Programming a Base Configuration for the ThunderSWITCH 24/3 (TNETX3270), ThunderSWITCH 16/3 (TNETX3190), ThunderSWITCH 12/3 (TNETX3151), and ThunderSWITCH 8/3 (TNETX3110)

APPLICATION REPORT: SPWA011

Networking Business Unit
February 1998

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE (“CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty, or endorsement thereof.

TRADEMARKS

TI and ThunderSWITCH are trademarks of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

PIC Telephone	(972) 644-5580
PIC FAX	(972) 480-7800
Email	networks@ti.com

Contents

Figures

Tables

Programming a Base Configuration for the ThunderSWITCH 24/3 (TNETX3270), ThunderSWITCH 16/3 (TNETX3190), ThunderSWITCH 12/3 (TNETX3151), and ThunderSWITCH 8/3 (TNETX3110)

Abstract

This application report describes how to program Texas Instruments' (TI™) ThunderSWITCH™ 24/3(TNETX3270), ThunderSWITCH 16/3 (TNETX3190), ThunderSWITCH 12/3 (TNETX3151), and ThunderSWITCH 8/3 (TNETX3110), also referred to in this document as the ThunderSWITCH devices. This report includes a description of the Electrically Erasable Programmable Read Only Memory (EEPROM) interface and handshaking protocol, configuration examples, description of the built-in data security protection, and product support information. Also included are CRC and EEPROM data read and write source code listings along with a CRC calculation program.

Introduction

The EEPROM interface is provided so the system level manufacturer can produce an unmanaged pre-configured system to their customers. Customers may also wish to change or reconfigure their system and retain their preferences. The EEPROM contains configuration, initialization, and CRC (Cyclic Redundancy Check) information accessed infrequently, typically at power up and after a reset.

EEPROMs sometimes corrupt the configuration and initialization information. Corruption is caused by erroneous or false data writes. Noise spikes, power supply glitches, or bus contention problems may cause a false write or erase cycle. Once the EEPROM data is corrupted, it cannot be cleared by simply removing the power; the data is permanently corrupted and the false data must be detected.

Texas Instruments provides a detection mechanism. If the ThunderSWITCH device detects the presence of an EEPROM, the contents are read and a CRC calculation is performed on the EEPROM data. The calculated CRC value is then compared to a previously written CRC (within the EEPROM). If the CRC values match, the ThunderSWITCH device is properly configured and processing begins. If the CRC values do not match, the device remains in a non-operational state and the FAULT LED is turned on. For EEPROM based system, a valid CRC is always required. Any time new data is written to the EEPROM, a new CRC value must be calculated (via software) and saved within the EEPROM.

The ThunderSWITCH device contains the logic to perform a CRC. The EEPROM CRC algorithm is based on the IEEE802.3u specification for frame check sequence (FCS) field. The following description of the CRC algorithm is adapted from the IEEE802.3u specification.

In the TNETX3270, TNETX3190, TNETX3151, and TNETX3110, the four-byte CRC value is stored in addresses 00FC-00FFh. The encoding is defined by the following generating polynomial:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

ThunderSWITCH-to-EEPROM interface

Either the 24C02 or 24C08 serial EEPROM device is used in a CPU-less environment. Both EEPROMs use a two wire serial interface for communication and are available in a small footprint package.

The 24C02 provides 2048 bits organized as 256 x 8. The 24C02 provides port and system configuration information to the ThunderSWITCH device (see Table 1 for details).

The 24C08 provides 8192 bits organized as 1024 x 8. The 24C08 provides port, system, and VLAN configuration information to the ThunderSWITCH device (see Table 1 for details).

