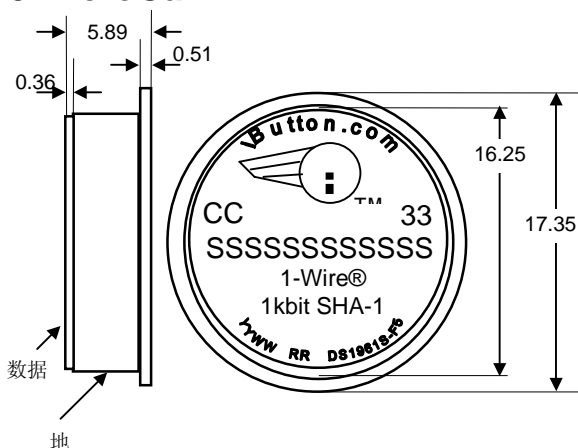


DS1961S 带有 SHA-1 引擎保护的 1k 位 EEPROM

特性

- 1128 位 5V EEPROM 存储器，分为四页，每页 256 位，64 位只写密钥和最多五个通用读/写寄存器
- 内置 512 位 SHA-1 引擎，用于计算 160 位信息鉴定码 (MAC) 或生成密钥
- 写访问需要知道密钥，并且能够计算和传送 160 位 MAC，以便鉴别真伪
- 可以对密钥和数据存储器加写保护（所有页或者只是第 0 页），或者将它们置于 EPROM 仿真模式 (“write to 0”，第 1 页)
- 读写操作可在很宽的电压范围 (2.8V 至 5.25V) 和温度范围 (-40°C 至 +85°C) 内进行

F5 MicroCan



图中所有尺寸的单位都是 mm。

iButton 共性

- 唯一的、由工厂光刻和测试的 64 位地址码 (8 位家族码 + 48 位序列号 + 8 位 CRC 校验码) 没有任何两个器件相同，保证绝对可溯
- 内置的多节点控制器保证与其它 1-Wire® 网络产品兼容
- 控制，寻址，数据和供电通过单一数据引脚实现
- 常规通信速率 16.3kbps；高速模式下通信速率达 142kbps

订购信息

DS1961S-F5
DS1961S-F3

F5 MicroCan™
F3 MicroCan

附件样例

DS9096P	自粘胶垫
DS9101	多用途夹
DS9093RA	安装固定环
DS9093A	环扣
DS9092	iButton 读取探头
DS9097U	通用 PC 串口适配器
DS1963S	4k 位 SHA iButton®, 用作协处理器

iButton 和 1-Wire 是 Dallas Semiconductor 的注册商标。
MicroCan 是 Dallas Semiconductor 的商标。

说明

DS1961S 在单一芯片内集成了 1024 位 EEPROM，64 位密钥，一个 8 个字节的寄存器/控制页（其中包含五个用户读/写字节），512 位 SHA-1 引擎，和一个全功能的 1-Wire 接口。每个 DS1961S 都有出厂时利用激光光刻写入芯片的 64 位 ROM 地址号，用来保证其绝对可溯性。数据按照 1-Wire 协议串行传送，只需一根数据线和返回地线。DS1961S 有一个称为暂存器的辅助存储区，在向主存储器，寄存器写入数据时，或者在安装新密钥时充当缓冲器。数据首先被存入暂存器，并可从这里读回。经过验证后，假定 DS1961S 接收到了匹配的 160 位信息鉴定码（MAC），那么复制暂存器命令将把数据传送到最终的存储单元。MAC 的计算涉及到存储在 DS1961S 中的密钥和包含器件地址号在内的附加数据。只有加载新的密钥时才无需提供 MAC。当读取存储页或是计算新密钥的时候，也可以激活 SHA-1 引擎来计算 160 位的 MAC，而不必加载它。DS1961S 的应用包括知识产权保护，防窜改数据载体，小额现金交易（例如自动售货机上的电子支付，停车收费系统，和公共交通）。

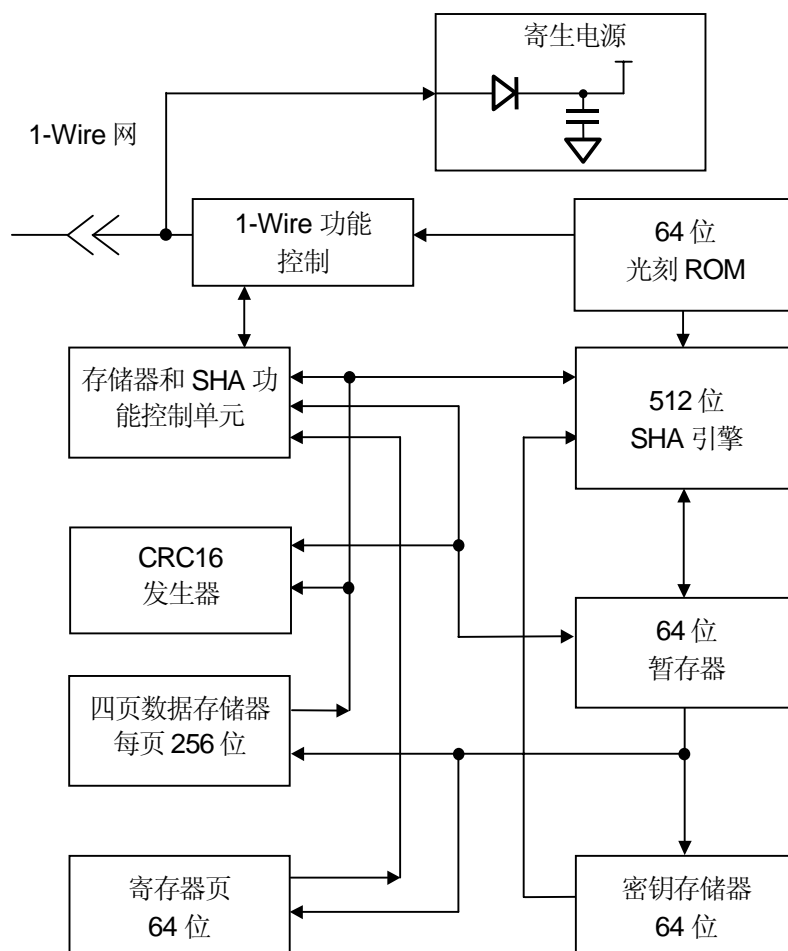
概述

图 1 中的框图说明了 DS1961S 的主控部分和存储单元之间的关系。DS1961S 有五个主要的数据部件：64 位光刻 ROM，64 位暂存器，四个 32 字节的 EEPROM 页，64 位寄存器页，64 位密钥存储器，和一个 512 位 SHA-1（安全散列算法）引擎。1-Wire 协议的分层结构见图 2。总线主控必须首先提供七个 ROM 操作命令中的一个：Read ROM, Match ROM, Search ROM, Skip ROM, Resume Communication, Overdrive Skip ROM 或 Overdrive Match ROM。一旦以标准速度完成 Overdrive ROM 命令，器件就进入高速模式，随后的所有通信都以高速进行。图 9 说明了协议所要求的这些 ROM 操作命令。成功地执行了 ROM 操作命令后，就可以进行存储器操作，主控制器可以发出七条存储器操作命令中的任一个。图 7 说明了有关这些存储器操作命令的协议。所有数据的读和写都是低位在前。

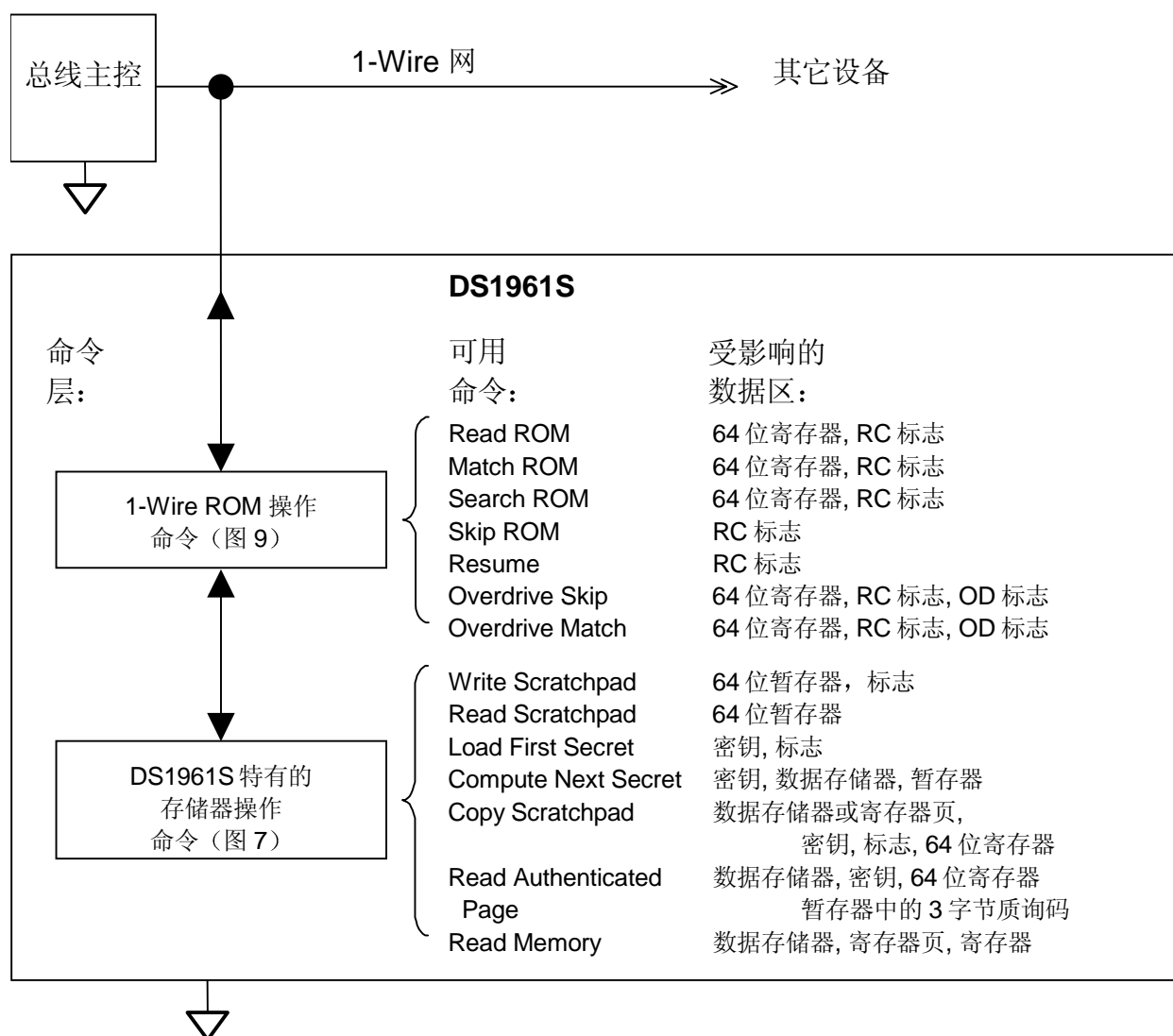
64 位光刻 ROM

每个 DS1961S 都有 64 位的唯一 ROM 代码。开头的八位是 1-Wire 家族代码。然后是 48 位的唯一序列号。最后 8 位是前 56 位的 CRC-8 检验码（图 3）。1-Wire CRC 校验码由一个包含移位寄存器和异或门的多项式发生器产生，如图 4 所示。生成多项式为 $X^8 + X^5 + X^4 + 1$ 。关于“Dallas 1-Wire Cyclic Redundancy Check”的更多信息参见 Dallas Semiconductor 的 *Book of DS19xx iButton Standards* (www.ibutton.com/ibuttons/standard.pdf)。移位寄存器初值为零。然后，从家族代码的最低有效位开始，每次移入一位。当 8 位家族代码全部移入后，再移序列号。当 48 位序列号也全部移入后，留在移位寄存器中的就是 CRC 值。移入 8 位 CRC 校验码后，移位寄存器应该全部归零。

