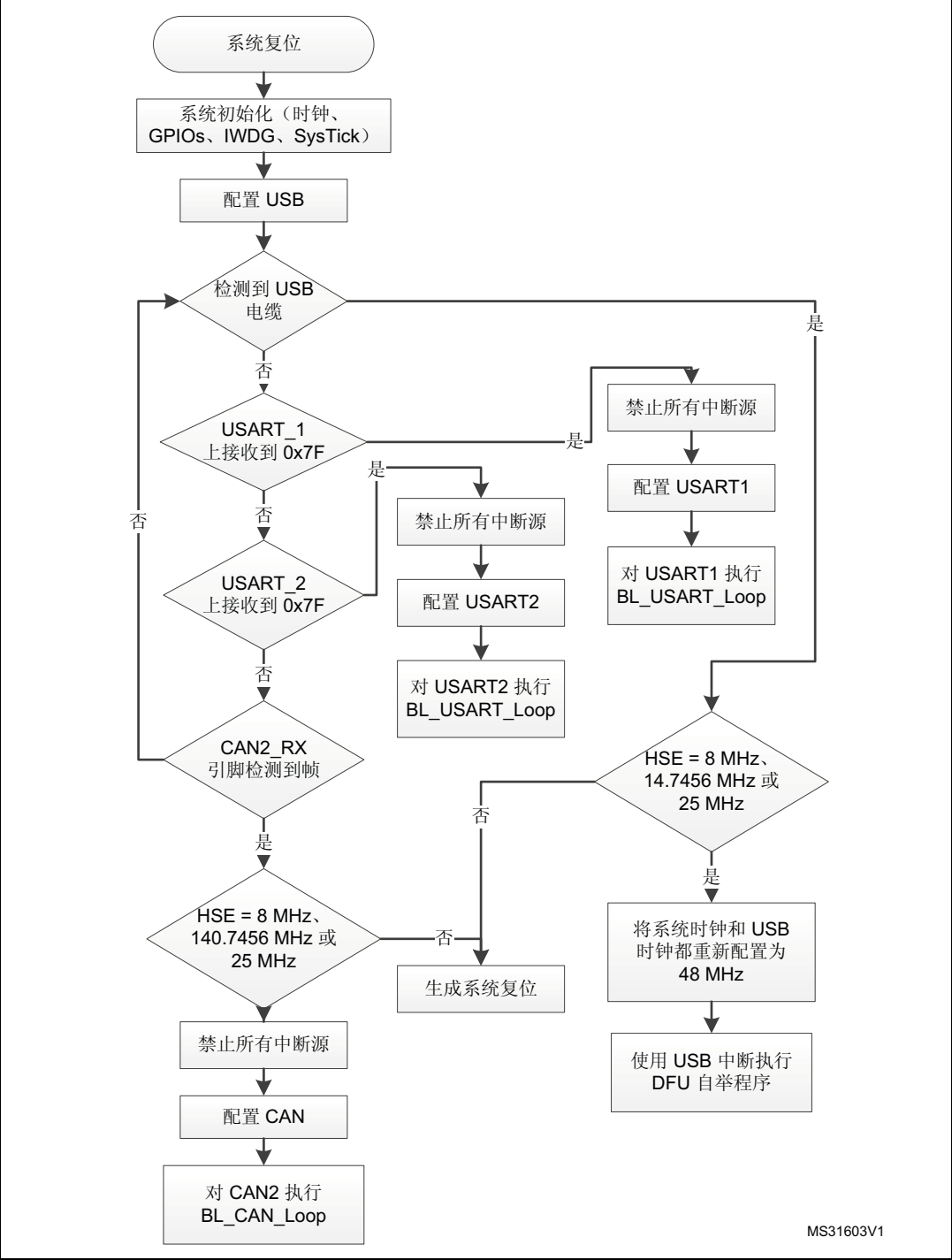
在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 或 CAN 自举程序，必须确保选择阶段没有任何 USB 电缆连接到 USB 外设。选择 USART 或 CAN 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，所有其它接口都禁止。

下图显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 2. STM32F105xx 和 STM32F107xx 器件的自举程序选择



5.4 自举程序版本

下表列出了 STM32F105xx 和 STM32F107xx 器件的自举程序版本和所有版本之间的变化。

表 7. STM32F105xx 和 STM32F107xx 自举程序版本

自举程序版本号	说明
V1.0	初始自举程序版本。
V2.0	<ul style="list-style-type: none"> – 更新自举程序检测机制，以修正检测阶段期间此自举程序未用外设的 GPIO 连接到低电平或悬空的问题。 有关详细信息，请参见第 5.4.2 节。 – 向量表设置为 0x1FFF B000 而不是 0x0000 0000 – 更新 Go 命令（针对所有自举程序）：USART1、USART2、CAN2、GPIOA、GPIOB、GPIOC 和 SysTick 外设寄存器设置为默认复位值 – DFU 自举程序：执行 Leave DFU 命令前将 USB 中断挂起清零 – DFU 子协议版本由 V1.0 变为 V1.2 – 自举程序版本更新为 V2.0
V2.1	<ul style="list-style-type: none"> – 修正第 5.4.4 节中所述的 PA9 过大功耗问题。 – 更正 Get-Version 命令（如 AN3155 中定义）。该命令将返回 0x22，而不是自举程序 V2.0 中的 0x20。更多详细信息，请参见第 5.4.3 节。 – 自举程序版本更新为 V2.1
V2.2	<ul style="list-style-type: none"> – 修正 DFU 选项字节描述符（由于可读 / 写但不可擦除，因此设置为“e”而不是“g”）。 – 修正用于 Flash 读/写/擦除操作的 DFU 轮询时序。 – 提高 DFU 自举程序接口的耐用性。 – 自举程序版本更新为 V2.2。

5.4.1 如何识别 STM32F105xx/107xx 自举程序版本

自举程序 V1.0 用于日期代码小于 937 的器件（有关如何查找器件标记上的日期代码，请参见 STM32F105xx 和 STM32F107xx 数据手册）。自举程序 V2.0 和 V2.1 用于日期代码大于或等于 937 的器件。

自举程序版本有两种区分方法：

- 使用 USART 自举程序时，V2.1 版本中更正了 AN2606 和 AN3155 中定义的 Get-Version 命令。该命令将返回 0x22，而不是自举程序 V2.0 中的 0x20。
- 自举程序代码开始位置的向量表中的值有所不同。对于自举程序 V2.0、V2.1 和 V2.2，用户软件（或通过 JTAG/SWD）在地址 0x1FFFB004 处读取的分别是 0x1FFFE945、0x1FFFE9A1 和 0x1FFFE9C1。
- DFU 版本如下：
 - 自举程序 V2.1 中为 V2.1 版
 - 自举程序 V2.2 中为 V2.2 版
 可从 DFU 器件描述符的 bcdDevice 字段中读取该版本。

5.4.2 日期代码小于 937 的 STM32F105xx/STM32F107xx 器件上自举程序不可用

说明

激活自举程序阶段，如果 USART1_RX (PA10)、USART2_RX (PD6, 重映射)、CAN2_Rx (PB5, 重映射)、OTG_FS_DM (PA11) 和 / 或 OTG_FS_DP (PA12) 引脚保持低电平或悬空，则自举程序不可用。

自举程序无法通过 CAN2 (重映射)、DFU (器件模式 OTG FS)、USART1 或 USART2 (重映射) 进行连接。

在 64 引脚封装中，USART2_RX 信号重映射引脚 PD6 不可用，而是在内部接地。这种情况下，自举程序完全不可用。

解决方案

- 对于 64 引脚封装。
无。自举程序不可用。
- 对于 100 引脚封装。
根据所用外设的不同，必须按照下述要求在激活自举程序阶段使未使用的外设引脚保持高电平：
 - 如果使用 USART1 连接自举程序，则 PD6 和 PB5 必须保持高电平。
 - 如果使用 USART2 连接自举程序，则 PA10、PB5、PA11 和 PA12 必须保持高电平。
 - 如果使用 CAN2 连接自举程序，则 PA10、PD6、PA11 和 PA12 必须保持高电平。
 - 如果使用 DFU 连接自举程序，则 PA10、PB5 和 PD6 必须保持高电平。

注：此限制仅适用于日期代码小于 937 的 STM32F105xx 和 STM32F107xx 器件。日期代码大于或等于 937 的 STM32F105xx 和 STM32F107xx 器件并不受此影响。有关如何查找器件标记上的日期代码，请参见 STM32F105xx 和 STM32F107xx 数据手册。

5.4.3 USART 自举程序 Get-Version 命令返回 0x20 而不是 0x22

说明

在 USART 模式下，Get-Version 命令（如 AN3155 中定义）返回 0x20 而不是 0x22。

此限制存在于自举程序版本 V1.0 和 V2.0 中，自举程序版本 2.1 对此进行了修正。

解决方案

无。

5.4.4 自举程序 V2.0 中插入 USB 电缆时 PA9 功耗过大

说明

从系统存储器模式自举后连接 USB 电缆时，PA9 引脚（连接 $V_{BUS}=5\text{ V}$ ）同时与配置为复用推挽的 USART TX 引脚共用，由于尚未对 USART 外设提供时钟，因此该引脚强制设置为 0。因而流经 PA9 I/O 的漏电流超出 25 mA，可能影响 I/O 板的可靠性。

自举程序版本 2.1 对此限制进行了修正，实现方法为：在 RX 引脚接收到正确的 0x7F 且对 USART 提供时钟时，将 PA9 配置为复用功能推挽。否则将 PA9 配置为复用输入悬空。

解决方案

无。

6 STM32F101xx 和 STM32F103xx 超大容量器件自举程序

本部分使用 STM32F10xxx 超大容量指代超大容量的 STM32F101xx 和 STM32F103xx 器件。

6.1 双存储区自举功能

对于 STM32F101xx 和 STM32F103xx 超大容量器件（这些器件具有两个 Flash 存储区：存储区 1 和存储区 2），可使用允许从存储区 1 或存储区 2 进行自举的附加自举机制（取决于 BFB2 位的状态（0x1FFFF800 处用户选项字节的位 19））。

- 当 BFB2 位置 1 且将自举引脚配置为从 Flash 自举（BOOT0 = 0 且 BOOT1 = x）时，复位后，器件从系统存储器自举，并执行实施双存储区自举模式的嵌入式自举程序代码：
 - 首先检查存储区 2 中的代码。如果其中包含有效代码（请参见下方 [注：](#)），则跳转到存储区 2 中的应用程序并退出自举程序。
 - 如果存储区 2 中的代码无效，则检查存储区 1 中的代码。如果代码有效（请参见下方“[注释](#)”），则跳转到存储区 1 中的应用程序。
 - 如果存储区 2 和存储区 1 均不包含有效代码（请参见下方“[注释](#)”），则按照以下部分所述方式正常执行自举程序操作（不跳转到 Flash 存储区）。更多详细信息，请参见 [图 3：STM32F10xxx 超大容量器件的自举程序选择](#)。
- BFB2 位置 1 时（默认状态），不执行双存储区自举机制。

注：当第一个数据（位于存储区起始地址处，应为堆栈指针）指向内部 SRAM 存储器的有效地址时（栈顶地址），该代码视为有效。如果第一个地址指向其它任何位置（内部 SRAM 以外），该代码则视为无效。

<http://www.st.com> 网站提供的 STM32F10x 标准外设库中介绍了双存储区自举模式示例 (FLASHDual_Boot)。

对于 STM32F101xx 和 STM32F103xx 超大容量器件，可选择 Flash、系统存储器或 SRAM 作为自举空间，如以下 [表 8](#) 所示。

表 8. 自举引脚和 BFB2 位配置

BFB2 位	自举模式选择引脚		自举模式	别名使用
	BOOT1	BOOT0		
1	X	0	用户 Flash	选择用户 Flash 作为自举空间
	0	1	系统存储器	选择系统存储器作为自举空间
	1	1	嵌入式 SRAM	选择嵌入式 SRAM 作为自举空间
0	X	0	系统存储器	选择系统存储器作为自举空间，然后执行双存储区机制
	0	1	系统存储器	选择系统存储器作为自举空间，然后执行双存储区机制
	1	1	嵌入式 SRAM	选择嵌入式 SRAM 作为自举空间

表 8 显示了超大容量器件进入系统存储器自举模式的两种情况：

1. 如果 BOOT 引脚配置如下：BOOT0 = 1 且 BOOT1 = 0
2. 或者如果：
 - a) BFB2 位复位且
 - b) 自举引脚配置如下：BOOT0 = 0 且 BOOT1 = x

注：满足下述 a、b 和 c 三个条件时，相当于将自举引脚配置为系统存储器自举模式（BOOT0 = 1 且 BOOT1 = 0）。此时将执行正常的自举程序操作。

- a) BFB2 位复位
- b) 两个存储区均不包含有效代码
- c) 自举引脚配置如下：BOOT0 = 0 且 BOOT1 = x

当 BFB2 位清零，且存储区 2 和/或存储区 1 包含有效用户应用程序代码时，始终执行双存储区自举模式（自举程序始终跳转到用户代码，而不会继续执行正常操作）。

因此，如果已将 BFB2 位清零（从存储区 2 自举），则要执行自举程序代码时必须：

——将地址 0x0808 0000（存储区 2 的基址）和 0x0800 0000（存储区 1 的基址）编程为 0x0，或者

——将 BFB2 位置 1，使 BOOT0 = 1 且 BOOT1 = 0。

6.2 自举程序配置

STM32F10xxx 超大容量器件嵌入式自举程序支持两种串行接口：USART1 和 USART2。

下表介绍了系统存储器自举模式下，自举程序需要使用的 STM32F10xxx 超大容量器件硬件资源。

表 9. 系统存储器自举模式下 STM32F10xxx 超大容量器件的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	时钟源	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。
	RAM	-	自地址 0x2000 0000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFF E000 起的 6 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 用于接收。
	USART1_TX 引脚	推挽输出	PA9 引脚：USART1 用于发送。
	检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平。		

表 9. 系统存储器自举模式下 STM32F10xxx 超大容量器件的配置（续）

自举程序	功能/外设	状态	备注
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。
	USART2_RX 引脚	输入	PD6 引脚：USART2 用于接收（重映射引脚）。
	USART2_TX 引脚	推挽输出	PD5 引脚：USART2 用于发送（重映射引脚）。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测主机串口波特率。

系统时钟由嵌入式内部高速 RC 提供，自举程序代码不需要外部石英时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如表 9 所示）将初始化为默认复位值。
如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

6.3 自举程序硬件要求

将 STM32F10xxx 超大容量器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

注：如第 6.1 节：双存储区自举功能所述，当 BFB2 位复位，擦除了两个存储区的起始地址，并将自举引脚配置为从 Flash 自举时，也可以通过软件执行系统存储器自举模式。

要在系统存储器自举模式下连接 STM32F10xxx 超大容量器件，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 当 BFB2 位清零，且存储区 2 和 / 或存储区 1 包含有效用户应用程序代码时，始终执行双存储区自举模式（自举程序始终跳转到用户代码，而不会继续执行正常操作）。因此，如果已将 BFB2 位清零（从存储区 2 自举），则要执行自举程序代码时必须：
 - 将地址 0x0808 0000（存储区 2 的基址）和 0x0800 0000（存储区 1 的基址）编程为 0x0，或者
 - 将 BFB2 位置 1，使 BOOT0 = 1 且 BOOT1 = 0。
- 通过以下硬件连接外设：
 - RS232 串行接口（例如 ST3232 RS232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。

6.4 自举程序选择

STM32F10xxx 超大容量嵌入式自举程序支持两种外设接口：USART1 和 USART2。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

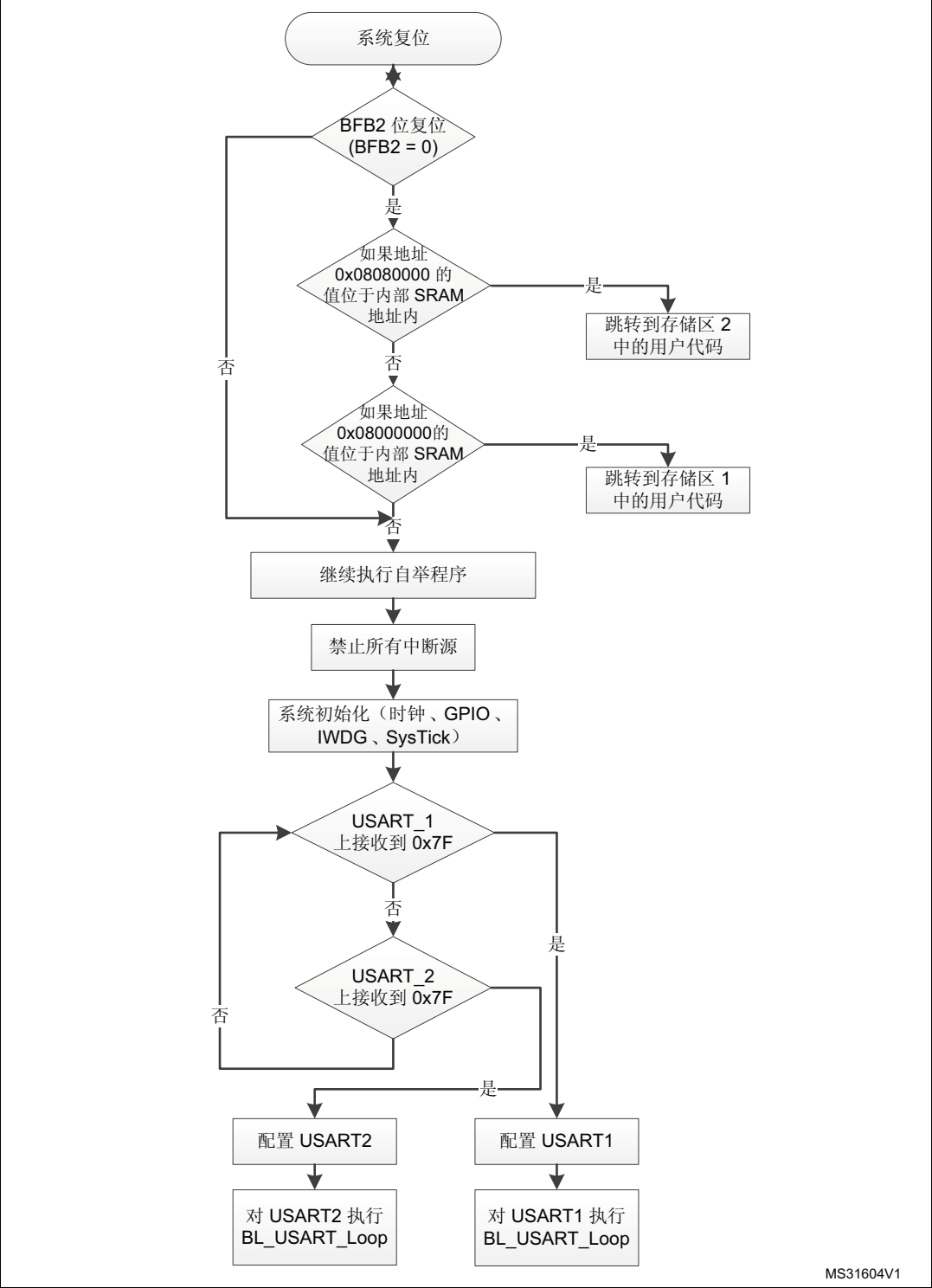
注：如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 6.3 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

为自举程序选择一个接口后，其它接口将禁止。

[图3](#) 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 3. STM32F10xxx 超大容量器件的自举程序选择



6.5 自举程序版本

表 10 列出了 STM32F101xx 和 STM32F103xx 超大容量器件的自举程序版本。

表 10. 超大容量器件自举程序版本

自举程序版本号	说明
V2.1	初始自举程序版本

7 STM32L151xx 和 STM32L152xx 中容量超低功耗器件自举程序

本部分使用 STM32L15xxx 中容量器件指代中容量 STM32L151xx 和 STM32L152xx 超低功耗器件。

7.1 自举程序配置

STM32L15xxx 中容量器件嵌入式自举程序支持两种串行接口：USART1 和 USART2 外设。
下表介绍了系统存储器自举模式下，自举程序需要使用的 STM32L15xxx 中容量器件硬件资源。

表 11. 系统存储器自举模式下 STM32L15xxx 器件的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	时钟源	HSI 使能	系统时钟频率等于 16 MHz。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FF00000 起的 4 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为电压范围 1。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 用于接收。
	USART1_TX 引脚	输出	PA9 引脚：USART1 用于发送。
	检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。
	USART2_RX 引脚	输入	PD06 引脚：USART2 用于接收。
	USART2_TX 引脚	输出	PD05 引脚：USART2 用于发送。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测主机串口波特率。

系统时钟由嵌入式内部高速 RC 提供，自举程序代码不需要外部石英时钟。
下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

7.2 自举程序硬件要求

将 STM32L15xxx 中容量器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

要在系统存储器自举模式下连接 STM32L15xxx 中容量器件，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 要使用的外设必须通过 RS-232 串行接口（例如 ST3232 RS-232 收发器）连接，并且：
 - 使用 USART1 时，该串行接口必须直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚
 - 使用 USART2 时，该串行接口必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART2。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM32L152-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见“STM32L152-EVAL 板用户手册”(UM1018)，可从意法半导体网站获取该手册：<http://www.st.com>。

7.3 自举程序选择

STM32L15xxx 中容量器件的嵌入式自举程序支持两种外设接口：USART1 和 USART2。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 7.2 节：自举程序硬件要求。

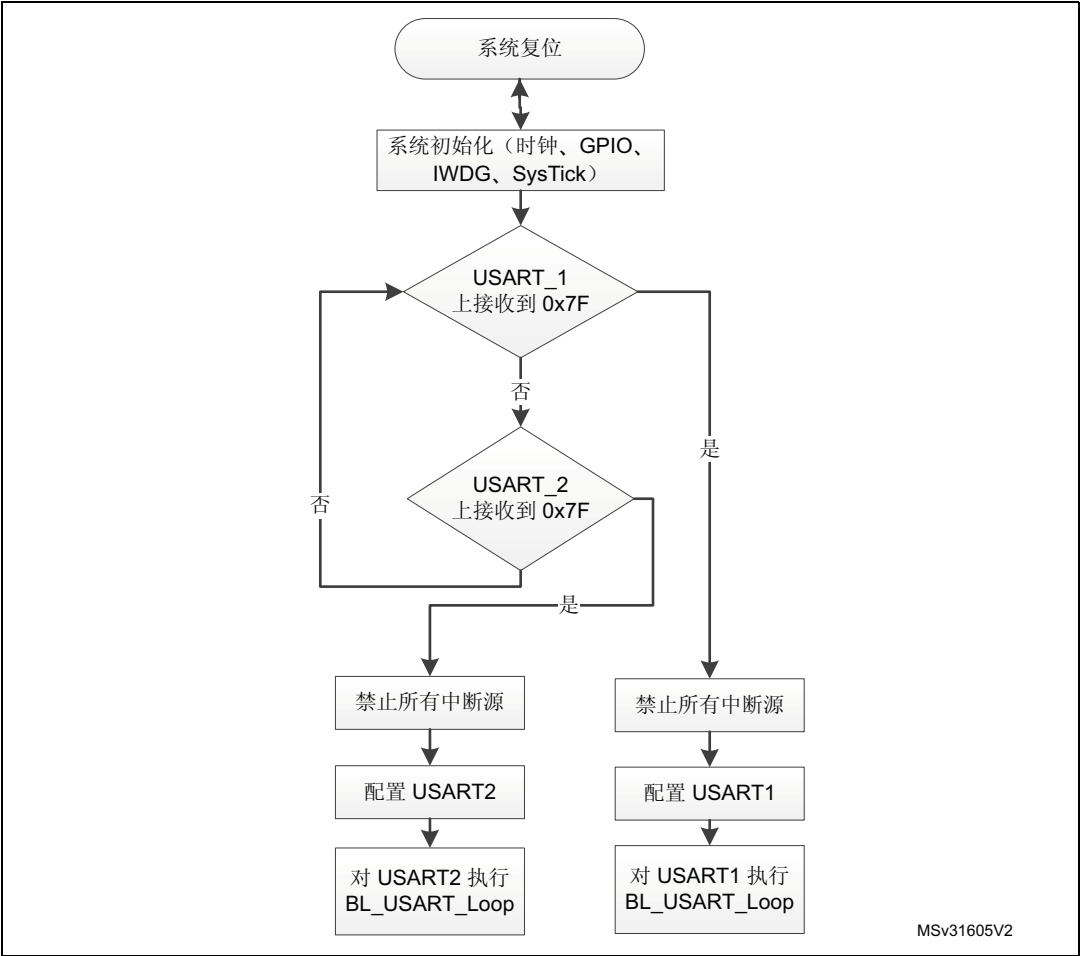
要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。