Table 1. ThunderSWITCH-to-EEPROM/DIO Mapping

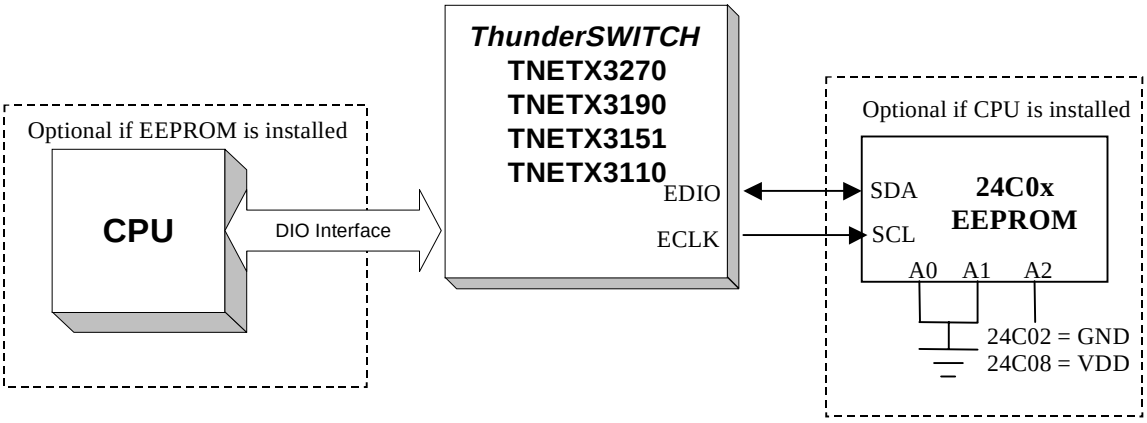
REGISTERS	Loadable using 24C02 EEPROM	Loadable using 24C08 EEPROM	DIO Address range
Port control registers	Yes	Yes	0000h:003Fh
Address lookup configuration registers	Yes	Yes	0040h:009Fh
System registers	Yes	Yes	00A0h:00FFh
VLAN registers	No	Yes	0100h:03FFh
Port status registers	No	No	0400h:043Fh
Address lookup registers	No	No	0440h:07FFh
System registers	No	No	0800h:0820h
Hardware reset	No	No	4000h:5FFFh
Statistic registers	No	No	8000h:A00Bh

The EEPROM size is detected automatically according to the address assigned to the EEPROM (shown in Figure 4):

A 24C02 EEPROM should have its **A0**, **A1**, and **A2** pins tied low.

A 24C08 EEPROM should have **A0** and **A1** pins tied low and **A2** pin tied high.

Figure 1. ThunderSWITCH-to-EEPROM interface



After the initial start condition, a slave address containing a device address of 000 is output on **EDIO**. Then **EDIO** is observed for an acknowledge from the EEPROM. If one is received, operation will continue for the 24C02 EEPROM. If none is received, another start condition will be issued, this time with device address 101. If this too receives no acknowledge, it is deduced that no EEPROM is present and device operation continues using the current register settings (that is, those following a hardware reset or those previously entered by software). When this device is driving **EDIO** it will only drive out a strong logical-0. When a logical-1 is intended to be driven out the pin must be resistively pulled high; an on-chip 50uA current-source pull-up device is provided on this pin. The system designer must decide if this is sufficient to achieve a logical-1 level in a timely manner or if an external supplementary resistor is required.

Once a 24C02 EEPROM is detected, (after power-up or after a reset) the ThunderSWITCH device reads the contents of the EEPROM (in ascending order from 000h to 0FFh) and initializes DIO registers 0000h - 00FBh. As the TNETX3270 reads the EEPROM data, it calculates a CRC; if the calculated CRC value matches the value stored at the EEPROM addresses 0FCh - 0FFh, theTNETX3270 begins operation.

If a 24C08 EEPROM is detected (after power-up or after a reset), the ThunderSWITCH device reads the contents of the EEPROM (in ascending order from 100h to 3FFh, followed by address 000h continuing through to 0FFh) and initializes DIO registers 0000h – 03FFh. As the ThunderSWITCH device reads the EEPROM data, it calculates a CRC; if the calculated CRC value matches the value stored at the EEPROM addresses 0FCh - 0FFh, the ThunderSWITCH device begins operation.

All ThunderSWITCH devices (TNETX3270, TNETX3190, TNETX3151, and TNETX3110) contain the same Device ID (02). Since this could cause havoc with software support, TI recommends that a unique identifier be placed at address A3h within the EEPROM. If the Device ID register = 02, the software should then read the EEPROM Device ID location. The following is a list of recommended Device IDs:

- TNETX3270 Device ID = 02
- TNETX3190 Device ID = 05
- TNETX3151 Device ID = 06
- TNETX3110 Device ID = 07

The EEPORM Device ID serves two purposes:

- It identifies the EEPROM for the given device.
- It uniquely identifies the ThunderSWITCH device.

Example 24C02 EEPROM Configurations for the ThunderSWITCH Devices

Table 2 through Table 11 show examples of the 24C02 EEPROM configuration. Notice that all values are in hexadecimal notation. The following list briefly describes the EEPROM settings (for more detail, refer to the appropriate device data sheet or programmers guide listed in Appendix C):

- Status – All ports are enabled.
- Forwarding – All ports are configured to support store and forward.
- Speed – All ports are configured to support autonegotiation.
- Transmit Pacing – This feature is disabled and should be disabled under normal operating conditions.
- Duplex – All ports are configured to support autonegotiation.
- TX/RX Access – All ports are access ports.
- Maxlength – All ports can receive a maximum frame size of 1518 bytes.
- Pause – All ports are configured to support autonegotiation of pause based flow control. All ports support half duplex (collision based) flow control.
- Re-negotiation- For normal operation the **reneg** bit within the EEPROM must be a 1. Failure to set this bit may cause speed and duplex mismatches between the ThunderSWITCH and the PHY during reset.
- Port 27 (Network Management port) is the uplink routing register.
- Port Trunking is not enabled.
- Threshold aging is performed and the age time is set for 80 seconds.
- PauseTime10 & 100 values are 0010h which is equivalent to 8K bit times

