PY25R128LA

Ultra-Low Power, 128M-bit

Serial Multi I/O RPMC Flash Memory Datasheet

Performance Highlight

- *Supply Range from 1.65 to 2.0V for Read, Erase and Program*
- *Ultra-Low Power consumption for Read, Erase and Program*
- *X1, X2 and X4 Multi I/O, QPI, DTR Support*
- *High reliability with 100K cycling and 20 Year-retention*
- *Permanently fixed QE bit, QE=1; and 4 I/O mode is enabled*
- *Provide Authentication feature by Monotonic Counter (MC) Feature*

1 Overview

General

- **Single 1.65 to 2.0V supply**
- **Industrial Temperature Range -40C to 85C**
- **Serial Peripheral Interface (SPI) Compatible: Mode 0 and Mode 3**
- **Single, Dual, Quad SPI, QPI, DTR**
  - Standard SPI: SCLK, CS#, SI, SO
  - Dual SPI: SCLK, CS#, IO0, IO1
  - Quad SPI: SCLK, CS#, IO0, IO1, IO2, IO3
  - QPI: SCLK, CS#, IO0, IO1, IO2, IO3
  - DTR: Double Transfer Rate Read
- **Flexible Architecture for Code and Data Storage**
  - Uniform 256-byte Page Program
  - Uniform 4K-byte Sector Erase
  - Uniform 32K/64K-byte Block Erase
  - Full Chip Erase
- **One Time Programmable (OTP) Security Register**
  - 3*1024-Byte Security Registers with OTP Lock
- **128-bit Unique ID for each device**
- **Support Replay Protection Monotonic Counter (RPMC)**
  - Four 32-bit Monotonic Counters
  - Volatile HMAC Key register
  - Non-volatile Root Key register
- **Fast Program and Erase Speed Typical**
  - 0.5ms Page program time
  - 50ms 4K-byte sector erase time
  - 0.16s 32K-byte block erase time
  - 0.2s 64K-byte block erase time
- **JEDEC Standard Manufacturer and Device ID Read Methodology**
- **Ultra-Low Power Consumption Typical**
  - 1uA Deep Power Down current
  - 30uA Standby current
  - 5.5mA Active Read current at 85MHz for STR 4IO
  - 7mA Active Program or Erase current
- **High Reliability**
  - 100,000 Program / Erase Cycles
  - 20-year Data Retention
- **Industry Standard Green Package Options**
  - 8-Lead SOP (208mil)
  - 16-Lead SOP (300mil)
  - 8-Pad WSON(6x5x0.75mm)
  - 8-Pad WSON(8x6x0.75mm)

2 Description

The PY25R128LA is a serial interface Flash memory device designed for use in a wide variety of high-volume consumer-based applications in which program code is shadowed from Flash memory into embedded or external RAM for execution. The flexible erase architecture of the device, with its page erase granularity it is ideal for data storage as well, eliminating the need for additional data storage devices.

The erase block sizes of the device have been optimized to meet the needs of today's code and data storage applications. By optimizing the size of the erase blocks, the memory space can be used much more efficiently. Because certain code modules and data storage segments must reside by themselves in their own erase regions, the wasted and unused memory space that occurs with large sectored and large block erase Flash memory devices can be greatly reduced. This increased memory space efficiency allows additional code routines and data storage segments to be added while still maintaining the same overall device density.

The device also contains an additional 3*1024-byte security registers with OTP lock (One-Time Programmable), can be used for purposes such as unique device serialization, system-level Electronic Serial Number (ESN) storage, locked key storage, etc.

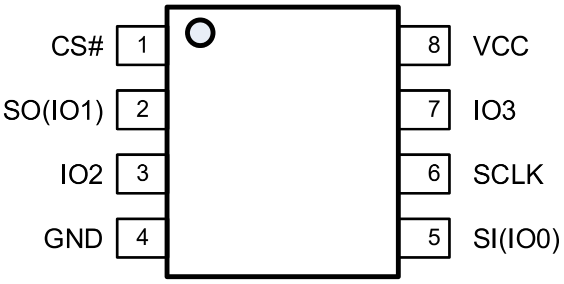
Specifically designed for use in many different systems, the device supports read, program, and erase operations with a wide supply voltage range of 1.65V to 2.0V. No separate voltage is required for programming and erasing.

The PY25R series is also equipped with an enhanced authentication security feature by Replay Protection Monotonic Counter (RPMC). It provides a high level secured communication between the flash device and the controller to detect Hardware attack.

3 Pin Definition

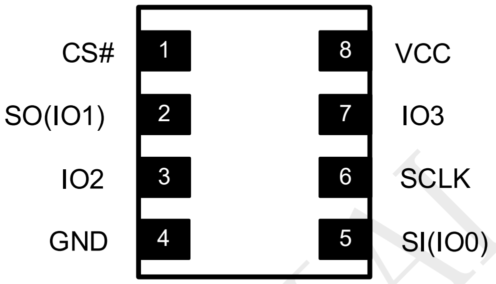
3.1 Pin Configurations

Figure 3-1 for SOP8 package



8-Lead SOP (208mil)

Figure 3-2 for WSON8 package



8-Pad WSON(6x5mm/8x6mm)

3.2 Pin Descriptions

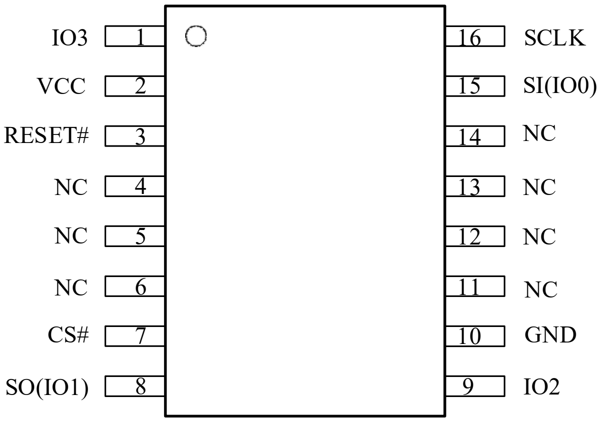
Table 3-1 Pin Descriptions for SOP8 and WSON8

No.	Symbol	Extension	Remarks
1	CS#	I	Chip select
2	SO(IO1)	I/O	Serial data output for 1 x I/O Serial data input and output for 2 x I/O read mode Serial data input and output for 4 x I/O read mode
3	IO2	I/O	Serial data input and output for 4 x I/O read mode
4	GND	-	Ground of the device
5	SI(IO0)	I/O	Serial data input for 1x I/O Serial data input and output for 2 x I/O read mode Serial data input and output for 4 x I/O read mode
6	SCLK	I	Serial interface clock input
7	IO3	I/O	Serial data input and output for 4 x I/O read mode
8	VCC	-	Power supply of the device

Note:

1. IO0 and IO1 are used for Standard and Dual SPI instructions
2. IO0 – IO3 are used for Quad SPI instructions.

Figure 3-3 for SOP16 package



16-Pin SOP (300mil)

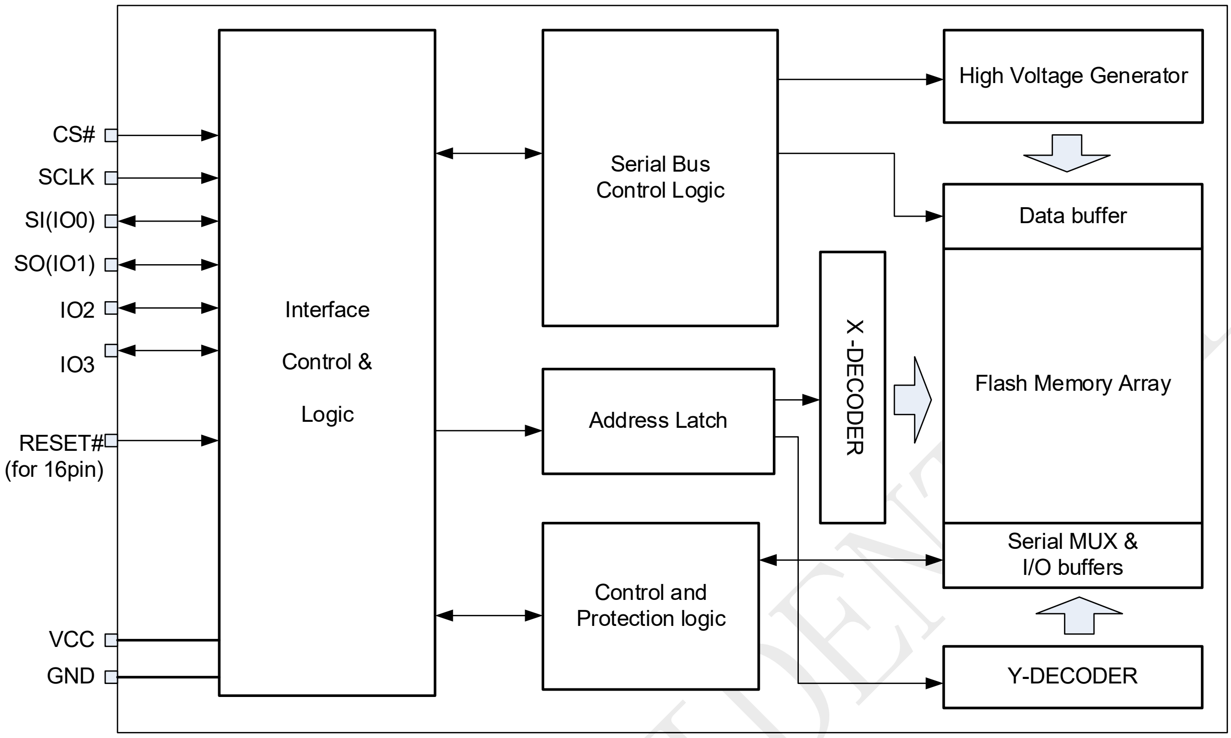
Table 3-2 Pin Descriptions for SOP16

No.	Symbol	Extension	Remarks
1	IO3	I/O	Serial data input and output for 4 x I/O read mode
2	VCC	-	Power supply of the device
3	RESET#	I	Hardware Reset Pin Active low
4	NC	-	No Connect
5	NC	-	No Connect
6	NC	-	No Connect
7	CS#	I	Chip select
8	SO(IO1)	I/O	Serial data output for 1 x I/O Serial data input and output for 2 x I/O read mode Serial data input and output for 4 x I/O read mode
9	IO2	I/O	Serial data input and output for 4 x I/O read mode
10	GND	-	Ground of the device
11	NC	-	No Connect
12	NC	-	No Connect
13	NC	-	No Connect
14	NC	-	No Connect
15	SI(IO0)	I/O	Serial data input for 1x I/O Serial data input and output for 2 x I/O read mode Serial data input and output for 4 x I/O read mode
16	SCLK	I	Serial interface clock input

Note:

1. IO0 and IO1 are used for Standard and Dual SPI instructions
2. IO0 – IO3 are used for Quad SPI instructions
3. The RESET# pin on SOP-16 package is a dedicated hardware reset pin regardless of device settings. If the reset function is not used, this pin can be left floating in the system.

4 Block Diagram



5 Electrical Specifications

5.1 Absolute Maximum Ratings

- Storage Temperature ....-65°C to +150°C
- Operation Temperature ....-40°C to +85°C
- Maximum Operation Voltage............. 2.5V
- Voltage on Any Pin with
  respect to Ground...........................-0.6V to VCC + 0.5V
- DC Output Current ....5.0 mA

NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 5-1 Pin Capacitance [1]

Symbol	Parameter	Max	Units	Test Condition
C_{OUT}	Output Capacitance	8	pF	V_{OUT}=GND
C_{IN}	Input Capacitance	6	pF	V_{IN}=GND

Note:

1. Test Conditions: T_A= 25°C, F = 1MHz, Vcc = 1.8V.

Figure 5-1 Input Test Waveforms and Measurement Level

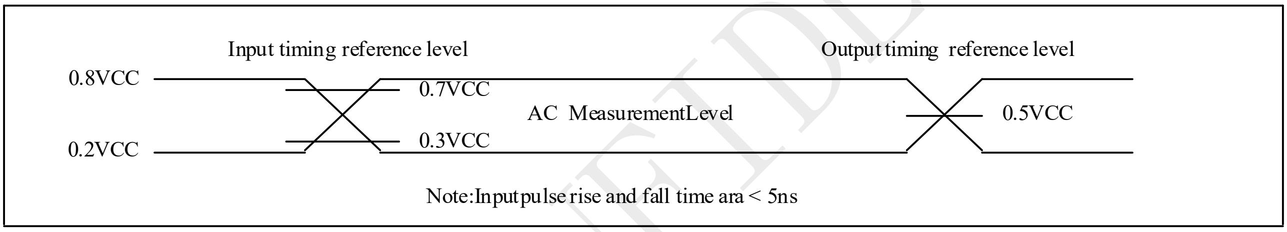
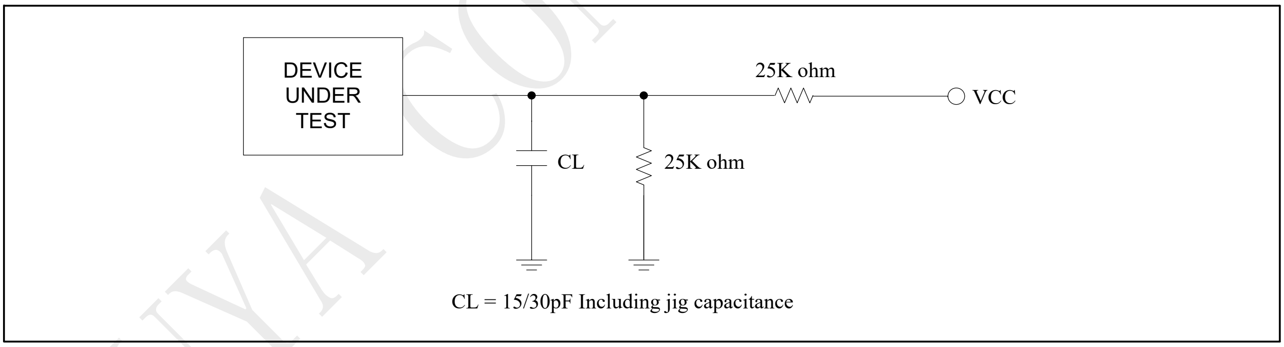

Input timing reference level
Output timing reference level
0.8VCC
0.7VCC
AC Measurement Level
0.5VCC
0.2VCC
0.3VCC
Note:Input pulse rise and fall time ara < 5ns

Figure 5-2 Output Loading



5.2 DC Characteristics

Table 5-2 DC Parameters (Ta=-40°C ~ +85°C)

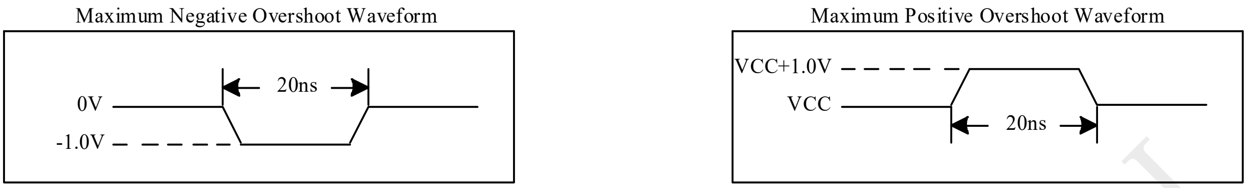
Symbol	Parameter	Conditions	1.65V~2.0V			Units
			Min	Typ[1]	Max	
I_{DPD}	Deep power down current	CS#=Vcc, all other inputs at 0V or Vcc		1	10	uA
I_{SB}	Standby current	CS#=VIH all inputs at CMOS levels		30	100	uA
I_{CC1}	Read current (STR) (1 IO)	f=50MHz; IOUT=0mA		2.5	4	mA
		f=85MHz; IOUT=0mA		3.5	6	mA
		f=133MHz; IOUT=0mA		5	8	mA
	Read current (STR) (2 IO)	f=50MHz; IOUT=0mA		3	5	mA
		f=85MHz; IOUT=0mA		4.5	7	mA
		f=133MHz; IOUT=0mA		6.5	9	mA
	Read current (STR) (4 IO)	f=50MHz; IOUT=0mA		4	6	mA
		f=85MHz; IOUT=0mA		5.5	8	mA
		f=133MHz; IOUT=0mA		7.5	10	mA
I_{CC2}	Read current (DTR) (1 IO)	f=33MHz; IOUT=0mA		2.5	4	mA
		f=50MHz; IOUT=0mA		3.0	6	mA
		f=80MHz; IOUT=0mA		4.5	8	mA
	Read current (DTR) (2 IO)	f=33MHz; IOUT=0mA		3.0	5	mA
		f=50MHz; IOUT=0mA		4.0	7	mA
		f=80MHz; IOUT=0mA		6.0	9	mA
	Read current (DTR) (4 IO)	f=33MHz; IOUT=0mA		4.5	7	mA
		f=50MHz; IOUT=0mA		5.5	9	mA
		f=80MHz; IOUT=0mA		8.0	11	mA
I_{CC3}	Program current	CS#=Vcc		8	12	mA
I_{CC4}	Erase current	CS#=Vcc		7	12	mA
I_{CC5}	RPMC OP1 Write Only	CS#=Vcc		4	20	mA
I_{CC6}	RPMC OP1 & Array Read	CLK=0.1VCC/0.9VCC, SO open		15	20	mA
I_{CC7}	RPMC OP1 & Array Program/Erase	CS#=Vcc		15	20	mA
I_{CC8}	RPMC OP2 Read only	CLK=0.1VCC/0.9VCC, SO open		4	20	mA
I_{CC9}	RPMC OP2 & Array Program/Erase	CS#=Vcc		15	20	mA
I_{LI}	Input load current	All inputs at CMOS level			4	uA
I_{LO}	Output leakage	All inputs at CMOS level			4	uA
V_{IL}	Input low voltage		-0.5		0.3Vcc	V
V_{IH}	Input high voltage		0.7Vcc		Vcc+0.3	V
V_{OL}	Output low voltage	IOL=100uA			0.2	V
V_{OH}	Output high voltage	IOH=-100uA	Vcc-0.2			V

Note:

1.Typical values measured at 1.8V @ 25°C for the 1.6V to 2.0V range.

2.The value guaranteed by characterization, not 100% tested in production.

Figure 5-3 Maximum Overshoot Waveform


Maximum Negative Overshoot Waveform
20ns
0V
-1.0V
Maximum Positive Overshoot Waveform
VCC+1.0V
VCC
20ns

During DC conditions, input or I/O signals should remain equal to or between VSS and VCC. During voltage transitions, inputs or I/Os may negative overshoot to -1.0V or positive overshoot to VCC + 1.0 V, for periods up to 20 ns.

5.3 AC Characteristics

Table 5-3-1 AC Parameters (Ta=-40°C ~ +85°C)

Symbol	Alt.	Parameter	1.65V~2.0V			Units
			Min	Typ	Max	
fSCLK	fC	Clock Frequency for all instructions except for special marking	D.C		133	MHz
fRSCLK	fR	Clock Frequency for READ instructions	D.C		80	MHz
fWSCLK	fW	Clock Frequency for 4IO WORD READ instructions	D.C		104	MHz
fDSCLK	fD	Clock Frequency for DTR instructions			80	MHz
fRPMC		Clock frequency for RPMC instruction			104	MHz
tCH[1]	tCLH	Clock High Time	3.3			ns
tCL[1]	tCLL	Clock Low Time, 45% x (1/fSCLK)	3.3			ns
tCLCH[3]		Clock Rise Time (peak to peak)	0.1			v/ns
tCHCL[3]		Clock Fall Time (peak to peak)	0.1			v/ns
tSLCH	tCSS	CS# Active Setup Time (relative to SCLK)	5			ns
tCHSL		CS# Not Active Hold Time (relative to SCLK)	5			ns
tDVCH	tDSU	Data in Setup Time	2			ns
tCHDX	tDH	Data in Hold Time	3			ns
tCHSH		CS# Active Hold Time (relative to SCLK)	5			ns
tSHCH		CS# Not Active Setup Time (relative to SCLK)	5			ns
tSHSL	tCSH	CS# Deselect Time from Read to next Read	20			ns
		CS# Deselect Time from Write, Erase, Program to Read Status Register	30			ns
tSHQZ[3]	tDIS	Output Disable Time			6	ns
tCLQV	tV	Clock Low to Output Valid Loading 30pF			7	ns
		Clock Low to Output Valid Loading 15pF			6	ns
tCLQX	tHO	Output Hold Time	1			ns
tDP		CS# High to Deep Power-down Mode			3	us
tRES1		CS# High to Standby Mode Without Electronic Signature Read			30	us
tRES2		CS# High to Standby Mode with Electronic Signature Read			30	us
tW		Write Status Register Cycle Time		2	12	ms
tWRK		Write Root Key Time		80	250	us
tUHK		Update HMAC Key Time		85	100	us
tIMC1		Increment Monotonic Counter Time		55	100	us
tIMC2		Increment Monotonic Counter Time (Counter Switching)		50	150	ms
tRQMC		Request Monotonic Counter Time		45	60	us

Note:

1. tCH + tCL must be greater than or equal to 1/ Frequency.
2. Only applicable as a constraint for a WRSR instruction.
3. The value guaranteed by characterization, not 100% tested in production.

Table 5-3-2 SPI Read Command Performance Comparison (MHz)

Read command	Dummy Cycles (VCC=1.65V~2.0V)			
	4	6	8	10
FREAD	-	-	133	-
DREAD	-	-	133	-
2READ	104 (default)	-	133	-
QREAD	-	-	133	-
4READ	-	104 (default)	-	133
DTR_FREAD	-	80	-	-
DTR_2READ	-	80	-	-
DTR_4READ	-	-	-	80

Table 5-3-3 QPI Read Command Performance Comparison (MHz)

Read command	Dummy Cycles (VCC=1.65V~2.0V)			
	4	6	8	10
4READ	70	104	120	133 (default)
BURST READ	70	104	120	133 (default)
DTR_FREAD	-	-	-	80
DTR_4READ	-	-	-	80
DTR_BURST READ	-	-	-	80

5.4 AC Characteristics for Program and Erase

Table 5-4 AC Parameters for Program and Erase (Ta=-40°C ~ +85°C)

Symbol	Parameter	1.65V~2.0V			Units
		Min	Typ[1]	Max	
T_{ESL}[4]	Erase Suspend Latency			30	us
T_{PSL}[4]	Program Suspend Latency			30	us
T_{PRS}[2]	Latency between Program Resume and next Suspend	0.5			us
T_{ERS}[3]	Latency between Erase Resume and next Suspend	0.5			us
t_{PSR}	Program Security Registers time (up to 256 bytes)		0.5	2.4	ms
t_{ESR}	Erase Security Registers time		50	240	ms
t_{PP}	Page program time (up to 256 bytes)		0.5	2.4	ms
t_{SE}	Sector erase time		50	240	ms
t_{BE1}	Block erase time for 32K bytes		0.16	0.8	s
t_{BE2}	Block erase time for 64K bytes		0.2	1.2	s
t_{CE}	Chip erase time		30	120	s

Note:

1. Typical values given for TA=25°C. Not 100% tested.
2. Program operation may be interrupted as often as system request. The minimum timing of tPRS must be observed before issuing the next program suspend command. However, in order for a Program operation to make progress, tPRS ≥ 100us must be included in resume-to-suspend loop(s). Not 100% tested.
3. Erase operation may be interrupted as often as system request. The minimum timing of tERS must be observed before issuing the next erase suspend command. However, in order for an Erase operation to make progress, tERS ≥ 100us must be included in resume-to-suspend loop(s). Not 100% tested.
4. Latency time is required to complete Erase/Program Suspend operation.
5. The value guaranteed by characterization, not 100% tested in production.

Figure 5-4 Serial Input Timing

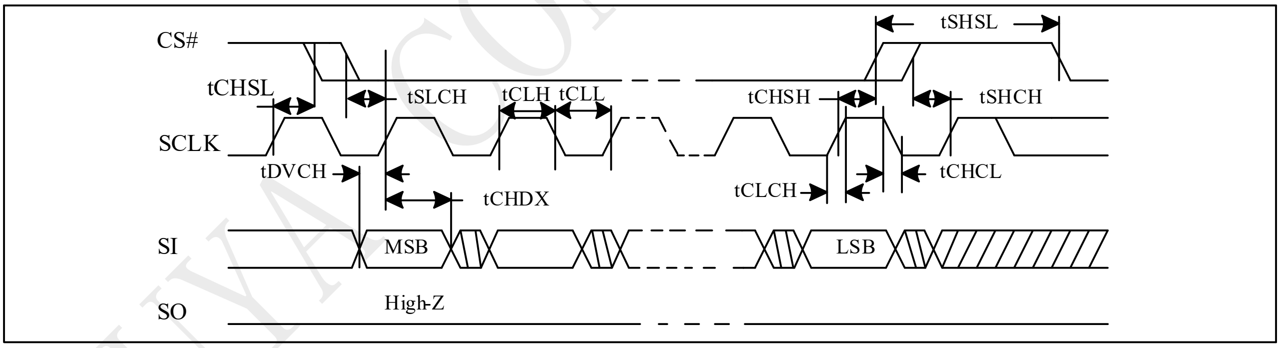
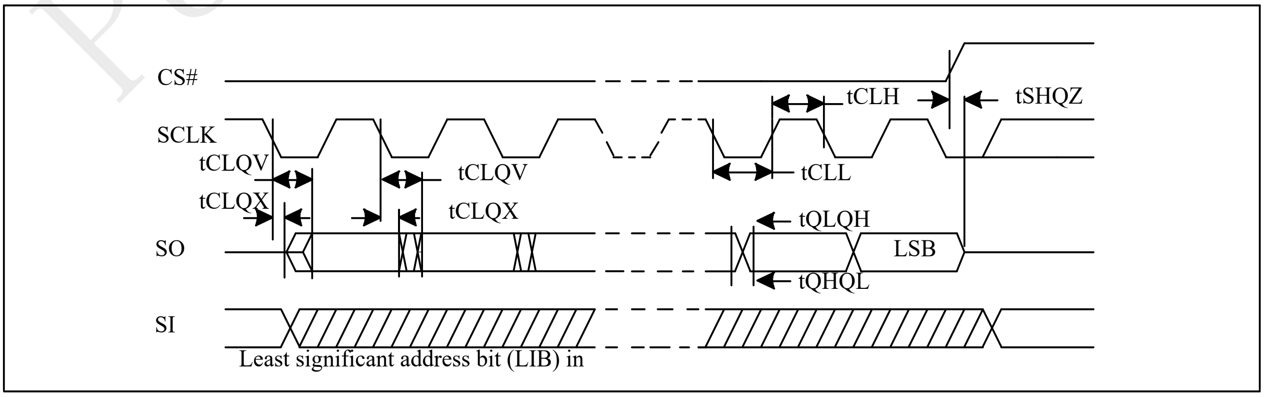


Figure 5-5 Output Timing



5.5 Operation Conditions

At Device Power-Up and Power-Down

AC timing illustrated in "Figure AC Timing at Device Power-Up" and "Figure Power-Down Sequence" are for the supply voltages and the control signals at device power-up and power-down. If the timing in the figures is ignored, the device will not operate correctly.

During power-up and power-down, CS# needs to follow the voltage applied on VCC to keep the device not to be selected. The CS# can be driven low when VCC reach Vcc(min.) and wait a period of tVSL.

Figure 5-6 AC Timing at Device Power-Up

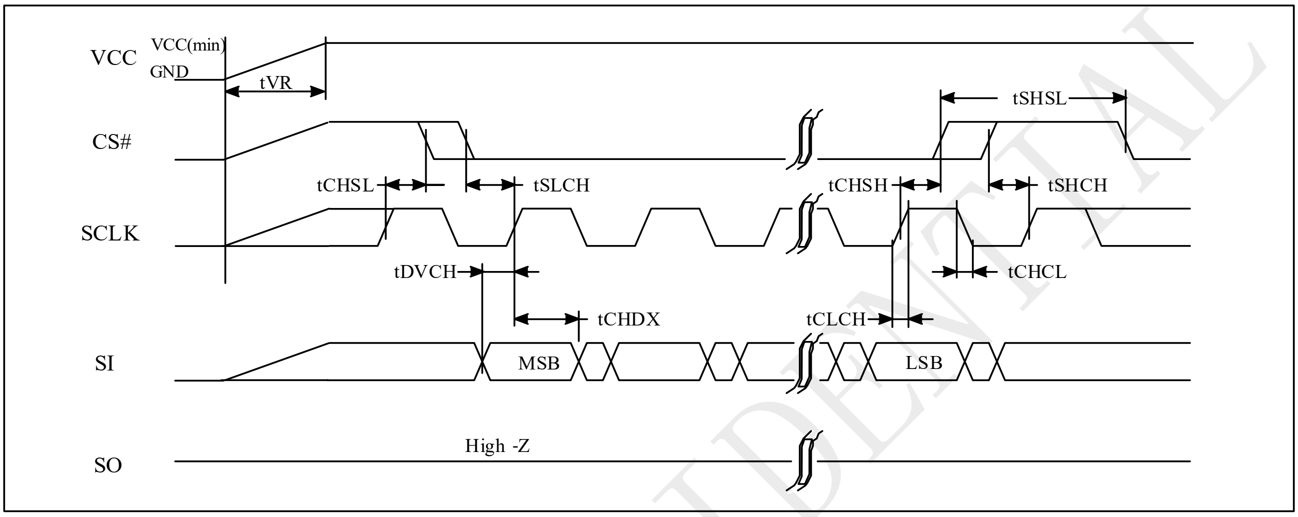
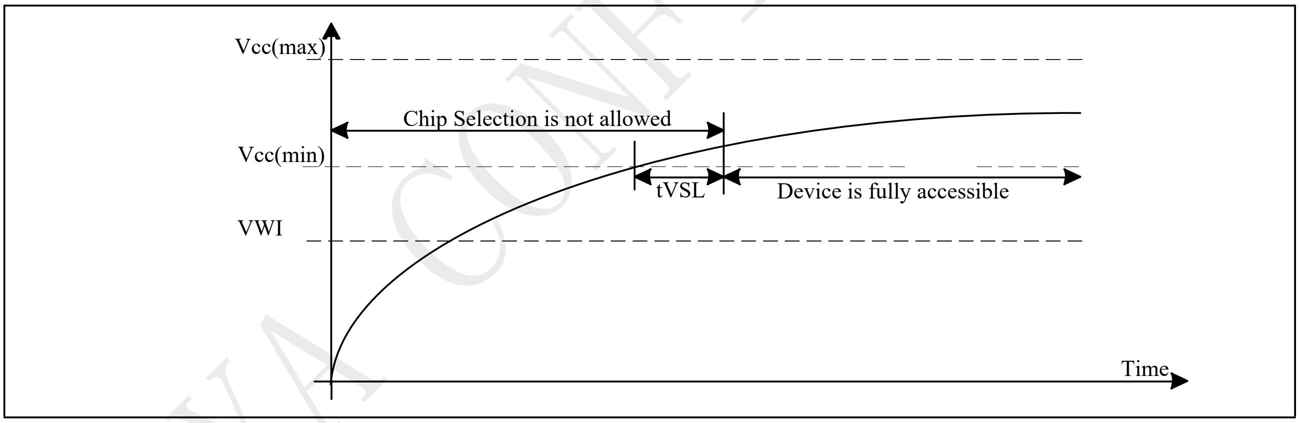


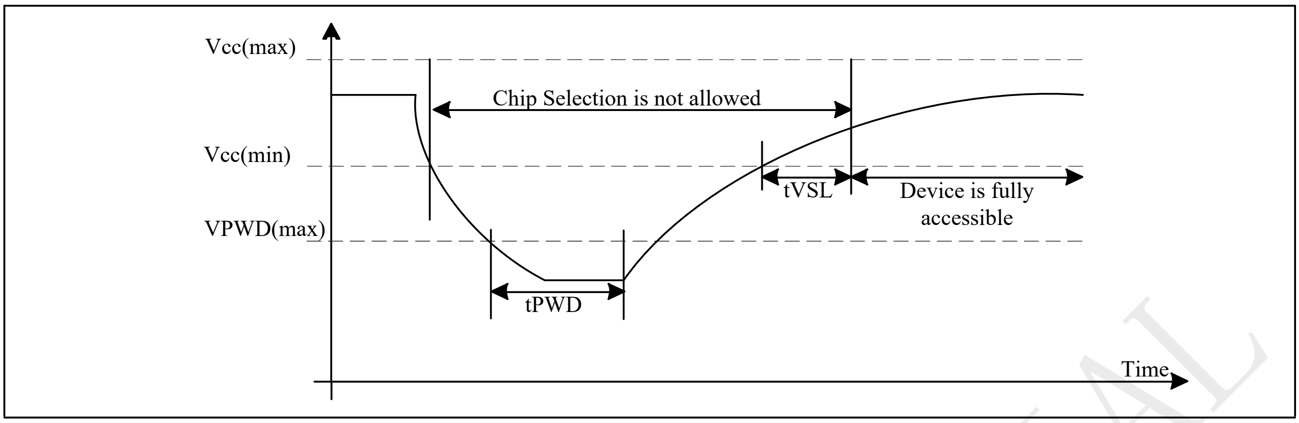
Figure 5-7 Power-Up Timing



Power Up/Down and Voltage Drop

For Power-down to Power-up operation, the VCC of flash device must below VPWD for at least tPWD timing. Please check the table below for more detail.

Figure 5-8 Power down-up Timing



Symbol	Parameter	Min	Max	Units
VPWD	VCC voltage needed to below VPWD for ensuring initialization will occur		1	V
tPWD	The minimum duration for ensuring initialization will occur	300		us
tVSL[1]	Vcc(min.) to device operation	2.5		ms
tVR	Vcc Rise Time	1	500000	us/V
VWI	Write Inhibit Voltage	1.1	1.4	V

Note: 1. tVSL average time is 500us except for power down at erase mode.

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).

6 Data Protection

During power transition, there may be some false system level signals which result in inadvertent erasure or programming. The device is designed to protect itself from these accidental write cycles.

The state machine will be reset as standby mode automatically during power up. In addition, the control register architecture of the device constrains that the memory contents can only be changed after specific command sequences have completed successfully.

In the following, there are several features to protect the system from the accidental write cycles during VCC power-up and power-down or from system noise.

- Power-on reset: to avoid sudden power switch by system power supply transition, the power-on reset may protect the Flash.
- Valid command length checking: The command length will be checked whether it is at byte base and completed on byte boundary.
- Write Enable (WREN) command: WREN command is required to set the Write Enable Latch bit (WEL) before issuing other commands to change data.
- Software Protection Mode: The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits define the section of the memory array that can be read but not change.
- Deep Power-Down Mode: By entering deep power down mode, the flash device is under protected from writing all commands except the Release from Deep Power-Down Mode command.

Protected Area Sizes

Table 6-1. PY25R128LA Protected Area Sizes (WPS=0, CMP bit = 0)

Status Register					Memory content			
BP4	BP3	BP2	BP1	BP0	Blocks	Addresses	Density	Portion
X	X	0	0	0	NONE	NONE	NONE	NONE
0	0	0	0	1	252 to 255	FC0000h – FFFFFFh	256KB	Upper 1/64
0	0	0	1	0	248 to 255	F80000h – FFFFFFh	512KB	Upper 1/32
0	0	0	1	1	240 to 255	F00000h – FFFFFFh	1MB	Upper 1/16
0	0	1	0	0	224 to 255	E00000h – FFFFFFh	2MB	Upper 1/8
0	0	1	0	1	192 to 255	C00000h – FFFFFFh	4MB	Upper 1/4
0	0	1	1	0	128 to 255	800000h – FFFFFFh	8MB	Upper 1/2
0	1	0	0	1	0 to 3	000000h – 03FFFFh	256KB	Lower 1/64
0	1	0	1	0	0 to 7	000000h – 07FFFFh	512KB	Lower 1/32
0	1	0	1	1	0 to 15	000000h – 0FFFFFh	1MB	Lower 1/16
0	1	1	0	0	0 to 31	000000h – 1FFFFFh	2MB	Lower 1/8
0	1	1	0	1	0 to 63	000000h – 3FFFFFh	4MB	Lower 1/4
0	1	1	1	0	0 to 127	000000h – 7FFFFFh	8MB	Lower 1/2
X	X	1	1	1	0 to 255	000000h – FFFFFFh	16MB	ALL
1	0	0	0	1	255	FFF000h – FFFFFFh	4KB	U - 1/4096
1	0	0	1	0	255	FFE000h – FFFFFFh	8KB	U - 1/2048
1	0	0	1	1	255	FFC000h – FFFFFFh	16KB	U - 1/1024
1	0	1	0	X	255	FF8000h – FFFFFFh	32KB	U - 1/512
1	0	1	1	0	255	FF8000h – FFFFFFh	32KB	U - 1/512
1	1	0	0	1	0	000000h – 000FFFh	4KB	L - 1/4096
1	1	0	1	0	0	000000h – 001FFFh	8KB	L - 1/2048
1	1	0	1	1	0	000000h – 003FFFh	16KB	L - 1/1024
1	1	1	0	X	0	000000h – 007FFFh	32KB	L - 1/512
1	1	1	1	0	0	000000h – 007FFFh	32KB	L - 1/512

Table 6-2. PY25R128LA Protected Area Sizes (WPS=0, CMP bit = 1)

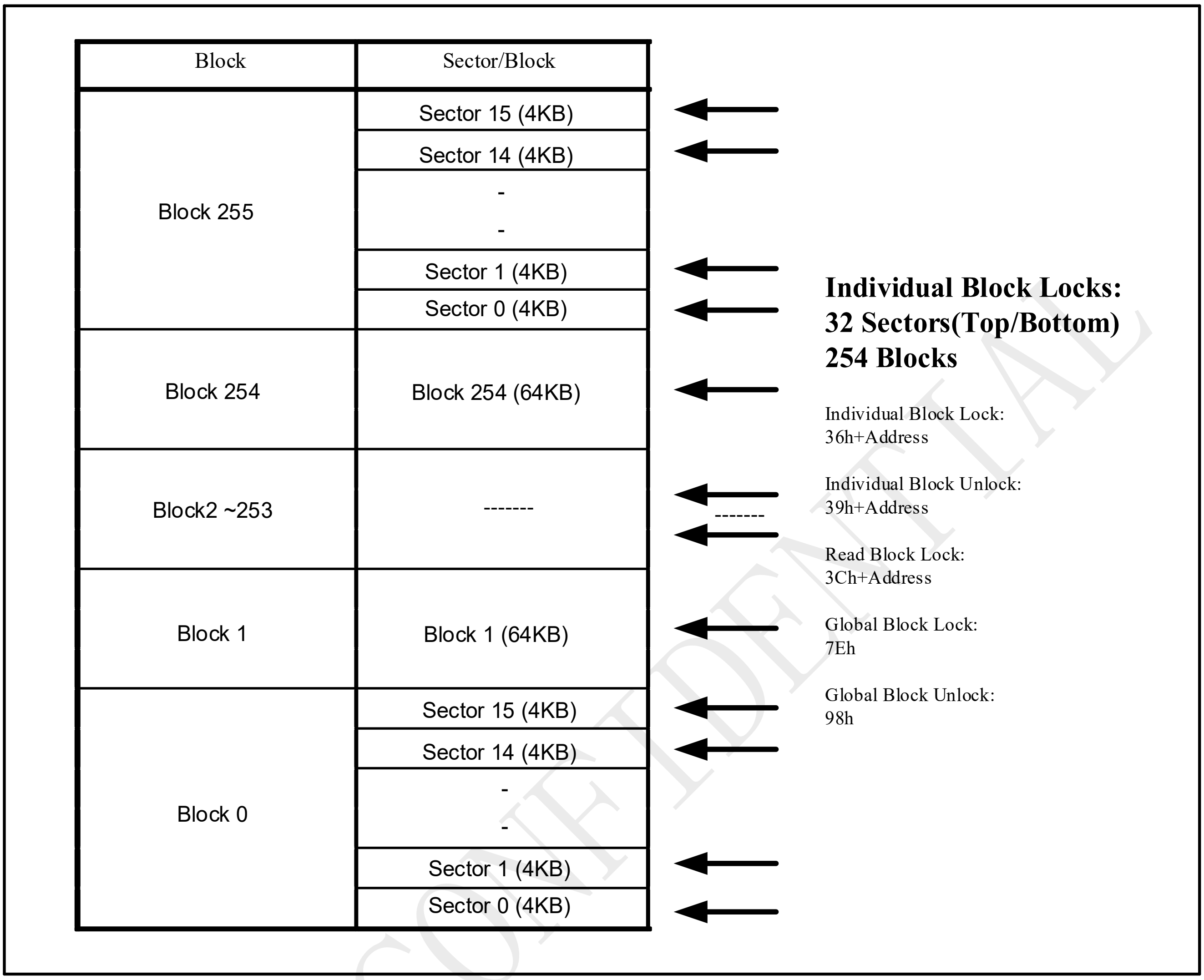
Status Register					Memory content			
BP4	BP3	BP2	BP1	BP0	Blocks	Addresses	Density	Portion
X	X	0	0	0	0 to 255	000000h - FFFFFFh	16MB	ALL
0	0	0	0	1	0 to 251	000000h - FBFFFFh	16,128KB	Lower 63/64
0	0	0	1	0	0 to 247	000000h – F7FFFFh	15,872KB	Lower 31/32
0	0	0	1	1	0 to 239	000000h - EFFFFFh	15MB	Lower 15/16
0	0	1	0	0	0 to 223	000000h - DFFFFFh	14MB	Lower 7/8
0	0	1	0	1	0 to 191	000000h - BFFFFFh	12MB	Lower 3/4
0	0	1	1	0	0 to 127	000000h - 7FFFFFh	8MB	Lower 1/2
0	1	0	0	1	4 to 255	040000h - FFFFFFh	16,128KB	Upper 63/64
0	1	0	1	0	8 to 255	080000h - FFFFFFh	15,872KB	Upper 31/32
0	1	0	1	1	16 to 255	100000h - FFFFFFh	15MB	Upper 15/16
0	1	1	0	0	32 to 255	200000h - FFFFFFh	14MB	Upper 7/8
0	1	1	0	1	64 to 255	400000h - FFFFFFh	12MB	Upper 3/4
0	1	1	1	0	128 to 255	800000h - FFFFFFh	8MB	Upper 1/2
X	X	1	1	1	NONE	NONE	NONE	NONE
1	0	0	0	1	0 to 255	000000h – FFEFFFh	16,380KB	L - 4095/4096
1	0	0	1	0	0 to 255	000000h – FFDFFFh	16,376KB	L - 2047/2048
1	0	0	1	1	0 to 255	000000h – FFBFFFh	16,368KB	L - 1023/1024
1	0	1	0	X	0 to 255	000000h – FF7FFFh	16,352KB	L - 511/512
1	0	1	1	0	0 to 255	000000h – FF7FFFh	16,352KB	L - 511/512
1	1	0	0	1	0 to 255	001000h – FFFFFFh	16,380KB	U - 4095/4096
1	1	0	1	0	0 to 255	002000h – FFFFFFh	16,376KB	U - 2047/2048
1	1	0	1	1	0 to 255	004000h – FFFFFFh	16,368KB	U -1023/1024
1	1	1	0	X	0 to 255	008000h – FFFFFFh	16,352KB	U - 511/512
1	1	1	1	0	0 to 255	008000h – FFFFFFh	16,352KB	U - 511/512

Note:

1. X=don't care

2. If any erase or program command specifies a memory that contains protected data portion, this command will be ignored.

Table 6-3. PY25R128LA Individual Block Protection (WPS=1)



Block	Sector/Block
Block 255	Sector 15 (4KB)
	Sector 14 (4KB)
	- -
	Sector 1 (4KB)
	Sector 0 (4KB)
Block 254	Block 254 (64KB)
Block2 ~253	-------
Block 1	Block 1 (64KB)
Block 0	Sector 15 (4KB)
	Sector 14 (4KB)
	- -
	Sector 1 (4KB)
	Sector 0 (4KB)

Individual Block Locks:
32 Sectors(Top/Bottom)
254 Blocks

Individual Block Lock:
36h+Address

Individual Block Unlock:
39h+Address

Read Block Lock:
3Ch+Address

Global Block Lock:
7Eh

Global Block Unlock:
98h

Note:

1. Individual Block/Sector protection is only valid when WPS=1.
2. All individual block/sector lock bits are set to 1 by default after power up, all memory array is protected.

7 Memory Address Mapping

The memory array can be erased in three levels of granularity including a full chip erase. The size of the erase blocks is optimized for both code and data storage applications, allowing both code and data segments to reside in their own erase regions.

Each device has	Each block has	Each sector has	Each page has	
16M	64/32K	4K	256	bytes
64K	256/128	16	-	pages
4096	16/8	-	-	sectors
256/512	-	-	-	blocks

PY25R128LA Memory Organization

Block	Sector	Address range	
255	4095	FFF000H	FFFFFFH
	……	……	……
	4080	FF0000H	FF0FFFH
254	4079	FEF000H	FEFFFFH
	……	……	……
	4064	FE0000H	FE0FFFH
……	……	……	……
	……	……	……
	……	……	……
……	……	……	……
	……	……	……
	……	……	……
2	47	02F000H	02FFFFH
	……	……	……
	32	020000H	020FFFH
1	31	01F000H	01FFFFH
	……	……	……
	16	010000H	010FFFH
0	15	00F000H	00FFFFH
	……	……	……
	0	000000H	000FFFH

8 Device Operation

Before a command is issued, status register should be checked to ensure device is ready for the intended operation.