自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

为自举程序选择一个接口后，其它接口将禁止。

下图显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 4. STM32L15xxx 中容量器件的自举程序选择



7.4 重要事项

中容量超低功耗器件的自举程序有一些特定的功能需要格外注意，如下所述：

- **STM32L15xxx 中容量器件自举程序固件**除了支持标准存储器（内部 Flash、内部 SRAM、可选字节和系统存储器）外，还支持数据存储器（从 0x08080000 到 0x08080FFF 的 4 KB 空间）。有关详细信息，请参见 PM0062 编程手册。
- **Flash 写操作**通过程序存储器的半页写操作执行。自举程序固件按非对齐地址管理半页写操作。因此，所有写操作只能采用字对齐形式（地址应为 4 的倍数）。要写入的数据量也必须为 4 的倍数（允许非对齐半页写操作地址）。请参见产品数据手册，了解写操作所需的持续时间。
- **数据存储器**允许执行读和写操作，但不能使用 Erase 命令进行擦除。向数据存储单元执行写操作时，自举程序固件会在任何写操作之前对此单元执行擦除操作。数据存储器的写操作必须采用字对齐形式（要写入的地址应为 4 的倍数），并且要写入的数据量也必须为 4 的倍数。要擦除数据存储单元，可在此单元中写入零。

- **选项字节**
地址为 0x1FF80000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
 有关保护级别的详细信息，请参见 PM0062 编程手册。
- **读保护**命令对应级别 1 保护。
- **禁止读保护**命令对应级别 0 保护。
- **批量擦除**命令尚未获得 STM32L15xxx 中容量器件自举程序固件的支持。要执行批量擦除操作，可采用以下两种方法：
 - 使用 Erase 命令逐个擦除所有扇区
 - 将保护级别设为级别 1。然后再设为级别 0（使用 Read protect 命令，然后使用 Read Unprotect 命令）。执行此操作后将批量擦除内部 Flash 内容（有关详细信息，请参见 PM0062 编程手册）。

7.5 自举程序版本

下表列出了 STM32L15xxx 中容量器件自举程序版本。

表 12. STM32L15xxx 中容量器件自举程序版本

自举程序版本号	说明	已知限制
V2.0	初始自举程序版本。	当通过一个不受支持的存储器地址和一个正确的地址校验和（即，地址 0x6000 0000）发出 Read Memory 命令或 Write Memory 命令时，自举程序器件会中止这一命令，但不会向主机发送 NACK (0x1F)。因此，接下来的两个字节（即，待读/写的字节数及其校验和）会被看作一条新命令及其校验和。 ⁽¹⁾

1. 如果待读/写的“数据数 - 1” (N-1) 不等于有效命令代码（0x00、0x01、0x02、0x11、0x21、0x31、0x43、0x44、0x63、0x73、0x82 或 0x92），则无法从主机发现缺陷，因为该命令始终会收到 NACK 应答（当作不受支持的新命令）。

8 STM32L151xx、STM32L152xx 和 STM32L162xx 大容量超低功耗器件自举程序

本部分使用 STM32L1xxxx 大容量器件指代 STM32L151xx、STM32L152xx 和 STM32L162xx 大容量超低功耗器件。

8.1 双存储区自举功能

STM32L1xxxx 大容量器件具有两个 Flash 存储区：存储区 1 和存储区 2。这两个存储区采用了附加的自举机制，允许根据 BFB2 位的状态（0x1FF8 0004 处用户选项字节的位 7）从存储区 2 或存储区 1 进行自举。

- 当 BFB2 位复位且将自举引脚配置为从 Flash 自举（BOOT0 = 0 且 BOOT1 = x）时，复位后，器件从系统存储器自举，并执行实施双存储区自举模式的嵌入式自举程序代码：
 - a) 首先检查存储区 2 中的代码。如果其中包含有效代码（请参见下方注释），则跳转到存储区 2 中的应用程序代码并退出自举程序。
 - b) 如果存储区 2 中的代码无效，则检查存储区 1 中的代码。如果代码有效（请参见下方注释），则跳转到存储区 1 中的应用程序。
 - c) 如果存储区 2 和存储区 1 均不包含有效代码（请参见下方注释），则按照以下部分所述方式正常执行自举程序操作，不跳转到 Flash 存储区。有关详细信息，请参见 [图 5：STM32L1xxxx 大容量器件的自举程序选择](#)。

3. BFB2 位置 1 时（默认状态），不执行双存储区自举机制。

注：当第一个数据（位于存储区起始地址处，应为堆栈指针）指向内部 SRAM 存储器的有效地址时（栈顶地址），该代码视为有效。如果第一个地址指向其它任何位置（内部 SRAM 以外），该代码则视为无效。

<http://www.st.com> 网站提供的 STM32L1xxxx 标准外设库中介绍了双存储区自举模式示例（FLASH\Dual_Boot）。

对于 STM32L1xxxx 大容量器件，可选择 Flash、系统存储器或 SRAM 作为自举空间，如以下 [表 13](#) 所示。

表 13. 自举引脚和 BFB2 位配置

BFB2 位	自举模式选择引脚		保护级别 2	存储区 2 有效	存储区 1 有效	自举模式	别名使用
	BOOT1	BOOT0					
1	X	0	X	X	X	用户 Flash	选择用户 Flash 存储区 1 作为自举空间
	0	1	无	X	X	系统存储器	选择系统存储器作为自举空间
	0	1	有	X	X	用户 Flash	选择用户 Flash 存储区 1 作为自举空间
	1	1	无	X	X	嵌入式 SRAM	选择嵌入式 SRAM 作为自举空间
	1	1	有	X	X	用户 Flash	选择用户 Flash 存储区 1 作为自举空间

表 13. 自举引脚和 BFB2 位配置（续）

BFB2 位	自举模式选择引脚		保护级别 2	存储区 2 有效	存储区 1 有效	自举模式	别名使用
	BOOT1	BOOT0					
0	X	0	X	有	X	系统存储器	选择用户 Flash 存储区 2 作为自举空间
				无	有	系统存储器	选择用户 Flash 存储区 1 作为自举空间
			无	无	无	系统存储器	选择系统存储器作为自举空间
			有	无	无	系统存储器	CPU 阻塞（停止）
	0	1	无	X	X	系统存储器	选择系统存储器作为自举空间
	1	1	无	X	X	嵌入式 SRAM	选择嵌入式 SRAM 作为自举空间
	X	1	有	有	X	系统存储器	选择用户 Flash 存储区 2 作为自举空间
				无	有	系统存储器	选择用户 Flash 存储区 1 作为自举空间
				无	无	系统存储器	CPU 阻塞（停止）

表 13 显示了 STM32L1xxxx 大容量器件进入系统存储器自举模式的三种情况：

- 如果 BOOT 引脚配置如下：
BOOT0 = 1 且 BOOT1 = 0
- 如果 BFB2 位复位且使能保护级别 2
- 如果 BFB2 位复位且自举引脚配置如下：
BOOT0 = 0 且 BOOT1 = x

注：满足下述 a、b 和 c 三个条件时，相当于将自举引脚配置为系统存储器自举模式（BOOT0 = 1 且 BOOT1 = 0）。此时将执行正常的自举程序操作。

- BFB2 位复位
- 两个存储区均不包含有效代码
- 自举引脚配置如下：BOOT0 = 0 且 BOOT1 = x

当 BFB2 位清零，且存储区 2 和/或存储区 1 包含有效用户应用程序代码时，始终执行双存储区自举模式（自举程序始终跳转到用户代码，而不会继续执行正常操作）。

因此，如果已将 BFB2 位清零（从存储区 2 自举），则要执行自举程序代码时必须：

- 将地址 0x0808 0000（存储区 2 的基址）和 0x0800 0000（存储区 1 的基址）编程为 0x0，或者
- 将 BFB2 位置 1，使 BOOT0 = 1 且 BOOT1 = 0。

8.2 自举程序配置

STM32L1xxxx 大容量器件嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。

下表介绍了系统存储器自举模式下，自举程序需要使用的 STM32L1xxxx 大容量器件硬件资源。

表 14. 系统存储器自举模式下 STM32L1xxxx 大容量器件的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 HSI 作为系统时钟，频率为 16 MHz。仅用于 USART1 和 USART2 自举程序以及在 USB 检测期间用于 DFU 自举程序（选择 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	只有 DFU 自举程序必须使用外部时钟，外部时钟必须介于以下频率范围：[24, 16, 12, 8, 6, 4, 3, 2] MHz。 PLL 用于生成 48 MHz USB 时钟和 32 MHz 系统时钟。
		-	使能时钟安全系统 (CSS) 中断以用于 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源		电压范围设置为电压范围 1。
	系统存储器	-	自地址 0x1FF0 0000 起的 8 KB 空间。此区域包含自举程序固件。
	RAM	-	自地址 0x2000 0000 起的 4 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		

表 14. 系统存储器自举模式下 STM32L1xxxx 大容量器件的配置（续）

自举程序	功能/外设	状态	备注
DFU 自举程序	USB_DM 引脚	输入或复用功能，通过 USB 自动控制	PA11: USB 发送接收数据线
	USB_DP 引脚		PA12: USB 发送接收数据线
	中断	使能	使能 USB 低优先级中断向量以用于 USB DFU 通信。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

注：对于 DFU 接口，需要提供外部时钟源 (HSE) 才能执行 USB 操作。HSE 值的检测通过自举程序固件完成，该过程以内部振荡器时钟 (HSI、MSI) 为依据。因此，当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时（理论值约为 1%），自举程序计算所得的 HSE 频率值可能不正确。这种情况下，自举程序 DFU 接口会出现故障，甚至完全无法工作。

对于 USARTx 自举程序，系统时钟由内部高速 RC 提供。该内部时钟同样用于 DFU 自举程序，但仅限选择阶段。选择阶段结束后，执行 DFU 自举程序时需要频率介于范围 [24, 16, 12, 8, 6, 4, 3, 2] MHz 的外部时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

8.3 自举程序硬件要求

将 STM32L1xxxx 大容量器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

注：如第 8.1 节：双存储区自举功能所述，当 BFB2 位复位，擦除了两个存储区的起始地址，并将自举引脚配置为从 Flash 自举时，也可以通过软件执行系统存储器自举模式。

要在系统存储器自举模式下连接 STM32L1xxxx 大容量器件，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 DFU (USB) 连接自举程序，则检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。

- 当 BFB2 位清零，且存储区 2 和 / 或存储区 1 包含有效用户应用程序代码时，始终执行双存储区自举模式（自举程序始终跳转到用户代码，而不会继续执行正常操作）。因此，如果已将 BFB2 位清零（从存储区 2 自举），则要执行自举程序代码时必须：
 - 将地址 0x0808 0000（存储区 2 的基址）和 0x0800 0000（存储区 1 的基址）编程为 0x0，或者
 - 将 BFB2 位置 1，使 BOOT0 = 1 且 BOOT1 = 0。
- 通过以下硬件连接外设：
 - RS232 串行接口（例如 ST3232 RS232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。

8.4 自举程序选择

STM32L1xxxx 大容量器件的嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 8.3 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

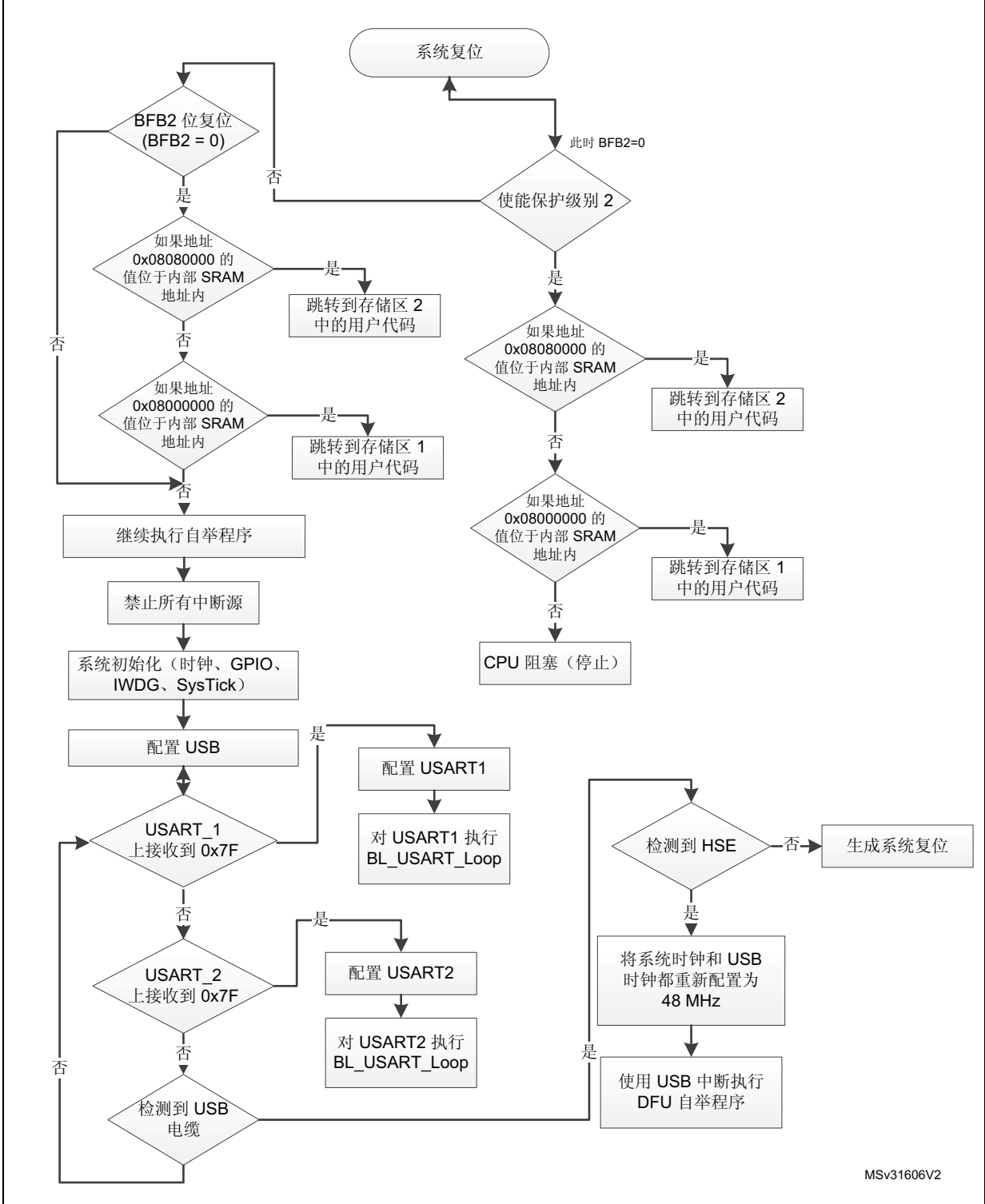
在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 自举程序，必须确保选择阶段没有任何 USB 主机连接到 USB 外设。选择 USART 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，其它接口将禁止。

[图 5：STM32L1xxxx 大容量器件的自举程序选择](#)显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 5. STM32L1xxxx 大容量器件的自举程序选择



8.5 重要事项

STM32L1xxxx 大容量器件的自举程序有一些特定的功能需要格外注意，如下所述：

- **STM32L1xxxx 大容量器件自举程序固件**除了支持标准存储器（内部 Flash、内部 SRAM、可选字节和系统存储器）外，还支持数据存储器（从 0x08080000 到 0x08082FFF 的 12 KB 空间）。有关详细信息，请参见 PM0062 编程手册。
- **Flash 写操作**通过程序存储器的半页写操作执行。自举程序固件按非对齐地址管理半页写操作。因此，所有写操作只能采用字对齐形式（地址应为 4 的倍数）。要写入的数据量也必须为 4 的倍数（允许非对齐半页写操作地址）。请参见产品数据手册，了解写操作所需的持续时间。
- **数据存储器**允许执行读和写操作，但不能使用 Erase 命令进行擦除。向数据存储单元执行写操作时，自举程序固件会在任何写操作之前对此单元执行擦除操作。数据存储器的写操作必须采用字对齐形式（要写入的地址应为 4 的倍数），并且要写入的数据量也必须为 4 的倍数。要擦除数据存储单元，可在此单元中写入零。
- **选项字节**
地址为 0x1FF80000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2有关保护级别的详细信息，请参见 PM0062 编程手册。
- **读保护**命令对应级别 1 保护。
- **禁止读保护**命令对应级别 0 保护。
- **批量擦除**命令未获得 STM32L1xxxx 大容量器件自举程序固件的支持。要执行批量擦除操作，可采用以下两种方法：
 - 使用 Erase 命令逐个擦除所有扇区。
 - 将保护级别设为级别 1。然后再设为级别 0（使用 Read protect 命令，然后使用 Read Unprotect 命令）。执行此操作后将批量擦除内部 Flash 内容（有关详细信息，请参见 PM0062 编程手册）。

8.6 自举程序版本

表 15 列出了 STM32L1xxxx 大容量器件的自举程序版本。

表 15. STM32L1xxxx 大容量器件自举程序版本

自举程序版本号	说明	已知限制
V4.1	初始自举程序版本	<ul style="list-style-type: none"> – 自举程序代码中 PA13 (JTMS/SWDIO) I/O 输出速度配置为 400 KHz，因此，在串行线模式下，一些调试器将无法在自举程序运行期间与器件相连。 – 选择 DFU 自举程序时，RTC 将复位，进而丢失所有 RTC 信息（日历、闹钟...），包括备份寄存器内容。注：选择 USART 自举程序时，RTC 配置（包括备份寄存器）保持不变。
V4.2	修正 V4.1 限制（仅适用于第 Z 版器件）。	<ul style="list-style-type: none"> – 当 BFB2=0 或读保护级别设为 2 时，跳转到存储区 1/ 存储区 2 后，堆栈上溢 8 字节。 解决方案：跳转到主程序之前，用户代码应强制放入启动文件栈顶地址。这可以在“Reset_Handler”程序中实现。 – 当用户代码的堆栈位于 SRAM 之外时（即地址 0x2000C000），此代码视为无效，因此自举程序不会跳转到该用户代码。这种情况可能在使用编译器时发生，因为此时会将堆栈置于 SRAM 栈顶的非物理地址（即地址 0x2000C000）处。 解决方案：将堆栈手动置于物理地址处。
V4.5	修正 V4.2 限制。 提高 DFU 接口耐用性（仅适用于第 Y 版器件）。	<ul style="list-style-type: none"> – 对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号（而不是 1 个 NACK 信号）。

9 STM32F205/215xx 和 STM32F207/217xx 自举程序

本部分使用 STM32F2xxxx 指代 STM32F205xx、STM32F207xx、STM32F215xx 和 STM32F217xx 器件。

STM32F2xxxx 器件可使用两种自举程序版本：

- V2.0 支持 USART1 和 USART3
此版本内置在 STM32F2xxxx 器件 B 版本中。
- V3.2 支持 USART1、USART3、CAN2 和 DFU（USB FS 设备）
此版本内置在 STM32F2xxxx 器件 X 和 Y 版本中。

9.1 自举程序 V2.x

9.1.1 自举程序配置

STM32F2xxxx 器件嵌入式自举程序 V2.x 支持两种串行接口：USART1 和 USART3 外设。

下表介绍了系统存储器自举模式下，自举程序 V2.x 需要使用的 STM32 器件硬件资源。

表 16. 系统存储器自举模式下 STM32F2xxxx 的配置

自举程序	功能/外设	状态	备注
适用于所有 自举程序	时钟源	HSI 使能	系统时钟频率等于 24 MHz。
	RAM	-	自地址 0x2000 0000 起的 8 KB 空间。
	系统存储器	-	自地址 0x1FFF 0000 起的 30688 字节空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为：[1.8V, 2.1V]（可使用自举程序命令在运行期间配置电压范围。）请注意，此范围内仅支持按照字节格式（不支持半字、字和双字操作）对内部 Flash 执行写操作。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 用于接收。
	USART1_TX 引脚	输出	PA9 引脚：USART1 用于发送。
	检测阶段期间，USART3_RX (PB11) 和 USART3_RX (PC11) 引脚必须保持高电平或低电平。		
USART3 自举程序 (PC10/PC11)	USART3	使能	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PC11 引脚：USART3 用于接收。
	USART3_TX 引脚	输出	PC10 引脚：USART3 用于发送。
	检测阶段期间，USART1_RX (PA10) 和 USART3_RX (PB11) 引脚必须保持高电平或低电平。		

表 16. 系统存储器自举模式下 STM32F2xxxx 的配置 (续)

自举程序	功能/外设	状态	备注
USART3 自举程序 (PB10/PB11)	USART3	使能	初始化后, USART3 的配置为: 8 位, 偶校验位和 1 个停止位
	USART3_RX 引脚	输入	PB11 引脚: USART3 用于接收。
	USART3_TX 引脚	输出	PB10 引脚: USART3 用于发送。
	检测阶段期间, USART1_RX (PA10) 和 USART3_RX (PC11) 引脚必须保持高电平或低电平。		
USART1 和 USART3 自举程序	SysTick 定时器	使能	用于自动检测主机串口波特率。

系统时钟由嵌入式内部高速 RC 提供。因此, 执行自举程序代码时无需外部石英时钟。

下载应用程序二进制代码后, 如果选择执行 Go 命令, 则在跳转到用户应用程序之前, 自举程序使用的外设寄存器 (如上表所示) 不会初始化为默认复位值。如果要使用这些寄存器, 应在用户应用程序中对其重新配置。因此, 如果应用程序正在使用 IWDG, 则必须调整 IWDG 预分频器值来满足应用程序的要求 (因为自举程序已将预分频器设置为最大值)。

9.1.2 自举程序硬件要求

将 STM32F2xxxx 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线, 它们能够在复位期间使 BOOT0 引脚保持高电平, 使 BOOT1 引脚保持低电平。

要在系统存储器自举模式下连接 STM32F2xxxx, 必须确认以下条件:

- 检测阶段期间, 此自举程序未使用的外设 RX 引脚必须保持已知电平 (低电平或高电平) 且不得悬空, 如下所述:
 - 如果使用 USART1 连接自举程序, 则检测阶段期间, USART3_RX (PC11 和 PB11) 引脚必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 USART3 (PB10/PB11) 连接自举程序, 则检测阶段期间, USART1_RX (PA10) 和另一 USART3_RX 引脚 (PC11) 必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 USART3 (PC10/PC11) 连接自举程序, 则检测阶段期间, USART1_RX (PA10) 和另一 USART3_RX 引脚 (PB11) 必须保持高电平或低电平, 并且不得悬空。
- 通过以下硬件连接外设:
 - RS232 串行接口 (例如 ST3232 RS232 收发器), 使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚; 使用 USART3 时则必须直接连接到 USART3_RX (PB11 或 PC11) 和 USART3_TX (PB10 或 PC10) 引脚。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用, 因此应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART3。

用户可借助 PC 串行小应用程序, 使用 RS232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM322xG-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息, 请参见 STM322xG-EVAL 板用户手册, 可从意法半导体网站获取该手册: <http://www.st.com>。

9.1.3 自举程序选择

STM32F2xxxx 嵌入式自举程序 V2.x 支持两种外设接口：USART1 和 USART3（PB10/PB11 和 PC10/PC11）。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

