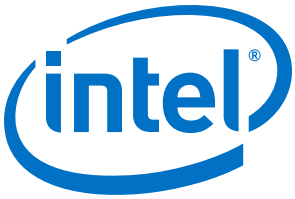
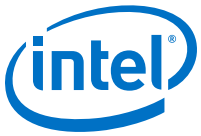


EPCQ-A Serial Configuration Device Datasheet

Subscribe
Send Feedback

CF52014 | 2018.04.11
Latest document on the web: PDF | HTML

Contents



1. EPCQ-A Serial Configuration Device Datasheet

Related Information

AN822: Intel® Configuration Device Migration Guideline

1.1. Supported Devices

Table 1. Supported Intel EPCQ-A Devices

Device	Memory Size (bits)	On-Chip Decompression Support	ISP Support	Cascading Support	Reprogrammable	Recommended Operating Voltage (V)
EPCQ4A	4,194,304	No	Yes	No	Yes	3.3
EPCQ16A	16,777,216	No	Yes	No	Yes	3.3
EPCQ32A	33,554,432	No	Yes	No	Yes	3.3
EPCQ64A	67,108,864	No	Yes	No	Yes	3.3
EPCQ128A	134,217,728	No	Yes	No	Yes	3.3

1.2. Features

EPCQ-A devices offer the following features:

- Serial or quad-serial FPGA configuration in devices that support active serial (AS) x1 or AS x4(1) configuration schemes
- Low cost, low pin count, and non-volatile memory
- 2.7-V to 3.6-V operation
- Available in 8-pin small-outline integrated circuit (SOIC) package for EPCQ4A, EPCQ16A, and EPCQ32A devices
- Available in 16-pin SOIC package for EPCQ64A and EPCQ128A devices
- Reprogrammable memory more than 100,000 program-erase 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
- Reprogrammable with an external microprocessor using the SRunner software driver
- In-system programming (ISP) support with the SRunner software driver

(1) AS x4 is not applicable for EPCQ4A.

- ISP support with Intel FPGA Download Cable II, Intel FPGA Download Cable, or Intel FPGA Ethernet Cable
- By default, the memory array is erased and the bits are set to 1
- More than 20-year data retention
- Supports JEDEC standard Serial Flash Discoverable Parameter (SFDP)

1.3. Operating Conditions

1.3.1. Absolute Maximum Ratings

Table 2. Absolute Maximum Ratings for EPCQ-A Devices

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supply voltage	With respect to GND	–0.6	4.6	V
V_I	DC input voltage	With respect to GND	–0.6	V_{CC}+0.4	V
T_{STG}	Storage temperature	No bias	–65	150	°C

1.3.2. Recommended Operating Conditions

Table 3. Recommended Operating Conditions for EPCQ-A Devices

Symbol	Parameter	Condition	Min	Max	Unit
V_{CC}	Supply voltage	(2)	2.7	3.6	V
T_A	Ambient temperature, Operating	For industrial use	-40	85	°C

1.3.3. DC Operating Conditions

Table 4. DC Operating Conditions for EPCQ-A Devices

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	V_{CC} - 0.2	—	V
V_{OL}	Low-level output voltage for EPCQ4A	I_{OL} = 100 µA	—	0.4	V
	Low-level output voltage for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A		—	0.2	V
I_I	Input leakage current	V_I =V_{CC} or GND	-2	2	µA
I_{OZ}	Tri-state output off-state current	V_O = V_{CC} or GND	-2	2	µA

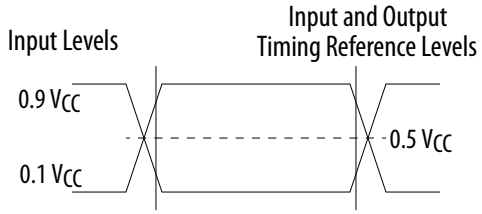
(2) V_{CC} voltage during a Read operation can operate across the min and max range but should not exceed ±10% of the programming (erase/write) voltage.

1.3.4. AC Measurement Conditions

Table 5. AC Measurement Conditions for EPCQ-A Devices

Symbol	Parameter	Min	Max	Unit
C_L	Load Capacitance	—	30	pF
T_R, T_F	Input Rise and Fall Times	—	5	ns
V_{IN}	Input Pulse Voltages	0.1 V_{CC} to 0.9 V_{CC}		V
IN	Input Timing Reference Voltages	0.3 V_{CC} to 0.7 V_{CC}		V
Out	Output Timing Reference Voltages	0.5 V_{CC} to 0.5 V_{CC}		V

Figure 1. AC Measurement I/O Waveform



1.3.5. ICC Supply Current

Table 6. I_{CC} Supply Current AC Measurement

Symbol	Parameter	Condition	Min	Max	Unit
I_{CC0}	V_{CC} supply current for EPCQ4A, EPCQ16A, EPCQ32A, and EPCQ64A	Standby	10	50	µA µA
	V_{CC} supply current for EPCQ128A		10	60	
I_{CC1}	V_{CC} supply current for EPCQ4A	During active power mode	1	5	mA
	V_{CC} supply current for EPCQ16A, EPCQ32A, and EPCQ64A		1	15	
	V_{CC} supply current for EPCQ128A		1	20	

1.3.6. Capacitance

Table 7. Capacitance for EPCQ-A Devices

Capacitance is sample-tested only at T_A = 25 °C and at V_{CC} = 3.0 V.

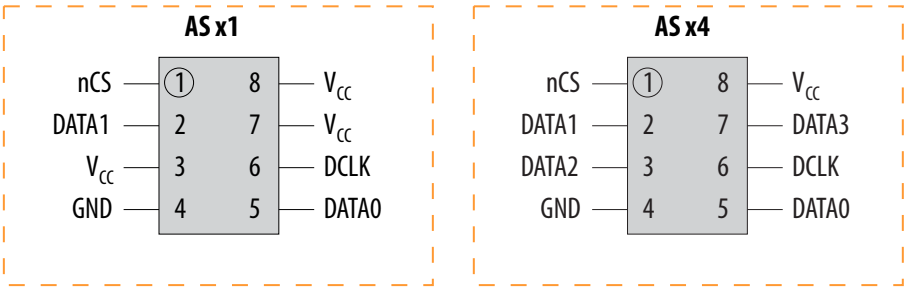
Symbol	Parameter	Condition	Min	Max	Unit
C_{IN}	Input pin capacitance	V_{IN} =0 V	—	6	pF
C_{OUT}	Output pin capacitance	V_{OUT} =0 V	—	8	pF

1.4. Pin Information

1.4.1. Pin-Out Diagram for EPCQ4A, EPCQ16A and EPCQ32A Devices

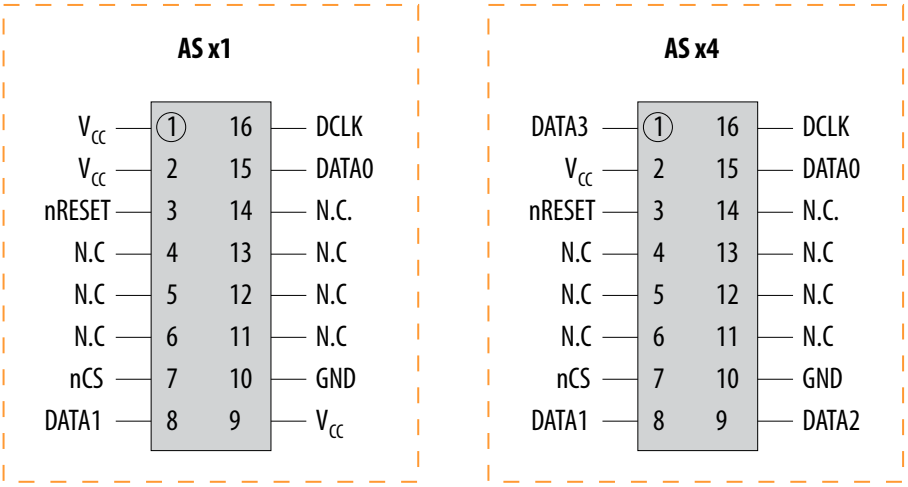
Figure 2. AS x1 and AS x4 Pin-Out Diagrams for EPCQ4A, EPCQ16A, and EPCQ32A Devices

Note: EPCQ4A supports AS x1 only.



