

IRL2306FI**Flash-type EEPROM integrated circuit with SPI serial interface**

YY — year of
manufacture

WW — week of
manufacture

Package type:

- 8-pin plastic DFN8
5×6×1.6 (1.27)
package.

Key specifications of integrated circuit:

- 64M bit data capacity;
- SPI serial interface available;
- Four 2 Mbyte blocks;
- 1024 sectors of 8 Kbytes each;
- Page size for programming 512 bytes;
- Ability to erase a sector, block, and the entire memory;
- Sector protection function against erase and write: hardware-controlled sector check to prevent erase and write;
- Hardware-controlled algorithm for automatic erase and verification of a page, sector, or entire memory;
- Hardware-controlled algorithm for automatic write and verification of data at a specified address;
- Software method for detecting the end of erase and write cycles;
- Built-in high-voltage programming and erase voltage generation circuit;
- Support for dual-SPI and quad-SPI modes;
- Support for Hold mode — suspension of operations on the SPI bus via an external pin;
- Support for Program/Erase Suspend modes — suspension of program/erase operations;
- Support for low power consumption mode;
- Data retention time 20 years;
- 20,000 data write/erase cycles;
- Supply voltage from 1.8 to 3.6 V;
- Current consumption in storage mode not more than 2 mA;
- Current in low power consumption mode not more than 400 μ A;
- Dynamic current consumption not more than 20 mA;
- Sampling time not more than 6 ns;
- Thermal resistance junction-to-case not more than 30°C/W;
- Moisture Sensitivity Level (MSL): 2a;
- Integrated circuit mass not more than 0.12 g;
- Operating temperature range from minus 40°C to plus 85°C.

Important: integrated circuits are sensitive to moisture.

General description and application areas

The IRL2306FI (hereinafter — integrated circuits) are non-volatile Flash memory devices with up to tens of thousands of rewrite cycles for storing and dynamically modifying data arrays and software in industrial and commercial applications.

Contents

1	Functional block diagram	4
2	Schematic	5
3	Pin description.....	6
4	Application and operating guidelines.....	7
5	Memory organization	8
6	Description of IC operation	9
6.1	Hardware-controlled and software-controlled data protection.....	10
6.2	SPI bus transfer suspend operation	12
6.3	Automatic error correction coding using Hamming code.....	13
6.4	Command description	13
6.5	Flash memory data read operation in SPI mode without dummy byte.....	15
6.6	Flash memory data read operation in SPI mode at maximum frequency	16
6.7	Flash memory data read operation in Dual SPI mode	16
6.8	Flash memory data read operation in Quad SPI mode	17
6.9	Data buffer programming operation in SPI mode	18
6.10	Data buffer programming operation in Dual SPI mode.....	19
6.11	Data buffer programming operation in Quad SPI mode	21
6.12	Sector erase operation	22
6.13	Block erase operation	23
6.14	Full memory erase operation	24
6.15	Program/erase suspend operation	25
6.16	Program/erase resume operation.....	27
6.17	Write enable operation	28
6.18	Write disable operation	28
6.19	Sector protection set operation	29
6.20	Sector protection removal operation	30
6.21	Sector protection register read operation	31
6.22	Status register 1 and 2 read operation	31
6.22.1	Status register description.....	31
6.22.2	SPRL bit.....	33
6.22.3	QE bit.....	33
6.22.4	SWP bits	34
6.22.5	WEL bit	34
6.22.6	BUSY bit	34
6.22.7	E_ERR bit.....	34
6.22.8	P_ERR bit	34
6.22.9	WPP bit.....	35
6.22.10	APS bit.....	35
6.22.11	ES and PS bits	35
6.23	Status register 1 write operation	35
6.24	Error Correcting Code (ECC) status register read operation	36
6.24.1	Status register description.....	36

6.24.2	DED_V bit	37
6.24.3	SEC_V bit	37
6.24.4	DED_R bit	38
6.24.5	SEC_R bit	38
6.24.6	ECC_EN bit	38
6.25	Reset operation.....	38
6.26	IC identification information read operation.....	39
6.27	Integrated circuit SFDP parameters read operation.....	40
6.28	Low power entry operation	50
6.29	Low power exit operation.....	51
6.30	Auto-boot mode	51
6.31	Auto-boot configuration register read operation	53
6.31.1	Description of the auto-boot configuration register.....	53
6.31.2	ABE bit	53
6.31.3	ABSD bits	54
6.31.4	ABSA bits	54
6.32	Auto-boot configuration register write operation.....	54
7	Typical integrated circuit connection diagram	56
8	Timing diagrams	57
9	Electrical parameters	59
10	Maximum permissible and limiting parameters	60
11	Reference parameters	63
12	Typical dependencies	64
13	Integrated circuit dimensional drawing.....	76
14	Ordering information.....	77

1 Functional block diagram

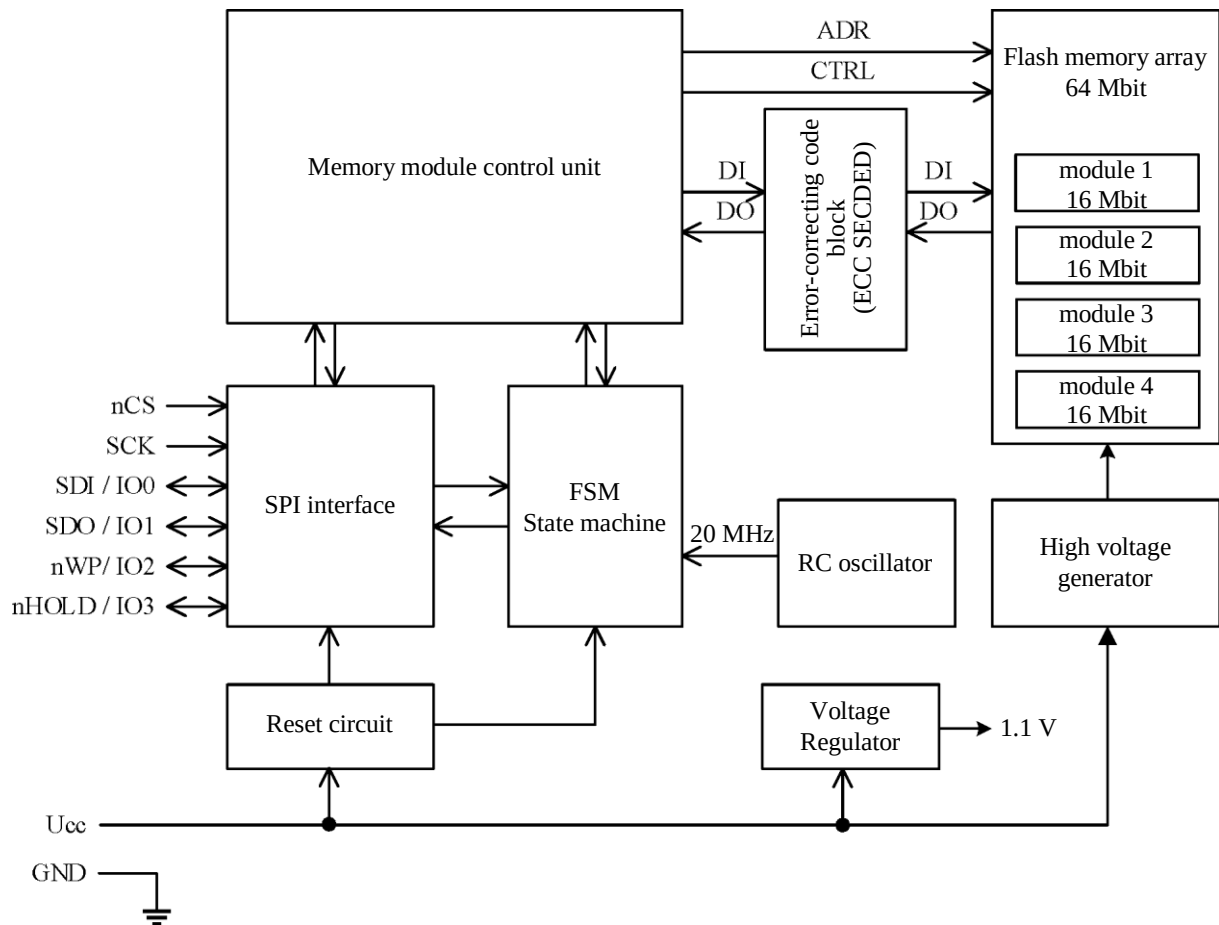


Figure 1 — Functional block diagram

2 Schematic

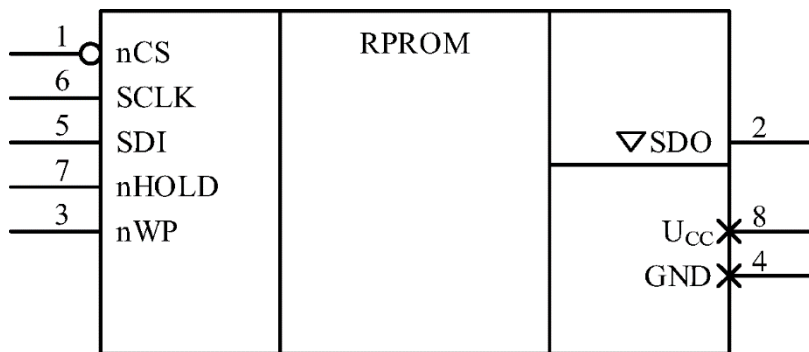


Figure 2 — Schematic

3 Pin description

Table 1 — Pin description

Pin number	Pin identification	Pin mapping and functions
1	nCS	SPI interface enables input with an internal pull-up resistor (active low).
2	SDO	SPI interface data input/output
3	nWP	Write protection signal input with internal pull-up resistor (active low); In QUAD SPI mode — data input/output
4	GND	Common
5	SDI	SPI interface data input/output
6	SCK	SPI interface clock signal input
7	nHOLD	SPI interface temporary suspension signal input with internal pull-up resistor (active low); In QUAD SPI mode — data input/output
8	U _{CC}	Supply (1.8 — 3.6) V

4 Application and operating guidelines

The IC pins feature gold plating of at least 0.0254 μm .

A typical connection diagram for the integrated circuits is shown in Figure 34.

Sequence for applying and removing supply voltage and input signals to the integrated circuits:

- application (powering on the integrated circuit) — ground, power, input signals, or simultaneously;
- removal (powering off the integrated circuit) — simultaneously or in reverse order.

The recommended rise time for applying supply voltage to the integrated circuit is not less than 10 μs . The time until the first memory access is not less than $t_{\text{PWR_ON}}$ from reaching 90% of the supply voltage value.

The dynamic parameters of the integrated circuit are guaranteed for input signal rise/fall times not exceeding 2 ns. The maximum rise/fall time for input signals is not more than 50 ns.

5 Memory organization

The ICs flash memory is segmented into universal erasable units: four physical blocks of 2 Mbytes each. Each block is further divided into 256 sectors of 8 Kbytes, totaling 1024 sectors.

Table 2 — Integrated circuit flash memory address space

Block (2 Mbytes)	Sector (8 Kbytes)	Address range (in hexadecimal)	
BA0	SA0	000000h	001FFFh
	SA1	002000h	003FFFh
	:	:	:
	SA255	1FE000h	1FFFFFFh
BA1	SA256	200000h	201FFFh
	SA257	202000h	203FFFh
	:	:	:
	SA511	3FE000h	3FFFFFFh
BA2	SA512	400000h	401FFFh
	SA513	402000h	403FFFh
	:	:	:
	SA767	5FE000h	5FFFFFFh
BA3	SA768	600000h	601FFFh
	SA769	602000h	603FFFh
	:	:	:
	SA1023	7FE000h	7FFFFFFh

6 Description of IC operation

The integrated circuit can be controlled by a host controller that issues instructions in SPI master mode. For basic SPI commands in serial single-bit mode, the master transmits the command code, address, and data via the serial SDI channel and receives data via the serial SDO channel. In Dual and Quad SPI modes, command codes and addresses are transmitted via the serial SDI/IO0 channel. Data is transmitted by the integrated circuit and received by the master in groups via the SDI/IO0 and SDO/IO1 bit pair for Dual SPI, or the SDI/IO0, SDO/IO2, nWP/IO3, nHOLD/IO4 bit quartet for Quad SPI.

The SPI protocol has four operating modes (0, 1, 2, or 3), differing in the polarity and phase of the SCK signal. The integrated circuit supports two most commonly used modes: 0 and 3. The timing diagram for SPI modes 0 and 3 is shown in Figure 3. The difference between them lies in the inactive state of the SCK line when the SPI master is in no-data-transfer mode. In both modes, data is always captured from the bus on the rising edge of SCK and always placed on the bus on the falling edge of SCK.

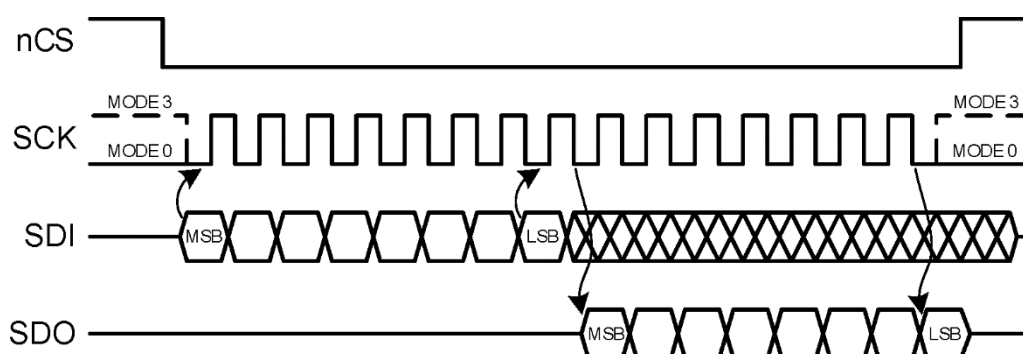


Figure 3 — Timing diagram for SPI modes 0 and 3

Valid instructions must always begin by setting the nCS line to a logical “0” state. Once nCS is set, the host controller must place a valid 8-bit command code on the SPI bus. Then, depending on the command type, address and/or data clocked by the host controller may be placed on the bus. All command codes and addresses are transmitted via the SDI line with the most significant bit (MSB) first. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit (MSB) first. Setting the nCS line to a logical “1” state initiates the execution of internal operations and completes the read operation.

Unsupported command codes are ignored, and the integrated circuit remains in an inactive state. Subsequent data on the SDI input is also ignored until the nCS line is set to a logical “1” state, and then to a logical “0” state to issue the next command. If the nCS line is set to a logical “1” state before the command code and address information transmission is complete, the operation will not start, and the integrated circuit will return to an inactive state to await the next command code.

If a high-level voltage is applied to the nCS pin and no internal programming or erase operations are being performed, the integrated circuit is in storage mode and awaits the next command code. In this mode, the current consumption decreases to I_{CCS} , and the outputs are in the off state.

To transmit an address consisting of address bits A22-A0, 3 bytes of information must be sent over the SPI bus. Address bit A23, transmitted over the bus, is always ignored, as the maximum addressable range of the memory device is 000000h-7FFFFh.

6.1 Hardware-controlled and software-controlled data protection

When power is applied, the integrated circuit does not recognize commands transmitted to it. The internal control interface automatically enters the command code waiting state after power-up and a time t_{PWR_ON} . In addition, the hardware-controlled and software-controlled protection methods described below prevent accidental data writing or erasing.

The requirement to issue a write enable command code before programming and erase commands programmatically ensures data protection against erroneous writing and erasing.

The integrated circuit implements a data protection scheme that allows setting a range of sector addresses protected from writing and erasing. Information about protected memory regions is stored in the non-volatile sector protection register and encoded by six bits BP0 — BP5 (see Table 3).

Table 3 — Truth table for the non-volatile sector protection register

BP5	BP4	BP3	BP2	BP1	BP0	Protected sectors	Unprotected sectors	Range
x	x	0	0	0	0	—	SA0 — SA1023	All unprotected
0	0	0	0	0	1	SA0	SA1 — SA1023	Lower 1/1024
0	0	0	0	1	0	SA0 — SA1	SA2 — SA1023	Lower 1/512
0	0	0	0	1	1	SA0 — SA3	SA4 — SA1023	Lower 1/256
0	0	0	1	0	0	SA0 — SA7	SA8 — SA1023	Lower 1/128
0	0	0	1	0	1	SA0 — SA15	SA16 — SA1023	Lower 1/64
0	0	0	1	1	0	SA0 — SA31	SA32 — SA1023	Lower 1/32
0	0	0	1	1	1	SA0 — SA63	SA64 — SA1023	Lower 1/16
0	0	1	0	0	0	SA0 — SA127	SA128 — SA1023	Lower 1/8
0	0	1	0	0	1	SA0 — SA255	SA256 — SA1023	Lower 1/4
0	x	1	0	1	0	SA0 — SA511	SA512 — SA1023	Lower 1/2
0	1	0	0	0	1	SA0 — SA767	SA768 — SA1023	Lower 3/4
0	1	0	0	1	0	SA0 — SA895	SA896 — SA1023	Lower 7/8
0	1	0	0	1	1	SA0 — SA959	SA960 — SA1023	Lower 15/16
0	1	0	1	0	0	SA0 — SA991	SA992 — SA1023	Lower 31/32
0	1	0	1	0	1	SA0 — SA1007	SA1008 — SA1023	Lower 63/64
0	1	0	1	1	0	SA0 — SA1015	SA1016 — SA1023	Lower 127/128
0	1	0	1	1	1	SA0 — SA1019	SA1020 — SA1023	Lower 255/256
0	1	1	0	0	0	SA0 — SA1021	SA1022 — SA1023	Lower 511/512
0	1	1	0	0	1	SA0 — SA1022	SA1023	Lower 1023/1024
1	0	0	0	0	1	SA1023	SA0 — SA1022	Upper 1/1024
1	0	0	0	1	0	SA1022 — SA1023	SA0 — SA1021	Upper 1/512
1	0	0	0	1	1	SA1020 — SA1023	SA0 — SA1019	Upper 1/256
1	0	0	1	0	0	SA1016 — SA1023	SA0 — SA1015	Upper 1/128
1	0	0	1	0	1	SA1008 — SA1023	SA0 — SA1007	Upper 1/64

BP5	BP4	BP3	BP2	BP1	BP0	Protected sectors	Unprotected sectors	Range
1	0	0	1	1	0	SA992 — SA1023	SA0 — SA991	Upper 1/32
1	0	0	1	1	1	SA960 — SA1023	SA0 — SA959	Upper 1/16
1	0	1	0	0	0	SA896 — SA1023	SA0 — SA895	Upper 1/8
1	0	1	0	0	1	SA768 — SA1023	SA0 - SA767	Upper 1/4
1	x	1	0	1	0	SA512 — SA1023	SA0 — SA511	Upper 1/2
1	1	0	0	0	1	SA256 — SA1023	SA0 — SA255	Upper 3/4
1	1	0	0	1	0	SA128 — SA1023	SA0 — SA127	Upper 7/8
1	1	0	0	1	1	SA64 — SA1023	SA0 — SA63	Upper 15/16
1	1	0	1	0	0	SA32 — SA1023	SA0 — SA31	Upper 31/32
1	1	0	1	0	1	SA16 — SA1023	SA0 — SA15	Upper 63/64
1	1	0	1	1	0	SA8 — SA1023	SA0 — SA7	Upper 127/128
1	1	0	1	1	1	SA4 — SA1023	SA0 — SA3	Upper 255/256
1	1	1	0	0	0	SA2 — SA1023	SA0 — SA1	Upper 511/512
1	1	1	0	0	1	SA1 — SA1023	SA0	Upper 1023/1024
x	x	1	0	1	1	SA0 — SA1023	-	All protected
x	x	1	1	0	0			
x	x	1	1	0	1			
x	x	1	1	1	0			
x	x	1	1	1	1			

In addition to the non-volatile sector protection register, the IC features an nWP pin, which allows hardware-controlled blocking of the erase operation for this register (disabling protection), but does not block the write operation (enabling protection). Thus, if a logical “0” state is applied to the nWP pin, the non-volatile sector protection register acts as a one-time programmable ROM.

The nWP pin only protects the sector protection register from changes. If the nWP pin is set to a logical “0” state before or after issuing a data write or erase command, or during the execution of an internal operation, it will not affect the outcome of its execution.

The nWP pin functionality is active only for SPI and Dual SPI modes when the QE configuration bit of Status Register 1 is set to “0”. When the QE bit is set to “1”, the nWP pin functionality is disabled, and this pin can be used as IO2 for data transfer in Quad SPI mode.

The nWP pin has an internal pull-up resistor to power and can be left unconnected if the integrated circuit is not used in Quad SPI mode.

In addition to hardware-controlled protection, the integrated circuit implements software-controlled protection for the non-volatile sector protection register. Software control over changes to this register is performed using the SPRL bit of Status Register 1. If the SPRL bit is in the “1” state, the sector protection register is locked and cannot be modified (the integrated circuit ignores these commands). If SPRL is in the logical “0” state, the register is unlocked and can be modified. After power-up, the SPRL bit is in the “0” state by default.

The functionality of the SPRL bit and the nWP pin is combined by a logical OR. Table 4 lists all data protection methods implemented in the integrated circuit.

Table 4 — Data protection methods in the integrated circuit

Hardware-controlled protection	Software-controlled protection (volatile)	Sector protection register (non-volatile)					
nWP pin	SPRL bit	BP5	BP4	BP3	BP2	BP1	BP0
blocks BPx bits from being erased	blocks BPx bits from being written/erased						

6.2 SPI bus transfer suspend operation

The nHOLD pin is used to suspend SPI bus transfers without interrupting the current command and without stopping SCK clocking.

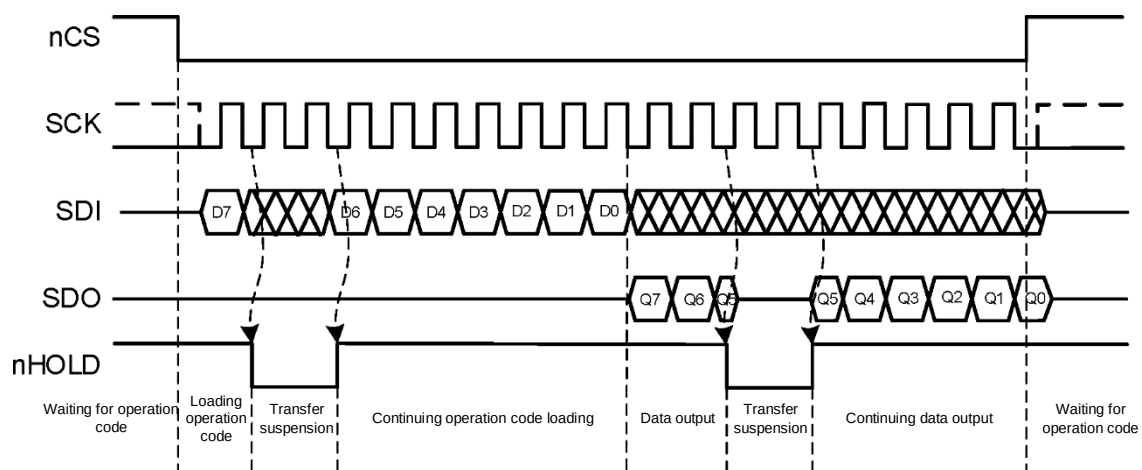
To enter the SPI bus transfer suspend state, the nCS and SCK pins must be in a logical “0” state. It is recommended to hold nCS in the “0” state throughout the entire transfer suspension time. If nCS transitions to a logical “1” state while the integrated circuit is in the SPI bus transfer suspend state, the SPI interface state machine will be reset to its default state (awaiting command code). If the nHOLD pin transitions to a logical “0” state when SCK is in a logical “1” state, the integrated circuit will enter the SPI bus transfer suspend state on the next falling edge of SCK. In the transfer suspend state, the SDO pin is in a high-impedance state, and the SDI and SCK pins can take any value.

To resume operation and exit the SPI bus transfer suspend state, the nHOLD pin must be set to a logical “1” state while nCS and SCK are in a logical “0” state. If the nHOLD pin transitions to a logical “1” state when SCK is in a logical “1” state, the integrated circuit will exit the transfer suspend state on the next falling edge of SCK.

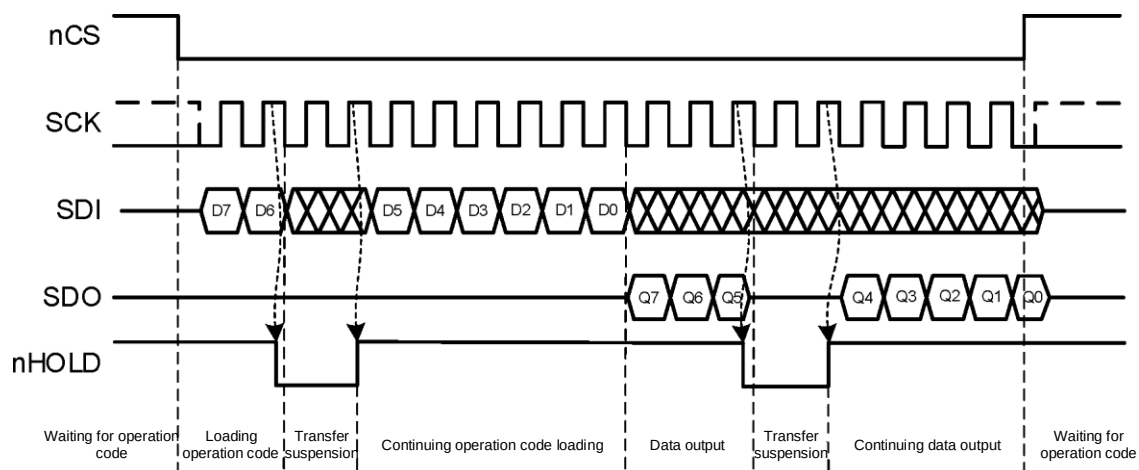
Setting nHOLD to a logical “0” state does not interrupt the execution of internal program/erase operations.

The nHOLD pin functionality is active only for SPI and Dual SPI modes when the QE configuration bit of Status Register 1 is set to “0”. When the QE bit is set to “1”, the nHOLD pin functionality is disabled, and this pin can be used as IO3 for data transfer in Quad SPI mode.

The nHOLD pin has an internal pull-up resistor to power and can be left unconnected if the integrated circuit is not used in Quad SPI mode.



a) nHOLD switching occurs when SCK is at “0”



b) nHOLD switching occurs when SCK is at “1”

Figure 4 — Timing diagram for suspending SPI bus transmission

6.3 Automatic error correction coding using Hamming code

To enhance the reliability of EEPROM operation during use, the integrated circuit incorporates a Hamming code (39,32) error-correction block with single error correction and double error detection (SECEDED) capabilities. Encoding and decoding are applied to all parts of the EEPROM address space, as well as to the non-volatile sector protection register. For each 4-byte data group, aligned to addresses A1 — A0, parity bits are calculated and stored in a hidden memory area next to their corresponding data group. Accordingly, the minimum data buffer length for programming is 4 bytes (see subsection 6.5).

During EEPROM operation, the error-correcting coding block signals double error detection and single error correction by writing information to the ECC status register. The ECCSRRead command is provided for accessing this register (see Table 5 and section 6.20).

6.4 Command description

The list of commands supported by the integrated circuit is provided in Table 5.

Command descriptions are provided in the following subsections.

