19-4891; Rev 8/09

DS1982
1Kb Add-Only iButton

SPECIAL FEATURES

- 1024 bits Electrically Programmable Read-Only Memory (EPROM) communicates with the economy of one signal plus ground
- EPROM partitioned into four 256-bit pages for randomly accessing packetized data
- Each memory page can be permanently write-protected to prevent tampering
- Device is an “add only” memory where additional data can be programmed into EPROM without disturbing existing data
- Architecture allows software to patch data by superseding an old page in favor of a newly programmed page
- Reduces control, address, data, power, and programming signals to a single data pin
- 8-bit family code specifies DS1982 communications requirements to reader
- Reads over a wide voltage range of 2.8V to 6.0V from -40°C to +85°C; programs at 11.5V to 12.0V from -40°C to +50°C

ORDERING INFORMATION

DS1982-F3+	F3 MicroCan
DS1982-F5+	F5 MicroCan

+Denotes a lead(Pb)-free/RoHS-compliant product.

EXAMPLES OF ACCESSORIES

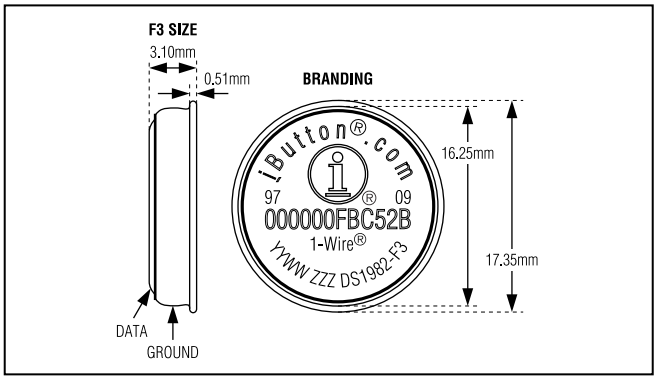
DS9096P	Self-Stick Adhesive Pad
DS9101	Multi-Purpose Clip
DS9093RA	Mounting Lock Ring
DS9093F	Snap-In Fob
DS9092	iButton Probe

COMMON iButton FEATURES

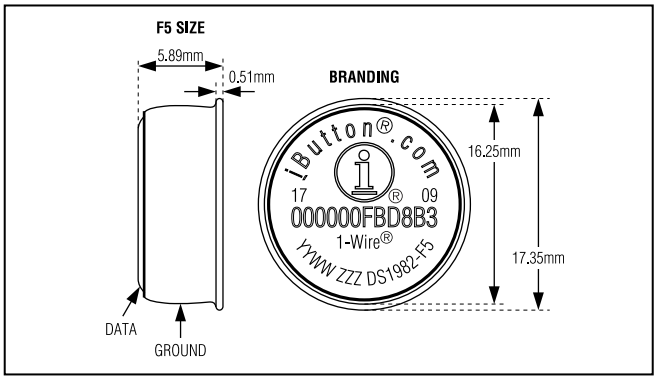
- Unique, factory-lasered and tested 64-bit registration number (8-bit family code + 48-bit serial number + 8-bit CRC tester) assures absolute traceability because no two parts are alike
- Multidrop controller for MicroLAN
- Digital identification and information by momentary contact
- Chip-based data carrier compactly stores information
- Data can be accessed while affixed to object
- Economically communicates to bus master with a single digital signal at 16.3kbps
- Standard 16mm diameter and 1-Wire® protocol ensure compatibility with iButton® 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

iButton and 1-Wire are registered trademarks of Maxim Integrated Products, Inc.

F3 MicroCan



F5 MicroCan



iButton DESCRIPTION

The DS1982 1Kb Add-Only iButton is a rugged read/write data carrier that identifies and stores relevant information about the product or person to which it is attached. This information can be accessed with minimal hardware, for example, a single port pin of a microcontroller. The DS1982 consists of a factory-lasered registration number that includes a unique 48-bit serial number, an 8-bit CRC, and an 8-bit Family Code (09h) plus 1Kb of EPROM that is user-programmable. The power to program and read the DS1982 is derived entirely from the 1-Wire communication line. Data is transferred serially via the 1-Wire protocol that requires only a single data lead and a ground return. The entire device can be programmed and then write-protected if desired. Alternatively, the part may be programmed multiple times with new data being appended to, but not overwriting, existing data with each subsequent programming of the device. Note: Individual bits can be changed only from a logical 1 to a logical 0, never from a logical 0 to a logical 1. A provision is also included for indicating that a certain page or pages of data are no longer valid and have been replaced with new or updated data that is now residing at an alternate page address. This page address redirection allows software to patch data and enhance the flexibility of the device as a standalone database. The 48-bit serial number that is factory-lasered into each DS1982 provides a guaranteed unique identity that allows for absolute traceability. The durable MicroCan package is highly resistant to harsh environments such as dirt, moisture, and shock. Its compact button-shaped profile is self-aligning with cup-shaped receptacles, allowing the DS1982 to be used easily by human operators or automatic equipment. Accessories permit the DS1982 to be mounted on printed circuit boards, plastic key fobs, photo-ID badges, ID bracelets, and many other objects. Applications include work-in-progress tracking, electronic travelers, access control, storage of calibration constants, and debit tokens.

OVERVIEW

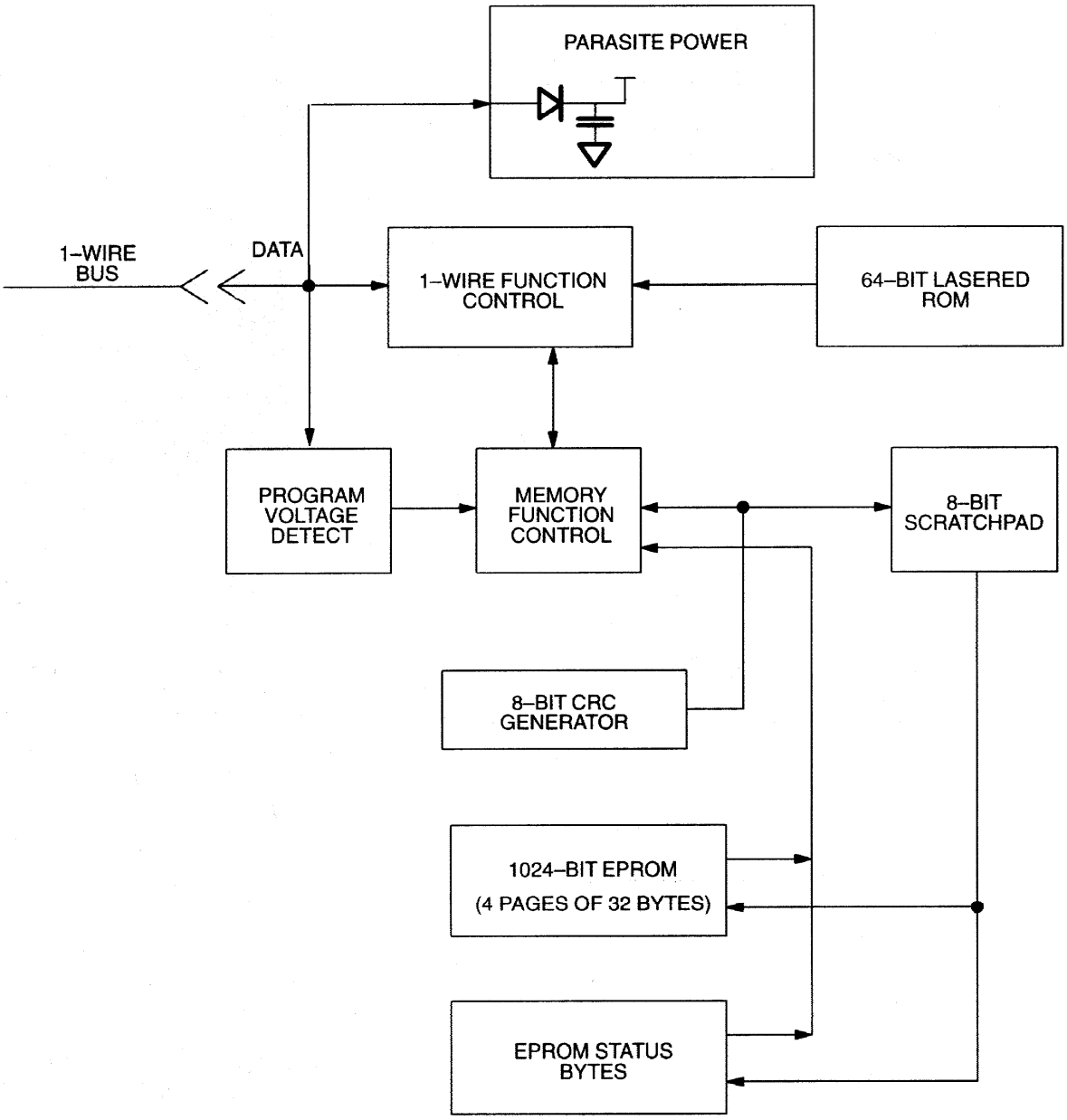
The block diagram in Figure 1 shows the relationships between the major control and memory sections of the DS1982. The DS1982 has three main data components: 1) 64-bit lasered ROM, 2) 1024-bit EPROM, and 3) EPROM Status Bytes. The device derives its power for read operations entirely from the 1-Wire communication line by storing energy on an internal capacitor during periods of time when the signal line is high and continues to operate off of this “parasite” power source during the low times of the 1-Wire line until it returns high to replenish the parasite (capacitor) supply. During programming, 1-Wire communication occurs at normal voltage levels and then is pulsed momentarily to the programming voltage to cause the selected EPROM bits to be programmed. The 1-Wire line must be able to provide 12 volts and 10 milliamperes to adequately program the EPROM portions of the part. Whenever programming voltages are present on the 1-Wire line a special high voltage detect circuit within the DS1982 generates an internal logic signal to indicate this condition. The hierarchical structure of the 1-Wire protocol is shown in Figure 2. The bus master must first provide one of the four ROM function commands: 1) Read ROM, 2) Match ROM, 3) Search ROM, 4) Skip ROM. These commands operate on the 64-bit lasered ROM portion of each device and can singulate a specific device if many are present on the 1-Wire line as well as indicate to the bus master how many and what types of devices are present. The protocol required for these ROM function commands is described in Figure 9. After a ROM function command is successfully executed, the memory functions that operate on the EPROM portions of the DS1982 become accessible and the bus master may issue any one of the five memory function commands specific to the DS1982 to read or program the various data fields. The protocol for these memory function commands is described in Figure 6. All data is read and written least significant bit first.

