EN25QX128A (2X)
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#, DQ$_0$, DQ$_1$, WP#, HOLD#/RESET#
  - Quad SPI: CLK, CS#, DQ$_0$, DQ$_1$, DQ$_2$, DQ$_3$
- High performance
  - Full voltage range
    - 104MHz clock rate for Single/Dual/Quad I/O Fast Read
  - Regulated voltage range: 3.0-3.6 volt
    - 133Mhz clock rate for Quad I/O Fast Read
- Support Serial Flash Discoverable
- Parameters (SFDP) signature
- Low power consumption
  - 6 mA typical active current
  - 1 µA typical standby 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 3x512 byte OTP security sectors
- Write suspend and resume
- Burst read with wrap(8/16/32/64 byte)
- Blank check bit
- Read Unique ID Number
- Minimum 100K endurance cycle
- Data retention time 20years
- Package Options
  - 8-pins SOP 200mil body width
  - 16-pins SOP 300mil body width
  - 8-contact VDFN/ WSON (6x5mm)
  - 8-contact VDFN/ WSON (8x6mm)
  - 8-contact USON (4x4X0.45mm)
  - All Pb-free packages are compliant RoHS, Halogen-Free and REACH.
- Industrial temperature Range

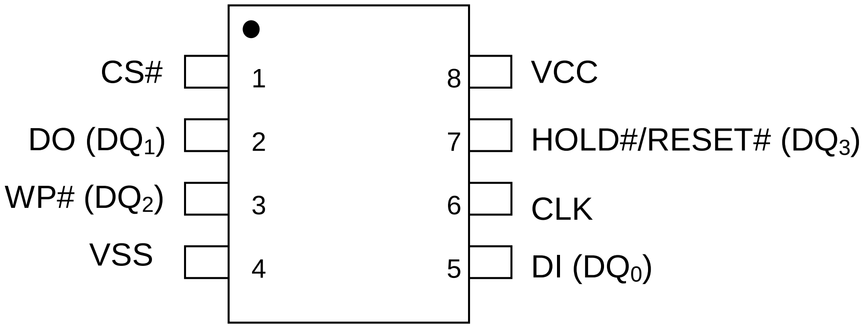
GENERAL DESCRIPTION

The device is a 128 Megabit (16,384K-byte) Serial Flash memory, with advanced write protection mechanisms. The device supports the single bit and four bits serial input and output commands via standard Serial Peripheral Interface (SPI) pins: Serial Clock, Chip Select, Serial DQ$_0$ (DI) and DQ$_1$ (DO), DQ$_2$ (WP#) and DQ$_3$ (HOLD#/RESET#). SPI clock frequencies of up to 104Mhz are supported allowing equivalent clock rates of 532Mhz (133Mhz 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.

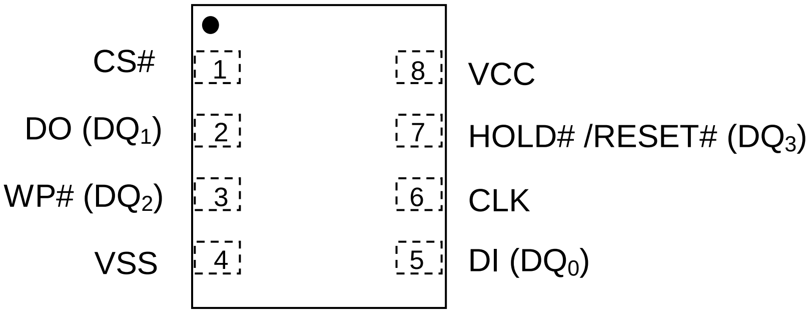
The device also offers a sophisticated method for protecting individual blocks against erroneous or malicious program and erase operations. By providing the ability to individually protect and unprotect blocks, a system can unprotect a specific block to modify its contents while keeping the remaining 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 device is designed to allow either single Sector/Block at a time or full chip erase operation. The device 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.

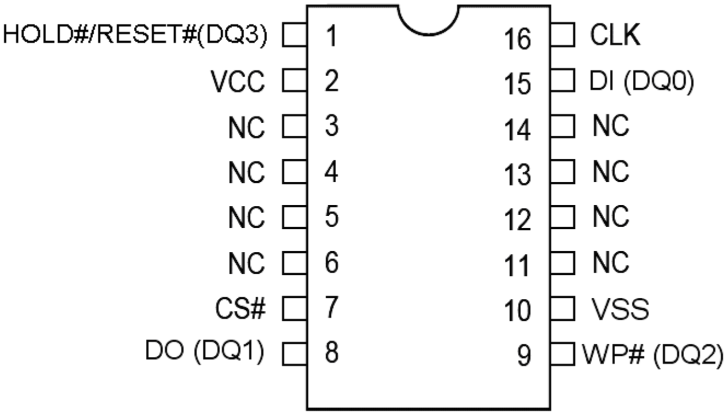
Figure.1 CONNECTION DIAGRAMS (TOP VIEW)



8 - LEAD SOP



8 – LEAD USON / VDFN / WSON



16 - LEAD SOP

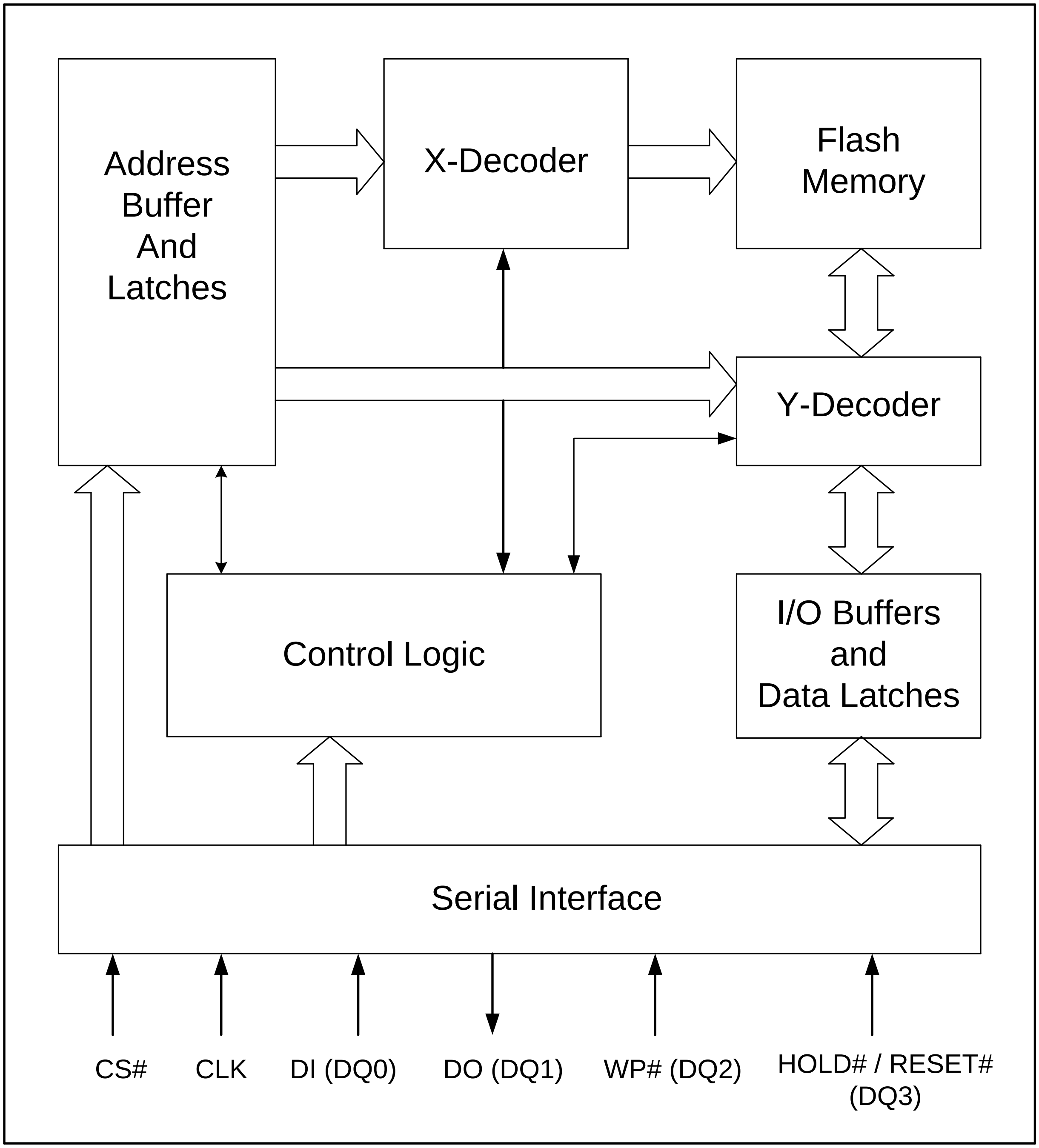
Table 1. Pin Names

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}
V_{CC}	Supply Voltage (2.7-3.6V)
V_{SS}	Ground
NC	No Connect

Note:

1. DQ_0 and DQ_1 are used for Dual and Quad instructions.
2. DQ_0 ~ DQ_3 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 ~ DQ_3 are used for Quad instructions.

SIGNAL DESCRIPTION

Serial Data Input, Output and IOs (DI, DO and DQ_0, DQ_1, DQ_2, DQ_3)

The device 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 DQ_0, DQ_1, DQ_2 and DQ_3) 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 (SR2.6, 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 (DQ_2) for Quad I/O operation.

HOLD (HOLD#)

The HOLD# pin allows the device to be paused while it is actively selected. When QE 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 (DQ_3) for Quad I/O operation.

RESET (RESET#)

The RESET# pin allows the device to be reset by the controller. When QE 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 (DQ_3) for Quad I/O operation. Set RESET# to low for a minimum period 1us (t_{HRST}) will interrupt any on-going instructions to have the device to initial state. The device can accept new instructions again in 28us (t_{HRSL}) 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	4080	FF0000h	FF0FFFh
254	509	4079	FEF000h	FEFFFFh
		⋮	⋮	⋮
	508	4064	FE0000h	FE0FFFh
253	507	4063	FDF000h	FDFFFFh
		⋮	⋮	⋮
	506	4048	FD0000h	FD0FFFh
⋮	⋮	⋮	⋮	⋮
242	485	3887	F2F000h	F2FFFFh
		⋮	⋮	⋮
	484	3872	F20000h	F20FFFh
241	483	3871	F1F000h	F1FFFFh
		⋮	⋮	⋮
	482	3856	F10000h	F10FFFh
240	481	3855	F0F000h	F0FFFFh
		⋮	⋮	⋮
	480	3840	F00000h	F00FFFh

64K Block	32K Block	Sector	Address range	
239	479	3839	EFF000h	EFFFFFh
		⋮	⋮	⋮
	478	3824	EF0000h	EF0FFFh
238	477	3823	EEF000h	EEFFFFh
		⋮	⋮	⋮
	476	3808	EE0000h	EE0FFFh
237	475	3807	EDF000h	EDFFFFh
		⋮	⋮	⋮
	474	3792	ED0000h	ED0FFFh
⋮	⋮	⋮	⋮	⋮
226	453	3631	E2F000h	E2FFFFh
		⋮	⋮	⋮
	452	3616	E20000h	E20FFFh
225	451	3615	E1F000h	E1FFFFh
		⋮	⋮	⋮
	450	3600	E10000h	E10FFFh
224	449	3599	E0F000h	E0FFFFh
		⋮	⋮	⋮
	448	3584	E00000h	E00FFFh

64K Block	32K Block	Sector	Address range	
223	447	3583	DFF000h	DFFFFFh
		⋮	⋮	⋮
	446	3568	DF0000h	DF0FFFh
222	445	3567	DEF000h	DEFFFFh
		⋮	⋮	⋮
	444	3552	DE0000h	DE0FFFh
221	443	3551	DDF000h	DDFFFFh
		⋮	⋮	⋮
	442	3536	DD0000h	DD0FFFh
⋮	⋮	⋮	⋮	⋮
210	421	3375	D2F000h	D2FFFFh
		⋮	⋮	⋮
	420	3360	D20000h	D20FFFh
209	419	3359	D1F000h	D1FFFFh
		⋮	⋮	⋮
	418	3344	D10000h	D10FFFh
208	417	3343	D0F000h	D0FFFFh
		⋮	⋮	⋮
	416	3328	D00000h	D00FFFh

64K Block	32K Block	Sector	Address range	
207	415	3327	CFF000h	CFFFFFh
		⋮	⋮	⋮
	414	3312	CF0000h	CF0FFFh
206	413	3311	CEF000h	CEFFFFh
		⋮	⋮	⋮
	412	3296	CE0000h	CE0FFFh
205	411	3295	CDF000h	CDFFFFh
		⋮	⋮	⋮
	410	3280	CD0000h	CD0FFFh
⋮	⋮	⋮	⋮	⋮
194	389	3119	C2F000h	C2FFFFh
		⋮	⋮	⋮
	388	3014	C20000h	C20FFFh
193	387	3103	C1F000h	C1FFFFh
		⋮	⋮	⋮
	386	3088	C10000h	C10FFFh
192	385	3087	C0F000h	C0FFFFh
		⋮	⋮	⋮
	384	3072	C00000h	C00FFFh

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

64K Block	32K Block	Sector	Address range	
191	383	3071	BFF000h	BFFFFFh
	382	⋮	⋮	⋮
		3056	BF0000h	BF0FFFh
190	382	3055	BEF000h	BEFFFFh
	380	⋮	⋮	⋮
		3040	BE0000h	BE0FFFh
189	379	3039	BDF000h	BDFFFFh
	378	⋮	⋮	⋮
		3024	BD0000h	BD0FFFh
⋮	⋮	⋮	⋮	⋮
178	357	2863	B2F000h	B2FFFFh
	356	⋮	⋮	⋮
		2848	B20000h	B20FFFh
177	355	2847	B1F000h	B1FFFFh
	354	⋮	⋮	⋮
		2832	B10000h	B10FFFh
176	353	2831	B0F000h	B0FFFFh
	352	⋮	⋮	⋮
		2816	B00000h	B00FFFh

64K Block	32K Block	Sector	Address range	
175	351	2815	AFF000h	AFFFFFh
	350	⋮	⋮	⋮
		2800	AF0000h	AF0FFFh
174	349	2799	AEF000h	AEFFFFh
	348	⋮	⋮	⋮
		2784	AE0000h	AE0FFFh
173	347	2783	ADF000h	ADFFFFh
	346	⋮	⋮	⋮
		2768	AD0000h	AD0FFFh
⋮	⋮	⋮	⋮	⋮
162	325	2607	A2F000h	A2FFFFh
	324	⋮	⋮	⋮
		2592	A20000h	A20FFFh
161	323	2591	A1F000h	A1FFFFh
	322	⋮	⋮	⋮
		2576	A10000h	A10FFFh
160	321	2575	A0F000h	A0FFFFh
	320	⋮	⋮	⋮
		2560	A00000h	A00FFFh

64K Block	32K Block	Sector	Address range	
159	319	2559	9FF000h	9FFFFFh
	318	⋮	⋮	⋮
		2544	9F0000h	9F0FFFh
158	317	2543	9EF000h	9EFFFFh
	316	⋮	⋮	⋮
		2528	9E0000h	9E0FFFh
157	315	2527	9DF000h	9DFFFFh
	314	⋮	⋮	⋮
		2512	9D0000h	9D0FFFh
⋮	⋮	⋮	⋮	⋮
146	293	2351	92F000h	92FFFFh
	292	⋮	⋮	⋮
		2336	920000h	920FFFh
145	291	2335	91F000h	91FFFFh
	290	⋮	⋮	⋮
		2320	910000h	910FFFh
144	289	2319	90F000h	90FFFFh
	288	⋮	⋮	⋮
		2304	900000h	900FFFh

64K Block	32K Block	Sector	Address range	
143	287	2303	8FF000h	8FFFFFh
	286	⋮	⋮	⋮
		2288	8F0000h	8F0FFFh
142	285	2287	8EF000h	8EFFFFh
	284	⋮	⋮	⋮
		2272	8E0000h	8E0FFFh
141	283	2271	8DF000h	8DFFFFh
	282	⋮	⋮	⋮
		2256	8D0000h	8D0FFFh
⋮	⋮	⋮	⋮	⋮
130	261	2095	82F000h	82FFFFh
	260	⋮	⋮	⋮
		2080	820000h	820FFFh
129	259	2079	81F000h	81FFFFh
	258	⋮	⋮	⋮
		2064	810000h	810FFFh
128	257	2063	80F000h	80FFFFh
	256	⋮	⋮	⋮
		2048	800000h	800FFFh

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

64K Block	32K Block	Sector	Address range	
127	255	2047	7FF000h	7FFFFFh
		⋮	⋮	⋮
	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	6FFFFFh
		⋮	⋮	⋮
	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	5FFFFFh
		⋮	⋮	⋮
	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	4FFFFFh
		⋮	⋮	⋮
	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	1023	3FF000h	3FFFFFh
		⋮	⋮	⋮
	126	1008	3F0000h	3F0FFFh
62	125	1007	3EF000h	3EFFFFh
		⋮	⋮	⋮
	124	992	3E0000h	3E0FFFh
61	123	991	3DF000h	3DFFFFh
		⋮	⋮	⋮
	122	976	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	2FFFFFh
		⋮	⋮	⋮
	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	1FFFFFh
		⋮	⋮	⋮
	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

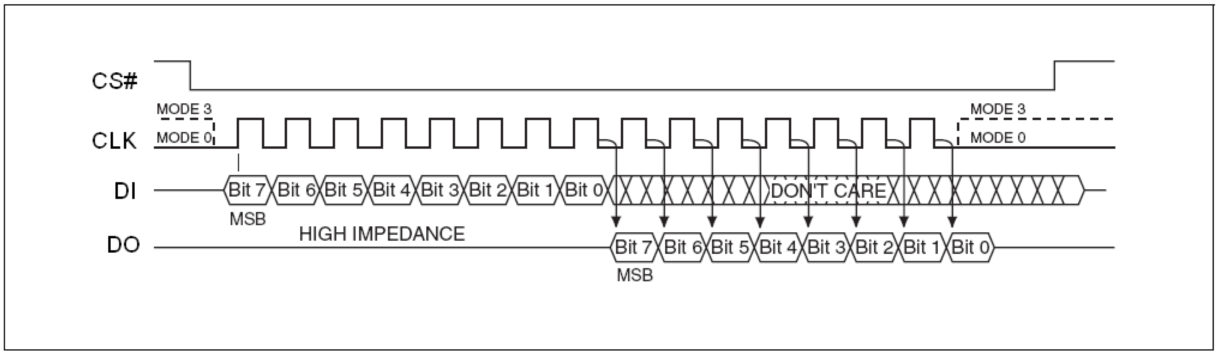
64K Block	32K Block	Sector	Address range	
15	31	255	0FF000h	0FFFFFh
		⋮	⋮	⋮
	30	240	0F0000h	0F0FFFh
14	29	239	0EF000h	0EFFFFh
		⋮	⋮	⋮
	28	224	0E0000h	0E0FFFh
13	27	223	0DF000h	0DFFFFh
		⋮	⋮	⋮
	26	208	0D0000h	0D0FFFh
⋮	⋮	⋮	⋮	⋮
2	5	47	02F000h	02FFFFh
		⋮	⋮	⋮
	4	32	020000h	020FFFh
1	3	31	01F000h	01FFFFh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

OPERATING FEATURES

Standard SPI Modes

The device 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



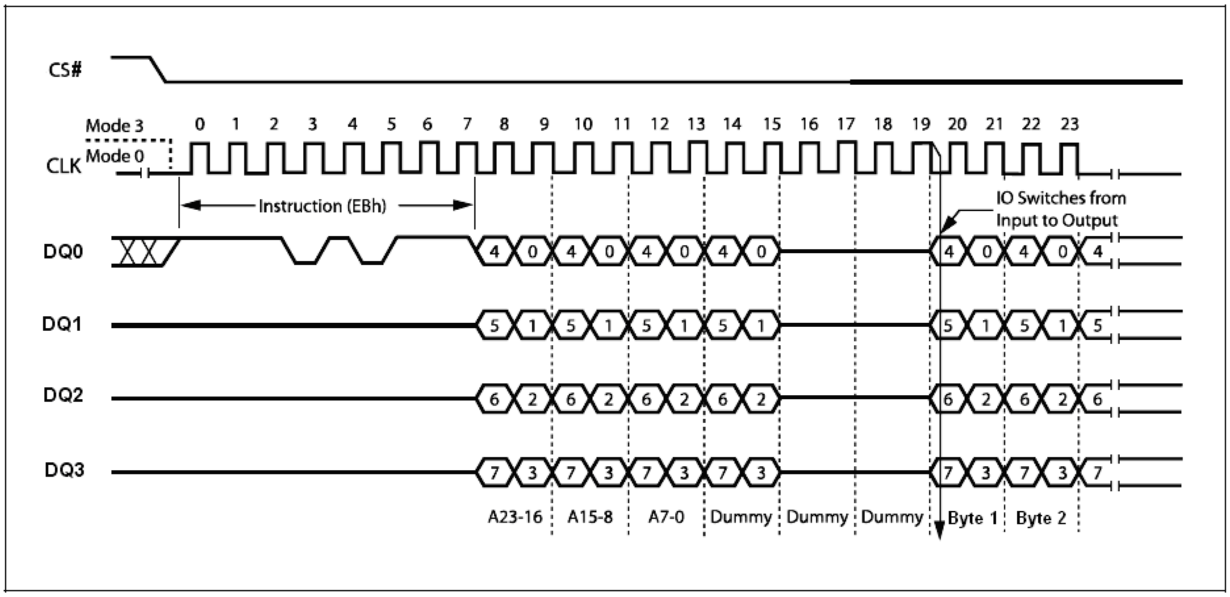
Dual SPI Instruction

The device 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; DQ_0 and DQ_1. All other operations use the standard SPI interface with single output signal.

Quad I/O SPI Modes

The device 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; DQ_0 and $DQ_{1,}$ and the WP# and HOLD#/RESET# pins become DQ_2 and DQ_3 respectively.

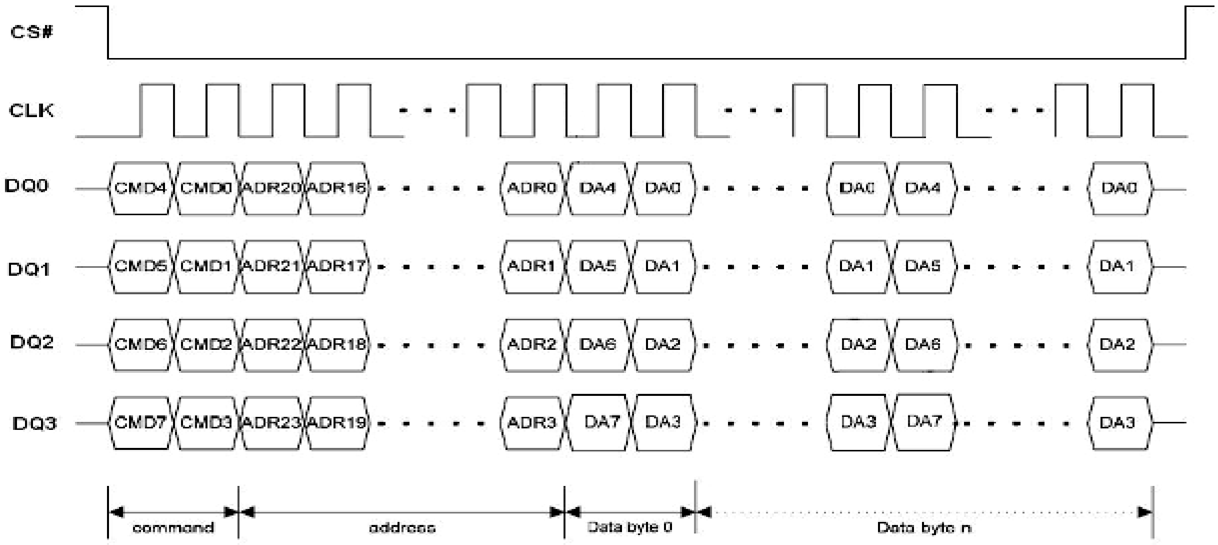
Figure 4. Quad SPI Modes



Full Quad SPI Modes (QPI)

The device 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; DQ$_0$ and DQ$_{1,}$ and the WP# and HOLD#/RESET# pins become DQ$_2$ and DQ$_3$ 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 device 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 (CMP, 4KBL, TB, 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 (CMP, 4KBL, TB, 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			
CMP Bit	4KBL Bit	T/B Bit	SR.4 Bit	SR.3 Bit	SR.2 Bit	Protect Areas	Addresses	Density(KB)	Portion
0	0	0	0	0	0	None	None	None	None
0	0	0	0	0	1	Block 252 to 255	FC0000h-FFFFFFh	256KB	Upper 1/64
0	0	0	0	1	0	Block 248 to 255	F80000h-FFFFFFh	512KB	Upper 1/32
0	0	0	0	1	1	Block 240 to 255	F00000h-FFFFFFh	1024KB	Upper 1/16
0	0	0	1	0	0	Block 224 to 255	E00000h-FFFFFFh	2048KB	Upper 1/8
0	0	0	1	0	1	Block 192 to 255	C00000h-FFFFFFh	4096KB	Upper 1/4
0	0	0	1	1	0	Block 128 to 255	800000h-FFFFFFh	8192KB	Upper 1/2
0	0	0	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
0	0	1	0	0	0	None	None	None	None
0	0	1	0	0	1	Block 0 to 3	000000h-03FFFFh	256KB	Lower 1/64
0	0	1	0	1	0	Block 0 to 7	000000h-07FFFFh	512KB	Lower 1/32
0	0	1	0	1	1	Block 0 to 15	000000h-0FFFFFh	1024KB	Lower 1/16
0	0	1	1	0	0	Block 0 to 31	000000h-1FFFFFh	2048KB	Lower 1/8
0	0	1	1	0	1	Block 0 to 63	000000h-3FFFFFh	4096KB	Lower 1/4
0	0	1	1	1	0	Block 0 to 127	000000h-7FFFFFh	8192KB	Lower 1/2
0	0	1	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
0	1	0	0	0	0	None	None	None	None
0	1	0	0	0	1	Block 255	FFF000h-FFFFFFh	4KB	Upper 1/4096
0	1	0	0	1	0	Block 255	FFE000h-FFFFFFh	8KB	Upper 1/2048
0	1	0	0	1	1	Block 255	FFC000h-FFFFFFh	16KB	Upper 1/1024
0	1	0	1	0	0	Block 255	FF8000h-FFFFFFh	32KB	Upper 1/512
0	1	0	1	0	1	Block 255	FF8000h-FFFFFFh	32KB	Upper 1/512
0	1	0	1	1	0	Block 255	FF8000h-FFFFFFh	32KB	Upper 1/512
0	1	0	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
0	1	1	0	0	0	None	None	None	None
0	1	1	0	0	1	Block 0	000000h-000FFFh	4KB	Lower 1/4096
0	1	1	0	1	0	Block 0	000000h-001FFFh	8KB	Lower 1/2048
0	1	1	0	1	1	Block 0	000000h-003FFFh	16KB	Lower 1/1024
0	1	1	1	0	0	Block 0	000000h-007FFFh	32KB	Lower 1/512
0	1	1	1	0	1	Block 0	000000h-007FFFh	32KB	Lower 1/512
0	1	1	1	1	0	Block 0	000000h-007FFFh	32KB	Lower 1/512

0	1	1	1	1	1	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	0	0	0	0	0	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	0	0	0	0	1	Block 0 to 251	000000h-FBFFFFh	16128KB	Lower 63/64
1	0	0	0	1	0	Block 0 to 247	000000h-F7FFFFh	15872KB	Lower 31/32
1	0	0	0	1	1	Block 0 to 239	000000h-EFFFFFh	15360KB	Lower 15/16
1	0	0	1	0	0	Block 0 to 223	000000h-DFFFFFh	14336KB	Lower 7/8
1	0	0	1	0	1	Block 0 to 191	000000h-BFFFFFh	12288KB	Lower 3/4
1	0	0	1	1	0	Block 0 to 127	000000h-7FFFFFh	8192KB	Lower 1/2
1	0	0	1	1	1	None	None	None	None
1	0	1	0	0	0	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	0	1	0	0	1	Block 4 to 255	040000h-FFFFFFh	16128KB	Upper 63/64
1	0	1	0	1	0	Block 8 to 255	080000h-FFFFFFh	15872KB	Upper 31/32
1	0	1	0	1	1	Block 16 to 255	100000h-FFFFFFh	15360KB	Upper 15/16
1	0	1	1	0	0	Block 32 to 255	200000h-FFFFFFh	14336KB	Upper 7/8
1	0	1	1	0	1	Block 64 to 255	400000h-FFFFFFh	12288KB	Upper 3/4
1	0	1	1	1	0	Block 128 to 255	800000h-FFFFFFh	8192KB	Upper 1/2
1	0	1	1	1	1	None	None	None	None
1	1	0	0	0	0	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	1	0	0	0	1	Block 0 to 255	000000h-FFEFFFh	16380KB	Lower 4095/4096
1	1	0	0	1	0	Block 0 to 255	000000h-FFDFFFh	16376KB	Lower 2047/2048
1	1	0	0	1	1	Block 0 to 255	000000h-FFBFFFh	16368KB	Lower 1023/1024
1	1	0	1	0	0	Block 0 to 255	000000h-FF7FFFh	16352KB	Lower 511/512
1	1	0	1	0	1	Block 0 to 255	000000h-FF7FFFh	16352KB	Lower 511/512
1	1	0	1	1	0	Block 0 to 255	000000h-FF7FFFh	16352KB	Lower 511/512
1	1	0	1	1	1	None	None	None	None
1	1	1	0	0	0	Block 0 to 255	000000h-FFFFFFh	16384KB	All
1	1	1	0	0	1	Block 0 to 255	001000h-FFFFFFh	16380KB	Upper 4095/4096
1	1	1	0	1	0	Block 0 to 255	002000h-FFFFFFh	16376KB	Upper 2047/2048
1	1	1	0	1	1	Block 0 to 255	004000h-FFFFFFh	16368KB	Upper 1023/1024
1	1	1	1	0	0	Block 0 to 255	008000h-FFFFFFh	16352KB	Upper 511/512
1	1	1	1	0	1	Block 0 to 255	008000h-FFFFFFh	16352KB	Upper 511/512
1	1	1	1	1	0	Block 0 to 255	008000h-FFFFFFh	16352KB	Upper 511/512
1	1	1	1	1	1	None	None	None	None

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 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 Resume	30h/7Ah						
Write Suspend	B0h/75h						
Write Enable (WERN)	06h						
Volatile Status Register Write Enable [3]	50h						
Write Disable (WRDI)	04h						
Read Status Register (RDSR)	05h	(S7-S0) [4]					continuous [5]
Write Status Register (WRSR)	01h	S7-S0	(SR2.7-SR2.0) [9]	(SR3.7-SR3.0) [9]			
Read Status Register 2 (RDSR2)	09h/35h	(SR2.7-SR2.0) [4]					continuous [5]
Write Status Register 2 (WRSR2)	31h	SR2.7-SR2.0					
Read Status Register 3 (RDSR3)	95h/15h	(SR3.7-SR3.0) [4]					continuous [5]
Write Status Register 3 (WRSR3)	C0h/11h	SR3.7-SR3.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)		
Read OTP array	48h	A23-A16	A15-A8	A7-A0	Dummy	D7-D0	(Next Byte) Continuous
Program OTP array	42h	A23-A16	A15-A8	A7-A0	D7-D0	D7-D0	(Next Byte) Continuous
Erase OTP array	44h	A23-A16	A15-A8	A7-A0			
Read SFDP mode and Unique ID Number	5Ah	A23-A16	A15-A8	A7-A0	dummy	(D7-D0)	(Next Byte) Continuous

Note:

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.
9. WREN(01h) support 8 or 16 or 24 bit register value input for status register, status register 2 and status register 3.

