AN3699

Using ATtiny817 to Emulate a 1-Wire® Device

Introduction

Author: Daniel Xiong, Microchip Technology Inc.

Communication with 1-Wire® devices is unique in that only one wire, in addition to the ground, is needed to communicate with another device. Power supply and communications are handled through only one connection. To communicate with a Maxim's 1-Wire device, only one general purpose I/O pin is needed. This application note shows how a 1-Wire target device, DS2433, can be emulated using an ATtiny817.

Features

- Supports Standard Speed Maxim's 1-Wire Protocol
- Compatible with All AVR® Devices
- Interrupt-Driven Implementation
- Requires No External Hardware

Table of Contents

1. Theory of Operation - Maxim's 1-Wire® Protocol

A 1-Wire bus uses only one wire for signaling and power. Communication is asynchronous and half-duplex, and it follows a strict host/target scheme. One or several target devices can be connected to the bus at the same time, but only one host should be connected to the bus.

The bus is idle high, so a pull-up resistor is necessary. To determine the value of the pull-up resistor, refer to the data sheet of the target device(s). All devices connected to the bus must be able to drive the bus low. An open-collector or open-drain buffer is required if a device is connected through a pin that can not be put in a Tri-State mode. Signaling on the 1-Wire bus is divided into time slots of 60 µs. One data bit is transmitted on the bus per time slot. Target devices are allowed to have a time base that differs significantly from the nominal time base. This, however, requires the timing of the host to be very accurate to ensure correct communication with target devices with different time bases. It is, therefore, very important to obey the time limits described in the following sections.

1.1 Basic Bus Signals

The host initiates every communication on the bus at the bit-level. This means that for every bit that is to be transmitted, regardless of direction, the host has to initiate the bit transmission. This is done by pulling the bus low, which will synchronize the timing logic of all units. There are five basic commands for communication on the 1-Wire bus:

1. 'Write 1'.
2. 'Write 0'.
3. 'Read'.
4. 'Reset'.
5. 'Presence'.

1.1.1 'Write 1' Signal

A 'Write 1' signal is shown in the figure below. The host pulls the bus low for 1 to 15 µs. Then, it releases the bus for the rest of the time slot.

Figure 1-1. 'Write 1' Signal

1.1.2 'Write 0' Signal

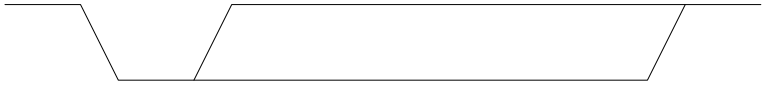
A 'Write 0' signal is shown in the figure below. The host pulls the bus low for a period of at least 60 µs, with a maximum length of 120 µs.

Figure 1-2. 'Write 0' Signal

1.1.3 'Read' Signal

A 'Read' signal is shown in the figure below. The host pulls the bus low for 1 to 15 µs. Then, the target holds the bus low if it wants to send a '0'. If it wants to send a '1', it simply releases the line. The bus should be sampled 15 µs after the bus was pulled low. As seen from the host's side, the 'Read' signal is a 'Write 1' signal. It is the internal state of the target, rather than the signal itself that dictates whether it is a 'Write 1' or 'Read' signal.

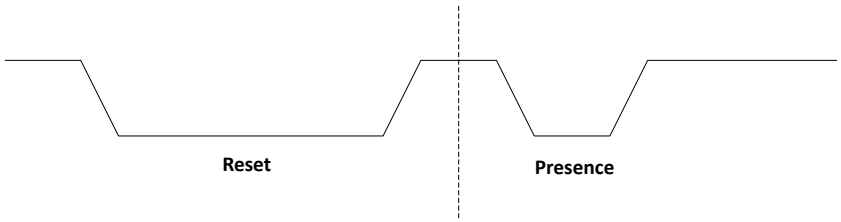
Figure 1-3. 'Read' Signal



1.1.4 'Reset/Presence' Signal

A 'Reset' and 'Presence' signal is shown in the figure below. Note that the time scale is different from the first waveforms. The host pulls the bus low for at least eight time slots, or 480 µs, and then releases it. This long low period is called the 'Reset' signal. If there is a target present, it will pull the bus low within 60 µs after it is released by the host and hold it low for at least 60 µs. This response is called a 'Presence' signal. If no presence signal is issued on the bus, the host must assume that no device is present on the bus, and further communication is not possible.

Figure 1-4. 'Reset' and 'Presence' Signal



1.1.5 Generating the Signals in Software

Generating the 1-Wire signals on an AVR by the software is straightforward. Simply changing the direction and value of a general purpose I/O pin and generating the required delay is sufficient. A detailed description of this is given in the Implementation section.