64-BIT LASERED ROM

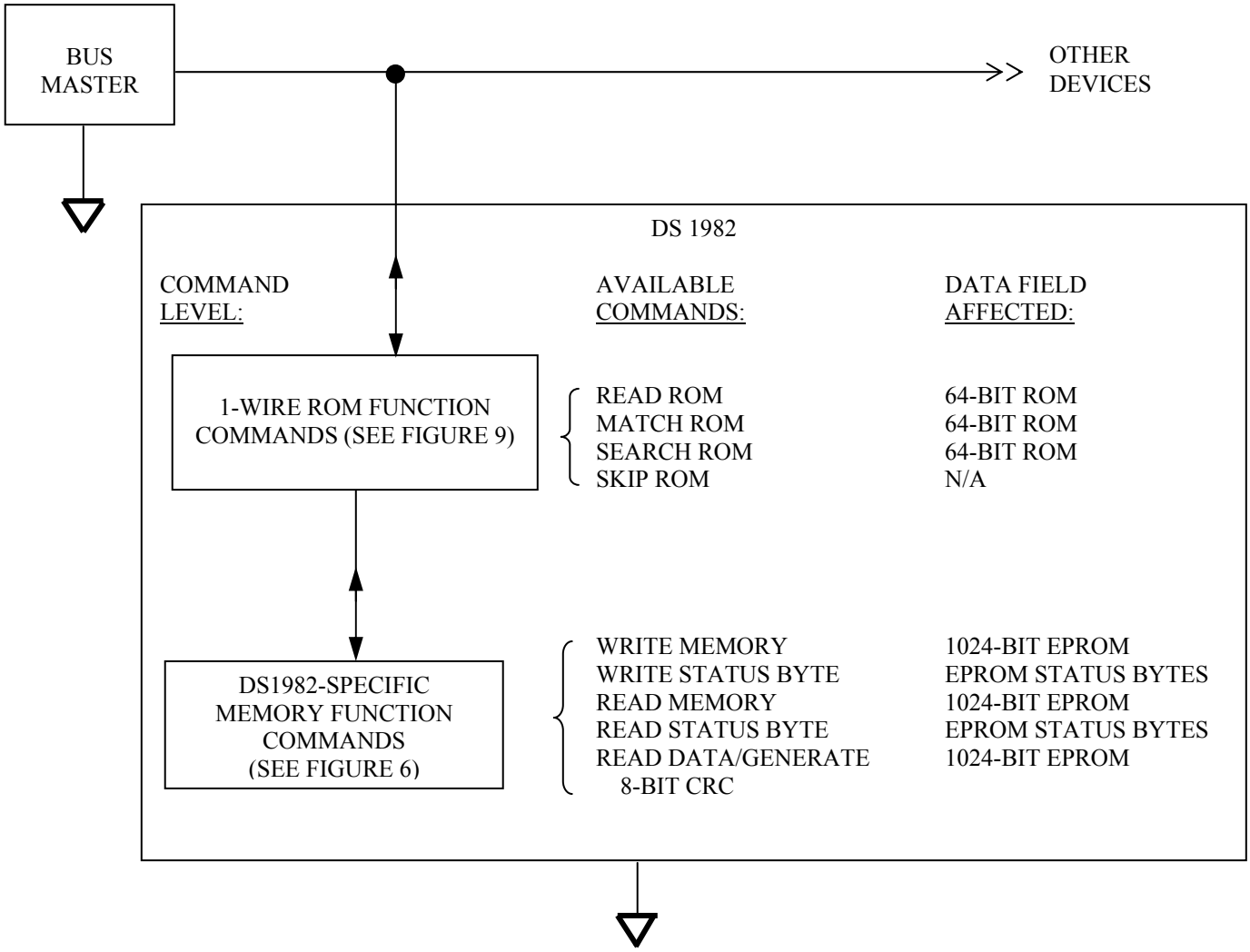
Each DS1982 contains a unique ROM code that is 64 bits long. The first 8 bits are a 1-Wire family code. The next 48 bits are a unique serial number. The last 8 bits are a CRC of the first 56 bits (see Figure 3). The 64-bit ROM and ROM Function Control section allow the DS1982 to operate as a 1-Wire device and follow the 1-Wire protocol detailed in the section "1-Wire Bus System." The memory functions required to read and program the EPROM sections of the DS1982 are not accessible until the ROM function protocol has been satisfied. This protocol is described in the ROM functions flow chart (Figure 9). The 1-Wire bus master must first provide one of four ROM function commands: 1) Read ROM, 2) Match ROM, 3) Search ROM, or 4) Skip ROM. After a ROM function sequence has been successfully executed, the bus master may then provide any one of the memory function commands specific to the DS1982 (Figure 6).

The 1-Wire CRC of the lasered ROM is generated using the polynomial $X^8 + X^5 + X^4 + 1$. Additional information about the Maxim 1-Wire Cyclic Redundancy Check is available in the Book of DS19xx iButton Standards. The shift register acting as the CRC accumulator is initialized to 0. Then starting with the least significant bit of the family code, 1 bit at a time is shifted in. After the 8^{th} bit of the family code has been entered, then the serial number is entered. After the 48^{th} 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 0s.

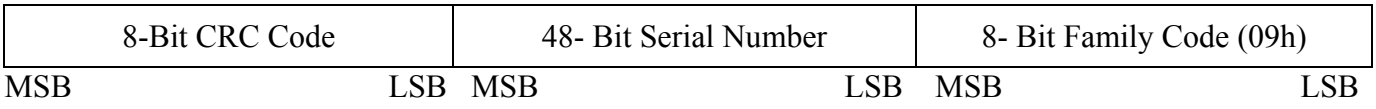
DS1982 BLOCK DIAGRAM Figure 1



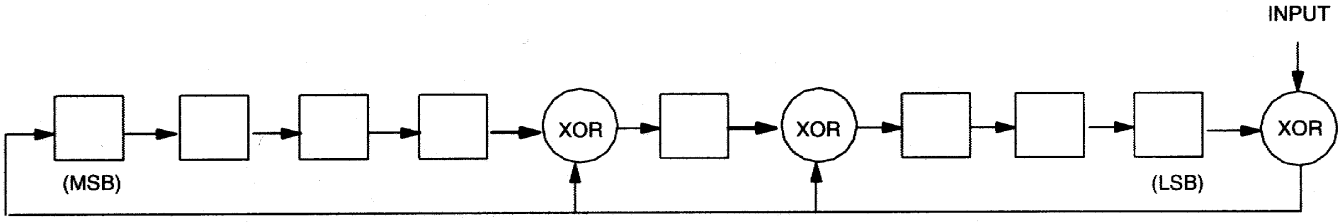
HIERARCHICAL STRUCTURE FOR 1-WIRE PROTOCOL Figure 2



64-BIT LASERED ROM Figure 3



1-WIRE CRC GENERATOR Figure 4



1024-BIT EPROM

The memory map in Figure 5 shows the 1024-bit EPROM section of the DS1982 that is configured as four pages of 32 bytes each. The 8-bit scratchpad is an additional register that acts as a buffer when programming the memory. Data is first written to the scratchpad and then verified by reading an 8-bit CRC from the DS1982 that confirms proper receipt of the data. If the buffer contents are correct, a programming voltage should be applied and the byte of data will be written into the selected address in memory. This process ensures data integrity when programming the memory. The details for reading and programming the 1024-bit EPROM portion of the DS1982 are given in the "Memory Function Commands" section.

EPROM STATUS BYTES

In addition to the 1024 bits of data memory the DS1982 provides 64 bits of status memory accessible with separate commands.

The EPROM Status Bytes can be read or programmed to indicate various conditions to the software interrogating the DS1982. The first byte of the EPROM status memory contains the Write-Protect Page bits that inhibit programming of the corresponding page in the 1024-bit main memory area if the appropriate write protection bit is programmed. Once a bit has been programmed in the Write-Protect Page byte, the entire 32-byte page that corresponds to that bit can no longer be altered but may still be read.

The next 4 bytes of the EPROM Status Memory contain the Page Address Redirection Bytes that indicate if one or more of the pages of data in the 1024-bit EPROM section have been invalidated and redirected to the page address contained in the appropriate redirection byte. The hardware of the DS1982 makes no decisions based on the contents of the Page Address Redirection Bytes. These additional bytes of status EPROM allow for the redirection of an entire page to another page address, indicating that the data in the original page is no longer considered relevant or valid. With EPROM technology, bits within a page can be changed from a logical 1 to a logical 0 by programming, but cannot be changed back. Therefore, it is not possible to simply rewrite a page if the data requires changing or updating, but with space permitting, an entire page of data can be redirected to another page within the DS1982 by writing the one’s complement of the new page address into the Page Address Redirection Byte that corresponds to the original (replaced) page.