Table 5 — List of supported commands

Command designation	Command code	Number of address bytes	Number of data bytes			Command purpose
			input	output		
				dummy	significant	
Read	03h (0000 0011b)	3	0	0	1–8M	Read data from flash memory via one channel at a frequency not exceeding 40 MHz
FRead	0Bh (0000 1011b)	3	0	1	1–8M	Read data from flash memory via one channel at maximum frequency
DORed	3Bh (0011 1011b)	3	0	1	1–8M	Read data from flash memory via two channels at max. frequency

Description for Flash-type EEPROM IC with SPI

Command designation	Command code	Number of address bytes	Number of data bytes			Command purpose
			input	output		
				dummy	significant	
QORead	6Bh (0110 1011)	3	0	1	1–8M	Read data from flash memory via four channels at maximum frequency
Program	02h (0000 0010b)	3	4 - 512 : 4	0	0	Program data buffer into flash memory, with data loading into buffer via one channel
DIProgram	A2h (1010 0010b)	3	4 - 512 : 4	0	0	Program data buffer into flash memory, with data loading into buffer via two channels
QIProgram	32h (0011 0010b)	3	4 - 512 : 4	0	0	Program data buffer into flash memory, with data loading into buffer via four channels
SErase	20h (0010 0000b)	3	0	0	0	Erase 8 KB sector
BErase	D8h (1101 1000b)	3	0	0	0	Erase 2 MB block
CErase	60h (0110 0000b)	0	0	0	0	Chip erase
	C7h (1100 0111b)					
PEsuspend	B0h (1011 0000b)	0	0	0	0	Suspend programming or erase
PEResume	D0h (1101 0000b)	0	0	0	0	Resume programming or erase
WriteEn	06h (0000 0110b)	0	0	0	0	Set WEL bit of Status Register 1 to “1”
WriteDis	04h (0000 0100b)	0	0	0	0	Set WEL bit of Status Register 1 to “0”
ProtectRead	E0h (1110 0000b)	0	0	0	1	Read sector protection register
Protect	E1h (1110 0001b)	0	1	0	0	Set sector protection
Unprotect	E2h (1110 0010b)	0	0	0	0	Remove sector protection

Command designation	Command code	Number of address bytes	Number of data bytes			Command purpose
			input	output		
				dummy	significant	
SR1Read	05h (0000 0101b)	0	0	0	1	Read status register 1
SR2Read	07h (0000 0111b)	0	0	0	1	Read status register 2
SR1Write	01h (0000 0001b)	0	1	0	0	Write status register 1
ECCSRRead	18h (0001 1000b)	0	0	0	1	Read error correction code status register
IDRead	9Fh (0001 1111b)	0	0	0	2	Read manufacturer ID and device ID
SFDPRead	5Ah (0101 1010b)	3	0	0	1 - 80	Read SFDP parameters of the integrated circuit
ABRRead	14h (0001 0100)	0	0	0	3	Read auto-load settings register
ABRWrite	15h (0001 0101b)	0	3	0	0	Write auto-load settings register
InSleep	B9h (1011 1001b)	0	0	0	0	Enter low power mode
OutSleep	ABh (1010 1011b)	0	0	0	0	Exit low power mode
Reset	F0h (1111 0000b)	0	1	0	0	Reset

6.5 Flash memory data read operation in SPI mode without dummy byte

The Read command with code 03h is used to read a continuous data stream from the flash memory in SPI mode at a frequency not exceeding 40 MHz.

To perform a read operation, the nCS line must be set to logical “0” and the command code transmitted to the integrated circuit. Following the command code transmission, a 3-byte address must be sent, specifying the starting address of the first byte.

After the address is transmitted, data is immediately output to the SDO line without delay. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit (MSB) first. The integrated circuit contains an internal address counter that automatically increments after each transmitted data byte. If the last byte (address 7FFFFh) of the memory array is read, the integrated circuit continues reading from the

beginning of the array (address 000000h). No delays occur during this process. When the nCS line is set to logical “1”, the read operation terminates, and the SDO line enters a high-impedance state. It is recommended to terminate the read operation at a byte boundary.

The timing diagram for the read operation with code 03h is shown in Figure 5.

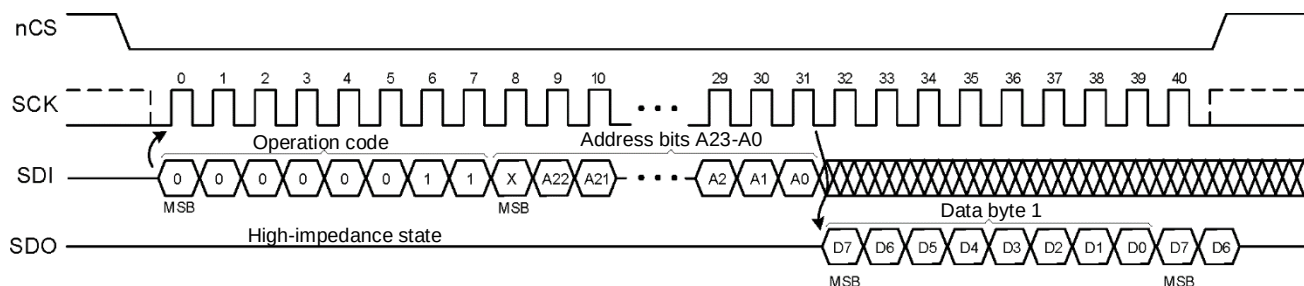


Figure 5 — Read operation with code 03h

6.6 Flash memory data read operation in SPI mode at maximum frequency

The FRead command with code 0Bh is used to read a continuous data stream from the integrated circuit in SPI mode at a frequency up to 100 MHz.

To perform a read operation, the nCS line must be set to logical “0” and the command code transmitted to the integrated circuit. Following the command code transmission, a 3-byte address must be sent, specifying the starting address of the first data byte. Following the address bytes, one dummy data byte must be transmitted.

After transmitting the entire required command sequence, data is output to the SDO line. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit (MSB) first. The integrated circuit contains an internal address counter that automatically increments after each transmitted data byte. If the last byte (address 7FFFFFFh) of the memory array is read, the integrated circuit continues reading from the beginning of the array (address 000000h). No delays occur during this process. When the nCS line is set to logical “1”, the read operation terminates, and the SDO line enters a high-impedance state. It is recommended to terminate the read operation at a byte boundary.

The timing diagram for the read operation with code 0Bh is shown in Figure 6.

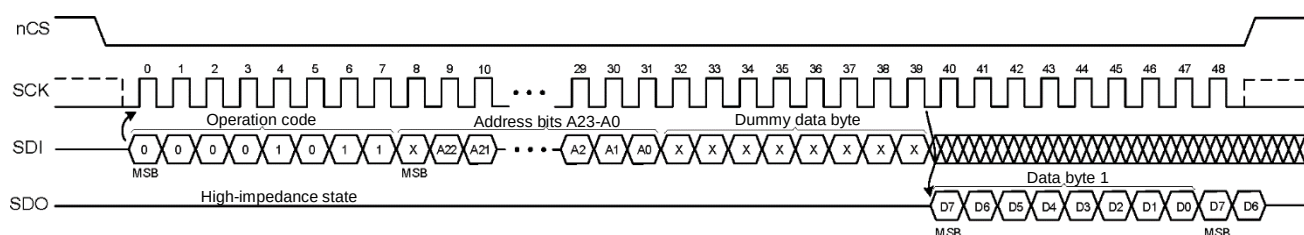


Figure 6 — Read operation with code 0Bh

6.7 Flash memory data read operation in Dual SPI mode

The DORRead command with code 3Bh is used to read a continuous data stream from the integrated circuit in Dual SPI mode at a frequency up to 100 MHz. Unlike the FRead command, the DORRead command allows reading two data bits at once per SCK clock cycle.

To perform a read operation, the nCS line must be set to logical “0” and the command code transmitted to the integrated circuit. Following the command code transmission, a 3-byte address

must be sent, specifying the starting address of the first data byte. Following the address bytes, one dummy data byte must be transmitted.

After transmitting the entire required command sequence, data is read on the SDI/IO0 and SDO/IO1 lines. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit (MSB) first. On the first SCK clock cycle, bit 7 of the first data byte will appear on the SDO/IO1 output, and bit 6 of the same data byte will appear on the SDI/IO0 output. On the next SCK clock cycle, bits 5 and 4 of the first data byte will appear on the SDO/IO1 and SDI/IO0 outputs, respectively. Thus, each data byte will be output in four SCK clock cycles. The integrated circuit contains an internal address counter that automatically increments after each transmitted data byte. If the last byte (address 7FFFFh) of the memory array is read, the integrated circuit continues reading from the beginning of the array (address 00000h). No delays occur during this process. When the nCS line is set to logical “1”, the read operation terminates, and the SDI/IO0 and SDO/IO1 lines enter a high-impedance state. It is recommended to terminate the read operation at a byte boundary.

The timing diagram for the read operation with code 3Bh is shown in Figure 7.

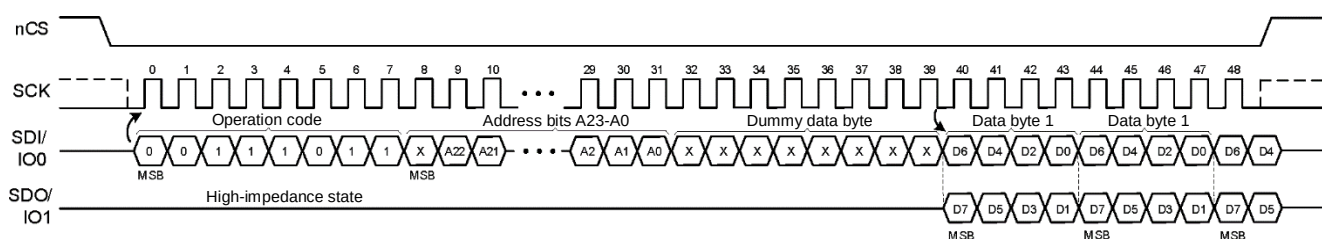


Figure 7 — Read operation with code 3Bh

6.8 Flash memory data read operation in Quad SPI mode

The QORed command with code 6Bh is used to read a continuous data stream from the memory integrated circuit in Quad SPI mode at a frequency up to 100 MHz. Unlike the FRead and DORed commands, the QORed command allows reading four data bits at once per SCK clock cycle.

For the QORed command to operate, the QE configuration bit of status register 1 must be set to “1”. If the QE bit is set to “0”, the QORed command will not be recognized by the integrated circuit.

To perform a read operation, the nCS line must be set to logical “0”, and the command code transmitted to the integrated circuit. Following the command code transmission, a 3-byte address must be sent, specifying the starting address of the first data byte. Following the address bytes, one dummy data byte must be transmitted.

After transmitting the entire required command sequence, data is read on the SDI/IO0, SDO/IO1, nWP/IO2, and nHOLD/IO3 lines. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit (MSB) first. On the first SCK clock cycle, bit 7 of the first data byte will appear on the nHOLD/IO3 output, bit 6 of the same data byte will appear on the nWP/IO2 output, bit 5 on the SDO/IO1 output, and bit 4 on the SDI/IO0 output. On the next SCK clock cycle, bits 3, 2, 1, and 0 of the first data byte will appear on the nHOLD/IO3, nWP/IO2, SDO/IO1, and SDI/IO0 outputs, respectively. Thus, each data byte will be output in two SCK clock cycles. The integrated circuit contains an internal address counter that automatically increments after each transmitted data byte. If the last byte

(address 7FFFFFFh) of the memory array is read, the integrated circuit continues reading from the beginning of the array (address 000000h). No delays occur during this process. When the nCS line is set to logical “1”, the read operation terminates, and the SDI/IO0, SDO/IO1, nWP/IO2, and nHOLD/IO3 lines enter a high-impedance state. It is recommended to terminate the read operation at a byte boundary.

The timing diagram for the read operation with code 6Bh is shown in Figure 8.

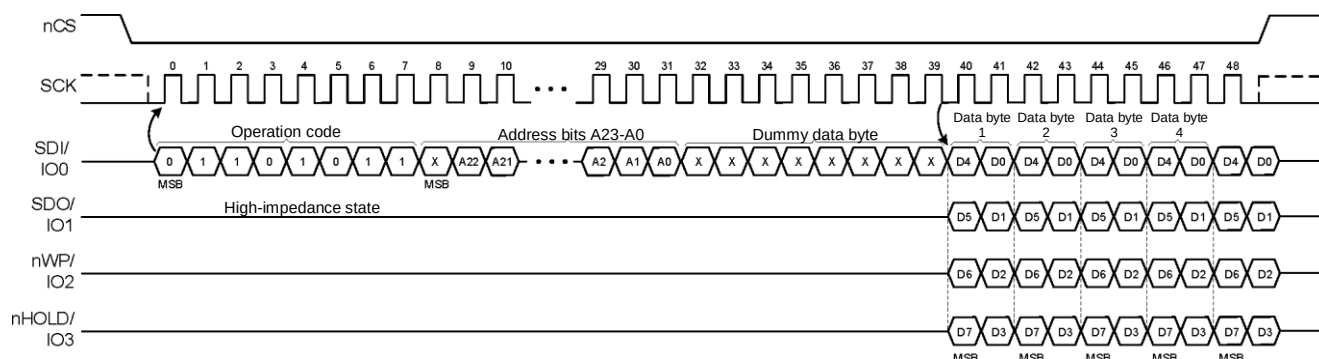


Figure 8 — Read operation with code 6Bh

6.9 Data buffer programming operation in SPI mode

Data buffer programming in SPI mode is performed using the Program command with code 02h and allows programming from 4 to 512 bytes into previously erased memory cells of the integrated circuit. During erase, all memory cells are set to logical “1” (value FFh). Before executing the Program command, the WriteEn command must be executed to set the WEL bit of Status Register 1 to logical “1”.

To execute the Program command, the command code (1 byte: 02h), the starting address (3 bytes), and the data to be programmed (from 4 to 512 bytes with a minimum buffer fill step of 4 bytes) must be transmitted to the integrated circuit via the SDI line. 02h), the starting address (3 bytes), and the data to be programmed (from 4 to 512 bytes with a minimum buffer fill step of 4 bytes). The least significant 2 bits of the address are not considered when loading data into the buffer. Data will be loaded with a starting address A1 — A0 equal to “0”.

If the loaded starting address does not fall on a 512-byte page boundary (address bits A8 — A0 are not in a logical “0” state), then any data loaded into the buffer and exceeding the page end (A8 — A0 equal to 1FFh) will be loaded to the beginning of the same page. For example, if the starting address, denoted A22 — A0, is 0001FCh, and eight bytes of data are loaded into the buffer, then the first four bytes of data will be programmed at addresses 0001FCh and 0001FFh, and the last four bytes of data will be programmed at addresses 000000h and 000003h. The remaining bytes of the page (addresses from 000004h to 0001FCh) will not be programmed and will remain in the erased state (FFh). Furthermore, if more than 512 bytes of data are transmitted to the integrated circuit, only the last 512 bytes will be stored in the internal buffer for subsequent programming.

After the nCS line is set to logical “1”, the integrated circuit will begin programming data from the buffer into memory cells starting at the address transmitted in the Program command. Data must be loaded into the buffer in a quantity that is a multiple of 4 bytes before the nCS line transitions to “1”. If the transmitted packet is not a multiple of 4 bytes, the integrated circuit will abort the operation, and the data will not be programmed into memory cells.

If the address transmitted in the Program command falls within a protected sector, the programming operation will not be performed, the integrated circuit will return to an inactive state after the nCS line transitions to logical “1”, and the APS bit of Status Register 2 will be set to “1”. The value of the WEL bit in Status Register 1 will be reset to “0” before the start of the programming operation if the Program command is successfully accepted, even if the address bits transmitted in the command point to a protected sector. The value of the WEL bit in Status Register 1 will not be reset to logical zero if the programming operation is interrupted by an incomplete address or data.

The waiting time for the programming operation to complete for the minimum allowed buffer size (4 bytes) is not less than t_{PR_WRD} , for a 512-byte buffer — not less than t_{PR_PG} . During the programming operation, it is recommended to use the Status Register 1 read operation, which indicates the status of the internal programming operation's execution. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during programming: if, after the programming operation, the memory cells do not contain the expected value, the P_ERR bit of status register 2 is set to logical “1”. Bits cannot be programmed back from “0” to “1”. Attempting to do so may interrupt the operation and set the P_ERR bit of Status Register 2 to logical “1”. Only an erase operation can change data bits from “0” to “1”.

The timing diagram for the programming operation with the Program command is shown in Figure 9.

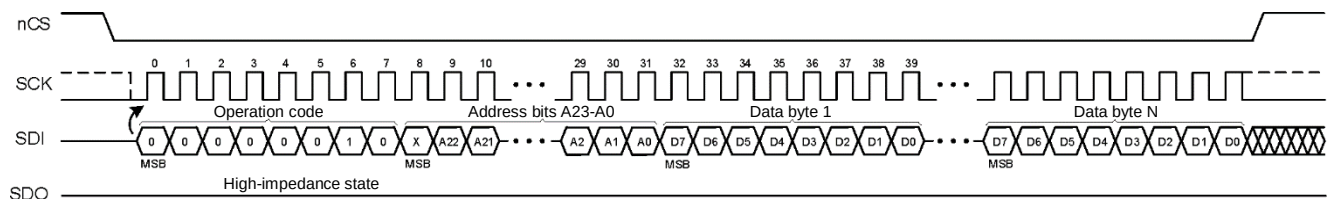


Figure 9 — Programming operation with code 02h

6.10 Data buffer programming operation in Dual SPI mode

Data buffer programming in Dual SPI mode is performed using the DIPProgram command (code A2h), allowing programming from 4 to 512 bytes into previously erased memory cells of the integrated circuit. Unlike the Program command, the DIPProgram command allows transmitting two data bits at once to the integrated circuit per SCK clock cycle. Before executing the DIPProgram command, the WriteEn command must be executed to set the WEL bit of Status Register 1 to logical “1”.

To execute the DIPProgram command, the command code (1 byte: A2h) and the starting address (3 bytes) must be transmitted to the integrated circuit via the SDI line, and the data for programming via the SDI/IO0 and SDO/IO1 lines (from 4 to 512 bytes with a minimum buffer fill step of 4 bytes). A2h) and the starting address (3 bytes), and the data for programming via the SDI/IO0 and SDO/IO1 lines (from 4 to 512 bytes with a minimum buffer fill step of 4 bytes). The least significant 2 bits of the address are not considered when loading data into the buffer. Data will be loaded with a starting address A1 — A0 equal to “0”.

After loading the starting address, the data for programming is transmitted to the integrated circuit on the SDI/IO0 and SDO/IO1 pins, two bits per SCK clock cycle. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most

significant bit (MSB) first. On the first SCK clock cycle, bit 7 of the first data byte must be transmitted on the SDO/IO1 line, and bit 6 of the same data byte must be transmitted on the SDI/IO0 line. On the next SCK clock cycle, bits 5 and 4 of the first data byte are transmitted on the SDO/IO1 and SDI/IO0 lines, respectively. Thus, each data byte will be transmitted in four SCK clock cycles. As with the Program command, all data received by the integrated circuit is stored in an internal buffer before being programmed into memory cells.

If the loaded starting address does not fall on a 512-byte page boundary (address bits A8 — A0 are not in a logical “0” state), then any data loaded into the buffer and exceeding the page end (A8 — A0 equal to 1FFh) will be loaded to the beginning of the same page. For example, if the starting address, denoted A22 — A0, is 0001FCh, and 8 bytes of data are loaded into the buffer, then the first 4 bytes of data will be programmed at addresses 0001FCh and 0001FFh, and the last 4 bytes of data will be programmed at addresses 000000h and 000003h. The remaining bytes of the page (addresses from 000004h to 0001FCh) will not be programmed and will remain in the erased state (FFh). Furthermore, if more than 512 bytes of data are transmitted to the integrated circuit, only the last 512 bytes will be stored in the internal buffer for subsequent programming.

After the nCS line is set to logical “1”, the integrated circuit will start programming data from the buffer into memory cells starting at the address sent in the DIPProgram command. Data must be loaded into the buffer in multiples of 4 bytes before the nCS line goes to “1”. If the transmitted packet is not a multiple of 4 bytes, the integrated circuit will abort the operation and the data will not be programmed into the memory cells.

If the address sent in the DIPProgram command is within a protected sector, the programming operation will not be performed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to “1”. The WEL bit in status register 1 will be cleared to “0” before the start of the programming operation if the DIPProgram command is successfully received, even if the address bits sent in the command point to a protected sector. The WEL bit in status register 1 will not be cleared to logical “0” if the programming operation is interrupted by an incomplete address or data.

The waiting time for the completion of the programming operation for the minimum allowed buffer size (4 bytes) is not less than t_{PR_WRD} , for a 512-byte buffer — not less than t_{PR_PG} . During the programming operation, it is recommended to use the status register 1 read operation, which shows the status of the internal programming operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during programming: if after the programming operation the memory cells do not contain the expected value, the P_ERR bit of status register 2 is set to logical “1”. Bits cannot be programmed back from “0” to “1”. Attempting to do so may interrupt the operation and set the P_ERR bit of status register 2 to logical “1”. Only an erase operation can change data bits from “0” to “1”.

The timing diagram for the programming operation with the DIPProgram command is shown in Figure 10.

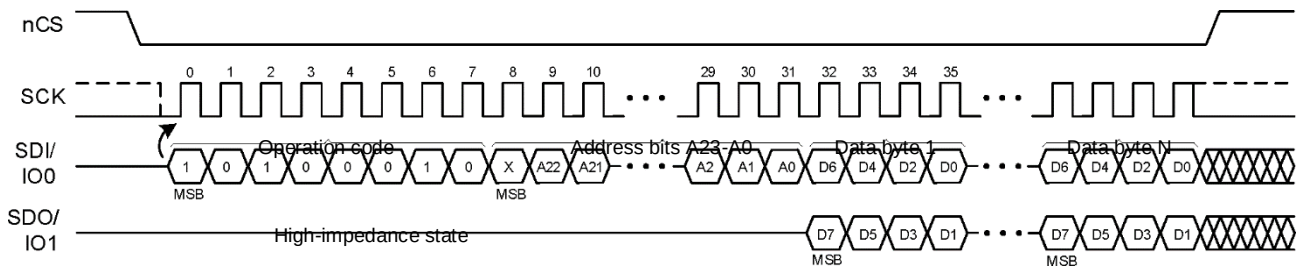


Figure 10 — Programming operation with code A2h

6.11 Data buffer programming operation in Quad SPI mode

Data buffer programming in Quad SPI mode is performed using the QIProgram command (code 32h), which allows programming from 4 to 512 bytes into previously erased memory cells of the integrated circuit. Unlike the Program and DIProgram commands, the QIProgram command allows transmitting four data bits to the integrated circuit at once per SCK clock cycle. Before executing the QIProgram command, the Write Enable command must be executed to set the WEL bit of status register 1 to logical “1”.

For the QIProgram command to operate, the QE configuration bit of status register 1 must be set to “1”. If the QE bit is set to “0”, the QIProgram command will not be recognized by the integrated circuit.

To execute the QIProgram command, the command code (1 byte: 32h) and the starting address (3 bytes) must be transmitted to the integrated circuit via the SDI line, and the data for programming via the SDI/IO0 and SDO/IO1, nWP/IO2, and nHOLD/IO3 lines (from 4 to 512 bytes with a minimum buffer fill step of 4 bytes). The lower 2 address bits are not considered when loading data into the buffer. Data will be loaded with a starting address A1 — A0 equal to “0”.

After loading the starting address, data for programming is transmitted to the integrated circuit on the SDI/IO0 and SDO/IO1, nWP/IO2, and nHOLD/IO3 pins, four bits per SCK clock cycle. Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus most significant bit first (MSB). On the first SCK clock cycle, bit 7 of the first data byte must be transmitted via the nHOLD/IO3 line, bit 6 of the same data byte must be transmitted via the nWP/IO2 line, bit 5 via SDO/IO1, and bit 4 via SDI/IO0. On the next SCK clock cycle, bits 3, 2, 1, and 0 of the first data byte are transmitted via the nHOLD/IO3, nWP/IO2, SDO/IO1, and SDI/IO0 lines, respectively. Thus, each data byte will be transmitted in two SCK clock cycles. As with the Program and DIProgram commands, all data entering the integrated circuit is stored in an internal buffer before being programmed into memory cells.

If the loaded starting address does not fall on a 512-byte page boundary (address bits A8 — A0 are not in a logical “0” state), then any data loaded into the buffer and exceeding the end of the page (A8 — A0 equal to 1FFh) will be loaded to the beginning of the same page. For example, if the starting address, denoted A22 — A0, is 0001FCh, and eight bytes of data are loaded into the buffer, then the first four bytes of data will be programmed at addresses 0001FCh and 0001FFh, and the last four bytes of data will be programmed at addresses 000000h and 000003h. The remaining bytes of the page (addresses from 000004h to 0001FCh) will not be programmed and will remain in the erased state (FFh). Additionally, if more than 512 bytes of data are transmitted to the integrated circuit, only the last 512 bytes will be stored in the internal buffer for subsequent programming.

After the nCS line is set to logical “1”, the integrated circuit will start programming data from the buffer into memory cells starting at the address sent in the QIProgram command. Data must be loaded into the buffer in multiples of 4 bytes before the nCS line goes to “1”. If the transmitted packet is not a multiple of 4 bytes, the integrated circuit will abort the operation and the data will not be programmed into the memory cells.

If the address sent in the QIProgram command is within a protected sector, the programming operation will not be performed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to “1”. The WEL bit in status register 1 will be cleared to “0” before the start of the programming operation if the QIProgram command is successfully received, even if the address bits sent in the command point to a protected sector. The WEL bit in status register 1 will not be cleared to logical “0” if the programming operation is interrupted by an incomplete address or data.

The waiting time for the completion of the programming operation for the minimum allowed buffer size (4 bytes) is not less than t_{PR_WRD} , for a 512-byte buffer — not less than t_{PR_PG} . During the programming operation, it is recommended to use the status register 1 read operation, which shows the status of the internal programming operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during programming: if after the programming operation the memory cells do not contain the expected value, the P_ERR bit of status register 2 is set to logical “1”. Bits cannot be programmed back from “0” to “1”. Attempting to do so may interrupt the operation and set the P_ERR bit of status register 2 to logical “1”. Only an erase operation can change data bits from “0” to “1”.

The timing diagram for the programming operation with the QIProgram command is shown in Figure 11.

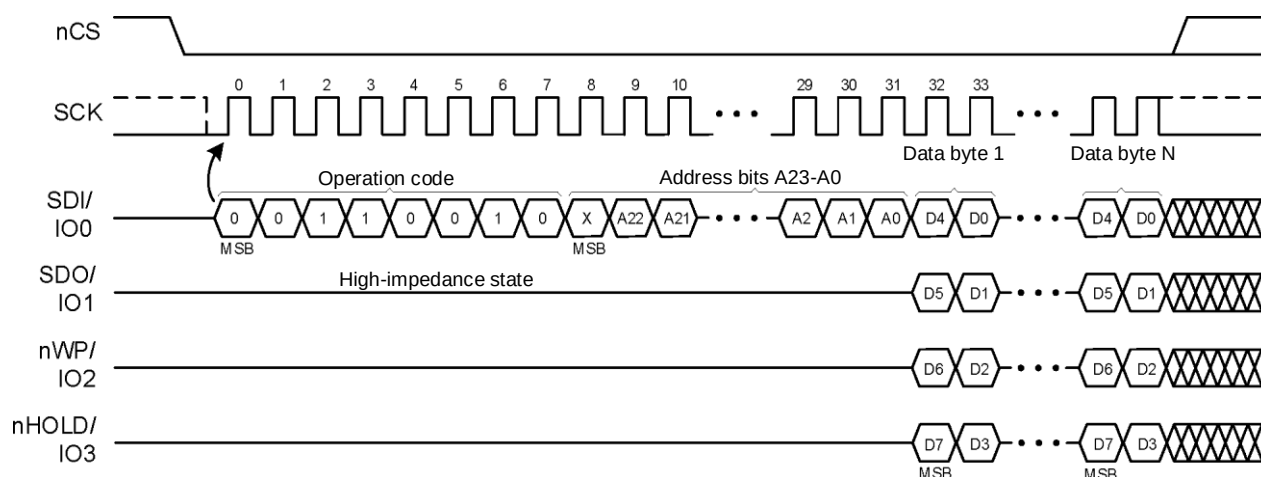


Figure 11 — Programming operation with code 32h

6.12 Sector erase operation

An 8 Kbyte sector can be erased (all bits set to logical “1”) by the SErase command. Operation code 20h is used to execute the command.

Before executing the SErase command, the WriteEn command must be executed to set the WEL bit of status register 1 to logical “1”.

To execute the SErase command, the nCS line must be set to logical “0”, and the command code (1 byte: 20h) and the address of the sector to be erased (3 bytes) must be transmitted. All subsequent data loaded into the device is ignored. After the nCS line is set to logical “1”, the corresponding sector erase begins.

The lower address bits A12 — A0 are not decoded when determining the number of the sector to be erased, so they can be in a logical “0” or “1” state. The sector address consists of address bits A22 — A13, allowing selection of one of 1024 sectors. The integrated circuit's address space is presented in Table 2.

Even though the lower address bits are not decoded, all three address bytes must be transmitted to the integrated circuit before the nCS line goes to logical “1”. If this does not occur on a byte boundary (multiple of 8 SCK clock cycles), the integrated circuit will abort the operation and the erase will not be performed.

If the address bits sent in the command point to a protected sector, the SErase command will also not be executed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to “1”.

The WEL bit in status register 1 will be cleared to logical “0” before the start of the erase operation if the SErase command is successfully received, even if the address bits sent in the command point to a protected sector. The WEL bit in status register 1 will not be cleared if the erase operation is interrupted by an incomplete address.

The waiting time for the completion of the sector erase operation is not less than t_{ER_SEC} . During the erase operation, it is recommended to use the status register 1 read operation, which shows the status of the internal erase operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during sector erase: if after the erase operation at least one memory cell is not equal to “1”, the E_ERR bit of status register 2 is set to logical “1”.

The timing diagram for the sector erase operation is shown in Figure 12.

