DS1963S
SHA iButton

www.iButton.com

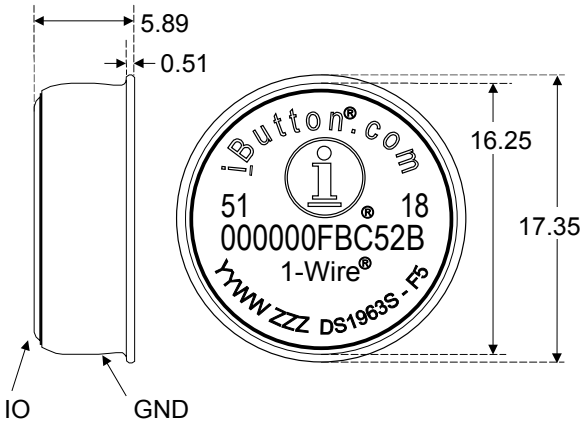
SPECIAL FEATURES

- 4096 bits of read/write nonvolatile (NV) memory organized as 16 pages of 256 bits each
- Eight memory pages with individual 64-bit secrets and 32-bit read-only non rolling-over page write cycle counters
- Secrets are write-only and have their own individual write cycle counters
- On-chip 512-bit SHA-1 (FIPS 180-1, ISO/IEC 10118-3) engine to compute a 160-bit Message Authentication Codes (MAC) and generate page secrets
- Device can operate as roaming iButton® or as coprocessor for a host computer
- 256–bit scratchpad ensures integrity of data transfer
- On-chip 16-bit CRC generator for safeguarding data transfers
- Overdrive mode boosts communication speed to 125 kbits per second
- Operating temperature range from -20°C to +85°C
- Over 10 years of data retention

COMMON iButton FEATURES

- Unique, factory–lasered and tested 64-bit registration number (8-bit family code + 48-bit serial number + 8-bit CRC tester) assures absolute traceability because no two parts are alike
- Multidrop controller for MicroLAN
- Digital identification and information by momentary contact
- Chip-based data carrier compactly stores information
- Data can be accessed while affixed to object
- Economically communicates to host with a single digital signal at 15.4 kbits per second
- Standard 16 mm diameter and 1-Wire® protocol ensure compatibility with iButton Device family
- Button shape is self-aligning with cup-shaped probes
- Durable stainless steel case engraved with registration number withstands harsh environments
- Easily affixed with self-stick adhesive backing, latched by its flange, or locked with a ring pressed onto its rim
- Presence detector acknowledges when reader first applies voltage
- Meets UL#913 (4th Edit.); Intrinsically Safe Apparatus, Approved under Entity Concept for use in Class I, Division 1, Group A, B, C and D Locations (application pending)

F5 MicroCan



All dimensions are shown in millimeters

ORDERING INFORMATION

DS1963S	F5 MicroCan

EXAMPLES OF ACCESSORIES

DS9096P	Self-Stick Adhesive Pad
DS9101	Multipurpose Clip
DS9093RA	Mounting Lock Ring
DS9093A	Snap-In Fob
DS9092	iButton Probe

iButton and 1-Wire are registered trademarks of Dallas Semiconductor.

iButton DESCRIPTION

The DS1963S Monetary iButton with SHA-1 Function is a rugged 4 kbit read/write data carrier that can be easily accessed with minimal hardware. Its NV memory acts as a localized database for public as well as protected data belonging to the owner of the device and the environment in which it is used. An integrated 512-bit SHA-1 engine can be activated to compute 160-bit message authentication codes (MAC) based on information stored in the device. Data is transferred serially via the 1-Wire protocol, which requires only a single data lead and a ground return. Using the TMEX file format (see Application Note 114) a single DS1963S can serve up to four independent applications, such as secure change purses for electronic payment at local transit systems, pay phones, parking systems or vending machines. The DS1963S is also intended to function as a coprocessor that assists the host in computing signatures, using a secure signing secret, when writing back the new balance to a roaming device after a purchase.

The DS1963S, like other SRAM-based iButtons, has an additional memory area called the scratchpad that acts as a buffer when writing to the main memory. The DS1963's scratchpad is also used for feeding data segments to the SHA-1 engine or receiving/comparing message authentication codes.

Data is first written to the scratchpad from where it can be read back. After the data has been verified, a copy scratchpad command will transfer the data to main memory. This process ensures data integrity in an environment that does not provide a reliable electric contact.

Each DS1963S has its own 64-bit ROM registration number that is factory lasered into the chip inside to provide a guaranteed unique identity for absolute traceability. The durable MicroCan package is highly resistant to environmental hazards such as dirt, moisture, and shock. Its compact coin-shaped profile is self-aligning with mating receptacles, allowing the DS1963S to be easily used by human operators. Accessories permit the DS1963S to be mounted on almost any surface including plastic key fobs, photo-ID badges and printed circuit boards.

SECURITY

A system that uses mobile data carriers consists mainly of three components, 1) host computers that read and write data carriers, 2) the data carriers ("slave devices") themselves, and 3) the users of the system who might be tempted to manipulate the data or to emulate the behavior of the data carrier. The DS1963S is designed to address all these areas of attacks without using any proprietary restricted algorithms. The security of the device is based on the Secure Hash Standard SHA-1, which is documented on the Internet at locations such as http://www.itl.nist.gov/div897/pubs/fip180-1.htm.

The table below shows a matrix of possible non-violent attacks in form of a truth table. The notes referenced in the table explain the typical methods to defeat the attacks. A more detailed description is found in the section "Application Overview" near the end of this document. For the full description of the functions used see section "Memory and SHA Function Commands" and the SHA-1 Computation and message formats.

	Authentic data	**Manipulated data**	
Authorized Host	See note 2	See notes 2 and 3	**Emulated slave**
	Normal operation	See note 3	**Authentic Slave**
Unauthorized Host	See note 1	Don't care	
	Don't care	Don't care	**Emulated slave**

Note 1: The device provides functions to authenticate the host based on a system-wide secret, the device's ROM Registration number and a user-selected pin number that is installed in one of the memory pages of a roaming data carrier.

Note 2: To find out whether a slave device is authentic the host writes a 3-byte "challenge" to the scratchpad before issuing a command to compute the SHA-1 MAC over the challenge, the data of a memory page, the page number, the page's write-cycle counter, the device's ROM Registration number, and the secret associated with that page. By varying the challenge every time it reads from a slave, the host can verify that the slave contains the correct secret and can perform the required SHA computation in the required time.

Note 3: Manipulated data can be discovered if the data in the slave device is "signed" by an authorized host. Signing consists of calculating a 160-bit SHA-1 MAC over the data to be protected, the write-cycle counter of the page on which it is to be stored, the ROM ID of the slave device in which it is to be stored, and any dedicated secret known only to authorized hosts. The MAC is stored together with the application data (a monetary value together with a transaction ID code, for example) in an appropriate memory page. To verify the authenticity of the data the host repeats the process of signing. Any change in the data, the cycle counter, data carrier or an invalid (not belonging to the system) signing-secret will make the verification of the signature fail.

OVERVIEW

The block diagram in Figure 1 shows the relationships between the major control and memory sections of the DS1963S. The DS1963S has six main data components: 1) 64-bit lasered ROM, 2) 256-bit scratch-pad, 3) eight 32-byte pages of general-purpose SRAM, 4) eight 32-byte pages of SRAM protected by write-cycle counters, 5) two 32-byte pages storing eight 64-bit secrets with individual write-cycle counters, and 6) a 512-bit SHA-1 Engine (SHA = Secure Hash Algorithm). The hierarchical structure of the 1-Wire protocol is shown in Figure 2. All write-cycle counters are 32 bits long and will not roll over once the maximum count has been reached. The contents of the counters is read together with the memory data using a special command. The bus master must first provide one of the seven ROM Function Commands, 1) Read ROM, 2) Match ROM, 3) Search ROM, 4) Skip ROM, 5) Resume Communication, 6) Overdrive-Skip ROM or 7) Overdrive-Match ROM. Upon completion of an Overdrive ROM command byte executed at standard speed, the device will enter Overdrive mode where all subsequent communication occurs at a higher speed. The protocol required for these ROM function commands is described in Figure 10. After a ROM function command is successfully executed, the memory functions become accessible and the master may provide any one of the eight memory function commands. The protocol for these memory function commands is described in Figure 7. All data is read and written least significant bit first.

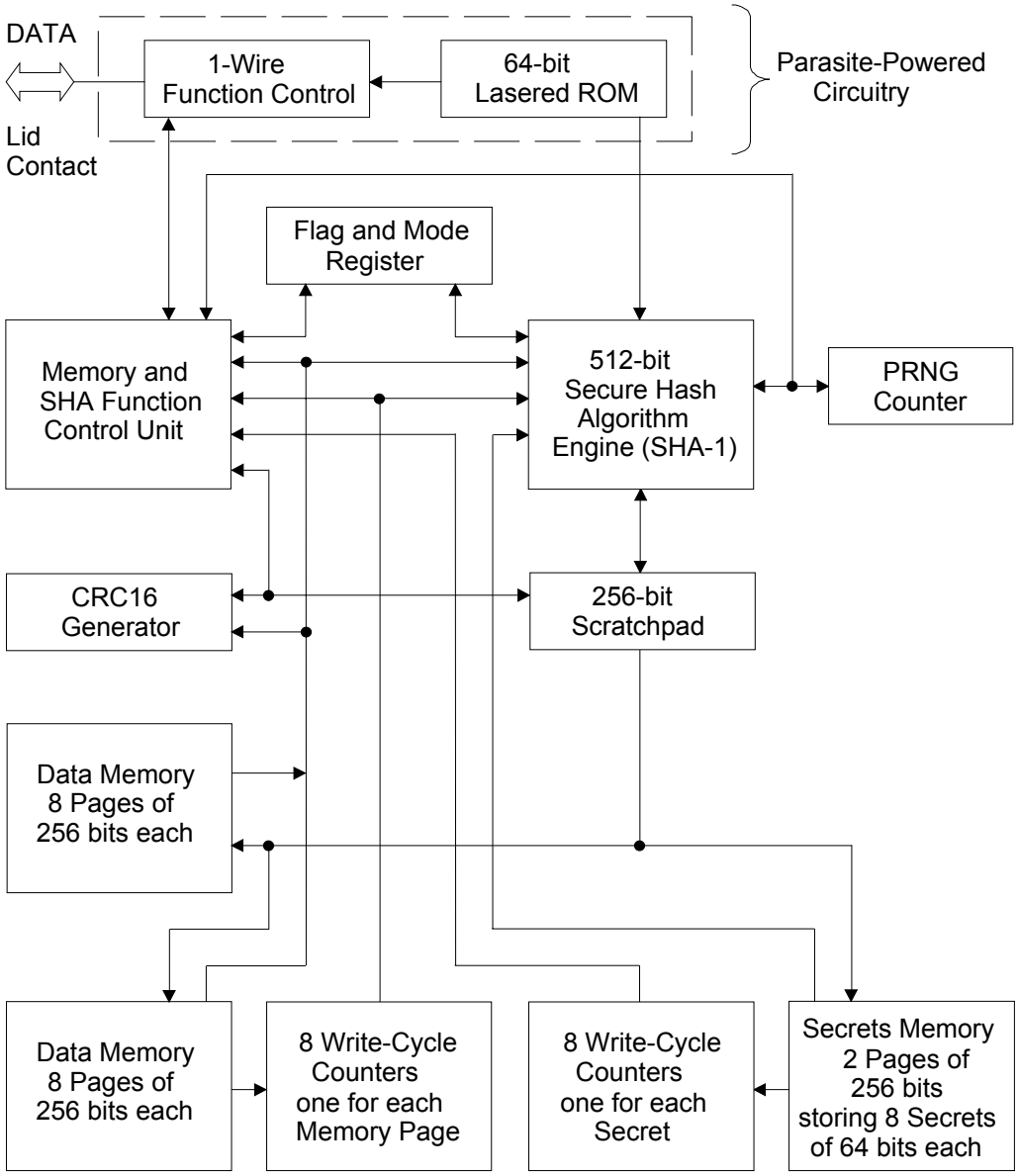
PARASITE POWER

The block diagram (Figure 1) shows the parasite-powered circuitry. This circuitry "steals" energy whenever the data contact is in the logic-high state. This stolen energy will provide sufficient power while the data contact is in a logic-low state as long as the specified timing and voltage requirements are met. The advantages of parasite power are two-fold: 1) by stealing energy off this input, the DS1963S-internal lithium reserves are conserved and 2) if the lithium is exhausted for any reason, the ROM may still be read normally. The remaining circuitry of the DS1963S is solely operated by lithium energy.

