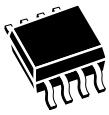
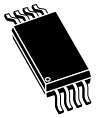
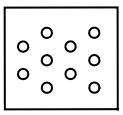
ST25DV04KC ST25DV16KC ST25DV64KC

Datasheet

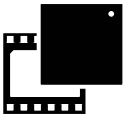
Dynamic NFC/RFID tag IC with 4-, 16-, or 64-Kbit EEPROM, fast transfer mode capability, and optimized I^2C


SO8


TSSOP8


WLCSP10


UFDFPN8


UFDFPN12

Near Field Communication™ CERTIFIED

Product status
ST25DV04KC
ST25DV16KC
ST25DV64KC

Features

Includes ST state-of-the-art patented technology

I²C interface

- Two-wire I²C serial interface supports 1 MHz protocol
- Single supply voltage: 1.8 to 5.5 V
- Multiple byte programming (up to 256 bytes)
- Configurable I²C target address

Contactless interface

- Based on ISO/IEC 15693
- NFC Forum Type 5 tag certified by the NFC Forum
- Support of all ISO/IEC 15693 modulations, coding, subcarrier modes and data rates
- Custom fast read access up to 53 Kbit/s
- Single and multiple blocks read (same for extended commands)
- Single and multiple (up to four) blocks write (same for extended commands)
- Internal tuning capacitance: 28.5 pF

Memory

- Up to 64 Kbit of EEPROM (depending on version)
- I²C interface accesses bytes
- RF interface accesses blocks of 4 bytes
- Write time:
  - From I²C: typical 5 ms for 1 up to 16 bytes
  - From RF: typical 5 ms for one block
- Data retention: 40 years
- Write cycles endurance:
  - 1 million at 25 °C
  - 600k at 85 °C
  - 500k at 105 °C
  - 400k at 125 °C

Fast transfer mode

- Fast data transfer between I²C and RF interfaces
- Half-duplex 256 bytes dedicated buffer

Energy harvesting

- Analog output pin to power external components

Data protection

- User memory: one to four configurable areas, protectable in read and/or write by three 64-bit passwords in RF and one 64-bit password in I²C

- System configuration: protected in write by a 64-bit password in RF and a 64-bit password in I²C

GPO

- Interruption pin configurable on multiple RF events (field change, memory write, activity, fast transfer, user set/reset/pulse), and I²C events (memory write completed, RF switch off)
- Open drain or CMOS output (depending on version)

Low power mode (10-ball and 12-pin packages only)

- Input pin to trigger low power mode

RF management

- RF command interpreter enabled/disabled from I²C host controller
- I²C priority: immediate RF switch off from I²C

Temperature range

- Range 6:
  - From -40 °C to 85 °C
- Range 8:
  - From -40 °C to 105 °C (UDFPN8 and UFDFPN12 only)
  - From -40 °C to 125 °C (SO8N and TSSOP8 only, 105 °C max on RF interface)

Package

- 8-pin, 10-ball, and 12-pin
- ECOPACK2 (RoHS compliant)

1 Description

The ST25DV04KC, ST25DV16KC, and ST25DV64KC devices (hereinafter referred collectively to as ST25DVxxKC) are NFC RFID tags offering, respectively, 4, 16, and 64-Kbit of electrically erasable programmable memory (EEPROM). These devices feature two interfaces: the first one is an I²C serial link that can be operated from a DC power supply, the second one is an RF link activated when the device acts as a contactless memory powered by the received carrier electromagnetic wave.

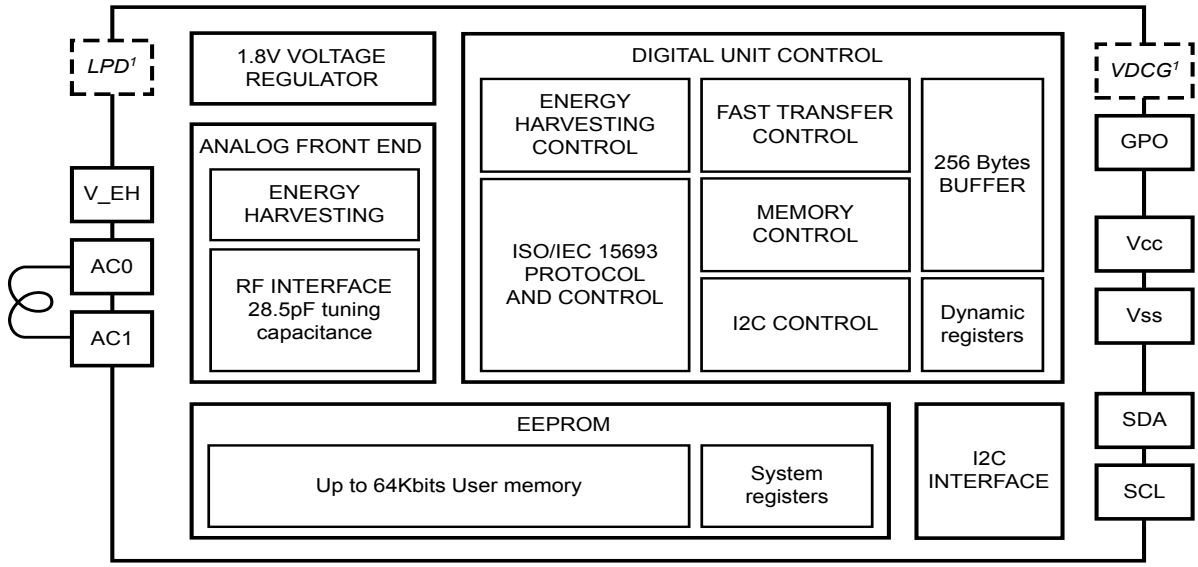
In I²C mode, the user memory contains 512 (ST25DV04KC), 2048 (ST25DV16KC), or 8192 (ST25DV64KC) bytes, which can be split in four flexible and protectable areas.

In RF mode, following ISO/IEC 15693 or NFC Forum Type 5 recommendations, the user memory contains 128 (ST25DV04KC), 512 (ST25DV16KC), or 2048 (ST25DV64KC) blocks of 4 bytes, which can be split in four flexible and protectable areas.

These devices offer a fast transfer mode between the RF and contact interfaces, thanks to a 256 bytes volatile buffer (also called mailbox). In addition, the GPO pin provides data about incoming events, like RF field detection, RF activity in progress, or mailbox message availability. An energy harvesting feature is also available.

1.1 Block diagram

Figure 1. Block diagram



1. V_{DCG} and LPD are included in the 10-ball and 12-pin packages only.

1.2 Packaging

The devices are provided in 8-pin, 10-ball, and 12-pin packages:

- SO8N, TSSOP8, or UDFPN8 8-pin packages for the open drain version of interrupt output.
- 10-ball (WLCSP) and 12-pin (UFDFPN12) for the CMOS version of interrupt output. These packages include an additional LPD pin to minimize the standby consumption.

Table 1. 8-pin packages signal names

Signal name	Function	Direction
V_EH	Energy harvesting	Power output
GPO	Interrupt output	Output
SDA	Serial data	I/O
SCL	Serial clock	Input
AC0, AC1	Antenna coils	-
V_{CC}	Supply voltage	Power
V_{SS}	Ground	-
EP (1)	Exposed pad	Must be left floating

1. *Available only on UFDPN8 packages.*

Figure 2. SO8N package connections

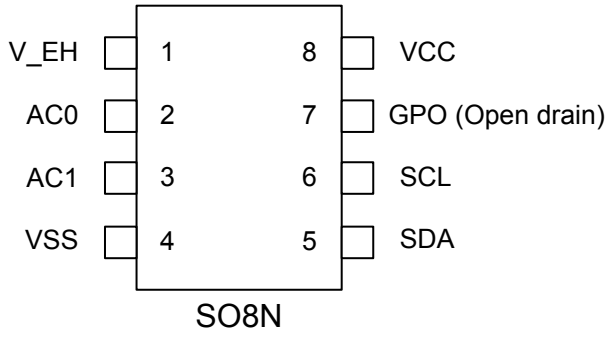


Figure 3. TSSOP8 package connections

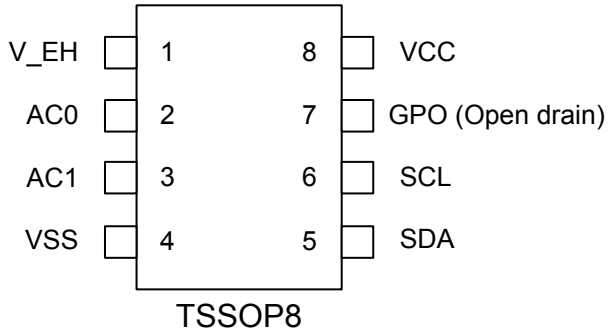


Figure 4. **UFDFN8 package connections**

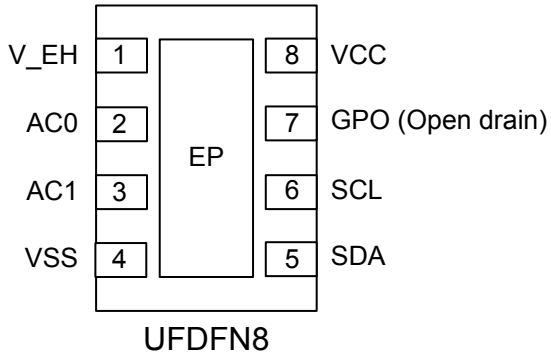


Table 2. **10-ball package signal names**

Signal name	Function	Direction
V_EH	Energy harvesting	Power output
GPO	Interrupt output	Output
SDA	Serial data	I/O
SCL	Serial clock	Input
AC0, AC1	Antenna coils	-
V_{CC}	Supply voltage	Power
V_{SS}	Ground	-
LPD	Low power down mode	Input
V_{DCG}	Supply voltage for GPO driver	Power

Figure 5. **WLCSP10 package connections**

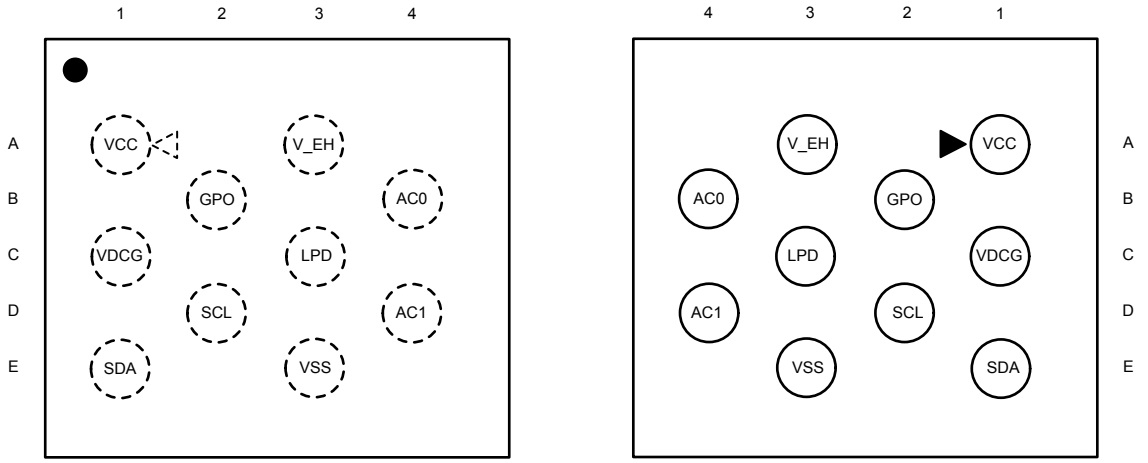
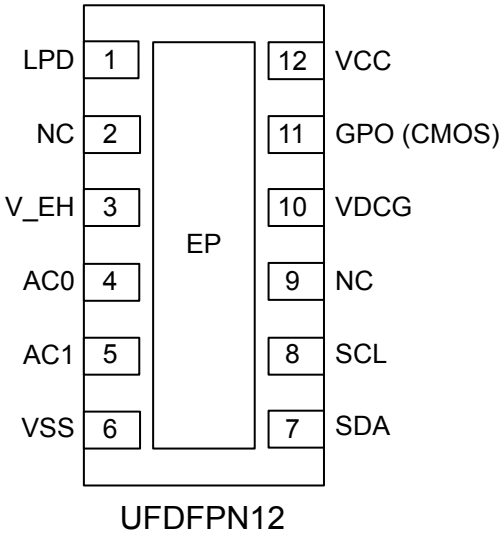


Table 3. 12-pin package signal names

Signal name	Function	Direction
V_EH	Energy harvesting	Power output
GPO	Interrupt output	Output
SDA	Serial data	I/O
SCL	Serial clock	Input
AC0, AC1	Antenna coils	-
V_{CC}	Supply voltage	Power
V_{SS}	Ground	-
LPD	Low power down mode	Input
V_{DCG}	Supply voltage for GPO driver	Power
NC	Not connected	-
EP	Exposed pad	Must be left floating

Figure 6. UFDFPN12 package connections



UFDFPN12

2 Signal descriptions

2.1 Serial link (SCL, SDA)

2.1.1 Serial clock (SCL)

This input signal is used to strobe all data in and out. In applications where this signal is used by target devices to synchronize the bus to a slower clock, the bus controller must have an open drain output, and a pull-up resistor must be connected from serial clock (SCL) to V_{CC}. See Section 9.2: I²C parameters to know how to calculate the value of this resistor.

2.1.2 Serial data (SDA)

This bidirectional signal is used to transfer data in or out. It is an open drain output that may be wire OR-ed with other open drain or open collector signals on the bus. A pull-up resistor must be connected from serial data (SDA) to V_{CC} (Figure 82 indicates how to calculate the value of the resistor).

2.2 Power control (V_{CC}, LPD, V_{SS})

2.2.1 Supply voltage (V_{CC})

This pin can be connected to an external DC supply voltage.

Note: An internal voltage regulator allows the external voltage applied on V_{CC} to supply the ST25DVxxKC, while preventing the internal power supply (rectified RF waveforms) to output a DC voltage on the V_{CC} pin.

2.2.2 Low power down (LPD)

This input signal is used to control an internal 1.8 V regulator delivering the internal supply. When LPD is high, the regulator is shut off, and its consumption is reduced below 1µA. The regulator has a turn on time in the 100 µs range, to be added to the boot duration, before the device becomes fully operational. The LPD pin is internally pulled down by a 10kohm resistor and can be left unconnected. The internal pull-down is deactivated when the voltage on the pin exceeds the VIH_LPD level defined in Table 249 and Table 250.

This feature is available only on the 10-ball and 12-pin packages.

2.2.3 Ground (V_{SS})

V_{SS} is the reference for the V_{CC} and V_{DCG} supply voltages and V_EH analogic output voltage.

2.3 RF link (AC0 AC1)

2.3.1 Antenna coil (AC0, AC1)

These inputs are used to connect the device to an external coil. Do not connect them to any other DC or AC path.

When correctly tuned, the coil is used to power and access the device using the ISO/IEC 15693 and ISO 18000-3 mode 1 protocols.

2.4 Process control (GPO, V_{DCG})

2.4.1 Driver supply voltage (V_{DCG})

This pin, available only on 10-ball and 12-pin packages, can be connected to an external DC supply voltage. It supplies the GPO (CMOS) driver block.

The devices cannot be powered by V_{DCG}. If V_{DCG} is left floating, there is no information available on the GPO (CMOS) pin.

2.4.2 General purpose output (GPO)

The devices feature a configurable output GPO pin used to provide RF and I²C activity information to an external device.

Depending on the package, there are two types of GPO output:

- The 8-pin package offers an open drain GPO output. This GPO pin must be connected to an external pull-up resistor (> 4.7 kΩ) to operate.
- The 10-ball and 12-pin packages offer a CMOS GPO output, which requires to connect the V_{DCG} pin to an external power supply. The interrupt consists in setting the state to a high level, or outputting a positive pulse on the GPO pin.

GPO pin is a configurable output signal, and can mix several interruption modes. By default, the GPO register sets the interruption mode as an RF field change detector. It is able to raise various events, like RF activity, Memory write completion, or fast transfer actions. It can authorize the RF side to directly drive the GPO pin using the Manage GPO command, to set the output state, or emit a single pulse (for example, to wake up an application). See Section 5.4: GPO for details.

2.5 Energy harvesting analog output (V_EH)

This analog output pin is used to deliver the analog voltage V_EH available when the Energy harvesting mode is enabled and if the RF field strength is sufficient. When the Energy harvesting mode is disabled or the RF field strength is not sufficient, V_EH pin is in high-Z state (see Section 5.5 for details).

Energy harvesting voltage output is not regulated.

3 Power management

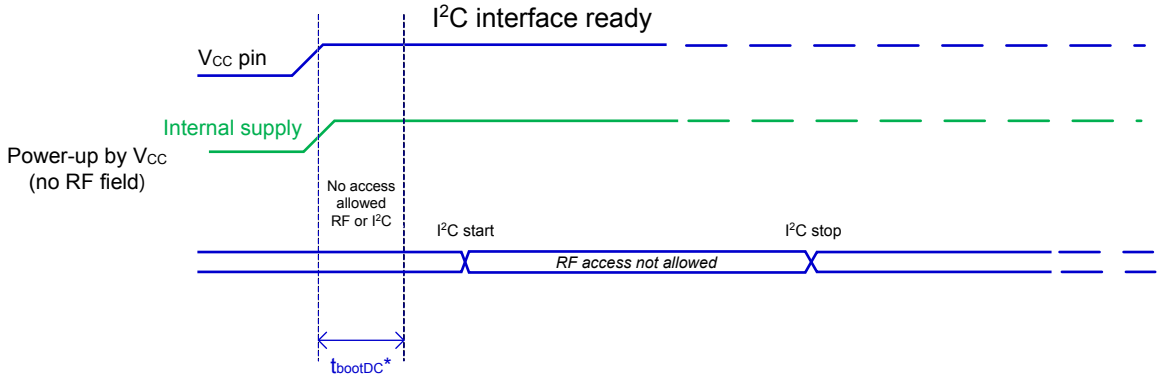
3.1 Wired interface

Operating supply voltage V_{CC}

In contact mode, prior to selecting the memory and issuing instructions to it, a valid and stable V_{CC} voltage within the specified [V_{CC}(min), V_{CC}(max)] range must be applied (see Table 246. I²C operating conditions). To maintain a stable DC supply voltage, it is recommended to decouple the V_{CC} line with a suitable capacitor (usually of the order of 10 nF and 100 nF) close to the V_{CC}/V_{SS} package pins.

This voltage must remain stable and valid until the end of the transmission of the instruction and, for a Write instruction, until the completion of the internal I²C write cycle (t_W). Instructions are not taken into account until completion of the boot sequence (see the figure below).

Figure 7. Power-up sequence (no RF field, LPD pin tied to V_{SS}, or package without LPD pin)



* If the LPD pin follows V_{CC} before going low, t_{bootDC} is equal to t_{boot_LPD} and starts only when LPD reaches the low level.
* When RF field is present before V_{CC} set up, boot is already done and t_{bootDC} is 0.

Power-up conditions

When the power supply is turned on, V_{CC} rises from V_{SS} to V_{CC}. V_{CC} must not rise faster than 1 V/µs.

Device reset in I²C mode

To prevent inadvertent write operations during power-up, a power-on reset (POR) circuit is included. At power-up the device does not respond to any I²C instruction until V_{CC} has reached the POR threshold voltage (this threshold is lower than the minimum V_{CC} operating voltage defined in Table 246. I²C operating conditions). When V_{CC} exceeds this threshold, the device is reset and enters the Standby power mode. The device must not be accessed until V_{CC} has reached a valid and stable voltage, within the specified [$V_{CC(min)}$, $V_{CC(max)}$] range, and before the time necessary to set up (t_{bootDC}) has passed.

In the version supporting LPD pin, the boot takes place only when LPD goes low.

In a similar way, during power-down, as soon as V_{CC} drops below the POR threshold, the device stops responding to any instruction sent to it, and the I²C address counter is reset.

Power-down mode

During power-down the device must be in Standby power mode (mode reached after decoding a Stop condition, assuming that there is no internal write cycle in progress).

3.2 Contactless interface

Device set in RF mode

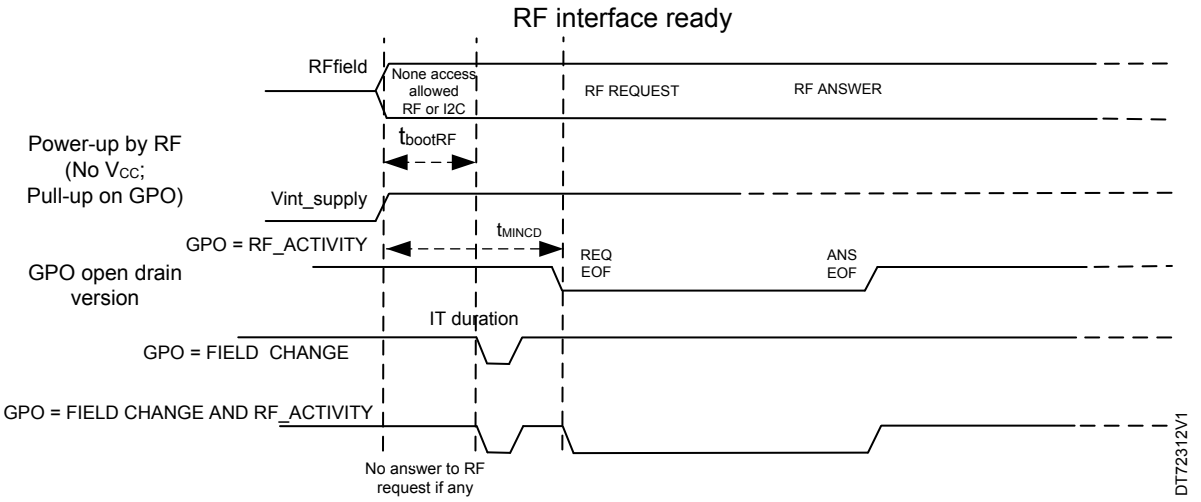
To ensure a proper boot of the RF circuitry, the RF field must be turned ON without any modulation for a minimum period of time t_{bootRF}. Before this time, the device ignores all received RF commands (see Figure 8).

Device reset in RF mode

To ensure a proper reset of the RF circuitry, the RF field must be turned off (100% modulation) for a minimum t_{RF_OFF} period of time.

The RF access can be temporarily or indefinitely disabled by setting the appropriate value in the RF_MNGT or RF_MNGT_Dyn registers.

Figure 8. RF power-up sequence (no DC supply)



4 Memory management

4.1 Memory organization overview

The ST25DVxxKC memory is divided in four main areas:

- User memory
- Dynamic registers
- Fast transfer mode buffer
- System configuration area

The user memory can be divided into four flexible user areas. Each area can be individually read- and/or - write-protected with one out of three specific 64-bit password.

The dynamic registers are accessible by RF or I^2C host and provide dynamic activity status or allow temporary activation or deactivation of some features.

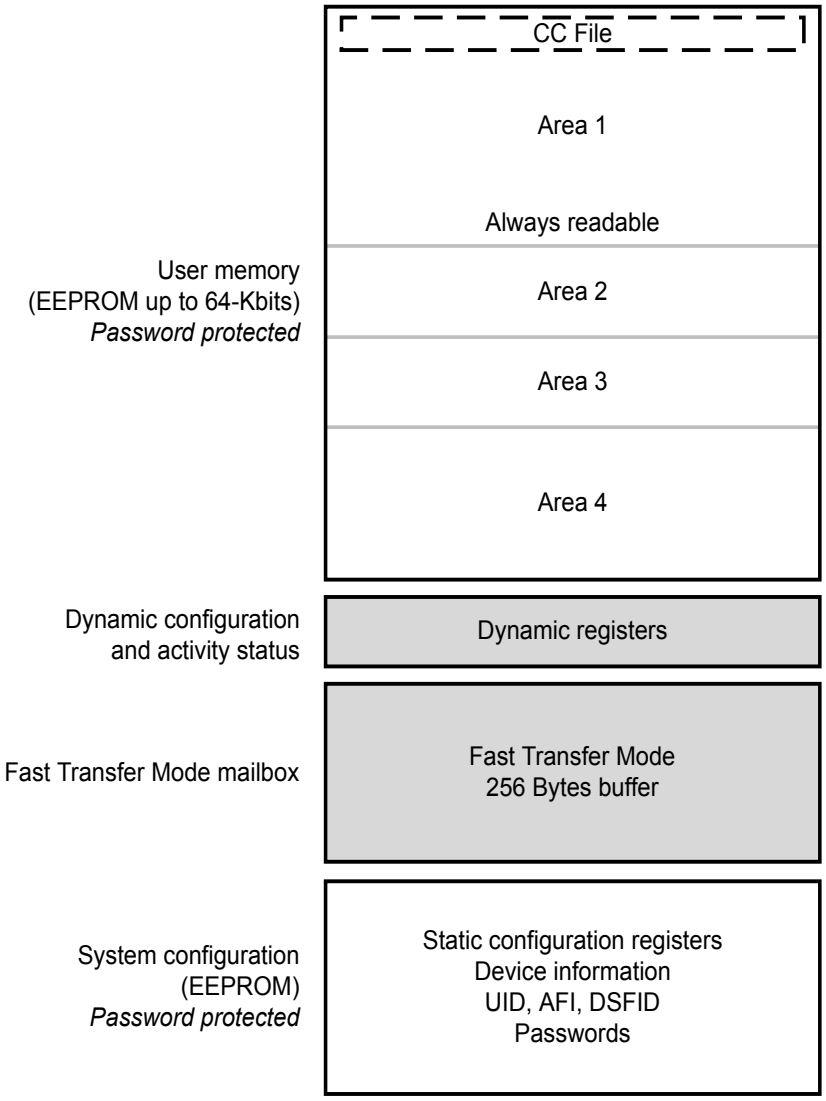
The devices also provide a 256-byte fast transfer mode buffer, acting as a mailbox between RF and I^2C interface, allowing fast data transfer between contact and contactless worlds.

Finally, the system configuration area contains static registers to configure all device features, which can be tuned by the user. Its access is protected by a 64-bit configuration password.

This system configuration area also includes read-only device information such as IC reference, memory size or IC revision, as well as a 64-bit block used to store the 64-bit unique identifier (UID), the AFI (default 00h) and DSFID (default 00h) registers. The UID is compliant with the ISO 15693 description, and its value is used during the anticollision sequence (Inventory). The UID value is written by ST during manifacturing. The AFI register stores the application family identifier. The DSFID register stores the data storage family identifier used in the anticollision algorithm.

The system configuration area includes five additional 64-bit blocks that store an I^2C password plus three RF user area access passwords and an RF configuration password.

Figure 9. Memory organization



4.2 User memory

This memory is accessible from both RF contactless and I^2C wired interfaces.

From RF interface, user memory is addressed as 4-byte blocks, starting at address 0. RF extended read and write commands can be used to address all memory blocks. Other read and write commands address only up to block FFh.

From I^2C interface, user memory is addressed as bytes, starting at address 0. Device select must set E2 = 0 (see Section 6.3: Device addressing). User memory can be read in continuity. Unlike the RF interface, there is no roll-over when the requested address reaches the end of the memory capacity.

Table 4 shows how memory is seen from the RF and I^2C interfaces.

Table 4. User memory as seen by RF and by I²C

RF command (block addressing)	User memory				I²C command (byte addressing)
Read Single Block Read Multiple Blocks Fast Read Single Block Fast Read Multiple Blocks Write Single Block Write Multiple Blocks Ext Read Single Block Ext Read Multiple Blocks Fast Ext Read Single Block Fast Ext Read Multi. Blocks Ext Write Single Block Ext Write Multiple Blocks	RF block (00)00h				I²C Read command I²C Write command Device select E2 = 0
	I²C byte 0003h	I²C byte 0002h	I²C byte 0001h	I²C byte 0000h	
	RF block (00)01h				
	I²C byte 0007h	I²C byte 0006h	I²C byte 0005h	I²C byte 0004h	
	RF block (00)02h				
	I²C byte 000Bh	I²C byte 000Ah	I²C byte 0009h	I²C byte 0008h	
	....				
	RF block (00)7Fh (1)				
	I²C byte 01FFh	I²C byte 01FEh	I²C byte 01FDh	I²C byte 01FCh	
	....				
	RF block (00)FFh (2)				
	I²C byte 03FFh	I²C byte 03FEh	I²C byte 03FDh	I²C byte 03FCh	
Ext Read Single Block Ext Read Multiple Blocks Fast Ext Read Single Block Fast Ext Read Multi. Blocks Ext Write Single Block Ext Write Multiple Blocks	RF block 0100h				
	I²C byte 0403h	I²C byte 0402h	I²C byte 0401h	I²C byte 0400h	
	....				
	RF block 01FFh (3)				
	I²C byte 07FFh	I²C byte 07FEh	I²C byte 07FDh	I²C byte 07FCh	
	....				
	RF block 07FFh (4)				
	I²C byte 1FFFh	I²C byte 1FFEh	I²C byte 1FFDh	I²C byte 1FFCh	

1. *Last block of user memory in ST25DV04KC.*
2. *Last block accessible with Read Single Block, Read Multiple Blocks, Fast Read Single Block, Fast Read Multiple Blocks, Write Single Block and Write Multiple Blocks RF commands.*
3. *Last block of user memory in ST25DV16KC.*
4. *Last block of user memory in ST25DV64KC.*

Note: In the factory all blocks of user memory are initialized to 00h.

4.2.1 User memory areas

The user memory can be split into different areas, each one with a distinct access privilege.

RF and I^2C write commands are legal only within the same area:

- In RF, Write Multiple Blocks and Extended Write Multiple Blocks command are not executed and return the error 0Fh if addresses cross an area border.
- In I^2C, a sequential write is not executed and all bytes with addresses crossing the area border are not acknowledged.

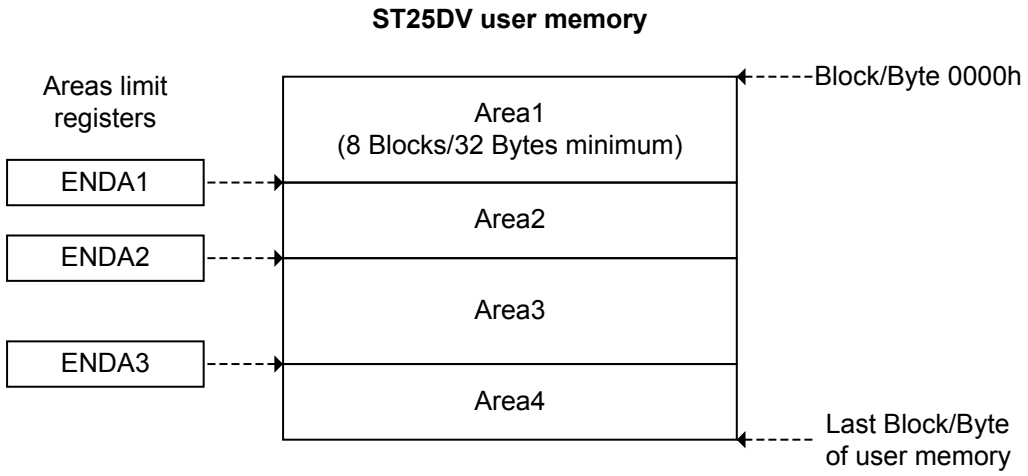
RF and I^2C read commands are allowed over multiple areas:

- In RF, Read Multiple Blocks and Extended read multiple Blocks (and related Fast commands) are executed and return all readable blocks until reaching a non readable block (address read protected or non available), even if addresses cross area borders.
- In I^2C, sequential read returns all readable bytes until reaching a non readable byte (address read protected or non available) even if addresses cross area borders. Non readable bytes return value FFh.

Each user memory area is defined by its ending address ENDA. The starting address is implicitly defined by the end of the preceding area.

There are three ENDA registers in the configuration system memory, used to define the end addresses of Area 1, Area 2, and Area 3. The end of Area 4 is always the last block/byte of memory, and is not configurable.

Figure 10. User memory areas



On factory delivery all ENDAs are set to maximum value, only Area1 exists and includes the full user memory.

A granularity of 8 blocks (32 bytes) is offered to code area ending points.

Areas end limits are coded in ENDA registers as follows:

- Last RF block address of area = 8 x $ENDA_i$ + 7 → $ENDA_i$ = int(Last $Area_i$ RF block address / 8)
- Last I^2C byte address of area = 32 * $ENDA_i$ + 31→ $ENDA_i$ = int(Last $Area_i$ I^2C byte address / 32)
- As a consequence, ENDA1 = 0 minimum size of Area 1 is 8 blocks (32 bytes).

Table 5. Maximum user memory block and byte addresses and $ENDA_i$ value

Device	Last user memory block address seen by RF	Last user memory byte address seen by I^2C	Maximum $ENDA_i$ value
ST25DV04KC	007Fh	01FFh	0Fh
ST25DV16KC	01FFh	07FFh	3Fh
ST25DV64KC	07FFh	1FFFh	FFh

Table 6. Areas and limit calculation from ENDA$_i$ registers

Area	Seen from RF interface	Seen from I²C interface
Area 1	Block 0000h … Block (ENDA1 * 8) + 7	Byte 0000h … Byte (ENDA1 * 32) + 31
Area 2	Block (ENDA1 + 1) * 8 … Block (ENDA2 * 8) + 7	Byte (ENDA1 + 1) * 32 … Byte (ENDA2 * 32) + 31
Area 3	Block (ENDA2 + 1) * 8 … Block (ENDA3 * 8) + 7	Byte (ENDA2 + 1) * 32 … Byte (ENDA3 * 32) + 31
Area 4	Block (ENDA3 + 1) * 8 … Last memory block	Byte (ENDA3 + 1) * 32 … Last memory byte

The organization of user memory in areas has the following characteristics:

- At least one area exists (Area1), starting at block/byte address 0000h and finishing at ENDA1, with ENDA1 = ENDA2 = ENDA3 = End of user memory (factory setting).
- Two areas can be defined by setting ENDA1 < ENDA2 = ENDA3 = End of user memory.
- Three areas can be defined by setting ENDA1 < ENDA2 < ENDA3 = End of user memory.
- A maximum of four areas can be defined by setting ENDA1 < ENDA2 < ENDA3 < End of user memory.
- Area 1 specificities
  - Start of Area1 is always block/byte address 0000h.
  - Area1 minimum size is 8 blocks (32 bytes) when ENDA1 = 00h.
  - Area1 is always readable.
- The last area always finishes on the last user memory block/byte address (ENDA4 does not exist).
- All areas are contiguous: end of Area(n) + one block/byte address is always the start of Area(n+1).

Area size programming

RF user must first open the RF configuration security session to write ENDAi registers.

I²C host must first open I²C security session to write ENDA registers.

When programming an ENDA register, the following rule must be respected:

- ENDA$_i$ - 1 < ENDA$_i$ ≤ ENDA$_i$+1 = End of memory.

This means that prior to programming any ENDA register, the following one (ENDA$_i$+1) must first be programmed to the last block/byte of memory:

- Successful ENDA3 programming condition: ENDA2 < ENDA3 ≤ End of user memory
- Successful ENDA2 programming condition: ENDA1 < ENDA2 ≤ ENDA3 = End of user memory
- Successful ENDA1 programming condition: ENDA1 ≤ ENDA2 = ENDA 3 = End of user memory

If this rule is not respected, an error 0Fh is returned in RF, NoAck is returned in I²C, and programming is not done.

To respect this rule, the following procedure is recommended when programming area size (even for changing only one area size):

1. Ends of Areas 3 and 2 must first be set to the end of memory while respecting the following order:
   a. If ENDA3 ≠ end of user memory, then set ENDA3 = end of memory; else, do not write ENDA3.
   b. If ENDA2 ≠ end of user memory, then set ENDA2 = end of memory; else, do not write ENDA2.
2. Then, desired area limits can be set respecting the following order:
   a. Set new ENDA1 value.
   b. Set new ENDA2 value, with ENDA2 > ENDA1.
   c. Set new ENDA3 value, with ENDA3 > ENDA2.

Example of successive user memory area setting (for ST25DV64KC):

1. Initial state, two areas are defined:
   a. ENDA1 = 10h (Last block of Area 1: (10h x 8) + 7 = 0087h)
   b. ENDA2 = FFh (Last block of Area 2: (FFh x 8) + 7 = 07FFh)
   c. ENDA3 = FFh (No Area 3)
      - Area 1 from Block 0000h to 0087h (136 blocks)
      - Area 2 from Block 0088h to 07FFh (1912 blocks)
      - There is no Area 3
      - There is no Area 4
2. Split of user memory in four areas:
   a. ENDA3 is not updated as it is already set to end of memory
   b. ENDA2 is not updated as it is already set to end of memory
   c. Set ENDA1 = 3Fh (Last block of Area 1: (3Fh x 8) + 7 = 01FFh)
   d. Set ENDA2 = 5Fh (Last block of Area 1: (5Fh x 8) + 7 = 02FFh)
   e. Set ENDA3 = BFh (Last block of Area 1: (BFh x 8) + 7 = 05FFh)
      - Area1 from Block 0000h to 01FFh (512 blocks)
      - Area2 from Block 0200h to 02FFh (256 blocks)
      - Area3 from Block 0300h to 05FFh (768 blocks)
      - Area4 from Block 0600h to 07FFh (512 blocks)
3. Return to a split in two equal areas:
   a. Set ENDA3 = FFh
   b. Set ENDA2 = FFh
   c. Set ENDA1 = 7Fh (Last block of Area 1: (7Fh x 8) + 7 = 03FFh)
      - Area1 from Block 0000h to 03FFh (1024 blocks)
      - Area2 from Block 0400h to 07FFh (1024 blocks)
      - There is no Area3
      - There is no Area 4

Programming ENDA3 to FFh in step 2.a would have resulted in into an error, since rule ENDAi-1 < ENDAi is not respected (ENDA2 = ENDA3 in that case).

Registers for user memory area configuration

Table 7. ENDA1 access

RF		I^2C	
Command	**Type**	**Address**	**Type**
Read Configuration (cmd code A0h) @05h Write Configuration (cmd code A1h) @05h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0005h	R always, W if I^2C security session is open

Table 8. ENDA1

Bit	Name	Function	Factory Value
b7-b0	ENDA1	End Area 1 = 8*ENDA1+7 when expressed in blocks (RF) End Area 1 = 32*ENDA1+31 when expressed in bytes (I^2C)	ST25DV04KC: 0Fh ST25DV16KC: 3Fh ST25DV64KC: FFh

Note: Refer to Table 13. System configuration memory map for ENDA1 register.

Table 9. ENDA2 access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @07h Write Configuration (cmd code A1h) @07h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0007h	R always, W if I^2C security session is open

Table 10. ENDA2

Bit	Name	Function	Factory Value
b7-b0	ENDA2	End Area 2 = 8 x ENDA2 + 7 when expressed in blocks (RF) End Area 2 = 32*ENDA2 + 31 when expressed in bytes (I^2C)	ST25DV04KC: 0Fh ST25DV16KC: 3Fh ST25DV64KC: FFh

Note: Refer to Table 13. System configuration memory map for ENDA2 register.

Table 11. ENDA3 access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @09h Write Configuration (cmd code A1h) @09h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0009h	R always, W if I^2C security session is open

Table 12. ENDA3

Bit	Name	Function	Factory Value
b7-b0	ENDA3	End Area 3 = 8 x ENDA3 + 7 when expressed in blocks (RF) End Area 3 = 32 x ENDA3 + 31 when expressed in bytes (I^2C)	ST25DV04KC: 0Fh ST25DV16KC: 3Fh ST25DV64KC: FFh

Note: Refer to Table 13. System configuration memory map for ENDA3 register.

4.3 System configuration area

In addition to EEPROM user memory, ST25DVxxKC includes a set of static registers located in the system configuration area memory (EEPROM nonvolatile registers). Those registers are set during device configuration (that is: area extension), or by the application (that is: area protection). Static registers content is read during the boot sequence and define basic ST25DVxxKC behaviour.

In RF, the static registers located in the system configuration area can be accessed via the dedicated Read Configuration and Write Configuration commands, with a pointer acting as the register address.

The RF configuration security session must first be open, by presenting a valid RF configuration password, to grant write access to system configuration registers.

The system configuration area write access by RF can also be deactivated by I²C host.

In I²C static registers located in the system configuration area can be accessed with I²C read and write commands with device select E2=1, E1=1. Readable system areas could be read in continuity.

I²C security session must first be open, by presenting a valid I²C password, to grant write access to system configuration registers.

The following table shows the complete map of the system configuration area, as seen by the RF and I²C interface.

Table 13. System configuration memory map

RF access		Static Register		I²C access		
Address	Type	Name	Function	Device select	Address	Type
00h	RW (1)	GPO1	Enable/disable GPO output and GPO ITs for RF events	E2=1, E1=1	0000h	RW (2)
01h	RW(1)	GPO2	Enable/disable GPO ITs for I²C events and set Interruption pulse duration	E2=1, E1=1	0001h	RW(2)
02h	RW(1)	EH_MODE	Energy harvesting default strategy after Power ON	E2=1, E1=1	0002h	RW(2)
03h	RW(1)	RF_MNGT	RF interface state after power ON	E2=1, E1=1	0003h	RW(2)
04h	RW(1)	RFA1SS	Area1 RF access protection	E2=1, E1=1	0004h	RW(2)
05h	RW(1)	ENDA1	Area 1 ending point	E2=1, E1=1	0005h	RW(2)
06h	RW(1)	RFA2SS	Area2 RF access protection	E2=1, E1=1	0006h	RW(2)
07h	RW(1)	ENDA2	Area 2 ending point	E2=1, E1=1	0007h	RW(2)
08h	RW(1)	RFA3SS	Area3 RF access protection	E2=1, E1=1	0008h	RW(2)
09h	RW(1)	ENDA3	Area 3 ending point	E2=1, E1=1	0009h	RW(2)
0Ah	RW(1)	RFA4SS	Area4 RF access protection	E2=1, E1=1	000Ah	RW(2)
No access		I2CSS	Area 1 to 4 I²C access protection	E2=1, E1=1	000Bh	RW(2)
N/A	RW (3) (4)	LOCK_CCFILE	Blocks 0 and 1 RF Write protection	E2=1, E1=1	000Ch	RW(2)
0Dh	RW(1)	FTM	Fast transfer mode authorization and watchdog setting.	E2=1, E1=1	000Dh	RW(2)
No access		I2C_CFG	I^2C target address configuration and enable/disable RF switch off from I^2C.	E2=1, E1=1	000Eh	RW(2)
0Fh	RW(1)	LOCK_CFG	Protect RF Write to system configuration registers	E2=1, E1=1	000Fh	RW(2)
N/A	WO (5)	LOCK_DSFID	DSFID lock status	E2=1, E1=1	0010h	RO
NA	WO (6)	LOCK_AFI	AFI lock status	E2=1, E1=1	0011h	RO
N/A	RW(5)	DSFID	DSFID value	E2=1, E1=1	0012h	RO
N/A	RW(6)	AFI	AFI value	E2=1, E1=1	0013h	RO
N/A	RO	MEM_SIZE	Memory size value in blocks, 2 bytes	E2=1, E1=1	0014h to 0015h	RO
	RO	BLK_SIZE	Block size value in bytes	E2=1, E1=1	0016h	RO
N/A	RO	IC_REF	IC reference value	E2=1, E1=1	0017h	RO

RF access		Static Register		I²C access		
Address	Type	Name	Function	Device select	Address	Type
NA	RO	UID	Unique identifier, 8 bytes	E2=1, E1=1	0018h to 001Fh	RO
No access		IC_REV	IC revision	E2=1, E1=1	0020h	RO
		-	ST reserved	E2=1, E1=1	0021h	RO
		-	ST reserved	E2=1, E1=1	0022h	RO
		-	ST reserved	E2=1, E1=1	0023h	RO
		I2C_PWD	I²C security session password, 8 bytes	E2=1, E1=1	0900h to 0907h	R/W (7) (8)
N/A	WO (9)	RF_PWD_0	RF configuration security session password, 8 bytes	No access		
N/A	WO(9)	RF_PWD_1	RF user security session password 1, 8 bytes			
N/A	WO(9)	RF_PWD_2	RF user security session password 2, 8 bytes			
N/A	WO(9)	RF_PWD_3	RF user security session password 3, 8 bytes			

1. *Write access is granted if RF configuration security session is open and configuration is not locked (LOCK_CFG register equals to 0).*
2. *Write access if the I²C security session is open.*
3. *Write access to bit 0 if Block 00h is not already locked and to bit 1 if Block 01h is not already locked.*
4. *LOCK_CCFILE content is only readable through reading the Block Security Status of blocks 00h and 001h (see Section 5.6.3: User memory protection)*
5. *Write access if DSFID is not locked.*
6. *Write access if the AFI is not locked.*
7. *Write access with I²C Write Password command, only after presenting a correct I²C password.*
8. *Read access is granted if I²C security session is open.*
9. *Write access only if corresponding RF security session is open.*

4.4 Dynamic configuration

ST25DVxxKC has a set of dynamic registers that allow temporary modification of its behavior or report on its activity. Dynamic registers are volatile and not restored to their previous values after POR.

Some static registers have an image in dynamic registers: dynamic register value is initialized with the static register value and can be updated by the application to temporarily modify the device behavior (for example, to set or reset energy harvesting). When a valid change occurs in a static register, in RF or I^2C, the corresponding dynamic register is automatically updated.

Other dynamic registers, automatically updated, contain indication on device activity (for instance: IT_STS_Dyn gives the interruption's status, MB_CTRL_Dyn gives the fast transfer mode mailbox control).

In RF, dynamic registers can be accessed via dedicated (Fast) Read Dynamic Configuration and (Fast) Write Dynamic Configuration commands, with a pointer acting as the register address. No password is needed to access dynamic registers.

In I^2C, dynamic registers can be accessed with I^2C read and write commands with device select E2 = 0, E1 = 1. Dynamic registers can be read in continuity, no password is needed to access them. Dynamic registers and fast transfer mode mailbox can be read in continuity, but not written in continuity.

The table below shows the complete map of dynamic registers, as seen by RF and I^2C interfaces.

Table 14. Dynamic registers memory map

RF access		Dynamic Registers		I^2C access		
Address	Type	Name	Function	Device select	Address	Type
00h	RO	GPO_CTRL_Dyn	GPO control	E2 = 0, E1 = 1	2000h	R/W
No access		-	Reserved	E2 = 0, E1 = 1	2001h	RO
02h	R/W	EH_CTRL_Dyn	Energy Harvesting management and usage status	E2 = 0, E1 = 1	2002h	R/W
No access		RF_MNGT_Dyn	RF interface usage management	E2 = 0, E1 = 1	2003h	R/W
		I2C_SSO_Dyn	I^2C security session status	E2 = 0, E1 = 1	2004h	RO
		IT_STS_Dyn	Interruptions status	E2 = 0, E1 = 1	2005h	RO
0Dh	R/W	MB_CTRL_Dyn	Fast transfer mode control and status	E2 = 0, E1 = 1	2006h	R/W
NA	RO	MB_LEN_Dyn	Length of fast transfer mode message	E2 = 0, E1 = 1	2007h	RO

4.5 Fast transfer mode mailbox

ST25DVxxKC fast transfer mode uses a dedicated mailbox buffer for transferring messages between RF and I^2C worlds. This mailbox contains up to 256 Bytes of data which are filled from the first byte.

Fast transfer mode mailbox is accessed in bytes from both RF and I^2C.

