

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。在芯片生产期间由 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. 适用产品

类型	料号或产品系列
微控制器	STM32L0 系列: STM32L031xx、STM32L041xx、STM32L051xx、 STM32L052xx、STM32L053xx、STM32L062xx、 STM32L063xx、STM32L07xxx、STM32L08xxx
	STM32L1 系列。
	STM32L4 系列: STM32L476xx、STM32L486xx
	STM32F0 系列: STM32F03xxx、STM32F04xxx、STM32F05xxx、 STM32F07xxx、STM32F098xx
	STM32F1 系列。
	STM32F2 系列。
	STM32F3 系列: STM32F301xx、STM32F302xx、STM32F303xx、 STM32F318xx、STM32F328xx、STM32F334xx、 STM32F358xx、STM32F373xx、STM32F378xx、 STM32F398xx
	STM32F4 系列: STM32F401xx、STM32F405xx、STM32F407xx、 STM32F410xx、STM32F411xx、STM32F412xx、 STM32F415xx、STM32F417xx、STM32F427xx、 STM32F429xx、STM32F437xx、STM32F439xx、 STM32F446xx、STM32F469xx、STM32F479xx
	STM32F7 系列: STM32F74xxx、STM32F75xxx

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 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（参见[第 1 节：相关文档](#)）。

目录

1	相关文档	15
2	词汇表	15
3	通用自举程序描述	19
3.1	自举程序激活	19
3.2	自举程序标识	20
3.3	硬件连接要求	25
3.4	自举程序存储器管理	27
4	STM32F03xx4/6 器件自举程序	29
4.1	自举程序配置	29
4.2	自举程序选择	30
4.3	自举程序版本	30
5	STM32F030xC 器件自举程序	31
5.1	自举程序配置	31
5.2	自举程序选择	32
5.3	自举程序版本	32
6	STM32F05xxx 和 STM32F030x8 器件自举程序	33
6.1	自举程序配置	33
6.2	自举程序选择	34
6.3	自举程序版本	34
7	STM32F04xxx 器件自举程序	35
7.1	自举程序配置	35
7.2	自举程序选择	37
7.3	自举程序版本	37
8	STM32F070x6 器件自举程序	38
8.1	自举程序配置	38
8.2	自举程序选择	40

8.3	自举程序版本	41
9	STM32F070xB 器件自举程序	42
9.1	自举程序配置	42
9.2	自举程序选择	44
9.3	自举程序版本	45
10	STM32F071xx/72xx 器件自举程序	46
10.1	自举程序配置	46
10.2	自举程序选择	48
10.3	自举程序版本	48
11	STM32F09xxx 器件自举程序	49
11.1	自举程序配置	49
11.2	自举程序选择	50
11.3	自举程序版本	50
12	STM32F10xxx 器件自举程序	51
12.1	自举程序配置	51
12.2	自举程序选择	52
12.3	自举程序版本	52
13	STM32F105xx/107xx 器件自举程序	53
13.1	自举程序配置	53
13.2	自举程序选择	55
13.3	自举程序版本	56
13.3.1	如何识别 STM32F105xx/107xx 自举程序版本	56
13.3.2	日期代码小于 937 的 STM32F105xx/STM32F107xx 器件上 自举程序不可用	57
13.3.3	USART 自举程序 Get-Version 命令返回 0x20 而不是 0x22	58
13.3.4	插入 USB 线缆时，PA9 功耗过大 (在自举程序 V2.0 中)	58
14	STM32F10xxx 超大容量器件自举程序	59
14.1	自举程序配置	59
14.2	自举程序选择	60

14.3	自举程序版本	60
15	STM32F2xxxx 器件自举程序	61
15.1	自举程序 V2.x	61
15.1.1	自举程序配置	61
15.1.2	自举程序选择	62
15.1.3	自举程序版本	63
15.2	自举程序 V3.x	64
15.2.1	自举程序配置	64
15.2.2	自举程序选择	66
15.2.3	自举程序版本	67
16	STM32F301xx/302x4(6/8) 器件自举程序	68
16.1	自举程序配置	68
16.2	自举程序选择	70
16.3	自举程序版本	70
17	STM32F302xB(C)/303xB(C) 器件自举程序	71
17.1	自举程序配置	71
17.2	自举程序选择	73
17.3	自举程序版本	73
18	STM32F302xD(E)/303xD(E) 器件自举程序	74
18.1	自举程序配置	74
18.2	自举程序选择	76
18.3	自举程序版本	77
19	STM32F303x4(6/8)/334xx/328xx 器件自举程序	78
19.1	自举程序配置	78
19.2	自举程序选择	79
19.3	自举程序版本	79
20	STM32F318xx 器件自举程序	80
20.1	自举程序配置	80
20.2	自举程序选择	81
20.3	自举程序版本	82

21	STM32F358xx 器件自举程序	83
21.1	自举程序配置	83
21.2	自举程序选择	84
21.3	自举程序版本	84
22	STM32F373xx 器件自举程序	85
22.1	自举程序配置	85
22.2	自举程序选择	87
22.3	自举程序版本	87
23	STM32F378xx 器件自举程序	88
23.1	自举程序配置	88
23.2	自举程序选择	89
23.3	自举程序版本	89
24	STM32F398xx 器件自举程序	90
24.1	自举程序配置	90
24.2	自举程序选择	91
24.3	自举程序版本	91
25	STM32F40xxx/41xxx 器件自举程序	92
25.1	自举程序 V3.x	92
25.1.1	自举程序配置	92
25.1.2	自举程序选择	94
25.1.3	自举程序版本	95
25.2	自举程序 V9.x	96
25.2.1	自举程序配置	96
25.2.2	自举程序选择	100
25.2.3	自举程序版本	101
26	STM32F401xB(C) 器件自举程序	102
26.1	自举程序配置	102
26.2	自举程序选择	105
26.3	自举程序版本	106
27	STM32F401xD(E) 器件自举程序	107

27.1	自举程序配置	107
27.2	自举程序选择	110
27.3	自举程序版本	111
28	STM32F410xx 器件自举程序	112
28.1	自举程序配置	112
28.2	自举程序选择	115
28.3	自举程序版本	116
29	STM32F411xx 器件自举程序	117
29.1	自举程序配置	117
29.2	自举程序选择	120
29.3	自举程序版本	121
30	STM32F412xx 器件自举程序	122
30.1	自举程序配置	122
30.2	自举程序选择	126
30.3	自举程序版本	127
31	STM32F42xxx/43xxx 器件自举程序	128
31.1	自举程序 V7.x	128
31.1.1	自举程序配置	128
31.1.2	自举程序选择	130
31.1.3	自举程序版本	132
31.2	自举程序 V9.x	132
31.2.1	自举程序配置	132
31.2.2	自举程序选择	136
31.2.3	自举程序版本	138
32	STM32F446xx 器件自举程序	139
32.1	自举程序配置	139
32.2	自举程序选择	143
32.3	自举程序版本	144
33	STM32F469xx/479xx 器件自举程序	145
33.1	自举程序配置	145

33.2	自举程序选择	149
33.3	自举程序版本	151
34	STM32F74xxx/75xxx 器件自举程序	152
34.1	自举程序 V7.x	152
34.1.1	自举程序配置	152
34.1.2	自举程序选择	155
34.1.3	自举程序版本	156
34.2	自举程序 V9.x	156
34.2.1	自举程序配置	156
34.2.2	自举程序选择	160
34.2.3	自举程序版本	161
35	STM32L031xx/041xx 器件自举程序	162
35.1	自举程序配置	162
35.2	自举程序选择	164
35.3	自举程序版本	164
36	STM32L05xxx/06xxx 器件自举程序	165
36.1	自举程序配置	165
36.2	自举程序选择	167
36.3	自举程序版本	167
37	STM32L07xxx/08xxx 器件自举程序	168
37.1	自举程序 V4.x	168
37.1.1	自举程序配置	168
37.1.2	自举程序选择	170
37.1.3	自举程序版本	170
37.2	自举程序 V11.x	171
37.2.1	自举程序配置	171
37.2.2	自举程序选择	173
37.2.3	自举程序版本	174
38	STM32L1xxx6(8/B)A 器件自举程序	175
38.1	自举程序配置	175
38.2	自举程序选择	176

38.3	自举程序版本	176
39	STM32L1xxx6(8/B) 器件自举程序	177
39.1	自举程序配置	177
39.2	自举程序选择	178
39.3	自举程序版本	178
40	STM32L1xxxC 器件自举程序	179
40.1	自举程序配置	179
40.2	自举程序选择	181
40.3	自举程序版本	181
41	STM32L1xxxD 器件自举程序	182
41.1	自举程序配置	182
41.2	自举程序选择	184
41.3	自举程序版本	185
42	STM32L1xxxE 器件自举程序	186
42.1	自举程序配置	186
42.2	自举程序选择	188
42.3	自举程序版本	189
43	STM32L476xx/486xx 器件自举程序	190
43.1	自举程序 V10.x	190
43.1.1	自举程序配置	190
43.1.2	自举程序选择	193
43.1.3	自举程序版本	195
43.2	自举程序 V9.x	196
43.2.1	自举程序配置	196
43.2.2	自举程序选择	199
43.2.3	自举程序版本	201
44	器件相关的自举程序参数	202
45	自举程序时序	204
45.1	自举程序启动时序	204

45.2	USART 连接时序	207
45.3	USB 连接时序	209
45.4	I2C 连接时序	211
45.5	SPI 连接时序	213
46	版本历史	214

表格索引

表 1.	适用产品	1
表 2.	自举程序激活模式	19
表 3.	嵌入式自举程序	21
表 4.	利用自举程序进行 STM32 F2、F4 和 F7 电压范围配置	28
表 5.	Write、Read、Erase 和 Go 命令所支持的存储区	28
表 6.	系统存储器自举模式下 STM32F03xx4/6 器件的配置	29
表 7.	STM32F03xx4/6 自举程序版本	30
表 8.	系统存储器自举模式下 STM32F030xC 器件的配置	31
表 9.	STM32F030xC 自举程序版本	32
表 10.	系统存储器自举模式下 STM32F05xxx 和 STM32F030x8 器件的配置	33
表 11.	STM32F05xxx 和 STM32F030x8 器件自举程序版本	34
表 12.	系统存储器自举模式下 STM32F04xxx 器件的配置	35
表 13.	STM32F04xxx 自举程序版本	37
表 14.	系统存储器自举模式下 STM32F070x6 器件的配置	38
表 15.	STM32F070x6 自举程序版本	41
表 16.	系统存储器自举模式下 STM32F070xB 器件的配置	42
表 17.	STM32F070xB 自举程序版本	45
表 18.	系统存储器自举模式下 STM32F071xx/72xx 器件的配置	46
表 19.	STM32F071xx/72xx 自举程序版本	48
表 20.	系统存储器自举模式下 STM32F09xxx 器件的配置	49
表 21.	STM32F09xxx 自举程序版本	50
表 22.	系统存储器自举模式下 STM32F10xxx 器件的配置	51
表 23.	STM32F10xxx 自举程序版本	52
表 24.	系统存储器自举模式下 STM32F105xx/107xx 器件的配置	53
表 25.	STM32F105xx/107xx 自举程序版本	56
表 26.	系统存储器自举模式下 STM32F10xxx 超大容量器件的配置	59
表 27.	STM32F10xxx 超大容量自举程序版本	60
表 28.	系统存储器自举模式下 STM32F2xxxx 器件的配置	61
表 29.	STM32F2xxxx 自举程序 V2.x 版本	63
表 30.	系统存储器自举模式下 STM32F2xxxx 器件的配置	64
表 31.	STM32F2xxxx 自举程序 V3.x 版本	67
表 32.	系统存储器自举模式下 STM32F301xx/302x4(6/8) 器件的配置	68
表 33.	STM32F301xx/302x4(6/8) 自举程序版本	70
表 34.	系统存储器自举模式下 STM32F302xB(C)/303xB(C) 器件的配置	71
表 35.	STM32F302xB(C)/303xB(C) 自举程序版本	73
表 36.	系统存储器自举模式下 STM32F302xD(E)/303xD(E) 器件的配置	74
表 37.	STM32F302xD(E)/303xD(E) 自举程序版本	77
表 38.	系统存储器自举模式下 STM32F303x4(6/8)/334xx/328xx 器件的配置	78
表 39.	STM32F303x4(6/8)/334xx/328xx 自举程序版本	79
表 40.	系统存储器自举模式下 STM32F318xx 器件的配置	80
表 41.	STM32F318xx 自举程序版本	82
表 42.	系统存储器自举模式下 STM32F358xx 器件的配置	83
表 43.	STM32F358xx 自举程序版本	84
表 44.	系统存储器自举模式下 STM32F373xx 器件的配置	85
表 45.	STM32F373xx 自举程序版本	87
表 46.	系统存储器自举模式下 STM32F378xx 器件的配置	88
表 47.	STM32F378xx 自举程序版本	89
表 48.	系统存储器自举模式下 STM32F398xx 器件的配置	90

表 49.	STM32F398xx 自举程序版本	91
表 50.	系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置	92
表 51.	STM32F40xxx/41xxx 自举程序 V3.x 版本	95
表 52.	系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置	96
表 53.	STM32F40xxx/41xxx 自举程序 V9.x 版本	101
表 54.	系统存储器自举模式下 STM32F401xB(C) 器件的配置	102
表 55.	STM32F401xB(C) 自举程序版本	106
表 56.	系统存储器自举模式下 STM32F401xD(E) 器件的配置	107
表 57.	STM32F401xD(E) 自举程序版本	111
表 58.	系统存储器自举模式下 STM32F410xx 器件的配置	112
表 59.	STM32F410xx 自举程序 V11.x 版本	116
表 60.	系统存储器自举模式下 STM32F411xx 器件的配置	117
表 61.	STM32F411xx 自举程序版本	121
表 62.	系统存储器自举模式下 STM32F412xx 器件的配置	122
表 63.	STM32F412xx 自举程序 V9.x 版本	127
表 64.	系统存储器自举模式下 STM32F42xxx/43xxx 器件的配置	128
表 65.	STM32F42xxx/43xxx 自举程序 V7.x 版本	132
表 66.	系统存储器自举模式下 STM32F42xxx/43xxx 器件的配置	133
表 67.	STM32F42xxx/43xxx 自举程序 V9.x 版本	138
表 68.	系统存储器自举模式下 STM32F446xx 器件的配置	139
表 69.	STM32F446xx 自举程序 V9.x 版本	144
表 70.	系统存储器自举模式下 STM32F469xx/479xx 器件的配置	145
表 71.	STM32F469xx/479xx 自举程序 V9.x 版本	151
表 72.	系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置	152
表 73.	STM32F74xxx/75xxx 自举程序 V7.x 版本	156
表 74.	系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置	156
表 75.	STM32F74xxx/75xxx 自举程序 V9.x 版本	161
表 76.	系统存储器自举模式下 STM32L031xx/041xx 器件的配置	162
表 77.	STM32L031xx/041xx 自举程序版本	164
表 78.	系统存储器自举模式下 STM32L05xxx/06xxx 器件的配置	165
表 79.	STM32L05xxx/06xxx 自举程序版本	167
表 80.	系统存储器自举模式下 STM32L07xxx/08xxx 器件的配置	168
表 81.	STM32L07xxx/08xxx 自举程序 V4.x 版本	170
表 82.	系统存储器自举模式下 STM32L07xxx/08xxx 器件的配置	171
表 83.	STM32L07xxx/08xxx 自举程序 V11.x 版本	174
表 84.	系统存储器自举模式下 STM32L1xxx6(8/B)A 器件的配置	175
表 85.	STM32L1xxx6(8/B)A 自举程序版本	176
表 86.	系统存储器自举模式下 STM32L1xxx6(8/B) 器件的配置	177
表 87.	STM32L1xxx6(8/B) 自举程序版本	178
表 88.	系统存储器自举模式下 STM32L1xxxC 器件的配置	179
表 89.	STM32L1xxxC 自举程序版本	181
表 90.	系统存储器自举模式下 STM32L1xxxD 器件的配置	182
表 91.	STM32L1xxxD 自举程序版本	185
表 92.	系统存储器自举模式下 STM32L1xxxE 器件的配置	186
表 93.	STM32L1xxxE 自举程序版本	189
表 94.	系统存储器自举模式下 STM32L476xx/486xx 器件的配置	190
表 95.	STM32L476xx/486xx 自举程序 V10.x 版本	195
表 96.	系统存储器自举模式下 STM32L476xx/486xx 器件的配置	196
表 97.	STM32L476xx/486xx 自举程序 V9.x 版本	201
表 98.	器件相关的自举程序参数	202
表 99.	STM32 器件的自举程序启动时序	204
表 100.	STM32 器件的 USART 自举程序最小时序	207

表 101. STM32 器件的 USB 自举程序最小时序..... 209

表 102. STM32 器件的 I2C 自举程序最小时序 211

表 103. STM32 器件的 SPI 自举程序最小时序 213

表 104. 文档版本历史 214

表 105. 中文文档版本历史..... 220



图片索引

图 1.	USART 连接	25
图 2.	USB 连接	25
图 3.	I2C 连接	26
图 4.	SPI 连接	26
图 5.	CAN 连接	27
图 6.	STM32F03xx4/6 器件的自举程序选择	30
图 7.	STM32F030xC 器件的自举程序选择	32
图 8.	STM32F05xxx 和 STM32F030x8 器件的自举程序选择	34
图 9.	STM32F04xxx 器件的自举程序选择	37
图 10.	STM32F070x6 器件的自举程序选择	40
图 11.	STM32F070xB 器件的自举程序选择	44
图 12.	STM32F071xx/72xx 器件的自举程序选择	48
图 13.	STM32F09xxx 器件的自举程序选择	50
图 14.	STM32F10xxx 器件的自举程序选择	52
图 15.	STM32F105xx/107xx 器件的自举程序选择	55
图 16.	STM32F10xxx 超大容量器件的自举程序选择	60
图 17.	STM32F2xxxx 器件的自举程序 V2.x 选择	62
图 18.	STM32F2xxxx 器件的自举程序 V3.x 选择	66
图 19.	STM32F301xx/302x4(6/8) 器件的自举程序选择	70
图 20.	STM32F302xB(C)/303xB(C) 器件的自举程序选择	73
图 21.	STM32F302xD(E)/303xD(E) 器件的自举程序选择	76
图 22.	STM32F303x4(6/8)/334xx/328xx 器件的自举程序选择	79
图 23.	STM32F318xx 器件的自举程序选择	81
图 24.	STM32F358xx 器件的自举程序选择	84
图 25.	STM32F373xx 器件的自举程序选择	87
图 26.	STM32F378xx 器件的自举程序选择	89
图 27.	STM32F398xx 器件的自举程序选择	91
图 28.	STM32F40xxx/41xxx 器件的自举程序 V3.x 选择	94
图 29.	STM32F40xxx/41xxx 器件的自举程序 V9.x 选择	100
图 30.	STM32F401xB(C) 器件的自举程序选择	105
图 31.	STM32F401xD(E) 器件的自举程序选择	110
图 32.	STM32F410xx 器件的自举程序 V11.x 选择	115
图 33.	STM32F411xx 器件的自举程序选择	120
图 34.	STM32F412xx 器件的自举程序 V9.x 选择	126
图 35.	STM32F42xxx/43xxx 自举程序 V7.x 的双存储区自举实现	130
图 36.	STM32F42xxx/43xxx 器件的自举程序 V7.x 选择	131
图 37.	STM32F42xxx/43xxx 自举程序 V9.x 的双存储区自举实现	136
图 38.	STM32F42xxx/43xxx 器件的自举程序 V9.x 选择	137
图 39.	STM32F446xx 器件的自举程序 V9.x 选择	143
图 40.	STM32F469xx/479xx 自举程序 V9.x 的双存储区自举实现	149
图 41.	STM32F469xx/479xx 器件的自举程序 V9.x 选择	150
图 42.	STM32F74xxx/75xxx 器件的自举程序 V7.x 选择	155
图 43.	STM32F74xxx/75xxx 器件的自举程序 V9.x 选择	160
图 44.	STM32L031xx/041xx 器件的自举程序选择	164
图 45.	STM32L05xxx/06xxx 器件的自举程序选择	167
图 46.	STM32L07xxx/08xxx 器件的自举程序 V4.x 选择	170
图 47.	STM32L07xxx/08xxx 器件的自举程序 V11.x 选择	173
图 48.	STM32L1xxx6(8/B)A 器件的自举程序选择	176

图 49.	STM32L1xxx6(8/B) 器件的自举程序选择	178
图 50.	STM32L1xxxC 器件的自举程序选择	181
图 51.	STM32L1xxxD 器件的自举程序选择	184
图 52.	STM32L1xxxE 器件的自举程序选择	188
图 53.	STM32L476xx/486xx 自举程序 V10.x 的双存储区自举实现	193
图 54.	STM32L476xx/486xx 器件的自举程序 V10.x 选择	194
图 55.	STM32L476xx/486xx 自举程序 V9.x 的双存储区自举实现	199
图 56.	STM32L476xx/486xx 器件的自举程序 V9.x 选择	200
图 57.	自举程序启动时序描述	204
图 58.	USART 连接时序描述	207
图 59.	USB 连接时序描述	209
图 60.	I2C 连接时序描述	211
图 61.	SPI 连接时序描述	213

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 www.st.com 提供的以下文档：

- 数据手册或产品简介
- 参考手册
- 应用笔记：
 - AN3154：STM32 自举程序中使用的 CAN 协议
 - AN3155：STM32 自举程序中使用的 USART 协议
 - AN3156：STM32 自举程序中使用的 USB DFU 协议
 - AN4221：STM32 自举程序中使用的 I2C 协议
 - AN4286：STM32 自举程序中使用的 SPI 协议

2 词汇表

F0 系列：

STM32F03xxx 指代 STM32F030x4、STM32F030x6、STM32F038x6、STM32F030xC、STM32F031x4 和 STM32F031x6 器件。

STM32F04xxx 指代 STM32F042x4 和 STM32F042x6 器件。

STM32F05xxx 和 STM32F030x8 器件指代 STM32F051x4、STM32F051x6、STM32F051x8、STM32F058x8 和 STM32F030x8 器件。

STM32F07xxx 指代 STM32F070x6、STM32F070xB、STM32F071xB STM32F072x8 和 STM32F072xB 器件。

STM32F09xxx 指代 STM32F091xx 和 STM32F098xx 器件。

F1 系列:

STM32F10xxx 指代小容量、中容量、大容量、小容量超值型、中容量超值型和超大容量超值型器件:

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。

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。

STM32F105xx/107xx 指代 STM32F105xx 和 STM32F107xx 器件。

STM32F10xxx 超大容量 指代 Flash 容量介于 768 KB 到 1 MB 之间的 STM32F101xx 和 STM32F103xx 器件。

F2 系列:

STM32F2xxxx 指代 STM32F215xx、STM32F205xx、STM32F207xx 和 STM32F217xx 器件。

F3 系列:

STM32F301xx/302x4(6/8) 指代 STM32F301x4、STM32F301x6、STM32F301x8、STM32F302x4、STM32F302x6 和 STM32F302x8 器件。

STM32F302xB(C)/303xB(C) 指代 STM32F302xB、STM32F302xC、STM32F303xB 和 STM32F303xC 器件。

STM32F302xD(E)/303xD(E) 指代 STM32F302xD、STM32F302xE、STM32F303xD 和 STM32F303xE 器件。

STM32F303x4(6/8)/334xx/328xx 指代 STM32F303x4、STM32F303x6、STM32F303x8、STM32F334x4、STM32F334x6、STM32F334x8 和 STM32F328x8 器件。

STM32F318xx 指代 STM32F318x8 器件。

STM32F358xx 指代 STM32F358xC 器件。

STM32F373xx 指代 STM32F373x8、STM32F373xB 和 STM32F373xC 器件。

STM32F378xx 指代 STM32F378xC 器件。

STM32F398xx 指代 STM32F398xE 器件。

F4 系列:

STM32F40xxx/41xxx 指代 STM32F405xx、STM32F407xx、STM32F415xx 和 STM32F417xx 器件。

STM32F401xB(C) 指代 STM32F401xB 和 STM32F401xC 器件。

STM32F401xD(E) 指代 STM32F401xD 和 STM32F401xE 器件。

STM32F410xx 指代 STM32F410x8 和 STM32F410xB 器件。

STM32F411xx 指代 STM32F411xD 和 STM32F411xE 器件。

STM32F412xx 指代 STM32F412Cx、STM32F412Rx、STM32F412Vx 和 STM32F412Zx 器件。

STM32F42xxx/43xxx 指代 STM32F427xx、STM32F429xx、STM32F437xx 和 STM32F439xx 器件。

STM32F446xx 指代 STM32F446xE 和 STM32F446xC 器件。

STM32F469xx/479xx 指代 STM32F469xE、STM32F469xG、STM32F469xI、STM32F479xG 和 STM32F479xI 器件。

F7 系列:

STM32F74xxx/75xxx 指代 STM32F745xx、STM32F746xx 和 STM32F756xx 器件。

L0 系列:

STM32L031xx/041xx 指代 STM32L031xx 和 STM32L041xx 器件。

STM32L05xxx/06xxx 指代 STM32L051xx、STM32L052xx、STM32L053xx、STM32L062xx 和 STM32L063xx 超低功耗器件。

STM32L07xxx/08xxx 指代 STM32L071xx、STM32L072xx、STM32L073xx、STM32L081xx、STM32L082xx 和 STM32L083xx 器件。

L1 系列:

STM32L1xxx6(8/B) 指代 STM32L1xxV6T6、STM32L1xxV6H6、STM32L1xxR6T6、STM32L1xxR6H6、STM32L1xxC6T6、STM32L1xxC6H6、STM32L1xxV8T6、STM32L1xxV8H6、STM32L1xxR8T6、STM32L1xxR8H6、STM32L1xxC8T6、STM32L1xxC8H6、STM32L1xxVBT6、STM32L1xxVBH6、STM32L1xxRBT6、STM32L1xxRBH6、STM32L1xxCBT6 和 STM32L1xxCBH6 超低功耗器件。

STM32L1xxx6(8/B)A 指代 STM32L1xxV6T6-A、STM32L1xxV6H6-A、STM32L1xxR6T6-A、STM32L1xxR6H6-A、STM32L1xxC6T6-A、STM32L1xxC6H6-A、STM32L1xxV8T6-A、STM32L1xxV8H6-A、STM32L1xxR8T6-A、STM32L1xxR8H6-A、STM32L1xxC8T6-A、STM32L1xxC8H6-A、STM32L1xxVBT6-A、STM32L1xxVBH6-A、STM32L1xxRBT6-A、STM32L1xxRBH6-A、STM32L1xxCBT6-A 和 STM32L1xxCBH6-A 超低功耗器件。

STM32L1xxxC 指代 STM32L1xxVCT6、STM32L1xxVCH6、STM32L1xxRCT6、STM32L1xxUCY6、STM32L1xxCCT6 和 STM32L1xxCCU6 超低功耗器件。

STM32L1xxxD 指代 STM32L1xxZDT6、STM32L1xxQDH6、STM32L1xxVDT6、STM32L1xxRDY6、STM32L1xxRDT6、STM32L1xxZCT6、STM32L1xxQCH6、STM32L1xxRCY6、STM32L1xxVCT6-A 和 STM32L1xxRCT6-A 超低功耗器件。

STM32L1xxxE 指代 STM32L1xxZET6、STM32L1xxQEH6、STM32L1xxVET6、STM32L1xxVEY6 和 STM32L1xxRET6 超低功耗器件。

L4 系列:

STM32L476xx/486xx 指代 STM32L476xE、STM32L476xG 和 STM32L486xG 器件。

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。
BL_SPI_Loop 指 SPI 自举程序执行循环。

3 通用自举程序描述

3.1 自举程序激活

自举程序通过应用 表 2 所述模式之一激活。

如果 Boot From Bank2 选项激活（针对支持此功能的产品），自举程序执行 Dual Boot 机制，如图 “STM32xxxx 的双存储区自举实现” 中所示，其中 STM32xxxx 是相关的 STM32 产品。

否则，执行自举程序选择协议，如图 “STM32xxxx 的自举程序 VY.x 选择” 所示，其中 STM32xxxx 是相关的 STM32 产品。

当读出保护 Level2 激活时，任何情况下 STM32 都不在系统存储器上自举，自举程序不能执行（除非从 Flash 用户代码跳转到它）。

表 2. 自举程序激活模式

模式	条件
Pattern1	Boot0(pin) = 1 和 Boot1(pin) = 0
Pattern2	Boot0(pin) = 1 和 nBoot1(bit) = 1
Pattern3	Boot0(pin) = 1, Boot1(pin) = 0 和 BFB2(bit) = 1
	Boot0(pin) = 0, BFB2(bit) = 0 且两个存储区均不包含有效代码
	Boot0(pin) = 1, Boot1(pin) = 0, BFB2(bit) = 0 两个存储区均不包含有效代码
Pattern4	Boot0(pin) = 1, Boot1(pin) = 0 和 BFB2(bit) = 1
	Boot0(pin) = 0, BFB2(bit) = 0 且两个存储区均不包含有效代码
	Boot0(pin) = 1, Boot1(pin) = 0 和 BFB2(bit) = 0
Pattern5	Boot0(pin) = 1, Boot1(pin) = 0 和 BFB2(bit) = 0
	Boot0(pin) = 0, BFB2(bit) = 1 且两个存储区均不包含有效代码
	Boot0(pin) = 1, Boot1(pin) = 0 和 BFB2 (bit) = 1
Pattern6	Boot0(pin) = 1, nBoot1(bit) = 1 和 nBoot0_SW(bit) = 1
	nBoot0(bit) = 0, nBoot1(bit) = 1 和 nBoot0_SW(bit) = 0
	Boot0(pin) = 0, nBoot0_SW (bit) = 1 且主 Flash 为空
Pattern7	Boot0(pin) = 1, nBoot1(bit) = 1 和 BFB2(bit) = 0
	Boot0(pin) = 0, BFB2(bit) = 1 且两个存储区均不包含有效代码
	Boot0(pin) = 1, nBoot1(bit) = 1 和 BFB2(bit) = 1
Pattern8	Boot(pin) = 0 和 BOOT_ADD0(optionbyte) = 0x0040
	Boot(pin) = 1 和 BOOT_ADD1(optionbyte) = 0x0040

除了上述模式之外，用户可通过从用户代码跳转到系统存储器来执行自举程序。跳转到自举程序之前，用户必须：

- 禁用所有外设时钟
- 禁用所用的 PLL
- 禁用中断
- 清空挂起的中断

通过离开自举程序激活条件并生成硬件复位，或使用 Go 命令来执行用户代码，可退出系统存储器自举模式。

注： 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（因为已将预分频器设置为最大值）。

3.2 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：使用一个 USART
 - X = 2：使用两个 USART
 - X = 3：使用 USART、CAN 和 DFU
 - X = 4：使用 USART 和 DFU
 - X = 5：使用 USART 和 I²C
 - X = 6：使用 I²C
 - X = 7：使用 USART、CAN、DFU 和 I²C
 - X = 8：使用 I²C 和 SPI
 - X = 9：使用 USART、CAN、DFU、I²C 和 SPI
 - X = 10：使用 USART、DFU 和 I²C
 - X = 11：使用 USART、I²C 和 SPI
 - X = 12：使用 USART 和 SPI
 - X = 13：使用 USART、DFU、I²C 和 SPI
 - **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	
F0	STM32F05xxx 和 STM32F030x8 器件		USART1/USART2	0x21	0x1FFFF7A6	USART (V3.1)
	STM32F030xx4/6		USART1	0x10	0x1FFFF7A6	USART (V3.1)
	STM32F030xC		USART1/I2C1	0x52	0x1FFFF6A6	USART (V3.1) I2C1(V1.0)
	STM32F04xxx		USART1/USART2/ I2C1/ DFU (USB 设备 FS)	0xA0	0x1FFFF6A6	USART (V3.1) DFU (V2.2) I2C (V1.0)
	STM32F071xx/72xx		USART1/USART2/ I2C1/ DFU (USB 设备 FS)	0xA1	0x1FFFF6A6	USART (V3.1) DFU (V2.2) I2C (V1.0)
	STM32F070x6		USART1/USART2/ DFU (USB 设备 FS) / I2C1	0xA2	0x1FFFF6A6	USART (V3.1) DFU (V2.2) I2C (V1.0)
	STM32F070xB		USART1/USART2/ DFU (USB 设备 FS) / I2C1	0xA2	0x1FFFF6A6	USART (V3.1) DFU (V2.2) I2C (V1.0)
	STM32F09xxx		USART1/USART2/ I2C1	0x50	0x1FFFF6A6	USART (V3.1) I2C (V1.0)
F1	STM32F10xxx	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)
		中容量超值型	USART1	0x10	0x1FFFF7D6	USART (V2.2)
		大容量超值型	USART1	0x10	0x1FFFF7D6	USART (V2.2)
	STM32F105xx/107xx		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)
	STM32F10xxx 超大容量		USART1/USART2 (重映射)	0x21	0x1FFFF7D6	USART (V3.0)
F2	STM32F2xxxx		USART1/USART3	0x20	0x1FFF77DE	USART (V3.0)
			USART1/USART3/ CAN2/ DFU (USB 设备 FS)	0x33	0x1FFF77DE	USART (V3.1) CAN (V2.0) 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	
F3	STM32F373xx	USART1/USART2/ DFU (USB 设备 FS)	0x41	0x1FFFF7A6	USART (V3.1) DFU (V2.2)
	STM32F378xx	USART1/USART2/ I2C1	0x50	0x1FFFF7A6	USART (V3.1) I2C (V1.0)
	STM32F302xB(C)/303xB(C)	USART1/USART2/ DFU (USB 设备 FS)	0x41	0x1FFFF796	USART (V3.1) DFU (V2.2)
	STM32F358xx	USART1/USART2/ I2C1	0x50	0x1FFFF796	USART (V3.1) I2C (V1.0)
	STM32F301xx/302x4(6/8)	USART1/USART2/ DFU (USB 设备 FS)	0x40	0x1FFFF796	USART (V3.1) DFU (V2.2)
	STM32F318xx	USART1/USART2/ I2C1/ I2C3	0x50	0x1FFFF796	USART (V3.1) I2C (V1.0)
	STM32F302xD(E)/303xD(E)	USART1/USART2/ DFU (USB 设备 FS)	0x40	0x1FFFF796	USART (V3.1) DFU (V2.2)
	STM32F303x4(6/8)/334xx/328xx	USART1/USART2/ I2C1	0x50	0x1FFFF796	USART (V3.1) I2C (V1.0)
	STM32F398xx	USART1/USART2/ I2C1/I2C3	0x50	0x1FFFF796	USART (V3.1) I2C (V1.0)
F4	STM32F40xxx/41xxx	USART1/USART3/ CAN2/ DFU (USB 设备 FS)	0x31	0x1FFF77DE	USART (V3.1) CAN (V2.0) DFU (V2.2)
		USART1/USART3/ CAN2 / DFU (USB 设备 FS) /I2C1/I2C2/I2C3/SPI1/SPI2	0x90	0x1FFF77DE	USART (V3.1) CAN (V2.0) DFU (V2.2) SPI(V1.1) I2C (V1.0)
	STM32F42xxx/43xxx	USART1/USART3/ CAN2 /DFU (USB 设备 FS) / I2C1/I2C2/I2C3	0x70	0x1FFF76DE	USART (V3.1) CAN (V2.0) DFU (V2.2) I2C (V1.0)
		USART1/USART3/ CAN2 / DFU (USB 设备 FS) / I2C1/I2C2/I2C3/SPI1/ SPI2/ SPI4	0x90	0x1FFF76DE	USART (V3.1) CAN (V2.0) DFU (V2.2) SPI(V1.1) I2C (V1.0)
	STM32F401xB(C)	USART1/USART2/ DFU (USB 设备 FS) / I2C1/I2C2/I2C3/ SPI1/SPI2/ SPI3	0xD1	0x1FFF76DE	USART (V3.1) DFU (V2.2) SPI(V1.1) I2C (V1.0)

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	
F4	STM32F401xD(E)	USART1/USART2/ DFU (USB 设备 FS) / I2C1/I2C2/I2C3/ SPI1/SPI2/ SPI3	0xD1	0x1FFF76DE	USART (V3.1) DFU (V2.2) SPI(V1.1) I2C (V1.1)
	STM32F410xx	USART1/USART2/ I2C1/I2C2/I2C4 SPI1/SPI2	0xB1	0x1FFF76DE	USART (V3.1) I2C (V1.2) SPI (V1.1)
	STM32F411xx	USART1/USART2/ DFU (USB 设备 FS) / I2C1/I2C2/I2C3/ SPI1/SPI2/ SPI3	0xD0	0x1FFF76DE	USART (V3.1) DFU (V2.2) SPI(V1.1) I2C (V1.1)
	STM32F412xx	USART1/USART2/ USART3/CAN2/ DFU (USB 设备 FS) / I2C1/I2C2/I2C3/I2C4/ SPI1/SPI3/SPI4	0x90	0x1FFF76DE	USART (V3.1) CAN (V2.0) DFU (V2.2) SPI (V1.1) I2C (V1.2)
	STM32F446xx	USART1/USART3/ CAN2 / DFU (USB 设 备 FS) / I2C1/I2C2/I2C3/SPI1/ SPI2/ SPI4	0x90	0x1FFF76DE	USART (V3.1) CAN (V2.0) DFU (V2.2) SPI(V1.1) I2C (V1.2)
	STM32F469xx/479xx	USART1/USART3/ I2C1/I2C2/I2C3/ CAN2/ DFU (USB 设备 FS) / SPI1/ SPI2/ SPI4	0x90	0x1FFF76DE	USART (V3.1) I2C (V1.2) CAN (V2.0) DFU (V2.2) SPI (V1.1)
F7	STM32F74xxx/75xxx	USART1/USART3/ I2C1/I2C2/I2C3/ CAN2/ DFU (USB 设备 FS)	0x70	0x1FF0EDBE	USART (V3.1) I2C (V1.2) CAN (V2.0) DFU (V2.2)
		USART1/USART3/ I2C1/I2C2/I2C3/ CAN2/ DFU (USB 设备 FS) / SPI1/SPI2/SPI4	0x90	0x1FF0EDBE	USART (V3.1) I2C (V1.2) CAN (V2.0) DFU (V2.2) SPI (V1.2)

表 3. 嵌入式自举程序（续）

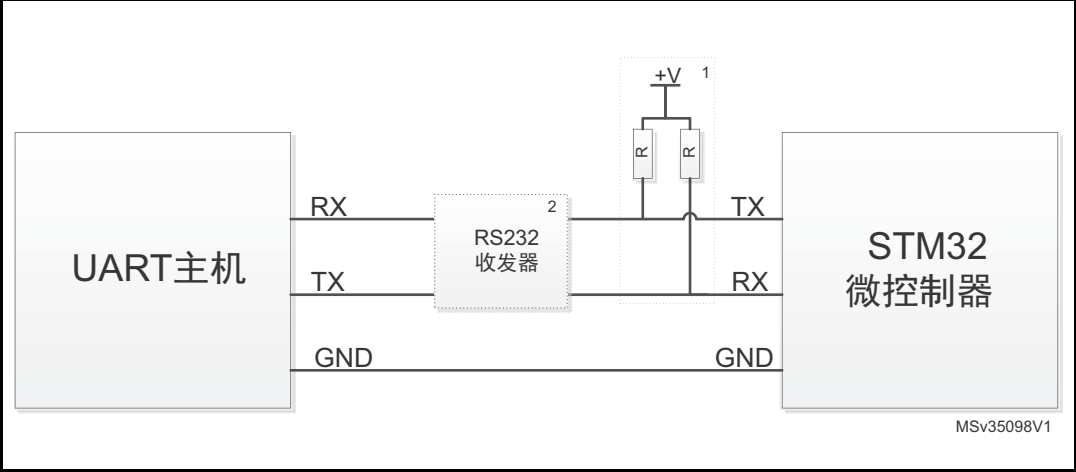
STM32 系列	器件	支持的串行外设	自举程序 ID		自举程序（协议）版本
			ID	存储单元	
L0	STM32L031xx/041xx	USART2/SPI1	0xC0	0x1FF00FFE	USART (V3.1) SPI (V1.1)
	STM32L05xxx/06xxx	USART1/USART2/SPI1/ SPI2	0xC0	0x1FF00FFE	USART (V3.1) SPI (V1.1)
	STM32L07xxx/08xxx	USART1/USART2/ DFU（USB 设备 FS）	0x40	0x1FF01FFE	USART (V3.1) DFU (V2.2)
		USART1/USART2/ SPI1/SPI2/	0xB0	0x1FF01FFE	USART (V3.1) SPI (V1.1)
L1	STM32L1xxx6(8/B)	USART1/USART2	0x20	0x1FF00FFE	USART (V3.0)
	STM32L1xxx6(8/B)A	USART1/USART2	0x20	0x1FF00FFE	USART (V3.1)
	STM32L1xxxC	USART1/USART2/ DFU （USB 设备 FS）	0x40	0x1FF01FFE	USART (V3.1) DFU (V2.2)
	STM32L1xxxD	USART1/USART2/ DFU （USB 设备 FS）	0x45	0x1FF01FFE	USART (V3.1) DFU (V2.2)
	STM32L1xxxE	USART1/USART2/ DFU （USB 设备 FS）	0x40	0x1FF01FFE	USART (V3.1) DFU (V2.2)
L4	STM32L476xx/486xx	USART1/USART2/ USART3/ I2C1/I2C2/I2C3/ DFU（USB 设备 FS）	0xA3	0x1FFF6FFE	USART (V3.1) I2C (V1.2) DFU (V2.2)
		USART1/USART2/ USART3/ I2C/I2C2/I2C3/ SPI1/SPI2/CAN1/ DFU（USB 设备 FS）	0x90	0x1FFF6FFE	USART (V3.1) I2C (V1.2) SPI (V1.1) CAN(V2.0) DFU(V2.2)

