Quad-Serial Configuration (EPCQ) Devices Datasheet

This datasheet describes quad-serial configuration (EPCQ) devices.

Supported Devices

Table 1 lists the supported Altera® EPCQ devices.

Table 1. Altera EPCQ Devices

Device	Memory Size (bits)	On-Chip Decompression Support	ISP Support	Cascading Support	Reprogrammable	Recommended Operating Voltage (V)
EPCQ16	16,777,216	No	Yes	No	Yes	3.3
EPCQ32	33,554,432	No	Yes	No	Yes	3.3
EPCQ64	67,108,864	No	Yes	No	Yes	3.3
EPCQ128	134,217,728	No	Yes	No	Yes	3.3
EPCQ256	268,435,456	No	Yes	No	Yes	3.3

Features

EPCQ devices offer the following features:

- Serial or quad-serial FPGA configuration in devices that support active serial (AS) x1 or AS x4 configuration schemes
- Low cost, low pin count, and non-volatile memory
- 2.7-V to 3.6-V operation
- Available in 8- and 16-pin small-outline integrated circuit (SOIC) package
- Reprogrammable memory with more than 100,000 erase or program cycles
- Write protection support for memory sectors using status register bits
- Fast read, extended dual input fast read, and extended quad input fast read of the entire memory using a single operation code
- Write bytes, extended dual input fast write bytes, and extended quad input fast write bytes of the entire memory using a single operation code
- Reprogrammable with an external microprocessor using the SRunner software driver
- In-system programming (ISP) support with the SRunner software driver
- ISP support with USB-Blaster™, EthernetBlaster II, or EthernetBlaster download cables

101 Innovation Drive
San Jose, CA 95134
www.altera.com

© 2012 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO 9001:2008 Registered

- By default, the memory array is erased and the bits are set to 1

Memory Array Organization

Table 2 lists the memory array organization in supported EPCQ devices.

Table 2. Memory Array Organization in EPCQ Devices

Details	EPCQ16	EPCQ32	EPCQ64	EPCQ128	EPCQ256
Bytes	2,097,152 bytes [16 megabits (Mb)]	4,194,304 bytes (32 Mb)	8,388,608 bytes (64 Mb)	16,777,216 bytes (128 Mb)	33,554,432 bytes (256 Mb)
Number of sectors	32	64	128	256	512
Bytes per sector	65,536 bytes [512 kilobits (Kb)]				
Total number of subsectors (1)	512	1,024	2,048	4,096	8,192
Bytes per subsector	4,096 bytes (32 Kb)				
Pages per sector	256				
Total number of pages	8,192	16,384	32,768	65,536	131,072
Bytes per page	256 bytes				

Note to Table 2:

(1) Every sector is further divided into 16 subsectors with 4 KB of memory. Because of this, there are 512 (32 x16) subsectors for the EPCQ16 device, 1,024 (64 x 16) subsectors for the EPCQ32 device, 2,048 (128 x 16) subsectors for the EPCQ64 device, 4,096 (256 x 16) subsectors for the EPCQ128 device, and 8,192 (512 x 16) subsectors for the EPCQ256 device.

Address Range for EPCQ16

Table 3 lists the address range for each sector in EPCQ16 devices.

Table 3. Address Range for Sectors 31..0 and Subsectors 511..0 in EPCQ16 Devices —Preliminary (Part 1 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
31	511	1FF000	1FFFFF
	510	1FE000	1FEFFF
	...	...	...
	498	1F2000	1F2FFF
	497	1F1000	1F1FFF
	496	1F0000	1F0FFF
30	495	1EF000	1EFFFF
	494	1EE000	1EEFFF
	...	...	...
	482	1E2000	1E2FFF
	481	1E1000	1E1FFF
	480	1E0000	1E0FFF

Table 3. Address Range for Sectors 31..0 and Subsectors 511..0 in EPCQ16 Devices —Preliminary (Part 2 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
1	31	1F000	1FFFF
	30	1E000	1EFFF
	...	...	...
	18	12000	12FFF
	17	11000	11FFF
	16	10000	10FFF
0	15	F000	FFFF
	14	E000	EFFF
	...	...	...
	2	2000	2FFF
	1	1000	1FFF
	0	H'0000000	H'0000FFF

Address Range for EPCQ32

Table 4 lists the address range for each sector in EPCQ32 devices.

Table 4. Address Range for Sectors 63..0 and Subsectors 1023..0 in EPCQ32 Devices —Preliminary (Part 1 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
63	1023	3FF000	3FFFFF
	1022	3FE000	3FEFFF
	...	...	...
	1010	3F2000	3F2FFF
	1009	3F1000	3F1FFF
	1008	3F0000	3F0FFF
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	...	...	...
	994	3E2000	3E2FFF
	993	3E1000	3E1FFF
	992	3E0000	3E0FFF

Table 4. Address Range for Sectors 63..0 and Subsectors 1023..0 in EPCQ32 Devices —Preliminary (Part 2 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
1	31	1F000	1FFFF
	30	1E000	1EFFF
	...	...	...
	18	12000	12FFF
	17	11000	11FFF
	16	10000	10FFF
0	15	F000	FFFF
	14	E000	EFFF
	...	...	...
	2	2000	2FFF
	1	1000	1FFF
	0	H'0000000	H'0000FFF

Address Range for EPCQ64

Table 5 lists the address range for each sector in EPCQ64 devices.

Table 5. Address Range for Sectors 127..0 and Subsectors 2047..0 in EPCQ64 Devices —Preliminary (Part 1 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
127	2047	7FF000	7FFFFF
	2046	7FE000	7FEFFF
	...	...	...
	2034	7F2000	7F2FFF
	2033	7F1000	7F1FFF
	2032	7F0000	7F0FFF
64	1039	40F000	40FFFF
	1038	40E000	40EFFF
	...	...	...
	1026	402000	402FFF
	1025	401000	401FFF
	1024	400000	400FFF

Table 5. Address Range for Sectors 127..0 and Subsectors 2047..0 in EPCQ64 Devices —Preliminary (Part 2 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
63	1023	3FF000	3FFFFF
	1022	3FE000	3FEFFF
	...	...	...
	1010	3F2000	3F2FFF
	1009	3F1000	3F1FFF
	1008	3F0000	3F0FFF
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	...	...	...
	994	3E2000	3E2FFF
	993	3E1000	3E1FFF
	992	3E0000	3E0FFF
1	31	1F000	1FFFF
	30	1E000	1EFFF
	...	...	...
	18	12000	12FFF
	17	11000	11FFF
	16	10000	10FFF
0	15	F000	FFFF
	14	E000	EFFF
	...	...	...
	2	2000	2FFF
	1	1000	1FFF
	0	H'0000000	H'0000FFF

Address Range for EPCQ128

Table 6 lists the address range for each sector in EPCQ128 devices.

Table 6. Address Range for Sectors 255..0 and Subsectors 4095..0 in EPCQ128 Devices —Preliminary (Part 1 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
255	4095	FFF000	FFFFFF
	4094	FFE000	FFEFFF
	...	...	...
	4082	FF2000	FF2FFF
	4081	FF1000	FF1FFF
	4080	FF0000	FF0FFF
254	4079	FEF000	FEFFFF
	4078	FEE000	FEEFFF
	...	...	...
	4066	FE2000	FE2FFF
	4065	FE1000	FE1FFF
	4064	FE0000	FE0FFF
129	2079	81F000	81FFFF
	2078	81E000	81EFFF
	...	...	...
	2066	812000	812FFF
	2065	811000	811FFF
	2064	810000	810FFF
128	2063	80F000	80FFFF
	2062	80E000	80EFFF
	...	...	...
	2050	802000	802FFF
	2049	801000	801FFF
	2048	800000	800FFF
127	2047	7FF000	7FFFFF
	2046	7FE000	7FEFFF
	...	...	...
	2034	7F2000	7F2FFF
	2033	7F1000	7F1FFF
	2032	7F0000	7F0FFF

Table 6. Address Range for Sectors 255..0 and Subsectors 4095..0 in EPCQ128 Devices —Preliminary (Part 2 of 2)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
64	1039	40F000	40FFFF
	1038	40E000	40EFFF
	...	...	...
	1026	402000	402FFF
	1025	401000	401FFF
	1024	400000	400FFF
63	1023	3FF000	3FFFFF
	1022	3FE000	3FEFFF
	...	...	...
	1010	3F2000	3F2FFF
	1009	3F1000	3F1FFF
	1008	3F0000	3F0FFF
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	...	...	...
	994	3E2000	3E2FFF
	993	3E1000	3E1FFF
	992	3E0000	3E0FFF
1	31	1F000	1FFFF
	30	1E000	1EFFF
	...	...	...
	18	12000	12FFF
	17	11000	11FFF
	16	10000	10FFF
0	15	F000	FFFF
	14	E000	EFFF
	...	...	...
	2	2000	2FFF
	1	1000	1FFF
	0	H'0000000	H'0000FFF

Address Range for EPCQ256

Table 7 lists the address range for each sector in EPCQ256 devices.

