
**Programming Specification for D3 ASIC’s Series**

# ASIC Configuration Memories

## Serial Bus Overview

The serial bus is a 2-wire bus. One wire (CLOCK) functions as a clock and is provided by the programmer, the second wire (DATA) is a bi-directional signal and is used to provide data and control information.

Information is transmitted on the serial bus in messages. Each MESSAGE is pre- ceded by a Start Condition and is ended with a Stop Condition. The message consists of an integer number of bytes, each byte consisting of 8 bits of data, followed by a 9th Acknowledge Bit. This Acknowledge Bit is provided by the recipient of the transmitted byte. This is possible because devices may only drive the DATA line Low. The system must provide a small pull-up current for the DATA line.

The MESSAGE FORMAT for read and write instructions consists of the bytes shown in “Bit Format” on page 2.

While writing, the programmer is responsible for issuing the instruction and data. While reading, the programmer issues the instruction and acknowledges the data from the Configurator as necessary.

Again, the Acknowledge Bit is asserted on the DATA line by the receiving device on a byte-by-byte basis.

The factory blanks devices to all zeros before shipping. The array cannot otherwise be “initialized” except by explicitly writing a known value to each location using the serial protocol described herein.

## Bit Format

Data on the DATA pin may change only during the CLOCK Low time; whereas Start and Stop Conditions are identified as transitions during the CLOCK High time.

### Write Instruction Message Format

ACK BIT
(CONFIGURATOR)

**STOP CONDITION**

**DATA BYTE n**

### Current Address Read (Extended to Sequential Read) Instruction Message Format

| START CONDITION | DEVICE ADDRESS | DATA BYTE 1 |
|---|---|---|

**ACK BIT (CONFIGURATOR)** **ACK BIT (PROGRAMMER)**

## Start and Stop Conditions

The Start Condition is indicated by a high-to-low transition of the DATA line when the CLOCK line is High. Similarly, the Stop Condition is generated by a low-to-high transi- tion of the DATA line when the CLOCK line is High, see Figure 1.

The Start Condition will return the device to the state where it is waiting for a Device Address (its normal quiescent mode).

The Stop Condition initiates an internally timed write signal whose maximum duration is $t_{WR}$ (refer to the AC Characteristics tables for actual value). During this time, the Config- urator must remain in programming mode (i.e., SER_EN is driven Low). DATA and CLOCK lines are ignored until the cycle is completed. Since the write cycle typically completes in less than $t_{WR}$ seconds, we recommend the use of “polling” as described in page 8. Input levels to all other pins should be held constant until the write cycle has been completed.

## Acknowledge Bit

The Acknowledge (ACK) Bit shown in Figure 1 is provided by the Configurator receiving the byte. The receiving Configurator can accept the byte by asserting a Low value on the DATA line, or it can refuse the byte by asserting (allowing the signal to be externally pulled up to) a High value on the DATA line. All bytes from accepted messages must be terminated by either an Acknowledge Bit or a Stop Condition. Following an ACK Bit, when the DATA line is released during an exchange of control between the Configurator and the Programmer, the DATA line may be pulled High temporarily as shown above due to the open-collector output nature of the line. Control of the line must resume before the next rising edge of the clock.

## Bit Ordering Protocol

The most significant bit is the first bit of a byte transmitted on the DATA line for the Device Address Byte and the EEPROM Address Bytes. It is followed by the lesser sig- nificant bits until the eighth bit, the least significant bit, is transmitted. However, for Data Bytes (both writing and reading), the first bit transmitted is the least significant bit. This protocol is shown in "Device Address Byte" and "EEPROM Address".

## Device Address Byte

The contents of the Device Address Byte are shown below, along with the order in which the bits are clocked into the device.

The CE pin cannot be used for device selection in programming mode (i.e., when SER_EN is drive Low).

**Device Address Byte**

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | R/W |
| 1st | 2nd | 3rd | 4th | 5th | 6th | 7th | 8th |

Where:R/W = 1 Read
= 0 Write

## EEPROM Address

The EEPROM Address consists of two bytes on the 256-Kbit parts, and three bytes on the 512-Kbit, 1- and 2- and 4-Mbit parts. Each Address Byte is followed by an Acknowl- edge Bit (provided by the Configurator). These bytes define the normal address space of the Configurator, as described below. The order in which each byte is clocked into the Configurator is also indicated. Unused bits in an Address Byte must be set to “0”. Excep- tions to this are:

- when setting the reset polarity;
- when reading Device and Manufacturer Codes.