- Device Node = 080028000001h. The Device Node is the source address for all pause frames transmitted by this device. (The Device Node shown above is only an example and must be changed for a production platform.)
- The following Device IDs are used so that software can uniquely identify the corresponding ThunderSWITCH device:
  - TNETX3270 Device ID = 02
  - TNETX3190 Device ID = 05
  - TNETX3151 Device ID = 06
  - TNETX3110 Device ID = 07
- New addresses are added to the lookup table without regard for the validity of the CRC; this is to increase the speed of learning addresses.
- 4 Megabytes of memory is used.
- Flow control threshold = 324 buffers. This limit allows each port to continue receiving a maximum length frame.

Table 2. TNETX3270 EEPROM Memory Map (24C02)

BASE	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0010:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0020:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0030:	30	1D	30	1D	30	1D	00	00	00	00	00	00	00	00	00	00
0040:	1B	1B	1B	00	0A	00	00	00	00	00	00	00	00	00	00	00
0050:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0060:	FF	FF	FF	07	FF	FF	FF	07	FF	FF	FF	07	FF	FF	FF	07
0070:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0080:	07	07	07	07	07	07	07	07	00	00	00	00	00	00	00	00
0090:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00A0:	00	80	00	**02**	08	00	28	00	00	01	00	00	00	00	00	00
00B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00E0:	00	15	00	0C	00	00	00	00	10	00	10	00	00	00	00	00
00F0:	44	01	00	00	00	00	00	00	DF	00	03	22	**B6**	**E7**	**53**	**65**

• • • • • • • • 1• CRC values in Table 2 are in **bold text** @ 0FC – 0FF.
• • • • • • • • 2• Device ID in Table 2 is in **bold text** @ address A3

Table 3. TNETX3190 EEPROM Memory Map (24C02)

BASE	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

```
0010: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 05 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 B5 C3 6D F6
```

• • • • • • • • 1• CRC values in Table 3 are in **bold text** @ address 0FC – 0FF.
• • • • • • • • 2• Device ID in Table 3 is in **bold text** @ address A3

Table 4. TNETX3151 EEPROM Memory Map (24C02)

```
BASE  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 00 00 00 00 00 00 00 00 30 1D 30 1D 30 1D 30 1D
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 06 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 F9 10 C6 FC
```

• • • • • • • • 1• CRC values in Table 4 are in **bold text** @ 0FC – 0FF.
• • • • • • • • 2• Device ID in Table 4 is in **bold text** @ address A3

Table 5. TNETX3110 EEPROM Memory Map (24C02)

```
BASE  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
```

```
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 07 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 AB 21 13 80
```

• • • • • • • 1• CRC values in Table 5 are in **bold text** @ 0FC – 0FF.
• • • • • • • 2• Device ID in Table 5 is in **bold text** @ address A3

Example 24C08 EEPROM Configurations for the ThunderSWITCH Devices

Table 6 through Table 9 list examples of the 24C08 EEPROM configuration. The following list briefly describes the EEPROM settings (for more detail, see the appropriate data sheet or programmers guide listed in Appendix C):

- Status

  All ports are enabled.

- Forwarding

  All ports are configured to support store and forward.

- Speed

  All ports are configured to support autonegotiation.

- Transmit Pacing

  This feature is disabled and should be disabled under normal operating conditions.

- Duplex

  All ports are configured to support autonegotiation.

- TX/RX Access Ports 0 – 23 are access ports (the portxQtag registers are used to insert a VLAN tag) and ports 24-26 are configured as non-access ports (can accept VLAN tagged frames).

- VLAN configuration (example shown for TNETX3270 only)

- Ports 0 – 5 and port 27 are assigned to VLAN3 (3h)
- Ports 6 – 13 and port 27 are assigned to VLAN71 (47h)
- Ports 14 – 27 are assigned to VLAN33 (21h)

- Maxlength

  Ports 0 – 23 will receive a maximum frame size of 1518 bytes; ports 24-26 will receive a maximum frame size of 1535 bytes (for TNETX3270 only, all others have a maximum frame size of 1518 bytes).

- Pause

  All ports are configured to support autonegotiation of pause based flow control; all ports support half duplex (collision based) flow control.

