

JEDEC STANDARD

EXpanded Serial Peripheral Interface (xSPI) for Non Volatile Memory Devices, Version 1.0

JESD251C

(Revision of JESD251B, September 2021)

May 2022

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION



NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by
©JEDEC Solid State Technology Association 2022
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2108

JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Contact JEDEC

Printed in the U.S.A.
All rights reserved

PLEASE!

DON'T VIOLATE
THE
LAW!

This document is copyrighted by JEDEC and may not be
reproduced without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

or refer to www.jedec.org under Standards-Documents/Copyright Information.

sobco

This page intentionally left blank

EXPANDED SERIAL PERIPHERAL INTERFACE (xSPI) FOR NON VOLATILE MEMORY DEVICES

Contents

	Page
Foreword	v
Introduction	v
1 Scope	1
2 Normative Reference	1
3 Terms and Definitions	2
3.1 Acronyms	4
3.2 Conventions	4
3.3 Keywords	5
3.4 Abbreviations	5
4 Key Features	6
4.1 General Features	6
4.2 Topology	6
4.3 Interface Features	6
5 xSPI Overview	7
5.1 xSPI Performance Enhancements	7
5.2 xSPI Topology and Signal Descriptions	7
5.3 xSPI Protocol Modes	8
5.4 xSPI Signal Protocols	9
5.5 Legacy SPI Compatibility	12
6 xSPI Transaction Descriptions	12
6.1 Target Selection	12
6.2 Clock, DS, and Information Transfer	12
6.3 Command Phase	13
6.4 Command Modifier Phase	13
6.5 Initial Access Latency Phase	14
6.6 Data Phase	14
6.7 Required versus Optional Commands	14
6.8 Maximum Frequency	15
6.9 Command Summary Tables	15
6.9.1 1S-1S-1S Mode	15
6.9.2 8D-8D-8D Profile 1.0	16
6.9.3 8D-8D-8D Profile 2.0	18
6.9.4 Command Summary Notes	20
6.9.5 Transaction Formats by Protocol Mode	21
6.10 Command Descriptions	30
6.10.1 Identification Commands	30
6.10.2 Profile 1.0 Commands	30

Contents (cont'd)

6.10.3	Profile 2.0 Commands	40
6.10.4	Profile 2.0 Commands with Extended Command Modifier.....	46
7	xSPI Mechanical	50
7.1	Introduction.....	50
7.2	Signal Descriptions	50
7.3	Fine-Pitch Ball Grid Array (FBGA) 24-Ball Footprint	53
7.3.1	Signal Assignments for 24-Ball FBGA Footprint.....	54
7.3.2	Package Physical Diagrams	54
7.4	0.5 mm-Pitch WLCSP (Wafer Level Chip Scale Package)	55
7.4.1	Ball Locations for a Minimally Sized Die	55
7.4.2	Ball Locations for a Large Die.....	56
7.4.3	Example for Small Die with “Core” 4x5 Ball Matrix.....	57
7.5	49-Ball BGA (Ball Grid Array) Package for Dual-Octal (x16 IO) xSPI.....	58
7.5.1	Ballout Descriptions.....	58
7.5.2	Signal Assignments for 49-Ball BGA Footprint.....	59
7.5.3	49-Ball BGA Package Layout	60
7.5.4	Package Physical Diagrams	60
8	xSPI Electrical	61
8.1	Introduction.....	61
8.2	xSPI Signals.....	61
8.3	Power Supply	62
8.4	Bus Capacitance and Programmable Output Driver Strength.....	62
8.4.1	Device Capacitance.....	62
8.4.2	xSPI Reference Load	63
8.4.3	Driver Types Definition.....	64
8.5	xSPI Input and Output Voltage Levels	66
8.6	Leakage Current.....	66
8.6.1	Leakage Current at 1.2 V and 1.8 V	66
8.6.2	Leakage Current at 3 V	66
8.7	AC Timing Specifications.....	67
8.7.1	Controller Output to Target Input Timing	67
8.7.2	Target Output to Controller Input Timing	68
8.7.3	xSPI CK and DS to CS# Signal Timing	70
Annex A	-(Informative) Differences Between Revisions.....	72

Contents (cont'd)

Tables

	Pages
Table 1 — 1S-1S-1S Bit Positions for 4 (and 3) Byte Addressing	10
Table 2 — 8D-8D-8D Bit Positions.....	11
Table 3 — Commands Used in 1S-1S-1S Protocol Mode (Required Commands)	15
Table 4 — Profile 1.0 Commands Used in 8D-8D-8D Protocol Mode (Required Commands).....	16
Table 5 — Profile 1.0 Commands Used in 8D-8D-8D Protocol Mode (Optional Commands)	17
Table 6 — Profile 2.0 Commands Used in 8D-8D-8D Protocol Mode	19
Table 7 — Read Wrapped Setup Configuration Data Value Options.....	33
Table 8 — Profile 2.0 Command Modifier Formats.....	41
Table 9 — Command and Modifier Bit Description	42
Table 10 — Profile 2.0 Command with Extended Modifier Format	46
Table 11 — Command and Modifier Bit Descriptions – HB1 Format.....	47
Table 12 — Commands with Extended Command Modifier Used in 8D-8D-8D Protocol Profile 2.0 Mode	48
Table 13 — xSPI Signals (Normative)	50
Table 14 — xSPI Optional Signals (Informative).....	51
Table 15 — Signal Assignments for 24-Ball FBGA Footprint.....	54
Table 16 — 49-Ball BGA Ballout Descriptions	58
Table 17 — Signal Assignments for 49-Ball BGA Footprint.....	59
Table 18 — Device Capacitance.....	62
Table 19 — I/O Driver Strength Types	64
Table 20 — Leakage Current at 1.2 V and 1.8 V	66
Table 21 — Leakage Current at 3 V	66
Table 22 — Clock Input Threshold Levels.....	67
Table 23 — xSPI Device Input Timing	68
Table 24 — xSPI Device Output Timing.....	69
Table 25 — Relative Timing of the CS#, DS, and CK Signals	71