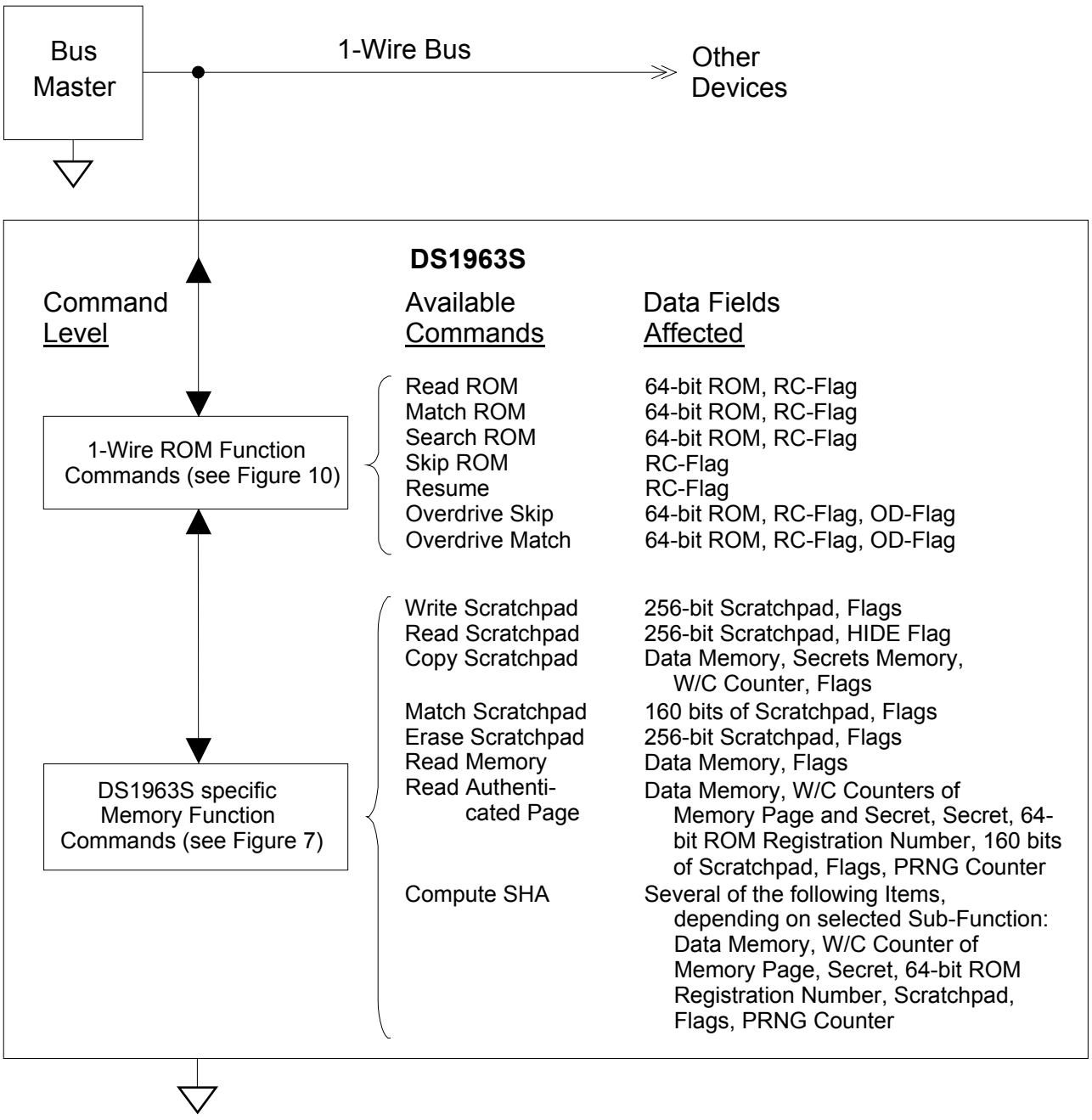
64-BIT LASERED ROM

Each DS1963S contains a unique ROM code that is 64 bits long. The first 8 bits are a 1-Wire family code. The next 48 bits are a unique serial number. The last 8 bits are a CRC of the first 56 bits. (See Figure 3). The 1-Wire CRC is generated using a polynomial generator consisting of a shift register and XOR gates as shown in Figure 4. The polynomial is $X^8 + X^5 + X^4 + 1$. Additional information about the Dallas 1-Wire Cyclic Redundancy Check is available in the *Book of DS19xx iButton Standards*. The shift register bits are initialized to zero. Then starting with the least significant bit of the family code, one bit at a time is shifted in. After the 8th bit of the family code has been entered, then the serial number is entered. After the 48th bit of the serial number has been entered, the shift register contains the CRC value. Shifting in the 8 bits of CRC should return the shift register to all zeros.

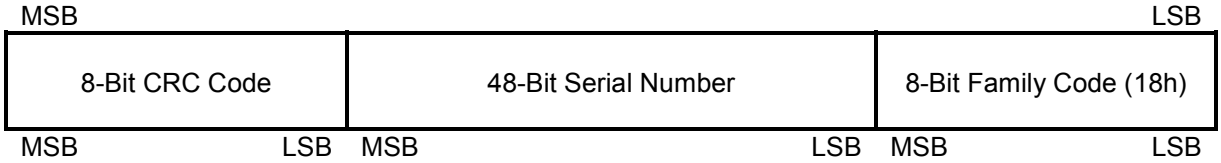
DS1963S BLOCK DIAGRAM Figure 1



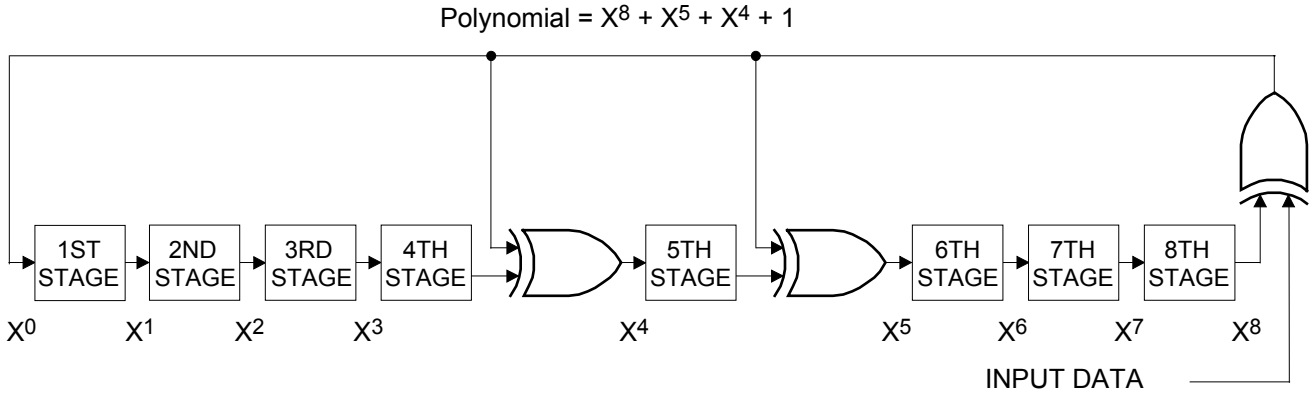
HIERARCHCAL STRUCTURE FOR 1-WIRE PROTOCOL Figure 2



64-BIT LASERED ROM Figure 3



1-WIRE CRC GENERATOR Figure 4



MEMORY MAP

As shown in the block diagram, the DS1963S has four memory areas: data memory, secrets memory, counter memory, and scratchpad. Each of these memory areas is organized in pages of 32 bytes. Figure 5 shows details. The scratchpad acts as a buffer when writing to data or secrets memory. Pages 0 to 15 have unrestricted read/write access. They account for the 4096 bits of NV SRAM. Pages 16 and 17 contain the eight 64-bit secrets to which the user only has write access. The secrets are readable only by the SHA engine, which uses them to compute message authentication codes. Sixteen 32-bit write-cycle counters count write-accesses to pages 8 to 15 as well as to the eight secrets. These counters are located in pages 19 and 20 and can be read without restriction. Page 21 contains a counter, which increments with every start of the SHA engine. This counter provides the seed for generating pseudo-random numbers and therefore is referred to as PRNG counter. Since the SHA engine requires about 20 times as much energy as copying the entire scratchpad to a memory location the PRNG counter can be used as indicator for the remaining energy reserves of the device. Page 18 is the physical location of the 32-byte scratchpad.

ADDRESS REGISTERS AND TRANSFER STATUS

The DS1963S employs three address registers: TA1, TA2 and E/S (Figure 6). Registers TA1 and TA2 must be loaded with the target address to which data will be written or from which data will read. Register E/S is a read-only byte counter and transfer-status register, used to verify data integrity with write commands. The lower 5 bits of the E/S register indicate the address of the last byte that has been written to the scratchpad for subsequent copying into main memory. This address is called Ending Offset. Bit 5 of the E/S register, called PF or "partial byte flag," is a logic-1 if the number of data bits sent by the master is not an integer multiple of 8. Bit 6 has no function; it always reads 0. Note that the lowest 5 bits of the target address also determine the address within the scratchpad where intermediate storage of data will begin. This address is called byte offset. If the target address (TA1) for a Write command is 3CH for example, then the scratchpad will store incoming data beginning at byte offset 1CH and will be full after only 4 bytes, resulting in an ending offset of 1FH. The ending offset together with the Partial Flag support the master checking the data integrity after a Write command. The highest valued bit of the E/S register, called AA or Authorization Accepted, acts as a flag to indicate that the data stored in the scratchpad has already been copied to the target memory address. Writing data to the scratchpad clears this flag.

DS1963S MEMORY MAP Figure 5

4k-Bit NV RAM

Data Memory with General Read/Write Access

Page #	Address Range	Secret #	Counter #	Counter Incr.
0	0000h to 001Fh	0	0	None
1	0020h to 003Fh	1	1	None
2	0040h to 005Fh	2	2	None
3	0060h to 007Fh	3	3	None
4	0080h to 009Fh	4	4	None
5	00A0h to 00BFh	5	5	None
6	00C0h to 00DFh	6	6	None
7	00E0h to 00FFh	7	7	None
8	0100h to 011Fh	0	0	With write cycle
9	0120h to 013Fh	1	1	With write cycle
10	0140h to 015Fh	2	2	With write cycle
11	0160h to 017Fh	3	3	With write cycle
12	0180h to 019Fh	4	4	With write cycle
13	01A0h to 01BFh	5	5	With write cycle
14	01C0h to 01DFh	6	6	With write cycle
15	01E0h to 01FFh	7	7	With write cycle

Secrets Memory with User Write Access Only

Page #	Address Range	Description
16	0200h to 0207h	Secret 0
	0208h to 020Fh	Secret 1
	0210h to 0217h	Secret 2
	0218h to 021Fh	Secret 3
17	0220h to 0227h	Secret 4
	0228h to 022Fh	Secret 5
	0230h to 0237h	Secret 6
	0238h to 023Fh	Secret 7

DS1963S MEMORY MAP (continued) Figure 5

Counter Memory with User Read Access Only

Page #	Address Range	Description
19	0260h to 0263h	Counter 0 (Write Cycles to Page 8)
	0264h to 0267h	Counter 1 (Write Cycles to Page 9)
	0268h to 026Bh	Counter 2 (Write Cycles to Page 10)
	026Ch to 026Fh	Counter 3 (Write Cycles to Page 11)
	0270h to 0273h	Counter 4 (Write Cycles to Page 12)
	0274h to 0277h	Counter 5 (Write Cycles to Page 13)
	0278h to 027Bh	Counter 6 (Write Cycles to Page 14)
	027Ch to 027Fh	Counter 7 (Write Cycles to Page 15)
20	0280h to 0283h	Write Cycle Counter Secret 0
	0284h to 0287h	Write Cycle Counter Secret 1
	0288h to 028Bh	Write Cycle Counter Secret 2
	028Ch to 028Fh	Write Cycle Counter Secret 3
	0290h to 0293h	Write Cycle Counter Secret 4
	0294h to 0297h	Write Cycle Counter Secret 5
	0298h to 029Bh	Write Cycle Counter Secret 6
	029Ch to 029Fh	Write Cycle Counter Secret 7
21	02A0h to 02A3h	PRNG Counter

ADDRESS REGISTERS Figure 6

Target Address (TA1)	T7	T6	T5	T4	T3	T2	T1	T0
Target Address (TA2)	T15	T14	T13	T12	T11	T10	T9	T8
Ending Address with Data Status (E/S) (Read Only)	AA	0	PF	E4	E3	E2	E1	E0

WRITING WITH VERIFICATION

To write data to the DS1963S, the scratchpad has to be used as intermediate storage. First the master issues the Write Scratchpad command to specify the desired target address, followed by the data to be written to the scratchpad. Under certain conditions (see Write Scratchpad command) the master will receive an inverted CRC16 of the command, address and data at the end of the write scratchpad command sequence. Knowing this CRC value, the master can compare it to the value it has calculated itself to decide if the communication was successful and proceed to the Copy Scratchpad command. If the master could not receive the CRC16, it should send the Read Scratchpad command to verify data integrity. As preamble to the scratchpad data, the DS1963S repeats the target address TA1 and TA2 and sends the contents of the E/S register. If the PF flag is set, data did not arrive correctly in the scratchpad. The master does not need to continue reading; it can start a new trial to write data to the scratchpad. Similarly, a set AA flag indicates that the device did not recognize the Write command. If everything went

correctly, both flags are cleared and the ending offset indicates the address of the last byte written to the scratchpad. Now the master can continue reading and verifying every data byte. After the master has verified the data, it can send the Copy Scratchpad command. This command must be followed exactly by the data of the three address registers TA1, TA2 and E/S. The master may obtain the contents of these registers by reading the scratchpad or derive it from the target address and the amount of data to be written. As soon as the DS1963S has received these bytes correctly, it will copy the data to the requested location beginning at the target address.

MEMORY AND SHA FUNCTION COMMANDS

Due to its design as a secure device the DS1963S has to behave differently from other Memory iButtons. Although the data memory of the DS1963S can be read the same way as any other NV SRAM based Memory iButton, attempts to read pages 16 and 17, which store the secrets, and page 18, the physical location of the scratchpad, will result in FFh-bytes rather than real data. The other functions that the DS1963S shares with regular Memory iButtons are governed by a flag called HIDE. Once this HIDE flag is cleared these functions behave the same as with other NV SRAM based devices. The HIDE flag is mainly controlled (set or cleared) by the functions that involve the SHA engine. In order to prevent scratchpad data from accidentally being exposed, the HIDE flag is automatically set as the parasite-powered circuit performs a power-on reset whenever the DS1963S returns to a probe point. The HIDE flag is then cleared by issuing an Erase Scratchpad command, which also erases all the data left in the scratchpad.

The "Memory and SHA Function Flow Chart" (Figure 7) describes the protocols necessary for accessing the memory and operating the SHA engine. The communication between master and DS1963S takes place either at regular speed (default, OD = 0) or at Overdrive Speed (OD = 1). If not explicitly set into the Overdrive Mode the DS1963S assumes regular speed.

Write Scratchpad Command [0Fh]

HIDE = 0, Target Address range 0000h to 01FFh only

After issuing the write scratchpad command, the master must first provide the 2–byte target address, followed by the data to be written to the scratchpad. The data will be written to the scratchpad starting at the byte offset (T4:T0). The ending offset (E4: E0) will be the byte offset at which the master stops writing data. Only full data bytes are accepted. If the last data byte is incomplete its content will be ignored and the partial byte flag PF will be set.

When executing the Write Scratchpad command the CRC generator inside the DS1963S (see Figure 12) calculates a CRC of the entire data stream, starting at the command code and ending at the last data byte sent by the master. This CRC is generated using the CRC16 polynomial by first clearing the CRC generator and then shifting in the command code (0FH) of the Write Scratchpad command, the Target Addresses TA1 and TA2 as supplied by the master and all the data bytes. The master may end the Write Scratchpad command at any time. However, if the ending offset is 11111b, the master may send 16 read time slots and will receive the CRC generated by the DS1963S.

HIDE = 1: Target Address range 0200h to 023Fh only

The function of the command is limited to selecting the secret that will be overwritten by the data currently stored in the scratchpad, which is typically the result of a previously executed Compute First Secret or Compute Next Secret command. The addresses of the eight secrets are shown in Figure 5. The address transmitted after the command code may point to anywhere within the address range of the secret. Following the target address the master may transmit data bytes as if writing to the scratchpad. Once as many data bytes have been transmitted as would fit into the scratchpad beginning at the specified target

address, the master may send 16 read time slots and will receive the CRC generated by the DS1963S. The data bytes are used in the CRC calculation but are not actually written to the scratchpad.