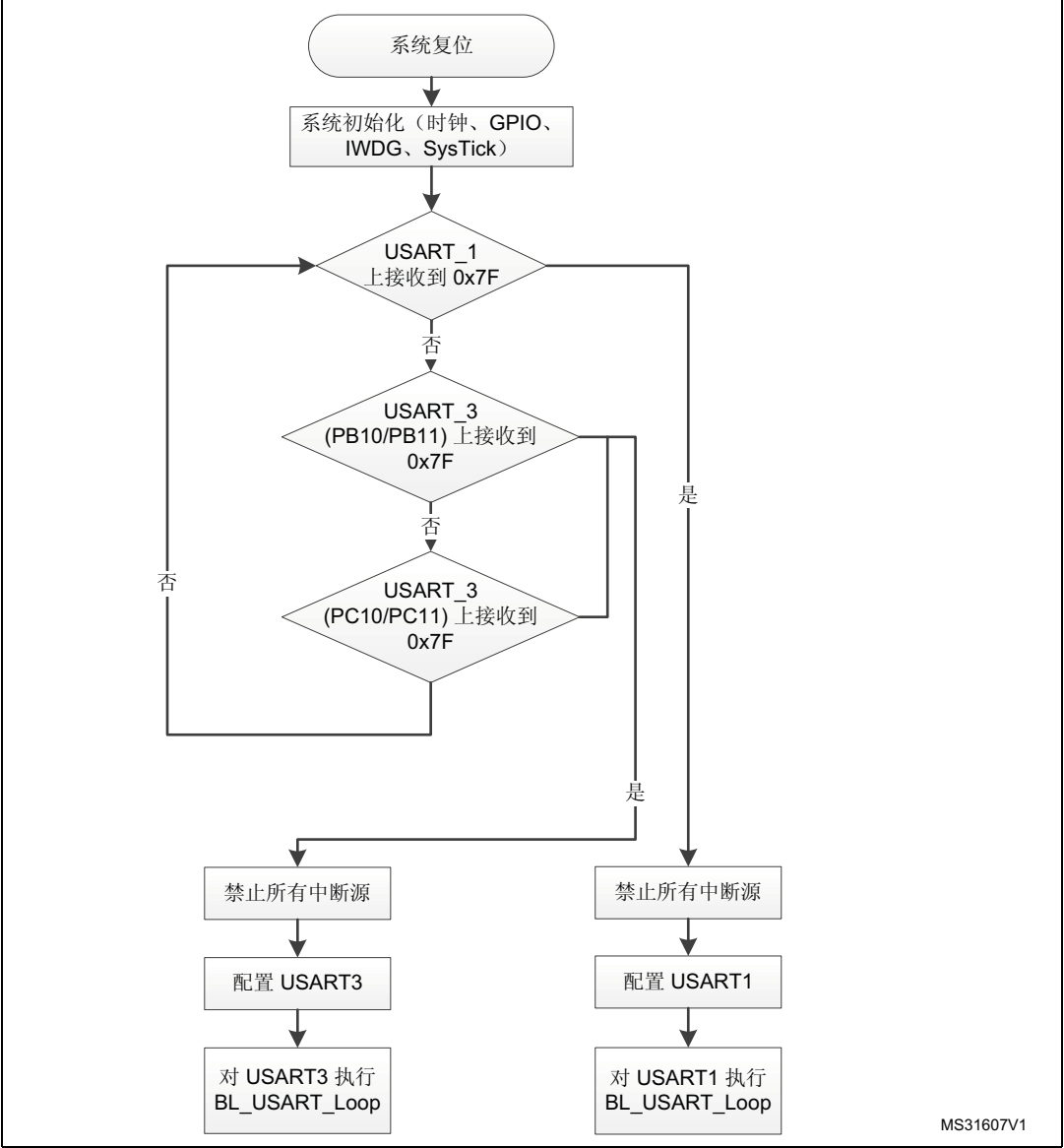
注：如下文所述，检测阶段期间，此自举程序未使用的外设 *RX* 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 9.1.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART3，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率检测序列，然后进入循环，等待发出任何 USART 自举程序命令。

为自举程序选择一个接口后，其它接口将禁止。

图6 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 6. STM32F2xxxx 器件的自举程序 V2.x 选择



9.1.4 重要事项

STM32F2xxxx 自举程序有一些特定的功能需要格外注意：

- **STM32F2xxxx** 自举程序固件除了支持标准存储器（内部 Flash、内部 SRAM、选项字节和系统存储器）外，还支持 **OTP 存储器**（从 0x1FFF 7800 到 0x1FFF 7A00 的 512 字节空间）。有关详细信息，请参见 PM0059 Flash 编程手册。
- **OTP 存储器** 允许执行读和写操作，但不能使用 **Erase** 命令进行擦除。向 OTP 存储单元执行写操作时，需确保相关保护位（OTP 存储器的最后 16 个字节）未复位。
- **选项字节**
地址为 0x1FFFC000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
 有关保护级别的详细信息，请参见 PM0059 编程手册。
- **读保护** 命令对应级别 1 保护。
- **禁止读保护** 命令对应级别 0 保护。
- **批量擦除** 命令在 STM32F2xxxx 中的执行时间比其它 STM32F 器件中长，这是由于它们的容量不同。请确保主机接口在发送批量擦除命令之后有足够长的超时时间来等待确认事件。
- **电压范围配置**
电压范围可通过自举程序软件实时更新。每次启动自举程序软件时（系统复位或跳转到自举程序代码后），电压范围都将设为默认值。自举程序软件可通过虚拟存储单元来修改此参数。该存储单元并非物理单元，但可根据所使用的协议（USART、CAN 或 DFU）通过常见的自举程序读/写操作进行读写。该存储单元包含 4 个字节，如表 17 所述。支持 1、2、3 或 4 字节访问。但保留字节应保持默认值 (0xFF)，否则请求将收到 NACK 应答。

表 17. 使用自举程序 V2.x 配置 STM32F2xxxx 电压范围

地址	大小	说明
0xFFFF0000	1 字节	该字节控制电压范围的当前值： 0x00: 电压范围 [1.8V, 2.1V] 0x01: 电压范围 [2.1V, 2.4V] 0x02: 电压范围 [2.4V, 2.7V] 0x03: 电压范围 [2.7V, 3.6V] 0x04: 电压范围 [2.7V, 3.6V] 并且使用双字节写/擦除操作。这种情况下，必须通过 VPP 引脚提供 9 V 电压（有关双字节写操作的详细信息，请参见 PM0059）。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0001	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0002	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0003	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。

9.1.5 自举程序 V2.x 版本

表 18 列出了 STM32F2xxxx 自举程序的 V2.x 版本。

表 18. STM32F2xxxx 自举程序 V2.x 版本

自举程序 版本号	说明	已知限制
V2.0	初始自举程序 V2.x 版本	当通过一个不受支持的存储器地址和一个正确的地址检验和（即，地址 0x6000 0000）发出 Read Memory 命令或 Write Memory 命令时，自举程序器件会中止这一命令，但不会向主机发送 NACK (0x1F)。因此，接下来的两个字节（即，待读 / 写的字节数及其校验和）会被看作一条新命令及其校验和。 ⁽¹⁾

1. 如果待读/写的“数据数 - 1” (N-1) 不等于有效命令代码 (0x00、0x01、0x02、0x11、0x21、0x31、0x43、0x44、0x63、0x73、0x82 或 0x92)，则无法从主机发现缺陷，因为该命令始终会收到 NACK 应答（当作不受支持的新命令）。

9.2 自举程序 V3.x

9.2.1 自举程序配置

STM32F2xxxx 器件的嵌入式自举程序 V3.x 支持四种串行外设：USART1、USART3、CAN2 和 DFU（USB FS 设备）。

表 19 介绍了系统存储器自举模式下，自举程序 V3.x 需要使用的 STM32F2xxxx 器件硬件资源。

表 19. 系统存储器自举模式下 STM32F2xxxx 的配置

自举程序	功能/外设	状态	备注
适用于所有 自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。 HSI 时钟源用于启动阶段（接口检测阶段）和选择 USARTx 接口期间（选择 CAN 或 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率等于 60 MHz。 仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE 时钟源。 外部时钟提供的频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	使能时钟安全系统 (CSS) 中断以用于 CAN 和 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	RAM	-	自地址 0x2000 0000 起的 8 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FF0 0000 起的 30688 字节空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为 [1.8V, 2.1V]。可使用自举程序命令在运行期间配置电压范围。请注意，此范围内仅支持按照字节格式（不支持半字、字和双字操作）对内部 Flash 执行写操作。

表 19. 系统存储器自举模式下 STM32F2xxxx 的配置（续）

自举程序	功能/外设	状态	备注
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	检测阶段期间，USART3_RX (PB11)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART3 自举程序 (PB10/PB11)	USART3	使能	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PB11 引脚：USART3 处于接收模式
	USART3_TX 引脚	输出	PB10 引脚：USART3 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART3 自举程序 (PC10/PC11)	USART3	使能	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PC11 引脚：USART3 处于接收模式
	USART3_TX 引脚	输出	PC10 引脚：USART3 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART1 和 USART3 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
CAN2 自举程序	CAN2	使能	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注： 由于 STM32F2xxxx CAN1 管理着 CAN2 和 SRAM 之间的通信，因此执行 CAN2 自举程序期间要为 CAN1 提供时钟。
	CAN2_RX 引脚	输入	PB05 引脚：CAN2 处于接收模式
	CAN2_TX 引脚	输出	PB13 引脚：CAN2 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
DFU 自举程序	USB_OTG_FS	使能	USB OTG FS 配置为强制器件模式。使能 USB_OTG_FS 中断向量以用于 USB DFU 通信。
	USB_OTG_FS_DM 引脚	输入	PA11 引脚：USB OTG FS DM 线
	USB_OTG_FS_DP 引脚	输出	PA12 引脚：USB OTG FS DP 线
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11) 和 CAN2_RX (PB05) 引脚必须保持高电平或低电平。		
CAN2 和 DFU 自举程序	TIM11	使能	此定时器用于确定外部时钟的频率值。 确定外部时钟频率之后，RCC 系统工作在 60 MHz 系统时钟下（使用 PLL）。

注: 对于 DFU 接口, 需要提供外部时钟源 (HSE) 才能执行 USB 操作。HSE 值的检测由自举程序固件完成, 该过程以内部振荡器时钟 (HSI) 为依据。因此, 当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时 (理论值约为 1%), 自举程序计算所得的 HSE 频率值可能不正确。这种情况下, 自举程序 DFU 接口会出现故障, 甚至完全无法工作。

对于 USARTx 自举程序, 系统时钟由嵌入式内部高速 RC 提供。这种情况下, 执行自举程序代码时无需外部石英时钟。该内部时钟同样用于 CAN 和 DFU (USB FS 设备), 但仅限选择阶段。选择阶段结束后, 执行 CAN 和 DFU 自举程序需要使用频率为 1 MHz 的倍数 (介于 4 MHz 到 26 MHz 之间) 的外部时钟。

CAN 和 DFU 自举程序采用外部时钟检测机制, 能够通过内部高速 RC 和 TIM11 定时器确定外部时钟的频率值。该机制的精度仅支持检测等于 1 MHz 倍数的频率, 且范围介于 4 MHz 到 26 MHz 之间。任何其它值都不适用, 否则将导致自举程序发生意外行为。

下载应用程序二进制代码后, 如果选择执行 Go 命令, 则在跳转到用户应用程序之前, 自举程序使用的外设寄存器 (如上表所示) 不会初始化为默认复位值。如果要使用这些寄存器, 应在用户应用程序中对其重新配置。因此, 如果应用程序正在使用 IWDG, 则必须调整 IWDG 预分频器值来满足应用程序的要求 (因为自举程序已将预分频器设置为最大值)。

9.2.2 自举程序硬件要求

将 STM32F2xxxx 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线, 它们能够在复位期间使 BOOT0 引脚保持高电平, 使 BOOT1 引脚保持低电平。

要在系统存储器自举模式下连接 STM32F2xxxx, 必须确认以下条件:

- 检测阶段期间, 此自举程序未使用的外设 RX 引脚必须保持已知电平 (低电平或高电平) 且不得悬空, 如下所述:
 - 如果使用 USART1 连接自举程序, 则检测阶段期间, USART3_RX (PC11 和 PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 USART3 (PB10/PB11) 连接自举程序, 则检测阶段期间, USART1_RX (PA10)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 USART3 (PC10/PC11) 连接自举程序, 则检测阶段期间, USART1_RX (PA10)、USART3_RX 引脚 (PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 CAN2 连接自举程序, 则检测阶段期间, USART1_RX (PA10)、USART3_RX (PC11 和 PB11)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平, 并且不得悬空。
 - 如果使用 DFU (USB FS 设备) 连接自举程序, 则检测阶段期间, USART1_RX (PA10)、USART3_RX (PC11 和 PB11) 和 CAN2_RX (PB05) 引脚必须保持高电平或低电平, 并且不得悬空。

- 通过以下硬件连接外设：
 - RS232 串行接口（例如 ST3232 RS232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
 - CAN 接口（CAN 收发器），必须将该接口直接连接到 CAN2_RX (PB5) 和 CAN2_TX (PB13) 引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART3。

用户可借助 PC 串行小应用程序，使用 RS232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM322xG-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见 STM322xG-EVAL 板用户手册，可从意法半导体网站获取该手册：<http://www.st.com>。

9.2.3 自举程序选择

STM32F2xxxx 嵌入式自举程序 V3.x 支持三种外设接口：USART1、USART3 (PB10/PB11 和 PC10/PC11)、CAN2 和 DFU (USB FS 设备)。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 9.2.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART3，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

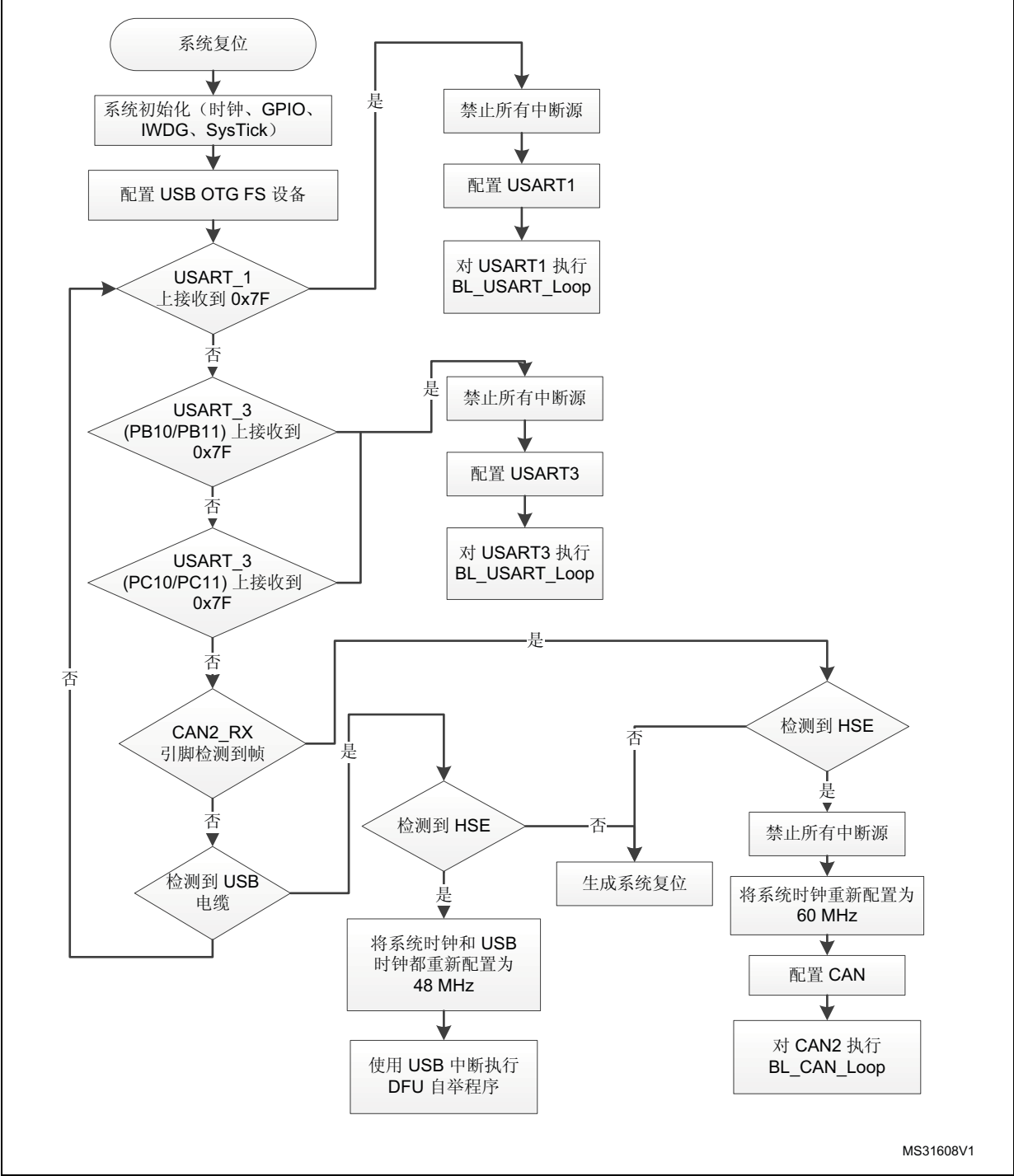
要使用 CAN2 接口，应将 CAN 电缆连接到 CAN2。自举程序在 CAN2_RX 引脚 (PB5) 上检测到数据帧之后，自举程序固件将进入 CAN 循环，开始确定外部时钟的频率值。支持等于 1 MHz 倍数的 HSE 频率，且其范围介于 4 MHz 到 26 MHz 之间。任何其它值都会导致意外行为，CAN 自举程序固件将进入无限循环并等待，直到接收到消息。如果没有外部时钟，则产生系统复位。

在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 或 CAN 自举程序，必须确保选择阶段没有任何 USB 主机连接到 USB 外设。选择 USART 或 CAN 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，所有其它接口都禁止。下图显示了自举程序选择机制。详细信息请参见各外设自举程序的对应部分。

图 7. STM32F2xxxx 器件的自举程序 V3.x 选择



9.2.4 重要事项

STM32F2xxxx 自举程序有一些特定的功能需要格外注意：

- **STM32F2xxxx 器件自举程序固件**除了支持标准存储器（内部 Flash、内部 SRAM、选项字节和系统存储器）外，还支持 **OTP 存储器**（从 0x1FFF 7800 到 0x1FFF 7A00 的 512 字节空间，详细信息请参见 PM0059 编程手册）。
- **OTP 存储器**允许执行读和写操作，但不能使用 **Erase** 命令进行擦除。向 **OTP 存储单元**执行写操作时，需确保相关保护位（OTP 存储器的最后 16 个字节）未复位。
- **选项字节**
地址为 0x1FFFC000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
 有关保护级别的详细信息，请参见 PM0059 编程手册。
- **读保护**命令对应级别 1 保护。
- **禁止读保护**命令对应级别 0 保护。
- **批量擦除**命令在 STM32F2xxxx 中的执行时间比其它 STM32 器件中长，这是由于它们的容量不同。请确保主机接口在发送批量擦除命令之后有足够长的超时时间来等待确认事件。
- **电压范围配置**
电压范围可通过自举程序软件实时更新。每次启动自举程序软件时（系统复位或跳转到自举程序代码后），电压范围都将设为默认值。自举程序软件可通过虚拟存储单元来修改此参数。该存储单元并非物理单元，但可根据所使用的协议（USART、CAN 或 DFU）通过常见的自举程序读/写操作进行读写。该存储单元包含 4 个字节，如表 20 所述。支持 1、2、3 或 4 字节访问。但保留字节应保持默认值 (0xFF)，否则请求将收到 NACK 应答。

表 20. 使用自举程序 V3.x 配置 STM32F2xxxx 电压范围

地址	大小	说明
0xFFFF0000	1 字节	该字节控制电压范围的当前值： 0x00: 电压范围 [1.8V, 2.1V] 0x01: 电压范围 [2.1V, 2.4V] 0x02: 电压范围 [2.4V, 2.7V] 0x03: 电压范围 [2.7V, 3.6V] 0x04: 电压范围 [2.7V, 3.6V] 并且使用双字写/擦除操作。这种情况下，必须通过 VPP 引脚提供 9 V 电压（有关双字写操作步骤的详细信息，请参见 PM0059）。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0001	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0002	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0003	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。

9.2.5 自举程序版本 V3.x

下表列出了 STM32F2xxxx 自举程序的 V3.x 版本。

表 21. STM32F2xxxx 自举程序 V3.x 版本

自举程序版本号	说明	已知限制
V3.2	初始自举程序版本 V3.x。	<div><div>– 当通过一个不受支持的存储器地址和一个正确的地址检验和（即，地址 0x6000 0000）发出 Read Memory 命令或 Write Memory 命令时，自举程序器件会中止这一命令，但不会向主机发送 NACK (0x1F)。因此，接下来的两个字节（即，待读/写的字节数及其校验和）会被看作一条新命令及其校验和⁽¹⁾。</div><div>– 选项字节、OTP 和设备功能描述符（位于 DFU 接口中）设置为“g”而不是“e”（不是可擦除存储区）。</div></div>
V3.3	修正 V3.2 限制。提高 DFU 接口耐用性。	<div><div>– 对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号（而不是 1 个 NACK 信号）。</div></div>

1. 如果待读 / 写的“数据数 - 1” (N-1) 不等于有效命令代码（0x00、0x01、0x02、0x11、0x21、0x31、0x43、0x44、0x63、0x73、0x82 或 0x92），则无法从主机发现缺陷，因为该命令始终会收到 NACK 应答（当作不受支持的新命令）。

10 STM32F405/415xx 和 STM32F407/417xx 自举程序

本部分使用 STM32F40xxx/41xxx 指代 STM32F405xx、STM32F407xx、STM32F415xx 和 STM32F417xx 器件。

10.1 自举程序配置

STM32F40xxx/41xxx 器件的嵌入式自举程序支持四种串行外设：USART1、USART3、CAN2 和 DFU（USB FS 设备）。

表 22 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F40xxx/41xxx 器件硬件资源。

表 22. 系统存储器自举模式下 STM32F40xxx/41xxx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。 HSI 时钟源用于启动阶段（接口检测阶段）和选择 USARTx 接口期间（选择 CAN 或 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率等于 60 MHz。 仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE 时钟源。 外部时钟提供的频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	使能时钟安全系统 (CSS) 中断以用于 CAN 和 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	RAM	-	自地址 0x2000 0000 起的 8 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFF 0000 起的 30688 字节空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为 [1.8V, 2.1V]。可使用自举程序命令在运行期间配置电压范围。请注意，此范围内仅支持按照字节格式（不支持半字、字和双字操作）对内部 Flash 执行写操作。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	检测阶段期间，USART3_RX (PB11)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		

表 22. 系统存储器自举模式下 STM32F40xxx/41xxx 的配置 (续)

自举程序	功能/外设	状态	备注
USART3 自举程序 (PB10/PB11)	USART3	使能	初始化后, USART3 的配置为: 8 位, 偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PB11 引脚: USART3 处于接收模式。
	USART3_TX 引脚	输出	PB10 引脚: USART3 处于发送模式。
	检测阶段期间, USART1_RX (PA10)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART3 自举程序 (PC10/PC11)	USART3	使能	初始化后, USART3 的配置为: 8 位, 偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PC11 引脚: USART3 处于接收模式。
	USART3_TX 引脚	输出	PC10 引脚: USART3 处于发送模式。
	检测阶段期间, USART1_RX (PA10)、USART3_RX (PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART1 和 USART3 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
CAN2 自举程序	CAN2	使能	初始化后, CAN2 的配置为: 波特率 125 kbps, 11 位标识符。 注: 由于 STM32F40xxx/41xxx CAN1 管理着 CAN2 和 SRAM 之间的通信, 因此执行 CAN2 自举程序期间要为 CAN1 提供时钟。
	CAN2_RX 引脚	输入	PB05 引脚: CAN2 处于接收模式。
	CAN2_TX 引脚	输出	PB13 引脚: CAN2 处于发送模式。
	检测阶段期间, USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
DFU 自举程序	USB_OTG_FS	使能	USB OTG FS 配置为强制器件模式。使能 USB_OTG_FS 中断向量以用于 USB DFU 通信。
	USB_OTG_FS_DM 引脚	输入	PA11 引脚: USB OTG FS DM 线。
	USB_OTG_FS_DP 引脚	输出	PA12 引脚: USB OTG FS DP 线。
	检测阶段期间, USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11) 和 CAN2_RX (PB05) 引脚必须保持高电平或低电平。		
CAN2 和 DFU 自举程序	TIM11	使能	此定时器用于确定外部时钟的频率值。 确定外部时钟频率之后, RCC 系统工作在 60 MHz 系统时钟下 (使用 PLL)。

注: 对于 DFU 接口, 需要提供外部时钟源 (HSE) 才能执行 USB 操作。HSE 值的检测由自举程序固件完成, 该过程以内部振荡器时钟 (HSI) 为依据。因此, 当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时 (理论值约为 1%), 自举程序计算所得的 HSE 频率值可能不正确。这种情况下, 自举程序 DFU 接口会出现故障, 甚至完全无法工作。