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)
Burst Read with Wrap	0Ch	24 bits	8 bits / 8 clocks	(D7-D0, …)	(Next Byte) 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			
32K Half Block Erase (HBE)	52h	24 bits			
64K Block Erase (BE)	D8h	24 bits			
Chip Erase (CE)	C7h/ 60h	0 bits			

Table 6. Manufacturer and Device Identification

OP Code	(M7-M0)	(ID15-ID0)	(ID7-ID0)
ABh			17h
90h	1Ch		17h
9Fh	1Ch	7118h	

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 device 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. **It is recommended to check the WIP bit and WSE/WSP bits in Status register and Status register2 before issuing the Software Reset command.**

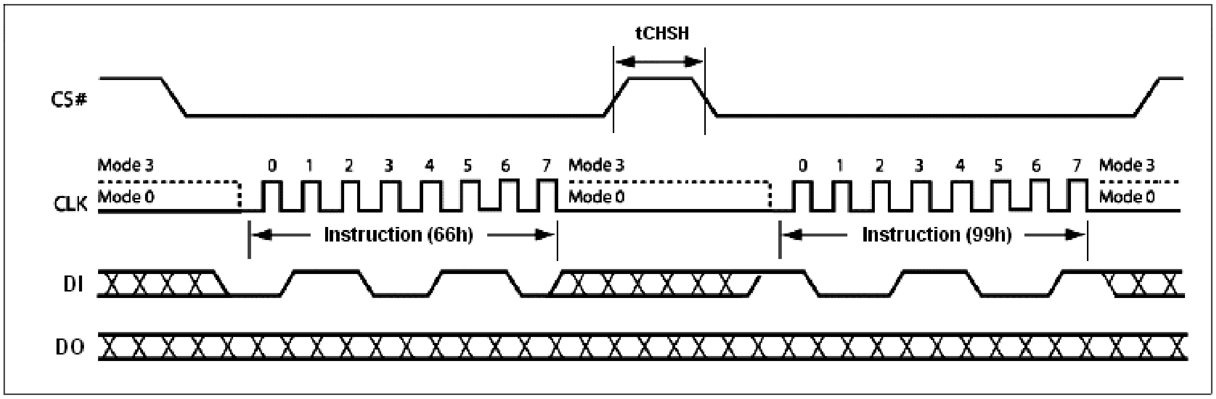


Figure 6. Reset-Enable and Reset Sequence Diagram

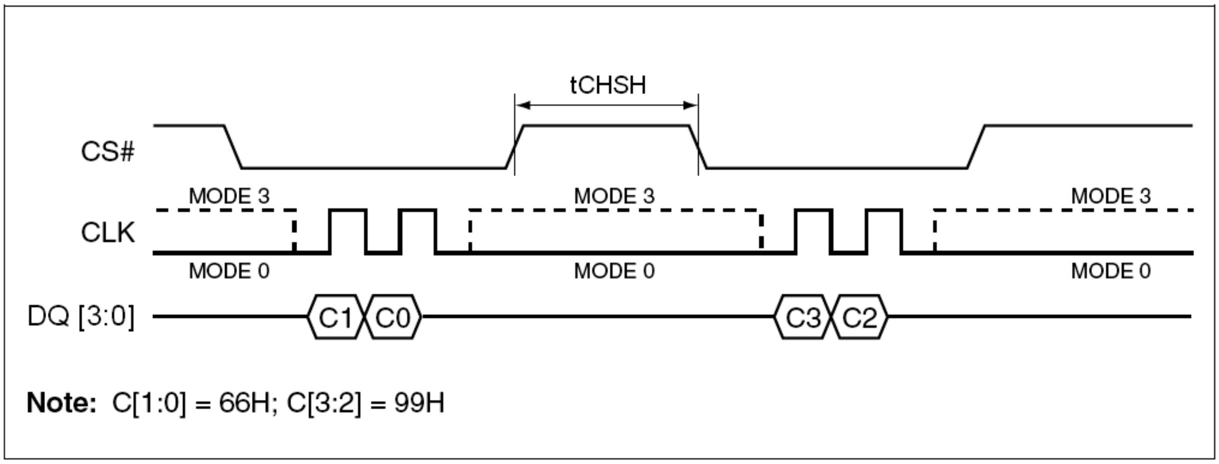
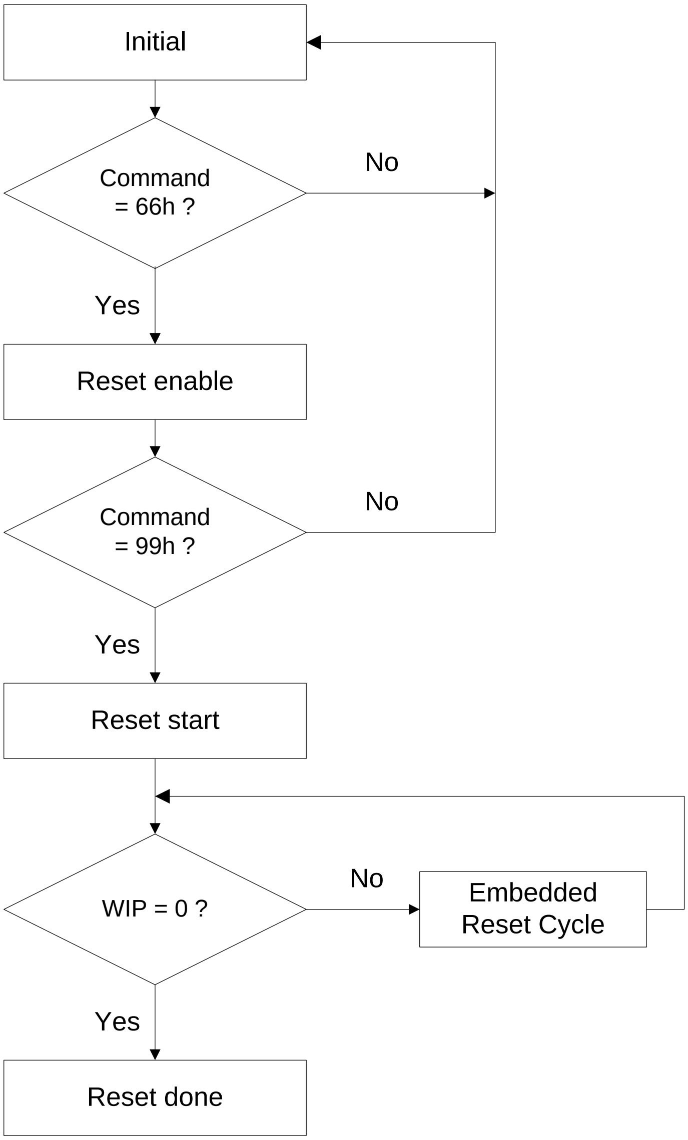


Figure 6.1 . Reset-Enable and Reset Sequence Diagram in QPI Mode

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 and suspend mode to back to SPI mode.
5. This flow can release the device from Deep power down mode.
6. The Status Register Bit and Status Register 2/3 Bits 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.
8. User can't do software/ hardware reset command while doing erase operation.

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), Quad Input Page Program (32h) and Quad output fast read (6Bh) modes while the Enable Quad Peripheral Interface mode (EQPI) (38h) turns on.

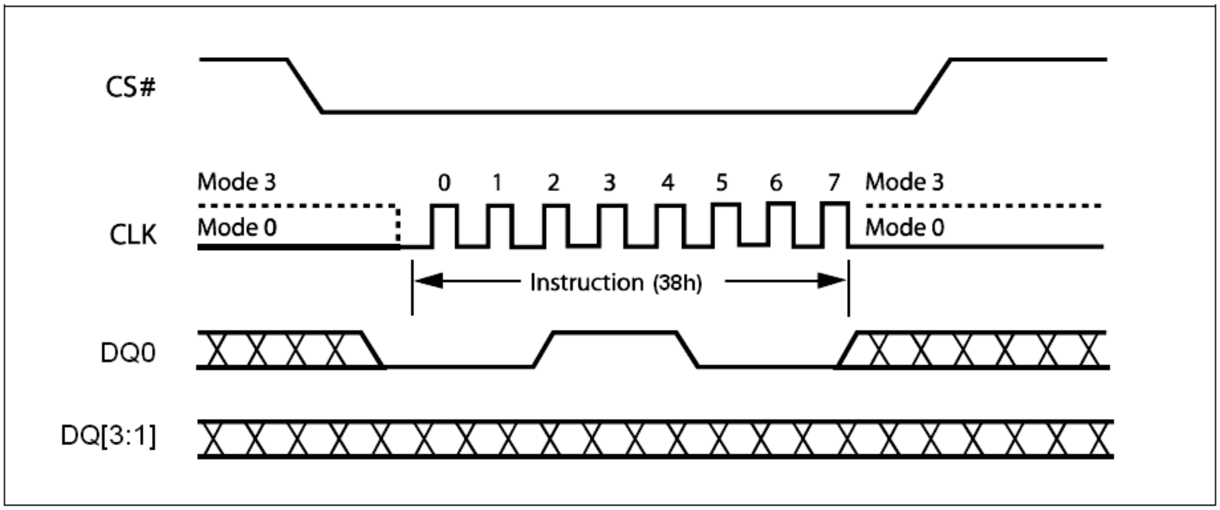


Figure 7. Enable Quad Peripheral Interface mode Sequence Diagram

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), Half Block Erase (HBE), 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.

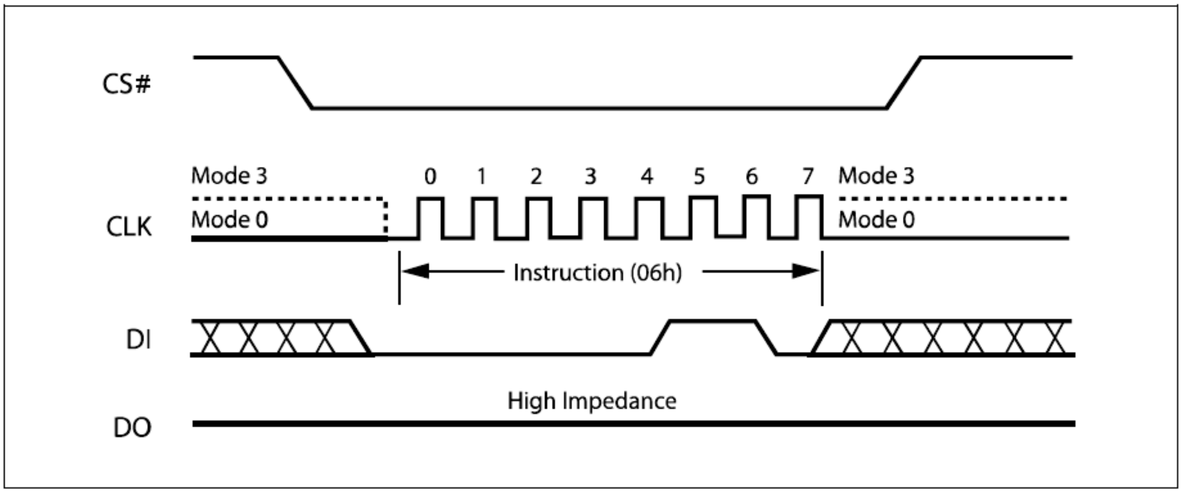


Figure 8. Write Enable Instruction Sequence Diagram

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 valid for 'Write Status Register' (01h), WRSR2 and WRSR3 commands 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) or WRSR2 or WRSR3. The Status Register bits will be refresh to Volatile Status Register (SR[7:2] or SR2[7:0] or SR3[7:0]) within tSHSL2 (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.

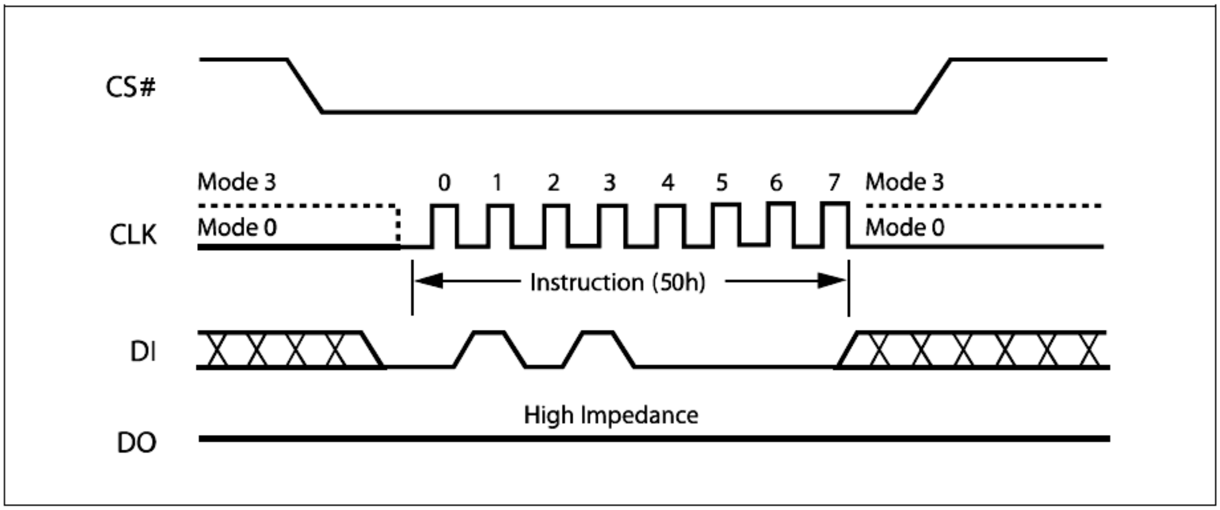


Figure 9. Volatile Status Register Write Enable Instruction Sequence Diagram

Write Disable (WRDI) (04h)

The Write Disable instruction (Figure 10) resets the Write Enable Latch (WEL) bit in the Status Register to a 0. 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.

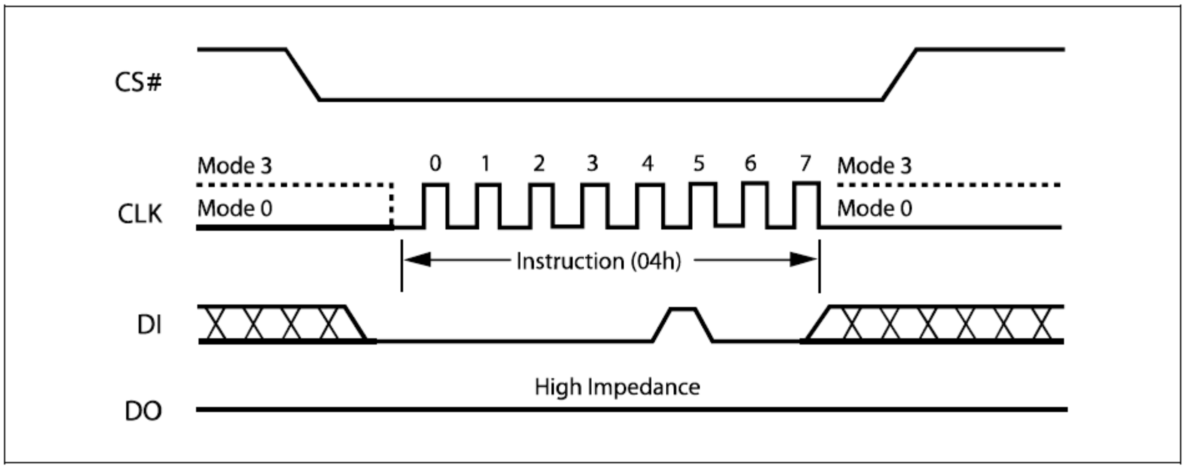


Figure 10. Write Disable Instruction Sequence Diagram

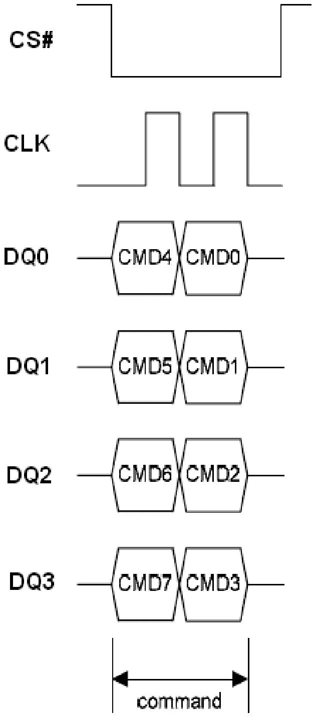


Figure 10.1 Write Enable/Disable Instruction Sequence in QPI Mode

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.

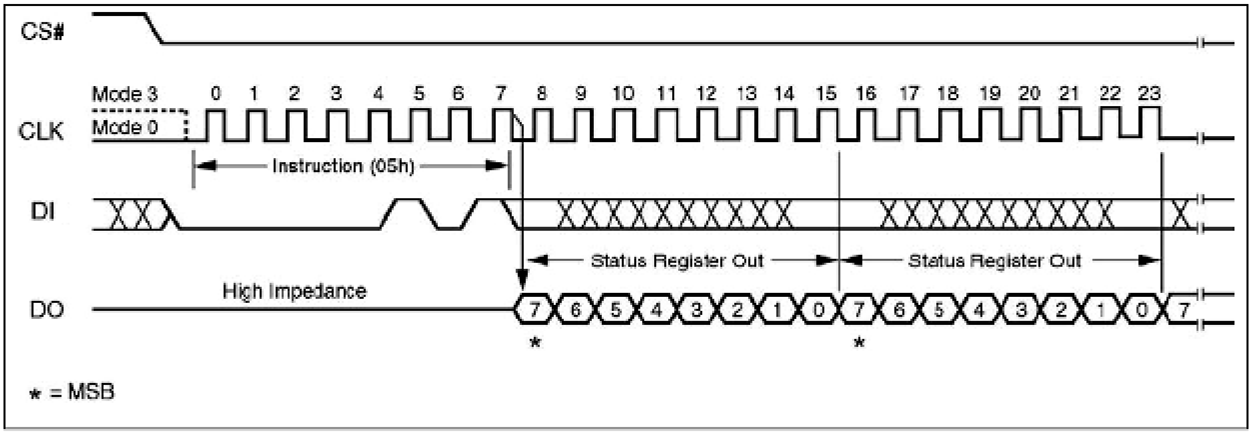


Figure 11. Read Status Register Instruction Sequence Diagram

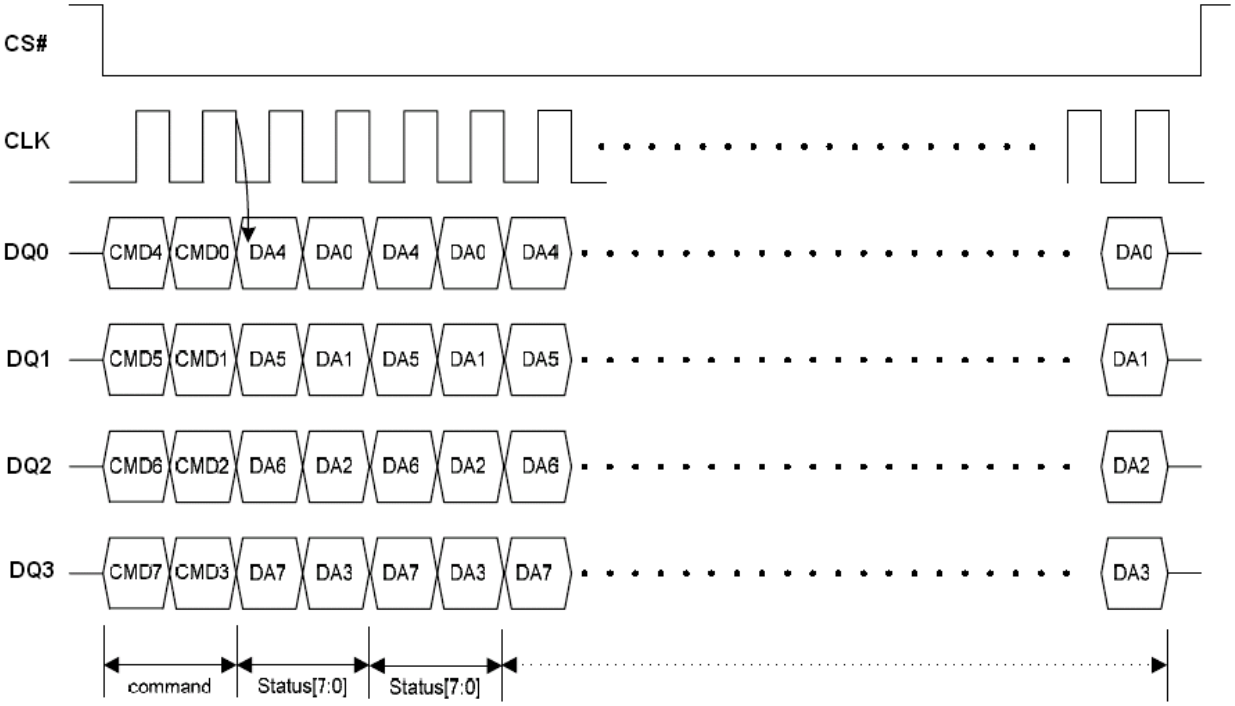

CS#
CLK
DQ0
CMD4 CMD0 DA4 DA0 DA4 DA0 DA4 DA0
DQ1
CMD5 CMD1 DA5 DA1 DA5 DA1 DA5 DA1
DQ2
CMD6 CMD2 DA6 DA2 DA6 DA2 DA6 DA2
DQ3
CMD7 CMD3 DA7 DA3 DA7 DA3 DA7 DA3
command
Status[7:0]
Status[7:0]

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 Status Register Protect	4KBL bit (4KB Boot Lock)	TB bit (Top / Bottom 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	0 = 64KB-Block (default 0) (note 2)	1 = Bottom 0 = Top (default 0) (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

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.

BP2, BP1, BP0 bits. The Block Protect (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 (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 (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 (BP2, BP1, BP0) bits are 0.

TB bit. The Top/Bottom Protect Bit (TB) controls if the Block Protect Bits (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.

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

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, SR2.6, SR6, 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.

Read Status Register 2 (RDSR 2) (09h/35h)

The Read Status Register 2 (RDSR2) instruction allows the Status Register 2 to be read. The Status Register 2 may be read at any time, even while a Write Suspend or Write Resume cycle is in progress. 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 2 continuously, as shown in Figure 12.

The instruction sequence is shown in Figure 12.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

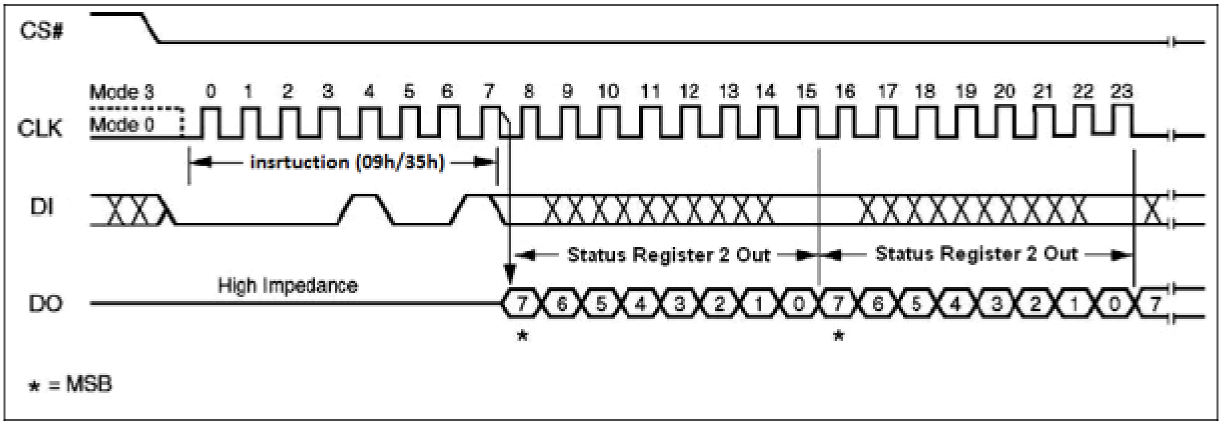


Figure 12. Read Status Register 2 Instruction Sequence Diagram

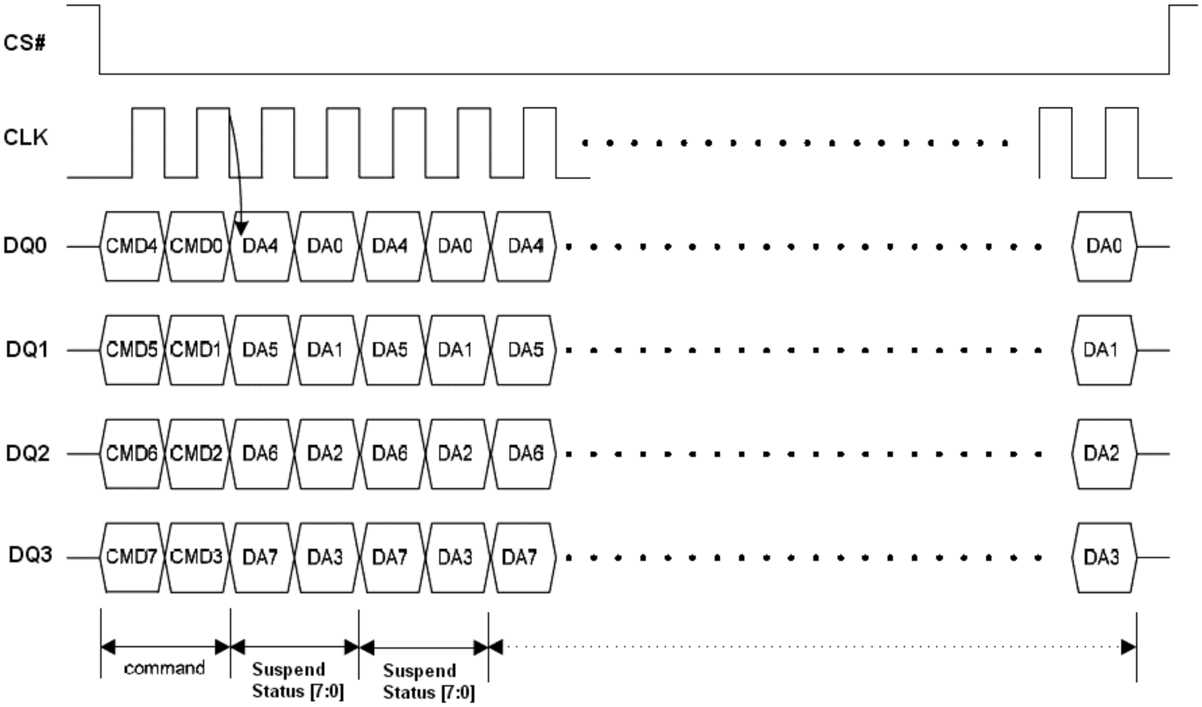

CS#
CLK
DQ0
CMD4 CMD0 DA4 DA0 DA4 DA0 DA4 DA0
DQ1
CMD5 CMD1 DA5 DA1 DA5 DA1 DA5 DA1
DQ2
CMD6 CMD2 DA6 DA2 DA6 DA2 DA6 DA2
DQ3
CMD7 CMD3 DA7 DA3 DA7 DA3 DA7 DA3
command
Suspend Status [7:0]
Suspend Status [7:0]

Figure 12.1 Read Status Register 2 Instruction Sequence in QPI Mode

Table 8. Status Register 2 Bit Locations

SR2.7	SR2.6	SR2.5	SR2.4	SR2.3	SR2.2	SR2.1	SR2.0
WSE (Write Suspend Erase status bit)	CMP bit	SPL0 bit	SPL1 bit	SPL2 bit	WSP (Write Suspend Program bits)	QE	Reserved bit
1 = Erase suspended 0 = Erase is not suspended	(note 2)	1 = OTP1 sector is protected	1 = OTP2 sector is protected	1 = OTP3 sector is protected	1 = Program suspended 0 = Program is not suspended	1 = WP# and HOLD#/RESET# disable 0 = WP# and HOLD#/RESET# enable (default 0)	
Indicator bit	Non-volatile / Volatile bit	OTP bit	OTP bit	OTP bit	Indicator bit	Non-volatile / Volatile bit	

Note:

1. The default of each Indicator bit is “0” at Power-up or after reset.
2. See the “Protected Area Sizes Sector Organization” table.