Read Scratchpad Command [AAh]

HIDE = 0:

The Read Scratchpad command allows verifying the target address, ending offset and the integrity of the scratchpad data. After issuing the command code the master begins reading. The first 2 bytes will be the target address. The next byte will be the ending offset/data status byte (E/S) followed by the scratchpad data beginning at the byte offset (T4: T0). The master may read data until the end of the scratchpad after which it will receive the inverted CRC generated by the DS1963S. If the master continues reading after the CRC all data will be logic 1's.

HIDE = 1:

The function of the command is limited to reading the target address and ending offset. Instead of scratchpad data the master will read logic 1's until, based on the target address read, the end of the scratchpad is reached, at which point the master will receive the CRC generated by the DS1963S. If the master continues reading all data will be logic 1's.

Copy Scratchpad [55h]

HIDE = 0, Target Address range 0000h to 01FFh only

The Copy Scratchpad command is used to copy data from the scratchpad to a memory page. After issuing the command, the master must provide a 3-byte authorization pattern, which should have been obtained by an immediately preceding Read Scratchpad command. This 3-byte pattern must exactly match the data contained in the three address registers (TA1, TA2, E/S, in that order). If the pattern matches, the AA (Authorization Accepted) flag will be set and the copy will begin. While the data is being copied the master will read logic 1's. A pattern of alternating 1's and 0's will be transmitted after the data has been copied until the master issues a reset pulse. Any attempt to reset the part will be ignored while the copy is in progress. The copy operation typically takes 30 µs. The master must read at least 8 bits of this alternating pattern. Otherwise the device might not properly respond to a subsequent Reset Pulse.

The data to be copied is determined by the three address registers. The scratchpad data from the beginning offset through the ending offset will be copied to memory, starting at the target address. Anywhere from 1 to 32 bytes may be copied to memory with this command. Only executing a write scratchpad command will clear the AA flag.

HIDE = 1: Target Address range 0200h to 023Fh only

The function follows the regular flow as described above if the target address and ending offset match the address of a secret. If the target address points to a location within the main memory address range but the HIDE-flag is set (due to a power-on reset of the parasite-powered circuit, for example) no scratchpad data will be copied. Conversely, one can copy known data ("password") to a secret by writing data to the scratchpad, setting the HIDE flag, issuing a Write Scratchpad command to select a secret and then issuing a Copy Scratchpad command. This procedure, however, is not recommended since it compromises the achievable level of security.

Read Memory [F0h]

The read memory command may be used to read memory pages 0 to 15, the write cycle counters located in pages 19 and 20 and the PRNG counter at the beginning of page 21. Trying to read the secrets that are located in pages 16 and 17 will not reveal any data. Attempts to read page 18 will return the scratchpad data if HIDE flag is cleared (HIDE=0) and FFh values if the flag is set (HIDE=1). After issuing the

command, the master must provide the 2-byte target address. After these 2 bytes, the master reads data beginning from the target address and may continue until the end of the PRNG counter and beyond. The 12 bytes following the PRNG counter are undefined. If the master continues reading the result will be logic 1's. It is important to realize that the target address registers will point to the last byte read. The ending offset/data status byte is unaffected.

The hardware of the DS1963S provides a means to accomplish error-free writing to the memory section. To safeguard reading data in the 1-Wire environment and to simultaneously speed up data transfers, it is recommended to packetize data into data packets of the size of one memory page each. Such a packet would typically store a master-calculated 16-bit CRC with each page of data to insure rapid, error-free data transfers that eliminate having to read a page multiple times to determine if the received data is correct or not. (See Application Note 114 for the recommended file structure, which is also referred to as TMEX Format.)

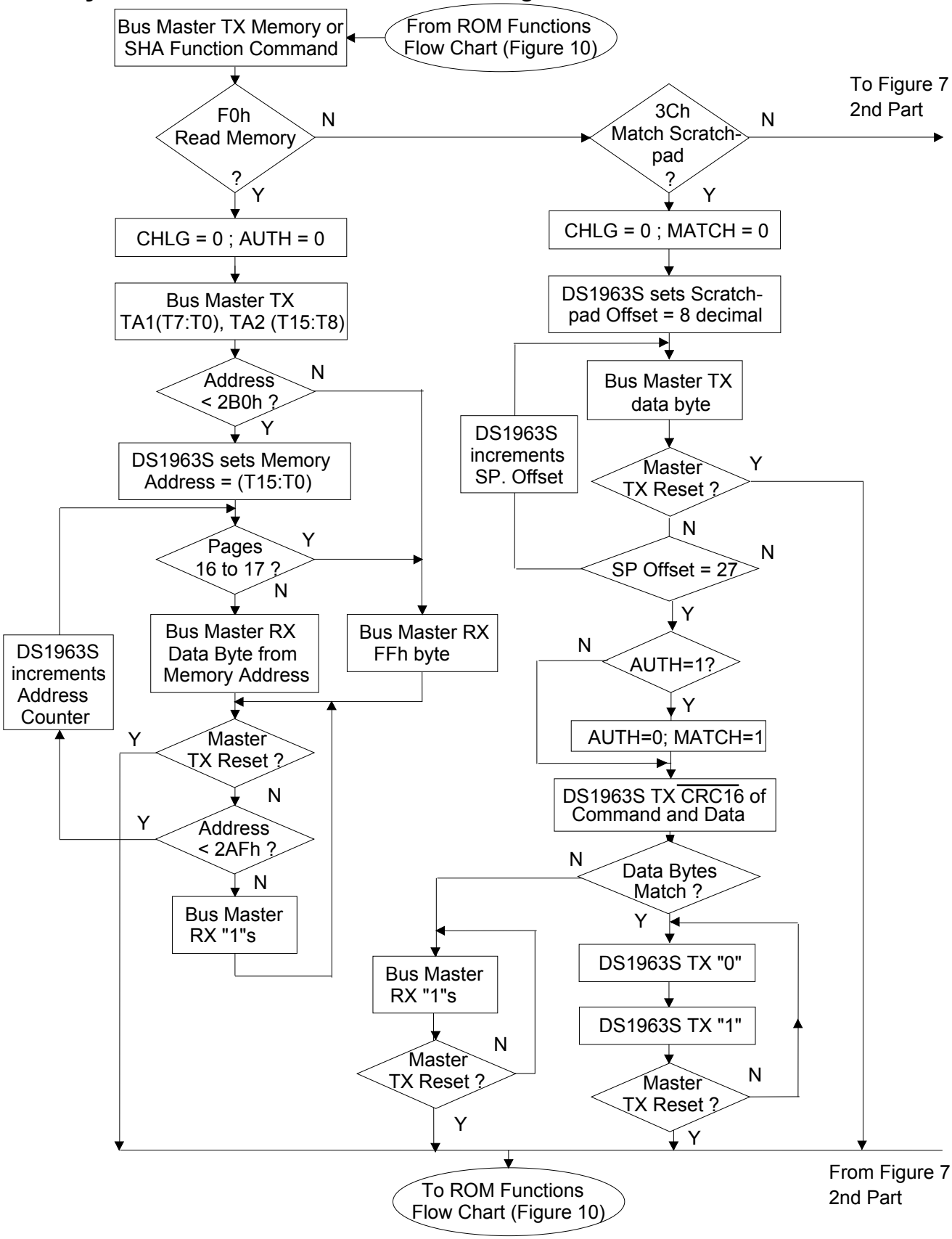
Erase Scratchpad [C3h]

The purpose of this command is to clear the HIDE flag and to wipe out data that might have been left in the scratchpad from a previous operation. After having issued the command code the bus master transmits a target address, as with the write scratchpad command, but no data. Next the whole scratchpad will be automatically filled with FFh bytes, regardless of the target address. This process takes approximately 32 µs during which the master reads 1's. After this the master reads a pattern of alternating 0's and 1's indicating that the command has completed. The master must read at least 8 bits of this alternating pattern. Otherwise the device might not properly respond to a subsequent Reset Pulse.

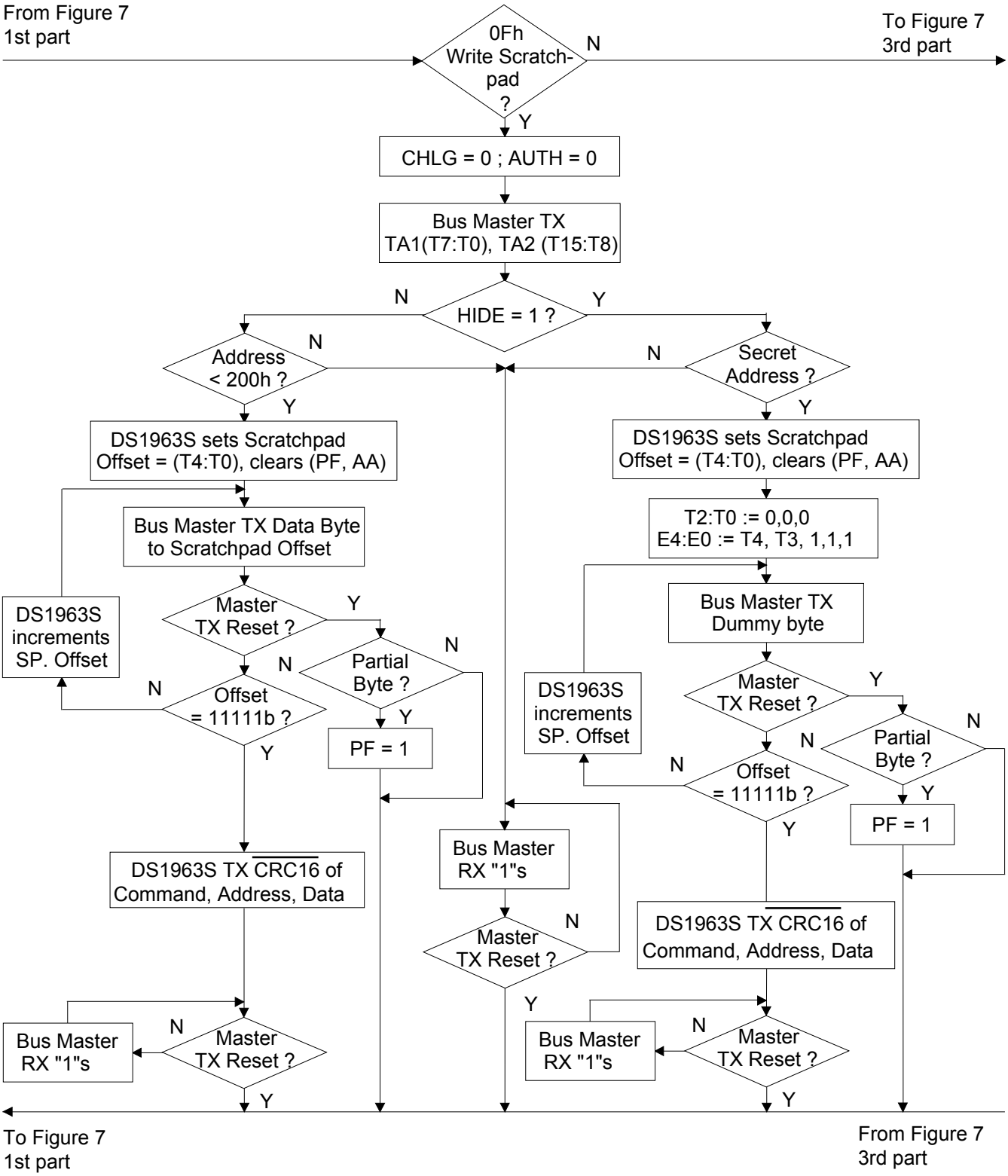
Match Scratchpad [3Ch]

SHA-1 MACs calculated by the DS1963S are written into the scratchpad. Some calculations such as those done by the Authenticate Host or Validate Data Page function cause the HIDE flag to be set as well. The Match Scratchpad command allows this data to be checked without making it publicly readable. The command compares the 160-bit Message Authentication Code which is found in scratchpad locations 8 through 27 after a SHA computation, as described in the sections "SHA-1 Computation Algorithm" and "SHA-1 Output Message Formats", to the result that the master has computed by its own means. After the master has issued the Match Scratchpad command code it transmits 1 byte after another starting with byte 8 and ending with byte 27. If all bytes match, the master will read a pattern of alternating 0's and 1's. If in addition the AUTH-flag was set, the MATCH-flag will be set. If the comparison was not successful the master will read all 1's. The master must read at least 8 bits after having received the inverted CRC16. Otherwise the device might not properly respond to a subsequent Reset Pulse.