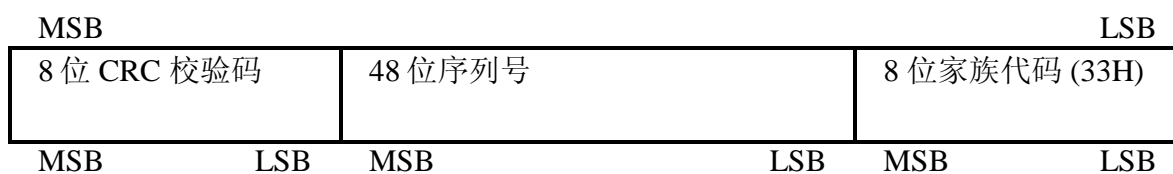
DS1961S 框图 图 1



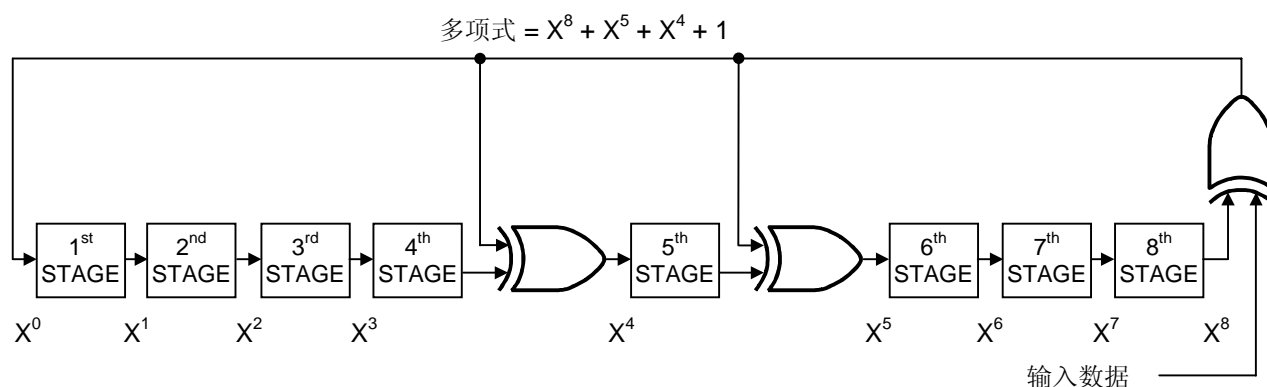
1-Wire 协议的层次结构图 2



64 位光刻 ROM 图 3



1-Wire CRC 发生器 图 4



存储器映像

DS1961S 有四个存储区：数据存储器，密钥存储器，含有特定功能和用户字节的寄存器页和暂存器。数据存储器每页 32 个字节。密钥、寄存器页和暂存器均为 8 字节。向数据存储器写数据，装载初始密钥，或者向寄存器页写入数据时，暂存器作为缓存器使用。

正如 *DS1961S* 存储器映像所示，数据存储器、密钥存储器和寄存器页位于一个线性地址空间中。数据存储器和寄存器页对读访问没有限制。但向数据存储器和寄存器页写数据则需要知道密钥。

DS1961S 存储器映像 图 5

数据存储器	地址	说明	注释
	0000H 至 001FH	数据存储器页 0	没有密钥不可写入
	0020H 至 003FH	数据存储器页 1	没有密钥不可写入
	0040H 至 005FH	数据存储器页 2	没有密钥不可写入
	0060H 至 007FH	数据存储器页 3	没有密钥不可写入
密钥	0080H 至 0087H	密钥存储器	不可读；写入无需密钥
寄存器页	0088H ¹	写保护密钥；008CH 至 008FH	代码 AAH 或 55H 激活保护
	0089H ¹	写保护页 0 至 3	代码 AAH 或 55H 激活保护
	008AH ¹	用户字节，自保护	代码 AAH 或 55H 激活保护
	008BH	工厂字节（只读）	读出为 AAH 或 55H；见内文
	008CH ¹	用户字节/页 1 的 EPROM 模式控制	代码 AAH 或 55H 激活模式
	008DH ¹	用户字节/仅写保护页 0	代码 AAH 或 55H 激活保护
	008EH 至 008FH	用户字节/制造商 ID	功能取决于工厂编程字节
	0090H 至 0097H	64 位地址码	(另一种读取地址码的方法)

¹ 一旦编程为 AAH 或 55H，该地址就成为只读。可以存储所有其它的代码，但既不能对地址加写保护，也不激活任何功能。

密钥的安装有两种方法，一是把数据从暂存器复制到密钥存储器，二是通过当前密钥和暂存器内容经过运算后生成新的密钥。密钥不能直接读取；只有 SHA 引擎能够访问它，以计算信息鉴定码。

地址 0090H 至 0097H 为读取器件的 ROM 地址号提供了另一个可供选择的方法。家族代码存在较低地址，随后是 48 位的序列号和存储在地址 0097H 的 8 位 CRC 校验码。从这些地址（0090H 至 0097H）读取时，总线主控能够接收到地址号的每一位，其顺序与 ROM 操作命令的顺序是一样的。

寄存器页

地址 0088H 至 008FH，也就是寄存器页，含有特定功能寄存器和通用用户字节以及一个工厂字节。一旦编程为 AAH 或 55H，这些字节中的大多数将被写保护而不能再更改。所有其它的代码将既无写保护，也不能激活与这个特定字节相关的特定功能。下表说明了这些特定功能：

地址	特定功能	如何激活特定功能
0088H	写保护密钥和地址 008CH 至 008FH 中的内容	向地址 0088H 写入 AAH 或 55H
0089H	同时写保护四个数据存储器页	向地址 0089H 写入 AAH 或 55H
008A	自保护地址 008AH（用户字节）	向地址 008AH 写入 AAH 或 55H
008B	写保护用户字节/制造商 ID（008EH 至 009FH）	由工厂向地址 008BH 写入 AAH 通常，从这个地址读到 55H，表明地址 008E 和 008F 是可读/写的用户字节，没有任何特定功能和锁定机制。代码 AAH 表明这两个字节被编程为 16 位制造商 ID，并在工厂内加了写保护。制造商 ID 是一个由用户提供的识别码，用来协助应用软件识别 DS1961S 所在的产品，以及快速找到可用的密钥。设置和注册制造商 ID 可与工厂联系。
008C	用户字节或数据存储器页 1 的 EPROM 模式激活。一旦激活 EPROM 模式，在数据存储器未加写保护的情况下，地址 0020H 至 003FH 中的位只能从逻辑 1 改为逻辑 0。	向地址 008CH 写入 AAH 或 55H
008D	用户字节或仅写保护数据存储器页 0	向地址 008DH 写入 AAH 或 55H
008E 至 008F	用户字节或制造商 ID	这些字节可在工厂加写保护或向地址 008BH 写入 AAH 实现写保护周期。

地址寄存器 图 6

位	7	6	5	4	3	2	1	0
目的地址 (TA1)	T7	T6	T5	T4	T3	T2 (0)	T1 (0)	T0 (0)
目的地址 (TA2)	T15	T14	T13	T12	T11	T10	T9	T8
结束地址及数据状态 (E/S)(只读)	AA	1	PF	1	1	E2 (1)	E1 (1)	E0 (1)

地址寄存器和传输状态

DS1961S 使用三个地址寄存器：TA1，TA2 和 E/S（图 6）。这些寄存器普遍用于许多其它 1-Wire 器件，但在 DS1961S 中的工作略有不同。寄存器 TA1 和 TA2 装载写入数据的地址或读取数据的源地址。寄存器 E/S 是一个只读的传输状态寄存器，用于验证写命令的数据完整性。因为 DS1961S 的暂存器只接收 8 字节的数据块，所以 TA1 的低三位总为 0，E/S 寄存器（结束偏移量）的低三位总是 1。这意味着暂存器中的所有数据随后都要复制到主存储器或密钥中。E/S 寄存器的第 5 位称为 PF 或“字节不全标志（partial byte flag）”，该位如果为逻辑 1 则意味着主控制器发送的数据位数不是 8 的整数倍，或者暂存器中的数据由于掉电的关系而成为无效数据。有效的写暂存器操作将清除 PF 位。第 3，4 和 6 位没有功能；读出时总为 1。利用 PF 标志，主控制器可以在写命令之后检验数据的完整性。E/S 寄存器的最高位称为 AA 或授权许可（authorization accepted），用以指示暂存器中的数据已复制到目的存储器地址。向暂存器中写入数据将清除该标志。

写入及验证

为了向 DS1961S 写入数据，必须把暂存器用作中间存储器。首先，主控制器发 Write Scratchpad（写暂存器）命令并指定目的地址，然后将数据写入暂存器。需要注意的是，数据必须写入存储器的 8 字节边界内，也就是说，目标地址的三个最低有效位（T2—T0）必须等于 000b。如果发送的 T2—T0 为非零值，器件将把这些位置为零，命令一执行完，就写入更改后的地址。此外，执行命令后，整个 8 字节的暂存器将被复制到存储器，因此，应该向暂存器写入 8 个字节的数据以确保复制的数据是已知的。在一定条件下（见 Write Scratchpad），主控将在 Write Scratchpad 命令序列的末端收到一个反码的 CRC16 校验码，用于校验命令、地址（主控发出的）和数据（注意，在非零 T2—T0 情况下，CRC 校验码的计算基于实际发送的目的地址，而非调整后的地址）。知道了 CRC 校验码，主控制器将其与自己计算出来的值相比较，就可以判断命令执行是否成功，并决定是否继续执行 Copy Scratchpad（复制暂存器）命令。如果主控制器未能收到 CRC16，则应该通过 Read Scratchpad（读暂存器）命令来验证数据的完整性。读暂存器时，在暂存器数据之前，DS1961S 会重新发回目的地址 TA1 和 TA2，以及 E/S 寄存器的内容。如果 PF 标志置位，那就说明数据未能正确送达暂存器，或者上一次写暂存后发生过掉电故障。主控不必再继续读；它可以尝试再次向暂存器写数据。类似地，如果 AA 标志置位而 PF 标志清零，则说明器件未能认可写命令。一切步骤正确的话，两个标志都清零。主控制器就可以继续读取和验证每个数据字节了。验证数据后，主控制器就可以发 Copy Scratchpad 命令了。该命令之后必须紧随三个

地址寄存器中的数据：TA1，TA2 和 E/S。主控制器应该通过读暂存命令获得这些寄存器的内容。

存储器和 SHA 操作命令

作为一个安全器件，DS1961S 与其它 1-Wire 存储器件的使用稍有不同。DS1961S 的大多数存储器可以象其它所有 1-Wire 存储器一样读取，但在尝试读取密钥时只能读到 FFH 字节，而不是真实数据。图 7 所示的存储器和 SHA 功能流程描述了访问存储器和操作 SHA 引擎的协议。主控制器与 DS1961S 之间的通信或者以标准速率（默认，OD=0），或者以高速模式（OD=1）进行。如果没有明确设定为高速模式，DS1961S 默认为标准速率。

Write Scratchpad [0FH]

Write Scratchpad 适用于数据存储区、密钥和寄存器页中的可写地址。如果总线主控发送的目的地址大于 90H，将不执行该命令。

发出 Write Scratchpad 命令后，主控制器必须首先提供 2 个字节的地址，随后是要写入暂存器的数据。数据将从暂存器的开头部分开始写入。值得注意的是，不论主控制器传送了多少个字节，结束偏移量（E2：E0）的值总是 111b。由于这个原因，主控制器应该总是发送 8 个字节的数据，尤其是载入的数据被用作密钥时。如果主控制器发送的数据少于 8 个字节，并且也没有读回暂存器进行验证，那么新密钥的一部分可能是主控制器所不知道的随机数。只有完整的数据字节才能被接受。如果最后一个数据字节不完整，该字节将被忽略，并置位字节不全标志 PF。

执行 Write Scratchpad 命令时，DS1961S 内部的 CRC 发生器（图 12）随着主控制器的发送过程，计算整个数据流的 CRC 校验码，始于命令码，止于最后一个数据字节。该 CRC 校验码利用 CRC16 多项式产生，它首先清除 CRC 发生器，然后移入 Write Scratchpad 的命令代码（0FH），接着是目的地址 TA1 和 TA2，以及所有的数据字节。要注意的是，尽管 DS1961S 在实际的 Write Scratchpad 命令中将设置 TA1 的位 T2—T0 为 000b，但是 CRC16 是根据主控发送的实际 TA1 来作计算的。主控可以随时终止 Write Scratchpad 命令。但是，如果装满暂存器的话，主控制器如果再发 16 个读时隙的话，就可以收到由 DS1961S 产生的 CRC 校验码。

如果 Write Scratchpad 命令的目的地址在密钥页（080H 至 087H）或寄存器页（88H 至 8FH）中，那么后续的 Read Scratchpad 命令将会从写保护的地址读到 AAH 或 55H，而不是 Write Scratchpad 命令中所写的值。同样，如果目的地址在第 1 页中，并且该页处于 EPROM 模式，那么从暂存器读回的数据将是写入暂存器的数据和目标存储器当前内容的逻辑与。

Read Scratchpad [AAH]

Read Scratchpad 可以用来验证目的地址和暂存器数据的完整性。发出命令码后，主控制器开始读数据。开头的两个字节是目的地址，其中 T2 至 T0 = 0。下一个字节是结束偏移量/数据状态字节

(E/S)，跟在后面的便是暂存器数据，它可能与主控制器最初发送的数据不同，尤其是当目的地地址为密钥存储器，寄存器页，或处于 EPROM 模式的存储器页 1 时。主控制器应该读到暂存器的最后一个字节，随后，就可以收到反码的 CRC。它基于 DS1961S 所发送的数据产生。如果主控在读取 CRC 校验码后继续读，那么读到的所有数据都将是逻辑 1。

Load First Secret [5AH]

