

**EN25QH128A (2T)****128 Megabit 3V Serial Flash Memory with 4Kbyte Uniform Sector****FEATURES**

- Single power supply operation
 - Full voltage range: 2.7-3.6 volt
- Serial Interface Architecture
 - SPI Compatible: Mode 0 and Mode 3
- 128 M-bit Serial Flash
 - 128 M-bit / 16,384 KByte /65,535 pages
 - 256 bytes per programmable page
- Standard, Dual or Quad SPI
 - Standard SPI: CLK, CS#, DI, DO, WP#, HOLD#/RESET#
 - Dual SPI: CLK, CS#, DQ₀, DQ₁, WP#, HOLD#/RESET#
 - Quad SPI: CLK, CS#, DQ₀, DQ₁, DQ₂, DQ₃
 - Configurable dummy cycle number
- High performance
 - Normal read
 - 83MHz
 - Fast read (Single Data Rate Mode)
 - Standard SPI: 104MHz with 1 dummy bytes
 - Dual SPI: 104MHz with 1 dummy bytes
 - Quad SPI: 104MHz with 3 dummy bytes
- Support Serial Flash Discoverable Parameters (SFDP) signature
- Low power consumption
 - 5 mA typical active current
 - 1µA typical power down current
- Uniform Sector Architecture:
 - 4096 sectors of 4-Kbyte
 - 512 blocks of 32-Kbyte
 - 256 blocks of 64-Kbyte
 - Any sector or block can be erased individually
- Software and Hardware Write Protection:
 - Write Protect all or portion of memory via software
 - Enable/Disable protection with WP# pin
- Software and Hardware Reset
- High performance program/erase speed
 - Page program time: 0.5ms typical
 - Sector erase time: 40ms typical
 - Half Block erase time 200ms typical
 - Block erase time 300ms typical
 - Chip erase time: 60 Seconds typical
- Volatile Status Register Bits.
- Lockable 512 byte OTP security sector
- Read Unique ID Number
- Minimum 100K endurance cycle
- Data retention time 20years
- Package Options
 - 8 pins SOP 200mil body width
 - 8 contact VDFN(5x6mm)
 - 8 pins PDIP
 - 16 pins SOP 300mil body width
 - 24 balls TFBGA (6x8mm)
 - 8 contact VDFN (6x8mm)
 - All Pb-free packages are compliant RoHS, Halogen-Free and REACH.
- Industrial temperature Range

GENERAL DESCRIPTION

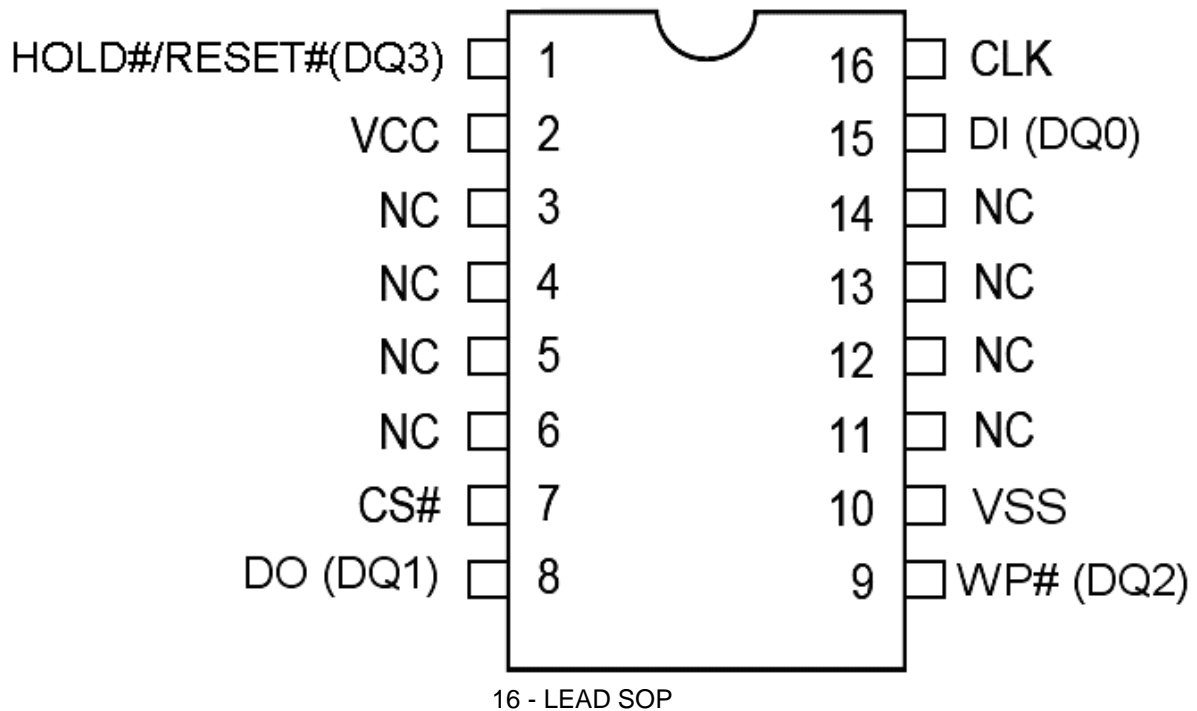
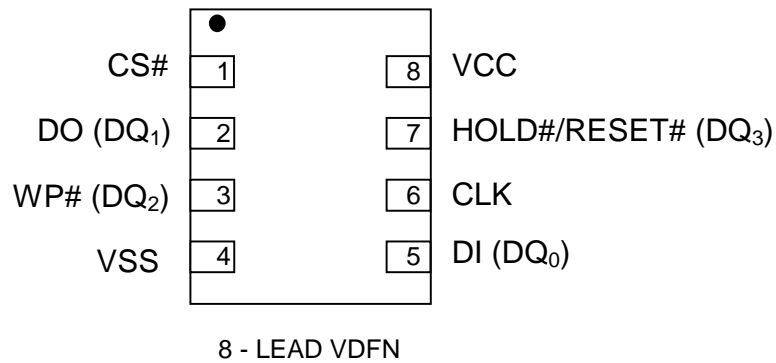
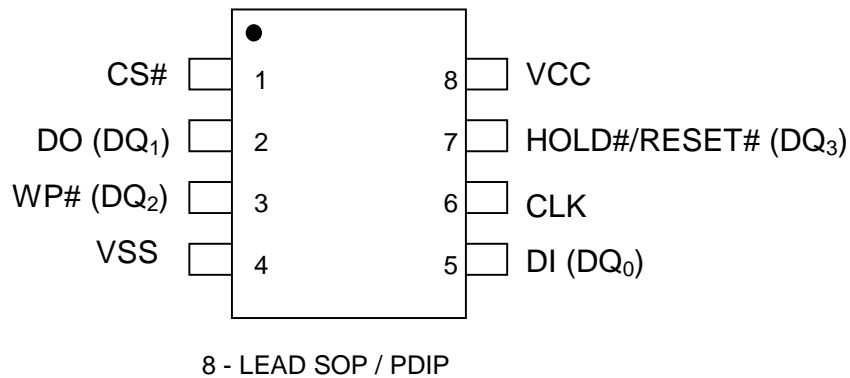
The EN25QH128A(2T) is a 128 Megabit (16,384K-byte) Serial Flash memory, with advanced write protection mechanisms. The EN25QH128A(2T) supports the single bit and four bits serial input and output commands via standard Serial Peripheral Interface (SPI) pins: Serial Clock, Chip Select, Serial DQ₀ (DI) and DQ₁(DO), DQ₂(WP#) and DQ₃(HOLD#/RESET#). SPI clock frequencies of up to 104Mhz are supported allowing equivalent clock rates of 416Mhz(104Mhz x 4) for Quad Output while using the Quad Output read instructions. The memory can be programmed 1 to 256 bytes at a time, using the Page Program instruction.

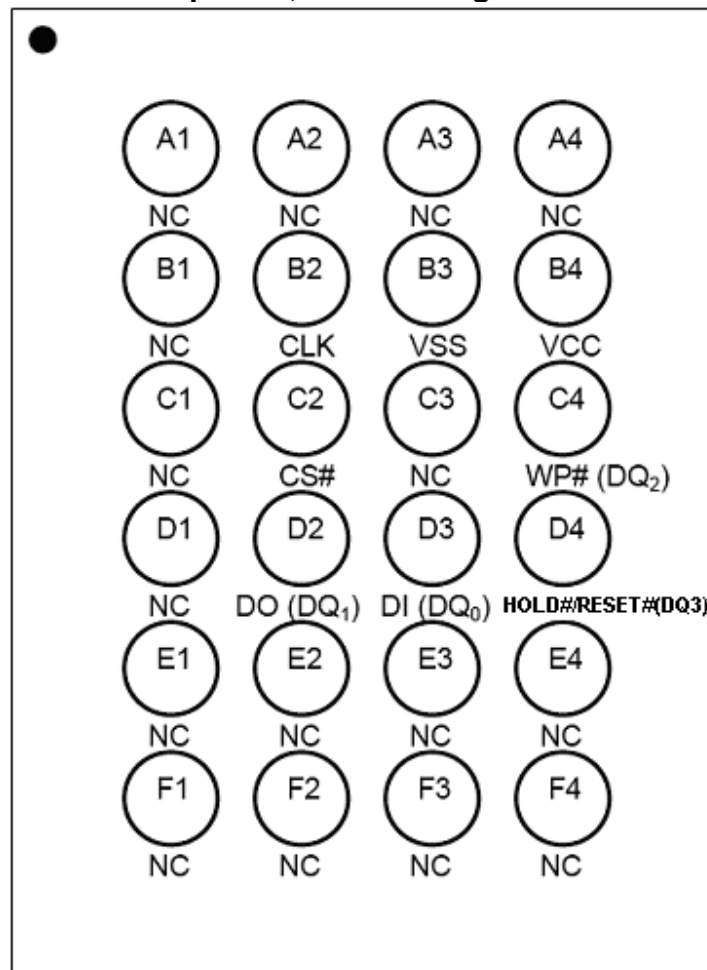
The EN25QH128A(2T) also offers a sophisticated method for protecting individual blocks against erroneous or malicious program and erase operations. By providing the ability to individually protect and unprotect blocks, a system can unprotect a specific block to modify its contents while keeping the remaining blocks of the memory array securely protected. This is useful in applications where program code is patched or updated on a subroutine or module basis or in applications where data storage segments need to be modified without running the risk of errant modifications to the program code segments.

The EN25QH128A(2T) is designed to allow either single Sector/Block at a time or full chip erase operation. The EN25QH128A(2T) can be configured to protect part of the memory as the software protected mode. The device can sustain a minimum of 100K program/erase cycles on each sector or block.



Figure.1 CONNECTION DIAGRAMS



Top View, Balls Facing Down


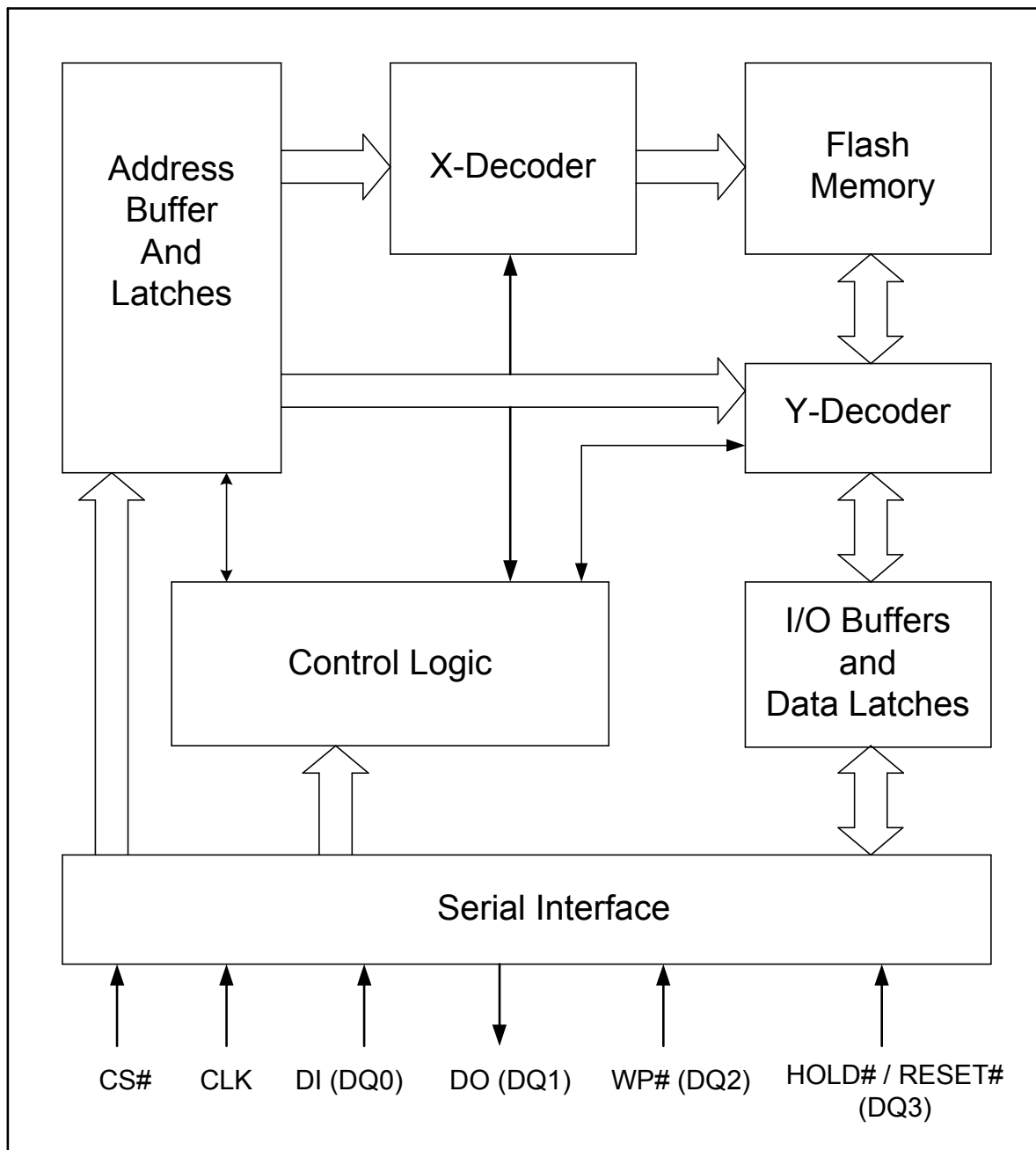
24 - Ball TFBGA

Table 1. Pin Names

Symbol	Pin Name
CLK	Serial Clock Input
DI (DQ ₀)	Serial Data Input (Data Input Output 0) ^{*1}
DO (DQ ₁)	Serial Data Output (Data Input Output 1) ^{*1}
CS#	Chip Enable
WP# (DQ ₂)	Write Protect (Data Input Output 2) ^{*2}
HOLD#/RESET# (DQ ₃)	HOLD# or RESET# pin (Data Input Output 3) ^{*2}
Vcc	Supply Voltage (2.7-3.6V)
Vss	Ground
NC	No Connect

Note:

1. DQ₀ and DQ₁ are used for Dual and Quad instructions.
2. DQ₀ ~ DQ₃ are used for Quad instructions,
 WP# & HOLD# (or RESET#) functions are only available for Standard/Dual SPI.

Figure 2. BLOCK DIAGRAM

Note:

1. DQ_0 and DQ_1 are used for Dual instructions.
2. $DQ_0 \sim DQ_3$ are used for Quad instructions.



SIGNAL DESCRIPTION**Serial Data Input, Output and IOs (DI, DO and DQ₀, DQ₁, DQ₂, DQ₃)**

The EN25QH128A(2T) support standard SPI, Dual SPI and Quad SPI operation. Standard SPI instructions use the unidirectional DI (input) pin to serially write instructions, addresses or data to the device on the rising edge of the Serial Clock (CLK) input pin. Standard SPI also uses the unidirectional DO (output) to read data or status from the device on the falling edge CLK.

Dual and Quad SPI instruction use the bidirectional IO pins to serially write instruction, addresses or data to the device on the rising edge of CLK and read data or status from the device on the falling edge of CLK.

Serial Clock (CLK)

The SPI Serial Clock Input (CLK) pin provides the timing for serial input and output operations. ("See SPI Mode")

Chip Select (CS#)

The SPI Chip Select (CS#) pin enables and disables device operation. When CS# is high the device is deselected and the Serial Data Output (DO, or DQ₀, DQ₁, DQ₂ and DQ₃) pins are at high impedance. When deselected, the devices power consumption will be at standby levels unless an internal erase, program or status register cycle is in progress. When CS# is brought low the device will be selected, power consumption will increase to active levels and instructions can be written to and data read from the device. After power-up, CS# must transition from high to low before a new instruction will be accepted.

Write Protect (WP#)

The Write Protect (WP#) pin can be used to prevent the Status Register from being written. Used in conjunction with the Status Register's Block Protect (SR.5, SR.4, SR.3, SR.2) bits and Status Register Protect (SRP) bits, a portion or the entire memory array can be hardware protected. The WP# function is only available for standard SPI and Dual SPI operation, when during Quad SPI, this pin is the Serial Data IO (DQ₂) for Quad I/O operation.

HOLD (HOLD#)

The HOLD# pin allows the device to be paused while it is actively selected. When WXDIS bit is "0" (factory default) and HRSW bit is '0' (factory default is '0'), the HOLD# pin is enabled. When HOLD# is brought low, while CS# is low, the DO pin will be at high impedance and signals on the DI and CLK pins will be ignored (don't care). The hold function can be useful when multiple devices are sharing the same SPI signals. The HOLD# function is only available for standard SPI and Dual SPI operation, when during Quad SPI, this pin is the Serial Data IO (DQ₃) for Quad I/O operation.

RESET (RESET#)

The RESET# pin allows the device to be reset by the controller. When WXDIS bit is "0" (factory default) and HRSW bit is '1' (factory default is '0'), the RESET# pin is enabled. The Hardware Reset function is only available for standard SPI and Dual SPI operation, when during Quad SPI, this pin is the Serial Data IO (DQ₃) for Quad I/O operation. Set RESET# to low for a minimum period 1us (tHRST) will interrupt any on-going instructions to have the device to initial state. The device can accept new instructions again in 28us (tHRSL) after RESET# back to high.



MEMORY ORGANIZATION

The memory is organized as:

- 16,777,216 bytes
- Uniform Sector Architecture
 - 256 blocks of 64-Kbyte
 - 512 blocks of 32-Kbyte
 - 4,096 sectors of 4-Kbyte
 - 65,536 pages (256 bytes each)

Each page can be individually programmed (bits are programmed from 1 to 0). The device is Sector, Block or Chip Erasable but not Page Erasable.



Table 2. Uniform Block Sector Architecture (1/4)

64K Block	32K Block	Sector	Address range	
255	511	4095	FFF000h	FFFFFFh
	510	⋮	⋮	⋮
254		509	4080	FF0000h
	508	4079	FEF000h	FEFFFFh
253		507	4064	FE0000h
	506	4063	FDF000h	FDFFFFh
252		505	4048	FD0000h
	251	504	3887	F2F000h
250		485	3872	F20000h
	249	484	3871	F1F000h
248		483	3856	F10000h
	247	482	3855	F0F000h
246		481	3840	F00000h
	245	480	3824	EF0000h
244		479	3823	EEF000h
	243	478	3808	EE0000h
242		477	3807	EDF000h
	241	476	3792	ED0000h
240		475	3631	E2F000h
	239	474	3616	E20000h
238		473	3615	E1F000h
	237	472	3600	E10000h
236		471	3599	E0F000h
	235	470	3584	E00000h

64K Block	32K Block	Sector	Address range	
234	469	3583	DF0000h	DF0FFFh
	468	3568	DF0000h	DF0FFFh
233		467	3567	DEF000h
	232	466	3552	DE0000h
231		465	3551	DDF000h
	230	464	3536	DD0000h
229		463	3375	D2F000h
	228	462	3360	D20000h
227		461	3359	D1F000h
	226	460	3344	D10000h
225		459	3343	D0F000h
	224	458	3328	D00000h
223		457	3327	CFF000h
	222	456	3312	CF0000h
221		455	3311	CEF000h
	220	454	3296	CE0000h
219		453	3295	CDF000h
	218	452	3280	CD0000h
217		451	3119	C2F000h
	216	450	3014	C20000h
215		449	3103	C1F000h
	214	448	3088	C10000h
213		447	3087	C0F000h
	212	446	3072	C00000h

64K Block	32K Block	Sector	Address range	
223	447	3583	DF0000h	DF0FFFh
	446	3568	DF0000h	DF0FFFh
222		445	3567	DEF000h
	444	3552	DE0000h	DE0FFFh
221		443	3551	DDF000h
	442	3536	DD0000h	DD0FFFh
220		441	3375	D2F000h
	440	3360	D20000h	D20FFFh
219		439	3359	D1F000h
	438	3344	D10000h	D10FFFh
218		437	3343	D0F000h
	436	3328	D00000h	D00FFFh

64K Block	32K Block	Sector	Address range	
207	415	3327	CFF000h	CFFFFFh
	414	3312	CF0000h	CF0FFFh
206		413	3311	CEF000h
	412	3296	CE0000h	CE0FFFh
205		411	3295	CDF000h
	410	3280	CD0000h	CD0FFFh
204		409	3119	C2F000h
	408	3014	C20000h	C20FFFh
203		407	3103	C1F000h
	406	3088	C10000h	C10FFFh
202		405	3087	C0F000h
	404	3072	C00000h	C00FFFh



Table 2. Uniform Block Sector Architecture (2/4)

64K Block	32K Block	Sector	Address range	
191	383	3071	BFF000h	BFFFFFFh
	382	∴	∴	∴
		382	3056	BF0000h
190	382	3055	BEF000h	BEFFFFFFh
	380	∴	∴	∴
		380	3040	BE0000h
189	379	3039	BDF000h	BDFFFFFFh
	378	∴	∴	∴
		378	3024	BD0000h
∴	∴	∴	∴	∴
178	357	2863	B2F000h	B2FFFFFFh
	356	∴	∴	∴
		356	2848	B20000h
177	355	2847	B1F000h	B1FFFFFFh
	354	∴	∴	∴
		354	2832	B10000h
176	353	2831	B0F000h	B0FFFFFFh
	352	∴	∴	∴
		352	2816	B00000h

64K Block	32K Block	Sector	Address range	
175	351	2815	AFF000h	AFFFFFFh
	350	∴	∴	∴
		350	2800	AF0000h
174	349	2799	AEF000h	AEFFFFFFh
	348	∴	∴	∴
		348	2784	AE0000h
173	347	2783	ADF000h	ADFFFFFFh
	346	∴	∴	∴
		346	2768	AD0000h
∴	∴	∴	∴	∴
162	325	2607	A2F000h	A2FFFFFFh
	324	∴	∴	∴
		324	2592	A20000h
161	323	2591	A1F000h	A1FFFFFFh
	322	∴	∴	∴
		322	2576	A10000h
160	321	2575	A0F000h	A0FFFFFFh
	320	∴	∴	∴
		320	2560	A00000h

64K Block	32K Block	Sector	Address range	
159	319	2559	9FF000h	9FFFFFFh
	318	∴	∴	∴
		318	2544	9F0000h
158	317	2543	9EF000h	9EFFFFFFh
	316	∴	∴	∴
		316	2528	9E0000h
157	315	2527	9DF000h	9DFFFFFFh
	314	∴	∴	∴
		314	2512	9D0000h
∴	∴	∴	∴	∴
146	293	2351	92F000h	92FFFFFFh
	292	∴	∴	∴
		292	2336	920000h
145	291	2335	91F000h	91FFFFFFh
	290	∴	∴	∴
		290	2320	910000h
144	289	2319	90F000h	90FFFFFFh
	288	∴	∴	∴
		288	2304	900000h

64K Block	32K Block	Sector	Address range	
143	287	2303	8FF000h	8FFFFFFh
	286	∴	∴	∴
		286	2288	8F0000h
142	285	2287	8EF000h	8EFFFFFFh
	284	∴	∴	∴
		284	2272	8E0000h
141	283	2271	8DF000h	8DFFFFFFh
	282	∴	∴	∴
		282	2256	8D0000h
∴	∴	∴	∴	∴
130	261	2095	82F000h	82FFFFFFh
	260	∴	∴	∴
		260	2080	820000h
129	259	2079	81F000h	81FFFFFFh
	258	∴	∴	∴
		258	2064	810000h
128	257	2063	80F000h	80FFFFFFh
	256	∴	∴	∴
		256	2048	800000h



Table 2. Uniform Block Sector Architecture (3/4)

64K Block	32K Block	Sector	Address range	
127	255	2047	7FF000h	7FFFFFFh
	254	∴	∴	∴
2032		7F0000h	7F0FFFh	
126	253	2031	7EF000h	7EFFFFh
	252	∴	∴	∴
2016		7E0000h	7E0FFFh	
125	251	2015	7DF000h	7DFFFFh
	250	∴	∴	∴
2000		7D0000h	7D0FFFh	
∴	∴	∴	∴	∴
114	229	1839	72F000h	72FFFFh
	228	∴	∴	∴
1824		720000h	720FFFh	
113	227	1823	71F000h	71FFFFh
	226	∴	∴	∴
1808		710000h	710FFFh	
112	225	1807	70F000h	70FFFFh
	224	∴	∴	∴
1792		700000h	700FFFh	

64K Block	32K Block	Sector	Address range	
111	223	1791	6FF000h	6FFFFFFh
	222	∴	∴	∴
1776		6F0000h	6F0FFFh	
110	221	1775	6EF000h	6EFFFFh
	220	∴	∴	∴
1760		6E0000h	6E0FFFh	
109	219	1759	6DF000h	6DFFFFh
	218	∴	∴	∴
1744		6D0000h	6D0FFFh	
∴	∴	∴	∴	∴
98	197	1583	62F000h	62FFFFh
	196	∴	∴	∴
1568		620000h	620FFFh	
97	195	1567	61F000h	61FFFFh
	194	∴	∴	∴
1552		610000h	610FFFh	
96	193	1551	60F000h	60FFFFh
	192	∴	∴	∴
1536		600000h	600FFFh	

64K Block	32K Block	Sector	Address range	
95	191	1535	5FF000h	5FFFFFFh
	190	∴	∴	∴
1520		5F0000h	5F0FFFh	
94	189	1519	5EF000h	5EFFFFh
	188	∴	∴	∴
1504		5E0000h	5E0FFFh	
93	187	1503	5DF000h	5DFFFFh
	186	∴	∴	∴
1488		5D0000h	5D0FFFh	
∴	∴	∴	∴	∴
82	165	1327	52F000h	52FFFFh
	164	∴	∴	∴
1312		520000h	520FFFh	
81	163	1311	51F000h	51FFFFh
	162	∴	∴	∴
1296		510000h	510FFFh	
80	161	1295	50F000h	50FFFFh
	160	∴	∴	∴
1280		500000h	500FFFh	

64K Block	32K Block	Sector	Address range	
79	159	1279	4FF000h	4FFFFFFh
	158	∴	∴	∴
1264		4F0000h	4F0FFFh	
78	157	1263	4EF000h	4EFFFFh
	156	∴	∴	∴
1248		4E0000h	4E0FFFh	
77	155	1247	4DF000h	4DFFFFh
	154	∴	∴	∴
1232		4D0000h	4D0FFFh	
∴	∴	∴	∴	∴
66	133	1071	42F000h	42FFFFh
	132	∴	∴	∴
1056		420000h	420FFFh	
65	131	1055	41F000h	41FFFFh
	130	∴	∴	∴
1040		410000h	410FFFh	
64	129	1039	40F000h	40FFFFh
	128	∴	∴	∴
1024		400000h	400FFFh	



Table 2. Uniform Block Sector Architecture (4/4)

64K Block	32K Block	Sector	Address range	
63	127	4095	3FF000h	3FFFFFFh
		⋮	⋮	⋮
	126	4080	3F0000h	3F0FFFh
62	125	4079	3EF000h	3EFFFFh
		⋮	⋮	⋮
	124	4064	3E0000h	3E0FFFh
61	123	4063	3DF000h	3DFFFFh
		⋮	⋮	⋮
	122	4048	3D0000h	3D0FFFh
⋮	⋮	⋮	⋮	⋮
50	101	815	32F000h	32FFFFh
		⋮	⋮	⋮
	100	800	320000h	320FFFh
49	99	799	31F000h	31FFFFh
		⋮	⋮	⋮
	98	784	310000h	310FFFh
48	97	783	30F000h	30FFFFh
		⋮	⋮	⋮
	96	768	300000h	300FFFh

64K Block	32K Block	Sector	Address range	
47	95	767	2FF000h	2FFFFFFh
		⋮	⋮	⋮
	94	752	2F0000h	2F0FFFh
46	93	751	2EF000h	2EFFFFh
		⋮	⋮	⋮
	92	736	2E0000h	2E0FFFh
45	91	735	2DF000h	2DFFFFh
		⋮	⋮	⋮
	90	720	2D0000h	2D0FFFh
⋮	⋮	⋮	⋮	⋮
34	69	559	22F000h	22FFFFh
		⋮	⋮	⋮
	68	544	220000h	220FFFh
33	67	543	21F000h	21FFFFh
		⋮	⋮	⋮
	66	528	210000h	210FFFh
32	65	527	20F000h	20FFFFh
		⋮	⋮	⋮
	64	512	200000h	200FFFh

64K Block	32K Block	Sector	Address range	
31	63	511	1FF000h	1FFFFFFh
		⋮	⋮	⋮
	62	496	1F0000h	1F0FFFh
30	61	495	1EF000h	1EFFFFh
		⋮	⋮	⋮
	60	480	1E0000h	1E0FFFh
29	59	479	1DF000h	1DFFFFh
		⋮	⋮	⋮
	58	464	1D0000h	1D0FFFh
⋮	⋮	⋮	⋮	⋮
18	37	303	12F000h	12FFFFh
		⋮	⋮	⋮
	36	288	120000h	120FFFh
17	35	287	11F000h	11FFFFh
		⋮	⋮	⋮
	34	272	110000h	110FFFh
16	33	271	10F000h	10FFFFh
		⋮	⋮	⋮
	32	256	100000h	100FFFh

64K Block	32K Block	Sector	Address range	
15	31	255	0FF000h	0FFFFFFh
		⋮	⋮	⋮
	30	240	0F0000h	0F0FFFh
14	29	239	0EF000h	0EFFFFh
		⋮	⋮	⋮
	28	224	0E0000h	0E0FFFh
13	27	223	0DF000h	0DFFFFh
		⋮	⋮	⋮
	26	208	0D0000h	0D0FFFh
⋮	⋮	⋮	⋮	⋮
2	5	47	02F000h	02FFFFh
		⋮	⋮	⋮
	4	32	020000h	020FFFh
1	3	31	01F000h	01FFFFh
		⋮	⋮	⋮
	2	16	010000h	010FFFh
0	1	15	00F000h	00FFFFh
		⋮	⋮	⋮
	0	0	000000h	000FFFh

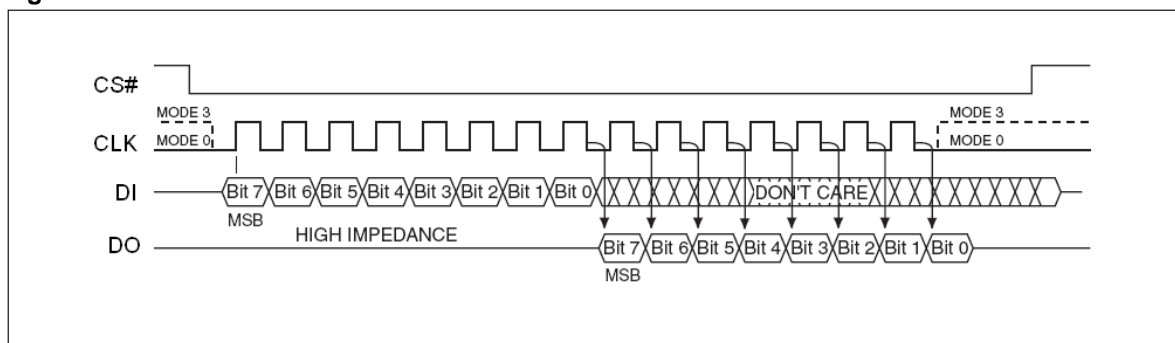


OPERATING FEATURES

Standard SPI Modes

The EN25QH128A(2T) is accessed through an SPI compatible bus consisting of four signals: Serial Clock (CLK), Chip Select (CS#), Serial Data Input (DI) and Serial Data Output (DO). Both SPI bus operation Modes 0 (0,0) and 3 (1,1) are supported. The primary difference between Mode 0 and Mode 3, as shown in Figure 3, concerns the normal state of the CLK signal when the SPI bus master is in standby and data is not being transferred to the Serial Flash. For Mode 0 the CLK signal is normally low. For Mode 3 the CLK signal is normally high. In either case data input on the DI pin is sampled on the rising edge of the CLK. Data output on the DO pin is clocked out on the falling edge of CLK.

