# THE DATASHEET OF

# S25FL064LABNFI041

www.win-source.net 0086-755-83957316

# 64 Mb (8 MB) FL-L flash

## SPI multi-I/O, 3.0 V

## General description

The FL-L family devices are flash non-volatile memory products using:

- Floating gate technology
- 65-nm process lithography

The FL-L family connects to a host system via a serial peripheral interface (SPI). Traditional SPI single bit serial input and output (single I/O or SIO) is supported as well as optional two bit (Dual I/O or DIO) and four bit wide Quad I/O (QIO), and Quad Peripheral Interface (QPI) commands. In addition, there are Double Data Rate (DDR) Read commands for QIO and QPI that transfer address and read data on both edges of the clock.

The architecture features a page programming buffer that allows up to 256 bytes to be programmed in one operation and provides individual 4 KB sector, 32 KB half block sector, 64 KB block sector, or entire chip erase.

By using FL-L family devices at the higher clock rates supported, with Quad commands, the instruction read transfer rate can match or exceed traditional parallel interface, asynchronous, NOR Flash memories, while reducing signal count dramatically.

The FL-L family products offer high densities coupled with the flexibility and fast performance required by a variety of mobile or embedded applications. Provides an ideal storage solution for systems with limited space, signal connections, and power. These memories offer flexibility and performance well beyond ordinary serial flash devices. They are ideal for code shadowing to RAM, executing code directly (XIP), and storing re-programmable data.

## Features

- Serial peripheral interface (SPI) with multi-I/O
  - Clock polarity and phase modes 0 and 3
  - Double data rate (DDR) option
  - Quad peripheral interface (QPI) option
  - Extended addressing: 24- or 32- bit address options
  - Serial command subset and footprint compatible with S25FL-A, S25FL1-K, S25FL-P, S25FL-S, and S25FS-S SPI families
  - Multi I/O command subset and footprint compatible with S25FL-P, S25FL-S and S25FS-S SPI families
- Read
  - Commands: Normal, Fast, Dual I/O, Quad I/O, DualO, QuadO, DDR Quad I/O
  - Modes: Burst wrap, Continuous (XIP), QPI
  - Serial flash discoverable parameters (SFDP) for configuration information
- Program architecture
  - 256-bytes page programming buffer
  - Program suspend and resume
- Erase architecture
  - Uniform 4 KB sector erase
  - Uniform 32 KB half block erase
  - Uniform 64 KB block erase
  - Chip erase
  - Erase suspend and resume
- 100,000 program-erase cycles, minimum
- 20 year data retention, minimum

Features

- Security features
  - Status and Configuration Register protection
  - Four Security Regions of 256-bytes each outside the main flash array
  - Legacy block protection: Block range
  - Individual and region protection
    - Individual block lock: Volatile individual sector/block
    - Pointer region: Non-volatile sector/block range
    - Power supply lock-down, password, or permanent protection of Security Regions 2 and 3 and pointer region
- Technology
  - 65-nm floating gate technology
- Single supply voltage with CMOS I/O
  - 2.7 V to 3.6 V
- Temperature range / grade
  - Industrial (–40°C to +85°C)
  - Industrial Plus (–40°C to +105°C)
  - Automotive, AEC-Q100 grade 3 (–40°C to +85°C)
  - Automotive, AEC-Q100 grade 2 (–40°C to +105°C)
  - Automotive, AEC-Q100 grade 1 (–40°C to +125°C)
- Packages (all Pb-free)
  - 8-lead SOIC 208 mil (SOC008)
  - 16-lead SOIC 300 mil (SO3016)
  - USON 4 × 4 mm (UNF008)
  - WSON 5 × 6 mm (WND008)
  - BGA-24 6 × 8 mm
    - 5 × 5 ball (FAB024) footprint
    - 4 × 6 ball (FAC024) footprint
  - Known good die (KGD) and known tested die

# Performance summary

**Table 1 Maximum read rates SDR**

| Command | Clock rate (MHz) | MBps |
|---|---|---|
| Read | 50 | 6.25 |
| Fast Read | 108 | 13.5 |
| Dual Read | 108 | 27 |
| Quad Read | 108 | 54 |

**Table 2 Maximum read rates DDR**

| Command | Clock rate (MHz) | MBps |
|---|---|---|
| DDR Quad Read | 54 | 54 |

**Table 3 Typical program and erase rates**

| Operation | KBps |
|---|---|
| Page programming | 569 |
| 4 KB sector erase | 61 |
| 32 KB half block erase | 106 |
| 64 KB block erase | 142 |

**Table 4 Typical current consumption**

| Operation | Typical current | Unit |
|---|---|---|
| Read 50 MHz | 10 | mA |
| Fast read 5MHz | 10 | mA |
| Fast read 10 MHz | 10 | mA |
| Fast read 20 MHz | 10 | mA |
| Fast read 50 MHz | 15 | mA |
| Fast read 108 MHz | 20 | mA |
| Quad I/O / QPI read 108 MHz | 20 | mA |
| Quad I/O / QPI DDR read 33 MHz | 15 | mA |
| Quad I/O / QPI DDR read 54 MHz | 17 | mA |
| Program | 17 | mA |
| Erase | 17 | mA |
| Standby SPI | 20 | µA |
| Standby QPI | 35 | µA |
| Deep power down | 2 | µA |

# Table of contents

**General description** ..... **1**
**Features** ..... **1**
**Performance summary** ..... **3**
**Table of contents** ..... **4**
**1 Product overview** ..... **6**
1.1 Migration notes ..... 6
**2 Connection diagrams** ..... **8**
2.1 SOIC 16-lead ..... 8
2.2 8- Connector packages ..... 8
2.3 BGA ball footprint ..... 9
**3 Signal descriptions** ..... **10**
3.1 Serial peripheral interface with multiple input / output (SPI-MIO) ..... 10
3.2 Input/output summary ..... 10
3.3 Multiple input / output (MIO) ..... 11
3.4 Serial Clock (SCK) ..... 11
3.5 Chip Select (CS#) ..... 11
3.6 Serial Input (SI) / IO0 ..... 11
3.7 Serial Output (SO) / IO1 ..... 12
3.8 Write Protect (WP#) / IO2 ..... 12
3.9 IO3 / RESET# ..... 12
3.10 RESET# ..... 13
3.11 Voltage Supply (VCC) ..... 13
3.12 Supply and Signal Ground ($V_{SS}$) ..... 13
3.13 Not Connected (NC) ..... 13
3.14 Reserved for Future Use (RFU) ..... 13
3.15 Do Not Use (DNU) ..... 13
**4 Block diagram** ..... **14**
4.1 System block diagrams ..... 14
**5 Signal protocols** ..... **16**
5.1 SPI clock modes ..... 16
5.2 Command protocol ..... 17
5.3 Interface states ..... 22
5.4 Data protection ..... 26
**6 Address space maps** ..... **27**
6.1 Overview ..... 27
6.2 Flash memory array ..... 27
6.3 ID address space ..... 28
6.4 JEDEC JESD216 serial flash discoverable parameters (SFDP) space ..... 28
6.5 Security Regions address space ..... 28
6.6 Registers ..... 29
**7 Data protection** ..... **49**
7.1 Security Regions ..... 49
7.2 Deep Power Down ..... 49
7.3 Write Enable commands ..... 50
7.4 Write Protect signal ..... 50
7.5 Status Register Protect (SRP1, SRP0) ..... 50
7.6 Array protection ..... 52
7.7 Individual and region protection ..... 57
**8 Commands** ..... **62**
8.1 Command set summary ..... 62
8.2 Identification commands ..... 69

Table of contents

8.3 Register Access commands ..... 73
8.4 Read Memory Array commands ..... 90
8.5 Program Flash Array commands ..... 100
8.6 Erase Flash Array commands ..... 102
8.7 Security Regions Array commands ..... 110
8.8 Individual Block Lock commands ..... 112
8.9 Pointer Region command ..... 117
8.10 Individual and Region Protection (IRP) commands ..... 118
8.11 Reset commands ..... 125
**9 Data integrity ..... 129**
9.1 Erase endurance ..... 129
9.2 Data retention ..... 129
**10 Software interface reference ..... 130**
10.1 JEDEC JESD216B serial flash discoverable parameters ..... 130
10.2 Device ID address map ..... 141
10.3 Initial delivery state ..... 141
**11 Electrical specifications ..... 142**
11.1 Absolute maximum ratings[59] ..... 142
11.2 Latchup characteristics ..... 142
11.3 Thermal resistance ..... 142
11.4 Operating ranges ..... 143
11.5 Power-up and power-down ..... 144
11.6 DC characteristics ..... 146
**12 Timing specifications ..... 150**
12.1 Key to switching waveforms ..... 150
12.2 AC test conditions ..... 150
12.3 Reset ..... 151
12.4 SDR AC characteristics ..... 154
12.5 DDR AC characteristics ..... 157
12.6 Embedded algorithm performance tables ..... 160
**13 Ordering information ..... 161**
13.1 Ordering part number ..... 161
13.2 Valid combinations — Standard ..... 162
13.3 Valid combinations — Automotive grade / AEC-Q100 ..... 162
**14 Physical diagrams ..... 163**
**Revision history ..... 169**

# 1 Product overview

## 1.1 Migration notes

### 1.1.1 Features comparison

The FL064L family is command subset and footprint compatible with prior generation FL-S, FL1-K and FL-P families.

**Table 1 SPI families comparison**

| Parameter | FL-L | FL-L | FL-S | FL1-K | FL-P |
|---|---|---|---|---|---|
| Technology node | 65-nm | 65-nm | 65-nm | 90-nm | 90-nm |
| Architecture | Floating gate | Floating gate | MIRRORBIT™ Eclipse | Floating gate | MIRRORBIT™ |
| Release date | In production | In production | In production | In production | In production |
| Density | 64 Mb | 256 Mb | 128 Mb–1 Gb | 16 Mb–64 Mb | 32 Mb–256 Mb |
| Bus width | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 | x1, x2, x4 |
| Supply voltage | 2.7 V–3.6 V | 2.7 V–3.6 V | 2.7 V–3.6 V / 1.65 V–3.6 V $V_{IO}$ | 2.7 V–3.6 V | 2.7 V–3.6 V |
| Normal Read Speed | 6 MBps (50 MHz) | 6 MBps (50 MHz) | 6 MBps (50 MHz) | 6 MBps (50 MHz) | 5 MBps (40 MHz) |
| Fast Read Speed | 13 MBps (108 MHz) | 16.5 MBps (133 MHz) | 17 MBps (133 MHz) | 13 MBps (108 MHz) | 13 MBps (104 MHz) |
| Dual Read Speed | 26 MBps (108 MHz) | 33 MBps (133 MHz) | 26 MBps (104 MHz) | 26 MBps (108 MHz) | 20 MBps (80 MHz) |
| Quad Read Speed | 52 MBps (108 MHz) | 66 MBps (133 MHz) | 52 MBps (104 MHz) | 52 MBps (108 MHz) | 40 MBps (80 MHz) |
| Quad Read Speed (DDR) | 54 MBps (54 MHz) | 66 MBps (66 MHz) | 80 MBps (80 MHz) | – | – |
| Program buffer size | 256 B | 256B | 256 B / 512 B | 256 B | 256 B |
| Erase sector/block size | 4 KB / 32 KB / 64 KB | 4 KB / 32 KB / 64 KB | 64 KB / 256 KB | 4 KB / 64 KB | 64 KB / 256 KB |
| Parameter sector size | – | – | 4 KB (option) | – | 4 KB |
| Sector / block erase rate (typ.) | 61 KB/s (4 KB)<br>106 KB/s (32 KB)<br>142 KB/s (64 KB) | 80 KB/s (4 KB)<br>168 KB/s (32 KB)<br>237 KB/s (64 KB) | 500 KB/s | 80 KB/s (4 KB)<br>128 KB/s (64 KB) | 130 KB/s |
| Page programming rate (typ.) | 569 KB/s (256 B) | 854 KB/s (256 B) | 1.2 MBps (256 B)<br>1.5 MBps (512 B) | 365 KB/s | 170 KB/s |
| Security Region / OTP | 1024 B | 1024 B | 1024 B | 768 B (3 × 256 B) | 506 B |
| Individual and region protection or advanced sector protection | Yes | Yes | Yes | Yes | No |
| Erase suspend/resume | Yes | Yes | Yes | Yes | No |

**Note**
1. Refer to individual datasheets for further details.

**Table 1 SPI families comparison** *(continued)*

| Parameter | FL-L | FL-L | FL-S | FL1-K | FL-P |
|---|---|---|---|---|---|
| Program suspend/resume | Yes | Yes | Yes | Yes | No |
| Operating temperature | –40°C to +85°C<br>–40°C to +105°C<br>–40°C to +125°C | –40°C to +85°C<br>–40°C to +105°C<br>–40°C to +125°C | –40°C to +85°C<br>–40°C to +105°C | –40°C to +85°C<br>–40°C to +105°C<br>–40°C to +125°C | –40°C to +85°C<br>–40°C to +105°C |

**Note**
1. Refer to individual datasheets for further details.

### 1.1.2 Known differences from prior generations

#### 1.1.2.1 Error reporting

FL-K, FL1-K and FL-P memories either do not have error status bits or do not set them if program or erase is attempted on a protected sector. This product family does have error reporting status bits for program and erase operations. These can be set when there is an internal failure to program or erase, or when there is an attempt to program or erase a protected sector. In these cases the program or erase operation did not complete as requested by the command. The P_ERR or E_ERR bits and the WIP bit will be set to and remain 1 in SR1V. The Clear Status Register command must be sent to clear the errors and return the device to STANDBY state.

#### 1.1.2.2 Status Register Protect 1 bit

The Configuration Register 1 SRP1 bit CR1V[0], locks the state of the Legacy Block Protection bits (SR1NV[5:2] & SR1V[5:2]), CMP_NV (CR1NV[6]) and TBPROT_NV bit (SR1NV[6]), as freeze did in prior generations. In the FS-S and FL-S families the Freeze bit also locks the state of the Configuration Register 1 BPNV_O bit (CR1NV[3]), and the secure silicon region (OTP) area.

#### 1.1.2.3 WRR Single Register Write

In some legacy SPI devices, a Write Registers (WRR) command with only one data byte would update Status Register 1 and clear some bits in Configuration Register 1, including the Quad mode bit. This could result in unintended exit from Quad mode. This product family only updates Status Register 1 when a single data byte is provided. The Configuration Register 1 is not modified in this case.

#### 1.1.2.4 Other legacy commands not supported

- Autoboot related commands
- Bank Address related commands
- Hold# replaced by the Reset#

#### 1.1.2.5 New features

This product family introduces new features to Infineon SPI category memories:

- Security Regions password protection
- IRP individual region protection

# 2 Connection diagrams

## 2.1 SOIC 16-lead

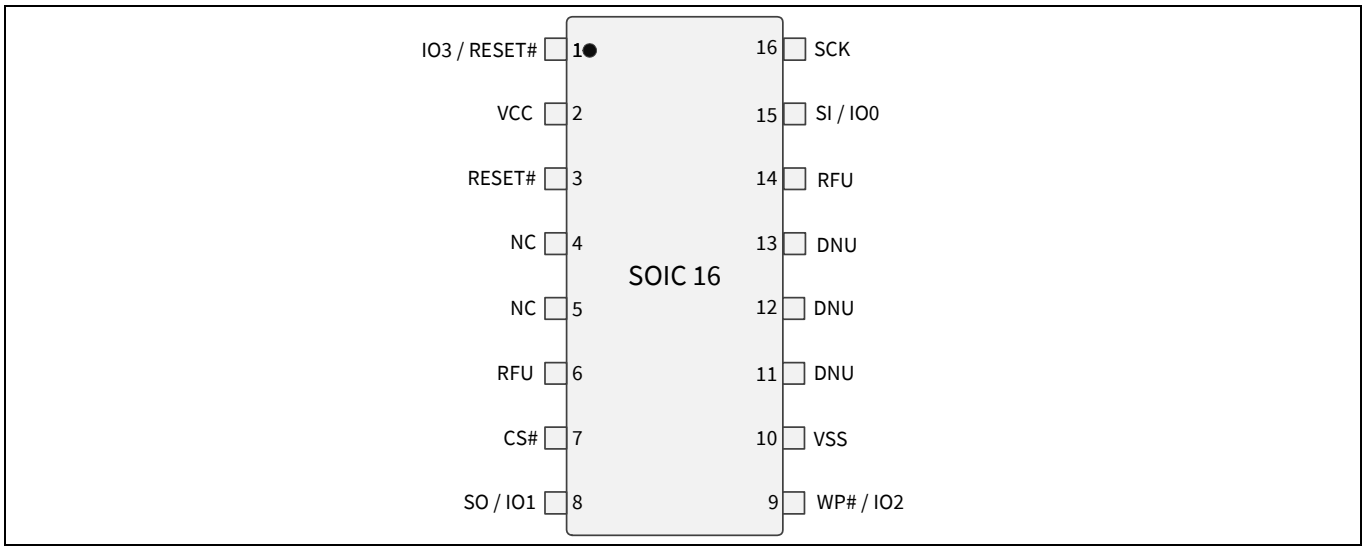



**Figure 1 16-lead SOIC package (SO3016), top view**

## 2.2 8- Connector packages

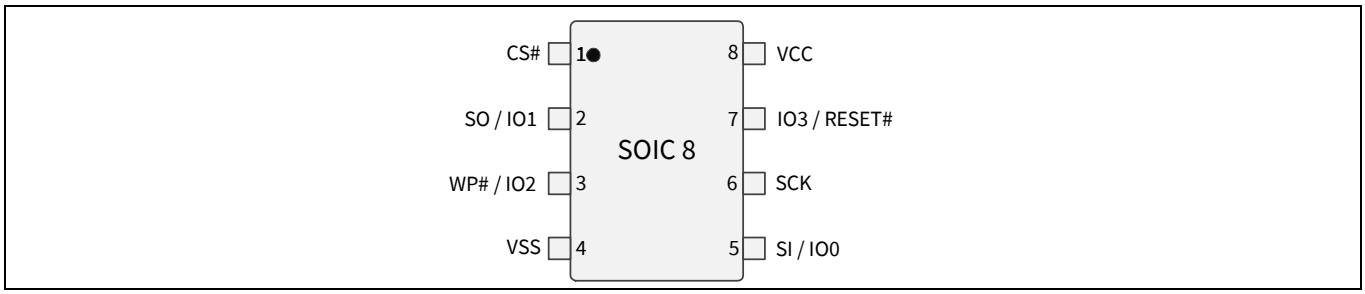



**Figure 2 8-pin plastic small outline package (SOIC8)**

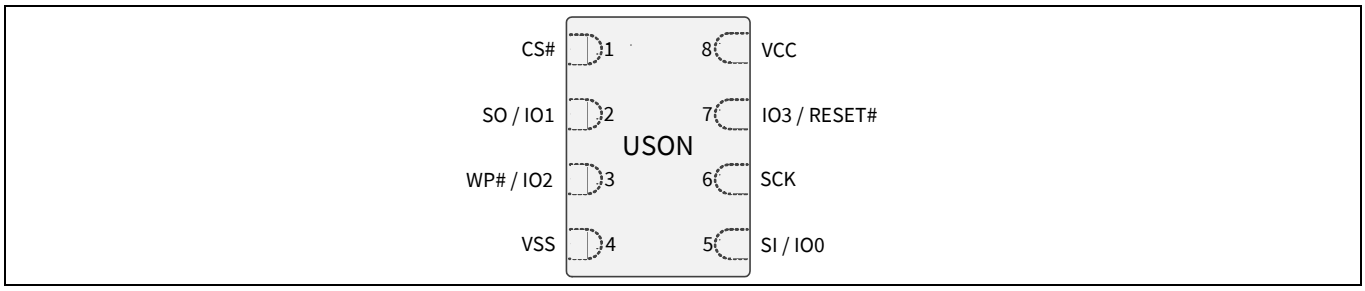



**Figure 3 8-connector package (USON 4 x 4), top view**

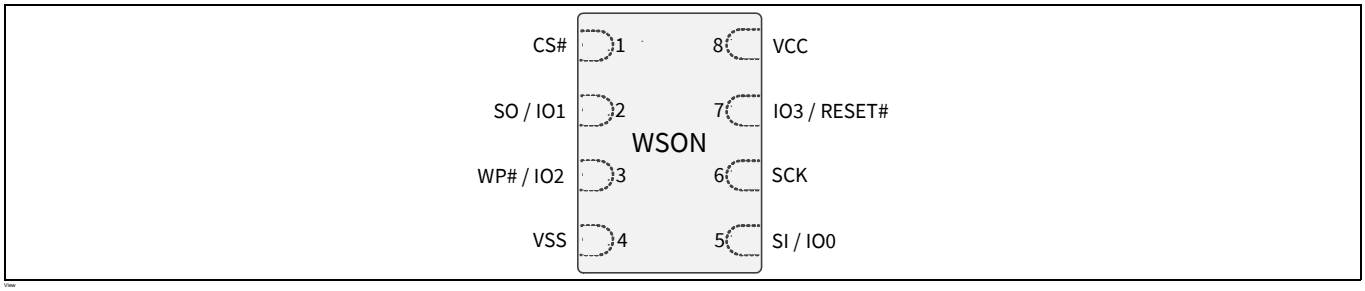



**Figure 4 8-connector package (WSON 5 x 6), top view**[2]

## 2.3 BGA ball footprint

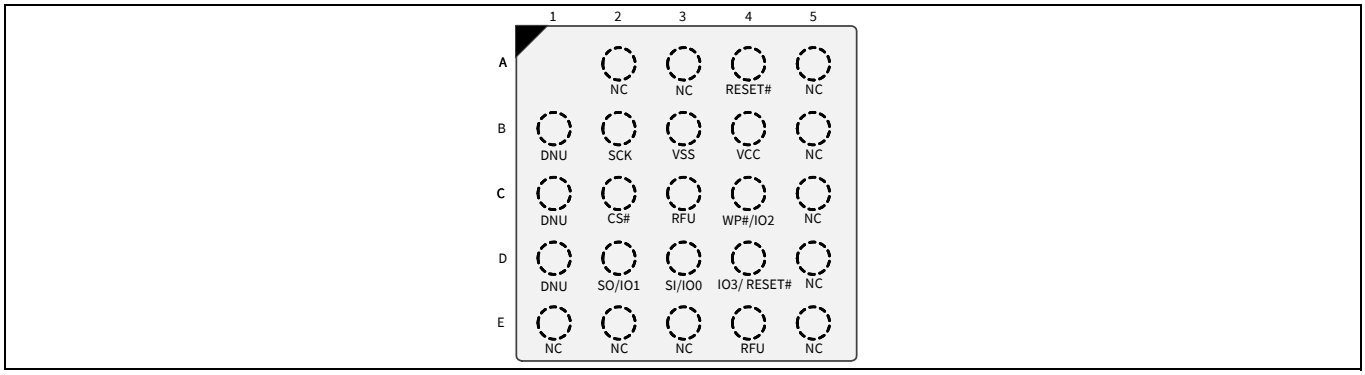



**Figure 5 24-ball BGA, 5 x 5 ball footprint (FAB024), top view**[3, 4]

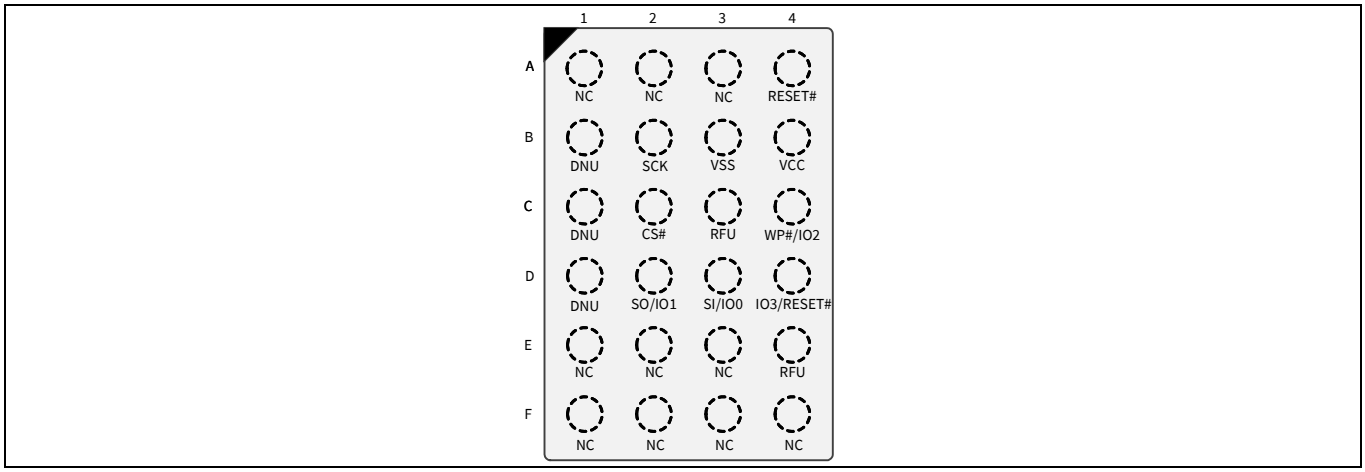



**Figure 6 24-ball BGA, 4 x 6 ball footprint (FAC024), top view**[4]

**Notes**
2. The RESET# input has an internal pull-up and may be left unconnected in the system if Quad mode and hardware reset are not in use.
3. Signal connections are in the same relative positions as FAC024 BGA, allowing a single PCB footprint to use either package.
4. The RESET# input has an internal pull-up and may be left unconnected in the system if Quad mode and hardware reset are not in use.

# 3 Signal descriptions

## 3.1 Serial peripheral interface with multiple input / output (SPI-MIO)

Many memory devices connect to their host system with separate parallel control, address, and data signals that require a large number of signal connections and larger package size. The large number of connections increase power consumption due to so many signals switching and the larger package increases cost.

The S25FL-L family reduces the number of signals for connection to the host system by serially transferring all control, address, and data information over six signals. This reduces the cost of the memory package, reduces signal switching power, and either reduces the host connection count or frees host connectors for use in providing other features.

The S25FL-L family uses the industry standard single bit SPI and also supports optional extension commands for two bit (Dual) and four bit (Quad) wide serial transfers. This multiple width interface is called SPI multi-I/O or SPI-MIO.

## 3.2 Input/output summary

**Table 2 Signal list**

| Signal name | Type | Description |
|---|---|---|
| RESET# | Input | **Hardware Reset**. Low = Device resets and returns to STANDBY state, ready to receive a command. The signal has an internal pull-up resistor and may be left unconnected in the host system if not used. |
| SCK | Input | **Serial Clock** |
| CS# | Input | **Chip Select** |
| SI / IO0 | I/O | **Serial Input** for Single Bit Data commands or IO0 for Dual or Quad commands. |
| SO / IO1 | I/O | **Serial Output** for Single Bit Data commands. IO1 for Dual or Quad commands. |
| WP# / IO2 | I/O | **Write Protect** when not in Quad mode (CR1V[1] = 0 and SR1NV[7] = 1).<br>**IO2** when in Quad mode (CR1V[1] = 1).<br>The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands or write protection. If write protection is enabled by SR1NV[7] = 1 and CR1V[1] = 0, the host system is required to drive WP# HIGH or LOW during a WRR or WRAR command. |
| IO3 / RESET# | I/O | **IO3** in Quad I/O mode, when Configuration Register 1 QUAD bit, CR1V[1] = 1, or in QPI mode, when Configuration Register 2 QPI bit, CR2V[3] = 1 and CS# is LOW.<br>**RESET#** when enabled by CR2V[7] = 1 and not in Quad I/O mode, CR1V[1] = 0, or when enabled in Quad mode, CR1V[1] = 1 and CS# is HIGH.<br>The signal has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad commands or RESET#. |
| $V_{CC}$ | Supply | **Power Supply** |
| $V_{SS}$ | Supply | **Ground** |
| NC | Unused | **Not Connected.** No device internal signal is connected to the package connector nor is there any future plan to use the connector for a signal. The connection may safely be used for routing space for a signal on a printed circuit board (PCB). However, any signal connected to an NC must not have voltage levels higher than $V_{CC}$. |

**Note**

5. Inputs with internal pull-ups or pull-downs drive less than 2 µA. Only during power-up is the current larger at 150 µA for 4 µs. Resistance of pull-ups or pull-down resistors with the typical process at Vcc = 3.3 V at –40°C is ~4.5 MΩ and at 90°C is ~6.6 MΩ.

**Table 2 Signal list** *(continued)*

| Signal name | Type | Description |
|---|---|---|
| RFU | Reserved | **Reserved for Future Use.** No device internal signal is currently connected to the package connector but there is potential future use of the connector for a signal. It is recommended to not use RFU connectors for PCB routing channels so that the PCB may take advantage of future enhanced features in compatible footprint devices. |
| DNU | Reserved | **Do Not Use.** A device internal signal may be connected to the package connector. The connection may be used by Infineon for test or other purposes and is not intended for connection to any host system signal. Any DNU signal related function will be inactive when the signal is at $V_{IL}$. The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to $V_{SS}$. Do not use these connections for PCB signal routing channels. Do not connect any host system signal to this connection. |

**Note**
5. Inputs with internal pull-ups or pull-downs drive less than 2 µA. Only during power-up is the current larger at 150 µA for 4 µs. Resistance of pull-ups or pull-down resistors with the typical process at Vcc = 3.3 V at –40°C is ~4.5 MΩ and at 90°C is ~6.6 MΩ.

## 3.3 Multiple input / output (MIO)

Traditional SPI single bit wide commands (single or SIO) send information from the host to the memory only on the serial input (SI) signal. Data may be sent back to the host serially on the serial output (SO) signal.

Dual or Quad Input / Output (I/O) commands send instructions to the memory only on the SI/IO0 signal. Address or data is sent from the host to the memory as bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as bit pairs on IO0 and IO1 or four bit (nibble) groups on IO0, IO1, IO2, and IO3.

QPI mode transfers all instructions, addresses, and data from the host to the memory as four bit (nibble) groups on IO0, IO1, IO2, and IO3. Data is returned to the host similarly as four bit (nibble) groups on IO0, IO1, IO2, and IO3.

## 3.4 Serial Clock (SCK)

This input signal provides the synchronization reference for the SPI interface. Instructions, addresses, or data input are latched on the rising edge of the SCK signal. Data output changes after the falling edge of SCK, in SDR commands.

## 3.5 Chip Select (CS#)

The Chip Select signal indicates when a command is transferring information to or from the device and the other signals are relevant for the memory device.

When the CS# signal is at the logic HIGH state, the device is not selected and all input signals are ignored and all output signals are high impedance. The device will be in the Standby Power mode, unless an internal embedded operation is in progress. An embedded operation is indicated by the Status Register 1 Write-In-Progress bit (SR1V[0]) set to 1, until the operation is completed. Some example embedded operations are: program, erase, or Write Registers (WRR) operations.

Driving the CS# input to the logic LOW state enables the device, placing it in the Active Power mode. After power-up, a falling edge on CS# is required prior to the start of any command.

## 3.6 Serial Input (SI) / IO0

This input signals used to transfer data serially into the device. It receives instructions, addresses, and data to be programmed. Values are latched on the rising edge of serial SCK clock signal. SI becomes IO0 - an input and output during dual and quad commands for receiving instructions, addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

## 3.7 Serial Output (SO) / IO1

This output signals used to transfer data serially out of the device. Data is shifted out on the falling edge of the serial SCK clock signal. SO becomes IO1 - an input and output during Dual and Quad commands for receiving addresses, and data to be programmed (values latched on rising edge of serial SCK clock signal) as well as shifting out data (on the falling edge of SCK in SDR commands, and on every edge of SCK, in DDR commands).

## 3.8 Write Protect (WP#) / IO2

When WP# is driven Low ($V_{IL}$), when the Status Register Protect 0 (SRP0_NV) or (SRP0) bit of Status Register 1 (SR1NV[7]) or (SR1V[7]) is set to a 1, it is not possible to write to Status Registers, Configuration Registers or DLR registers. In this situation, the command selecting SR1NV, SR1V, CR1NV,CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV is ignored, and no error is set.

This prevents any alteration of the legacy block protection settings. As a consequence, all the data bytes in the memory area that are protected by the legacy block protection feature are also hardware protected against data modification if WP# is Low during commands changing Status Registers, Configuration Registers or DLR registers, with SRP0_NV set to 1. Similarly, the Security Region lock bits (LB3–LB0) are protected against programming.

The WP# function is not available when the Quad mode is enabled (CR1V[1] = 1) or QPI mode is enabled (CR2V[3] = 1). The WP# function is replaced by IO2 for input and output during Quad mode or QPI mode is enabled (CR2V[3] = 1) for receiving addresses, and data to be programmed (values are latched on rising edge of the SCK signal) as well as shifting out data on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

WP# has an internal pull-up resistance; when unconnected, WP# is at $V_{IH}$ and may be left unconnected in the host system if not used for Quad mode or QPI mode or protection.

## 3.9 IO3 / RESET#

IO3 is used for input and output during Quad mode (CR1V[1] = 1) or QPI mode is enabled (CR2V[3] = 1) for receiving addresses, and data to be programmed (values are latched on rising edge of the SCK signal) as well as shifting out data (on the falling edge of SCK, in SDR commands, and on every edge of SCK, in DDR commands).

The IO3 / RESET# input may also be used to initiate the hardware reset function when the IO3 / RESET# feature is enabled by writing Configuration Register 2 volatile or non-volatile bit 7 (CR2V[7] = 1)or (CR2NV[7] = 1). The input is only treated as RESET# when the device is not in Quad modes (114,144,444), CR1V[1] = 0, or when CS# is HIGH. When Quad modes are in use, CR1V[1] = 1or QPI mode is enabled (CR2V[3] = 1), and the device is selected with CS# LOW, the IO3 / RESET# is used only as IO3 for information transfer. When CS# is HIGH, the IO3 / RESET# is not in use for information transfer and is used as the reset input. By conditioning the reset operation on CS# HIGH during Quad modes (114,144,444), the reset function remains available during Quad modes (114, 144, 444).

When the system enters a reset condition, the CS# signal must be driven HIGH as part of the reset process and the IO3 / RESET# signal is driven LOW. When CS# goes HIGH, the IO3 / RESET# input transitions from being IO3 to being the reset input. The reset condition is then detected when CS# remains HIGH and the IO3 / RESET# signal remains LOW for $t_{RP}$. If a reset is not intended, the system is required to actively drive IO3 / RESET# to HIGH along with CS# being driven HIGH at the end of a transfer of data to the memory. Following transfers of data to the host system, the memory will drive IO3 HIGH during $t_{CS}$. This will ensure that IO3 / RESET# is not left floating or being pulled slowly to high by the internal or an external passive pull-up. Thus, an unintended reset is not triggered by the IO3 / RESET# not being recognized as high before the end of $t_{RP}$.

The IO3 / RESET# input reset feature is disabled when (CR2V[7] = 0).

The IO3 / RESET# input has an internal pull-up resistor and may be left unconnected in the host system if not used for Quad mode or the reset function. The internal pull-up will hold IO3 / RESET# HIGH after the host system has actively driven the signal high and then stops driving the signal.

Note that IO3 / RESET# input cannot be shared by more than one SPI-MIO memory if any of them are operating in Quad I/O mode as IO3 being driven to or from one selected memory may look like a reset signal to a second non-selected memory sharing the same IO3 / RESET# signal.

## 3.10 RESET#

The RESET# input provides a hardware method of resetting the device to STANDBY state, ready for receiving a command. When RESET# is driven to logic LOW ($V_{IL}$) for at least a period of $t_{RP}$, the device starts the hardware reset process.

RESET# causes the same initialization process as is performed when power comes up and requires $t_{PU}$ time.

RESET# may be asserted LOW at any time. To ensure data integrity any operation that was interrupted by a hardware reset should be reinitiated once the device is ready to accept a command sequence.

RESET# has an internal pull-up resistor and may be left unconnected in the host system if not used. The internal pull-up will hold Reset HIGH after the host system has actively driven the signal HIGH and then stops driving the signal.

The RESET# input is not available on all packages options. When not available the RESET# input of the device is tied to the inactive state.

## 3.11 Voltage Supply ($V_{CC}$)

$V_{CC}$ is the voltage source for all device internal logic. It is the single voltage used for all device internal functions including read, program, and erase.

## 3.12 Supply and Signal Ground ($V_{SS}$)

$V_{SS}$ is the common voltage drain and ground reference for the device core, input signal receivers, and output drivers.

## 3.13 Not Connected (NC)

No device internal signal is connected to the package connector nor is there any future plan to use the connector for a signal. The connection may safely be used for routing space for a signal on a printed circuit board (PCB).

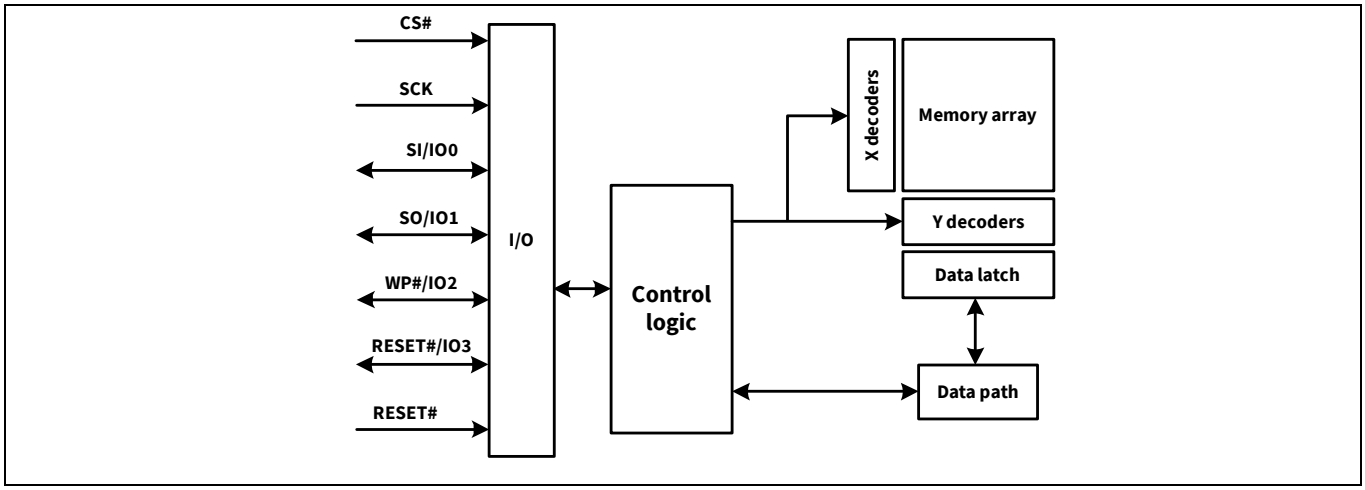
## 3.14 Reserved for Future Use (RFU)

No device internal signal is currently connected to the package connector but there is potential future use of the connector. It is recommended to not use RFU connectors for PCB routing channels so that the PCB may take advantage of future enhanced features in compatible footprint devices.

## 3.15 Do Not Use (DNU)

A device internal signal may be connected to the package connector. The connection may be used by Infineon for test or other purposes and is not intended for connection to any host system signal. Any DNU signal related function will be inactive when the signal is at $V_{IL}$. The signal has an internal pull-down resistor and may be left unconnected in the host system or may be tied to $V_{SS}$. Do not use these connections for PCB signal routing channels. Do not connect any host system signal to these connections.

# 4 Block diagram




## 4.1 System block diagrams

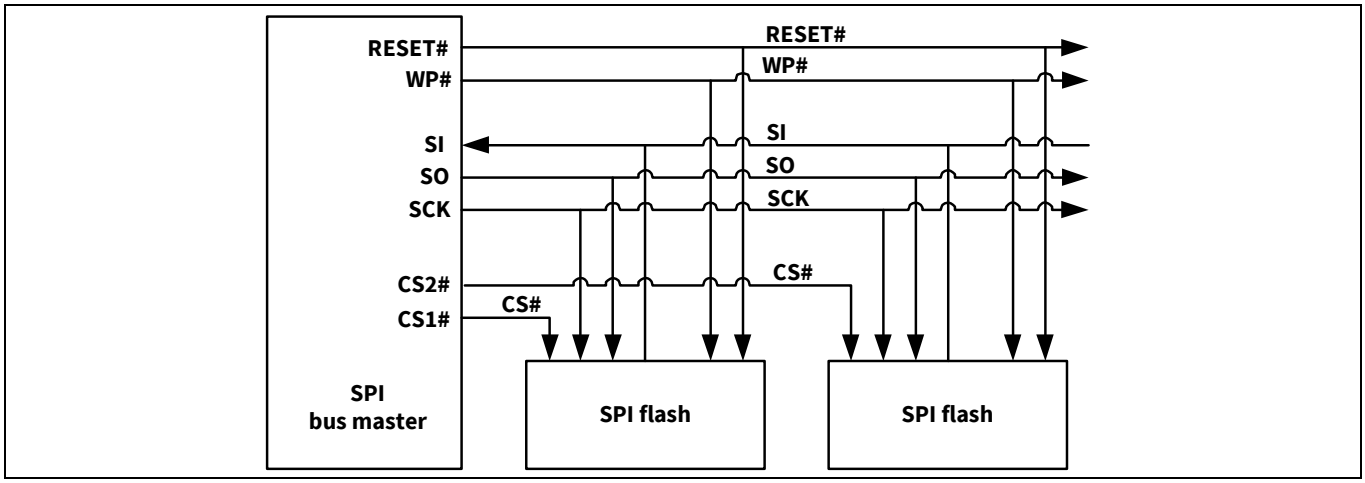



**Figure 7 Bus master and memory devices on the SPI bus - Single bit data path**

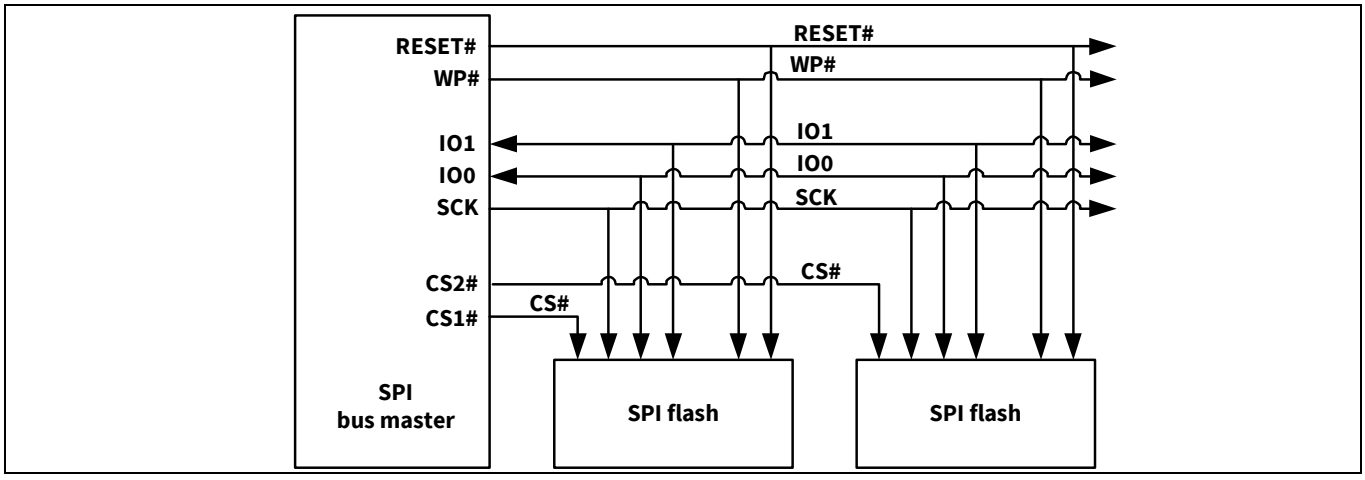



**Figure 8 Bus master and memory devices on the SPI bus - Dual bit data path**

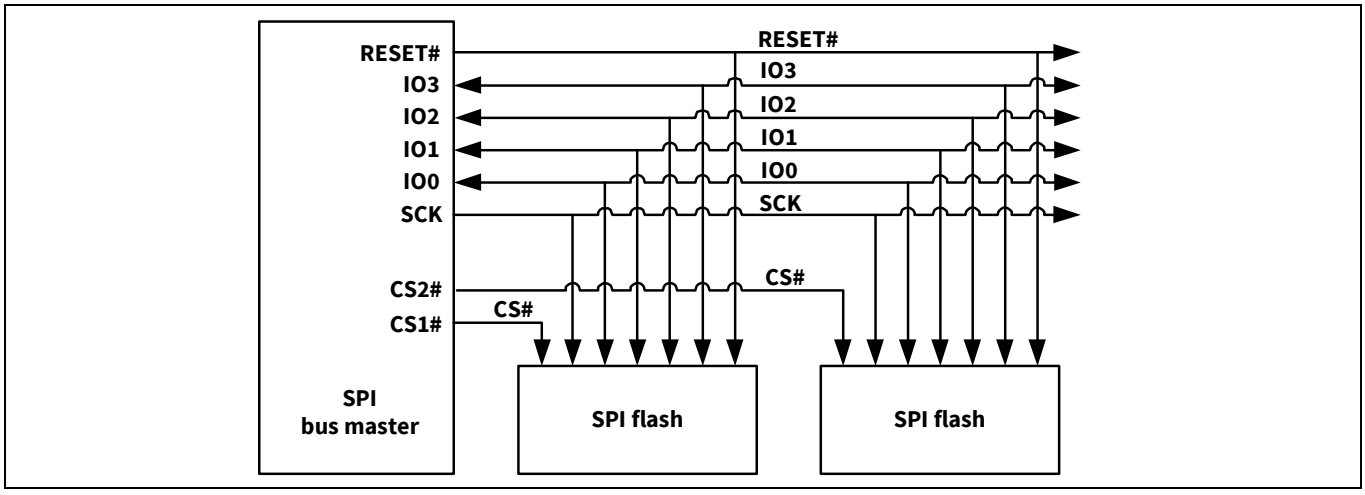



**Figure 9 Bus master and memory devices on the SPI bus - Quad bit data path - separate RESET#**

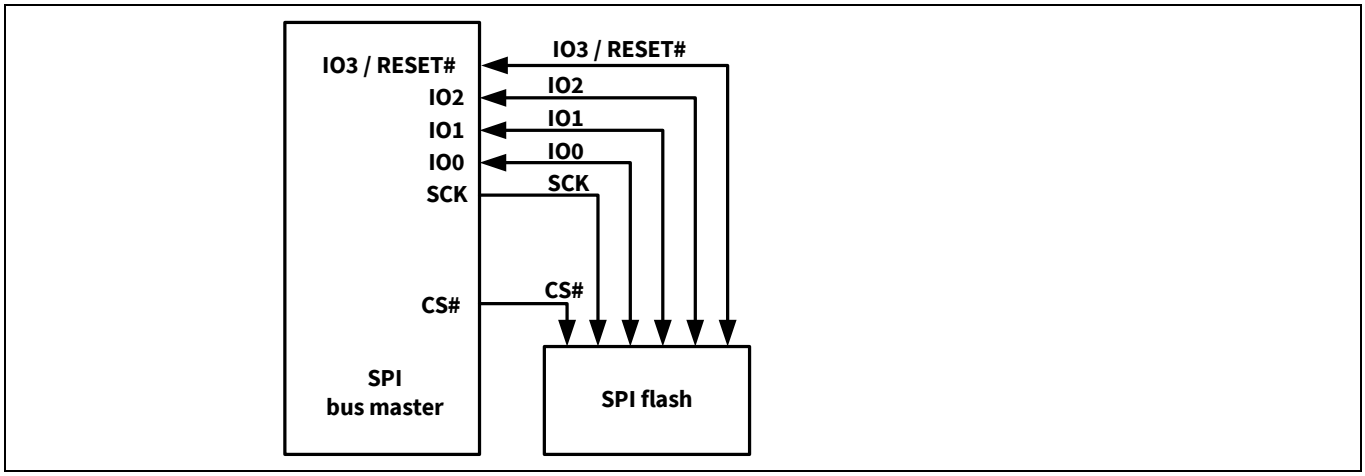



**Figure 10 Bus master and memory devices on the SPI bus - Quad bit data path - I/O3 / RESET#**

# 5 Signal protocols

## 5.1 SPI clock modes

### 5.1.1 Single data rate (SDR)

The FL-L family can be driven by an embedded micro-controller (bus master) in either of the two following clocking modes.

- **Mode 0** with clock polarity (CPOL) = 0 and, clock phase (CPHA) = 0
- **Mode 3** with CPOL = 1 and, CPHA = 1

For these two modes, input data into the device is always latched in on the rising edge of the SCK signal and the output data is always available from the falling edge of the SCK clock signal.

The difference between the two modes is the clock polarity when the bus master is in Standby mode and not transferring any data.

- SCK will stay at logic LOW state with CPOL = 0, CPHA = 0
- SCK will stay at logic HIGH state with CPOL = 1, CPHA = 1

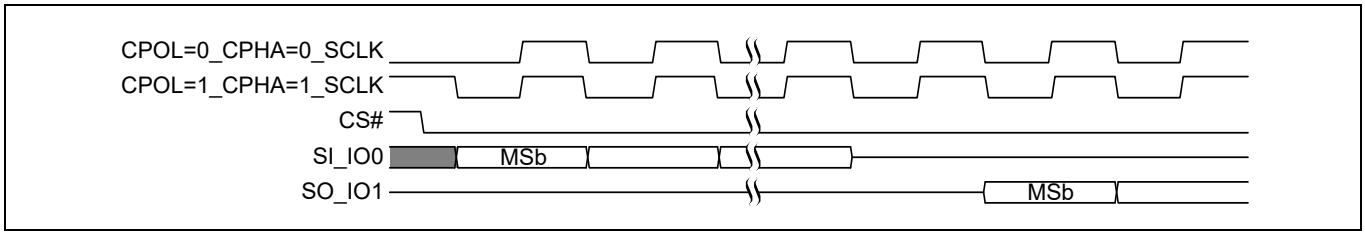



**Figure 11 SPI SDR modes supported**

Timing diagrams throughout the remainder of the document are generally shown as both Mode 0 and 3 by showing SCK as both HIGH and LOW at the fall of CS#. In some cases, a timing diagram may show only Mode 0 with SCK LOW at the fall of CS#. In such a case, Mode 3 timing simply means clock is HIGH at the fall of CS# so no SCK rising edge set up or hold time to the falling edge of CS# is needed for Mode 3.

SCK cycles are measured (counted) from one falling edge of SCK to the next falling edge of SCK. In Mode 0 the beginning of the first SCK cycle in a command is measured from the falling edge of CS# to the first falling edge of SCK because SCK is already LOW at the beginning of a command.

### 5.1.2 Double data rate (DDR)

Mode 0 and Mode 3 are also supported for DDR commands. In DDR commands, the instruction bits are always latched on the rising edge of clock, the same as in SDR commands. However, the address and input data that follow the instruction are latched on both the rising and falling edges of SCK. The first address bit is latched on the first rising edge of SCK following the falling edge at the end of the last instruction bit. The first bit of output data is driven on the falling edge at the end of the last access latency (dummy) cycle.

SCK cycles are measured (counted) in the same way as in SDR commands, from one falling edge of SCK to the next falling edge of SCK. In Mode 0 the beginning of the first SCK cycle in a command is measured from the falling edge of CS# to the first falling edge of SCK because SCK is already low at the beginning of a command.

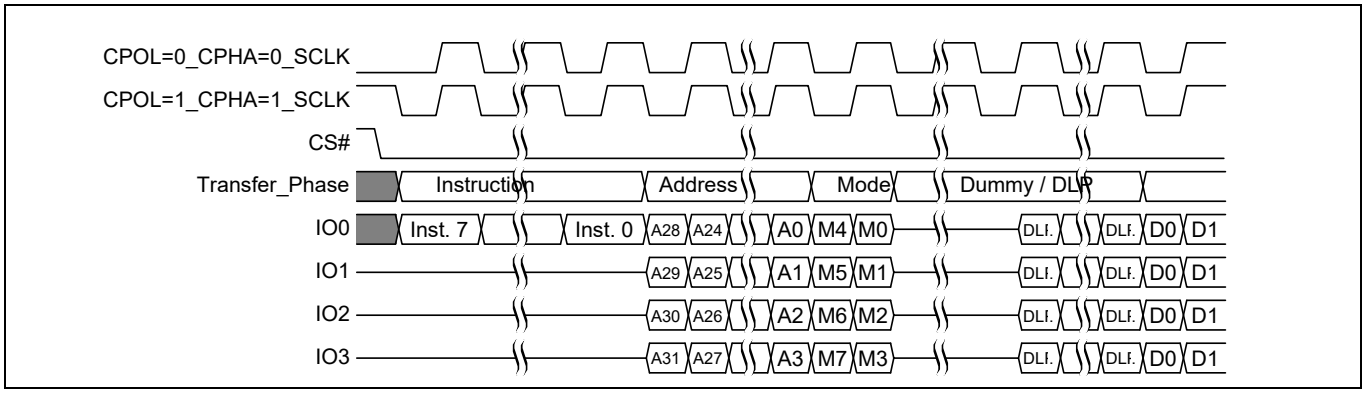



**Figure 12 SPI DDR modes supported**

## 5.2 Command protocol

All communication between the host system and FL-L family memory devices is in the form of units called commands. See **"Commands"** on page 62 for definition and details for all commands.

All commands begin with an 8-bit instruction that selects the type of information transfer or device operation to be performed. Commands may also have an address, instruction modifier, latency period, data transfer to the memory, or data transfer from the memory. All instruction, address, and data information is transferred sequentially between the host system and memory device.

Command protocols are also classified by a numerical nomenclature using three numbers to reference the transfer width of three command phases:

- instruction;
- address and instruction modifier (Continuous Read mode bits);
- data.

Single bit wide commands start with an instruction and may provide an address or data, all sent only on the SI signal. Data may be sent back to the host serially on the SO signal. This is referenced as a 1-1-1 command protocol for single bit width instruction, single bit width address and modifier, single bit data.

Dual-output or quad-output commands provide an address sent from the host as serial on SI (IO0) then followed by dummy cycles. Data is returned to the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-1-2 for Dual-O and 1-1-4 for Quad-O command protocols.

Dual or quad input / output (I/O) commands provide an address sent from the host as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3 then followed by dummy cycles. Data is returned to the host similarly as bit pairs on IO0 and IO1 or, four bit (nibble) groups on IO0, IO1, IO2, and IO3. This is referenced as 1-2-2 for dual I/O and 1-4-4 for quad I/O command protocols.

The FL-L family also supports a QPI mode in which all information is transferred in 4 bit width, including the instruction, address, modifier, and data. This is referenced as a 4-4-4 command protocol.

Commands are structured as follows:

- Each command begins with CS# going LOW and ends with CS# returning HIGH. The memory device is selected by the host driving the Chip Select (CS#) signal LOW throughout a command.
- The serial clock (SCK) marks the transfer of each bit or group of bits between the host and memory.
- Each command begins with an eight bit (byte) instruction. The instruction selects the type of information transfer or device operation to be performed. The instruction transfers occur on SCK rising edges. However, some Read commands are modified by a prior Read command, such that the instruction is implied from the earlier command. This is called Continuous Read mode. When the device is in Continuous Read mode, the instruction bits are not transmitted at the beginning of the command because the instruction is the same as the Read command that initiated the Continuous Read mode. In Continuous Read mode the command will begin with the read address. Thus, Continuous Read mode removes eight instruction bits from each Read command in a series of same type Read commands.
- The instruction may be stand alone or may be followed by address bits to select a location within one of several address spaces in the device. The instruction determines the address space used. The address may be either a 24-bit or a 32-bit, byte boundary, address. The address transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- In legacy SPI mode, the width of all transfers following the instruction are determined by the instruction sent. Following transfers may continue to be single bit serial on only the SI or Serial Output (SO) signals, they may be done in two bit groups per (dual) transfer on the IO0 and IO1 signals, or they may be done in 4 bit groups per (quad) transfer on the IO0–IO3 signals. Within the dual or quad groups the least significant bit is on IO0. More significant bits are placed in significance order on each higher numbered IO signal. Single bits or parallel bit groups are transferred in most to least significant bit order.
- In QPI mode, the width of all transfers is a 4 bit wide (Quad) transfer on the IO0–IO3 signals.
- Dual and quad I/O read instructions send an instruction modifier called Continuous Read mode bits, following the address, to indicate whether the next command will be of the same type with an implied, rather than an explicit, instruction. These mode bits initiate or end the Continuous Read mode. In Continuous Read mode, the next command thus does not provide an instruction byte, only a new address and mode bits. This reduces the time needed to send each command when the same command type is repeated in a sequence of commands. The mode bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- The address or mode bits may be followed by write data to be stored in the memory device or by a read latency period before read data is returned to the host.
- Write data bit transfers occur on SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- SCK continues to toggle during any read access latency period. The latency may be zero to several SCK cycles (also referred to as dummy cycles). At the end of the read latency cycles, the first read data bits are driven from the outputs on SCK falling edge at the end of the last read latency cycle. The first read data bits are considered transferred to the host on the following SCK rising edge. Each following transfer occurs on the next SCK rising edge, in SDR commands, or on every SCK edge, in DDR commands.
- If the command returns read data to the host, the device continues sending data transfers until the host takes the CS# signal HIGH. The CS# signal can be driven HIGH after any transfer in the read data sequence. This will terminate the command.
- At the end of a command that does not return data, the host drives the CS# input HIGH. The CS# signal must go HIGH after the eighth bit, of a stand alone instruction or, of the last write data byte that is transferred. That is, the CS# signal must be driven HIGH when the number of bits after the CS# signal was driven LOW is an exact multiple of eight bits. If the CS# signal does not go HIGH exactly at the eight bit boundary of the instruction or write data, the command is rejected and not executed.
- All instruction, address, and mode bits are shifted into the device with the most significant bits (MSb) first. The data bits are shifted in and out of the device MSb first. All data is transferred in byte units with the lowest address byte sent first. Following bytes of data are sent in lowest to highest byte address order i.e. the byte address increments.
- All attempts to read the flash memory array during a program, erase, or a write cycle (embedded operations) are ignored. The embedded operation will continue to execute without any affect. A very limited set of commands are accepted during an embedded operation. These are discussed in the individual command descriptions.
- Depending on the command, the time for execution varies. A command to read status information from an executing command is available to determine when the command completes execution and whether the command was successful.

## 5.2.1 Command sequence examples

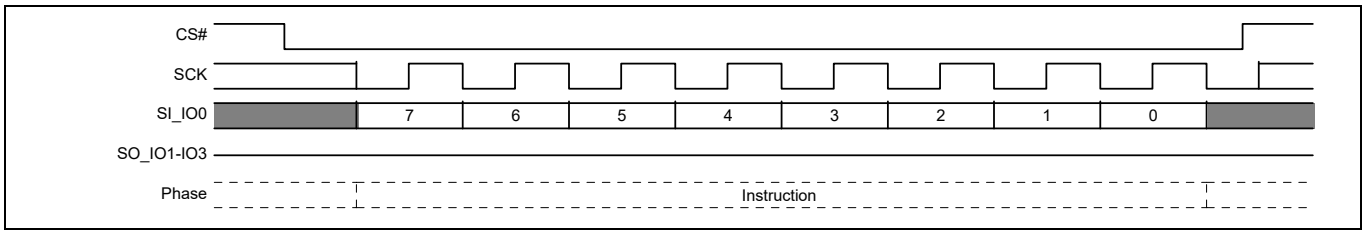



**Figure 13 Standalone Instruction command**

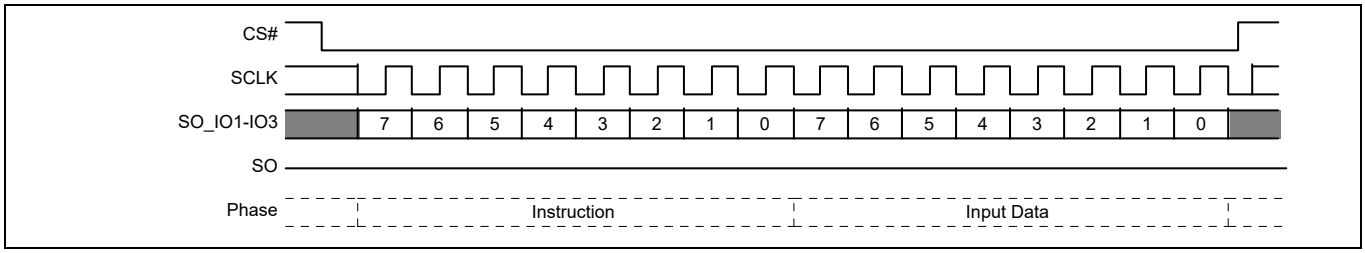



**Figure 14 Single Bit Wide Input command**

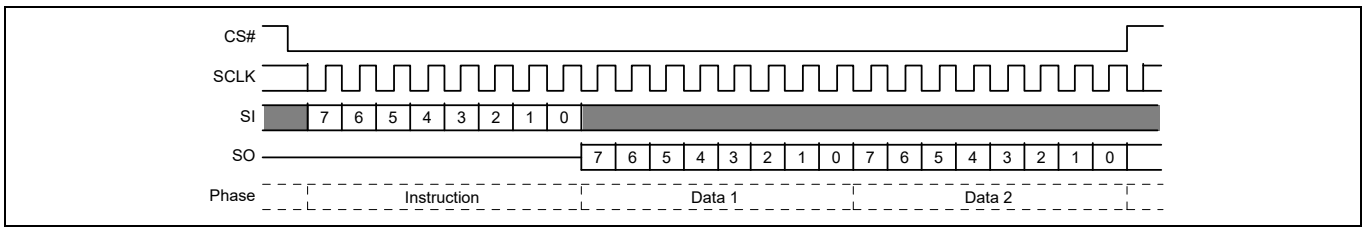



**Figure 15 Single Bit Wide Output command without latency**

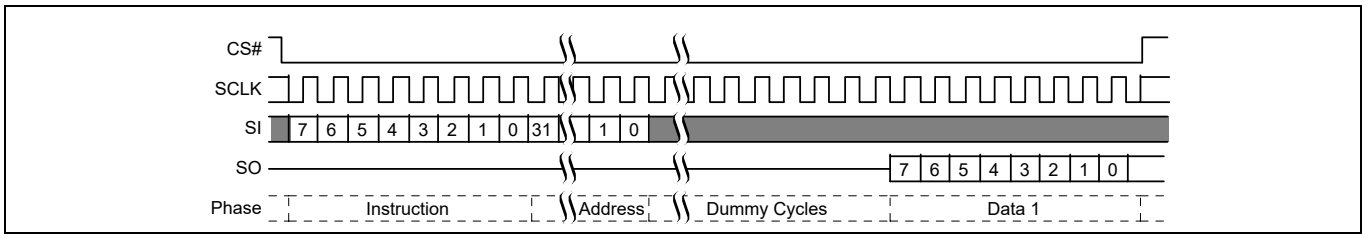



**Figure 16 Single Bit Wide I/O command with latency**

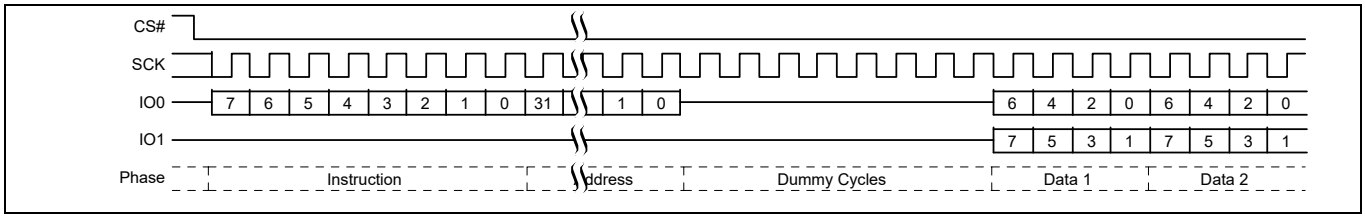



**Figure 17 Dual Output Read command**

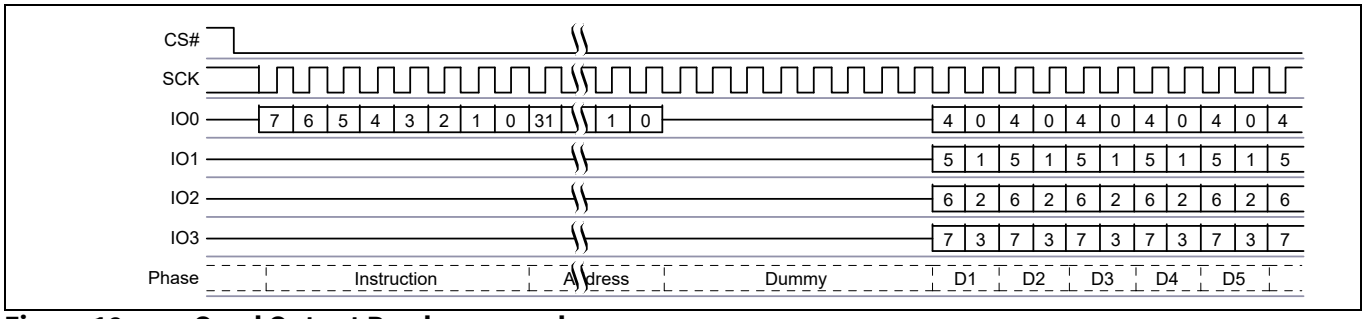



**Figure 18 Quad Output Read command**

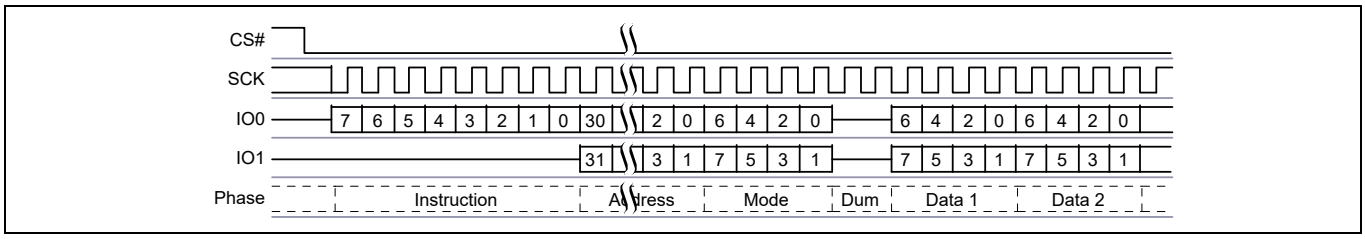



**Figure 19 Dual I/O command**

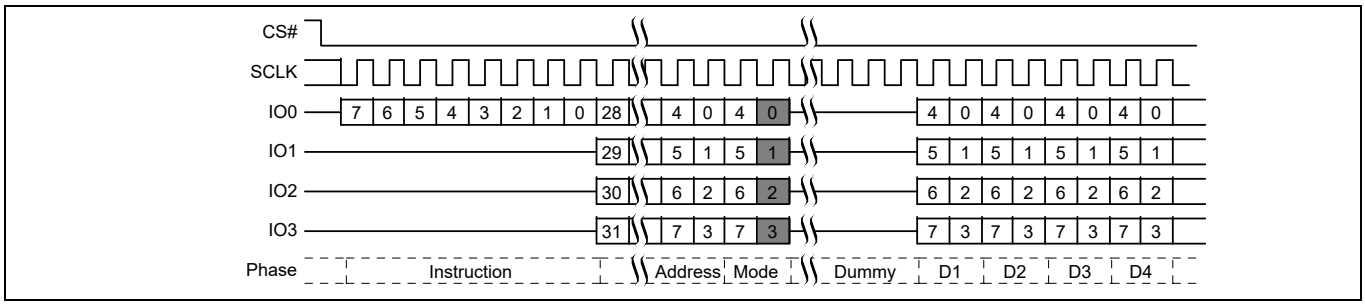



**Figure 20 Quad I/O command**[6]

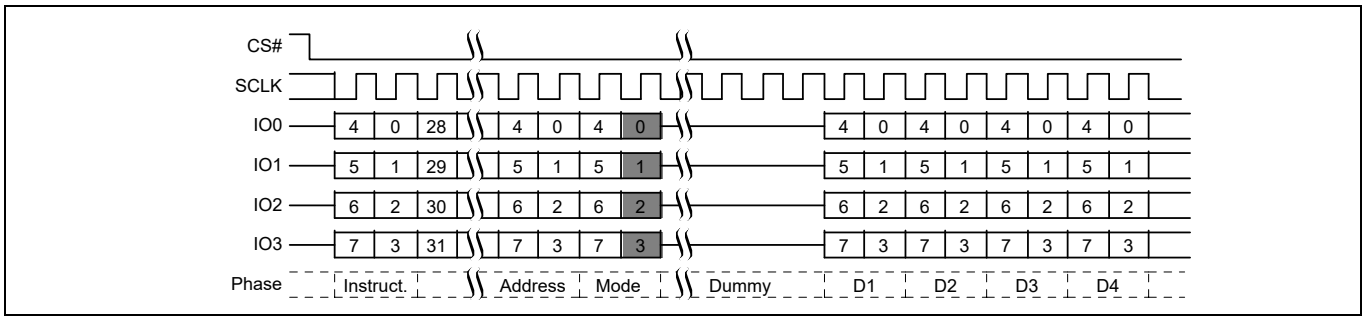



**Figure 21 Quad I/O Read command in QPI mode**[6]

**Note**
6. The gray bits are optional, the host does not have to drive bits during that cycle.

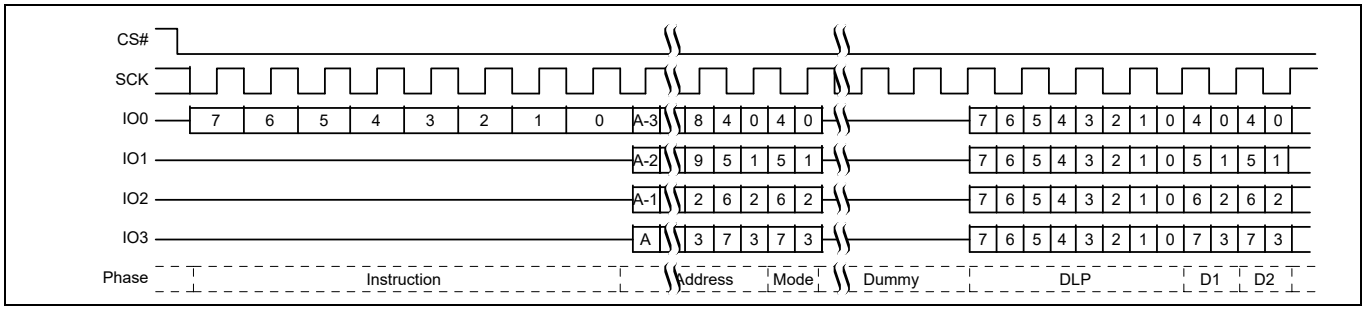



**Figure 22 DDR Quad I/O Read command**

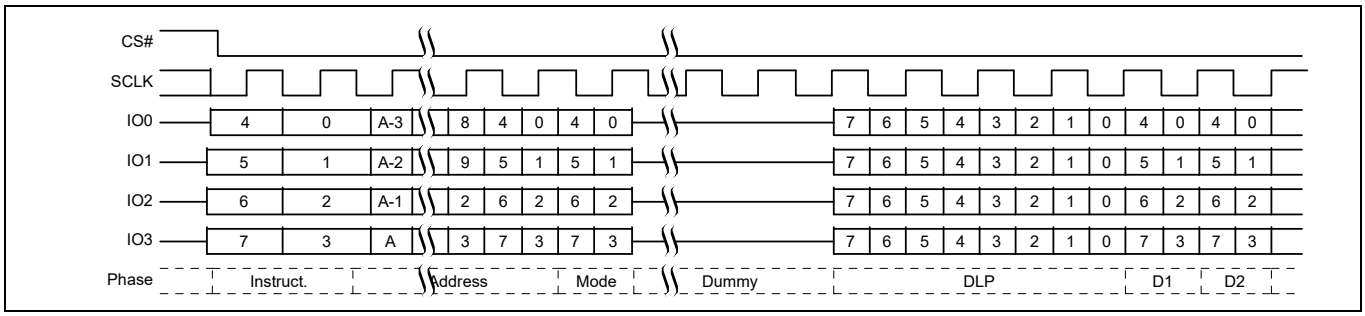



**Figure 23 DDR Quad I/O Read command QPI mode**

Additional sequence diagrams, specific to each command, are provided in **"Commands"** on page 62.