Table 7. Address Range for Sectors 511..0 and Subsectors 8191..0 in EPCQ256 Devices —Preliminary (Part 1 of 3)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
511	8191	1FFF000	1FFFFFF
	8190	1FFE000	1FFEFFF
	...	...	...
	8178	1FF2000	1FF2FFF
	8177	1FF1000	1FF1FFF
	8176	1FF0000	1FF0FFF
510	8175	1FEF000	1FEFFFF
	8174	1FEE000	1FEEFFF
	...	...	...
	8162	1FE2000	1FE2FFF
	8161	1FE1000	1FE1FFF
	8160	1FE0000	1FE0FFF
257	4127	101F000	101FFFF
	4126	101E000	101EFFF
	...	...	...
	4114	1012000	1012FFF
	4113	1011000	1011FFF
	4112	1010000	1010FFF
256	4111	100F000	100FFFF
	4110	100E000	100EFFF
	...	...	...
	4098	1002000	1002FFF
	4097	1001000	1001FFF
	4096	1000000	1000FFF
255	4095	FFF000	FFFFFF
	4094	FFE000	FFEFFF
	...	...	...
	4082	FF2000	FF2FFF
	4081	FF1000	FF1FFF
	4080	FF0000	FF0FFF

Table 7. Address Range for Sectors 511..0 and Subsectors 8191..0 in EPCQ256 Devices —Preliminary (Part 2 of 3)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
254	4079	FEF000	FEFFFF
	4078	FEE000	FEEFFF
	...	...	...
	4066	FE2000	FE2FFF
	4065	FE1000	FE1FFF
	4064	FE0000	FE0FFF
129	2079	81F000	81FFFF
	2078	81E000	81EFFF
	...	...	...
	2066	812000	812FFF
	2065	811000	811FFF
	2064	810000	810FFF
128	2063	80F000	80FFFF
	2062	80E000	80EFFF
	...	...	...
	2050	802000	802FFF
	2049	801000	801FFF
	2048	800000	800FFF
127	2047	7FF000	7FFFFF
	2046	7FE000	7FEFFF
	...	...	...
	2034	7F2000	7F2FFF
	2033	7F1000	7F1FFF
	2032	7F0000	7F0FFF
64	1039	40F000	40FFFF
	1038	40E000	40EFFF
	...	...	...
	1026	402000	402FFF
	1025	401000	401FFF
	1024	400000	400FFF
63	1023	3FF000	3FFFFF
	1022	3FE000	3FEFFF
	...	...	...
	1010	3F2000	3F2FFF
	1009	3F1000	3F1FFF
	1008	3F0000	3F0FFF

Table 7. Address Range for Sectors 511..0 and Subsectors 8191..0 in EPCQ256 Devices —Preliminary (Part 3 of 3)

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	...	...	...
	994	3E2000	3E2FFF
	993	3E1000	3E1FFF
	992	3E0000	3E0FFF
1	31	1F000	1FFFF
	30	1E000	1EFFF
	...	...	...
	18	12000	12FFF
	17	11000	11FFF
	16	10000	10FFF
0	15	F000	FFFF
	14	E000	EFFF
	...	...	...
	2	2000	2FFF
	1	1000	1FFF
	0	H'0000000	H'0000FFF

Memory Operations

This section describes the operations that you can use to access the memory in EPCQ devices. When performing the operation, addresses and data are shifted in and out of the device serially, with the MSB first.

Timing Requirements

When the active low chip select (nCS) signal is driven low, shift in the operation code into the EPCQ device using the serial data (DATA) pin. Each operation code bit is latched into the EPCQ device on the rising edge of the DCLK.

While executing an operation, shift in the desired operation code, followed by the address or data bytes as listed in Table 8. The device must drive the nCS pin high after the last bit of the operation sequence is shifted in.

For read operations, the data read is shifted out on the DATA pin. You can drive the nCS pin high when any bit of the data is shifted out.

For write and erase operations, drive the nCS pin high at a byte boundary, that is in a multiple of eight clock pulses. Otherwise, the operation is rejected and not executed.

All attempts to access the memory contents while a write or erase cycle is in progress are rejected, and the write or erase cycle continues unaffected.

Addressing Mode

The 3-byte addressing mode is enabled by default. To access the EPCQ256 memory, you must use the 4-byte addressing mode. In 4-byte addressing mode, the address width is 32-bit address. To enable the 4-byte addressing mode, you must execute the 4BYTEADDREN operation. This addressing mode takes effect immediately after you execute the 4BYTEADDREN operation and remains active in the subsequent power-ups. To disable the 4-byte addressing mode, you must execute the 4BYTEADDREX operation.

☞ If you are using the Quartus® II software or the SRunner software to program the EPCQ256 device, you do not need to execute the 4BYTEADDREN operation. These software automatically enable the 4-byte addressing mode when programming the device.

Summary of Operation Codes

Table 8 lists the supported operations.

Table 8. Operation Codes for EPCQ Devices—Preliminary

Operation	Operation Code (1)	Address Bytes	Dummy Bytes	Data Bytes	DCLK f_{MAX} (MHz) (2)
Read status	b'0000 0101	0	0	1 to infinite (3)	100
Read bytes	b'0000 0011	3 or 4	0	1 to infinite (3)	50
Read device identification	b'1001 111x	0	0	1 to 20 (3)	100
Fast read	b'0000 1011	3 or 4	8 (5)	1 to infinite (3)	100
Extended dual input fast read	b'1011 1011	3 or 4	8 (5)	1 to infinite (3)	100
Extended quad input fast read	b'1110 1011	3 or 4	10 (5)	1 to infinite (3)	100
Write enable	b'0000 0110	0	0	0	100
Write disable	b'0000 0100	0	0	0	100
Write status	b'0000 0001	0	0	1	100
Write bytes	b'0000 0010	3 or 4	0	1 to 256 (4)	100
Extended dual input fast write bytes	b'1101 0010	3 or 4	0	1 to 256 (4)	100
Extended quad input fast write bytes	b'0001 0010	3 or 4	0	1 to 256 (4)	100
Erase bulk	b'1100 0111	0	0	0	100
Erase sector	b'1101 1000	3 or 4	0	0	100
4BYTEADDREN (6)	b'1011 0111	0	0	0	100
4BYTEASSREX (6)	b'1110 1001	0	0	0	100

Notes to Table 8:

(1) List MSB first and LSB last.

(2) Pending characterization data.

(3) The status register, data, or read device identification is read out at least once on the DATA1 pin and is continuously read out until the nCS pin is driven high.

(4) A write bytes operation requires at least one data byte on the DATA1 pin. If more than 256 bytes are sent to the device, only the last 256 bytes are written to the memory.

(5) You can configure the number of dummy bytes.

(6) This operation is applicable for EPCQ256 device only.

4BYTEADDREN AND 4BYTEADDREX Operations

To enable 4BYTEADDREN or 4BYTEADDREX operations, you can select the device by driving the `nCS` signal low, followed by shifting in the operation code through `DATA0`.

You must execute a write enable operation before you can enable the 4BYTEADDREN or 4BYTEADDREX operations. For more information, refer to "Write Enable Operation".

Figure 1 shows the timing diagram for the 4BYTEADDREN operation.

Figure 1. 4BYTEADDREN Timing Diagram

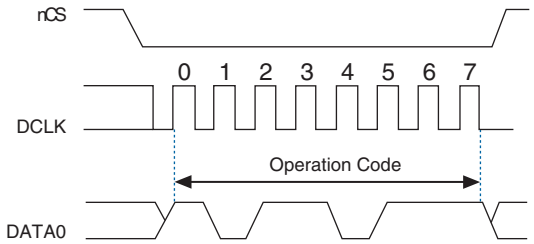
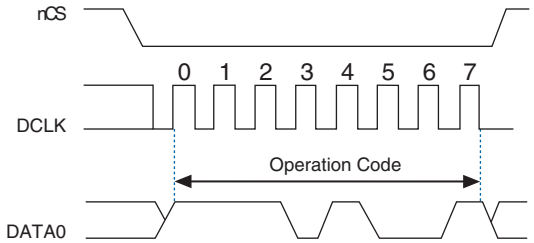


Figure 2 shows the timing diagram for the 4BYTEADDREX operation.

Figure 2. 4BYTEADDREX Timing Diagram

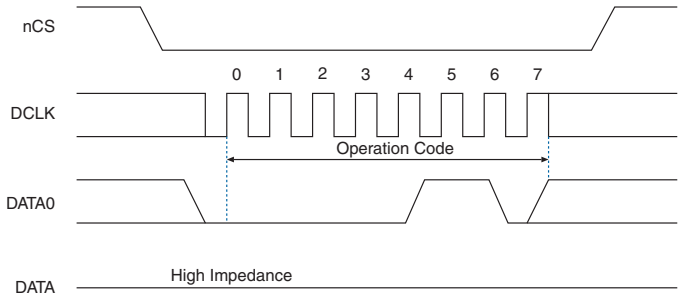


Write Enable Operation

When you enable the write enable operation, the write enable latch bit is set to `1` in the status register. You must execute this operation before the write bytes, write status, erase bulk, erase sector, extended dual input fast write bytes, extended quad input fast write bytes, 4BYTEADDREN, and 4BYTEADDREX operations.

Figure 3 shows the timing diagram for the write enable operation.

Figure 3. Write Enable Operation Timing Diagram



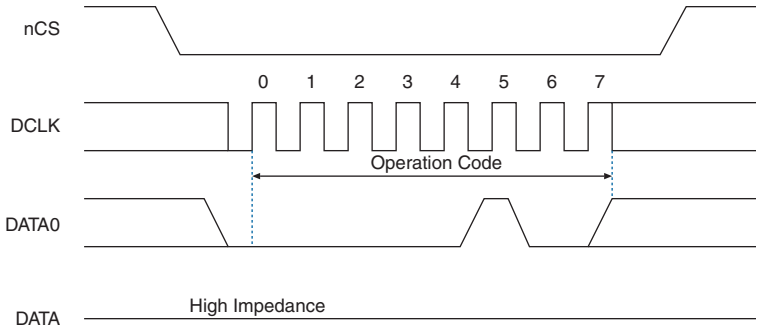
Write Disable Operation

The write disable operation resets the write enable latch bit in the status register. To prevent the memory from being written unintentionally, the write enable latch bit is automatically reset when implementing the write disable operation, and under the following conditions:

- Power up
- Write bytes operation completion
- Write status operation completion
- Erase bulk operation completion
- Erase sector operation completion
- Extended dual input fast write bytes operation completion
- Extended quad input fast write bytes operation completion

Figure 4 shows the timing diagram for the write disable operation.