Figure 3. SPI Modes

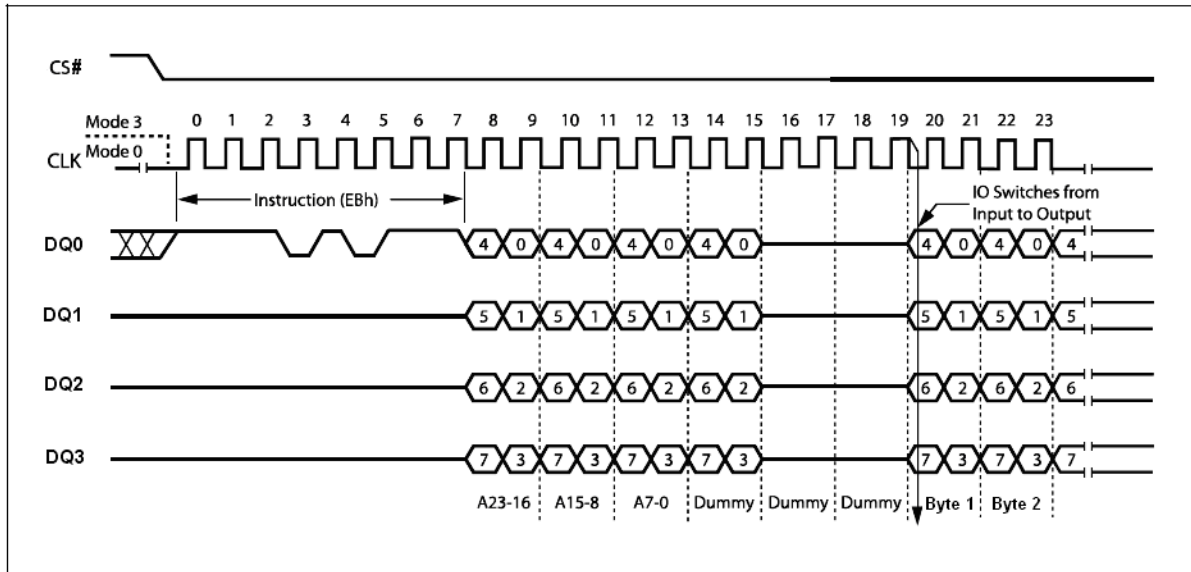


Dual SPI Instruction

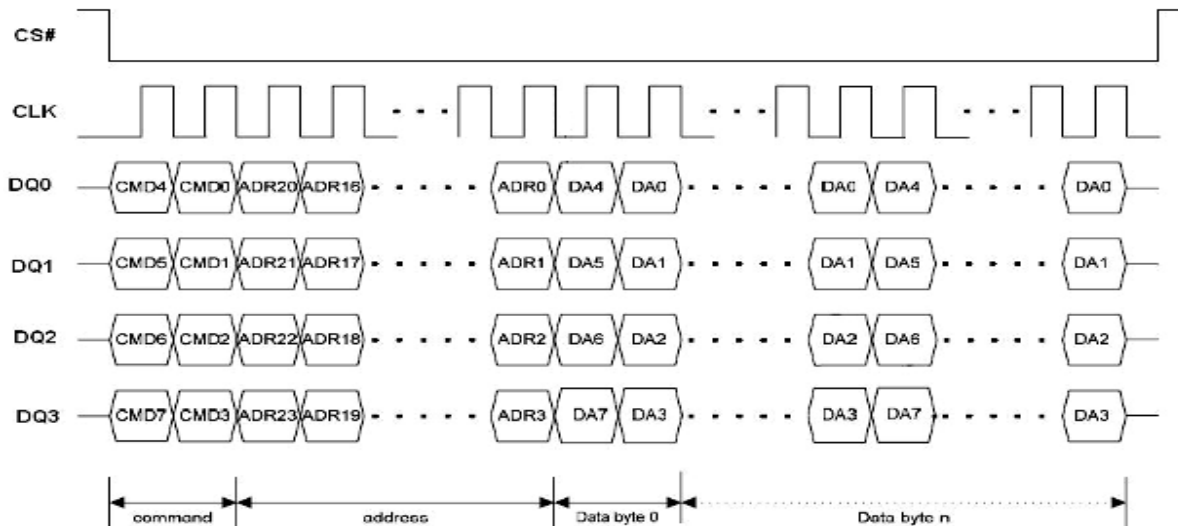
The EN25QH128A(2T) supports Dual SPI operation when using the “Dual Output Fast Read and Dual I/O FAST_READ “ (3Bh and BBh) instructions. These instructions allow data to be transferred to or from the Serial Flash memory at two to three times the rate possible with the standard SPI. The Dual Read instructions are ideal for quickly downloading code from Flash to RAM upon power-up (code-shadowing) or for application that cache code-segments to RAM for execution. The Dual output feature simply allows the SPI input pin to also serve as an output during this instruction. When using Dual SPI instructions the DI and DO pins become bidirectional I/O pins; DQ₀ and DQ₁. All other operations use the standard SPI interface with single output signal.

Quad I/O SPI Modes

The EN25QH128A(2T) supports Quad input/output operation when using the Quad I/O Fast Read (EBh). This instruction allows data to be transferred to or from the Serial Flash memory at four to six times the rate possible with the standard SPI. The Quad Read instruction offer a significant improvement in continuous and random access transfer rates allowing fast code-shadowing to RAM or for application that cache code-segments to RAM for execution. When using Quad SPI instruction the DI and DO pins become bidirectional I/O pins; DQ₀ and DQ₁, and the WP# and HOLD#/RESET# pins become DQ₂ and DQ₃ respectively.

Figure 4. Quad SPI Modes

Full Quad SPI Modes (QPI)

The EN25QH128A(2T) also supports Full Quad SPI Mode (QPI) function while using the Enable Quad Peripheral Interface mode (EQPI) (38h). When using Quad SPI instruction the DI and DO pins become bidirectional I/O pins; DQ₀ and DQ₁, and the WP# and HOLD#/RESET# pins become DQ₂ and DQ₃ respectively.

Figure 5. Full Quad SPI Modes


**Page Programming**

To program one data byte, two instructions are required: Write Enable (WREN), which is one byte, and a Page Program (PP) or Quad Input Page Program (QPP) sequence, which consists of four bytes plus data. This is followed by the internal Program cycle (of duration t_{PP}).

To spread this overhead, the Page Program (PP) or Quad Input Page Program (QPP) instruction allows up to 256 bytes to be programmed at a time (changing bits from 1 to 0) provided that they lie in consecutive addresses on the same page of memory.

Sector Erase, Half Block Erase, Block Erase and Chip Erase

The Page Program (PP) or Quad Input Page Program (QPP) instruction allows bits to be reset from 1 to 0. Before this can be applied, the bytes of memory need to have been erased to all 1s (FFh). This can be achieved a sector at a time, using the Sector Erase (SE) instruction, half a block at a time using the Half Block Erase (HBE) instruction, a block at a time using the Block Erase (BE) instruction or throughout the entire memory, using the Chip Erase (CE) instruction. This starts an internal Erase cycle (of duration t_{SE} , t_{HBE} , t_{BE} or t_{CE}). The Erase instruction must be preceded by a Write Enable (WREN) instruction.

Polling During a Write, Program or Erase Cycle

A further improvement in the time to Write Status Register (WRSR), Program (PP, QPP) or Erase (SE, HBE, BE or CE) can be achieved by not waiting for the worst case delay (t_W , t_{PP} , t_{SE} , t_{HBE} , t_{BE} or t_{CE}). The Write In Progress (WIP) bit is provided in the Status Register so that the application program can monitor its value, polling it to establish when the previous Write cycle, Program cycle or Erase cycle is complete.

Active Power, Stand-by Power and Deep Power-Down Modes

When Chip Select (CS#) is Low, the device is enabled, and in the Active Power mode. When Chip Select (CS#) is High, the device is disabled, but could remain in the Active Power mode until all internal cycles have completed (Program, Erase, Write Status Register). The device then goes into the Stand-by Power mode. The device consumption drops to I_{CC1} .

The Deep Power-down mode is entered when the specific instruction (the Enter Deep Power-down Mode (DP) instruction) is executed. The device consumption drops further to I_{CC2} . The device remains in this mode until another specific instruction (the Release from Deep Power-down Mode and Read Device ID (RDI) instruction) is executed.

All other instructions are ignored while the device is in the Deep Power-down mode. This can be used as an extra software protection mechanism, when the device is not in active use, to protect the device from inadvertent Write, Program or Erase instructions.

Write Protection

Applications that use non-volatile memory must take into consideration the possibility of noise and other adverse system conditions that may compromise data integrity. To address this concern the EN25QH128A(2T) provides the following data protection mechanisms:

- Power-On Reset and an internal timer (t_{PUW}) can provide protection against inadvertent changes while the power supply is outside the operating specification.
- Program, Erase and Write Status Register instructions are checked that they consist of a number of clock pulses that is a multiple of eight, before they are accepted for execution.
- All instructions that modify data must be preceded by a Write Enable (WREN) instruction to set the Write Enable Latch (WEL) bit. This bit is returned to its reset state by the following events:
 - Power-up
 - Write Disable (WRDI) instruction completion or Write Status Register (WRSR) instruction completion or Page Program (PP), Quad Input Page Program (QPP) instruction completion or Sector Erase (SE) instruction completion or Half Block Erase (HBE) / Block Erase (BE) instruction completion or Chip Erase (CE) instruction completion
 - Software/Hardware Reset completion
- The Block Protect (BP3, BP2, BP1, BP0) bits allow part of the memory to be configured as read-only. This is the Software Protected Mode (SPM).
- The Write Protect (WP#) signal allows the Block Protect (BP3, BP2, BP1, BP0) bits and Status Register Protect (SRP) bit to be protected. This is the Hardware Protected Mode (HPM).
- In addition to the low power consumption feature, the Deep Power-down mode offers extra software protection from inadvertent Write, Program and Erase instructions, as all instructions are ignored except one particular instruction (the Release from Deep Power-down instruction).



Table 3. Protected Area Sizes Sector Organization

Status Register Content					Memory Content			
T/B Bit	SR.5 Bit	SR.4 Bit	SR.3 Bit	SR.2 Bit	Protect Areas	Addresses	Density(KB)	Portion
0	0	0	0	0	None	None	None	None
0	0	0	0	1	Block 252 to 255	FC0000h-FFFFFFh	256KB	Upper 4/256
0	0	0	1	0	Block 248 to 255	F80000h-FFFFFFh	512KB	Upper 8/256
0	0	0	1	1	Block 240 to 255	F00000h-FFFFFFh	1024KB	Upper 16/256
0	0	1	0	0	Block 224 to 255	E00000h-FFFFFFh	2048KB	Upper 32/256
0	0	1	0	1	Block 192 to 255	C00000h-FFFFFFh	4096KB	Upper 64/256
0	0	1	1	0	Block 128 to 255	800000h-FFFFFFh	8192KB	Upper 128/256
0	0	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
0	1	0	0	0	None	None	None	None
0	1	0	0	1	Block 0 to 3	000000h-03FFFFh	256KB	Lower 4/256
0	1	0	1	0	Block 0 to 7	000000h-07FFFFh	512KB	Lower 8/256
0	1	0	1	1	Block 0 to 15	000000h-0FFFFFFh	1024KB	Lower 16/256
0	1	1	0	0	Block 0 to 31	000000h-1FFFFFFh	2048KB	Lower 32/256
0	1	1	0	1	Block 0 to 63	000000h-3FFFFFFh	4096KB	Lower 64/256
0	1	1	1	0	Block 0 to 127	000000h-7FFFFFFh	8192KB	Lower 128/256
0	1	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	0	0	0	0	None	None	None	None
1	0	0	0	1	Block 0 to 251	000000h-FBFFFFh	16128KB	Lower 252/256
1	0	0	1	0	Block 0 to 247	000000h-F7FFFFh	15872KB	Lower 248/256
1	0	0	1	1	Block 0 to 239	000000h-EFFFFFFh	15360KB	Lower 240/256
1	0	1	0	0	Block 0 to 223	000000h-DFFFFFFh	14336KB	Lower 224/256
1	0	1	0	1	Block 0 to 191	000000h-BFFFFFFh	12288KB	Lower 192/256
1	0	1	1	0	Block 0 to 127	000000h-7FFFFFFh	8192KB	Lower 128/256
1	0	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	1	0	0	0	None	None	None	None
1	1	0	0	1	Block 4 to 255	040000h-FFFFFFh	16128KB	Upper 252/256
1	1	0	1	0	Block 8 to 255	080000h-FFFFFFh	15872KB	Upper 248/256
1	1	0	1	1	Block 16 to 255	100000h-FFFFFFh	15360KB	Upper 240/256
1	1	1	0	0	Block 32 to 255	200000h-FFFFFFh	14336KB	Upper 224/256
1	1	1	0	1	Block 64 to 255	400000h-FFFFFFh	12288KB	Upper 192/256
1	1	1	1	0	Block 128 to 255	800000h-FFFFFFh	8192KB	Upper 128/256
1	1	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All

**Enable Boot Lock**

The Enable Boot Lock feature enables user to lock the 64KB-block/sector on the top/bottom of the device for protection. This feature is activated by configuring 4KBL/TB bits and programming EBL bit to '1'. The TB bit and 4KBL bits can only be programmed once.

The bits' definitions are described in the following table.

Table 4. The Enable Boot Lock feature

Type	Register	Description	Function
Non-volatile/ Volatile bit	SR.6	Enable Boot lock	0 (default)
			1 : Lock selected 64KB-Block/Sector
OTP/Volatile bit	SR.3	Top/Bottom Protect	0 : Top (default)
			1 : Bottom
	SR.4	4KB Boot Lock	0 : 64KB-Block (default)
			1 : Sector

**INSTRUCTIONS**

All instructions, addresses and data are shifted in and out of the device, most significant bit first. Serial Data Input (DI) is sampled on the first rising edge of Serial Clock (CLK) after Chip Select (CS#) is driven Low. Then, the one-byte instruction code must be shifted in to the device, most significant bit first, on Serial Data Input (DI), each bit being latched on the rising edges of Serial Clock (CLK).

The instruction set is listed in Table 5. Every instruction sequence starts with a one-byte instruction code. Depending on the instruction, it might be followed by address bytes, or data bytes, or both or none. Chip Select (CS#) must be driven High after the last bit of the instruction sequence has been shifted in. In the case of a Read Data Bytes (READ), Read Data Bytes at Higher Speed (Fast_Read), Dual Output Fast Read (3Bh), Dual I/O Fast Read (BBh), Quad Output Fast Read (6Bh), Quad Input/Output FAST_READ (EBh), Read Status Register (RDSR), Release from Deep Power-down, and Read Device ID (RDI) instruction, the shifted-in instruction sequence is followed by a data-out sequence. Chip Select (CS#) can be driven High after any bit of the data-out sequence is being shifted out.

In the case of a write instruction, Chip Select (CS#) must be driven High exactly at a byte boundary, otherwise the instruction is rejected, and is not executed. That is, Chip Select (CS#) must be driven High when the number of clock pulses after Chip Select (CS#) being driven Low is an exact multiple of eight. For Page Program, if at any time the input byte is not a full byte, nothing will happen and WEL will not be reset.

In the case of multi-byte commands of Page Program (PP), Quad Input Page Program (QPP), and Release from Deep Power Down (RES) minimum number of bytes specified has to be given, without which, the command will be ignored.

In the case of Page Program, if the number of byte after the command is less than 4 (at least 1 data byte), it will be ignored too. In the case of SE and HBE / BE, exact 24-bit address is a must, any less or more will cause the command to be ignored.

All attempts to access the memory array during a Write Status Register cycle, Program cycle or Erase cycle are ignored, and the internal Write Status Register cycle, Program cycle or Erase cycle continues unaffected.



Table 5A. Instruction Set

Instruction Name	Byte 1 Code	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	n-Bytes
RSTEN	66h						
RST ⁽¹⁾	99h						
EQPI	38h						
RSTQPI ⁽²⁾	FFh						
Write Enable (WERN)	06h						
Volatile Status Register Write Enable ⁽³⁾	50h						
Write Disable (WRDI)/ Exit OTP mode	04h						
Read Status Register (RDSR)	05h	(S7-S0) ⁽⁴⁾					continuous ⁽⁵⁾
Write Status Register (WRSR)	01h	S7-S0					
Read Status Register 3 (RDSR3)	95h	(S7-S0) ⁽⁴⁾					
Write Status Register 3 (WRSR3)	C0h	S7-S0					
Deep Power-down	B9h						
Release from Deep Power-down, and read Device ID (RES)	ABh	dummy	dummy	dummy	(ID7-ID0)		(6)
Release from Deep Power-down (RDP)							
Manufacturer/ Device ID	90h	dummy	dummy	00h	(M7-M0)	(ID7-ID0)	(7)
				01h	(ID7-ID0)	(M7-M0)	
Read Identification (RDID)	9Fh	(M7-M0)	(ID15-ID8)	(ID7-ID0)	(8)		
Enter OTP mode	3Ah						
Read SFDP mode and Unique ID Number	5Ah	A23-A16	A15-A8	A7-A0	dummy	(D7-D0)	(Next Byte) continuous

Notes:

1. RST command only executed if RSTEN command is executed first. Any intervening command will disable Reset.
2. Release Full Quad SPI or Fast Read Enhanced mode. Device accepts eight-clocks command in Standard SPI mode, or two-clocks command in Full Quad SPI mode.
3. Volatile Status Register Write Enable command must precede WRSR command without any intervening commands to write data to Volatile Status Register.
4. Data bytes are shifted with Most Significant Bit first. Byte fields with data in parenthesis "()" indicate data being read from the device on the DO pin.
5. The Status Register contents will repeat continuously until CS# terminate the instruction.
6. The Device ID will repeat continuously until CS# terminates the instruction.
7. The Manufacturer ID and Device ID bytes will repeat continuously until CS# terminates the instruction.
00h on Byte 4 starts with MID and alternate with DID, 01h on Byte 4 starts with DID and alternate with MID.
8. (M7-M0) : Manufacturer, (ID15-ID8) : Memory Type, (ID7-ID0) : Memory Capacity.



Table 5B. Instruction Set (Read Instruction)

Instruction Name	OP Code	Address bits	Dummy bits / Clocks (Default)	Data Out	Remark
Read Data	03h	24 bits	0	(D7-D0, ...)	(Next Byte) continuous
Fast Read	0Bh	24 bits	8 bits / 8 clocks	(D7-D0, ...)	(Next Byte) continuous
Dual Output Fast Read	3Bh	24 bits	8 bits / 8 clocks	(D7-D0, ...)	(one byte Per 4 clocks, continuous)
Dual I/O Fast Read	BBh	24 bits	8 bits / 4 clocks	(D7-D0, ...)	(one byte Per 4 clocks, continuous)
Quad I/O Fast Read	EBh	24 bits	24 bits / 6 clocks	(D7-D0, ...)	(one byte per 2 clocks, continuous)
Quad Output Fast Read	6Bh	24 bits	8 bits / 8 clocks	(D7-D0, ...)	(one byte per 2 clocks, continuous)

Table 5C. Instruction Set (Program Instruction)

Instruction Name	OP Code	Address bits	Dummy bits Clocks (Default)	Data In	Remark
Page Program (PP)	02h	24 bits	0	(D7-D0, ...)	(Next Byte) continuous
Quad Input Page Program (QPP)	32h	24 bits	0	(D7-D0, ...)	(one byte per 2 clocks, continuous)

Table 5D. Instruction Set (Erase Instruction)

Instruction Name	OP Code	Address bits	Dummy bits Clocks (Default)	Data In	Remark
Sector Erase (SE)	20h	24 bits	0	(D7-D0, ...)	
32K Half Block Erase (HBE)	52h	24 bits	0	(D7-D0, ...)	
64K Block Erase (BE)	D8h	24 bits	0	(D7-D0, ...)	
Chip Erase (CE)	C7h/ 60h	24 bits	0	(D7-D0, ...)	

Table 5E. Instruction Set (Read Instruction support mode and apply dummy cycle setting)

Instruction Name	OP Code	Start From SPI/QPI ⁽¹⁾		Dummy Byte ⁽²⁾	
		SPI	QPI	Start From SPI	Start From QPI
Read Data	03h	Yes	No	N/A	N/A
Fast Read	0Bh	Yes	Yes	8 clocks	By SR3.4~5
Dual Output Fast Read	3Bh	Yes	No	8 clocks	N/A
Dual I/O Fast Read	BBh	Yes	No	4 clocks	N/A
Quad Output Fast Read	6Bh	Yes	No	8 clocks	N/A
Quad I/O Fast Read	EBh	Yes	Yes	By SR3.4~5	By SR3.4~5

Note:

- 'Start From SPI/QPI' means if this command is initiated from SPI or QPI mode.
- The dummy byte settings please refer to table 9



Table 6. Manufacturer and Device Identification

OP Code	(M7-M0)	(ID15-ID0)	(ID7-ID0)
ABh			17h
90h	1Ch		17h
9Fh	1Ch	7018h	

Reset-Enable (RSTEN) (66h) and Reset (RST) (99h)

The Reset operation is used as a system (software) reset that puts the device in normal operating Ready mode. This operation consists of two commands: Reset-Enable (RSTEN) and Reset (RST).

To reset the EN25QH128A(2T) the host drives CS# low, sends the Reset-Enable command (66h), and drives CS# high. Next, the host drives CS# low again, sends the Reset command (99h), and drives CS# high.

The Reset operation requires the Reset-Enable command followed by the Reset command. Any command other than the Reset command after the Reset-Enable command will disable the Reset-Enable.

A successful command execution will reset the status registers, see Figure 6 for SPI Mode and Figure 6.1 for Quad Mode. A device reset during an active Program or Erase operation aborts the operation, which can cause the data of the targeted address range to be corrupted or lost. Depending on the prior operation, the reset timing may vary. Recovery from a Write operation requires more software latency time (t_{SR}) than recovery from other operations.

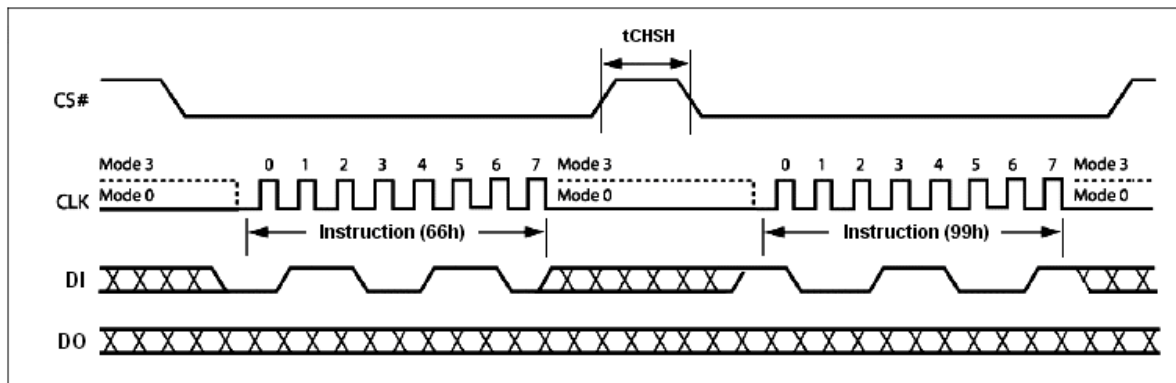


Figure 6. Reset-Enable and Reset Sequence Diagram

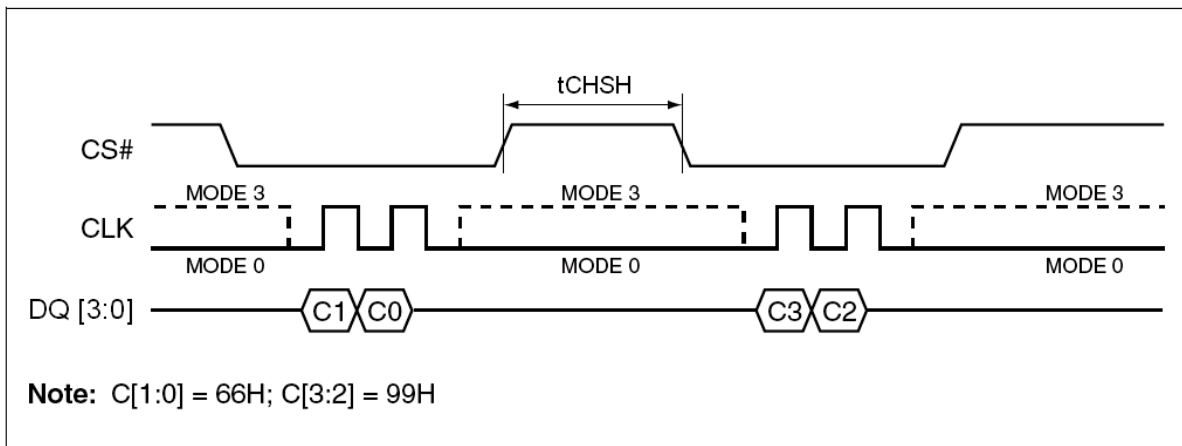
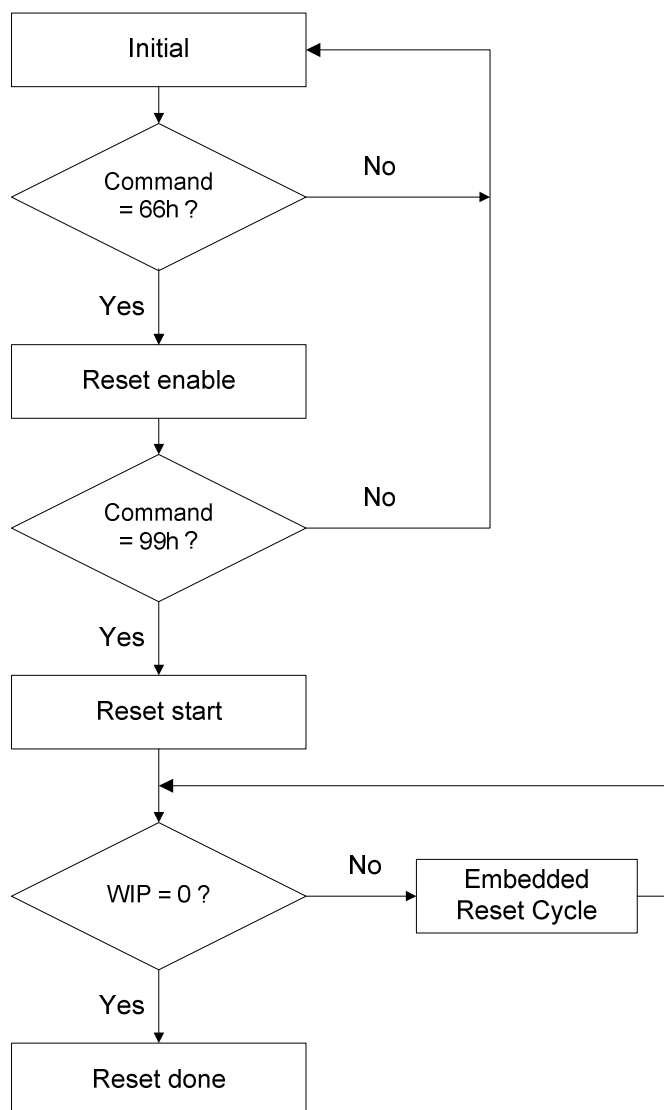


Figure 6.1 . Reset-Enable and Reset Sequence Diagram in QPI Mode

Software Reset Flow

Note:

1. Reset-Enable (RSTEN) (66h) and Reset (RST) (99h) commands need to match standard SPI or EQPI (quad) mode.
2. Continue (Enhance) EB mode need to use quad Reset-Enable (RSTEN) (66h) and quad Reset (RST) (99h) commands.
3. If user is not sure it is in SPI or Quad mode, we suggest to execute sequence as follows:
Quad Reset-Enable (RSTEN) (66h) -> Quad Reset (RST) (99h) -> SPI Reset-Enable (RSTEN) (66h) -> SPI Reset (RST) (99h) to reset.
4. The reset command could be executed during embedded program and erase process, QPI mode, Continue EB mode to back to SPI mode.
5. This flow can release the device from Deep power down mode.
6. The Status Register Bit will reset to default value after reset done.
7. If user reset device during erase, the embedded reset cycle software reset latency will take about 28us in worst case.

Enable Quad Peripheral Interface mode (EQPI) (38h)

The Enable Quad Peripheral Interface mode (EQPI) instruction will enable the flash device for Quad SPI bus operation. Upon completion of the instruction, all instructions thereafter will be 4-bit multiplexed input/output until a power cycle or “Reset Quad I/O instruction” instruction, as shown in Figure 7. The device did not support the Read Data Bytes (READ) (03h), Dual Output Fast Read (3Bh), Dual Input/Output FAST_READ (BBh) and Quad output fast read (6Bh) modes while the Enable Quad Peripheral Interface mode (EQPI) (38h) turns on.