1.2 ROM Function Commands

Every 1-Wire device contains a globally unique 64-bit identifier number stored in ROM. This number can be used to facilitate addressing or identification of individual devices on the bus. The identifier consists of three parts: An 8-bit family code, a 48-bit serial number, and an 8-bit CRC computed from the first 56 bits. A small set of commands, called ROM function commands that operate on the 64-bit identifier is defined. The table below lists the six defined ROM commands.

Table 1-1. ROM Commands

Command	Code	Usage
READ ROM	33H	Identification
SKIP ROM	CCH	Skip addressing
MATCH ROM	55H	Address specific device
SEARCH ROM	F0H	Obtain IDs of all devices on the bus
OVERDRIVE SKIP ROM	3CH	Overdrive version of SKIP ROM
OVERDRIVE MATCH ROM	69H	Overdrive version of MATCH ROM

1.2.1 READ ROM Command

The 'READ ROM' command can be used on a bus with a single target to read the 64-bit unique identifier. If there are several target devices connected to the bus, the result of this command will be the AND result of all target device identifiers. A failed CRC indicates the presence of several targets.

1.2.2 SKIP ROM Command

The 'SKIP ROM' command can be used without targeting any specific target. On a one-target bus, the 'SKIP ROM' command is sufficient for addressing. On a multiple-target bus, the 'SKIP ROM' command can be used to address all devices at once. This is only useful when sending commands to target devices, e.g., to start temperature conversions on several temperature sensors at once. It is not possible to use the 'SKIP ROM' command when reading from target devices on a multiple-target bus.

1.2.3 MATCH ROM Command

Use the 'MATCH ROM' command to address individual target devices on the bus. After the 'MATCH ROM' command, the complete 64-bit identifier is transmitted on the bus. When done, only the device with this identifier is allowed to respond until the next reset pulse is received.

1.2.4 SEARCH ROM Command

The 'SEARCH ROM' command can be used when the identifiers of all target devices are not known in advance. It makes it possible to discover the identifiers of all the targets present on the bus. First, the 'SEARCH ROM' command is transmitted on the bus. Then, the target reads one bit from the bus. Each target places the first bit of its identifier on the bus. The target will read this as the logical AND result of the first bit of all target identifiers. The target then reads one more bit from the bus. Each target then places the complement of the first bit of its identifier on the bus. The target will read this as the logical AND of the complement of the first bit of the identifier of all targets. If all devices have '1' as the first bit, the target will read 10b. Similarly, if all devices have '0' as the first bit, the target will read 01b. In these cases, the bit can be stored as the first bit of all addresses. The target will then write back this bit, which in effect, will tell all targets to keep sending identifier bits. If there are devices with both '0' and '1' as the first bit in the identifier on the bus, the target will read 00. In this case, the target must choose whether to continue with the addresses that have '0' in this position or '1'. The choice is transmitted on the bus, in effect making all targets that do not have this bit in this position of the identifier enter an idle state.

The target then goes on to read the next bit, and the process repeats until all 64 bits are read. The target should then have discovered one complete 64-bit identifier. To discover more identifiers, the 'SEARCH ROM' command should be run again, but this time a different choice for the bit value should be made the first time there is a discrepancy. Repeating this once for each target device should discover all targets. Note that when one search has been performed, all targets except one should have entered an idle state. It is now possible to communicate with the active target without specifically addressing it with the 'MATCH ROM' command.

1.2.5 OVERDRIVE ROM Commands

The overdrive ROM commands are outside the scope of this document. This application note only covers standard speed.

1.3 Memory Function Commands

Memory function commands are commands that are specific to one device or a class of devices. These commands typically deal with reading and writing of internal memory and registers in target devices. Several memory function commands are defined, but not all commands are used by all devices. The order of writes and reads is specific to each device, not part of the general specification. The table below lists the memory function commands of DS2433.

Table 1-2. Memory Function Commands of DS2433

Command	Code	Usage
WRITE SCRATCHPAD	0FH	Write data to scratchpad
READ SCRATCHPAD	AAH	Verify scratchpad data and target address
COPY SCRATCHPAD	55H	Copy data from the scratchpad to memory
READ MEMORY	F0H	Read data from memory

1.3.1 WRITE SCRATCHPAD Command

This command is used to write data to the scratchpad. The scratchpad is an additional page that acts as a buffer when writing to memory. After issuing the write SCRATCHPAD command, the host 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 host 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.

1.3.2 READ SCRATCHPAD Command

This command is used to verify scratchpad data and target address. After issuing the READ SCRATCHPAD command, the host begins reading. The first two 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 host may read data until the end of the scratchpad, after which the data read will be all logic 1s.

1.3.3 COPY SCRATCHPAD Command

This command is used to copy data from the scratchpad to memory. After issuing the COPY SCRATCHPAD command, the host must provide a 3-byte authorization pattern, which can be obtained by reading the scratchpad for verification. This pattern must exactly match the data contained in the three address registers (TA1, TA2, E/S, in that order). If the pattern matches, the Authorization Accepted (AA) flag will be set, and the copy will begin. A copy takes a maximum of 5 ms, during which the voltage on the 1-Wire bus must not fall below 2.8V. A pattern of alternating 1s and 0s will be received after the data has been copied until a Reset Pulse is issued by the host.