The status and control bits of the Suspend Status Register 2 are as follows:

WSE bit. The Write Suspend Erase Status (WSE) bit indicates when an Erase operation has been suspended. The WSE bit is “1” after the host issues a suspend command during an Erase operation. Once the suspended Erase resumes, the WSE bit is reset to “0”.

WSP bit. The Write Suspend Program Status (WSP) bit indicates when a Program operation has been suspended. The WSP is “1” after the host issues a suspend command during the Program operation. Once the suspended Program resumes, the WSP bit is reset to “0”.

SPL2 bit. The SPL2 bit is non-volatile One Time Program (OTP) bit in status register that provide the write protect control and status to the security sector 2. User can read/program/erase security sector 2 as normal sector while SPL2 value is equal 0, after SPL2 is programmed with 1 by WRSR command, the security sector 2 is protected from program and erase operation. The SPL2 bit can only be programmed once.

SPL1 bit. The SPL1 bit is non-volatile One Time Program (OTP) bit in status register that provide the write protect control and status to the security sector 1. User can read/program/erase security sector 1 as normal sector while SPL1 value is equal 0, after SPL1 is programmed with 1 by WRSR command, the security sector 1 is protected from program and erase operation. The SPL1 bit can only be programmed once.

SPL0 bit. The SPL0 bit is non-volatile One Time Program (OTP) bit in status register that provide the write protect control and status to the security sector 0. User can read/program/erase security sector 0 as normal sector while SPL0 value is equal 0, after SPL0 is programmed with 1 by WRSR command, the security sector 0 is protected from program and erase operation. The SPL0 bit can only be programmed once.

CMP bit. The Complement Protect bit(CMP) is a non-volatile bit in Status Register 2. It is used in conjunction with 4KBL, TB, BP2, BP1, BP0 bits to provide mode flexibility for the array protection. The default setting is CMP=0.

QE bit. The Quad Enable (QE) bit is a non-volatile Read/Write bit in the Status Register 2 to disable WP# and Hold#/RESET# before Quad operation. When it is “0” (factory default), the WP# and HOLD#/RESET# are enabled. On the other hand, while QE bit is “1”, the WP# and HOLD#/RESET# are disabled.
No matter QE is “0” or “1”, the system can executes Quad Input/ Output FAST_READ (EBh) or EQPI (38h) command directly. User can use Flash Programmer to set QE bit as “1” and then the host system can let WP# and HOLD# keep floating in SPI mode.

Reserved bit. Status Register 2 bit locations SR2.0 is reserved for future use. Current devices will read 0 for these bit locations. It is recommended to mask out the reserved bit when testing the Suspend Status Register. Doing this will ensure compatibility with future devices.

Read Status Register 3 (RDSR 3) (95h/15h)

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.

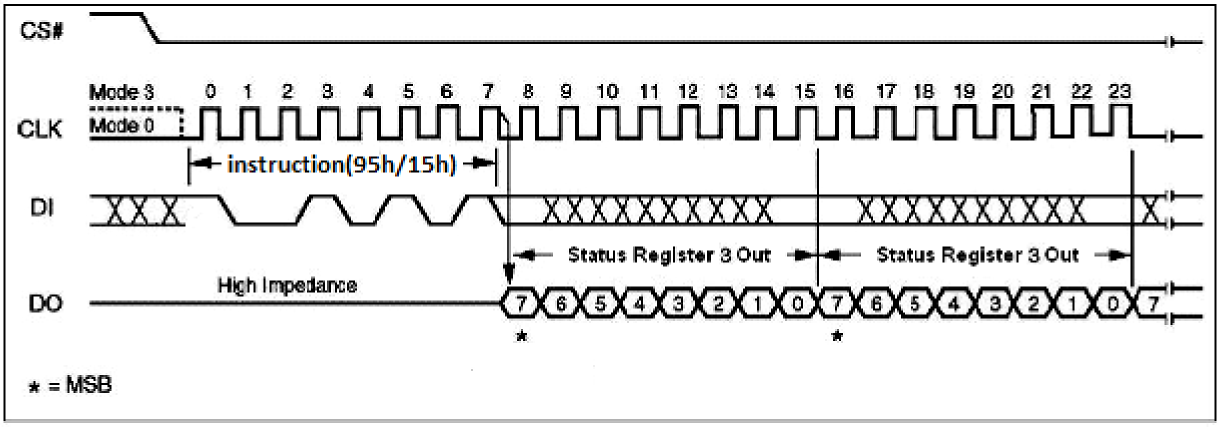


Figure 13. Read Status Register 3 Instruction Sequence Diagram

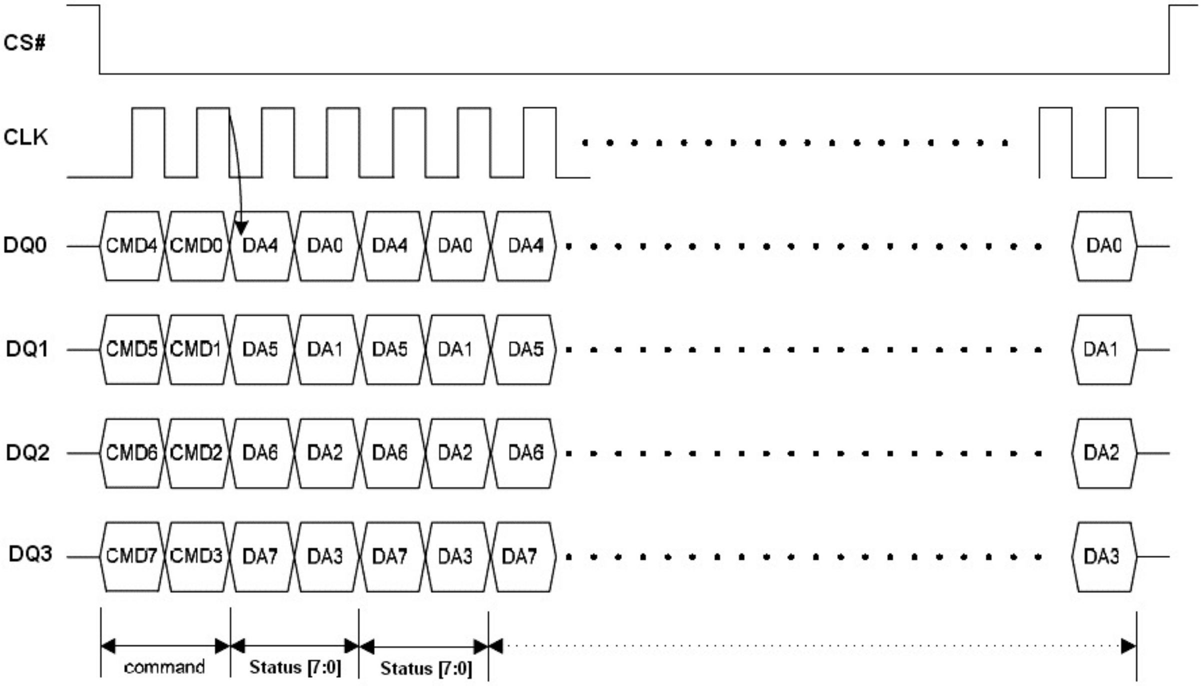

CS#
CLK
DQ0
CMD4 CMD0 DA4 DA0 DA4 DA0 DA4 DA0
DQ1
CMD5 CMD1 DA5 DA1 DA5 DA1 DA5 DA1
DQ2
CMD6 CMD2 DA6 DA2 DA6 DA2 DA6 DA2
DQ3
CMD7 CMD3 DA7 DA3 DA7 DA3 DA7 DA3
command
Status [7:0]
Status [7:0]

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 Driving Strength. The Output Driving Strength bits indicate the status of output Drive Strength in I/O pins.

Blank check bit. This bit is related with whole chip blank as factory default. Once any byte is programmed, this bit turns to 0 and will not be restored by further erase operation.

HRSW bit. The HOLD#/RESET# switch bit (HRSW bit), Non-Volatile / 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 QE bit is “0”. If QE bit is set to “1”, the HOLD# and RESET# functions are disabled, the pin acts as a dedicated data I/O pin.

Burst Length. The Burst Length bits indicate the status of wrap burst read length.

Table 9. Status Register 3 Bit Locations

SR3.7	SR3.6	SR3.5	SR3.4	SR3.3	SR3.2	SR3.1	SR3.0
HRSW bit (HOLD#/RESET# switch)	Output Drive Strength		Burst Length		Blank check	reserved	
1 = RESET# enable 0 = HOLD# enable (default 0)	00 = 67% Deafult 01 = 100% 10 = 50% (1/2) Drive 11 = 33% (1/3) Drive		00 = 8 Bytes(default) 01 = 16 Bytes 10 = 32 Bytes 11 = 64 Bytes		**1 = flash is blank after ship out (default)** 0 = flash had been programmed		
Non-volatile/volatile bit	Non-volatile/volatile bit		Non-volatile/volatile bit		Indicator bit		

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 or data bytes on Serial Data Input (DI). The WRSR instruction also support multi bytes data input to set other status registers.

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 or 16th or 32th 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 (4KBL, TB, 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.

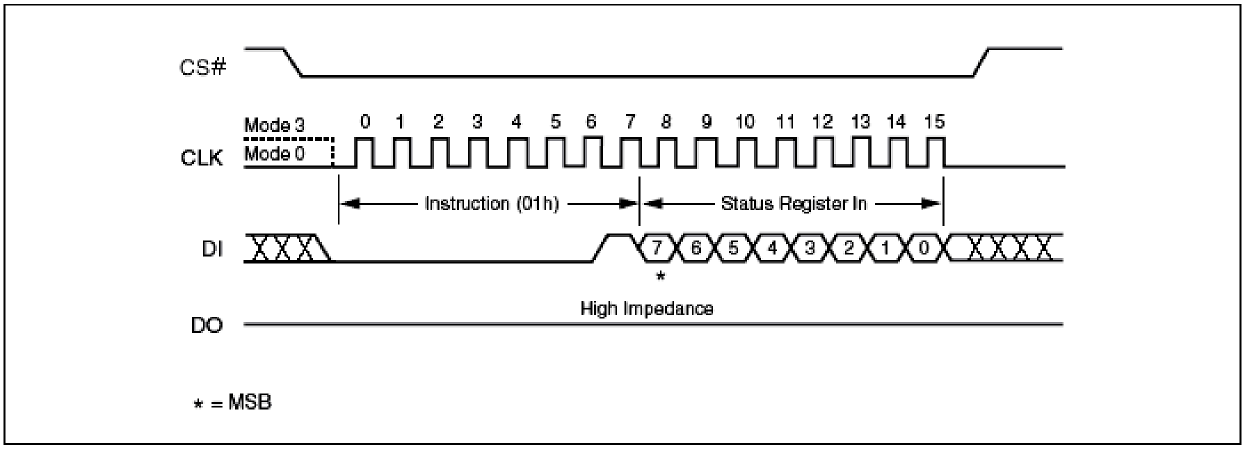


Figure 14. Write Status Register Instruction Sequence Diagram

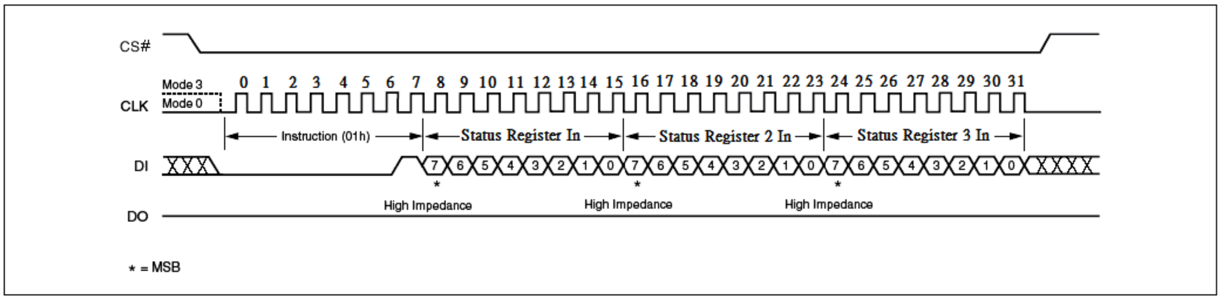


Figure 14. Write Status Register Instruction Sequence Diagram (multi byte)

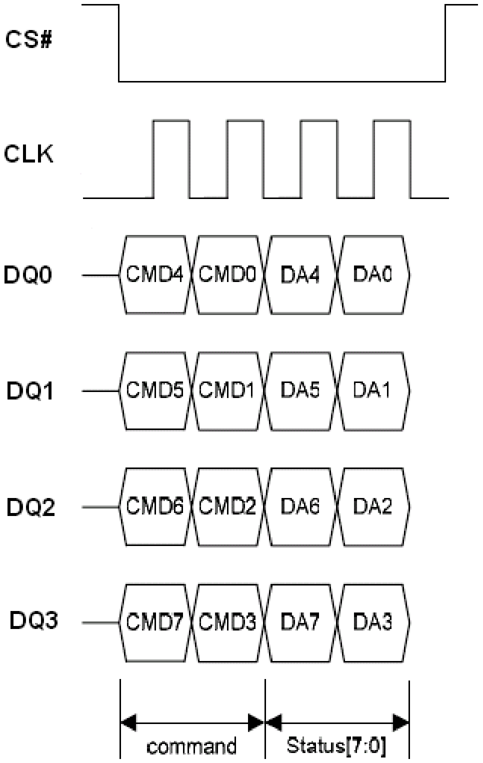


Figure 14.1 Write Status Register Instruction Sequence in QPI Mode

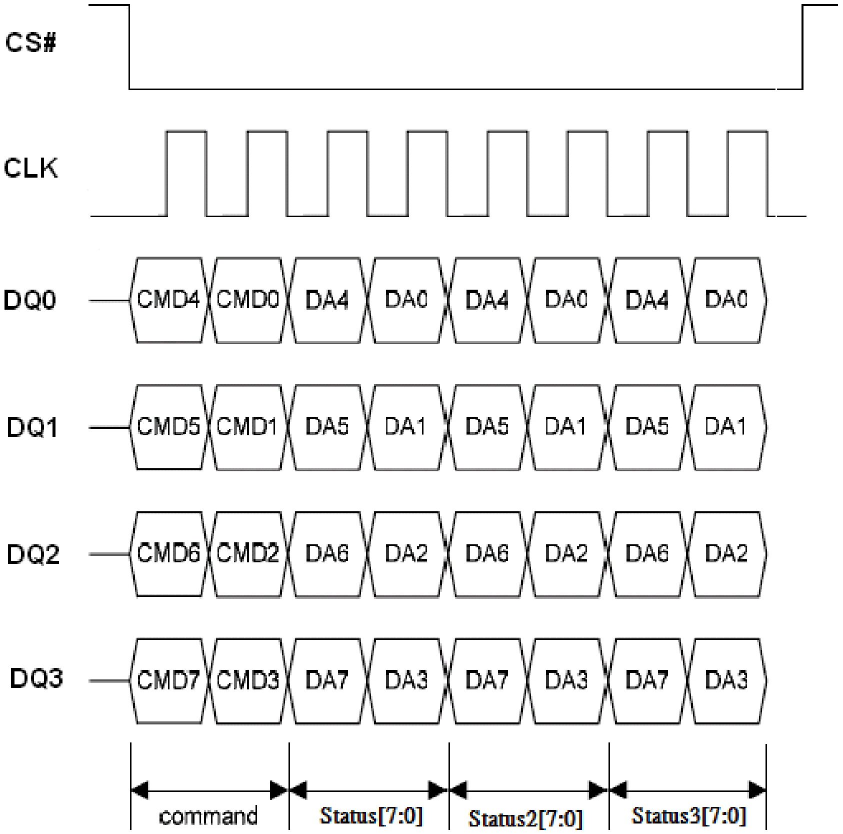


Figure 14.1 Write Status Register Instruction Sequence in QPI Mode (multi byte)

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.

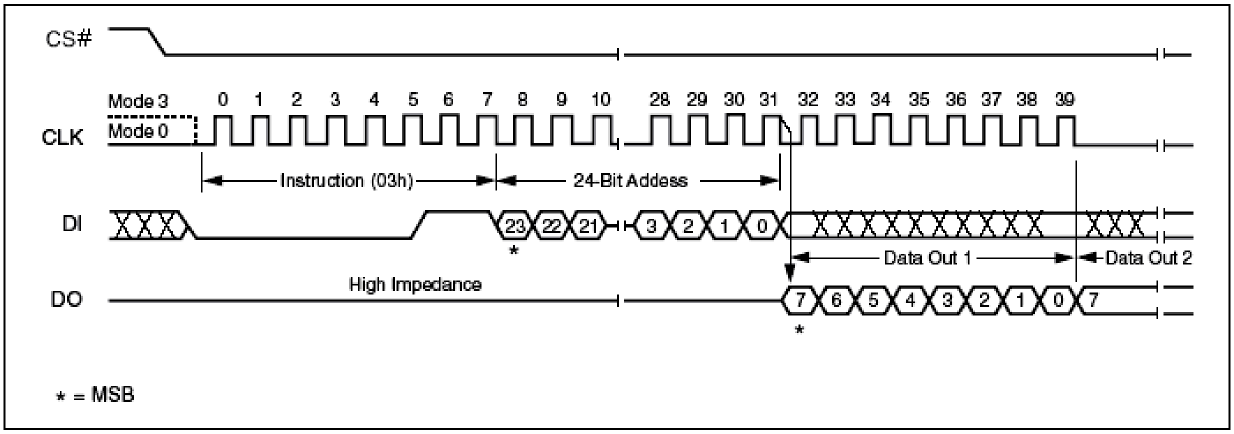


Figure 15. Read Data Instruction Sequence Diagram

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.

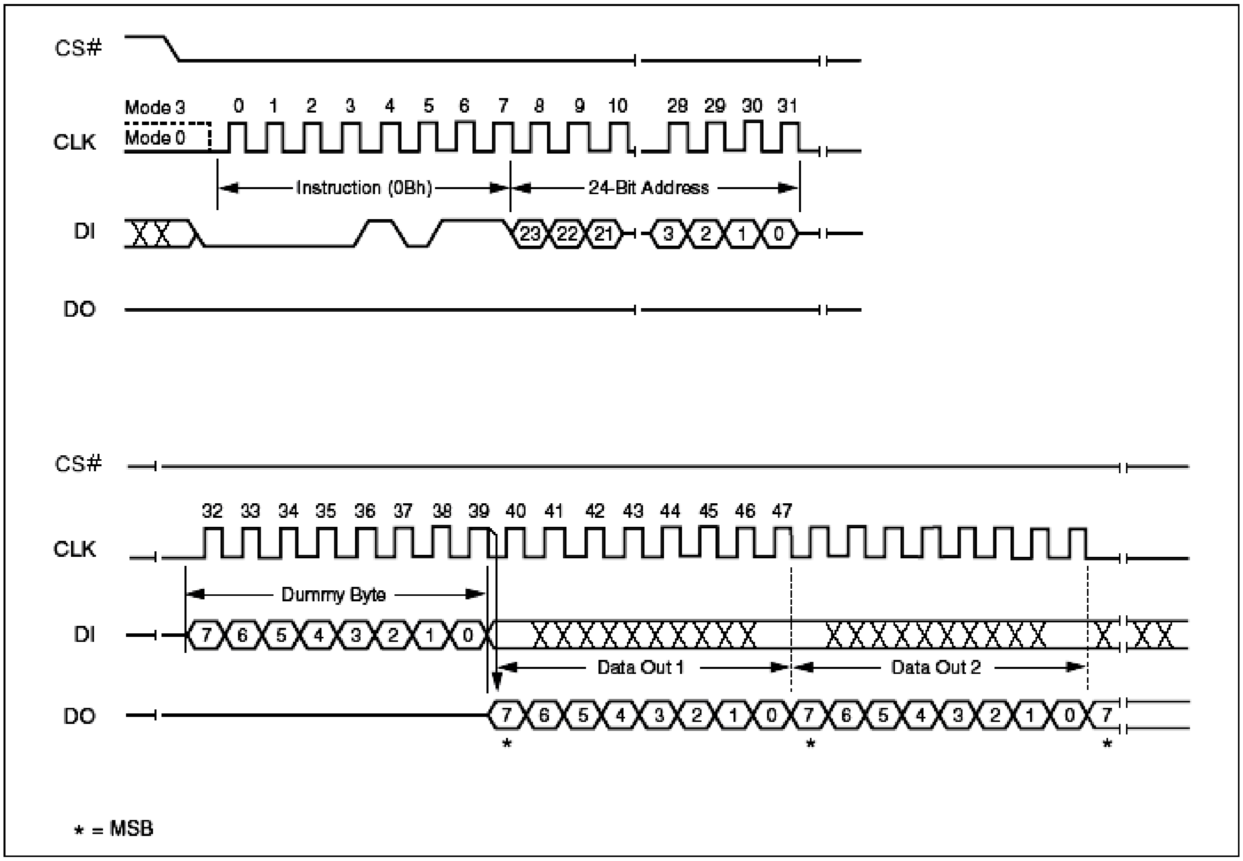


Figure 16. Fast Read Instruction Sequence Diagram

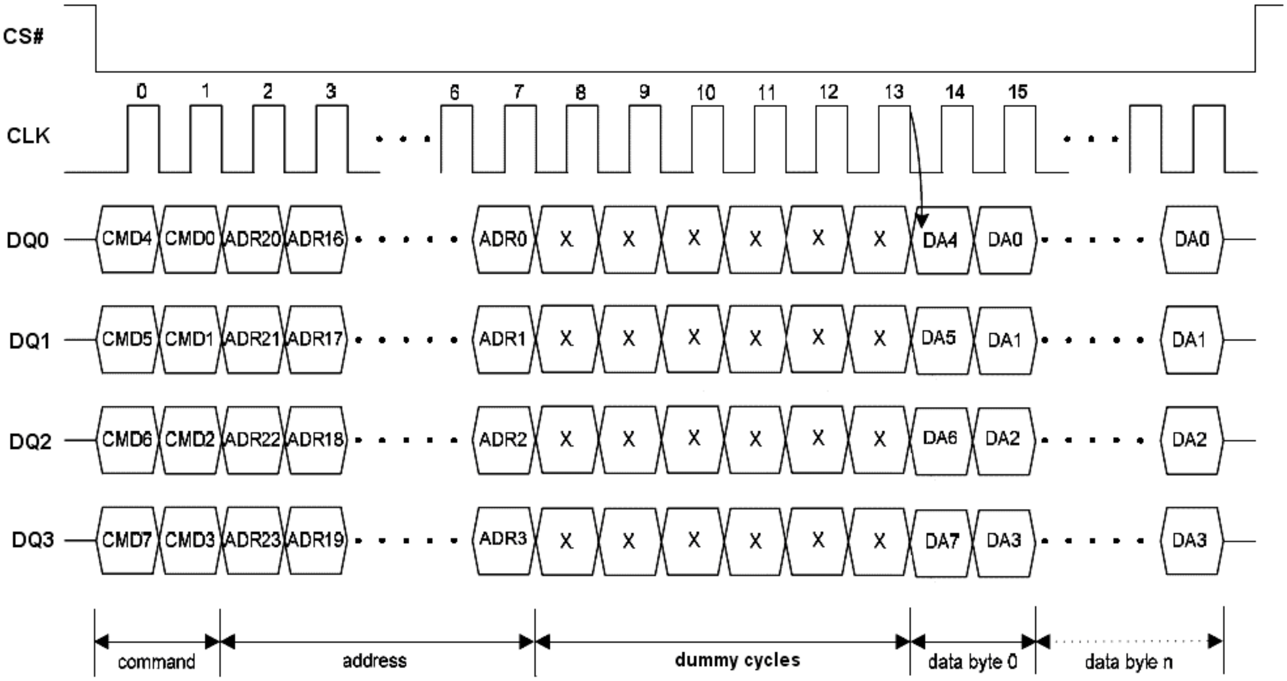


Figure 16.1 Fast Read Instruction Sequence in QPI Mode

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, DQ_0 and DQ_1, instead of just DQ_0. This allows data to be transferred from the device 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.

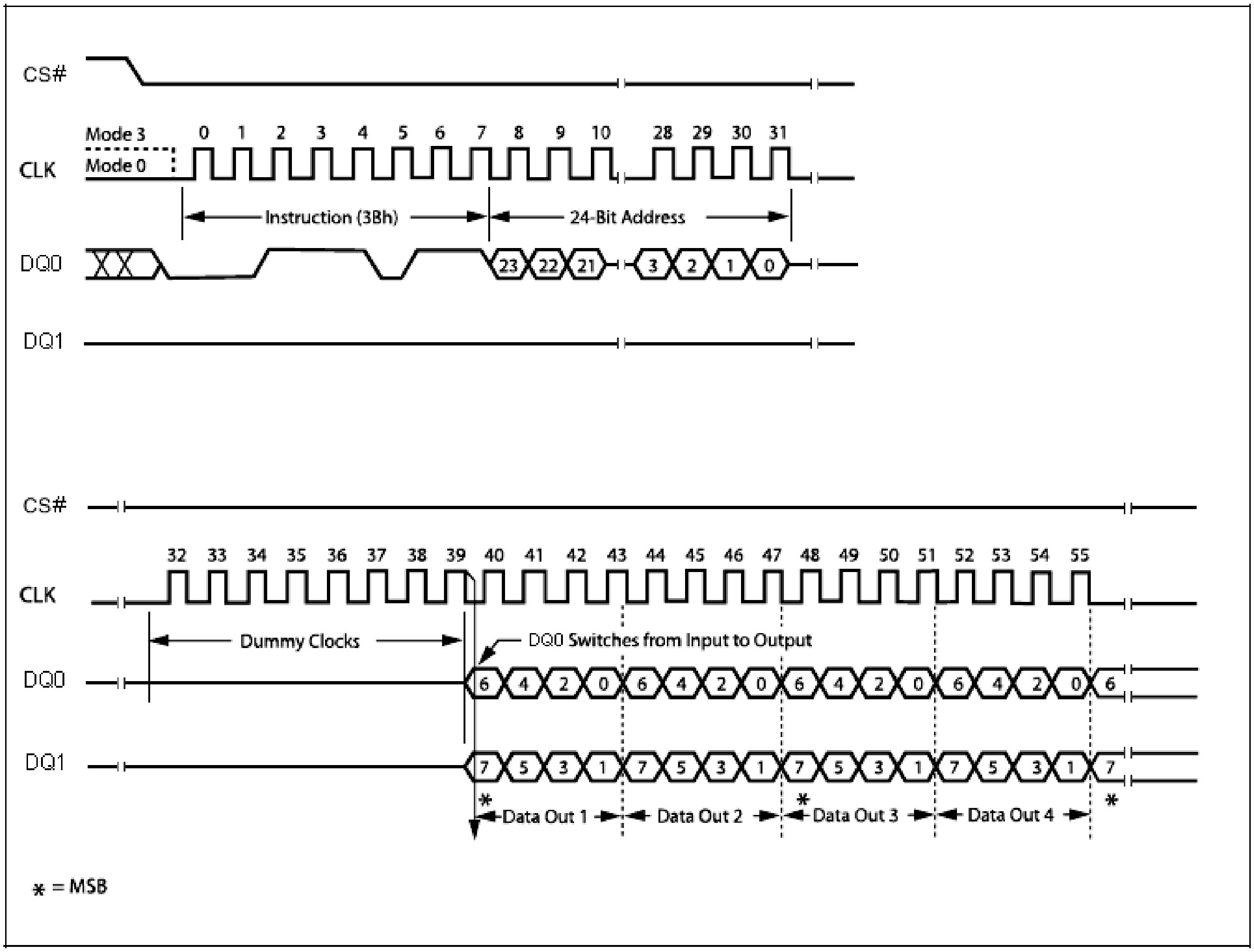


Figure 17. Dual Output Fast Read Instruction Sequence Diagram

Dual Input / Output FAST_READ (BBh)

The Dual I/O Fast Read (BBh) instruction allows for improved random access while maintaining two IO pins, DQ_0 and DQ_1. 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.