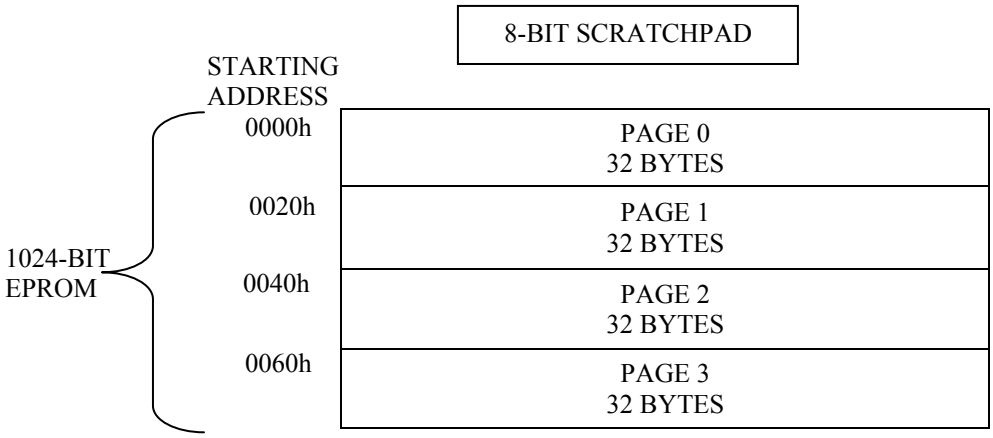
This architecture allows the user’s software to make a “data patch” to the EPROM by indicating that a particular page or pages should be replaced with those indicated in the Page Address Redirection Bytes.

If a Page Address Redirection Byte has a FFH value, the data in the main memory that corresponds to that page is valid. If a Page Address Redirection Byte has some other hex value, the data in the page corresponding to that redirection byte is invalid, and the valid data can now be found at the one’s complement of the page address indicated by the hex value stored in the associated Page Address Redirection Byte. A value of FDH in the redirection byte for page 1, for example, would indicate that the updated data is now in page 2. The details for reading and programming the EPROM status memory portion of the DS1982 are given in the Memory Function Commands section.

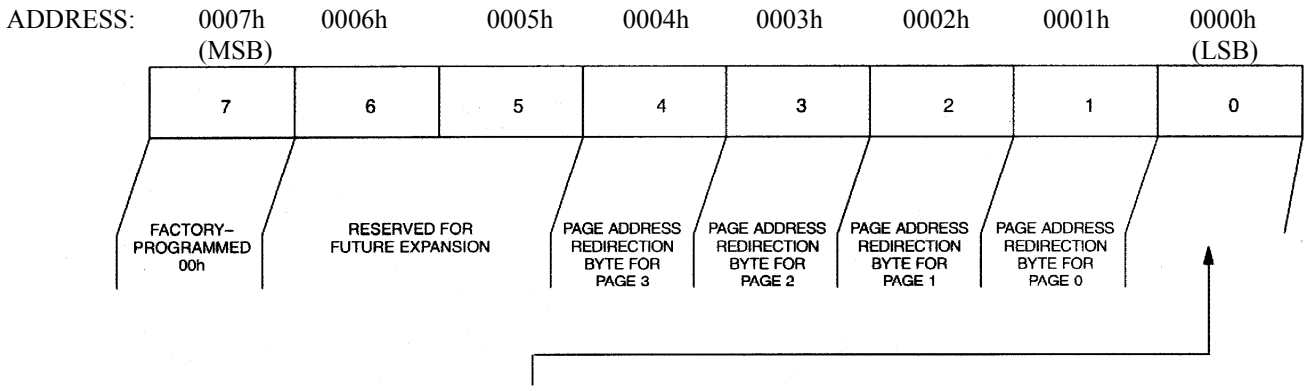
MEMORY FUNCTION COMMANDS

The "Memory Function Flow Chart" (Figure 6) describes the protocols necessary for accessing the various data fields within the DS1982. The Memory Function Control section, 8-bit scratchpad, and the Program Voltage Detect circuit combine to interpret the commands issued by the bus master and create the correct control signals within the device. A 3-byte protocol is issued by the bus master. It is comprised of a command byte to determine the type of operation and 2 address bytes to determine the specific starting byte location within a data field. The command byte indicates if the device is to be read or written. Writing data involves not only issuing the correct command sequence but also providing a 12-volt programming voltage at the appropriate times. To execute a write sequence, a byte of data is first loaded into the scratchpad and then programmed into the selected address. Write sequences always occur a byte at a time. To execute a read sequence, the starting address is issued by the bus master and data is read from the part beginning at that initial location and continuing to the end of the selected data field or until a reset sequence is issued. All bits transferred to the DS1982 and received back by the bus master are sent least significant bit first.