1.4.2. Pin-Out Diagram for EPCQ64A and EPCQ128A Devices

Figure 3. AS x1 and AS x4 Pin-Out Diagrams for EPCQ64A and EPCQ128A Devices



Notes:

N.C pins must be left unconnected.

There is an internal pull-up resistor for the dedicated nRESET pin. If the reset function is not needed, connect this pin to Vcc or leave it unconnected.

1.4.3. EPCQ-A Device Pin Description

Table 8. EPCQ-A Device Pin Description

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-A device. During write or program operations, data are latched at rising edges of the DCLK signal. This pin is equivalent to the ASDI pin in EPCS devices. 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-A device. The data are latched at rising edges 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-A device to the FPGA. The data is shifted out at falling edges of the DCLK signal. During the quad input fast write bytes operation, this pin acts as an input pin that serially transfers data into the EPCQ-A device. The data is latched at rising edges 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-A device to the FPGA. The data is shifted out at falling edges 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-A device to the FPGA during read or configuration operations. The transition of the signal is at falling edges of the DCLK signal. This pin is equivalent to the DATA pin in EPCS devices. During the quad input fast write bytes operation, this pin acts as an input signal pin that serially transfers data into the EPCQ-A device. The data is latched at rising edges of the DCLK signal. During extended quad input fast read operations, this pin acts as an output signal pin that serially transfer data out of the EPCQ-A device to the FPGA. The data is shifted out at falling edges of the DCLK signal. During read, configuration, or program operations, you can enable the EPCQ-A device by pulling the nCS signal low.
DATA2	—	—	3	9	I/O	For AS x1 mode, this pin must connect to V_{CC}. For AS x4 mode, use this pin as an output signal that serially transfers data out of the EPCQ-A device to the FPGA during read or configuration operations. The transition of the signal is at falling edges 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-A device to the FPGA. The data is shifted out at falling edges of the DCLK signal.

continued...

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		
DATA3	—	—	7	1	I/O	For AS x1 mode, this pin must connect to V_{CC}. For AS x4 mode, use this pin as an output signal that serially transfers data out of the EPCQ-A device to the FPGA during read or configuration operations. The transition of the signal is at falling edges 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-A device to the FPGA. The data is shifted out at falling edges 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[3:0] pins are tri-stated. When this signal is low, the device is enabled and is in active mode. After power up, the EPCQ-A device requires a falling edge on the nCS signal before you begin any operation.
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 DATA[3:0] pins are latched to the EPCQ-A device at rising edges of the DCLK signal. The data on the DATA[3:0] pins change after the falling edge of the DCLK signal and are latched in to the FPGA on the next falling edge of the DCLK signal.
nRESET	—	3	—	3	Input	Dedicated hardware reset pin. When it's driven low for a minimum period of ~1µS, the EPCQ-A device will terminate any external or internal operations and return to its power-on state. There is an internal pull-up resistor for the dedicated nRESET pin on the SOIC-16 package. If the reset function is not needed, you can connect it to V_{CC} or leave it unconnected.
V_{CC}	8	2	8	2	Power	Connect the power pins to a 3.3-V power supply.
GND	4	10	4	10	Ground	Ground pin.

1.5. Device Package and Ordering Code

1.5.1. Package

The EPCQ4A, EPCQ16A, and EPCQ32A devices are available in 8-pin SOIC packages. The EPCQ64A and EPCQ128A devices are available in 16-pin SOIC packages.

1.5.2. Ordering Code

Table 9. EPCQ-A Device Ordering Codes

Device	Ordering Code(3)
EPCQ4A	EPCQ4ASI8N
EPCQ16A	EPCQ16ASI8N
EPCQ32A	EPCQ32ASI8N
EPCQ64A	EPCQ64ASI16N
EPCQ128A	EPCQ128ASI16N

1.6. Memory Array Organization

Table 10. Supported Memory Array Organization in EPCQ-A Devices

Details	EPCQ4A	EPCQ16A	EPCQ32A	EPCQ64A	EPCQ128A
Bytes	524,288 bytes [4 megabits (Mb)]	2,097,152 bytes [16 Mb]	4,194,304 bytes (32 Mb)	8,388,608 bytes (64 Mb)	16,777,216 bytes (128 Mb)
Number of sectors	8	32	64	128	256
Bytes per sector	65,536 bytes [512 kilobits (Kb)]				
Total numbers of subsectors (4)	128	512	1,024	2,048	4,096
Bytes per subsector	4,096 bytes (32 Kb)				
Pages per sector	256				
Total number of pages	2,048	8,192	16,384	32,768	65,536
Bytes per page	256 bytes				

1.6.1. Address Range for EPCQ4A

Table 11. Address Range for Sectors 7..0 and Subsectors 127..0 in EPCQ4A Devices

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
7	127	7F000	7FFFF
	126	7E000	7EFFF
	..	..	..
	114	72000	72FFF
	113	71000	71FFF

continued...

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

(4) Every sector is further divided into 16 subsectors with 4 KB of memory. Therefore, there are 128 (32 x 16) subsectors for the EPCQ4A device, 512 (32 x 16) subsectors for the EPCQ16A device, 1,024 (64 x 16) subsectors for the EPCQ32A device, 2,048 (128 x 16) subsectors for the EPCQ64A device, and 4,096 (256 x 16) subsectors for the EPCQ128A device.

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
	112	70000	70FFF
6	111	6F000	6FFFF
	110	6E000	6EFFF
	..	..	..
	98	62000	62FFF
	97	61000	61FFF
	96	60000	60FFF
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

1.6.2. Address Range for EPCQ16A

Table 12. Address Range for Sectors 31..0 and Subsectors 511..0 in EPCQ16A Devices

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

continued...

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
	480	1E0000	1E0FFF
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

1.6.3. Address Range for EPCQ32A

Table 13. Address Range for Sectors 63..0 and Subsectors 1023..0 in EPCQ32A Devices

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

continued...

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

1.6.4. Address Range for EPCQ64A

Table 14. Address Range for Sectors 127..0 and Subsectors 2047..0 in EPCQ64A Devices

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
63	1023	3FF000	3FFFFF
	1022	3FE000	3FEFFF
	.	.	.
	1010	3F2000	3F2FFF
	1009	3F1000	3F1FFF
	1008	3F0000	3F0FFF
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	.	.	.

continued...

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
	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

1.6.5. Address Range for EPCQ128A

Table 15. Address Range for Sectors 255..0 and Subsectors 4095..0 in EPCQ128A Devices

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

continued...