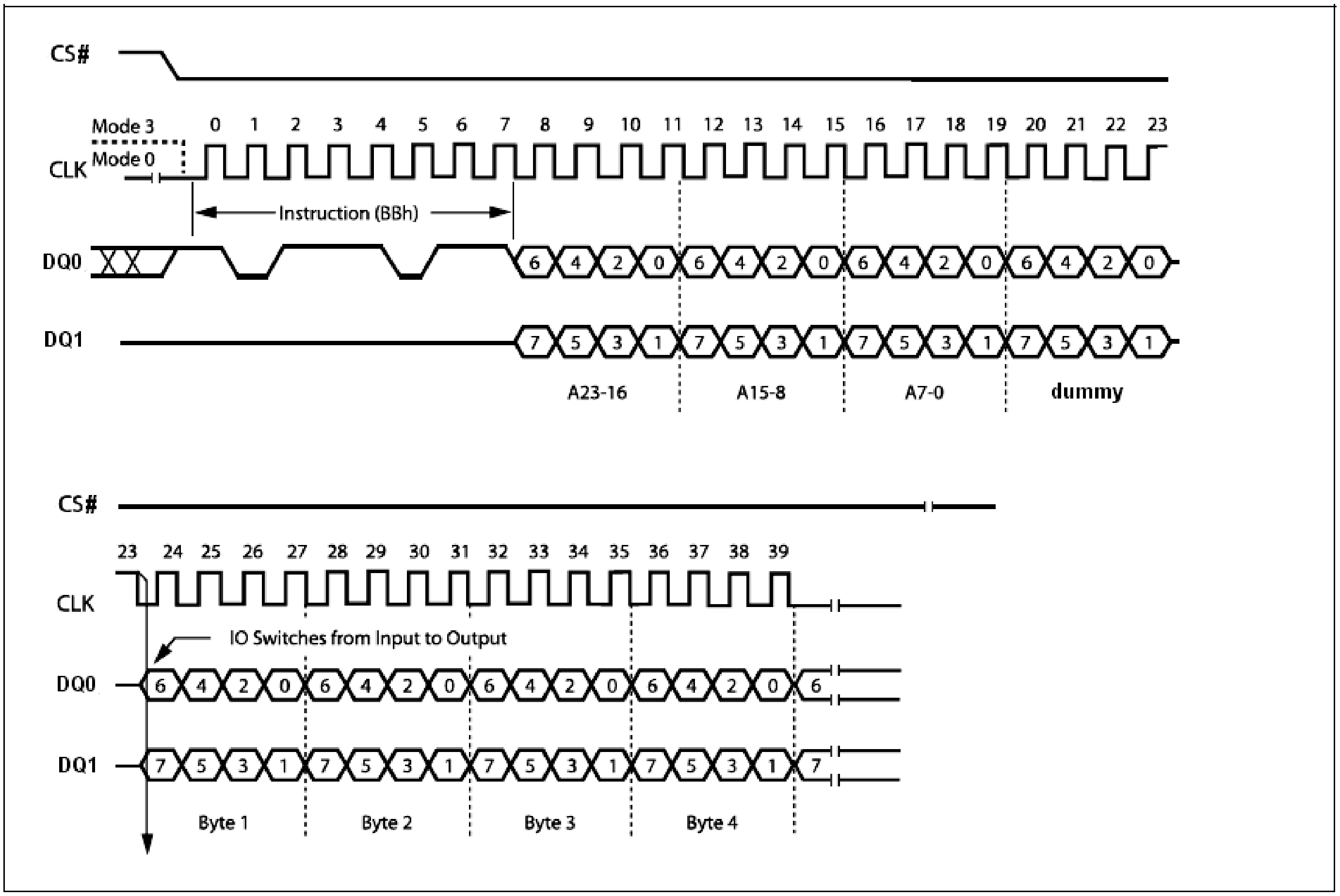


Figure 18. Dual Input / Output Fast Read Instruction Sequence Diagram

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, DQ_0, DQ_1, DQ_2 and DQ_3 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 DQ_0 -> 8 dummy clocks -> data out interleave on DQ_3, DQ_2, DQ_1 and DQ_0 -> 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 QE bit in Status Register is 0.

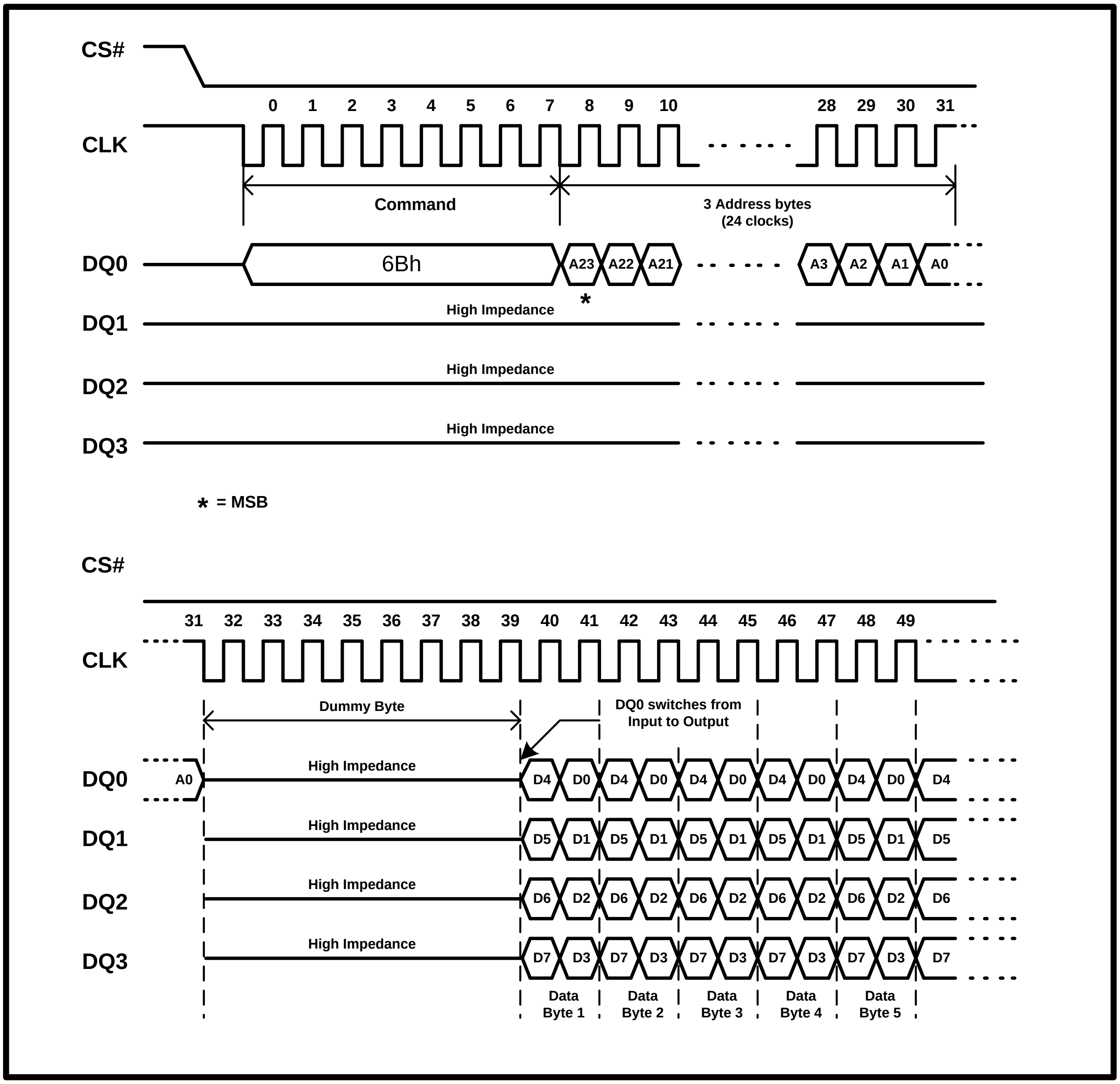


Figure 19. Quad Output Fast Read Instruction Sequence Diagram

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, DQ_0, DQ_1, DQ_2 and DQ_3 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 DQ_3, DQ_2, DQ_1 and DQ_0 -> 6 dummy clocks -> data out interleave on DQ_3, DQ_2, DQ_1 and DQ_0 -> 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.

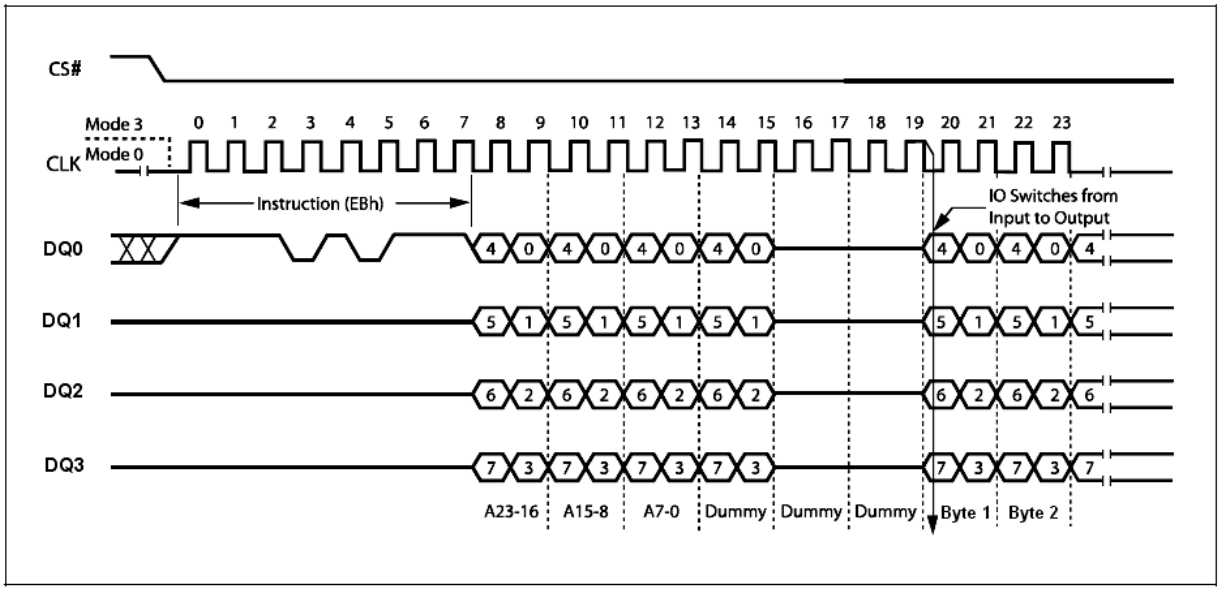


Figure 20. Quad Input / Output Fast Read Instruction Sequence Diagram

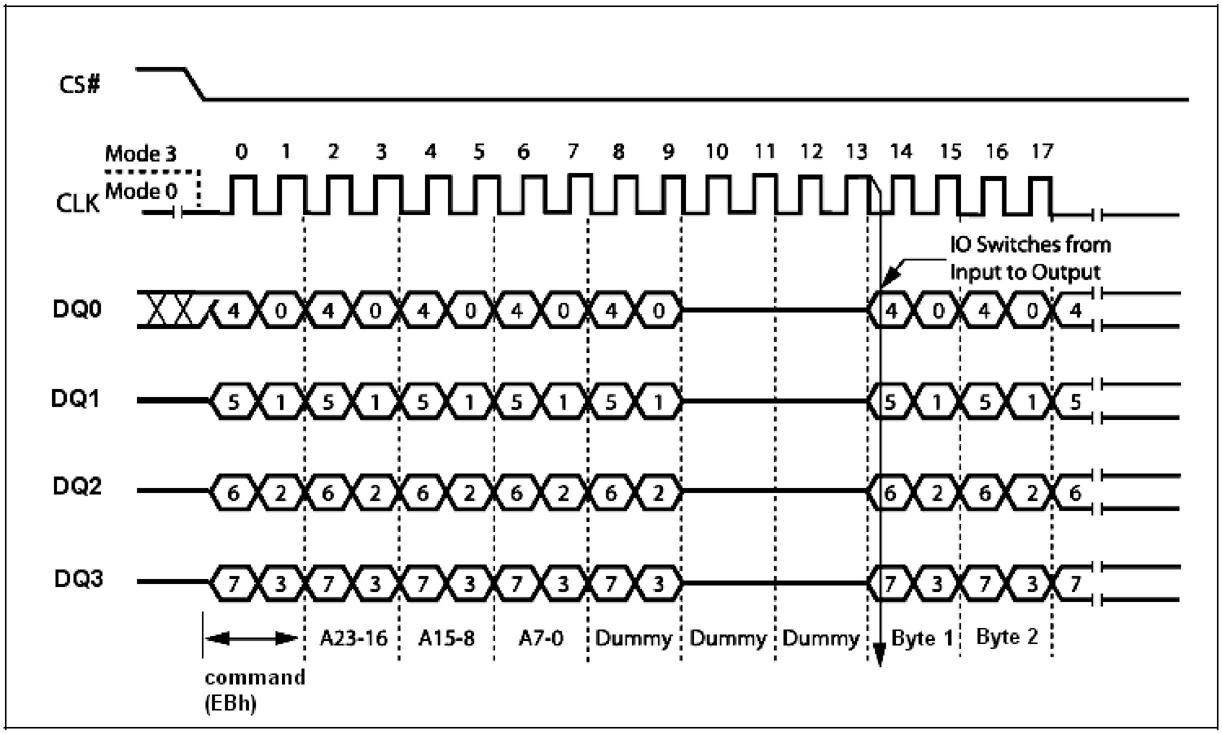


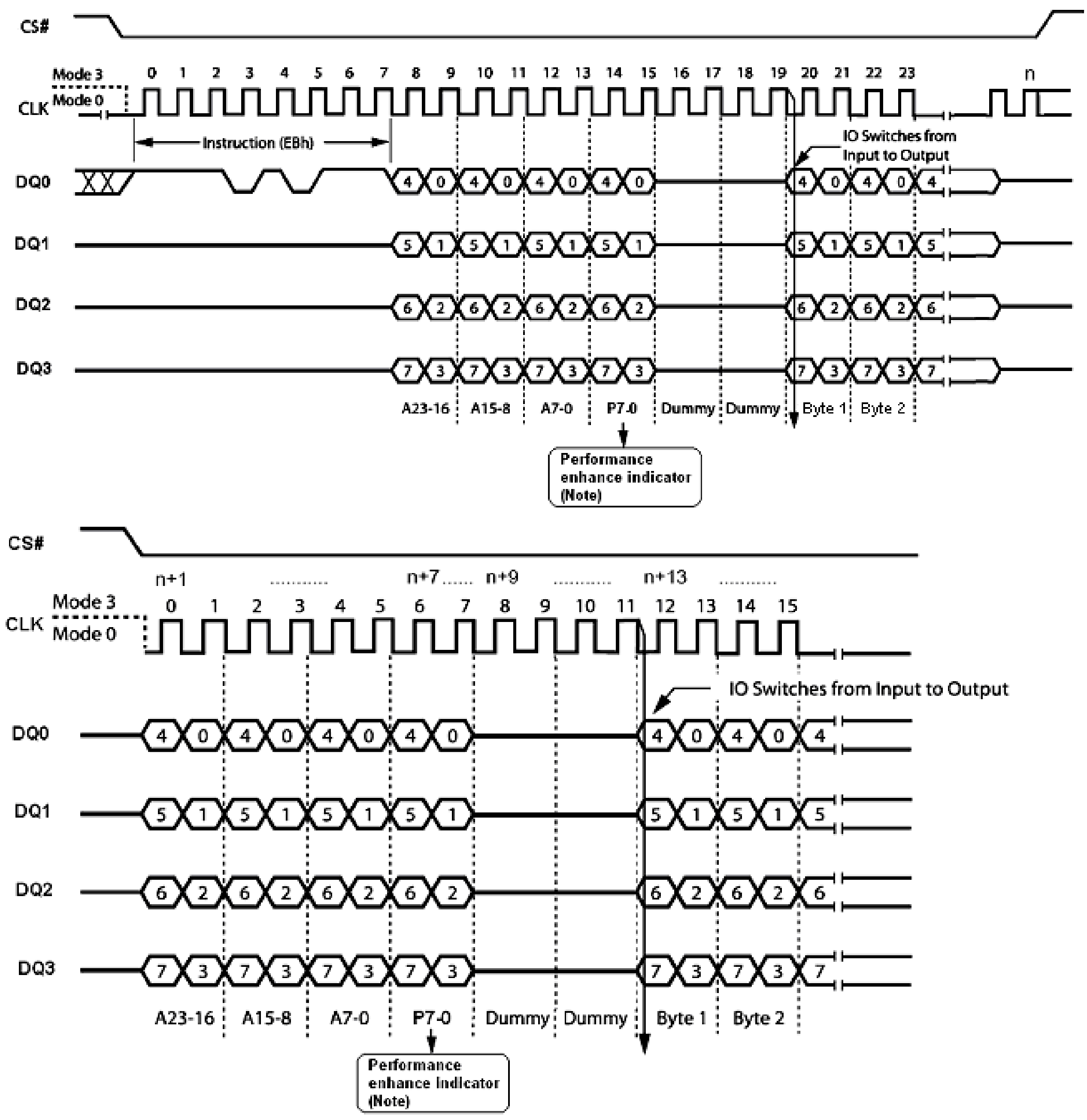
Figure 20.1. Quad Input / Output Fast Read Instruction Sequence in QPI Mode

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 DQ_3, DQ_2, DQ_1 and DQ_0 -> performance enhance toggling bit P[7:0] -> 4 dummy clocks -> data out interleave on DQ_3, DQ_2, DQ_1 and DQ_0 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 command (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.

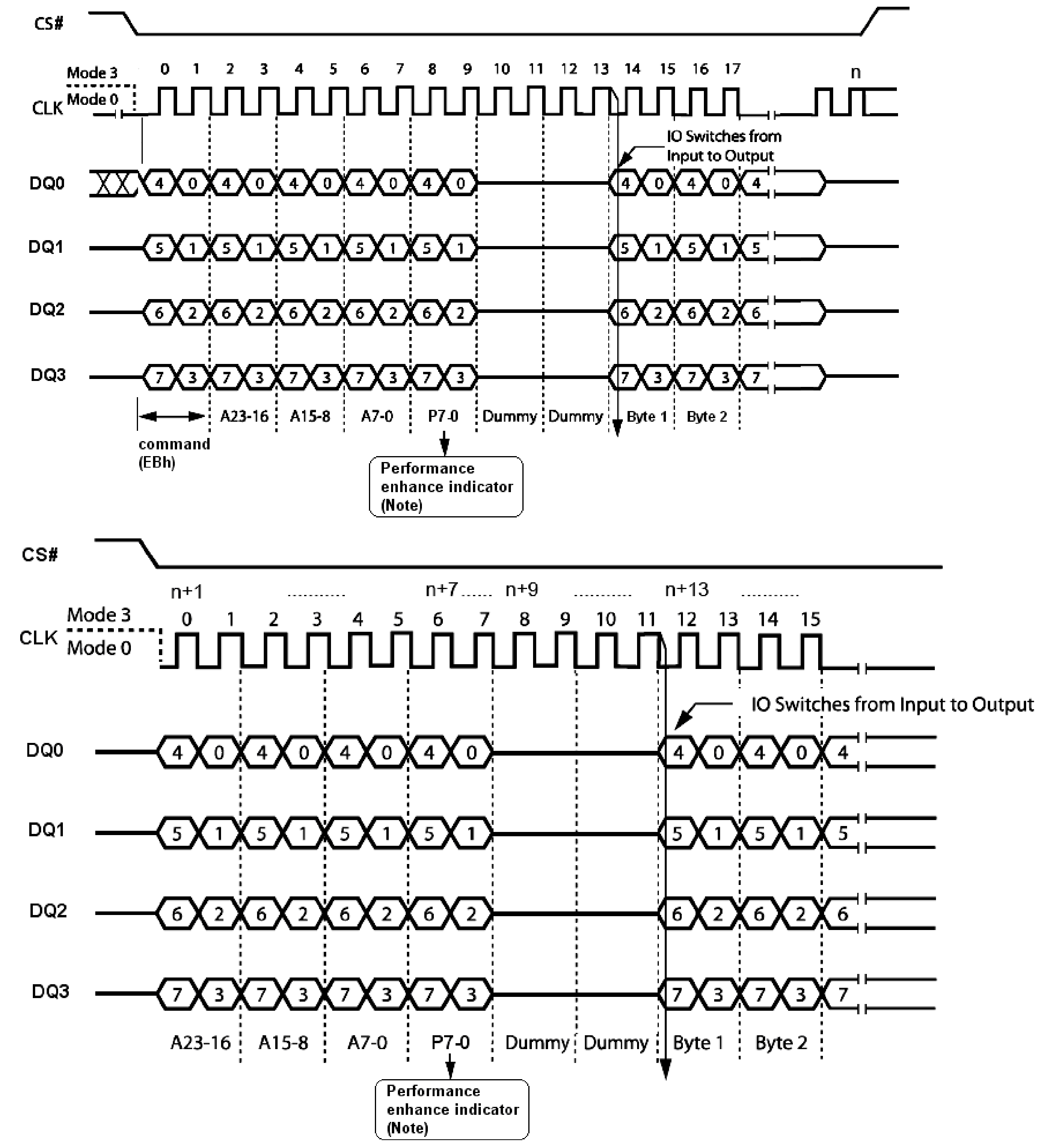
While Program/ Erase/ Write Status Register is in progress, Quad Input/ Output FAST_READ (EBh) instruction is rejected without impact on the Program/ Erase/ Write Status Register current cycle.

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



Note: Performance enhance mode, if P7 ≠ P3 & P6 ≠ P2 & P5 ≠ P1 & P4 ≠ P0 (Toggling), ex: A5, 5A, 0F
Reset the performance enhance mode, if P7 = P3 or P6 = P2 or P5 = P1 or P4 = P0, ex: AA, 00, FF

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



Note: Performance enhance mode, if P7 ≠ P3 & P6 ≠ P2 & P5 ≠ P1 & P4 ≠ P0 (Toggling), ex: A5, 5A, 0F
Reset the performance enhance mode, if P7 = P3 or P6 = P2 or P5 = P1 or P4 = P0, ex: AA, 00, FF

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

Read Burst (0Ch)

This device supports Read Burst with wrap in both SPI and QPI mode. To execute a Read Burst with wrap operation the host drivers CS# low, and sends the Read Burst with wrap (0Ch) command cycle, followed by three address bytes and one dummy byte (8 clocks) in SPI mode (Figure 22) or default three dummy byte (6 clocks) in QPI mode (Figure 22.1).

After the dummy byte, the device outputs data on the falling edge of the CLK signal starting from the specific address location. The data output stream is continuous through all addresses until terminated by a low-to high transition of CS# signal.

During Read Burst, the internal address point automatically increments until the last byte of the burst reached, then jumps to first byte of the burst. All bursts are aligned to addresses within the burst length, see Table 11. For example, if the burst length is 8 bytes, and the start address is 06h, the burst sequence should be: 06h, 07h, 00h, 01h, 02h, 03h, 04h, 05, 06, etc. The pattern would repeat until the command was terminated by pulling CS# as high status.

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

Table 11. Burst Address Range

Burst length	Burst wrap (A[7:A0]) address range
8 Bytes (default)	00-07H, 08-0FH, 10-17H, 18-1FH...
16 Bytes	00-0FH, 10-1FH, 20-2FH, 30-3FH...
32 Bytes	00-1FH, 20-3FH, 40-5FH, 60-7FH...
64 Bytes	00-3FH, 40-7FH, 80-BFH, C0-FFH

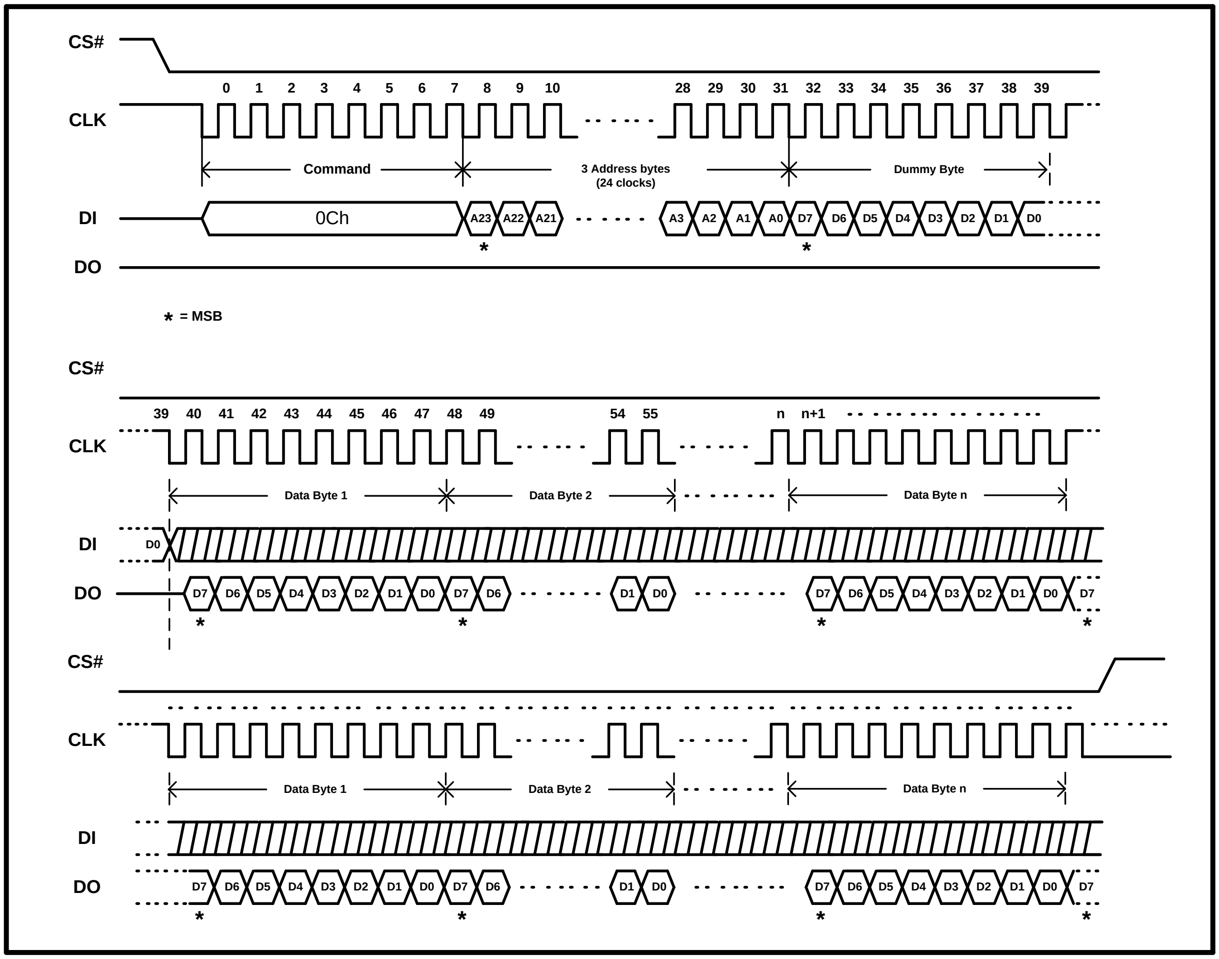


Figure 22. Read Burst Instruction Sequence Diagram

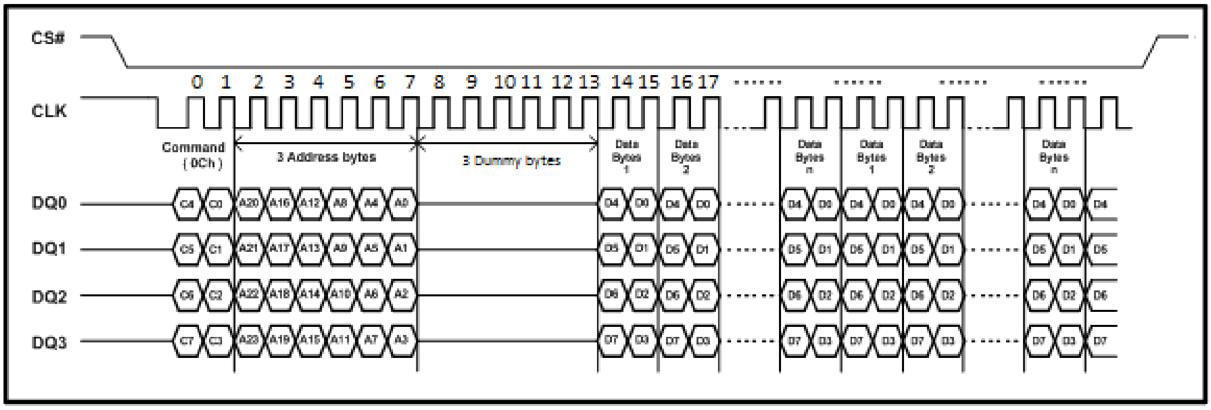


Figure 22.1 Read Burst Instruction Sequence Diagram in QPI mode

Write Status Register 2 (31h/01h)

The Write Status Register 2(31h) command can be used to set SPL0/SPL1/SPL2 OTP bits , QE bit and CMP bit. To set these bits to the host driver CS# low, sends the Write Status Register 2(31h) and one data byte, then drivers CS# high, In QPI mode, a cycle is two nibbles, or two clocks, long, most significant nibble first.
01h(WRSR) command also can set status register2.