密钥未被写保护时，Load First Secret（首次装载密钥）命令可用暂存器中的内容替换器件的当前密钥。这条命令不需要知晓器件的当前密钥。在执行 Load First Secret 命令之前，主控制器必须利用密钥的起始地址（0080H）把新密钥写入暂存器。发出 Load First Secret 命令后，主控制器必须提供一个 3 字节的授权模式，这个数据应该通过紧邻此条命令的前一个 Read Scratchpad 命令获得。这 3 个字节的模式数据必须与三个地址寄存器（依次为 TA1，TA2，E/S）中的数据完全匹配。如果模式匹配，而且密钥未加写保护，AA 标志将置位，并开始复制数据。暂存器内容的所有 8 个字节的数据都将被复制到密钥存储单元。器件内部数据的传输最多消耗 10ms，在此期间 1-Wire 总线上的电压一定不要低于 2.8V。复制完数据后，它将交替地发送 1 和 0，直至主控制器发出复位脉冲（Reset Pulse）为止。

除了 Load First Secret 命令，还可以通过 Copy Scratchpad 命令装载一个新的密钥。不过，这个方法需要知道当前密钥，并要计算 160 位的 MAC。

Compute Next Secret [33H]

一些应用对安全性的要求要比利用单一的、直接写入密钥所能达到的安全水平要高。为增加安全性，DS1961S 能够基于当前密钥、一个指定的存储器页的内容、以及暂存器中所有数据组成的部分密钥计算出一个新的密钥。在密钥未被写保护的情况下，要安装计算出来的密钥，主控制器需要发 Compute Next Secret（计算下一密钥）命令，这条命令将激活 512 位 SHA-1 引擎。表 1 说明了有关的各种数据成分是如何进入 SHA 引擎的，以及 SHA 结果的一部分是如何载入密钥存储单元的。稍后，本文将介绍 SHA 的算法。Compute Next Secret 命令可以根据需要多次使用，以便提高安全性水平。总线主控制器不必知道器件的当前密钥，就可以成功计算出一个新密钥，并用它覆盖现存的密钥。

Compute Next Secret 命令所需的 SHA-1 输入数据 表 1

M0[31:24] = (SS + 0)	M0[23:16] = (SS + 1)	M0[15:8] = (SS + 2)	M0[7:0] = (SS + 3)
M1[31:24] = (PP + 0)	M1[23:16] = (PP + 1)	M1[15:8] = (PP + 2)	M1[7:0] = (PP + 3)
M2[31:24] = (PP + 4)	M2[23:16] = (PP + 5)	M2[15:8] = (PP + 6)	M2[7:0] = (PP + 7)
M3[31:24] = (PP + 8)	M3[23:16] = (PP + 9)	M3[15:8] = (PP + 10)	M3[7:0] = (PP + 11)
M4[31:24] = (PP + 12)	M4[23:16] = (PP + 13)	M4[15:8] = (PP + 14)	M4[7:0] = (PP + 15)
M5[31:24] = (PP + 16)	M5[23:16] = (PP + 17)	M5[15:8] = (PP + 18)	M5[7:0] = (PP + 19)
M6[31:24] = (PP + 20)	M6[23:16] = (PP + 21)	M6[15:8] = (PP + 22)	M6[7:0] = (PP + 23)
M7[31:24] = (PP + 24)	M7[23:16] = (PP + 25)	M7[15:8] = (PP + 26)	M7[7:0] = (PP + 27)
M8[31:24] = (PP + 28)	M8[23:16] = (PP + 29)	M8[15:8] = (PP + 30)	M8[7:0] = (PP + 31)
M9[31:24] = FFH	M9[23:16] = FFH	M9[15:8] = FFH	M9[7:0] = FFH
M10[31:24] = MPX	M10[23:16] = (SP + 1)	M10[15:8] = (SP + 2)	M10[7:0] = (SP + 3)
M11[31:24] = (SP + 4)	M11[23:16] = (SP + 5)	M11[15:8] = (SP + 6)	M11[7:0] = (SP + 7)
M12[31:24] = (SS + 4)	M12[23:16] = (SS + 5)	M12[15:8] = (SS + 6)	M12[7:0] = (SS + 7)
M13[31:24] = FFH	M13[23:16] = FFH	M13[15:8] = FFH	M13[7:0] = 80H
M14[31:24] = 00H	M14[23:16] = 00H	M14[15:8] = 00H	M14[7:0] = 00H
M15[31:24] = 00H	M15[23:16] = 00H	M15[15:8] = 01H	M15[7:0] = B8H

Compute Next Secret 结果

(SS + 0) := E[7:0]	(SS + 1) := E[15:8]	(SS + 2) := E[23:16]	(SS + 3) := E[31:24]
(SS + 4) := D[7:0]	(SS + 5) := D[15:8]	(SS + 6) := D[23:16]	(SS + 7) := D[31:24]

符号说明

Mt	SHA 引擎的输入缓冲器 $0 \leq t \leq 15$; 32 位字
SS	密钥的起始地址 (80H)
PP	存储器页的起始地址 见图 5 所示存储器映像
(SP + n)	暂存器第 n 字节
MPX	MPX[7] = 0; MPX[6] = 0; MPX[5:0] = (SP + 0)[5:0]
D, E	32 位字, 160 位 SHA 结果的一部分

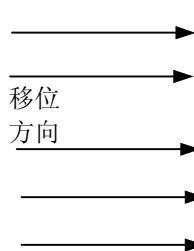
发出 Compute Next Secret 命令后, 主控制器必须提供一个 2 字节的地址, 用于指定提供 256 位 SHA 输入数据的存储器页。目的地址 TA1 的低五位不起作用。如果目的地址有效, 例如在 0000H 至 007FH 范围, 而且密钥未加写保护, SHA 引擎将启动, 并在 2ms 内计算出一个新的密钥, 然后自动将其复制到密钥寄存器中。替换密钥最多需要 10ms。在这段时间以及计算密钥的过程中, 1-Wire 总线上的电压一定不能低于 2.8V。复制完成后, DS1961S 用 AAH 字节填充暂存器。然后, 交替地发送 1 和 0 到总线, 直到主控制器发出复位脉冲为止。

由于暂存器的内容被用做部分密钥, 因此, 暂存器必须在发 Compute Next Secret 命令之前, 用 Write Scratchpad 命令给暂存器写入已知的 8 字节数据。否则的话, 新密钥将取决于以前的操作留在暂存器中的数据。

Copy Scratchpad [55H]

DS1961S 的数据存储器可以随意读取。然而，执行 Copy Scratchpad 要向存储器或寄存器页写入新的数据，就需要知道器件的密钥，并且能够执行 SHA-1 运算，以产生 160 位的 MAC，这样才可启动由暂存器到存储器的数据传送过程。主控制器可以在软件中计算 MAC，或者把 DS1963S 用做协处理器。协处理器方法的好处是密钥可以隐藏在协处理器 iButton 中。向 DS1961S 发送 MAC 运算结果的顺序如表 2 所示。表 3 说明了各种数据元素是如何进入 SHA 引擎的。有关 SHA 算法的说明，参见本文档的后续部分。

信息鉴定码传送顺序² 表 2

E[31:24]	E[23:16]	E[15:8]	E[7:0]	
D[31:24]	D[23:16]	D[15:8]	D[7:0]	
C[31:24]	C[23:16]	C[15:8]	C[7:0]	
B[31:24]	B[23:16]	B[15:8]	B[7:0]	
A[31:24]	A[23:16]	A[15:8]	A[7:0]	

发出 Copy Scratchpad 命令后，主控制器必须提供一个 3 字节的授权模式，这个数据应该通过紧邻此条命令的前一个 Read Scratchpad 命令获得。这 3 个字节的模式数据必须与三个地址寄存器（依次为 TA1，TA2，E/S）中的数据完全匹配。如果授权码匹配，而且目标存储器未加写保护，DS1961S 将启动它的 SHA 引擎，基于当前密钥、暂存器中的所有数据、所寻址的存储器页的前 28 个字节数据、以及 DS1961S 的地址码（不含 CRC 校验码），计算一个 160 位的 MAC。同时，主控制器也利用同样的数据计算一个 MAC，并把它发送给 DS1961S，以便证明它有权写 EEPROM。然后，主控制器需要等待 10ms，在此期间，1-Wire 总线上的电压一定不能低于 2.8V。如果 DS1961S 生成的 MAC 与主控制器计算的 MAC 相匹配，DS1961S 将置位 AA 标志，并将整个暂存器的内容复制到数据 EEPROM。作为一个复制成功的指示，主控制器将能读到一个 1 和 0 交替出现的序列，直到它发出复位脉冲。如果是一个全 0 图案，则说明未发生复制操作。

² 发送从寄存器 E 开始，最低有效位在前。

Copy Scratchpad 命令的 SHA-1 输入数据 表 3

目的地址：存储器页 0 至 3

M0[31:24] = (SS + 0)	M0[23:16] = (SS + 1)	M0[15:8] = (SS + 2)	M0[7:0] = (SS + 3)
M1[31:24] = (PP + 0)	M1[23:16] = (PP + 1)	M1[15:8] = (PP + 2)	M1[7:0] = (PP + 3)
M2[31:24] = (PP + 4)	M2[23:16] = (PP + 5)	M2[15:8] = (PP + 6)	M2[7:0] = (PP + 7)
M3[31:24] = (PP + 8)	M3[23:16] = (PP + 9)	M3[15:8] = (PP + 10)	M3[7:0] = (PP + 11)
M4[31:24] = (PP + 12)	M4[23:16] = (PP + 13)	M4[15:8] = (PP + 14)	M4[7:0] = (PP + 15)
M5[31:24] = (PP + 16)	M5[23:16] = (PP + 17)	M5[15:8] = (PP + 18)	M5[7:0] = (PP + 19)
M6[31:24] = (PP + 20)	M6[23:16] = (PP + 21)	M6[15:8] = (PP + 22)	M6[7:0] = (PP + 23)
M7[31:24] = (PP + 24)	M7[23:16] = (PP + 25)	M7[15:8] = (PP + 26)	M7[7:0] = (PP + 27)
M8[31:24] = (SP + 0)	M8[23:16] = (SP + 1)	M8[15:8] = (SP + 2)	M8[7:0] = (SP + 3)
M9[31:24] = (SP + 4)	M9[23:16] = (SP + 5)	M9[15:8] = (SP + 6)	M9[7:0] = (SP + 7)
M10[31:24] = MP	M10[23:16] = FAMC	M10[15:8] = SN0	M10[7:0] = SN1
M11[31:24] = SN2	M11[23:16] = SN3	M11[15:8] = SN4	M11[7:0] = SN5
M12[31:24] = (SS + 4)	M12[23:16] = (SS + 5)	M12[15:8] = (SS + 6)	M12[7:0] = (SS + 7)
M13[31:24] = FFH	M13[23:16] = FFH	M13[15:8] = FFH	M13[7:0] = 80H
M14[31:24] = 00H	M14[23:16] = 00H	M14[15:8] = 00H	M14[7:0] = 00H
M15[31:24] = 00H	M15[23:16] = 00H	M15[15:8] = 01H	M15[7:0] = B8H

符号说明

Mt	SHA 引擎的输入缓冲器 $0 \leq t \leq 15$; 32 位字
SS	密钥的起始地址 (80H)
PP	存储器页的起始地址 见图 5 所示存储器映像, 存储器页 0 至 3
(SP + n)	暂存器第 n 字节
MP	MP[7:4] = 0000 (对于 Copy Scratchpad) MP[3:0] = T8:T5 (等同于十六进制页码)
FAMC	家族代码 = 33H
SNx	器件的序列号 SN0 = 最低字节, SN5 = 最高字节 SN6 为 CRC 校验码

在复制数据到寄存器页的时候需要特别小心。为了防止无意中锁定某个特定功能寄存器或用户字节, 建议首先读取寄存器页, 然后在暂存器中修改后再全部写回。在写寄存器页 (或通过复制暂存命令建立密钥) 时, SHA 引擎的 M1 至 M7 输入数据将是当前密钥 (M1, M2), 寄存器页的当前内容 (M3, M4), 全部 64 位地址码 (M5, M6), 和 4 个字节 FFH (M7)。

Read Authenticated Page [A5H]

利用命令 Read Authenticated Page (读验证页), 主控制器可以获得全部或部分存储器页的数据和一个 MAC。通过 MAC, 主控制器能够判定存储在 DS1961S 中的密钥是否对于某特定应用有效。DS1961S 利用自己的密钥、指定存储器页的所有数据、自己的地址码和一个 3 字节的质询来计算 MAC, 这个 3 字节质询是由主控制器在发 Read Authenticated Page 命令之前提前写入暂存器的。为此, 主控制器可以使用 Write Scratchpad 命令, 采用数据存储器内的任意目的地址, 将质询写入暂存器。有关质询的部分被写入暂存器偏移地址 4, 5 和 6 中。作为另外一种选择, 主控制器也可