In RF, mailbox is read via a dedicated (Fast) Read Message command. Read can start from any address value inside the mailbox, between 00h and FFh. Writing in the mailbox is done via the (Fast) Write Message command in one shot, always starting at mailbox address 00h. No password is needed to access mailbox from RF, but fast transfer mode must be enabled.

In I^2C, mailbox read can start from any address value between 2008h and 2107h. Write mailbox MUST start from address 2008h to a max of 2107h. No password is needed to access mailbox from I^2C, but fast transfer mode must be enabled.

The table below shows the map of fast transfer mode mailbox, as seen by RF interface and by I^2C interface.

Table 15. Fast transfer mode mailbox memory map

RF access		Fast transfer mode buffer		I^2C access		
Address	Type	Name	Function	Device select	Address	Type
00h	R/W	MB_Dyn Byte 0	Fast transfer mode buffer (256-Bytes)	E2=0, E1=1	2008h	R/W
01h	R/W	MB_Dyn Byte 1		E2=0, E1=1	2009h	R/W
…	…	…		E2=0, E1=1	...	...
FEh	R/W	MB_Dyn Byte 254		E2=0, E1=1	2106h	R/W
FFh	R/W	MB_Dyn Byte 255		E2=0, E1=1	2107h	R/W

5 ST25DVxxKC specific features

The devices offer the following features:

- A fast transfer mode (FTM), to achieve a fast link between RF and contact worlds, via a 256-byte buffer called mailbox. This mailbox dynamic buffer of 256 bytes can be filled or emptied via either RF or I²C.
- A GPO pin, which indicates incoming events to the contact side, like RF events (RF field changes, RF activity, RF writing completion, or mailbox message availability) or I²C events (I²C write completion, RF switch off from I²C).
- An energy harvesting element to deliver mW of power when external conditions make it possible.
- RF management, which allows ST25DVxxKC to ignore RF requests.

All these features can be programmed by setting the static and/or dynamic registers. The devices can be partially customized using configuration registers located in the system area. These registers are:

- Dedicated to data memory organization and protection: $ENDA_i$, I2CSS, RFAiSS, LOCK_CCFILE.
- Dedicated to fast transfer mode: FTM
- Dedicated to observation: GPO, IT_TIME
- Dedicated to RF: RF_MNGT, EH_MODE
- Dedicated the structure: LOCK_CFG
- Dedicated to I²C configuration: I2C_CFG

A set of additional registers (such as (DSFID, AFI, IC_REF) allows the user to identify and customize the product .

In I²C

Read accesses to the static configuration register is always allowed, except for passwords. For dedicated registers, write access is granted after prior successful presentation of the I²C password. The configuration register is located from address 0000h to 00FFh in the system area.

In RF

Dedicated commands Read Configuration and Write Configuration must be used to access the static configuration registers. Update is only possible when the access right was granted by presenting the RF configuration password (RF_PWD_0), and if the system configuration was not previously locked by the I²C host (LOCK_CFG=1), which acts as a security controller.

After any valid write access to the static configuration registers, the new configuration is immediately applied.

Some of the static registers have a dynamic image (notice _Dyn) preset with the static register value: GPO_CTRL_Dyn, EH_CTRL_Dyn, RF_MNGT_Dyn and MB_CTRL_Dyn.

When it exists, ST25DVxxKC uses the dynamic configuration register to manage its processes. A dynamic configuration register updated by the application recovers its default static value after a power on reset (POR).

Other dynamic registers are dedicated to process monitoring:

- I2C_SSO_Dyn is dedicated to data memory protection
- MB_LEN_Dyn, MB_CTRL_Dyn are dedicated to fast transfer mode
- IT_STS_Dyn is dedicated to interrupt

In I²C, read and write of the dynamic registers is done using the usual I²C read and write command at the dedicated address (E2 = 0 and E1 = 1 in device select).

In RF read or write, accesses to the dynamic registers are associated to the dedicated commands Read Dynamic Configuration, Write Dynamic Configuration, and Read Message Length.

5.1 Fast transfer mode (FTM)

5.1.1 Fast transfer mode registers

Static registers

Table 16. FTM access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @0Dh Write Configuration (cmd code A1h) @0Dh	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 000Dh	R always, W if I^2C security session is open

Table 17. FTM

Bit	Name	Function	Factory value
b0	MB_MODE	0: Enabling fast transfer mode is forbidden. 1: Enabling fast transfer mode is authorized.	0b
b3-b1	MB_WDG	Watchdog duration = $2^{(MB_WDG - 1)}$ x 30 ms ± 6 ms If MB_WDG = 0 watchdog is disabled	000b
b7-b4	RFU	-	0000b

Note: Refer to Table 13. System configuration memory map for the FTM register.

Dynamic registers

Table 18. MB_CTRL_Dyn access

RF		I^2C	
Command	Type	Address	Type
Read Dynamic Configuration (cmd code ADh) @0Dh Fast Read Dynamic Configuration (cmd code CDh) @0Dh Write Dynamic Configuration (cmd code AEh) @0Dh Fast Write Dynamic Configuration (cmd code CEh) @0Dh	b0: R always, W only if MB_MODE = 1 and V_{CC} is present b7-b1: RO	E2 = 0, E1 = 1, 2006h	b0: R always, W only if MB_MODE = 1 and V_{CC} is present b7-b1: RO

Table 19. MB_CTRL_Dyn

Bit	Name	Function	Factory value
b0	MB_EN (1)	0: Disable FTM, FTM mailbox is empty 1: Enable FTM	0b
b1	HOST_PUT_MSG	0: No I²C message in FTM mailbox 1: I²C has Put a message in FTM mailbox	0b
b2	RF_PUT_MSG	0: No RF message in FTM mailbox 1: RF has Put message in FTM mailbox	0b
b3	RFU	-	0b
b4	HOST_MISS_MSG	0: No message missed by I²C 1: I²C did not read RF message before watchdog time out	0b
b5	RF_MISS_MSG	0: No message missed by RF 1: RF did not read message before watchdog time out	0b
b6	HOST_CURRENT_MSG	0: No message or message not coming from I²C 1: Current Message in FTM mailbox comes from I²C	0b
b7	RF_CURRENT_MSG	0: No message or message not coming from RF 1: Current Message in FTM mailbox comes from RF	0b

1. *MB_EN bit is automatically reset to 0 if MB_MODE bit in FTM register is reset to 0.*

Note: Refer to Table 14. Dynamic registers memory map for the MB_CTRL_Dyn register.

Table 20. MB_LEN_Dyn access

RF		I²C	
Command	Type	Address	Type
Read Message Length (cmd code ABh) Fast Read Message Length (cmd code CBh)	RO	E2=0, E1=1, 2007h	RO

Table 21. MB_LEN_Dyn

Bit	Name	Function	Factory value
b7-b0	MB_LEN	Size in byte, minus 1 byte, of message contained in FTM mailbox (automatically set by ST25DVxxKC)	0h

Note: Refer to Table 14. Dynamic registers memory map for the MB_LEN_Dyn register.

5.1.2 Fast transfer mode usage

ST25DVxxKC acts as mailbox between RF (reader, smartphone) and an I²C host (microcontroller). Each interface can send a message containing up to 256 bytes of data to the other interface through that mailbox.

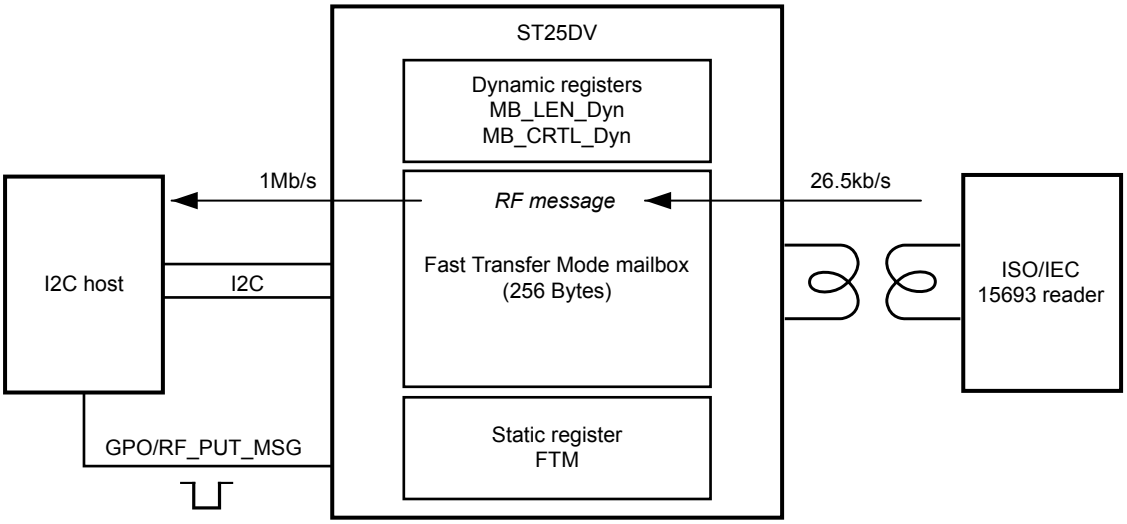
To send data from RF reader to I²C host, fast transfer mode must be enabled, the mailbox must be free, V_{CC} must be present, and the RF user must first write the message containing data in the mailbox.

I²C host is then informed (by interruption on GPO output or polling on MB_CTRL_Dyn register) that a message from RF is present in the mailbox.

Once the complete message has been read by I²C, mailbox is considered free again and is available for receiving a new message (data is not cleared).

The RF user is informed that the message has been read by the I²C host by polling on MB_CTRL_Dyn register.

Figure 11. RF to I²C fast transfer mode operation



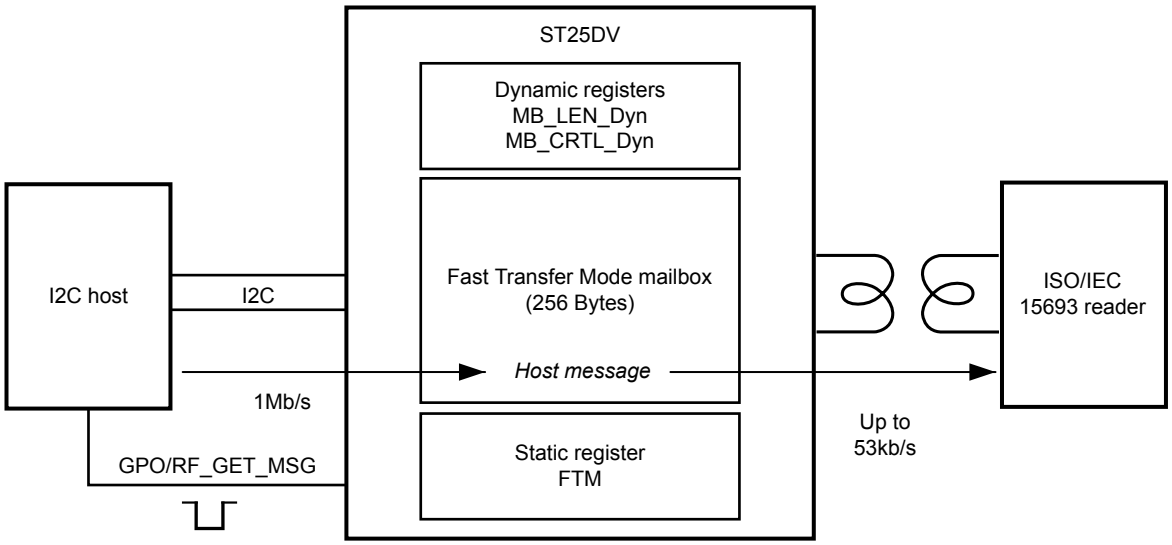
To send data from the I²C host to the RF reader, fast transfer mode must be enabled, the mailbox must be free, V_{CC} power must be present, and the I²C host must first write the message containing data in the mailbox.

The RF user must poll MB_CTRL_Dyn register to check for the presence of a message from I²C in the mailbox.

Once the complete message has been read by RF user, the mailbox is considered free again and is available for receiving a new message (data is not cleared).

The I²C host is informed that message has been read by RF user through a GPO interruption or by polling on the MB_CTRL_Dyn register.

Figure 12. I²C to RF fast transfer mode operation



V_{CC} supply source is mandatory to activate this feature.

No precedence rule is applied: the first request is served first.

Adding a message is possible only when fast transfer mode is enabled (MB_EN = 1) and mailbox is free (HOST_PUT_MSG and RF_PUT_MSG cleared), which is the case after POR or after complete reading of I^2C message by RF, or complete reading of RF message by I^2C.

A watchdog limits the message availability in time: when a time-out occurs, the mailbox is considered free, and the HOST_MISS_MSG or RF_MISS_MSG bits is set into MB_CTRL_Dyn register. The data contained in the mailbox is not cleared after a read or after the watchdog has been triggered: message data is still available for read and until fast transfer mode is disabled. HOST_CURRENT_MSG and RF_CURRENT_MSG bits indicate the source of the current data.

The message is stored in a buffer (256 bytes), and the write operation is done immediately.

Caution: *The data written in user memory (EEPROM), either from I^2C or from RF, transit via the 256-byte buffer. Fast transfer mode must be deactivated (MB_EN = 0) before starting any write operation in user memory, otherwise the command is NotACK for I^2C, or gets an answer 0Fh for RF, and programming is not done.*

I^2C access to mailbox

The access by I^2C can be done by dedicated address mapping to mailbox (2008h to 2107h) with device select E2 = 0, E1 = 1.

I^2C reading operation does not support rollover. Therefore data out is set to FFh when the internal address counter reaches the message end.

The RF_PUT_MSG is cleared after reaching the STOP that follows the reading of the last message byte, and the mailbox is considered free (but the message is not cleared and it is still present in the mailbox) until a new message is written or the mailbox is deactivated.

An I^2C reading operation never deletes the HOST_PUT_MSG, and the message remains available for RF.

An I^2C read can start at any address inside the mailbox (between address 2008h and 2107h).

An I^2C write operation must start from the first mailbox location, at address 2008h. After reaching the mailbox border at address 2107h, all bytes are NACK, and the command is not executed (no rollover).

At the end of a successful I^2C message write, the message length is automatically set into MB_LEN_Dyn register, and HOST_PUT_MSG bit is set into MB_CTRL_Dyn register, and the write access to the mailbox is not possible until the mailbox has been released again. MB_LEN_Dyn contains the size of the message in byte, minus 1.

RF access to mailbox

The RF control and access to mailbox is possible using dedicated custom commands:

- Read Dynamic Configuration and Fast Read Dynamic Configuration to check availability of mailbox.
- Write Dynamic Configuration and Fast Write Dynamic configuration to enable or disable fast transfer mode.
- Read Message Length and Fast Read Message Length to get the length of the contained message,
- Read Message and Fast Read Message to download the content of the mailbox,
- Write Message and Fast Write Message to put a new message in mailbox. (New length is automatically updated after completion of a successful Write Message or Fast Write Message command).

HOST_PUT_MSG is cleared following a valid reading of the last message byte, and mailbox is considered free (but message is not cleared and is still present in the mailbox) until a new message is written or mailbox is deactivated.

An RF read can start at any address of inside the message, but returns an error 0Fh if trying to read after the last byte of the message.

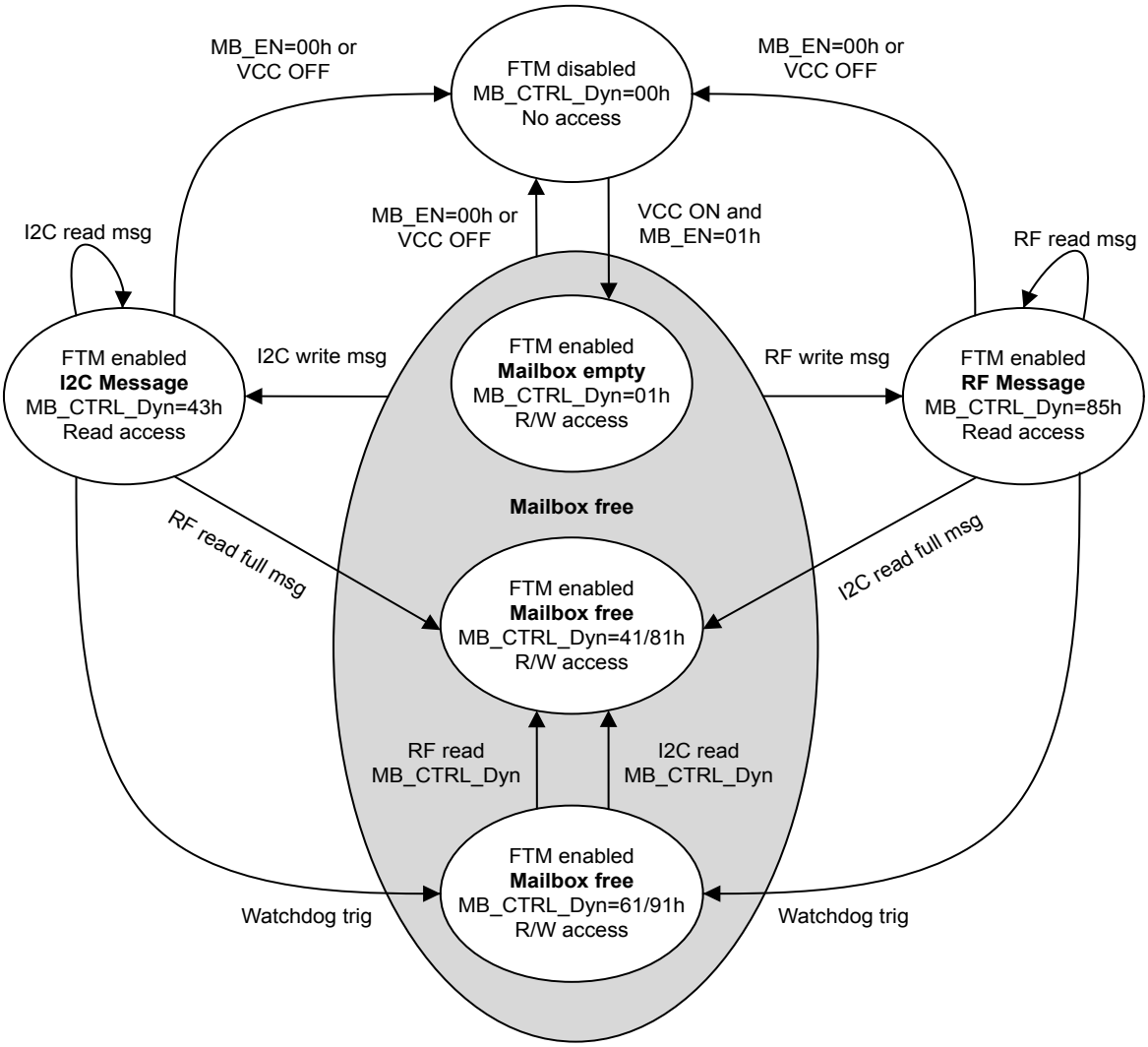
An RF reading operation never deletes the RF_PUT_MSG and the message remains available for I^2C.

At the end of a successful RF message write, the message length is automatically set in MB_LEN_Dyn register, RF_PUT_MSG bit is set in MB_CTRL_Dyn register, and write access to the mailbox is not possible until it has been freed again.

The presence of a DC supply is mandatory to get RF access to the mailbox. VCC_ON can be checked reading the dynamic register EH_CTRL_Dyn.

To get more details about sequences to prepare and initiate a tast transfer, to detect progress of a fast transfer or to control and execute a fast transfer, refer to AN4910 "*Data exchange data between wired (I^2C) and wireless (RF ISO 15693) using fast transfer mode supported by ST25DV-I2C Series*".

Figure 13. Fast transfer mode mailbox access management



Note: Assuming MB_MODE = 1b and assuming that no error occurred.

5.2 RF management feature

5.2.1 RF management registers

Table 22. RF_MNGT access

RF		I²C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @03h Write Configuration (cmd code A1h) @03h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0003h	R always, W if I²C security session is open

Table 23. RF_MNGT

Bit	Name	Function	Factory value
b0	RF_DISABLE	0: RF commands executed 1: RF commands not executed (error 0Fh returned)	0b
b1	RF_SLEEP	0: RF communication enabled 1: RF communication disabled (the device remains silent)	0b
b7-b2	RFU	-	000000b

Note: Refer to Table 13. System configuration memory map for the RF_MNGT register.

Table 24. RF_MNGT_Dyn access

RF		I²C	
Command	Type	Address	Type
No access		E2 = 0, E1 = 1, 2003h	R always, W always

Table 25. RF_MNGT_Dyn

Bit	Name	Function	Factory value
b0	RF_DISABLE	0: RF mode is defined by RF_OFF and RF_SLEEP bits 1: RF commands not executed (error 0Fh returned)	0b
b1	RF_SLEEP	0: RF mode is defined by RF_OFF and RF_DISABLE bits 1: RF communication disabled (the device remains silent)	0b
b2	RF_OFF	0: RF mode is defined by RF_SLEEP and RF_DISABLE bits 1: RF is reset, and communication disabled (RF security sessions and ISO15693 state are reset, and the device remains silent)	0b
b7-b3	RFU	-	0000000b

Note: Refer to Table 14. Dynamic registers memory map for the RF_MNGT_Dyn register.

The RF_OFF bit access is defined as followed:

- read only with user memory I²C target address, followed by memory address of RF_MNGT_Dyn register.
- write to 1 only with I²C "RFSwitchOff" command.
- write to 0 only with I²C "RFSwitchOn" command.
- cannot be accessed by any other I²C target address or by RF.

The RF_DISABLE and RF_SLEEP bits are accessible in Read and Write with the user memory I²C salve address, followed by memory address of RF_MNGT_Dyn register.

5.2.2 RF management

Communication capabilities with an RF reader can be controlled by configuring the device in one of the available modes:

- RF normal mode (default mode)
- RF disable mode
- RF sleep mode
- RF off mode

The RF_MNGT and RF_MNGT_Dyn registers are used to configure and control the RF mode.

At boot time, and each time the RF_MNGT register is updated, content of RF_MNGT_Dyn register is copied from the RF_MNGT register. The content of the register RF_MNGT_Dyn can be updated on the fly, to temporarily modify the behavior of the device without affecting the static value of RF_MNGT register, which is recovered at next POR.

RF normal mode:

In normal usage, and if I^2C interface is not busy (see Section 5.3: Interface arbitration) , the device processes the RF request and respond accordingly. In this mode, all bits of RF_MNGT_Dyn are set to 0.

RF disable mode:

In disable mode, RF commands are interpreted but not executed. In case of a valid command, the device responds after t_1 with the error 0Fh, and remains mute to the Inventory command.

ISO15693 state and RF security sessions status are unchanged.

In this mode, bit 0 of RF_MNGT_Dyn (RF_DISABLE) is set to 1 and all other bits are set to 0

RF sleep mode:

In sleep mode, all RF communication are disabled and RF interface doesn't interpret any RF commands.

ISO15693 state and RF security sessions status are unchanged.

In this mode, bit 1 of RF_MNGT_Dyn, RF_SLEEP, is set to 1 and bit 2, RF_OFF, is set to 0 (bit 0, RF_DISABLE is don't care).

RF off mode:

In off mode, all RF communication are disabled and RF interface doesn't interpret any RF commands.

ISO15693 state is reset and RF security sessions are closed.

In this mode, bit 2 of RF_MNGT_Dyn, RF_OFF, is set to 1 and other bits are don't care.

RF sleep and RF disable modes are controlled through writing in RF_SLEEP and RF_DISABLE bits in RF_MNGT register from RF or I^2C and RF_MNGT_Dyn register from I^2C.

RF off mode is controlled exclusively from I^2C. RFSwitchOff and RFSwitchOn commands, respectively, allow to switch off and on the RF. Entering RF off mode sets the RF_MNGT_Dyn bit 2, RF_OFF to 1 (see Section 5.3.

RF off mode has priority over RF sleep mode, which has priority over RF disable mode.

The effect of updating RF_MNGT or RF_MNGT_Dyn registers is immediate.

The Effect of RFSwtichOff command can be immediate or effective at the end of a write in progress in EEPROM, to avoid data corruption. A pulse can be generated on GPO pin to indicate to the I^2C host exactly when the device enters in RF off mode following a valid RFSwitchOff command.

RF off mode can be exited with the RFSwitchOn command, or by removing the V_{CC} power supply. Exiting RF off mode resets bit 2 (RF_OFF) of RF_MNGT_Dyn register. When exiting RF off mode, the state machine is set to Reset to Ready state, and all RF security sessions are closed.

Table 26. RF modes summary

RF mode	RF requests treatment	ISO15693 state	RF security sessions
Normal	Executed normally	Changed by relevant RF requests	Changed by relevant RF requests
Disable	Not executed Error 0Fh returned when possible	Unchanged	Unchanged

RF mode	RF requests treatment	ISO15693 state	RF security sessions
Sleep	Not processed, not answered	Unchanged	Unchanged
Off	Not processed, not answered	Reset (back to reset to ready state)	Reset (all sessions closed)

The following table summarizes the effect of RF_OFF, RF_SLEEP and RF_DISABLE bits, as well as the I^2C busy state on any RF request.

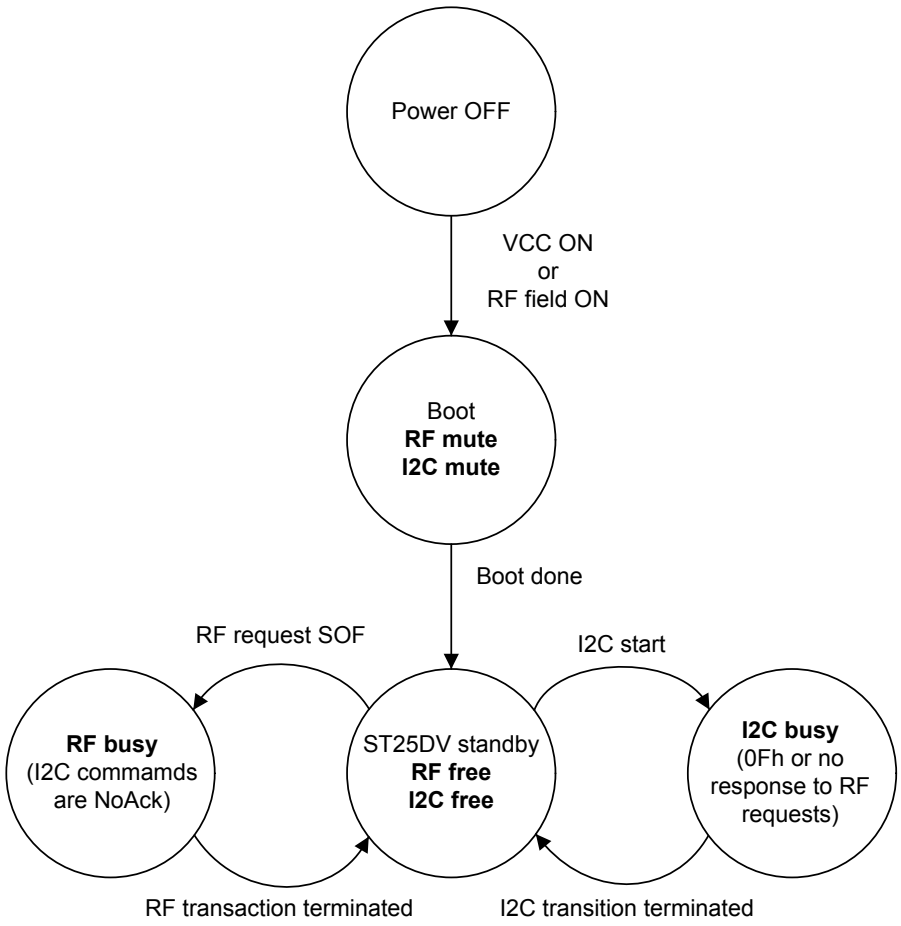
Table 27. RF modes configuration bits and effect on RF requests

RF_OFF	RF_SLEEP	RF_DISABLE	I^2C busy	Effect on RF request
0	0	0	0	Processed
0	0	0	1	Not processed and answered with error 0Fh when possible
0	0	1	x	Not processed and answered with error 0Fh when possible
0	1	x	x	Not processed, not answered
1	x	x	x	Not processed, not answered

5.3 Interface arbitration

The device automatically arbitrates the exclusive usage of RF and I²C interfaces, according to a *first-talks-first-served* scheme.

Figure 14. Arbitration between RF and I²C



RF transaction is terminated:

- at response EOF if answer
- at request EOF is no answer
- at RF field OFF

I²C transaction is terminated:

- at the end of EEPROM programming time after the stop condition of a successful write into the user memory or system configuration (see Section 6.4: I²C write operations for write time calculation)
- at stop condition for any other I²C transaction
- at V_{CC} power off
- at any I²C error (terminated before stop condition)
- at I²C timeout if it occurs

When RF is busy, I²C interface answers by NoAck to all I²C commands, except RFSwitchOff and RFSwitchOn.

When I²C is busy, RF commands receive no response (Inventory, Stay quiet, addressed commands) or error code 0Fh for any other command.

5.3.1 I²C priority

When RF is in sleep mode or in off mode, RF commands are not interpreted, and RF cannot by busy. I²C is then free to exclusively access the device.

Entering in RF sleep mode implies that the I²C host writes into the RF_MNGT_Dyn register, which is not possible immediately if RF is busy.

If the I²C host needs to get exclusive and immediate access to the device, a switch off (and on) of the RF interface is available.

A specific RFSwitchOff command allows the I²C controller to switch off RF immediately, or at the end of an RF write in progress in EEPROM, even if an RF command is ongoing.

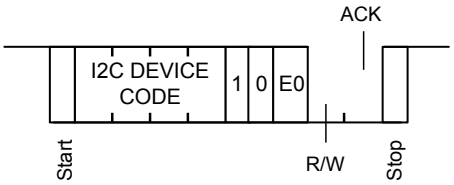
A specific RFSwitchOn command allows the I²C controller to switch on the RF immediately (RF returns to RF mode defined by the RF_MNGT_Dyn register).

Bit 5 of the I2C_CFG static register (I2C_RF_SWITCHOFF_EN) allows to authorize or forbid the RF switch off and switch on from I²C.

The RFSwitchOff command is defined as follows:

- START condition, followed by the I²C “RFSwitchOff” salve address (1 byte), followed by the acknowledge bit from the device, followed by STOP condition.
- See Section 6.3: Device addressing for I²C RFSwitchOff target address value explanation.
- RFSwitchOff slave address is not acknowledged only if I2C_CFG register bit 5 (I2C_RF_SWITCHOFF_EN) is set to 0, is always acknowledged otherwise (even if RF is busy).

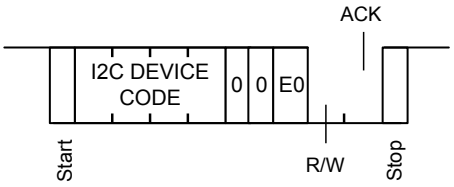
Figure 15. RFSwitchOff command



The RFSwitchOn command is defined as follows:

- START condition, followed by RFSwitchOn slave address (1 byte), followed by acknowledge bit from the device, followed by STOP condition.
- See Section 6.3: Device addressing for I²C RFSwitchOn target address value explanation.
- RFSwitchOn slave address is not acknowledged only if I2C_CFG register bit 5 (I2C_RF_SWITCHOFF_EN) is set to 0, is always acknowledged otherwise (even if RF is busy).

Figure 16. RFSwitchOn command



When the device receives the RFSwitchOff command outside of any RF command processing, it sets the RF in off mode (see Section 5.2.2: RF management). If GPO interruption RF_OFF is enabled, a pulse is emitted on the GPO pin after the stop condition of the RFSwitchOff command.

When the device receives the RFSwitchOff command concurrently to an RF command, there are two possible cases:

- If there is a write in progress in EEPROM, following an RF write command execution, the RF is set in off mode at the completion of the write in memory. The device does not answer to the RF request, but data are written into memory. If GPO interruption RF_OFF is enabled, a pulse is emitted on the GPO pin at the end of all write programming cycles.
- If there is no write in progress in EEPROM, the RF is set in RF off mode immediately. The device does not answer to the RF request. If GPO interruption RF_OFF is enabled, a pulse is emitted on the GPO pin after the stop condition of the RFSwitchOff command.

Once in RF off mode, the I²C host gets exclusive access to the device, whatever the incoming RF requests (ignored).

5.4 GPO

This signal is used to alert the I²C host of external RF events or device activity, and of specific I²C events. A host interruption can be requested for several reasons. RF user can also directly drive GPO pin level using a dedicated RF command.

5.4.1 Interrupt capabilities on RF events

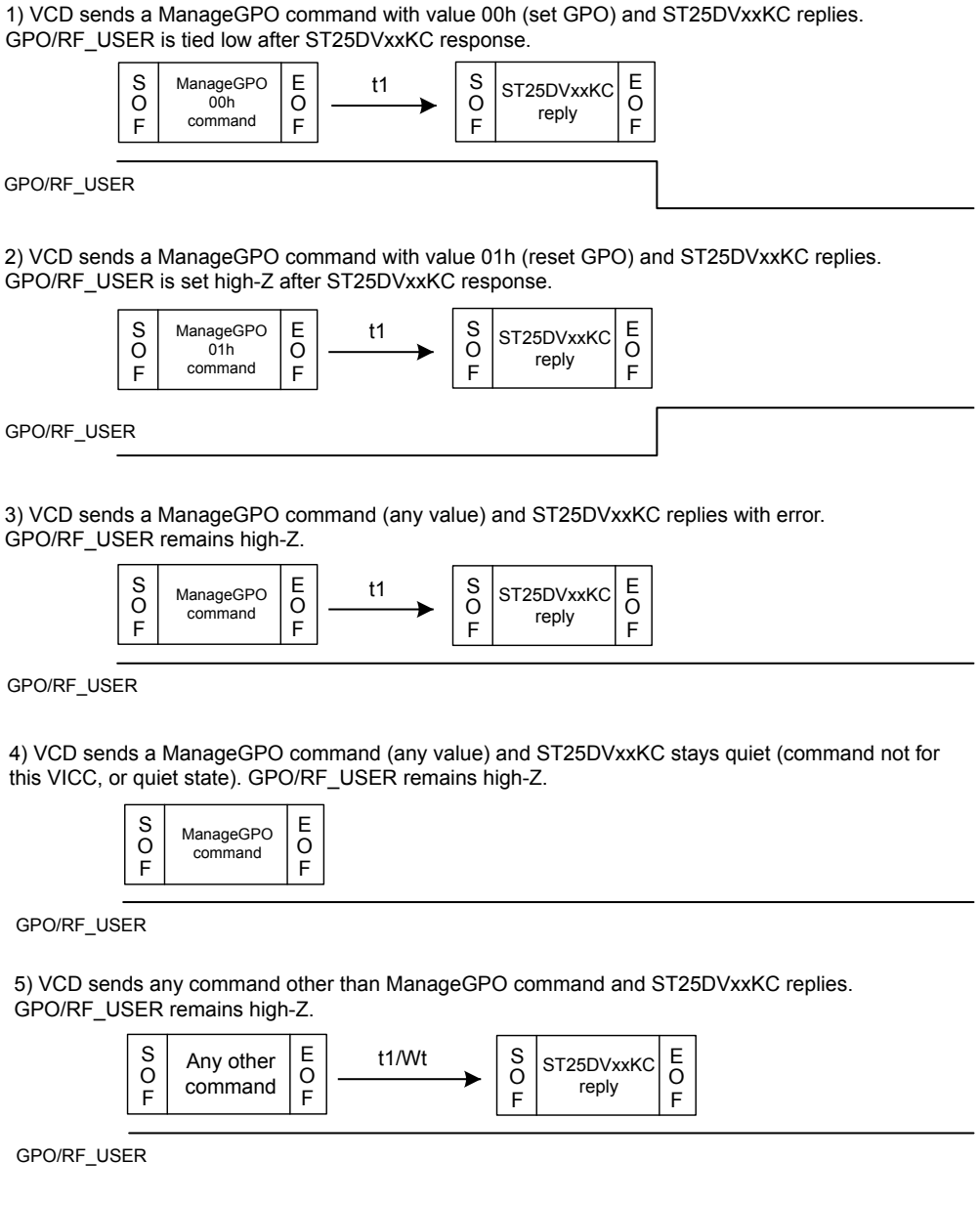
The device supports multi interruption mode and can report several events occurring through RF interface.

In this section, all drawings refer to the open drain version of GPO output (8-pin packages). The behavior of the CMOS version (12-pin package) is obtained by inverting the GPO curve polarity, and replacing the word “released” or “high-Z” by “pull to ground”.

RF_USER

- GPO output level is controlled by Manage GPO command (set or reset)
- When RF_USER is activated, GPO level is changed after EOF of the device response to a Manage GPO set or reset command (see Section 7.6.30: Manage GPO)
- RF_USER is prevalent over all other GPO events when set by Manage GPO command (other interrupts are still visible in IT_STS_Dyn status register, but do not change GPO output level)

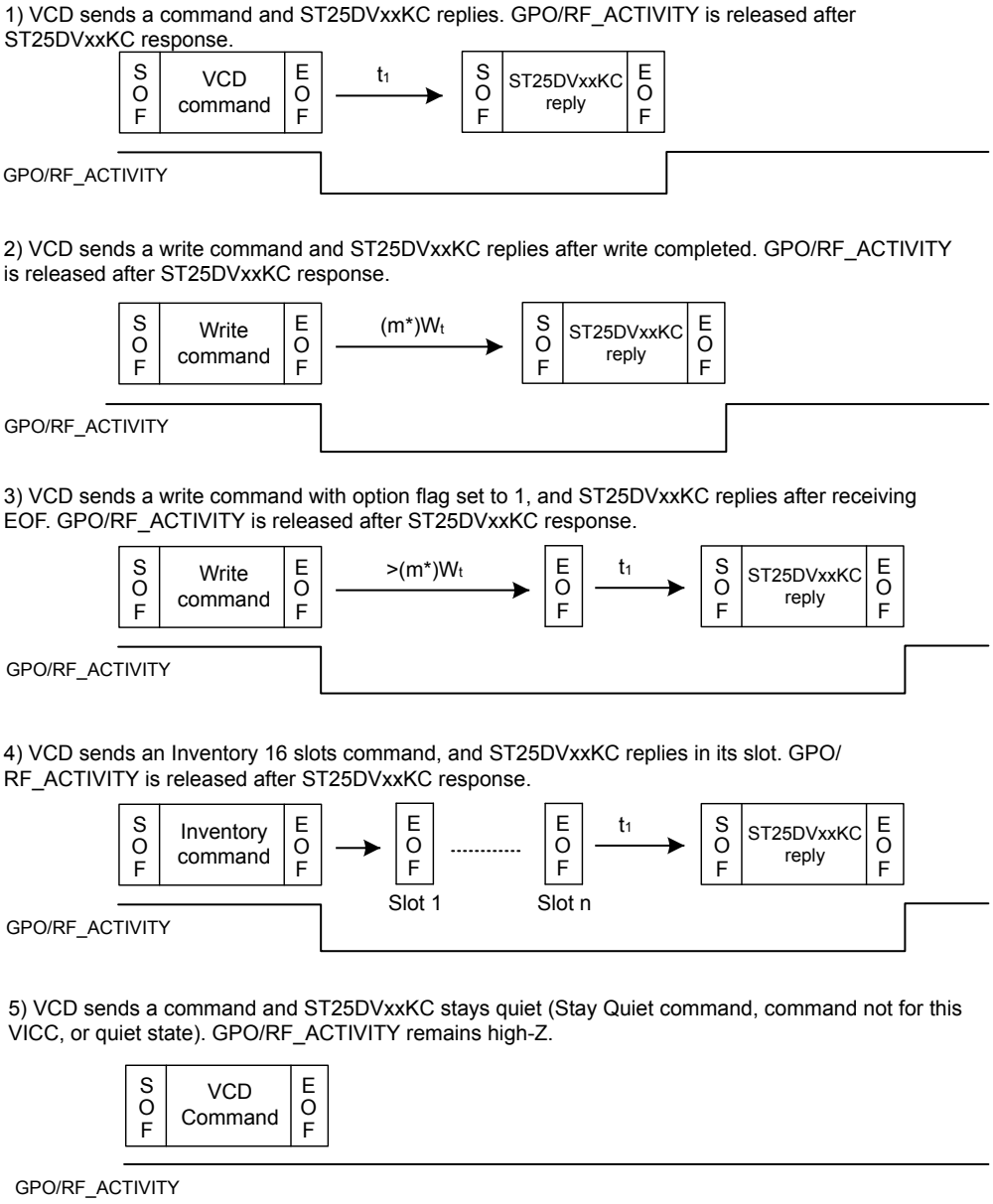
Figure 17. RF_USER sequence



RF_ACTIVITY

- GPO output level reflects the RF activity
- When RF_ACTIVITY is activated, a GPO output level changes from RF command EOF to the device response EOF

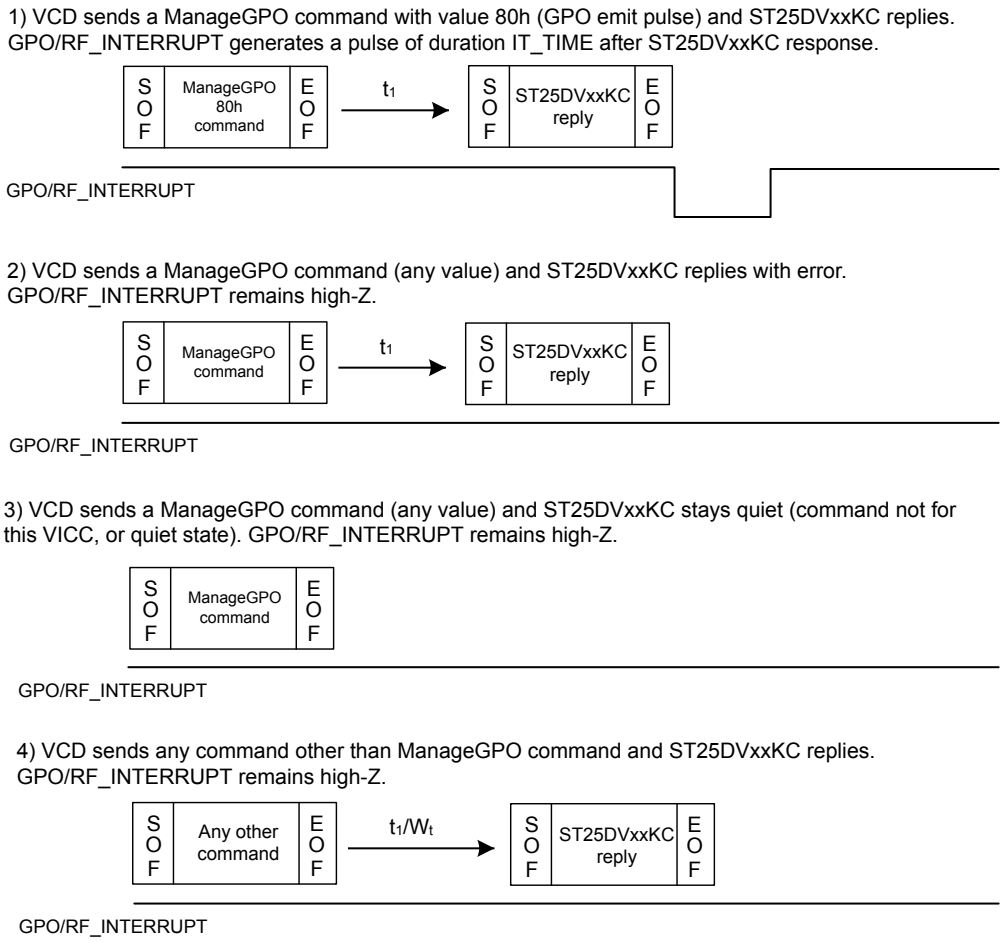
Figure 18. RF_ACTIVITY sequence



RF_INTERRUPT

- A pulse is emitted on GPO by Manage GPO command (interrupt).
- When RF_INTERRUPT is activated, a pulse of duration IT_TIME is emitted after EOF of ST25DVxxKC response to a Manage GPO interrupt command (see Section 7.6.30: Manage GPO).

Figure 19. RF_INTERRUPT sequence



FIELD_CHANGE

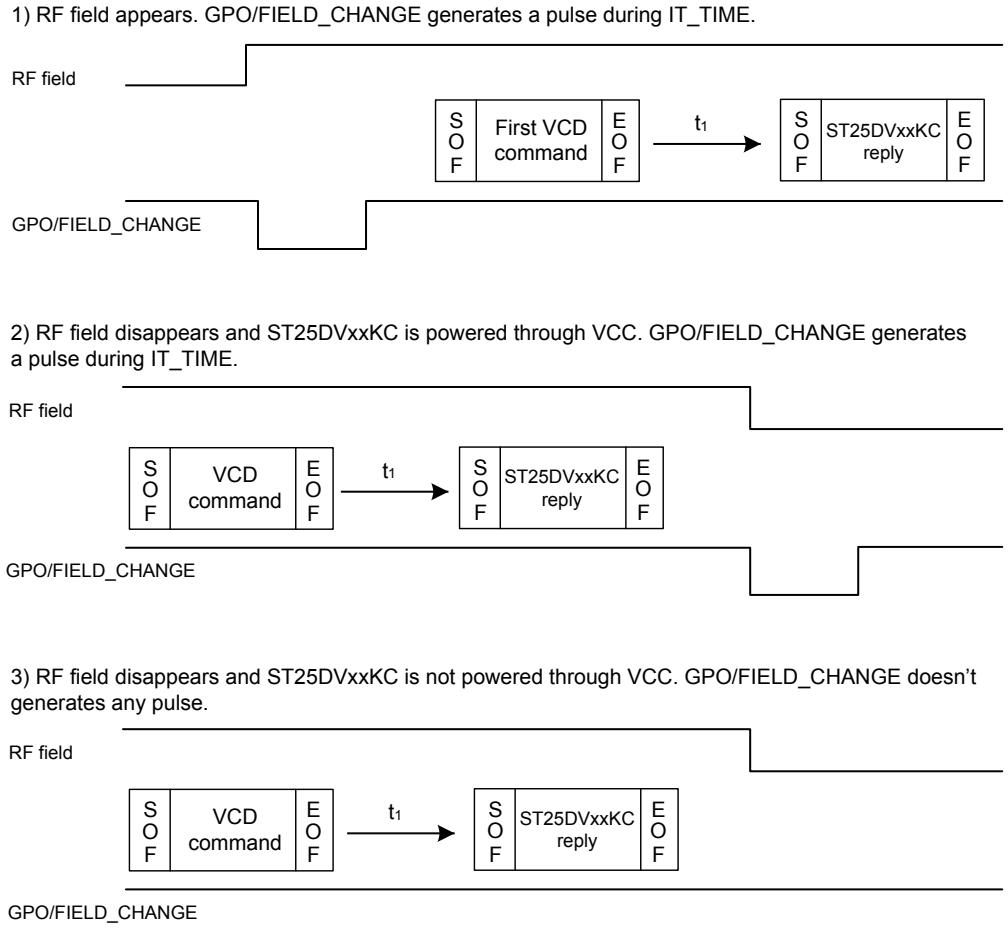
- A pulse is emitted on GPO to signal a change in RF field state
- When FIELD_CHANGE is activated, and when RF field appear or disappear, GPO emits a pulse of duration IT_TIME
- If RF field disappears, the pulse is emitted only if V_{CC} power supply is present
- If RF is configured in RF_SLEEP mode or is in RF_OFF state, field changes are not reported on GPO, even if FIELD_CHANGE event is activated, as shown in Table 28

Table 28. FIELD_CHANGE when RF is disabled, or in sleep, or off mode

RF_DISABLE	RF_SLEEP	RF_OFF	GPO behavior when FIELD_CHANGE is enabled
0	0	0	A pulse is emitted on GPO if RF field appears or disappears (1)
1	0	0	IT_STS_Dyn register is updated.
X	1	0	GPO remains high-Z (open drain version) or tied to ground (CMOS version).
X	X	1	IT_STS_Dyn register is not updated.