Figure 4. Write Disable Operation Timing Diagram



Read Status Operation

You can use the read status operation to read the status register. Figure 5 and Figure 6 show the status bits in the status register of the EPCQ devices.

Figure 5. EPCQ16 and EPCQ32 Status Register Status Bits

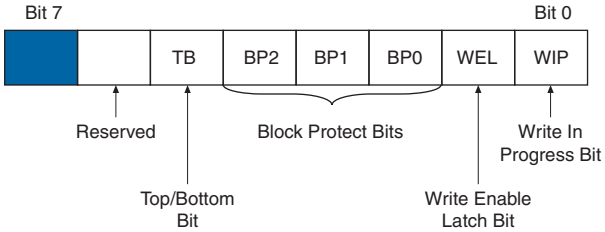
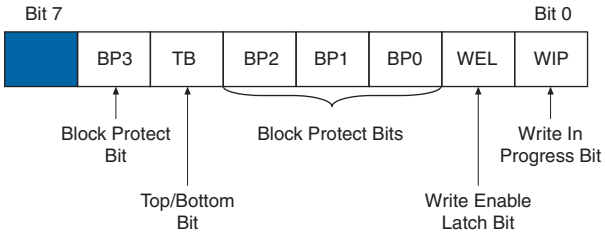


Figure 6. EPCQ64, EPCQ128, and EPCQ256 Status Register Status Bits



If you set the write in progress bit to `1`, the EPCQ device executes a write or erase cycle and `0` indicates that no write or erase cycle is in progress.

If you set the write enable latch bit to `0`, the EPCQ device rejects a write or erase cycle. You must set the write enable latch bit to `1` before every write bytes, write status, erase bulk, and erase sector operations.

Use the top or bottom bit (TB bit) with the block protect bits to determine that the protected area starts from the top or bottom of the memory array. When the top or bottom bit is set to `0`, the protected area starts from the top of the memory array. When the top or bottom bit is set to `1`, the protected area starts from the bottom of the memory array.

The non-volatile block protect bits determine the area of the memory protected from being written or erased unintentionally. Table 9 through Table 18 on page 20 list the protected area in EPCQ16, EPCQ32, EPCQ64, EPCQ128, and EPCQ256 devices with reference to the block protect bits. The erase bulk operation is only available when all the block protect bits are set to `0`. When any of the block protect bits are set to `1`, the relevant area is protected from being written by a write bytes operation or erased by an erase sector operation.

Table 9. Block Protection Bits in EPCQ16 when TB Bit is Set to 0

Status Register Content				Memory Content	
TB Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	0	0	None	All sectors
0	0	0	1	Upper 32	Sectors 0 to 30
0	0	1	0	Upper 16	Sectors 0 to 29
0	0	1	1	Upper 8	Sectors 0 to 27
0	1	0	0	Upper 4	Sectors 0 to 23
0	1	0	1	Upper half	Sectors 0 to 15
0	1	1	0	All sectors	None
0	1	1	1	All sectors	None

Table 10. Block Protection Bits in EPCQ16 when TB Bit is Set to 1

Status Register Content				Memory Content	
TB Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
1	0	0	0	None	All sectors
1	0	0	1	Lower 32	Sectors 1 to 31
1	0	1	0	Lower 16	Sectors 2 to 31
1	0	1	1	Lower 8	Sectors 4 to 31
1	1	0	0	Lower 4	Sectors 8 to 31
1	1	0	1	Lower half	Sectors 16 to 31
1	1	1	0	All sectors	None
1	1	1	1	All sectors	None

Table 11. Block Protection Bits in EPCQ32 when TB Bit is Set to 0

Status Register Content				Memory Content	
TB Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	0	0	None	All sectors
0	0	0	1	Upper 64	Sectors 0 to 62
0	0	1	0	Upper 32	Sectors 0 to 61
0	0	1	1	Upper 16	Sectors 0 to 59
0	1	0	0	Upper 8	Sectors 0 to 55
0	1	0	1	Upper 4	Sectors 0 to 47
0	1	1	0	Upper half	Sectors 0 to 31
0	1	1	1	All sectors	None

Table 12. Block Protection Bits in EPCQ32 when TB Bit is Set to 1

Status Register Content				Memory Content	
TB Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
1	0	0	0	None	All sectors
1	0	0	1	Lower 64	Sectors 1 to 63
1	0	1	0	Lower 32	Sectors 2 to 63
1	0	1	1	Lower 16	Sectors 4 to 63
1	1	0	0	Lower 8	Sectors 8 to 63
1	1	0	1	Lower 4	Sectors 16 to 63
1	1	1	0	Lower half	Sectors 32 to 63
1	1	1	1	All sectors	None

Table 13. Block Protection Bits in EPCQ64 when TB Bit is Set to 0

Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	0	0	0	None	All sectors
0	0	0	0	1	Upper 128	Sectors 0 to 126
0	0	0	1	0	Upper 64	Sectors 0 to 125
0	0	0	1	1	Upper 32	Sectors 0 to 123
0	0	1	0	0	Upper 16	Sectors 0 to 119
0	0	1	0	1	Upper 8	Sectors 0 to 111
0	0	1	1	0	Upper quarter	Sectors 0 to 95
0	0	1	1	1	Upper half	Sectors 0 to 63
0	1	0	0	0	All sectors	None
0	1	0	0	1	All sectors	None
0	1	0	1	0	All sectors	None
0	1	0	1	1	All sectors	None
0	1	1	0	0	All sectors	None
0	1	1	0	1	All sectors	None
0	1	1	1	0	All sectors	None
0	1	1	1	1	All sectors	None

Table 14. Block Protection Bits in EPCQ64 when TB Bit is Set to 1

Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
1	0	0	0	0	None	All sectors
1	0	0	0	1	Lower 128	Sectors 1 to 127
1	0	0	1	0	Lower 64	Sectors 2 to 127
1	0	0	1	1	Lower 32	Sectors 4 to 127
1	0	1	0	0	Lower 16	Sectors 8 to 127
1	0	1	0	1	Lower 8	Sectors 16 to 127
1	0	1	1	0	Lower quarter	Sectors 32 to 127
1	0	1	1	1	Lower half	Sectors 64 to 127
1	1	0	0	0	All sectors	None
1	1	0	0	1	All sectors	None
1	1	0	1	0	All sectors	None
1	1	0	1	1	All sectors	None
1	1	1	0	0	All sectors	None
1	1	1	0	1	All sectors	None
1	1	1	1	0	All sectors	None
1	1	1	1	1	All sectors	None

Table 15. Block Protection Bits in EPCQ128 when TB Bit is Set to 0

Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	0	0	0	None	All sectors (sectors 0 to 255)
0	0	0	0	1	Upper 256	Sectors 0 to 254
0	0	0	1	0	Upper 128	Sectors 0 to 253
0	0	0	1	1	Upper 64	Sectors 0 to 251
0	0	1	0	0	Upper 32	Sectors 0 to 247
0	0	1	0	1	Upper 16	Sectors 0 to 239
0	0	1	1	0	Upper 8	Sectors 0 to 223
0	0	1	1	1	Upper quarter	Sectors 0 to 191
0	1	0	0	0	Upper half (sector 255)	Lower half (sectors 0 to 127)
0	1	0	0	1	All sectors	None
0	1	0	1	0	All sectors	None
0	1	0	1	1	All sectors	None
0	1	1	0	0	All sectors	None
0	1	1	0	1	All sectors	None
0	1	1	1	0	All sectors	None
0	1	1	1	1	All sectors (128 Mb, 256 sectors)	None

Table 16. Block Protection Bits in EPCQ128 when TB Bit is Set to 1

Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
1	0	0	0	0	None	All sectors (sectors 0 to 255)
1	0	0	0	1	Lower 256	Sectors 1 to 255
1	0	0	1	0	Lower 128	Sectors 2 to 255
1	0	0	1	1	Lower 64	Sectors 4 to 255
1	0	1	0	0	Lower 32	Sectors 8 to 255
1	0	1	0	1	Lower 16	Sectors 16 to 255
1	0	1	1	0	Lower 8	Sectors 32 to 255
1	0	1	1	1	Lower quarter	Sectors 64 to 255
1	1	0	0	0	Lower half	Sectors 128 to 255
1	1	0	0	1	All sectors	None
1	1	0	1	0	All sectors	None
1	1	0	1	1	All sectors	None
1	1	1	0	0	All sectors	None
1	1	1	0	1	All sectors	None
1	1	1	1	0	All sectors	None
1	1	1	1	1	All sectors	None

Table 17. Block Protection Bits in EPCQ256 when TB Bit is Set to 0

Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
0	0	0	0	0	None	All sectors
0	0	0	0	1	Upper 512	Sectors 0 to 511
0	0	0	1	0	Upper 256	Sectors 0 to 510
0	0	0	1	1	Upper 128	Sectors 0 to 508
0	0	1	0	0	Upper 64	Sectors 0 to 504
0	0	1	0	1	Upper 32	Sectors 0 to 496
0	0	1	1	0	Upper 16	Sectors 0 to 480
0	0	1	1	1	Upper 8	Sectors 0 to 448
0	1	0	0	0	Upper quarter	Sectors 0 to 384
0	1	0	0	1	Upper half	Sectors 0 to 256
0	1	0	1	0	All sectors	None
0	1	0	1	1	All sectors	None
0	1	1	0	0	All sectors	None
0	1	1	0	1	All sectors	None
0	1	1	1	0	All sectors	None
0	1	1	1	1	All sectors	None