以将执行前一命令时，偶然留在暂存器中的数据作为一个质询。160 位 MAC 的传送方法与 Copy Scratchpad 命令中的情况完全一样，见表 2，只是数据流向改为由 DS1961S 至主控制器。执行 Read Authenticated Page 命令时输入 SHA 引擎的数据见表 4 所示。

主控制器发出命令代码并指定了有效的目的地址后，它将收到从目的地址开始，一直到数据页末尾的存储器页数据、一个 FFH 字节和一个反码的 CRC，该 CRC 码用于校验命令代码、目的地址、已传送的数据和 FFH 字节。CRC 校验码接收完毕后，主控制器等待 2.0ms，在此期间，1-Wire 总线上的电压不能低于 2.8V。在这段时间内，DS1961S 的 SHA 引擎利用密钥、选定页的 32 个数据字节、器件的地址码（不包括 CRC 校验码）和 3 字节质询计算 MAC。然后，主控制器就可读取 160 位 MAC，随后是一个反码的 CRC，以确保数据传输的可靠性。如果在 CRC 校验码后主控继续读取数据，它收到的将是交替的 0 和 1，直至发出复位脉冲。

Read Authenticated Page 命令的 SHA-1 输入数据 表 4

M0[31:24] = (SS + 0)	M0[23:16] = (SS + 1)	M0[15:8] = (SS + 2)	M0[7:0] = (SS + 3)
M1[31:24] = (PP + 0)	M1[23:16] = (PP + 1)	M1[15:8] = (PP + 2)	M1[7:0] = (PP + 3)
M2[31:24] = (PP + 4)	M2[23:16] = (PP + 5)	M2[15:8] = (PP + 6)	M2[7:0] = (PP + 7)
M3[31:24] = (PP + 8)	M3[23:16] = (PP + 9)	M3[15:8] = (PP + 10)	M3[7:0] = (PP + 11)
M4[31:24] = (PP + 12)	M4[23:16] = (PP + 13)	M4[15:8] = (PP + 14)	M4[7:0] = (PP + 15)
M5[31:24] = (PP + 16)	M5[23:16] = (PP + 17)	M5[15:8] = (PP + 18)	M5[7:0] = (PP + 19)
M6[31:24] = (PP + 20)	M6[23:16] = (PP + 21)	M6[15:8] = (PP + 22)	M6[7:0] = (PP + 23)
M7[31:24] = (PP + 24)	M7[23:16] = (PP + 25)	M7[15:8] = (PP + 26)	M7[7:0] = (PP + 27)
M8[31:24] = (PP + 28)	M8[23:16] = (PP + 29)	M8[15:8] = (PP + 30)	M8[7:0] = (PP + 31)
M9[31:24] = FFH	M9[23:16] = FFH	M9[15:8] = FFH	M9[7:0] = FFH
M10[31:24] = MP	M10[23:16] = FAMC	M10[15:8] = SN0	M10[7:0] = SN1
M11[31:24] = SN2	M11[23:16] = SN3	M11[15:8] = SN4	M11[7:0] = SN5
M12[31:24] = (SS + 4)	M12[23:16] = (SS + 5)	M12[15:8] = (SS + 6)	M12[7:0] = (SS + 7)
M13[31:24] = (SP + 4)	M13[23:16] = (SP + 5)	M13[15:8] = (SP + 6)	M13[7:0] = 80H
M14[31:24] = 00H	M14[23:16] = 00H	M14[15:8] = 00H	M14[7:0] = 00H
M15[31:24] = 00H	M15[23:16] = 00H	M15[15:8] = 01H	M15[7:0] = B8H

符号说明

Mt	SHA 引擎的输入缓冲器 $0 \leq t \leq 15$; 32 位字
SS	密钥起始地址 (80H)
PP	存储器页起始地址 见图 5 所示存储器映像，存储器页 0 至 3
FAMC	家族代码 = 33H
MP	MP[7:4] = 0100, MP[3:0] = T8:T5 (等同于十六进制页码)
SNx	器件的 ROM 序列号 SN0 = 最低字节, SN5 = 最高字节
(SP + n)	暂存器第 n 字节

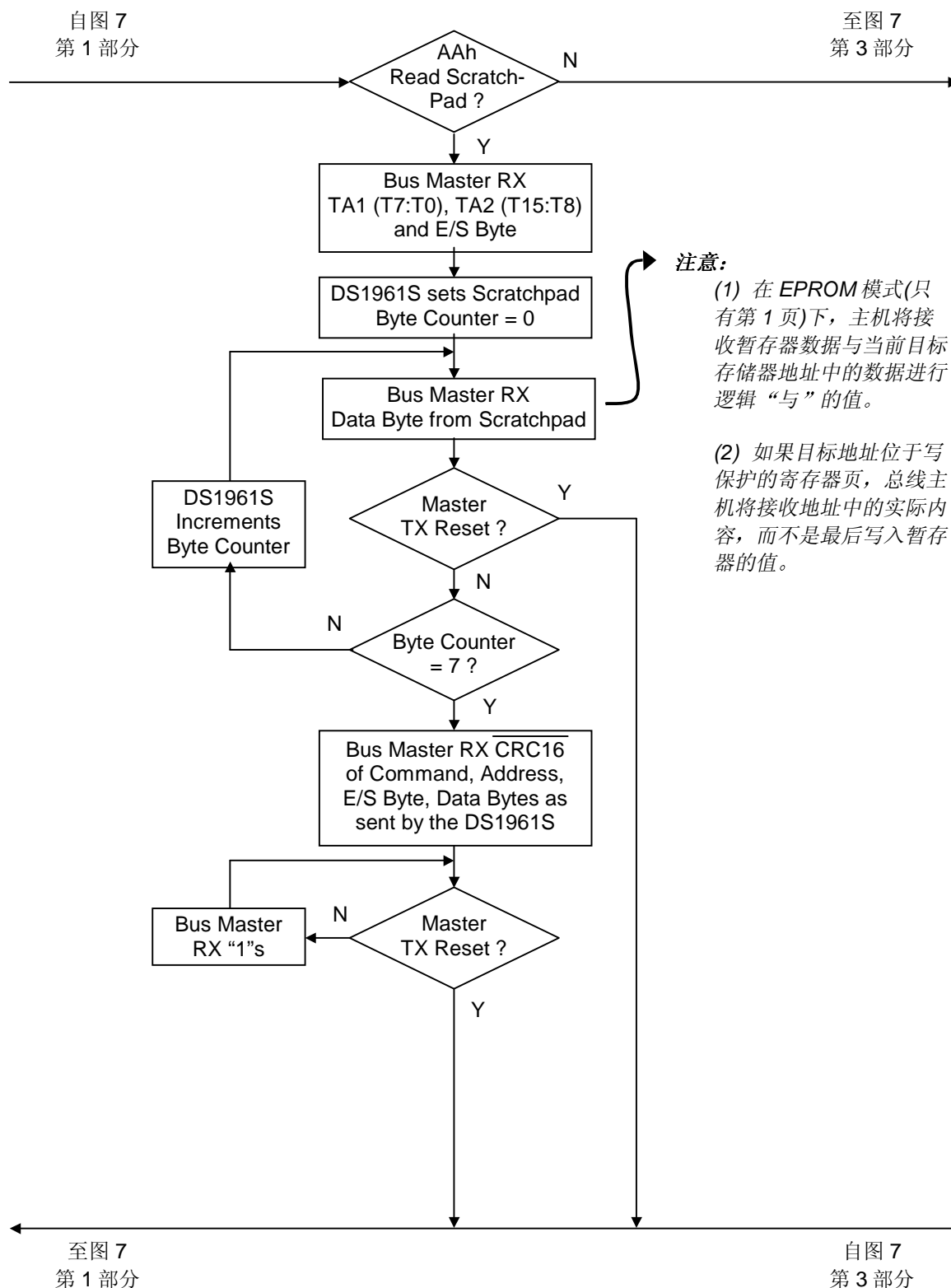
Read Memory [F0H]

Read Memory（读存储器）可以用来读取除密钥之外的所有存储器。尝试读取密钥时不会获得任何数据。发出命令后，主控制器须提供 2 个字节的地址。这两个字节之后，主控制器读取从目的地址开始的数据，可以一直读到地址 0097H。如果继续读，结果将是逻辑 1。应该注意的是，目的地址寄存器将指向最后一个读取的字节。结束偏移量/数据状态字节和暂存器不受影响。

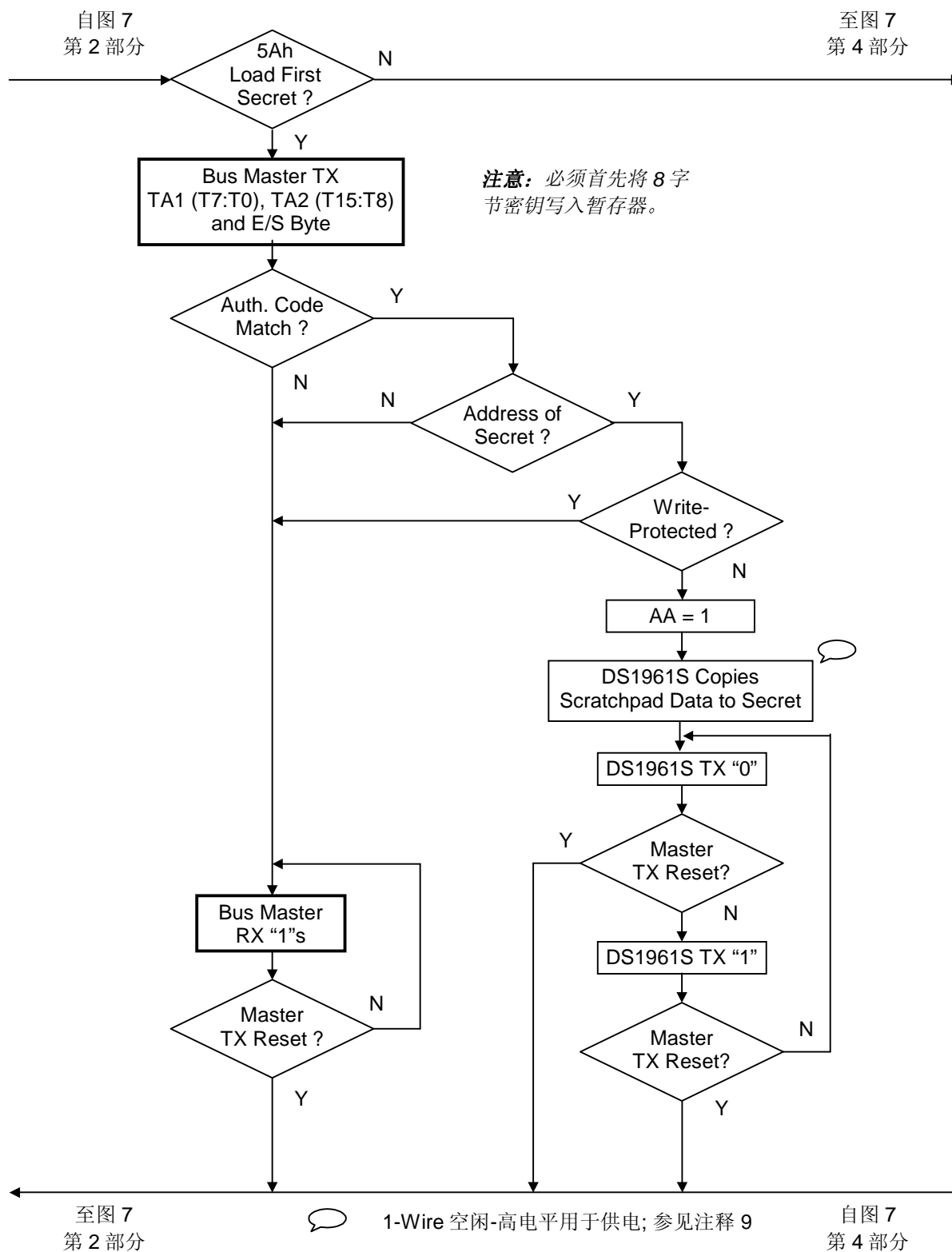
DS1961S 提供的硬件手段能够保证写入存储单元的数据正确无误。为了保证在 1-Wire 环境下读取数据的可靠性，同时提高数据传输的速率，建议将数据按照存储器页的大小进行分组。然后，在每个分组内包含一个由主控制器计算的、针对每页数据的 16 位 CRC 校验码。这样，主控制器就不必多次重复地读取一页数据来检验数据的正确与否，从而保证了快速、无误地传输数据（推荐的文件结构参见应用笔记 114，有时也称之为 TMEX 格式）。