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。

3.3 硬件连接要求

要使用 USART 自举程序，主机必须通过串口电缆连接到所需 USARTx 接口的 (RX) 和 (TX) 引脚。

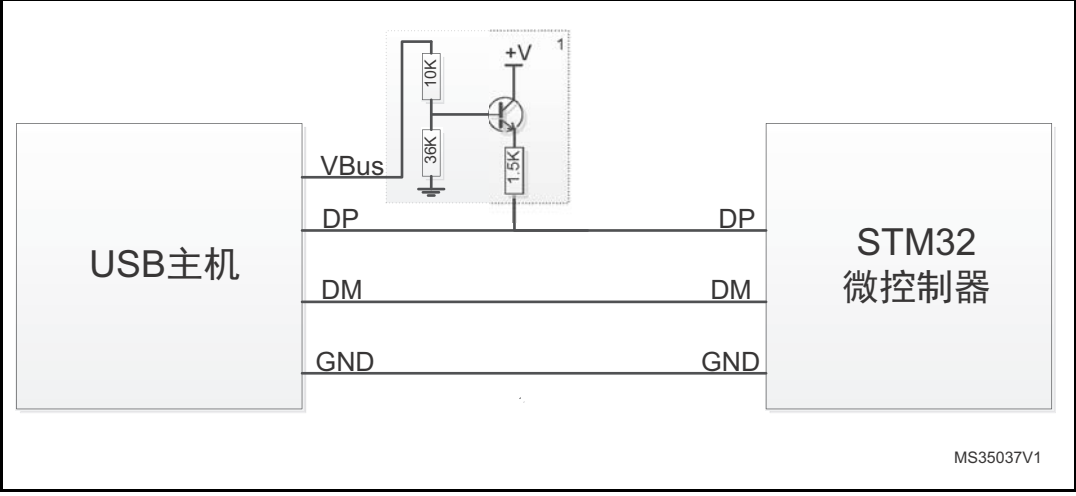
图 1.USART 连接



- 1. 如果上拉电阻未连接到主机端，则应添加一个上拉电阻。
- 2. 必须连接 RS232 收发器，来适配 STM32 器件和主机之间的电平（3.3V - 12V）。

注：+V 典型值为 3.3 V，R 典型值为 100KOhm。该值取决于应用和所用硬件。
要使用 DFU，应将微控制器的 USB 接口和 USB 主机（即 PC）连接起来。

图 2.USB 连接

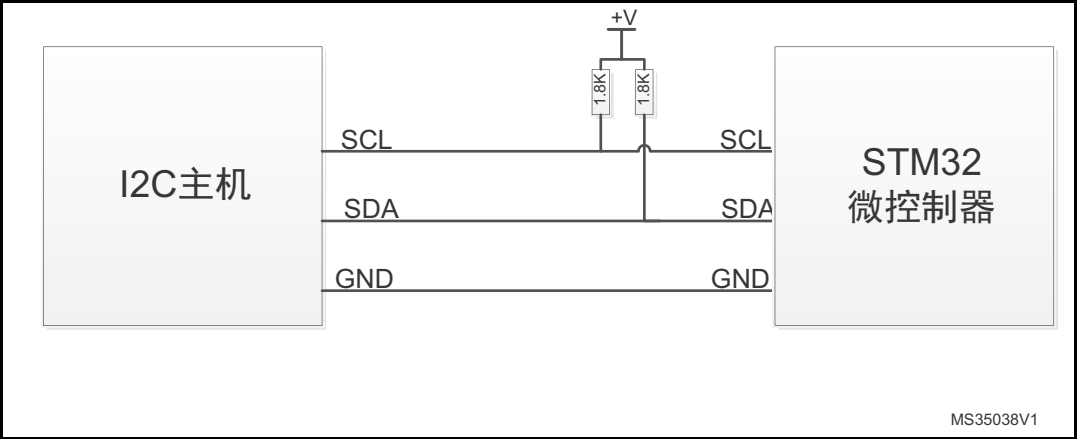


- 1. 需要时，此附加电路可利用 VBus 将上拉电阻连接到 (DP) 引脚。请参考产品章节（描述了系统存储器自举模式中 STM32 配置的表格）来了解是否需要将外部上拉电阻连接到 (DP) 引脚。

注：+V 典型值为 3.3 V。该值取决于应用和所用硬件。

要使用 I2C 自举程序，可利用数据（SDA）和时钟（SCL）引脚，将主机（主设备）和所需 I2Cx 接口（从设备）连接起来。（SDA）线和（SCL）线都须连接一个 1800 欧姆的上拉电阻。

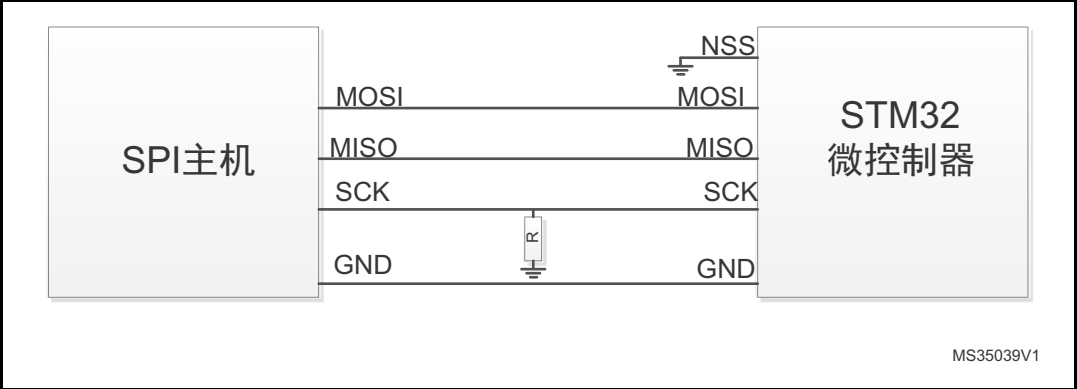
图 3.I2C 连接



注：+V 典型值为 3.3 V。该值取决于应用和所用硬件。

要使用 SPI 自举程序，可利用 (MOSI)、(MISO) 和 (SCK) 引脚，将主机（主设备）和所需 SPIx 接口（从设备）连接起来。(NSS) 引脚必须连接到 (GND)。应连接一个下拉电阻到 (SCK) 线。

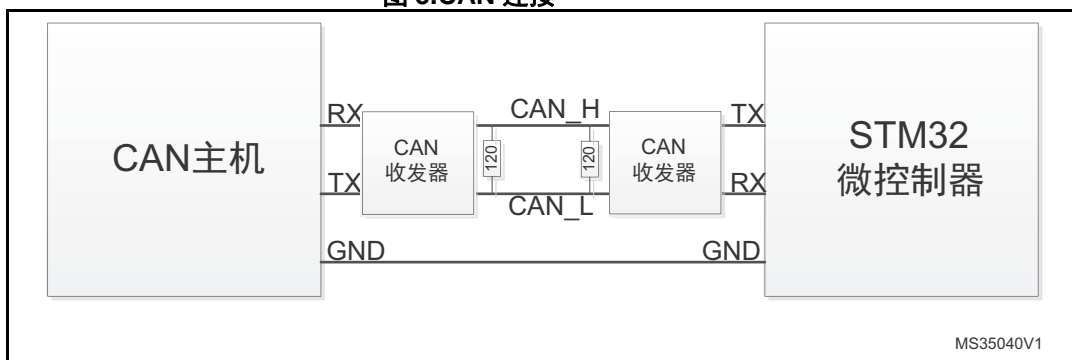
图 4.SPI 连接



注：R 典型值为 10KOhm。该值取决于应用和所用硬件。

要使用 CAN 接口，主机必须通过 CAN 收发器和串口电缆连接到所需 CANx 接口的 (RX) 和 (TX) 引脚。应添加一个 120 Ohm 的电阻作为终端电阻。

图 5.CAN 连接



注：当自举程序固件支持 DFU 时，强制要求在其他接口选择阶段中，无 USB 主机连接到 USB 外设。选择阶段之后，用户可插入一个 USB 线缆而不影响所选自举程序执行，产生系统复位的命令除外。

在自举程序启动时（检测阶段），建议将不用的自举程序接口（USART_RX、SPI_MOSI、CAN_RX 和 USB D+/D- 线，如果存在的话）的 RX 引脚保持为已知的（低或高）电平。在检测阶段将这些引脚浮空，可能会激活不用的接口。

3.4 自举程序存储器管理

因此，所有使用自举程序命令的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）。

一些产品嵌入了具有特殊功能的自举程序：

- 一些产品不支持批量擦除操作。要使用自举程序执行批量擦除操作，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。
- 除了标准存储器（内部 Flash、内部 SRAM、选择字节和系统存储器）之外，STM32 L1 和 L0 系列产品的自举程序固件还支持数据存储器。起始地址和该区域的大小取决于产品，请参考产品参考手册来获取更多信息。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。
- 除了标准存储器（内部 Flash、内部 SRAM、选择字节和系统存储器）之外，STM32 F2、F4、F7 和 L4 系列产品的自举程序固件还支持 OTP 存储器。起始地址和该区域的大小取决于产品，请参考产品参考手册来获取更多信息。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，需确保相关保护位为非零值。

- 对于STM32 F2、F4和F7系列产品，内部flash写操作格式取决于电压范围。默认支持按照字节格式（不支持半字、字和双字操作）执行写操作。要提高写操作的速度，用户应当使用适当的电压范围，可支持以半字、字或双字执行写操作，并利用自举程序软件通过虚拟内存单元来实时更新该配置。该存储单元并非物理单元，但可根据所使用的协议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，如下表所述。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 4. 利用自举程序进行 STM32 F2、F4 和 F7 电压范围配置

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，请参见产品参考手册）。 其它：所有其它值都不适用，否则将收到 NACK 应答。
0xFFFF0001	1 字节	Reserved 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	Reserved 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	Reserved 0xFF：默认值。 其它：所有其它值都不适用，否则将收到 NACK 应答。

下表中根据自举程序命令列出了其有效的存储区。

表 5. Write、Read、Erase 和 Go 命令所支持的存储区

存储区	Write 命令	Read 指令	擦除命令	Go 指令
Flash	支持	支持	支持	支持
RAM	支持	支持	不支持	支持
系统存储器	不支持	支持	不支持	不支持
数据存储器	支持	支持	不支持	不支持
OTP 存储器	支持	支持	不支持	不支持



4 STM32F03xx4/6 器件自举程序

4.1 自举程序配置

STM32F03xx4/6 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 6. 系统存储器自举模式下 STM32F03xx4/6 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	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	启用	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。
USART1 自举程序 (PA14/PA15)	USART1	启用	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	PA15 引脚：USART1 处于接收模式。
	USART1_TX 引脚	输出	PA14 引脚：USART1 处于发送模式。
USART1 自举程序	SysTick 定时器	启用	用于自动检测主机串口波特率。

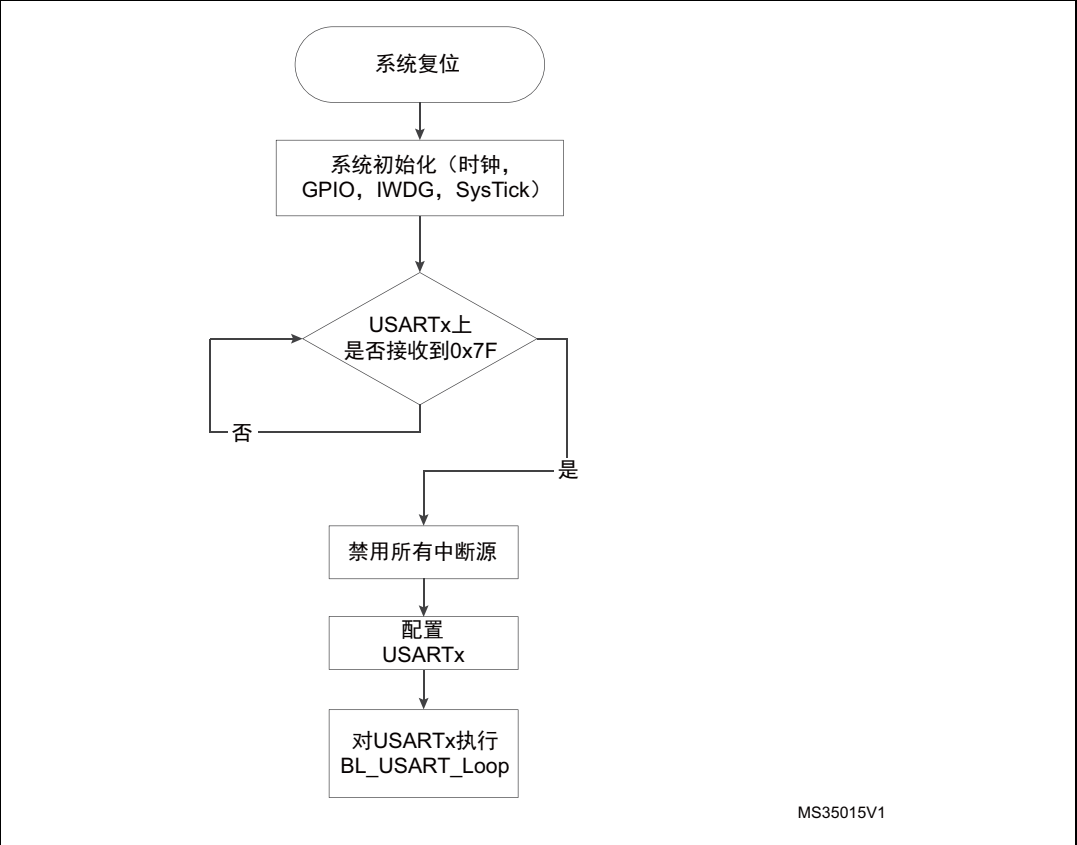
系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

注： STM32F03xx4/6 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) 已由自举程序使用。

4.2 自举程序选择

下图显示了自举程序选择机制。

图 6.STM32F03xx4/6 器件的自举程序选择



4.3 自举程序版本

下表列出了 STM32F03xx4/6 器件自举程序版本。

表 7.STM32F03xx4/6 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。

5 STM32F030xC 器件自举程序

5.1 自举程序配置

STM32F030xC 自举程序通过应用模式 6 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 8. 系统存储器自举模式下 STM32F030xC 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	系统时钟配置为 48 MHz，使用 HSI 8 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间包含自举程序固件。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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	PA15 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000001x (其中 x = 0 对应写，x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

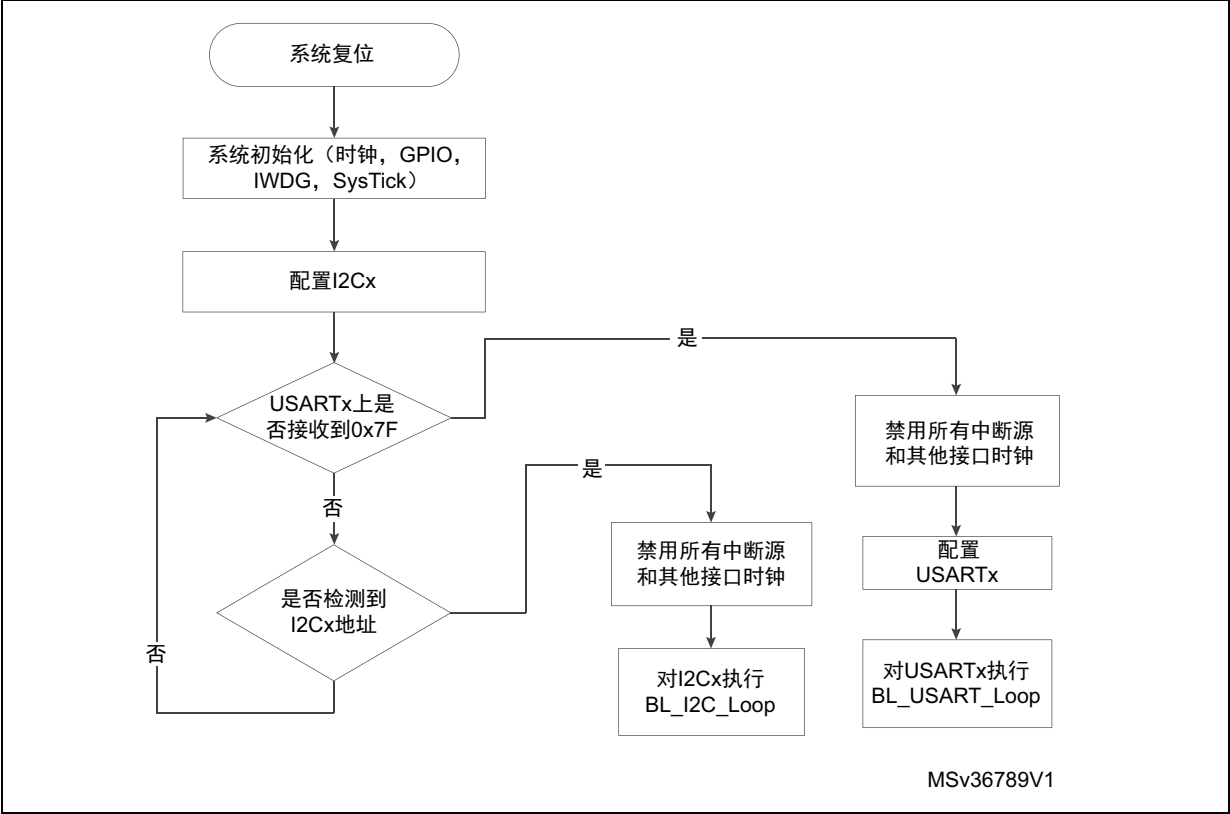
注： STM32F030xC 器件在自举程序模式下使用 USART2 自举之后，系统发生复位之前，无法再实现串行线调试 (SWD) 通信，因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

5.2 自举程序选择

下图显示了自举程序选择机制。

图 7.STM32F030xC 器件的自举程序选择



5.3 自举程序版本

下表列出了 STM32F030xC 器件自举程序版本。

表 9.STM32F030xC 自举程序版本

自举程序版本号	说明	已知限制
V5.2	初始自举程序版本	无

6 STM32F05xxx 和 STM32F030x8 器件自举程序

6.1 自举程序配置

STM32F05xxx 和 STM32F030x8 器件自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 10. 系统存储器自举模式下 STM32F05xxx 和 STM32F030x8 器件的配置

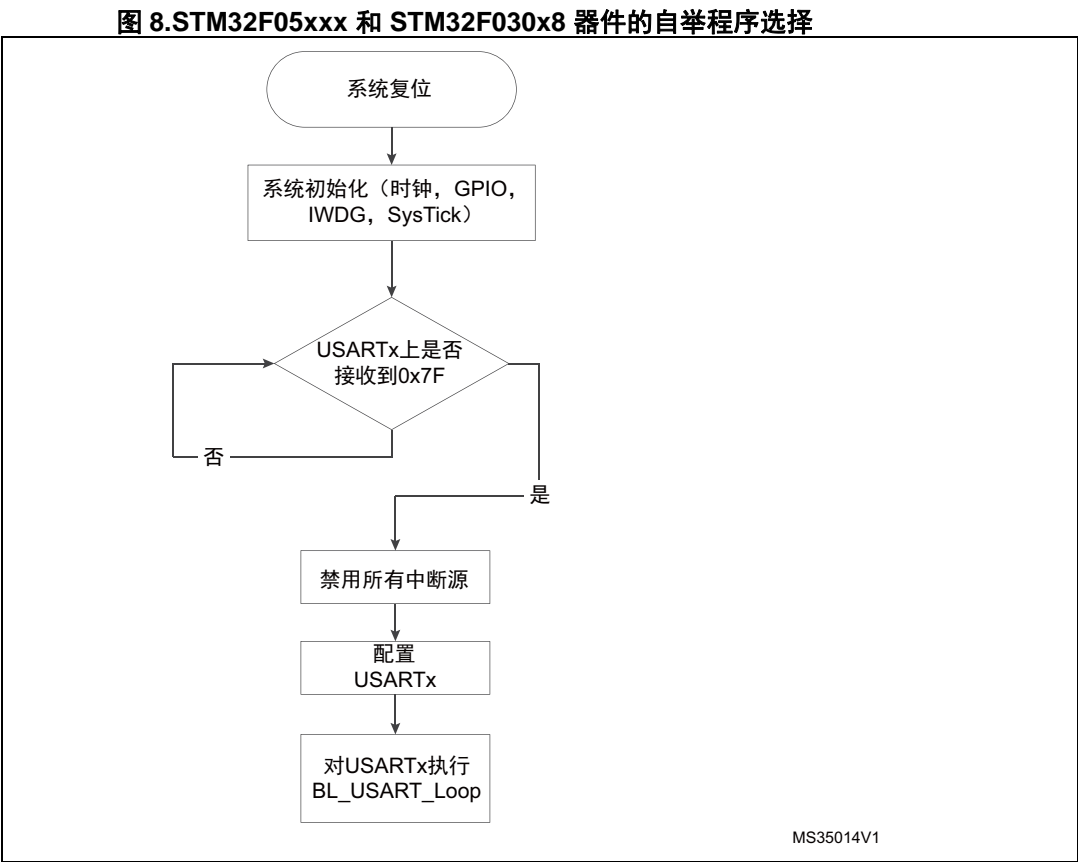
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率等于 24 MHz（使用由 HSI 驱动的 PLL）。 1 个 Flash 等待周期。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFFE000 起的 3 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	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。
USART2 自举程序	USART2	启用	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	PA15 引脚：USART2 处于接收模式。
	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式。
USART1 和 USART2 自举程序	SysTick 定时器	启用	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，自举程序执行不需要外部石英时钟。

注： STM32F05xxx 和 STM32F030x8 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_TX) 使用。

6.2 自举程序选择

下图显示了自举程序选择机制。



6.3 自举程序版本

下表列出了 STM32F05xxx 和 STM32F030x8 器件自举程序版本。

表 11.STM32F05xxx 和 STM32F030x8 器件自举程序版本

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设置（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。

7 STM32F04xxx 器件自举程序

7.1 自举程序配置

STM32F04xxx 自举程序通过应用模式 6 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 12. 系统存储器自举模式下 STM32F04xxx 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	系统时钟频率为 48 MHz，使用 HSI 48 MHz 为时钟源。
		-	对于 DFU 自举程序，使能时钟恢复系统（Clock Recovery System，CRS），支持 USB 以 HSI 48 MHz 作为时钟。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFC400 起的 13 KB 空间包含自举程序固件。
USART1 自举程序	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	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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	PA15 引脚：USART2 处于接收模式
USART1 和 USART2 自举程序	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111110x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。

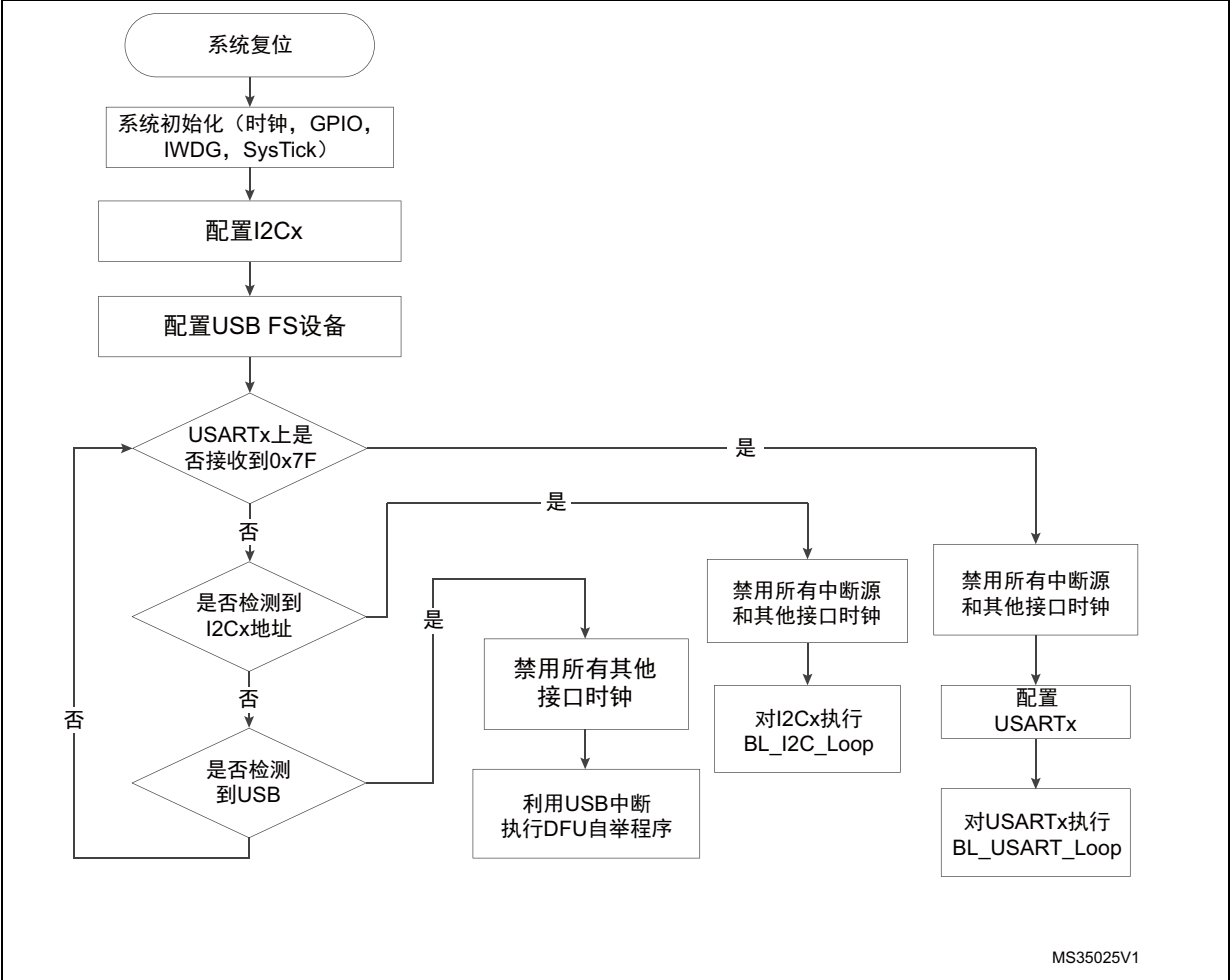
注： STM32F04xxx 器件在自举程序模式下使用 USART2 自举之后，系统发生复位之前，无法再实现串行线调试 (SWD) 通信，因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

7.2 自举程序选择

下图显示了自举程序选择机制。

图 9.STM32F04xxx 器件的自举程序选择



7.3 自举程序版本

下表列出了 STM32F04xxx 器件自举程序版本：

表 13.STM32F04xxx 自举程序版本

自举程序版本号	说明	已知限制
V10.0	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设置为（0）。

8 STM32F070x6 器件自举程序

8.1 自举程序配置

STM32F070x6 自举程序通过应用模式 6 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 14. 系统存储器自举模式下 STM32F070x6 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	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、12、8、6、4 MHz。PLL 用来为 USB 和系统时钟产生 48 MHz 频率。
		-	为 HSE 使能时钟安全系统（Clock Security System, CSS）中断。外部时钟的任何故障（或移除）都会产生系统复位。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFC400 起的 13 KB 空间包含自举程序固件。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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	PA15 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b0111110x（其中 x = 0 对应写，x = 1 对应读）
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

表 14. 系统存储器自举模式下 STM32F070x6 器件的配置（续）

自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	USB FS 配置为强制器件模式。使能 USB FS 中断向量以用于 USB DFU 通信。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB FS DM 线
	USB_DP 引脚		PA12 引脚：USB FS DP 线 无需外部上拉电阻。

注： 如果 HSI 误差超过了 1%，则自举程序不能正常工作。

注： STM32F070x6 器件在自举程序模式下使用 USART2 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。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟（HSE）：

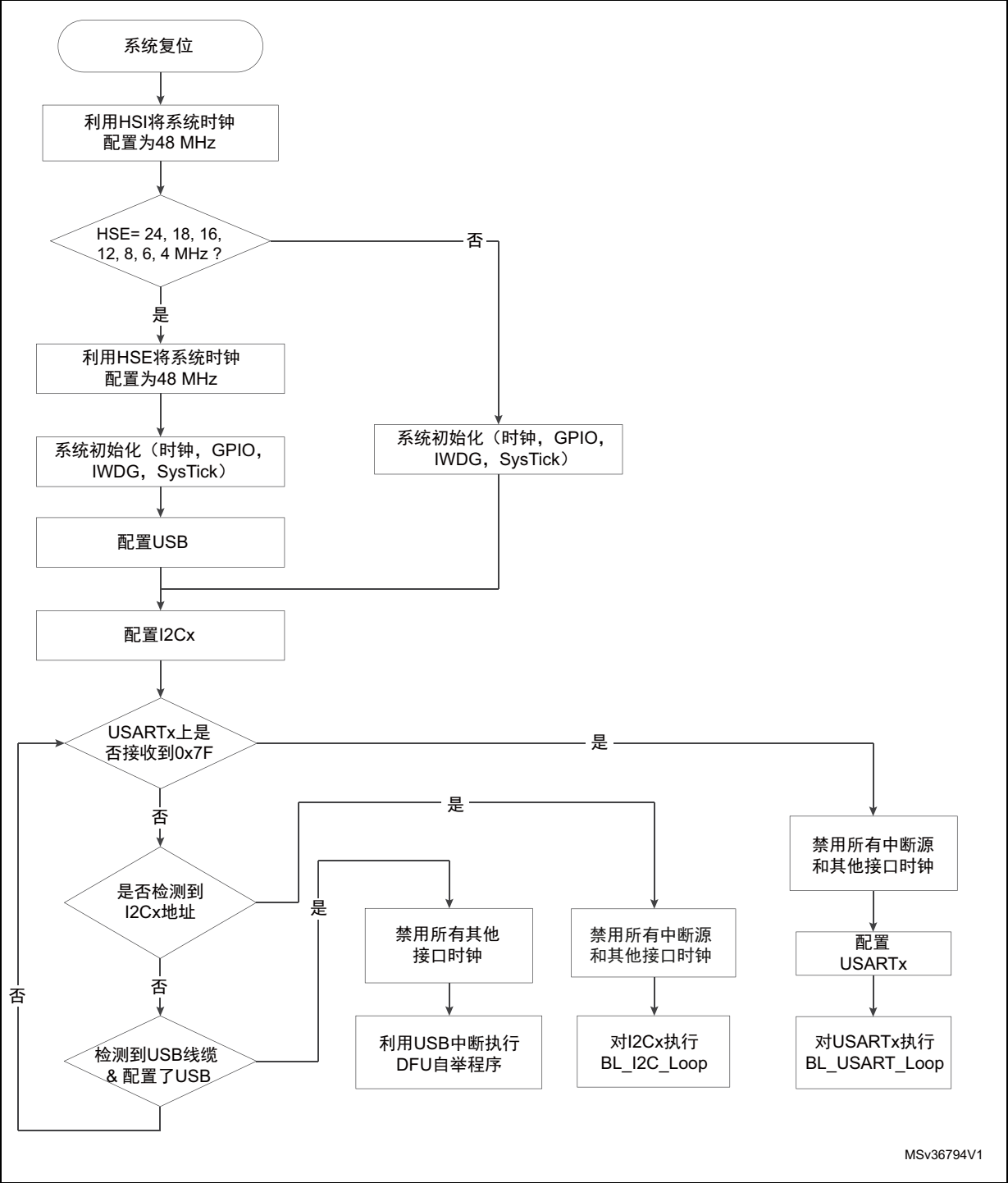
- 如果存在 HSE 且其值为 24、18、16、12、8、6 或 4 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1、USART2 和 I2C1 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1、USART2 和 I2C1 功能正常。

如果自举程序启动时连接了外部时钟（HSE），则它必须保持，因为它会用作系统时钟源。

8.2 自举程序选择

下图显示了自举程序选择机制。

图 10.STM32F070x6 器件的自举程序选择



MSv36794V1



8.3 自举程序版本

下表列出了 STM32F070x6 器件自举程序版本。

表 15.STM32F070x6 自举程序版本

自举程序版本号	说明	已知限制
V10.2	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设置为（0）。
V10.3	时钟配置固定为 HSI 8 MHz	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设置为（0）。

9 STM32F070xB 器件自举程序

9.1 自举程序配置

STM32F070xB 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 16. 系统存储器自举模式下 STM32F070xB 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	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、12、8、6、4 MHz。PLL 用来为 USB 和系统时钟产生 48 MHz 频率。
		-	为 HSE 使能时钟安全系统（Clock Security System, CSS）中断。外部时钟的任何故障（或移除）都会产生系统复位。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
USART1 自举程序	系统存储器	-	自地址 0x1FFFC800 起的 12 KB 空间包含自举程序固件。
	USART1	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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	PA15 引脚：USART2 处于接收模式
USART1 和 USART2 自举程序	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b0111011x （其中 x = 0 对应写，x = 1 对应读）
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
DFU 自举程序	USB	启用	USB FS 配置为强制器件模式。使能 USB FS 中断向量以用于 USB DFU 通信。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB FS DM 线
	USB_DP 引脚		PA12 引脚：USB FS DP 线 无需外部上拉电阻。

注： 如果 HSI 误差超过了 1%，则自举程序不能正常工作。

注： STM32F070xB 器件在自举程序模式下使用 USART2 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。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟 (HSE)：

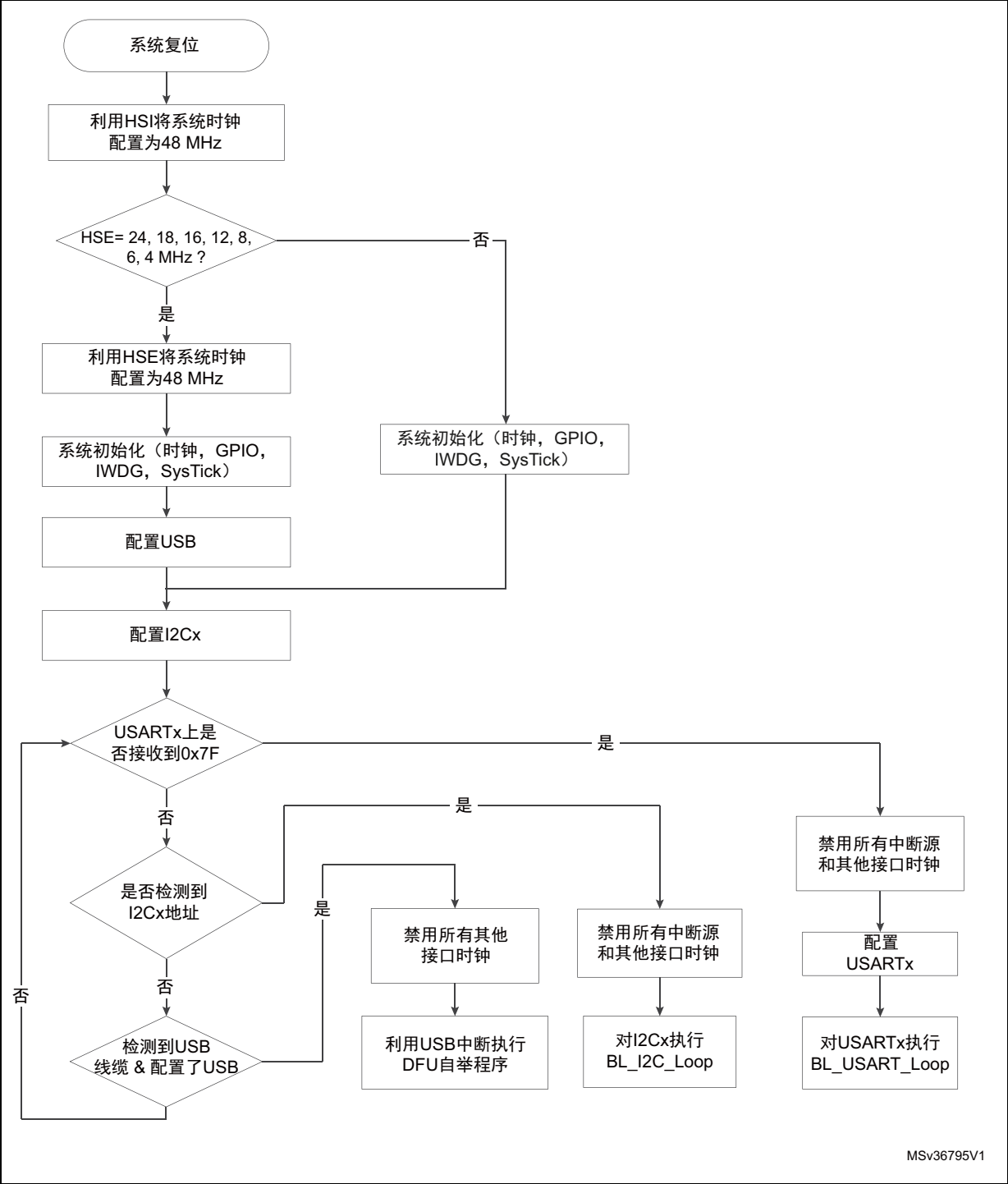
- 如果存在 HSE 且其值为 24、18、16、12、8、6 或 4 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1、USART2 和 I2C1 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1、USART2 和 I2C1 功能正常。

如果自举程序启动时连接了外部时钟 (HSE)，则它必须保持，因为它会用作系统时钟源。

9.2 自举程序选择

下图显示了自举程序选择机制。

图 11.STM32F070xB 器件的自举程序选择



MSv36795V1



9.3 自举程序版本

下表列出了 STM32F070xB 器件自举程序版本。

表 17.STM32F070xB 自举程序版本

自举程序版本号	说明	已知限制
V10.2	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设置为（0）。
V10.3	时钟配置固定为 HSI 8 MHz	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设置为（0）。

10 STM32F071xx/72xx 器件自举程序

10.1 自举程序配置

STM32F071xx/72xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 18. 系统存储器自举模式下 STM32F071xx/72xx 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	系统时钟频率为 48 MHz，使用 HSI 48 MHz 为时钟源。
		-	对于 DFU 自举程序，使能时钟恢复系统（Clock Recovery System，CRS），支持 USB 以 HSI 48 MHz 作为时钟。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFC800 起的 12 KB 空间包含自举程序固件
USART1 自举程序	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	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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	PA15 引脚：USART2 处于接收模式
USART1 和 USART2 自举程序	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111011x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。

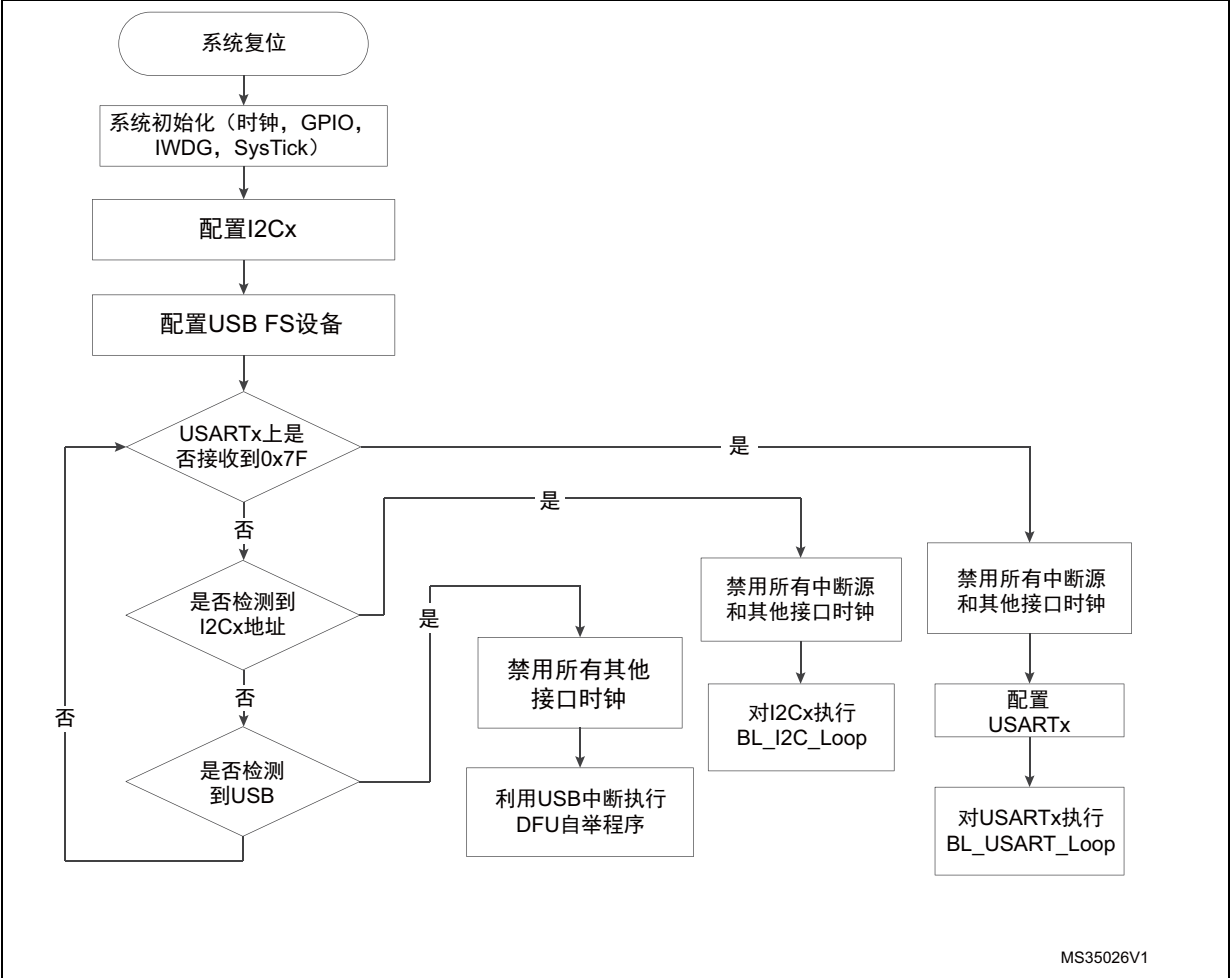
注: STM32F071xx/72xx 器件在自举程序模式下使用 USART2 自举之后, 系统发生复位之前, 无法再实现串行线调试 (SWD) 通信, 因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

系统时钟由嵌入式内部高速 RC 提供, 自举程序执行不需要外部石英时钟。

10.2 自举程序选择

下图显示了自举程序选择机制。

图 12.STM32F071xx/72xx 器件的自举程序选择



MS35026V1

10.3 自举程序版本

下表列出了 STM32F071xx/72xx 器件自举程序版本。

表 19.STM32F071xx/72xx 自举程序版本

自举程序版本号	说明	已知限制
V10.1	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设置为（0）。

11 STM32F09xxx 器件自举程序

11.1 自举程序配置

STM32F09xxx 自举程序通过应用模式 6 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 20. 系统存储器自举模式下 STM32F09xxx 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	系统时钟配置为 48 MHz，使用 HSI 48 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间包含自举程序固件。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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	PA15 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA14 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000001x (其中 x = 0 对应写，x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

