DALLAS SEMICONDUCTOR

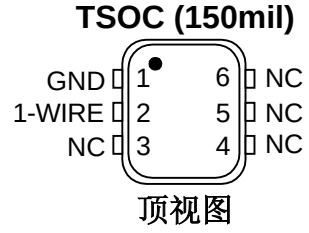
DS2432
带有 SHA-1 引擎保护的 1k 位 1-Wire® EEPROM

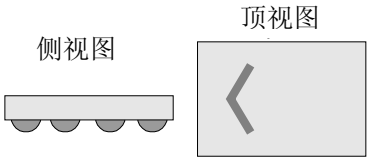
www.dalsemi.com

特性

- 1128 位 5V EEPROM 存储器，分为四页，每页 256 位，64 位只写密钥和多达五个通用读/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(“写入 0”，第 1 页)
- 唯一的、由工厂光刻并经过测试的 64 位注册号没有任何两个器件相同，保证绝对可溯
- 内置多点控制，保证兼容于其它 1-Wire 网络产品
- 将控制、寻址、数据和供电集于一个数据引脚
- 直接与微处理器的单个端口连接，通信速率达 16.3kbps
- 高速模式下速率可提高至 142kbps
- 低成本、6 引脚 TSOC 表面贴封装或倒装芯片
- 可以在-40°C 至 +85°C、2.8V 至 5.25V 宽压范围内进行读、写操作

引脚配置





封装的机械规格请参考： www.dalsemi.com。

定购信息

DS2432P	6 引脚 TSOC 封装
DS2432P/T&R	DS2432P 卷带
DS2432X	倒装芯片，卷带

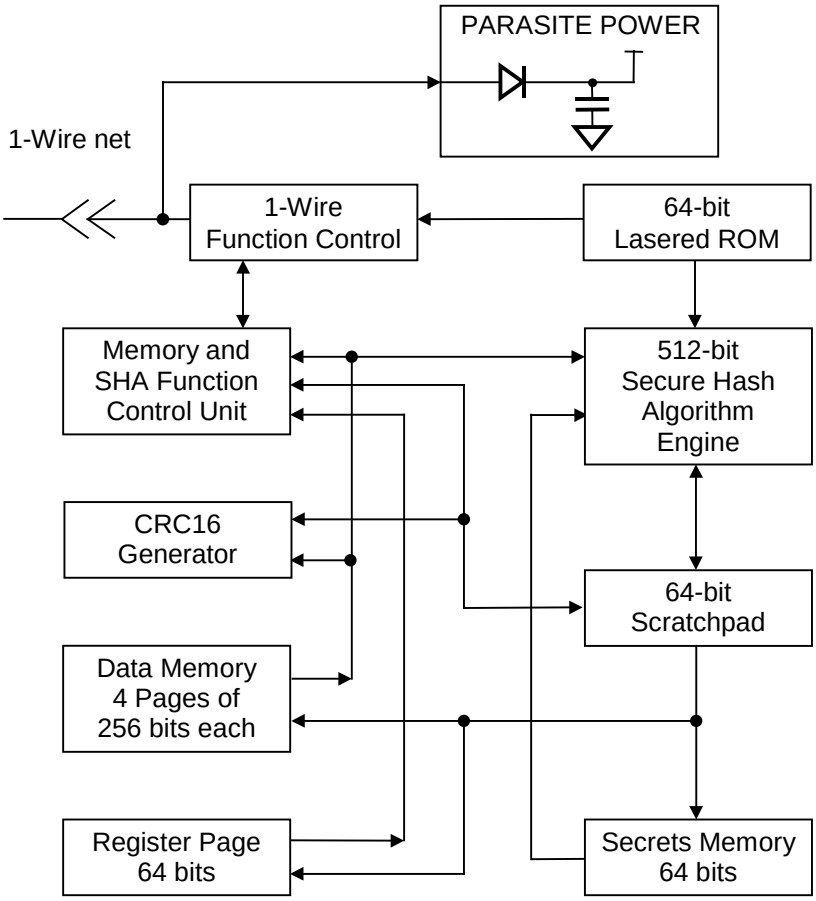
简介

DS2432 在单个芯片内集成了 1024 位 EEPROM、64 位密钥、一个 8 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 DS2432 具有自身的、由工厂刻入的 64 位 ROM 注册码，可确保唯一识别、绝对可溯。数据按照 1-Wire 协议串行传送，只需一根数据线和返回地线。DS2432 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 DS2432 接收到了匹配的 160 位 MAC，那么 Copy Scratchpad（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 DS2432 中(包含器件身份寄存器)的密钥和附加数据。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。DS2432 的典型应用包括：知识产权安全性检测、消费品的售后管理和数据装载器认证等。

概述

图 1 中的框图说明了 DS2432 的主控部分和存储单元之间的关系。DS2432 有五个主要的数据部件：1）64 位光刻 ROM，2）64 位暂存器，3）四个 32 字节的 EEPROM 页，4）64 位寄存器页，5）64 位密钥存储器，6）一个 512 位 SHA-1（安全散列算法）引擎。1-Wire 协议分层结构见图 2。总线主机必须首先提供七个 ROM 操作命令中的一个：1）Read ROM, 2） Match ROM, 3）Search ROM, 4）Skip ROM, 5）Resume Communication, 6）Overdrive Skip ROM 或 7）Overdrive Match ROM。一旦以标准速率完成 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，主机可以发出七条存储器和 SHA 操作命令中的任何一个。图 7 说明了有关这些存储器和 SHA 操作命令的协议。所有数据读写都是 LSB 在前。

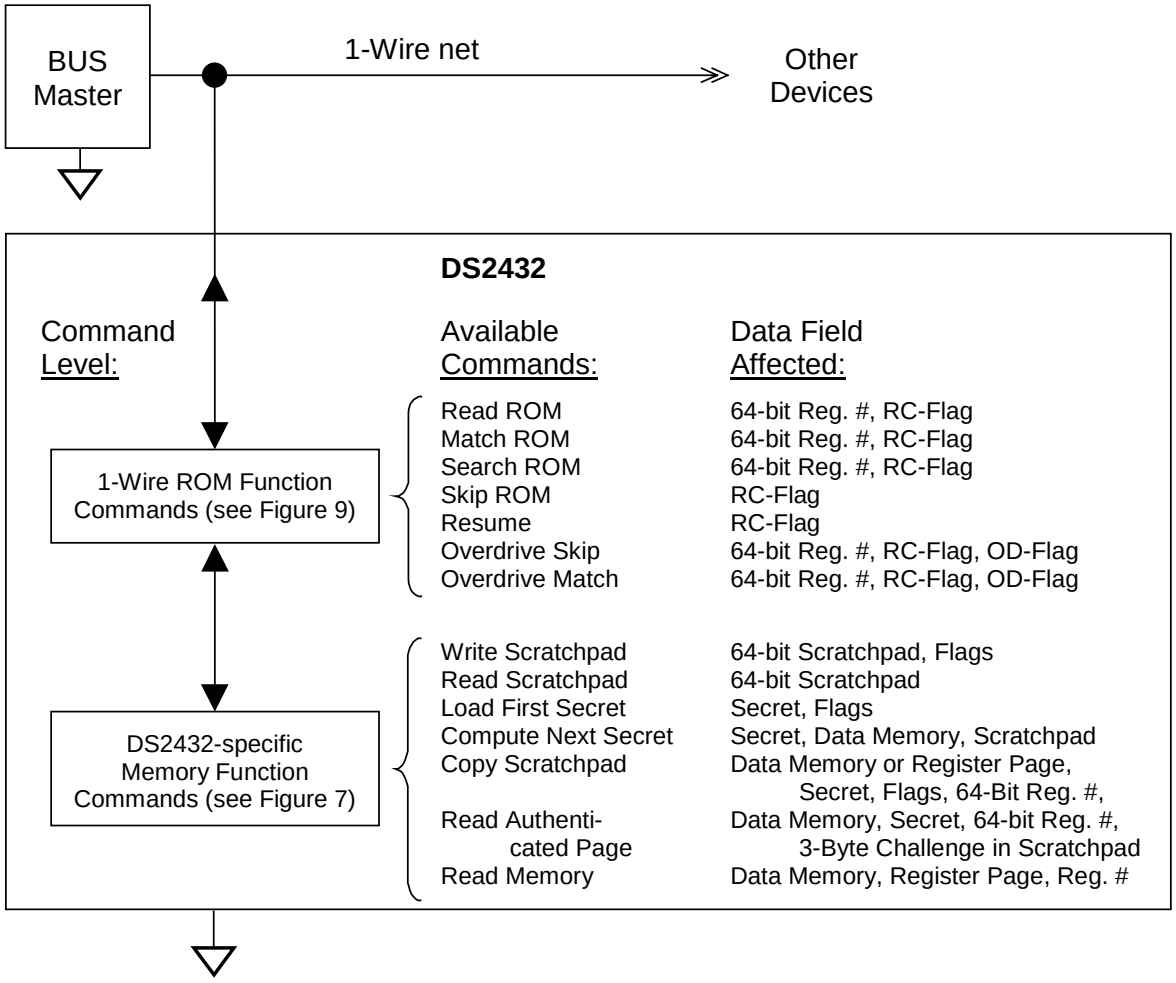
DS2432 原理框图 图 1



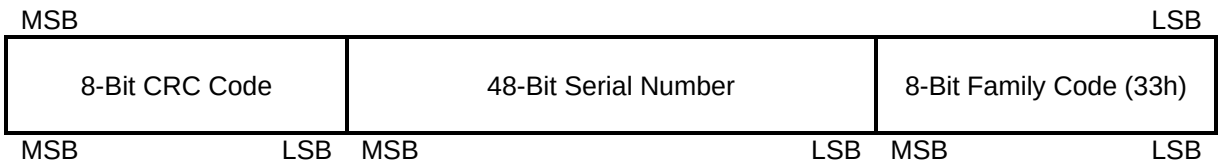
64 位光刻 ROM

每个DS2432 都有一个 64 位的唯一ROM代码。前 8 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 CRC”的更多信息参见Dallas Semiconductor的“Book of DS19xx iButton Standards”。移位寄存器初值为零。然后，从家族代码的LSB开始，每次移入一位。当家族代码第 8 位移入后，再移入序列号。当序列号第 48 位也移入后，留在移位寄存器中的就是CRC值。移入八位CRC校验码后，移位寄存器应该全部归零。