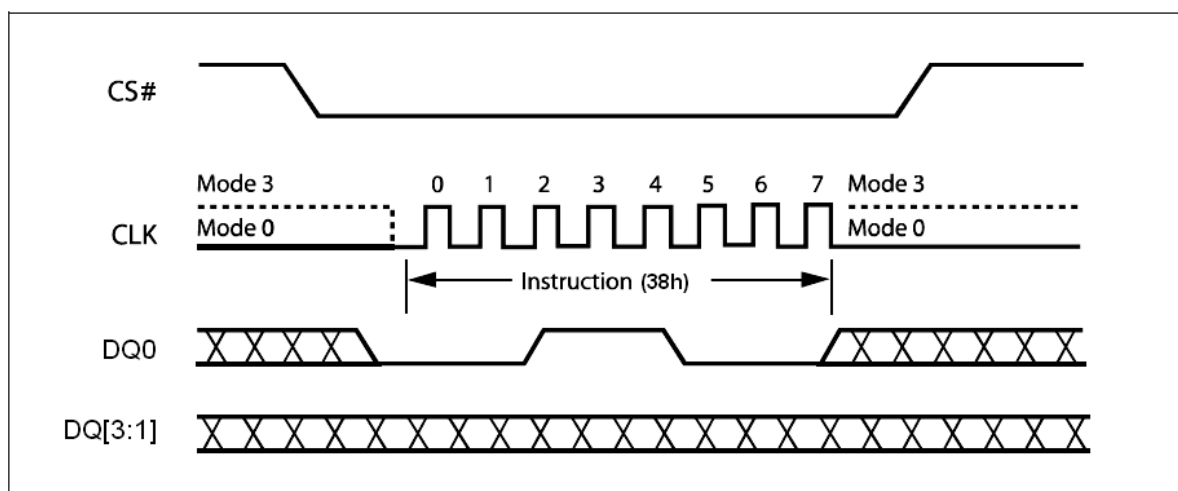


Figure 7. Enable Quad Peripheral Interface mode Sequence Diagram

Reset Quad I/O (RSTQIO) (FFh)

The Reset Quad I/O instruction resets the device to 1-bit Standard SPI operation. To execute a Reset Quad I/O operation, the host drives CS# low, sends the Reset Quad I/O command cycle (FFh) then, drives CS# high. This command can't be used in Standard SPI mode.

User also can use the FFh command to release the Quad I/O Fast Read Enhancement Mode. The detail description, please see the Quad I/O Fast Read Enhancement Mode section.

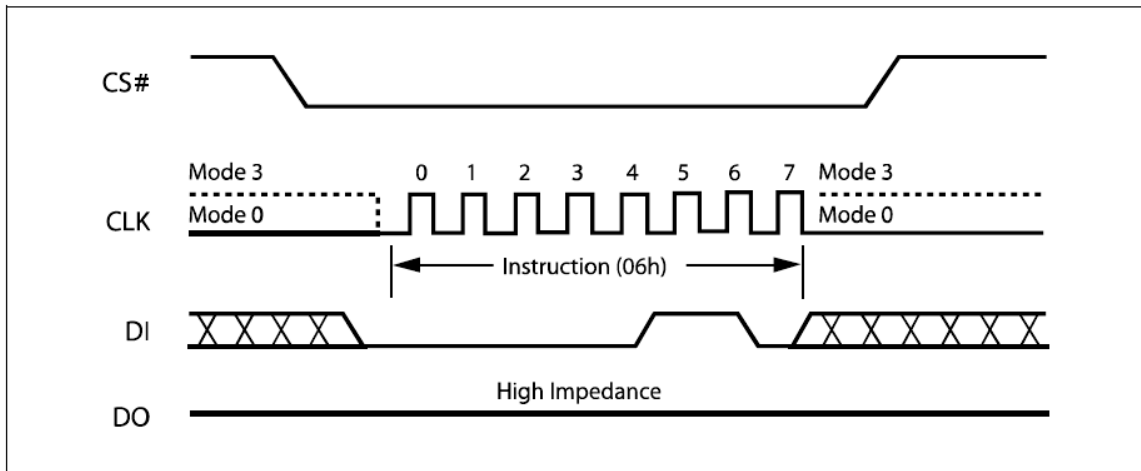
Note:

If the system is in the Quad I/O Fast Read Enhance Mode in QPI Mode, it is necessary to execute FFh command by two times. The first FFh command is to release Quad I/O Fast Read Enhance Mode, and the second FFh command is to release EQPI Mode.

Write Enable (WREN) (06h)

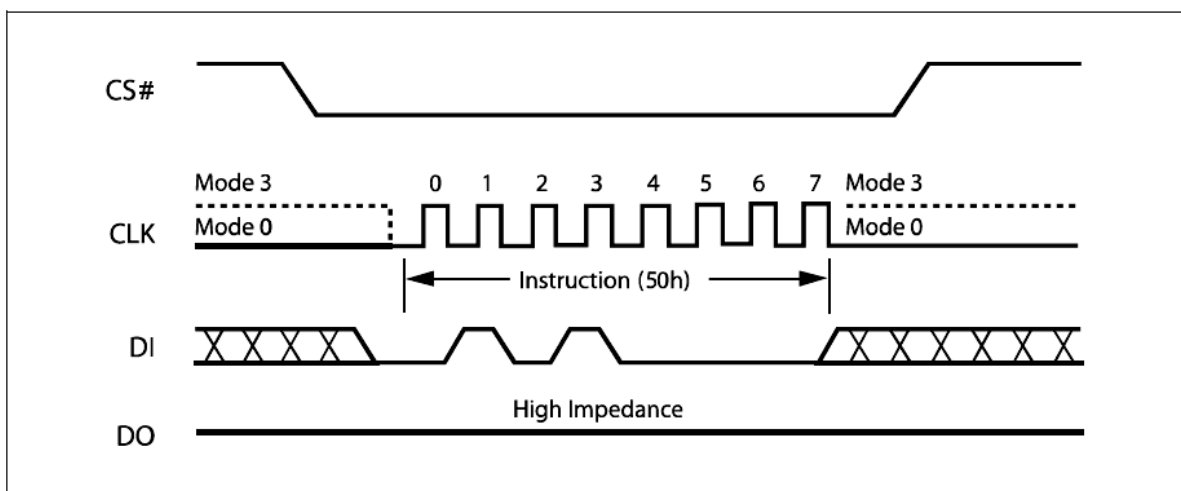
The Write Enable (WREN) instruction (Figure 8) sets the Write Enable Latch (WEL) bit. The Write Enable Latch (WEL) bit must be set prior to every Page Program (PP), Quad Input Page Program (QPP), Sector Erase (SE), Block Erase (BE), Chip Erase (CE) and Write Status Register (WRSR) instruction. The Write Enable (WREN) instruction is entered by driving Chip Select (CS#) Low, sending the instruction code, and then driving Chip Select (CS#) High.

The instruction sequence is shown in Figure 10.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 8. Write Enable Instruction Sequence Diagram
Volatile Status Register Write Enable (50h)

This feature enable user to change memory protection schemes quickly without waiting for the typical non-volatile bit write cycles or affecting the endurance of the Status Register non-volatile bits. The Volatile Status Register Write Enable (50h) command won't set the Write Enable Latch (WEL) bit, it is only valid for 'Write Status Register' (01h) command to change the Volatile Status Register bit values. To write to Volatile Status Register, issue the Volatile Status Register Write Enable (50h) command prior issuing WRSR (01h). The Status Register bits will be refresh to Volatile Status Register (SR[7:2]) within t_{SHSL2} (50ns). Upon power off or the execution of a Software/Hardware Reset, the volatile Status Register bit values will be lost, and the non-volatile Status Register bit values will be restored. The instruction sequence is shown in Figure 9.

The instruction sequence is shown in Figure 10.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 9. Volatile Status Register Write Enable Instruction Sequence Diagram



Write Disable (WRDI) (04h)

The Write Disable instruction (Figure 10) resets the Write Enable Latch (WEL) bit in the Status Register to a 0 or exit from OTP mode to normal mode. The Write Disable instruction is entered by driving Chip Select (CS#) low, shifting the instruction code “04h” into the DI pin and then driving Chip Select (CS#) high. Note that the WEL bit is automatically reset after Power-up and upon completion of the Write Status Register, Page Program, Sector Erase, Half Block Erase (HBE), Block Erase (BE) and Chip Erase instructions.

The instruction sequence is shown in Figure 10.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

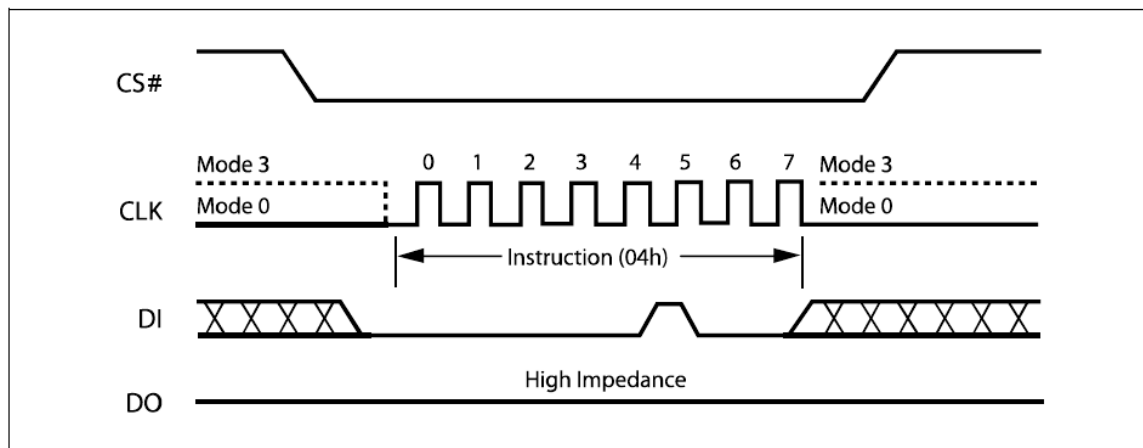


Figure 10. Write Disable Instruction Sequence Diagram

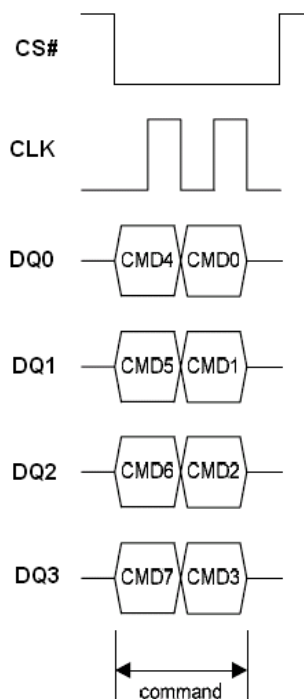


Figure 10.1 Write Enable/Disable Instruction Sequence in QPI Mode



Read Status Register (RDSR) (05h)

The Read Status Register (RDSR) instruction allows the Status Register to be read. The Status Register may be read at any time, even while a Program, Erase or Write Status Register cycle is in progress. When one of these cycles is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Status Register continuously, as shown in Figure 11.

The instruction sequence is shown in Figure 11.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

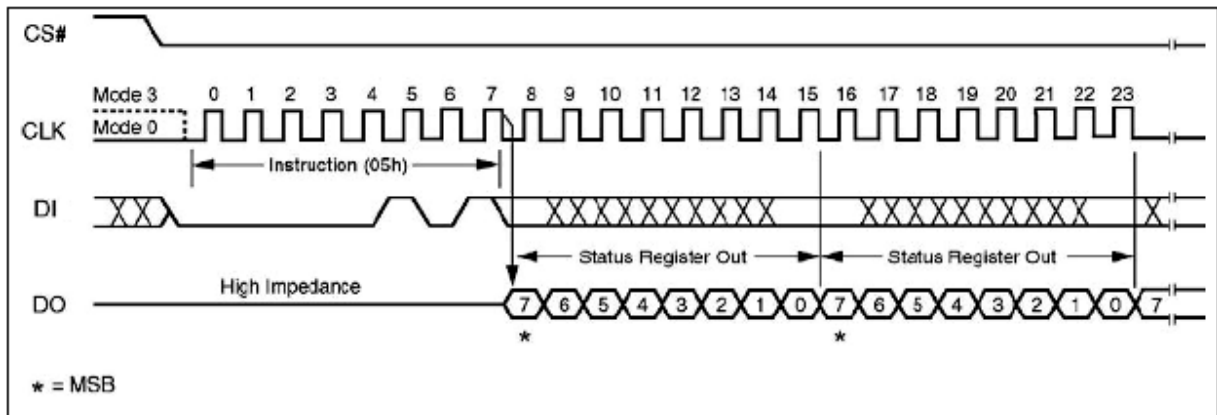


Figure 11. Read Status Register Instruction Sequence Diagram

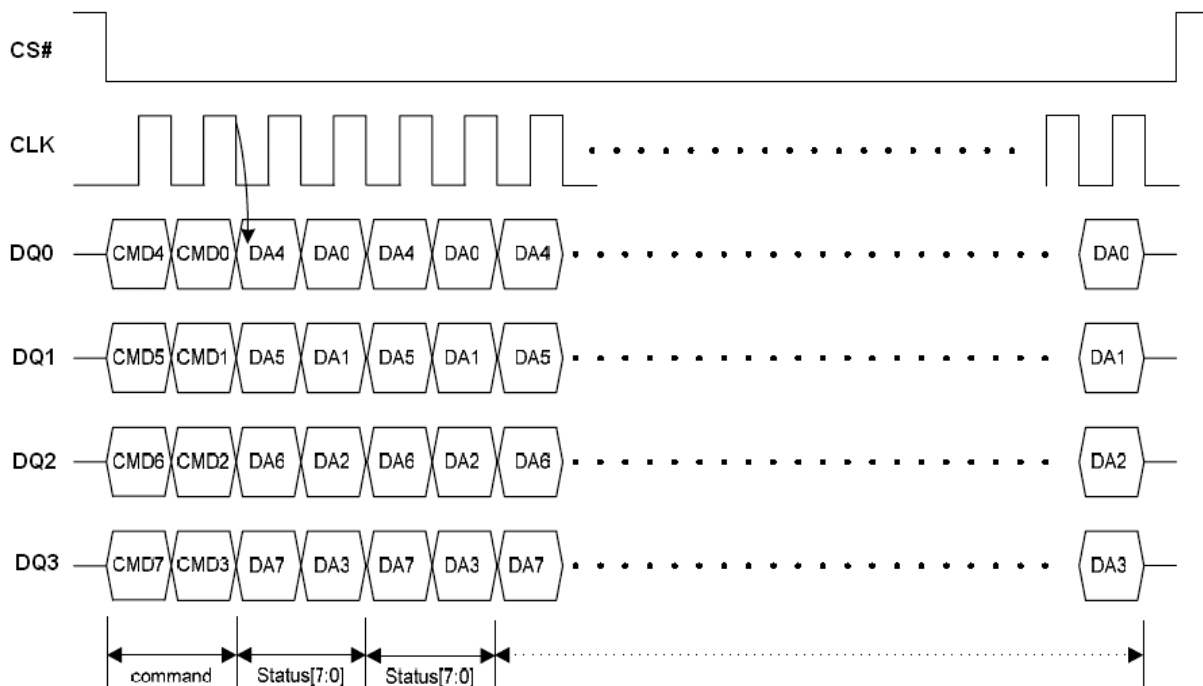


Figure 11.1 Read Status Register Instruction Sequence in QPI Mode



Table 7. Status Register Bit Locations

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
SRP bit	EBL bit (Enable boot lock)	BP3 bit	BP2 bit	BP1 bit	BP0 bit	WEL bit	WIP bit
OTP_LOCK bit	WXDIS bit	HRSW bit	4KBL bit (4KB Boot Lock)	TB bit (Top / Bottom Protect)	Reserved		

Table 7.1 Status Register Bit Locations (In Normal mode)

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
SRP Status Register Protect	EBL bit (Enable Boot Lock)	BP3 bit (Block Protect)	BP2 bit (Block Protect)	BP1 bit (Block Protect)	BP0 bit (Block Protect)	WEL bit (Write Enable Latch)	WIP bit (Write In Progress bit)
1 = status register write disable	1 = Lock selected 64KB- Block/Sector	(note 2)	(note 2)	(note 2)	(note 2)	1 = write enable 0 = not write enable	1 = write operation 0 = not in write operation
Non-volatile/ Volatile bit	Non-volatile/ Volatile bit	Non-volatile/ Volatile bit	Non-volatile/ Volatile bit	Non-volatile/ Volatile bit	Non-volatile/ Volatile bit	indicator bit	indicator bit

Table 7.2 Status Register Bit Locations (In OTP mode)

SR.7	SR.6	SR.5	SR.4	SR.3	SR.2	SR.1	SR.0
OTP_LOCK bit	WXDIS bit (WP# and HOLD#/RESET# disabled)	HRSW bit (HOLD#/RESET# switch)	4KBL bit (4KB Boot Lock)	TB bit (Top / Bottom Protect)	Reserved bit	WEL bit (Write Enable Latch)	WIP bit (Write In Progress bit)
1 = OTP sector is protected	1 = WP# and HOLD#/RESET# disable 0 = WP# and HOLD#/RESET# enable (default 0)	1 = RESET# enable 0 = HOLD# enable (default 0)	1 = Sector 0 = 64KB-Block (default 0)	1 = Bottom 0 = Top (default 0)		1 = write enable 0 = not write enable	1 = write operation 0 = not in write operation
OTP bit	OTP / Volatile bit	OTP / Volatile bit	OTP / Volatile bit	OTP / Volatile bit		indicator bit	indicator bit

Note

1. In OTP mode, SR.7 bit is served as OTP_LOCK bit; SR.6 bit is served as WXDIS bit; SR.5 bit is served as HRSW bit; SR.4 bit is served as 4KBL bit; SR.3 bit is served as TB bit; SR.1 bit is served as WEL bit and SR.0 bit is served as WIP bit.
2. See the table 3 "Protected Area Sizes Sector Organization".
3. When executed the (RDSR) (05h) command, the WIP (SR.0) value is the same as WIP (SR2.0) in table 8.



The status and control bits of the Status Register are as follows:

WIP bit. The Write In Progress (WIP) bit indicates whether the memory is busy with a Write Status Register, Program or Erase cycle. When set to 1, such a cycle is in progress, when reset to 0 no such cycle is in progress.

WEL bit. The Write Enable Latch (WEL) bit indicates the status of the internal Write Enable Latch. When set to 1 the internal Write Enable Latch is set, when set to 0 the internal Write Enable Latch is reset and no Write Status Register, Program or Erase instruction is accepted.

BP3, BP2, BP1, BP0 bits. The Block Protect (BP3, BP2, BP1, BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase instructions. These bits are written with the Write Status Register (WRSR) instruction. When one or both of the Block Protect (BP3, BP2, BP1, BP0) bits is set to 1, the relevant memory area (as defined in Table 3.) becomes protected against Page Program (PP), Quad Input Page Program (QPP), Sector Erase (SE) and , Half Block Erase (HBE), Block Erase (BE), instructions. The Block Protect (BP3, BP2, BP1, BP0) bits can be written and provided that the Hardware Protected mode has not been set. The Chip Erase (CE) instruction is executed if and only if all Block Protect (BP3, BP2, BP1, BP0) bits are 0 and EBL bit is 0.

EBL bit. The Enable Boot Lock (EBL) bit is used to enable the Boot Lock feature. When this bit is programmed to '1', the sector/block selected by the TB bit and 4KBL bit will be locked.

SRP bit. The Status Register Protect (SRP) bit is operated in conjunction with the Write Protect (WP#) signal. The Status Register Write Protect (SRP) bit and Write Protect (WP#) signal allow the device to be put in the Hardware Protected mode (when the Status Register Protect (SRP) bit is set to 1, and Write Protect (WP#) is driven Low). In this mode, the non-volatile bits of the Status Register (SRP, SR.5, SR.4, SR.3, SR.2) become read-only bits and the Write Status Register (WRSR) instruction is no longer accepted for execution.

In OTP mode, SR.7, SR.6, SR.5, SR.4, SR.3, SR.1 and SR.0 are served as OTP_Lock bit, WXDIS bit, HRSW bit, 4KBL bit, TB bit, WEL bit and WIP bit.

TB bit. The Top/Bottom Protect Bit (TB) controls if the Block Protect Bits (BP3, BP2, BP1, BP0) protect from the Top (TB = 0) or the Bottom (TB = 1) of the array as shown in the Status Register Memory Protection table. It also controls if the Top (TB=0) or the Bottom (TB=1) 64KB-block/sector is protected when Boot Lock feature is enabled. The factory default setting is TB = 0. The TB bit can be set with the Write Status Register instruction in OTP mode.

4KBL bit, The 4KB Boot Lock bit (4KBL) is set by WRSR command in OTP mode. It is used to set the protection area size as block (64KB) or sector (4KB).

WXDIS bit. The WP# and HOLD#/RESET# Disable bit (WXDIS bit), OTP / Volatile bit, it indicates the WP# and HOLD#/RESET# are enabled or not. When it is "0" (factory default), the WP# and HOLD#/RESET# are enabled. On the other hand, while WXDIS bit is "1", the WP# and HOLD#/RESET# are disabled. If the system executes Quad mode commands, this WXDIS bit becomes no affection since WP# and HOLD#/RESET# function will be disabled by Quad mode commands.

HRSW bit. The HOLD#/RESET# switch bit (HRSW bit), OTP / Volatile bit, the HRSW bit is used to determine whether HOLD# or RESET# function should be implemented on the hardware pin. When it is "0" (factory default), the pin acts as HOLD#; when it is "1", the pin acts as RESET#. However, HOLD# or RESET# functions are only available when WXDIS bit is "0". If WXDIS bit is set to "1", the HOLD# and RESET# functions are disabled, the pin acts as a dedicated data I/O pin.

OTP_LOCK bit. This bit is served as OTP_LOCK bit, user can read/program/erase OTP sector as normal sector while OTP_LOCK value is equal 0, after OTP_LOCK is programmed with 1 by WRSR command, the OTP sector is protected from program and erase operation. The OTP_LOCK bit can only be programmed once.

Reserved bit. Status Register bit locations SR.2 in OTP mode is reserved for future use.



Read Status Register 3 (RDSR 3) (95h)

The Read Status Register 3 (RDSR3) instruction allows the Status Register 3 to be read. The Status Register 3 may be read at any time. When one of these bytes is in progress, it is recommended to check the Write In Progress (WIP) bit before sending a new instruction to the device. It is also possible to read the Read Status Register 3 continuously, as shown in Figure 13.

The instruction sequence is shown in Figure 13.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

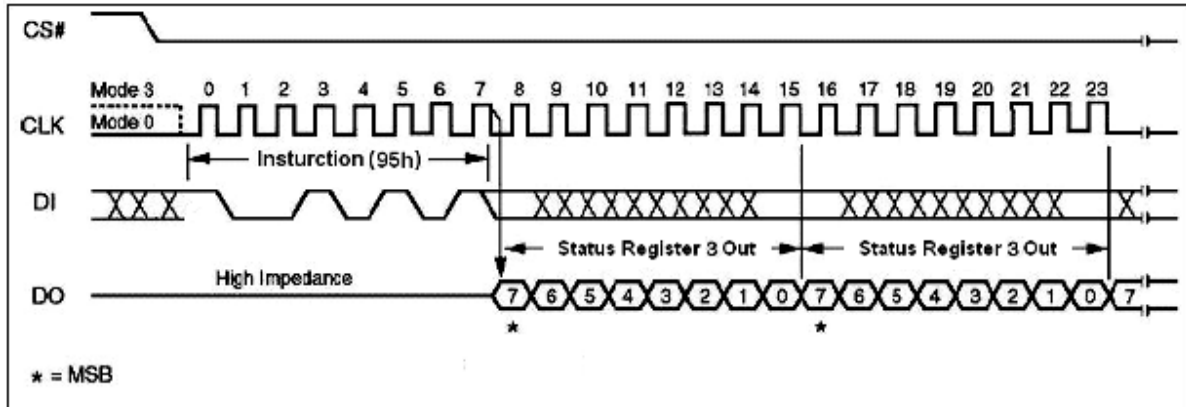


Figure 13. Read Status Register 3 Instruction Sequence Diagram

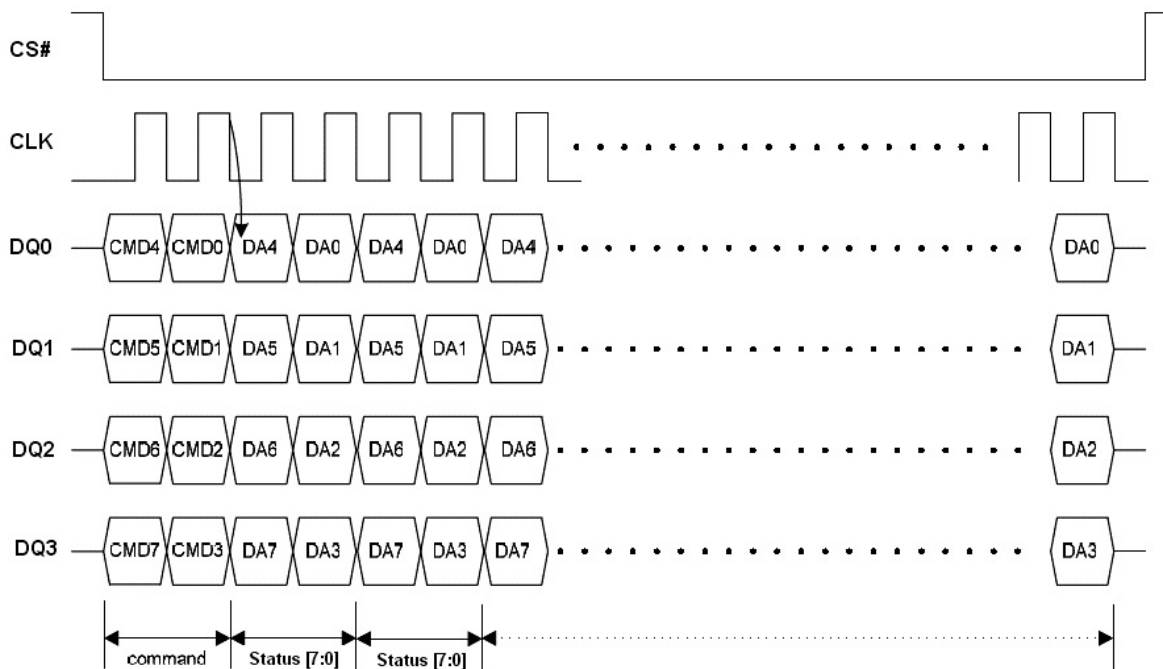


Figure 13.1 Read Status Register 3 Instruction Sequence in QPI Mode



The status and control bits of the Status Register 3 are as follows:

Output Drive Strength. The Output Drive Strength (SR3.3 and SR3.2) bits indicate the status of output Drive Strength in I/O pins.

Dummy Byte. The Dummy Byte (SR3.5 and SR3.4) bits indicate the status of the number of dummy byte in high performance read.

Reserved bit. SR3.7, SR3.6, SR3.1 and SR3.0 are reserved for future use.

Table 9. Status Register 3 Bit Locations

SR3.7	SR3.6	SR3.5	SR3.4	SR3.3	SR3.2	SR3.1	SR3.0
Reserved	Reserved	Dummy Byte ⁽¹⁾ Default = 00		Output Drive Strength		Reserved	Reserved
		00 = 3 Bytes 01 = 2 Bytes 10 = 4 Bytes 11 = 5 Bytes	00 = 67% Default 01 = 100% 10 = 50% (1/2) Drive 11 = 33% (1/3) Drive				
volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit	volatile bit

Note:

- 2 Bytes (4 clocks in Quad mode), 3 Bytes (6 clocks in Quad mode), 4 Bytes (8 clocks in Quad mode), 5 Bytes (10 clocks in Quad mode)

Table 10. SR3.4 and SR3.5 Status (for Dummy Bytes)

Instruction Name	Op Code	Start Address ⁽¹⁾	Dummy Byte settings
			<=104MHz
Fast Read	0Bh	Byte	00 (3)
		Word	01 (2)
		Dword	01 (2)
Quad IO Fast Read	EBh	Byte	00 (3)
		Word	01 (2)
		Dword	01 (2)

Note 1:

“Dword” means the start address is 4-byte aligned (i.e. Start Address is 0, 4, 8...), “Word” means the start address is 2-byte aligned (i.e. Start Address is 0, 2, 4, 8...) and “Byte” means the start address can be anywhere without 2-byte or 4-byte aligned.

Write Status Register (WRSR) (01h)

The Write Status Register (WRSR) instruction allows new values to be written to the Status Register. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded and executed, the device sets the Write Enable Latch (WEL).

The Write Status Register (WRSR) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code and the data byte on Serial Data Input (DI).

The instruction sequence is shown in Figure 14. The Write Status Register (WRSR) instruction has no effect on S1 and S0 of the Status Register. Chip Select (CS#) must be driven High after the eighth bit of the data byte has been latched in. If not, the Write Status Register (WRSR) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Write Status Register cycle (whose duration is t_w) is initiated. While the Write Status Register cycle is in progress, the Status Register may still be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Write Status Register cycle, and is 0 when it is completed. When the cycle is completed, the Write Enable Latch (WEL) is reset.

The Write Status Register (WRSR) instruction allows the user to change the values of the Block Protect (BP3, BP2, BP1, BP0) bits, to define the size of the area that is to be treated as read-only, as defined in



Table 3. The Write Status Register (WRSR) instruction also allows the user to set or reset the Status Register Protect (SRP) bit in accordance with the Write Protect (WP#) signal. The Status Register Protect (SRP) bit and Write Protect (WP#) signal allow the device to be put in the Hardware Protected Mode (HPM). The Write Status Register (WRSR) instruction is not executed once the Hardware Protected Mode (HPM) is entered.

The instruction sequence is shown in Figure 14.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

NOTE :

In the OTP mode without enabling Volatile Status Register function (50h), WRSR command is used to program OTP_LOCK bit, WXDIS bit, HRSW bit, TB bit and 4KBL bit to '1', but these bits can only be programmed once.