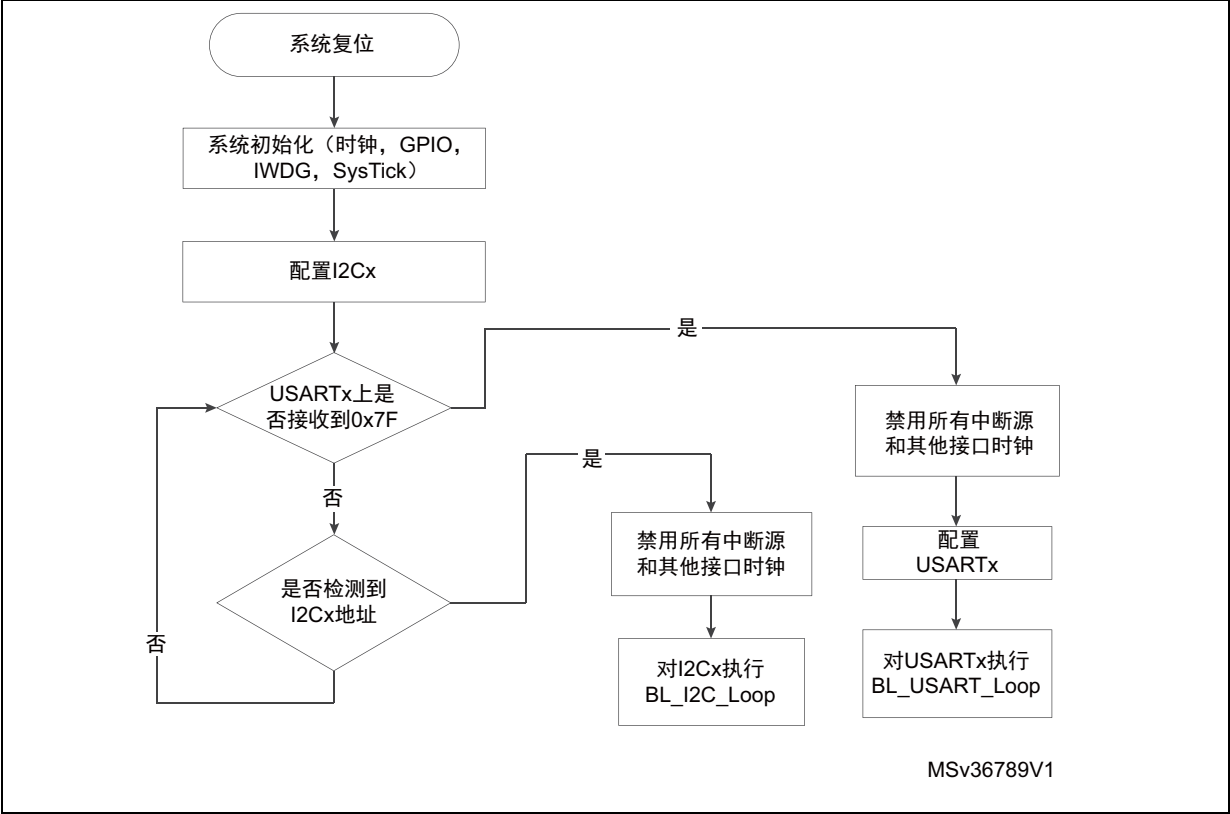
注： STM32F09xxx 器件在自举程序模式下使用 USART2 自举之后，系统发生复位之前，无法再实现串行线调试 (SWD) 通信，因为 SWD 使用的 PA14 引脚 (SWCLK) 已由自举程序 (USART2_RX) 使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

11.2 自举程序选择

下图显示了自举程序选择机制。

图 13.STM32F09xxx 器件的自举程序选择



11.3 自举程序版本

下表列出了 STM32F09xxx 器件自举程序版本。

表 21.STM32F09xxx 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	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设置为（0）。

12 STM32F10xxx 器件自举程序

12.1 自举程序配置

STM32F10xxx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 22. 系统存储器自举模式下 STM32F10xxx 器件的配置

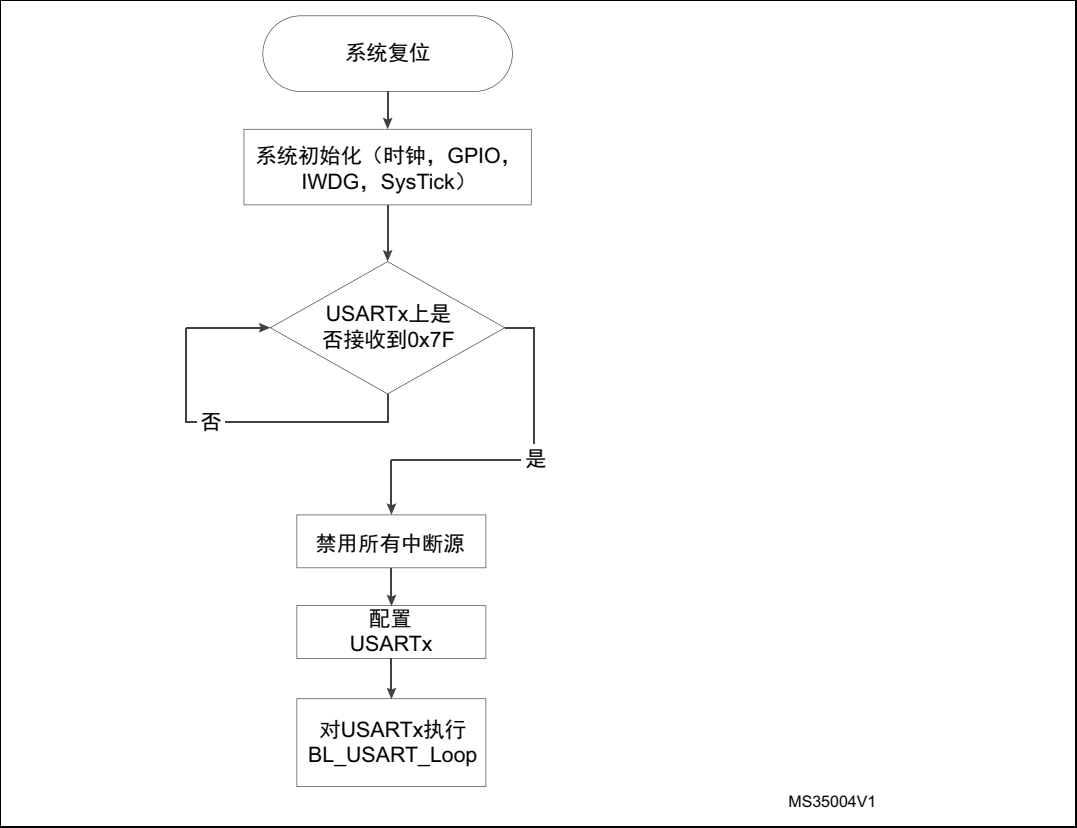
自举程序	功能 / 外设	状态	备注
USART1 自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。
	RAM	-	自地址 0x20000000 起的 512 个字节空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFFF000 起的 2 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1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

12.2 自举程序选择

下图显示了自举程序选择机制。

图 14.STM32F10xxx 器件的自举程序选择



12.3 自举程序版本

下表列出了 STM32F10xxx 器件自举程序版本。

表 23.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

13 STM32F105xx/107xx 器件自举程序

13.1 自举程序配置

STM32F105xx/107xx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 24. 系统存储器自举模式下 STM32F105xx/107xx 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 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 0x20000000 起的 4 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	启用	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	推挽输出	PA9 引脚：USART1 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	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）

表 24. 系统存储器自举模式下 STM32F105xx/107xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
CAN2 自举程序	CAN2	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 CAN1 管理着 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）。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_VBUS 引脚	输入	PA9：电源电压线
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB_DM 线
	USB_DP 引脚		PA12 引脚：USB_DP 线。 无需外部上拉电阻。

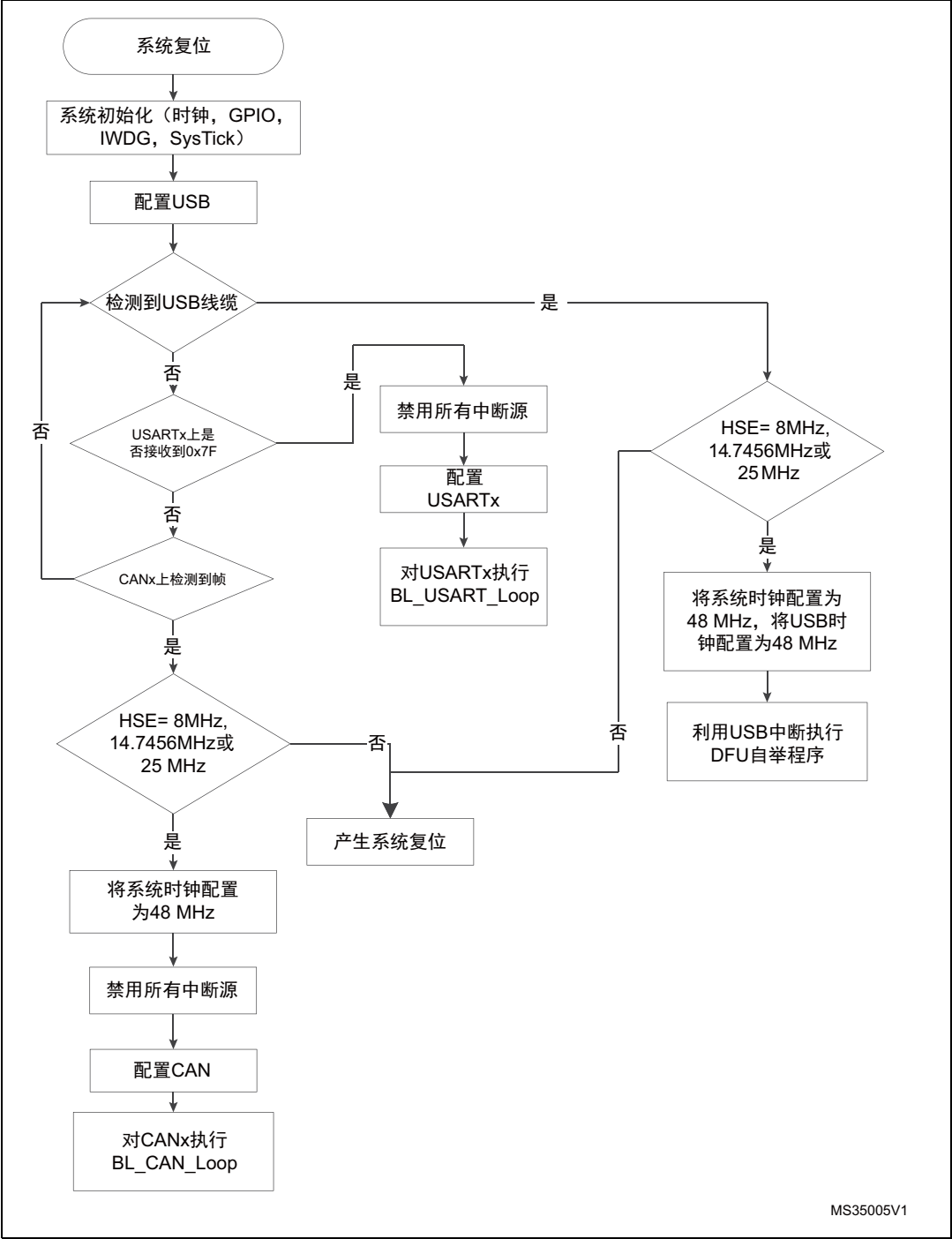
对于 USARTx 自举程序，系统时钟由内部高速 RC 提供。该内部时钟同样用于 DFU 和 CAN 自举程序，但仅限选择阶段。选择阶段结束后，执行 DFU 和 CAN 自举程序需要使用外部时钟（8 MHz、14.7456 MHz 或 25 MHz）。



13.2 自举程序选择

下图显示了自举程序选择机制。

图 15.STM32F105xx/107xx 器件的自举程序选择



13.3 自举程序版本

下表列出了 STM32F105xx/107xx 器件自举程序版本。

表 25.STM32F105xx/107xx 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。 更多详细信息，请参见第 5.3.2 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、GPIOD 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 5.3.4 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 5.3.3 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。

13.3.1 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。

13.3.2 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。

13.3.3 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。

13.3.4 插入 USB 线缆时，PA9 功耗过大（在自举程序 V2.0 中）

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。

14 STM32F10xxx 超大容量器件自举程序

14.1 自举程序配置

STM32F10xxx 超大容量器件自举程序通过应用模式 3 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源：

表 26. 系统存储器自举模式下 STM32F10xxx 超大容量器件的配置

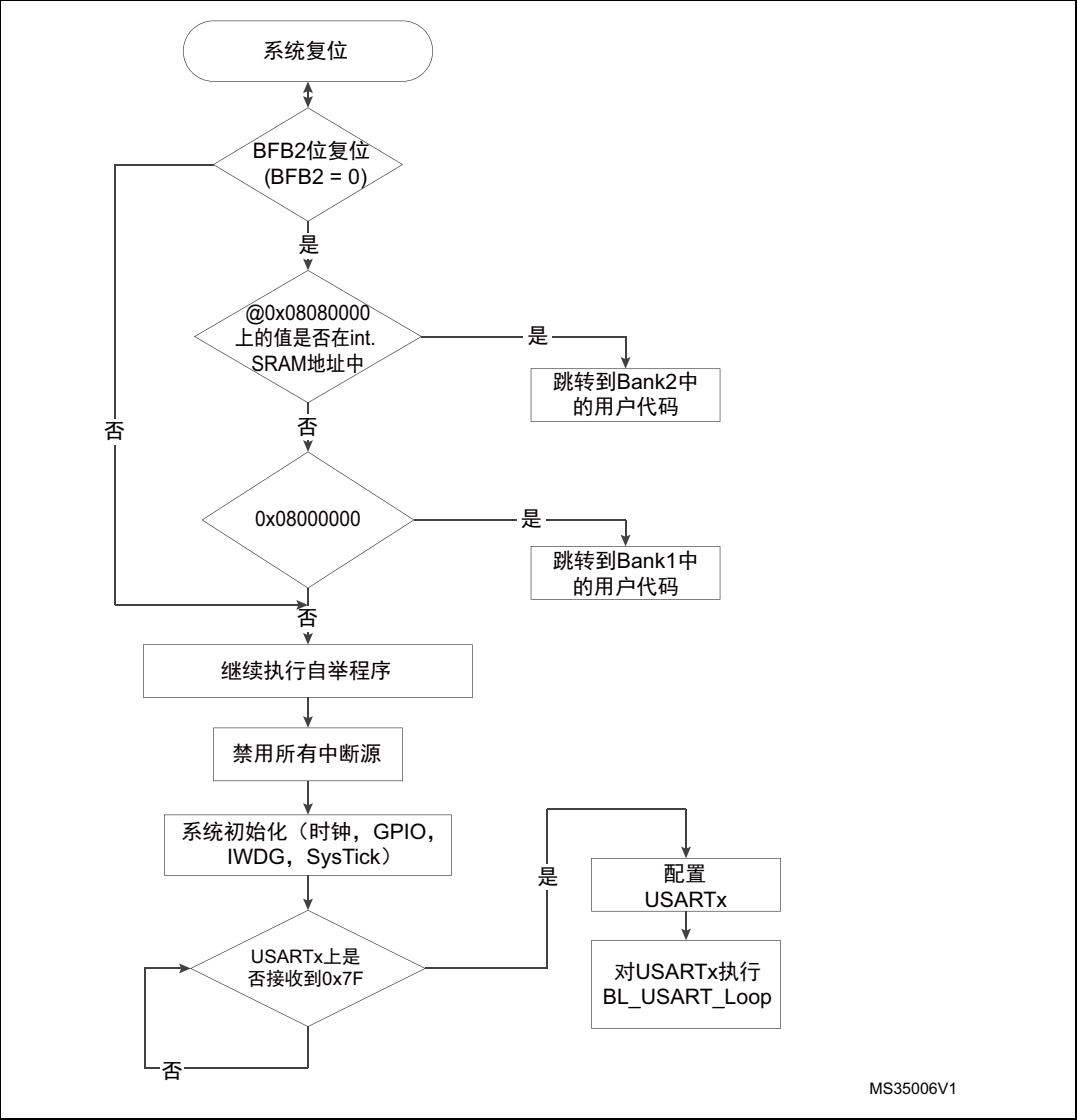
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 PLL 作为系统时钟，频率为 24 MHz。
	RAM	-	自地址 0x2000 0000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FFF E000 起的 6 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚：USART1 处于接收模式
	USART1_TX 引脚	推挽输出	PA9 引脚：USART1 处于发送模式
USART2 自举程序	USART2	启用	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）。
USART1 和 USART2 自举程序	SysTick 定时器	启用	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，自举程序执行不需要外部石英时钟。

14.2 自举程序选择

下图显示了自举程序选择机制。

图 16.STM32F10xxx 超大容量器件的自举程序选择



14.3 自举程序版本

下表列出了 STM32F10xxx 超大容量器件自举程序版本：

表 27.STM32F10xxx 超大容量自举程序版本

自举程序版本号	说明
V2.1	初始自举程序版本



15 STM32F2xxxx 器件自举程序

STM32F2xxxx 器件可使用两种自举程序版本：

- V2.x 支持 USART1 和 USART3
此版本内置在 STM32F2xxxx 器件 B 版本中。
- V3.x 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。

15.1 自举程序 V2.x

15.1.1 自举程序配置

STM32F2xxxx 自举程序通过应用模式 1 激活（详述参见表 2：自举程序激活模式）。下表介绍了该自举程序使用的硬件资源。

表 28. 系统存储器自举模式下 STM32F2xxxx 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	系统时钟频率为 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 [1.62 V, 2.1 V]。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。可使 用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PC10/PC11)	USART3	启用	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

表 28. 系统存储器自举模式下 STM32F2xxxx 器件的配置（续）

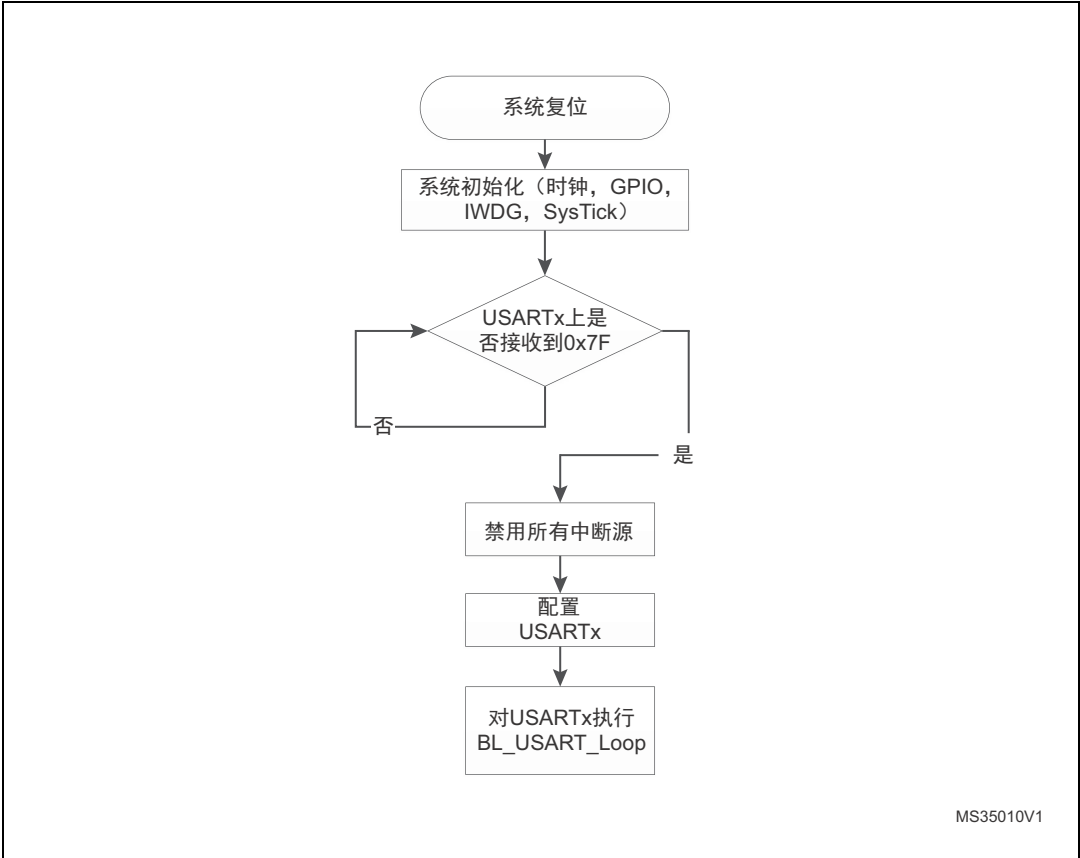
自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PB10/PB11)	USART3	启用	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位
	USART3_RX 引脚	输入	PB11 引脚：USART3 处于接收模式
	USART3_TX 引脚	输出	PB10 引脚：USART3 处于发送模式
USART1 和 USART3 自举程序	SysTick 定时器	启用	用于自动检测主机串口波特率。

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。

15.1.2 自举程序选择

下图显示了自举程序选择机制。

图 17.STM32F2xxxx 器件的自举程序 V2.x 选择



15.1.3 自举程序版本

下表列出了 STM32F10xxx 器件 V2.x 自举程序版本：

表 29.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。 对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。 ⁽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）。

15.2 自举程序 V3.x

15.2.1 自举程序配置

STM32F2xxx 自举程序通过应用模式 1 激活（详述参见表 2：自举程序激活模式）。下表介绍了该自举程序使用的硬件资源。

表 30. 系统存储器自举模式下 STM32F2xxx 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 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	系统时钟频率为 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 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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
USART3 自举程序 (PC10/PC11)	USART3	启用	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

表 30. 系统存储器自举模式下 STM32F2xxxx 器件的配置（续）

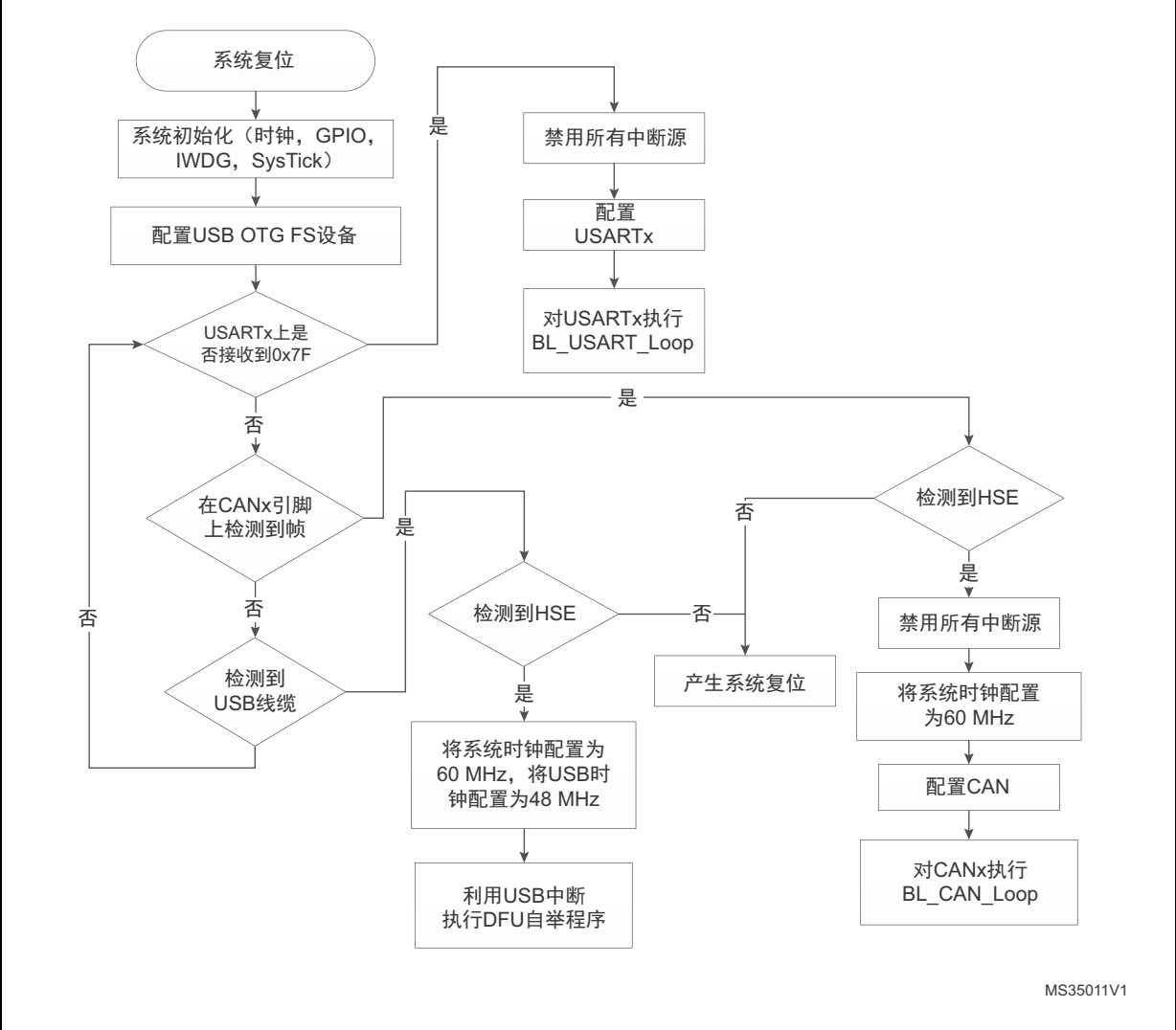
自举程序	功能 / 外设	状态	备注
USART1 和 USART3 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
CAN2 自举程序	CAN2	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 CAN1 管理着 CAN2 和 SRAM 之间的通信，因此执行 CAN2 自举程序期间要为 CAN1 提供时钟。
	CAN2_RX 引脚	输入	PB5 引脚：CAN2 处于接收模式
	CAN2_TX 引脚	输出	PB13 引脚：CAN2 处于发送模式
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

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 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。

15.2.2 自举程序选择

下图显示了自举程序选择机制。

图 18.STM32F2xxxx 器件的自举程序 V3.x 选择



15.2.3 自举程序版本

下表列出了 STM32F2xxxx 器件 V3.x 自举程序版本：

表 31.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。	<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⁾。 – 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）。
V3.3	修正 V3.2 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）。 – 对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。

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）。

16 STM32F301xx/302x4(6/8) 器件自举程序

16.1 自举程序配置

STM32F301xx/302x4(6/8) 自举程序通过应用模式 2 激活（详述参见[表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 32. 系统存储器自举模式下 STM32F301xx/302x4(6/8) 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	系统时钟频率为 48 MHz，使用 HSI 48 MHz 为时钟源。
		HSE 使能	外部时钟可用于所有的自举程序接口，且应为下列值之一：24、18、16、12、9、8、6、4、3 MHz。 PLL 用来生成 USB 48 MHz 时钟，以及生成 48 MHz 的系统时钟。
		-	对于 DFU 自举程序，时钟安全系统（CSS）中断使能。外部时钟的任何故障（或移动）都会产生系统复位。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 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	启用	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 32. 系统存储器自举模式下 STM32F301xx/302x4(6/8) 器件的配置（续）

自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11: USB DM 线。
	USB_DP 引脚		PA12: USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟（HSE）：

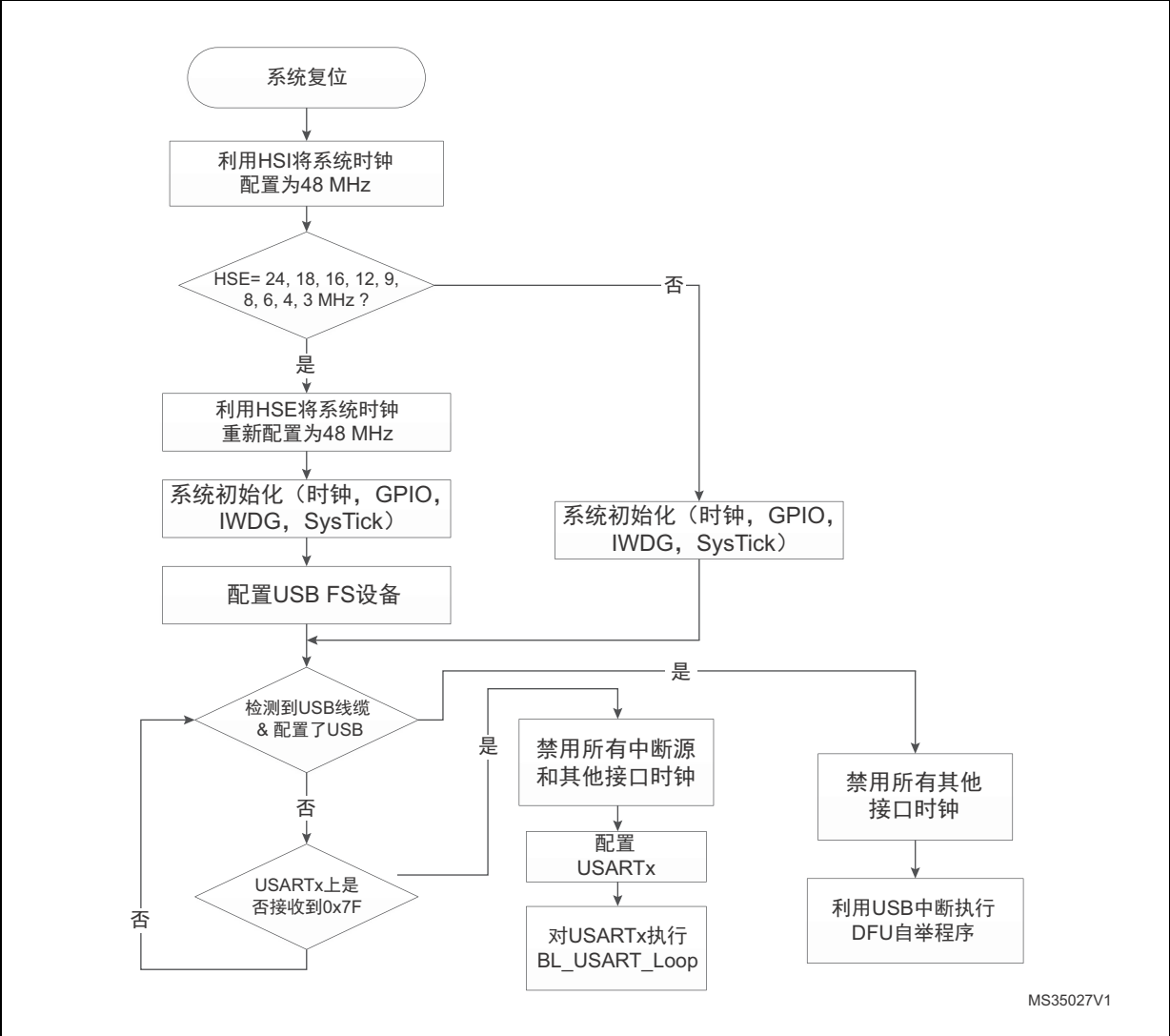
- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

如果自举程序启动时连接了外部时钟（HSE），则它必须保持，因为它会用作系统时钟源。

16.2 自举程序选择

下图显示了自举程序选择机制。

图 19.STM32F301xx/302x4(6/8) 器件的自举程序选择



MS35027V1

16.3 自举程序版本

下表列出了 STM32F301xx/302x4(6/8) 器件自举程序版本：

表 33.STM32F301xx/302x4(6/8) 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	无

17 STM32F302xB(C)/303xB(C) 器件自举程序

17.1 自举程序配置

STM32F302xB(C)/303xB(C) 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 34. 系统存储器自举模式下 STM32F302xB(C)/303xB(C) 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	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 个字节空间供自举程序固件使用。
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	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
USART2 自举程序	USART2	启用	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
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚 -		PA12：USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟（HSE）：

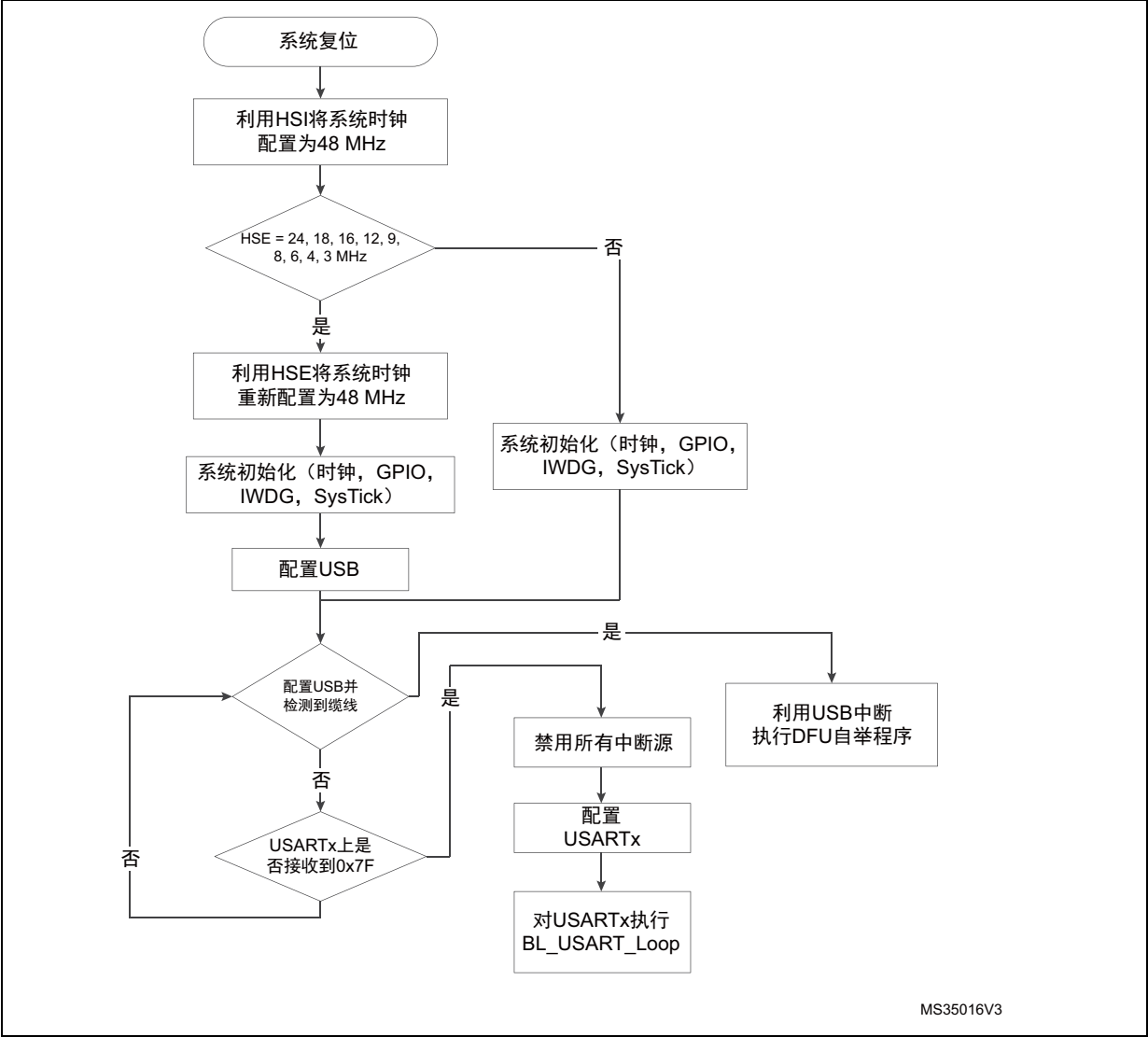
- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

如果自举程序启动时连接了外部时钟（HSE），则它必须保持，因为它会用作系统时钟源。

17.2 自举程序选择

下图显示了自举程序选择机制。

图 20.STM32F302xB(C)/303xB(C) 器件的自举程序选择



17.3 自举程序版本

下表列出了 STM32F302xB(C)/303xB(C) 器件自举程序版本。

表 35.STM32F302xB(C)/303xB(C) 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

18 STM32F302xD(E)/303xD(E) 器件自举程序

18.1 自举程序配置

STM32F302xD(E)/303xD(E) 自举程序通过应用模式 2 激活（详述参见[表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 36. 系统存储器自举模式下 STM32F302xD(E)/303xD(E) 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	系统时钟频率为 48 MHz，使用 HSI 48 MHz 为时钟源。
		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。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间包含自举程序固件
USART1 自举程序	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	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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	PA3 引脚：USART2 处于接收模式
USART1 和 USART2 自举程序	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
DFU 自举程序	USB	启用	USB FS 配置为强制器件模式。使能 USB FS 中断向量以用于 USB DFU 通信。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB FS DM 线。
	USB_DP 引脚		PA12 引脚：USB FS DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟（HSE）：

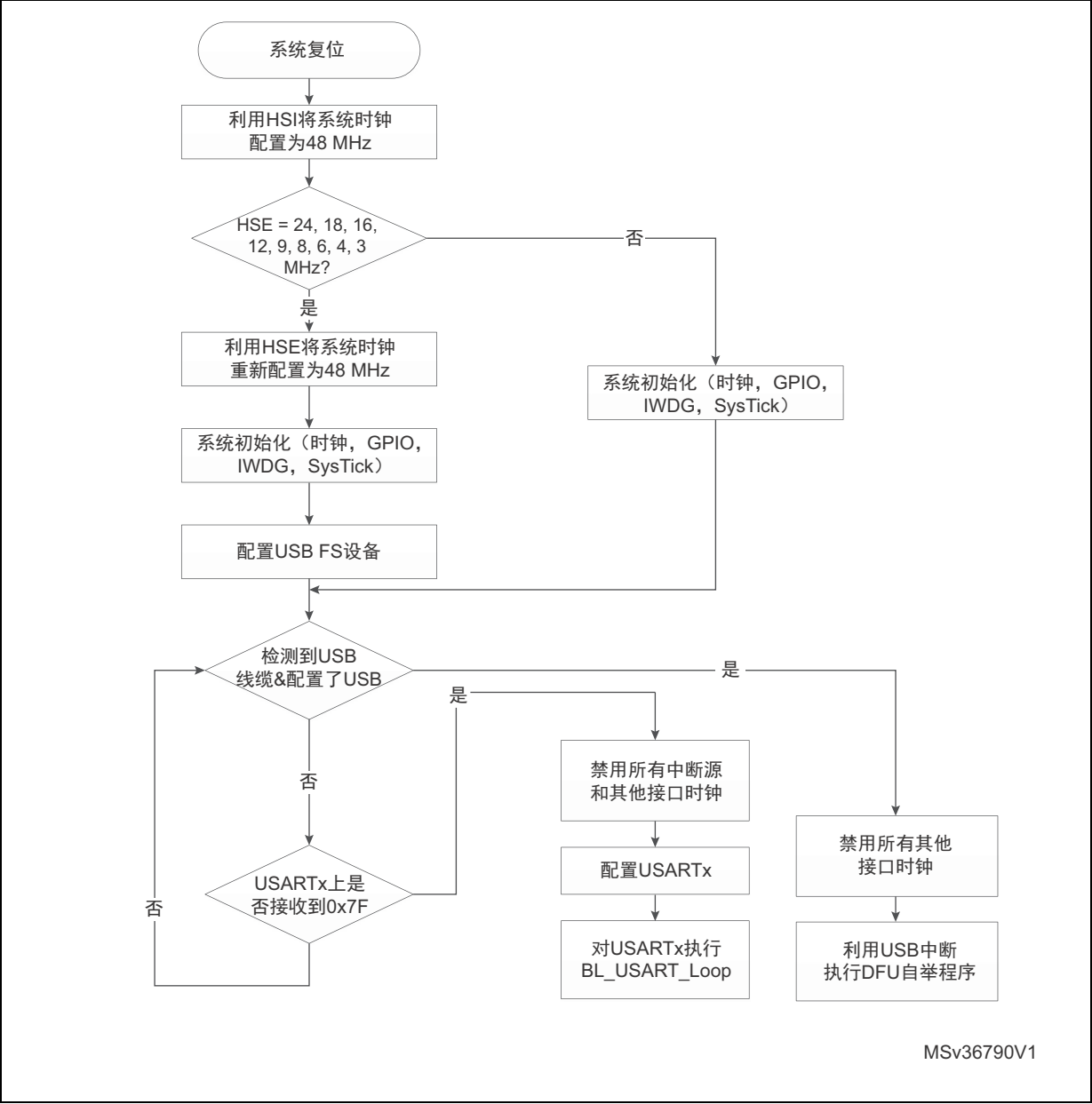
- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

如果自举程序启动时连接了外部时钟（HSE），则它必须保持，因为它会用作系统时钟源。

18.2 自举程序选择

下图显示了自举程序选择机制。

图 21.STM32F302xD(E)/303xD(E) 器件的自举程序选择



18.3 自举程序版本

下表列出了 STM32F302xD(E)/303xD(E) 器件自举程序版本。

表 37.STM32F302xD(E)/303xD(E) 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	无

19 STM32F303x4(6/8)/334xx/328xx 器件自举程序

19.1 自举程序配置

STM32F303x4(6/8)/334xx/328xx 自举程序通过应用模式 2 激活（详述参见[表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 38. 系统存储器自举模式下 STM32F303x4(6/8)/334xx/328xx 器件的配置

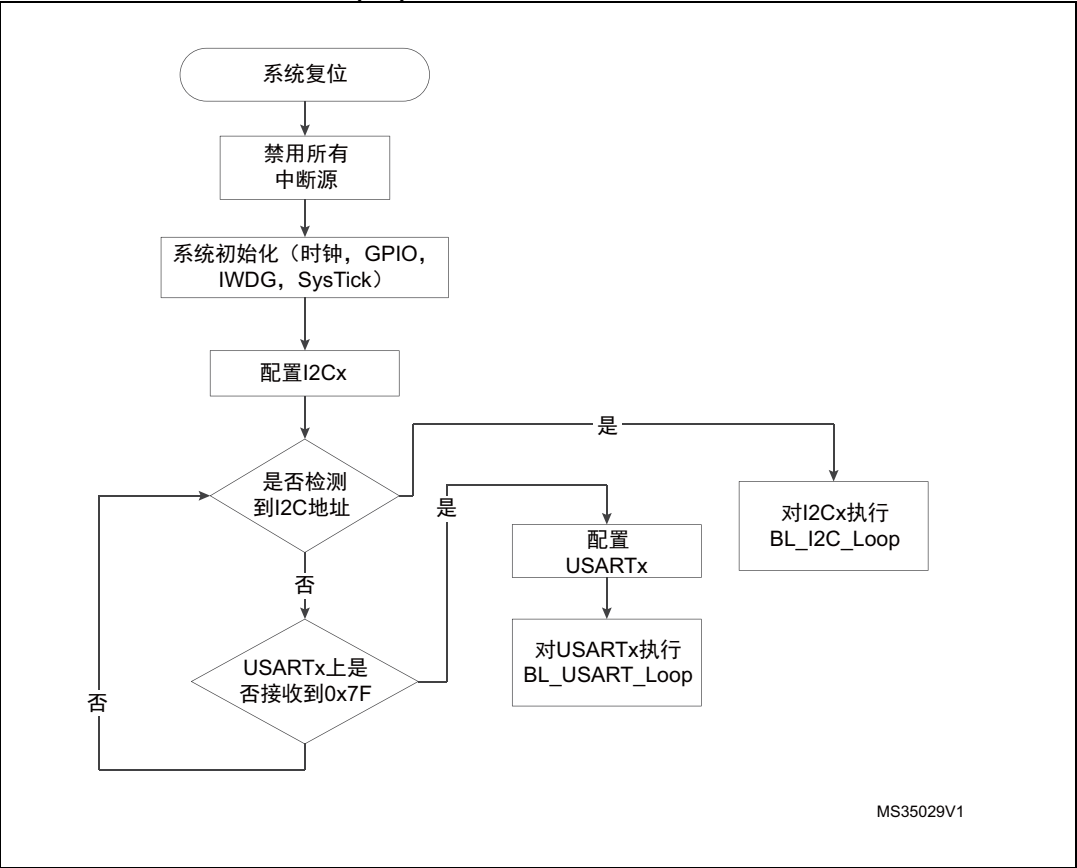
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 60 MHz，使用 HSI 8 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间包含自举程序固件
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