Table 18. Block Protection Bits in EPCQ256 when TB Bit is Set to 1

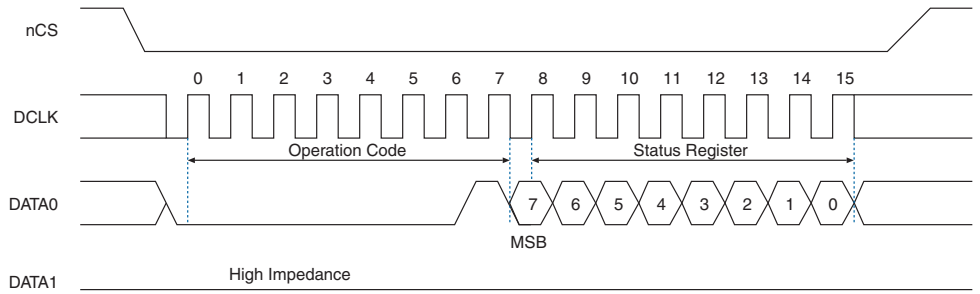
Status Register Content					Memory Content	
TB Bit	BP3 Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
1	0	0	0	0	None	All sectors
1	0	0	0	1	Lower 512	Sectors 1 to 511
1	0	0	1	0	Lower 256	Sectors 2 to 511
1	0	0	1	1	Lower 128	Sectors 4 to 511
1	0	1	0	0	Lower 64	Sectors 8 to 511
1	0	1	0	1	Lower 32	Sectors 16 to 511
1	0	1	1	0	Lower 16	Sectors 32 to 511
1	0	1	1	1	Lower 8	Sectors 64 to 511
1	1	0	0	0	Lower quarter	Sectors 128 to 511
1	1	0	0	1	Lower half	Sectors 256 to 511
1	1	0	1	0	All sectors	None
1	1	0	1	1	All sectors	None
1	1	1	0	0	All sectors	None
1	1	1	0	1	All sectors	None
1	1	1	1	0	All sectors	None
1	1	1	1	1	All sectors	None

Write Status Operation

The write status operation does not affect the write enable latch and write in progress bits. You can use the write status operation to set the status register block protection and top or bottom bits. Therefore, you can implement this operation to protect certain memory sectors. For more information, refer to Table 15 on page 19 through Table 18. After setting the block protect bits, the protected memory sectors are treated as read-only memory. You must execute the write enable operation before the write status operation.

Figure 7 shows the timing diagram for the write status operation.

Figure 7. Write Status Operation Timing Diagram



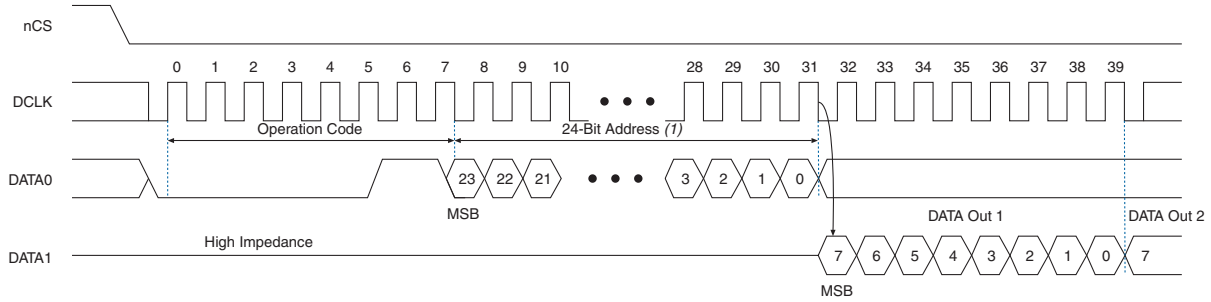
Immediately after the nCS signal drives high, the device initiates the self-timed write status cycle. The self-timed write status cycle usually takes 5 ms for all EPCQ devices and is guaranteed to be less than 8 ms. For more information, refer to t_{WS} in Table 20 on page 32. You must account for this delay to ensure that the status register is written with the desired block protect bits. Alternatively, you can check the write in progress bit in the status register by executing the read status operation while the self-timed write status cycle is in progress. Set the write in progress bit to 1 during the self-timed write status cycle and 0 when it is complete.

Read Bytes Operation

When you execute the read bytes operation, you first shift in the read bytes operation code, followed by a 3-byte addressing mode (`A[23..0]`) or a 4-byte addressing mode (`A[31..0]`). Each address bit must be latched in on the rising edge of the `DCLK` signal. After the address is latched in, the memory contents of the specified address are shifted out serially on the `DATA1` pin, beginning with the MSB. For reading Raw Programming Data File (**.rpd**), the content is shifted out serially beginning with the LSB. Each data bit is shifted out on the falling edge of the `DCLK` signal. The maximum `DCLK` frequency during the read bytes operation is 50 MHz.

Figure 8 shows the timing diagram for the read bytes operation.

Figure 8. Read Bytes Operation Timing Diagram



Note to Figure 8:

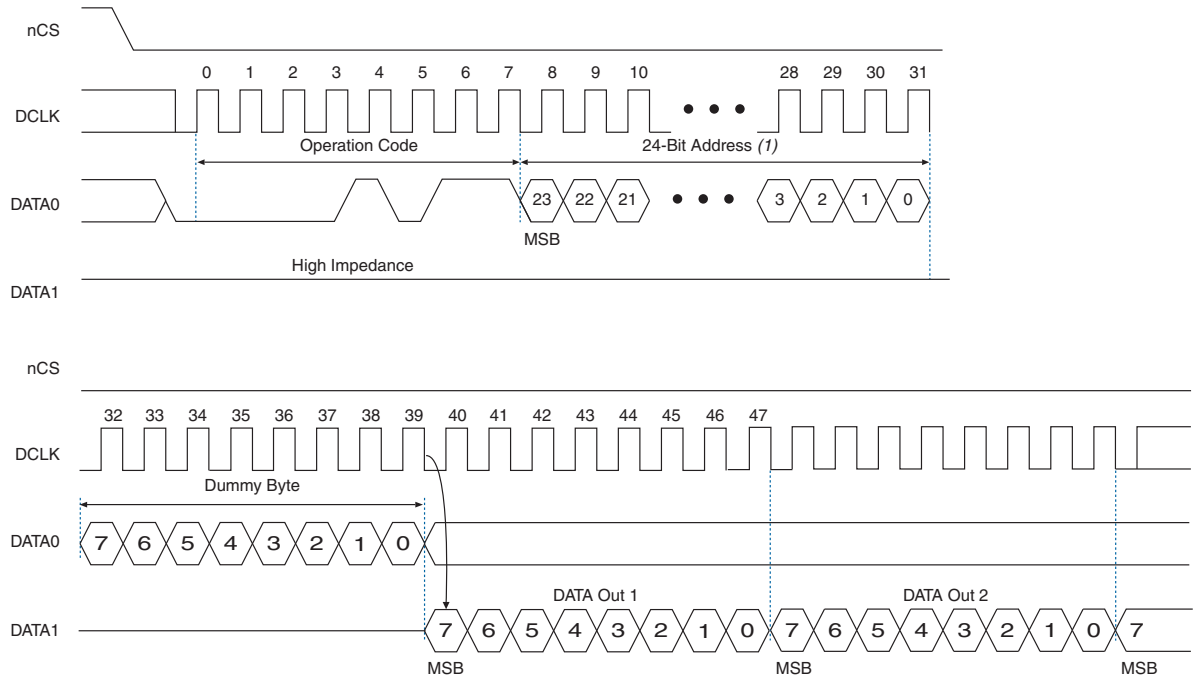
(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

The first byte address can be at any location. The device automatically increases the address to the next higher address after shifting out each byte of data. Therefore, the device can read the whole memory with a single read bytes operation. When the device reaches the highest address, the address counter restarts at `0x000000`, allowing the memory contents to be read out indefinitely until the read bytes operation is terminated by driving the `nCS` signal high. If the read bytes operation is shifted in while a write or erase cycle is in progress, the operation is not executed and does not affect the write or erase cycle in progress.

Fast Read Operation

When you execute the fast read operation, you first shift in the fast read operation code, followed by a 3-byte addressing mode (`A[23..0]`) or a 4-byte addressing mode (`A[31..0]`), and a dummy byte with each bit being latched-in during the rising edge of the `DCLK` signal. Then, the memory contents at that address is shifted out on `DATA1` with each bit being shifted out at a maximum frequency of 100 MHz during the falling edge of the `DCLK` signal. Figure 9 shows the operation sequence of the fast read operation.

Figure 9. Fast Read Operation Timing Diagram



Note to Figure 9:

(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

The first byte address can be at any location. The device automatically increases the address to the next higher address after shifting out each byte of data. Therefore, the device can read the whole memory with a single fast read operation. When the device reaches the highest address, the address counter restarts at `0x000000`, allowing the read sequence to continue indefinitely.