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

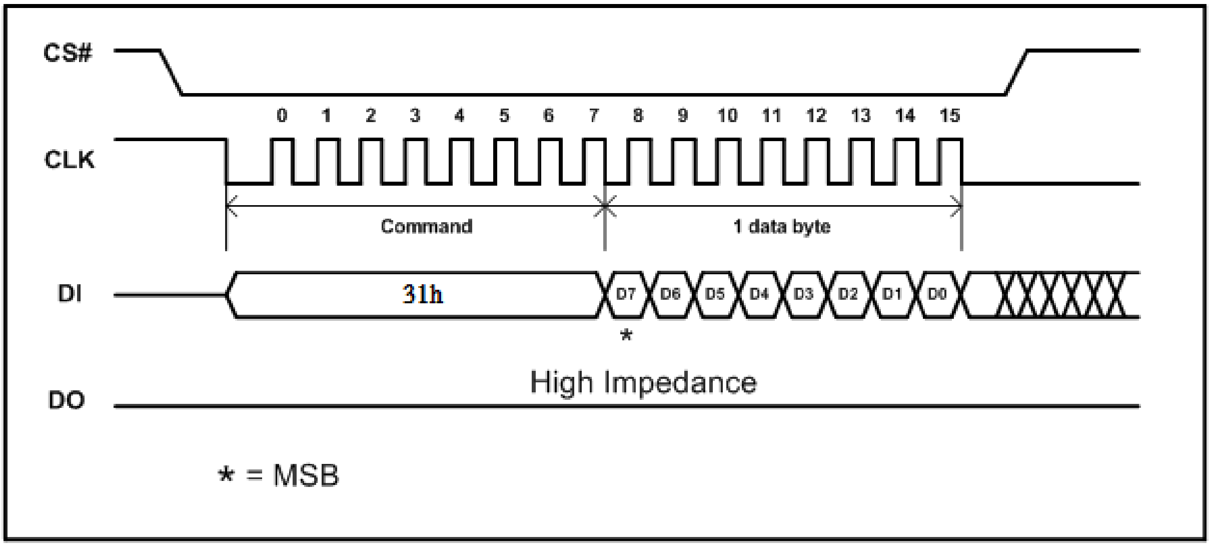
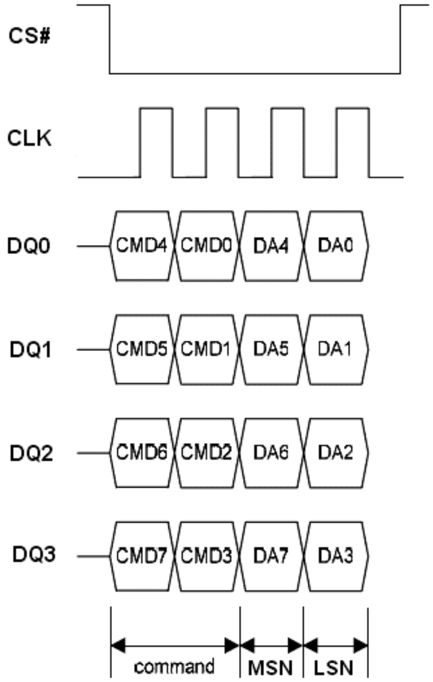


Figure 23. Write Status Register 2 Instruction Sequence Diagram



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

Figure 23.1 Write Status Register 2 Instruction Sequence Diagram in QPI mode

Write Status Register 3 (C0h/11h/01h)

The Write Status Register 3 (C0h/11h) command can be used to set output drive strength in I/O pins, HOLD/RESET# selection and burst read length setting. To set these bits to the host driver CS# low, sends the Write Status Register 3 (C0h or 11h) and one data byte, then drivers CS# high, In QPI mode, a cycle is two nibbles, or two clocks, long, most significant nibble first.
01h(WRSR) command also can set status register3.

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

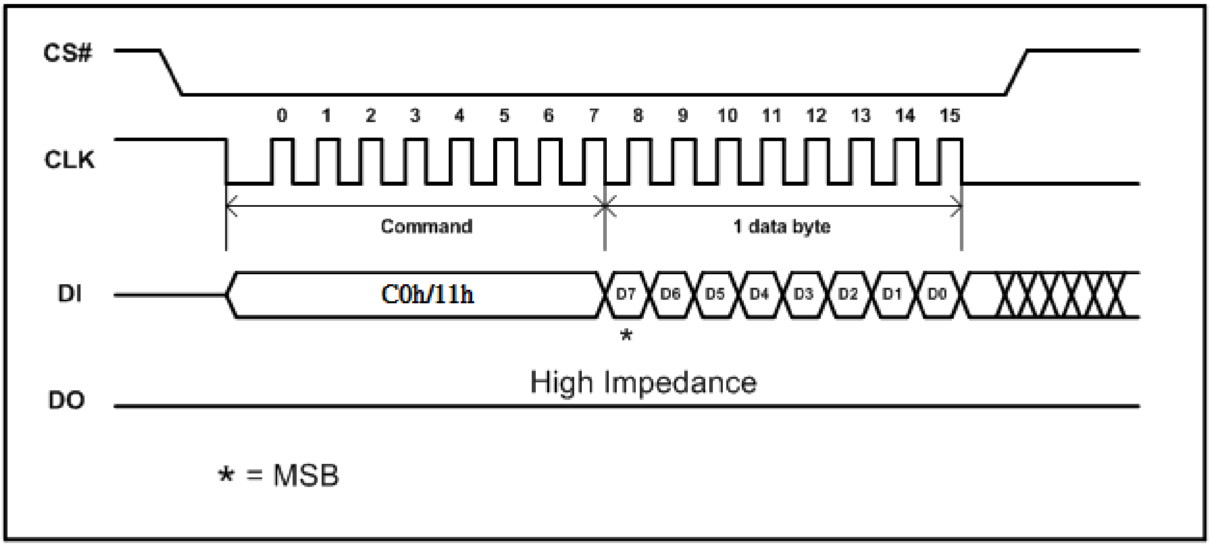
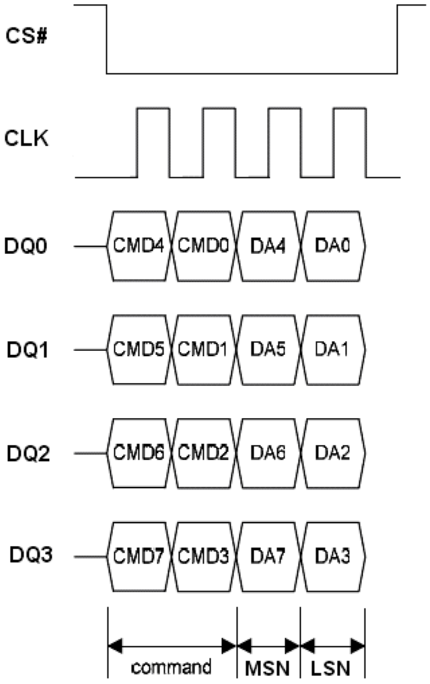


Figure 23. Write Status Register 3 Instruction Sequence Diagram



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

Figure 23.1 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 24. 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 24.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

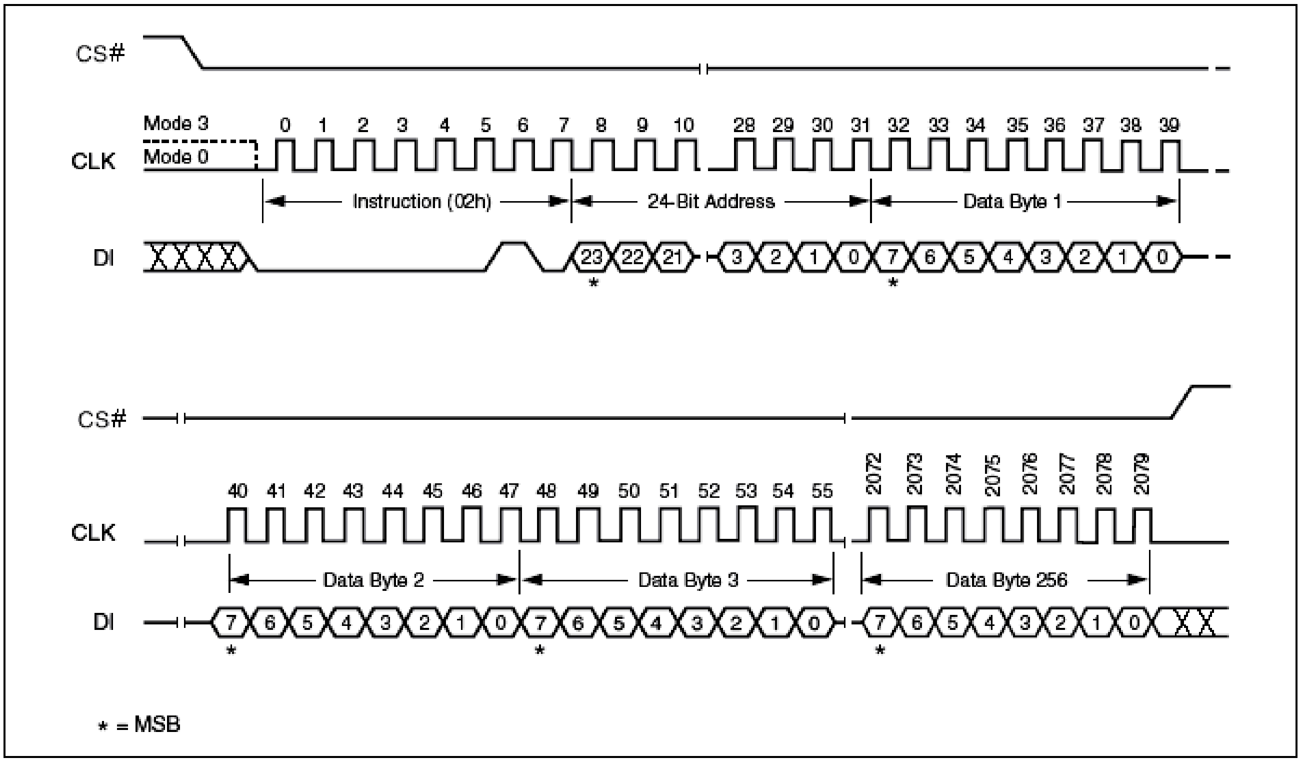


Figure 24. Page Program Instruction Sequence Diagram

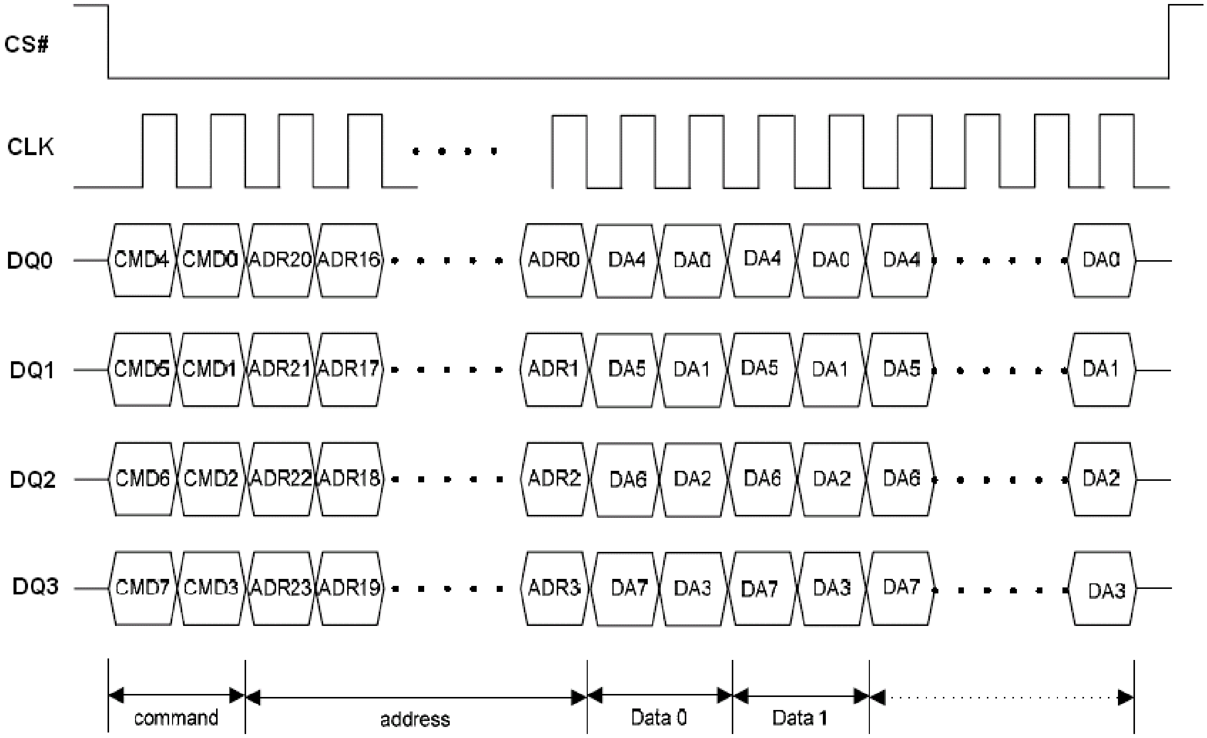


Figure 24.1 Program Instruction Sequence in QPI Mode

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: DQ_0, DQ_1, DQ_2 and DQ_3. 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 (QE) 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 25.

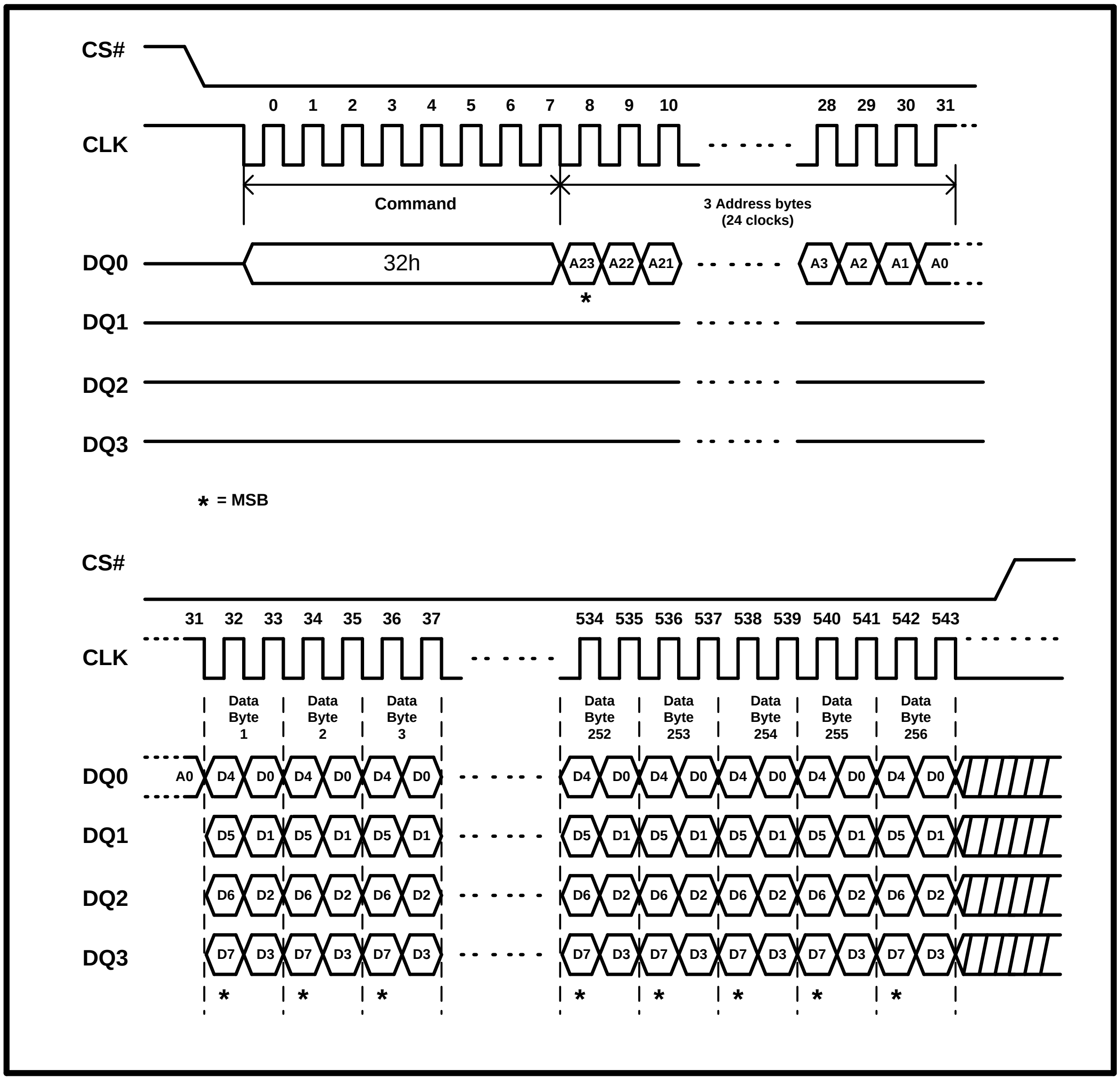


Figure 25. Quad Input Page Program Instruction Sequence Diagram (SPI Mode only)

Write Suspend (B0h/75h)

Write Suspend allows the interruption of Sector Erase, Block Erase or Page Program operations in order to erase, program, or read data in another portion of memory. The original operation can be continued with Write Resume command. The instruction sequence is shown in Figure 26.

Only one write operation can be suspended at a time; if an operation is already suspended, the device will ignore the Write Suspend command. Write Suspend during Chip Erase is ignored; Chip Erase is not a valid command while a write is suspended.

Suspend to suspend ready timing: 28us.
Resume to another suspend timing: min 0.3us. typ 200us.

Note:

User can use resume to another suspend minimum timing for issue next suspend after resume, but the device needs equal or longer typical time to make other progress after resume command.

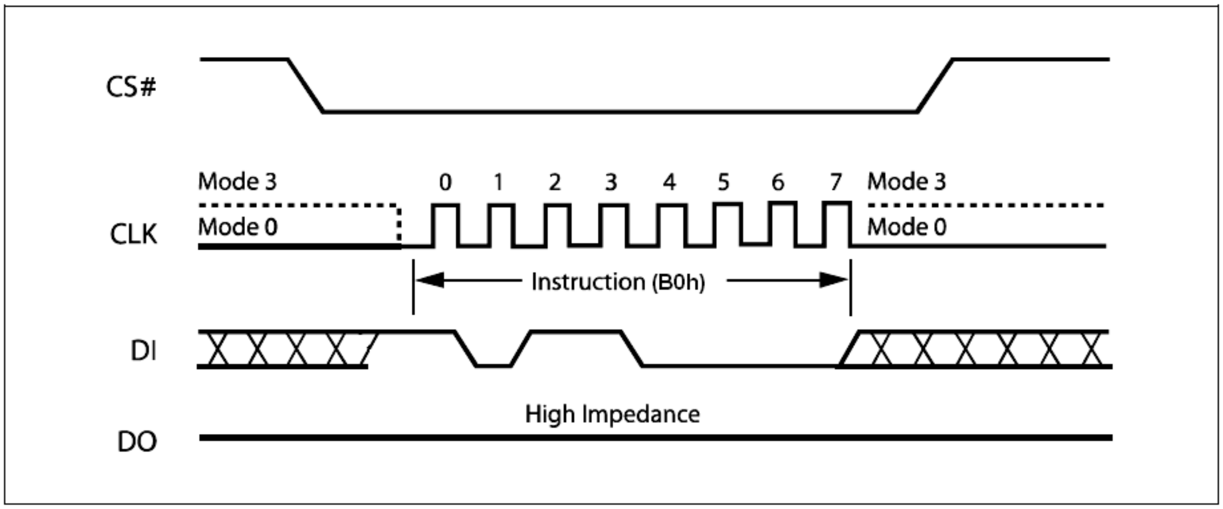


Figure 26. Write Suspend Instruction Sequence Diagram

Write Suspend During Sector Erase or Block Erase

Issuing a Write Suspend instruction during Sector Erase or Block Erase allows the host to program or read any block that was not being erased. The device will ignore any programming commands pointing to the suspended sector(s). Any attempt to read from the suspended sector(s) will output unknown data because the Sector or Block Erase will be incomplete.

To execute a Write Suspend operation, the host drives CS# low, sends the Write Suspend command cycle (B0h), then drives CS# high. A cycle is two nibbles long, most significant nibble first. The Suspend Status register indicates that the erase has been suspended by changing the WSE bit from “0” to “1”, but the device will not accept another command until it is ready. To determine when the device will accept a new command, poll the WIP bit in the Suspend Status register or after issue program suspend command, latency time 28us is needed before issue another command. For “Suspend to Read”, “Resume to Read”, “Resume to Suspend” timing specification please note Figure 26.1, 26.2 and 26.3.

Write Suspend During Page Programming

Issuing a Write Suspend instruction during Page Programming allows the host to erase any sector or read any page that is not being programmed. Erase commands pointing to the suspended sector(s) will be ignored. Any attempt to read from the suspended page will output unknown data because the program will be incomplete.

To execute a Write Suspend operation, the host drives CS# low, sends the Write Suspend command cycle (B0h), then drives CS# high. A cycle is two nibbles long, most significant nibble first. The Suspend Status register indicates that the programming has been suspended by changing the WSP bit from “0” to “1”, but the device will not accept another command until it is ready. To determine when the device will accept a new command, poll the WIP bit in the Suspend Status register or after issue program suspend command, latency time 28us is needed before issue another command. For “Suspend to Read”, “Resume to Read”, “Resume to Suspend” timing specification please note Figure 26.1, 26.2 and 26.3.

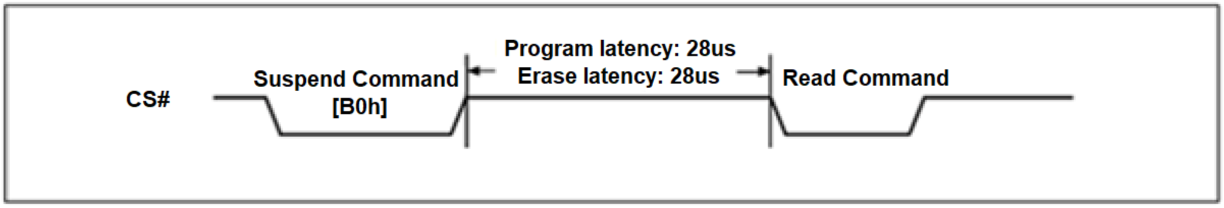


Figure 26.1 Suspend to Read Latency

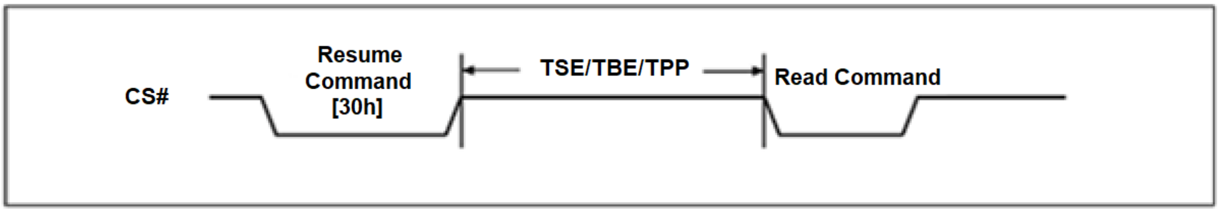


Figure 26.2 Resume to Read Latency

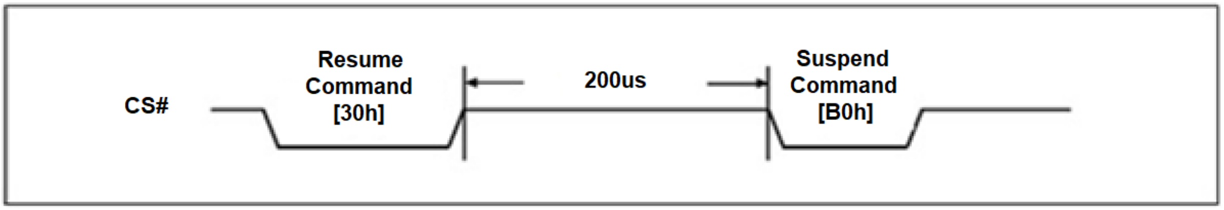


Figure 26.3 Resume to Suspend Latency

The instruction sequence is shown in Figure 27.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

Write Resume (30h/7Ah)

Write Resume restarts a Write command that was suspended, and changes the suspend status bit in the Status register 2(WSE or WSP) back to "0".

The instruction sequence is shown in Figure 27. To execute a Write Resume operation, the host drives CS# low, sends the Write Resume command cycle (30h), then drives CS# high. A cycle is two nibbles long, most significant nibble first. To determine if the internal, self-timed Write operation completed, poll the WIP bit in the Suspend Status register, or wait the specified time t_{SE}, t_{HBE}, t_{BE} or t_{PP} for Sector Erase, Block Erase, or Page Programming, respectively. The total write time before suspend and after resume will not exceed the uninterrupted write times t_{SE}, t_{HBE}, t_{BE} or t_{PP}. Resume to another suspend operation requires latency time of 200us.

The instruction sequence is shown in Figure 27.1 while using the Enable Quad Peripheral Interface mode (EQPI) (38h) command.

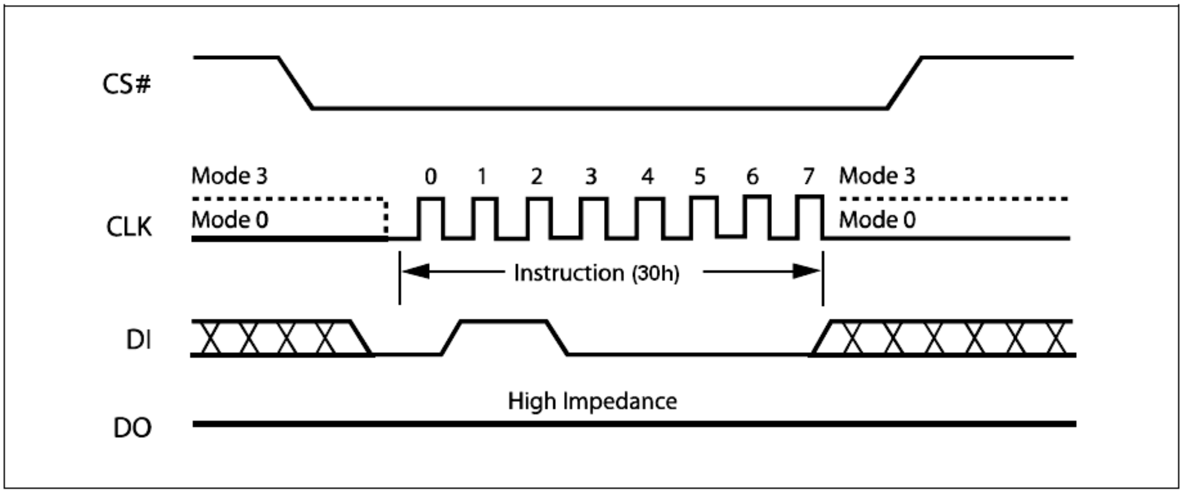


Figure 27. Write Resume Instruction Sequence Diagram

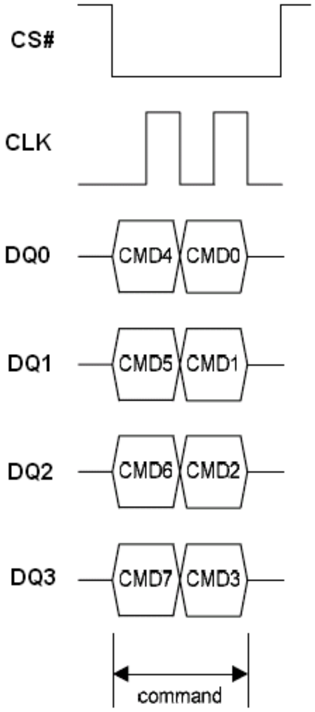


Figure 27.1. Write Suspend/Resume Instruction Sequence in QPI Mode

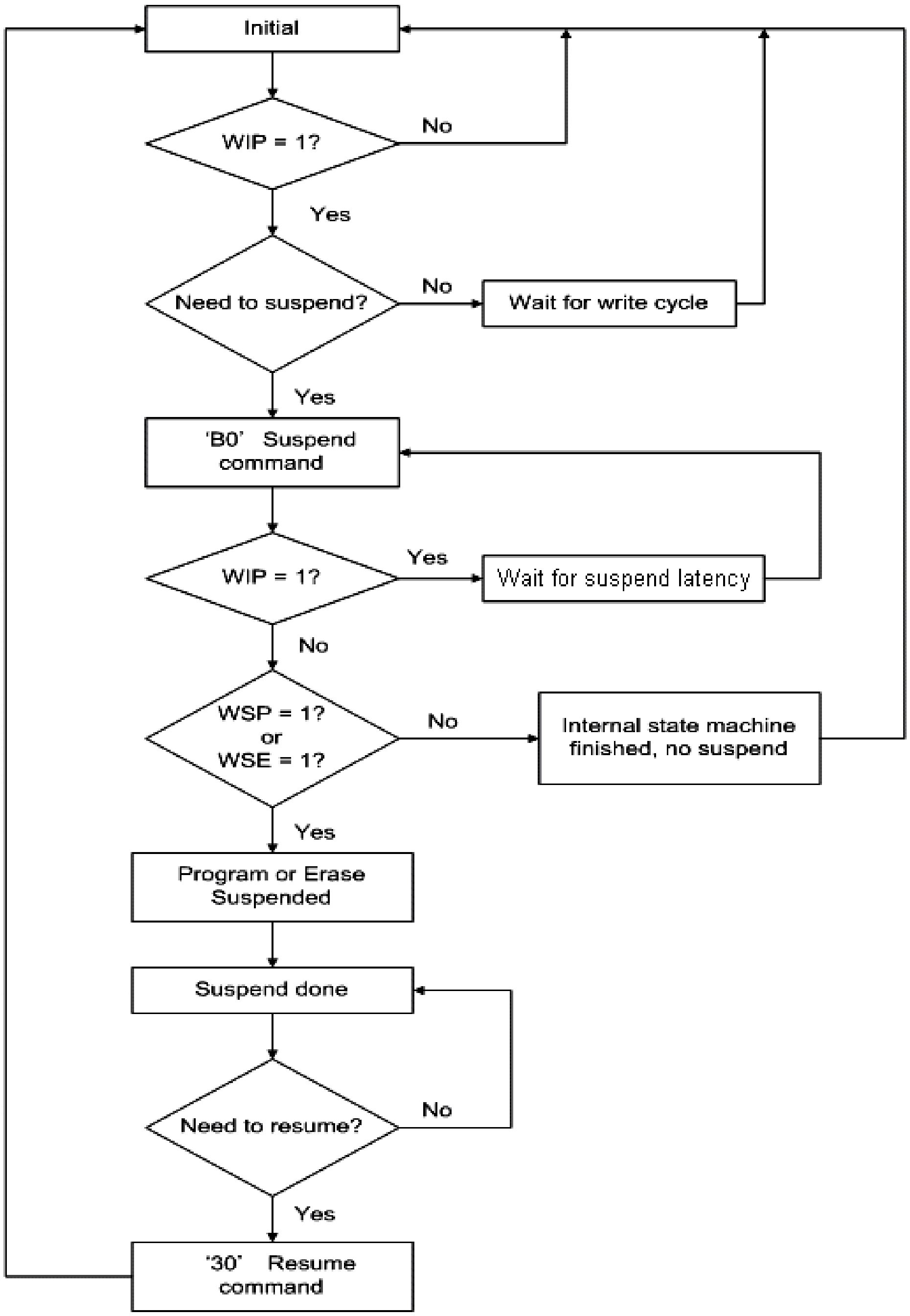


Figure 27.2. Write Suspend/Resume Flow

Note:

1. The 'WIP' can be either checked by command '09'or '05' polling.
2. 'Wait for write cycle' can be referring to maximum write cycle time or polling the WIP.
3. 'Wait for suspend latency', after issue program suspend command, latency time 28us is needed before issue another command or polling the WIP.
4. The 'WSP' and 'WSE' can be checked by command '09' polling.
5. 'Suspend done' means the chip can do further operations allowed by suspend spec.