Figures

	Pages
Figure 1 — xSPI System Topology	8
Figure 2 — Command Categories	16
Figure 3 — Instruction Categories.....	18
Figure 4 — 1S-1S-1S Format 0.A Command.....	21
Figure 5 — 1S-1S-1S Format 0.B Command and Read Data.....	22
Figure 6 — 1S-1S-1S Format 0.D Command, 4-byte Address, and Read Data	22
Figure 7 — 1S-1S-1S Format 0.F Command, 4-Byte Address, ‘x’ Latency Cycles, and Read Data.....	23
Figure 8 — 1S-1S-1S Format 0.G Command and Write Data	23
Figure 9 — 1S-1S-1S Format 0.I Command and 4-Byte Address.....	24
Figure 10 — 1S-1S-1S Format 0.K Command, 4-Byte Address, and Write Data.....	24

Contents (cont'd)

Figure 11 — Profile 1.0 8D-8D-8D Format 1.A: Command and Command Extension.....	25
Figure 12 — Profile 1.0 8D-8D-8D Format 1.B: Command and Command Extension, Address, ‘x’ Latency Cycles, and Read Data	25
Figure 13 — Profile 1.0 8D-8D-8D Format 1.C: Command and Command Extension, Address.....	26
Figure 14 — Profile 1.0 8D-8D-8D Format 1.D: Command and Command Extension, Address, and Write Data	26
Figure 15 — Profile 2.0 8D-8D-8D Format 2.A: Command, 5-byte Address Modifier, ‘x’ Latency Cycles, and Read Data	27
Figure 16 — Profile 2.0 8D-8D-8D Format 2.A: Command, 5-byte Address Modifier, ‘2x’ Latency Cycles, and Read Data	27
Figure 17 — Profile 2.0 8D-8D-8D Format 2.B: Command, 5-byte Address, Initial Access Latency, and Write Data	28
Figure 18 — Profile 2.0 8D-8D-8D Format 2.B: Command, 5-byte Address, 2x Initial Access Latency, and Write Data	28
Figure 19 — Profile 2.0 8D-8D-8D Format 2.B: Command, 5-byte Address, Zero Latency, and Write Data	29
Figure 20 — 3-byte Address Format	30
Figure 21 — 4-byte Address Format	30
Figure 22 — xSPI Flash Memory Map.....	31
Figure 23 — FBGA 24-Ball Footprint.....	53
Figure 24 — Ball Location Diagram (Top View – Balls Down).....	55
Figure 25 — Ball Location Diagram Large Die (Top View – Balls Down).....	56
Figure 26 — Example for Small Die with “Core” 4x5 Ball Matrix	57
Figure 27 — 49-Ball BGA Ball Location Diagram (Top View – Balls Down)	60
Figure 28 — xSPI Controller-Target Connections	61
Figure 29 — xSPI Target Internal Power Diagram	62
Figure 30 — xSPI Reference Load for 200 MHz DDR Operation.....	63
Figure 31 — Target I/O	65
Figure 32 — Controller I/O	65
Figure 33 — xSPI Target Data Input Timing	67
Figure 34 — xSPI Target Data Output Timing.....	68
Figure 35 — CK to CS# Signal Timing.....	70
Figure 36 — DS to CS# Signal Timing	70

Foreword

This standard is intended for use by SoC, ASIC, ASSP, and FPGA developers or vendors interested in incorporating a controller interface having a low signal count and high data transfer bandwidth with access to multiple sources of target devices compliant with the interface. It is also, intended for use by peripheral developers or vendors interested in providing target devices compliant with the standard, including non-volatile memories, volatile memories, graphics peripherals, networking peripherals, FPGAs, sensors, etc.

This document was prepared by the JC-42.4_3 Serial Flash task group authorized by the JC-42.4 Non-Volatile Memory subcommittee.

Introduction

This standard defines a low signal count controller-target interface for high speed byte or word serial communication with peripheral devices.

The standard defines commands for general purpose read and write of any type of peripheral device and specific commands for non-volatile memory device functions. The standard includes limited hardware and software backward compatibility with Serial Peripheral Interface (SPI) controllers widely used in the electronics industry.

Electrical DC and AC characteristics are defined for operation with a 3 V, 1.8 V, or 1.2 V power supply.

Signaling protocols are defined for command and data transfers widths of 1, or 8 bits using either Single Data Rate (SDR) or Double Data Rate (DDR) transfers at up to 200 MHz with up to 400 MT/s.

Also, defined is a Fine-pitch Ball Grid Array (FBGA) footprint with package outline options compliant with JEDEC Publication No. 95 (JEP95), MO-234.

The purpose of this standard is to define a minimum set of requirements for JEDEC standard compliance. There are a number of existing devices that may contain legacy and/or unique manufacturer's functions in addition to the minimum JEDEC requirements. Some of these functions are defined in this document as optional operations.

This page intentionally left blank

EXPANDED SERIAL PERIPHERAL INTERFACE (XSPI) FOR NON VOLATILE MEMORY DEVICES

From JEDEC Board Ballot JCB-21-68, formulated under the cognizance of the JC-42.4 Subcommittee on Non-Volatile Memory Devices, item 1775.74.

1 Scope

This standard specifies the eXpanded Serial Peripheral Interface (xSPI) for Non Volatile Memory Devices, which provides high data throughput, low signal count, and limited backward compatibility with legacy Serial Peripheral Interface (SPI) devices. It is primarily for use in computing, automotive, Internet Of Things (IOT), embedded systems and mobile systems, between host processing and peripheral devices. The xSPI electrical interface can deliver up to 400 MBytes per second raw data throughput.

2 Normative Reference

The following normative documents contain provisions that through reference in this text, constitutes provisions of this standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated. For undated references, the latest edition of the normative document referred to applies.

JEDEC Manual, JM7.01, *Style Manual for Standards and Other Publications of JEDEC*.

JEDEC Standard, JESD88E, *Dictionary of Terms for Solid-State Technology*.