存储器 and SHA 功能流程

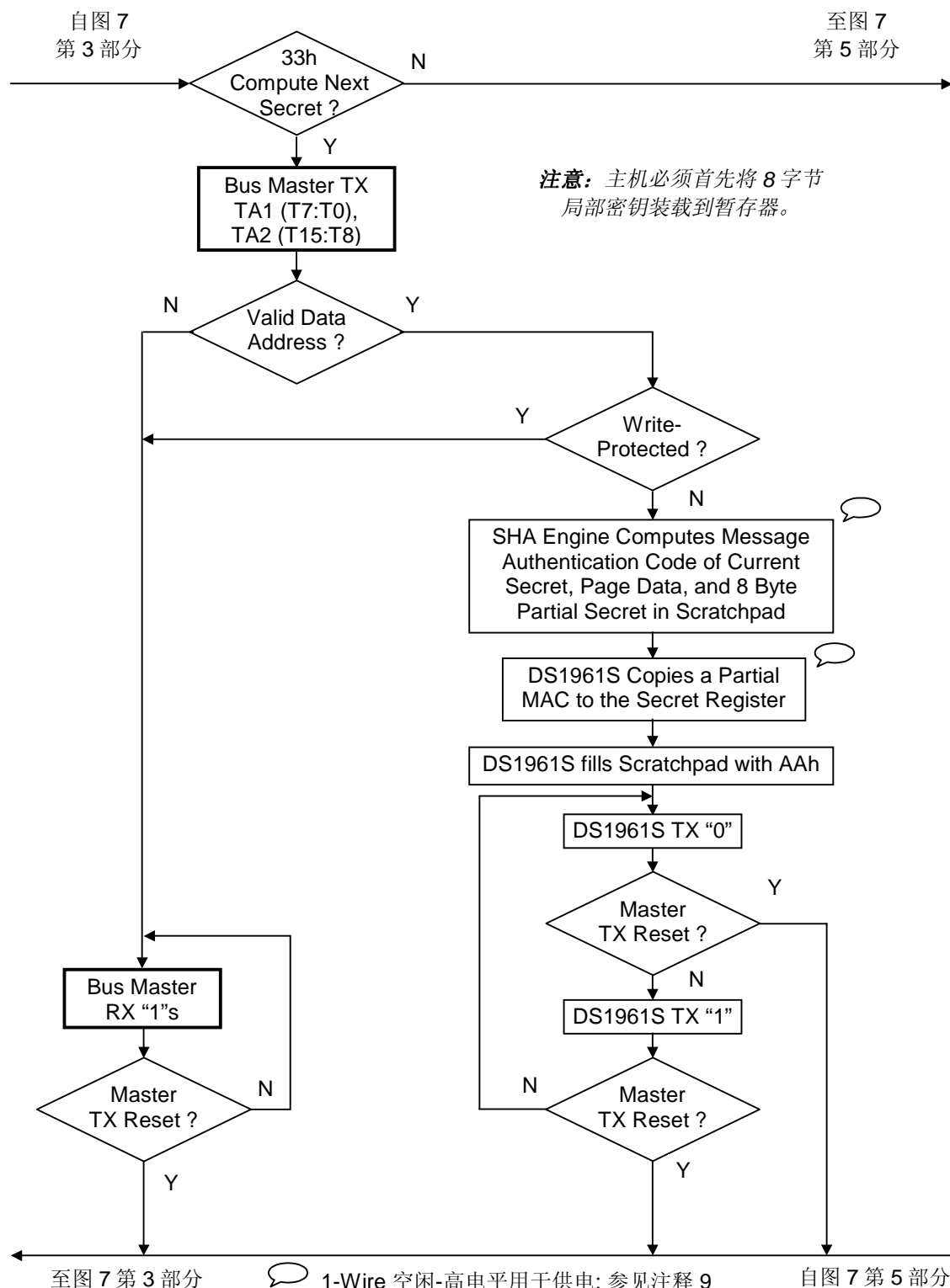
图 7 (续)



存储器 and SHA 功能流程 图 7 (续)

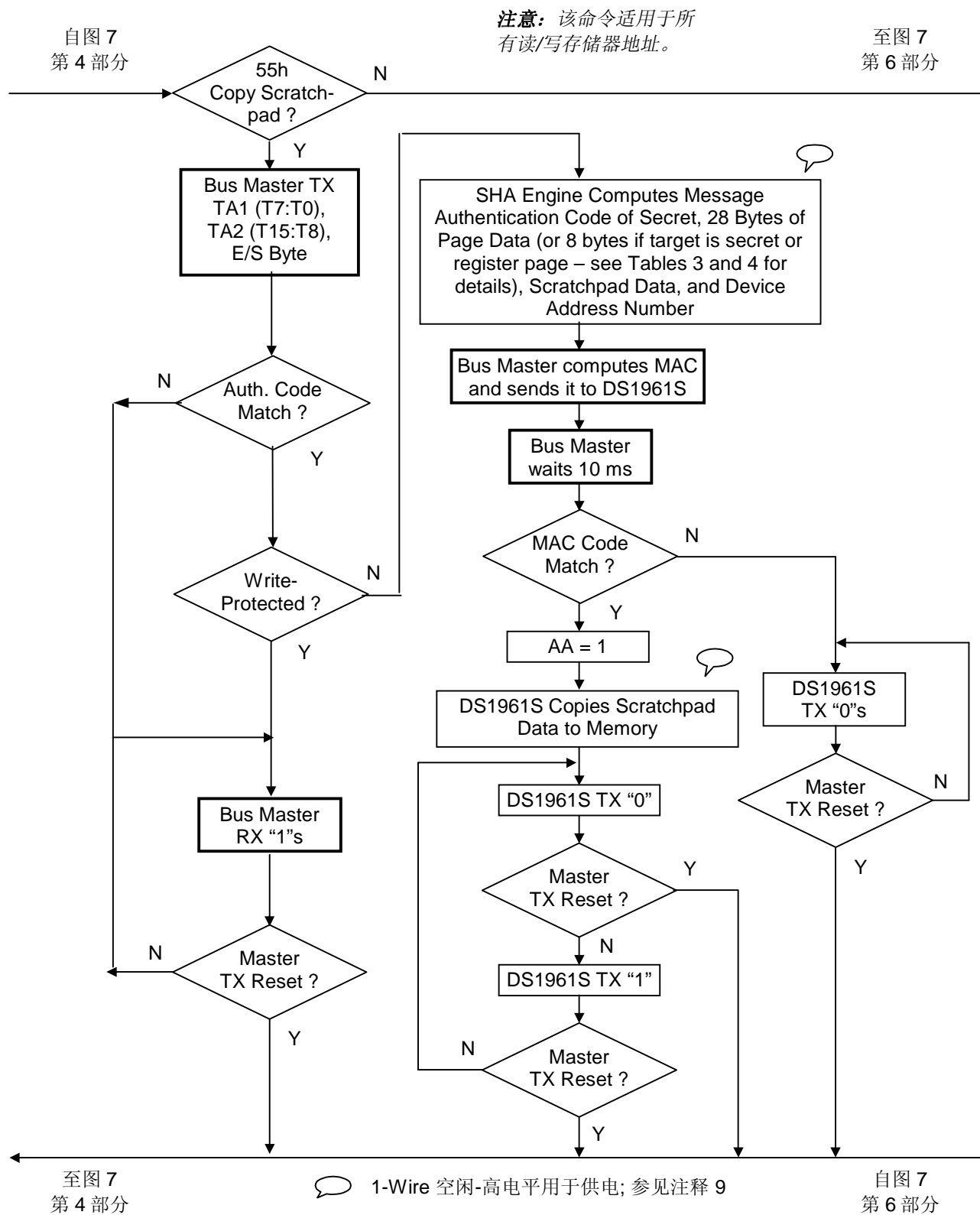


存储器 and SHA 功能流程 图 7 (续)



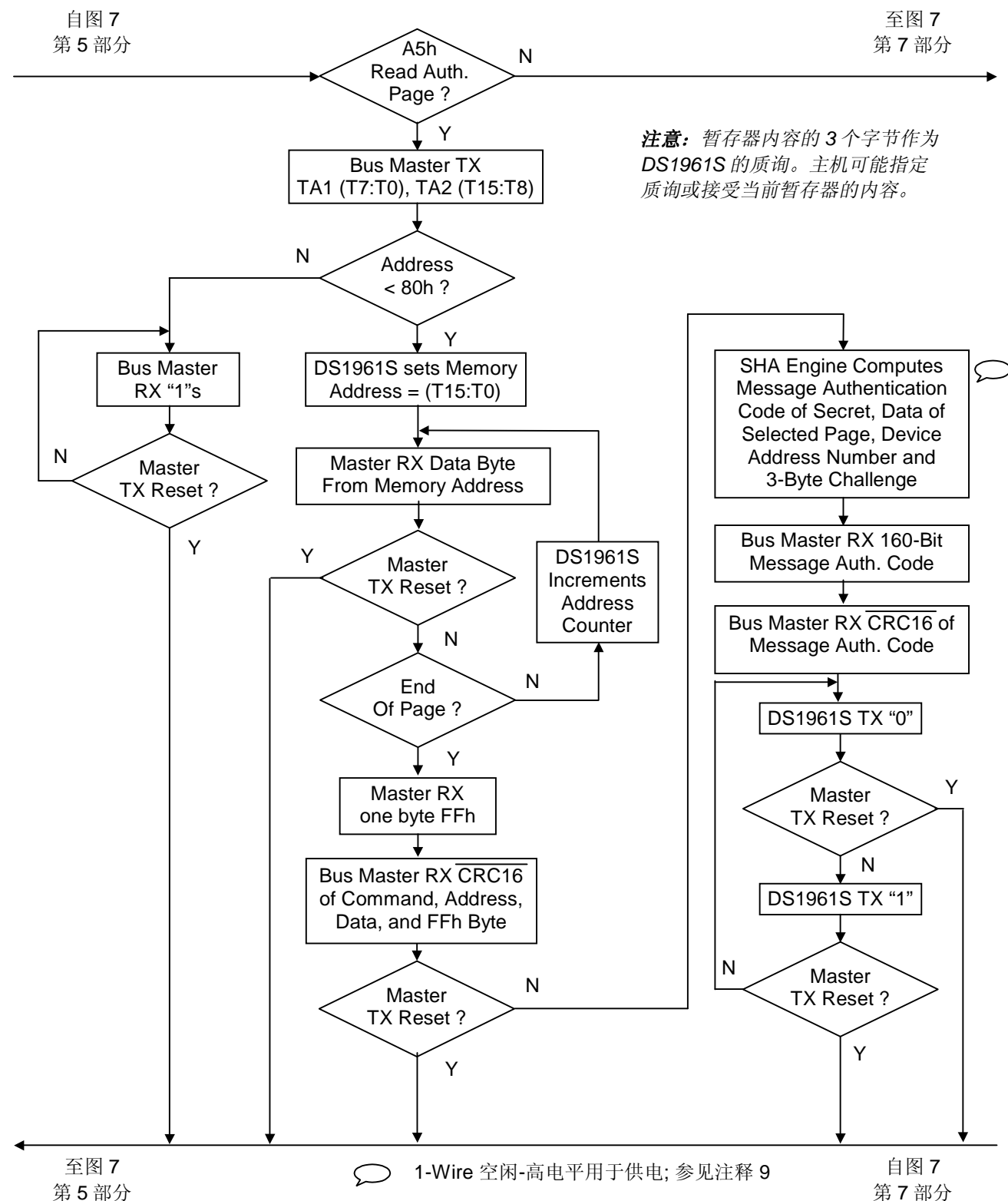
存储器 and SHA 功能流程

图 7 (续)



存储器 and SHA 功能流程

图 7 (续)



SHA-1 算法

以下有关 SHA 算法的说明译自安全散列标准 (Secure Hash Standard) SHA-1 文档, 该文档可从 NIST 网站下载 (www.itl.nist.gov/fipspubs/fip180-1.htm)。该算法采用 16 个 32 位字 M_t ($0 \leq t \leq 15$) 作为输入数据, 如表 1, 2 和 4 所示, 分别被用于 Compute Next Secret, Copy Scratchpad 和 Read Authenticated Page 命令。SHA 算法涉及到一个称为 W_t ($0 \leq t \leq 79$) 的 80 个 32 位字的序列, 一个称为 K_t ($0 \leq t \leq 79$) 的 80 个 32 位字的序列, 一个布尔函数 $f_t(B, C, D)$ ($0 \leq t \leq 79$), 其中 B, C 和 D 为 32 位字, 以及另外三个 32 位字, 称为 A, E 和 TMP 。SHA 算法用到的操作有不带进位的算术加 (“+”), 逻辑反或 1 的补码 (“\”), 异或 (“ \oplus ”), 逻辑与 (“ \wedge ”), 逻辑或 (“ \vee ”), 赋值 (“:=”), 以及 32 位字的循环移位。表达式 “ $S^n(X)$ ” 表示将 X 向左循环移 n 位, X 是一个 32 位字。

函数 f_t 定义如下:

$$\begin{aligned} f_t(B, C, D) &= (B \wedge C) \vee ((B \vee) \wedge D) & (0 \leq t \leq 19) \\ &B \oplus C \oplus D & (20 \leq t \leq 39) \\ &(B \wedge C) \vee (B \wedge D) \vee (C \wedge D) & (40 \leq t \leq 59) \\ &B \oplus C \oplus D & (60 \leq t \leq 79) \end{aligned}$$