2- and 4-Mbit Page Length

| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | $A_{E18}$ | $A_{E17}$ | $A_{E16}$ | ACK | $A_{E15}$ | $A_{E14}$ | $A_{E13}$ | $A_{E12}$ | $A_{E11}$ | $A_{E10}$ | $A_{E9}$ | $A_{E8}$ | ACK | $A_{E7}$ | $A_{E6}$ | $A_{E5}$ | $A_{E4}$ | $A_{E3}$ | $A_{E2}$ | $A_{E1}$ | $A_{E0}$ ACK |

## Programming Summary: Write to Whole Device

Notes: 1. Pull-up resistor required on DATA line

- Data byte received/sent LSB to MSB
- These signals have “don’t care” conditions for the D3 ASIC 512/010/002 and D3 ASIC 040.

- The 512-Kbit, and 1-, 2- and 4-Mbit parts require three EEPROM address bytes; all three bytes must be individually ACK'd by the EEPROM.

## EEPROM Address is Defined as:

| | | | | | | |
|---|---|---|---|---|---|---|
| 256 | | | $0X_8X_7X_6$ | $X_5X_4X_3X_2$ | $X_1X_000$ | 0000 |
| 512 | 0000 | 0000 | $X_8X_7X_6X_5$ | $X_4X_3X_2X_1$ | $X_0000$ | 0000 |
| 010 | 0000 | $000X_9$ | $X_8X_7X_6X_5$ | $X_4X_3X_2X_1$ | $X_0000$ | 0000 |
| 002 | 0000 | $00X_{9X8}$ | $X_7X_6X_5X_4$ | $X_3X_2X_1X_0$ | 0000 | 0000 |
| 040 | 000 | $0X_{10X9X8}$ | $X_7X_6X_5X_4$ | $X_3X_2X_1X_0$ | 0000 | 0000 |

Note: 1. where $X_n$ ... $X_0$ is (PAGE_COUNT)\b where $X_n$ ... $X_0$ is (PAGE_COUNT)\b

## T_BYTE Per Page

| | |
|---|---|
| D3 ASIC 256 | 64 |
| D3 ASIC 512/010 | 128 |
| D3 ASIC 002 | 256 |
| D3 ASIC 040 | 256 |

## T_PAGE

| | |
|---|---|
| D3 ASIC 256 | 512 |
| D3 ASIC 512 | 512 |
| D3 ASIC 010 | 1024 |
| D3 ASIC 002 | 1024 |
| D3 ASIC 040 | 2048 |

**SER_EN ≤ Low CE ≤ Low[3]**
**RESET/OE ≤ Low[3]**

# Programming Summary: Read from Whole Device

Notes: 1. Pull-up resistor required on DATA line
- Data byte received/sent LSB to MSB
- These signals have "don't care" conditions for the D3 ASIC 512/010/002 and D3 ASIC 040.
- The 512-Kbit, and 1-, 2- and 4-Mbit parts require three EEPROM address bytes; all three bytes must be individually ACK'd by the EEPROM.

## EEPROM Address is Defined as:

| | |
|---|---|
| 256 | 00 00 \h |
| 512/010/002/040 | 00 00 00 \h |

## TT_BYTE

| | |
|---|---|
| D3 ASIC 256 | 32768 \d |
| D3 ASIC 512 | 65536 \d |
| D3 ASIC 010 | 131072 \d |
| D3 ASIC 002 | 262144 \d |
| D3 ASIC 040 | 524288 \d |

# Programming Summary: Write Reset Polarity

Notes: 1. Pull-up resistor required on DATA line
- Data byte received/sent LSB to MSB
- These signals have "don't care" conditions for the D3 ASIC 512/010/002 and D3 ASIC 040.
- The 512-Kbit, and 1-, 2- and 4-Mbit parts require three EEPROM address bytes; all three bytes must be individually ACK'd by the EEPROM.
- Drive RESET/OE high for active low RESET, active high OE. Drive RESET/OE low for active high RESET, active low OE.
- The 512-Kbit, and 1-, 2- and 4-Mbit parts require four data bytes of the same value to program the reset polarity; all four bytes must be individually ACK'd by the EEPROM.

## EEPROM Address is Defined as:

| | |
|---|---|
| D3 ASIC 256 | 3F FF \h |
| D3 ASIC 512/010 | 02 00 00 \h |

| | |
|---|---|
| D3 ASIC 002 | 400 000 \h |
| D3 ASIC 040 | 400 000 \h |

## Data Byte

The organization of the Data Byte is shown below. Note that in this case, the Data Byte is clocked into the device LSB first and MSB last.