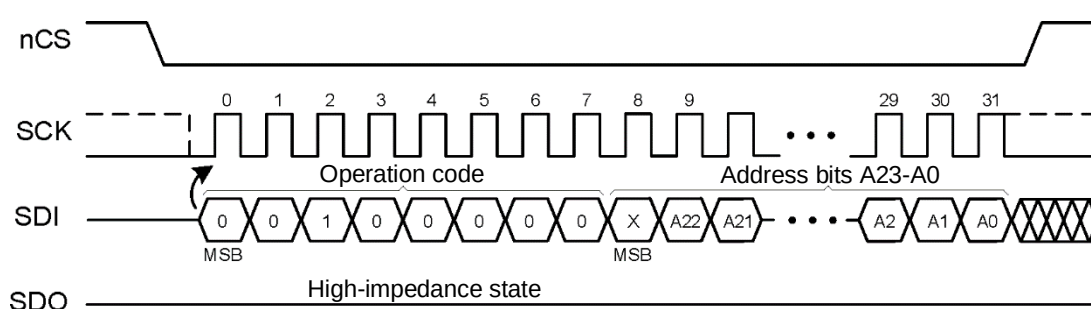


Figure 12 — Sector erase operation with code 20h

6.13 Block erase operation

A 2 MB block can be erased (all bits set to logical “1”) by the BErase command. Operation code D8h is used to execute the command.

Before executing the BErase command, the WriteEn command must be executed to set the WEL bit of status register 1 to logical “1”.

To execute the BErase command, the nCS line must be set to logical “0”, and the command code (1 byte: D8h) and the address of the block to be erased (3 bytes) must be transmitted. All

subsequent data loaded into the device is ignored. After the nCS line is set to logical “1”, the corresponding block erase begins.

The lower address bits A20 — A0 are not decoded when determining the number of the block to be erased, so they can be in a logical “0” or “1” state. The block address consists of address bits A22 — A21, allowing selection of one of four blocks. The integrated circuit's address space is presented in Table 2.

Even though the lower address bits are not decoded, all three address bytes must be transmitted to the integrated circuit before the nCS line goes to logical “1”. If this does not occur on a byte boundary (multiple of 8 bits), the integrated circuit will abort the operation and the erase will not be performed.

If the address bits sent in the command point to a block containing one or more protected sectors, the BErase command will also not be executed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to logical “1”.

The WEL bit in status register 1 will be cleared to logical “0” before the start of the erase operation if the BErase command is successfully received, even if the address bits sent in the command point to a block containing protected sectors. The WEL bit in status register 1 will not be cleared if the erase operation is interrupted by an incomplete address.

The waiting time for the completion of the block erase operation is not less than t_{ER_BLK} . During the erase operation, it is recommended to use the status register 1 read operation, which shows the status of the internal erase operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during block erase: if after the erase operation at least one memory cell is not equal to “1”, the E_ERR bit of status register 2 is set to logical “1”.

The timing diagram for the block erase operation is shown in Figure 13.

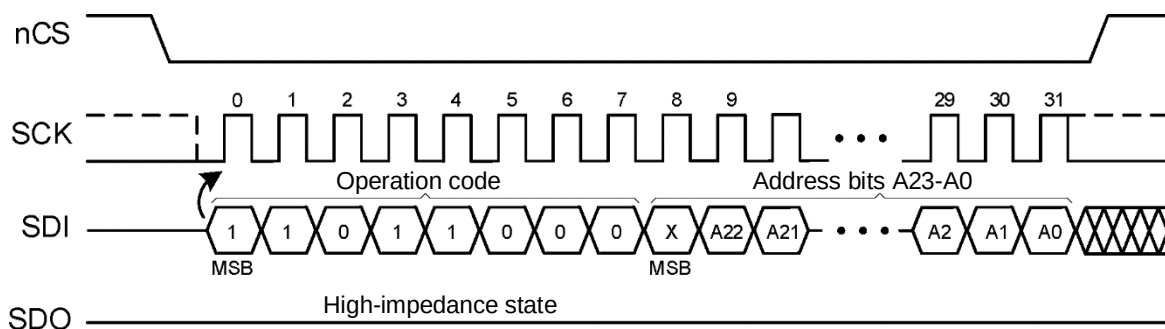


Figure 13 — Block erase operation with code D8h

6.14 Full memory erase operation

The entire memory can be erased (all bits set to logical “1”) by the CErase command. Before executing the CErase command, the WriteEn command must be executed, which sets the WEL bit of the status register to logical “1”.

To execute the CErase command, the integrated circuit must be sent code 60h or C7h (both codes are valid). When erasing the entire memory, there is no need to transmit address bytes to the integrated circuit. Any data after the operation code transmission will be ignored. After the nCS line is set to logical “1”, the entire memory erase process will begin. nCS must switch on a byte

boundary (multiple of 8 SCK clock cycles), otherwise the erase operation will not be performed. If the memory array blocks have protected sectors, the CErase command will not be executed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to logical “1”.

The WEL bit in the status register will be cleared to logical “0” before the start of the erase operation if the CErase command is successfully received, even if the memory array blocks contain protected sectors. The WEL bit in the status register will not be cleared to logical “0” if the command was not fully transmitted.

The waiting time for the completion of the full memory erase operation is not less than t_{ER_CHIP} . During the erase operation, it is recommended to use the status register 1 read operation, which shows the status of the internal erase operation. This will reduce the waiting time for the operation to complete.

The integrated circuit also performs an error detection algorithm during erase: if after the erase operation at least one memory cell is not equal to “1”, the E_ERR bit of status register 2 is set to logical “1”.

The timing diagram for the full memory erase operation is shown in Figure 14.

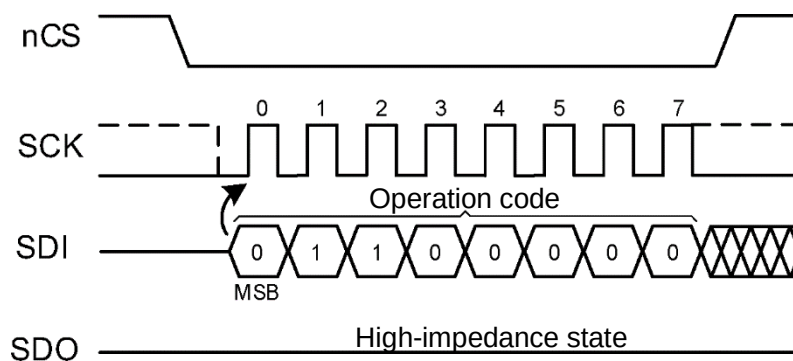


Figure 14 — Full memory erase operation with code 60h

6.15 Program/erase suspend operation

In complex systems, high-priority system requests often arise that require immediate reading of data from flash memory or programming data into flash memory. In such cases, the system may not wait for the microseconds or milliseconds required to complete the flash memory programming or erase cycle. The PESuspend command allows suspending the execution of a block/sector programming or erase operation so that other operations with the integrated circuit can be performed. For example, by suspending an erase operation in a specific sector or block, the system can perform operations such as programming or reading in another block. Table 6 lists the operations that are allowed and not allowed during program or erase suspend.

Table 6 — List of allowed commands during program or erase suspend

Command designation	Program suspend	Erase suspend	Comment
Read	+	+	After program suspend, all sectors are accessible except the one involved in programming.
FRead	+	+	
DORed	+	+	

Description for Flash-type EEPROM IC with SPI

Command designation	Program suspend	Erase suspend	Comment
QORead	+	+	After erase suspend, all blocks are accessible except the one involved in erasing
Program	-	+	After erase suspend, all blocks are accessible except the one involved in erasing
DIPProgram	-	+	
QIPProgram	-	+	
SErase	-	-	
BErase	-	-	
CErase	-	-	
PEsuspend	-	-	
PEResume	+	+	
WriteEn	+	+	
WriteDis	+	+	
ProtectRead	+	+	
Protect	-	-	
Unprotect	-	-	
SR1Read	+	+	
SR2Read	+	+	
SR1Write	-	-	
ECCSRRead	+	+	
IDRead	+	+	
SFDPRead	-	-	
ABRRead	+	+	
ABRWrite	-	-	
InSleep	-	-	
OutSleep	-	-	
Reset	+	+	
Note — The table uses the following notations: + — command allowed; - — command forbidden			

Since the need to suspend a program or erase operation arises immediately, there is no need to set the WEL bit of status register 1 to “1” before issuing the PESuspend command. To suspend a program or erase operation, code B0h must be transmitted. Any data after the command code transmission will be ignored. nCS must switch to logical “1” on a byte boundary (multiple of 8 SCK clock cycles), otherwise the program/erase suspend operation will not be performed. After the nCS line is set to logical “1”, the currently executing program or erase operation will be suspended within time $t_{W(SUSP_PR)} / t_{W(SUSP_ER)}$. The PS bit or ES bit in status register 2 is set to logical “1”, indicating that the program or erase operation has been suspended. Additionally, the BUSY bit in status register 1 will indicate that the device is ready to receive a command to start another operation.

When attempting to read using Read, FRead, DORed, QORead commands from a 512-byte page for which a programming operation is suspended, as well as for a 2 MB block for which an erase operation is suspended, undefined data will appear on the integrated circuit's outputs.

Data programming operations (Program, DIProgram, QIProgram) are prohibited for a 2 MB block for which an erase operation has been suspended, even if it is a sector erase operation within that block. If an attempt is made to program this block, the command will not be executed, the integrated circuit will return to an inactive state after the nCS line is set to logical “1”, the APS bit of status register 2 will be set to logical “1”, and the WEL bit in status register 1 will be reset to logical “0”. Erase operations are prohibited for all sectors/blocks if a programming or erase operation has been suspended. If an erase is attempted, the command will not be accepted by the integrated circuit, and the value of the WEL bit in status register 1 will not be reset to logical “0”.

If a Reset command is issued while an erase is suspended in a sector/block, the operation will be aborted, and the contents of that sector/block will remain in an undefined state. However, if a Reset command is issued while programming is suspended, the operation will be aborted, but only the content of the page being programmed will remain in an undefined state. The other pages in the 8 KB sector will retain their previous content.

If a command prohibited during programming or erase suspension, such as the Protect command, is sent, the IC will ignore it, and the operation will not be performed. The state of the WEL bit in status register 1, as well as the BP bits of the sector protection register, will not be affected.

The timing diagram for the program/erase suspend operation is shown in Figure 15.

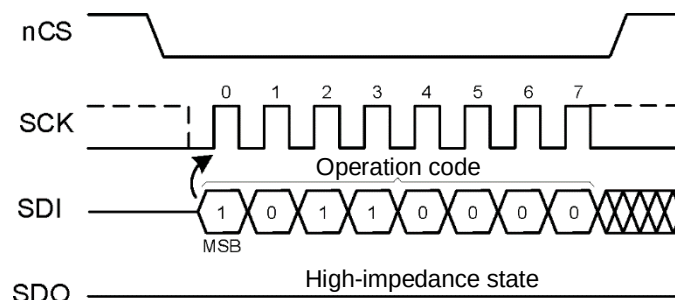


Figure 15 — Program/erase suspend operation

6.16 Program/erase resume operation

The PEResume command allows resuming a previously suspended programming or erase operation from the point at which it was suspended. As with the PESuspend command, there is no need to issue a WriteEn command before issuing the PEResume command.

To resume programming or erasing, the command code D0h must be transmitted to the integrated circuit. Any data after the code transmission will be ignored. nCS must switch at a byte boundary (multiple of 8 SCK clocks), otherwise the erase operation will not be performed. After the nCS line is set to logical “1”, the currently suspended programming or erase operation will be resumed within time $t_{W(RSUME)}$. The PS bit or ES bit in status register 2 will be reset to “0” to indicate that the programming or erase operation is no longer suspended. Additionally, the BUSY bit in status register 1 will indicate that the device is busy performing a programming or erase operation.

While the device is busy resuming a programming or erase operation, any attempts to issue a PESuspend command will be ignored. Therefore, if a resumed programming or erase operation needs to be suspended again later, the system must either wait for the entire t_{RESUME} time before issuing the PESuspend command, or check the state of the BUSY bit or the corresponding PS or

ES bits in status register 2 to determine if the previously suspended programming or erase operation has resumed.

The timing diagram for the program/erase resume operation is shown in Figure 16.

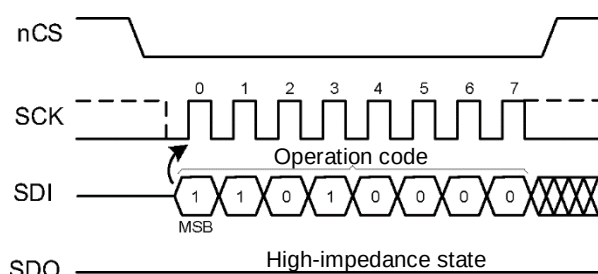


Figure 16 — Program/erase resume operation

6.17 Write enable operation

The WriteEn command is used to set the write enable bit WEL of status register 1 to “1”. The WEL bit must be set before executing Program, DIProgram, QIProgram, SErase, BErase, CErase, Protect, Unprotect, SR1Write commands. This allows these commands to be executed in two stages, reducing the possibility of accidental or erroneous execution of these commands. If the WEL bit in status register 1 is not set before issuing one of these commands, the command will not be executed.

To execute the WriteEn command, the code 06h must be transmitted to the integrated circuit. Any data after the command code transmission will be ignored. nCS must switch at a byte boundary (multiple of 8 SCK clocks), otherwise the WEL bit setting operation will not be performed. After the nCS line is set to logical “1”, the WEL bit of status register 1 is set to “1”. The operation code must be fully loaded into the memory module before the nCS signal changes, otherwise the operation will be aborted and the WEL bit will not change.

6.18 Write disable operation

The WriteDis command is used to reset the write enable bit WEL of status register 1 to “0”. After this, the Program, DIProgram, QIProgram, SErase, BErase, CErase, Protect, Unprotect, SR1Write commands cannot be executed. Other conditions for resetting the WEL bit are given in subsection 6.22 “Status Register 1 and 2 Read Operation”.

When issuing the WriteDis command, the nCS line must be at logical “0”, and the command code 04h must be loaded into the memory module. Loading address bytes into the memory module is not required; all data transmitted after the operation code is ignored. After the nCS line transitions to logical “1”, the WEL bit of the status register is reset to “0”. The operation code must be fully loaded into the memory module before the nCS signal changes, otherwise the operation will be aborted and the WEL bit will not change.

Timing diagrams for write enable and disable operations are shown in Figures 17 and 18.

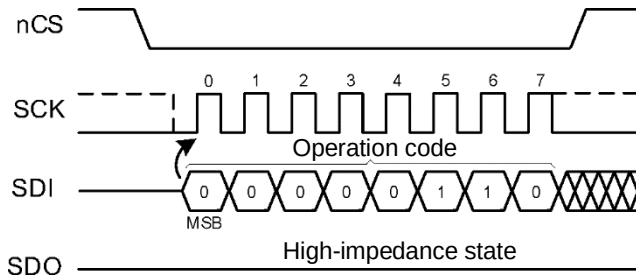


Figure 17 — Write enable operation

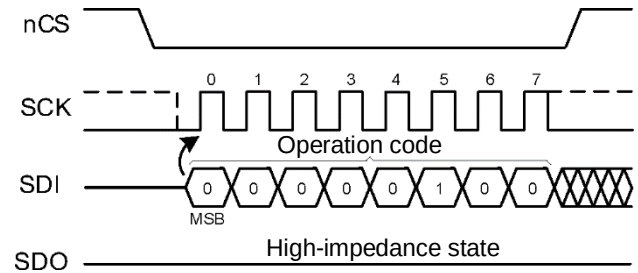


Figure 18 — Write disable operation

6.19 Sector protection set operation

The integrated circuit has a 6-bit non-volatile sector protection register that allows setting protection against programming and erase operations for a selected address range of flash memory sectors. The range size is determined by the value of the BP bits (see Table 3). Upon integrated circuit delivery, all bits of the protection register are by default in a logical “0” state, meaning all sectors are unprotected and can be programmed or erased.

Before issuing the Protect command, the WriteEn command must be used to set the write enable bit WEL to “1”. To execute the Protect command, the command code (1 byte: E1h) and data with the new value of the BP bits (1 byte). The most significant 2 bits of data are not considered when loading data, as the protection register is 6-bit. Table 3 shows the possible values of the BP bits. Any additional data transmitted to the memory device after this is ignored. After the nCS line is set to logical “1”, the integrated circuit will begin programming the new value of the BP bits into the memory cells of the non-volatile sector protection register.

The nCS line must transition to “1” at a byte boundary (multiple of 8 SCK clocks), otherwise the memory module will abort the operation, and the BP bits will not be programmed into the memory cells.

If an attempt is made to set protection again without first removing it, the operation will not be performed, the memory module will return to an inactive state after the nCS line is set to logical “1”, and the APS bit of status register 2 will be set to logical “1”. The value of the WEL bit in status register 1 will be reset to logical “0” before the start of the protection setting operation if the Protect command is successfully accepted, even if a repeated attempt to set protection was made. To change the state of the BP bits from “1” to “0” in the non-volatile data protection register, it is first necessary to remove protection using the Unprotect command and only after that proceed to set protection using the Protect command.

The waiting time for the completion of the sector protection setting operation is not less than t_{PRT} . During sector protection setup, it is recommended to use the status register 1 read operation, which shows the status of the internal programming operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during protection setting: if, after the programming operation, the non-volatile sector protection register cells do not contain the expected value, the P_ERR bit of status register 2 is set to “1”.

To protect against accidental or erroneous removal or setting of sector protection, the register can be locked from modification using the SPRL bit of status register 1. If the sector protection register is locked, any attempt to issue a Protect command will be ignored, the integrated circuit

will reset the WEL bit of status register 1 to logical “0” and return to the command code waiting state.

The timing diagram for the sector protection set operation is shown in Figure 19.

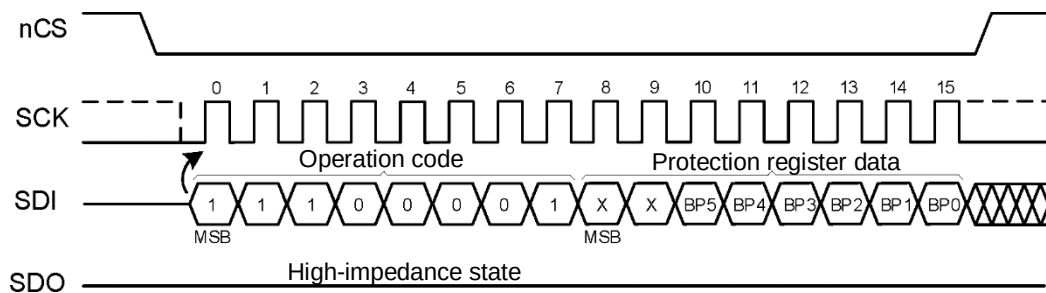


Figure 19 — Sector protection set operation

6.20 Sector protection removal operation

Issuing the Unprotect command resets the sector protection register to all “0”s.

Before issuing the Unprotect command, the WriteEn command must be used to set the write enable bit WEL to “1”. To execute the Unprotect command, only the command code (1 byte: E2h). Unlike the Protect command, there is no need to transmit a data byte with the state of the BP bits, as the Unprotect command can only set the BP bits to all “0”s, meaning that all memory sectors are unprotected from writing and erasing. Therefore, any additional data transmitted to the memory device after the command code is ignored.

After the nCS line is set to logical “1”, the integrated circuit will begin erasing the BP bits in the memory cells (setting to “0”) of the non-volatile sector protection register.

The nCS line must transition to “1” at a byte boundary (multiple of 8 SCK clocks), otherwise the memory module will abort the operation, and the BP bits will not be changed to “0”.

The waiting time for the completion of the sector protection removal operation is not less than t_{UNPRT} . During sector protection removal, it is recommended to use the status register 1 read operation, which shows the status of the internal erase operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm during protection removal: if, after the erase operation, the cells of the non-volatile register are not equal to “0”, the E_ERR bit of status register 2 is set to “1”.

To protect against accidental or erroneous removal or setting of sector protection, the register can be locked from modification using the SPRL bit of status register 1. If the sector protection register is locked, any attempt to issue an Unprotect command will be ignored, the integrated circuit will reset the WEL bit of status register 1 to logical “0” and return to the command code waiting state.

The timing diagram for the sector protection removal operation is shown in Figure 20.

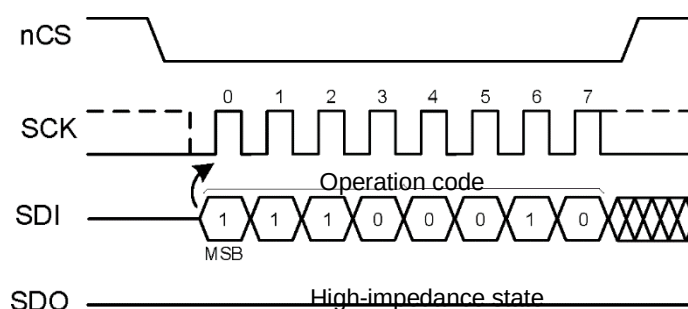


Figure 20 — Sector protection removal operation

6.21 Sector protection register read operation

The sector protection register can be read to determine the current protection status of flash memory sectors against programming and erase operations.

To read the sector protection register, the $\overline{\text{nCS}}$ line must be in a logical “0” state and the command code (1 byte: E0h) loaded into the integrated circuit. After loading the command code, the integrated circuit begins outputting data on the SDO line with the BP bits of the protection register. The most significant 2 bits of data in the byte are insignificant, as the protection register is 6-bit. Table 3 shows the possible values of the BP bits.

Transitioning the $\overline{\text{nCS}}$ line to logical “1” interrupts the read operation and puts the SDO line into a high-impedance state. The $\overline{\text{nCS}}$ line can transition to an inactive state at any time; reading a full byte is not required.

In addition to the sector protection register read operation, the SWP bits in status register 1 allow determining that all sectors, some, or none of the sectors are protected from modification.

The timing diagram for the sector protection register status read operation is shown in Figure 21.

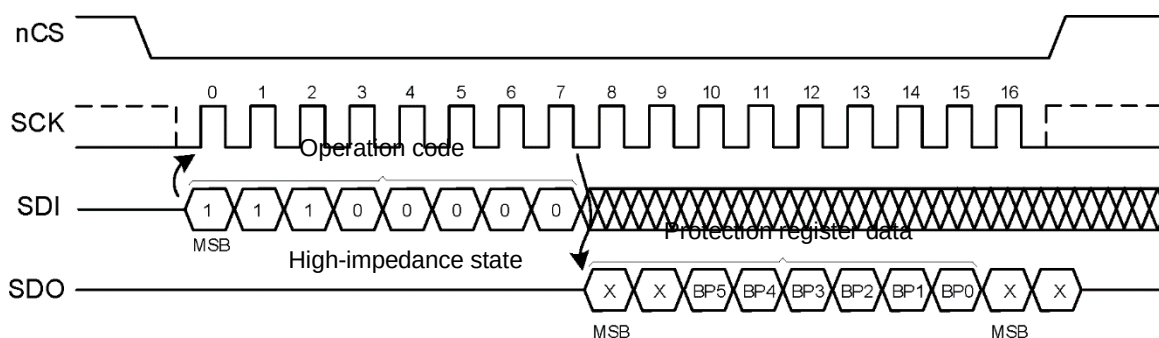


Figure 21 — Sector protection register read operation

6.22 Status register 1 and 2 read operation

6.22.1 Status register description

Status registers 1 and 2 are designed to determine the integrated circuit's busy/ready state, sector protection status, and many other functions. The registers can be read at any time, even during internal programming and erase operations.

To read status register 1, the $\overline{\text{nCS}}$ line must be set to logical “0” and the command code (1 byte: 05h) must be transmitted to the integrated circuit, and for reading status register 2 — the command code (1 byte: 07h). After receiving the command code, the integrated circuit outputs the data of the corresponding status register on the SDO line. After the byte is unloaded, data

Description for Flash-type EEPROM IC with SPI

output repeats as long as the nCS line remains at logical “0” and pulses are present on the SCK pin. Status register data is constantly updated during internal operations, so re-reading will result in new data being output.

The timing diagram for the status register read operation is shown in Figure 22.

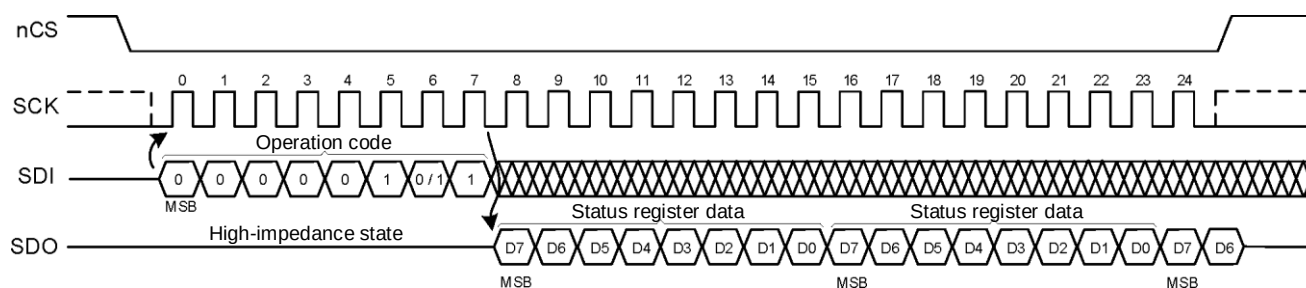


Figure 22 — Status register read operation

Transitioning the nCS line to logical “1” interrupts the status register read operation and puts the SDO line into a high-impedance state. The nCS line can change state at any time; reading a full byte of data is not required.

The assignment of bits for status registers 1 and 2 is given in Tables 7 and 8.

Table 7 — Status register 1 bit assignment

Bit number	Bit name		Access	Description	
7	SPRL	Sector protection register lock bit	R/W	0	Sector protection register not locked (default)
				1	Sector protection register locked for changes
6	QE	Quad SPI configuration bit	R/W	0	Quad SPI commands unavailable (default upon delivery). nWP and nHOLD input functionality active
				1	Quad SPI commands available. nWP and nHOLD input functionality replaced by IO2 and IO3
5:4	-	Reserved	R	0	Reserved
3:2	SWP	Sector protection status bits	R	00	All sectors unprotected (default upon delivery)
				01	Some sectors are protected. Reading the sector protection register will allow determining which specific sectors are protected
				10	Reserved for future use
				11	All sectors protected
1	WEL	Write enable bit	R	0	Integrated circuit not available for writing (default)
				1	Integrated circuit available for writing
0	BUSY	IC ready status bit	R	0	Integrated circuit ready
				1	Integrated circuit busy with internal operation

Notes

- Only bits 7 and 6 of status register 1 can be modified by the SR1Write command.
- Access designation:
R/W — read and write;
R — read only

Table 8 — Status register 2 bit assignment

Bit number	Bit name		Access	Description	
7	-	Reserved	R	0	Reserved
6	E_ERR	Erase error bit	R	0	No erase errors detected
				1	Erase error detected
5	P_ERR	Programming error bit	R	0	No programming errors detected
				1	Programming error detected
4	WPP	nWP pin status bit	R	0	Active “0” on nWP input. BPR register locked for protection removal
				1	Inactive “1” on nWP input. BPR register unlocked
3	APS	Attempt to erase/write to protected sector bit	R	0	No attempt to erase/write to a locked or busy sector detected
				1	Attempt to erase/write to a locked or busy sector detected
2	-	Reserved	R	0	Reserved
1	ES	Erase suspend bit	R	0	No blocks with suspended erase
				1	Erase suspended in one of the blocks
0	PS	Program suspend bit	R	0	No sectors with suspended programming
				1	Programming suspended in one of the sectors
Note — Access designation: R — read only					

6.22.2 SPRL bit

The SPRL bit is used to control the modification of the bits in the non-volatile sector protection register. If SPRL is in a logical “1” state, the protection register cannot be modified by Protect and Unprotect commands (the integrated circuit ignores these commands). If SPRL is in a logical “0” state, the sector protection register can be modified. After power-up, the SPRL bit is by default in a logical “0” state. The Reset command does not affect the state of the SPRL bit. The SR1Write command is used to change the state of the SPRL bit.

6.22.3 QE bit

The QE bit is used to change the functionality of the nWP and nHOLD pins. When this bit is set to “1”, it expands the device's data bus to 4-bit — i.e., it switches to Quad SPI mode. The nWP and nHOLD pins lose their primary functionality and become data input/output signals IO2 and IO3, respectively. All commands except QIProgram and QORRead still function without changes. The QE bit must be set to “1” when using QIProgram and QORRead commands. If QE is in the “0” state, QIProgram and QORRead commands are unavailable (the integrated circuit ignores these commands). The QE bit is stored in the non-volatile configuration register and retains its state after power-down. By default, the QE bit is in the “0” state upon delivery. The SR1Write command is used to change the state of the QE bit. The Reset command does not change the state of the QE bit.