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
	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
62	1007	3EF000	3EFFFF
	1006	3EE000	3EEFFF
	.	.	.
	994	3E2000	3E2FFF
	993	3E1000	3E1FFF

continued...

Sector	Subsector	Address Range (Byte Addresses in HEX)	
		Start	End
	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

1.7. Memory Operations

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

1.7.1. Timing Requirements

When the active low chip select (nCS) signal is driven low, shift in the operation code into the EPCQ-A device using theDATA0 pin. Each operation code bit is latched into the EPCQ-A device at rising edges of the DCLK signal.

While executing an operation, shift in the desired operation code, followed by the address or data bytes. See related information for more information about the address and data bytes. 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[3:0] pins. 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.

1.8. Status Register

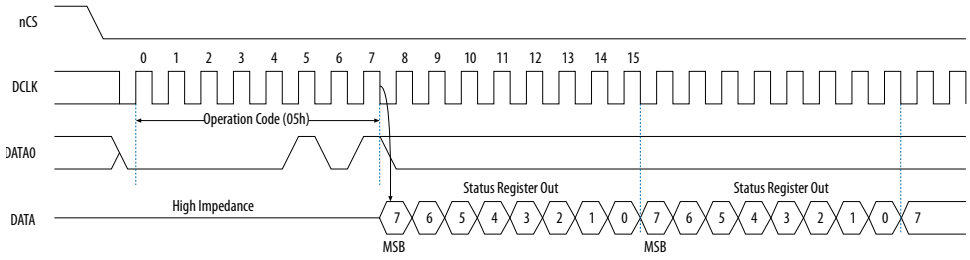
Table 16. Status Register Bits

Bit	R/W	Default Value	Name	Value	Description
7	R/W	0[5]	Reserved		
6	R/W	0[5]	Reserved		
5	R/W	0	TB (Top/Bottom Bit)	• 1=Protected area starts from the bottom of the memory array. • 0=Protected area starts from the top of the memory array.	Determine that the protected area starts from the top or bottom of the memory array.
4	R/W	0	BP2[6]	Table 17 on page 18 through Table 21 on page 20 list the protected area with reference to the block protect bits.	Determine the area of the memory protected from being written or erased unintentionally.
3	R/W	0	BP1[6]		
2	R/W	0	BP0[6]		
1	R	0	WEL (Write Enable Latch Bit)	• 1=Allows the following operation to run: — Write Bytes — Write Status Register — Erase Bulk — Erase Sector • 0=Rejects the above mentioned operations.	Allows or rejects certain operation to run.
0	R	0	WIP (Write in Progress Bit)	• 1=One of the following operation is in progress: — Write Status Register — Write Bytes — Erase • 0=no write or erase cycle in progress	Indicates if there is a command in progress.

1.8.1. Read Status Operation

The status register can be read continuously and at anytime, including during a write or erase operations.

Figure 4. Read Status Operation Timing Diagram



(5) Do not program these bits to 1.

(6) The erase bulk and erase die 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 17. Block Protection Bits in EPCQ4A

Status Register Content				Memory Content	
TB Bit	BP2 Bit	BP1 Bit	BP0 Bit	Protected Area	Unprotected Area
x	0	0	0	None	All sectors
0	0	0	1	Sector 7	Sectors (0 to 6)
0	0	1	0	Sectors (6 to 7)	Sectors (0 to 5)
0	0	1	1	Sectors (4 to 7)	Sectors (0 to 3)
1	0	0	1	Sector 0	Sectors (1 to 7)
1	0	1	0	Sectors (0 to 1)	Sectors (2 to 7)
1	0	1	1	Sectors (0 to 3)	Sectors (4 to 7)
x	1	x	x	All sectors	None

Table 18. Block Protection Bits in EPCQ16A

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	Sector 31	Sectors (0 to 30)
0	0	1	0	Sectors (30 to 31)	Sectors (0 to 29)
0	0	1	1	Sectors (28 to 31)	Sectors (0 to 27)
0	1	0	0	Sectors (24 to 31)	Sectors (0 to 23)
0	1	0	1	Sectors (16 to 31)	Sectors (0 to 15)
0	1	1	0	All sectors	None
0	1	1	1	All sectors	None
1	0	0	0	None	All sectors
1	0	0	1	Sector 0	Sectors (1 to 31)
1	0	1	0	Sectors (0 to 1)	Sectors (2 to 31)
1	0	1	1	Sectors (0 to 3)	Sectors (4 to 31)
1	1	0	0	Sectors (0 to 7)	Sectors (8 to 31)
1	1	0	1	Sectors (0 to 15)	Sectors (16 to 31)
1	1	1	0	All sectors	None
1	1	1	1	All sectors	None

Table 19. Block Protection Bits in EPCQ32A

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	Sector 63	Sectors (0 to 62)
0	0	1	0	Sectors (62 to 63)	Sectors (0 to 61)

continued...

Status Register Content				Memory Content	
TB Bit	**BP2 Bit**	**BP1 Bit**	**BP0 Bit**	**Protected Area**	**Unprotected Area**
0	0	1	1	Sectors (60 to 63)	Sectors (0 to 59)
0	1	0	0	Sectors (56 to 63)	Sectors (0 to 55)
0	1	0	1	Sectors (48 to 63)	Sectors (0 to 47)
0	1	1	0	Sectors (32 to 63)	Sectors (0 to 31)
0	1	1	1	All sectors	None
1	0	0	0	None	All sectors
1	0	0	1	Sector 0	Sectors (1 to 63)
1	0	1	0	Sectors (0 to 1)	Sectors (2 to 63)
1	0	1	1	Sectors (0 to 3)	Sectors (4 to 63)
1	1	0	0	Sectors (0 to 7)	Sectors (8 to 63)
1	1	0	1	Sectors (0 to 15)	Sectors (16 to 63)
1	1	1	0	Sectors (0 to 31)	Sectors (32 to 63)
1	1	1	1	All sectors	None

Table 20. Block Protection Bits in EPCQ64A

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	Sectors (126 to 127)	Sectors (0 to 125)
0	0	1	0	Sectors (124 to 127)	Sectors (0 to 123)
0	0	1	1	Sectors (120 to 127)	Sectors (0 to 119)
0	1	0	0	Sectors (112 to 127)	Sectors (0 to 111)
0	1	0	1	Sectors (96 to 127)	Sectors (0 to 95)
0	1	1	0	Sectors (64 to 127)	Sectors (0 to 63)
0	1	1	1	All sectors	None
1	0	0	0	None	All sectors
1	0	0	1	Sectors (0 to 1)	Sectors (2 to 127)
1	0	1	0	Sectors (0 to 3)	Sectors (4 to 127)
1	0	1	1	Sectors (0 to 7)	Sectors (8 to 127)
1	1	0	0	Sectors (0 to 15)	Sectors (16 to 127)
1	1	0	1	Sectors (0 to 31)	Sectors (32 to 127)
1	1	1	0	Sectors (0 to 63)	Sectors (64 to 127)
1	1	1	1	All sectors	None