DS1982 MEMORY MAP Figure 5

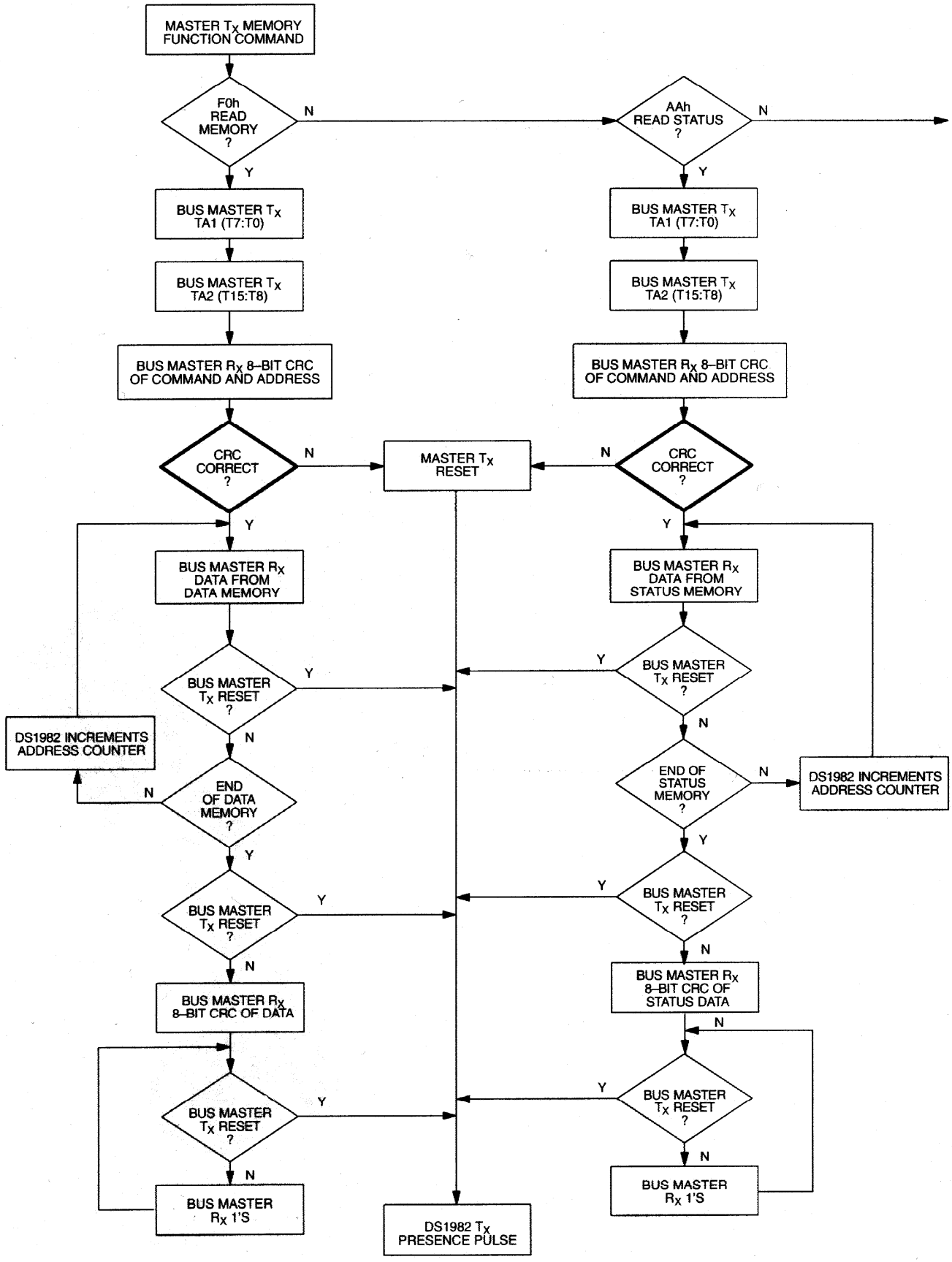


EPROM STATUS BYTES

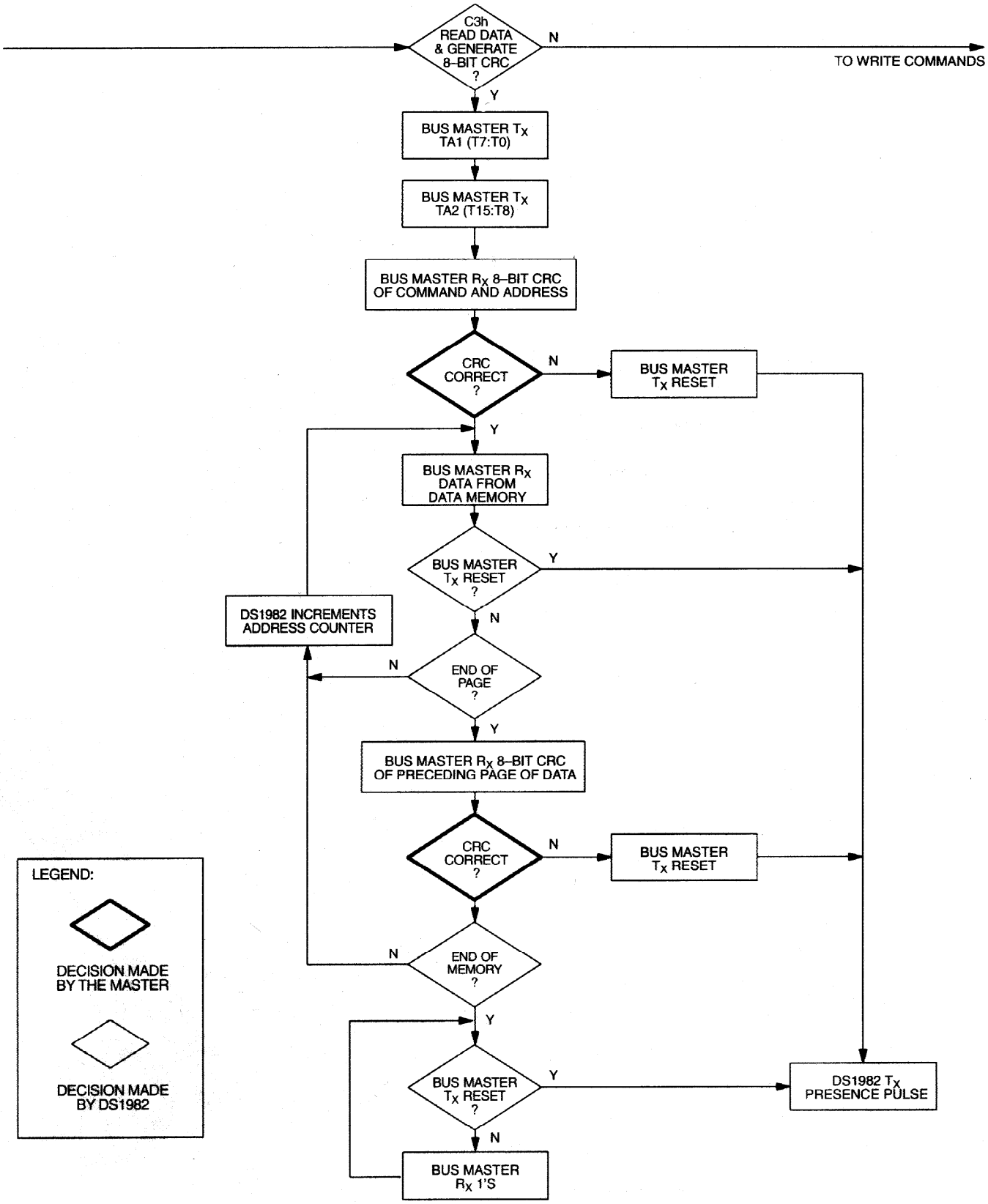


BIT 0	WRITE PROTECT PAGE 0
BIT 1	WRITE PROTECT PAGE 1
BIT 2	WRITE PROTECT PAGE 2
BIT 3	WRITE PROTECT PAGE 3
BIT 4-7	BITMAP OF USED PAGES (RESERVED FOR TMEX)

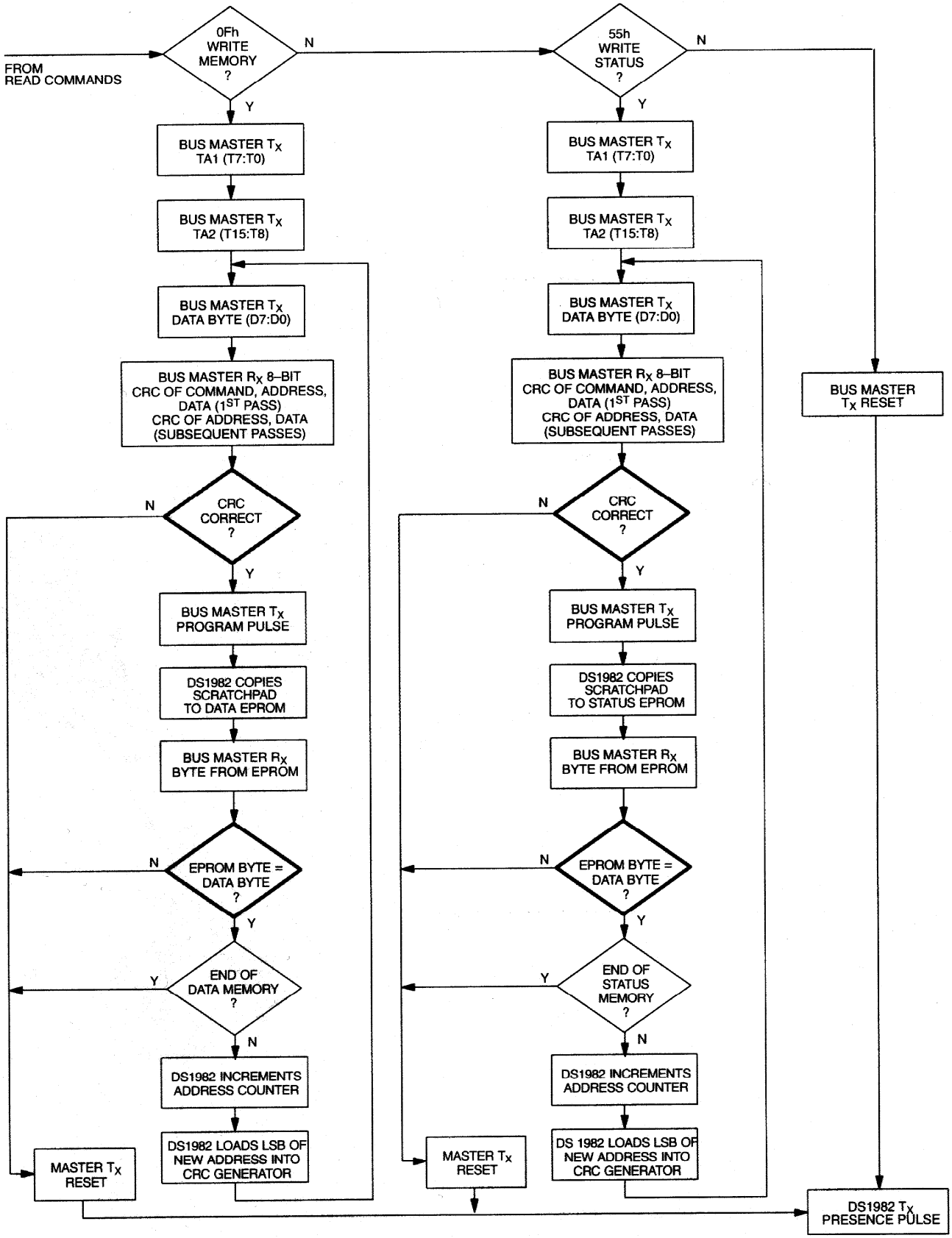
MEMORY FUNCTION FLOW CHART Figure 6



MEMORY FUNCTION FLOW CHART (cont'd) Figure 6



MEMORY FUNCTION FLOW CHART (cont'd) Figure 6



READ MEMORY [F0h]

The Read Memory command is used to read data from the 1024-bit EPROM data field. The bus master follows the command byte with a 2-byte address (TA1=(T7:T0), TA2=(T15:T8)) that indicates a starting byte location within the data field. An 8–bit CRC of the command byte and address bytes is computed by the DS1982 and read back by the bus master to confirm that the correct command word and starting address were received. If the CRC read by the bus master is incorrect, a Reset Pulse must be issued and the entire sequence must be repeated. If the CRC received by the bus master is correct, the bus master issues read time slots and receives data from the DS1982 starting at the initial address and continuing until the end of the 1024-bit data field is reached or until a Reset Pulse is issued. If reading occurs through the end of memory space, the bus master may issue eight additional read time slots and the DS1982 will respond with an 8-bit CRC of all data bytes read from the initial starting byte through the last byte of memory. After the CRC is received by the bus master, any subsequent read time slots will appear as logical 1s until a Reset Pulse is issued. Any reads ended by a Reset Pulse prior to reaching the end of memory will not have the 8-bit CRC available.

Typically a 16-bit CRC would be stored with each page of data to ensure 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 Book of DS19xx iButton Standards, Chapter 7 for the recommended file structure to be used with the 1-Wire environment.) If CRC values are imbedded within the data, a Reset Pulse may be issued at the end of memory space during a Read Memory command.

READ STATUS [AAh]

The Read Status command is used to read data from the EPROM Status data field. The bus master follows the command byte with a two-byte address (TA1=(T7:T0), TA2=(T15:T8)) that indicates a starting byte location within the data field. An 8–bit CRC of the command byte and address bytes is computed by the DS1982 and read back by the bus master to confirm that the correct command word and starting address were received. If the CRC read by the bus master is incorrect, a Reset Pulse must be issued and the entire sequence must be repeated. If the CRC received by the bus master is correct, the bus master issues read time slots and receives data from the DS1982 starting at the supplied address and continuing until the end of the EPROM Status data field is reached. At that point the bus master will receive an 8-bit CRC that is the result of shifting into the CRC generator all of the data bytes from the initial starting byte through the final factory-programmed byte that contains the 00h value.

This feature is provided since the EPROM Status information may change over time making it impossible to program the data once and include an accompanying CRC that will always be valid. Therefore, the Read Status command supplies an 8-bit CRC that is based on and always is consistent with the current data stored in the EPROM Status data field.

After the 8-bit CRC is read, the bus master will receive logical 1s from the DS1982 until a Reset Pulse is issued. The Read Status command sequence can be exited at any point by issuing a Reset Pulse.