6.22.4 SWP bits

The SWP bits indicate the protection status of the integrated circuit's sectors. Three-bit combinations are possible, indicating that the integrated circuit is unprotected from programming/erasing, partially protected, or fully protected. If the SWP bits indicate that the integrated circuit has protected sectors, then reading the BP bits of the sector protection register using the ProtectRead command is required to identify these sectors.

6.22.5 WEL bit

The WEL bit indicates the current status of the internal write enable state. When this bit is in a logical "0" state, the memory module is not available for commands such as Program, DIProgram, QIProgram, SErase, BErase, CErase, Protect, Unprotect, SR1Write. After power-up, the WEL bit is in a logical "0" state. The WEL bit can be automatically reset to "0" under the following conditions:

- successful completion of the Write Disable command;
- after a correct Reset command is issued;
- after a correct Program/DIProgram/QIProgram buffer programming command is issued;
- after a correct SErase/BErase/CErase erase command is issued;
- after a correct Protect/Unprotect command is issued;
- after a correct SR1Write command is issued.

6.22.6 BUSY bit

The BUSY bit is used to determine the status of an internal program/erase operation, sector protection/unprotection, or status register write. A logical "0" state of the BUSY bit indicates that all internal operations are complete, and the integrated circuit is in a command code waiting state. A logical "1" state of the BUSY bit indicates that the integrated circuit is busy with an internal operation.

6.22.7 E_ERR bit

The E_ERR bit indicates whether the last erase, sector unprotection, or status register write operation completed successfully. If at least one bit was not erased during these operations, the E_ERR bit is set to "1". If the operation was interrupted by a Reset command, the E_ERR bit is also set to "1". The E_ERR bit is not set to one if the operation is interrupted while attempting to erase a protected area, or if the WEL bit was not set before the operation. The E_ERR bit is cleared before the start of each erase, sector unprotection, and status register write operation.

6.22.8 P_ERR bit

The P_ERR bit indicates whether the last programming, sector protection, or status register write operation completed successfully. If at least one bit during these operations does not match the expected value, the P_ERR bit is set to "1". If the operation was interrupted by a Reset command, the P_ERR bit is also set to "1". The P_ERR bit is not set to "1" if the operation is interrupted while attempting to write data to a protected area, or if the WEL bit was not set before the operation. The P_ERR bit is cleared before the start of each programming, sector protection, and status register write operation.

6.22.9 WPP bit

The WPP bit indicates the state of the nWP pin. If nWP is in a logical “0” state, then this bit will be in a “0” state and vice versa. If Quad SPI mode is active and the QE bit is set to “1”, the WPP bit is set to a logical “1” state.

6.22.10 APS bit

The APS bit signals an attempt to program or erase a protected memory area, as well as an attempt to change the state of the BP bits of the non-volatile sector protection register, with sector protection already set. The APS bit is cleared before the start of each programming, erase, or sector protection operation. The APS bit is set to a logical “1” state under the following conditions:

- attempt to erase a block with protected sectors using the BErase command;
- attempt to erase a sector that falls within the range of protected sectors using the SErase command;
- attempt to erase the entire memory array containing blocks with protected sectors using the CErase command;
- attempt to program data into a page that falls within the range of protected sectors using Program, DIProgram, QIProgram commands;
- attempt to set sector protection using the Protect command when protection is already set for some range of sectors (BP bits of the sector protection register are not all equal to “0”).

6.22.11 ES and PS bits

The ES bit indicates the presence of a suspended erase operation in the integrated circuit, the PS bit indicates a suspended programming operation.

6.23 Status register 1 write operation

This operation is necessary to modify the SPRL and QE bits of Status Register 1. Before executing the SR1Write command, the WriteEn command must be issued to set the WEL bit of Status Register 1 to a logical “1”.

When executing the SR1Write command, the nCS line must be in a logical “0” state and the command code 01h must be loaded into the integrated circuit along with one data byte. Only bits 7 and 6 of the data byte are recognized by the memory module; the remaining bits can take any value. Any additional data bytes sent to the memory module after this are ignored. After transitioning the nCS line to a logical “1” state, several scenarios are possible.

If only the SPRL bit is modified, the WEL bit of the status register is reset to a logical “0” state, the SPRL bit is set to the new state, and no waiting for operation completion is required.

If both the SPRL and QE bits are modified, the WEL bit of the status register is reset to a logical “0” state, the SPRL bit is set to the new state, and a program/erase cycle is initiated to write the new state of the QE bit to the non-volatile configuration register. The waiting time for the QE bit write operation to complete is at least $t_{CYW(NVR)}$. During the writing of this bit, it is recommended to use the Status Register 1 read operation, which shows the status of the internal operation. This will reduce the waiting time for the operation to complete.

The IC performs an error detection algorithm when writing the state of the QE bit: if, after the operation, the non-volatile configuration register cell does not contain the expected value “0”, the E_ERR bit of Status Register 2 is set; if the cell does not contain the expected value “1”, the P_ERR bit of Status Register 2 is set.

The full data byte must be loaded into the memory module before changing the nCS line state to a logical “1”, i.e., at the byte boundary (multiple of 8 SCK clocks), otherwise the integrated circuit will abort the operation, the state of the SPRL and QE bits will not change, and the value of the WEL bit of Status Register 1 will also not change.

The timing diagram for the Status Register 1 write operation is shown in Figure 23.

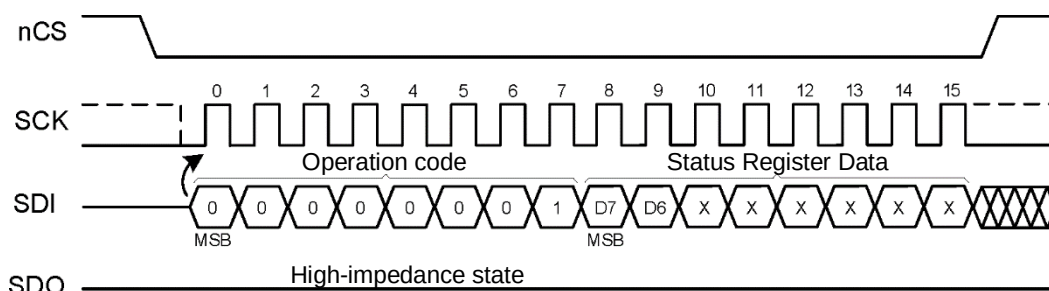


Figure 23 — Status register 1 write operation

6.24 Error Correcting Code (ECC) status register read operation

6.24.1 Status register description

The ECC status register is used to determine the results of the error correcting code block operation. The register can be read at any time, even during internal programming and erase operations.

To read the ECC status register, the nCS line must be set to a logical “0” state and the command code (1 byte: 18h) must be transmitted to the integrated circuit. After receiving the command code, the integrated circuit outputs data on the SDO line at each SCK clock cycle. After the byte is unloaded, data output is repeated until the nCS line transitions to a logical “1” state and pulses are present on the SCK pin. The status register data is constantly updated during internal operations, so re-reading will result in new data.

The timing diagram for the status register read operation is shown in Figure 24.

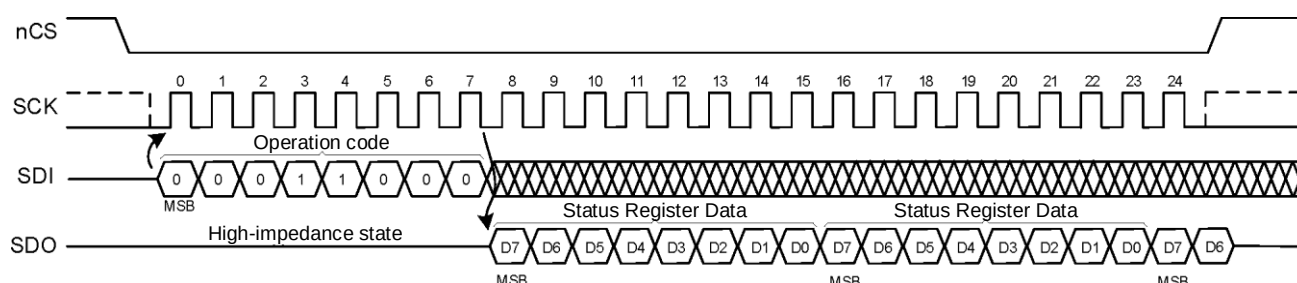


Figure 24 — Error Correcting Code status register read operation

Transitioning the nCS line to “1” interrupts the status register read operation and puts the SDO line into a high-impedance state. The nCS line state can be changed at any time; reading a full data byte is not required.

The assignment of ECC status register bits is given in Table 9.

Table 9 — ECC Status register bit assignment

Bit number	Bit Name		Access	Description	
7:5	-	Reserved	R	0	Reserved
4	DED_V	Double error detection bit during verification	R	0	No double errors detected during the last programming or erase or sector protection/unprotection operation
				1	Double error detected during the last programming or erase or sector protection/unprotection operation
3	SEC_V	Single error detection bit during verification	R	0	No single errors detected during the last programming or erase or sector protection/unprotection operation
				1	Single error detected and corrected during the last programming or erase or sector protection/unprotection operation
2	DED_R	Double error detection bit during read	R	0	No double errors detected during read
				1	Double error detected during the last read operation
1	SEC_R	Single error detection bit during read	R	0	No single errors detected during read
				1	A single error was detected and corrected during the last read operation.
0	ECC_EN	Error detection and correction enable bit	R	1	Error detection and correction enabled
Note — Access designation: R — read only					

6.24.2 DED_V bit

The DED_V bit signals the detection of a double error during the execution of the last initiated program/erase command or sector protection/unprotection (Program, DIProgram, QIProgram, SErase, BErase, CErase, CBErase, Protect, Unprotect). Double error detection means that the data during the last initiated program/erase operation or sector protection/unprotection was not corrected by the error correcting code block, since the built-in SECDED algorithm only allows correction of single errors.

6.24.3 SEC_V bit

The SEC_V bit signals the detection and correction of a single error during the execution of the last initiated program/erase command or sector protection/unprotection (Program, DIProgram, QIProgram, SErase, BErase, CErase, CBErase, Protect, Unprotect). Single error detection and correction means that the data during the last initiated program/erase operation or sector protection/unprotection was corrected by the error correcting code block.

6.24.4 DED_R bit

The DED_R bit signals the detection of a double error during the last read operation or sector protection register read (Read, FRead, DORed, QORed, ProtectRead). Double error detection means that the data during reading was not corrected by the error correcting code block, since the built-in SECDED algorithm only allows correction of single errors.

6.24.5 SEC_R bit

The SEC_R bit signals the detection and correction of a single error during the last read operation or sector protection register read (Read, FRead, DORed, QORed, ProtectRead). Single error detection and correction means that the data during reading was corrected by the error correcting code block.

6.24.6 ECC_EN bit

The ECC_EN bit is the enable bit for error detection and correction during program/erase operations, sector protection/unprotection, read, and sector protection register read operations. In this version of the integrated circuit, this bit is always in a logical “1” state, so the error correcting code block performs double error detection and single error correction during the aforementioned operations.

6.25 Reset operation

In some applications, it is necessary to prematurely interrupt a programming or erase operation. The Reset command allows immediate interruption of any ongoing operation and returns the integrated circuit to a command code waiting state. For this, there is no need to transmit the WriteEn command before issuing the Reset command. That is, the Reset command does not depend on the state of the WEL bit of Status Register 1.

When executing the Reset command, the nCS line must be in a logical “0” state, and the command code F0h must be loaded into the memory module. Address bytes are not transmitted with this command, but an acknowledgment byte D0h must be transmitted immediately after the operation code. Any additional bytes transmitted to the memory module after the acknowledgment byte are ignored. When the nCS line transitions to a logical “1” state, the currently executing operation is interrupted. The integrated circuit will transition to a command code waiting state within no more than 40 μ s when interrupting a programming or sector erase operation, and within no more than 180 μ s when interrupting a block or full integrated circuit erase operation. If a programming or erase operation is interrupted in this manner, its correct execution is not guaranteed.

It is not recommended to interrupt program/erase operations, as this may reduce the number of possible rewrite cycles and decrease the integrated circuit's lifespan.

The Reset command does not affect the bits of the non-volatile sector protection register, the SPRL and QE bits of Status Register 1, or the ABE, ABSD, ABSA bits of the AutoBoot register. However, the WEL, PS, and ES bits will be reset to their default state. If a reset operation is performed during an erase suspension, the suspended operation is interrupted, and the content of the erased area remains in an undefined state. If a reset operation is performed during a programming suspension, the suspended operation is interrupted, and the content of the page being programmed will remain undefined. The other pages in the sector where this page is located will retain their previous content. If a reset operation is performed when neither

programming nor erase operations are active, the integrated circuit will reconfigure and become available within 2.5 μ s.

The command code and acknowledgment byte must be fully loaded into the memory module before changing the nCS line to a logical “1” state, i.e., at the byte boundary (multiple of 8 SCK clocks), otherwise the reset operation will not be performed.

The timing diagram for the reset operation is shown in Figure 25.

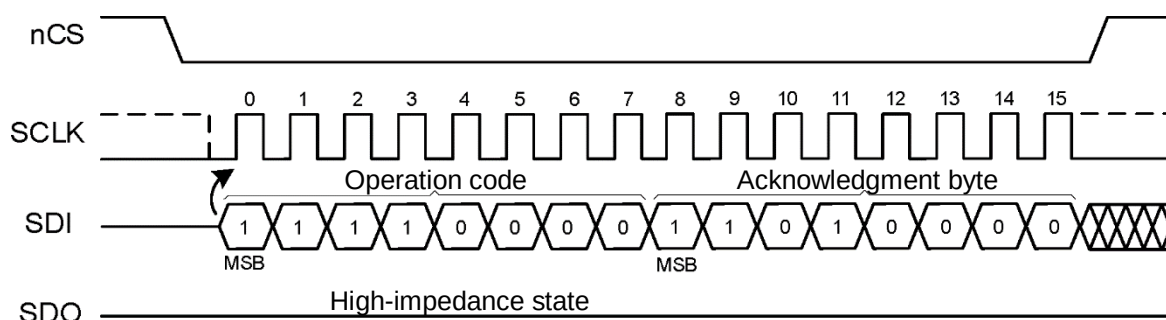


Figure 25 — Reset operation

6.26 IC identification information read operation

The IDRead command is used to read identification information. To perform this operation, the nCS line must be set to a logical “0” state and the command code 9Fh must be issued. After this, the integrated circuit begins outputting identification data on the SDO line at each SCK clock cycle. The first byte is the manufacturer identification code, the next byte is the integrated circuit identification code. After this, data output is repeated until the nCS line transitions to a logical “1” state, which terminates the read operation and puts the SDO line into a high-impedance state. The nCS line can be changed at any time, meaning that reading a full byte is not required.

The values of the identification information bytes are given in Table 10.

Table 10 — Identification Information Byte Values

Byte number	Type	Meaning
1	Manufacturer identification code	01h
2	Integrated circuit identification code	DCh

The timing diagram for the identification information read operation is shown in Figure 26.

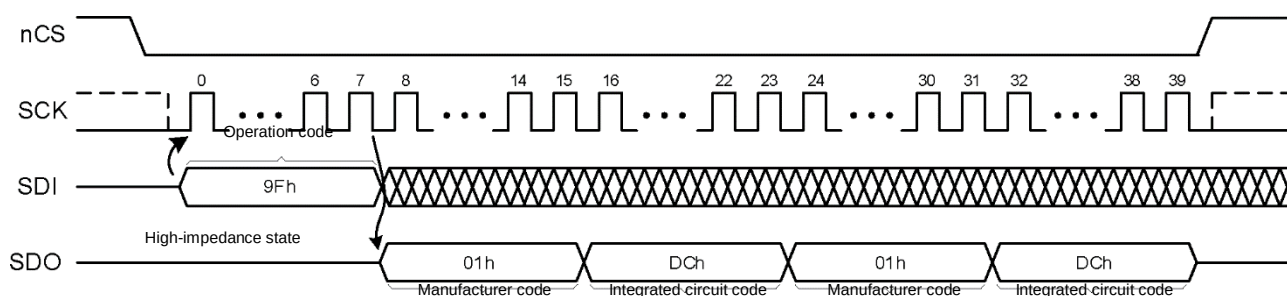


Figure 26 — IC identification information read operation

6.27 Integrated circuit SFDP parameters read operation

The Serial Flash Discoverable Parameter (SFDP) standard provides a serial method for describing the functional capabilities of serial flash devices in a standard set of internal parameter tables. These parameter tables can be queried by host system software to make the necessary adjustments for accommodating various functions from different manufacturers. SFDP is a JEDEC standard, JESD216. The concept is similar to that outlined in the introduction to the JEDEC JESD68 standard for CFI (Common Flash Interface).

The SFDPRead command with code 5Ah is used to read data from the built-in SFDP table at frequencies up to 100 MHz. This command sequence is the same as for a read operation with the FRead command: the nCS line must be set to a logical “0” state, and the command code (1 byte: 5Ah) must be transmitted to the integrated circuit. Following the command code transmission, a 3-byte address must be sent, specifying the starting address of the first word to be read within the SFDP table (from 00h to 4Fh). Following the address bytes, one dummy data byte must be transmitted.

After transmitting the entire required command sequence, data is output on the SDO line. Data is transmitted byte by byte, starting from the least significant. Each data byte is output on the bus with the most significant bit first (MSB). If the last word (address 4Fh) of the SFDP table is read, the integrated circuit continues reading, but the output will be unknown data. When the nCS line is set to a logical “1” state, the read operation terminates, and the SDO line transitions to a high-impedance state. It is recommended to terminate the read operation at the byte boundary.

The timing diagram for the SFDP read operation is shown in Figure 27.

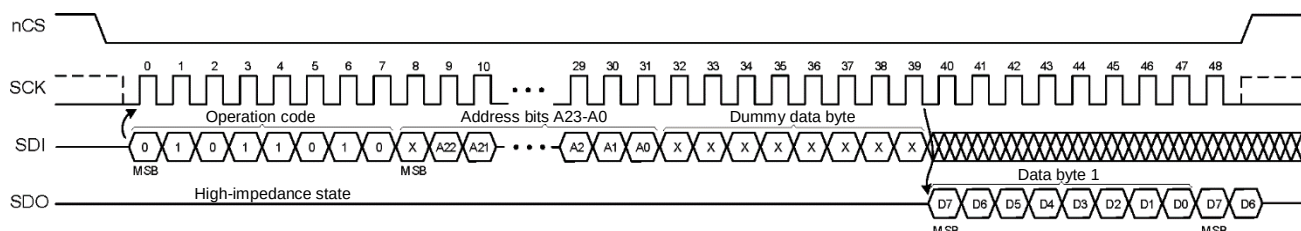


Figure 27 — Integrated circuit SFDP parameters read operation

The SFDP parameter table data is provided in Table 11.

Table 11 — Serial Flash Discoverable Parameter (SFDP) table

Address byte	Address bit	Data	Description
SFDP Table Header: 1st DWORD (SFDP Header: 1st DWORD)			
00h	A7:A0	53h	SFDP abbreviation in ASCII format ASCII “S”
01h	A15:A8	46h	ASCII “F”
02h	A23:A16	44h	ASCII “D”
03h	A31:A24	50h	ASCII “P”
SFDP table header: 2nd DWORD (SFDP header: 2nd DWORD)			
04h	A7:A0	06h	SFDP version: minor number 06h — JEDEC JESD216 Revision B

Address byte	Address bit	Data	Description
05h	A15:A8	01h	SFDP version: major number 01h — version number compatible with any software using the SFDP table
06h	A23:A16	00h	Number of flash device parameter table headers 00h — one flash device parameter table header
07h	A31:A24	FFh	Not used FFh — default
Parameter table header: 1st DWORD (Parameter header: 1st DWORD)			
08h	A7:A0	00h	Identification number 00h — indicates that the parameter table is defined by JEDEC JC-42.4
09h	A15:A8	06h	SFDP table minor version number 06h — JESD216 revision B
0Ah	A23:A16	01h	SFDP table major version number 01h — version number compatible with any software using the SFDP table
0Bh	A31:A24	10h	Length of parameter table in words 10h — 16 words, each word 4 data bytes
Parameter table header: 2nd DWORD (Parameter header: 2nd DWORD)			
0Ch	A7:A0	10h	Pointer to the start address of the parameter table (3 bytes) 10h — 0th byte of the parameter table start address
0Dh	A15:A8	00h	00h — 1st byte of the parameter table start address
0Eh	A23:A16	00h	00h — 2nd byte of the parameter table start address
0Fh	A31:A24	FFh	Not used FFh — default
Parameter table: 1st word (JEDEC basic flash parameter table: 1st DWORD)			
10h	A1:A0	FFh	Erasable block/sector size 11b — for flash devices that do not support 4 Kbyte block/sector erase
	A2		Write granularity 1b — for flash devices with 64 bytes or more buffer programming
	A3		Write enable instruction required to write to the volatile status register 1b — for flash devices that require writing 00h to the status register to enable programming/erase operations
	A4		Write enable command code for writing to the volatile status register 1b — code — 06h (if bit A3 is set to 1) Note — If the flash memory status register is non-volatile, bits A3 and A4 are set to 00b
	A7:A5		Not used 111b — default
11h	A15:A8	FFh	4 Kbyte block/sector erase command code FFh — for flash devices that do not support 4 Kbyte block/sector erase

Address byte	Address bit	Data	Description
12h	A16	C1h	(1-1-2) Fast read support 1b — for flash devices supporting (1-1-2) fast read Notes 1 (1-1-2) — means that the command code is set only on the 1st input line (SDI), the address is also set only on the 1st input line (SDI), and data is output on 2 output lines (SDI and SDO). 2 Fast Read — is reading at a higher frequency implemented by pipelining output data during N wait cycles after the command code and address are provided
	A18:A17		Number of address bytes Number of bytes used when addressing the flash memory cell array during read, program, and erase operations. 00b: 3 address bytes
	A19		Support for data transfer on clock signal rising and falling edges 0b — for flash devices that do not support data transfer on clock signal rising and falling edges
	A20		(1-2-2) Fast read support 0b — for flash devices that do not support (1-2-2) fast read Note — (1-2-2) — means that the command code is set on the 1st input line (SDI), the address is set on 2 input lines (SDI and SDO), and data is also output on 2 output lines (SDI and SDO)
	A21		(1-4-4) Fast read support 0b — for flash devices that do not support (1-4-4) fast read Note — (1-4-4) — means that the command code is set on the 1st input line (SDI), the address is set on 4 input lines (SDI, SDO, nWP, nHOLD), and data is also output on 4 output lines (SDI, SDO, nWP, nHOLD)
	A22		(1-1-4) Fast read support 1 — for flash devices supporting (1-1-4) fast read Note — (1-1-4) — means that the command code is set on the 1st input line (SDI), the address is also set on the 1st input line (SDI), and data is output on 4 output lines (SDI, SDO, nWP, nHOLD)
	A23		Not used 1b — default
13h	A31:A24	FFh	Not used FFh — default
Parameter table: 2nd word (JEDEC basic flash parameter table: 2nd DWORD)			
14h	A7:A0	FFh	Flash information capacity 03FFFFFFh — 64 Mbit
15h	A15:A8	FFh	
16h	A23:A16	FFh	
17h	A31:A24	03h	
Parameter table: 3rd word (JEDEC basic flash parameter table: 3rd DWORD)			

Description for Flash-type EEPROM IC with SPI

Address byte	Address bit	Data	Description
18h	A4:A0	00h	(1-4-4) fast read. Required number of wait cycles until valid data appears 00000b — no wait cycles are required, as (1-4-4) fast read is not supported
	A7:A5		(1-4-4) fast read. Number of SCK cycles required to transmit mode bits 000b — SCK cycles for mode bit transmission are not required, as (1-4-4) fast read is not supported
19h	A15:A8	FFh	(1-4-4) fast read. Command code FFh — (1-4-4) fast read is not supported
1Ah	A20:A16	08h	(1-1-4) fast read. Required number of wait cycles until valid data appears 01000b — 8 wait cycles are required until valid data appears
	A23:A21		(1-1-4) fast read. Number of SCK cycles required to transmit mode bits 000b — mode bit transmission is not supported
1Bh	A31:A24	6Bh	(1-1-4) fast read. Command code 6Bh — (1-1-4) fast read command code
Parameter table: 4th word (JEDEC basic flash parameter table: 4th DWORD)			
1Ch	A4:A0	08h	(1-1-2) fast read. Required number of wait cycles until valid data appears 01000b — 8 wait cycles are required until valid data appears
	A7:A5		(1-1-2) fast read. Number of SCK cycles required to transmit mode bits 000b — mode bit transmission is not supported
1Dh	A15:A8	3Bh	(1-1-2) fast read. Command code 3Bh — (1-1-2) fast read command code
1Eh	A20:A16	00h	(1-2-2) fast read. Required number of wait cycles until valid data appears 00000b — no wait cycles are required, as (1-2-2) fast read is not supported
	A23:A21		(1-2-2) fast read. Number of SCK cycles required to transmit mode bits 000b — SCK cycles for mode bit transmission are not required, as (1-2-2) fast read is not supported
1Fh	A31:A24	FFh	(1-2-2) fast read. Command code FFh — (1-2-2) fast read is not supported
Parameter table: 5th word (JEDEC basic flash parameter table: 5th DWORD)			
20h	A0	EEh	(2-2-2) Fast Read support 0b — for flash devices that do not support (2-2-2) fast read. Note — (2-2-2) — means that the command code is set on 2 input lines (SDI and SDO), the address is set on 2 input lines (SDI and SDO), and data is also output on 2 output lines (SDI and SDO)
	A3:A1		Not used 111b — default

Description for Flash-type EEPROM IC with SPI

Address byte	Address bit	Data	Description
	A4		(4-4-4) Fast Read support 0b: For flash devices NOT supporting (4-4-4) fast read. Note — (4-4-4) — means that the command code is set on 4 input lines (SDI, SDO, nWP, nHOLD), the address is also set on 4 input lines (SDI, SDO, nWP, nHOLD), and data is output on 4 output lines (SDI, SDO, nWP, nHOLD)
	A7:A5		Not used 111b — default
21h	A15:A8	FFh	Not used FFh — default
22h	A23:A16	FFh	Not used FFh — default
23h	A31:A24	FFh	Not used FFh — default
Parameter table: 6th word (JEDEC basic flash parameter table: 6th DWORD)			
24h	A7:A0	FFh	Not used FFh — default
25h	A15:A8	FFh	Not used FFh — default
26h	A20:A16	00h	(2-2-2) fast read. Required number of wait cycles until valid data appears 00000b — no wait cycles are required, as (2-2-2) fast read is not supported
	A23:A21		(2-2-2) fast read. Number of SCK cycles required to transmit mode bits 000b — SCK cycles for mode bit transmission are not required, as (2-2-2) fast read is not supported
27h	A31:A24	FFh	(2-2-2) fast read. Command code FFh — (2-2-2) fast read is not supported
Parameter table: 7th word (JEDEC basic flash parameter table: 7th DWORD)			
28h	A7:A0	FFh	Not used FFh — default
29h	A15:A8	FFh	Not used FFh — default
2Ah	A20:A16	00h	(4-4-4) fast read. Required number of wait cycles until valid data appears 00000b — no wait cycles are required, as (4-4-4) fast read is not supported
	A23:A21		(4-4-4) fast read. Number of SCK cycles required to transmit mode bits 000b — SCK cycles for mode bit transmission are not required, as (4-4-4) fast read is not supported
2Bh	A31:A24	FFh	(4-4-4) fast read. Command code FFh — (4-4-4) fast read is not supported

Address byte	Address bit	Data	Description
Parameter table: 8th word (JEDEC basic flash parameter table: 8th DWORD)			
2Ch	A7:A0	0Dh	Erasable region 1. Size Number of bytes. Specified as a power of 2. 0Dh — erasable region size — 8 Kbyte sector (2^{13} bytes)
2Dh	A15:A8	20h	Erasable region 1. Command code 20h — 8 Kbyte sector erase command code
2Eh	A23:A16	15h	Erasable region 2. Size Number of bytes. Specified as a power of 2. 15h — erasable region size — 2 Mbyte block (2^{21} bytes)
2Fh	A31:A24	D8h	Erasable region 2. Command code D8h — 2 Mbyte block erase command code
Parameter table: 9th word (JEDEC basic flash parameter table: 9th DWORD)			
30h	A7:A0	00h	Erasable region 3. Size Number of bytes. Specified as a power of 2. 00h: region does not exist
31h	A15:A8	FFh	Erasable region 3. Command code FFh: region does not exist
32h	A23:A16	00h	Erasable region 4. Size Number of bytes. Specified as a power of 2. 00h — region does not exist
33h	A31:A24	FFh	Erasable region 4. Command code FFh — region does not exist
Parameter table: 10th word (JEDEC basic flash parameter table: 10th DWORD)			
34h	A3:A0	F0h	Ratio of typical erase time to maximum erase time. The coefficient is calculated by the formula $N = T_{MAX}/(2 \cdot T_{TYP}) - 1$, where T_{MAX} — maximum erase time; T_{TYP} — typical erase time. 0000b — $N = 0$ — maximum erase time is 2 times the typical erase time
	A7:A4		Erasable region 1. Typical erase time
35h	A10:A8	18h	The typical erase time is given by the formula $T_{TYP} = (\text{Count} + 1) \cdot \text{Units}$, where $\text{Count} = \text{A8:A4}$ — value from 0 to 31; $\text{Units} = \text{A10:A9}$ — units of time measurement (00b — 1 ms; 01b — 16 ms; 10b — 128 ms; 11b — 1 s). $\text{A8:A4} = 01111b$ — $\text{Count} = 15$; $\text{A10:A9} = 00b$ — $\text{Units} = 1$ ms Typical erase time for an 8 Kbyte sector $(15 + 1) \cdot 1 \text{ ms} = 16 \text{ ms}$
	A15:A11		Erasable region 2. Typical erase time The typical erase time is given by the formula $T_{TYP} = (\text{Count} + 1) \cdot \text{Units}$, where $\text{Count} = \text{A15:A11}$ — value from 0 to 31;