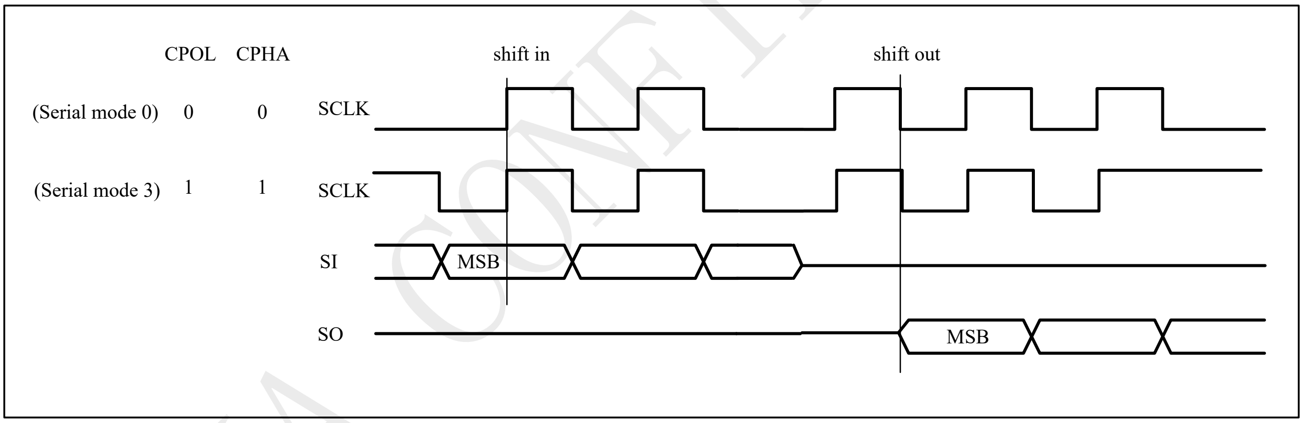
When incorrect command is inputted to this LSI, this LSI becomes standby mode and keeps the standby mode until next CS# falling edge. In standby mode, SO pin of this LSI should be High-Z. When correct command is inputted to this LSI, this LSI becomes active mode and keeps the active mode until next CS# rising edge.

Input data is latched on the rising edge of Serial Clock (SCLK) and data shifts out on the falling edge of SCLK. The difference of serial peripheral interface mode 0 and mode 3 is shown as Figure 8-1.

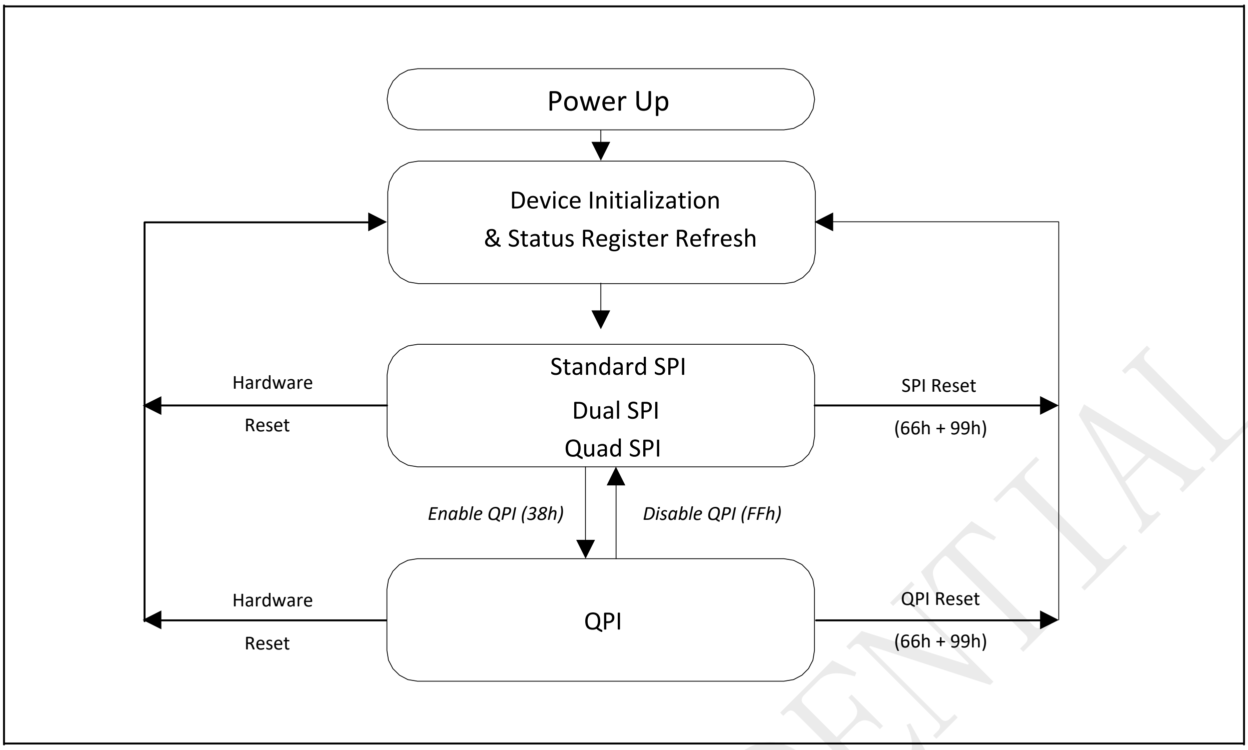
For the following instructions: RDID, RDSR, RDSR1, RDSCUR, READ, FREAD, DREAD, 2READ, 4READ, QREAD, RDSFDP, RES, REMS, DREMS, QREMS, the shifted-in instruction sequence is followed by a data-out sequence. After any bit of data being shifted out, the CS# can be high. For the following instructions: WREN, WRDI, WRSR, SE, BE32K, BE, CE, PP, QPP, DP, ERSCUR, PRSCUR, SUSPEND, RESUME, RSTEN, RST, the CS# must go high exactly at the byte boundary; otherwise, the instruction will be rejected and not executed.

During the progress of Write Status Register, Program, Erase operation, to access the memory array is neglected and not affect the current operation of Write Status Register, Program, Erase.

Figure 8-1 Serial Peripheral Interface Modes Supported



Note: CPOL indicates clock polarity of serial master, CPOL=1 for SCLK high while idle, CPOL=0 for SCLK low while not transmitting. CPHA indicates clock phase. The combination of CPOL bit and CPHA bit decides which serial mode is supported.



Standard SPI

The PY25R128LA features a serial peripheral interface on 4 signals bus: Serial Clock (SCLK), Chip Select (CS#), Serial Data Input (SI) and Serial Data Output (SO). Both SPI bus mode 0 and 3 are supported. Input data is latched on the rising edge of SCLK and data shifts out on the falling edge of SCLK.

Dual SPI

The PY25R128LA supports Dual SPI operation when using the “Dual Output Fast Read” and “Dual I/O Fast Read” (3BH and BBH) commands. These commands allow data to be transferred to or from the device at two times the rate of the standard SPI. When using the Dual SPI command, the SI and SO pins become bidirectional I/O pins: IO0 and IO1.

Quad SPI

The PY25R128LA supports Quad SPI operation when using the “Quad Output Fast Read”, “Quad I/O Fast Read” (6BH, EBH) commands. These commands allow data to be transferred to or from the device at four times the rate of the standard SPI. When using the Quad SPI command, the SI and SO pins become bidirectional I/O pins: IO0 and IO1. For PY25R128LA, the QE bit is set to 1 as default and cannot be changed.

QPI

The PY25R128LA supports Quad Peripheral Interface (QPI) operations only when the device is switched from Standard/Dual/Quad SPI mode to QPI mode using the “Enable the QPI (38H)” command. The QPI mode utilizes all four IO pins to input the command code. Standard/Dual/Quad SPI mode and QPI mode are exclusive. Only one mode can be active at any given times. “Enable the QPI (38H)”and “Disable the QPI(FFH)” commands are used to switch between these two modes. Upon power-up and after software reset using “” Reset (99H)” command, the default state of the device is Standard/Dual/Quad SPI mode. For PY25R128LA, the QE bit is set to 1 as default and cannot be changed.

SPI / QPI DTR Read Instructions

To effectively improve the read operation throughput without increasing the serial clock frequency, PY25R128LA introduces multiple DTR (Double Transfer Rate) Read instructions that support Standard/Dual/Quad SPI and QPI modes. The byte-long instruction code is still latched into the device on the rising edge of the serial clock similar to all other SPI/QPI instructions. Once a DTR instruction code is accepted by the device, the address input and data output will be latched on both rising and falling edges of the serial clock.

Software Reset & Hardware RESET# pin

The PY25R128LA can be reset to the initial power-on state by a software Reset sequence, either in SPI mode or QPI mode. This sequence must include two consecutive commands: Enable Reset (66h) & Reset (99h). If the command sequence is successfully accepted, the device will take approximately 30uS (tReady) to reset. No command will be accepted during the reset period.

For the SOIC-16, PY25R128LA provides a dedicated hardware/RESET pin. Drive the /RESET pin low for a minimum period of 1uS(tRESET) will interrupt any on-going external/internal operations of the flash device and reset the device to its initial power-on state, while RPMC not support hardware reset. Hardware/RESET pin has higher priority than other SPI input signals (/CS, CLK, IOs).

Notes:
1. Hardware/RESET pin is available on SOIC-16 package only.

RPMC OPERATIONS

To prevent hardware vulnerability attack, RPMC provides a building block towards providing Confidentiality and Integrity of read/write flash memory data. The PY25R128LA is equipped with four 32-bit Monotonic counters and can be addressed individually by the 8-bit Counter address. These Monotonic counters are used by the SPI flash controllers to ensure the physical authenticity of the attached flash devices.

RPMC operation is based on the HMAC-SHA-256 cryptographic algorithm. HMAC-SHA-256 is a type of keyed hash algorithm that is constructed from the SHA-256 hash function and used as a Hash-based Message Authentication Code (HMAC). The HMAC process mixes a secret key with the message data, hashes the result with the hash function, mixes that hash value with the secret key again, and then applies the hash function a second time. The output hash is 256 bits in length.

An HMAC can be used to determine whether a message sent over an insecure channel has been tampered with, provided that the sender and receiver share a secret key. The sender computes the hash value for the original data and sends both the original data and hash value as a single message. The receiver recalculates the hash value on the received message and checks that the computed HMAC matches the transmitted HMAC.

Any change to the data or the hash value results in a mismatch, because knowledge of the secret key is required to change the message and reproduce the correct hash value. Therefore, if the original and computed hash values match, the message is authenticated.

RPMC Initialization

RPMC operation is initialized in an OEM manufacturing environment by issuing the “Write Root Key register” command. When this command is successfully received and executed, a 256-bit Root Key will be written into the flash device permanently, and the corresponding Monotonic counters will also be initialized to 0. After the initialization procedure, the Root Key value can no longer be altered or accessed externally through the SPI interface. The initialized Monotonic counters is ready to accept the authenticated RPMC commands.

RPMC Operation

Once the root key and the Monotonic counters have been initialized, every time after the device is powered on, prior to any RPMC operations, the external SPI flash controller must update the HMAC Key register in the PY25R128LA.

After initializing the HMAC key register, there are two different RPMC operations can be performed. “Increment Monotonic counters” is used to increase the Monotonic counters value by 1. “Request Monotonic counters” is used to read out the existing Monotonic counters data.

Operations Allowed / Disallowed During RPMC Operation

The RPMC operation is independent to the other SPI flash operations. The RPMC input command OP1 (9Bh) will initial internal operations after the authenticated command is accepted by the device. During the internal operation period, the BUSY bit in the RPMC Status Register (bit 0) will be set to 1. The RPMC internal operation cannot be suspended and can only be interrupted by the Device Reset command (66h/99h). While the RPMC internal operation is going on, other SPI flash commands can be issued and executed. Please refer to the table below for details.

Flash Operations	RPMC Behavior
Suspend/Resume	RPMC OP1/OP2 operations cannot be suspended.
Hard Reset#	RPMC cannot be hardware reset.
Hold	RPMC don’t support hold function.
SPI flash memory read/ program/ erase	All SPI flash memory read/ program/ erase commands can be accepted while a RPMC OP1/OP2 operation is on-going, vice versa.
RPMC OP1 command	RPMC OP1 command will be ignored while a RPMC operation is on-going.
RPMC OP2 command read status/data	RPMC OP2 command can be accepted while a RPMC OP1 operation is on-going. This is the way to polling if the RPMC operation has finished or not.
Soft reset	The soft reset sequence can be issued any time during SPI flash or RPMC operations.

9 Commands

9.1 Commands listing

Figure 9-1 Command set (Standard/Dual/Quad SPI)

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function description
Read						
Fast Read Array	FREAD	0BH	3	8	1+	n bytes read out until CS# goes high
Normal Read Array	READ	03H	3	0	1+	n bytes read out until CS# goes high
Read Dual Output	DREAD	3BH	3	8	1+	n bytes read out by Dual output
Read 2IO	2READ	BBH	3	4(8)	1+	n bytes read out by 2IO
Read Quad Output	QREAD	6BH	3	8	1+	n bytes read out by Quad output
Read 4IO	4READ	EBH	3	6(10)	1+	n bytes read out by 4IO
Read Word 4IO	WREAD	E7H	3	4	1+	n bytes word read out by 4IO
Program and Erase						
Sector Erase (4K bytes)	SE	20H	3	0	0	erase selected sector
Block Erase (32K bytes)	BE32	52H	3	0	0	erase selected 32K block
Block Erase (64K bytes)	BE	D8H	3	0	0	erase selected 64K block
Chip Erase	CE	60H/C7H	0	0	0	erase whole chip
Page Program	PP	02H	3	0	1+	program selected page
Quad page program	QPP	32H	3	0	1+	quad input to program selected page
Program/Erase Suspend	PES	75H	0	0	0	suspend program/erase operation
Program/Erase Resume	PER	7AH	0	0	0	continue program/erase operation
Protection						
Write Enable	WREN	06H	0	0	0	sets the write enable latch bit
Write Disable	WRDI	04H	0	0	0	resets the write enable latch bit
Volatile SR/CR Write Enable	VWREN	50H	0	0	0	Write enable for volatile SR/CR
Individual Block Lock	SBLK	36H	3	0	0	Individual block lock
Individual Block Unlock	SBULK	39H	3	0	0	Individual block unlock
Read Block Lock Status	RDBLOCK	3DH	3	0	0	Read individual block lock register
Global Block Lock	GBLK	7EH	0	0	0	Whole chip block protect
Global Block Unlock	GBULK	98H	0	0	0	Whole chip block unprotect
Security						
Erase Security Registers	ERSCUR	44H	3	0	0	Erase security registers
Program Security Registers	PRSCUR	42H	3	0	1+	Program security registers
Read Security Registers	RDSCUR	48H	3	8	1+	Read value of security register

Command set (Standard/Dual/Quad SPI)

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function
Status Register						
Read Status Register	RDSR	05H	0	0	1	read out status register
Read Status Register-1	RDSR1	35H	0	0	1	Read out status register-1
Read Configure Register	RDCR	15H	0	0	1	Read out configure register
Write Status Register	WRSR	01H	0	0	1-2	Write data to status registers and status registers-1
Write Status Register-1	WRSR1	31H	0	0	1	Write data to status registers-1

Write Configure Register	WRCR	11H	0	0	1	Write data to configuration register
Other Commands						
Reset Enable	RSTEN	66H	0	0	0	Enable reset
Reset	RST	99H	0	0	0	Reset
Enable QPI	QPIEN	38H	0	0	0	Enable QPI mode
Read Manufacturer/device ID	RDID	9FH	0	0	1 to 3	output JEDEC ID: 1-byte manufacturer ID & 2-byte device ID
Read Manufacture ID	REMS	90H	3		1+	Read manufacturer ID/device ID data
Dual Read Manufacture ID	DREMS	92H	3	4	1+	Dual output read manufacture/device ID
Quad Read Manufacture ID	QREMS	94H	3	6	1+	Quad output read manufacture/device ID
Deep Power-down	DP	B9H	0	0	0	enters deep power-down mode
Release Deep Power-down/Read Electronic ID	RDP/RES	ABH	3	0	1	Read electronic ID data
Set burst length	SBL	77H	0	0	0	Set burst length
Read SFDP	RDSFDP	5AH	3	8	1+	Read SFDP parameter
Release read enhanced		FFH				Release from read enhanced
Read unique ID	RUID	4BH	3	8	1+	Read unique ID

Command set (QPI Instructions)

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function description
Read						
Burst Read with Wrap		0CH	3	10/4/6/8	1+	n bytes burst read with wrap by 4IO
Read Word 4x I/O		EBH	3	10/4/6/8	1+	n bytes read out by 4IO
Program and Erase						
Page Program	PP	02H	3	0	1+	program selected page
Sector Erase (4K bytes)	SE	20H	3	0	0	erase selected sector
Block Erase (32K bytes)	BE32	52H	3	0	0	erase selected 32K block
Block Erase (64K bytes)	BE	D8H	3	0	0	erase selected 64K block
Chip Erase	CE	60H/C7H	0	0	0	erase whole chip
Program/Erase Suspend	PES	75H	0	0	0	suspend program/erase operation
Program/Erase Resume	PER	7AH	0	0	0	continue program/erase operation
Protection						
Write Enable	WREN	06H	0	0	0	sets the write enable latch bit
Volatile SR/CR Write Enable	VWREN	50H	0	0	0	Write enable for volatile SR/CR
Write Disable	WRDI	04H	0	0	0	resets the write enable latch bit
Individual Block Lock	SBLK	36H	3	0	0	Individual block lock
Individual Block Unlock	SBULK	39H	3	0	0	Individual block unlock
Read Block Lock Status	RDBLOCK	3DH	3	0	0	Read individual block lock register
Global Block Lock	GBLK	7EH	0	0	0	Whole chip block protect

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function description
Global Block Unlock	GBULK	98H	0	0	0	Whole chip block unprotect
Status Register						
Read Status Register	RDSR	05H	0	0	1	read out status register
Read Status Register-1	RDSR1	35H	0	0	1	Read out status register-1
Read Configure Register	RDCR	15H	0	0	1	Read out configure register
Write Status Register	WRSR	01H	0	0	1-2	Write data to status registers-0 and status registers-1
Write Status Register-1	WRSR1	31H	0	0	1	Write data to status registers-1
Write Configure Register	WRCR	11H	0	0	1	Write data to configuration register
Other Commands						
Deep Power-down	DP	B9H	0	0	0	enters deep power-down mode
Release Deep Power-down/Read Electronic ID	RDP/RES	ABH	3	0	1	Read electronic ID data
Set Read Parameters	SRP	C0H	0	0	1	Set read dummy and wrap
Read Manufacture ID	REMS	90H	3		1+	Read manufacturer ID/device ID data
Read Manufacturer/device ID	RDID	9FH	0	0	1 to 3	output JEDEC ID: 1-byte manufacturer ID & 2-byte device ID
Read SFDP	RDSFDP	5AH	3	10/4/6/8	1+	Read SFDP parameter
Disable QPI		FFH				Release from read enhanced
Reset Enable	RSTEN	66H	0	0	0	Enable reset
Reset	RST	99H	0	0	0	Reset

Command set (DTR Instructions)

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function description
DTR Fast Read	DTRFRD	0DH	3	6	1+	DTR n byte fast read out
DTR 2IO Read	2DTRD	BDH	3	6	1+	DTR n byte read out by 2IO
DTR 4IO Read	4DTRD	EDH	3	10	1+	DTR n byte read out by 4IO

Command set (DTR QPI Instructions)

Commands	Abbr.	Code	ADR Bytes	DMY Cycles	Data Bytes	Function description
DTR Burst Read with Wrap		0EH	3	10	1+	DTR n bytes burst read with wrap by 4IO
DTR Fast Read	DTRFRD	0DH	3	10	1+	DTR n byte fast read out
DTR 4IO Read	4DTRD	EDH	3	10	1+	DTR n byte fast read out

NOTE:

1. Dual Output data

IO0 = (D6, D4, D2, D0)

IO1 = (D7, D5, D3, D1)

2. Dual Input Address

IO0 = A22, A20, A18, A16, A14, A12, A10, A8 A6, A4, A2, A0, M6, M4, M2, M0
IO1 = A23, A21, A19, A17, A15, A13, A11, A9 A7, A5, A3, A1, M7, M5, M3, M1

3. Quad Output Data
IO0 = (D4, D0, …)
IO1 = (D5, D1, …)
IO2 = (D6, D2, …)
IO3 = (D7, D3, …)

4. Quad Input Address
IO0 = A20, A16, A12, A8, A4, A0, M4, M0
IO1 = A21, A17, A13, A9, A5, A1, M5, M1
IO2 = A22, A18, A14, A10, A6, A2, M6, M2
IO3 = A23, A19, A15, A11, A7, A3, M7, M3

5. Fast Read Quad I/O Data
IO0 = (x, x, x, x, D4, D0, …)
IO1 = (x, x, x, x, D5, D1, …)
IO2 = (x, x, x, x, D6, D2, …)
IO3 = (x, x, x, x, D7, D3, …)

6. Fast Word Read Quad I/O Data
IO0 = (x, x, D4, D0, …)
IO1 = (x, x, D5, D1, …)
IO2 = (x, x, D6, D2, …)
IO3 = (x, x, D7, D3, …)

7. Fast Word Read Quad I/O Data: the lowest address bit must be 0.

8. QPI Command, Address, Data input/output format:
IO0= C4, C0, A20, A16, A12, A8, A4, A0, D4, D0, D4, D0,
IO1= C5, C1, A21, A17, A13, A9, A5, A1, D5, D1, D5, D1
IO2= C6, C2, A22, A18, A14, A10, A6, A2, D6, D2, D6, D2
IO3= C7, C3, A23, A19, A15, A11, A7, A3, D7, D3, D7, D3

9. Security Registers Address:
Security Register1: A23-A16=00H, A15-A10=000100, A9-A0= Byte Address;
Security Register2: A23-A16=00H, A15-A10=001000, A9-A0= Byte Address;
Security Register3: A23-A16=00H, A15-A10=001100, A9-A0= Byte Address;

Command set (RPMC OP1 Instructions)

RPMC Operation	Command (Byte0)	Cmd Type (Byte1)	Addr Bytes (Byte2)	Reserved (Byte3)	Data Bytes	Signature Bytes
Write Root Key Register	9BH	00H	Counter Addr[7:0]	00H	Byte4-35 RootKey[255:0]	Byte36-63 TruncatedSign[223:0]
Update HMAC Key Register	9BH	01H	Counter Addr[7:0]	00H	Byte4-7 KeyData[31:0]	Byte8-39 Signature[255:0]

Increase Monotonic Counter	9BH	02H	Counter Addr[7:0]	00H	Byte4-7 CounterData[31:0]	Byte8-39 Signature[255:0]
Request Monotonic Counter	9BH	03H	Counter Addr[7:0]	00H	Byte4-15 Tag[95:0]	Byte16-47 Signature[255:0]
Reserved Commands	9BH	04H ~ FFH	Reserved			

Command set (RPMC OP2 Instructions)

RPMC Operation	Command (Byte0)	Dummy (Byte1)	Data Output			
Read RPMC Status / Data	96H	dummy	Byte2 Status[7:0]	Byte3-14 Tag[95:0]	Byte15-18 CounterData[31:0]	Byte19-50 Signature[255:0]

Note:

1. All RPMC instructions are in Standard SPI format. Each Input/Output Byte requires 8 clocks.
2. The Reserved (Byte3) for RPMC OP1 must be all 0s (00000000’b).
3. The controller may terminate the Read RPMC Status/Data instruction at any time without going through the entire data output sequence.
4. When BUSY=1, the device will output the RPMC_Status[7:0] value continuously until /CS terminates the instruction. The device will not output Tag[95:0], CounterData[31:0] & Signature[255:0] fields when BUSY=1. Once BUSY becomes 0, another OP2 command must be issued to read out the correct Tag[95:0], CounterData[31:0] & Signature[255:0] fields.

9.2 Write Enable (WREN) (06H)

The Write Enable (WREN) instruction is for setting Write Enable Latch (WEL) bit. For those instructions like PP, QPP, SE, BE32K, BE, CE and WRSR, WRSR1, WRCR, ERSCUR, PRSCUR which are intended to change the device content, should be set every time after the WREN instruction setting the WEL bit.

The sequence of issuing WREN instruction is: CS# goes low→ sending WREN instruction code→ CS# goes high.

Figure 9-2 Write Enable (WREN) Sequence (Command 06h)

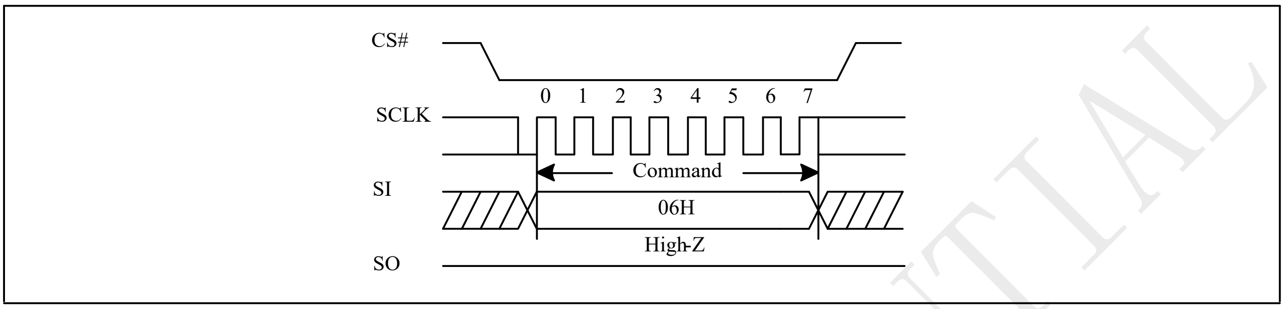
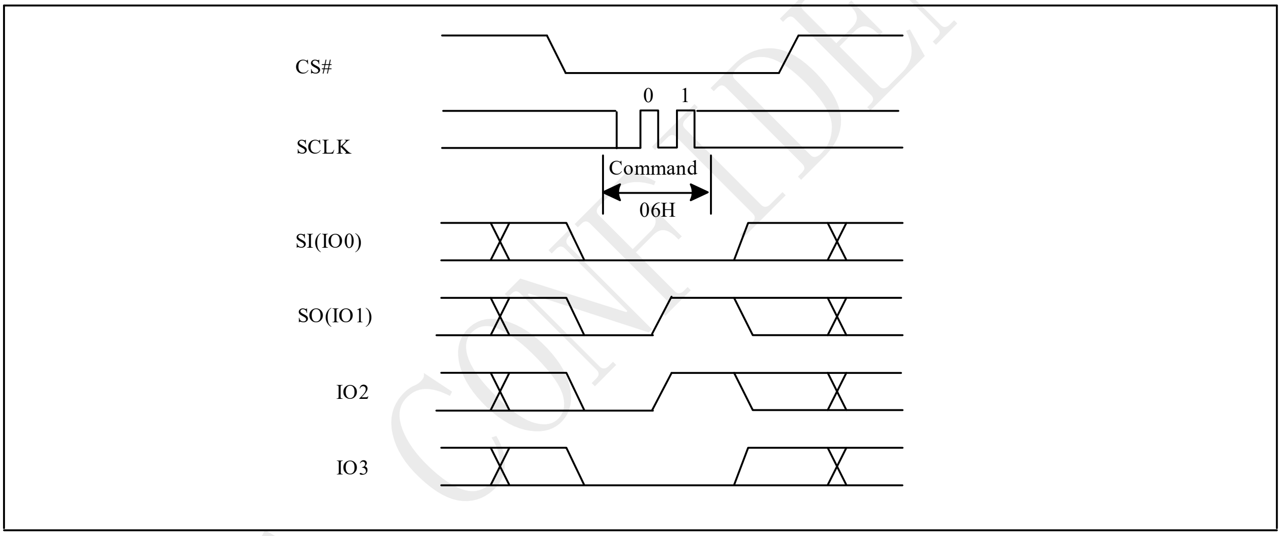


Figure 9-2a Write Enable (WREN) Sequence (QPI)



9.3 Write Disable (WRDI) (04H)

The Write Disable (WRDI) instruction is for resetting Write Enable Latch (WEL) bit.

The sequence of issuing WRDI instruction is: CS# goes low→ sending WRDI instruction code→ CS# goes high.

The WEL bit is reset by following situations:

- Power-up
- Write Disable (WRDI) instruction completion
- Write Status Register (WRSR/WRSR1/WRCR) instruction completion
- Page Program (PP) instruction completion
- Quod Page Program (QPP) instruction completion
- Sector Erase (SE) instruction completion
- Block Erase (BE32K, BE) instruction completion
- Chip Erase (CE) instruction completion
- Erase Security Register (ERSCUR) instruction completion
- Program Security Register (PRSCUR) instruction completion
- Reset (Soft Reset, Hard Reset) instruction completion
- Write Enable for Volatile Register

Figure 9-3 Write Disable (WRDI) Sequence (Command 04h)

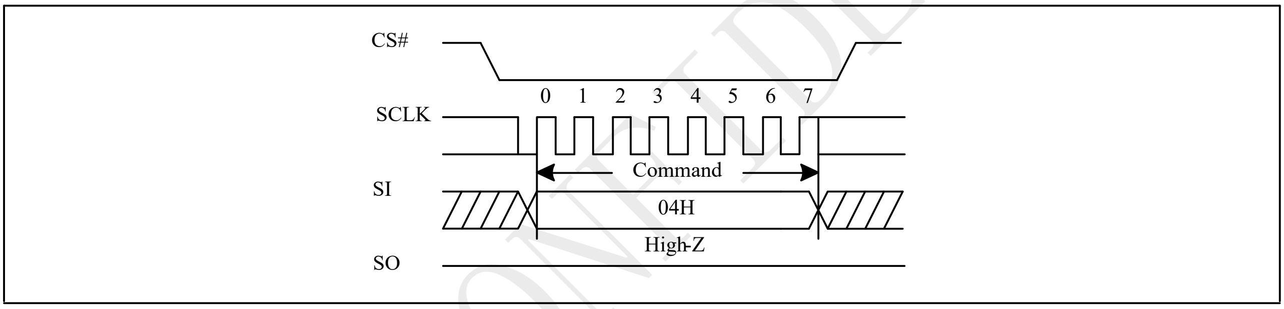
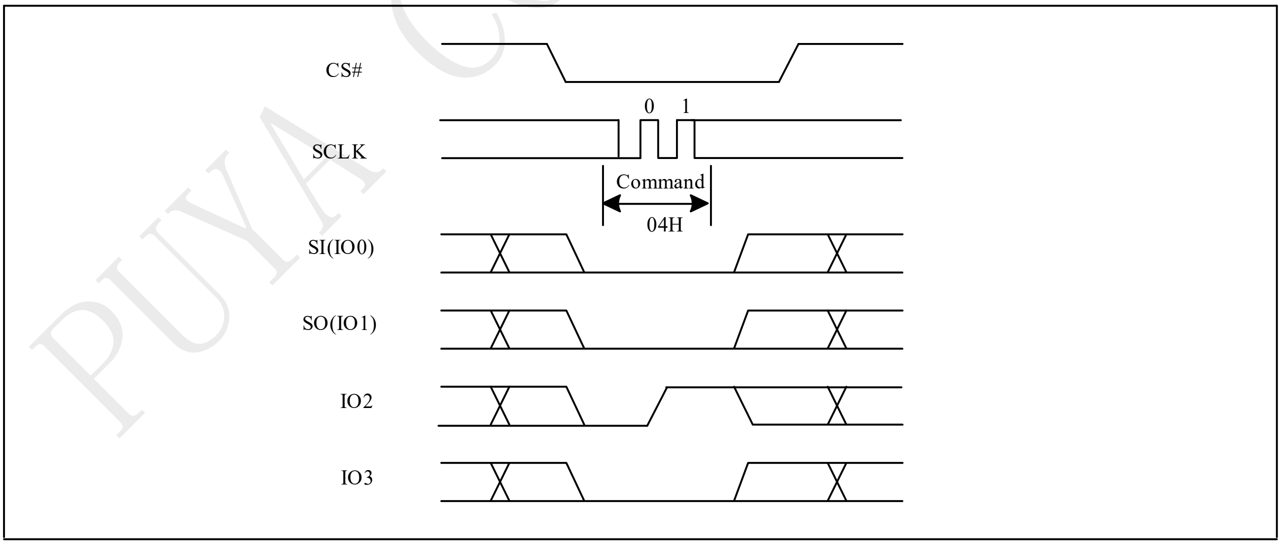


Figure 9-3a Write Disable (WRDI) Sequence (QPI)



9.4 Write Enable for Volatile Register (50H)

The non-volatile Status Register and Configure Register bits can also be written to as volatile bits. This gives more flexibility to change the system configuration and memory protection schemes quickly without waiting for the typical non-volatile bit write cycles or affecting the endurance of the Status Register and Configure Register non-volatile bits. The Write Enable for Volatile Register command must be issued prior to a Write Status Register or Write Configure Register command, and any other commands cannot be inserted between them. The Write Enable for Volatile Register command will not set the Write Enable Latch, LB3, LB2 and LB1 bits, it is only valid for the Write Status Register or Write Configure Register command to change the volatile Register bit values.

The sequence of issuing Write Enable for Volatile Register instruction is: CS# goes low→ sending Write Enable for Volatile Register instruction code→ CS# goes high.

Figure 9-4 Write Enable for Volatile Register Sequence (Command 50h)

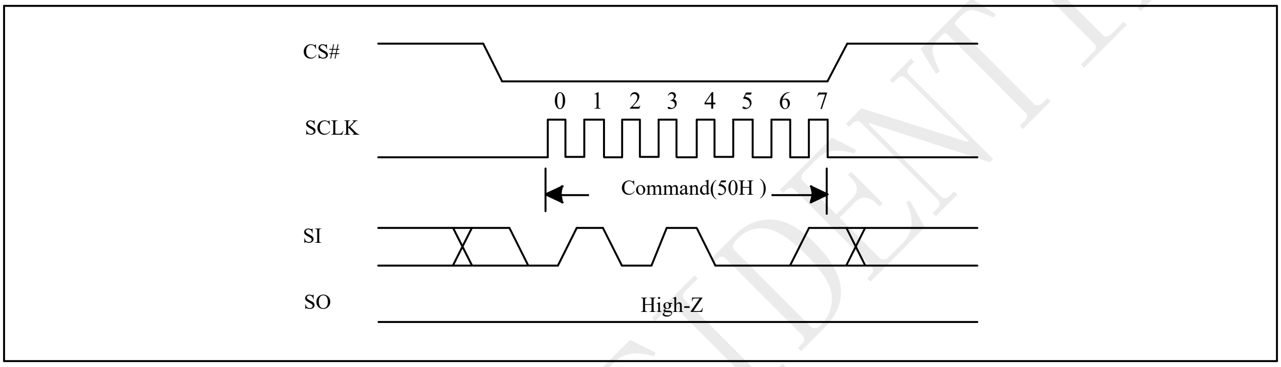
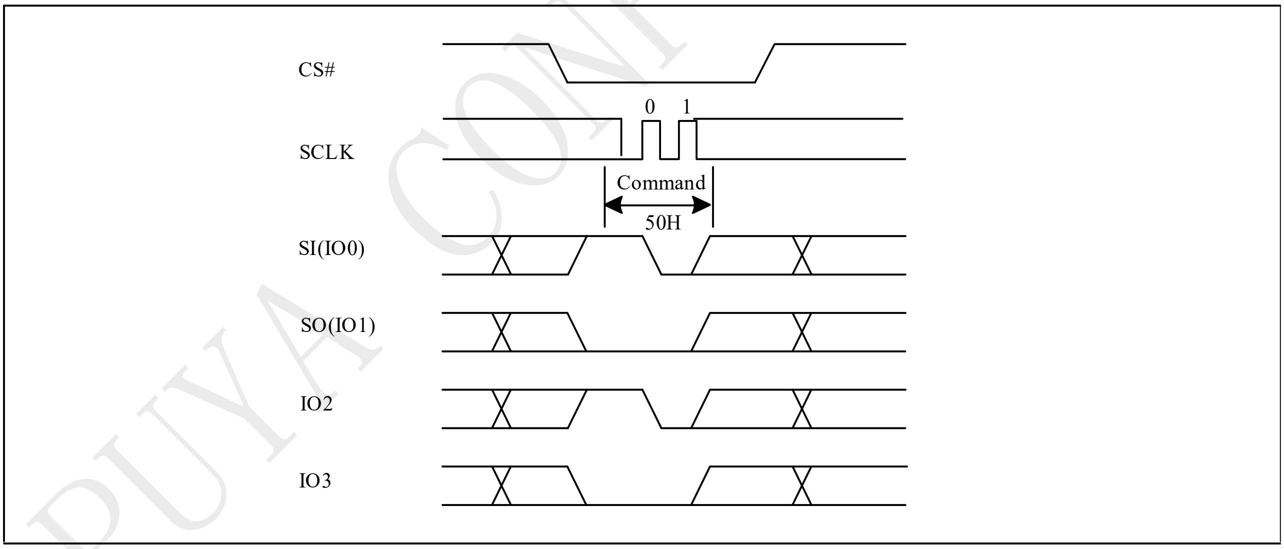


Figure 9-4a Write Enable for Volatile Register Sequence (QPI)



9.5 Read Status Register (RDSR) (05H/35H)

The RDSR instruction is for reading Status Register Bits. The Read Status Register can be read at any time (even in program/erase/write status register condition). It is recommended to check the Write in Progress (WIP) bit before sending a new instruction when a program, erase, or write status register operation is in progress. For command code “05H”, the SO will output Status Register bits S7~S0. The command code “35H”, the SO will output Status Register bits S15~S8.

The sequence of issuing RDSR instruction is: CS# goes low→ sending RDSR instruction code→ Status Register data out on SO. The SIO [3:1] are "don't care".

Figure 9-5 Read Status Register (RDSR) Sequence (Command 05h or 35h)

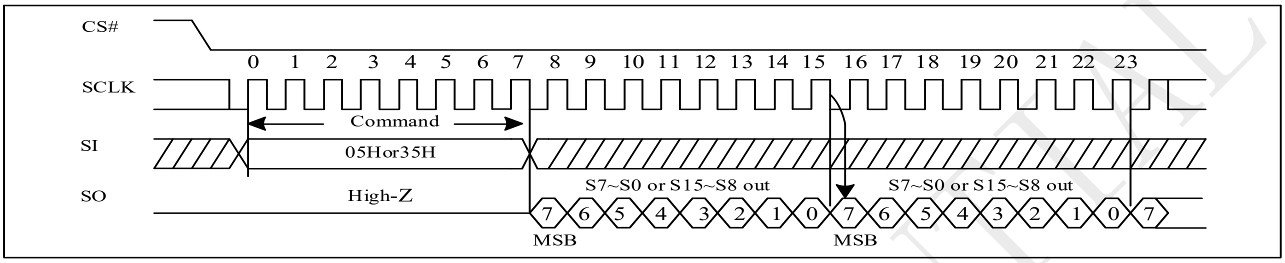
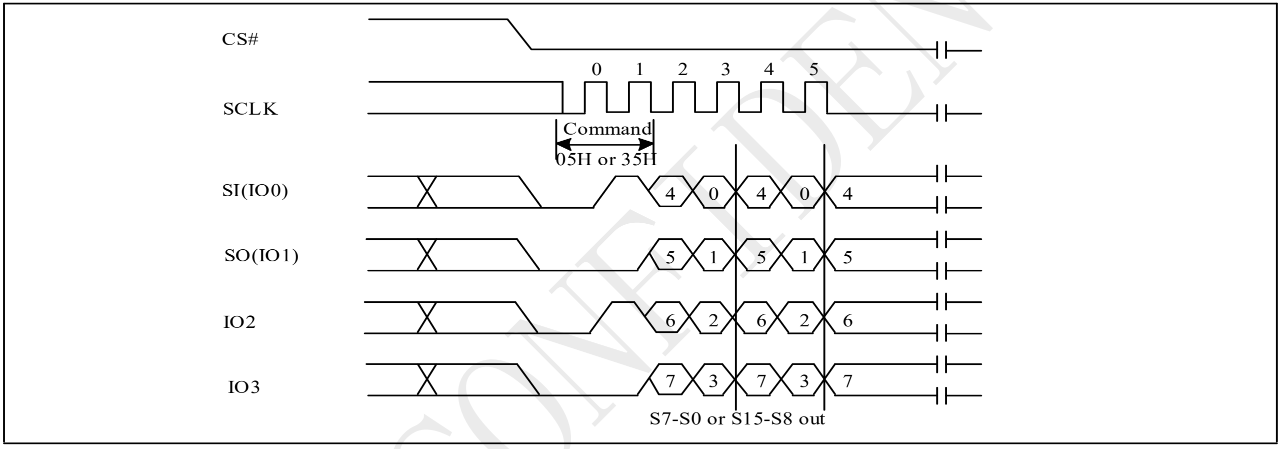


Figure 9-5a Read Status Register (RDSR) Sequence (QPI)



Status Register

BIT	S15	S14	S13	S12	S11	S10	S9	S8
Definition	SUS	CMP	LB3	LB2	LB1	EP_FAIL	QE	SRP1
Volatile	RO	N	OTP	OTP	OTP	RO	N	N
Default	0	0	0	0	0	0	1	0

BIT	S7	S6	S5	S4	S3	S2	S1	S0
Definition	SRP0	BP4	BP3	BP2	BP1	BP0	WEL	WIP
Volatile	N	N	N	N	N	N	V	RO
Default	0	0	0	0	0	0	0	0

Note: 1. RO=read only, N=non-volatile, V= volatile.

The definition of the status register bits is as below:

WIP bit.

The Write in Progress (WIP) bit indicates whether the memory is busy in program/erase/write status register progress. When WIP bit sets to 1, means the device is busy in program/erase/write status register progress, when WIP bit sets 0, means the device is not in program/erase/write status register 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 command is accepted.

BP4, BP3, BP2, BP1, BP0 bits.

The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits are non-volatile. They define the size of the area to be software protected against Program and Erase commands. These bits are written with the Write Status Register (WRSR) command. When the Block Protect (BP4, BP3, BP2, BP1, BP0) bits are set to 1, the relevant memory area (as defined in Table “Protected Area Sizes”). becomes protected against Page Program (PP), Sector Erase (SE) and Block Erase (BE) commands. The Block Protect (BP4, BP3, BP2, BP1, and BP0) bits can be written provided that the Hardware Protected mode has not been set. The Chip Erase (CE) command is executed, only if the Block Protect (BP4, BP3, BP2, BP1and BP0) are set to “None protected”.

SRP1, SRP0 bits.

The Status Register Protect (SRP1 and SRP0) bits are non-volatile Read/Write bits in the status register. The SRP bits control the method of write protection: software protection, hardware protection, power supply lock-down or one-time programmable protection

SRP1	SRP0	Status Register	Description
0	0	Software Protected	The Status Register and Configure Register can be written to after a Write Enable command.
1	0	Power Supply Lock-Down(1)	Status Register and Configure Register are protected and cannot be written to again until the next Power-Down, Power-Up cycle.
1	1	One Time Program(2)	Status Register and Configure Register are permanently protected and cannot be written to.

1. When SRP1, SRP0= (1, 0), Only a Power-Down, Power-Up cycle will change SRP1, SRP0 to (0, 0) state, not include soft reset and hard reset.
2. This feature is available on special order. Please contact PUYA for details.

QE bit.

The Quad Enable (QE) bit is an OTP bit in the Status Register which is permanently set to "1". The flash always performs Quad I/O mode.

EP_FAIL bit.

The Erase/Program Fail bit is a read only bit which shows the status of the last Program/Erase (PP, QPP, SE, BE32K, BE, CE and ERSCUR, PRSCUR) operation. The bit will be set to "1"if the program/erase operation failed or interrupted by reset or the program/erase region was protected. It will be automatically cleared to "0" if the next program/erase operation succeeds. Please note that it will not interrupt or stop any operation in the flash memory.

LB3, LB2, LB1, bits.

The LB3, LB2, LB1, bits are non-volatile One Time Program (OTP) bits in Status Register (S13-S11) that provide the write protect control and status to the Security Registers. The default state of LB3-LB1are0, the security registers are unlocked. The LB3-LB1bitscan be set to 1 individually using the Write Register instruction. The LB3-LB1bits are One Time Programmable, once it’s set to 1, the Security Registers will become read-only permanently.

CMP bit

The CMP bit is a non-volatile Read/Write bit in the Status Register(S14). It is used in conjunction the BP4-BP0 bits to provide more flexibility for the array protection. Please see the table “Protected Area Size” for details. The default setting is CMP=0.

SUS bit

The SUS bit is read only bit in the status register (S15) that is set to 1 after executing a Program/Erase Suspend (75H) command. The SUS bit is cleared to 0 by Program/Erase Resume (7AH) command and a power-down, power-up cycle, soft reset, hard reset.

9.6 Read Configure Register (RDCR) (15H)

The RDCR instruction is for reading Configure Register Bits. The Read Configure Register can be read at any time (even in program/erase/write status register condition).

The sequence of issuing RDCR instruction is: CS# goes low→ sending RDCR instruction code→ Configure Register data out on SO. The SIO [3:1] are "don't care".