1. *assuming that GPO output is enabled (GPO_EN = 1).*

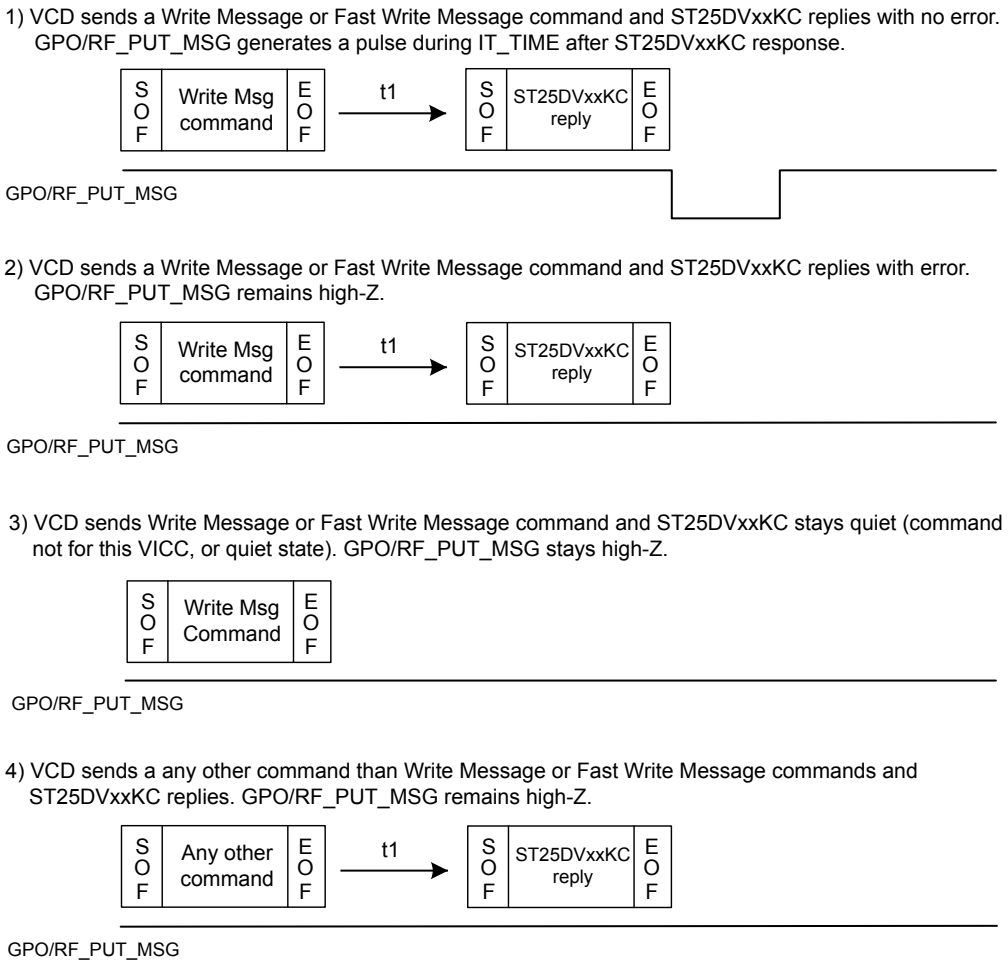
Figure 20. FIELD_CHANGE sequence



RF_PUT_MSG

- A pulse is emitted on GPO when a message is successfully written by RF in fast transfer mode mailbox
- When RF_PUT_MSG is activated, a pulse of duration IT_TIME is emitted on GPO at completion of valid Write Message or Fast Write Message commands (after EOF of device response)

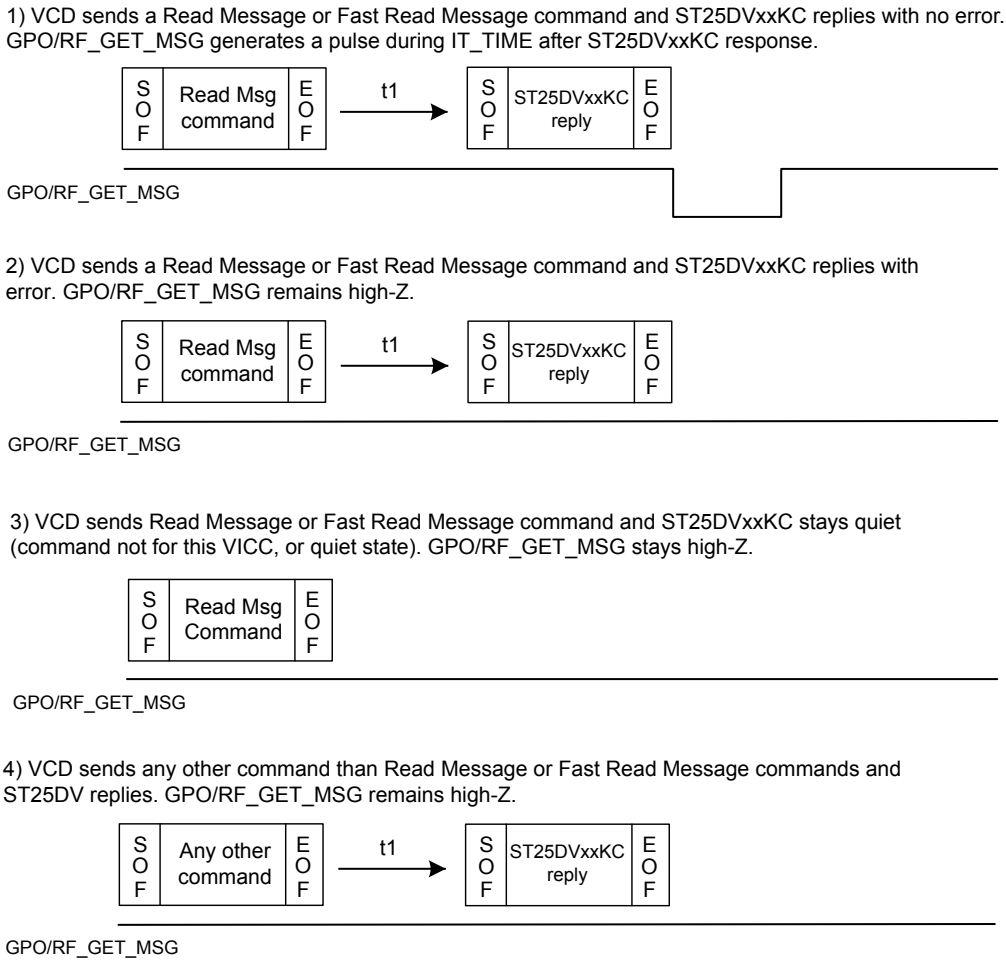
Figure 21. RF_PUT_MSG sequence



RF_GET_MSG

- A pulse is emitted on GPO when RF has successfully read a message, up to its last byte, in fast transfer mode mailbox
- When RF_GET_MSG is activated, a pulse of duration IT_TIME is emitted on GPO at completion of valid Read Message or Fast Read Message commands (after EOF of ST25DVxxKC response), and end of message has been reached

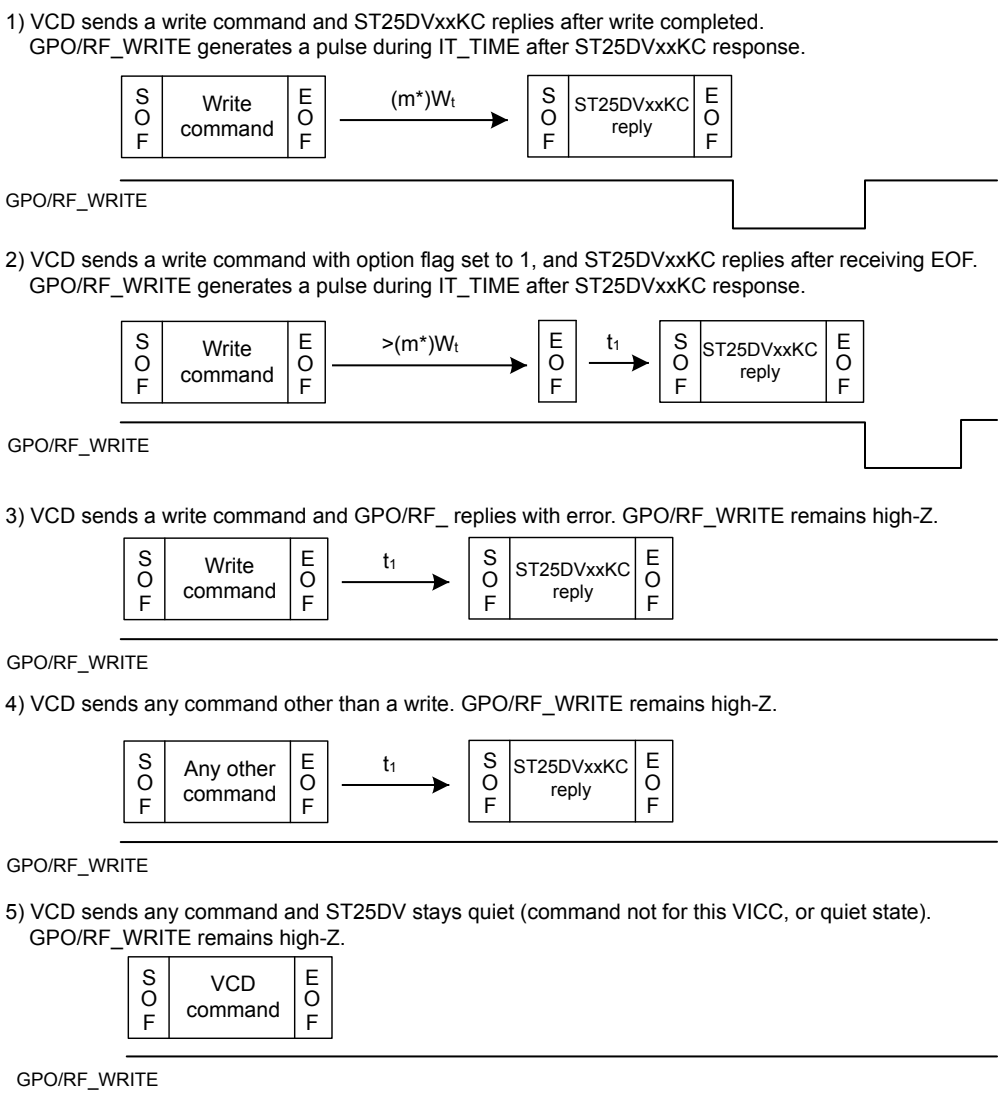
Figure 22. RF_GET_MSG sequence



RF_WRITE

- When RF_WRITE is activated, a pulse of duration IT_TIME is emitted at completion of a valid RF write operation in EEPROM (after EOF of ST25DVxxKC response)
- Following commands trigger the RF_WRITE interrupt after a valid write operation in EEPROM:
  - Write Single Block
  - Extended Write Single Block
  - Write Multiple Block
  - Extended Write Multiple Block
  - Lock Block
  - Extended Lock Block
  - Write AFI
  - Lock AFI
  - Write DSFID
  - Lock DSFID
  - Write Configuration
  - Write Password
- Writing in dynamic registers or fast transfer mode mailbox does not trigger RF_WRITE interrupt (no write operation in EEPROM)

Figure 23. RF_WRITE sequence



5.4.2 Interrupt capabilities on I²C events

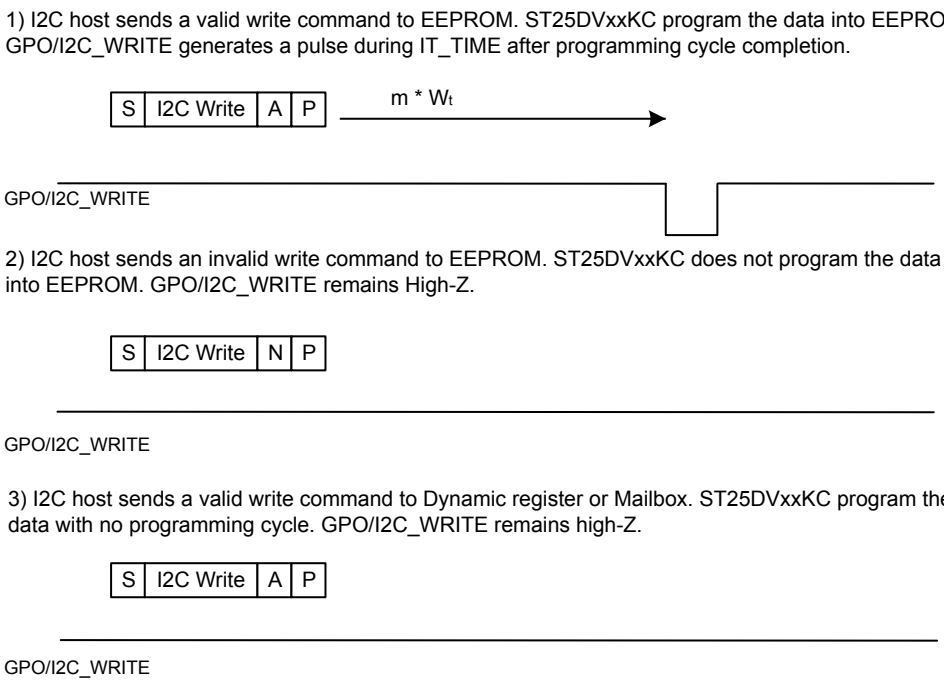
On top of RF events, the device provides two additional I²C events that can trigger an interrupt on the GPO pin.

In this section, all drawings refer to the open drain version of GPO output (8-pin packages). The behavior of the CMOS version (12-pin package) is obtained by inverting the GPO curve polarity, and replacing the word “released” or “high-Z” by “pull to ground”. The supported events are described in the next subsections.

I2C_WRITE

- When I2C_WRITE is activated, a pulse of duration IT_TIME is emitted at completion of a valid I²C write operation in EEPROM (after the I²C STOP condition).
- Writing in dynamic registers or fast transfer mode mailbox does not trigger the I2C_WRITE interrupt (no write operation in EEPROM).
- The purpose of this GPO interrupt is to inform the I²C host when the I²C write programming cycle in EEPROM is completed, meaning the I²C bus and RF interface are free for new operation.

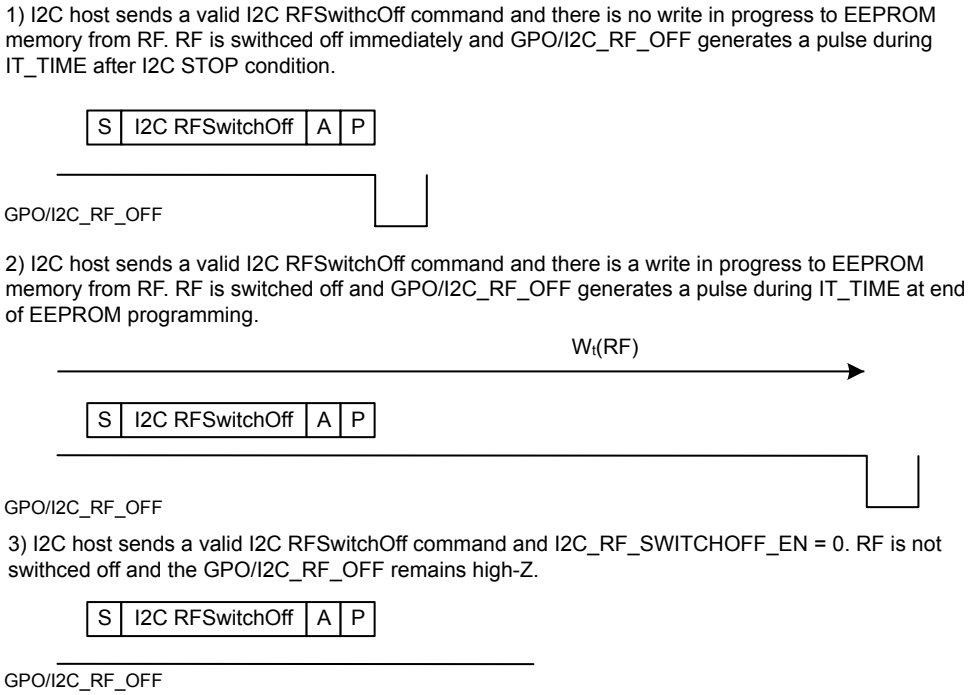
Figure 24. GPO/I2C_WRITE sequence



I2C_RF_OFF

- When I2C_RF_OFF is activated, a pulse of duration IT_TIME is emitted:
  - After the I²C STOP condition of a successful I²C “RFSwitchOff” command if no RF write to EEPROM is ongoing.
  - After the end of all blocks programming if the STOP condition of a successful I²C “RFSwitchOff” command happens during an RF write to EEPROM.
- The purpose of this GPO interrupt is to inform the I²C controller when the I²C RFSwitchOff command has switched off the RF (RF is in off mode), as the timing action of the I²C RFSwitchOff can vary if an EEPROM write from the RF is ongoing.

Figure 25. GPO/I2C_RF_OFF sequence



5.4.3 GPO and power supply

When at the same time RF field is present and V_{CC} is ON, GPO acts as configured in GPO1, GPO2, and GPO_CTRL_Dyn registers. Both RF and I²C events are reflected to the GPO pin.

When V_{CC} is ON and no RF field is present, GPO acts as configured in GPO2 and GPO_CTRL_Dyn registers. Only I²C events are reflected on the GPO pin. IT_STS_Dyn register is maintained unchanged until next I²C read of V_{CC} power off.

When RF field is present and V_{CC} is OFF, GPO acts as configured in GPO1 (and GPO2 for IT_TIME configuration only) and GPO_CTRL_Dyn registers. Only RF events are reflected on the GPO pin (assuming pull-up resistor is supplied with correct voltage for open drain version, or V_{DCG} voltage is supplied for CMOS version). Exception is FIELD_CHANGE when RF field is falling, which cannot be reported on GPO output if V_{CC} is off (no power supply).

Table 29. GPO interrupt capabilities according to RF field and V_{CC}

RF field	V_{CC}	LPD	GPIO pin
OFF	OFF	Don't care	Remains high-Z (open drain version) or is tied to ground (CMOS version)
ON	OFF	Don't care	State is function of RF events(1)(2)
OFF	ON	High	Remains high-Z (open drain version) or is tied to ground (CMOS version)
ON	ON	High	State is function of RF events(1)(2)
OFF	ON	Low/unconnected	State is function of I^2C events
ON	ON	Low/unconnected	State is function of both RF and I^2C events(1)

1. *If pull-up resistor is powered (open drain) or V_{DCG} is powered (CMOS).*
2. *Except FIELD_CHANGE in case of RF field falling.*

5.4.4 GPO registers

Four registers are dedicated to this feature, namely two static registers in system configuration, and two dynamic registers.

Table 30. GPO1 access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @00h Write Configuration (cmd code A1h) @00h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0000h	R always, W if I^2C security session is open

Table 31. GPO1

Bit	Name	Function	Factory value
b0	GPO_EN	0: GPO output is disabled. GPO is high-Z (open drain version) or is tied to ground (CMOS version). 1: GPO output is enabled. GPO outputs enabled interrupts.	1b
b1	RF_USER_EN	0: disabled 1: GPO output level is controlled by Manage GPO command (set/reset).	0b
b2	RF_ACTIVITY_EN	0: disabled 1: GPO output level changes from RF command EOF to response EOF.	0b
b3	RF_INTERRUPT_EN	0: disabled 1: GPO output level is controlled by Manage GPO command (pulse).	0b
b4	FIELD_CHANGE_EN	0: disabled 1: A pulse is emitted on GPO, when RF field appears or disappears.	1b
b5	RF_PUT_MSG_EN	0: disabled 1: A pulse is emitted on GPO at completion of valid RF Write Message command.	0b
b6	RF_GET_MSG_EN	0: disabled 1: A pulse is emitted on GPO at completion of valid RF Read Message command if end of message has been reached.	0b
b7	RF_WRITE_EN	0: disabled 1: A pulse is emitted on GPO at completion of valid RF write operation in EEPROM.	0b

Note: Refer to Table 13. System configuration memory map for the GPO1 register:

- *Enables the interruption source, and enable GPO output.*
- *Several interruption sources can be enabled simultaneously.*
- *The updated value is valid for the next command (except for the RF_WRITE interrupt, which is valid right after EOF of the Write Configuration command if enabled through RF).*
- *The GPO_EN bit (b0) is used to disable GPO output. The interruptions are still reported in IT_STS_Dyn register.*

Table 32. GPO2 access

RF		I²C	
Command	**Type**	**Address**	**Type**
Read Configuration (cmd code A0h) @01h Write Configuration (cmd code A1h) @01h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0001h	R always, W if I²C security session is open

Table 33. GPO2

Bit	Name	Function	Factory value
b0	I2C_WRITE_EN	0: disabled 1: A pulse is emitted on GPO at completion of valid I²C write operation in EEPROM	0b
b1	I2C_RF_OFF_EN	0: disabled 1: A pulse is emitted on GPO when I²C host has successfully switched the RF off.	0b
b4-b2	IT_TIME	Pulse duration = 301 µs - IT_TIME x 37.65 µs ± 2 µs	011b
b7-b5	RFU	-	000b

Note: Refer to Table 13. System configuration memory map for the GPO2 register.

- Defines interrupt pulse duration on GPO pin for the flowing events: RF_INTERRUPT, FIELD_CHANGE, RF_PUT_MSG, RF_GET_MSG, RF_WRITE , I2C_RF_OFF_EN and I2C_WRITE_EN.
- See Eq. (1) for interrupt duration calculation.

Table 34. GPO_CTRL_Dyn access

RF		I²C	
Command	**Type**	**Address**	**Type**
Read Dynamic Configuration (command code ADh) @00h Fast Read Dynamic Configuration (command code CDh) @00h	RO	E2 = 0, E1 = 1, 2000h	b7-b1: RO b0 : R always, W always

Table 35. GPO_CTRL_Dyn

Bit	Name	Function	Factory value
b7-b1	RFU	-	0000000b
b0	GPO_EN	0: GPO output is disabled. GPO is high-Z (open drain version) or is tied to ground (CMOS version). 1: GPO output is enabled. GPO outputs enabled interrupts.	1b

Note: Refer to Table 14. Dynamic registers memory map for the GPO_CTRL_Dyn register.

- Allows I^2C host to dynamically enable or disable GPO output by writing in GPO_EN bit (b0).
- GPO_EN bit of GPO_CTRL_Dyn register is prevalent over GPO_EN bit of GPO register.
- At power up, and each time GPO1 register is updated, GPO_EN bit content is copied from GPO register.
- GPO_CTRL_Dyn is a volatile register. Value is maintained only if at least one of the two power sources is present (RF field or V_{CC}).
- GPO_CTRL_Dyn bit 0 (GPO_EN) can be written even if I^2C security session is closed (I^2C password not presented) but is read only for RF user.
- Modifying GPO_CTRL_Dyn bit 0 (GPO_EN), does not affect the value of GPO register bit 0 GPO_EN

Table 36. IT_STS_Dyn access

RF		I^2C	
Command	Type	Address	Type
No access		E2 = 0, E1 = 1, 2005h	RO

Table 37. IT_STS_Dyn

Bit	Name	Function	Factory value
b0	RF_USER	0: Manage GPO reset GPO 1: Manage GPO set GPO	0b
b1	RF_ACTIVITY	0: No RF access 1: RF access	0b
b2	RF_INTERRUPT	0: No Manage GPO interrupt request 1: Manage GPO interrupt request	0b
b3	FIELD_FALLING	0: No RF field falling 1: RF Field falling	0b
b4	FIELD_RISING	0: No RF field rising 1: RF field rising	0b
b5	RF_PUT_MSG	0: No message put by RF in FTM mailbox 1: Message put by RF in FTM mailbox	0b
b6	RF_GET_MSG	0: No message read by RF from FTM mailbox 1: Message read by RF from FTM mailbox, and 'end of message' reached	0b
b7	RF_WRITE	0: No write in EEPROM 1: Write in EEPROM	0b

Note: Refer to Table 14. Dynamic registers memory map for the IT_STS_Dyn register.

- Cumulates all events which generate interruptions. It should be checked by I^2C host to know which event triggered an interrupt on GPO pin.
- When enabled, RF events are reported in IT_STS_Dyn register even if GPO output is disabled though the GPO_EN bit.
- Once read, the IT_STS_Dyn register is cleared (set to 00h).
- At power up, IT_STS_Dyn content is cleared (set to 00h).
- IT_STS_Dyn is a volatile register. Value is maintained only if at least one of the two power sources is present (RF field or V_{CC}).

5.4.5 Configuring GPO

GPO and interruption pulse duration can be configured by RF user or by I²C host. One or more interrupts can be enabled at same time.

RF user can use Read Configuration and Write Configuration commands to set accordingly the GPO1 and GPO2 registers, after presenting a valid RF configuration password to open RF configuration security session.

I²C host can write GPO1 and GPO2 registers, after presenting a valid I²C password to open I²C security session.

Enabling or disabling GPO output:

- RF user and I²C host can disable or enable GPO output at power up time by writing in GPO_EN bit 0 of GPO1 register (if write access is granted).
- I²C host can temporarily enable or disable GPO output at any time by toggling GPO_EN bit 0 of GPO_CTRL_Dyn register. No password is required to write into GPO_CTRL_Dyn register.
- Disabling GPO output by writing in GPO_EN bit (either in GPO1 or in GPO_CTRL_Dyn registers) does not disable interruption report in IT_STS_Dyn status register.

Table 38. Enabling or disabling GPO interruptions

GPO1 bit 0: GPO_EN	GPO_CTRL_Dyn bit 0: GPO_EN	GPO output
0	0	GPO remains high-Z (open drain version) or is tied to ground (CMOS version).
1	0	GPO remains high-Z (open drain version) or is tied to ground (CMOS version).
0	1	Activated RF and I²C events are reported on GPO output.(1)
1	1	Activated RF and I²C events are reported on GPO output. (1)

1. *If pull-up resistor is powered (open drain version) or V_{DCG} is powered (CMOS version).*

Interruption pulse duration configuration:

- Interrupt pulse duration is configured by writing pulse duration value in bits 4 to 2 (IT_TIME) of GPO2 register
- Pulse duration is calculated with the following equation

IT pulse duration equation:

$$ITpulse\ duration = 301\mu s - IT_TIME \times 37.65\mu s \pm 2\mu s \quad (1)$$

5.5 Energy harvesting (EH)

5.5.1 Energy harvesting registers

Table 39. EH_MODE access

RF		I²C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @02h Write Configuration (cmd code A1h) @02h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0002h	R always, W if I²C security session is open

Table 40. EH_MODE

Bit	Name	Function	Factory value
b0	EH_MODE	0: EH forced after boot 1: EH on demand only	1b
b7-b1	RFU	-	0000000b

Note: Refer to Table 13. System configuration memory map for the EH_MODE register.

Table 41. EH_CTRL_Dyn access

RF		I²C	
Command	Type	Address	Type
Read Dynamic Configuration (cmd code ADh) @02h Fast Read Dynamic Configuration (cmd code CDh) @02h Write Dynamic Configuration (cmd code AEh) @02h Fast Write Dynamic Configuration (cmd code CEh) @02h	b0: R always, W always b1 - b7: RO	E2 = 0, E1 = 1, 2002h	b0: R always, W always b1-b7 : RO

Table 42. EH_CTRL_Dyn

Bit	Name	Function	Factory value
b0	EH_EN	0: Disable EH feature 1: Enable EH feature	0b
b1	EH_ON	0: EH feature is disabled 1: EH feature is enabled	0b
b2	FIELD_ON	0: RF field is not detected 1: RF field is present and ST25DVxxKC may communicate in RF	Depends upon of power source
b3	VCC_ON	0: No DC supply detected on V_{CC} pin or Low Power Down mode is forced (LPD is high) 1: V_{CC} supply is present and Low Power Down mode is not forced (LPD is low)	Depends upon power source
b7-b4	RFU	-	0b

Note: Refer to Table 14. Dynamic registers memory map for the EH_CTRL_Dyn register.

5.5.2 Energy harvesting description

The usage of this element can be defined in configuration register EH_MODE. When the Energy harvesting mode is disabled or the RF field strength is not sufficient, the energy harvesting analog voltage output V_EH is in high-Z state.

EH_MODE static register is used to define the default strategy after boot.

At boot EH_EN (in EH_CTRL_Dyn register) is set depending EH_MODE value as shown in the following table.

Table 43. Energy harvesting at power-up

EH_MODE	EH_EN (at boot)	Energy harvesting at power-up
0	1	EH enabled after boot (when possible)
1	0	EH disabled initially, EH delivered on demand (when possible)

Writing 0 in EH_MODE at any time after boot sets EH_EN bit to 1, and activates energy harvesting. Writing 1 in EH_MODE at any time after boot does not modify EH_EN bit (until next reboot), and does not modify the current state.

EH_CTRL_Dyn allows to activate or deactivate on the fly the Energy harvesting (EH_EN) and bring information on actual state of EH and state of power supplies :

- EH_ON set reflects the EH_EN bit value
- FIELD_ON is set in presence of an RF field
- VCC_ON is set when Host power supply is on, and low power-down mode is not forced.

During boot, EH is not delivered to avoid alteration in device configuration.

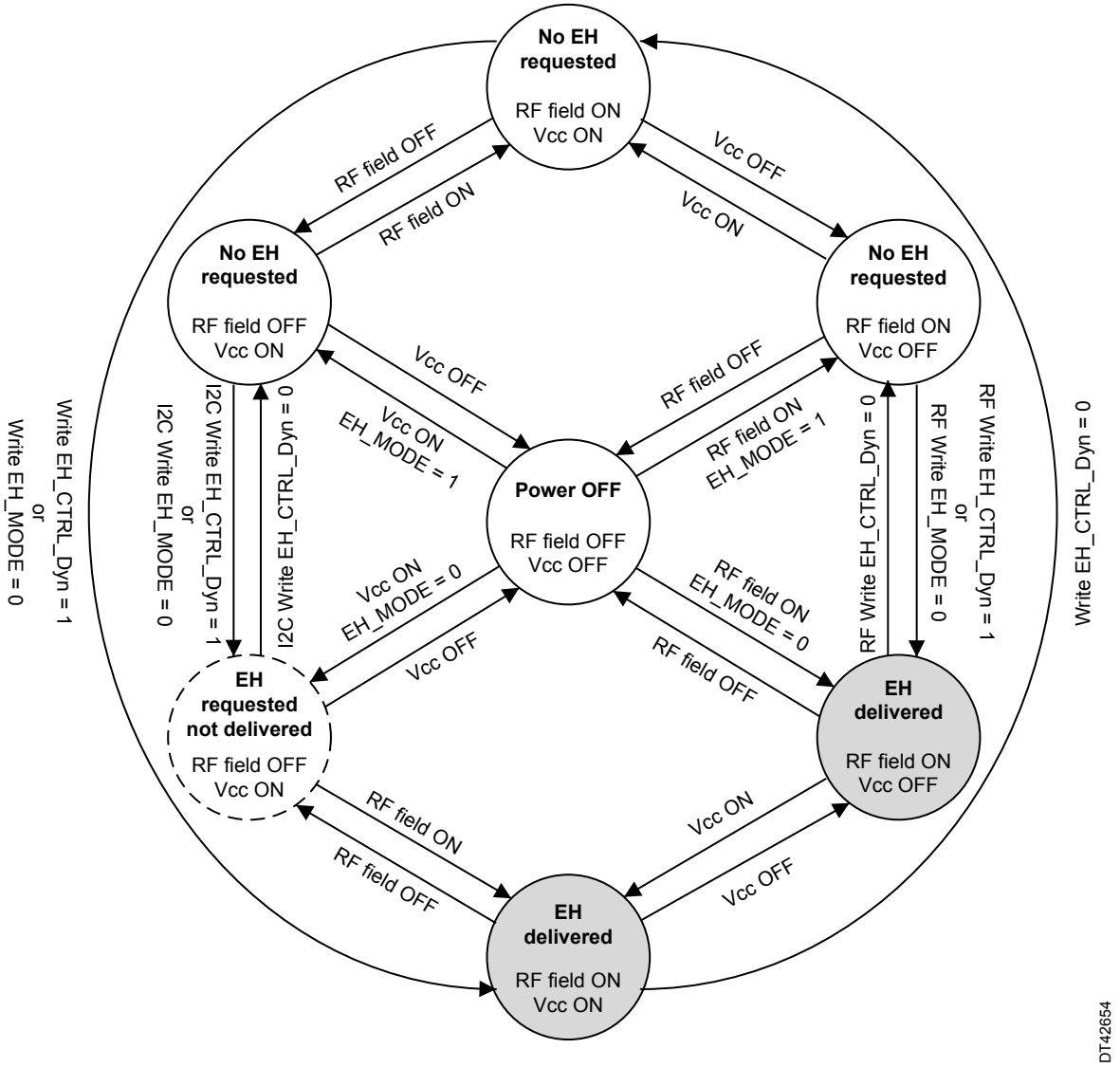
Caution: *Communication is not guaranteed during EH delivery. Refer to AN4913 "Energy harvesting delivery impact on ST25DV-I2C Series behaviour during RF communication".*

Energy harvesting can be set even if the device is in RF disabled, RF Sleep mode, or in Low power modes. In all these cases, the device delivers power on V_EH pin if RF field is present. Energy harvesting voltage output is not regulated.

5.5.3 EH delivery state diagram

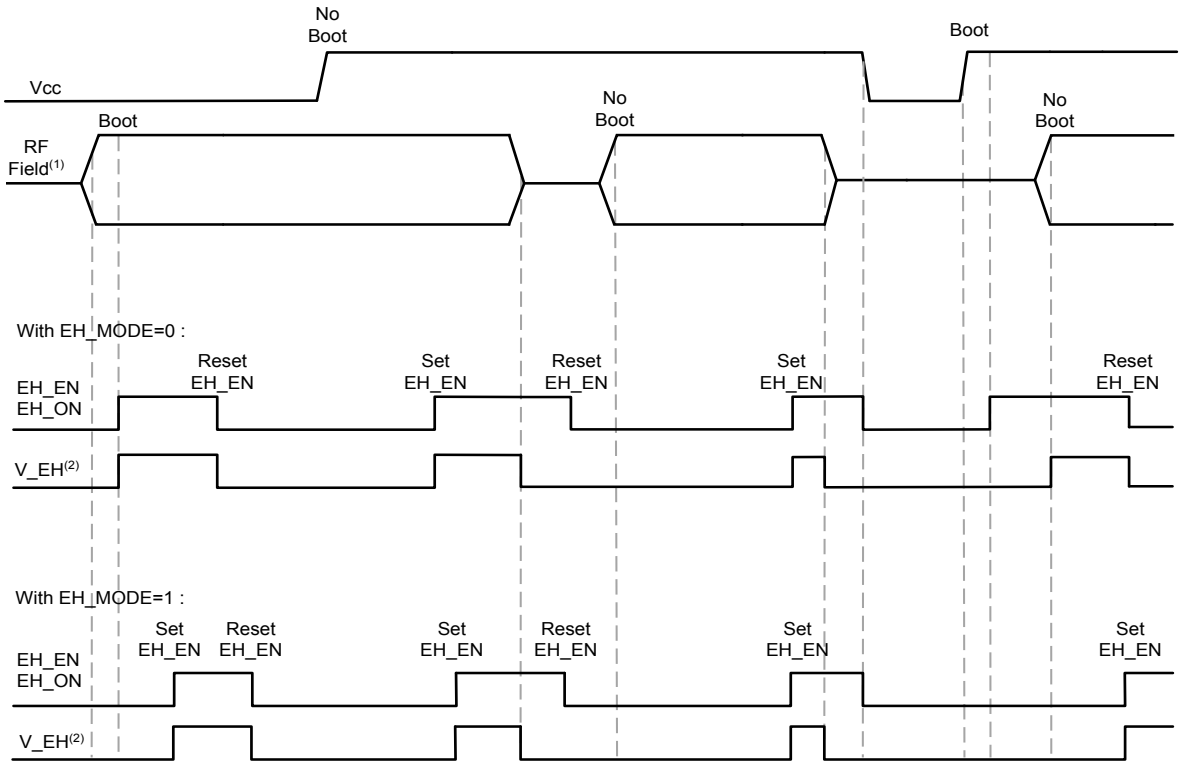
Power is delivered on the V_EH pin only if harvested energy is sufficient to supply the device and there is additional power. In Figure 26 the states where power is delivered are indicated in gray.

Figure 26. **EH delivery state diagram**



5.5.4 EH delivery sequence

Figure 27. **ST25DVxxKC Energy Harvesting Delivery Sequence**



1. *We suppose that the captured RF power is sufficient to trig EH delivery.*
2. *V_EH = 1 means some µW are available on V_EH pin.*
   V_EH = 0 means V_EH pin is in high-Z.

5.6 Data protection

ST25DVxxKC provides a special data protection mechanism based on passwords that unlock security sessions. User memory can be protected for read and/or write access and system configuration can be protected from write access, both from RF and I^2C assess.

5.6.1 Data protection registers

Table 44. **RFA1SS access**

RF		I^2C	
Command	**Type**	**Address**	**Type**
Read Configuration (cmd code A0h) @04h Write Configuration (cmd code A1h) @04h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0004h	R always, W if I^2C security session is open

Table 45. RFA1SS

Bit	Name	Function	Factory value
b1-b0	PWD_CTRL_A1	00: Area 1 RF user security session can't be open by password 01: Area 1 RF user security session is open by RF_PWD_1 10: Area 1 RF user security session is open by RF_PWD_2 11: Area 1 RF user security session is open by RF_PWD_3	00b
b3-b2	RW_PROTECTION_A1	00: Area 1 RF access: Read always allowed / Write always allowed 01: Area 1 RF access: Read always allowed, Write allowed if RF user security session is open 10: Area 1 RF access: Read always allowed, Write allowed if RF user security session is open 11: Area 1 RF access: Read always allowed, Write always forbidden	00b
b7-b4	RFU	-	0000b

Note: Refer to Table 13. System configuration memory map for the RFA1SS register.

Table 46. RFA2SS access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @06h Write Configuration (cmd code A1h) @06h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0006h	R always, W if I^2C security session is open

Table 47. RFA2SS

Bit	Name	Function	Factory value
b1-b0	PWD_CTRL_A2	00: Area 2 RF user security session cannot be open by password 01: Area 2 RF user security session is open by RF_PWD_1 10: Area 2 RF user security session is open by RF_PWD_2 11: Area 2 RF user security session is open by RF_PWD_3	00b
b3-b2	RW_PROTECTION_A2	00: Area 2 RF access: Read always allowed, Write always allowed 01: Area 2 RF access: Read always allowed, Write allowed if RF user security session is open 10: Area 2 RF access: Read allowed if RF user security session is open, Write allowed if RF user security session is open 11: Area 2 RF access: Read allowed if RF user security session is open, Write always forbidden	00b

Note: Refer to Table 13. System configuration memory map for the RFA2SS register.

Table 48. RFA3SS access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @08h Write Configuration (cmd code A1h) @08h	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 0008h	R always, W if I^2C security session is open

Table 49. RFA3SS

Bit	Name	Function	Factory value
b1-b0	PWD_CTRL_A3	00: Area 3 RF user security session can't be open by password 01: Area 3 RF user security session is open by RF_PWD_1 10: Area 3 RF user security session is open by RF_PWD_2 11: Area 3 RF user security session is open by RF_PWD_3	00b
b3-b2	RW_PROTECTION_A3	00: Area 3 RF access: Read always allowed / Write always allowed 01: Area 3 RF access: Read always allowed, Write allowed if RF user security session is open 10: Area 3 RF access: Read allowed if RF user security session is open, Write allowed if RF user security session is open 11: Area 3 RF access: Read allowed if RF user security session is open, Write always forbidden	00b
b7-b4	RFU	-	0000b

Note: Refer to Table 13. System configuration memory map for the RFA3SS register.

Table 50. RFA4SS access

RF		I^2C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @0Ah Write Configuration (cmd code A1h) @0Ah	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 000Ah	R always, W if I^2C security session is open

Table 51. RFA4SS

Bit	Name	Function	Factory value
b1-b0	PWD_CTRL_A4	00: Area 4RF user security session can't be open by password 01: Area 4 RF user security session is open by RF_PWD_1 10: Area 4 RF user security session is open by RF_PWD_2 11: Area 4 RF user security session is open by RF_PWD_3	00b
b3-b2	RW_PROTECTION_A4	00: Area 4 RF access: Read always allowed, Write always allowed 01: Area 4 RF access: Read always allowed, Write allowed if RF user security session is open 10: Area 4 RF access: Read allowed if RF user security session is open, Write allowed if RF user security session is open 11: Area 4 RF access: Read allowed if RF user security session is open, Write always forbidden	00b
b7-b4	RFU	-	0000b

Note: Refer to Table 13. System configuration memory map for the RFA4SS register.

Table 52. I2CSS access

RF		I^2C	
Command	Type	Address	Type
No access		E2 = 1, E1 = 1, 000Bh	R always, W if I^2C security session is open

Table 53. I2CSS

Bit	Name	Function	Factory value
b1-b0	RW_PROTECTION_A1	00: Area 1 I^2C access: Read always allowed, Write always allowed 01: Area 1 I^2C access: Read always allowed, Write allowed if I^2C user security session is open 10: Area 1 I^2C access: Read always allowed, Write always allowed 11: Area 1 I^2C access: Read always allowed, Write allowed if I^2C user security session is open	00b
b3-b2	RW_PROTECTION_A2	00: Area 2 I^2C access: Read always allowed, Write always allowed 01: Area 2 I^2C access: Read always allowed, Write allowed if I^2C user security session is open 10: Area 2 I^2C access: Read allowed if I^2C user security session is open, Write always allowed 11: Area 2 I^2C access: Read allowed if I^2C security session is open, Write allowed if I^2C security session is open	00b
b5-b4	RW_PROTECTION_A3	00: Area 3 I^2C access: Read always allowed, Write always allowed 01: Area 3 I^2C access: Read always allowed, Write allowed if I^2C user security session is open 10: Area 3 I^2C access: Read allowed if I^2C user security session is open, Write always allowed 11: Area 3 I^2C access: Read allowed if I^2C security session is open, Write allowed if I^2C security session is open	00b
b7-b6	RW_PROTECTION_A4	00: Area 4 I^2C access: Read always allowed, Write always allowed 01: Area 4 I^2C access: Read always allowed, Write allowed if I^2C user security session is open 10: Area 4 I^2C access: Read allowed if I^2C user security session is open, Write always allowed 11: Area 4 I^2C access: Read allowed if I^2C security session is open, Write allowed if I^2C security session is open	00b

Note: Refer to Table 13. System configuration memory map for the I2CSS register.

Table 54. LOCK_CCFILE access

RF		I²C	
Command	Type	Address	Type
Lock Block (cmd code 22h) @00h/01h (1) Ext Lock Block (cmd code 32h) @00h/01h Read Block (cmd code 20h) @00h/01h Fast Read Block(1) (cmd code C0h) @00h/01h Ext Read Block(1) (cmd code 30h) @00h/01h Fast Ext Read Block(1) (cmd code C4h) @00h/01h Read Multi Block(1) (cmd code 23h) @00h/01h Ext Read Multi Block(1) (cmd code 33h) @00h/01h Fast Read Multi Block(1) (cmd code C3h) @00h/01h Fast Ext Read Multi Block(1) (cmd code C5h) @00h/01h Get Multi Block SS (cmd code 2Ch) @00h/01h Ext Get Multi Block SS (cmd code 3Ch) @00h/01h	R always b0: W if Block 00h is not already locked, b1: W if Block 01h is not already locked.	E2 = 1, E1 = 1, 000Ch	R always, W if I²C security session is open

1. *With option flag set to 1.*

Table 55. LOCK_CCFILE

Bit	Name	Function	Factory value
b0	LCKBCK0	0: Block @ 00h is not Write locked 1: Block @ 00h is Write locked	0b
b1	LCKBCK1	0: Block @ 01h is not Write locked 1: Block @ 01h is Write locked	0b
b7-b2	RFU	-	000000b

Note: Refer to Table 13. System configuration memory map for the LOCK_CCFILE register.

Table 56. LOCK_CFG access

RF		I²C	
Command	Type	Address	Type
Read Configuration (cmd code A0h) @0Fh Write Configuration (cmd code A1h) @0Fh	R always, W if RF configuration security session is open and configuration not locked	E2 = 1, E1 = 1, 000Fh	R always, W if I²C security session is open

Table 57. LOCK_CFG

Bit	Name	Function	Factory value
b0	LCK_CFG	0: Configuration is unlocked 1: Configuration is locked for RF write access	0b
b7-b1	RFU	-	0000000b

Note: Refer to Table 13. System configuration memory map for the LOCK_CFG register.

Table 58. I2C_PWD access

RF		I²C	
Command	Type	Address	Type
No access		E2 = 1, E1 = 1, 0900h to 0907h, Present/Write password command format.	R if I²C security session is open, W if I²C security session is open

Table 59. I2C_PWD

I²C address	Bit	Name	Function	Factory value
0900h	b7-b0	I2C_PWD	Byte 7 (MSB) of password for I²C security session	00h
0901h	b7-b0		Byte 6 of password for I²C security session	00h
0902h	b7-b0		Byte 5 of password for I²C security session	00h
0903h	b7-b0		Byte 4 of password for I²C security session	00h
0904h	b7-b0		Byte 3 of password for I²C security session	00h
0905h	b7-b0		Byte 2 of password for I²C security session	00h
0906h	b7-b0		Byte 1 of password for I²C security session	00h
0907h	b7-b0		Byte 0 (LSB) of password for I²C security session	00h

Note: Refer to Table 13. System configuration memory map for the I2C_PWD register.

Table 60. RF_PWD_0 access

RF		I²C	
Command	Type	Address	Type
Present Password (cmd code B3h) Write Password (cmd code B1h)	WO if RF configuration security session is open	No access	

Table 61. RF_PWD_0

Bit	Name	Function	Factory value
b7-b0	RF_PWD_0	Byte 0 (LSB) of password for RF configuration security session	00h
		Byte 1 of password for RF configuration security session	00h
		Byte 2 of password for RF configuration security session	00h
		Byte 3 of password for RF configuration security session	00h
		Byte 4 of password for RF configuration security session	00h
		Byte 5 of password for RF configuration security session	00h
		Byte 6 of password for RF configuration security session	00h
		Byte 7 (MSB) of password for RF configuration security session	00h

Note: Refer to Table 13. System configuration memory map for the RF_PWD_0 register.

Table 62. RF_PWD_1 access

RF		I²C	
Command	Type	Address	Type
Present Password (cmd code B3h) Write Password (cmd code B1h)	WO if RF configuration security session is open with RF password 1	No access	

Table 63. RF_PWD_1

Bit	Name	Function	Factory value
b7-b0	RF_PWD_1	Byte 0 (LSB) of password 1 for RF user security session	00h
		Byte 1 of password 1 for RF user security session	00h
		Byte 2 of password 1 for RF user security session	00h
		Byte 3 of password 1 for RF user security session	00h
		Byte 4 of password 1 for RF user security session	00h
		Byte 5 of password 1 for RF user security session	00h
		Byte 6 of password 1 for RF user security session	00h
		Byte 7 (MSB) of password 1 for RF user security session	00h

Note: Refer to Table 13. System configuration memory map for the RF_PWD_1 register.