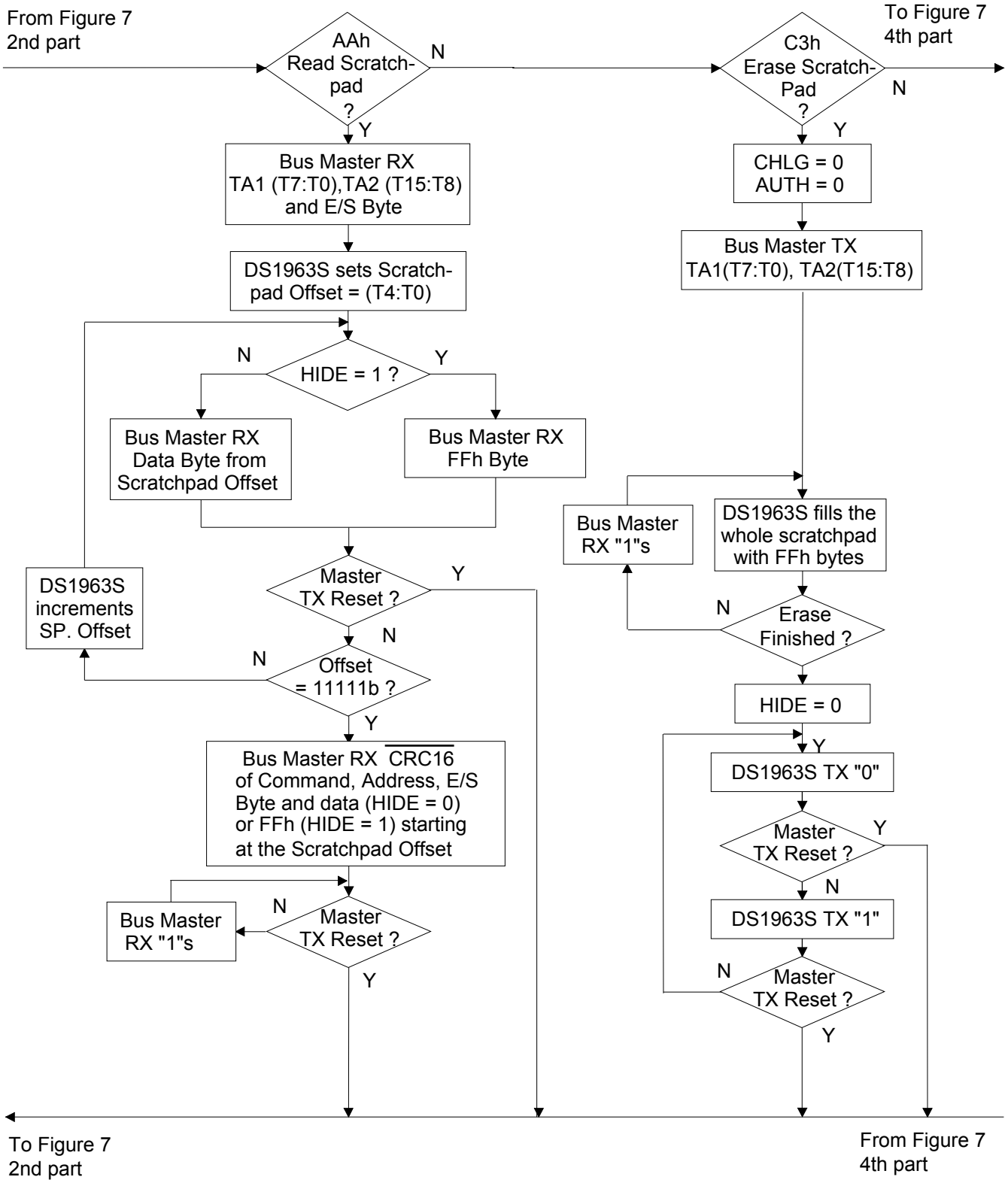
Memory and SHA Function Flow Chart Figure 7



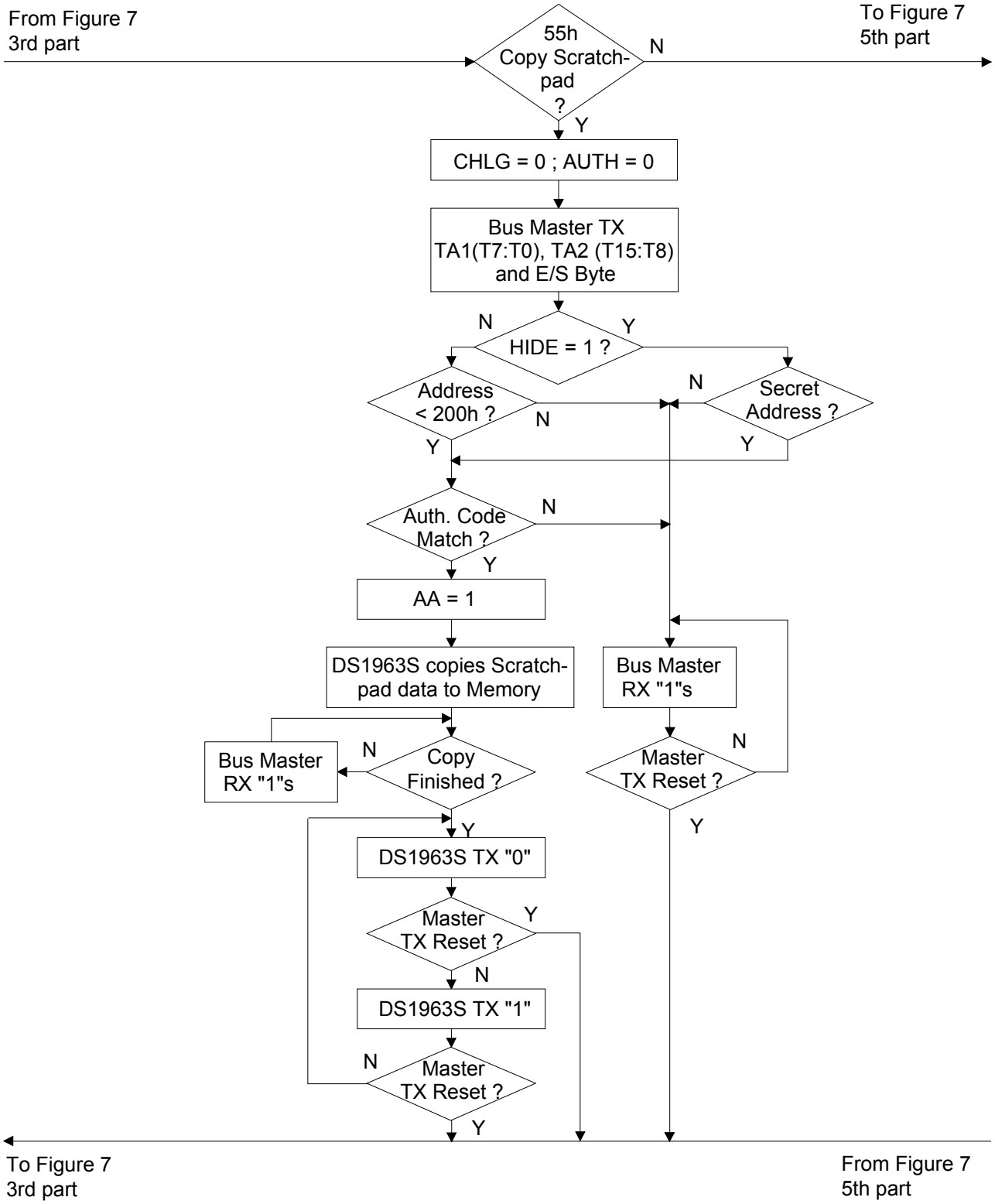
Memory and SHA Function Flow Chart (continued) Figure 7



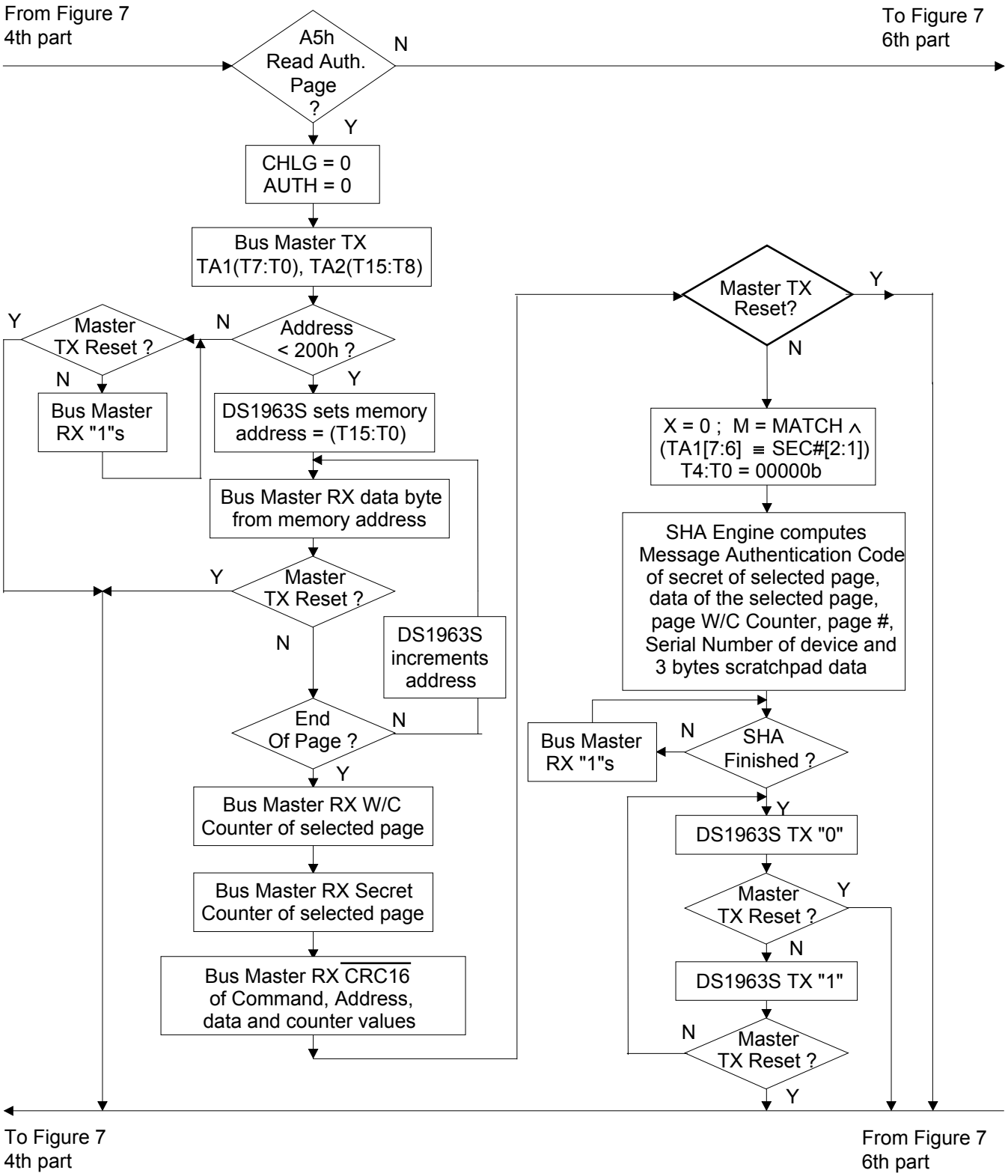
Memory and SHA Function Flow Chart (continued) Figure 7



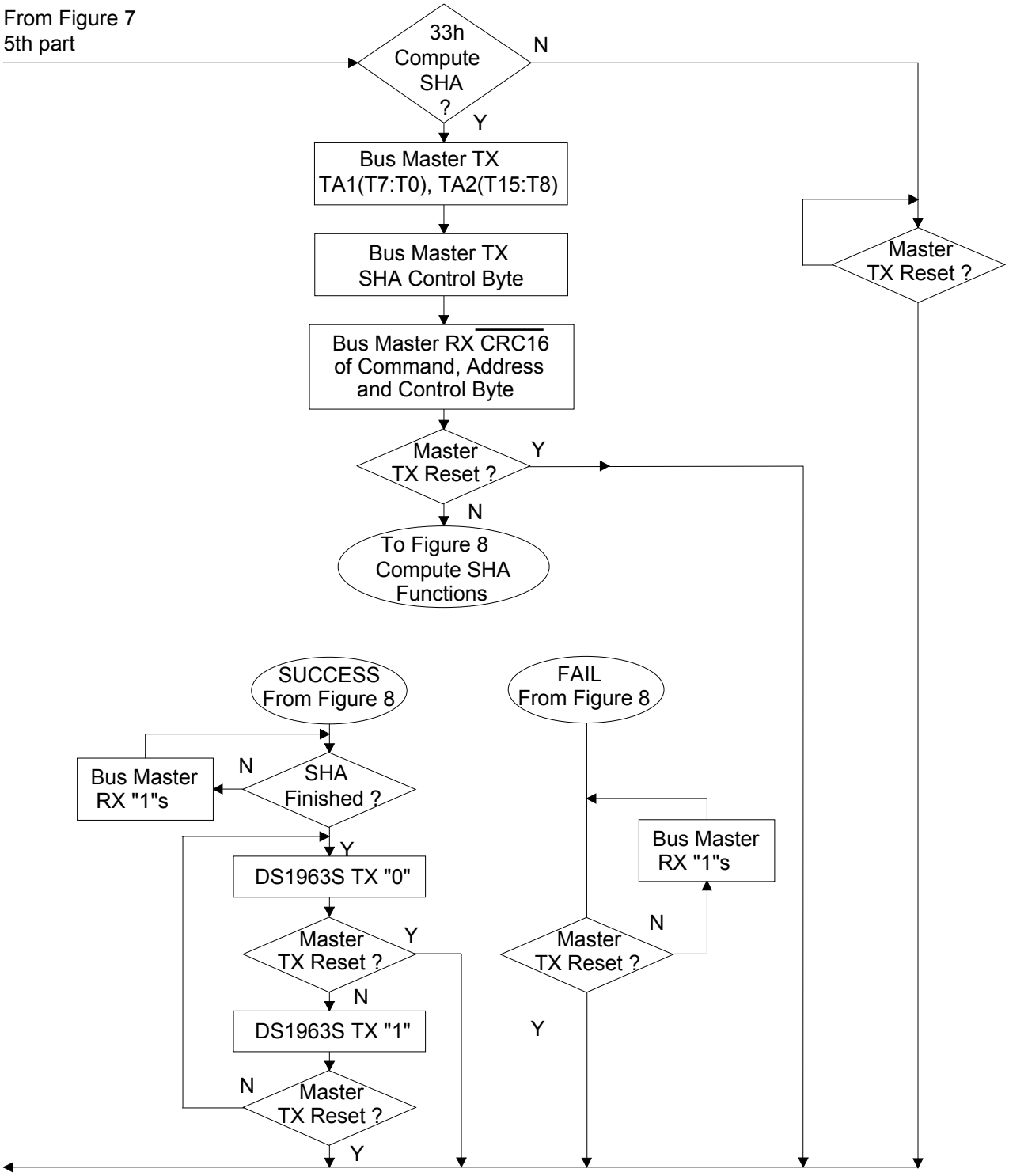
Memory and SHA Function Flow Chart (continued) Figure 7



Memory and SHA Function Flow Chart (continued) Figure 7



Memory and SHA Function Flow Chart (continued) Figure 7



Read Authenticated Page [A5h]

This command, which is applicable to pages 0 to 15 only, combines reading a full or partial memory page with the computation of a SHA-1 message authentication code. After the master has issued the command code and specified a valid target address it will receive the page data beginning at the target address through the end of the data page, the value of the write-cycle counter of the page, the value of the write cycle of the secret associated with the page and the inverted CRC of the command code, target address, page data and counter values. Immediately after the CRC is received the SHA engine begins the computation of the message authentication code over the secret associated with the selected page, all 32 data bytes of the selected page, the page write cycle counter, page number, the device's ROM registration number and the 3-byte "challenge" that is taken from the scratchpad locations 20 through 22. The result of the SHA computation is then placed in the scratchpad from location 8 through 27 for the master to read. While the SHA computation takes place the master will read all 1's. As the computation is finished the pattern will change to alternating 0's and 1's. The master must read at least 8 bits of this alternating pattern. Otherwise the device might not properly respond to a subsequent Reset Pulse. Typically the master will next take all the page data, etc., compute the message authentication code on its own (see the Compute SHA command, "Validate Data Page" function), and compare it to the data in the scratchpad to determine whether the DS1963S knows the correct secret associated with the data page.