Address byte	Address bit	Data	Description
36h	A17:A16	01h	Units = A17:A16 — units of time measurement (00b — 1 ms; 01b — 16 ms; 10b — 128 ms; 11b — 1 s). A15:A11 = 00011b — Count = 3; A17:A16 = 01b — Units = 16 ms Typical erase time for a 2 Mbyte block (3 + 1)·16 ms = 64 ms
	A23:A18		Erasable region 3. Typical erase time
37h	A24	00h	A24:A18 = 0000000b — region does not exist
	A31:A25		Erasable region 4. Typical erase time A31:A25 = 0000000b — region does not exist
Parameter table: 11th word (JEDEC basic flash parameter table: 11th DWORD)			
38h	A3:A0	90h	Ratio of typical programming time to maximum programming time The coefficient is calculated by the formula $N = T_{\text{MAX}} / (2 \cdot T_{\text{TYP}} - 1)$, where T_{MAX} — maximum programming time; T_{TYP} — typical programming time. 0000b — N = 0 — maximum programming time is 2 times the typical programming time
	A7:A4		Page size Specified as a power of 2. 1001b — programmable region size — 512 byte page (2 ⁹ bytes)
39h	A13:A8	39h	Typical page programming time. The typical page programming time is given by the formula $T_{\text{TYP}} = (\text{Count} + 1) \cdot \text{Units}$, where Count = A12:A8 — value from 0 to 31; Units = A13 — units of time measurement (0b — 8 μs; 1b — 64 μs). A12:A8 = 11001b — Count = 25; A13 = 1b — Units = 64 μs. Typical programming time for a 512 byte page is given by the formula $T_{\text{TYP}} = (25 + 1) \cdot 64 \mu\text{s} = 1664 \mu\text{s}$
	A15:A14		Typical programming time for the first byte
3Ah	A18:A16	00h	A18:A14 = 00000b — byte programming is not supported by the flash device
	A23:A19		Typical programming time for an additional byte A23:A19 = 00000b — byte programming is not supported by the flash device
3Bh	A30:A24	8Dh	Typical chip erase time The typical chip erase time is given by the formula $T_{\text{TYP}} = (\text{Count} + 1) \cdot \text{Unit}$, where Count = A28:A24 — value from 0 to 31; Units = A30:A29 — units of time measurement (00b — 16 ms; 01b — 256 ms; 10b — 4 s; 11b — 64 s). A28:A24 = 01101b — Count = 13; A30:A29 = 00b — Units = 16 ms. Typical chip erase time (13+1)·16 ms = 224 ms

Address byte	Address bit	Data	Description
	A31		Not used 1b — default
Parameter table: 12th word (JEDEC basic flash parameter table: 12th DWORD)			
3Ch	A3:A0	ECh	Forbidden operations during programming suspend A0=0b — erase operation is forbidden for any memory region; A1=0b — new page programming operation is forbidden for any memory region; A2=1b — read operation is forbidden only for the page where programming is suspended; A3=1b — no additional restrictions on programming and erase operations. A3:A0 = 1100b = Ch
	A7:A4		Forbidden operations during erase suspend A4=0b — new erase operation is forbidden for any memory region; A5=1b — page programming operation is forbidden only for the block where erase is suspended; A6=1b — read operation is forbidden only for the block where erase is suspended; A7=1b — no additional restrictions on programming and erase operations. A7:A4 = 1110b = Eh
3Dh	A8	C3h	Not used 1b — default
	A12:A9		Time interval between programming resume and next suspend The typical value of the time interval between programming resume and next suspend is given by the formula $T_{TYP} = (Count+1) \cdot 64 \mu s$, where Count = A12:A9 — value from 0 to 15. 0001b — Count = 1. Typical value of the time interval between programming resume and next suspend $(1+1) \cdot 64 \mu s = 128 \mu s$
	A15:A13		Programming suspend time
3Eh	A19:A16	18h	Maximum time required for the flash device to suspend the programming operation and be ready to accept allowed command codes. The maximum page programming suspend time is given by the formula $t_{SUSP_PR} = (Count+1) \cdot Unit$, where Count = A17:A13 — value from 0 to 31; Units = A19:A18 — units of time measurement (00b — 128 ns; 01b — 1 μs ; 10b — 8 μs ; 11b — 64 μs). A17:A13 = 00110b — Count = 6; A19:A18 = 10b — Units = 8 μs . Maximum page programming suspend time $(6+1) \cdot 8 \mu s = 56 \mu s$

Description for Flash-type EEPROM IC with SPI

Address byte	Address bit	Data	Description
	A23:A20		Time interval between erase resume and next suspend The typical value of the time interval between erase resume and next suspend is given by the formula $T_{TYP} = (Count+1) \cdot 64 \mu s$, where Count = A23:A20 — value from 0 to 15. A23:A20 = 0001b — Count = 1 Typical value of the time interval between erase resume and next suspend $(1+1) \cdot 64 \mu s = 128 \mu s$
3Fh	A30:A24	03h	Erase suspend time Maximum time required for the flash device to suspend the erase operation and be ready to accept allowed command codes. The maximum erase suspend time is given by the formula $t_{SUSP_ER} = (Count+1) \cdot Units$, where Count = A28:A24 — value from 0 to 31; Units = A30:A29 — units of time measurement (00b — 128 ns; 01b — 1 μs ; 10b — 8 μs ; 11b — 64 μs). A28:A24 = 00011b — Count = 3; A30:A29 = 00b — Units = 128 ns. Maximum sector/block erase suspend time $(3+1) \cdot 128 ns = 512 ns$
	A31		Support for program/erase suspend/resume operations 0b: For flash devices supporting suspend/resume operations
Parameter table: 13th word (JEDEC basic flash parameter table: 13th DWORD)			
40h	A7:A0	D0h	Programming resume command code
41h	A15:A8	B0h	Programming suspend command code
42h	A23:A16	D0h	Erase resume command code
43h	A31:A24	B0h	Erase suspend command code
Parameter table: 14th word (JEDEC basic flash parameter table: 14th DWORD)			
44h	A1:A0	F7h	Not used 11b — default
	A3:A2		Operation status tracking 01b — For flash devices supporting the legacy method of tracking operation status using instruction 05h and checking bit 0 (BUSY) of the status register: — if BUSY = 0 — flash device is ready; — if BUSY = 1 — flash device is busy performing an operation
	A7:A4		Not used 1111b — default

Address byte	Address bit	Data	Description
45h	A14:A8	A7h	Chip exit time from low power mode The chip exit time from low power mode is given by the formula $t_{SLP_OFF} = (Count+1) \cdot Units$, where Count = A12:A8 — value from 0 to 31; Units = A14:A13 — units of time measurement (00b — 128 ns; 01b — 1 μs; 10b — 8 μs; 11b — 64 μs). A12:A8 = 00111b — Count = 7; A14:A13 = 01b — Units = 1 μs. Chip exit time from low power consumption is given by the formula $t_{SLP_OFF} = (7+1) \cdot 1 \mu s = 8 \mu s$
	A15		Command code for exiting low power mode
46h	A22:A16	D5h	A22:A15 = 10101011b — ABh — command code for exiting low power mode
	A23		Command code for entering low power mode A30:A23 = 10111001b — B9h — command code for entering low power mode
47h	A30:A24	5Ch	Low power mode support 0b — for flash devices supporting low power mode
	A31		
Parameter table: 15th word (JEDEC basic flash parameter table: 15th DWORD)			
48h	A3:A0	00h	Disable 4-4-4 mode 0000b — 4-4-4 mode is not supported by the flash device
	A7:A4		Enable 4-4-4 mode A8:A4 = 00000b — 4-4-4 mode is not supported by the flash device
49h	A8	90h	0-4-4 mode support 0b — 0-4-4 mode is not supported by the flash device
	A9		
	A15:A10		Exit 0-4-4 mode 100100b — 0-4-4 mode is not supported by the flash device
4Ah	A19:A16	28h	Enter 0-4-4 mode 1000b — 0-4-4 mode is not supported by the flash device
	A22:A20		Requirements for enabling 1-1-4 (QUAD) mode This field indicates whether the flash device contains a QE bit used to enable 1-1-4 (QUAD) mode. If QE exists, this field also defines the bit's location and the method for setting/clearing the bit. 010b — QE is bit #6 of status register 1. It is set and cleared using the write to status register 1 (SR1Write) command. Writing a logical “1” to the QE bit enables 1-1-4 (QUAD) mode, writing a logical “0” disables 1-1-4 (QUAD) mode
	A23		Disable HOLD or Reset This field determines whether HOLD or RESET can be disabled via the configuration register. 0b — disabling HOLD or Reset is not supported by the flash device
4Bh	A31:A24	FFh	Not used FFh — default

Address byte	Address bit	Data	Description
Parameter table: 16th word (JEDEC basic flash parameter table: 16th DWORD)			
4Ch	A6:A0	F0h	Volatile or non-volatile status register 1 A4:A0 = 10000b — status register 1 contains both volatile and non-volatile bits. To write to status register 1, a write enable command (code — 06h) must be issued first; A6:A5 = 11b — not used
	A7		Not used 1b — default
4Dh	A13:A8	08h	Software reset and recovery sequence support This field indicates how to return the flash device to its default state. 001000b — issue a reset command (code — F0h)
	A15:A14		Exit 4-byte addressing mode A21:A14 = 00000000b — 4-byte addressing is not supported by the flash device
4Eh	A21:A16	C0h	Not used 11b: default
	A23:A22		
4Fh	A31:A24	80h	Enter 4-byte addressing mode 80h — 4-byte addressing is not supported by the flash device

6.28 Low power entry operation

The InSleep command is used to enter low power mode. When the device is in this mode, all commands except OutSleep will be ignored. Low power mode can be used as an additional protection mechanism against unintentional programming and erase operations.

To execute the InSleep command, the command code B9h must be transmitted to the integrated circuit. Any data after the command code transmission will be ignored. After the nCS line is set to logical “1”, the integrated circuit will enter low power mode within time t_{SLP_ON} . nCS must switch at a byte boundary (a multiple of 8 SCK clocks), otherwise the integrated circuit will abort the operation and return to standby mode.

The InSleep command will be ignored if an internal program/erase operation, sector protection set/clear, or status register 1 write is in progress.

The timing diagram for the low power entry operation is shown in Figure 28.

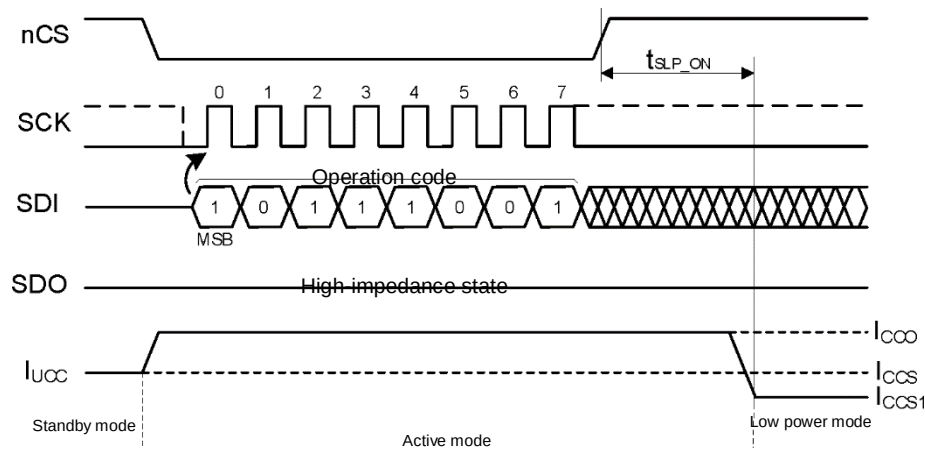


Figure 28 — Low power entry operation

6.29 Low power exit operation

To exit low power mode and resume normal device operation, the OutSleep command must be issued. The OutSleep command is the only command the device recognizes while in this mode.

To exit low power mode, the command code ABh must be transmitted to the integrated circuit. Any data after the command code transmission will be ignored. After the nCS line is set to logical “1”, the integrated circuit will exit low power mode and enter standby mode within time t_{SLP_OFF} . nCS must switch at a byte boundary (a multiple of 8 SCK clocks), otherwise the integrated circuit will abort the operation and remain in low power mode. After the device returns to standby mode, read, program, or erase operations become available.

The timing diagram for the low power exit operation is shown in Figure 29.

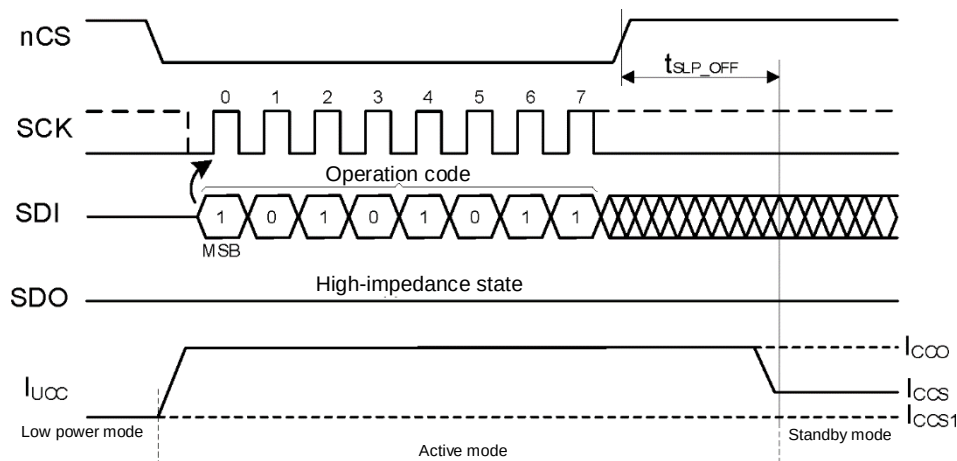


Figure 29 — Low power exit operation

6.30 Auto-boot mode

Auto-boot mode allows the host device to retrieve boot data from the memory integrated circuit without needing to send a read command. This saves 32 or more initial SCK clocks and simplifies the logic required to initiate boot data reading.

To initiate auto-boot mode, power must be reset or a Reset command issued, then the nCS line must be transitioned from logical “1” to “0”, and the SCK clock signal started. Data will start to be output from the address defined in the auto-boot register and with a delay also defined in this

register. By default, the minimum delay is 8 SCK clocks. Data loading can be performed at a frequency of up to 100 MHz.

Data is transmitted byte by byte, starting with the least significant. Each data byte is placed on the bus with the most significant bit first (MSB). The integrated circuit contains an internal address counter that automatically increments after each transmitted data byte. If the last word (address 7FFFFFFh) of the memory array is read, the integrated circuit continues reading from the beginning of the array (address 000000h). No delays occur during this process. Data will be continuously output until nCS switches to a logical “1” state. When the nCS line is set to “1”, the read operation stops, the integrated circuit exits auto-boot mode and enters standby mode, and the SDO line or the SDI/IO0, SDO/IO1, nWP/IO2, and nHOLD/IO3 lines (if QE=1) enter a high-impedance state. If the nCS line is set to “1” before the first data byte is transmitted, the integrated circuit will remain in auto-boot mode. After entering standby mode, the device is ready to receive all commands from Table 5. Auto-boot mode will only be re-initiated after a power reset or a Reset command.

If the QE bit of status register 1 is set to “1”, data will be read on the SDI/IO0, SDO/IO1, nWP/IO2, and nHOLD/IO3 lines, similar to the QORead command. On the first SCK clock, bit 7 of the first data byte will appear on the nHOLD/IO3 line, bit 6 of the same data byte will appear on the nWP/IO2 line, bit 5 on the SDO/IO1 line, and bit 4 on the SDI/IO0 line. On the next SCK clock, bits 3, 2, 1, and 0 of the first data byte will appear on the nHOLD/IO3, nWP/IO2, SDO/IO1, and SDI/IO0 outputs, respectively. Thus, each data byte will be output in 2 SCK clocks.

If the QE bit of status register 1 is set to “0”, data will be read only on the SDO line, similar to the FRead command.

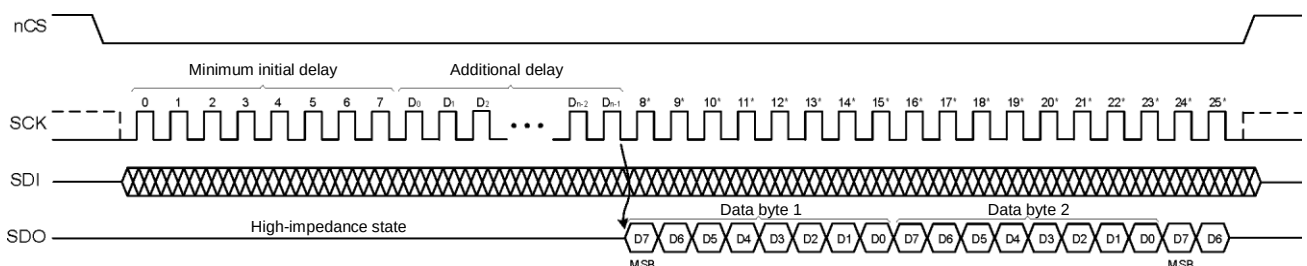


Figure 30 — Read in auto-boot mode, QE bit=0

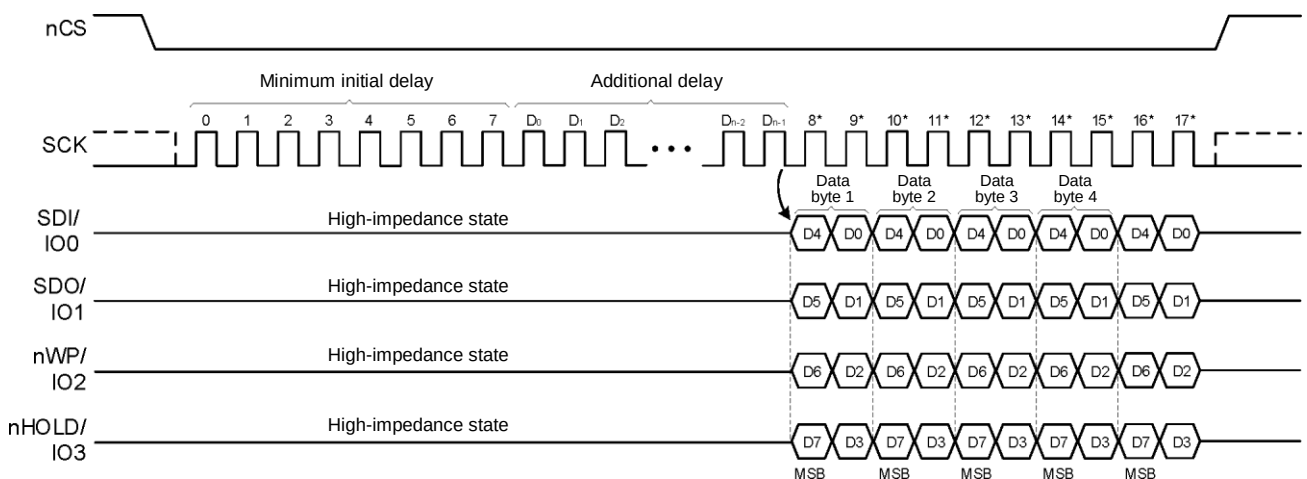


Figure 31 — Read in auto-boot mode, QE bit=1

6.31 Auto-boot configuration register read operation

6.31.1 Description of the auto-boot configuration register

The auto-boot register can be read to determine the activity of the auto-boot mode and its operational configuration. The register can be read at any time, even during internal programming and erase operations.

To read the auto-boot register, the nCS line must be set to logical “0” and the ABRRead command code (1 byte: 14h) must be transmitted to the integrated circuit. After receiving the command code, the integrated circuit begins outputting 3 bytes of auto-boot register data on the SDO line at each SCK clock cycle. After the 3 bytes are output, data output repeats as long as the nCS line remains in the “0” state and pulses are present on the SCK pin.

The timing diagram for the auto-boot configuration register read operation is shown in Figure 32.

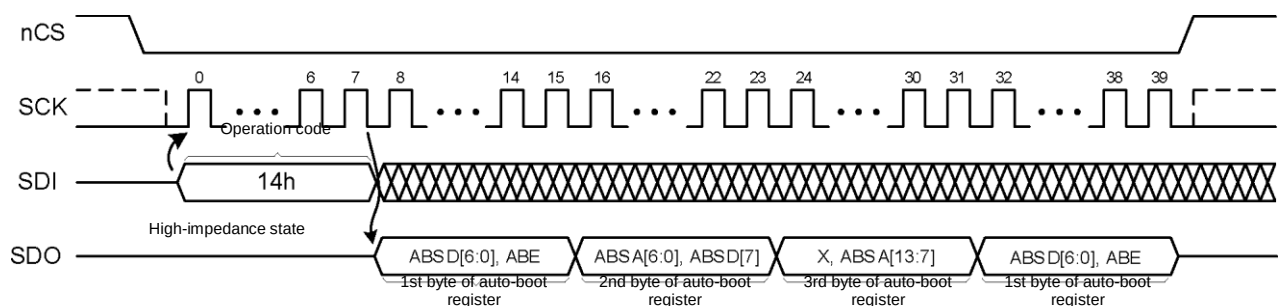


Figure 32 — Auto-boot configuration register read operation

Transitioning the nCS line to a logical “1” state interrupts the auto-boot configuration register read operation and places the SDO line in a high-impedance state. The nCS line state can change at any time; reading a full data byte is not required.

The assignment of bits in the auto-boot configuration register is given in Table 12.

Table 12 — AutoBoot register bit assignment

Bit number	Bit name	Access	Description	
23	-	-	Insignificant. Can take any value.	
22:9	ABSA[13:0]	R/W	Starting address for data reading, aligned to a 512-byte page boundary (A8 — A0 all 0). Sets higher addresses A22 — A9 (default address at delivery is 0000h)	
8:1	ABSD[7:0]	R/W	The value of the additional initial delay in SCK clocks, which will be added to the minimum 8 wait clocks. (by default, the additional delay at delivery is 00h)	
0	ABE	R/W	0	Auto-boot mode is disabled (default at delivery)
			1	Auto-boot mode is enabled
Note — Access designation: R/W — read and write				

6.31.2 ABE bit

The ABE bit is used to enable/disable auto-boot mode. When this bit is set to “1”, the mode is enabled, and accordingly, upon power-up, the integrated circuit will start outputting data and

will ignore commands sent to it until the first data byte is read. If the ABE bit is in the “0” state, then upon power-up, the integrated circuit is in normal mode and ready to receive any commands from Table 5. The ABE bit is stored in the non-volatile configuration register and retains its state after power-down. By default, the ABE bit is in the “0” state at delivery. The ABRWrite command is used to change the state of the ABE bit. The Reset command does not change the state of the ABE bit, but it allows re-initiating data output if ABE is “1” and data output has already occurred previously.

6.31.3 ABSD bits

The eight ABSD bits are intended to set the value of an additional initial delay in SCK clocks, which will be added to the minimum 8 wait clocks, from the transition of the nCS line to a logical “0” state until the first data appears at the integrated circuit's output. The additional delay can be set from 2 to 256 SCK clocks; when ABSD = 0h, no additional delay is used. The ABSD bits are stored in the non-volatile configuration register and retain their state after power-down. By default, the ABSD bits are in the “0” state at delivery. The ABRWrite command is used to change the state of the ABSD bits. The Reset command does not change the state of the ABSD bits.

6.31.4 ABSA bits

The fourteen ABSA bits are intended for setting the starting address for data reading in auto-boot mode. The starting address is aligned to a 512-byte page boundary (A8 — A0 all “0”). The higher addresses A22 — A9 are set using the ABSA bits. The ABSA bits are stored in the non-volatile configuration register and retain their state after power-down. By default, the ABSA bits are in the “0” state at delivery. The ABRWrite command is used to change the state of the ABSA bits. The Reset command does not change the state of the ABSA bits.

6.32 Auto-boot configuration register write operation

This operation is necessary to modify the ABE, ABSD, ABSA bits of the auto-boot register. Before executing the ABRWrite command, the WriteEn command must be issued to set the WEL bit of status register 1 to logical “1”.

When executing the ABRWrite command, the nCS line must be in a logical “0” state, and the command code 15h must be loaded into the integrated circuit along with three bytes of data. Any additional data bytes sent to the integrated circuit after this are ignored. After the nCS line is set to logical “1”, the integrated circuit will begin writing the new value of the ABE, ABSD, ABSA bits to the memory cells of the non-volatile auto-boot configuration register.

To write the new state of the ABE, ABSD, ABSA bits to memory cells, a program/erase operation is initiated. The waiting time for the completion of the ABE, ABSD, ABSA bits write operation is not less than $t_{CYW(NVR)}$. During the writing of these bits, it is recommended to use the status register 1 read operation, which shows the status of the internal operation's execution. This will reduce the waiting time for the operation to complete.

The integrated circuit executes an error detection algorithm when writing the state of the ABE, ABSD, ABSA bits: depending on which stage of non-volatile register writing (erase or programming stage) a failure occurred, the E_ERR or P_ERR bits of status register 2 may be set.

The full 3 bytes of data must be loaded into the memory module before changing the nCS line to a logical “1” state, i.e., at a byte boundary (a multiple of 8 SCK clocks), otherwise the integrated circuit will abort the operation, the state of the ABE, ABSD, ABSA bits will not change, and the value of the WEL bit of status register 1 will also not change.

The timing diagram for the non-volatile auto-boot configuration register write operation is shown in Figure 33.