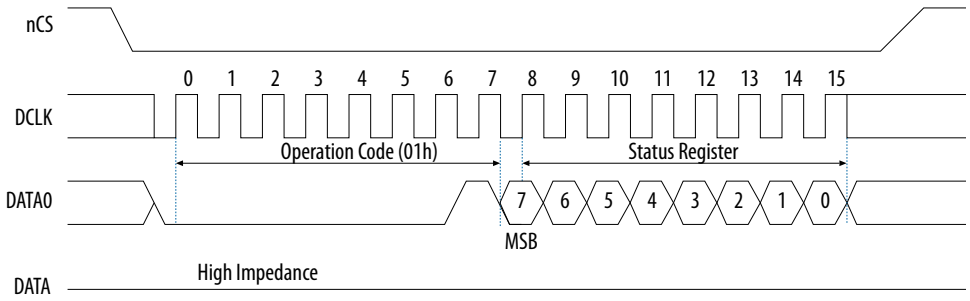
Table 21. Block Protection Bits in EPCQ128A

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	Sectors (252 to 255)	Sectors (0 to 251)
0	0	1	0	Sectors (248 to 255)	Sectors (0 to 247)
0	0	1	1	Sectors (240 to 255)	Sectors (0 to 239)
0	1	0	0	Sectors (224 to 255)	Sectors (0 to 223)
0	1	0	1	Sectors (192 to 255)	Sectors (0 to 191)
0	1	1	0	Sectors (128 to 255)	Sectors (0 to 127)
0	1	1	1	All sectors	None
1	0	0	0	None	All sectors
1	0	0	1	Sectors (0 to 3)	Sectors (4 to 255)
1	0	1	0	Sectors (0 to 7)	Sectors (8 to 255)
1	0	1	1	Sectors (0 to 15)	Sectors (16 to 255)
1	1	0	0	Sectors (0 to 31)	Sectors (32 to 255)
1	1	0	1	Sectors (0 to 63)	Sectors (64 to 255)
1	1	1	0	Sectors (0 to 127)	Sectors (128 to 255)
1	1	1	1	All sectors	None

1.8.2. 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. 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 5. 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 10 ms for all EPCQ-A devices and is guaranteed to be less than 15 ms. For details about t_{WS}, refer to the related information below. 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. Write in progress bit is 1 during the self-timed write status cycle and 0 when it is complete.

1.9. Summary of Operation Codes

Operation	Operation Code[7]	Address Bytes	Dummy Cycles	Data Bytes	DCLK f_{MAX} (MHz)
Read status	05h	0	0	1 to infinite[8]	100
Read bytes	03h	3	0	1 to infinite[8]	50
Read device identification	9Fh	0	2	1	100
Read silicon identification	ABh	0	3	1	100
Fast read	0Bh	3	8	1 to infinite[8]	100
Extended dual input fast read	BBh	3	4	1 to infinite[8]	100
Extended quad input fast read [9]	EBh	3	6	1 to infinite[8]	100
Write enable	06h	0	0	0	100
Write disable	04h	0	0	0	100
Write status	01h	0	0	1	100
Write bytes	02h	3	0	1 to 256[10]	100
Quad input fast write bytes[9]	32h	3	0	1 to 256[10]	100
Erase bulk	C7h	0	0	0	100
Erase sector	D8h	3	0	0	100
Erase subsector	20h	3	0	0	100
Read SFDP register[9]	5Ah	3	8	1 to 256	100

1.9.1. Read Bytes Operation (03h)

When you execute the read bytes operation, you first drive the nCS pin low and shift in the read bytes operation code, followed by a 3-byte address (A[23..0]). Each address bit must be latched in at rising edges 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**),

(7) List MSB first and LSB last.

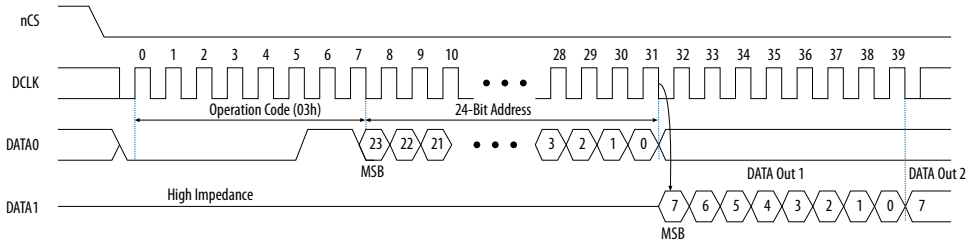
(8) The status register or data is read out at least once and is continuously read out until the nCS pin is driven high.

(9) This operation is not applicable for EPCQ4A.

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

the content is shifted out serially beginning with the LSB. Each data bit is shifted out at falling edges of the DCLK signal. The maximum DCLK frequency during the read bytes operation is 50 MHz.

Figure 6. Read Bytes Operation Timing Diagram

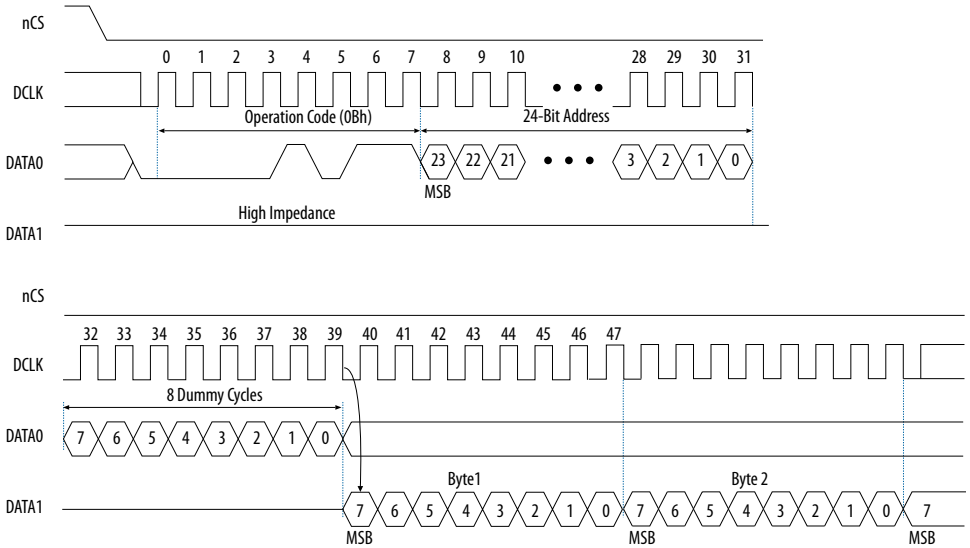


The first byte address can be at any location. The device automatically increases the address to the next higher address after shifting out each byte of data. Therefore, the device can read the whole memory with a single read bytes operation. When the device reaches the highest address, the address counter 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.

1.9.2. Fast Read Operation (0Bh)

When you execute the fast read operation, you first shift in the fast read operation code, followed by a 3-byte address (`A[23..0]`), and 8 dummy cycles with each bit being latched-in at rising edges 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 at falling edges of the DCLK signal.