JEDEC Standard, JESD99C, *Terms, Definitions, and Letter Symbols for Microelectronic Devices*.

JEDEC Standard, JESD100B.01, *Terms, Definitions, and Letter Symbols For Microcomputers, Microprocessors, and Memory Integrated Circuits*.

JEDEC Publication No. 95, MO-234, *Low Profile Rectangular Ball Grid Array Family, 6.00 x 8.00 x 1.20 or 1.00 mm, 24 ball package*.

JEDEC Standard, JESD8-26, *1.2 V High-Speed LVCMOS (HS_LVCMOS) Interface*.

JEDEC Standard, JESD8-31, *1.8 V High-Speed LVCMOS (HS_LVCMOS) Interface*.

JEDEC Standard, JESD8-xx, *3 V High-Speed LVCMOS (HS_LVCMOS) Interface. (In progress)*

JEDEC Standard, JESD216, *Serial Flash Discoverable Parameters (SFDP)*.

JEDEC Standard, JESD252, *Serial Flash Reset Signaling Protocol*.

JEDEC Standard, JESD84-B51, *eMMC HS400 mode*.

3 Terms and Definitions

The current version of JESD88, Dictionary of Terms for Solid-State Technology, is the governing document for terms used in this standard. The following terms and definitions are provided for ease of reference:

byte (B): (1) A binary character string operated upon as a unit and usually shorter than a computer word. (Ref. ANSI X3.172.)

NOTE A byte is usually eight bits. Within this standard, it is an 8-bit data value with the most significant bit as bit 7 and the least significant bit as bit 0.

Controller: An xSPI device that drives chip select signals to enable one other xSPI device to receive or transmit data on the interface at a given time.

Data Strobe (DS): Target to controller strobe signal to capture read data sent by the target. Target to controller signal to indicate additional initial access latency needed by the target. Controller to target signal to indicate masking of individual bytes within write data words. DS initial latency indication and write data mask behavior is used by a limited subset of xSPI commands.

Double Data Rate (DDR): Information (e.g., command, modifier, address, or data) is transferred on each edge of a related clock.

double word: A character string or binary element string that, in a given system, has twice the length of a word.

NOTE For the purpose of this document “**Dword**” is a 32-bit data value with most significant bit labeled as bit 31 and least significant bit as bit 0.

Single Data Rate (SDR): Information (e.g., command, modifier, address, or data) is transferred on one edge of a related clock.

flash EEPROM (FEEPROM): An EEPROM in which clearing can be performed only on blocks or on the entire array.

NOTE For the purpose of this document “**Flash**” is a type of block erased, multiple time programmable, nonvolatile memory.

gigabyte (GB): commonly used as a prefix to units of semiconductor storage capacity and meaning 230 [1 073 741 824] bytes

host: The computer system, test system, or other device that writes data to, and reads data from, a memory device. Within this standard, it is an xSPI controller: An entity or a device with the characteristics of a primary computing device that includes one or more xSPI controllers. The xSPI controller drives out all chip select (CS) signals to xSPI target devices to select one target device at a time to interact with the controller. The controller provides transfer type and address information to target devices.

HS400: High Speed DDR interface timing mode of eMMC (JESD84), with up to 400 MT/s at 200 MHz

LOW, HIGH: Binary interface states with defined assignment to a voltage level. LOW is $\leq V_{IL}$ or V_{OL} , HIGH is $\geq V_{IH}$ or V_{OH} .

3 Terms and Definitions (cont'd)

Kilobyte (Kbyte, KB): 1024 or 2^{10} bytes.

mega (M) (as a prefix to units of semiconductor storage capacity): A multiplier equal to 1 048 576 (2^{20} or K^2 , where $K = 1024$).

Megabyte (MB): 1,048,576 or 2^{20} bytes.

open-drain output: An open-circuit output whose internal connection is to the drain of a field effect transistor.

NOTE Within this standard, it is an output operation mode where an external resistor or current source is used to pull the output level to HIGH, and an internal transistor pulls the output level to LOW. Multiple outputs of this type from different devices may be tied together to create a wire-NOR signal.

peripheral equipment: In a data processing system, any equipment, distinct from the central processing unit, that may provide the system with outside communication or additional facilities. (Ref. ANSI X3.172.)

NOTE For the purpose of this document “**peripheral**” refers to an xSPI target, see definition.

Power On Reset (POR): A hardware reset process that leaves the device in a defined state, triggered by the rise of power supply voltage levels from near ground to the operating range level.

push-pull output: Two open-circuit outputs operating in complementary fashion so that as the resistance of one increases, the resistance of the other decreases. Within this standard, an output operation mode where a complementary pair of transistors is used to alternately push the interface level to HIGH or pull it to LOW.

quad word: A character string or binary element string that, in a given system, has four times the length of a word. Within this standard: a 64-bit data value with most significant bit labeled as bit 63 and least significant bit as 0.

Target: An xSPI device that receives a chip select signal to enable receive or transmit of data on the interface at a given time.

word: A character string or a binary element string that it is convenient to consider as an entity. (Adapted from ANSI X3.172.)

NOTE For memories, it is common practice to use the term “word” generically for any number of bits that occupy a single address location. Within this standard: a 16-bit data value with most significant bit labeled as bit 15 and least significant bit as bit 0.

xSPI target: An entity or a device with the characteristics of peripheral equipment that is selected by an xSPI controller.

NOTE The xSPI target is made active by its chip select (CS#) input to interact with the controller. The target receives transfer type and address information in order to identify which data to receive from or transmit to the xSPI controller.

3.1 Acronyms

DDR	Double Data Rate. Data is transferred on each edge of a related clock.
GB	Gigabyte: 1,073,741,824 or 2^{30} bytes.
I/O or IO	Input or Output – A signal port that may receive or drive voltage levels to communicate with other devices.
KB	Kilobyte: 1024 or 2^{10} bytes.
MB	Megabyte: 1,048,576 or 2^{20} bytes
MB/s	Megabytes per second
MSb, LSb	Most Significant Bit or Least Significant Bit
MT/s	Mega Transfers per second
NVM	Non-Volatile Memory
UI	Unit Interval; It is one bit nominal time. For example, UI=5 ns at 200 MHz SDR, UI = 2.5 ns for 200 MHz DDR.
xSPI	eXtended Serial Peripheral Interface

3.2 Conventions

This standard follows some conventions used in other JEDEC documents.