## 5.3 Interface states

This section describes the input and output signal levels as related to the SPI interface behavior.

**Table 3 Interface states summary**

| Interface state | $V_{CC}$ | SCK | CS# | RESET# | IO3 / RESET# | WP# / IO2 | SO / IO1 | SI / IO0 |
|---|---|---|---|---|---|---|---|---|
| Power-Off | <$V_{CC}$ (low) | X | X | X | X | X | Z | X |
| Low Power<br>Hardware Data Protection | <$V_{CC}$ (cut-off) | X | X | X | X | X | Z | X |
| Power-On (cold) Reset | ≥$V_{CC}$ (min) | X | HH | X | X | X | Z | X |
| Hardware (warm) Reset Non-Quad mode | ≥$V_{CC}$ (min) | X | X | HL | HL | X | Z | X |
| Hardware (warm) Reset Quad mode | ≥$V_{CC}$ (min) | X | HH | HL | HL | X | Z | X |
| Interface Standby | ≥$V_{CC}$ (min) | X | HH | HH | HH | X | Z | X |
| Instruction Cycle (Legacy SPI) | ≥$V_{CC}$ (min) | HT | HL | HH | HH | HV | Z | HV |
| Single Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | Z | HV |
| Single Latency (dummy) cycle | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | Z | X |
| Single Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | MV | X |
| Dual Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | HV | HV |
| Dual Latency (dummy) Cycle | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | X | X |
| Dual Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HH | X | MV | MV |
| Quad Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HV | HV | HV | HV |
| Quad Latency (dummy) cycle | ≥$V_{CC}$ (min) | HT | HL | HH | X | X | X | X |
| Quad Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | MV | MV | MV | MV |
| DDR Quad Input Cycle<br>Host to Memory Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | HV | HV | HV | HV |
| DDR Latency (dummy) cycle | ≥$V_{CC}$ (min) | HT | HL | HH | X | X | X | X |
| DDR Quad Output Cycle<br>Memory to Host Transfer | ≥$V_{CC}$ (min) | HT | HL | HH | MV | MV | MV | MV |

Legend
Z = No driver - floating signal
HL = Host driving $V_{IL}$
HH = Host driving $V_{IH}$
HV = Either HL or HH
X = HL or HH or Z
HT = Toggling between HL and HH
ML = Memory driving $V_{IL}$
MH = Memory driving $V_{IH}$
MV = Either ML or MH

### 5.3.1 Power-off

When the core supply voltage is at or below the $V_{CC\ (Low)}$ voltage, the device is considered to be powered off. The device does not react to external signals, and is prevented from performing any program or erase operation.

### 5.3.2 Low power hardware data protection

When $V_{CC}$ is less than $V_{CC\ (ut\text{-}off)}$, the memory device will ignore commands to ensure that program and erase operations can not start when the core supply voltage is out of the operating range. When the core voltage supply remains at or below the $V_{CC\ (Low)}$ voltage for ≥ $t_{PD}$ time, then rises to ≥ $V_{CC\ (Minimum)}$ the device will begin its Power On Reset (POR) process. POR continues until the end of $t_{PU}$. During $t_{PU}$ the device does not react to external input signals nor drive any outputs. Following the end of $t_{PU}$ the device transitions to the Interface Standby state and can accept commands. For additional information on POR see **"Power-on (cold) reset"** on page 151.

### 5.3.3 Hardware (warm) reset

A configuration option is provided to allow IO3 / RESET# to be used as a hardware reset input when the device is not in any quad or QPI mode or when it is in any quad mode or QPI mode and CS# is HIGH. In quad or QPI mode on some packages a separate reset input is provided (RESET #). When IO3 / RESET# or RESET# is driven LOW for $t_{RP}$ time the device starts the hardware reset process. The process continues for $t_{RPH}$ time. Following the end of both $t_{RPH}$ and the reset hold time following the rise of RESET# ($t_{RH}$) the device transitions to the Interface STANDBY state and can accept commands. For additional information on hardware reset see **"Reset"** on page 151.

### 5.3.4 Interface standby

When CS# is HIGH, the SPI interface is in STANDBY state. Inputs other than RESET# are ignored. The interface waits for the beginning of a new command. The next interface state is Instruction Cycle when CS# goes LOW to begin a new command.

While in interface STANDBY state the memory device draws standby current ($I_{SB}$) if no embedded algorithm is in progress. If an embedded algorithm is in progress, the related current is drawn until the end of the algorithm when the entire device returns to standby current draw.

### 5.3.5 Instruction cycle (Legacy SPI mode)

When the host drives the MSb of an instruction and CS# goes LOW, on the next rising edge of SCK the device captures the MSb of the instruction that begins the new command. On each following rising edge of SCK the device captures the next lower significance bit of the 8 bit instruction. The host keeps CS# LOW, and drives the Write Protect (WP#) and IO3 / RESET# signals as needed for the instruction. However, WP# is only relevant during instruction cycles of a WRR or WRAR command or any other commands which affect Status registers, Configuration Registers and DLR Registers, and is other wise ignored. IO3 / RESET# is driven HIGH when the device is not in Quad mode (CR1V[1] = 0) or QPI mode (CR2V[3] = 0) and hardware reset is not required.

Each instruction selects the address space that is operated on and the transfer format used during the remainder of the command. The transfer format may be Single, Dual O, Quad O, Dual I/O, or Quad I/O, or DDR Quad I/O. The expected next interface state depends on the instruction received.

Some commands are stand alone, needing no address or data transfer to or from the memory. The host returns CS# HIGH after the rising edge of SCK for the eighth bit of the instruction in such commands. The next interface state in this case is Interface Standby.

### 5.3.6 Instruction cycle (QPI mode)

In QPI mode, when CR2V[3] = 1, instructions are transferred 4 bits per cycle. In this mode, instruction cycles are the same as a quad input cycle. See **"QPP or QOR address input cycle"** on page 25.

### 5.3.7 Single input cycle - Host to Memory transfer

Several commands transfer information after the instruction on the single serial input (SI) signal from host to the memory device. The host keeps RESET# HIGH, CS# LOW, and drives SI as needed for the command. The memory does not drive the serial output (SO) signal.

The expected next interface state depends on the instruction. Some instructions continue sending address or data to the memory using additional single input cycles. Others may transition to single latency, or directly to single, dual, or quad output cycle states.

### 5.3.8 Single latency (dummy) cycle

Read commands may have zero to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the Latency Code in the Configuration Register (CR3V[3:0]). During the latency cycles, the host keeps RESET# and IO3 / RESET# HIGH, CS# LOW and SCK toggles. The write protect (WP#) signal is ignored. The host may drive the SI signal during these cycles or the host may leave SI floating. The memory does not use any data driven on SO or other I/O signals during the latency cycles. The memory does not drive the serial output (SO) or I/O signals during the latency cycles.

The next interface state depends on the command structure i.e. the number of latency cycles, and whether the read is single, dual, or quad width.

### 5.3.9 Single output cycle - Memory to Host transfer

Several commands transfer information back to the host on the single serial output (SO) signal. The host keeps RESET# and IO3 / RESET# HIGH, CS# LOW. The write protect (WP#) signal is ignored. The memory ignores the serial input (SI) signal. The memory drives SO with data.

The next interface state continues to be Single output Cycle until the host returns CS# to HIGH ending the command.

### 5.3.10 Dual input cycle - Host to Memory transfer

The read dual I/O command transfers two address or mode bits to the memory in each cycle. The host keeps RESET# and IO3 / RESET# HIGH, CS# LOW. The write protect (WP#) signal is ignored. The host drives address on SI / IO0 and SO / IO1.

The next interface state following the delivery of address and mode bits is a dual latency cycle if there are latency cycles needed or dual output cycle if no latency is required.

### 5.3.11 Dual latency (dummy) cycle

Read commands may have zero to several latency cycles during which read data is read from the main Flash memory array before transfer to the host. The number of latency cycles are determined by the latency code in the Configuration Register (CR3V[3:0]). During the latency cycles, the host keeps RESET# and IO3 / RESET# HIGH, CS# LOW, and SCK continues to toggle. The write protect (WP#) signal is ignored. The host may drive the SI / IO0 and SO / IO1 signals during these cycles or the host may leave SI / IO0 and SO / IO1 floating. The memory does not use any data driven on SI / IO0 and SO / IO1 during the latency cycles. The host must stop driving SI / IO0 and SO / IO1 on the falling edge of SCK at the end of the last latency cycle. It is recommended that the host stop driving them during all latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the SI / IO0 and SO / IO1 signals during the latency cycles.

The next interface state following the last latency cycle is a dual output cycle.

### 5.3.12 Dual output cycle - Memory to Host transfer

The read dual output and read dual I/O return data to the host two bits in each cycle. The host keeps RESET# and IO3 / RESET# HIGH, CS# LOW. The write protect (WP#) signal is ignored. The memory drives data on the SI / IO0 and SO / IO1 signals during the dual output cycles on the falling edge of SCK.

The next interface state continues to be dual output cycle until the host returns CS# to HIGH ending the command.

### 5.3.13 QPP or QOR address input cycle

The Quad Page Program and Quad Output Read commands send address to the memory only on IO0. The other IO signals are ignored. The host keeps RESET# and IO3 / RESET# HIGH, CS# LOW, and drives IO0.

For QPP the next interface state following the delivery of address is the quad input cycle. For QOR the next interface state following address is a quad latency cycle if there are latency cycles needed or quad output cycle if no latency is required.

### 5.3.14 Quad input cycle - Host to Memory transfer

The Quad I/O Read command transfers four address or mode bits to the memory in each cycle. In QPI mode, the Quad I/O Read and Page Program commands transfer four data bits to the memory in each cycle, including the instruction cycles. The host keeps CS# LOW, and drives the IO signals.

For Quad I/O Read, the next interface state following the delivery of address and mode bits is a quad latency cycle if there are latency cycles needed or quad output cycle if no latency is required. For QPI mode page program, the host returns CS# HIGH following the delivery of data to be programmed and the interface returns to STANDBY state.

### 5.3.15 Quad latency (dummy) cycle

Read commands may have zero to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the latency code in the Configuration Register (CR3V[3:0]). During the latency cycles, the host keeps CS# LOW and continues to toggle SCK. The host may drive the IO signals during these cycles or the host may leave the IO floating. The memory does not use any data driven on IO during the latency cycles. The host must stop driving the IO signals on the falling edge at the end of the last latency cycle. It is recommended that the host stop driving them during all latency cycles so that there is sufficient time for the host drivers to turn off before the memory begins to drive at the end of the latency cycles. This prevents driver conflict between host and memory when the signal direction changes. The memory does not drive the IO signals during the latency cycles.

The next interface state following the last latency cycle is a quad output cycle.

### 5.3.16 Quad output cycle - Memory to Host transfer

The Quad-O and Quad I/O Read returns data to the host four bits in each cycle. The host keeps CS# LOW. The memory drives data on IO0-IO3 signals during the quad output cycles.

The next interface state continues to be quad output cycle until the host returns CS# to HIGH ending the command.

### 5.3.17 DDR quad input cycle - Host to Memory transfer

The DDR Quad I/O Read command sends address, and mode bits to the memory on all the IO signals. Four bits are transferred on the rising edge of SCK and four bits on the falling edge in each cycle. The host keeps CS# LOW.

The next interface state following the delivery of address and mode bits is a DDR latency cycle.

### 5.3.18 DDR latency cycle

DDR Read commands may have one to several latency cycles during which read data is read from the main flash memory array before transfer to the host. The number of latency cycles are determined by the latency code in the Configuration Register (CR3V[3:0]). During the latency cycles, the host keeps CS# LOW. The host may not drive the IO signals during these cycles. So that there is sufficient time for the host drivers to turn off before the memory begins to drive. This prevents driver conflict between host and memory when the signal direction changes. The memory has an option to drive all the IO signals with a Data Learning Pattern (DLP) during the last 4 latency cycles. The DLP option should not be enabled when there are fewer than five latency cycles so that there is at least one cycle of high impedance for turn around of the IO signals before the memory begins driving the DLP. When there are more than 4 cycles of latency the memory does not drive the IO signals until the last four cycles of latency.

The next interface state following the last latency cycle is a DDR quad output cycle, depending on the instruction.

### 5.3.19 DDR quad output cycle - Memory to Host transfer

The DDR quad I/O read command returns bits to the host on all the IO signals. Four bits are transferred on the rising edge of SCK and four bits on the falling edge in each cycle. The host keeps CS# LOW.

The next interface state continues to be DDR quad output cycle until the host returns CS# to HIGH ending the command.

## 5.4 Data protection

Some basic protection against unintended changes to stored data are provided and controlled purely by the hardware design. These are described below. Other software managed protection methods are discussed in the software section of this document.

### 5.4.1 Power-up

When the core supply voltage is at or below the $V_{CC\ (Low)}$ voltage, the device is considered to be powered off. The device does not react to external signals, and is prevented from performing any program or erase operation. User is not allowed to enter any valid command during tPU.

### 5.4.2 Low power

When $V_{CC}$ is less than $V_{CC\ (Cut\text{-}off)}$ the memory device will ignore commands to ensure that program and erase operations can not start when the core supply voltage is out of the operating range.

### 5.4.3 Clock pulse count

The device verifies that all data modifying commands consist of a clock pulse count that is a multiple of eight bit transfers (byte boundary) before executing them. A command not ending on an 8 bit (byte) boundary is ignored and no error status is set for the command.

### 5.4.4 Deep power down (DPD)

In DPD mode the device responds only to the Resume from DPD command (RES ABh). All other commands are ignored during DPD mode, thereby protecting the memory from program and erase operations. If the IO3 / RESET# function has been enabled (CR2V[7] = 1) or if RESET# is active, IO3 / RESET# or RESET# going LOW will start a hardware reset and release the device from DPD mode.

# 6 Address space maps

## 6.1 Overview

### 6.1.1 Extended address

The FL-L family supports 32-bit (4-byte) addresses to enable higher density devices than allowed by previous generation (legacy) SPI devices that supported only 24-bit (3-byte) addresses. A 24-bit, byte resolution, address can access only 16 MB (128 Mb) maximum density. A 32-bit, byte resolution, address allows direct addressing of up to a 4 GB (32 Gb) address space.

Legacy commands continue to support 24-bit addresses for backward software compatibility. Extended 32-bit addresses are enabled in two ways:

- Extended address mode — a volatile Configuration Register bit that changes all legacy commands to expect 32-bits of address supplied from the host system.
- 4-byte address commands — that perform both legacy and new functions, which always expect 32-bit address.

The default condition for extended address mode, after power-up or reset, is controlled by a non-volatile configuration bit. The default extended address mode may be set for 24- or 32-bit addresses. This enables legacy software compatible access to the first 128 Mb of a device or for the device to start directly in 32-bit address mode.

### 6.1.2 Multiple address spaces

Many commands operate on the main Flash memory array. Some commands operate on address spaces separate from the main Flash array. Each separate address space uses the full 24- or 32-bit address but may only define a small portion of the available address space.

## 6.2 Flash memory array

The main Flash array is divided into uniform erase units called physical blocks (64 KB), half blocks (32 KB) and sectors (4 KB).

**Table 4 S25FL064L sector address map**

| Block size (KB) | Block count | Block range | Half block size (KB) | Half block count | Half block range | Sector size (KB) | Sector count | Sector range | Address range (byte address) | Notes |
|---|---|---|---|---|---|---|---|---|---|---|
| 64 | 1 | BA00 | 32 | 1 | HBA00 | 4 | 1 | SA00 | 0000000h–0000FFFh | Sector starting address — Sector ending address |
| | | | : | : | : | : | : | : | : | |
| | | | 32 | 2 | HBA01 | 4 | 16 | SA15 | 000F000h–000FFFFh | |
| : | : | : | : | : | : | : | : | : | : | |
| 64 | 128 | BA127 | 32 | 255 | HBA254 | 4 | 2032 | SA2031 | 07F0000h–07F0FFFh | |
| | | | : | : | : | : | : | : | : | |
| | | | 32 | 256 | HBA255 | 4 | 2048 | SA2047 | 07FF000h–07FFFFFh | |

## 6.3 ID address space

The RDID command (9Fh) reads information from a separate flash memory address space for device identification (ID). See **"Device ID address map"** on page 141 for the tables defining the contents of the ID address space. The ID address space is programmed by Infineon and read-only for the host system.

### 6.3.1 Device Unique ID

A 64-bit unique number is located in 8-bytes of the unique device ID address space see **Table 44**. This Unique ID may be used as a software readable serial number that is unique for each device.

## 6.4 JEDEC JESD216 serial flash discoverable parameters (SFDP) space

The RSFDP command (5Ah) reads information from a separate flash memory address space for device identification, feature, and configuration information, in accord with the JEDEC JESD216 standard for serial flash discoverable parameters. The ID address space is incorporated as one of the SFDP parameters. See **"JEDEC JESD216B serial flash discoverable parameters"** on page 130 for the tables defining the contents of the SFDP address space. The SFDP address space is programmed by Infineon and read-only for the host system.

## 6.5 Security Regions address space

Each FL-L family memory device has a 1024-byte Security Regions address space that is separate from the main flash array. The Security Regions area is divided into 4, individually lockable 256-byte regions. The Security Regions memory space is intended to hold information that can be temporarily protected or permanently locked from further program or erase.

The regions data bytes are erased to FFh when shipped from Infineon. The regions may be programmed and erased like any other flash memory address space when not protected or locked. Each region can be individually erased. The Security Region lock bits (CR1NV[5:2]) are located in the Configuration Register 1. The Security Region lock bits are one time programmable (OTP) and after being programmed (set to 1) a lock bit permanently protects the related region from further erase or programming.

Regions 2 and 3 also have temporary protection from program or erase by the Protection Register (PR) NVLock bit. The NVLock bit is volatile and set or cleared by the IRP logic and commands. See **"Protection Register (PR)"** on page 46.

The Security Region Password Protection bit in the IRP Register (IRP[2]) allows Regions 2 and 3 to be protected from program and erase operations until a password is provided. The Security Region Read Protection bit in the IRP Register (IRP[6]) allows region 3 to also be protected from read operations until a password is provided. Attempting to read in a region, that is protected from read, returns invalid and undefined data. See **"Individual and Region Protection Register (IRP)"** on page 44.

Attempting to erase or program in a region that is locked or protected will fail with the P_ERR or E_ERR bit in SR2V[6:5] set to '1'. (see **"Status Register 2 Volatile (SR2V)"** on page 32 for detail descriptions).

**Table 5 Security Region address map**

| Region | Byte address range (hex) | Initial delivery state (hex) |
|---|---|---|
| Region 0 | 000 to 0FF | All bytes = FF |
| Region 1 | 100 to 1FF | All bytes = FF |
| Region 2 | 200 to 2FF | All bytes = FF |
| Region 3 | 300 to 3FF | All bytes = FF |

## 6.6 Registers

Registers are small groups of memory cells used to configure how the FL-L family memory device operates or to report the status of device operations. The registers are accessed by specific commands. The commands (and hexadecimal instruction codes) used for each register are noted in each register description.

In legacy SPI memory devices the individual register bits could be a mixture of volatile, non-volatile, or one time programmable (OTP) bits within the same register. In some configuration options the type of a register bit could change e.g. from non-volatile to volatile.

The FL-L family uses separate non-volatile or volatile memory cell groups (areas) to implement the different register bit types. However, the legacy registers and commands continue to appear and behave as they always have for legacy software compatibility. There is a non-volatile and a volatile version of each legacy register when that legacy register has volatile bits or when the command to read the legacy register has zero read latency. When such a register is read the volatile version of the register is delivered. During power-on reset (POR), hardware reset, or software reset, the non-volatile version of a register is copied to the volatile version to provide the default state of the volatile register. When Non-volatile Register bits are written the non-volatile version of the register is erased and programmed with the new bit values and the volatile version of the register is updated with the new contents of the non-volatile version. When OTP bits are programmed the non-volatile version of the register is programmed and the appropriate bits are updated in the volatile version of the register. When Volatile register bits are written, only the volatile version of the register has the appropriate bits updated.

The type for each bit is noted in each register description. The default state shown for each bit refers to the state after power-on reset, hardware reset, or software reset if the bit is volatile. If the bit is non-volatile or OTP, the default state is the value of the bit when the device is shipped from Infineon. Special attention must be given when writing the non-volatile registers that there is a stable power supply with no disruption, this will guarantee the correct data is written to the register.

**Table 6 Register description**

| Register | Type | Bits | Abbreviation |
|---|---|---|---|
| Status Register 1 | Non-volatile | 7:0 | SR1NV[7:0] |
| | Volatile | 7:0 | SR1V[7:0] |
| Status Register 1 | Volatile | 7:0 | SR2V[7:0] |
| Configuration Register 1 | Non-volatile/OTP | 7:0 | CR1NV[7:0] |
| | Volatile | 7:0 | CR1V[7:0] |
| Configuration Register 2 | Non-volatile | 7:0 | CR2NV[7:0] |
| | Volatile | 7:0 | CR2V[7:0] |
| Configuration Register 3 | Non-volatile | 7:0 | CR3NV[7:0] |
| | Volatile | 7:0 | CR3V[7:0] |
| Individual and Region Protection Register | OTP | 15:0 | IRP[15:0] |
| Password Register | OTP | 63:0 | PASS[63:0] |
| Individual Block Lock Access Register | Volatile | 7:0 | IBLAR[7:0] |
| Pointer Region Protection Register | Non-volatile | 31:0 | PRPR[31:0] |
| DDR Data Learning Registers | OTP | 7:0 | DLRNV[7:0] |
| | Volatile | 7:0 | DLRV[7:0] |

### 6.6.1 Status Register 1

#### 6.6.1.1 Status Register 1 Non-volatile (SR1NV)

Related Commands: Non-volatile Write Enable (WREN 06h), Write Disable (WRDI 04h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h)

**Table 7 Status Register 1 Non-volatile (SR1NV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | SRP0_NV | Status Register Protect 0 Default | Non-volatile | 0 | Provides the default state for SRP0. |
| 6 | SEC_NV | Sector / Block Protect | Non-volatile | 0 | Provides the defaults state for SEC |
| 5 | TBPROT_NV | TBPROT Default | Non-volatile | 0 | Provides the default state for TBPROT |
| 4 | BP_NV2 | Legacy Block Protection Default | Non-volatile | 000b | Provides the default state for BP bits. |
| 3 | BP_NV1 | | | | |
| 2 | BP_NV0 | | | | |
| 1 | WEL_D | WEL Default | Non-volatile read only | 0 | Provides the default state for the WEL status. Not user programmable. |
| 0 | WIP_D | WIP Default | Non-volatile read only | 0 | Provides the default state for the WIP status. Not user programmable. |

**Status Register Protect Non-volatile (SRP0_NV) SR1NV[7]**: Provides the default state for SRP0. See **"Status Register Protect (SRP1, SRP0)"** on page 50.

**Sector / block protect (SEC_NV) SR1NV[6]**: Provides the default state for SEC.

The type for each bit is noted in each register description. The default state shown for each bit refers to the state after power-on reset, hardware reset, or software reset if the bit is volatile. If the bit is non-volatile or OTP, the default state is the value of the bit when the device is shipped from Infineon. Special attention must be given when writing the non-volatile registers that there is a stable power supply with no disruption, this will guarantee the correct data is written to the register.

**Top or Bottom Protection (TBPROT_NV) SR1NV[5]:** Provides the default state for TBPROT.

**Legacy Block Protection (BP_NV3, BP_NV2, BP_NV1, BP_NV0) SR1NV[4:2]**: Provides the default state for BP_2 to BP_0 bits.

**Write Enable Latch Default (WEL_D) SR1NV[1]**: Provides the default state for the WEL Status in SR1V[1]. This bit is programmed by Infineon and is not user programmable.

**Write in Progress Default (WIP_D) SR1NV[0]**: Provides the default state for the WIP Status in SR1V[0]. This bit is programmed by Infineon and is not user programmable.

### 6.6.1.2 Status Register 1 Volatile (SR1V)

Related commands: Read Status Register 1 (RDSR1 05h), Write Enable for Volatile (WRENV 50h), Write Registers (WRR 01h), Clear Status Register (CLSR 30h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). This is the register displayed by the RDSR1 command.

**Table 8 Status Register 1 Volatile (SR1V)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | SRP0 | Status Register Protect 0 | Volatile | SR1NV | 1 = Locks state of SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV when WP# is LOW, by not executing any commands that would affect SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV<br>0 = No register protection, even when WP# is LOW. |
| 6 | SEC | Sector / Block Protect | Volatile | | 0 = BP2–BP0 protect 64 kB blocks<br>1 = BP2–BP0 protect 4 kB sectors |
| 5 | TBPROT | Top or Bottom Relative Protection | Volatile | | 1 = BP starts at bottom (Low address)<br>0 = BP starts at top (High address) |
| 4 | BP2 | Legacy Block Protection Volatile | Volatile | | Protects the selected range of sectors (blocks) from program or erase. |
| 3 | BP1 | | | | |
| 2 | BP0 | | | | |
| 1 | WEL | Write Enable Latch | Volatile read only | | 0 = Not write enabled, no embedded operation can start<br>1 = write enable, embedded operation can start<br>This bit is not affected by WRR or WRAR, only WREN WRENV, WRDI and CLSR commands affect this bit. |
| 0 | WIP | Write in Progress | Volatile read only | | 1 = Device busy, an embedded operation is in progress such as program or erase<br>0 = Ready device is in Standby mode and can accept commands<br>This bit is not affected by WRR or WRAR, it only provides WIP status. |

**Status Register Protect 0 (SRP0) SR1V[7]:** Places the device in the Hardware Protected mode when this bit is set to 1 and the WP# input is driven LOW. In this mode, any command that change status registers or Configuration Registers are ignored and not accepted for execution, effectively locking the state of the Status Registers and Configuration Registers SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV bits, by making the registers read-only. If WP# is HIGH, Status Registers and Configuration Registers SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV may be changed and Configuration Registers SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV may be changed. WP# has no effect on the writing of any other registers. SRP0 tracks any changes to the non-volatile version of this bit (SRP0_NV). When QPI or QIO mode is enabled (CR2V[3] or CR1V[1] = '1') the internal WP# signal level is = 1 because the WP# external input is used as IO2 when either mode is active. This effectively turns off hardware protection. The Register SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are unlocked and can be written. See **"Status Register Protect (SRP1, SRP0)"** on page 50.

**Sector / Block Protect (SEC) SR1V[6]:** This bit controls if the block protect bits (BP2, BP1, BP0) protect either 4 kB Sectors (SEC = 1) or 64kB Blocks (SEC = 0). See **"Legacy block protection"** on page 53 for a description of how the SEC bit value select the memory array area protected.

**TBPROT SR1V[5]:** This bit defines the reference point of the legacy block protection bits BP2, BP1, and BP0 in the Status Register. As described in the status register section, the BP2–0 bits allow the user to optionally protect a portion of the array, ranging from 1/64, ¼, ½, etc., up to the entire array. When TBPROT is set to '0' the legacy block protection is defined to start from the top (maximum address) of the array. When TBPROT is set to a '1' the legacy block protection is defined to start from the bottom (zero address) of the array. TBPROT tracks any changes to the non-volatile version of this bit (TBPROT_NV).

**Legacy Block Protection (BP2, BP1, BP0) SR1V[4:2]**: These bits define the main flash array area to be protected against program and erase commands. See **"Legacy block protection"** on page 53 for a description of how the BP bit values select the memory array area protected.

**Write Enable Latch (WEL) SR1V[1]**: The WEL bit must be set to 1 to enable program, write, or erase operations as a means to provide protection against inadvertent changes to memory or register values. The Write Enable (WREN) command execution sets the write enable latch to a '1' to allow any Program, Erase, or Write commands to execute afterwards. The Write Disable (WRDI) command can be used to set the write enable latch to a '0' to prevent all Program, Erase, and Write commands from execution. The WEL bit is cleared to 0 at the end of any successful program, write, or erase operation. Following a failed operation the WEL bit may remain set and should be cleared with a CLSR command. After a power down / power up sequence, hardware reset, or software reset, the write enable latch is set to a WEL_D. The WRR or WRAR command does not affect this bit.

**Write in Progress (WIP) SR1V[0]**: Indicates whether the device is performing a program, write, erase operation, or any other operation, during which a new operation command will be ignored. When the bit is set to a '1' the device is busy performing an operation. While WIP is '1', only Read Status (RDSR1 or RDSR2), Read Any Register (RDAR), Erase / Program Suspend (EPS), Clear Status Register (CLSR), and Software Reset (RSTEN 66h followed by RST 99h) commands are accepted. EPS command will only be accepted if memory array erase or program operations are in progress. The Status Register E_ERR and P_ERR bits are updated while WIP = 1. When P_ERR or E_ERR bits are set to one, the WIP bit will remain set to one indicating the device remains busy and unable to receive new operation commands. A Clear Status Register (CLSR) command must be received to return the device to Standby mode. When the WIP bit is cleared to 0 no operation is in progress. This is a read-only bit.

### 6.6.2 Status Register 2 Volatile (SR2V)

Related Commands: Read Status Register 2 (RDSR2 07h), Read Any Register (RDAR 65h). Status Register 2 does not have user programmable non-volatile bits, all defined bits are volatile read only status. The default state of these bits are set by hardware.

**Table 9 Status Register 1 volatile (SR2V)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | – | 0 | Reserved for Future Use |
| 6 | E_ERR | Erase Error occurred | Volatile read only | 0 | 1 = Error occurred<br>0 = No error |
| 5 | P_ERR | Programming Error occurred | Volatile read only | 0 | 1 = Error occurred<br>0 = No Error |
| 4 | RFU | Reserved | – | 0 | Reserved for Future Use |
| 3 | RFU | Reserved | – | 0 | Reserved for Future Use |
| 2 | RFU | Reserved | – | 0 | Reserved for Future Use |
| 1 | ES | Erase Suspend | Volatile read only | 0 | 1 = In Erase Suspend mode<br>0 = Not in Erase Suspend mode |
| 0 | PS | Program Suspend | Volatile read only | 0 | 1 = In Program Suspend mode<br>0 = Not in Program Suspend mode |

**Erase Error (E_ERR) SR2V[6]**: The Erase Error bit is used as an erase operation success or failure indication. When the Erase Error bit is set to a '1' it indicates that there was an error in the last erase operation. This bit will also be set when the user attempts to erase an individual protected main memory sector or erase a locked Security Region. The Chip Erase command will set E_ERR if a protected sector is found during the command execution. When the Erase Error bit is set to a '1' this bit can be cleared to zero with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR or WRAR commands.

**Program Error (P_ERR) SR2V[5]**: The program error bit is used as a program operation success or failure indication. When the program error bit is set to a '1' it indicates that there was an error in the last program operation. This bit will also be set when the user attempts to program within a protected main memory sector, or program within a locked Security Region. When the Program Error bit is set to a '1' this bit can be cleared to zero with the Clear Status Register (CLSR) command. This is a read-only bit and is not affected by the WRR or WRAR commands.

**Erase Suspend (ES) SR2V[1]**: The Erase Suspend bit is used to determine when the device is in Erase Suspend mode. This is a status bit that cannot be written by the user. When erase suspend bit is set to '1', the device is in Erase Suspend mode. When erase suspend bit is cleared to '0', the device is not in Erase Suspend mode. Refer to **"Program or Erase Suspend (PES 75h)"** on page 106 for details about the Erase Suspend/Resume commands.

**Program Suspend (PS) SR2V[0]:** The Program Suspend bit is used to determine when the device is in Program Suspend mode. This is a status bit that cannot be written by the user. When Program Suspend bit is set to '1', the device is in Program Suspend mode. When the program suspend bit is cleared to '0', the device is not in Program Suspend mode. Refer to **"Program or Erase Suspend (PES 75h)"** on page 106 for details.

### 6.6.3 Configuration Register 1

Configuration register 1 controls certain interface and data protection functions. The register bits can be changed using the WRR command with sixteen input cycles or with the WRAR command.

#### 6.6.3.1 Configuration Register 1 Non-volatile (CR1NV)

Related commands: Non-volatile Write Enable (WREN 06h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 10 Configuration Register 1 Non-volatile (CR1NV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | SUS_D | Suspend Status Default | Non-volatile read only | 0 | Provides the default state for the suspend status. Not user programmable. |
| 6 | CMP_NV | Complement Protection Default | Non-volatile | 0 | Provides the default state for CMP. |
| 5 | LB3 | Security Region Lock bits | OTP | 0 | OTP lock Bits 3:0 for Security Regions 3:0<br>0 = Security Region not locked<br>1 = Security Region permanently locked |
| 4 | LB2 | | | 0 | |
| 3 | LB1 | | | 0 | |
| 2 | LB0 | | | 0 | |
| 1 | QUAD_NV | Quad Default | Non-volatile | 0 | Provides the default state for QUAD. |
| 0 | SRP1_D | Status Register Protect 1 Default | OTP | 0 | When IRP[2:0] = '111' SRP1_D bit is programmable.<br>Lock current state of SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV<br>1 = Registers permanently locked<br>0 = Registers not protected by SRP1 after POR |

**Suspend Erase/Program Status (SUS_D) CR1NV[7]:** Provides the default state for the SUS bit in CR1V[7]. This bit is not user programmable.

**Complement Protect (CMP_NV) CR1NV[6]:** Provides the default state for the CMP bit in CR1V[6].

**Security Region Lock bits (LB3, LB2, LB1, LB0) CR1NV[5:2]:** Provide the OTP write protection control of the Security Regions. When an LB bit is set to 1 the related Security Region can no longer be programmed or erased.

**Quad Data Width Non-volatile (QUAD_NV) CR1NV[1]**: Provides the default state for the quad bit in CR1V[1]. The WRR or WRAR command affects this bit. Programming CR1NV[1] = 1 will default operation to allow quad-data-width commands at power-on or reset.

**Status Register Protect 1 Default (SRP1_D) CR1NV[0]**: Provides the default state for the SRP1 bit in CR1V[0]. When IRP[2:0] = '111' the SRP1_D OTP bit is user programmable. When SRP1_D = '1' Registers SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are permanently locked. See **"Status Register Protect (SRP1, SRP0)"** on page 50.

### 6.6.3.2 Configuration Register 1 Volatile (CR1V)

Related Commands: Read Configuration Register 1 (RDCR1 35h), Write Enable for Volatile (WRENV 50h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). This is the register displayed by the RDCR1 commands.

**Table 11 Configuration Register 1 Volatile (CR1V)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | SUS | Suspend Status | Volatile read only | CR1NV | 1 = Erase / program suspended<br>0 = Erase / program not suspended |
| 6 | CMP | Complement Protection | Volatile | | 0 = Normal protection map<br>1 = Inverted protection map |
| 5 | LB3 | Volatile Copy of Security Region Lock bits | Volatile read only | | Not user writable<br>See CR1NV[5:2]<br>OTP lock bits 3:0 for Security Regions 3:0<br>0 = Security Region not locked<br>1 = Security Region permanently locked |
| 4 | LB2 | | | | |
| 3 | LB1 | | | | |
| 2 | LB0 | | | | |
| 1 | QUAD | Quad I/O mode | Volatile | | 1 = Quad<br>0 = Dual or serial |
| 0 | SRP1 | Status Register Protect 1 | Volatile | | Lock current state of SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV<br>1 = Registers locked<br>0 = Registers un-locked |

**Suspend Status (SUS) CR1V[7]**: The Suspend Status bit is used to determine when the device is in Erase or Program Suspend mode. This is a status bit that cannot be written by the user. When Suspend Status bit is set to '1', the device is in Erase or Program Suspend mode. When Suspend Status bit is cleared to '0', the device is not in Erase or Program Suspend mode. Refer to **"Program or Erase Suspend (PES 75h)"** on page 106 for details about the Erase/Program Suspend/Resume commands. Complement protection (CMP) CR1V[6]: CMP is used in conjunction with TBPROT, BP3, BP2, BP1 and BP0 bits to provide more flexibility for the array protection map, to protect from 1/2 to all of the array.

**LB[3:0] CR1V[5:2]:** These bits are volatile copies of the related OTP bits of CR1NV. These bits track any changes to the related OTP version of these bits.

**Quad Data Width (QUAD) CR1V[1]**: When set to '1', this bit switches the data width of the device to 4-bit - Quad mode. That is, WP# becomes IO2 and IO3 / RESET# becomes an active I/O signal when CS# is LOW or the RESET# input when CS# is HIGH. The WP# input is not monitored for its normal function and is internally set to HIGH (inactive). The commands for Serial, and Dual I/O Read still function normally but, there is no need to drive the WP# input for those commands when switching between commands using different data path widths. Similarly, there is no requirement to drive the IO3 / RESET# during those commands (while CS# is LOW). The Quad bit must be set to one when using the Quad Output Read, Quad I/O Read, DDR Quad I/O Read. The Volatile Register Write for QIO mode has a short and well defined time ($t_{QEN}$) to switch the device interface into QIO mode and ($t_{QEX}$) to switch the device back to SPI mode. Following commands can then be immediately sent in QIO protocol. While QPI mode is entered or exited by the QPIEN and QPIEX commands, or by setting the CR2V[3] bit to 1, the Quad Data Width mode is in use whether the QUAD bit is set or not.

**Status Register Protect 1(SRP1) CR1V[0]**: The SRP1 bit, when set to '1', protects the current state of the SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV registers by preventing any write of these registers. See **"Status Register Protect (SRP1, SRP0)"** on page 50.

As long as the SRP1 bit remains cleared to logic 0 the SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV, and DLRV registers are not protected by SRP1. However, these registers may be protected by SRP0 (SR1V[7]) and the WP# input.

Once the SRP1 bit has been written to a logic 1 it can only be cleared to a logic 0 by a power-off to power-on cycle or a hardware reset. Software reset will not affect the state of the SRP1 bit.

The CR1V[0] SRP1 bit is volatile and the default state of SRP1 after power-on comes from SRP1_D in CR1NV[0]. The SRP1 bit can be set in parallel with updating other values in CR1V by a single WRR or WRAR command.

## 6.6.4 Configuration Register 2

Configuration Register 2 controls certain interface functions. The register bits can be read and changed using the Read Any Register and Write Any Register commands. The non-volatile version of the register provides the ability to set the POR, hardware reset, or software reset state of the controls. The volatile version of the register controls the feature behavior during normal operation.

### 6.6.4.1 Configuration Register 2 Non-volatile (CR2NV)

Related Commands: Non-volatile Write Enable (WREN 06h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 12 Configuration Register 2 Non-volatile (CR2NV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | IO3R_NV | IO3_Reset | Non-volatile | 0 | 1 = Enabled -- IO3_RESET is used as IO3 / RESET# input when CS# is HIGH or Quad mode is disabled CR1V[1] = 0 or QPI is disabled (CR3V[3] = 0)<br>0 = Disabled -- IO3 has no alternate function, hardware reset is disabled. Provides the default state for the IO3 / RESET# function enable |
| 6 | OI_NV | Output Impedance | | 1 | Provides the default output impedance state See **Table 13**. |
| 5 | | | | 1 | |
| 4 | RFU | Reserved | | 0 | Reserved for Future Use |
| 3 | QPI_NV | QPI | | 0 | 1 = Enabled -- QPI (4-4-4) protocol in use<br>0 = Disabled -- legacy SPI protocols in use, instruction is always serial on SI<br>Provides the default state for QPI mode. |
| 2 | WPS_NV | Write Protect Selection | | 0 | Provides the default state for WPS<br>0 = Legacy protection<br>1 = Individual block lock |
| 1 | ADP_NV | Address Length at Power-up | | 0 | Provides the default state for address length<br>1 = 4-byte address<br>0 = 3-byte address |
| 0 | RFU | Reserved | | 0 | Reserved for Future Use |

**IO3 _Reset Non-volatile CR2NV[7]:** This bit controls the POR, hardware reset, or software reset state of the IO3 signal behavior. Most legacy SPI devices do not have a hardware reset input signal due to the limited signal count and connections available in traditional SPI device packages. The FL-L family provides the option to use the IO3 signal as a hardware reset input when the IO3 signal is not in use for transferring information between the host system and the memory. This non-volatile IO3_Reset Configuration bit enables the device to start immediately (boot) with IO3 enabled for use as a RESET# signal.

**Output Impedance Non-volatile CR2NV[6:5]:** These bits control the POR, hardware reset, or software reset state of the IO signal output impedance (drive strength). Multiple drive strength are available to help match the output impedance with the system printed circuit board environment to minimize overshoot and ringing. These Non-Volatile Output Impedance Configuration bits enable the device to start immediately (boot) with the appropriate drive strength.

**Table 13 Output impedance control**

| CR2NV[6:5] impedance selection | Typical impedance to $V_{SS}$ (Ω) | Typical impedance to $V_{CC}$ (Ω) | Notes |
|---|---|---|---|
| 00 | 18 | 21 | – |
| 01 | 26 | 28 | – |
| 10 | 47 | 45 | – |
| 11 | 71 | 64 | Factory default |

**QPI Non-volatile CR2NV[3]:** This bit controls the POR, hardware reset, or software reset state of the expected instruction width for all commands. Legacy SPI commands always send the instruction one bit wide (serial I/O) on the SI (IO0) signal. The FL-L family also supports the QPI mode in which all transfers between the host system and memory are 4 bits wide on IO0 to IO3, including all instructions. This Non-volatile QPI Configuration bit enables the device to start immediately (boot) in QPI mode rather than the Legacy Serial Instruction mode. The recommended procedure for moving to QPI mode is to first use the QPIEN (38h) command, the WRR or WRAR command can also set CR2V[3] = 1, QPI mode. The Volatile Register Write for QPI mode has a short and well defined time ($t_{QEN}$) to switch the device interface into QPI mode and ($t_{QEX}$) to switch the device back to SPI mode Following commands can then be immediately sent in QPI protocol. The WRAR command can be used to program CR2NV[3] = 1, followed by polling of SR1V[0] to know when the programming operation is completed. Similarly, to exit QPI mode use the QPIEX (F5h) command. The WRR or WRAR command can also be used to clear CR2V[3] = 0.

**Write Protect Selection Non-volatile CR2NV[2]:** This bit controls the POR, hardware reset, or software reset state of the write protect method. This Non-volatile Configuration bit enables the device to start immediately (boot) with individual block lock protection rather than legacy block protection.

**Address Length at Power-up Non-volatile CR2NV[1]:** This bit controls the POR, hardware reset, or software reset state of the expected address length for all commands that require address and are not fixed 3-byte or 4-byte only address. Most commands that need an address are Legacy SPI commands that traditionally used 3-byte (24- bit) address. For device densities greater than 128 Mb a 4-byte (32-bit) address is required to access the entire memory array. The Address Length Configuration bit is used to change all 3-byte address commands to expect 4-byte address. See **Table 33** for command address length. This Non-volatile Address Length Configuration bit enables the device to start immediately (boot) in 4-byte Address mode rather than the legacy 3-byte Address mode.

### 6.6.4.2 Configuration Register 2 Volatile (CR2V)

Related commands: Read Configuration Register 2 (RDCR2 15h), Read Any Register (RDAR 65h), Write Enable for Volatile (WRENV 50h), Write Register (WRR 01h), Write Any Register (WRAR 71h), Enter 4-byte Address mode (4BEN B7h), Exit 4-byte Address mode (4BEX E9h), Enter QPI (38h), Exit QPI (F5h). This is the register displayed by the RDCR2 command.

**Table 14 Configuration Register 2 Volatile (CR2V)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | IO3R | IO3_Reset | Volatile | CR2NV | 1 = Enabled -- IO3 is used as RESET# input when CS# is HIGH or Quad mode is disabled CR1V[1] = 0 or QPI is disabled (CR3V[3] = 0)<br>0 = Disabled -- IO3 has no alternate function, hardware reset through IO3 / RESET# input is disabled |
| 6 | OI | Output Impedance | | | See **Table 13**. |
| 5 | | | | | |
| 4 | RFU | Reserved | | | Reserved for Future Use |
| 3 | QPI | QPI | | | 1 = Enabled -- QPI (4-4-4) protocol in use<br>0 = Disabled -- legacy SPI protocols in use, instruction is always serial on SI |
| 2 | WPS | Write Protect Selection | | | 0 = Legacy block protection<br>1 = Individual block lock |
| 1 | ADP | Address Length at Power-up | Volatile read only | | Read Status Only bit<br>1 = 4-byte address<br>0 = 3-byte address |
| 0 | ADS | Address Length Status | Volatile | CR2NV[1] | Current Address mode<br>1 = 4-byte address<br>0 = 3-byte address |

**IO3 Reset CR2V[7]:** This bit controls the IO3 / RESET# signal behavior. This Volatile IO3 Reset Configuration bit enables the use of IO3 as a RESET# input during normal operation when CS# is HIGH or Quad mode is disabled (CR1V[1] = 0) or QPI is disabled (CR3V[3] = 0).

**Output Impedance CR2V[6:5]:** These bits control the IO signal output impedance (drive strength). This Volatile Output Impedance Configuration bit enables the user to adjust the drive strength during normal operation.

**QPI CR2V[3]:** This bit controls the expected instruction width for all commands. This Volatile QPI Configuration bit enables the device to Enter and Exit QPI mode during normal operation. When this bit is set to QPI mode, the Quad mode is active, independent of the setting of QIO mode (CR1V[1]). When this bit is cleared to Legacy SPI mode, the Quad bit is not affected. The QPI CR2V[3] bit can also be set to ‘1’ by the QPIEN (38h) command and set to ‘0’ by the QPIEX (F5h) command.

**Table 15 QPI and QIO Mode Control bits**

| QPI CR2V[3] | QUAD CR1V[1] | Description |
|---|---|---|
| 0 | 0 | SIO mode: Single and Dual Read, WP#/IO2 input is in use as WP# pin and IO3 / RESET# input is in use as RESET# pin |
| 0 | 1 | QIO mode: Single, Dual, and Quad Read, WP#/IO2 input is in use as IO2 and IO3 / RESET# input is in use as IO3 or RESET# pin |
| 1 | X | QPI mode: Quad Read, WP#/IO2 input is in use as IO2 and IO3 / RESET# input is in use as IO3 or RESET# pin |

**Write Protect Selection CR2V[2]:** This bit selects which array protection method is used; **"Legacy block protection"** on page 53) or **"Individual block lock (IBL) protection"** on page 55. These Volatile Configuration bits enable the user to change protection method during normal operation.

**Address Length at Power-on (ADP) CR2V[1]:** This bit is read only and shows what the address length will be after power-on reset, hardware reset, or software reset for all commands that require address and are not fixed 3-byte or 4-byte address.

**Address Length Status (ADS) CR2V[0]:** This bit controls the expected address length for all commands that require address and are not fixed 3-byte or 4-byte address. See **Table 33** for command address length. This Volatile Address Length Configuration bit enables the address length to be changed during normal operation. The 4-byte Address mode (4BEN) command directly sets this bit into 4-byte Address mode and the (4BEX) command exits sets this bit back into 3-byte Address mode. This bit is also updated when the address length non-volatile CR2NV[1] bit is updated.

## 6.6.5 Configuration Register 3

Configuration Register 3 controls the main flash array read commands burst wrap behavior and read latency. The burst wrap configuration does not affect commands reading from areas other than the main flash array e.g. Read commands for registers or Security Regions. The non-volatile version of the register provides the ability to set the start up (boot) state of the controls as the contents are copied to the volatile version of the register during the POR, hardware reset, or software reset. The volatile version of the register controls the feature behavior during normal operation.

The register bits can be read and changed using the, Read Configuration 3 (RDCR3 33h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). The volatile version of the register can also be written by the Set Burst Length (77h) command.

### 6.6.5.1 Configuration Register 3 Non-volatile (CR3NV)

Related commands: Non-volatile Write Enable (WREN 06h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

**Table 16 Configuration Register 3 Non-volatile (CR3NV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | Non-volatile | 0 | Reserved for Future Use |
| 6 | WL_NV | Wrap Length Default | | 1 | 00 = 8-byte wrap<br>01 = 16-byte wrap<br>10 = 32-byte wrap<br>11 = 64-byte wrap |
| 5 | | | | 1 | |
| 4 | WE_NV | Wrap Enable Default | | 1 | 0 = Wrap enabled<br>1 = Wrap disabled |
| 3 | RL_NV | Read Latency Default | | 1 | 0 to 15 latency (dummy) cycles following Read Address or Continuous Mode bits. |
| 2 | | | | 0 | |
| 1 | | | | 0 | |
| 0 | | | | 0 | |

**Wrap Length Non-volatile CR3NV[6:5]:** These bits controls the POR, hardware reset, or software reset state of the wrapped read length and alignment.

**Wrap Enable Non-volatile CR3NV[4]:** This bit controls the POR, hardware reset, or software reset state of the wrap enable. The commands affected by Wrap Enable are: Quad I/O Read, QPI Read, DDR Quad I/O Read and DDR QPI Read. This configuration bit enables the device to start immediately (boot) in Wrapped Burst Read mode rather than the Legacy Sequential Read mode.

**Read Latency Non-volatile CR3NV[3:0]:** These bits control the POR, hardware reset, or software reset state of the read latency (dummy cycle) delay in all variable latency read commands. The following read commands have a variable latency period between the end of address or mode and the beginning of read data returning to the host:

- The latency delay per clock frequency for the following commands are: One dummy cycle for all clock frequency's. The default latency code of '0' is one dummy cycle.
  - Data Learning Pattern Read DLPRD (1-1-1) or (4-4-4)
  - IRP Read IRPRD (1-1-1) or (4-4-4))
  - Protect Register Read PRRD (1-1-1) or (4-4-4)
  - Password read PASSRD (1-1-1) or (4-4-4)
- The latency delay per clock frequency for the following commands are shown in **Table 17** and **Table 18** below. The default latency code of '0' is 8 dummy cycles.
  - Fast Read FAST_READ (1-1-1)
  - Quad-O Read QOR, 4QOR (1-1-4)
  - Dual-O Read DOR, 4DOR (1-1-2)
  - Dual I/O Read DIOR, 4DIOR (1-2-2)
  - Quad I/O Read QIOR, 4QIOR (1-4-4) or (4-4-4)
  - DDR Quad I/O Read DDRQIOR, 4DDRQIOR(1-4-4)
  - Security Regions Read SECRR (1-1-1) or (4-4-4)
  - Read Any Register RDAR (1-1-1) or (4-4-4)
  - Read serial flash discoverable parameters RSFDP (1-1-1) or (4-4-4)

The non-volatile read latency configuration bits set the number of read latency (dummy cycles) in use so the device can start immediately (boot) with an appropriate read latency for the host system.

**Table 17 Latency code (cycles) versus frequency**

| Latency code 0 | Read command maximum frequency (MHz) | | | | | | |
|---|---|---|---|---|---|---|---|
| | Fast Read (1-1-1) | Dual-O Read (1-1-2) | Dual I/O Read (1-2-2) | Quad-O Read (1-1-4) | Quad I/O Read (1-4-4) | Quad I/O Read QPI (4-4-4) | DDR Quad I/O (1-4-4) QPI (4-4-4) |
| | Mode cycles = 0 | Mode cycles = 0 | Mode cycles = 4 | Mode cycles = 0 | Mode cycles = 2 | Mode cycles = 2 | Mode cycles = 1 |
| | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 |
| 1 | 50 | 50 | 75 | 35 | 35 | 35 | 20 |
| 2 | 65 | 65 | 85 | 45 | 45 | 45 | 25 |
| 3 | 75 | 75 | 95 | 55 | 55 | 55 | 35 |
| 4 | 85 | 85 | 108 | 65 | 65 | 65 | 45 |
| 5 | 95 | 95 | 108 | 75 | 75 | 75 | 54 |
| 6 | 108 | 105 | 108 | 85 | 85 | 85 | 54 |
| 7 | 108 | 108 | 108 | 95 | 95 | 95 | 54 |
| 8 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 9 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 10 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 11 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 12 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 13 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 14 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |
| 15 | 108 | 108 | 108 | 108 | 108 | 108 | 54 |

**Table 18 Latency code (cycles) versus frequency**

| Latency code 0 | Read Command Maximum Frequency (MHz) | | | | | |
|---|---|---|---|---|---|---|
| | Read Any Register (1-1-1) | Read Any Register QPI (4-4-4) | Security region read (1-1-1) | Security region read QPI (4-4-4) | Read SFDP RSFDP (1-1-1) | Read SFDP RSFDP QPI (4-4-4) |
| | Mode cycles = 0 | Mode cycles = 0 | Mode cycles = 0 | Mode cycles = 0 | Mode cycles = 0 | Mode cycles = 0 |
| | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 | Dummy cycles = 8 |
| 1 | 50 | 15 | 50 | 15 | 50 | 15 |
| 2 | 65 | 25 | 65 | 25 | 65 | 25 |
| 3 | 75 | 35 | 75 | 35 | 75 | 35 |
| 4 | 85 | 45 | 85 | 45 | 85 | 45 |
| 5 | 95 | 55 | 95 | 55 | 95 | 55 |
| 6 | 108 | 65 | 108 | 65 | 108 | 65 |
| 7 | 108 | 75 | 108 | 75 | 108 | 75 |
| 8 | 108 | 85 | 108 | 85 | 108 | 85 |
| 9 | 108 | 95 | 108 | 95 | 108 | 95 |
| 10 | 108 | 108 | 108 | 108 | 108 | 108 |
| 11 | 108 | 108 | 108 | 108 | 108 | 108 |
| 12 | 108 | 108 | 108 | 108 | 108 | 108 |
| 13 | 108 | 108 | 108 | 108 | 108 | 108 |
| 14 | 108 | 108 | 108 | 108 | 108 | 108 |
| 15 | 108 | 108 | 108 | 108 | 108 | 108 |

**Notes**

7. SCK frequency > 108 MHz SDR, or 54 MHz DDR is not supported by these devices.
8. The Dual I/O, Quad I/O, QPI, DDR Quad I/O, and DDR QPI command protocols include Continuous Read Mode bits following the address. The clock cycles for these bits are not counted as part of the latency cycles shown in the **Table 18**. Example: the Legacy Quad I/O command has 2 Continuous Read mode cycles following the address. Therefore, the legacy quad I/O command without additional read latency is supported only up to the frequency shown in the table for a read latency of 0 cycles. By increasing the variable read latency the frequency of the Quad I/O command can be increased to allow operation up to the maximum supported 108 MHz frequency and QPI maximum supported 108 MHz.
9. Other commands have fixed latency. For example, Read always has zero read latency, read unique ID has 32 dummy cycles and release from deep power-down has 24 dummy cycles.

### 6.6.5.2 Configuration Register 3 Volatile (CR3V)

Related commands: Read Configuration 3 (RDCR3 33h), Write Enable for Volatile (WRENV 50h), Write Registers (WRR 01h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h), Set Burst Length (SBL 77h). This is the register displayed by the RDCR3 command.

**Table 19 Configuration Register 3 Volatile (CR3V)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | Volatile | CR3NV | Reserved for Future Use |
| 6 | WL | Wrap Length | | | 00 = 8-byte wrap<br>01 = 16-byte wrap<br>10 = 32-byte wrap<br>11 = 64-byte wrap |
| 5 | | | | | |
| 4 | WE | Wrap Enable | | | 0 = Wrap enabled<br>1 = Wrap disabled |
| 3 | RL | Read Latency | | | 0 to 15 latency (dummy) cycles following Read Address or Continuous Mode bits. |
| 2 | | | | | |
| 1 | | | | | |
| 0 | | | | | |

**Wrap Length CR3V[6:5]:** These bits controls the wrapped read length and alignment during normal operation. These Volatile Configuration bits enable the user to adjust the burst wrapped read length during normal operation.

**Wrap Enable CR3V[4]:** This bit controls the burst wrap feature. This Volatile Configuration bit enables the device to Enter and Exit Burst Wrapped Read mode during normal operation. When CR3V[4] = 1, the Wrap mode is not enabled and unlimited length sequential read is performed. When CR3V[4] = 0, the Wrap mode is enabled and a fixed length and aligned group of 8-, 16-, 32-, or 64-bytes is read starting at the byte address provided by the Read command and wrapping around at the group alignment boundary.

**Read Latency CR3V[3:0]:** These bits set the read latency (dummy cycle) delay in Variable Latency Read commands. These volatile configuration bits enable the user to adjust the read latency during normal operation to optimize the latency for different commands or, at different operating frequencies, as needed.

### 6.6.6 Individual and Region Protection Register (IRP)

Related commands: IRP Read (IRPRD 2Bh) and IRP Program (IRPP 2Fh), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

The IRP Register is a 16-bit OTP memory location used to permanently configure the behavior of individual and region protection (IRP) features. IRP does not have User Programmable Volatile bits, all defined bits are OTP.

The default state of the IRP bits are programmed by Infineon.

**Table 20 IRP Register (IRP)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 15 to 7 | RFU | Reserved | OTP | All bits are 1 | Reserved for Future Use |
| 6 | SECRRP | Security Region 3 Read Password mode enable bit | OTP | 1 | 0 = Security Region 3 Read Password mode selected<br>1 = Security Region 3 Read Password not selected<br>IRP[6] is programmable if IRP[2:0] = '111' |
| 5 | RFU | Reserved | OTP | 1 | Reserved for Future Use |
| 4 | IBLLBB | IBL Lock Boot bit | OTP | 1 | 0 = All individual IBL bits are set to '1' at power-up in the unprotected state<br>1 = All individual IBL bits are set to '0' at power-up in the protected state<br>IRP[4] is programmable if IRP[2:0] = '111' |
| 3 | RFU | Reserved | OTP | 1 | Reserved for future use |
| 2 | PWDMLB | Password Protection Mode Lock bit | OTP | 1 | 0 = Password Protection mode permanently enabled.<br>1 = Password Protection mode not permanently enabled.<br>IRP[2] is programmable if IRP[2:0] = '111' |
| 1 | PSLMLB | Power Supply Lock-Down Protection Mode Lock bit | OTP | 1 | 0 = Power Supply Lock-Down Protection mode permanently enabled.<br>1 = Power Supply Lock-Down Protection mode not permanently enabled.<br>IRP[1] is programmable if this is enabled by IRP[2:0] = '111' |
| 0 | PERMLB | Permanent Protection lock | OTP | 1 | 0 = Permanent Protection mode permanently enabled.<br>1 = Permanent Protection mode not permanently enabled.<br>IRP[0] is programmable if IRP[2:0] = '111' |

**Security Regions Read Password Mode Enable (SECRRP) IRP[6]:** When programmed to '0', SECRRP enables the Security Region 3 Read Password mode when PWDMLB bit IRP[2] is program at same time or later. The SECRRP bit can only be programmed when IRP[2:0] = '111', if not programming will fail with P_ERR set to '1'. See **"Security Region read password protection"** on page 61.

**IBL Lock Boot bit (IBLLBB) IRP[4]:** The default state is 1, all individual IBL bits are set to '0' in the protected state, following power-up, hardware reset, or software reset. In order to Program or Erase the Array the Global IBL Unlock or the Sector / Block IBL Unlock command must be given before the Program or Erase commands. When programmed to 0, all the individual IBL bits are in the un-protected state following power-up, hardware reset, or software reset. The IBLLBB bit can only be programmed when IRP[2:0] = '111', if not programming will fail with P_ERR set to '1'. See **"Individual block lock (IBL) protection"** on page 55.

**Password Protection Mode Lock bit (PWDMLB) IRP[2]:** When programmed to '0', the Password Protection mode is permanently selected to protect the Security Regions 2 and 3 and pointer region. The PWDMLB bit can only be programmed when IRP[2:0] = '111', if not programming will fail with P_ERR set to 1. See **"Password Protection mode"** on page 60.

After the Password Protection mode is selected by programming IRP[2] = '0', the state of all IRP bits are locked and permanently protected from further programming. Attempting to program any IRP bits will result in a programming error with P_ERR set to 1.

The password must be programmed and verified, before the Password mode (IRP[2] = 0) is set.

**Power Supply Lock-Down Protection Mode Lock bit (PSLMLB) IRP[1]:** When programmed to 0, the Power Supply Lock-down Protection mode is permanently selected. The PSLMLB bit can only be programmed when IRP[2:0] = '111', if not programming will fail with P_ERR set to '1'.

After the Power Supply Lock-down Protection mode is selected by programming IRP[1] = '0', the state of all IRP bits are locked and permanently protected from further programming. Attempting to program any IRP bits will result in a programming error with P_ERR set to '1'. See **"IRP Register"** on page 59.

**Permanent Protection Lock bit (PERMLB) IRP[0]:** When programmed to 0, the permanent Protection Lock bit permanently protects the Pointer Region and Security Regions 2 and 3, This bit provides a simple way to permanently protect the Pointer Region and Security Regions 2 and 3 without the use of a password or the PRL command. See **"IRP Register"** on page 59.

PWDMLB (IRP[2]), PSLMLB (IRP[1]) and PERMLB(IRP[0]) are mutually exclusive, only one may be programmed to zero. IRP bits may only be programmed while IRP[2:0] = '111'. Attempting to program IRP bits when IRP[2:0] is not = '111' will result in a programming error with P_ERR set to '1'. The IRP Protection mode should be selected during system configuration to ensure that a malicious program does not select an undesired Protection mode at a later time. By locking all the protection configuration via the IRP mode selection, later alteration of the protection methods by malicious programs is prevented.

### 6.6.7 Password Register (PASS)

Related commands: Password Read (PASSRD E7h) and Password Program (PASSP E8h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h). The PASS register is a 64-bit OTP memory location used to permanently define a password for the Individual and region protection (IRP) feature. PASS does not have user programmable volatile bits, all defined bits are OTP. A volatile copy of PASS is used to satisfy read latency requirements but the volatile register is not user writable or further described. The Password can not be read or programmed after IRP[2] is programmed to '0'. See **Table 20**.

**Table 21 Password Register (PASS)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 63 to 0 | PWD | Hidden password | OTP | FFFFFFFF–FFFFFFFFh | Non-volatile OTP storage of 64-bit password. The password is no longer readable after the Password Protection mode is selected by programming IRP register bit 2 to zero. |

### 6.6.8 Protection Register (PR)

Related commands: Protection Register Read (PRRD A7h) Protection Register Lock (PRL A6h), Read Any Register (RDAR 65h).

PR does not have separate user programmable non-volatile bits, all defined bits are volatile read only status. The default state of the RFU bits is set by hardware. There is no non-volatile version of the PR register.

The NVLOCK bit is used to protect the Security Regions 2 and 3 and pointer region protection. When NVLOCK[0] = 0, the Security Regions 2 and 3 and pointer region protection can not be changed.

**Table 22 Protection Status Register (PR)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 | RFU | Reserved | Volatile read only | 00h | Reserved for Future Use |
| 6 | SECRRP | Security Regions Read Password | | IRP[6] | 0 = Security Region 3 password protected from read when NVLOCK = 0<br>1 = Security Region 3 not password protected from read |
| 5 | RFU | Reserved | | 0 | Reserved for Future Use |
| 4 | RFU | Reserved | | 0 | Reserved for Future Use |
| 3 | RFU | Reserved | | 0 | Reserved for Future Use |
| 2 | RFU | Reserved | | 0 | Reserved for Future Use |
| 1 | RFU | Reserved | | 0 | Reserved for Future Use |
| 0 | NVLOCK[10] | Protect Non-volatile Configuration | | IRP[2] and IRP[0] | 0 = Security Regions 2 and 3 and pointer region write protected<br>1 = Security Regions 2 and 3 and pointer region may be written |

**Note**
10.The Command Protection Register Lock (PRL), sets the NVLOCK = 1.

### 6.6.9 Individual Block Lock Access Register (IBLAR)

Related commands: IBL Read (IBLRD 3Dh or 4IBLRD E0h), IBL Lock (IBL 36h or 4IBL E1h), IBL Unlock (IBLUL 39h or 4IBUL E2h), Global IBL lock (GBL 7Eh), Global IBL Unlock (GBUL 98h).