Figure 7. Fast Read Operation Timing Diagram



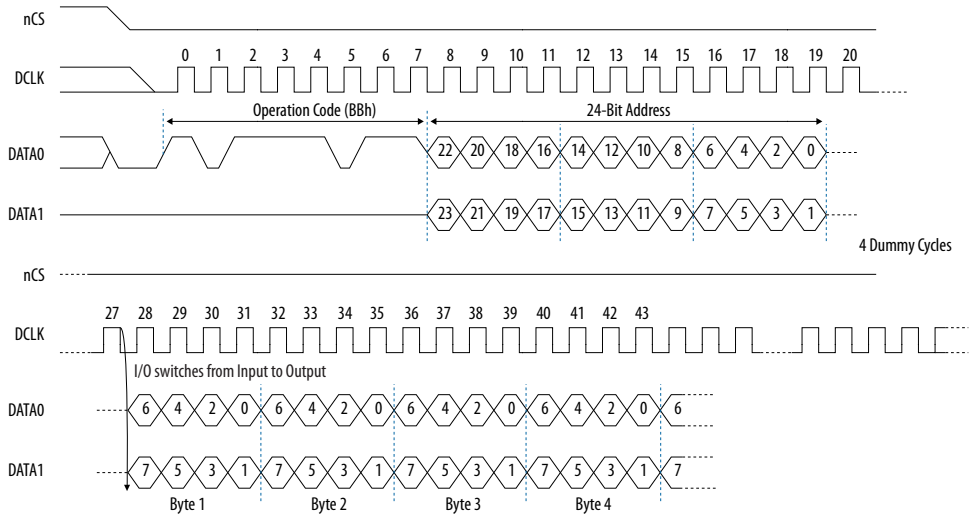
The first byte address can be at any location. The device automatically increases the address to the next higher address after shifting out each byte of data. Therefore, the device can read the whole memory with a single fast read operation. When the device reaches the highest address, the address counter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

1.9.3. Extended Dual Input Fast Read Operation (BBh)

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.

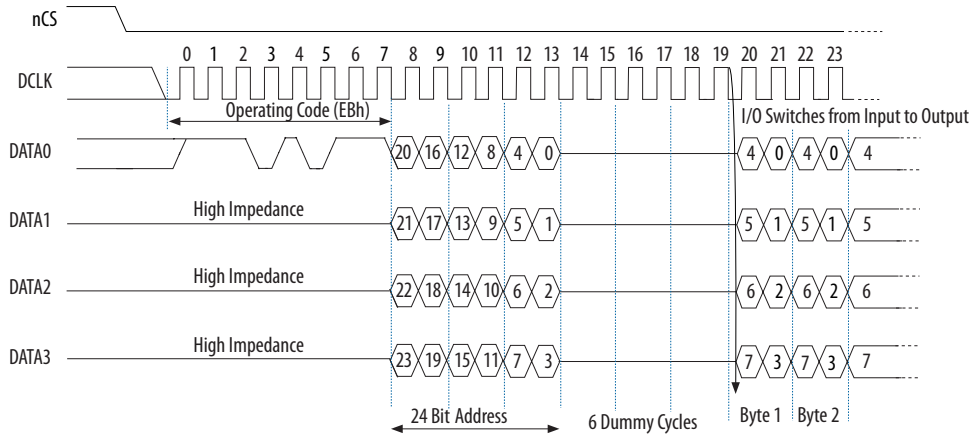
Figure 8. Extended Dual Input Fast Read Operation Timing Diagram



1.9.4. Extended Quad Input Fast Read Operation (EBh)

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 9. Extended Quad Input Fast Read Operation



1.9.5. Read Device Identification Operation (9Fh)

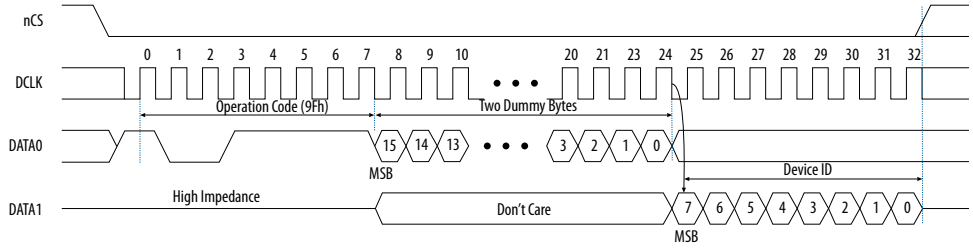
This operation reads the 8-bit device identification of the EPCQ-A device from the DATA1 output pin. If this operation is shifted in while an erase or write cycle is in progress, the operation is not executed and does not affect the erase or write cycle in progress.

Table 22. EPCQ-A Device Identification

EPCQ-A Device	Device ID (Binary Value)
EPCQ4A	b'0001 0011
EPCQ16A	b'0001 0101
EPCQ32A	b'0001 0110
EPCQ64A	b'0001 0111
EPCQ128A	b'0001 1000

The 8-bit device identification of the EPCQ-A device is shifted out on the DATA1 pin at falling edges of the DCLK signal.

Figure 10. Read Device Identification Operation Timing Diagram



1.9.6. Read Silicon Identification Operation (ABh)

This operation reads the 8-bit silicon ID of the EPCQ-A device from the DATA1 output pin. If this operation is shifted in during an erase or write cycle, it is ignored and does not affect the cycle that is in progress.

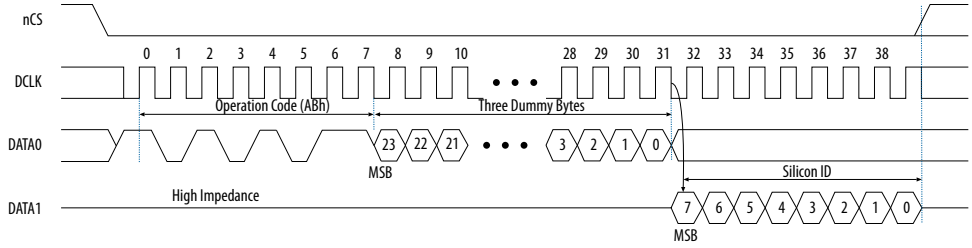
Note: This operation is applicable to EPCQ4A, EPCQ16A and EPCQ64A devices only.

The device implements the read silicon ID operation by driving the nCS signal low and then shifting in the read silicon ID operation code, followed by three dummy bytes on the DATA0 pin. The 8-bit silicon ID of the EPCQ-A device is then shifted out on the DATA1 pin at falling edges of the DCLK signal. The device can terminate the read silicon ID operation by driving the nCS signal high after reading the silicon ID at least one time. Sending additional clock cycles on DCLK while nCS is driven low can cause the silicon ID to be shifted out repeatedly.

Table 23. EPCQ-A Silicon Identification

EPCQ-A Device	Silicon ID (Binary Value)
EPCQ4A	b'0001 0010
EPCQ16A	b'0001 0100
EPCQ64A	b'0001 0110

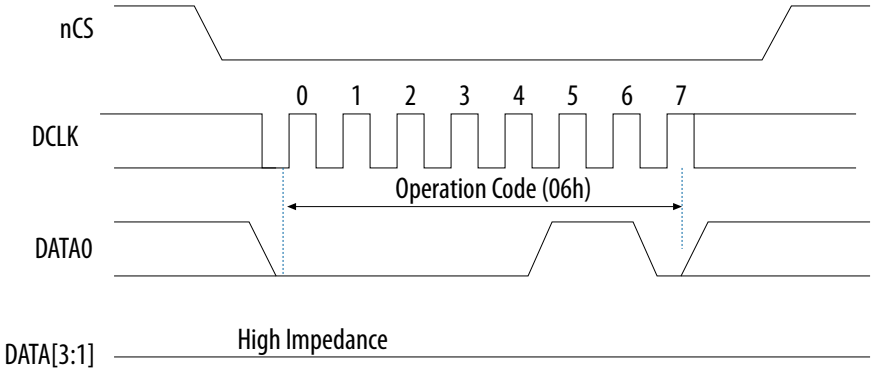
Figure 11. Read Silicon Identification Operation Timing Diagram



1.9.7. Write Enable Operation (06h)

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, and quad input fast write bytes operations.