READ DATA/GENERATE 8-BIT CRC [C3h]

The Read Data/Generate 8-bit CRC command is used to read data from the 1024-bit EPROM memory field. The bus master follows the command byte with a two-byte address (TA1=(T7:T0), TA2=(T15:T8)) that indicates a starting byte location within the data field. An 8-bit CRC of the command byte and address bytes is computed by the DS1982 and read back by the bus master to confirm that the correct command word and starting address were received. If the CRC read by the bus master is incorrect, a Reset Pulse must be issued and the entire sequence must be repeated. If the CRC received by the bus master is correct, the bus master issues read time slots and receives data from the DS1982 starting at the

initial address and continuing until the end of a 32-byte page is reached. At that point the bus master will send eight additional read time slots and receive an 8-bit CRC that is the result of shifting into the CRC generator all of the data bytes from the initial starting byte to the last byte of the current page. Once the 8-bit CRC has been received, data is again read from the 1024-bit EPROM data field starting at the next page. This sequence will continue until the final page and its accompanying CRC are read by the bus master. Thus each page of data can be considered to be 33 bytes long, the 32 bytes of user-programmed EPROM data and an 8-bit CRC that gets generated automatically at the end of each page.

This type of read differs from the Read Memory command that simply reads each page until the end of address space is reached. The Read Memory command only generates an 8-bit CRC at the end of memory space that often might be ignored, since in many applications the user would store a 16-bit CRC with the data itself in each page of the 1024-bit EPROM data field at the time the page was programmed. The Read Data/Generate 8-bit CRC command provides an alternate read capability for applications that are "bit-oriented" rather than "page-oriented" where the 1024-bit EPROM information may change over time within a page boundary, making it impossible to program the page once and include an accompanying CRC that will always be valid. Therefore, the Read Data/Generate 8-Bit CRC command concludes each page with the DS1982 generating and supplying an 8-bit CRC that is based on and therefore is always consistent with the current data stored in each page of the 1024-bit EPROM data field. After the 8-bit CRC of the last page is read, the bus master will receive logical 1s from the DS1982 until a Reset Pulse is issued. The Read Data/Generate 8-Bit CRC command sequence can be exited at any point by issuing a Reset Pulse.

WRITE MEMORY [0Fh]

The Write Memory command is used to program the 1024–bit EPROM data field. The bus master will follow the command byte with a two byte starting address (TA1=(T7:T0), TA2=(T15:T8)) and a byte of data (D7:D0). An 8-bit CRC of the command byte, address bytes, and data byte is computed by the DS1982 and read back by the bus master to confirm that the correct command word, starting address, and data byte were received.

The highest starting address within the DS1982 is 007FH. If the bus master sends a starting address higher than this, the nine most significant address bits are set to 0 by the internal circuitry of the chip. This will result in a mismatch between the CRC calculated by the DS1982 and the CRC calculated by the bus master, indicating an error condition.

If the CRC read by the bus master is incorrect, a Reset Pulse must be issued and the entire sequence must be repeated. If the CRC received by the bus master is correct, a programming pulse (12 volts on the 1-Wire bus for 480 μs) is issued by the bus master. Prior to programming, the entire unprogrammed 1024-bit EPROM data field will appear as logical 1s. For each bit in the data byte provided by the bus master that is set to a logical 0, the corresponding bit in the selected byte of the 1024–bit EPROM will be programmed to a logical 0 after the programming pulse has been applied at that byte location.

After the 480 μs programming pulse is applied and the data line returns to a 5-volt level, the bus master issues eight read time slots to verify that the appropriate bits have been programmed. The DS1982 responds with the data from the selected EPROM address sent least significant bit first. This byte contains the logical AND of all bytes written to this EPROM data address. If the EPROM data byte contains 1s in bit positions where the byte issued by the master contains 0s, a Reset Pulse should be issued and the current byte address should be programmed again. If the DS1982 EPROM data byte contains 0s in the same bit positions as the data byte, the programming was successful and the DS1982 will automatically increment its address counter to select the next byte in the 1024-bit EPROM data field. The least

significant byte of the new 2-byte address will also be loaded into the 8-bit CRC generator as a starting value. The bus master will issue the next byte of data using eight write time slots.
As the DS1982 receives this byte of data into the scratchpad, it also shifts the data into the CRC generator that has been preloaded with the LSB of the current address, and the result is an 8-bit CRC of the new data byte and the LSB of the new address. After supplying the data byte, the bus master will read this 8-bit CRC from the DS1982 with eight read time slots to confirm that the address incremented properly and the data byte was received correctly. If the CRC is incorrect, a Reset Pulse must be issued and the Write Memory command sequence must be restarted. If the CRC is correct, the bus master will issue a programming pulse and the selected byte in memory will be programmed.

Note that the initial pass through the Write Memory flow chart will generate an 8-bit CRC value that is the result of shifting the command byte into the CRC generator, followed by the 2 address bytes, and finally the data byte. Subsequent passes through the Write Memory flow chart due to the DS1982 automatically incrementing its address counter will generate an 8-bit CRC that is the result of loading (not shifting) the LSB of the new (incremented) address into the CRC generator and then shifting in the new data byte.

For both of these cases, the decision to continue (to apply a program pulse to the DS1982) is made entirely by the bus master, since the DS1982 will not be able to determine if the 8-bit CRC calculated by the bus master agrees with the 8-bit CRC calculated by the DS1982. If an incorrect CRC is ignored and a program pulse is applied by the bus master, incorrect programming could occur within the DS1982. Also note that the DS1982 will always increment its internal address counter after the receipt of the eight read time slots used to confirm the programming of the selected EPROM byte. The decision to continue is again made entirely by the bus master; therefore if the EPROM data byte does not match the supplied data byte but the master continues with the Write Memory command, incorrect programming could occur within the DS1982. The Write Memory command sequence can be exited at any point by issuing a Reset Pulse.

WRITE STATUS [55h]

The Write Status command is used to program the EPROM Status data field. The bus master will follow the command byte with a 2-byte starting address (TA1=(T7:T0), TA2=(T15:T8)) and a byte of status data (D7:D0). An 8-bit CRC of the command byte, address bytes, and data byte is computed by the DS1982 and read back by the bus master to confirm that the correct command word, starting address, and data byte were received.

If the CRC read by the bus master is incorrect, a Reset Pulse must be issued and the entire sequence must be repeated. If the CRC received by the bus master is correct, a programming pulse (12 volts on the 1-Wire bus for 480 μs) is issued by the bus master. Prior to programming, the first 7 bytes of the EPROM Status data field will appear as logical 1s. For each bit in the data byte provided by the bus master that is set to a logical 0, the corresponding bit in the selected byte of the EPROM Status data field will be programmed to a logical 0 after the programming pulse has been applied at that byte location. The 8th byte of the EPROM status byte data field is factory-programmed to contain 00h.

After the 480 μs programming pulse is applied and the data line returns to a 5-volt level, the bus master issues eight read time slots to verify that the appropriate bits have been programmed. The DS1982 responds with the data from the selected EPROM Status address sent least significant bit first. This byte contains the logical AND of all bytes written to this EPROM Status Byte address. If the EPROM Status Byte contains 1s in bit positions where the byte issued by the master contained 0s, a Reset Pulse should be issued and the current byte address should be programmed again. If the DS1982 EPROM Status Byte

contains 0s in the same bit positions as the data byte, the programming was successful and the DS1982 will automatically increment its address counter to select the next byte in the EPROM Status data field. The least significant byte of the new 2-byte address will also be loaded into the 8-bit CRC generator as a starting value. The bus master will issue the next byte of data using eight write time slots.

As the DS1982 receives this byte of data into the scratchpad, it also shifts the data into the CRC generator that has been preloaded with the LSB of the current address and the result is an 8-bit CRC of the new data byte and the LSB of the new address. After supplying the data byte, the bus master will read this 8-bit CRC from the DS1982 with eight read time slots to confirm that the address incremented properly and the data byte was received correctly. If the CRC is incorrect, a Reset Pulse must be issued and the Write Status command sequence must be restarted. If the CRC is correct, the bus master will issue a programming pulse and the selected byte in memory will be programmed.

Note that the initial pass through the Write Status flow chart will generate an 8-bit CRC value that is the result of shifting the command byte into the CRC generator, followed by the 2 address bytes, and finally the data byte. Subsequent passes through the Write Status flow chart due to the DS1982 automatically incrementing its address counter will generate an 8-bit CRC that is the result of loading (not shifting) the LSB of the new (incremented) address into the CRC generator and then shifting in the new data byte.

For both of these cases, the decision to continue (to apply a program pulse to the DS1982) is made entirely by the bus master, since the DS1982 will not be able to determine if the 8-bit CRC calculated by the bus master agrees with the 8-bit CRC calculated by the DS1982. If an incorrect CRC is ignored and a program pulse is applied by the bus master, incorrect programming could occur within the DS1982. Also note that the DS1982 will always increment its internal address counter after the receipt of the eight read time slots used to confirm the programming of the selected EPROM byte. The decision to continue is again made entirely by the bus master, therefore if the EPROM data byte does not match the supplied data byte but the master continues with the Write Status command, incorrect programming could occur within the DS1982. The Write Status command sequence can be ended at any point by issuing a Reset Pulse.

1-WIRE BUS SYSTEM

The 1-Wire bus is a system which has a single bus master and one or more slaves. In all instances, the DS1982 is a slave device. The bus master is typically a microcontroller. The discussion of this bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal type and timing). A 1-Wire protocol defines bus transactions in terms of the bus state during specified 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 an open drain connection or 3-state outputs. The DS1982 is an open drain part with an internal circuit equivalent to that shown in Figure 7. The bus master can be the same equivalent circuit. If a bidirectional pin is not available, separate output and input pins can be tied together.

The bus master requires a pullup resistor at the master end of the bus, with the bus master circuit equivalent to the one shown in Figures 8a and 8b. The value of the pullup resistor should be approximately 5 kΩ for short line lengths.