对于 USARTx 自举程序，系统时钟由嵌入式内部高速 RC 提供。这种情况下，执行自举程序代码时无需外部石英时钟。该内部时钟同样用于 CAN 和 DFU（USB FS 设备），但仅限选择阶段。选择阶段结束后，执行 CAN 和 DFU 自举程序需要使用频率为 1 MHz 的倍数（介于 4 MHz 到 26 MHz 之间）的外部时钟。

CAN 和 DFU 自举程序采用外部时钟检测机制，能够通过内部高速 RC 和 TIM11 定时器确定外部时钟的频率值。该机制的精度仅支持检测等于 1 MHz 倍数的频率，且范围介于 4 MHz 到 26 MHz 之间。任何其它值都不适用，否则将导致自举程序发生意外行为。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的外设寄存器（如上表所示）不会初始化为默认复位值。如果要使用这些寄存器，应在用户应用程序中对其重新配置。因此，如果应用程序正在使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

10.2 自举程序硬件要求

将 STM32F40xxx/41xxx 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

要在系统存储器自举模式下连接 STM32F40xxx/41xxx，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART3_RX (PC11 和 PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART3 (PB10/PB11) 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART3 (PC10/PC11) 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USART3_RX 引脚 (PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 必须保持高电平或低电平，并且不得悬空。
 - 如果使用 CAN2 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USART3_RX (PC11 和 PB11)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 DFU（USB FS 设备）连接自举程序，则检测阶段期间，USART1_RX (PA10)、USART3_RX (PC11 和 PB11) 和 CAN2_RX (PB05) 引脚必须保持高电平或低电平，并且不得悬空。
- 通过以下硬件连接外设：
 - RS232 串行接口（例如 ST3232 RS232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
 - CAN 接口（CAN 收发器），必须将该接口直接连接到 CAN2_RX (PB5) 和 CAN2_TX (PB13) 引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART3。

用户可借助 PC 串行小应用程序，使用 RS232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM324xG-EVAL 板提供了实现此控制所需的硬件。有关详细信息，请参见 STM324xG-EVAL 板用户手册，可从意法半导体网站获取该手册：<http://www.st.com>。

10.3 自举程序选择

STM32F40xxx/41xxx 嵌入式自举程序支持三种外设接口：USART1、USART3（PB10/PB11 和 PC10/PC11）、CAN2 和 DFU（USB FS 设备）。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 10.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART3，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

要使用 CAN2 接口，应将 CAN 电缆连接到 CAN2。自举程序在 CAN2_RX 引脚 (PB5) 上检测到数据帧之后，自举程序固件将进入 CAN 循环，开始确定外部时钟的频率值。

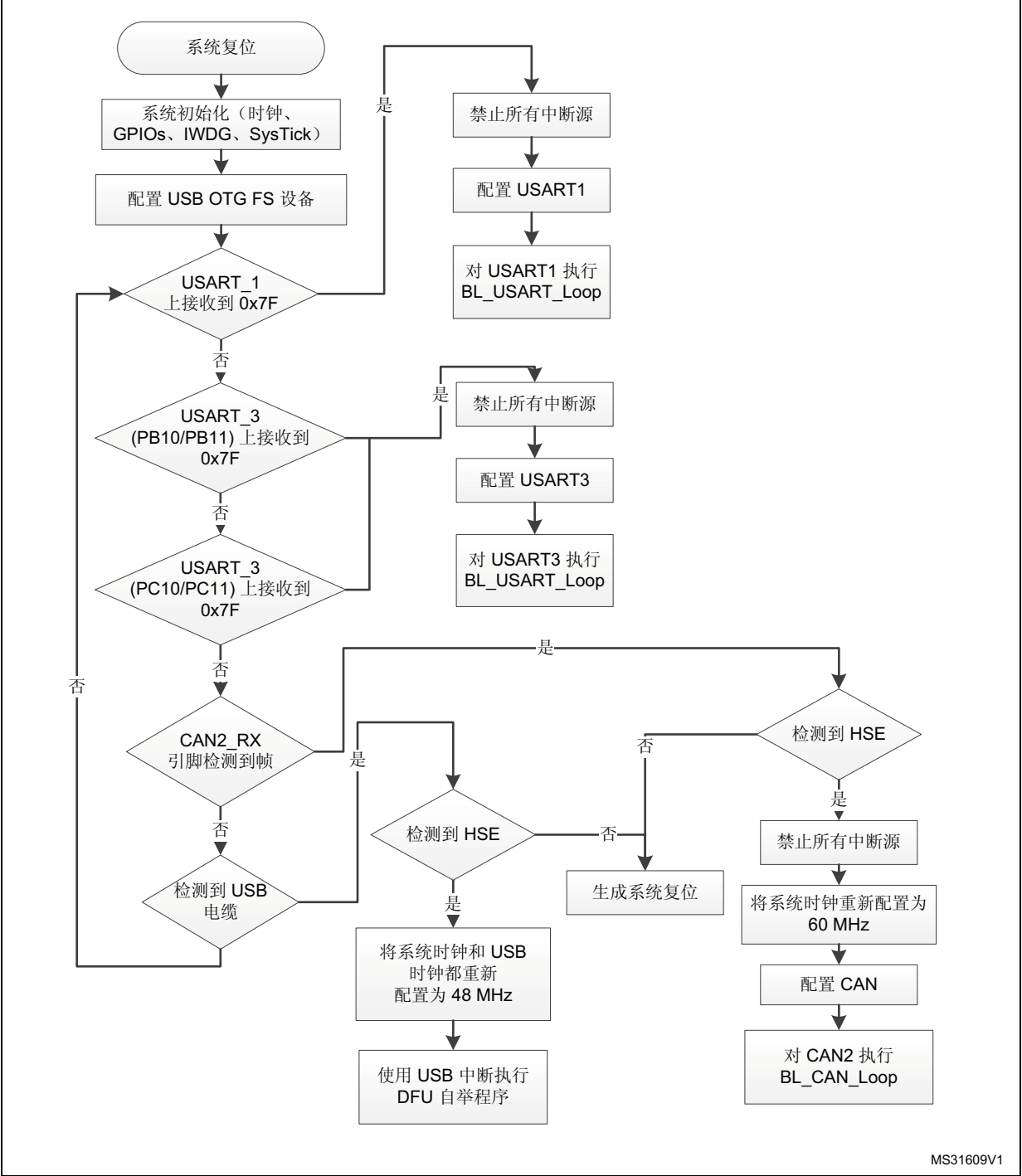
支持等于 1 MHz 倍数的 HSE 频率，且其范围介于 4 MHz 到 26 MHz 之间。任何其它值都会导致意外行为，CAN 自举程序固件将进入无限循环并等待，直到接收到消息。如果没有外部时钟，则产生系统复位。

在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 或 CAN 自举程序，必须确保选择阶段没有任何 USB 主机连接到 USB 外设。选择 USART 或 CAN 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，所有其它接口都禁止。下图显示了自举程序选择机制。详细信息请参见各外设自举程序的对应部分。

图 8. STM32F40xxx/41xxx 器件的自举程序选择



10.4 重要事项

STM32F40xxx/41xxx 自举程序有一些特定的功能需要格外注意：

- **STM32F40xxx/41xxx 器件自举程序固件**除了支持标准存储器（内部 Flash、内部 SRAM、选项字节和系统存储器）外，还支持 **OTP 存储器**（从 0x1FFF 7800 到 0x1FFF 7A00 的 512 字节空间，详细信息请参见 RM0090 参考手册）。
- **OTP 存储器**允许执行读和写操作，但不能使用 **Erase** 命令进行擦除。向 OTP 存储单元执行写操作时，需确保相关保护位（OTP 存储器的最后 16 个字节）未复位。
- **选项字节**
地址为 0x1FFFC000。支持三个保护级别：
 - － 级别 0
 - － 级别 1
 - － 级别 2有关保护级别的详细信息，请参见 PM0081 编程手册。
- **读保护命令**对应级别 1 保护。
- **禁止读保护命令**对应级别 0 保护。
- **批量擦除命令**在 STM32F40xxx/41xxx 中的执行时间比其它 STM32 器件中长，这是由于它们的容量不同。请确保主机接口在发送批量擦除命令之后有足够长的超时时间来等待确认事件。
- **电压范围配置**
电压范围可通过自举程序软件实时更新。每次启动自举程序软件时（系统复位或跳转到自举程序代码后），电压范围都将设为默认值。自举程序软件可通过虚拟存储单元来修改此参数。该存储单元并非物理单元，但可根据所使用的协议（USART、CAN 或 DFU）通过常见的自举程序读/写操作进行读写。该存储单元包含 4 个字节，如表 20 所述。支持 1、2、3 或 4 字节访问。但保留字节应保持默认值 (0xFF)，否则请求将收到 NACK 应答。

表 23. 使用自举程序配置 STM32F40xxx/41xxx 电压范围

地址	大小	说明
0xFFFF0000	1 字节	该字节控制电压范围的当前值： 0x00: 电压范围 [1.8V, 2.1V] 0x01: 电压范围 [2.1V, 2.4V] 0x02: 电压范围 [2.4V, 2.7V] 0x03: 电压范围 [2.7V, 3.6V] 0x04: 电压范围 [2.7V, 3.6V] 并且使用双字写/擦除操作。这种情况下，必须通过 VPP 引脚提供 9 V 电压（有关双字写操作步骤的详细信息，请参见 PM0081）。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0001	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0002	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0003	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。

10.5 自举程序版本

下表介绍了 STM32F40xxx/41xxx 自举程序版本。

表 24. STM32F40xxx/41xxx 自举程序版本

自举程序版本号	说明	已知限制
V3.0	初始自举程序版本。	<ul style="list-style-type: none"> 当通过一个不受支持的存储器地址和一个正确的地址检验和（即，地址 0x6000 0000）发出 Read Memory 命令或 Write Memory 命令时，自举程序器件会中止这一命令，但不会向主机发送 NACK (0x1F)。因此，接下来的两个字节（即，待读/写的字节数及其校验和）会被看作一条新命令及其校验和⁽¹⁾。 选项字节、OTP 和设备功能描述符（位于 DFU 接口中）设置为“g”而不是“e”（不是可擦除存储区）。
V3.1	修正 V3.0 限制。提高 DFU 接口耐用性。	<ul style="list-style-type: none"> 对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号（而不是 1 个 NACK 信号）。

1. 如果待读/写的“数据数 - 1” (N-1) 不等于有效命令代码 (0x00、0x01、0x02、0x11、0x21、0x31、0x43、0x44、0x63、0x73、0x82 或 0x92)，则无法从主机发现缺陷，因为该命令始终会收到 **NACK** 应答（当作不受支持的新命令）。

11

STM32F051x4、STM32F051x6 和 STM32F051x8 器件自举程序

本部分使用 STM32F051xx 指代 STM32F051x4、STM32F051x6 和 STM32F051x8 器件。

11.1

自举程序配置

STM32F051xx 器件嵌入式自举程序支持两种串行接口：USART1 和 USART2 外设。

下表介绍了系统存储器自举模式下，自举程序需要使用的 STM32F051xx 器件硬件资源。

表 25. 系统存储器自举模式下 STM32F051xx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	时钟源	HSI 使能	系统时钟频率等于 24 MHz（使用由 HSI 驱动的 PLL）。1 个 Flash 等待周期。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFFE000 起的 3 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART2_RX (PA14) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。
	USART2_RX 引脚	输入	PA14 引脚：USART2 处于接收模式。
	USART2_TX 引脚	输出	PA15 引脚：USART2 处于发送模式。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测主机串口波特率。

注： STM32F051x6 和 STM32F051x8 器件在自举程序模式下自举之后，系统发生复位之前，无法再实现串行线调试 (SWD) 通信，因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

11.2 自举程序硬件要求

将 STM32F051xx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLINK 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F051xx 器件，必须确认以下条件：

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PA15) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 要使用的外设必须通过 RS232 串行接口（例如 ST3232 RS232 收发器）连接，并且：
 - 使用 USART1 时，该串行接口必须直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚
 - 使用 USART2 时，该串行接口必须直接连接到 USART2_RX (PA15) 和 USART2_TX (PA14) 引脚。
- USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART2。

用户可借助 PC 串行小应用程序，使用 RS232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM320518-EVAL 板提供了实现此控制所需的硬件。有关详细信息，请参见“STM320518-EVAL 板用户手册” (UM1537)，可从意法半导体网站获取该手册：<http://www.st.com>。

11.3 自举程序选择

STM32F051xx 器件嵌入式自举程序支持两种外设接口：USART1 和 USART2。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

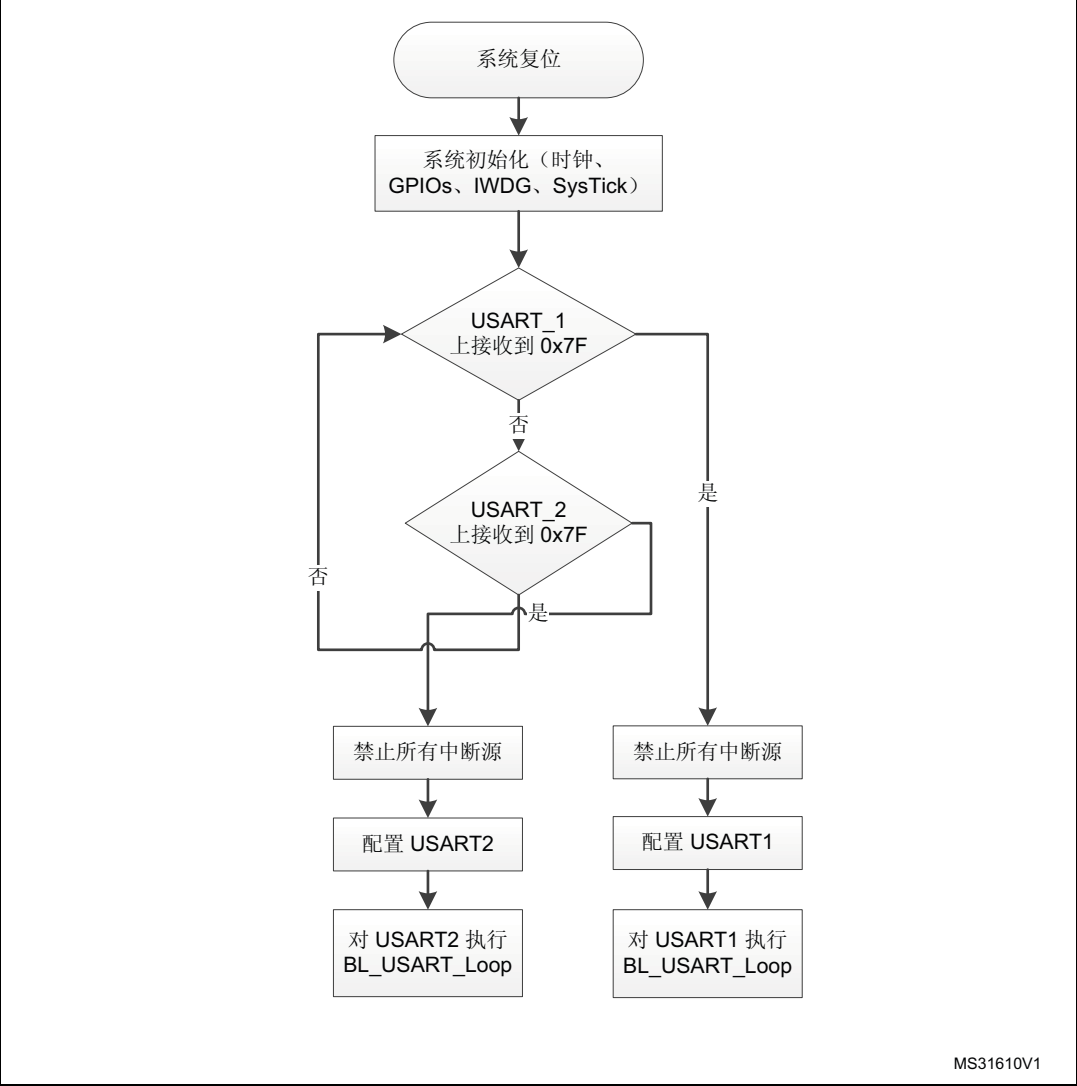
*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 11.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

为自举程序选择一个接口后，其它接口将禁止。

下图显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 9. STM32F051xx 器件的自举程序选择



11.4 重要事项

STM32F051xx 器件的自举程序有一些特定的功能需要格外注意，如下所述：

- 选项字节
 - 地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
 有关保护级别的详细信息，请参见 RM0091 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x0800 0000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x0000 0000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0091 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须将向量表映射到 SRAM，而且此时应通过软件将 SRAM 映射到地址 0x0000 0000。

11.5 自举程序版本

下表列出了 STM32F051xx 自举程序版本。

表 26. STM32F051xx 自举程序版本

自举程序版本号	说明	已知限制
V2.1	初始自举程序版本	当用户应用程序在配置 HSI TRIM 位的值（RCC_CR 寄存器中）后跳转到自举程序时，HSITRIM 值将在启动自举程序时复位为默认值 0。 对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号，而不是 1 个 NACK 信号。

12 STM32L151xx 和 STM32L152xx 中容量增强型超低功耗器件自举程序

本部分使用 STM32L15xxx 中容量增强型器件指代 STM32L151xx 和 STM32L152xx 中容量增强型超低功耗器件。

12.1 自举程序配置

STM32L15xxx 中容量增强型器件的嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。

表 27 介绍了系统存储器自举模式下，自举程序需要使用的 STM32L15xxx 中容量增强型器件硬件资源。

表 27. 系统存储器自举模式下 STM32L15xxx 中容量增强型器件的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 HSI 作为系统时钟，频率为 16 MHz。仅用于 USART1 和 USART2 自举程序以及在 USB 检测期间用于 DFU 自举程序（选择 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	只有 DFU 自举程序必须使用外部时钟，外部时钟必须介于以下频率范围： [24, 16, 12, 8, 6, 4, 3, 2] MHz。 PLL 用于生成 48 MHz USB 时钟和 32 MHz 系统时钟。
		-	使能时钟安全系统 (CSS) 中断以用于 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源		电压范围设置为电压范围 1。
	系统存储器	-	自地址 0x1FF0 0000 起的 8 KB 空间。此区域包含自举程序固件。
USART1 自举程序	RAM	-	自地址 0x2000 0000 起的 4 KB 空间供自举程序固件使用。
	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		



表 27. 系统存储器自举模式下 STM32L15xxx 中容量增强型器件的配置（续）

自举程序	功能/外设	状态	备注
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式。
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式。
	检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		
DFU 自举程序	USB_DM 引脚	输入或复用功能，通过 USB 自动控制	PA11：USB 发送接收数据线。
	USB_DP 引脚		PA12：USB 发送接收数据线。
	中断	使能	使能 USB 低优先级中断向量以用于 USB DFU 通信。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

注：对于 DFU 接口，需要提供外部时钟源 (HSE) 才能执行 USB 操作。HSE 值的检测通过自举程序固件完成，该过程以内部振荡器时钟 (HSI、MSI) 为依据。因此，当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时（理论值约为 1%），自举程序计算所得的 HSE 频率值可能不正确。这种情况下，自举程序 DFU 接口会出现故障，甚至完全无法工作。

对于 USARTx 自举程序，系统时钟由嵌入式内部高速 RC 提供。该内部时钟同样用于 DFU 自举程序，但仅限选择阶段。选择阶段结束后，执行 DFU 自举程序时需要频率介于范围 [24, 16, 12, 8, 6, 4, 3, 2] MHz 的外部时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

12.2 自举程序硬件要求

将 STM32L15xxx 中容量增强型器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 **BOOT0** 引脚保持高电平，使 **BOOT1** 引脚保持低电平。

要在系统存储器自举模式下连接 STM32L15xxx 中容量增强型器件，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 **USART1** 连接自举程序，则检测阶段期间，**USART2_RX** (PD6)、**USB_DM** (PA11) 和 **USB_DP** (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 **USART2** 连接自举程序，则检测阶段期间，**USART1_RX** (PA10)、**USB_DM** (PA11) 和 **USB_DP** (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 **DFU (USB)** 连接自举程序，则检测阶段期间，**USART1_RX** (PA10) 和 **USART2_RX** (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
- 连接外设。
 - **RS-232** 串行接口（例如 **ST3232 RS-232** 收发器），使用 **USART1** 时必须将该接口直接连接到 **USART1_RX** (PA10) 和 **USART1_TX** (PA9) 引脚；使用 **USART2** 时则必须直接连接到 **USART2_RX** (PD6) 和 **USART2_TX** (PD5) 引脚。
 - 经认证的 **USB** 电缆，必须使用此类电缆连接到微控制器（也可使用 **ESD** 保护电路）。

USART1_CK、**USART1_CTS** 和 **USART1_RTS** 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 **GPIO**。这同样适用于 **USART2**。

12.3 自举程序选择

STM32L15xxx 中容量增强型器件的嵌入式自举程序支持三种串行接口：**USART1**、**USART2** 和 **DFU (USB)**。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 **Flash**。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

注： 如下文所述，检测阶段期间，此自举程序未使用的外设 **RX** 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 12.2 节：自举程序硬件要求。

要将 **USART** 自举程序用于 **USART1** 或 **USART2**，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 **0x7F** 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 **USART** 自举程序命令。

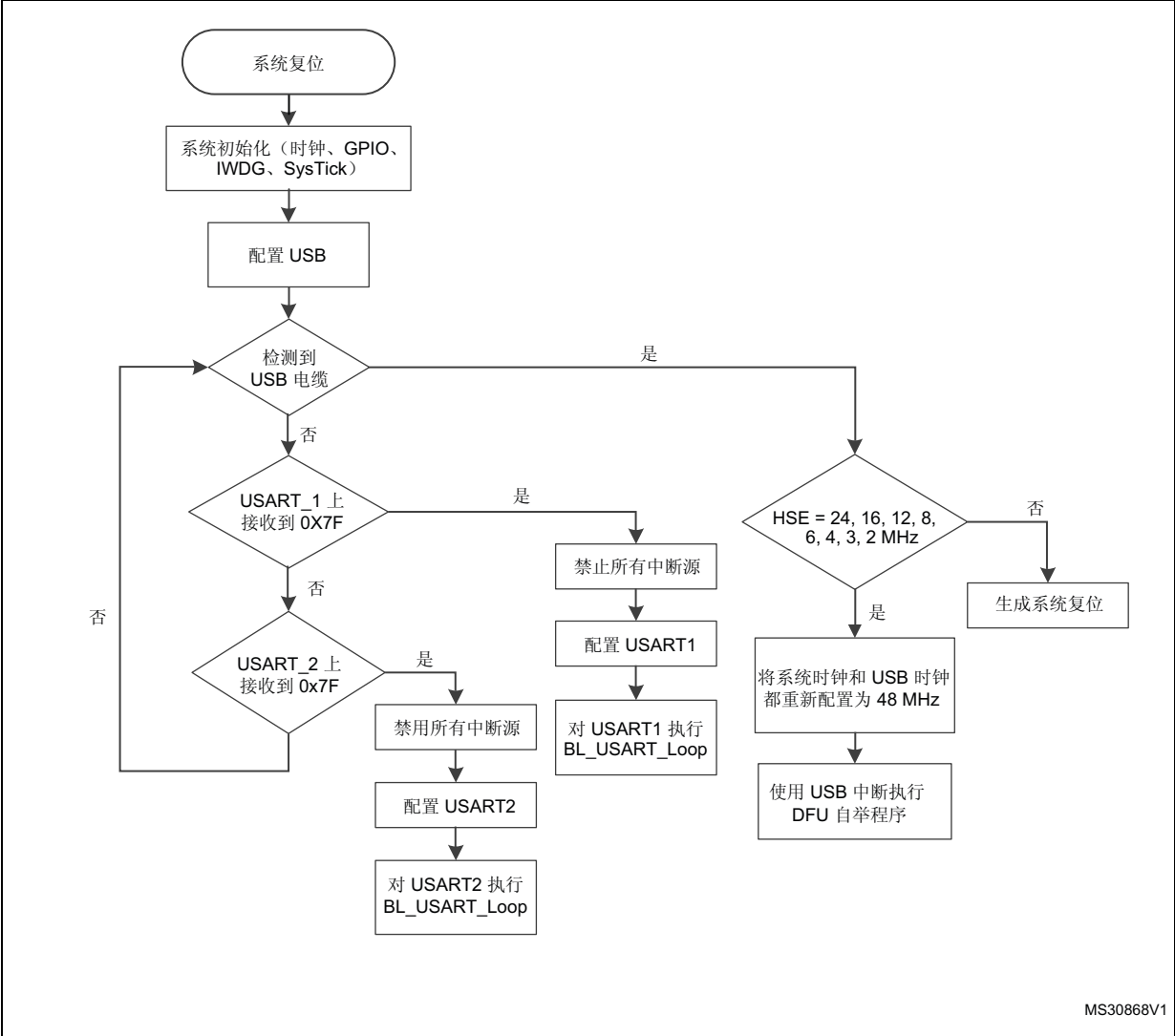
在自举程序固件选择序列期间，无论何时将 **USB** 电缆插入微控制器的 **USB** 接口，自举程序都会进入 **DFU** 自举程序循环，等待发出任何 **DFU** 自举程序命令。