- Re-negotiation- For normal operation the **reneg** bit within the EEPROM must be a 1. Failure to set this bit may cause speed and duplex mismatches between the ThunderSWITCH and the PHY during reset.

- Port 27 (Network Management port) is the uplink routing register.

- Port Trunking is not enabled.

- Threshold aging is performed and the age time is set for 80 seconds.

- PauseTime10 & 100 values are 0010h, which is equivalent to 8K bit times

- Device Node = 080028000001h.

  The Device Node is the source address for all pause frames transmitted by this device. (The Device Node shown above is only an example and must be changed for a production platform.)

- The following Device IDs are used so that software can identify the unique ThunderSWITCH device:
  - TNETX3270 Device ID = 02
  - TNETX3190 Device ID = 05
  - TNETX3151 Device ID = 06
  - TNETX3110 Device ID = 07

- New addresses are added to the lookup table without regard for the validity of the CRC; this is to increase the speed of learning addresses.

- 4 Megabyte of memory is used.
- Flow control threshold = 324 buffers. This limit allows each port to continue receiving a maximum length frame.

Table 6. TNETX3270 EEPROM Memory Map (24C08)

BASE	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0010:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0020:	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D	30	1D
0030:	30	00	30	00	30	00	00	00	00	00	00	00	00	00	00	00
0040:	1B	1B	1B	00	0A	00	00	00	00	00	00	00	00	00	00	00
0050:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0060:	FF	FF	FF	07	FF	FF	FF	07	FF	FF	FF	07	FF	FF	FF	07
0070:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0080:	07	07	07	07	07	07	07	07	00	00	00	00	00	00	00	00
0090:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00A0:	00	80	00	**02**	08	00	28	00	00	01	00	00	00	00	00	00
00B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00E0:	00	15	00	0C	00	00	00	00	10	00	10	00	00	00	00	00
00F0:	44	01	00	00	00	00	00	00	DF	00	03	22	**00**	**30**	**45**	**01**
0100:	3F	00	00	08	00	C0	FF	0F	C0	3F	00	08	00	00	00	00
0110:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0120:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0130:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0140:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0150:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0160:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0170:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0180:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0190:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01A0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01B0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01C0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01D0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01E0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
01F0:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0200:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0210:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0220:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0230:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0240:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

```
0250: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0260: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0270: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0280: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0290: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0300: 03 00 21 00 47 00 00 00 00 00 00 00 00 00 00 00
0310: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0320: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0360: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0370: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0380: 03 00 03 00 03 00 03 00 03 00 03 00 47 00 47 00
0390: 47 00 47 00 47 00 47 00 47 00 47 00 21 00 21 00
03A0: 21 00 21 00 21 00 21 00 21 00 21 00 21 00 21 00
03B0: 21 00 21 00 21 00 00 00 00 00 00 00 00 00 00 00
03C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

• • • • • • • 1• CRC values in Table 6 are in **bold text** @ 0FC – 0FF.
• • • • • • • 2• Device ID in Table 6 is in **bold text** @ address A3

Table 7. TNETX3190 EEPROM Memory Map (24C08)

```
BASE  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 05 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 5B AC 76 75
0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0190: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0200: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0210: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0220: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0240: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0250: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0260: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0270: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0280: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0290: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0300: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0310: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0320: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0360: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0370: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
0380: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0390: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

• • • • • • • 1• CRC values in Table 7 are in **bold text** @ 0FC – 0FF.
• • • • • • • 2• Device ID in Table 7 is in **bold text** @ address A3

Table 8. TNETX3151 EEPROM Memory Map (24C08)

```
BASE  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 00 00 00 00 00 00 00 00 30 1D 30 1D 30 1D 30 1D
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 06 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 17 7F DD 7F
0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0190: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0200: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0210: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0220: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0240: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0250: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0260: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0270: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0280: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0290: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0300: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0310: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0320: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0360: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0370: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0380: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0390: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

• • • • • • • • 1• CRC values in Table 8 are in **bold text** @ 0FC – 0FF.
• • • • • • • • 2• Device ID in Table 8 is in **bold text** @ address A3

Table 9. TNETX3110 EEPROM Memory Map (24C08)

```
BASE  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F
0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020: 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D 30 1D
0030: 30 1D 30 1D 30 1D 00 00 00 00 00 00 00 00 00 00
0040: 1B 1B 1B 00 0A 00 00 00 00 00 00 00 00 00 00 00
0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
0060: FF FF FF 07 FF FF FF 07 FF FF FF 07 FF FF FF 07
0070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080: 07 07 07 07 07 07 07 07 00 00 00 00 00 00 00 00
0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0: 00 80 00 07 08 00 28 00 00 01 00 00 00 00 00 00
00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0: 00 15 00 0C 00 00 00 00 10 00 10 00 00 00 00 00
00F0: 44 01 00 00 00 00 00 00 DF 00 03 22 45 4E 08 03
0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0190: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0200: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0210: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0220: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0240: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0250: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0260: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0270: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0280: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0290: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0300: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0310: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
0320: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0360: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0370: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0380: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0390: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

• • • • • • • 1• CRC values in Table 9 are in **bold text** @ 0FC – 0FF.