Figure 9-6 Read Status Register (RDCR) Sequence (Command 15h)

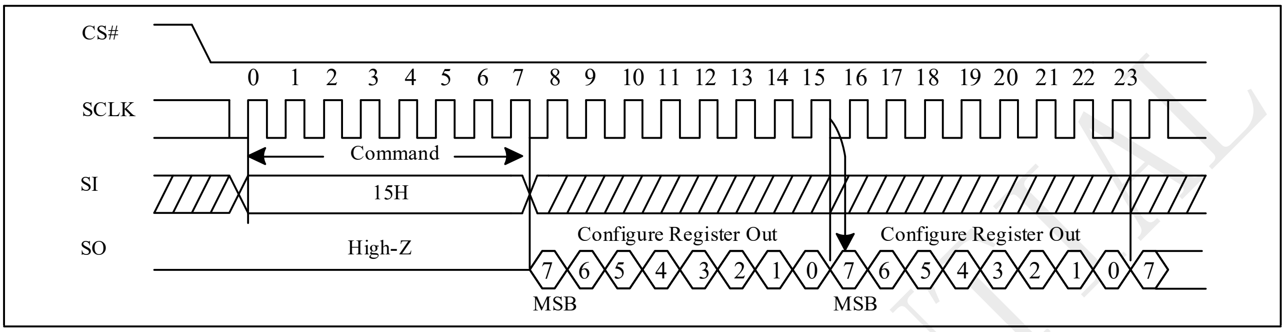
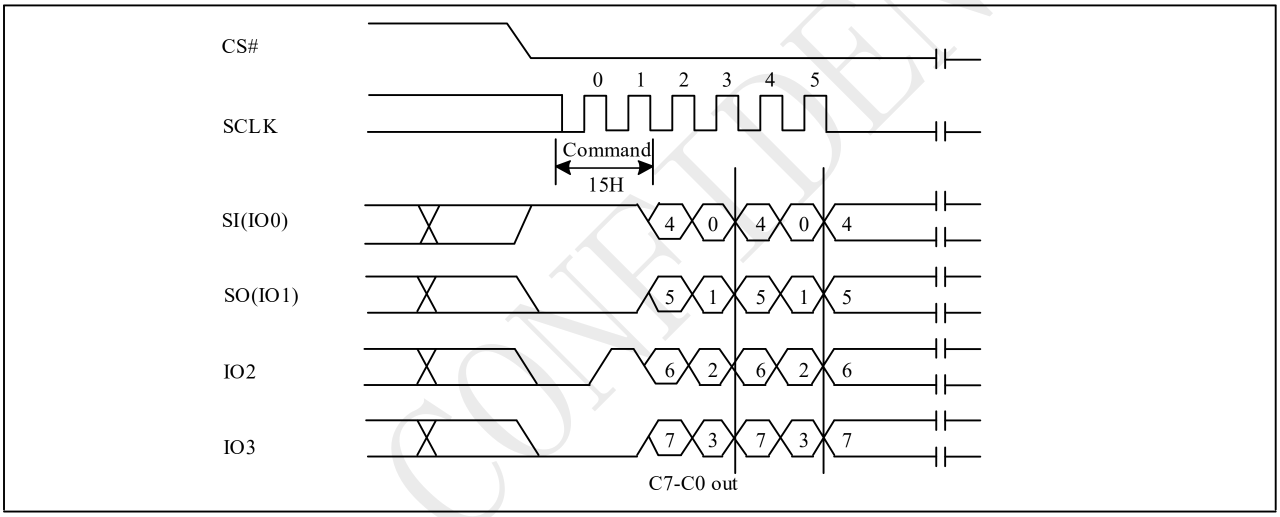


Figure 9-6a Read Status Register (RDCR) Sequence (QPI)



Configure Register

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Definition	Reserved	DRV1	DRV0	Reserved	Reserved	WPS	DC	DLP
Volatile	N	N	N	Reserved	Reserved	N	V	V
Default	0	0	0	0	0	0	0	0

Note:1. N=non-volatile, V= volatile.

DRV1 & DRV0 bit.

The DRV1 & DRV0 bits are non-volatile Read/Write bits which are used to determine the output driver strength for the Read operations.

DRV1, DRV0	Drive Strength
0,0 (default)	100%
0,1	50%
1,0	100%
1,1	100%

WPS bit

The WPS bit is a non-volatile Read/Write bit in the Configure Register which is used to select which Write Protect scheme should be used. When WPS=0(default), the device will use the combination of CMP, BP [4:0] bits to protect a specific area of the memory array. When WPS=1, the device will utilize the Individual Block Locks to protect any individual sector or blocks. The default value for all Individual Block Lock bits is 1 upon device power on or after reset.

DC bit

The Dummy Cycle (DC) bit is a volatile bit. The Dummy Cycle (DC) bit can be used to configure the number of dummy clocks for “SPI 2 X IO Read (BBH)” command, “SPI 4X I/O Read (EBH)” command.

Dummy Cycle Table

Mode	Command	DC1	Number of dummy cycles
SPI command	BB SPI	0(default)	4
		1	8
	EB SPI	0(default)	6
		1	10

DLP bit.

The DLP bit is Data Learning Pattern Enable bit, which is volatile writable by 11H command. For DTR Read commands, a pre-defined “Data Learning Pattern” can be used by the flash memory controller to determine the flash data output timing on I/O pins. When DLP=1, in dummy cycles, the flash will output “00110100” Data Learning Pattern sequence on each of the I/O pins. During this period, controller can fine tune the data latching timing for each I/O pins to achieve optimum system performance. DLP=0(default) will disable the Data Learning Pattern output.

9.7 Write Status Register (WRSR) (01H/31H)

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

The Write Status Register (WRSR) command has no effect on S15, S10, S1 and S0 of the Status Register. CS# must be driven high after the eighth or sixteen bits of the data byte has been latched in. If not, the Write Status Register (WRSR) command is not executed. If CS# is driven high after eighth bit of the data byte, the S15~S8 bits will keep the previous value. As soon as CS# is driven high, the self-timed Write Status Register cycle (whose duration is tW) 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) command allows the user to change the values of the Block Protect (BP4, BP3, BP2, BP1, and BP0) bits, to define the size of the area that is to be treated as read-only, as defined in Table1.

The sequence of issuing WRSR instruction is: CS# goes low→ sending WRSR instruction code→ Status Register data on SI→CS# goes high.

The CS# must go high exactly at the 8 bits or 16 bits data boundary; otherwise, the instruction will be rejected and not executed. The self-timed Write Status Register cycle time (tW) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Write Status Register cycle is in progress. The WIP sets 1 during the tW timing, and sets 0 when Write Status Register Cycle is completed, and the Write Enable Latch (WEL) bit is reset.

For compatibility with the old product, both 01H command followed by 2 bytes data and 31H command can allows new values to be written to the Status Register -1.

Figure 9-7 Write Status Register (WRSR) Sequence (Command 01h or 31h)

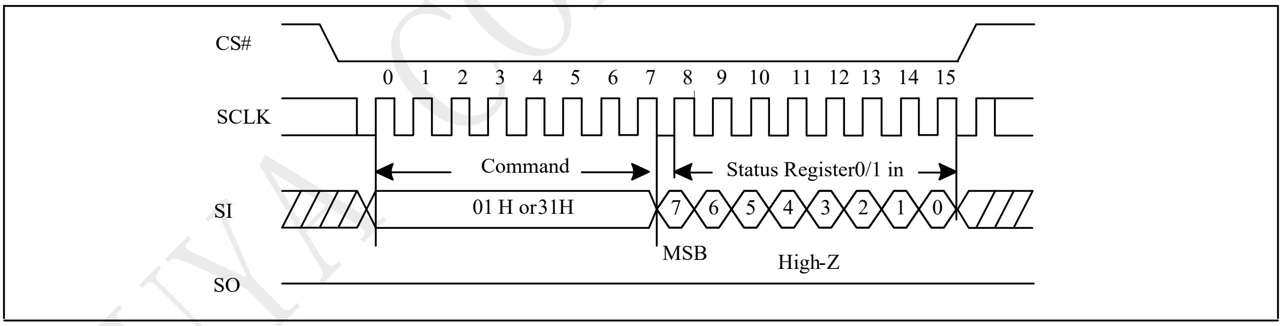


Figure 9-7a Write Status Register (WRSR) Sequence (QPI)

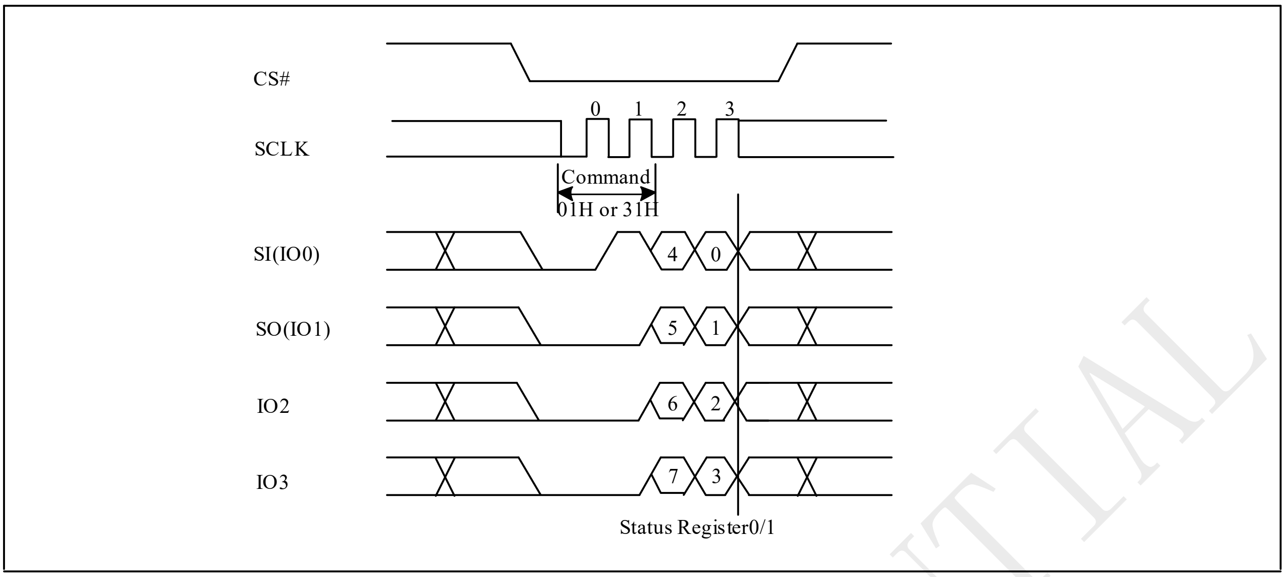


Figure 9-7b Write Status Register (WRSR) Sequence (Command 01, 2 bytes data)

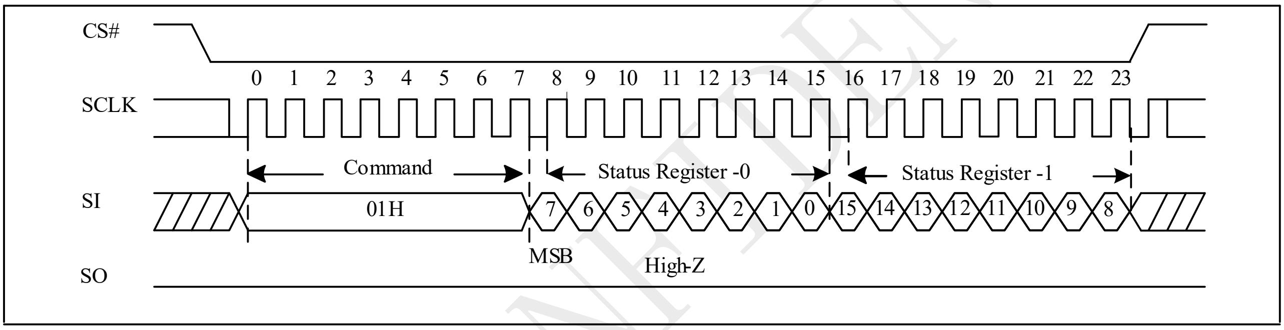
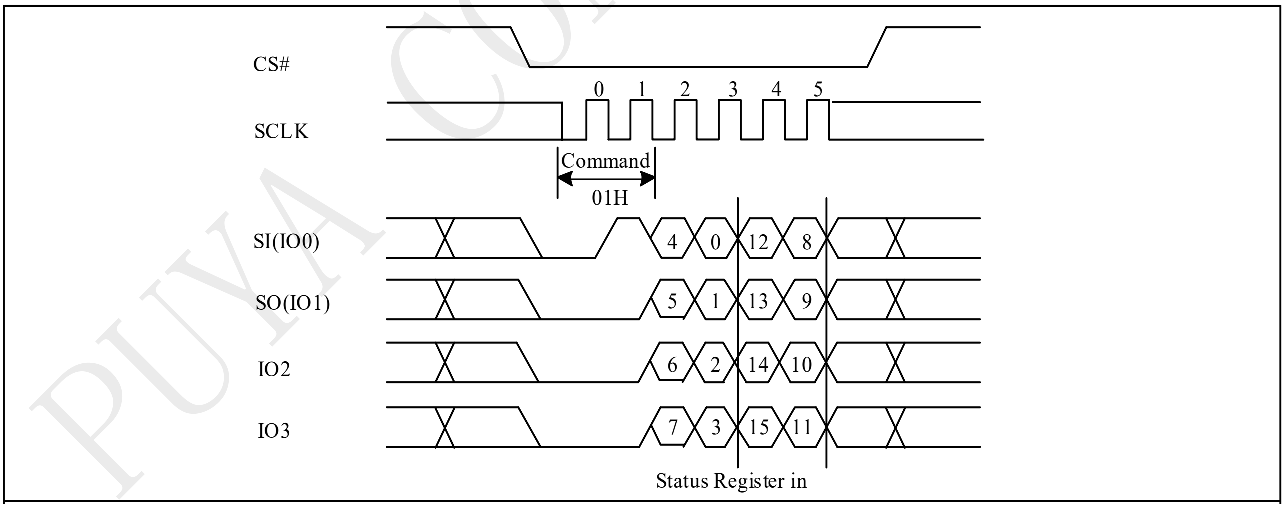


Figure 9-7c Write Status Register (WRSR) Sequence (QPI)



9.8 Write Configure Register (WRCR) (11H)

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

The sequence of issuing WRCR instruction is: CS# goes low→ sending WRCR instruction code→ Configure Register data on SI→CS# goes high.

The CS# must go high exactly at the 8 bits data boundary; otherwise, the instruction will be rejected and not executed. The self-timed Write Status Register cycle time (tW) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Write Status Register cycle is in progress. The WIP sets 1 during the tW timing, and sets 0 when Write Configure Register Cycle is completed, and the Write Enable Latch (WEL) bit is reset.

Figure 9-8 Write Configure Register (WRCR) Sequence (Command 11h)

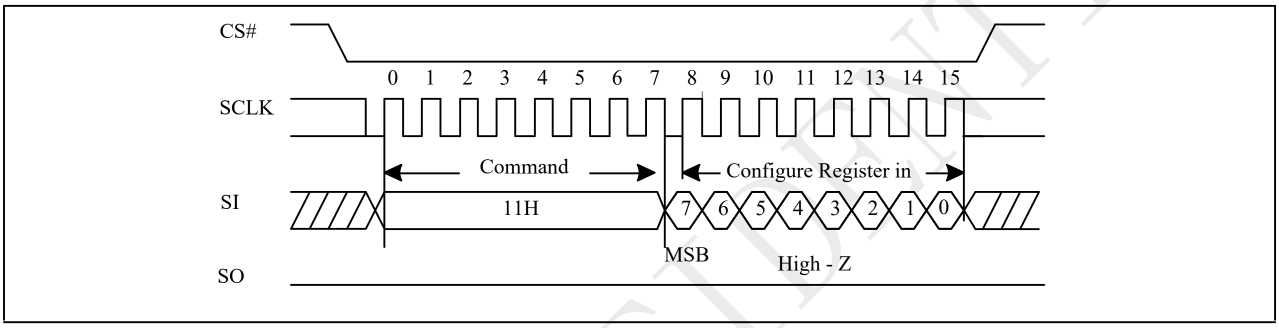
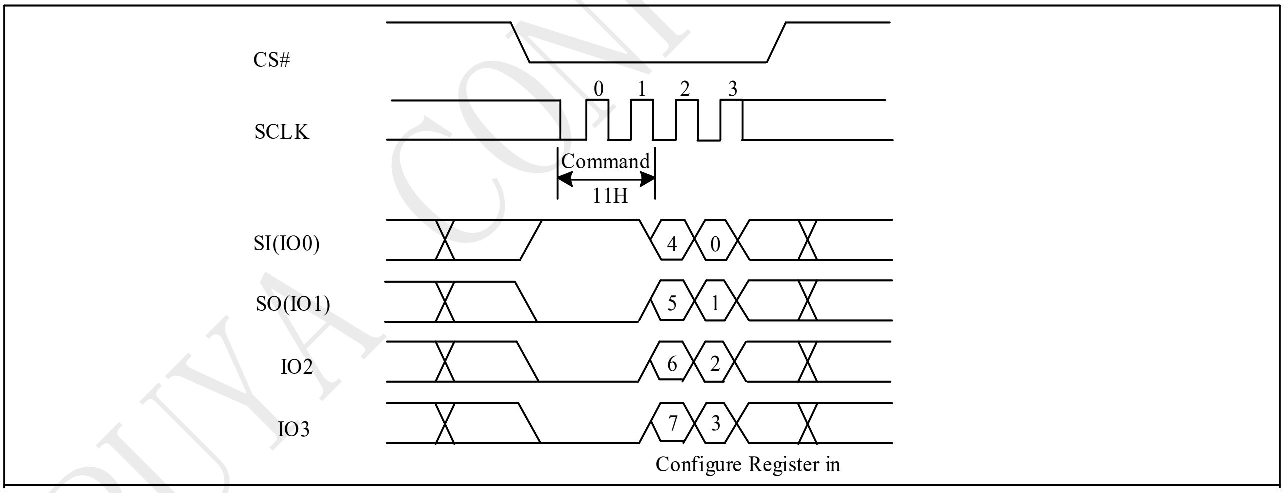


Figure 9-8a Write Configure Register (WRCR) Sequence (QPI)

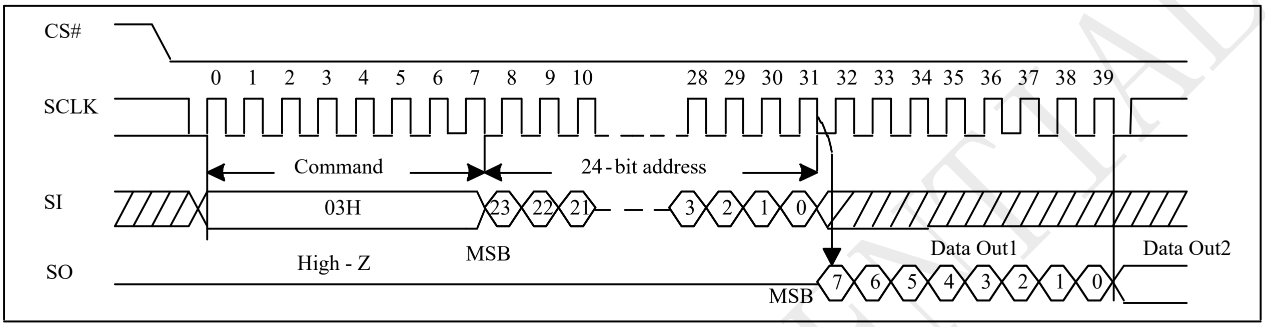


9.9 Read Data Bytes (READ) (03H)

The read instruction is for reading data out. The address is latched on rising edge of SCLK, and data shifts out on the falling edge of SCLK at a maximum frequency fR. The first address byte 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 READ instruction. The address counter rolls over to 0 when the highest address has been reached.

The sequence of issuing READ instruction is: CS# goes low→ sending READ instruction code→ 3-byte address on SI→ data out on SO→ to end READ operation can use CS# to high at any time during data out.

Figure 9-9 Read Data Bytes (READ) Sequence (Command 03h)



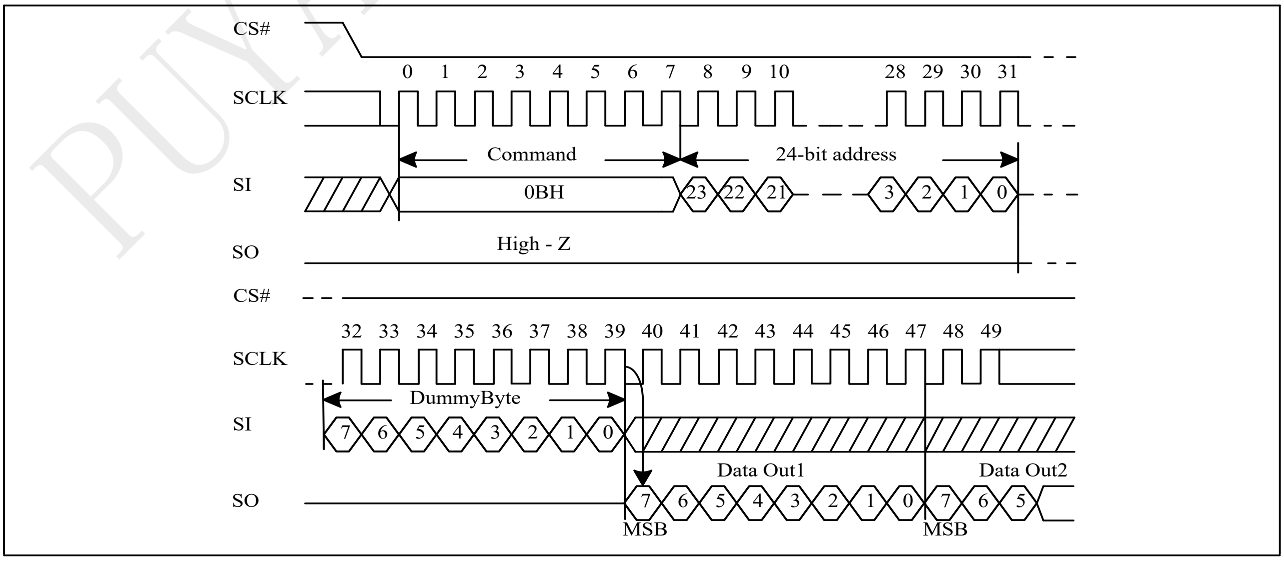
9.10 Fast Read (FREAD) (0BH)

The FAST READ instruction is for quickly reading data out. The address is latched on rising edge of SCLK, and data of each bit shifts out on the falling edge of SCLK at a maximum frequency fC. The first address byte 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 FREAD instruction. The address counter rolls over to 0 when the highest address has been reached.

The sequence of issuing FREAD instruction is: CS# goes low→ sending FREAD instruction code→3-byte address on SI→ 1-dummy byte address on SI→ data out on SO→ to end FREAD operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, FREAD instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

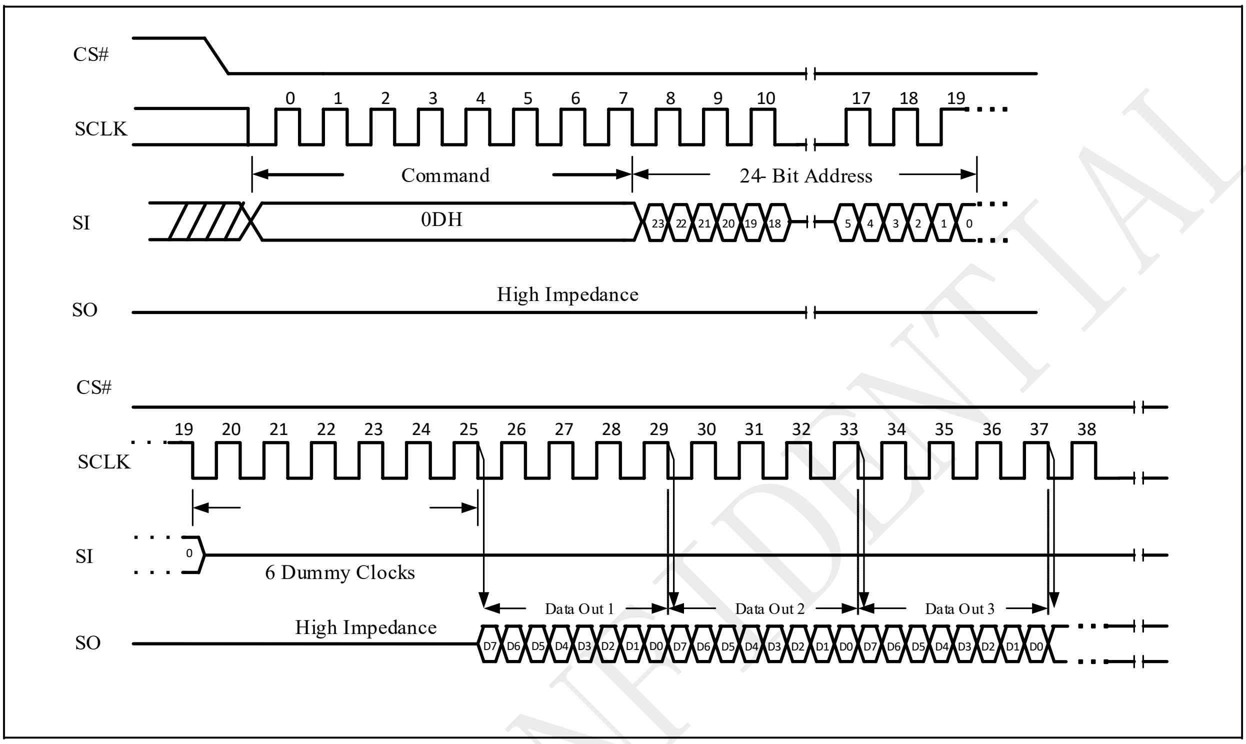
Figure 9-10 Fast Read (FREAD) Sequence (Command 0Bh)



9.11 DTR Fast Read (DTR_FREAD) (0DH)

The DTR Fast Read instruction is similar to the Fast Read instruction except that the 24-bit address input and the data output require DTR (Double Transfer Rate) operation. This is accomplished by adding six “dummy” clocks after the 24-bit address. The dummy clocks allow the devices internal circuits additional time for setting up the initial address. During the dummy clocks the data value on the DO pin is a “don’t care”.

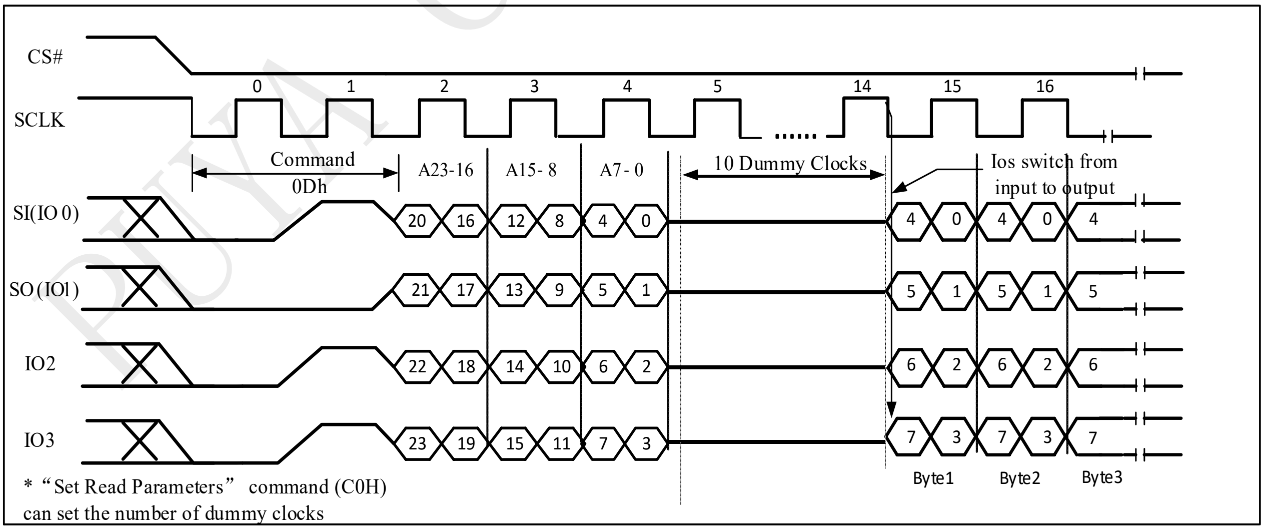
Figure 9-11 DTR Fast Read Sequence (Command 0Dh)



DTR Fast Read in QPI Mode

The DTR Fast Read instruction is also supported in QPI mode.

Figure 9-11a DTR Fast Read Sequence (QPI)



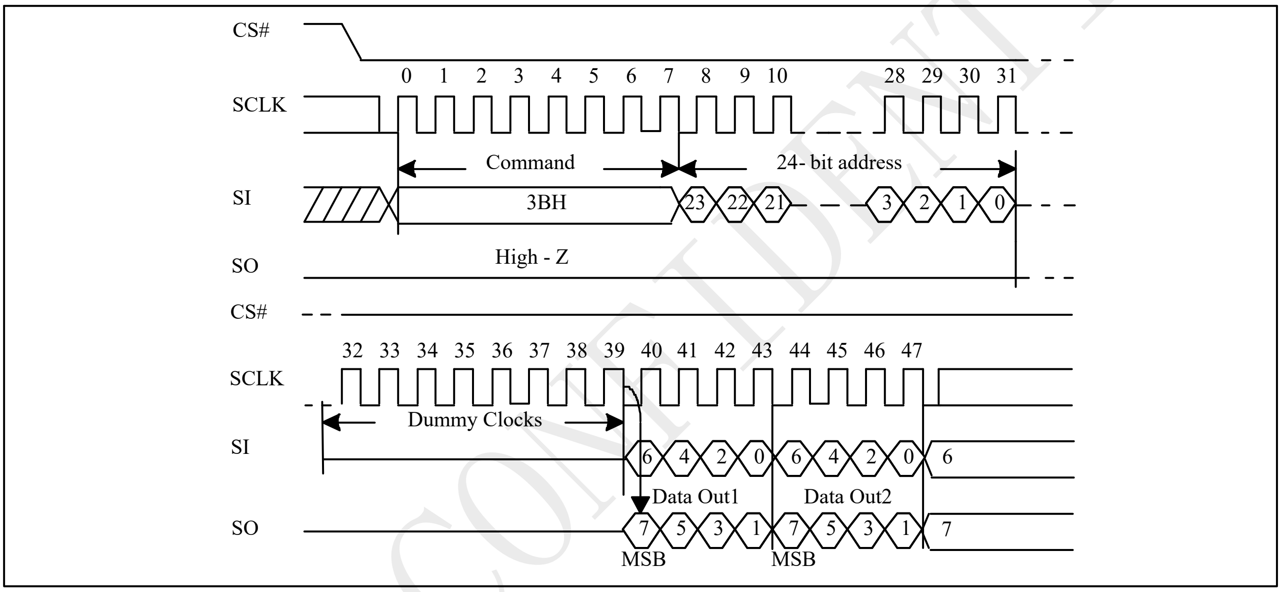
9.12 Dual Read (DREAD) (3BH)

The DREAD instruction enables double throughput of Serial NOR Flash in read mode. The address is latched on rising edge of SCLK, and data of every two bits (interleave on 2 I/O pins) shift out on the falling edge of SCLK at a maximum frequency fT. The first address byte 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 DREAD instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing DREAD instruction, the following data out will perform as 2-bit instead of previous 1-bit.

The sequence of issuing DREAD instruction is: CS# goes low → sending DREAD instruction → 3-byte address on SI → 8-bit dummy cycle → data out interleave on SIO1 & SIO0 → to end DREAD operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, DREAD instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

Figure 9-12 Dual Read Mode Sequence (Command 3Bh)



9.13 2IO Read (2READ) (BBH)

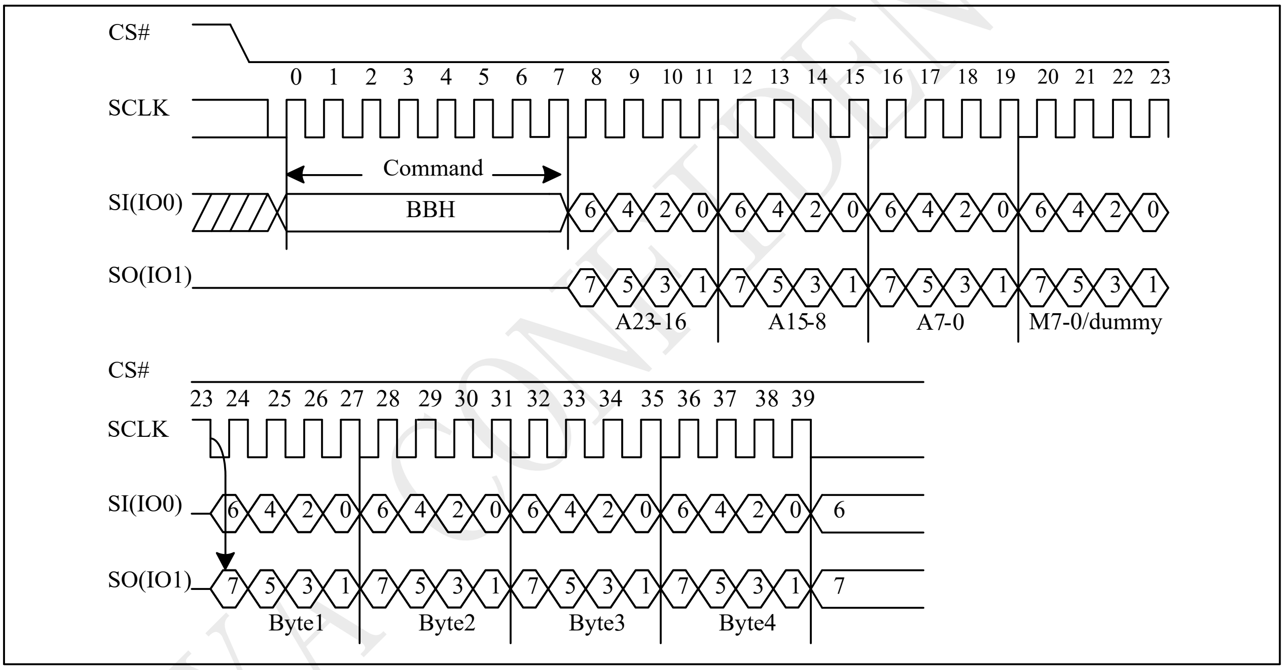
The 2READ instruction enables Double Transfer Rate of Serial NOR Flash in read mode. The address is latched on rising edge of SCLK, and data of every two bits (interleave on 2 I/O pins) shift out on the falling edge of SCLK at a maximum frequency fT. The first address byte can be at any location. The 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 2READ instruction. The address counter rolls over to 0 when the highest address has been reached.

Once writing 2READ instruction, the following address/dummy/data out will perform as 2-bit instead of previous 1-bit.

The sequence of issuing 2READ instruction is: CS# goes low→ sending 2READ instruction→ 24-bit address interleave on SIO1 & SIO0→ 8-bit dummy cycle on SIO1 & SIO0→ data out interleave on SIO1 & SIO0→ to end 2READ operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, 2READ instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

Figure 9-13 2IO Read Sequence (Command BBh M5-4 ≠ (1,0))



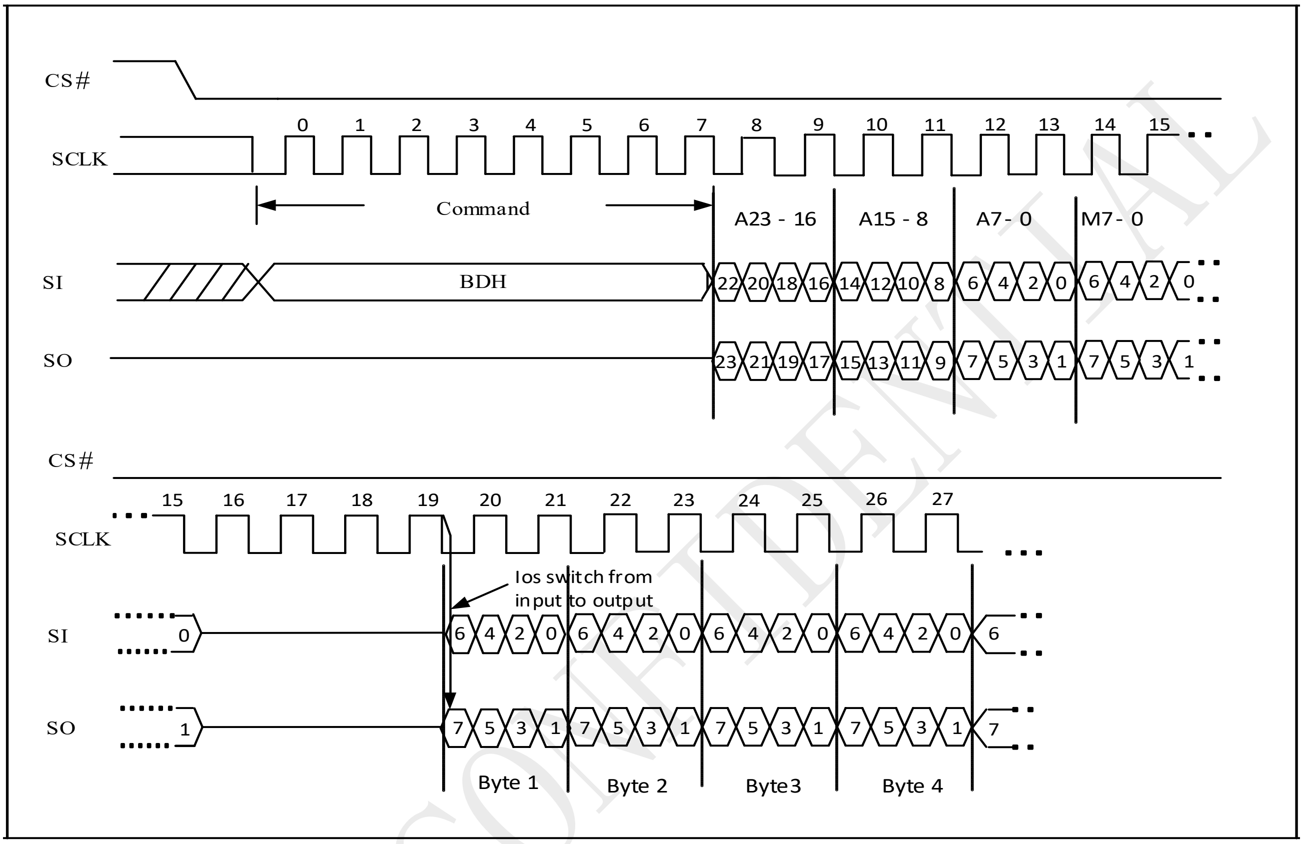
Note:

1. M [5-4] = (1,0) is inhibited.
2. DC bit can set the number of dummy clocks.

9.14 DTR 2IO Read (DTR_2READ) (BDH)

The DTR 2IO Read (BDh) instruction allows for improved random access while maintaining two IO pins, IO0 and IO1. It is similar to the DREAD (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.

Figure 9-14 DTR 2IO Read Sequence (Command BDh M5-4 ≠ (1,0))



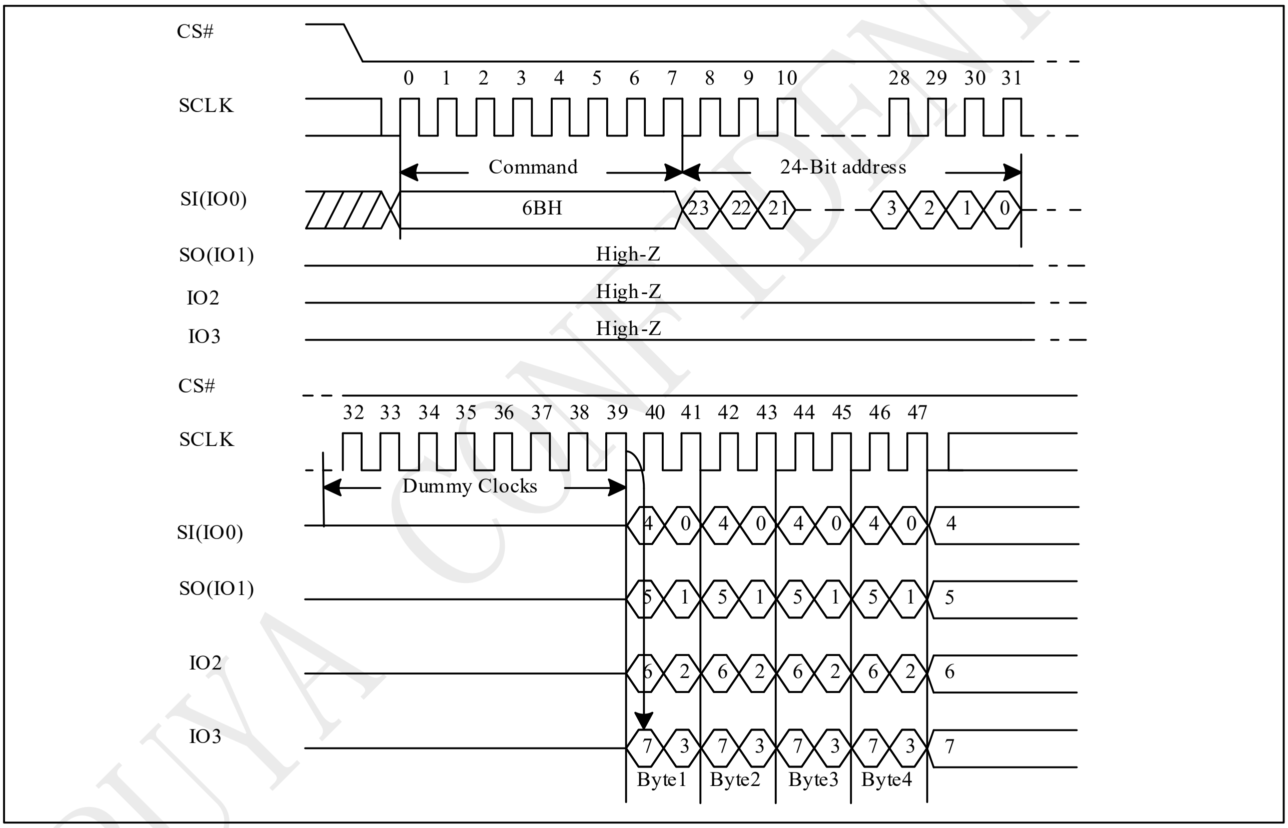
9.15 Quad Read (QREAD) (6BH)

The QREAD instruction enable quad throughput of Serial NOR Flash in read mode. A Quad Enable (QE) bit of status Register must be set to "1" before sending the QREAD instruction. The address is latched on rising edge of SCLK, and data of every four bits (interleave on 4 I/O pins) shift out on the falling edge of SCLK at a maximum frequency fQ. The first address byte 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 QREAD instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing QREAD instruction, the following data out will perform as 4-bit instead of previous 1-bit.

The sequence of issuing QREAD instruction is: CS# goes low→ sending QREAD instruction → 3-byte address on SI → 8-bit dummy cycle → data out interleave on SIO3, SIO2, SIO1 & SIO0→ to end QREAD operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, QREAD instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

Figure 9-15 Quad Read Sequence (Command 6Bh)



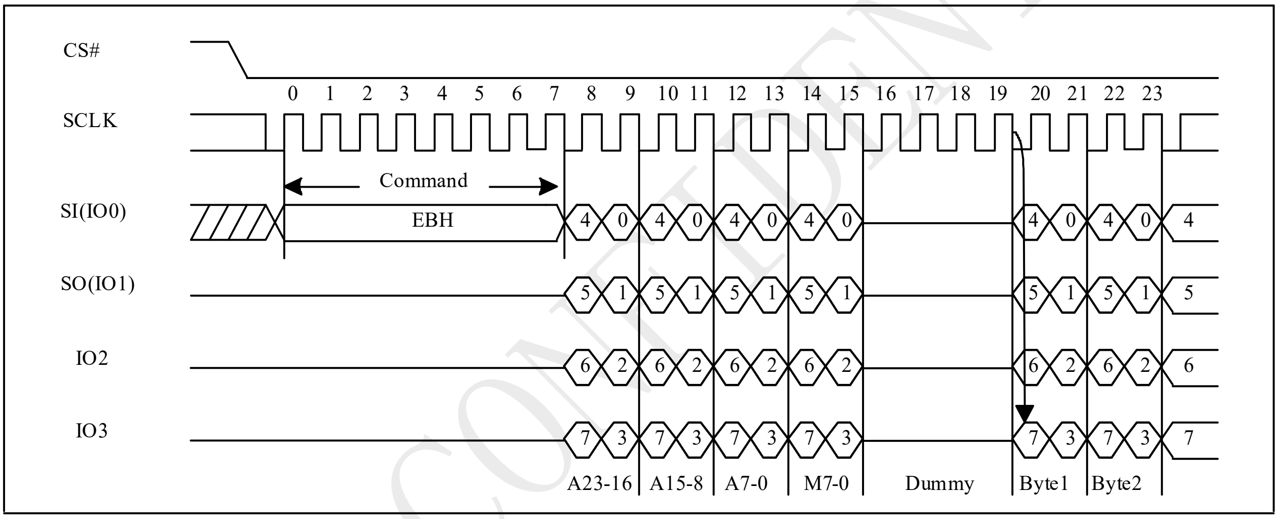
9.16 4IO Read (4READ) (EBH)

The 4READ instruction enable quad throughput of Serial NOR Flash in read mode. A Quad Enable (QE) bit of status Register must be set to "1" before sending the 4READ instruction. The address is latched on rising edge of SCLK, and data of every four bits (interleave on 4 I/O pins) shift out on the falling edge of SCLK at a maximum frequency fQ. The first address byte 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 4READ instruction. The address counter rolls over to 0 when the highest address has been reached. Once writing 4READ instruction, the following address/dummy/data out will perform as 4-bit instead of previous 1-bit.

The sequence of issuing 4READ instruction is: CS# goes low→ sending 4READ instruction→ 24-bit address interleave on SIO3, SIO2, SIO1 & SIO0→2+4 dummy cycles→data out interleave on SIO3, SIO2, SIO1 & SIO0→ to end 4READ operation can use CS# to high at any time during data out.

While Program/Erase/Write Status Register cycle is in progress, 4READ instruction is rejected without any impact on the Program/Erase/Write Status Register current cycle.

Figure 9-16 4IO Read Sequence (Command EBh M5-4 ≠ (1,0))



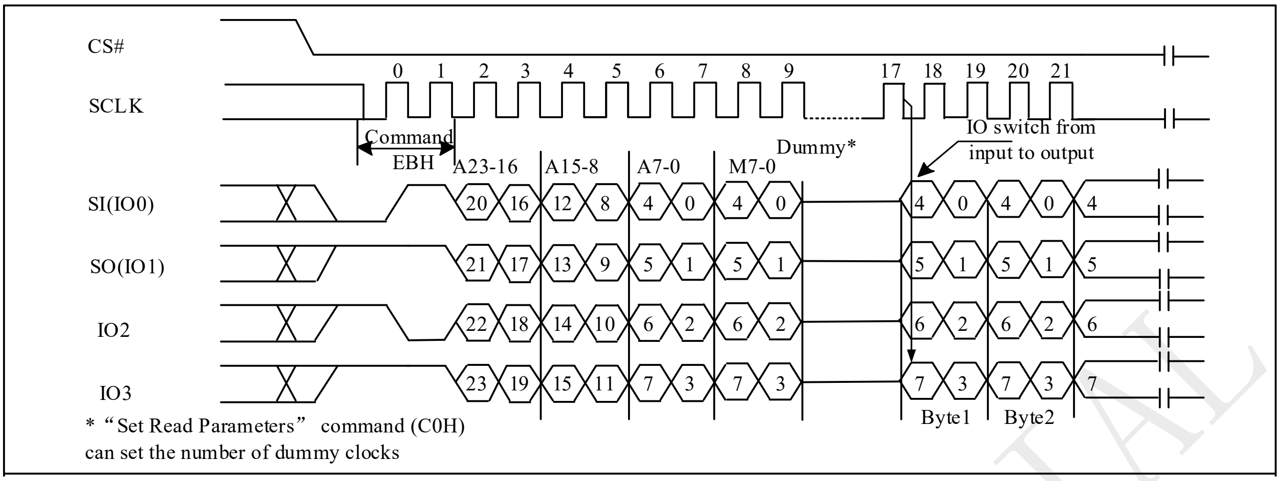
Note:
1. Hi-impedance is inhibited for the two clock cycles.
2. M [5-4] = (1,0) is inhibited.
3. DC bit can set the number of dummy clocks.