IBLAR does not have user programmable non-volatile bits, all bits are a representation of the volatile bits in the IBL array. The default state of the IBL array bits is set by hardware. There is no non-volatile version of the IBLAR register.

**Table 23 IBL Access Register (IBLAR)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 to 0 | IBL | Read or Write IBL for Individual Sectors / Blocks | Volatile | IRP[4] = 1 then 00h else FFh | 00h = IBL for the sector / block addressed is set to ‘0’ by the IBL, 4IBL and GBL commands protecting that sector from program or erase operations.<br>FFh = IBL for the sector / block addressed is cleared to ‘1’ by the IBUL, 4IBUL and GBUL commands not protecting that sector from program or erase operations. |

**Notes**
11.See **Figure 25**.
12.The IBL bits maybe read by the IBLRD and 4IBLRD commands.

### 6.6.10 Pointer Region Protection Register (PRPR)

Related commands: Set Pointer Region (SPRP FBh or 4SPRP E3h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

PRPR contains user programmable non-volatile bits. The default state of the PRPR bits is set by hardware. There is no volatile version of the PRPR register. See **"Pointer region protection (PRP)"** on page 56 for additional details.

**Table 24 PRP Register (PRPR)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| A31 to A23 | RFU | Reserved | Non-volatile | 11111111b | Reserved for Future Use |
| A22 to A16 | PRPAD | PRP address | | FFh | Pointer address A22 to A16 |
| A15 to A12 | – | – | | Fh | Pointer address A15 to A12 |
| A11 | PRPALL | PRP Protect All | | 1 | 0 = Protect pointer region selected sectors<br>1 = Protect all sectors |
| A10 | PRPEN | PRP Enable | | 1 | 0 = Enable pointer region protection<br>1 = Disable pointer region protection |
| A9 | PRPTB | PRP Top/ Bottom | | 1 | 0 = Pointer region protection starts from the top (high address)<br>1 = Pointer region protection starts from the bottom (low address) |
| A8 | RFU | Reserved | | 1 | Reserved for Future Use |
| A7 to A0 | RFU | Reserved | | FFh | Reserved for Future Use |

### 6.6.11 DDR Data Learning Registers

Related commands: Program DLRNV (PDLRNV 43h), Write DLRV (WDLRV 4Ah), Data Learning Pattern Read (DLPRD 41h), Read Any Register (RDAR 65h), Write Any Register (WRAR 71h).

The Data Learning Pattern (DLP) resides in an 8-bit Non-volatile Data Learning Register (DLRNV) as well as an 8-bit Volatile Data Learning Register (DLRV). When shipped from Infineon, the DLRNV value is 00h. Once programmed, the DLRNV cannot be reprogrammed or erased; a copy of the data pattern in the DLRNV will also be written to the DLRV. The DLRV can be written to at any time, but on hardware and software reset or power cycles the data pattern will revert back to what is in the DLRNV. During the learning phase described in the SPI DDR modes, the DLP will come from the DLRV. Each IO will output the same DLP value for every clock edge. For example, if the DLP is 34h (or binary 00110100) then during the first clock edge all IO's will output 0; subsequently, the 2nd clock edge all I/O's will output 0, the 3rd will output 1, etc.

When the DLRV value is 00h, no preamble data pattern is presented during the dummy phase in the DDR commands.

**Table 25 Non-volatile Data Learning Register (DLRNV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 to 0 | NVDLP | Non-volatile Data Learning Pattern | OTP | 00h | OTP value that may be transferred to the host during DDR Read command latency (dummy) cycles to provide a training pattern to help the host more accurately center the data capture point in the received data bits. |

**Table 26 Volatile Data Learning Register (DLRV)**

| Bits | Field name | Function | Type | Default state | Description |
|---|---|---|---|---|---|
| 7 to 0 | VDLP | Volatile Data Learning Pattern | Volatile | Takes the value of DLRNV during POR or Reset | Volatile copy of the NVDLP used to enable and deliver the data learning pattern (DLP) to the outputs. The VDLP may be changed by the host during system operation. |

# 7 Data protection

## 7.1 Security Regions

The device has a 1024-byte address space that is separate from the main flash array. This area is divided into 4, individually lockable, 256-byte length regions. See **“Security Regions address space”** on page 28.

The Security Region memory space is intended for increased system security. The data values can “mate” a flash component with the system CPU/ASIC to prevent device substitution. The Security Region address space is protected by the Security Region Lock bits or the Protection Register NVLOCK bit (PR[0]). See **“Security Region Lock bits (LB3, LB2, LB1, LB0)”** on page 49.

### 7.1.1 Reading Security Region memory regions

The Security Region Read command (SECRR) uses the same protocol as Fast Read. Read operations outside the valid 1024-byte Security Region address range will yield indeterminate data. See **“Security Regions Read (SECRR 48h)”** on page 111.

Security Region 3 may be password protected from read by setting the PWDMLB bit IRP[2] = 0 and SECRRP bit IRP[6] = 0 when NVLOCK = 0.

### 7.1.2 Programming the Security Regions

The protocol of the Security Region programming command (SECRP) is the same as page program. See **“Security Region Program (SECRP 42h)”** on page 110.

The valid address range for Security Region program is depicted in **Table 5**. Security Region program operations outside the valid Security Region address range will be ignored, without P_ERR in SR2V[5] set to ‘1’.

Security Regions 2 and 3 may be password protected from programming by setting the PWDMLB bit IRP[2] = 0.

### 7.1.3 Erasing the Security Regions

The protocol of the Security Region Erasing command (SECRE) is the same as sector erase. See **“Security Region Erase (SECRE 44h)”** on page 110.

The valid address range for Security Region Erase is depicted in **Table 5**. Security Region erase operations outside the valid Security Region address range will be ignored, without E_ERR in SR2V set to ‘1’.

Security Regions 2 and 3 may be password protected from erasing by setting the PWDMLB bit IRP[2] = 0.

### 7.1.4 Security Region Lock bits (LB3, LB2, LB1, LB0)

The Security Region lock bits (LB3, LB2, LB1, LB0) are Non-volatile One Time Program (OTP) bits in Configuration Register 1(CR1NV[5:2]) that provide the write protect control and status to the Security Regions. The default state of Security Regions 0 to 3 are unlocked. LB[3:0] can be set to 1 individually using the Write Status Registers or Write Any Register command. LB[3:0] are one time programmable (OTP), once it’s set to ‘1’, the corresponding 256-byte Security Region will become read-only permanently.

## 7.2 Deep Power Down

The Deep Power Down (DPD) command offers an alternative means of data protection as all commands are ignored during the DPD state, except for the release from Deep Power Down (RES ABh) command and hardware reset. Thus, preventing any program or erase during the DPD state.

## 7.3 Write Enable commands

### 7.3.1 Write Enable (WREN)

The Write Enable (WREN) command must be written prior to any command that modifies non-volatile data. The WREN command sets the Write Enable Latch (WEL) bit. The WEL bit is cleared to 0 (disables writes) during power-up, hardware and software reset, or after the device completes the following commands:

- Reset
- Page Program (PP or 4PP)
- Quad Page Program (QPP or 4QPP)
- Sector Erase (SE or 4SE)
- Half Block Erase (HBE or 4HBE)
- Block Erase (BE or 4BE)
- Chip Erase (CE)
- Write Disable (WRDI)
- Write Registers (WRR)
- Write Any Register (WRAR)
- Security Region Erase (SECRE)
- Security Region Byte Programming (SECRP)
- Individual and Region Protection Register Program (IRPP)
- Password Program (PASSP)
- Clear Status Register (CLSR)
- Set Pointer Region Protection (SPRP or 4SPRP)
- Program Non-volatile Data Learning Register (PDLRNV)
- Write Volatile Data Learning Register (WDLRV)

### 7.3.2 Write Enable for Volatile Registers (WRENV)

The Write Enable Volatile (WRENV) command must be written prior to Write Register (WRR) command that modifies volatile registers data.

## 7.4 Write Protect signal

When not in Quad mode (CR1V[1] = 0) or QPI mode (CR2V[3] = 0), the Write Protect (WP#) input in combination with the Status Register Protect 0 (SRP0) bit (SR1NV[7]) provide hardware input signal controlled protection. When WP# is LOW and SRP0 is set to ‘1’ Status Register 1 (SR1NV and SR1V), Configuration register (CR1NV, CR1V, CR2NV, CR2V and CR3NV) and DDR Data Learning Registers (DLRNV and DLRV) are protected from alteration. This prevents disabling or changing the protection defined by the Legacy Block Protect bits or Security Region Lock bits. See **“Status Register 1”** on page 30.

## 7.5 Status Register Protect (SRP1, SRP0)

The Status Register Protect bits (SRP1 and SRP0) are volatile bits in the configuration and status registers (CR1V[0] and SR1V[7]). The SRP bits control the method of write protection for SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV: Software Protection, Hardware Protection, or Power Supply Lock-Down.

**Table 27 Status Register Protection bits (high security)**

| SRP1_D CR1NV[0] | SRP1 CR1V[0] | SRP0 SR1V[7] | WP# | Status Register | Description |
|---|---|---|---|---|---|
| 0 | 0 | 0 | X | Software Protection | WP# pin has no control. SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV can be written. [factory default] |
| 0 | 0 | 1 | 0 | Hardware Protected | When WP# pin is LOW SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are locked and can not be written[13, 16]. |
| 0 | 0 | 1 | 1 | Hardware Unprotected | When WP# pin is HIGH SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are unlocked and can be written[13]. |
| 0 | 1 | X | X | Power Supply Lock-Down | SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are protected and can not be written to again until the next power-down, power-up cycle[14]. |
| 1 | 1 | X | X | One Time Program | SRP1_D CR1NV[0] = 1 SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are permanently protected and can not be written[15]. |

**Notes**

13. SRP0 is reloaded from SRP0_NV (SR1NV[7]) default state after a power-down, power-up cycle, software or hardware reset. To enable Hardware Protection mode by the WP# pin at power-up set the SRP0_NV bit to '1'.
14. When SRP1 = 1, a power-down, power-up cycle, or hardware reset, will change SRP1 to 0 as SRP1 is reloaded from SRP1_D.
15. SRP1_D can be written only when IRP[2:0] ='111'. When SRP1_D CR1NV[0] = '1' a power-down, power-up cycle, or hardware reset, will reload SRP1 from SRP1_D = '1' the volatile bit SRP1 is not writable, thus providing OTP protection. When SRP1_D is programmed to 1, Recommended that SRP0_NV should also be programmed to 1 as an indication that OTP protection is in use.
16. When QPI or QIO mode is enabled (CR2V[3] or CR1V[1] = '1') the internal WP# signal level is = 1 because the WP# external input is used as IO2 when either mode is active. This effectively turns off hardware protection when SRP1-SRP0 = 01b. The Register SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV are unlocked and can be written.
17. WIP, WEL, and SUS (SR1[1:0] and CR1[7]) are Volatile Read Only Status bits that are never affected by the Write Status Registers command.
18. The non-volatile version of SR1NV, CR1NV, CR2NV and CR3NV are not writable when protected by the SRP bits and WP# as shown in the table. The non-volatile version of these Status Register bits are selected for writing when the Write Enable (06h) command precedes the Write Status Registers (01h) command or the Write Any Register (71h) command.
19. The volatile version of registers SR1V, CR1V and CR2V are not writable when protected by the SRP bits and WP# as shown in the **Table 27**. The volatile version of these Status Register bits are selected for writing when the Write Enable for Volatile Status Register (50h) command precedes the Write Status Registers (01h) command or the Write Enable (06h) command precedes the Write Any Register (71h) command.
20. The Volatile CR3V bits are not protected by the SRP bits and may be written at any time by volatile (50h) Write Enable command preceding the Write Status Registers (01h) command. The WRAR (71h) and SBL (77h) commands are alternative ways to Write bits in the CR3V register.
21. During system power up and boot code execution: Trusted boot code can determine whether there is any need to change SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV values. If no changes are needed the SRP1 bit (CR1V[0]) can be set to 1 to protect the SR1NV, SR1V, CR1NV, CR1V, CR2NV, CR2V, CR3NV, DLRNV and DLRV registers from changes during the remainder of normal system operation while power remains on.

## 7.6 Array protection

There are three types of memory array protection: Legacy block (LBP), individual block lock (IBL) and pointer region (PRP). The Write Protect Selection (WPS) bit is used by the user to enable one of two protection mechanisms: legacy block (LBP) protection (WPS CR2V[2] = 0)or individual block lock (IBL) protection (WPS CR2V[2] = 1). See **"Configuration Register 2 Volatile (CR2V)"** on page 38. Only one protection mechanism can be enabled at one time. The legacy block protection is the default protection and is mutually exclusive with the IBL protection scheme. The pointer region protection is enabled by the Set Pointer Region Protection command or the WRAR command by the value of A10 = 0. See **"Pointer Region command"** on page 117. When the pointer region protection is enabled it is logically ORed with the legacy block protection or individual block lock protection.

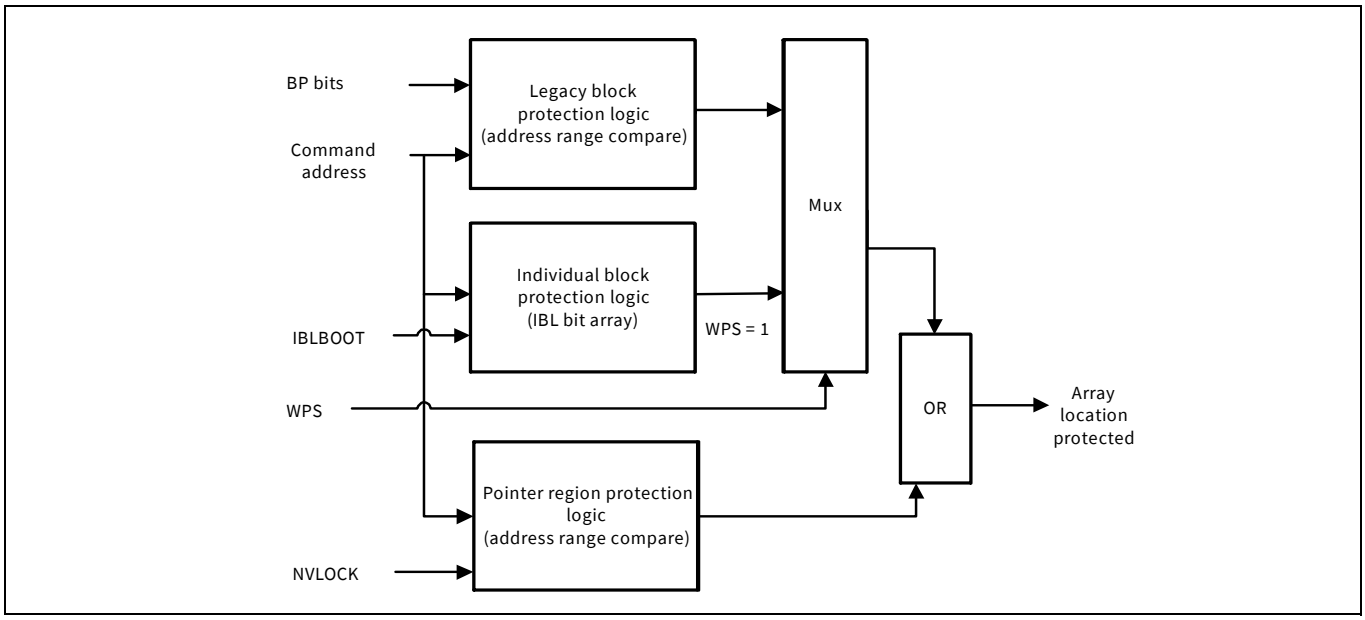



**Figure 24 WPS selection of LBP or IBL and PRP array protection**

### 7.6.1 Legacy block protection

The Legacy Block Protect bits Status Register bits BP2, BP1, BP0 -- SR1V[4:2]) in combination with the Configuration Register TBPROT (SR1V[5])bit, CMP (CR1V[6] bit and SEC (SR1V[6]) can be used to protect an address range of the main flash array from program and erase operations. The size of the range is determined by the value of the BP bits and the upper or lower starting point of the range is selected by the TBPROT bit of the configuration register (SR1V[5]). The protection is complemented when the CMP bit (CR1V[6]) is set to 1.

If the pointer region protection is enabled this region protection is logically ORed with the legacy block protection region.

**Table 28 S25FL064L legacy block protection (CMP = 0)**

| Status Register | | | | | 64 Mb block protection (CMP = 0) | | | |
|---|---|---|---|---|---|---|---|---|
| SEC | TBPROT | BP2 | BP1 | BP0 | Protected block(s) | Protected addresses | Protected density | Protected portion |
| X | X | 0 | 0 | 0 | None | None | None | None |
| 0 | 0 | 0 | 0 | 1 | 126 and 127 | 7E0000h–7FFFFFh | 128 kB | Upper 1/64 |
| 0 | 0 | 0 | 1 | 0 | 124 thru 127 | 7C0000h–7FFFFFh | 256 kB | Upper 1/32 |
| 0 | 0 | 0 | 1 | 1 | 120 thru 127 | 780000h–7FFFFFh | 512 kB | Upper 1/16 |
| 0 | 0 | 1 | 0 | 0 | 112 thru 127 | 700000h–7FFFFFh | 1 MB | Upper 1/8 |
| 0 | 0 | 1 | 0 | 1 | 96 thru 127 | 600000h–7FFFFFh | 2 MB | Upper 1/4 |
| 0 | 0 | 1 | 1 | 0 | 64 thru 127 | 400000h–7FFFFFh | 4 MB | Upper 1/2 |
| 0 | 1 | 0 | 0 | 1 | 0 and 1 | 000000h–01FFFFh | 128 kB | Lower 1/64 |
| 0 | 1 | 0 | 1 | 0 | 0 thru 3 | 000000h–03FFFFh | 256 kB | Lower 1/32 |
| 0 | 1 | 0 | 1 | 1 | 0 thru 7 | 000000h–07FFFFh | 512 kB | Lower 1/16 |
| 0 | 1 | 1 | 0 | 0 | 0 thru 15 | 000000h–0FFFFFh | 1 MB | Lower 1/8 |
| 0 | 1 | 1 | 0 | 1 | 0 thru 31 | 000000h–1FFFFFh | 2 MB | Lower 1/4 |
| 0 | 1 | 1 | 1 | 0 | 0 thru 63 | 000000h–3FFFFFh | 4 MB | Lower 1/2 |
| X | X | 1 | 1 | 1 | 0 thru 127 | 000000h–7FFFFFh | 8 MB | All |
| 1 | 0 | 0 | 0 | 1 | 127 | 7FF000h–7FFFFFh | 4 kB | Upper 1/2048 |
| 1 | 0 | 0 | 1 | 0 | 127 | 7FE000h–7FFFFFh | 8 kB | Upper 1/1024 |
| 1 | 0 | 0 | 1 | 1 | 127 | 7FC000h–7FFFFFh | 16 kB | Upper 1/512 |
| 1 | 0 | 1 | 0 | X | 127 | 7F8000h–7FFFFFh | 32 kB | Upper 1/256 |
| 1 | 1 | 0 | 0 | 1 | 0 | 000000h–000FFFh | 4 kB | Lower 1/2048 |
| 1 | 1 | 0 | 1 | 0 | 0 | 000000h–001FFFh | 8 kB | Lower 1/1024 |
| 1 | 1 | 0 | 1 | 1 | 0 | 000000h–003FFFh | 16 kB | Lower 1/512 |
| 1 | 1 | 1 | 0 | X | 0 | 000000h–007FFFh | 32 kB | Lower 1/256 |

**Note**
22.X = don’t care.

**Table 29 S25FL064L legacy complement block protection (CMP = 1)**

| Status Register | | | | | 64 Mb legacy block protection (CMP = 1) | | | |
|---|---|---|---|---|---|---|---|---|
| SEC | TBPORT | BP2 | BP1 | BP0 | Protected block(s) | Protected addresses | Protected density | Protected portion |
| X | X | 0 | 0 | 0 | 0 thru 127 | 000000h–7FFFFFh | 8 MB | ALL |
| 0 | 0 | 0 | 0 | 1 | 0 thru 125 | 000000h–7DFFFFh | 8,064 kB | Lower 63/64 |
| 0 | 0 | 0 | 1 | 0 | 0 thru 123 | 000000h–7BFFFFh | 7,936 kB | Lower 31/32 |
| 0 | 0 | 0 | 1 | 1 | 0 thru 119 | 000000h–77FFFFh | 7,680 kB | Lower 15/16 |
| 0 | 0 | 1 | 0 | 0 | 0 thru 111 | 000000h–6FFFFFh | 7 MB | Lower 7/8 |
| 0 | 0 | 1 | 0 | 1 | 0 thru 95 | 000000h–5FFFFFh | 5 MB | Lower 3/4 |
| 0 | 0 | 1 | 1 | 0 | 0 thru 63 | 000000h–3FFFFFh | 4 MB | Lower 1/2 |
| 0 | 1 | 0 | 0 | 1 | 2 thru 127 | 020000h–7FFFFFh | 8,064 kB | Upper 63/64 |
| 0 | 1 | 0 | 1 | 0 | 4 thru 127 | 040000h–7FFFFFh | 7,936 kB | Upper 31/32 |
| 0 | 1 | 0 | 1 | 1 | 8 thru 127 | 080000h–7FFFFFh | 7,680 kB | Upper 15/16 |
| 0 | 1 | 1 | 0 | 0 | 16 thru 127 | 100000h–7FFFFFh | 7 MB | Upper 7/8 |
| 0 | 1 | 1 | 0 | 1 | 32 thru 127 | 200000h–7FFFFFh | 5 MB | Upper 3/4 |
| 0 | 1 | 1 | 1 | 0 | 64 thru 127 | 400000h–7FFFFFh | 4 MB | Upper 1/2 |
| X | X | 1 | 1 | 1 | None | None | None | None |
| 1 | 0 | 0 | 0 | 1 | 0 thru 127 | 000000h–7FEFFFh | 8,188 kB | Lower 2047/2048 |
| 1 | 0 | 0 | 1 | 0 | 0 thru 127 | 000000h–7FDFFFh | 8,184 kB | Lower 1023/1024 |
| 1 | 0 | 0 | 1 | 1 | 0 thru 127 | 000000h–7FBFFFh | 8,176 kB | Lower 511/512 |
| 1 | 0 | 1 | 0 | X | 0 thru 127 | 000000h–7F7FFFh | 8,160 kB | Lower 255/256 |
| 1 | 1 | 0 | 0 | 1 | 0 thru 127 | 001000h–7FFFFFh | 8,188 kB | Upper 2047/2048 |
| 1 | 1 | 0 | 1 | 0 | 0 thru 127 | 002000h–7FFFFFh | 8,184 kB | Upper 1023/1024 |
| 1 | 1 | 0 | 1 | 1 | 0 thru 127 | 004000h–7FFFFFh | 8,176 kB | Upper 511/512 |
| 1 | 1 | 1 | 0 | X | 0 thru 127 | 008000h–7FFFFFh | 8,160 kB | Upper 255/256 |

**Note**
23.X = don’t care.

### 7.6.2 Individual block lock (IBL) protection

Individual block lock bits (IBL) are volatile, with one bit for each sector / block, and each bit can be individually modified. By issuing the IBL or GBL commands, a IBL bit is set to ’0’ protecting each related sector / block. By issuing the IBUL or GUL commands, a IBL bit is cleared to ’1’ unprotecting each related sector or block. By issuing the IBLRD command the state of each IBL bit can be read. This feature allows software to easily protect individual sectors / blocks against inadvertent changes, yet does not prevent the easy removal of protection when changes are needed. The IBL’s can be set or cleared as often as needed as they are volatile bits.

Every main 64 KB block and the 4 KB sectors in bottom and top blocks has a volatile individual block lock bit (IBL) associated with it. When a sector / block IBL bit is ‘0’, the related sector/block is protected from program and erase operations.

If the pointer region protection is enabled this protected region is logically ORed with the IBL bits.

Following power-up, hardware reset, or software reset the default state [IBLLBB = 1] (see **Table 20**) all individual IBL bits are set to ‘0’ in the protected state. In order to program or erase the array the global IBL unlock or the Sector / Block IBL Unlock command must be given before the Program or Erase commands. When [IBLLBB = 0], all the individual IBL bits are set to ‘1’ in the un-protected state following power-up, hardware reset, or software reset.

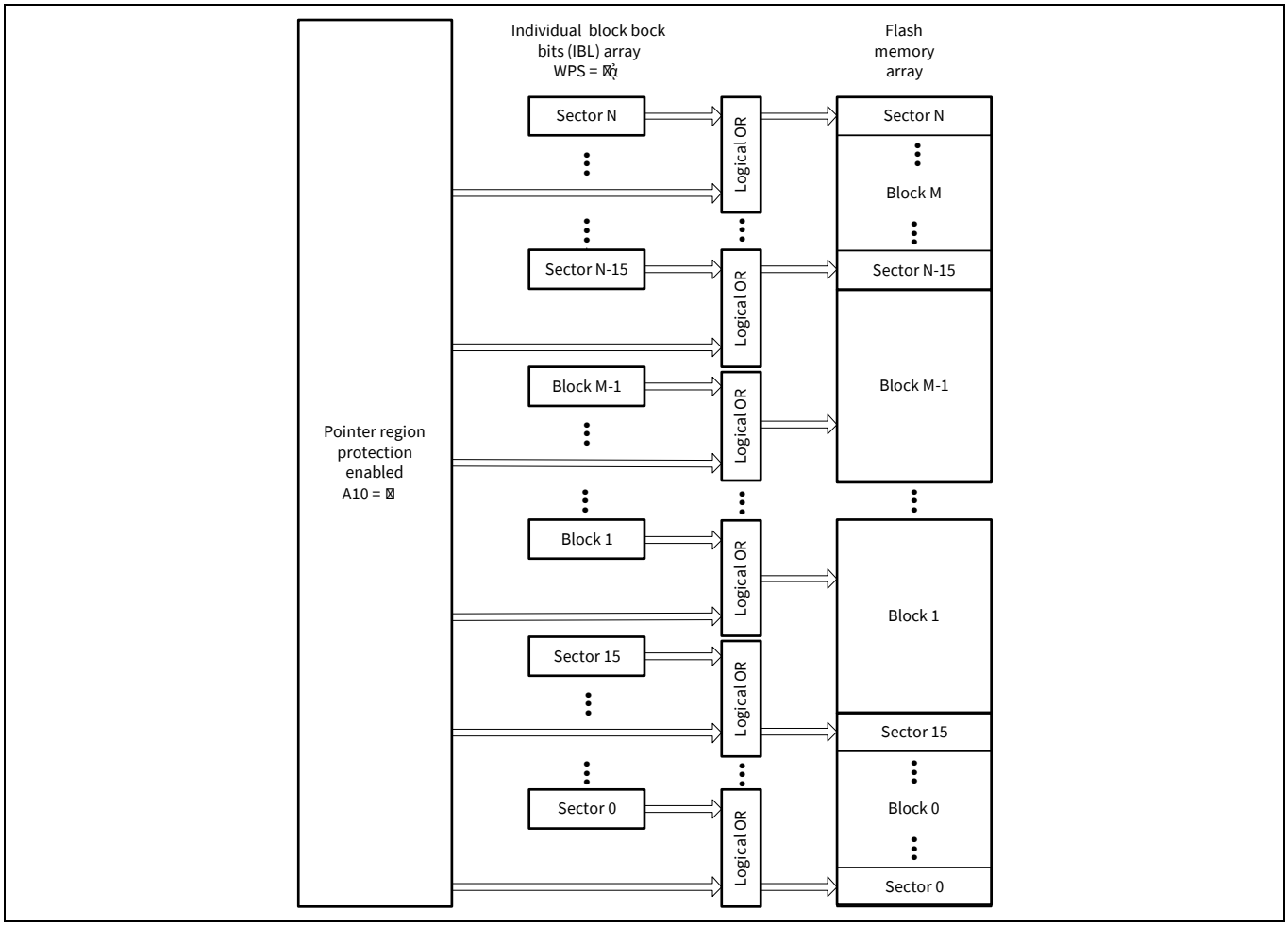



**Figure 25 Individual block lock / pointer region protection control**

**Notes**
24.The ‘M’ is the top 64 KB block.
25.The ‘N’ is the top 4 KB sector.

### 7.6.3 Pointer region protection (PRP)

The pointer region protection is defined by a non-volatile address pointer that selects any 4 KB sector as the boundary between protected and unprotected regions in the memory. This provides a protection scheme with individual sector granularity that remains in effect across power cycles and reset operations. PRP settings can also be protected from modification until the next power cycle, until a password is supplied, or can be permanently locked. PRP can be used in combination with either the legacy block protection or individual block lock protection methods. When enabled, PRP protection is logically ORed with the protection method selected by the WPS bit (CR2V[2])

The set pointer region protection (SPRP FBh or 4SPRP E3h) command (see **"Pointer Region command"** on page 117) or Write Any Register (WRAR 71h) command to write the PRPR register (see **"Write Any Register (WRAR 71h)"** on page 86) is used to enable or disable PRP, and set the pointer value.

After the set block/pointer protection command is given or Write Any Register (WRAR 71h) command to write the PRPR register, the value of A10 enables or disables the pointer protection mechanism. If A10 = 1, then the pointer protection region is disabled. This is the default state, and the rest of pointer values are don't care. If A10 = 0, then the pointer protection region is enabled. The value of A10 is written in the Non-volatile Pointer bit in the PRPR. The pointer address values for RFU bits are don't care but these bit locations will read back as ones. See **"Pointer Region Protection Register (PRPR)"** on page 47 for additional information on the PRPR.

If the pointer protection mechanism is enabled, the pointer value determines the block boundary between the protected and the unprotected regions in the memory. The pointer boundary is set by the three (A23–A12) or four (A31–A12) address bytes written to the non-volatile pointer value in the PRPR. The area that is unprotected will be inclusive of the 4KB sector selected by the pointer value.

The value of A9 is used to determine whether the region that is unprotected will start from the top (highest address) or bottom (lowest address) of the memory array to the location of the pointer. If A9 = 0 when the SPRP or 4SPRP command is issued followed by a the address, then the 4 kB sector which includes that address and all the sectors from the bottom up (zero to higher address) will be unprotected. If A9 = 1 when the SPRP or 4SPRP-command is issued followed by address then the 4 kB sector which includes that address and all the sectors from the top down (max to lower address) will be unprotected. The value of A9 is in the non-volatile pointer value in the PRPR.

The A11 bit can be used to protect all sectors. If A11 = 1, then all sectors are protected. If A11 = 0, then the unprotected range will be determined by Amax–A12. The value of A11 is in the non-volatile pointer value in the PRPR.

The SPRP or 4SPRP command is ignored during a suspend operation because the pointer value cannot be erased and re-programmed during a suspend.

The SPRP or 4SPRP command is ignored if NVLOCK PR[0] = 0.

The Read Any Register 65h command (see **"Read Any Register (RDAR 65h)"** on page 83) reads the contents of PRP access register. This allows the contents of the pointer to be read out for test and verification.

**Table 30 PRP table**

| A11 | A10 | A9 | Protect address range | Unprotect address range | Comment |
|---|---|---|---|---|---|
| x | 1 | x | None | All | A10 = 1 is PRP disabled (this is the default state and the rest of pointer value is don't care). |
| 0 | 0 | 0 | 1FFFFFF to (A[31:12]+1) | A[31:12] to 0000000 | The 4 kB sector which includes that address and all the sectors from the bottom up (zero to higher address) will be unprotected. |
| 0 | 0 | 1 | (A[31;12]-1) to 0000000 | 1FFFFFF to A[31:12] | The 4kB sector which includes that address and all the sectors from the Top down (max to lower address) will be unprotected. |
| 1 | 0 | x | 1FFFFFF to 000000 | Not Applicable | A10 = 0 and A11 = 1 means protect all sectors and Amax–A12 are don't care. |

If the pointer protect scheme is active (A10 = 0), and the pointer protects any portion of the address space to which an Erase command is applied, the Erase command fails. For example, if the pointer protection is protecting 4 KB of the array that would be affected by a Block Erase command, that erase command fails. Chip Erase (CEh) command is ignored if PRP is enabled (A10 = 0) and this will set the E_ERR status bit.

If the pointer region protection is enabled this protection is logically ORed with either the legacy block protection region if WPS CR2V[2] = 0 or individual block lock protection if WPS CR2V[2] = 1 (See **Figure 24**).

## 7.7 Individual and region protection

Individual and region protection (IRP) is the name used for a set of independent hardware and software methods used to disable or enable programming or erase operations on Security Regions 2 and 3 and the Pointer Region Protection Register.

Each method manages the state of the NVLOCK bit (PR[0]). When NVLOCK = 1, the Security Regions 2 and 3 and the Pointer Region Protection Register (PRPR) may be programmed and erased. When NVLOCK = 0, the Security Regions 2 and 3 and PRPR can not be programmed or erased. Note, the Security Regions 2 and 3 are also protected respectively by LB2 or LB3 = 1 (CR1NV[4:5]).

Power supply lock-down protection is the default method. This method sets the NVLOCK bit to ‘1’ during POR or hardware reset so that the NVLOCK related areas and registers are unprotected by a device reset. The PRL (A6h) command clears the NVLOCK bit to ‘0’ to protect the NVLOCK related areas and registers. There is no command in the power supply lock-down method to set the NVLOCK bit to ‘1’, therefore the NVLOCK bit will remain at ‘0’ until the next power-off or hardware reset. The power supply lock-down method allows boot code the option of changing Security Regions 2 and 3 or the value in PRPR, by programming or erasing these non-volatile areas, then protecting these non-volatile areas from further change for the remainder of normal system operation by clearing the NVLOCK bit to ‘0’. This is sometimes called boot-code controlled protection.

The password method clears the Protection Register NVLOCK bit to 0 and sets the SECRRP bit = IRP[6] during POR or hardware reset to protect the NVLOCK related areas and registers. The SECRRP bit determines whether Security Region 3 is readable. A 64-bit password may be permanently programmed and hidden for the password method. The PASSU (EAh) command can be used to provide a password for comparison with the hidden password. If the password matches, the NVLOCK bit is set to ‘1’ to unprotect the NVLOCK related areas and registers. The PRL (A6h) command can be used to clear the NVLOCK bit to ‘0’ to turn on protection again.

The permanent method permanently sets the SECRRP bit = 1 and clears NVLOCK to 0. This permanently protects the Security Regions 2 and 3 and the PRPR.The selection of the NVLOCK bit management method is made by Programming OTP bits in the IRP Register (IRP[2 or 1 or 0] so as to permanently select the method used. An overview of all methods is shown in **Figure 26**.

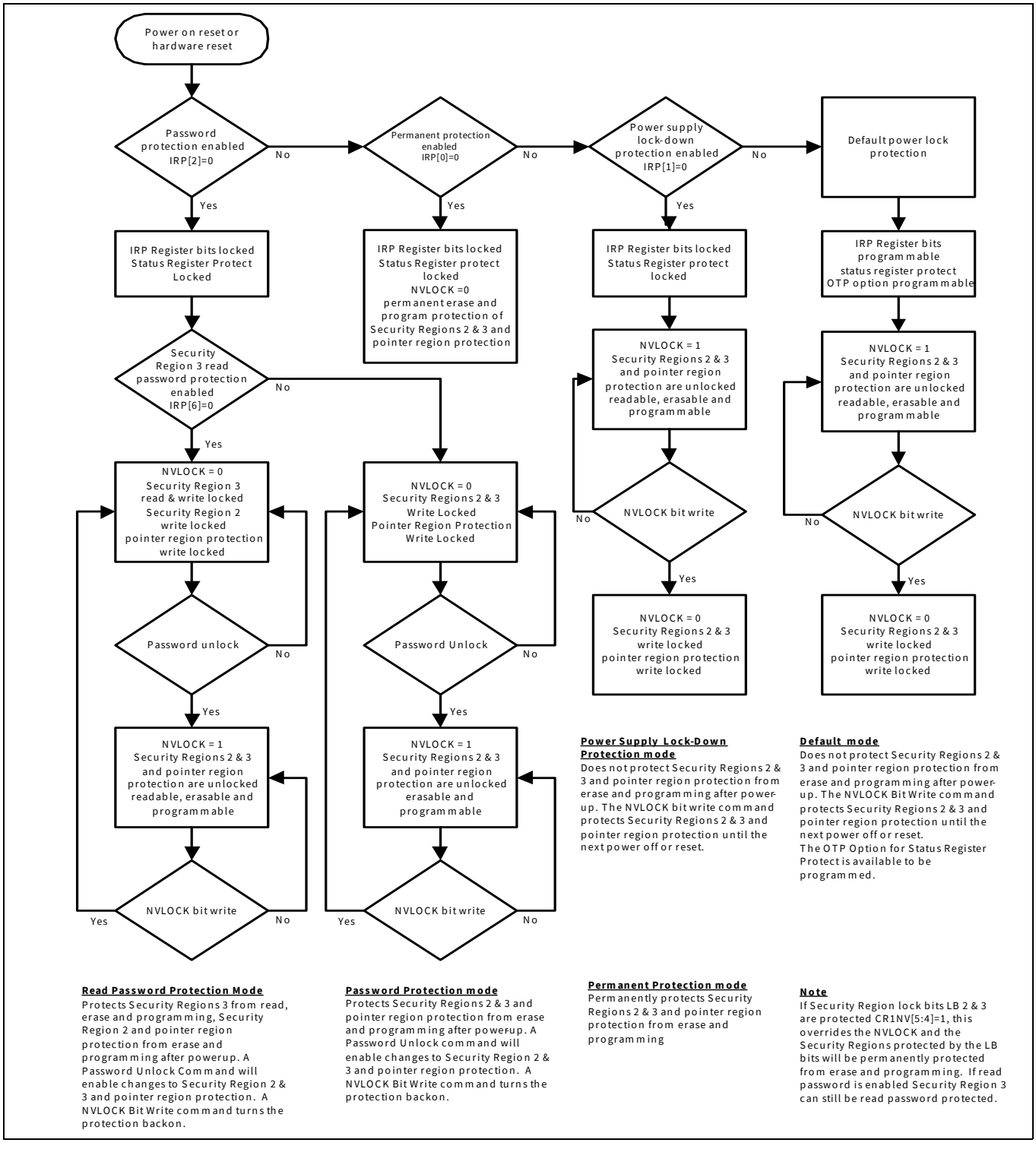



**Figure 26 Permanent, password and power supply lock-down protection overview**

### 7.7.1 IRP Register

The IRP register is used to permanently configure the behavior of individual and region protection (IRP) features, see **Table 20**.

As shipped from the factory, all devices default to the Power Supply Lock-Down Protection mode, with all regions unprotected.

The device programmer or host system must then choose which protection method to use by programming one of the One-time Programmable bits, permanent, Power Supply Lock-Down or Password Protection mode. Programming one of these bits locks the part permanently in the selected mode:

Factory defaults IRP Register

- IRP[6] = '1' = Read Password Protection mode not enabled.
- IRP[4] = '1' = IBL bits power-up in protected state.
- IRP[2] = '1' = Password Protection mode not enabled.
- IRP[1] = '1' = Power Supply Lock-Down Protection mode not enabled but is the default mode.
- IRP[0] = '1' = Permanent Protection mode not enabled.

IRP register programming rules:

- If the Read Password mode is chosen, the SECRRP bit must be programmed prior or at the same time as setting the Password Protection mode lock bits IRP[2].
- If the IBL bits power-up in Unprotected mode is chosen, the IBLLBB bit must be programmed prior or at the same time as setting one of the Protection Mode Lock bits IRP[2:0].
- If the Password mode is chosen, the password must be programmed prior to setting the Password Protection Mode Lock bits IRP[2].
- The Protection modes are mutually exclusive, only one may be selected. Once one of the Protection modes is selected IPRP[2:0], the IRP Register bits are permanently protected from programming and no further changes to the OTP Register bits is allowed. If an attempt to change any of the register bits above, after the Protection mode is selected, the operation will fail and P_ERR (SR2V[5]) will be set to 1.

The programming time of the IRP Register is the same as the typical page programming time. The system can determine the status of the IRP register programming operation by reading the WIP bit in the Status Register. See **"Status Register 1"** on page 30 for information on WIP. See **"Password Protection mode"** on page 60.

#### 7.7.1.1 IBL Lock Boot bit

The default IBL Lock bit IRP[4] = 1, all the IBL bits on power-up or reset (after a hardware reset or software reset) to the "protected state". If the IBL Lock bit IRP[4] = 0 (programmed), the IBL power-up or reset to the "unprotected state".

### 7.7.2 Protection Register (PR)

#### 7.7.2.1 NVLOCK bit (PR[0])

The NVLOCK bit is a volatile bit for protecting:

- Pointer Region Protection Register
- Security Regions 2 and 3

When cleared to '0', NVLOCK locks the related regions. When set to '1', it allows the related regions to be changed. See **"Protection Register (PR)"** on page 46 for more information.

The PRL command is used to clear the NVLOCK bit to '0'. The NVLOCK bit should be cleared to '0' only after all the related regions are configured to the desired settings.

In Power Supply Lock-Down Protection mode, the NVLOCK is set to '1' during POR or a hardware reset. A Software Reset command does not affect the NVLOCK bit. When cleared to '0', no Software command sequence can set the NVLOCK bit to '1', only another hardware reset or power-up can set the NVLOCK bit.

In the Password Protection mode, the NVLOCK bit is cleared to '0' during POR, or a hardware reset. The NVLOCK bit can only be set to '1' by the Password Unlock command.

The permanent method permanently clears NVLOCK to 0. This permanently protects the Security Regions 2 and 3 and the PRPR.

#### 7.7.2.2 Security Region Read Password Lock bit (SECRRP, PR[6])

The SECRRP bit is a volatile bit for read protecting Security Region 3. When SECRRP[6] = 0, the Security Region 3 can not be read, See **"Protection Register (PR)"** on page 46 for more information.

In the Password Protection mode, the SECRRP bit is set equal to IRP[6] during POR or software or hardware reset. The NVLOCK bit can only be set to '1' by the Password Unlock command. A software reset does not affect the NVLOCK bit.

The permanent method permanently sets the SECRRP bit = 1. This permanently leaves Security Region 3 readable.

### 7.7.3 Password Protection mode

Password Protection mode allows an even higher level of security than the Power Supply Lock-Down Protection mode, by requiring a 64-bit password for unlocking the NVLOCK bit. In addition to this password requirement, after power up, hardware reset, the NVLOCK bit is cleared to '0' to ensure protection after power-up or reset. Successful execution of the password unlock command by entering the entire password sets the NVLOCK bit to 1, allowing for sector NVLOCK related areas and registers modifications.

Password protection notes:

- Once the password is programmed and verified, the Password mode (IRP[2] = 0) must be set in order to prevent reading the password.
- The Password Program command is only capable of programming '0's. Programming a '1' after a cell is programmed as a '0' results in the cell left as a '0' with no programming error set.
- The password is all '1's when shipped from Infineon. It is located in its own memory space and is accessible through the use of the Password Program, Password Read, RDAR, and WRAR commands.
- All 64-bit password combinations are valid as a password.
- The Password mode, once programmed, prevents reading the 64-bit password and further password programming. All further Program and Read commands to the password region are disabled and these commands are ignored or return undefined data. There is no means to verify what the password is after the Password mode lock bit is selected. Password verification is only allowed before selecting the password Protection mode.
- The Protection mode lock bits are not erasable.

- The exact password must be entered in order for the unlocking function to occur. If the Password Unlock command provided password does not match the hidden internal password, the unlock operation fails in the same manner as a programming operation on a protected sector. The P_ERR bit is set to one, the WIP bit remains set, and the NVLOCK bit remains cleared to 0.
- The Password Unlock command cannot be accepted any faster than once every 100 µs ± 20 µs. This makes it take an unreasonably long time (58 million years) for a hacker to run through all the 64-bit combinations in an attempt to correctly match a password. The Read Status Register 1 command may be used to read the WIP bit to determine when the device has completed the Password Unlock command or is ready to accept a New Password command. When a valid password is provided the Password Unlock command does not insert the 100 µs delay before returning the WIP bit to zero.
- If the password is lost after selecting the Password mode, there is no way to set the NVLOCK bit = 1.

### 7.7.4 Security Region read password protection

The Security Region read password protection enables protecting Security Region 3 from read, program and erase.

Security Region read password protection is an optional addition to the Password Protection mode (described above). The Security Regions read password protection is enabled when the user programs SECRRP bit IRP[6] = 0. The SECRRP bit IRP[6] must be programmed prior or at the same time as setting the Password Protection mode lock bits IRP[2].

The Security Regions read password protection is not active until the password is programmed, IRP[2] is programmed to 0.

When the SECRRP (PR[6]) bit is set to 0 the Security Region 3 is not readable. If these regions are read the resulting data is invalid and undefined.

### 7.7.5 Recommended IRP protection process

During system manufacture, the Flash device configuration should be defined by:

- Programming the Security Regions as desired.
- Set Pointer Region Protection Register as desired
- Program the Password register (PASS) if password protection will be used.
- Program the IRP Register as desired, including the selection of permanent, Power Supply Lock-Down or Password IRP Protection mode in IRP[2:0]. It is very important to explicitly select a Protection mode so that later accidental or malicious programming of the IRP register is prevented. This is to ensure that only the intended protection features are enabled. Before or while programming the IRP register:
  - The IBLLBB bit (IRP[4]) may be used to cause all the IBL bits to power up in the unprotected state.
  - The SECRRP bit (IRP[6]) may be programmed to select Security Regions read password protection to use the password to control read access to the Security Region 3.

During system power up and boot code execution: If the Power Supply Lock-Down Protection mode is in use, trusted boot code can determine whether there is any need to modify the NVLOCK related areas or registers. If no changes are needed the NVLOCK bit can be cleared to 0 via the PRL command to protect the NVLOCK related areas or registers from changes during the remainder of normal system operation while power remains on.

# 8 Commands

All communication between the host system and FL-L family memory devices is in the form of units called commands. See **"Command protocol"** on page 17 for details on command protocols.

Although host software in some cases is used to directly control the SPI interface signals, the hardware interfaces of the host system and the memory device generally handle the details of signal relationships and timing. For this reason, signal relationships and timing are not covered in detail within this software interface focused section of the document. Instead, the focus is on the logical sequence of bits transferred in each command rather than the signal timing and relationships. Following are some general signal relationship descriptions to keep in mind. For additional information on the bit level format and signal timing relationships of commands, see **"Command protocol"** on page 17.

- The host always controls the Chip Select (CS#), Serial Clock (SCK), and Serial Input (SI) - SI for single bit wide transfers. The memory drives Serial Output (SO) for single bit read transfers. The host and memory alternately drive the IO0-IO3 signals during dual and quad transfers.
- All commands begin with the host selecting the memory by driving CS# LOW before the first rising edge of SCK. CS# is kept low throughout a command and when CS# is returned HIGH the command ends. Generally, CS# remains LOW for eight bit transfer multiples to transfer byte granularity information. No commands will be accepted if CS# is returned HIGH not at an 8-bit boundary.

## 8.1 Command set summary

### 8.1.1 Extended addressing

- Instructions that always require a 4-byte address, used to access up to 32 Gb of memory:

**Table 31 Extended address 4-byte address commands**

| Command name | Function | Instruction (hex) |
|---|---|---|
| 4READ | Read | 13 |
| 4FAST_READ | Read fast | 0C |
| 4DOR | Dual output read | 3C |
| 4QOR | Quad output read | 6C |
| 4DIOR | Dual I/O read | BC |
| 4QIOR | Quad I/O read | EC |
| 4DDRQIOR | DDR quad I/O read | EE |
| 4PP | Page program | 12 |
| 4QPP | Quad page program | 34 |
| 4SE | Sector erase | 21 |
| 4HBE | Half block erase | 53 |
| 4BE | Block erase | DC |
| 4IBLRD | IBL read | E0 |
| 4IBL | IBL lock | E1 |
| 4IBUL | IBL unlock | E2 |
| 4SPRP | Set pointer region protection | E3 |

- A 4 -byte Address mode for backward compatibility to the 3-byte address instructions. The standard 3-byte instructions can be used in conjunction with a 4-byte Address mode controlled by the Address Length Configuration bit (CR2V[0]). The default value of CR2V[0] is loaded from CR2NV[1] (following power up, hardware reset, or software reset), to enable default 3-byte (24-bit) or 4-byte (32-bit) addressing. When the address length (CR2V[0]) set to 1, the legacy commands are changed to require 4-bytes (32 bits) for the address field. The following instructions can be used in conjunction with the 4-byte Address mode configuration to switch from 3-bytes to 4-bytes of address field.

**Table 32 Extended address 4-byte Address mode with 3-byte Address commands**

| Command name | Function | Instruction (hex) |
|---|---|---|
| RSFDP | Read SFDP | 5A |
| READ | Read | 03 |
| FAST_READ | Read Fast | 0B |
| DOR | Dual Output Read | 3B |
| QOR | Quad Output Read | 6B |
| DIOR | Dual I/O Read | BB |
| QIOR | Quad I/O Read | EB |
| DDRQIOR | DDR Quad I/O Read | ED |
| PP | Page Program | 02 |
| QPP | Quad Page Program | 32 |
| SE | Sector Erase | 20 |
| HBE | Half Block Erase | 52 |
| BE | Block Erase | D8 |
| RDAR | Read Any Register | 65 |
| WRAR | Write Any Register | 71 |
| SECRE | Security Region Erase | 44 |
| SECRP | Security Region Program | 42 |
| SECRR | Security Region Read | 48 |
| IBLRD | IBL Read | 3D |
| IBL | IBL Lock | 36 |
| IBUL | IBL Unlock | 39 |
| SPRP | Set Pointer Region Protection | FB |

### 8.1.2 Command summary by function

**Table 33 FL-L family command set (sorted by function)**

| Function | Command name | Command description | Instruction value (hex) | Maximum frequency (MHz) | Address length (bytes) | QPI |
|---|---|---|---|---|---|---|
| Read device ID | RDID | Read ID (JEDEC Manufacturer ID) | 9F | 108 | 0 | Yes |
| | RSFDP | Read JEDEC serial flash discoverable parameters | 5A | 108 | 3 or 4 | Yes |
| | RDQID | Read Quad ID | AF | 108 | 0 | Yes |
| | RUID | Read Unique ID | 4B | 108 | 0 | Yes |
| Register access | RDSR1 | Read Status Register 1 | 05 | 108 | 0 | Yes |
| | RDSR2 | Read Status Register 2 | 07 | 108 | 0 | No |
| | RDCR1 | Read Configuration Register 1 | 35 | 108 | 0 | No |
| | RDCR2 | Read Configuration Register 2 | 15 | 108 | 0 | No |
| | RDCR3 | Read Configuration Register 3 | 33 | 108 | 0 | No |
| | RDAR | Read Any Register | 65 | 108 | 3 or 4 | Yes |
| | WRR | Write Register (Status-1 and Configuration-1,2,3) | 01 | 108 | 0 | Yes |
| Register access | WREN | Write Enable for Non-volatile Data Change | 06 | 108 | 0 | Yes |
| | WRENV | Write Enable for Volatile Status and Configuration Registers | 50 | 108 | 0 | Yes |
| | WRAR | Write Any Register | 71 | 108 | 3 or 4 | Yes |
| | CLSR | Clear Status Register | 30 | 108 | 0 | Yes |
| | 4BEN | Enter 4-byte Address mode | B7 | 108 | 0 | Yes |
| | 4BEX | Exit 4-byte Address mode | E9 | 108 | 0 | Yes |
| | SBL | Set Burst Length | 77 | 108 | 0 | Yes |
| | QPIEN | Enter QPI | 38 | 108 | 0 | No |
| | QPIEX | Exit QPI | F5 | 108 | 0 | Yes |
| | DLPRD | Data Learning Pattern Read | 41 | 108 | 0 | Yes |
| | PDLRNV | Program NV Data Learning Register | 43 | 108 | 0 | Yes |
| | WDLRV | Write Volatile Data Learning Register | 4A | 108 | 0 | Yes |

**Note**
26.Commands not supported in QPI mode have undefined behavior if sent when the device is in QPI mode.

**Table 33 FL-L family command set (sorted by function)** *(continued)*

| Function | Command name | Command description | Instruction value (hex) | Maximum frequency (MHz) | Address length (bytes) | QPI |
|---|---|---|---|---|---|---|
| Read flash array | READ | Read | 03 | 50 | 3 or 4 | No |
| | 4READ | Read | 13 | 50 | 4 | No |
| | FAST_READ | Fast Read | 0B | 108 | 3 or 4 | No |
| | 4FAST_READ | Fast Read | 0C | 108 | 4 | No |
| | DOR | Dual Output Read | 3B | 108 | 3 or 4 | No |
| | 4DOR | Dual Output Read | 3C | 108 | 4 | No |
| | QOR | Quad Output Read | 6B | 108 | 3 or 4 | No |
| | 4QOR | Quad Output Read | 6C | 108 | 4 | No |
| | DIOR | Dual I/O Read | BB | 108 | 3 or 4 | No |
| | 4DIOR | Dual I/O Read | BC | 108 | 4 | No |
| | QIOR | Quad I/O Read (CR1V[1] = 1) or CR2V[3] = 1 | EB | 108 | 3 or 4 | Yes |
| | 4QIOR | Quad I/O Read (CR1V[1] = 1) or CR2V[3] = 1 | EC | 108 | 4 | Yes |
| | DDRQIOR | DDR Quad I/O Read (CR1V[1] = 1 or CR2V[3] = 1) | ED | 54 | 3 or 4 | Yes |
| | 4DDRQIOR | DDR Quad I/O Read (CR1V[1] = 1 or CR2V[3] = 1) | EE | 54 | 4 | Yes |
| Program flash array | PP | Page Program | 02 | 108 | 3 or 4 | Yes |
| | 4PP | Page Program | 12 | 108 | 4 | Yes |
| | QPP | Quad Page Program | 32 | 108 | 3 or 4 | No |
| | 4QPP | Quad Page Program | 34 | 108 | 4 | No |
| Erase flash array | SE | Sector Erase | 20 | 108 | 3 or 4 | Yes |
| | 4SE | Sector Erase | 21 | 108 | 4 | Yes |
| | HBE | Half Block Erase | 52 | 108 | 3 or 4 | Yes |
| | 4HBE | Half Block Erase | 53 | 108 | 4 | Yes |
| | BE | Block Erase | D8 | 108 | 3 or 4 | Yes |
| | 4BE | Block Erase | DC | 108 | 4 | Yes |
| | CE | Chip Erase | 60 | 108 | 0 | Yes |
| | CE | Chip Erase (alternate instruction) | C7 | 108 | 0 | Yes |
| Erase /program suspend /resume | EPS | Erase / Program Suspend | 75 | 108 | 0 | Yes |
| | EPR | Erase / Program Resume | 7A | 108 | 0 | Yes |
| Security Region array | SECRE | Security Region Erase | 44 | 108 | 3 or 4 | Yes |
| | SECRP | Security Region Program | 42 | 108 | 3 or 4 | Yes |
| | SECRR | Security Region Read | 48 | 108 | 3 or 4 | Yes |

**Note**
26.Commands not supported in QPI mode have undefined behavior if sent when the device is in QPI mode.

**Table 33 FL-L family command set (sorted by function)** *(continued)*

| Function | Command name | Command description | Instruction value (hex) | Maximum frequency (MHz) | Address length (bytes) | QPI |
|---|---|---|---|---|---|---|
| Array protection | IBLRD | IBL Read | 3D | 108 | 3 or 4 | Yes |
| | 4IBLRD | IBL Read | E0 | 108 | 4 | Yes |
| | IBL | IBL Lock | 36 | 108 | 3 or 4 | Yes |
| | 4IBL | IBL Lock | E1 | 108 | 4 | Yes |
| | IBUL | IBL Unlock | 39 | 108 | 3 or 4 | Yes |
| | 4IBUL | IBL Unlock | E2 | 108 | 4 | Yes |
| | GBL | Global IBL Lock | 7E | 108 | 0 | Yes |
| | GBUL | Global IBL Unlock | 98 | 108 | 0 | Yes |
| | SPRP | Set Pointer Region Protection | FB | 108 | 3 or 4 | Yes |
| | 4SPRP | Set Pointer Region Protection | E3 | 108 | 4 | Yes |
| Individual and region protection | IRPRD | IRP Register Read | 2B | 108 | 0 | Yes |
| | IRPP | IRP Register Program | 2F | 108 | 0 | Yes |
| | PRRD | Protection Register Read | A7 | 108 | 0 | Yes |
| | PRL | Protection Register Lock (NVLOCK bit Write) | A6 | 108 | 0 | Yes |
| | PASSRD | Password Read | E7 | 108 | 0 | Yes |
| | PASSP | Password Program | E8 | 108 | 0 | Yes |
| | PASSU | Password Unlock | EA | 108 | 0 | Yes |
| Reset | RSTEN | Software Reset Enable | 66 | 108 | 0 | Yes |
| | RST | Software Reset | 99 | 108 | 0 | Yes |
| | MBR | Mode Bit Reset | FF | 108 | 0 | Yes |
| Deep power down | DPD | Deep Power-down | B9 | 108 | 0 | Yes |
| | RES | Release from Deep Power down / Device Id | AB | 108 | 0 | Yes |

**Note**
26.Commands not supported in QPI mode have undefined behavior if sent when the device is in QPI mode.

### 8.1.3 Read Device identification

There are multiple commands to read information about the device manufacturer, device type, and device features. SPI memories from different vendors have used different commands and formats for reading information about the memories. The FL-L family supports the three device information commands.

### 8.1.4 Register read or write

There are multiple registers for reporting embedded operation status or controlling device configuration options. There are commands for reading or writing these registers. Registers contain both volatile and non-volatile bits. non-volatile bits in registers are automatically erased and programmed as a single (write) operation.

#### 8.1.4.1 Monitoring operation status

The host system can determine when a write, program, erase, suspend or other embedded operation is complete by monitoring the Write-In Progress (WIP) bit in the Status Register. The Read from Status Register 1 command or Read Any Register command provides the state of the WIP bit. The Read from Status Register 1 or Read Any Register command provides the state of the program error (P_ERR) and erase error (E_ERR) bits in the status register indicate whether the most recent program or erase command has not completed successfully. When P_ERR or E_ERR bits are set to one, the WIP bit will remain set to one indicating the device remains busy and unable to receive most new operation commands. Only status reads (RDSR1 05h, RDSR2 07h), Read Any Register (RDAR 65h), Read Configuration RDCR1, RDCR2 and RDCR3, Status Clear (CLSR 30h), and Software Reset (RSTEN 66h followed by RST 99h) are valid commands when P_ERR or E_ERR is set to 1. A Clear Status Register (CLSR) command must be sent to return the device to STANDBY state. Alternatively, Hardware Reset, or Software Reset (RSTEN 66h followed by RST 99h) may be used to return the device to STANDBY state.

#### 8.1.4.2 Configuration

There are commands to read, write, and protect registers that control interface path width, interface timing, interface address length, and some aspects of data protection.

### 8.1.5 Read flash array

Data may be read from the memory starting at any byte boundary. Data bytes are sequentially read from incrementally higher byte addresses until the host ends the data transfer by driving CS# input High. If the byte address reaches the maximum address of the memory array, the read will continue at address zero of the array.

Burst wrap read can be enabled by the Set burst length (SBL 77h) command with the requested wrapped read length and alignment, see **“Set Burst Length (SBL 77h)”** on page 87. Burst Wrap read is only for Quad I/O and QPI modes.

There are several different read commands to specify different access latency and data path widths. Double data rate (DDR) commands also define the address and Data bit relationship to both SCK edges:

- The Read command provides a single address bit per SCK rising edge on the SI/IO0signal with read data returning a single bit per SCK falling edge on the SO/IO1signal. This command has zero latency between the address and the returning data but is limited to a maximum SCK rate of 50 MHz.
- Other Read commands have a latency period between the address and returning data but can operate at higher SCK frequencies. The latency depends on a configuration register read latency value.
- The Fast Read command provides a single address bit per SCK rising edge on the SI/IO0 signal with read data returning a single bit per SCK falling edge on the SO/IO1 signal.
- Dual or Quad Output Read commands provide address on SI/IO0 pin on the SCK rising edge with read data returning two bits, or four bits of data per SCK falling edge on the IO0–IO3 signals.
- Dual or Quad I/O Read commands provide address two bits or four bits per SCK rising edge with read data returning two bits, or four bits of data per SCK falling edge on the IO0–IO3 signals. Continuous read feature is enabled if the mode bits value is Axh.
- Quad Double Data Rate Read commands provide address four bits per every SCK edge with read data returning four bits of data per every SCK edge on the IO0–IO3 signals. Continuous read feature is enabled if the Mode bits value is Axh.

### 8.1.6 Program flash array

Programming data requires two commands: Write enable (WREN), and Page Program (PP, 4PP, QPP, 4QPP). The Page Program command accepts from 1-byte up to 256 consecutive bytes of data (page) to be programmed in one operation. Programming means that bits can either be left at 1, or programmed from 1 to 0. Changing bits from 0 to 1 requires an erase operation.

### 8.1.7 Erase flash array

The Sector Erase, Half Block Erase, Block Erase, or Chip Erase commands set all the bits in a sector or the entire memory array to 1. A bit needs to be first erased to 1 before programming can change it to a 0. While bits can be individually programmed from a 1 to 0, erasing bits from 0 to 1 must be done on a sector-wide, half block-wide, block-wide or array-wide (chip) level. The Write Enable (WREN) command must precede an erase command.

### 8.1.8 Security Regions, legacy block protection, and individual and region protection

There are commands to read and program a separate one time protection (OTP) array for permanently protected data such as a serial number. There are commands to control a contiguous group (block) of flash memory array sectors that are protected from program and erase operations.There are commands to control which individual flash memory array sectors are protected from program and erase operations. There is a mode to limit read access of Security Region 3 until a password is supplied.

### 8.1.9 Reset

There are commands to reset to the default conditions present after power on to the device. However, the Software Reset commands do not affect the current state of the SRP1 or NVLOCK bits. In all other respects a Software Reset is the same as a Hardware Reset.

There is a command to reset (exit from) the Continuous Read mode.

### 8.1.10 Reserved

Some instructions are reserved for future use. In this generation of the FL-L family some of these command instructions may be unused and not affect device operation, some may have undefined results.

Some commands are reserved to ensure that a legacy or alternate source device command is allowed without effect. This allows legacy software to issue some commands that are not relevant for the current generation FL-L family with the assurance these commands do not cause some unexpected action.

Some commands are reserved for use in special versions of the FL-L not addressed by this document or for a future generation. This allows new host memory controller designs to plan the flexibility to issue these command instructions. The command format is defined if known at the time this document revision is published.

## 8.2 Identification commands

### 8.2.1 Read identification (RDID 9Fh)

The Read Identification (RDID) command provides read access to manufacturer identification, device identification. The manufacturer identification is assigned by JEDEC. The device identification values are assigned by Infineon.

Any RDID command issued while a program, erase, or write cycle is in progress is ignored and has no effect on execution of the program, erase, or write cycle that is in progress.

The RDID instruction is shifted on SI / IO0. After the last bit of the RDID instruction is shifted into the device, a byte of manufacturer identification, two bytes of device identification, will be shifted sequentially out on SO / IO1, As a whole this information is referred to as ID. See **"Device ID address map"** on page 141 for the detail description of the ID contents.

Continued shifting of output beyond the end of the defined ID address space will provide undefined data. The RDID command sequence is terminated by driving CS# to the logic HIGH state anytime during data output. The RDID command is supported up to 108 MHz.

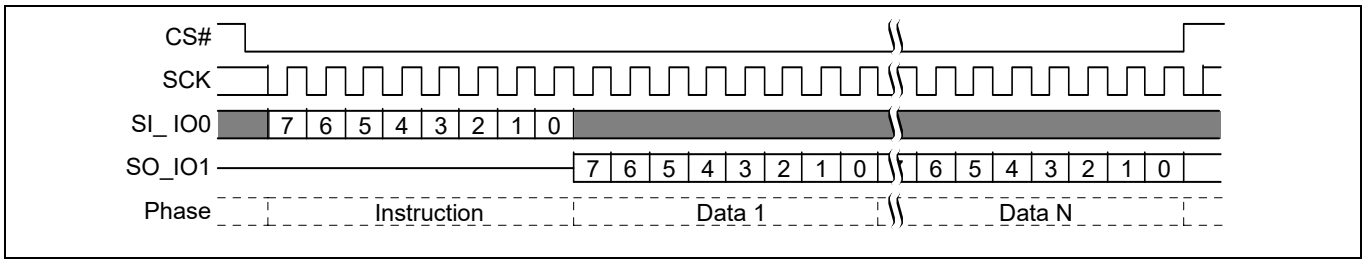



**Figure 27 Read Identification (RDID) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

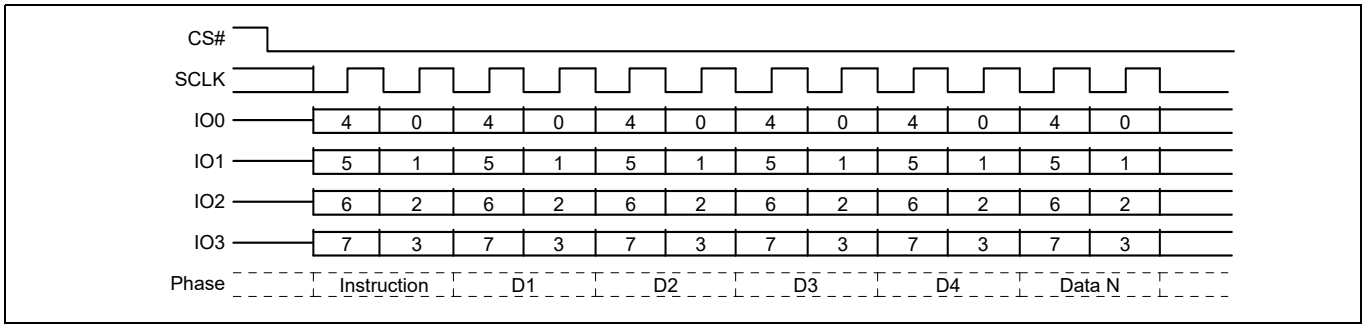



**Figure 28 Read Identification (RDID) QPI mode command**

## 8.2.2 Read Quad Identification (RDQID AFh)

The Read Quad Identification (RDQID) command provides read access to manufacturer identification, device identification. This command is an alternate way of reading the same information provided by the RDID command while in QPI mode. In all other respects the command behaves the same as the RDID command.

The command is recognized only when the device is in QPI mode (CR2V[3] = 1) or Quad mode (CR1V[1] = 1). The instruction is shifted in on IO0–IO3 for QPI mode and IO0 for Quad mode. After the last bit of the instruction is shifted into the device, a byte of manufacturer identification, two bytes of device identification will be shifted sequentially out on IO0–IO3. As a whole this information is referred to as ID. See **"Device ID address map"** on page 141 for the detail description of the ID contents.

Continued shifting of output beyond the end of the defined ID address space will provide undefined data. The command sequence is terminated by driving CS# to the logic HIGH state anytime during data output.