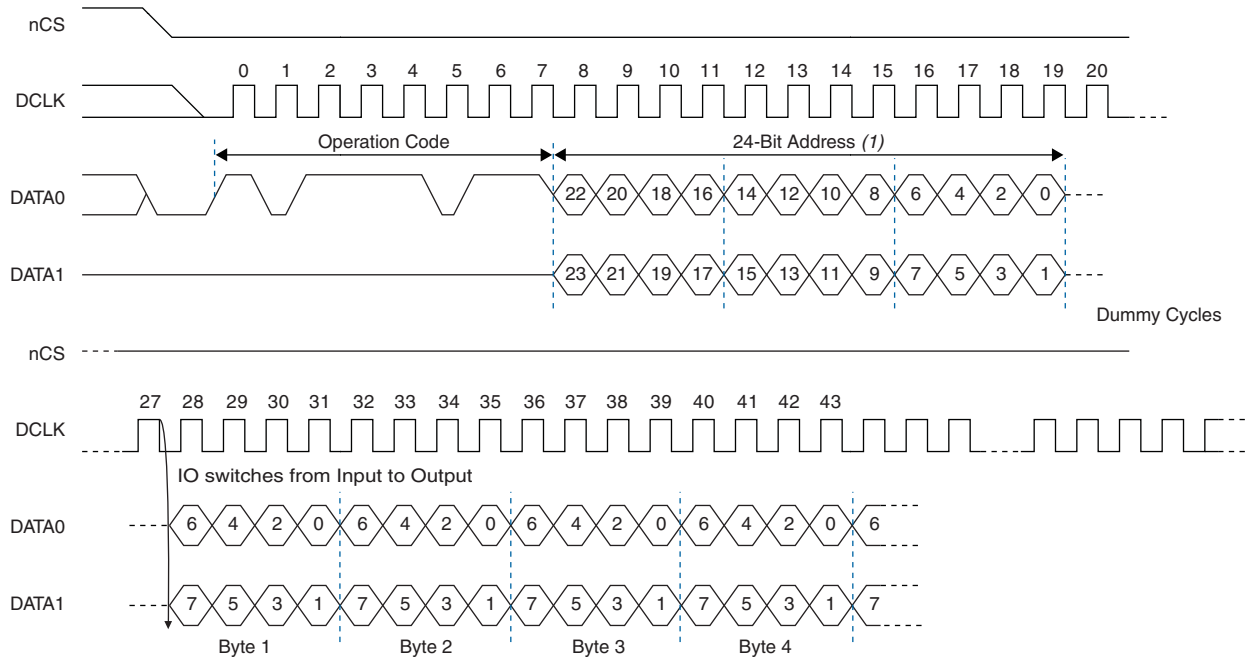
You can terminate the fast read operation by driving the `nCS` signal high at any time during data output. If the fast read operation is shifted in while an erase, program, or write cycle is in progress, the operation is not executed and does not affect the erase, program, or write cycle in progress.

Extended Dual Input Fast Read Operation

This operation is similar to the fast read operation except that the data and addresses are shifted in and out on the DATA0 and DATA1 pins.

Figure 10 shows the operation sequence of the extended dual input fast read operation.

Figure 10. Extended Dual Input Fast Read Operation Timing Diagram



Note to Figure 10:

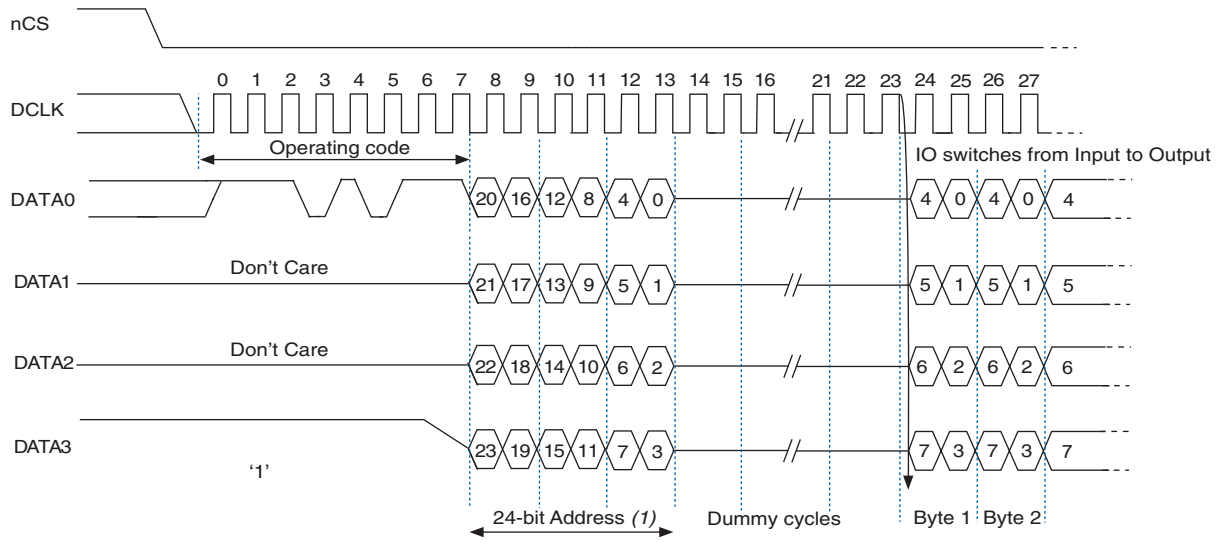
(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

Extended Quad Input Fast Read Operation

This operation is similar to the extended dual input fast read operation except that the data and addresses are shifted in and out on the DATA0, DATA1, DATA2, and DATA3 pins.

Figure 11 shows the operation sequence of the extended quad input fast read operation.

Figure 11. Extended Quad Input Fast Read Operation Timing Diagram



Note to Figure 11:

(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

Read Device Identification Operation

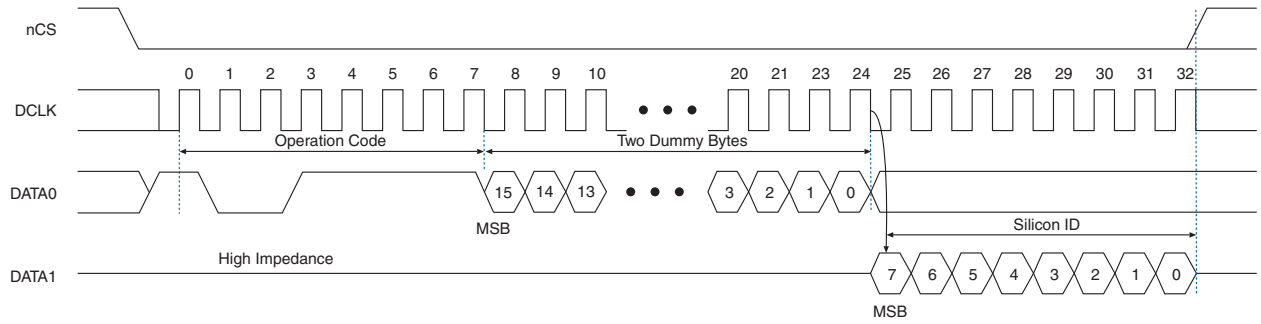
This operation reads the 8-bit device identification of the EPCQ device from the DATA1 output pin. If this operation is shifted in while an erase or write cycle is in progress, the operation is not executed and does not affect the erase or write cycle in progress. Table 19 lists the EPCQ device identifications.

Table 19. EPCQ Device Identification

EPCQ Device	Silicon ID (Binary Value)
EPCQ16	b'0001 0101
EPCQ32	b'0001 0110
EPCQ64	b'0001 0111
EPCQ128	b'0001 1000
EPCQ256	b'0001 1001

The 8-bit device identification of the EPCQ device is shifted out on the DATA1 pin on the falling edge of the DCLK signal. Figure 12 shows the operation sequence of the read device identification operation.

Figure 12. Read Device Identification Operation Timing Diagram



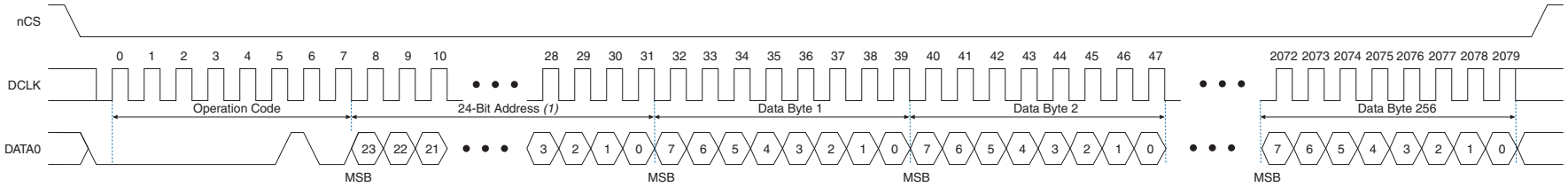
Write Bytes Operation

This operation allows bytes to be written to the memory. You must execute the write enable operation before the write bytes operation. After the write bytes operation is completed, the write enable latch bit in the status register is set to `0`.

When you execute the write bytes operation, you shift in the write bytes operation code, followed by a 3-byte addressing mode (`A[23..0]`) or a 4-byte addressing mode (`A[31..0]`), and at least one data byte on the `DATA0` pin. If the eight LSBs (`A[7..0]`) are not all `0`, all sent data that goes beyond the end of the current page is not written into the next page. Instead, this data is written at the start address of the same page. You must ensure the `nCS` signal is set low during the entire write bytes operation.

Figure 13 shows the operation sequence of the write bytes operation.

Figure 13. Write Bytes Operation Timing Diagram



Note to Figure 13:

(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

If more than 256 data bytes are shifted into the EPCQ device with a write bytes operation, the previously latched data is discarded and the last 256 bytes are written to the page. However, if less than 256 data bytes are shifted into the EPCQ device, they are guaranteed to be written at the specified addresses and the other bytes of the same page are not affected.

The device initiates a self-timed write cycle immediately after the `nCS` signal is driven high. For more information about the self-timed write cycle time, refer to t_{WB} in Table 20 on page 32. You must account for this amount of delay before another page of memory is written. Alternatively, you can check the write in progress bit in the status register by executing the read status operation while the self-timed write cycle is in progress. The write in progress bit is set to `1` during the self-timed write cycle and `0` when it is complete.