## Writing

Writing to the normal address space takes place in pages. A page is 64 bytes long in 256-Kbit parts; 128 bytes long in 512-Kbit and 1-Mbit parts, and 256 bytes long in the 2- and 4-Mbit parts. The page boundaries are, respectively, addresses where $A_{E6}$ down to $A_{EOS}$ are all zero, and $A_{E6}$ down to $A_{E0}$ are all zero. Writing can start at any address within a page and the number of bytes written must be 64 for the 256-Kbit parts, 128 for the 512-Kbit and 1-Mbit parts, and 256 for the 2- and 4-Mbit parts. The first byte is writ- ten at the transmitted address. The address is incremented in the Configurator following the receipt of each Data Byte. Only the lower bits of the address (6, 7 or 8, depending on the page length) are incremented. Thus, after writing to the last byte address within the given page, the address will roll over to the first byte address of the same page.

**Data Byte**

LSB MSB

| D0 | D1 | D2 | D3 ASIC | D4 | D5 | D6 | D7 |
|---|---|---|---|---|---|---|---|
| 1st | 2nd | 3rd | 4th | 5th | 6th | 7th | 8th |

```
A Write Instruction consists of a Start
    Condition
    a Device Address Byte with R/W = 0
      An Acknowledge Bit from the Configurator
    MS Byte of the EEPROM Address (512-Kbit, and 1-, 2- and 4-Mbit parts only)
      An Acknowledge Bit from the Configurator (Next)
    Byte of the EEPROM Address
      An Acknowledge Bit from the Configurator LS Byte
    of EEPROM Address
      An Acknowledge Bit from the Configurator
    One or more Data Bytes (sent to the Configurator)
      Each followed by an Acknowledge Bit from the Configurator a Stop
    Condition
```

**WRITE POLLING:** On receipt of the Stop Condition, the Configurator enters an inter- nally-timed write cycle. While the Configurator is busy with this write cycle, it will not acknowledge any transfers. The programmer can start the next page write by sending the Start Condition followed by the Device Address, in effect polling the Configurator. If this is not acknowledged, then the programmer should abandon the transfer without asserting a Stop Condition. The programmer can then repeatedly initiate a write instruc- tion as above, until an acknowledge is received. When the Acknowledge Bit is received, the write instruction should continue by sending the first EEPROM Address Byte to the Configurator.

An alternative to write polling would be to wait a period of $t_{WR}$ before sending the next page of data or exiting the programming mode. All signals must be maintained during the entire write cycle.

## Reading

Read instructions are initiated similarly to write instructions, but the R/W bit in the Device Address is set to one. There are three variants of the read instruction: current address read, random read and sequential read.

For all reads, it is important to understand that the internal Data Byte address counter maintains the last address accessed during the previous read or write operation, incre- mented by one. This address remains valid between operations as long as the chip power is maintained and the device remains in 2-wire access mode (i.e., SER_EN is driven Low). If the last operation was a read at address *n*, then the current address would be *n* + 1. If the final operation was a write at address *n*, then the current address would again be *n* + 1 with one exception. If address *n* was the last byte address in the page, the incremented address *n* + 1 would “roll over” to the first byte address on the next page.

**CURRENT ADDRESS READ:** Once the Device Address (with the R/W select bit set to High) is clocked in and acknowledged by the Configurator, the Data Byte at the current address is serially clocked out by the Configurator in response to the clock from the pro- grammer. The programmer generates a Stop Condition to accept the single byte of data and terminate the read instruction.

```
A Current Address Read instruction consists of a Start
    Condition
    a Device Address with R/W = 1
      An Acknowledge Bit from the Configurator a Data
    Byte from the Configurator
    a Stop Condition from the programmer.
```

**RANDOM READ:** A Random Read is a Current Address Read preceded by an aborted write instruction. The write instruction is only initiated for the purpose of loading the EEPROM Address Bytes. Once the Device Address Byte and the EEPROM Address Bytes are clocked in and acknowledged by the Configurator, the programmer immedi- ately initiates a Current Address Read.

```
A Random Address Read instruction consists of a Start
  Condition
  a Device Address with R/W = 0
    An Acknowledge Bit from the Configurator
  MS Byte of the EEPROM Address (512-Kbit, and 1-, 2- and 4-Mbit parts only)
    An Acknowledge Bit from the Configurator (Next)
  Byte of the EEPROM Address
    An Acknowledge Bit from the Configurator LS Byte
  of EEPROM Address
    An Acknowledge bit from the Configurator a Start
  Condition
  a Device Address with R/W = 1
    An Acknowledge Bit from the Configurator a Data
  Byte from the Configurator
  a Stop Condition from the programmer.
```