序列 W_t ($0 \leq t \leq 79$) 定义如下:

$$\begin{aligned} W_t &:= M_t & (0 \leq t \leq 15) \\ &S^1(W_{t-3} \oplus W_{t-8} \oplus W_{t-14} \oplus W_{t-16}) & (16 \leq t \leq 79) \end{aligned}$$

序列 K_t ($0 \leq t \leq 79$) 定义如下:

$$\begin{aligned} K_t &:= 5A827999H & (0 \leq t \leq 19) \\ &6ED9EBA1H & (20 \leq t \leq 39) \\ &8F1BBCDCH & (40 \leq t \leq 59) \\ &CA62C1D6H & (60 \leq t \leq 79) \end{aligned}$$

变量 A, B, C, D, E 初始化如下:

$$\begin{aligned} A &:= 67452301H \\ B &:= EFCDAB89H \\ C &:= 98BADCFEH \\ D &:= 10325476H \\ E &:= C3D2E1F0H \end{aligned}$$

当 t 从 0 循环至 79, 执行了下面的一系列计算后, 160 位 MAC 是 A, B, C, D 和 E 的串联 (不考虑任何进位):

$$\begin{aligned} TMP &:= S^5(A) + f_t(B, C, D) + W_t + K_t + E \\ E &:= D \\ D &:= C \\ C &:= S^{30}(B) \\ B &:= A \\ A &:= TMP \end{aligned}$$

主控制器可以按照表 3 所示的寄存器和位顺序, 通过 Read Authenticated Page 命令读取 MAC。与 Copy Scratchpad 命令相比, 位的传送顺序是一样的; 不过, 主控制器必须计算 MAC, 并将其发

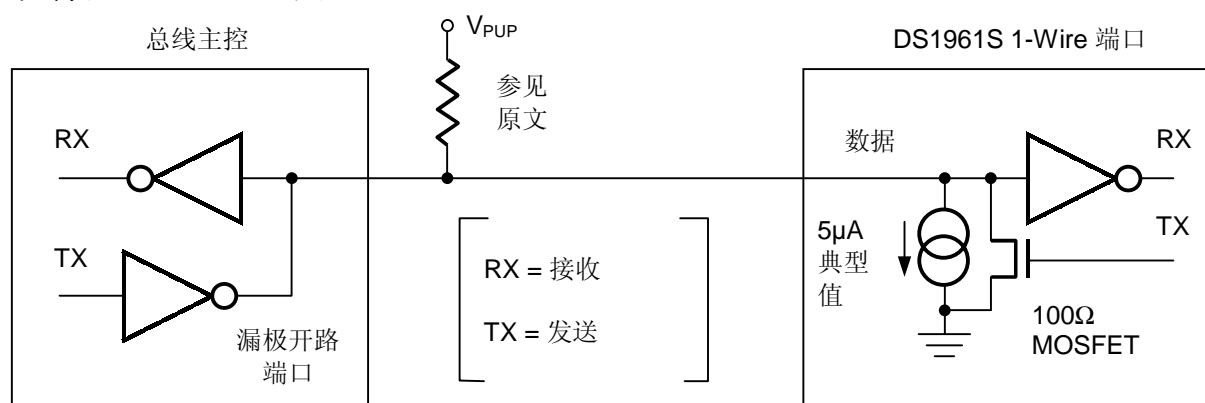
送给 DS1961S。在执行 Compute Next Secret 命令时 MAC 不会暴露。SHA 运算寄存器 E 和 D 的内容被直接复制到密钥寄存器，如表 1 所示。

1-Wire 总线系统

1-Wire 单总线系统是用一根数据线连接单个主机和一台或多台从机设备的系统。任何情况下，DS1961S 都作为从机设备来使用。总线上主机为典型的微处理器，对单总线系统的论述分为以下三个部分：硬件结构、处理流程和 1-Wire 信令（信号类型和定时）。1-Wire 通信协议规定总线的收发按照特殊时隙下的总线状态进行，由主机发出的同步脉冲下降沿初始化。需要了解更多关于通讯协议详细描述，请参考 *Book of DS19xx iButton Standards* 第四章。

硬件配置

图 8



硬件配置

1-Wire 单总线系统中只定义了一根数据线，所以，保证在适当的时间驱动总线上的每个设备是非常重要的。为使上述操作易于实现，挂接在 1-Wire 总线上的每个装置必须都带有一个漏极开路或三态端口连接数据线。DS1961S 的 1-Wire 端口是漏极开路的，其内部等效电路如图 8 所示。多点总线由连接了多个从机设备的 1-Wire 总线组成。在标准速率下，1-Wire 总线的最大速率为 16.3kbps。在高速模式下，速率可达 142kbps。为了在任意速率下执行存储器和 SHA 操作命令，DS1961S 需要的 1-Wire 上拉电阻的最大值为 2.2kΩ。当与几个 DS1961S 同时通信时，例如安装同样的密钥给几个器件，在器件从暂存器向 EEPROM 传送数据和更新防篡改寄存器时，应该利用一个上拉至 V_{PUP} 的低阻抗上拉旁路这个电阻。

1-Wire 总线的空闲状态是高电平。如果由于某种原因需暂停通信，如果想稍候恢复通信的话，总线必须保持在空闲状态。如果不是这样，而总线又处于低电平状态超过 16μs（高速模式）或 120μs（常规速度），那么总线上的一个或多个器件将被复位。

处理流程

通过 1-Wire 端口访问 DS1961S 的协议如下：

- 初始化
- ROM 操作命令
- 存储器或 SHA 操作命令
- 交易/数据

初始化

1-Wire 总线上所有的传输操作均从初始化过程开始。初始化过程由主机发出的复位脉冲和从机发出的在线应答脉冲（Presence Pulse）组成。在线应答脉冲使主机检测到 DS1961S 挂接在总线上，并且已经准备就绪。详细内容请参阅 *1-Wire* 信令一节。

ROM 功能命令

一旦主机检测到在线应答脉冲，就可以发出 DS1961S 支持的七条 ROM 功能命令。所有 ROM 操作命令的长度为八位。以下列出了这些命令的简要介绍（参考图 9 中的流程图）：

Read ROM [33H]

此条命令允许主机读取 DS1961S 的 8 位家族码、48 位唯一的序列号和 8 位 CRC 校验码。此命令适用于总线上只有一个从机的情况。如果总线上连接了多个从机设备，当同一时间每个从机设备都响应此条命令时，就必然要发生数据冲突（漏极开路输出将产生一个线与结果）。结果导致主机读取的家族码和 48 位序列号无效。

Match ROM [55H]

Match ROM 命令跟随 64 位地址码，允许主机访问多从机总线系统中某个特定的 DS1961S。只有与 64 位地址码完全匹配的 DS1961S 才会响应主机随后发出的存储器功能命令。所有其它从机将处于等待复位脉冲状态。这条命令既适用于单从机系统，也适用于多从机系统。

Search ROM [F0H]

系统初次上电时，总线主机可能并不知道 1-Wire 总线上从机设备的数目和它们的 64 位地址码，而 Search ROM 命令能够使得总线主机通过排除法来检测出总线上所有从机设备的 64 位地址码。Search ROM 过程其实只是 3 个简单步骤的重复：读一位、读此位的补码，然后写这一位的期望值，主机对地址码的每一位数据都执行这简单的 3 步操作。在完全通过一次审查操作后，总线主机就能读出一台从机设备的 64 位内容。其余从机设备的地址码可由另外的操作检测出来。关于 Search ROM 命令更全面的讨论，请参考 *Book of DS19xx iButton Standards* 第五章，并且在此章中还包括一个实例。

Skip ROM [CCH]

Skip ROM 命令在单从机总线系统中允许主机直接访问存储器和 SHA 功能，而无须提供 64 位地址码，节省时间。如果总线上挂接了不止一个从机设备，而且在 Skip ROM 命令后发出了一条 Read 命令，总线上的从机设备就会同时传送数据，从而引起数据冲突（漏极开路输出将产生一个线与结果）。

Overdrive Skip ROM [3CH]

在单点总线上发出该命令的时候，总线主控不需要 64 位的地址码就可以访问存储器和 SHA 功能，从而节省了时间。不同于通常的 Skip ROM 命令，Overdrive Skip ROM 命令将 DS1961S 设置成高速模式（OD = 1）。该命令代码后面的所有通信都发生在高速模式下，直到有一个最短持续 480μs 的复位脉冲把总线上的所有器件都复位到标准速率（OD = 0）。

在多点总线上发出该命令时，所有支持高速模式的器件都被置为高速模式。随后，为了寻址特定的高速模式器件，必须发出一个高速模式的复位脉冲，接着运用 Match ROM 或 Search ROM 命令。这将加速搜索过程。如果总线上有多个支持高速模式的从机，并且 Overdrive Skip ROM 命令后接着就是 Read 命令，那么由于多个从机同时发送，总线上就会发生数据冲突（多个开漏输出下拉将产生线与结果）。

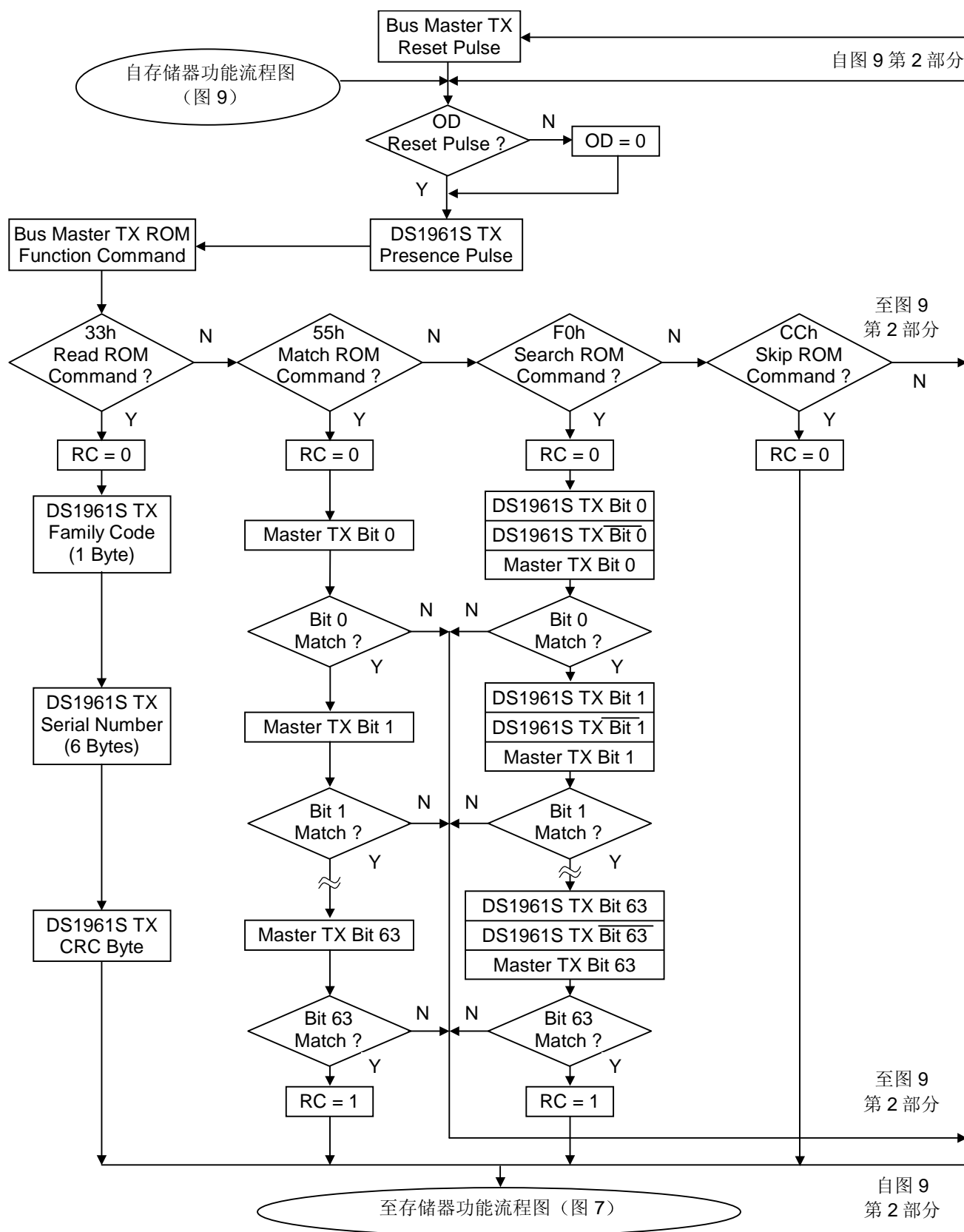
Overdrive Match ROM [69H]