Compute SHA [33h]

The Compute SHA command provides the environment for six functions that employ the SHA engine to generate message authentication codes in different ways. The seventh way to run the SHA engine is through the Read Authenticated Page command, which has been described above to some extent. The full details of all SHA computations are found in this section. Table 1 gives an overview of these functions.

SHA Functions Overview Table 1

Command or Function Name	Roaming Button	Coprocessor Button	Applicability
Read Authenticated Page	yes	N/A	Pages 0 to 15
Validate Data Page Function	N/A	yes	Pages 0 to 15
Sign Data Page Function	N/A	yes	Pages 0 and 8 only
Compute Challenge Function	yes	N/A	Not with pages 0 and 8
Authenticate Host Function	yes	N/A	Not with pages 0 and 8
Compute first Secret Function	yes	yes	Pages 0 to 15
Compute next Secret Function	yes	yes	Pages 0 to 15

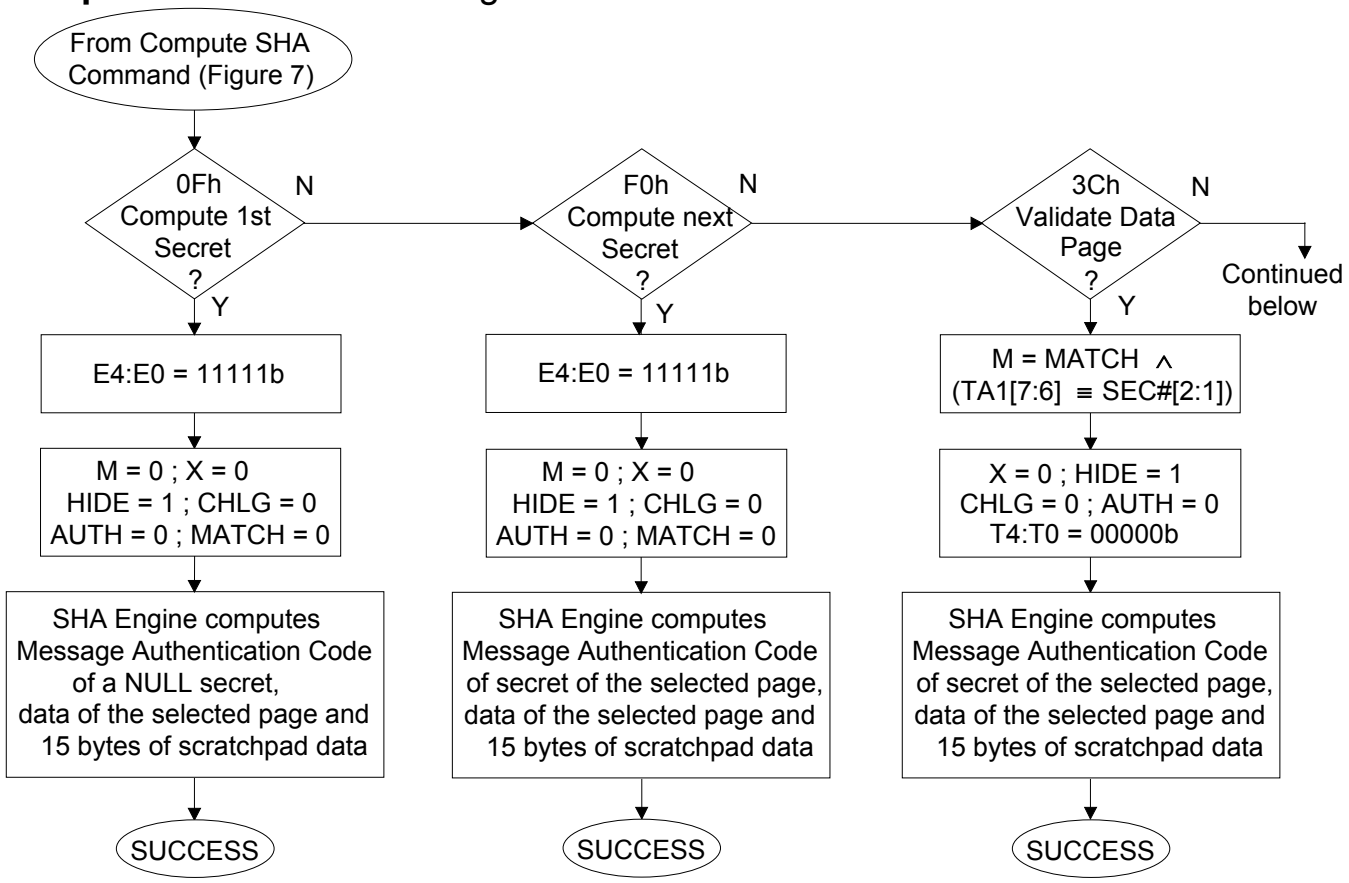
A DS1963S device can be used in a system in two ways: a) as mobile data carrier that is associated to a person that carries it, and b) as a coprocessor and data safe for a host computer or "bus master". Either application requires a secret to be installed in the DS1963S device. The functions needed to install secrets in one or more steps are called Compute First Secret and Compute Next Secret. A DS1963S that works as a coprocessor has to accomplish two functions: a) verify whether a roaming device belongs to the system (i. e., whether it knows the secret), and b) generate or check a signature that protects data from manipulation. These functions are accomplished by the functions Validate Data Page and Sign Data Page.

The main SHA function of a roaming device is Read Authenticated Page, which provides the coprocessor device data and message authentication code needed for the Validate Data Page function. The two remaining SHA functions that a roaming device may have to execute are Compute Challenge and Authenticate Host. These functions are not used in applications such as vending machines. However, they are essential for user- and host authentication, which will set the MATCH flag of the roaming device. Since the MATCH flag is part of the SHA computation of Read Authenticated Page, Validate Data Page, and Sign Data Page, the resulting message authentication code depends on and therefore reveals whether the host

authentication was successful. User- and host authentication, if implemented, prevents the use of a DS1963S as a coprocessor device since it requires several steps to get the MATCH flag set.

After transmitting the command code the bus master selects a memory page and its secret by transmitting a target address anywhere within the page. Next the master transmits the SHA Control byte, which is a code for one of the six SHA functions that can be performed. Next the master receives a CRC over the command code, address, and control byte. As the CRC is received and the control byte and address were valid the SHA engine will start immediately and compute a message authentication code as described in Figure 8. While the SHA computation takes place the master will read all 1's. As the computation is finished the pattern will change to alternating 0's and 1's. The master must read at least 8 bits of this alternating pattern. Otherwise the device might not properly respond to a subsequent Reset Pulse. In case of an invalid control byte or address the master will continue reading all 1's until it issues a Reset Pulse. The exact location of the various data segments as they enter the input buffer of the SHA engine is shown in Table 2.

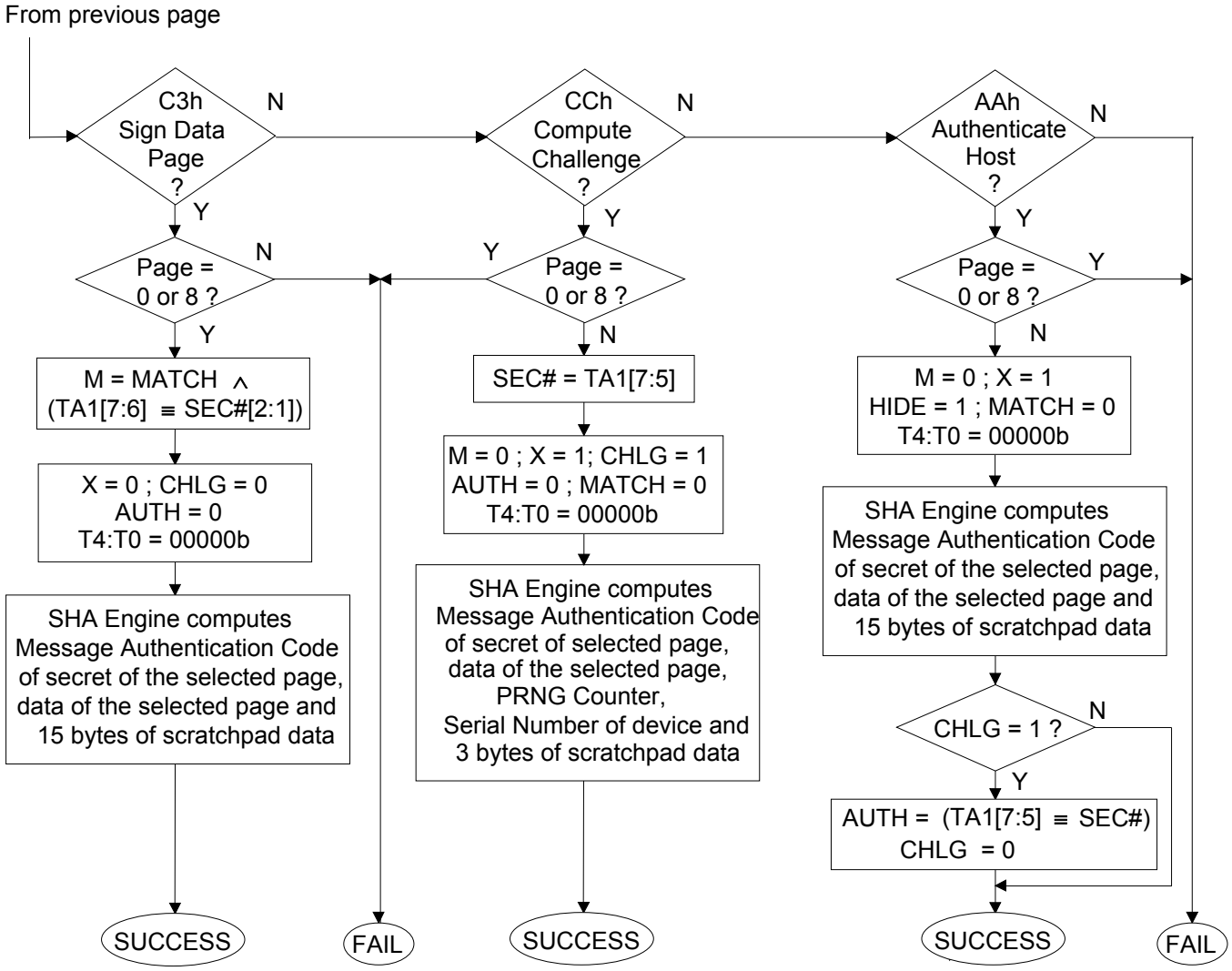
Compute SHA Functions Figure 8



Read Authenticated Page and Compute Challenge allow the master to input a 3-byte "challenge" in the computation via scratchpad locations 20 through 22. All other data is taken from the selected memory page, associated secret, cycle counter, ROM Registration number and flags. With Compute First Secret and Compute Next Secret the scratchpad locations 8 through 22 need to be filled with a partial secret before the SHA computation takes place. A coprocessor device performing a Validate Data Page or Sign Data Page command must have in scratchpad bytes 8 through 11 the (incremented) value of the cycle counter of the selected memory page of the roaming device, and it must have in bytes 13 through 19 the ROM Registration Number (without CRC), and in byte 12 the page number. A roaming device

performing an Authenticate Host command should have first performed the Compute Challenge function in order to fill the scratchpad with pseudo-random data.

Compute SHA Functions (continued) Figure 8



The Compute Challenge function stores the upper 3 bits of TA1 in a latch called SEC#, which is next used with the Authenticate Host function. Only if Authenticate Host and Compute Challenge call upon the same memory page (same secret) will the AUTH-flag be set. This prevents the AUTH-flag from being set with the secret of another page that may belong to a different application or service provider.

The two most significant bits of SEC# are also used with Validate Data Page, Sign Data Page and Read Authenticated Page when determining the M control bit. This makes a difference only for those applications that use host/user authentication. The M control bit is only set if the MATCH-flag is set and the target memory page is adjacent to the page that was used for authentication. This allocates one pair of secrets (0 and 1, 2 and 3, 4 and 5, 6 and 7) and the pages associated with these secrets to one service provider.

SHA-1 Input Message Formats Table 2

Read Authenticated Page command, Compute Challenge function

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] = (CC+0)	M9[23:16] = (CC+1)	M9[15:8] = (CC+2)	M9[7:0] = (CC+3)
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+20)	M13[23:16] = (SP+21)	M13[15:8] = (SP+22)	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** See Figure 5, Memory Map, memory pages 16 and 17
CC	**Starting address of cycle counter** *Read Authenticated Page Command:* Write cycle counter of selected memory page, see Figure 5, Memory Map, memory page 19; the LS-byte of the counter is stored at the lower address *Compute Challenge:* PRNG Counter; see Figure 5, Memory Map, memory page 21; the LS-byte of the counter is stored at the lower address
PP	**Starting address of memory page** See Figure 5, Memory Map, memory pages 0 through 15
FAMC	**Family Code = 18h**
MP	MP[7] = Control bit M, see Figure 7, Read Authenticated Page, and Figure 8 MP[6] = Control bit X, see Figure 7, Read Authenticated Page, and Figure 8 MP[5:4] = 00b 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** The counting of n is in decimal

SHA-1 Input Message Formats (continued) Table 2

Validate Data Page, Sign Data Page, Authenticate Host, Compute First Secret, Compute Next Secret

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] = (SP+8)	M9[23:16] = (SP+9)	M9[15:8] = (SP+10)	M9[7:0] = (SP+11)
M10[31:24] = MPX	M10[23:16] = (SP+13)	M10[15:8] = (SP+14)	M10[7:0] = (SP+15)
M11[31:24] = (SP+16)	M11[23:16] = (SP+17)	M11[15:8] = (SP+18)	M11[7:0] = (SP+19)
M12[31:24] = (SS+4)	M12[23:16] = (SS+5)	M12[15:8] = (SS+6)	M12[7:0] = (SS+7)
M13[31:24] = (SP+20)	M13[23:16] = (SP+21)	M13[15:8] = (SP+22)	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** See Figure 5, Memory Map, memory pages 16 and 17 With Compute First Secret the secret data is replaced by all zeros.
PP	**Starting address of memory page** See Figure 5, Memory Map, memory pages 0 through 15
MPX	MPX[7] = Control bit M, see Figure 8 MPX[6] = Control bit X, see Figure 8 MPX[5:0] = (SP+12)[5:0]
(SP+n)	**Byte n of scratchpad** The counting of n is in decimal

The SHA functions as well as the memory functions involve several flags that may affect the function itself and the result of functions executed in subsequent steps. These flags are HIDE, CHLG, AUTH and MATCH. Table 3 summarizes the operation of these flags. The only command that does not change any flag is Read Scratchpad. Note that a power-on reset of the parasite-powered 1-Wire front end of the device also affects the flags. This "Return to Probe" condition occurs typically when a DS1963S device makes contact with a read/write probe of a host computer or bus master or if the connection is intermittent. The most apparent is the HIDE-flag. If set it prevents the user from reading the data in the scratchpad; the current value of the target address and E/S byte remain readable, though. The HIDE-flag also affects the Write Scratchpad and Copy Scratchpad commands. The other three flags are used in special situations only and remain cleared most of the time. The flags CHLG and AUTH act as a pair in the host/user authentication process to ensure that commands are executed in a certain sequence. If the sequence is correct and the subsequent Match Scratchpad command results in matching data, the MATCH flag is set. The MATCH flag then may affect Validate Data Page, Sign Data Page or Read Authenticated Page.