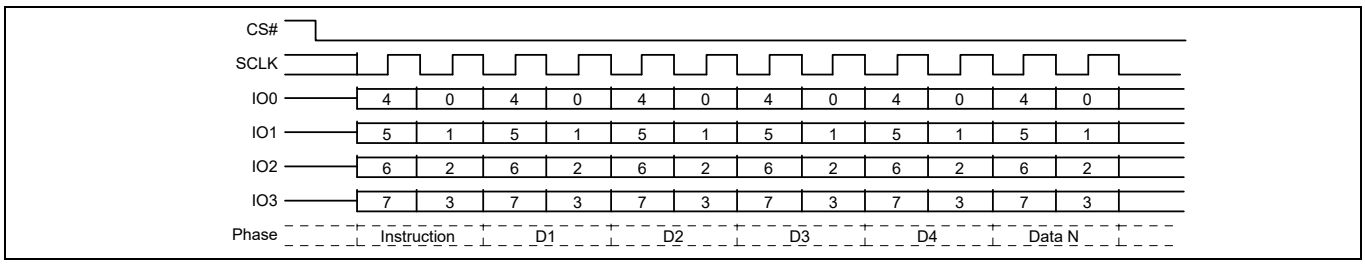



**Figure 29 Read Quad Identification (RDQID) command sequence QPI mode**

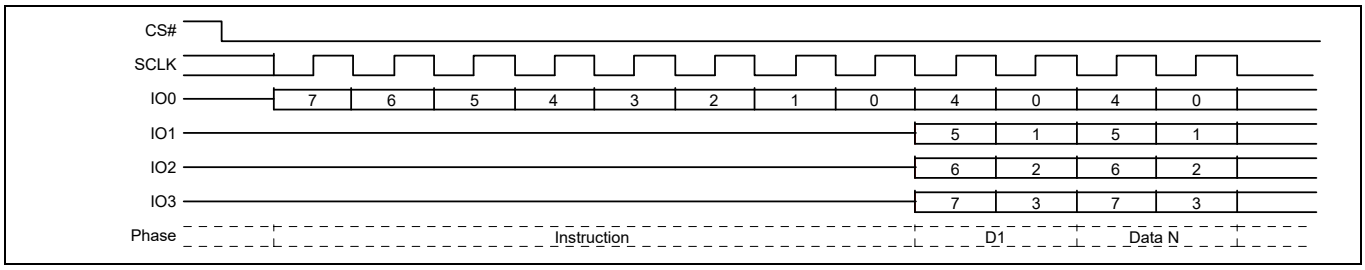



**Figure 30 Read Quad Identification (RDQID) command sequence Quad mode**

### 8.2.3 Read serial flash discoverable parameters (RSFDP 5Ah)

The command is initiated by shifting on SI the instruction code “5Ah”, followed by a 24-bit (3-byte) address or 32-bit (4-byte) address (depending on the current address length configuration of CR2V[0]), followed by the number of read latency (dummy cycles) set by the variable read latency configuration in CR3V[3:0].

The SFDP bytes are then shifted out on SO/IO1 starting at the falling edge of SCK after the dummy cycles. The SFDP bytes are always shifted out with the MSb first. If the 24-bit (3-byte) address or 32-bit (4-byte) address is set to any non-zero value, the selected location in the SFDP space is the starting point of the data read. This enables random access to any parameter in the SFDP space. In SPI mode the RSFDP command is supported up to 108 MHz.

The variable read latency should be set to 8 cycles for compliance with the JEDEC JESD216 SFDP standard. The non-volatile default variable read latency in CR3NV is set to 8 dummy cycles when the device is shipped from Infineon. However, because the RSFDP command uses the same implementation as other variable address length and latency read commands, users are free to modify the address length and latency of the command if desired.

Continuous (sequential) read is supported with the Read SFDP command.

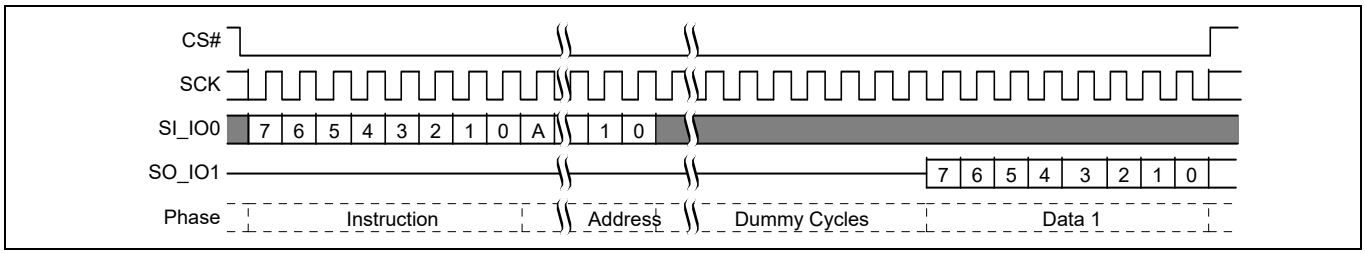



**Figure 31 RSFDP command sequence**[27]

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

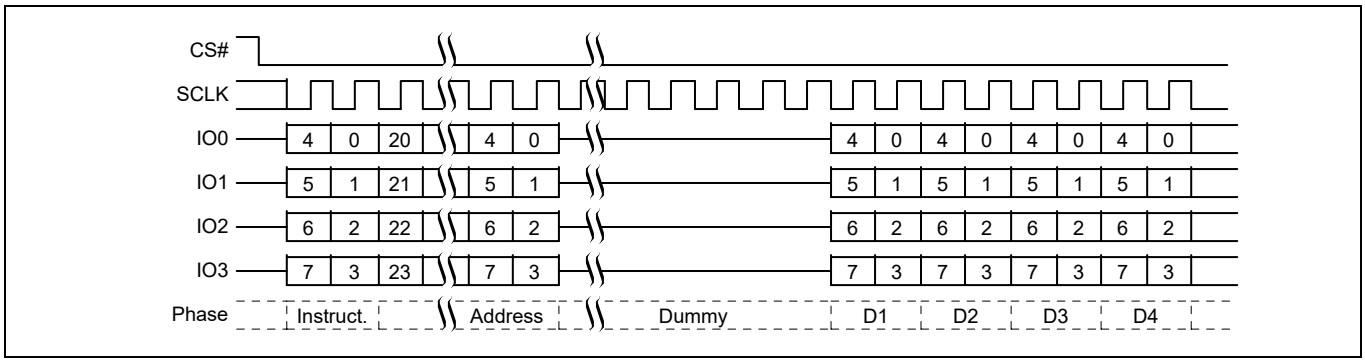



**Figure 32 RSFDP QPI mode command sequence**

**Note**
27.MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command 13h.

### 8.2.4 Read Unique ID (RUID 4Bh)

The Read Identification (RUID) command provides read access to factory set read only 64-bit number that is unique to each device.

The RUID instruction is shifted on SI followed by four dummy bytes or 16 dummy bytes QPI (32 clock cycles). This latency period (i.e., dummy bytes) allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0-IO3 are "don't care" and may be high impedance.

Then the 8-bytes of Unique ID will be shifted sequentially out on SO / IO1.

Continued shifting of output beyond the end of the defined Unique ID address space will provide undefined data. The RUID command sequence is terminated by driving CS# to the logic HIGH state anytime during data output.

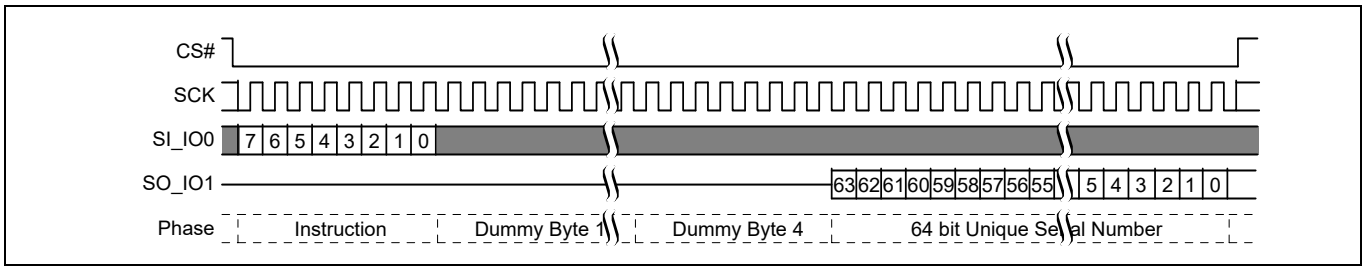



**Figure 33 Read Unique ID (RUID) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

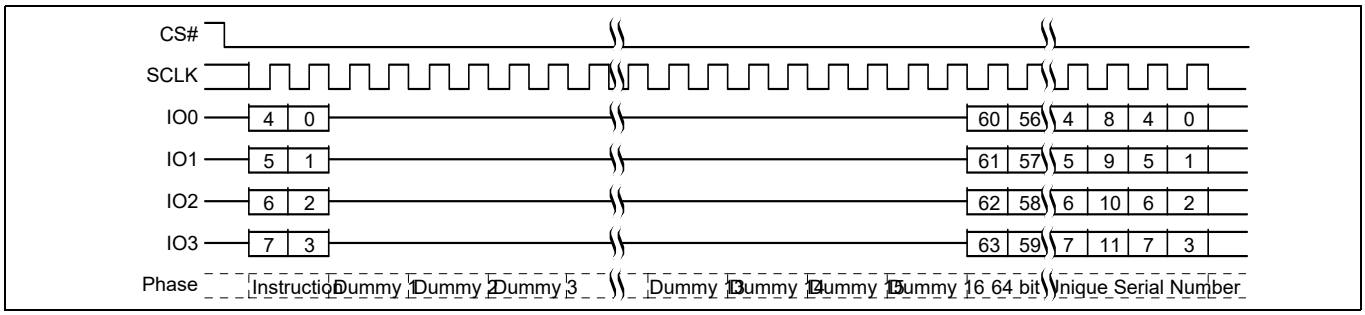



**Figure 34 Read Unique ID (RUID) QPI mode command**

## 8.3 Register Access commands

### 8.3.1 Read Status Register 1 (RDSR1 05h)

The Read Status Register 1 (RDSR1) command allows the Status Register 1 contents to be read from SO/IO1.

The volatile version of Status Register 1 (SR1V) contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read Status Register 1 continuously by providing multiples of eight clock cycles. The status is updated for each eight cycle read.

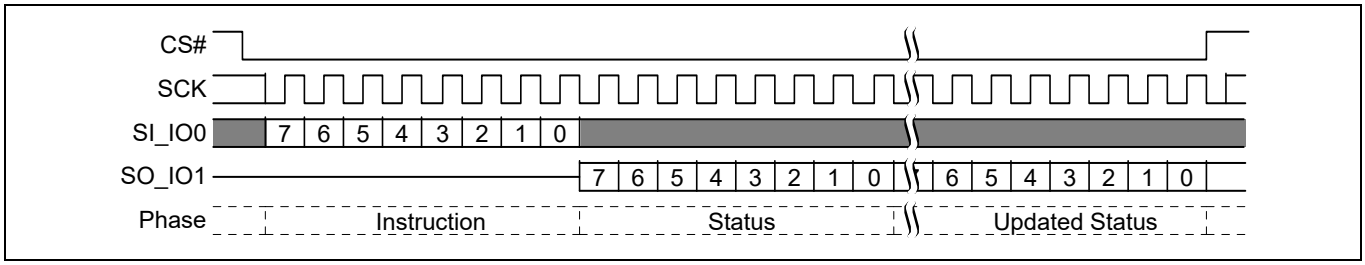



**Figure 35 Read Status Register 1 (RDSR1) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

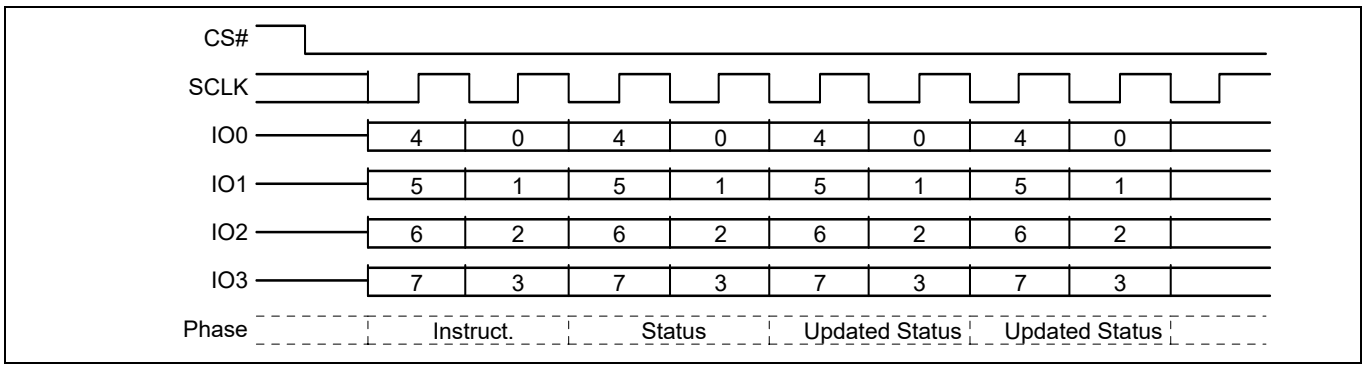



**Figure 36 Read Status Register 1 (RDSR1) QPI mode command**

### 8.3.2 Read Status Register 2 (RDSR2 07h)

The Read Status Register 2 (RDSR2) command allows the Status Register 2 contents to be read from SO/IO1.

The volatile Status Register 2 SR2V contents may be read at any time, even while a program, erase, or write operation is in progress. It is possible to read the Status Register 2 continuously by providing multiples of eight clock cycles. The status is updated for each eight cycle read.

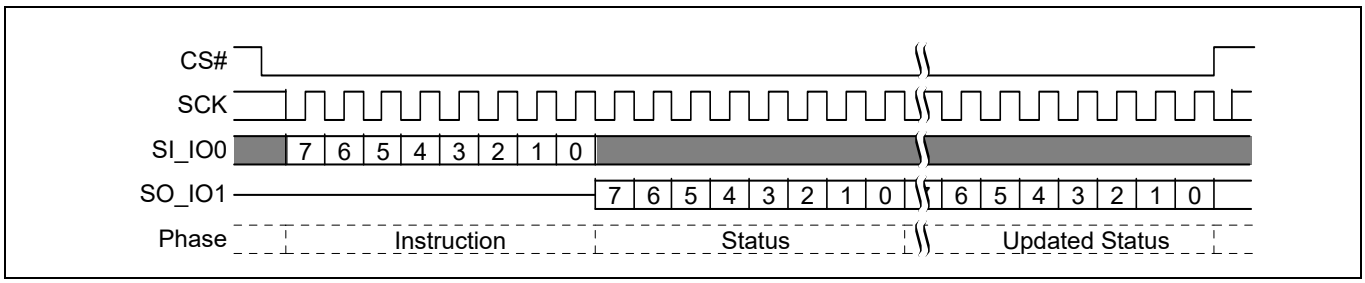



**Figure 37 Read Status Register 2 (RDSR2) command**

In QPI mode, status register 2 may be read via the Read Any Register command, see **"Read Any Register (RDAR 65h)"** on page 83.

### 8.3.3 Read Configuration Registers (RDCR1 35h) (RDCR2 15h) (RDCR3 33h)

The Read Configuration Register (RDCR1, RDCR2, RDCR3) commands allows the volatile Configuration Registers (CR1V, CR2V, CR3V) contents to be read from SO/IO1.

It is possible to read CR1V, CR2V and CR3V continuously by providing multiples of eight clock cycles. The Configuration Registers contents may be read at any time, even while a program, erase, or write operation is in progress.

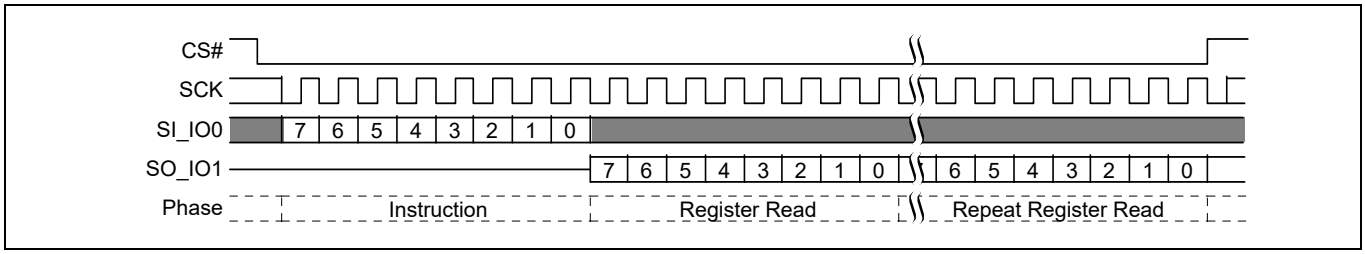



**Figure 38 Read Configuration Register (RDCR1) (RDCR2) (RDCR3) command sequence**

In QPI mode, configuration register 1, 2 and 3 may be read via the Read Any Register command, see **"Read Any Register (RDAR 65h)"** on page 83.

### 8.3.4 Write Registers (WRR 01h)

The Write Registers (WRR) command allows new values to be written to the Status Register 1, Configuration Register 1, Configuration Register 2 and Configuration Register 3. Before the Write Registers (WRR) command can be accepted by the device, a Write Enable (WREN) or Write Enable for Volatile Registers (WRENV) command must be received. After the Write Enable (WREN) command has been decoded successfully, the device will set the write enable latch (WEL) in the Status Register to enable Non-volatile Write operations and direct the values in the following WRR command to the Non-volatile SR1NV, CR1NV, CR2NV and CR3NV registers. After the Write Enable for Volatile Registers (WRENV) command has been decoded successfully, the device directs the values in the following WRR command to the volatile SR1V, CR1V, CR2V and CRV3 registers.

The Write Registers (WRR) command is entered by shifting the instruction and the data bytes on SI/IO0. The Status Register is one data byte in length.

A WRR operation directed to non-volatile registers by a preceding WREN command, first erases non-volatile registers then programs the new value as a single operation, then copies the new non-volatile values to the volatile version of the registers. A WRR operation directed to volatile registers by a preceding WRENV command, updates the volatile registers without affecting the related non-volatile register values. The Write Registers (WRR) command will set the P_ERR or E_ERR bits if there is a failure in the WRR operation. See **"Status Register 2 Volatile (SR2V)"** on page 32 for a description of the Error bits. The device hangs busy until Clear Status Register (CLSR) is used to clear the error and WIP for return to Standby. Any Status or Configuration Register bit reserved for the future must be written as a '0'.

CS# must be driven to the logic HIGH state after the eighth, sixteenth, twenty-fourth, or thirty-second bit of data has been latched. If not, the Write Registers (WRR) command is not executed. If CS# is driven HIGH after the:

- eighth cycle then only the Status Register 1 is written
- sixteenth cycle both the Status 1 and Configuration 1 Registers are written;
- twenty-fourth cycle Status 1 and Configuration 1 and 2 Registers are written;
- thirty-second cycle Status 1 and Configuration 1, 2, and 3 Registers are written.

As soon as CS# is driven to the logic HIGH state, the self-timed Write Registers (WRR) operation is initiated. While the Write Registers (WRR) operation 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 a ‘1’ during the Self-Timed Write Registers (WRR) operation, and is a ‘0’ when it is completed. When the Write Registers (WRR) operation is completed, the write enable latch (WEL) is set to ‘0’.

The WRR command is protected from a hardware and software reset, the Hardware Reset and Software Reset command are ignored and have no effect on the execution of the WRR command.

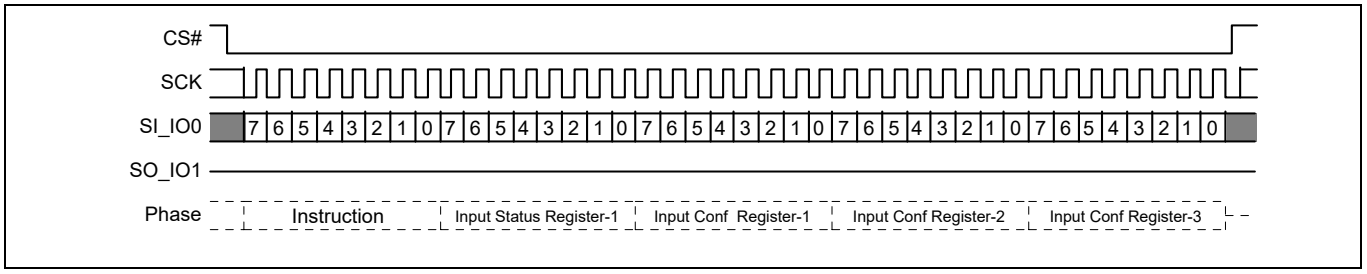



**Figure 39 Write Registers (WRR) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

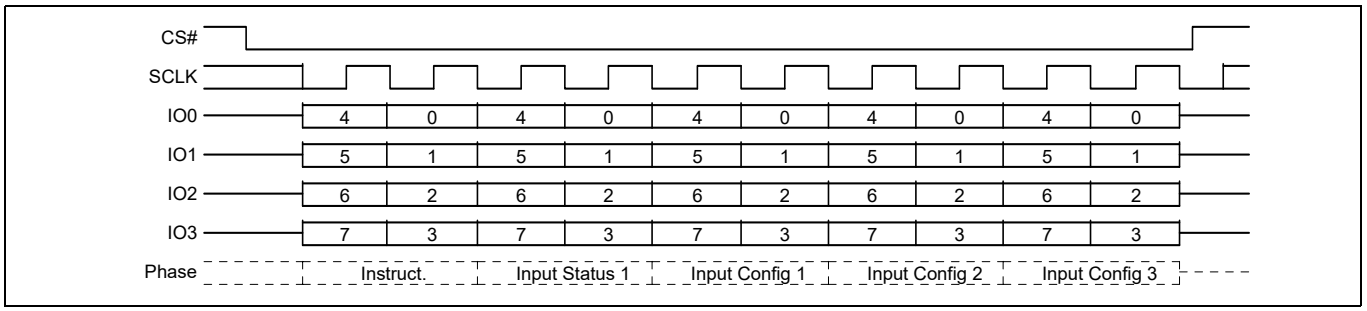



**Figure 40 Write Register (WRR) command sequence QPI mode**

The Write Registers (WRR) command allows the user to change the values of the Legacy Block Protection bits in either the non-volatile Status Register 1 or in the volatile Status Register 1, to define the size of the area that is to be treated as read-only.

The Write Registers (WRR) command also allows the user to set the Status Register Protect 0 (SRP0) bit to a ‘1’ or ‘0’. The Status Register Protect 0 (SRP0) bit and Write Protect (WP#) signal allow the BP bits to be hardware protected.

When the Status Register Protect 0 (SRP0 SR1V[7]) bit is a ‘0’, it is possible to write to the Status Register provided that the WREN or WRENV command has previously been sent, regardless of whether Write Protect (WP#) signal is driven to the logic HIGH or logic LOW state.

When the Status Register Protect 0 (SRP0) bit is set to ‘1’, two cases need to be considered, depending on the state of Write Protect (WP#):

- If Write Protect (WP#) signal is driven to the logic HIGH state, it is possible to write to the Status and Configuration Registers provided that the WREN or WRENV command has previously been sent before the WRR command.
- If Write Protect (WP#) signal is driven to the logic LOW state, it is not possible to write to the Status and Configuration Registers even if the WREN or WRENV command has previously been sent before the WRR command. Attempts to write to the Status and Configuration Registers are rejected, not accepted for execution, and no error indication is provided. As a consequence, all the data bytes in the memory area that are protected by the Legacy Block Protection bits of the Status Register, are also hardware protected by WP#.

**Note** The WP# hardware protection can be provided:

- by setting the Status Register Protect 0 (SRP0) bit after driving write protect (WP#) signal to the logic LOW state;
- or by driving Write Protect (WP#) signal to the logic LOW state after setting the Status Register Protect 0 (SRP0) bit to a '1'.

The only way to release the hardware protection is to pull the Write Protect (WP#) signal to the logic HIGH state. If WP# is permanently tied HIGH, hardware protection of the BP bits can never be activated.

Hardware protection is disabled when Quad mode is enabled (CR1V[1] = 1) or QPI mode is enabled (CR2V[3] = 1) because WP# becomes IO2; therefore, it cannot be utilized.

See **"Status Register Protect (SRP1, SRP0)"** on page 50 for a table showing the SRP and WP# control of Status and Configuration protection.

### 8.3.5 Write Enable (WREN 06h)

The Write Enable (WREN) command sets the write enable latch (WEL) bit of the Status Register 1 (SR1V[1]) to a '1'. The Write Enable Latch (WEL) bit must be set to a '1' by issuing the Write Enable (WREN) command to enable Write, Program and Erase commands.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0. Without CS# being driven to the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0, the write enable operation will not be executed.

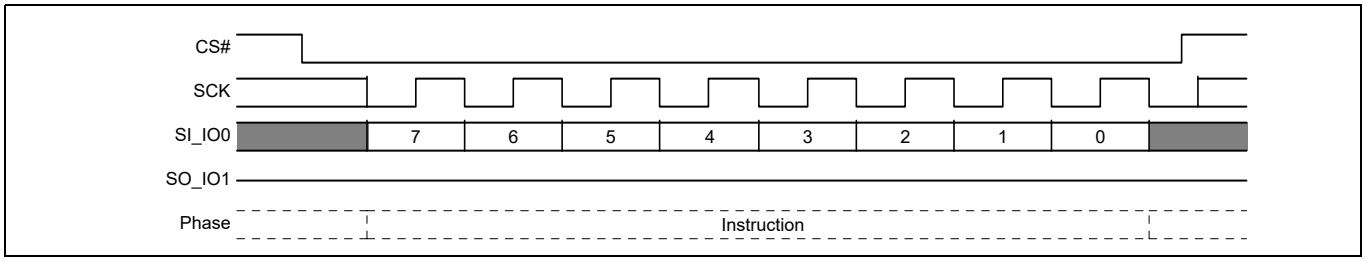



**Figure 41 Write Enable (WREN) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

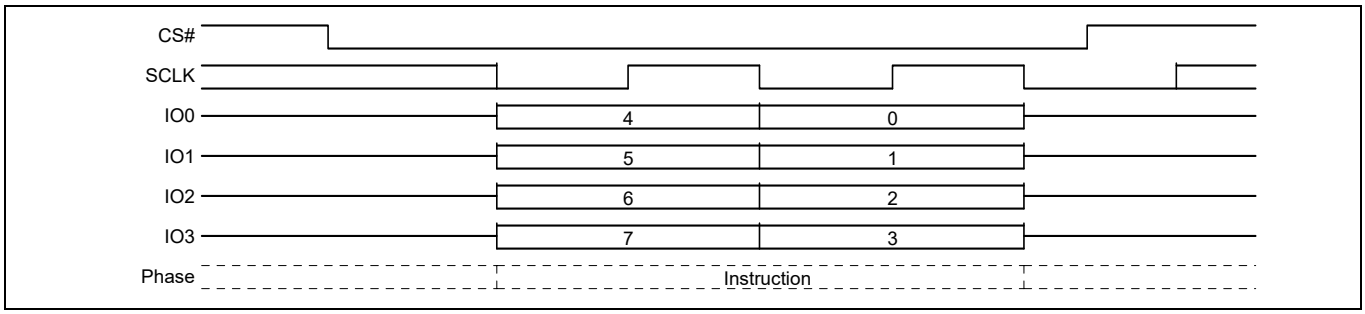



**Figure 42 Write Enable (WREN) command sequence QPI mode**

### 8.3.6 Write Disable (WRDI 04h)

The Write Disable (WRDI) command clears the Write Enable Latch (WEL) bit of the Status Register 1 (SR1V[1]) to a '0'.

The Write Enable Latch (WEL) bit may be cleared to a '0' by issuing the Write Disable (WRDI) command to disable Page Program (PP, 4PP, QPP, 4QPP), Sector Erase (SE), Half Block Erase (HBE), Block Erase (BE), Chip Erase (CE), Write Registers (WRR or WRAR), Security Region Erase (SECRE), Security Region Program (SECRP), and other commands, that require WEL be set to '1' for execution. The WRDI command can be used by the user to protect memory areas against inadvertent writes that can possibly corrupt the contents of the memory. The WRDI command is ignored during an embedded operation while WIP bit = 1.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0. Without CS# being driven to the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0, the write disable operation will not be executed.

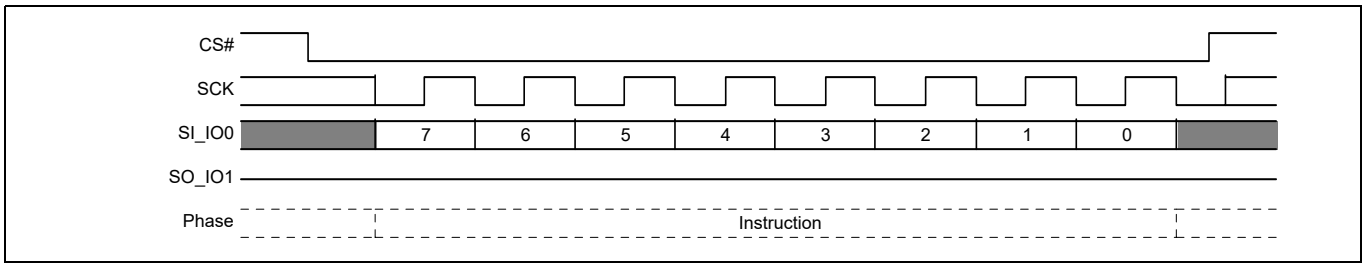



**Figure 43 Write Disable (WRDI) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

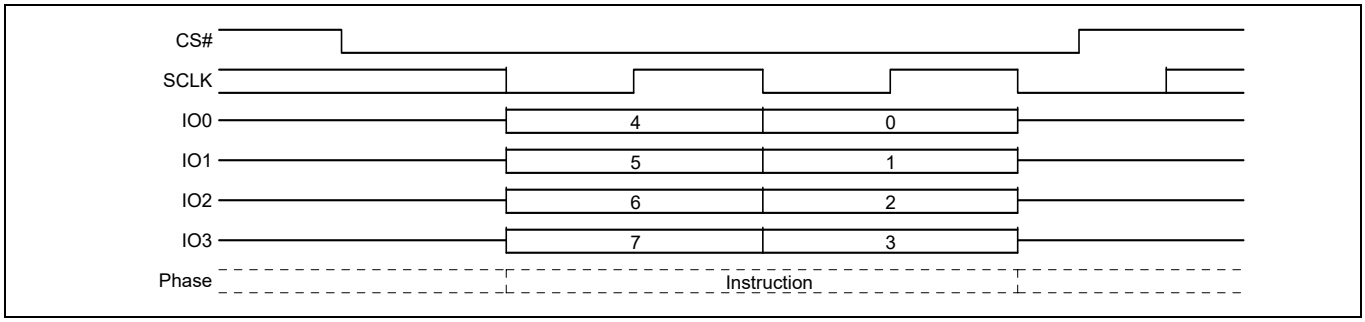



**Figure 44 Write Disable (WRDI) command sequence QPI mode**

### 8.3.7 Write Enable for Volatile Registers (WRENV 50h)

The volatile SR1V, CR1V, CR2V and CR3V registers described in **"Registers"** on page 29, can be written by sending the WRENV command followed by the WRR command. 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 or configuration non-volatile register bits. The WRENV command will not set the Write Enable Latch (WEL) bit, WRENV is used only to direct the following WRR command to change the volatile status and configuration register bit values.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0. Without CS# being driven to the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0, the write enable operation will not be executed.

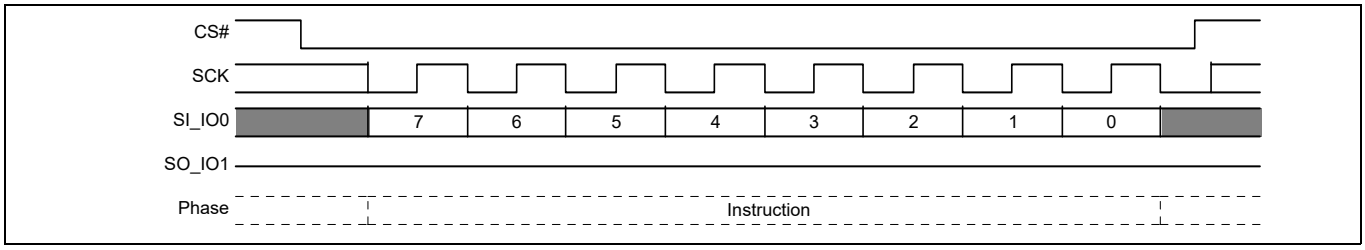



**Figure 45 Write Enable for Volatile Registers (WRENV) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

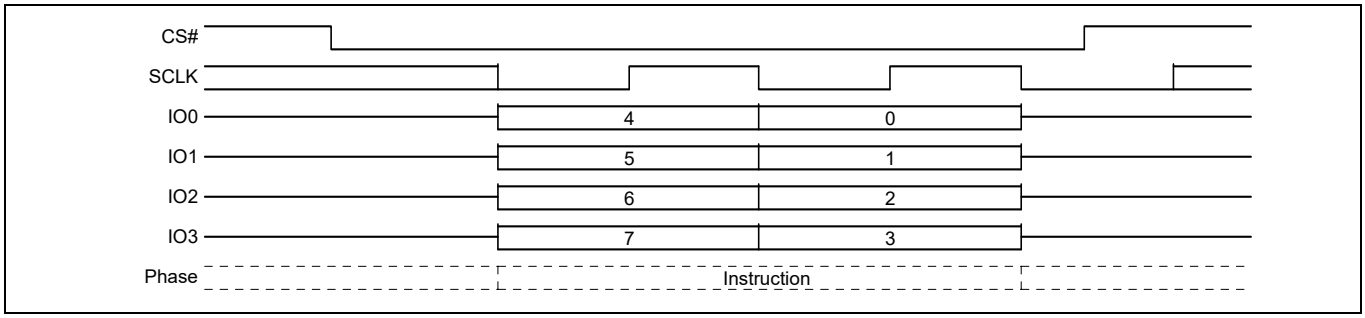



**Figure 46 Write Enable for Volatile Registers (WRENV) command sequence QPI mode**

## 8.3.8 Clear Status Register (CLSR 30h)

The Clear Status Register command clears the WIP (SR1V[0]), WEL (SR1V[1]), P_ERR (SR2V[5]), and E_ERR (SR2V[6]) bits to '0'. It is not necessary to set the WEL bit before a Clear Status Register command is executed. The Clear Status Register command will be accepted even when the device remains busy with WIP set to 1, as the device does remain busy when either Error bit is set.

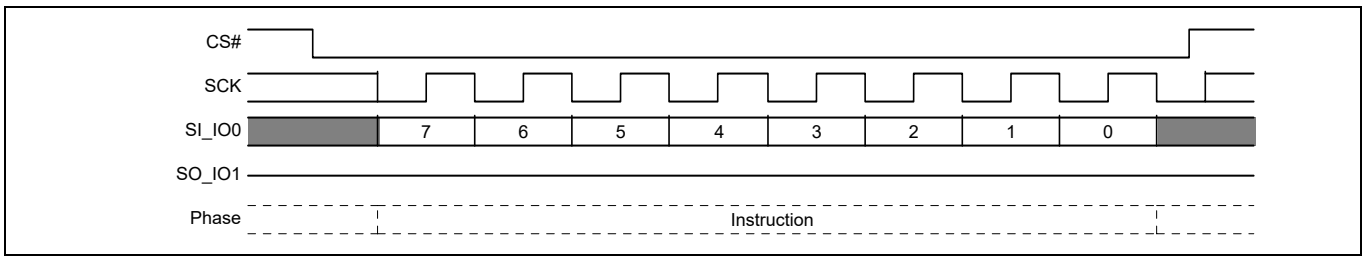



**Figure 47 Clear Status Register (CLSR) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

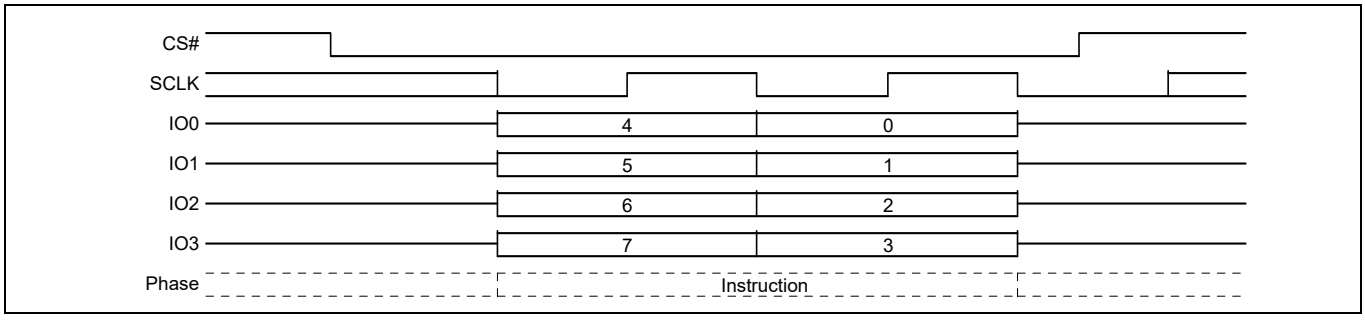



**Figure 48 Clear Status Register (CLSR) QPI mode**

### 8.3.9 Program DLRNV (PDLRNV 43h)

Before the Program DLRNV (PDLRNV) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device will set the write enable latch (WEL) to enable the PDLRNV operation.

The PDLRNV command is entered by shifting the instruction and the data byte on SI/IO0.

CS# must be driven to the logic HIGH state after the eighth (8th) bit of data has been latched. If not, the PDLRNV command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed PDLRNV operation is initiated. While the PDLRNV operation 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 a '1' during the self-timed PDLRNV cycle, and a is 0 when it is completed. The PDLRNV operation can report a program error in the P_ERR bit of the Status Register. When the PDLRNV operation is completed, the Write Enable Latch (WEL) is set to a '0'. The maximum clock frequency for the PDLRNV command is 108 MHz.

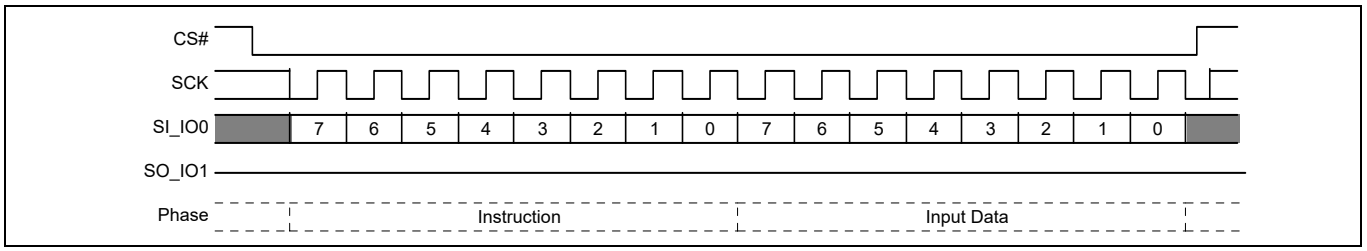



**Figure 49 Program DLRNV (PDLRNV) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

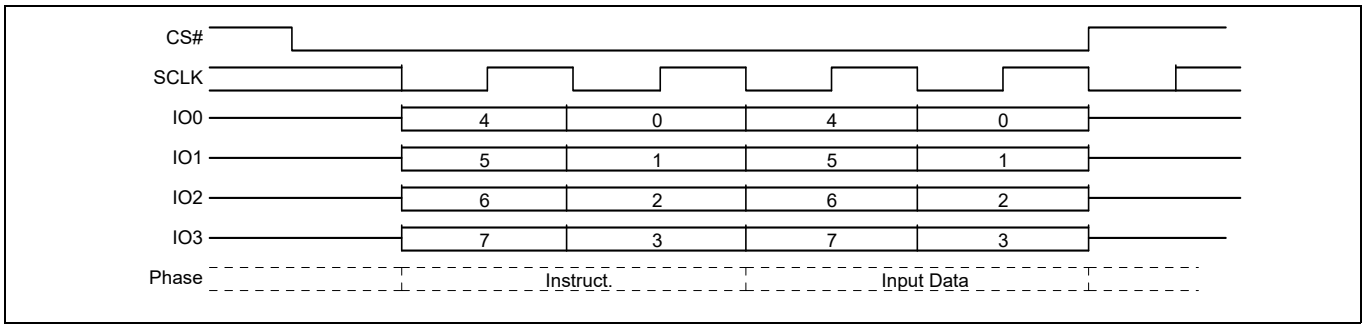



**Figure 50 Program DLRNV (PDLRNV) command sequence – QPI mode**

### 8.3.10 Write DLRV (WDLRV 4Ah)

Before the Write DLRV (WDLRV) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device will set the Write Enable Latch (WEL) to enable WDLRV operation.

The WDLRV command is entered by shifting the instruction and the data byte on SI/IO0.

CS# must be driven to the logic HIGH state after the eighth (8th) bit of data has been latched. If not, the WDLRV command is not executed. As soon as CS# is driven to the logic HIGH state, the WDLRV operation is initiated with no delays. The maximum clock frequency for the WDLRV command is 108 MHz.

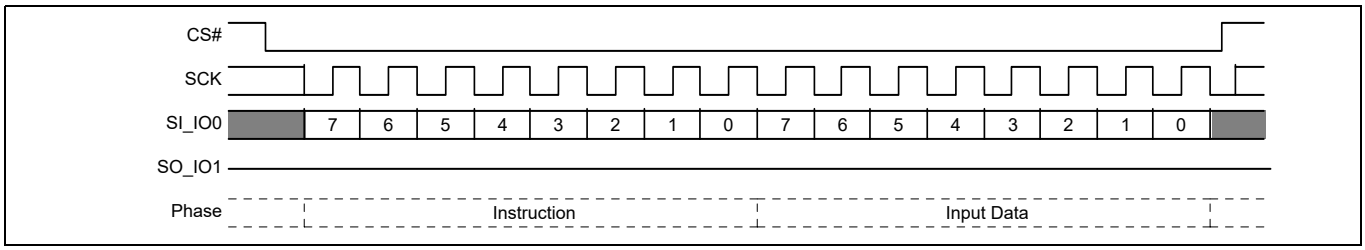



**Figure 51 Write DLRV (WDLRV) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

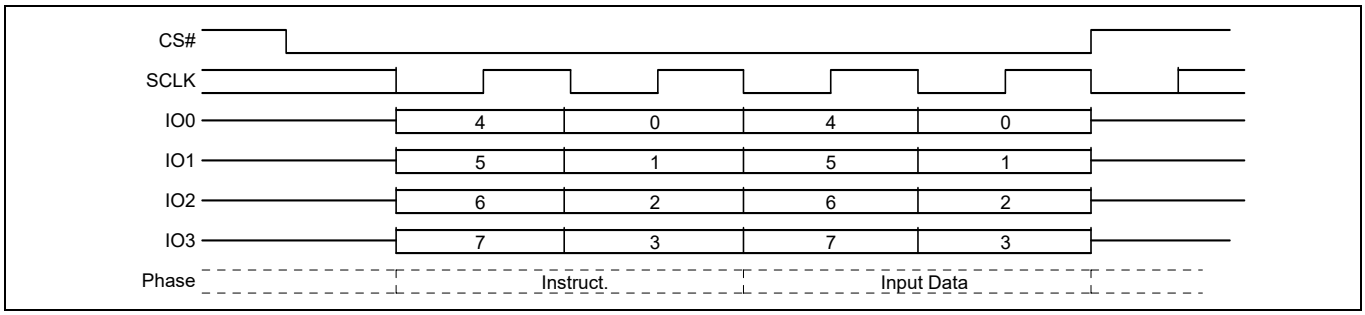



**Figure 52 Write DLRV (WDLRV) command sequence – QPI mode**

### 8.3.11 Data Learning Pattern Read (DLPRD 41h)

The instruction 41h is shifted into SI/IO0 by the rising edge of the SCK signal followed by one dummy cycle. This latency period allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance. Then the 8-bit DLP is shifted out on SO/IO1. It is possible to read the DLP continuously by providing multiples of eight clock cycles. The maximum operating clock frequency for the DLPRD command is 108 MHz.

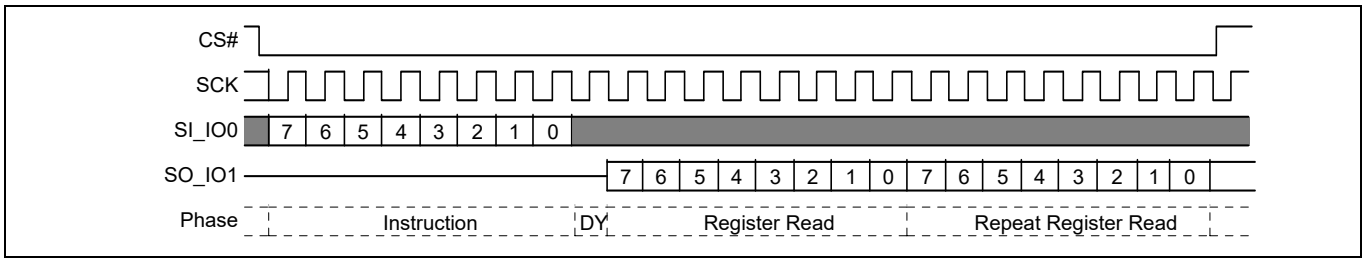



**Figure 53 DLP Read (DLPRD) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in and returning data out on IO0–IO3.

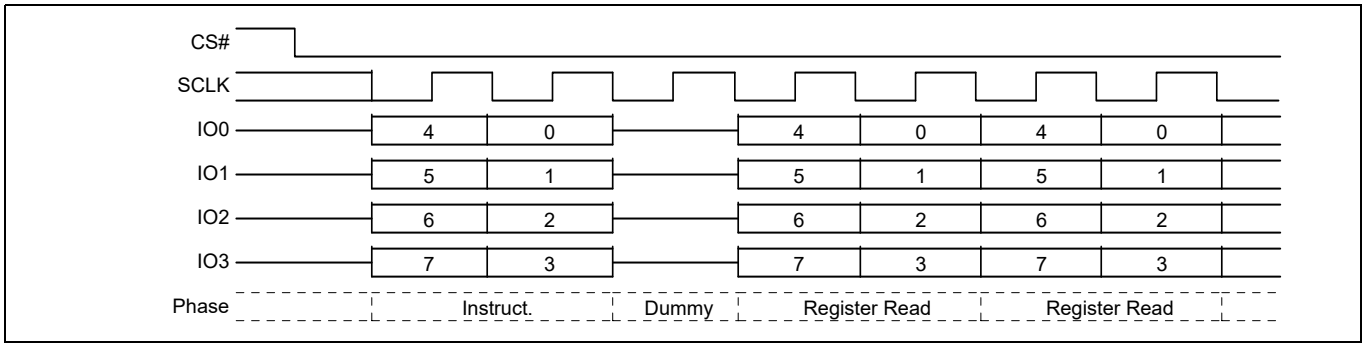



**Figure 54 DLP Read (DLPRD) command sequence – QPI mode**

### 8.3.12 Enter 4-byte Address mode (4BEN B7h)

The Enter 4-byte Address mode (4BEN) command sets the volatile address length status (ADS) bit (CR2V[0]) to 1 to change all 3-byte Address commands to require 4-bytes of address. This command will not affect 4-byte only commands which will still continue to expect 4-bytes of address.

To return to 3-byte Address mode the 4BEX command clears the Volatile Address Length bit CR2V[0] = 0). The WRAR command can also clear the volatile address length bit CR2V[0] = 0). Also, a hardware or software reset may be used to return to the 3-byte Address mode if the Non-Volatile Address Length bit CR2NV[1] = 0.

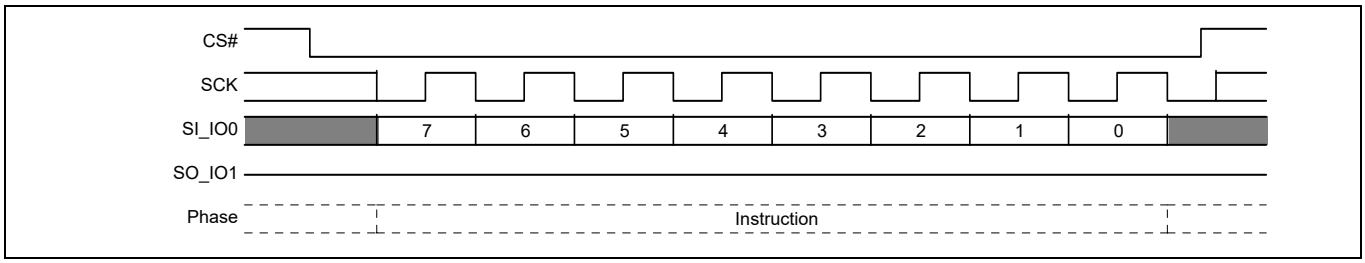



**Figure 55 Enter 4-byte Address mode (4BEN B7h) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

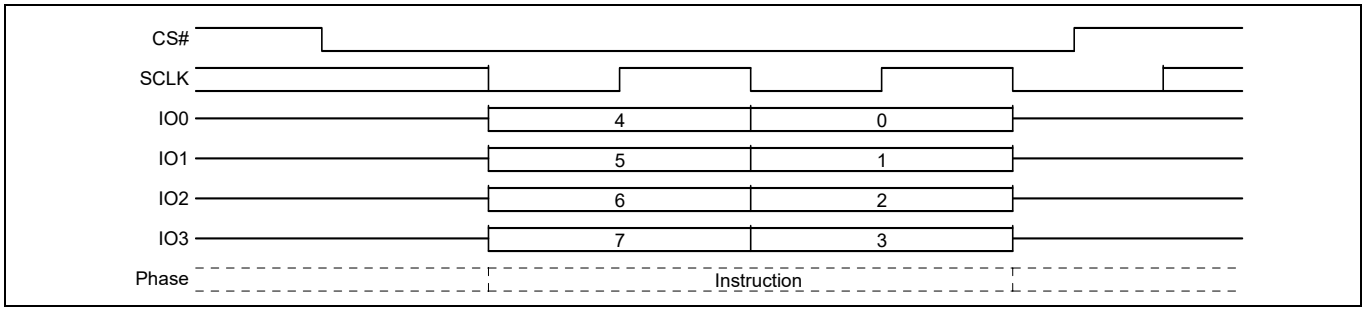



**Figure 56 Enter 4-byte Address QPI mode**

### 8.3.13 Exit 4-byte Address mode (4BEX E9h)

The exit 4-byte Address Mode (4BEX) command sets the volatile address length status (ADS) bit (CR2V[0]) to 0 to change most 4-byte Address commands to require 3-bytes of address. This command will not affect 4-byte only commands which will still continue to expect 4-bytes of address.

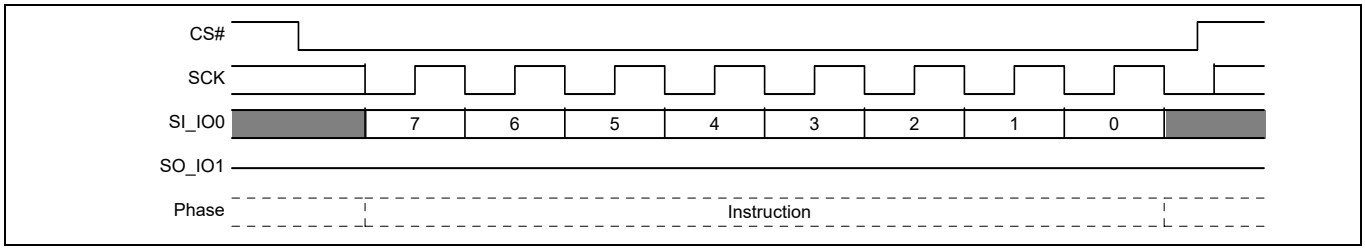



**Figure 57 Exit 4-byte Address mode (4BEX E9h) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

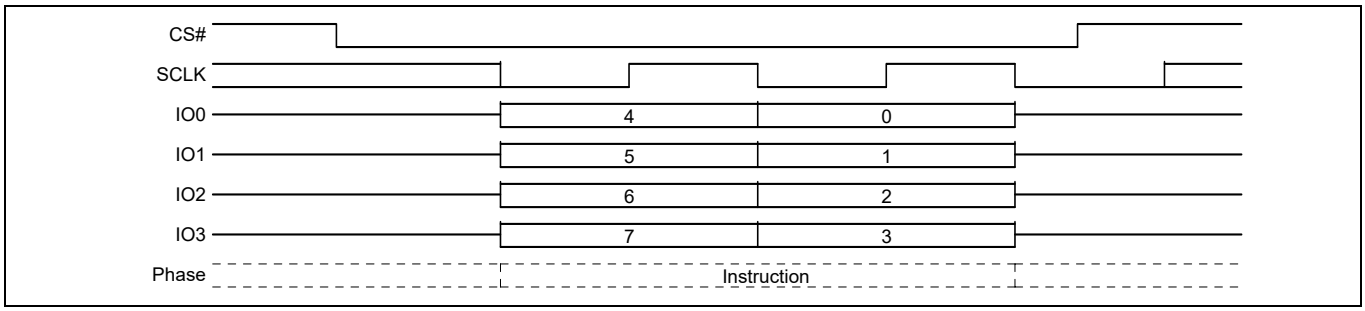



**Figure 58 Exit 4-byte Address QPI mode**

### 8.3.14 Read Any Register (RDAR 65h)

The Read Any Register (RDAR) command provides a way to read device registers. The instruction is followed by a 3 or 4-byte address (depending on the address length configuration CR2V[0]), followed by a number of latency (dummy) cycles set by CR3V[3:0]. Then the selected register contents are returned. If the read access is continued the same addressed register contents are returned until the command is terminated - only one register is read by each RDAR command.

Reading undefined locations provides undefined data.

The RDAR command may be used during embedded operations to Read Status Register 1 (SR1V).

The RDAR command is not used for reading registers that act as a window into a larger array: IBLAR. There are separate commands required to select and read the location in the array accessed.

The RDAR command will read invalid data from the PASS register locations if the IRP Password Protection mode is selected by programming IRP[2] to 0.

**Table 34 Register address map**

| Byte Address (hex) | Register name | Description |
|---|---|---|
| 000000 | SR1NV | Non-volatile Status and Configuration Registers reading of Non-volatile Status and Configuration Registers actually reads the volatile registers |
| 000001 | N/A | |
| 000002 | CR1NV | |
| 000003 | CR2NV | |
| 000004 | CR3NV | |
| 000005 | NVDLP | |
| ... | N/A | – |
| 000020 | PASS[7:0] | Non-volatile Password Register |
| 000021 | PASS[15:8] | |
| 000022 | PASS[23:16] | |
| 000023 | PASS[31:24] | |
| 000024 | PASS[39:32] | |
| 000025 | PASS[47:40] | |
| 000026 | PASS[55:48] | |
| 000027 | PASS[63:56] | |
| ... | N/A | – |
| 000030 | IRP[7:0] | Non-volatile |
| 000031 | IRP[15:8] | |
| ... | N/A | – |
| 000039 | PRPR[A15:A8] | Pointer Region Protection Register A15:A8 |
| 00003A | PRPR[A23:A16] | Pointer Region Protection Register A23:A16 |
| 00003B | N/A | – |
| ... | N/A | – |
| 800000 | SR1V | Volatile Status and Configuration Registers |
| 800001 | SR2V | |
| 800002 | CR1V | |
| 800003 | CR2V | |
| 800004 | CR3V | |
| 800005 | VDLP | |
| ... | N/A | – |
| 800040 | PR | Volatile Protection Register |
| ... | N/A | – |

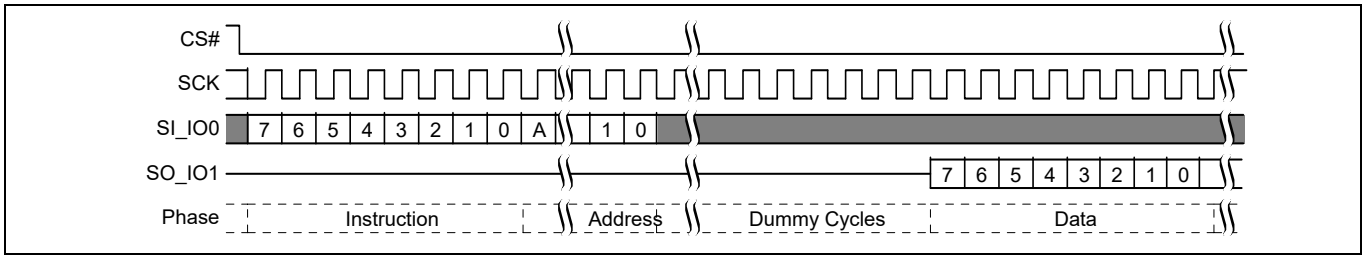



**Figure 59 Read Any Register Read command sequence**[28]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in and returning data out on IO0–IO3.

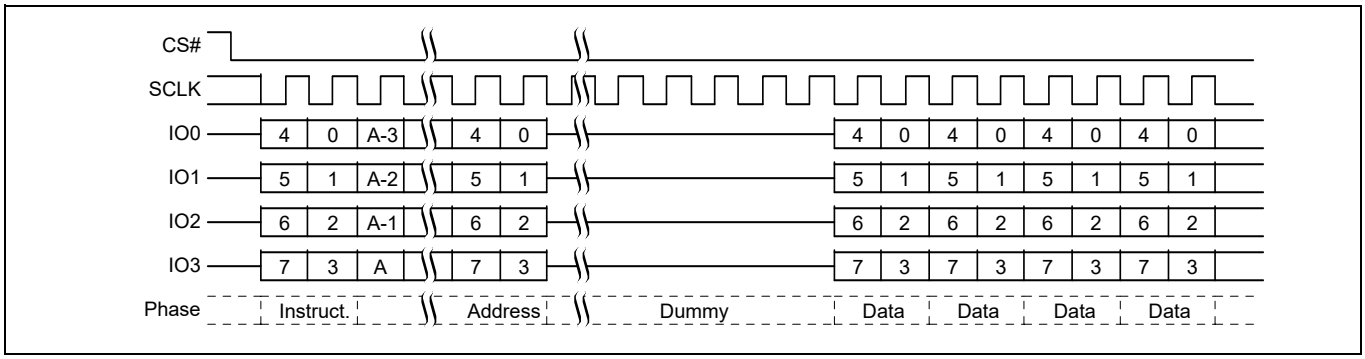



**Figure 60 Read Any Register, QPI mode, command sequence**[28]

**Note**
28.A = MSb of address = 23 for address length CR2V[0] = 0, or 31 for CR2V[0] = 1.

### 8.3.15 Write Any Register (WRAR 71h)

The Write Any Register (WRAR) command provides a way to Write Any Device Register - non-volatile or volatile. The instruction is followed by a 3 or 4-byte address (depending on the address length configuration CR2V[0]), followed by one byte of data to write in the address selected register.

Before the WRAR command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations. The WIP bit in SR1V may be checked to determine when the operation is completed. The P_ERR and E_ERR bits in SR2V may be checked to determine if an error occurred during the operation.

Some registers have a mixture of bit types and individual rules controlling which bits may be modified. Some bits are read only, some are OTP.

Read Only bits are never modified and the related bits in the WRAR command data byte are ignored without setting a program or erase error indication (P_ERR or E_ERR in SR2V). Hence, the value of these bits in the WRAR data byte do not matter.

OTP bits may only be programmed to the level opposite of their default state. Writing of OTP bits back to their default state is ignored and no error is set.

Non-volatile bits which are changed by the WRAR data, require non-volatile register write time ($t_W$) to be updated. The update process involves an erase and a program operation on the non-volatile register bits. If either the erase or program portion of the update fails the related Error bit in SR2V and WIP in SR1V will be set to 1.

Volatile bits which are changed by the WRAR data, require the volatile register write time ($t_{CS}$) to be updated.

Status Register 1 may be repeatedly read (polled) to monitor the Write-in Progress (WIP) bit (SR1V[0]) to determine when the register write is completed and Status Register 1 for the Error bits (SR2V[6,5]) to determine if there is write failure. If there is a write failure, the Clear Status command is used to clear the error status and enable the device to return to STANDBY state. When the WRAR operation is completed, the write enable latch (WEL) is set to a ‘0’.

However, the PR register can not be written by the WRAR command. The PR register contents are treated as read Only bits. Only the NVLOCK Bit Write (PRL) command can write the PR register.

The WRAR command to write the SR1NV, CR1NV CR2NV and CR3NV is protected from a Hardware and Software Reset, the WRAR command to all other register are reset from a Hardware or Software Reset.

The WRAR command sequence and behavior is the same as the PP or 4PP command with only a single byte of data provided. See **“Page Program (PP 02h or 4PP 12H)”** on page 100.

The address map of the registers is the same as shown for **Table 34**.

### 8.3.16 Set Burst Length (SBL 77h)

The Set Burst Length (SBL) command is used to configure the burst wrap feature. Burst wrap is used in conjunction with Quad I/O Read and DDR Quad I/O read, in QIO or QPI modes, to access a fixed length and alignment of data. Certain applications can benefit from this feature by improving the overall system code execution performance. The Burst Wrap feature allows applications that use cache, to start filling a cache line with instruction or data from a critical address first, then fill the remainder of the cache line afterwards within a fixed length (8/16/32/64-bytes) of data, without issuing multiple read commands.

The Set burst length command is initiated by driving the CS# pin LOW and then shifting the instruction code “77h” followed by 24 dummy bits and 8 “wrap length bits (WL[7]–WL[0])”. The command sequence is shown in **Figure 61** and **Figure 62**. Wrap Length bit WL[7] and the lower nibble WL[3:0] are not used. See Configuration Register 3 (CR3V[6:4]) for the encoding of WL[6]–WL[4] in **“Configuration Register 3”** on page 39.

Once WL[6:4] is set by a Set Burst Length command, all the following “Quad I/O Read” commands will use the WL[6:4] setting to access the 8/16/32/64-byte section of data. Note, Configuration Register 1 Quad bit CR1V[1] or Configuration Register 2 QPI bit CR2V[3] must be set to 1 in order to use the Quad I/O Read and Set Burst Length commands. To exit the “wrap around” function and return to normal Read operation, another Set Burst with Wrap command should be issued to set WL4 = 1. The default value of WL[6:4] upon power on, hardware or software reset as set in the CR2NV[6:4]. Use WRR or WRAR command to set the default wrap length in CR2NV[6:4].

The Set Burst Length (SBL) command writes only to CR3V[6:4] bits to enable or disable the wrapped read feature and set the wrap boundary. The SBL command cannot be used to set the read latency in CR3V[3:0]. The WRAR command must be used to set the read latency in CR3V or CR3NV.

See **Table 35** for CR3V[6:5] values for wrap boundary’s and start address. When enabled the wrapped read feature changes the related read commands from sequentially reading until the command ends, to reading sequentially wrapped within a group of bytes.

When the Wrap mode is not enabled (**Table 16** and **Table 19**), an unlimited length sequential read is performed.

When the Wrap mode is enabled (**Table 16** and **Table 19**) a fixed length and aligned group of 8-, 16-, 32-, or 64-bytes is read starting at the byte address provided by the read command and wrapping around at the group alignment boundary.

The group of bytes is of length and aligned on an 8-, 16-, 32-, or 64-byte boundary. CR3V[6:5] selects the boundary. See **“Configuration Register 3 Volatile (CR3V)”** on page 43.

The starting address of the Read command selects the group of bytes and the first data returned is the addressed byte. Bytes are then read sequentially until the end of the group boundary is reached. If the Read continues the address wraps to the beginning of the group and continues to read sequentially. This wrapped read sequence continues until the command is ended by CS# returning HIGH.

**Table 35 Example burst wrap sequences**

| CR3V value (hex) | Wrap boundary (bytes) | Start address (hex) | Address sequence (hex) |
|---|---|---|---|
| 1X | Sequential | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, ... |
| 00 | 8 | XXXXXX00 | 00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, ... |
| 00 | 8 | XXXXXX07 | 07, 00, 01, 02, 03, 04, 05, 06, 07, 00, 01, ... |
| 01 | 16 | XXXXXX02 | 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03, ... |
| 01 | 16 | XXXXXX0C | 0C, 0D, 0E, 0F, 00, 01, 02, 03, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, ... |
| 02 | 32 | XXXXXX0A | 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, ... |
| 02 | 32 | XXXXXX1E | 1E, 1F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 00, ... |
| 03 | 64 | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 00, 01, 02, ... |
| 03 | 64 | XXXXXX2E | 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D,, ... |

The power-on reset, hardware reset, or software reset default burst length can be changed by programming CR3NV with the desired value using the WRAR command.

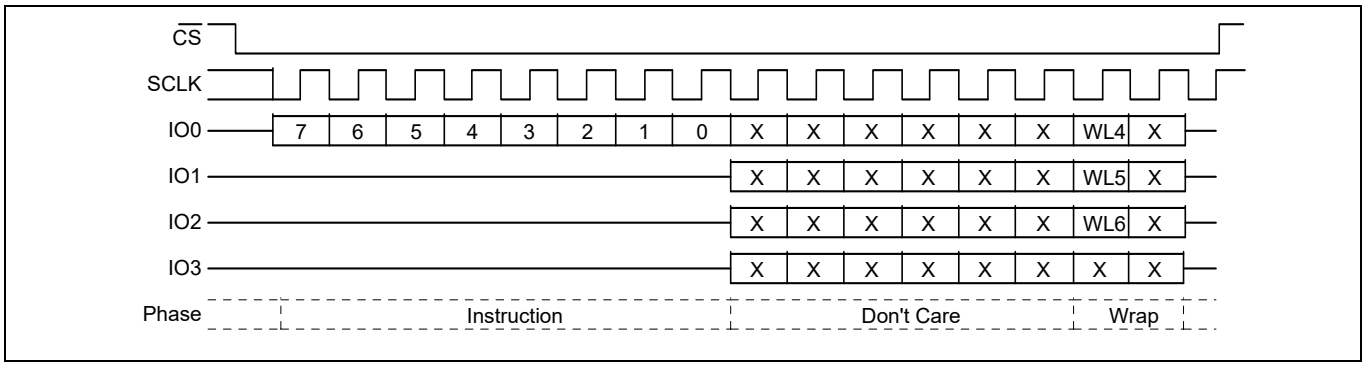



**Figure 61 Set Burst Length command sequence Quad I/O mode**

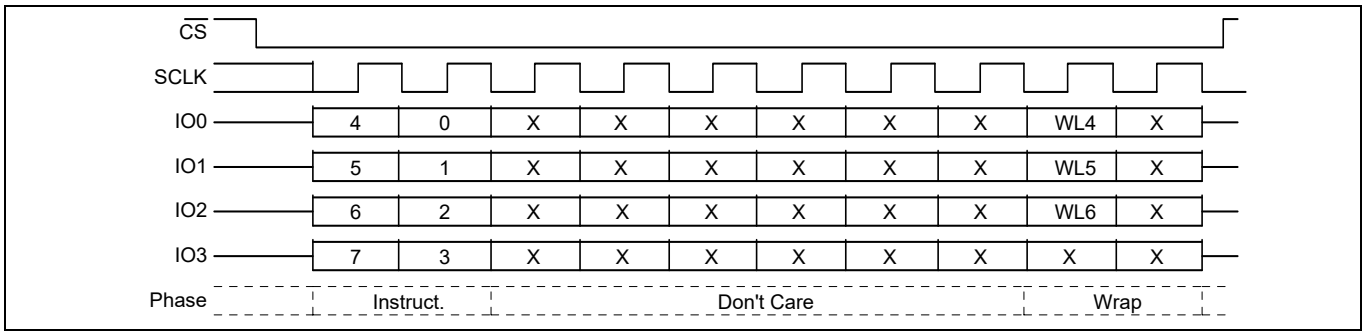



**Figure 62 Set Burst Length command sequence QPI mode**

### 8.3.17 Enter QPI mode (QPIEN 38h)

The Enter QPI Mode (QPIEN) command enables the QPI mode by setting the Volatile QPI bit (CR2V[3] = 1). See **Table 14**. The time required to enter QPI mode is $t_{QEN}$, see **Table 54**, no other commands are allowed during the $t_{QEN}$ transition time to QPI mode.

To return to SPI mode the QPIEX command or a write to register (CR2V[3] = 0) is required. A power on reset, hardware, or software reset will also return the part to SPI mode if the non-volatile QPI (CR2NV[3] = 0). See **Table 12**.