19.2 自举程序选择

下图显示了自举程序选择机制。

图 22.STM32F303x4(6/8)/334xx/328xx 器件的自举程序选择



19.3 自举程序版本

下表列出了 STM32F303x4(6/8)/334xx/328xx 器件自举程序版本：

表 39.STM32F303x4(6/8)/334xx/328xx 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	无

20 STM32F318xx 器件自举程序

20.1 自举程序配置

STM32F318xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 40. 系统存储器自举模式下 STM32F318xx 器件的配置

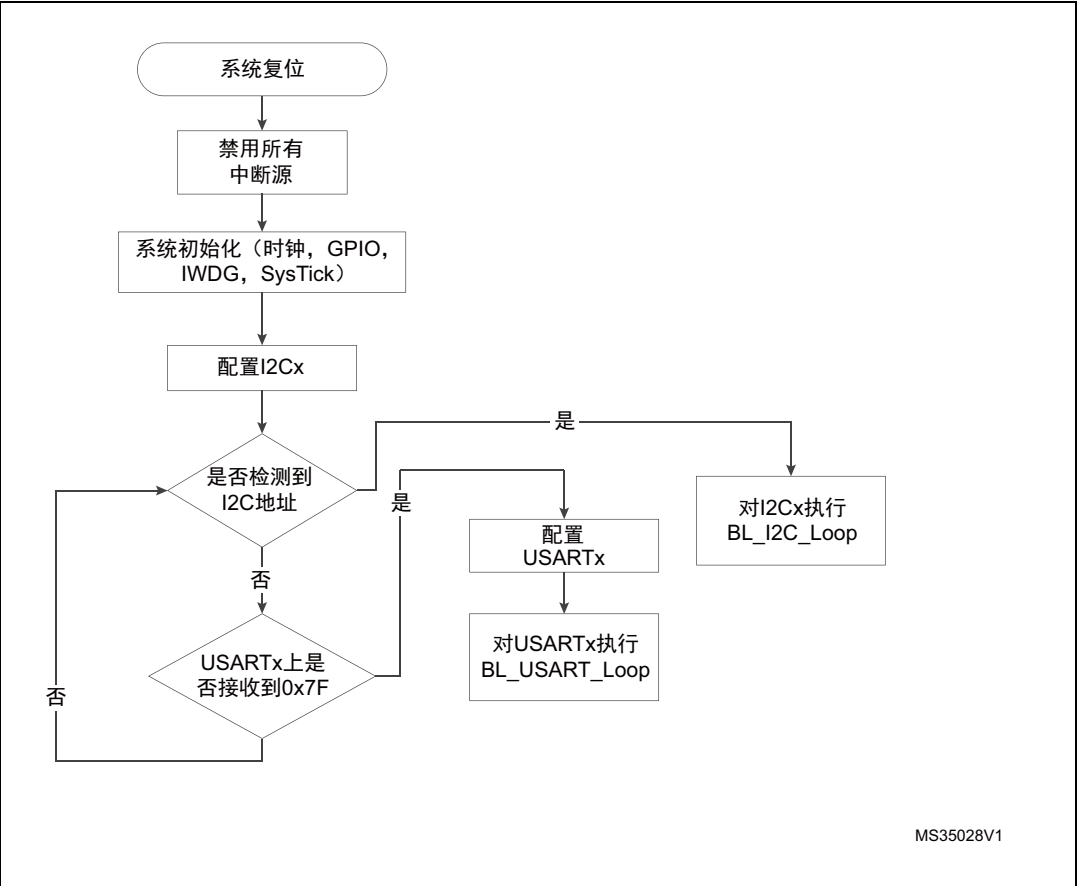
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 60 MHz，使用 HSI 8 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间包含自举程序固件
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111101x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111101x（其中 x = 0 对应写，x = 1 对应读）且数字滤波器禁用。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB5 引脚：数据线在开漏模式下使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

20.2 自举程序选择

下图显示了自举程序选择机制。

图 23.STM32F318xx 器件的自举程序选择



20.3 自举程序版本

下表列出了 STM32F318xx 器件自举程序版本：

表 41.STM32F318xx 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	无



21 STM32F358xx 器件自举程序

21.1 自举程序配置

STM32F358xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 42. 系统存储器自举模式下 STM32F358xx 器件的配置

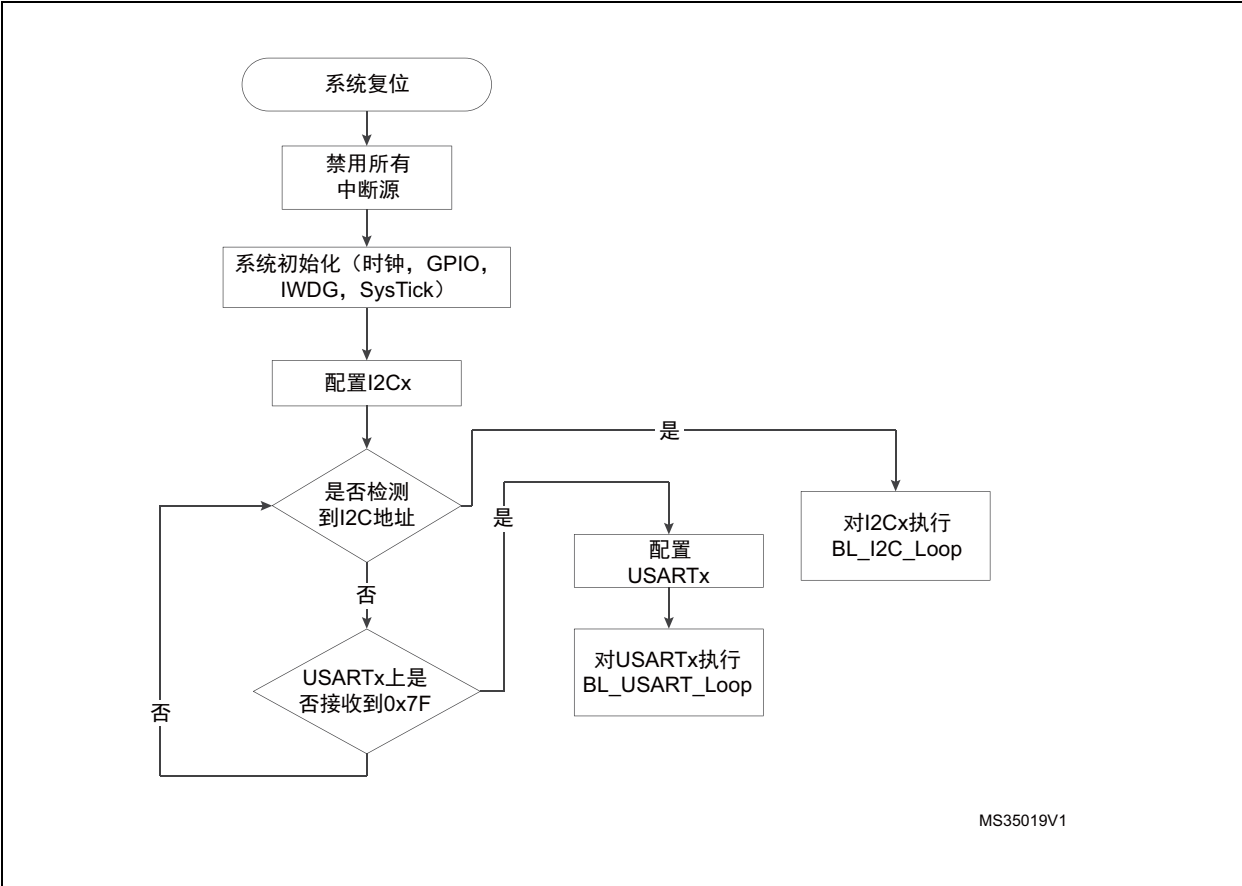
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	使用 HSI 作为系统时钟，频率为 8 MHz。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。禁止窗口功能。
	系统存储器	-	自地址 0x1FFFD800 起的 8 KB 空间。此区域包含自举程序固件。
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 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。
USART2 自举程序	USART2	启用	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。
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0110111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

21.2 自举程序选择

下图显示了自举程序选择机制。

图 24.STM32F358xx 器件的自举程序选择



21.3 自举程序版本

下表列出了 STM32F358xx 器件自举程序版本。

表 43.STM32F358xx 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。

22 STM32F373xx 器件自举程序

22.1 自举程序配置

STM32F373xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 44. 系统存储器自举模式下 STM32F373xx 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	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	启用	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	启用	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
USART2 自举程序	USART2	启用	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
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

自举程序具有两种操作情况，取决于自举程序启动时是否存在外部时钟（HSE）：

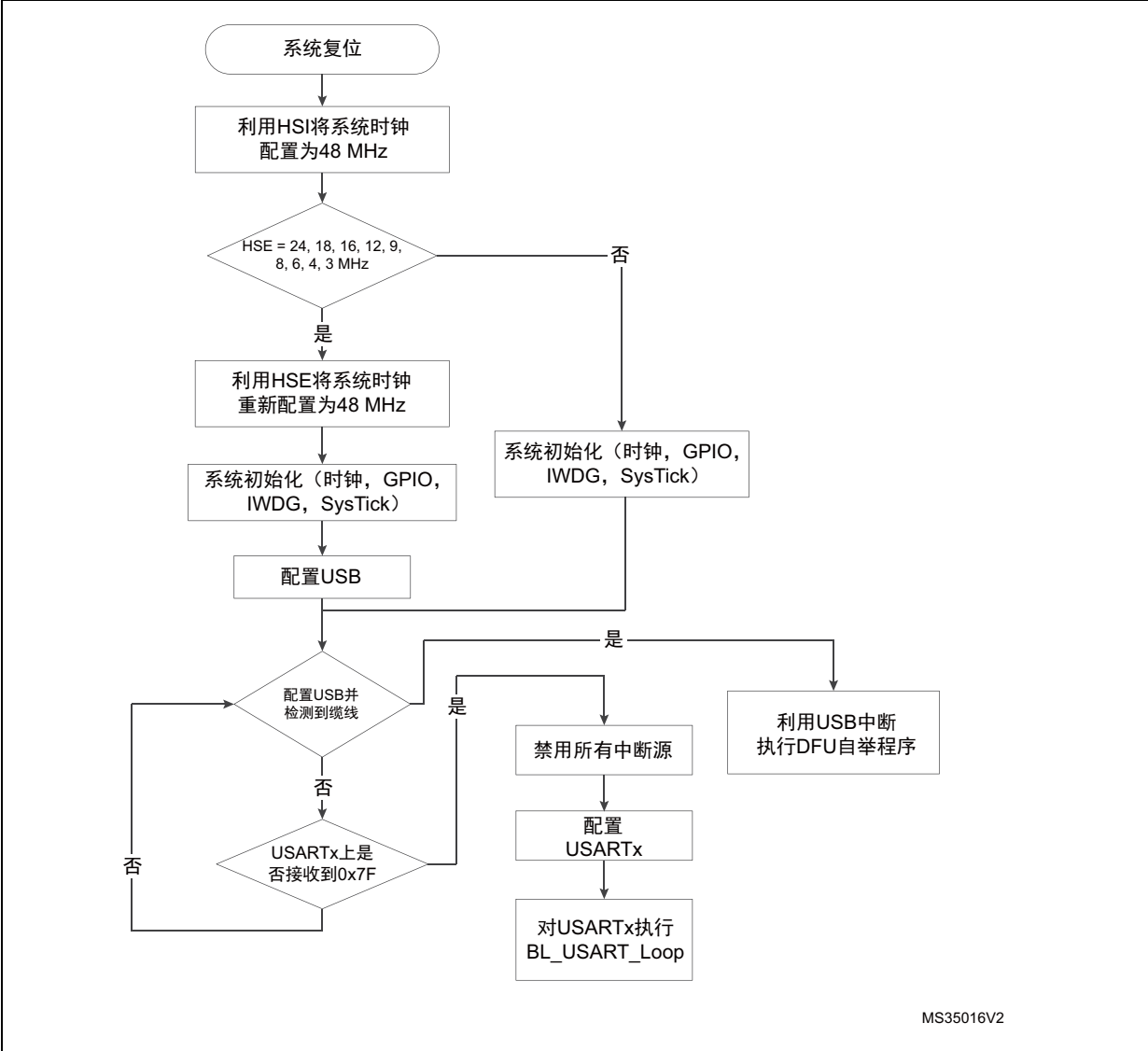
- 如果存在 HSE 且其值为 24、18、16、12、9、8、6、4 或 3 MHz，则使用 HSE 作为时钟源并将系统时钟配置为 48 MHz。DFU 接口、USART1 和 USART2 功能正常，可用于与器件自举程序进行通信。
- 如果没有 HSE，则使用 HSI 默认时钟源且只有 USART1 和 USART2 功能正常。

注：如果自举程序启动时连接了外部时钟（HSE），则它必须保持，因为它会用作系统时钟源。

22.2 自举程序选择

下图显示了自举程序选择机制。

图 25.STM32F373xx 器件的自举程序选择



22.3 自举程序版本

下表列出了 STM32F373xx 器件自举程序版本。

表 45.STM32F373xx 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

23 STM32F378xx 器件自举程序

23.1 自举程序配置

STM32F378xx 自举程序通过应用模式 2 激活（详述参见表 2：自举程序激活模式）。下表介绍了该自举程序使用的硬件资源。

表 46. 系统存储器自举模式下 STM32F378xx 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 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	启用	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。
USART2 自举程序	USART2	启用	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。
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0110111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

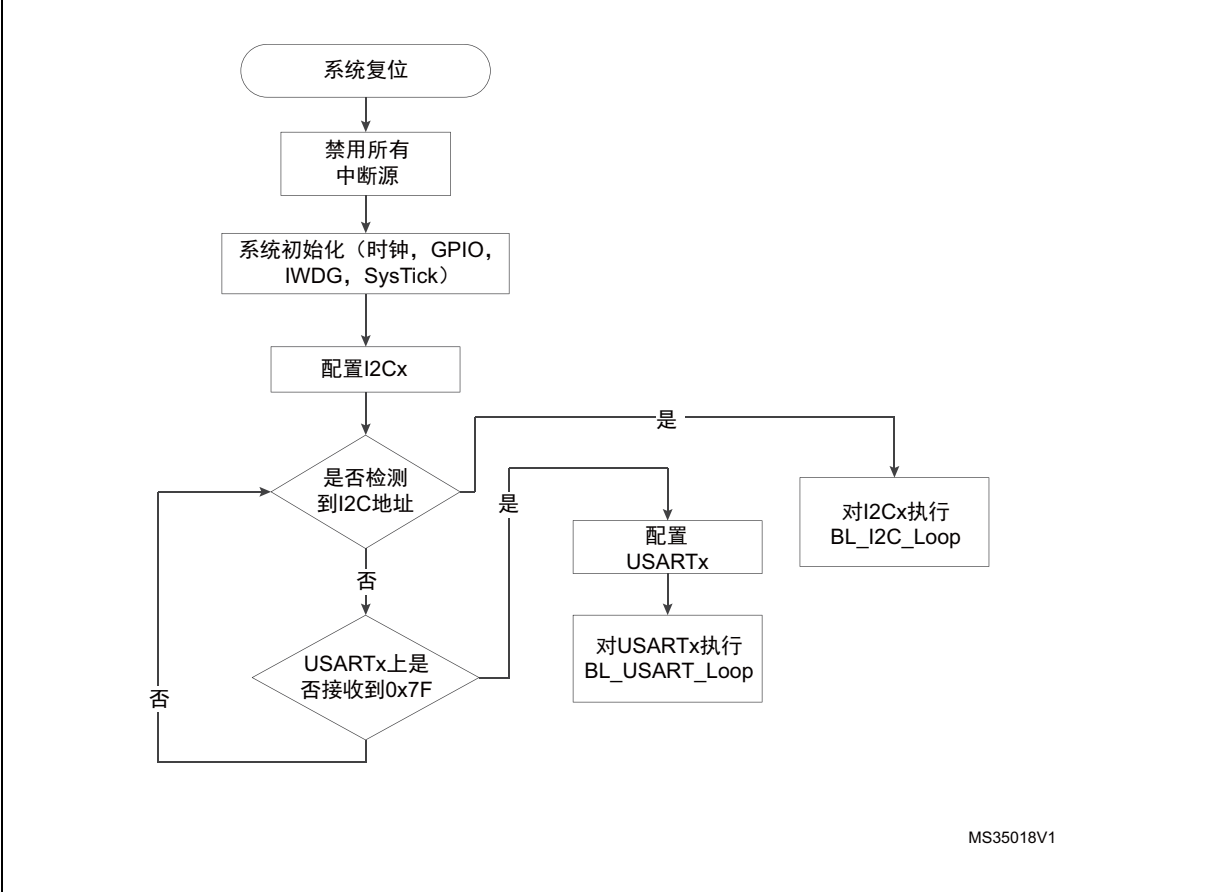


系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟。

23.2 自举程序选择

下图显示了自举程序选择机制。

图 26.STM32F378xx 器件的自举程序选择



23.3 自举程序版本

下表列出了 STM32F378xx 器件自举程序版本。

表 47.STM32F378xx 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。

24 STM32F398xx 器件自举程序

24.1 自举程序配置

STM32F398xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 48. 系统存储器自举模式下 STM32F398xx 器件的配置

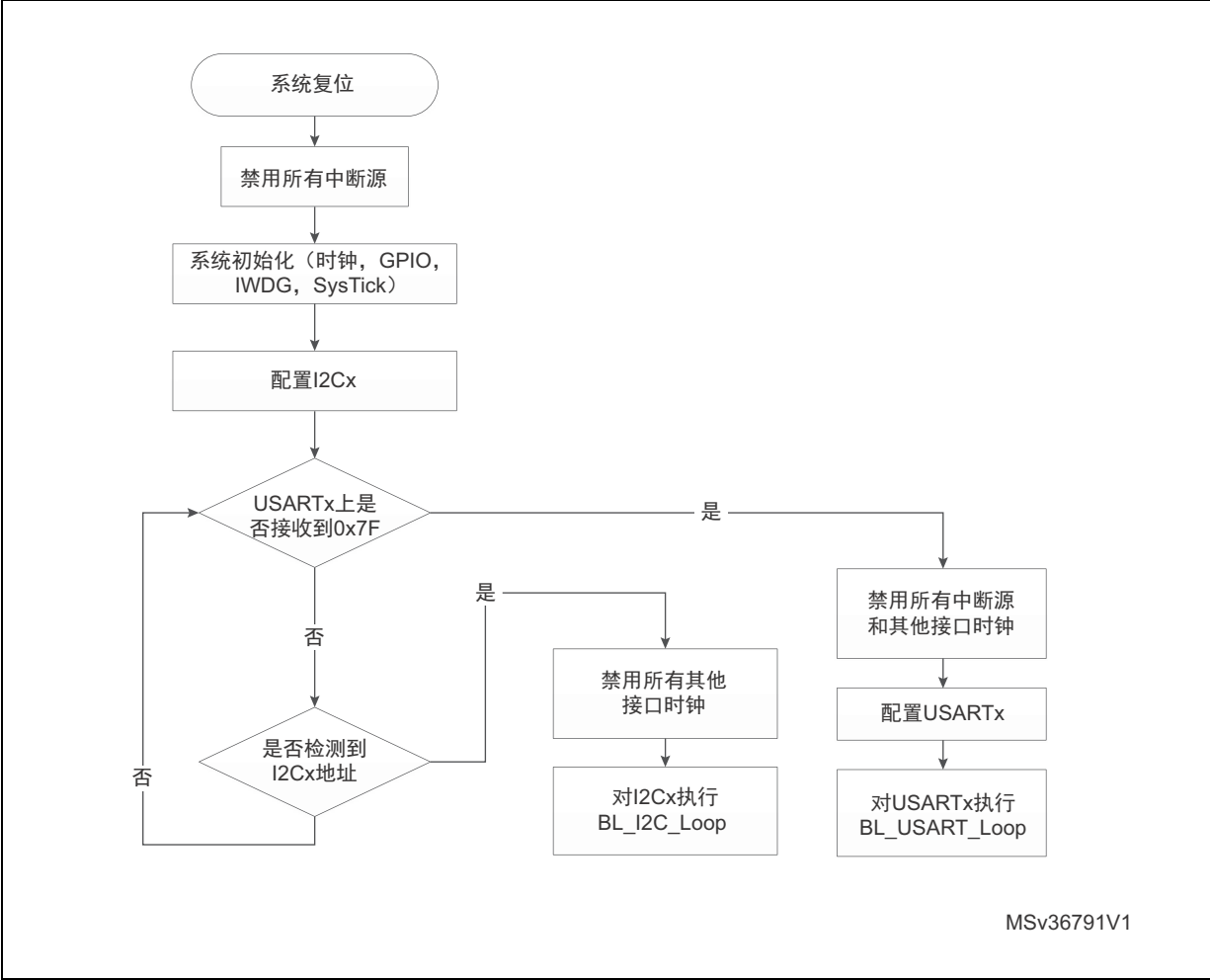
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 60 MHz，使用 HSI 8 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 6 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFFD800 起的 7 KB 空间包含自举程序固件
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB5 引脚：数据线在开漏模式下使用。

对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。

24.2 自举程序选择

下图显示了自举程序选择机制。

图 27.STM32F398xx 器件的自举程序选择



24.3 自举程序版本

下表列出了 STM32F398xx 器件自举程序版本。

表 49.STM32F398xx 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	无

25 STM32F40xxx/41xxx 器件自举程序

25.1 自举程序 V3.x

25.1.1 自举程序配置

STM32F40xxx/41xxx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 50. 系统存储器自举模式下 STM32F40xxx/41xxx 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 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	系统时钟频率为 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 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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

表 50. 系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置（续）

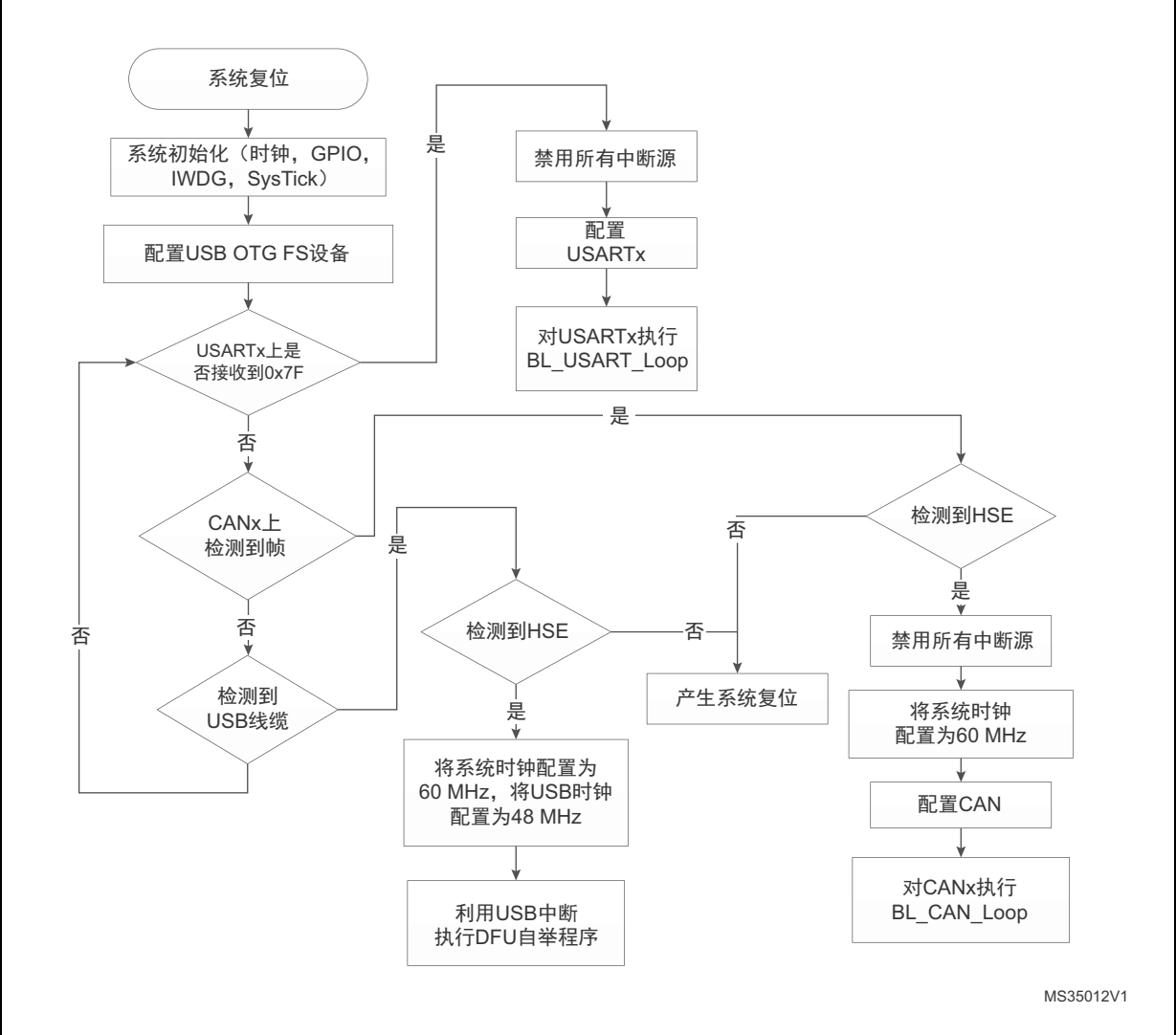
自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PC10/PC11)	USART3	启用	初始化后，USART3 的配置为：8 位，偶校验位和 1 个停止位。
	USART3_RX 引脚	输入	PC11 引脚：USART3 处于接收模式
	USART3_TX 引脚	输出	PC10 引脚：USART3 处于发送模式
USART1 和 USART3 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
CAN2 自举程序	CAN2	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 CAN1 管理着 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	PB5 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
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

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 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。

25.1.2 自举程序选择

下图显示了自举程序选择机制。

图 28.STM32F40xxx/41xxx 器件的自举程序 V3.x 选择



25.1.3 自举程序版本

下表列出了 STM32F40xxx/41xxx 器件 V3.x 自举程序版本：

表 51.STM32F40xxx/41xxx 自举程序 V3.x 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⁾。 – 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）。 – 对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。

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）。

25.2 自举程序 V9.x

25.2.1 自举程序配置

STM32F40xxx/41xxx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

注：自举程序 V9.x 版本仅嵌入到 STM32F405xx/415xx WCSP90 封装器件中。

表 52. 系统存储器自举模式下 STM32F40xxx/41xxx 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 PLL 作为系统时钟，频率为 60 MHz。 HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 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	系统时钟频率为 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 12 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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

表 52. 系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后, CAN2 的配置为: 波特率 125 kbps, 11 位标识符。 注意: 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。7 位从地址: 0b0111010x (其中 x = 0 对应写, x = 1 对应读)。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚: 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚: 数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。7 位从地址: 0b0111010x (其中 x = 0 对应写, x = 1 对应读)。
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚: 时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚: 数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。7 位从地址: 0b0111010x (其中 x = 0 对应写, x = 1 对应读)。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚: 时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚: 数据线在开漏模式下使用。

表 52. 系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置（续）

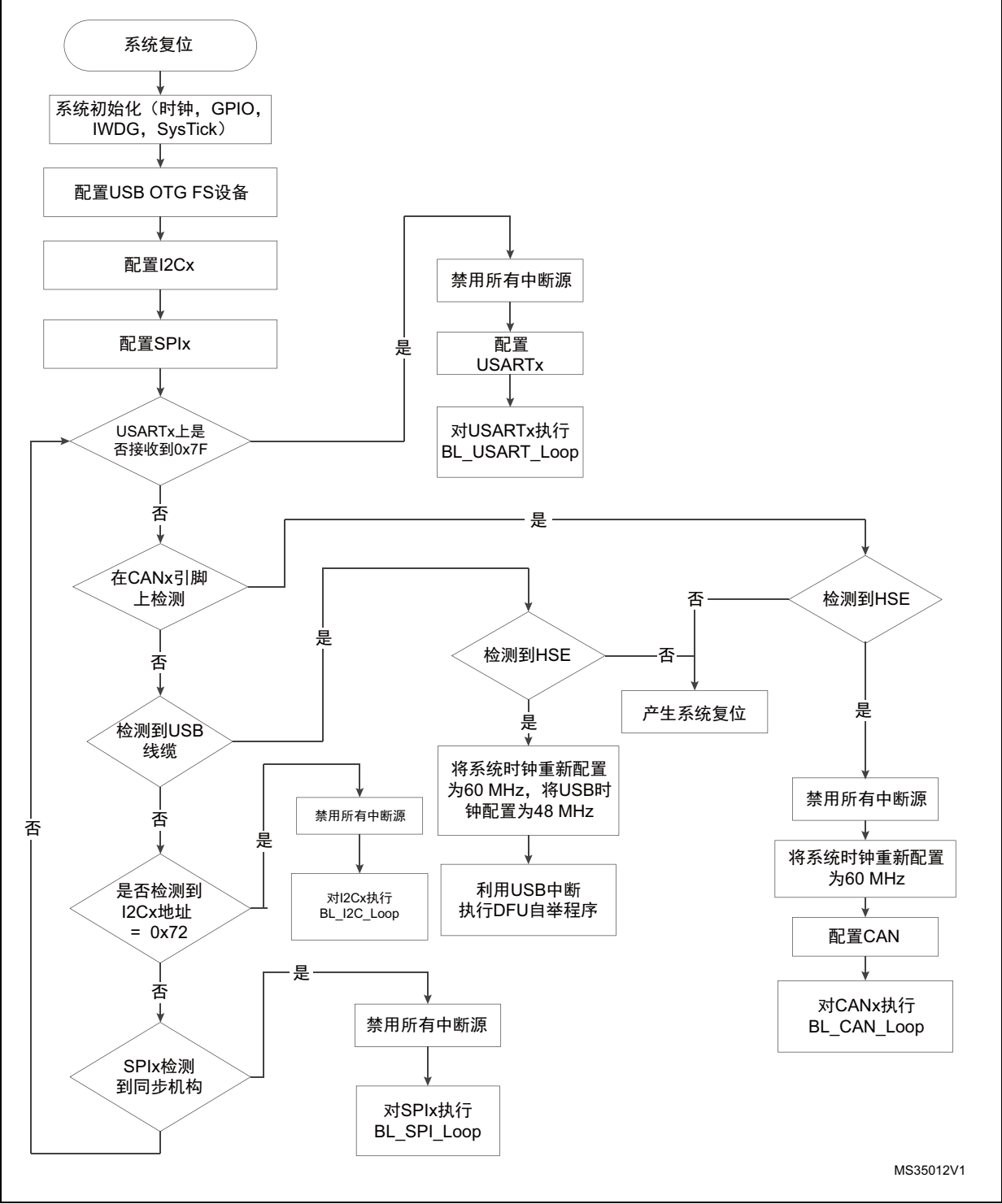
自举程序	功能 / 外设	状态	备注
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PI3 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PI2 引脚：从数据输入线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PI1 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PI0 引脚：从芯片选择引脚，用于推挽下拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

25.2.2 自举程序选择

下图显示了自举程序选择机制。

图 29.STM32F40xxx/41xxx 器件的自举程序 V9.x 选择



25.2.3 自举程序版本

下表列出了 STM32F40xxx/41xxx 器件 V9.x 自举程序版本 .

表 53.STM32F40xxx/41xxx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	此自举程序是自举程序 v3.1 的更新版本。 该新版本自举程序支持 I2C1、I2C2、I2C3、SPI1 和 SPI2 接口。 此自举程序所用的 RAM 从 8KB 增加到了 12KB。 此自举程序的 ID 为 0x90。 连接时间增加了。	<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）。– 对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。

26 STM32F401xB(C) 器件自举程序

26.1 自举程序配置

STM32F401xB(C) 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 54. 系统存储器自举模式下 STM32F401xB(C) 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 PLL 作为系统时钟，频率为 60 MHz。HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 接口期间（选择 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率为 60 MHz。仅当选择 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 12 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 54. 系统存储器自举模式下 STM32F401xB(C) 器件的配置（续）

自举程序	功能 / 外设	状态	备注
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB3 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB4 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽下拉模式。

表 54. 系统存储器自举模式下 STM32F401xB(C) 器件的配置 (续)

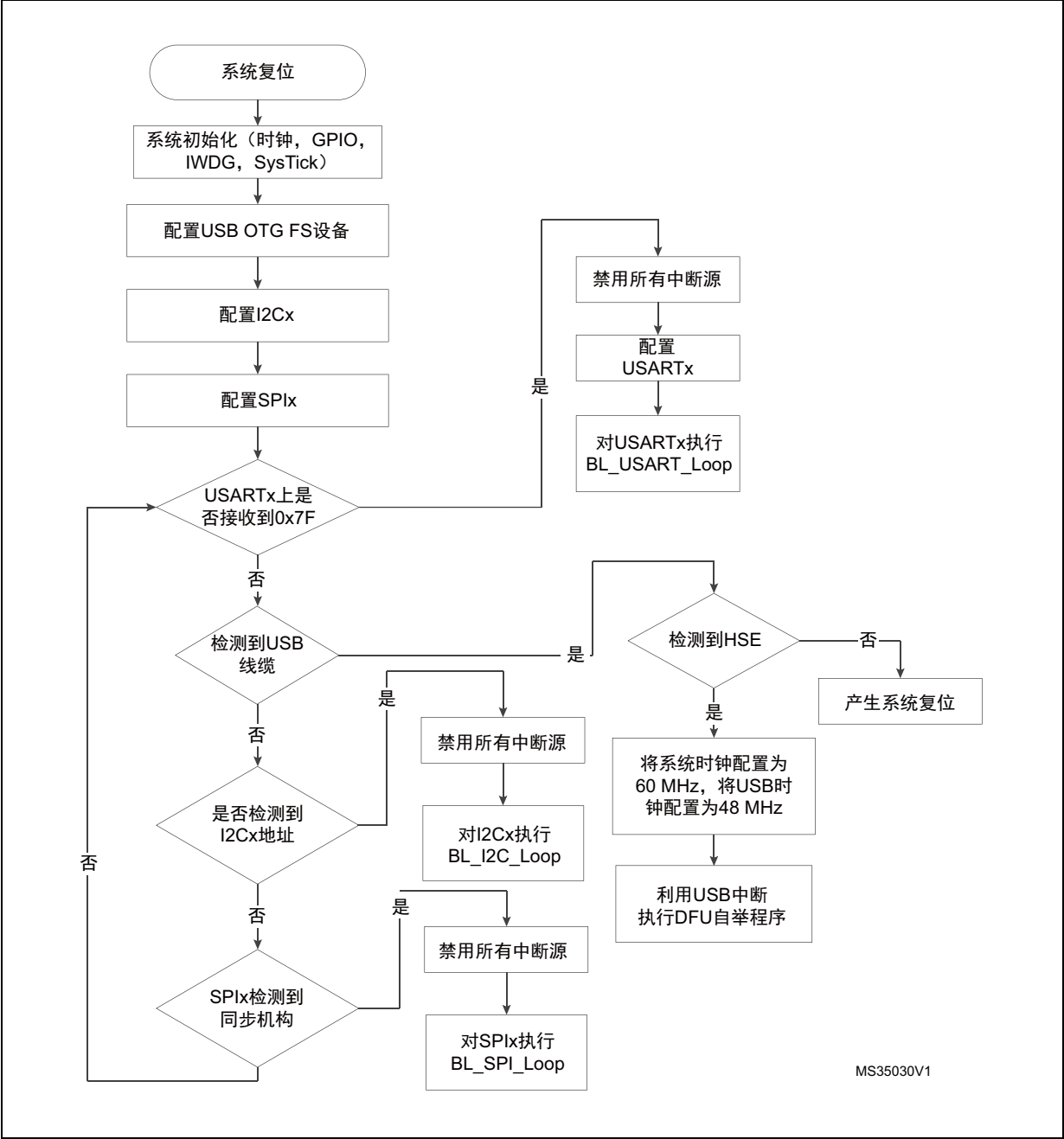
自举程序	功能 / 外设	状态	备注
SPI3 自举程序	SPI3	启用	SPI3 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI3_MOSI 引脚	输入	PC12 引脚：从数据输入线，用于推挽下拉模式
	SPI3_MISO 引脚	输出	PC11 引脚：从数据输出线，用于推挽下拉模式
	SPI3_SCK 引脚	输入	PC10 引脚：从时钟线，用于推挽下拉模式
	SPI3_NSS 引脚	输入	PA15 引脚：从芯片选择引脚，用于推挽下拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

26.2 自举程序选择

下图显示了自举程序选择机制。

图 30.STM32F401xB(C) 器件的自举程序选择



26.3 自举程序版本

下表列出了 STM32F401xB(C) 器件自举程序版本。

表 55.STM32F401xB(C) 自举程序版本

自举程序版本号	说明	已知限制
V13.0	初始自举程序版本	无



27 STM32F401xD(E) 器件自举程序

27.1 自举程序配置

STM32F401xD(E) 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 56. 系统存储器自举模式下 STM32F401xD(E) 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 PLL 作为系统时钟，频率为 60 MHz。HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 接口期间（选择 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率为 60 MHz。仅当选择 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 12 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 56. 系统存储器自举模式下 STM32F401xD(E) 器件的配置 (续)

自举程序	功能 / 外设	状态	备注
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB3 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB4 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽下拉模式。

表 56. 系统存储器自举模式下 STM32F401xD(E) 器件的配置（续）

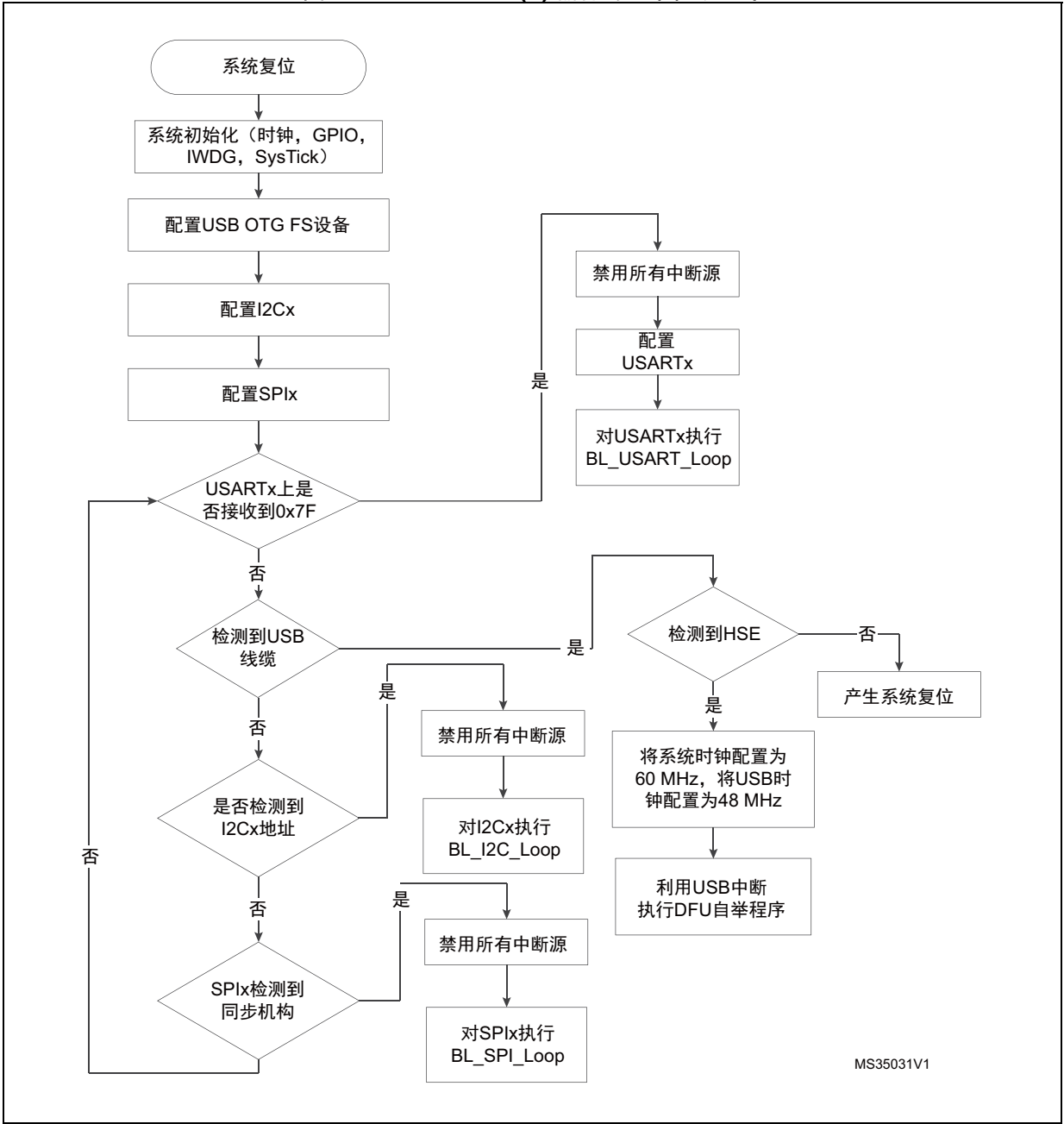
自举程序	功能 / 外设	状态	备注
SPI3 自举程序	SPI3	启用	SPI3 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI3_MOSI 引脚	输入	PC12 引脚：从数据输入线，用于推挽下拉模式
	SPI3_MISO 引脚	输出	PC11 引脚：从数据输出线，用于推挽下拉模式
	SPI3_SCK 引脚	输入	PC10 引脚：从时钟线，用于推挽下拉模式
	SPI3_NSS 引脚	输入	PA15 引脚：从芯片选择引脚，用于推挽下拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