• • • • • • • 2• Device ID in Table 9 is in **bold text** @ address A3

Conclusion

For unmanaged systems, the TNETX3270 EEPROM interface allows the designer to build a pre-configured system. Although EEPROMs are not error free, Texas Instruments provides an error detection mechanism on all ThunderSWITCH products. Remember any time new data is written to the EEPROM, a new CRC value must also be calculated (via software) and written to the EEPROM.

Appendix A. EEPROM Protocol and Handshaking Sequences

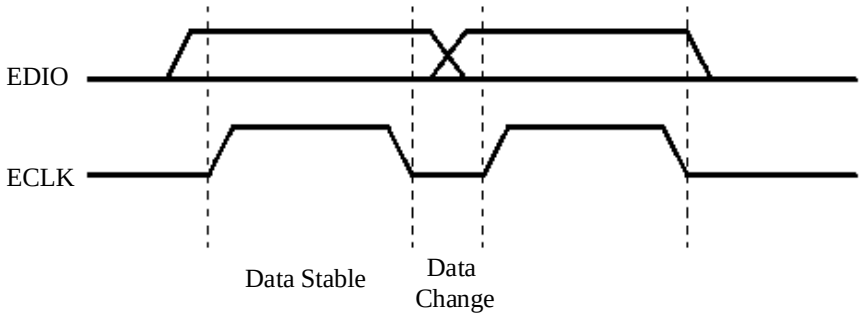
The basic protocol and handshaking sequences for reading and writing to the EEPROM via the DIO interface are shown in Figure 2 through Figure 6.

NOTE:
EDIO and ECLK pins are controlled by the ThunderSWITCH device EDATA, ETXEN, and ECLK Serial Interface I/O Register, SIO.

Valid Data

Valid data (EDIO) can change only when ECLK is low (see Figure 2). Changing the EDIO signal when ECLK is high indicates a start or stop condition.

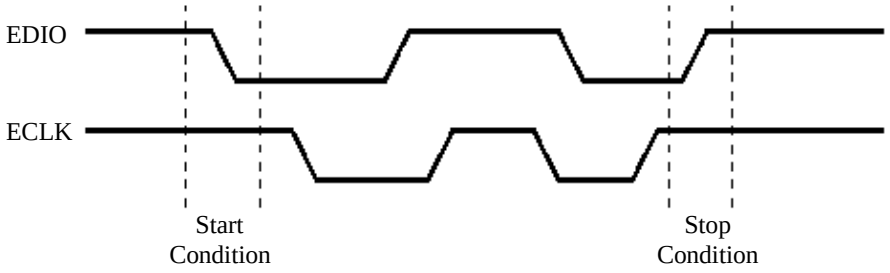
Figure 2. EEPROM Valid Data Timing Diagram



Start and Stop States

A high-to-low transition of EDIO while ECLK is high represents a start-up condition. This is followed by a 4-bit device code and a 3-bit device address (000b). A stop condition is indicated by a low-to-high transition of EDIO while ECLK is high; this is preceded by a data read or write. Figure 3 illustrates the start and stop conditions.

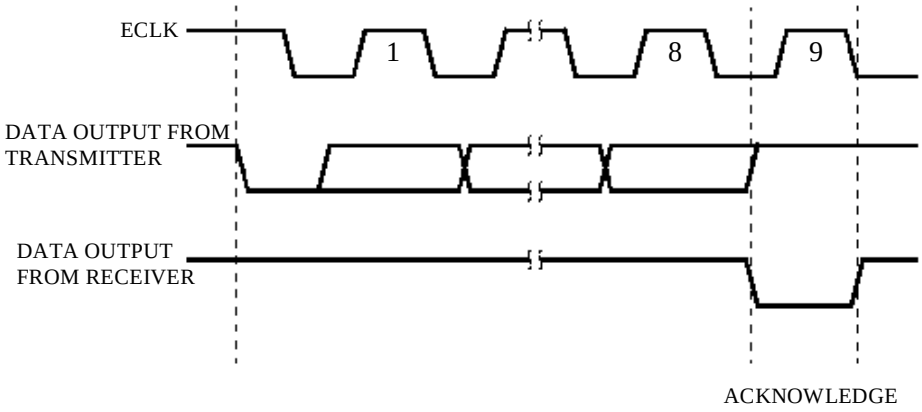
Figure 3. Start-up and Stop Bits Timing Diagram



Data Transfer Acknowledge

Another important handshake signal is the acknowledge signal, which indicates a successful data transfer. The TNETX3270, TNETX3190, TNETX3151, or TNETX3110 releases the bus after transmitting eight bits. During the ninth clock cycle, the EEPROM pulls EDIO low. Figure 4 illustrates the acknowledge sequence.