A multidrop bus consists of a 1-Wire bus with multiple slaves attached. The 1-Wire bus has a maximum data rate of 16.3kbps. If the bus master is also required to perform programming of the EPROM portions of the DS1982, a programming supply capable of delivering up to 10 milliamps at 12 volts for 480 μs is required. 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 120 μs, one or more of the devices on the bus may be reset.

TRANSACTION SEQUENCE

The sequence for accessing the DS1982 via the 1-Wire port is as follows:

- Initialization
- ROM Function Command
- Memory Function Command
- Read/Write Memory/Status

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 a presence pulse(s) transmitted by the slave(s).

The presence pulse lets the bus master know that the DS1982 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 four ROM function commands. All ROM function commands are 8 bits long. A list of these commands follows (refer to flowchart in Figure 9):

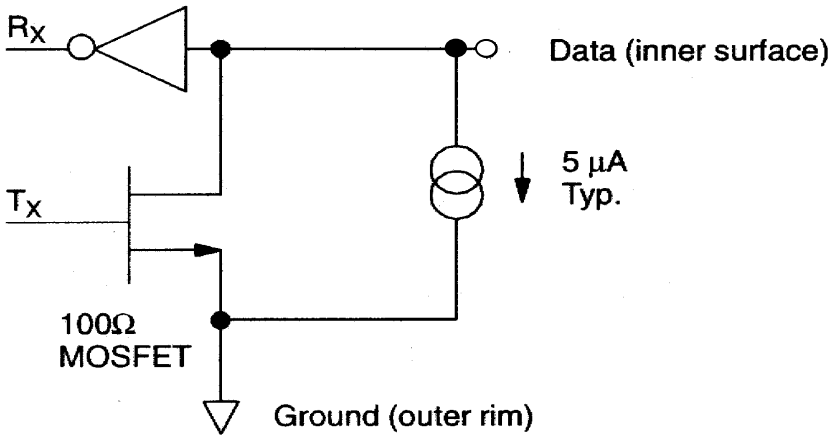
Read ROM [33h]

This command allows the bus master to read the DS1982's 8-bit family code, unique 48-bit serial number, and 8-bit CRC. This command can be used only if there is a single DS1982 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).

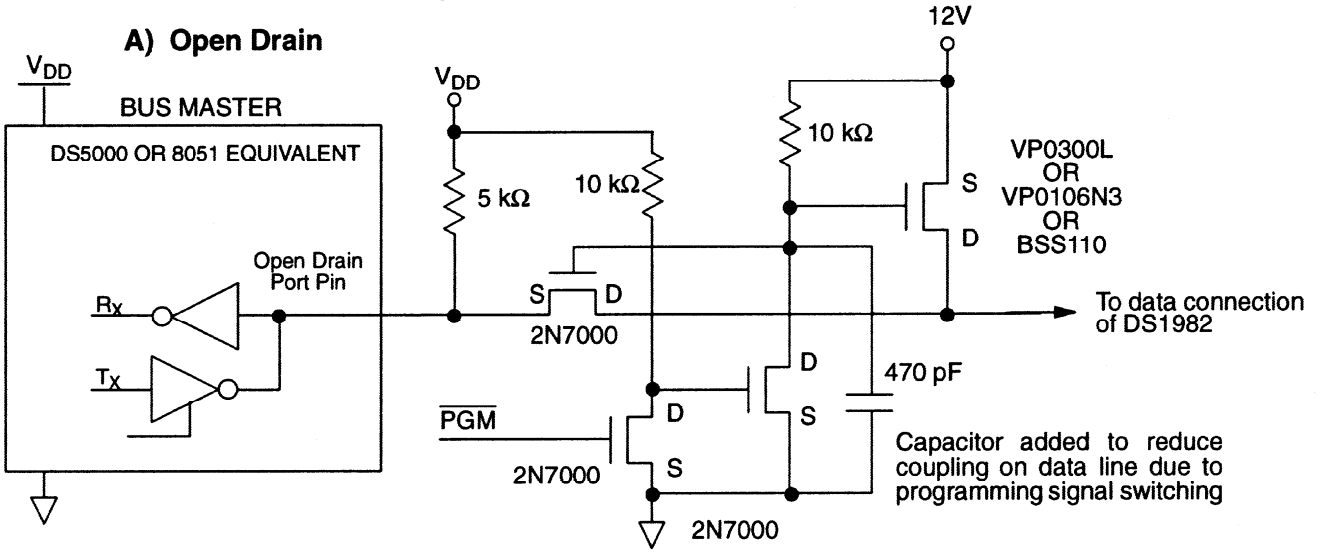
Match ROM [55h]

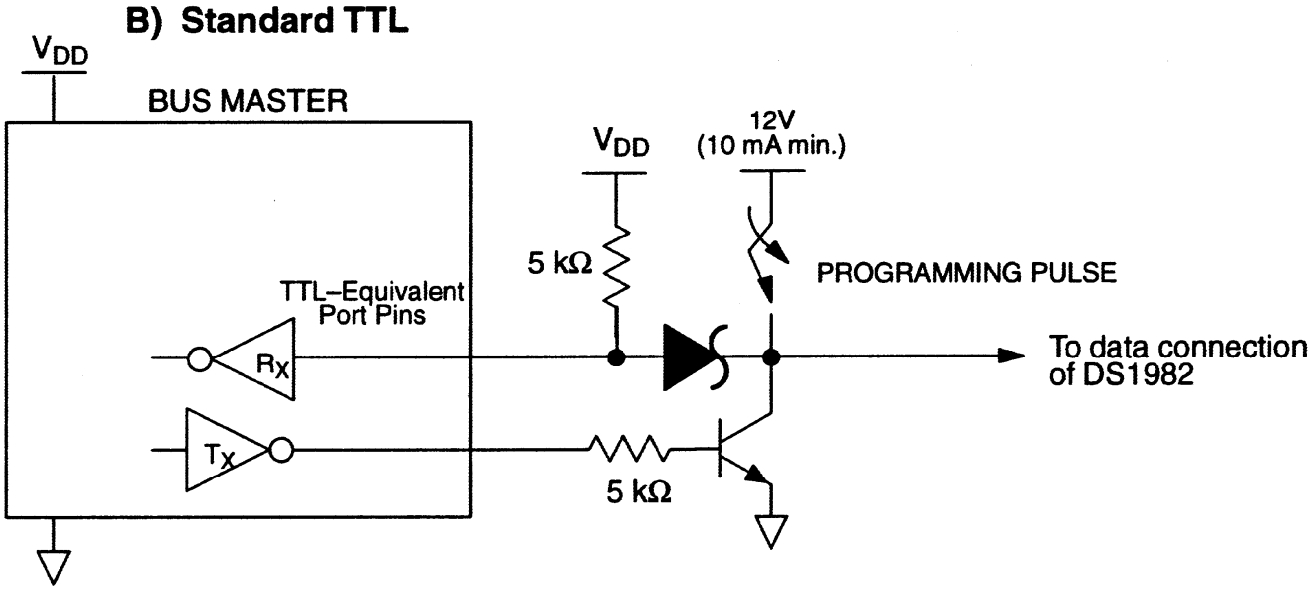
The match ROM command, followed by a 64–bit ROM sequence, allows the bus master to address a specific DS1982 on a multidrop bus. Only the DS1982 that exactly matches the 64-bit ROM sequence will respond to the subsequent memory function command. All slaves that do not match the 64-bit ROM sequence will wait for a Reset Pulse. This command can be used with a single or multiple devices on the bus.