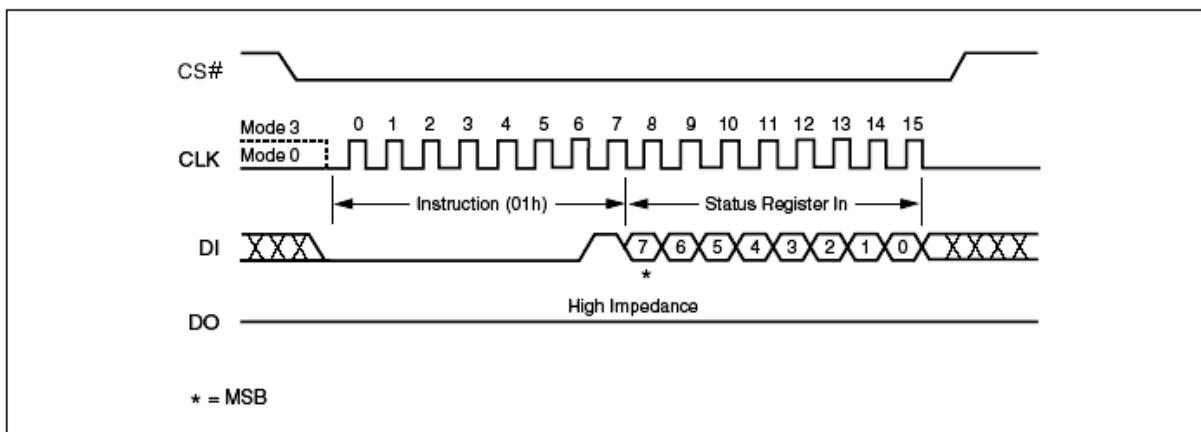


Figure 14. Write Status Register Instruction Sequence Diagram

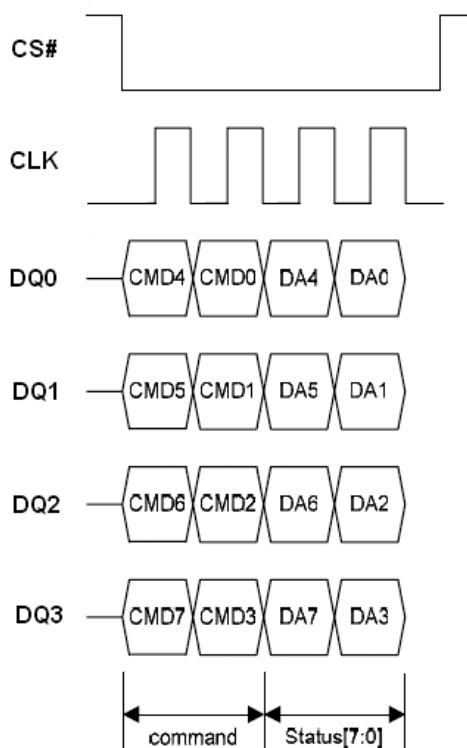


Figure 14.1 Write Status Register Instruction Sequence in QPI Mode

Read Data Bytes (READ) (03h)

The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read Data Bytes (READ) instruction is followed by a 3-byte address (A23-A0), each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency f_R , during the falling edge of Serial Clock (CLK).

The instruction sequence is shown in Figure 15. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Read Data Bytes (READ) instruction. When the highest address is reached, the address counter rolls over to 000000h, allowing the read sequence to be continued indefinitely.

The Read Data Bytes (READ) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes (READ) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

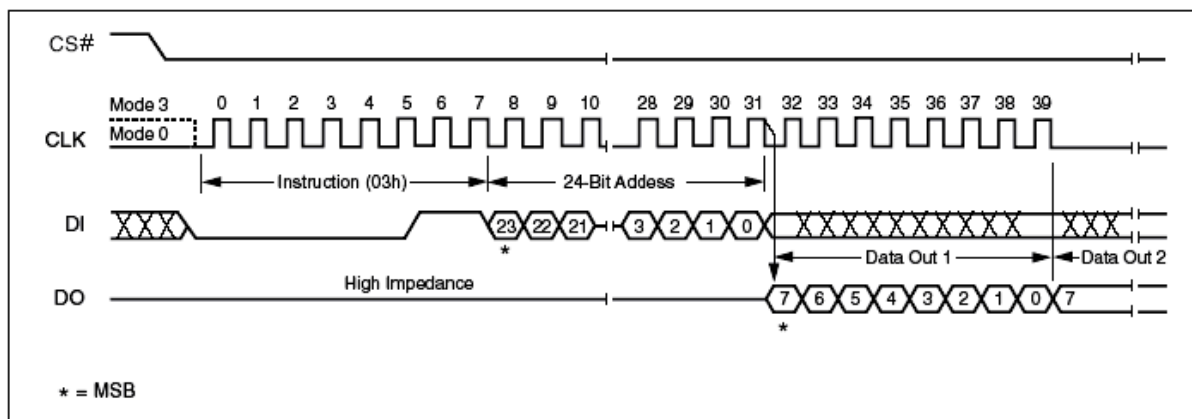


Figure 15. Read Data Instruction Sequence Diagram

Read Data Bytes at Higher Speed (FAST_READ) (0Bh)

The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read Data Bytes at Higher Speed (FAST_READ) instruction is followed by a 3-byte address (A23-A0) and a dummy byte, each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency F_R , during the falling edge of Serial Clock (CLK).

The instruction sequence is shown in Figure 16. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Read Data Bytes at Higher Speed (FAST_READ) instruction. When the highest address is reached, the address counter rolls over to 000000h, allowing the read sequence to be continued indefinitely.

The Read Data Bytes at Higher Speed (FAST_READ) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes at Higher Speed (FAST_READ) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

The instruction sequence is shown in Figure 16.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

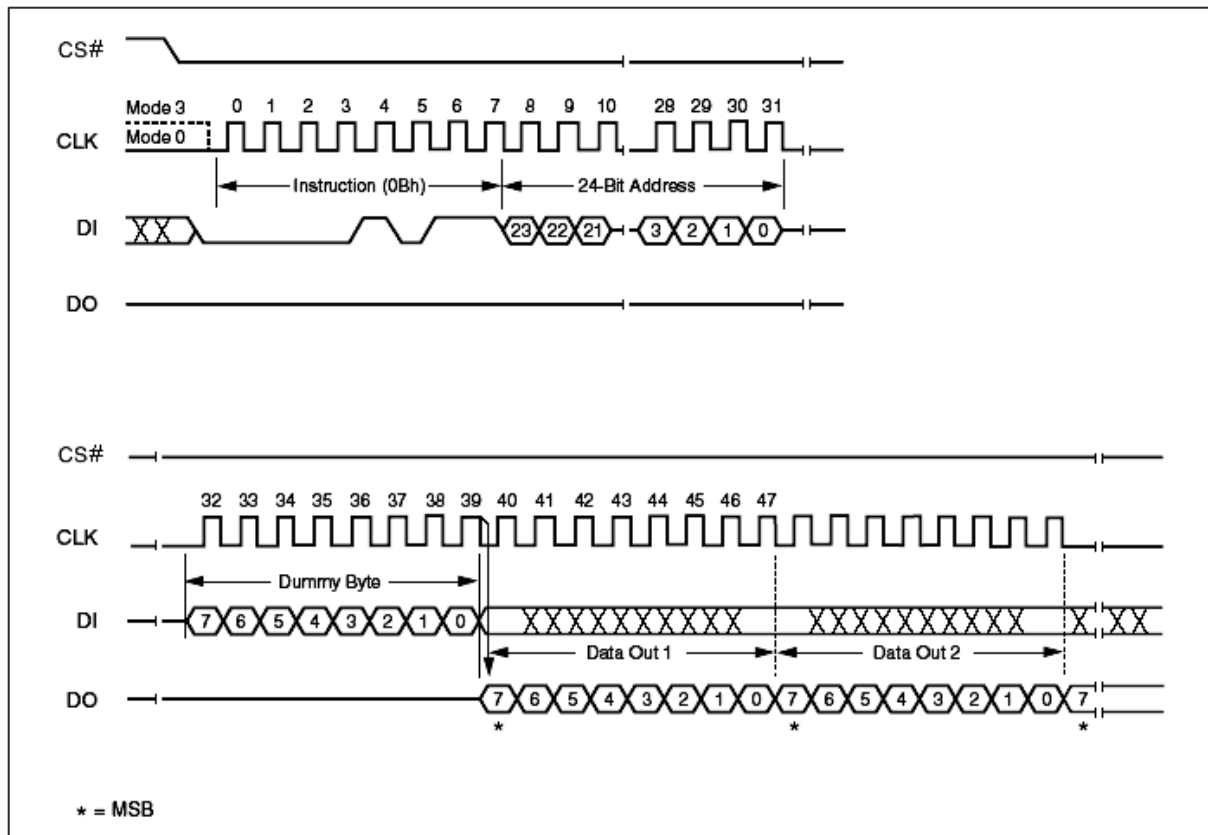


Figure 16. Fast Read Instruction Sequence Diagram

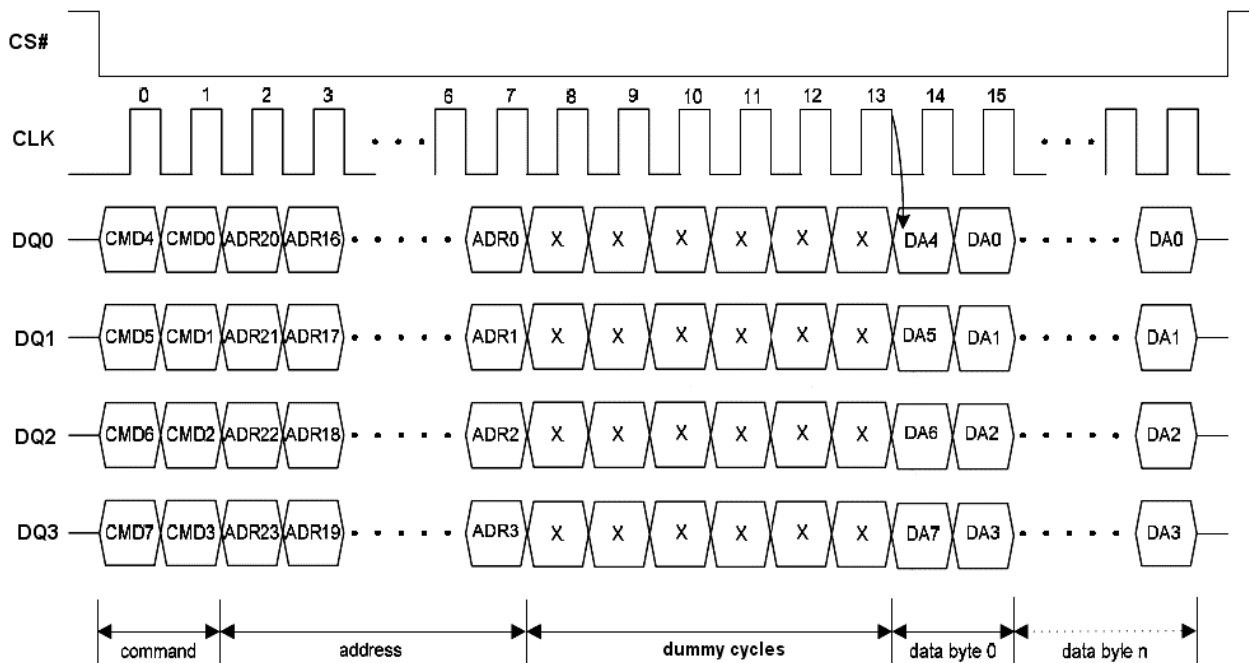


Figure 16.1 Fast Read Instruction Sequence in QPI Mode



Dual Output Fast Read (3Bh)

The Dual Output Fast Read (3Bh) is similar to the standard Fast Read (0Bh) instruction except that data is output on two pins, DQ₀ and DQ₁, instead of just DQ₀. This allows data to be transferred from the EN25QH128A(2T) at twice the rate of standard SPI devices. The Dual Output Fast Read instruction is ideal for quickly downloading code from to RAM upon power-up or for applications that cache code-segments to RAM for execution.

Similar to the Fast Read instruction, the Dual Output Fast Read instructions can operation at the highest possible frequency of FR (see AC Electrical Characteristics). This is accomplished by adding eight “dummy clocks after the 24-bit address as shown in Figure 17. The dummy clocks allow the device’s internal circuits additional time for setting up the initial address. The input data during the dummy clock is “don’t care”. However, the DI pin should be high-impedance prior to the falling edge of the first data out clock.

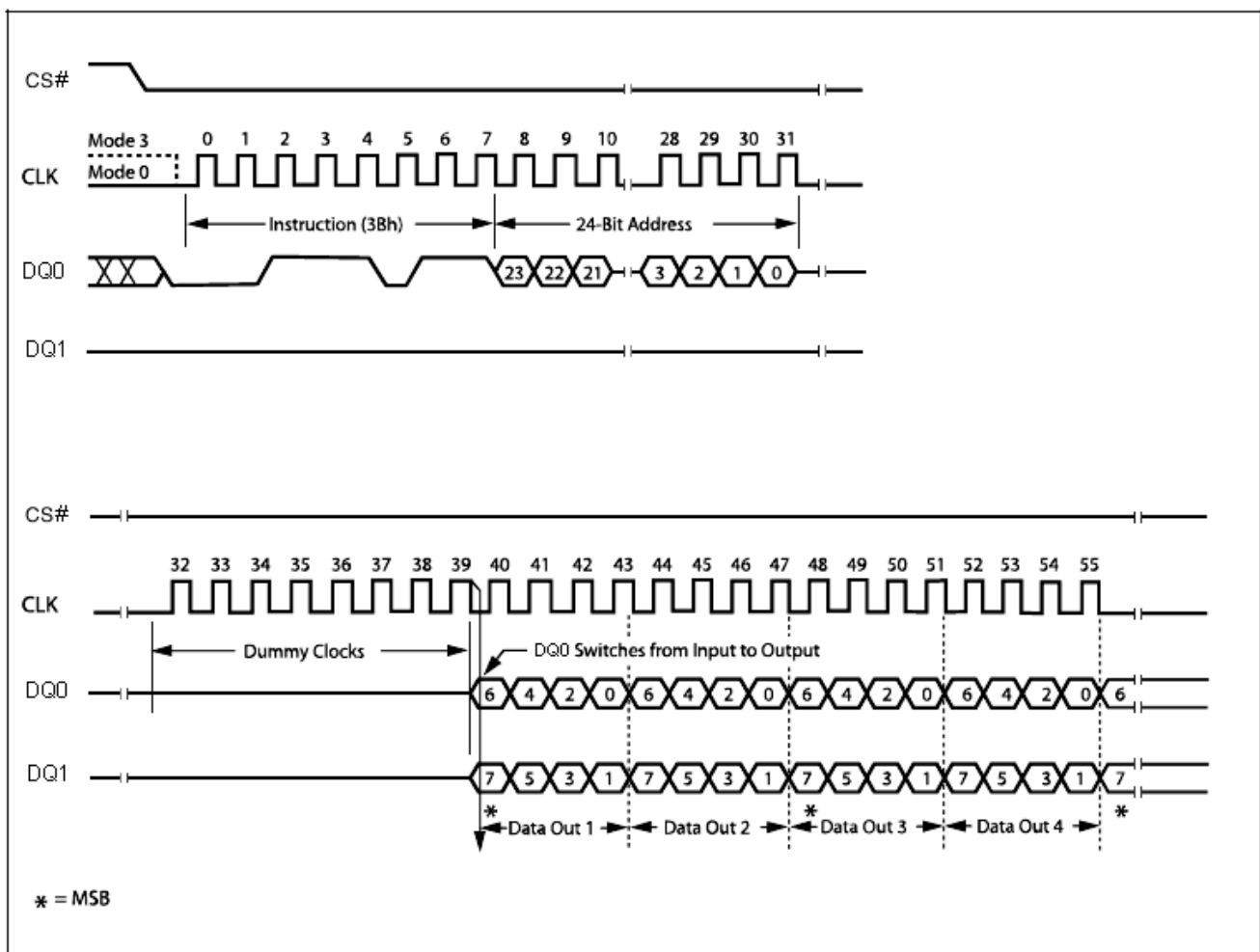


Figure 17. Dual Output Fast Read Instruction Sequence Diagram



Dual Input / Output FAST_READ (BBh)

The Dual I/O Fast Read (BBh) instruction allows for improved random access while maintaining two IO pins, DQ₀ and DQ₁. It is similar to the Dual Output Fast Read (3Bh) instruction but with the capability to input the Address bits (A23-0) two bits per clock. This reduced instruction overhead may allow for code execution (XIP) directly from the Dual SPI in some applications.

The Dual I/O Fast Read instruction enable double throughput of Serial Flash in read mode. The address is latched on rising edge of CLK, and data of every two bits (interleave 2 I/O pins) shift out on the falling edge of CLK at a maximum frequency. The first address can be at any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single Dual I/O Fast Read instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing Dual I/O Fast Read instruction, the following address/dummy/data out will perform as 2-bit instead of previous 1-bit, as shown in Figure 18.

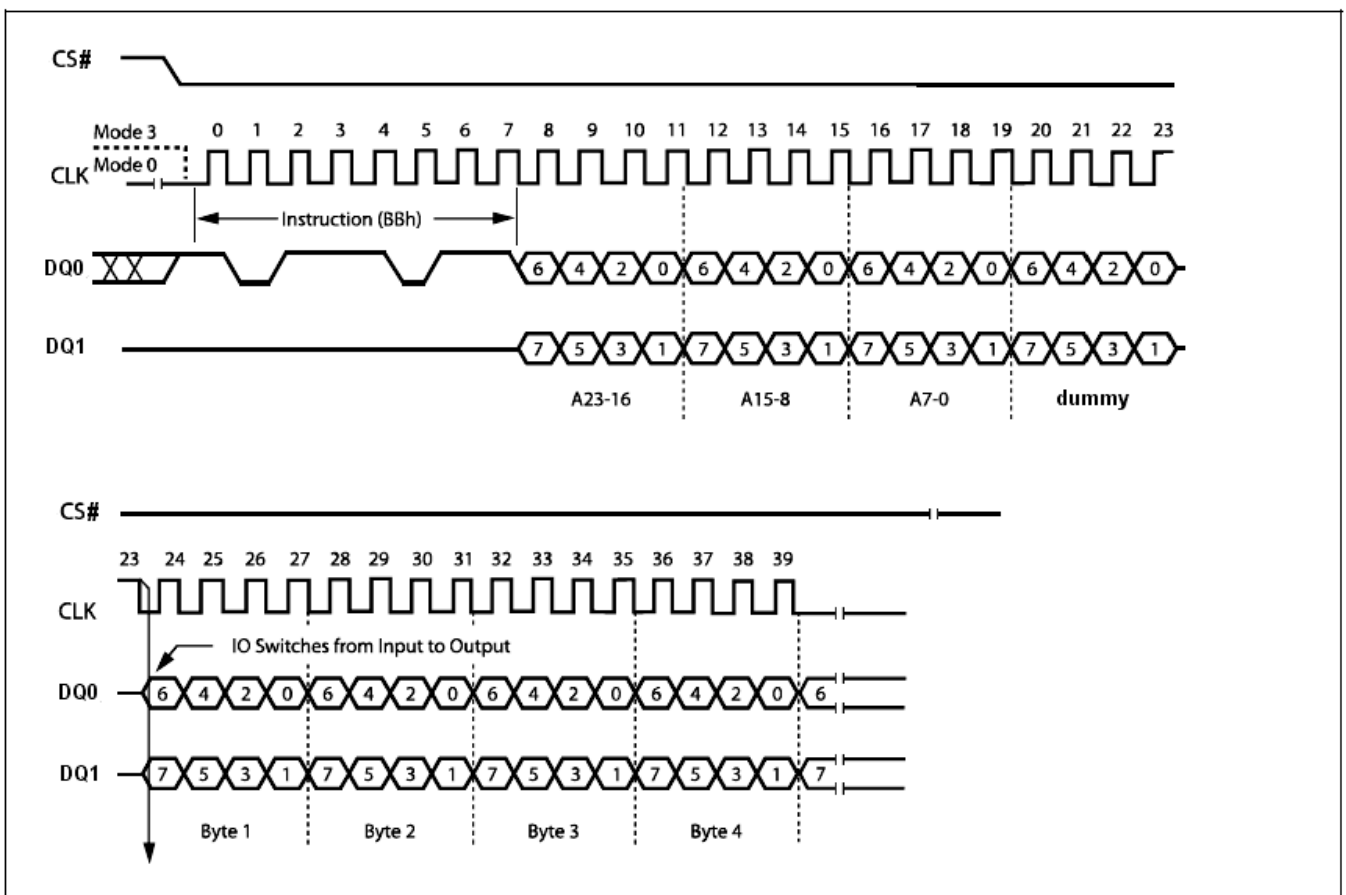


Figure 18. Dual Input / Output Fast Read Instruction Sequence Diagram



Quad Output Fast Read (6Bh)

The Quad Output Fast Read (6Bh) instruction is similar to the Dual Output Fast Read (3Bh) instruction except that data is output through four pins, DQ₀, DQ₁, DQ₂ and DQ₃ and eight dummy clocks are required prior to the data output. The Quad Output dramatically reduces instruction overhead allowing faster random access for code execution (XIP) directly from the Quad SPI.

The Quad Output Fast Read (6Bh) address is latching on rising edge of CLK, and data of every four bits (interleave on 4 I/O pins) shift out on the falling edge of CLK at a maximum frequency F_R. The first address can be any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single Quad Output Fast Read instruction. The address counter rolls over to 0 when the highest address has been reached.

The sequence of issuing Quad Output Fast Read (6Bh) instruction is: CS# goes low -> sending Quad Output Fast Read (6Bh) instruction -> 24-bit address on DQ₀-> 8 dummy clocks -> data out interleave on DQ₃, DQ₂, DQ₁ and DQ₀-> to end Quad Output Fast Read (6Bh) operation can use CS# to high at any time during data out, as shown in Figure 19. The WP#(DQ₂) and HOLD#/RESET#(DQ₃) need to drive high before address input if WXDIS bit in Status Register is 0.

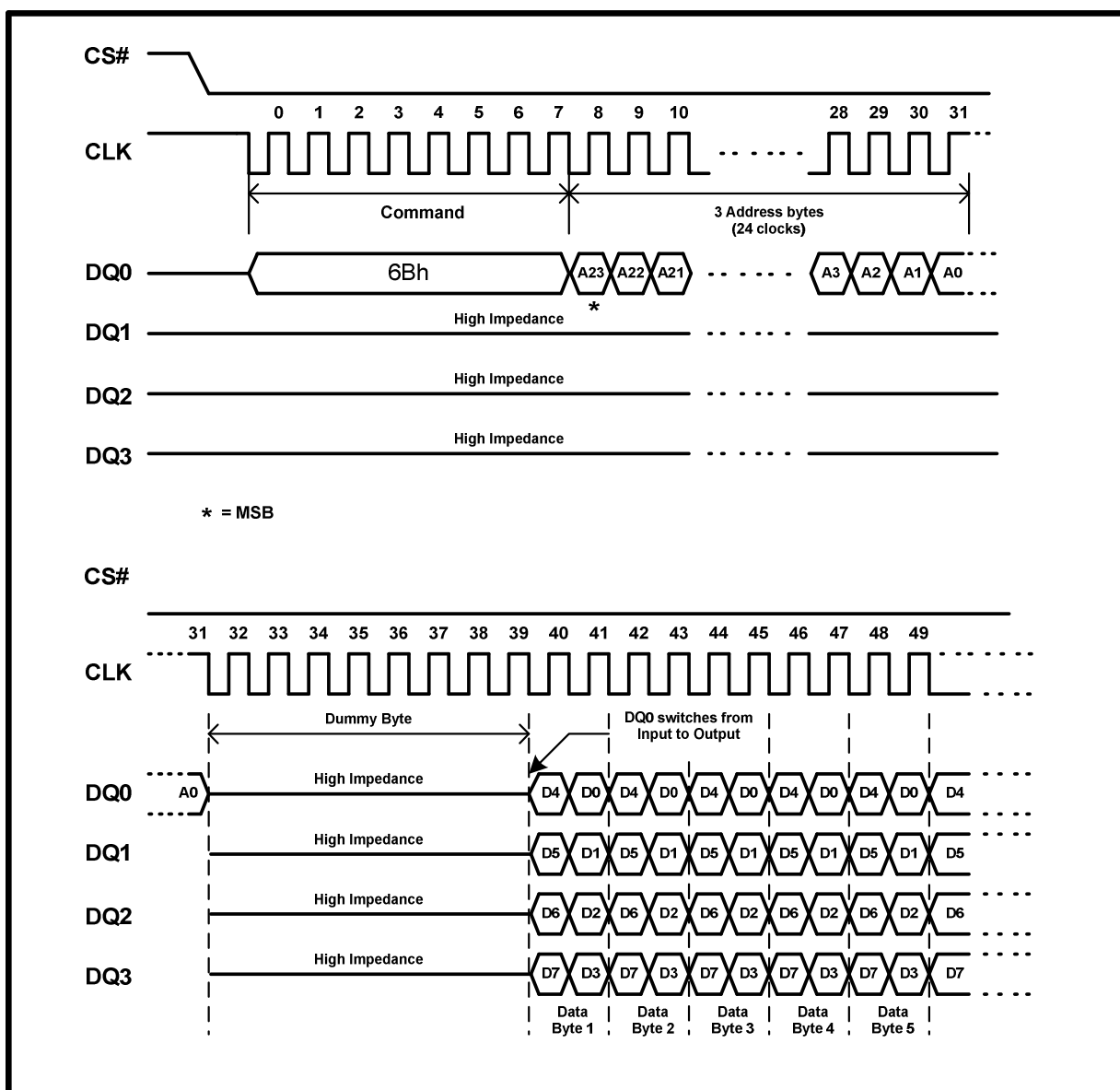


Figure 19. Quad Output Fast Read Instruction Sequence Diagram



Quad Input / Output FAST_READ (EBh)

The Quad Input/Output FAST_READ (EBh) instruction is similar to the Dual I/O Fast Read (BBh) instruction except that address and data bits are input and output through four pins, DQ₀, DQ₁, DQ₂ and DQ₃ and six dummy clocks are required prior to the data output. The Quad I/O dramatically reduces instruction overhead allowing faster random access for code execution (XIP) directly from the Quad SPI.

The Quad Input/Output FAST_READ (EBh) instruction enable quad throughput of Serial Flash in read mode. The address is latching on rising edge of CLK, and data of every four bits (interleave on 4 I/O pins) shift out on the falling edge of CLK at a maximum frequency F_R. The first address can be any location. The address is automatically increased to the next higher address after each byte data is shifted out, so the whole memory can be read out at a single Quad Input/Output FAST_READ instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing Quad Input/Output FAST_READ instruction, the following address/dummy/data out will perform as 4-bit instead of previous 1-bit.

The sequence of issuing Quad Input/Output FAST_READ (EBh) instruction is: CS# goes low -> sending Quad Input/Output FAST_READ (EBh) instruction -> 24-bit address interleave on DQ₃, DQ₂, DQ₁ and DQ₀ -> 6 dummy clocks -> data out interleave on DQ₃, DQ₂, DQ₁ and DQ₀ -> to end Quad Input/Output FAST_READ (EBh) operation can use CS# to high at any time during data out, as shown in Figure 20. The instruction sequence is shown in Figure 20.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