Figure 12. Write Enable Operation Timing Diagram

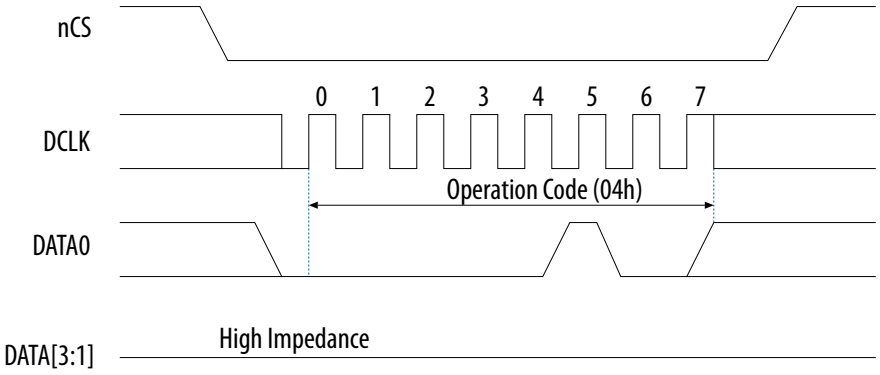


1.9.8. Write Disable Operation (04h)

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
- Quad input fast write bytes operation completion

Figure 13. Write Disable Operation Timing Diagram

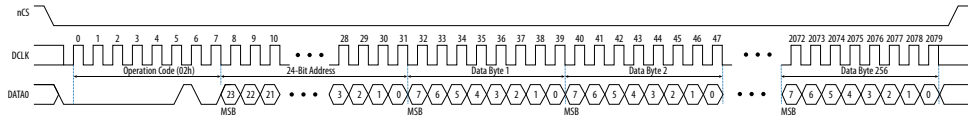


1.9.9. Write Bytes Operation (02h)

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 address (A[23..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 14. Write Bytes Operation Timing Diagram



If more than 256 data bytes are shifted into the EPCQ-A 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-A device, they are guaranteed to be written at the specified addresses and the other bytes of the same page are not affected.

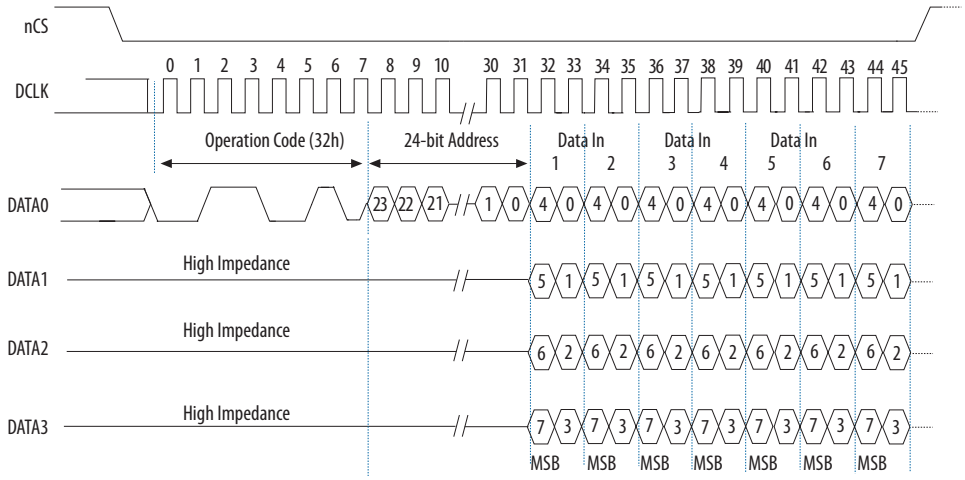
The device initiates a self-timed write cycle immediately after the nCS signal is driven high. For details about the self-timed write cycle time, refer to t_{WB} in the related information below. You must account for this amount of delay before another page of memory is written. Alternatively, you can 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.

Note: You must erase all the memory bytes of EPCQ-A 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

1.9.10. Quad Input Fast Write Bytes Operation (32h)

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

Figure 15. Quad Input Fast Write Bytes Operation Timing Diagram

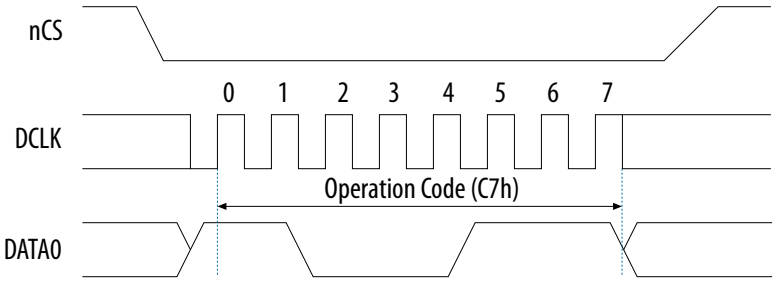


1.9.11. Erase Bulk Operation (C7h)

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. Erase Bulk Operation Timing Diagram



The device initiates a self-timed erase bulk cycle immediately after the nCS signal is driven high. For details about the self-timed erase bulk cycle time, refer to t_{EB} in the related information below.

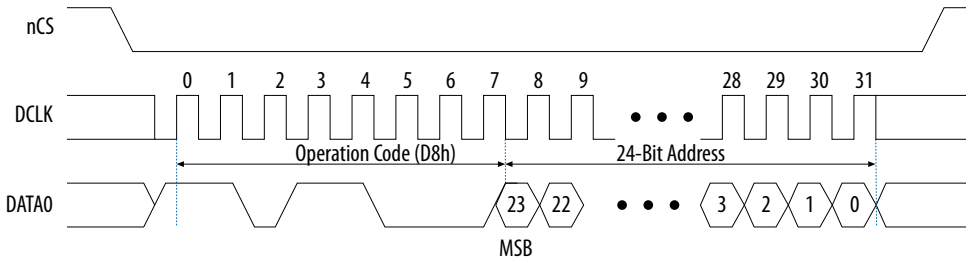
You must account for this delay before accessing the memory contents. Alternatively, you can check the write in progress bit in the status register by executing the read status operation while the self-timed erase cycle is in progress. The write in progress bit is set to 1 during the self-timed erase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

1.9.12. Erase Sector Operation (D8h)

The erase sector operation allows you to erase a certain sector in the EPCQ-A 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 address (A[23..0]) of the chosen sector on the DATA0 pin. The 3-byte address for the erase sector operation can be any address inside the specified sector. Drive the nCS signal high after the eighth bit of the erase sector operation code has been latched in.

Figure 17. Erase Sector Operation Timing Diagram



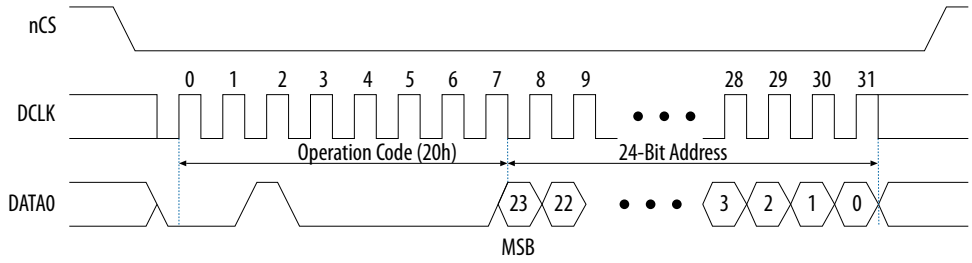
The device initiates a self-timed erase sector cycle immediately after the nCS signal is driven high. For details about the self-timed erase sector cycle time, refer to t_{ES} in the related information below. You must account for this amount of delay before another page of memory is written. Alternatively, 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.