要使用 **USART** 自举程序，必须确保选择阶段没有任何 **USB** 主机连接到 **USB** 外设。选择 **USART** 自举程序后，用户可插入一条 **USB** 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，其它接口将禁止。

图 10 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 10. STM32L15xxx 中容量增强型器件的自举程序选择



12.4 重要事项

STM32L15xxx 中容量增强型器件的自举程序有一些特定的功能需要格外注意，如下所述。

- STM32L15xxx 中容量增强型器件自举程序固件除了支持标准存储器（内部 Flash、内部 SRAM、选项字节和系统存储器）外，还支持数据存储器（从 0x08080000 到 0x08081FFF 的 8 KB 空间）。有关详细信息，请参见 PM0062 编程手册。
- Flash 写操作通过程序存储器的半页写操作执行。自举程序固件按非对齐地址管理半页写操作。因此，所有写操作都只能采用字对齐形式（地址应为 4 的倍数）。要写入的数据量也必须为 4 的倍数（允许非对齐半页写操作地址）。请参见产品数据手册，了解写操作所需的持续时间。
- 数据存储器允许执行读和写操作，但不能使用 Erase 命令进行擦除。向数据存储单元执行写操作时，自举程序固件会在任何写操作之前对此单元执行擦除操作。数据存储器的写操作必须采用字对齐形式（要写入的地址应为 4 的倍数），并且要写入的数据量也必须为 4 的倍数。要擦除数据存储单元，可在此单元中写入零。
- 选项字节
地址为 0x1FF80000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2有关保护级别的详细信息，请参见 PM0062 编程手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 批量擦除命令尚未获得 STM32L15xxx 中容量增强型器件自举程序固件的支持。要执行批量擦除操作，可采用两种方法。
 - 使用 Erase 命令逐个擦除所有扇区。
 - 将保护级别设为级别 1。然后再设为级别 0（使用 Read protect 命令，然后使用 Read Unprotect 命令）。执行此操作后将批量擦除内部 Flash 内容（有关详细信息，请参见 PM0062 编程手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

12.5 自举程序版本

表 28 列出了 STM32L15xxx 中容量增强型器件的自举程序版本。

表 28. STM32L15xxx 中容量增强型器件自举程序版本

自举程序版本号	说明	已知限制
V4.0	初始自举程序版本	对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号，而不是 1 个 NACK 信号。



13 STM32F050x4 和 STM32F050x6 器件自举程序

本部分使用 STM32F050xx 指代 STM32F050x4 和 STM32F050x6 器件。

13.1 自举程序配置

STM32F050xx 器件内嵌式自举程序支持一种串行接口：USART1 外设。

表 29 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F050xx 器件硬件资源。

表 29. 系统存储器自举模式下 STM32F050xx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	时钟源	HSI 使能	系统时钟频率等于 24 MHz（使用由 HSI 驱动的 PLL）。1 个 Flash 等待周期。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFFE000 起的 3 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序 (PA10/PA9)	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART1_RX (PA14) 引脚必须保持高电平或低电平。		
USART1 自举程序 (PA14/PA15)	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA14 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA15 引脚：USART1 处于发送模式。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 自举程序	SysTick 定时器	使能	用于自动检测主机串口波特率。

注：STM32F050x4 和 STM32F050x6 器件在自举程序模式下自举之后，系统发生复位之前，无法再实现串行线调试 (SWD) 通信。这是因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

13.2 自举程序硬件要求

将 STM32F050xx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLINK 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F050xx 器件，必须确认以下条件。

- 检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空，如下所述：
 - 如果使用 USART1 (PA10/PA9) 连接自举程序，则检测阶段期间，USART1_RX (PA15) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART1 (PA14/PA15) 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 要使用的外设必须通过 RS-232 串行接口（例如 ST3232 RS-232 收发器）连接，并且：
 - 使用 USART1 (PA10/PA9) 时，该串行接口必须直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚
 - 使用 USART1 (PA14/PA15) 时，该串行接口必须直接连接到 USART1_RX (PA15) 和 USART1_TX (PA14) 引脚。
- USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 GPIO。

13.3 自举程序选择

STM32F050xx 器件内嵌式自举程序支持 USART1 这一种外设接口。该外设接口映射可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

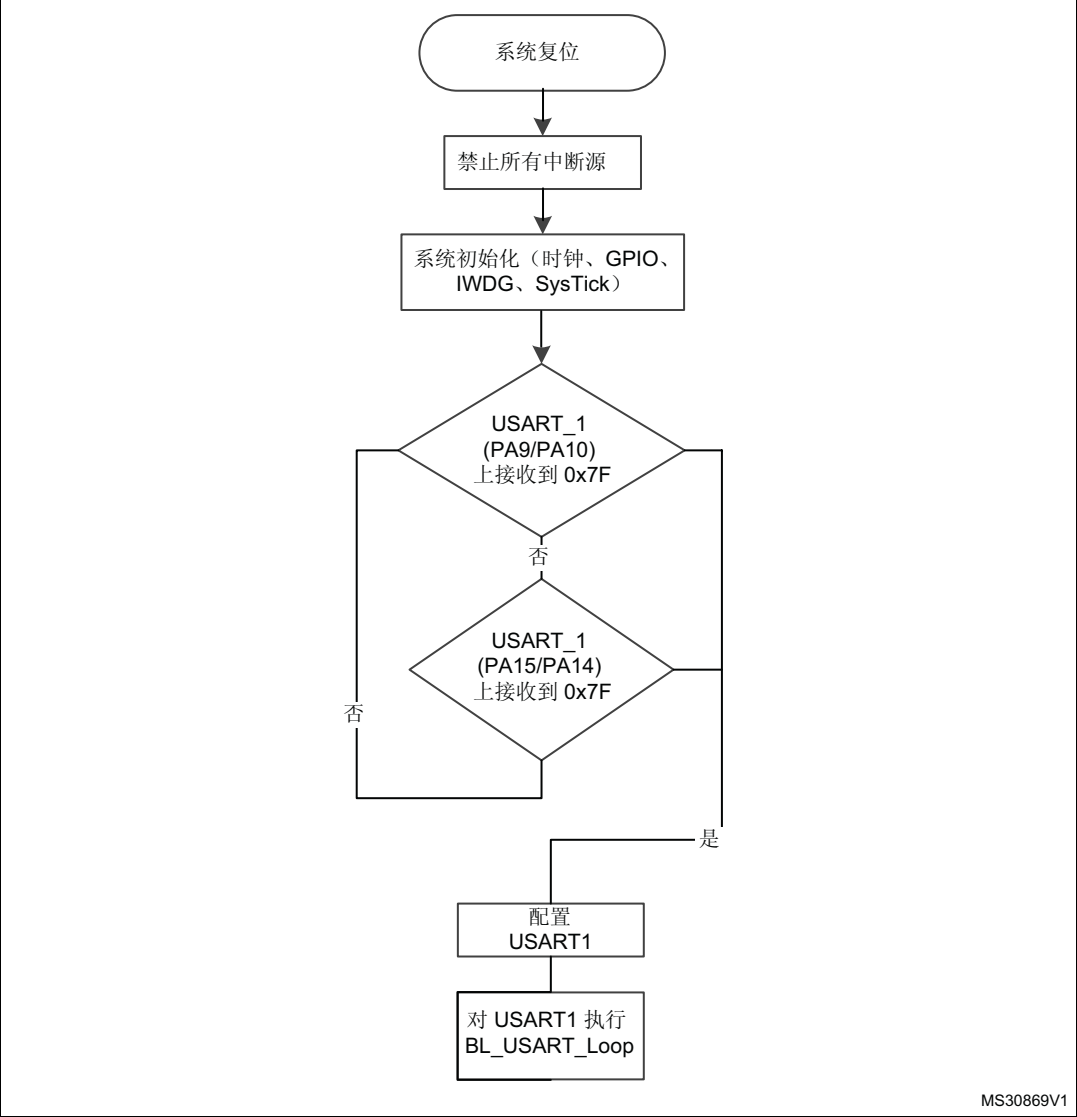
嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 13.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1，可将串行电缆连接到所需映射接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

图 11 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 11. STM32F050xx 器件的自举程序选择



13.4 重要事项

STM32F050xx 器件的自举程序有一些特定的功能需要格外注意，如下所述。

- 选项字节
地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2有关保护级别的详细信息，请参见 RM0091 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序。

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x08000000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x00000000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0091 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

13.5 自举程序版本

表 30 列出了 STM32F050xx 自举程序版本。

表 30. STM32F050xx 自举程序版本

自举程序版本号	说明	已知限制
V1.0	初始自举程序版本	对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号，而不是 1 个 NACK 信号。



14 STM32F372xx 和 STM32F373xx 器件自举程序

本部分使用 STM32F37xxx 指代 STM32F372xx 和 STM32F373xx 器件。

14.1 自举程序配置

STM32F37xxx 器件嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。

表 31 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F37xxx 器件硬件资源。

表 31. 系统存储器自举模式下 STM32F37xxx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	启动时，使用 HSI 将系统时钟频率配置为 48 MHz。如果没有外部时钟 (HSE)，系统时钟将由 HSI 提供。
		HSE 使能	外部时钟可用于所有自举程序接口，其频率值应为如下之一：24、18、16、12、9、8、6、4 或 3 MHz。 PLL 用于生成 48 MHz USB 时钟和 48 MHz 系统时钟。
		-	使能时钟安全系统 (CSS) 中断以用于 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间。此区域包含自举程序固件
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用。
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		



表 31. 系统存储器自举模式下 STM32F37xxx 的配置（续）

自举程序	功能/外设	状态	备注
DFU 自举程序	USB_DM 引脚	复用功能，通过 USB 自动控制	PA11: USB 发送接收数据线
	USB_DP 引脚		PA12: USB 发送接收数据线
	中断	使能	使能 USB 低优先级中断向量以用于 USB DFU 通信。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

分两种工作情况：

- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 Mhz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

HSE 值的检测由自举程序固件完成，该过程以内部振荡器时钟 (HSI) 为依据。因此，当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时（理论值约为 1%），自举程序计算所得的 HSE 频率值可能不正确。这种情况下，自举程序 DFU 接口会出现故障，甚至完全无法工作。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

14.2 自举程序硬件要求

将 STM32F37xxx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLINK 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F37xxx 器件，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 DFU (USB) 连接自举程序，则检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
- 对于 DFU 接口，必须在 USB_DP (PA12) 引脚连接一个 1.5 K (+/- 5%) 的上拉电阻。有关详细信息，请参见通用串行总线规范版本 2.0 的第 7.1.5.1 节。



- 通过以下硬件连接外设：
 - 使用 USART1 时，必须将 RS-232 串行接口（例如 ST3232 RS-232 收发器）直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM32373C-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见“STM32373C-EVAL 板用户手册”，可从意法半导体网站 (www.st.com) 获取该手册。

14.3 自举程序选择

STM32F37xxx 嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

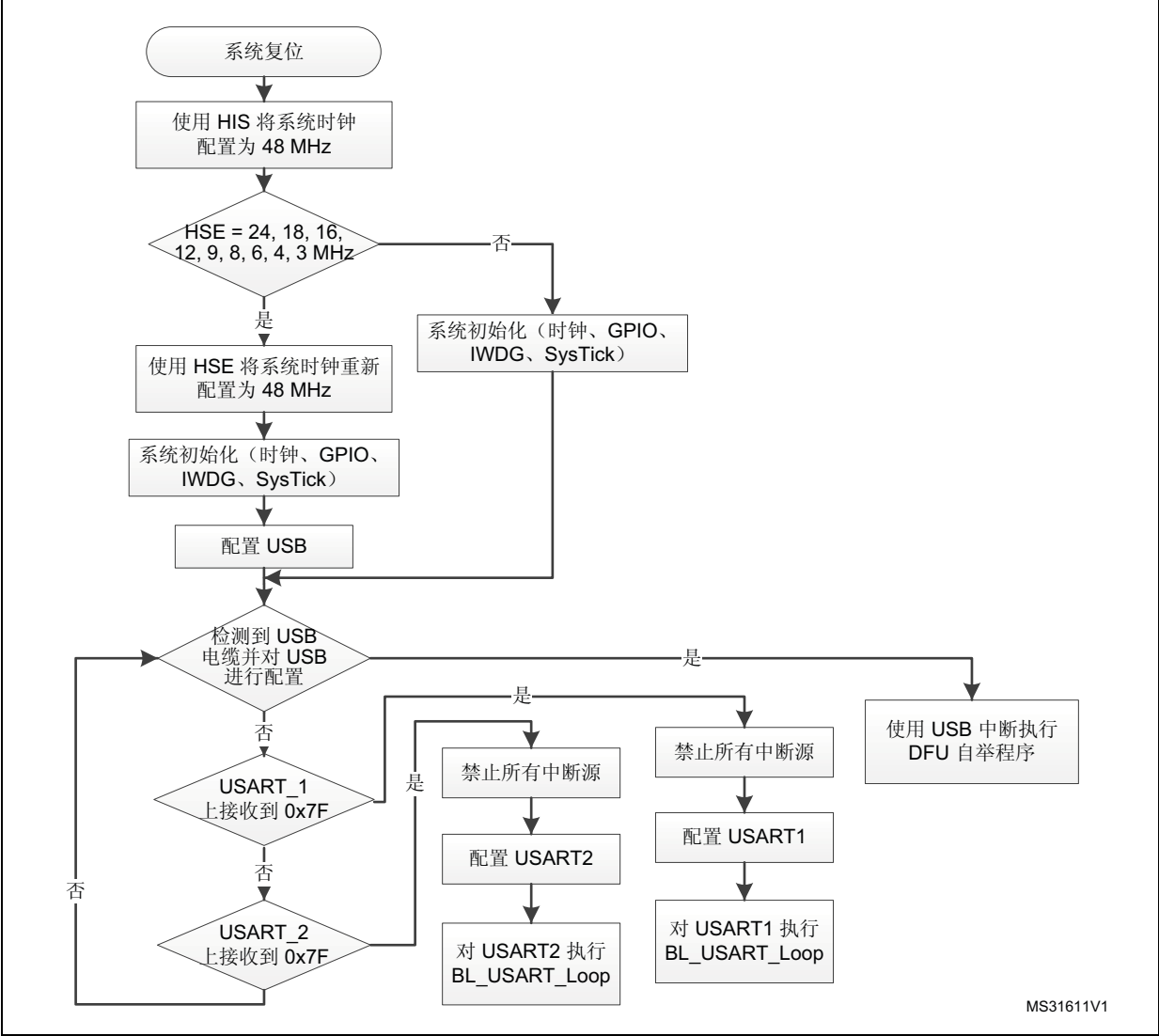
*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 14.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 自举程序，必须确保选择阶段没有 USB 主机连接到 USB 外设（或者在没有外部时钟的情况下，即使连接了主机，USB 也无法启动）。选择 USART 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。为自举程序选择一个接口后，其它接口将禁止。[图12](#)显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 12. STM32F37xxx 器件的自举程序选择



MS31611V1

14.4 重要事项

STM32F37xxx 器件的自举程序有一些特定的功能需要格外注意，如下所述。

- 选项字节
 - 地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
- 有关保护级别的详细信息，请参见 RM0313 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x08000000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x00000000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0313 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

14.5 自举程序版本

表 32 列出了 STM32F37xxx 器件的自举程序版本。

表 32. STM32F37xxx 自举程序版本

自举程序版本号	说明	已知限制
V4.1	初始自举程序版本	无

15 STM32F302xx 和 STM32F303xx 器件自举程序

本部分使用 STM32F30xxx 指代 STM32F302xx 和 STM32F303xx 器件。

15.1 自举程序配置

STM32F30xxx 器件嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)

表 33 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F30xxx 器件硬件资源。

表 33. 系统存储器自举模式下 STM32F30xxx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	启动时，使用 HSI 将系统时钟频率配置为 48 MHz。如果没有外部时钟 (HSE)，系统时钟将由 HSI 提供。
		HSE 使能	外部时钟可用于所有自举程序接口，其频率值应为如下之一：24、18、16、12、9、8、6、4 或 3 MHz。 PLL 用于生成 48 MHz USB 时钟和 48 MHz 系统时钟。
		-	使能时钟安全系统 (CSS) 中断以用于 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间。此区域包含自举程序固件。
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用。
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平。		

表 33. 系统存储器自举模式下 STM32F30xxx 的配置（续）

自举程序	功能/外设	状态	备注
DFU 自举程序	USB_DM 引脚	复用功能，通过 USB 自动控制	PA11: USB 发送接收数据线
	USB_DP 引脚		PA12: USB 发送接收数据线
	中断	使能	使能 USB 低优先级中断向量以用于 USB DFU 通信。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

分两种工作情况：

- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 Mhz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

HSE 值的检测由自举程序固件完成，该过程以内部振荡器时钟 (HSI) 为依据。因此，当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时（理论值约为 1%），自举程序计算所得的 HSE 频率值可能不正确。这种情况下，自举程序 DFU 接口会出现故障，甚至完全无法工作。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已将预分频器设置为最大值）。

15.2 自举程序硬件要求

将 STM32F30xxx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLINK 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F30xxx 器件，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10)、USB_DM (PA11) 和 USB_DP (PA12) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 DFU (USB) 连接自举程序，则检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
- 对于 DFU 接口，必须在 USB_DP (PA12) 引脚连接一个 1.5 K (+/- 5%) 的上拉电阻。有关详细信息，请参见通用串行总线规范版本 2.0 的第“7.1.5.1”节。

- 通过以下硬件连接外设：
 - RS-232 串行接口（例如 ST3232 RS-232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚；使用 USART2 时则必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM32303C-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见“STM32303C-EVAL 板用户手册”，可从意法半导体网站 (www.st.com) 获取该手册。

15.3 自举程序选择

STM32F30xxx 嵌入式自举程序支持三种串行接口：USART1、USART2 和 DFU (USB)。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

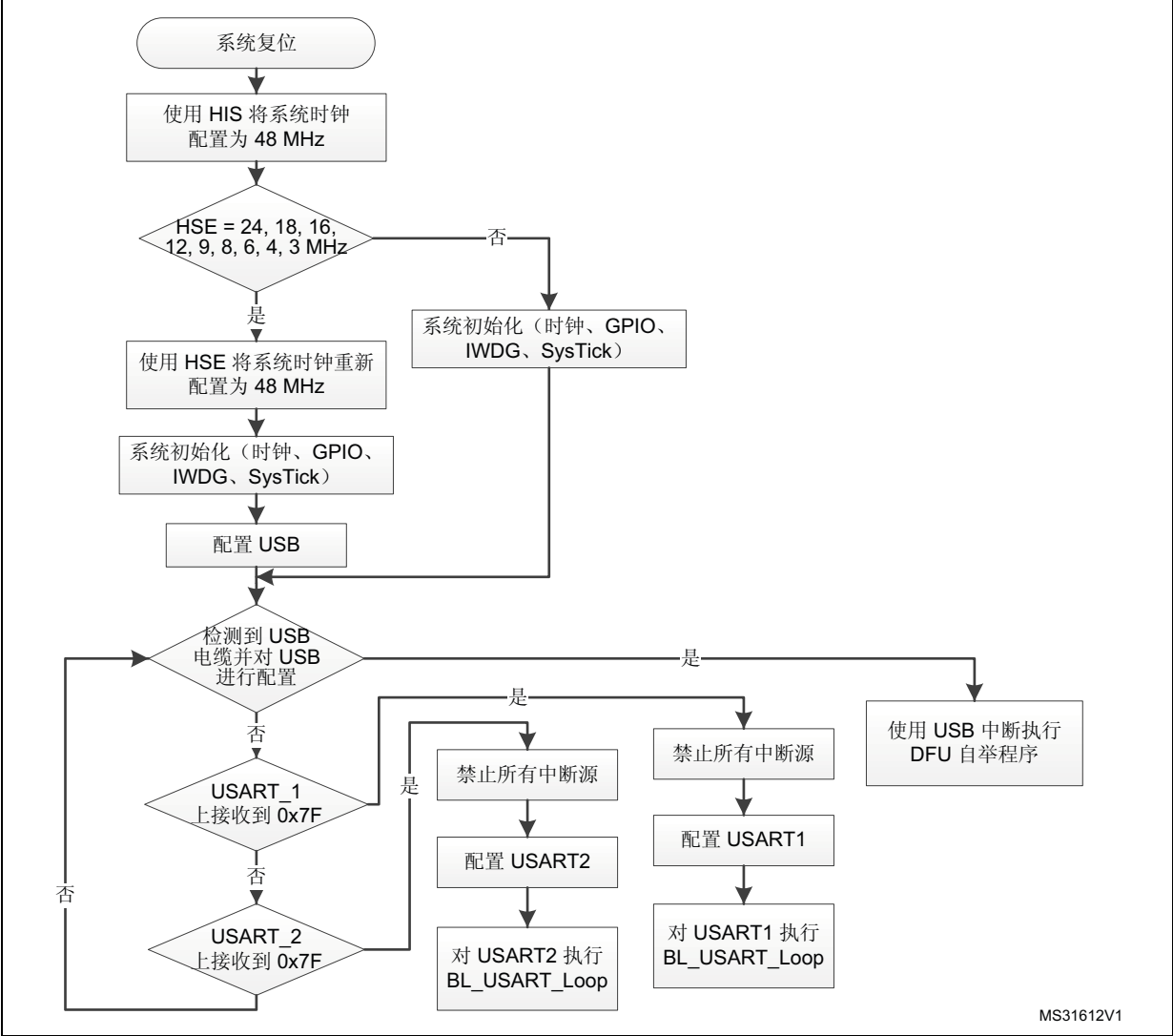
嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

*注：*如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 15.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 自举程序，必须确保选择阶段没有 USB 主机连接到 USB 外设（或者在没有外部时钟的情况下，即使连接了主机，USB 也无法启动）。选择 USART 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。为自举程序选择一个接口后，其它接口将禁止。[图13](#) 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 13. STM32F30xxx 器件的自举程序选择



MS31612V1

15.4 重要事项

STM32F30xxx 器件的自举程序有一些特定的功能需要格外注意，如下所述。

- 选项字节
 - 地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
- 有关保护级别的详细信息，请参见 RM0316 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序。

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x08000000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x00000000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0316 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

15.5 自举程序版本

表 34 列出了 STM32F30xxx 器件的自举程序版本。

表 34. STM32F30xxx 自举程序版本

自举程序版本号	说明	已知限制
V4.1	初始自举程序版本	无

16 STM32F383xx 器件自举程序

本部分使用 STM32F38xxx 指代 STM32F383xx 器件。

16.1 自举程序配置

STM32F38xxx 器件嵌入式自举程序支持三种串行接口：USART1、USART2 和 I2C1。

表 35 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F38xxx 器件硬件资源。

表 35. 系统存储器自举模式下 STM32F38xxx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 HSI 作为系统时钟，频率为 8 MHz。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。禁止窗口功能。
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间。此区域包含自举程序固件。
	RAM	-	自地址 0x20000000 起的 4 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式。
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
I2C1 自举程序	I2C1	使能	HSI 作为 I2C 时钟源。初始化后，I2C1 的配置为：I2C 速度：400 KHz，7 位地址，从模式，从地址：0x6E，模拟滤波器开启。
	I2C1_SCL 引脚	输入/输出	PB6 引脚：I2C1 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入/输出	PB7 引脚：I2C1 数据线在开漏模式下使用。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

对于 I2C1、USART1 和 USART2 自举程序，系统时钟由嵌入式内部高速 RC 提供。这种情况下，执行自举程序代码时无需外部石英时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如表 35 所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已事先将预分频器设置为最大值）。