A binary number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 and 1 immediately followed by a lower-case b (e.g., 0101b). Spaces may be included in binary number representations to increase readability or delineate field boundaries (e.g., 0 0101 1010b).

A hexadecimal number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 through 9 and/or the upper-case English letters A through F immediately followed by a lower-case h (e.g., FA23h). Spaces may be included in hexadecimal number representations to increase readability or delineate field boundaries (e.g., B FD8C FA23h).

A decimal number is represented in this standard by any sequence of digits consisting of only the Western-Arabic numerals 0 through 9 not immediately followed by a lower-case b or lower-case h (e.g., 25).

A range of numeric values is represented in this standard in the form "a to z", where a is the first value included in the range, all values between a and z are included in the range, and z is the last value included in the range (e.g., the representation "0h to 3h" includes the values 0h, 1h, 2h, and 3h).

Other conventions used in this document:

Signals that are considered active when LOW have signal names that end with the hash / pound special character (i.e., have a suffix of #).

3.3 Keywords

Several keywords are used to differentiate levels of requirements and options, as follow:

Can - A keyword used for statements of possibility and capability, whether material, physical, or causal (can equals is able to).

Expected - A keyword used to describe the behavior of the hardware or software in the design models assumed by this standard. Other hardware and software design models may also be implemented.

Ignored - A keyword that describes bits, bytes, or fields whose values are not checked by the recipient.

Mandatory - A keyword that indicates items required to be implemented as defined by this standard.

May - A keyword that indicates a course of action permissible within the limits of the standard (may equals is permitted).

Must - The use of the word must is deprecated and shall not be used when stating mandatory requirements; must is used only to describe unavoidable situations.

Obsolete - A keyword indicating that an item was defined in prior standards but has been removed from this standard.

Optional - A keyword that describes features which are not required to be implemented by this standard. However, if any optional feature defined by the standard is implemented, it shall be implemented as defined by the standard.

Preferred - A keyword used to identify a feature option which is the preferred option to implement for compliance with anticipated future revisions of this standard.

Reserved - A keyword used to describe objects—bits, bytes, and fields—or the code values assigned to these objects in cases where either the object or the code value is set aside for future standardization.

Usage and interpretation may be specified by future extensions to this or other standards. A reserved object shall be zeroed or, upon development of a future standard, set to a value specified by such a standard. The recipient of a reserved object shall not check its value. The recipient of a defined object shall check its value and reject reserved code values.

Shall - A keyword that indicates a mandatory requirement strictly to be followed in order to conform to the standard and from which no deviation is permitted (“shall” equals “is required to”). Designers are required to implement all such mandatory requirements to assure interoperability with other products conforming to this standard.

Should - A keyword used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (“should” equals “is recommended that”).

Will - The use of the word will is deprecated and shall not be used when stating mandatory requirements; will is only used in statements of fact.

3.4 Abbreviations

etc.: And so forth (Latin: et cetera)

e.g.: For example (Latin: exempli gratia)

i.e.: That is (Latin: id est)

4 Key Features

The following is a summary of the eXtended Serial Peripheral Interface (xSPI) features:

4.1 General Features

- Speed grades with data transfer rates up to
 - 400 MT/s (200 MHz Clock)
 - 333 MT/s (167 MHz Clock)
 - 266 MT/s (133 MHz Clock)
 - 200 MT/s (100 MHz Clock)

4.2 Topology

- Single controller, multiple targets, per interface port
 - One chip select per target

4.3 Interface Features

- Input/Output Power Supplies Options:
 - 1.2 V, 1.8 V, or 3 V
- Signal Count Options:
 - Eleven-signal target interface (Clock, CS#, 8-bit Data bus, Data Strobe)
 - Twelve-signal target interface (Differential Clock, CS#, 8-bit Data bus, Data Strobe)
 - Interface may operate in 1S-1S-1S IO mode without data strobe at ≤ 50 MHz SDR
- IO Protocol Modes:
 - Transaction phases = Command-Modifier-Data
 - Transfer bit width options in each phase = $W = 1$, or 8
 - Data rate options in each phase = $R = S$ for SDR or D for DDR
 - 1S-1S-1S (24 bit addressing) for power-up and configuration
 - 8D-8D-8D (32 bit addressing in Profile 1.0 or 45 bit addressing in profile 2.0)
- Target Reset Options:
 - Power-on Reset (POR) -- mandatory
 - In-band Reset -- preferred
 - Separate Reset Signal -- optional
- Power Management Option
 - Deep Power Down (DPD) enter and exit commands

5 xSPI Overview

5.1 xSPI Performance Enhancements

Serial interface performance can be enhanced by increasing the basic clock rate, adding source synchronous clocking, using Double Data Rate (DDR), or by adding parallel data (IO) lines. xSPI allows all four methods.

Higher clock rates with accurate information transfer are possible by using a separate clock signal for each direction of information transfer. This is known as source synchronous clocking; providing a timing reference (clock / strobe) from the source of information, with minimum skew relative to the related information (data). This enables accurate timing of information capture at the destination device as the period of valid information decreases at higher clock rates.

DDR can be used to double performance by relating information transfers to both rising and falling edges of the timing reference.

Up to 8 parallel data (IO) lines are used in xSPI to further increase system performance by transferring more parallel bits in each clock cycle.

5.2 xSPI Topology and Signal Descriptions

xSPI has a single controller, multiple target bus topology with source synchronous clocking.

Each target device is selected by a separate LOW active Chip Select (CS#) signal from the controller device.