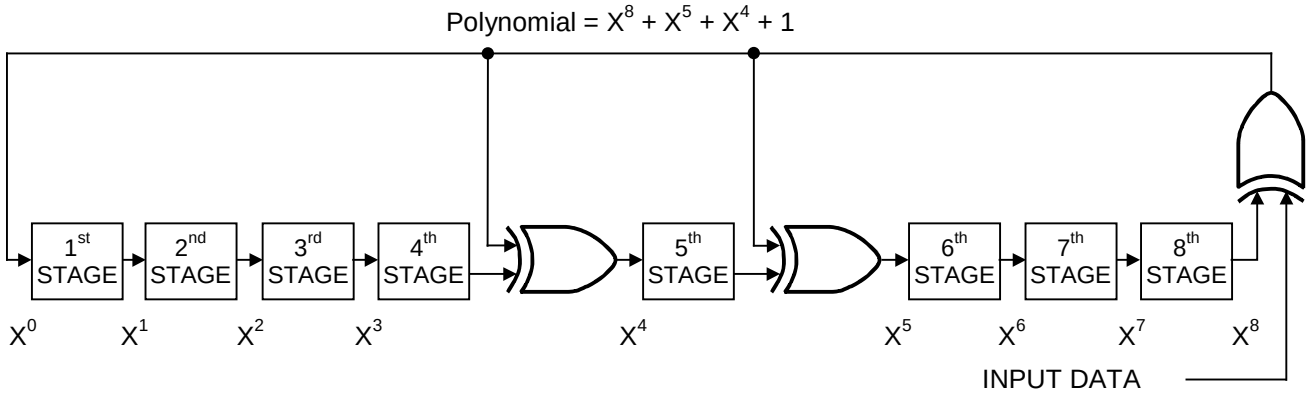
1-Wire 协议的层次结构 图 2



64 位光刻 ROM 图 3



1-Wire CRC 发生器 图 4



存储器图

DS2432 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。

如图 5 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

DS2432 存储器图 图 5

Address Range	Description	Note
0000h to 001Fh	Data Memory Page 0	No write-access without secret
0020h to 003Fh	Data Memory Page 1	No write-access without secret
0040h to 005Fh	Data Memory Page 2	No write-access without secret
0060h to 007Fh	Data Memory Page 3	No write-access without secret
0080h to 0087h	Secrets Memory	No read access; no secret for write access
0088h [1)]	Write-protect secret, 008Ch to 008Fh	Protection activated by code AAh or 55h
0089h [1)]	Write-protect pages 0 to 3	Protection activated by code AAh or 55h
008Ah [1)]	User byte, self-protecting	Protection activated by code AAh or 55h
008Bh	Factory byte (read only)	Reads either AAh or 55h; see text
008Ch [1)]	User byte/EPROM mode control for page 1	Mode activated by code AAh or 55h
008Dh [1)]	User byte/Write-protect page 0 **only**	Protection activated by code AAh or 55h
008Eh to 008Fh	User Bytes/Manufacturer ID	Function depends on factory byte
0090h to 0097h	64-Bit Registration Number	(Alternate readout)

[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 0088h 至 008Fh，也被称为寄存器页，含有特定功能寄存器，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。其它所有代码既不能写保护，也不能激活与这个特定字节相关的特殊功能。特殊功能为：1）仅写保护密钥，2）同时写保护四个数据存储器页，3）仅激活数据存储器页 1 的 EPROM 模式，4）仅激活数据存储器页 0 的 EPROM 模式。一旦 EPROM 模式被激活，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。

工厂字节读取结果为 55h 或 AAh。通常这个地址读取到的是 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 DS2432 相关的产品，以及快速找到可用的密钥。设置和注册制造商 ID 请与工厂联系。

地址 0090h 至 0097h 被称为身份寄存器，通常身份寄存器存有该器件 ROM 注册号的一个拷贝，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）读取数据时，总线主机接收到的注册号每一位顺序都与使用 ROM 功能命令相同。

地址寄存器 图 6

Bit #	7	6	5	4	3	2	1	0
Target Address (TA1)	T7	T6	T5	T4	T3	T2 (0)	T1 (0)	T0 (0)
Target Address (TA2)	T15	T14	T13	T12	T11	T10	T9	T8
Ending Address with Data Status (E/S) (Read Only)	AA	1	PF	1	1	E2 (1)	E1 (1)	E0 (1)

地址寄存器和传输状态

DS2432 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 DS2432 中的工作略有不同。寄存器 TA1 和 TA2 装载写入数据的目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 DS2432 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 则意味着主机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，主机可以在写命令

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。

带验证的写操作

为了向 DS2432 写入数据，需要把暂存器用作中间存储器。首先，主机发 Write Scratchpad（写暂存器）命令并指定目的地址和要写入暂存器的数据。需要注意的是，数据必须按 8 字节边界写入存储器内，目的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，器件将把这些位强制置为零，命令序列结束后写入修改后的地址。此外，执行命令时暂存器内的所有 8 个字节将拷贝到存储器，因此，应该向暂存器写入八个字节的数据，以保证所拷贝的数据是已知的。在一定条件下（参考 Write Scratchpad），Write Scratchpad 命令序列结束时，主机将接收命令、地址（实际发送地址）和数据取反后的 CRC16 校验码，该校验码计算时使用的地址和数据均为主机实际发送的值，而不是在非零 T2..T0 情况下的修正值。知道了 CRC 值，主机能够将接收到的 CRC 与自己计算的结果进行比较来判断通信是否成功，是否执行 Copy Scratchpad 命令。如果主机不能接收 CRC16，应该执行一次 Read Scratchpad 来验证写入数据的完整性。读暂存器时，作为暂存数据的导码，DS2432 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，则说明数据没有正确送达暂存器，或是上一次写暂存器后发生过掉电故障。主机不需要继续读操作，可以启动新一轮写暂存器操作。同样，授权许可（AA）标志置位、PF 标志清零，则说明器件未能正确识别写命令。如果每一过程都是正确的话，两个标志位将被清零。主机可以连续地读数据、验证数据字节。完成数据验证后，主机可以发 Copy Scratchpad 等命令。该命令必须跟随三个地址寄存器 TA1、TA2 和 E/S 的数据。主机应该通过读取暂存器获得这些寄存器的内容。

存储器和 SHA 命令

作为一个安全器件，DS2432 与其它 1-Wire 存储器使用稍有不同。DS2432 的大多数存储器可以像其它所有 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 功能流程图”描述了访问存储器和操作 SHA 引擎的协议。主机与 DS2432 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，DS2432 默认为标准速率。

Write Scratchpad [0Fh]

Write Scratchpad（写暂存器）适用于数据存储器、密钥和寄存器页中的可写地址。如果总线主机发送的目的地址大于 90h，将不执行该命令。

发出 Write Scratchpad 命令后，主机必须首先提供 2 个字节的目的地址，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，不论主机传送了多少个字节，结束偏移量（E2..E0）的值总是 111b。由于这个原因，主机应该总是发送 8 个字节的数据，尤其是载入的数据被用作密钥时。如果主机发送的数据少于 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机所不知道的随机数。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，DS2432 内部的 CRC 发生器（见图 12）随着主机的发送过程，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 DS2432 在实际的 Write

Scratchpad 命令中将设置 TA1 的 T2..T0 位为 000b，但是 CRC16 是根据主机发送的实际 TA1 进行计算的。主机可以随时终止 Write Scratchpad 命令。但是，如果暂存器已装满，主机可以再发 16 个读时隙接收由 DS2432 产生的 CRC 校验码。

如果试图以数据存储器(00h-7Fh)或寄存器页(88h 至 8Fh)为目标地址执行 Write Scratchpad 命令，那么，随后的 Read Scratchpad 命令对于写保护区域的地址读到的数据是 AAh 或 55h，而不是 Write Scratchpad 命令写入的数据。同样，如果目标地址位于第 1 页或是在 EPROM 模式下的页，从暂存器读回的数据将是最初暂存器的数据与当前目标存储区域内容的逻辑“与”。

Read Scratchpad [AAh]

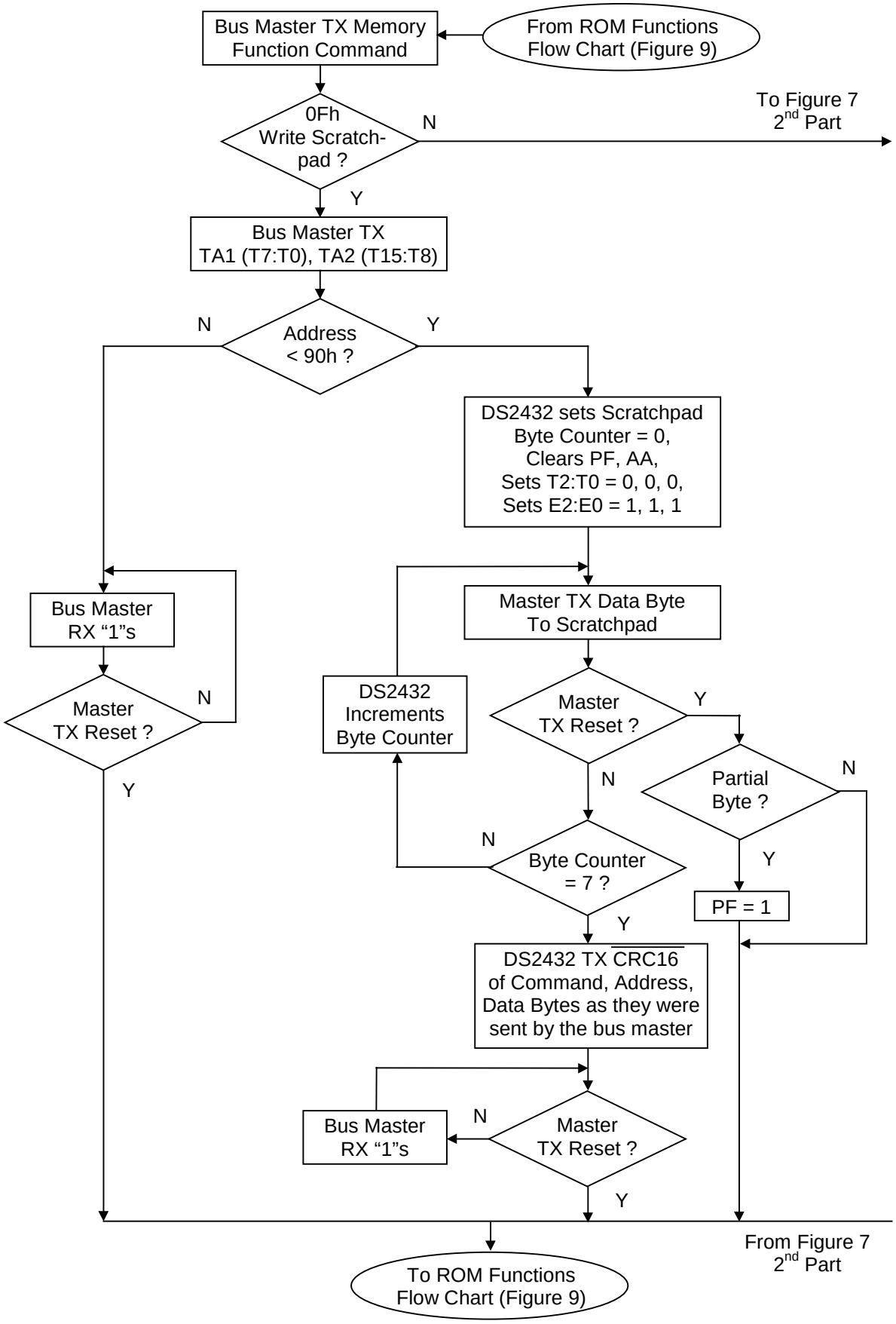
Read Scratchpad（读暂存器）可以用来验证目的地址和暂存器数据的完整性。发出命令码后，主机开始读数据。开头的两个字节是目的地址，其中 T2 至 T0 = 0。下一个字节是结束偏移量/数据状态字节（E/S），跟在后面的便是暂存器数据，它可能与主机最初发送的数据不同，尤其是当目的地址为密钥存储器、寄存器页、存储器页 1（处于 EPROM 模式）时，或使用 Refresh Scratchpad 时。这些情况下，暂存器数据有可能与 Write Scratchpad 或 Refresh Scratchpad 命令实际发送的数据不同。主机应该读到暂存器的最后一个字节，随后，就可以收到反码的 CRC。它基于 DS2432 所发送的数据计算产生。如果主机读取 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]