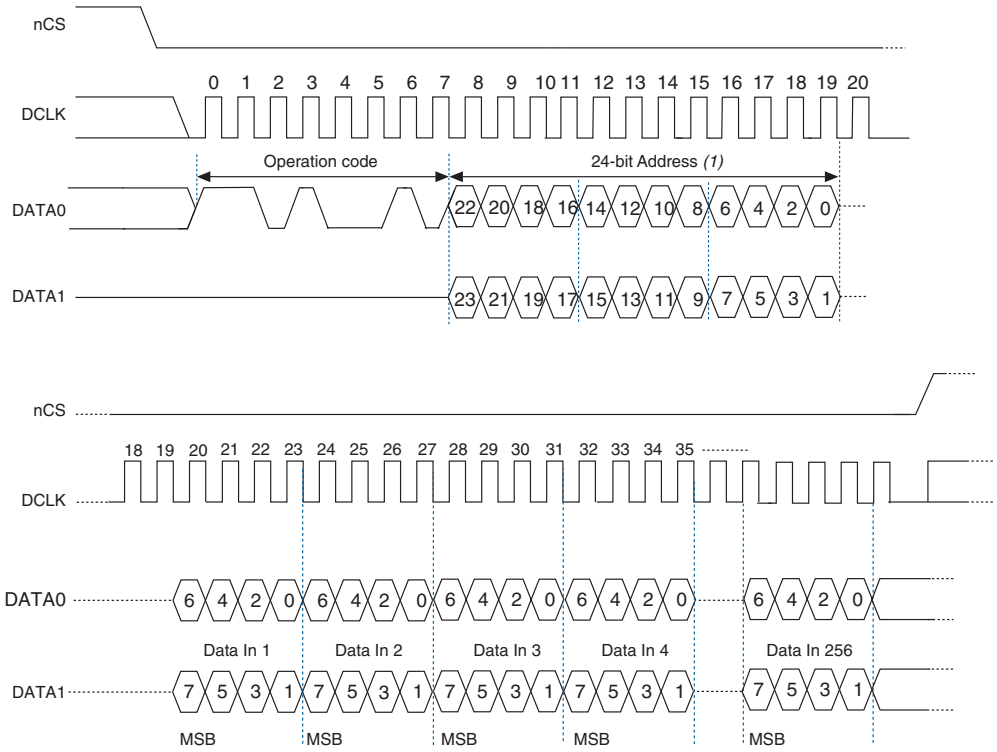
You must erase all the memory bytes of EPCQ devices before you implement the write bytes operation. You can erase all the memory bytes by executing the erase sector operation in a sector or the erase bulk operation throughout the entire memory.

Extended Dual Input Fast Write Bytes Operation

This operation is similar to the write bytes operation except that the data and addresses are shifted in on the DATA0 and DATA1 pins.

Figure 14 shows the operation sequence of the extended dual input fast write bytes operation.

Figure 14. Extended Dual Input Fast Write Bytes Timing Diagram



Note to Figure 14:

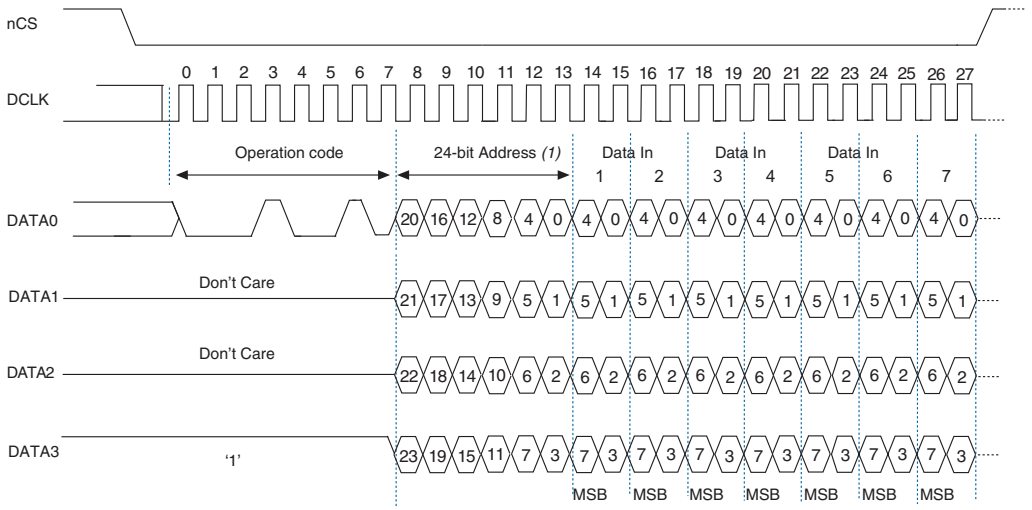
(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

Extended Quad Input Fast Write Bytes Operation

This operation is similar to the extended dual input fast write bytes operation except that the data and addresses are shifted in on the DATA0, DATA1, DATA2, and DATA3 pins.

Figure 15 shows the operation sequence of the extended quad input fast write bytes operation.

Figure 15. Extended Quad Input Fast Write Bytes Operation Timing Diagram



Note to Figure 15:

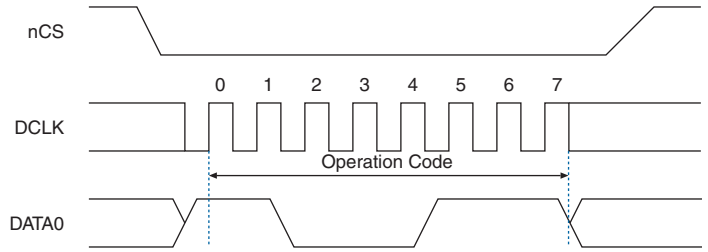
(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

Erase Bulk Operation

This operation sets all the memory bits to `1` or `0xFF`. Similar to the write bytes operation, you must execute the write enable operation before the erase bulk operation.

You can implement the erase bulk operation by driving the `nCS` signal low and then shifting in the erase bulk operation code on the `DATA0` pin. The `nCS` signal must be driven high after the eighth bit of the erase bulk operation code has been latched in. Figure 16 shows the erase bulk operation.

Figure 16. Erase Bulk Operation Timing Diagram



The device initiates a self-timed erase bulk cycle immediately after the `nCS` signal is driven high. For more information about the self-timed erase bulk cycle time, refer to t_{WB} in Table 20 on page 32.

You must account for this delay before accessing the memory contents. Alternatively, you can check the write in progress bit in the status register by executing the read status operation while the self-timed erase cycle is in progress. The write in progress bit is set to `1` during the self-timed erase cycle and `0` when it is complete. The write enable latch bit in the status register is reset to `0` before the erase cycle is complete.

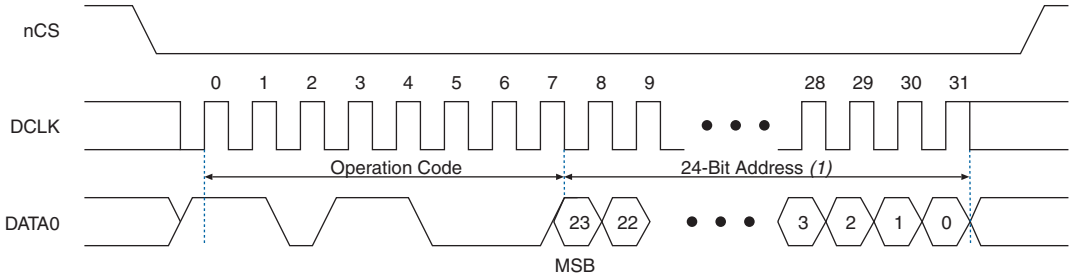
Erase Sector Operation

The erase sector operation allows you to erase a certain sector in the EPCQ device by setting all the bits inside the sector to `1` or `0xFF`. This operation is useful if you want to access the unused sectors as a general purpose memory in your applications. You must execute the write enable operation before the erase sector operation.

When you execute the erase sector operation, you must first shift in the erase sector operation code, followed by the 3-byte addressing mode (`A[23..0]`) or the 4-byte addressing mode (`A[31..0]`) of the chosen sector on the `DATA0` pin. The 3-byte addressing mode or the 4-byte addressing mode for the erase sector operation can be any address inside the specified sector. For more information about the sector address range, refer to Table 6 on page 6 and Table 7 on page 8. Drive the `nCS` signal high after the eighth bit of the erase sector operation code has been latched in.

Figure 17 shows the erase sector operation.

Figure 17. Erase Sector Operation Timing Diagram



Note to Figure 17:

(1) To access the entire EPCQ256 memory, use 4-byte addressing mode. In the 4-byte addressing mode, the address width is 32-bit address.

The device initiates a self-timed erase sector cycle immediately after the `nCS` signal is driven high. For more information about the self-timed erase sector cycle time, refer to t_{ES} in Table 20 on page 32. You must account for this amount of delay before another page of memory is written. Alternatively, you can check the write in progress bit in the status register by executing the read status operation while the self-timed erase cycle is in progress. The write in progress bit is set to `1` during the self-timed erase cycle and `0` when it is complete. The write enable latch bit in the status register is set to `0` before the self-timed erase cycle is complete.

Power Mode

EPCQ devices support active and standby power modes. When the `nCS` signal is low, the device is enabled and is in active power mode. The FPGA is configured while the EPCQ device is in active power mode. When the `nCS` signal is high, the device is disabled but remains in active power mode until all internal cycles are completed, such as write or erase operations. The EPCQ device then goes into standby power mode. The I_{CC1} and I_{CC0} parameters list the V_{CC} supply current when the device is in active and standby power modes. For more information, refer to Table 25 on page 36.

Timing Information

Figure 18 shows the timing waveform for the write operation.