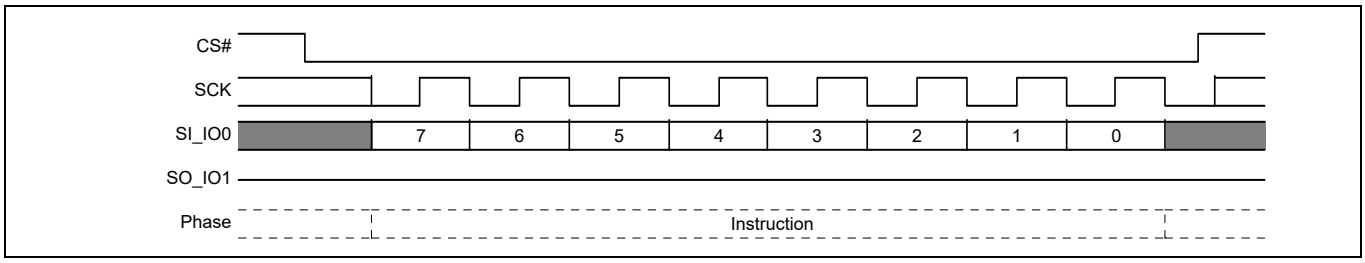



**Figure 63 Enter QPI Mode (QPIEN 38h) command sequence**

### 8.3.18 Exit QPI mode (QPIEX F5h)

The Exit QPI mode (QPIEX) command disables the QPI mode by setting the Volatile QPI bit (CR2V[3] = 0) and returning to SPI mode. See **Table 14**. The time required to exit QPI mode is $t_{QEX}$, see **Table 54**, no other commands are allowed during the $t_{QEX}$ transition time to exit the QPI mode.

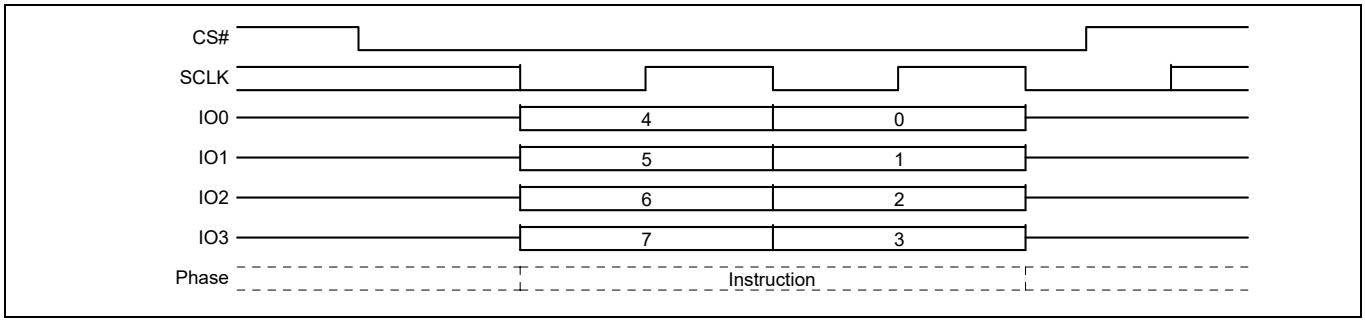



**Figure 64 Exit QPI (QPIEX F5h) command sequence**

## 8.4 Read Memory Array commands

Read commands for the main flash array provide many options for prior generation SPI compatibility or enhanced performance SPI:

- Some commands transfer address or data on each rising edge of SCK. These are called Single Data Rate commands (SDR).
- Some SDR commands transfer address one bit per falling edge of SCK and return data 1-bit of data per rising edge of SCK. These are called Single Width commands.
- Some SDR commands transfer both address and data 2 or 4 bits per rising edge of SCK. These are called Dual I/O for 2-bit, Quad I/O, and QPI for 4-bit. QPI also transfers instructions 4 bits per rising edge.
- Some commands transfer address and data on both the rising edge and falling edge of SCK. These are called Double Data Rate (DDR) commands.
- There are DDR commands for 4 bits of address or data per SCK edge. These are called Quad I/O DDR and QPI DDR for 4 bit per edge transfer.

All of these commands, except QPI Read, begin with an instruction code that is transferred one bit per SCK rising edge. QPI Read transfers the instruction 4 bits per SCK rising edge.The instruction is followed by either a 3- or 4-byte address transferred at SDR or DDR. Commands transferring address or data 2 or 4 bits per clock edge are called Multiple I/O (MIO) commands. For FL-L family devices at 256 Mb or higher density, the traditional SPI 3-byte addresses are unable to directly address all locations in the memory array. Separate 4-byte Address Read commands are provided for access to the entire address space. These devices may be configured to take a 4-byte address from the host system with the traditional 3-byte Address commands. The 4-byte Address mode for traditional commands is activated by setting the Address Length bit in Configuration Register 2 to ‘1’. In the S25FL128L higher order address bits above A22 in the 4-byte address commands, or commands using 4-byte Address mode are not relevant and are ignored because the flash array is only 64 Mb in size.

The Dual I/O, Quad I/O and QPI commands provide a performance improvement option controlled by mode bits that are sent following the address bits. The mode bits indicate whether the command following the end of the current read will be another read of the same type, without an instruction at the beginning of the read. These mode bits give the option to eliminate the instruction cycles when doing a series of dual or quad read accesses.

Some commands require delay cycles following the address or mode bits to allow time to access the memory array - read latency. The delay or read latency cycles are traditionally called dummy cycles. The dummy cycles are ignored by the memory thus any data provided by the host during these cycles is “don’t care” and the host may also leave the SI signal at high impedance during the dummy cycles. When MIO commands are used the host must stop driving the IO signals (outputs are high impedance) before the end of last dummy cycle. When DDR commands are used the host must not drive the I/O signals during any dummy cycle. The number of dummy cycles varies with the SCK frequency or performance option selected via the Configuration Register 2 (CR3V[3:0]) latency code. Dummy cycles are measured from SCK falling edge to next SCK falling edge. SPI outputs are traditionally driven to a new value on the falling edge of each SCK. Zero dummy cycles means the returning data is driven by the memory on the same falling edge of SCK that the host stops driving address or mode bits.

The DDR commands may optionally have an 8 edge Data Learning Pattern (DLP) driven by the memory, on all data outputs, in the dummy cycles immediately before the start of data. The DLP can help the host memory controller determine the phase shift from SCK to data edges so that the memory controller can capture data at the center of the data eye.

When using SDR I/O commands at higher SCK frequencies (>50 MHz), an LC that provides 1 or more dummy cycles should be selected to allow additional time for the host to stop driving before the memory starts driving data, to minimize I/O driver conflict. When using DDR I/O commands with the DLP enabled, an LC that provides 5 or more dummy cycles should be selected to allow 1 cycle of additional time for the host to stop driving before the memory starts driving the 4 cycle DLP.

Each Read command ends when CS# is returned High at any point during data return. CS# must not be returned High during the mode or dummy cycles before data returns as this may cause Mode bits to be captured incorrectly; making it indeterminate as to whether the device remains in Continuous Read mode.

### 8.4.1 Read (read 03h or 4READ 13h)

The instruction

- 03h (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 03h (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 13h is followed by a 4-byte address (A31–A0)

Then the memory contents, at the address given, are shifted out on SO/IO1.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

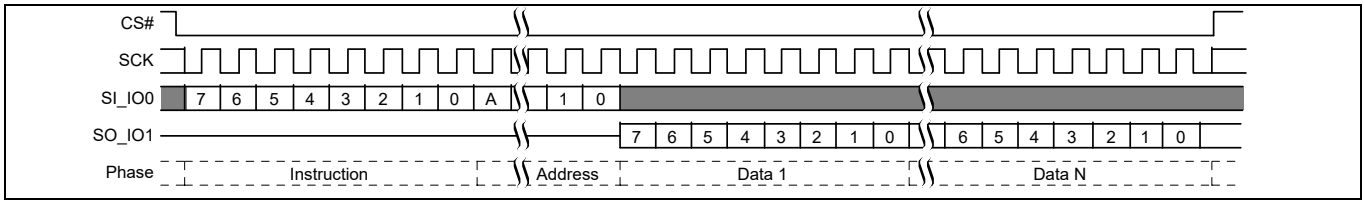



**Figure 65 Read command sequence**[29]

### 8.4.2 Fast Read (FAST_READ 0Bh or 4FAST_READ 0Ch)

The instruction

- 0Bh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 0Bh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 0Ch is followed by a 4-byte address (A31–A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR3V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on SO/IO1 is “don’t care” and may be high impedance. Then the memory contents, at the address given, are shifted out on SO/IO1.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

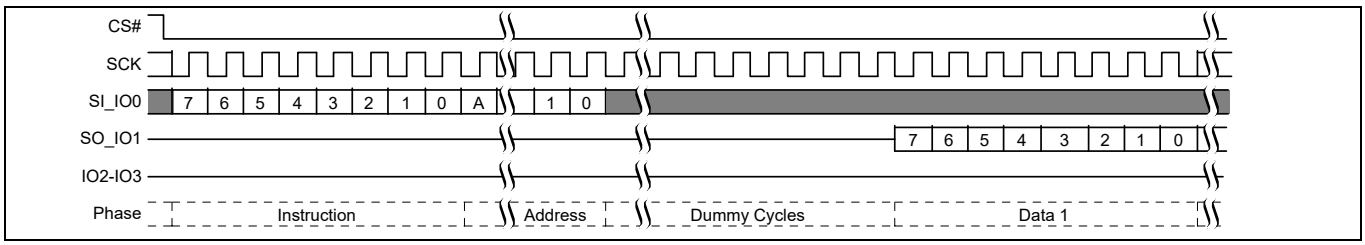



**Figure 66 Fast Read (FAST_READ) command sequence**

**Note**
29.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command 13h.

### 8.4.3 Dual Output Read (DOR 3Bh or 4DOR 3Ch)

The instruction

- 3Bh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 3Bh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 3Ch is followed by a 4-byte address (A31–A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR3V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on IO0 (SI) and IO1 (S0) is “don’t care” and may be high impedance.

Then the memory contents, at the address given, is shifted out two bits at a time through IO0 (SI) and IO1 (SO). Two bits are shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

For Dual Output Read commands, there are dummy cycles required after the last address bit is shifted into IO0 (SI) before data begins shifting out of IO0 and IO1.

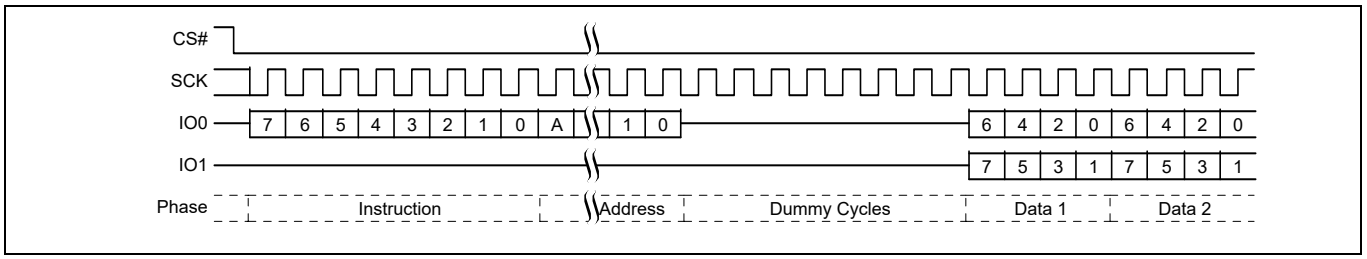



**Figure 67 Dual Output Read command sequence**[30]

**Note**
30.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command 3Ch.

### 8.4.4 Quad Output Read (QOR 6Bh or 4QOR 6Ch)

The instruction

- 6Bh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 6Bh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 6Ch is followed by a 4-byte address (A31–A0)

The address is followed by dummy cycles depending on the latency code set in the Configuration Register CR3V[3:0]. The dummy cycles allow the device internal circuits additional time for accessing the initial address location. During the dummy cycles the data value on IO0–IO3 is “don’t care” and may be high impedance.

Then the memory contents, at the address given, is shifted out four bits at a time through IO0–IO3. Each nibble (4 bits) is shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

For quad output read commands, there are dummy cycles required after the last address bit is shifted into IO0 before data begins shifting out of IO0–IO3.

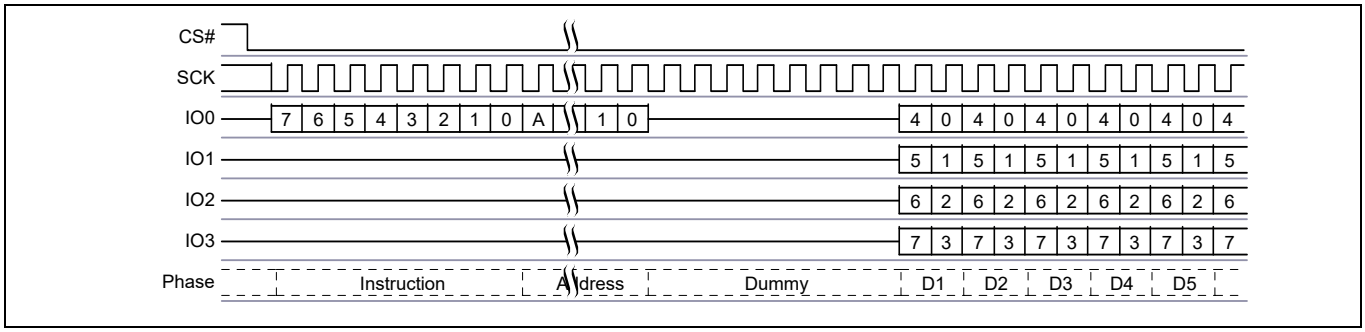



**Figure 68 Output Read command sequence**[31]

**Note**
31.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command 6Ch.

### 8.4.5 Dual I/O Read (DIOR BBh or 4DIOR BCh)

The instruction

- BBh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- BBh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- BCh is followed by a 4-byte address (A31–A0)

The Dual I/O Read commands improve throughput with two I/O signals — IO0 (SI) and IO1 (SO). This command takes input of the address and returns read data two bits per SCK rising edge. In some applications, the reduced address input and data output time might allow for code execution in place (XIP) i.e. directly from the memory device.

The Dual I/O Read command has Continuous Read Mode bits that follow the address so, a series of Dual I/O Read commands may eliminate the 8-bit instruction after the first Dual I/O Read command sends a mode bit pattern of Axh that indicates the following command will also be a Dual I/O Read command. The first Dual I/O Read command in a series starts with the 8-bit instruction, followed by address, followed by four cycles of mode bits, followed by an optional latency period. If the mode bit pattern is Axh the next command is assumed to be an additional Dual I/O Read command that does not provide instruction bits. That command starts with address, followed by mode bits, followed by optional latency.

Variable latency may be added after the Mode bits are shifted into SI and SO before data begins shifting out of IO0 and IO1. This latency period (dummy cycles) allows the device internal circuitry enough time to access data at the initial address. During the dummy cycles, the data value on SI and SO are “don’t care” and may be high impedance. The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR3V[3:0].

The continuous read feature removes the need for the Instruction bits in a sequence of read accesses and greatly improves code execution (XIP) performance. The upper nibble (bits 7–4) of the mode bits control the length of the next Dual I/O Read command through the inclusion or exclusion of the first byte instruction code. The lower nibble (bits 3-0) of the mode bits are “don’t care” (“x”) and may be high impedance. If the Mode bits equal Axh, then the device remains in dual I/O Continuous Read mode and the next address can be entered (after CS# is raised high and then asserted low) without the BBh or BCh instruction, as shown in **Figure 70**; thus, eliminating eight cycles of the command sequence. The following sequences will release the device from dual I/O Continuous Read mode; after which, the device can accept standard SPI commands:

- During the dual I/O continuous read command sequence, if the Mode bits are any value other than Axh, then the next time CS# is raised HIGH the device will be released from dual I/O Continuous Read mode.
- Send the Mode Reset command.

Note that the four mode bit cycles are part of the device’s internal circuitry latency time to access the initial address after the last address cycle that is clocked into IO0 (SI) and IO1 (SO).

It is important that the I/O signals be set to high-impedance at or before the falling edge of the first data out clock. At higher clock speeds the time available to turn off the host outputs before the memory device begins to drive (bus turn around) is diminished. It is allowed and may be helpful in preventing I/O signal contention, for the host system to turn off the I/O signal outputs (make them high impedance) during the last two “don’t care” mode cycles or during any dummy cycles.

Following the latency period the memory content, at the address given, is shifted out two bits at a time through IO0 (SI) and IO1 (SO). Two bits are shifted out at the SCK frequency at the falling edge of SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

CS# should not be driven HIGH during mode or dummy bits as this may make the mode bits indeterminate.

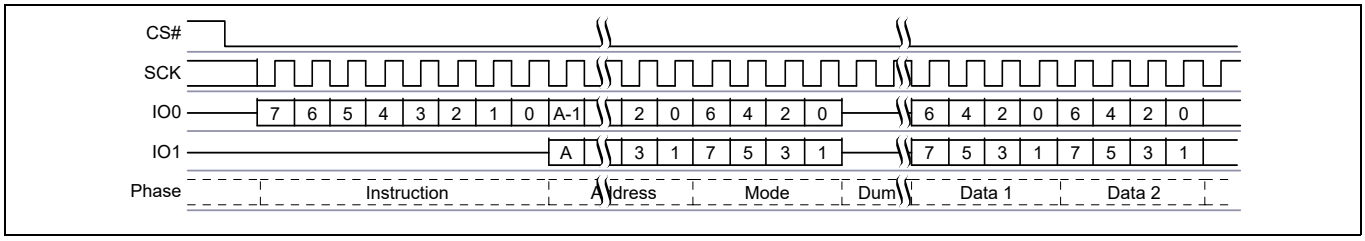



**Figure 69 Dual I/O Read command sequence**[32, 33]

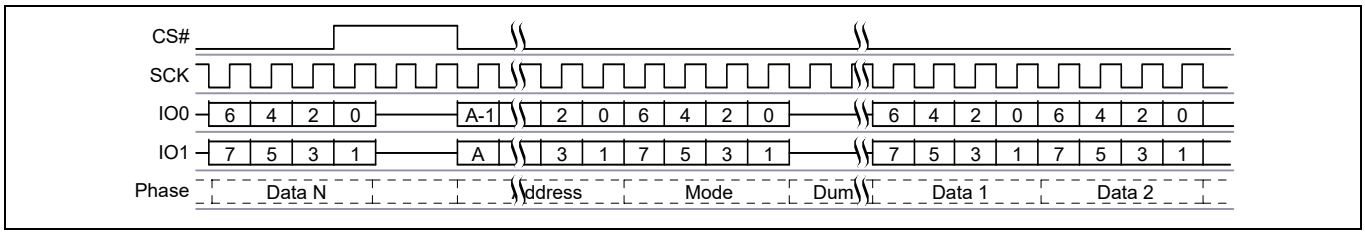



**Figure 70 Dual I/O Continuous Read command sequence**[32]

**Notes**

32.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command BCh.

33.Least significant 4 bits of mode are don't care and it is optional for the host to drive these bits. The host may turn off drive during these cycles to increase bus turn around time between mode bits from host and returning data from the memory.

### 8.4.6 Quad I/O Read (QIOR EBh or 4QIOR ECh)

The instruction,

- EBh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- EBh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- ECh is followed by a 4-byte address (A31–A0)

The Quad I/O Read command improves throughput with four I/O signals IO0–IO3. It allows input of the address bits four bits per serial SCK clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from FL-L family devices. The Quad bit of the Configuration Register 1 must be set (CR1V[1] = 1) or the QPI bit of Configuration Register 2 must be set (CR2V[1] = 1 to enable the quad capability of FL-L family devices.

For the Quad I/O Read command, there is a latency required after the mode bits (described below) before data begins shifting out of IO0–IO3. This latency period (i.e., dummy cycles) allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance. The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR3V[3:0].

Following the latency period, the memory contents at the address given, is shifted out four bits at a time through IO0–IO3. Each nibble (4 bits) is shifted out at the SCK frequency by the falling edge of the SCK signal.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

Address jumps can be done without the need for additional Quad I/O Read instructions. This is controlled through the setting of the Mode bits (after the address sequence, as shown in **Figure 71**. This added feature removes the need for the instruction sequence and greatly improves code execution (XIP). The upper nibble (bits 7–4) of the Mode bits control the length of the next Quad I/O instruction through the inclusion or exclusion of the first byte instruction code. The lower nibble (bits 3–0) of the mode bits are "don't care" ("x"). If the mode bits equal Axh, then the device remains in Quad I/O High Performance Read mode and the next address can be entered (after CS# is raised HIGH and then asserted LOW) without requiring the EBh or ECh instruction, as shown in **Figure 73**; thus, eliminating eight cycles for the command sequence. The following sequences will release the device from Quad I/O High Performance Read mode; after which, the device can accept standard SPI commands:

- During the Quad I/O Read command sequence, if the Mode bits are any value other than Axh, then the next time CS# is raised HIGH, the device will be released from quad I/O high Performance Read mode.
- Send the mode Reset command.

Note that the two Mode bit clock cycles and additional WAIT states (i.e., dummy cycles) allow the device's internal circuitry latency time to access the initial address after the last address cycle that is clocked into IO0–IO3.

It is important that the IO0–IO3 signals be set to high-impedance at or before the falling edge of the first data out clock. At higher clock speeds the time available to turn off the host outputs before the memory device begins to drive (bus turn around) is diminished. It is allowed and may be helpful in preventing IO0–IO3 signal contention, for the host system to turn off the IO0–IO3 signal outputs (make them high impedance) during the last "don't care" mode cycle or during any dummy cycles.

CS# should not be driven HIGH during mode or dummy bits as this may make the Mode bits indeterminate.

In QPI mode, (CR2V[3] = 1) the quad I/O instructions are sent 4 bits per SCK rising edge. The remainder of the command protocol is identical to the Quad I/O commands.

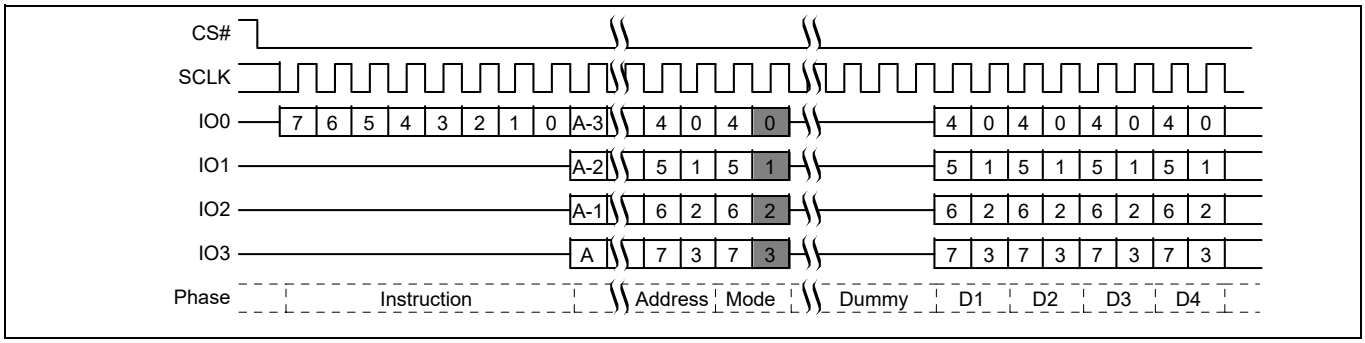



**Figure 71 Quad I/O Read Initial Access command sequence**[34]

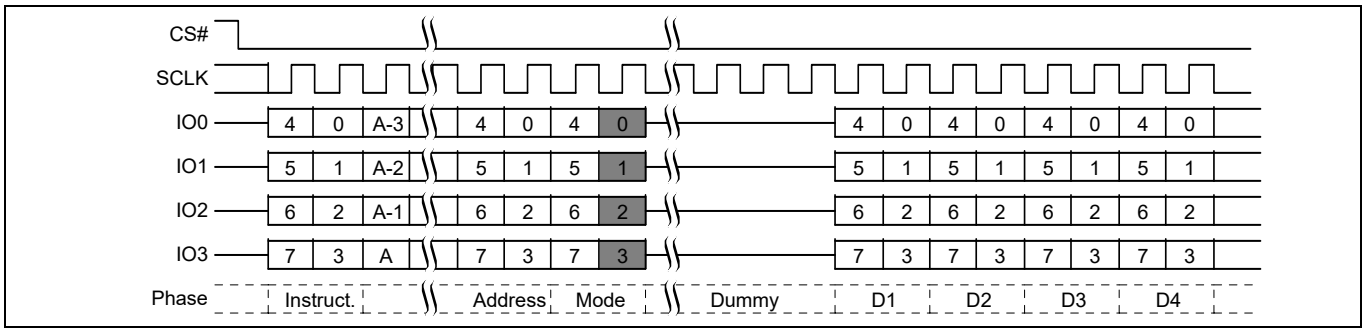



**Figure 72 Quad I/O Read Initial Access command sequence QPI mode**[34]

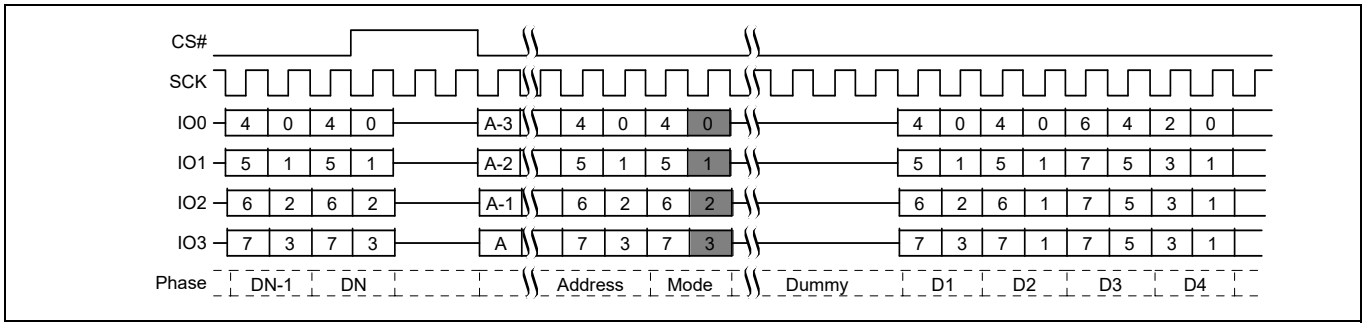



**Figure 73 Continuous Quad I/O Read command sequence**[34, 35]

**Notes**
34.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command ECh.
35.The same sequence is used in QPI mode.

### 8.4.7 DDR Quad I/O Read (EDh, EEh)

The DDR Quad I/O Read command improves throughput with four I/O signals IO0–IO3. It is similar to the Quad I/O Read command but allows input of the address four bits on every edge of the clock. In some applications, the reduced instruction overhead might allow for code execution (XIP) directly from FL-L Family devices. The Quad bit of the Configuration Register 1 must be set (CR1V[1] = 1) or the QPI bit of Configuration Register 2 must be set (CR2V[1] = 1 to enable the Quad capability of FL-L family devices.

The instruction

- EDh (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- EDh (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- EEh is followed by a 4-byte address (A31–A0)

The address is followed by Mode bits. Then the memory contents, at the address given, is shifted out, in a DDR fashion, with four bits at a time on each clock edge through IO0–IO3.

The maximum operating clock frequency for DDR quad I/O read command is 54 MHz.

For DDR Quad I/O Read, there is a latency required after the last address and Mode bits are shifted into the IO0–IO3 signals before data begins shifting out of IO0–IO3. This latency period (dummy cycles) allows the device's internal circuitry enough time to access the initial address. During these latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance. When the data learning pattern (DLP) is enabled the host system must not drive the IO signals during the dummy cycles. The IO signals must be left high impedance by the host so that the memory device can drive the DLP during the dummy cycles.

The number of dummy cycles is determined by the frequency of SCK. The latency is configured in CR3V[3:0].

Mode bits allow a series of Quad I/O DDR commands to eliminate the 8-bit instruction after the first command sends a complementary mode bit pattern. This feature removes the need for the eight bit SDR instruction sequence and dramatically reduces initial access times (improves XIP performance). The mode bits control the length of the next DDR Quad I/O Read operation through the inclusion or exclusion of the first byte instruction code. If the upper nibble (IO[7:4]) and lower nibble (IO[3:0]) of the Mode bits are complementary (i.e. 5h and Ah) the device transitions to Continuous DDR Quad I/O Read mode and the next address can be entered (after CS# is raised HIGH and then asserted low) without requiring the EDh or EEh instruction, thus eliminating eight cycles from the command sequence. The following sequences will release the device from Continuous DDR Quad I/O Read mode; after which, the device can accept standard SPI commands:

- During the DDR Quad I/O Read command sequence, if the mode bits are not complementary the next time CS# is raised HIGH and then asserted low the device will be released from DDR Quad I/O Read mode.
- Send the Mode Reset command.

The address can start at any byte location of the memory array. The address is automatically incremented to the next higher address in sequential order after each byte of data is shifted out. The entire memory can therefore be read out with one single read instruction and address 000000h provided. When the highest address is reached, the address counter will wrap around and roll back to 000000h, allowing the read sequence to be continued indefinitely.

CS# should not be driven HIGH during mode or dummy bits as this may make the mode bits indeterminate. Note that the memory devices may drive the IOs with a preamble prior to the first data value. The preamble is a data learning pattern (DLP) that is used by the host controller to optimize data capture at higher frequencies. The preamble drives the IO bus for the four clock cycles immediately before data is output. The host must be sure to stop driving the IO bus prior to the time that the memory starts outputting the preamble.

The preamble is intended to give the host controller an indication about the round trip time from when the host drives a clock edge to when the corresponding data value returns from the memory device. The host controller will skew the data capture point during the preamble period to optimize timing margins and then use the same skew time to capture the data during the rest of the read operation. The optimized capture point will be determined during the preamble period of every read operation. This optimization strategy is intended to compensate for both the PVT (process, voltage, temperature) of both the memory device and the host controller as well as any system level delays caused by flight time on the PCB.

Although the data learning pattern (DLP) is programmable, the following example shows example of the DLP of 34h. The DLP 34h (or 00110100) will be driven on each of the active outputs (i.e. all four IOs). This pattern was chosen to cover both “DC” and “AC” data transition scenarios. The two DC transition scenarios include data low for a long period of time (two half clocks) followed by a high going transition (001) and the complementary low going transition (110). The two AC transition scenarios include data low for a short period of time (one half clock) followed by a high going transition (101) and the complementary low going transition (010). The DC transitions will typically occur with a starting point closer to the supply rail than the AC transitions that may not have fully settled to their steady state (DC) levels. In many cases the DC transitions will bound the beginning of the data valid period and the AC transitions will bound the ending of the data valid period. These transitions will allow the host controller to identify the beginning and ending of the valid data eye. Once the data eye has been characterized the optimal data capture point can be chosen. See **“DDR Data Learning Registers”** on page 48 for more details.

In QPI mode, (CR2V[3] = 1) the DDR quad I/O instructions are sent 4 bits at SCK rising edge. The remainder of the command protocol is identical to the DDR Quad I/O commands.

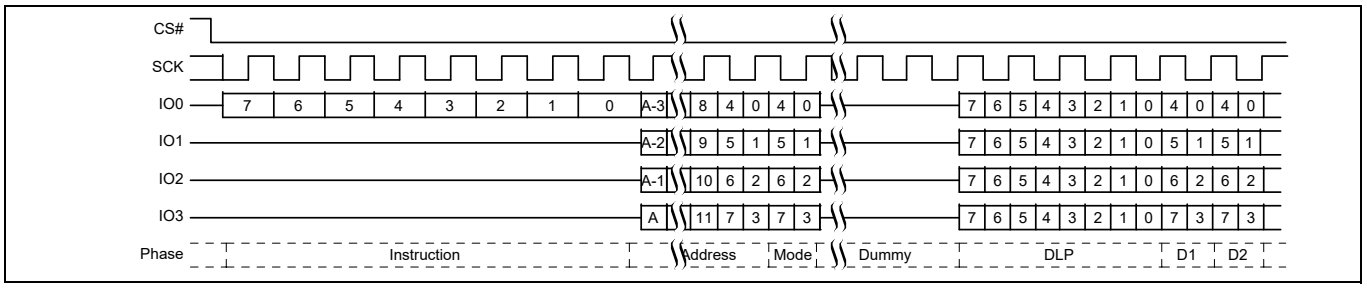



**Figure 74 DDR Quad I/O Read initial access**[36, 37]

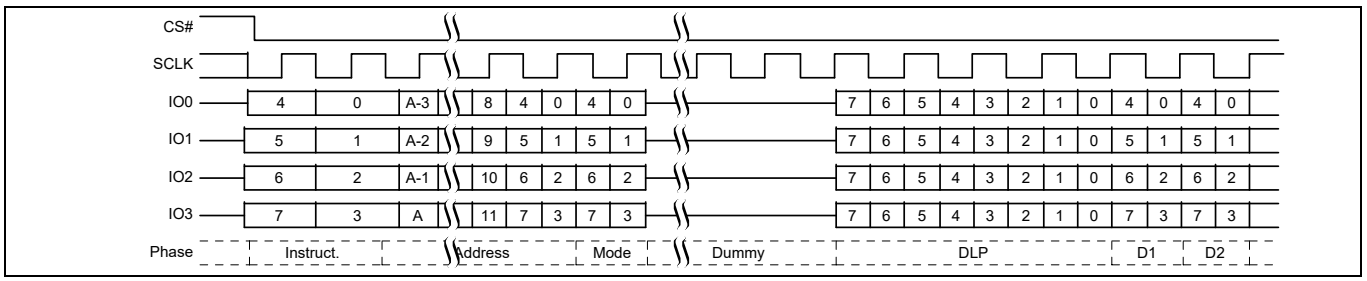



**Figure 75 DDR Quad I/O Read initial access QPI mode**[36, 37]

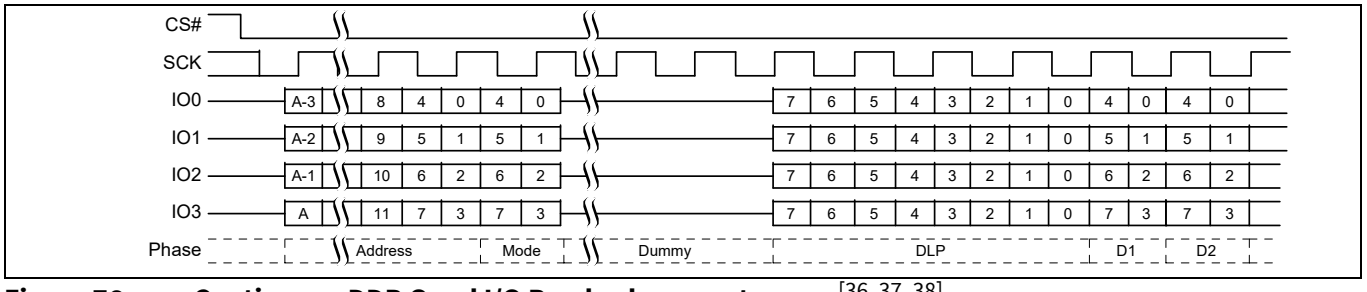



**Figure 76 Continuous DDR Quad I/O Read subsequent access**[36, 37, 38]

**Notes**
36.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1 or command EEh.
37.Example DLP of 34h (or 00110100).
38.The same sequence is used in QPI mode.

## 8.5 Program Flash Array commands

### 8.5.1 Program Granularity

#### 8.5.1.1 Page Programming

Page Programming is done by loading a page buffer with data to be programmed and issuing a Programming command to move data from the buffer to the memory array. This sets an upper limit on the amount of data that can be programmed with a single Programming command. Page Programming allows up to a page size 256 bytes to be programmed in one operation. The page is aligned on the page size address boundary. It is possible to program from one bit up to a page size in each page programming operation. 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.

#### 8.5.1.2 Single byte programming

Single byte Programming allows full backward compatibility to the legacy standard SPI page programming (PP) command by allowing a single byte to be programmed anywhere in the memory array.

### 8.5.2 Page Program (PP 02h or 4PP 12H)

The Page Program (PP) command allows bytes to be programmed in the memory (changing bits from 1 to 0). Before the Page Program (PP) commands can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded successfully, the device sets the write enable latch (WEL) in the Status Register to enable any write operations.

The instruction

- 02h (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 02h (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 12h is followed by a 4-byte address (A31–A0)

and at least one data byte on SI/IO0. Up to a page can be provided on SI/IO0 after the 3-byte address with instruction 02h or 4-byte address with instruction 12h has been provided. As with the Write and Erase commands, the CS# pin must be driven HIGH after the eighth bit of the last byte has been latched. If this is not done the Page Program command will not be executed. After CS# is driven HIGH, the self-timed Page Program command will commence for a time duration of $t_{PP}$.

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.

The programming process is managed by the Flash memory device internal control logic. After a Programming command is issued, the programming operation status can be checked using the Read Status Register 1 command. The WIP bit (SR1V[0]) will indicate when the programming operation is completed. The P_ERR bit (SR2V[5]) will indicate if an error occurs in the programming operation that prevents successful completion of programming. This includes attempted programming of a protected area.

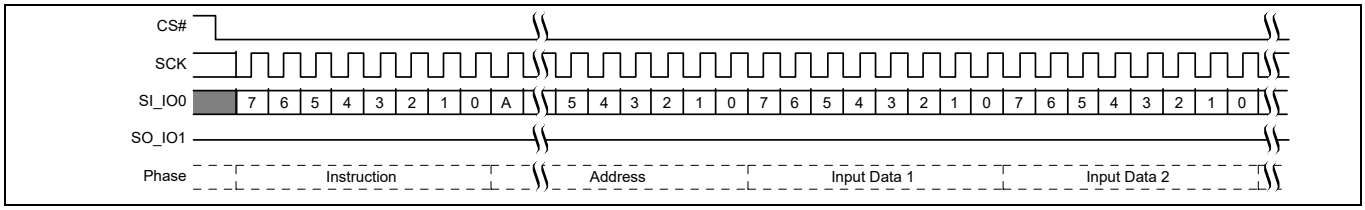



**Figure 77 Page Program (PP 02h or 4PP 12h) command sequence**[39]

**Note**
39.A = MSb of address = A23 for PP 02h with CR2V[0] = 0, or A31 for PP 02h with CR2V[0] = 1, or for 4PP 12h.

This command is also supported in QPI mode. In QPI mode, the instruction address and data is shifted in on IO0–IO3.

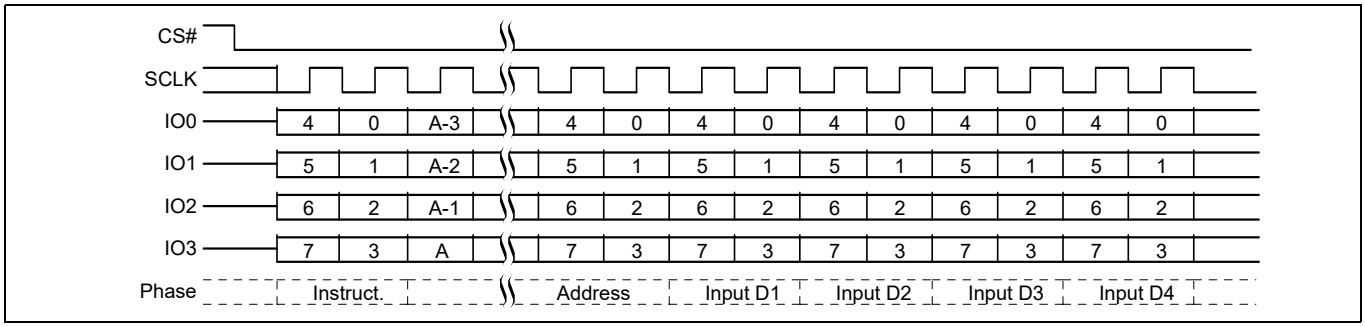



**Figure 78 Page Program (PP 02h or 4PP 12h) QPI mode command sequence**[40]

## 8.5.3 Quad Page Program (QPP 32h or 4QPP 34h)

The Quad-Input Page Program (QPP) command allows bytes to be programmed in the memory (changing bits from 1 to 0). The Quad-Input Page Program (QPP) command allows up to a page of data to be loaded into the page buffer using four signals: IO0–IO3. QPP can improve performance for PROM programmer and applications that have slower clock speeds (< 12 MHz) by loading 4 bits of data per clock cycle. Systems with faster clock speeds do not realize as much benefit for the QPP command since the inherent page program time becomes greater than the time it takes to clock-in the data. The maximum frequency for the QPP command is 108MHz.

To use Quad Page Program the Quad Enable bit in the Configuration Register must be set (QUAD = 1). A Write Enable command must be executed before the device will accept the QPP command (Status Register 1, WEL = 1).

The instruction

- 32h (CR2V[0] = 0) is followed by a 3-byte address (A23–A0) or
- 32h (CR2V[0] = 1) is followed by a 4-byte address (A31–A0) or
- 34h is followed by a 4-byte address (A31–A0)

and at least one data byte, into the IO signals. Data must be programmed at previously erased (FFh) memory locations.

All other functions of QPP are identical to Page Program. The QPP command sequence is shown in the figure below.

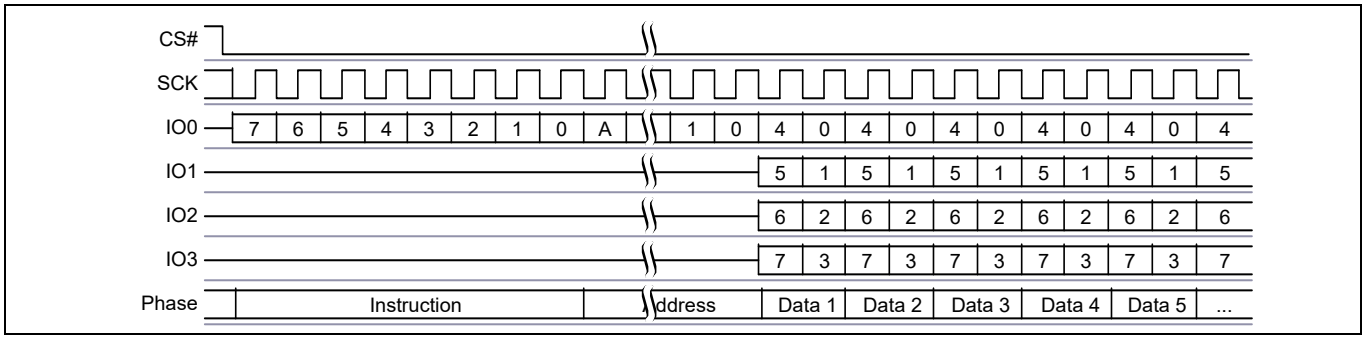



**Figure 79 Quad Page Program command sequence**[40]

**Note**
40.A = MSb of address = A23 for QPP 32h with CR2V[0] = 0, or A31 for QPP 32h with CR2V[0] = 1, or for 4QPP 34h.

## 8.6 Erase Flash Array commands

### 8.6.0.1 Sector Erase (SE 20h or 4SE 21h)

The Sector Erase (SE) command sets all bits in the addressed sector to 1 (all bytes are FFh). Before the Sector Erase (SE) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations.

The instruction

- 20h [CR2V[0] = 0] is followed by a 3-byte address (A23–A0), or
- 20h [CR2V[0] = 1] is followed by a 4-byte address (A31–A0), or
- 21h is followed by a 4-byte address (A31–A0)

CS# must be driven into the logic HIGH state after the twenty-fourth or thirty-second bit of the address has been latched in on SI/IO0. This will initiate the beginning of internal erase cycle, which involves the pre-programming and erase of the chosen sector of the flash memory array. If CS# is not driven HIGH after the last bit of address, the sector erase operation will not be executed.

As soon as CS# is driven HIGH, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write in Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a '1'. when the erase cycle is in progress and a '0' when the erase cycle has been completed.

A SE or 4SE command applied to a sector that has been write protected through the legacy block protection, Individual block lock or pointer region protection will not be executed and will set the E_ERR status.

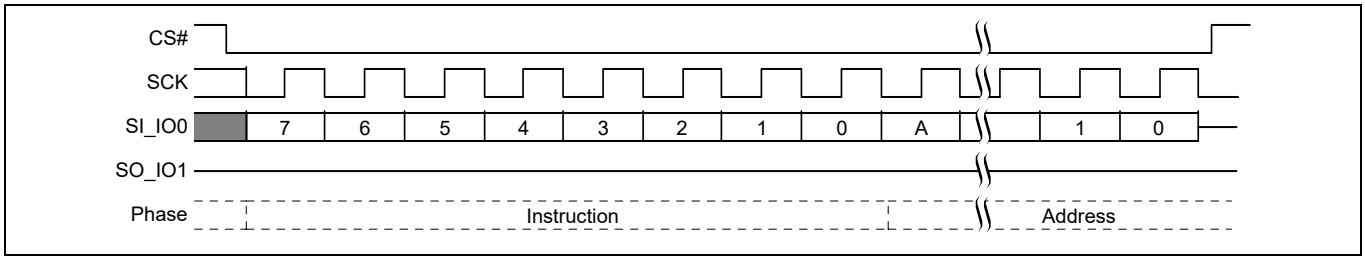



**Figure 80 Sector Erase (SE 20h or 4SE 21h) command sequence**[41]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

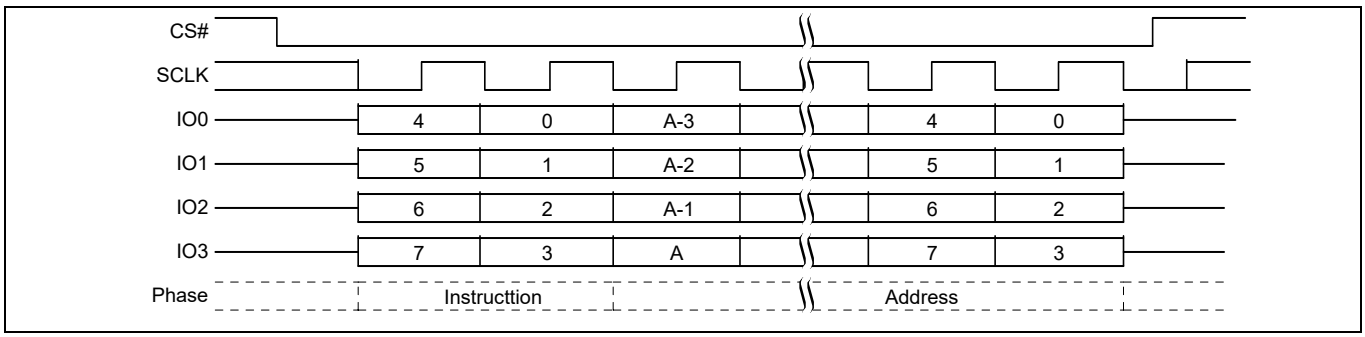



**Figure 81 Sector Erase (SE 20h or 4SE 21h) QPI mode command sequence**[41]

**Note**
41.A = MSb of address = A23 for SE 20h with CR2V[0] = 0, or A31 for SE 20h with CR2V[0] = 1 or for 4SE 21h.

### 8.6.1 Half Block Erase (HBE 52h or 4HBE 53h)

The Half Block Erase (HBE) command sets all bits in the addressed half block to 1 (all bytes are FFh). Before the Half Block Erase (HBE) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations.

The instruction

- 52h [CR2V[0] = 0] is followed by a 3-byte address (A23–A0), or
- 52h [CR2V[0] = 1] is followed by a 4-byte address (A31–A0), or
- 53h is followed by a 4-byte address (A31–A0)

CS# must be driven into the logic HIGH state after the twenty-fourth or thirty-second bit of address has been latched in on SI/IO0. This will initiate the erase cycle, which involves the pre-programming and erase of each sector of the chose block. If CS# is not driven HIGH after the last bit of address, the half block erase operation will not be executed.

As soon as CS# is driven into the logic HIGH state, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-in Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a '1' when the erase cycle is in progress and a '0' when the erase cycle has been completed.

A Half Block Erase (HBE) command applied to a block that has been write protected through the legacy block protection, individual block lock or pointer region protection will not be executed and will set the E_ERR status.

If a Half Block Erase command is applied and if any region, sector or block in the half block erase area is protected the erase will not be executed on the 32 KB range and will set the E_ERR status.

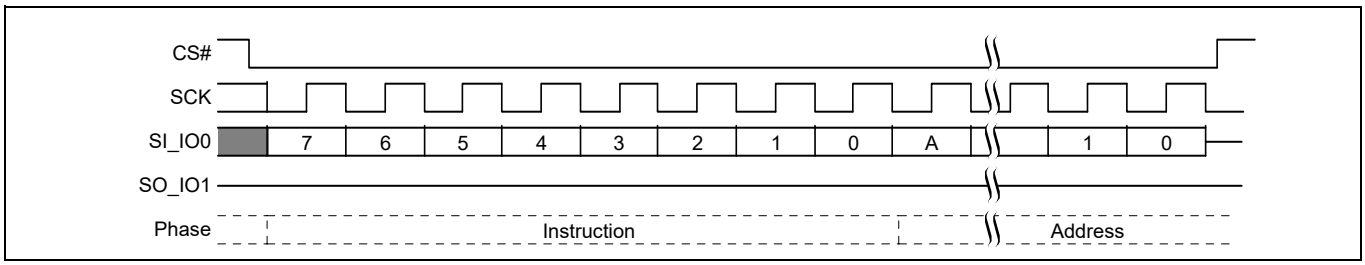



**Figure 82 Half Block Erase (HBE 52h or 4HBE 53h) command sequence**[42, 43]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

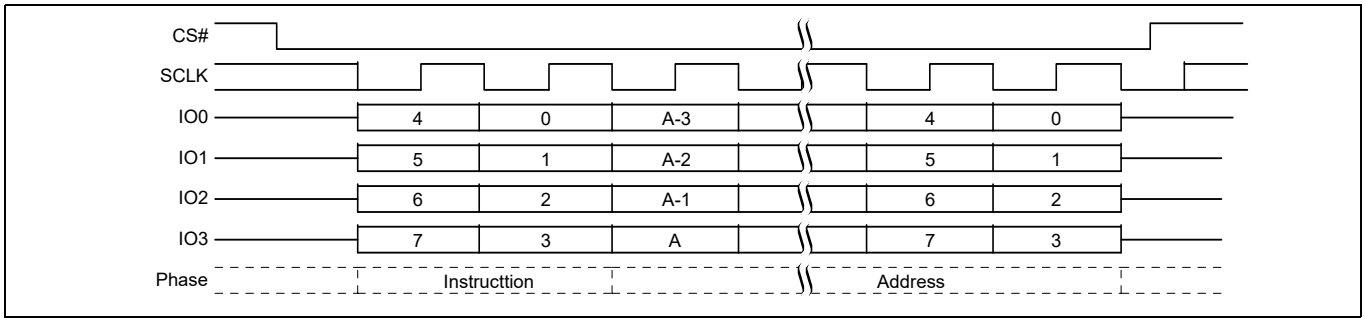



**Figure 83 Half Block Erase (HBE 52h or 4HBE 53h) QPI mode command sequence**[42, 43]

**Notes**
42.A = MSb of address = A23 for HBE 52h with CR2V[0] = 0, or A31 for HBE 52h with CR2V[0] = 1 or 4HBE 53h.
43.When A[15] = 0 the sectors 0–7 of block are erased and A[15] = 1 then sectors 8–15 of Block are erased.

### 8.6.2 Block Erase (BE D8h or 4BE DCh)

The Block Erase (BE) command sets all bits in the addressed block to 1 (all bytes are FFh). Before the Block Erase (BE) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations.

The instruction

- D8h [CR2V[0] = 0] is followed by a 3-byte address (A23–A0), or
- D8h [CR2V[0] = 1] is followed by a 4-byte address (A31–A0), or
- DCh is followed by a 4-byte address (A31–A0)

CS# must be driven into the logic HIGH state after the twenty-fourth or thirty-second bit of address has been latched in on SI/IO0. This will initiate the erase cycle, which involves the pre-programming and erase of each sector of the chosen block. If CS# is not driven HIGH after the last bit of address, the block erase operation will not be executed.

As soon as CS# is driven into the logic HIGH state, the internal erase cycle will be initiated. With the internal erase cycle in progress, the user can read the value of the Write-in Progress (WIP) bit to check if the operation has been completed. The WIP bit will indicate a '1' when the erase cycle is in progress and a '0' when the erase cycle has been completed.

A Block Erase (BE) command applied to a block that has been write protected through the legacy block protection, individual block lock or pointer region protection will not be executed and will set the E_ERR status.

If a Block Erase command is applied and if any region or sector area is protected the erase will not be executed on the 64 KB range and will set the E_ERR status.

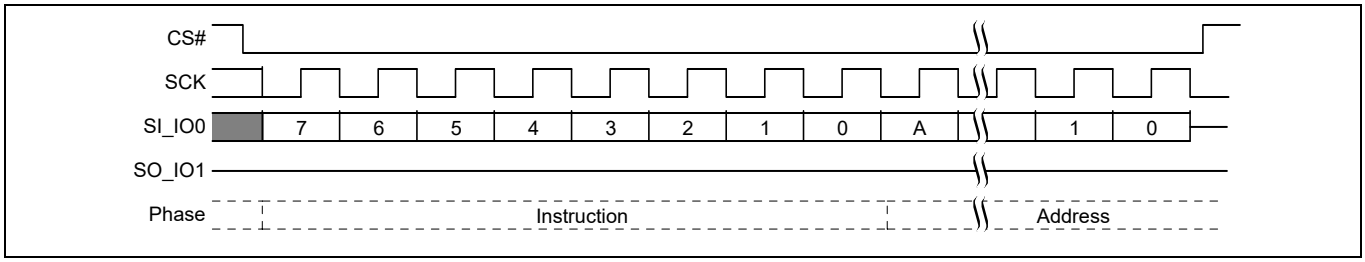



**Figure 84 Block Erase (BE D8h or 4BE DCh) command sequence**[44]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

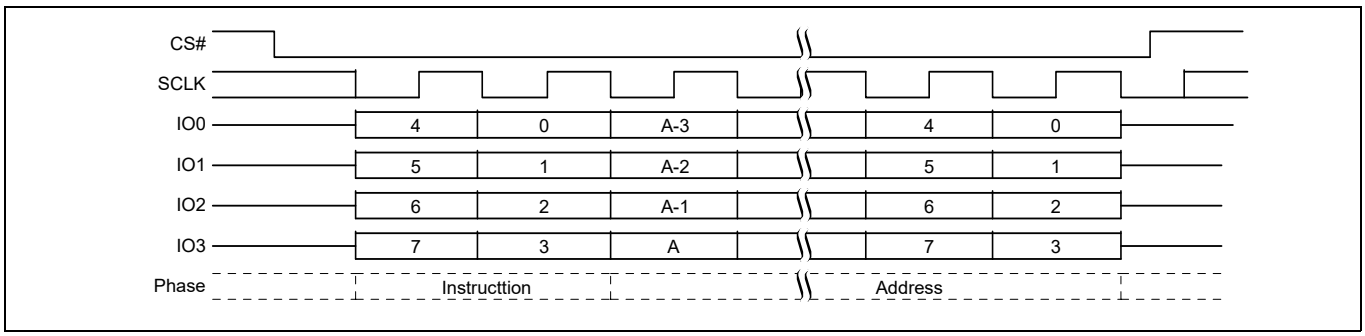



**Figure 85 Block Erase (BE D8h or 4BE DCh) QPI mode command sequence**[44]

**Note**
44.A = MSb of address = A23 for BE D8h with CR2V[0] = 0, or A31 for BE D8h with CR2V[0] = 1 or 4BE DCh.

### 8.6.3 Chip Erase (CE 60h or C7h)

The Chip Erase (CE) command sets all bits to 1 (all bytes are FFh) inside the entire flash memory array. Before the CE command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI/IO0. This will initiate the erase cycle, which involves the pre-programming and erase of the entire flash memory array. If CS# is not driven HIGH after the last bit of instruction, the CE operation will not be executed.

As soon as CS# is driven into the logic HIGH state, the erase cycle will be initiated. With the erase cycle in progress, the user can read the value of the Write-in Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a '1' when the erase cycle is in progress and a '0' when the erase cycle has been completed.

A CE command will not be executed when the legacy block protection, individual block lock or pointer region protection set to protect any sector or block and this will set the E_ERR status bit.

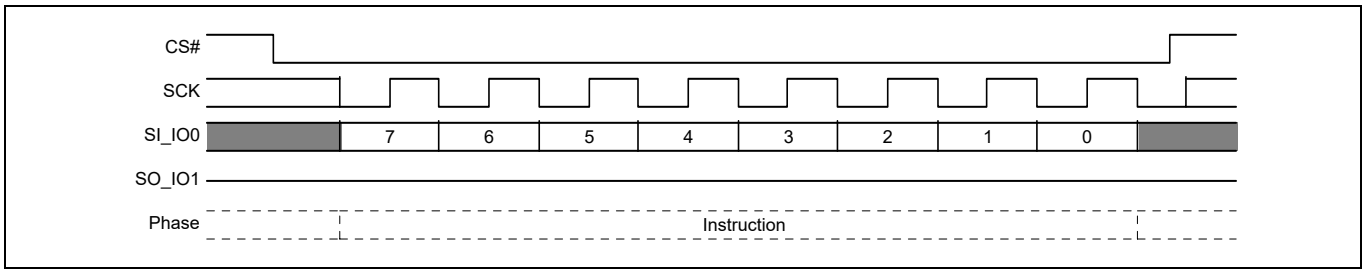



**Figure 86 Chip Erase command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

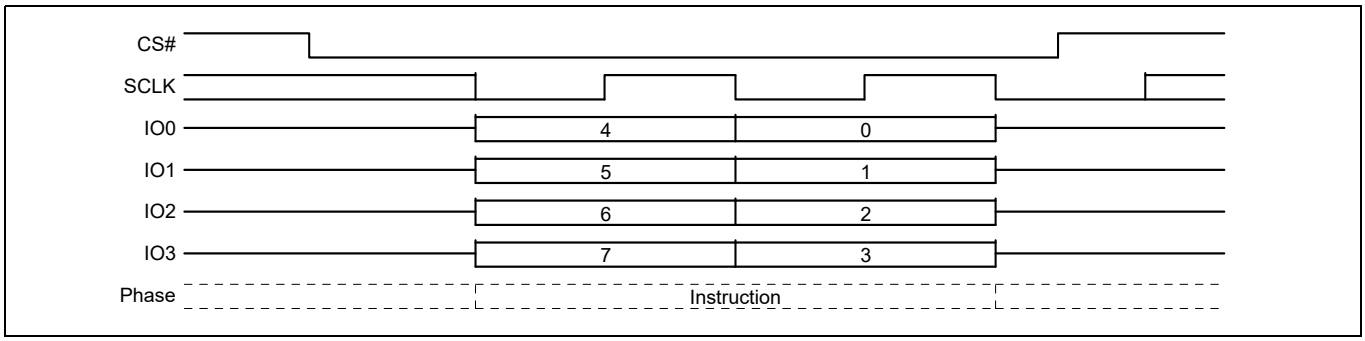



**Figure 87 Chip Erase command sequence QPI mode**

### 8.6.4 Program or Erase Suspend (PES 75h)

The PES command allows the system to interrupt a programming or erase operation and then read from any other non-erase-suspended sector or non-program-suspended-page. Program or Erase Suspend is valid only during a programming or sector erase, half block erase or block erase operation. A chip erase operation cannot be suspended.

The Write-in Progress (WIP) bit in Status Register 1 (SR1V[0]) must be checked to know when the programming or erase operation has stopped. The program suspend status bit in the Status Register 1 (SR2[0]) can be used to determine if a programming operation has been suspended or was completed at the time WIP changes to 0. The Erase Suspend Status bit in the Status Register 1 (SR2[1]) can be used to determine if an erase operation has been suspended or was completed at the time WIP changes to 0. The time required for the suspend operation to complete is $t_{SL}$, see **Table 57**.

An erase can be suspended to allow a program operation or a read operation. During an erase suspend, the IBL array may be read to examine sector protection and written to remove or restore protection on a sector to be programmed. The Protection bits will not be rechecked when the operation is resumed so any changes made will not impact current in progress operation.

A program operation may be suspended to allow a read operation.

A new suspend operation is not allowed with-in an already suspended erase or program operation. The suspend command is ignored in this situation.

**Table 36 Commands allowed during Program or Erase Suspend**

| Instruction name | Instruction code (hex) | Allowed during Erase Suspend | Allowed during Program Suspend | Comment |
|---|---|---|---|---|
| READ | 03 | X | X | All array reads allowed in suspend |
| RDSR1 | 05 | X | X | Needed to read WIP to determine end of suspend process |
| RDAR | 65 | X | X | Alternate way to read WIP to determine end of suspend process |
| RDSR2 | 07 | X | X | Needed to read suspend status to determine whether the operation is suspended or complete. |
| RDCR1 | 35 | X | X | Needed to read Configuration Register 1 |
| RDCR2 | 15 | X | X | Needed to read Configuration Register 2 |
| RDCR3 | 33 | X | X | Needed to read Configuration Register 3 |
| RUID | 4B | X | X | Needed to read Unique Id |
| RDID | 9F | X | X | Needed to read Device Id |
| RDQID | AF | X | X | Needed to read Quad Device Id |
| RSFDP | 5A | X | X | Needed to read SFDP |
| SBL | 77 | X | X | Needed to set burst length |
| WREN | 06 | X | X | Required for program command within Erase Suspend |
| WRDI | 04 | X | X | Required for program command within Erase Suspend |
| PP | 02 | X | – | Required for array program during Erase Suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |

**Note**
45. For all Quad commands the Quad Enable CR1V[1] bit (See **Table 11**) needs to be set to '1' before initial program or erase, since the WRR/WRAR commands are not allowed inside of the suspend state.

**Table 36 Commands allowed during Program or Erase Suspend** *(continued)*

| Instruction name | Instruction code (hex) | Allowed during Erase Suspend | Allowed during Program Suspend | Comment |
|---|---|---|---|---|
| QPP | 32 | X | – | Required for array program during erase suspend. Only allowed if there is no other program suspended program operation (SR2V[0] = 0). A program command will be ignored while there is a suspended program. If a program command is sent for a location within an erase suspended sector the program operation will fail with the P_ERR bit set. |
| CLSR | 30 | X | X | Clear status may be used if a program operation fails during erase suspend. |
| EPR | 7A | X | X | Required to resume from erase or program suspend |
| RSTEN | 66 | X | X | Reset allowed anytime |
| RST | 99 | X | X | Reset allowed anytime |
| FAST_READ | 0B | X | X | All array reads allowed in suspend |
| DOR | 3B | X | X | All array reads allowed in suspend |
| DIOR | BB | X | X | All array reads allowed in suspend |
| IBLRD | 3D | X | X | It may be necessary to remove and restore individual block lock during erase suspend to allow programming during erase suspend. |
| IBL | 36 | X | X | It may be necessary to restore individual block lock during erase suspend to allow programming during erase suspend. |
| IBUL | 39 | X | X | It may be necessary to remove individual block lock during erase suspend to allow programming during erase suspend. |
| QOR | 6B | X | X | Read Quad output (3-byte Address)[45] |
| QIOR | EB | X | X | All array reads allowed in suspend[45] |
| MBR | FF | X | X | May need to reset a read operation during suspend |
| SECRP | 42 | X | – | All Security Regions program allowed in erase suspend |
| SECRR | 48 | X | X | All Security Regions reads allowed in suspend |

**Note**
45. For all Quad commands the Quad Enable CR1V[1] bit (See **Table 11**) needs to be set to '1' before initial program or erase, since the WRR/WRAR commands are not allowed inside of the suspend state.

All command not included in **Table 36** are not allowed during erase or program suspend. The WRR, WRAR, or SPRP commands are not allowed during erase or program suspend, it is therefore not possible to alter the legacy block protection bits or pointer region protection during erase suspend.

Reading at any address within an erase-suspended sector or program-suspended page produces undetermined data.

After an erase-suspended program operation is complete, the device returns to the Erase-Suspend mode. The system can determine the status of the program operation by reading the WIP bit in the Status Register, just as in the standard program operation.

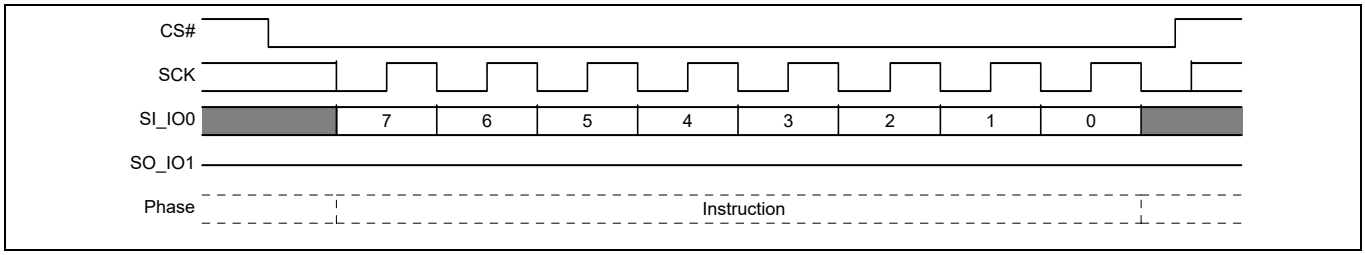



**Figure 88 Program or Erase Suspend command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

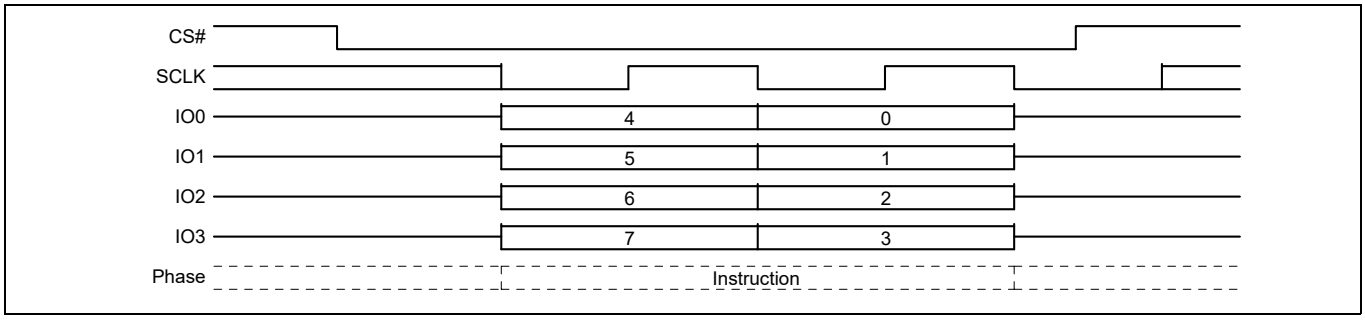



**Figure 89 Program or Erase Suspend command sequence QPI mode**

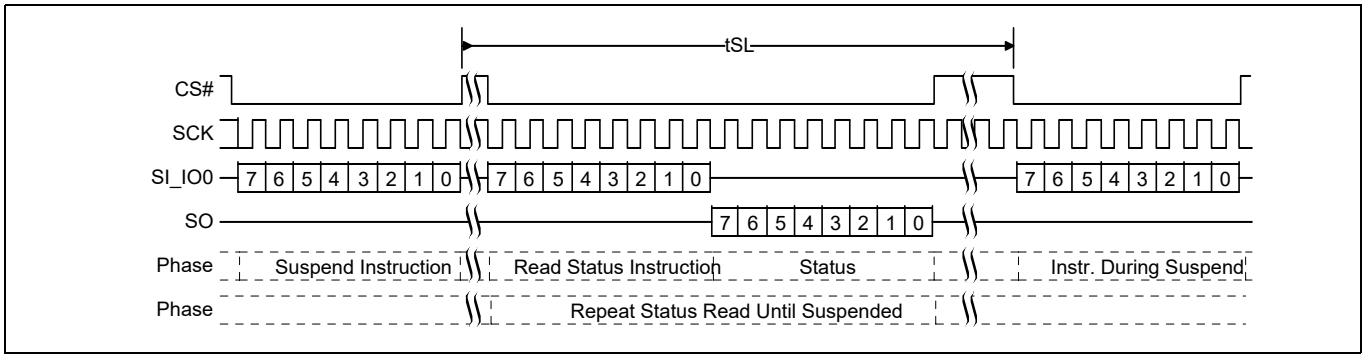



**Figure 90 Program or Erase Suspend command with continuing instruction commands sequence**

### 8.6.5 Erase or Program Resume (EPR 7Ah)

After program or read operations are completed during a Program or Erase Suspend the erase or Program Resume command is sent to continue the suspended operation.