通过 Overdrive Match ROM 命令，后接以高速模式发送的 64 位地址码，总线主控可以在多点总线上找到某个特定的 DS1961S，并将它设置成高速模式。只有 64 位地址码精确匹配的 DS1961S 才会响应后续的存储器或 SHA 操作命令。那些通过前面的 Overdrive Skip 或 Overdrive Match 命令已被置为高速模式的从机将继续保持高速模式。直到有一个最短持续时间 480μs 的复位脉冲发出后，所有高速模式的器件将返回常规速度。命令 Overdrive Match ROM 适用于总线上有单个或多个器件的情况。

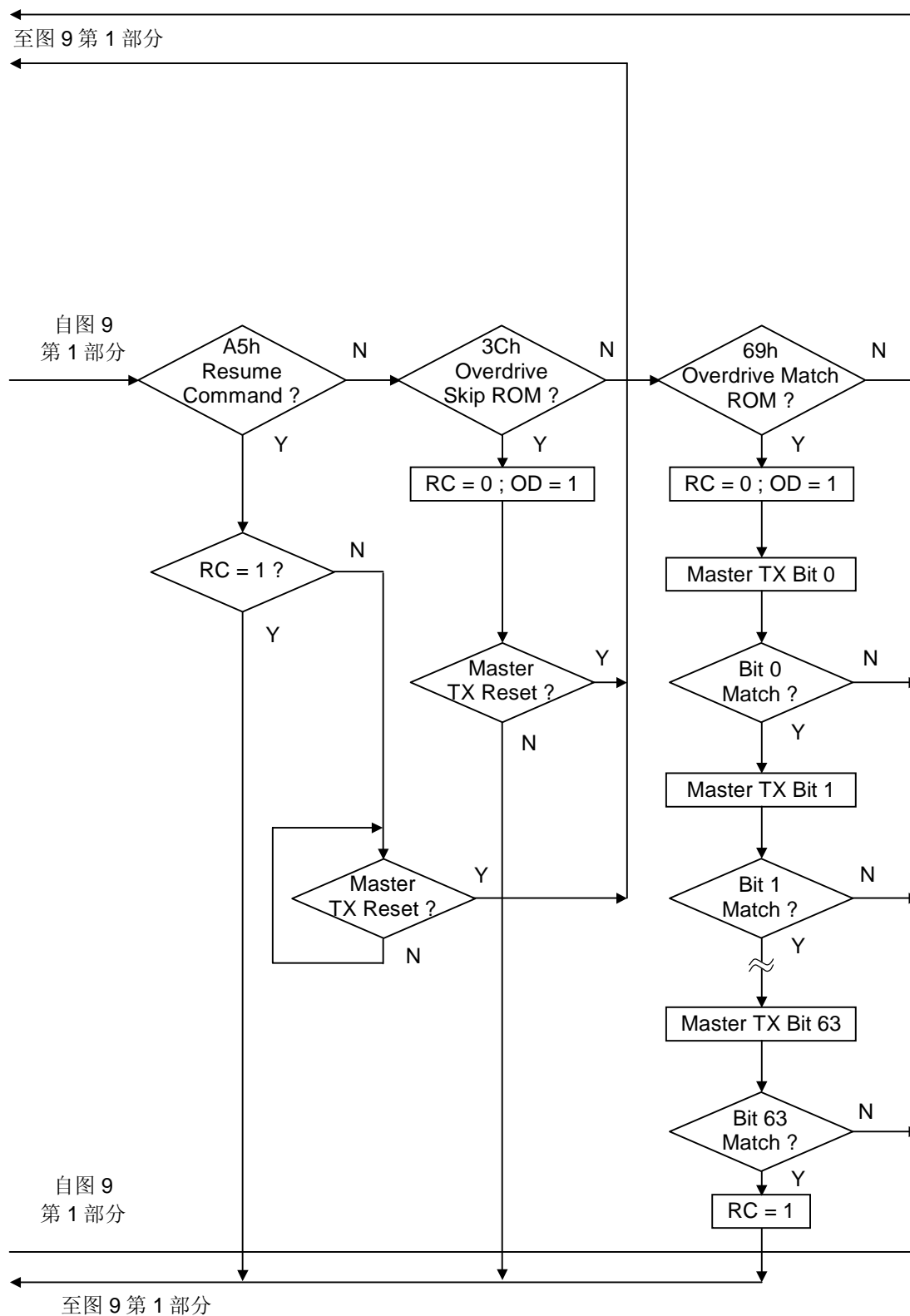
Resume Command [A5H]

在一个典型应用中，要写满一个 32 字节的存储器页，往往需要多次访问 DS1961S。这意味着在多点环境中，每次访问都要重复执行 Match ROM 命令和发送 64 位地址码。为了提高多点环境中的数据吞吐率，设置了 Resume Command 功能。该功能检测 RC 位的状态，如果置位，就直接传递控制给存储器和 SHA 功能，类似于 Skip ROM 命令。设置 RC 位的唯一方法是成功地执行 Match ROM，Search ROM 或 Overdrive Match ROM 命令。一旦设置了 RC 位，利用 Resume Command 功能就可重复访问同一器件。对于总线上另一器件的访问将清除 RC 位，以防两个或更多的器件同时响应 Resume Command 功能。

ROM 功能流程 图 9



ROM 功能流程 图 9 (续)

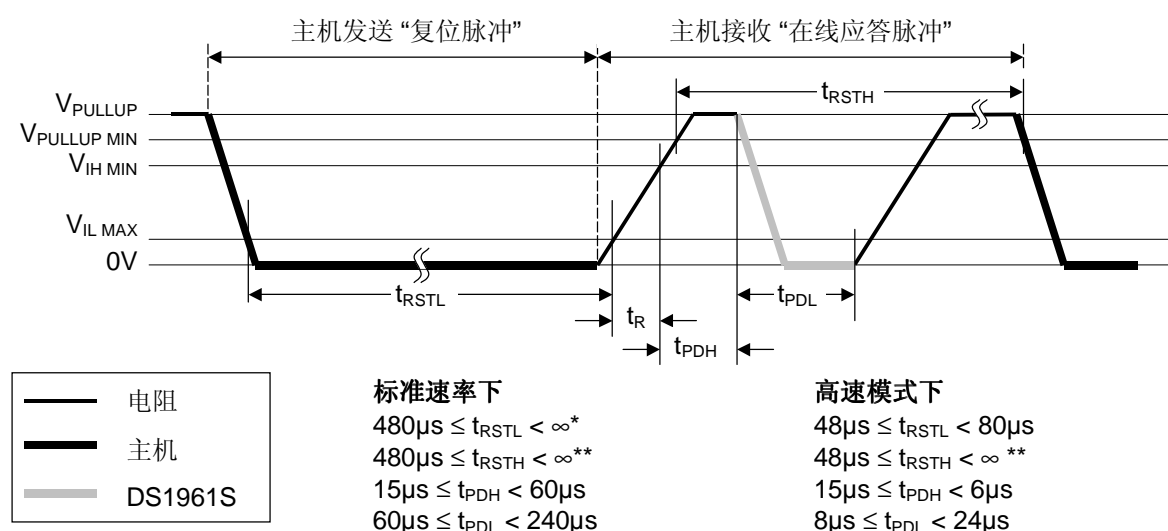


1-Wire 信号

为了保证数据的完整性，DS1961S 具有一个严格的信号协议。该协议在一条线上定义了四种类型的信号：包括复位脉冲和应答脉冲的复位序列，写 0，写 1，和读数据。除了应答脉冲以外，所有其它信号均由总线主控发出。DS1961S 能够以两种不同速度通信：常规速度和高速模式。如果没有明确设定为高速模式，DS1961S 就以常规速度通信。高速模式下，所有波形均采用快速定时。

与 DS1961S 进行通信所需的初始化时序见图 10。复位脉冲后的应答脉冲表明 DS1961S 已经准备好发送或接收数据。总线主控发送 (TX) 复位脉冲 (t_{RSTL} ，标准速率下最小是 $480\mu s$ ，高速模式下是 $48\mu s$)。然后总线主控释放数据线，进入接收模式 (RX)。通过上拉电阻 1-Wire 总线进入高电平状态。在数据引脚上检测到上升沿后，DS1961S 等待 (t_{PDH} ，正常速率下是 $15\mu s$ 至 $60\mu s$ ，高速下是 $2\mu s$ 至 $6\mu s$)，然后发送应答脉冲 (t_{PDL} ，正常速率下是 $60\mu s$ 至 $240\mu s$ ，高速下是 $8\mu s$ 至 $24\mu s$)。 $480\mu s$ 或更长时间的复位脉冲将使器件从高速模式下退出进入标准速率。如果 DS1961S 处于高速模式，而且复位脉冲也小于 $80\mu s$ ，那么器件仍处于高速模式。

初始化过程“复位和应答脉冲” 图 10



*为了避免 1-Wire 总线上的其它器件屏蔽掉中断信号， $t_{RSTL} + t_R$ 应当总是小于 $960\mu s$ 。

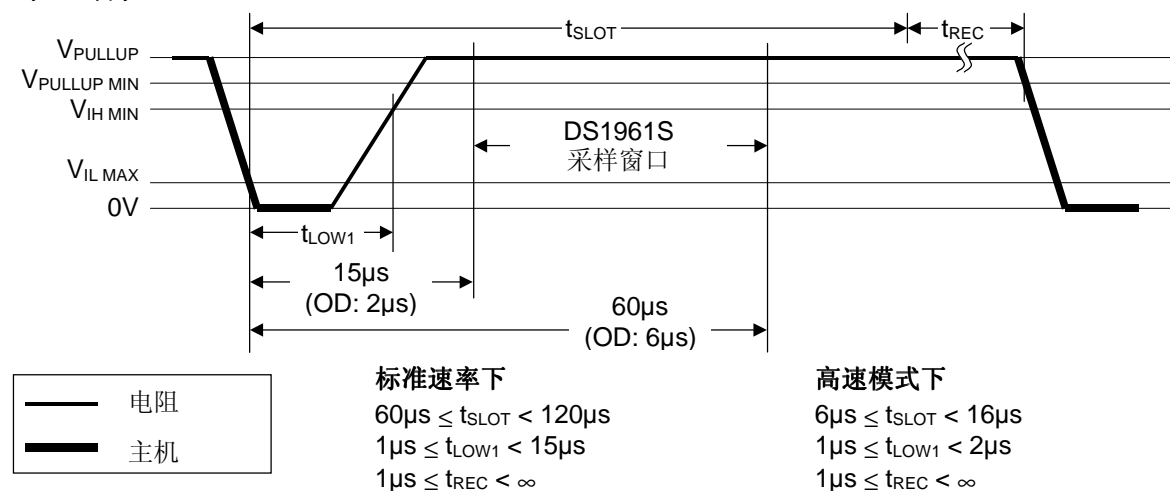
**包括恢复时间在内。

读/写时隙

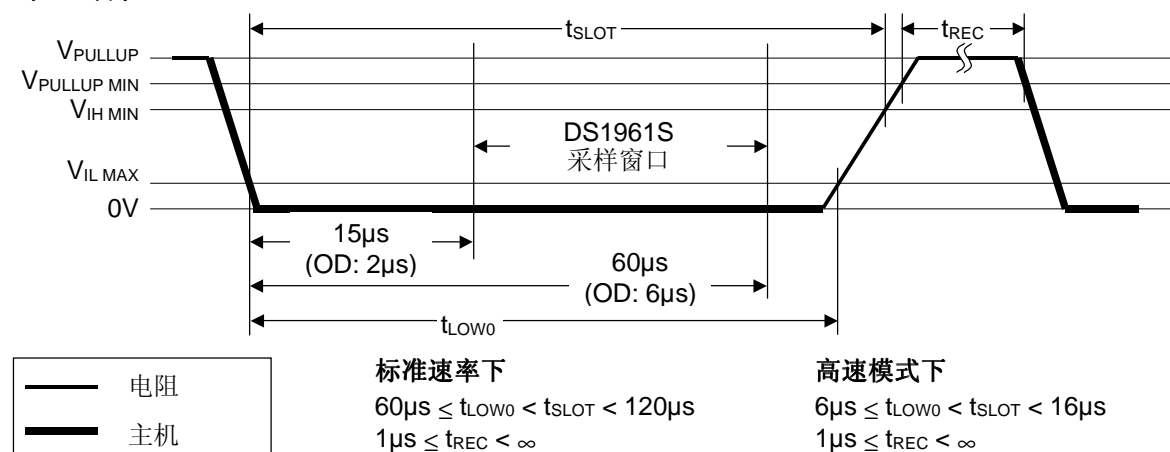
图 11 说明了写和读时隙的定义。主控通过拉低数据线来启动所有的时隙。通过触发内部延时电路，数据线的下降沿使 DS1961S 与主控同步。在写时隙中，延迟电路决定 DS1961S 何时采样数据线。对读时隙来说，如果将要发送一个零，那么延时电路决定了 DS1961S 将数据线拉低多长时间。如果数据位是 1，那么 DS1961S 将不会拉低数据线。