A clock (CK) signal from the controller to all target devices provides a timing reference for interface operations. The clock is the timing reference for information transfer on the IO signals from the controller to the selected target. The clock is also a timing reference for interface operations within the controller and target interface logic, such as the counting of access latency time. The clock may optionally be provided by the controller as a differential clock using two signals (CK and CK#). The remaining text and block diagrams in this standard may only describe one clock (CK) signal for simplicity but, where ever CK is described or illustrated, a differential pair of CK, CK# may be used. Some xSPI compliant target devices require a differential clock. A differential clock reduces clock duty cycle distortion and effectively doubles the clock slew rate, providing improved timing margins for target devices using a differential clock.

The IO signals provide a bidirectional, variable width path, for information transfer between the controller and selected target. Different targets on the same bus may use a different IO width for information transfer.

The Data Strobe (DS) signal provides a timing reference for information transfer on the IO signals from the target to the controller. DS is an output signal from the xSPI target device that is aligned with the edges of output data and is used by the controller interface to optimize high frequency read data capture. DS may optionally be used as a signal from the target to the controller, to indicate the duration of initial access latency or, as a signal from the controller to the target, to indicate write data masking.

5.2 xSPI Topology and Signal Descriptions (cont'd)

Controller and target devices may have separate power supplies and ground references for internal (core) logic functions versus the xSPI interface receiver inputs and output drivers. The ability to separate the core and IO power supplies may help to reduce IO noise injection into the core power supply. However, devices are not required to separate their core and IO power supplies. These power supplies may be tied together in parallel at the device terminals. The core power supply is labeled V_{DD} and the interface power supply is labeled V_{DDQ} . The core ground reference is labeled V_{SS} and the interface ground reference is labeled V_{SSQ} .

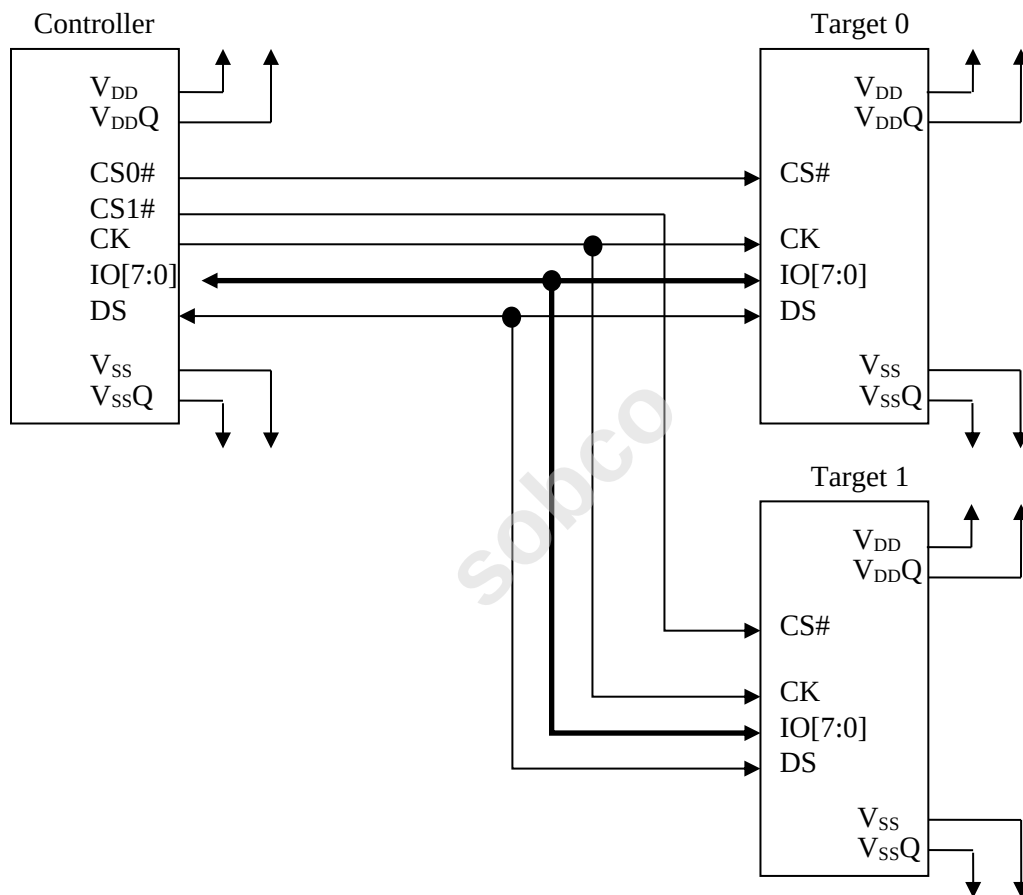


Figure 1 — xSPI System Topology

5.3 xSPI Protocol Modes

No information provided at time of publication.

5.4 xSPI Signal Protocols

During the time that CS# is active (LOW) the clock signal (CK) is toggled while command information is first transferred on the data (IO) signals from the controller to the target. The clock continues to toggle during any period required for information access in the target. The clock continues to toggle during the transfer of read data from the target to the controller or write data from the controller to the target. When the controller has transferred the desired amount of data, the controller drives the CS# inactive (HIGH). The period during which CS# is active is called a transaction on the bus.

While CS# is inactive, the CK is not required to toggle. CK may stop toggling when LOW as a means of lowering power consumption or inserting delay within a transaction for flow control by the controller. CK must always complete at least one rising edge and one falling edge before stopping at LOW. This requirement for a minimum of one rising and falling edge in turn requires that DDR transfers always occur in two transfer increments, e.g., two bytes (word) for 8-bit wide transfers. SDR transfers must occur in one byte increments. Some target devices may have limitations on the maximum time that CK may stop (remain LOW) or, on the maximum time CK may be HIGH (i.e., a minimum clock frequency during a transaction); these are target device specific system design consideration.

There are up to four phases of activity within each transaction:

- Command transfer from controller to target
- Command Modifier transfer from controller to target
- Initial Access Latency (also used for IO signal direction turn around in a read transaction)
- Data transfer (target to controller in a read transaction or controller to target in a write transaction)

The command transfer occurs at the beginning of every transaction. The command modifier, initial access latency, and data transfer phases are optional and their presence depends on the protocol mode or command transferred.