16.2 自举程序硬件要求

将 STM32F38xxx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLink 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F38xxx 器件，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 连接外设。
 - 如果使用 I2C1 连接自举程序，则必须将 I2C 接口（I2C 主器件）直接连接到 I2C1_SCL (PB6) 和 I2C1_SDA (PB7) 引脚。
 - 如果使用 USART 连接自举程序，则必须使用 RS-232 串行接口（如 ST3232 RS-232 收发器）且：
 - 使用 USART1 时，该串行接口必须直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚。
 - 使用 USART2 时，该串行接口必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
- USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用。因此，应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。
- SDA 线和 SCL 线都须连接一个 1800 欧姆的上拉电阻。此阻值用来修正定时寄存器值。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM32373C-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见“STM32373C-EVAL 板用户手册”，可从意法半导体网站 (www.st.com) 获取该手册。

16.3 自举程序选择

STM32F38xxx 器件嵌入式自举程序支持三种外设接口：I2C1、USART1 和 USART2。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

要将 I2C 自举程序用于 I2C1 接口，可将主机主器件连接到目标从器件 STM32 I2C1。通过数据引脚 (SDA) 和时钟引脚 (SCL) 将主器件与目标器件 STM32 连接在一起。STM32 I2C 接口在总线上识别其自身地址 (0x6E) 后，自举程序固件立即进入无限循环并等待，直到接收到 I2C 自举程序命令。

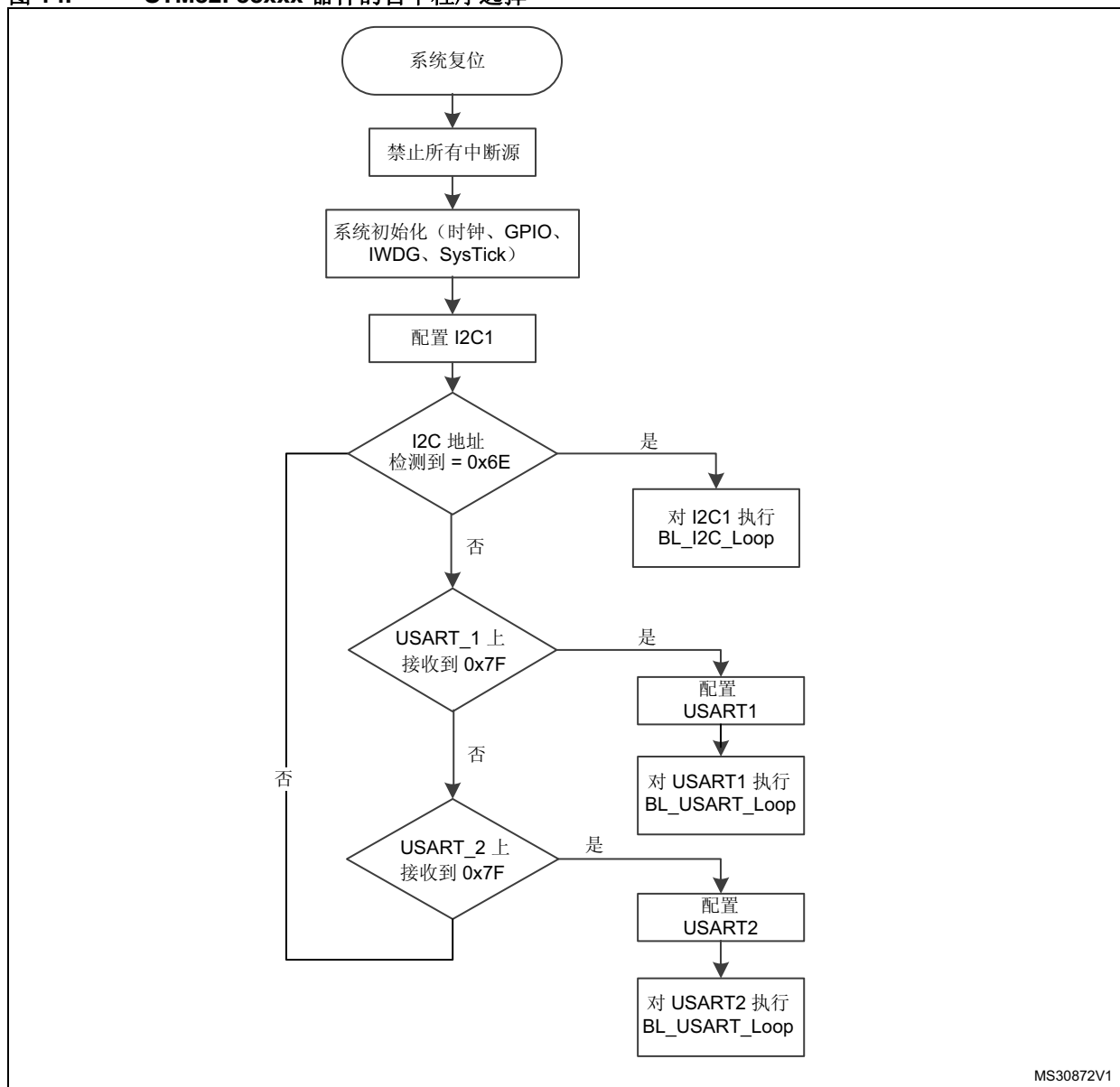
要将 USART 自举程序用于 USART1 或 USART2，可将串行电缆连接到所需接口。自举程序在此接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入无限循环并等待，直到接收到 USART 自举程序命令。

请注意，对于 USART1 和 USART2 接口，自举程序支持的最大波特率为 57600 波特。

为自举程序选择一个接口后，其它接口将禁止。

图14 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 14. STM32F38xxx 器件的自举程序选择



16.4 重要事项

STM32F38xxx 器件的自举程序有一些特定的功能需要格外注意，如下所述。

- 选项字节
地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2有关保护级别的详细信息，请参见 RM0313 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x08000000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x00000000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0313 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

16.5 自举程序版本

表 36 列出了 STM32F38xxx 器件的自举程序版本。

表 36. STM32F38xxx 自举程序版本

自举程序版本号	说明	已知限制
V5.0	初始自举程序版本	对于 USART1 和 USART2 接口，自举程序支持的最大波特率为 57600 波特。



17 STM32F313xx 器件自举程序

本部分使用 STM32F31xxx 指代 STM32F313xx 器件。

17.1 自举程序配置

STM32F31xxx 器件嵌入式自举程序支持三种串行接口：USART1、USART2 和 I2C1。

表 37 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F31xxx 器件硬件资源。

表 37. 系统存储器自举模式下 STM32F31xxx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 HSI 作为系统时钟，频率为 8 MHz。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。禁止窗口功能。
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间。此区域包含自举程序固件。
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平。		
USART2 自举程序	USART2	使能	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。USART2 使用其重映射引脚。
	USART2_RX 引脚	输入	PD6 引脚：USART2 处于接收模式。
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式。
	检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平。		
USART1 和 USART2 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
I2C1 自举程序	I2C1	使能	HSI 作为 I2C 时钟源。初始化后，I2C1 的配置为：I2C 速度：400 KHz，7 位地址，从模式，从地址：0x6E，模拟滤波器开启。
	I2C1_SCL 引脚	输入/输出	PB6 引脚：I2C1 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入/输出	PB7 引脚：I2C1 数据线在开漏模式下使用。
	检测阶段期间，USART1_RX (PA10) 和 USART2_RX (PD6) 引脚必须保持高电平或低电平。		

对于 I2C1、USART1 和 USART2 自举程序，系统时钟由嵌入式内部高速 RC 提供。这种情况下，执行自举程序代码时无需外部石英时钟。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的所有外设寄存器（如上表所示）将初始化为默认复位值。

如果用户应用程序使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已事先将预分频器设置为最大值）。

17.2 自举程序硬件要求

将 STM32F31xxx 器件置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够使 BOOT0 引脚保持高电平，同时将选项字节（起始地址为 0x1FFFF800）中的 nBOOT1 位置 1。可通过 STLink 实用程序或等同工具将该位置 1。

要在系统存储器自举模式下连接 STM32F31xxx 器件，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART2_RX (PD6) 引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART2 连接自举程序，则检测阶段期间，USART1_RX (PA10) 引脚必须保持高电平或低电平，并且不得悬空。
- 连接外设。
 - 如果使用 I2C1 连接自举程序，则必须将 I2C 接口（I2C 主器件）直接连接到 I2C1_SCL (PB6) 和 I2C1_SDA (PB7) 引脚。
 - 如果使用 USART 连接自举程序，则必须使用 RS-232 串行接口（如 ST3232 RS-232 收发器）且：
 - 使用 USART1 时，该串行接口必须直接连接到 USART1_RX (PA10) 和 USART1_TX (PA9) 引脚。
 - 使用 USART2 时，该串行接口必须直接连接到 USART2_RX (PD6) 和 USART2_TX (PD5) 引脚。
- USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。这同样适用于 USART2。
- SDA 线和 SCL 线都须连接一个 1800 欧姆的上拉电阻。此阻值用来修正定时寄存器值。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM32303C-EVAL 板提供了实现此控制功能所需的硬件。有关详细信息，请参见“STM32303C-EVAL 板用户手册”，可从意法半导体网站 (www.st.com) 获取该手册。

17.3 自举程序选择

STM32F31xxx 器件嵌入式自举程序支持三种外设接口：**I2C1**、**USART1** 和 **USART2**。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 **Flash**。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

要将 **I2C** 自举程序用于 **I2C1** 接口，可将主机主器件连接到目标从器件 **STM32 I2C1**。通过数据引脚 (**SDA**) 和时钟引脚 (**SCL**) 将主器件与目标器件 **STM32** 连接在一起。**STM32 I2C** 接口在总线上识别其自身地址 (**0x6E**) 后，自举程序固件立即进入无限循环并等待，直到接收到 **I2C** 自举程序命令。

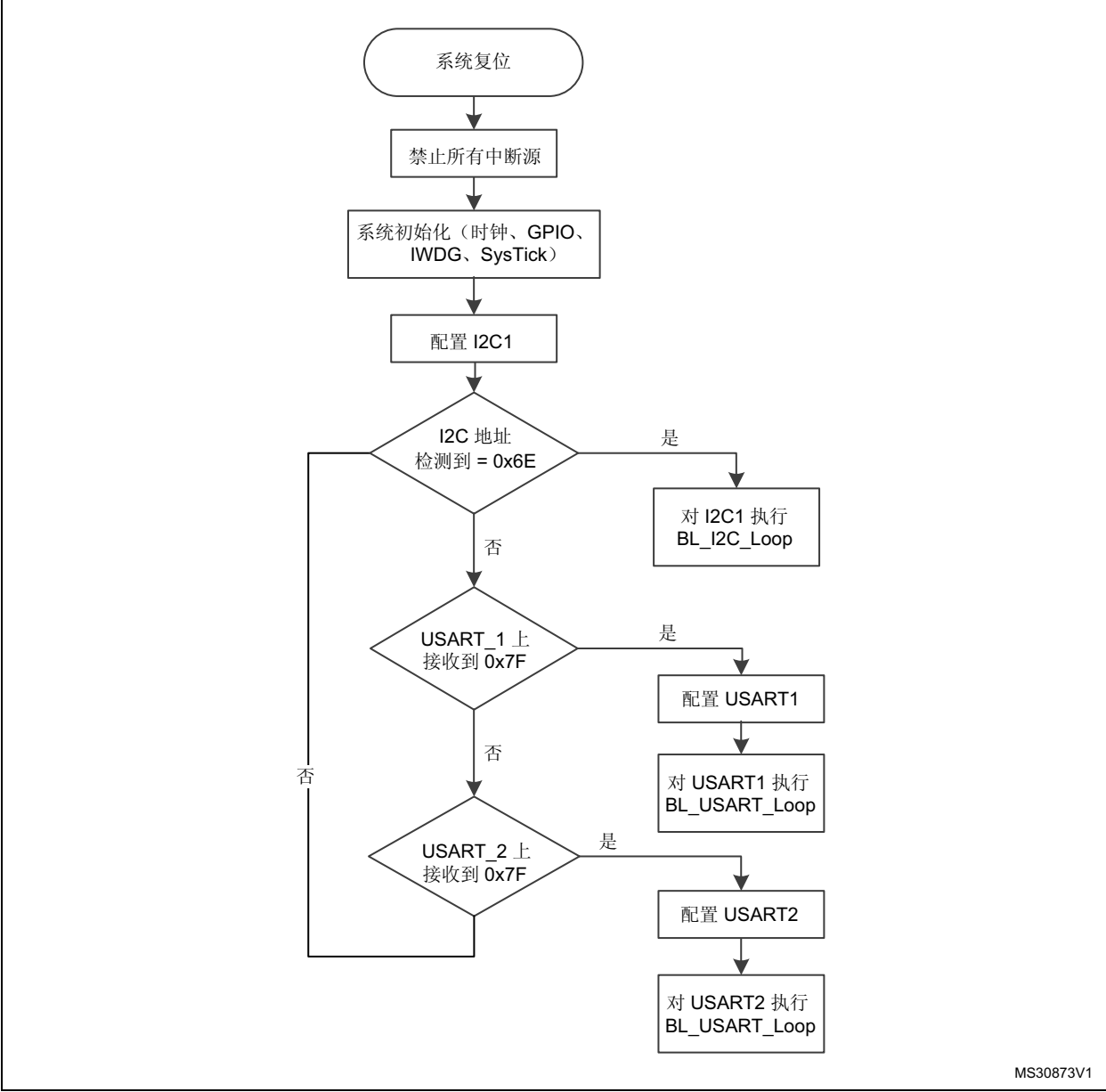
要将 **USART** 自举程序用于 **USART1** 或 **USART2**，可将串行电缆连接到所需接口。自举程序在此接口检测到数据字节 **0x7F** 后，自举程序固件立即执行自动波特率序列，然后进入无限循环并等待，直到接收到 **USART** 自举程序命令。

请注意，对于 **USART1** 和 **USART2** 接口，自举程序支持的最大波特率为 **57600** 波特。

为自举程序选择一个接口后，其它接口将禁止。

[图15](#) 显示了自举程序检测机制。详细信息请参见各外设自举程序的对应部分。

图 15. STM32F31xxx 器件的自举程序选择



17.4 重要事项

STM32F31xxx 器件的自举程序有一些特定的功能需要格外注意。

- 选项字节
地址为 0x1FFFF800。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2有关保护级别的详细信息，请参见 RM0313 参考手册。
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 跳转到用户应用程序

如果必须通过自举程序固件将使用中断的应用程序下载到 Flash 地址 0x08000000，则该应用程序必须在启动时且使能任何中断之前通过软件将 Flash 映射到地址 0x00000000。这可通过编程 SYSCFG_CFGR1 寄存器中的 MEM_MODE 位来实现（有关软件存储器映射的详细信息，请参见 RM0313 参考手册）。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

17.5 自举程序版本

表 38 列出了 STM32F31xxx 器件的自举程序版本。

表 38. STM32F31xxx 自举程序版本

自举程序版本号	说明	已知限制
V5.0	初始自举程序版本	对于 USART1 和 USART2 接口，自举程序支持的最大波特率为 57600 波特。

18 STM32F427xx 和 STM32F437xx 器件自举程序

本部分使用 STM32F427xx/437xx 指代 STM32F427xx 和 STM32F437xx 器件。

18.1 自举程序配置

STM32F427xx/437xx 器件嵌入式自举程序支持四种串行外设：USART1、USART3、CAN2 和 DFU（USB FS 设备）。

表 39 介绍了系统存储器自举模式下，自举程序需要使用的 STM32F427xx/437xx 器件硬件资源。

表 39. 系统存储器自举模式下 STM32F427xx/437xx 的配置

自举程序	功能/外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。 HSI 时钟源用于启动阶段（接口检测阶段）和选择 USARTx 接口期间（选择 CAN 或 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率等于 60 MHz。 仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE 时钟源。 外部时钟提供的频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	使能时钟安全系统 (CSS) 中断以用于 CAN 和 DFU 自举程序。外部时钟发生任何故障（或被移除）都将产生系统复位。
	RAM	-	自地址 0x20000000 起的 8 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFF0000 起的 30424 字节包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为 [1.8V, 2.1V]。可使用自举程序命令在运行期间配置电压范围。请注意，此范围内仅支持按照字节格式（不支持半字、字和双字操作）对内部 Flash 执行写操作。
USART1 自举程序	USART1	使能	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式。
	检测阶段期间，USART3_RX (PB11)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		

表 39. 系统存储器自举模式下 STM32F427xx/437xx 的配置（续）

自举程序	功能/外设	状态	备注
USART3 自举程序 (PB10/PB11)	USART3	使能	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PB11 引脚：USART3 处于接收模式
	USART3_TX 引脚	输出	PB10 引脚：USART3 处于发送模式
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PC11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART3 自举程序 (PC10/PC11)	USART3	使能	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PC11 引脚：USART3 处于接收模式。
	USART3_TX 引脚	输出	PC10 引脚：USART3 处于发送模式。
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、CAN2_RX (PB05)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
USART1 和 USART3 自举程序	SysTick 定时器	使能	用于自动检测用于 USARTx 自举程序的主机串口波特率。
CAN2 自举程序	CAN2	使能	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注：由于 STM32F4xx CAN1 管理着 CAN2 和 SRAM 之间的通信，因此执行 CAN2 自举程序期间要为 CAN1 提供时钟。
	CAN2_RX 引脚	输入	PB05 引脚：CAN2 处于接收模式。
	CAN2_TX 引脚	输出	PB13 引脚：CAN2 处于发送模式。
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11)、OTG_FS_DM (PA11) 和 OTG_FS_DP (PA12) 引脚必须保持高电平或低电平。		
DFU 自举程序	USB_OTG_FS	使能	USB OTG FS 配置为强制器件模式。使能 USB_OTG_FS 中断向量以用于 USB DFU 通信。
	USB_OTG_FS_DM 引脚	输入	PA11 引脚：USB OTG FS DM 线。
	USB_OTG_FS_DP 引脚	输出	PA12 引脚：USB OTG FS DP 线。
	检测阶段期间，USART1_RX (PA10)、USART3_RX (PB11)、USART3_RX (PC11) 和 CAN2_RX (PB05) 引脚必须保持高电平或低电平。		
	TIM11	使能	此定时器用于确定外部时钟的频率值。 确定外部时钟频率之后，RCC 系统工作在 60 MHz 系统时钟下（使用 PLL）。

注：对于 DFU 接口，需要提供外部时钟源 (HSE) 才能执行 USB 操作。HSE 值的检测由自举程序固件完成，该过程以内部振荡器时钟 (HSI) 为依据。因此，当由于温度或其它条件导致内部振荡器精度发生变化并超出容差范围时（理论值约为 1%），自举程序计算所得的 HSE 频率值可能不正确。这种情况下，自举程序 DFU 接口会出现故障，甚至完全无法工作。

对于 USARTx 自举程序，系统时钟由嵌入式内部高速 RC 提供。这种情况下，执行自举程序代码时无需外部石英时钟。该内部时钟同样用于 CAN 和 DFU（USB FS 设备），但仅限选择阶段。选择阶段结束后，执行 CAN 和 DFU 自举程序需要使用频率为 1 MHz 的倍数（介于 4 MHz 到 26 MHz 之间）的外部时钟。

CAN 和 DFU 自举程序采用外部时钟检测机制，能够通过内部高速 RC 和 TIM11 定时器确定外部时钟的频率值。该机制的精度仅支持检测等于 1 MHz 倍数的频率，且范围介于 4 MHz 到 26 MHz 之间。任何其它值都不适用，否则将导致自举程序发生意外行为。

下载应用程序二进制代码后，如果选择执行 Go 命令，则在跳转到用户应用程序之前，自举程序使用的外设寄存器（如上表所示）不会初始化为默认复位值。如果要使用这些寄存器，应在用户应用程序中对其重新配置。因此，如果应用程序正在使用 IWDG，则必须调整 IWDG 预分频器值来满足应用程序的要求（因为自举程序已事先将预分频器设置为最大值）。

18.2 自举程序硬件要求

将 STM32F427xx/437xx 置于系统存储器自举模式所需的硬件包含任何相关电路、开关或跳线，它们能够在复位期间使 BOOT0 引脚保持高电平，使 BOOT1 引脚保持低电平。

要在系统存储器自举模式下连接 STM32F427xx/437xx，必须确认以下条件。

- 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。
 - 如果使用 USART1 连接自举程序，则检测阶段期间，USART3_RX（PC11 和 PB11）、CAN2_RX（PB05）、OTG_FS_DM（PA11）和 OTG_FS_DP（PA12）引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART3（PB10/PB11）连接自举程序，则检测阶段期间，USART1_RX（PA10）、USART3_RX（PC11）、CAN2_RX（PB05）、OTG_FS_DM（PA11）和 OTG_FS_DP（PA12）必须保持高电平或低电平，并且不得悬空。
 - 如果使用 USART3（PC10/PC11）连接自举程序，则检测阶段期间，USART1_RX（PA10）、USART3_RX 引脚（PB11）、CAN2_RX（PB05）、OTG_FS_DM（PA11）和 OTG_FS_DP（PA12）必须保持高电平或低电平，并且不得悬空。
 - 如果使用 CAN2 连接自举程序，则检测阶段期间，USART1_RX（PA10）、USART3_RX（PC11 和 PB11）、OTG_FS_DM（PA11）和 OTG_FS_DP（PA12）引脚必须保持高电平或低电平，并且不得悬空。
 - 如果使用 DFU（USB FS 设备）连接自举程序，则检测阶段期间，USART1_RX（PA10）、USART3_RX（PC11 和 PB11）和 CAN2_RX（PB05）引脚必须保持高电平或低电平，并且不得悬空。
- 连接外设。
 - RS-232 串行接口（例如 ST3232 RS-232 收发器），使用 USART1 时必须将该接口直接连接到 USART1_RX（PA10）和 USART1_TX（PA9）引脚；使用 USART2 时则必须直接连接到 USART2_RX（PD6）和 USART2_TX（PD5）引脚。
 - CAN 接口（CAN 收发器），必须将该接口直接连接到 CAN2_RX（PB5）和 CAN2_TX（PB13）引脚。
 - 经认证的 USB 电缆，必须使用此类电缆连接到微控制器（也可使用 ESD 保护电路）。

USART1_CK、USART1_CTS 和 USART1_RTS 引脚未使用，因此应用程序可将这些引脚用于其它外设或 GPIO。该注释同样适用于 USART3。

用户可借助 PC 串行小应用程序，使用 RS-232 串行接口分别通过 CTS 线和 DCD 线来控制 BOOT0 和复位引脚。用户必须使用全零调制解调器电缆。STM324x7I_EVAL 板提供了实现此控制所需的硬件。有关详细信息，请参见“STM324x7I_EVAL 板用户手册”，可从意法半导体网站 (www.st.com) 获取该手册。

18.3 自举程序选择

STM32F427xx/437xx 嵌入式自举程序支持三种外设接口：USART1、USART3（PB10/PB11 和 PC10/PC11）、CAN2 和 DFU（USB FS 设备）。其中任何一种外设接口都可用于与自举程序进行通信，以及将应用程序代码下载到内部 Flash。

嵌入式自举程序固件可自动检测要使用的外设接口。在无限循环中，它将检测支持的自举程序接口上进行的任何通信。

注： 如下文所述，检测阶段期间，此自举程序未使用的外设 RX 引脚必须保持已知电平（低电平或高电平）且不得悬空。有关详细信息，请参见第 18.2 节：自举程序硬件要求。

要将 USART 自举程序用于 USART1 或 USART3，可将串行电缆连接到所需接口。自举程序在该接口检测到数据字节 0x7F 后，自举程序固件立即执行自动波特率序列，然后进入循环，等待发出任何 USART 自举程序命令。

要使用 CAN2 接口，应将 CAN 电缆连接到 CAN2。自举程序在 CAN2_RX 引脚 (PB5) 上检测到数据帧之后，自举程序固件将进入 CAN 循环，开始确定外部时钟的频率值。

支持等于 1 MHz 倍数的 HSE 频率，且其范围介于 4 MHz 到 26 MHz 之间。任何其它值都会导致意外行为，CAN 自举程序固件将进入无限循环并等待，直到接收到消息。如果没有外部时钟，则产生系统复位。