读/写时序图 图 11

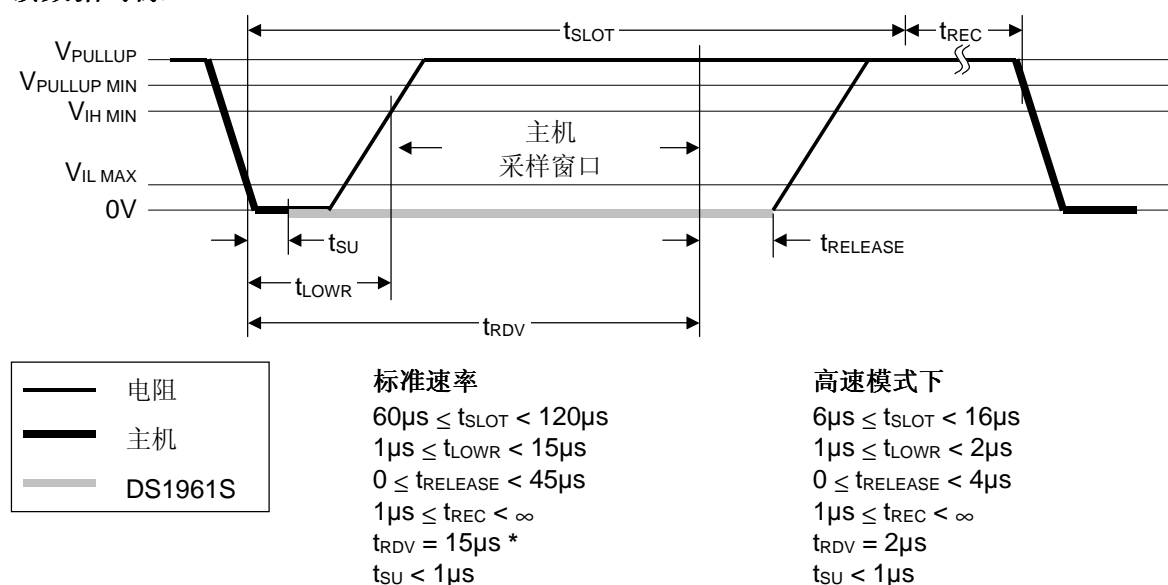
写 1 时序



写 0 时序



读数据时隙



*主控的最佳采样点在尽可能接近 t_{RDV} 的末端，同时又不超出 t_{RDV} 。对读 1 时隙而言，这会为上拉电阻留出足够的时间将数据线拉到高电平。对读零时隙来讲，这个值能够保证采样动作发生在最快的 1-Wire 器件释放总线之前 ($t_{\text{RELEASE}} = 0$)。

CRC 生成

DS1961S 有两种类型的循环冗余校验 (CRC)。其中一种是 8 位的，在出厂时就已经计算好了，并用激光写入 64 位 ROM 的最高字节中。该 CRC 的等价多项式是 $X^8 + X^5 + X^4 + 1$ 。为了确定 ROM 数据是否被无差错地读取，总线主控可用 64 位 ROM 的前 56 位计算 CRC 值，并将其与从 DS1961S 读来的值相比较。读 ROM 的时候，接收到的是 8 位 CRC 校验码的原码形式（未求反的）。

另一类 CRC 是 16 位的，根据标准的 CRC16 多项式函数 $X^{16} + X^{15} + X^2 + 1$ 产生。该 CRC 校验码用于检测执行 Read Authenticated Page 命令时的错误，或者在读或写暂存器的时候，快速检验数据传送的正确性。在 iButton 扩展文件结构中用于差错检验的也是同一种 CRC。与 8 位 CRC 校验码不同的是，16 位 CRC 校验码通常是以反码的形式发送或回送。DS1961S 芯片内部的 CRC 发生器（图 12）用于在图 7 所示的命令流程中计算一个新的 16 位 CRC 校验码。总线主控通过比较由器件读来的 CRC 校验码和自己根据数据计算出的 CRC 校验码，据此来决定是继续某一操作还是重读有 CRC 错误的数据部分。

在 Write Scratchpad 命令中，首先清除 CRC 发生器，然后移入命令代码，目的地址 TA1 (T2 至 T0 均置 0) 和 TA2，以及所有主控发送的数据字节。只有当暂存器被写满的时候 DS1961S 才发送该 CRC 校验码。

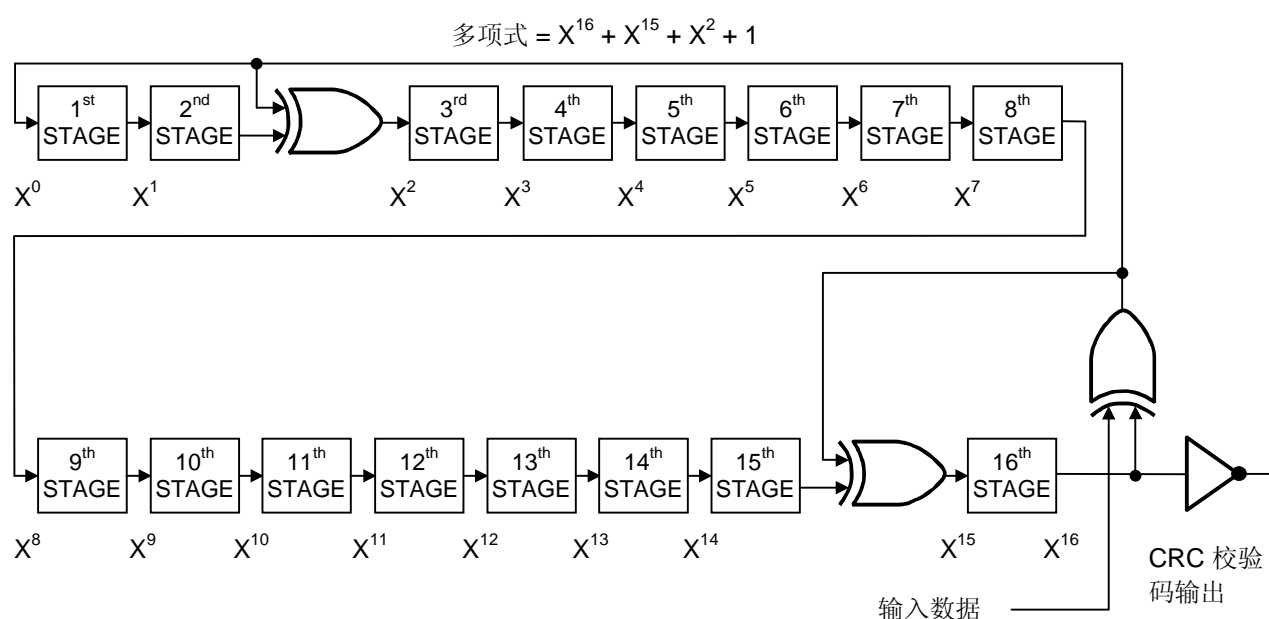
在 Read Scratchpad 命令中，首先清空 CRC 发生器，然后移入命令代码，目的地址 TA1 和 TA2，E/S 字节，和暂存器数据，它们可能已被 DS1961S 调整过（见 Write Scratchpad 命令），最后产生了 CRC 校验码。只有读到暂存器末尾的时候，DS1961S 才发送该 CRC 校验码。

在 Read Authenticated Page 命令中，16 位 CRC 校验码是清空 CRC 发生器并移入命令字节、两个地址字节、数据字节、和 FFH 字节后的结果。跟在 MAC 结果后面的 CRC 校验码是在清空 CRC 发生器后，按照主控接收的位序移入 160 位 MAC 后产生的。

关于产生 CRC 校验码的详细资料，以及用硬件和软件实现的具体实例，参见 *Book of DS19xx iButton Standards*。

CRC-16 硬件和多项式描述

图 12



极限参数*

任意引脚对地的电压范围	-0.5V 至+5.5V
工作温度范围	-40°C 至+85°C
存储温度范围	-55°C 至+125°C
焊接温度范围	见 J-STD-020A 规范

* 这只是一个应力条件下的参数，并不意味着器件可以在符合或超出该规定所提及的工作条件下执行功能。长期暴露器件于极限条件会影响其可靠性。

直流电特性(V_{PUP} = 2.8V 至 5.25V; -40°C 至 +85°C)

参数	符号	最小	典型	最大	单位	注释
1-Wire 输入高电平	V _{IH}	2.2			V	1,7
1-Wire 输入低电平	V _{IL}			0.30	V	1,8
1-Wire 输出低电平 @ 4mA	V _{OL}			0.4	V	1
1-Wire 输出高电平	V _{OH}		V _{PUP}		V	1,2
输入负载电流	I _L		5		μA	3
编程电流	I _{LPROG}		500		μA	9

电容(T_A = +25°C)

参数	符号	最小	典型	最大	单位	注释
1-Wire I/O	C _{IN/OUT}		100	800	pF	5

使用寿命(V_{PUP} = 5.0V; T_A = +25°C)

参数	符号	最小	典型	最大	单位	注释
写/擦除次数	N _{CYCLE}	50k			-	
数据保持	t _{DRET}	10			年	

交流电特性**标准速率**(V_{PUP} = 2.8V 至 5.25V; -40°C 至 +85°C)

参数	符号	最小	典型	最大	单位	注释
时隙	t _{SLOT}	60		120	μs	
写 1 低电平时间	t _{LOW1}	1		15	μs	
写 0 低电平时间	t _{LOW0}	60		120	μs	
读操作低电平时间	t _{LOWR}	1		15	μs	
读数据有效	t _{RDV}		15		μs	10
释放时间	t _{RELEASE}	0	15	45	μs	
读数据建立	t _{SU}			1	μs	4
恢复时间	t _{REC}	1			μs	
复位高电平时间	t _{RSTH}	480			μs	
复位低电平时间	t _{RSTL}	480			μs	6
在线检测高电平	t _{PDHIGH}	15		60	μs	
在线检测低电平	t _{PDLOW}	60		240	μs	
编程时间	t _{PROG}			10	ms	
SHA 计算时间	t _{CSHA}		1	2	ms	9

交流电特性

高速模式

($V_{PUP} = 2.8V$ 至 $5.25V$; $-40^{\circ}C$ 至 $+85^{\circ}C$)

参数	符号	最小	典型	最大	单位	注释
时隙	t_{SLOT}	6		16	μs	
写 1 低电平时间	t_{LOW1}	1		2	μs	
写 0 低电平时间	t_{LOW0}	6		16	μs	
读低电平时间	t_{LOWR}	1		2	μs	
读数据有效	t_{RDV}		2		μs	10
释放时间	$t_{RELEASE}$	0	1.5	4	μs	
读数据建立	t_{SU}			1	μs	4
恢复时间	t_{REC}	1			μs	
复位高电平时间	t_{RSTH}	48			μs	
复位高电平时间	t_{RSTL}	48		80	μs	
在线检测高电平	t_{PDHIGH}	2		6	μs	
在线检测高电平	t_{PDLOW}	8		24	μs	
编程时间	t_{PROG}			10	ms	
SHA 计算时间	t_{CSHA}		1	2	ms	9

注释:

- 1) 所有电压都相对于地。
- 2) V_{PUP} = 外部上拉电压。
- 3) 输入负载是对地的。
- 4) 读数据建立时间是指主机为读一位数而必须将 1-Wire 总线拉低的时间。在下降沿的 $1\mu s$ 时间内数据保证有效。
- 5) 初次通电时, 数据引脚上的电容可能达 $800pF$ 。如果用 $5k\Omega$ 的电阻将数据线上拉至 V_{PUP} , 那么在上电 $5\mu s$ 后, 寄生电容不会影响到正常通信。
- 6) 复位低电平时间 (t_{RSTL}) 的最大值应当控制在 $960\mu s$ 内, 以允许中断信号, 否则, 它将屏蔽或遮蔽中断脉冲。
- 7) V_{IH} 是外部上拉电阻和 V_{PUP} 的函数。
- 8) 在一定的低电压条件下, V_{ILMAX} 必须被减少到最多 $0.5V$, 以确保正确的应答脉冲。 V_{IL} 是 V_{PUP} 和复位低电平时间的函数。
- 9) 在写 EEPROM 操作或计算信息鉴定码 (MAC) 的过程中, 1-Wire 总线上的电平一定不能低于 $2.8V$ 。计算 MAC 最多用 $2ms$, 将暂存器的数据复制到 EEPROM 最多用 $10ms$ 。为实现可靠通信, 我们建议 1-Wire 主控等待相应的最大执行时间。
- 10) 主控的最佳采样点应当尽可能地接近 t_{RDV} 的终止时间而又不超过 t_{RDV} 。对读 1 时隙而言, 这会留给上拉电阻足够的时间来使数据线恢复到高电平。对读零时隙而言, 这样可以保证在最快的 1-Wire 器件释放数据线以前读取数据 ($t_{RELEASE}=0$)。