The number of parallel IO signals used during the command modifier and data phases depends on the current protocol mode or command transferred. The initial access latency phase does not use the IO signals for information transfer. The protocol mode options are described by the data rate and the IO width (number of IO signals) used during the command, command modifier, and data phases in the following format:

WR-WR-WR where:

- The first WR is the command bit width and rate
- The second WR is the command modifier bit width and rate
- The third WR is the data bit width and rate

The bit width value may be 1, or 8. R has a value of S for SDR, or D for DDR. SDR has the same transfer value during the rising and falling edge of a clock cycle. DDR may have different transfer values during the rising and falling edges of each clock.

Examples:

- 1S-1S-1S means that the command is 1-bit wide SDR, the command modifier is 1-bit wide SDR, and the data is one bit wide SDR.
- 8D-8D-8D means that the command, command modifier, and data transfers are always 8 bits wide DDR.

5.4 xSPI Signal Protocols (cont'd)

Protocol Modes defined for the xSPI interface:

- 1S-1S-1S
 - One IO signal used during command transfer, command modifier transfer, and data transfer. All phases are SDR.
- 8D-8D-8D
 - Eight IO signals used during command transfer, command modifier transfer, and data transfer. All phases are DDR.

A target device must implement both of the above protocol modes.

The 1S-1S-1S mode is the preferred default protocol following Power-On-Reset (POR) but, target devices may be configured to reset into the Octal mode. Target devices may implement other modes of operation. The supported modes are indicated in the SFDP.

For example, 8D-8D-8D mode can be made the default mode if so desired. The controller must determine the default protocol mode of each target after POR. This may be done through prior knowledge of the system design. The controller may later reconfigure a target to use other target supported modes. In some devices switching between modes is done through issuing a command or a series of commands while in other devices that change is done through a modification of bits in control registers. The modes supported by a target and the method for switching between modes are indicated in the SFDP.

A protocol mode phase using single bit transfer uses IO[0] to transfer information from controller to target and IO[1] to transfer information from target to controller. On each IO, information is placed on the IO line in Most Significant bit (MSb) to Least Significant bit (LSb) order within each byte. Sequential command modifier bytes are transferred in highest order to lowest order sequence. Sequential data bytes are transferred in lowest address to highest address order.

Table 1 — 1S-1S-1S Bit Positions for 4 (and 3) Byte Addressing

IO	Command Bits	Command Modifier Bits (Address)	Latency	Data Byte 0	Data Byte 1
0	7, 6, 5, 4, 3, 2, 1, 0	31 (23), 30 (22), ... 1, 0	X ...	X ...	X ...
1	X ...	X ...	X ...	7, 6, 5, 4, 3, 2, 1, 0	7, 6, 5, 4, 3, 2, 1, 0
2	X ...	X ...	X ...	X ...	X ...
3	X ...	X ...	X ...	X ...	X ...
4	X ...	X ...	X ...	X ...	X ...
5	X ...	X ...	X ...	X ...	X ...
6	X ...	X ...	X ...	X ...	X ...
7	X ...	X ...	X ...	X ...	X ...

See 6.9.5, Transaction Formats by Protocol Mode for waveform examples.

5.4 xSPI Signal Protocols (cont'd)

A protocol mode phase using eight IO signals uses IO[7:0]. The LSb of each byte is placed on IO[0] with each higher order bit on the successively higher numbered IO signals. Sequential command modifier bytes are transferred in highest order to lowest order sequence. Sequential data bytes in SDR are transferred in lowest address to highest address order. Sequential data bytes in DDR are transferred only in byte pairs (words) where the byte order depends on the order in which the bytes are written or programmed in that protocol mode. Sequential data bytes are transferred in lowest address to highest address order.

Table 2 — 8D-8D-8D Bit Positions

IO	Command Bits	Command Modifier Bits (Command Ext. and Address)					Latency	Data Word 0		Data Word 1	
0	0	0	24	16	...	0	X ...	0	0	0	0
1	1	1	25	17	...	1	X ...	1	1	1	1
2	2	2	26	18	...	2	X ...	2	2	2	2
3	3	3	27	19	...	3	X ...	3	3	3	3
4	4	4	28	20	...	4	X ...	4	4	4	4
5	5	5	29	21	...	5	X ...	5	5	5	5
6	6	6	30	22	...	6	X ...	6	6	6	6
7	7	7	31	23	...	7	X ...	7	7	7	7

IO signals not in use in a particular phase are undefined may or may not be driven by the controller or target device; i.e., these signals may be in a high impedance state (floating and indicated by X in the bit position tables). In single bit transfers the IO[7:2] signals may be high impedance.

During the period of data transfer in a read transaction, the Data Strobe (DS) signal is driven by the target and transitions edge aligned with the IO signal data transitions. DS is used as an additional output signal with the same timing characteristics as other data outputs but with the guarantee of transitioning with every data bit transferred. The DS signal transitions can be received and internally phase shifted by the controller to be used as an internal read data clock to capture each data bit transferred.

DS transitions may have deterministic or variable timing with reference to CK. Some target devices may be deterministic and delay the toggling of DS for a fixed number of cycles during the initial access latency period then always follow CK transitions during the remainder of a transaction. Some target devices may have variable DS timing by delaying the toggling of DS for a variable number of cycles during the initial access latency phase and then may delay toggling during some following data transfer phase CK cycles – as a means of flow control on target to controller information transfers. However, like CK, DS must always delay (stop) when LOW and always have one rising and falling edge before stopping.

The deterministic or variable DS behavior is target device specific. The controller may treat a delay of DS toggling for more than a number of CK cycles as a transaction error and end the transaction by taking CS# HIGH. The variable or deterministic DS behavior and the number of cycles indicating an error are indicated in the SFDP.

Some commands define other functions for DS during the command and command modifier phase or during write data transfers. These additional functions are described in the command descriptions.