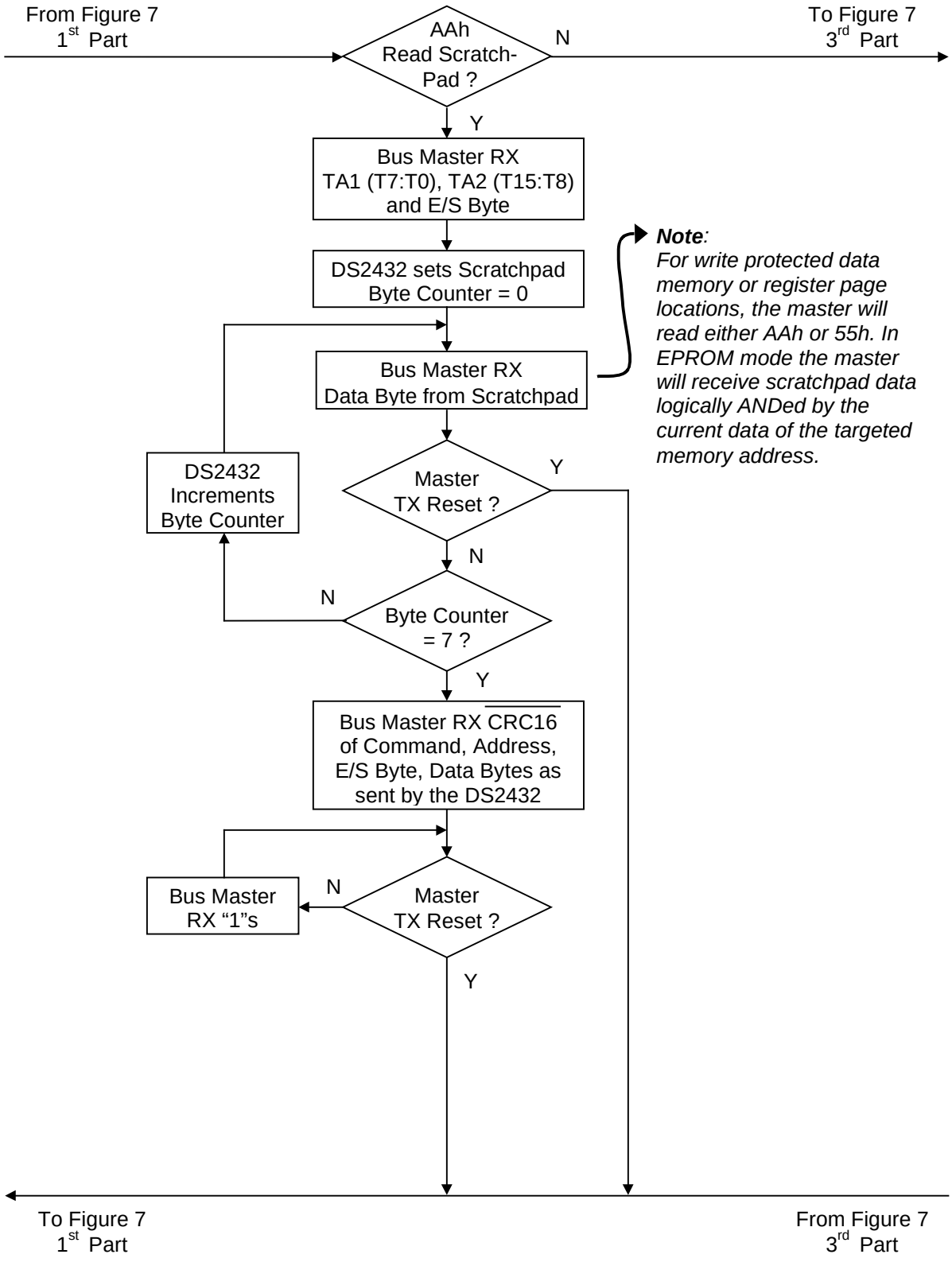
如果密钥未加写保护，Load First Secret（首次装载密钥）命令用暂存器数据替换器件现有的密钥。该命令不需要了解器件的当前密钥。在执行 Load First Secret 命令前，主机必须使用密钥起始地址（0080h）将新的密钥写入暂存器。发出 Load First Secret 后，主机必须提供一个 3 字节的授权码，此数据应该通过紧邻此条命令的前一个 Read Scratchpad 命令获得。这 3 个字节的数据必须与三个地址寄存器中（依次为 TA1，TA2，E/S）的数据完全匹配。如果数据匹配，而且密钥未加写保护，AA（接受授权）标志将置位，并开始复制数据。暂存器内容的所有 8 个字节的数据都将被复制到密钥存储单元。器件内部的数据传输时间最多持续 10ms，1-Wire 总线的电压不得低于 2.8V。数据拷贝之后，在主机发送复位脉冲之前，总线上将传送交替的“1”和“0”码。

除了使用 Load First Secret 命令外，也可以采用 Copy Scratchpad 命令装载新的密钥。但是，这种方式需要事先知道当前的密钥，并计算 160 位的 MAC。