27.2 自举程序选择

下图显示了自举程序选择机制。

图 31.STM32F401xD(E) 器件的自举程序选择



MS35031V1

27.3 自举程序版本

下表列出了 STM32F401xD(E) 器件自举程序版本。

表 57.STM32F401xD(E) 自举程序版本

自举程序版本号	说明	已知限制
V13.1	初始自举程序版本	无

28 STM32F410xx 器件自举程序

28.1 自举程序配置

STM32F410xx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 58. 系统存储器自举模式下 STM32F410xx 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	对于配置为 60 MHz 的系统时钟，以及 USART 和 I2C 自举程序操作，启动时 HSI 用作其时钟源。
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 30431 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	-	电压范围为 [1.8V, 3.6V] 在此范围内： - 3 个 Flash 等待周期。 - 系统时钟 60 MHz。 - ART 加速器使能。 - Flash 写操作按字节执行（请参考自举程序存储器管理章节来获取更多信息）。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USARTx 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。

表 58. 系统存储器自举模式下 STM32F410xx 器件的配置 (续)

自举程序	功能 / 外设	状态	备注
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB11 引脚：数据线在开漏模式下使用。
I2C4 自举程序	I2C4	启用	I2C4 配置为： I2C 速度：高达 1 MHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000111x（其中 x = 0 对应写，x = 1 对应读）。
	I2C4_SCL 引脚	输入 / 输出	PB15 引脚：对于 STM32F410Cx/Rx 器件，时钟线用于开漏模式。 PB10 引脚：对于 STM32F410Tx 器件，时钟线用于开漏模式。
	I2C4_SDA 引脚	输入 / 输出	PB14 引脚：对于 STM32F410Cx/Rx 器件，数据线用于开漏模式。 PB3 引脚：对于 STM32F410Tx 器件，数据线用于开漏模式。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，对于 STM32F410Cx/Rx 器件，用于推挽下拉模式。 PB5 引脚：从数据输入线对于 STM32F410Tx 器件，用于推挽下拉模式。
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，对于 STM32F410Cx/Rx 器件，用于推挽下拉模式。 PB4 引脚：从数据输出线，对于 STM32F410Tx 器件，用于推挽下拉模式。
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式。
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，对于 STM32F410Cx/Rx 器件，用于推挽上拉模式。 PA15 引脚：从芯片选择引脚，对于 STM32F410Tx 器件，用于推挽上拉模式。

表 58. 系统存储器自举模式下 STM32F410xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PC3 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PC2 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽上拉模式。

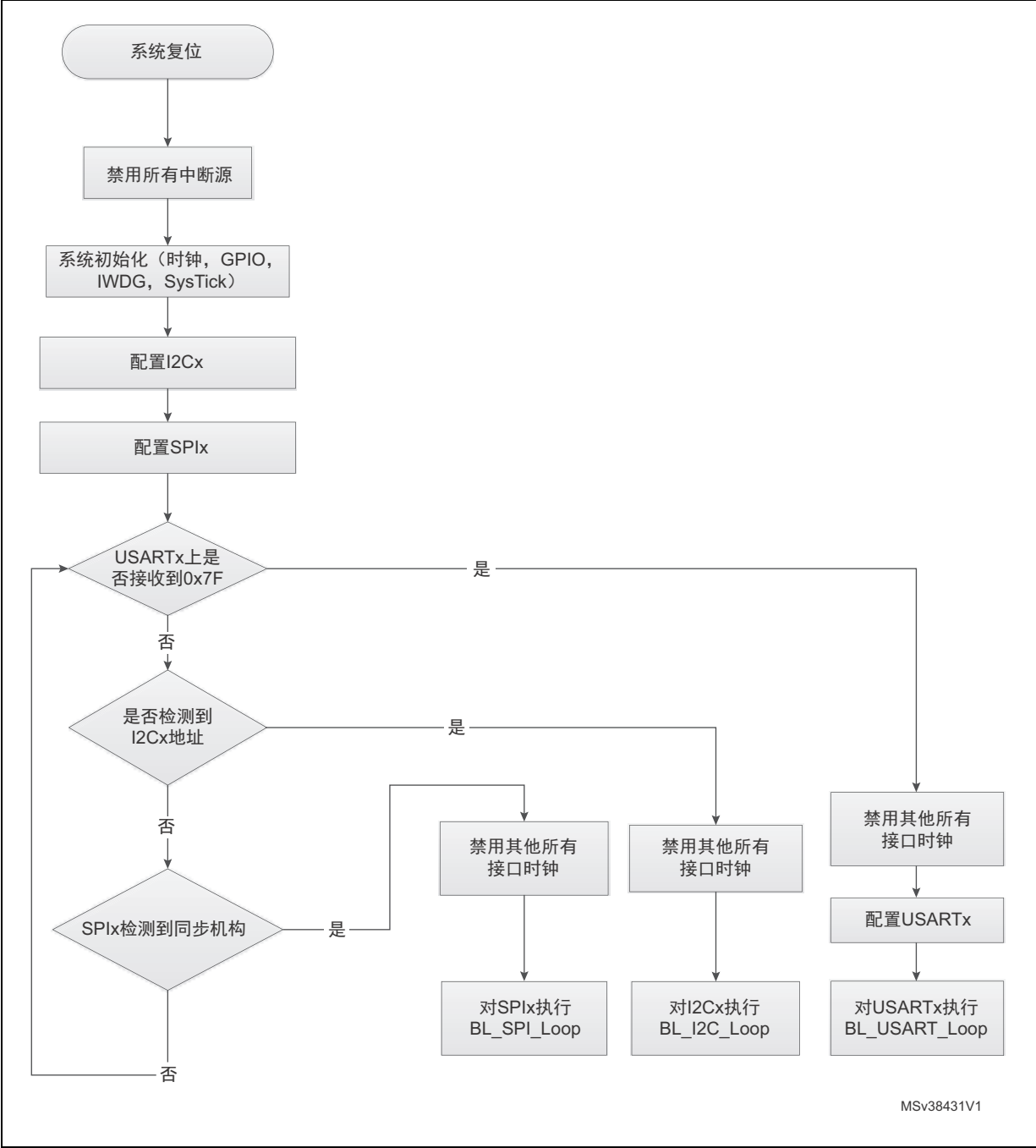
对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。



28.2 自举程序选择

图 32 显示了自举程序选择机制。

图 32.STM32F410xx 器件的自举程序 V11.x 选择



28.3 自举程序版本

下表列出了 STM32F410xx 器件自举程序 V11.x 版本。

表 59.STM32F410xx 自举程序 V11.x 版本

自举程序版本号	说明	已知限制
V11.0	初始自举程序版本	无
V11.1	对于 STM32F410Tx 器件，支持 I2C4 和 SPI1。	无



29 STM32F411xx 器件自举程序

29.1 自举程序配置

STM32F411xx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 60. 系统存储器自举模式下 STM32F411xx 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 PLL 作为系统时钟，频率为 60 MHz。HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 接口期间（选择 DFU 自举程序后，时钟源将由外部晶振提供）。
		HSE 使能	系统时钟频率为 60 MHz。仅当选择 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 12 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 60. 系统存储器自举模式下 STM32F411xx 器件的配置 (续)

自举程序	功能 / 外设	状态	备注
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB3 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111001x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB4 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽下拉模式。

表 60. 系统存储器自举模式下 STM32F411xx 器件的配置（续）

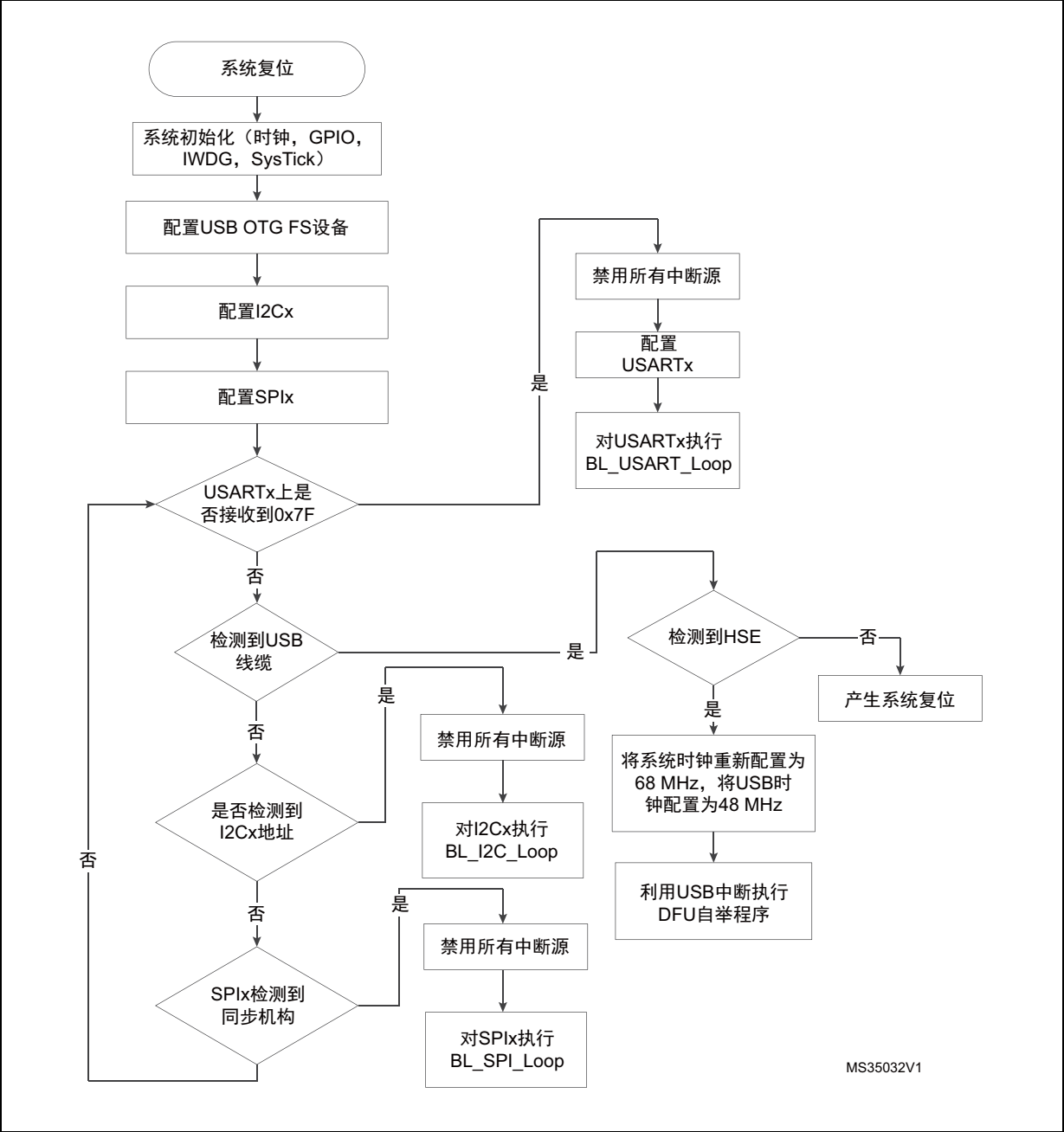
自举程序	功能 / 外设	状态	备注
SPI3 自举程序	SPI3	启用	SPI3 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI3_MOSI 引脚	输入	PC12 引脚：从数据输入线，用于推挽下拉模式
	SPI3_MISO 引脚	输出	PC11 引脚：从数据输出线，用于推挽下拉模式
	SPI3_SCK 引脚	输入	PC10 引脚：从时钟线，用于推挽下拉模式
	SPI3_NSS 引脚	输入	PA15 引脚：从芯片选择引脚，用于推挽下拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

29.2 自举程序选择

下图显示了自举程序选择机制。

图 33.STM32F411xx 器件的自举程序选择



29.3 自举程序版本

下表列出了 STM32F411xx 器件自举程序版本。

表 61.STM32F411xx 自举程序版本

自举程序版本号	说明	已知限制
V13.0	初始自举程序版本	无

30 STM32F412xx 器件自举程序

30.1 自举程序配置

STM32F412xx 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。表中介绍了该自举程序使用的硬件资源。

表 62. 系统存储器自举模式下 STM32F412xx 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	对于配置为 60 MHz 的系统时钟，以及 USART 和 I2C 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE。在此情况下，使用 HSE 作为时钟源并将系统时钟配置为 60 MHz。HSE 频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	时钟安全系统 (CSS) 中断使能以用于 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 16 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 61440 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）。
USART1 自举程序	电源	-	电压范围为 [1.8V, 3.6V] 在此范围内： - 3 个 Flash 等待周期。 - 系统时钟 60 MHz。 - ART 加速器使能。 - Flash 写操作按字节执行（请参考自举程序存储器管理章节来获取更多信息）。
	USART1	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式

表 62. 系统存储器自举模式下 STM32F412xx 器件的配置 (续)

自举程序	功能 / 外设	状态	备注
USART3 自举程序	USART3	启用	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
USARTx 自举程序	SysTick 定时器	启用	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	启用	初始化后, CAN2 的配置为: 波特率 125 kbps, 11 位标识符。 注意: 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000110x (其中 x = 0 对应写, x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚: 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚: 数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000110x (其中 x = 0 对应写, x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚: 时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚: 数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000110x (其中 x = 0 对应写, x = 1 对应读)
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚: 时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PB4 引脚: 数据线在开漏模式下使用。
I2C4 自举程序	I2C4	启用	I2C4 配置为: I2C 速度: 高达 1 MHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000110x (其中 x = 0 对应写, x = 1 对应读)
	I2C4_SCL 引脚	输入 / 输出	PB15 引脚: 时钟线在开漏模式下使用。
	I2C4_SDA 引脚	输入 / 输出	PB14 引脚: 数据线在开漏模式下使用。

表 62. 系统存储器自举模式下 STM32F412xx 器件的配置 (续)

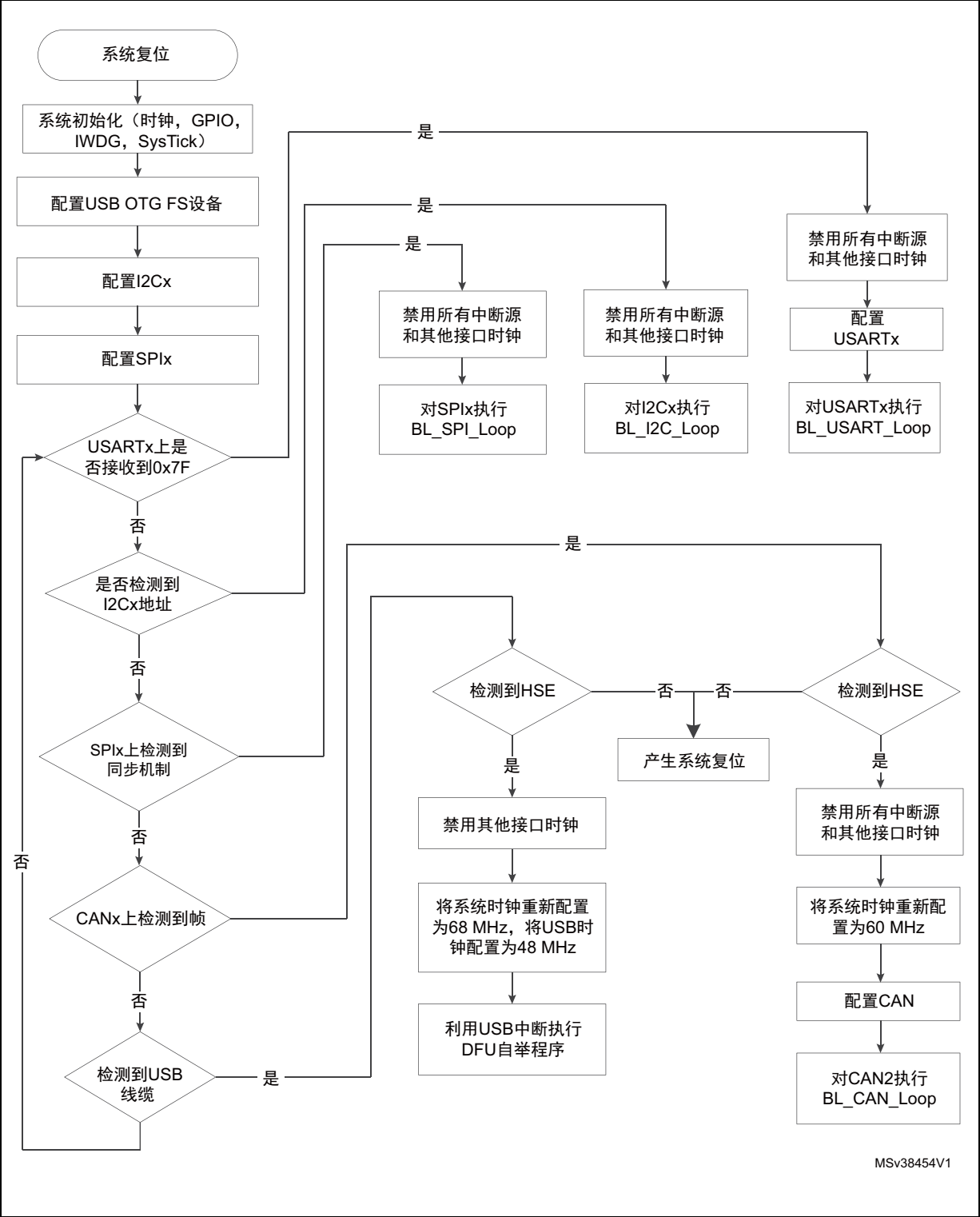
自举程序	功能 / 外设	状态	备注
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI3 自举程序	SPI3	启用	SPI3 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI3_MOSI 引脚	输入	PC12 引脚：从数据输入线，用于推挽下拉模式
	SPI3_MISO 引脚	输出	PC11 引脚：从数据输出线，用于推挽下拉模式
	SPI3_SCK 引脚	输入	PC10 引脚：从时钟线，用于推挽下拉模式
	SPI3_NSS 引脚	输入	PA15 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI4 自举程序	SPI4	启用	SPI4 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI4_MOSI 引脚	输入	PE14 引脚：从数据输入线，用于推挽下拉模式
	SPI4_MISO 引脚	输出	PE13 引脚：从数据输出线，用于推挽下拉模式
	SPI4_SCK 引脚	输入	PE12 引脚：从时钟线，用于推挽下拉模式
	SPI4_NSS 引脚	输入	PE11 引脚：从芯片选择引脚，用于推挽上拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB DM 线。
	USB_DP 引脚		PA12 引脚：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx 和 I2Cx 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 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。

30.2 自举程序选择

图 34 显示了自举程序选择机制。

图 34.STM32F412xx 器件的自举程序 V9.x 选择



30.3 自举程序版本

下表列出了 STM32F412xx 器件自举程序 V9.x 版本。

表 63.STM32F412xx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	初始自举程序版本	无

31 STM32F42xxx/43xxx 器件自举程序

31.1 自举程序 V7.x

31.1.1 自举程序配置

STM32F42xxx/43xxx 自举程序通过应用模式 5 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 64. 系统存储器自举模式下 STM32F42xxx/43xxx 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 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 USART 或 I2C 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	系统时钟频率为 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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

表 64. 系统存储器自举模式下 STM32F42xxx/43xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚：数据线在开漏模式下使用。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx 和 I2Cx 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 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。

31.1.2 自举程序选择

下图显示了自举程序选择机制。

图 35.STM32F42xxx/43xxx 自举程序 V7.x 的双存储区自举实现

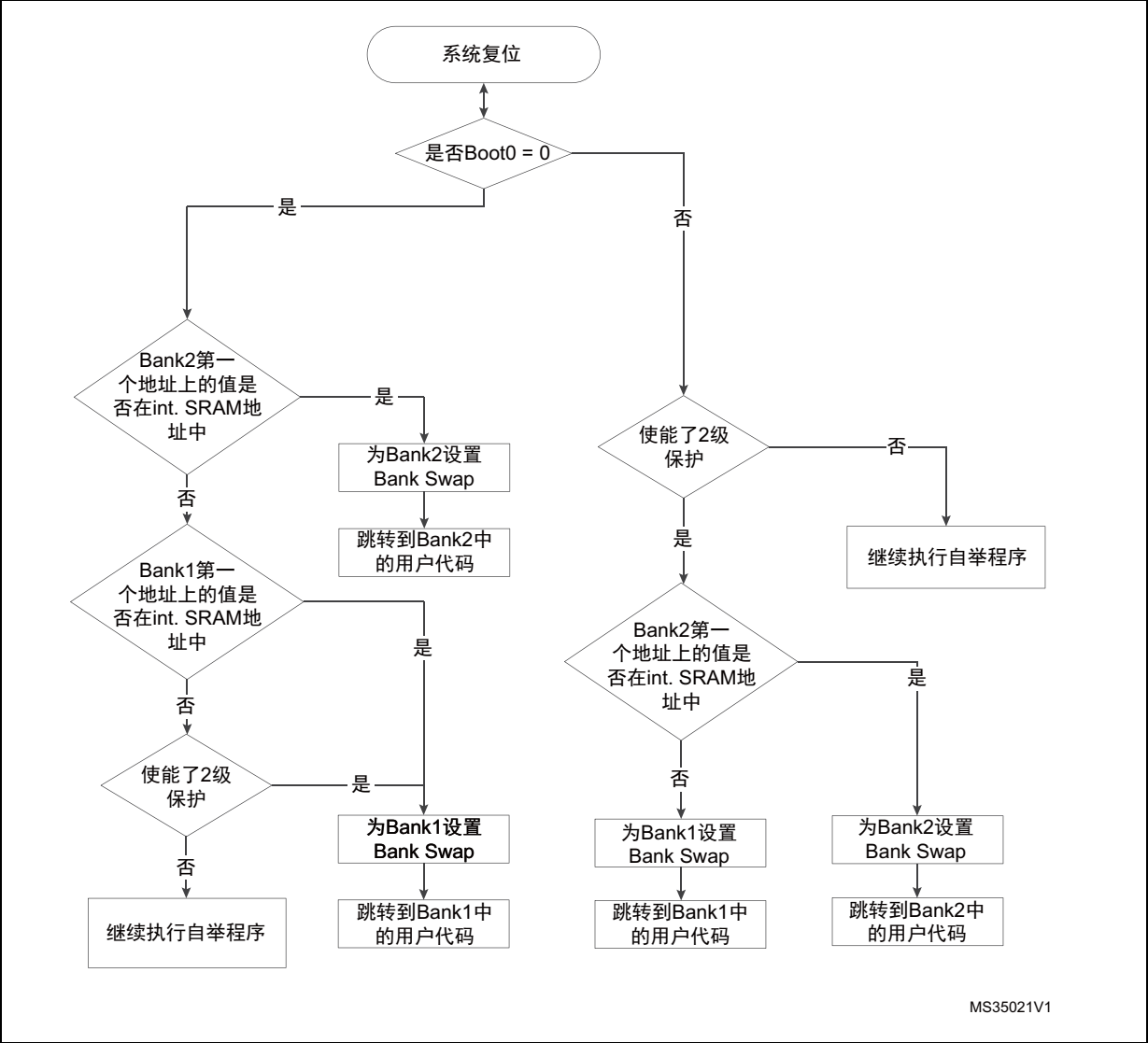
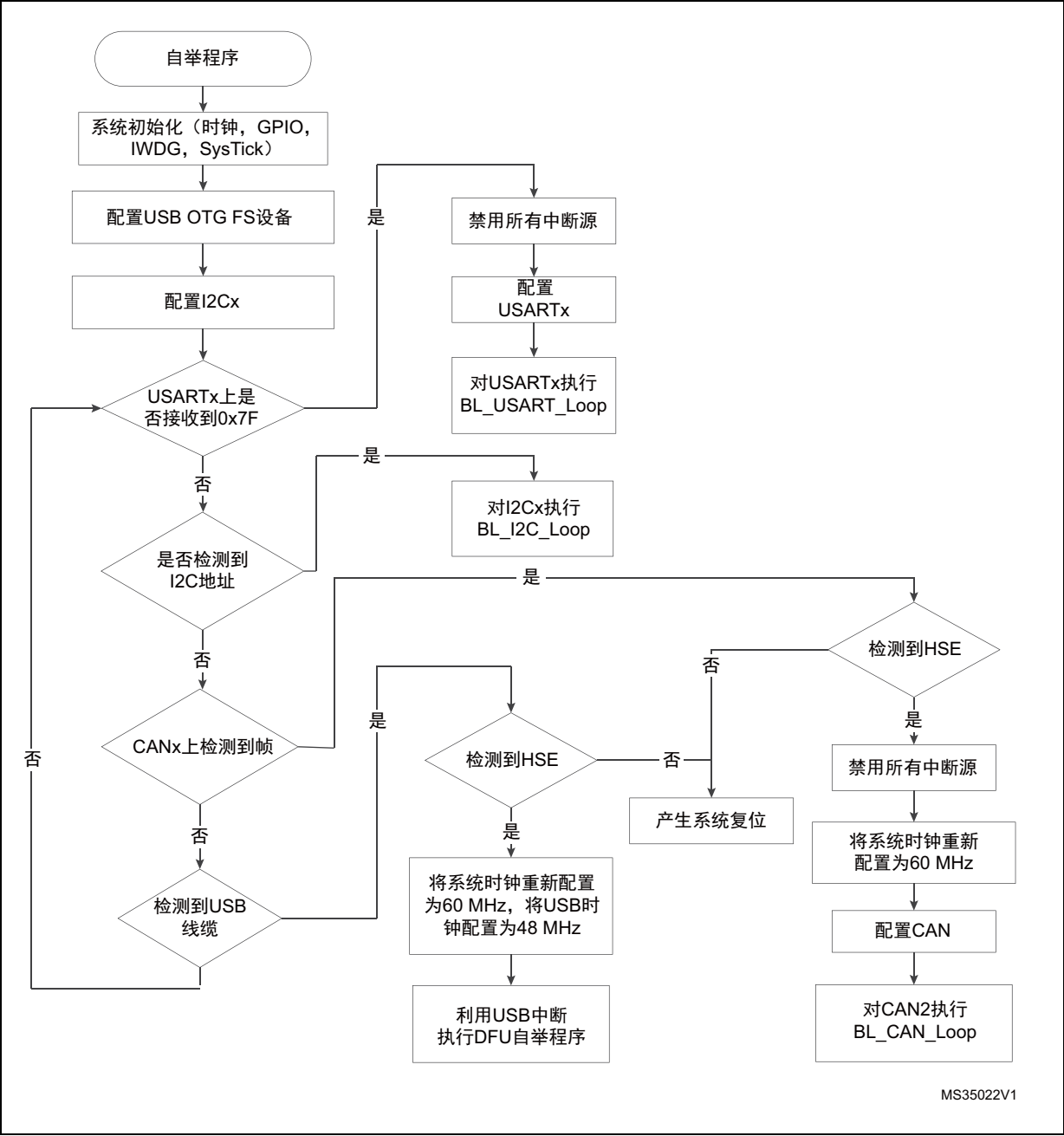


图 36.STM32F42xxx/43xxx 器件的自举程序 V7.x 选择



31.1.3 自举程序版本

下表列出了 STM32F42xxx/43xxx 器件自举程序 V7.x 版本。

表 65.STM32F42xxx/43xxx 自举程序 V7.x 版本

自举程序版本号	说明	已知限制
V7.0	初始自举程序版本	对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。

31.2 自举程序 V9.x

31.2.1 自举程序配置

STM32F42xxx/43xxx 自举程序通过应用模式 5 激活（详述参见[表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。



表 66. 系统存储器自举模式下 STM32F42xxx/43xxx 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 PLL 作为系统时钟，频率为 60 MHz。 HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 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	系统时钟频率为 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 12 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。

表 66. 系统存储器自举模式下 STM32F42xxx/43xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
CAN2 自举程序	CAN2	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b0111000x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。

表 66. 系统存储器自举模式下 STM32F42xxx/43xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PI3 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PI2 引脚：从数据输入线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PI1 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PI0 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI4 自举程序	SPI4	启用	SPI4 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI4_MOSI 引脚	输入	PE14 引脚：从数据输入线，用于推挽下拉模式
	SPI4_MISO 引脚	输出	PE13 引脚：从数据输出线，用于推挽下拉模式
	SPI4_SCK 引脚	输入	PE12 引脚：从时钟线，用于推挽下拉模式
	SPI4_NSS 引脚	输入	PE11 引脚：从芯片选择引脚，用于推挽下拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

31.2.2 自举程序选择

下图显示了自举程序选择机制。

图 37.STM32F42xxx/43xxx 自举程序 V9.x 的双存储区自举实现

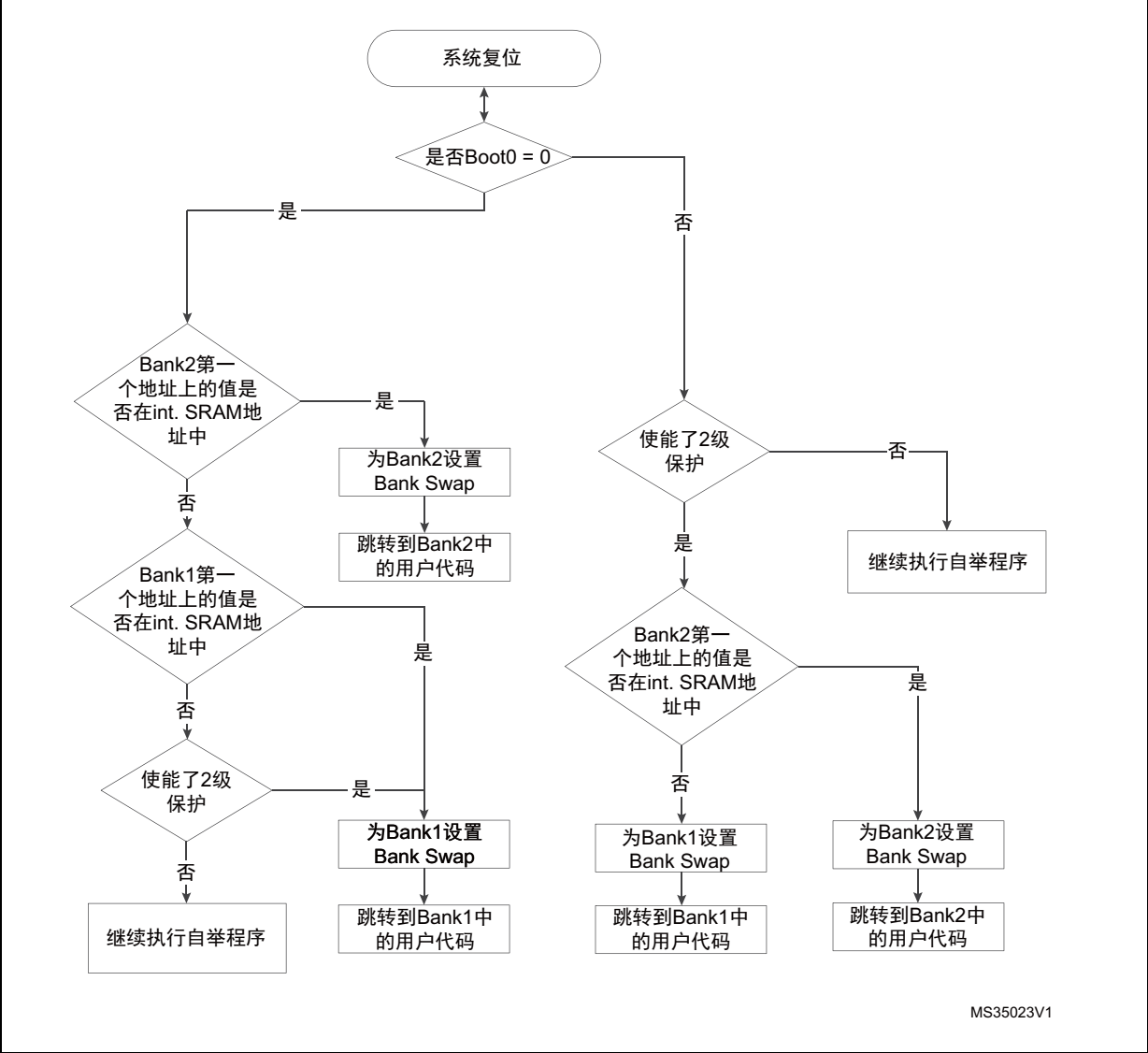
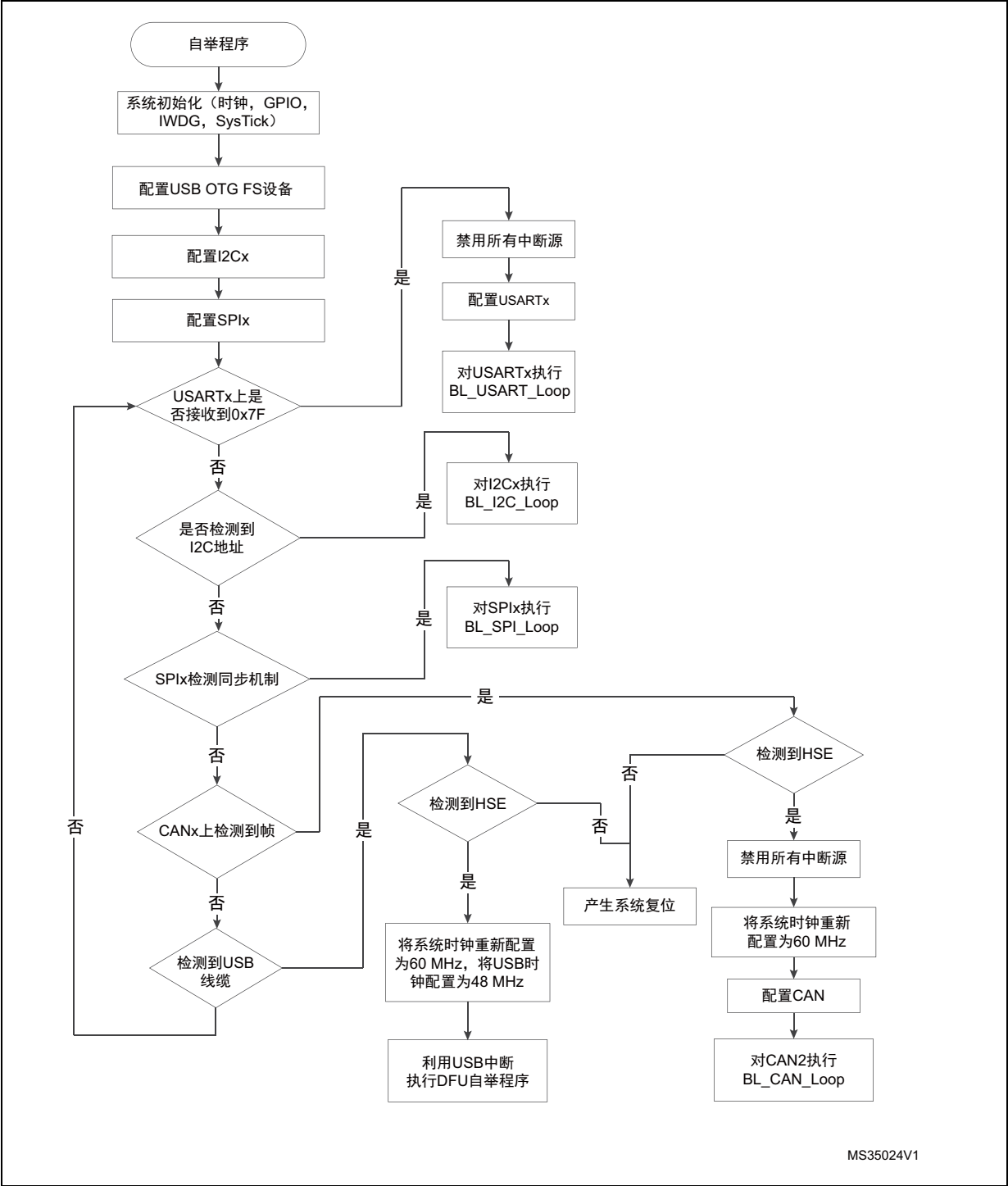


图 38.STM32F42xxx/43xxx 器件的自举程序 V9.x 选择



31.2.3 自举程序版本

下表列出了 STM32F42xxx/43xxx 器件自举程序 V9.x 版本。

表 67.STM32F42xxx/43xxx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	此自举程序是自举程序 v7.0 的更新版本。 该新版本自举程序支持 I2C2、I2C3、SPI1、SPI2 和 SPI4 接口。 此自举程序所用的 RAM 从 8KB 增加到了 12KB。 此自举程序的 ID 为 0x90 连接时间增加了。	无
V9.1	此自举程序是自举程序 v9.0 的更新版本。此新版本实现了新的 I2C No-stretch 命令（I2C 协议 v1.1），并且对于所有协议（USB、USART、CAN、I2C 和 SPI），能够在 RDP1 使能时利用 ReadOutUnprotect 命令禁用 PcROP。此自举程序的 ID 为 0x91	对于 CAN 接口，Write Unprotect 命令不起作用。作为替代，您可以使用 Write Memory 命令，并直接写入选项字节以禁用写保护。



32 STM32F446xx 器件自举程序

32.1 自举程序配置

STM32F446xx 自举程序通过应用模式 1 激活（详述参见[表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 68. 系统存储器自举模式下 STM32F446xx 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	对于配置为 60 MHz 的系统时钟，以及 USART、I2C 和 SPI 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE。在此情况下，使用 HSE 作为时钟源并将系统时钟配置为 60 MHz。HSE 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 12 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	-	电压范围为 [1.71 V, 3.6 V]。 在此范围内： - 3 个 Flash 等待周期。 - 系统时钟 60 MHz。 - 禁用预读取。 - Flash 写操作按字节执行（请参考自举程序存储器管理章节来获取更多信息）。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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

表 68. 系统存储器自举模式下 STM32F446xx 器件的配置 (续)

自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后, CAN2 的配置为: 波特率 125 kbps, 11 位标识符。 注意: 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b0111100x (其中 x = 0 对应写, x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚: 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚: 数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b0111100x (其中 x = 0 对应写, x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚: 时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚: 数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b0111100x (其中 x = 0 对应写, x = 1 对应读)
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚: 时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚: 数据线在开漏模式下使用。

表 68. 系统存储器自举模式下 STM32F446xx 器件的配置 (续)

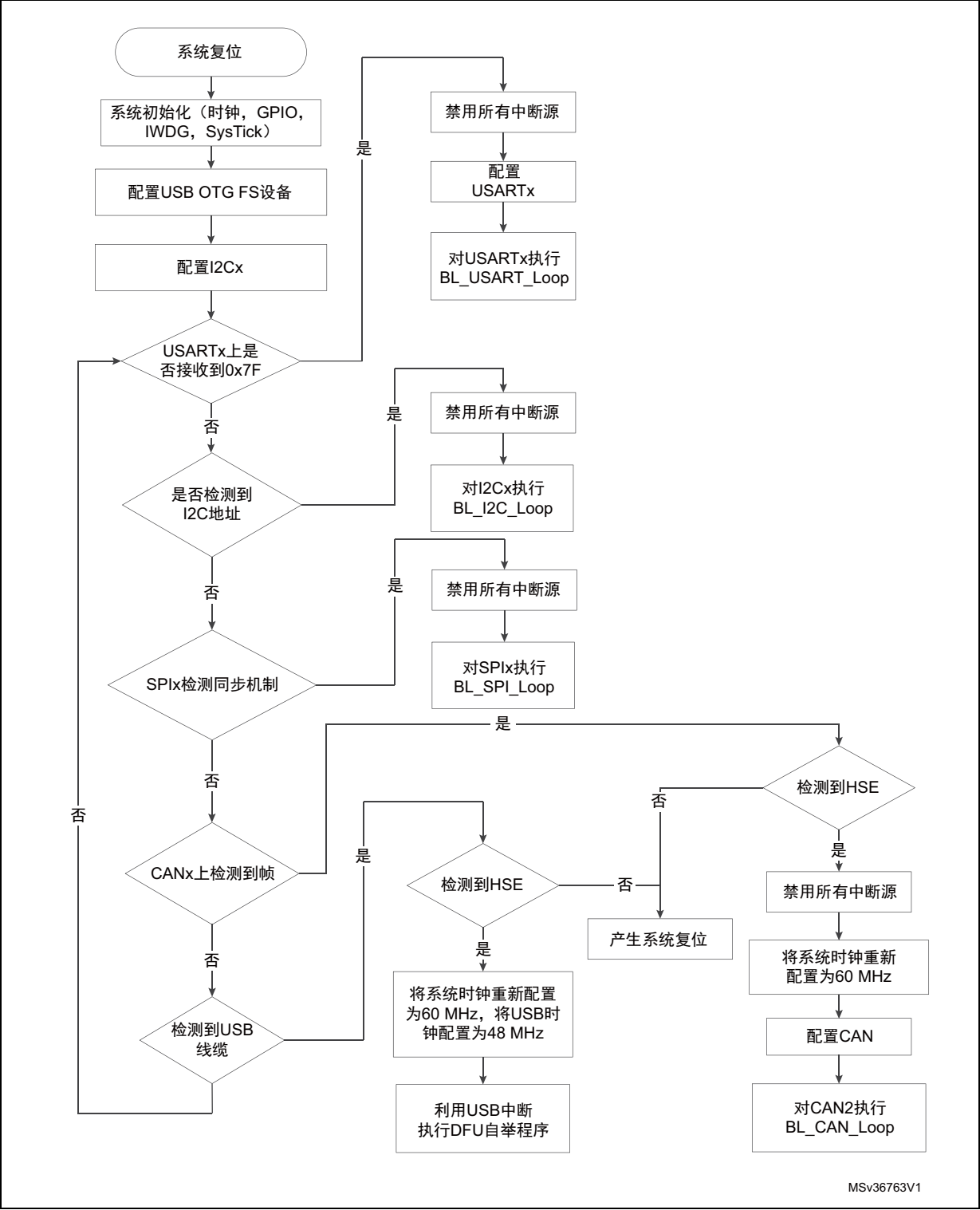
自举程序	功能 / 外设	状态	备注
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PC7 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI4 自举程序	SPI4	启用	SPI4 配置为： 从模式，全双工 (Full Duplex)，8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI4_MOSI 引脚	输入	PE14 引脚：从数据输入线，用于推挽下拉模式
	SPI4_MISO 引脚	输出	PE13 引脚：从数据输出线，用于推挽下拉模式
	SPI4_SCK 引脚	输入	PE12 引脚：从时钟线，用于推挽下拉模式
	SPI4_NSS 引脚	输入	PE11 引脚：从芯片选择引脚，用于推挽上拉模式。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM17	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx 和 I2Cx 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 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。