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.

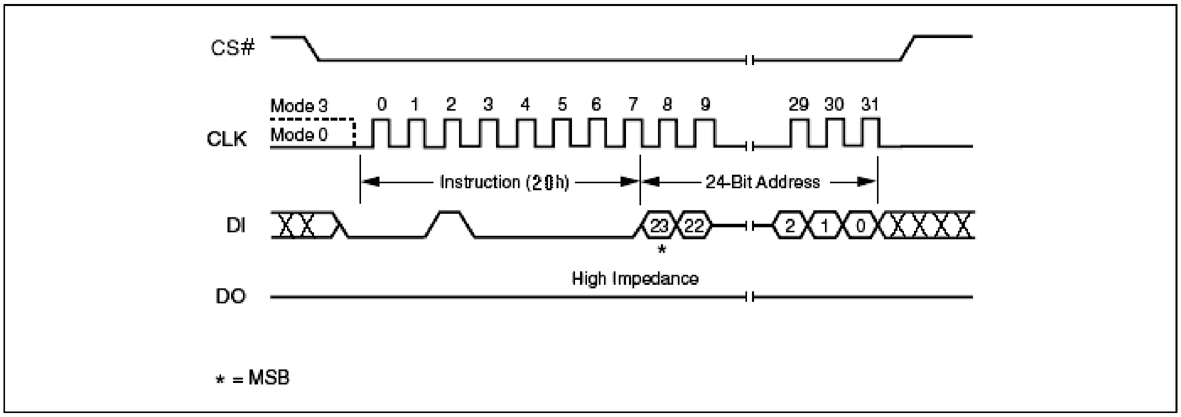


Figure 28. Sector Erase Instruction Sequence Diagram

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.

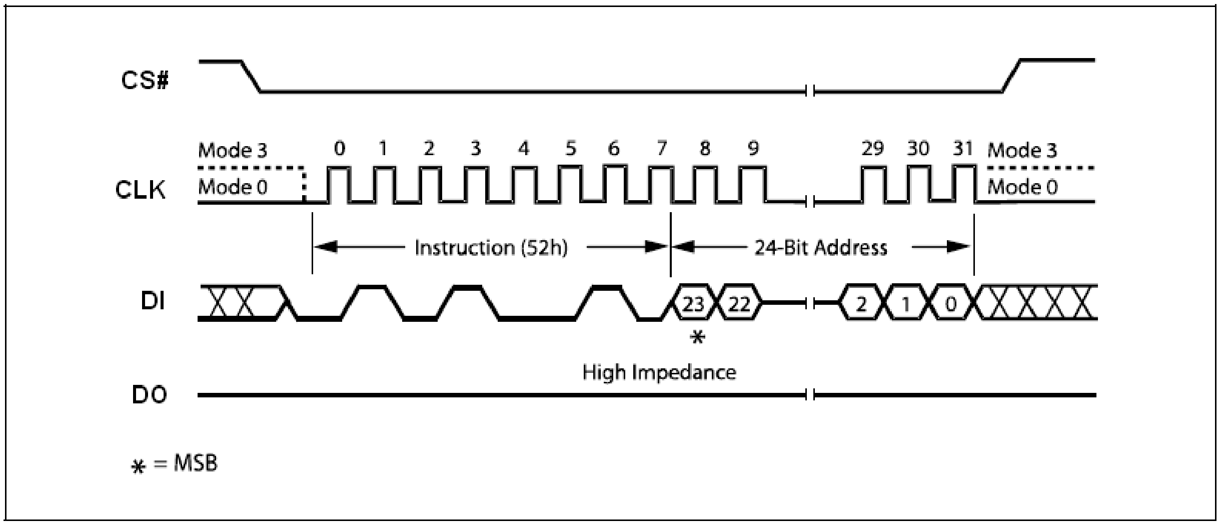


Figure 29. 32KB Half Block Erase Instruction Sequence Diagram

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.

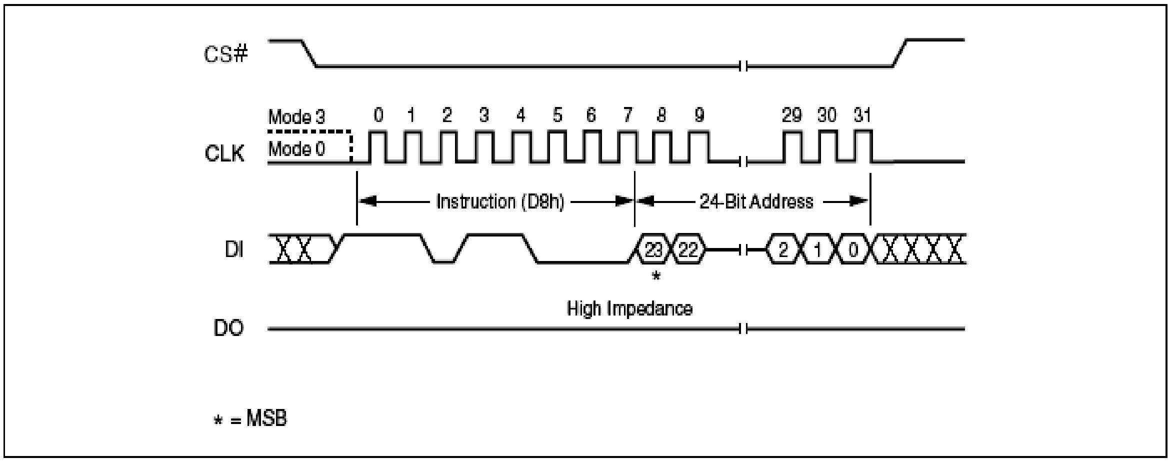


Figure 30. 64K Block Erase Instruction Sequence Diagram

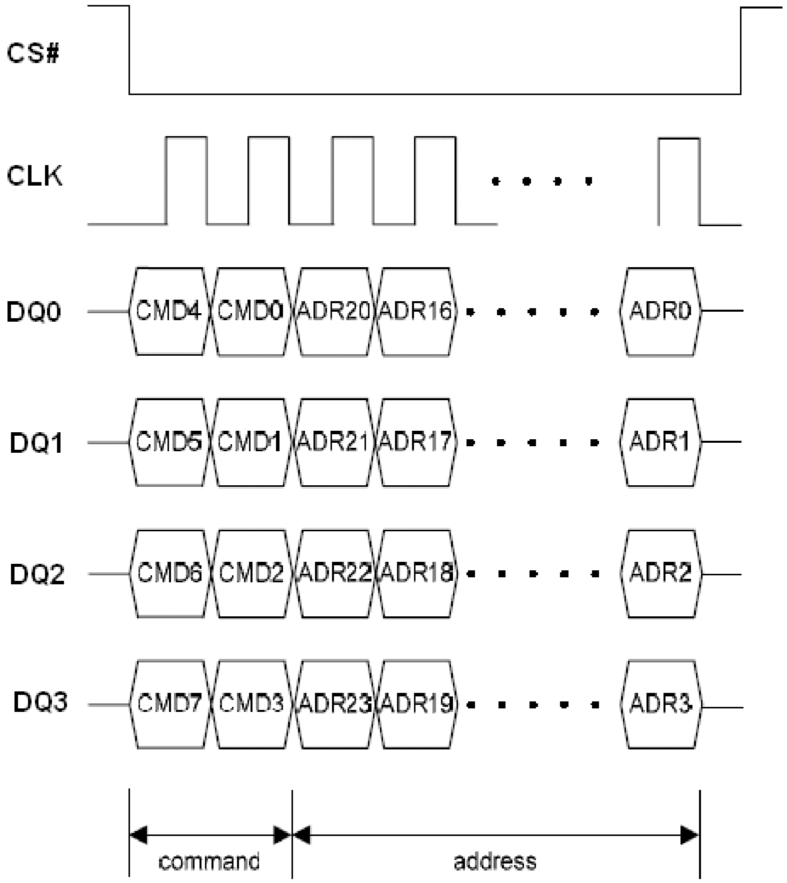


Figure 30.1 Block/Sector Erase Instruction Sequence in QPI Mode

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. 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.

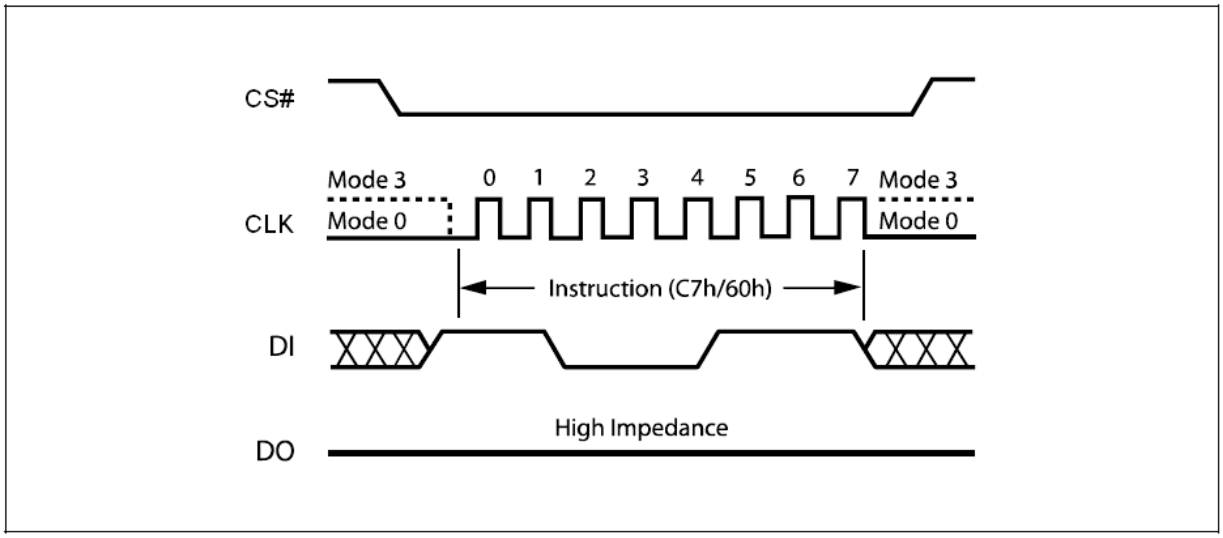


Figure 31. Chip Erase Instruction Sequence Diagram

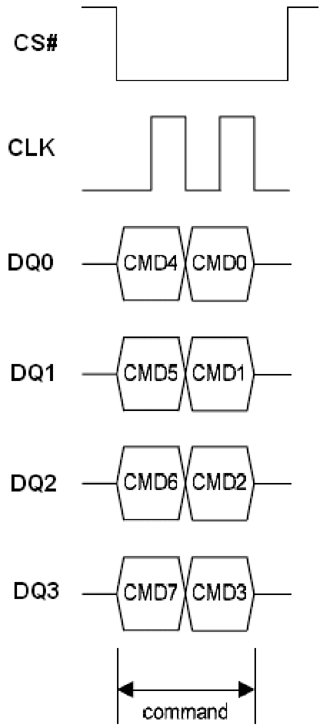


Figure 31.1 Chip Erase Sequence in QPI Mode

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.

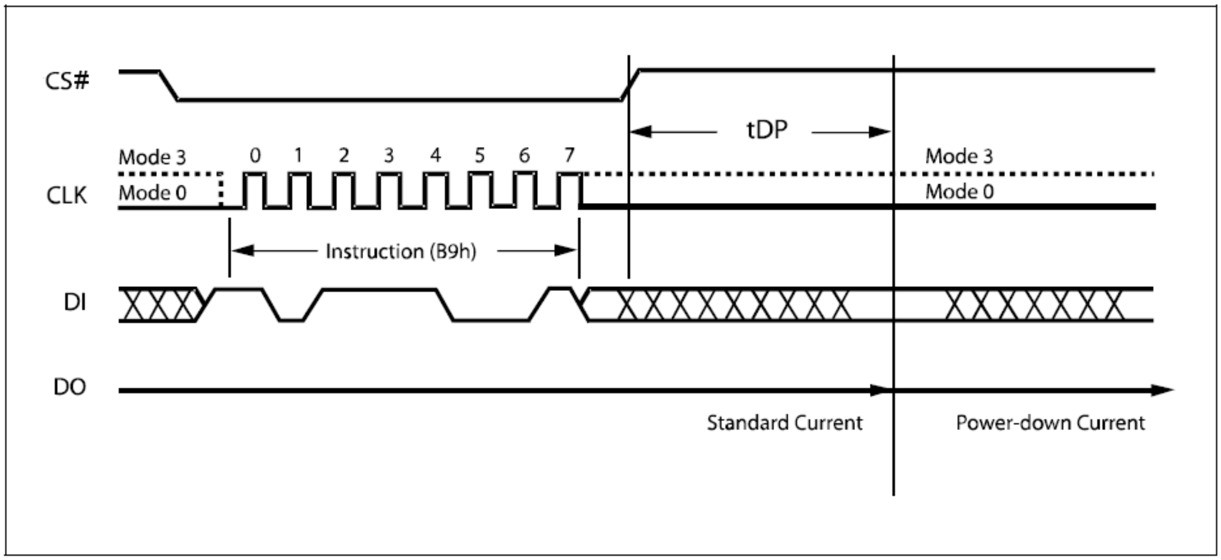


Figure 32. Deep Power-down Instruction Sequence Diagram

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 device 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.

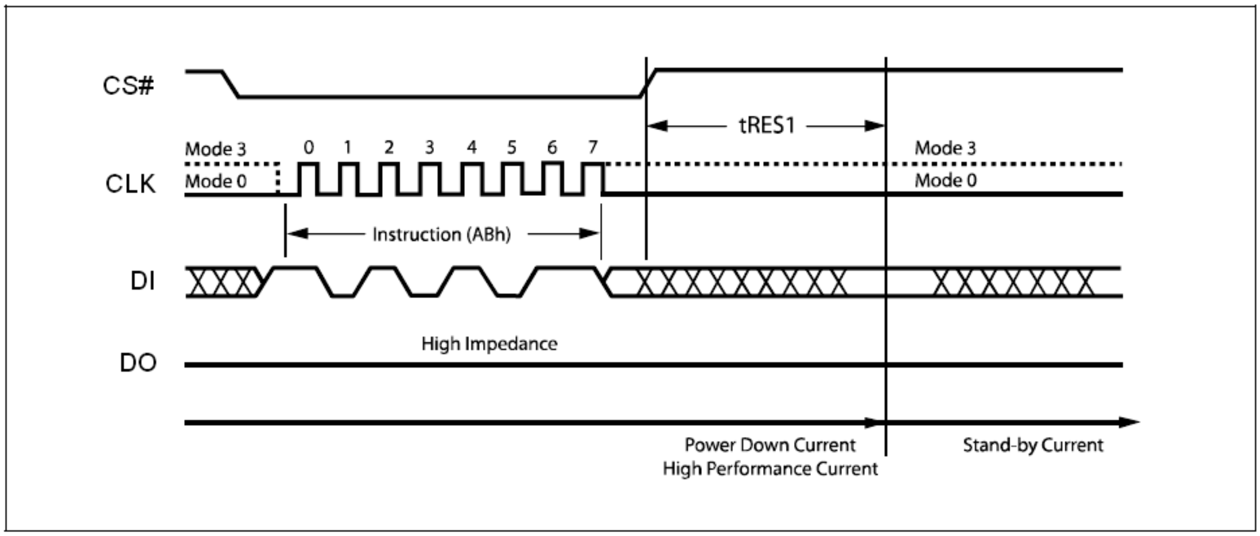


Figure 33. Release Power-down Instruction Sequence Diagram

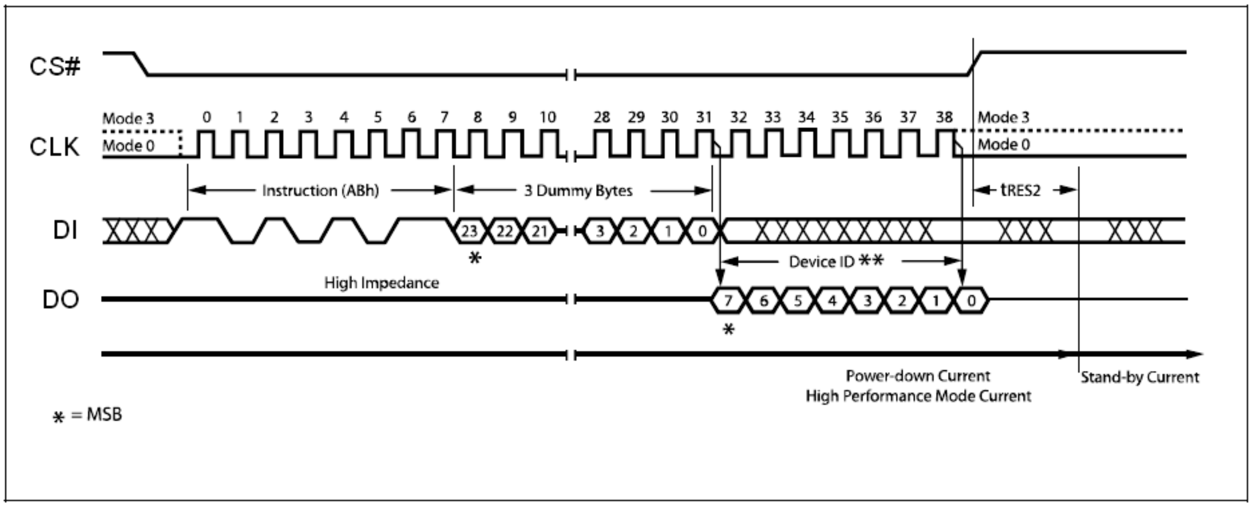


Figure 34. Release Power-down / Device ID Instruction Sequence Diagram

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 device 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.

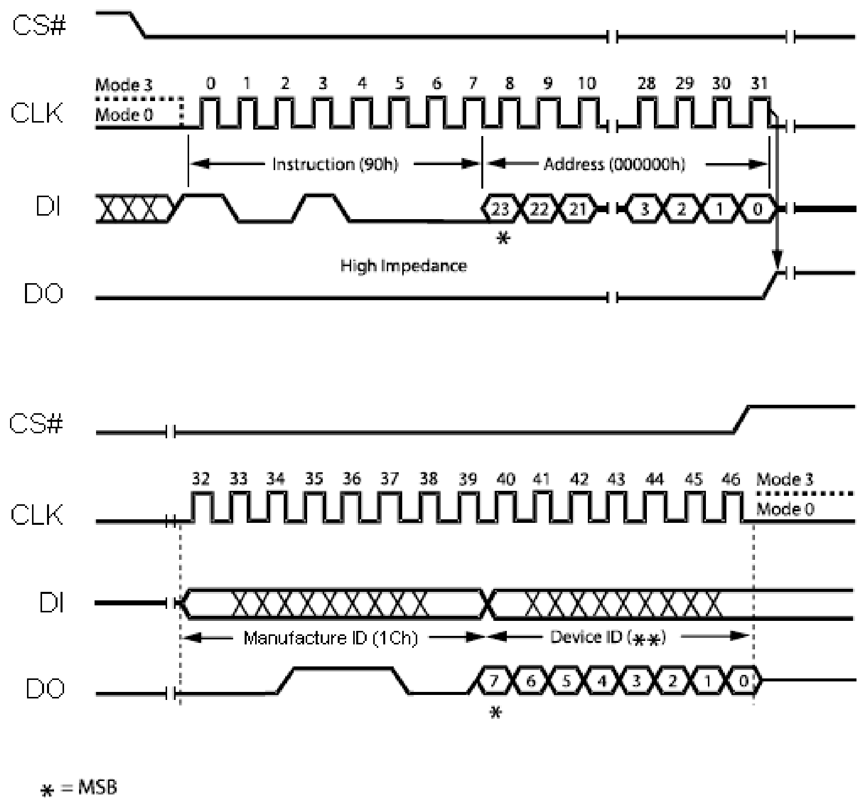


Figure 35. Read Manufacturer / Device ID Diagram

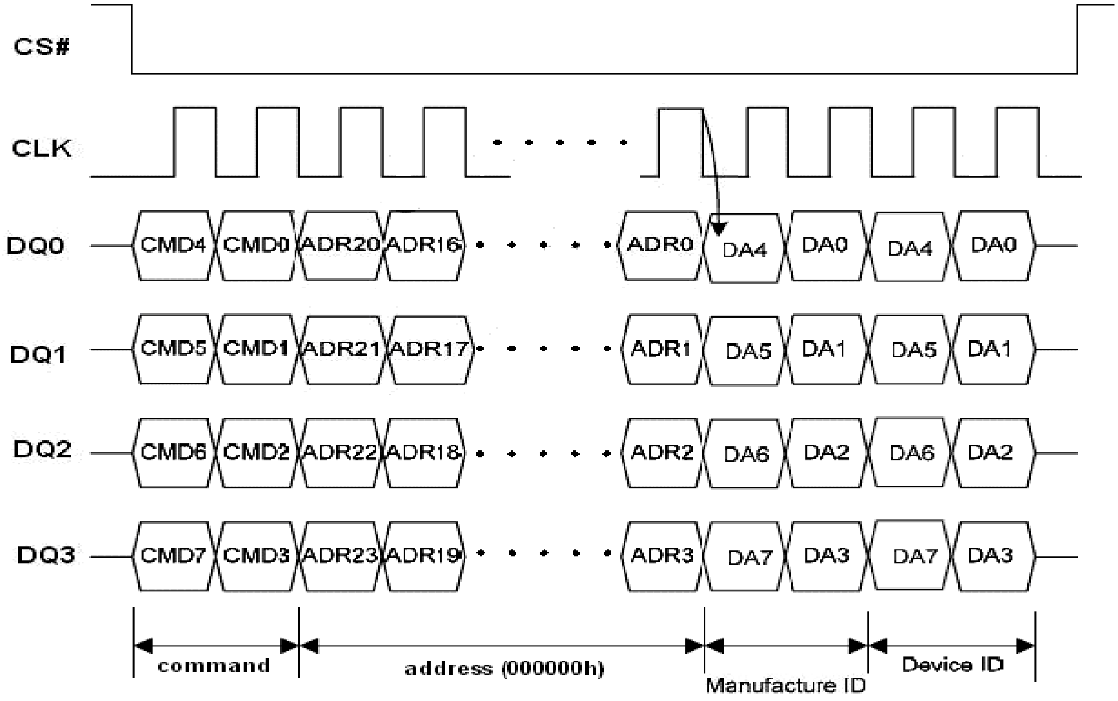


Figure 35.1. Read Manufacturer / Device ID Diagram in QPI Mode

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.

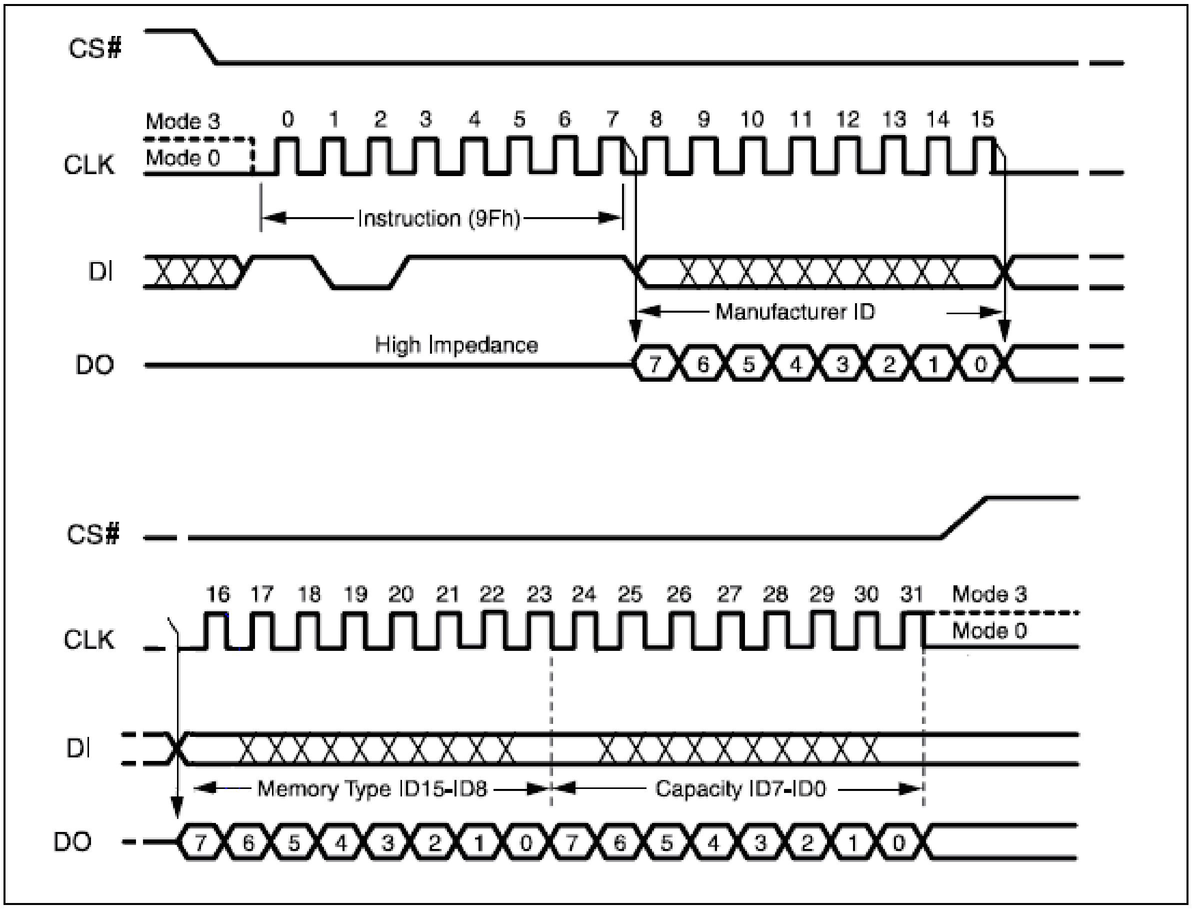


Figure 36. Read Identification (RDID)

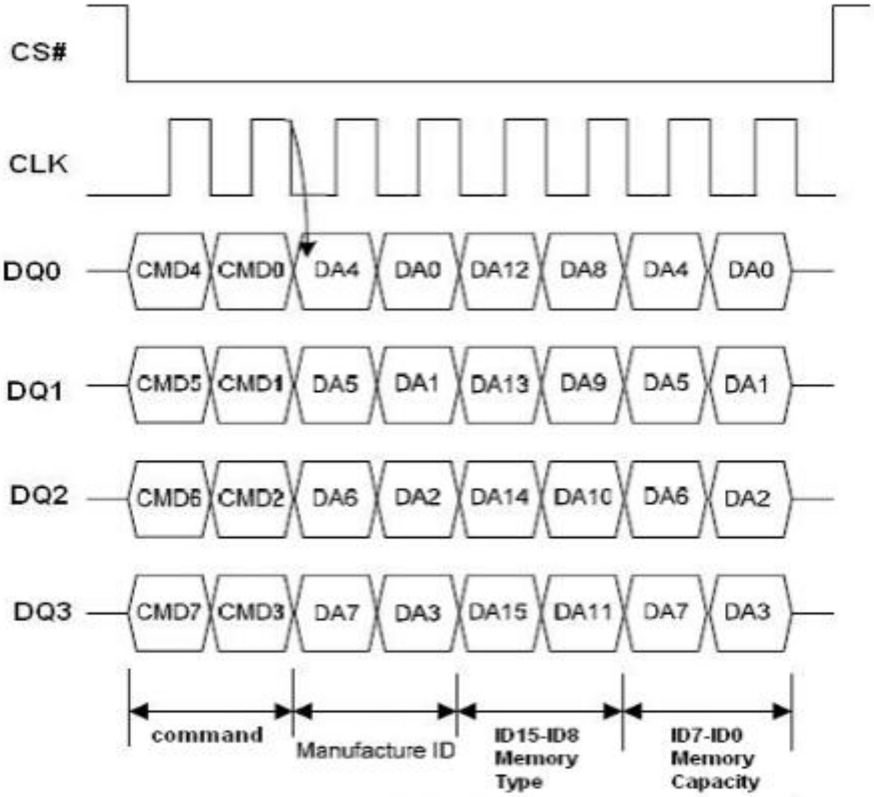


Figure 36.1. Read Identification (RDID) in QPI Mode

Program OTP array (42h)

The Program OTP array operation is similar to the Page Program instruction. It allows from one byte to 256 bytes of security register data to be programmed at previously erased (FFh) memory locations. A Write Enable instruction must be executed before the device will accept the Program OTP array Instruction. The instruction is initiated by driving the CS# pin low then shifting the instruction code “42h” followed by a 24-bit address (A23-A0) and at least one data byte, into the DI pin. The CS# pin must be held low for the entire length of the instruction while data is being sent to the device.