After an Erase or Program Resume command is issued, the WIP bit in the Status Register 1 will be set to a 1 and the suspended operation will resume if one is suspended. If there is no suspended program or erase operation the Resume command is ignored.

Program or erase operations may be interrupted as often as necessary e.g. a Program Suspend command could immediately follow a Program Resume command but, but in order for a program or erase operation to progress to completion there must be some periods of time between resume and the next suspend command greater than or equal to $t_{RNS}$. See **Table 57**.

The Program Suspend Status bit in the Status Register 1 (SR2[0]) can be used to determine if a programming operation has been suspended or was completed at the time WIP changes to 0. The Erase Suspend Status bit in the Status Register 1 (SR2[1]) can be used to determine if an erase operation has been suspended or was completed at the time WIP changes to 0. See **"Status Register 2 Volatile (SR2V)"** on page 32.

An erase or program resume command must be written to resume a suspended operation.

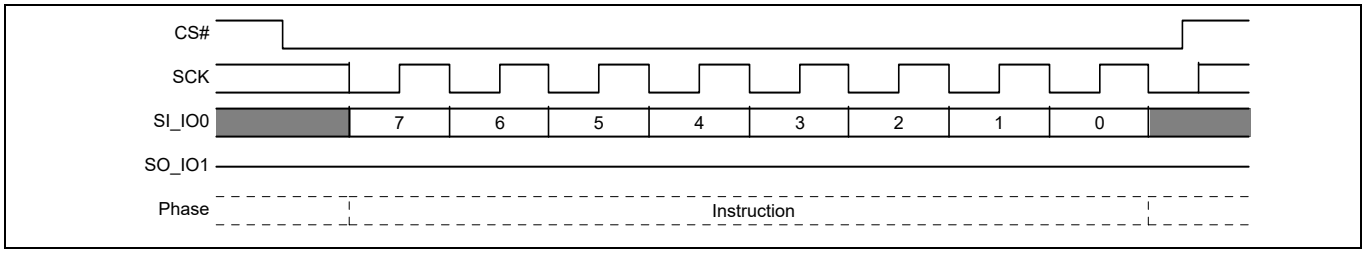



**Figure 91 Erase or Program Resume command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

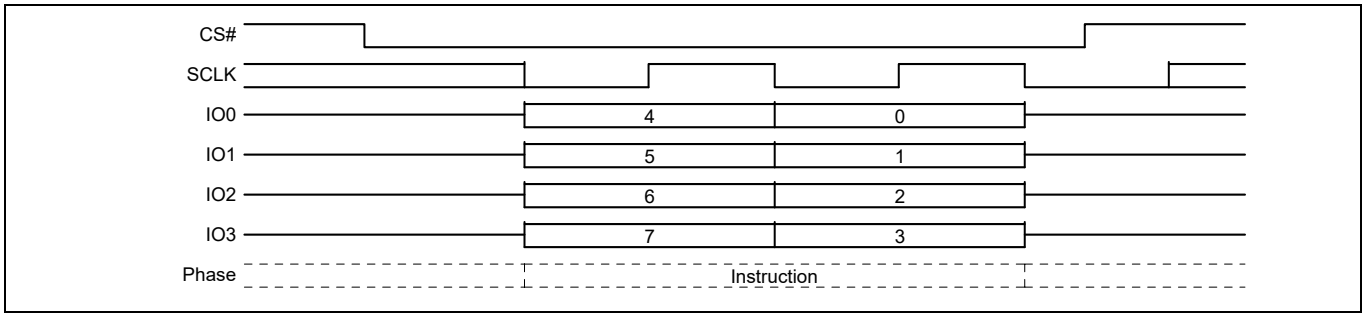



**Figure 92 Erase or Program Resume command sequence QPI mode**

## 8.7 Security Regions Array commands

The Security Regions commands select which region to use by address A15 to A8 as shown below.

- Security Region 0: A23–16 = 00h; A15–8 = 00h; A7–0 = Byte address
- Security Region 1: A23–16 = 00h; A15–8 = 01h; A7–0 = Byte address
- Security Region 2: A23–16 = 00h; A15–8 = 02h; A7–0 = Byte address
- Security Region 3: A23–16 = 00h; A15–8 = 03h; A7–0 = Byte address

### 8.7.1 Security Region Erase (SECRE 44h)

The Security Region Erase command erases data in the Security Region, which is in a different address space from the main array data. The Security Region is 1024 bytes so, the address bits for S25FL064L (A22 to A10) must be zero for this command. Each region can be individually erased. Refer to **"Security Regions address space"** on page 28 for details on the Security Region.

Before the Security Region Erase command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations. The WIP bit in SR1V may be checked to determine when the operation is completed. The E_ERR bit in SR2V may be checked to determine if an error occurred during the operation.

The Security Region Lock bits (CR1NV[2-5]) in the Configuration Register 1 can be used to protect the Security Region for erase. Once a Lock bit is set to 1, the corresponding Security Regions will be permanently locked, Attempting to erase a region that is locked will fail with the E_ERR bit in SR2V[6] set to '1'.

When the Protection Register NVLOCK bit = '0', Security Region 2 and 3 are protected from program or erase. Attempting to erase in a region that locked will fail with the E_ERR bits in SR2V[6] set to '1'. See **"NVLOCK bit (PR[0])"** on page 60.

The Password Protection Mode Lock bit (IRP[2]) allows regions 2 and 3 to be protected from erase operations until the correct password is provided to enable erasing of these Security Regions. Attempting to erase in a region that is password locked will fail with the E_ERR bit in SR2V[6] set to '1'.

The protocol of the Security Region erase command is the same as the Sector Erase command. See **"Sector Erase (SE 20h or 4SE 21h)"** on page 102 for the command sequence. QPI mode is supported.

### 8.7.2 Security Region Program (SECRP 42h)

The Security Region Program command programs data in the Security Region, which is in a different address space from the main array data. The Security Region is 1024 bytes so, the Address bits for S25FL064L (A22 to A10) must be zero for this command. Refer to **"Security Regions address space"** on page 28 for details on the Security Region.

Before the Security Region Program command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations. The WIP bit in SR1V may be checked to determine when the operation is completed. The P_ERR bit in SR2V may be checked to determine if any error occurred during the operation.

To program the Security Region array in bit granularity, the rest of the bits within a data byte can be set to '1'.

Each region in the Security Region memory space can be programmed one or more times, provided that the region is not locked. However, for the best data integrity, it is recommended that one or more 16-byte length and aligned groups of bytes be programed together and programmed only once between erase operations within each region.

The Security Region Lock bits (CR1NV[2-5]) in the Configuration Register 1 can be used to protect the Security Regions for programming. Once a Lock bit is set to 1, the corresponding Security Region will be permanently locked. Attempting to program zeros or ones in a region that is locked (protected) will fail with the P_ERR bit in SR2V[5] set to '1'. Programming ones in a un-protected area does not cause an error and does not set P_ERR (see **"Configuration Register 1"** on page 34 for detail descriptions).

When the Protection Register NVLOCK bit = '0', Security Regions 2 and 3 are protected from program or erase. Attempting to program in a region that locked will fail with the P_ERR bit in SR2V[5] set to '1'. See **"NVLOCK bit (PR[0])"** on page 60.

The Password Protection Mode Lock bit (IRP[2]) allows regions 2 and 3 to be protected from programming operations until the correct password is provided to enable programming of these Security Regions 2 and 3. Attempting to program in a region that is password locked will fail with the P_ERR bit in SR2V[5] set to '1'. See **"Password Protection mode"** on page 60.

The protocol of the Security Region program command is the same as the Page Program command. See **"Page Programming"** on page 100 for the command sequence. QPI mode is supported.

### 8.7.3 Security Regions Read (SECRR 48h)

The Security Region Read (SECRR) command provides a way to read data from the Security Regions. The Security Region is 1024 bytes so, the address bits for S25FL064L (A22 to A10) must be zero for this command. Refer to **"Security Regions address space"** on page 28 for details on the Security Regions.

The instruction is followed by a 3 or 4 byte address (depending on the address length configuration CR2V[0], followed by a number of latency (dummy) cycles set by CR3V[3:0]. Then the selected register data are returned. The protocol of the Security Region read command will not wrap to the starting address after the Security Region address is at its maximum; instead, the data beyond the maximum address will be undefined. The Security Region read command read latency is set by the latency value in CR3V[3:0].

The Security Region Read Password Mode Enable bit (IRP[6]) allows regions 3 to be protected from read operations until the correct password is provided to enable reading of this Security Region. Attempting to read in region 3 that is password locked will return invalid and undefined data. See **"Security Region read password protection"** on page 61.

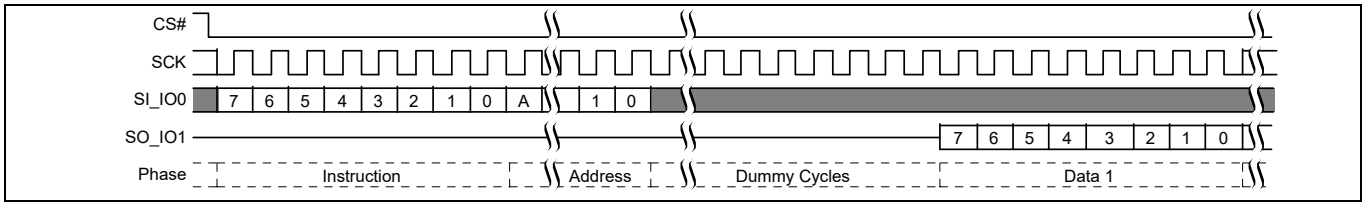



**Figure 93 Security Regions Read command sequence**[46]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in and returning data out on IO0–IO3.

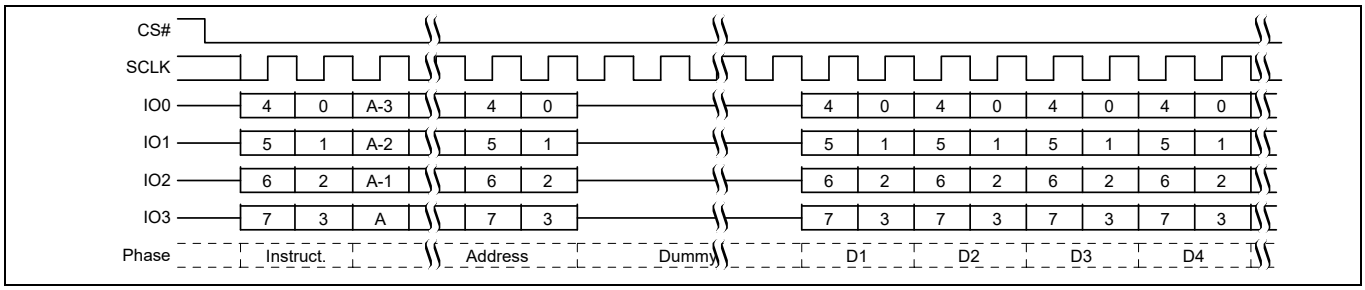



**Figure 94 Security Regions Read command sequence QPI mode**[47]

**Notes**
46.A = MSb of address = 23 for address length CR2V[0] = 0, or 31 for CR2V[0] = 1.
47.A = MSb of address = 23 for CR2V[0] = 0, or 31 for CR2V[0] = 1.

## 8.8 Individual Block Lock commands

In order to use Individual Block Lock, the IBL protection scheme must be selected by the WPS bit in Configuration Register 2 CR2V[2] = 1. If if IBL protection scheme is not selected CR2V[2] = 0 the IBL commands are ignored.

individual block Lock bits (IBL) are volatile, with one for each sector / block, and can be individually modified. By issuing the IBL or GBL commands, a IBL bit is set to '0' protecting each related sector / block. By issuing the IBUL or GUL commands, a IBL bit is cleared to '1' unprotecting each related sector or block. By issuing the IBLRD command the state of each IBL bit protection can be read.

### 8.8.1 IBL Read (IBLRD 3Dh or 4IBLRD E0h)

The IBLRD/4IBLRD command allows reading the state of each IBL bit protection.

The instruction is latched into SI by the rising edge of the SCK signal. The instruction is followed by the 24- or 32- bit address, depending on the address length configuration CR2V[0], selecting location zero within the desired sector.

Then the 8-bit IBL access register contents are shifted out on the serial output SO/IO1. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the same IBL access register continuously by providing multiples of eight clock cycles. The address of the IBL register does not increment so this is not a means to read the entire IBL array. Each location must be read with a separate IBL read command.

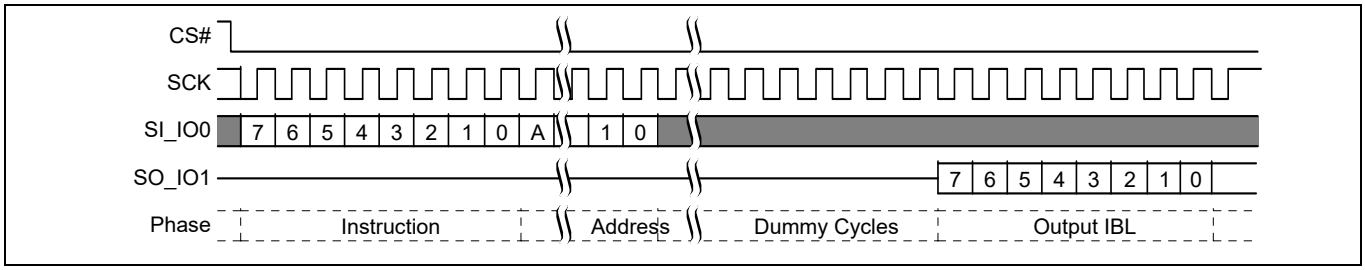



**Figure 95 IBLRD command sequence**[48, 49]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in and returning data out on IO0–IO3.

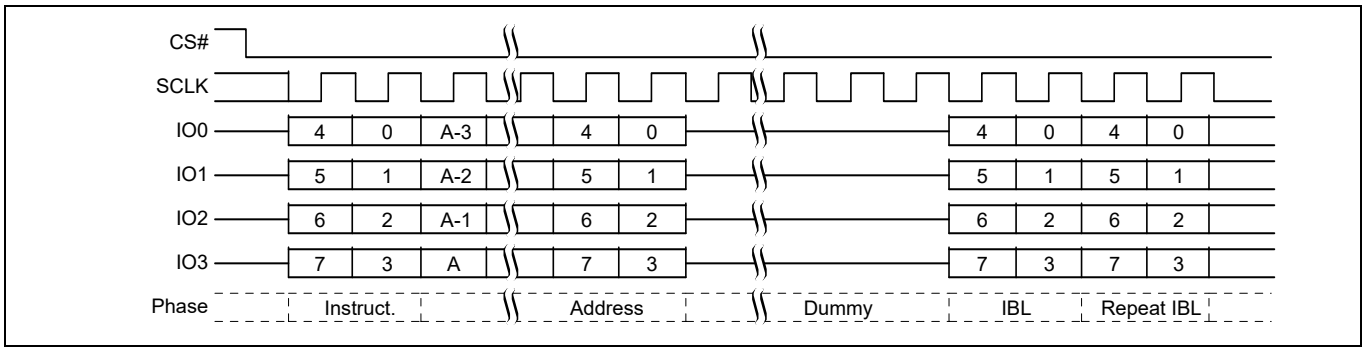



**Figure 96 IBLRD command sequence QPI**[48, 49]

**Notes**
48.A = MSb of address = 23 for Address length CR2V[0] = 0, or 31 for CR2V[0] = 1 with command 3Dh.
49.A = MSb of address = 31 with command E0h.

### 8.8.2 IBL Lock (IBL 36h or 4IBL E1h)

The IBL/4IBL commands sets the selected IBL bit to '0' protecting each related sector / block.

The IBL command is entered by driving CS# to the logic LOW state, followed by the instruction, followed by the 24- or 32-bit address, depending on the address length configuration CR2V[0]. The IBL command affects the WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# must be driven to the logic HIGH state after the 24- or 32-bit address (depending on the address length configuration CR2V[0]) has been latched in. As soon as CS# is driven to the logic HIGH state, the self-timed IBL operation is initiated. While the IBL operation 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 a '1' during the self-timed IBL operation, and is a '0' when it is completed.

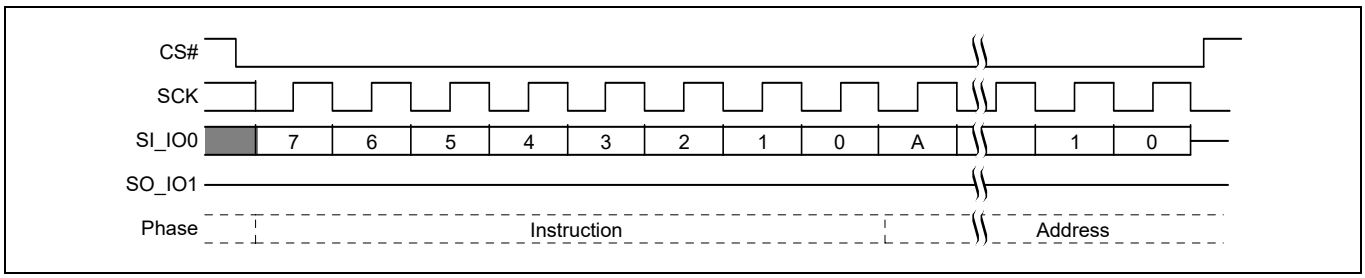



**Figure 97 IBL command sequence**[50, 51]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

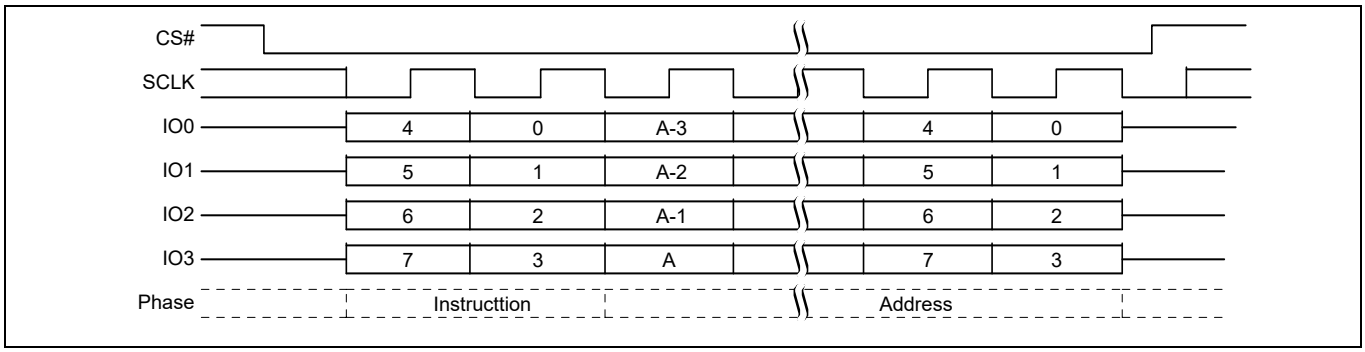



**Figure 98 IBL command sequence QPI mode**[50, 51]

**Notes**
50.A = MSb of address = 23 for Address length CR2V[0] = 0, or 31 for CR2V[0] = 1 with command 36h.
51.A = MSb of address = 31 with command E1h.

### 8.8.3 IBL Unlock (IBUL 39h or 4IBUL E2h)

The IBUL/4IBULcommands clears the selected IBL bit to ‘1’ unprotecting each related sector / block.

The IBUL command is entered by driving CS# to the logic LOW state, followed by the instruction, followed by the 24- or 32- bit address, depending on the address length configuration CR2V[0]. The IBUL command affects the WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# must be driven to the logic HIGH state after the 24- or 32-bit address (depending on the address length configuration CR2V[0]) has been latched in. As soon as CS# is driven to the logic HIGH state, the self-timed IBL operation is initiated. While the IBUL operation 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 a ‘1’ during the self-timed IBUL operation, and is a ‘0’ when it is completed.

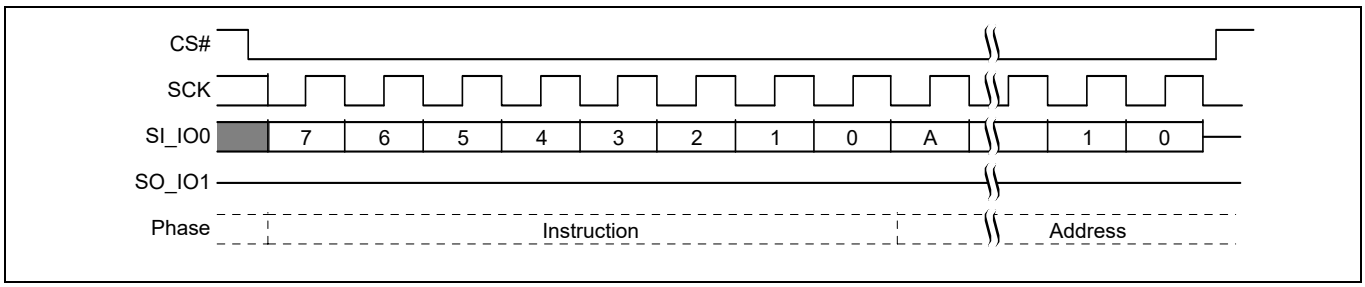



**Figure 99 IBUL command sequence**[51, 52]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

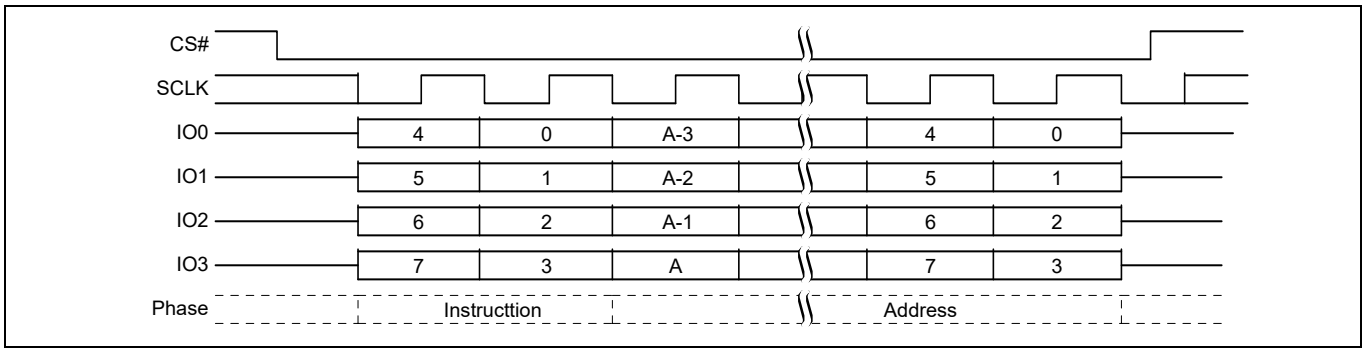



**Figure 100 IBUL command sequence QPI mode**[52, 53]

**Notes**
52.A = MSb of address = 23 for Address length (CR2V[0] = 0, or 31 for CR2V[0] = 1 with command 39h.
53.A = MSb of address = 31 with command E2h.

### 8.8.4 Global IBL Lock (GBL 7Eh)

The GBL commands sets all the IBL bits to ‘0’ protecting all sectors / blocks.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI. This will initiate the GBL. If CS# is not driven HIGH after the last bit of instruction, the GBL operation will not be executed.

As soon as CS# is driven into the logic HIGH state, the GBL will be initiated. With the GBL in progress, the user can read the value of the Write-in Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a ‘1’ when the GBL is in progress and a ‘0’ when the GBL has been completed.

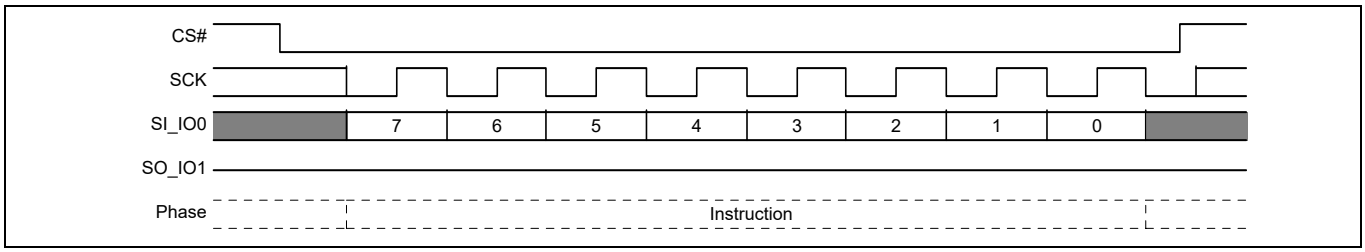



**Figure 101 Global IBL lock (GBL) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

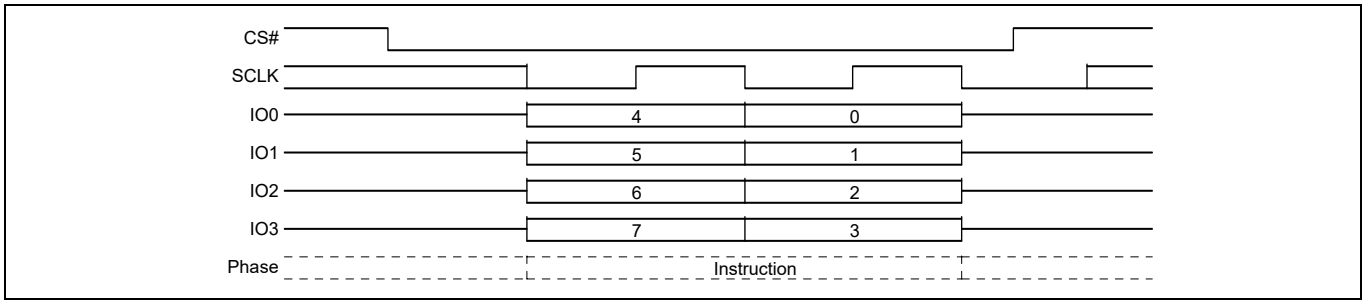



**Figure 102 Global IBL lock (GBL) command sequence QPI mode**

## 8.8.5 Global IBL Unlock (GBUL 98h)

The GBUL commands clears all the IBL bits to ‘1’ unprotecting all sectors / blocks.

CS# must be driven into the logic HIGH state after the eighth bit of the instruction byte has been latched in on SI. This will initiate the GBUL If CS# is not driven HIGH after the last bit of instruction, the GBUL operation will not be executed.

As soon as CS# is driven into the logic HIGH state, the GBL will be initiated. With the GBL in progress, the user can read the value of the Write-in Progress (WIP) bit to determine when the operation has been completed. The WIP bit will indicate a ‘1’ when the GBUL is in progress and a ‘0’ when the GBUL has been completed.

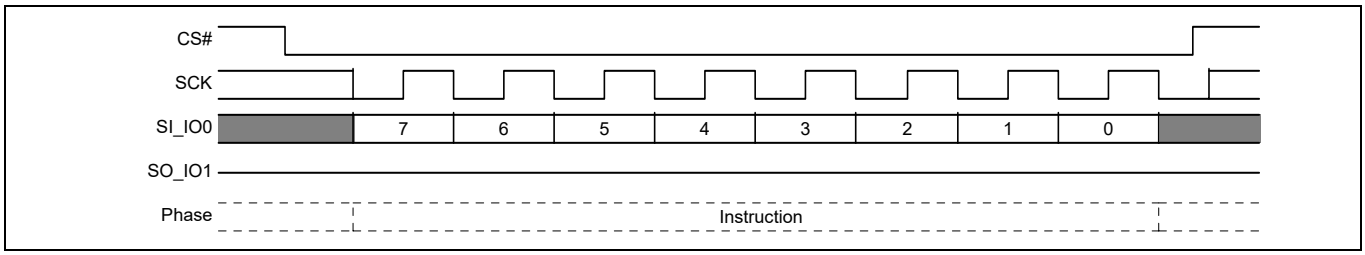



**Figure 103 Global IBL Unlock (GBUL) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

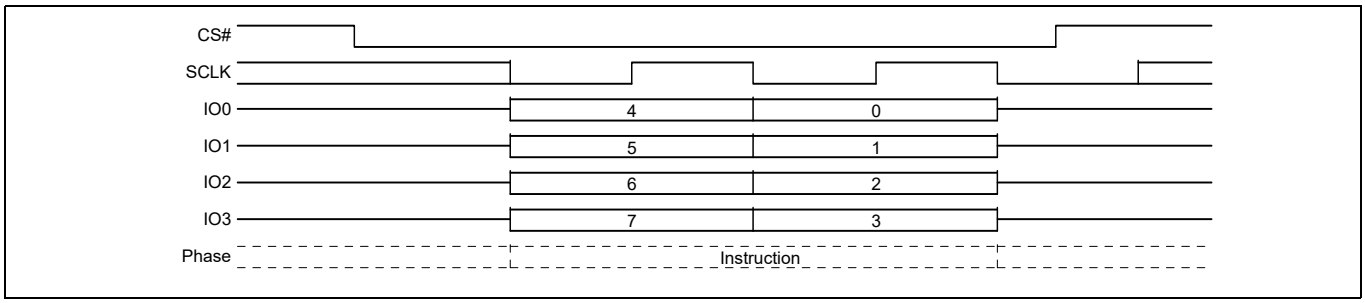



**Figure 104 Global IBL Unlock (GBUL) command sequence QPI mode**

## 8.9 Pointer Region command

### 8.9.1 Set Pointer Region Protection (SPRP FBh or 4SPRP E3h)

The SPRP or 4SPRP command is ignored during a suspend operation because the pointer value cannot be erased and re-programmed during a suspend.

The SPRP or 4SPRP command is ignored if default power supply lock-down protection NVLOCK PR[0] = 0 or power supply lock-down protection enabled IRP[1] = 0 or password protection enabled IRP[2] = 0 and NVLOCK PR[0] = 0.

Before the SPRP or 4SPRP command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the write enable latch (WEL) in the Status Register to enable any write operations.

The SPRP or 4SPRP command is entered by driving CS# to the logic LOW state, followed by the instruction, followed by the 24- or 32-bit address, depending on the address length configuration CR2V[0], see **"Pointer region protection (PRP)"** on page 56 for details on address values to select protection options.

CS# must be driven to the logic HIGH state after the last bit of address has been latched in. If not, the SPRP command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed SPRP operation is initiated. While the SPRP operation is in progress, the Status Register may be read to check the value of the Write-in Progress (WIP) bit. The WIP bit is a '1' during the self-timed SPRP operation, and is a '0' when it is completed. When the SPRP operation is completed, the write enable latch (WEL) is set to a '0'. The SPRP or 4SPRP command will set the P_ERR or E_ERR bits if there is a failure in the set pointer region protection operation.

For details on the address pointer defining a sector boundary between protected and unprotected regions in the memory, see **"Pointer region protection (PRP)"** on page 56.

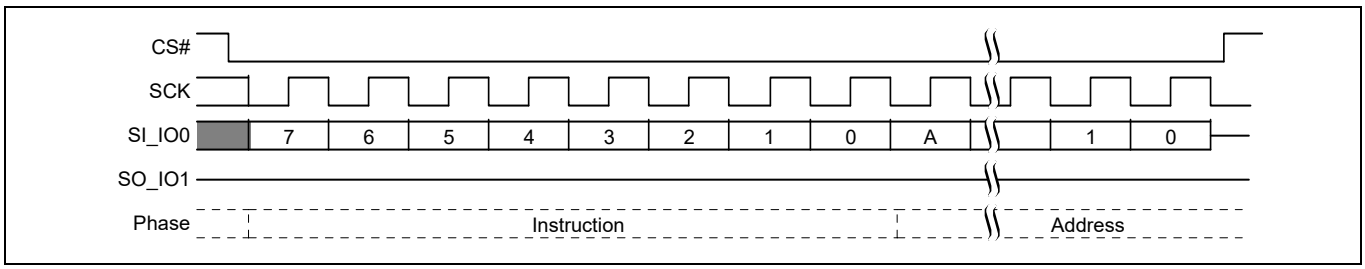



**Figure 105 SPRP command sequence**[54, 55]

This command is also supported in QPI mode. In QPI mode, the instruction and address is shifted in on IO0–IO3.

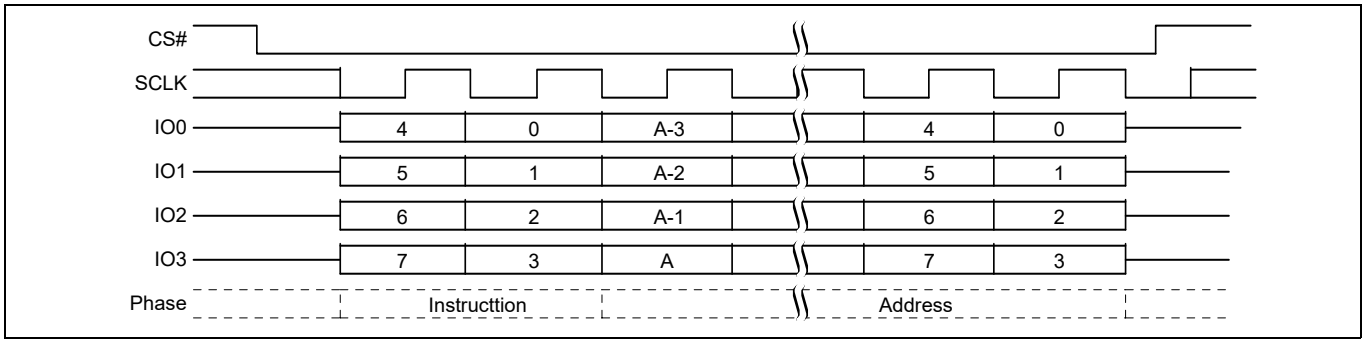



**Figure 106 SPRP command sequence QPI mode**[54, 55]

**Notes**
54.A = MSb of address = 23 for address length (CR2V[0] = 0, or 31 for CR2V[0] = 1 with command FDh.
55.A = MSb of address = 31 with command E3h.

## 8.10 Individual and Region Protection (IRP) commands

### 8.10.1 IRP Register Read (IRPRD 2Bh)

The IRP Register Read instruction 2Bh is shifted into SI/IO0 by the rising edge of the SCK signal followed by one dummy cycle. This latency period allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance.

Then the 16-bit IRP Register contents are shifted out on the serial output S0/IO1,least significant byte first. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the IRP register continuously by providing multiples of 16 clock cycles.

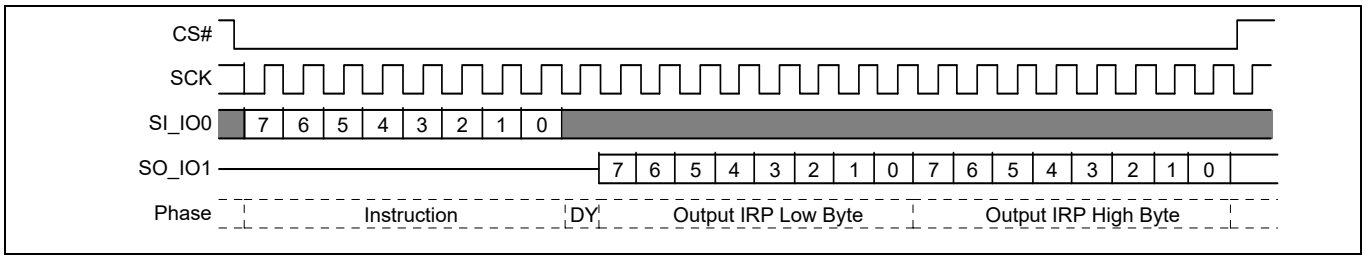



**Figure 107 IRPRD command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in and returning data out on IO0–IO3.

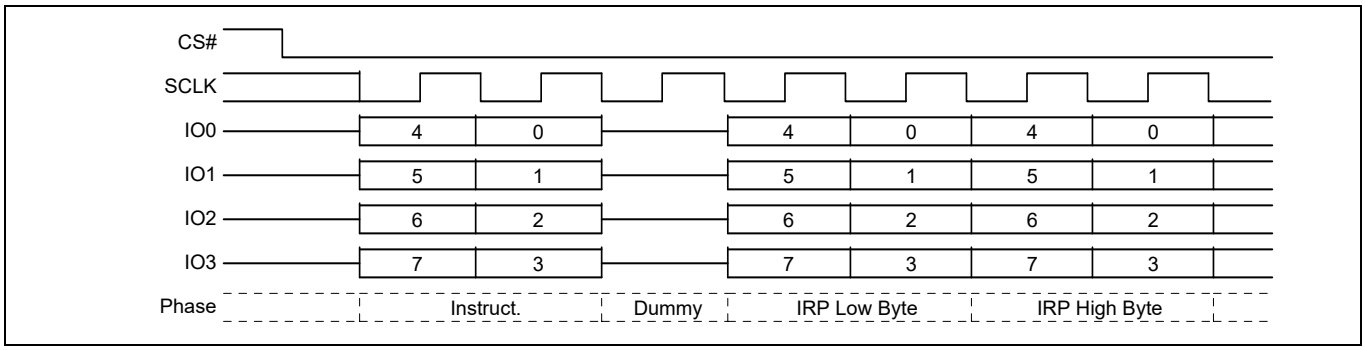



**Figure 108 IRPRD command sequence – QPI mode**

## 8.10.2 IRP Program (IRPP 2Fh)

Before the IRP program (IRPP) command can be accepted by the device, a Write Enable (WREN) command must be issued. After the Write Enable (WREN) command has been decoded, the device will set the write enable latch (WEL) in the Status Register to enable any write operations.

The IRPP command is entered by driving CS# to the logic LOW state, followed by the instruction and two data bytes on SI, least significant byte first. The IRP Register is two data bytes in length.

The IRPP command affects the P_ERR and WIP bits of the Status and Configuration Registers in the same manner as any other programming operation.

CS# input must be driven to the logic HIGH state after the sixteenth bit of data has been latched in. If not, the IRPP command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed IRPP operation is initiated. While the IRPP operation 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 a '1' during the self-timed IRPP operation, and is a '0' when it is completed. When the IRPP operation is completed, the write enable latch (WEL) is set to a '0'.

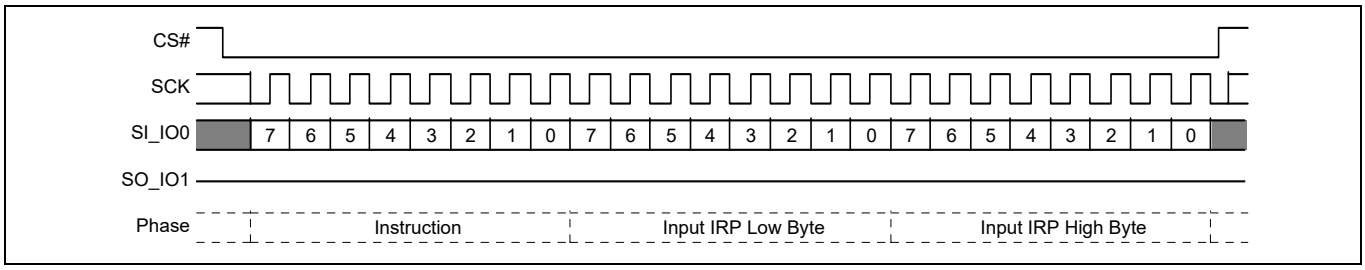



**Figure 109 IRP Program (IRPP) command**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

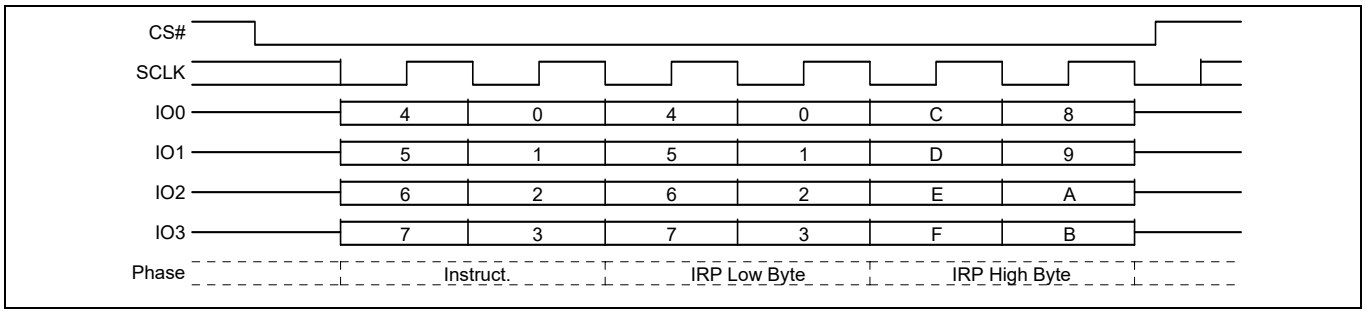



**Figure 110 IRP Program (IRPP) command QPI**

### 8.10.3 Protection Register Read (PRRD A7h)

The Protection Register Read (PRRD) command allows the Protection Register contents to be read out of SO/IO1. The read instruction A7h is shifted into SI by the rising edge of the SCK signal followed by one dummy cycle. This latency period allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on IO0–IO3 are "don't care" and may be high impedance.

Then the 8-bit Protection Register contents are shifted out on the serial output SO/IO1. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the Protection register continuously by providing multiples of eight clock cycles.

The Protection Register contents may only be read when the device is in STANDBY state with no other operation in progress.

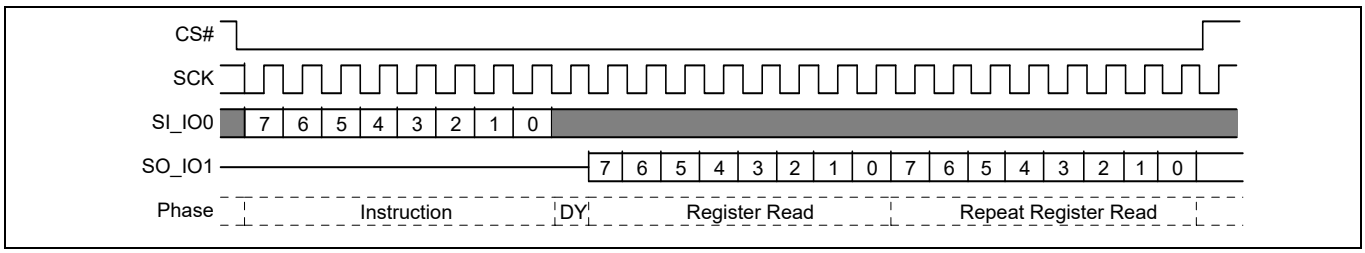



**Figure 111 Protection Register Read (PRRD) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in and returning data out on IO0–IO3.

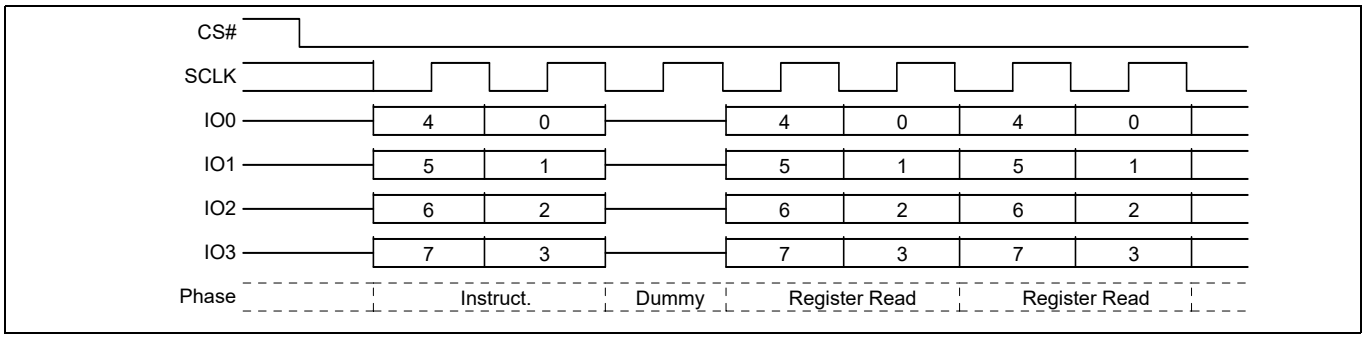



**Figure 112 Protection Register Read (PRRD) command sequence – QPI mode**

### 8.10.4 Protection Register Lock (PRL A6h)

The Protection Register Lock (PRL) command clears the NVLOCK bit (PR[0]) to zero and loads the IRP[6] value in to SECRRP (PR[6]). See **"Protection Register (PR)"** on page 46. Before the PRL command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device, which sets the write enable latch (WEL) in the Status Register to enable any write operations.

The PRL command is entered by driving CS# to the logic LOW state, followed by the instruction.

CS# must be driven to the logic HIGH state after the eighth bit of instruction has been latched in. If not, the PRL command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed PRL operation is initiated. While the PRL operation is in progress, the Status Register may still be read to check the value of the Write-in Progress (WIP) bit. The WIP bit is a '1' during the self-timed PRL operation, and is a '0' when it is completed. When the PRL operation is completed, the write enable latch (WEL) is set to a '0'.

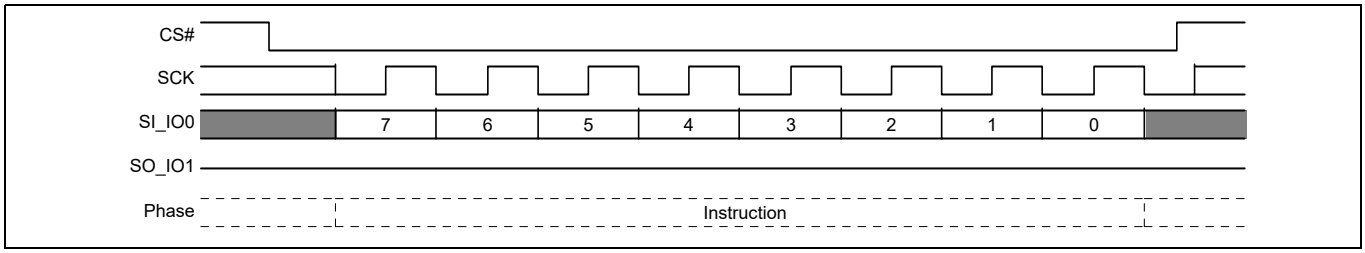



**Figure 113 Protection Register Lock (PRL) command sequence**

This command is also supported in QPI mode. In QPI mode the instruction is shifted in on IO0–IO3.

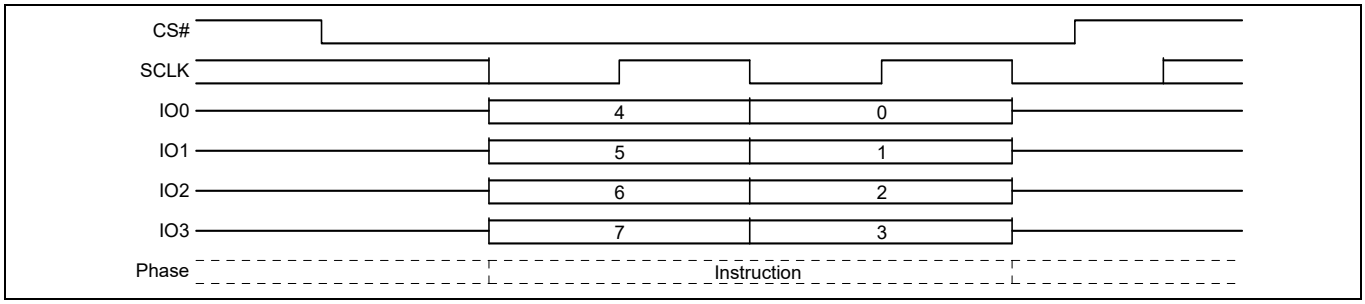



**Figure 114 Protection Register Lock (PRL) command sequence – QPI mode**

### 8.10.5 Password Read (PASSRD E7h)

The correct password value may be read only after it is programmed and before the Password mode has been selected by programming the Password Protection Mode bit to 0 in the IRP Register (IRP[2]). After the Password Protection mode is selected the password is no longer readable, the PASSRD command will output undefined data.

The PASSRD command is shifted into SI followed by one dummy cycle. This latency period allows the device's internal circuitry enough time to access data at the initial address. During latency cycles, the data value on are "don't care" and may be high impedance.

Then the 64-bit password is shifted out on the serial output, least significant byte first, most significant bit of each byte first. Each bit is shifted out at the SCK frequency by the falling edge of the SCK signal. It is possible to read the password continuously by providing multiples of 64 clock cycles.

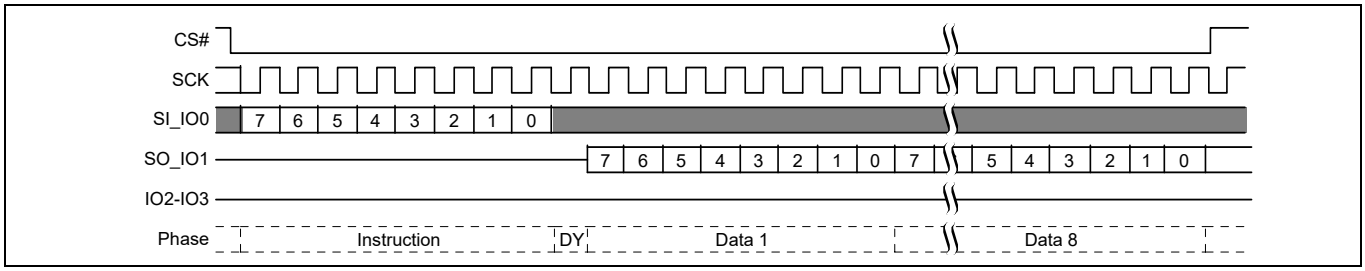



**Figure 115 Password Read (PASSRD) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in and returning data out on IO0–IO3.

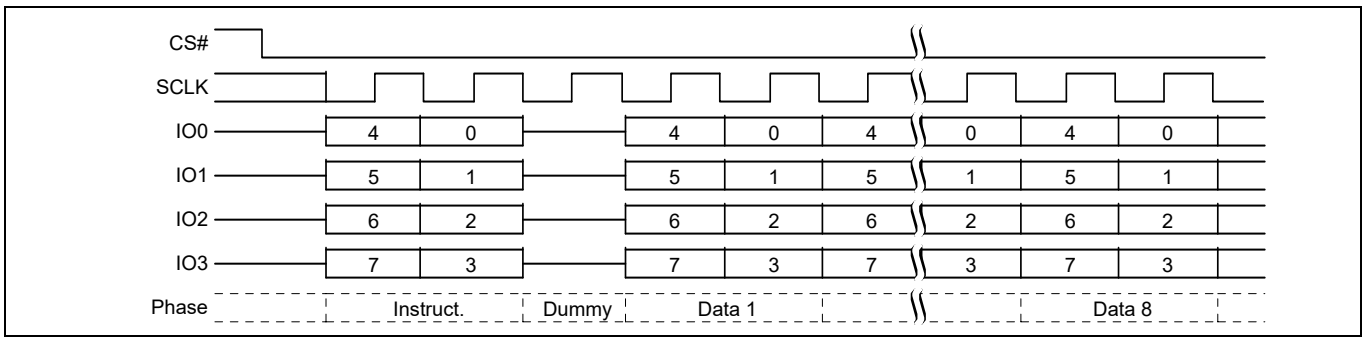



**Figure 116 Password Read (PASSRD) command sequence – QPI mode**

### 8.10.6 Password Program (PASSP E8h)

Before the Password Program (PASSP) command can be accepted by the device, a Write Enable (WREN) command must be issued and decoded by the device. After the Write Enable (WREN) command has been decoded, the device sets the write enable latch (WEL) to enable the PASSP operation.

The password can only be programmed before the Password mode is selected by programming the Password Protection Mode bit to 0 in the IRP Register (IRP[2]). After the Password Protection mode is selected the PASSP command is ignored.

The PASSP command is entered by driving CS# to the logic LOW state, followed by the instruction and the password data bytes on SI/IO0, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

CS# must be driven to the logic HIGH state after the sixty-fourth ($64^{th}$) bit of data has been latched. If not, the PASSP command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed PASSP operation is initiated. While the PASSP operation 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 a '1' during the self-timed PASSP cycle, and is a '0' when it is completed. The PASSP command can report a program error in the P_ERR bit of the status register. When the PASSP operation is completed, the write enable latch (WEL) is set to a '0'.

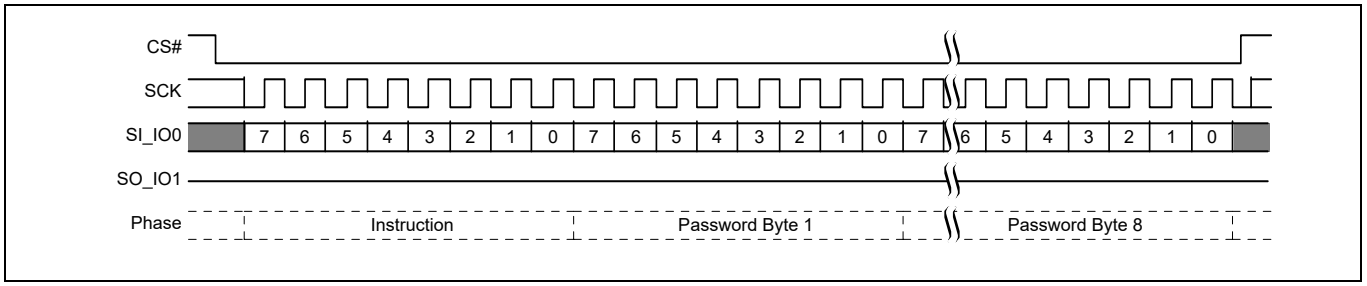



**Figure 117 Password Program (PASSP) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

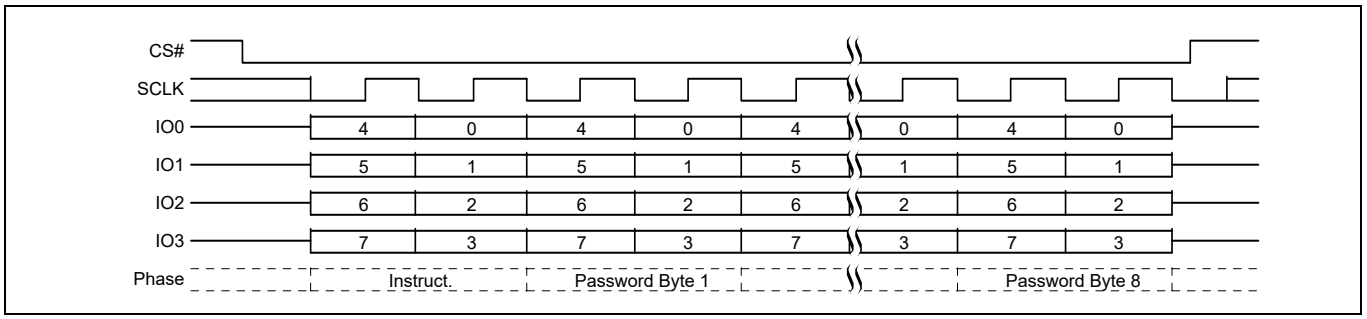



**Figure 118 Password Program (PASSP) command sequence QPI mode**

### 8.10.7 Password Unlock (PASSU EAh)

The PASSU command is entered by driving CS# to the logic LOW state, followed by the instruction and the password data bytes on SI, least significant byte first, most significant bit of each byte first. The password is sixty-four (64) bits in length.

CS# must be driven to the logic HIGH state after the sixty-fourth ($64^{th}$) bit of data has been latched. If not, the PASSU command is not executed. As soon as CS# is driven to the logic HIGH state, the self-timed PASSU operation is initiated. While the PASSU operation 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 a '1' during the self-timed PASSU cycle, and is a '0' when it is completed.

If the PASSU command supplied password does not match the hidden password in the Password Register, an error is reported by setting the P_ERR bit to 1. The WIP bit of the status register also remains set to 1. It is necessary to use the CLSR command to clear the Status Register, the Software Reset command (RSTEN 66h followed by RST 99h) to reset the device, or drive the RESET# and IO3 / RESET# input to initiate a hardware reset, in order to return the P_ERR and WIP bits to 0. This returns the device to standby STANDBY, ready for new commands such as a retry of the PASSU command.

If the password does match, the NVLOCK bit is set to '1'.

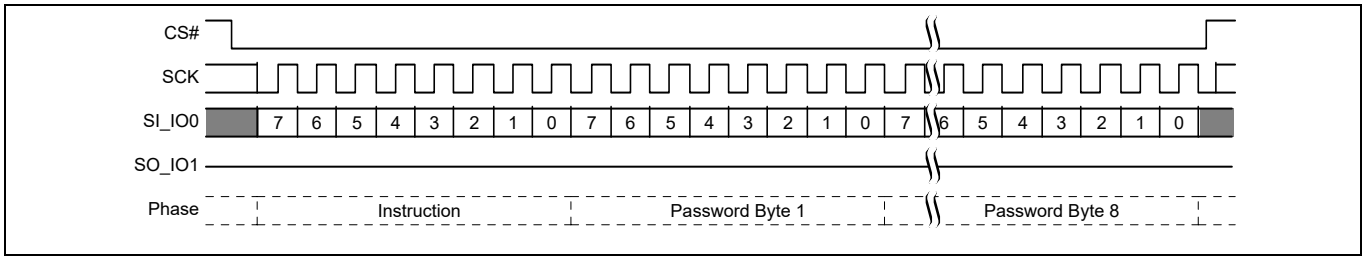



**Figure 119 Password Unlock (PASSU) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction and data is shifted in on IO0–IO3.

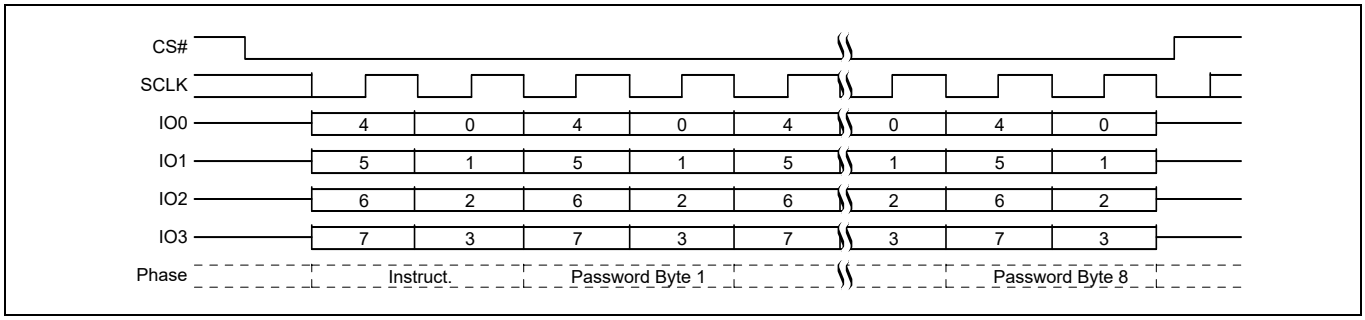



**Figure 120 Password Unlock (PASSU) command sequence QPI mode**

## 8.11 Reset commands

Software controlled Reset commands restore the device to its initial power up state, by reloading volatile registers from non-volatile default values. If a software reset is initiated during a erase, program or writing of a register operation the data in that sector, page or Register is not stable, the operation that was interrupted needs to be initiated again.

However, the volatile SRP1 bit in the Configuration Register CR1V[0] and the volatile NVLOCK bit in the Protection Register are not changed by a software reset. The software reset cannot be used to circumvent the SRP1 or NVLOCK bit protection mechanisms for the other security configuration bits.

The SRP1 bit and the NVLOCK bit will remain set at their last value prior to the software reset. To clear the SRP1 bit and set the NVLOCK bit to its Protection mode selected power on state, a full power-on-reset sequence or hardware reset must be done.

A Software Reset command (RSTEN 66h followed by RST 99h) is executed when CS# is brought HIGH at the end of the instruction and requires $t_{RPH}$ time to execute.

In the case of a previous power-up reset (POR) failure to complete, a Reset command triggers a full power up sequence requiring $t_{PU}$ to complete.

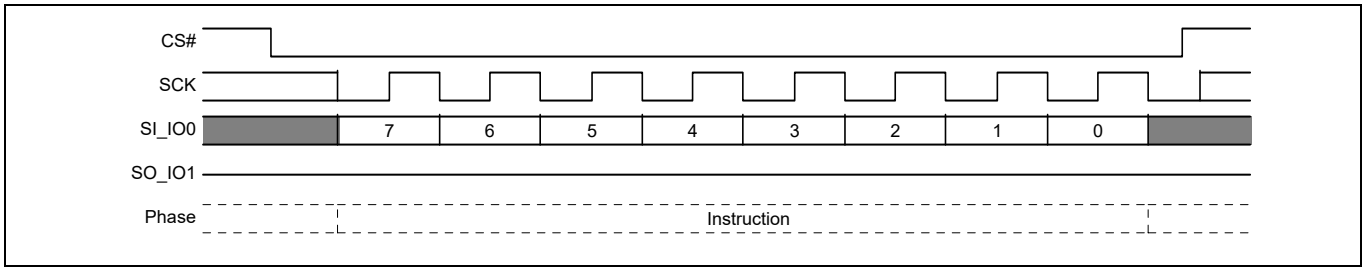



**Figure 121 Software Reset / Mode Bit Reset command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

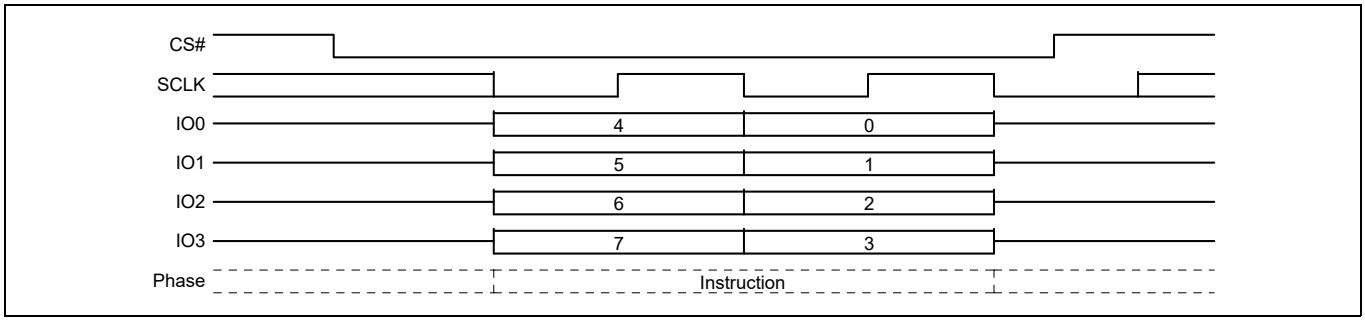



**Figure 122 Software Reset / Mode Bit command sequence – QPI mode**

### 8.11.1 Software Reset Enable (RSTEN 66h)

The Reset Enable (RSTEN) command is required immediately before a Software Reset command (RST 99h) such that a Software Reset is a sequence of the two commands. Any command other than RST following the RSTEN command, will clear the reset enable condition and prevent a later RST command from being recognized.

### 8.11.2 Software Reset (RST 99h)

The Reset (RST) command immediately following a RSTEN command, initiates the Software Reset process. Any command other than RST following the RSTEN command, will clear the reset enable condition and prevent a later RST command from being recognized.

### 8.11.3 Mode Bit Reset (MBR FFh)

The Mode Bit Reset (MBR) command is used to return the device from Continuous High Performance Read mode back to normal Standby awaiting any new command. Because the hardware RESET# input may be disabled and a device that is in a Continuous High Performance Read mode may not recognize any normal SPI command, a System Hardware Reset or Software Reset command may not be recognized by the device. It is recommended to use the MBR command after a system reset when the RESET# signal is not available or, before sending a Software Reset, to ensure the device is released from Continuous High Performance Read mode.

The MBR command sends ones on SI/IO0 for eight SCK cycles. IO1–IO3 are “don’t care” during these cycles.

### 8.11.4 Deep Power Down commands

### 8.11.5 Deep Power Down (DPD B9h)

Although the standby current during normal operation is relatively low, standby current can be further reduced with the Deep Power Down command. The lower power consumption makes the Deep Power Down (DPD) command especially useful for battery powered applications (see $I_{CC1}$ and $I_{CC2}$ in (**“DC characteristics”** on page 146). The command is initiated by driving the CS# pin LOW and shifting the instruction code “B9h”.

The CS# pin must be driven HIGH after the eighth bit has been latched. If this is not done the Deep Power Down command will not be executed. After CS# is driven HIGH, the power-down state will be entered within the time duration of $t_{DP}$ (**Table 54**). While in the power-down state only the release from Deep Power Down / Device ID command, which restores the device to normal operation, will be recognized. All other commands are ignored. This includes the Read Status Register command, which is always available during normal operation. Ignoring all but one command also makes the power down state a useful condition for securing maximum write protection.

While in the Deep Power Down mode the device will only accept a hardware reset which will initiate a power on reset that will restore the device to normal operation. The device always powers-up in the normal operation with the standby current of $I_{CC1}$.

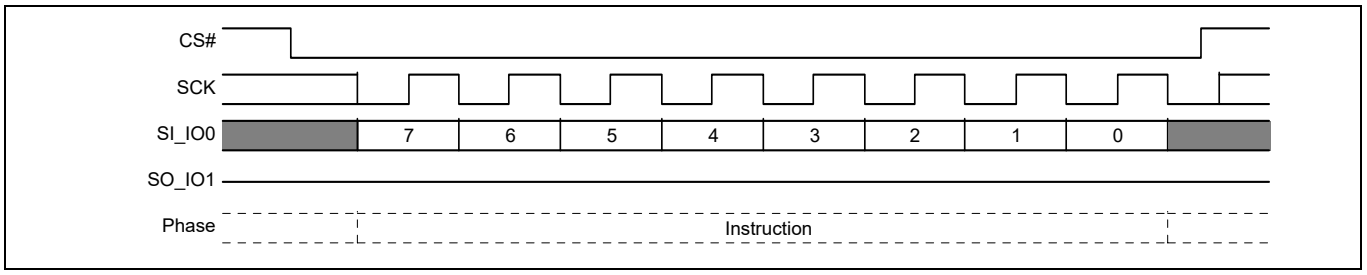



**Figure 123 Deep Power Down (DPD) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

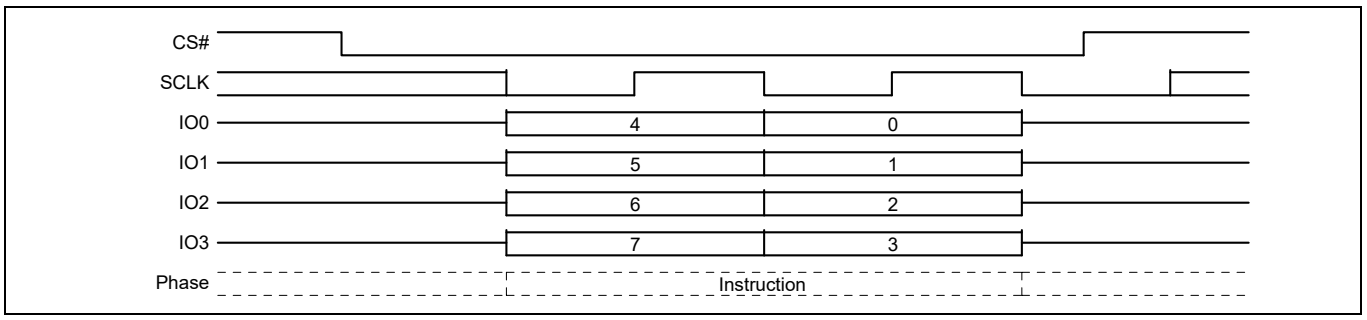



**Figure 124 Deep Power Down (DPD) command sequence – QPI mode**

### 8.11.6 Release from Deep Power Down / Device ID (RES ABh)

The release from Deep Power Down / Device ID command is a multi-purpose command. It can be used to release the device from the deep power-down state, or obtain the devices electronic identification (ID) number.