1.9.13. Erase Subsector Operation (20h)

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

When you execute the erase subsector operation, you must first shift in the erase subsector operation code, followed by the 3-byte address (A[23..0]) of the chosen subsector on the DATA0 pin. The 3-byte address for the erase subsector operation can be any address inside the specified subsector. For details about the subsector address range, refer to the related information below. Drive the nCS signal high after the eighth bit of the erase subsector operation code has been latched in.

Figure 18. Erase Subsector Operation Timing Diagram



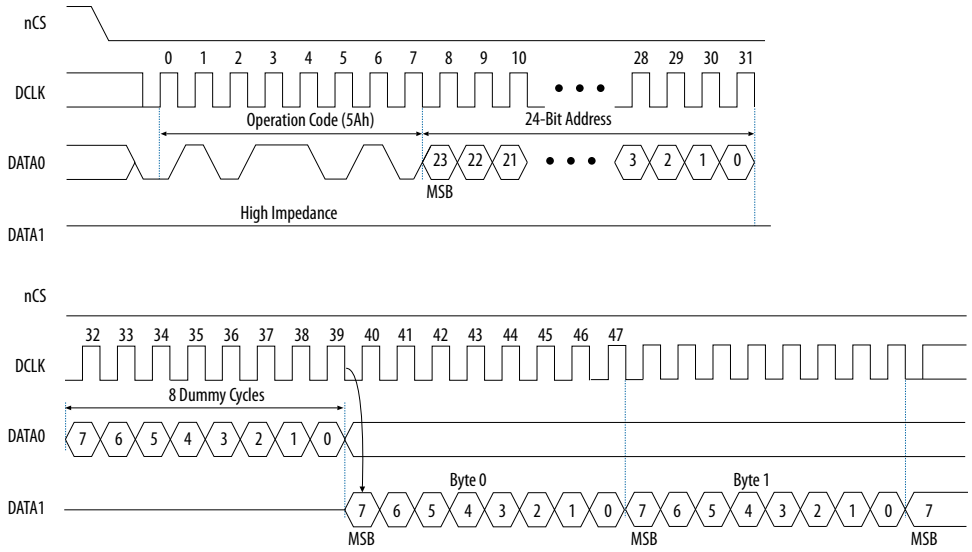
The device initiates a self-timed erase subsector cycle immediately after the nCS signal is driven high. For details about the self-timed erase subsector cycle time, refer to related the information below. You must account for this amount of delay before another page of memory is written. Alternatively, 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.

1.9.14. Read SFDP Register Operation (5Ah)

The 256-byte SFDP register contains information about device configurations, available operations and other features.

The Read SFDP Register operation is compatible with the JEDEC SFDP standard, JESD216A. For SFDP register values and descriptions, please refer to Appendix: SFDP Register Definitions on page 34.

Figure 19. Read SFDP Register Operation Timing Diagram



Initiate the Read SFDP operation by driving the nCS pin low and shifting the operations code followed by a 3-byte address into the DATA0 pin. The 3-byte address content:

- A[23..8] = 0
- A[7..0] = Defines the starting byte address for the 256-byte SFDP register

Eight dummy cycles are required before the SFDP register contents are shifted out on the falling edge of the 40th DCLK with the most significant bit (MSB) first.

Related Information

Appendix: SFDP Register Definitions on page 34

1.10. Power Mode

EPCQ-A 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-A 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-A 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.

1.11. Timing Information

1.11.1. Write Operation Timing

Figure 20. Write Operation Timing Diagram

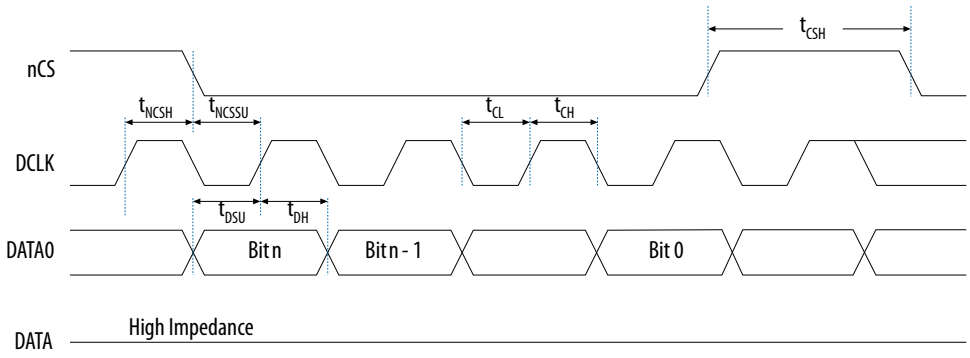


Table 24. Write Operation Parameters

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}	DCLK high time for EPCQ4A.	4	—	—	ns
	DCLK high time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	3.4			
t_{CL}	DCLK low time for EPCQ4A.	4	—	—	ns
	DCLK low time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	3.4			
t_{NCSSU}	Chip select (nCS) setup time	5	—	—	ns
t_{NCSH}	Chip select (nCS) hold time	5	—	—	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 for EPCQ4A.	5	—	—	ns
	DATA[] hold time after the rising edge on DCLK for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	3			
t_{CSH}	Chip select (nCS) high time for EPCQ4A.	100	—	—	ns
	Chip select (nCS) high time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	10 or 50[11]			
t_{WB}[12] [13]	Write bytes cycle time for EPCQ4A.	—	0.4	0.8	ms

continued...

(11) 10 ns for read and 50 ns for write, erase or program.

Symbol	Parameter	Min	Typical	Max	Unit
	Write bytes cycle time for EPCQ16A.		0.4	3	
	Write bytes cycle time for EPCQ32A and EPCQ128A.		0.7	3	
	Write bytes cycle time for EPCQ64A.		0.8	3	
t_{WS} (12)	Write status cycle time	—	10	15	ms
t_{EB} (12)	Erase bulk cycle time for EPCQ4A	—	1	4	s
	Erase bulk cycle time for EPCQ16A		5	25	
	Erase bulk cycle time for EPCQ32A		10	50	
	Erase bulk cycle time for EPCQ64A		20	100	
	Erase bulk cycle time for EPCQ128A		40	200	
t_{ES} (12)	Erase sector cycle time for EPCQ4A.	—	150	1000	ms
	Erase sector cycle time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.			2000	
t_{ESS} (12)	Erase subsector cycle time for EPCQ4A.	—	30	300	ms
	Erase subsector cycle time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.		45	400	

1.11.2. Read Operation Timing

Figure 21. Read Operation Timing Diagram

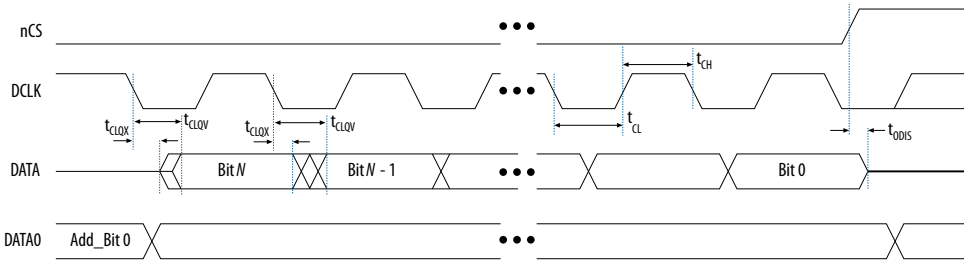


Table 25. Read Operation Parameters

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 for EPCQ4A.	4 or 6 (14)	—	ns

continued...