Table 64. RF_PWD_2 access

RF		I²C	
Command	Type	Address	Type
Present Password (cmd code B3h) Write Password (cmd code B1h)	WO if RF user security session is open with RF password 2	No access	

Table 65. RF_PWD_2

Bit	Name	Function	Factory value
b7-b0	RF_PWD_2	Byte 0 (LSB) of password 2 for RF user security session	00h
		Byte 1 of password 2 for RF user security session	00h
		Byte 2 of password 2 for RF user security session	00h
		Byte 3 of password 2 for RF user security session	00h
		Byte 4 of password 2 for RF user security session	00h
		Byte 5 of password 2 for RF user security session	00h
		Byte 6 of password 2 for RF user security session	00h
		Byte 7 (MSB) of password 2 for RF user security session	00h

Note: Refer to Table 13. System configuration memory map for the RF_PWD_2 register.

Table 66. RF_PWD_3 access

RF		I²C	
Command	Type	Address	Type
Present Password (cmd code B3h) Write Password (cmd code B1h)	WO if RF user security session is open with RF password 3	No access	

Table 67. RF_PWD_3

Bit	Name	Function	Factory value
b7-b0	RF_PWD_3	Byte 0 (LSB) of password 3for RF user security session	00h
		Byte 1 of password 3 for RF user security session	00h
		Byte 2 of password 3 for RF user security session	00h
		Byte 3 of password 3 for RF user security session	00h
		Byte 4 of password 3 for RF user security session	00h
		Byte 5 of password 3 for RF user security session	00h
		Byte 6 of password 3 for RF user security session	00h
		Byte 7 (MSB) of password 3 for RF user security session	00h

Note: Refer to Table 13. System configuration memory map for the RF_PWD_3 register.

Table 68. I2C_SSO_Dyn access

RF		I²C	
Command	Type	Address	Type
No access		E2 = 0, E1 = 1, 2004h	RO

Table 69. I2C_SSO_Dyn

Bit	Name	Function	Factory value
b7-b1	RFU	-	0b
b0	I2C_SSO	0: I²C security session close 1: I²C security session open (set or reset via I²C Present password command)	0b

Note: Refer to Table 13. System configuration memory map for the I2C_SSO_Dyn register.

5.6.2 Passwords and security sessions

The device provides protection of user memory and system configuration static registers. RF user and I²C host can access those protected data by opening security sessions with passwords. Access rights are more restricted when security sessions are closed, and less restricted when security sessions are open.

Dynamic registers and fast transfer mode mailbox are not protected by any security session.

There are three type of security sessions, as shown in the following table.

Table 70. Security session type

Security session	Open by presenting	Right granted when security session is open, and until it is closed
RF user	RF password 1, 2 or 3 (1) (RF_PWD_1, RF_PWD_2, RF_PWD_3)	RF user access to protected user memory as defined in RFA$_i$SS registers RF user write access to RF password 1, 2 or 3(2)
RF configuration	RF password 0 (RF_PWD_0)	RF user write access to configuration static registers RF user write access to RF password 0
I^{2}C	I^{2}C password (I2C_PWD)	I^{2}C host access to protected user memory as defined in I2CSS register I^{2}C host write access to configuration static registers I^{2}C host write access to I^{2}C password

1. *Password number must be the same as the one selected for protection.*
2. *Write access to the password number corresponding to the password number presented.*

All passwords are 64-bit long, default value is 0000000000000000h.

The passwords management is organized around RF and I^{2}C dedicated set of commands to access the dedicated registers in system configuration area where passwords are stored.

The dedicated password commands in RF mode are:

- Write Password command (code B1h): see Section 7.6.36: Present Password.
- Present Password command (code B3h): see Section 7.6.36: Present Password.

RF user possible actions for security sessions are:

- **Open RF user security session**: Present Password command, with password number 1, 2 or 3 and the valid corresponding password
- **Write RF password**: Present Password command, with password number (0, 1, 2 or 3) and the current valid corresponding password. Then Write Password command, with same password number (0, 1, 2 or 3) and the new corresponding password.
- **Close RF user security session**: Present Password command, with a different password number than the one used to open session or any wrong password. Or remove tag from RF field (POR). Presenting a password with an invalid password number doesn't close the session.
- **Open RF configuration security session**: Present Password command, with password number 0 and the valid password 0.
- **Close RF configuration security session**: Present Password command, with a password number different than 0, or password number 0 and wrong password 0. Or remove tag from RF field (POR). Presenting a password with an invalid password number doesn't close the session.

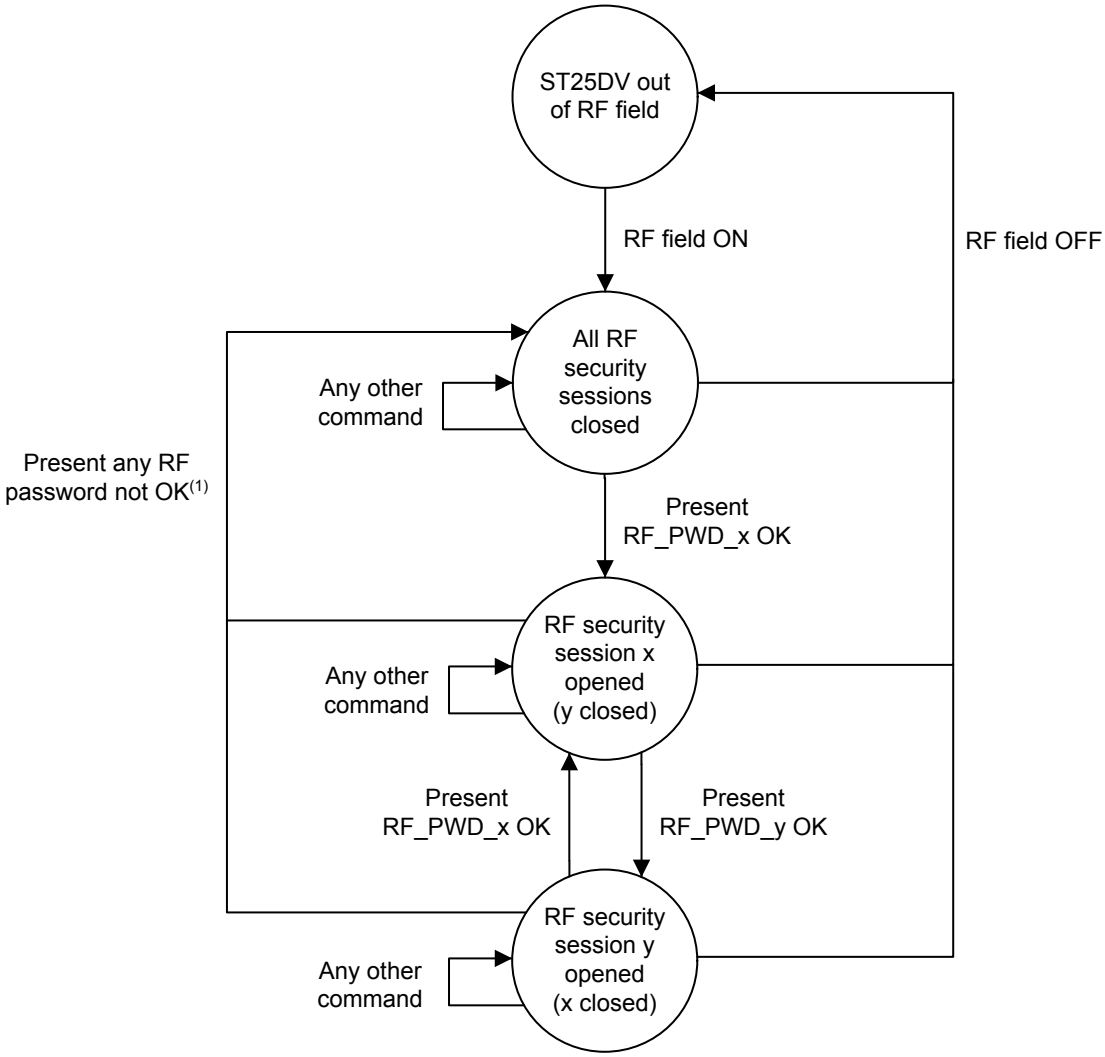
Opening any new RF security session (user or configuration) automatically closes the previously open one.

There is no interaction between I^{2}C and RF security sessions. Both are independent, and can run in parallel.

Caution: *If the device is powered through V_{CC}, removing V_{CC} during an RF command can abort it. As a consequence, before writing a new password, RF user must check if V_{CC} is ON, by reading EH_CTRL_Dyn register bit 3 (VCC_ON), and ask the host to maintain or to shut down V_{CC}, before issuing the Write Password command, to avoid password corruption.*

To make the application more robust, it is recommended to use addressed or selected mode during write password operations to get the traceability of which tags/UID have been programmed.

Figure 28. RF security sessions management



1. Presenting a password with an invalid password number doesn't close the session.

The dedicated password commands in I²C mode are:

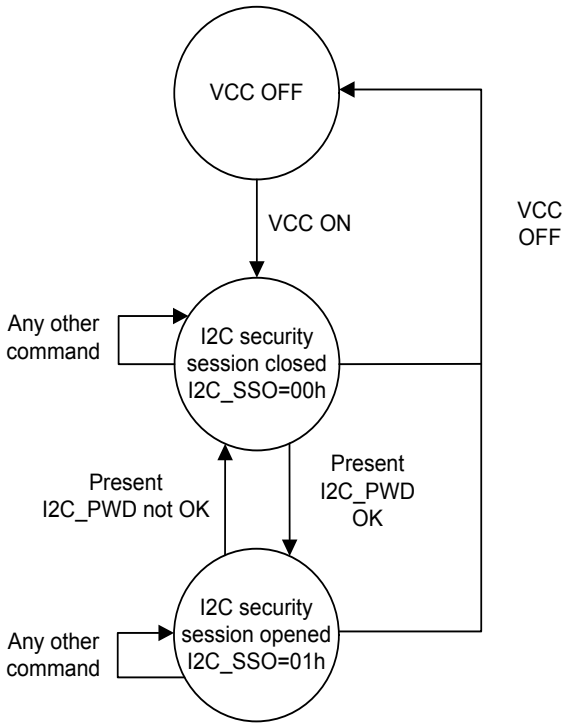
- I²C Write Password command: see Section 6.6.2: I²C write password command description.
- I²C Present Password command: see Section 6.6.1: I²C present password command description.

I²C host possible actions for security sessions are:

- **Open I ² C security session**: I²C Present Password command with valid I²C password.
- **Write I ² C password**: I²C Present Password command with valid I²C password. Then I²C Write Password command with new I²C password.
- **Close I ² C security session**: I²C Present Password command with wrong I²C password. Or remove tag V_{CC} power supply (POR).
- **Check if I ² C security session is open**: I²C host can read the current status (open or closed) of I²C security session by reading the I2C_SSO_Dyn register.

There is no interaction between I²C and RF security sessions. Both are independent and can run in parallel.

Figure 29. I²C security sessions management



5.6.3 User memory protection

On factory delivery, areas are not protected.

Each area can be individually protected in read and/or write access from RF and I²C.

Area 1 is always readable (from RF and I²C).

Furthermore, RF blocks 0 and 1 (I²C bytes 0000h to 0007h) can be independently write locked.

User memory protection from RF access

In RF mode, each memory area of the ST25DVxxKC can be individually protected by one out of three available passwords (RF password 1, 2 or 3), and each area can also have individual Read/Write access conditions.

For each area, an RFA$_i$SS register is used to:

- Select the RF password that unlock the RF user security session for this area
- Select the protection against read and write operations for this area

(See Table 45. RFA1SS, Table 47. RFA2SS, Table 49. RFA3SS, and Table 51. RFA4SS for details about available read and write protections).

Note: Setting 00b in PWD_CTRL_A$_i$ field means that RF user security session cannot be open by any password for the corresponding area.

When updating RFA$_i$SS registers, the new protection value is effective immediately after the register write completion.

- Rf blocks 0 and 1 are exceptions to this protection mechanism:
  - RF blocks 0 and 1 can be individually write locked by issuing a (Ext) Lock Single Block RF command. Once locked, they cannot be unlock through RF. LOCK_CCFILE register is automatically updated when using (Ext) Lock Single Block command.
  - An RF user needs no password to lock blocks 0 and/or 1.
  - Locking blocks 0 and/or 1 is possible even if the configuration is locked (LOCK_CFG=1).
  - Locking blocks 0 and/or 1 is possible even if the area is write locked.
  - Unlocking area1 (through RFA1SS register) does not unlock blocks 0 and 1 if they have been locked though (Ext) Lock Block command.
  - Once locked, the RF user cannot unlock blocks 0 and/or 1 (can be done by I^2C host).

Note: When areas size are modified (ENDAi registers), RFAiSS registers are not modified.

User memory protection from I^2C access

In I^2C mode, each area can also have individual Read/Write access conditions, but only one I^2C password is used to unlock I^2C security session for all areas.

The I2CSS register is used to set protection against read and write operation for each area (see Table 53. I2CSS for details about available read and write protections).

When updating I2CSS registers, the new protection value is effective immediately after the register write completion.

I^2C user memory Bytes 0000h to 0003h (RF Block 0) and 0004h to 0007h (RF Block 1) can be individually locked and unlocked by writing in the LOCK_CCFILE register (by group of 4 Bytes), independently of Area 1 protection. Unlocking Area 1 (through I2CSS register) does not unlock those bytes if they have been locked though the LOCK_CCFILE register.

Note: When areas size are modified (ENDAi registers), I2CSS register is not modified.

Retrieve the security status of a user memory block or byte

RF user can read a block security status by issuing following RF commands:

- (Ext) Get Multiple Blocks Security Status command.
- (Ext) (Fast) Read Single Block with option flag set to 1.
- (Ext) (Fast) Read Multiple Blocks with option flag set to 1.

ST25DV responds with a Block security status containing a Lock_bit flag as specified in ISO 15693 standard. This lock_bit flag is set to one if block is locked against write.

Lock_bit flag value may vary if corresponding RF user security session is open or closed.

I^2C host can retrieve a block security status by reading the I2CSS register to get security status of the corresponding area and by reading the I2C_SSO_Dyn register to know if I^2C security session is open or closed.

For blocks 0 and 1 (Bytes 0000h to 0007h in I^2C user memory), lock status can also be read in the LOCK_CCFILE register.

5.6.4 System memory protection

By default, system memory (static registers) is write protected, both in RF and I^2C.

I^2C host must open the I^2C security session (by presenting a valid I^2C password) to enable write access to system configuration static registers.

I^2C host doesn't have read or write access to RF passwords.

By default, I^2C host can read all system configuration static registers (except RF passwords)

In RF, to enable write access to system configuration static registers, RF user must open the RF configuration security session (by presenting a valid RF password 0) and system configuration must not be locked (LOCK_CFG=00h).

RF doesn't have read or write access to I^2C password.

By default, RF user can read all system configuration static registers, except all passwords, LOCK_CCFILE, LOCK_DSFID and LOCK_AFI.

RF configuration lock:

- RF write access to system configuration static registers can be locked by writing 01h in the LOCK_CFG register (by RF or I^2C).
- RF user cannot unlock system configuration if LOCK_CFG=01h, even after opening RF configuration security session (only I^2C host can unlock system configuration).
- When system configuration is locked (LOCK_CFG=01h), it is still possible to change RF passwords (0 to 3).

Device identification registers:

- AFI and DFSID registers can be independently locked by RF user, issuing respectively a Lock AFI and a Lock DSFID command. Lock is definitive: once locked, AFI and DSFID registers cannot be unlocked (either by RF or I^2C). System configuration locking mechanism (LOCK_CFG=01h) does not lock AFI and DSFID registers.
- Other device identification registers (MEM_SIZE, BLK_SIZE, IC_REF, UID, IC_REV) are read only registers for both RF and I^2C.

5.7 Device parameter registers

Table 71. LOCK_DSFID access

RF		I^2C	
Command	Type	Address	Type
Lock DSFID (cmd code 2Ah)	WO if DSFID not locked	E2=1, E1=1, 0010h	RO

Table 72. LOCK_DSFID

Bit	Name	Function	Factory value
b0	LOCK_DSFID	0: DSFID is not locked 1: DSFID is locked	0b
b7-b1	RFU	-	0000000b

Note: Refer to Table 13. System configuration memory map for the LOCK_DSFID register.

Table 73. LOCK_AFI access

RF		I^2C	
Command	Type	Address	Type
Lock AFI (cmd code 28h)	WO if AFI not locked	E2=1, E1=1, 0011h	RO

Table 74. LOCK_AFI

Bit	Name	Function	Factory value
b0	LOCK_AFI	0: AFI is not locked 1: AFI is locked	0b
b7-b1	RFU	-	0000000b

Note: Refer to Table 13. System configuration memory map for the LOCK_AFI register.

Table 75. DSFID access

RF		I²C	
Command	Type	Address	Type
Inventory (cmd code 01h) Get System Info (cmd code 2Bh) Ext Get System Info (cmd code 3Bh) Write DSFID (cmd code 28h)	R always, W if DSFID not locked	E2=1, E1=1, 0012h	RO

Table 76. DSFID

Bit	Name	Function	Factory value
b7-b0	DSFID	ISO/IEC 15693 Data Storage Format Identifier	00h

Note: Refer to Table 13. System configuration memory map for the DSFID register.

Table 77. AFI access

RF		I²C	
Command	Type	Address	Type
Inventory (cmd code 01h) Get System Info (cmd code 2Bh) Ext Get System Info (cmd code 3Bh) Write AFI (cmd code 27h)	R always, W if AFI not locked	E2=1, E1=1, 0013h	RO

Table 78. AFI

Bit	Name	Function	Factory value
b7-b0	AFI	ISO/IEC 15693 Application Family Identifier	00h

Note: Refer to Table 13. System configuration memory map for the AFI register.

Table 79. MEM_SIZE access

RF		I²C	
Command	Type	Address	Type
Get System Info (cmd code 2Bh) (1) Ext Get System Info (cmd code 3Bh)	RO	E2=1, E1=1, 0014h to 0015h	RO

1. *Only ST25DV04KC*

Table 80. MEM_SIZE

I²C Address	Bit	Name	Function	Factory value
0014h	b7-b0	MEM_SIZE	Address 0014h: LSB byte of the memory size expressed in RF blocks	ST25DV04KC: 7Fh ST25DV16KC: FFh ST25DV64KC: FFh
0015h	b7-b0		Address 0015h: MSB byte of the memory size expressed in RF blocks	ST25DV04KC: 00h ST25DV16KC: 01h ST25DV64KC: 07h

Note: Refer to Table 13. System configuration memory map for the MEM_SIZE register.

Table 81. BLK_SIZE access

RF		I²C	
Command	Type	Address	Type
Get System Info (cmd code 2Bh) (1) Ext Get System Info (cmd code 3Bh)	RO	E2=1, E1=1, 0016h	RO

1. *Only ST25DV04KC*

Table 82. BLK_SIZE

Bit	Name	Function	Factory value
b7-b0	BLK_SIZE	RF user memory block size	03h

Note: Refer to Table 13. System configuration memory map for the BLK_SIZE register.

Table 83. IC_REF access

RF		I²C	
Command	Type	Address	Type
Get System Info (cmd code 2Bh) Ext Get System Info (cmd code 3Bh)	RO	E2=1, E1=1, 0017h	RO

Table 84. IC_REF

Bit	Name	Function	Factory value
b7-b0	IC_REF	ISO/IEC 15693 IC Reference	ST25DV04KC-IE: 50h ST25DV16KC-IE: 51h ST25DV64KC-IE: 51h ST25DV04KC-JF: 50h ST25DV16KC-JF: 51h ST25DV64KC-JF: 51h

Note: Refer to Table 13. System configuration memory map for the IC_REF register.

Table 85. UID access

RF		I²C	
Command	Type	Address	Type
Inventory (cmd code 01h) Get System Info (cmd code 2Bh) Ext Get System Info (cmd code 3Bh)	RO	E2=1, E1=1, 0018h to 001Fh	RO

Table 86. UID

I²C Address	Bit	Name	Function	Factory ²alue
0018h	b7-b0	UID	ISO/IEC 15693 UID byte 0 (LSB)	IC manufacturer serial number
0019h			ISO/IEC 15693 UID byte 1	
001Ah			ISO/IEC 15693 UID byte 2	
001Bh			ISO/IEC 15693 UID byte 3	
001Ch			ISO/IEC 15693 UID byte 4	
001Dh			ISO/IEC 15693 UID byte 5: ST Product code	ST25DV04KC-IE: 50h ST25DV16KC-IE: 51h ST25DV64KC-IE: 51h ST25DV04KC-JF: 52h ST25DV16KC-JF: 53h ST25DV64KC-JF: 53h
001Eh			ISO/IEC 15693 UID byte 6: IC Mfg code	02h
001Fh			ISO/IEC 15693 UID byte 7 (MSB)	E0h

Note: Refer to Table 13. System configuration memory map for the UID register.

Table 87. IC_REV access

RF		I²C	
Command	Type	Address	Type
No access		E2=1, E1=1, 0020h	RO

Table 88. IC_REV

Bit	Name	Function	Factory value
b7-b0	IC_REV	IC revision	Depending on revision

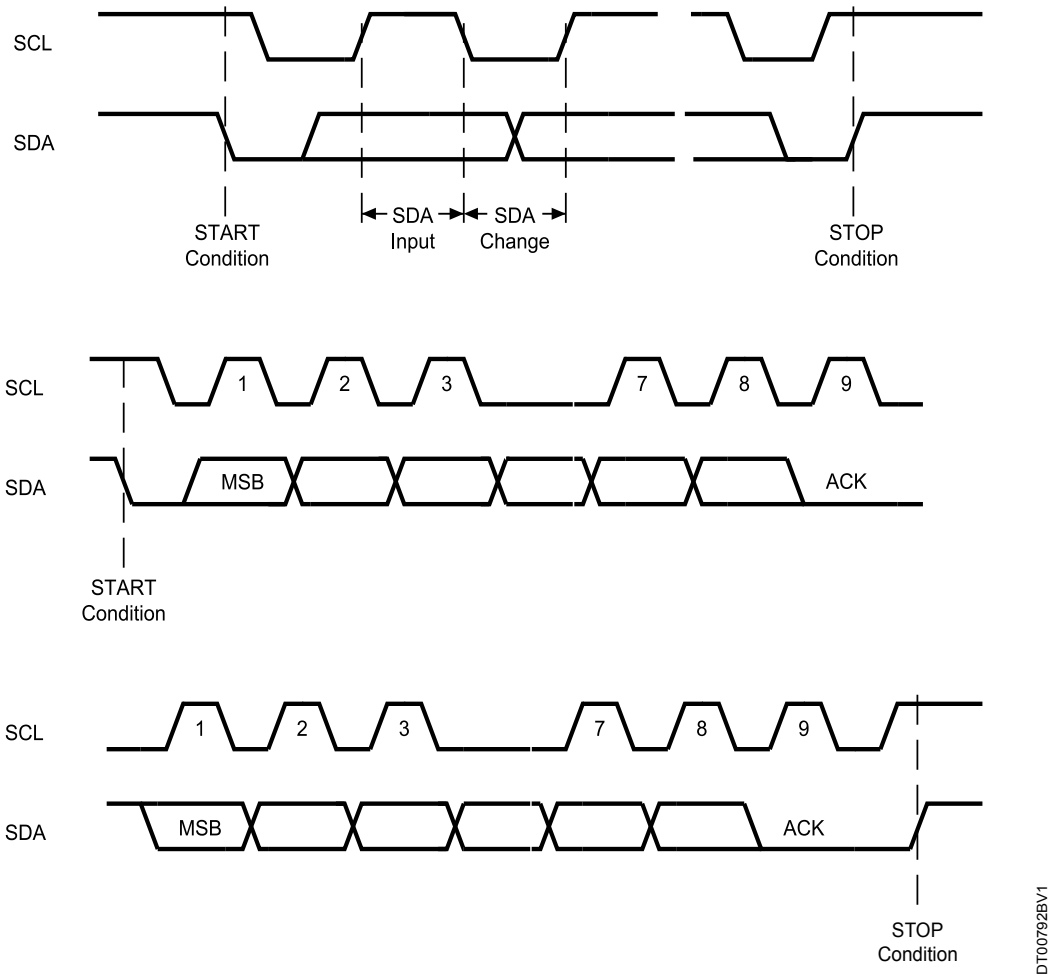
Note: Refer to Table 13. System configuration memory map for the IC_REV register.

6 I²C operation

6.1 I²C protocol

The device supports the I²C protocol. This is summarized in Figure 30. I²C bus protocol. Any device that sends data to the bus is defined as a transmitter, and any device that reads data is defined as a receiver. The device that controls the data transfer is known as the bus controller, and the other as the target device. A data transfer can only be initiated by the bus controller, which also provides the serial clock for synchronization. The ST25DVxxKC device is a target in all communications.

Figure 30. I²C bus protocol



6.1.1 Start condition

Start is identified by a falling edge of serial data (SDA) while the serial clock (SCL) is stable in the high state. A Start condition must precede any data transfer command. The device continuously monitors (except during a write cycle) the SDA and the SCL for a Start condition, and does not respond unless one is given.

6.1.2 Stop condition

Stop is identified by a rising edge of serial data (SDA) while the serial clock (SCL) is stable and driven high. A Stop condition terminates communication between the device and the bus controller. A Read command that is followed by NoAck can be followed by a Stop condition to force the device into the Standby mode. A Stop condition at the end of a Write command triggers the internal write cycle.

6.1.3 Acknowledge bit (ACK)

The acknowledge bit is used to indicate a successful byte transfer. The bus transmitter, whether a bus controller or a target device, releases the serial data (SDA) after sending eight bits of data. During the ninth clock pulse period, the receiver pulls the SDA low to acknowledge the receipt of the eight data bits.

6.1.4 Data input

During data input, the device samples serial data (SDA) on the rising edge of the serial clock (SCL). For correct device operation, the SDA must be stable during the rising edge of the SCL, and the SDA signal must change only when the SCL is driven low.

6.2 I²C timeout

During the execution of an I²C operation, RF communications are not possible.

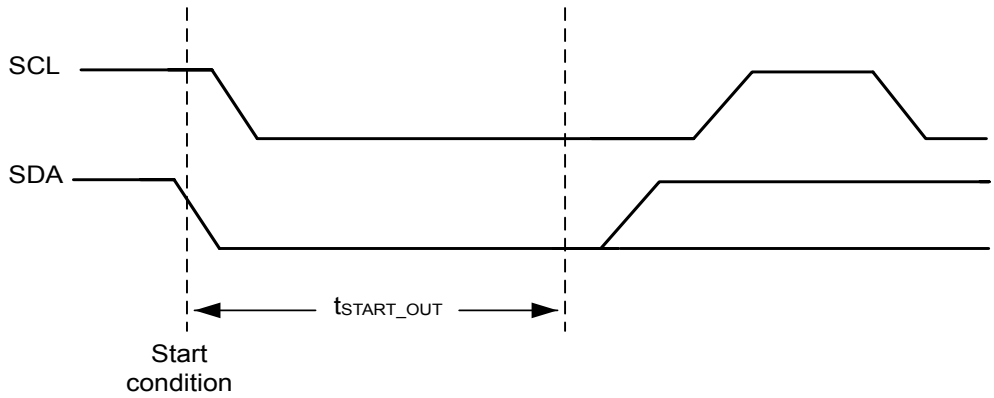
To prevent RF communication freezing due to inadvertent indeterminate instructions sent to the I²C bus, the ST25DVxxKC features a timeout mechanism that automatically resets the I²C logic block.

6.2.1 I²C timeout on Start condition

I²C communication with the ST25DVxxx starts with a valid Start condition, followed by a device select code.

If the delay between the Start condition and the following rising edge of the serial clock (SCL) that samples the most significant of the device select exceeds the t_{START_OUT} time (see Table 249. I²C DC characteristics up to 85 °C and Table 250. I²C DC characteristics up to 125 °C), the I²C logic block is reset and further incoming data transfer is ignored until the next valid Start condition.

Figure 31. I²C timeout on Start condition



6.2.2 I²C timeout on clock period

During data transfer on the I²C bus, if the serial clock pulse width high (t_{CHCL}) or serial clock pulse width low (t_{CLCH}) exceeds the maximum value specified in Table 251. I²C AC characteristics up to 85 °C and Table 252. I²C AC characteristics up to 125 °C, the I²C logic block is reset and any further incoming data transfer is ignored until the next valid Start condition.

6.3 Device addressing

To start a communication between the bus controller and the target device, the bus controller must initiate a Start condition. Following this, the bus controller sends the device select code, shown in Appendix B.1: Device select codes (on serial data (SDA), the most significant bit first).

The device select code consists of a 4-bit device type identifier (I2C_DEVICE_CODE) and a 3-bit chip enable "Address" (E2, E1, E0). Chip Enable bits E2 and E1 are used to select ST25DVxxKC memory to address (user or system) and to send special I²C "RFSwitchOff"and I2C "RFSwitchOn" commands.

The eighth bit is the Read/Write bit (RW). It is set to 1 for read and to 0 for write operations. Refer to the table below.

Table 89. Device select code

ST25DvxxKC function	I²C device type identifier				E2	E1	E0	R/notW
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
User memory	I2C_DEVICE_CODE[3:0]				0	1	I2C_E0	1/0
System memory					1	1		1/0
I2C RFSwitchOn					0	0		0
I2C RFSwitchOff					1	0		0

The 4-bit device type identifier and the chip enable bit E0 are configurable through the I2C_CFG static register.

Table 90. I2C_CFG access

RF		I2C	
Command	Type	Address	Type
No access		E2=1, E1=1, 000Eh	R always, W if the I2C security session is open

Table 91. I2C_CFG

Bit	Name	Function	Factory value
b3-b0	I2C_DEVICE_CODE	Device code (bits [7:4]) of I2C target address	1010b
b4	I2C_E0	E0 bit (bit 1) of I2C target address	1b
b5	I2C_RF_SWITCHOFF_EN	0: I2C cannot switch off/on RF with I2C « RFSwitchOff/On » commands. 1: I2C can switch off/on RF with i2C « RFSwitchOfff/On » commands	0b
b7-b6	RFU	-	00b

Note: Refer to Table 13. System configuration memory map for the UID register.

Change in I2C_CFG command is immediate after the STOP condition of the I2C write to this register. Next I2C accesses shall use the new value of I2C_DEVICE_CODE and I2C_E0 to address the ST25DVxxKC.

If a match occurs on the device select code, the corresponding device gives an acknowledgment on serial data (SDA) during the ninth bit time. If the device does not match the device select code, it deselects itself from the bus, and goes into Standby mode.

Table 92. Operating modes

Mode	RW bit	Bytes	Initial sequence
Current address read	1	1	Start, device select, RW = 1
Random address read	0	1	Start, device select, RW = 0, address
	1		reStart, device select, RW = 1
Sequential read	1	≥ 1	Similar to the current or random address read
Byte write	0	1	Start, device select, RW = 0
Sequential write	0	≤ 256 bytes	Start, device select, RW = 0

6.4 I²C write operations

Following a Start condition, the bus controller sends a device select code with the Read/Write bit (RW) reset to 0. The device acknowledges this, and waits for two address bytes. The device responds to each address byte with an acknowledge bit, and then waits for the data byte.

Each data byte in the memory has a 16-bit (two bytes) address. The most significant byte (see Table 93) is sent first, followed by the least significant byte (see Table 94). The bits from b15 to b0 form the address of the byte in memory.

Table 93. Address most significant byte

b15	b14	b13	b12	b11	b10	b9	b8

Table 94. Address least significant byte

b7	b6	b5	b4	b3	b2	b1	b0

When the bus controller generates a Stop condition immediately after the Ack bit (in the tenth-bit time slot), either at the end of a byte write or a sequential write, the internal write cycle is triggered. A Stop condition at any other time slot does not trigger the internal write cycle.

After the Stop condition, the delay t_W, and the successful completion of a Write operation, the device's internal address counter is incremented automatically, to point to the next byte address after the last one that was modified.

After an unsuccessful write operation, the device enters in I²C dead state: the internal address counter is not incremented, and waits for a full new I²C instruction (the address counter stops to be incremented after the first NoAck bit).

During the internal write cycle, the serial data (SDA) signal is disabled, and the device does not respond to any requests.

Caution: *I²C writing data in user memory (EEPROM) transit via the 256 bytes fast transfer mode buffer. Fast transfer mode must be deactivated before starting any write operation in user memory, otherwise the command is not acknowledged, programming is not done, and the device goes in standby mode.*

6.4.1 I²C byte write

After the device select code and the address bytes, the bus controller sends one data byte.

If byte write is not inhibited, the device replies with Ack.

If byte write is inhibited, the device replies with NoAck.

The bus controller terminates the transfer by generating a Stop condition (see Figure 32. Write mode sequences when write is not inhibited).

For byte write in EEPROM (user memory or system configuration), internal programming starts after the STOP condition, for a duration of t_W (as defined in Table 249. I²C DC characteristics up to 85 °C and Table 250. I²C DC characteristics up to 125 °C).

For writes in fast transfer mode buffer or dynamic registers, internal programming is immediate at the STOP condition.

If byte write is inhibited, the device replies with NoAck. The bus controller terminates the transfer by generating a Stop condition and the byte location is not modified (see Figure 33. Write mode sequences when write is inhibited).

Byte write is inhibited if the byte complies with one of the following conditions:

- The byte is in the user memory and is write protected with the LOCK_CCFILE register.
- The byte is in the user memory and is write protected with the I2CSS register, and the I²C security session is closed.
- The byte is in the user memory and fast transfer mode is activated.
- The byte is in the system memory and is a Read only register.
- The byte is in the system memory and the I²C security session is closed.
- The byte is in the fast transfer mode's mailbox and is not the first byte of the mailbox.
- The byte is in the fast transfer mode's mailbox and the mailbox is busy.
- The byte is in the fast transfer mode's mailbox and the fast transfer mode is not activated.
- The byte is in the dynamic registers area and is a Read only register.

6.4.2 I²C sequential write

The I²C sequential write allows up to 256 bytes to be written in one command, provided they are all located in the same user memory area and are all located in writable addresses.

After each byte is transferred, the internal byte address counter is incremented.

For each byte sent by the bus controller:

- If byte write is not inhibited, the device replies with Ack.
- If byte write is inhibited, the device replies with NoAck.

The transfer is terminated by the bus controller generating a Stop condition:

- For writes in EEPROM (user memory or system configuration), if all bytes have been Ack'ed, internal programming of all bytes starts after the stop condition, for a duration dependent on the number of bytes to write (see below).
- For writes in fast transfer mode buffer or dynamic registers, if all bytes have been Ack'ed, internal programming is done immediately after the stop condition.
- If some bytes have been NotAck'ed, no internal programming is done (0 byte written).

Byte write is inhibited if the byte complies with the conditions described in Section 6.4.1: I²C byte write, in addition:

- The byte is in the user memory but does not belong to the same area than the previous received byte (area border crossing is forbidden).
- 256 write occurrences have already been reached in the same sequential write.
- More than one byte is trying to be written in the system area.

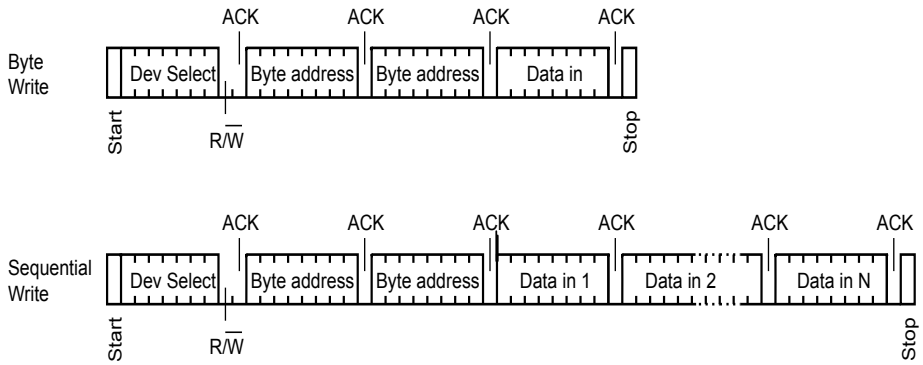
Seen from I²C, user memory is internally organized as rows of 16 bytes. Data located in the same row share the same most significant memory address bits b15 to b4.

I²C sequential write programming time in the EEPROM memory is dependent on this internal organization: total programming time is the I^2C write time t_W (as defined in Table 249 and Table 250) multiplied by the number of internal EEPROM pages where the data must be programmed, including incomplete pages.

This means an I²C sequential write allows from 1 up to 16 bytes to be programmed in EEPROM in t_W, provided that they all share the same most significant memory address bits b15 to b4.

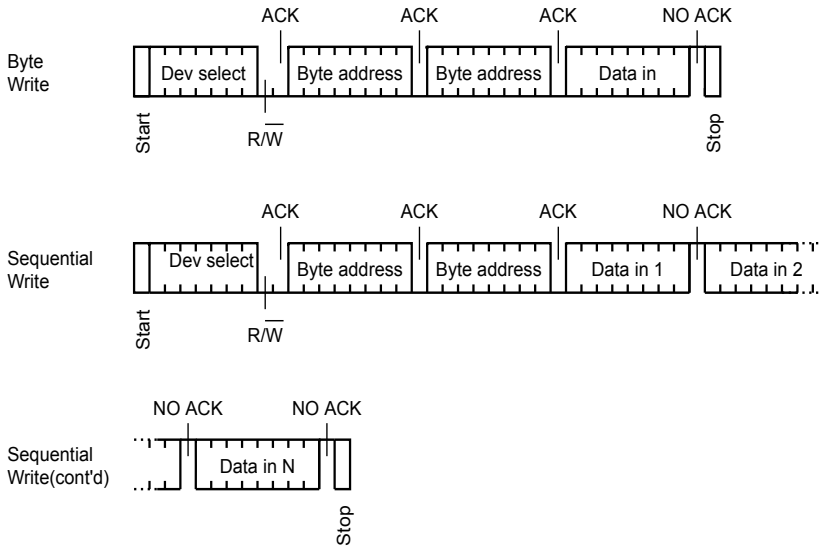
For example, a successful I²C sequential write of 40 bytes, starting at address 0010h, has a programming time (starting after the STOP condition) of 3 x t_W. An I²C sequential write of 40 bytes, starting at address 0008h, has a programming time of 4 x t_W.

Figure 32. Write mode sequences when write is not inhibited



Note: N ≤ 256

Figure 33. Write mode sequences when write is inhibited



Note: N ≤ 256

6.4.3 Minimizing system delays by polling on ACK

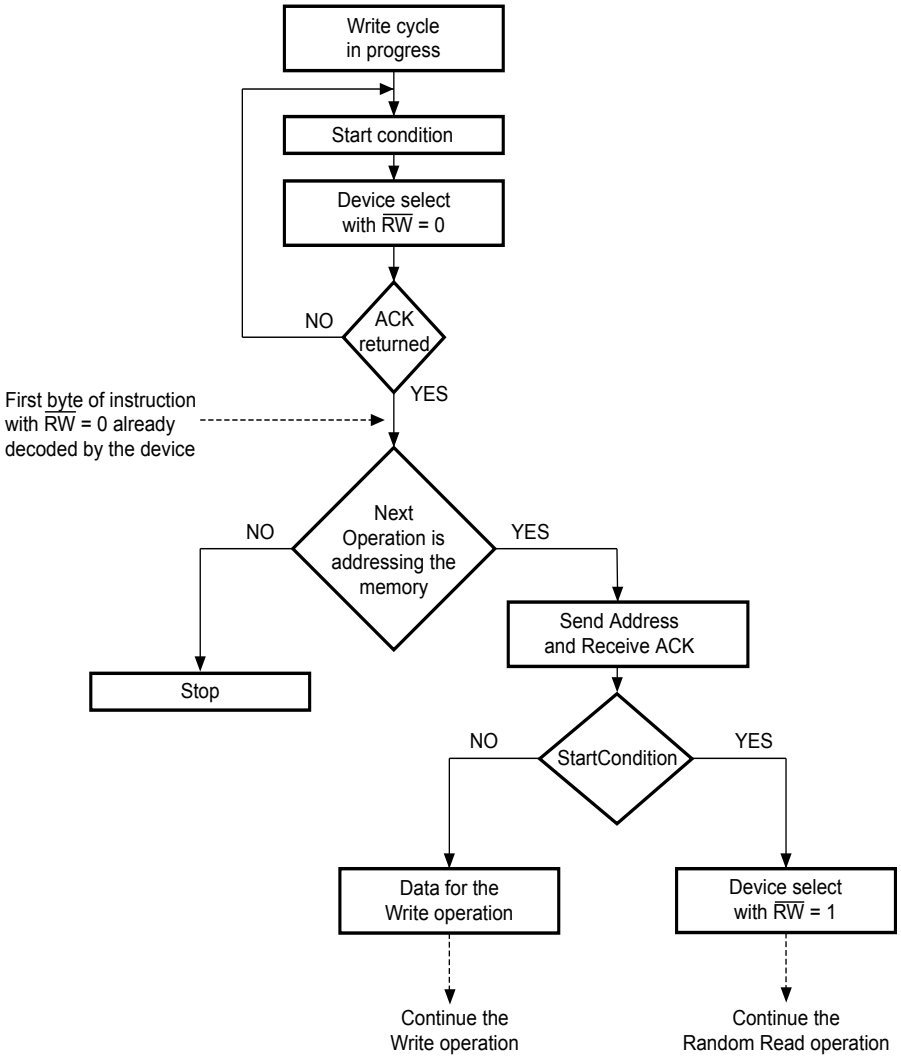
During the internal write cycle, the device disconnects itself from the bus, and writes a copy of the data from its internal latches to the memory cells. The maximum I²C write time (t_w) is shown in Table 251. I²C AC characteristics up to 85 °C and Table 252. I²C AC characteristics up to 125 °C, but the typical time is shorter. To make use of this, a polling sequence can be used by the bus controller.

The sequence, as shown in Figure 34. Write cycle polling flowchart using ACK is:

- Initial condition: a write cycle is in progress.
- Step 1: the bus controller issues a Start condition followed by a device select code (the first byte of the new instruction).
- Step 2: if the device is busy with the internal write cycle, no Ack is returned and the bus controller goes back to the Step 1. If the device has terminated the internal write cycle, it responds with an Ack, indicating that the device is ready to receive the second part of the instruction (the first byte of this instruction having been sent during Step 1).

Note: There is no need of polling when writing in dynamic registers or in the mailbox, since programming time is null.

Figure 34. Write cycle polling flowchart using ACK



6.5 I²C read operations

Read operation in user memory is performed successfully only if:

- Area to which the byte belongs is not read protected by the I2CSS register.
- Area to which the byte belongs is read protected by the I2CSS register, but I²C security session is open.

Read operations in system memory and dynamic registers are done independently of any protection mechanism, except I2C_PWD register which needs I²C security session to be open first.

Read operation in fast transfer mode's mailbox is performed successfully only if fast transfer mode is activated.

If read is not successful, ST25DVxxKC releases the bus and I²C host reads byte value FFh.

After the successful completion of a read operation, the device's internal address counter is incremented by one, to point to the next byte address.

After an unsuccessful read operation, ST25DVxxKC enters in I²C dead state: internal address counter is not incremented, and ST25DVxxKC is waiting for a full new I²C instruction.

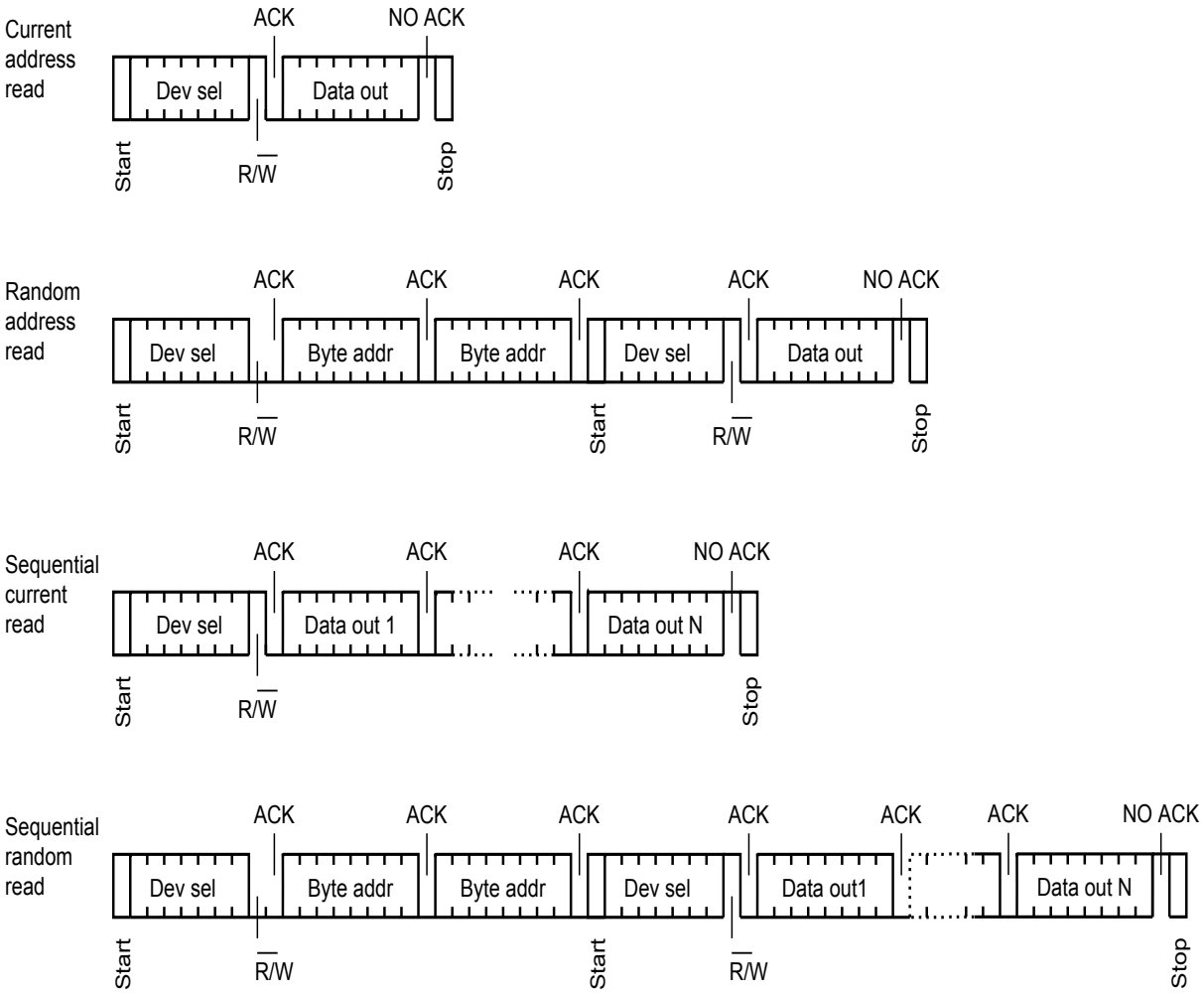
6.5.1 Random address read

A dummy write is first performed to load the address into this address counter (as shown in Figure 35), but without sending a Stop condition. Then, the bus controller sends another Start condition (reStart), and repeats the device select code, with the Read/Write bit (RW) set to 1. The device acknowledges this, and outputs the contents of the addressed byte. The bus controller must not acknowledge the byte, and terminates the transfer with a Stop condition.