The Program OTP array instruction sequence is shown in Figure 46. The OTP array Lock Bits (SPL0-SPL3) in Status Register2 can be used to OTP protect the OTP array data. Once a lock bit is set to 1, the corresponding OTP array will be permanently locked, Program OTP array instruction to that register will be ignored.

This command also supports QPI mode.

Table 11. OTP Sector Address

Sector	Sector Size	Address Range
4095	512 byte	FFF000h – FFF1FFh
4094	512 byte	FFE000h – FFE1FFh
4093	512 byte	FFD000h – FFD1FFh

Note: The OTP sector is mapping to sector 4095, 4094 and 4093.

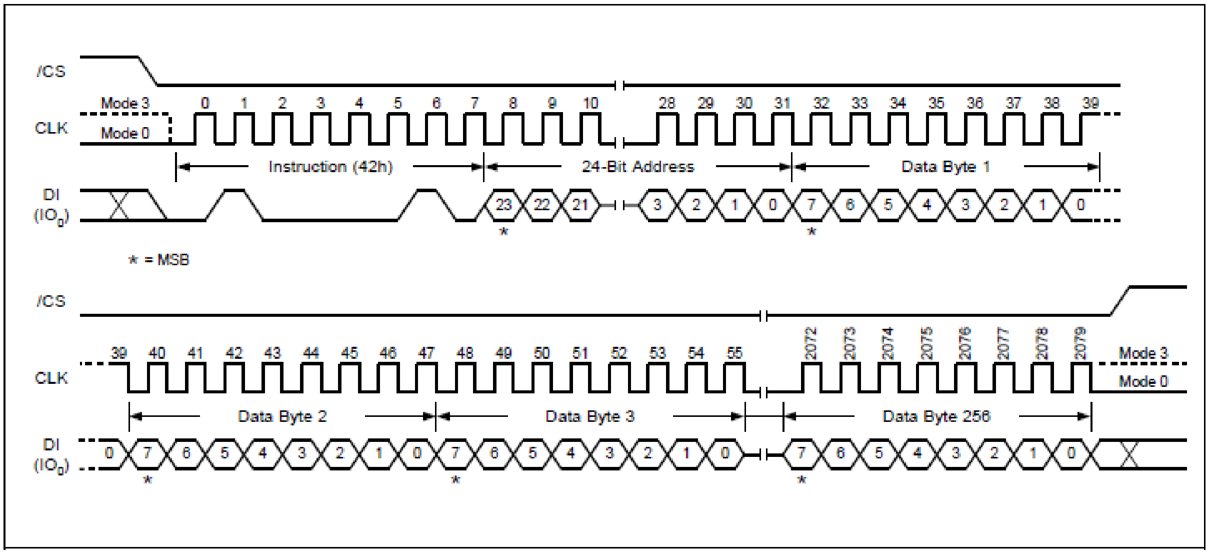


Figure. Program OTP array

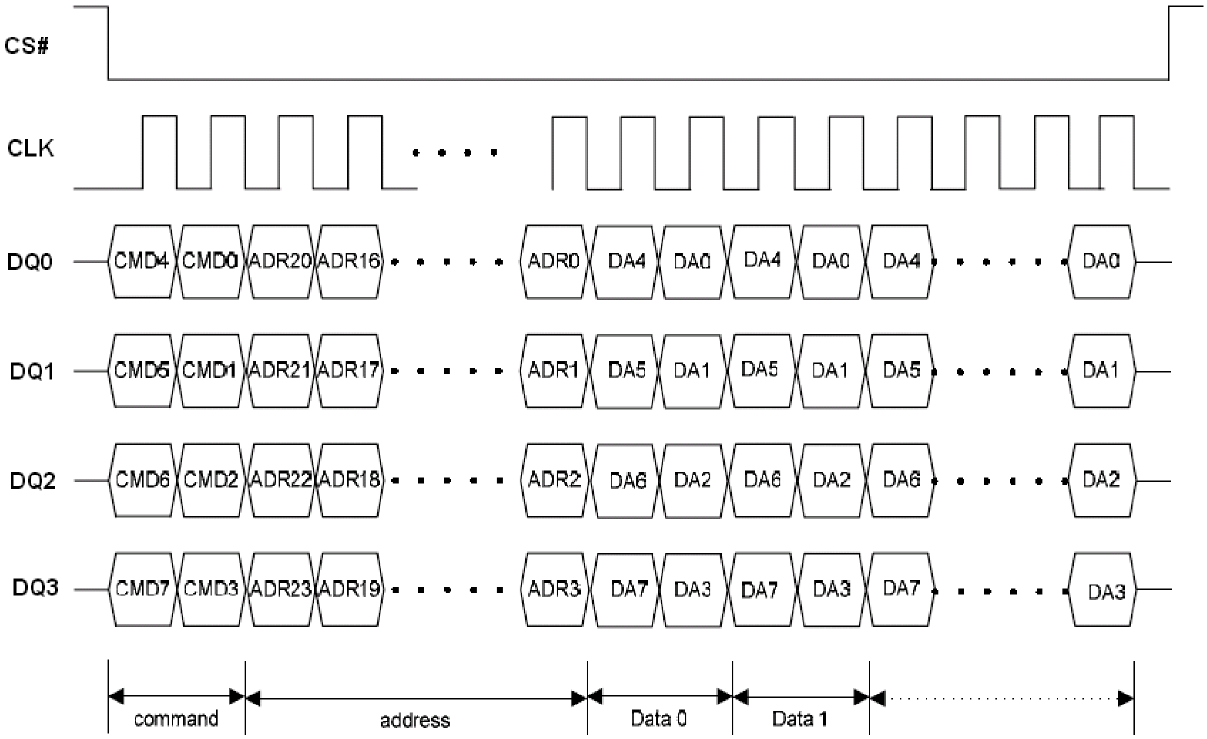

CS#
CLK
DQ0
DQ1
DQ2
DQ3
CMD4 CMD0 ADR20 ADR16 ADR0 DA4 DA0 DA4 DA0 DA4 DA0
CMD5 CMD1 ADR21 ADR17 ADR1 DA5 DA1 DA5 DA1 DA5 DA1
CMD6 CMD2 ADR22 ADR18 ADR2 DA6 DA2 DA6 DA2 DA6 DA2
CMD7 CMD3 ADR23 ADR19 ADR3 DA7 DA3 DA7 DA3 DA7 DA3
command
address
Data 0
Data 1

Figure. Program OTP array (QPI mode)

Read OTP array (48h)

The Read OTP array instruction is similar to the Fast Read instruction and allows one or more data bytes to be sequentially read from one of the three OTP array. The instruction is initiated by driving the CS# pin low and then shifting the instruction code “48h” followed by a 24-bit address (A23-A0) and eight “dummy” clocks into the DI pin.

The code and address bits are latched on the rising edge of the CLK pin. After the address is received, the data byte of the addressed memory location will be shifted out on the DO pin at the falling edge of CLK with most significant bit (MSB) first. The byte address is automatically incremented to the next byte address after each byte of data is shifted out. Once the byte address reaches the last byte of the register (byte address FFh), it will reset to address 00h, the first byte of the register, and continue to increment. The instruction is completed by driving CS# high. The Read OTP array instruction sequence is shown in Figure 47. If a Read OTP array instruction is issued while an Erase, Program or Write cycle is in process (WIP=1) the instruction is ignored and will not have any effects on the current cycle. The Read OTP array instruction allows clock rates from D.C. to a maximum of FR (see AC Electrical Characteristics).

This command also supports QPI mode.

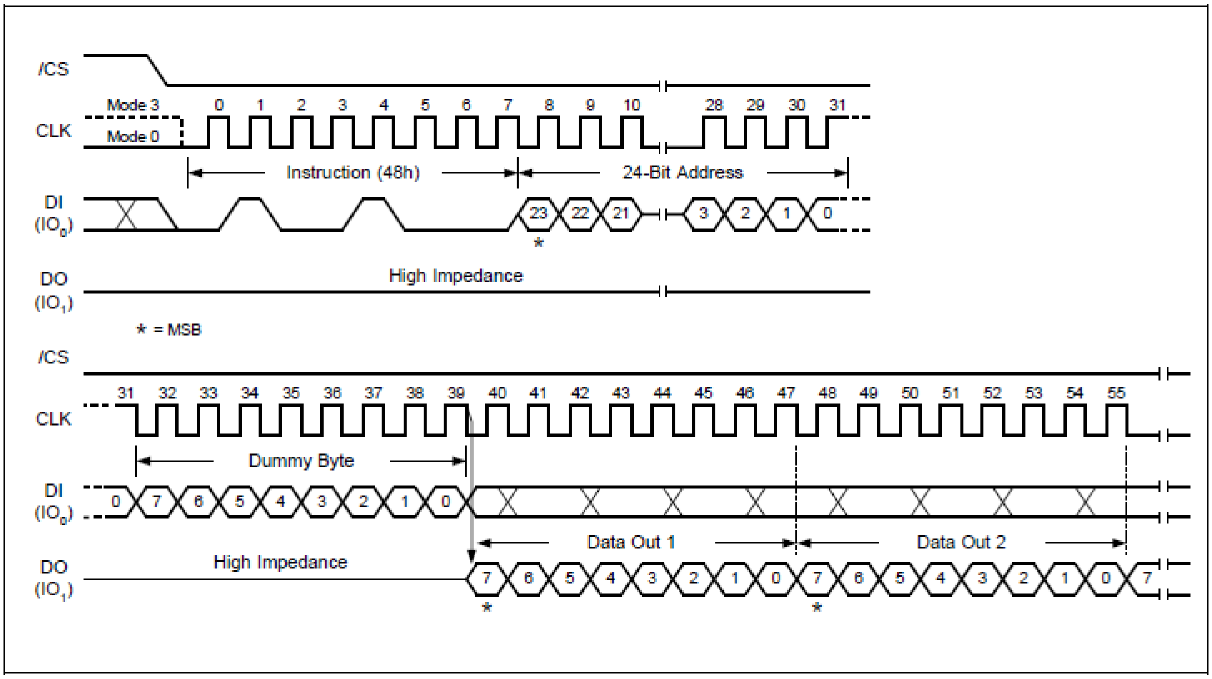


Figure. Read OTP array

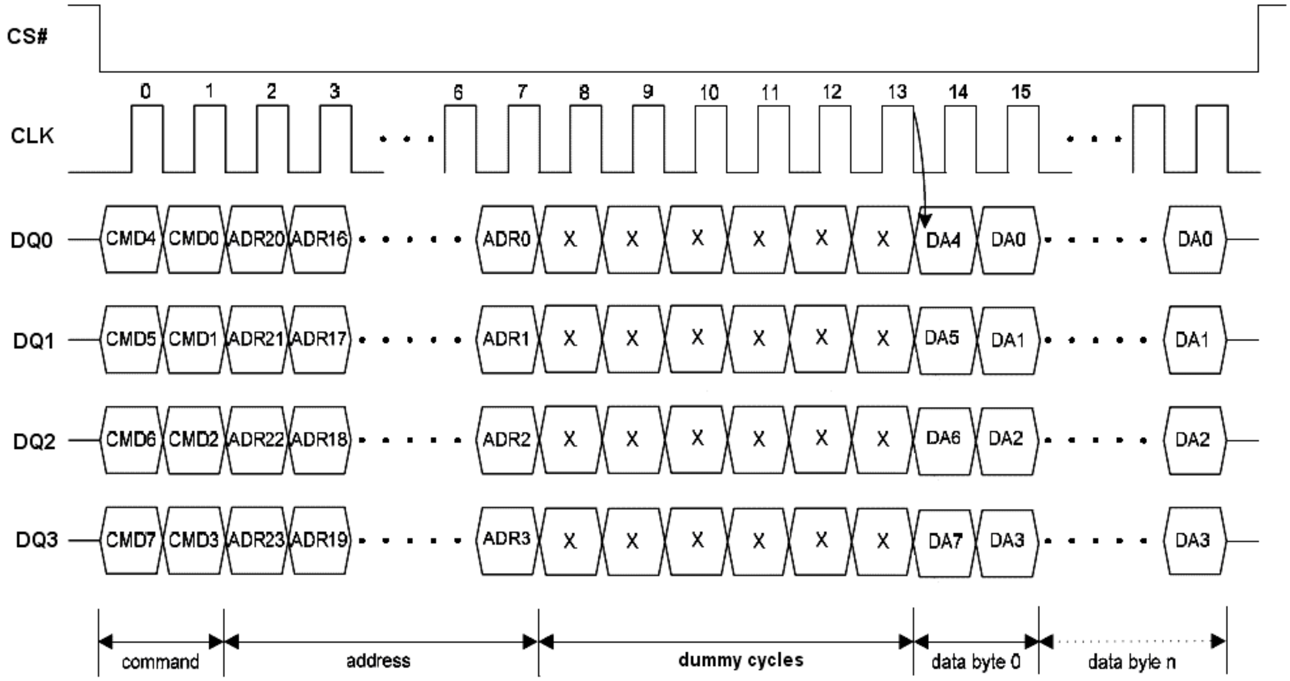


Figure. Read OTP array (QPI mode)

Erase OTP array (44h)

The device offers three set of 512-byte OTP array which can be erased and programmed individually. These registers may be used by the system manufacturers to store security and other important information separately from the main memory array.

The Erase OTP array instruction is similar to the Sector Erase instruction. A Write Enable instruction must be executed before the device will accept the Erase OTP array Instruction (Status Register bit WEL must equal 1). The instruction is initiated by driving the CS# pin low and shifting the instruction code “44h” followed by a 24-bit address (A23-A0) to erase one of the three security registers.

The Erase OTP array instruction sequence is shown in Figure 45. The CS# pin must be driven high after the eighth bit of the last byte has been latched. If this is not done the instruction will not be executed. After CS# is driven high, the self-timed Erase OTP array operation will commence for a time duration of tSE (See AC Characteristics). While the Erase OTP array cycle is in progress, the Read Status Register instruction may still be accessed for checking the status of the WIP bit. The WIP bit is a 1 during the erase cycle and becomes a 0 when the cycle is finished and the device is ready to accept other instructions again. After the Erase OTP array cycle has finished the Write Enable Latch (WEL) bit in the Status Register is cleared to 0. The Security Register Lock Bits (SPL0-3) in the Status Register-2 can be used to OTP protect the security registers. Once a lock bit is set to 1, the corresponding security register will be permanently locked, Erase OTP array instruction to that register will be ignored.

This command supports QPI mode

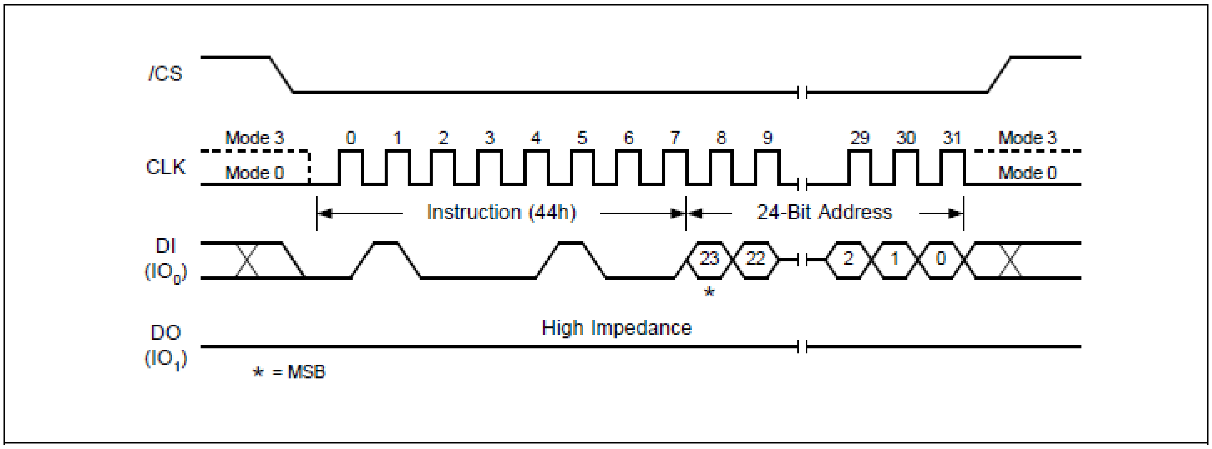


Figure. Erase OTP array

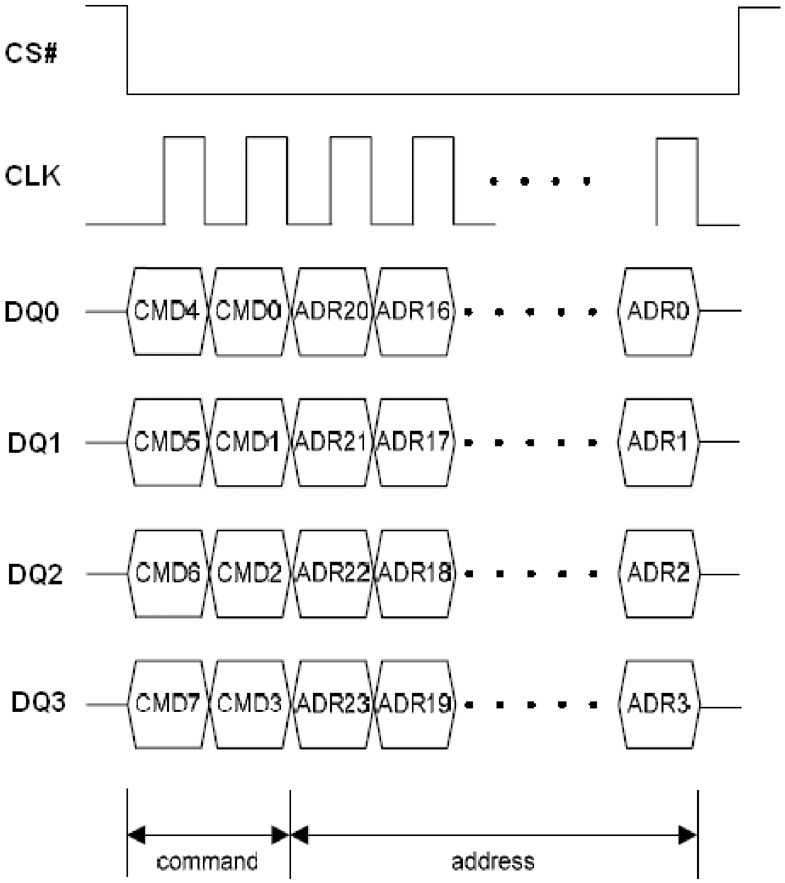


Figure. Erase OTP array (QPI mode)

Read SFDP Mode and Unique ID Number (5Ah) (the SFDP table length support 512 byte length include unique ID)

Read SFDP Mode

Device 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.

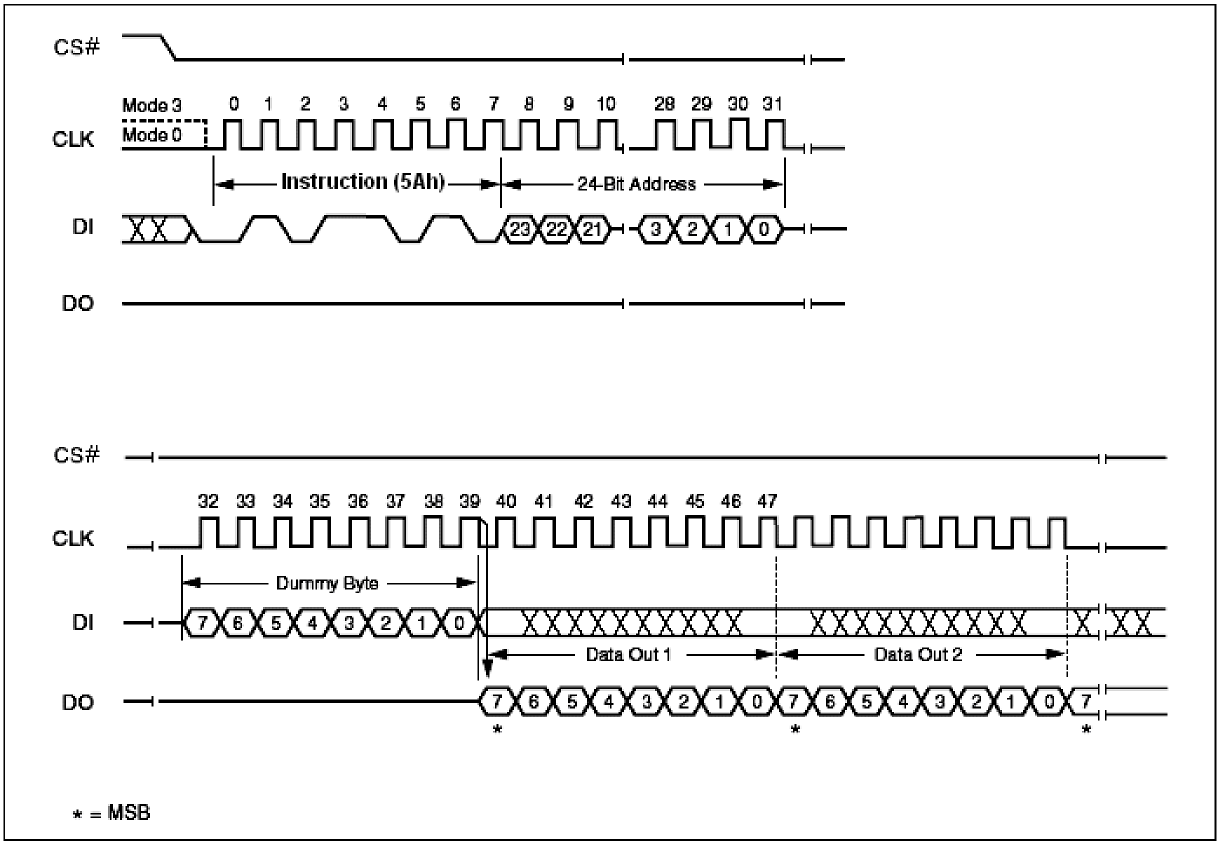


Figure 38. Read SFDP Mode Instruction Sequence Diagram

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

Description	Address (h) (Byte Mode)	Address (Bit)	Data (h)	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 (b/h)	Data (h)	Comment
Block / Sector Erase sizes Identifies the erase granularity for all Flash Components	30h	00	01b	EDh	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	20h	4 KB Erase Support (FFh = not supported)
		09			
		10			
		11			
		12			
		13			
		14			
		15			
Supports (1-1-2) Fast Read Device supports single input opcode & address and dual output data Fast Read	32h	16	1b	F1h	0 = not supported 1 = supported
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	1b		0 = not supported 1 = supported
Unused		23	1b		Reserved
Unused	33h	24	FFh	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 (h)	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 (b/h)	Data (h)	Comment
(1-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	38h	00	00100b	44h	4 dummy clocks
		01			
		02			
		03			
		04			
Quad Input Address Quad Output (1-4-4) Fast Read Number of Mode Bits		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	EBh	
		09			
		10			
		11			
		12			
		13			
		14			
		15			
(1-1-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ah	16	01000b	08h	Not supported
		17			
		18			
		19			
		20			
(1-1-4) Fast Read Number of Mode Bits		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	6Bh	

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

Description	Address (h) (Byte Mode)	Address (Bit)	Data (b/h)	Data (h)	Comment
(1-1-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Ch	00	01000b	08h	8 dummy clocks
		01			
		02			
		03			
		04			
(1-1-2) Fast Read Number of Mode Bits		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	3Bh	Not supported
(1-2-2) Fast Read Number of Wait states (dummy clocks) needed before valid output	3Eh	16	00100b	04h	4 dummy clocks
		17			
		18			
		19			
		20			
(1-2-2) Fast Read Number of Mode Bits		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	BBh	Not supported

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

Description	Address (h) (Byte Mode)	Address (Bit)	Data (b/h)	Data (h)	Comment
Supports (2-2-2) Fast Read Device supports dual input opcode & address and dual output data Fast Read.	40h	00	0b	FEh	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.		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	FFh	Reserved

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

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

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

Description	Address (h) (Byte Mode)	Address (Bit)	Data (b/h)	Data (h)	Comment
Reserved. These bits default to all 1's	49h : 48h	15 : 00	FFh	FFh	Reserved
(4-4-4) Fast Read Number of Wait states (dummy clocks) needed before valid output	4Ah	16	00100b	44h	4 dummy clocks
		17			
		18			
		19			
		20			
(4-4-4) Fast Read Number of Mode Bits		21	010b		8 mode bits
		22			
		23			
(4-4-4) Fast Read Opcode Opcode for quad input opcode/address, quad output data Fast Read.	4Bh	31 : 24	EBh	EBh	Must Enter EQPI Mode firstly

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

Description	Address (h) (Byte Mode)	Address (Bit)	Data (h)	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 (h)	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 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 (h)	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.

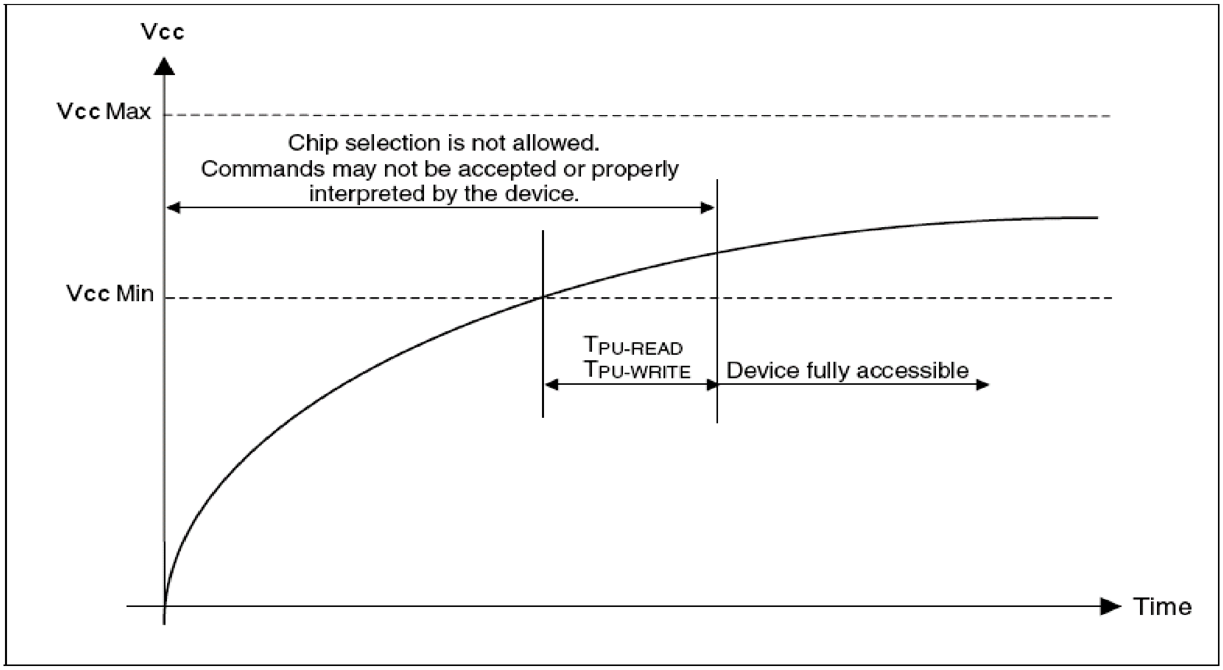

Vcc
Vcc Max
Chip selection is not allowed.
Commands may not be accepted or properly
interpreted by the device.
Vcc Min
TPU-READ
TPU-WRITE
Device fully accessible
Time

Figure 39. Power-up Timing

Table 15. Power-Up Timing

Symbol	Parameter	Min.	Unit
$T_{PU\text{-}READ}$ (1)	V_{CC} Min to Read Operation	100	µs
$T_{PU\text{-}WRITE}$ (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°C to 85°C; V_{CC} = 2.7-3.6V)

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}	-	1	20	µA
I_{CC2}	Deep Power-down Current	CS# = V_{CC}, V_{IN} = V_{SS} or V_{CC}	-	1	20	µA
I_{CC3}	Operating Current (READ)	CLK = 0.1 V_{CC} / 0.9 V_{CC} at 104MHz, DQ = open	-	7	14	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 80MHz, DQ = open	-	6	12	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 133MHz, Quad Output Read, DQ = open	-	15	30	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 104MHz, Quad Output Read, DQ = open	-	11	22	mA
		CLK = 0.1 V_{CC} / 0.9 V_{CC} at 80MHz, Quad Output Read, DQ = open	-	10	20	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		0.7 V_{CC}	-	V_{CC}+0.4	V
V_{OL}	Output Low Voltage	I_{OL} = 100 µA, V_{CC} = V_{CC} Min.	-	-	0.3	V
V_{OH}	Output High Voltage	I_{OH} = -100 µA, V_{CC} = V_{CC} 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