(12) The Write Operation Timing Diagram does not show these parameters.

(13) The t_{WB} parameter is for a complete page write operation.

(14) 4 ns for fast read and 6 ns for read.

Symbol	Parameter	Min	Max	Unit
	DCLK high time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	3.4 or 9[15]	—	ns
t_{CL}	DCLK low time for EPCQ4A.	4 or 6[14]	—	ns
	DCLK low time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	3.4 or 9[15]		ns
t_{ODIS}	Output disable time after read	—	7	ns
t_{CLQV}	Clock low to output valid for EPCQ4A.	—	8	ns
	Clock low to output valid for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	—	6	
t_{CLQX}	Output hold time for EPCQ4A.	0	—	ns
	Output hold time for EPCQ16A, EPCQ32A, EPCQ64A, and EPCQ128A.	1.5	—	

1.12. Programming and Configuration File Support

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

You can program the EPCQ-A device in-system by an external microprocessor using the SRunner software driver. The SRunner software driver is developed for embedded EPCQ-A device programming that you can customize to fit in different embedded systems. The SRunner software driver reads **.rpd** files and writes to the EPCQ-A devices. The programming time is comparable to the Intel Quartus Prime 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-A device is different from the other data and address bytes.

During the ISP of an EPCQ-A device using the Intel FPGA download cables, 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-A device. When programming is complete, the download cable releases the interface pins of the EPCQ-A device and the nCE pin of the FPGA and pulses the nCONFIG signal to start the configuration process.

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

(15) 3.4 ns for fast read and 9 ns for read.

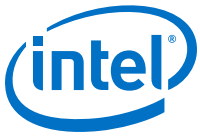
Related Information

Using the Intel FPGA Serial Flash Loader IP Core with the Intel Quartus Prime Software

1.13. Appendix: SFDP Register Definitions

Address	EPCQ16A	EPCQ32A	EPCQ64A	EPCQ128A
00H	53h	53h	53h	53h
01H	46h	46h	46h	46h
02H	44h	44h	44h	44h
03H	50h	50h	50h	50h
04H	05h	05h	05h	05h
05H	01h	01h	01h	01h
06H	00h	00h	00h	00h
07H	FFh	FFh	FFh	FFh
08H	00h	00h	00h	00h
09H	05h	05h	05h	05h
0AH	01h	01h	01h	01h
0BH	10h	10h	10h	10h
0CH	80h	80h	80h	80h
0DH	00h	00h	00h	00h
0EH	00h	00h	00h	00h
0FH	FFh	FFh	FFh	FFh
10H … 7FH	FFh	FFh	FFh	FFh
80H	E5h	E5h	E5h	E5h
81H	20H	20H	20H	20h
82H	F9h	F9h	F9h	F9h
83H	FFh	FFh	FFh	FFh
84H	FFh	FFh	FFh	FFh
85H	FFh	FFh	FFh	FFh
86H	FFh	FFh	FFh	FFh
87H	00h	01h	03h	07h
88H	44h	44h	44h	44h
89H	EBh	EBh	EBh	EBh
8AH	08H	08H	08H	08H
8BH	6Bh	6Bh	6Bh	6Bh
8CH	08h	08h	08h	08h

continued...



Address	EPCQ16A	EPCQ32A	EPCQ64A	EPCQ128A
8DH	3Bh	3Bh	3Bh	3Bh
8EH	42h	42h	42h	42h
8FH	BBh	BBh	BBh	BBh
90H	FEh	FEh	FEh	FEh
91H	FFh	FFh	FFh	FFh
92H	FFh	FFh	FFh	FFh
93H	FFh	FFh	FFh	FFh
94H	FFh	FFh	FFh	FFh
95H	FFh	FFh	FFh	FFh
96H	00h	00h	00h	00h
97H	00h	00h	00h	00h
98H	FFh	FFh	FFh	FFh
99H	FFh	FFh	FFh	FFh
9AH	40h	40h	40h	40h
9BH	EBh	EBh	EBh	EBh
9CH	0Ch	0Ch	0Ch	0Ch
9DH	20h	20h	20h	20h
9EH	0Fh	0Fh	0Fh	0Fh
9FH	52h	52h	52h	52h
A0H	10h	10h	10h	10h
A1H	D8h	D8h	D8h	D8h
A2H	00h	00h	00h	00h
A3H	00h	00h	00h	00h
A4H	36h	36h	36h	36h
A5H	02h	02h	02h	02h
A6H	A6h	A6h	A6h	A6h
A7H	00h	00h	00h	00h
A8H	82h	82h	82h	82h
A9H	EAh	EAh	EAh	EAh
AAH	14h	14h	14h	14h
ABH	B3h	C2h	C4h	C9h
ACH	E9h	E9h	E9h	E9h
ADH	63h	63h	63h	63h
AEH	76h	76h	76h	76h

continued...

Address	EPCQ16A	EPCQ32A	EPCQ64A	EPCQ128A
AFH	33h	33h	33h	33h
B0H	7Ah	7Ah	7Ah	7Ah
B1H	75h	75h	75h	75h
B2H	7Ah	7Ah	7Ah	7Ah
B3H	75h	75h	75h	75h
B4H	F7h	F7h	F7h	F7h
B5H	A2h	A2h	A2h	A2h
B6H	D5h	D5h	D5h	D5h
B7H	5Ch	5Ch	5Ch	5Ch
B8H	19h	19h	19h	19h
B9H	F7h	F7h	F7h	F7h
BAH	4Dh	4Dh	4Dh	4Dh
BBH	FFh	FFh	FFh	FFh
BCH	E9h	E9h	E9h	E9h
BDH	30h	30h	30h	30h
BEH	F8h	F8h	F8h	F8h
BFH	80h	80h	80h	80h

Related Information

Read SFDP Register Operation (5Ah) on page 29

1.14. Document Revision History for EPCQ-A Serial Configuration Device Datasheet

Document Version	Changes
2018.04.11	• Updated EPCQA instances to EPCQ-A. • Added SFDP register operation code, description, timing diagram, and register definitions. • Added a note stating that t_{WB} is the time for a complete page write in *Write Operation Parameters*. • Removed a note in *Summary of Operation Codes* table stating that Read silicon identification command is not applicable to EPCQ32A and EPCQ128A.
2018.03.13	• Removed I_{MAX} and T_{AMB} parameters from the *Absolute Maximum Ratings for EPCQ-A Devices* table.
2018.02.15	• Added I_{MAX} and T_{AMB} parameters in the *Absolute Maximum Ratings for EPCQ-A Devices* table. • Added data retention feature information.

Date	Version	Changes
December 2017	2017.12.15	• Updated maximum t_{ODIS} in the *Read Operation Parameters* table. • Removed t_{nCLK2D} from *Read Operation Parameters* table.
August 2017	2017.08.02	• Updated ordering code number. • Added link to AN822: Configuration Device Migration Guideline. • Updated Extended Quad Input Fast Read and Quad Input Fast Write operations timing diagram.
July 2017	2017.07.28	Initial release.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Intel:

EPCQ4ASI8N EPCQ16ASI8N EPCQ32ASI8N