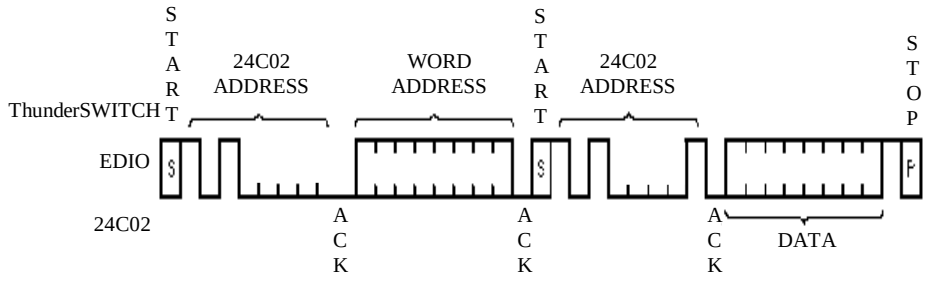
Figure 4. EEPROM Acknowledge Response Timing Diagram



EEPROM Read States

Figure 5 illustrates an EEPROM data read sequence. Example 1 shows the code providing this sequence.

Figure 5. EEPROM Data Read Timing Diagram



Example 1. Code for EEPROM Data Read Sequence

```
//---------------------------------------------------------------------
-
//  TS_EeRdByte() - read byte of data from eeprom (see 24C02
//                  device specification)
//
//  Parameters:
//    IN
//       addr           WORD  Address on EEPROM to read from
//       len            WORD  Number of bytes to read
//       pdata          *unsigned char  Data read
//
//  Return val:
//       success        int     1
//       failure        int     0
//---------------------------------------------------------------------
-
int TS_EeRdByte( WORD addr, WORD len, unsigned char *pdata )
{
    fnDioRd = &TS_DioRdByte;
    fnDioWr = &TS_DioWrByte;
    fnDioWrIm = &TS_DioWrByteIm;
    AddrSio = SIO_XCTRL;

    while( len-- )
        if( !My_EeRdByte( addr++, pdata++) )
            return(0);
    return(1);
}
//---------------------------------------------------------------------
-
//  My_EeRdByte() - read byte of data from eeprom (see Exel XL24C02
//                  device specification)
///
```

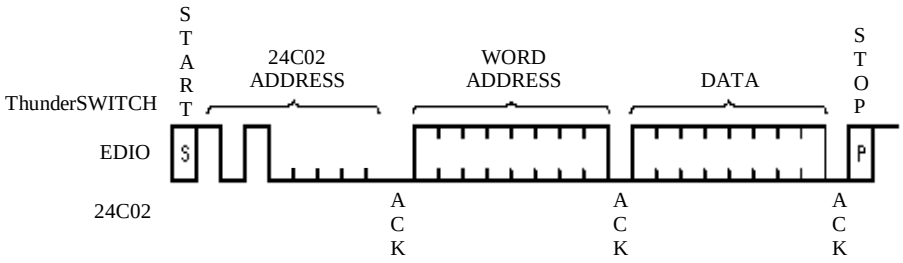
```
//  Parameters:
//    IN
//       addr           WORD  Address on EEPROM to read from
//       data           *BYTE Data read
//
//  Return val:
//    int               1 = read successful, 0 otherwise
//---------------------------------------------------------------------
-
static int My_EeRdByte( WORD addr, BYTE *data )
{
   BYTE  sio, i, temp = 0x00;

   (*fnDioRd)(AddrSio, 1, &sio);

   // send eeprom start sequence

   set(ECLOK)
   set(EDATA)
   set(ETXEN)
   clr(EDATA)
   clr(ECLOK)

   // select EEPROM device to use and set up to write address to read
   // from out to device
   TS_EeSel(WRITE);

   // if the EEPROM doesn't ACK, something's wrong
   if (!TS_EeAck())
      return (0);

   // Send address in EEPROM to read from
   TS_EeAddr(addr);

   // if the EEPROM doesn't ACK, something's wrong
   if (!TS_EeAck())
      return (0);

        // The Sel, Ack and Addr routines have changed what's in SIO_XCTRL
   (*fnDioRd)(AddrSio, 1, &sio);

   // send start access sequence for the read
   set(ETXEN)
   set(ECLOK)
   set(EDATA)
   set(ETXEN)
   clr(EDATA)
   clr(ECLOK)

   // send EEPROM device selection sequence and set up for read
   TS_EeSel(READ);

   if (!TS_EeAck())
      return(0);
```

```
   *data = 0;
   (*fnDioRd)(AddrSio, 1, &sio);

   for ( i = 0x80; i; i >>= 1 )
                                                                    {
      set(ECLOK)

      (*fnDioRd)(AddrSio, 1, &temp);
      if (temp & EDATA)
         *data |= i;

      clr(ECLOK)
   }

   tog(ECLOK)

   // send stop access sequence
   clr(EDATA)
   set(ECLOK)
   set(EDATA)
   clr(ETXEN)

   return(1);
}
```