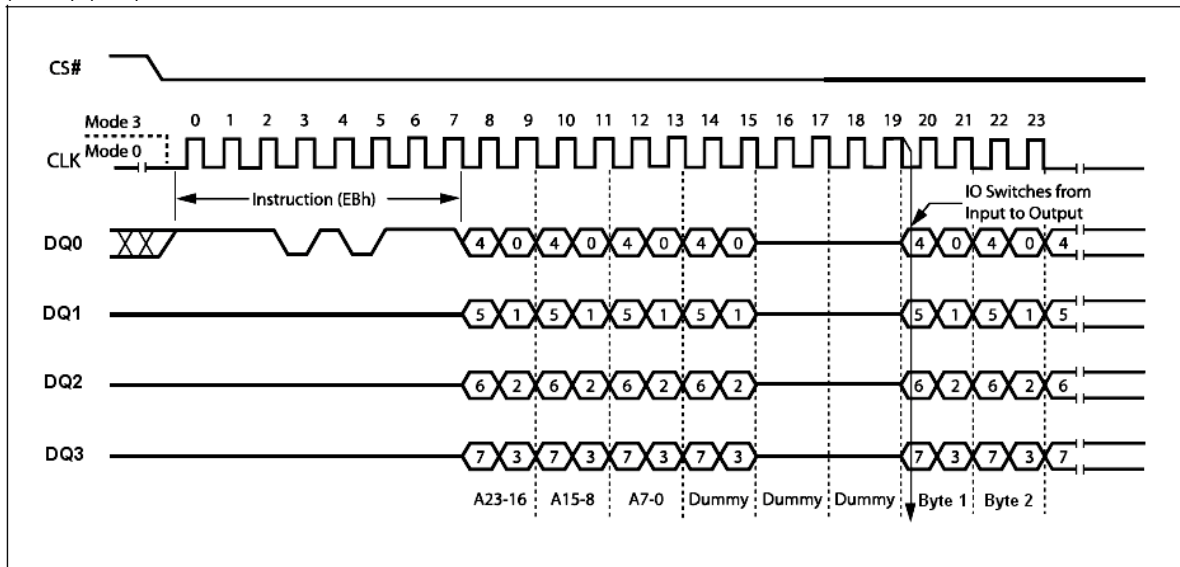


Figure 20. Quad Input / Output Fast Read Instruction Sequence Diagram

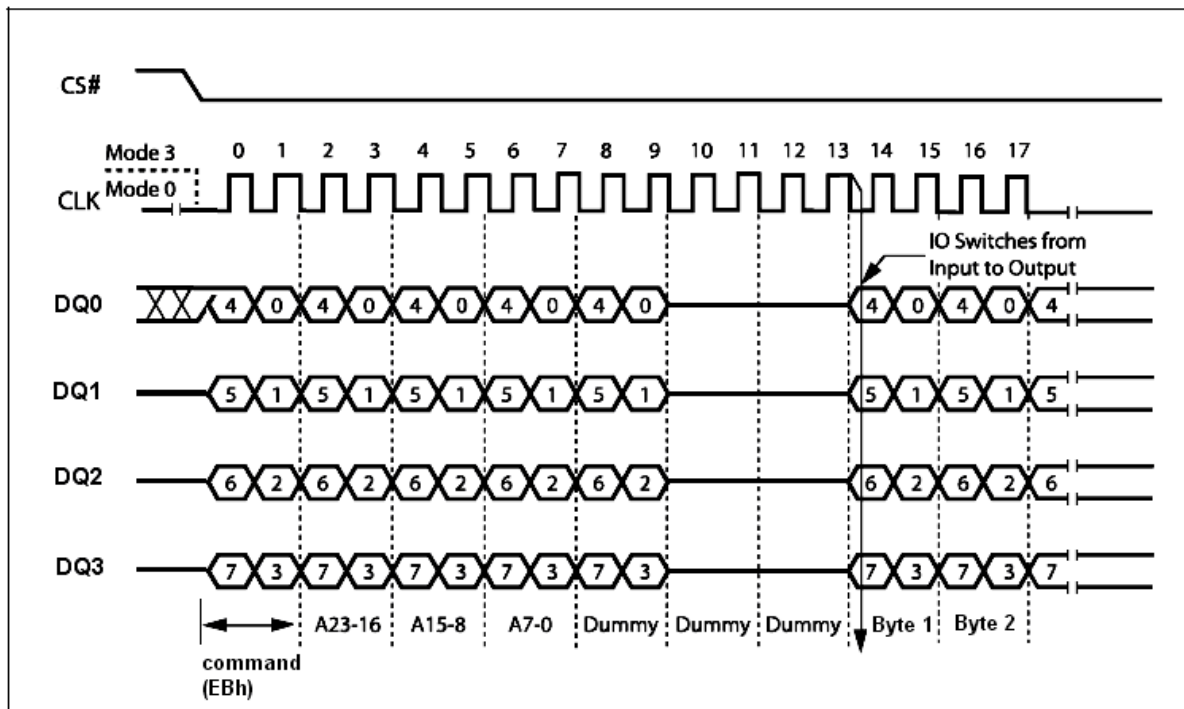


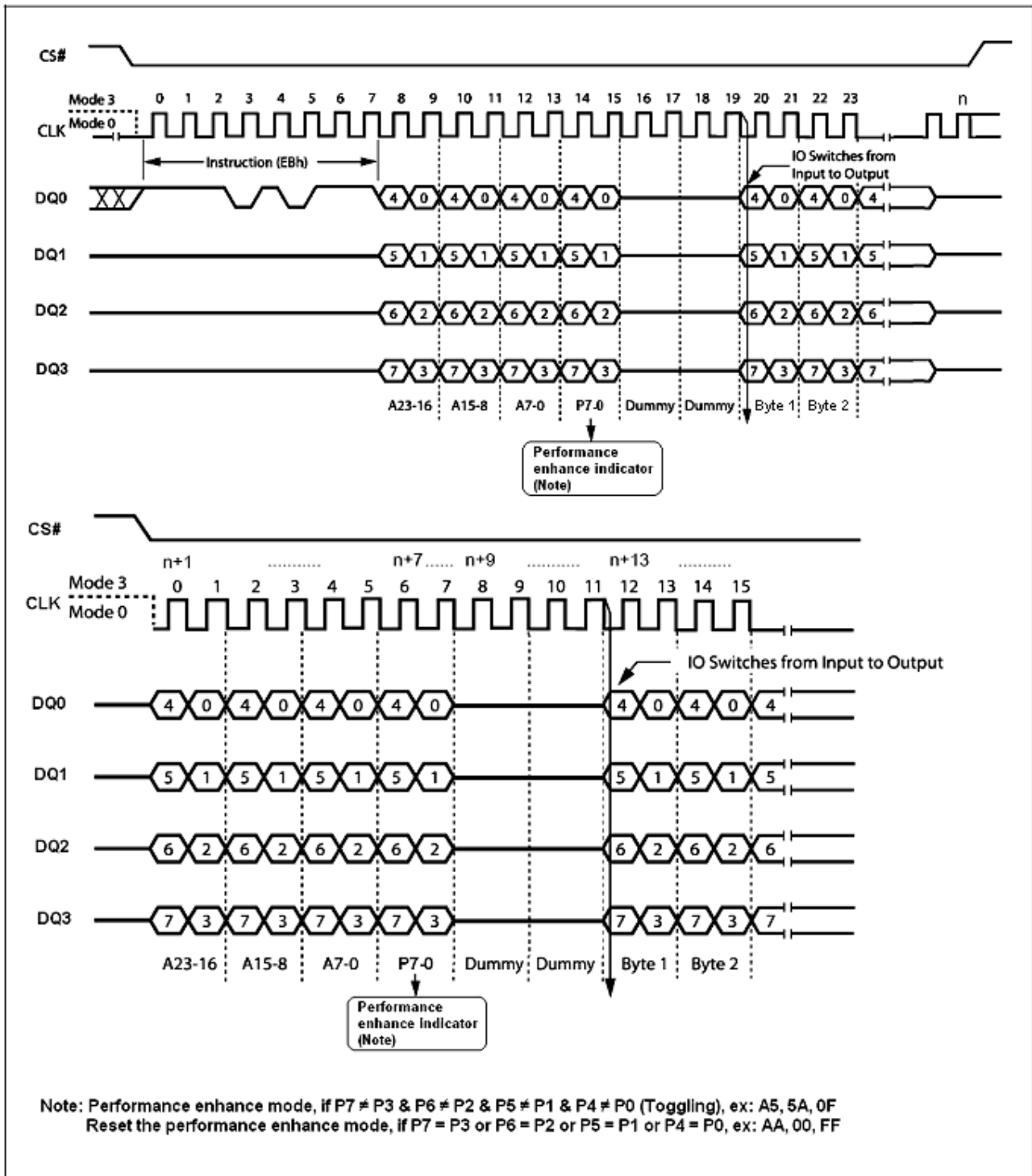
Figure 20.1. Quad Input / Output Fast Read Instruction Sequence in QPI Mode

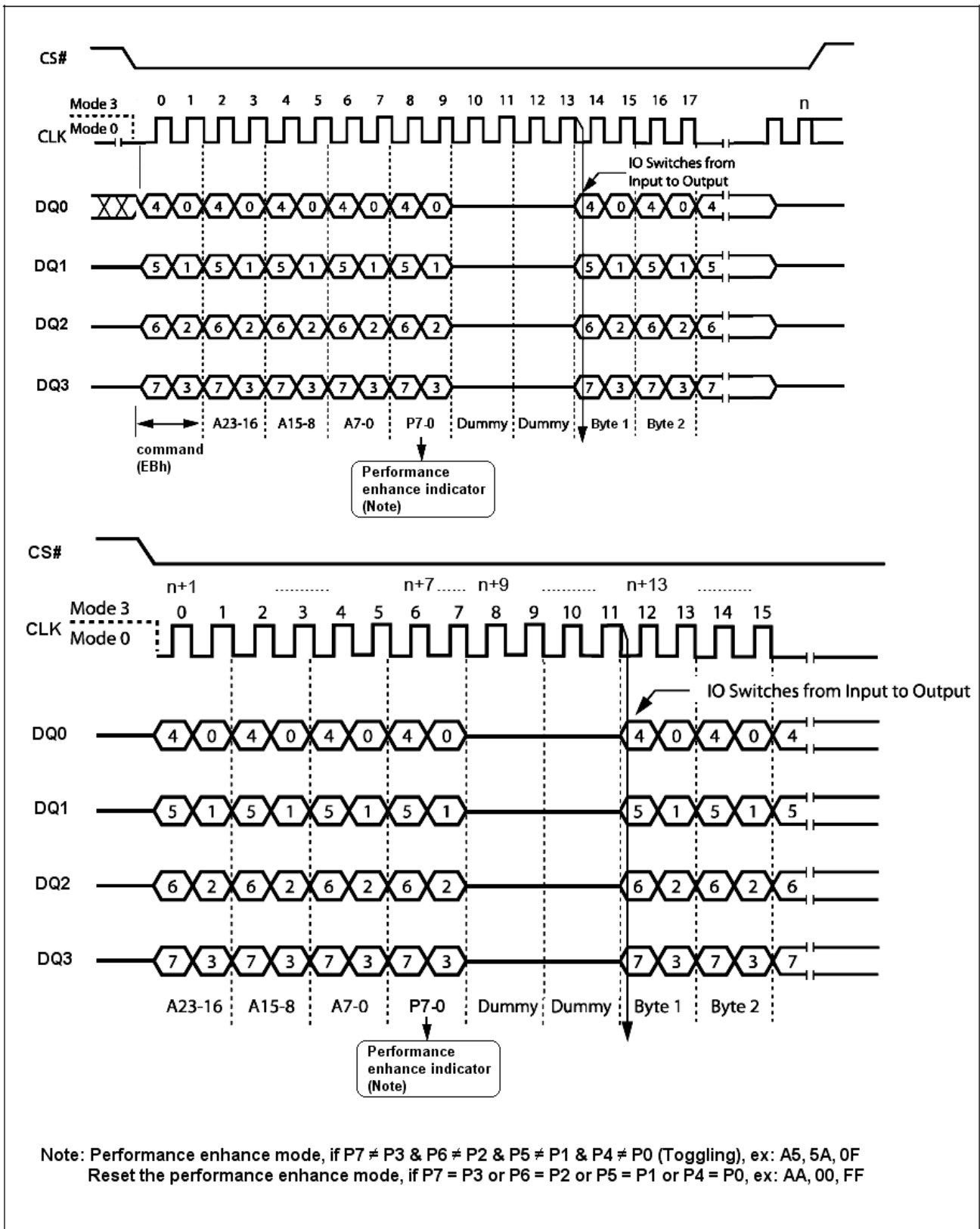
Another sequence of issuing Quad Input/Output FAST_READ (EBh) instruction especially useful in random access is : CS# goes low -> sending Quad Input/Output FAST_READ (EBh) instruction -> 24-bit address interleave on DQ₃, DQ₂, DQ₁ and DQ₀ -> performance enhance toggling bit P[7:0] -> 4 dummy clocks -> data out interleave on DQ₃, DQ₂, DQ₁ and DQ₀ till CS# goes high -> CS# goes low (reduce Quad Input/Output FAST_READ (EBh) instruction) -> 24-bit random access address, as shown in Figure 21.

In the performance – enhancing mode, P[7:4] must be toggling with P[3:0] ; likewise P[7:0] = A5h, 5Ah, F0h or 0Fh can make this mode continue and reduce the next Quad Input/Output FAST_READ (EBh) instruction. Once P[7:4] is no longer toggling with P[3:0] ; likewise P[7:0] = FFh, 00h, AAh or 55h. These commands will reset the performance enhance mode. And afterwards CS# is raised or issuing FFh (CS# goes high -> CS# goes low -> sending FFh -> CS# goes high) instead of no toggling, the system then will escape from performance enhance mode and return to normal operation.

While Program/ Erase/ Write Status Register is in progress, Quad Input/Output FAST_READ (EBh) instruction is rejected without impact on the Program/ Erase/ Write Status Register current cycle.

The instruction sequence is shown in Figure 21.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 21. Quad Input/Output Fast Read Enhance Performance Mode Sequence Diagram


Figure 21.1 Quad Input/Output Fast Read Enhance Performance Mode Sequence in QPI Mode



Write Status Register 3 (C0h)

The Write Status Register 3 (C0h) command can be used to set output drive strength in I/O pins and the number of dummy byte in high performance read. To set the output drive strength and the number of dummy byte to the host driver CS# low, sends the Write Status Register 3 (C0h) and one data byte, then drivers CS# high, After power-up or reset, the output drive strength is set to full drive (00b) and the dummy byte is set to 3 bytes (00b), please refer to Table 9 for Status Register 3 data and Figure 22 for the sequence. In QPI mode, a cycle is two nibbles, or two clocks, most significant nibble first.

The instruction sequence is shown in Figure 22.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

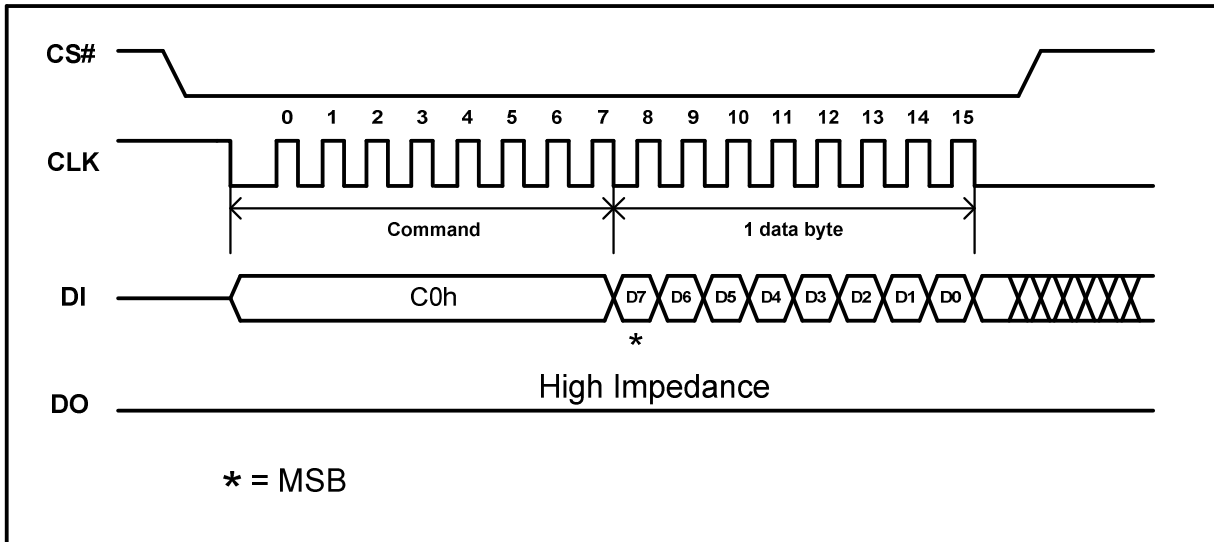
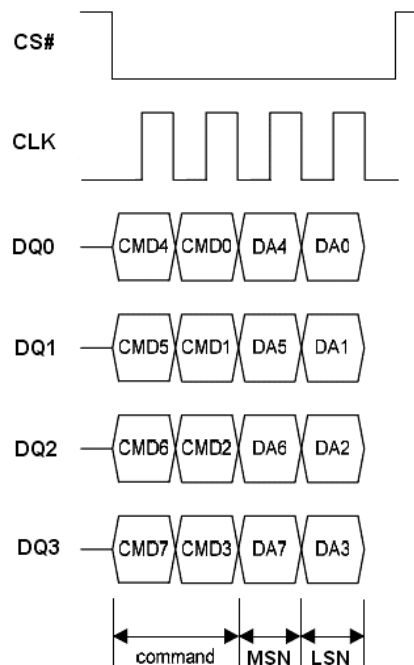


Figure 22. Write Status Register 3 Instruction Sequence Diagram



Note: MSN = Most Significant Nibble,
LSN = Least Significant Nibble

Figure 22.1 Write Status Register 3 Instruction Sequence Diagram in QPI mode



Page Program (PP) (02h)

The Page Program (PP) instruction allows bytes to be programmed in the memory. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Page Program (PP) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, three address bytes and at least one data byte on Serial Data Input (DI). If the 8 least significant address bits (A7-A0) are not all zero, all transmitted data that goes beyond the end of the current page are programmed from the start address of the same page (from the address whose 8 least significant bits (A7-A0) are all zero). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 23. If more than 256 bytes are sent to the device, previously latched data are discarded and the last 256 data bytes are guaranteed to be programmed correctly within the same page. If less than 256 Data bytes are sent to device, they are correctly programmed at the requested addresses without having any effects on the other bytes of the same page.

Chip Select (CS#) must be driven High after the eighth bit of the last data byte has been latched in, otherwise the Page Program (PP) instruction is not executed.

As soon as Chip Select (CS#) is driven high, the self-timed Page Program cycle (whose duration is t_{pp}) is initiated. While the Page Program cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Page Program cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Page Program (PP) instruction applied to a page which is protected by the Block Protect (BP3, BP2, BP1, BP0) bits (see Table 3) is not executed.

The instruction sequence is shown in Figure 23.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

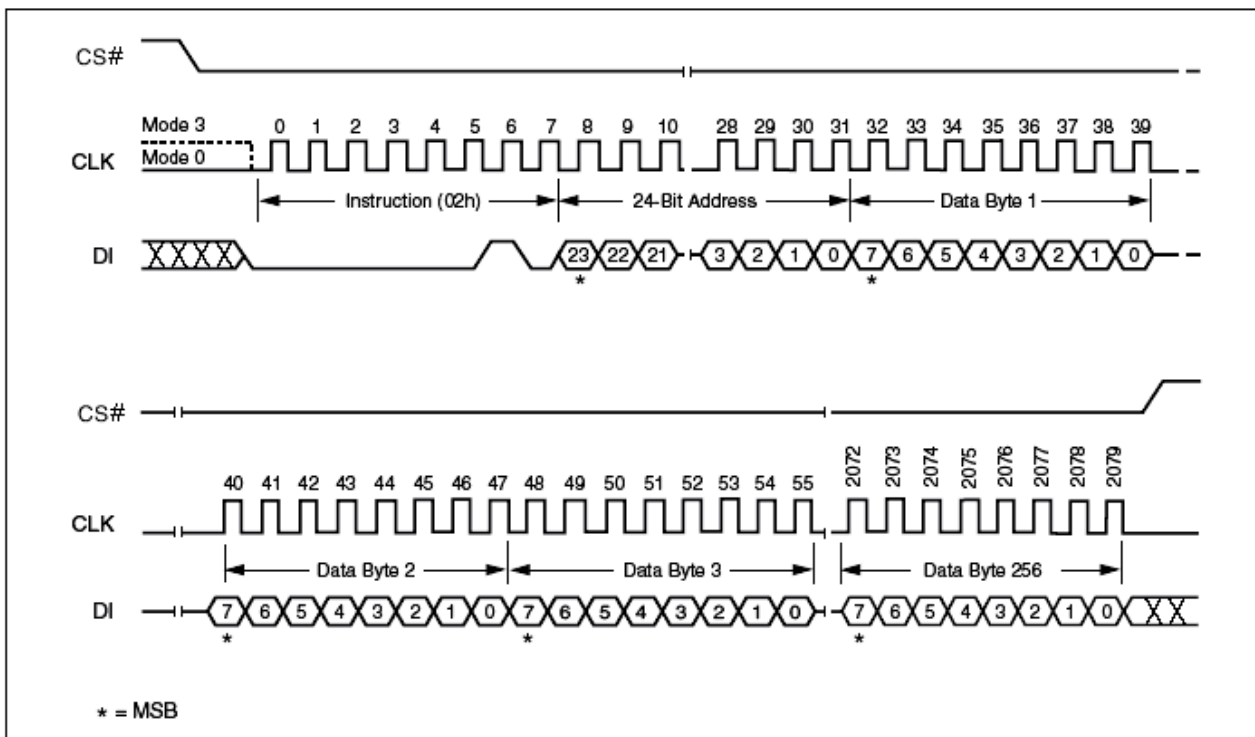


Figure 23. Page Program Instruction Sequence Diagram

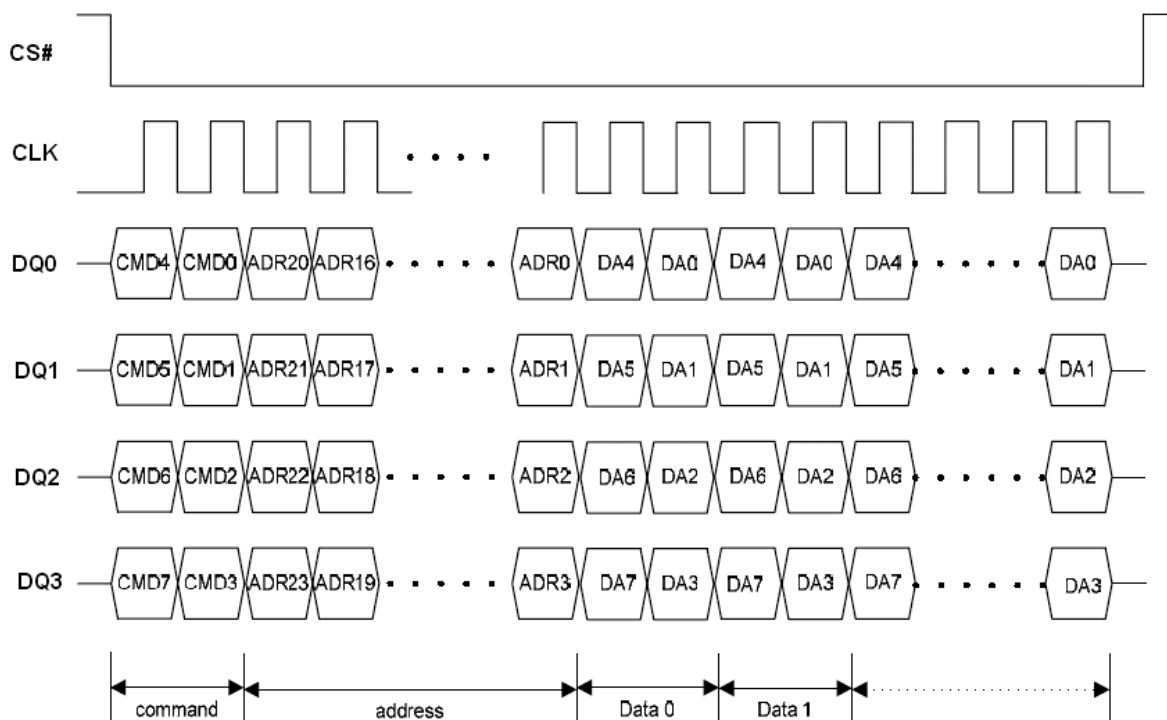


Figure 23.1 Program Instruction Sequence in QPI Mode

Quad Input Page Program (QPP) (32h)

The Quad Page Program (QPP) instruction allows up to 256 bytes of data to be programmed at previously erased (FFh) memory locations using four pins: DQ₀, DQ₁, DQ₂ and DQ₃. The Quad Page Program can improve performance for PROM Programmer and applications that have slow clock speeds < 5MHz. Systems with faster clock speed will not realize much benefit for the Quad Page Program instruction since the inherent page program time is much greater than the time it take to clock-in the data.

To use Quad Page Program (QPP) the WP# and HOLD#/RESET# Disable (WXDIS) bit in Status Register must be set to 1. A Write Enable instruction must be executed before the device will accept the Quad Page Program (QPP) instruction (SR.1, WEL=1). The instruction is initiated by driving the CS# pin low then shifting the instruction code “32h” followed by a 24-bit address (A23-A0) and at least one data byte, into the IO pins. The CS# pin must be held low for the entire length of the instruction while data is being sent to the device. All other functions of Quad Page Program (QPP) are identical to standard Page Program. The Quad Page Program (QPP) instruction sequence is shown in Figure 24.

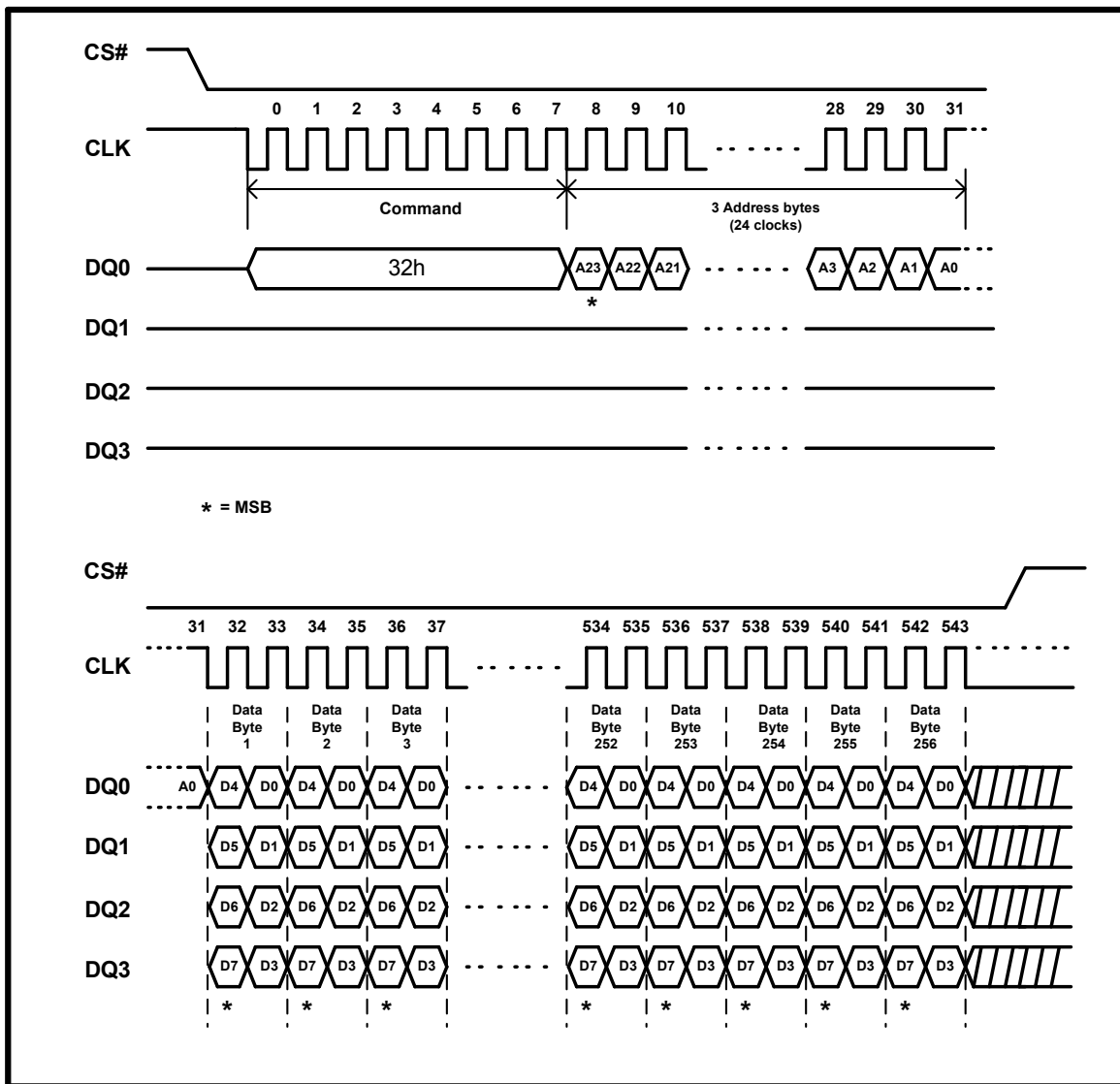


Figure 24. Quad Input Page Program Instruction Sequence Diagram (SPI Mode only)

Sector Erase (SE) (20h)

The Sector Erase (SE) instruction sets to 1 (FFh) all bits inside the chosen sector. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Sector Erase (SE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Sector (see Table 2) is a valid address for the Sector Erase (SE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 28. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Sector Erase (SE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Sector Erase cycle (whose duration is t_{SE}) is initiated. While the Sector Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Sector Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Sector Erase (SE) instruction applied to a sector which is protected by the Block Protect (BP3, BP2, BP1, BP0) bits (see Table 3) or Boot Lock feature will be ignored.

The instruction sequence is shown in Figure 30.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

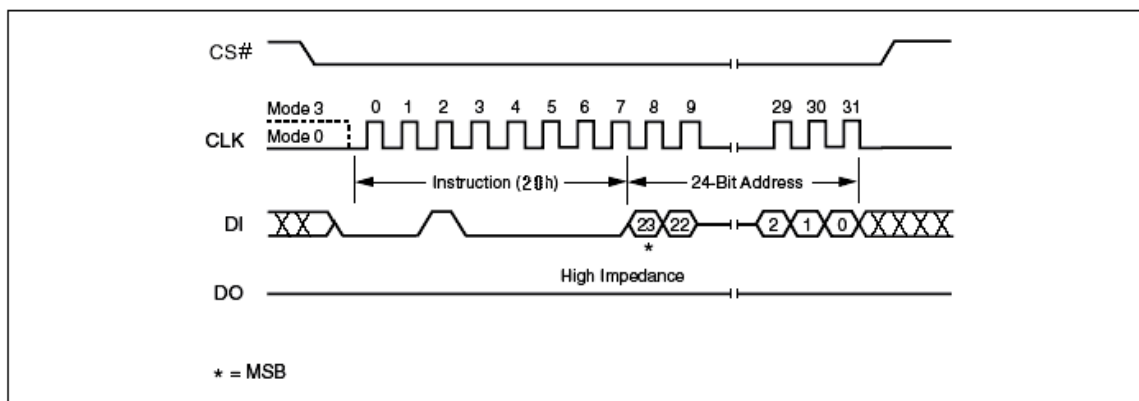


Figure 28. Sector Erase Instruction Sequence Diagram

32KB Half Block Erase (HBE) (52h)

The Half Block Erase (HBE) instruction sets to 1 (FFh) all bits inside the chosen block. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Half Block Erase (HBE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Block (see Table 2) is a valid address for the Half Block Erase (HBE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 29. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Half Block Erase (HBE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Block Erase cycle (whose duration is t_{HBE}) is initiated. While the Half Block Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Half Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Half Block Erase (HBE) instruction applied to a block which is protected by the Block Protect (BP3, BP2, BP1, BP0) bits (see Table 3) or Boot Lock feature will be ignored.