4IO Read in QPI mode

The 4READ instruction is also supported in QPI mode. When QPI mode is enabled, the number of dummy clocks is configured by the “Set Read Parameters (C0h)” instruction to accommodate a wide range of applications with different needs for either maximum Fast Read frequency or minimum data access latency. Depending on the Read Parameter Bits P [5:4] setting, the number of dummy clocks can be configured. In QPI mode, the bits M7- 0 is also considered as dummy clocks.

“Wrap Around” feature is not available in QPI mode for 4IO Read instruction. To perform a read operation with fixed data length wrap around in QPI mode, a dedicated “Burst Read with Wrap” (0Ch) instruction must be used.

Figure 9-16a 4IO Read in QPI mode Sequence (QPI M5-4 ≠ (1,0))



Note:

1. Hi-impedance is inhibited for the two clock cycles.
2. M [5-4] = (1,0) is inhibited.

9.17 Set Burst Read(77H)

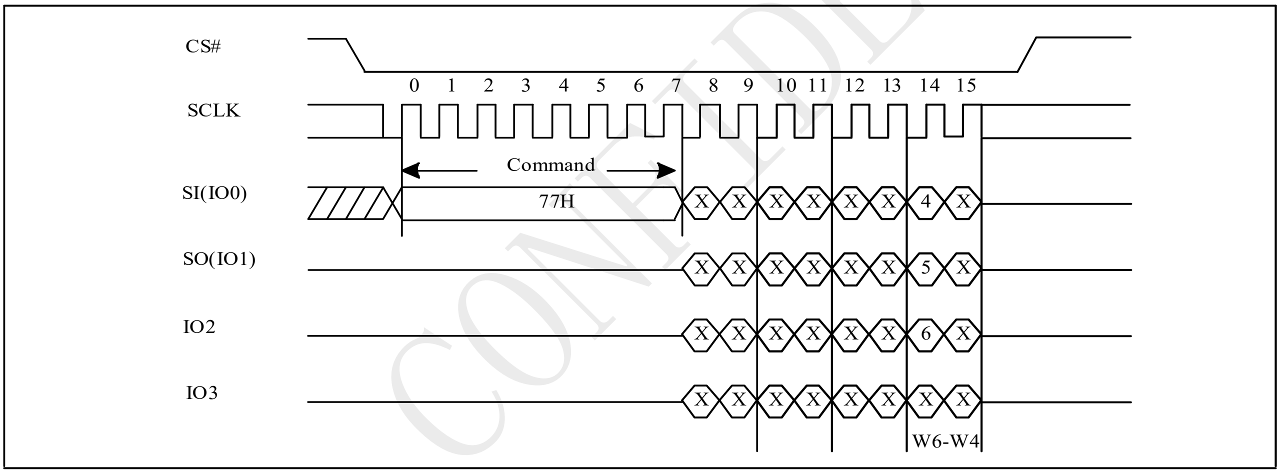
The Set Burst with Wrap command is used in conjunction with “4IO Read (EBH)”, “DTR 4IO Read (EDH)”, “4IO Word Read (E7H)” commands to access a fixed length of 8/16/32/64-byte section within a 256-byte page, in standard SPI mode.

The Set Burst with Wrap command sequence: CS# goes low → Send Set Burst with Wrap command → Send 24 dummy bits→ Send 8 bits “Wrap bits” → CS# goes high.

W6, W5	W4=0		W4=1 (default)	
	Wrap Around	Wrap Length	Wrap Around	Wrap Length
0,0	Yes	8-byte	No	N/A
0,1	Yes	16-byte	No	N/A
1,0	Yes	32-byte	No	N/A
1,1	Yes	64-byte	No	N/A

If the W6-W4 bits are set by the Set Burst with Wrap command, all the following “4IO Read (EBH)”, “DTR 4IO Read (EDH)”, “4IO Word Read (E7H)” commands will use the W6-W4 setting to access the 8/16/32/64-byte section within any page. To exit the “Wrap Around” function and return to normal read operation, another Set Burst with Wrap command should be issued to set W4=1.

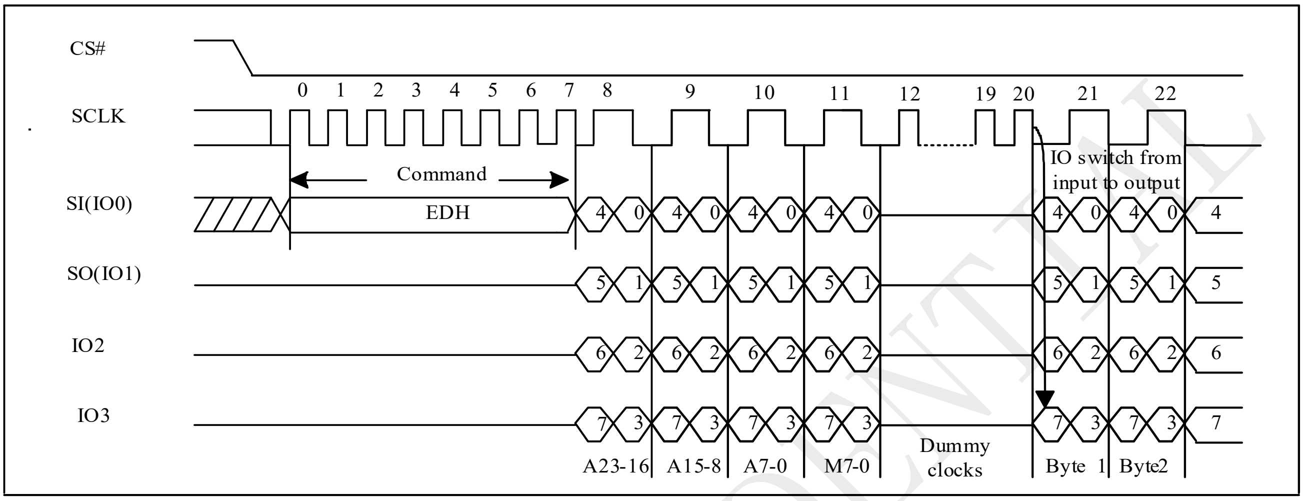
Figure 9-17 Set Burst Read (SBL) Sequence (Command 77h)



9.18 DTR 4IO Read (DTR_4READ) (EDH)

The DTR 4IO Read (EDh) instruction is similar to the DTR 2IO Read (BDh) instruction, except that address and data bits are input and output through four pins IO0, IO1, IO2 and IO3 and1 byte-data(M7-0) and 9 Dummy clocks are required in SPI mode 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.

Figure 9-18 DTR 4IO Read Mode Sequence (Command EDh M5-4 ≠ (1,0))



Note:
1. Hi-impedance is inhibited for the mode clock cycles.
2. M [5-4] = (1,0) is inhibited.

DTR 4IO Read with “8/16/32/64-Byte Wrap Around” in Standard SPI mode

The DTR 4IO Read instruction can also be used to access a specific portion within a page by issuing a “Set Burst Read” (77h) command prior to EDh. The “Set Burst Read” (77h) command can either enable or disable the “Wrap Around” feature for the following EDh commands. When “Wrap Around” is enabled, the data being accessed can be limited to either an 8, 16, 32 or 64-byte section of a 256-byte page. The output data starts at the initial address specified in the instruction, once it reaches the ending boundary of the 8/16/32/64-byte section, the output will wrap around to the beginning boundary automatically until /CS is pulled high to terminate the command.

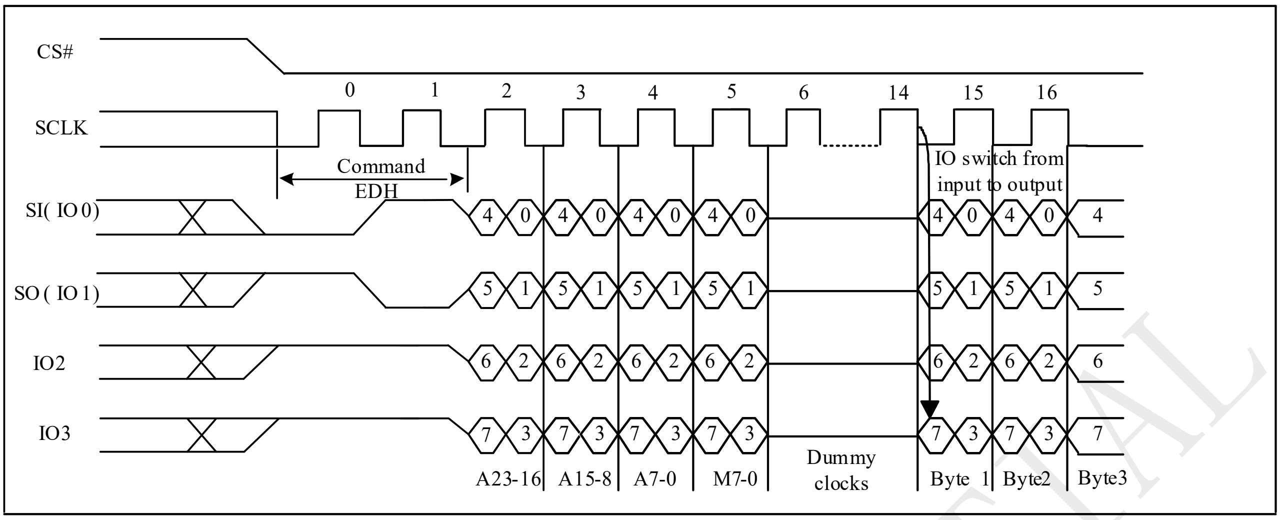
The Burst with Wrap feature allows applications that use cache to quickly fetch a critical address and then fill the cache afterwards within a fixed length (8/16/32/64-byte) of data without issuing multiple read commands. The “Set Burst Read” instruction allows three “Wrap Bits”, W6-4 to be set. The W4 bit is used to enable or disable the “Wrap Around” operation while W6-5 is used to specify the length of the wrap around section within a page.

DTR 4IO Read (EDh) in QPI Mode

The DTR 4IO Read instruction is also supported in QPI mode. In QPI mode, the bits M7-0 is also considered as dummy clocks.

“Wrap Around” feature is not available in QPI mode for DTR 4IO Read instruction. To perform a read operation with fixed data length wrap around in QPI mode, a dedicated “Burst Read with Wrap” (0Ch) instruction must be used.

Figure 9-18b DTR 4IO Read Mode Sequence (QPI EDh M5-4 ≠ (1,0))



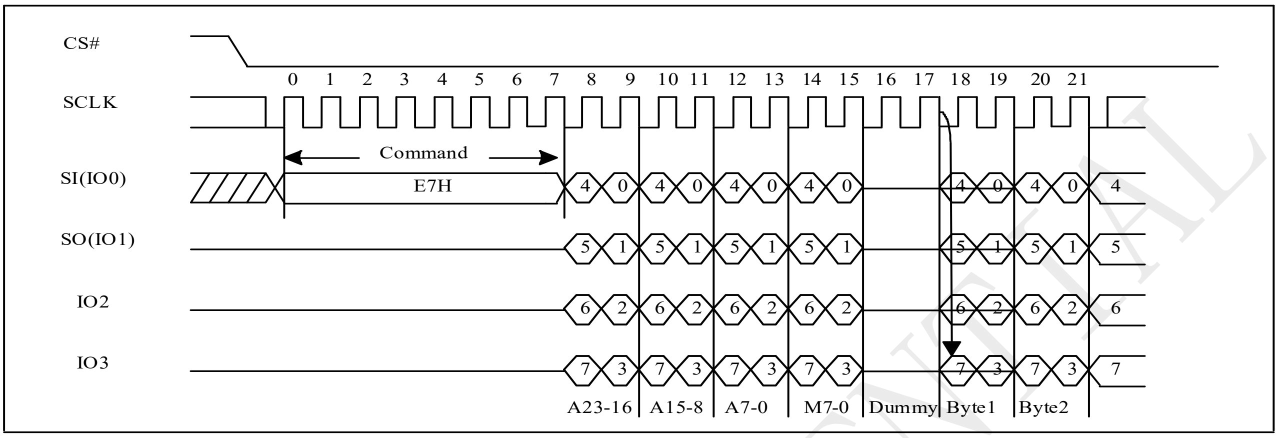
Note:

1. Hi-impedance is inhibited for the two clock cycles.
2. M [5-4] = (1,0) is inhibited.

9.19 4IO Word Read(E7H)

The 4IO Word Read command is similar to the 4 IO Read command except that the lowest address bit (A0) must equal 0 and only 2-dummy clock. 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 Quad Enable bit (QE) of Status Register (S9) must be set to enable for the 4IO Word read command.

Figure 9-19 4IO Word Read Sequence (M5-4 ≠ (1,0))



4IO Word Read with “8/16/32/64-Byte Wrap Around “in Standard SPI mode

The 4IO Word Read command can be used to access a specific portion within a page by issuing “Set Burst with Wrap” (77H) commands prior to E7H. The “Set Burst with Wrap” (77H) command can either enable or disable the “Wrap Around “feature for the following E7H commands. When “Wrap Around “is enabled, the data being accessed can be limited to either an8/16/32/64-byte section of a 256-byte page. The output data starts at the initial address specified in the command, once it reaches the ending boundary of the 8/16/32/64-byte section, the output will wrap around the beginning boundary automatically until CS# is pulled high to terminate the command.

The Burst with Wrap feature allows applications that use cache to quickly fetch a critical address and then fill the cache afterwards within a fixed length (8/16/32/64-byte) of data without issuing multiple read commands. The “Set Burst with Wrap “command allows three “Wrap Bits” W6-W4 to be set. The W4 bit is used to enable or disable the “Wrap Around “operation while W6-W5 issued to specify the length of the wrap around section within a page.

9.20 Set Read Parameters(C0H)

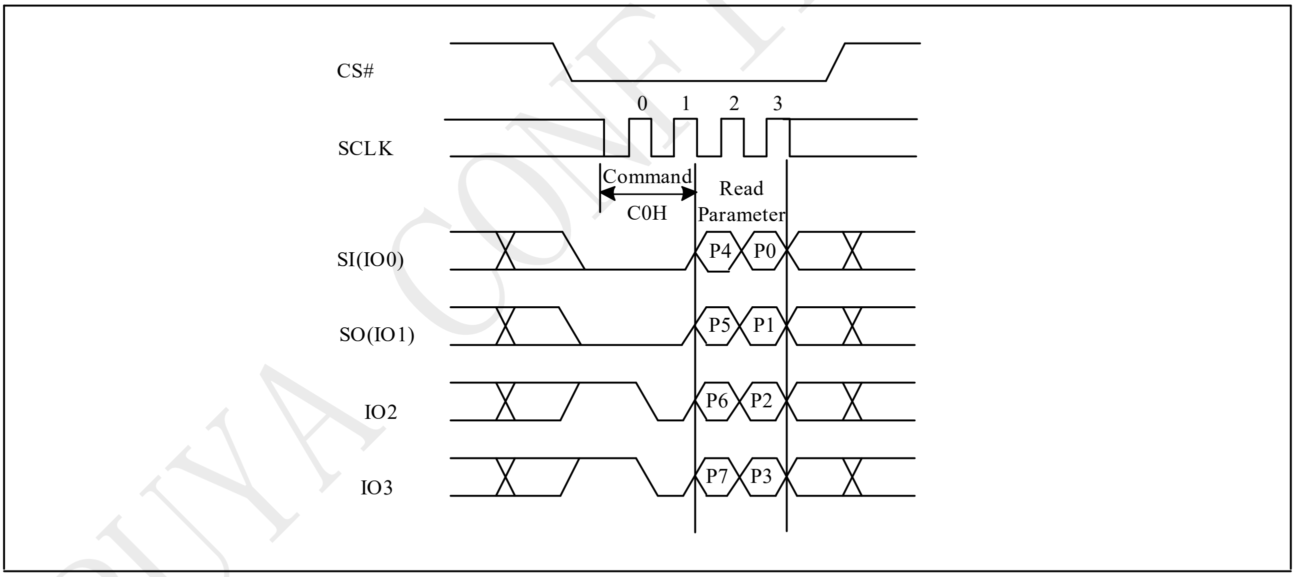
In QPI mode, to accommodate a wide range of applications with different needs for either maximum read frequency or minimum data access latency, “Set Read Parameters (C0h)” instruction can be used to configure the number of dummy clocks for “4IO Read (EBh)” , “Burst Read with Wrap (0Ch)” ,“Read SFDP Mode (5Ah)”instructions, and to configure the number of bytes of “Wrap Length” for the “Burst Read with Wrap (0Ch)”, “DTR Burst Read with Wrap (0EH)” instruction.

In Standard SPI mode, the “Set Read Parameters (C0h)” instruction is not accepted. The dummy clocks for various Fast Read instructions in Standard/Dual/Quad SPI mode are configured by DC bit. The “Wrap Length” is set by W5-4 bit in the “Set Burst with Wrap (77h)” instruction. This setting will remain unchanged when the device is switched from Standard SPI mode to QPI mode.

The default “Wrap Length” after a power up or a Reset instruction and the default number of dummy clocks can refer to the following table. The number of dummy clocks is only programmable for “4IO Read (EBh)” & “Burst Read with Wrap (0Ch)” instructions in the QPI mode. Whenever the device is switched from SPI mode to QPI mode, the number of dummy clocks should be set again, prior to any EBh or 0Ch instructions.

P5-P4	Dummy Clocks	P1-P0	Wrap Length
0,0	10	0,0	8-byte
0,1	4	0,1	16-byte
1,0	6	1,0	32-byte
1,1	8	1,1	64-byte

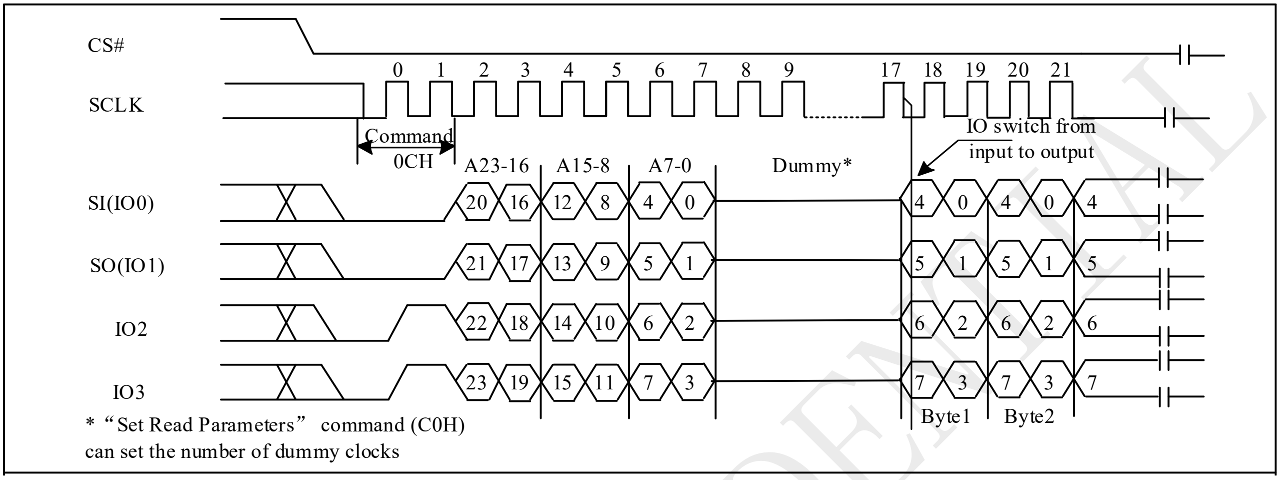
Figure 9-20 Set Read Parameters Sequence (QPI)



9.21 Burst Read with Wrap (0CH)

The “Burst Read with Wrap (0CH)” command provides an alternative way to perform the read operation with “Wrap Around” in QPI mode. This command is similar to the “Fast Read (0BH)” command in QPI mode, except the addressing of the read operation will “Wrap Around” to the beginning boundary of the “Wrap Around” once the ending boundary is reached. The “Wrap Length” and the number of dummy clocks can be configured by the “Set Read Parameters (C0H)” command.

Figure 9-21 Burst Read with Wrap Sequence (QPI)

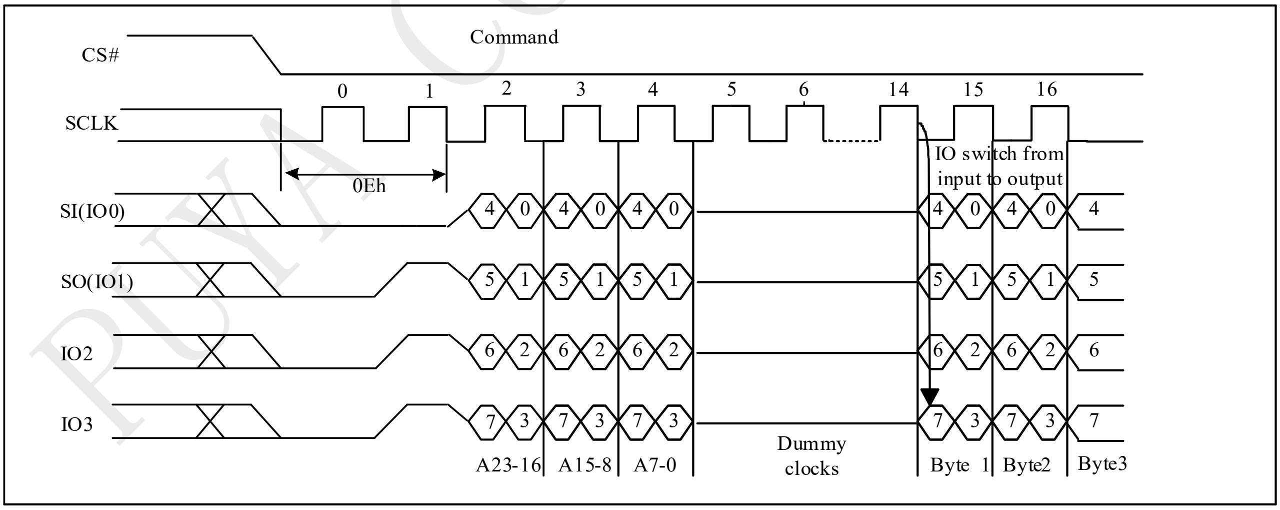


* “Set Read Parameters” command (C0H) can set the number of dummy clocks

9.22 DTR Burst Read with Wrap (0EH)

The “DTR Burst Read with Wrap (0EH)” command provides an alternative way to perform the read operation with “Wrap Around” in QPI mode. This command is similar to the “Fast Read (0BH)” command in QPI mode, except the addressing of the read operation will “Wrap Around” to the beginning boundary of the “Wrap Around” once the ending boundary is reached. The “Wrap Length” can be configured by the “Set Read Parameters (C0H)” command.

Figure 9-22 Burst Read with Wrap Sequence (QPI)



9.23 Data Learning Pattern

The data learning pattern supports system/memory controller determine valid window of data output more easily and improve data capture reliability while the flash memory is running in high frequency.

Data learning pattern can be enabled or disabled by setting the bit0 of Configure Register (data learning pattern enable bit). Once the DLP bit is set, the data learning pattern is inputted into dummy cycles.

The data learning pattern is a fixed 8-bit data pattern (00110100). For STR (single transfer rate) 1 x I/O and Dual Read (3BH) instructions, the complete 8 bits will start to output right after the last address bit. For DTR (double transfer rate) 1 x I/O instructions, the complete 8 bits will start to output right after 2 dummy cycles. For STR/DTR (double transfer rate)2x I/O instructions, the complete 8 bits will start to output right after 4 dummy/M7-0 cycles. For STR/DTR (double transfer rate)4x I/O instructions, the complete 8 bits will start to output right after 2 dummy/M7-0 cycles. While dummy cycle is not sufficient of 8 cycles, the rest of the DLP bits will be cut.

Figure 9-23 Fast Read with DLP bits output Sequence

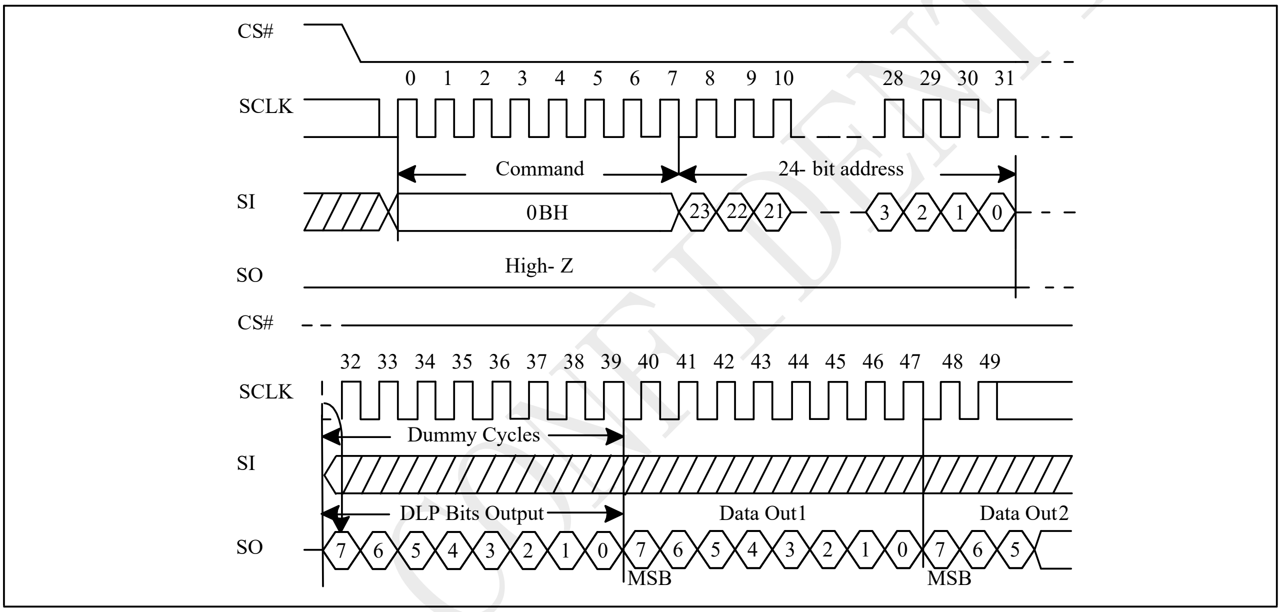


Figure 9-23a Fast Read with DLP bits output Sequence in QPI mode

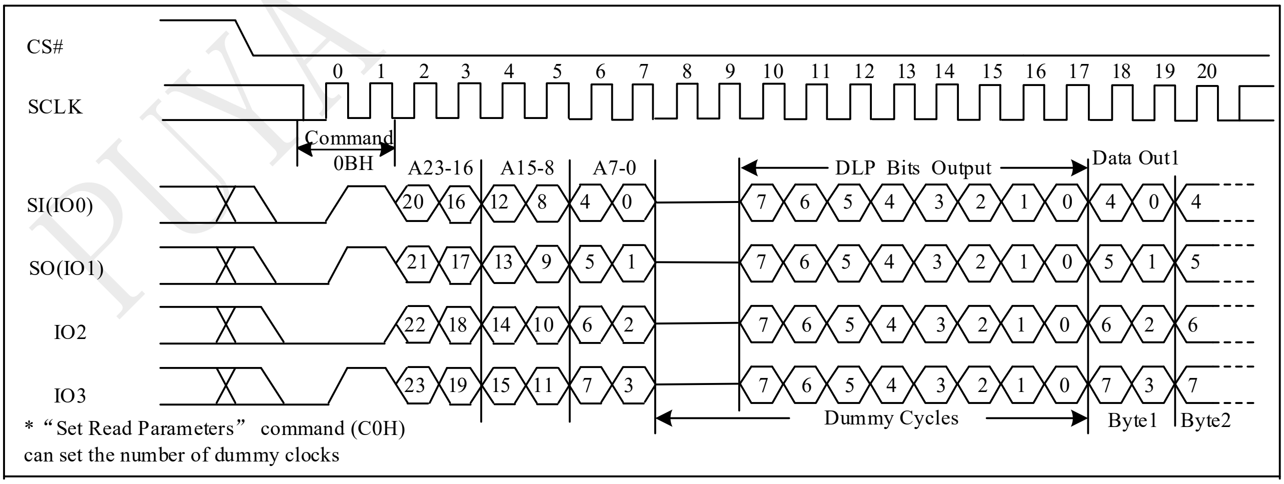


Figure 9-23b Dual Read with DLP bits output Sequence

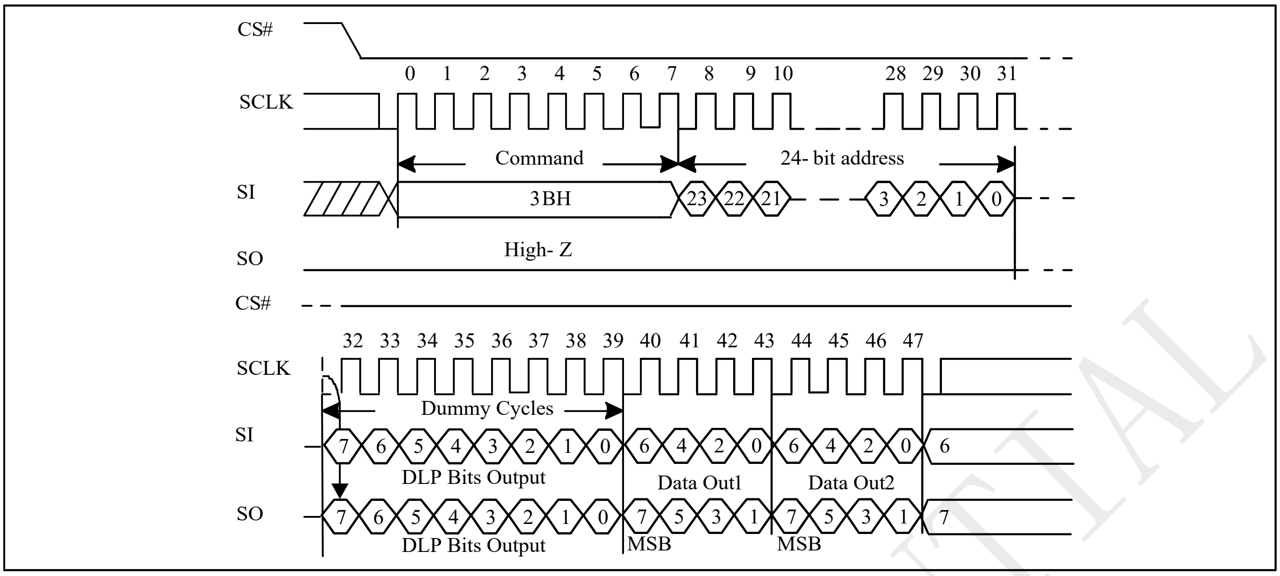


Figure 9-23c 2IO Read (choose 8 dummy) with DLP bits output Sequence

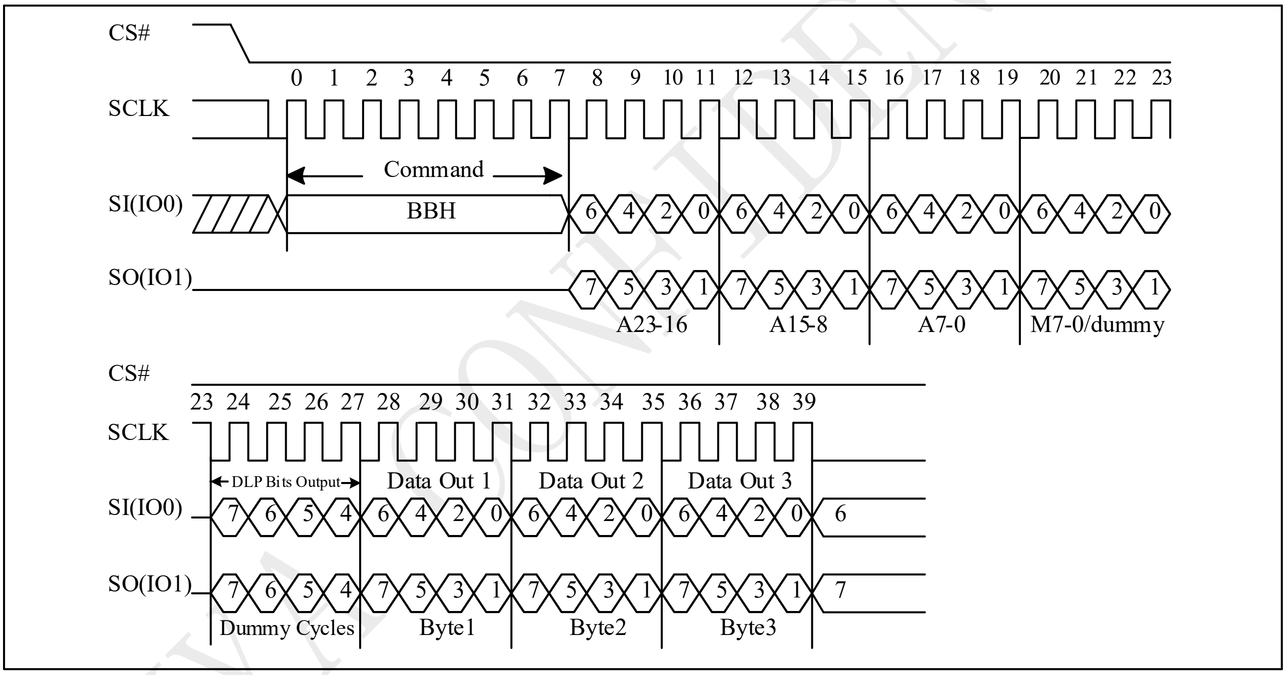


Figure 9-23d Quad Read with DLP bits output Sequence

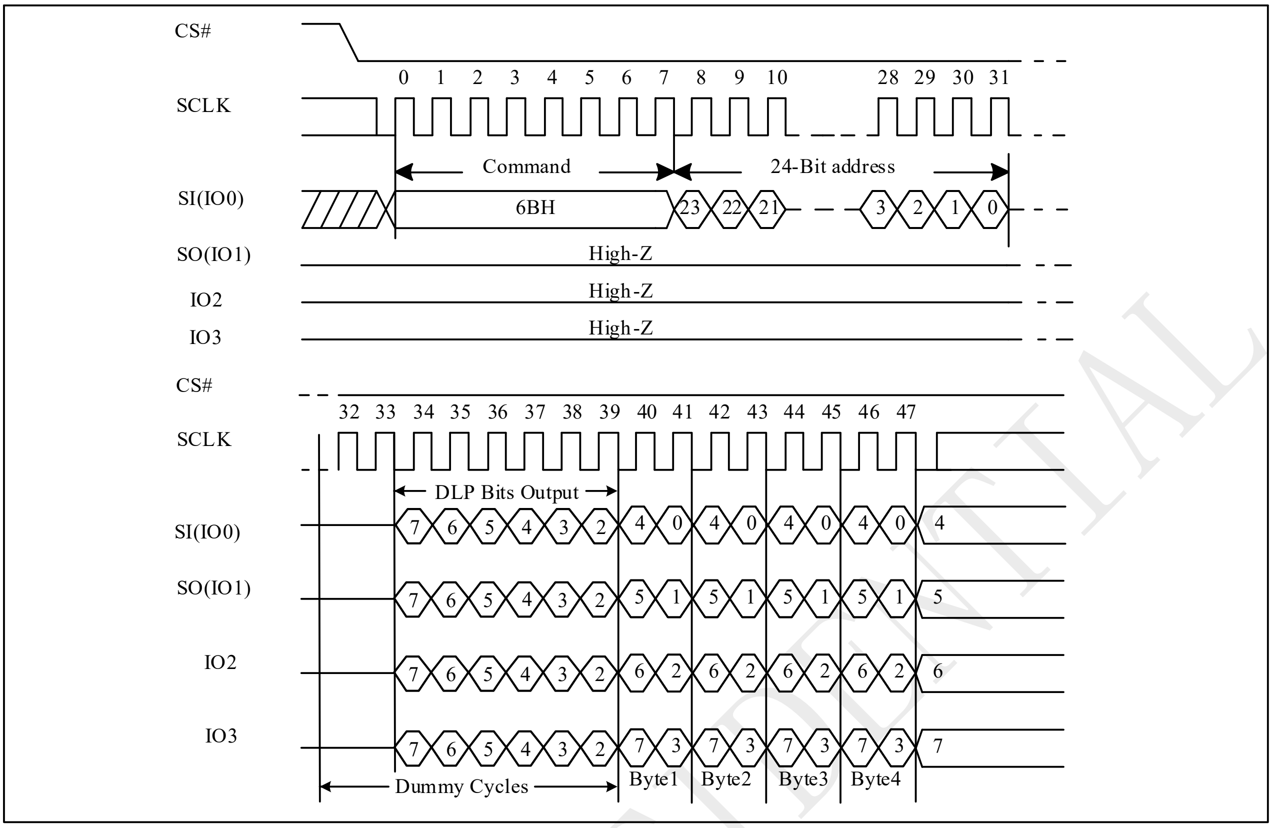


Figure 9-23e 4IO Read with DLP bits output Sequence

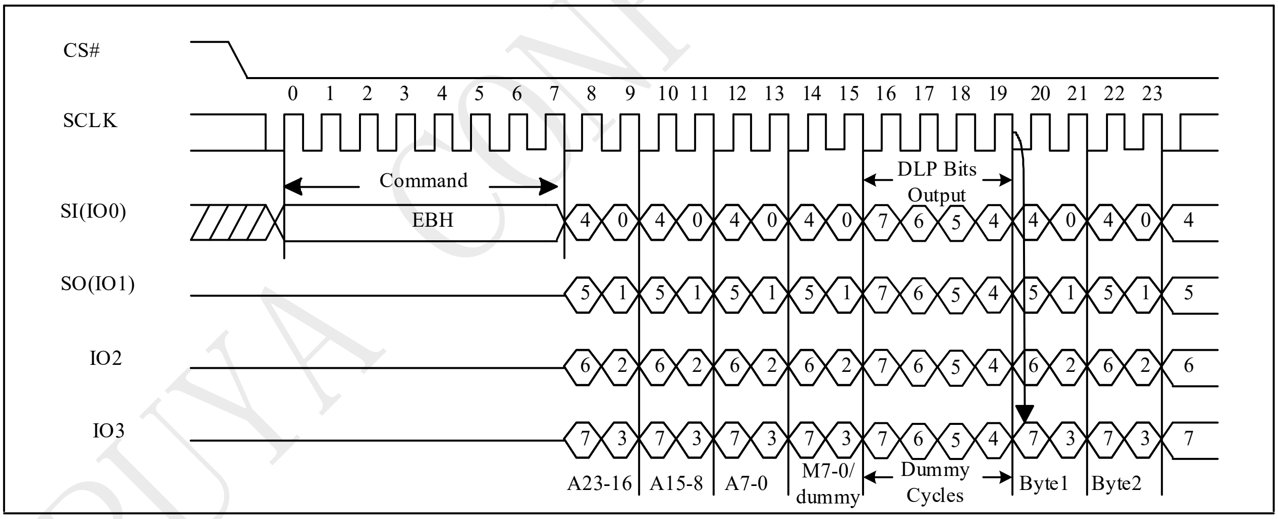
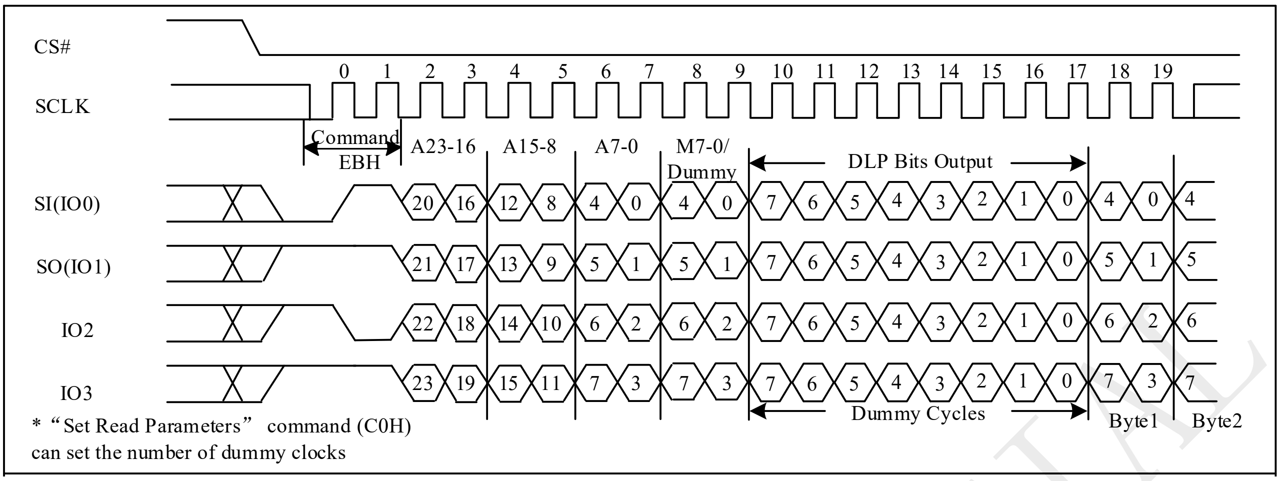


Figure 9-23f 4IO Read with DLP bits output Sequence in QPI mode



* “Set Read Parameters” command (C0H) can set the number of dummy clocks

Figure 9-23g 4IO Word Read with DLP bits output Sequence

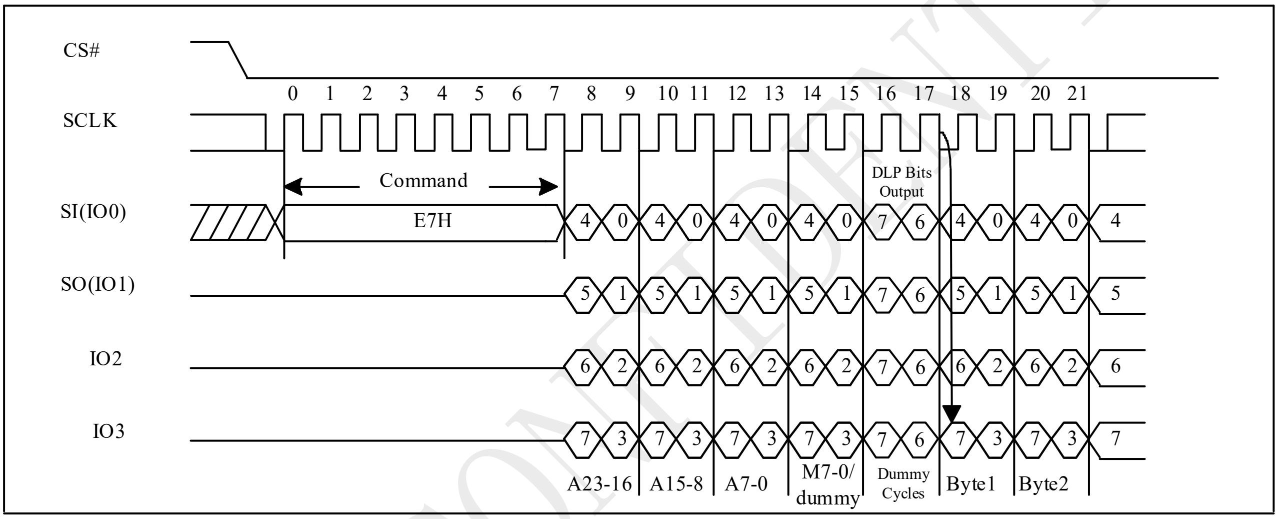
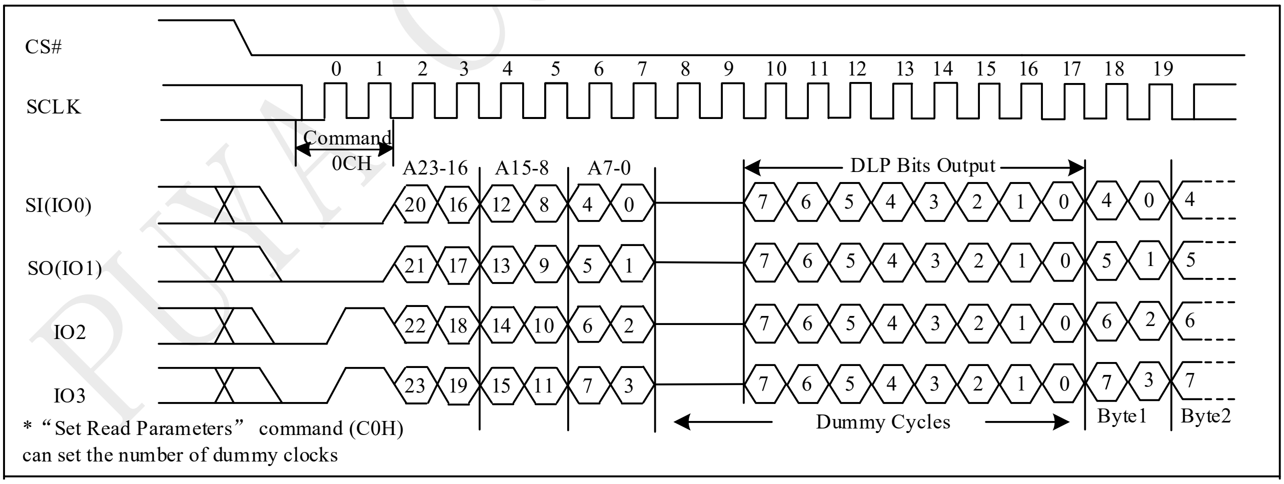