32.2 自举程序选择

下图显示了自举程序选择机制。

图 39.STM32F446xx 器件的自举程序 V9.x 选择



32.3 自举程序版本

下表列出了 STM32F446xx 器件自举程序 V9.x 版本：

表 69.STM32F446xx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	初始自举程序版本	无



33 STM32F469xx/479xx 器件自举程序

33.1 自举程序配置

STM32F469xx/479xx 自举程序通过应用模式 5 激活（详述参见表 2）。下表介绍了该自举程序使用的硬件资源。

表 70. 系统存储器自举模式下 STM32F469xx/479xx 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 PLL 作为系统时钟，频率为 60 MHz。HSI 时钟源用于启动阶段（接口检测阶段）和选择 USART、SPI 或 I2C 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	系统时钟频率为 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 12 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFF0000 起的 30431 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 [1.62 V, 2.1 V]。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。可使用自举程序命令在运行期间配置电压范围。
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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

表 70. 系统存储器自举模式下 STM32F469xx/479xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 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
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000100x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000100x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PF0 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF1 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000100x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚：数据线在开漏模式下使用。

表 70. 系统存储器自举模式下 STM32F469xx/479xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PI3 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PI2 引脚：从数据输入线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PI1 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PI0 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI4 自举程序	SPI4	启用	SPI4 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI4_MOSI 引脚	输入	PE14 引脚：从数据输入线，用于推挽下拉模式
	SPI4_MISO 引脚	输出	PE13 引脚：从数据输出线，用于推挽下拉模式
	SP4_SCK 引脚	输入	PE12 引脚：从时钟线，用于推挽下拉模式
	SPI4_NSS 引脚	输入	PE11 引脚：从芯片选择引脚，用于推挽上拉模式。
DFU 自举程序	USB	启用	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_DM 引脚	输入 / 输出	PA11 引脚：USB DM 线。
	USB_DP 引脚		PA12 引脚：USB DP 线。 无需外部上拉电阻。

对于 USARTx 和 I2Cx 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 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 48 MHz 之间）的外部时钟。

33.2 自举程序选择

图 40 显示了自举程序选择机制。

图 40.STM32F469xx/479xx 自举程序 V9.x 的双存储区自举实现

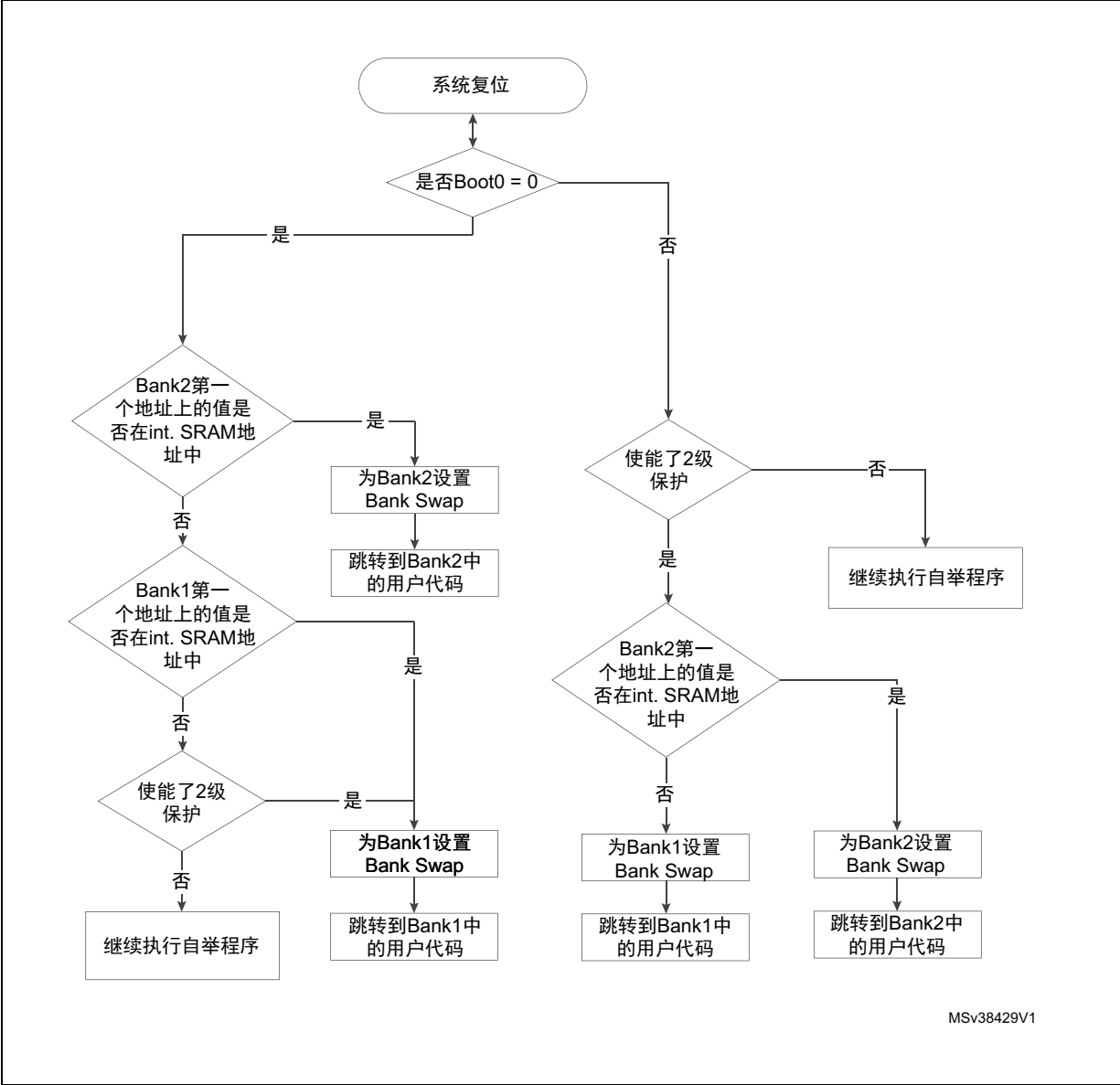
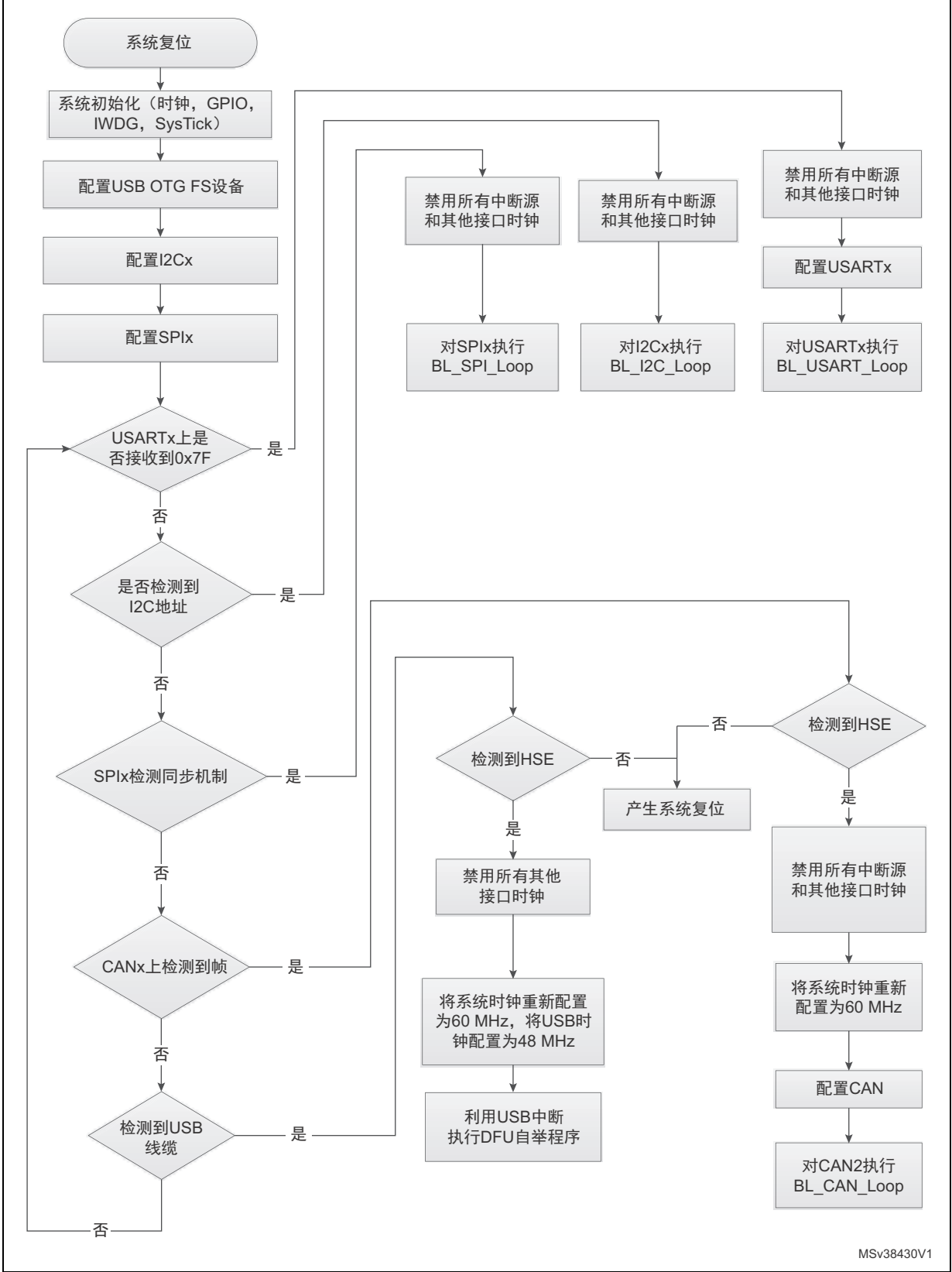


图 41.STM32F469xx/479xx 器件的自举程序 V9.x 选择



33.3 自举程序版本

下表列出了 STM32F469xx/479xx 器件 V9.x 自举程序版本：

表 71.STM32F469xx/479xx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	初始自举程序版本	无

34 STM32F74xxx/75xxx 器件自举程序

- STM32F74xxx/75xxx 器件可使用两种自举程序版本：
- V7.x 支持 USART1, USART3, CAN2, I2C1, I2C2, I2C3 和 DFU（USB FS 设备）。该版本嵌入 STM32F74xxx/75xxx rev.A 器件。
 - V9.x 支持 USART1, USART3, CAN2, I2C1, I2C2, I2C3, SPI1, SPI2, SPI4 和 DFU（USB FS 设备）。该版本嵌入 STM32F74xxx/75xxx rev.Z 器件。

34.1 自举程序 V7.x

34.1.1 自举程序配置

STM32F74xxx/75xxx 自举程序通过应用模式 8 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 72. 系统存储器自举模式下 STM32F74xxx/75xxx 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	对于配置为 60 MHz 的系统时钟，以及 USART 和 I2C 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE。在此情况下，使用 HSE 作为时钟源并将系统时钟配置为 60 MHz。 HSE 频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	时钟安全系统 (CSS) 中断 使能以用于 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 16 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 61440 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）。
USART1 自举程序	电源	-	电压范围为 [1.8V, 3.6V]。在此范围内： - 3 个 Flash 等待周期。 - 系统时钟 60 MHz。 - ART 加速器使能。 - Flash 写操作按字节执行（请参考自举程序存储器管理章节来获取更多信息）。
	USART1	启用	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

表 72. 系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART3 自举程序 (PB10/PB11)	USART3	启用	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
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后, CAN2 的配置为: 波特率 125 kbps, 11 位标识符。 注意: 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000101x (其中 x = 0 对应写, x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚: 时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚: 数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000101x (其中 x = 0 对应写, x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚: 时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚: 数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为: I2C 速度: 高达 400 KHz, 7 位地址, 从模式, 模拟滤波器开启。 7 位从地址: 0b1000101x (其中 x = 0 对应写, x = 1 对应读)
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚: 时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚: 数据线在开漏模式下使用。

表 72. 系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB DM 线。
	USB_DP 引脚		PA12 引脚：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

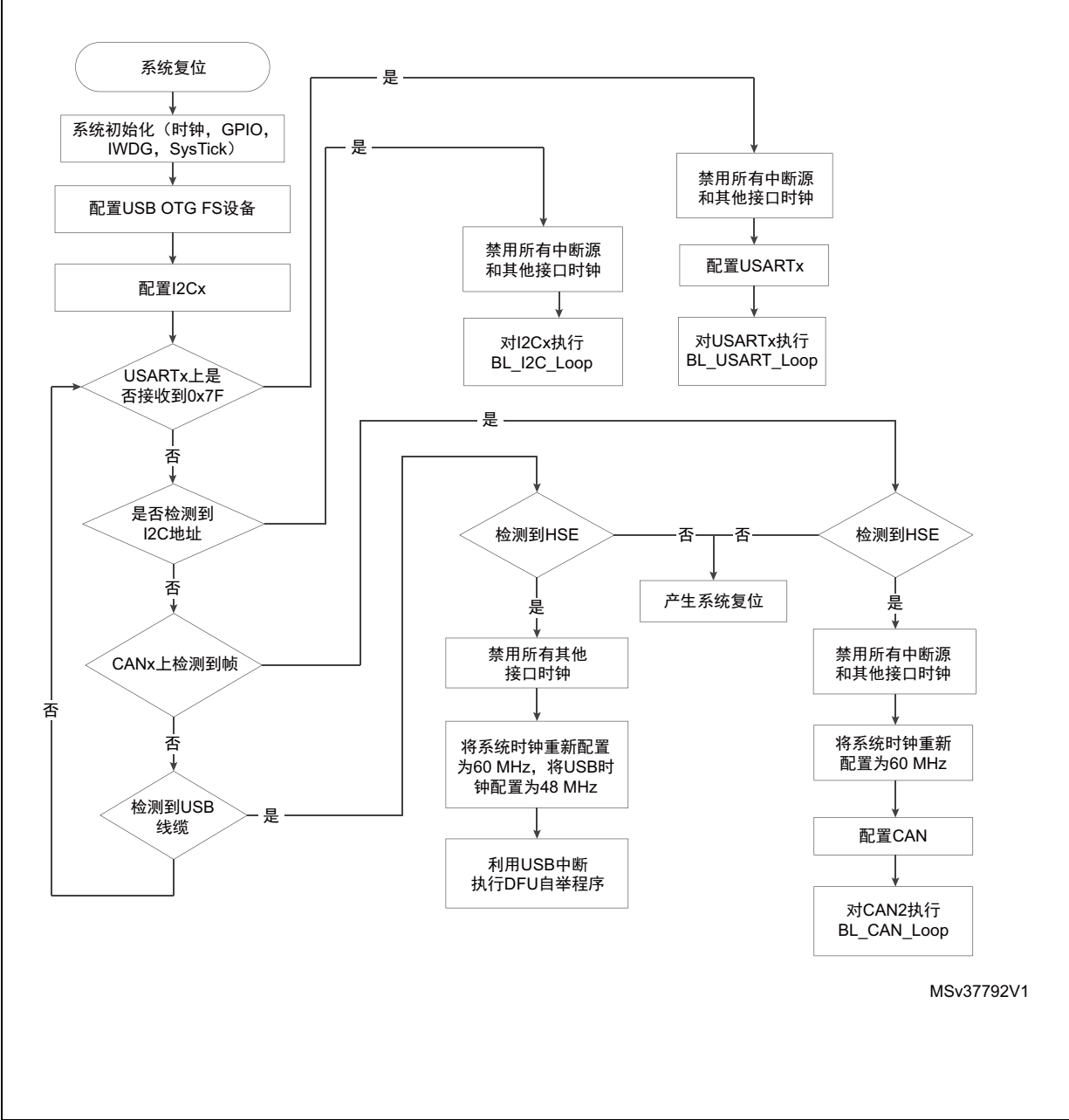
对于 USARTx 和 I2Cx 自举程序，系统时钟由嵌入式内部高速 RC 提供。该内部时钟同样用于 CAN 和 DFU（USB FS 设备），但仅限选择阶段。选择阶段结束后，执行 CAN 和 DFU 自举程序需要使用频率为 1 MHz 的倍数（介于 4 MHz 到 26 MHz 之间）的外部时钟。



34.1.2 自举程序选择

下图显示了自举程序选择机制。

图 42.STM32F74xxx/75xxx 器件的自举程序 V7.x 选择



MSv37792V1

34.1.3 自举程序版本

下表列出了 STM32F74xxx/75xxx 器件自举程序 V7.x 版本：

表 73. STM32F74xxx/75xxx 自举程序 V7.x 版本

自举程序版本号	说明	已知限制
V7.0	初始自举程序版本	无

34.2 自举程序 V9.x

34.2.1 自举程序配置

STM32F74xxx/75xxx 自举程序通过应用模式 8 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 74. 系统存储器自举模式下 STM32F74xxx/75xxx 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	对于配置为 60MHz 的系统时钟，以及 USART、I2C 和 SPI 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	仅当选择 CAN 或 DFU（USB FS 设备）接口时才使用 HSE。在此情况下，使用 HSE 作为时钟源并将系统时钟配置为 60 MHz。HSE 频率必须为 1 MHz 的倍数，且范围介于 4 MHz 到 26 MHz 之间。
		-	时钟安全系统 (CSS) 中断使能以用于 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 16 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 61440 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	-	电压范围为 [1.8V, 3.6V] 在此范围内： - 3 个 Flash 等待周期。 - 系统时钟 60MHz。 - ART 加速器使能。 - Flash 写操作按字节执行（请参考自举程序存储器管理章节来获取更多信息）。

表 74. 系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART1 自举程序	USART1	启用	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
USART3 自举程序 (PB10/PB11)	USART3	启用	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
USART3 自举程序 (PC10/PC11)	USART3	启用	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
USART1 和 USART3 自举程序	SysTick 定时器	启用	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	启用	初始化后，CAN2 的配置为：波特率 125 kbps，11 位标识符。 注意： 由于 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	PB5 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
I2C1 自举程序	I2C1	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000101x (其中 x = 0 对应写，x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB9 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000101x (其中 x = 0 对应写，x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PF1 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PF0 引脚：数据线在开漏模式下使用。

表 74. 系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000101x (其中 x = 0 对应写，x = 1 对应读)
	I2C3_SCL 引脚	输入 / 输出	PA8 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC9 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PI3 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PI2 引脚：从数据输入线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PI1 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PI0 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI4 自举程序	SPI4	启用	SPI4 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI4_MOSI 引脚	输入	PE14 引脚：从数据输入线，用于推挽下拉模式
	SPI4_MISO 引脚	输出	PE13 引脚：从数据输出线，用于推挽下拉模式
	SPI4_SCK 引脚	输入	PE12 引脚：从时钟线，用于推挽下拉模式
	SPI4_NSS 引脚	输入	PE11 引脚：从芯片选择引脚，用于推挽上拉模式。

表 74. 系统存储器自举模式下 STM32F74xxx/75xxx 器件的配置（续）

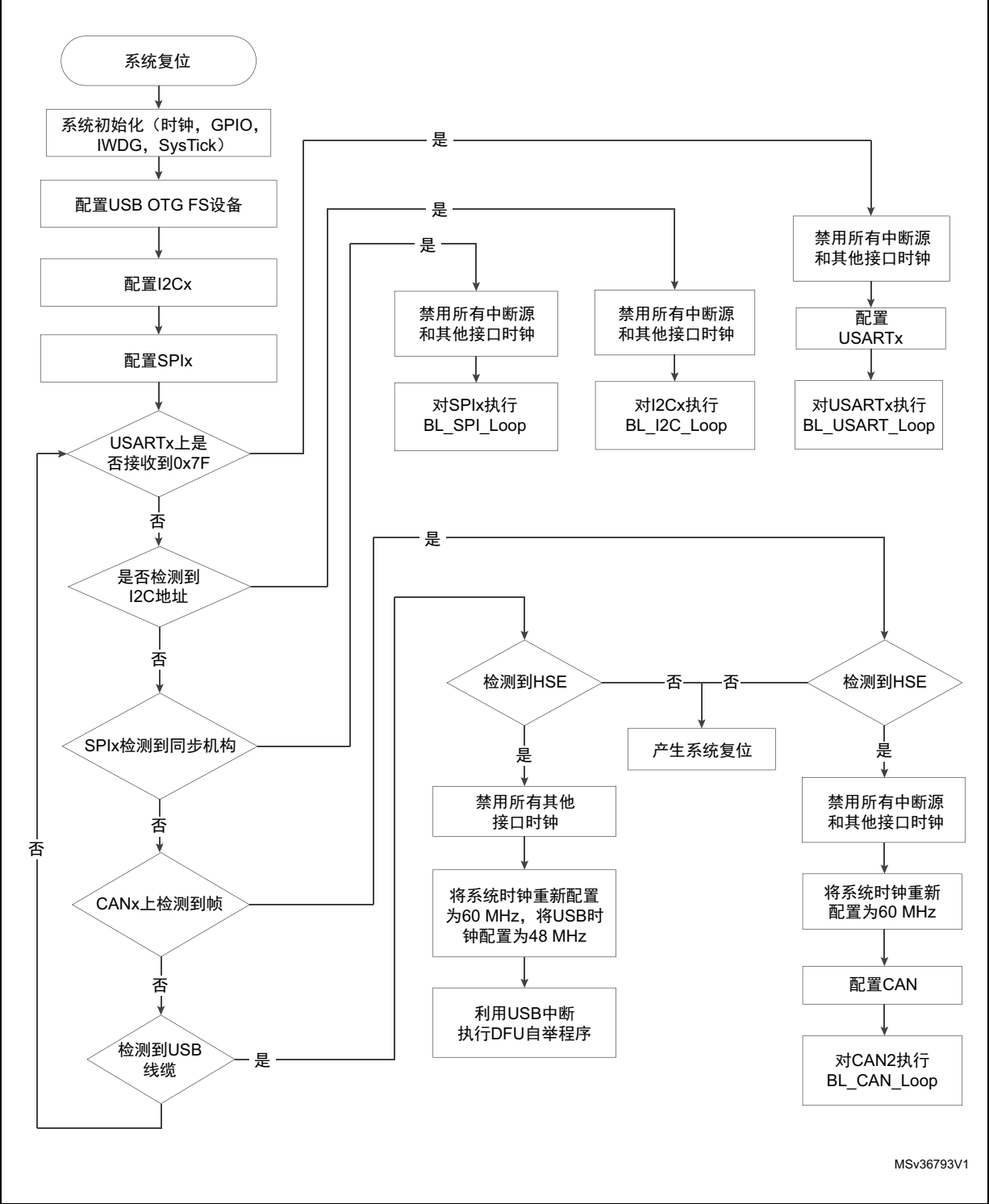
自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB DM 线。
	USB_DP 引脚		PA12 引脚：USB DP 线 无需外部上拉电阻。
CAN2 和 DFU 自举程序	TIM11	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 60 MHz。

对于 USARTx、I2Cx 和 SPIx 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 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。

34.2.2 自举程序选择

图 43 显示了自举程序选择机制。

图 43.STM32F74xxx/75xxx 器件的自举程序 V9.x 选择



34.2.3 自举程序版本

下表列出了 STM32F74xxx/75xxx 器件自举程序 V9.x 版本：

表 75.STM32F74xxx/75xxx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	初始自举程序版本	无

35 STM32L031xx/041xx 器件自举程序

35.1 自举程序配置

STM32L031xx/041xx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 76. 系统存储器自举模式下 STM32L031xx/041xx 器件的配置

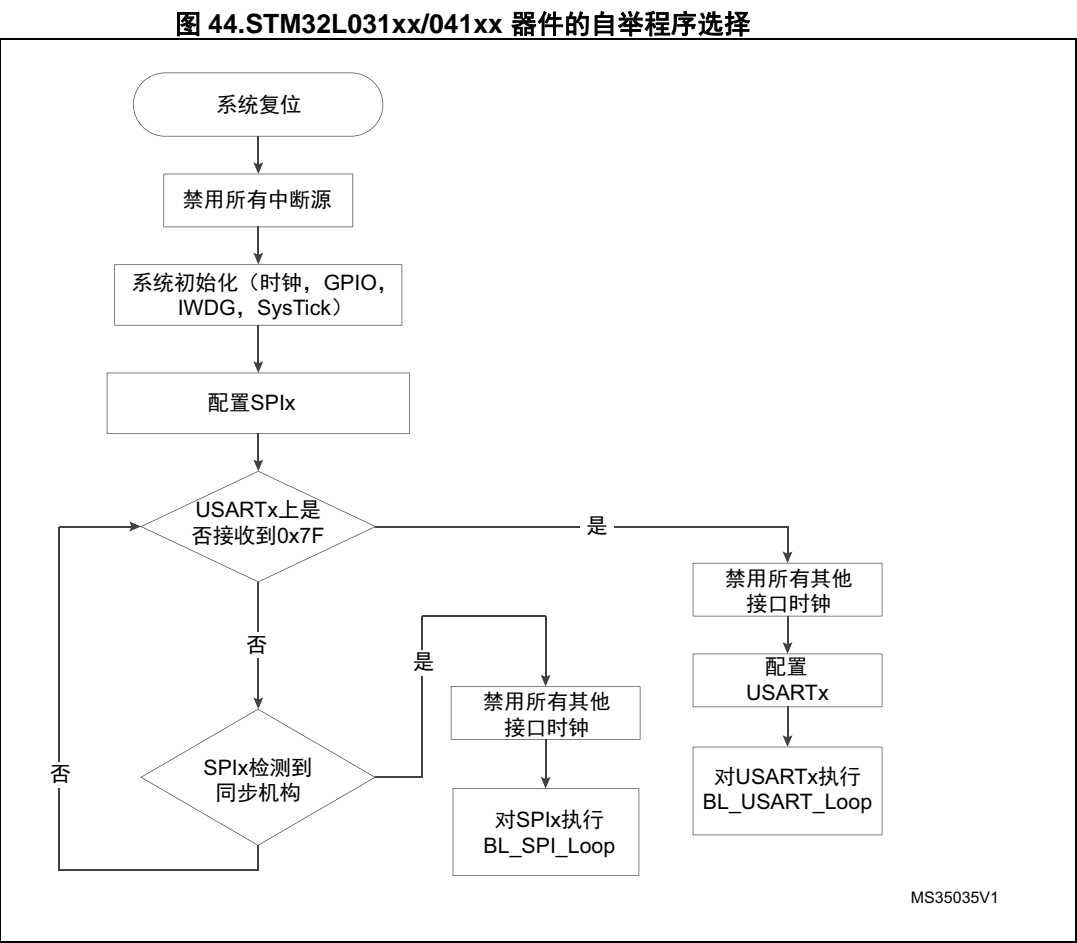
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 32 MHz，使用 HSI 16 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 4 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 4 KB 空间包含自举程序固件
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART2 自举程序 (PA9/PA10)	USART2	启用	初始化后，USART2 的配置为：8 位，偶校验位和 1 个停止位。
	USART2_RX 引脚	输入	PA10 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA9 引脚：USART2 处于发送模式
USART2 自举程序 (PA2/PA3)	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。 注：如果 SPI 主设备不使用 IO，则此 IO 可连接到 GND。

对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。

对于该产品，自举程序 Read/Write 命令不支持 SRAM 存储空间。

35.2 自举程序选择

图 44 显示了自举程序选择机制。



35.3 自举程序版本

表 77 列出了 STM32L031xx/041xx 器件自举程序版本：

表 77.STM32L031xx/041xx 自举程序版本

自举程序版本号	说明	已知限制
V12.0	初始自举程序版本	无

36 STM32L05xxx/06xxx 器件自举程序

36.1 自举程序配置

STM32L05xxx/06xxx 自举程序通过应用模式 1 激活（详述参见表 2：自举程序激活模式）。下表介绍了该自举程序使用的硬件资源。

表 78. 系统存储器自举模式下 STM32L05xxx/06xxx 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	系统时钟频率为 32 MHz，使用 HSI 16 MHz 为时钟源。
	电源	-	电压范围设置为电压范围 1。
	RAM	-	自地址 0x20000000 起的 4 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）。
USART1 自举程序	USART1	启用	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
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。

表 78. 系统存储器自举模式下 STM32L05xxx/06xxx 器件的配置（续）

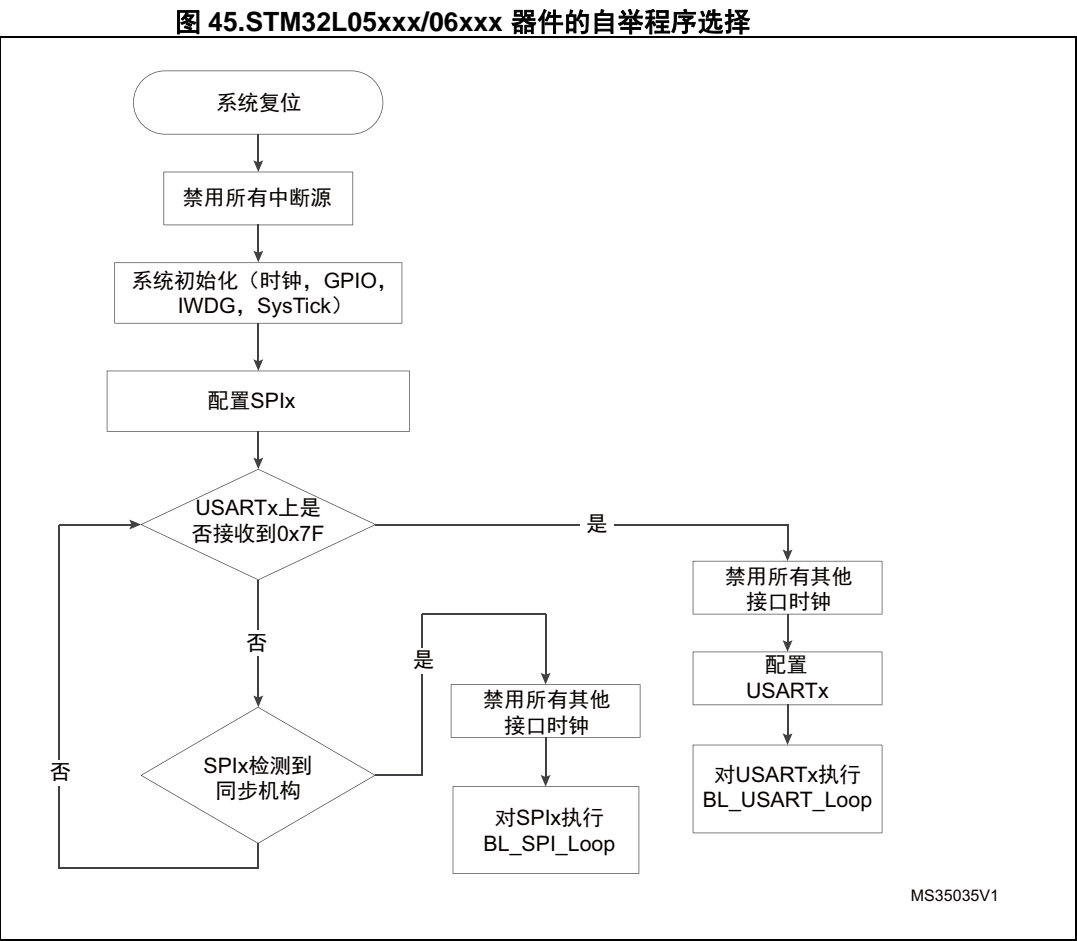
自举程序	功能 / 外设	状态	备注
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽下拉模式。

对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。



36.2 自举程序选择

下图显示了自举程序选择机制。



36.3 自举程序版本

下表列出了 STM32L05xxx/06xxx 器件自举程序版本：

表 79.STM32L05xxx/06xxx 自举程序版本

自举程序版本号	说明	已知限制
V12.0	初始自举程序版本	无

37 STM32L07xxx/08xxx 器件自举程序

STM32L07xxx/08xxx 器件可使用两种自举程序版本：

- V4.x 支持 USART1， USART2 和 DFU（USB FS 设备）。
该版本嵌入 STM32L072xx/73xx 和 STM32L082xx/83xx 器件。
- V11.x 支持 USART1， USART2， I2C1， I2C2， SPI1 和 SPI2。
此版本内置在其他 STM32L071xx/081xx 器件。

37.1 自举程序 V4.x

37.1.1 自举程序配置

STM32L07xxx/08xxx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。
[表 80](#) 显示了该自举程序使用的硬件资源。

表 80. 系统存储器自举模式下 STM32L07xxx/08xxx 器件的配置

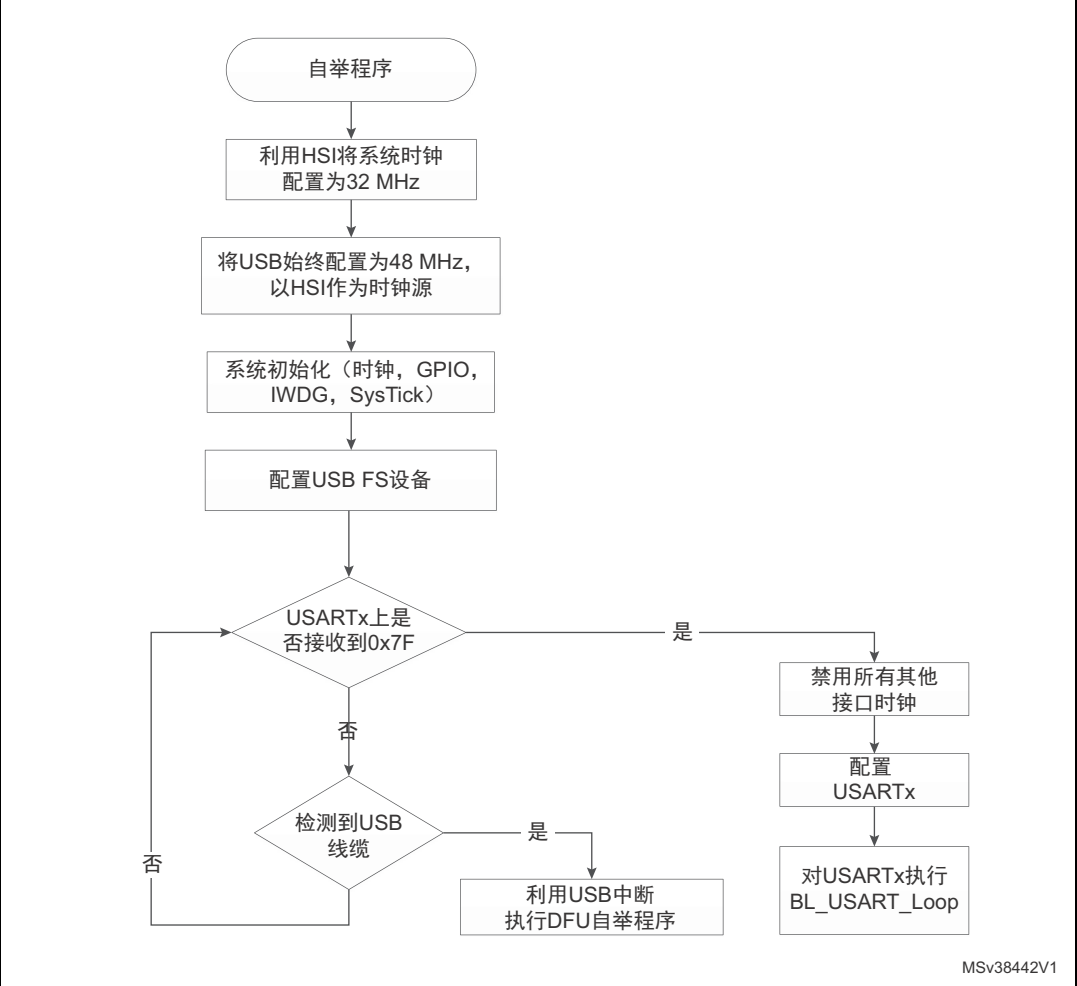
自举程序	功能 / 外设	状态	备注
适用于所有 自举程序	RCC	HSI 使能	系统时钟频率为 32 MHz，使用 HSI 16 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 4 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FF00000 起的 8 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
USART1 自举程序	USART1	启用	初始化后， USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 引脚	输入	PA10 引脚： USART2 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚： USART2 处于发送模式
USART2 自举程序	USART2	启用	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	PA3 引脚： USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚： USART2 处于发送模式
USARTx 自举程序	SysTick 定时器	启用	用于自动检测用于 USARTx 自举程序的主机串口波特率。
DFU 自举程序	USB	启用	USB FS 配置为强制器件模式。使能 USB FS 中断向量以用于 USB DFU 通信。
	USB_DM 引脚	输入 / 输出	PA11 引脚： USB FS DM 线
	USB_DP 引脚		PA12 引脚： USB FS DP 线 无需外部上拉电阻。

对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。

37.1.2 自举程序选择

图 46 显示了自举程序选择机制。

图 46.STM32L07xxx/08xxx 器件的自举程序 V4.x 选择



37.1.3 自举程序版本

表 81 列出了 STM32L07xxx/08xxx 器件自举程序版本：

表 81.STM32L07xxx/08xxx 自举程序 V4.x 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	无

37.2 自举程序 V11.x

37.2.1 自举程序配置

STM32L07xxx/08xxx 自举程序通过应用模式 2 激活（详述参见 [表 2：自举程序激活模式](#)）。
[表 82](#) 显示了该自举程序使用的硬件资源。

表 82. 系统存储器自举模式下 STM32L07xxx/08xxx 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	系统时钟频率为 32 MHz，使用 HSI 16 MHz 为时钟源。
	RAM	-	自地址 0x20000000 起的 5 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FF00000 起的 8 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	启用	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：USART2 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART2 处于发送模式
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USARTx 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000010x （其中 x = 0 对应写，x = 1 对应读）
	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。

表 82. 系统存储器自举模式下 STM32L07xxx/08xxx 器件的配置（续）

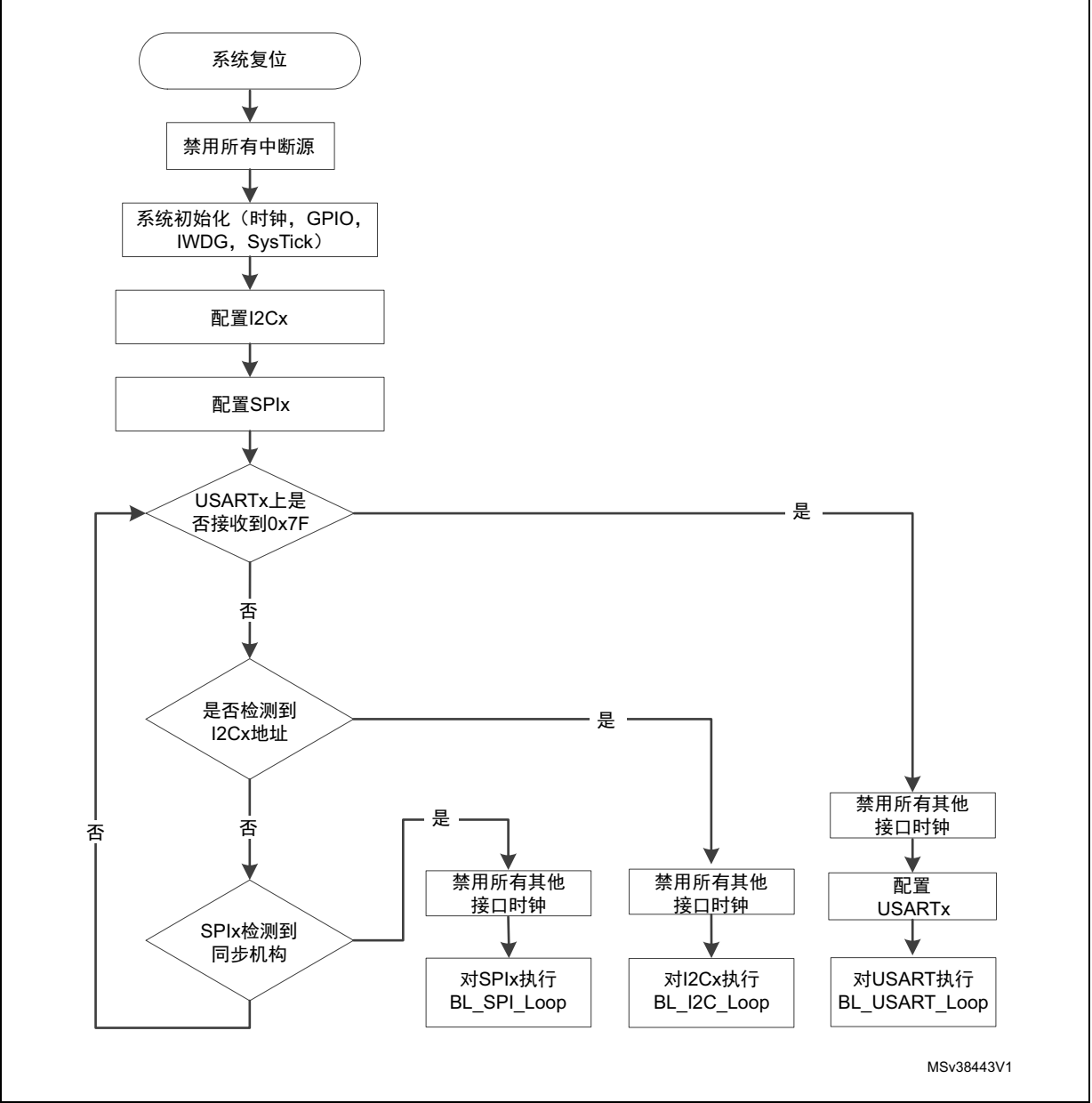
自举程序	功能 / 外设	状态	备注
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000010x (其中 x = 0 对应写，x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：I2C2 时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB11 引脚：I2C2 数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽上拉模式。 注： 如果 SPI 主设备不适用 IO，则此 IO 可连接到 Gnd。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽上拉模式。 注： 如果 SPI 主设备不适用 IO，则此 IO 可连接到 GND。

对于所有自举程序接口，系统时钟由内部高速 RC 提供。因此，执行自举程序时无需外部石英时钟。

37.2.2 自举程序选择

图 47 显示了自举程序选择机制。

图 47.STM32L07xxx/08xxx 器件的自举程序 V11.x 选择



37.2.3 自举程序版本

下表列出了 STM32L07xxx/08xxx 器件自举程序版本：

表 83.STM32L07xxx/08xxx 自举程序 V11.x 版本

自举程序版本号	说明	已知限制
V11.0	废弃版本（不用）	无
V11.1	初始自举程序版本	无



38 STM32L1xxx6(8/B)A 器件自举程序

38.1 自举程序配置

STM32L1xxx6(8/B)A 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 84. 系统存储器自举模式下 STM32L1xxx6(8/B)A 器件的配置