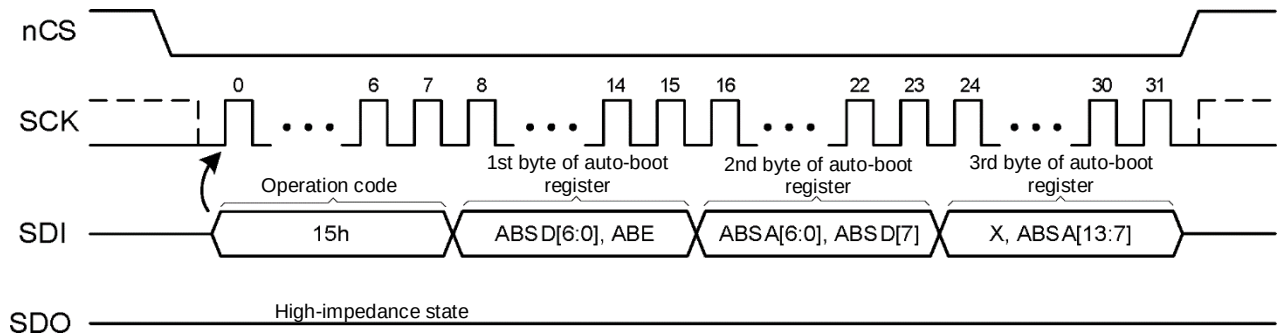
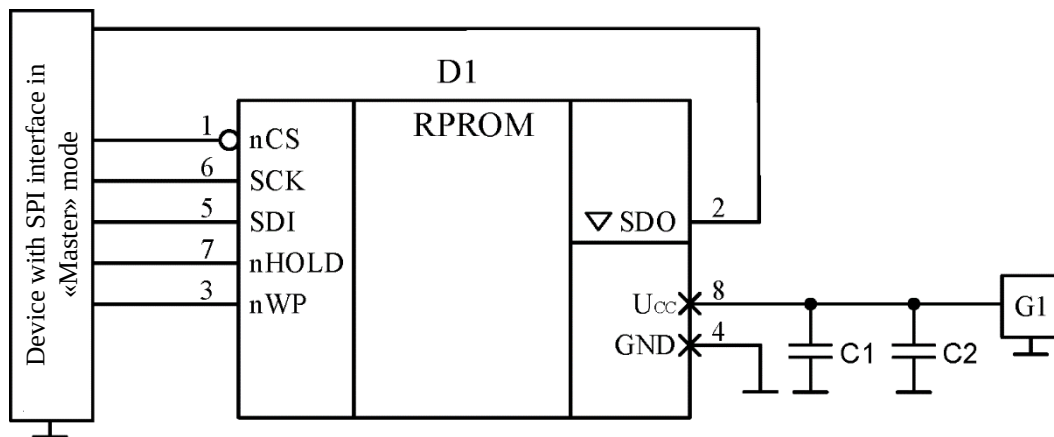


Figure 33 — Auto-boot configuration register write operation

7 Typical integrated circuit connection diagram



C1 — 0.1 $\mu\text{F} \pm 10\%$ capacitor;

C2 — 10 $\mu\text{F} \pm 10\%$ capacitor;

D1 — the IRL2306FI;

G1 — DC power supply (1.8–3.6) V

Figure 34 — Typical integrated circuit connection diagram during operation

8 Timing diagrams

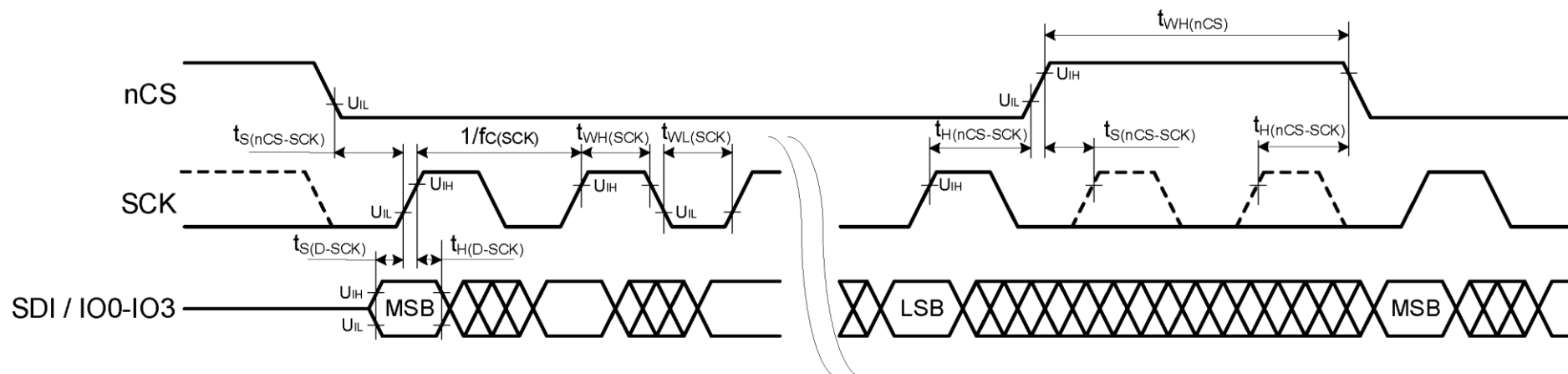


Figure 35 — SPI serial channel write timing diagram

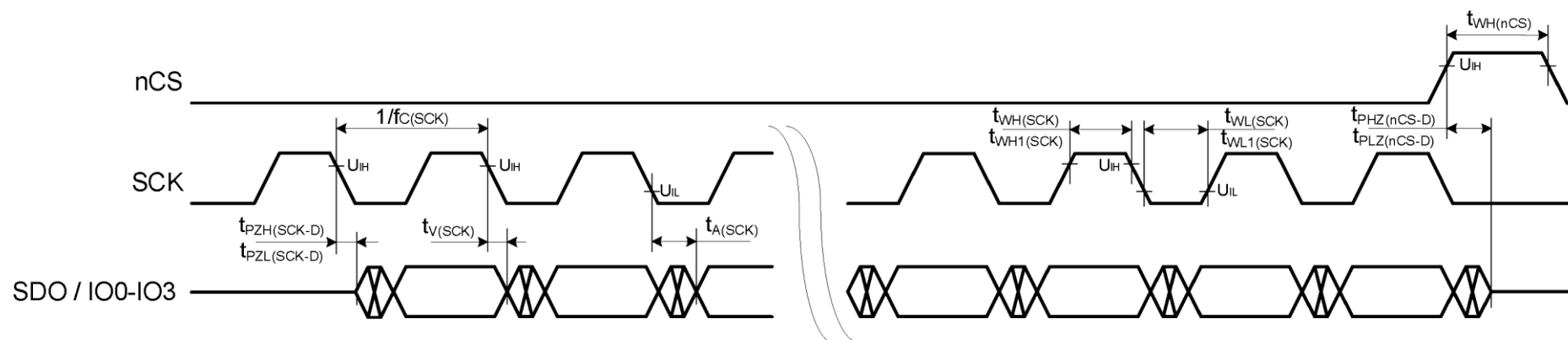


Figure 36 — SPI serial channel read timing diagram

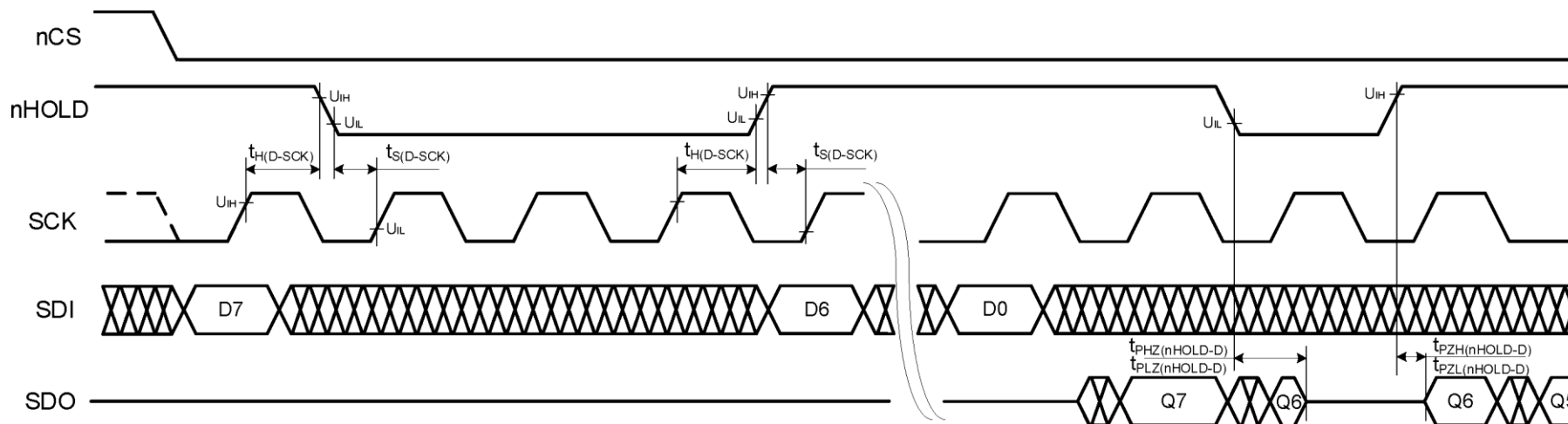


Figure 37 — Timing diagram for $nHOLD$ signal

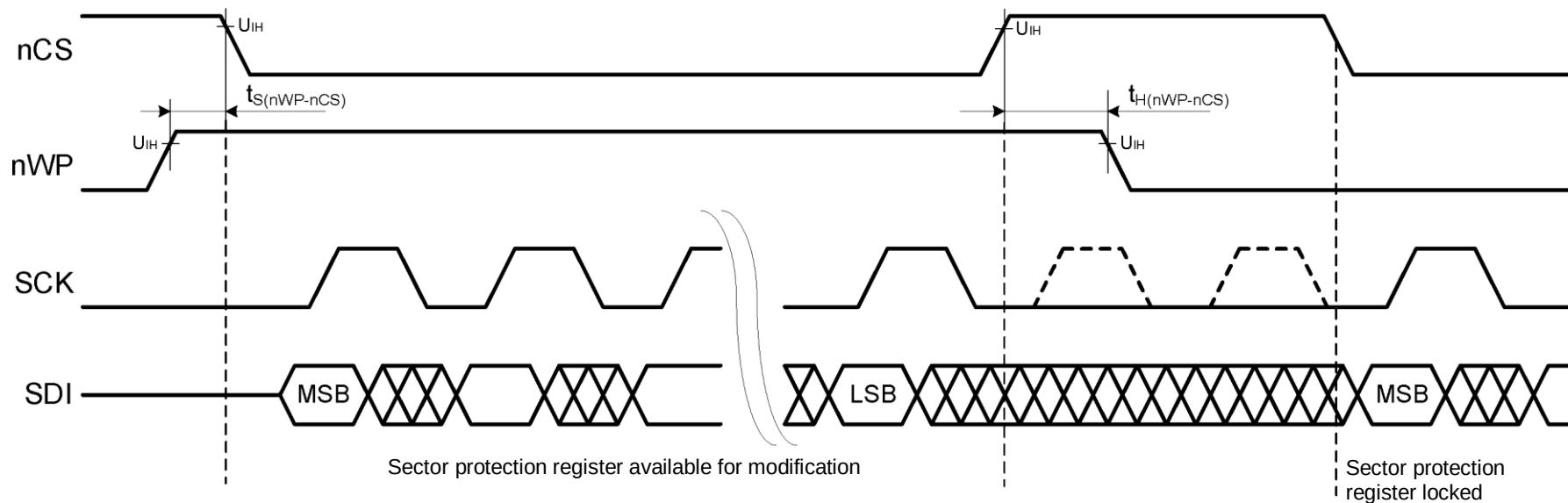


Figure 38 — Timing diagram for nWP signal

9 Electrical parameters

Table 13 — Electrical parameters of integrated circuits during acceptance and delivery

Parameter name, unit of measurement, measurement mode	Parameter	Parameter normal value		Ambient temperature, °C
		Not less than	Not more than	
High-level output voltage, V	U _{OH}	0.8 · U _{CC}	–	25, 85, — 40
Low-level output voltage, V	U _{OL}	–	0.2 · U _{CC}	
High-level input leakage current, µA	I _{ILH}	–10	10	
Low-level input leakage current, µA	I _{ILL}	–10	10	
Low-level input current, µA, on inputs with pull-down resistor	I _{IL}	–135	–	
High-level output current in “Off” state, µA	I _{OZH}	–10	10	
Low-level output current in “Off” state, µA	I _{OZL}	–10	10	
Dynamic current consumption, mA, no load	I _{CCO}	–	20	
Current consumption in storage mode, mA	I _{CCS}	–	2	
Current consumption in low power mode, mA	I _{CCS1}	–	0.4*	
Data sampling time on SCK falling edge, ns, at:	t _{A(SCK)}			
– U _{CC} ≥ 3.0 V;		–	6	
– 2.25 V ≤ U _{CC} < 3.0 V;		–	8	
– U _{CC} < 2.25 V		–	10	
Note — The “minus” sign before the current norm indicates the direction of current flowing out of the integrated circuit pin. The absolute value of the current meter readings is taken as the current value				

Integrated circuits must be resistant to electrostatic discharge with a potential of at least 2000 V.

Number of data program/erase cycles is not less than 20,000.

Data retention time — not less than 20 years.

10 Maximum permissible and limiting parameters

Table 14 — Maximum permissible operating modes and limiting electrical modes of integrated circuits

Parameter name, unit of measurement	Parameter	Maximum permissible mode		Limiting mode	
		Not less than	Not more than	Not less than	Not more than
Supply voltage, V	U_{CC}	1.8	3.6	—	3.9
Low-level input voltage, V	U_{IL}	0	$0.2 \cdot U_{CC}$	−0.3	—
High-level input voltage, V	U_{IH}	$0.8 \cdot U_{CC}$	3.6	—	3.9
Low-level output current, mA	I_{OL}	—	4	—	8
High-level output current, mA	I_{OH}	−4	—	−8	—
Low-level voltage in “Off” state, V	U_{OZL}	0	—	−0.3	—
High-level voltage in “Off” state, V	U_{OZH}	—	3.6	—	3.9
Sector erase operation completion wait time, ms	t_{ER_SEC}	32	—	—	—
Block erase operation completion wait time, ms	t_{ER_BLK}	100	—	—	—
Chip erase operation completion wait time, ms	t_{ER_CHIP}	400	—	—	—
4-byte word programming operation completion wait time, μ s	t_{PR_WRD}	52	—	—	—
512-byte page programming operation completion wait time, ms	t_{PR_PG}	1.65	—	—	—
Sector unprotect operation completion wait time, ms	t_{UNPRT}	32	—	—	—
Sector protect operation completion wait time, μ s	t_{PRT}	52	—	—	—
Non-volatile configuration register write cycle time, ms	$t_{CYW(NVR)}$	32	—	—	—
SPI interface clock frequency, MHz, at:	$f_{C(SCK)}$				
— $U_{CC} \geq 3.0$ V;		—	100	—	—
— 2.25 V $\leq U_{CC} < 3.0$ V;		—	50	—	—
— $U_{CC} < 2.25$ V		—	25	—	—
SCK clock signal high level duration, ns, at:	$t_{WH(SCK)}$				
— $U_{CC} \geq 3.0$ V;		4	—	—	—
— 2.25 V $\leq U_{CC} < 3.0$ V;		8	—	—	—
— $U_{CC} < 2.25$ V		10	—	—	—
SCK clock signal low level duration, ns, at:	$t_{WL(SCK)}$				
— $U_{CC} \geq 3.0$ V;		6	—	—	—
— 2.25 V $\leq U_{CC} < 3.0$ V;		8	—	—	—
— $U_{CC} < 2.25$ V		10	—	—	—

Parameter name, unit of measurement	Parameter	Maximum permissible mode		Limiting mode	
		Not less than	Not more than	Not less than	Not more than
SPI interface clock frequency in read mode with code 03h, MHz, at: – $U_{CC} \geq 2.25 \text{ V}$; – $U_{CC} < 2.25 \text{ V}$	$f_{C1(SCK)}$	–	40	–	–
		–	25		
SCK clock signal high level duration, ns, in read mode with code 03h	$t_{WH1(SCK)}$	10	–	–	–
SCK clock signal low level duration, ns, in read mode with code 03h	$t_{WL1(SCK)}$	10	–	–	–
SDI/IO0, SDO/IO1, nWP/IO2, nHOLD/IO3 input signal settling time relative to SCK clock signal edge, ns, at: – $U_{CC} \geq 3.0 \text{ V}$; – $2.25 \text{ V} \leq U_{CC} < 3.0 \text{ V}$; – $U_{CC} < 2.25 \text{ V}$	$t_{SU(D-SCK)}$	3	–	–	–
		5	–	–	–
		7	–	–	–
SDI/IO0, SDO/IO1, nWP/IO2, nHOLD/IO3 input signal hold time relative to SCK clock signal edge, ns – $U_{CC} \geq 3.0 \text{ V}$; – $2.25 \text{ V} \leq U_{CC} < 3.0 \text{ V}$; – $U_{CC} < 2.25 \text{ V}$	$t_{H(D-SCK)}$	1	–	–	–
		3	–	–	–
		5	–	–	–
nCS signal low/high level settling time relative to SCK clock signal edge, ns – $U_{CC} \geq 3.0 \text{ V}$; – $2.25 \text{ V} \leq U_{CC} < 3.0 \text{ V}$; – $U_{CC} \leq 2.25 \text{ V}$	$t_{SU(nCS-SCK)}$	6	–	–	–
		8	–	–	–
		10	–	–	–
nCS signal low/high level hold time relative to SCK clock signal edge, ns – $U_{CC} \geq 3.0 \text{ V}$; – $2.25 \text{ V} \leq U_{CC} < 3.0 \text{ V}$; – $U_{CC} < 2.25 \text{ V}$	$t_{H(nCS-SCK)}$	4	–	–	–
		6	–	–	–
		8	–	–	–
nCS signal high level duration, ns	$t_{WH(nCS)}$	10	–	–	–
nWP signal low/high level settling time relative to nCS falling edge, ns	$t_{SU(nWP-nCS)}$	5	–	–	–
nWP signal low/high level hold time relative to nCS signal edge, ns	$t_{H(nWP-nCS)}$	5	–	–	–
Output load capacitance, pF	C_L	–	15	–	–

Parameter name, unit of measurement	Parameter	Maximum permissible mode		Limiting mode	
		Not less than	Not more than	Not less than	Not more than
Notes					
1 The value of parameters t_{ER_SEC} , t_{ER_BLK} , t_{ER_CHIP} , t_{PR_WRD} , t_{PR_PG} , t_{UNPRT} , t_{PRT} , $t_{CYW(NWR)}$ corresponds to the minimum wait time for program/erase operations to complete without tracking the operation status. To reduce program/erase time, it is recommended to use the status register 1 read operation.					
2 Simultaneous exposure to two or more limiting conditions is not allowed					

11 Reference parameters

Table 15 — ICs reference parameters

Parameter name, unit of measurement, measurement mode	Parameter	Parameter normal value		Ambient temperature, °C
		Not less than	Not more than	
Time from power supply application to first access to the integrated circuit, μs	$t_{\text{PWR_ON}}$	150	—	25, 85, -40
Power supply hold time relative to the last completed operation, μs	$t_{\text{PWR_OFF}}$	0	—	
Integrated circuit transition time to low power mode, ns	$t_{\text{SLP_ON}}$	—	10	
Integrated circuit exit time from low power mode, μs	$t_{\text{SLP_OFF}}$	—	7.5	
Erase suspend time, ns	$t_{\text{SUSP_ER}}$	—	400	
Program suspend time, μs	$t_{\text{SUSP_PR}}$	—	52	
Program / erase resume time, ns	t_{RESUME}	—	150	
Data signal propagation delay timing when output transitions from low (high) level to “Shutdown” state on nCS signal rising edge, ns	$t_{\text{PHZ(nCS-D)}}$ $t_{\text{PLZ(nCS-D)}}$	—	10	
Data signal propagation delay timing when output transitions from “Shutdown” state to low (high) level on SCK clock signal falling edge, ns	$t_{\text{PZH(SCK-D)}}$ $t_{\text{PZL(SCK-D)}}$	0	—	
Data signal propagation delay timing when output transitions from low (high) level to “Shutdown” state on nHOLD falling edge, ns	$t_{\text{PHZ(nHOLD-D)}}$ $t_{\text{PLZ(nHOLD-D)}}$	—	10	
Data signal propagation delay timing when output transitions from “Shutdown” state to low (high) level on nHOLD signal rising edge, ns	$t_{\text{PZH(nHOLD-D)}}$ $t_{\text{PZL(nHOLD-D)}}$	0	—	
Output data hold time after SCK clock signal falling edge, ns	$t_{\text{V(SCK)}}$	0	—	

Input capacitance $C_{\text{I}} = 7 \text{ pF}$ at 1 MHz.

Input/output capacitance $C_{\text{I/O}} = 7 \text{ pF}$ at 1 MHz.

12 Typical dependencies

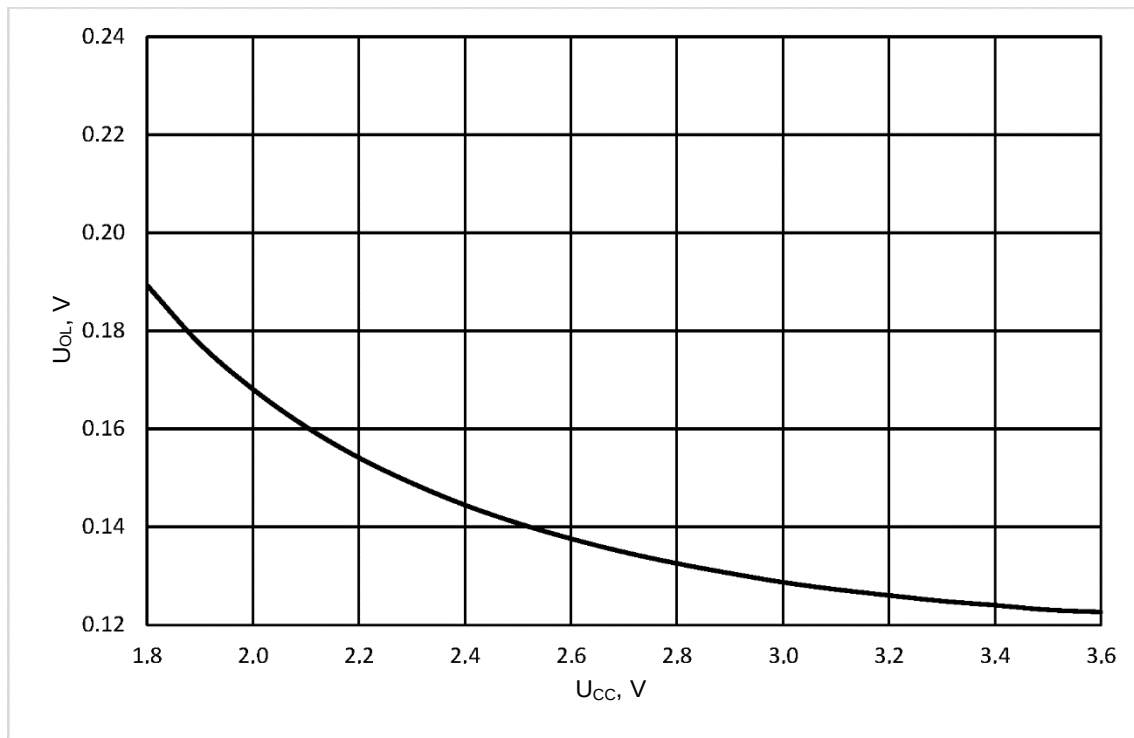


Figure 39 — Low-level output voltage U_{OL} vs. supply voltage U_{CC} at $T = 25^{\circ}\text{C}$ and $I_{OL} = 4\text{ mA}$

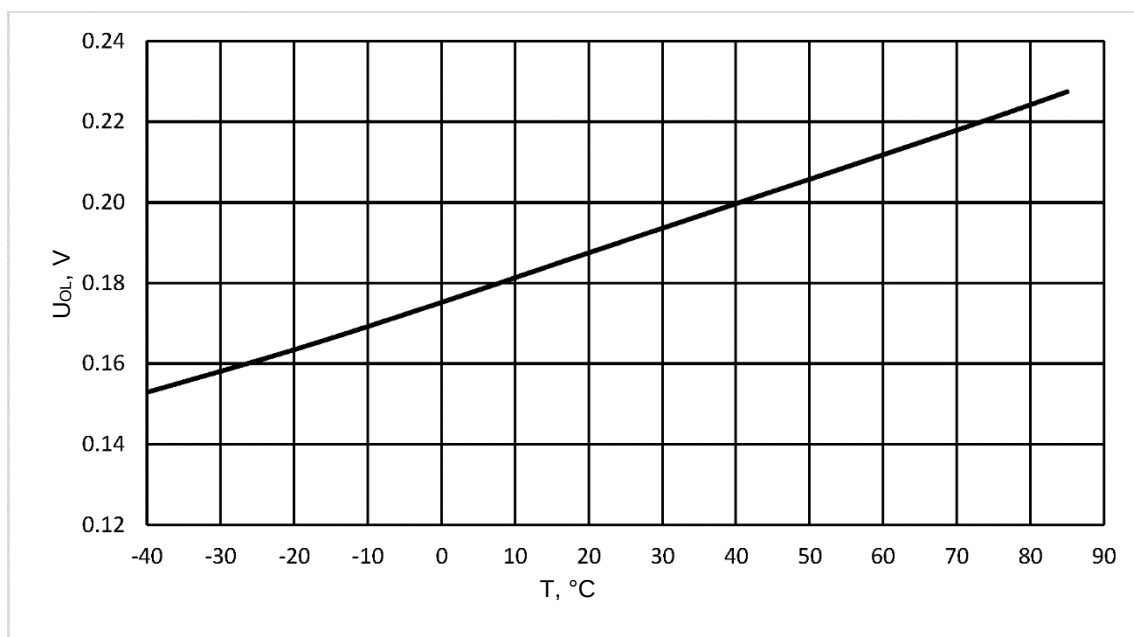


Figure 40 — Low-level output voltage U_{OL} vs. temperature T at $U_{CC} = 1.8\text{ V}$ and $I_{OL} = 4\text{ mA}$

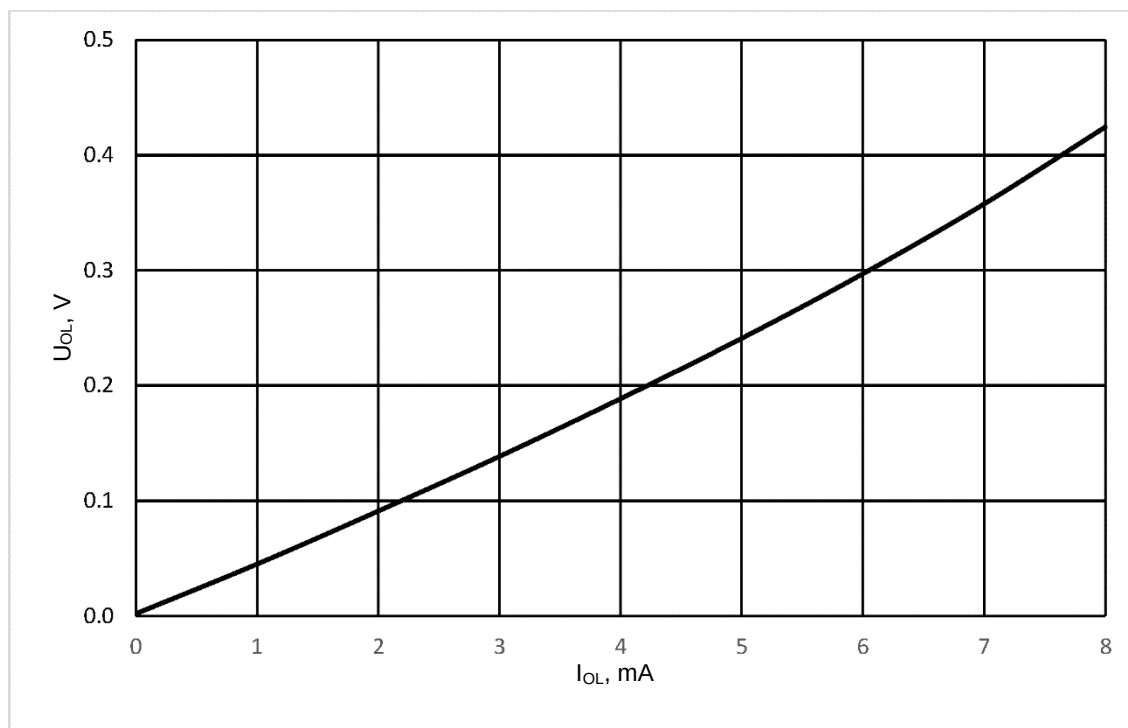


Figure 41 — Low-level output voltage U_{OL} vs. load current I_{OL} at $U_{CC} = 1.8$ V, $T = 25^{\circ}\text{C}$

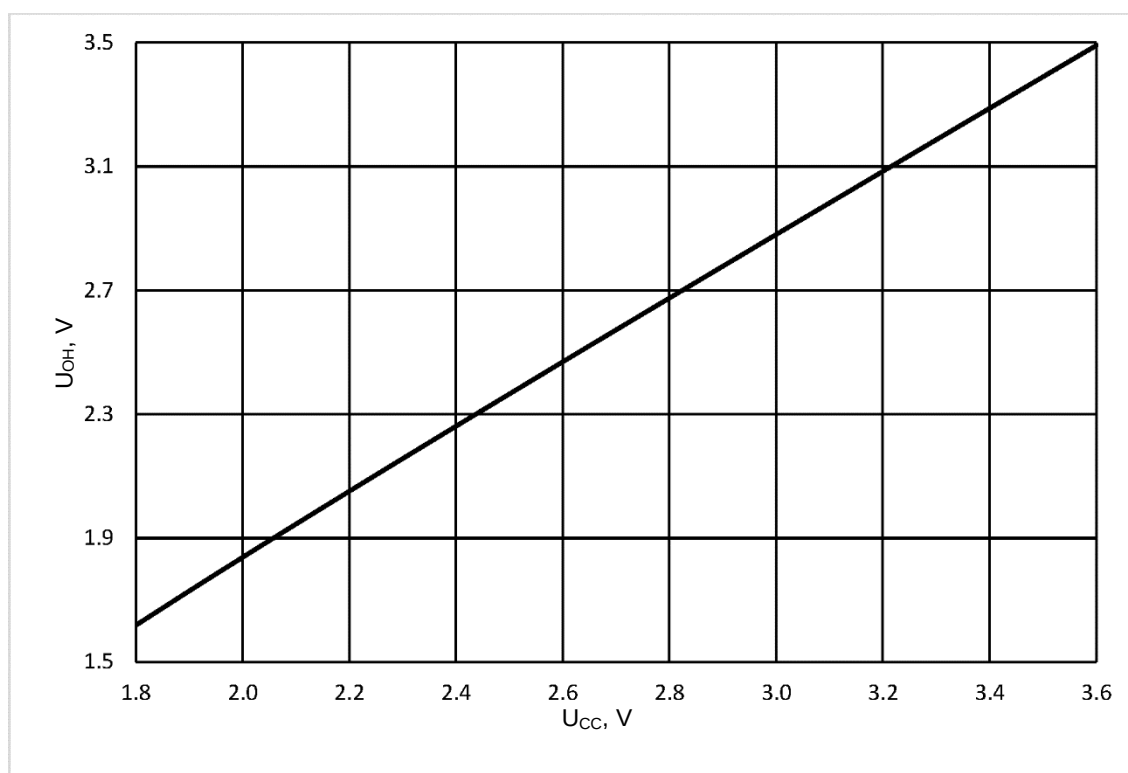


Figure 42 — High-level output voltage U_{OH} vs. supply voltage U_{CC} at $T = 25^{\circ}\text{C}$ and $I_{OH} = -4$ mA