Figure 9-23h Burst Read with Wrap with DLP bits output Sequence



* “Set Read Parameters” command (C0H) can set the number of dummy clocks

Figure 9-23i DTR Fast Read with DLP bits output Sequence

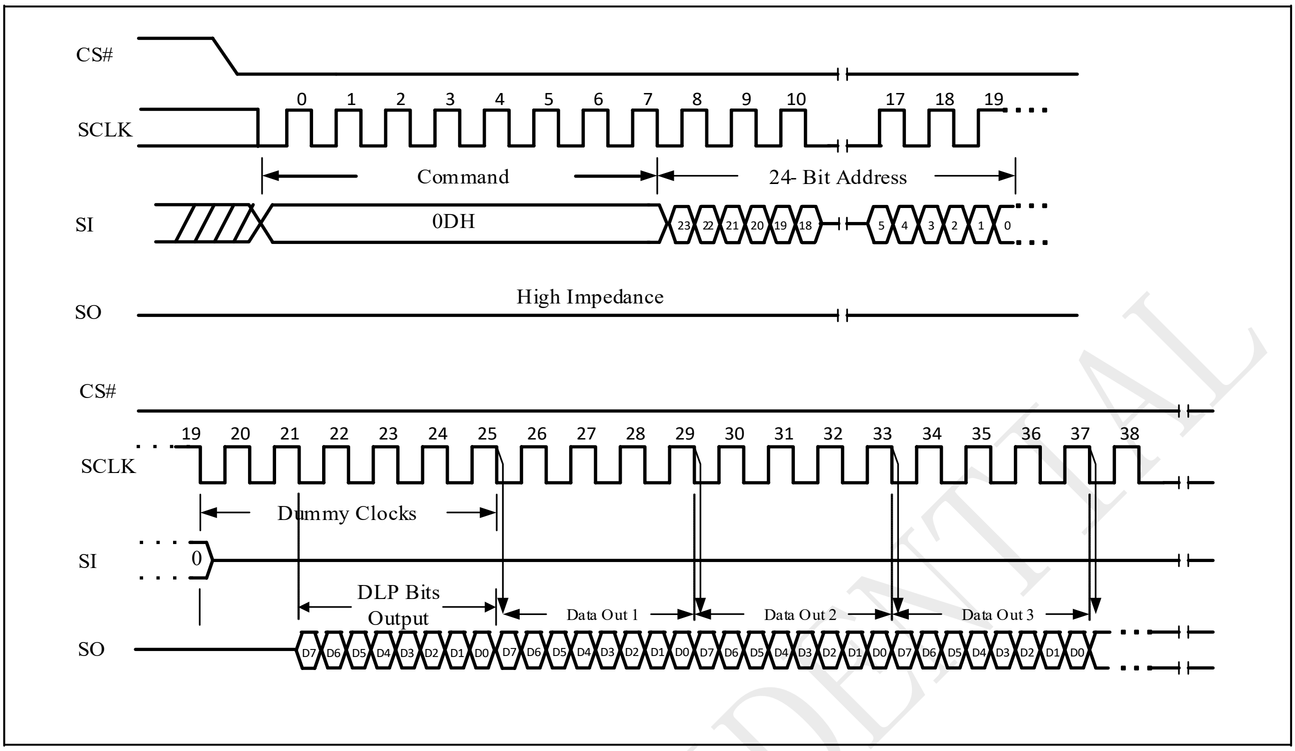


Figure 9-23j DTR Fast Read with DLP bits output Sequence in QPI mode

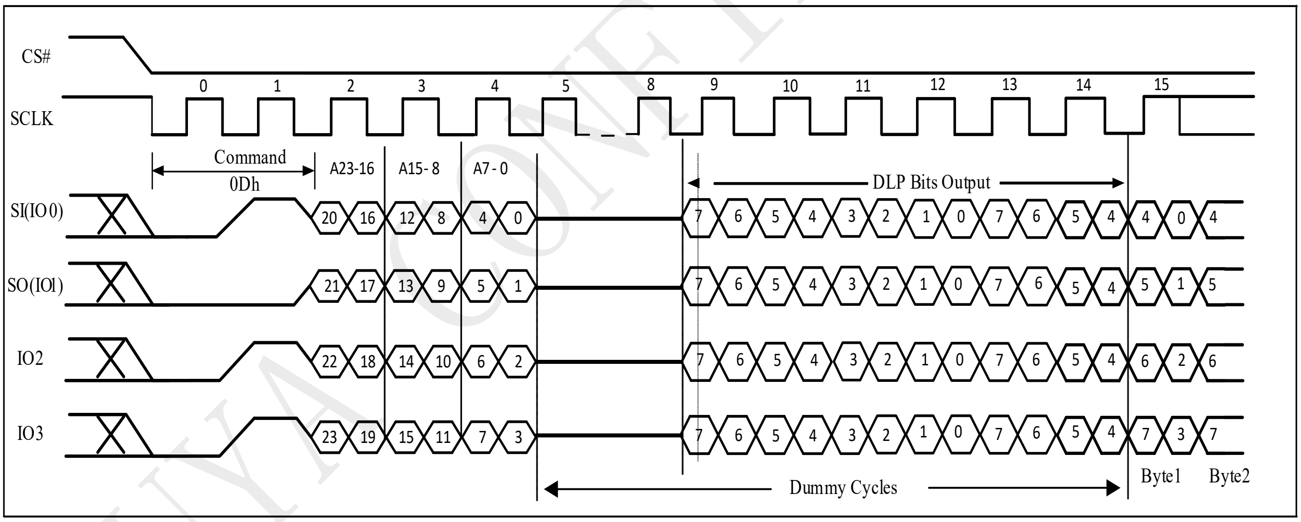


Figure 9-23k DTR 2IO Read with DLP bits output Sequence

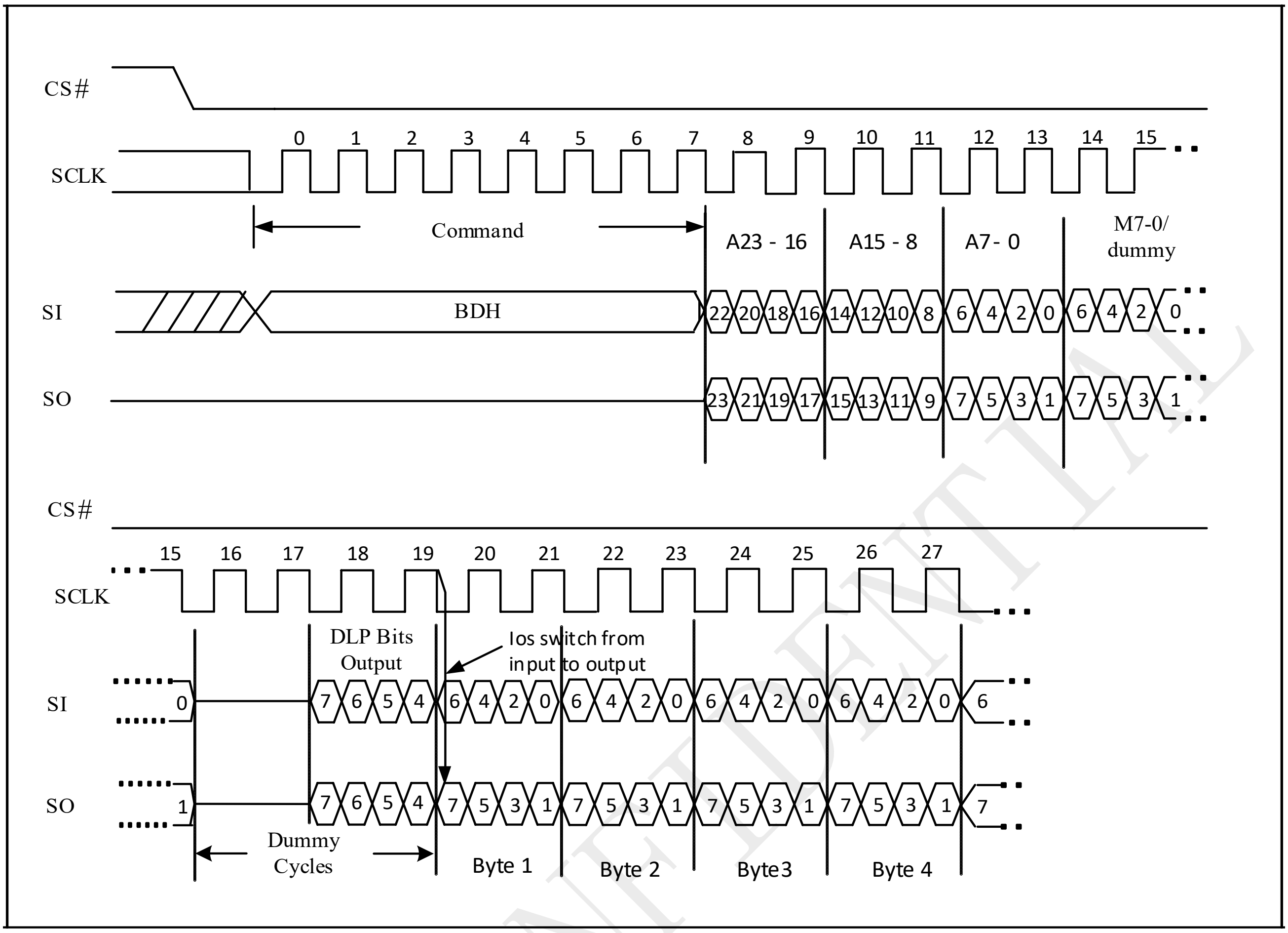


Figure 9-23l DTR 4IO Read with DLP bits output Sequence

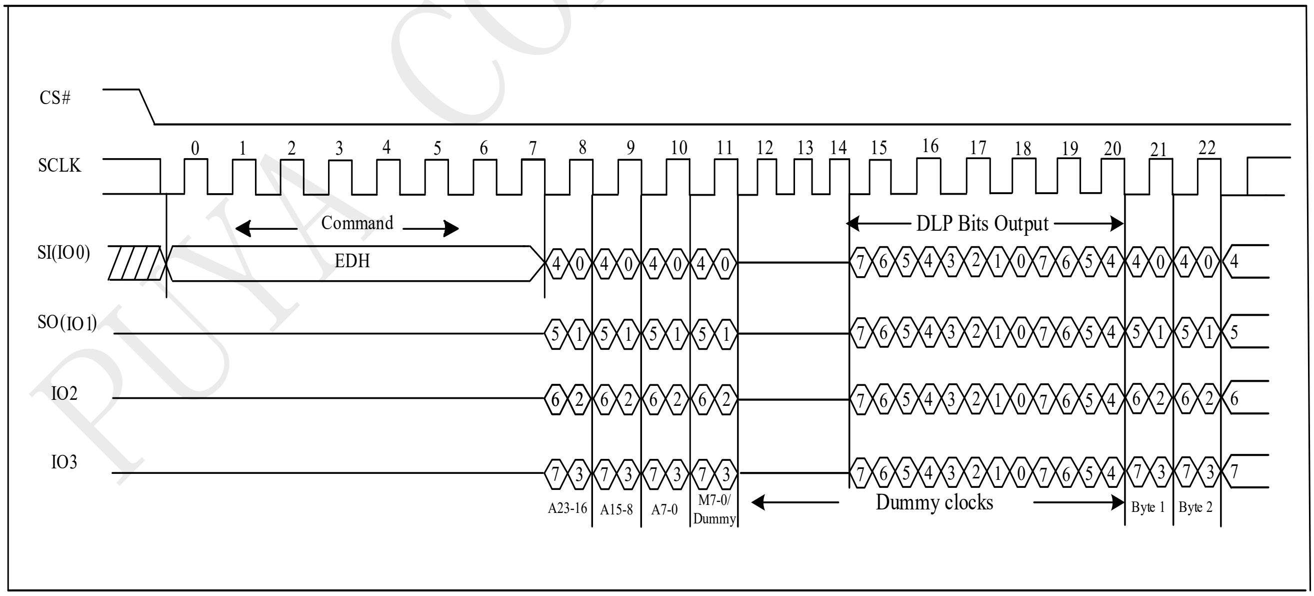


Figure 9-23m DTR 4IO Read with DLP bits output Sequence in QPI mode

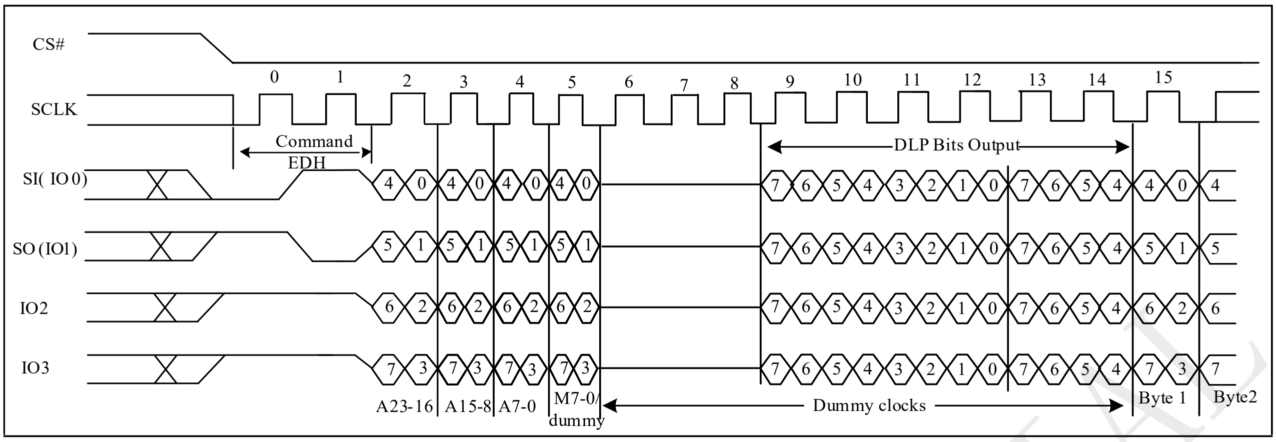


Figure 9-23n DTR Burst Read with Wrap with DLP bits output Sequence

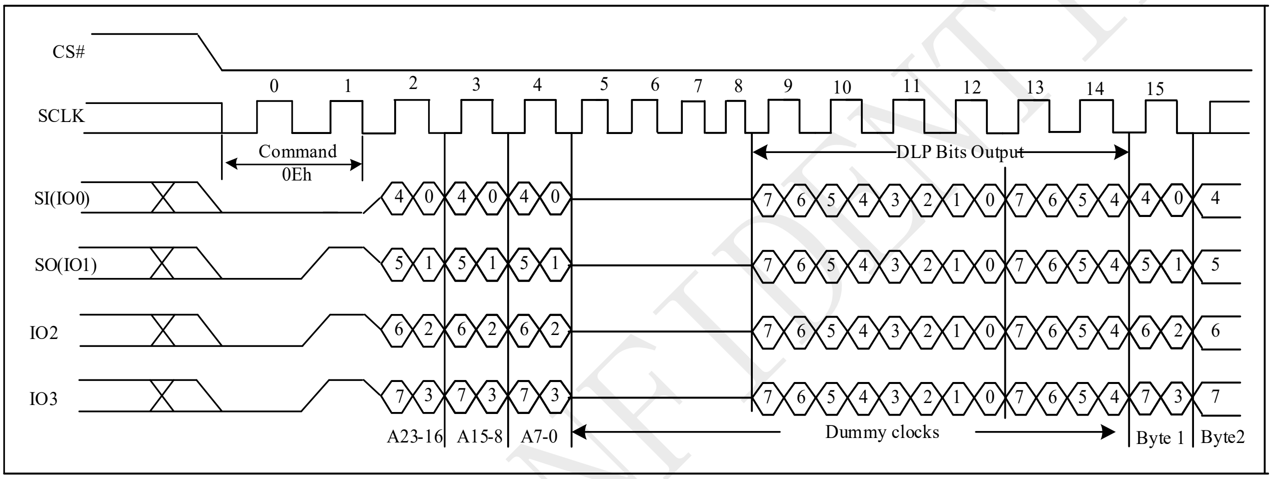


Figure 9-23o Read Security Registers with DLP bits output Sequence

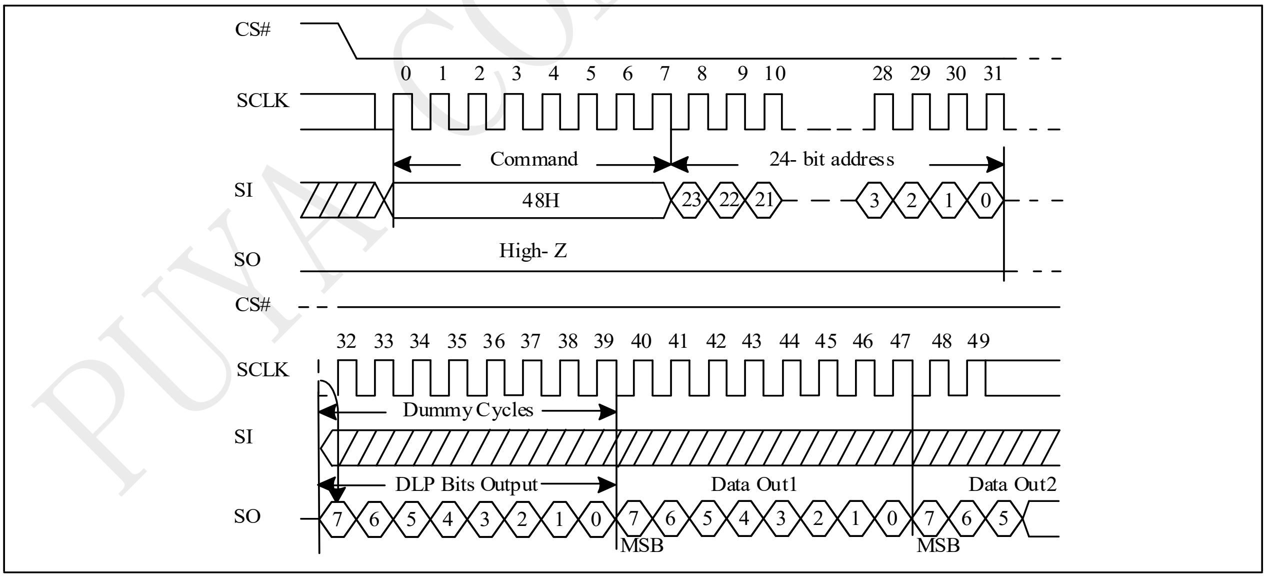


Figure 9-23p Read Unique ID with DLP bits output Sequence

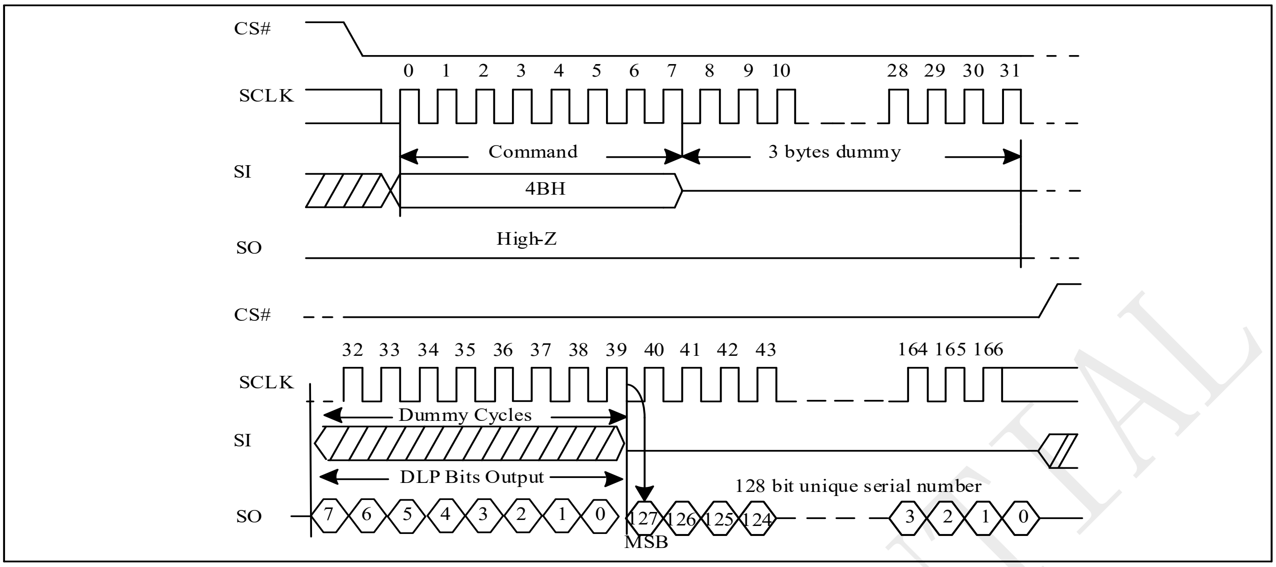


Figure 9-23q Read Serial Flash Discoverable Parameter with DLP bits output Sequence

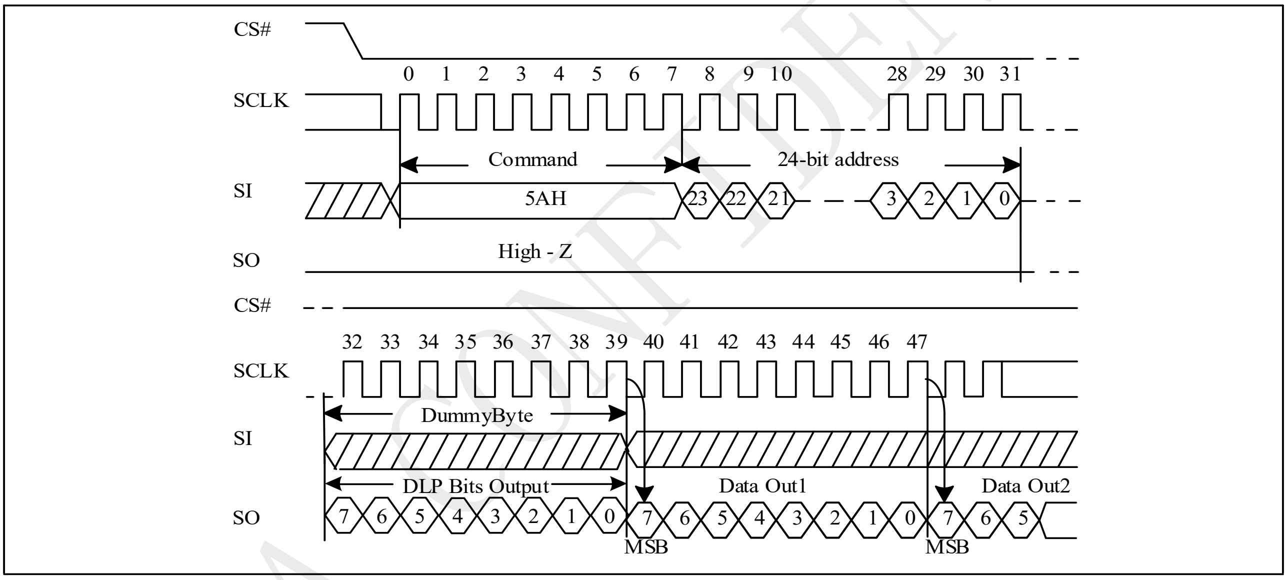
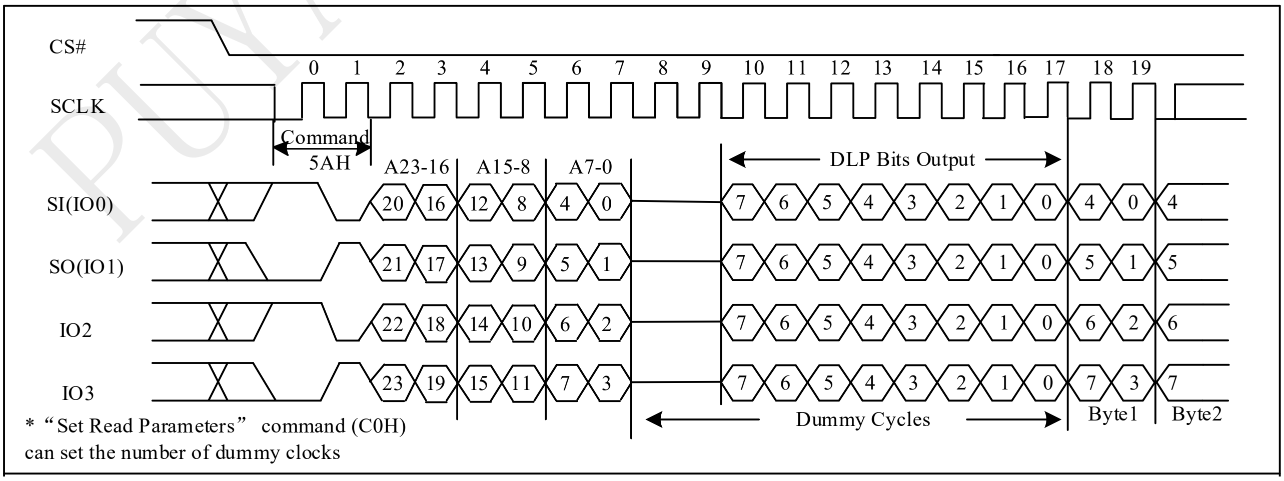


Figure 9-23r Read Serial Flash Discoverable Parameter with DLP bits output Sequence in QPI mode

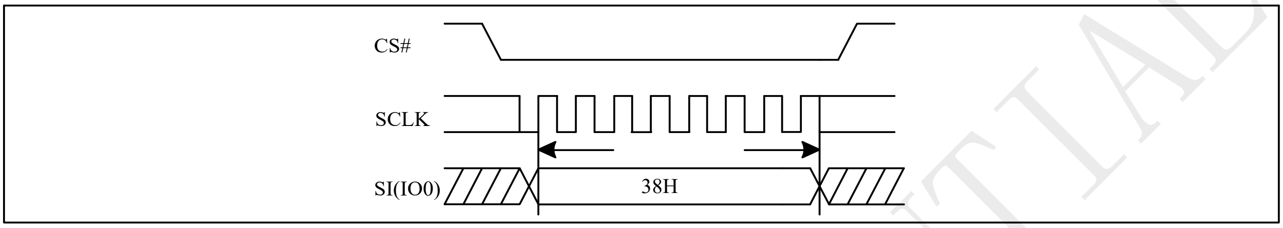


* "Set Read Parameters" command (C0H) can set the number of dummy clocks

9.24 Enable QPI (38H)

The device support both Standard/Dual/Quad SPI and QPI mode. The “Enable QPI (38H)” command can switch the device from SPI mode to QPI mode. See the command Table 2a for all support QPI commands. In order to switch the device to QPI mode, the Quad Enable (QE) bit in Status Register-1 must be set to 1 first, and “Enable QPI (38H)” command must be issued. If the QE bit is 0, the “Enable QPI (38H)” command will be ignored and the device will remain in SPI mode. When the device is switched from SPI mode to QPI mode, the existing Write Enable Latch and Program/Erase Suspend status, and the Wrap Length setting will remain unchanged.

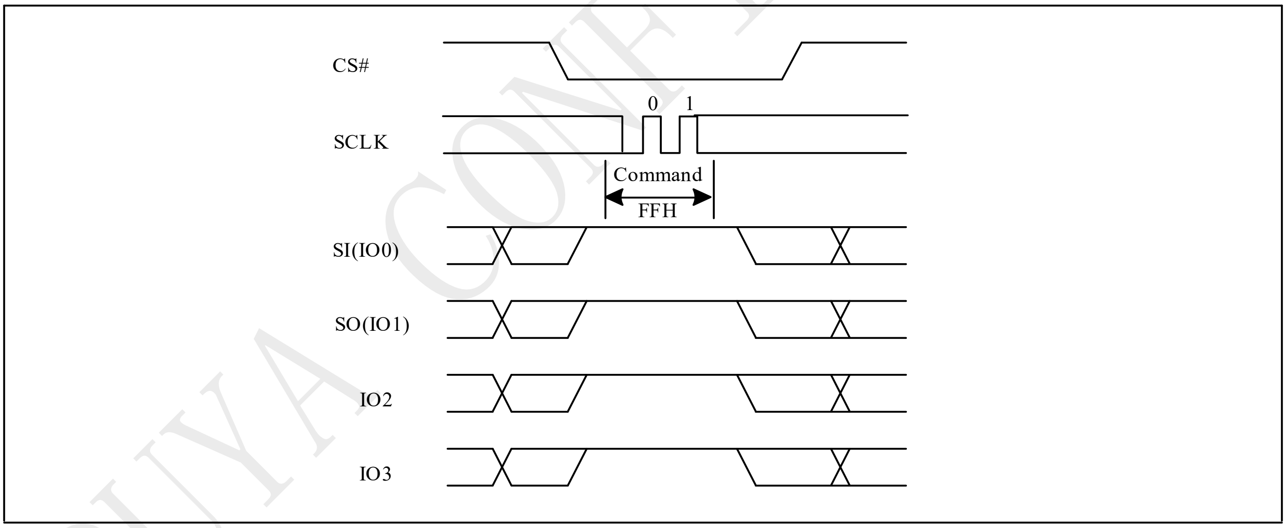
Figure 9-24 Enable QPI Sequence (38h)



9.25 Disable QPI (FFH)

To exit the QPI mode and return to Standard/Dual/Quad SPI mode, the “Disable QPI (FFH)” command must be issued. When the device is switched from QPI mode to SPI mode, the existing Write Enable Latch and Program/Erase Suspend status, and the Wrap Length setting will remain unchanged.

Figure 9-25 Disable QPI Sequence (QPI)



9.26 Sector Erase (SE) (20H)

The Sector Erase (SE) instruction is for erasing the data of the chosen sector to be "1". A Write Enable (WREN) Instruction must execute to set the Write Enable Latch (WEL) bit before sending the Sector Erase (SE). Any address of the sector is a valid address for Sector Erase (SE) instruction. The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

Address bits [Am-A12] (Am is the most significant address) select the sector address.

The sequence of issuing SE instruction is: CS# goes low→ sending SE instruction code→ 3-byte address on SI→ CS# goes high. The SIO [3:1] are don't care.

Figure 9-26 Sector Erase (SE) Sequence (Command 20h)

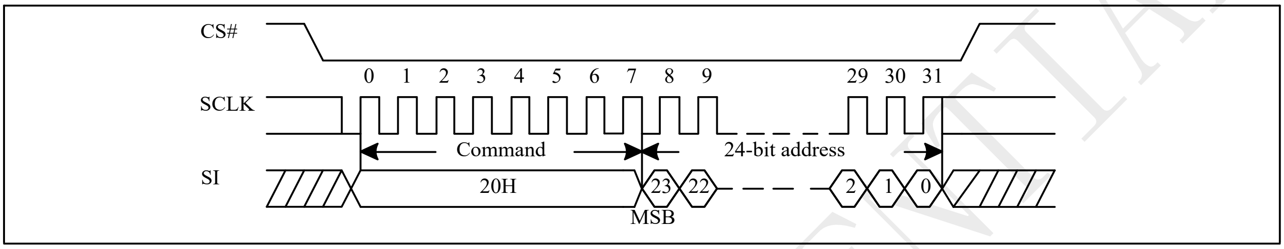
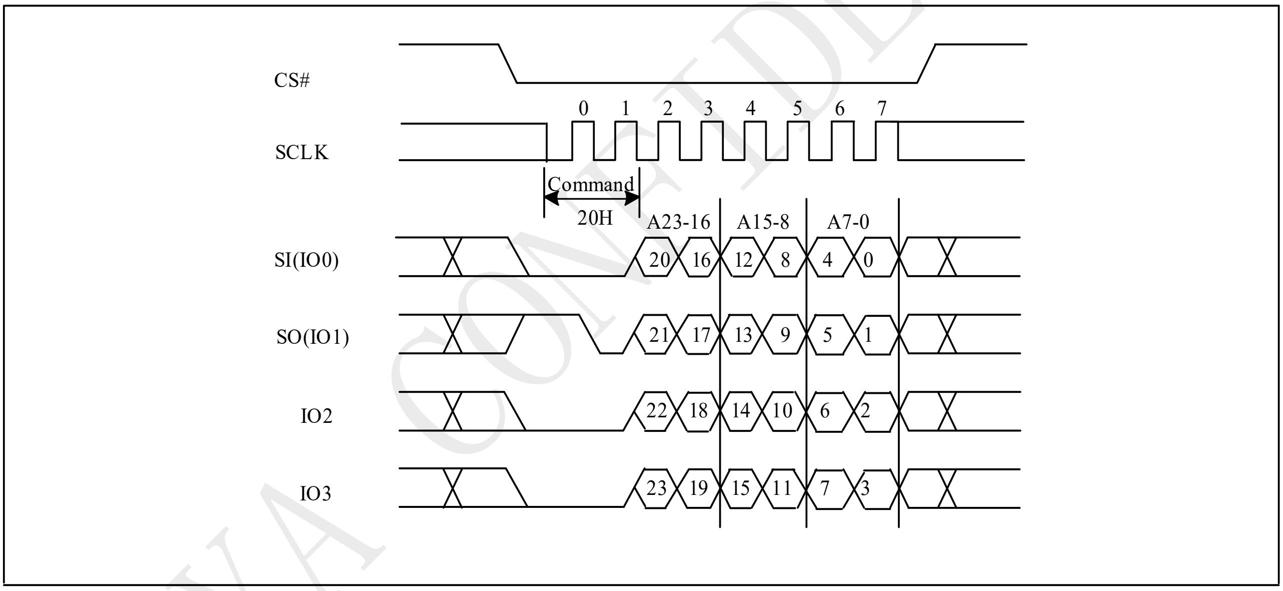


Figure 9-26a Sector Erase (SE) Sequence (QPI)



The self-timed Sector Erase Cycle time(tSE) is initiated as soon as Chip Select (CS#) goes high. The Write in progress (WIP) bit still can be check out during the Sector Erase cycle is in progress. The WIP sets 1 during the tSE timing, and sets0 when Sector Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the sector is protected by BP4, BP3, BP2, BP1, BP0 bits, the Sector Erase (SE) instruction will not be executed on the sector.

9.27 Block Erase (BE32K) (52H)

The Block Erase (BE32K) instruction is for erasing the data of the chosen block to be "1". The instruction is used for 32K-byte block erase operation. A Write Enable (WREN) instruction must be executed to set the Write Enable Latch (WEL) bit before sending the Block Erase (BE32K). Any address of the block is a valid address for Block Erase (BE32K) instruction. The CS# must go high exactly at the byte boundary (the least significant bit of address byte has been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing BE32K instruction is: CS# goes low → sending BE32K instruction code → 3-byte address on SI → CS# goes high. The SIO [3:1] are don't care.

The self-timed Block Erase Cycle time (tBE1) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked while the Block Erase cycle is in progress. The WIP sets during the tBE1 timing, and clears when Block Erase Cycle is completed, and the Write Enable Latch (WEL) bit is cleared. If the block is protected by BP4, BP3, BP2, BP1, BP0 bits, the array data will be protected (no change) and the WEL bit still be reset.

Figure 9-27 Block Erase 32K(BE32K) Sequence (Command 52h)

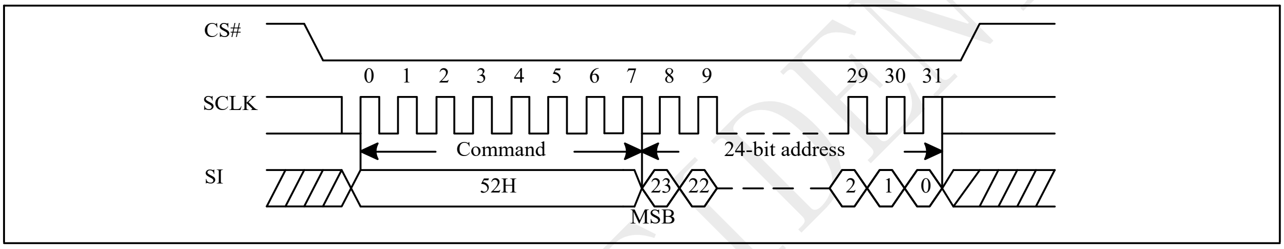
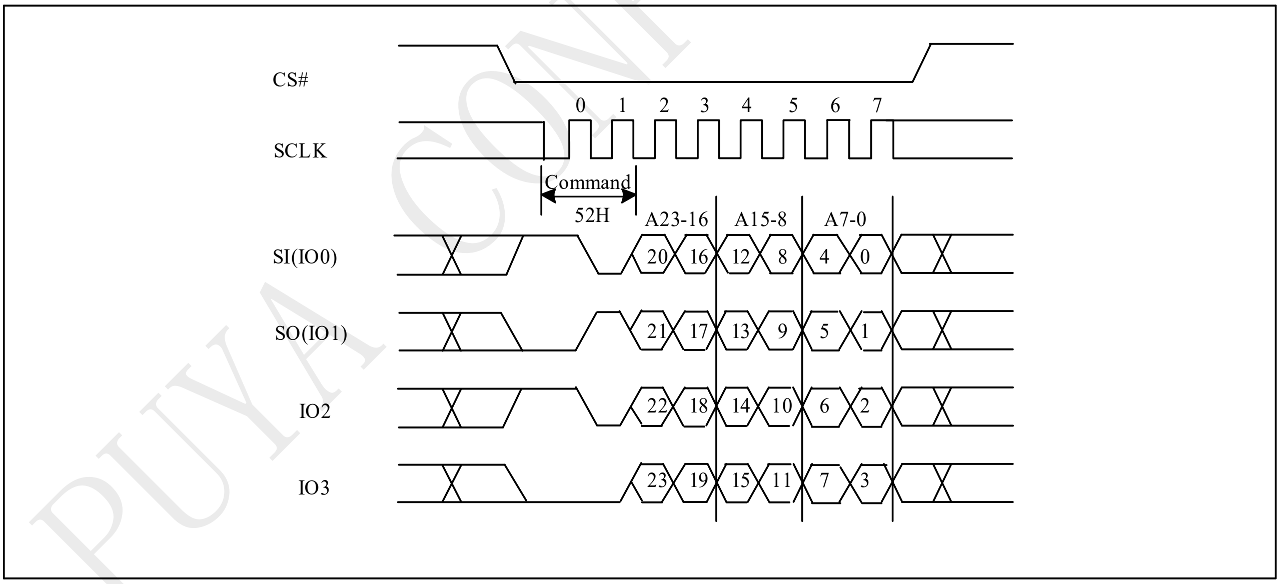


Figure 9-27a Block Erase 32K(BE32K) Sequence (QPI)



9.28 Block Erase (BE) (D8H)

The Block Erase (BE) instruction is for erasing the data of the chosen block to be "1". The instruction is used for 64K-byte block erase operation. A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Block Erase (BE). Any address of the block is a valid address for Block Erase (BE) instruction. The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing BE instruction is: CS# goes low→ sending BE instruction code→ 3-byte address on SI→CS# goes high. The SIO [3:1] are "don't care".

The self-timed Block Erase Cycle time (tBE2) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Block Erase cycle is in progress. The WIP sets 1 during the tBE2 timing, and sets 0 when Block Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the block is protected by BP4, BP3, BP2, BP1, BP0 bits, the Block Erase (BE) instruction will not be executed on the block.

Figure 9-28 Block Erase (BE) Sequence (Command D8h)

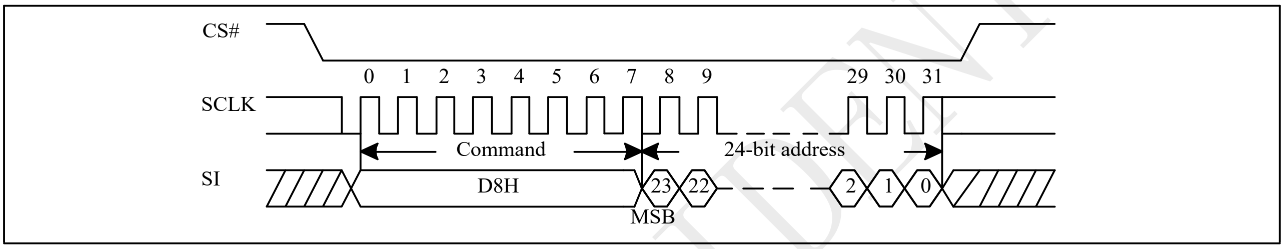
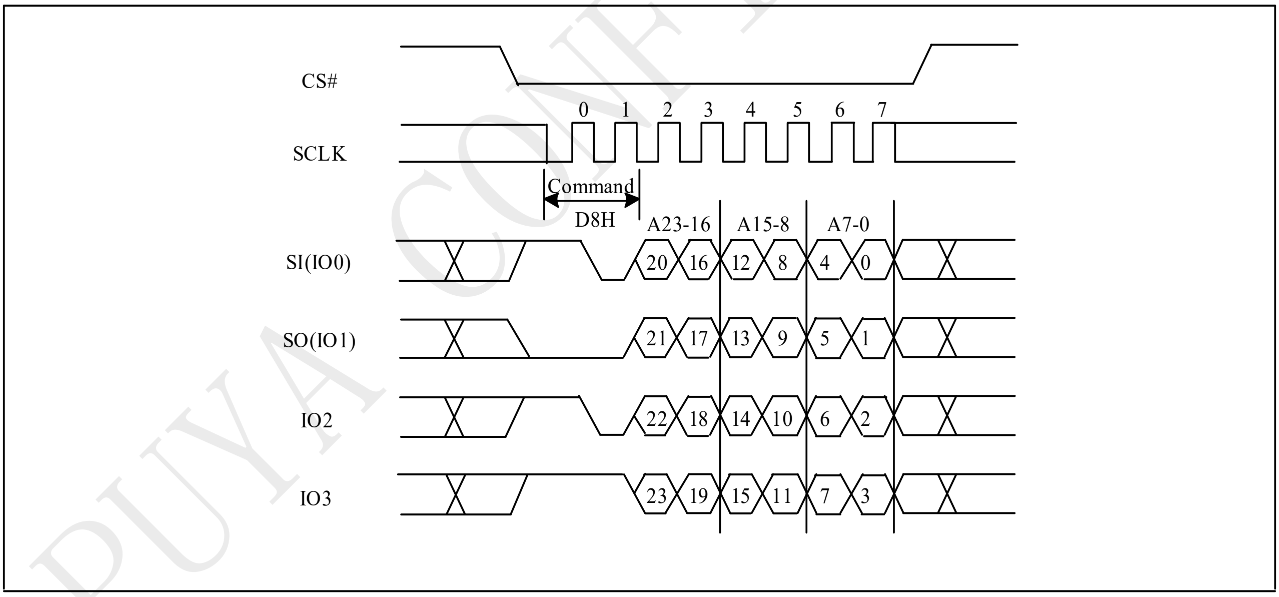


Figure 9-28a Block Erase (BE) Sequence (QPI)



9.29 Chip Erase (CE) (60H/C7H)

The Chip Erase (CE) instruction is for erasing the data of the whole chip to be "1". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Chip Erase (CE). The CS# must go high exactly at the byte boundary (the latest eighth of address byte been latched-in); otherwise, the instruction will be rejected and not executed.

The sequence of issuing CE instruction is: CS# goes low → sending CE instruction code → CS# goes high. The SIO [3:1] are "don't care".

The self-timed Chip Erase Cycle time (tCE) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Chip Erase cycle is in progress. The WIP sets 1 during the tCE timing, and sets 0 when Chip Erase Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the chip is protected by BP4, BP3, BP2, BP1, BP0 bits, the Chip Erase (CE) instruction will not be executed. It will be only executed when all Block Protect (BP4, BP3, BP2, BP1, BP0) are set to “None protected”.

Figure 9-29 Chip Erase (CE) Sequence (Command 60h or C7h)

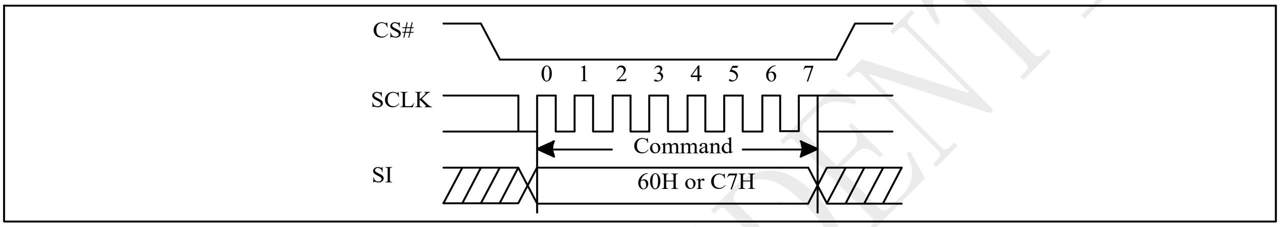
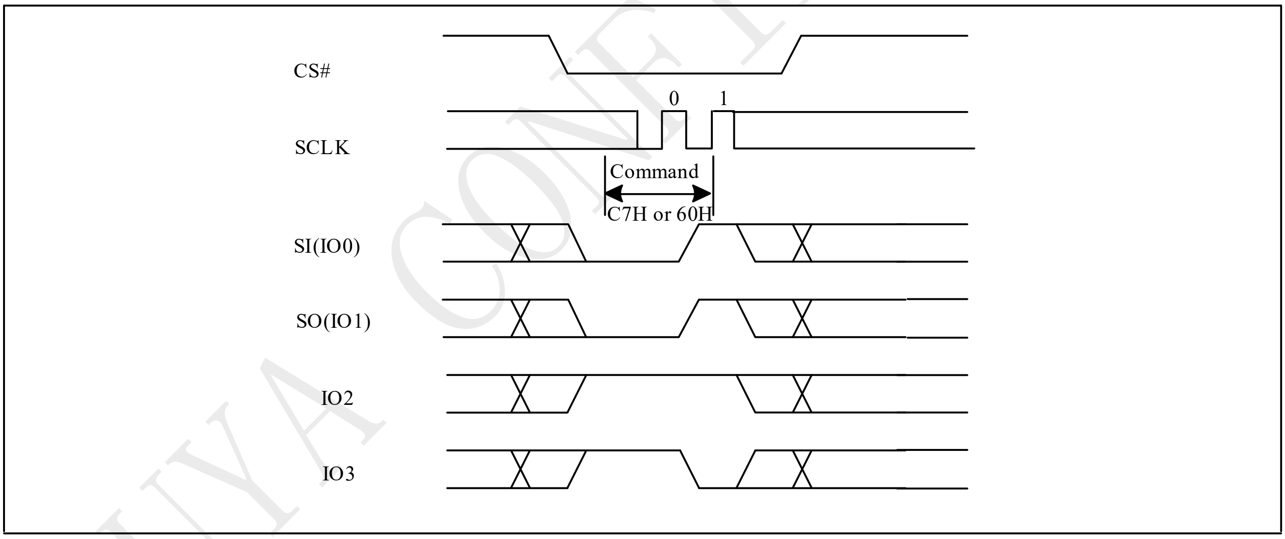


Figure 9-29a Chip Erase (CE) Sequence (QPI)



9.30 Page Program (PP) (02H)

The Page Program (PP) instruction is for programming the memory to be "0". A Write Enable (WREN) instruction must execute to set the Write Enable Latch (WEL) bit before sending the Page Program (PP). The device programs only the last 256 data bytes sent to the device. If the entire 256 data bytes are going to be programmed, A7-A0(The eight least significant address bits) should be set to 0. If the eight least significant address bits(A7-A0) are not all 0, all transmitted data going beyond the end of the current page are programmed from the start address of the same page (from the address A7-A0 are all 0). If more than 256 bytes are sent to the device, the data of the last 256-byte is programmed at the request page and previous data will be disregarded. If less than 256 bytes are sent to the device, the data is programmed at the requested address of the page.