6.5.2 Current address read

Following a Start condition, the bus controller sends only a device select code with the Read/Write bit ($R\overline{W}$) set to 1. The device acknowledges this, and outputs the byte addressed by the internal address counter. The counter is then incremented. The bus controller terminates the transfer with a Stop condition, as shown in the figure below, without acknowledging the byte.

Figure 35. Read mode sequences



6.5.3 Sequential read access

This operation can be used after a Current address read or a Random address read. The bus controller does acknowledge the data byte output, and sends additional clock pulses so that the device continues to output the next byte in sequence. To terminate the stream of bytes, the bus controller must not acknowledge the last byte, and must generate a Stop condition, as shown in Figure 35.

The output data comes from consecutive addresses, with the internal address counter automatically incremented after each byte output.

Sequential read in user memory:

- The Sequential read can cross area borders. The device continues to output data bytes until the internal address counter reaches an unreadable address (either the address that does not exist or if read protected with the I²C security session closed).
- When the internal address counter reaches an unreadable address, the device releases the SDA line and continues to output FFh.
- There is no roll-over at the end of user memory. On ST25DV64KC, when the internal address counter reaches the end of user memory, the device continues to output bytes located in the Dynamic registers area, until it reaches an unreadable address.

Sequential read in system memory:

- There is no roll over after reaching the end of system memory (the device returns only FFh after the last system memory byte address).

Sequential read in dynamic registers:

- It is possible to read sequentially dynamic register and fast transfer mode's mailbox (contiguous I^2C addresses). There is no roll over at the end of the dynamic registers area.

Sequential read in the mailbox:

- There is no roll over at the end of the mailbox (the device returns only FFh after the last mailbox memory byte address).

6.5.4 Acknowledge in read mode

For all Read commands, the device waits, after each byte read, for an acknowledgment during the ninth bit time. If the bus controller does not drive serial data (SDA) low during this time, the device terminates the data transfer and switches to its Standby mode.

6.6 I²C password management

The device controls I²C security session using an I²C 64-bit password. This I²C password is managed with two I²C dedicated commands: I²C present password and I²C write password.

6.6.1 I²C present password command description

The I²C present password command is used in I²C mode to present the password to the ST25DVxxKC. This is used to open an I²C security session or to allow I²C password modification (see Section 5.6: Data protection for detailed explanation about password usage).

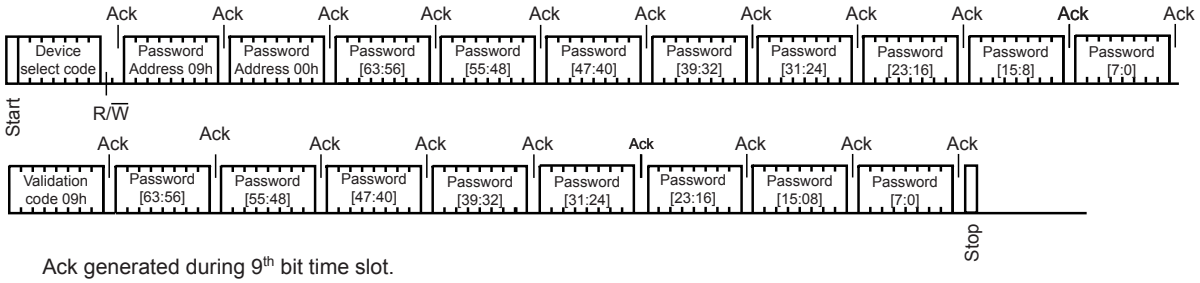
Following a Start condition, the bus controller sends a device select code with the Read/$\overline{\text{Write}}$ bit (R$\overline{\text{W}}$) reset to 0 and the chip enable bit E2 at 1 and E1 at 1. The device acknowledges this, as shown in Figure 36. I²C present password sequence, and waits for two I²C password address bytes, 09h and 00h. The device responds to each address byte with an acknowledge bit, and then waits for the eight password data bytes, the validation code, 09h, and a resend of the eight password data bytes. The most significant byte of the password is sent first, followed by the least significant byte.

It is necessary to send the 64-bit password twice to prevent any data corruption during the sequence. If the two 64-bit passwords sent are not exactly the same, the ST25DVxxKC does not start the internal comparison.

When the bus controller generates a Stop condition, immediately after the Ack bit (during the 10^{th} bit time slot), the ST25DVxxKC compares the 64 received data bits with the 64 bits of the stored I^2C password. If the values match, the I²C security session is open, and the I2C_SSO_Dyn register is set to 01h. If the values do not match, the I²C security session is closed and the I2C_SSO_dyn register is set to 00h.

I2C_SSO_Dyn is a Dynamic register that can be checked via I²C host to know if the I²C security session is open.

Figure 36. I²C present password sequence



Ack generated during 9th bit time slot.

6.6.2 I²C write password command description

The I²C write password command is used to update the I²C password value (register I2C_PWD). It cannot be used to update any of the RF passwords. After the write cycle, the new I²C password value is automatically activated. The I²C password value can only be modified after issuing a valid I²C present password command.

Following a Start condition, the bus controller sends a device select code with the Read/$\overline{\text{Write}}$ bit (R$\overline{\text{W}}$) reset to 0 and the chip enable bit E2 at 1 and E1 at 1. The device acknowledges this, as shown in Figure 37. I²C write password sequence, and waits for the two I²C password address bytes, 09h and 00h. The device responds to each address byte with an acknowledge bit, and then waits for the four password data bytes, the validation code, 07h, and a resend of the eight password data bytes. The most significant byte of the password is sent first, followed by the least significant byte.

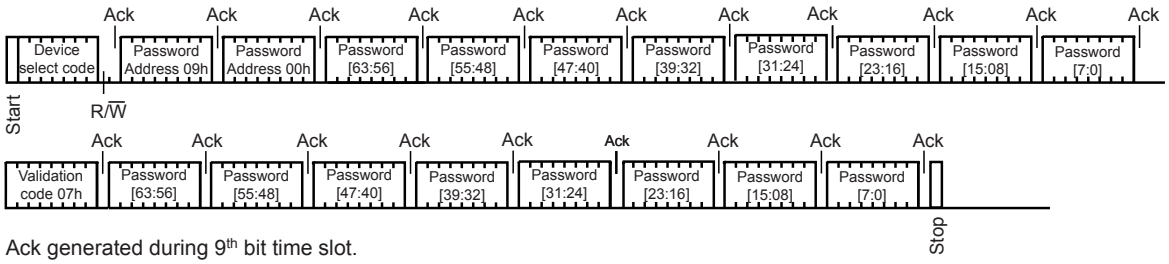
It is necessary to send twice the 64-bit password to prevent any data corruption during the write sequence. If the two 64-bit passwords sent are not exactly the same, the ST25DVxxKC does not modify the I²C password value.

When the bus controller generates a Stop condition immediately after the Ack bit (during the 10th bit time slot), the internal write cycle is triggered. A Stop condition at any other time does not trigger the internal write cycle.

During the internal write cycle, the serial data (SDA) signal is disabled internally, and the device does not respond to any requests.

Caution: *I²C write password command data transits via the 256-byte fast transfer mode's buffer. Consequently fast transfer mode must be deactivated before issuing a write password command, otherwise the command is NotACK (after address LSB), and programming is not done and the device goes in Standby mode.*

Figure 37. I²C write password sequence



Ack generated during 9th bit time slot.

7 RF operation

Contactless exchanges are performed in RF mode as specified by ISO/IEC 15693 or NFC Forum Type 5. The device communicates via the 13.56 MHz carrier electromagnetic wave on which incoming data are demodulated from the received signal amplitude modulation (ASK: amplitude shift keying). The received ASK wave is 10% or 100% modulated with a data rate of 1.6 kbit/s using the 1/256 pulse coding mode, or with a data rate of 26 kbit/s using the 1/4 pulse coding mode.

Outgoing data are generated by the ST25DVxxKC load variation using Manchester coding with one or two subcarrier frequencies at 423 and 484 kHz. Data are transferred from the device at 6.6 kbit/s in low data rate mode, and at 26 kbit/s in high data rate mode. The 53 kbit/s in high data rate mode is supported in one subcarrier frequency at 423 kHz.

The ST25DVxxKC follows ISO/IEC 15693 or NFC Forum Type 5 recommendation for radio-frequency power and signal interface and for anticollision and transmission protocol.

7.1 RF communication

7.1.1 Access to a ISO/IEC 15693 device

The dialog between the “RF reader” and the ST25DVxxKC takes place as follows:

- activation of the ST25DVxxKC by the RF operating field of the reader
- transmission of a command by the reader (ST25DVxxKC detects carrier amplitude modulation)
- transmission of a response by the ST25DVxxKC using load modulation

These operations use the RF power transfer and communication signal interface described below (see Power transfer, Frequency and Operating field). This technique is called RTF (Reader talk first).

Operating field

The ST25DVxxKC operates continuously between the minimum and maximum values of the electromagnetic field H defined in Table 256. RF characteristics. The Reader has to generate a field within these limits.

Power transfer

Power is transferred to the ST25DVxxKC by radio frequency at 13.56 MHz via coupling antennas in the ST25DVxxKC and the Reader. The RF operating field of the reader is transformed on the ST25DVxxKC antenna to an AC voltage which is rectified, filtered and internally regulated. During communications, the amplitude modulation (ASK) on this received signal is demodulated by the ASK demodulator

Frequency

The ISO 15693 standard defines the carrier frequency (f_C) of the operating field as 13.56 MHz ±7 kHz.

7.2 RF communication and energy harvesting

As the current consumption can affect the AC signal delivered by the antenna, RF communications with ST25DVxxKC are not guaranteed during voltage delivery on the energy harvesting analog output V_EH.

7.3 Fast transfer mode mailbox access in RF

Thanks to dedicated commands, the RF interface has the possibility to check Mailbox availability, and the capability to access it directly to put or get a message from it (see Section 5.1: Fast transfer mode (FTM) for specific features).

7.4 RF protocol description

7.4.1 Protocol description

The transmission protocol (or simply “the protocol”) defines the mechanism used to exchange instructions and data between the VCD (Vicinity Coupling Device) and the ST25DVxxKC in both directions. It is based on the concept of “VCD talks first”.

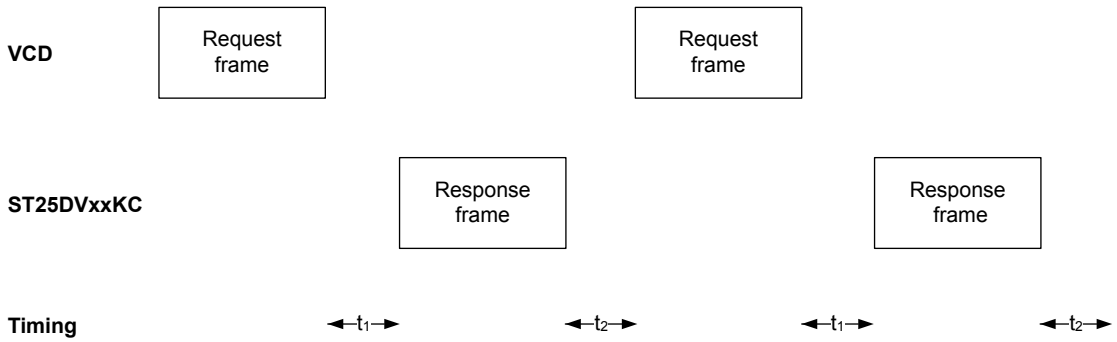
This means that the device does not start transmitting unless it has received and properly decoded an instruction sent by the VCD. The protocol is based on an exchange of a request from the VCD, and a response from the device.

Each request and each response are contained in a frame. The frame is delimited by a Start of Frame (SOF) and End of Frame (EOF).

The protocol is bit-oriented. The number of bits transmitted in a frame is a multiple of eight (8), that is an integer number of bytes.

A single-byte field is transmitted least significant bit (LSBit) first. A multiple-byte field is transmitted least significant byte (LSByte) first, and each byte is transmitted least significant bit (LSBit) first.

Figure 38. Protocol timing



7.4.2 States referring to RF protocol

The device can be in Power-off, Ready, Quiet, or Selected state.

Transitions between these states are specified in Figure 39 and Table 95.

Power-off state

The device is in this state when it does not receive enough energy from the VCD.

Ready state

The device is in this state when it receives enough energy from the VCD. When in the Ready state, it answers any request where the Select_flag is not set.

Quiet state

When in this state, the device answers any request with the Address_flag set, except for Inventory requests.

Selected state

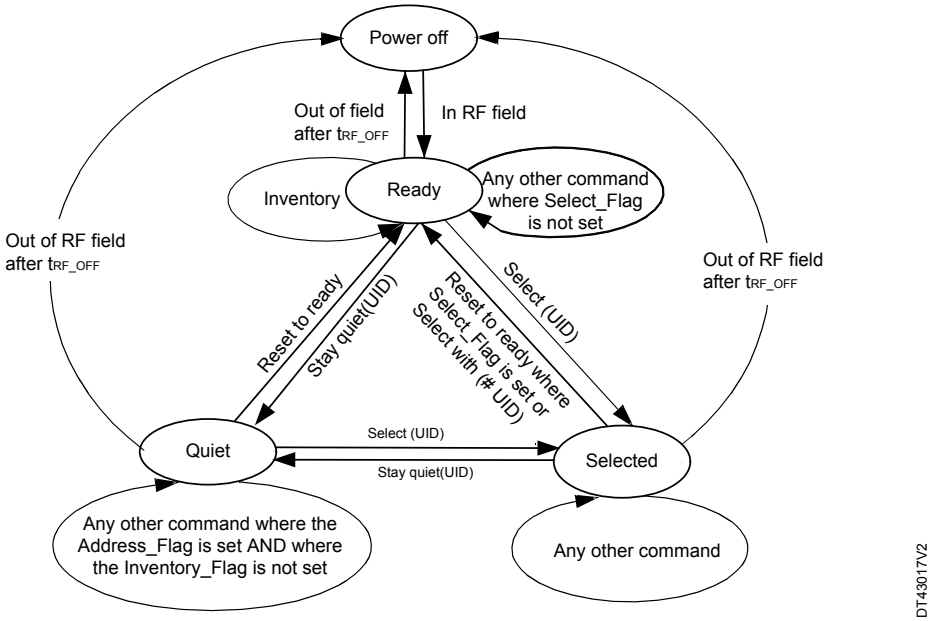
In this state the device answers any request in all modes (see Section 7.4.3: Modes):

- Request in Select mode with the Select_flag set
- Request in Addressed mode if the UID matches
- Request in Non-Addressed mode as it is the mode for general requests

Table 95. Response dependance upon Request_flags

Flags	Address_flag		Select_flag	
	1 Addressed	0 Non addressed	1 Selected	0 Non selected
Device in Ready or Selected state (devices in Quiet state do not answer)	-	X	-	X
Device in Selected state	-	X	X	-
Device in Ready, Quiet, or Selected state (the device matching the UID)	X	-	-	X
Error (03h) or no response (command dependent)	X	-	X	-

Figure 39. State transition diagram



1. *The device returns to the Power off state if the tag is out of the RF field for at least t_{RF_OFF}.*

The intention of the state transition method is that only one device can bein the Selected state at a time.

When the Select_flag is set to 1, the request must not contain a unique ID.

When the Address_flag is set to 0, the request must not contain a unique ID.

7.4.3 Modes

The term “mode” refers to the mechanism used in a request to specify the set of ST25DVxxKC devices that shall execute the request.

Addressed mode

When the Address_flag is set to 1 (Addressed mode), the request contains the Unique ID (UID) of the addressed ST25DVxxKC.

Any ST25DVxxKC that receives a request with the Address_flag set to 1 compares the received Unique ID to its own. If it matches, then the ST25DVxxKC executes the request (if possible) and returns a response to the VCD as specified in the command description.

If the UID does not match, then it remains silent.

Non-addressed mode (general request)

When the Address_flag is cleared to 0 (Non-Addressed mode), the request does not contain a Unique ID.

Select mode

When the Select_flag is set to 1 (Select mode), the request does not contain a unique ID. The ST25DVxxKC in the Selected state that receives a request with the Select_flag set to 1 executes it and returns a response to the VCD as specified in the command description.

Only the ST25DVxxKC in the Selected state answers a request where the Select_flag is set to 1.

The system design ensures that only one ST25DVxxKC can be in the Select state at a time.

7.4.4 Request format

The request consists of:

- an SOF,
- flags,
- a command code,
- parameters and data,
- a CRC,
- an EOF.

Table 96. General request format

SOF	Request_flags	Command code	Parameters	Data	2 bytes CRC	EOF

7.4.5 Request flags

In a request, the “flags” field specifies the actions to be performed by the ST25DVxxKC and whether corresponding fields are present or not.

The flags field consists of eight bits. Bit 3 (Inventory_flag) of the request flag defines the contents of the four MSBs (bits 5 to 8). When bit 3 is reset (0), bits 5 to 8 define the selection criteria. When bit 3 is set (1), bits 5 to 8 define the Inventory parameters.

Table 97. Definition of request flags 1 to 4

Bit	Flag	Level	Description
1	Subcarrier_flag (1)	0	A single subcarrier frequency is used
		1	Two subcarrier frequencies are used
2	Data_rate_flag (2)	0	Low data rate is used
		1	High data rate is used
3	Inventory_flag	0	The meaning of flags 5 to 8 is described in Table 98
		1	The meaning of flags 5 to 8 is described in Table 99
4	Protocol_extension_flag	0	No Protocol format extension
		1	Protocol format extension, reserved for future use

1. *Subcarrier_flag refers to the ST25DVxxKC-to-VCD communication.*
2. *Data_rate_flag refers to the ST25DVxxKC-to-VCD communication.*

Table 98. Request flags 5 to 8 when Inventory_flag (bit 3) = 0

Bit	Flag	Level	Description
5	Select flag (1)	0	The request is executed by any ST25DVxxKC according to the setting of Address_flag
		1	The request is executed only by the ST25DVxxKC in Selected state
6	Address flag	0	The request is not addressed. UID field is not present. The request is executed by all ST25DVxxKCs.
		1	The request is addressed. UID field is present. The request is executed only by the ST25DVxxKC whose UID matches the UID specified in the request.
7	Option flag	0	Option not activated.
		1	Option activated.
8	RFU	0	-

1. *If the Select_flag is set to 1, the Address_flag is set to 0 and the UID field is not present in the request.*

Table 99. Request flags 5 to 8 when Inventory_flag (bit 3) = 1

Bit	Flag	Level	Description
5	AFI flag	0	AFI field is not present
		1	AFI field is present
6	Nb_slots flag	0	16 slots
		1	1 slot
7	Option flag	0	-
8	RFU	0	-

7.4.6 Response format

The response consists of:

- an SOF
- flags
- parameters and data
- a CRC
- an EOF

Table 100. General response format

SOF	Response_flags	Parameters	Data	2 byte CRC	EOF

7.4.7 Response flags

In a response, the flags indicate how actions have been performed by the ST25DVxxKC and whether corresponding fields are present or not. The response flags consist of eight bits.

Table 101. Definitions of response flags 1 to 8

Bit Nb	Flag	Level	Description
Bit 1	Error_flag	0	No error
		1	Error detected. Error code is in the “Error” field.
Bit 2	ResponseBuffer Validity_flag	0	Not supported, always set to 0
Bit 3	Final response_flag	0	Not supported, always set to 0
Bit 4	Extension flag	0	Not supported, always set to 0
Bit 6-5	Block security status length_flag	0	Not supported, always set to 0
Bit 7	Waiting time extension request_flag	0	Not supported, always set to 0
Bit 8	RFU	0	-

7.4.8 Response and error code

If the Error_flag is set by the ST25DVxxKC in the response, the Error code field is present and provides information about the error that occurred.

Error codes not specified in Table 102 are reserved for future use.

Table 102. Response error code definition

Error code	Meaning
01h	Command is not supported.
02h	Command is not recognized (format error).
03h	The option is not supported.
0Fh	Error with no information given.
10h	The specified block is not available.
11h	The specified block is already locked and thus cannot be locked again.
12h	The specified block is locked and its contents cannot be changed.
13h	The specified block was not successfully programmed.
14h	The specified block was not successfully locked.
15h	The specified block is protected in read.

7.5 Timing definition

t_1: response delay

Upon detection of the rising edge of the EOF received from the VCD, the ST25DVxxKC waits for a t_{1nom} time before transmitting its response to a VCD request or switching to the next slot during an inventory process. Values of t_1 are given in Table 103.

t_2: VCD new request delay

t_2 is the time after which the VCD may send an EOF to switch to the next slot when one or more ST25DVxxKC responses have been received during an Inventory command. It starts from the reception of the EOF from the ST25DVxxKCs.

The EOF sent by the VCD may be either 10% or 100% modulated regardless of the modulation index used for transmitting the VCD request to the ST25DVxxKC.

t_2 is also the time after which the VCD may send a new request to the ST25DVxxKC, as described in Figure 38.

Values of t_2 are given in Table 103.

t_3: VCD new request delay when no response is received from the device

t_3 is the time after which the VCD may send an EOF to switch to the next slot when no response has been received.

The EOF sent by the VCD may be either 10% or 100% modulated regardless of the modulation index used for transmitting the VCD request to the device.

From the time the VCD has generated the rising edge of an EOF:

- If this EOF is 100% modulated, the VCD waits for a time at least equal to t_{3min} for 100% modulation before sending a new EOF.
- If this EOF is 10% modulated, the VCD waits for a time at least equal to t_{3min} for 10% modulation before sending a new EOF.

Table 103. Timing values

	Minimum (min) values		Nominal (nom) values	Maximum (max) values
	100% modulation	10% modulation		
t_1	4320 / f_c = 318.6 µs		4352 / f_c = 320.9 µs	4384 / f_c = 323.3 µs (1)
t_2	4192 / f_c = 309.2 µs		No t_{nom}	No t_{max}
t_3	t_{1max} (2) + t_{SOF} (3)	t_{1max} (2) + t_{NRT} (4) + t_{2min}	No t_{nom}	No t_{max}

1. *VCD request is not interpreted during the first milliseconds following the RF field rising.*
2. *t_{1max} does not apply for write-alike requests. Timing conditions for write-alike requests are defined in the command description.*
3. *t_{SOF} is the time taken by the ST25DVxxKC to transmit an SOF to the VCD. t_{SOF} depends on the current data rate: High data rate or Low data rate.*
4. *t_{NRT} is the nominal response time of the device, it depends upon the response data rate, subcarrier modulation mode, and size of the expected response frame.*

Note: The tolerance of specific timings is ± 32 / f_C.

7.6 RF commands

7.6.1 RF command code list

The device supports the following legacy and extended RF command set:

- **Inventory**, used to perform the anticollision sequence.
- **Stay Quiet**, used to put the ST25DVxxKC in quiet mode, where it does not respond to any inventory command.
- **Select**, used to select the ST25DVxxKC. After this command, the ST25DVxxKC processes all Read/Write commands with Select_flag set.
- **Reset To Ready**, used to put the ST25DVxxKC in the ready state.
- **Read Single Block** and **Extended Read Single Block**, used to read the 32 bits of the selected block and its locking status.
- **Write Single Block** and **Extended Write Single Block**, used to write and verify the new content for an update of a 32 bit block, provided that it is not in a locked memory area.
- **Read Multiple Blocks** and **Extended Read Multiple Blocks**, used to read the selected blocks, and send back their values.
- **Write Multiple Blocks** and **Extended Write Multiple Blocks**, used to write and verify the new content for an update of up to four blocks located in the same memory area, which was not previously locked for writing.
- **Write AFI**, used to write the 8-bit value in the AFI register.
- **Lock AFI**, used to lock the AFI register.
- **Write DSFID**, used to write the 8-bit value in the DSFID register.
- **Lock DSFID**, used to lock the DSFID register.
- **Get System information** and **Extended Get System Information**, used to provide the system information value.
- **Get Multiple Blocks Security Status** and **Extended Get Multiple Blocks Security Status**, used to provide security status of memory blocks.
- **Write Password**, used to update the 64-bit of the selected areas or configuration password, but only after presenting the current one.
- **Lock Block** and **Extended Lock block**, used to write the security status bits of the first two blocks of the user memory.
- **Present Password**, enables the user to present a password to open a security session.
- **Fast Read Single Block** and **Fast Extended Read Single Block**, used to read the 32 bits of the selected block and its locking status at double data rate.
- **Fast Read Multiple Blocks** and **Fast Extended Read Multiple Blocks**, used to read the selected blocks and send back their value at double data rate.
- **Read Message**, used to read up to 256 byte of the Mailbox.
- **Read Message Length**, used to read the Mailbox message length.
- **Fast Read Message**, used to read up to 256 byte of the mailbox, at double data rate.
- **Write Message**, used to write up to 256 byte in the Mailbox.
- **Fast Read Message Length**, used to read the mailbox length, at double data rate.
- **Fast Write Message**, used to write up to 256 bytes in the mailbox, with answer at double data rate.
- **Read Configuration**, used to read static configuration registers.
- **Write Configuration**, used to write static configuration registers.
- **Read Dynamic Configuration,** used to read dynamic register.
- **Write Dynamic Configuration ,** used to write dynamic register.
- **Fast Read Dynamic Configuration**, used to read dynamic register, at double data rate.
- **Fast Write Dynamic Configuration**, used to write dynamic register, with answer at double data rate.
- **Manage GPO**, used to drive GPO output value when corresponding GPO mode is enabled.

7.6.2 Command codes list

The ST25DVxxKC supports the commands described in this section. Their codes are given in Table 104.

Table 104. Command codes

Command code standard	Function	Command code custom	Function
01h	Inventory	A0h	Read Configuration
02h	Stay Quiet	A1h	Write Configuration
20h	Read Single Block	A9h	Manage GPO
21h	Write Single Block	AAh	Write Message
22h	Lock Block	ABh	Read Message Length
23h	Read Multiple Blocks	ACh	Read Message
24h	Write Multiple Blocks	ADh	Read Dynamic Configuration
25h	Select	AEh	Write Dynamic Configuration
26h	Reset to Ready	B1h	Write Password
27h	Write AFI	B3h	Present Password
28h	Lock AFI	C0h	Fast Read Single Block
29h	Write DSFID	C3h	Fast Read Multiple Blocks
2Ah	Lock DSFID	CDh	Fast Read Dynamic Configuration
2Bh	Get System Info	CEh	Fast Write Dynamic Configuration
2Ch	Get Multiple Blocks Security Status	-	-
30h	Extended Read Single Block	C4h	Fast Extended Read Single Block
31h	Extended Write Single Block	C5h	Fast Extended Read Multiple Blocks
32h	Extended Lock Block	CAh	Fast Write Message
33h	Extended Read Multiple Blocks	CBh	Fast Read Message Length
34h	Extended Write Multiple Blocks	CCh	Fast Read Message
3Bh	Extended Get System Info	-	-
3Ch	Extended Get Multiple Blocks Security Status	-	-

7.6.3 General command rules

In case of a valid command, the following paragraphs describe the expected behaviour for each command.

But in case of an invalid command, in a general manner, the ST25DVxxKC behaves as follows:

1. If flag usage is incorrect, the error code 03h is issued only if the right UID is used in the command, otherwise no response is issued.
2. The error code 02h is issued if the custom command is used with the manufacturer code different from the ST one.

Another case is if I^2C is busy. In this case, any RF command (except Inventory, Select, Stay quiet and Reset to ready) gets 0Fh error code as response only:

- If select flag and address flags are not set at the same time (except if ST25DVxxKC is in quiet state)
- If select flag is set and ST25DVxxKC is in selected state.

For all other commands, if I^2C is busy, no response is issued by ST25DVxxKC.

7.6.4 Inventory

Upon receiving the Inventory request, the device runs the anticollision sequence. The Inventory_flag is set to 1. The meaning of flags 5 to 8 is shown in Table 99.

The request contains:

- the flags
- the Inventory command code (01)
- the AFI if the AFI flag is set
- the mask length
- the mask value if mask length is different from 0
- the CRC

The device does not generate any answer in case of error.

Table 105. Inventory request format

Request SOF	Request_flags	Inventory	Optional AFI	Mask length	Mask value	CRC16	Request EOF
-	8 bits	01h	8 bits	8 bits	0 - 64 bits	16 bits	-

The response contains:

- the flags
- the Unique ID

Table 106. Inventory response format

Response SOF	Response_flags	DSFID	UID	CRC16	Response EOF
-	8 bits	8 bits	64 bits	16 bits	-

During an Inventory process, if the VCD does not receive an RF response, it waits for a time t_3 before sending an EOF to switch to the next slot. t_3 starts from the rising edge of the request EOF sent by the VCD.

- If the VCD sends a 100% modulated EOF, the minimum value of t_3 is:
  $t_3min = 4384 / f_C$ (323.3 µs) + t_{SOF}
- If the VCD sends a 10% modulated EOF, the minimum value of t_3 is:
  $t_3min = 4384 / f_C$ (323.3 µs) + t_{NRT} + t_{2min}

where:

- t_{SOF} is the time required by the device to transmit an SOF to the VCD,
- t_{NRT} is the nominal response time of the device.

t_{NRT} and t_{SOF} are dependent upon the device to VCD data rate and subcarrier modulation mode.

Note: In case of error, no response is sent by the device.

7.6.5 Stay Quiet

On receiving this command, the device enters the Quiet state if no error occurs, and does not send back a response. There is no response to the Stay Quiet command even if an error occurs.

The Option_flag is not supported. The Inventory_flag must be set to 0.

When in the Quiet state the device does not process any request if the Inventory_flag is set, and processes any Addressed request.

The device exits the Quiet state when:

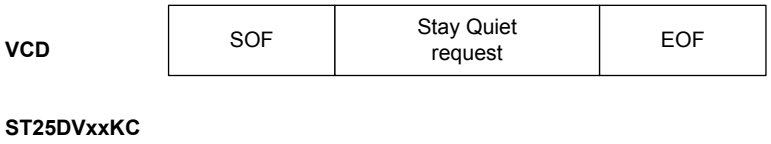
- it goes out of the RF field (goes to Power off state)
- it receives a Select request (goes to Selected state)
- it receives a Reset to Ready request (goes to the Ready state).

Table 107. **Stay Quiet request format**

Request SOF	Request flags	Stay Quiet	UID	CRC16	Request EOF
-	8 bits	02h	64 bits	16 bits	-

The Stay Quiet command must always be executed in Addressed mode (Select_flag is reset to 0 and Address_flag is set to 1).

Figure 40. **Stay Quiet frame exchange**



7.6.6 Read Single Block

On receiving this command, the device reads the requested block and sends back its 32-bit value in the response. The Option_flag is supported, when set response include the Block Security Status. The Inventory_flag must be set to 0.

Block number is coded on 1 byte and only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed using this command.

Table 108. **Read Single Block request format**

Request SOF	Request_flags	Read Single Block	UID (1)	Block number	CRC16	Request EOF
-	8 bits	20h	64 bits	8 bits	16 bits	-

1. *This the field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number

Table 109. **Read Single Block response format when Error_flag is not set**

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits	32 bits	16 bits	-

1. *This field is optional.*

Response parameters:

- Block security status if Option_flag is set (see Table 110)
- Four bytes of block data

Table 110. **Block security status**

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use. All at 0.							0: Current block not locked 1: Current block locked

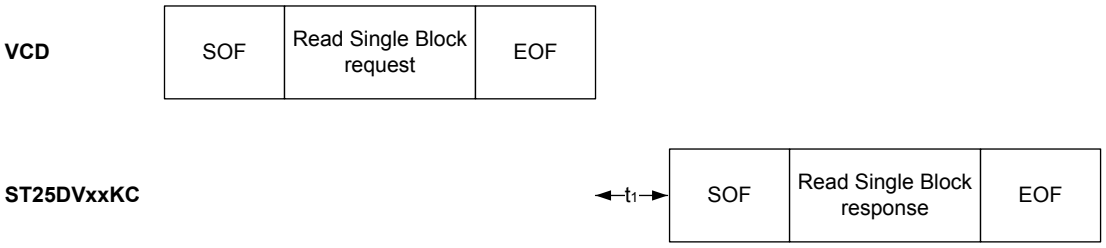
Table 111. Read Single Block response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option not supported
  - 0Fh: error with no information
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 41. Read Single Block frame exchange



7.6.7 Extended Read Single Block

On receiving this command, the device reads the requested block and sends back its 32-bit value in the response.

When the Option_flag is set, the response includes the Block Security Status.

Block number is coded on 2 bytes, hence all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 112. Extended Read Single Block request format

Request SOF	Request_flags	Extended Read Single Block	UID (1)	Block number	CRC16	Request EOF
-	8 bits	30h	64 bits	16 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number (LSB first)

Table 113. Extended Read Single Block response format when Error_flag is not set

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits	32 bits	16 bits	-

1. *This field is optional.*

Response parameters:

- Block security status if Option_flag is set (see Table 114)
- Four bytes of block data

Table 114. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use. All at 0.							0: Current block not locked 1: Current block locked

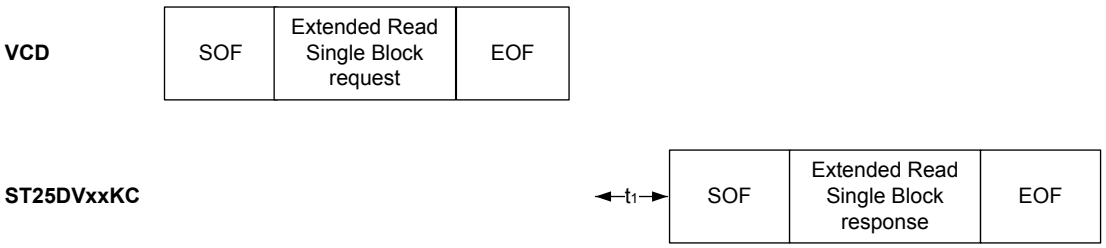
Table 115. Extended Read Single Block response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option not supported or no response
  - 0Fh: error with no information
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 42. Extended Read Single Block frame exchange



7.6.8 Write Single Block

When it receives this command, the device writes the data contained in the request to the targeted block and reports whether the write operation was successful in the response. When the Option_flag is set, the device waits for an isolated EOF to respond. The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not program correctly the data into the memory. The W_t time is equal to t_{1nom} + N × 302 µs (N is an integer).

Block number is coded on 1 byte, and only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 116. Write Single Block request format

Request SOF	Request_flags	Write Single Block	UID (1)	Block number	Data	CRC16	Request EOF
-	8 bits	21h	64 bits	8 bits	32 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number
- Data

Table 117. Write Single Block response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the writing cycle.

Table 118. Write Single Block response format when Error_flag is set

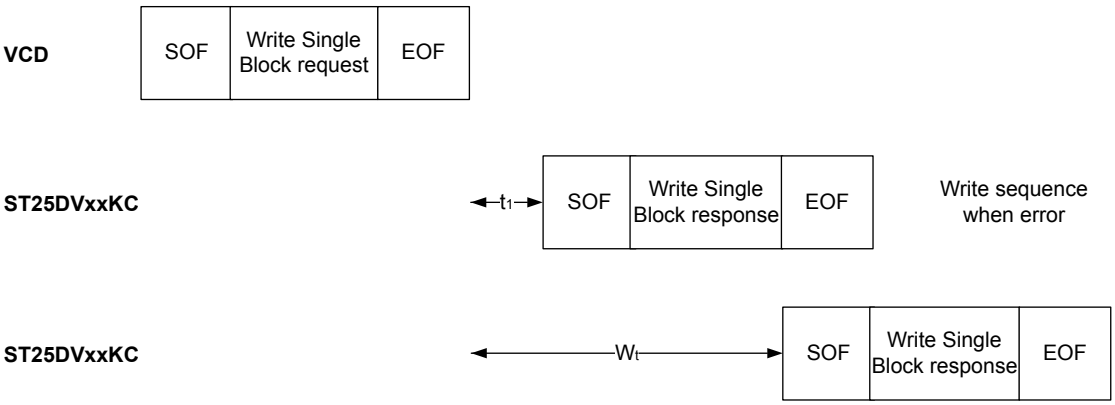
Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 12h: the specified block is locked or protected and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Note: For more details, see Figure 9. Memory organization.

Figure 43. Write Single Block frame exchange



7.6.9 Extended Write Single Block

When the device receives this command, it writes the data contained in the request to the targeted block, and reports whether the write operation was successful in the response. When the Option_flag is set, it waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise data may be not correctly programmed into the memory. The W_t time is equal to t_{1nom} + N × 302 µs (N is an integer).

Block number is coded on two bytes, all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 119. Extended Write Single Block request format

Request SOF	Request_flags	Extended Write Single Block	UID (1)	Block number	Data	CRC16	Request EOF
-	8 bits	31h	64 bits	16 bits	32 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number (LSB first)
- Data

Table 120. Extended Write Single Block response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the writing cycle.

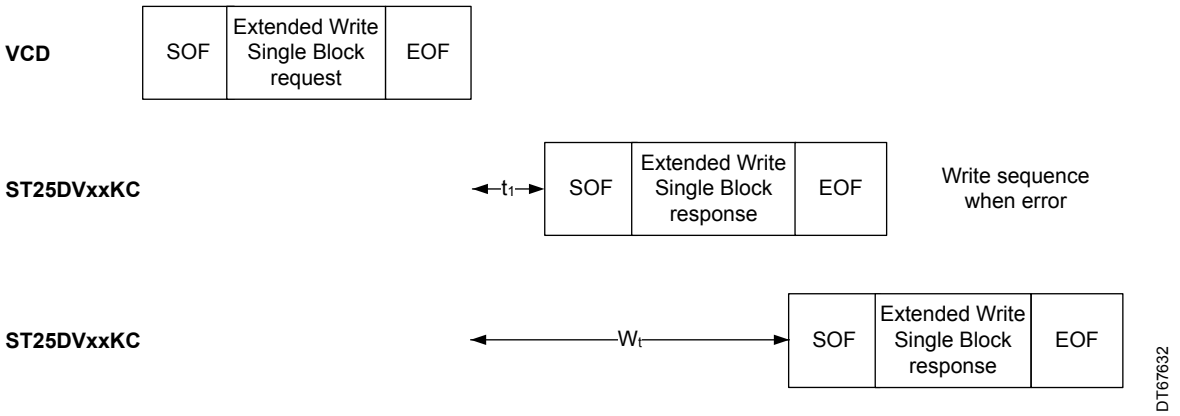
Table 121. Extended Write Single Block response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 12h: the specified block is locked and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Figure 44. **Extended Write Single Block frame exchange**



7.6.10 Lock Block

When the device receives this command, it locks the single block value permanently and protects its content against new writings.

This command applies only to blocks 0 and 1, which may include a CC file.

The Option_flag is supported, when set the device waits for an isolated EOF.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not lock correctly the block value in memory. The W_t time is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 122. **Lock block request format**

Request SOF	Request_flags	Lock block	UID (1)	block number	CRC16	Request EOF
-	8 bits	22h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- Only block numbers 0 and 1 are allowed

Table 123. **Lock block response format when Error_flag is not set**

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

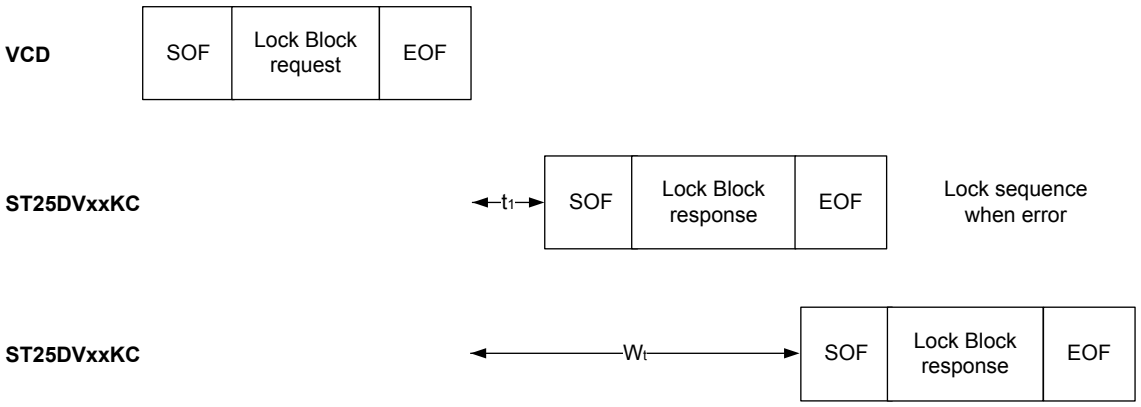
Table 124. **Lock block response format when Error_flag is set**

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option not supported
  - 10h: block not available
  - 11h: the specified block is already locked and thus cannot be locked again
  - 14h: the specified block was not successfully locked

Figure 45. Lock Block frame exchange



7.6.11 Extended Lock Block

When receives this command, the device locks a single block value permanently, and protects its content against new writings.

This command applies only to blocks 0 and 1, which may include a CC file.

When the Option_flag is set, the device waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not lock correctly the single block value in memory. The W_t time is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 125. Extended Lock Block request format

Request SOF	Request_flags	Extended Lock block	UID (1)	block number	CRC16	Request EOF
-	8 bits	32h	64 bits	16 bits	16 bits	-

1. *The field is optional.*

Request parameter:

- Request flags
- UID (optional)
- Only block numbers 0 and 1 are allowed

Table 126. Extended Lock Block response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

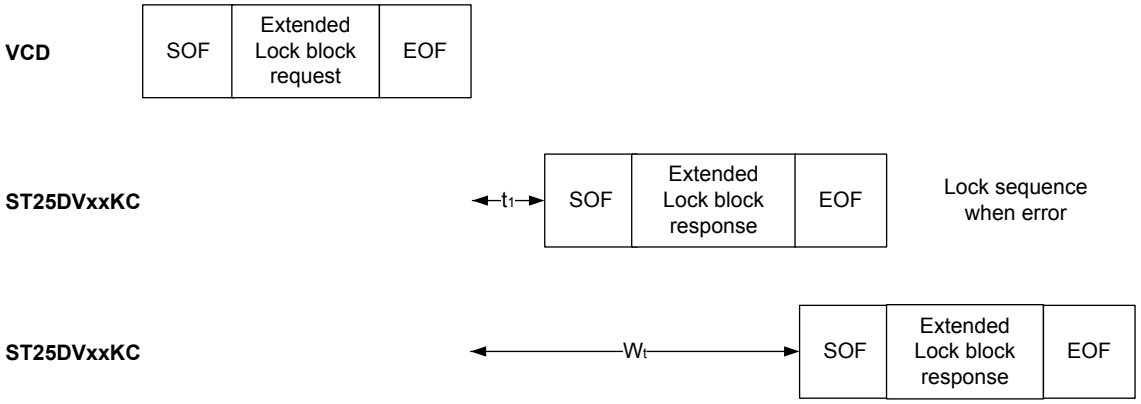
Table 127. **Extended Lock Block response format when Error_flag is set**

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option not supported
  - 10h: block not available
  - 11h: the specified block is already locked and cannot be locked again
  - 14h: the specified block was not successfully locked

Figure 46. **Extended Lock Block frame exchange**



7.6.12 Read Multiple Blocks

When receiving this command, the device reads the selected blocks and sends back their value in multiples of 32 bits in the response. The blocks are numbered from 00h to FFh in the request and the value is minus one (–1) in the field. For example, if the “Number of blocks” field contains the value 06h, seven blocks are read. The maximum number of blocks is fixed at 256. Read Multiple Blocks command can cross areas borders, and returns all blocks until reaching a non readable block (block read protected or out of memory). When the Option_flag is set, the response returns the Block Security Status.

The Inventory_flag must be set to 0.

Block number is coded on 1 byte, and only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 128. **Read Multiple Blocks request format**

Request SOF	Request_flags	Read Multiple Block	UID (1)	First block number	Number of blocks	CRC16	Request EOF
-	8 bits	23h	64 bits	8 bits	8 bits	16 bits	-

1. *The field is optional.*

Request parameters:

- Request flags
- UID (optional)
- First block number
- Number of blocks

Table 129. Read Multiple Blocks response format when Error_flag is not set

Response SOF	Response_ flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits (2)	32 bits(2)	16 bits	-

1. *The field is optional.*
2. *Repeated as needed.*

Response parameters:

- Block security status if Option_flag is set (see Table 130)
- N blocks of data

Table 130. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use. All at 0.							0: Current block not locked 1: Current block locked

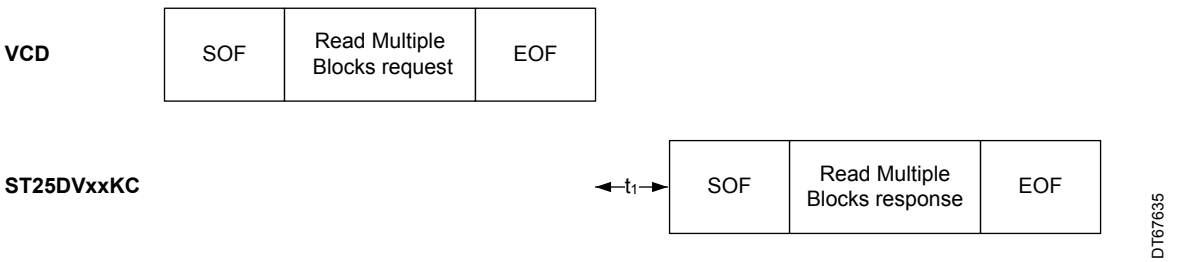
Table 131. Read Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 47. Read Multiple Blocks frame exchange



7.6.13 Extended Read Multiple Blocks

When receives this command, the device reads the selected blocks and sends back their value in multiples of 32 bits in the response. The blocks are numbered from 00h to last block of memory in the request, and the value is minus one (-1) in the field. For example, if the “Number of blocks” field contains the value 06h, seven blocks are read. The maximum number of blocks is fixed at 2047. This command can cross areas borders, and returns all blocks until reaching a non readable block (block read protected or out of memory). When the Option_flag is set, the response returns the Block Security Status.

The Inventory_flag must be set to 0.

Block number is coded on two bytees, hence all memory blocks of ST25DV16KC and ST25DV64KC can be addressed..

Table 132. Extended Read Multiple Blocks request format

Request SOF	Request_flags	Extended Read Multiple Block	UID (1)	First block number	Number of blocks	CRC16	Request EOF
-	8 bits	33h	64 bits	16 bits	16 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- First block number (LSB first)
- Number of blocks (LSB first)

Table 133. Extended Read Multiple Blocks response format when Error_flag is not set

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits (2)	32 bits (2)	16 bits	-

1. *This field is optional.*
2. *Repeated as needed.*

Response parameters:

- Block security status if Option_flag is set (see Table 130)
- N blocks of data

Table 134. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use. All at 0							0: Current block not locked 1: Current block locked

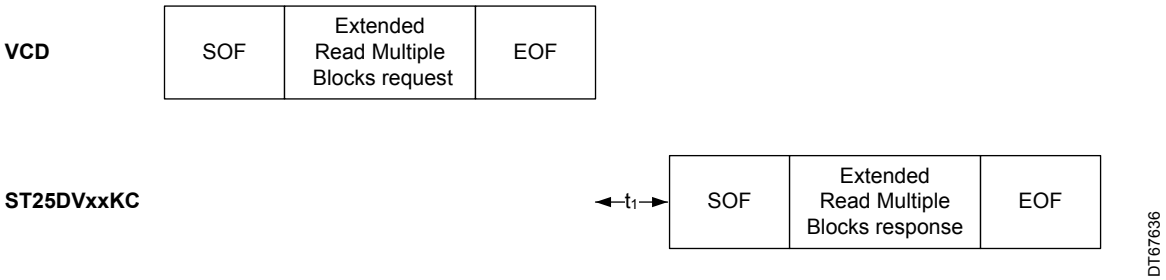
Table 135. Extended Read Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 48. **Extended Read Multiple Blocks frame exchange**



7.6.14 Write Multiple Blocks

When it receives this command, the device writes the data contained in the request to the requested blocks, and reports whether the write operation is successful in the response. The command supports up to four blocks, data field must be coherent with the number of blocks to program.