The instruction sequence is shown in Figure 30.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

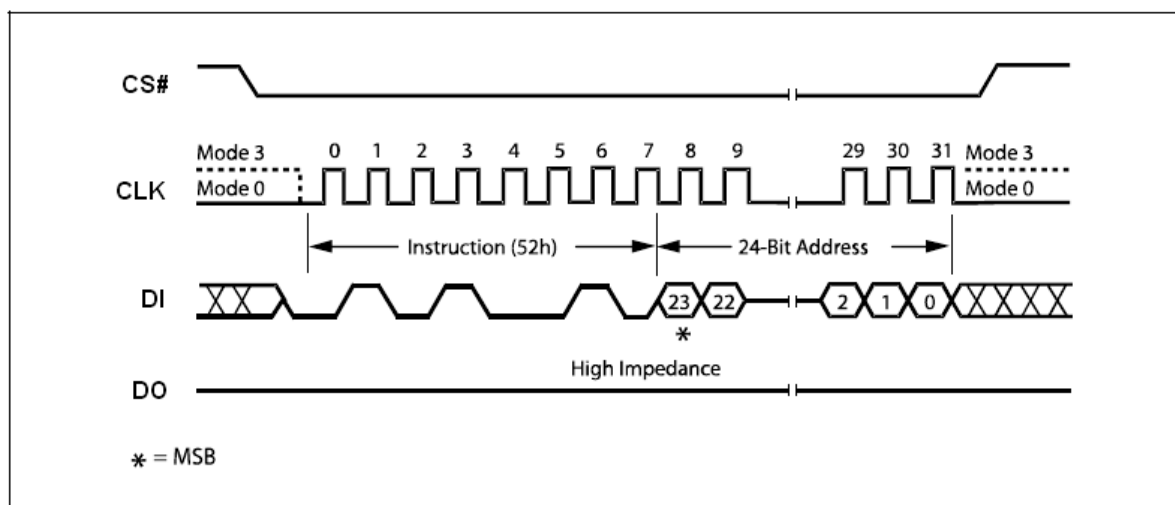


Figure 29. 32KB Half Block Erase Instruction Sequence Diagram

64K Block Erase (BE) (D8h)

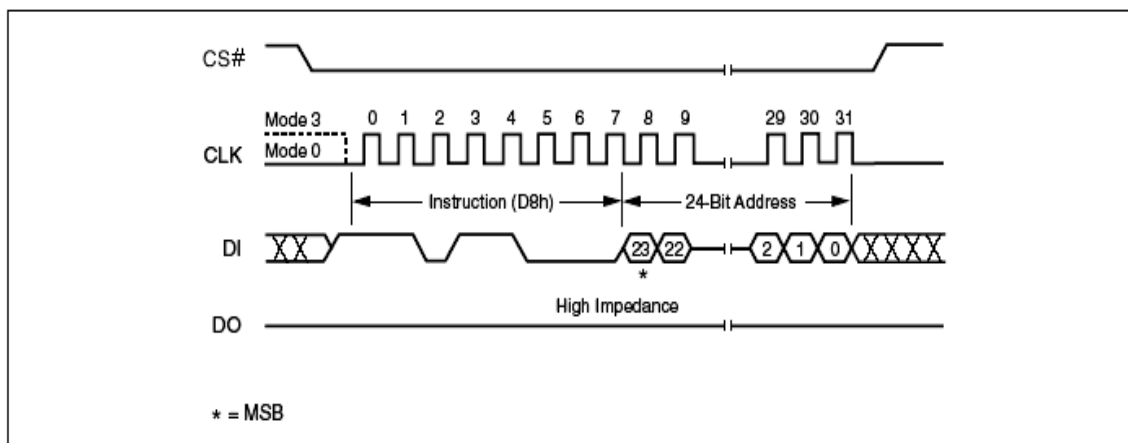
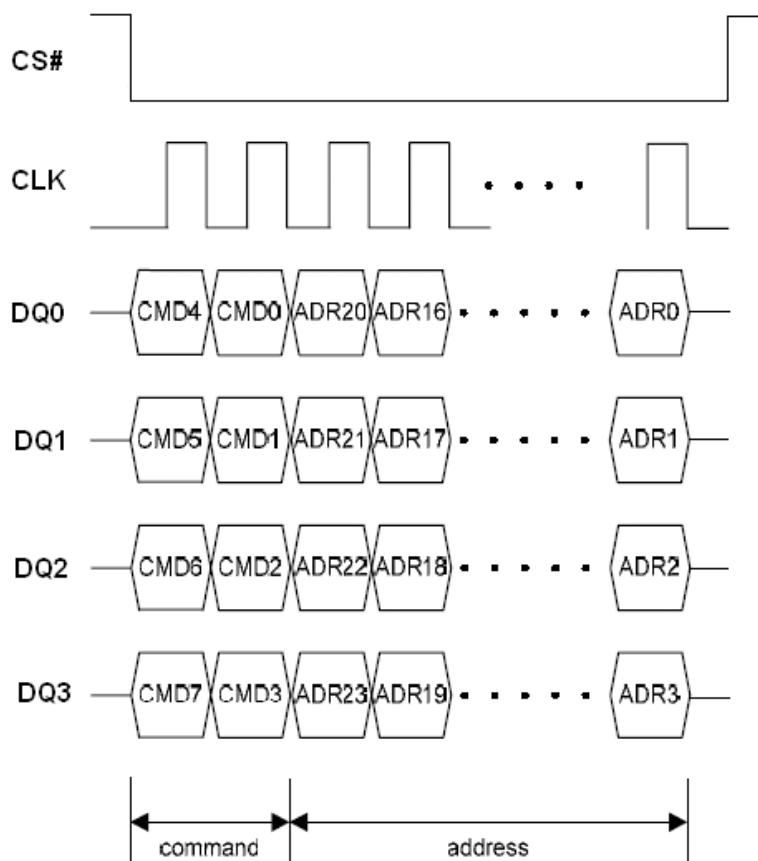
The Block Erase (BE) instruction sets to 1 (FFh) all bits inside the chosen block. Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Block Erase (BE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code, and three address bytes on Serial Data Input (DI). Any address inside the Block (see Table 2) is a valid address for the Block Erase (BE) instruction. Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 30. Chip Select (CS#) must be driven High after the eighth bit of the last address byte has been latched in, otherwise the Block Erase (BE) instruction is not executed. As soon as Chip Select (CS#) is driven High, the self-timed Block Erase cycle (whose duration is t_{BE}) is initiated. While the Block Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Block Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

A Block Erase (BE) instruction applied to a block which is protected by the Block Protect (BP3, BP2, BP1, BP0) bits (see Table 3) or Boot Lock feature will be ignored.

The instruction sequence is shown in Figure 30.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 30. 64K Block Erase Instruction Sequence Diagram

Figure 30.1 Block/Sector Erase Instruction Sequence in QPI Mode
Chip Erase (CE) (C7h/60h)

The Chip Erase (CE) instruction sets all bits to 1 (FFh). Before it can be accepted, a Write Enable (WREN) instruction must previously have been executed. After the Write Enable (WREN) instruction has been decoded, the device sets the Write Enable Latch (WEL).

The Chip Erase (CE) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code on Serial Data Input (DI). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 31. Chip Select (CS#) must be driven High after the eighth bit of the instruction code has been latched in, otherwise the Chip Erase instruction is not executed. As



soon as Chip Select (CS#) is driven High, the self-timed Chip Erase cycle (whose duration is t_{CE}) is initiated. While the Chip Erase cycle is in progress, the Status Register may be read to check the value of the Write In Progress (WIP) bit. The Write In Progress (WIP) bit is 1 during the self-timed Chip Erase cycle, and is 0 when it is completed. At some unspecified time before the cycle is completed, the Write Enable Latch (WEL) bit is reset.

The Chip Erase (CE) instruction is executed only if all Block Protect (BP3, BP2, BP1, BP0) bits are 0 and EBL bit is 0. The Chip Erase (CE) instruction is ignored if one or more blocks are protected.

The instruction sequence is shown in Figure 31.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

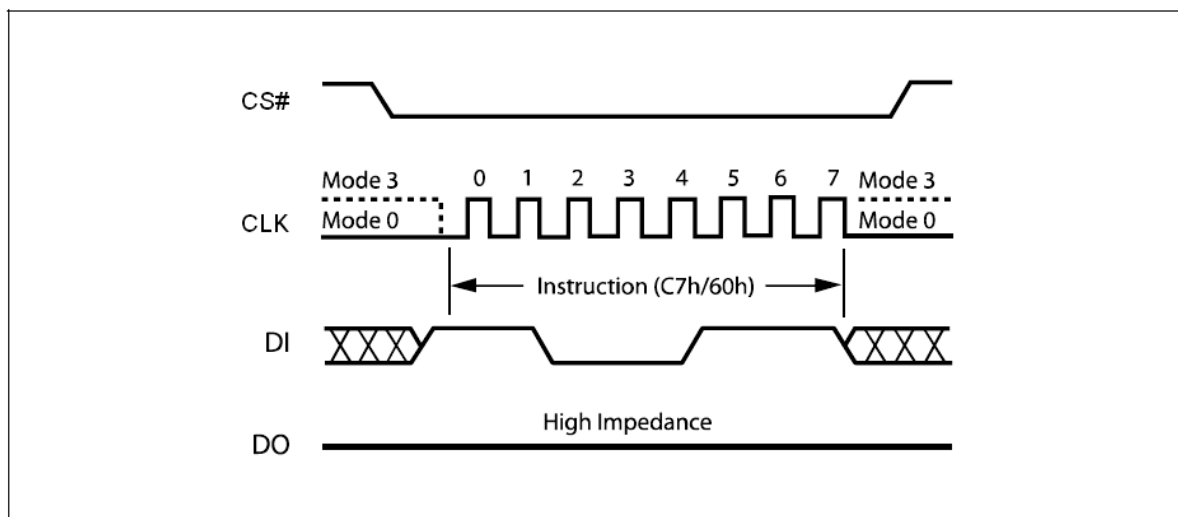


Figure 31. Chip Erase Instruction Sequence Diagram

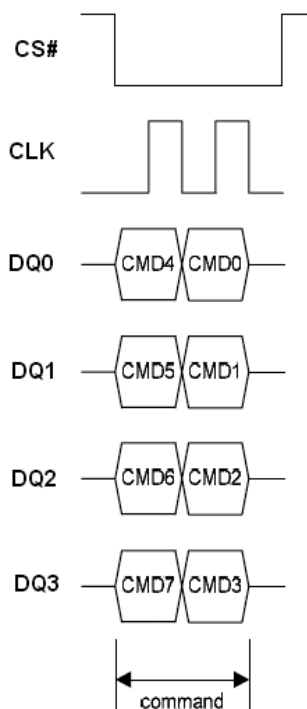


Figure 31.1 Chip Erase Sequence in QPI Mode



Deep Power-down (DP) (B9h)

Executing the Deep Power-down (DP) instruction is the only way to put the device in the lowest consumption mode (the Deep Power-down mode). It can also be used as an extra software protection mechanism, while the device is not in active use, since in this mode, the device ignores all Write, Program and Erase instructions.

Driving Chip Select (CS#) High deselects the device, and puts the device in the Standby mode (if there is no internal cycle currently in progress). But this mode is not the Deep Power-down mode. The Deep Power-down mode can only be entered by executing the Deep Power-down (DP) instruction, to reduce the standby current (from I_{CC1} to I_{CC2} , as specified in Table 16.)

Once the device has entered the Deep Power-down mode, all instructions are ignored except the Release from Deep Power-down, Read Device ID (RDI) and Software Reset instruction which release the device from this mode. The Release from Deep Power-down and Read Device ID (RDI) instruction also allows the Device ID of the device to be output on Serial Data Output (DO).

The Deep Power-down mode automatically stops at Power-down, and the device always Powers-up in the Standby mode. The Deep Power-down (DP) instruction is entered by driving Chip Select (CS#) Low, followed by the instruction code on Serial Data Input (DI). Chip Select (CS#) must be driven Low for the entire duration of the sequence.

The instruction sequence is shown in Figure 32. Chip Select (CS#) must be driven High after the eighth bit of the instruction code has been latched in, otherwise the Deep Power-down (DP) instruction is not executed. As soon as Chip Select (CS#) is driven High, it requires a delay of t_{DP} before the supply current is reduced to I_{CC2} and the Deep Power-down mode is entered.

Any Deep Power-down (DP) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

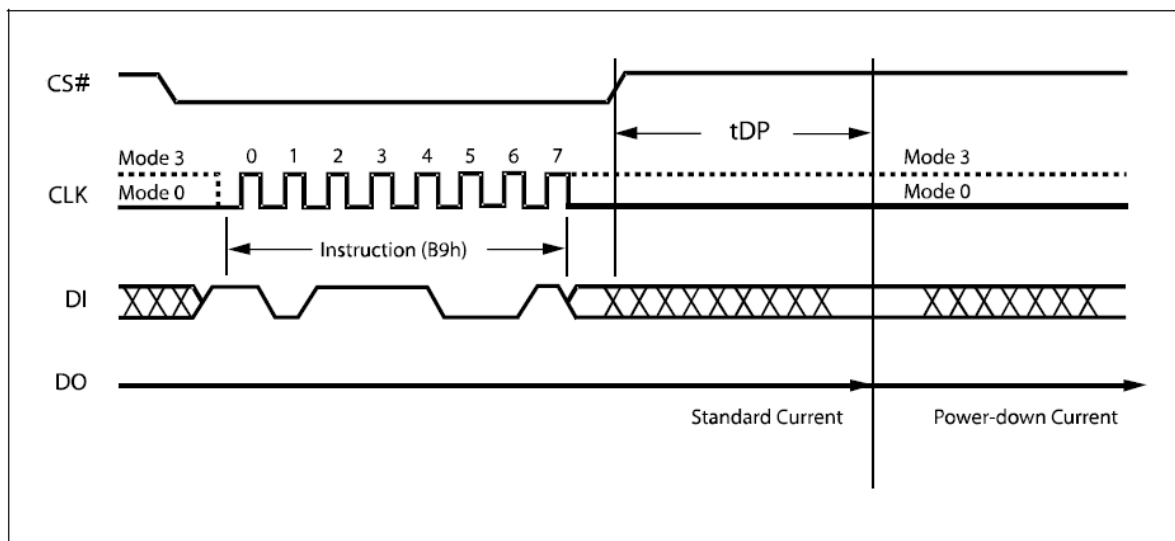


Figure 32. Deep Power-down Instruction Sequence Diagram

Release from Deep Power-down and Read Device ID (RDI)

Once the device has entered the Deep Power-down mode, all instructions are ignored except the Release from Deep Power-down and Read Device ID (RDI) instruction. Executing this instruction takes the device out of the Deep Power-down mode.

Please note that this is not the same as, or even a subset of, the JEDEC 16-bit Electronic Signature that is read by the Read Identifier (RDID) instruction. The old-style Electronic Signature is supported for reasons of backward compatibility, only, and should not be used for new designs. New designs should, instead, make use of the JEDEC 16-bit Electronic Signature, and the Read Identifier (RDID) instruction.

When used only to release the device from the power-down state, the instruction is issued by driving the CS# pin low, shifting the instruction code “ABh” and driving CS# high as shown in Figure 33. After the time duration of t_{RES1} (See AC Characteristics) the device will resume normal operation and other instructions will be accepted. The CS# pin must remain high during the t_{RES1} time duration.



When used only to obtain the Device ID while not in the power-down state, the instruction is initiated by driving the CS# pin low and shifting the instruction code “ABh” followed by 3-dummy bytes. The Device ID bits are then shifted out on the falling edge of CLK with most significant bit (MSB) first as shown in Figure 34. The Device ID value for the EN25QH128A(2T) are listed in Table 6. The Device ID can be read continuously. The instruction is completed by driving CS# high.

When Chip Select (CS#) is driven High, the device is put in the Stand-by Power mode. If the device was not previously in the Deep Power-down mode, the transition to the Stand-by Power mode is immediate. If the device was previously in the Deep Power-down mode, though, the transition to the Standby Power mode is delayed by t_{RES2} , and Chip Select (CS#) must remain High for at least t_{RES2} (max), as specified in Table 18. Once in the Stand-by Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

Except while an Erase, Program or Write Status Register cycle is in progress, the Release from Deep Power-down and Read Device ID (RDI) instruction always provides access to the 8bit Device ID of the device, and can be applied even if the Deep Power-down mode has not been entered.

Any Release from Deep Power-down and Read Device ID (RDI) instruction while an Erase, Program or Write Status Register cycle is in progress, is not decoded, and has no effect on the cycle that is in progress.

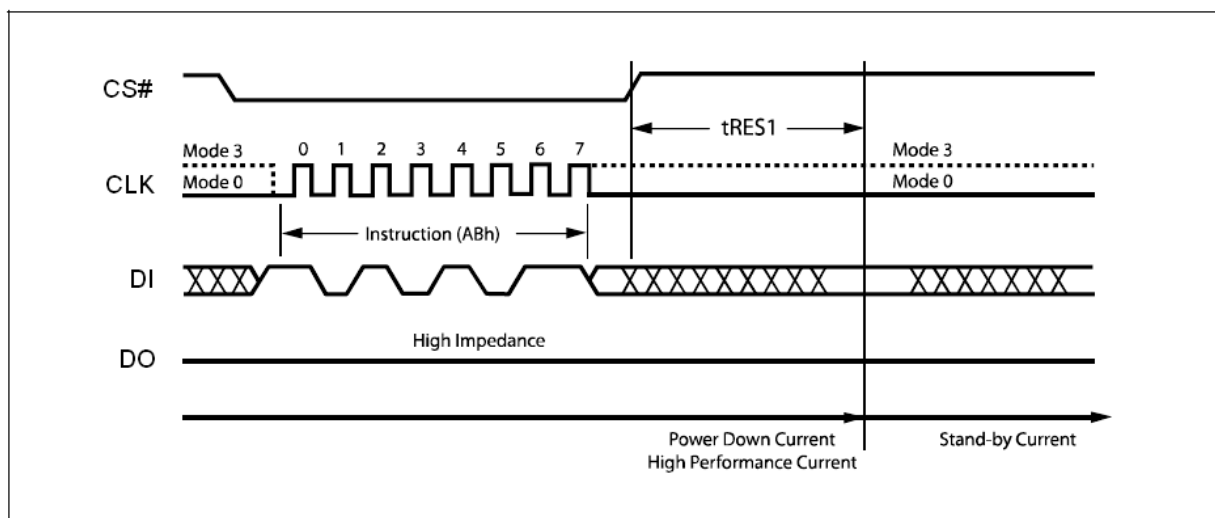


Figure 33. Release Power-down Instruction Sequence Diagram

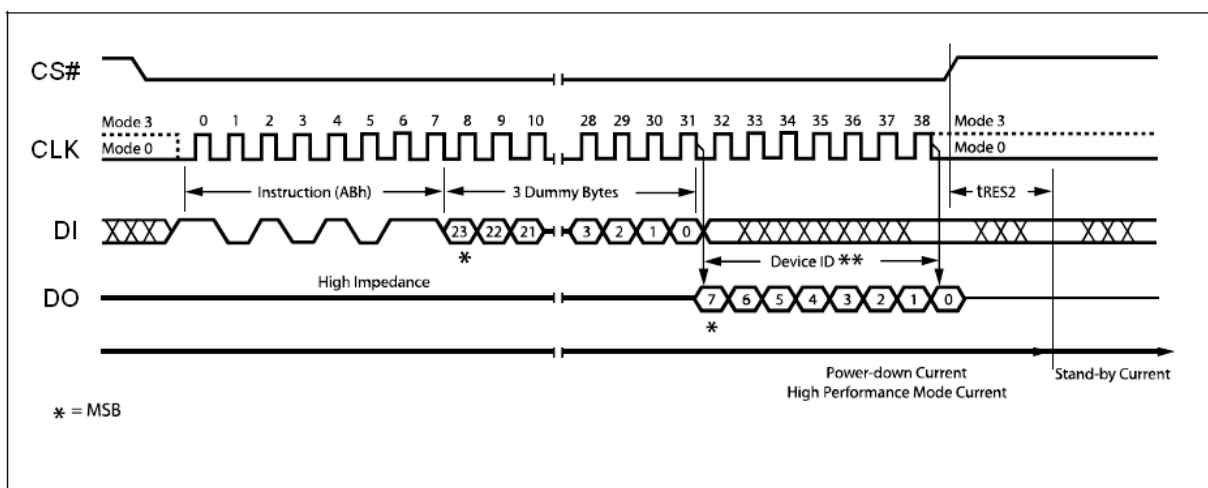


Figure 34. Release Power-down / Device ID Instruction Sequence Diagram



Read Manufacturer / Device ID (90h)

The Read Manufacturer/Device ID instruction is an alternative to the Release from Power-down / Device ID instruction that provides both the JEDEC assigned manufacturer ID and the specific device ID.

The Read Manufacturer/Device ID instruction is very similar to the Release from Power-down / Device ID instruction. The instruction is initiated by driving the CS# pin low and shifting the instruction code “90h” followed by a 24-bit address (A23-A0) of 000000h. After which, the Manufacturer ID for Eon (1Ch) and the Device ID are shifted out on the falling edge of CLK with most significant bit (MSB) first as shown in Figure 35. The Device ID values for the EN25QH128A(2T) are listed in Table 6. If the 24-bit address is initially set to 000001h the Device ID will be read first

The instruction sequence is shown in Figure 35.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

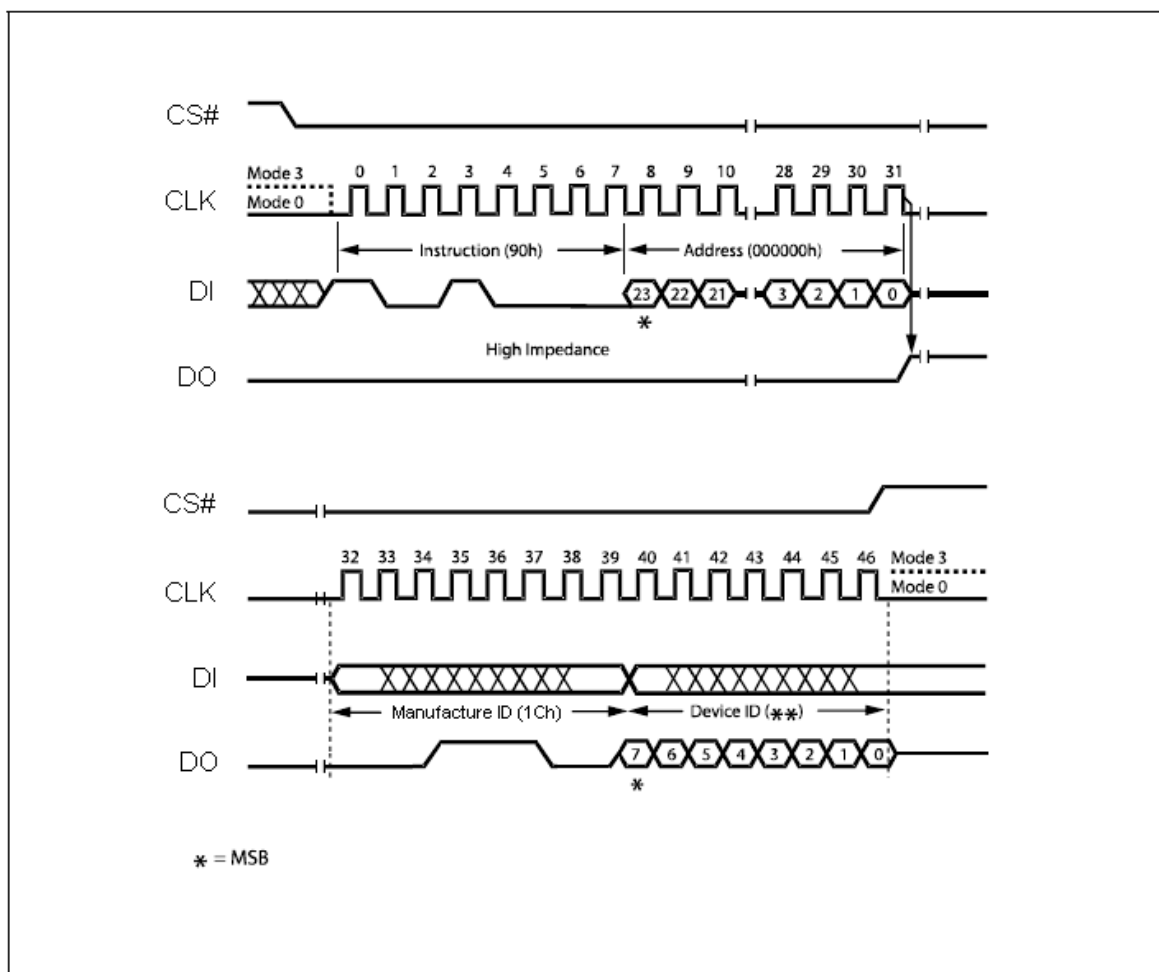


Figure 35. Read Manufacturer / Device ID Diagram

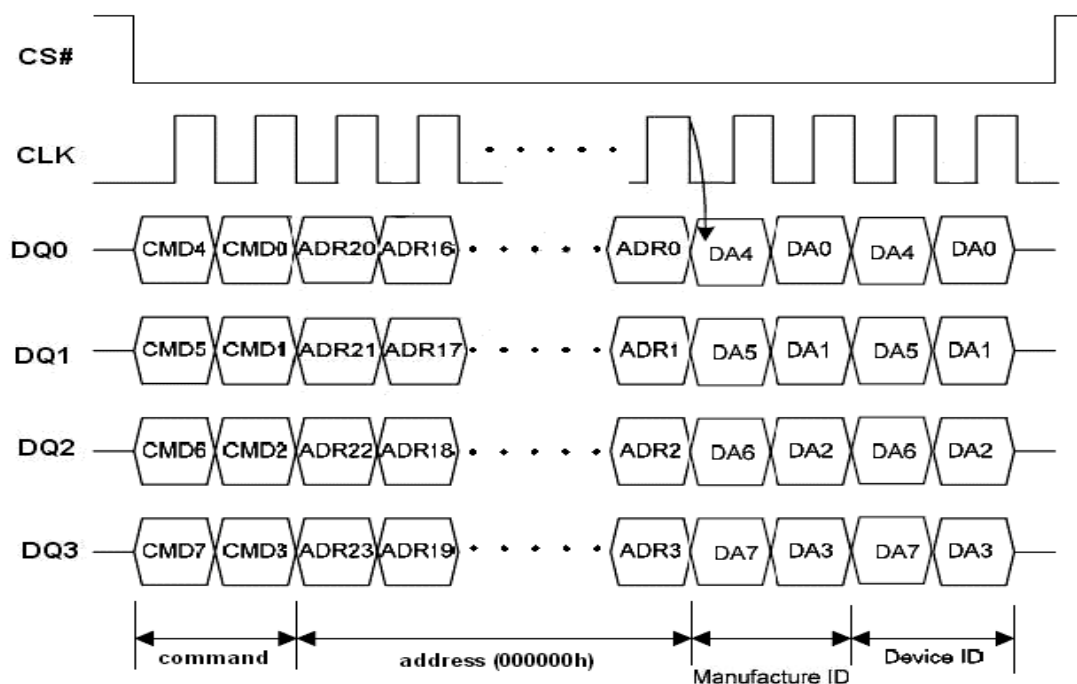


Figure 35.1. Read Manufacturer / Device ID Diagram in QPI Mode

Read Identification (RDID) (9Fh)

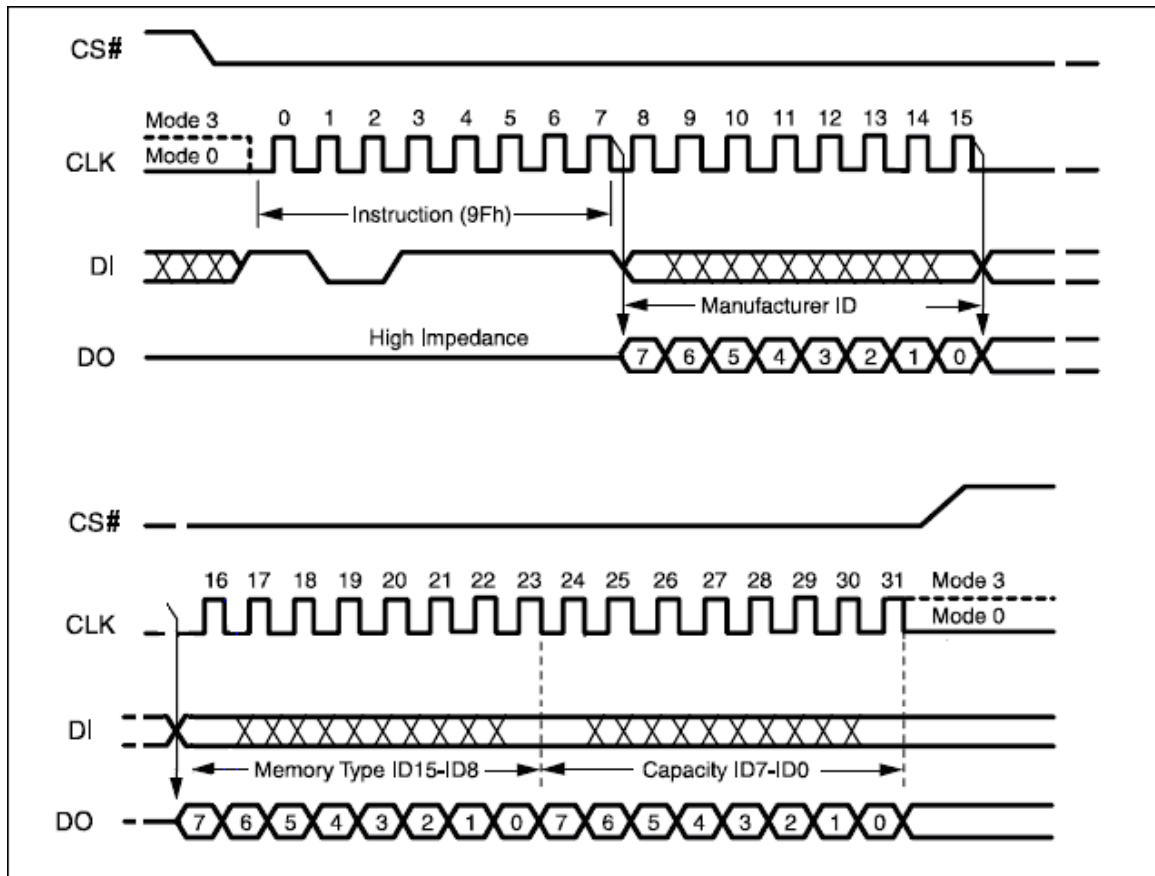
The Read Identification (RDID) instruction allows the 8-bit manufacturer identification to be read, followed by two bytes of device identification. The device identification indicates the memory type in the first byte, and the memory capacity of the device in the second byte.

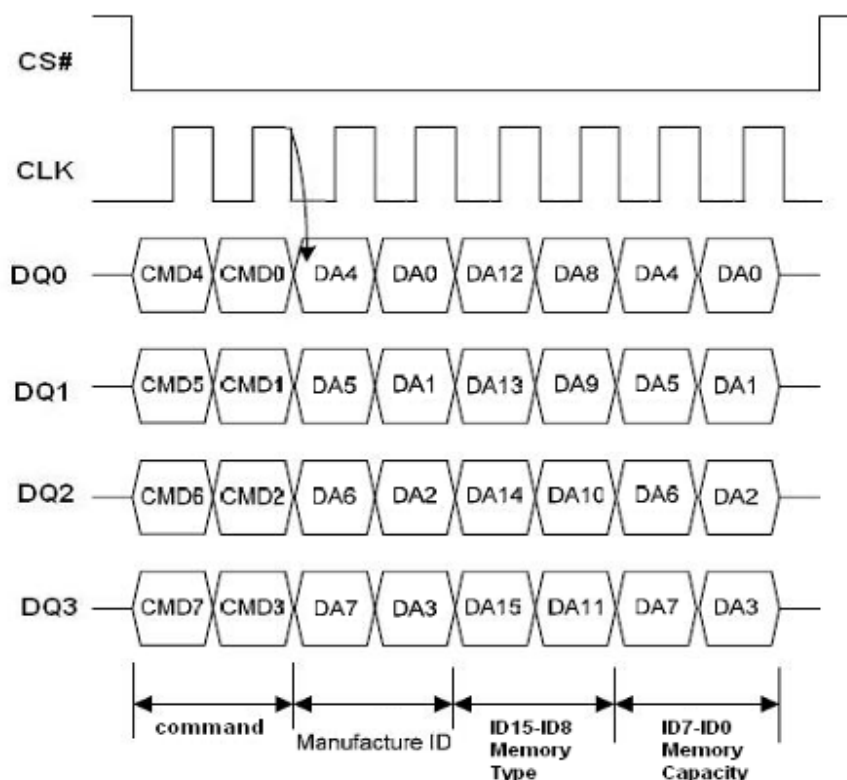
Any Read Identification (RDID) instruction while an Erase or Program cycle is in progress, is not decoded, and has no effect on the cycle that is in progress. The Read Identification (RDID) instruction should not be issued while the device is in Deep Power down mode.

The device is first selected by driving Chip Select Low. Then, the 8-bit instruction code for the instruction is shifted in. This is followed by the 24-bit device identification, stored in the memory, being shifted out on Serial Data Output, each bit being shifted out during the falling edge of Serial Clock. The instruction sequence is shown in Figure 36. The Read Identification (RDID) instruction is terminated by driving Chip Select High at any time during data output.