For the very best performance, programming should be done in full pages of 256 bytes aligned on 256-byte boundaries with each Page being programmed only once. Using the Page Program (PP) command to load an entire page, within the page boundary, will save overall programming time versus loading less than a page into the program buffer.

It is possible to program from one byte up to a page size in each Page programming operation. Please refer to the PY25Q serial flash application note for multiple byte program operation within one page.

The sequence of issuing PP instruction is: CS# goes low→ sending PP instruction code→3-byte address on SI→ at least 1-byte on data on SI→ CS# goes high.

The CS# must be kept low during the whole Page Program cycle; The CS# must go high exactly at the byte boundary (the latest eighth bit of data being latched in), otherwise the instruction will be rejected and will not be executed.

The self-timed Page Program Cycle time (tPP) is initiated as soon as Chip Select (CS#) goes high. The Write in Progress (WIP) bit still can be checked during the Page Program cycle is in progress. The WIP sets 1 during the tPP timing, and sets 0 when Page Program Cycle is completed, and the Write Enable Latch (WEL) bit is reset. If the page is protected by BP4, BP3, BP2, BP1, BP0 bits, the Page Program (PP) instruction will not be executed. The SIO [3:1] are "don't care".

Figure 9-30 Page Program (PP) Sequence (Command 02h)

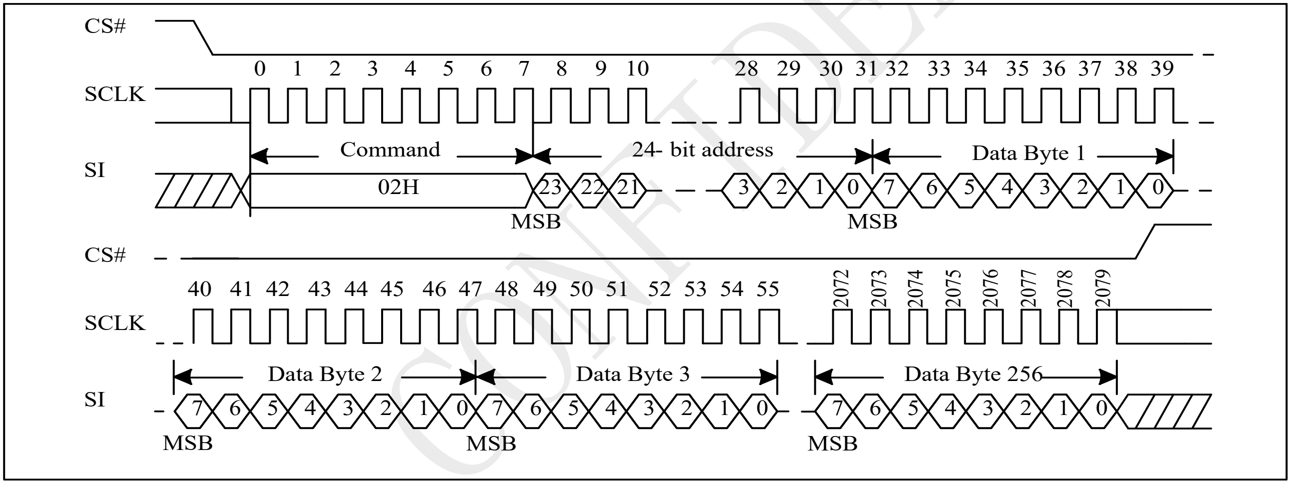
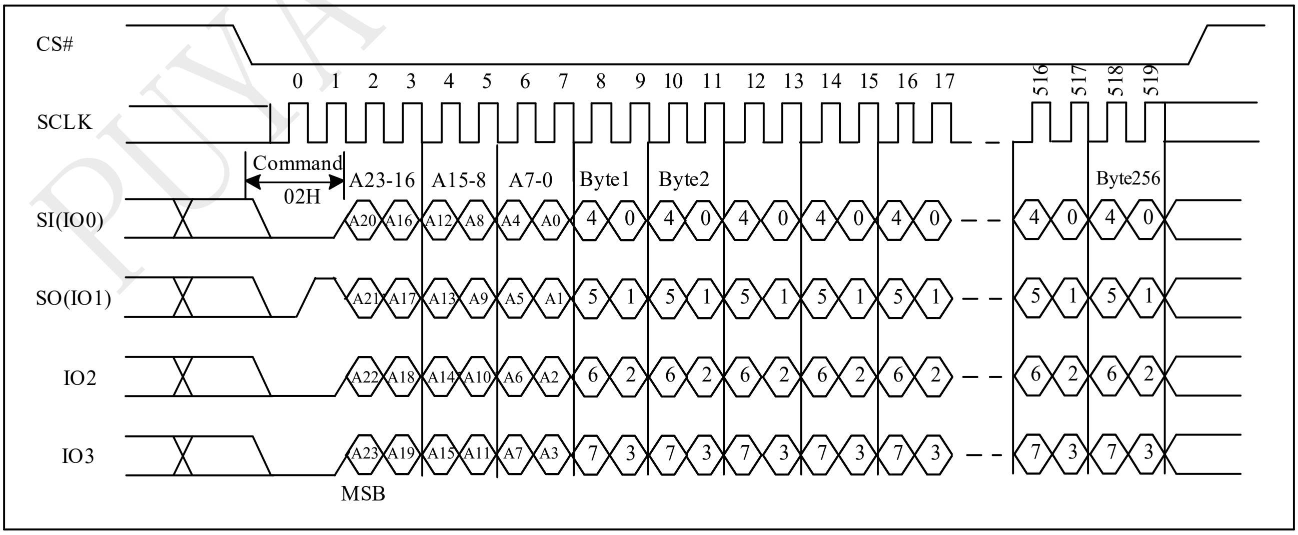


Figure 9-30a Page Program (PP) Sequence (QPI)

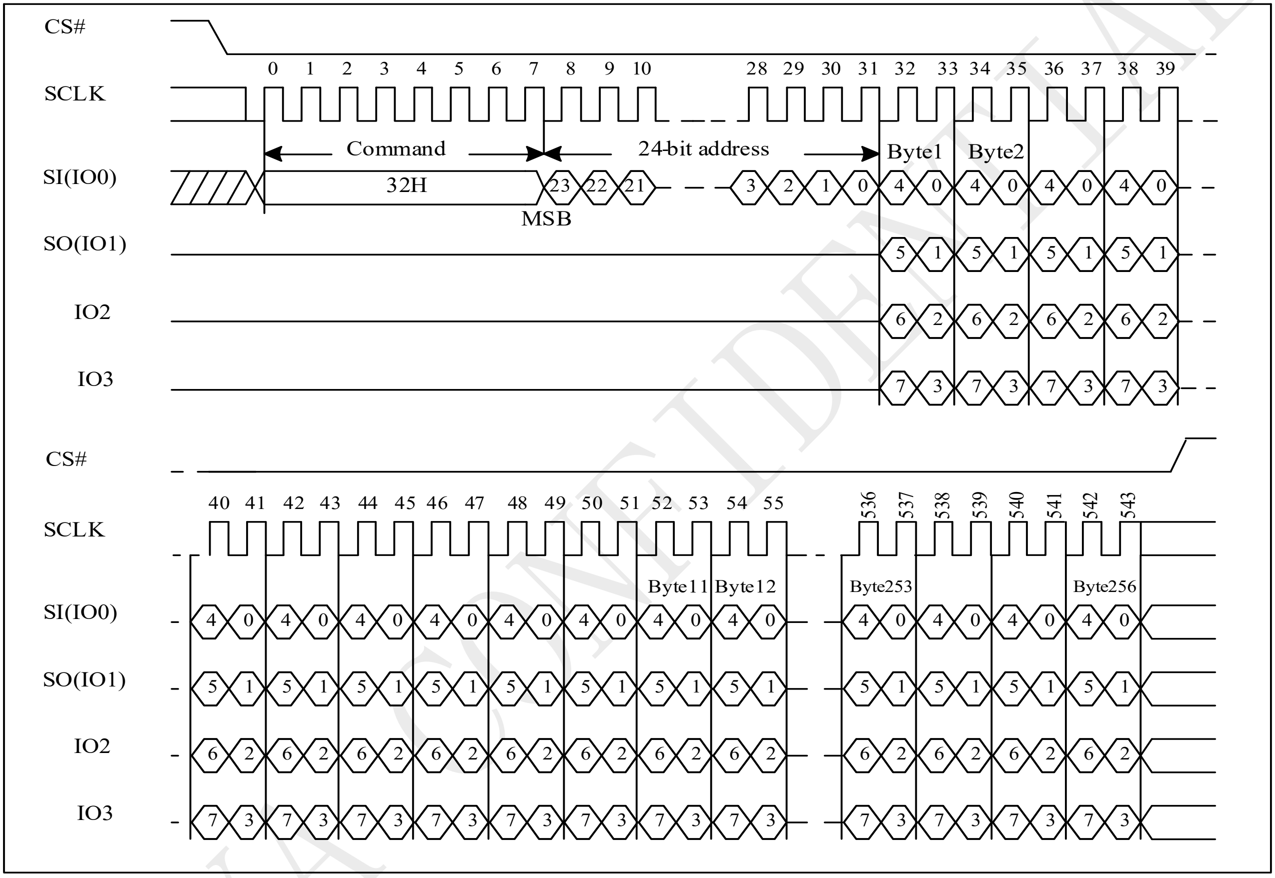

CS#
SCLK
SI(IO0)
SO(IO1)
IO2
IO3
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
516 517 518 519
Command
02H
A23-16
A15-8
A7-0
Byte1
Byte2
Byte256
A20 A16 A12 A8 A4 A0 4 0 4 0 4 0 4 0 4 0 4 0 4 0
A21 A17 A13 A9 A5 A1 5 1 5 1 5 1 5 1 5 1 5 1 5 1
A22 A18 A14 A10 A6 A2 6 2 6 2 6 2 6 2 6 2 6 2 6 2
A23 A19 A15 A11 A7 A3 7 3 7 3 7 3 7 3 7 3 7 3 7 3
MSB

9.31 Quad Page Program (QPP) (32H)

The Quad Page Program (QPP) instruction is for programming the memory to be "0". A Write Enable (WREN)instruction must execute to set the Write Enable Latch (WEL) bit and Quad Enable (QE) bit must be set to "1" before sending the Quad Page Program (QPP). The Quad Page Programming takes four pins: SIO0, SIO1, SIO2, and SIO3 as data input, which can improve programmer performance and the effectiveness of application. The QPP operation frequency supports as fast as fQPP. The other function descriptions are as same as standard page program.

The sequence of issuing QPP instruction is: CS# goes low→ sending QPP instruction code→ 3-byte address on SIO0 → at least 1-byte on data on SIO [3:0] →CS# goes high.

Figure 9-31 Quad Page Program (QPP) Sequence (Command 32h)



9.32 Erase Security Registers (ERSCUR) (44H)

The product provides three 1024-byte Security Registers 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 Security Registers command is similar to Sector/Block Erase command. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit.

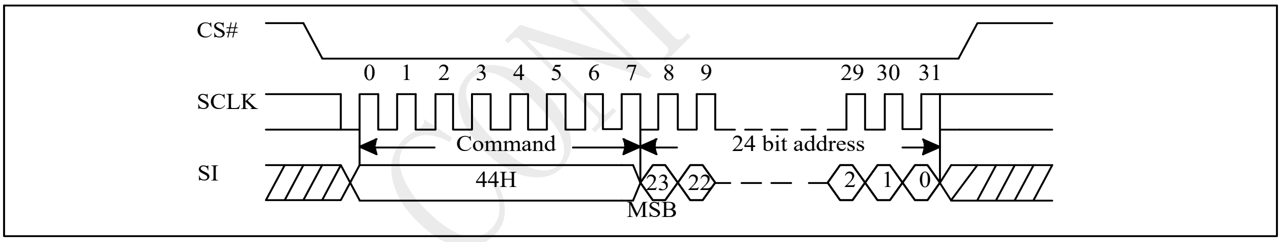
The Erase Security Registers command sequence: CS# goes low → sending ERSCUR instruction →□sending 24-bit address → CS# goes high.

CS# must be driven high after the eighth bit of the command code has been latched in; otherwise the Erase Security Registers command is not executed. As soon as CS# is driven high, the self-timed Erase Security Registers cycle (whose duration is tSE) is initiated. While the Erase Security Registers 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 Erase Security Registers cycle, and is 0 when it is completed. The Security Registers Lock Bit (LB3-1) in the Status Register can be used to OTP protect the security registers.

Once the LB bit is set to 1, the Security Registers will be permanently locked; the Erase Security Registers command will be ignored.

Address	A23-16	A15-12	A11-10	A9-0
Security Register #1	00H	0001	00	Don't care
Security Register #2	00H	0010	00	Don't care
Security Register #3	00H	0011	00	Don't care

Figure 9-32 Erase Security Registers (ERSCUR) Sequence (Command 44h)



9.33 Program Security Registers (PRSCUR) (42H)

The Program Security Registers command is similar to the Page Program command. It allows from 1 to 256bytes Security Registers data to be programmed. A Write Enable (WREN) command must previously have been executed to set the Write Enable Latch (WEL) bit before sending the Program Security Registers command.

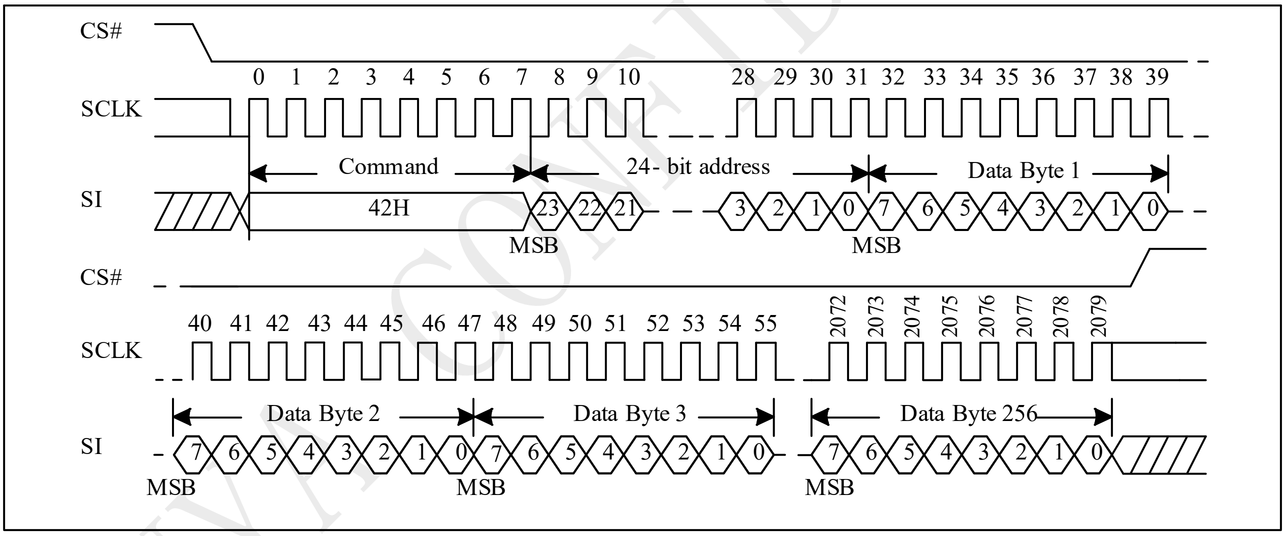
The Program Security Registers command sequence: CS# goes low → sending PRSCUR instruction →□sending 24-bit address → sending at least one-byte data → CS# goes high.

As soon as CS# is driven high, the self-timed Program Security Registers cycle (whose duration is tPP) is initiated. While the Program Security Registers 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 Program Security Registers cycle, and is 0 when it is completed.

If the Security Registers Lock Bit (LB3-1) is set to 1, the Security Registers will be permanently locked. Program Security Registers command will be ignored.

Address	A23-16	A15-12	A11-10	A9-0
Security Register #1	00H	0001	00	Byte Address
Security Register #2	00H	0010	00	Byte Address
Security Register #3	00H	0011	00	Byte Address

Figure 9-33 Program Security Registers (PRSCUR) Sequence (Command 42h)



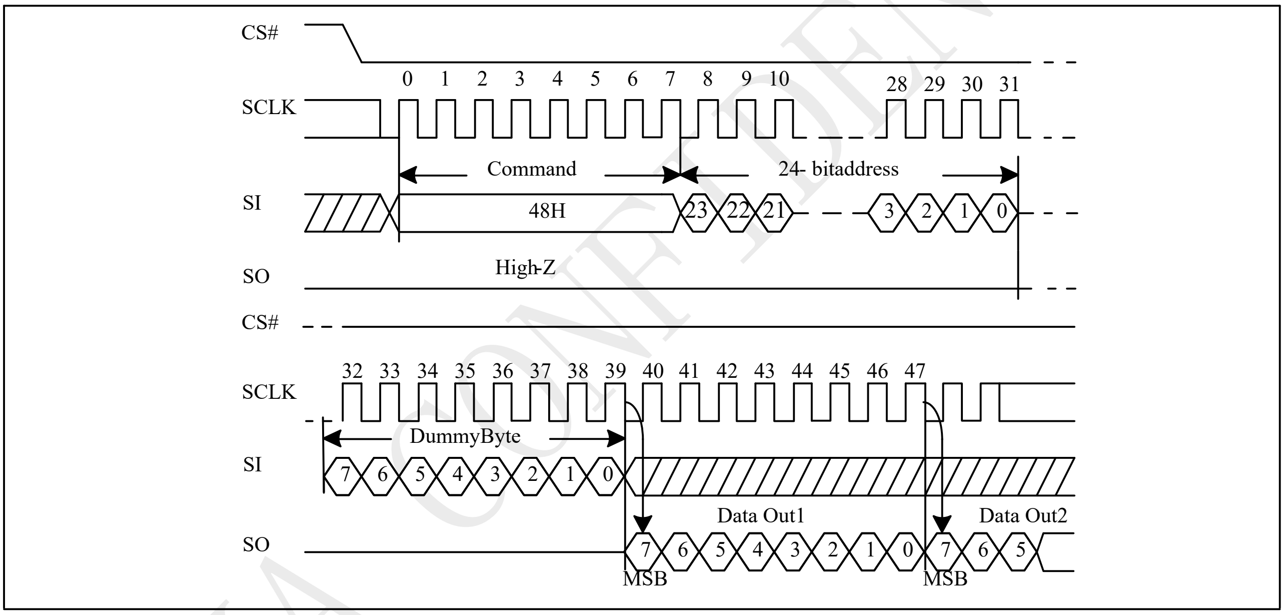
9.34 Read Security Registers (RDSCUR) (48H)

The Read Security Registers command is similar to Fast Read command. The command is followed by a 3-byte address (A23-A0) and a dummy byte, each bit being latched-in during the rising edge of SCLK. Then the memory content, at that address, is shifted out on SO, each bit being shifted out, at a Max frequency fC, during the falling edge of SCLK. 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. Once the A9-A0 address reaches the last byte of the register (Byte 3FFH), it will reset to 000H, the command is completed by driving CS# high.

The sequence of issuing RDSCUR instruction is: CS# goes low → sending RDSCUR instruction → sending 24-bit address → 8-bit dummy byte → Security Register data out on SO → CS# goes high.

Address	A23-16	A15-12	A11-10	A9-0
Security Register #1	00H	0001	00	Byte Address
Security Register #2	00H	0010	00	Byte Address
Security Register #3	00H	0011	00	Byte Address

Figure 9-34 Read Security Registers (RDSCUR) Sequence (Command 48h)



9.35 Deep Power-down (DP) (B9H)

The Deep Power-down (DP) instruction is for setting the device on the minimizing the power consumption (to entering the Deep Power-down mode), the standby current is reduced from ISB1 to ISB2). The Deep Power-down mode requires the Deep Power-down (DP) instruction to enter, during the Deep Power-down mode, the device is not active and all Write/Program/Erase instruction are ignored. When CS# goes high, it's only in standby mode not deep power-down mode. It's different from Standby mode.

The sequence of issuing DP instruction is: CS# goes low→ sending DP instruction code→ CS# goes high.

Once the DP instruction is set, all instruction will be ignored except the Release from Deep Power-down mode (RDP), Read Electronic Signature (RES) instruction, and soft reset instruction (66H, 99H). (RES instruction to allow the ID been read out). When Power- down, the deep power-down mode automatically stops, and when power-up, the device automatically is in standby mode. For RDP instruction the CS# must go high exactly at the byte boundary (the latest eighth bit of instruction code been latched-in); otherwise, the instruction will not be executed. As soon as Chip Select (CS#) goes high, a delay of tDP is required before entering the Deep Power-down mode and reducing the current to ISB2.

Figure 9-35 Deep Power-down (DP) Sequence (Command B9h)

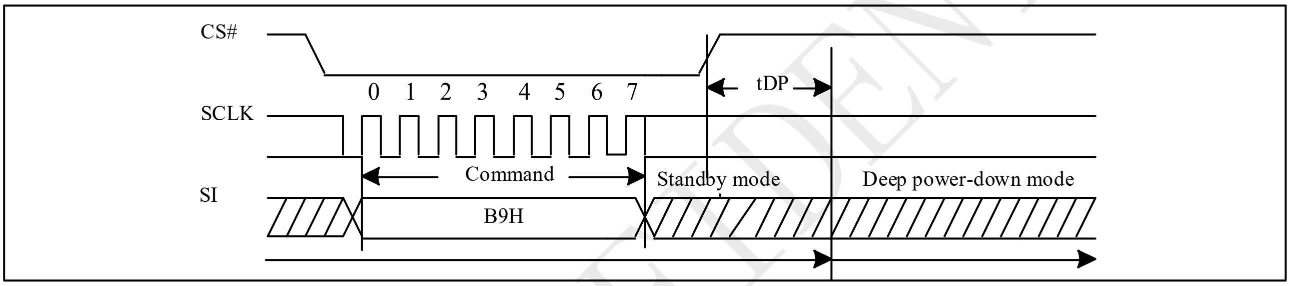
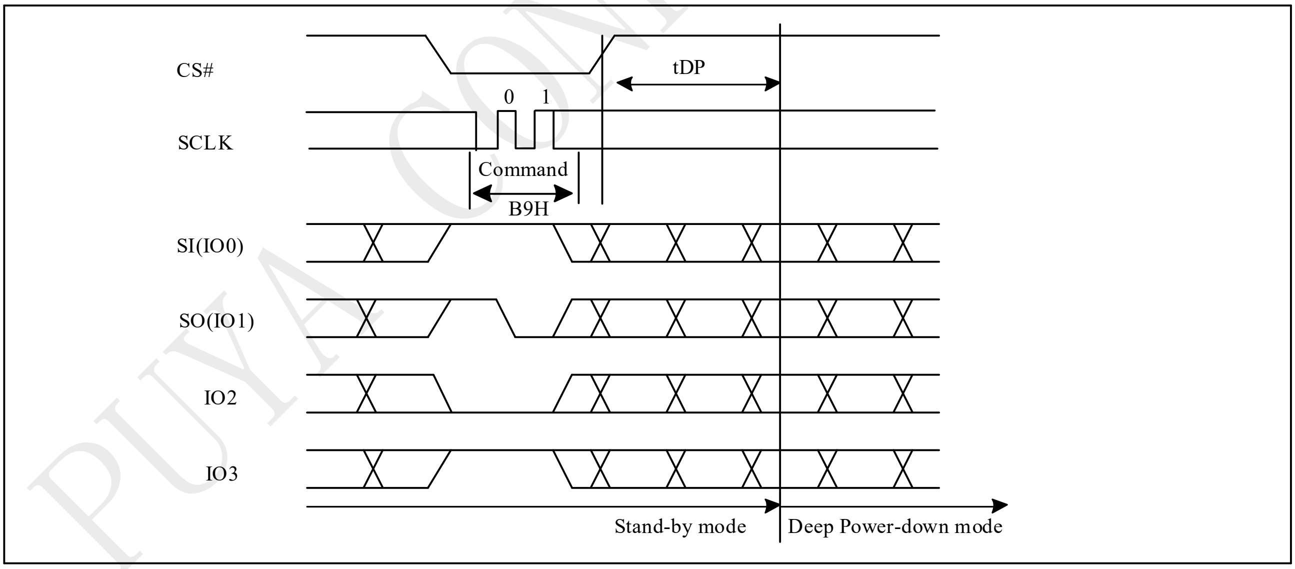


Figure 9-35a Deep Power-down (DP) Sequence (QPI)



9.36 Release from Deep Power-Down (RDP), Read Electronic Signature (RES) (ABH)

The Release from Deep Power-down (RDP) instruction is terminated by driving Chip Select (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 Stand-by Power mode is delayed by tRES2, and Chip Select (CS#) must remain High for at least tRES2(max). Once in the Stand-by Power mode, the device waits to be selected, so that it can receive, decode and execute instructions.

RES instruction is for reading out the old style of 8-bit Electronic Signature, whose values are shown as table of ID Definitions. This is not the same as RDID instruction. It is not recommended to use for new design. For new design, please use RDID instruction. Even in Deep power-down mode, the RDP and RES are also allowed to be executed, and even the device is in progress of program/erase/write cycle; there's no effect on the current program/erase/ write cycle in progress.

The RES instruction is ended by CS# goes high after the ID been read out at least once. The ID outputs repeatedly if continuously send the additional clock cycles on SCLK while CS# is at low. If the device was not previously in Deep Power-down mode, the device transition to standby mode is immediate. If the device was previously in Deep Power-down mode, there's a delay of tRES2 to transit to standby mode, and CS# must remain to high at least tRES2 (max). Once in the standby mode, the device waits to be selected, so it can be received code, and execute instruction.

The RDP instruction is for releasing from Deep Power-Down Mode.

Figure 9-36 Read Electronic Signature (RES) Sequence (Command ABh)

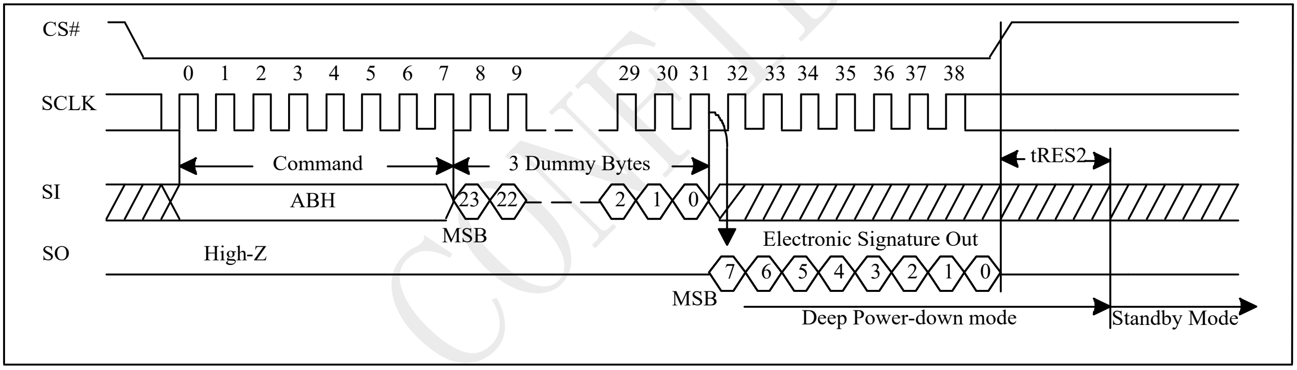


Figure 9-36a Read Electronic Signature (RES) Sequence (QPI)

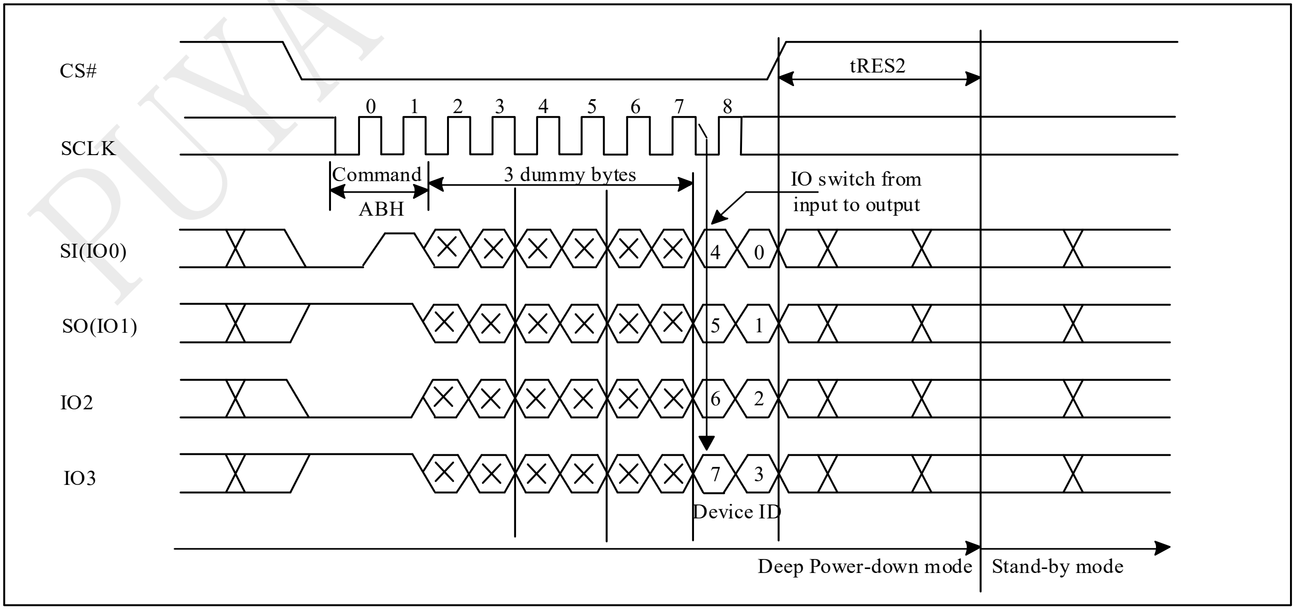


Figure 9-36b Release from Deep Power-down (RDP) Sequence (Command ABh)

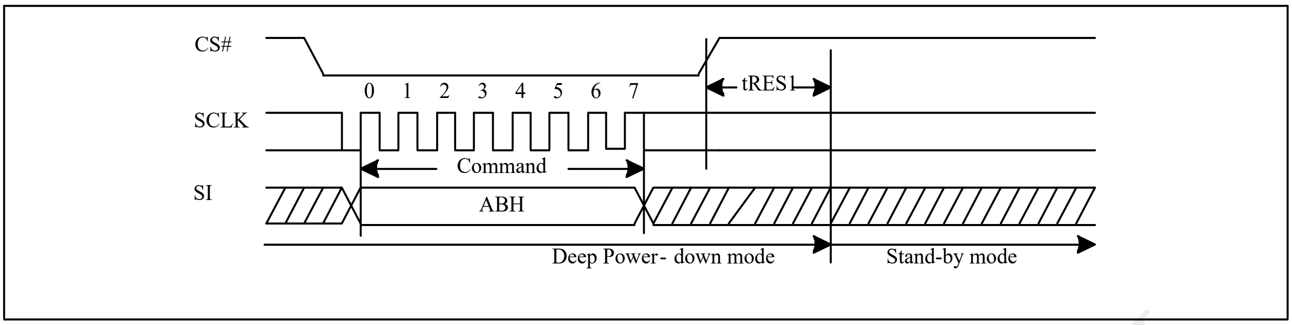
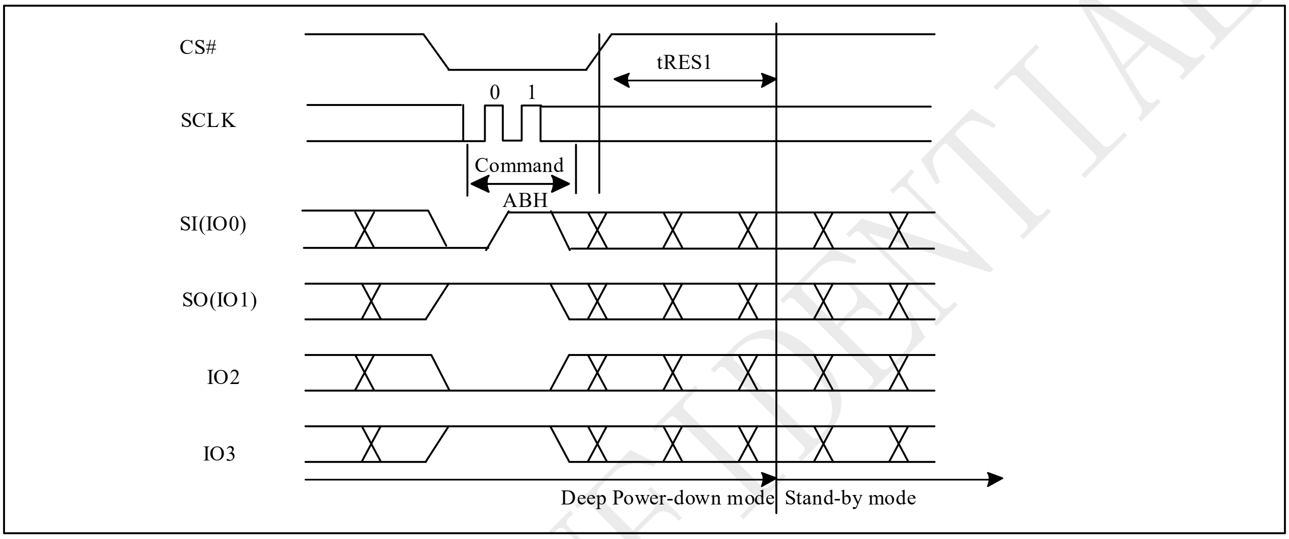


Figure 9-36c Release from Deep Power-down (RDP) Sequence (QPI)



9.37 Read Electronic Manufacturer ID & Device ID (REMS) (90H)

The REMS instruction returns both the JEDEC assigned manufacturer ID and the device ID. The Device ID values are listed in "Table ID Definitions".

The REMS instruction is initiated by driving the CS# pin low and sending the instruction code "90h" followed by two dummy bytes and one address byte (A7~A0). After which the manufacturer ID for PUYA (85h) and the device ID are shifted out on the falling edge of SCLK with the most significant bit (MSB) first. If the address byte is 00h, the manufacturer ID will be output first, followed by the device ID. If the address byte is 01h, then the device ID will be output first, followed by the manufacturer ID. While CS# is low, the manufacturer and device IDs can be read continuously, alternating from one to the other. The instruction is completed by driving CS# high.

Figure 9-37 Read Electronic Manufacturer & Device ID (REMS) Sequence (Command 90h)

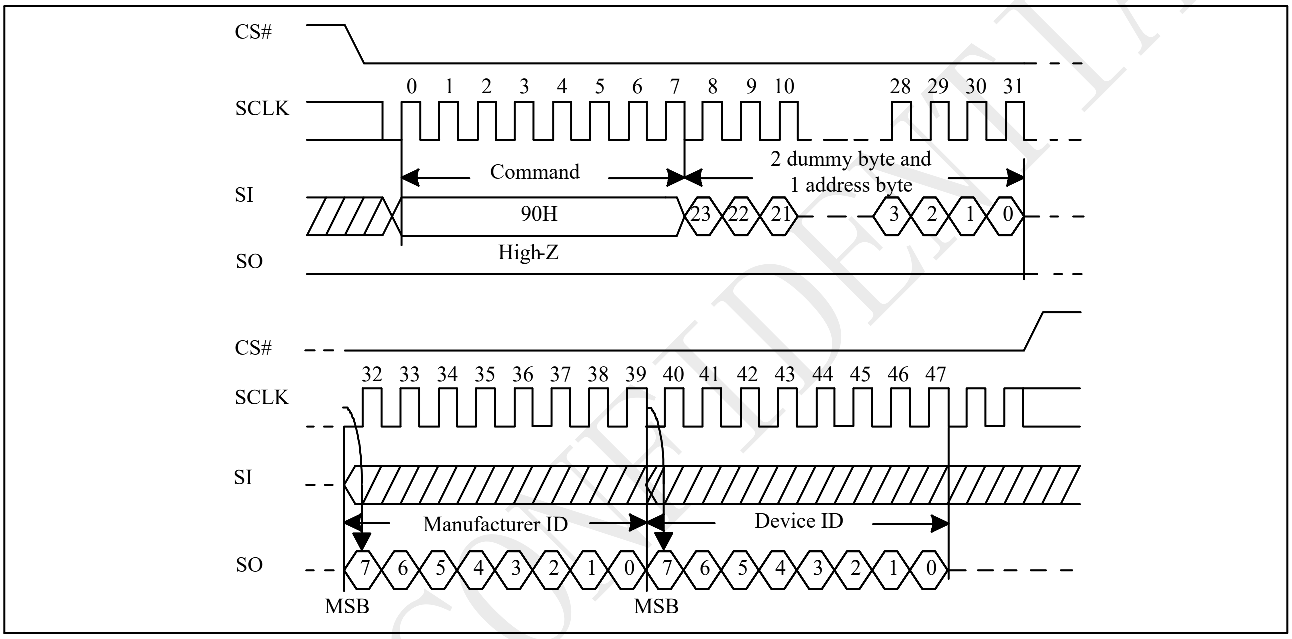
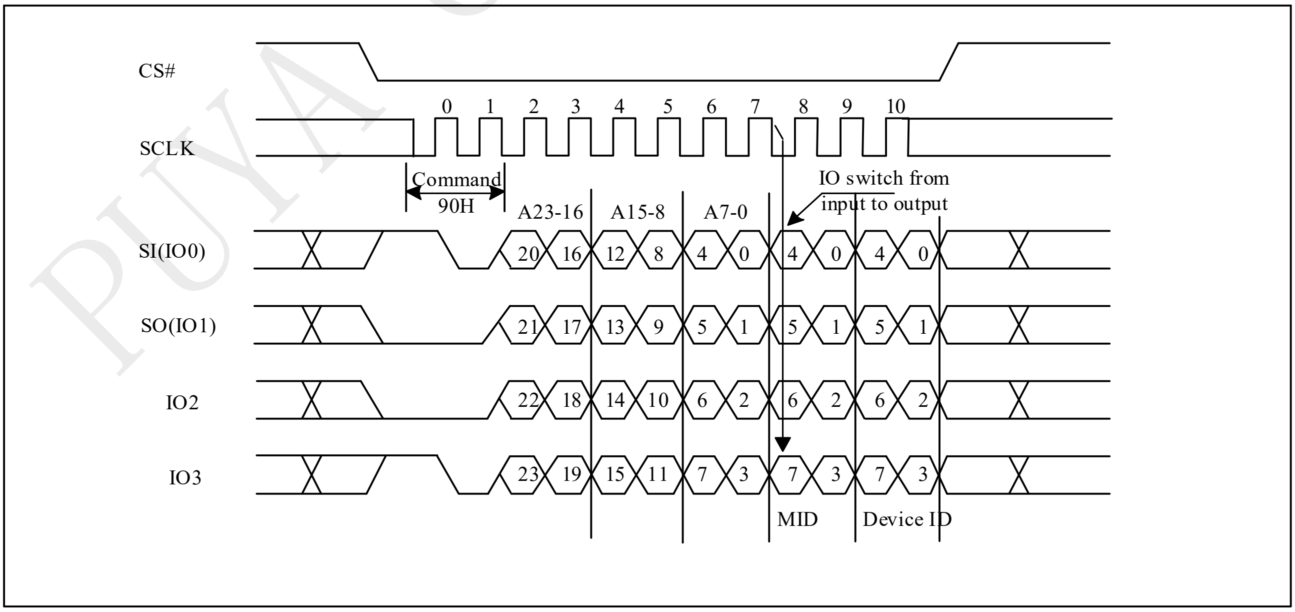


Figure 9-37a Read Electronic Manufacturer & Device ID (REMS) Sequence (QPI)

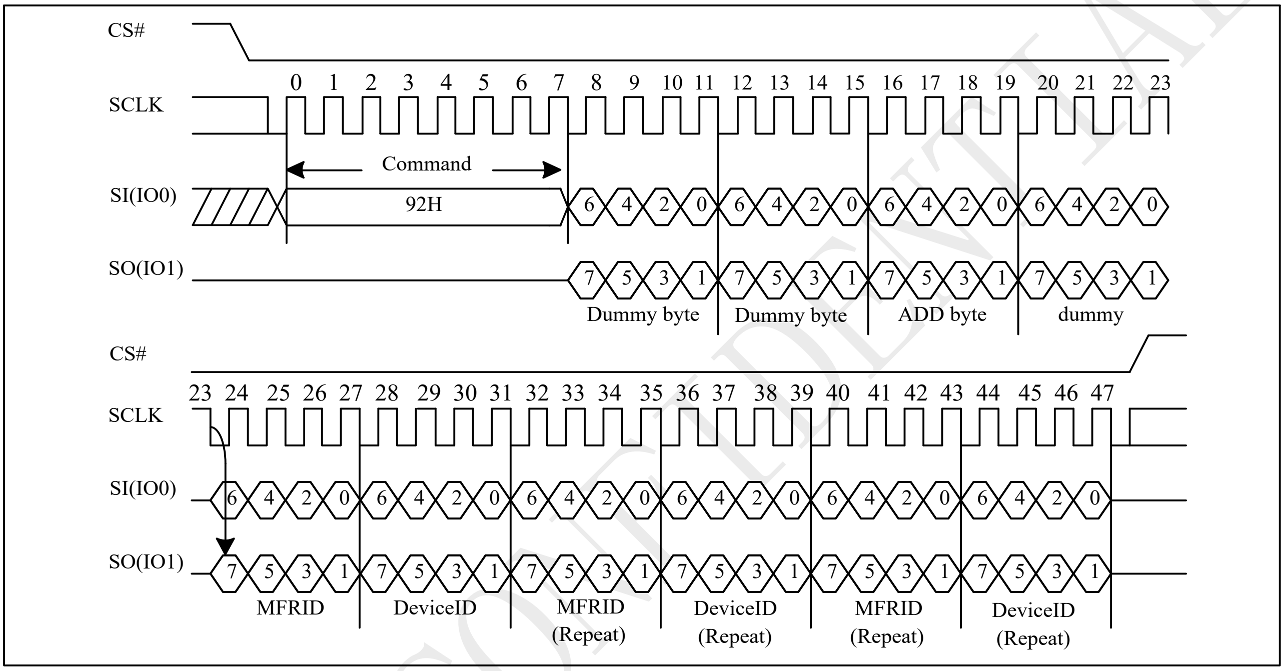


9.38 Dual I/O Read Electronic Manufacturer ID & Device ID (DREMS) (92H)

The DREMS instruction is similar to the REMS command and returns the JEDEC assigned manufacturer ID which takes two pins: SIO0, SIO1 as address input and ID output I/O

The instruction is initiated by driving the CS# pin low and shift the instruction code "92h" followed by two dummy bytes and one-byte address (A7~A0). After which, the Manufacturer ID for PUYA (85h) and the Device ID are shifted out on the falling edge of SCLK with most significant bit (MSB) first. If the one-byte address is initially set to 01h, then the device ID will be read first and then followed by the Manufacturer ID. The Manufacturer and Device ID scan be read continuously, alternating from one to the other. The instruction is completed by driving CS# high.