Device Flag Summary Table 3

Command, Function or Condition	HIDE	CHLG	AUTH	MATCH
Return to Probe Condition	SET	-----	-----	-----
Read Memory Command	-----	CLEARED	CLEARED	-----
Match Scratchpad Command	-----	CLEARED	CLEARED	Note 1)
Write Scratchpad Command	-----	CLEARED	CLEARED	-----
Read Scratchpad Command	-----	-----	-----	-----
Erase Scratchpad Command	CLEARED	CLEARED	CLEARED	-----
Copy Scratchpad Command	-----	CLEARED	CLEARED	-----
Read Authenticated Page Command	-----	CLEARED	CLEARED	-----
Validate Data Page Function	SET	CLEARED	CLEARED	-----
Sign Data Page Function	-----	CLEARED	CLEARED	-----
Compute Challenge Function	-----	SET	CLEARED	CLEARED
Authenticate Host Function	SET	CLEARED	Note 2)	CLEARED
Compute first Secret Function	SET	CLEARED	CLEARED	CLEARED
Compute next Secret Function	SET	CLEARED	CLEARED	CLEARED

1) The flag is SET if the data matches and the AUTH flag was set before command execution; otherwise the flag is CLEARED. Setting the MATCH flag requires the successful execution of Compute Challenge, Authenticate Host and Match Scratchpad in an uninterrupted sequence.
2) Is SET only if CHLG flag was set before command execution; otherwise is cleared.

SHA-1 COMPUTATION ALGORITHM

This description of the SHA computation is adapted from the Secure Hash Standard SHA-1 document referenced on page 2 of this data sheet. The algorithm takes as its input data 16, 32-bit words M_t ($0 \le t \le 15$), as shown in Table 2, SHA-1 Input Message Formats. The SHA computation involves a sequence of eighty 32-bit words called W_t ($0 \le t \le 79$), a sequence of eighty 32-bit words called K_t ($0 \le t \le 79$), a Boolean function f_t (B, C, D) ($0 \le t \le 79$) with B, C and D being 32-bit words, and three more 32-bit words called A, E and TMP. The operations required for the SHA computation are arithmetic addition without carry ("+"), logical inversion or 1's complement ("\"), EXCLUSIVE OR ("⊕"), logical AND ("∧"), logical OR ("∨"), assignment (":="), and circular shifting within a 32-bit word. The expression "$S^n(X)$" represents a circular shift of X by n positions to the left, with X being a 32-bit word.

The function f_t is defined as follows:

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

The sequence W_t ($0 \le t \le 79$) is defined as follows:

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

The sequence K_t ($0 \le t \le 79$) is defined as follows:

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

The variables A, B, C, D, E are initialized as follows:

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

The 160-bit MAC is the concatenation of A, B, C, D, and E after looping through the following set of computations for t = 0 to 79 (discarding any carry-out):

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

The Message Authentication Code is loaded into the scratchpad of the DS1963S in two different ways, depending on the selected SHA function. With Compute First Secret and Compute Next Secret 64 bits of the MAC are used in a repeating pattern in order to allow it to be copied to any of the eight secrets. With all other SHA functions the full 160-bit result is loaded into the scratchpad. Table 4 shows the placement of bytes in the scratchpad.

SHA-1 Output Message Formats Table 4

Partial Code (Compute First Secret and Compute Next Secret only)

(SP+0) := E[7:0]	(SP+1) := E[15:8]	(SP+2) := E[23:16]	(SP+3) := E[31:24]
(SP+4) := D[7:0]	(SP+5) := D[15:8]	(SP+6) := D[23:16]	(SP+7) := D[31:24]
(SP+8) := E[7:0]	(SP+9) := E[15:8]	(SP+10) := E[23:16]	(SP+11) := E[31:24]
(SP+12) := D[7:0]	(SP+13) := D[15:8]	(SP+14) := D[23:16]	(SP+15) := D[31:24]
(SP+16) := E[7:0]	(SP+17) := E[15:8]	(SP+18) := E[23:16]	(SP+19) := E[31:24]
(SP+20) := D[7:0]	(SP+21) := D[15:8]	(SP+22) := D[23:16]	(SP+23) := D[31:24]
(SP+24) := E[7:0]	(SP+25) := E[15:8]	(SP+26) := E[23:16]	(SP+27) := E[31:24]
(SP+28) := D[7:0]	(SP+29) := D[15:8]	(SP+30) := D[23:16]	(SP+31) := D[31:24]

Full 160-Bit Code (all other SHA functions)

(SP+8) := E[7:0]	(SP+9) := E[15:8]	(SP+10) := E[23:16]	(SP+11) := E[31:24]
(SP+12) := D[7:0]	(SP+13) := D[15:8]	(SP+14) := D[23:16]	(SP+15) := D[31:24]
(SP+16) := C[7:0]	(SP+17) := C[15:8]	(SP+18) := C[23:16]	(SP+19) := C[31:24]
(SP+20) := B[7:0]	(SP+21) := B[15:8]	(SP+22) := B[23:16]	(SP+23) := B[31:24]
(SP+24) := A[7:0]	(SP+25) := A[15:8]	(SP+26) := A[23:16]	(SP+27) := A[31:24]

1-WIRE BUS SYSTEM

The 1-Wire bus is a system that has a single bus master and one or more slaves. In all instances the DS1963S is a slave device. The bus master is typically a microcontroller or PC. For small configurations the 1-Wire communication signals can be generated under software control using a single port pin. For multisensor networks, the DS2480B 1-Wire line driver chip or serial port adapters based on this chip (DS9097U series) are recommended. This simplifies the hardware design and frees the microprocessor from responding in real-time.

The discussion of this bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal types and timing). The 1-Wire protocol defines bus transactions in terms of the bus state during specific time slots that are initiated on the falling edge of sync pulses from the bus master. For a more detailed protocol description, refer to Chapter 4 of the *Book of DS19xx iButton Standards*.

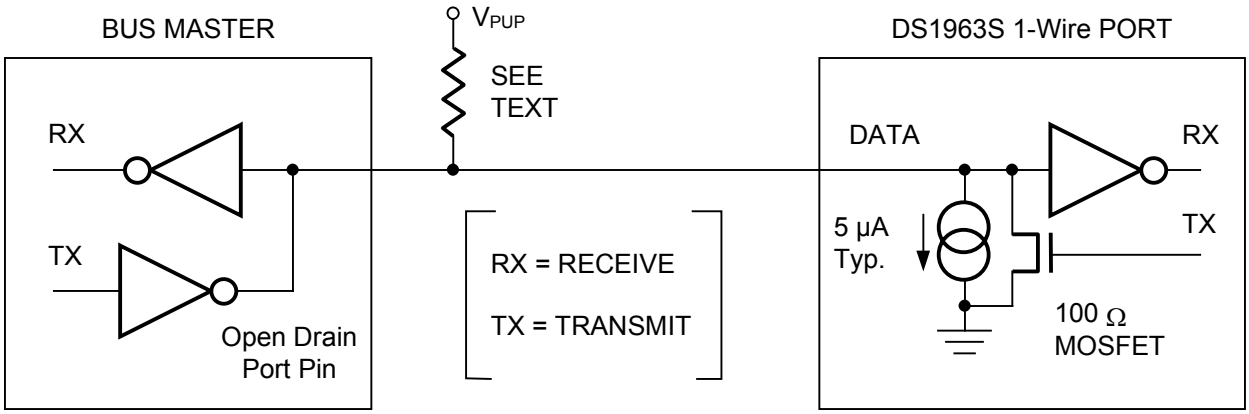
HARDWARE CONFIGURATION

The 1-Wire bus has only a single line by definition; it is important that each device on the bus be able to drive it at the appropriate time. To facilitate this, each device attached to the 1-Wire bus must have open drain or tri-state outputs. The 1-Wire port of the DS1963S is open-drain with an internal circuit equivalent to that shown in Figure 9.

A multidrop bus consists of a 1-Wire bus with multiple slaves attached. At standard speed the 1-Wire bus has a maximum data rate of 16.3kbits per second. The speed can be boosted to 142kbits per second by activating the Overdrive mode. The DS1963S is not guaranteed to be fully compliant to the iButton Standard. Its maximum data rate in standard speed mode is 15.4kbits per second and 125kbits per second in Overdrive. The value of the pullup resistor primarily depends on the network size and load conditions. For most applications the maximum value of 2.2kΩ will be adequate.

The idle state for the 1-Wire bus is high. If for any reason a transaction needs to be suspended, the bus **must** be left in the idle state if the transaction is to resume. If this does not occur and the bus is left low for more than 16µs (Overdrive speed) or more than 120µs (standard speed), one or more devices on the bus may be reset. With the DS1963S the bus must be left low for no longer than 15.4µs at Overdrive speed to ensure that none of the slave devices on the 1-Wire bus performs a reset. Despite of its limited compliance, the DS1963S will communicate properly when used in conjunction with a DS2480B 1-Wire driver and serial port adapters that are based on this driver chip.

HARDWARE CONFIGURATION Figure 9



TRANSACTION SEQUENCE

The protocol for accessing the DS1963S via the 1-Wire port is as follows:

- Initialization
- ROM Function Command
- Memory or SHA Function Command
- Transaction/Data

INITIALIZATION

All transactions on the 1-Wire bus begin with an initialization sequence. The initialization sequence consists of a reset pulse transmitted by the bus master followed by presence pulse(s) transmitted by the slave(s). The presence pulse lets the bus master know that the DS1963S is on the bus and is ready to operate. For more details, see the "1-Wire Signaling" section.

ROM FUNCTION COMMANDS

Once the bus master has detected a presence, it can issue one of the seven ROM function commands that the DS1963S supports. All ROM function commands are 8 bits long. A list of these commands follows (refer to flowchart in Figure 10):

Read ROM [33h]

This command allows the bus master to read the DS1963S's 8-bit family code, unique 48-bit serial number, and 8-bit CRC. This command should only be used if there is a single slave on the bus. If more than one slave is present on the bus, a data collision will occur when all slaves try to transmit at the same time (open drain will produce a wired-AND result). The resultant family code and 48-bit serial number read by the master will be invalid.

Match ROM [55h]

The match ROM command, followed by a 64-bit ROM sequence, allows the bus master to address a specific DS1963S on a multidrop bus. Only the DS1963S that exactly matches the 64-bit ROM sequence will respond to the following memory function command. All slaves that do not match the 64-bit ROM sequence will wait for a reset pulse. This command can be used with a single or multiple devices on the bus.

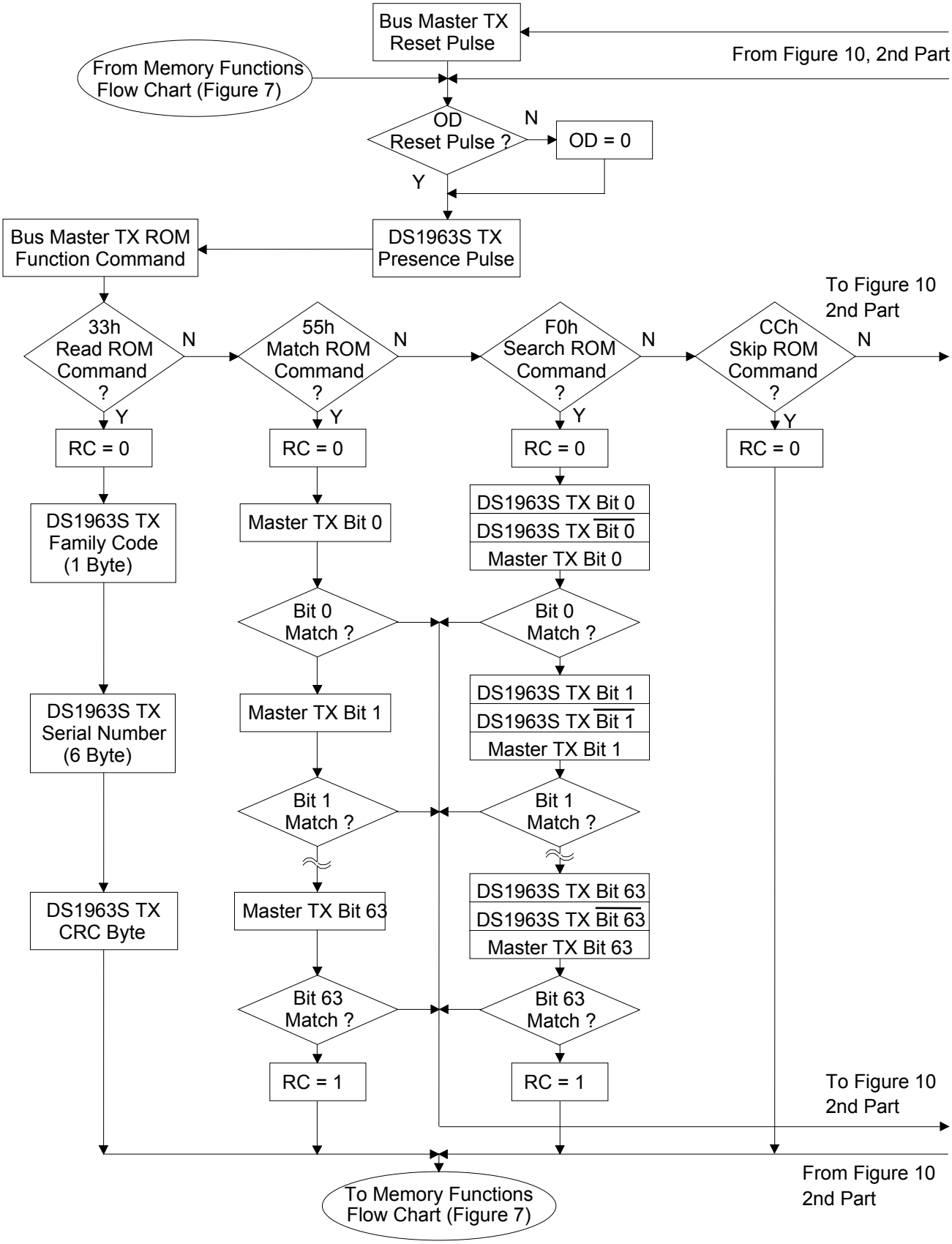
Search ROM [F0h]

When a system is initially brought up, the bus master might not know the number of devices on the 1-Wire bus or their 64-bit ROM codes. The search ROM command allows the bus master to use a process of elimination to identify the 64-bit ROM codes of all slave devices on the bus. The search ROM process is the repetition of a simple 3-step routine: read a bit, read the complement of the bit, then write the desired value of that bit. The bus master performs this 3-step routine on each bit of the ROM. After one complete pass, the bus master knows the 64-bit ROM code of one device. Additional passes will identify the ROM codes of the remaining devices. See Chapter 5 of the *Book of DS19xx iButton Standards* for a comprehensive discussion of a search ROM, including an actual example.