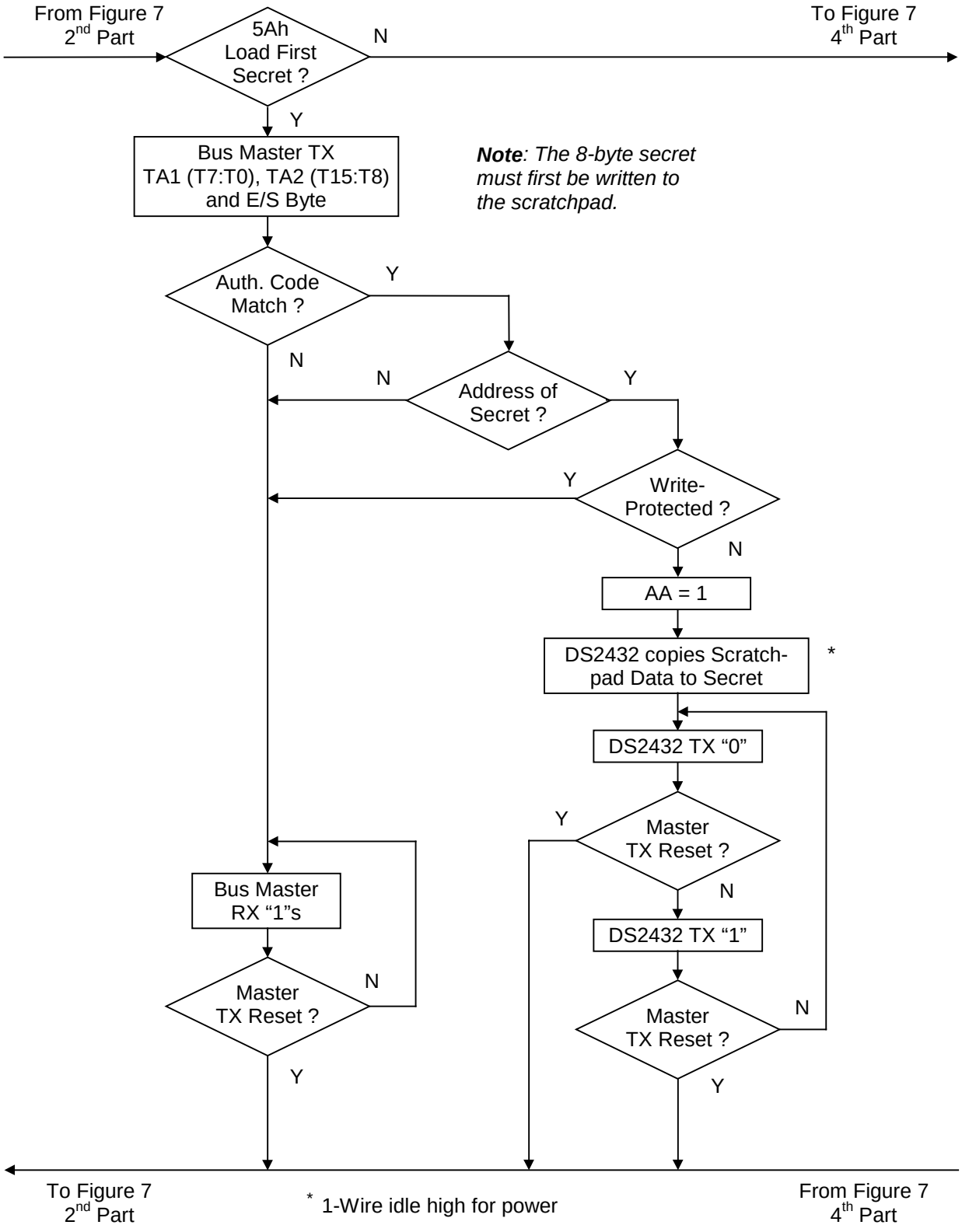
存储器和 **SHA** 功能流程图 图 7



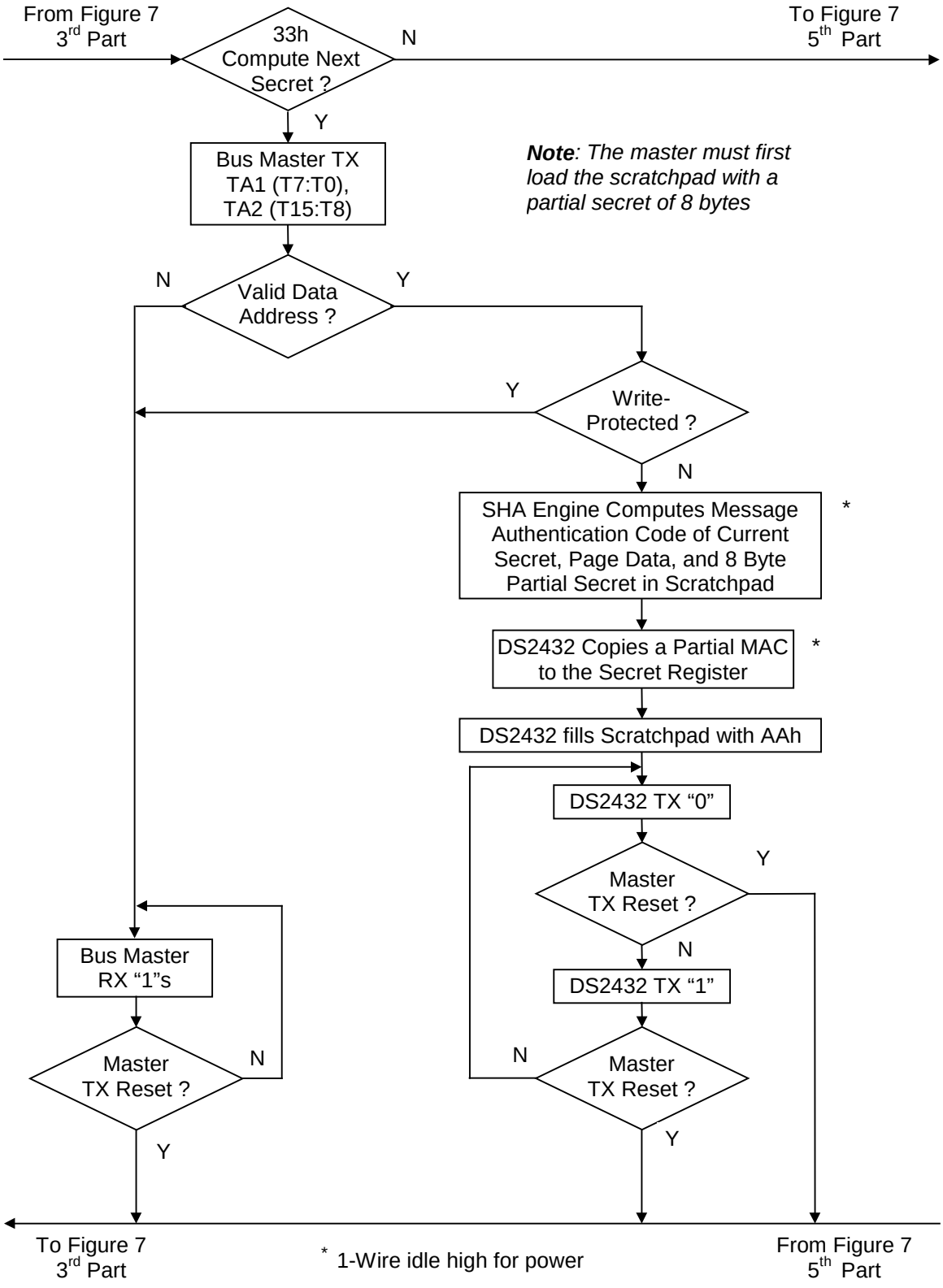
存储器和 **SHA** 功能流程图 (续) 图 7



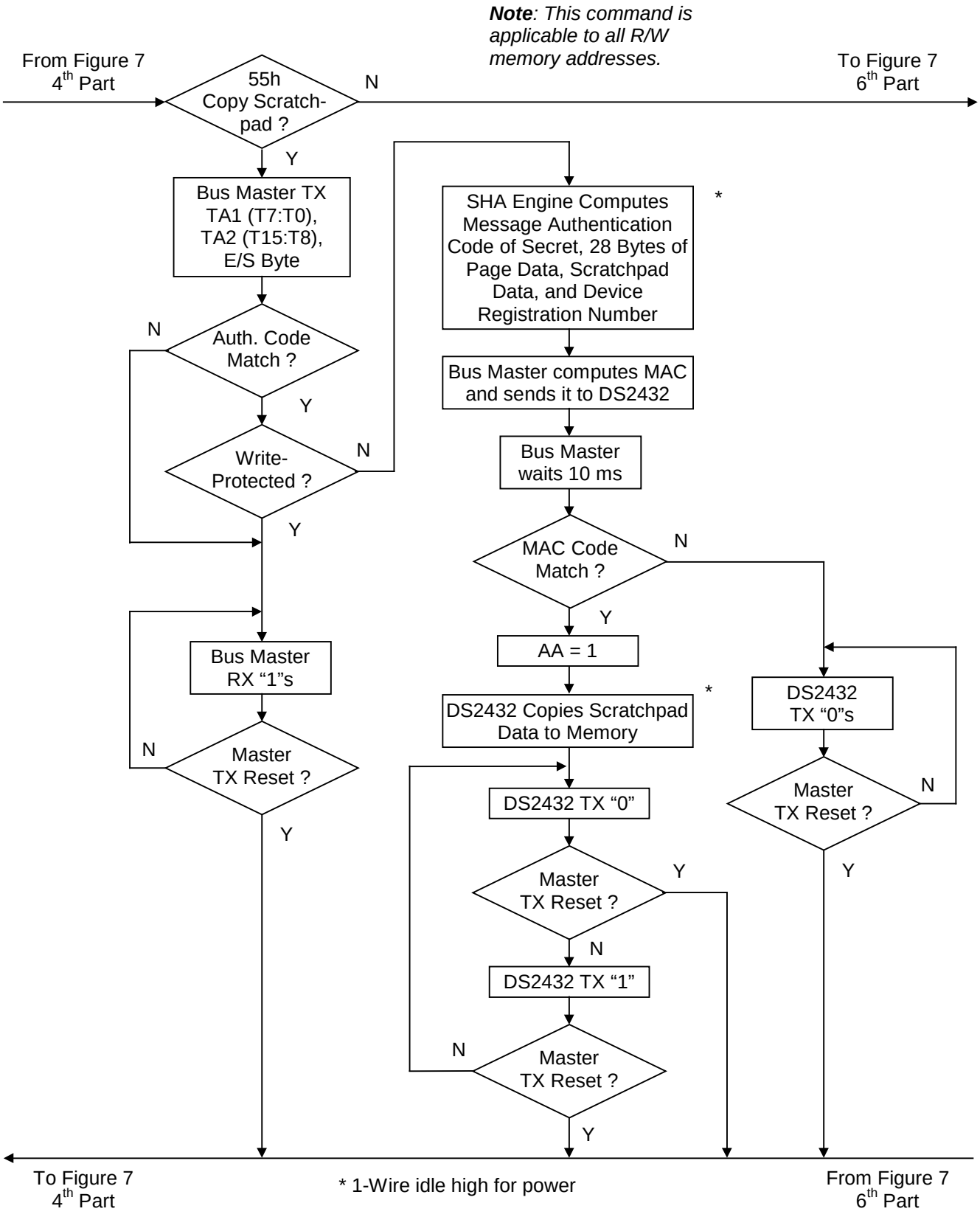
存储器和 **SHA** 功能流程图 (续) 图 7



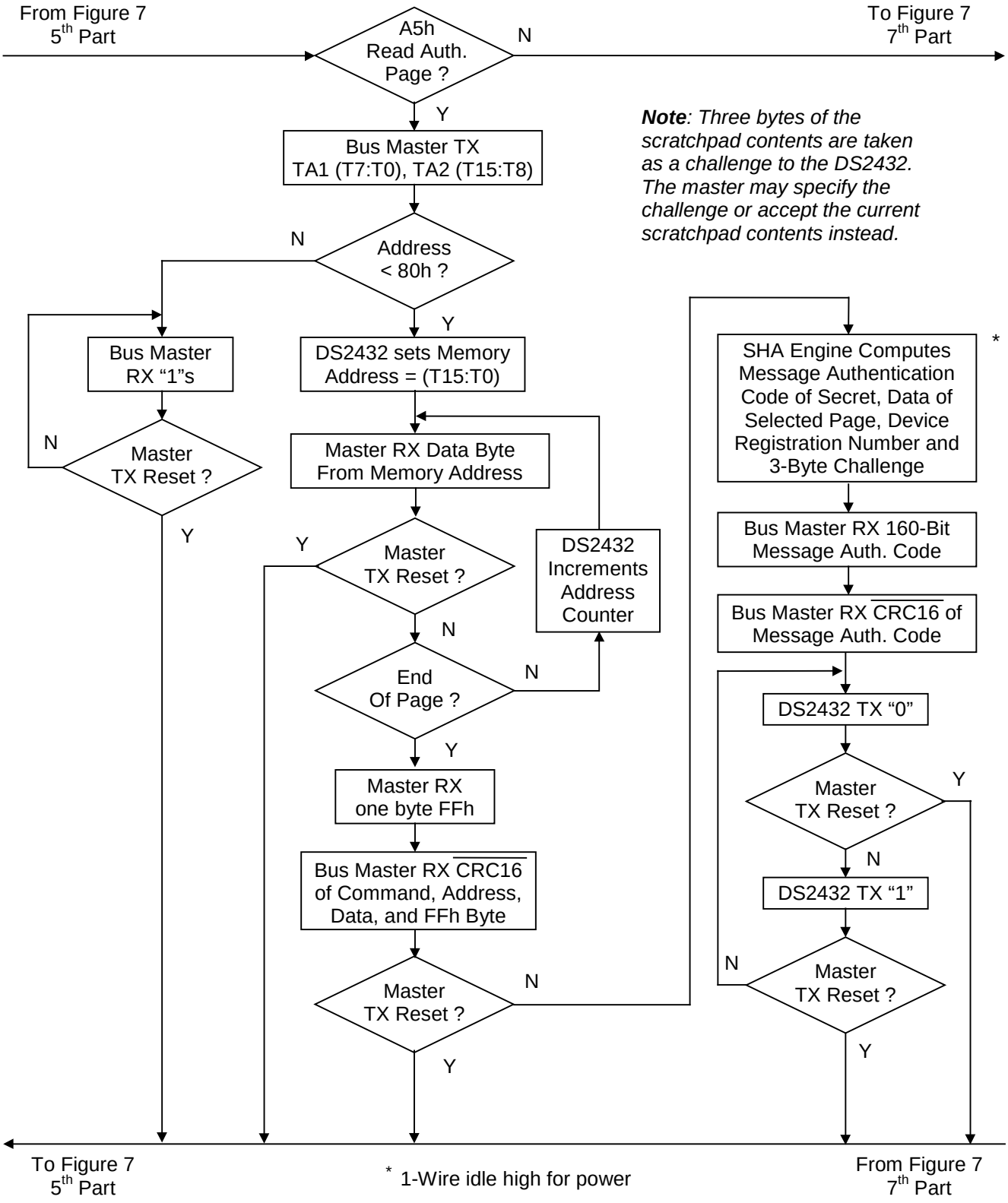
存储器和 SHA 功能流程图 (续) 图 7



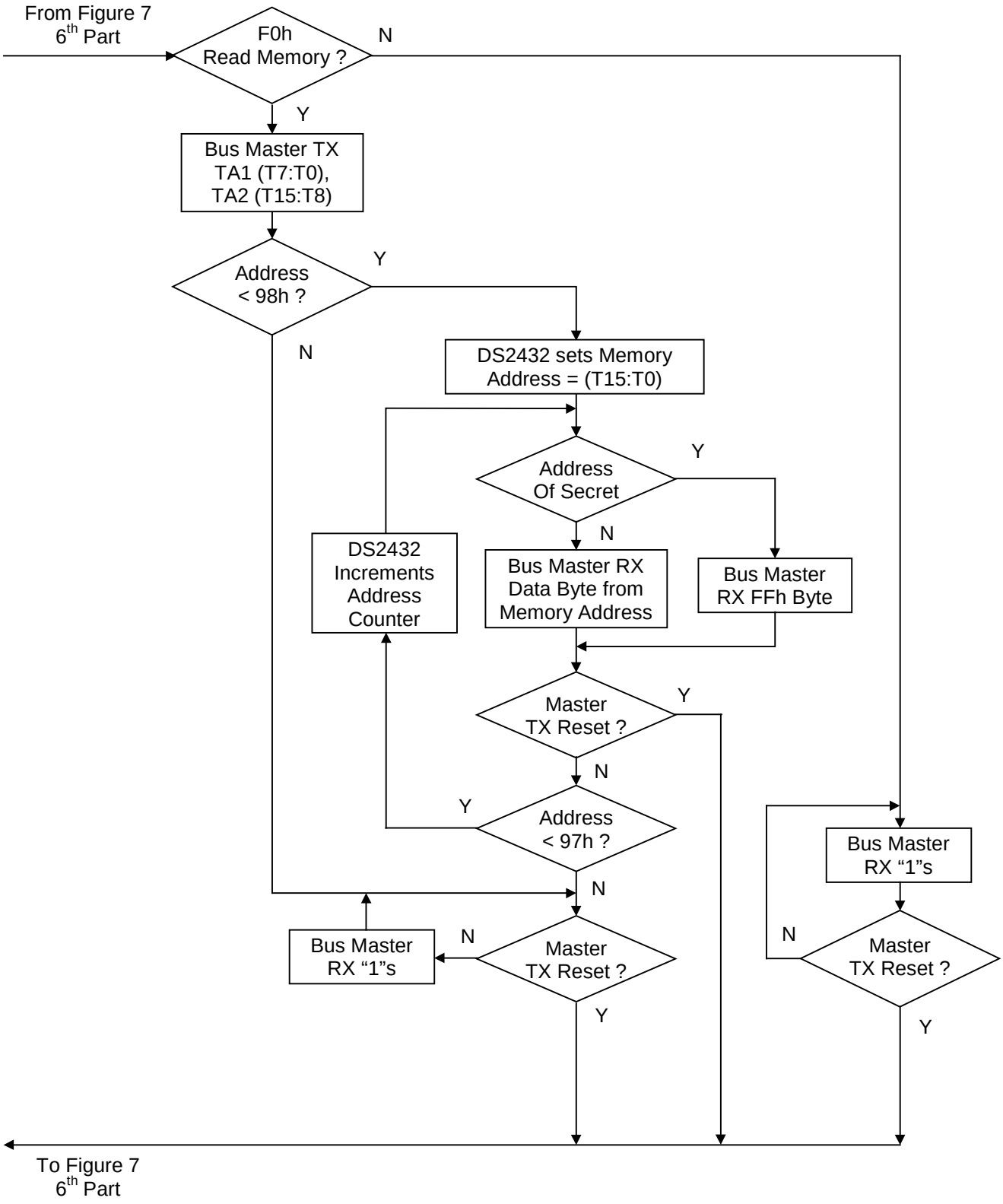
存储器和 **SHA** 功能流程图 (续) 图 7



存储器和 **SHA** 功能流程图 (续) 图 7



存储器和 **SHA** 功能流程图 (续) 图 7



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，DS2432 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，主机需要发 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。总线主机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。

用于 Compute Next Secret 命令的 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 的结果

(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]

Legend

Mt	**Input buffer of SHA engine** 0 ≤ t ≤ 15; 32-bit words
SS	**Starting address of secret (80h)**
PP	**Starting address of memory page** See Memory Map, memory pages 0 through 3
(SP+n)	**Byte n of scratchpad**
MPX	MPX[7] = 0; MPX[6] = 0; MPX[5:0] = (SP+0)[5:0]
D, E	32-bit words, portions of the 160-bit SHA result

发出 Compute Next Secret 命令后，主机必须提供一个 2 字节的目的地址，用于指定提供 256 位 SHA 输入数据的存储器页。目的地址 TA1 的低五位被忽略。如果目的地址有效，如在 0000h 至 007Fh 范围内，而且密钥未加写保护，SHA 引擎将启动，在 2ms 时间内计算出一个新的密钥，然

后将其自动复制到密钥寄存器中。替换密钥需要 10ms，在此期间和计算密钥时，1-Wire 总线上的电平不能低于 2.8V。复制完成后，DS2432 用 AAh 字节填充暂存器，在主机发送复位脉冲之前总线上将传输交替的“1”和“0”码。

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 8 字节已知数据。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]

DS2432 的数据存储器可以随意读取。然而，执行Copy Scratchpad（复制暂存器）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。主机可以在软件中计算MAC，或者用 DS1963S作为协处理器。协处理器的好处是密钥可以隐藏在协处理器iButton中。向DS2432 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。

信息认证码传送顺序 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]

Shift Direction

传送始于寄存器 E，最低有效位优先。

发出 Copy Scratchpad 后，主机必须提供一个 3 字节的授权码，此数据应该通过紧邻此条命令的前一个 Read Scratchpad 命令获得。这 3 个字节的数据必须与三个地址寄存器中（依次为 TA1、TA2、E/S）的数据完全匹配。如果授权码匹配，而且目的存储器未加写保护，DS2432 将启动 SHA 引擎，基于当前密钥、暂存器内所有数据、所寻址的存储器页的前 28 个字节和 DS2432 的注册码（不包括 CRC 校验码）计算一个 160 位 MAC。同时，主机也利用同样的数据计算一个 MAC，并把它发送给 DS2432，以便证明它有权写 EEPROM。然后，主机需要等待 10ms，在此期间，1-Wire 总线上的电平一定不能低于 2.8V。如果 DS2432 生成的 MAC 与主机计算的 MAC 相匹配，DS2432 将置位 AA 标志（接受授权），并将整个暂存器的内容拷贝到数据 EEPROM。成功复制后，在主机发送复位脉冲之前应该能够读到交替的“1”和“0”码。如果独到的数据全部是“0”，则说明没有发生数据复制。

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。在向寄存器页写数据（或通过 Copy Scratchpad 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）、身份寄存器的全部内容（M5，M6）和 4 个字节 FFh（M7）。

用于 Copy Scratchpad 命令的 SHA-1 输入数据 表 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

Legend

Mt	**Input buffer of SHA engine** $0 \le t \le 15$; 32-bit words
SS	**Starting address of secret (80h)**
PP	**Starting address of memory page** See Memory Map, memory pages 0 through 3
(SP+n)	**Byte n of scratchpad**
MP	MP[7:4] = 0000 for Copy Scratchpad MP[3:0] = T8:T5 (equivalent to page number in hex)
FAMC	**Family Code = 33h**