Figure 9-38 DUAL I/O Read Electronic Manufacturer & Device ID (DREMS) Sequence (Command 92h)

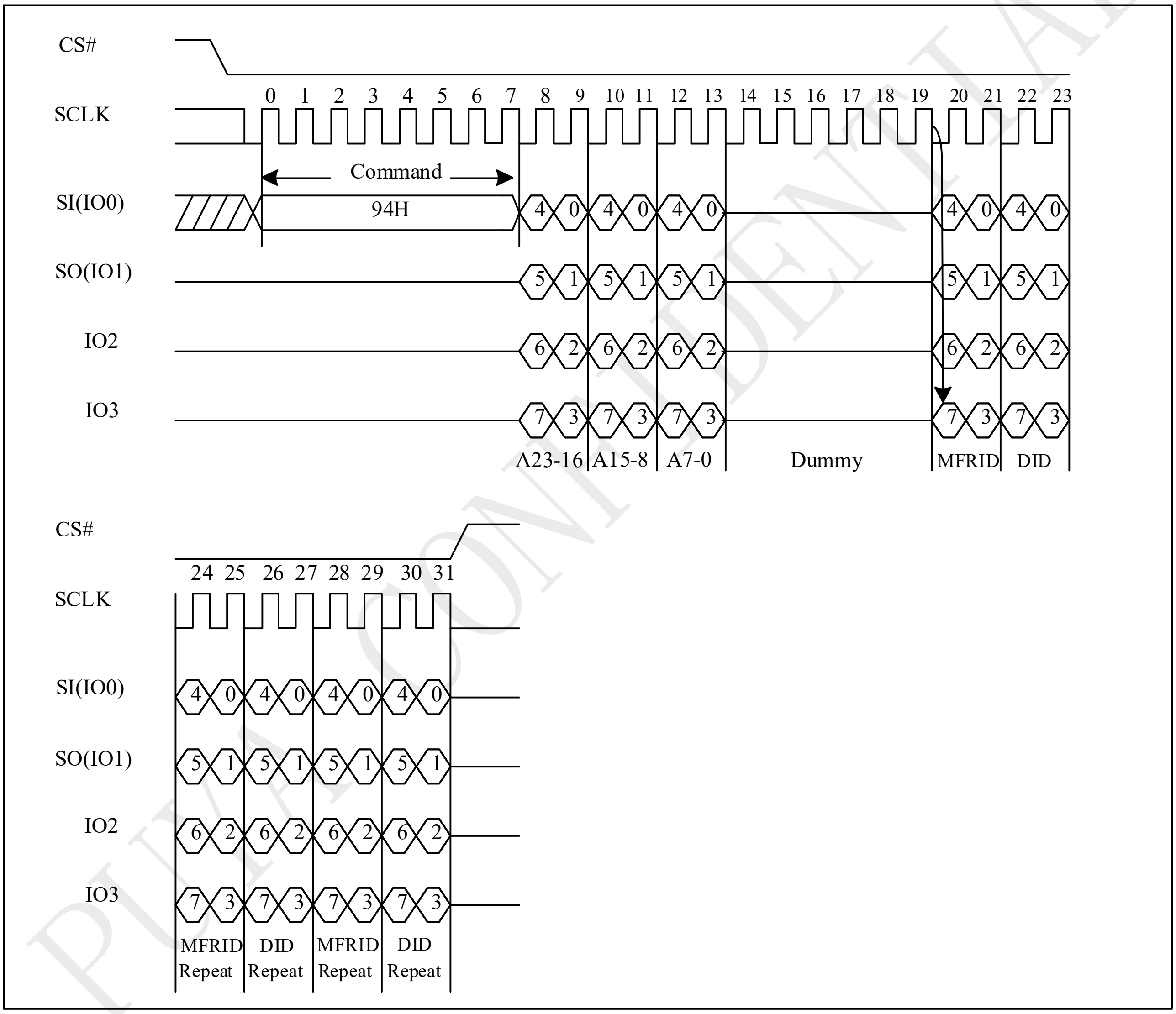


9.39 Quad I/O Read Electronic Manufacturer ID & Device ID (QREMS) (94H)

The QREMS instruction is similar to the REMS command and returns the JEDEC assigned manufacturer ID which takes four pins: SIO0, SIO1, SIO2, SIO3 as address input and ID output I/O.

The instruction is initiated by driving the CS# pin low and shift the instruction code "94h" followed by two dummy bytes and one-byte address (A7~A0). After which, the Manufacturer ID for PUYA (85h) and the Device ID are shifted out on the falling edge of SCLK with most significant bit (MSB)first. If the one-byte address is initially set to01h, then the device ID will be read first and then followed by the Manufacturer ID. The Manufacturer and Device ID scan be read continuously, alternating from one to the other. The instruction is completed by driving CS# high.

Figure 9-39 QUAD I/O Read Electronic Manufacturer & Device ID (QREMS) Sequence (Command 94h)



9.40 Read Identification (RDID) (9FH)

The RDID instruction is for reading the manufacturer ID of 1-byte and followed by Device ID of 2-byte. The PUYA Manufacturer ID and Device ID are list as "Table ID Definitions”.

The sequence of issuing RDID instruction is: CS# goes low→ sending RDID instruction code → 24-bits ID data out on SO→ to end RDID operation can use CS# to high at any time during data out. While Program/Erase operation is in progress, it will not decode the RDID instruction, so there's no effect on the cycle of program/erase operation which is currently in progress. When CS# goes high, the device is at standby stage.

Figure 9-40 Read Identification (RDID) Sequence (Command 9Fh)

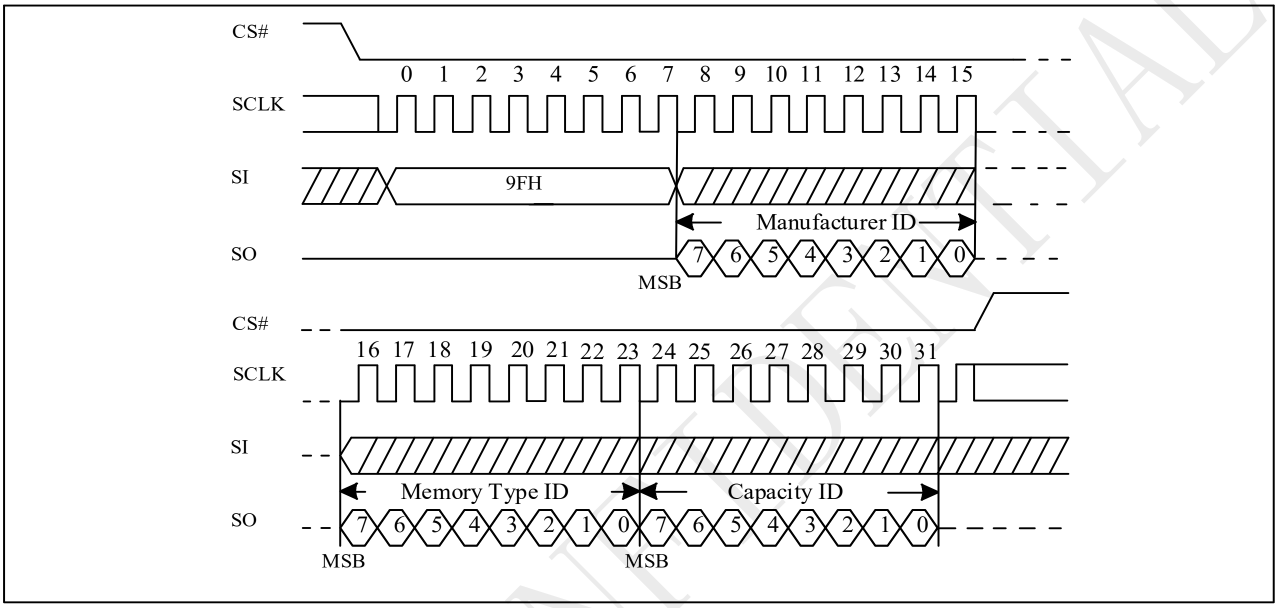


Figure 9-40a Read Identification (RDID) Sequence (QPI)

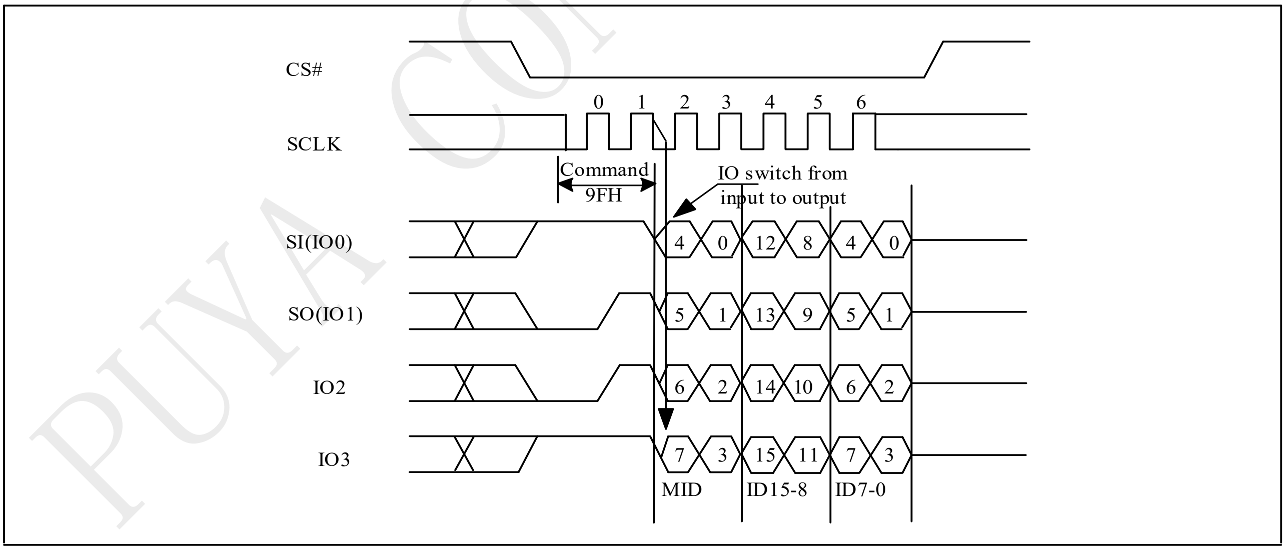


Table ID Definitions

PY25R128LA	RDID command	manufacturer ID	memory type	memory density
		85	63	18
	RES command	electronic ID		
		17		
	REMS command	manufacturer ID		device ID
		85		17

9.41 Program/Erase Suspend/Resume

The Suspend instruction interrupts a Page Program, Sector Erase, or Block Erase operation to allow access to the memory array. After the program or erase operation has entered the suspended state, the memory array can be read except for the page being programmed or the sector or block being erased.

Readable Area of Memory While a Program or Erase Operation is Suspended

Suspended Operation	Readable Region of Memory Array
Page Program	All but the Page being programmed
Sector Erase(4KB)	All but the 4KB Sector being erased
Block Erase(32KB)	All but the 32KB Block being erased
Block Erase(64KB)	All but the 64KB Block being erased

When the Serial NOR Flash receives the Suspend instruction, there is a latency of tPSL or tESL before the Write Enable Latch (WEL) bit clears to “0” and the SUS sets to “1”, after which the device is ready to accept one of the commands listed in "Table Acceptable Commands During Program/Erase Suspend after tPSL/tESL" (e.g. FAST READ). Refer to " AC Characteristics" for tPSL and tESL timings. "Table Acceptable Commands During Suspend (tPSL/tESL not required)" lists the commands for which the tPSL and tESL latencies do not apply. For example, RDSR, RDSCUR, RSTEN, and RST can be issued at any time after the Suspend instruction.

Status Register bit 15 (SUS) can be read to check the suspend status. The SUS (Program/Erase Suspend Bit) sets to “1” when a program or erase operation is suspended. The SUS clears to “0” when the program or erase operation is resumed

Acceptable Commands During Program/Erase Suspend after tPSL/tESL

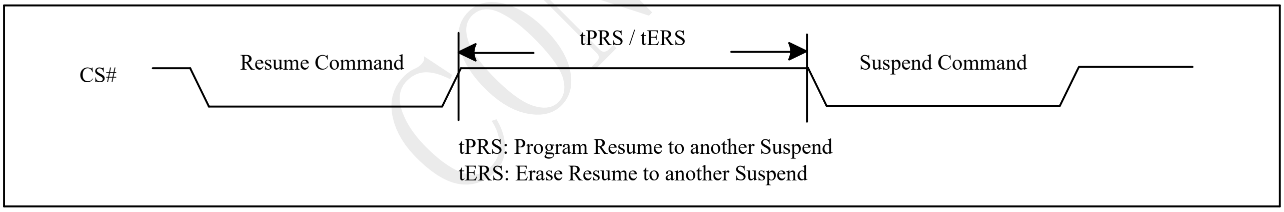
Command name	Command Code	Suspend Type	
		Program Suspend	Erase Suspend
READ	03H	•	•
FAST READ	0BH	•	•
DTRFRD	0DH	•	•
DREAD	3BH	•	•
QREAD	6BH	•	•
2READ	BBH	•	•
2DTRD	BDH	•	•
4READ	EBH	•	•
Word read	E7H	•	•
4DTRD	EDH	•	•
Burst Read with Wrap	0CH	•	•
DTR Burst Read with Wrap	0EH	•	•
QPIEN	38H	•	•
Disable QPI	FFH	•	•
RDSFDP	5AH	•	•
RDID	9FH	•	•
REMS	90H	•	•
DREMS	92H	•	•
QREMS	94H	•	•
SBL	77H	•	•
Set Read Parameter	C0H	•	•
WREN	06H		•

Command name	Command Code	Suspend Type	
		Program Suspend	Erase Suspend
RESUME	7AH	•	•
PP	02H		•
QPP	32H		•
Erase Security Registers	44H		
Program Security Registers	42H		•
read Security Registers	48H	•	•
Individual Block Lock	36H		
Individual Block Unlock	39H		•
Read Block Lock Status	3DH	•	•
Global Block Lock	7EH		
Global Block Unlock	98H		•
WRDI	04H	•	•

Acceptable Commands During Suspend（tPSL/tESL not required）

Command name	Command Code	Suspend Type	
		Program Suspend	Erase Suspend
RDSR	05H	•	•
RDSR1	35H	•	•
RDCR	15H	•	•
RES	ABH	•	•
RSTEN	66H	•	•
RST	99H	•	•
NOP	00H	•	•

Figure 9-41 Resume to Suspend Latency


CS#
Resume Command
tPRS / tERS
Suspend Command
tPRS: Program Resume to another Suspend
tERS: Erase Resume to another Suspend

9.42 Erase Suspend to Program(75H)

The “Erase Suspend to Program” feature allows Page Programming while an erase operation is suspended. Page Programming is permitted in any unprotected memory except within the sector of a suspended Sector Erase operation or within the block of a suspended Block Erase operation. The Write Enable (WREN) instruction must be issued before any Page Program instruction.

A Page Program operation initiated within a suspended erase cannot itself be suspended and must be allowed to finish before the suspended erase can be resumed. The Status Register can be polled to determine the status of the Page Program operation. The WEL and WIP bits of the Status Register will remain “1” while the Page Program operation is in progress and will both clear to “0” when the Page Program operation completes.

Figure 9-42 Suspend to Read/Program Latency

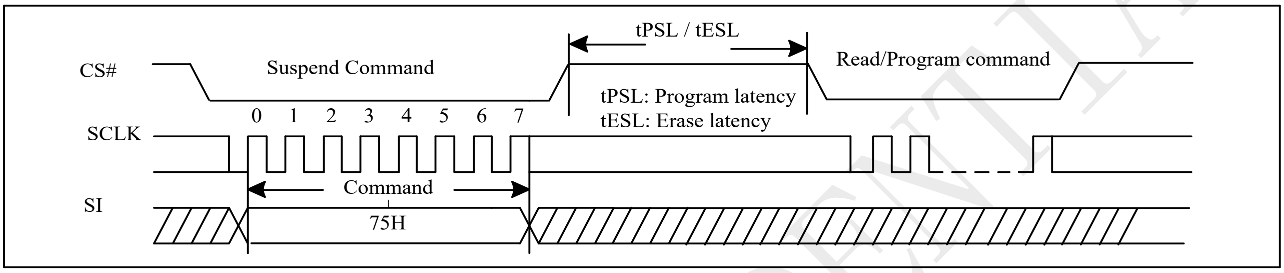
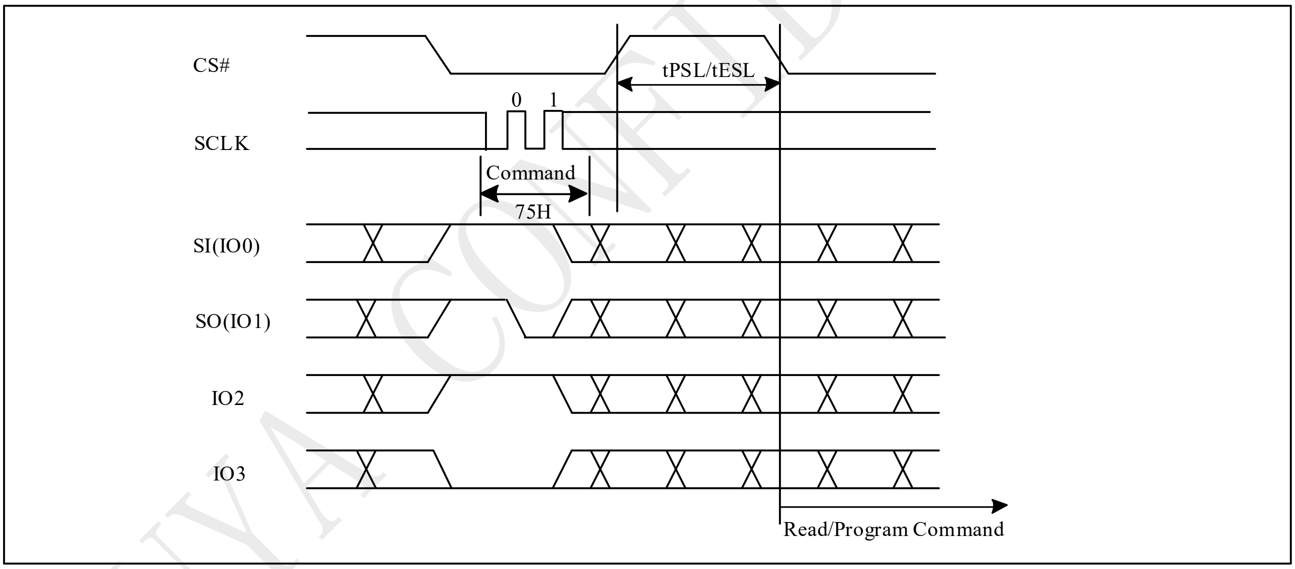


Figure 9-42a Suspend to Read/Program Latency (QPI)



Note:

1. Please note that Program only available after the Erase-Suspend operation
2. To check suspend ready information, please read status register bit15 (SUS).

9.43 Program Resume and Erase Resume(7AH)

The Resume instruction resumes a suspended Page Program, Sector Erase, or Block Erase operation. Before issuing the Resume instruction to restart a suspended erase operation, make sure that there is no Page Program operation in progress.

Immediately after the Serial NOR Flash receives the Resume instruction, the WEL and WIP bits are set to “1” and the SUS is cleared to “0”. The program or erase operation will continue until finished ("Resume to Read Latency") or until another Suspend instruction is received. A resume-to-suspend latency of tPRS or tERS must be observed before issuing another Suspend instruction ("Resume to Suspend Latency").

Figure 9-43 Resume to Read Latency

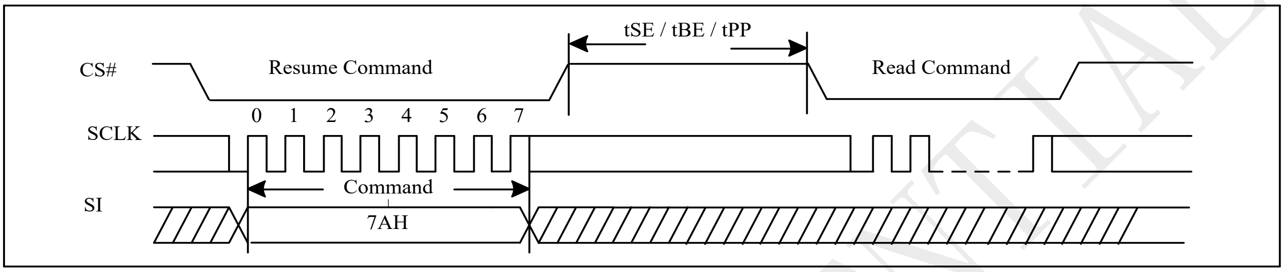
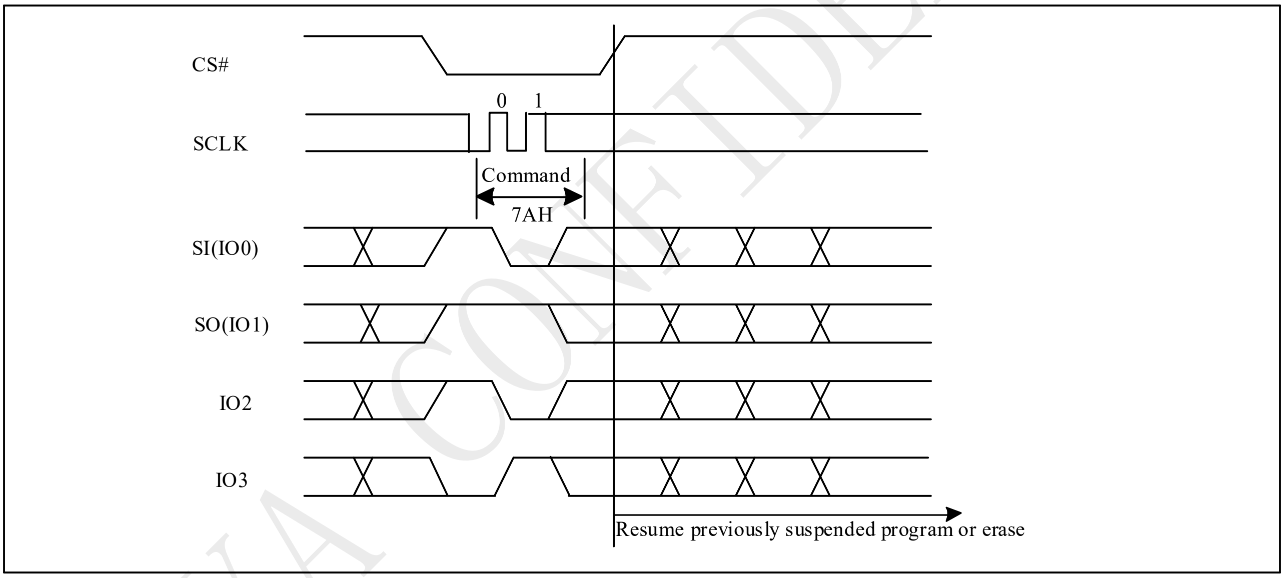


Figure 9-43a Resume to Read Latency (QPI)



9.44 No Operation (NOP)

The "No Operation" command is only able to terminate the Reset Enable (RSTEN) command and will not affect any other command.

The SIO [3:1] are don't care.

9.45 Individual Block Lock (SBLK) (36H)

The Individual Block Lock provides an alternative way to protect the memory array from adverse Erase/Program. In order to use the Individual Block Locks, the WPS bit in Configure Register must be set to 1. If WPS=0, the write protection will be determined by the combination of CMP, BP [4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The SBLK instruction is for write protection a specified block (or sector) of memory, using AMAX-A16 or (AMAX-A12) address bits to assign a 64Kbyte block (or 4K bytes sector) to be protected as read only.

The WREN (Write Enable) instruction is required before issuing SBLK instruction.

The sequence of issuing SBLK instruction is: CS# goes low → send SBLK (36h) instruction →send 3-byte address assign one block (or sector) to be protected on SI pin → CS# goes high. The CS# must go high exactly at the byte boundary, otherwise the instruction will be rejected and not be executed.

Figure 9-45 Individual Block Lock (Command 36H)

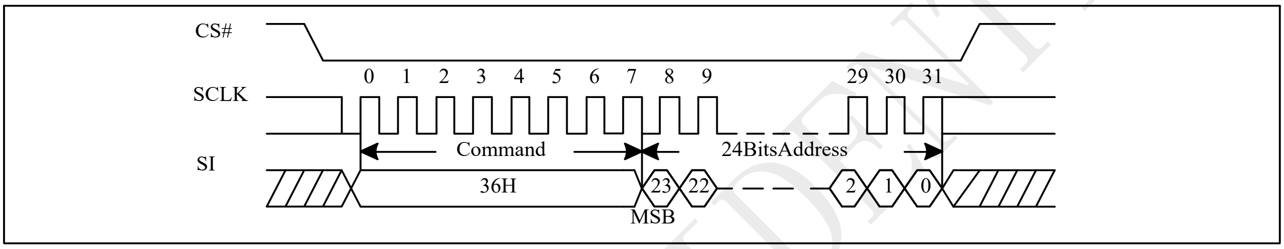
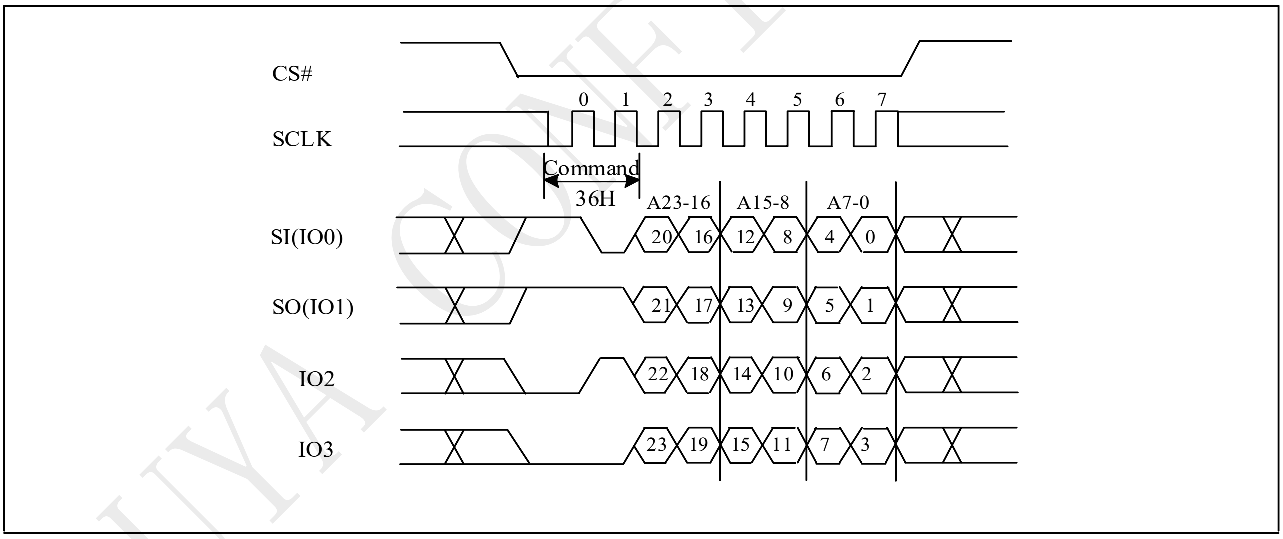


Figure 9-45a Individual Block Lock (QPI)



9.46 Individual Block Unlock (SBULK) (39H)

The Individual Block Lock provides an alternative way to protect the memory array from adverse Erase/Program. In order to use the Individual Block Locks, the WPS bit in Configure Register must be set to 1. If WPS=0, the write protection will be determined by the combination of CMP, BP [4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The SBULK instruction will cancel the block (or sector) write protection state using AMAX-A16 or (AMAX-A12) address bits to assign a 64Kbyte block (or 4K bytes sector) to be unprotected.

The WREN (Write Enable) instruction is required before issuing SBULK instruction.

The sequence of issuing SBULK instruction is: CS# goes low → send SBULK (39h) instruction →send 3-byte address assign one block (or sector) to be protected on SI pin → CS# goes high. The CS# must go high exactly at the byte boundary, otherwise the instruction will be rejected and not be executed.

Figure 9-46 Individual Block Unlock (Command 39H)

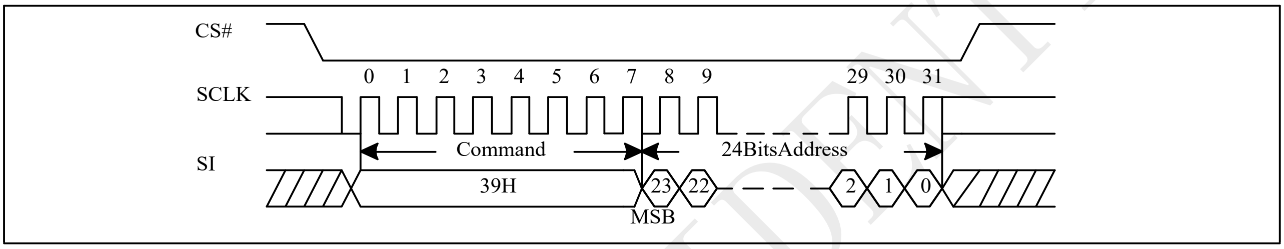
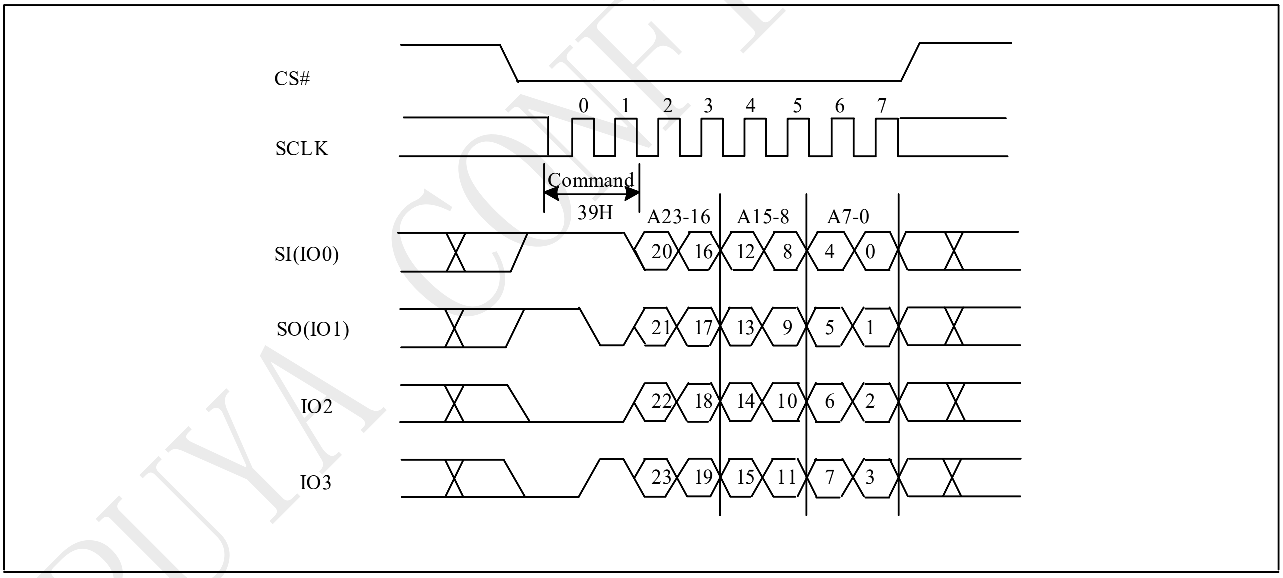


Figure 9-46a Individual Block Unlock (QPI)



9.47 Read Block Lock Status (RDBLK) (3DH)

The Individual Block Lock provides an alternative way to protect the memory array from adverse Erase/Program. In order to use the Individual Block Locks, the WPS bit in Configure Register must be set to 1. If WPS=0, the write protection will be determined by the combination of CMP, BP [4:0] bits in the Status Registers. The Individual Block Lock bits are volatile bits. The default values after device power up or after a Reset are 1, so the entire memory array is being protected.

The RDBLOCK instruction is for reading the status of protection lock of a specified block (or sector), using AMAX-A16 (or AMAX-A12) address bits to assign a 64K bytes block (4K bytes sector) and read protection lock status bit which the first byte of Read-out cycle. The status bit is"1" to indicate that this block has be protected, that user can read only but cannot write/program /erase this block. The status bit is "0" to indicate that this block hasn't be protected, and user can read and write this block.

The sequence of issuing RDBLOCK instruction is: CS# goes low → send RDBLOCK (3Dh) instruction → send 3-byte address to assign one block on SI pin → read block's protection lock status bit on SO pin → CS# goes high.

Both SPI (8 clocks) and QPI (2 clocks) command cycle can accept by this instruction.

Figure 9-47 Read Block Lock Status (Command 3DH)

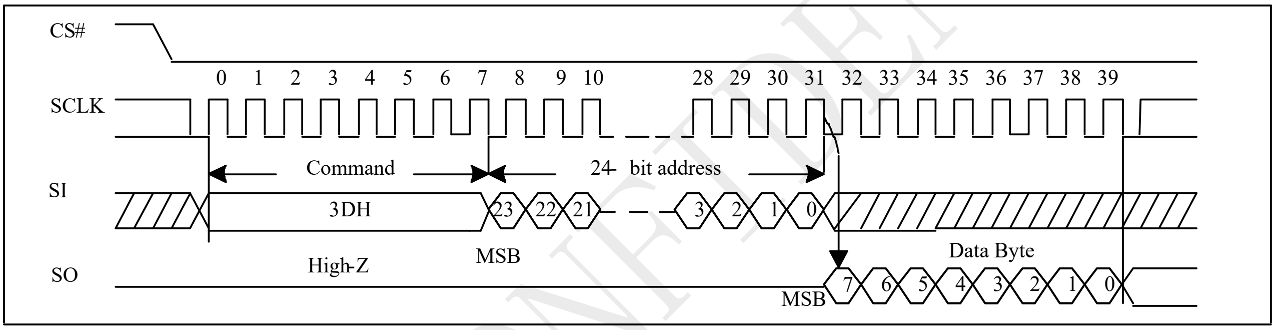
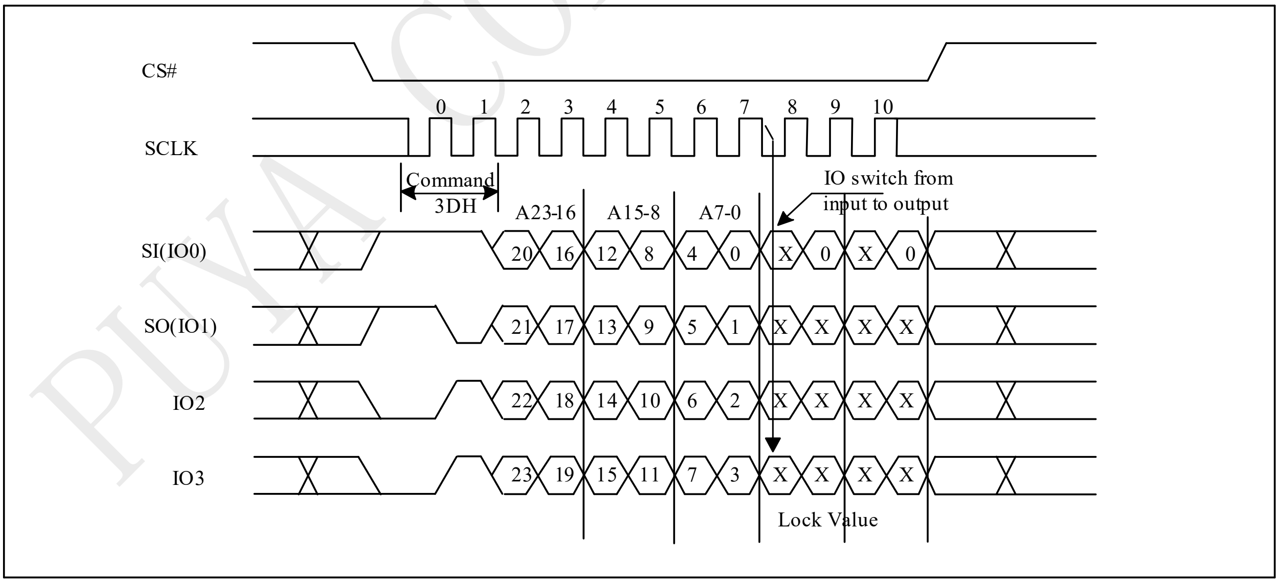


Figure 9-47a Read Block Lock Status (QPI)



9.48 Global Block Lock (GBLK) (7EH)

The GBLK instruction is for enable the lock protection block of the whole chip. The WREN (Write Enable) instruction is required before issuing GBLK instruction.

The sequence of issuing GBLK instruction is: CS# goes low → send GBLK (7Eh) instruction →CS# goes high.

Both SPI (8 clocks) and QPI (2 clocks) command cycle can accept by this instruction. The SIO [3:1] are "don't care" in SPI mode.

The CS# must go high exactly at the byte boundary, otherwise, the instruction will be rejected and not be executed.

Figure 9-48 Global Block Lock (Command 7EH)

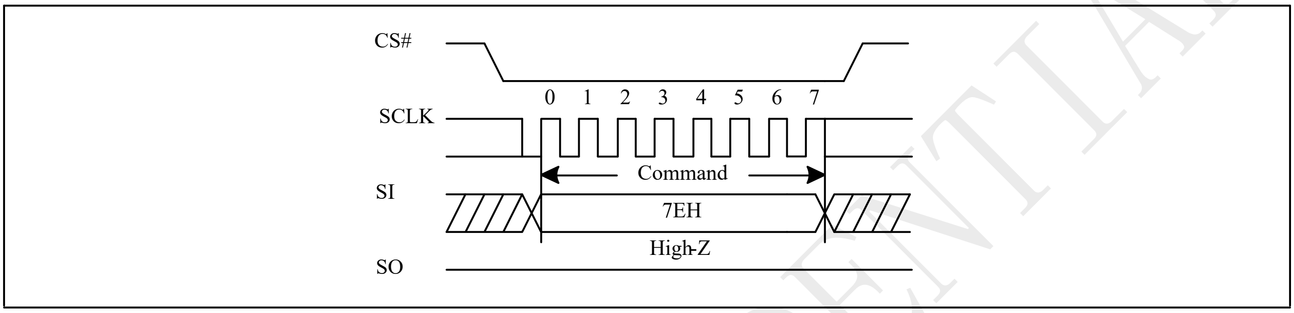
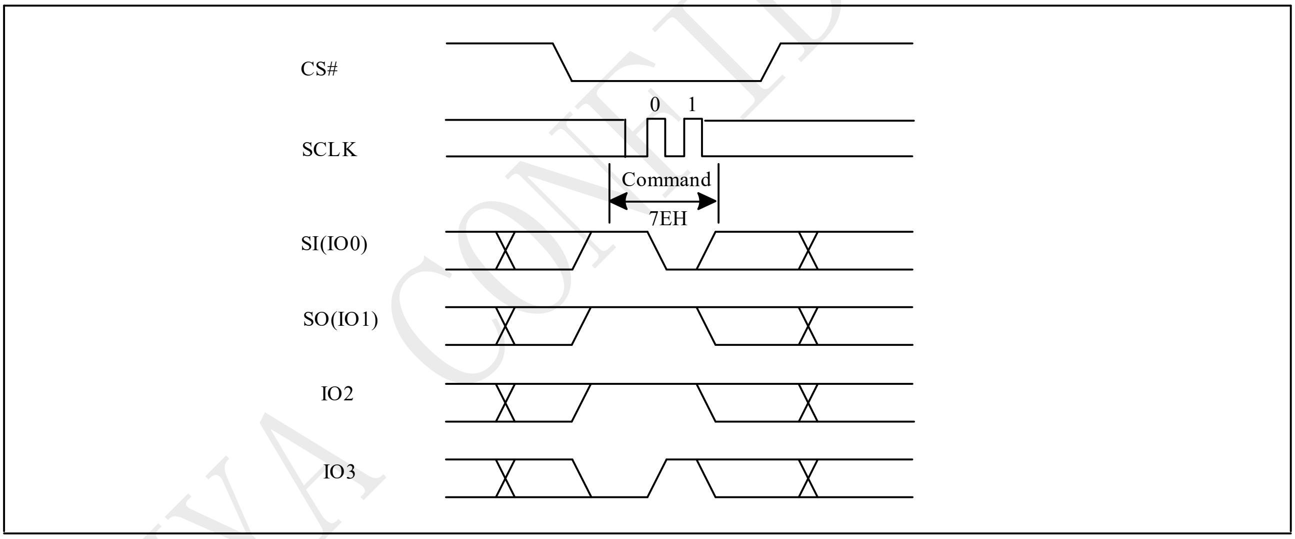


Figure 9-48a Global Block Lock (QPI)



9.49 Global Block Unlock (GBULK) (98H)

The GBULK instruction is for disable the lock protection block of the whole chip. The WREN (Write Enable) instruction is required before issuing GBULK instruction.

The sequence of issuing GBULK instruction is: CS# goes low → send GBULK (98h) instruction →CS# goes high.

Both SPI (8 clocks) and QPI (2 clocks) command cycle can accept by this instruction. The SIO [3:1] are "don't care" in SPI mode.

The CS# must go high exactly at the byte boundary, otherwise, the instruction will be rejected and not be executed.

Figure 9-49 Global Block Unlock (Command 98H)

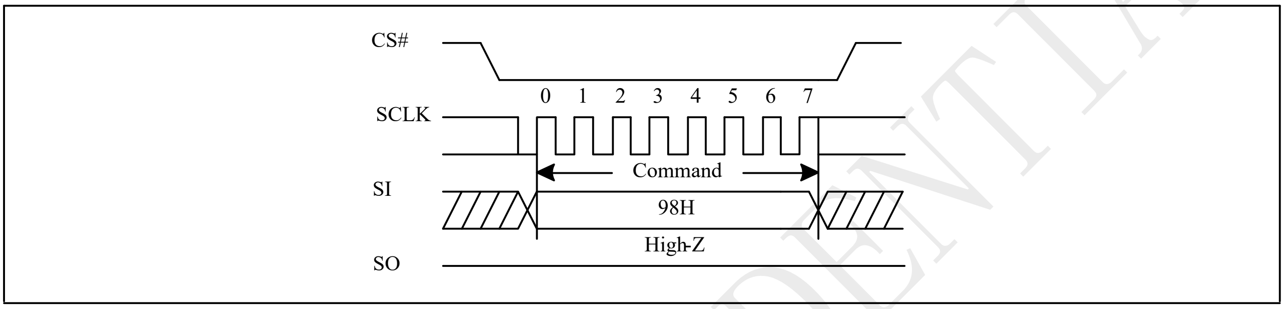
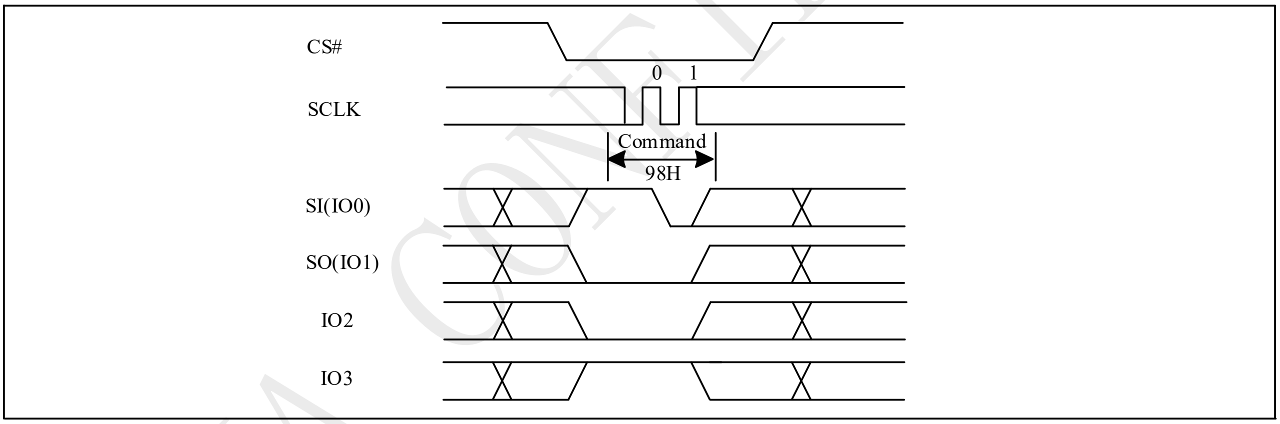


Figure 9-49a Global Block Unlock (QPI)



9.50 Software Reset (RSTEN/RST) (66H/99H)

The Software Reset operation combines two instructions: Reset-Enable (RSTEN) command and Reset (RST) command. It returns the device to a standby mode. All the volatile bits and settings will be cleared then, which makes the device return to the default status as power on, except the EP_FAIL bit.

To execute Reset command (RST), the Reset-Enable (RSTEN) command must be executed first to perform the Reset operation. If there is any other command to interrupt after the Reset-Enable command, the Reset-Enable will be invalid.

The SIO [3:1] are "don't care".

If the Reset command is executed during program or erase operation, the operation will be disabled, the data under processing could be damaged or lost.