The number of blocks in the request is one less than the number of blocks that the device must write (for instance, Number of block = 2 means that 3 blocks must be written).

If some blocks overlap, or overlap end of user memory, the device returns an error code and none of the blocks are programmed. When the Option_flag is set, the device waits for an isolated EOF to respond. During the RF write cycle Wt, there must be no modulation, otherwise the device may not program correctly the data into the memory. W_t is equal to t_{1nom} + m × 302 µs < 20 ms (m is an integer, function of the number of blocks to program).

The Inventory_flag must be set to 0.

Block number is coded on one byte, only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 136. **Write Multiple Blocks request format**

Request SOF	Request_flags	Write Multiple Blocks	UID (1)	First block number	Number of blocks (2)	Data	CRC16	Request EOF
-	8 bits	24h	64 bits	8 bits	8 bits	32 bits (3)	16 bits	-

1. *This field is optional.*
2. *The number of blocks in the request is the number of blocks that the VICC must write minus 1.*
3. *Repeated as needed.*

Request parameters:

- Request flags
- UID (optional)
- First Block number
- Number of blocks
- Data

Table 137. **Write Multiple Blocks response format when Error_flag is not set**

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the writing cycle.

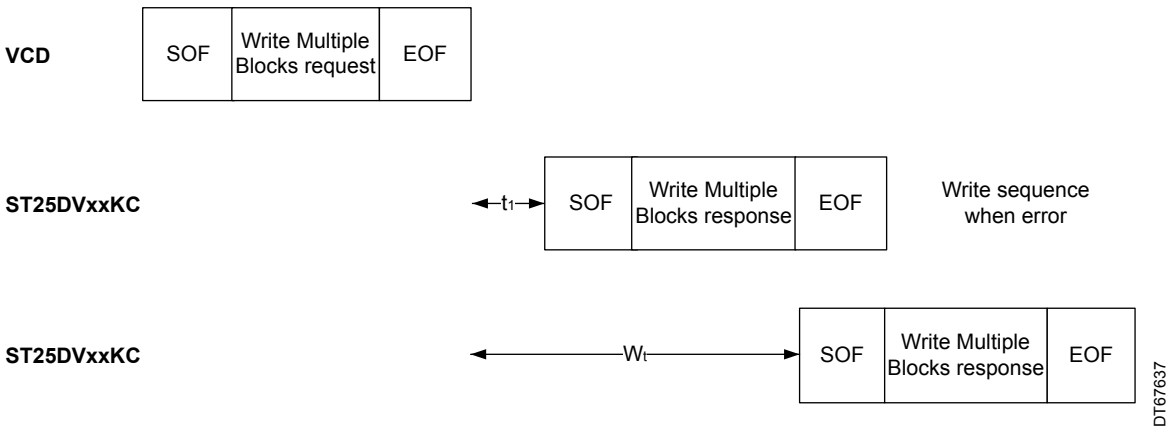
Table 138. Write Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 12h: the specified block is locked and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Figure 49. Write Multiple Blocks frame exchange



7.6.15 Extended Write Multiple Blocks

When it receives this command, the device writes the data contained in the request to the targeted blocks and reports whether the write operation were successful in the response. Up to four blocks are supported, data field must be coherent with number of blocks to program.

If some blocks overlaps areas, or overlap end of user memory the device returns an error code and none of the blocks are programmed.

The number of blocks in the request is one less than the number of blocks to be written (as an example, Number of block = 2 means 3 blocks to be written).

When the Option_flag is set, the device waits for an isolated EOF to respond. During the RF write cycle, there must be no modulation, otherwise the device may not program correctly the data into the memory. W_t is equal to t_{1nom} + m × 302 µs < 20 ms (m is an integer function of the number of blocks to program).

The inventory_flag must be set to 0.

Block number is coded on two bytes, all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 139. Extended Write Multiple Block request format

Request SOF	Request_flags	Extended Write Multiple Blocks	UID (1)	First block number	Number of blocks (2)	Data	CRC16	Request EOF
-	8 bits	34h	64 bits	16 bits	16 bits	32 bits(3)	16 bits	-

1. *This field is optional.*
2. *The number of blocks in the request is one less than the number of blocks that the VICC must write.*
3. *Repeated as needed.*

Request parameters:

- Request flags
- UID (optional)
- First block number (LSB first)
- Number of block (LSB first)
- Data (from first to last block)

Table 140. Extended Write Multiple Blocks response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the writing cycle.

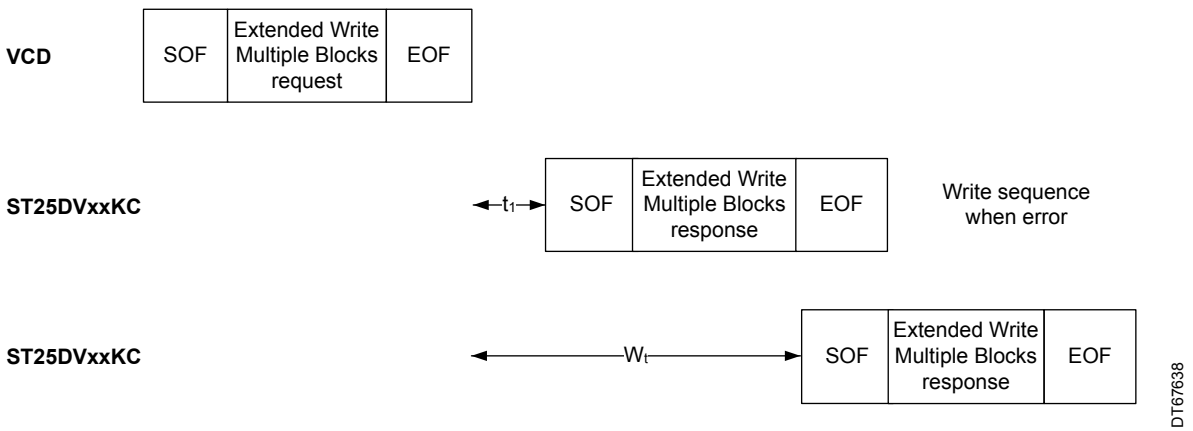
Table 141. Extended Write Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 12h: the specified block is locked and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Figure 50. Extended Write Multiple Blocks frame exchange



7.6.16 Select

When receiving the Select command:

- If the UID is equal to its own UID, the device enters or stays in the Selected state and sends a response.
- If the UID does not match its own UID, the selected device returns to the Ready state and does not send a response.

The device answers with an error code only if the UID is equal to its own UID. If not, no response is generated. If an error occurs, the device remains in its current state.

The Option_flag is not supported, and the Inventory_flag must be set to 0.

Table 142. Select request format

Request SOF	Request_flags	Select	UID	CRC16	Request EOF
-	8 bits	25h	64 bits	16 bits	-

Request parameter:

- UID

Table 143. Select response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

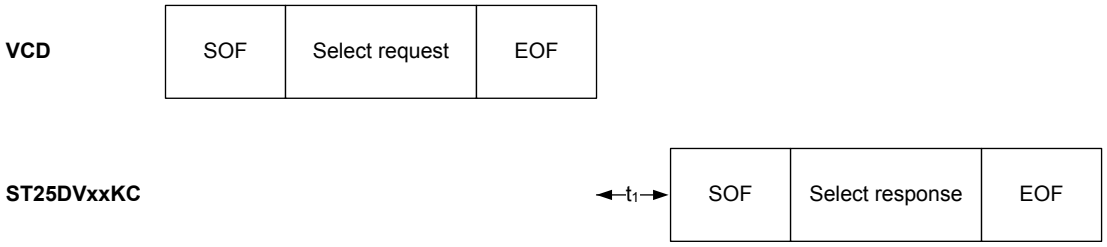
Table 144. Select response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: the option is not supported
  - 0Fh: error with no information given

Figure 51. Select frame exchange



7.6.17 Reset to Ready

When it receives this command, the device returns to the Ready state if no error occurs. In the Addressed mode, the device answers an error code only if the UID is equal to its own UID. If not, no response is generated.

The Option_flag is not supported, and the Inventory_flag must be set to 0.

Table 145. Reset to Ready request format

Request SOF	Request_flags	Reset to Ready	UID (1)	CRC16	Request EOF
-	8 bits	26h	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- UID (optional)

Table 146. Reset to Ready response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

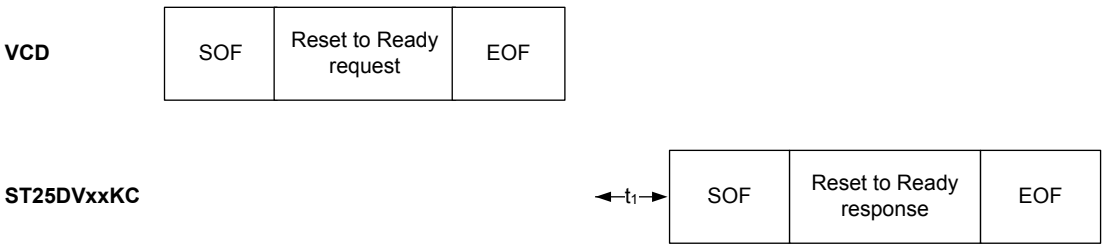
Table 147. Reset to Ready response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: the option is not supported
  - 0Fh: error with no information given

Figure 52. Reset to Ready frame exchange



7.6.18 Write AFI

When it receives this request, the device programs the 8-bit AFI value to its memory. When the Option_flag is set, it waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not write correctly the AFI value into the memory. W_t is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 148. Write AFI request format

Request SOF	Request_flags	Write AFI	UID (1)	AFI	CRC16	Request EOF
-	8 bits	27h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- AFI

Table 149. Write AFI response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

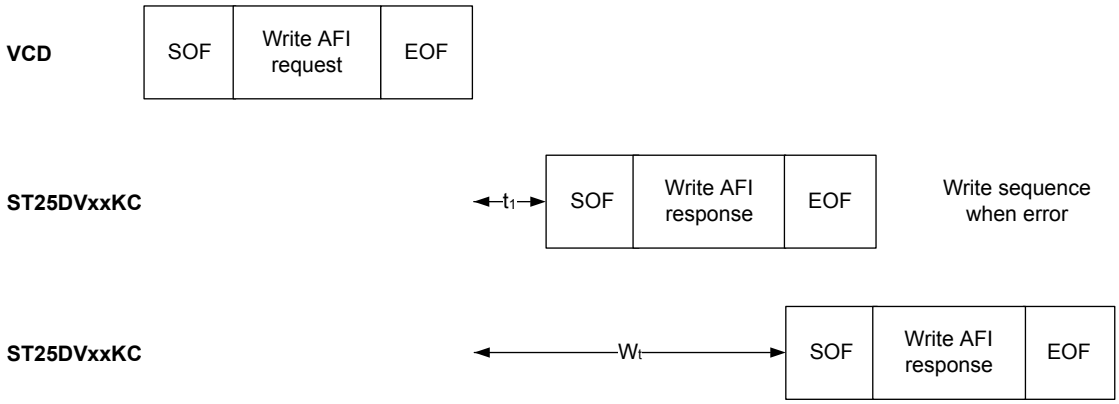
Table 150. Write AFI response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 12h: the specified block is locked and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Figure 53. Write AFI frame exchange



7.6.19 Lock AFI

When it receives this request, the device locks the AFI value permanently. When the Option_flag is set, it waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not write correctly the AFI value into the memory. W_t is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 151. Lock AFI request format

Request SOF	Request_flags	Lock AFI	UID (1)	CRC16	Request EOF
-	8 bits	28h	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request Flags
- UID (optional)

Table 152. Lock AFI response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

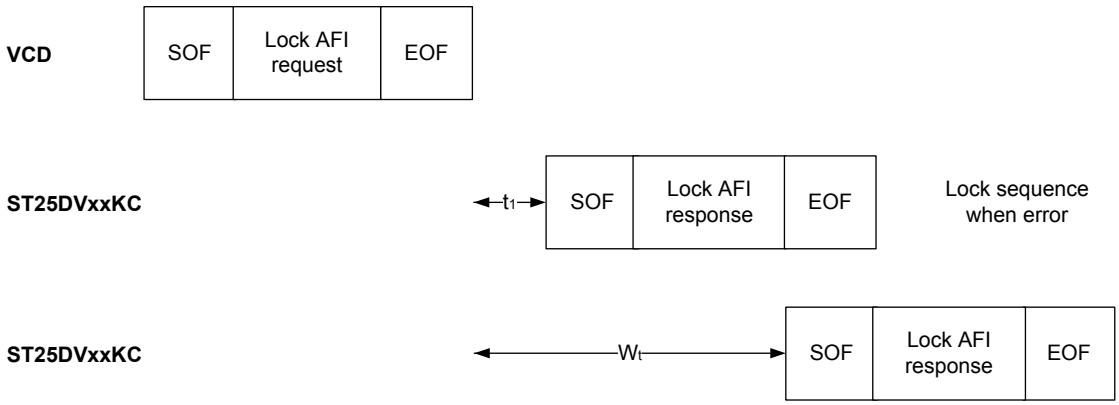
Table 153. Lock AFI response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 11h: the specified block is already locked and thus cannot be locked again
  - 14h: the specified block was not successfully locked

Figure 54. Lock AFI frame exchange



7.6.20 Write DSFID

When it receives this request, the device programs the 8-bit DSFID value to its memory. When the Option_flag is set, it waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not write correctly the AFI value into the memory. W_t is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 154. Write DSFID request format

Request SOF	Request_flags	Write DSFID	UID (1)	DSFID	CRC16	Request EOF
-	8 bits	29h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- DSFID

Table 155. Write DSFID response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter

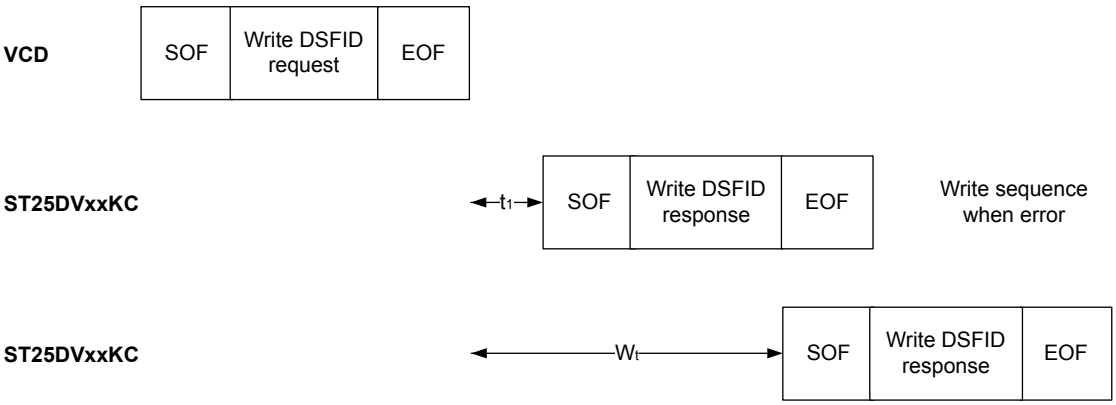
Table 156. Write DSFID response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 12h: the specified block is locked and its contents cannot be changed
  - 13h: the specified block was not successfully programmed

Figure 55. Write DSFID frame exchange



7.6.21 Lock DSFID

When it receives this request, the device locks the DSFID value permanently. When the Option_flag is set, it waits for an isolated EOF to respond.

The Inventory_flag must be set to 0.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not write correctly the AFI value into the memory. W_t is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 157. Lock DSFID request format

Request SOF	Request_flags	Lock DSFID	UID (1)	CRC16	Request EOF
-	8 bits	2Ah	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)

Table 158. Lock DSFID response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter.

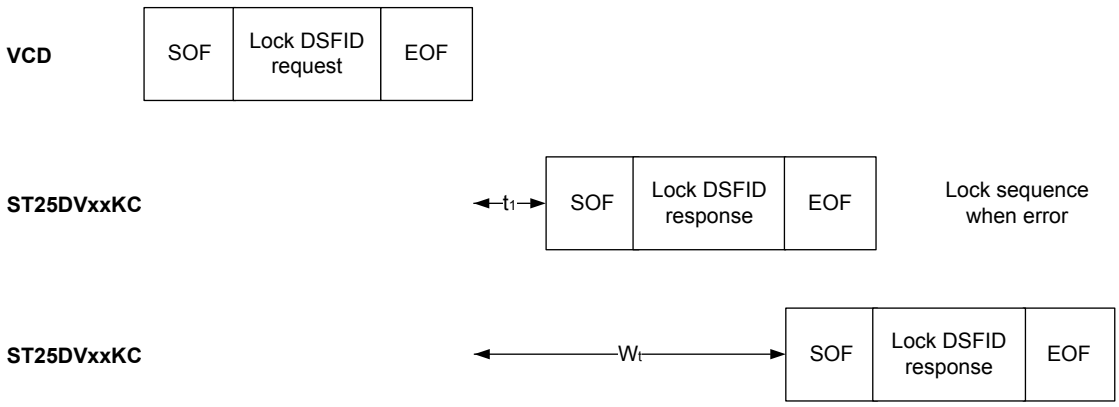
Table 159. Lock DSFID response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 11h: the specified block is already locked and thus cannot be locked again
  - 14h: the specified block was not successfully locked

Figure 56. Lock DSFID frame exchange



7.6.22 Get System Info

When it receives this command, the device sends back its information data in the response.

The Option_flag is not supported. The Inventory_flag must be set to 0. This command can be issued in both Addressed and Non Addressed modes.

Table 160. Get System Info request format

Request SOF	Request_flags	Get System Info	UID (1)	CRC16	Request EOF
-	8 bits	2Bh	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)

Table 161. Get System Info response format when Error_flag is not set

Device	Response SOF	Response flags	Information flags	UID	DSFID	AFI	Mem. Size	IC ref.	CRC16	Response EOF
ST25DV64KC ST25DV16KC	-	00h	0Bh	64 bits	8 bits	8 bits	NA (1)	51h	16 bits	-
ST25DV04KC			0Fh				037Fh	50h		

1. *Field not present in this configuration*

Response parameters:

- Information flags set to 0Bh/0Fh. DSFID, AFI and IC reference fields are present.
- UID code on 64 bits
- DSFID value
- AFI value
- MemSize: block size in bytes and memory size in number of blocks (only for ST25DV04KC configurations)

Table 162. Memory size

MSB				LSB	
16	14	13	9	8	1
RFU		Block size in bytes		Number of blocks	
0h		03h		7Fh	

- ST25DVxxKC IC reference: the 8 bits are significant.

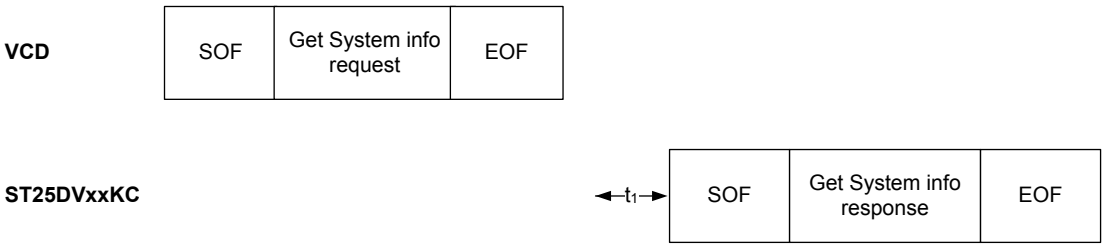
Table 163. Get System Info response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	01h	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: Option not supported
  - 0Fh: error with no information given

Figure 57. **Get System Info frame exchange**



7.6.23 Extended Get System Info

When it receives this command, the device sends back its information data in the response.

The Option_flag is not supported. The Inventory_flag must be set to 0. The command can be issued in both Addressed and Non-Addressed modes.

Table 164. **Extended Get System Info request format**

Request SOF	Request_flags	Extended Get System Info	Parameter request field	UID (1)	CRC16	Request EOF
-	8 bits	3Bh	8 bits	64 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)

Table 165. **Parameter request list**

Bit	Flag name	Value	Description
b1	DSFID	0	No request of DSFID
		1	Request of DSFID
b2	AFI	0	No request of AFI
		1	Request of AFI
b3	VICC memory size	0	No request of data field on VICC memory size
		1	Request of data field on VICC memory size
b4	IC reference	0	No request of information on IC reference
		1	Request of information on IC reference
b5	MOI	1	Information on MOI always returned in response flag
b6	VICC command list	0	No request of Data field of all supported commands
		1	Request of Data field of all supported commands
b7	CSI Information	0	No request of CSI list
		1	Request of CSI list
b8	Extended Get System Info parameter Field	0	One byte length of the Extended Get System Info parameter field

Table 166. Extended Get System Info response format when Error_flag is not set

Response SOF	Response_flags	Information flags	UID	DSFID (1) (2)	AFI (1) (2)	Other fields (1) (2)	CRC16	Response EOF
-	00h	8 bits(1)	64 bits	8 bits	8 bits	Up to 64 bits (3)	16 bits	-

1. *See Table 167.*
2. *This field is optional.*
3. *The number of bytes is function of the selected parameter list.*

Response parameters:

- Information flag defining which fields are present
- UID code on 64 bits
- DSFID value (if requested in Parameters request field)
- AFI value (if requested in Parameters request field)
- Other fields:
  - VICC memory size (if requested in Parameters request field)
  - ICRef (if requested in Parameters request field)
  - VICC command list (if requested in Parameters request field)

Table 167. Response information flag

Bit	Meaning if bit is set	Comment	
b1	DSFID	0	DSFID field is not present
		1	DSFID field is present
b2	AFI	0	AFI field is not present
		1	AFI field is present
b3	VICC memory size	0	Data field on VICC memory size is not present.
		1	Data field on VICC memory size is present.
b4	IC reference	0	Information on IC reference field is not present.
		1	Information on IC reference field is present
b5	MOI	0	1 byte addressing
		1	2-byte addressing
b6	VICC command list	0	Data field of all supported commands is not present
		1	Data field of all supported commands is present
b7	CSI information	0	CSI list is not present
b8	Info flag field	0	One byte length of Info flag field

Table 168. Response other fields: VICC memory size

MSB		LSB
24 22	21 17	16 01
RFU	Block size in bytes	Number of blocks
0h	03h	007Fh (ST25DV04KC) 01FFh (ST25DV16KC) 07FFh (ST25DV64KC)

Table 169. Response other fields: ICRef

1 byte
ICRef
50h (ST25DV04KC)
51h (ST25DV16KC)
51h (ST25DV64KC)

Table 170. Response other fields: VICC command list

MSB							LSB
32	25	24	17	16	09	08	01
Byte 4		Byte 3		Byte 2		Byte 1	
00h		3Fh		3Fh		FFh	

Table 171. Response other fields: VICC command list - Byte 1

Bit	Meaning if bit is set	Comment
b1	Read Single Block is supported	-
b2	Write Single Block is supported	-
b3	Lock Single Block is supported	-
b4	Read Multiple Blocks is supported	-
b5	Write Multiple Blocks is supported	-
b6	Select is supported	Including Select state
b7	Reset to Ready is supported	-
b8	Get Multiple Blocks Security Status is supported	-

Table 172. Response other fields: VICC command list - Byte 2

Bit	Meaning if bit is set	Comment
b1	Write AFI is supported	-
b2	Lock AFI is supported	-
b3	Write DSFID is supported	-
b4	Lock DSFID is supported	-
b5	Get System Information is supported	-
b6	Custom commands are supported	-
b7	RFU	0 shall be returned
b8	RFU	0 shall be returned

Table 173. Response other fields: VICC command list - Byte 3

Bit	Meaning if bit is set	Comment
b1	Extended Read Single Block is supported	-
b2	Extended Write Single Block is supported	-
b3	Extended Lock Single Block is supported	-
b4	Extended Read Multiple Blocks is supported	-
b5	Extended Write Multiple Blocks is supported	-
b6	Extended Get Multiple Blocks Security Status is supported	-
b7	RFU	0 must be returned
b8	RFU	0 must be returned

Table 174. Response other fields: VICC command list - Byte 4

Bit	Meaning if bit is set	Comment
b1	Read Buffer is supported	Response Buffer is supported
b2	Select Secure State is supported	VCD or Mutual authentication are supported
b3	Final Response always includes crypto result	Flag b3 is set in the Final response
b4	AuthComm crypto format is supported	-
b5	SecureComm crypto format is supported	-
b6	KeyUpdate is supported	-
b7	Challenge is supported	-
b8	If set to 1, an additional byte is transmitted	0 must be returned

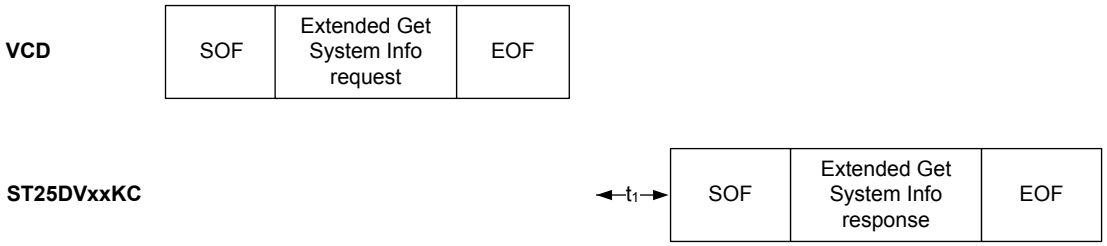
Table 175. Extended Get System Info response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	01h	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: Option not supported
  - 0Fh: error with no information given

Figure 58. Extended Get System Info frame exchange



7.6.24 Get Multiple Blocks Security Status

When it receives this command, the devicesends back its security status for each address block: 0 when the block is writable, 1 when it is locked for writing. The security status is defined by the area security status (and by LCK_CCFILE register for blocks 0 and 1). The blocks are numbered from 00h up to the maximum memory block number in the request, and the value is minus one (–1) in the field. For example, a value of “06”, in the “Number of blocks” field requests returns the security status of seven blocks. This command does not send an error if number of blocks overlap areas or overlap the end of the user memory.

The number of blocks is coded on 1 byte, only first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 176. Get Multiple Blocks Security Status request format

Request SOF	Request_flags	Get Multiple Blocks Security Status	UID (1)	First block number	Number of blocks	CRC16	Request EOF
-	8 bits	2Ch	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- First block number
- Number of blocks

Table 177. Get Multiple Blocks Security Status response format when Error_flag is not set

Response SOF	Response_flags	Block security status	CRC16	Response EOF
-	8 bits	8 bits (1)	16 bits	-

1. *Repeated as needed.*

Response parameters:

- Block security status

Table 178. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use All at 0							0: Current block not locked 1: Current block locked

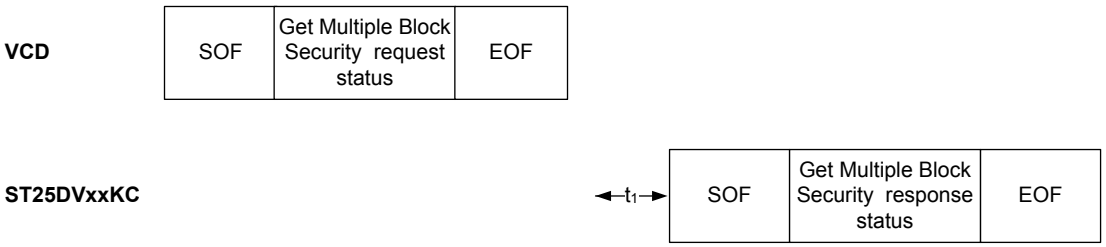
Table 179. Get Multiple Blocks Security Status response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: the option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available

Figure 59. **Get Multiple Blocks Security Status frame exchange**



7.6.25 Extended Get Multiple Blocks Security Status

When receiving this command, the device sends back the security status for each address block: 0 when the block is writable, 1 when it is locked for writing. The block security status is defined by the area security status. The blocks are numbered from 00h up to the maximum memory block number in the request, and the value is minus one (–1) in the field. For example, a value of '06' in the “Number of blocks” field requests to return the security status of seven blocks.

This command does not send an error if number of blocks overlap areas or overlap the end of the user memory.

The number of blocks is coded on two bytes, all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 180. **Extended Get Multiple Blocks Security Status request format**

Request SOF	Request_flags	Extended Get Multiple Blocks Security Status	UID(1)	First block number	Number of blocks	CRC16	Request EOF
-	8 bits	3Ch	64 bits	16 bits	16 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- First block number (LSB first)
- Number of blocks (LSB first)

Table 181. **Extended Get Multiple Blocks Security Status response format when Error_flags is not set**

Response SOF	Response_flags	Block security status	CRC16	Response EOF
-	8 bits	8 bits (1)	16 bits	-

1. *Repeated as needed.*

Response parameters:

- Block security status

Table 182. **Block security status**

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use All at 0							0: Current block not locked 1: Current block locked

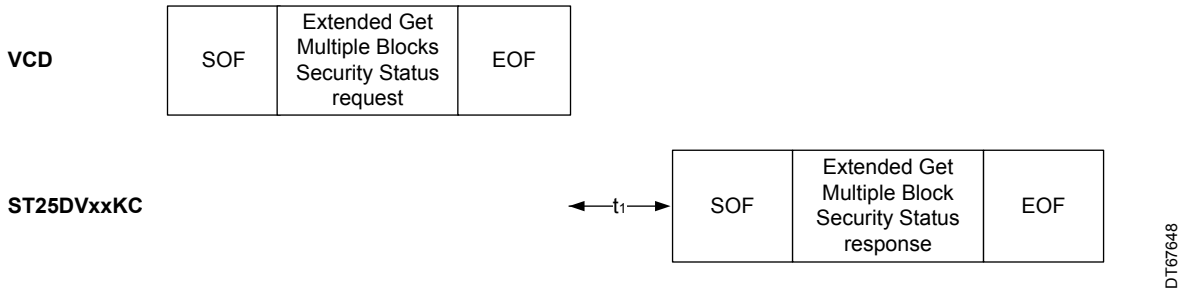
Table 183. Extended Get Multiple Blocks Security Status response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 03h: the option is not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available

Figure 60. Extended Get Multiple Blocks Security Status frame exchange



7.6.26 Read Configuration

When it receives this command, the device reads the static system configuration register at the Pointer address and sends back its 8-bit value in the response.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 184. Read Configuration request format

Request SOF	Request_flags	Read Configuration	IC Mfg code	UID (1)	Pointer	CRC16	Request EOF
-	8 bits	A0h	02h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Note: Refer to Table 13. System configuration memory map for details on register addresses.

Request parameters:

- System configuration register pointer
- UID (optional)

Table 185. Read Configuration response format when Error_flag is not set

Response SOF	Response_flags	Register value	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameters:

- One byte of data: system configuration register

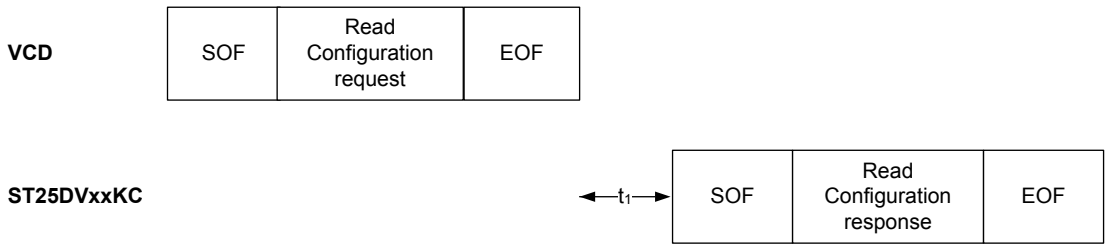
Table 186. Read Configuration response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set
  - 02h: command not recognized
  - 03h: the option is not supported
  - 10h: block not available
  - 0Fh: error with no information given

Figure 61. Read Configuration frame exchange



7.6.27 Write Configuration

This command is used to write static system configuration register. It must be preceded by a valid presentation of the RF configuration password to open the RF configuration security session.

When it receives the command, the device writes the data contained in the request to the system configuration register at the Pointer address, and reports whether the write operation was successful in the response or not.

When the Option_flag is set, the device waits for an isolated EOF to respond. The Inventory_flag is not supported.

During the RF write cycle W_t, there must be no modulation, otherwise the device may not program correctly the data into the Configuration byte. W_t is equal to t_{1nom} + N × 302 µs (N is an integer).

Table 187. Write Configuration request format

Request SOF	Request_flags	Write Configuration	IC Mfg code	UID (1)	Pointer	Register value(2)	CRC16	Request EOF
-	8 bits	A1h	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*
2. *Before updating the register value, check the meaning of each bit in previous sections.*

Request parameters:

- Request flags
- Register pointer
- Register value
- UID (optional)

Table 188. Write Configuration response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Note: Refer to Table 13. System configuration memory map for details on register addresses.

Response parameter:

- No parameter. The response is sent back after the writing cycle.

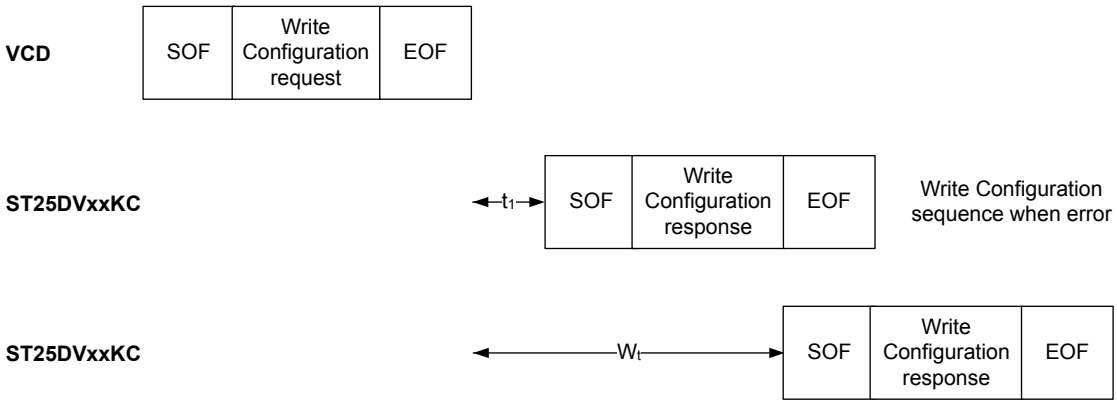
Table 189. Write configuration response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option is not supported
  - 0Fh: error with no information given
  - 10h: block not available
  - 12h: block already locked, content cannot change
  - 13h: the specified block was not successfully programmed

Figure 62. **Write Configuration exchange**



7.6.28 Read Dynamic Configuration

When it receives this command, the device reads the Dynamic register address indicated by the pointer and sends back its 8-bit value in the response.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 190. **Read Dynamic Configuration request format**

Request SOF	Request_flags	Read Dynamic Configuration	IC Mfg code	UID (1)	Pointer address	CRC16	Request EOF
-	8 bits	ADh	02h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- UID (optional)

Table 191. **Read Dynamic Configuration response format when Error_flag is not set**

Response SOF	Response_flags	Data	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameters:

- One byte of data

Note: Refer to Table 13. System configuration memory mapfor details on register addresses.

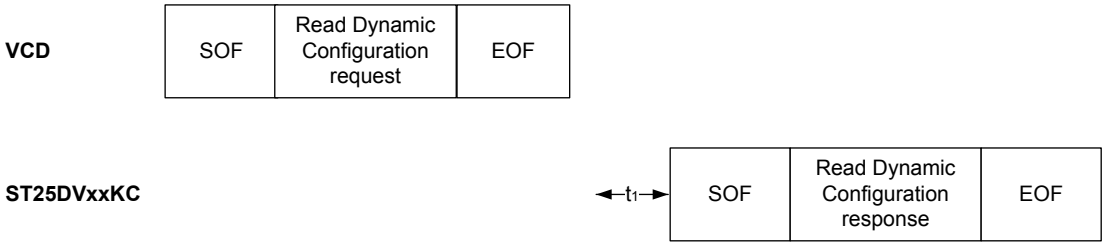
Table 192. **Read Dynamic Configuration response format when Error_flag is set**

Response SOF	Response_flags	error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error given with no information
  - 10h: block not available

Figure 63. **Read Dynamic Configuration frame exchange**



7.6.29 Write Dynamic Configuration

When it receives this command, the device updates the Dynamic register addressed by the pointer. The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 193. **Write Dynamic Configuration request format**

Request SOF	Request_flags	Write Dynamic Configuration	IC Mfg code	UID (1)	Pointer address	Register value	CRC16	Request EOF
-	8 bits	AEh	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Pointer address
- Register value

Table 194. **Write Dynamic Configuration response format when Error_flag is not set**

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameters:

- No parameter. The response is sent back after t_1.

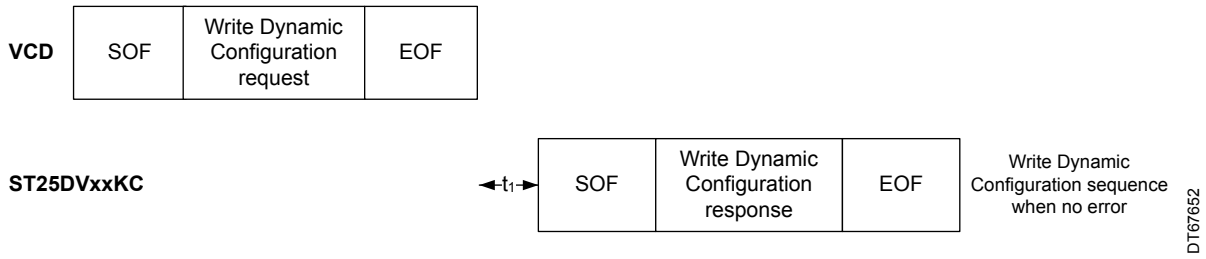
Table 195. **Write Dynamic Configuration response format when Error_flag is set**

Response SOF	Response_flags	error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: block not available

Figure 64. **Write Dynamic Configuration frame exchange**



7.6.30 Manage GPO

Depending upon the command argument, the device forces the GPO output level if RF_USER interrupt is enabled, or sends a pulse on GPO output if RF_INTERRUPT is enabled. If neither RF_USER nor RF_INTERRUPT are enabled, the command is not executed, and the device responds with an error code 0F.

The pulse duration is defined by IT_TIME (bits 4 to 2) of GPO2 static register, and occurs just after the command response.

For the 12-pin and 10-ball packages version (CMOS GPO output):

- Set means that the GPO pin is driven to a high level (V_{DCG})
- Reset pulls the GPO pin to a low level (V_{SS})
- The IT corresponds to a transmission of a positive pulse on the GPO pin

For the 8-pin package version (open drain GPO output):

- Set means that the GPO pin is driven to a low level (V_{SS})
- Reset releases the GPO (high impedance), thanks to an external pull-up, the high level is recovered
- IT corresponds to the GPO pin driven to ground during the IT duration, then pin is released

Option_flag is not supported. The Inventory_flag must be set to 0.

Table 196. **Manage GPO request format**

Request SOF	Request_ flags	Manage GPO	IC Mfg code	UID (1)	GPO VAL(2)	CRC16	Request EOF
-	8 bits	A9h	02h	64 bits	8 bits	16 bits	-

1. *This field is optional.*
2. *See Table 197. GPOVAL*

Table 197. **GPOVAL**

GPOVAL	IT	GPO pin output
0xxxxxx0b	RF_USER enabled	Pin pull to low level (open drain version), pin driven to high level (CMOS version).
0xxxxxx1b	RF_USER enabled	Pin released (open drain version), pin driven to low level (CMOS version).
1xxxxxxb	RF_INTERRUPT enabled	GPO pin pulled to low level during IT time, then released (open drain version). Pin driven to high level during IT time, then to low level (CMOS version).
Any other condition		GPO pin released (open drain version), or driven to low level (CMOS version).

Request parameters:

- Request flag
- UID (optional)
- Data: Define static or dynamic Interrupt

Table 198. Manage GPO response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the write cycle.

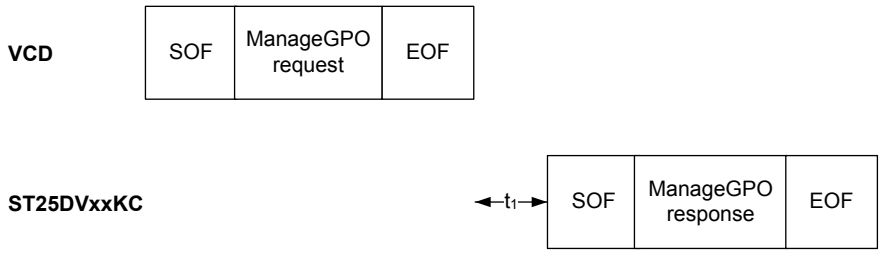
Table 199. ManageGPO response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 13h: the specified block was not successfully programmed (this error is generated if the ManageGPO GPOVAL value is not in line with the GPO interrupts setting as specified in Table 197)

Figure 65. Manage GPO frame exchange



7.6.31 Write Message

When it receives this command, the device puts the data contained in the request into the Mailbox buffer, updates the MB_LEN_Dyn register, and sets bit RF_PUT_MSG in MB_CTRL_Dyn register. It then reports if the write operation was successful in the response. The mailbox contains up to 256 data bytes, which are filled from the first location 00h. MSGlength parameter of the command is the number of Data bytes minus - 1 (00h for 1 byte of data, FFh for 256 bytes of data). The command can be executed only when the mailbox is accessible by RF (fast transfer mode is enabled, previous RF message was read or time-out occurs, no I^2C message to be read). User can check it by reading b1 of MB_CTRL_Dyn “HOST_PUT_MSG” which must be reset to 0. The Option_flag is not supported (refer to Section 5.1).

Table 200. Write Message request format

Request SOF	Request_ flags	Write Message	IC Mfg code	UID (1)	MSGLength	Message Data	CRC16	Request EOF
-	8 bits	AAh	02h	64 bits	1 byte	(MSGLength + 1) bytes	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Message Length
- Message Data

Table 201. Write Message response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the write cycle.

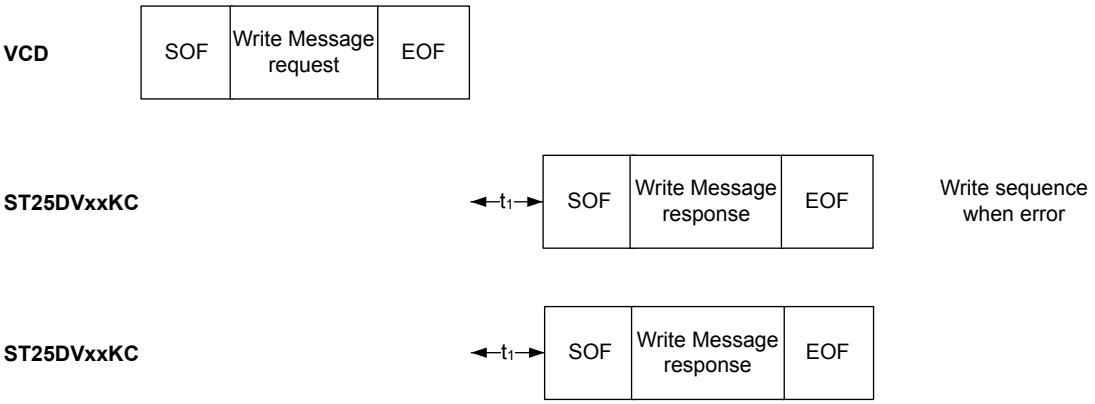
Table 202. Write Message response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given

Figure 66. Write Message frame exchange



7.6.32 Read Message Length

When it receives this command, the device reads the MB_LEN_Dyn register, which contains the Mailbox message length, and sends back its 8-bit value in the response.

The value of MB_LEN_Dyn returned is the size of the message length in bytes - 1.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 203. Read Message Length request format

Request SOF	Request_flags	Read Message Length	IC Mfg code	UID (1)	CRC16	Request EOF
-	8 bits	ABh	02h	64 bits	16 bits	-

1. *The field is optional.*

Request parameters:

- UID (optional)

Table 204. Read Message Length response format when Error_flag is not set

Response SOF	Response_flags	Data	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameters:

- One byte of data: MB_LEN_Dyn register value

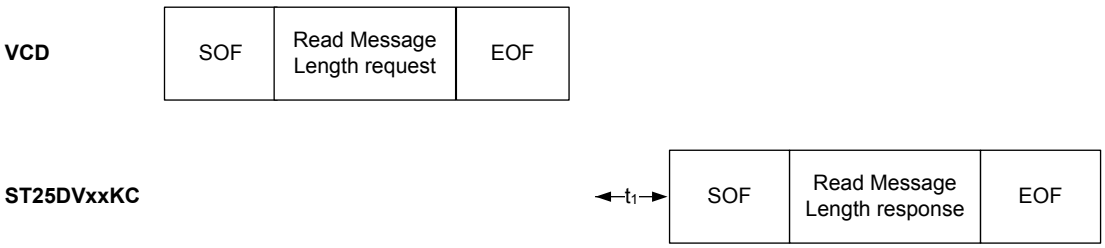
Table 205. Read Message Length response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error given with no information

Figure 67. Read Message Length frame exchange



7.6.33 Read Message

When it receives this command, the device reads up to 256 bytes in the Mailbox from the location specified by MBpointer, and sends back their value in the response. First MailBox location is '00'. When Number of bytes is set to 00h and MBPointer is 00h, the MB_LEN bytes of the full message are returned. Otherwise, Read Message command returns (Number of Bytes + 1) bytes (01h returns 2 bytes, FFh returns 256 bytes).

An error is reported if (Pointer + Nb of bytes + 1) is greater than the message length. RF Reading of the last byte of the mailbox message automatically clears b1 of MB_CTRL_Dyn “HOST_PUT_MSG”, and allows RF to put a new message.

The Option_flag is not supported. The Inventory_flag must be set to 0.