DS1982 EQUIVALENT CIRCUIT Figure 7

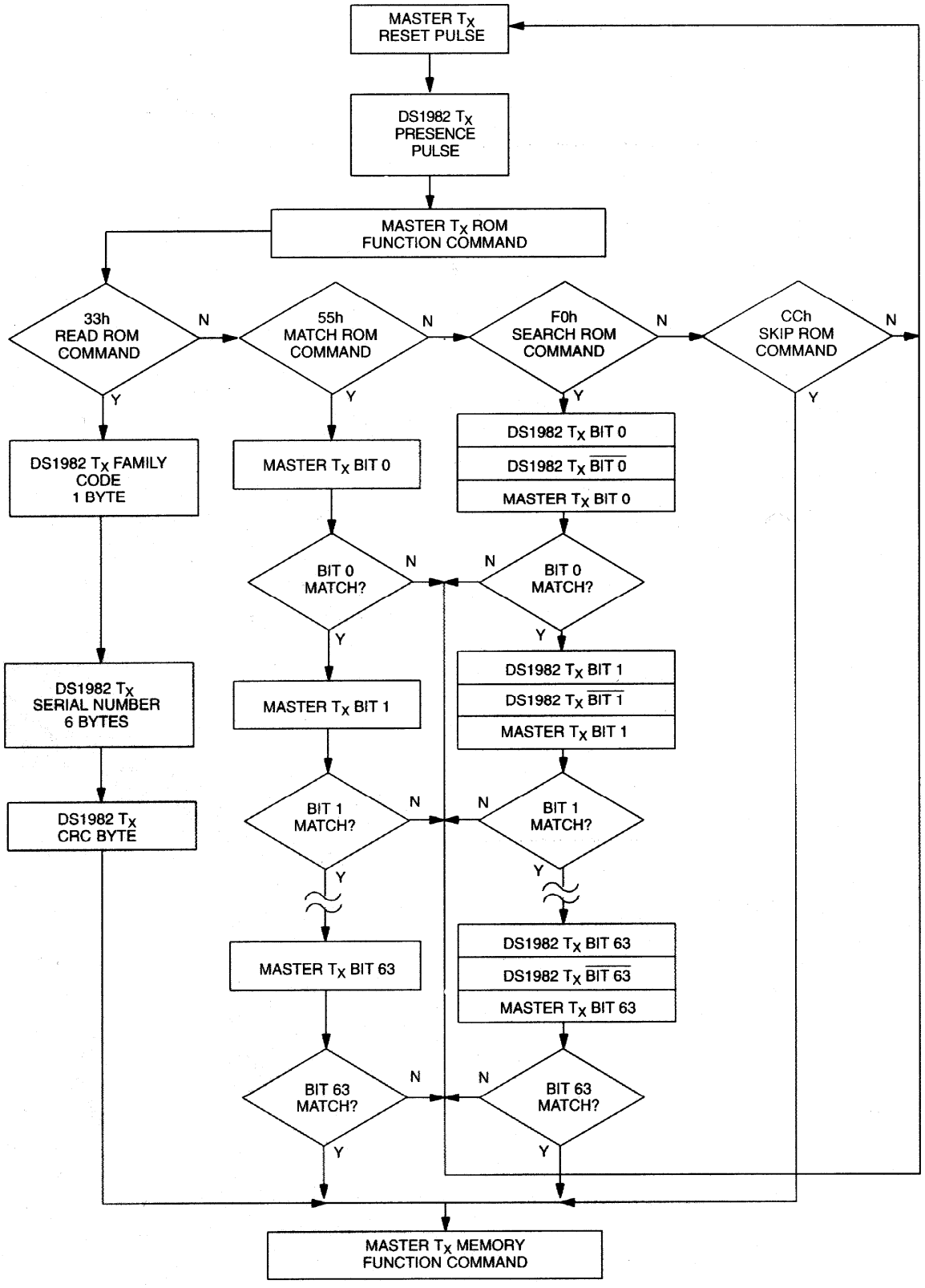


BUS MASTER CIRCUIT Figure 8





ROM FUNCTIONS FLOW CHART Figure 9



Skip ROM [CCh]

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

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 ROM search process is the repetition of a simple, three-step routine: read a bit, read the complement of the bit, then write the desired value of that bit. The bus master performs this simple, three-step routine on each bit of the ROM. After one complete pass, the bus master knows the contents of the ROM in one device. The remaining number of devices and their ROM codes may be identified by additional passes. See Chapter 5 of the Book of DS19xx iButton Standards for a comprehensive discussion of a ROM search, including an actual example.

1-Wire Signaling

The DS1982 requires strict protocols to ensure data integrity. The protocol consists of five types of signaling on one line: Reset Sequence with Reset Pulse and Presence Pulse, Write 0, Write 1, Read Data and Program Pulse. All these signals except presence pulse are initiated by the bus master. The initialization sequence required to begin any communication with the DS1982 is shown in Figure 10. A Reset Pulse followed by a presence pulse indicates the DS1982 is ready to accept a ROM command. The bus master transmits (TX) a Reset Pulse (t_{RSTL} , minimum 480 μs). The bus master then releases the line and goes into receive mode (RX). The 1-Wire bus is pulled to a high state via the pullup resistor. After detecting the rising edge on the 1-Wire line, the DS1982 waits (t_{PDH} , 15-60 μs) and then transmits the presence pulse (t_{PDL} , 60-240 μs).

Read/Write Time Slots

The definitions of write and read time slots are illustrated in Figure 11. All time slots are initiated by the master driving the data line low. The falling edge of the data line synchronizes the DS1982 to the master by triggering a delay circuit in the DS1982. During write time slots, the delay circuit determines when the DS1982 will sample the data line. For a read data time slot, if a 0 is to be transmitted, the delay circuit determines how long the DS1982 will hold the data line low overriding the 1 generated by the master. If the data bit is a 1, the device will leave the read data time slot unchanged.

PROGRAM PULSE

To copy data from the 8-bit scratchpad to the EPROM Data or Status Memory, a program pulse of 12 volts is applied to the data line after the bus master has confirmed that the CRC for the current byte is correct. During programming, the bus master controls the transition from a state where the data line is idling high via the pullup resistor to a state where the data line is actively driven to a programming voltage of 12 volts providing a minimum of 10 mA of current to the DS1982. This programming voltage (Figure 12) should be applied for 480 μs, after which the bus master returns the data line to an idle high state controlled by the pullup resistor. Note that due to the high voltage programming requirements for any 1-Wire EPROM device, it is not possible to multidrop non-EPROM based 1-Wire devices with the DS1982 during programming. An internal diode within the non-EPROM based 1-Wire devices will attempt to clamp the data line at approximately 8 volts and could potentially damage these devices.

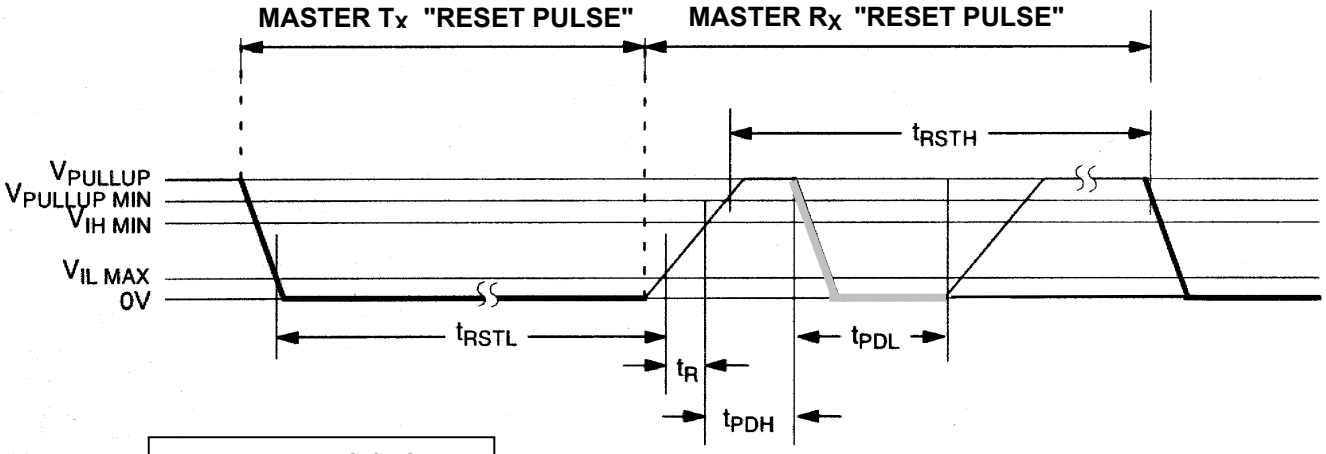
CRC GENERATION

The DS1982 has an 8-bit CRC stored in the most significant byte of the 64-bit ROM. The bus master can compute a CRC value from the first 56 bits of the 64-bit ROM and compare it to the value stored within the DS1982 to determine if the ROM data has been received error-free by the bus master. The equivalent polynomial function of this CRC is: $X^8 + X^5 + X^4 + 1$.

Under certain conditions, the DS1982 also generates an 8-bit CRC value using the same polynomial function shown above and provides this value to the bus master to validate the transfer of command, address, and data bytes from the bus master to the DS1982. The Memory Function Flow Chart of Figure 6 indicates that the DS1982 computes an 8-bit CRC for the command, address, and data bytes received for the Write Memory and the Write Status commands and then outputs this value to the bus master to confirm proper transfer. Similarly the DS1982 computes an 8–bit CRC for the command and address bytes received from the bus master for the Read Memory, Read Status, and Read Data/Generate 8-Bit CRC commands to confirm that these bytes have been received correctly. The CRC generator on the DS1982 is also used to provide verification of error-free data transfer as each page of data from the 1024-bit EPROM is sent to the bus master during a Read Data/Generate 8-bit CRC command, and for the 8 bytes of information in the status memory field.

In each case where a CRC is used for data transfer validation, the bus master must calculate a CRC value using the polynomial function given above and compare the calculated value to either the 8-bit CRC value stored in the 64-bit ROM portion of the DS1982 (for ROM reads) or the 8-bit CRC value computed within the DS1982. The comparison of CRC values and decision to continue with an operation are determined entirely by the bus master. There is no circuitry on the DS1982 that prevents a command sequence from proceeding if the CRC stored in or calculated by the DS1982 does not match the value generated by the bus master. Proper use of the CRC as outlined in the flow chart of Figure 6 can result in a communication channel with a very high level of integrity. For more details on generating CRC values including example implementations in both hardware and software, see the Book of DS19xx iButton Standards.