**SEQUENTIAL READ:** Sequential Reads follow either a Current Address Read or a Random Address Read. After the programmer receives a Data Byte, it may respond with an Acknowledge Bit. As long as the Configurator receives an Acknowledge Bit, it will continue to increment the Data Byte address and serially clock out sequential Data Bytes until the memory address limit is reached. The Sequential Read instruction is ter- minated when the programmer does not respond with an Acknowledge Bit, but instead generates a Stop Condition following the receipt of a Data Byte.

## Programmer Functions

The following programmer functions are supported while the Configurator is in program- ming mode (i.e., when SER_EN is driven Low):

- Reading the Manufacturer’s Code and the Device Code.
- Programing the device.
- Verifying the device.
- Setting the Reset Polarity option.

In the order given above, they are performed in the following manner. The same proto- col and operations are used for both 5V and 3.3V devices, as well as for the Altera pinout variants except where stated.

**Reading Manufacturer's and Device Codes**

The D3 ASIC 512/010/002 Configurators use a different algorithm than the D3 ASIC 256 Configurators, the sequential reading of these bytes are accomplished by performing a Random Read at EEPROM Address 040000H.

On D3 ASIC 002/040 Configurators, the sequential reading of these bytes are accom- plished by performing a Random Read at EEPROM Address 100000H.

On D3 ASIC 256 Configurators, the sequential read is done at EEPROM Address 0 by performing a Current Address Read with the following additional DC voltages set:

```
RESET/OE     = 0V

CE           = 11.5 ± 0.5V
```

The correct codes are[1]:

```
Manufacturers Code - Byte 0 1E   All Device Code    - Byte 1 77    D3 ASIC 256
                                                     37    D3 ASIC 512
                                                     F7    D3 ASIC 010
                                                     78    D3 ASIC 002
                                                     74    D3 ASIC 040
```

Note: 1. The Manufacturer's Code and Device Code are read using the byte ordering specified for Data Bytes; i.e., LSB first, MSB last. These procedures are not supported by the supplied ISP reference design schematics for D3 ASIC 65/128/256(A) Configurators.
After reading the manufacturer identification bytes, a hardware power cycle (power on reset) is required in order to access the actual location of the memory in the program- ming mode. For the D3 ASIC series devices, toggling the SER_EN pin from Low to High and the Low with respect to the programming clock cycle will automatically exit the manufacturer identification read mode without power cycle.

**Programming the Device**

All the bytes in a given page must be written. The page access order is not important but it is suggested that the Configurator be written sequentially from address 0. Writing is accomplished by using the DATA and CLOCK pins.

For the D3 ASIC 256 Configurators only, two additional programming pins must be set as follows:

```
RESET/OE (OE)= 0V (Write protection disable) CE (nCS)
             = 0V
```

*Important Note on D3 ASIC Series Configurators Programming*

The first byte of data will not be cached for read back during ASIC Configuration (i.e., when SER_EN is driven High) until the Configurator is power-cycled. This may be criti- cal in cascaded ISP applications where the first byte of the second or subsequent EEPROM is likely to change between updated bitstreams.

**Verifying the Device**

All bytes in the Configurator should be read and compared to their intended values.
Reading is done using the CLOCK and DATA pins.

For the D3 ASIC 256 Configurator, two additional programming pins must be set as follows:

```
RESET/OE (OE)= 0V (Write protection disable) CE (nCS)
             = 0V
```

**RESET Polarity Option**

All Configurators in the D3 ASIC Series have the ability to change the polarity of the RESET/OE pin. This is required to allow the devices to properly configure various ASIC families. The default condition is active Low OE and active High RESET.

The D3 ASIC 256 Configurators use a different algorithm from the D3 ASIC 512/010/002 Configurators; the algorithms are described below.