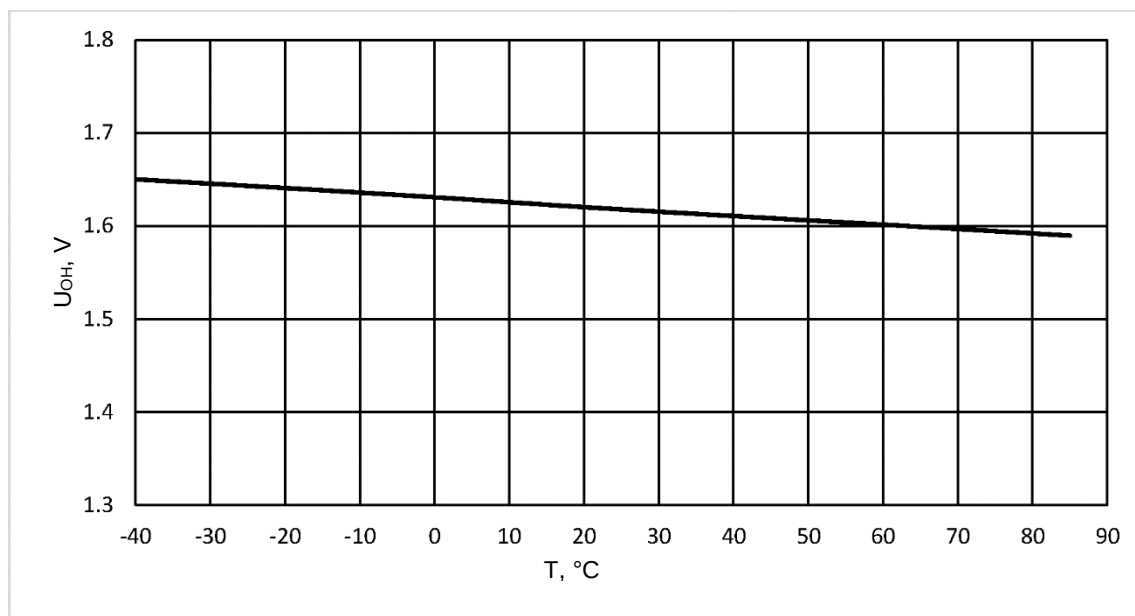


Figure 43 — High-level output voltage U_{OH} vs. temperature T at $U_{CC} = 1.8$ V and $I_{OH} = -4$ mA

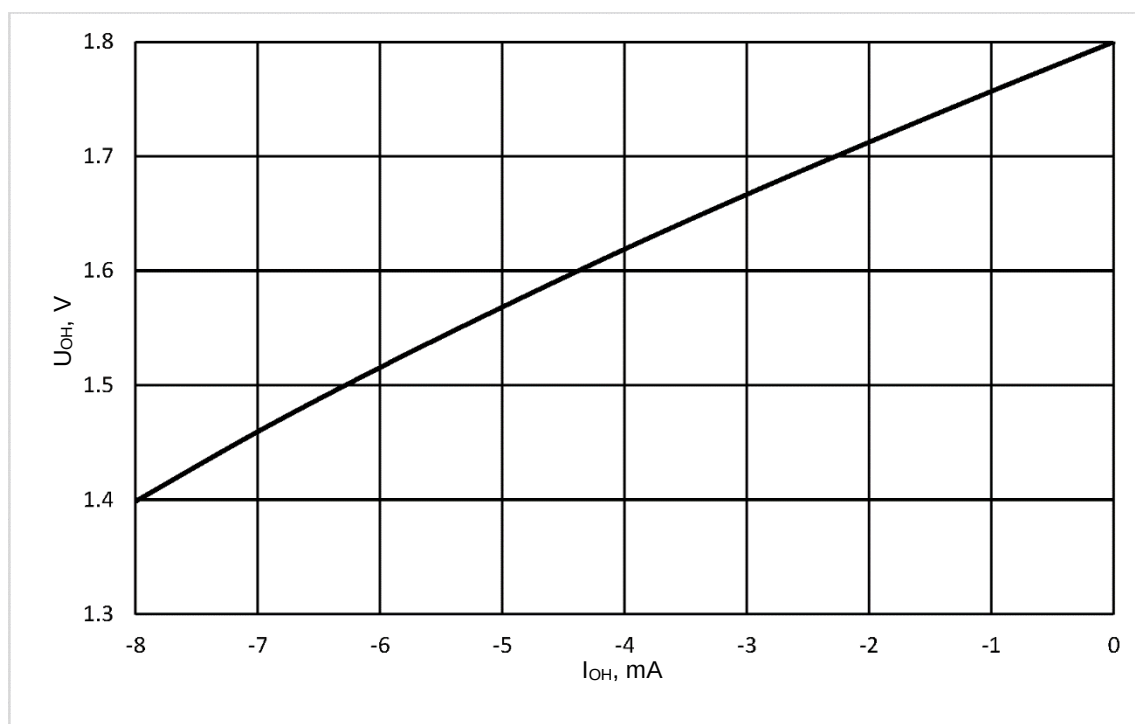


Figure 44 — High-level output voltage U_{OH} vs. load current I_{OH} at $U_{CC} = 1.8$ V, $T = 25^{\circ}\text{C}$

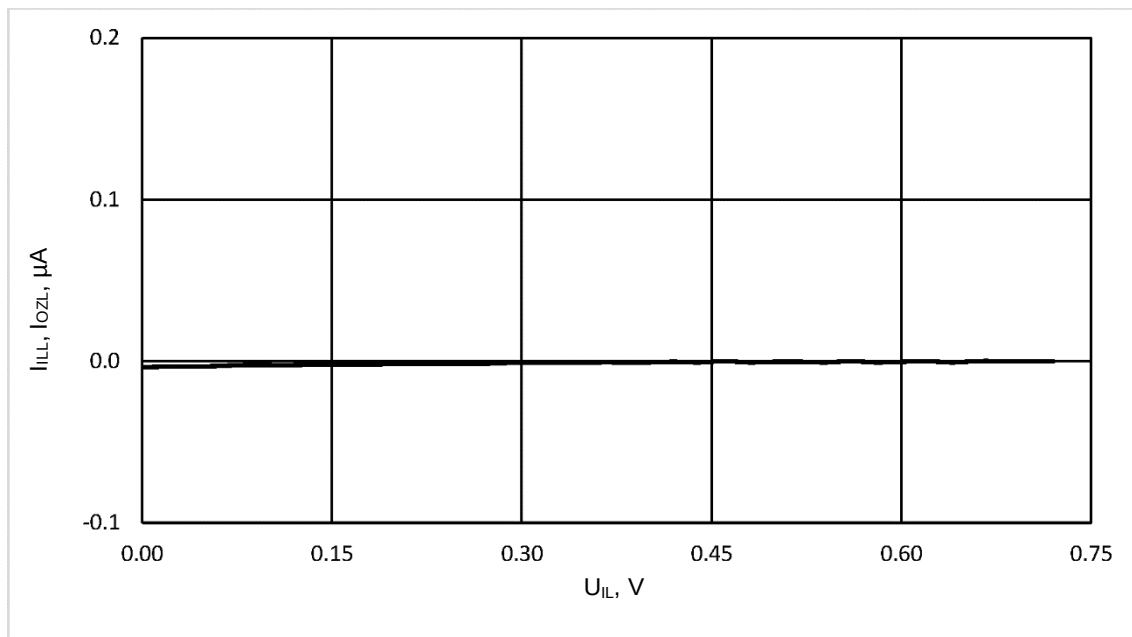


Figure 45 — Low-level input leakage current I_{IL} , low-level output current in “Shutdown” state I_{OZL} vs. low-level input voltage U_{IL} at $U_{CC} = 3.6$ V, $T = 25$ °C, $U_{IH} = U_{CC}$

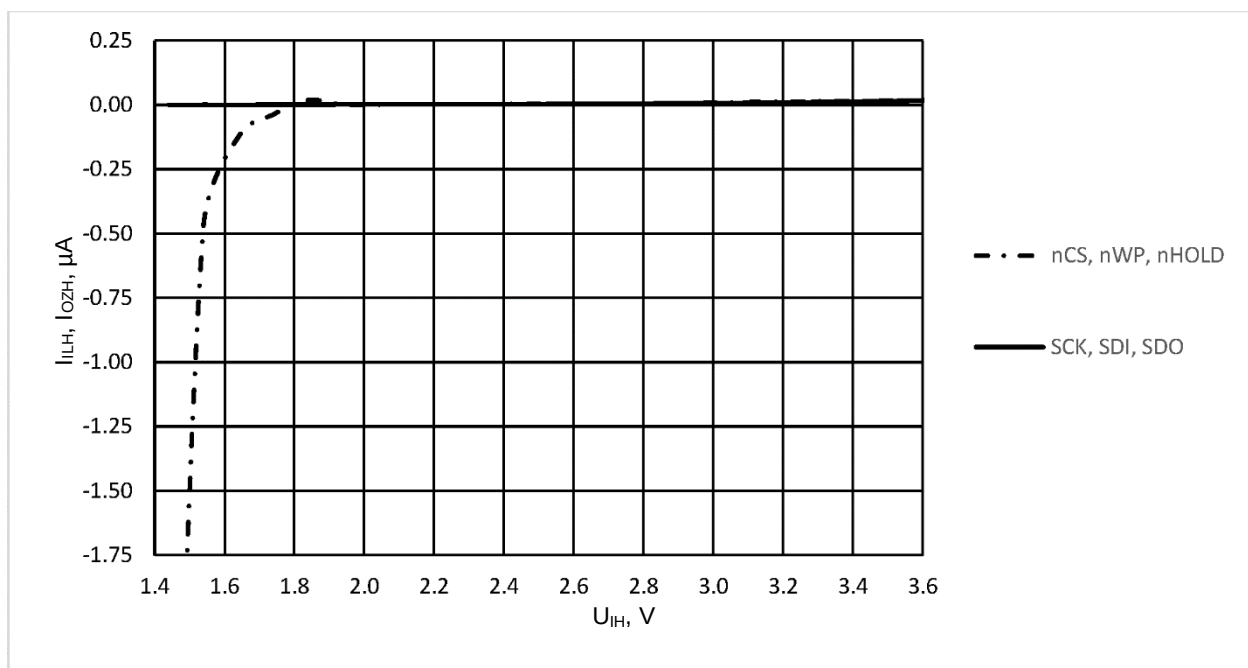


Figure 46 — High-level input leakage current I_{ILH} , high-level output current in “Shutdown” state I_{OZH} vs. high-level input voltage U_{IH} at $U_{CC} = 1.8$ V, $U_{IL} = 0$ V, $T = 25$ °C

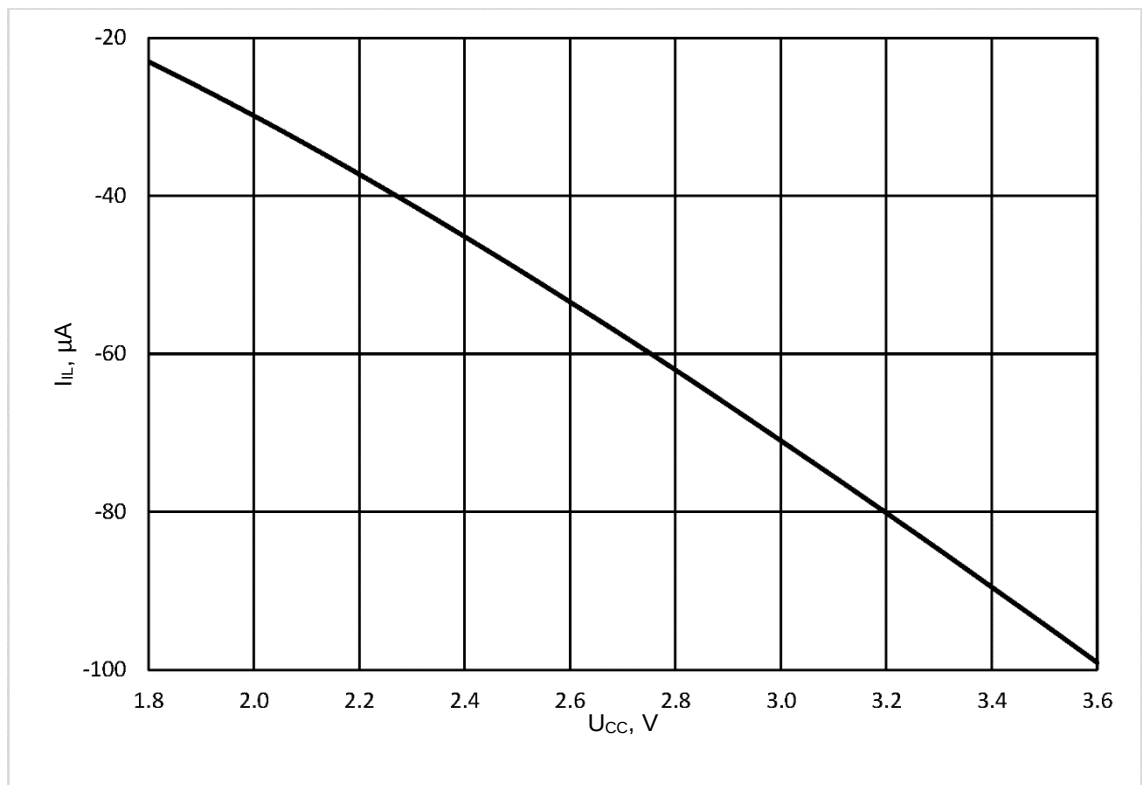


Figure 47 — Low-level input current I_{IL} vs. supply voltage U_{CC} at $T = 25\text{ °C}$, $U_{IL} = 0\text{ V}$, $U_{IH} = U_{CC}$

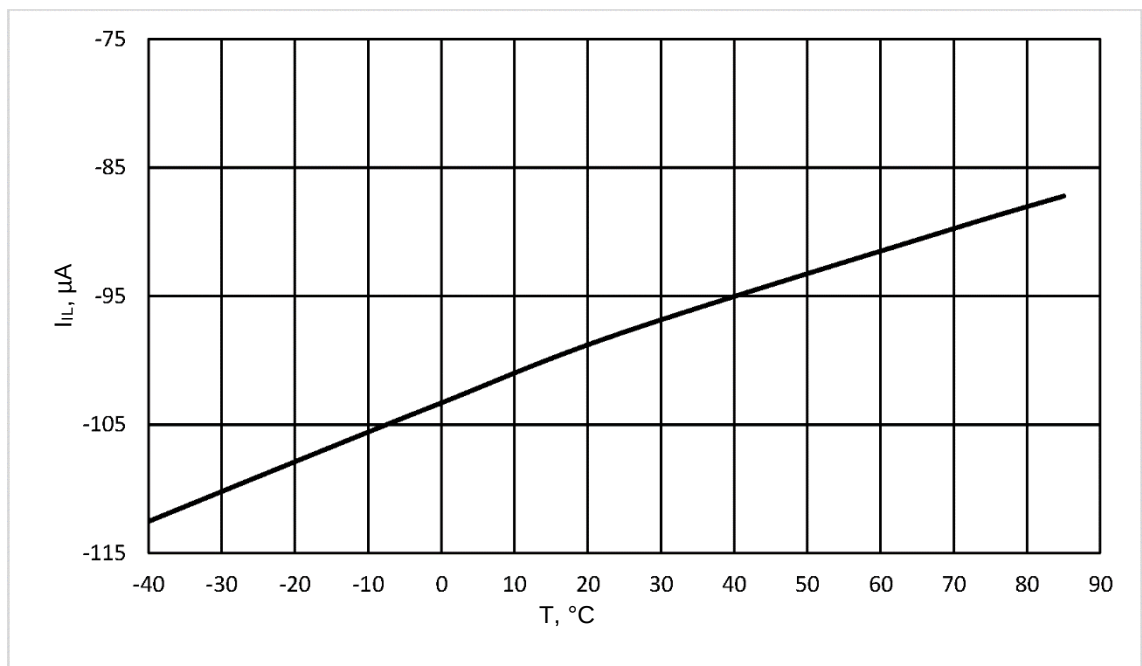


Figure 48 — Low-level input current I_{IL} vs. temperature T at $U_{CC} = 3.6\text{ V}$, $U_{IL} = 0\text{ V}$, $U_{IH} = U_{CC}$

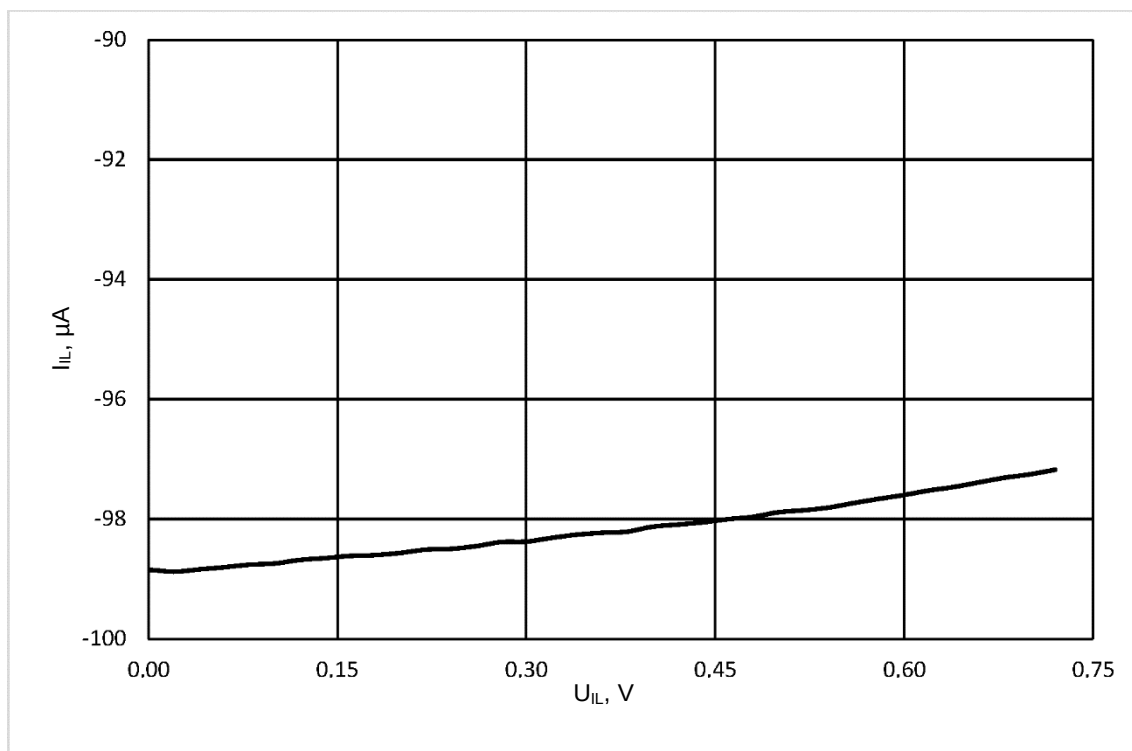


Figure 49 — Low-level input current I_{IL} vs. low-level input voltage U_{IL} at $U_{CC} = 3.6$ V, $T = 25^{\circ}C$, $U_{IH} = U_{CC}$

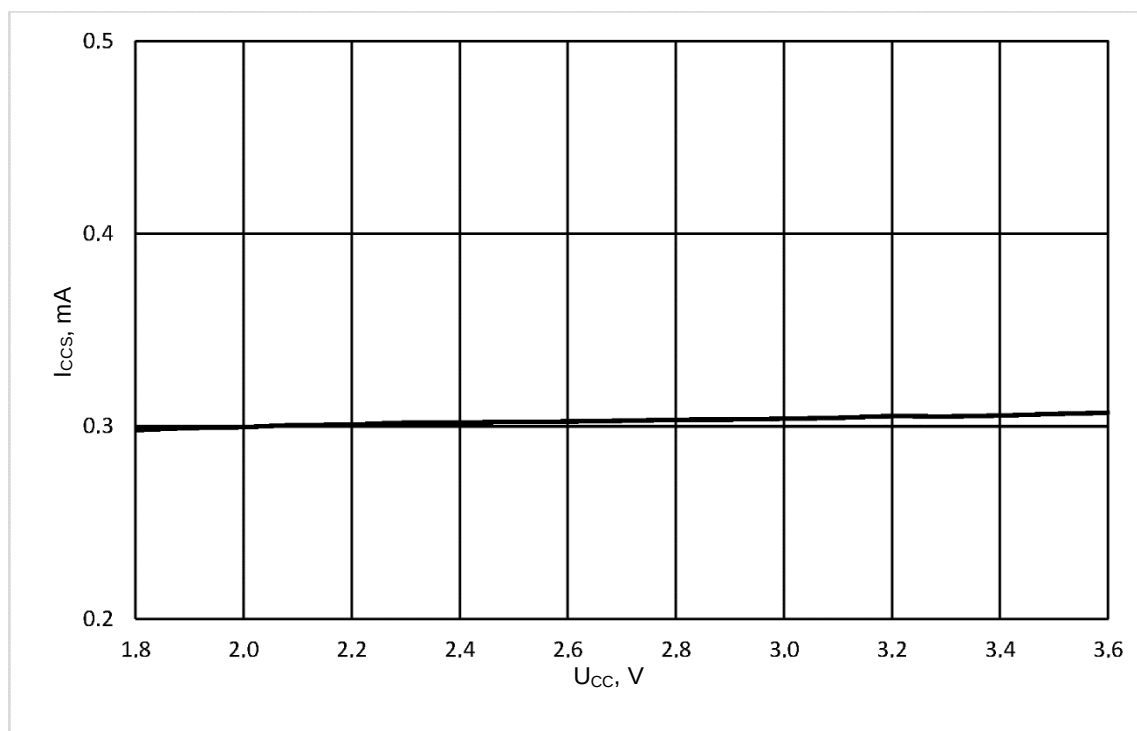


Figure 50 — Standby current I_{CCS} vs. supply voltage U_{CC} at $T = 25^{\circ}C$, $U_{IL} = 0$ V, $U_{IH} = U_{CC}$, $U_{nCS} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz

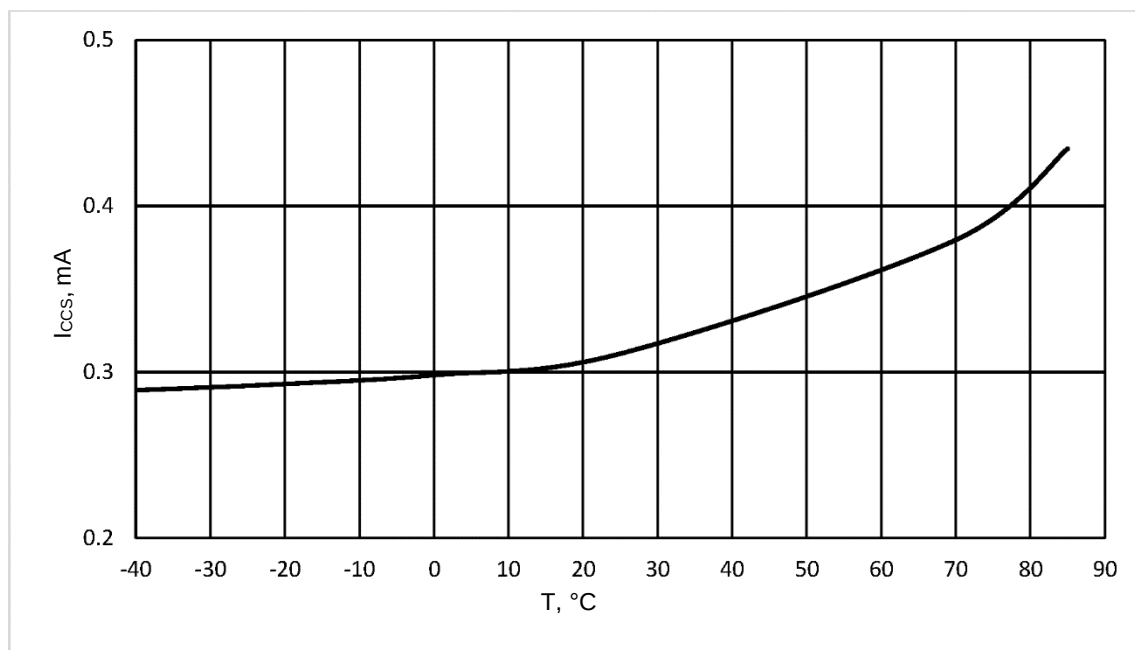


Figure 51 — Standby current I_{CCS} vs. temperature T at $U_{CC} = 3.6$ V, $U_{IL} = 0$ V, $U_{IH} = U_{CC}$, $U_{nCS} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz

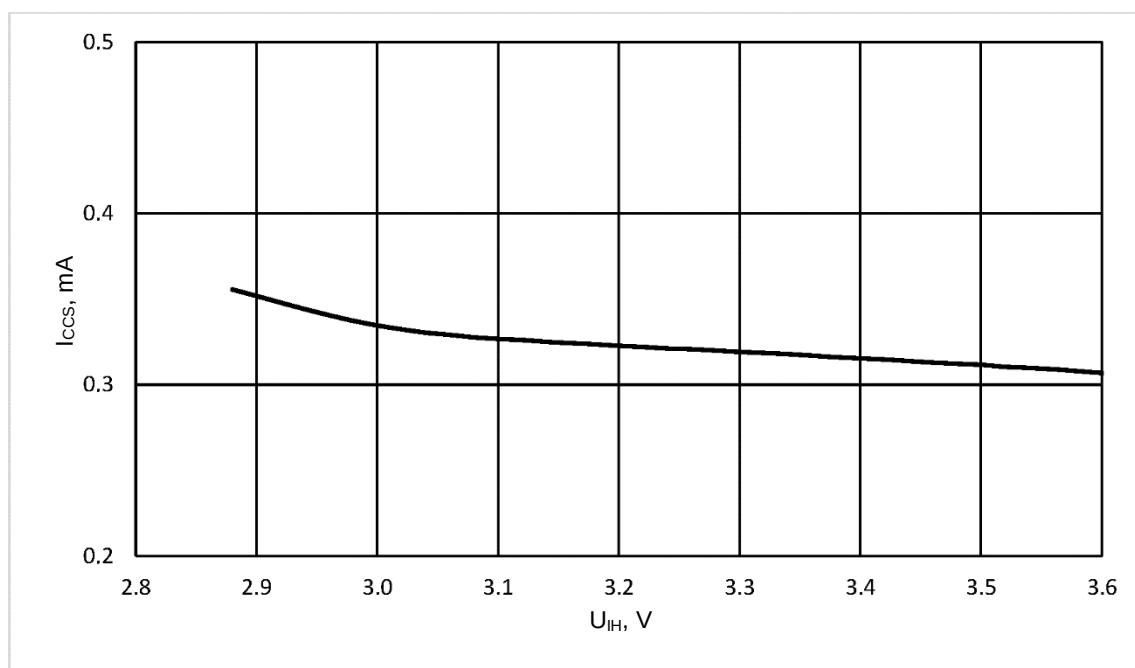


Figure 52 — Standby current I_{CCS} vs. high-level input voltage U_{IH} at $U_{CC} = 3.6$ V, $U_{IL} = 0$ V, $U_{nCE} = U_{IH}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz, $T = 25^{\circ}\text{C}$