SNx	**Serial number of device** SN0 = least significant byte, SN5 = most significant byte. The CRC is not used

Read Authenticated Page [A5h]

利用命令 Read Authenticated Page（读验证页），主机可以获得全部或部分存储器页的数据和一个 MAC。利用 MAC，主机能够判定存储在 DS2432 中的密钥是否对于某个特定应用有效。DS2432 根据自己的密钥、指定存储器页的所有数据、自身的注册码和一个 3 字节的质询来计算 MAC，这个 3 字节质询是由主机在发 Read Authenticated Page 命令之前提前写入暂存器的。为此，主机可以使用 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。有关的质询部分为第 5、第 6 和第 7 个字节。作为另外一种选择，主机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，只是数据流向改为从 DS2432 至主机。执行 Read Authenticated Page 命令时输入到 SHA 引擎的数据见表 4。

主机发出命令代码并指定了目的地址（TA1 和 TA2）后，它将接收从目的地址开始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 码由命令代码、目的地址、已传送的数据和 FFh 字节产生。CRC 校验码接收完毕后，主机等待 2.0ms，在此期间，1-Wire 总线上的电平不能低于 2.8V。这段时间内，DS2432 的 SHA 引擎利用密钥、选定页的 32 个数据字节、器件的注册码（不包括 CRC 校验码）和 3 字节质询计算 MAC。然后，主机就可读取 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 校验码后主机继续读取数据，在它发送复位脉冲遣将收到交替的“1”和“0”码。

用于 **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

Legend

Mt	**Input buffer of SHA engine** $0 \le t \le 15$; 32-bit words
SS	**Starting address of secret (80h)**
PP	**Starting address of memory page** See Memory Map, memory pages 0 through 3
FAMC	**Family Code = 33h**
MP	MP[7:4] = 0100 MP[3:0] = T8:T5 (equivalent to page number in hex)
SNx	**ROM Serial number of device** SN0 = least significant byte, SN5 = most significant byte The CRC is not used
(SP+n)	**Byte n of Scratchpad**

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。尝试读取密钥时将得到不相关的数据。主机发出命令后，必须提供 2 个字节的目的地址。在这两个字节之后，主机从目的地址开始读取数据，可以一直读到地址 0097h。之后，如果主机继续读数，结果将全是逻辑 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。

DS2432 的硬件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机计算的、针对每页数据的 16 位 CRC 校验码。这样，主机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）。

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）。该算法采用十六个 32 位字 Mt（$0 \le t \le 15$）作为输入数据，如表 1，3A 和 3B 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 Wt（$0 \le t \le 79$）的八十个 32 位字的序列，一个称为 Kt（$0 \le t \le 79$）的八十个 32 位字的序列，一个布尔函数 ft（B，C，D）（$0 \le t \le 79$），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“Sn(X)”表示将 X 向左循环移 n 位，X 是一个 32 位字。

函数 ft 定义如下：

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

序列 Wt （$0 \le t \le 79$）定义如下：

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

序列 Kt（$0 \le t \le 79$）定义如下：

$$
\begin{aligned}
K_t := {} & 5A827999h && (0 \le t \le 19)\\
& 6ED9EBA1h && (20 \le t \le 39)\\
& 8F1BBCDCh && (40 \le t \le 59)\\
& CA62C1D6h && (60 \le t \le 79)
\end{aligned}
$$

变量 A、B、C、D、E 初始化如下：

A := 67452301h
B := EFCDAB89h
C := 98BADCFEh
D := 10325476h
E := C3D2E1F0h

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）：

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

主机可以按照表 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，主机必须计算 MAC，并将其发送给 DS2432。在执行 Compute Next Secret 命令时 MAC 不会暴露。SHA 运算寄存器 D 和 E 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