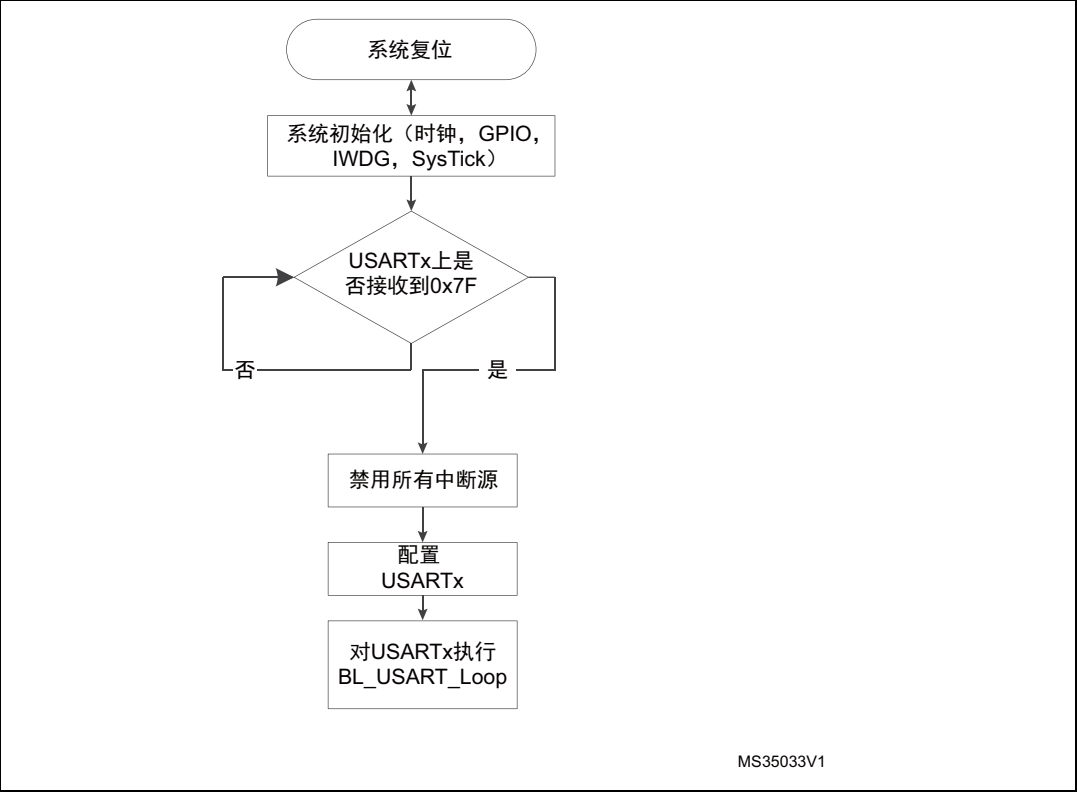
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 16 MHz。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FF00000 起的 4 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为电压范围 1。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 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
USART2 自举程序	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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，自举程序执行不需要外部石英时钟。

38.2 自举程序选择

下图显示了自举程序选择机制。

图 48.STM32L1xxx6(8/B)A 器件的自举程序选择



38.3 自举程序版本

下表列出了 STM32L1xxx6(8/B)A 器件自举程序版本：

表 85.STM32L1xxx6(8/B)A 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，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）。



39 STM32L1xxx6(8/B) 器件自举程序

39.1 自举程序配置

STM32L1xxx6(8/B) 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 86. 系统存储器自举模式下 STM32L1xxx6(8/B) 器件的配置

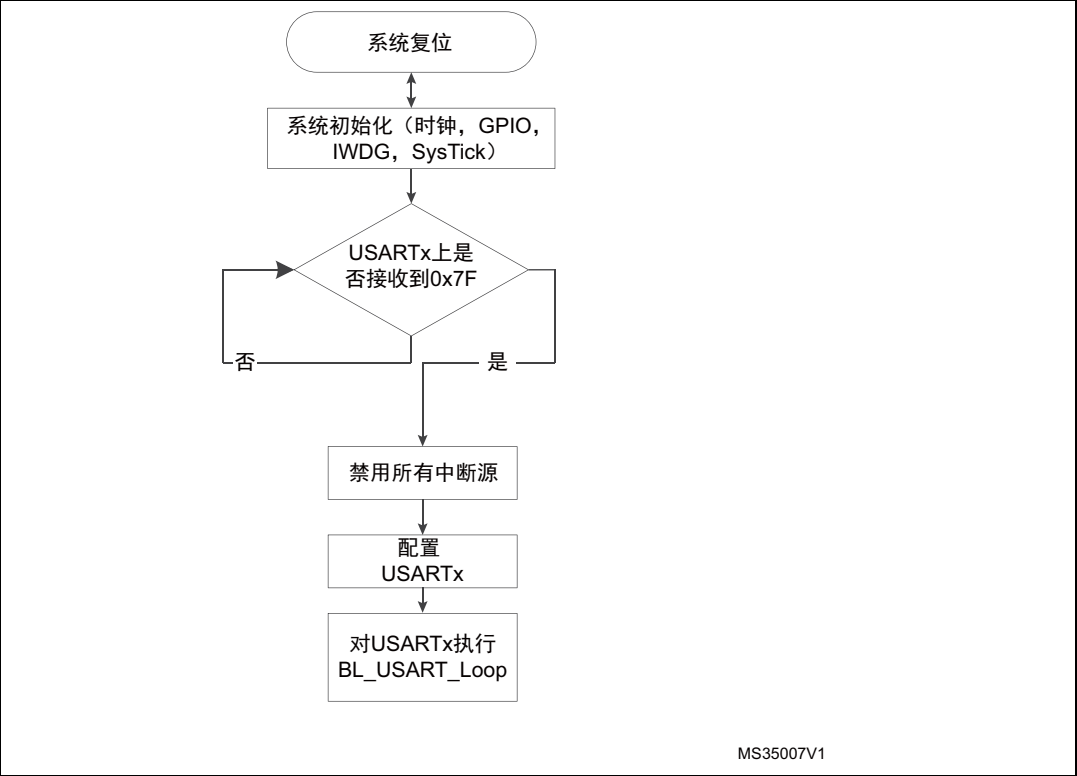
自举程序	功能 / 外设	状态	备注
适用于所有自举程序	RCC	HSI 使能	系统时钟频率为 16 MHz。
	RAM	-	自地址 0x20000000 起的 2 KB 空间供自举程序固件使用。
	系统存储器	-	自地址 0x1FF00000 起的 4 KB 空间包含自举程序固件。
	IWDG	-	独立看门狗 (IWDG) 预分频器配置为最大值，并定期进行刷新以防止看门狗复位（如果用户事先使能硬件 IWDG 选项）。
	电源	-	电压范围设置为电压范围 1。
USART1 自举程序	USART1	启用	初始化后，USART1 的配置为：8 位，偶校验位和 1 个停止位。
	USART1_RX 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
USART2 自举程序	USART2	启用	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 处于接收模式
	USART2_TX 引脚	输出	PD5 引脚：USART2 处于发送模式
USART1 和 USART2 自举程序	SysTick 定时器	启用	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，自举程序执行不需要外部石英时钟。

39.2 自举程序选择

下图显示了自举程序选择机制。

图 49.STM32L1xxx6(8/B) 器件的自举程序选择



39.3 自举程序版本

下表列出了 STM32L1xxx6(8/B) 器件自举程序版本：

表 87.STM32L1xxx6(8/B) 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，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）。



40 STM32L1xxxC 器件自举程序

40.1 自举程序配置

STM32L1xxxC 自举程序通过应用模式 1 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 88. 系统存储器自举模式下 STM32L1xxxC 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，时钟源由外部石英时钟提供）。
		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 0x20000000 起的 4 个字节空间供自举程序固件使用。
	USART1	启用	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 和 USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	SysTick 定时器	启用	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	启用	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

表 88. 系统存储器自举模式下 STM32L1xxC 器件的配置（续）

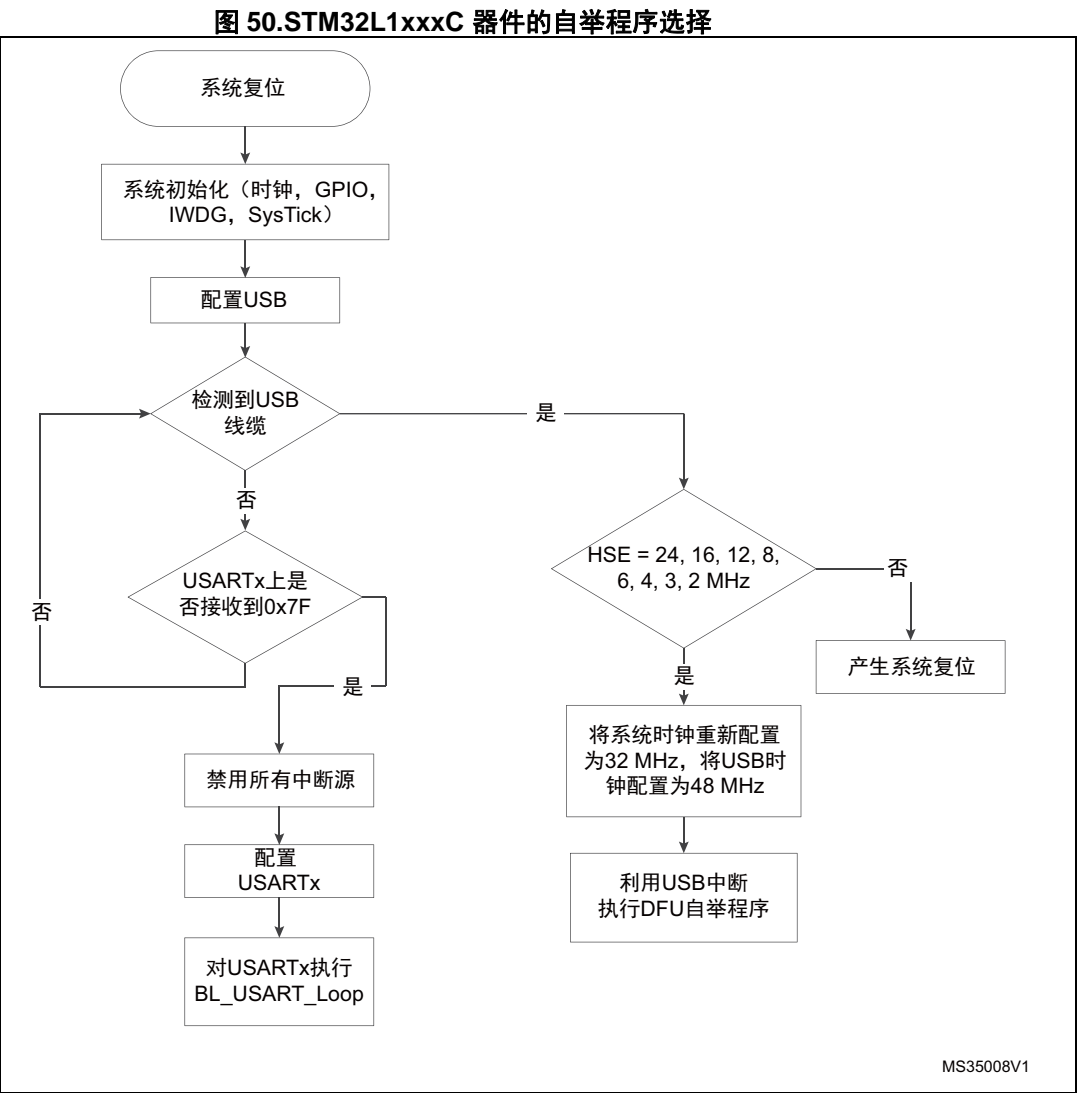
自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11: USB DM 线。
	USB_DP 引脚		PA12: USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

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。



40.2 自举程序选择

下图显示了自举程序选择机制。



40.3 自举程序版本

下表列出了 STM32L1xxxC 器件自举程序版本：

表 89.STM32L1xxxC 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。

41 STM32L1xxxD 器件自举程序

41.1 自举程序配置

STM32L1xxxD 自举程序通过应用模式 4 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 90. 系统存储器自举模式下 STM32L1xxxD 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，时钟源将由外部石英时钟提供）。
		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 0x20000000 起的 4 个字节空间供自举程序固件使用。
USART1 自举程序	USART1	启用	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
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	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

表 90. 系统存储器自举模式下 STM32L1xxxD 器件的配置（续）

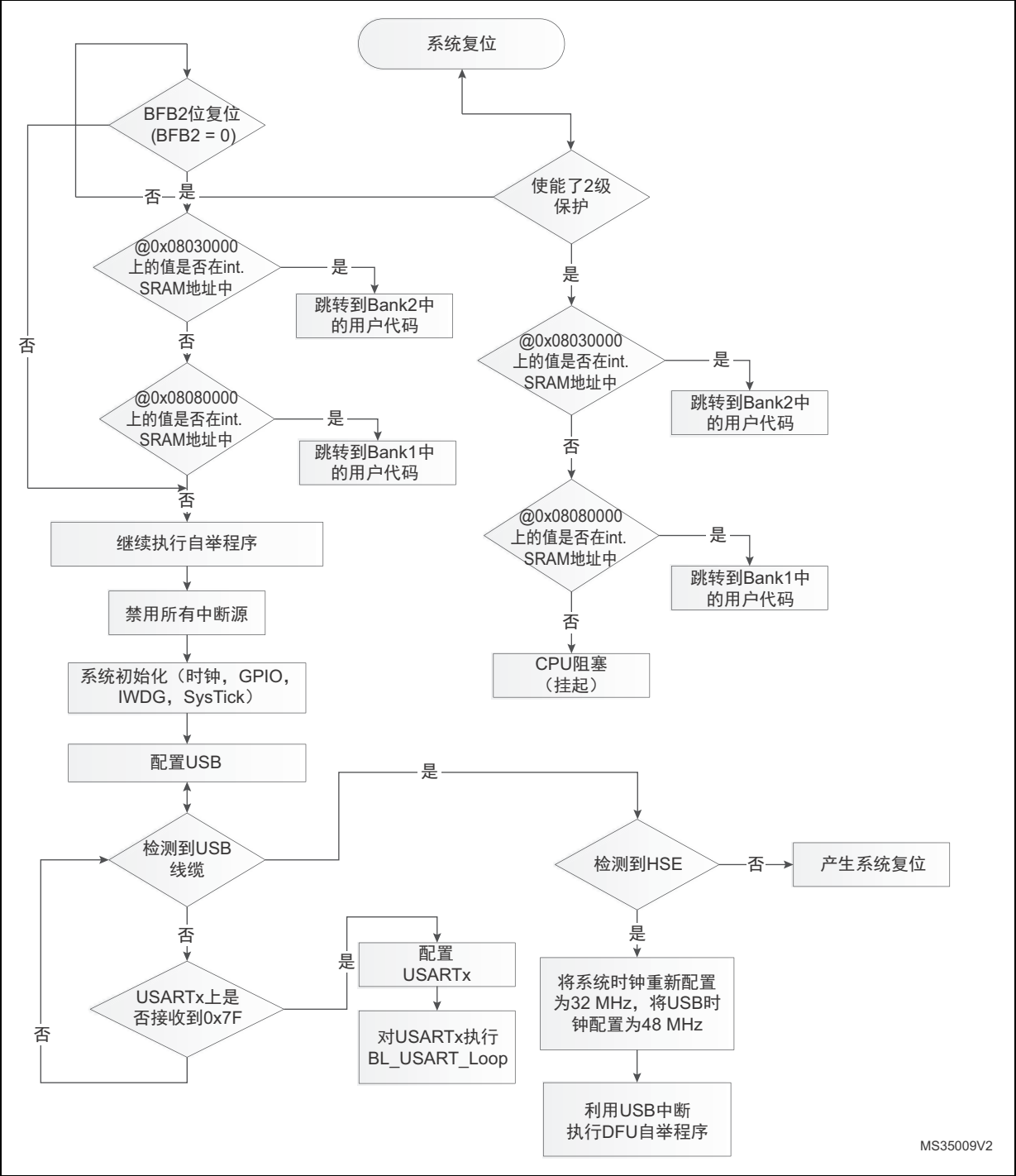
自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11: USB DM 线。
	USB_DP 引脚		PA12: USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

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。

41.2 自举程序选择

下图显示了自举程序选择机制。

图 51.STM32L1xxxD 器件的自举程序选择



41.3 自举程序版本

下表列出了 STM32L1xxD 器件自举程序版本：

表 91.STM32L1xxD 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。注意：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）。

42 STM32L1xxxE 器件自举程序

42.1 自举程序配置

STM32L1xxxE 自举程序通过应用模式 4 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 92. 系统存储器自举模式下 STM32L1xxxE 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，时钟源将由外部石英时钟提供）。
		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 0x20000000 起的 4 个字节空间供自举程序固件使用。
USART1 自举程序	USART1	启用	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
USART1 和 USART2 自举程序	SysTick 定时器	启用	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	启用	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

表 92. 系统存储器自举模式下 STM32L1xxxE 器件的配置（续）

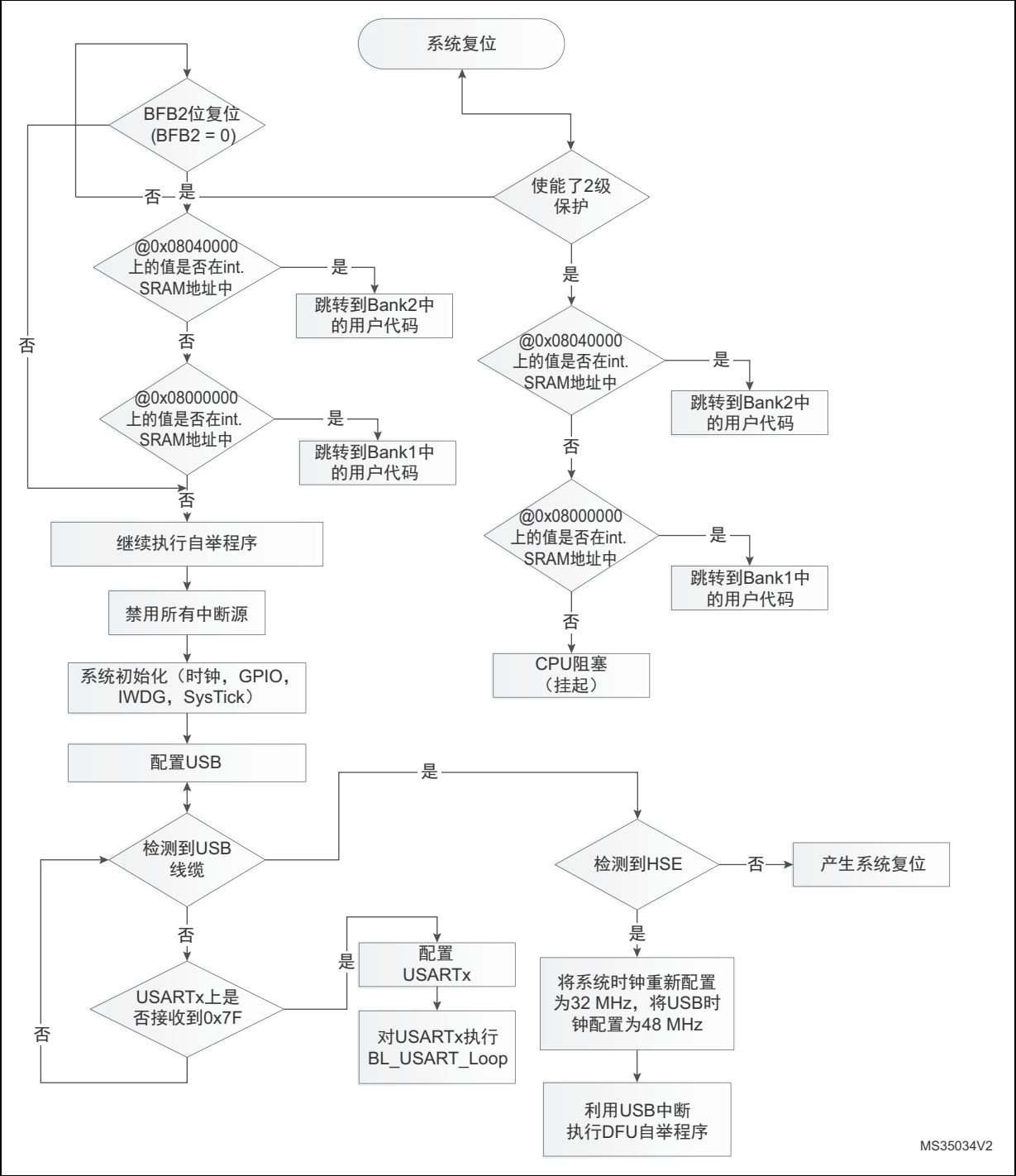
自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	使用 USB 全速模式
	USB_DM 引脚	输入 / 输出	PA11: USB DM 线。
	USB_DP 引脚		PA12: USB DP 线 必须连接一个 1.5 KOhm 的外部上拉电阻到 USB_DP 引脚。

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。

42.2 自举程序选择

下图显示了自举程序选择机制。

图 52.STM32L1xxxE 器件的自举程序选择



MS35034V2

42.3 自举程序版本

下表列出了 STM32L1xxxE 器件自举程序版本：

表 93.STM32L1xxxE 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）。

43 STM32L476xx/486xx 器件自举程序

STM32L476xx/486xx 器件可使用两种自举程序版本：

- V10.x 支持 USART1、USART2、I2C1、I2C2 和 DFU（USB FS 设备）。该版本嵌入 STM32L476xx/486xx rev.A, rev.Z 和 rev.B 器件。
- V9.x 支持 USART1、USART2、I2C1、I2C2、SPI1 和 SPI2，CAN1 和 DFU（USB FS 设备）。该版本嵌入 STM32L476xx/486xx rev.Y 器件。

43.1 自举程序 V10.x

43.1.1 自举程序配置

STM32L476xx/486xx 自举程序通过应用模式 7 激活（详述参见 [表 2：自举程序激活模式](#)）。下表介绍了该自举程序使用的硬件资源。

表 94. 系统存储器自举模式下 STM32L476xx/486xx 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	对于配置为 72 MHz 的系统时钟，以及 USART 和 I2C 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	HSE 仅在选择了 USB 接口且无 LSE 时使用。HSE 必须是下列值其中之一：[24,20,18,16,12,9,8,6,4] MHz。
		LSE 已使能	LSE 用来微调 MSI，MSI 配置为 48 MHz，为 USB 时钟源。LSE 必须等于 32,768 KHz。如果未检测到 LSE，则若连接了 USB，可使用 HSE 来代替。
		MSI 已使能	MSI 配置为 48 MHz，且用作 USB 时钟源。仅当检测到 LSE 时才使用 MSI，否则（未检测到 LSE），若连接了 USB 则使用 HSE。
		-	当 LSE 或 HSE 使能时，使能时钟安全系统 (CSS) 中断。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 12 KB 空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFF0000 起的 28672 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）。
USART1 自举程序	电源	-	如果选择了电压调节范围 2，则 DFU 不能用于与自举程序之间的通信。自举程序固件不能配置 PWR_CR1 寄存器中的电压调节范围值。
	USART1	启用	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
USART2 自举程序	USART1_TX 引脚	输出	PA9 引脚：USART1 处于发送模式
	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式

表 94. 系统存储器自举模式下 STM32L476xx/486xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USART3 自举程序	USART3	启用	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
USARTx 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000011x (其中 x = 0 对应写，x = 1 对应读)
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址：0b1000011x (其中 x = 0 对应写，x = 1 对应读)
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB11 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。 7 位从地址为 0b1000011x (其中 x = 0 对应写，x = 1 对应读)
	I2C3_SCL 引脚	输入 / 输出	PC0 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC1 引脚：数据线在开漏模式下使用。
DFU 自举程序	USB	启用	USB OTG FS 配置为强制器件模式
	USB_DM 引脚	输入 / 输出	PA11：USB DM 线。
	USB_DP 引脚		PA12：USB DP 线 无需外部上拉电阻。
	TIM17	启用	该定时器用来确定 HSE 的值。一旦确定了 HSE 频率，可利用 PLL 和 HSE 将系统时钟配置为 72 MHz。

对于 USARTx 和 I2Cx 自举程序，无需外部时钟。

USB 自举程序（DFU）需要 LSE（低速外部时钟）或 HSE（高速外部时钟）：

- 例如，LSE 存在时，无论 HSE 是否存在，MSI 都会由 LSE 进行配置和调节，可提供一个等于 48 MHz 的精确时钟，该频率是 USB 的时钟源。系统时钟以 24 MHz 的 HSI 为时钟。
- 假如存在 HSE，则系统时钟和 USB 时钟会分别配置为 24 MHz 和 48 MHz，以 HSE 作为时钟源。

43.1.2 自举程序选择

图 53 和图 54 显示了自举程序选择机制。

图 53. STM32L476xx/486xx 自举程序 V10.x 的双存储区自举实现

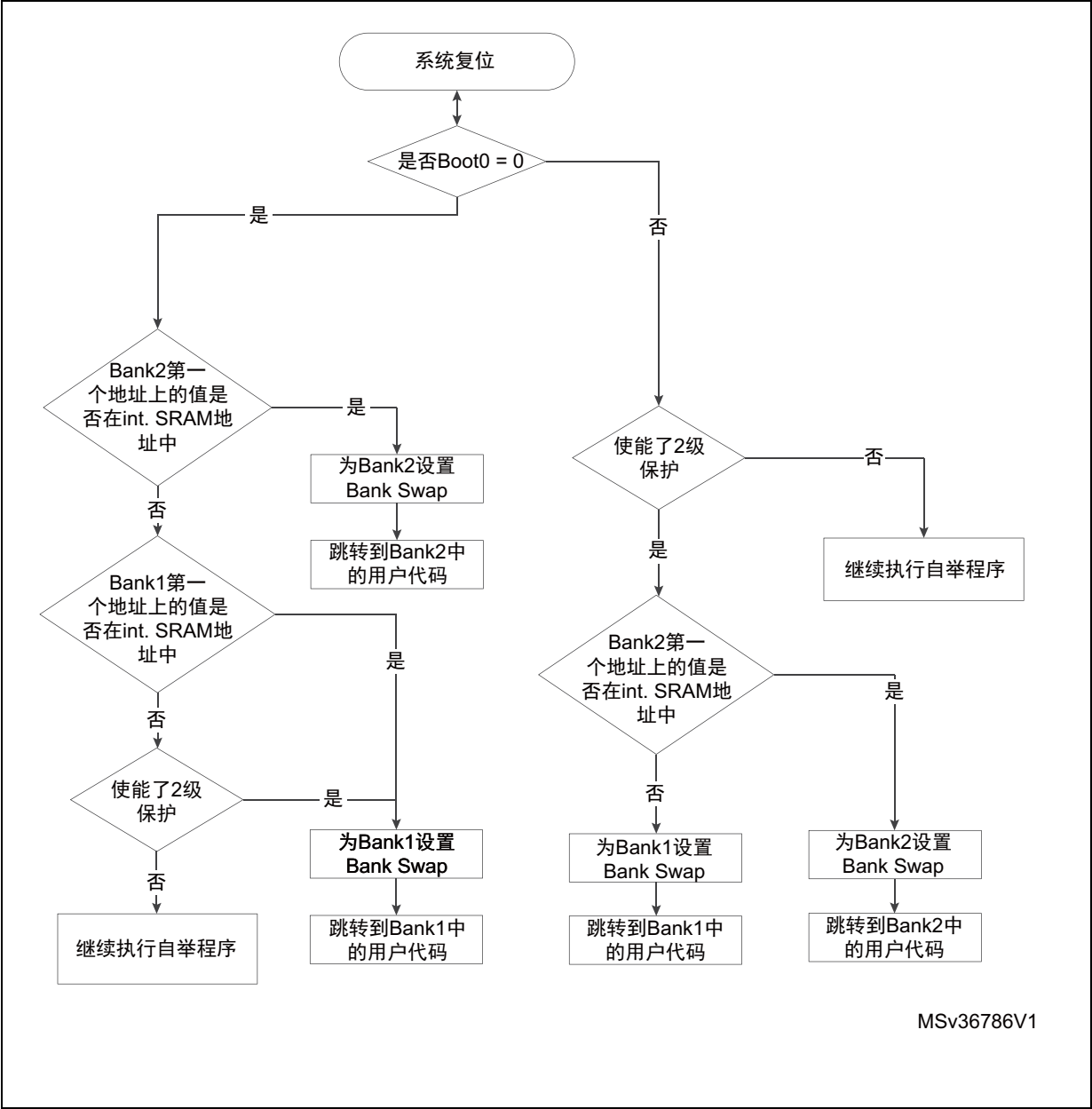
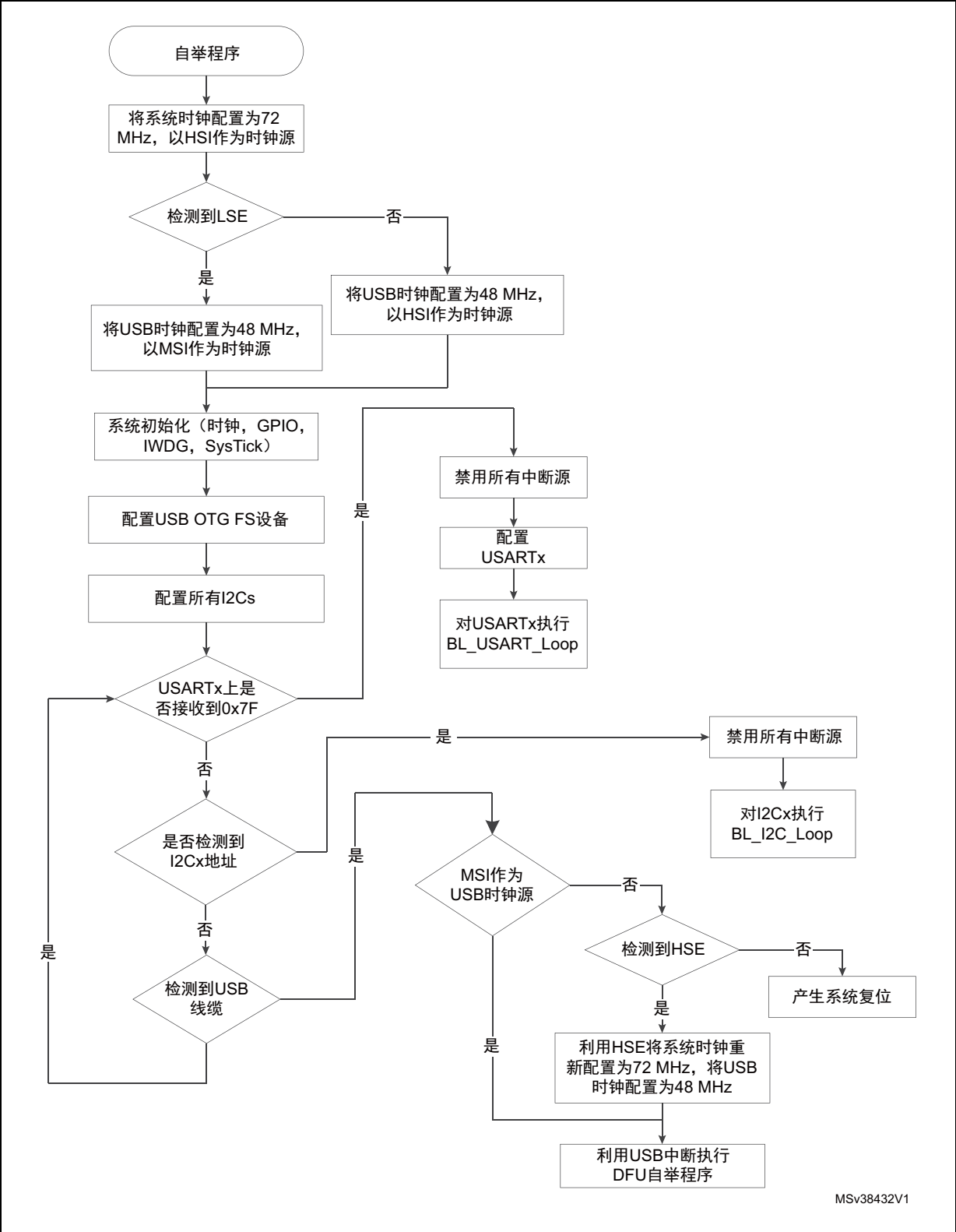


图 54.STM32L476xx/486xx 器件的自举程序 V10.x 选择



MSv38432V1



43.1.3 自举程序版本

下表列出了 STM32L476xx/486xx 器件自举程序 V10.x 版本：

表 95.STM32L476xx/486xx 自举程序 V10.x 版本

自举程序版本号	说明	已知限制
V10.0	废弃版本（不用）	无
V10.1	初始自举程序版本	向 SRAM 中写入是损坏的
V10.2	Fix write in SRAM issue	无
V10.3	增加了支持以 MSI 作为 USB 时钟源（MSI 由 LSE 调节）。更新了双存储区自举功能，以支持用户堆栈映射到 SRAM2 中的情况。	无

43.2 自举程序 V9.x

43.2.1 自举程序配置

STM32L476xx/486xx 自举程序通过应用模式 7 激活（详述参见表 2：自举程序激活模式）。下表介绍了该自举程序使用的硬件资源。

表 96. 系统存储器自举模式下 STM32L476xx/486xx 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	对于配置为 72 MHz 的系统时钟，以及 USART 和 I2C 自举程序操作，启动时 HSI 用作其时钟源。
		HSE 使能	HSE 仅在选择了 USB 接口且无 LSE 时使用。HSE 必须是下列值的其中之一：[24,20,18,16,12,8,6,4] MHz。如果使用了 USB，则系统时钟为 72 MHz；如果使用了 CAN，则系统时钟为 60 MHz。
		LSE 已使能	LSE 用来微调 MSI，MSI 配置为 48 MHz，为 USB 时钟源。LSE 必须等于 32,768 KHz。如果未检测到 LSE，则若连接了 USB，可使用 HSE 来代替。
		MSI 已使能	MSI 配置为 48 MHz，且用作 USB 时钟源。仅当检测到 LSE 时才使用 MSI，否则（未检测到 LSE），若连接了 USB 则使用 HSE。
		CSS	当 LSE 或 HSE 使能时，使能时钟安全系统 (CSS) 中断。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 12544 个字节空间供自举程序固件使用
	系统存储器	-	自地址 0x1FFF0000 起的 28672 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	-	如果选择了电压调节范围 2，则 DFU 不能用于与自举程序之间的通信。自举程序固件不能配置 PWR_CR1 寄存器中的电压调节范围值。
USART1 自举程序	USART1	启用	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：USART2 处于接收模式
	USART1_TX 引脚	输出	PA9 引脚：USART2 处于发送模式
USART2 自举程序	USART2	启用	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	PA3 引脚：USART2 处于接收模式
	USART2_TX 引脚	输出	PA2 引脚：USART2 处于发送模式
USART3 自举程序	USART3	启用	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

表 96. 系统存储器自举模式下 STM32L476xx/486xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
USARTx 自举程序	SysTick 定时器	启用	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	启用	I2C1 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000011x（其中 x = 0 对应写，x = 1 对应读）。
	I2C1_SCL 引脚	输入 / 输出	PB6 引脚：时钟线在开漏模式下使用。
	I2C1_SDA 引脚	输入 / 输出	PB7 引脚：数据线在开漏模式下使用。
I2C2 自举程序	I2C2	启用	I2C2 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000011x（其中 x = 0 对应写，x = 1 对应读）。
	I2C2_SCL 引脚	输入 / 输出	PB10 引脚：时钟线在开漏模式下使用。
	I2C2_SDA 引脚	输入 / 输出	PB11 引脚：数据线在开漏模式下使用。
I2C3 自举程序	I2C3	启用	I2C3 配置为： I2C 速度：高达 400 KHz，7 位地址，从模式，模拟滤波器开启。7 位从地址：0b1000011x（其中 x = 0 对应写，x = 1 对应读）。
	I2C3_SCL 引脚	输入 / 输出	PC0 引脚：时钟线在开漏模式下使用。
	I2C3_SDA 引脚	输入 / 输出	PC1 引脚：数据线在开漏模式下使用。
SPI1 自举程序	SPI1	启用	SPI1 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件。
	SPI1_MOSI 引脚	输入	PA7 引脚：从数据输入线，用于推挽下拉模式
	SPI1_MISO 引脚	输出	PA6 引脚：从数据输出线，用于推挽下拉模式
	SPI1_SCK 引脚	输入	PA5 引脚：从时钟线，用于推挽下拉模式
	SPI1_NSS 引脚	输入	PA4 引脚：从芯片选择引脚，用于推挽下拉模式。
SPI2 自举程序	SPI2	启用	SPI2 配置为： 从模式，全双工（Full Duplex），8 位 MSB，速度达 8 MHz，极性：CPOL 低，CPHA 低，NSS 硬件
	SPI2_MOSI 引脚	输入	PB15 引脚：从数据输入线，用于推挽下拉模式
	SPI2_MISO 引脚	输出	PB14 引脚：从数据输出线，用于推挽下拉模式
	SPI2_SCK 引脚	输入	PB13 引脚：从时钟线，用于推挽下拉模式
	SPI2_NSS 引脚	输入	PB12 引脚：从芯片选择引脚，用于推挽下拉模式。
CAN1 自举程序	CAN1	启用	初始化后，CAN1 的配置为：波特率 125 kbps，11 位标识符。
	CAN1_RX 引脚	输入	PB8 引脚：CAN1 处于接收模式
	CAN1_TX 引脚	输出	PB9 引脚：CAN1 处于发送模式

表 96. 系统存储器自举模式下 STM32L476xx/486xx 器件的配置（续）

自举程序	功能 / 外设	状态	备注
DFU 自举程序	USB	启用	USB FS 配置为强制器件模式。使能 USB FS 中断向量以用于 USB DFU 通信。 注：VDDUSB IO 必须连接到 3.3V，以支持 USB 运行。
	USB_DM 引脚	输入 / 输出	PA11 引脚：USB FS DM 线
	USB_DP 引脚		PA12 引脚：USB FS DP 线 无需外部上拉电阻。

假如存在 HSE，则系统时钟和 USB 时钟会分别配置为 72 MHz 和 48 MHz，以 PLL（由 HSE 计时）作为时钟源。



43.2.2 自举程序选择

下图显示了自举程序选择机制。

图 55.STM32L476xx/486xx 自举程序 V9.x 的双存储区自举实现

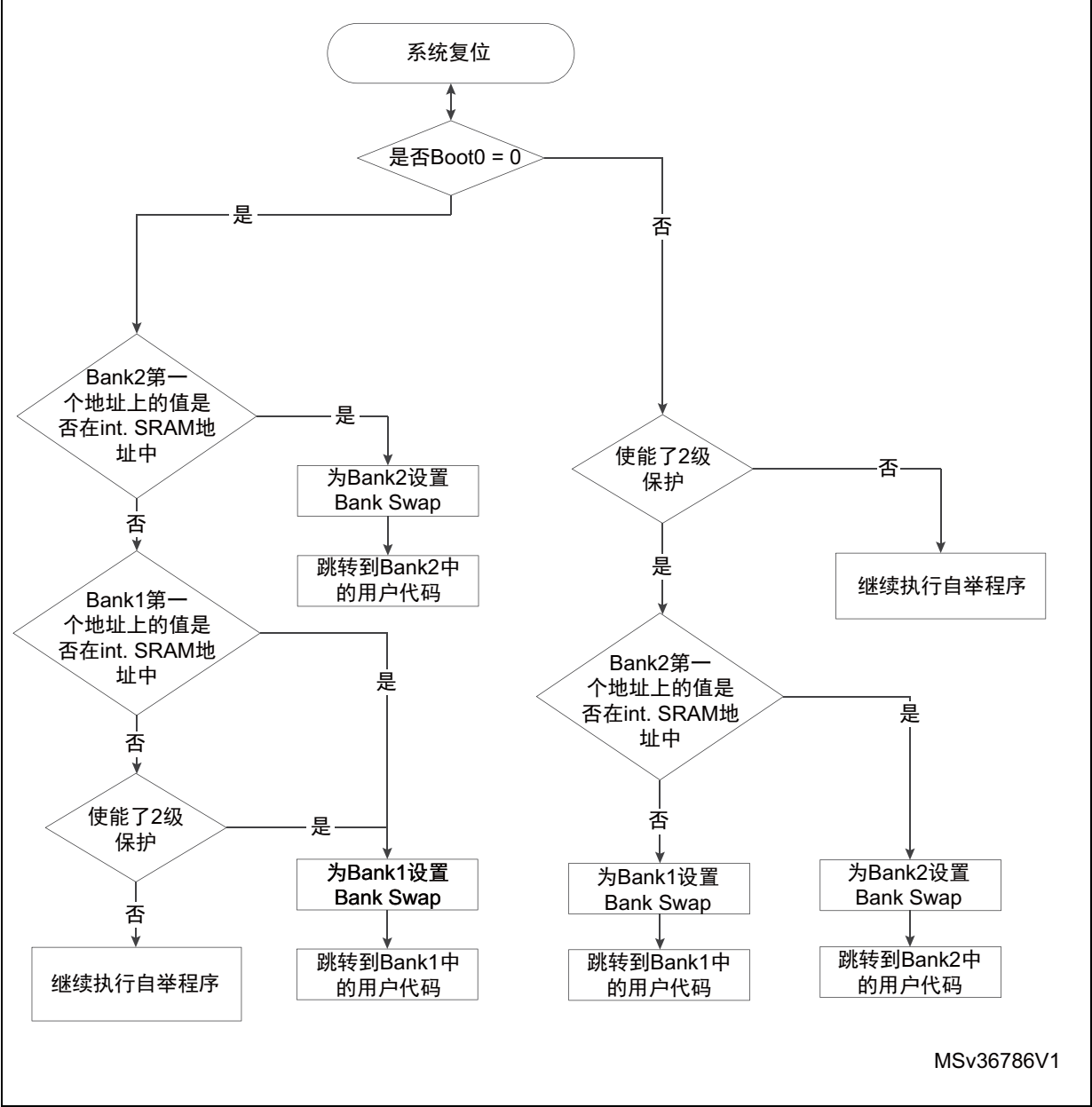
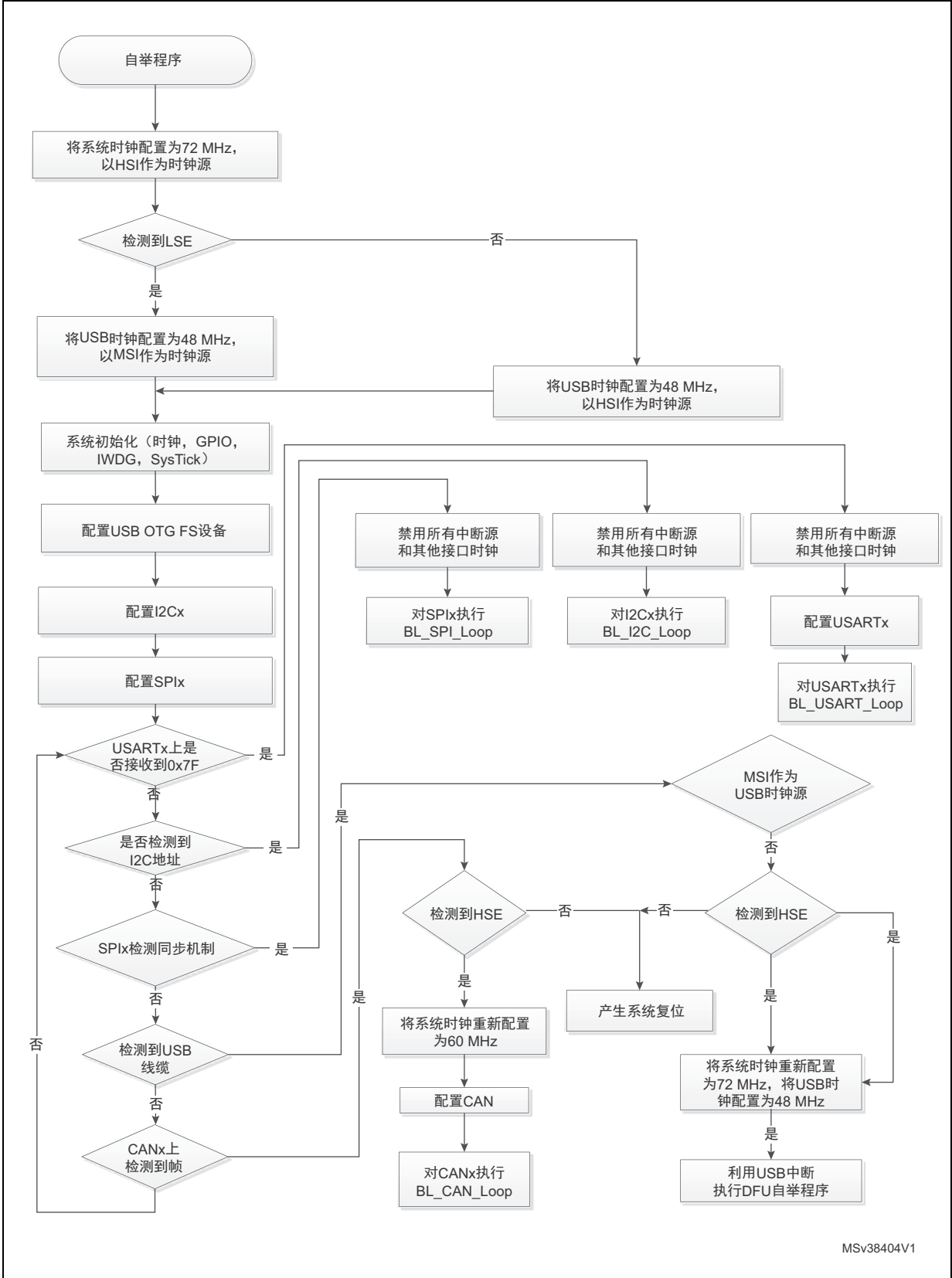