Figure 18. Write Operation Timing Diagram

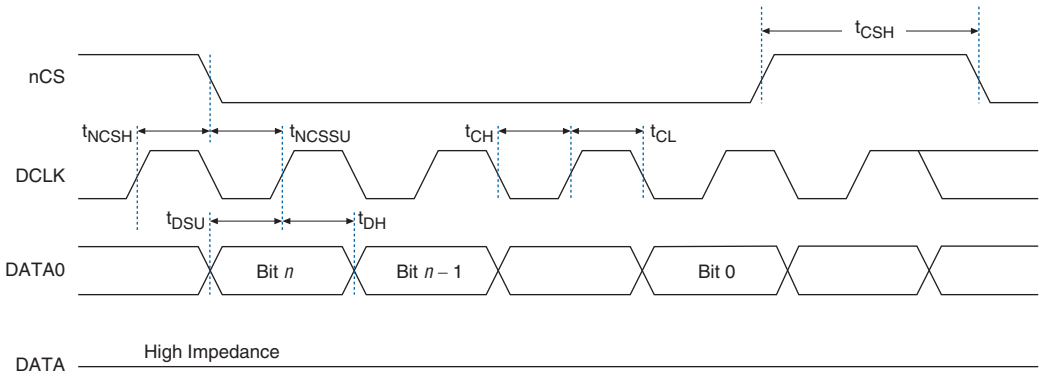


Table 20 lists the EPCQ device timing parameters for the write operation.

Table 20. Write Operation Parameters—Preliminary (Part 1 of 2)

Symbol	Parameter	Min	Typical	Max	Unit
f_{WCLK}	Write clock frequency (from the FPGA, download cable, or embedded processor) for write enable, write disable, read status, read device identification, write bytes, erase bulk, and erase sector operations	—	—	100	MHz
t_{CH} (1)	DCLK high time	4	—	—	ns
t_{CL} (1)	DCLK low time	4	—	—	ns
t_{NCSSU}	Chip select (nCS) setup time	4	—	—	ns
t_{NCSH}	Chip select (nCS) hold time	4	—	—	ns
t_{DSU}	DATA[] in setup time before the rising edge on DCLK	2	—	—	ns
t_{DH}	DATA[] hold time after the rising edge on DCLK	3	—	—	ns
t_{CSH}	Chip select (nCS) high time	50	—	—	ns
t_{WB} (2)	Write bytes cycle time	—	0.6	5	ms
t_{WS} (2)	Write status cycle time	—	1.3	8	ms
t_{EB} (2)	Erase bulk cycle time for EPCQ16	—	170	250	s
	Erase bulk cycle time for EPCQ32	—	170	250	s
	Erase bulk cycle time for EPCQ64	—	60	250	s
	Erase bulk cycle time for EPCQ128	—	170	250	s
	Erase bulk cycle time for EPCQ256	—	240	480	s
t_{ES} (2)	Erase sector cycle time for EPCQ16	—	0.7	3	s
	Erase sector cycle time for EPCQ32				
	Erase sector cycle time for EPCQ64				
	Erase sector cycle time for EPCQ128				
	Erase sector cycle time for EPCQ256				

Table 20. Write Operation Parameters—Preliminary (Part 2 of 2)

Symbol	Parameter	Min	Typical	Max	Unit
t_{ESS} (2)	Erase subsector cycle time for EPCQ16	—	0.3	1.5	s
	Erase subsector cycle time for EPCQ32				
	Erase subsector cycle time for EPCQ64				
	Erase subsector cycle time for EPCQ128				
	Erase subsector cycle time for EPCQ256				

Notes to Table 20:

(1) The value must be larger than or equal to $1/f_{WCLK}$.

(2) Figure 18 does not show these parameters.

Figure 19 shows the timing waveform for the read operation.

Figure 19. Read Operation Timing Diagram

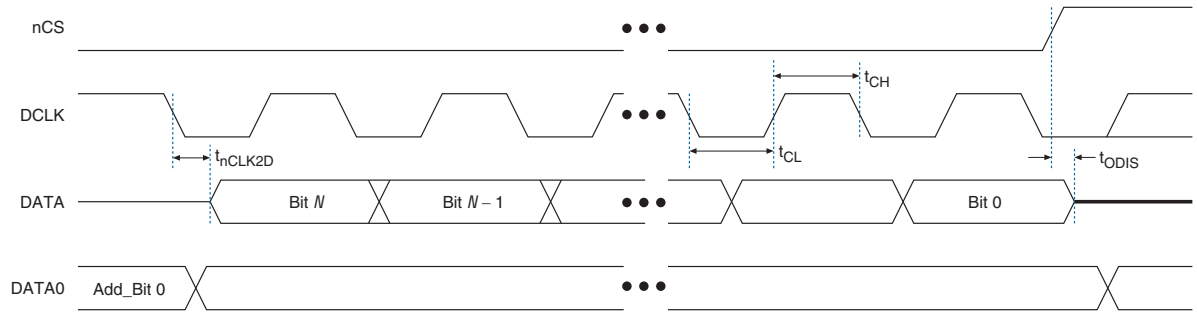


Table 21 lists the EPCQ device timing parameters for the read operation.

Table 21. Read Operation Parameters—Preliminary

Symbol	Parameter	Min	Max	Unit
f_{RCLK}	Read clock frequency (from the FPGA or embedded processor) for read bytes operations	—	50	MHz
	Fast read clock frequency (from the FPGA or embedded processor) for fast read bytes operation	—	100	MHz
t_{CH}	DCLK high time	4	—	ns
t_{CL}	DCLK low time	4	—	ns
t_{ODIS}	Output disable time after read	—	8	ns
t_{nCLK2D}	Clock falling edge to DATA	—	7	ns

Programming and Configuration File Support

The Quartus II software provides programming support for EPCQ devices. When you select an EPCQ device, the Quartus II software automatically generates the Programmer Object File (**.pof**) to program the device. The software allows you to select the appropriate EPCQ device density that most efficiently stores the configuration data for the selected FPGA.

You can program the EPCQ device in-system by an external microprocessor using the SRunner software driver. The SRunner software driver is developed for embedded EPCQ device programming that you can customize to fit in different embedded systems. The SRunner software driver reads **.rpd** files and writes to the EPCQ devices. The programming time is comparable to the Quartus II software programming time. Because the FPGA reads the LSB of the **.rpd** data first during the configuration process, the LSB of **.rpd** bytes must be shifted out first during the read bytes operation and shifted in first during the write bytes operation.

Writing and reading the **.rpd** file to and from the EPCQ device is different from the other data and address bytes.

During the ISP of an EPCQ device using the USB-Blaster, EthernetBlaster II, or EthernetBlaster download cable, the cable pulls the `nCONFIG` signal low to reset the FPGA and overrides the 10-kΩ pull-down resistor on the `nCE` pin of the FPGA. The download cable then uses the interface pins depending on the selected AS mode to program the EPCQ device. When programming is complete, the download cable releases the interface pins of the EPCQ device and the `nCE` pin of the FPGA and pulses the `nCONFIG` signal to start the configuration process.

The FPGA can program the EPCQ device in-system using the JTAG interface with the serial flash loader (SFL). This solution allows you to indirectly program the EPCQ device using the same JTAG interface that is used to configure the FPGA.

For more information about configuration, refer to the configuration chapter in the appropriate device handbook.

For more information about SFL, refer to *AN 370: Using the Serial FlashLoader with the Quartus II software*.

For more information about programming and configuration support, refer to the following documents:

- *USB-Blaster Download Cable User Guide*
- *EthernetBlaster II Communications Cable User Guide*
- *EthernetBlaster Communications Cable User Guide*

Operating Conditions

Table 22 through Table 26 list information about the absolute maximum ratings, recommended operating conditions, DC operating conditions, I_{CC} supply current, and capacitance for EPCQ devices.

Table 22. Absolute Maximum Ratings—Preliminary

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supply voltage	With respect to GND	–0.6	4.0	V
V_I	DC input voltage	With respect to GND	–0.6	4.0	V
I_{MAX}	DC V_{CC} or GND current	—	—	20	mA
I_{OUT}	DC output current per pin	—	–25	25	mA
P_D	Power dissipation	—	—	54	mW
T_{STG}	Storage temperature	No bias	–65	150	°C
T_{AMB}	Ambient temperature	Under bias	–65	135	°C
T_J	Junction temperature	Under bias	—	135	°C

Table 23. Recommended Operating Conditions—Preliminary

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supply voltage	(1)	2.7	3.6	V
V_I	Input voltage	With respect to GND	–0.5	0.4 + V_{CC}	V
V_O	Output voltage	—	0	V_{CC}	V
T_A	Operating temperature	For commercial use	0	70	°C
		For industrial use	–40	85	°C
t_R	Input rise time	—	—	5	ns
t_F	Input fall time	—	—	5	ns

Note to Table 23:

(1) The maximum V_{CC} rise time is 100 ms.

Table 24. DC Operating Conditions—Preliminary

Symbol	Parameter	Condition	Min	Max	Unit
V_{IH}	High-level input voltage	—	0.7 x V_{CC}	V_{CC} + 0.4	V
V_{IL}	Low-level input voltage	—	–0.5	0.3 x V_{CC}	V
V_{OH}	High-level output voltage	I_{OH} = –100 µA (1)	V_{CC} - 0.2	—	V
V_{OL}	Low-level output voltage	I_{OL} = 1.6 mA (2)	—	0.4	V
I_I	Input leakage current	V_I =V_{CC} or GND	–10	10	µA
I_{OZ}	Tri-state output off-state current	V_O = V_{CC} or GND	–10	10	µA

Notes to Table 24:

(1) The I_{OH} parameter refers to the high-level TTL or CMOS output current.

(2) The I_{OL} parameter refers to the low-level TTL or CMOS output current.

Table 25. I_{CC} Supply Current—Preliminary

Symbol	Parameter	Condition	Min	Max	Unit
I_{CC0}	V_{CC} supply current (standby)	—	—	100	µA
I_{CC1}	V_{CC} supply current (during active power mode)	—	5	20	mA

Table 26. Capacitance (1)—Preliminary

Symbol	Parameter	Condition	Min	Max	Unit
C_{IN}	Input pin capacitance	V_{IN} = 0 V	—	6	pF
C_{OUT}	Output pin capacitance	V_{OUT} = 0 V	—	8	pF

Note to Table 26:

(1) Capacitance is sample-tested only at T_A = 25 x C and at a 20-MHz frequency.