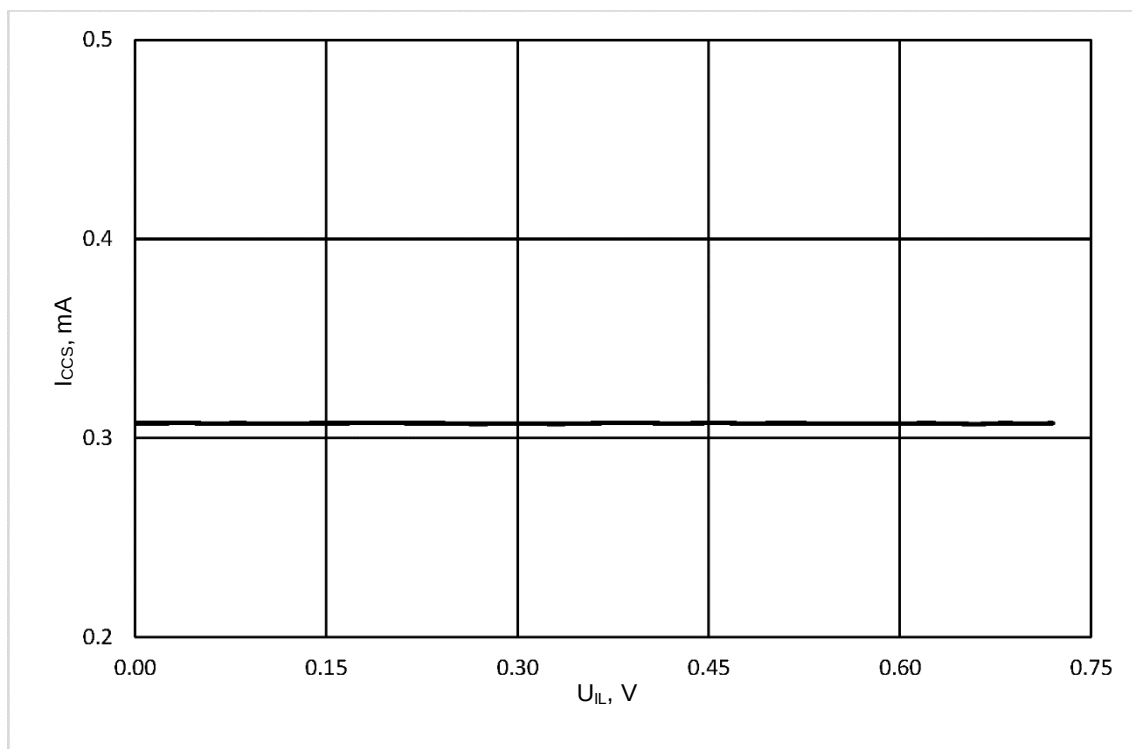


Figure 53 — Standby current I_{CCS} vs. low-level input voltage U_{IL} at $U_{CC} = 3.6$ V, $U_{IH} = U_{CC}$, $U_{nCE} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz, $T = 25^{\circ}\text{C}$

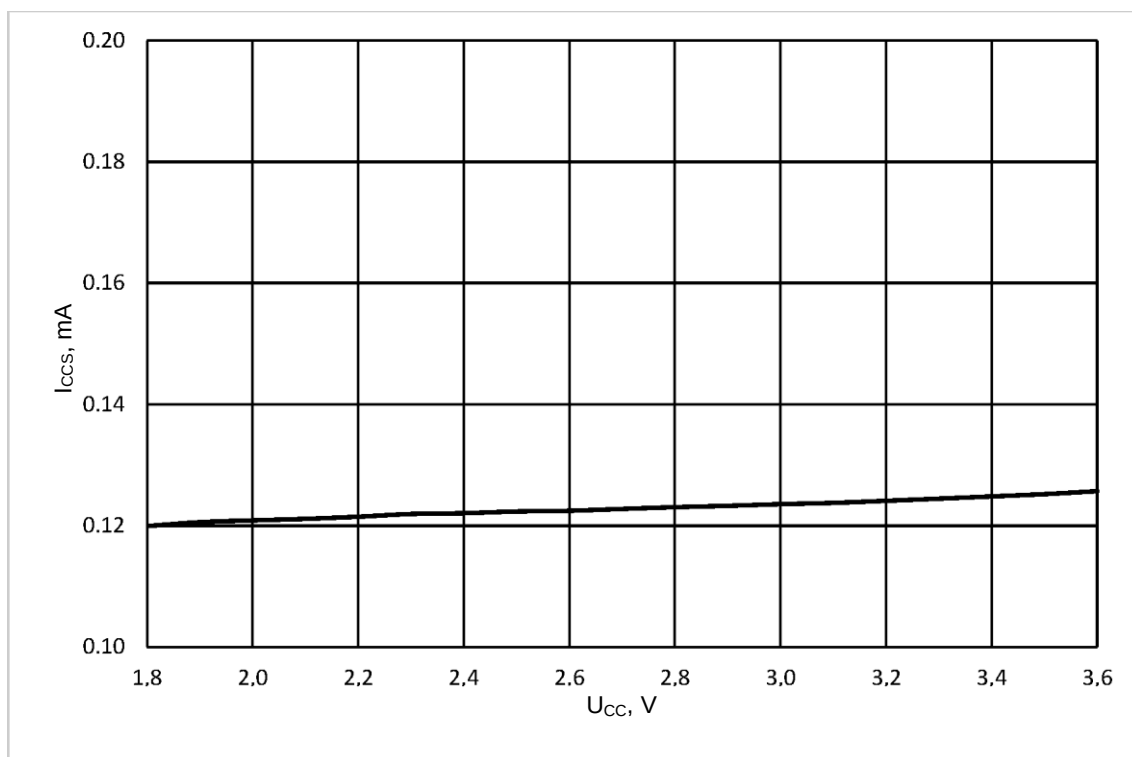


Figure 54 — Low-power mode current I_{CCS1} vs. supply voltage U_{CC} at $T = 25^{\circ}\text{C}$, $U_{IL} = 0$ V, $U_{IH} = U_{CC}$, $U_{nCS} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz

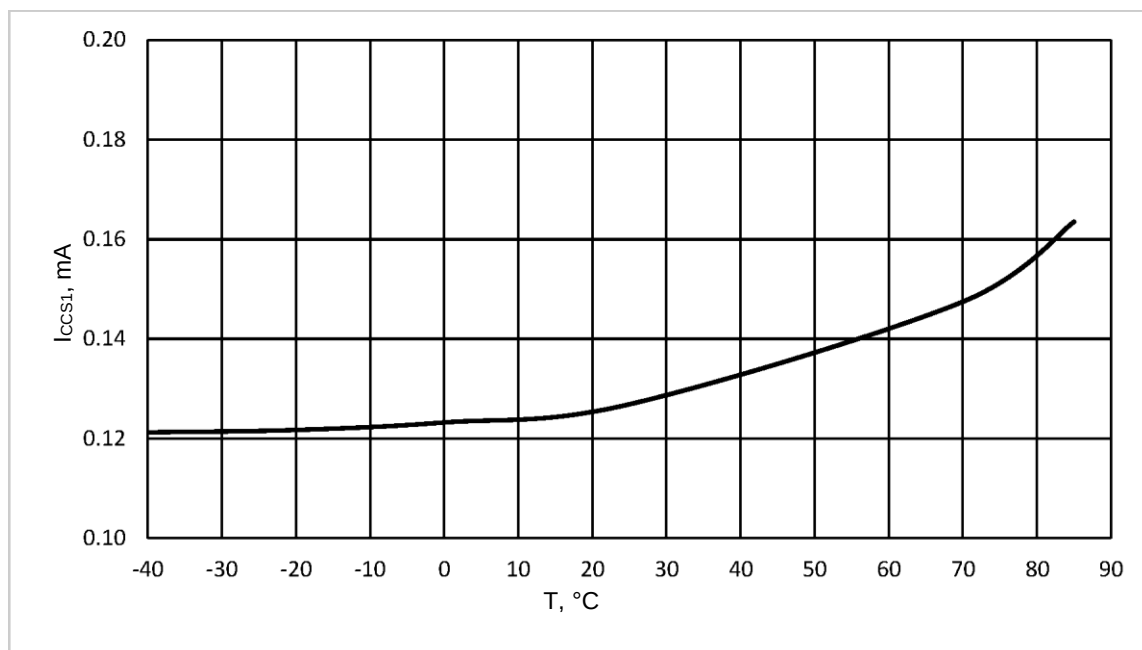


Figure 55 — Low-power mode current I_{CCS1} vs. temperature T at $U_{CC} = 3.6$ V, $U_{IL} = 0$ V, $U_{IH} = U_{CC}$, $U_{nCS} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz

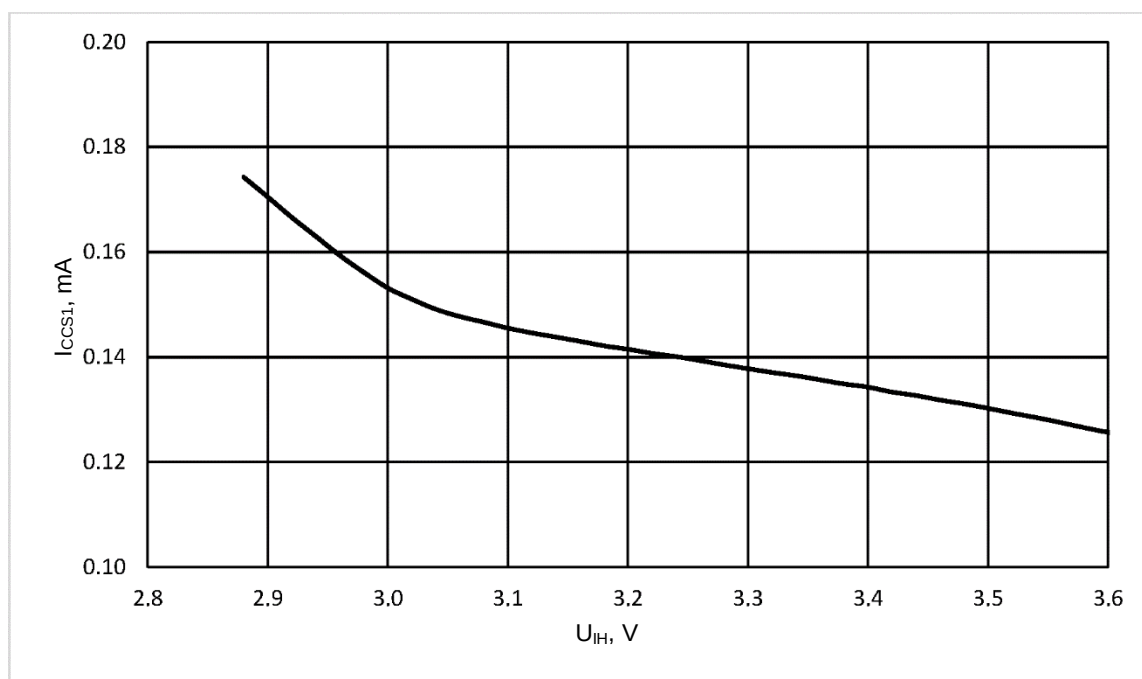


Figure 56 — Low-power mode current I_{CCS1} vs. high-level input voltage U_{IH} at $U_{CC} = 3.6$ V, $U_{IL} = 0$ V, $U_{nCE} = U_{IH}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz, $T = 25^{\circ}\text{C}$

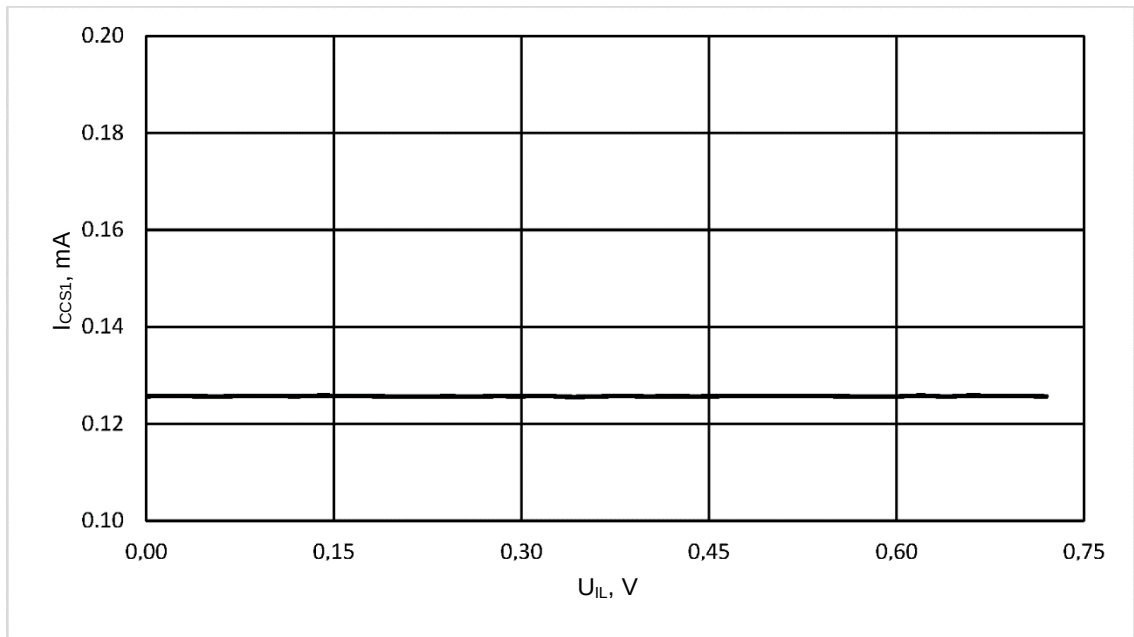


Figure 57 — Low-power mode current I_{CCS1} vs. low-level input voltage U_{IL} at $U_{CC} = 3.6$ V, $U_{IH} = U_{CC}$, $U_{nCE} = U_{CC}$, $U_{nWP} = U_{nHOLD} = Z$, $f_{C(SCK)} = 0$ MHz, $T = 25^{\circ}\text{C}$

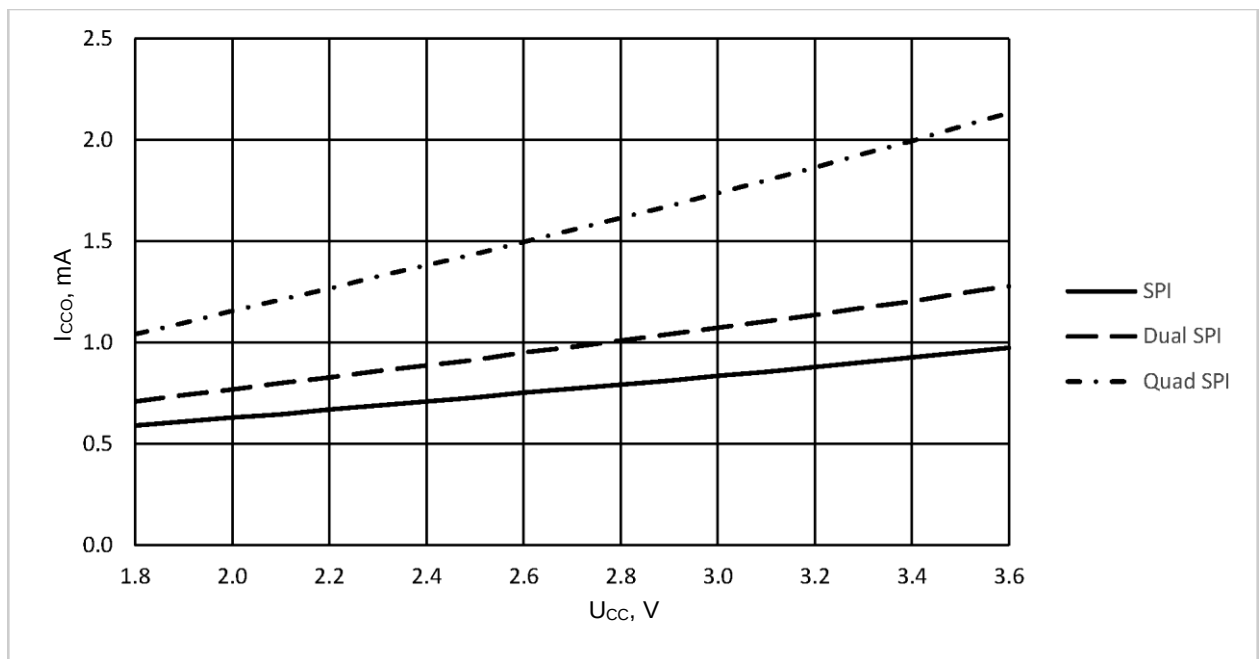


Figure 58 — Dynamic current consumption I_{CCO} during read in SPI, Dual SPI and Quad SPI modes vs. supply voltage U_{CC} at $T = 25^{\circ}\text{C}$, $U_{IL} = 0$ V, $U_{IH} = U_{CC}$, $f_{C(SCK)} = 25$ MHz

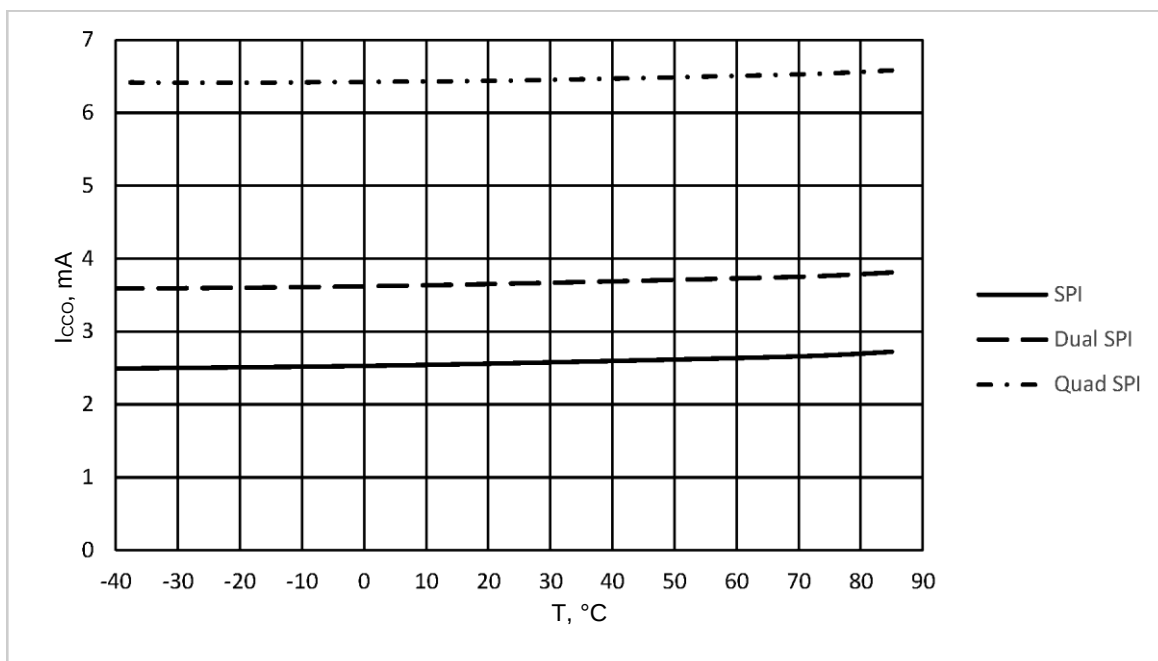


Figure 59 — Dynamic current consumption I_{CCO} during read in SPI, Dual SPI and Quad SPI modes vs. temperature T at $U_{CC} = 3.6\text{ V}$, $f_{C(SCK)} = 100\text{ MHz}$, $U_{IL} = 0\text{ V}$, $U_{IH} = U_{CC}$

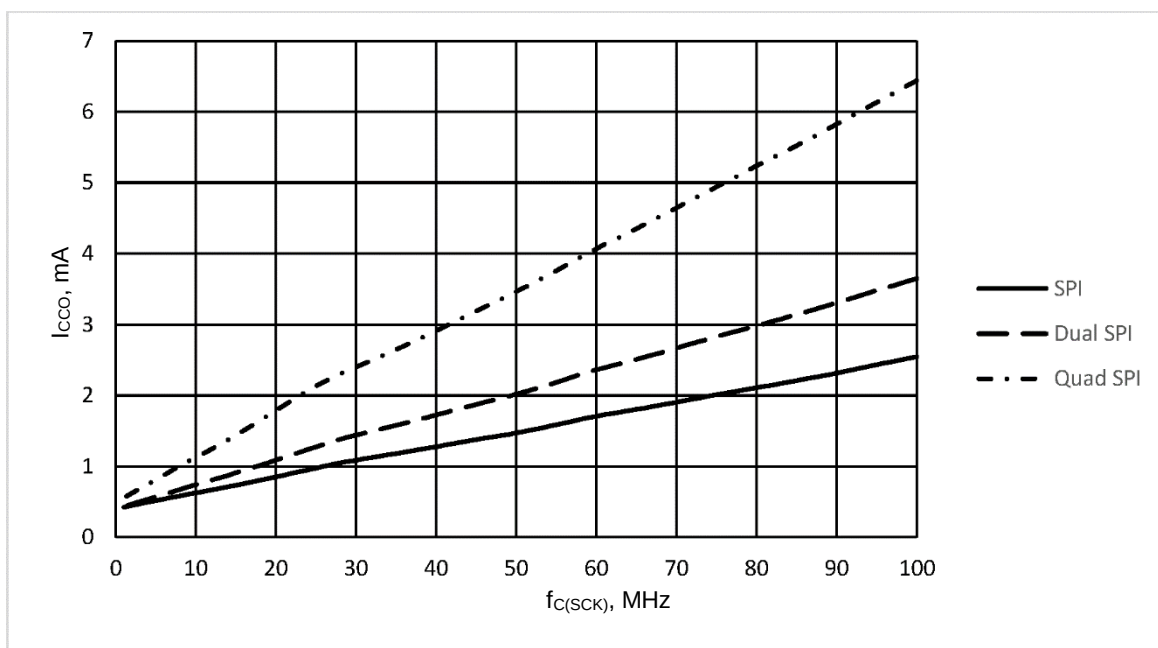


Figure 60 — Dynamic current consumption I_{CCO} during read in SPI, Dual SPI and Quad SPI modes vs. SPI interface clock frequency $f_{C(SCK)}$ at $U_{CC} = 3.6\text{ V}$, $U_{IL} = 0\text{ V}$, $U_{IH} = U_{CC}$ and $T = 25^\circ\text{C}$

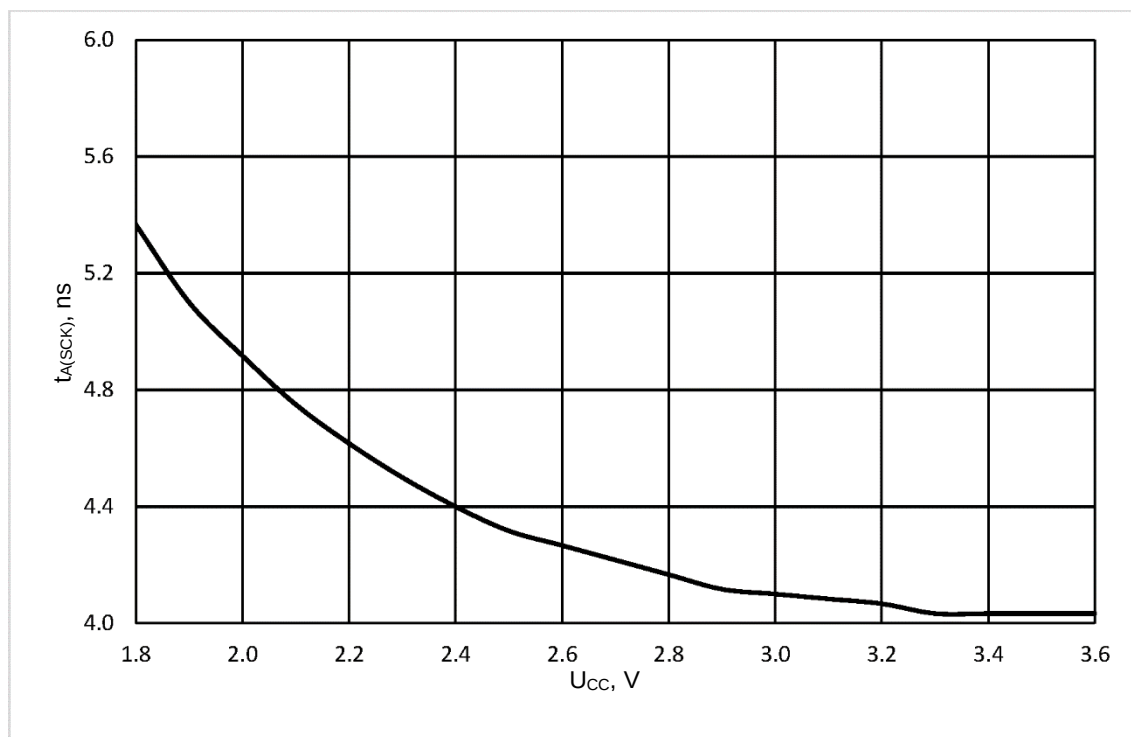


Figure 61 — SCK falling edge data access time $t_{A(SCK)}$ vs. supply voltage U_{CC} at $T = 25^{\circ}\text{C}$,
 $f_{C(SCK)} = 25 \text{ MHz}$, $U_{IL} = 0 \text{ V}$, $U_{IH} = U_{CC}$

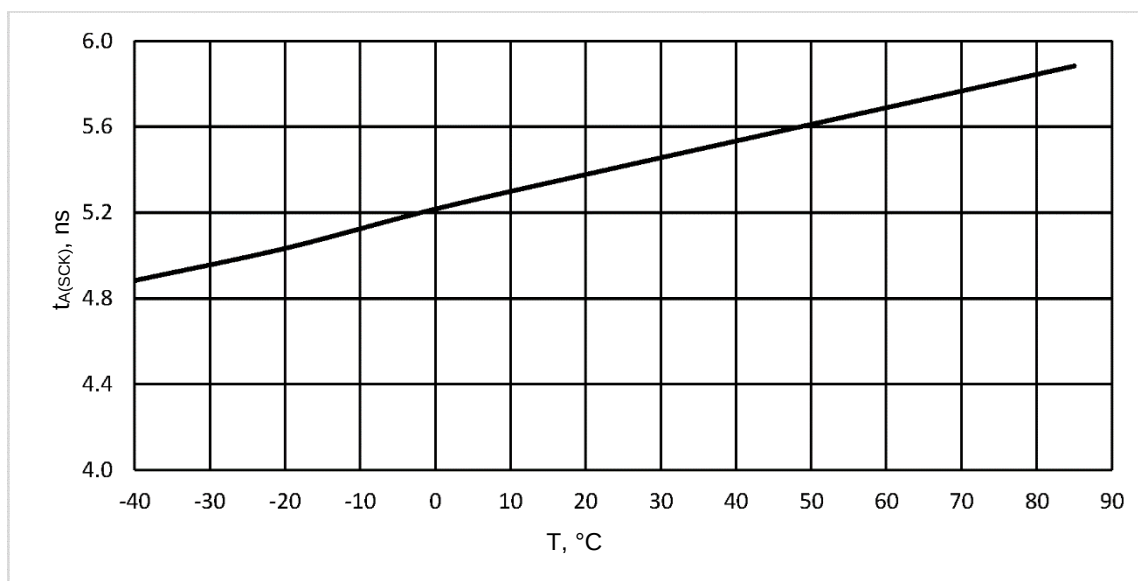


Figure 62 — SCK falling edge data access time $t_{A(SCK)}$ vs. temperature T at $U_{CC} = 1.8 \text{ V}$,
 $f_{C(SCK)} = 25 \text{ MHz}$, $U_{IL} = 0 \text{ V}$, $U_{IH} = U_{CC}$

13 Integrated circuit dimensional drawing

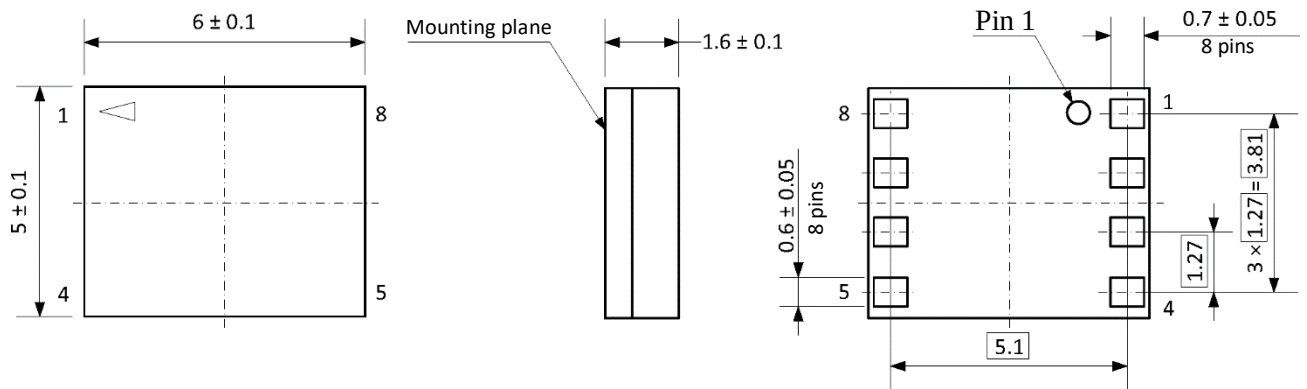


Figure 63 — Integrated circuit in DFN8 5×6×1.6 (1.27) package

14 Ordering information

Designation	Marking	Package type:	Temperature range, °C
IRL2306FI	2306	DFN8 5×6×1.6 (1.27)	from — 40 to 85