Table 206. Read Message request format

Request SOF	Request_flags	Read Message	IC Mfg code	UID (1)	MBpointer	Number of Bytes	CRC16	Request EOF
-	8 bits	ACh	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)
- Pointer (start at 00h)
- Number of bytes is one less then the requested data

Table 207. Read Message response format when Error_flag is not set

Response SOF	Response_flags	Mailbox content	CRC16	Response EOF
-	8 bits	(Number of bytes + 1) bytes (1)	16 bits	-

1. *Number of message bytes when Number of bytes is set to 00h.*

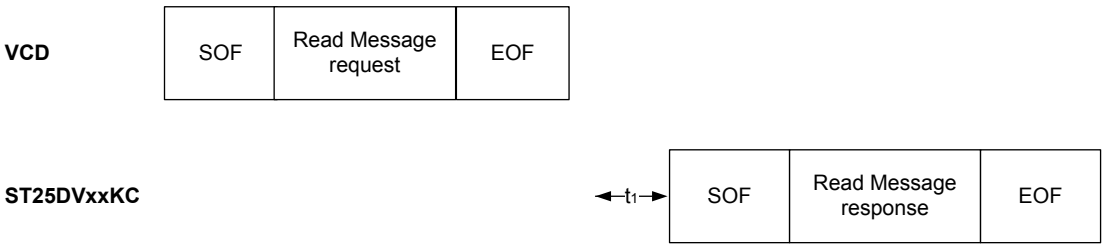
Response parameters:

- (number of data + 1) data bytes

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given

Figure 68. Read Message frame exchange



7.6.34 Fast Read Message

On receiving the Fast Read Message command, the ST25DVxxKC reads up to 256 byte in the Mailbox from the location specified by MBpointer and sends back their value in the response. First MailBox location is '00'. When Number of bytes is set to 00h and MBPointer is equals to 00h, the MB_LEN bytes of the full message are returned. Otherwise, Fast Read Message command returns (Number of Bytes + 1) bytes (i.e. 01h returns 2 bytes, FFh returns 256 bytes).

An error is reported if (Pointer + Nb of bytes + 1) is greater than the message length..

RF Reading of the last byte of mailbox message automatically clears b1 of MB_CTRL_Dyn “HOST_PUT_MSG” and allows RF to put a new message.

The data rate of the response is multiplied by 2 compated to Read Message.

The subcarrier_flag should be set to 0, otherwise the ST25DVxxKC answers with an error code. The Option_flag is not supported, and the Inventory_flag must be set to 0.

Table 208. Fast Read Message request format

Request SOF	Request_flags	Fast Read Message	IC Mfg code	UID(1)	MBpointer	Number of Bytes	CRC16	Request EOF
-	8 bits	CCh	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional*

Request parameters:

- Request flag
- UID (Optional)
- Pointer (start at 00h)
- Number of bytes is one less than the requested data

Table 209. Fast Read Message response format when Error_flag is NOT set

Response SOF	Response_flags	Mailbox content	CRC16	Response EOF
-	8 bits	(Number of bytes + 1) bytes(1)	16 bits	64 bits

1. *Number of message Bytes when Number of Bytes is set to 00h*

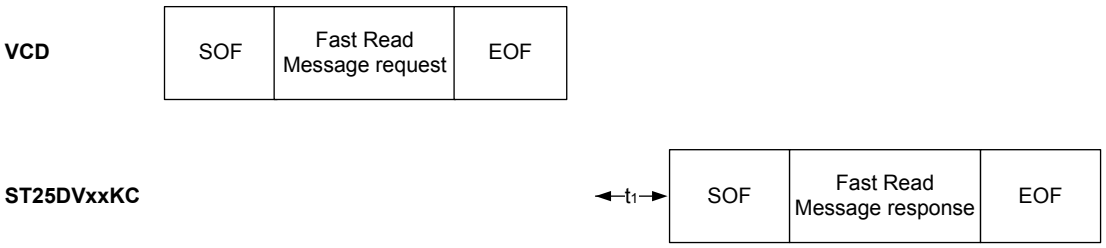
Response parameters:

- (number of bytes + 1) data bytes

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given

Figure 69. Fast Read Message frame exchange



7.6.35 Write Password

When it receives this command, the device uses the data contained in the request to write the password and reports whether the operation was successful in the response. It is possible to modify a Password value only after issuing a valid Present Password command (of the same password number). When the Option_flag is set, the device waits for an isolated EOF to respond. Refer to Section 5.6: Data protection for details on password management. The Inventory_flag must be set to 0.

During the RF write cycle time, W_t, there must be no modulation, otherwise the device may not correctly program the data into the memory.

W_t is equal to t_{1nom} + N × 302 µs (N is an integer). After a successful write, the new value of the selected password is automatically activated. It is not required to present the new password value until power-down.

Caution: *If the device is powered through V_{CC}, removing it during Write Password command can abort the command. As a consequence, before writing a new password, RF user must check if V_{CC} is ON, by reading EH_CTRL_Dyn register bit 3 (VCC_ON), and eventually ask host to maintain or to shut down V_{CC}, during the Write Password command, to avoid password corruption.*

To make the application more robust, it is recommended to use addressed or selected mode during write password operations, to get the traceability of programmed tags/UID.

Table 210. Write Password request format

Request SOF	Request_flags	Write password	IC Mfg code	UID (1)	Password number	Data	CRC16	Request EOF
-	8 bits	B1h	02h	64 bits	8 bits	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- Password number:
  - 00h = RF configuration password RF_PWD_0,
  - 01h = RF_PWD_1,
  - 02h = RF_PWD_2,
  - 03h = RF_PWD_3,
  - other = Error
- Data

Table 211. Write Password response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- no parameter.

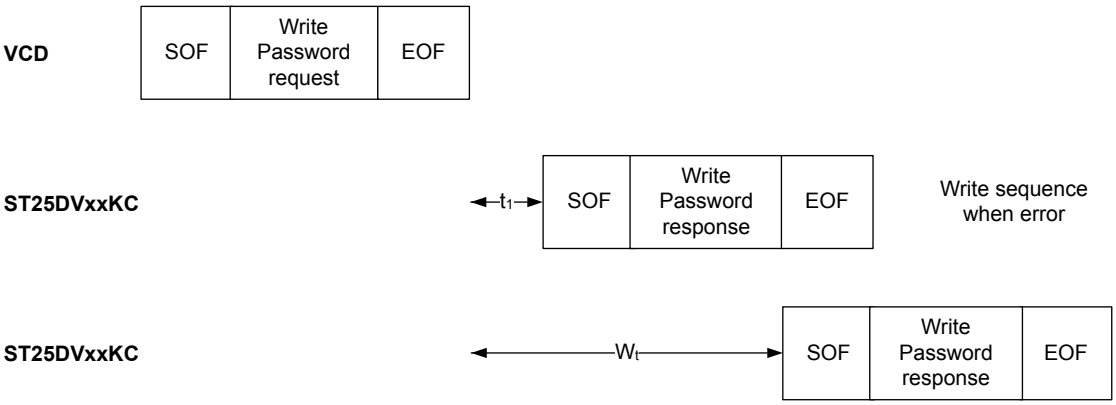
Table 212. Write Password response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 10h: the password number is incorrect
  - 12h: update right not granted, Present Password command not previously executed successfully
  - 13h: the specified block was not successfully programmed

Figure 70. **Write Password frame exchange**



7.6.36 Present Password

When it receives this command, the device compares the requested password with the data contained in the request and reports if the operation has been successful in the response. After a successful command, the security session associate to the password is open as described in Section 5.6: Data protection.

The Option_flag is not supported, and the Inventory_flag must be set to 0.

Table 213. **Present Password request format**

Request SOF	Request_flags	Present Password	IC Mfg code	UID (1)	Password number	Password	CRC16	Request EOF
-	8 bits	B3h	02h	64 bits	8 bits	64 bits	16 bits	-

1. *This field is optional.*

Request parameter:

- Request flags
- UID (optional)
- Password Number:
  - 00h = RF configuration password RF_PWD_0
  - 01h = RF_PWD_1
  - 02h = RF_PWD_2
  - 03h = RF_PWD_3
  - other = Error
- Password

Table 214. **Present Password response format when Error_flag is not set**

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameter:

- No parameter. The response is sent back after the write cycle.

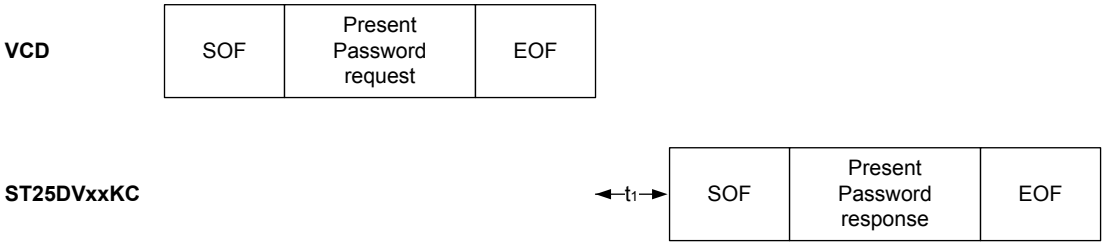
Table 215. **Present Password response format when Error_flag is set**

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: the present password is incorrect
  - 10h: the password number is incorrect

Figure 71. Present Password frame exchange



7.6.37 Fast Read Single Block

When it receives this command, the device reads the requested block and sends back its 32-bit value in the response. When the Option_flag is set, the response includes the Block Security Status. The data rate of the response is multiplied by 2.

The subcarrier_flag should be set to 0, otherwise the device answers with an error code.

The Inventory_flag must be set to 0.

Block number is coded on 1 byte, only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 216. Fast Read Single Block request format

Request SOF	Request_flags	Fast Read Single Block	IC Mfg code	UID (1)	Block number	CRC16	Request EOF
-	8 bits	C0h	02h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number

Table 217. Fast Read Single Block response format when Error_flag is not set

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits	32 bits	16 bits	-

1. *This field is optional.*

Response parameters:

- Block security status if Option_flag is set (see Table 218)
- Four bytes of block data

Table 218. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0

Reserved for future use All at 0	0: Current block not locked 1: Current block locked

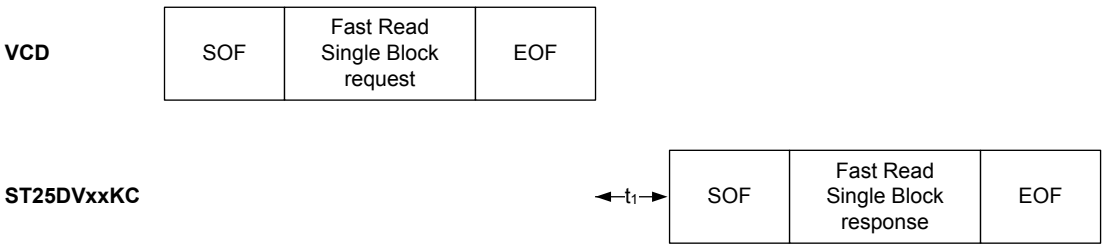
Table 219. Fast Read Single Block response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 72. Fast Read Single Block frame exchange



7.6.38 Fast Extended Read Single Block

When it receives this command, the device reads the requested block and sends back its 32-bit value in the response. When the Option_flag is set, the response includes the Block Security Status. The data rate of the response is multiplied by 2.

The subcarrier_flag should be set to 0, otherwise the device answers with an error code.

The Inventory_flag must be set to 0.

Block number is coded on two bytes, all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 220. Fast Extended Read Single Block request format

Request SOF	Request_flags	Fast Extended Read Single Block	IC Mfg code	UID (1)	Block number	CRC16	Request EOF
-	8 bits	C4h	02h	64 bits	16 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flags
- UID (optional)
- Block number (from LSB byte to MSB byte)

Table 221. Fast Extended Read Single Block response format when Error_flag is not set

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits	32 bits	16 bits	-

1. *This field is optional.*

Response parameters:

- Block security status if Option_flag is set (see the table below)
- Four bytes of block data

Table 222. Block security status

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use All at 0							0: Current block is not locked 1: Current block is locked

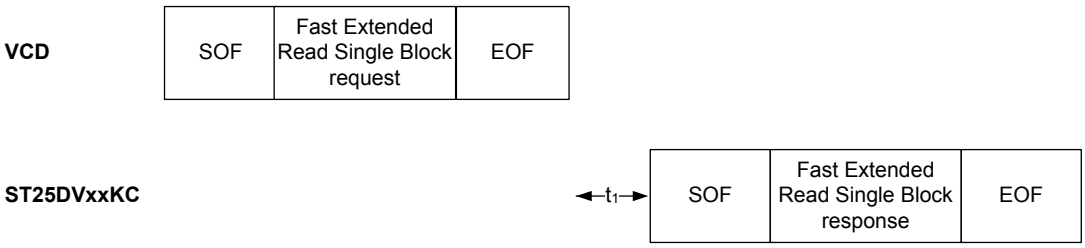
Table 223. Fast Extended Read Single Block response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: the specified block is not available
  - 15h: the specified block is read-protected

Figure 73. Fast Extended Read Single Block frame exchange



7.6.39 Fast Read Multiple Blocks

When it receives this command, the device reads the selected blocks and sends back their value in multiples of 32 bits in the response. The blocks are numbered from 00h up to the last block of user memory in the request, and the value is minus one (–1) in the field. For example, if the “Number of blocks” field contains the value 06h, seven blocks are read. The maximum number of blocks is fixed to 256. This command can cross areas borders, and returns all blocks until reaching a non readable block (block read protected or out of memory).

When the Option_flag is set, the response includes the Block Security Status. The data rate of the response is multiplied by 2.

The subcarrier_flag should be set to 0, otherwise the device answers with an error code.

The Inventory_flag must be set to 0.

Block number is coded on 1 byte, only the first 256 blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 224. Fast Read Multiple Blocks request format

Request SOF	Request_flags	Fast Read Multiple Blocks	IC Mfg code	UID (1)	First block number	Number of blocks	CRC16	Request EOF
-	8 bits	C3h	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)
- First block number
- Number of blocks

Table 225. Fast Read Multiple Blocks response format when Error_flag is not set

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits (2)	32 bits(2)	16 bits	-

1. *This field is optional.*
2. *Repeated as needed.*

Response parameters:

- Block security status if Option_flag is set (see Table 226)
- N block of data

Table 226. Block security status if Option_flag is set

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use All at 0							0: Current not locked 1: Current locked

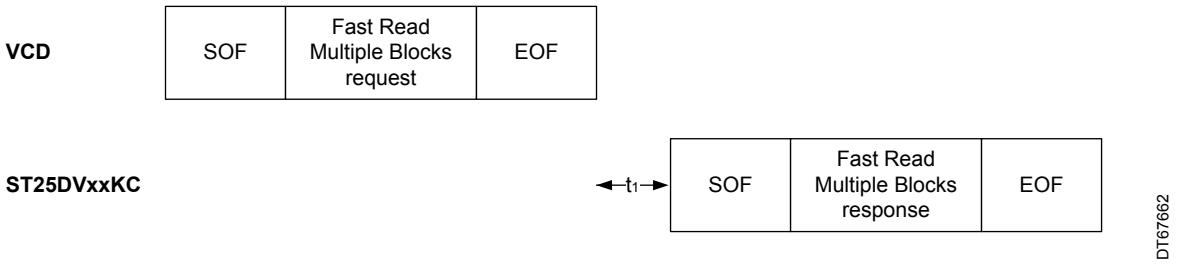
Table 227. Fast Read Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 0Fh: error with no information given
  - 03h: the option is not supported
  - 10h: block address not available
  - 15h: block read-protected

Figure 74. **Fast Read Multiple Blocks frame exchange**



7.6.40 Fast Extended Read Multiple Blocks

When it receives this command, the device reads the selected blocks and sends back their value in multiples of 32 bits in the response. The blocks are numbered from 00h up to the last block of memory in the request and the value is minus one (–1) in the field. For example, if the “Number of blocks” field contains the value 06h, seven blocks are read. The maximum number of blocks is fixed to 2047. This command can cross areas borders, and returns all blocks until reaching a non readable block (block read protected or out of memory).

When the Option_flag is set, the response includes the Block Security Status. The data rate of the response is multiplied by 2.

The subcarrier_flag should be set to 0, otherwise the device answers with an error code.

The Inventory_flag must be set to 0.

Block number is coded on two bytes, all memory blocks of ST25DV16KC and ST25DV64KC can be addressed.

Table 228. **Fast Extended Read Multiple Blocks request format**

Request SOF	Request_flags	Fast Extended Read Multiple Blocks	IC Mfg code	UID (1)	First block number	Block Number	CRC16	Request EOF
-	8 bits	C5h	02h	64 bits	16 bits	16 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)
- First block number (LSB first)
- Number of blocks (LSB first)

Table 229. **Fast Extended Read Multiple Blocks response format when Error_flag is not set**

Response SOF	Response_flags	Block security status (1)	Data	CRC16	Response EOF
-	8 bits	8 bits(2)	32 bits(2)	16 bits	-

1. *This field is optional.*
2. *Repeated as needed*

Response parameters:

- Block security status if Option_flag is set (see Table 230)
- N block of data

Table 230. **Block security status if Option_flag is set**

b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Reserved for future use							0: Current not locked

All at 0	1: Current locked

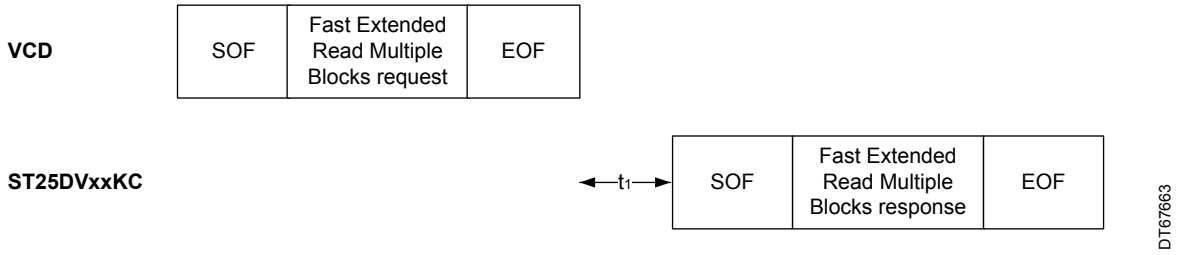
Table 231. Fast Extended Read Multiple Blocks response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: the option is not supported
  - 0Fh: error with no information given
  - 10h: block address not available
  - 15h: block read-protected

Figure 75. Fast Extended Read Multiple Blocks frame exchange



7.6.41 Fast Write Message

When it receives this command, the device puts the data contained in the request into the mailbox buffer, updates the Message Length register MB_LEN_Dyn, and sets Mailbox loaded bit RF_PUT_MSG. It then reports if the write operation was successful in the response. The mailbox contains up to 256 data bytes which are filled from the first location 00h. MSGlength is the number of Data bytes minus - 1 (00h for 1 byte of data, FFh for 256 bytes of data). The command can be executed only when Mailbox is accessible by RF (previous RF message was read or time-out occurs, no I^2C message to be read). User can check it by reading b1 of MB_CTRL_Dyn "HOST_PUT_MSG", which must be reset to 0. (refer to Section 5.1: Fast transfer mode (FTM)).

- The data rate of the response is multiplied by 2 compared with Write Message command.
- The Option_flag is not supported.
- The Inventory_flag must be set to 0.
- The subcarrier_flag should be set to 0, otherwise the device answers with an error code.

Table 232. Fast Write Message request format

Request SOF	Request_flags	Fast Write Message	IC Mfg code	UID (1)	MSGLength	Message Data	CRC16	Request EOF
-	8 bits	CAh	02h	64 bits	1 byte	(MsgLength + 1) bytes	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)
- Message length
- Message data

Table 233. Fast Write Message response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameters:

- No parameter. The response is sent back after the write cycle.

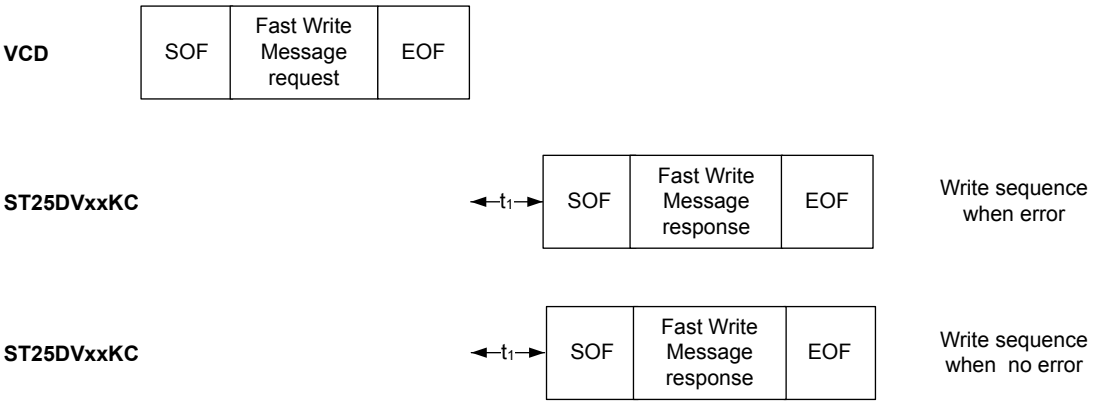
Table 234. Fast Write Message response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given

Figure 76. Fast Write Message frame exchange



7.6.42 Fast Read Message Length

When it receives this command, the device reads the MB_LEN_dyn register, which contains the mailbox message length, and sends back its 8-bit value in the response.

The value of MB_LEN_Dyn returned is the size of the message length in bytes - 1.

The Option_flag is not supported. The Inventory_flag must be set to 0.

The subcarrier_flag must be set to 0, otherwise the device answers with an error code.

The data rate of the response is multiplied by 2 compared to Read Message Length command.

Table 235. Fast Read Message Length request format

Request SOF	Request_flags	Fast Read Message Length	IC Mfg code	UID (1)	CRC16	Request EOF
-	8 bits	CBh	02h	64 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)

Table 236. Fast Read Message Length response format when Error_flag is not set

Response SOF	Response_flags	Data	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameters:

- One byte of data: volatile Control register.

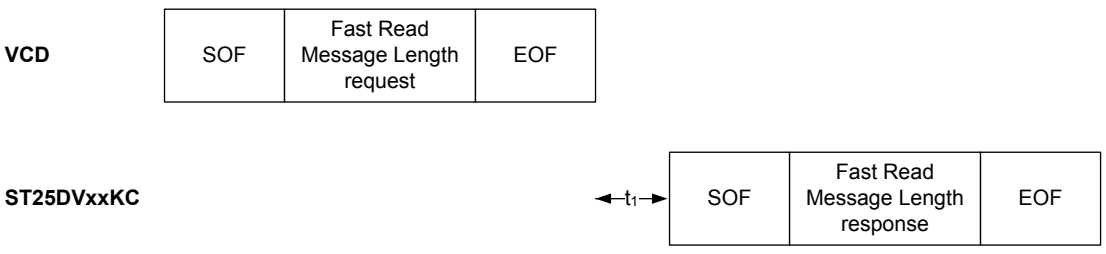
Table 237. Fast Read Message Length response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command option not recognized
  - 03h: command not supported
  - 0Fh: error with no information given

Figure 77. Fast Read Message Length frame exchange



7.6.43 Fast Read Dynamic Configuration

When it receives this command, the device reads the Dynamic register address by the pointer and sends back its 8-bit value in the response.

The Option_flag is not supported. The Inventory_flag must be set to 0.

The subcarrier_flag must be set to 0, otherwise the device answers with an error code.

The data rate of the response is multiplied by 2 compared to Read Dynamic configuration command.

Table 238. Fast Read Dynamic Configuration request format

Request SOF	Request_flags	Fast Read Dynamic configuration	IC Mfg code	UID (1)	Pointer address	CRC16	Request EOF
-	8 bits	CDh	02h	64 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)

Table 239. Fast Read Dynamic Configuration response format when Error_flag is not set

Response SOF	Response_flags	Data	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameters:

- One byte of data

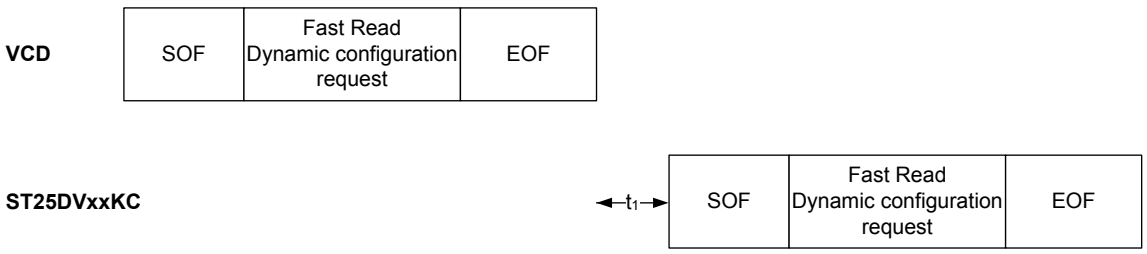
Table 240. Fast Read Dynamic Configuration response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: block not available

Figure 78. Fast Read Dynamic Configuration frame exchange



7.6.44 Fast Write Dynamic Configuration

When it receives this command, the device updates the Dynamic register addressed by the pointer.
The Option_flag is not supported. The Inventory_flag must be set to 0.
The data rate of the response is multiplied by 2 compared to Write Dynamic Configuration command.

Table 241. Fast Write Dynamic Configuration request format

Request SOF	Request_flags	Fast Write Dynamic Configuration	IC Mfg code	UID (1)	Pointer address	Register Value	CRC16	Request EOF
-	8 bits	CEh	02h	64 bits	8 bits	8 bits	16 bits	-

1. *This field is optional.*

Request parameters:

- Request flag
- UID (optional)
- Pointer address
- Register value

Table 242. Fast Write Dynamic Configuration response format when Error_flag is not set

Response SOF	Response_flags	CRC16	Response EOF
-	8 bits	16 bits	-

Response parameters:

- No parameter. The response is sent back after t_1.

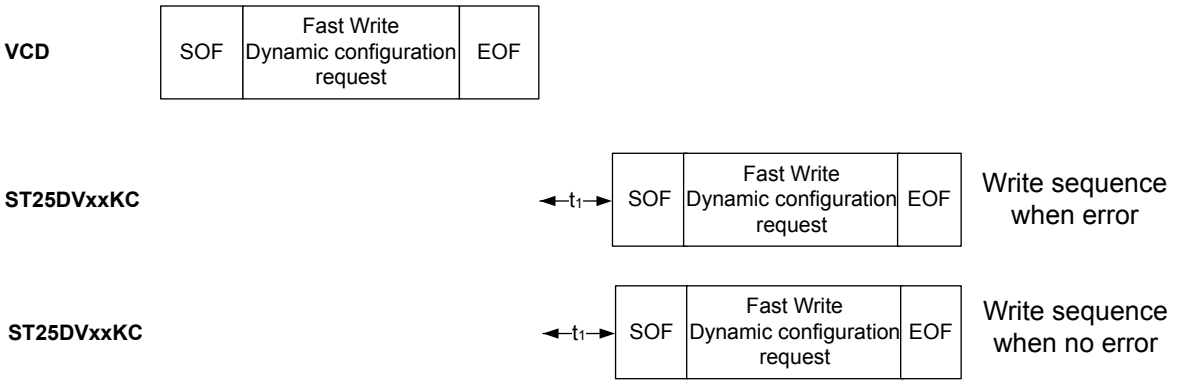
Table 243. Fast Write Dynamic Configuration response format when Error_flag is set

Response SOF	Response_flags	Error code	CRC16	Response EOF
-	8 bits	8 bits	16 bits	-

Response parameter:

- Error code as Error_flag is set:
  - 02h: command not recognized
  - 03h: command option not supported
  - 0Fh: error with no information given
  - 10h: block not available

Figure 79. **Fast Write Dynamic Configuration frame exchange**



8 Unique identifier (UID)

The device is uniquely identified by a 64-bit unique identifier (UID), which complies with ISO/IEC 15963 and ISO/IEC 7816-6. The UID is a read-only code and comprises:

- eight MSBs with a value of E0h
- the IC manufacturer code “ST 02h” on 8 bits (ISO/IEC 7816-6/AM1)
- the product code on 8 bits
- a unique serial number on 40 bits

Table 244. UID format

MSB				LSB			
63	56	55	48	47	40	39	0
0xE0		0x02		ST product code (1)		Unique serial number	

1. *See Table 86. UID for ST product code value definition.*

With the UID, each device can be addressed uniquely and individually during the anticollision loop and for one-to-one exchanges with a VCD.

9 Device parameters

9.1 Maximum ratings

Stressing the device above the ratings listed in Table 245 may cause permanent damage to the device. These are stress ratings only, and operation of the device, at these or any other conditions above those indicated in the operating sections of this specification, is not implied. Exposure to absolute maximum rating conditions for extended periods may affect the device reliability. Device mission profile (application conditions) is compliant with JEDEC JESD47 qualification standard. Extended mission profiles can be assessed on demand.

Refer also to the STMicroelectronics SURE program and other relevant quality documents.

Table 245. Absolute maximum ratings

Symbol	Parameter				Min.	Max.	Unit
T_A	Ambient operating temperature	Range 6	All packages	RF and I^2C interfaces	-40	85	°C
		Range 8	UFDFPN8, UFDFPN12	RF and I^2C interfaces	- 40	105	°C
			SO8N, TSSOP	RF interface	- 40	105	°C
				I^2C interface	- 40	125	°C
T_{STG}	Storage temperature	UFDFPN8 (MLP8),SO8N, TSSOP8, UFDFPN12, WLCSP10			- 65	150	°C
T_{LEAD}	Lead temperature during soldering				see note (1)		°C
V_{IO}	I^2C input or output range				- 0.50	6.5	V
V_{CC}	I^2C supply voltage				- 0.50	6.5	V
$I_{OL_MAX_SDA}$	DC output current on pin SDA (when equal to 0)				-	5	mA
$I_{OL_MAX_GPO}$	DC output current on pin GPO (when equal to 0)				-	1.5	mA
V_{MAX_1}	RF input voltage amplitude peak to peak between AC0 and AC1, V_{SS} pin left floating(2)			V_{AC0} - V_{AC1}	-	11	V
V_{MAX_2}	AC voltage between AC0 and V_{SS}, or AC1 and V_{SS} (2)			V_{AC0} - V_{SS}, or V_{AC1} - V_{SS}	- 0.50	5.5	V
V_{ESD}	Electrostatic discharge voltage (human body model) (3)			All pins	-	2000	V

1. *Compliant with JEDEC Std J-STD-020C (for small body, Sn-Pb or Pb assembly), the ST ECOPACK 7191395 specification, and the European directive on Restrictions on Hazardous Substances (RoHS) 2002/95/EU.*
2. *Evaluated by characterization – Not tested in production.*
3. *AEC-Q100-002 (compliant with JEDEC Std JESD22-A114, C1 = 100 pF, R1 = 1500 Ω, R2 = 500 Ω)*

9.2 I²C parameters

This section summarizes the operating and measurement conditions, and the DC and AC characteristics of the device in I²C mode. The parameters are derived from tests performed under the measurement conditions summarized in the relevant tables. Check that the operating conditions in the circuit match the measurement conditions when relying on the quoted parameters.

Table 246. I²C operating conditions

Symbol	Parameter			Min.	Max.	Unit
V_{CC}	Supply voltage			1.8	5.5	V
T_A	Ambient operating temperature	Range 6	All packages	-40	85	°C
		Range 8	UFDFPN8, UFDFPN12	-40	105	°C
			SO8N, TSSOP8	-40	125	°C

Table 247. AC test measurement conditions

Symbol	Parameter	Min.	Max.	Unit
C_L	Load capacitance	100	-	pF
t_r, t_f	Input rise and fall times	-	50	ns
V_{hi-lo}	Input levels	0.2 V_{CC} to 0.8 V_{CC}		V
$V_{ref(t)}$	Input and output timing reference levels	0.3 V_{CC} to 0.7 V_{CC}		V

Figure 80. AC test measurement I/O waveform

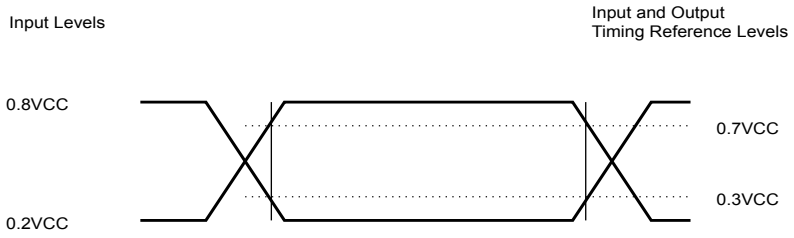


Table 248. Input parameters

Symbol	Parameter	Min.	Max.	Unit
C_{IN}	Input capacitance (SDA)	-	8	pF
C_{IN}	Input capacitance (other pins)	-	6	pF
t_{NS} (1)	Pulse width ignored (input filter on SCL and SDA)	-	80	ns

1. *Evaluated by characterization – Not tested in production.*

Table 249. I²C DC characteristics up to 85 °C

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
I_{LI}	Input leakage current (SCL, SDA)	V_{IN} = V_{SS} or V_{CC} device in Standby mode	-	0.03	0.1	µA
	Input leakage current (LPD)	V_{IN} = V_{SS} device in Standby mode	-	0.1	0.5	µA
I_{LO}	Output leakage current (SDA)	SDA in high-Z, external voltage applied on SDA: V_{SS} or V_{CC}	-	0.03	0.1	µA
$I_{CC_E^2}$	Operating supply current (device select E^2 address) read (1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	116	160	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	220	240	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	510	550	
I_{CC_MB}	Operating supply current (device select MB address) read(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	116	160	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	220	240	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	510	550	
I_{CC0}	Operating supply current (device select E^2 address) write(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	110	210	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	110	220	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	130	250	
I_{CC0_MB}	Operating supply current (device select MB address) write(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	170	200	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	280	300	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	520	600	
I_{CC1} (LPD = 1)	Low power down supply current	V_{CC} = 1.8 V	-	0.84	1.5	µA
		V_{CC} = 3.3 V	-	1.3	2	
		V_{CC} = 5.5 V	-	1.7	3	
I_{CC1_PON} (LPD = 0)	Static standby supply current after power ON or device select stop or time out	V_{CC} = 1.8 V	-	72	100	µA
		V_{CC} = 3.3 V	-	76	100	
		V_{CC} = 5.5 V	-	87	120	
V_{IL}	Input low voltage (SDA, SCL)	V_{CC} = 1.8 V	-0.45	-	0.25 V_{CC}	V
		V_{CC} = 3.3 V	-0.45	-	0.3 V_{CC}	
		V_{CC} = 5.5 V	-0.45	-	0.3 V_{CC}	
V_{IL_LPD}	Input low voltage (LPD)	V_{CC} = 3.3 V	-0.45	-	0.2 V_{CC}	V

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
V_{IH}	Input high voltage (SDA, SCL)	V_{CC} = 1.8 V	0.75 V_{CC}	-	6.5	V
		V_{CC} = 3.3 V	0.75 V_{CC}	-	6.5	
		V_{CC} = 5.5 V	0.75 V_{CC}	-	6.5	
V_{IH_LPD}	Input high voltage (LPD)	V_{CC} = 1.8 V	0.85 V_{CC}	-	6.5	V
		V_{CC} = 3.3 V	0.85 V_{CC}	-	6.5	
		V_{CC} = 5.5 V	0.85 V_{CC}	-	6.5	
V_{OL_SDA}	Output low voltage SDA (1 MHz)	I_{OL} = 1 mA, V_{CC} = 1.8 V	-	0.05	0.4	V
		I_{OL} = 2.1 mA, V_{CC} = 3.3 V	-	0.075	0.4	
		I_{OL} = 3 mA, V_{CC} = 5.5 V	-	0.09	0.4	
$V_{CC_Power_up}$	Device select acknowledge	f_C = 100 kHz(2)	-	1.48	1.7	V

1. *SCL, SDA connected to ground or V_{CC}. SDA connected to V_{CC} through a pull-up resistor.*
2. *Evaluated by characterization – Not tested in production.*

Table 250. I²C DC characteristics up to 125 °C

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
I_{LI}	Input leakage current (SCL, SDA)	V_{IN} = V_{SS} or V_{CC} device in Standby mode	-	0.03	0.1	µA
	Input leakage current (LPD)	V_{IN} = V_{SS} device in Standby mode	-	0.1	0.5	
I_{LO}	Output leakage current (SDA)	SDA in high-Z, external voltage applied on SDA: V_{SS} or V_{CC}	-	0.03	0.1	µA
$I_{CC_E^2}$	Operating Supply current (Device select E^2 address) Read (1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	126	180	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	230	260	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	510	550	
I_{CC_MB}	Operating supply current (Device select MB Address) Read(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	126	180	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	230	260	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	510	550	
I_{CC0}	Operating supply current (Device select E^2 address) Write(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	120	220	µA
		V_{CC} = 3.3 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	120	230	
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	140	260	
I_{CC0_MB}	Operating Supply current (Device select MB Address) Write(1)	V_{CC} = 1.8 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	180	220	µA
		V_{CC} = 3.3 V, f_C = 1 MHz	-	290	320	

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
I_{CC0_MB}	Operating Supply current (Device select MB Address) Write(1)	(rise/fall time < 50 ns)				µA
		V_{CC} = 5.5 V, f_C = 1 MHz (rise/fall time < 50 ns)	-	520	600	
I_{CC1} (LPD = 1)	Low power down supply current	V_{CC} = 1.8 V	-	2.5	5	µA
		V_{CC} = 3.3 V	-	3	6	
		V_{CC} = 5.5 V	-	4	7	
I_{CC1_PON} (LPD = 0)	Static Standby supply current after power ON or device select stop or time out	V_{CC} = 1.8 V	-	78	110	µA
		V_{CC} = 3.3 V	-	82	110	
		V_{CC} = 5.5 V	-	95	130	
V_{IL}	Input low voltage (SDA, SCL)	V_{CC} = 1.8 V	-0.45	-	0.25 V_{CC}	V
		V_{CC} = 3.3 V	-0.45	-	0.3 V_{CC}	
		V_{CC} = 5.5 V	-0.45	-	0.3 V_{CC}	
V_{IL_LPD}	Input low voltage (LPD)	V_{CC} = 3.3 V	-0.45	-	0.2 V_{CC}	V
V_{IH}	Input high voltage (SDA, SCL)	V_{CC} = 1.8 V	0.75 V_{CC}	-	6.5	V
		V_{CC} = 3.3 V	0.75 V_{CC}	-	6.5	
		V_{CC} = 5.5 V	0.75 V_{CC}	-	6.5	
V_{IH_LPD}	Input high voltage (LPD)	V_{CC} = 1.8 V	0.85 V_{CC}+1	-	6.5	V
		V_{CC} = 3.3 V	0.85 V_{CC}+1	-	6.5	
		V_{CC} = 5.5 V	0.85 V_{CC}+1	-	6.5	
V_{OL_SDA}	Output low voltage SDA (1 MHz)	I_{OL} = 1 mA, V_{CC} = 1.8 V	-	0.05	0.4	V
		I_{OL} = 2.1 mA, V_{CC} = 3.3 V	-	0.08	0.4	
		I_{OL} = 3 mA, V_{CC} = 5.5 V	-	0.1	0.4	
$V_{CC_Power_up}$	Device select acknowledge	f_C = 100 kHz(2)	-	1.48	1.7	V

1. *SCL, SDA connected to ground or V_{CC}. SDA connected to V_{CC} through a pull-up resistor.*
2. *Evaluated by characterization – Not tested in production.*

Table 251. I²C AC characteristics up to 85 °C

Test conditions specified in Table 246. I²C operating conditions					
Symbol	**Alt.**	**Parameter**	**Min.**	**Max.**	**Unit**
f_C	f_{SCL}	Clock frequency	0.05	1000	kHz
t_{CHCL}	t_{HIGH}	Clock pulse width high(1)	0.26	25000(2)	µs
t_{CLCH}	t_{LOW}	Clock pulse width low(1)	0.5	25000 (3)	µs
t_{START_OUT}	-	I²C timeout on Start condition(1)	35	-	ms
t_{XH1XH2}	t_R	Input signal rise time(1)	(4)	(4)	ns
t_{XL1XL2}	t_F	Input signal fall time(1)	(4)	(4)	ns
t_{DL1DL2}	t_F	SDA (out) fall time(1)	20	120	ns
t_{DXCX}	$t_{SU:DAT}$	Data in set up time(1)	50	-	ns
t_{CLDX}	$t_{HD:DAT}$	Data in hold time	0	-	ns
t_{CLQX}	t_{DH}	Data out hold time(5)	100	-	ns
t_{CLQV}	t_{AA}	Clock low to next data valid (access time)(6)	-	450	ns
t_{CHDX}	$t_{SU:STA}$	Start condition set up time(7)	250	-	ns
t_{DLCL}	$t_{HD:STA}$	Start condition hold time	0.25	35000(8)	µs
t_{CHDH}	$t_{SU:STO}$	Stop condition set up time	250	-	ns
t_{DHDL}	t_{BUF}	Time between Stop condition and next Start condition	1400	-	ns
t_W	-	I²C write time (9)	-	5	ms
t_{bootDC}	-	RF OFF and LPD = 0(1)	-	0.6	ms
$t_{bootLPD}$	-	RF OFF(1)	-	0.6	ms

1. *Evaluated by characterization – Not tested in production.*
2. *t_{CHCL} timeout.*
3. *t_{CLCH} timeout.*
4. *There are no min. or max. values for the input signal rise and fall times. It is recommended by the I²C specification that the input signal rise and fall times be less than 120 ns when f_C < 1 MHz.*
5. *To avoid spurious Start and Stop conditions, a minimum delay is placed between SCL=1 and the falling or rising edge of SDA.*
6. *t_{CLQV} is the time (from the falling edge of SCL) required by the SDA bus line to reach 0.8 V_{CC} in a compatible way with the I²C specification (which specifies $t_{SU:DAT}$ (min) = 100 ns), assuming that the $R_{bus} \times C_{bus}$ time constant is less than 150 ns (as specified in Figure 82.*
7. *For a reStart condition, or following a write cycle.*
8. *t_{DLCL} timeout*
9. *I²C write time for 1 byte, up to 16 bytes in EEPROM (user memory), provided they are all located in the same memory row, meaning that the most significant memory address bits (b15-b4) are the same*

Table 252. I²C AC characteristics up to 125 °C

Test conditions specified in Table 246. I²C operating conditions					
Symbol	**Alt.**	**Parameter**	**Min.**	**Max.**	**Unit**
f_C	f_{SCL}	Clock frequency	0.05	1000	kHz
t_{CHCL}	t_{HIGH}	Clock pulse width high	0.26	25000 (1)	µs
t_{CLCH}	t_{LOW}	Clock pulse width low	0.5	25000 (2)	µs
t_{START_OUT}	-	I²C timeout on Start condition(3)	35	-	ms
t_{XH1XH2}	t_R	Input signal rise time(3)	(4)	-(4)	ns
t_{XL1XL2}	t_F	Input signal fall time(3)	(4)	(4)	ns
t_{DL1DL2}	t_F	SDA (out) fall time(3)	20	120	ns
t_{DXCX}	$t_{SU:DAT}$	Data in set up time(3)	50	-	ns
t_{CLDX}	$t_{HD:DAT}$	Data in hold time	0	-	ns
t_{CLQX}	t_{DH}	Data out hold time(5)	100	-	ns
t_{CLQV}	t_{AA}	Clock low to next data valid (access time)(6)	-	450	ns
t_{CHDX}	$t_{SU:STA}$	Start condition set up time(7)	250	-	ns
t_{DLCL}	$t_{HD:STA}$	Start condition hold time	0.25	35000 (8)	µs
t_{CHDH}	$t_{SU:STO}$	Stop condition set up time	250	-	ns
t_{DHDL}	t_{BUF}	Time between Stop condition and next Start condition	1400	-	ns
t_W	-	I²C write time (9)	-	5.5	ms
t_{bootDC}	-	RF OFF and LPD = 0(3)	-	0.6	ms
t_{boot_LPD}	-	RF OFF(3)	-	0.6	ms

1. *t_{CHCL} timeout.*
2. *t_{CLCH} timeout.*
3. *Evaluated by characterization – Not tested in production.*
4. *There are no min. or max. values for the input signal rise and fall times. It is recommended by the I²C specification that the input signal rise and fall times be less than 120 ns when f_C < 1 MHz.*
5. *To avoid spurious Start and Stop conditions, a minimum delay is placed between SCL=1 and the falling or rising edge of SDA.*
6. *t_{CLQV} is the time (from the falling edge of SCL) required by the SDA bus line to reach 0.8 V_{CC} in a compatible way with the I²C specification (which specifies $t_{SU:DAT}$ (min) = 100 ns), assuming that the R_{bus} × C_{bus} time constant is less than 150 ns (as specified in the Figure 82. I²C Fast mode (f_C = 1 MHz): maximum R_{bus} value versus bus parasitic capacitance (C_{bus})).*
7. *For a restart condition, or following a write cycle.*
8. *t_{DLCL} timeout.*
9. *I²C write time for 1 byte, up to 16 bytes in EEPROM (user memory) provided they are all located in the same memory row, meaning that the most significant memory address bits (b15-b4) are the same.*

Figure 81. I²C AC waveforms

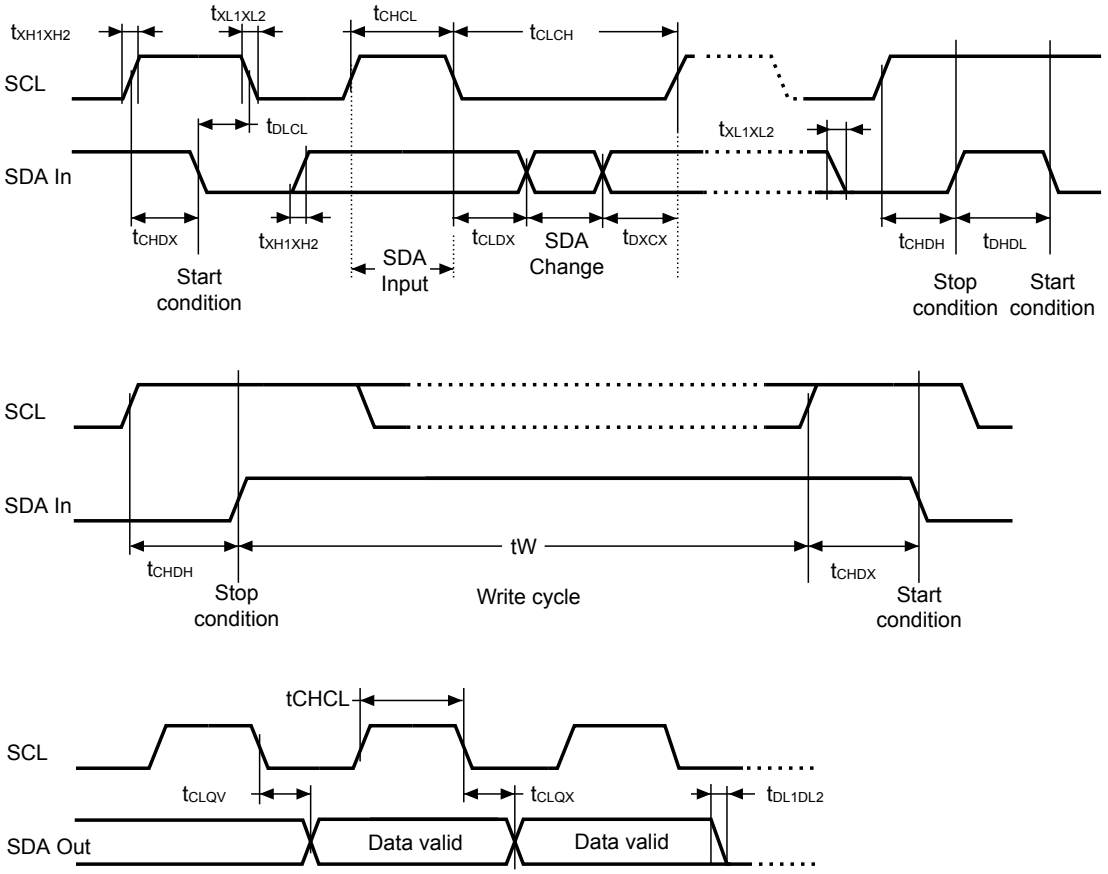
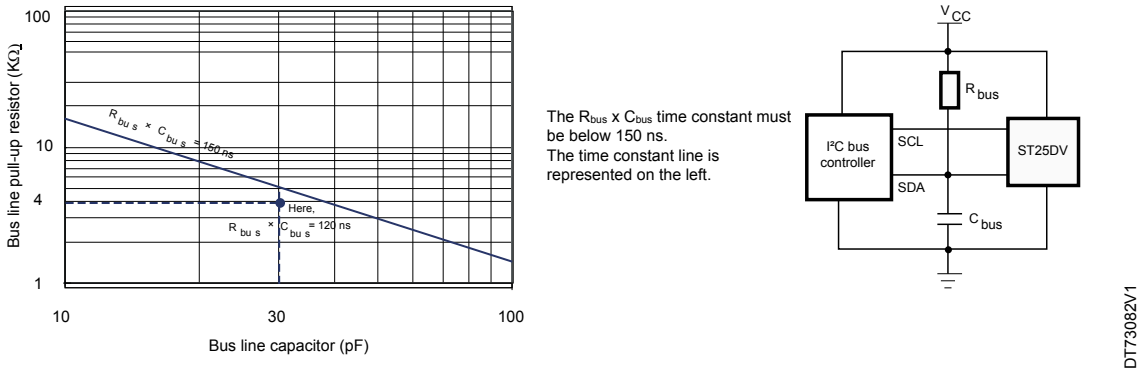


Figure 82 shows how to calculate the value of the pull-up resistor. In the applications where this method of synchronization is not employed, the pull-up resistor is unnecessary, provided that the bus controller has a push-pull (rather than open drain) output.