Pin Information

Figure 20 and Figure 21 show the EPCQ device in an 8-pin device. Figure 22 and Figure 23 show the EPCQ device in a 16-pin device. The following lists the control pins on the EPCQ device:

- Serial data 3 (`DATA3`)
- Serial data 2 (`DATA2`)
- Serial data 1 (`DATA1`)
- Serial data 0 (`DATA0`)
- Serial clock (`DCLK`)
- Chip select (`nCS`)

For more information about configuration, refer to the configuration chapter in the appropriate device handbook.

Figure 20. AS x1 Pin-Out Diagram for EPCQ16 and EPCQ32 Devices

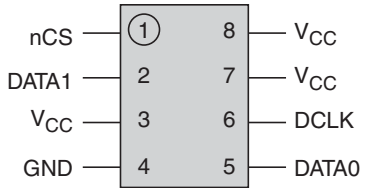


Figure 21. AS x4 Pin-Out Diagram for EPCQ16 and EPCQ32 Devices

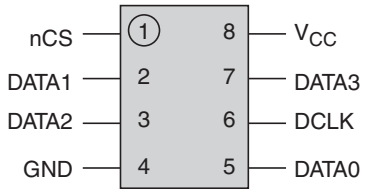
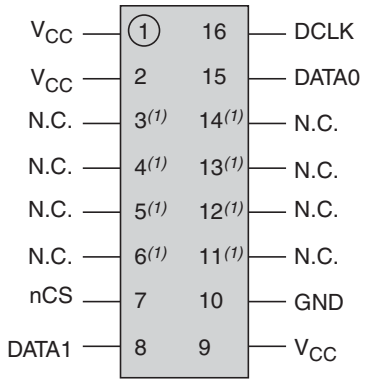


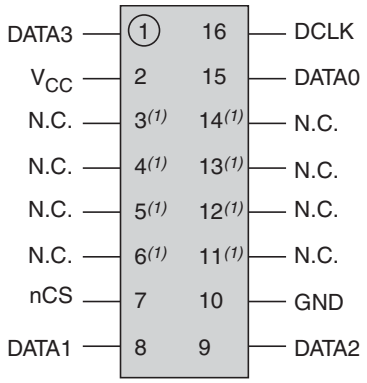
Figure 22. AS x1 Pin-Out Diagram for EPCQ64, EPCQ128, and EPCQ256 Devices



Note to Figure 22:

(1) You can leave these pins floating or you can connect them to V_{CC} or GND.

Figure 23. AS x4 Pin-Out Diagram for EPCQ64, EPCQ128, and EPCQ256 Devices



Note to Figure 23:

(1) You can leave these pins floating or you can connect them to V_{CC} or GND.

Table 27 lists the pin description of the EPCQ device.

Table 27. EPCQ Device Pin Description—Preliminary (Part 1 of 3)

Pin Name	AS x1 Pin-Out Diagram		AS x4 Pin-Out Diagram		Pin Type	Description
	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package		
DATA0	5	15	5	15	I/O	For AS x1 mode, use this pin as an input signal pin to write or program the EPCQ device. During write or program operations, the data is latched on the rising edge of the DCLK signal. For AS x4 mode, use this pin as an I/O signal pin. During write or program operations, this pin acts as an input pin that serially transfers data into the EPCQ device. The data is latched on the rising edge of the DCLK signal. During read or configuration operations, this pin acts as an output signal pin that serially transfers data out of the EPCQ device to the FPGA. The data is shifted out on the falling edge of the DCLK signal. During the extended quad input fast write bytes or extended dual input fast write bytes operations, this pin acts as an input pin that serially transfers data into the EPCQ device. The data is latched on the rising edge of the DCLK signal. During extended dual input fast read or extended quad input fast read operations, this pin acts as an output signal pin that serially transfers data out of the EPCQ device to the FPGA. The data is shifted out on the falling edge of the DCLK signal.
DATA1	2	8	2	8	I/O	For AS x1 and x4 modes, use this pin as an output signal pin that serially transfers data out of the EPCQ device to the FPGA during read or configuration operations. The transition of the signal is on the falling edge of the DCLK signal. During the extended dual input fast write bytes or extended quad input fast write bytes operation, this pin acts as an input signal pin that serially transfers data into the EPCQ device. The data is latched on the rising edge of the DCLK signal. During extended dual input fast read or extended quad input fast read operations, this pin acts as an output signal pin that serially transfer data out of the EPCQ device to the FPGA. The data is shifted out on the falling edge of the DCLK signal. During read, configuration, or program operations, you can enable the EPCQ device by pulling the nCS signal low.

Table 27. EPCQ Device Pin Description—Preliminary (Part 2 of 3)

Pin Name	AS x1 Pin-Out Diagram		AS x4 Pin-Out Diagram		Pin Type	Description
	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package		
DATA2	—	—	3	9	I/O	For AS x1 mode, extended dual input fast write bytes operation and extended dual input fast read operation, this pin must connect to a 3.3-V power supply. For AS x4 mode, use this pin as an output signal that serially transfers data out of the EPCQ device to the FPGA during read or configuration operations. The transition of the signal is on the falling edge of the DCLK signal. During the extended quad input fast write bytes operation, this pin acts as an input pin that serially transfers data into the EPCQ device. The data is latched on the rising edge of the DCLK signal. During the extended quad input fast read operation, this pin acts as an output signal pin that serially transfers data out of the EPCQ device to the FPGA. The data is shifted out on the falling edge of the DCLK signal.
DATA3	—	—	7	1	I/O	For AS x1 mode, extended dual input fast write bytes operation and extended dual input fast read operation, this pin must connect to a 3.3-V power supply. For AS x4 mode, use this pin as an output signal that serially transfers data out of the EPCQ device to the FPGA during read or configuration operations. The transition of the signal is on the falling edge of the DCLK signal. During the extended quad input fast write bytes operation, this pin acts as an input pin that serially transfers data into the EPCQ device. The data is latched on the rising edge of the DCLK signal. During the extended quad input fast read operation, this pin acts as an output signal pin that serially transfers data out of the EPCQ device to the FPGA. The data is shifted out on the falling edge of the DCLK signal.
nCS	1	7	1	7	Input	The active low nCS input signal toggles at the beginning and end of a valid operation. When this signal is high, the device is deselected and the DATA pin is tri-stated. When this signal is low, the device is enabled and is in active mode. After power up, the EPCQ device requires a falling edge on the nCS signal before you begin any operation.

Table 27. EPCQ Device Pin Description—Preliminary (Part 3 of 3)

Pin Name	AS x1 Pin-Out Diagram		AS x4 Pin-Out Diagram		Pin Type	Description
	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package	Pin Number in 8-Pin SOIC Package	Pin Number in 16-Pin SOIC Package		
DCLK	6	16	6	16	Input	The FPGA provides the DCLK signal. This signal provides the timing for the serial interface. The data presented on the DATA0 pin is latched to the EPCQ device on the rising edge of the DCLK signal. The data on the DATA pin changes after the falling edge of the DCLK signal and is latched in to the FPGA on the next falling edge of the DCLK signal.
V_{CC}	3, 7, 8	1, 2, 9	8	2	Power	Connect the power pins to a 3.3-V power supply.
GND	4	10	4	10	Ground	Ground pin.

Device Package and Ordering Code

This section describes the package offered in EPCQ devices and the ordering codes for each EPCQ device.

Package

The EPCQ16 and EPCQ32 devices are available in 8-pin SOIC packages. The EPCQ64, EPCQ128, and EPCQ256 devices are available in 16-pin SOIC packages.

For a 16-pin SOIC package, you can migrate vertically from the EPCQ128 device to the EPCQ256 device. If you use the AS x1 configuration scheme, you can migrate to the EPCS16, EPCS64, or EPCS128 serial configuration devices.

For more information about migration to EPCS, refer to the *Serial Configuration (EPCS) Devices Datasheet*.

Ordering Code

Table 28 lists the ordering codes for EPCQ devices.

Table 28. EPCQ Device Ordering Codes

Device	Ordering Code (1)
EPCQ16	EPCQ16SI8N
EPCQ32	EPCQ32SI8N
EPCQ64	EPCQ64SI16N
EPCQ128	EPCQ128SI16N
EPCQ256	EPCQ256SI16N

Note to Table 28:

(1) N indicates that the device is lead free.

Document Revision History

Table 29 lists the revision history for this document.

Table 29. Document Revision History

Date	Version	Changes
July 2012	3.0	■ Added Table 3, Table 4, and Table 5 to include the address range for EPCQ16, EPCQ32, and EPCQ64 devices. ■ Added Table 9, Table 10, Table 11, Table 12, Table 13, and Table 14 to include the block protection bits for EPCQ16, EPCQ32, and EPCQ64 devices. ■ Added Figure 5, Figure 20 and Figure 21 to include EPCQ16 and EPCQ32 devices. ■ Updated the "Device Package and Ordering Code" section. ■ Updated Table 1, Table 2, Table 19, Table 20, Table 27, and Table 28 to include EPCQ16, EPCQ32, and EPCQ64 devices. ■ Updated the address bytes for the extended quad input fast write bytes operation in Table 8. ■ Updated Figure 22 and Figure 23 to include EPCQ64 devices.
January 2012	2.0	■ Added Figure 2. ■ Updated "Read Bytes Operation" and "Fast Read Operation" sections. ■ Updated Figure 1, Figure 3, Figure 4, Figure 7, and Figure 13. ■ Updated Table 5, Table 11, Table 12, and Table 14. ■ Minor text edits.
June 2011	1.0	Initial release.