5.5 Legacy SPI Compatibility

xSPI target devices support the 1S-1S-1S protocol for limited backward compatibility with the legacy single IO Serial Peripheral Interface (SPI) protocol. The 1S-1S-1S mode supports a limited subset of SPI protocol commands, to enable basic target device identification in the same protocol used by legacy SPI target devices. This backward compatibility can simplify the POR and configuration process for systems that may use legacy SPI targets on the xSPI interface signals. Once each target device is identified, it is expected that the protocol will be switched to 8D-8D-8D.

6 xSPI Transaction Descriptions

6.1 Target Selection

Each transaction starts with CS# for a particular target device going from HIGH to LOW. Each transaction ends when that CS# returns from LOW to HIGH. While the CS# is HIGH, the target ignores all other xSPI signals. While CS# is LOW, the CK signal edges cause the transfer of information on the IO signals. CS# must transition from HIGH to LOW or LOW to HIGH while CK is LOW. Thus, CK is LOW when idle or stopped, and is equivalent to the legacy SPI Mode 0 with Clock Polarity (CPOL) = 0 and, Clock Phase (CPHA) = 0. This is the only legacy SPI clock mode supported in xSPI.

6.2 Clock, DS, and Information Transfer

The clock may be a free running (always toggling) signal but is recommended to only toggle while CS# is LOW to minimize power consumption. CK signal edges cause the transfer of information on the IO signals. If the differential clock option is used, the crossing point of CK and CK# causes the transfer of information on the IO signals.

NOTE A free running clock will meet the timing in this standard only at relatively low frequencies. For high frequency operation, clock should be gated.

During information transfer from controller to target, the IO signals are driven to a valid HIGH or LOW state for at least a set-up time (t_{ISU}) before a CK edge and are held valid for at least a hold time (t_{IH}) after the CK edge – the CK edge is within the valid IO (data) window. This occurs during the command, command modifier, and write data phases of a transaction.

During an initial access latency phase no information is transferred but CK toggles to act as a timing reference to target device interface logic so that it progresses through the initial access process for data. Legacy SPI protocols sometimes referred to this phase as Dummy cycles.

During information transfer from target to controller; i.e., a read data transfer phase, CK continues to toggle so that the target interface progresses through the read data transfer process. However, CK is not the timing reference for when read data is valid. During a read data phase, DS toggles edge aligned with the IO signals to serve as the read data timing reference. However, for backward compatibility with legacy SPI targets, in the 1S-1S-1S mode, the controller may use CK as the timing reference for read data. This legacy SPI compatible method of clocking read data may be restricted to lower frequency operation depending on the controller and target output driver characteristics. The frequency expectation for commands operating without DS is indicated in the SFDP.

Some commands use DS during the command and command modifier phases to indicate an extension of the initial latency phase. Some commands use DS during a write data phase as a data mask to control byte writes.

6.3 Command Phase

An eight bit command is always the first information transferred in a transaction. The first transfer of command bits happens on the first rising edge of CK after CS# is LOW. In an SDR protocol, any additional command bit transfers occur on the following rising CK edges until the command is completely transferred. In 1S-1S-1S protocol mode this requires 8 rising CK edges. In 8D-8D-8D protocol mode the command is transferred by the first rising edge of CK.

The command transferred determines the format of any remaining transfers during the transaction.

The xSPI protocol uses a subset of the same 8-bit command codes as in the de-facto legacy SPI protocol. To simplify the design of both the controller and target device interfaces, the commands supported vary depending on the protocol mode in use. The xSPI protocol also adds some new command codes not found in legacy SPI protocols.

NOTE Some xSPI devices have a mode in which the command phase is not required. While in this mode, commonly called XiP Mode, the controller skips sending the command and the target device implicitly performs the command that was executed in the previous transaction. While in a current transaction, the controller may indicate that the next transaction will also be in XiP Mode by sending a pre-defined data value during the initial latency phase.

6.4 Command Modifier Phase

Some commands have no following information transfer. These are single byte commands that need no additional information. The transaction ends with CS# returning HIGH after such a command is transferred.

Some commands are followed by an additional byte that serves to further define or confirm the command behavior. In the command summary tables, the Command Extension column indicates:

- **NA** – Not Applicable, for commands that do not have a command extension byte. In this standard, NA is only used in 1S-1S-1S mode.
- **RI** – Repeat or Invert, for commands that either transfer the command again or transfer a bitwise inverted copy of the command as the command extension. It is target specific as to whether a repeated or inverted command is expected. The expectation is indicated in the SFDP. In this standard, RI is only used in profile 1.0 of 8D-8D-8D mode. Note that in this mode the host device must send either a repeated or inverted command while the target device is not required to read both.
- **Hex value** – for commands that use the command extension byte to further define the command function, the specific hexadecimal value of the command extension byte is shown. In this standard, Hex is only used in profile 2.0 of 8D-8D-8D mode.

Some commands or a command extension byte are followed by multiple bytes of additional command modifier that is generally used as an address within a memory array. The number of additional modifier bytes may be defined by the command or depend on a modifier (address) length configuration. The number of additional modifier bytes is shown in the additional modifier column of the command summary tables. A zero indicates that no additional modifier bytes are provided.

6.5 Initial Access Latency Phase

Some commands have no following information transfer. These commands need no additional information transfer after the command modifier phase and therefore no latency phase. The transaction ends with CS# returning HIGH after such a command or command modifier is transferred.

Some commands do not need a latency period before transferring data. Some read commands require data transfer on the cycle following the command modifier phase. Generally, target devices will restrict the use of these commands to lower operating frequencies. Some write commands requires data transfer on the cycle following the command modifier phase.

In the command summary tables a zero (0) in the Initial Access Latency column indicates the command is deterministic and has no initial access latency phase. A non-zero value indicates a deterministic initial access latency. A value of zero or a number plus (0+, 1+, ...) indicates the command has a minimum initial latency that may vary based on a configuration setting or other conditions present during the transaction. Whether the command has deterministic, configurable, or variable latency that requires monitoring when DS transitions is indicated in the SFDP.