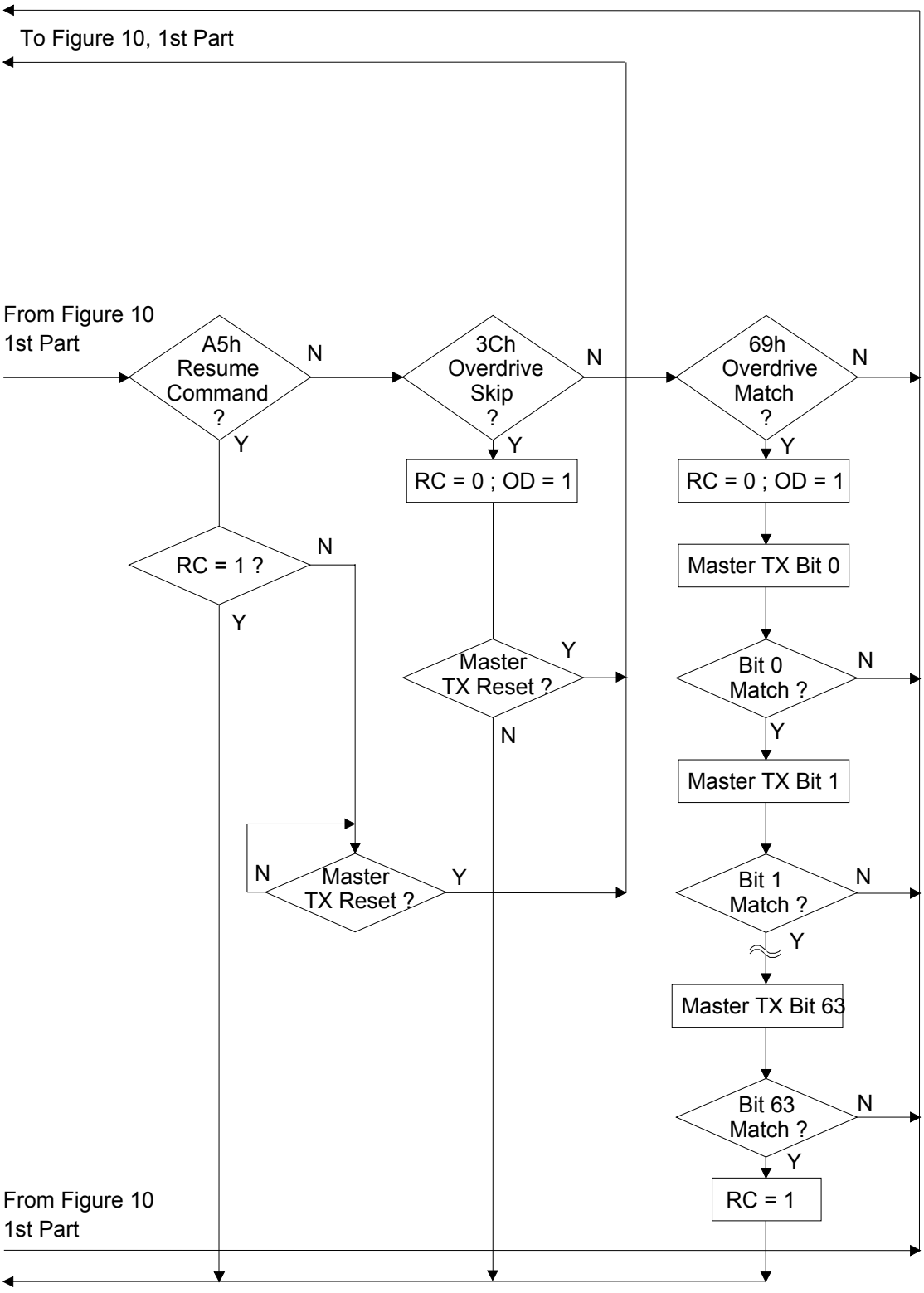
Skip ROM [CCh]

This command can save time in a single drop bus system by allowing the bus master to access the memory and SHA functions without providing the 64-bit ROM code. If more than one slave is present on the bus and, for example, a read command is issued following the Skip ROM command, data collision will occur on the bus as multiple slaves transmit simultaneously (open drain pulldowns will produce a wired-AND result).

ROM FUNCTIONS FLOW CHART Figure 10



ROM FUNCTIONS FLOW CHART (continued) Figure 10



Overdrive Skip ROM [3Ch]

On a single-drop bus this command can save time by allowing the bus master to access the memory functions without providing the 64-bit ROM code. Unlike the normal Skip ROM command the Overdrive Skip ROM sets the DS1963S in the Overdrive Mode (OD = 1). All communication following this command code has to occur at Overdrive Speed until a reset pulse of minimum 480 µs duration resets all devices on the bus to regular speed (OD = 0).

When issued on a multidrop bus this command will set all Overdrive-supporting devices into Overdrive mode. To subsequently address a specific Overdrive-supporting device, a reset pulse at Overdrive speed has to be issued followed by a Match ROM or Search ROM command sequence. This will speed up the search process. If more than one Overdrive-supporting slave is present on the bus and the Overdrive Skip ROM command is followed by a read command, data collision will occur on the bus as multiple slaves transmit simultaneously (open drain pulldowns will produce a wire-AND result).

Overdrive Match ROM [69h]

The Overdrive Match ROM command, followed by a 64-bit ROM sequence transmitted at Overdrive Speed, allows the bus master to address a specific DS1963S on a multidrop bus and to simultaneously set it in Overdrive Mode. Only the DS1963S that exactly matches the 64-bit ROM sequence will respond to the subsequent memory or SHA function command. Slaves already in Overdrive mode from a previous Overdrive Skip or a successful Overdrive Match command will remain in Overdrive mode. All Overdrive-capable slaves will return to regular speed at the next Reset Pulse of minimum 480 µs duration. The Overdrive Match ROM command can be used with a single or multiple devices on the bus.

Resume Command [A5h]

In a typical application the DS1963S needs to be accessed several times to complete a monetary transaction. The number of accesses increases further if host/user-Authentication is also performed. In a multidrop environment this means that the 64-bit ROM sequence of a Match ROM command has to be repeated for every access. To maximize the data throughput in a multidrop environment the Resume Command function was implemented. This function checks the status of the RC bit and, if it is set, directly transfers control to the Memory and SHA functions, similar to a Skip ROM command. The only way to set the RC bit is through successfully executing the Match ROM, Search ROM or Overdrive Match ROM command. Once the RC bit is set, the device can repeatedly be accessed through the Resume Command function. Accessing another device on the bus will clear the RC bit, preventing two or more devices from simultaneously responding to the Resume Command function.

1-WIRE SIGNALING

The DS1963S requires strict protocols to ensure data integrity. The protocol consists of four types of signaling on one line: Reset Sequence with Reset Pulse and Presence Pulse, Write 0, Write 1, and Read Data. Except for the presence pulse the bus master initiates all these signals. The DS1963S can communicate at two different speeds: standard speed and Overdrive speed. If not explicitly set into the Overdrive mode, the DS1963S will communicate at standard speed. While in Overdrive Mode the fast timing applies to all waveforms.

To get from idle to active, the voltage on the 1-Wire line needs to fall from V_{PUP} below the threshold V_{TL}. To get from active to idle, the voltage needs to rise from V_{ILMAX} past the threshold V_{TH}. The voltage V_{ILMAX} is relevant for the DS1963S when determining a logical level, but not for triggering any events.

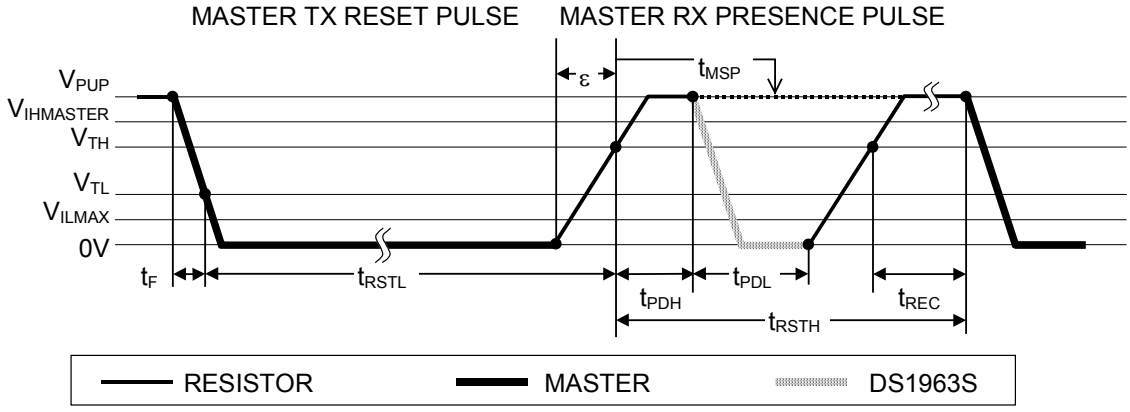
The initialization sequence required to begin any communication with the DS1963S is shown in Figure 11. A Reset Pulse followed by a Presence Pulse indicates the DS1963S is ready to receive data, given the

correct ROM and memory function command. In a mixed population network, the reset low time t_{RSTL} needs to be long enough for the slowest 1-Wire slave device to recognize it as a reset pulse. This duration is 480µs at standard speed and 48µs at Overdrive speed. If the bus master uses slew-rate control on the falling edge, it must pull down the line for $t_{RSTL} + t_F$ to compensate for the edge. A t_{RSTL} duration of 480µs or longer will exit the Overdrive Mode returning the device to standard speed. If the DS1963S is in Overdrive Mode and t_{RSTL} is no longer than 80µs, the device will remain in Overdrive Mode.

After the bus master has released the line it goes into receive mode (RX). Now, the 1-Wire bus is pulled to V_{PUP} via the pullup resistor or, in case of a DS2480B driver, by active circuitry. When the threshold V_{TH} is crossed, the DS1963S waits for t_{PDH} and then transmits a Presence Pulse by pulling the line low for t_{PDL}. To detect a presence pulse, the master must test the logical state of the 1-Wire line at t_{MSP}.

The t_{RSTH} window must be at least the sum of t_{PDHMAX}, t_{PDLMAX}, and t_{RECMIN}. Immediately after t_{RSTH} is expired, the DS1963S is ready for data communication. In a mixed population network, t_{RSTH} should be extended to minimum 480µs at standard speed and 48µs at Overdrive speed to accommodate other 1-Wire devices.

INITIALIZATION PROCEDURE (RESET AND PRESENCE PULSES) Figure 11



Read/Write Time Slots

Data communication with the DS1963S takes place in time slots that carry a single bit each. Write time slots transport data from bus master to slave. Read time-slots transfer data from slave to master. The definitions of the write and read time slots are illustrated in Figure 12.

All communication begins with the master pulling the data line low. As the voltage on the 1-Wire line falls below the threshold V_{TL}, the DS1963S starts its internal timing generator that determines when the data line will be sampled during a write time slot and how long data will be valid during a read time slot.

Master to Slave

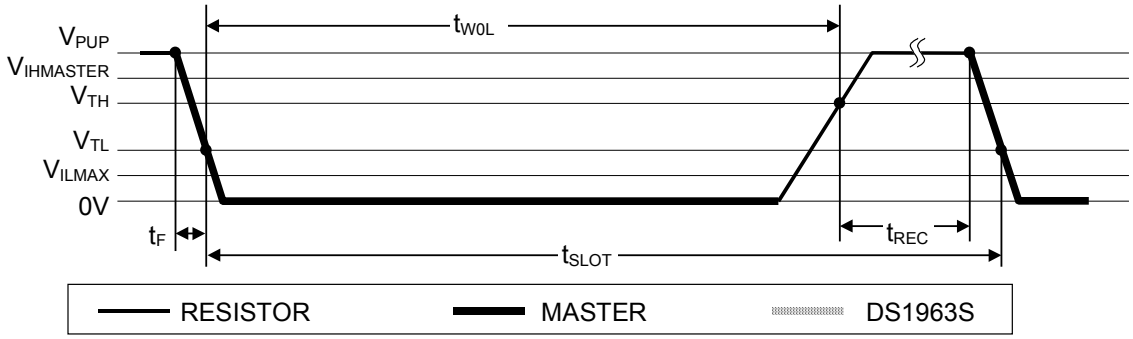
For a **write-one** time slot, the voltage on the data line must have crossed the V_{THMAX} threshold after the write-one low time t_{W1LMAX} is expired. For a **write-zero** time slot, the voltage on the data line must stay below the V_{THMIN} threshold until the write-zero low time t_{W0LMIN} is expired. For most reliable communication the voltage on the data line should not exceed V_{ILMAX} during the entire t_{W0L} window. After the V_{THMAX} threshold has been crossed, the DS1963S needs a recovery time t_{REC} before it is ready for the next time slot.