EEPROM Write States

Figure 6 shows an EEPROM data write sequence. Example 2 shows the code providing this sequence.

Figure 6. EEPROM Data Write Timing Diagram



Example 2. Code for EEPROM Data Write Sequence

```
//---------------------------------------------------------------------
-
//  TS_EeWrByte() -  write a byte to the EEPROM
//
//  Parameters
//       addr           WORD   address to write to on EEPROM
//       len            WORD   number of bytes to write
//       data           *unsigned char  data to write
//
//  Return Value
//       success        int     1
//       failure        int     0
//---------------------------------------------------------------------
-
int TS_EeWrByte( WORD addr, WORD len, unsigned char *pdata )
{
    fnDioRd = &TS_DioRdByte;
    fnDioWr = &TS_DioWrByte;
    fnDioWrIm = &TS_DioWrByteIm;
    AddrSio = SIO_XCTRL;
    while( len-- )
        if( !My_EeWrByte( addr++, *pdata++ ) )
            return(0);
    return(1);
}

//---------------------------------------------------------------------
-
//  My_EeWrByte() -  write a byte to the EEPROM
//
//  Parameters
//     addr          WORD  address to write to on EEPROM
//     data          BYTE  data to write
//
//  Return Value
//     int           1 on success, 0 otherwise
//---------------------------------------------------------------------
-
static int My_EeWrByte( WORD addr, BYTE data)
{
   BYTE  sio, i;

   (*fnDioRd)(AddrSio, 1, &sio);

   // send eeprom start sequence

   set(ECLOK)
   set(EDATA)
   set(ETXEN)
   clr(EDATA)
   clr(ECLOK)

   // select EEPROM device to use and set up to write address to read
```

```
   // from out to device
   TS_EeSel(WRITE);

   // if the EEPROM doesn't ACK, something's wrong
   if (!TS_EeAck())
      return (0);

   // Send address in EEPROM to read from
   TS_EeAddr(addr);

   // if the EEPROM doesn't ACK, something's wrong
   if (!TS_EeAck())
      return (0);

                  // The Ack, Sel and Addr routines have changed SIO_XCTRL
   (*fnDioRd)(AddrSio, 1, &sio);
   set(ETXEN)

   for (i = 0x80; i; i >>= 1)
   {
      if ( i & data )
      set(EDATA)
      else
         clr(EDATA)

      tog(ECLOK)
   }

   if ( !TS_EeAck() )
      return (0);

   (*fnDioRd)(AddrSio, 1, &sio);

   // send stop access sequence
   set(ETXEN)
   clr(EDATA)
   set(ECLOK)
   set(EDATA)
   clr(ETXEN)

   // wait until eeprom writes the data
   do
   {
      (*fnDioRd)(AddrSio, 1, &sio);
// start sequence
      set(ECLOK)
      set(EDATA)
      set(ETXEN)
      clr(EDATA)
      clr(ECLOK)
      TS_EeSel(WRITE);
   } while ( !TS_EeAck() );
}
```

Appendix B. CRC Subroutine

The following subroutine calculates the CRC value that will be stored in the TNETX3270, TNETX3190, TNETX3151, and TNETX3110 EEPROM. The CRC is calculated bit-wise, starting with the MSB and ending with the LSB. The most significant CRC byte should be placed at address 0FC and the least significant CRC byte should be placed at address 0FF. Once calculated, the CRC value is written to the EEPROM as shown in Table 10