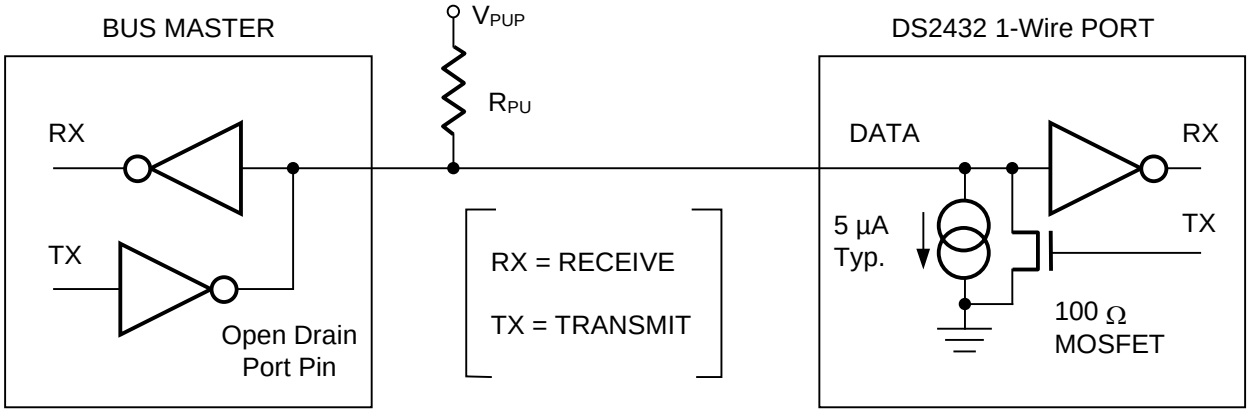
1-Wire总线系统由一个总线主机和一个或多个从器件组成。在所有应用实例中，DS2432 都作为从器件使用，总线主机通常是一个微控制器。对 1-Wire总线系统的讨论分为 3 个部分：硬件配置、处理流程和 1-Wire信令(信号类型和时序)。1-Wire协议根据特定时隙中总线的状态来工作，这些特定时隙始于总线主机发出的同步脉冲的下降沿。如需了解更多 1-Wire协议的详细描述，请参见“Book of DS19xx iButton Standards”第 4 章。

硬件配置

1-Wire总线只定义了一条数据线，所以，保证在适当的时间驱动总线上的每个器件非常重要。为了达到这一目的，接在 1-Wire总线上的每个器件都必须具有漏极开路或三态输出。DS2432 的 1-Wire端口为漏极开路，其内部等效电路如图 8 所示。多点总线由连接了多个从机器件的 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，DS2432 需要的 1-Wire上拉电阻最大值为 2.2kΩ。当与几个DS2432 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。如因某种原因需要暂停通信，稍候要恢复通信的话，总线必须保持在空闲状态。如果不是这样，当总线处于低电平状态超过 16µs（高速模式）或 120µs（常规速率）时，总线上的一个或多个器件将被复位。

硬件配置 图 8



处理流程

通过 1-Wire 端口访问 DS2432 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 DS2432 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 DS2432 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（见图 9 中的流程图）：

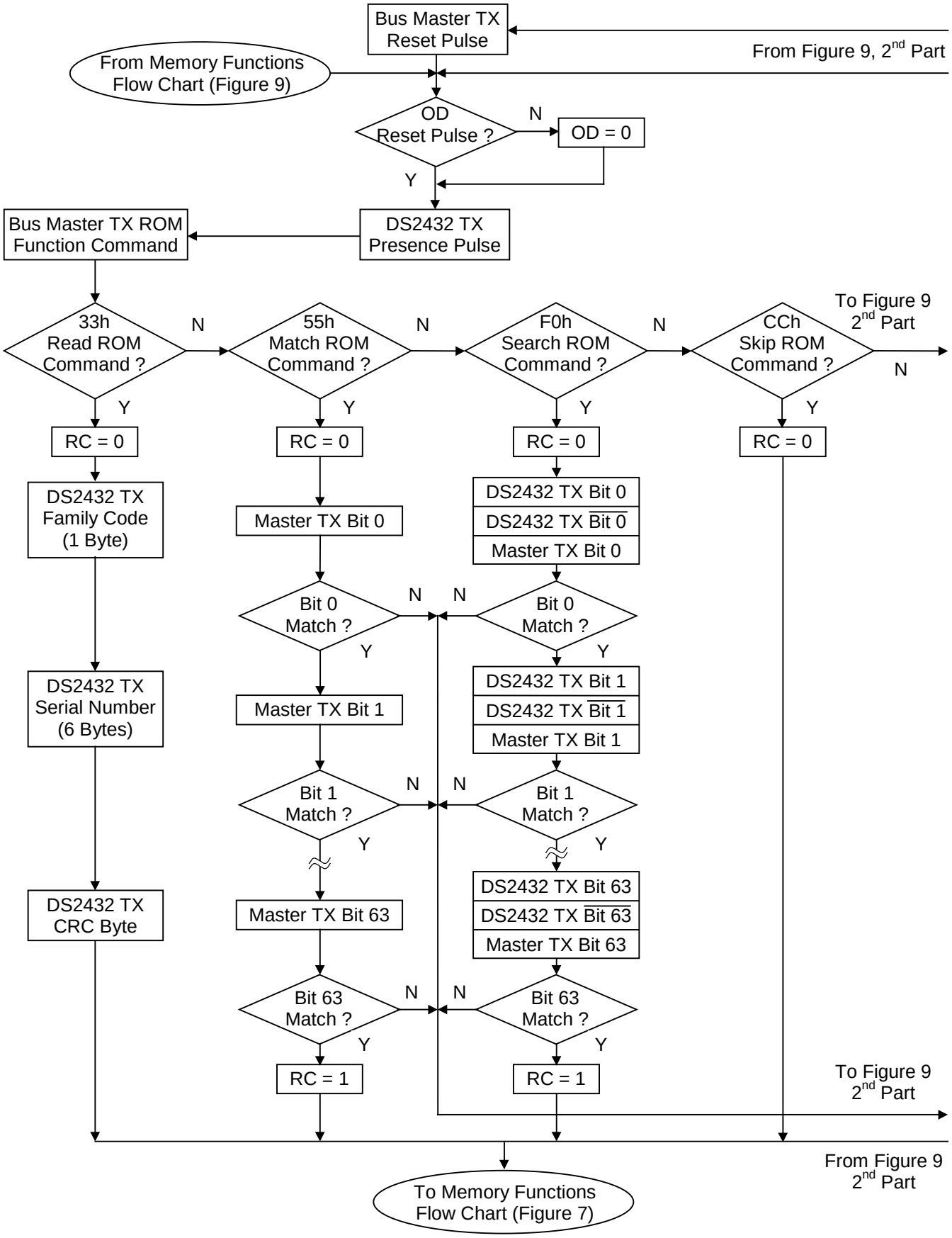
Read ROM [33h]

此条命令允许主机读取 DS2432 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。

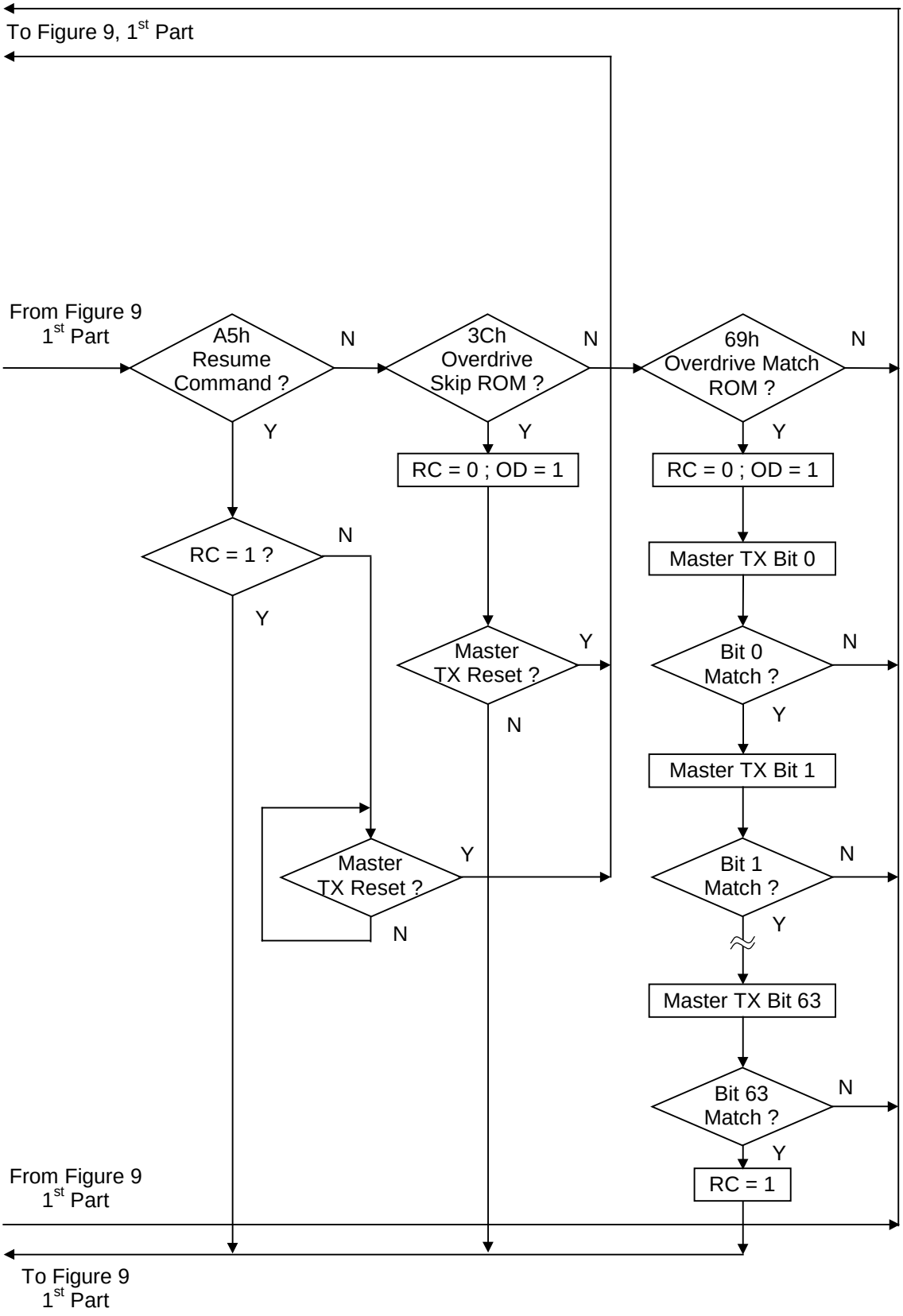
Match ROM [55h]