When Chip Select is driven High, the device is put in the Standby Power mode. Once in the Standby Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

The instruction sequence is shown in Figure 36.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 36. Read Identification (RDID)


Figure 36.1. Read Identification (RDID) in QPI Mode
Enter OTP Mode (3Ah)

This Flash support OTP mode to enhance the data protection, user can use the Enter OTP mode (3Ah) command for entering this mode. In OTP mode, the Status Register S7 bit is served as OTP_LOCK bit, S6 bit is served as WXDIS bit, S5 bit is served as HRSW bit, S4 bit is served as 4KBL bit, S3 bit is served as TB bit, S1 bit is served as WEL bit and S0 bit is served as WIP bit. They can be read by RDSR command.

This Flash has an extra 512 bytes OTP sector, user must issue ENTER OTP MODE command to read, program or erase OTP sector. After entering OTP mode, the OTP sector is mapping to sector 4095, **SRP bit** becomes OTP_LOCK bit. The Chip Erase, Block Erase and Half Block Erase commands are also disabled.

In OTP mode, user can read other sectors, but program/erase other sectors only allowed when they are not protected by Block Protect (BP3, BP2, BP1, BP0) bits and Block Lock feature. The OTP sector can **only** be erased by Sector Erase (20h) command. The Chip Erase (C7h/ 60h), 64K Block Erase (D8h) and 32K Half Block Erase (52h) commands are disable in OTP mode.

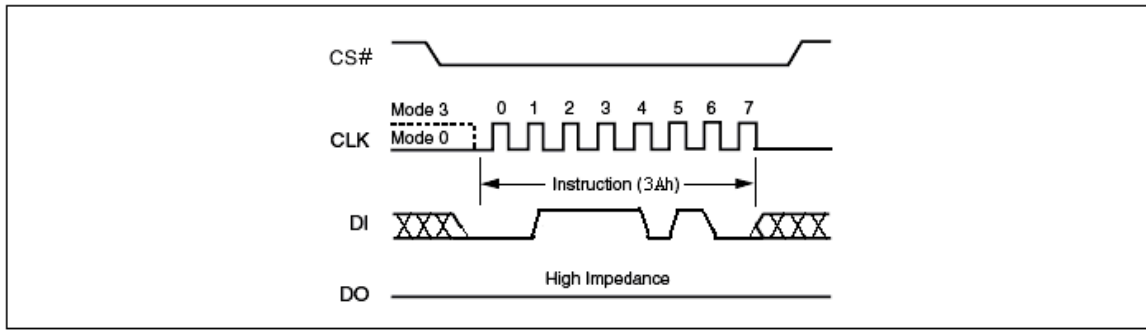
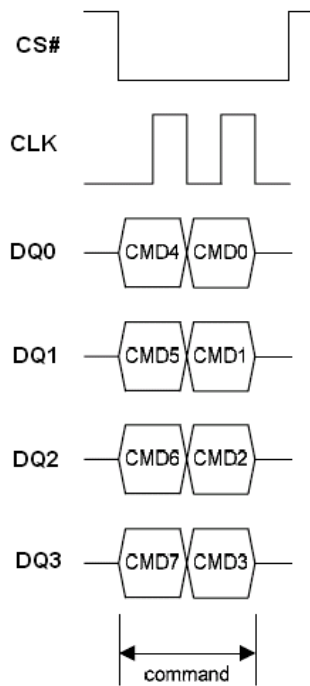
Table 11. OTP Sector Address

Sector	Sector Size	Address Range
4095	512 byte	FFF000h – FFF1FFh

Note: The OTP sector is mapping to sector 4095

WRSR command is used to program OTP_LOCK bit, TB bit, 4KBL bit to '1', but these bits only can be programmed once. User can use WRDI (04h) command to exit OTP mode.

The instruction sequence is shown in Figure 37.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.


Figure 37. Enter OTP Mode

Figure 37.1 Enter OTP Mode Sequence in QPI Mode



Read SFDP Mode and Unique ID Number (5Ah)

Read SFDP Mode

EN25QH128A(2T) features Serial Flash Discoverable Parameters (SFDP) mode. Host system can retrieve the operating characteristics, structure and vendor specified information such as identifying information, memory size, operating voltage and timing information of this device by SFDP mode.

The device is first selected by driving Chip Select (CS#) Low. The instruction code for the Read SFDP Mode is followed by a 3-byte address (A23-A0) and a dummy byte, each bit being latched-in during the rising edge of Serial Clock (CLK). Then the memory contents, at that address, is shifted out on Serial Data Output (DO), each bit being shifted out, at a maximum frequency F_R , during the falling edge of Serial Clock (CLK).

The instruction sequence is shown in Figure 38. The first byte addressed can be at any location. The address is automatically incremented to the next higher address after each byte of data is shifted out. The whole memory can, therefore, be read with a single Serial Flash Discoverable Parameters (SFDP) instruction. When the highest address is reached, the address counter rolls over to 0x00h, allowing the read sequence to be continued indefinitely. The Serial Flash Discoverable Parameters (SFDP) instruction is terminated by driving Chip Select (CS#) High. Chip Select (CS#) can be driven High at any time during data output. Any Read Data Bytes at Serial Flash Discoverable Parameters (SFDP) instruction, while an Erase, Program or Write cycle is in progress, is rejected without having any effects on the cycle that is in progress.

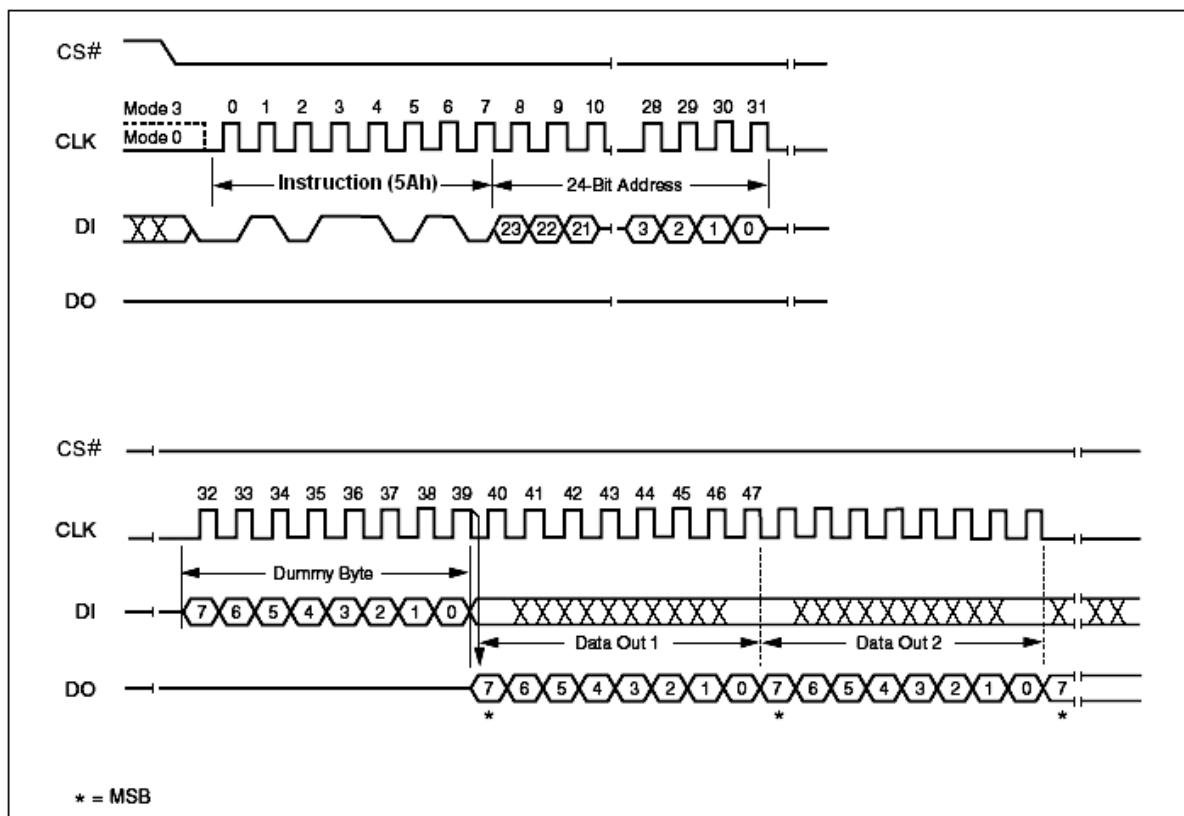


Figure 38. Read SFDP Mode Instruction Sequence Diagram

**Table 12. Serial Flash Discoverable Parameters (SFDP) Signature and Parameter Identification Data Value (Advanced Information)**

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
SFDP Signature	00h	07 : 00	53h	Signature [31:0]: Hex: 50444653
	01h	15 : 08	46h	
	02h	23 : 16	44h	
	03h	31 : 24	50h	
SFDP Minor Revision Number	04h	07 : 00	00h	Star from 0x00
SFDP Major Revision Number	05h	15 : 08	01h	Star from 0x01
Number of Parameter Headers (NPH)	06h	23 : 16	00h	1 parameter header
Unused	07h	31 : 24	FFh	Reserved
ID Number	08h	07 : 00	00h	JEDEC ID
Parameter Table Minor Revision Number	09h	15 : 08	00h	Star from 0x00
Parameter Table Major Revision Number	0Ah	23 : 16	01h	Star from 0x01
Parameter Table Length (in DW)	0Bh	31 : 24	09h	9 DWORds
Parameter Table Pointer (PTP)	0Ch	07 : 00	30h	000030h
	0Dh	15 : 08	00h	
	0Eh	23 : 16	00h	
Unused	0Fh	31 : 24	FFh	Reserved



Table 13. Parameter ID (0) (Advanced Information) 1/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment	
Block / Sector Erase sizes Identifies the erase granularity for all Flash Components	30h	00	01b	00 = reserved 01 = 4KB erase 10 = reserved 11 = 64KB erase	
		01			
Write Granularity		02	1b	0 = No, 1 = Yes	
Write Enable Instruction Required for Writing to Volatile Status Register		03	01b	00 = N/A 01 = use 50h opcode 11 = use 06h opcode	
Write Enable Opcode Select for Writing to Volatile Status Register		04			
Unused		05	111b	Reserved	
		06			
		07			
4 Kilo-Byte Erase Opcode		31h	08	20h	4 KB Erase Support (FFh = not supported)
			09		
	10				
	11				
	12				
	13				
	14				
Supports (1-1-2) Fast Read Device supports single input opcode & address and dual output data Fast Read	32h	15	1b	0 = not supported 1 = supported	
		16			
Address Byte Number of bytes used in addressing for flash array read, write and erase.		17	00b	00 = 3-Byte 01 = 3- or 4-Byte (e.g. defaults to 3-Byte mode; enters 4-Byte mode on command) 10 = 4-Byte 11 = reserved	
		18			
Supports Double Data Rate (DDR) Clocking Indicates the device supports some type of double transfer rate clocking.		19	0b	0 = not supported 1 = supported	
Supports (1-2-2) Fast Read Device supports single input opcode, dual input address, and dual output data Fast Read		20	1b	0 = not supported 1 = supported	
Supports (1-4-4) Fast Read Device supports single input opcode, quad input address, and quad output data Fast Read		21	1b	0 = not supported 1 = supported	
Supports (1-1-4) Fast Read Device supports single input opcode & address and quad output data Fast Read		22	0b	0 = not supported 1 = supported	
Unused		23	1b	Reserved	
Unused		33h	24	FFh	Reserved
	25				
	26				
	27				
	28				
	29				
	30				
31					



Table 13. Parameter ID (0) (Advanced Information) 2/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Flash Memory Density	37h : 34h	31 : 00	7FFFFFFh	128 Mbits

Table 13. Parameter ID (0) (Advanced Information) 3/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
(1-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	38h	00	1Fh	Configurable
		01		
		02		
		03		
		04		
Quad Input Address Quad Output (1-4-4) Fast Read Number of Mode Bits	38h	05	010b	8 mode bits
		06		
		07		
(1-4-4) Fast Read Opcode Opcode for single input opcode, quad input address, and quad output data Fast Read.	39h	08	EBh	
		09		
		10		
		11		
		12		
		13		
		14		
(1-1-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ah	16	00000b	Not supported
		17		
		18		
		19		
		20		
(1-1-4) Fast Read Number of Mode Bits	3Ah	21	000b	Not supported
		22		
		23		
(1-1-4) Fast Read Opcode Opcode for single input opcode & address and quad output data Fast Read.	3Bh	31 : 24	6Bh	



Table 13. Parameter ID (0) (Advanced Information) 4/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
(1-1-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ch	00	01000b	8 dummy clocks
		01		
		02		
		03		
		04		
(1-1-2) Fast Read Number of Mode Bits	3Ch	05	000b	Not supported
		06		
		07		
(1-1-2) Fast Read Opcode Opcode for single input opcode & address and dual output data Fast Read.	3Dh	15 : 08	3Bh	Not supported
(1-2-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Eh	16	00100b	4 dummy clocks
		17		
		18		
		19		
		20		
(1-2-2) Fast Read Number of Mode Bits	3Eh	21	000b	Not supported
		22		
		23		
(1-2-2) Fast Read Opcode Opcode for single input opcode, dual input address, and dual output data Fast Read.	3Fh	31 : 24	BBh	Not supported

Table 13. Parameter ID (0) (Advanced Information) 5/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment	
Supports (2-2-2) Fast Read Device supports dual input opcode & address and dual output data Fast Read.	40h	00	0b	0 = not supported 1 = supported	
Reserved. These bits default to all 1's		01	111b	Reserved	
		02			
		03			
Supports (4-4-4) Fast Read Device supports Quad input opcode & address and quad output data Fast Read.		40h	04	1b	0 = not supported 1 = supported (EQPI Mode)
Reserved. These bits default to all 1's			05	111b	Reserved
			06		
	07				
Reserved. These bits default to all 1's	43h : 41h	31 : 08	FFh	Reserved	



Table 13. Parameter ID (0) (Advanced Information) 6/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Reserved. These bits default to all 1's	45h : 44h	15 : 00	FFh	Reserved
(2-2-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	46h	16	00000b	Not supported
		17		
		18		
		19		
(2-2-2) Fast Read Number of Mode Bits	46h	20	000b	Not supported
		21		
		22		
(2-2-2) Fast Read Opcode Opcode for dual input opcode & address and dual output data Fast Read.	47h	23	FFh	Not supported
		31 : 24		

Table 13. Parameter ID (0) (Advanced Information) 7/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Reserved. These bits default to all 1's	49h : 48h	15 : 00	FFh	Reserved
(4-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	4Ah	16	1Fh	Configurable
		17		
		18		
		19		
(4-4-4) Fast Read Number of Mode Bits	4Ah	20	010b	8 mode bits
		21		
		22		
(4-4-4) Fast Read Opcode Opcode for quad input opcode/address, quad output data Fast Read.	4Bh	23	EBh	Must Enter EQPI Mode firstly
		31 : 24		

Table 13. Parameter ID (0) (Advanced Information) 8/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Sector Type 1 Size	4Ch	07 : 00	0Ch	4 KB
Sector Type 1 Opcode	4Dh	15 : 08	20h	
Sector Type 2 Size	4Eh	23 : 16	0Fh	32 KB
Sector Type 2 Opcode	4Fh	31 : 24	52h	

Table 14. Parameter ID (0) (Advanced Information) 9/9

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Sector Type 3 Size	50h	07 : 00	10h	64 KB
Sector Type 3 Opcode	51h	15 : 08	D8h	
Sector Type 4 Size	52h	23 : 16	00h	Not Supported
Sector Type 4 Opcode	53h	31 : 24	FFh	Not Supported

**Read Unique ID Number**

The Read Unique ID Number instruction accesses a factory-set read-only 96-bit number that is unique to each EN25QH128A(2T) device. The ID number can be used in conjunction with user software methods to help prevent copying or cloning of a system. The Read Unique ID instruction is initiated by driving the CS# pin low and shifting the instruction code “5Ah” followed by a three bytes of addresses, 0x80h, and one byte of dummy clocks. After which, the 96-bit ID is shifted out on the falling edge of CLK as shown in Figure 38.

Table 14. Unique ID Number

Description	Address (h) (Byte Mode)	Address (Bit)	Data	Comment
Unique ID Number	80h : 8Bh	95 : 00	By die	



Power-up Timing

All functionalities and DC specifications are specified for a Vcc ramp rate of greater than 1V per 100 ms (0V to 2.7V in less than 270 ms). See Table 15 and Figure 39 for more information.

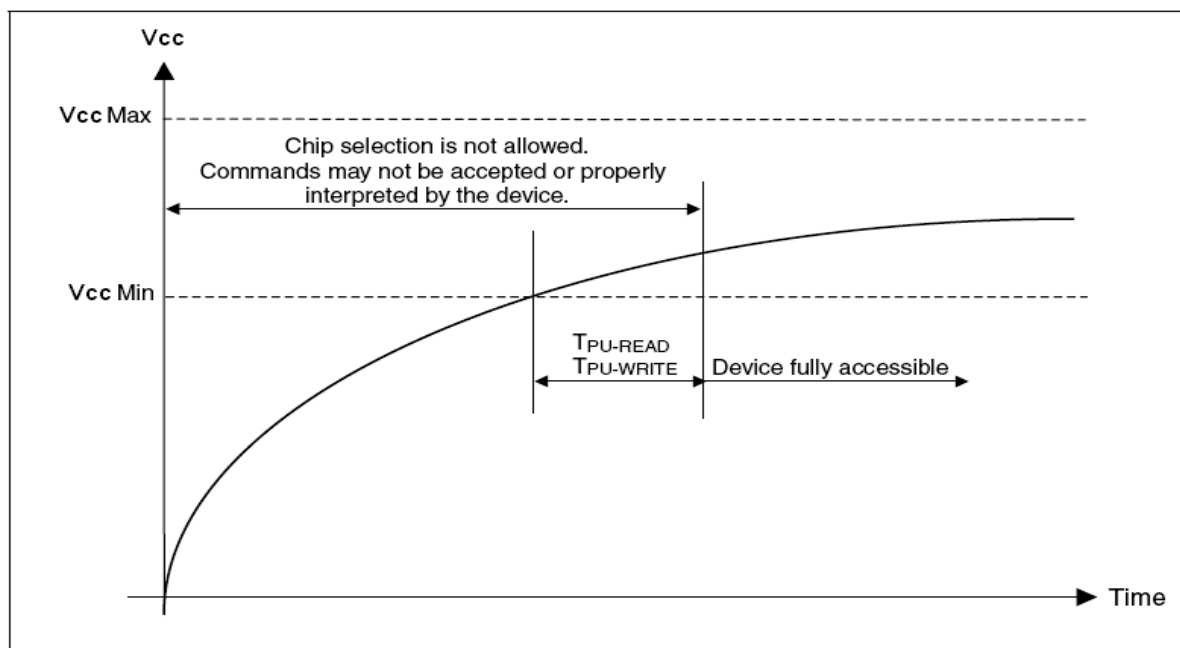


Figure 39. Power-up Timing

Table 15. Power-Up Timing

Symbol	Parameter	Min.	Unit
T _{PU-READ} ⁽¹⁾	V _{CC} Min to Read Operation	100	μs
T _{PU-WRITE} ⁽¹⁾	V _{CC} Min to Write Operation	100	μs

Note:

- 1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

INITIAL DELIVERY STATE

The device is delivered with the memory array erased: all bits are set to 1 (each byte contains FFh). The Status Register contains 00h (all Status Register bits are 0).



Table 16. DC Characteristics

($T_a = -40^{\circ}\text{C}$ to 85°C ; $V_{CC} = 2.7\text{-}3.6\text{V}$)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{LI}	Input Leakage Current			1	± 2	μA
I_{LO}	Output Leakage Current			1	± 2	μA
I_{CC1}	Standby Current	$CS\# = V_{CC}, V_{IN} = V_{SS}$ or V_{CC}			20	μA
I_{CC2}	Deep Power-down Current	$CS\# = V_{CC}, V_{IN} = V_{SS}$ or V_{CC}			20	μA
I_{CC3}	Operating Current (READ)	CLK = $0.1 V_{CC} / 0.9 V_{CC}$ at 104MHz, DQ = open		10	25	mA
		CLK = $0.1 V_{CC} / 0.9 V_{CC}$ at 33MHz, DQ = open		5	12	mA
		CLK = $0.1 V_{CC} / 0.9 V_{CC}$ at 104MHz, Quad Output Read, DQ = open		14	35	mA
		CLK = $0.1 V_{CC} / 0.9 V_{CC}$ at 33MHz, Quad Output Read, DQ = open		7	17	mA
I_{CC4}	Operating Current (PP)	$CS\# = V_{CC}$		9	30	mA
I_{CC5}	Operating Current (WRSR)	$CS\# = V_{CC}$			25	mA
I_{CC6}	Operating Current (SE)	$CS\# = V_{CC}$		13	25	mA
I_{CC7}	Operating Current (BE)	$CS\# = V_{CC}$		15	25	mA
V_{IL}	Input Low Voltage		-0.5		$0.2 V_{CC}$	V
V_{IH}	Input High Voltage		$0.7V_{CC}$		$V_{CC}+0.4$	V
V_{OL}	Output Low Voltage	$I_{OL} = 100 \mu\text{A}, V_{CC}=V_{CC} \text{ Min.}$			0.3	V
V_{OH}	Output High Voltage	$I_{OH} = -100 \mu\text{A}, V_{CC}=V_{CC} \text{ Min.}$	$V_{CC}-0.2$			V

Table 17. AC Measurement Conditions

Symbol	Parameter	Min.	Max.	Unit
C_L	Load Capacitance	30		pF
	Input Rise and Fall Times		5	ns
	Input Pulse Voltages	$0.2V_{CC}$ to $0.8V_{CC}$		V
	Input Timing Reference Voltages	$0.3V_{CC}$ to $0.7V_{CC}$		V
	Output Timing Reference Voltages	$V_{CC} / 2$		V

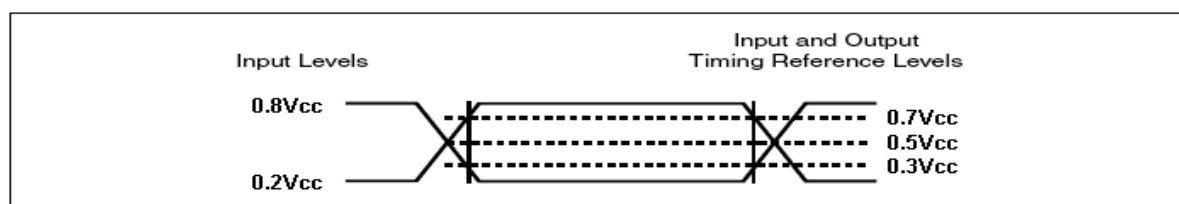


Figure 40. AC Measurement I/O Waveform



Table 18. AC Characteristics

(T_a = - 40°C to 85°C; V_{CC} = 2.7-3.6V)

Symbol	Alt	Parameter	Min	Typ	Max	Unit
F _R	f _C	Serial SPI Clock Frequency for: PP, QPP, SE, HBE, BE, CE, DP, RES, RDP, WREN, WRDI, WRSR, WRSR3, Fast Read	D.C.		104	MHz
		Serial SPI Clock Frequency for: RDSR, RDSR3, RDID,	D.C.		104	MHz
		Serial Dual/Quad Clock Frequency for: PP, QPP, SE, HBE, BE, CE, DP, RES, RDP, WREN, WRDI, WRSR, WRSR3, RDSR, RDSR3, RDID, Fast Read, Dual Output Fast Read, Dual I/O Fast Read, Quad I/O Fast Read	D.C.		104	MHz
f _R		Serial Clock Frequency for READ	D.C.		83	MHz
t _{CH} ¹		Serial Clock High Time	3.5			ns
t _{CL} ¹		Serial Clock Low Time	3.5			ns
t _{CLCH} ²		Serial Clock Rise Time (Slew Rate)	0.1			V / ns
t _{CHCL} ²		Serial Clock Fall Time (Slew Rate)	0.1			V / ns
t _{SLCH}	t _{CSS}	CS# Active Setup Time	5			ns
t _{CHSH}	t _{CSS}	CS# Active Hold Time	5			ns
t _{SHCH}	t _{CSS}	CS# Not Active Setup Time	5			ns
t _{CHSL}	t _{CSS}	CS# Not Active Hold Time	5			ns
t _{SHSL}	t _{CSH}	CS# High Time for read	30			ns
	t _{CSH}	CS# High Time for program/erase	30			ns
t _{SHSL2}	t _{CSH}	Volatile Register Write Time	50			ns
t _{SHQZ} ²	t _{DIS}	Output Disable Time			6	ns
t _{CLQX}	t _{HO}	Output Hold Time	0			ns
t _{DVCH}	t _{DSU}	Data In Setup Time	2			ns
t _{CHDX}	t _{DH}	Data In Hold Time	3			ns
t _{HLCH}		HOLD# Low Setup Time (relative to CLK)	5			ns
t _{HHCH}		HOLD# High Setup Time (relative to CLK)	5			ns
t _{CHHH}		HOLD# Low Hold Time (relative to CLK)	5			ns
t _{CHHL}		HOLD# High Hold Time (relative to CLK)	5			ns
t _{CLQV}	t _V	Output Valid from CLK			7	ns
t _{WHSL} ³		Write Protect Setup Time before CS# Low	20			ns
t _{SHWL} ³		Write Protect Hold Time after CS# High	100			ns
t _{DP} ²		CS# High to Deep Power-down Mode			3	µs
t _{RES1} ²		CS# High to Standby Mode without Electronic Signature read			3	µs
t _{RES2} ²		CS# High to Standby Mode with Electronic Signature read			1.8	µs
t _W		Write Status Register Cycle Time		10	50	ms
t _{PP}		Page Programming Time		0.5	3	ms
t _{SE}		Sector Erase Time		0.04	0.3	s
t _{HBE}		Half Block Erase Time		0.2	1	s
t _{BE}		Block Erase Time		0.3	2	s
t _{CE}		Chip Erase Time		60	200	s
t _{HRST}		RESET# low period to reset the device	1			µs
t _{HRSL}		RESET# high to next instruction	28			µs



t_{SHRV}		Deselect to RESET# valid in quad mode	8		ns
	t_{SR}	Software Reset Latency	WIP = write operation		28 μ s
			WIP = not in write operation		0 μ s

- Note:**
- $t_{CH} + t_{CL}$ must be greater than or equal to $1/f_C$
 - Value guaranteed by characterization, not 100% tested in production.
 - Only applicable as a constraint for a Write status Register instruction when Status Register Protect Bit is set at 1.4.

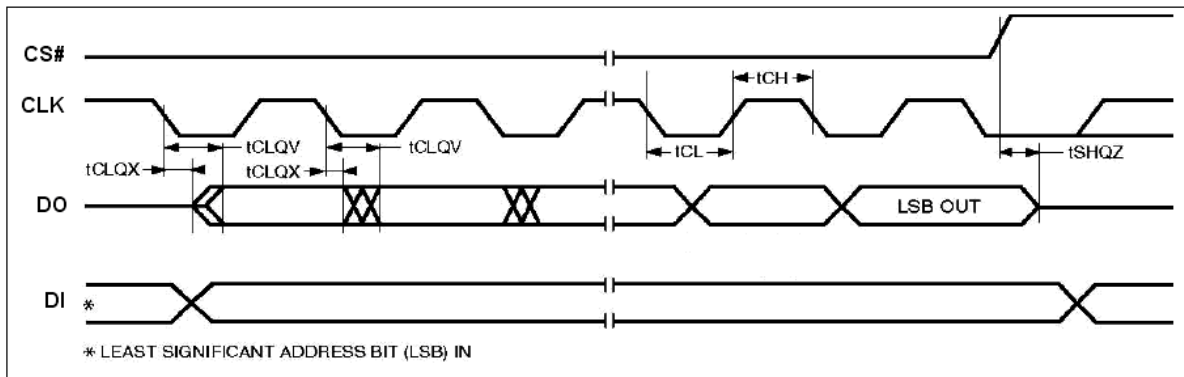


Figure 41. Serial Output Timing

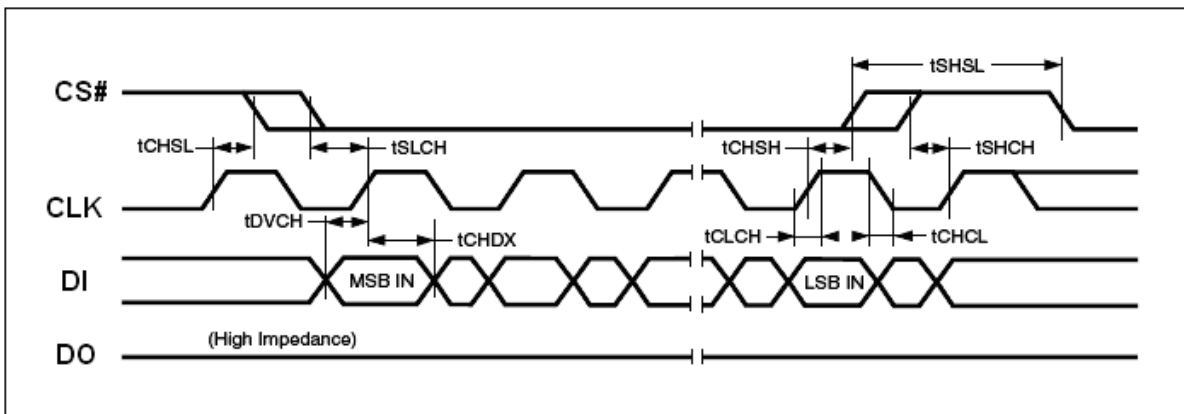


Figure 42. Input Timing

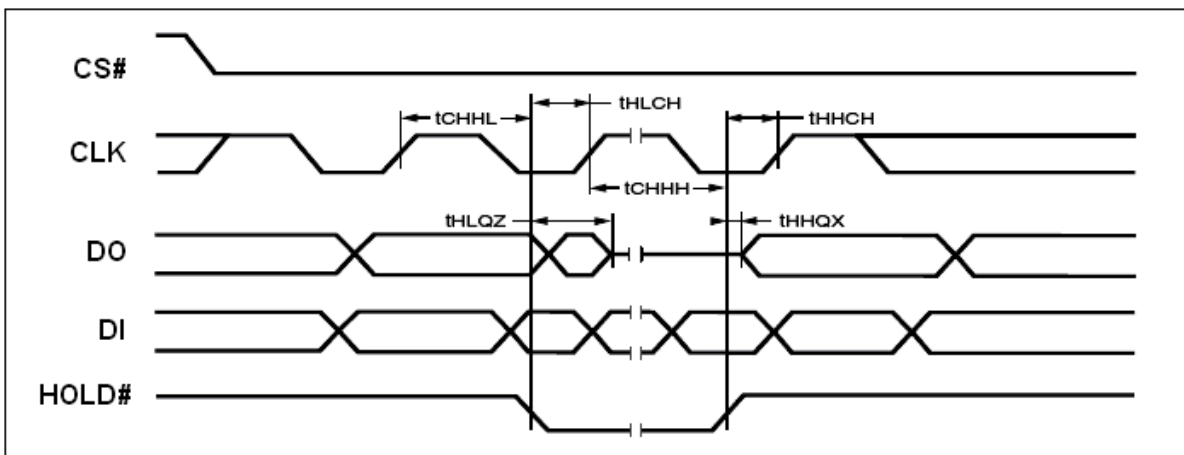
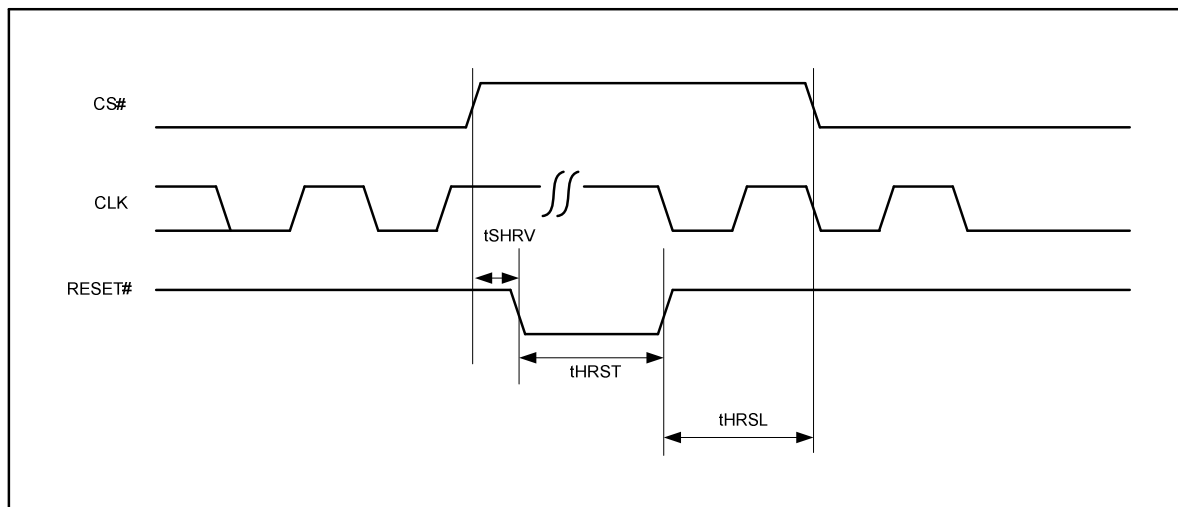


Figure 43. Hold Timing


Figure 44. Reset Timing

ABSOLUTE MAXIMUM RATINGS

Stresses above the values so mentioned above may cause permanent damage to the device. These values are for a stress rating only and do not imply that the device should be operated at conditions up to or above these values. Exposure of the device to the maximum rating values for extended periods of time may adversely affect the device reliability.