*D3 ASIC 256 Configurator RESET/OE Polarity Programming*

Setting the polarity option active High OE (active Low RESET): Write Data Byte “FF” to address 3FFFH, with two additional programming pins set to the following:

RESET/OE (OE)= Vcc +/- 0.25V CE (nCS) = Vcc +/- 0.25V

Setting the polarity option active Low OE (active High RESET): Write a byte “FF” to address 3FFFH, with two additional programming pins set to the following:

RESET/OE (OE)= 0V

CE (nCS) = Vcc +/- 0.25V

Verifying the RESET Polarity: Power up the device with:

RESET/OE (OE)= 0V CE = 0V

SER_EN = Vcc +/- 0.25V CLK (DCLK) = 0V

DATA = Input to programmer

In this condition, if the DATA pin is tri-stated, then the RESET/OE (OE) fuse is pro- grammed for active High OE (active Low RESET); if the DATA pin reads a “0” or a “1”, the RESET/OE (OE) fuse is active Low OE (active High RESET).

*D3 ASIC 512/010 Configurator RESET/OE (OE) Polarity Programming*

Setting the polarity option active High OE (active Low RESET): Write four bytes “FF FF FF FF” to addresses 20000H - 20003H.

Setting the polarity option active Low OE (active High RESET): Write four bytes “00 00 00 00” to addresses 20000H - 20003H.

Verifying the RESET/OE Polarity D3 ASIC 512/010 Configurators: Perform a Random Read of four Data Bytes from addresses 20000H - 20003H. If the data is “00 00 00 00” then the fuse is programmed for active Low OE (active High RESET); if the data is “FF FF FF FF” then the fuse is programmed for active High OE (active Low RESET).

*D3 ASIC 002/040 Configurator RESET/OE Polarity Programming*

Setting the polarity option active High OE (active Low RESET): Write four bytes “FF FF FF FF” to addresses 400000H - 400003H.

Setting the polarity option active Low OE (active High RESET): Write four bytes “00 00 00 00” to addresses 400000H - 400003H.

Verifying the RESET/OE Polarity D3 ASIC 002/040 Configurators: Perform a Random Read of four Data Bytes from addresses 400000H - 400003H. If the data is “00 00 00 00” then the fuse is programmed for active Low OE (active High RESET); if the data is “FF FF FF FF” then the fuse ISP programmed for active High OE (ACTIVE LOW RESET).

## DC Characteristics in Programming Mode (SER_EN)

$V_{CC}$ = 3.3V - 5V ± 10%

| Symbol | Parameter | Test Condition | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{CC}$ | Supply Voltage | | 3.0 | 4.13 | 5.25 | V |
| $I_{CC}$ | Supply Current | $V_{CC}$ = 3.6 | | 2.0 | 5.0 | mA |
| $I_{LL}$ | Input Leakage Current | $V_{IN}$ = $V_{CC}$ or $V_{SS}$ | | 0.10 | 10 | µA |
| $I_{LO}$ | Output Leakage Current | $V_{OUT}$ = $V_{CC}$ or $V_{SS}$ | | 0.05 | 10 | µA |
| $V_{IH}$ | High-level Input Voltage | | $V_{CC}$ x 0.7 | | $V_{CC}$ + 0.5 | V |
| $V_{IL}$ | Low-level Input Voltage | | -0.5 | | 0.4 | V |
| $V_{OL}$ | Output Low-level Voltage | $I_{OL}$ = 2.1 mA | | | 0.4 | V |

## AC Characteristics

$V_{CC}$ = 3.3V - 5V ± 10%

| Symbol | Parameter | Min | Max | Units |
|---|---|---|---|---|
| $f_{CLOCK}$ | Clock Frequency, Clock | | 400 | KHz |
| $t_{LOW}$ | Clock Pulse Width Low | 1.2 | | µs |
| $t_{HIGH}$ | Clock Pulse Width High | 1.2 | | µs |
| $t_{AA}$ | Clock Low to Data Out Valid | | 0.9 | µs |
| $t_{BUF}$ | Time the bus must be free before a new transmission can start | 1.2 | | µs |
| $t_{HD;STA}$ | Start Hold Time | 0.6 | | µs |
| $t_{SU;STA}$ | Start Setup Time | 0.6 | | µs |
| $t_{HD\ DAT}$ | Data In Hold Time | 0.1 | | µs |
| $t_{SU\ DAT}$ | Data In Setup Time | 0.1 | | µs |
| $t_R$ | Inputs Rise Time | | 0.3 | µs |
| $t_F$ | Inputs Fall Time | | 0.3 | µs |
| $t_{SU\ STO}$ | Stop Setup Time | 0.6 | | µs |
| $t_{DH}$ | Data Out Hold Time | 0 | | µs |
| $t_{WR}$ | Write Cycle Time | | 25 | ms |