READ/WRITE TIMING DIAGRAM Figure 12

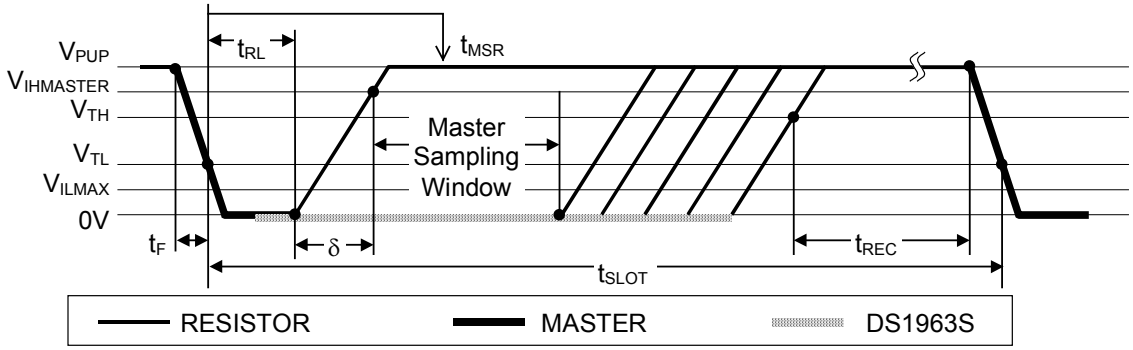
Write-One Time Slot



Write-Zero Time Slot



Read-Data Time Slot



Slave to Master

A **read-data** time slot begins like a write-one time slot. The voltage on the data line must remain below V_{TLMIN} until the read low time t_{RL} is expired. During the t_{RL} window, when responding with a 0, the DS1963S will start pulling the data line low; its internal timing generator determines when this pull-down ends and the voltage starts rising again. When responding with a 1, the DS1963S will not hold the data line low at all, and the voltage starts rising as soon as t_{RL} is over.

The sum of t_{RL} + δ (rise rime) on one side and the internal timing generator of the DS1963S on the other side define the master sampling window (t_{MSRMIN} to t_{MSRMAX}) in which the master must perform a read from the data line. For most reliable communication, t_{RL} should be as short as permissible and the master should read close to but no later than t_{MSRMAX}. After reading from the data line, the master must wait until t_{SLOT} is expired. This guarantees sufficient recovery time t_{REC} for the DS1963S to get ready for the next time slot.

CRC GENERATION

With the DS1963S there are two different types of CRCs (Cyclic Redundancy Checks). One CRC is an 8-bit type. It is computed at the factory and lasered into the most significant byte of the 64-bit ROM. The equivalent polynomial function of this CRC is $X^8 + X^5 + X^4 + 1$. To determine whether the ROM data has been read without error the bus master can compute the CRC value from the first 56 bits of the 64-bit ROM and compare it to the value read from the DS1963S. This 8-bit CRC is received in the true form (non-inverted) when reading the ROM.

The other CRC is a 16-bit type, generated according to the standardized CRC16-polynomial function $X^{16} + X^{15} + X^2 + 1$. This CRC is used for error detection with the Read Authenticated Page command, Compute SHA, when reading the scratchpad and for fast verification of a data transfer when writing to the scratchpad. It is the same type of CRC as is used with NV RAM based iButtons for error detection within the iButton Extended File Structure. In contrast to the 8-bit CRC, the 16-bit CRC is always returned or sent in the complemented (inverted) form. A CRC-generator inside the DS1963S chip (Figure 13) will calculate a new 16-bit CRC as shown in the command flow chart of Figure 7. The bus master may compare the CRC value read from the device to the one it calculates from the data and decides whether to continue with an operation or to re-read the portion of the data with the CRC error.

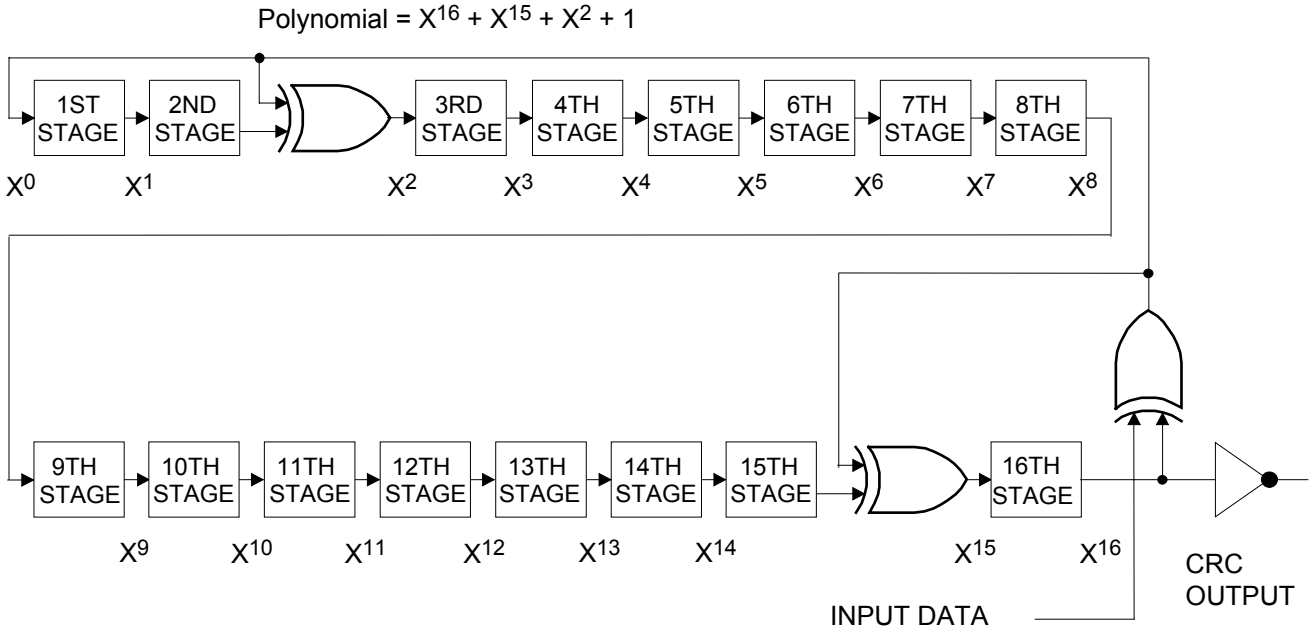
With the Write Scratchpad command the CRC is generated by first clearing the CRC generator and then shifting in the command code, the Target Addresses TA1 and TA2 and all the data bytes. The DS1963S will transmit this CRC only if the data bytes written to the scratchpad include scratchpad ending offset 11111b. The data may start at any location within the scratchpad. This algorithm applies regardless of the state of the HIDE-flag. However, if the HIDE-flag is set the data bytes that follow the target address are used for the CRC-calculation only. They are not received in the scratchpad.

With the Read Scratchpad command the CRC is generated by first clearing the CRC generator and then shifting in the command code, the Target Addresses TA1 and TA2, the E/S byte, and the scratchpad data starting at the scratchpad offset. The DS1963S will transmit this CRC only if the reading continues through the end of the scratchpad, regardless of the actual ending offset. If the HIDE-flag is set the CRC-calculation uses FFh-bytes instead of the scratchpad data, which remains hidden.

With the Read Authenticated Page command the 16-bit CRC value is the result of shifting the command byte into the cleared CRC generator, followed by the two address bytes, the data bytes, and the values of the write-cycle counters of the addressed memory page and its associated secret. The write cycle counters are shifted in with their least significant byte first. With the Compute SHA command the CRC results from shifting the command byte into the cleared CRC generator, followed by the Target Addresses TA1 and TA2 and the SHA Control byte.

For more details on generating CRC values including example implementations in both hardware and software, see the *Book of DS19xx iButton Standards*.

CRC-16 HARDWARE DESCRIPTION AND POLYNOMIAL Figure 13



PHYSICAL SPECIFICATION

Size	See mechanical drawing
Weight	3.3 grams
Expected Service Life	See graphic
Safety	Meets UL#913 (4th Edit.); Intrinsically Safe Apparatus, Approval under Entity Concept for use in Class I, Division 1, Group A, B, C and D Locations (application pending)

ABSOLUTE MAXIMUM RATINGS*

IO Voltage to GND	-0.5V, +6V
IO sink current	20mA
Temperature Range	-40°C to +85°C
Junction Temperature	+150°C
Storage Temperature Range	-25°C to +50°C

* 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. These devices must not be exposed to temperatures over +70°C for extended time periods.

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

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
IO Pin General Data							
1-Wire Pullup Resistance	R_{PUP}				2.2	kΩ	1, 2
Input Capacitance	C_{IO}			100	800	pF	3
Input Load Current	I_L	IO pin at V_{PUP}			9	µA	4
High-to-Low Switching Threshold	V_{TL}		0.7		2.9	V	5, 6, 7
Input Low Voltage	V_{IL}				0.30	V	1, 5, 8
Low-to-High Switching Threshold	V_{TH}		0.6		2.9	V	5, 6, 9
Output Low Voltage at 4mA	V_{OL}				0.4	V	5, 10
Fall Time	t_F				5	µs	1
Recovery Time	t_{REC}	Standard Speed, R_{PUP}=2.2kΩ	5			µs	1, 15
		Overdrive Speed, R_{PUP}=2.2kΩ	2				
		Overdrive Speed, Directly Prior to Reset Pulse; R_{PUP}=2.2kΩ	5				

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
Timeslot Duration	t_{SLOT}	Standard Speed	69			µs	1, 15
		Standard Speed, -20°C to +85°C	65				
		Overdrive Speed, V_{PUP} > 4.5V	8				
IO Pin, 1-Wire Reset, Presence Detect Cycle							
Reset Low Time	t_{RSTL}	Standard Speed	540		960	µs	1, 14
		Standard Speed, -20°C to +85°C	480		960		
		Overdrive Speed, V_{PUP} > 4.5V	48		80		
Presence Detect High Time	t_{PDH}	Standard Speed	17		60	µs	14
		Overdrive Speed, V_{PUP} > 4.5V	1.8		6		
Presence Detect Low Time	t_{PDL}	Standard Speed	78		260	µs	14
		Standard Speed, -20°C to +85°C	78		240		
		Overdrive Speed, V_{PUP} > 4.5V	7.7		24		
Presence Detect Sample Time	t_{MSP}	Standard Speed	60		95	µs	1
		Overdrive Speed, V_{PUP} > 4.5V	6		9.5		
IO Pin, 1-Wire Write							
Write-0 Low Time	t_{W0L}	Standard Speed	64		120	µs	1, 14
		Standard Speed, -20°C to +85°C	60		120		
		Overdrive Speed, V_{PUP} > 4.5V	6		15.4		
Write-1 Low Time	t_{W1L}	Standard Speed	5		15 - ε	µs	1, 11
		Overdrive Speed	1		2 - ε		
IO Pin, 1-Wire Read							
Read Low Time	t_{RL}	Standard Speed	5		15 - δ	µs	1, 12
		Overdrive Speed	1		2 - δ		
Read Sample Time	t_{MSR}	Standard Speed	t_{RL} + δ		15	µs	1, 12
		Overdrive Speed	t_{RL} + δ		2		
SHA-1 Engine							
Computation Time	t_{SHA}			0.4	1.15	ms	
Number of SHA-1 Computations	N_{SHA}		(see graphs)			---	13

NOTES

1) System requirement.
2) Maximum allowable pullup resistance is a function of the number of 1-Wire devices in the system and 1-Wire recovery times. The specified value here applies to systems with only one device and with the minimum 1-Wire recovery times. For more heavily loaded systems, an active pullup such as that found in the DS2480 may be required.
3) Capacitance on the data pin could be 800pF when power is first applied. If a 2.2kΩ resistor is used to pull up the data line to $V_{PUP;}$ the parasite capacitance will not affect normal communications 2.5µs after power has been applied.
4) Input load is to ground.
5) All voltages are referenced to ground.
6) V_{TL}, V_{TH} are a function of the internal supply voltage.
7) Voltage below which, during a falling edge on IO, a logic 0 is detected.
8) The voltage on IO needs to be less or equal to V_{ILMAX} whenever the master drives the line low.
9) Voltage above which, during a rising edge on IO, a logic 1 is detected.
10) The I-V characteristic is linear for voltages less than 1V.
11) ε is the time required for the pullup circuitry to pull the voltage on IO up from V_{IL} to V_{TH}.
12) δ represents the time required for the pullup circuitry to pull the voltage on IO up from V_{IL} to the input high threshold of the bus master.
13) The number of SHA-1 computations possible with the built-in energy source depends on the operating and storage temperature of the device.
14) Highlighted numbers are **not** in compliance with the published iButton standards. See comparison tables below.
15) The recovery time was intentionally increased from the standard value of 1µs to a longer value to improve the parasitic power supply of the device. This change improves the performance of the chip and is not considered a non-compliance to the published standard.

Non-Compliance Table for T_A = -40°C to +85°C

Parameter Name	Standard Values				DS1963S Values			
	Standard Speed		Overdrive Speed		Standard Speed		Overdrive Speed	
	min	max	min	max	min	max	min	max
t_{SLOT} (incl. t_{REC})	61µs	(undef.)	7µs	(undef.)	69µs	(undef.)	8µs	(undef.)
t_{RSTL}	480µs	(undef.)	48µs	80µs	540µs	960µs	48µs	80µs
t_{PDH}	15µs	60µs	2µs	6µs	17µs	60µs	1.8µs	6µs
t_{PDL}	60µs	240µs	8µs	24µs	78µs	260µs	7.7µs	24µs
t_{W0L}	60µs	120µs	6µs	16µs	64µs	120µs	6µs	15.4µs
t_{SLS}, t_{SPD}	15µs	60µs	2µs	6µs	19µs	64µs	2µs	4.8µs

Non-Compliance Table for T_A = -20°C to +85°C

Parameter Name	Standard Values				DS1963S Values			
	Standard Speed		Overdrive Speed		Standard Speed		Overdrive Speed	
	min	max	min	max	min	max	min	max
t_{SLOT}	61µs	(undef.)	7µs	(undef.)	65µs	(undef.)	8µs	(undef.)
t_{RSTL}	480µs	(undef.)	48µs	80µs	480µs	960µs	48µs	80µs
t_{PDH}	15µs	60µs	2µs	6µs	17µs	60µs	1.8µs	6µs
t_{PDL}	60µs	240µs	8µs	24µs	78µs	240µs	7.7µs	24µs
t_{W0L}	60µs	120µs	6µs	16µs	60µs	120µs	6µs	15.4µs
t_{SLS}, t_{SPD}	15µs	60µs	2µs	6µs	19µs	60µs	2µs	4.8µs

Expected Service Life vs Temperature