To release the device from the deep power-down state, the command is issued by driving the CS# pin LOW, shifting the instruction code “ABh” and driving CS# HIGH. Release from deep power-down will take the time duration of $t_{RES}$ (**Table 54**) before the device will resume normal operation and other commands are accepted. The CS# pin must remain HIGH during the $t_{RES}$ time duration.

When used only to obtain the Device ID while not in the deep power-down state, the command is initiated by driving the CS# pin LOW and shifting the instruction code “ABh” followed by 3-dummy bytes. The Device ID bits are then shifted out on the falling edge of CLK with most significant bit (MSb) first. The Device ID values for the S25FL-L family is listed in and **Table 43**. Continued shifting of output beyond the end of the defined ID address space will provide undefined data. The command is completed by driving CS# HIGH.

When used to release the device from the deep power-down state and obtain the device ID, the command is the same as previously described, and shown in **Figure 127** and **Figure 128**, except that after CS# is driven HIGH it must remain HIGH for a time duration of $t_{RES}$. After this time duration the device will resume normal operation and other commands will be accepted. If the release from Deep Power-down / Device ID command is issued while an erase, program or write cycle is in process (when BUSY equals 1) the command is ignored and will not have any effects on the current cycle.

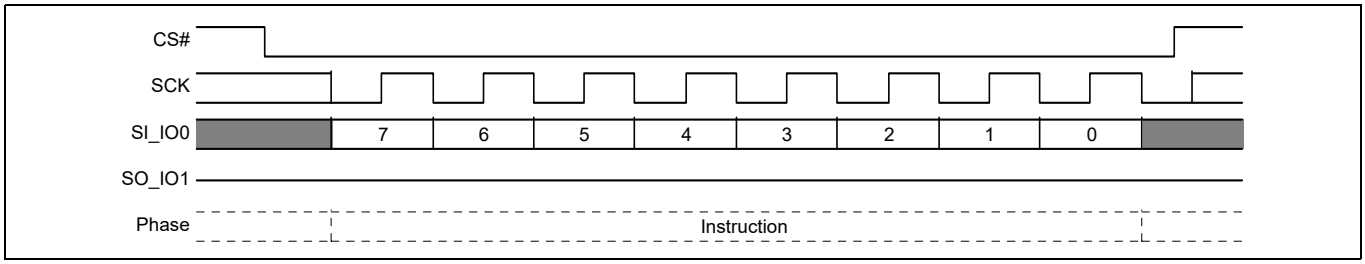



**Figure 125 Release from Deep Power Down (RES) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3.

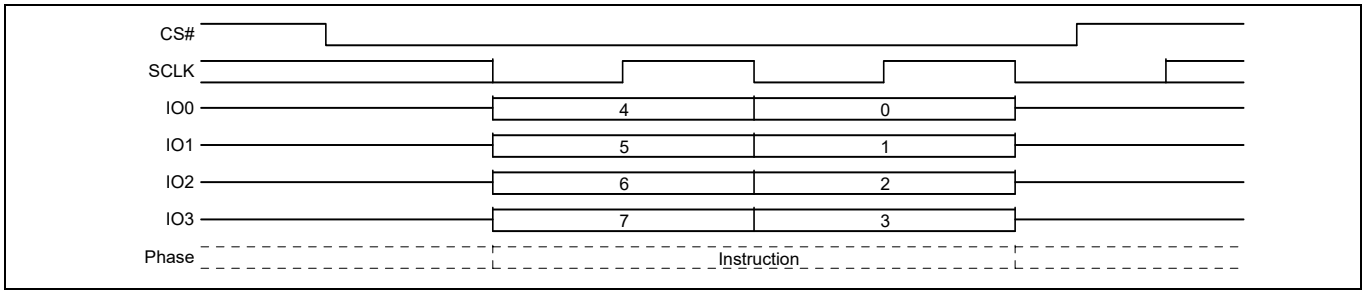



**Figure 126 Release from Deep Power Down (RES) command sequence – QPI mode**

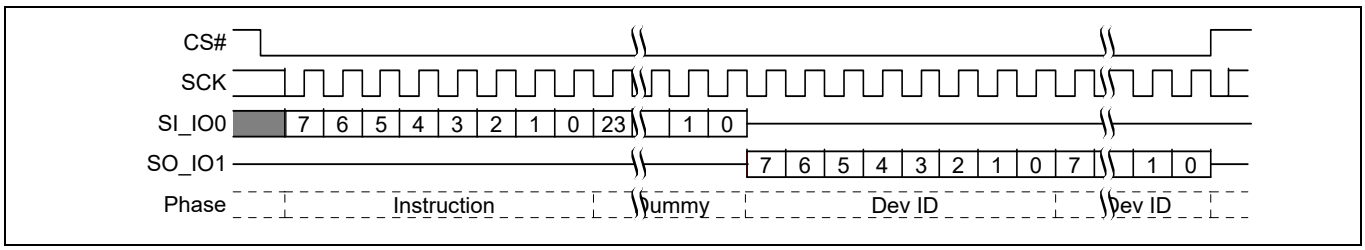



**Figure 127 Read Identification (RES) command sequence**

This command is also supported in QPI mode. In QPI mode, the instruction is shifted in on IO0–IO3 and the returning data is shifted out on IO0–IO3.

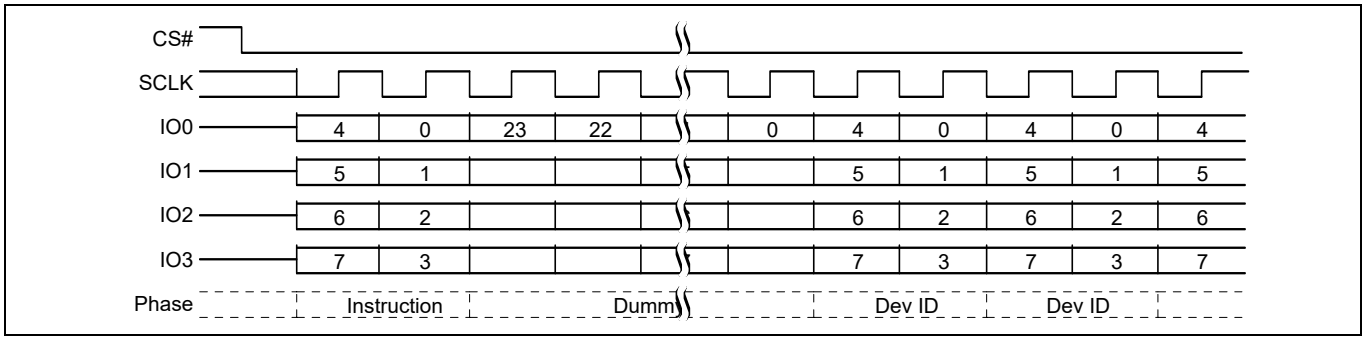



**Figure 128 Read Identification (RES) QPI mode command**

# 9 Data integrity

## 9.1 Erase endurance

**Table 37 Erase endurance**

| Parameter | Minimum | Unit |
|---|---|---|
| Program/erase cycles per main flash array sectors | 100 K | PE cycle |
| Program/erase cycles Security Region or Non-volatile Register Array[56] | 1 K | PE cycle |

**Note**
56.Each write command to a Non-volatile Register causes a PE cycle on the entire Non-volatile Register Array.

## 9.2 Data retention

**Table 38 Data retention**

| Parameter | Test conditions | Minimum time | Unit |
|---|---|---|---|
| Data retention time | 10 K program/erase cycles | 20 | Years |
| | 100 K program/erase cycles | 2 | Years |

Contact Infineon and FAE for further information on the data integrity. An application note is available at: **www.infineon.com/support**.

# 10 Software interface reference

## 10.1 JEDEC JESD216B serial flash discoverable parameters

This document defines the serial flash discoverable parameters (SFDP) revision B data structure used in the following Infineon serial flash devices:

- S25FL-L family

These data structure values are an update to the earlier revision SFDP data structure currently existing in the above devices.

The Read SFDP (RSFDP) command (5Ah) reads information from a separate flash memory address space for device identification, feature, and configuration information, in accord with the JEDEC JESD216B standard for serial flash discoverable parameters.

The SFDP data structure consists of a header table that identifies the revision of the JESD216 header format that is supported and provides a revision number and pointer for each of the SFDP parameter tables that are provided. The parameter tables follow the SFDP header. However, the parameter tables may be placed in any physical location and order within the SFDP address space. The tables are not necessarily adjacent nor in the same order as their header table entries.

The SFDP header points to the following parameter tables:

- Basic flash
  - This is the original SFDP table. It has a few modified fields and new additional field added at the end of the table.
- 4-byte address instruction
  - This is the original SFDP table. It has a few modified fields and new additional field added at the end of the table.

The physical order of the tables in the SFDP address space is: SFDP header, Basic flash sector map, 4-byte Instruction.

The SFDP address space is programmed by Infineon and read-only for the host system.

### 10.1.1 Serial flash discoverable parameters (SFDP) address map

The SFDP address space has a header starting at address zero that identifies the SFDP data structure and provides a pointer to each parameter. One basic flash parameter is mandated by the JEDEC JESD216B standard. Optional parameter tables for 4-byte address instructions follow the basic flash table.

**Table 39 SFDP overview map**

| Byte address | Description |
|---|---|
| 0000h | Location zero within JEDEC JESD216B SFDP space - start of SFDP header |
| ... | Remainder of SFDP header followed by undefined space |
| 0300h | Start of SFDP parameter |
| ... | Remainder of SFDP JEDEC parameter followed by undefined space |

### 10.1.2 SFDP header field definitions

**Table 40 SFDP header**

| SFDP byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 00h | SFDP header 1st DWORD | 53h | This is the entry point for read SFDP (5Ah) command i.e. location zero within SFDP space<br>ASCII “S” |
| 01h | | 46h | ASCII “F” |
| 02h | | 44h | ASCII “D” |
| 03h | | 50h | ASCII “P” |
| 04h | SFDP header 2nd DWORD | 06h | SFDP minor revision (06h = JEDEC JESD216 Revision B)<br>- This revision is backward compatible with all prior minor revisions. SFDP reading and parsing software will work with higher minor revision numbers than the software was designed to handle. Software designed for a higher revisions must know how to handle earlier revisions. Example: SFDP reading and parsing software for minor revision 0 will still work with minor revision 6. SFDP reading and parsing software for minor revision 6 must be designed to also read minor revision 0 or 5. Do not do a simple compare on the minor revision number, looking only for a match with the revision number that the software is designed to handle. There is no problem with using a higher number minor revision. |
| 05h | | 01h | SFDP major revision<br>This is the original major revision. This major revision is compatible with all SFDP reading and parsing software. |
| 06h | | 01h | Number of parameter headers (zero based, 01h = 2 parameters) |
| 07h | | FFh | Unused |
| 08h | Parameter header 0 1st DWORD | 00h | Parameter ID LSB (00h = JEDEC SFDP Basic SPI flash parameter) |
| 09h | | 06h | Parameter minor revision (06h = JESD216 revision B) |
| 0Ah | | 01h | Parameter major revision (01h = The original major revision - all SFDP software is compatible with this major revision. |
| 0Bh | | 10h | Parameter table length (in double words = Dwords = 4-byte units)<br>10h = 16 Dwords |
| 0Ch | Parameter header 0 2nd DWORD | 00h | Parameter table pointer byte 0 (Dword = 4-byte aligned)<br>JEDEC Basic SPI flash parameter byte offset = 0300h address |
| 0Dh | | 03h | Parameter table pointer byte 1 |
| 0Eh | | 00h | Parameter table pointer byte 2 |
| 0Fh | | FFh | Parameter ID MSB (FFh = JEDEC defined parameter) |
| 10h | Parameter header 1 1st DWORD | 84h | Parameter ID LSB (84h = SFDP 4-byte address instructions parameter) |
| 11h | | 00h | Parameter minor revision (00h = Initial version as defined in JESD216 Revision B) |
| 12h | | 01h | Parameter major revision (01h = The original major revision - all SFDP software that recognizes this parameter’s ID is compatible with this major revision. |
| 13h | | 02h | Parameter table length (in double words = Dwords = 4-byte units) (2h = 2 Dwords) |

**Table 40 SFDP header** *(continued)*

| SFDP byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 14h | Parameter header 1 2nd DWORD | 40h | Parameter table pointer byte 0 (Dword = 4-byte aligned)<br>JEDEC parameter byte offset = 0340h |
| 15h | | 03h | Parameter table pointer byte 1 |
| 16h | | 00h | Parameter table pointer byte 2 |
| 17h | | FFh | Parameter ID MSB (FFh = JEDEC defined Parameter) |

### 10.1.3 JEDEC SFDP basic SPI flash parameter

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B**

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 00h | JEDEC basic flash parameter Dword-1 | E5h | Start of SFDP JEDEC parameter<br>Bits 7:5 = unused = 111b<br>Bit 4:3 = 05h is volatile status register write instruction and status register is default non-volatile = 00b<br>Bit 2 = Program buffer > 64 bytes = 1<br>Bits 1:0 = Uniform 4 KB erase is supported through out the device = 01b |
| 01h | | 20h | Bits 15:8 = Uniform 4 KB erase instruction = 20h |
| 02h | | FBh | Bit 23 = Unused = 1b<br>Bit 22 = Supports QOR (1-1-4)read, Yes = 1b<br>Bit 21 = Supports QIO (1-4-4) read, Yes = 1b<br>Bit 20 = Supports DIO (1-2-2) read, Yes = 1b<br>Bit19 = Supports DDR, Yes = 1b<br>Bit 18:17 = Number of address bytes, 3 or 4 = 01b<br>Bit 16 = Supports fast read SIO and DIO Yes = 1b |
| 03h | | FFh | Bits 31:24 = Unused = FFh |
| 04h | JEDEC basic flash parameter Dword-2 | FFh | Density in bits, zero based,<br>64 Mb = 03FFFFFFh |
| 05h | | FFh | |
| 06h | | FFh | |
| 07h | | 03h<br>64Mb | |
| 08h | JEDEC basic flash parameter Dword-3 | 48h | Bits 7:5 = number of QIO mode cycles = 010b<br>Bits 4:0 = number of fast read QIO Dummy cycles = 01000b for default latency code |
| 09h | | EBh | Fast Read QIO instruction code |
| 0Ah | | 08h | Bits 23:21 = number of quad out mode cycles = 000b<br>Bits 20:16 = number of quad out dummy cycles = 01000b for default latency code |
| 0Bh | | 6Bh | Quad out instruction code |
| 0Ch | JEDEC basic flash parameter Dword-4 | 08h | Bits 7:5 = number of dual out mode cycles = 000b<br>Bits 4:0 = number of dual out dummy cycles = 01000b for default latency code |
| 0Dh | | 3Bh | Dual out instruction code |
| 0Eh | | 88h | Bits 23:21 = number of dual I/O mode cycles = 100b<br>Bits 20:16 = number of dual I/O dummy cycles = 01000b for default latency code |
| 0Fh | | BBh | Dual I/O instruction code |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 10h | JEDEC basic flash parameter Dword-5 | FEh | Bits 7:5 RFU = 111b<br>Bit 4 = QPI supported = 1b<br>Bits 3:1 RFU = 111b<br>Bit 0 = Dual all not supported = 0b |
| 11h | | FFh | Bits 15:8 = RFU = FFh |
| 12h | | FFh | Bits 23:16 = RFU = FFh |
| 13h | | FFh | Bits 31:24 = RFU = FFh |
| 14h | JEDEC basic flash parameter Dword-6 | FFh | Bits 7:0 = RFU = FFh |
| 15h | | FFh | Bits 15:8 = RFU = FFh |
| 16h | | FFh | Bits 23:21 = number of dual all mode cycles = 111b<br>Bits 20:16 = number of dual all dummy cycles = 11111b |
| 17h | | FFh | Dual all instruction code |
| 18h | JEDEC basic flash parameter Dword-7 | FFh | Bits 7:0 = RFU = FFh |
| 19h | | FFh | Bits 15:8 = RFU = FFh |
| 1Ah | | 48h | Bits 23:21 = number of QPI mode cycles = 010b<br>Bits 20:16 = number of QPI dummy cycles = 01000b for default latency code |
| 1Bh | | EBh | QPI fast read instruction code (Same as QIO when QPI is enabled) |
| 1Ch | JEDEC basic flash parameter Dword-8 | 0Ch | Sector type 1 size 2^N bytes = 4 KB = 0Ch (for uniform 4KB) |
| 1Dh | | 20h | Sector type 1 instruction |
| 1Eh | | 0Fh | Sector type 2 size 2^N bytes = 32 KB = 0Fh (for uniform 32KB) |
| 1Fh | | 52h | Sector type 2 instruction |
| 20h | JEDEC basic flash parameter Dword-9 | 10h | Sector type 3 size 2^N bytes = 64 KB = 10h (for uniform 64KB) |
| 21h | | D8h | Sector type 3 instruction |
| 22h | | 00h | Sector type 4 size 2^N bytes = not supported = 00h |
| 23h | | FFh | Sector type 4 instruction = not supported = FFh |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 24h | JEDEC basic flash parameter Dword-10 | 31h | Bits 31:30 = Sector type 4 erase, typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = RFU = 11b<br>Bits 29:25 = Sector type 4 erase, typical time count = RFU = 1_1111b (typ erase time = count + 1 * units = RFU = 11111)<br>Bits 24:23 = Sector type 3 erase, typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16 ms = 10b<br>Bits 22:18 = Sector type 3 erase, typical time count = 0_0011b (typ erase time = count + 1 * units = 4 * 128 ms = 512 ms)<br>Bits 17:16 = Sector type 2 erase, typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16ms = 01b<br>Bits 15:11 = Sector type 2 erase, typical time count = 1_0010b (typ erase time = count + 1 * units = 19 * 16 ms = 304 ms)<br>Bits 10:9 = Sector type 1 erase, typical time units (00b: 1 ms, 01b: 16 ms, 10b: 128 ms, 11b: 1 s) = 16 ms = 01b<br>Bits 8:4 = Sector type 1 erase, typical time count = 0_0011b (typ erase time = count + 1 * units = 4 * 16 ms = 64 ms)<br>Bits 3:0 = Count = (max erase time / (2 * typical erase time)) – 1 = 0001b<br>Multiplier from typical erase time to maximum erase time = 4x multiplier<br>Max erase time = 2 * (Count + 1) * typ erase time<br><br>Binary fields: 11-11111-10-00011-01-10010-01-00011-0001<br>Nibble format: 1111_1111_0000_1101_1001_0010_0011_0001<br>Hex format: FF_0D_92_31 |
| 25h | | 92h | |
| 26h | | 0Dh | |
| 27h | | FFh | |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 28h | JEDEC basic flash parameter Dword-11 | 81h | Bits 23 = Byte program typical time, additional byte units (0b:1 µs, 1b:8 µs) = 1 µs = 0b<br>Bits 22:19 = Byte program typical time, additional byte count, (count + 1) * units, count = 1001b, (typ Program time = count + 1 * units = 10 * 1 µs = 10 µs<br>Bits 18 = Byte program typical time, first byte units (0b:1 µs, 1b:8 µs) = 1 µs = 1b<br>Bits 17:14 = Byte program typical time, first byte count, (count + 1) * units, count = 1001b, (typ program time = count + 1 * units = 10 * 8 µs = 80 µs<br>Bits 13 = Page program typical time units (0b:8 µs, 1b:64 µs) = 64 µs = 1b<br>Bits 12:8 = Page program typical time count, (count + 1) * units, count = 00110b, (typ Program time = count + 1 * units = 7 * 64 µs = 450 µs)<br>Bits 7:4 = N = 1000b, page size= 2^N = 256B page<br>Bits 3:0 = Count = 0001b = (max page program time / (2 * typ page program time)) – 1<br>Multiplier from typical page program time to maximum page program time = 4x multiplier<br>Max page program time = 2 * (count + 1) * typ page program time<br><br>Binary fields: 0-1001-1-1001-1-00110-1000-0001<br>Nibble format: 0100_1110_0110_0110_1000_0001<br>Hex format: 4E_66_81 |
| 29h | | 66h | |
| 2Ah | | 4Eh | |
| 2Bh | | CDh 64Mb | 64 Mb = 1100_1101 = CD<br>Bit 31 Reserved = 1b<br>Bits 30:29 = Chip erase, typical time units (00b: 16 ms, 01b: 256 ms, 10b: 4 s, 11b: 64 s) = 4s = 10b<br>Bits 28:24 = Chip erase, typical time count, (count + 1) * units, count = 01100b, (typ program time = count + 1 * units = 14 * 4 s = 56 s |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 2Ch | JEDEC Basic Flash Parameter Dword-12 | CCh | Bit 31 = Suspend and resume supported = 0b<br>Bits 30:29 = Suspend in-progress erase max latency units (00b: 128 ns, 01b: 1 µs, 10b: 8 µs, 11b: 64 µs) = 8 µs = 10b<br>Bits 28:24 = Suspend in-progress erase max latency count = 00100b, max erase suspend latency = count + 1 * units = 5 * 8 µs = 40 µs<br>Bits 23:20 = Erase resume to suspend interval count = 0001b, interval = count + 1 * 64 µs = 2 * 64 µs = 128 µs<br>Bits 19:18 = Suspend in-progress program max latency units (00b: 128 ns, 01b: 1 µs, 10b: 8 µs, 11b: 64 µs) = 8 µs = 10b<br>Bits 17:13 = Suspend in-progress program max latency count = 00100b, max erase suspend latency = count + 1 * units = 5 * 8 µs = 40 µs<br>Bits 12:9 = Program resume to suspend interval count = 0001b, interval = count + 1 * 64 µs = 2 * 64 µs = 128 µs<br>Bit 8 = RFU = 1b<br>Bits 7:4 = Prohibited operations during erase suspend<br>= xxx0b: May not initiate a new erase anywhere (erase nesting not permitted)<br>+ xx0xb: May not initiate a page program anywhere<br>+ x1xxb: May not initiate a read in the erase suspended sector size<br>+ 1xxxb: The erase and program restrictions in bits 5:4 are sufficient<br>= 1100b<br>Bits 3:0 = Prohibited operations during program suspend<br>= xxx0b: May not initiate a new erase anywhere (erase nesting not permitted)<br>+ xx0xb: May not initiate a new page program anywhere (program nesting not permitted)<br>+ x1xxb: May not initiate a read in the program suspended page size<br>+ 1xxxb: The erase and program restrictions in bits 1:0 are sufficient<br>= 1100b<br><br>Binary fields: 0-10-00100-0001-10-00100-0001-1-1100-1100<br>Nibble format: 0100_0100_0001_1000_1000_0011_1100_1100<br>Hex format: 44_18_83_CC |
| 2Dh | | 83h | |
| 2Eh | | 18h | |
| 2Fh | | 44h | |
| 30h | JEDEC basic flash parameter Dword-13 | 7Ah | Bits 31:24 = Erase suspend instruction = 75h<br>Bits 23:16 = Erase resume instruction = 7Ah<br>Bits 15:8 = Program suspend instruction = 75h<br>Bits 7:0 = Program resume instruction = 7Ah |
| 31h | | 75h | |
| 32h | | 7Ah | |
| 33h | | 75h | |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 34h | JEDEC basic flash parameter Dword-14 | F7h | Bit 31 = Deep power down supported = supported = 0<br>Bits 30:23 = Enter deep power down instruction = B9h = 1011_1001b<br>Bits 22:15 = Exit deep power down instruction = ABh = 1010_1011b<br>Bits 14:13 = Exit deep power down to next operation delay units = (00b: 128 ns, 01b: 1 µs, 10b: 8 µs, 11b: 64 µs) = 1 µs = 01b<br>Bits 12:8 = Exit deep power down to next operation delay count = 00010b, Exit deep power down to next operation delay = (count + 1) * units = 3 * 1 µs = 3 µs<br>Bits 7:4 = RFU = Fh<br>Bit 3:2 = Status Register polling device busy = 01b: Legacy status polling supported = Use legacy polling by reading the Status Register with 05h instruction and checking WIP bit[0] (0 = ready; 1 = busy).<br>Bits 1:0 = RFU = 11b<br><br>Binary fields: 0-10111001-10101011-01-00010-1111-01-11<br>Nibble format: 0101_1100_1101_0101_1010_0010_1111_0111<br>Hex format: 5C_D5_A2_F7 |
| 35h | | A2h | |
| 36h | | D5h | |
| 37h | | 5Ch | |
| 38h | JEDEC basic flash parameter Dword-15 | 22h | Bits 31:24 = RFU = FFh<br>Bit 23 = Hold and WP disable = not supported = 0b<br>Bits 22:20 = quad enable requirements<br>= 101b: QE is bit 1 of the status register 2. Status register 1 is read using Read Status instruction 05h. Status register 2 is read using instruction 35h. QE is set via Write Status instruction 01h with two data bytes where bit 1 of the second byte is one. It is cleared via write status with two data bytes where bit 1 of the second byte is zero.<br>Bits 19:16 0-4-4 mode entry method<br>= xxx1b: mode bits[7:0] = A5h Note: QE must be set prior to using this mode + x1xxb: mode bits[7:0] = Axh + 1xxxb: RFU = 1101b<br>Bits 15:10 0-4-4 mode exit method = xx_xxx1b: mode bits[7:0] = 00h will terminate this mode at the end of the current read operation<br>+ xx_1xxxb: Input Fh (mode bit reset) on DQ0–DQ3 for 8 clocks. This will terminate the mode prior to the next read operation.<br>+ 11_x1xx: RFU= 111101<br>Bit 9 = 0-4-4 mode supported = 1<br>Bits 8:4 = 4-4-4 mode enable sequences<br>= 0_0010b: issue instruction 38h<br>Bits 3:0 = 4-4-4 mode disable sequences<br>= 0010b: 4-4-4 issues F5h instruction<br><br>Binary fields: 11111111-0-101-1101-111101-1-00010-0010<br>Nibble format: 1111_1111_0101_1101_1111_0110_0010_0010<br>Hex format: FF_5D_F6_22 |
| 39h | | F6h | |
| 3Ah | | 5Dh | |
| 3Bh | | FFh | |

**Table 41 Basic SPI flash parameter, JEDEC SFDP Rev B** *(continued)*

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 3Ch | JEDEC basic flash parameter Dword-16 | E8h | Bits 31:24 = Enter 4-byte addressing<br>= xxxx_xxx1b: issue instruction B7 (preceding write enable not required<br>= xxxx_1xxxb: 8-bit Volatile Bank Register used to define A[30:24] bits. MSb (bit[7]) is used to enable/disable 4-byte Address mode. When MSb is set to '1', 4-byte Address mode is active and A[30:24] bits are don't care. Read with instruction 16h. Write instruction is 17h with 1 byte of data. When MSb is cleared to '0', select the active 128 Mb segment by setting the appropriate A[30:24] bits and use 3-byte addressing.<br>+ xx1x_xxxxb: Supports dedicated 4-byte address instruction set. Consult vendor data sheet for the instruction set definition or look for 4-byte address parameter table.<br>+ 1xxx_xxxxb: Reserved = 10100001b<br>Bits 23:14 = Exit 4-byte addressing<br>= xx_xxxx_xxx1b:issue instruction E9h to exit 4-byte Address mode (write enable instruction 06h is not required)<br>= xx_xxxx_1xxxb: 8-bit Volatile Bank Register used to define A[30:24] bits. MSb (bit[7]) is used to enable/disable 4-byte Address mode. When MSb is cleared to '0', 3-byte Address mode is active and A30:A24 are used to select the active 128 Mb memory segment. Read with instruction 16h. Write instruction is 17h, data length is 1 byte.<br>+ xx_xx1x_xxxxb: Hardware reset<br>+ xx_x1xx_xxxxb: Software reset (see bits 13:8 in this DWORD)<br>+ xx_1xxx_xxxxb: Power cycle<br>+ x1_xxxx_xxxxb: Reserved<br>+ 1x_xxxx_xxxxb: Reserved<br>= 1111100001b<br>Bits 13:8 = Soft reset and rescue sequence support<br>= x1_xxxxb: issue reset enable instruction 66h, then issue reset instruction 99h. The reset enable, reset sequence may be issued on 1,2, or 4 wires depending on the device operating mode = 010000b<br>Bit 7 = RFU = 1<br>Bits 6:0 = Volatile or Non-volatile Register and write enable instruction for Status Register 1 = xxx_1xxxb:<br>Non-volatile/Volatile Status register 1 powers-up to last written value in the Non-volatile Status register, use instruction 06h to enable write to Non-volatile Status register. Volatile Status register may be activated after power-up to override the Non-volatile Status register, use instruction 50h to enable write and activate the volatile status register.<br>+ x1x_xxxxb: Reserved<br>+ 1xx_xxxxb: Reserved<br>= 1101000b<br><br>Binary fields: 10100001-1111100001-010000-1-1101000<br>Nibble format: 1010_0001_1111_1000_0101_0000_1110_1000<br>Hex format: A1_F8_60_E8 |
| 3Dh | | 50h | |
| 3Eh | | F8h | |
| 3Fh | | A1h | |

### 10.1.4 JEDEC SFDP 4-byte address instruction table

**Table 42 4-byte address instruction, JEDEC SFDP Rev B**

| SFDP parameter relative byte address | SFDP Dword name | Data | Description |
|---|---|---|---|
| 40h | JEDEC 4-byte address instructions parameter Dword-1h | FBh | Supported = 1, not supported = 0<br>Bits 31:20 = RFU = FFFh<br>Bit 19 = Support for Non-volatile Individual Sector Lock Write command, Instruction = E3h = 0<br>Bit 18 = Support for Non-volatile Individual Sector Lock Read command, Instruction = E2h = 0<br>Bit 17 = Support for Volatile Individual Sector Lock Write command, Instruction = E1h = 1<br>Bit 16 = Support for Volatile Individual Sector lock Read command, Instruction = E0h = 1<br>Bit 15 = Support for (1-4-4) DTR_Read command, instruction = EEh = 1<br>Bit 14 = Support for (1-2-2) DTR_Read command, instruction = BEh = 0<br>Bit 13 = Support for (1-1-1) DTR_Read command, instruction = 0Eh = 0<br>Bit 12 = Support for Erase command – Type 4 = 0<br>Bit 11 = Support for Erase command – Type 3 = 1<br>Bit 10 = Support for Erase command – Type 2 = 1<br>Bit 9 = Support for Erase command – Type 1 = 1<br>Bit 8 = Support for (1-4-4) Page Program command, instruction = 3Eh = 0<br>Bit 7 = Support for (1-1-4) Page Program command, instruction = 34h = 1<br>Bit 6 = Support for (1-1-1) Page Program command, instruction = 12h = 1<br>Bit 5 = Support for (1-4-4) FAST_READ command, instruction = ECh = 1<br>Bit 4 = Support for (1-1-4) FAST_READ command, instruction = 6Ch = 1<br>Bit 3 = Support for (1-2-2) FAST_READ command, instruction = BCh = 1<br>Bit 2 = Support for (1-1-2) FAST_READ command, instruction = 3Ch = 0<br>Bit 1 = Support for (1-1-1) FAST_READ command, instruction = 0Ch = 1<br>Bit 0 = Support for (1-1-1) READ command, Instruction = 13h = 1<br><br>Nibble format: 1111_1111_1111_0011_1000_1110_1111_1011<br>Hex format: FF_F3_8E_FB |
| 41h | | 8Eh | |
| 42h | | F3h | |
| 43h | | FFh | |
| 44h | JEDEC4-byte address instructions parameter Dword-2h | 21h | Bits 31:24 = FFh = Instruction for erase type 4: RFU<br>Bits 23:16 = DCh = Instruction for erase type 3 block<br>Bits 15:8 = 52h = Instruction for erase type 2 half block<br>Bits 7:0 = 21h = Instruction for erase type 1 sector |
| 45h | | 52h | |
| 46h | | DCh | |
| 47h | | FFh | |

## 10.2 Device ID address map

### 10.2.1 Field definitions

**Table 43 Manufacturer device type**

| Byte address | Data | Description |
|---|---|---|
| 00h | 01h | Manufacturer ID for Infineon |
| 01h | 60h | Device ID most significant byte - Memory interface type |
| 02h | 17h (64 Mb) | Device ID least significant byte - Density and features |
| 03h | Undefined | Reserved for future use |

**Table 44 Unique device ID**

| Byte address | Data | Description |
|---|---|---|
| 00h to 07 | 8-byte unique Device ID | 64-bit unique ID number.<br>See section **"Device Unique ID"** on page 28. |

## 10.3 Initial delivery state

The device is shipped from Infineon with non-volatile bits set as follows:

- The entire memory array is erased: all bits are set to 1 (each byte contains FFh).
- The Security Region address space has all bytes erased to FFh.
- The SFDP address space contains the values as defined in the description of the SFDP address space.
- The ID address space contains the values as defined in the description of the ID address space.
- The Status Register 1 non-volatile contains 00h (all SR1NV bits are cleared to 0's).
- The Configuration Register 1 non-volatile contains 00h.
- The Configuration Register 2 non-volatile contains 60h.
- The Configuration Register 3 non-volatile contains 78h.
- The Password Register contains FFFFFFFF–FFFFFFFFh
- The IRP Register bits are FFFDh for standard part and FFFFh for high security part.
- The PRPR Register bits are FFFFFFh

# 11 Electrical specifications

## 11.1 Absolute maximum ratings[59]

Storage temperature plastic packages.....–65°C to +150°C
Ambient temperature with power applied.....–65°C to +125°C
$V_{CC}$.....–0.5 V to +4.0 V
Input voltage with respect to ground (VSS)[58].....–0.5 V to $V_{CC}$ + 0.5 V
Output short circuit current[57].....100 mA

## 11.2 Latchup characteristics

**Table 45 Latchup specification[60]**

| Description | Min | Max | Unit |
|---|---|---|---|
| Input voltage with respect to $V_{SS}$ on all input only connections | –1.0 | $V_{CC}$ + 1.0 | V |
| Input voltage with respect to $V_{SS}$ on all I/O connections | –1.0 | $V_{CC}$ + 1.0 | V |
| $V_{CC}$ current | –100 | +100 | mA |

## 11.3 Thermal resistance

**Table 46 Thermal resistance**

| Parameter | Description | Test condition | SL3016 | SOC008 | FAB024 | FAC024 | WND008 | UNF008 | Unit |
|---|---|---|---|---|---|---|---|---|---|
| Theta JA | Thermal resistance (junction to ambient) | Test conditions follow standard test methods and procedures for measuring thermal impedance in accordance with EIA/JESD51. with still air (0 m/s). | 45.7 | 65.8 | 46.9 | 46.9 | 32.9 | 34.0 | °C/W |
| Theta JB | Thermal resistance (junction to board) | | 26.6 | 39.6 | 30.4 | 30.4 | 9.1 | 8.0 | °C/W |
| Theta JC | Thermal resistance (junction to case) | | 13.1 | 33.8 | 20.9 | 20.9 | 25.2 | 28.0 | °C/W |

**Notes**
57. See **"Input signal overshoot"** on page 143 for allowed maximums during signal transition.
58. No more than one output may be shorted to ground at a time. Duration of the short circuit should not be greater than one second.
59. Stresses above those listed under **"Absolute maximum ratings[59]"** on page 142 may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this data sheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.
60. Excludes power supply $V_{CC}$. Test conditions: $V_{CC}$ = 3.0 V, one connection at a time tested, connections not being tested are at $V_{SS}$.

## 11.4 Operating ranges

Operating ranges define those limits between which the functionality of the device is guaranteed.

### 11.4.1 Power supply voltages

$V_{CC}$ ..... 2.7 V to 3.6 V

### 11.4.2 Temperature ranges

| Parameter | Symbol | Devices | Spec | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| Ambient temperature | $T_A$ | Industrial (I) | –40 | +85 | °C |
| | | Industrial Plus (V) | –40 | +105 | °C |
| | | Automotive, AEC-Q100 grade 3 (A) | –40 | +85 | °C |
| | | Automotive, AEC-Q100 grade 2 (B) | –40 | +105 | °C |
| | | Automotive, AEC-Q100 grade 1 (M) | –40 | +125 | °C |

### 11.4.3 Input signal overshoot

During DC conditions, input or I/O signals should remain equal to or between $V_{SS}$ and $V_{CC}$. During voltage transitions, inputs or I/Os may overshoot VSS to –1.0 V or overshoot to $V_{CC}$ + 1.0 V, for periods up to 20 ns.

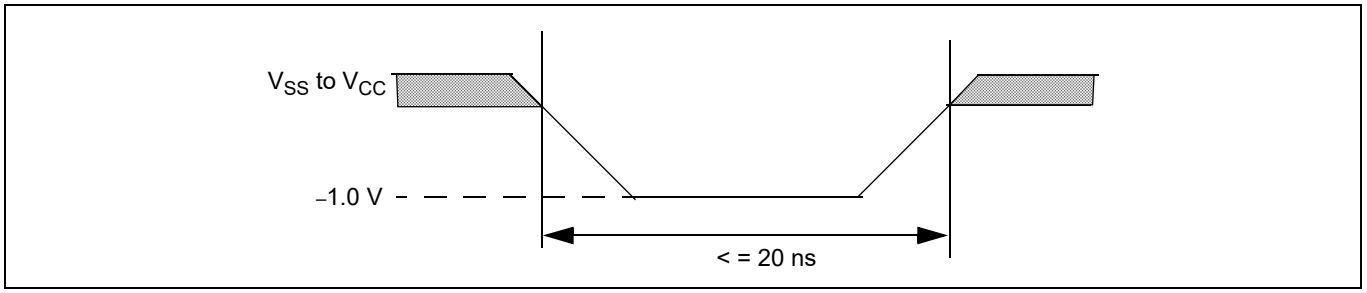



**Figure 129 Maximum negative overshoot waveform**

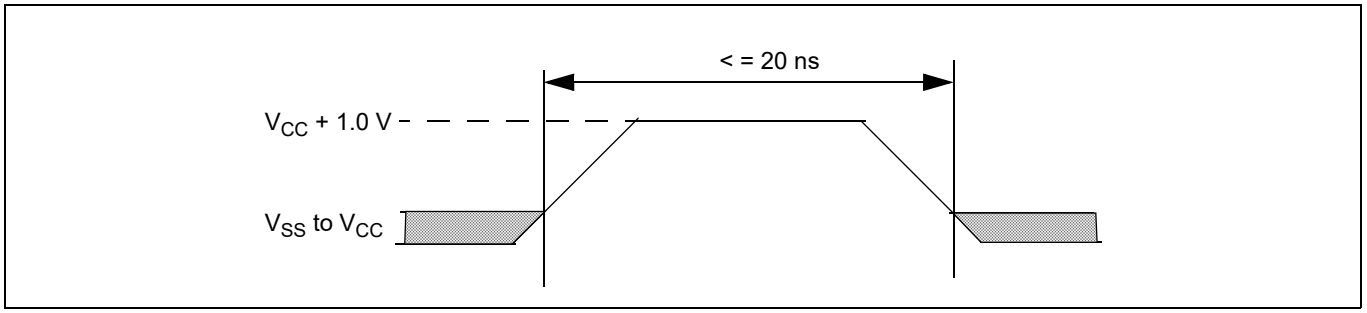



**Figure 130 Maximum positive overshoot waveform**

## 11.5 Power-up and power-down

The device must not be selected at power-up or power-down (that is, CS# must follow the voltage applied on $V_{CC}$) until $V_{CC}$ reaches the correct value as follows:

- $V_{CC}$ (min) at power-up, and then for a further delay of $t_{PU}$
- $V_{SS}$ at power-down

User is not allowed to enter any command until a valid delay of $t_{PU}$ has elapsed after the moment that $V_{CC}$ rises above the minimum $V_{CC}$ threshold. See **Figure 131**. However, correct operation of the device is not guaranteed if $V_{CC}$ returns below $V_{CC}$ (min) during $t_{PU}$. No command should be sent to the device until the end of $t_{PU}$.

The device draws $I_{POR}$ during $t_{PU}$. After power-up ($t_{PU}$), the device is in Standby mode, draws CMOS standby current ($I_{SB}$), and the WEL bit is reset.

During power-down or if supply voltage drops below $V_{CC}$(cut-off), the supply voltage must stay below $V_{CC}$(low) for a period of $t_{PD}$ for the part to initialize correctly on power-up. See **Figure 132**. If during a voltage drop the $V_{CC}$ stays above $V_{CC}$(cut-off) the part will stay initialized and will work correctly when $V_{CC}$ is again above $V_{CC}$(min). In the event power-on reset (POR) did not complete correctly after power up, the assertion of the RESET# signal or receiving a Software Reset command (RSTEN 66h followed by RST 99h) will restart the POR process.

If $V_{CC}$ drops below the $V_{CC\ (Cut\text{-}off)}$ during an embedded program or erase operation the embedded operation may be aborted and the data in that memory area may be incorrect.

Normal precautions must be taken for supply rail decoupling to stabilize the $V_{CC}$ supply at the device. Each device in a system should have the $V_{CC}$ rail decoupled by a suitable capacitor close to the package supply connection (this capacitor is generally of the order of 0.1 µF).

**Table 47 Power-up / power-down voltage and timing**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ (min) | $V_{CC}$ (minimum operation voltage) | 2.7 | – | V |
| $V_{CC}$ (cut-off) | $V_{CC}$ (cut off where re-initialization is needed) | 2.4[61] | – | |
| $V_{CC}$ (low) | $V_{CC}$ (low voltage for initialization to occur) | 1.0[62] | – | |
| $t_{PU}$ | $V_{CC}$(min) to read operation | – | 300 | µs |
| $t_{PD}$ | $V_{CC}$(low) time | 10.0 | – | |

**Notes**
61.Re-initialization is needed if $V_{CC}$ drops below 2.4 V.
62.$V_{CC}$ need to go below 1.0 V for initialization to occur.

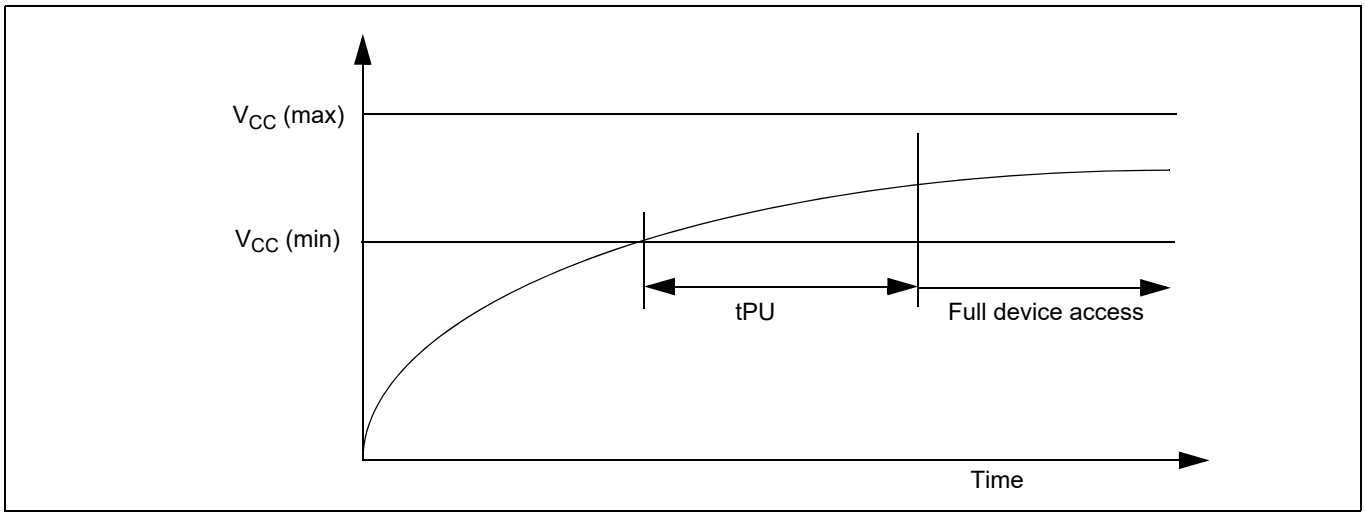



**Figure 131 Power-up**[63, 64]

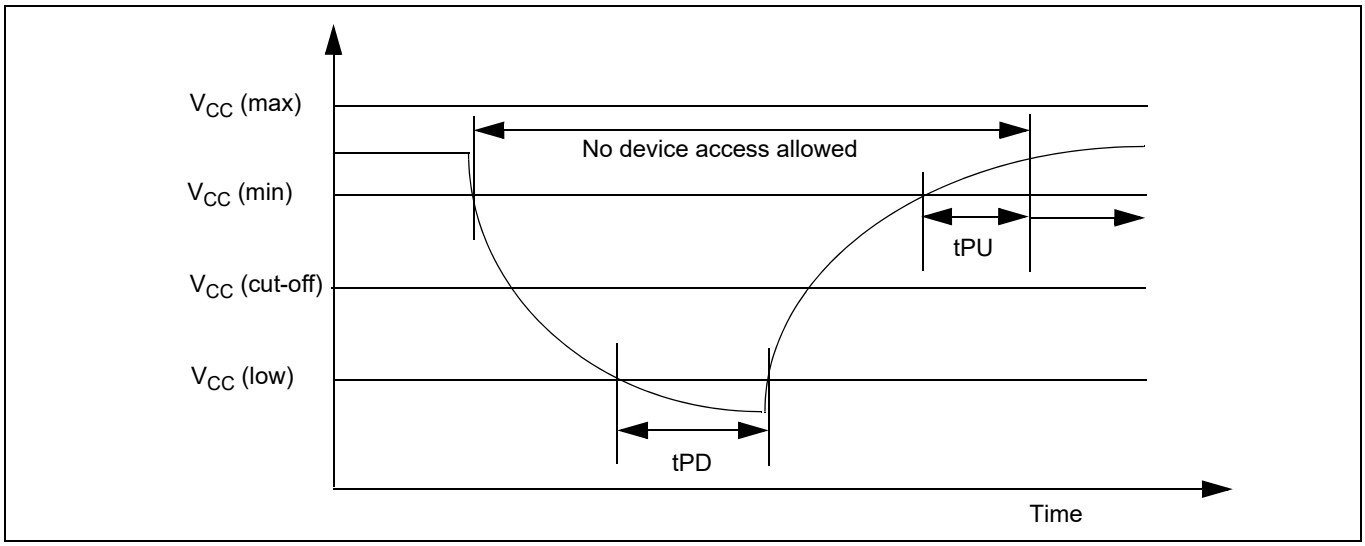



**Figure 132 Power-down and voltage drop**

**Notes**
63.Re-initialization is needed if $V_{CC}$ drops below 2.4 V.
64.$V_{CC}$ need to go below 1.0 V for initialization to occur.

## 11.6 DC characteristics

**Table 48 DC characteristics — Operating temperature range –40°C to +85°C**

| Symbol | Parameter | Test conditions | Min | Typ[65] | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage | | –0.5 | – | 0.3 × $V_{CC}$ | V |
| $V_{IH}$ | Input high voltage | | 0.7 × $V_{CC}$ | – | $V_{CC}$ + 0.4 | V |
| $V_{OL}$ | Output low voltage | $I_{OL}$ = 0.1 mA, $V_{CC}$ = $V_{CC}$ min | | – | 0.2 | V |
| $V_{OH}$ | Output high voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ – 0.2 | – | | V |
| $I_{LI}$ | Input leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | – | ±2 | µA |
| $I_{LO}$ | Output leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | | ±2 | µA |
| $I_{CC1}$ | Active power supply current (READ)[66] | Serial SDR @ 5 MHz<br>Serial SDR @ 10 MHz<br>Serial SDR @ 20 MHz<br>Serial SDR @ 50 MHz<br>Serial SDR @ 108 MHz<br>QIO/QPI SDR @ 108 MHz<br>QIO/QPI DDR @ 30 MHz<br>QIO/QPI DDR @ 54 MHz | – | 10<br>10<br>10<br>15<br>20<br>20<br>15<br>17 | 15<br>15<br>15<br>20<br>25<br>30<br>20<br>25 | mA |
| $I_{CC2}$ | Active power supply current (page program) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC3}$ | Active power supply current (WRR or WRAR) | CS# = $V_{CC}$ | – | 11 | 20 | mA |
| $I_{CC4}$ | Active power supply current (SE) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC5}$ | Active power supply current (HBE, BE) | CS# = $V_{CC}$ | – | 15 | 25 | mA |
| $I_{SB}$ | Standby current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: SPI, dual I/O and Quad I/O modes | – | 20 | 30 | µA |
| | | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: QPI mode | – | 35 | 55 | µA |
| $I_{DPD}$ | Deep power down current | RESET#, CS# = $V_{CC}$, $V_{IN}$ = GND or $V_{CC}$ | – | 2 | 20 | µA |
| $I_{POR}$[67] | Power on reset current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | – | 3 | 5 | mA |

**Notes**
65. Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 3.0 V.
66. Outputs unconnected during read data return. Output switching current is not included.
67. In-rush/peak current up to 25 mA during POR with current specified represent time average for $t_{PU}$ duration.

**Table 49 DC characteristics — Operating temperature range –40°C to +105°C**

| Symbol | Parameter | Test conditions | Min | Typ[68] | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage | – | –0.5 | – | $0.3 \times V_{CC}$ | V |
| $V_{IH}$ | Input high voltage | – | $0.7 \times V_{CC}$ | – | $V_{CC}$+0.4 | V |
| $V_{OL}$ | Output low voltage | $I_{OL}$ = 0.1 mA, $V_{CC}$ = $V_{CC}$ min | | – | 0.2 | V |
| $V_{OH}$ | Output high voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ - 0.2 | – | | V |
| $I_{LI}$ | Input leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | – | ±4 | µA |
| $I_{LO}$ | Output leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | – | ±4 | µA |
| $I_{CC1}$ | Active power supply current (READ)[69] | Serial SDR @ 5 MHz<br>Serial SDR @ 10 MHz<br>Serial SDR @ 20 MHz<br>Serial SDR @ 50 MHz<br>Serial SDR @ 108 MHz<br>QIO/QPI SDR @ 108 MHz<br>QIO/QPI DDR @ 30 MHz<br>QIO/QPI DDR @ 54 MHz | – | 10<br>10<br>10<br>15<br>20<br>20<br>15<br>17 | 15<br>15<br>20<br>25<br>30<br>30<br>15<br>25 | mA |
| $I_{CC2}$ | Active power supply current (page program) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC3}$ | Active power supply current (WRR or WRAR) | CS# = $V_{CC}$ | – | 11 | 20 | mA |
| $I_{CC4}$ | Active power supply current (SE) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC5}$ | Active power supply current (HBE, BE) | CS# = $V_{CC}$ | – | 15 | 25 | mA |
| $I_{SB}$ | Standby current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: SPI, dual I/O and Quad I/O modes | – | 20 | 40 | µA |
| | | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: QPI mode | – | 35 | 70 | µA |
| $I_{DPD}$ | Deep power down current | RESET#, CS# = $V_{CC}$, $V_{IN}$ = GND or $V_{CC}$ | – | 2 | 30 | µA |
| $I_{POR}$[70] | Power on reset current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | – | 3 | 7 | mA |

**Notes**
68. Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 3.0 V.
69. Outputs unconnected during read data return. Output switching current is not included.
70. In-rush/peak current up to 25 mA during POR with current specified represent time average for $t_{PU}$ duration.

**Table 50 DC Characteristics — Operating temperature range -40°C to +125°C**

| Symbol | Parameter | Test conditions | Min | Typ[71] | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low voltage | – | –0.5 | – | 0.3 × $V_{CC}$ | V |
| $V_{IH}$ | Input high voltage | – | 0.7 × $V_{CC}$ | – | $V_{CC}$ + 0.4 | V |
| $V_{OL}$ | Output low voltage | $I_{OL}$ = 0.1 mA, $V_{CC}$ = $V_{CC}$ min | | – | 0.2 | V |
| $V_{OH}$ | Output high voltage | $I_{OH}$ = –0.1 mA | $V_{CC}$ – 0.2 | – | | V |
| $I_{LI}$ | Input leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | – | ±4 | µA |
| $I_{LO}$ | Output leakage current | $V_{CC}$ = $V_{CC}$ Max, $V_{IN}$ = $V_{IH}$ or $V_{SS}$, CS# = $V_{IH}$ | – | – | ±4 | µA |
| $I_{CC1}$ | Active power supply current (READ)[72] | Serial SDR @ 5 MHz<br>Serial SDR @ 10 MHz<br>Serial SDR @ 20 MHz<br>Serial SDR @ 50 MHz<br>Serial SDR @ 108 MHz<br>QIO/QPI SDR @ 108 MHz<br>QIO/QPI DDR @ 30 MHz<br>QIO/QPI DDR @ 54 MHz | – | 10<br>10<br>10<br>15<br>20<br>20<br>15<br>17 | 15<br>15<br>20<br>25<br>30<br>30<br>15<br>25 | mA |
| $I_{CC2}$ | Active power supply current (Page Program) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC3}$ | Active power supply current (WRR or WRAR) | CS# = $V_{CC}$ | – | 11 | 20 | mA |
| $I_{CC4}$ | Active power supply current (SE) | CS# = $V_{CC}$ | – | 17 | 25 | mA |
| $I_{CC5}$ | Active power supply current (HBE, BE) | CS# = $V_{CC}$ | – | 15 | 25 | mA |
| $I_{SB}$ | Standby current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: SPI, dual I/O and Quad I/O modes | – | 20 | 60 | µA |
| | | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$: QPI mode | – | 35 | 70 | µA |
| $I_{DPD}$ | Deep power down current | RESET#, CS# = $V_{CC}$, $V_{IN}$ = GND or $V_{CC}$ | – | 2 | 40 | µA |
| $I_{POR}$[73] | Power on reset current | RESET#, CS# = $V_{CC}$; SI, SCK = $V_{CC}$ or $V_{SS}$ | – | 3 | 9 | mA |

**Notes**
71. Typical values are at $T_{AI}$ = 25°C and $V_{CC}$ = 3.0 V.
72. Outputs unconnected during read data return. Output switching current is not included.
73. In-rush/peak current up to 25 mA during POR with current specified represent time average for $t_{PU}$ duration.

### 11.6.1 Active Power and Standby Power modes

The device is enabled and in the Active Power mode when Chip Select (CS#) is Low. When CS# is HIGH, the device is disabled, but may still be in an Active Power mode until all program, erase, and write operations have completed. The device then goes into the Standby Power mode, and power consumption drops to $I_{SB}$.

### 11.6.2 Deep Power Down Power mode (DPD)

The Deep Power Down mode is enabled by inputing the command instruction code “B9h” and the power consumption drops to $I_{DPD}$. In DPD mode the device responds only to the resume from DPD command (RES ABh) or Hardware reset (RESET# and IO3 / RESET#). All other commands are ignored during DPD mode.

# 12 Timing specifications

## 12.1 Key to switching waveforms

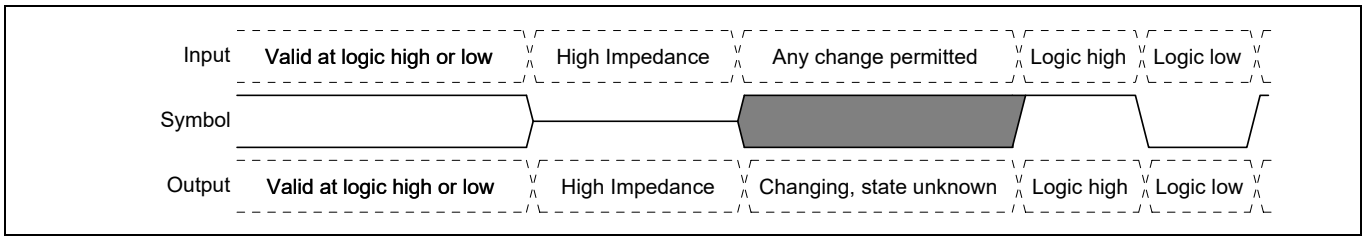



**Figure 133 Waveform element meanings**

## 12.2 AC test conditions

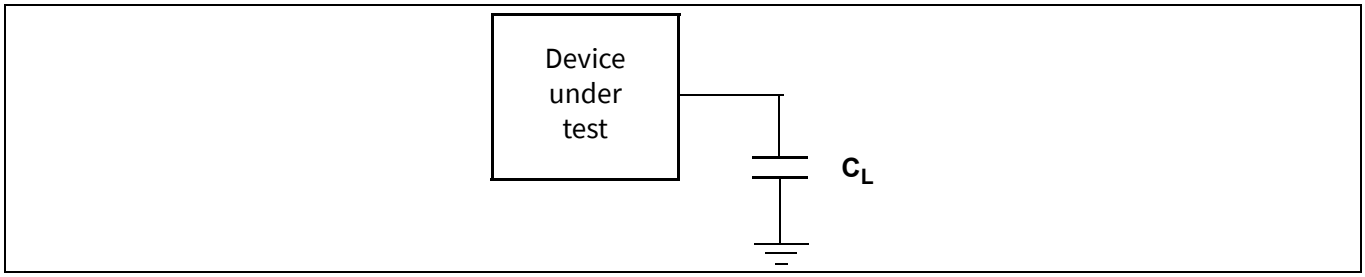



**Figure 134 Test setup**

**Table 51 AC measurement conditions**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $C_L$ | Load capacitance | – | 15 / 30[74] | pF |
| – | Input pulse voltage | 0.2 × $V_{CC}$ | 0.8 × $V_{CC}$ | V |
| – | Input timing ref Voltage | 0.5 × $V_{CC}$ | | |
| – | Output timing ref voltage | 0.5 × $V_{CC}$ | | |

**Notes**
74. Load capacitance depends on the operation frequency or mode of operation.
75. AC characteristics tables assume clock and data signals have the same slew rate (slope). See **"SDR AC characteristics[79]"** on page 154 note **84** for slew Rates at operating frequency's.

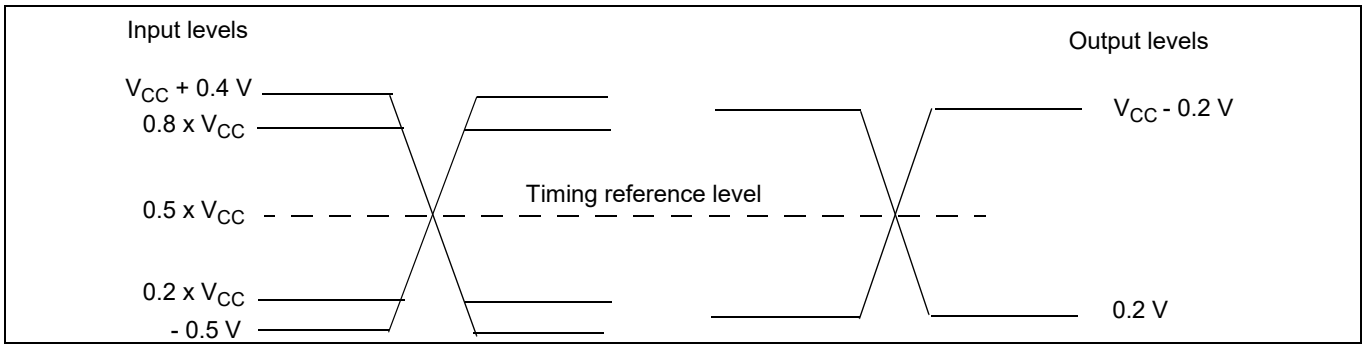



**Figure 135 Input, output, and timing reference levels**

### 12.2.1 Capacitance characteristics

**Table 52 Capacitance**

| Symbol | Parameter | Test conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $C_{IN}$ | Input capacitance (applies to SCK, CS#, RESET#, IO3 / RESET#) | 1 MHz | – | 8 | pF |
| $C_{OUT}$ | Output capacitance (applies to All I/O) | 1 MHz | – | 8 | pF |

## 12.3 Reset

If a hardware reset is initiated during a erase, program or writing of a register operation the data in that sector, page or register is not stable, the operation that was interrupted needs to be initiated again. If a hardware reset is initiated during a software reset operation, the hardware reset might be ignored.

### 12.3.1 Power-on (cold) reset

The device executes a power-on reset (POR) process until a time delay of $t_{PU}$ has elapsed after the moment that $V_{CC}$ rises above the minimum $V_{CC}$ threshold. See **Figure 131** and **Table 47**. The device must not be selected (CS# to go HIGH with $V_{CC}$) during power-up ($t_{PU}$), i.e. no commands may be sent to the device until the end of $t_{PU}$.

RESET# and IO3 / RESET# reset function is ignored during POR. If RESET# or IO3 / RESET# is LOW during POR and remains low through and beyond the end of $t_{PU}$, CS# must remain HIGH until $t_{RH}$ after RESET# and IO3 / RESET# returns HIGH. RESET# and IO3 / RESET# must return HIGH for greater than $t_{RS}$ before returning low to initiate a hardware reset.

The IO3 / RESET# input functions as the RESET# signal when CS# is HIGH for more than $t_{CS}$ time or when Quad or QPI mode is not enabled CR1V[1] = 0 or CR2V[3] = 0.

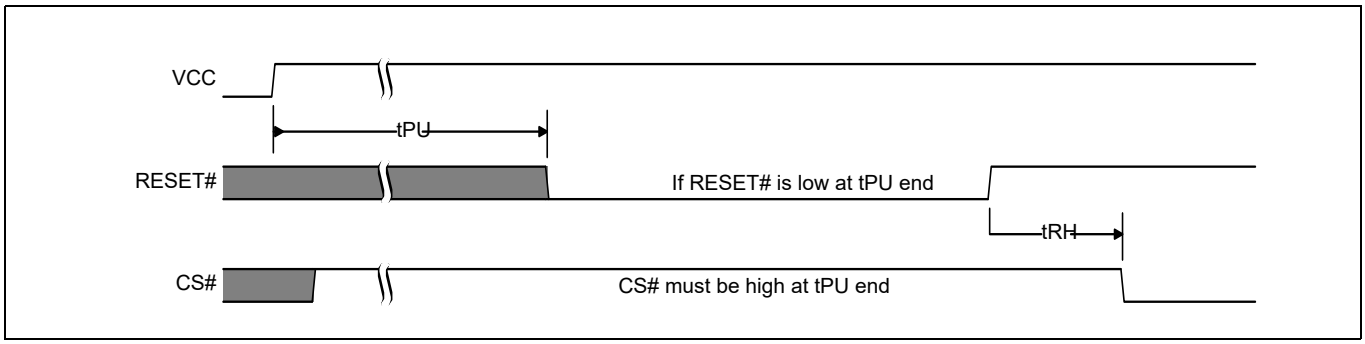



**Figure 136 Reset LOW at the end of POR**

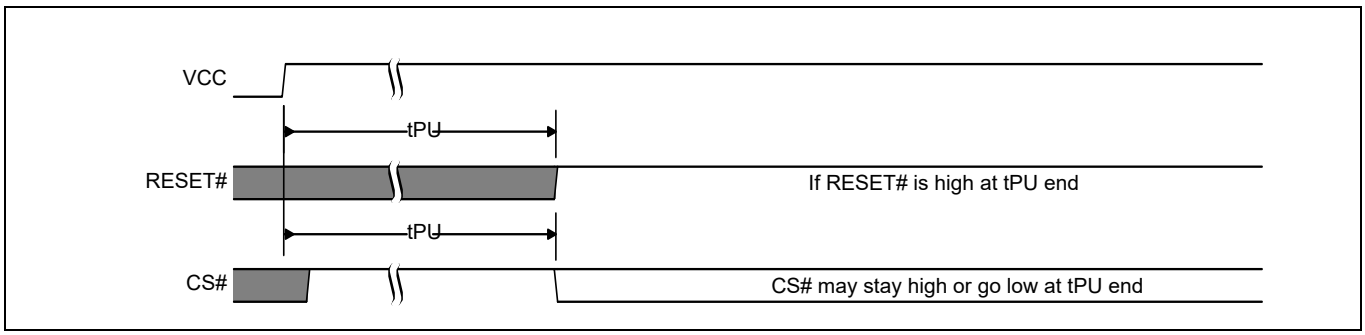



**Figure 137 Reset HIGH at the end of POR**

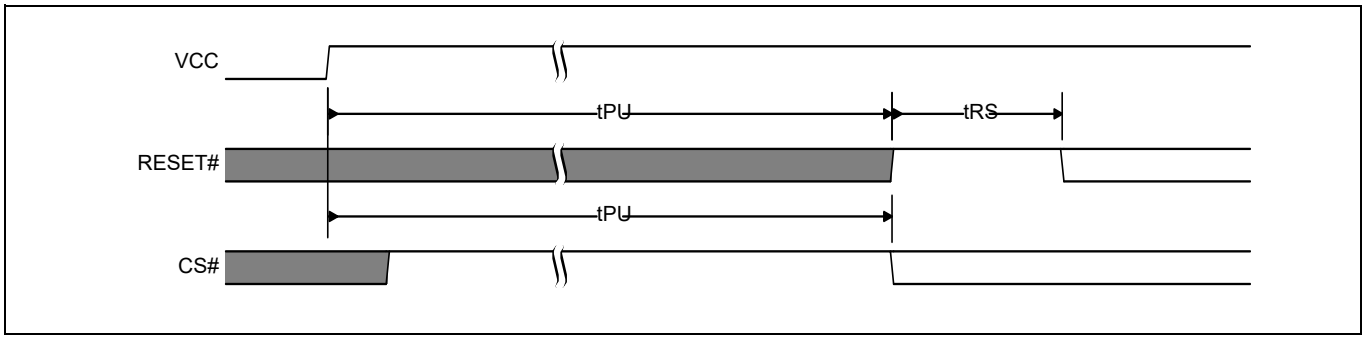



**Figure 138 POR followed by hardware reset**

## 12.3.2 RESET # and IO3 / RESET# input initiated hardware (warm) reset

The RESET# and IO3 / RESET# inputs can function as the RESET# signal. Both inputs can initiate the reset operation under conditions.

The RESET# input initiates the reset operation when transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, the device will reset register states in the same manner as power-on reset but, does not go through the full reset process that is performed during POR. The hardware reset process requires a period of $t_{RPH}$ to complete. The RESET# input is available only on the SOIC 16 lead and BGA ball packages.

The IO3 / RESET# input initiates the reset operation under the following when CS# is HIGH for more than $t_{CS}$ time or when quad or QPI mode is not enabled CR1V[1] = 0 or CR2V[3] = 0. The IO3 / RESET# input has an internal pull-up to $V_{CC}$ and may be left unconnected if quad or QPI mode is not used. The $t_{CS}$ delay after CS# goes HIGH gives the memory or host system time to drive IO3 HIGH after its use as a quad or QPI mode I/O signal while CS# was LOW. The internal pull-up to $V_{CC}$ will then hold IO3 / RESET# HIGH until the host system begins driving IO3 / RESET#. The IO3 / RESET# input is ignored while CS# remains HIGH during $t_{CS}$, to avoid an unintended reset operation. If CS# is driven LOW to start a new command, IO3 / RESET# is used as IO3.

When the device is not in quad or QPI mode or, when CS# is HIGH, and IO3 / RESET# transitions from $V_{IH}$ to $V_{IL}$ for > $t_{RP}$, following $t_{CS}$, the device will reset register states in the same manner as POR but, does not go through the full reset process that is performed during POR.

The hardware reset process requires a period of $t_{RPH}$ to complete. If the POR process did not complete correctly for any reason during power-up ($t_{PU}$), RESET# going LOW will initiate the full POR process instead of the hardware reset process and will require $t_{PU}$ to complete the POR process.

The Software Reset command (RSTEN 66h followed by RST 99h) is independent of the state of RESET # and IO3 / RESET#. If RESET# and IO3 / RESET# is HIGH or unconnected, and the software reset instructions are issued, the device will perform software reset.