图 56.STM32L476xx/486xx 器件的自举程序 V9.x 选择



MSv38404V1

43.2.3 自举程序版本

下表列出了 STM32L476xx/486xx 器件自举程序 V9.x 版本：

表 97.STM32L476xx/486xx 自举程序 V9.x 版本

自举程序版本号	说明	已知限制
V9.0	初始自举程序版本	无

44 器件相关的自举程序参数

对于所有 STM32 器件，每种串行外设的自举程序协议命令集和序列都相同。但是，一些参数取决于器件和自举程序版本：

- PID（产品 ID）
- 当请求 Read Memory、Go 和 Write Memory 命令时，自举程序接收有效 RAM 存储地址（自举程序执行过程中所用 RAM 区域不可访问）。
- 系统存储区。

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 98. 器件相关的自举程序参数

STM32 系列	器件		PID	BL ID	RAM 存储器	系统 记忆
F0	STM32F05xxx 和 STM32F030x8 器件		0x440	0x21	0x20000800 - 0x20001FFF	0x1FFFE000 - 0x1FFFF7FF
	STM32F03xx4/6		0x444	0x10	0x20000800 - 0x20000FFF	
	STM32F030xC		0x442	0x52	0x20001800 - 0x20007FFF	0x1FFFD800 - 0x1FFFF7FF
	STM32F04xxx		0x445	0xA0	NA	0x1FFFC400 - 0x1FFFF7FF
	STM32F070x6		0x445	0xA2	NA	0x1FFFC400 - 0x1FFFF7FF
	STM32F070xB		0x448	0xA2	NA	0x1FFFC800 - 0x1FFFF7FF
	STM32F071xx/72xx		0x448	0xA1	0x20001800 - 0x20003FFF	0x1FFFC800 - 0x1FFFF7FF
	STM32F09xxx		0x442	0x50	NA	0x1FFFD800 - 0x1FFFF7FF
F1	STM32F10x xx	小容量	0x412	NA	0x20000200 - 0x200027FF	0x1FFFF000 - 0x1FFFF7FF
		中容量	0x410	NA	0x20000200 - 0x20004FFF	
		大容量	0x414	NA	0x20000200 - 0x2000FFFF	
		中容量超值型	0x420	0x10	0x20000200 - 0x20001FFF	
		大容量超值型	0x428	0x10	0x20000200 - 0x20007FFF	
	STM32F105xx/107xx		0x418	NA	0x20001000 - 0x2000FFFF	0x1FFFB000 - 0x1FFFF7FF
	STM32F10xxx 超大容量		0x430	0x21	0x20000800 - 0x20017FFF	0x1FFFE000 - 0x1FFFF7FF
F2	STM32F2xxxx		0x411	0x20	0x20002000 - 0x2001FFFF	0x1FFF0000 - 0x1FFF77FF
				0x33		

表 98. 器件相关的自举程序参数（续）

STM32 系列	器件	PID	BL ID	RAM 存储器	系统 记忆
F3	STM32F373xx	0x432	0x41	0x20001400 - 0x20007FFF	0x1FFFD800 - 0x1FFFF7FF
	STM32F378xx		0x50	0x20001000 - 0x20007FFF	
	STM32F302xB(C)/303xB(C)	0x422	0x41	0x20001400 - 0x20009FFF	
	STM32F358xx		0x50		
	STM32F301xx/302x4(6/8)	0x439	0x40	0x20001800 - 0x20003FFF	
	STM32F318xx		0x50		
	STM32F303x4(6/8)/ 334xx/328xx	0x438	0x50	0x20001800 - 0x20002FFF	
	STM32F302xD(E)/303xD(E)	0x446	0x40	0x20001800 - 0x2000FFFF	
	STM32F398xx	0x446	0x50	0x20001800 - 0x2000FFFF	
F4	STM32F40xxx/41xxx	0x413	0x31	0x20002000 - 0x2001FFFF	0x1FFF0000 - 0x1FFF77FF
	0x90		0x20003000 - 0x2001FFFF		
	STM32F42xxx/43xxx	0x419	0x70	0x20003000 - 0x2002FFFF	
	0x90				
	STM32F401xB(C)	0x423	0xD1	0x20003000 - 0x2000FFFF	
	STM32F401xD(E)	0x433	0xD1	0x20003000 - 0x20017FFF	
	STM32F410xx	0x458	0xB1	0x20003000 - 0x20007FFF	
	STM32F411xx	0x431	0xD0	0x20003000 - 0x2001FFFF	
	STM32F412xx	0x441	0x90	0x20003000 - 0x2003FFFF	
STM32F446xx	0x421	0x90	0x20003000 - 0x2001FFFF		
STM32F469xx/479xx	0x434	0x90	0x20003000 - 0x2005FFFF		
F7	STM32F74xxx/75xxx	0x449	0x70	0x20004000 - 0x2004FFFF	0x1FF00000 - 0x1FF0EDBF
			0x90	0x20004000 - 0x2004FFFF	0x1FF00000 - 0x1FF0EDBF
L0	STM32L031xx/041xx	0x425	0xC0	0x20001000 - 0x20002000	0x1FF00000 - 0x1FF00FFF
	STM32L05xxx/06xxx	0x417	0xC0	0x20001000 - 0x20001FFF	0x1FF00000 - 0x1FF00FFF
	STM32L07xxx/08xxx	0x447	0x40	0x20001000 - 0x20004FFF	0x1FF00000 - 0x1FF01FFF
0xB0			0x20001400 - 0x20004FFF		
L1	STM32L1xxx6(8/B)	0x416	0x20	0x20000800 - 0x20003FFF	0x1FF00000 - 0x1FF01FFF
	STM32L1xxx6(8/B)A	0x429	0x20	0x20001000 - 0x20007FFF	
	STM32L1xxxC	0x427	0x40		
	STM32L1xxxD	0x436	0x45	0x20001000 - 0x2000BFFF	
	STM32L1xxxE	0x437	0x40	0x20001000 - 0x20013FFF	
L4	STM32L476xx/486xx	0x415	0xA3	0x20003000 - 0x20017FFF	0x1FFF0000 - 0x1FFF6FFF
			0x90	0x20003100 - 0x20017FFF	

45 自举程序时序

本章给出了自举程序固件的典型时序，可用来确保主机和设备之间的正确同步。

本文将介绍两种时序：

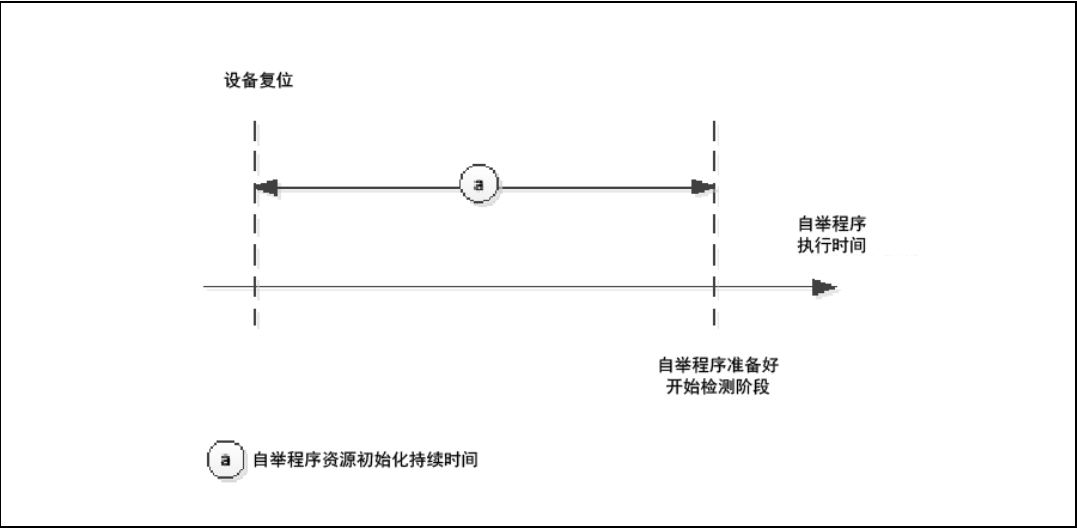
- STM32 器件自举程序资源初始化时间。
- 通信接口选择时间。

这些时序之后，自举程序准备好接收和执行主机命令。

45.1 自举程序启动时序

自举程序复位后，主机应当等待，直至 STM32 自举程序准备好利用专门接口通信启动检测阶段。这个时间对应于自举程序启动时序，期间初始化自举程序所用资源。

图 57. 自举程序启动时序描述



下表含有每种 STM32 产品的最小启动时序：

表 99.STM32 器件的自举程序启动时序

器件		最小自举程序启动 (ms)	HSE 超时 (ms)
STM32F03xx4/6		1.612	NA
STM32F05xxx 和 STM32F030x8 器件		1.612	NA
STM32F04xxx		0.058	NA
STM32F071xx/72xx		0.058	NA
STM32F070x6	HSE 已连接	3	200
	HSE 未连接	230	
STM32F070xB	HSE 已连接	6	200
	HSE 未连接	230	

表 99.STM32 器件的自举程序启动时序（续）

器件		最小自举程序启动 (ms)	HSE 超时 (ms)
STM32F09xxx		2	NA
STM32F10xxx		1.227	NA
STM32F105xx/107xx	PA9 引脚低电平	1.396	NA
	PA9 引脚高电平	524.376	
STM32F10xxx 超大容量		1.227	NA
STM32F2xxxx	V2.x	134	NA
	V3.x	84.59	0.790
STM32F301xx/302x4(6/8)	HSE 已连接	45	560.5
	HSE 未连接	560.8	
STM32F302xB(C)/303xB(C)	HSE 已连接	43.4	2.236
	HSE 未连接	2.36	
STM32F302xD(E)/303xD	HSE 已连接	7.53	NA
	HSE 未连接	146.71	NA
STM32F303x4(6/8)/334xx/328xx		0.155	NA
STM32F318xx		0.182	NA
STM32F358xx		1.542	NA
STM32F373xx	HSE 已连接	43.4	2.236
	HSE 未连接	2.36	
STM32F378xx		1.542	NA
STM32F398xx		1.72	NA
STM32F40xxx/41xxx	V3.x	84.59	0.790
	V9.x	74	96
STM32F401xB(C)		74.5	85
STM32F401xD(E)		74.5	85
STM32F410xx		0.614	NA
STM32F411xx		74.5	85
STM32F412xx		0.614	180
STM32F429xx/439xx	V7.x	82	97
	V9.x	74	97
STM32F446xx		73.61	96
STM32F469xx/479xx		73.68	230
STM32F74xxx/75xxx		16.63	50
STM32L031xx/041xx		0.62	NA
STM32L05xxx/06xxx		0.22	NA

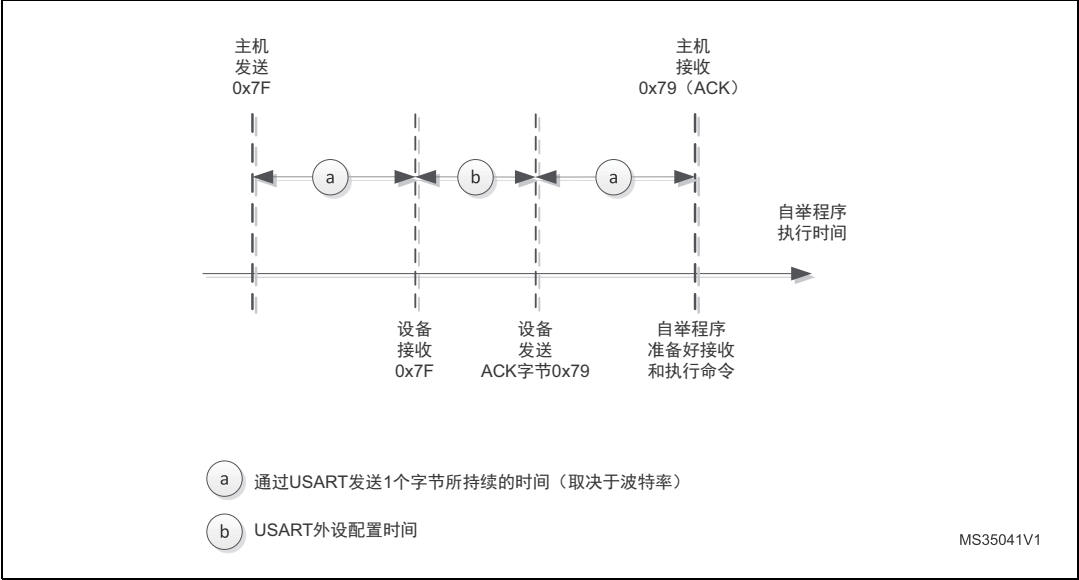
表 99.STM32 器件的自举程序启动时序（续）

器件			最小自举程序启动 (ms)	HSE 超时 (ms)
STM32L07xxx/08xxx	V4.x		0.61	NA
	V11.x		0.71	NA
STM32L1xxx6(8/B)A			0.542	NA
STM32L1xxx6(8/B)			0.542	NA
STM32L1xxxC			0.708	80
STM32L1xxxD			0.708	80
STM32L1xxxE			0.708	200
STM32L476xx/486xx	V10.x	LSE 已连接	55	100
		LSE 未连接	2560	
	V9.x	LSE 已连接	55,40	100
		LSE 未连接	2560,51	

45.2 USART 连接时序

USART 连接时序是主机在发送同步数据（0x7F）和接收第一个确认响应（0x79）之间所应等待的时间。

图 58.USART 连接时序描述



1. 接收到任何不同于 0x7F（或线路故障）的其他字符会导致自举程序以错误的波特率开始通信。自举程序测量 0x7F 中第一个比特的上升沿到 0x7F 中最后一个比特的下降沿之间的信号长度，来推导出波特率值
2. 自举程序不会将所计算的波特率重新调整为标准波特率值（即 1200, 9600, 115200, ...）。

注：对于 STM32F105xx/107xx 系列器件，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 引脚时将自举程序检测时间缩至最短，可在 USART 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 100.STM32 器件的 USART 自举程序最小时序

器件		一个 USART 字节发送周期 (ms)	USART 配置 (ms)	USART 连接 (ms)
STM32F03xx4/6		0.078125	0.0064	0.16265
STM32F05xxx 和 STM32F030x8 器件		0.078125	0.0095	0.16575
STM32F04xxx		0.078125	0.007	0.16325
STM32F071xx/72xx		0.078125	0.007	0.16325
STM32F070x6		0.078125	0.014	0.17
STM32F070xB		0.078125	0.08	0.23
STM32F09xxx		0.078125	0.07	0.22
STM32F10xxx		0.078125	0.002	0.15825
STM32F105xx/107xx	PA9 引脚低电平	0.078125	0.007	0.16325
	PA9 引脚高电平		105	105.15625

表 100.STM32 器件的 USART 自举程序最小时序 (续)

器件		一个 USART 字节发送周期 (ms)	USART 配置 (ms)	USART 连接 (ms)
STM32F10xxx 超大容量		0.078125	0.006	0.16225
STM32F2xxxx	V2.x	0.078125	0.009	0.16525
	V3.x			
STM32F301xx/302x4(6/8)	HSE 已连接	0.078125	0.002	0.15825
	HSE 未连接			
STM32F302xB(C)/303xB(C)	HSE 已连接	0.078125	0.002	0.15825
	HSE 未连接			
STM32F302xD(E)/303xD		0.078125	0.002	0.15885
STM32F303x4(6/8)/334xx/328xx		0.078125	0.002	0.15825
STM32F318xx		0.078125	0.002	0.15825
STM32F358xx		0.15625	0.001	0.3135
STM32F373xx	HSE 已连接	0.078125	0.002	0.15825
	HSE 未连接			
STM32F378xx		0.15625	0.001	0.3135
STM32F398xx		0.078125	0.002	0.15885
STM32F40xxx/41xxx	V3.x	0.078125	0.009	0.16525
	V9.x		0.0035	0.15975
STM32F401xB(C)		0.078125	0.00326	0.15951
STM32F401xD(E)		0.078125	0.00326	0.15951
STM32F410xx		0.078125	0.002	0.158
STM32F411xx		0.078125	0.00326	0.15951
STM32F412xx		0.078125	0.002	0.158
STM32F429xx/439xx	V7.x	0.078125	0.007	0.16325
	V9.x		0.00326	0.15951
STM32F446xx		0.078125	0.004	0.16
STM32F469xx/479xx		0.078125	0.003	0.159
STM32F74xxx/75xxx		0.078125	0.065	0.22
STM32L031xx/041xx		0.078125	0.018	0.174
STM32L05xxx/06xxx		0.078125	0.018	0.17425
STM32L07xxx/08xxx	V4.x	0.078125	0.017	0.173
	V11.x	0.078125	0.017	0.158
STM32L1xxx6(8/B)A		0.078125	0.008	0.16425
STM32L1xxx6(8/B)		0.078125	0.008	0.16425
STM32L1xxxC		0.078125	0.008	0.16425

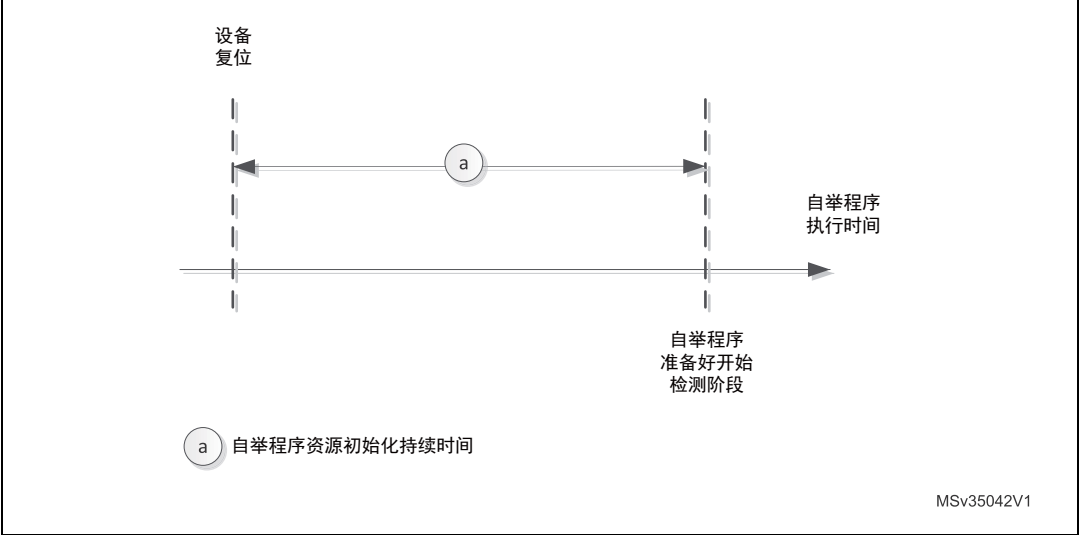
表 100.STM32 器件的 USART 自举程序最小时序（续）

器件		一个 USART 字节发送周期 (ms)	USART 配置 (ms)	USART 连接 (ms)
STM32L1xxD		0.078125	0.008	0.16425
STM32L1xxE		0.078125	0.008	0.16425
STM32L476xx/486xx	V10.x	0.078125	0.003	0.159
	V9.x	0.078125	0.003	0.159

45.3 USB 连接时序

USB 连接时序是插入 USB 线缆到与设备建立正确连接之间主机所应等待的时间。该时序包括了枚举和 DFU 元件配置。USB 连接取决于主机。

图 59.USB 连接时序描述



注：对于 STM32F105xx/107xx 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）。这是由基于帧起始 (SOF) 检测的 HSE 自动检测机制引起的。

表 101.STM32 器件的 USB 自举程序最小时序

器件		USB 连接 (ms)
STM32F04xxx		350
STM32F070x6		待定
STM32F070xB		320
STM32F105xx/107xx	HSE = 25 MHz	460
	HSE = 14.7465 MHz	4500
	HSE = 8 MHz	13700
STM32F2xxxx		270
STM32F301xx/302x4(6/8)		300

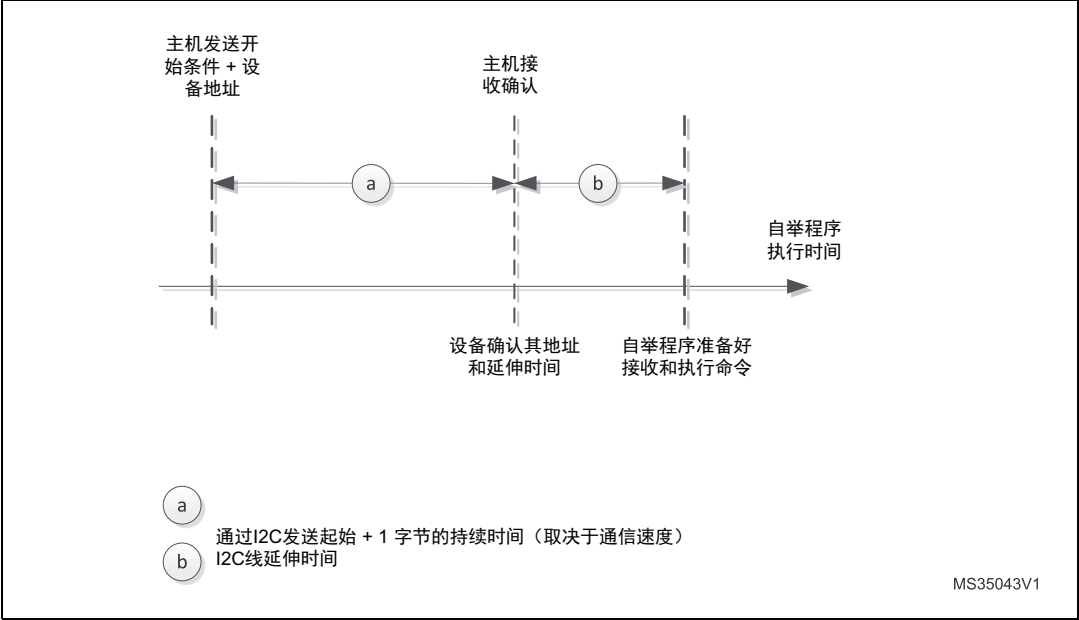
表 101. STM32 器件的 USB 自举程序最小时序 (续)

器件		USB 连接 (ms)
STM32F302xB(C)/303xB(C)		300
STM32F302xD(E)/303xD		100
STM32F373xx		300
STM32F40xxx/41xxx	V3.x	270
	V9.x	250
STM32F401xB(C)		250
STM32F401xD(E)		250
STM32F411xx		250
STM32F412xx		380
STM32F429xx/439xx	V7.x	250
	V9.x	250
STM32F446xx		200
STM32F469xx/479xx		270
STM32F74xxx/75xxx		230
STM32L07xxx/08xxx		140
STM32L1xxxC		849
STM32L1xxxD		849
STM32L476xx/486xx	V10.x	300
	V9.x	

45.4 I2C 连接时序

I2C 连接时序是发送 I2C 设备地址到发送命令代码之间主机所应等待的时间。该时序包括了 I2C 线路延伸持续时间。

图 60.I2C 连接时序描述



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。

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考虑 Flash 操作持续时间。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 102.STM32 器件的 I2C 自举程序最小时序

器件	启动条件 + 发送一个 I2C 字节 (ms)	I2C 线路延伸 (ms)	I2C 连接 (ms)	I2C 超时 (ms)
STM32F04xxx	0.0225	0.0025	0.025	1000
STM32F071xx/72xx	0.0225	0.0025	0.025	1000
STM32F070x6	0.0225	0.002	0.245	1000
STM32F070xB	0.0225	0.002	0.245	1000
STM32F09xxx	0.0225	0.002	0.245	1000
STM32F410xx	0.0225	0.002	0.245	1000
STM32F469xx/479xx	0.0225	0.002	0.245	1000
STM32F74xxx/75xxx	0.0225	0.002	0.245	500
STM32F303x4(6/8)/334xx/328xx	0.0225	0.0027	0.0252	1000

表 102.STM32 器件的 I2C 自举程序最小时序 (续)

器件		启动条件 + 发送 一个 I2C 字节 (ms)	I2C 线路延伸 (ms)	I2C 连接 (ms)	I2C 超时 (ms)
STM32F318xx		0.0225	0.0027	0.0252	1000
STM32F358xx		0.0225	0.0055	0.028	10
STM32F378xx		0.0225	0.0055	0.028	10
STM32F398xx		0.0225	0.002	0.245	1500
STM32F40xxx/41xxx		0.0225	0.0022	0.0247	1000
STM32F401xB(C)		0.0225	0.0022	0.0247	1000
STM32F401xD(E)		0.0225	0.0022	0.0247	1000
STM32F411xx		0.0225	0.0022	0.0247	1000
STM32F412xx		0.0225	0.002	0.245	1000
STM32F429xx/439xx	V7.x	0.0225	0.0033	0.0258	1000
	V9.x		0.0022	0.0247	
STM32F446xx		0.0225	0.002	0.245	1000
STM32L07xxx/08xxx		0.0225	0.002	0.245	1000
STM32L476xx/486xx	V10.x	0.0225	0.002	0.245	1000
	V9.x	0.0225	0.002	0.245	1000

45.5 SPI 连接时序

SPI 连接时序是主机在发送同步数据（0xA5）和接收第一个确认响应（0x79）之间所应等待的时间。

图 61.SPI 连接时序描述

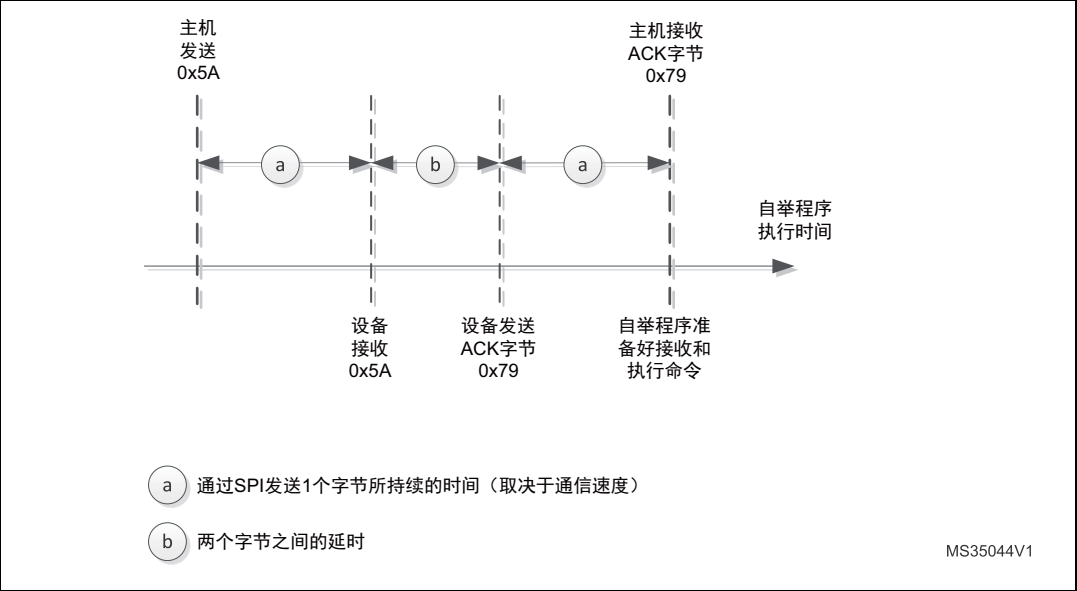


表 103.STM32 器件的 SPI 自举程序最小时序

器件	一个 SPI 字节发送周期 (ms)	两字节之间的延迟 (ms)	SPI 连接 (ms)
所有产品	0.001	0.008	0.01

46 版本历史

表 104. 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 1 月 22 日	2	<p>在产的所有 STM32（rev.B 和 rev.Z）都含有本应用笔记中所述的自举程序。</p> <p>修改：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：第 1.3 节：硬件要求，第 1.5 节：选择 USART 波特率，第 1.6 节：使用自举程序和章节：</p> <p>注 2 链接到表 3：自举程序命令中的 Get，Get Version & Read Protection Status 和 Get ID 命令，增加了注 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 5 月 26 日	3	<p>少量文本更改。在表：系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟中增加了 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>删除修改，修改了注 3，在第 9 页的表 3：自举程序命令中增加了注 1。</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，修改了注 3。</p> <p>修改了第 18 页的字节 8。</p> <p>第 18 页的第 2.5 节：Go 命令增加了注释。</p> <p>修改了第 20 页的图 11：Go 命令：器件端。</p> <p>第 21 页的第 2.6 节：Write Memory 命令增加了注释。</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：字节程序命令。</p> <p>修改了注释并在第 27 页第 2.8 节：Write Protect 命令中增加了注释。</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 1 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（参见表 4：自举程序版本）。</p>



表 104. 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，自举程序执行不需要外部石英时钟 中增加了 IWDG。增加了 注释。</p> <p>在整个文档中将 BL 更改为自举程序。</p> <p>在 表：系统时钟由嵌入式内部高速 RC 提供，自举程序执行不需要外部石英时钟 中修改了 Go 命名说明。</p> <p>第 2.4 节：Read Memory 命令 中修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 下面的注释。</p> <p>删除了 第 2.5 节：Go 命令 中的注释并增加了注释。</p> <p>指定了起始 RAM 地址，并在 第 2.6 节：Write Memory 命令 中增加了注释。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>第 2.8 节：Write Protect 命令 中增加了注释。</p>
2010 年 3 月 9 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）。更改了 前言。增加了 词汇表。</p>
2010 年 4 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>术语表：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：STM32F10xxx 超大容量器件自举程序。</p> <p>表 65：增加了超大容量器件的相关信息。</p>
2010 年 10 月 8 日	8	在 表 3 和 表 65 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 4 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 32 节：自举程序时序</p>
2011 年 6 月 6 日	12	<p>更新了：</p> <ul style="list-style-type: none"> – 表 12：TM32L1xxx6(8/B) 自举程序版本 – 表 17：系统存储器自举模式下 STM32F2xxxx 器件的配置 – 表 18：STM32F2xxxx 自举程序 V2.x 版本 – 表 20：STM32F2xxxx 自举程序 V3.x 版本
2011 年 11 月 28 日	13	<p>增加了 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：STM32F10xxx 器件自举程序 标题和概述中增加了超值型器件。</p>

表 104. 文档版本历史 (续)

日期	版本	变更
2012 年 7 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 8：STM32F105xx/107xx 自举程序版本 中增加了自举程序版本 V2.2。</p> <p>在 第 5.3.1 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 15：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 16：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 器件的配置 下增加了 DFU 接口的相关注释。在 表 20：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 21：系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置 中更新了 STM32F2xx 和 STM32F4xx 系统存储器结束地址。</p> <p>在 表 21：系统存储器自举模式下 STM32F40xxx/41xxx 器件的配置 下增加了 DFU 接口的相关注释。在 表 22：STM32F40xxx/41xxx 自举程序 V3.x 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 65：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 75：面向大容量超低功耗器件的 USART 自举程序时序，和 表 78：面向 STM32F051xx 器件的 USART 自举程序时序。</p> <p>增加了 表 88：面向大容量超低功耗器件的 USB 最小时序。</p>
2013 年 1 月 24 日	15	<p>更新了整个文档的通用产品名称（见 词汇表）。</p> <p>新增了以下各节：</p> <ul style="list-style-type: none"> – 第 8 节：STM32L1xxxC 器件自举程序。 – 第 13 节：STM32F031xx 器件自举程序。 – 第 14 节：STM32F373xx 器件自举程序。 – 第 15 节：STM32F302xB(C)/303xB(C) 器件自举程序。 – 第 16 节：STM32F378xx 器件自举程序。 – 第 17 节：STM32F358xx 器件自举程序。 – 第 18 节：STM32F427xx/437xx 器件自举程序。 – 第 34.3 节：I2C 自举程序时序特性。 <p>更新了 第 1 节：相关文献 和 第 2 节：词汇表。</p> <p>增加了 表 79 到 表 85（USART 自举程序时序）。</p> <p>用 图 16，以及图 18、19 和 42，替代了 图 6。</p> <p>修改了表 3、5、9、11、17、20、21、22 至 13、27、29、31、33、35、37 和 65。</p> <p>删除了 第 3.3 节：硬件连接要求 中的“X = 6：使用了一个 USART”。</p> <p>第 12.1 节：自举程序配置 中用地址 0x1FFF F802 替代了地址 0x1FFFF 8002。</p> <p>修改了 注：第 28 页，第 6.2 节：自举程序选择 和 第 9.2 节：自举程序选择 中自举程序代码的相关执行步骤。</p>

表 104. 文档版本历史（续）

日期	版本	变更
2013 年 2 月 6 日	16	<p>在整个文档中增加了 I²C 相关的信息。 改进了 表 1：可应用产品和 第 1 节：相关文献。 表 3：嵌入式自举程序修改如下： – 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” 更新了图 6、9 和 11。 增加了 词汇表中的 注：和 第 3.1 节：自举程序激活中的 注：。 替换内容： – 在表 17、19、19、22、21、27、37 和 59 中将“1.62 V”替换为“1.8 V” – 在表 33 的 RAM 行中将“5 KB”替换为“4 KB” – 在表 65 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：硬件连接要求中将“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 10 节：STM32F2xxxx 器件自举程序中将“STM32F2xxxx 器件版本 Y”替换为“STM32F2xxxx 器件版本 X 和 Y” – 在表 11、15 和 26 中将“电压范围 2”替换为“电压范围 1”。</p>
2013 年 5 月 21 日	17	<p>更新了： – 前言 – 第 2 节：词汇表 – 第 3.3 节：硬件连接要求 – 第 7 节：STM32L1xxx6(8/B) 器件自举程序中包括了超值型 STM32L100 – 第 32.2 节：USART 连接时序 – 第 34.2 节：USB 自举程序时序特性 – 第 34.3 节：I2C 自举程序时序特性 – 表 1：适用产品 – 表 3：嵌入式自举程序 – 表 25：系统存储器自举模式下 STM32F051xx 器件的配置 – 表 27：系统存储器自举模式下 STM32F031xx 器件的配置 – 表 65：器件相关的自举程序参数 – 图 17：STM32F031xx 器件的自举程序选择 增加了 第 19 节：STM32F429xx/439xx 器件自举程序。</p>

表 104. 文档版本历史 (续)

日期	版本	变更
2014 年 5 月 19 日	18	<p>增加:</p> <ul style="list-style-type: none"> – 图 1 到图 5, 图 52, 图 6, 图 25, 图 26, 图 24, 从图 37 到图 57, 图 61 – 表 4, 表 92, 表 93, 从表 6 到表 45, 从表 46 到表 43, 从表 66 到表 67, 从表 到表 103 – 第 38.4 节, 第 31.2 节, 第 45.1 节, 第 45.5 节 – 第 4 节, 第 22 节, 第 23 节, 第 21 节, 从第 16 节到第 43 节 – 图 1, 图 2, 图 3 和图 4 下面的注释 <p>更新了:</p> <ul style="list-style-type: none"> – 更新了从第 3 节到第 6 节和第 17 节, 第 31 节和第 31 节, 篇章结构分三个子章节来组织: 自举程序配置, 自举程序选择和自举程序版本。 – 更新了第 43 节和第 45 节 – 更新了图 25 和图 20 的框图。 – 固定了表 64 中 STM32F429xx/439xx 器件的 I2C 地址 – 表 1、表 2、表 3、表 24、表 86、表 88、表 90、表 28、表 30、表 50、表 98 – 从图 14, 到图 28, 图 8, 从图 57 到图 61 – 表 87 上的注释
2014 年 7 月 29 日	19	<p>更新了:</p> <ul style="list-style-type: none"> – 表 2 下面的注释 – 图 51 和图 52 – 第 2 节: 词汇表 – 第 31 节: STM32F42xxx/43xxx 器件自举程序上用 STM32F42xxx/43xxx 替换了 STM32F427xx/437xx 的参考 – 用 “STM32F07xxx” 替换了 “STM32F072xx” – 用 “STM32F051xx 和 STM32F030x8 器件 ” 替换了 “STM32F051xx”。 – 表 24, 表 30, 表 50, 表 92, 表 64, 表 66, 表 12, 表 18, 表 54, 表 56 和表 60 上 OTG_FS_DP 和 OTG_FS_DM 相关的注释栏 – 表 92 上 USB_DM 相关注释栏。 – 表 3 上用 “STM32F42xxx/43xxx” 替换了 “STM32F429xx/439xx” 的参考 – 表 66 上 SPI2_MOSI、SPI2_MISO、SPI2_SCK 和 SPI2_NSS 引脚的相关注释栏 <p>增加了:</p> <ul style="list-style-type: none"> – 表 2 下面的注释 – 表 1, 第 2 节: 词汇表, 表 99, 表 100, 表 101, 表 102 上 STM32F411 的参考 – 第 29 节: STM32F411xx 器件自举程序 <p>删除了表 3, 第 2 节: 词汇表, 表 98, 表 99, 表 100, 表 101 上 STM32F427xx/437xx 的参考</p>
2014 年 11 月 24 日	20	<p>更新了:</p> <ul style="list-style-type: none"> – 表 92 和表 78 上 “SPI1_NSS 引脚” 和 “SPI2_NSS 引脚” 行的注释 – 表 54、表 56 和表 60 上 “SPI1_NSS 引脚”、“SPI2_NSS 引脚” 和 “PI3_NSS 引脚” 行的注释 – 图 1

表 104. 文档版本历史 (续)

日期	版本	变更
2015 年 3 月 11 日	21	<p>更新了:</p> <ul style="list-style-type: none"> – 表 1、表 3、表 22、表 26、表 86、表 28、表 30、表 31、表 50、表 92、表 10、表 11、表 6、表 34、表 64、表 66、表 12、表 13、表 18、表 19、表 32、表 84、表 94、表 98、表 99、表 100、表 101 和 表 102 – 图 54 – 第 2 章: 词汇表 – 第 3.1 节和 第 3.4 节 <p>增加了:</p> <ul style="list-style-type: none"> – 第 43 节: STM32L476xx/486xx 器件自举程序和 第 32 节: STM32F446xx 器件自举程序
2015 年 6 月 9 日	22	<p>增加了:</p> <ul style="list-style-type: none"> – 第 8 节: STM32F070x6 器件自举程序 – 第 9 节: STM32F070xB 器件自举程序 – 第 11 节: STM32F09xxx 器件自举程序 – 第 18 节: STM32F302xD(E)/303xD(E) 器件自举程序第 24 节: STM32F398xx 器件自举程序 – 第 34 节: STM32F74xxx/75xxx 器件自举程序 – 第 43.2 节: 自举程序 V9.x – 图 58 上的注释 1 和 2 <p>更新了:</p> <ul style="list-style-type: none"> – 表 1 – 第 2 节: 词汇表 – 表 2 – 表 3 – 第 3.4 节: 自举程序存储器管理 – 表 98、表 99、表 100、表 101 和 表 102
2015 年 9 月 29 日	23	<p>增加了:</p> <ul style="list-style-type: none"> – 第 28 节: STM32F410xx 器件自举程序 – 第 33 节: STM32F469xx/479xx 器件自举程序 – 第 35 节: STM32L031xx/041xx 器件自举程序 – 第 37 节: STM32L07xxx/08xxx 器件自举程序 <p>更新了:</p> <ul style="list-style-type: none"> – 表 1 – 第 2 节: 词汇表 – 表 3 – 图 54、表 96、表 99、表 100、表 101、表 102
2015 年 11 月 2 日	24	<p>更新了:</p> <ul style="list-style-type: none"> – 表 1、表 3、表 98、表 99、表 100、表 101、表 102 – 第 33 节 <p>增加了:</p> <ul style="list-style-type: none"> – 第 25.2.1 节上的注释 – 第 30 节

表 105. 中文文档版本历史

日期	版本	变更
2016 年 6 月 3 日	1	中文初始版本。



重要通知 - 请仔细阅读

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和 / 或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2016 STMicroelectronics - 保留所有权利 2016