INITIALIZATION PROCEDURE “RESET AND PRESENCE PULSES” Figure 10

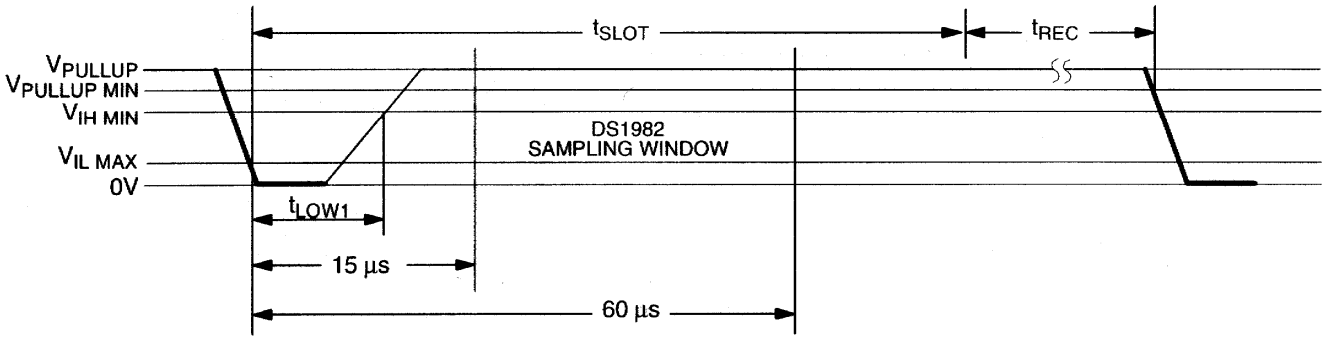


480 µs ≤ t_{RSTL} < *
480 µs ≤ t_{RSTH} < ∞ (includes recovery time)
15 µs ≤ t_{PDH} < 60 µs
60 µs ≤ t_{PDL} < 240 µs

**In order not to mask interrupt signaling by other devices on the 1-Wire bus, t_{RSTL} + t_R should always be less than 960 µs.*

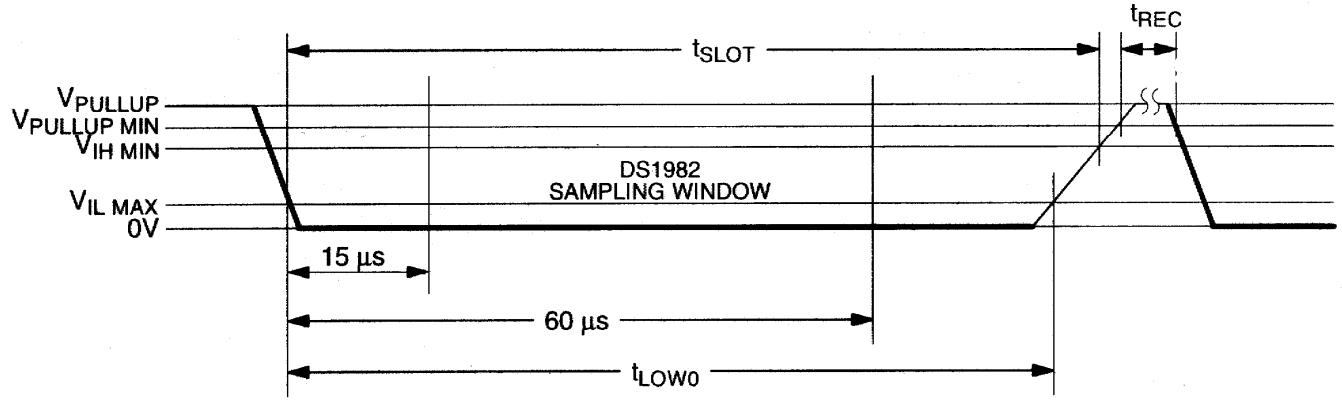
READ/WRITE TIMING DIAGRAM Figure 11

Write-1 Time Slot



$60\ \mu s \le t_{SLOT} < 120\ \mu s$
$1\ \mu s \le t_{LOW1} < 15\ \mu s$
$1\ \mu s \le t_{REC} < \infty$

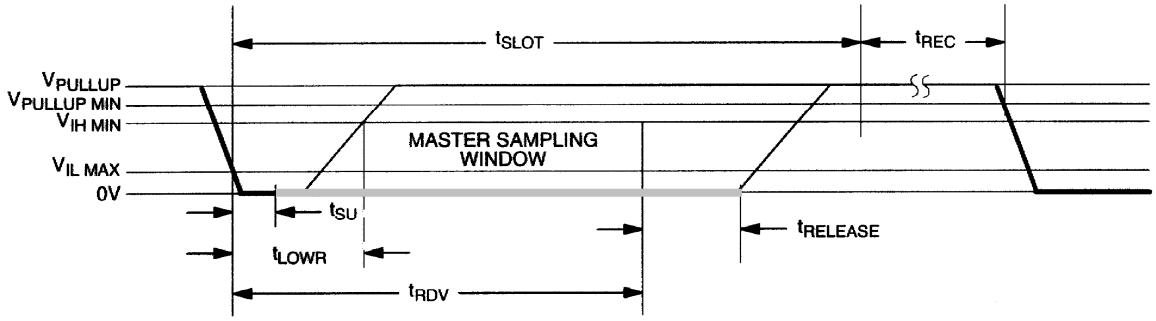
Write-0 Time Slot



$60\ \mu s \le t_{LOW0} < t_{SLOT} < 120\ \mu s$
$1\ \mu s \le t_{REC} < \infty$



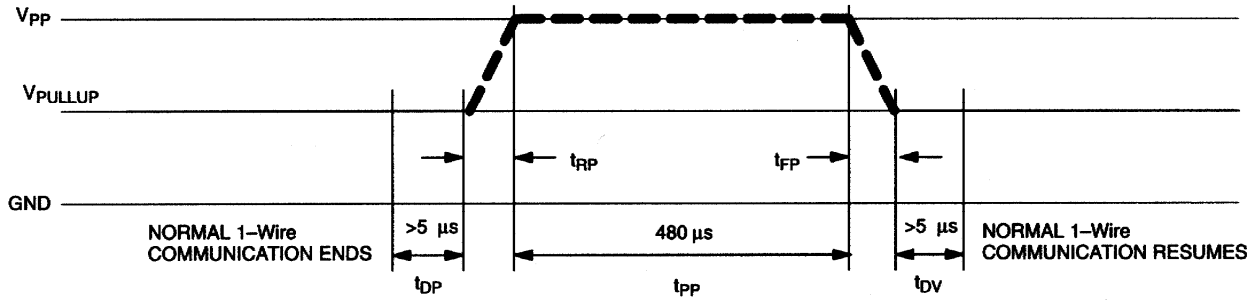
Read-data Time Slot

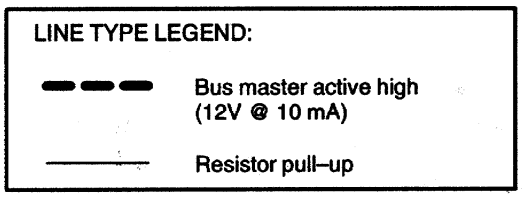




$60\ \mu s \le t_{SLOT} < 120\ \mu s$
$1\ \mu s \le t_{LOWR} < 15\ \mu s$
$0 \le t_{RELEASE} < 45\ \mu s$
$1\ \mu s \le t_{REC} < \infty$
$t_{RDV} = 15\ \mu s$
$t_{SU} < 1\ \mu s$

PROGRAM PULSE TIMING DIAGRAM Figure 12





ABSOLUTE MAXIMUM RATINGS*

Voltage on any Pin Relative to Ground	–0.5V to +12.0V
Operating Temperature	–40°C to +85°C
Storage Temperature	–55°C to +125°C

* This is a stress rating only and functional operation of the device at these or any other conditions outside 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 6.0V; -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Logic 1	V_{IH}	2.2			V	1, 6
Logic 0	V_{IL}	-0.3		+0.8	V	1, 11
Output Logic Low @ 4 mA	V_{OL}			0.4	V	1
Output Logic High	V_{OH}		V_{PUP}	6.0	V	1, 2
Input Load Current	I_L		5		µA	3
Operating Charge	Q_{OP}			30	nC	7, 8
Programming Voltage @ 10 mA	V_{PP}	11.5		12.0	V	

CAPACITANCE (T_A = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Data (1-Wire)	$C_{IN/OUT}$			800	pF	9

AC ELECTRICAL CHARACTERISTICS (V_{PUP}=2.8V to 6.0V; -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 Data Valid	t_{RDV}	exactly 15			µs	
Release Time	$t_{RELEASE}$	0	15	45	µs	
Read Data Setup	t_{SU}			1	µs	5
Recovery Time	t_{REC}	1			µs	
Reset Time High	t_{RSTH}	480			µs	4
Reset Time Low	t_{RSTL}	480			µs	
Presence Detect High	t_{PDHIGH}	15		60	µs	
Presence Detect Low	t_{PDLOW}	60		240	µs	
Delay to Program	t_{DP}	5			µs	10
Delay to Verify	t_{DV}	5			µs	10
Program Pulse Width	t_{PP}	480		5000	µs	10,12
Program Voltage Rise Time	t_{RP}	0.5		5.0	µs	10
Program Voltage Fall Time	t_{FP}	0.5		5.0	µs	10

NOTES:

1. All voltages are referenced to ground.

2. V_{PUP} = external pullup voltage.

3. Input load is to ground.

4. An additional reset or communication sequence cannot begin until the reset high time has expired.

5. Read data setup time refers to the time the host must pull the 1-Wire bus low to read a bit. Data is guaranteed to be valid within 1 µs of this falling edge and will remain valid for 14 µs minimum. (15 µs total from falling edge on 1-Wire bus.)

6. V_{IH} is a function of the external pullup resistor and the pullup voltage.

7. 30 nanocoulombs per 72 time slots @ 5.0V.

8. At V_{CC} =5.0V with a 5 kΩ pullup to V_{CC} and a maximum time slot of 120 µs.

9. Capacitance on the data pin could be 800 pF when power is first applied. If a 5 kΩ resistor is used to pull up the data line to V_{CC} , 5 ms after power has been applied the parasite capacitance will not affect normal communications.

10. Maximum 1-Wire voltage for programming parameters is 11.5V to 12.0V; temperature range is -40°C to +50°C.

11. Under certain low voltage conditions V_{ILMAX} may have to be reduced to as much as 0.5V to always guarantee a presence pulse.

12. The accumulative duration of the programming pulses for each address must not exceed 5 ms.

REVISION HISTORY

REVISION DATE	DESCRIPTION	PAGES CHANGED
7/08	Updated the *F3 MicroCan* and *F5 MicroCan* face brands with the latest per PCN H020201.	1
8/09	Added plus signs (+) to the *Ordering Information* to reflect the conversion to a lead-free product. Deleted the UL bullet from the *Features* section.	1

Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408-737-7600

© 2009 Maxim Integrated Products Maxim is a registered trademark of Maxim Integrated Products, Inc.