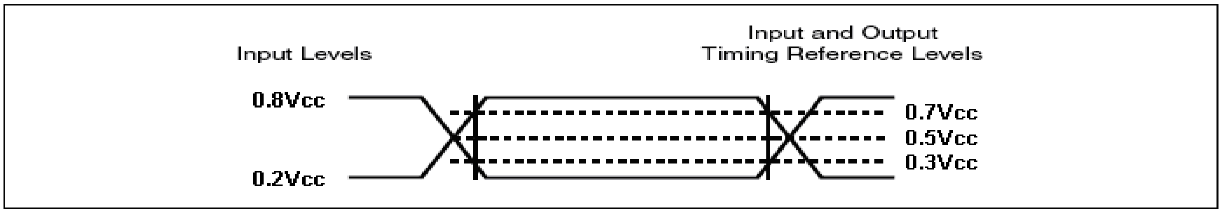


Figure 40. AC Measurement I/O Waveform

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
		Serial Quad Clock Frequency for: Quad Output Fast Read, Quad I/O Fast Read (Vcc: 3-3.6v)	D.C.	-	133	MHz
f_R		Serial Clock Frequency for READ	D.C.	-	50	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}		CS# Active Hold Time	5	-	-	ns
t_{SHCH}		CS# Not Active Setup Time	5	-	-	ns
t_{CHSL}		CS# Not Active Hold Time	5	-	-	ns
t_{SHSL}	t_{CSH}	CS# High Time for read	30	-	-	ns
		CS# High Time for program/erase	30	-	-	ns
t_{SHSL} [2]	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 (30pF)	-	-	8	ns
		Output Valid from CLK (15pF)	-	-	6	ns
t_{WHSL3}		Write Protect Setup Time before CS# Low	20	-	-	ns
t_{SHWL3}		Write Protect Hold Time after CS# High	100	-	-	ns
t_{DP} [2]		CS# High to Deep Power-down Mode	-	-	3	µs

Table 18. AC Characteristics-Continued

Symbol	Alt	Parameter		Min	Typ	Max	Unit
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	µs
			WIP = not in write operation	-	-	0	µs

Note:

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

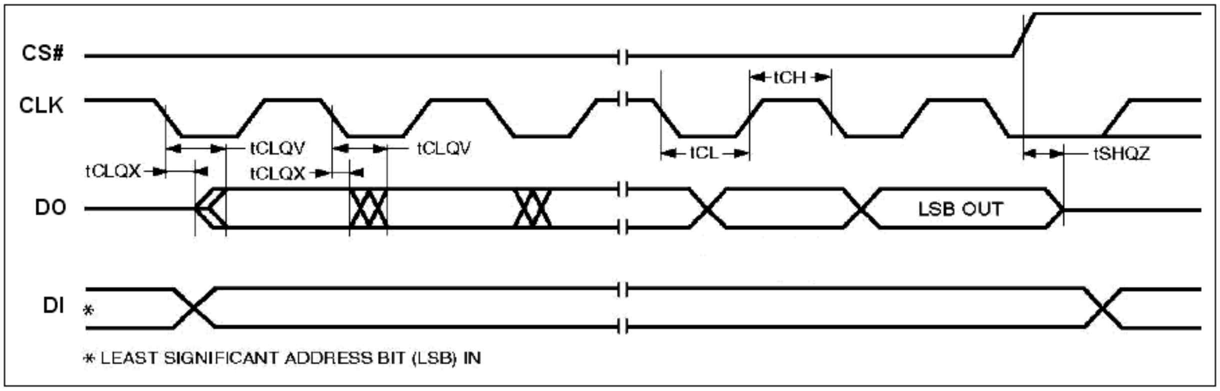


Figure 41. Serial Output Timing

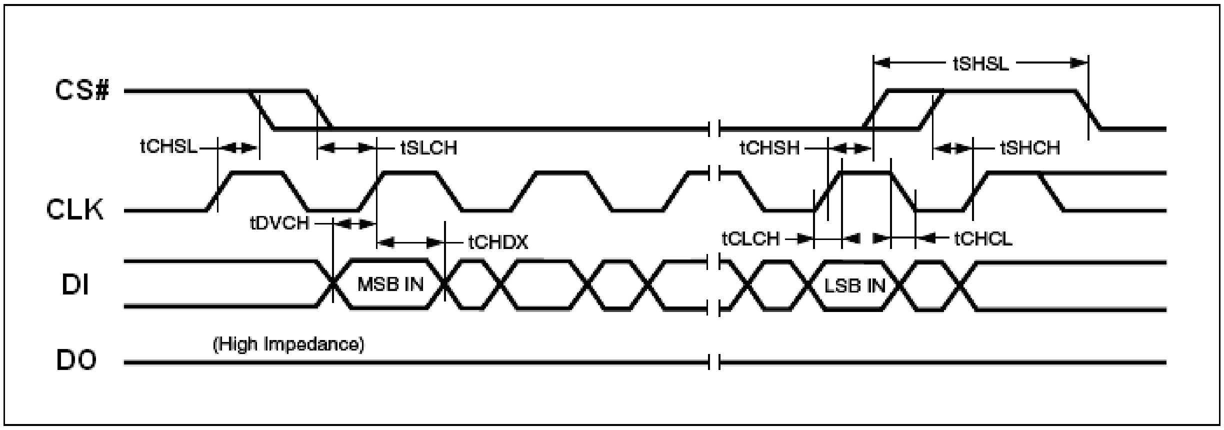


Figure 42. Input Timing

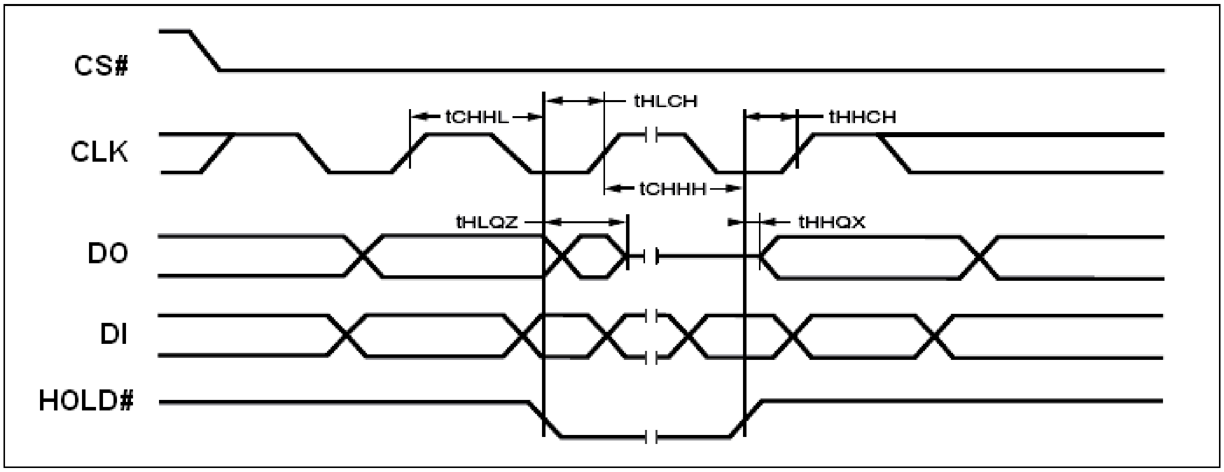


Figure 43. Hold Timing

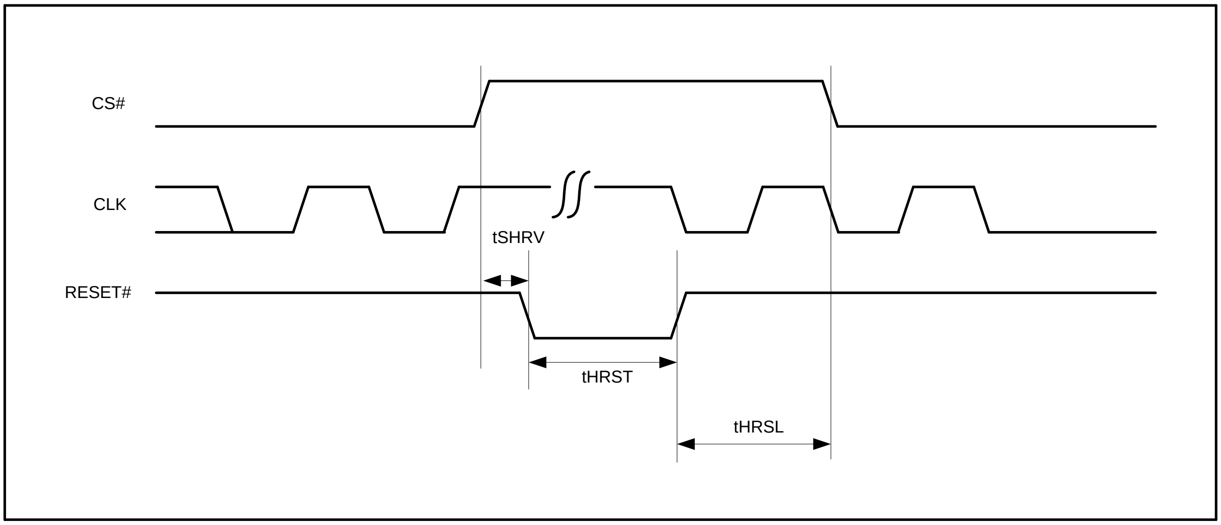


Figure 44. Reset Timing

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
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

Note:

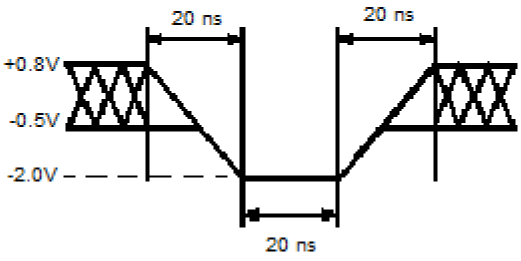
1. No more than one output shorted at a time. Duration of the short circuit should not be greater than one second.
2. Minimum DC voltage on input or I/O pins is -0.5V. During voltage transitions, inputs may undershoot V_{SS} to -1.0V for periods of up to 50ns and to -2.0V for periods of up to 20ns. See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

RECOMMENDED OPERATING RANGES[1]

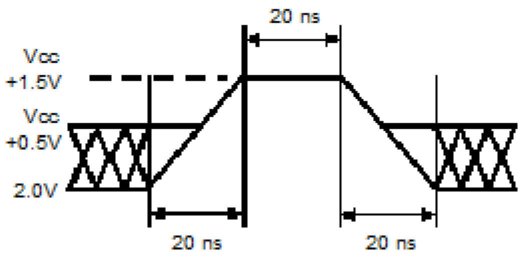
Parameter	Value	Unit
Ambient Operating Temperature Industrial Devices	-40 to 85	°C
Operating Supply Voltage V_{CC}	Full: 2.7 to 3.6	V

Note:

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

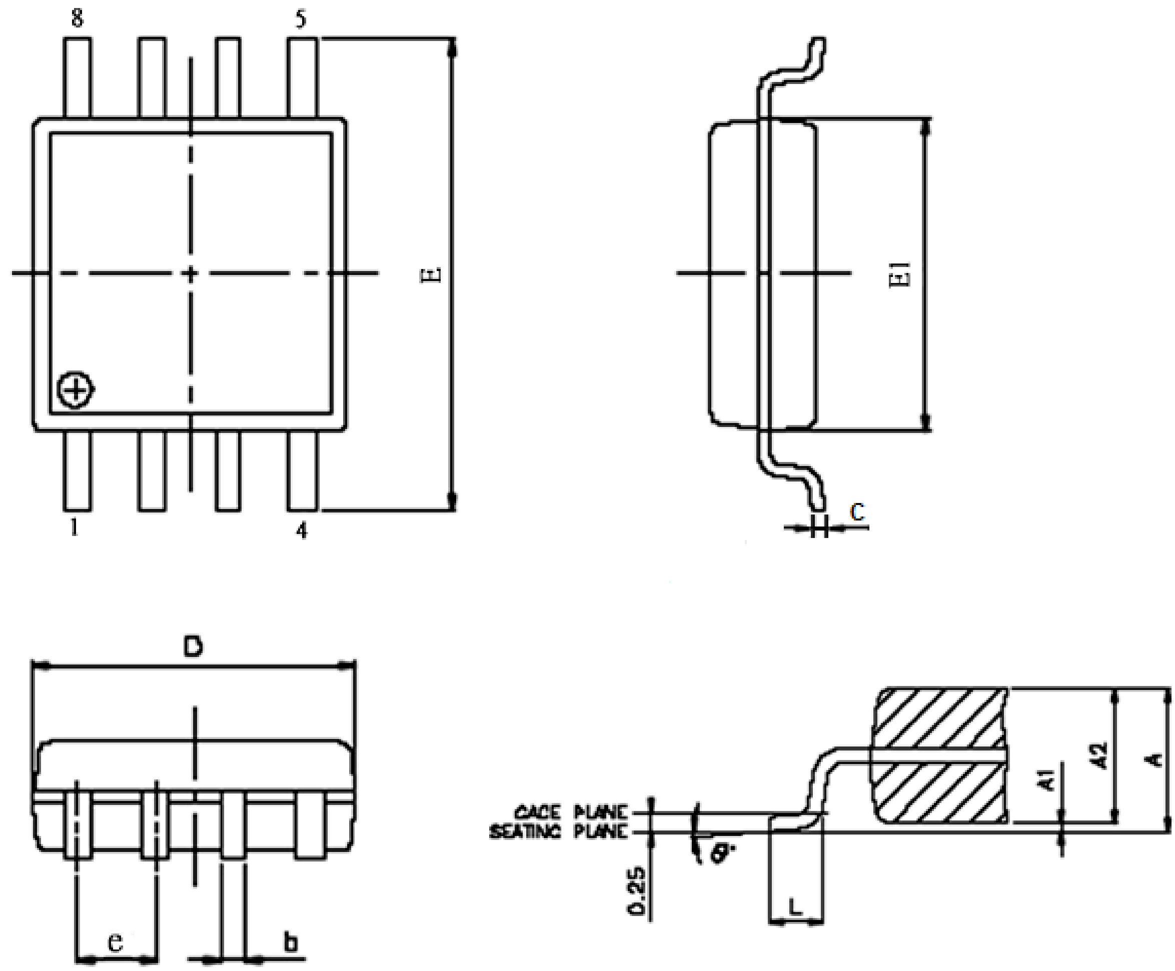
(V_{CC} = 2.7-3.6V)

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°C and a frequency of 20MHz.

PACKAGE MECHANICAL

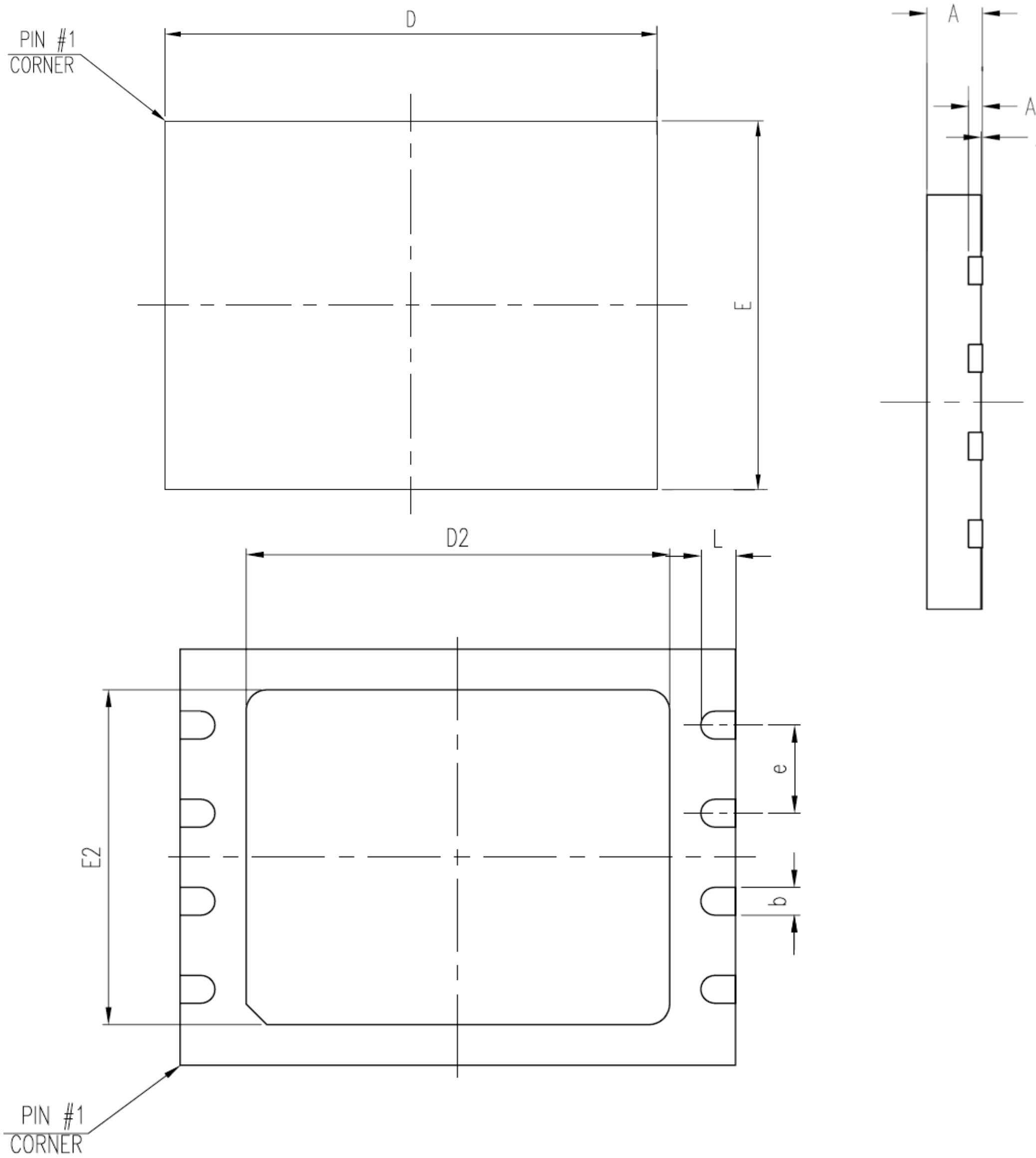
Figure 45. SOP 200 mil (official name = 208 mil)



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
C	0.19	0.200	0.25
L	0.5	0.65	0.80
θ	0^0	4^0	8^0

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

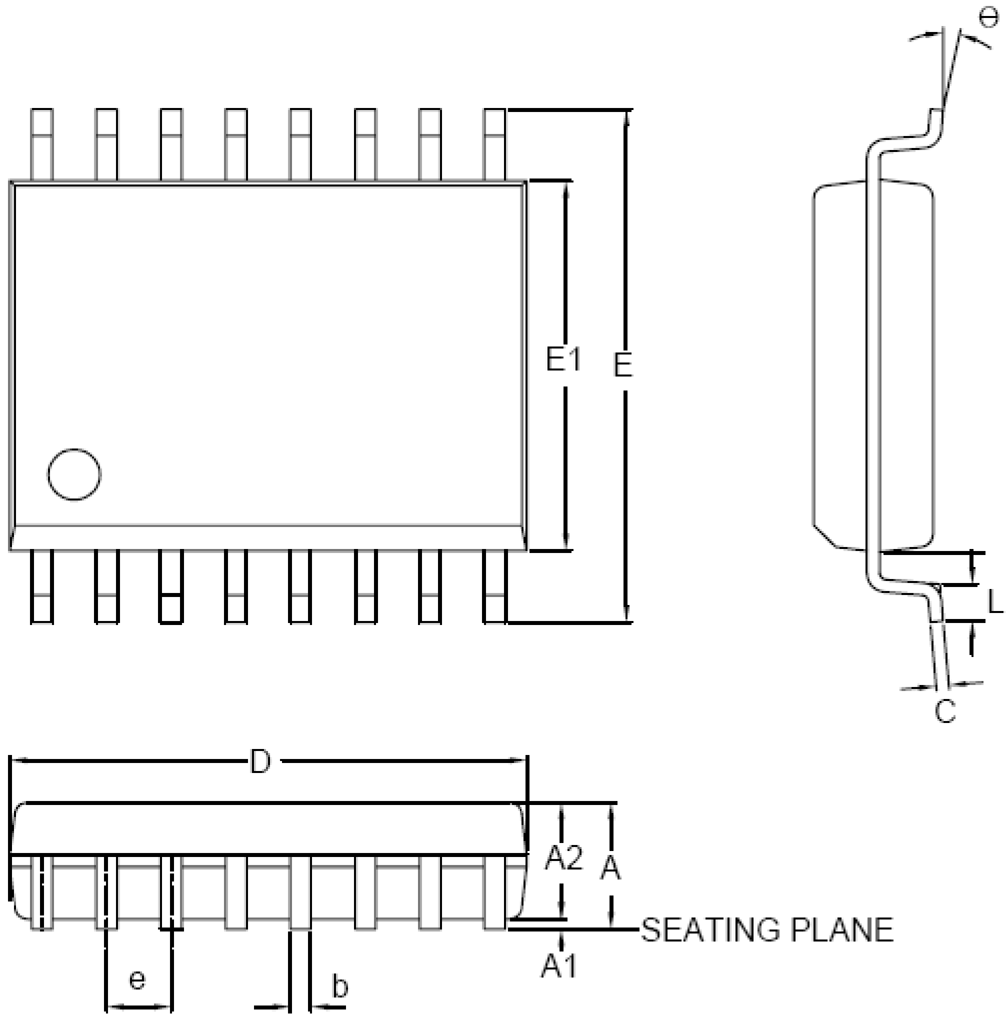
Figure 46. VDFN/ WSON 8 (6x5 mm)



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

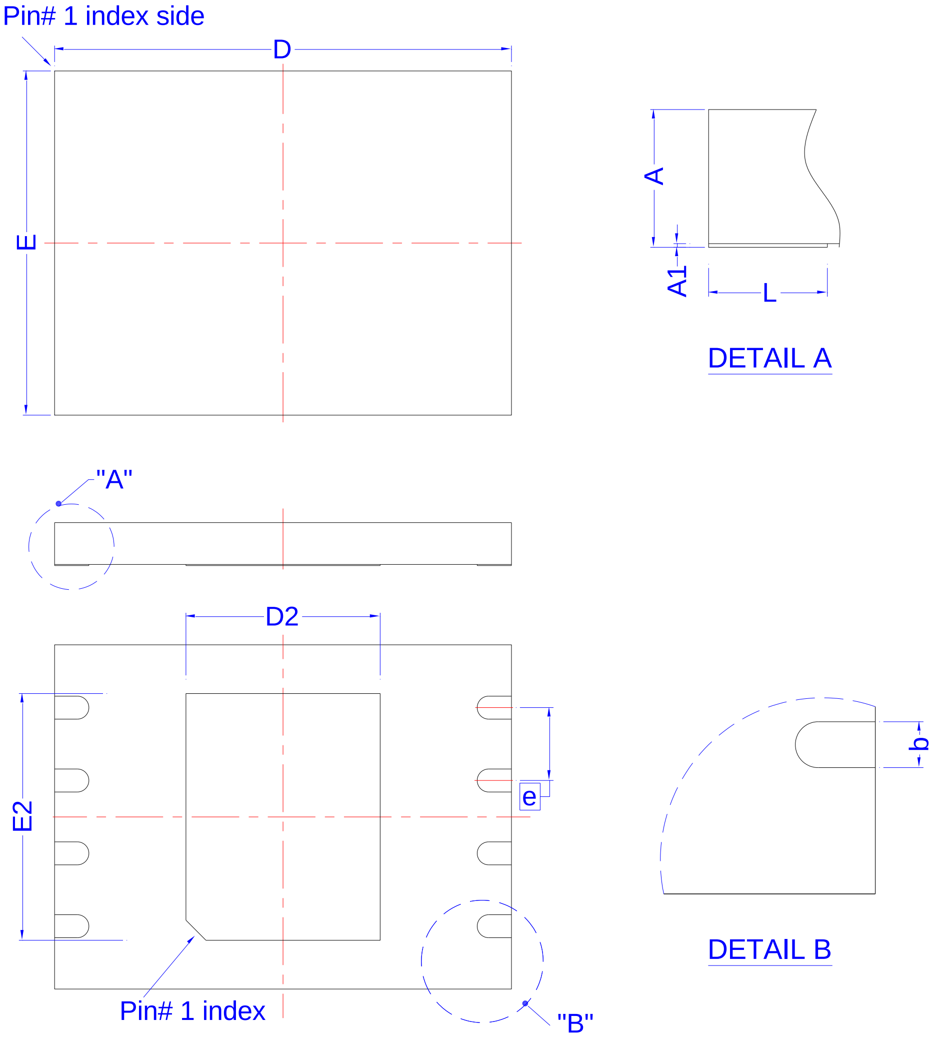
Figure 47. 16 LEAD SOP 300 mil



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	0^0	5^0	8^0

Note : 1. Coplanarity: 0.1 mm

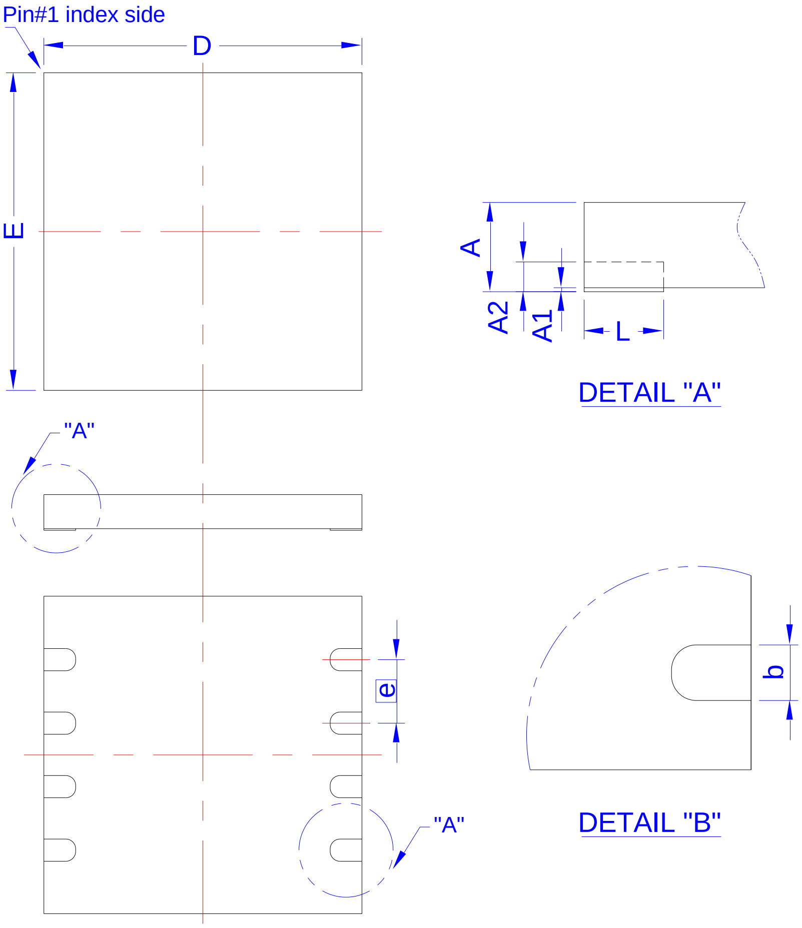
Figure 48. 8-LEAD VDFN/ WSON (8x6 mm)



Symbol	Dimension in mm			Dimension in inch		
	Min	Norm	Max	Min	Norm	Max
A	0.70	0.75	0.80	0.028	0.030	0.031
A1	0.00	0.02	0.05	0.000	0.001	0.002
b	0.35	0.40	0.48	0.014	0.016	0.019
D	8.00 BSC			0.315 BSC		
D2	3.30	3.40	3.50	0.130	0.134	0.138
E	6.00 BSC			0.236 BSC		
E2	4.20	4.30	4.40	0.165	0.169	0.173
e	1.27 BSC			0.050 BSC		
L	0.40	0.50	0.60	0.016	0.020	0.024

Controlling dimension: millimeter
(Revision date: Jul 14 2022)

Figure 49. USON 8 (4x4x0.45 mm) without expose metal pad



Symbol	Dimension in mm			Dimension in inch		
	Min	Norm	Max	Min	Norm	Max
A	0.40	0.45	0.50	0.016	0.018	0.020
A1	0.00	0.02	0.05	0.000	0.001	0.002
A2	--	0.15	--	--	0.006	--
b	0.25	0.30	0.35	0.010	0.012	0.014
D	3.90	4.00	4.10	0.154	0.157	0.161
E	3.90	4.00	4.10	0.154	0.157	0.161
e	0.80 BSC			0.031 BSC		
L	0.35	0.40	0.45	0.014	0.016	0.018

Controlling dimension : millimeter
(Revision date : Jul 16 2018)

ORDERING INFORMATION

EN25QX128A - 104 H I P 2X

DIFFERENTIATION CODE
2XE = USON/WSON package without Expose metal pad

PACKAGING CONTENT
P = RoHS, Halogen-Free and REACH compliant

TEMPERATURE RANGE
I = Industrial (-40°C to +85°C)

PACKAGE
H = 8-pin 200mil SOP
F = 16-pin 300mil SOP
W = 8-contact VDFN/ WSON (6x5mm)
Y = 8-contact VDFN/ WSON (8x6 mm)
XH = 8-contact USON (4x4x0.45mm)

SPEED
104 = 104 MHz

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

Revisions List

Revision No	Description	Date
1.0	Initial Release	2022/05/04
1.1	1. Modify the specification of suspend / resume time 2. Modify SFDP table 3. Revision 8-LEAD VDFN / WSON (8x6 mm) package dimension 4. Modify ordering information	2022/07/29

Important Notice

All rights reserved.

No part of this document may be reproduced or duplicated in any form or by any means without the prior permission of ESMT.

The contents contained in this document are believed to be accurate at the time of publication. ESMT assumes no responsibility for any error in this document, and reserves the right to change the products or specification in this document without notice.

The information contained herein is presented only as a guide or examples for the application of our products. No responsibility is assumed by ESMT for any infringement of patents, copyrights, or other intellectual property rights of third parties which may result from its use. No license, either express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of ESMT or others.

Any semiconductor devices may have inherently a certain rate of failure. To minimize risks associated with customer's application, adequate design and operating safeguards against injury, damage, or loss from such failure, should be provided by the customer when making application designs.

ESMT's products are not authorized for use in critical applications such as, but not limited to, life support devices or system, where failure or abnormal operation may directly affect human lives or cause physical injury or property damage. If products described here are to be used for such kinds of application, purchaser must do its own quality assurance testing appropriate to such applications.