6.6 Data Phase

Some commands have no following information transfer. These commands need no additional information transfer after the command modifier phase and therefore have no data phase. The transaction ends with CS# returning HIGH after such a command or command modifier is transferred. In the command summary tables this is indicated by a zero (0) in the data bytes column.

Other commands may transfer one data byte and repeat the data byte if the transaction is continued. This is indicated as a one (1) in the data bytes column.

Other commands may transfer one or more sequential data bytes and is indicated as a one plus (1+) in the data bytes column. Other commands may always transfer an even number of sequential data bytes (always transferring two byte words) and is indicated as a two plus (2+) in the data bytes column.

6.7 Required versus Optional Commands

Some commands that are required, if a protocol mode and command type subset is supported, are listed in the command summary Table 3 Commands used in 1S-1S-1S protocol mode (required commands) and Table 4 Profile 1.0 commands used in 8D-8D-8D protocol mode (required commands) Optional commands are listed in command summary Table 5 Profile 1.0 commands used in 8D-8D-8D protocol mode (optional commands).

Many commands are optional. The commands that are supported by a target device must be indicated in the SFDP.

However, in 8D-8D-8D protocol mode there are two command profiles that although optional as a profile, some commands from one, the other or, both of the command profiles must be supported. These command profiles are:

- Profile 1.0
- Profile 2.0

6.7 Required versus Optional Commands (cont'd)

Profile 1.0 commands directly affect information in memory target devices. The commands may read from a memory array, write to registers, program non-volatile memory, erase non-volatile memory, etc.

Transport commands are not specific to memory devices. These commands move data into (write) or out of (read) a target xSPI device in a general purpose way. The commands read or write to either a memory address space or a register address space. The commands also specify whether the read or write transfers data in a linear or wrapped sequence. The meaning of the data transferred is target device specific. The target device accessed by these commands may be a volatile or non-volatile memory or some type of peripheral such as a graphics display, or network adapter, or some other function implemented in a Field Programmable Gate Array (FPGA).

If a command is required when Profile 1.0 is supported, it is indicated with 1 in the Req. column. If a command is required when Profile 2.0 is supported, it is indicated with 1 in the Req. column.

It is compliant with the xSPI standard to implement the required commands from either the required Profile 1.0 commands, or the required Profile 2.0 commands or, the required commands from both of these profiles.

6.8 Maximum Frequency

Some commands, if supported, are only required to operate at or below a maximum frequency. If a maximum frequency is defined for a command it is shown in the maximum frequency column of the command summary tables.

6.9 Command Summary Tables

The main characteristics of commands for each protocol mode are summarized in the following tables. Details of each command implementation are in the Command Description clauses.

6.9.1 1S-1S-1S Mode

Table 3 shows the required commands in the 1S-1S-1S protocol mode. It is expected that this mode will be used only for identification and configuration of the target device. As soon as the device is identified and configured, it is expected that the target device will be switched to a higher data rate protocol mode. See 6.10.2.21 Enter Another Protocol Mode for more information. Note that each command uses one of the Transaction Formats described in 6.9.5 Transaction Formats by Protocol Mode.

Table 3 — Commands Used in 1S-1S-1S Protocol Mode (Required Commands)

Command Function	Command Code (Hex)	Command Extension (Hex)	Additional Modifier Bytes	Initial Latency (CK Cycles)	Data Bytes	Req.	Transaction Format	Req. Freq. (MHz)
Read SFDP	5A	NA	3	8	1+	1	0.E	50+
Read Zero Latency	03	NA	3	0	1+	1	0.C	50+

6.9.2 8D-8D-8D Profile 1.0

Commands are grouped into two Categories:

6.1 Category 1: Non-WREN Enabled Commands require a single CS# cycle

6.2 Category 2: WREN Enabled Commands must be prefixed by one additional CS# cycle for a write enable (WREN) command

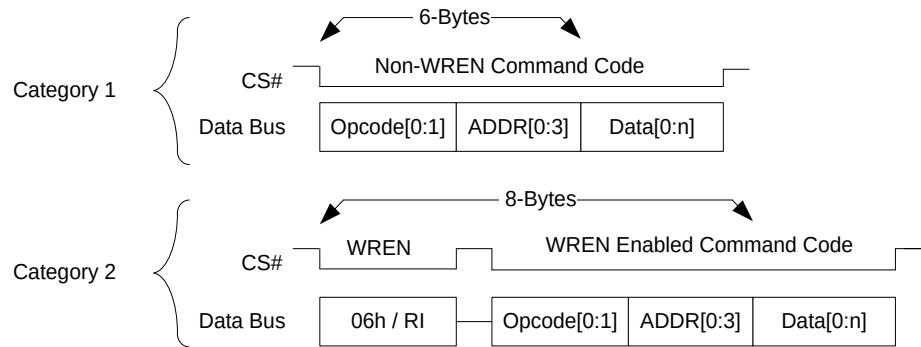


Figure 2 — Command Categories

Table 4 — Profile 1.0 Commands Used in 8D-8D-8D Protocol Mode (Required Commands)

Command Function	Command Code (Hex)	Category	Command Extension (Hex)	Additional Modifier Bytes	Initial Latency (CK Cycles)	Data Bytes	Transaction Format	Req. Max. Freq. (MHz)
Read Fast	0B / EE / SFDP	1	RI	4	1+	2+	1.B	
Write Enable (WREN)	06	1	RI	0	0	0	1.A	
Write Disable	04	1	RI	0	0	0	1.A	
Program	02 / 12 / SFDP	2	RI	4	0	2+	1.D	
Erase 64+Kbytes	D8 / DC / SFDP	2	RI	4	0	0	1.C	
Program / Erase Suspend	B0 / 75 / SFDP	1	RI	0	0	0	1.A	
Program / Erase Resume	30 / 7A / D0 / SFDP	1	RI	0	0	0	1.A	
Read Status Register	05	1	RI	0 / 4 / SFDP	4 / 8 / SFDP	1	1.B	