Figure 9-50 Software Reset Recovery

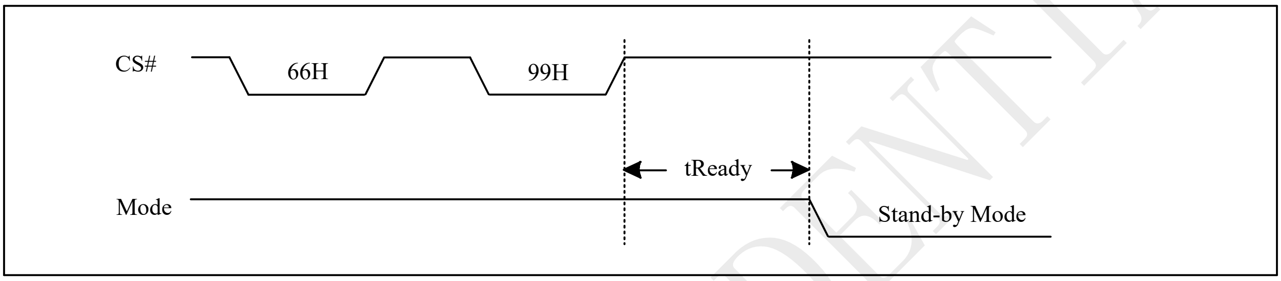


Figure 9-50a Reset Sequence

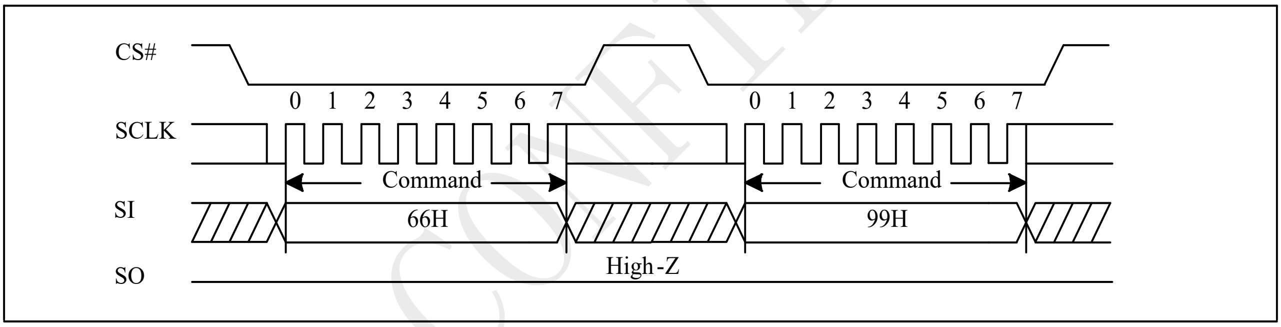
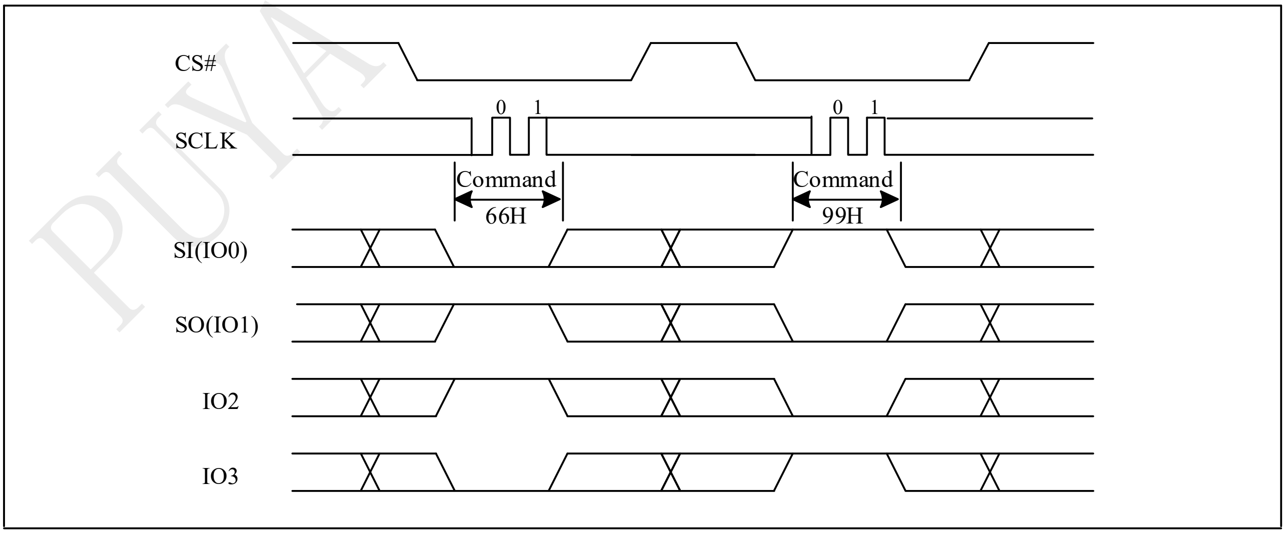


Figure 9-50b Reset Sequence (QPI)



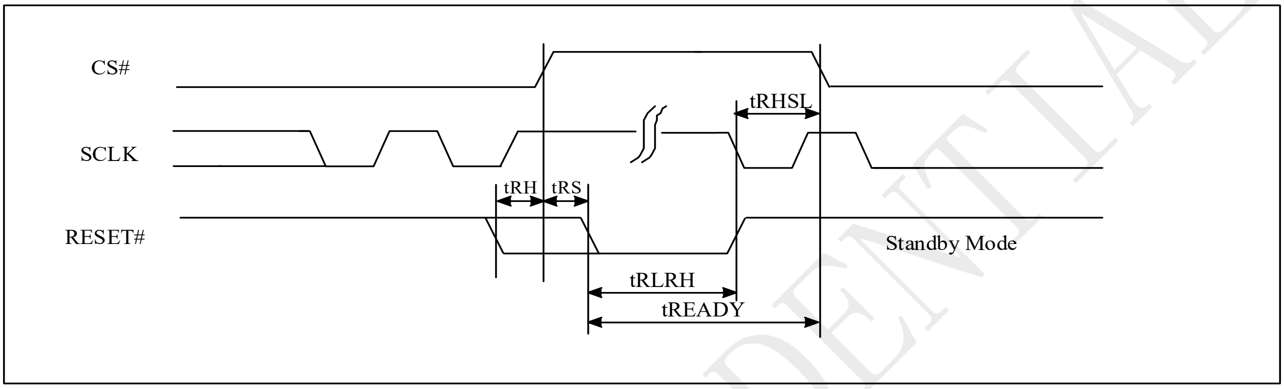
9.51 RESET

Driving the RESET# pin low for a period of tRLRH or longer will reset the device. After reset cycle, the device is at the following states:

- Standby mode
- All the volatile bits such as WEL/WIP/SRAM lock bit will return to the default status as power on.

If the device is under programming or erasing, driving the RESET# pin low will also terminate the operation and data could be lost. During the resetting cycle, the SO data becomes high impedance and the current will be reduced to minimum.

Figure 9-51 RESET Timing



RESET Timing

Symbol	Parameter	Min	Typ	Max	Units
tRHSL	Reset# high before CS# low	1			us
tRS	Reset# setup time	15			ns
tRH	Reset# hold time	15			ns
tRLRH	Reset# low pulse width	1			us
tREADY	Reset Recovery time (except Erase/WRSR/ WRSR1/WRCR)			30	us
	Reset Recovery time (for Erase)		8	12	ms
	Reset Recovery time (for WRSR/ WRSR1/WRCR)		2	12	ms

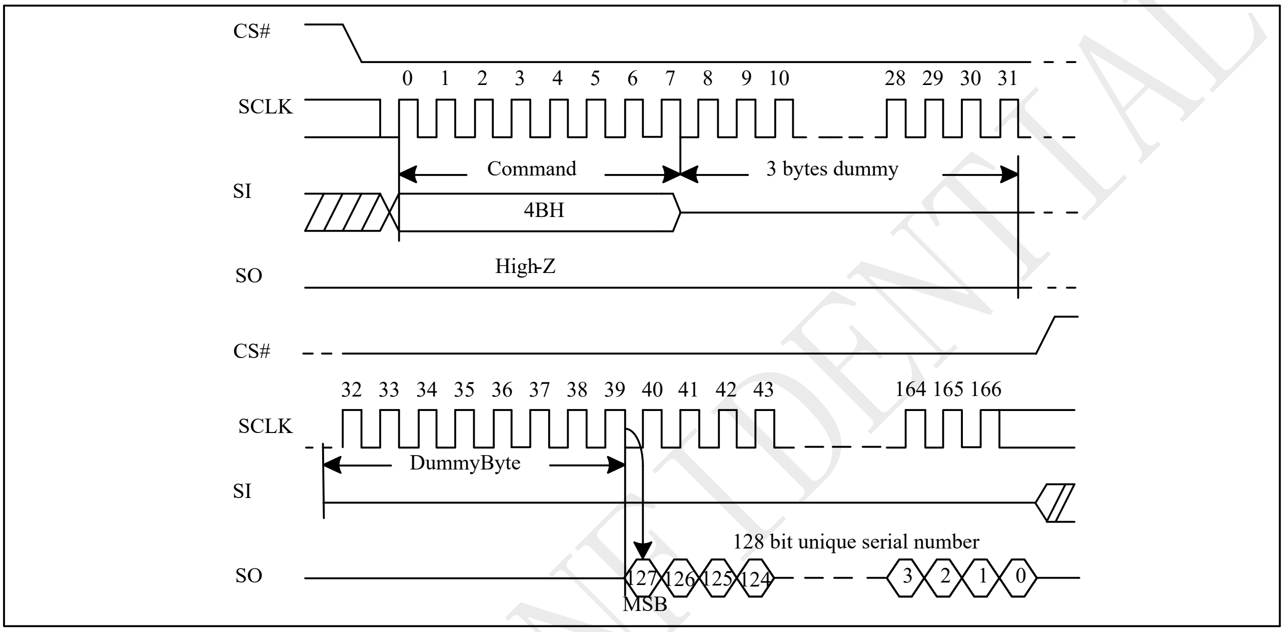
9.52 Read Unique ID(RUID) (4BH)

The Read Unique ID command accesses a factory-set read-only 128bit number that is unique to each PY25Qxx device. The Unique ID can be used in conjunction with user software methods to help prevent copying or cloning of a system.

The Read Unique ID command sequence: CS# goes low →sending Read Unique ID command →Dummy Byte1 →Dummy Byte2 →Dummy Byte3 →Dummy Byte4→128bit Unique ID Out →CS# goes high.

The command sequence is show below.

Figure 9-52 Read Unique ID (RUID) Sequence (Command 4Bh)



9.53 Read SFDP Mode (RDSFDP) (5AH)

The Serial Flash Discoverable Parameter (SFDP) standard provides a consistent method of describing the functional and feature capabilities of serial flash devices in a standard set of internal parameter tables. These parameter tables can be interrogated by host system software to enable adjustments needed to accommodate divergent features from multiple vendors. The concept is similar to the one found in the Introduction of JEDEC Standard, JESD68 on CFI.

The sequence of issuing RDSFDP instruction is same as FREAD: CS# goes low→ send RDSFDP instruction (5Ah) →send 3 address bytes on SI pin→ send 1 dummy byte on SI pin→ read SFDP code on SO→ to end RDSFDP operation can use CS# to high at any time during data out.

Detail SFDP data please contact Puya.

Figure 9-53 Read Serial Flash Discoverable Parameter (RDSFDP) Sequence

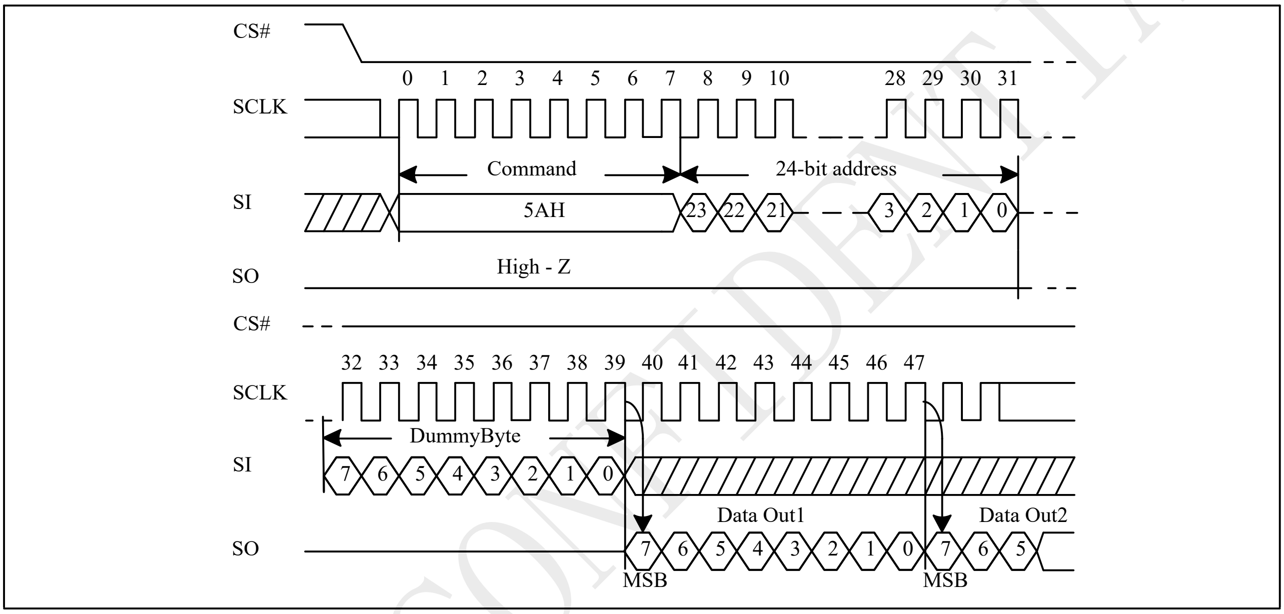
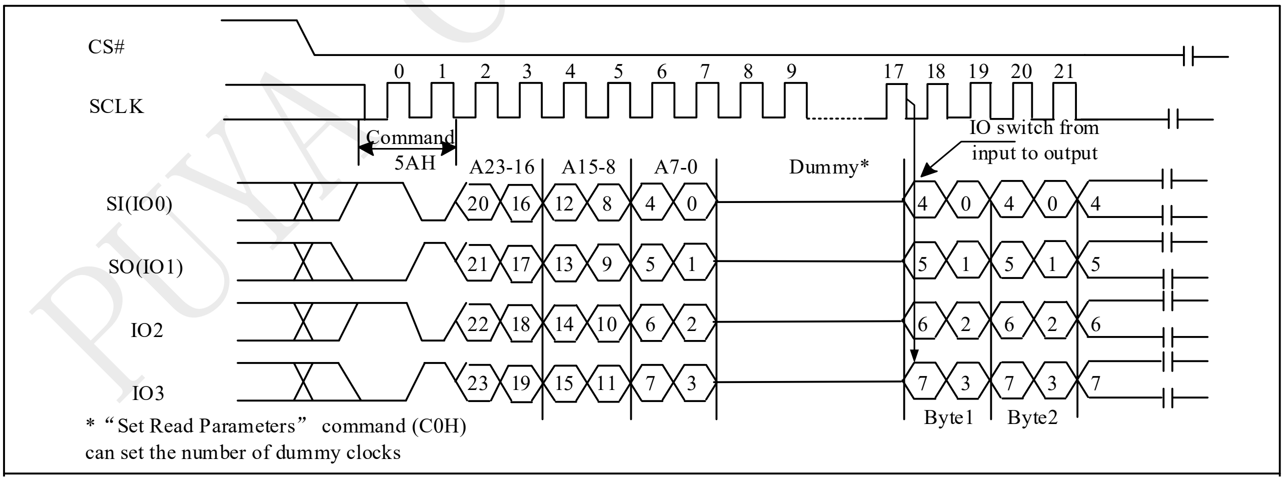


Figure 9-53a Read Serial Flash Discoverable Parameter (RDSFDP) Sequence (QPI)



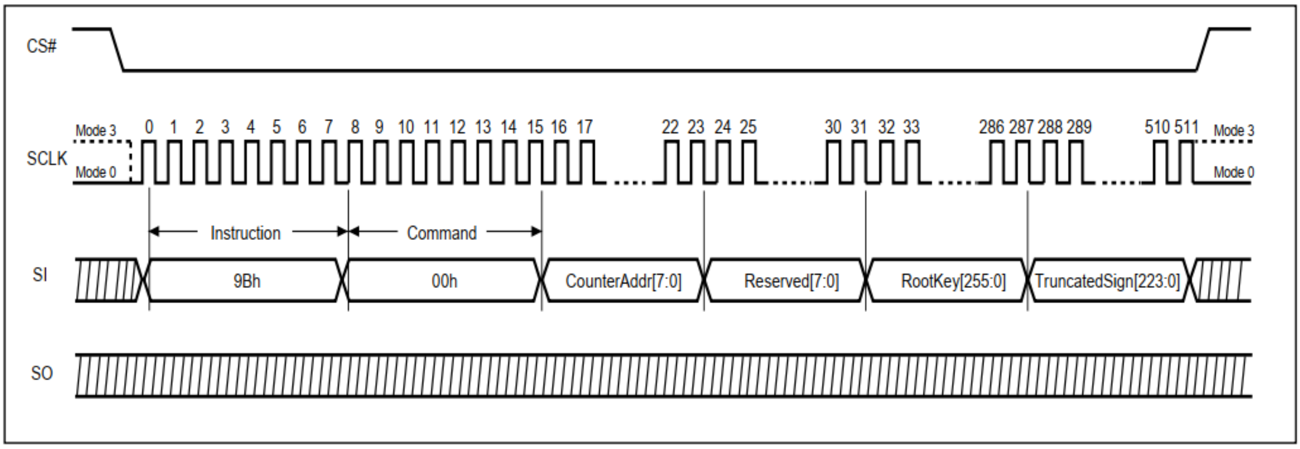
9.54 Write Root Key Register (9Bh + 00h)

The Write Root Key Register instruction is used to program the RPMC root key. The PY25R128HA supports 4 monotonic counters, and each counter has a corresponding root key. Note that the Root Key Register is OTP (One-Time-Programmable).

TruncatedSign[223:0]=HMAC(RootKey[255:0], (9Bh, 00h, CounterAddr[7:0], Reserved[7:0]))

If RootKey[255:0] value is 256'hFF..FFh, RootKey is used as a temporary key, and if the monotonic counter has not been initialized, the corresponding monotonic counter value will be set to 0.

The instruction execution result is reflected to MC Status which can be read with the Read Monotonic Counter Status/Data (96h) instruction.



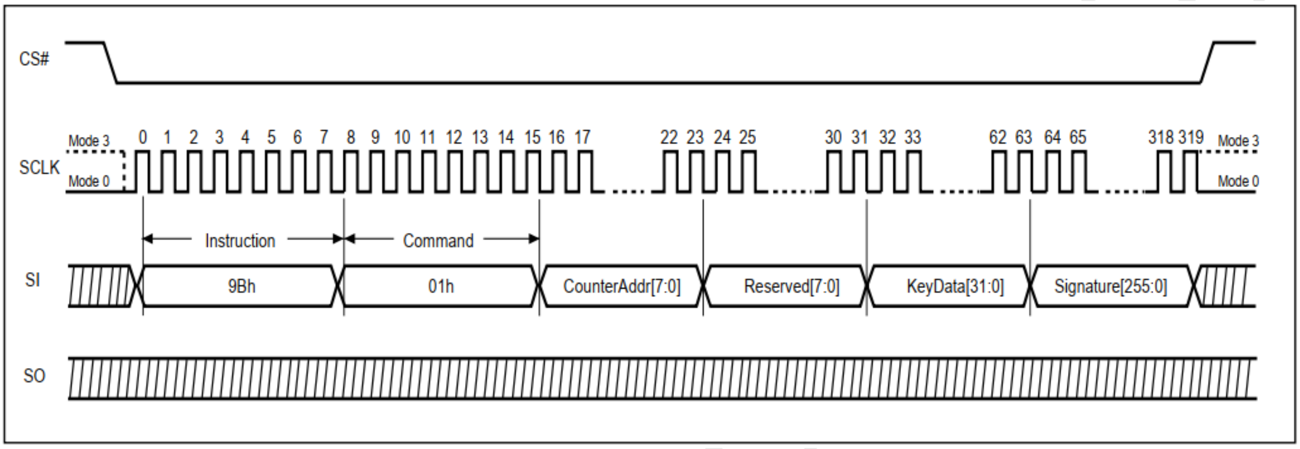
9.55 Update HMAC Key Register (9Bh + 01h)

The Update HMAC Key Register is used to update HMAC Key value. As with the RootKey, there are 4 HMAC Key registers too. Since the HMAC key register is volatile, the HMAC key should be updated after a power cycle.

HMAC Key[CounterAddr][255:0] = HMAC(RootKey[CounterAddr][255:0], KeyData[31:0])

Signature[255:0] = HMAC(HMAC Key[CounterAddr][255:0], (9Bh, 01h, CounterAddr[7:0].Reserved[7:0], KeyData[31:0]))

The instruction execution result is reflected to MC Status, which can be read with the Read Monotonic Counter Status/Data (96h) instruction.



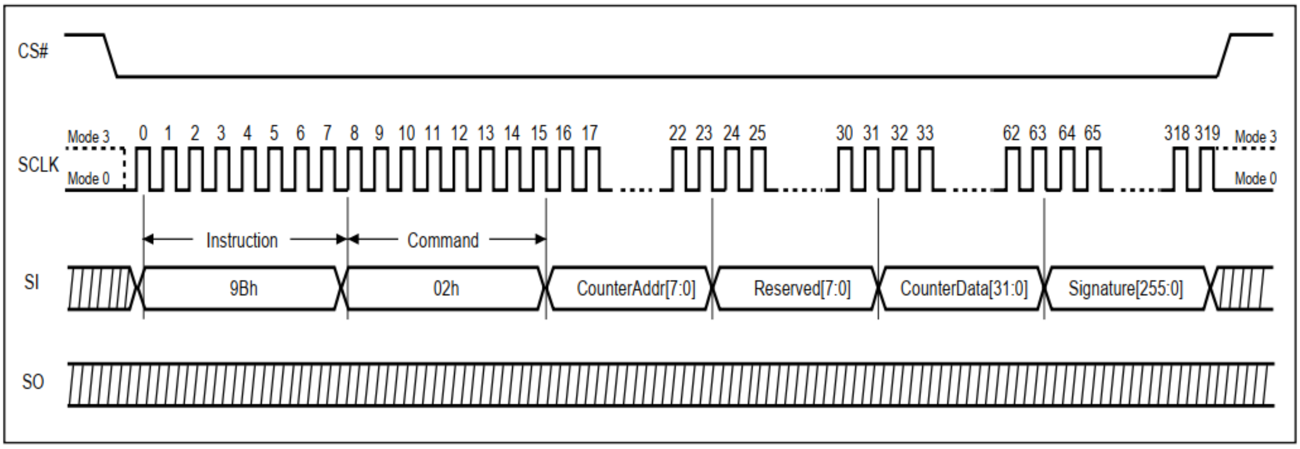
9.56 Increment Monotonic Counter (9Bh + 02h)

The Increment Monotonic Counter instruction is used to increment the monotonic counter by 1.

Signature[255:0]=HMAC(HMAC Key[CounterAddr][255:0], (9Bh, 02h, CounterAddr[7:0]. Reserved[7:0], CounterData[31:0]*))

*CounterData[31:0] is current counter value read from PY25R128HA.

The instruction execution result is reflected to MC Status which can be read with the Read Monotonic Counter Status/Data (96h) instruction.

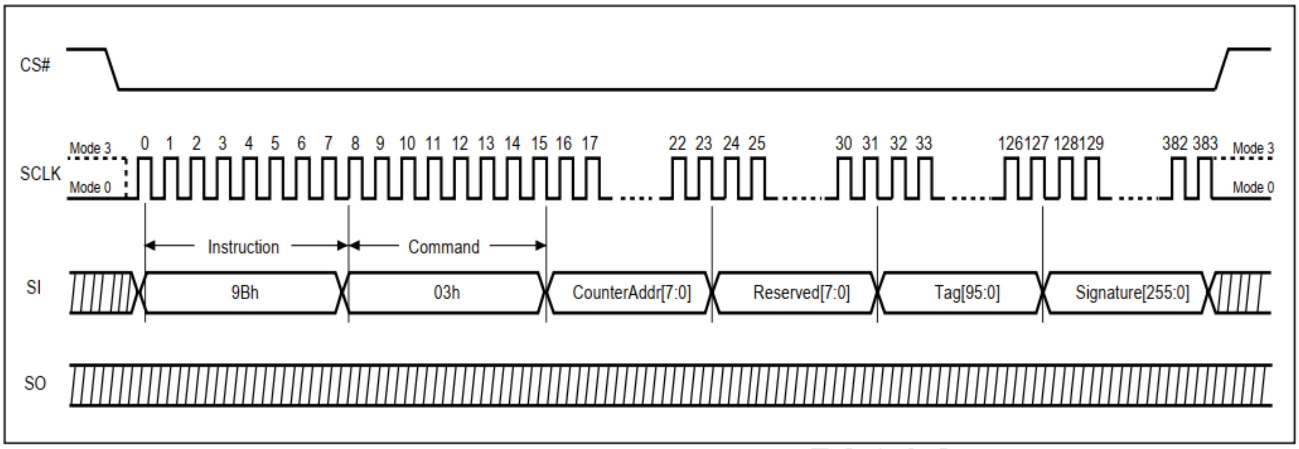


9.57 Request Monotonic Counter (9Bh + 03h)

The Request Monotonic Counter instruction is used to read current counter value from the PY25R128HA.

Signature[255:0]=HMAC(HMAC Key[CounterAddr][255:0], (9Bh, 03h, CounterAddr[7:0]. Reserved[7:0], Tag[95:0]))

The instruction execution result is reflected to MC Status which can be read with the Read Monotonic Counter Status/Data (96h) instruction.



After MC Status[0] BUSY=0, another Read Monotonic Counter Status/Data instruction (96h) should be issued to read out Tag[95:0], Counter Data[31:0] and Signature[255:0].

Signature[255:0]=HMAC(HMAC Key[CounterAddr][255:0], (Tag[95:0], Counter Data[31:0]))

9.58 Reserved Commands (9Bh + 04h~FFh)

The Reserved Commands should not be used.

The instruction execution result is reflected to MC Status which can be read with the Read Monotonic Counter Status/Data (96h) instruction.

9.59 Read Monotonic Counter Status/Data (96h)

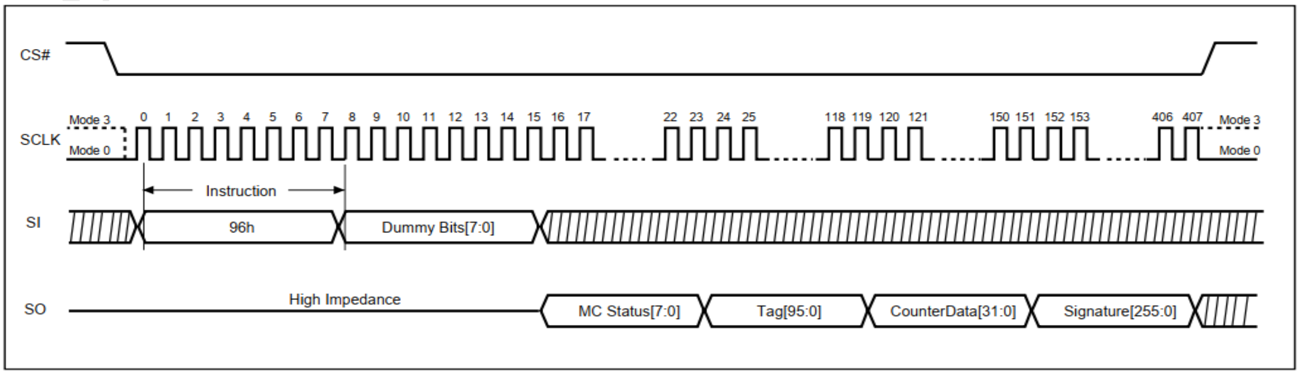
The Read Monotonic Counter Status/Data instruction is used to read the previous OP1 command execution result status or counter data of RPMC.

The status, which is instruction dependent, is listed as below table.

Command Type	MC Status [7:0]	Description
All OP1 common	1000 0000	Successful completion
	0xxx xxx1	Busy
	0xxx x1xx	Reserved command is issued
WRK (9Bh+00h)	0xxx xx1x	1. Root Key Register overwrite; 2. Counter Address is out of range; 3. Truncated signature mismatch.
	0xxx x1xx	Incorrect payload size
UHK (9Bh+01h)	0xxx xx1x	Monotonic Counter is uninitialized
	0xxx x1xx	1. Signature mismatch; 2. Counter Address is out of range; 3. Incorrect payload size
INC_MC (9Bh+02h)	0xxx x1xx	1. Signature mismatch; 2. Counter Address is out of range; 3. Incorrect payload size
	0xxx 1xxx	1. HMAC Key Register is uninitialized; 2. Monotonic Counter is uninitialized
	0xx1 xxxx	Counter Data[31:0] mismatch
Req_MC (9Bh+03h)	0xxx x1xx	1. Signature mismatch; 2. Counter Address is out of range; 3. Incorrect payload size
	0xxx 1xxx	1. HMAC Key Register is uninitialized; 2. Monotonic Counter is uninitialized
Independent status	0000 0000	Power on state
	0x1x xxxx	Fatal Error
	last value	MC status[7:0] will not be updated until 9Bh instructions is received, will stay as last value.

When previous OP1 command is Request Monotonic counters (9Bh + 03h), if SPI Flash returns successful completion (0x80) in the MC Status, then it must also return valid values in the Tag, Counter data and Signature field. If returns error flags, the values returned in Tag, Counter data and Signature field are invalid, the controller should ignore the data.

When MC Status[0] BUSY=1, the device will continuously output the MC status[7:0] until /CS terminates the instruction. It will not output Tag, Counter data and Signature field. Once MC Status[0] BUSY=0, another OP2 command should be issued to read out the status, Tag, Counter data and Signature field.



10 Ordering Information

PY 25 R 128 L A - SU H - I T

Company Designator

PY = Puya Semiconductor

Product Family

25 = SPI interface flash

Product Serial

R = R Serial with RPMC, QE=1 can not be changed

Memory Density

128 = 128Mb

Operation Voltage

L = 1.65V~2.0V

Generation

A = Gen.A

Package Type

SU = SOP8 208mil
WX = WSON8 6x5x0.75mm

SM = SOP16 300mil
WZ = WSON8 8x6x0.75mm

Plating Technology

H:RoHS Compliant , Halogen-free,Antimony - free

Device Grade

I =-40~85C

Packing Type

T = TUBE
R = TAPE & REEL
Y = TRAY

11 Valid Part Numbers and Top Marking

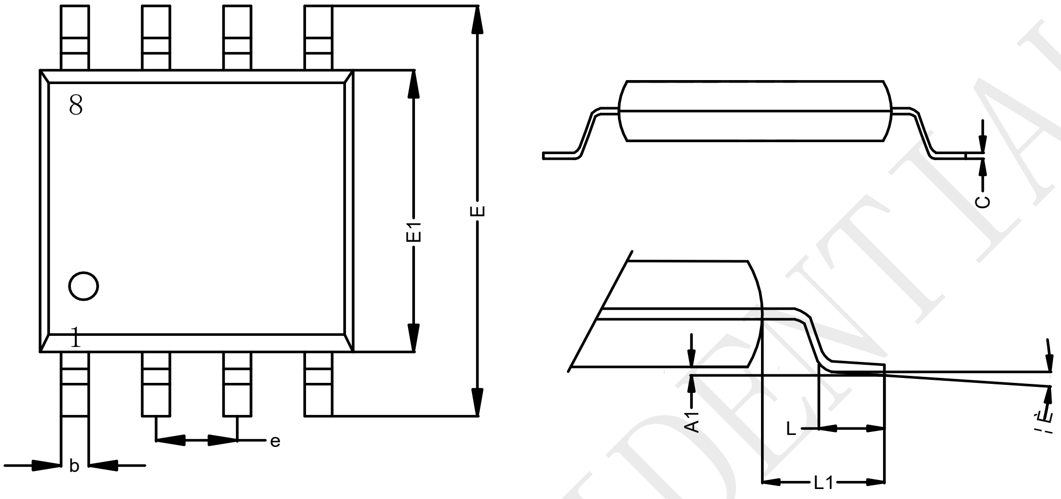
The following table provides the valid part numbers for the PY25R128LA Flash Memory. Please contact PUYA for specific availability by density and package type. PUYA Flash memories use a 17-digit Product Number for ordering.

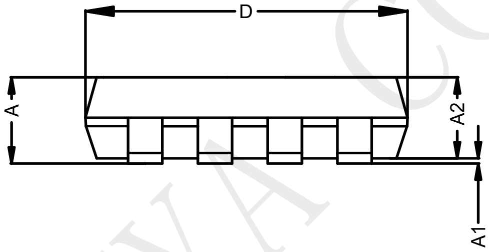
Package Type	Product Number	Density	Top Side Marking	Temp.	Packing Type
SU SOP8 208mil	PY25R128LA-SUH-IT	128M-bit	PY25R128LA xxxxxxx	85C	Tube
SU SOP8 208mil	PY25R128LA-SUH-IR	128M-bit	PY25R128LA xxxxxxx	85C	Reel
SM SOP16 300mil	PY25R128LA-SMH-IT	128M-bit	PY25R128LA xxxxxxx	85C	Tube
SM SOP16 300mil	PY25R128LA-SMH-IR	128M-bit	PY25R128LA xxxxxxx	85C	Reel
WX WSON 6x5mm	PY25R128LA-WXH-IR	128M-bit	PY25R128LA xxxxxxx	85C	Reel
WZ WSON 8x6mm	PY25R128LA-WZH-IR	128M-bit	PY25R128LA xxxxxxx	85C	Reel

Note: The package marked with "*", if necessary, please contact Puya sales.

12 Package Information

12.1 8-Lead SOP (208mil)





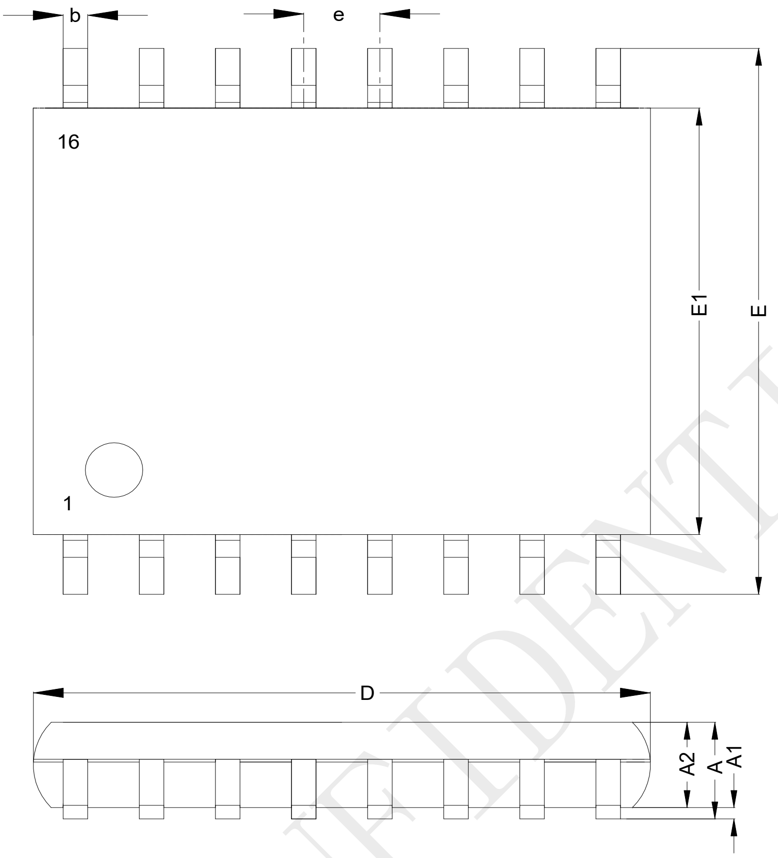
Common Dimensions
(Unit of Measure=millimeters)

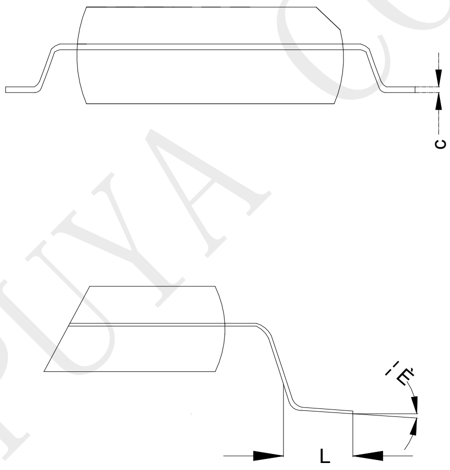
Symbol	Min	Typ	Max
A	-	-	2.150
A1	0.050	-	0.250
A2	1.700	-	1.900
b	0.310	-	0.510
c	0.150	-	0.250
D	5.130	5.230	5.330
E	7.800	7.900	8.000
E1	5.180	5.280	5.380
e	-	1.270	-
L	0.500	-	0.800
L1	-	1.310	-
¦È	0	-	8¡ã

Note：1. Dimensions are not to scale

PUYA	TITLE Puya SOP8(208MIL) POD	DRAWING NO. QRPD-0005	REV 1.0

12.2 16-Lead SOP (300mil)





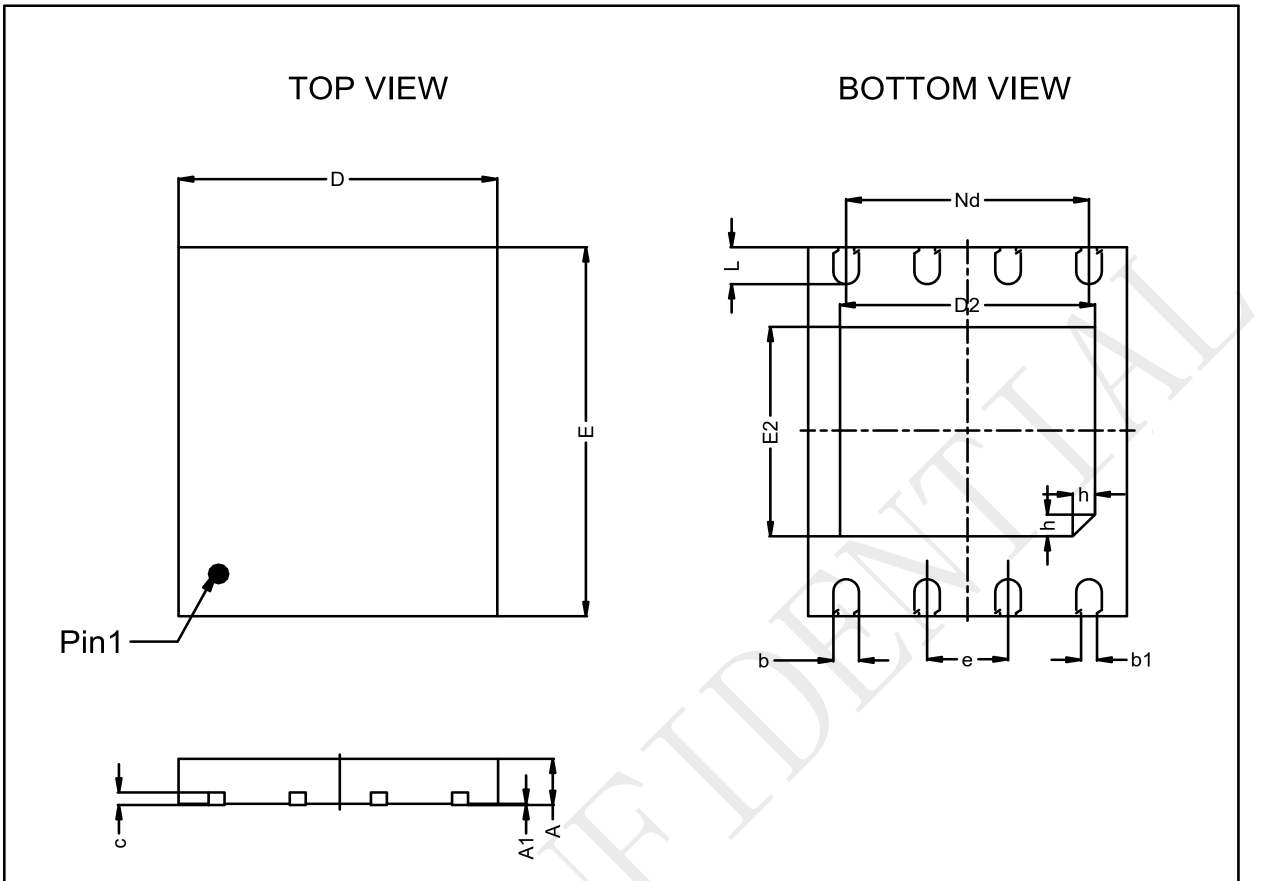
Common Dimensions
(Unit of Measure=millimeters)

Symbol	Min	Typ	Max
A	-	-	2.65
A1	0.10	-	0.30
A2	-	2.30	-
b	0.33	-	0.51
c	0.23	-	0.32
D	10.10	-	10.50
E	10.00	-	10.65
E1	7.40	-	7.60
e	1.27BSC		
L	0.40	-	1.27
¦È	0	-	8¡ã

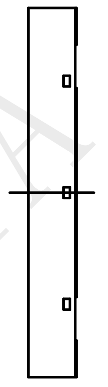
Note£º Dimensions are not to scale.

PUYA	TITLE PUYA SOP16 (300mil) POD	DRAWING NO. QRPD-0055	REV 1.0

12.3 8-Pad WSON(6x5x0.75mm)



SIDE VIEW



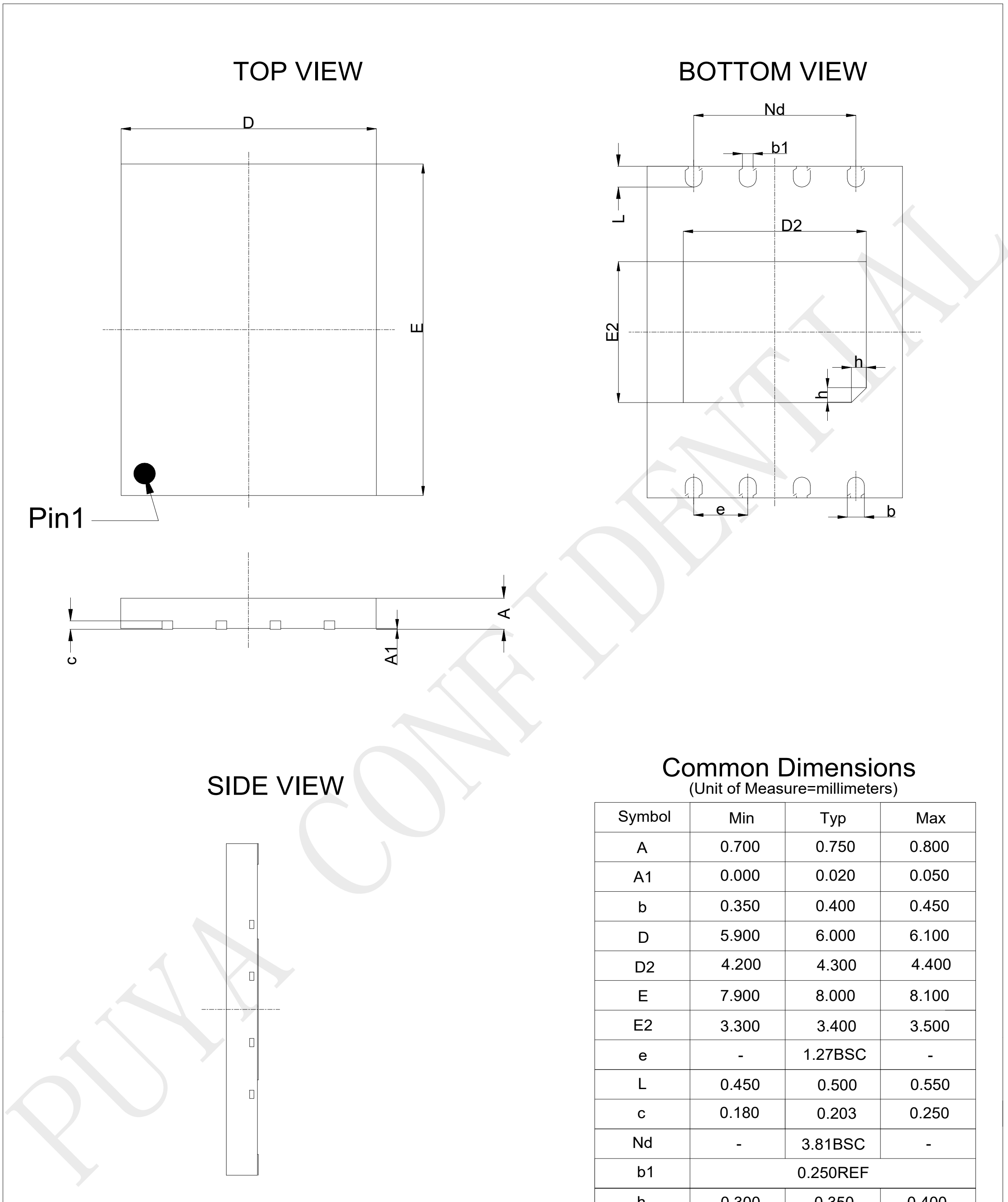
Common Dimensions
(Unit of Measure=millimeters)

Symbol	Min	Typ	Max
A	0.700	0.750	0.800
A1	0.000	0.020	0.050
b	0.350	0.400	0.450
D	4.900	5.000	5.100
D2	3.900	4.000	4.100
E	5.900	6.000	6.100
E2	3.300	3.400	3.500
e	-	1.27BSC	-
L	0.550	0.600	0.650
c	0.180	0.203	0.250
Nd	-	3.81BSC	-
b1	0.250REF		
h	0.300	0.350	0.400

Note：1. Dimensions are not to scale

PUYA	TITLE Puya WSON8L 6x5X0.75-1.27PITCH POD	DRAWING NO. QRPD-0015	REV 1.0

12.4 8-Pad WSON(8x6x0.75mm)



Common Dimensions
(Unit of Measure=millimeters)

Symbol	Min	Typ	Max
A	0.700	0.750	0.800
A1	0.000	0.020	0.050
b	0.350	0.400	0.450
D	5.900	6.000	6.100
D2	4.200	4.300	4.400
E	7.900	8.000	8.100
E2	3.300	3.400	3.500
e	-	1.27BSC	-
L	0.450	0.500	0.550
c	0.180	0.203	0.250
Nd	-	3.81BSC	-
b1	0.250REF		
h	0.300	0.350	0.400

Note：1. Dimensions are not to scale

PUYA	TITLE Puya WSON8L 8X6X0.75-1.27PITCH POD	DRAWING NO. QRPD-0017	REV 1.1

13 Revision History

Rev.	Date	Description	Note
1.0	2024-04-18	Initial release	-
1.1	2024-05-28	Correct the POD of SU(SOP8 208mil)package	-
1.2	2024-12-13	1.correct timing chart typo 2.correct ordering information 3.Delete SFDP table	-

IMPORTANT NOTICE

Puya reserve the right to make changes,corrections,enhancements,modifications to Puya products and/or to this document at any time with -out notice.Purchasers should obtain the latest relevant information of Puya products before placing orders.

Puya products are sold pursuant to terms and conditions of sale in place at the time of order acknowedgement.

Purchasers are solely responsible for the choice and use of Puya products.Puya does not provide service support and assumes no responsi-bility when products that are used on its own or designated third party products.

Puya hereby disclaims any license to any intellectual property rights,express or implied.

Resale of Puya products with provisions inconsistent with the information set forth herein shal void any warranty granted by Puya.

Any with Puya or Puya logo are trademarks of Puya.All other product or service names are the property of their respective owners.

The information in this document supersedes and replaces the information in the previous version.

Puya Semiconductor Co.,Ltd.-All rights reserved