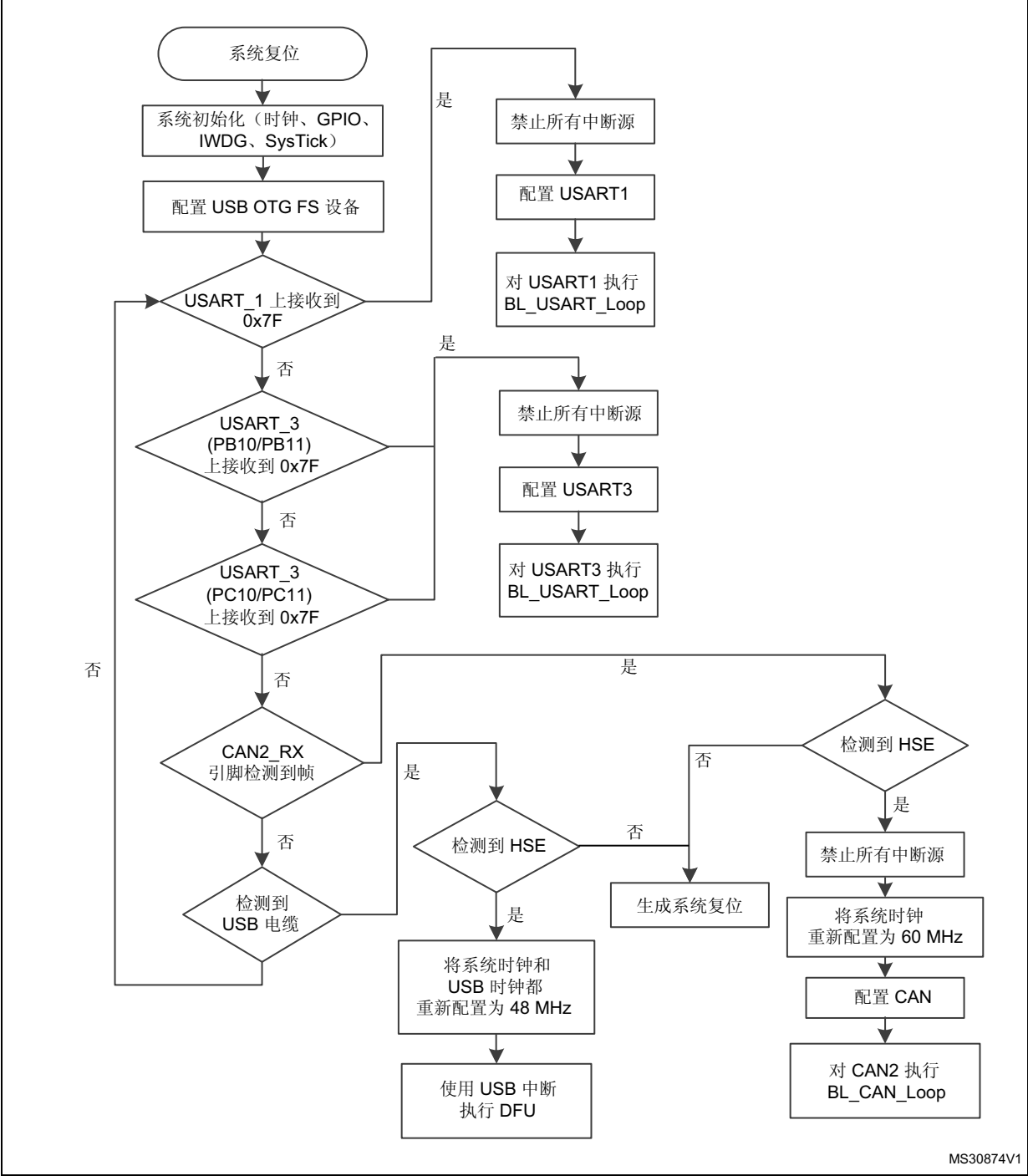
在自举程序固件选择序列期间，无论何时将 USB 电缆插入微控制器的 USB 接口，自举程序都会进入 DFU 自举程序循环，等待发出任何 DFU 自举程序命令。

要使用 USART 或 CAN 自举程序，必须确保选择阶段没有任何 USB 主机连接到 USB 外设。选择 USART 或 CAN 自举程序后，用户可插入一条 USB 电缆，这并不会影响所选自举程序的执行（会产生系统复位的命令除外）。

为自举程序选择一个接口后，所有其它接口都禁止。

图 16 显示了自举程序选择机制。详细信息请参见各外设自举程序的对应部分。

图 16. STM32F427xx/437xx 器件的自举程序选择



18.4 重要事项

STM32F427xx/437xx 自举程序有一些特定的功能需要格外注意。

- STM32F427xx/437xx 器件自举程序固件除了支持标准存储器（内部 Flash、内部 SRAM、选项字节和系统存储器）外，还支持 OTP 存储器（从 0x1FFF7800 到 0x1FFF7A0F 的 528 字节空间）。
- OTP 存储器允许执行读和写操作，但不能使用 Erase 命令进行擦除。向 OTP 存储单元执行写操作时，需确保相关保护位（OTP 存储器的最后 16 个字节）未复位。
- 选项字节
地址为 0x1FFFC000 和 0x1FFEC000。支持三个保护级别：
 - 级别 0
 - 级别 1
 - 级别 2
- 读保护命令对应级别 1 保护。
- 禁止读保护命令对应级别 0 保护。
- 批量擦除命令在 STM32F427xx/437xx 中的执行时间比其它 STM32 器件中长，这是由于它们的容量不同。请确保主机接口在发送批量擦除命令之后有足够长的超时时间来等待确认事件。
- 电压范围配置
电压范围可通过自举程序软件实时更新。每次启动自举程序软件时（系统复位或跳转到自举程序代码后），电压范围都将设为默认值。自举程序软件可通过虚拟存储单元来修改此参数。该存储单元并非物理单元，但可根据所使用的协议（USART、CAN 或 DFU）通过常见的自举程序读/写操作进行读写。该存储单元包含 4 个字节，如表 40 所述。支持 1、2、3 或 4 字节访问。但保留字节应保持默认值 (0xFF)，否则请求将收到 NACK 应答。

表 40. 使用自举程序配置 STM32F427xx/437xx 电压范围

地址	大小	说明
0xFFFF0000	1 字节	该字节控制电压范围的当前值。 0x00: 电压范围 [1.8 V, 2.1 V] 0x01: 电压范围 [2.1 V, 2.4 V] 0x02: 电压范围 [2.4 V, 2.7 V] 0x03: 电压范围 [2.7 V, 3.6 V] 0x04: 电压范围 [2.7 V, 3.6 V] 并且使用双字写 / 擦除操作。 这种情况下，必须通过 VPP 引脚提供 9 V 电压（有关双字写操作步骤的详细信息，请参见 PM0081）。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0001	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0002	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0003	1 字节	保留。 0xFF: 默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。

如果应用程序加载到 Flash 中 0x08000000 以外的地址，则必须重新定位向量表以将加载应用程序的地址作为起始地址。

18.5 自举程序版本

表 41 显示了 STM32F427xx/437xx 自举程序版本。

表 41. STM32F427xx/437xx 自举程序版本

自举程序版本号	说明	已知限制
V3.0	初始自举程序版本。	对于 USART 接口，当发送 Read Memory 或 Write Memory 命令且 RDP 电平有效时，将发送两个连续的 NACK 信号，而不是 1 个 NACK 信号。



19 器件相关的自举程序参数

对于所有 STM32 器件，每种串行外设（USART、CAN、USB 和 I²C）的自举程序协议命令集和序列都相同。但某些参数与器件相关。对一些命令，某些参数值可能取决于所使用的器件。这些参数如下：

- PID（产品 ID），该参数因器件而异
- 支持 Read Memory、Go 和 Write Memory 命令时，自举程序支持的有效存储器地址（RAM、Flash、系统存储器、选项字节区域）。
- 执行 Write Protect 命令时使用的 Flash 扇区的大小。

下表显示了生产中的每种 STM32 器件自举程序的上述参数的值。

表 42. 器件相关的自举程序参数

STM32 系列	器件	产品（器件）ID	RAM 存储器	Flash	Flash 扇区大小	选项字节区域	系统存储器
F1	小容量	0x412	0x20000200 到 0x20002800	0x08000000 到 0x08008000	4 KB（4 页，每页 1 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFF000 - 0x1FFFF800
	中容量	0x410	0x20000200 到 0x20005000	0x08000000 到 0x08020000	4 KB（4 页，每页 1 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFF000 - 0x1FFFF800
	大容量	0x414	0x20000200 到 0x20010000	0x08000000 到 0x08080000	4 KB（2 页，每页 2 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFF000 - 0x1FFFF800
	互连型	0x418	0x20001000 到 0x20010000	0x08000000 到 0x08040000	4 KB（2 页，每页 2 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFB000 - 0x1FFFF800
	中容量超值型	0x420	0x20000200 到 0x20002000	0x08000000 到 0x08020000	4 KB（4 页，每页 1 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFF000 - 0x1FFFF800
	大容量超值型	0x428	0x20000200 到 0x20008000	0x08000000 到 0x08080000	4 KB（2 页，每页 2 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFF000 - 0x1FFFF800
	超大容量	0x430	0x20000800 到 0x20018000	0x08000000 到 0x08100000	4 KB（2 页，每页 2 KB）	0x1FFFF800 - 0x1FFFF80F	0x1FFFE000 - 0x1FFFF800
L1	中容量超低功耗	0x416	0x20000800 到 0x20004000	0x08000000 到 0x08020000	4 KB（16 页，每页 256 字节）	0x1FF80000 - 0x1FF80010	0x1FF00000 - 0x1FF01000
	大容量超低功耗	0x436	0x20001000 到 0x2000C000	0x08000000 到 0x08060000	4 KB（16 页，每页 256 字节）	0x1FF80000 - 0x1FF80020	0x1FF00000 - 0x1FF02000
	中容量增强型超低功耗	0x427	0x20001000 到 0x20008000	0x08000000 到 0x08040000	4 KB（16 页，每页 256 字节）	0x1FF80000 - 0x1FF80020	0x1FF00000 - 0x1FF02000
F2	STM32F2xxx	0x411	0x20002000 到 0x20020000	0x08000000 到 0x08100000	12 个扇区（4x16 KB、1x64 KB、7x128 KB）	0x1FFFC000 - 0x1FFFC00F	0x1FFF0000 - 0x1FFF77FF
F0	STM32F051xx	0x440	0x20000800 到 0x20002000	0x08000000 到 0x08010000	4 KB（4 页，每页 1 KB）	0x1FFFF800 - 0x1FFFF80B	0x1FFFE000 - 0x1FFFF800
	STM32F050xx	0x440	0x20000800 到 0x20002000	0x08000000 到 0x08010000	4 KB（4 页，每页 1 KB）	0x1FFFF800 - 0x1FFFF80B	0x1FFFE000 - 0x1FFFF800

表 42. 器件相关的自举程序参数（续）

STM32 系列	器件	产品 (器件) ID	RAM 存储器	Flash	Flash 扇区大小	选项字节区域	系统存储器
F4	STM32F40xxx/ 41xxx	0x413	0x20002000 到 0x20020000	0x08000000 到 0x08100000	12 个扇区 (4x16 KB、 1x64 KB、 7x128 KB)	0x1FFFC000 - 0x1FFFC00F	0x1FFF0000 - 0x1FFF77FF
	STM32F427xx/ 437xx	0x419	0x20002000 到 0x20030000	0x08000000 到 0x08200000	24 个扇区 (8x16 KB、 2x64 KB、 14x128 KB)	0x1FFFC000 - 0x1FFFC00F 0x1FFEC000 - 0x1FFEC00F	0x1FFF0000 - 0x1FFF77FF
F3	STM32F37xxx	0x432	0x20001400 到 0x20008000	0x08000000 到 0x08040000	4 KB (2 页, 每页 2 KB)	0x1FFFF800 - 0x1FFFF80F	0x1FFFD800 - 0x1FFFF7FF
	STM32F30xxx	0x422	0x20001400 到 0x2000A000	0x08000000 到 0x08040000	4 KB (2 页, 每页 2 KB)	0x1FFFF800 - 0x1FFFF80F	0x1FFFD800 - 0x1FFFF7FF
	STM32F38xxx	0x432	0x20001000 到 0x20008000	0x08000000 到 0x08040000	4 KB (2 页, 每页 2 KB)	0x1FFFF800 - 0x1FFFF80F	0x1FFFD800 - 0x1FFFF7FF
	STM32F31xxx	0x422	0x20001400 到 0x2000A000	0x08000000 到 0x08040000	4 KB (2 页, 每页 2 KB)	0x1FFFF800 - 0x1FFFF80F	0x1FFFD800 - 0x1FFFF7FF

20 自举程序时序特性

本部分根据产品介绍了自举程序固件的主要启动时序。这些时序可用于设置连接超时，即主机在与自举程序建立同步之前等待的时间。

本文将介绍三种时序：

- 与产品相关且直接提取自产品数据手册的硬件相关时序。
- 与总线的波特率和数据量相关的通信相关时序。这些时序仅取决于通信接口配置和主机行为。
- 与自举程序软件操作相关的自举程序软件相关时序。

除非特别说明，否则本部分所述的全部时序都以毫秒 (ms) 表示。

20.1 USART 自举程序时序特性

使用 USART 自举程序时，需要考虑主机操作的两种主要时序：

- 时序 A
自举程序复位后，此时序相当于主机在发送同步数据 (0x7F) 以正确配置自举程序波特率检测之前等待的时间。本部分将此时序称为时序 A。
- 时序 B
发送同步数据 (0x7F) 后，此时序相当于主机在收到第一个确认响应（表示自举程序准备好接收和执行主机命令）之前等待的时间。本部分将此时序称为时序 B。

时序 A 和时序 B 由不同的子时序组成，如图 17 所述。

图 17. USART 自举程序时序波形

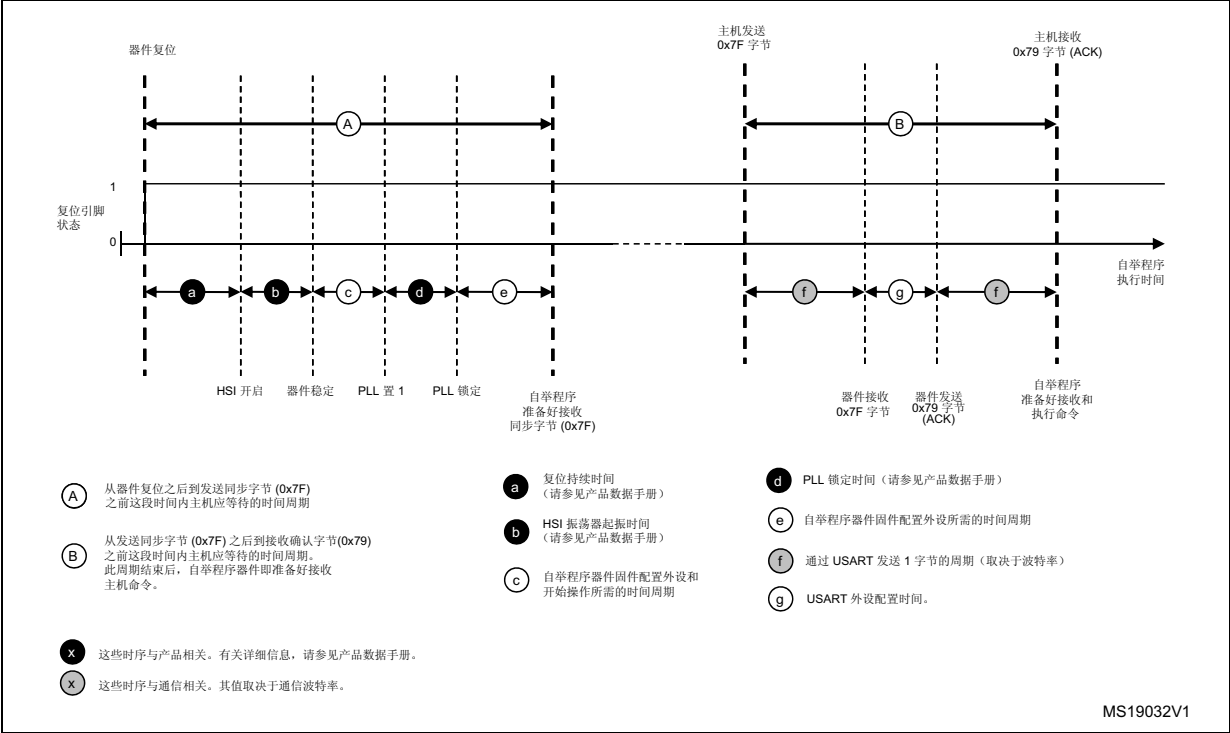


表 43、表 44、表 45、表 46、表 47、表 48、表 49、表 50 和表 51 列出了每种产品的时序值。

表 43. 小/中/大容量超值型器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.004	-	ms
d	PLL 锁相时间	0.2	-	ms
e	自举程序固件操作	0.002	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.002	-	ms
A	时间 = a + b + c + d + e	1.207	4.708	ms
B	时间 = (2 x f) + g	0.15825	15.002	ms

表 44. 超大容量器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.02	-	ms
d	PLL 锁相时间	0.2	-	ms
e	自举程序固件操作	0.006	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.006	-	ms
A	时间 = a + b + c + d + e	1.227	4.728	ms
B	时间 = (2 x f) + g	0.16225	15.006	ms

表 45. 互连型器件的 USART 自举程序时序 (PA9 引脚为低电平)

时间	说明	最小值	最大值	单位
a	复位持续时间	1	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.025	-	ms
d	PLL 锁相时间	0.35	-	ms
e	自举程序固件操作	0.02	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.007	-	ms
A	时间 = a + b + c + d + e	1.396	4.897	ms
B	时间 = (2 x f) + g	0.16325	15.007	ms

对于互连型器件，PA9 引脚 (USB_VBUS) 用于检测 USB 主机连接。仅当 PA9 引脚在检测阶段具有高电平（即主机与该端口相连并在 USB 总线上提供 5 V 电压）时，才会对 USB 外设执行初始化。此时初始化和关闭 USB 外设需要耗用更长的时间。

要在未使用 PA9 引脚时将互连型器件的自举程序检测时间缩至最短，可在检测阶段器件复位之后，使 PA9 引脚保持低电平，直到发送器件 ACK 信号。

表 46. 互连型器件的 USART 自举程序时序 (PA9 引脚为高电平)

时间	说明	最小值	最大值	单位
a	复位持续时间	1	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.025	-	ms
d	PLL 锁相时间	0.35	-	ms
e	自举程序固件操作	523	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	105	-	ms
A	时间 = a + b + c + d + e	524.376	527.877	ms
B	时间 = (2 x f) + g	105.1563	120	ms

表 47. 中容量超低功耗器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.4	1.6	ms
b	MSI 振荡器稳定时间	-	0.04	ms
c	自举程序固件操作	0.064	-	ms
d	HSI 振荡器起振时间	0.0037	0.006	ms
e	自举程序固件操作	0.034	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.008	-	ms
A	时间 = a + b + c + d + e	0.5417	1.744	ms
B	时间 = (2 x f) + g	0.16425	15.008	ms

表 48. 大容量超低功耗器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.4	1.6	ms
b	MSI 振荡器稳定时间		0.07	ms
c	自举程序固件操作	0.058		ms
d	HSI 振荡器起振时间		0.006	ms
e	自举程序固件操作	0.174		ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0078		ms
A	时间 = a + b + c + d + e	0.708	1.908	ms
B	时间 = (2 x f) + g	0.16405	15.0078	ms

表 49. STM32F2xxxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.5	3.0	ms
b	HSI 振荡器起振时间	0.0022	0.004	ms
c	自举程序固件操作	0.01	-	ms
d	PLL 锁相时间	0.075	0.2	ms
e	自举程序固件操作	84	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.009	-	ms
A	时间 = a + b + c + d + e	84.5872	87.214	ms
B	时间 = (2 x f) + g	0.16525	15.009	ms

表 50. STM32F40xxx/41xxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.5	3.0	ms
b	HSI 振荡器起振时间	0.0022	0.004	ms
c	自举程序固件操作	0.01	-	ms
d	PLL 锁相时间	0.075	0.2	ms
e	自举程序固件操作	84	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.009	-	ms
A	时间 = a + b + c + d + e	84.5872	87.214	ms
B	时间 = (2 x f) + g	0.16525	15.009	ms

表 51. STM32F051xx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c + d + e	自举程序固件操作	0.111		ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0095		ms
A	时间 = a + b + c + d + e	1.612	4.613	ms
B	时间 = (2 x f) + g	0.16575	15.0095	ms

表 52. 中容量增强型器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.4	1.6	ms
b	MSI 振荡器稳定时间	0.07	0.07	ms
c	自举程序固件操作	0.0558	-	ms
d	HSI 振荡器起振时间	0.0037	0.006	ms
e	自举程序固件操作	0.157	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0078	-	ms
A	时间 = a + b + c + d + e	0.6865	1.8888	ms
B	时间 = (2 x f) + g	0.16405	15.0078	ms

表 53. STM32F050xx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c + d + e	自举程序固件操作	0.09	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0064	-	ms
A	时间 = a + b + c + d + e	1.591	4.592	ms
B	时间 = (2 x f) + g	0.16265	15.0064	ms

表 54. STM32F37xxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.015	-	ms
d	PLL 锁相时间	0.2	0.2	ms
e	自举程序固件操作	43.2	-	ms
c + d ⁽¹⁾	自举程序固件操作	-	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.00245	-	ms
A	时间 = a + b + c + d + e	44.916	47.917	ms
B	时间 = (2 x f) + g	0.1587	15.0025	ms

1. 没有外部时钟 (HSE) 时, 此时序用来代替时序 “d” 和 “e”。

表 55. STM32F30xxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c	自举程序固件操作	0.6	-	ms
d	PLL 锁相时间	0.2	0.2	ms
e	自举程序固件操作	41	-	ms
c + d ⁽¹⁾	自举程序固件操作	-	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0022		ms
A	时间 = a + b + c + d + e	43.301	46.302	ms
B	时间 = (2 x f) + g	0.15845	15.0022	ms

1. 没有外部时钟 (HSE) 时, 此时序用来代替时序 “d” 和 “e”。

表 56. STM32F38xxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c + d + e	自举程序固件操作	0.041		ms
f	一个 USART 字节发送周期	0.15625	7.5	ms
g	自举程序固件操作	0.0099		ms
A	时间 = a + b + c + d + e	1.542	4.543	ms
B	时间 = (2 x f) + g	0.3224	15.0099	ms

表 57. STM32F31xxx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	1.5	4.5	ms
b	HSI 振荡器起振时间	0.001	0.002	ms
c + d + e	自举程序固件操作	0.0401		ms
f	一个 USART 字节发送周期	0.15625	7.5	ms
g	自举程序固件操作	0.0095		ms
A	时间 = a + b + c + d + e	1.5411	4.5421	ms
B	时间 = (2 x f) + g	0.322	15.0095	ms

表 58. STM32F427xx 和 STM32F437xx 器件的 USART 自举程序时序

时间	说明	最小值	最大值	单位
a	复位持续时间	0.5	3.0	ms
b	HSI 振荡器起振时间	0.0022	0.004	ms
c	自举程序固件操作	0.0089	-	ms
d	PLL 锁相时间	0.075	0.2	ms
e	自举程序固件操作	81.047	-	ms
f	一个 USART 字节发送周期	0.078125	7.5	ms
g	自举程序固件操作	0.0058	-	ms
A	时间 = a + b + c + d + e	81.6331	84.2599	ms
B	时间 = (2 x f) + g	0.16205	15.0058	ms

20.2 USB 自举程序时序特性

使用 USB 自举程序时，需要考虑的主机操作主要时序如下：

- 时序 A
自举程序复位后，此时序相当于主机在启动器件的连接序列前等待的时间。该时序与 USART 连接超时相似，如第 20.1 节所述。本部分将此时序称为时序 A。
- 时序 B
启动连接序列时，此时序相当于器件要与主机建立正确连接所需的时间（表示自举程序准备好接收和执行主机命令）。此时序包括枚举和 DFU 组件配置（例如内部 Flash）。本部分将此时序称为时序 B。

对于互连型器件，如果外部 HSE 晶振频率不等于 25 MHz（14.7456 MHz 或 8 MHz），则器件在与主机建立正确连接之前，将执行多个失败的枚举（包含连接/断开连接序列）。这是由基于 SOF 检测的 HSE 自动检测机制引起的。

时序 A 和 B 由不同的子时序组成，如图 18 所述。有关时序 A 的值（与 USART 自举程序完全相同），请参见表 43、表 44、表 45、表 46、表 47、表 48、表 49、表 50 和表 51；有关时序 B 的值，请参见表 59、表 60、表 61 和表 62。