命令后面跟随 64 位注册号，允许主机访问多从机总线系统中某个特定的 DS2432。只有与 64 位注册号完全匹配的 DS2432 才会响应主机随后发出的存储器功能命令。所有其它从机将等待复位脉冲。这条命令既适用于单从机系统，也适用于多从机系统。

ROM 功能流程图 图 9


Bus Master TX Reset Pulse
From Figure 9, 2nd Part
From Memory Functions Flow Chart (Figure 9)
OD Reset Pulse ?
N
OD = 0
Y
DS2432 TX Presence Pulse
Bus Master TX ROM Function Command
33h Read ROM Command ?
55h Match ROM Command ?
F0h Search ROM Command ?
CCh Skip ROM Command ?
To Figure 9 2nd Part
N
Y
RC = 0
DS2432 TX Family Code (1 Byte)
DS2432 TX Serial Number (6 Bytes)
DS2432 TX CRC Byte
Master TX Bit 0
Master TX Bit 1
Master TX Bit 63
DS2432 TX Bit 0
DS2432 TX Bit 0
Master TX Bit 0
DS2432 TX Bit 1
DS2432 TX Bit 1
Master TX Bit 1
DS2432 TX Bit 63
DS2432 TX Bit 63
Master TX Bit 63
Bit 0 Match ?
Bit 1 Match ?
Bit 63 Match ?
RC = 1
To Figure 9 2nd Part
From Figure 9 2nd Part
To Memory Functions Flow Chart (Figure 7)

ROM 功能流程图 (续) 图 9



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 位注册号，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 位注册号。Search ROM过程其实只是简单的 3 步骤重复：读一位、读此位的补码，然后写这一位的期望值，主机对注册号的每一位数据都执行这简单的 3 步骤操作。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。其余从机设备的注册号可经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 5 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 位注册号，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，总线主机不需要 64 位的注册号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 DS2432 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。如果总线上有多于一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 位注册号，总线主机可以在多点总线上找到某个特定的 DS2432，并将它设置成高速模式。只有 64 位注册码精确匹配的 DS2432 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率。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]

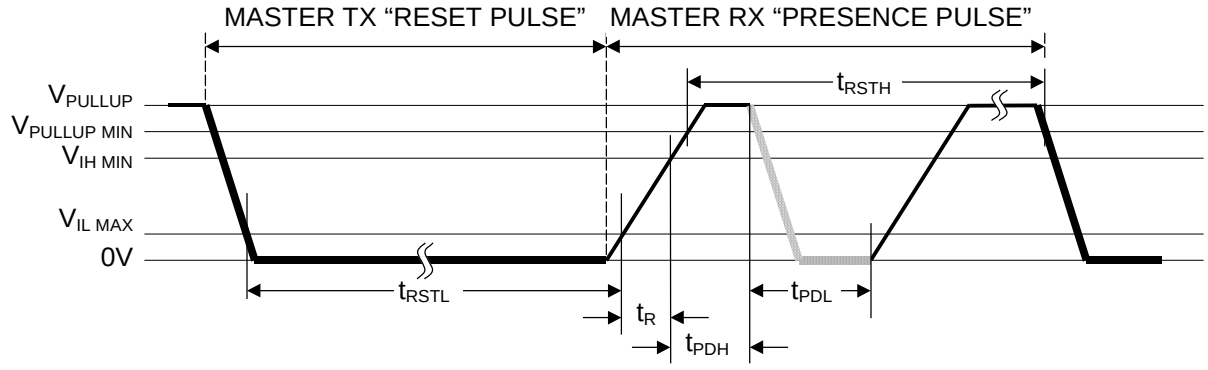
在一个典型应用中，要写满一个 32 字节的存储器页，往往需要多次访问 DS2432。这意味着在多点环境中，每次访问都要重复执行 Match ROM 命令和发送 64 位注册号。为了提高多点环境中的数据吞吐率，设置了 Resume Command 功能。该功能检测 RC 位的状态，如果置位，就直接传递控制给存储器和 SHA 功能，类似于 Skip ROM 命令。设置 RC 位的唯一方法是成功地执行 Match ROM，Search ROM 或 Overdrive Match ROM 命令。一旦设置了 RC 位，利用 Resume Command 功能就可重复访问同一器件。对于总线上另一器件的访问将清除 RC 位，以防两个或更多的器件同时响应 Resume Command 功能。

1- Wire 信令

为了保证数据的完整性，DS2432 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：包括复位脉冲和在线应答脉冲的复位序列、写 0、写 1 和读数据。除了在线应答脉冲以外，所有其它信号均由总线主机发出。DS2432 能够以两种不同速率通信：标准速率和高速模式。如果没有明确设定为高速模式，DS2432 就以标准速率通信。高速模式下，所有波形均采用快速定时。

复位脉冲后面跟随一个在线应答脉冲表明DS2432 已经准备好发送或接收数据。总线主机发送（TX）一个复位脉冲（t_{RSTL}，标准速率下至少 480µs，高速模式下至少 48µs）。随后，主机将释放总线，进入接收模式 (RX)。这时 1-Wire总线通过上拉电阻被拉高。当DS2432 在数据引脚上检测到上升沿后，等待一段时间(t_{PDH}，标准速率下 15 至 60µs，高速模式下 2 至 6µs)，然后发送在线应答脉冲(t_{PDL}，标准速率下 60 至 240µs，高速模式下 8 至 24µs)。480µs或更长时间的复位脉冲将使器件退出高速模式，恢复到标准速率。如果DS2432 处于高速模式下，并且复位脉冲时间不高于 80µs，则器件保持高速模式。

初始化时序“复位脉冲和在线应答脉冲” 图 10



REGULAR SPEED	OVERDRIVE SPEED
480 µs ≤ t_{RSTL} < ∞*	48 µs ≤ t_{RSTL} < 80 µs
480 µs ≤ t_{RSTH} < ∞**	48 µs ≤ t_{RSTH} < ∞ **
15 ≤ t_{PDH} < 60 µs	15 ≤ t_{PDH} < 6 µs
60 µs ≤ t_{PDL} < 240	8 µs ≤ t_{PDL} < 24

* 为了避免 1-Wire总线上的其它器件屏蔽中断信号，t_{RSTL} + t_R应始终小于 960µs。

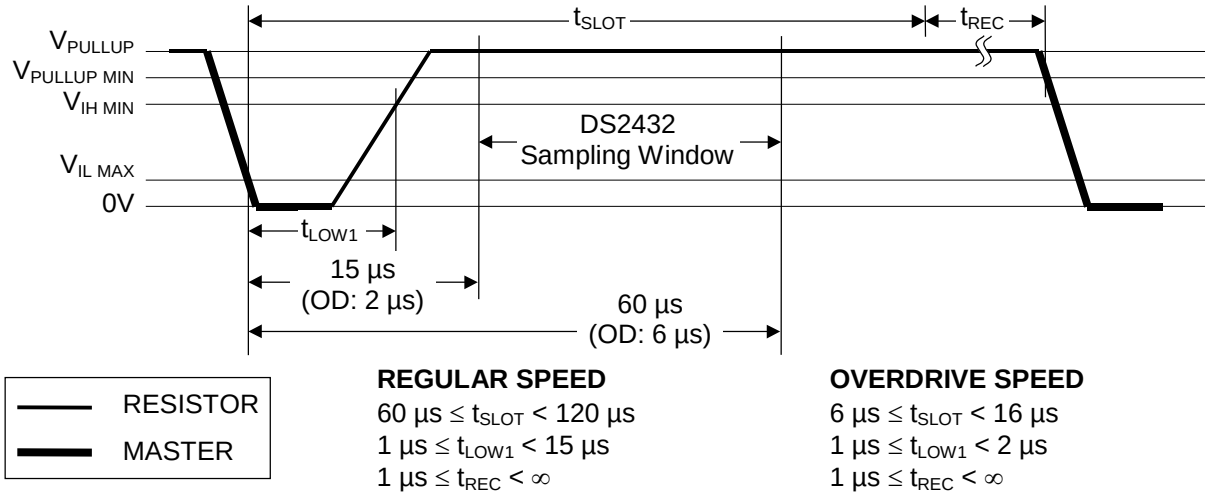
** 包括恢复时间。

读/写时隙

读、写时隙的定义如图 11 所示。主机通过拉低数据线来启动所有时隙。数据线的下降沿通过触发内部延迟电路使 DS2432 与主机同步。在写时隙中，延迟电路可确定什么时候 DS2432 采样数据线。对读数据时隙来说，如果发送的是“0”，那么延迟电路将决定 DS2432 数据线保持为低的时间。如果数据位是“1”，则 DS2432 无需将数据线拉低。

读/写时序图 图 11

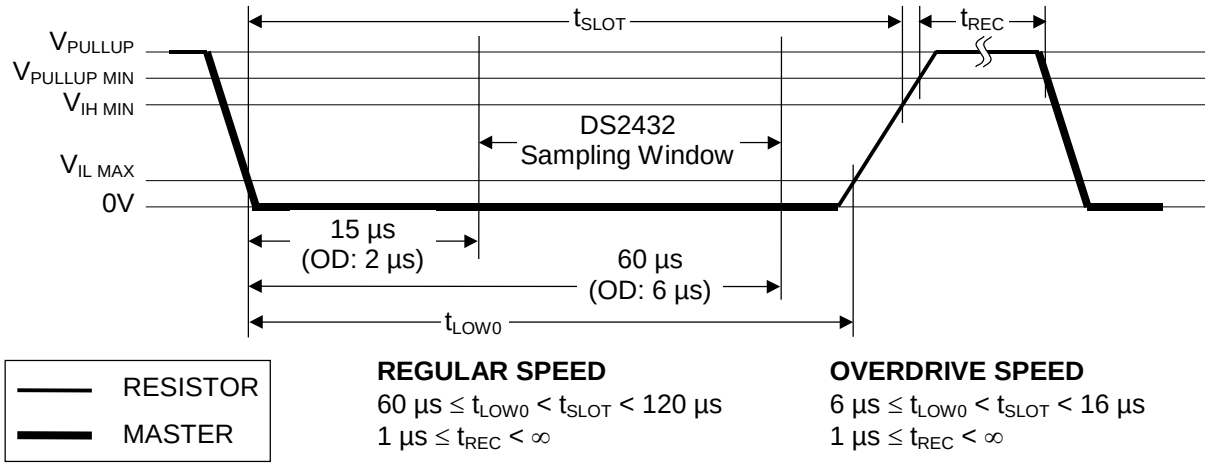
Write-one Time Slot



REGULAR SPEED
60 µs ≤ t_{SLOT} < 120 µs
1 µs ≤ t_{LOW1} < 15 µs
1 µs ≤ t_{REC} < ∞

OVERDRIVE SPEED
6 µs ≤ t_{SLOT} < 16 µs
1 µs ≤ t_{LOW1} < 2 µs
1 µs ≤ t_{REC} < ∞