Table 10. CRC Generator Program

```
//*******************************************************
// Program: crc_gen
//
// Description:  CRC generator program
//
// This program returns a pointer to a CRC result stored
// in the order it should be transferred on the network
// (least significant nibble first).
//
// Here is an example:
//
//      Calculated CRC: 58cea5b6
//      Transferred on network as: 6b5aec85
//      Stored in memory as:
//
//         crc_result[0] = 0xb6
//         crc_result[1] = 0xa5
//         crc_result[2] = 0xce
//         crc_result[3] = 0x58
//*******************************************************
#include <stdio.h>
#include <string.h>
#include <math.h>

typedef unsigned char  BYTE;  // this better be 8 bits in width
typedef unsigned short WORD;  // this better be 16 bits in width

#define AND(a,b)    ((a)&&(b))
#define XOR(a,b)    ((a)^=(b))

int fb_reg[32];     // 32-bit feedback shift register
int cntl_signal;    // Control signal
int input_signal;   // Input signal

void init_regs()
{
   int i;

   for(i=0;i<32;++i) fb_reg[i] = 1; // initialization logic
}

void shift_feedback_regs()
{
   int I, and_cntl;

   and_cntl = AND(cntl_signal, XOR(fb_reg[31],input_signal));
```

```
    for(i=31; i>=27; --i)
        fb_reg[i] = fb_reg[i-1];
    fb_reg[26] = XOR(fb_reg[25],and_cntl);
    for(i=25; i>=24; --i)
        fb_reg[i] = fb_reg[i-1];
    for(i=23; i>=22; --i)
        fb_reg[i] = XOR(fb_reg[i-1],and_cntl);
    for(i=21; i>=17; --i)
        fb_reg[i] = fb_reg[i-1];
    fb_reg[16] = XOR(fb_reg[15],and_cntl);
    for(i=15; i>=13; --i)
        fb_reg[i] = fb_reg[i-1];
    for(i=12; i>=10; --i)
        fb_reg[i] = XOR(fb_reg[i-1],and_cntl);
    fb_reg[9] = fb_reg[8];
    for(i= 8; i>= 7; --i)
        fb_reg[i] = XOR(fb_reg[i-1],and_cntl);
    fb_reg[6] = fb_reg[5];
    for(i= 5; i>= 4; --i)
        fb_reg[i] = XOR(fb_reg[i-1],and_cntl);
    fb_reg[3] = fb_reg[2];
    for(i= 2; i>= 1; --i)
        fb_reg[i] = XOR(fb_reg[i-1],and_cntl);
    fb_reg[0] = and_cntl;
}

BYTE *crc_gen(BYTE *data, int length)
{
    int          crc[32], i, j, k;
    static BYTE  crc_result[4];
    BYTE         tmpdata;
    int          bits[32];

    cntl_signal = 1;    // Set the Control input to 1
    init_regs();        // Preload shift registers to all 1's

    // Initialize result
    for (i=0; i<4; i++) crc_result[i]=0x00;

    i=0;
    while (i < length)
    {
        for (j=0;j<4;j++)
        {
            if (i >= length) cntl_signal=0;
            tmpdata = *(data+i);
            for (k=0;k<8;k++)
            {
                bits[j*8+k] = tmpdata&0x01;
                tmpdata >>= 1;
            }
            i++;
        }
        for(j=0;j<=31;++j)
        {
            if(bits[j] >= 0)
            {
                input_signal =bits[j];
                shift_feedback_regs();     // Shift the registers
            }
        }
```

```
   }

   // Set Control to 0, when all the frame has been shifted in
   cntl_signal = 0;

   // Invert the output of fb_reg[31]
   crc[0] = !fb_reg[31];

   for(i=1;i<=31;++i)
   {
      shift_feedback_regs();
      crc[i] = !fb_reg[31];
   }

   // Put result in memory
   for (i=0;i<4;i++)
   {
      for (j=0; j<8; j++)
      {
         crc[i*8+j] <<= j;
         crc_result[i] |= crc[i*8+j];
      }
   }
   return (crc_result);
}
```

Appendix C. Related Documentation

Table 11 lists specific product names, part numbers, and their documentation's literature number.

Table 11. Related Documentation

Product Name	Part Number	Literature Number
Desktop TNETX3270 Reference Design Kit	TNETX3270EVM	N/A
TNETX3270 Data Sheet	N/A	SPWS043
TNETX3270 Programmers Guide	N/A	SPAU002
TNETX3190 Data Sheet	N/A	Not available at time of printing
TNETX3190 Programmers Guide	N/A	Not available at time of printing
TNETX3151 Data Sheet	N/A	Not available at time of printing
TNETX3151 Programmers Guide	N/A	Not available at time of printing
TNETX3110 Data Sheet	N/A	Not available at time of printing
TNETX3110 Programmers Guide	N/A	Not available at time of printing

World Wide Web Support

Our world wide web site at **www.ti.com**. contains the most up to date product information, revisions, and additions. New users must register with TI&ME before they can access the data sheet archive. TI&ME allows users to build custom information pages and receive automatic new product updates via email.

Email Support

For technical issues or clarification on switching products, please send a detailed email to **networks@ti.com**. Questions receive prompt attention and are usually answered within one business day.