Additional notes:

- If both RESET# and IO3 / RESET# input options are available use only one reset option in your system. IO3 / RESET# input reset operation can be disable by setting CR2NV[7] = 0 (See **Table 12**) setting the IO3_RESET to only operate as IO3. The RESET# input can be disable by not connecting or tying the RESET# input to $V_{IH}$. RESET# and IO3 / RESET# must be high for $t_{RS}$ following $t_{PU}$ or $t_{RPH}$, before going low again to initiate a hardware reset.
- When IO3 / RESET# is driven LOW for at least a minimum period of time ($t_{RP}$), following $t_{CS}$, the device terminates any operation in progress, makes all outputs high impedance, and ignores all read/write commands for the duration of $t_{RPH}$. The device resets the interface to STANDBY state.
- If Quad or QPI mode and the IO3 / RESET# feature are enabled, the host system should not drive IO3 low during $t_{CS,}$ to avoid driver contention on IO3. Immediately following commands that transfer data to the host in quad or QPI mode, e.g. Quad I/O read, the memory drives IO3 / RESET# HIGH during $t_{CS,}$ to avoid an unintended reset operation. Immediately following commands that transfer data to the memory in Quad mode, e.g. page program, the host system should drive IO3 / RESET# HIGH during $t_{CS,}$ to avoid an unintended reset operation.
- If Quad or QPI mode is not enabled, and if CS# is LOW at the time IO3 / RESET# is asserted LOW, CS# must return HIGH during $t_{RPH}$ before it can be asserted low again after $t_{RH}$.

**Table 53 Hardware reset parameters**[76, 77, 78]

| Parameter | Description | Limit | Time | Unit |
|---|---|---|---|---|
| $t_{RS}$ | Reset setup - prior reset end and RESET# HIGH before RESET# LOW | Min | 50 | ns |
| $t_{RPH}$ | Reset pulse hold - RESET# LOW to CS# LOW | Min | 100 | µs |
| $t_{RP}$ | RESET# pulse width | Min | 200 | ns |
| $t_{RH}$ | Reset hold - RESET# HIGH before CS# LOW | Min | 150 | ns |

**Notes**
76. RESET# and IO3 / RESET# Low is ignored during power-up ($t_{PU}$). If RESET# is asserted during the end of $t_{PU}$, the device will remain in the reset state and $t_{RH}$ will determine when CS# may go LOW.
77. If quad or QPI mode is enabled, IO3 / RESET# Low is ignored during $t_{CS}$.
78. Sum of $t_{RP}$ and $t_{RH}$ must be equal to or greater than $t_{RPH}$.

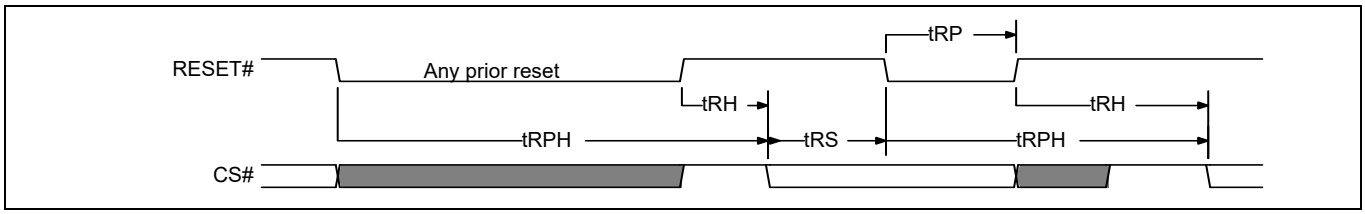



**Figure 139 Hardware reset using RESET# input**

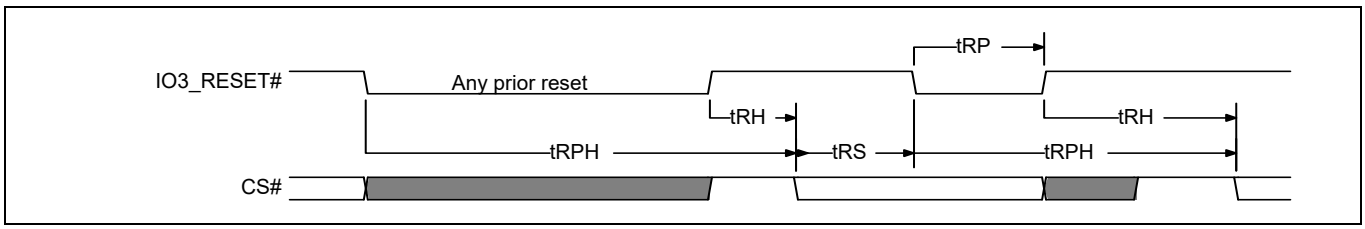



**Figure 140 Hardware reset when Quad or QPI mode is not enabled and IO3 / RESET# is enabled**

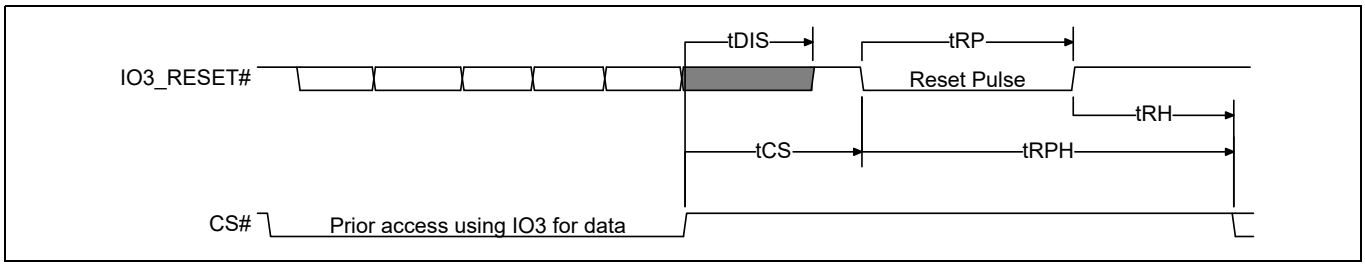



**Figure 141 Hardware reset when Quad or QPI mode and IO3 / RESET# are enabled**

## 12.4 SDR AC characteristics

**Table 54 SDR AC characteristics**[79]

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $F_{SCK, R}$ | SCK clock frequency for READ and 4READ instructions | DC | 50 | MHz |
| $F_{SCK, C}$ | SCK Clock frequency for the following dual and quad commands: QOR, 4QOR, DIOR, 4DIOR, QIOR, 4QIOR | DC | 108 | MHz |
| $P_{SCK}$ | SCK clock period | 1/ $F_{SCK}$ | – | MHz |
| $t_{WH}$, $t_{CH}$ | Clock high time | 50% $P_{SCK}$ – 5% | – | ns |
| $t_{WL}$, $t_{CL}$ | Clock Low time | 50% $P_{SCK}$ – 5% | – | ns |
| $t_{CRT}$, $t_{CLCH}$ | Clock rise time (slew rate)[80] | 0.1 | – | V/ns |
| $t_{CFT}$, $t_{CHCL}$ | Clock fall time (slew rate)[80] | 0.1 | – | V/ns |
| $t_{CS}$ | CS# high time (any read instructions) | 20 | – | ns |
| | CS# high time (All other non-read instructions) | 50 | – | ns |
| $t_{CSS}$ | CS# active setup time (relative to SCK) | 3 | – | ns |
| $t_{CSH}$ | CS# active hold time (relative to SCK) | 5 | – | ns |
| $t_{SU}$ | Data in setup time | 3 | – | ns |
| $t_{HD}$ | Data in hold time | 2 | – | ns |
| $t_V$ | Clock low to output valid | | 8[80]<br>6[81] | ns |
| $t_{HO}$ | Output hold time | 1 | | ns |
| $t_{DIS}$ | Output disable time[82]<br>Output disable time (when reset feature and Quad mode are both enabled) | – | 8<br>20[83] | ns |
| $t_{WPS}$ | WP# setup time[84] | 20 | – | ns |
| $t_{WPH}$ | WP# hold time[84] | 100 | – | ns |
| $T_{DP}$ | CS# High to Deep Power Down mode | – | 3 | µs |
| $T_{RES}$ | CS# High to release from Deep Power Down mode | – | 5 | µs |
| $t_{QEN}$ | QIO or QPI Enter mode, time needed to issue next command | – | 1.5 | µs |
| $t_{QEXN}$ | QIO or QPI Exit mode, time needed to issue next command | – | 1 | µs |

**Notes**
79. $t_{CRT}$, $t_{CLCH}$ clock rise and fall slew rate for fast clock (108 MHz) min is 1.5 V/ns and for slow clock (50 MHz) min is 1.0 V/ns.
80. Full $V_{CC}$ range and CL = 30 pF.
81. Full $V_{CC}$ range and CL = 15 pF.
82. Output HI-Z is defined as the point where data is no longer driven.
83. $t_{DIS}$ require additional time when the reset feature and Quad mode are enabled (CR2V[7] = 1 and CR1V[1] = 1).
84. Only applicable as a constraint for WRR or WRAR instruction when SRP0 is set to a 1.

## 12.4.1 Clock timing

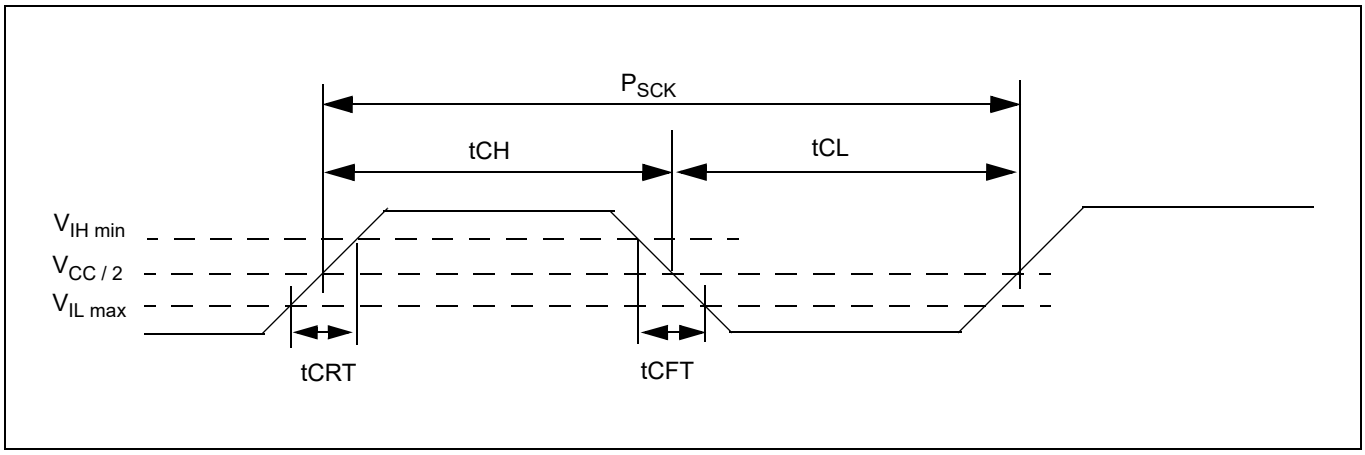



**Figure 142 Clock timing**

## 12.4.2 Input / output timing

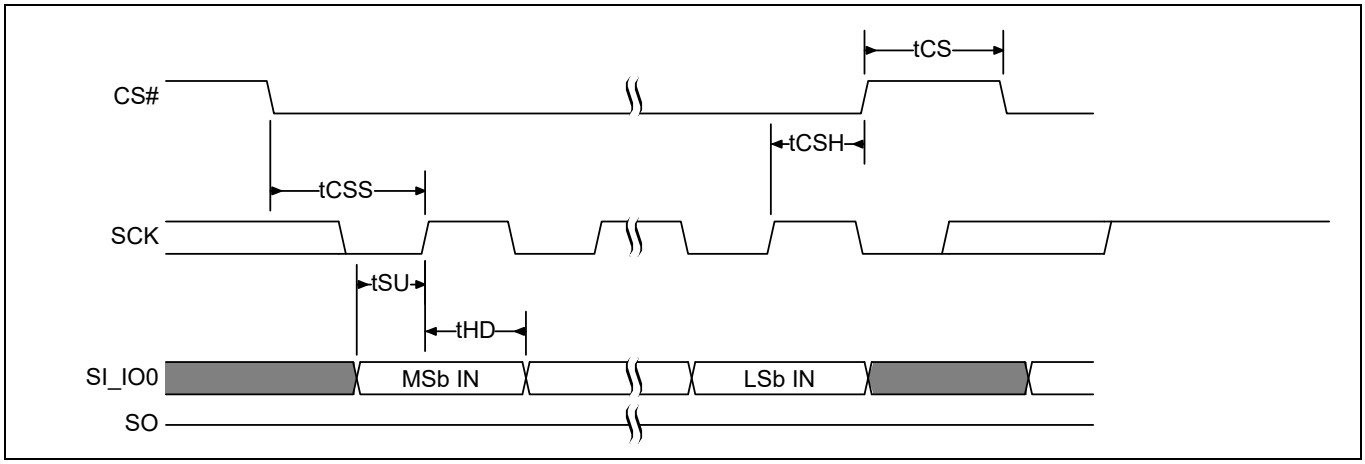



**Figure 143 SPI single bit input timing**

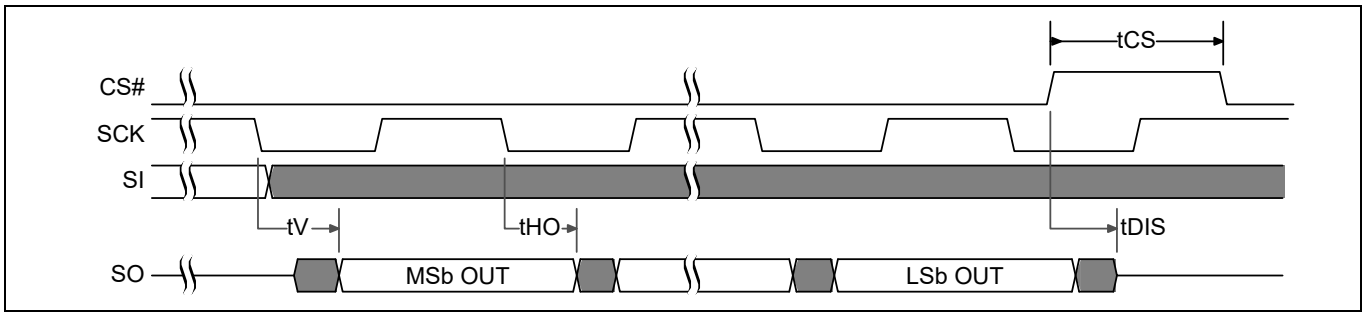



**Figure 144 SPI single bit output timing**

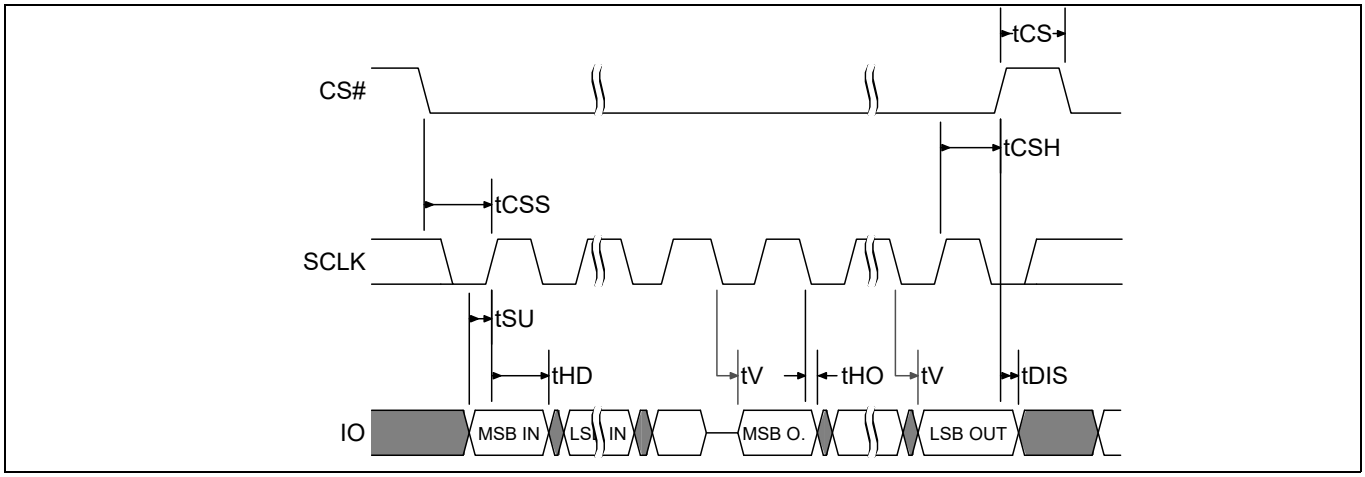



**Figure 145 SDR MIO timing**

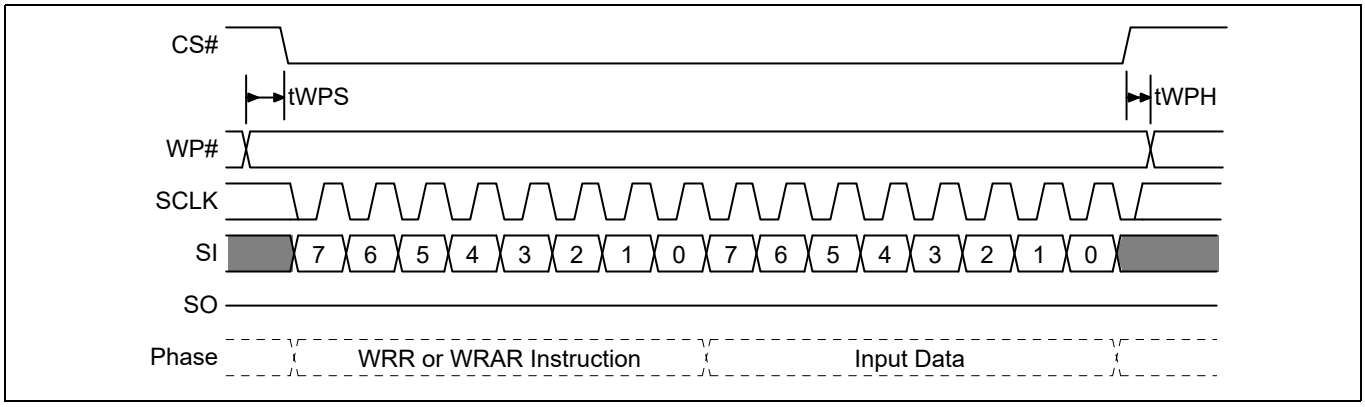



**Figure 146 WP# input timing**

## 12.5 DDR AC characteristics

**Table 55 DDR AC characteristics 54 MHz operation**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $F_{SCK, R}$ | SCK clock frequency for DDR READ instruction | DC | 54 | MHz |
| $P_{SCK, R}$ | SCK clock period for DDR READ instruction | 1/ $F_{SCK}$ | – | ns |
| $t_{crt}$ | Clock rise time (slew rate) | 1.5 | – | V/ns |
| $t_{cft}$ | Clock fall time (slew rate) | 1.5 | – | V/ns |
| $t_{WH}$, $t_{CH}$ | Clock high time | 50% $P_{SCK}$ – 5% | – | ns |
| $t_{WL}$, $t_{CL}$ | Clock low time | 50% $P_{SCK}$ – 5% | – | ns |
| $t_{CS}$ | CS# HIGH time (read instructions)<br>CS# HIGH time (read instructions when reset feature is enabled) | 20<br>50 | – | ns |
| $t_{CSS}$ | CS# active setup time (relative to SCK) | 3 | – | ns |
| $t_{SU}$ | IO in setup time | 3 | – | ns |
| $t_{HD}$ | IO in hold time | 2 | – | ns |
| $t_V$ | Clock low to output valid | – | 8[85]<br>6[86] | ns |
| $t_{HO}$ | Output hold time | 1 | – | ns |
| $t_{DIS}$ | Output disable time<br>Output disable time (when reset feature is enabled) | – | 8<br>20 | ns |
| $t_{O_skew}$ | First IO to last IO data valid time | – | 600[87] | ps |

**Notes**
85.Full $V_{CC}$ range and CL = 30 pF.
86.Full $V_{CC}$ range and CL = 15 pF.
87.Not tested.

### 12.5.1 DDR input timing

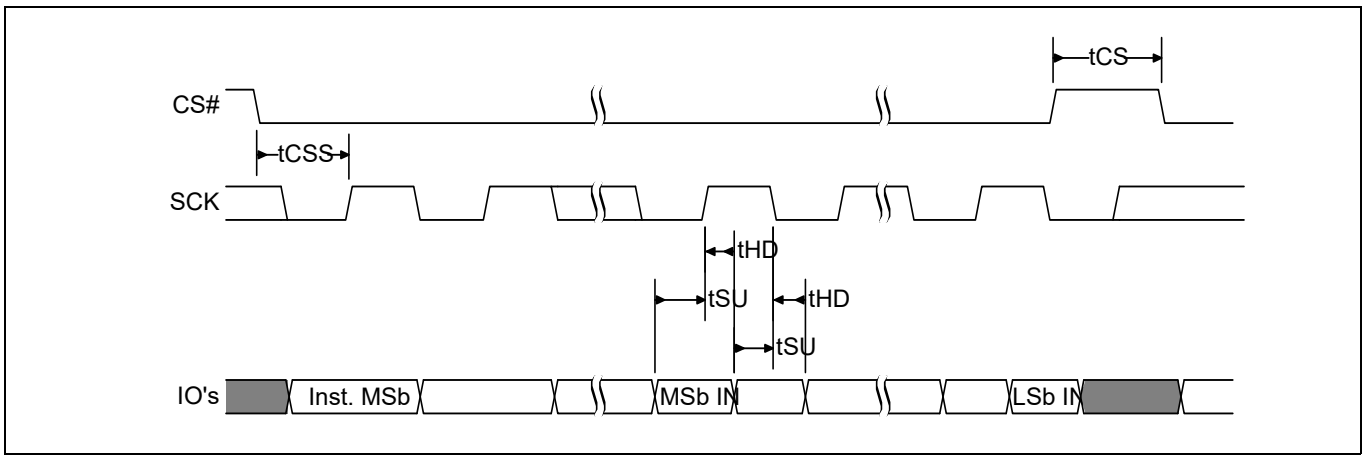



**Figure 147 SPI DDR input timing**

## 12.5.2 DDR output timing

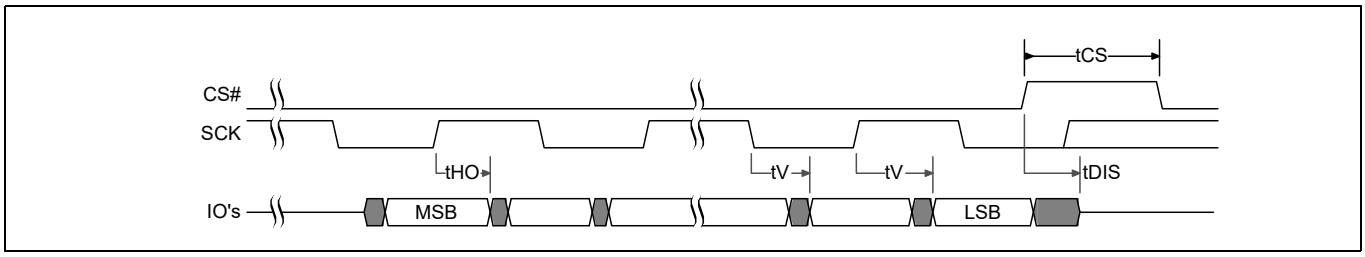



**Figure 148 SPI DDR output timing**

## 12.5.3 DDR data valid timing using DLP

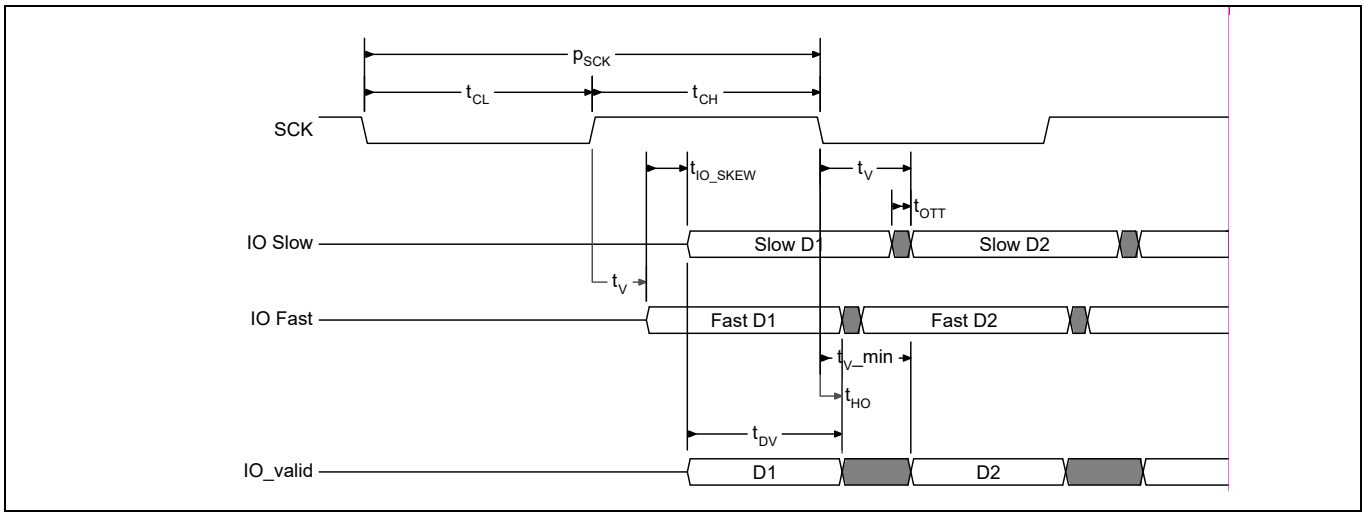



**Figure 149 SPI DDR data valid window**

Timing specifications

The minimum data valid window ($t_{DV}$) and $t_V$ minimum can be calculated as follows:

$t_{DV}$ = minimum half clock cycle time ($t_{CLH}$[88]) - $t_{OTT}$[90] - $t_{IO_SKEW}$[89]

$t_V$ _min = $t_{HO}$ + $t_{IO_SKEW}$ + $t_{OTT}$

Example:

- 66 MHz clock frequency = 15 ns clock period, DDR operations and duty cycle of 45% or higher
  - $t_{CLH}$ = 0.45 × PSCK = 0.45 × 15 ns = 6.75 ns
- $t_{OTT}$ calculation[91] is bus impedance of 45 ohm and capacitance of 37 pf, with timing reference of 0.75 $V_{CC}$, the rise time from 0 to 1 or fall time 1 to 0 is 1.4[94] × RC time constant (Tau)[93] = 1.4 × 1.67 ns = 2.34 ns
  - $t_{OTT}$ = rise time or fall time = 2.34 ns.
- Data valid window
  - $t_{DV}$ = $t_{CLH}$ – $t_{IO_SKEW}$ – $t_{OTT}$ = 6.75 ns – 600 ps – 2.34ns = 3.81ns
- $t_V$ minimum
  - $t_V$ _min = $t_{HO}$ + $t_{IO_SKEW}$ + $t_{OTT}$ = 1.0 ns + 600 ps + 2.34 ns = 3.94 ns

**Notes**

88. $t_{CLH}$ is the shorter duration of $t_{CL}$ or $t_{CH}$.
89. $t_{IO_SKEW}$ is the maximum difference (delta) between the minimum and maximum $t_V$ (output valid) across all IO signals.
90. $t_{OTT}$ is the maximum output transition time from one valid data value to the next valid data value on each IO.
91. $t_{OTT}$ is dependent on system level considerations including:
    a. Memory device output impedance (drive strength).
    b. System level parasitics on the IOs (primarily bus capacitance).
    c. Host memory controller input $V_{IH}$ and $V_{IL}$ levels at which 0 to 1 and 1 to 0 transitions are recognized.
    d. $t_{OTT}$ is not a specification tested by Infineon, it is system dependent and must be derived by the system designer based on the above considerations.
92. $t_{DV}$ is the data valid window.
93. Tau = R (output impedance) x C (load capacitance).
94. Multiplier of Tau time for voltage to rise to 75% of $V_{CC}$.

## 12.6 Embedded algorithm performance tables

**Table 56 Program and erase performance**

| Symbol | Parameter | Min | Typ[95] | Max | Unit |
|---|---|---|---|---|---|
| $t_W$ | Non-volatile Register write time | – | 220 | 1200 | ms |
| $t_{PP}$ | Page programming (256 bytes) | – | 450 | 1350 | µs |
| $t_{BP1}$ | Byte programming (first byte) | – | 75 | 90 | µs |
| $t_{BP2}$ | Additional byte programming (after first byte) | – | 10 | 30 | µs |
| $t_{SE}$ | Sector erase time (4 KB physical sectors) | – | 65 | 320 | ms |
| $t_{HBE}$ | Half block erase time (32 KB physical sectors) | – | 300 | 600 | ms |
| $t_{BE}$ | Block erase time (64 KB physical sectors) | – | 450 | 1150 | ms |
| $t_{CE}$ | Chip erase time (S25FL064L) | – | 55 | 150 | sec |

**Notes**
95. Typical program and erase times assume the following conditions: 25°C, $V_{CC}$ = 3.0 V; checkerboard data pattern.
96. The programming time for any OTP programming command is the same as $t_{PP}$. This includes IRPP 2Fh, PASSP E8h and PDLRNV 43h.

**Table 57 Program or erase suspend AC parameters**

| Parameter | Typical | Max | Unit | Comments |
|---|---|---|---|---|
| Suspend latency ($t_{SL}$) | – | 40 | µs | The time from suspend command until the WIP bit is 0. |
| Resume to next suspend ($t_{RNS}$) | 100 | – | µs | Is the time needed to issue the next suspend command. |

# 13 Ordering information

## 13.1 Ordering part number

The ordering part number is formed by a valid combination of the following:

**S25FL 064 L AB M F I 00 1**

- **Packing type**
  0 = Tray
  1 = Tube
  3 = 13” Tape and reel
- **Model number (additional ordering options)**
  00 = SOIC16 (300 mil)
  01 = SOIC8 (208 mil) / 8-contact WSON footprint
  02 = 5x5 ball BGA footprint
  03 = 4x6 ball BGA footprint
  04 = USON (4 x 4mm)
- **Temperature range**
  I = Industrial (–40°C to +85°C)
  V = Industrial Plus (–40°C to +105°C)
  A = Automotive, AEC-Q100 grade 3 (–40°C to +85°C)
  B = Automotive, AEC-Q100 grade 2 (–40°C to +105°C)
  M = Automotive, AEC-Q100 grade 1 (–40°C to +125°C)
- **Package materials**[97]
  F = Halogen free, lead (Pb)-free
  H = Halogen free, lead (Pb)-free
- **Package type**
  M = 8-lead SOIC / 16-lead SOIC
  N = USON 4 x 4 mm / WSON 5 x 6 mm
  B = 24-ball BGA 6 x 8 mm package, 1.00 mm pitch
- **Speed**
  AB =108 MHz SDR and 54 MHz DDR
- **Device technology**
  L = Floating gate process technology
- **Density**
  064 = 64 Mb
- **Device family**
  S25FL memory 3.0 V-only, SPI flash memory

**Note**
97.Halogen free definition is in accordance with IEC 61249-2-21 specification.

## 13.2 Valid combinations — Standard

Valid combinations list configurations planned to be supported in volume for this device. Contact your local sales office to confirm availability of specific valid combinations and to check on newly released combinations.

**Table 58 Valid Combinations — Standard**

| Base ordering part number | Speed option | Package and temperature | Model number | Packing type | Package marking |
|---|---|---|---|---|---|
| S25FL064L | AB | MFI, MFV | 00, 01 | 0, 1, 3 | FL064L + (temp) + F + (model number) |
| | AB | NFI, NFV | 01, 04 | 0, 1, 3 | FL064L + (temp) + F + (model number) |
| | AB | BHI, BHV | 02, 03 | 0, 3 | FL064L + (temp) + H + (model number) |

## 13.3 Valid combinations — Automotive grade / AEC-Q100

The **Table 59** lists configurations that are automotive grade / AEC-Q100 qualified and are planned to be available in volume. The table will be updated as new combinations are released. Consult your local sales representative to confirm availability of specific combinations and to check on newly released combinations.

Production part approval process (PPAP) support is only provided for AEC-Q100 grade products.

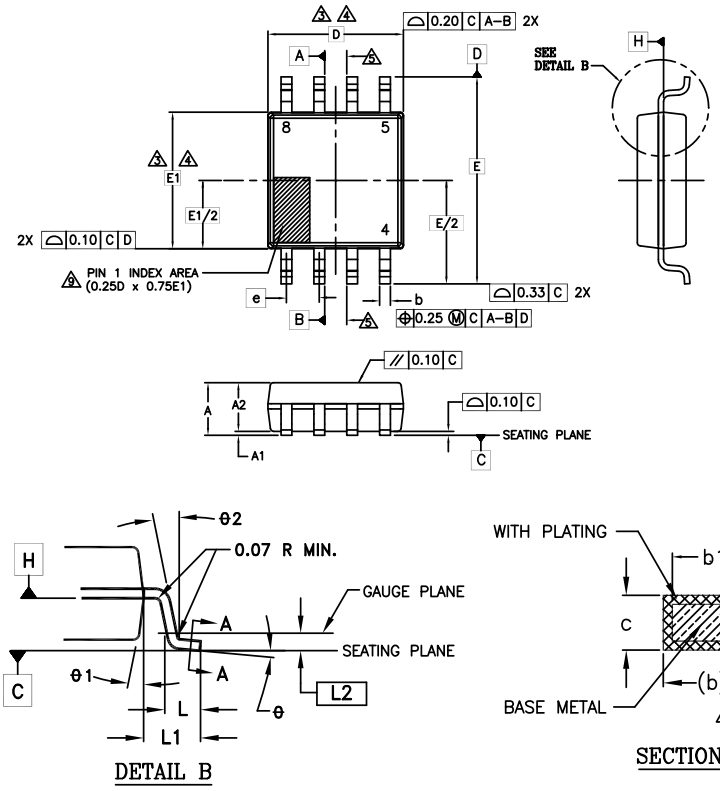
Products to be used in end-use applications that require ISO/TS-16949 compliance must be AEC-Q100 grade products in combination with PPAP. Non–AEC-Q100 grade products are not manufactured or documented in full compliance with ISO/TS-16949 requirements.

AEC-Q100 grade products are also offered without PPAP support for end-use applications that do not require ISO/TS-16949 compliance.

**Table 59 Valid combinations — Automotive grade / AEC-Q100**

| Base ordering part number | Speed option | Package and temperature | Model number | Packing type | Package marking |
|---|---|---|---|---|---|
| S25FL064L | AB | MFA, MFB, MFM | 00, 01 | 0, 1, 3 | FL064L + (temp) + F + (model number) |
| | AB | NFA, NFB, NFM | 01, 04 | 0, 1, 3 | FL064L + (temp) + F + (model number) |
| | AB | BHA, BHB, BHM | 02, 03 | 0, 3 | FL064L + (temp) + H + (model number) |

# 14 Physical diagrams



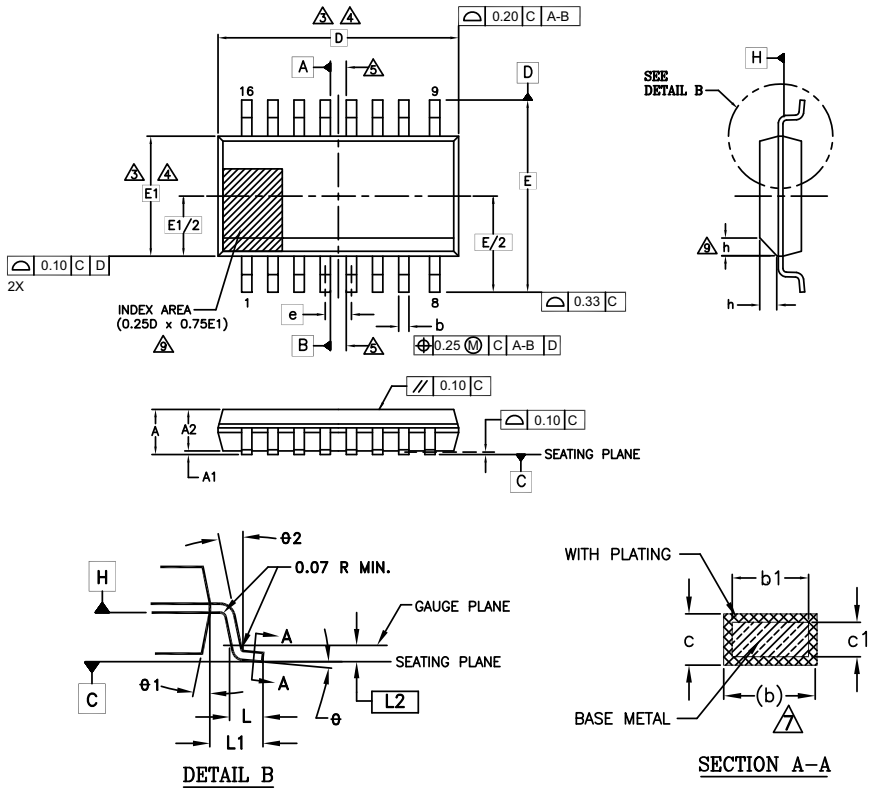

| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | 1.75 | - | 2.16 |
| A1 | 0.05 | - | 0.25 |
| A2 | 1.70 | - | 1.90 |
| b | 0.36 | - | 0.48 |
| b1 | 0.33 | - | 0.46 |
| c | 0.19 | - | 0.24 |
| c1 | 0.15 | - | 0.20 |
| D | 5.28 BSC | | |
| E | 8.00 BSC | | |
| E1 | 5.28 BSC | | |
| e | 1.27 BSC | | |
| L | 0.51 | - | 0.76 |
| L1 | 1.36 REF | | |
| L2 | 0.25 BSC | | |
| N | 8 | | |
| θ | 0° | - | 8° |
| θ 1 | 5° | - | 15° |
| θ 2 | 0-8° REF | | |

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. DIMENSIONING AND TOLERANCING PER ASME Y14.5M - 1994.
3. DIMENSION D DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 mm PER END. DIMENSION E1 DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 mm PER SIDE. D AND E1 DIMENSIONS ARE DETERMINED AT DATUM H.
4. THE PACKAGE TOP MAY BE SMALLER THAN THE PACKAGE BOTTOM. DIMENSIONS D AND E1 ARE DETERMINED AT THE OUTMOST EXTREMES OF THE PLASTIC BODY EXCLUSIVE OF MOLD FLASH, TIE BAR BURRS, GATE BURRS AND INTERLEAD FLASH, BUT INCLUSIVE OF ANY MISMATCH BETWEEN THE TOP AND BOTTOM OF THE PLASTIC BODY.
5. DATUMS A AND B TO BE DETERMINED AT DATUM H.
6. "N" IS THE MAXIMUM NUMBER OF TERMINAL POSITIONS FOR THE SPECIFIED PACKAGE LENGTH.
7. THE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 TO 0.25 mm FROM THE LEAD TIP.
8. DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10 mm TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. THE DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE LEAD FOOT.
9. THIS CHAMFER FEATURE IS OPTIONAL. IF IT IS NOT PRESENT, THEN A PIN 1 IDENTIFIER MUST BE LOCATED WITHIN THE INDEX AREA INDICATED.
10. LEAD COPLANARITY SHALL BE WITHIN 0.10 mm AS MEASURED FROM THE SEATING PLANE.

002-15548 Rev. **

**Figure 150 8-lead SOIC (5.28 × 5.28 × 2.16 mm) SOC008 package outline, 002-15548**









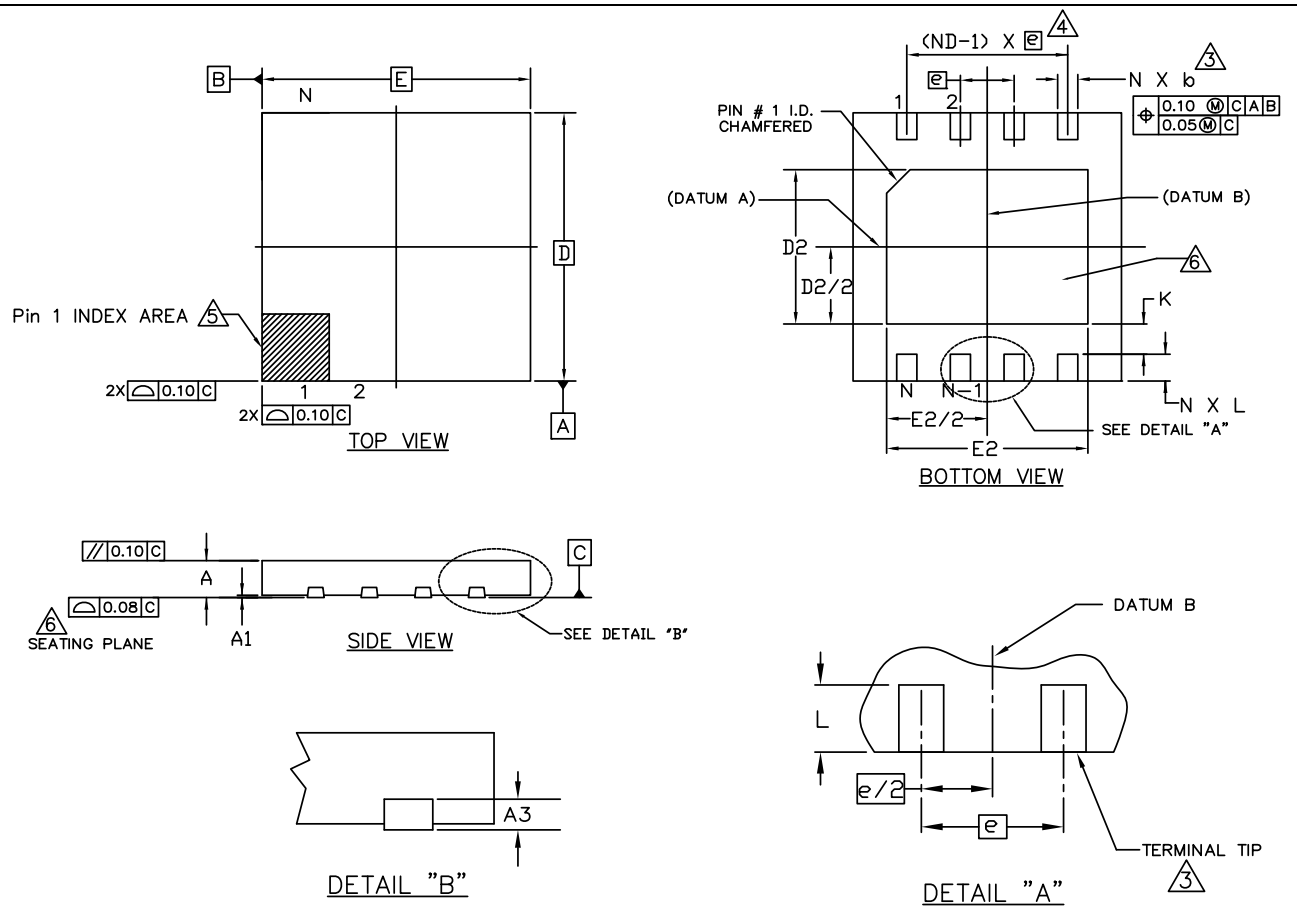

| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | 2.35 | - | 2.65 |
| A1 | 0.10 | - | 0.30 |
| A2 | 2.05 | - | 2.55 |
| b | 0.31 | - | 0.51 |
| b1 | 0.27 | - | 0.48 |
| c | 0.20 | - | 0.33 |
| c1 | 0.20 | - | 0.30 |
| D | 10.30 BSC | | |
| E | 10.30 BSC | | |
| E1 | 7.50 BSC | | |
| e | 1.27 BSC | | |
| L | 0.40 | - | 1.27 |
| L1 | 1.40 REF | | |
| L2 | 0.25 BSC | | |
| N | 16 | | |
| h | 0.25 | - | 0.75 |
| θ | 0° | - | 8° |
| θ1 | 5° | - | 15° |
| θ2 | 0° | - | - |

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. DIMENSIONING AND TOLERANCING PER ASME Y14.5M - 1994.
3. DIMENSION D DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 mm PER END. DIMENSION E1 DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 mm PER SIDE. D AND E1 DIMENSIONS ARE DETERMINED AT DATUM H.
4. THE PACKAGE TOP MAY BE SMALLER THAN THE PACKAGE BOTTOM. DIMENSIONS D AND E1 ARE DETERMINED AT THE OUTMOST EXTREMES OF THE PLASTIC BODY EXCLUSIVE OF MOLD FLASH, TIE BAR BURRS, GATE BURRS AND INTERLEAD FLASH, BUT INCLUSIVE OF ANY MISMATCH BETWEEN THE TOP AND BOTTOM OF THE PLASTIC BODY.
5. DATUMS A AND B TO BE DETERMINED AT DATUM H.
6. "N" IS THE MAXIMUM NUMBER OF TERMINAL POSITIONS FOR THE SPECIFIED PACKAGE LENGTH.
7. THE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 TO 0.25 mm FROM THE LEAD TIP.
8. DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.10 mm TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. THE DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE LEAD FOOT.
9. THIS CHAMFER FEATURE IS OPTIONAL. IF IT IS NOT PRESENT, THEN A PIN 1 IDENTIFIER MUST BE LOCATED WITHIN THE INDEX AREA INDICATED.
10. LEAD COPLANARITY SHALL BE WITHIN 0.10 mm AS MEASURED FROM THE SEATING PLANE.

002-15547 Rev. *A

**Figure 151 16-lead SOIC (10.30 × 7.50 × 2.65 mm) SO3016/SL3016/SS3016 package outline, 002-15547**



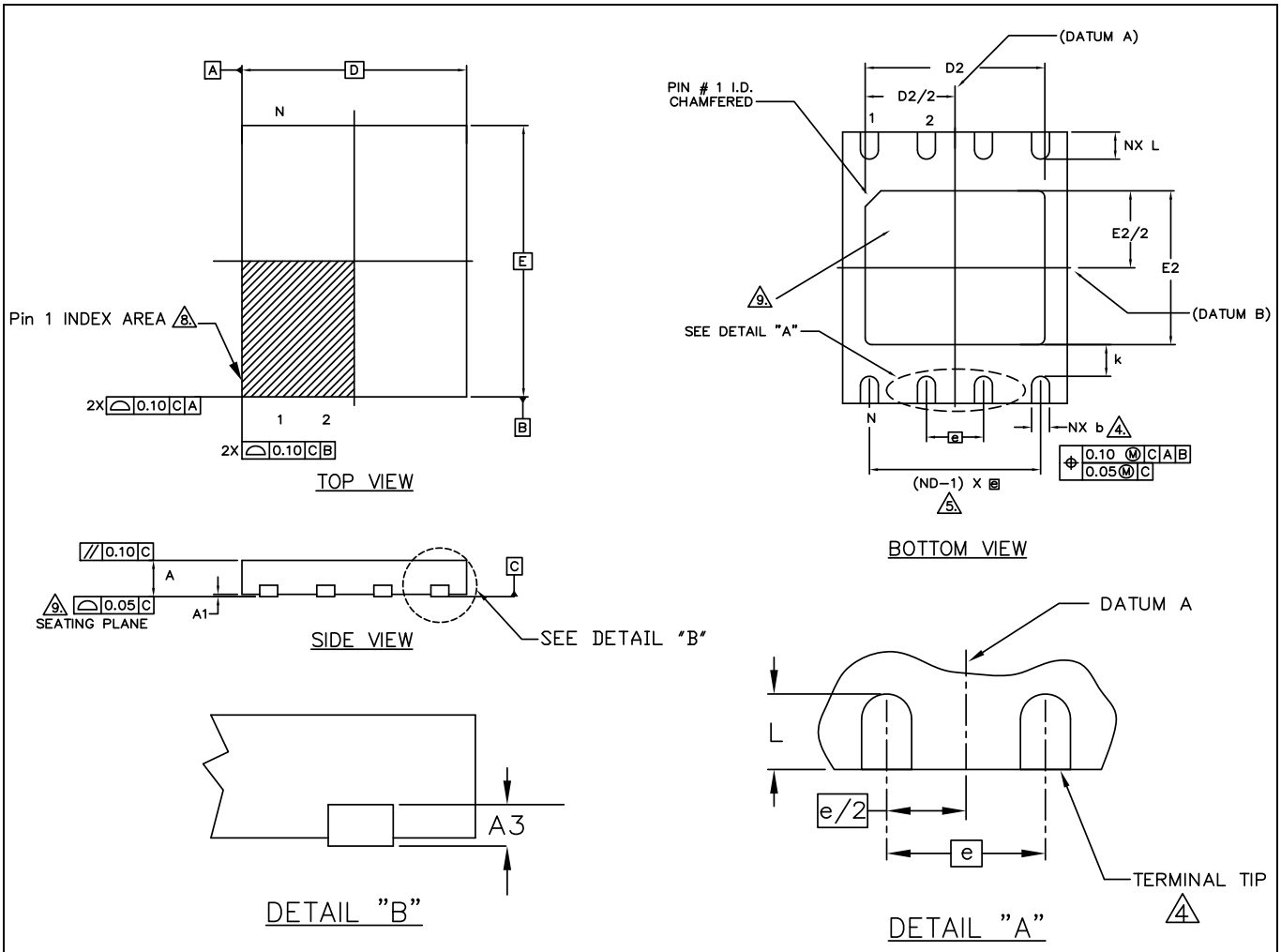

| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| e | | 0.80 BSC. | |
| N | | 8 | |
| ND | | 4 | |
| L | 0.35 | 0.40 | 0.45 |
| b | 0.25 | 0.30 | 0.35 |
| D2 | 2.20 | 2.30 | 2.40 |
| E2 | 2.90 | 3.00 | 3.10 |
| D | | 4.00 BSC | |
| E | | 4.00 BSC | |
| A | 0.50 | 0.55 | 0.60 |
| A1 | 0.00 | 0.035 | 0.05 |
| A3 | | 0.152 REF | |
| K | 0.20 | - | - |

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. N IS THE TOTAL NUMBER OF TERMINALS.
3. DIMENSION "b" APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.15 AND 0.30mm FROM TERMINAL TIP. IF THE TERMINAL HAS THE OPTIONAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION "b" SHOULD NOT BE MEASURED IN THAT RADIUS AREA.
4. ND REFERS TO THE NUMBER OF TERMINALS ON D SIDE.
5. PIN #1 ID ON TOP WILL BE LOCATED WITHIN THE INDICATED ZONE.
6. COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.
7. JEDEC SPECIFICATION NO. REF: N/A

002-16243 Rev. *A

**Figure 152 8-lead DFN ((4.0 × 4.0 × 0.6 mm) UNF008, 2.3 × 3.0 mm E-Pad (Sawn)) package outline, 002-16243**



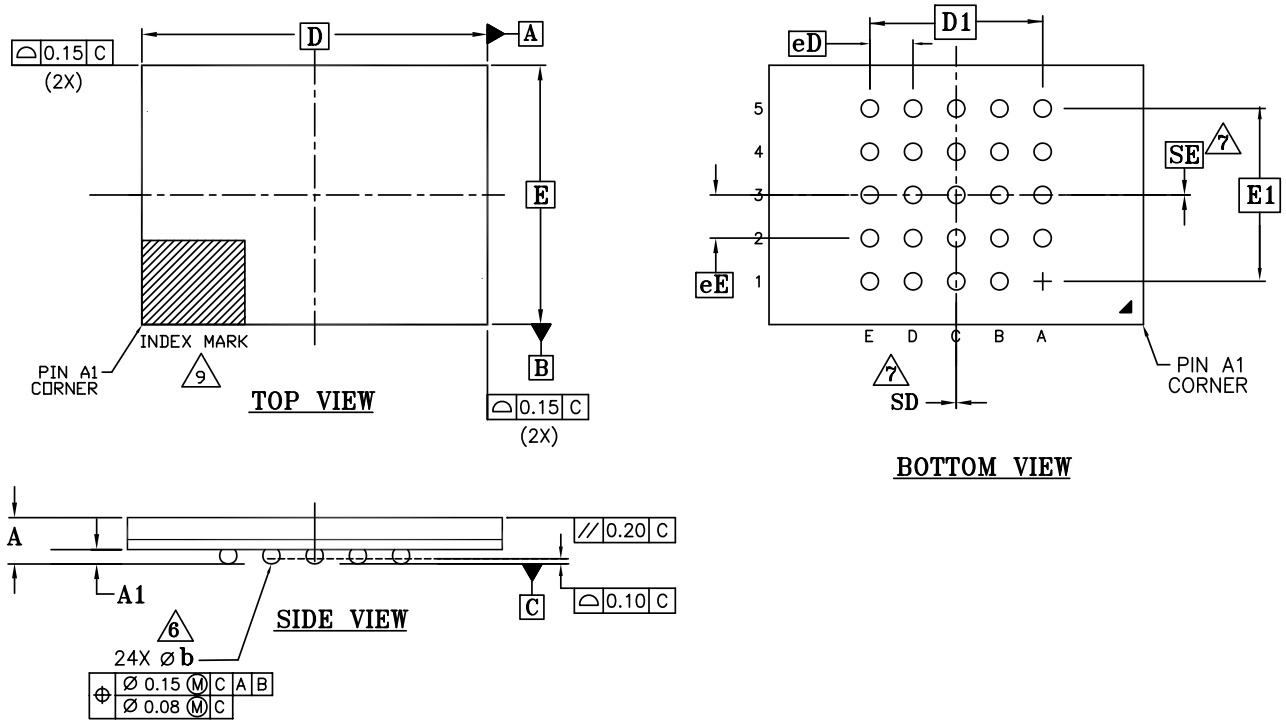

| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| e | | 1.27 BSC. | |
| N | | 8 | |
| ND | | 4 | |
| L | 0.55 | 0.60 | 0.65 |
| b | 0.35 | 0.40 | 0.45 |
| D2 | 3.90 | 4.00 | 4.10 |
| E2 | 3.30 | 3.40 | 3.50 |
| D | | 5.00 BSC | |
| E | | 6.00 BSC | |
| A | 0.70 | 0.75 | 0.80 |
| A1 | 0.00 | 0.02 | 0.05 |
| A3 | | 0.20 REF | |
| K | | 0.20 MIN. | |

NOTES:

1. DIMENSIONING AND TOLERANCING CONFORMS TO ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. N IS THE TOTAL NUMBER OF TERMINALS.
4. DIMENSION "b" APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.15 AND 0.30mm FROM TERMINAL TIP. IF THE TERMINAL HAS THE OPTIONAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION "b" SHOULD NOT BE MEASURED IN THAT RADIUS AREA.
5. ND REFERS TO THE NUMBER OF TERMINALS ON D SIDE.
6. MAX. PACKAGE WARPAGE IS 0.05mm.
7. MAXIMUM ALLOWABLE BURR IS 0.076mm IN ALL DIRECTIONS.
8. PIN #1 ID ON TOP WILL BE LOCATED WITHIN THE INDICATED ZONE.
9. BILATERAL COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.
10. A MAXIMUM 0.15mm PULL BACK (L1) MAY BE PRESENT.

002-18755 Rev. **

**Figure 153 8-lead DFN ((5.0 × 6.0 × 0.8 mm) WND008, 4.0 × 3.4 mm E-Pad (Sawn)) package outline, 002-18755**




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | - | - | 1.20 |
| A1 | 0.20 | - | - |
| D | 8.00 BSC | | |
| E | 6.00 BSC | | |
| D1 | 4.00 BSC | | |
| E1 | 4.00 BSC | | |
| MD | 5 | | |
| ME | 5 | | |
| N | 24 | | |
| ⌀ b | 0.35 | 0.40 | 0.45 |
| eE | 1.00 BSC | | |
| eD | 1.00 BSC | | |
| SD | 0.00 BSC | | |
| SE | 0.00 BSC | | |

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JEP95, SECTION 3, SPP-020.
4. e REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.
   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.
   N IS THE NUMBER OF POPULATED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.

△6 DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.

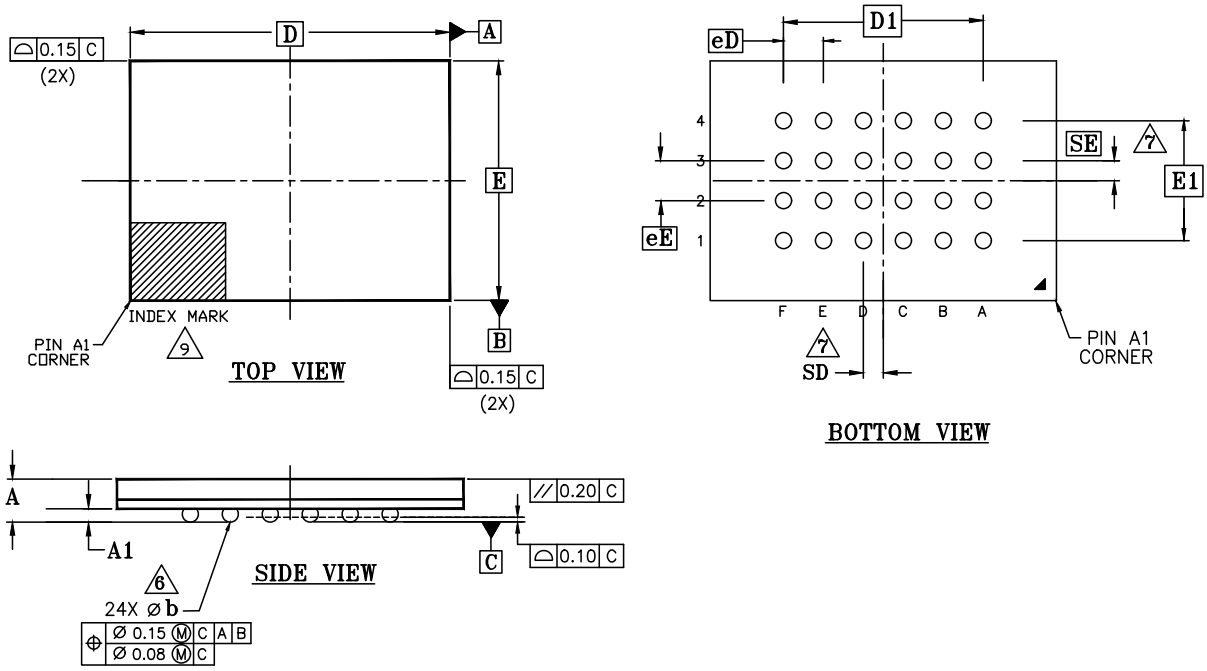
△7 "SD" AND "SE" ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.
WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" OR "SE" = 0.
WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" = eD/2 AND "SE" = eE/2.

8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.

△9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.

002-15534 Rev. **

**Figure 154 24-ball FBGA (8.0 × 6.0 × 1.2 mm) FAB024 package outline, 002-15534**




| SYMBOL | DIMENSIONS | | |
|---|---|---|---|
| | MIN. | NOM. | MAX. |
| A | - | - | 1.20 |
| A1 | 0.25 | - | - |
| D | 8.00 BSC | | |
| E | 6.00 BSC | | |
| D1 | 5.00 BSC | | |
| E1 | 3.00 BSC | | |
| MD | 6 | | |
| ME | 4 | | |
| N | 24 | | |
| Ø b | 0.35 | 0.40 | 0.45 |
| eE | 1.00 BSC | | |
| eD | 1.00 BSC | | |
| SD | 0.50 BSC | | |
| SE | 0.50 BSC | | |

NOTES:

1. DIMENSIONING AND TOLERANCING METHODS PER ASME Y14.5M-1994.
2. ALL DIMENSIONS ARE IN MILLIMETERS.
3. BALL POSITION DESIGNATION PER JEP95, SECTION 3, SPP-020.
4. e REPRESENTS THE SOLDER BALL GRID PITCH.
5. SYMBOL "MD" IS THE BALL MATRIX SIZE IN THE "D" DIRECTION.
   SYMBOL "ME" IS THE BALL MATRIX SIZE IN THE "E" DIRECTION.
   N IS THE NUMBER OF POPULATED SOLDER BALL POSITIONS FOR MATRIX SIZE MD X ME.
6. DIMENSION "b" IS MEASURED AT THE MAXIMUM BALL DIAMETER IN A PLANE PARALLEL TO DATUM C.
7. "SD" AND "SE" ARE MEASURED WITH RESPECT TO DATUMS A AND B AND DEFINE THE POSITION OF THE CENTER SOLDER BALL IN THE OUTER ROW.
   WHEN THERE IS AN ODD NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" OR "SE" = 0.
   WHEN THERE IS AN EVEN NUMBER OF SOLDER BALLS IN THE OUTER ROW, "SD" = eD/2 AND "SE" = eE/2.
8. "+" INDICATES THE THEORETICAL CENTER OF DEPOPULATED BALLS.
9. A1 CORNER TO BE IDENTIFIED BY CHAMFER, LASER OR INK MARK, METALLIZED MARK INDENTATION OR OTHER MEANS.

002-15535 Rev. *A

**Figure 155 24-ball FBGA (8.0 × 6.0 × 1.2 mm) FAC024 package outline, 002-15535**

# Revision history

| Document version | Date | Description of changes |
|---|---|---|
| ** | 2016-07-27 | Initial release. |
| *A | 2016-09-26 | Changed status from Advance to Preliminary.<br>Updated **Features**:<br>Added Automotive Grade related information.<br>Updated **Data integrity**:<br>Updated **Data retention**:<br>Updated **Table 38**.<br>Updated **Ordering information**:<br>Updated **Ordering part number**:<br>Updated details corresponding to “01” under “Model Number (Additional Ordering Options)”.<br>Added Automotive Grade related information.<br>Removed “Valid Combinations”.<br>Added **Valid combinations — Standard**.<br>Added **Valid combinations — Automotive grade / AEC-Q100**.<br>Updated **Physical diagrams**:<br>Updated SOIC 8-Lead, 208 mil Body Width (SOC008):<br>Replaced existing spec with 002-15548 **.<br>Updated USON 4 x 4 mm (UNF008):<br>Replaced existing spec with 002-16243 **.<br>Updated Ball Grid Array, 24-ball 6 x 8 mm (FAB024):<br>Replaced existing spec with 002-15534 **.<br>Updated Ball Grid Array, 24-ball 6 x 8 mm (FAC024):<br>Replaced existing spec with 002-15535 **.<br>Updated Other Resources:<br>Updated Link to Application Notes:<br>Updated hyperlinks. |
| *B | 2017-01-13 | Changed status from Preliminary to Final.<br>Updated **Commands**:<br>Updated **Command set summary**:<br>Updated **Command summary by function**:<br>Updated **Table 33**.<br>Updated **Register Access commands**:<br>Updated **Read Any Register (RDAR 65h)**:<br>Updated **Table 34**.<br>Updated **Data integrity**:<br>Updated **Data retention**:<br>Updated description below table.<br>Updated **Electrical specifications**:<br>Updated **Power-up and power-down**:<br>Updated **Table 47**.<br>Updated **DC characteristics**:<br>Updated **Table 48**.<br>Updated **Table 49**.<br>Updated **Table 50**.<br>Updated **Ordering information**:<br>Updated **Ordering part number**:<br>Added “04” under “Model Number”.<br>Completing Sunset Review. |

Revision history

| Document version | Date | Description of changes |
|---|---|---|
| *C | 2017-05-15 | Removed Extended Temperature Range Options (–40°C to +125°C) from datasheet.<br>Updated **Product overview**:<br>Updated **Migration notes**:<br>Updated **Features comparison**:<br>Updated **Table 1**.<br>Updated **Connection diagrams**:<br>Added **SOIC 16-lead**.<br>Updated **Commands**:<br>Updated **Command set summary**:<br>Updated **Command summary by function**:<br>Updated **Table 33**.<br>Updated **Register Access commands**:<br>Updated **Read Status Register 2 (RDSR2 07h)**:<br>Replaced “Status Register 1” with “Status Register 2” in all instances.<br>Updated **Software interface reference**:<br>Updated **JEDEC JESD216B serial flash discoverable parameters**:<br>Updated **JEDEC SFDP basic SPI flash parameter**:<br>Updated **Table 41**.<br>Updated **Ordering information**:<br>Updated **Ordering part number**:<br>Added WSON 5 × 6 mm option to “N” under “Package Type”.<br>Added 16-lead SOIC option to “M” under “Package Type”.<br>Updated **Valid combinations — Standard**:<br>Updated **Table 58**.<br>Updated **Valid combinations — Automotive grade / AEC-Q100**:<br>Updated **Table 59**.<br>Updated **Physical diagrams**:<br>Added “SOIC 16-Lead, 300 mil Body Width (SO3016)”.<br>Added “WSON 5x 6mm (WND008)”.<br>Updated Other Resources:<br>Updated Link to Cypress Flash Roadmap:<br>Updated hyperlinks.<br>Updated Link to Software:<br>Updated hyperlinks.<br>Updated Link to Application Notes:<br>Updated hyperlinks.<br>Updated to new template.<br>Completing Sunset Review. |
| *D | 2018-04-04 | Updated **Signal protocols**:<br>Updated **Command protocol**:<br>Updated **Command sequence examples**:<br>Updated **Figure 22**.<br>Updated **Figure 23**.<br>Updated **Commands**:<br>Updated **Read Memory Array commands**:<br>Updated **DDR Quad I/O Read (EDh, EEh)**:<br>Updated **Figure 74**.<br>Updated **Figure 75**.<br>Updated **Figure 76**.<br>Updated **Software interface reference**:<br>Updated **Device ID address map**:<br>Updated **Field definitions**:<br>Updated **Table 44**. |

Revision history

| Document version | Date | Description of changes |
|---|---|---|
| *D (cont.) | 2018-04-04 | Updated **Electrical specifications**:<br>Updated **Latchup characteristics**:<br>Updated **Table 45**.<br>Updated **DC characteristics**:<br>Updated **Table 48**.<br>Updated **Table 49**.<br>Updated **Table 50**.<br>Updated **Timing specifications**:<br>Updated **DDR AC characteristics**:<br>Removed “DDR Minimum Data Valid Window”.<br>Added **DDR data valid timing using DLP**. |
| *E | 2018-07-11 | Updated **Timing specifications**:<br>Updated **DDR AC characteristics**:<br>Updated the **DDR data valid timing using DLP**:<br>Updated description.<br>Updated **Ordering information**:<br>Updated **Ordering part number**:<br>Changed Low-halogen to Halogen free under “Package Materials”.<br>Added Note 97 and referred the same note in “Package Materials”.<br>Updated Other Resources:<br>Updated Glossary:<br>Updated Definition of MSb and LSb.<br>Completing Sunset Review. |
| *F | 2019-01-29 | Updated **Performance summary**:<br>Updated “Typical Current Consumption” table.<br>Updated **Software interface reference**:<br>Updated **JEDEC JESD216B serial flash discoverable parameters**:<br>Updated **JEDEC SFDP basic SPI flash parameter**:<br>Updated **Table 41**.<br>Updated **Electrical specifications**:<br>Updated **Power-up and power-down**:<br>Updated **Table 47**.<br>Updated to new template. |
| *G | 2022-07-21 | Updated Document Title to read as “S25FL064L, 64 Mb (8 MB) FL-L flash SPI multi-I/O, 3.0 V”.<br>Updated **Electrical specifications**:<br>Updated **Thermal resistance**:<br>Updated **Table 46**.<br>Removed “Other resources”.<br>Migrated to Infineon template. |
| *H | 2023-04-17 | Updated **Connection diagrams**:<br>Updated **SOIC 16-lead**:<br>Updated **Figure 1**.<br>Updated **BGA ball footprint**:<br>Updated **Figure 5**.<br>Updated **Physical diagrams**:<br>spec 002-15535 – Changed revision from ** to *A.<br>Updated to new template. |

**Trademarks**
All referenced product or service names and trademarks are the property of their respective owners.

**Edition 2023-04-17**
**Published by**

**Infineon Technologies AG**
**81726 Munich, Germany**

**© 2023 Infineon Technologies AG. All Rights Reserved.**

**Do you have a question about this document?**
**Email:**
**erratum@infineon.com**

**Document reference**
**002-12878 Rev. *H**

**IMPORTANT NOTICE**

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics (“Beschaffenheitsgarantie”).

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer’s compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer’s products and any use of the product of Infineon Technologies in customer’s applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer’s technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

**WARNINGS**

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies’ products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View S25FL064LABNFI041 on WIN SOURCE
- Cypress Semiconductor Corp Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management