1.3.4 READ MEMORY Command

This command is used to read data from memory. After issuing the READ MEMORY command, the host must provide the 2-byte target address. After the two bytes, the host reads data beginning from the target address and may continue until the end of memory, at which point logic 1s will be read. It is of importance to realize that the target address registers will contain the address provided. The ending offset/data status byte is unaffected.

1.4 Putting It All Together

All 1-Wire devices follow a basic communication sequence:

1. The host sends the 'Reset' pulse.
2. The target/targets responds/respond with a 'Presence' pulse.
3. The host sends a ROM command; this effectively addresses one or several target devices.
4. The host sends a memory function command.

Note: It is not necessary to complete the entire sequence. For example, it is possible to send a new 'Reset' after finishing a ROM command to start a new communication.

2. Implementation

It is possible to implement the 1-Wire protocol in software only, without using any special hardware. This solution has the advantage that the only hardware required is one general purpose I/O pin (GPIO). Since all GPIO pins on the AVR are bi-directional and have selectable internal pull-up resistors, the AVR can control a 1-Wire bus with no external support-circuitry. In case the internal pull-up resistor is not suitable for the current configuration of the host, only one external resistor is necessary. This solution uses the GPIO falling edge interrupt to capture the falling edge of the 1-Wire bus and then implements the protocol in the interrupt handler.

2.1 Initialization

The initialization procedure is very straightforward. Set the 1-Wire pins in Input mode, enable the internal pull-up resistor, and assign an interrupt handler to the GPIO interrupt.

2.2 Bit-Level Functions

The bit-level functions are implemented according to the 1-Wire protocol. All timing parameters comply with the recommended values in the DS2433 data sheet.

Bit-level Function	Description
uint8_t OW_detect_reset (void);	This function is used to detect the ‘Reset’ signal
void OW_presence(void);	This function is used to send ‘Presence’ signal
void OW_write_bit (uint8_t write_data);	This function is used to transmit a bit of data to the host device
uint8_t OW_read_bit (void);	This function is used to receive a bit of data from the host device

Two macros: OW_PULL_BUS_LOW() and OW_RELEASE_BUS() are included to drive the bus low and to release the bus. These are implemented as macros because they occur frequently, and the overhead caused by function calls is unwanted because of the strict timing requirements.

2.3 Byte-Level Functions

Byte-level Function	Description
void OW_write_byte (uint8_t write_data);	This function is used to transmit a byte of data to the host device
uint8_t OW_read_byte (void);	This function is used to receive a byte of data from the host device

2.4 ROM Commands

All general ROM commands for standard speed communication are implemented. Also, all general ROM commands are implemented in **void OW_InterruptHandler(void)** function using a state machine. The state machine is implemented according to the DS2433 data sheet, but it can be used when emulating any 1-Wire devices. The simplest ROM command is the SKIP ROM command; it changes the bus state from waiting for a ROM command to waiting for a memory function command.

2.5 Memory Function Commands

All memory function commands for standard speed communication are implemented. Also, all memory function commands are implemented in **void DS2433_Memory_Function(uint8_t cmd)** function using a state machine. The state machine is implemented according to the DS2433 data sheet, and it can only be used when emulating DS2433.

2.6 Cyclic Redundancy Check (CRC)

Cyclic Redundancy Check (CRC) is used by 1-Wire devices to ensure data integrity. The theory behind CRC is outside the scope of this document and will not be further discussed.

Two different CRCs are commonly found in 1-Wire devices, also found in the DS2433 data sheet:

- One 8-bit CRC (Maxim's 1-Wire CRC, DOW-CRC, or simply CRC8): Used in the ROM section of all 1-Wire devices; CRC8 is also in some devices used to verify other data, such as commands issued on the bus.
- One 16-bit CRC (CRC16): Used by some devices to check for errors on bigger data sets.

CRC Function	Description
uint8_t crc8_maxim(uint8_t *data, uint8_t length)	This function is used to compute CRC8
uint16_t crc16_maxim(uint8_t *data, uint8_t length)	This function is used to compute CRC16

3. Conclusion

This application note shows how to emulate a 1-Wire target device, DS2433, using an ATtiny817 microcontroller. Only standard speed is described in this application note. To emulate other 1-Wire target devices, the **void DS2433_Memory_Function(uint8_t cmd)** function needs to be implemented according to the device's data sheet.

4. References

Data sheet for the 4 Kb 1-Wire EEPROM.

5. Revision History

Table 5-1.

Doc Rev.	Date	Comments
A	11/2020	Initial document release

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN: 978-1-5224-6953-7

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820