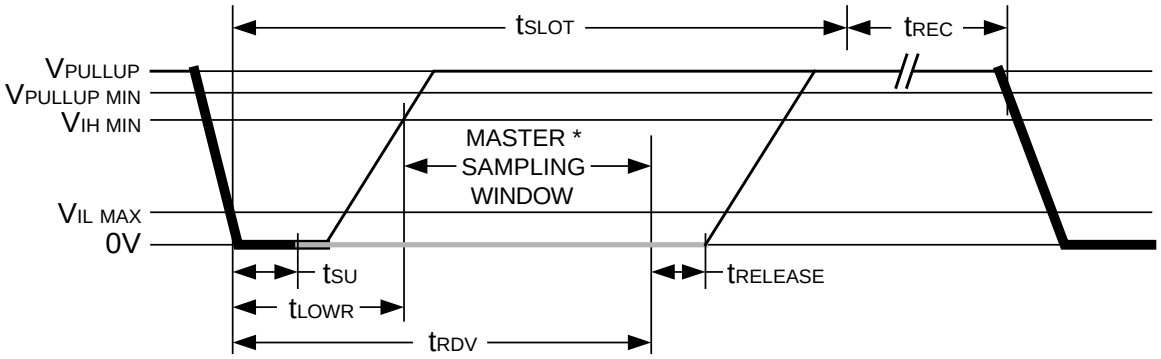
Write-zero Time Slot



REGULAR SPEED
60 µs ≤ t_{LOW0} < t_{SLOT} < 120 µs
1 µs ≤ t_{REC} < ∞

OVERDRIVE SPEED
6 µs ≤ t_{LOW0} < t_{SLOT} < 16 µs
1 µs ≤ t_{REC} < ∞

Read-data Time Slot



Waveform Legend:
- RESISTOR
- MASTER
- DS2432

	REGULAR SPEED	OVERDRIVE SPEED
	$60\mu s \le t_{SLOT} < 120\mu s$	$6\mu s \le t_{SLOT} < 16\mu s$
	$1\mu s \le t_{LOWR} < 15\mu s$	$1\mu s \le t_{LOWR} < 2\mu s$
	$0\mu s \le t_{RELEASE} < 45\mu s$	$0\mu s \le t_{RELEASE} < 4\mu s$
	$1\mu s \le t_{REC} < \infty$	$1\mu s \le t_{REC} < \infty$
	$t_{RDV} = 15\mu s$ *	$t_{RDV} = 2\mu s$ *
	$t_{SU} < 1\mu s$	$t_{SU} < 1\mu s$

* 主机的最佳采样点应尽可能靠近t_{RDV}，不要超出t_{RDV}。执行读 1 时隙时，这样做会给上拉电阻留出足够的时间以将总线恢复为高电平。执行读 0 时隙时，这将确保在最快的 1-Wire器件释放总线前 ($t_{RELEASE}$ = 0) 执行读操作。

CRC 生成

DS2432 有两种类型的循环冗余校验（CRC）。其中一种类型是 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，总线主机可用 64 位ROM的前 56 位计算CRC值，并将其与从 DS2432 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、写或更新暂存器的时候，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。DS2432 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。总线主机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，来决定是继续某一操作还是重读有CRC错误的数据部分。

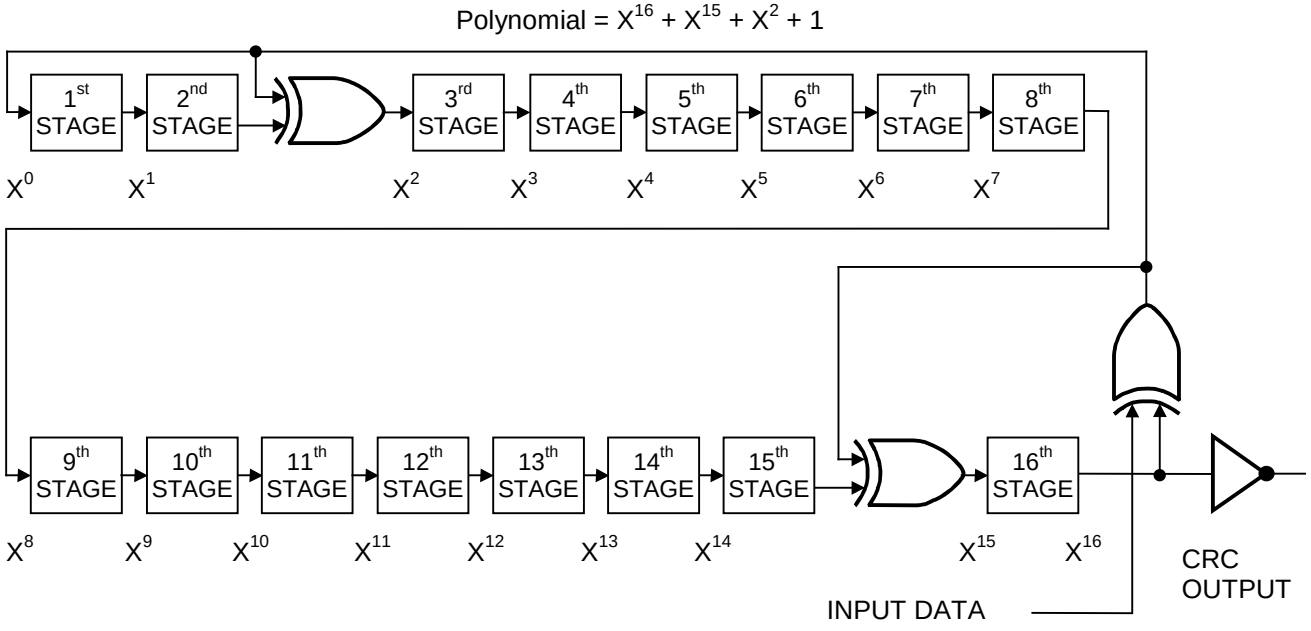
执行 Write Scratchpad 命令时，首先清除 CRC 发生器，然后移入命令代码，目的地址 TA1（T2 至 T0 均置 0）和 TA2，以及所有主机发送的数据字节。只有当暂存器完全写入时，DS2432 才发送该 CRC 校验码。

执行 Read Scratchpad 命令时，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 DS2432 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，DS2432 才发送该 CRC 校验码。

执行 Read Authenticated Page 命令时，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，按照主机接收的位序移入 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，请参考 “Book of DS19xx iButton Standards”.

CRC-16 硬件和多项式描述 图 12



ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +5.5V
Operating Temperature	-40°C to +85°C
Storage Temperature	-55°C to +125°C
Soldering Temperature	See J-STD-020A specification

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

DC ELECTRICAL CHARACTERISTICS (V_{PUP} =2.8V to 5.25V; -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
1-Wire Input High	V_{IH}	2.2			V	1, 7
1-Wire Input Low	V_{IL}	-0.3		TBD	V	1, 8
1-Wire Output Low @ 4 mA	V_{OL}			0.4	V	1
1-Wire Output High	V_{OH}		V_{PUP}		V	1, 2
Input Load Current	I_L		5		µA	3
Programming Current	I_{LPROG}		500		µA	9

CAPACITANCE (t_A = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
1-Wire I/O	$C_{IN/OUT}$		100	800	pF	5

ENDURANCE (V_{PUP} =5.0V; T_A = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Write/Erase Cycles	N_{CYCLE}	50k			—	
Data Retention	t_{DRET}	10			years	

AC ELECTRICAL CHARACTERISTICS
REGULAR SPEED (V_{PUP}=2.8V to 5.25V; -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Time Slot	t_{SLOT}	60		120	µs	
Write 1 Low Time	t_{LOW1}	1		15	µs	
Write 0 Low Time	t_{LOW0}	60		120	µs	
Read Low Time	t_{LOWR}	1		15	µs	
Read Data Valid	t_{RDV}		15		µs	10
Release Time	$t_{RELEASE}$	0	15	45	µs	
Read Data Setup	t_{SU}			1	µs	4
Recovery Time	t_{REC}	1			µs	
Reset High Time	t_{RSTH}	480			µs	
Reset Low Time	t_{RSTL}	480			µs	6
Presence Detect High	t_{PDHIGH}	15		60	µs	
Presence Detect Low	t_{PDLOW}	60		240	µs	
Programming Time	t_{PROG}			10	ms	
SHA Computation Time	t_{CSHA}		1.0	2.0	ms	9

AC ELECTRICAL CHARACTERISTICS
OVERDRIVE SPEED

(V_{PUP}=2.8V to 5.25V; -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Time Slot	t_{SLOT}	6		16	µs	
Write 1 Low Time	t_{LOW1}	1		2	µs	
Write 0 Low Time	t_{LOW0}	6		16	µs	
Read Low Time	t_{LOWR}	1		2	µs	
Read Data Valid	t_{RDV}		2		µs	10
Release Time	$t_{RELEASE}$	0	1.5	4	µs	
Read Data Setup	t_{SU}			1	µs	4
Recovery Time	t_{REC}	1			µs	
Reset High Time	t_{RSTH}	48			µs	
Reset Low Time	t_{RSTL}	48		80	µs	
Presence Detect High	t_{PDHIGH}	2		6	µs	
Presence Detect Low	t_{PDLOW}	8		24	µs	
Programming Time	t_{PROG}			10	ms	
SHA Computation Time	t_{CSHA}		1.0	2.0	ms	9

注释:

1. 所有电压参考地。
2. V_{PUP} = 外部上拉电压。
3. 输入负载以地为参考。
4. 读数据建立时间是指主机为读取数据而必须将 1-Wire 总线拉低的时间。在下降沿 1µs 后，数据应保证有效。
5. 首次加电时，数据引脚上产生的电容可能会达到 800pF。如果采用一个 5kΩ上拉电阻将数据线拉高至V_{PUP}，则上电 5µs之后该寄生电容就不会对正常通信产生影响了。
6. 复位低电平时间(t_{RSTL})的最大值应被限制在 960µs以内，这样中断信号可以工作；否则可能会掩盖或屏蔽中断脉冲
7. V_{IH} 是外部上拉电阻和V_{PUP} 的函数。
8. 在某些低电压情况下，V_{ILMAX}可能必须降至 0.5V，以保证有在线应答脉冲时可正常工作。V_{IL} 是V_{PUP} 和复位低电平时间的函数。
9. 在写 EEPROM 操作和计算 MAC 期间，1-Wire 总线电压不能低于 2.8V。计算 MAC 最多需要 2.0ms。将暂存器数据拷贝到 EEPROM，最多需要 10ms。
10. 主机的最佳采样点应尽可能靠近t_{RDV}，但不能超过t_{RDV}。执行读 1 时隙时，这样做会给上拉电阻留出足够的时间来使总线恢复为高电平；执行读 0 时隙时，这将确保在最快的 1-Wire器件释放总线前($t_{RELEASE}$ = 0)执行读操作。