Parameter	Value	Unit
Storage Temperature	-65 to +150	C
Plastic Packages	-65 to +125	C
Output Short Circuit Current ¹	200	mA
Input and Output Voltage (with respect to ground) ²	-0.5 to V _{CC} +0.5	V
V _{CC}	-0.5 to V _{CC} +0.5	V

Notes:

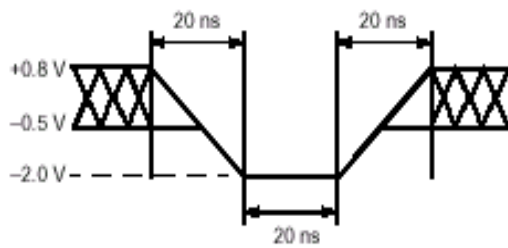
1. No more than one output shorted at a time. Duration of the short circuit should not be greater than one second.
2. Minimum DC voltage on input or I/O pins is -0.5 V. During voltage transitions, inputs may undershoot V_{SS} to -1.0V for periods of up to 50ns and to -2.0 V for periods of up to 20ns. See figure below. Maximum DC voltage on output and I/O pins is V_{CC} + 0.5 V. During voltage transitions, outputs may overshoot to V_{CC} + 2.0 V for periods up to 20ns. See figure below.

RECOMMENDED OPERATING RANGES¹

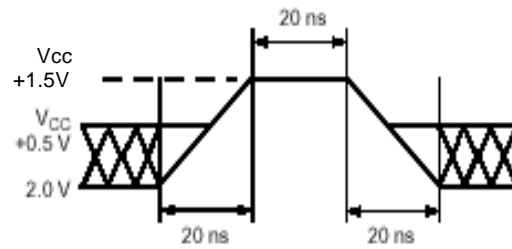
Parameter	Value	Unit
Ambient Operating Temperature Industrial Devices	-40 to 85	C
Operating Supply Voltage V _{CC}	Full: 2.7 to 3.6	V

Notes:

1. Recommended Operating Ranges define those limits between which the functionality of the device is guaranteed.



Maximum Negative Overshoot Waveform



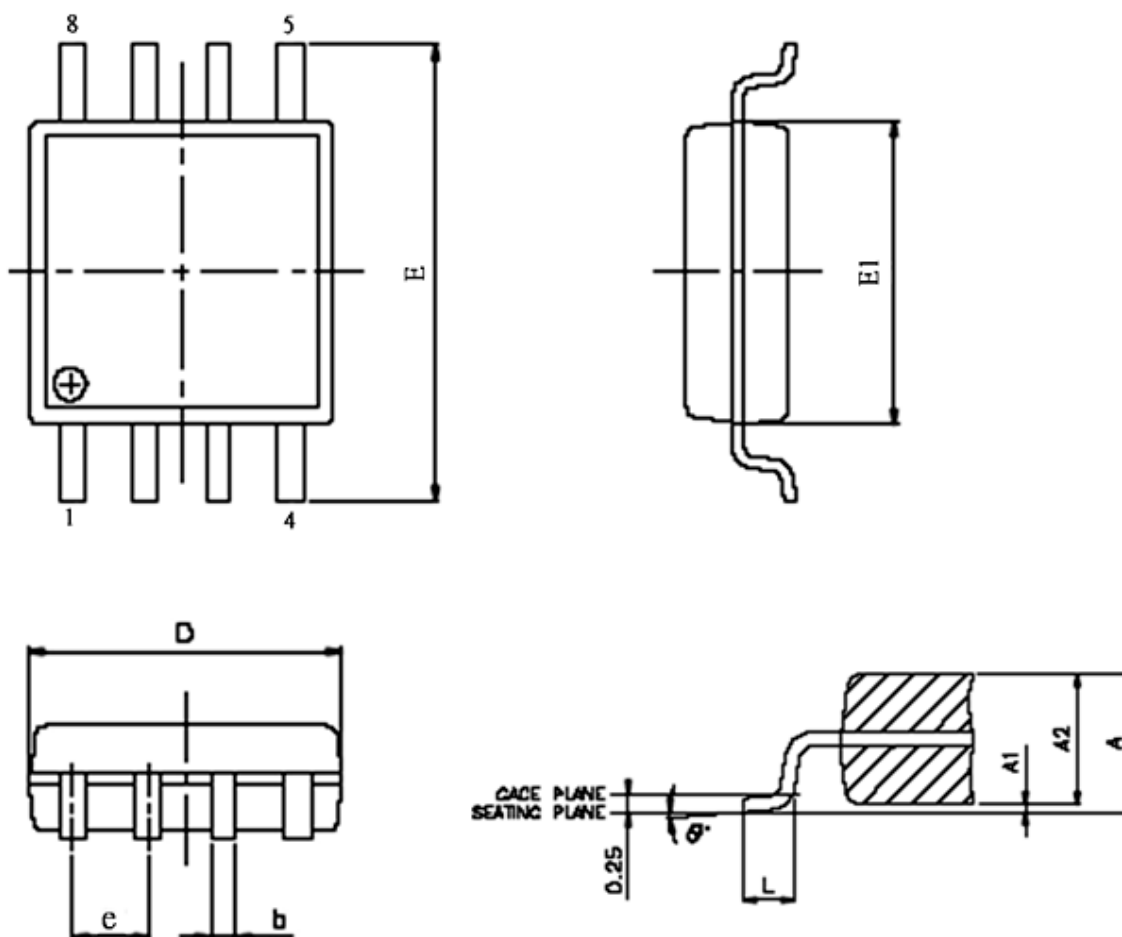
Maximum Positive Overshoot Waveform

Table 19. CAPACITANCE

 ($V_{CC} = 2.7-3.6V$)

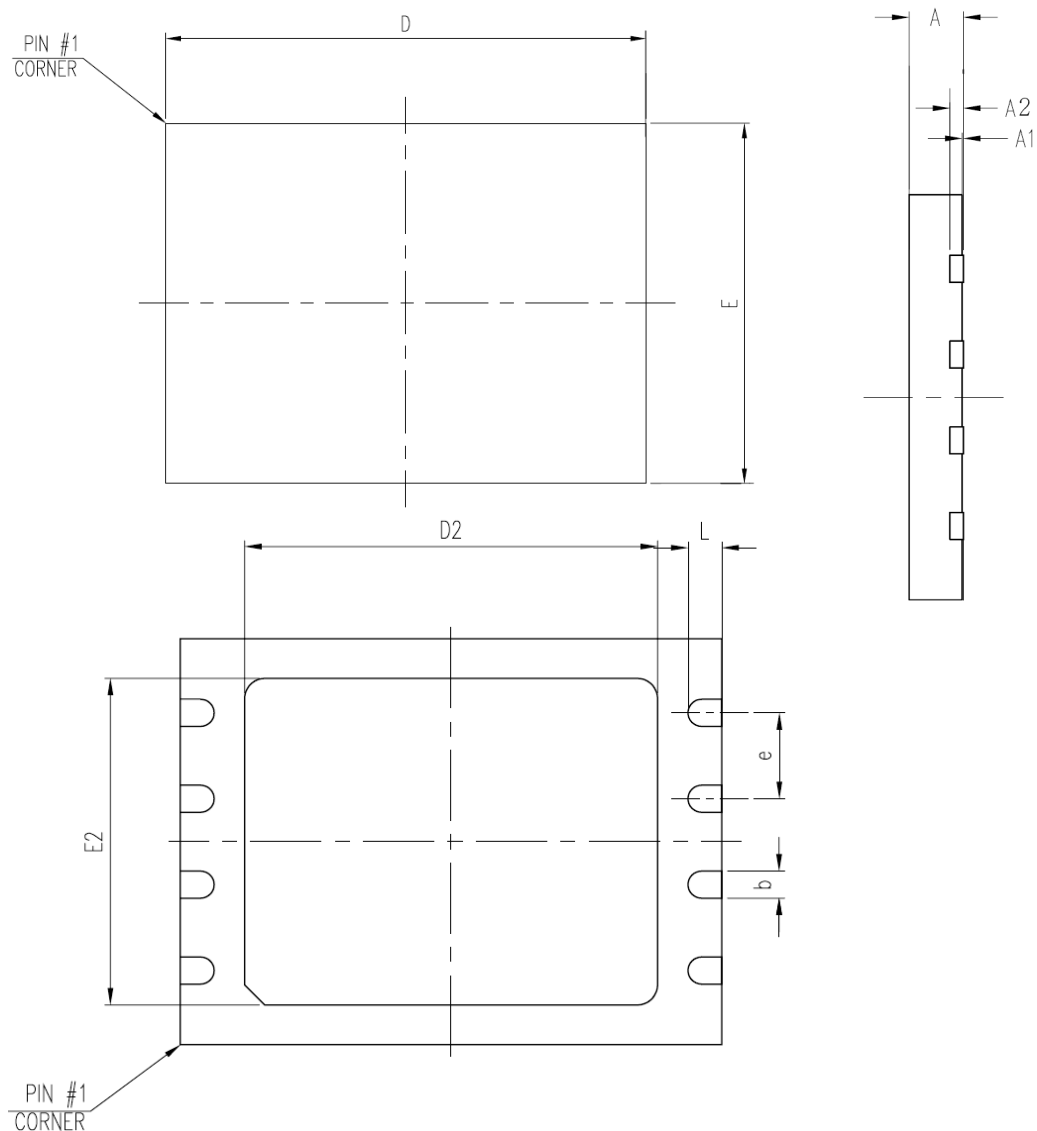
Parameter Symbol	Parameter Description	Test Setup	Typ	Max	Unit
C_{IN}	Input Capacitance	$V_{IN} = 0$		6	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0$		8	pF

Note : Sampled only, not 100% tested, at $T_A = 25^\circ C$ and a frequency of 20MHz.

PACKAGE MECHANICAL
Figure 45. SOP 200 mil (official name = 208 mil)


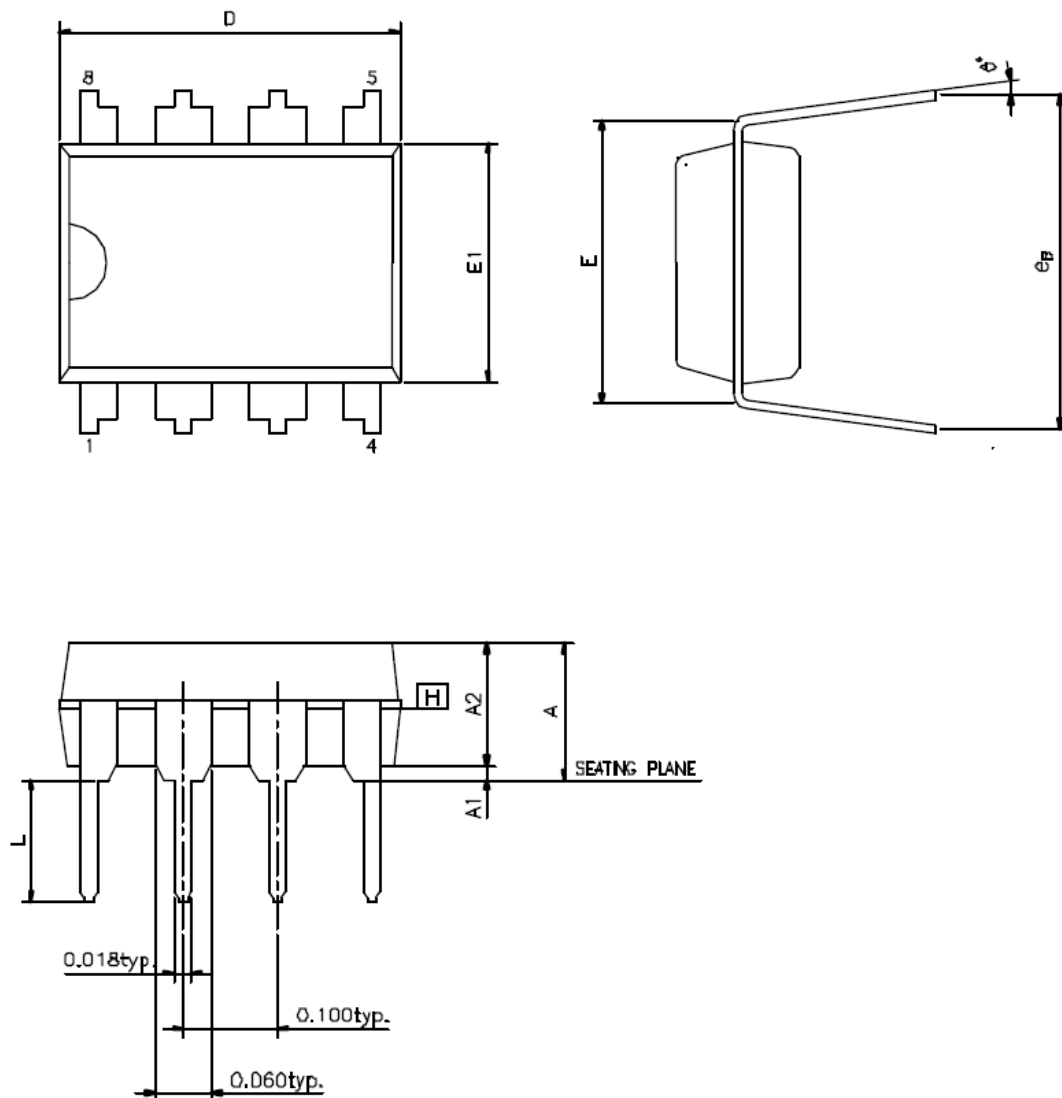
SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	1.75	1.975	2.20
A1	0.05	0.15	0.25
A2	1.70	1.825	1.95
D	5.15	5.275	5.40
E	7.70	7.90	8.10
E1	5.15	5.275	5.40
e	---	1.27	---
b	0.35	0.425	0.50
L	0.5	0.65	0.80
θ	0°	4°	8°

Note : 1. Coplanarity: 0.1 mm
 2. Max. allowable mold flash is 0.15 mm
 at the pkg ends, 0.25 mm between leads.

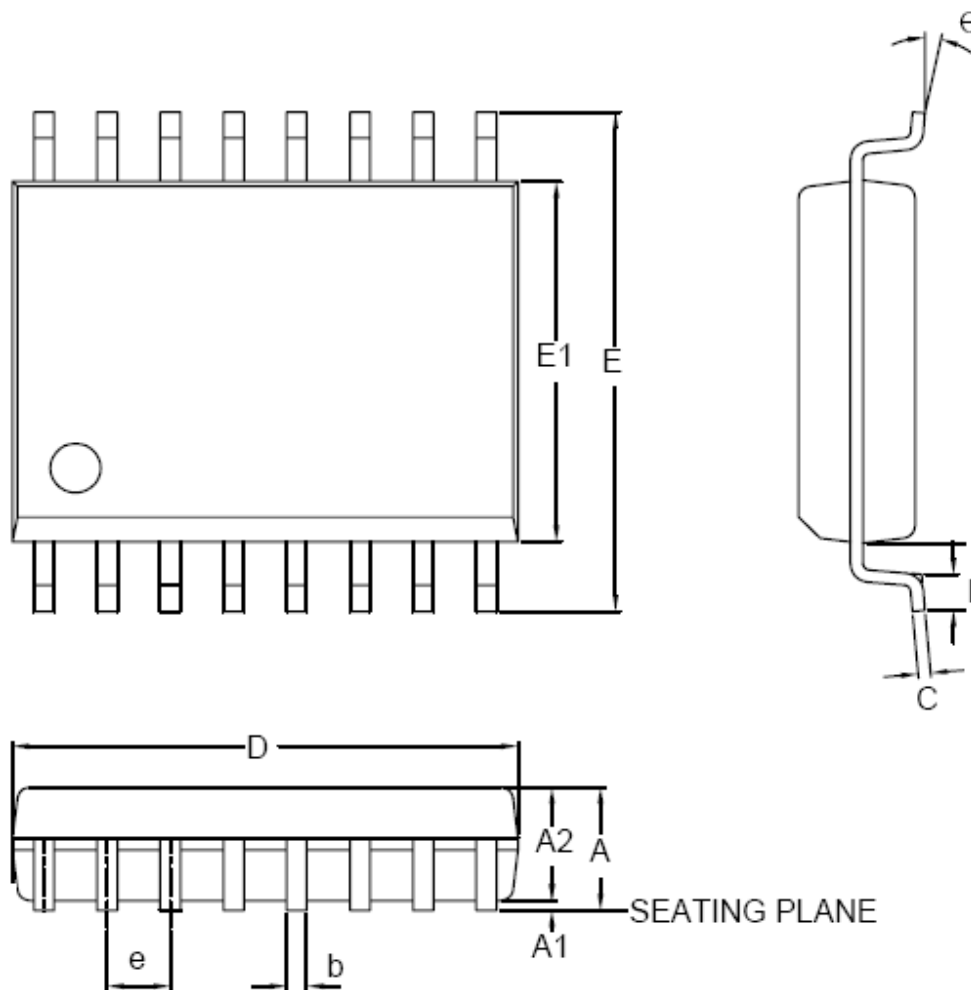
Figure 46. VDFN8 (5x6mm)


SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	0.70	0.75	0.80
A1	0.00	0.02	0.04
A2	---	0.20	---
D	5.90	6.00	6.10
E	4.90	5.00	5.10
D2	3.30	3.40	3.50
E2	3.90	4.00	4.10
e	---	1.27	---
b	0.35	0.40	0.45
L	0.55	0.60	0.65

Note: 1. Coplanarity: 0.1 mm

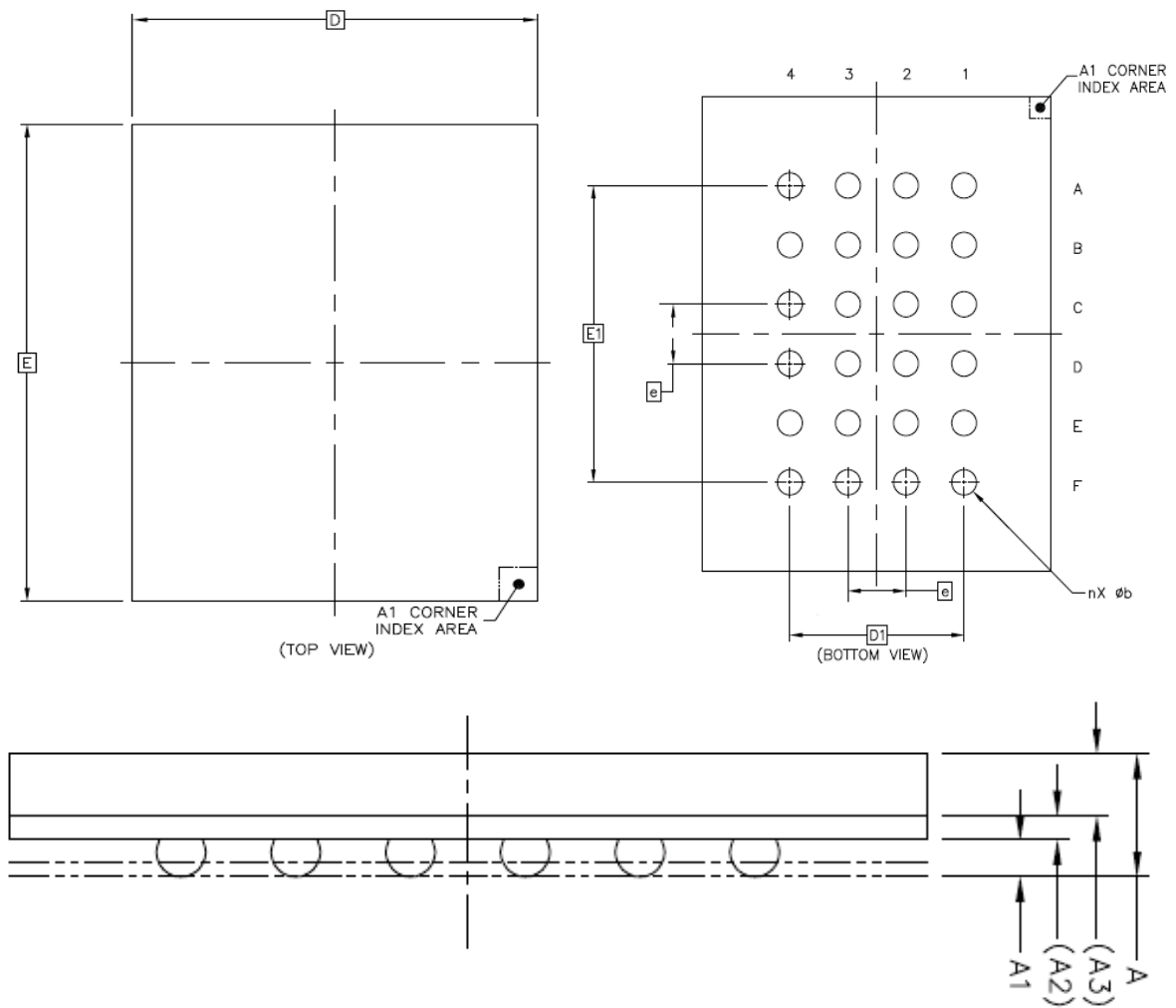
Figure 47. PDIP8


SYMBOL	DIMENSION IN INCH		
	MIN.	NOR	MAX
A	---	---	0.210
A1	0.015	---	---
A2	0.125	0.130	0.135
D	0.355	0.365	0.400
E	0.300	0.310	0.320
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
e_B	0.310	0.350	0.375
θ°	0	7	15

Figure 48. 16 LEAD SOP 300 mil


SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	---	---	2.65
A1	0.10	0.20	0.30
A2	2.25	---	2.40
C	0.20	0.25	0.30
D	10.10	10.30	10.50
E	10.00	---	10.65
E1	7.40	7.50	7.60
e	---	1.27	---
b	0.31	---	0.51
L	0.4	---	1.27
θ	0°	5°	8°

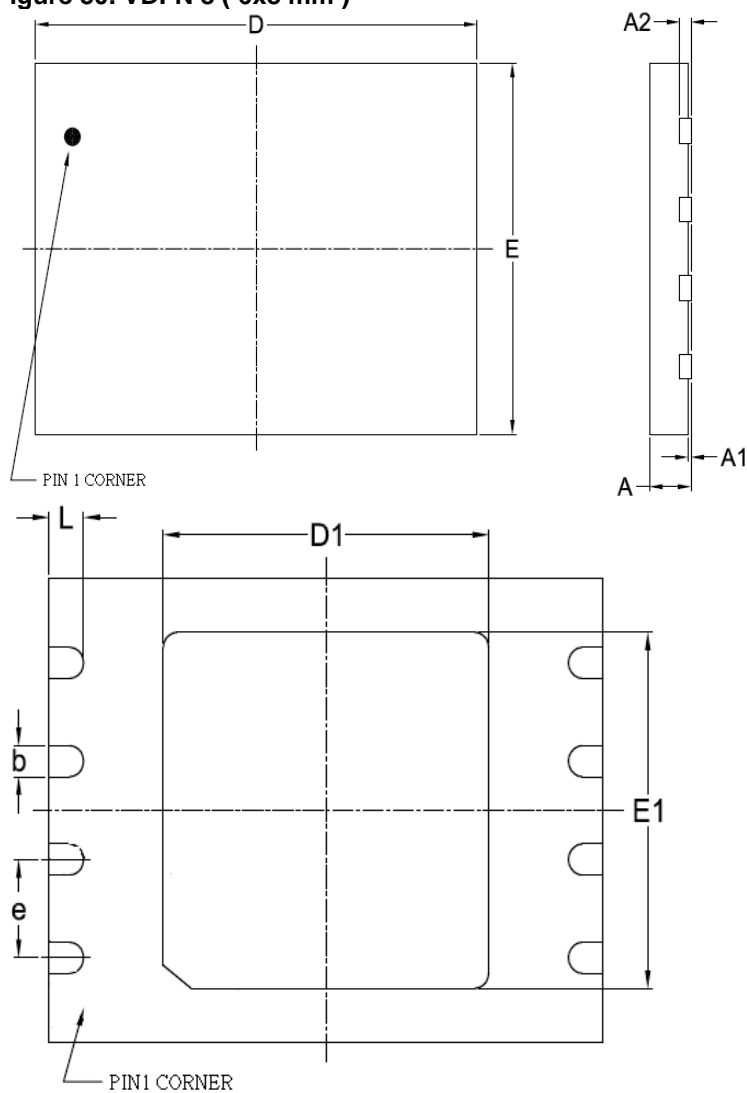
Note : 1. Coplanarity: 0.1 mm

Figure 49. 24-ball Thin Profile Fine-Pitch Ball Grid Array (6 x 8 mm) Package


SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	---	---	1.20
A1	0.27	---	0.37
A2	0.21 REF		
A3	0.54 REF		
D	6 BSC		
E	8 BSC		
D1	---	3.00	---
E1	---	5.00	---
e	---	1.00	---
b	---	0.40	---

Note : 1. Coplanarity: 0.1 mm

Figure 50. VDFN 8 (6x8 mm)



Notice:
This package can't contact to metal trace or pad on board due to expose metal pad underneath the package.



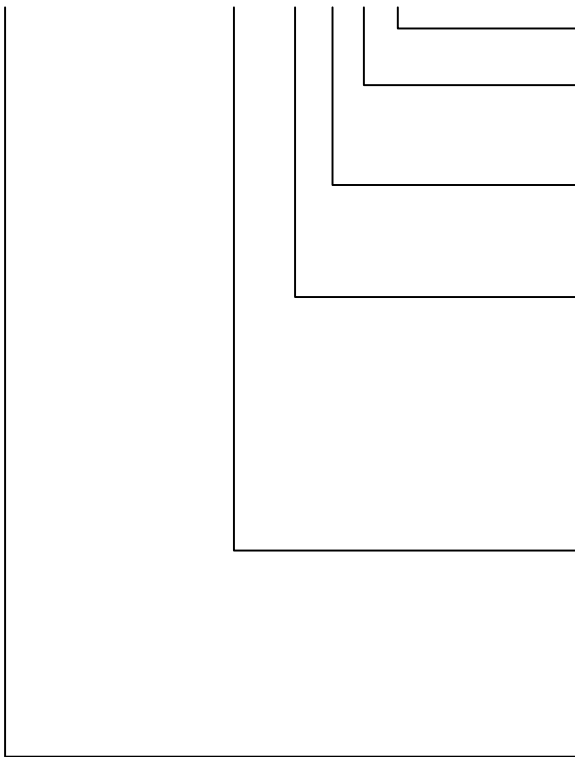
SYMBOL	DIMENSION IN MM		
	MIN.	NOR	MAX
A	0.70	0.75	0.80
A1	0.00	0.02	0.05
A2	---	0.20	---
D	7.90	8.00	8.10
E	5.90	6.00	6.10
D1	4.65	4.70	4.75
E1	4.55	4.60	4.65
e	---	1.27	---
b	0.35	0.40	0.48
L	0.4	0.50	0.60

Note : 1. Coplanarity: 0.1 mm



ORDERING INFORMATION

EN25QH128A - 104 H I P 2T



DIFFERENTIATION CODE

PACKAGING CONTENT

P = RoHS, Halogen-Free and REACH compliant

TEMPERATURE RANGE

I = Industrial (-40°C to +85°C)

PACKAGE

- H = 8-pin 200mil SOP
- W = 8-pin VDFN (5x6mm)
- Q = 8-pin PDIP
- F = 16-pin 300mil SOP
- BB = 24-ball TFBGA (6 x 8 x 1.2mm)
- Y = 8-pin VDFN (6x8mm)

SPEED

104 = 104 MHz

BASE PART NUMBER

EN = Eon Silicon Solution Inc.
25QH = 3V Serial Flash with 4KB Uniform-Sector
128 = 128 Megabit (16,384K x 8)
A = version identifier

**Revisions List**

Revision No	Description	Date
A	Initial Release	2016/06/27
B	Add VDFN (6x8mm) Package	2017/01/03
C	1.Remove "Suspend/Resume" 2.Delete Program / Erase Fail Flag bit of Read Status Register 2 3.Modify the specification of tSHSL, 30ns for both read and program/erase. 4. Output Driving Strength revised.	2017/05/05