注：对于 USB 接口而言，由于连接时序取决于环境和主机配置（节点（集线器）数）、主机速度、USB 总线的通信量、主机加载...），因此仅提供最短时序。

图 18. USB 自举程序时序波形

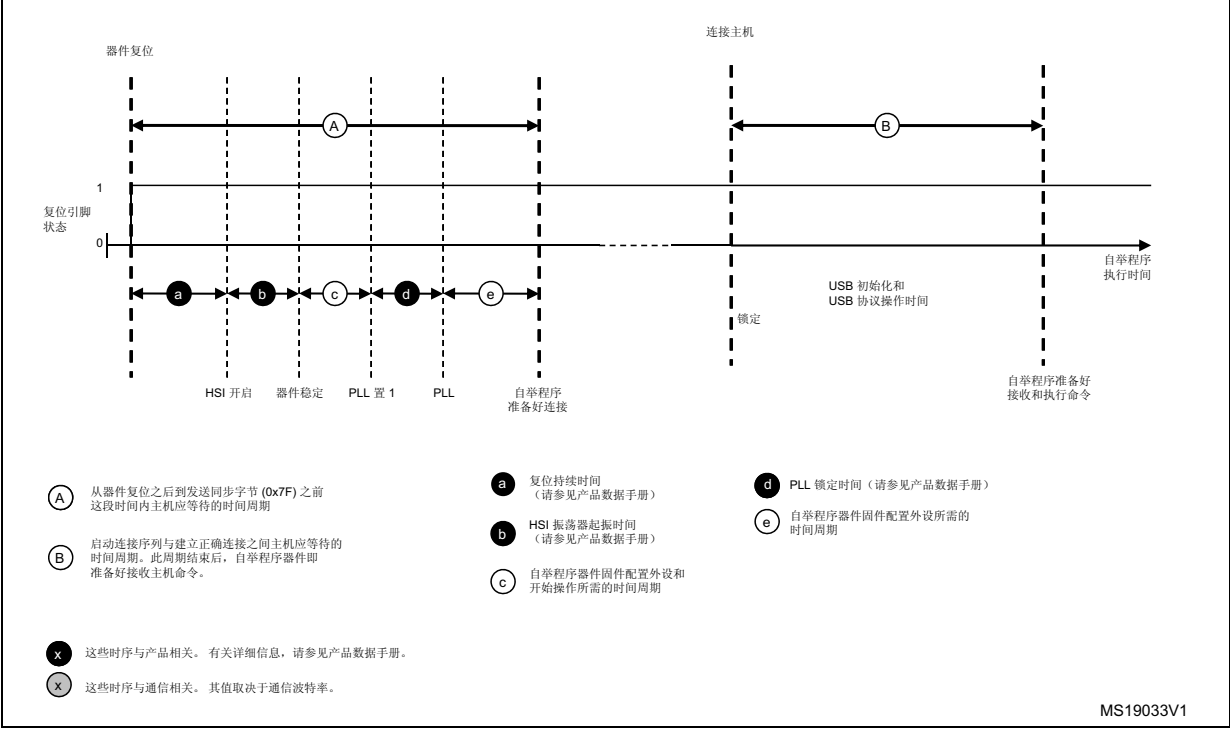


表 59. 互连型器件的 USB 最短时序

时间	说明	25 MHz	14.7456 MHz	8 MHz	单位
a	复位持续时间	1	1	1	ms
b	HSI 振荡器起振时间	0.001	0.001	0.001	ms
c	自举程序固件操作	0.025	0.025	0.025	ms
d	PLL 锁相时间	0.35	0.35	0.35	ms
e	自举程序固件操作	523	523	523	ms
A	时间 = a + b + c + d + e	524.376	524.376	524.376	ms
B	建立连接	460	4500	13700	ms

表 60. 大容量超低功耗器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	0.4	ms
b	MSI 振荡器稳定时间	0.07	ms
c	自举程序固件操作	0.058	ms
d	HSI 振荡器起振时间	0.0037	ms
e	自举程序固件操作	0.174	ms
A	时间 = a + b + c + d + e	0.7057	ms
B	建立连接	849	ms

表 61. STM32F2xxx 器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	0.5	ms
b	HSI 振荡器起振时间	0.0022	ms
c	自举程序固件操作	0.01	ms
d	PLL 锁相时间	0.075	ms
e	自举程序固件操作	84	ms
A	时间 = a + b + c + d + e	84.5872	ms
B	建立连接	54	ms

表 62. STM32F40xxx/41xxx 器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	0.5	ms
b	HSI 振荡器起振时间	0.0022	ms
c	自举程序固件操作	0.01	ms
d	PLL 锁相时间	0.075	ms
e	自举程序固件操作	84	ms
A	时间 = a + b + c + d + e	84.5872	ms
B	建立连接	54	ms

表 63. 中容量增强型器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	0.4	ms
b	MSI 振荡器稳定时间	0.07	ms
c	自举程序固件操作	0.0558	ms
d	HSI 振荡器起振时间	0.0037	ms
e	自举程序固件操作	0.157	ms
A	时间 = a + b + c + d + e	0.6865	ms
B	建立连接	849	ms

表 64. STM32F37xxx 器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	1.5	ms
b	HSI 振荡器起振时间	0.001	ms
c	自举程序固件操作	0.015	ms
d	PLL 锁相时间	0.2	ms
c + d(1)	自举程序固件操作（如果没有 HSE）		
e	自举程序固件操作	43.2	ms
A	时间 = a + b + c + d + e	44.916	ms
B	建立连接	560	ms

表 65. STM32F30xxx 器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	1.5	ms
b	HSI 振荡器起振时间	0.001	ms
c	自举程序固件操作	0.6	ms
d	PLL 锁相时间	0.2	ms
e	自举程序固件操作	41	ms
A	时间 = a + b + c + d + e	43.301	ms
B	建立连接	620	ms

表 66. STM32F427xx/437xx 器件的 USB 最短时序

时间	说明	最小值	单位
a	复位持续时间	0.5	ms
b	HSI 振荡器起振时间	0.0022	ms
c	自举程序固件操作	0.0089	ms
d	PLL 锁相时间	0.075	ms
e	自举程序固件操作	81.047	ms
A	时间 = a + b + c + d + e	81.6331	ms
B	建立连接	773.13	ms

对于 STM32F2xxxx、STM32F40xxx/41xxx 和 STM32F427xx/437xx 器件自举程序，时序值与 HSE 晶振频率无关。HSE 晶振频率值使用 TIM11 定时器和 HSI 内部振荡器通过周期测量来检测。



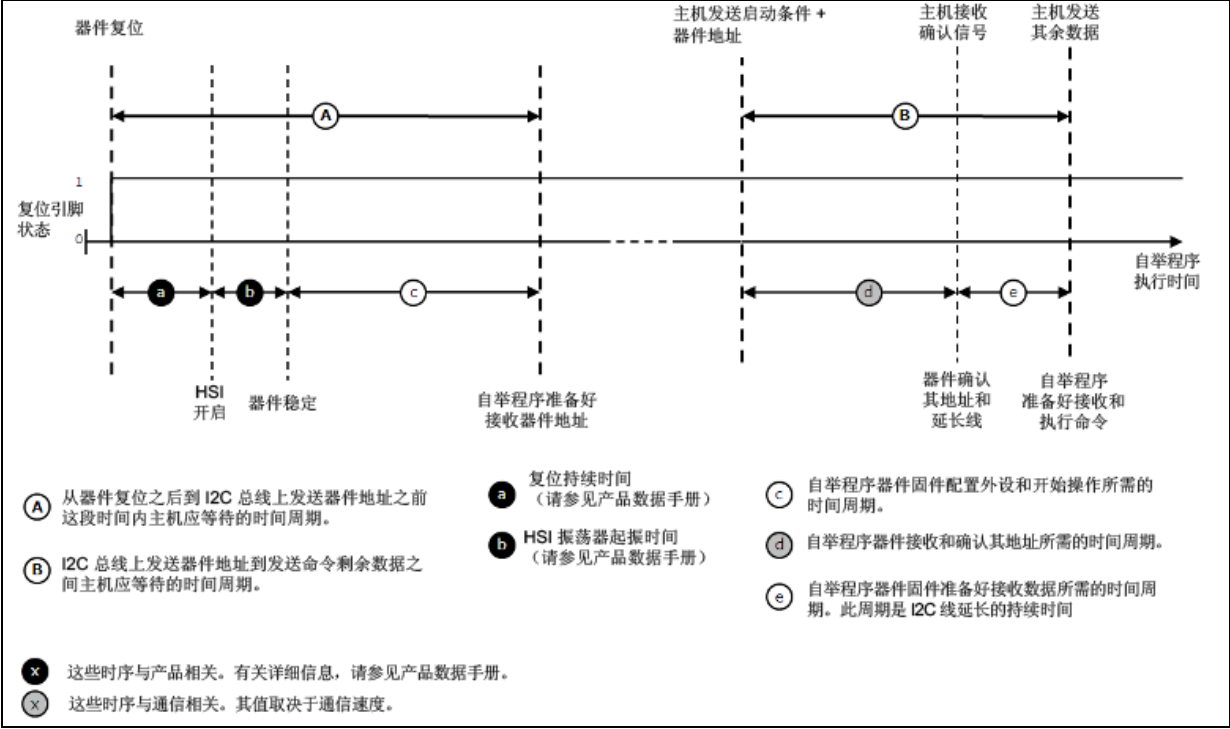
20.3 I2C 自举程序时序特性

对于 I2C 自举程序，需要确定两个主要时序：

- 自举程序器件复位后，主机在启动器件的连接序列之前应等待多长时间。本部分将此时序称为时序 A。
- 发送器件地址后，器件在检测到启动条件后需要多长时间才准备好接收剩余数据。此时序包括 I2C 线上确认与接收第一个数据之间的时钟线延长周期。本部分将此时序称为时序 B。

这两个时序（A 和 B）都由不同的子时序组成，如下图所述：

图 19. I2C 自举程序时序波形



注：I2C 通信中采用了超时机制，只有遵守超时机制才能正常执行自举程序命令。此超时在同一命令中的两个 I2C 帧之间实施（例如，对于 Write memory 命令，将在命令发送帧与地址存储发送帧之间插入超时）。此外也将在同一 I2C 帧中的两个连续数据接收或发送操作之间插入同一超时周期。如果超时周期已过，则生成系统复位以避免自举程序崩溃。有关每种产品的 I2C 超时值，请参见以下各表。

在“擦除存储器”命令和“禁止读出保护”命令中，应在主机端实施超时时考虑页擦除持续时间。发送要擦除的页的代码后，主机应等待，直到自举程序器件执行页擦除并完成擦除命令的剩余步骤。

表 67. STM32F38xxx 器件的 I2C 最短时序

时间	说明	最小值	单位
a	复位持续时间	1.5	ms
b	HSI 振荡器起振时间	0.001	ms
c	自举程序固件操作	0.041	ms
d	启动条件 + 1 字节发送	0.0225	ms
e	I2C 线延长持续时间	0.0055	ms
A	时间 = a + b + c	1.542	ms
B	时间 = d + e	0.028	ms
-	I2C 超时	10	ms

表 68. STM32F31xxx 器件的 I2C 最短时序

时间	说明	最小值	单位
a	复位持续时间	1.5	ms
b	HSI 振荡器起振时间	0.001	ms
c	自举程序固件操作	0.0401	ms
d	启动条件 + 1 字节发送	0.0225	ms
e	I2C 线延长持续时间	0.0054	ms
A	时间 = a + b + c	1.5411	ms
B	时间 = d + e	0.0279	ms
-	I2C 超时	10	ms

21 版本历史

表 69. 文档版本历史

日期	版本	变更
2007 年 10 月 22 日	1	初始版本。
2008 年 01 月 22 日	2	<p>生产中的所有 STM32（版本 B 和版本 Z）都包括此应用笔记中所述的自举程序。</p> <p>修改了：第 3.1 节：自举程序激活和第 1.4 节：自举程序代码序列。</p> <p>增加了：第 1.3 节：硬件要求、第 1.5 节：选择 USARTx 波特率、第 1.6 节：使用自举程序和第 3.2 节：退出系统存储器自举模式。</p> <p>增加了与 Get、Get Version 和 Read Protection Status 以及 Get ID 命令相关的注 2（表 3：自举程序命令）和注 3。</p> <p>从文档中删除“permanent”一词 (Permanent Write Unprotect/Readout Protect/Unprotect)。少量文本更改。</p> <p>将自举程序版本升级为 2.0。</p>
2008 年 05 月 26 日	3	<p>少量文本更改。在表 4：系统存储器自举模式下 STM32F10xxx 器件的配置增加了 RAM 和系统存储器。</p> <p>删除了第 8 页的第 1.6 节：使用自举程序。</p> <p>修改了擦除和注 3，并将注 1 增加到第 9 页的表 3：自举程序命令。</p> <p>修改了第 11 页的字节 3。</p> <p>修改了第 13 页的字节 2。</p> <p>修改了第 15 页的字节 2、字节 3-4 和字节 5，并且修改了注 3。</p> <p>修改了第 18 页的字节 8。</p> <p>在第 18 页的第 2.5 节：Go 命令中增加了注释。</p> <p>修改了第 20 页的图 11：Go 命令：器件端。</p> <p>在第 21 页的第 2.6 节：Write Memory 命令中增加了注释。</p> <p>修改了第 24 页的字节 8。</p> <p>修改了图 14：Erase Memory 命令：主机端和图 15：Erase Memory 命令：器件端。</p> <p>修改了第 26 页的字节 3。</p> <p>第 9 页的表 3：自举程序命令。</p> <p>在第 27 页的第 2.8 节：Write Protect 命令中修改并增加了注释。</p> <p>修改了图 16：Write Protect 命令：主机端、图 17：Write Protect 命令：器件端、图 19：Write Unprotect 命令：器件端、图 21：Readout Protect 命令：器件端和图 23：Readout Unprotect 命令：器件端。</p>
2009 年 01 月 29 日	4	<p>此应用笔记同样适用于 STM32F102xx 微控制器。</p> <p>将自举程序版本更新为 V2.2（请参见表 4：自举程序版本）。</p>

表 69. 文档版本历史 (续)

日期	版本	变更
2009 年 11 月 19 日	5	<p>在表：系统时钟由嵌入式内部高速 RC 提供，自举程序代码不需要外部石英时钟。中增加了 IWDG。增加了注。</p> <p>在整个文档中将 BL 更改为自举程序。</p> <p>修改了表：系统时钟由嵌入式内部高速 RC 提供，自举程序代码不需要外部石英时钟。中的 Go 命令说明。</p> <p>在第 2.4 节：Read Memory 命令中更正了自举程序等待的字节数。</p> <p>修改了图 10：Go 命令：主机端下的注释。</p> <p>删除了第 2.5 节：Go 命令中的注释，同时增加了注释。</p> <p>在第 2.6 节：Write Memory 命令中指定了起始 RAM 地址并增加了注释。将 Write Memory 命令发送到选项字节区域时擦除了所有选项。</p> <p>修改了图 11：Go 命令：器件端。</p> <p>修改了图 13：Write Memory 命令：器件端。</p> <p>在第 2.7 节：Erase Memory 命令中增加了注释并修改了主机发送的字节 3 和 4。</p> <p>在第 2.8 节：Write Protect 命令中增加了注释。</p>
2010 年 03 月 09 日	6	<p>重新整理应用笔记。增加了超值型和互连型器件自举程序（替换 AN2662）。</p> <p>更改了前言。增加了词汇表。</p>
2010 年 04 月 20 日	7	<p>相关文档：增加了超大容量器件的数据手册和编程手册。</p> <p>词汇表：增加了超大容量器件。</p> <p>表 3：增加了超大容量器件的相关信息。</p> <p>第 4.1 节：自举程序配置：更新了第一个句子。</p> <p>第 5.1 节：自举程序配置：更新了第一个句子。</p> <p>增加了第 6 节：STM32F101xx 和 STM32F103xx 超大容量器件自举程序。</p> <p>表 42：增加了超大容量器件的相关信息。</p>
2010 年 10 月 08 日	8	在表 3 和表 42 中增加了大容量超值型器件的信息。
2010 年 10 月 14 日	9	删除了淘汰器件的参考资料。
2010 年 11 月 26 日	10	增加了超低功耗器件的信息。
2011 年 04 月 13 日	11	<p>增加了 STM32F205/215xx 和 STM32F207/217xx 器件的相关信息。</p> <p>增加了第 20 节：自举程序时序特性。</p>
2011 年 06 月 06 日	12	<p>更新了：</p> <ul style="list-style-type: none"> 表 12：STM32L15xxx 中容量器件自举程序版本 表 16：系统存储器自举模式下 STM32F2xxx 的配置 表 18：STM32F2xxx 自举程序 V2.x 版本 表 21：STM32F2xxx 自举程序 V3.x 版本
2011 年 11 月 28 日	13	<p>增加 STM32F405/415xx 和 STM32F407/417xx 自举程序以及 STM32F105xx/107xx 自举程序 V2.1 的相关信息。</p> <p>在第 4 节：STM32F100xx、STM32F101xx、STM32F102xx、STM32F103xx 中容量和大容量超值型自举程序标题和概述中增加了超值型器件。</p>

表 69. 文档版本历史 (续)

日期	版本	变更
2012 年 07 月 30 日	14	<p>增加了从 STM32F051x6/STM32F051x8 器件到大容量超低功耗 STM32L151xx、STM32L152xx 器件的自举程序的相关信息。</p> <p>在 第 3.1 节: 自举程序激活 中增加了 BOOT1 位的情况。</p> <p>在 表 3: 嵌入式自举程序 中更新了互连型、大容量超低功耗 STM32F2xx 和 STM32F4xx 器件。</p> <p>在 表 7: STM32F105xx 和 STM32F107xx 自举程序版本 中增加了自举程序版本 V2.2。</p> <p>在 第 5.4.1 节: 如何识别 STM32F105xx/107xx 自举程序版本 中增加了自举程序 V2.2。</p> <p>在 表 14: 系统存储器自举模式下 STM32L1xxxx 大容量器件的配置 下增加了 DFU 接口的相关注释。在 表 15: STM32L1xxxx 大容量器件自举程序版本 中增加了 V4.2 自举程序已知限制, 更新了说明并增加了 V4.5 自举程序。</p> <p>在 表 19: 系统存储器自举模式下 STM32F2xxxx 的配置 下增加了 DFU 接口的相关注释。在 表 21: STM32F2xxxx 自举程序 V3.x 版本 中增加了 V3.2 自举程序已知限制并增加了 V3.3 自举程序。在 表 22: 系统存储器自举模式下 STM32F40xxx/41xxx 的配置 中更新了 STM32F2xx 和 STM32F4xx 系统存储器结束地址。</p> <p>在 表 22: 系统存储器自举模式下 STM32F40xxx/41xxx 的配置 下增加了 DFU 接口的相关注释。在 表 24: STM32F40xxx/41xxx 自举程序版本 中增加了 V3.0 自举程序已知限制并增加了 V3.1 自举程序。</p> <p>在 表 26: STM32F051xx 自举程序版本 中增加了自举程序 V2.1 已知限制。</p> <p>在 表 42: 器件相关的自举程序参数 中更新了 STM32F051x6/x8 系统存储器结束地址。</p> <p>增加了 表 48: 大容量超低功耗器件的 USART 自举程序时序 和 表 51: STM32F051xx 器件的 USART 自举程序时序。</p> <p>增加了 表 60: 大容量超低功耗器件的 USB 最短时序。</p>

表 69. 文档版本历史 (续)

日期	版本	变更
2013 年 01 月 24 日	15	<p>更新了整个文档中的通用产品名 (请参见 词汇表)。</p> <p>新增了以下各节:</p> <ul style="list-style-type: none"> – 第 12 节: STM32L151xx 和 STM32L152xx 中容量增强型超低功耗器件自举程序。 – 第 13 节: STM32F050x4 和 STM32F050x6 器件自举程序。 – 第 14 节: STM32F372xx 和 STM32F373xx 器件自举程序。 – 第 15 节: STM32F302xx 和 STM32F303xx 器件自举程序。 – 第 16 节: STM32F383xx 器件自举程序。 – 第 17 节: STM32F313xx 器件自举程序。 – 第 18 节: STM32F427xx 和 STM32F437xx 器件自举程序。 – 第 20.3 节: I2C 自举程序时序特性。 <p>更新了 第 1 节: 相关文档 和 第 2 节: 词汇表。</p> <p>增加了 表 52 到 表 58 (USART 自举程序时序)。</p> <p>替换了 图 1 到 图 9, 以及图 12、13 和 19。</p> <p>修改了表 3、4、9、11、16、21、22、24 到 27、29、31、33、35、37、39 和 42。</p> <p>第 3.3 节: 自举程序标识 在中删除了 “X = 6: 使用一个 USART”。</p> <p>在 第 11.1 节: 自举程序配置 中将地址 0x1FFFF 8002 替换为地址 0x1FFF F802。</p> <p>在 第 25 页的注:、第 6.3 节: 自举程序硬件要求 和 第 8.3 节: 自举程序硬件要求 中修改了与自举程序代码执行相关的步骤。</p>
2013 年 02 月 06 日	16	<p>在整个文档中增加了与 I²C 相关的信息。</p> <p>简化了 表 1: 适用的产品 和 第 1 节: 相关文档。</p> <p>修改了 表 3: 嵌入式自举程序, 如下所示:</p> <ul style="list-style-type: none"> – 将 “V6.0” 替换为 “V1.0” – 在 STM32F31xx 行中将 “0x1FFFF7A6” 替换为 “0x1FFFF796” – 在 STM32F051xx 行中将 “0x1FFF7FA6” 替换为 “0x1FFFF7A6” <p>更新了图 1、4 和 5。</p> <p>在 词汇表 中增加了 注:, 在 第 3.1 节: 自举程序激活 中增加了 注:。../...</p>
2013 年 02 月 06 日	16	<p>(续)</p> <p>替换内容:</p> <ul style="list-style-type: none"> – 在表 16、17、19、20、22、23、39 和 40 中将 “1.62 V” 替换为 “1.8 V”。 – 在表 35 的行 RAM 中将 “5 KB” 替换为 “4 KB”。 – 在表 42 的行 F3 中将 “127 页 (每页 2 KB)” 替换为 “4 KB (2 页, 每页 2 KB)”。 – 在 第 3.3 节: 自举程序标识 中将 “将自举程序 ID 编程在器件存储器的最后两个字节中” 替换为 “将自举程序 ID 编程在器件存储器的最后一个字节地址 - 1 对应的单元中”。 – 在 第 9 节: STM32F205/215xx 和 STM32F207/217xx 自举程序 中将 “STM32F2xxxx 器件版本 Y” 替换为 “STM32F2xxxx 器件版本 X 和 Y”。 – 在表 11、14 和 26 中将 “电压范围 2” 替换为 “电压范围 1”。

请仔细阅读：

中文翻译仅为方便阅读之目的。该翻译也许不是对本文档最新版本的翻译，如有任何不同，以最新版本的英文原版文档为准。

本文档中信息的提供仅与ST产品有关。意法半导体公司及其子公司（“ST”）保留随时对本文档及本文所述产品与服务进行变更、更正、修改或改进的权利，恕不另行通知。

所有ST产品均根据ST的销售条款出售。

买方自行负责对本文所述ST产品和服务的选择和使用，ST概不承担与选择或使用本文所述ST产品和服务相关的任何责任。

无论之前是否有任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为ST授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在ST的销售条款中另有说明，否则，ST对ST产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

意法半导体的产品不得应用于武器。此外，意法半导体产品也不是为下列用途而设计并不得应用于下列用途：（A）对安全性有特别要求的应用，例如，生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）汽车应用或汽车环境，且/或（D）航天应用或航天环境。如果意法半导体产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向意法半导体发出了书面通知，采购商仍将独自承担因此而导致的任何风险，意法半导体的产品设计规格明确指定的汽车、汽车安全或医疗工业领域专用产品除外。根据相关政府主管部门的规定，ESCC、QML或JAN正式认证产品适用于航天应用。

经销的ST产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致ST针对本文所述ST产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大ST的任何责任。

ST和ST徽标是ST在各个国家或地区的商标或注册商标。

本文档中的信息取代之前提供的所有信息。

ST徽标是意法半导体公司的注册商标。其他所有名称是其各自所有者的财产。

© 2014 STMicroelectronics 保留所有权利

意法半导体集团公司

澳大利亚 - 比利时 - 巴西 - 加拿大 - 中国 - 捷克共和国 - 芬兰 - 法国 - 德国 - 中国香港 - 印度 - 以色列 - 意大利 - 日本 - 马来西亚 - 马耳他 - 摩洛哥 - 菲律宾 - 新加坡 - 西班牙 - 瑞典 - 瑞士 - 英国 - 美国

www.st.com