Figure 82. I²C Fast mode (f_C = 1 MHz): maximum R_{bus} value versus bus parasitic capacitance (C_{bus})



9.3 GPO characteristics

This section summarizes the operating and measurement conditions of the GPO feature. The parameters in the DC and AC characteristic tables that follow are derived from tests performed under the measurement conditions summarized in the relevant tables.

Table 253. GPO DC characteristics up to 85 °C

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$V_{OL_GPO_CMOS}$	Output low voltage (GPO CMOS)	V_{DCG} = 1.8 V, I_{OL} = 0.5 mA	-	-	0.4	V
		V_{DCG} = 3.3 V, I_{OL} = 0.5 mA	-	-	0.4	V
		V_{DCG} = 5.5 V, I_{OL} = 0.5 mA	-	-	0.4	V
$V_{OH_GPO_CMOS}$	Output high voltage (GPO CMOS)	V_{DCG} = 1.8 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
		V_{DCG} = 3.3 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
		V_{DCG} = 5.5 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
$V_{OL_GPO_OD}$	Output low voltage (GPO open drain)	I_{OL} = 1 mA, VCC = 1.8 V	-	0.28	0.4	V
		I_{OL} = 1 mA, VCC = 3.3 V	-	0.2	0.4	
		I_{OL} = 1 mA, VCC = 5.5 V	-	0.2	0.4	
$I_{L_GPO_OD}$	Output leakage current (GPO open drain)	GPO in Hi-Z, external voltage applied on: GPO, V_{SS} or V_{CC}	-0.15	0.06	0.15	µA
I_{LI_VDCG}	Input leakage (V_{DCG})	V_{DCG} = 5.5 V	-	-	0.1	µA

Table 254. GPO DC characteristics up to 125 °C

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$V_{OL_GPO_CMOS}$	Output low voltage (GPO CMOS)	V_{DCG} = 1.8 V, I_{OL} = 0.5 mA	-	-	0.4	V
		V_{DCG} = 3.3 V, I_{OL} = 0.5 mA	-	-	0.4	V
		V_{DCG} = 5.5 V, I_{OL} = 0.5 mA	-	-	0.4	V
$V_{OH_GPO_CMOS}$	Output high voltage (GPO CMOS)	V_{DCG} = 1.8 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
		V_{DCG} = 3.3 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
		V_{DCG} = 5.5 V, I_{OL} = 0.5 mA	V_{DCG}-0.4	-	-	V
$V_{OL_GPO_OD}$	Output low voltage (GPO open drain)	I_{OL} = 1 mA, VCC = 1.8 V	-	0.28	0.4	V
		I_{OL} = 1 mA, VCC = 3.3 V	-	0.22	0.4	
		I_{OL} = 1 mA, VCC = 5.5 V	-	0.21	0.4	
$I_{L_GPO_OD}$	Output leakage current (GPO open drain)	GPO in Hi-Z, external voltage applied on: GPO, V_{SS} or V_{CC}	-0.15	0.06	0.15	µA
I_{LI_VDCG}	Input leakage (V_{DCG})	V_{DCG} = 5.5 V	-	-	0.1	µA

Table 255. GPO AC characteristics

Symbol	Parameter	Test condition	Min.	Max	Unit
$t_{r_GPO_CMOS}$	Output rise time(1)	C_L = 30 pF, V_{DCG} = 1.8 V to 5.5 V	-	50	ns
$t_{f_GPO_CMOS}$	Output fall time(1)	C_L = 30 pF, V_{DCG}=1.8 V to 5.5 V	-	50	ns

1. *Evaluated by characterization – Not tested in production.*

9.4 RF electrical parameters

This section summarizes the operating and measurement conditions, and the DC and AC characteristics of the device in RF mode.

The parameters in the following tables are derived from tests performed under the measurement conditions summarized in the relevant tables. Check that the operating conditions in the circuit match the measurement conditions when relying on the quoted parameters.

Table 256. RF characteristics

Symbol	Parameter	Condition		Min	Typ	Max	Unit
f_{CC}	External RF signal frequency	-		13.553	13.56	13.567	MHz
H_ISO	Operating field according to ISO(1)	Range 6	T_A = -40 °C to 85 °C	150	-	5000	mA/m
		Range 8	T_A = -40 °C to 105 °C				
$MI_{CARRIER}$	10% carrier modulation index MI=(A-B)/(A+B)(1)	150 mA/m > H_ISO > 1000 mA/m		10	-	30	%
	100% carrier modulation index(1)	MI=(A-B)/(A+B)		95	-	100	
t_{MINCD}	Minimum time from carrier generation to first data(1)	From H-field min		-	-	1	ms
f_{SH}	Subcarrier frequency high(1)	$f_{CC}/32$		-	423.75	-	kHz
f_{SL}	Subcarrier frequency low(1)	$f_{CC}/28$		-	484.28	-	kHz
t_1	Time for ST25DVxxKC response(1)	$4352/f_{CC}$		318.6	320.9	323.3	µs
t_2	Time between commands(1)	$4192/f_{CC}$		309	311.5	314	µs
t_3	Time between commands(1)	$4384/f_{CC}$		323.3	-	-	µs
W_{t_Block}	RF User memory write time (including internal Verify) (1)(2)	1 block		-	5.2	-	ms
		4 blocks		-	19.7	-	ms
W_{t_Byte}	RF system memory write time including internal Verify)(1)(2)	1 byte		-	4.9	-	ms
W_{t_MB}	RF Mailbox write time (from VCD request SOF to ST25DVxxKC response EOF)(1)(2)	256 bytes		-	80.7	-	ms
Read_MB	RF Mailbox read time (from VCD request SOF to ST25DVxxKC response EOF) (1)(2)	256 bytes		-	81	-	ms
C_{TUN}	Internal tuning capacitor(3)	f = 13.56 MHz		26.5	28.5	30.5	pF
V_{BACK}	Backscattered level as defined by ISO test(1)	-		10	-	-	mV
V_{MIN_1}	RF input voltage amplitude between AC0 and AC1, V_{SS} pin left floating, VAC0-VAC1 peak to peak(1)	Inventory and Read operations		-	4.8	-	V
		Write operations		-	5.25	-	V
V_{MIN_2}	AC voltage between AC0 and V_{SS} or between AC1 and V_{SS} (1)	Inventory and Read operations		-	2.25	-	V
		Write operations		-	2.7	-	V
t_{bootRF}	Without DC supply (No V_{CC})(1)	Set up time		-	0.6	-	ms
t_{RF_OFF}	RF OFF time(1)	Chip reset		2	-	-	ms

1. *Evaluated by characterization – not tested in production.*
2. *For VCD request coded in 1 out of 4 and ST25DVxxKC response in high data rate, single sub carrier.*
3. *Evaluated by characterization at 25°C – tested in production at 25 °C by correlating industrial tester measure with characterization results..*

Note: All timing characterization where performed on a reference antenna with the following characteristics:

- *ISO antenna class 1*
- *Tuning frequency = 13.7 MHz*

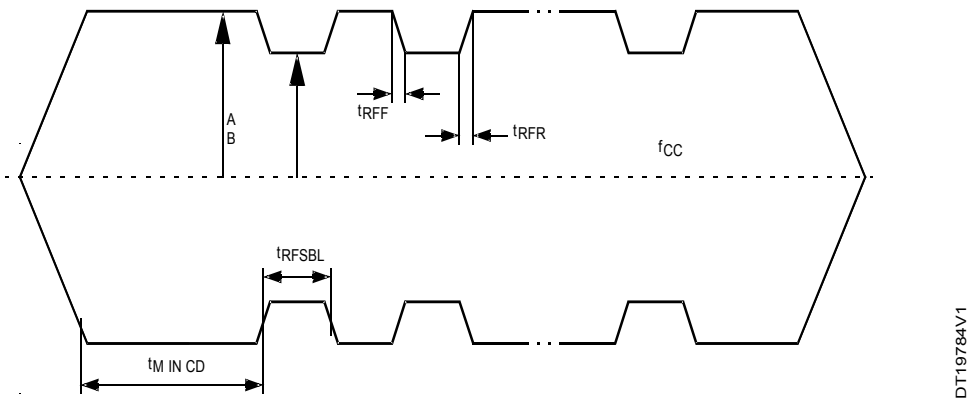
Table 257. Operating conditions

Symbol	Parameter		Min.	Max.	Unit
T_A	Ambient operating temperature	Range 6	-40	85	°C
		Range 8	-40	105	

Figure 83. ASK modulated signal shows an ASK modulated signal from the VCD to the ST25DVxxKC. The test conditions for the AC/DC parameters are:

- Close coupling condition with tester antenna (1 mm)
- ST25DVxxKC performance measured at the tag antenna
- ST25DVxxKC synchronous timing, transmit and receive

Figure 83. ASK modulated signal



9.5 Thermal characteristics

Table 258. Thermal characteristics

Symbol	Parameter	Value	Unit
Θ_{JA}	Thermal resistance junction-ambient SO8N 4.9 x 6 mm, 1.27 mm pitch package(1)	219	°C/W
	Thermal resistance junction-ambient TSSOP8 3 x 6.4 mm, 0.65 mm pitch package(1)	255	
	Thermal resistance junction-ambient UFDFN8 2 × 3 mm, 0.5 mm pitch package(1)(2)	67	

1. *Jedec JESD51-7 2s2p board*
2. *Exposed pad soldered to board*

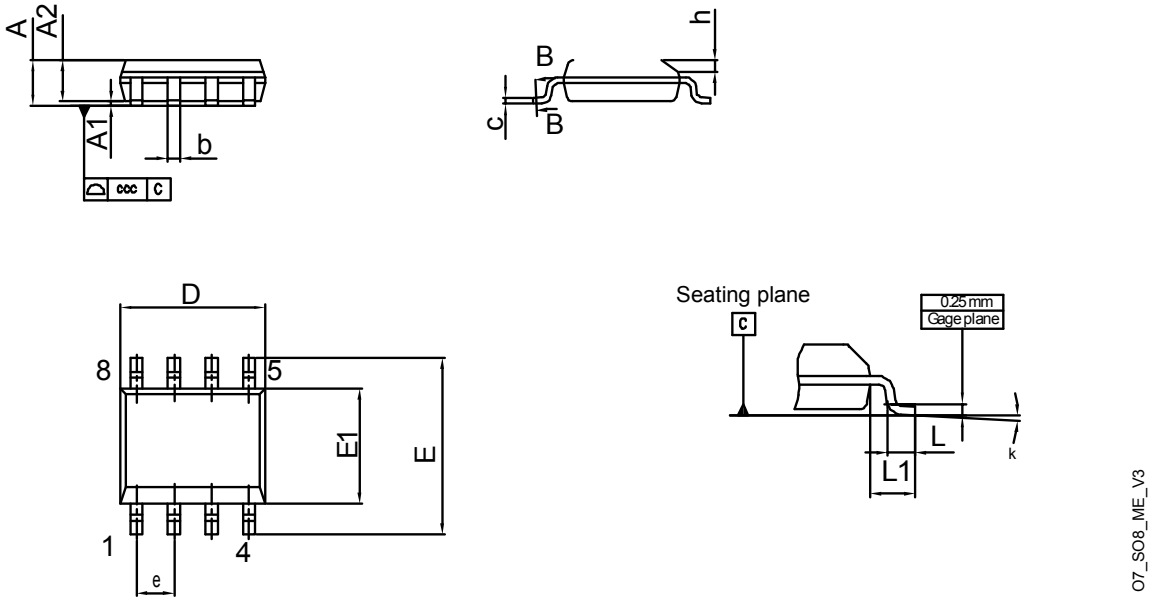
10 Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK packages, depending on their level of environmental compliance. ECOPACK specifications, grade definitions and product status are available at: www.st.com. ECOPACK is an ST trademark.

10.1 SO8N package information

This SO8N is an 8-lead, 4.90 x 6.00 mm, plastic small outline package.

Figure 84. SO8N - Outline



Note: The drawing is not to scale.

Table 259. SO8N - Mechanical data

Symbol	Millimeters			Inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.75	-	-	0.0689
A1	0.10	-	0.25	0.0039	-	0.0098
A2	1.25	-	-	0.0492	-	-
b	0.28	-	0.48	0.0110	-	0.0189
c	0.17	-	0.23	0.0067	-	0.0091
D[2]	4.80	4.90	5.00	0.1890	0.1929	0.1969
E	5.80	6.00	6.20	0.2283	0.2362	0.2441
E1[3]	3.80	3.90	4.00	0.1496	0.1535	0.1575
e	-	1.27	-	-	0.0500	-
h	0.25	-	0.50	0.0098	-	0.0197
L	0.40	-	1.27	0.0157	-	0.0500
L1	-	1.04	-	-	0.0409	-
k	0°	-	8°	0°	-	8°
ccc	-	-	0.10	-	-	0.0039

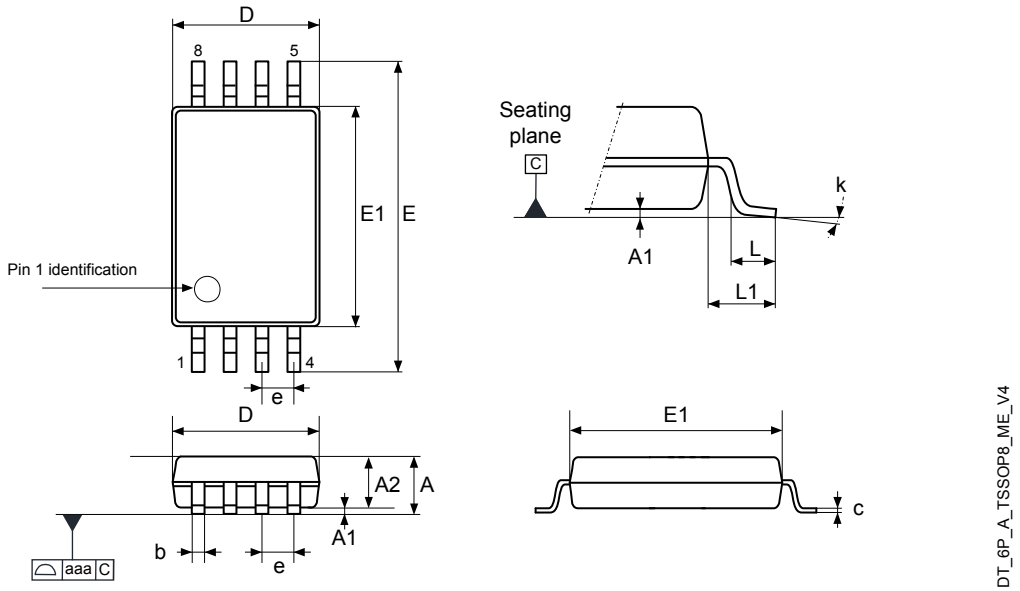
1. *Values in inches are converted from mm and rounded to four decimal digits.*
2. *Dimension “D” does not include mold flash, protrusions or gate burrs. Mold flash, protrusions or gate burrs shall not exceed 0.15 mm per side*
3. *Dimension “E1” does not include interlead flash or protrusions. Interlead flash or protrusions shall not exceed 0.25 mm per side.*

Note: The package top can be smaller than the package bottom. Dimensions D and E1 are determined at the outermost extremes of the plastic body exclusive of the mold flash, tie bar burrs, gate burrs, and interleads flash, but including any mismatch between the top and bottom of the plastic body. The measurement side for the mold flash, protrusions, or gate burrs is the bottom side.

10.2 TSSOP8 package information

This TSSOP is an 8-lead, 3 x 6.4 mm, 0.65 mm pitch, thin shrink small outline package.

Figure 85. TSSOP8 – Outline



1. *Drawing is not to scale.*

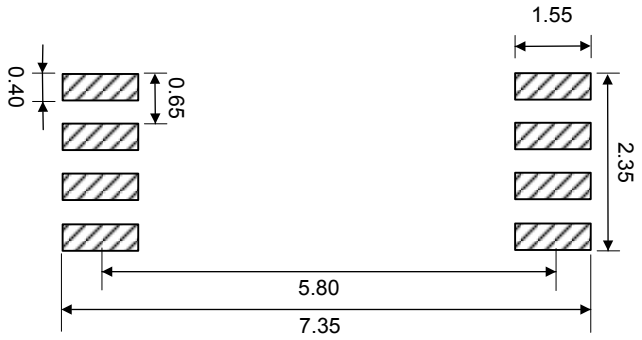
Table 260. TSSOP8 - Mechanical data

Symbol	millimeters			inches (1)		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A	-	-	1.200	-	-	0.0472
A1	0.050	-	0.150	0.0020	-	0.0059
A2	0.800	1.000	1.050	0.0315	0.0394	0.0413
b	0.190	-	0.300	0.0075	-	0.0118
c	0.090	-	0.200	0.0035	-	0.0079
D(2)	2.900	3.000	3.100	0.1142	0.1181	0.1220
e	-	0.650	-	-	0.0256	-
E	6.200	6.400	6.600	0.2441	0.2520	0.2598
E1(3)	4.300	4.400	4.500	0.1693	0.1732	0.1772
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1	-	1.000	-	-	0.0394	-
k	0°	-	8°	0°	-	8°
aaa	-	-	0.100	-	-	0.0039

1. *Values in inches are converted from mm and rounded to four decimal digits.*
2. *Dimension D does not include mold flash, protrusions, or gate burrs. Mold flash, protrusions, or gate burrs shall not exceed 0.15 mm per side.*
3. *Dimension E1 does not include interlead flash or protrusions. Interlead flash or protrusions shall not exceed 0.25 mm per side.*

Note: The package top may be smaller than the package bottom. Dimensions D and E1 are determinated at the outermost extremes of the plastic body exclusive of the mold flash, tie bar burrs, gate burrs, and interleads flash, but including any mismatch between the top and bottom of the plastic body. The measurement side for the mold flash, protrusions, or gate burrs is the bottom side.

Figure 86. TSSOP8 – Footprint example

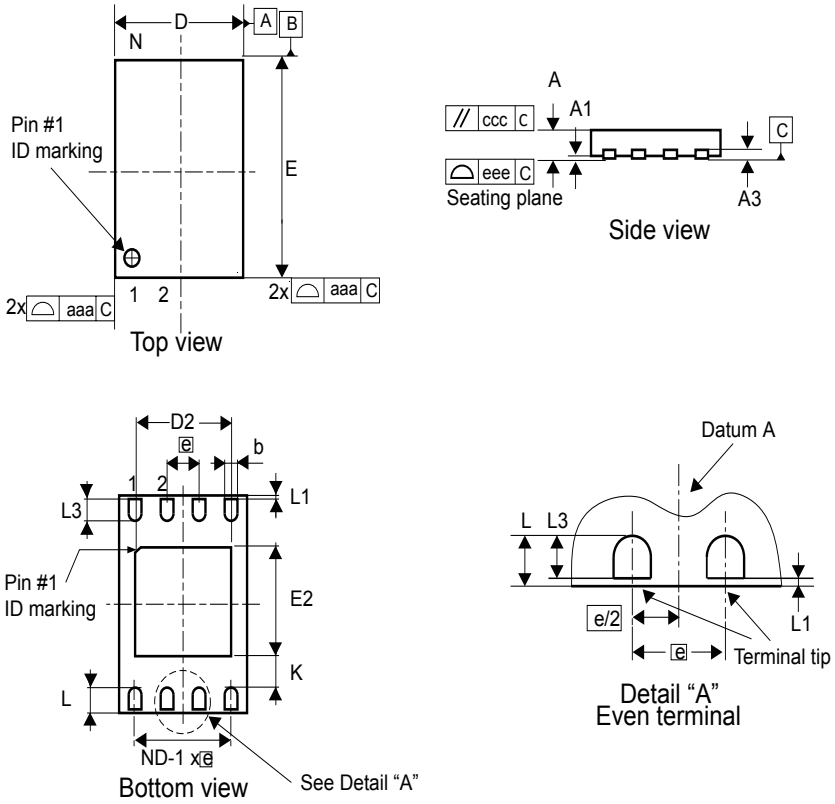


1. *Dimensions are expressed in millimeters.*

10.3 UFDFN8 package information

UFDFPN8 is an 8-lead, 2 × 3 mm, 0.5 mm pitch ultra thin profile fine pitch dual flat package.

Figure 87. UFDFN8 - Outline



1. *Max. package warpage is 0.05 mm.*
2. *Exposed copper is not systematic and can appear partially or totally according to the cross section.*
3. *Drawing is not to scale.*

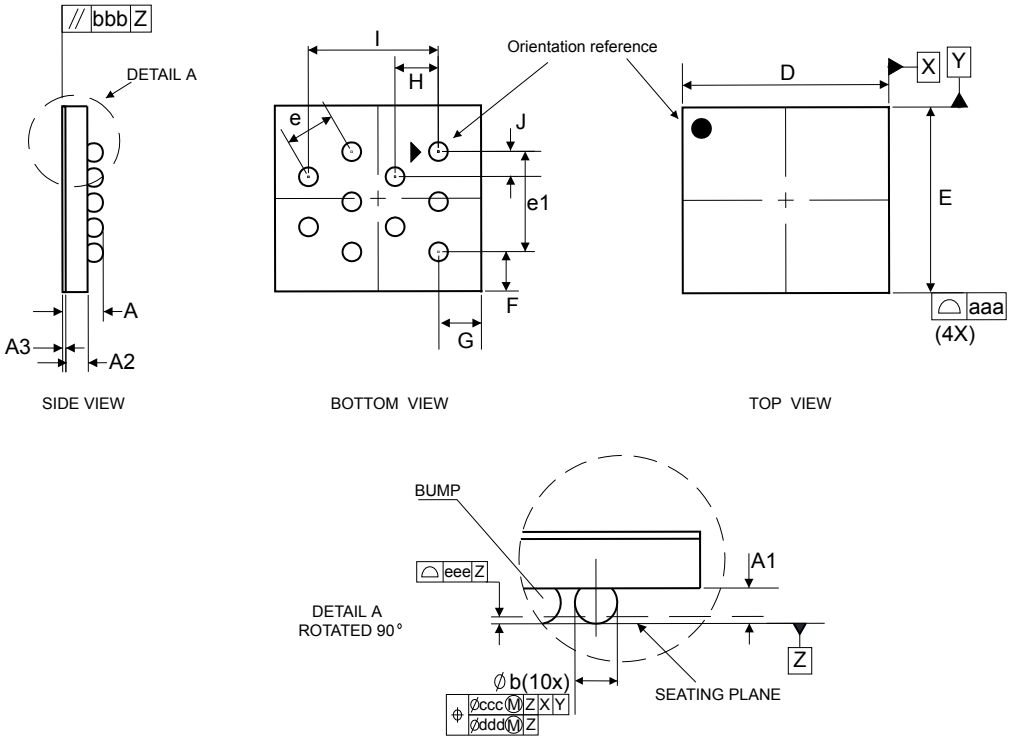
Table 261. UFDFN8 - Mechanical data

Symbol	millimeters			inches [1]		
	Min	Typ	Max	Min	Typ	Max
A	0.450	0.550	0.600	0.0177	0.0217	0.0236
A1	0.000	0.020	0.050	0.0000	0.0008	0.0020
b[2]	0.200	0.250	0.300	0.0079	0.0098	0.0118
D	1.900	2.000	2.100	0.0748	0.0787	0.0827
D2	1.200	-	1.600	0.0472	-	0.0630
E	2.900	3.000	3.100	0.1142	0.1181	0.1220
E2	1.200	-	1.600	0.0472	-	0.0630
e	-	0.500	-	0.0197		
K	0.300	-	-	0.0118	-	-
L	0.300	-	0.500	0.0118	-	0.0197
L1	-	-	0.150	-	-	0.0059
L3	0.300	-	-	0.0118	-	-
aaa	-	-	0.150	-	-	0.0059
bbb	-	-	0.100	-	-	0.0039
ccc	-	-	0.100	-	-	0.0039
ddd	-	-	0.050	-	-	0.0020
eee [3]	-	-	0.080	-	-	0.0031

1. *Values in inches are converted from mm and rounded to 4 decimal digits.*
2. *Dimension b applies to plated terminal and is measured between 0.15 and 0.30 mm from the terminal tip.*
3. *Applied for exposed die paddle and terminals. Exclude embedding part of exposed die paddle from measuring.*

10.4 WLCSP10 package information

Figure 88. **WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale package outline**



1. *Drawing is not to scale.*
2. *Dimension is measured at the maximum bump diameter parallel to primary datum Z.*
3. *Primary datum Z and seating plane are defined by the spherical crowns of the bump.*
4. *Bump position designation per JESD 95-1, SPP-010.*

Table 262. **WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale mechanical data**

Symbol	millimeters			inches [1]		
	Min	Typ	Max	Min	Typ	Max
A	0.265	0.295	0.325	0.0104	0.0116	0.0128
A1	-	0.095	-	-	0.0037	-
A2	-	0.175	-	-	0.0069	-
A3	-	0.025	-	-	0.0010	-
b	-	0.185	-	-	0.0073	-
D	-	1.649	1.669	-	0.0649	0.0657
E	-	1.483	1.503	-	0.0584	0.0592
e	-	0.400	-	-	0.0157	-
e1	-	0.800	-	-	0.0315	-
H	-	0.346	-	-	0.0136	-
I	-	1.039	-	-	0.0409	-
J	-	0.200	-	-	0.0079	-

Symbol	millimeters			inches [1]		
	Min	Typ	Max	Min	Typ	Max
F	-	0.314	-	-	0.0124	-
G	-	0.342	-	-	0.0135	-
aaa	-	0.110	-	-	0.0043	-
bbb	-	0.110	-	-	0.0043	-
ccc	-	0.110	-	-	0.0043	-
ddd	-	0.060	-	-	0.0024	-
eee	-	0.060	-	-	0.0024	-

1. *Values in inches are converted from mm and rounded to 4 decimal digits.*

Figure 89. WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale recommended footprint

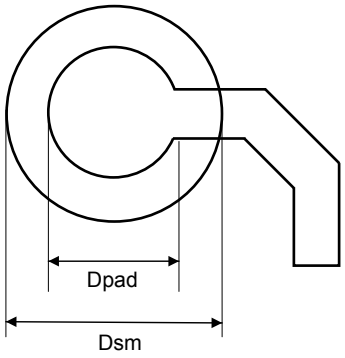


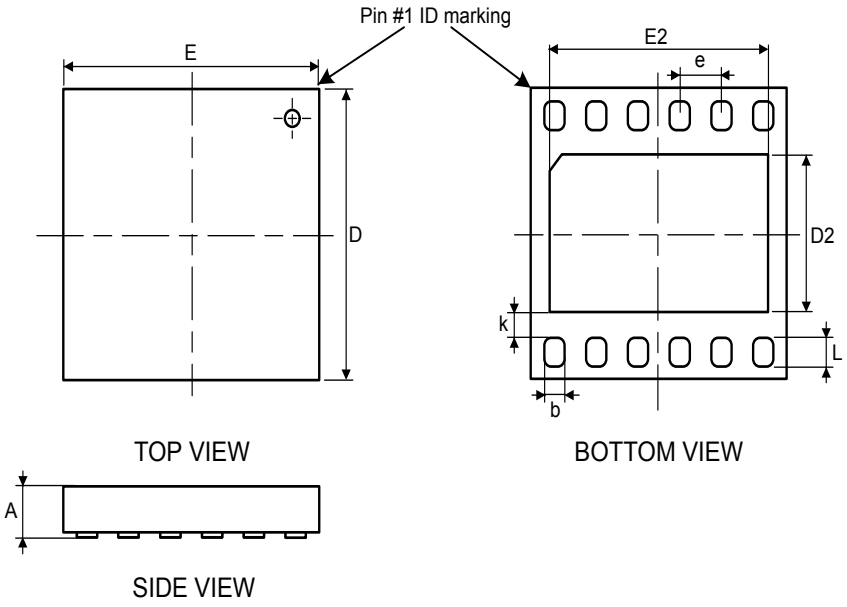
Table 263. WLCSP10 recommended PCB design rules

Dimension	Recommended values
Pitch	0.4 mm
Dpad	0,225 mm
Dsm	0.290 mm typ. (depends on soldermask registration tolerance)
Stencil opening	0.250 mm
Stencil thickness	0.100 mm

10.5 UFDFPN12 package information

UFDFPN12 is an 12-lead, 3 x 3 mm, 0.5 mm pitch ultra thin profile fine pitch dual flat package.

Figure 90. UFDFPN12 - Outline



1. *Drawing is not to scale.*

Table 264. UFDFPN12 - Mechanical data

Symbol	millimeters			inches (1)		
	Min	Typ	Max	Min	Typ	Max
A(2)	0.45	0.55	0.60	0.0177	0.0217	0.0236
b	0.20	0.25	0.30	0.0079	0.0098	0.0118
D	2.95	3.00	3.10	0.1161	0.1181	0.1220
D2	1.35	1.40	1.45	0.0531	0.0551	0.0571
e		0.50			0.0197	
E	2.95	3.00	3.10	0.1161	0.1181	0.1220
E2	2.50	2.55	2.60	0.0984	0.1004	0.1024
L	0.25	0.30	0.35	0.0098	0.0118	0.0138
k		0.40			0.0157	

1. *Values in inches are converted from mm and rounded to 4 decimal digits.*
2. *Package total thickness.*

11 Ordering information

Table 265. Ordering information scheme

Example: ST25DV 64K C - IE 6 D 3

Device type

ST25DV = Dynamic NFC/RFID tag based on ISO 15693 and NFC T5T

Memory size

04K = 4 Kbits

16K = 16 Kbits

64K = 64 Kbits

Version

C

Device Features

IE = I2C and GPO open drain, fast transfer mode and energy harvesting

JF = I2C and GPO CMOS, fast transfer mode, energy harvesting and low power mode

Device grade

6 = industrial: device tested with standard test flow over - 40 to 85 °C

8 = industrial device tested with standard test flow over -40 to 105 °C (UFDFPN8 and UFDFPN12 only) or over -40 to 125 °C (SO8N and TSSOP8 only, 105 °C only for RF interface)

Package

S = SO8N

T = TSSOP8

D = UFDFPN12

C = UFDFPN8

L = WLCSP (thin 10 balls) (Only for 04K version)

Capacitance

3 = 28.5 pF

Note: Parts marked as “ES” or “E” are not yet qualified and therefore not approved for use in production. ST is not responsible for any consequences resulting from such use. In no event will ST be liable for the customer using any of these engineering samples in production. ST’s Quality department must be contacted prior to any decision to use these engineering samples to run a qualification activity.

Appendix A Bit representation and coding for fast commands

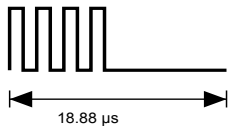
Data bits are encoded using Manchester coding, according to the following schemes. For the low data rate, same subcarrier frequency or frequencies is/are used. In this case, the number of pulses is multiplied by 4 and all times increase by this factor. For the Fast commands using one subcarrier, all pulse numbers and times are divided by 2.

A.1 Bit coding using one subcarrier

A.1.1 High data rate

For the fast commands, a logic 0 starts with four pulses at 423.75 kHz (f_C/32) followed by an unmodulated time of 9.44 µs, as shown in Figure 91.

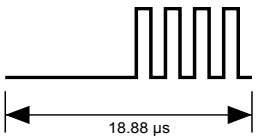
Figure 91. Logic 0, high data rate, fast commands



DT12066bV1

For the Fast commands, a logic 1 starts with an unmodulated time of 9.44 µs followed by four pulses of 423.75 kHz (f_C/32), as shown in Figure 92.

Figure 92. Logic 1, high data rate, fast commands

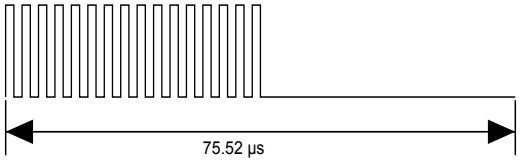


DT12067bV1

A.1.2 Low data rate

For the Fast commands, a logic 0 starts with 16 pulses at 423.75 kHz (f_C/32) followed by an unmodulated time of 37.76 µs, as shown in Figure 93.

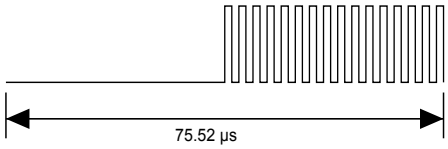
Figure 93. Logic 0, low data rate, fast commands



DT12069bV1

For the Fast commands, a logic 1 starts with an unmodulated time of 37.76 µs followed by 16 pulses at 423.75 kHz (f_C/32), as shown in Figure 94.

Figure 94. **Logic 1, low data rate, fast commands**



DT12071bV1

Note: For fast commands, bit coding using two subcarriers is not supported.

A.2 ST25DVxxKC to VCD frames

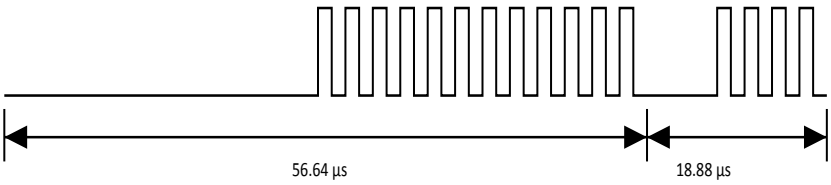
Frames are delimited by an SOF and an EOF. They are implemented using code violation. Unused options are reserved for future use. For the low data rate, the same subcarrier frequency or frequencies is/are used. In this case, the number of pulses is multiplied by 4. For the Fast commands using one subcarrier, all pulse numbers and times are divided by 2.

A.3 SOF when using one subcarrier

A.3.1 High data rate

For the Fast commands, the SOF comprises an unmodulated time of 28.32 µs, followed by 12 pulses at 423.75 kHz (f_C/32), and a logic 1 that consists of an unmodulated time of 9.44 µs followed by four pulses at 423.75 kHz, as shown in Figure 95.

Figure 95. **Start of frame, high data rate, one subcarrier, fast commands**

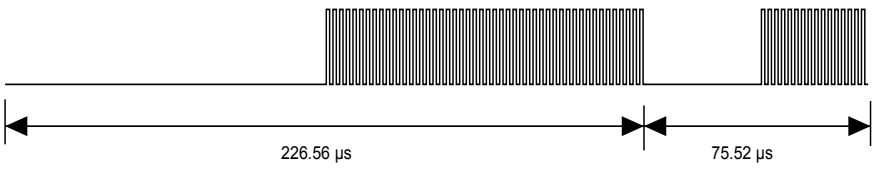


DT12079bV1

A.3.2 Low data rate

For the Fast commands, the SOF comprises an unmodulated time of 113.28 µs, followed by 48 pulses at 423.75 kHz (f_C/32), and a logic 1 that includes an unmodulated time of 37.76 µs followed by 16 pulses at 423.75 kHz, as shown in Figure 96.

Figure 96. **Start of frame, low data rate, one subcarrier, fast commands**



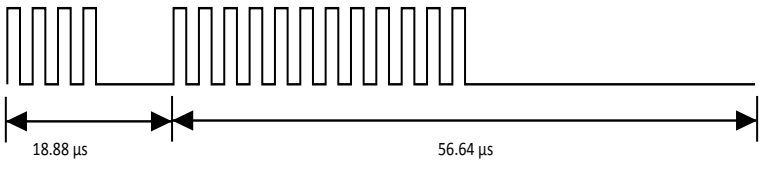
DT12081bV1

A.4 EOF when using one subcarrier

A.4.1 High data rate

For the Fast commands, the EOF comprises a logic 0 that includes four pulses at 423.75 kHz and an unmodulated time of 9.44 µs, followed by 12 pulses at 423.75 kHz (f_C/32) and an unmodulated time of 37.76 µs, as shown in Figure 97.

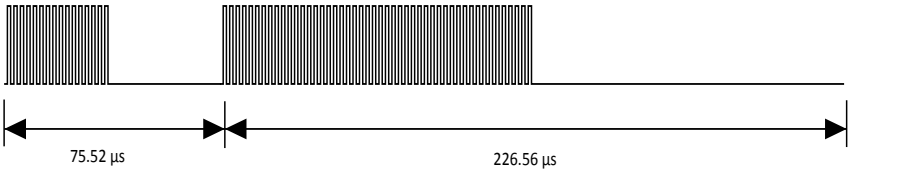
Figure 97. End of frame, high data rate, one subcarrier, fast commands



A.4.2 Low data rate

For the Fast commands, the EOF comprises a logic 0 that includes 16 pulses at 423.75 kHz and an unmodulated time of 37.76 µs, followed by 48 pulses at 423.75 kHz (f_C/32) and an unmodulated time of 113.28 µs, as shown in Figure 98.

Figure 98. End of frame, low data rate, one subcarrier, fast commands



Note: For SOF and EOF in fast commands, bit coding using two subcarriers is not supported.

Appendix B I²C sequences

B.1 Device select codes

Following table assumes default values for I2C_DEVICE_CODE[3:0] (1010b) and E0 (1b) bits. Device select value should be adapted to I2C_DEVICE_CODE[3:0] and E0 values programmed into the I2C_CFG static register if different from default factory values.

Table 266. Device select usage

Device select value		Comment
Hexadecimal	Binary	
-	1010 E211 R/$\overline{W}$	Device select generic E2 = 0b User memory, Dynamic registers, FTM mailbox E2 = 1b System memory
A6h	1010 0110b	User memory, Dynamic registers, FTM mailbox writing
A7h	1010 0111b	User memory, Dynamic registers, FTM mailbox reading
AEh	1010 1110b	System memory writing
AFh	1010 1111b	System memory reading

B.2 I²C Byte writing and polling

B.2.1 I²C byte write in user memory

Table 267. Byte write in user memory when write operation is allowed

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send Address LSB (1 byte)
-	ACK	Ninth bit
DATA	-	Send Data (1 byte)
-	ACK	Ninth bit
Stop	-	Start of programming

Table 268. Polling during programming after byte writing in user memory

Request/Response Frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	NoACK	Ninth bit Device Busy
Start A6h	-	Device select for writing
-	NoACK	Ninth bit device busy
...	...	... Device select for writing
...	...	... Ninth bit device busy
Start A6h	-	Device select for writing
-	ACK	Ninth bit device ready Programming completed
Stop	-	End of polling

Table 269. Byte Write in user memory when write operation is not allowed

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send Address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send Address LSB (1 byte)
-	ACK	Ninth bit
DATA	-	Send data
-	NoACK	Ninth bit: Write access not granted or FTM activated.
Stop	-	No programming Device return in Standby

B.2.2 I²C byte writing in dynamic registers and polling

Table 270. Byte Write in dynamic register (if not read only)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
Dynamic register ADDRESS_LSB	-	Send address LSB (1 byte) Dynamic register are located from address 2000h to 2007h, some are only readable
-	ACK	Ninth bit
DATA	-	Send data
-	ACK	Ninth bit
Stop	-	**Immediate** update of Dynamic register

Table 271. Polling during programming after byte write in dynamic register

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit device busy Dynamic register updates is immediate
Stop	-	End of polling

Table 272. Byte Write in dynamic register if read only

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send address MSB (1 byte)
-	NoACK	Ninth bit
RO Dynamic Register ADDRESS_LSB	-	Send address LSB (1 byte) Addresses 2001h, 2004h, 2005h and 2007h are read only registers.
-	ACK	Ninth bit
DATA	-	Send Data
-	NoACK	Ninth bit
Stop	-	No programming Device returns in Standby

B.2.3 I²C byte write in mailbox and polling

Table 273. Byte Write in mailbox when mailbox is free from RF message and fast transfer mode is activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send mailbox address MSB (1 byte)
-	ACK	Ninth bit
08h	-	Send address LSB (1 byte) Write must be done at the first address of the mailbox
-	ACK	Ninth bit
DATA	-	Send data
-	ACK	Ninth bit
Stop	-	Immediate update of mailbox

Table 274. Byte Write in mailbox when mailbox is not free from RF message fast transfer mode is not activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send mailbox address MSB (1 byte)
-	ACK	Ninth bit
08h	-	Send address LSB (1 byte) Write must be done at the first address of the mailbox
-	ACK	Ninth bit
DATA	-	Send data
-	NoACK	Ninth bit access Mailbox busy or FTM not activated
Stop	-	No programming Device return in Standby

B.2.4 I²C byte write and polling in system memory

Table 275. Byte write if I²C security session is open and register is not RO

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
DATA	-	Send data
-	ACK	Ninth bit
Stop	-	Start of programming

Table 276. Polling during programming after byte write if I²C security session is open and register is not RO

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	NoACK	Ninth bit device busy
Start AEh	-	Device select for writing
-	NoACK	Ninth bit device busy
Start AEh	-	Device select for writing
-	...	Ninth bit
Start AEh	-	Device select for writing
-	ACK	Ninth bit device ready Programing completed
Stop	-	End of polling

Table 277. Byte write if I²C security session is closed or register is RO

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	9th bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
DATA	-	Send data
-	NoACK	Ninth bit
Stop	-	No programming Device return in Standby

B.3 I²C sequential writing and polling

B.3.1 I²C sequential write in user memory and polling

Table 278. Sequential write user memory when write operation is allowed and all bytes belong to same area

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
DATA 0	-	Send Data 0
-	ACK	Ninth bit
DATA 1	-	Send data 1
-	ACK	Ninth bit
...	-	...
-	...	...
DATA n	-	Send data n n ≤ 256
-	ACK	Ninth bit
Stop	-	Start of programming

Table 279. Polling during programming after sequential write in user memory when write operation allowed and all bytes belong to the same area.

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	NoACK	Ninth bit device busy
Start A6h	-	Device select for writing
-	NoACK	Ninth bit device busy
Start A6h	-	Device select for writing
-	...	Ninth bit device busy
Start A6h	-	Device select for writing
-	ACK	Ninth bit device ready Programming completed
Stop	-	End of polling

Table 280. Sequential write in user memory when write operation is allowed and crossing over area border

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
DATA 0	-	Send Data 0
-	ACK	Ninth bit
DATA 1	-	Send data 1
-	ACK	Ninth bit
...	-	...
-	...	...
DATA n	-	Send data n Address is located in the next memory area
-	NoACK	Ninth bit
Stop	-	No programming Device return in Standby

Table 281. Polling during programming after sequential write in user memory when write operation is allowed and crossing over area border

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit device ready No programming
Stop	-	End of polling

B.3.2 I²C sequential write in mailbox and polling

Table 282. Sequential write in mailbox when mailbox is free from RF message and fast transfer mode is activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send mailbox Address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send mailbox address LSB (1 byte)
-	ACK	Ninth bit
DATA 0	-	Send Data 0
-	ACK	Ninth bit
DATA 1	-	Send Data 1
-	ACK	Ninth bit
...	-	...
-	...	...
DATA n	-	Send data n n ≤ 256
-	ACK	Ninth bit
Stop	-	Immediate mailbox content update

Table 283. Polling during programming after sequential write in the mailbox

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit device ready Mailbox is immediately updated
Stop	-	End of polling

B.4 I²C read current address

B.4.1 I²C current address read in user memory

Table 284. Current byte read in user memory if read operation is allowed (depending on area protection and RF user security session)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data located on the last pointed address+1, or at address 0 after power-up, in user memory
NO_ACK	-	Ninth bit
Stop	-	End of Reading

Table 285. Current Read in user memory if read operation is not allowed (depending on area protection and RF user security session)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	FFh	Read of data not allowed ST25DV release SDA
NO_ACK	-	Ninth bit
Stop	-	End of Reading

B.5 I²C random address read

B.5.1 I²C random address read in user memory

Table 286. Random byte read in user memory if read operation is allowed (depending on area protection and RF user security session)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data
NO_ACK	-	Ninth bit
Stop	-	End of Reading

Table 287. Random byte read in user memory if operation is not allowed (depending on area protection and RF user security)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	FFh	Read of data not allowed release SDA
NO_ACK	-	Ninth bit
Stop	-	End of Reading

B.5.2 I²C random address read in system memory

Table 288. Byte read system memory (Static register or I²C password after a valid Present I²C password)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start AFh	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.5.3 I²C random address read in dynamic registers

Table 289. Random byte read in dynamic registers

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.6 I²C sequential read

B.6.1 I²C sequential read in user memory

Table 290. Sequential read user memory if read operation is allowed (depending on area protection and RF user security session) and all bytes belong to the same area

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h0	-	Device select for reading
-	ACK	Ninth bit
-	DATA 0	Receive data 0
ACK	-	Ninth bit
-	DATA 1	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	DATA n	Receive data n
NO_ACK	-	Ninth bit
Stop	-	End of Reading

Table 291. Sequential read user memory if read operation allowed (depending on area protection and RF user security session) but crossing area border

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA 0	Receive data 0
ACK	-	Ninth bit
-	DATA 1	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	DATA n	Receive data last address available
ACK	-	Ninth bit
-	FFh	Data is located in the next memory area ST25DV release SDA
ACK	-	Ninth bit
-	...	...
...	-	...
-	FFh	Data is located in the next memory area ST25DV release SDA
Stop	-	End of reading

Table 292. Sequential Read user memory if read operation is allowed (depending on area protection and RF user security session)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	FFh	ST25DV release SDA Reading access not granted
ACK	-	Ninth bit
-	...	...
...	-	...
-	FFh	ST25DV release SDA Reading access not granted
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.6.2 I²C sequential read in system memory

Table 293. Sequential in read system memory (I²C security session open if reading I2C_PWD)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
ADDRESS_MSB	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start AF7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data 0
ACK	-	Ninth bit
-	DATA	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	DATA	Receive data n
NO_ACK	-	Ninth bit
Stop	-	End of Reading

Table 294. **Sequential Read system memory when access is not granted (I²C password I2C_PWD)**

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
90h	-	Send address MSB (1 byte)
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte)
-	ACK	Ninth bit
Start AFh	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data 0
-	FFh	ST25DV release SDA Reading access is not granted
ACK	-	Ninth bit
-	...	...
...	-	...
-	FFh	ST25DV release SDA Reading access is not granted
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.6.3 I²C sequential read in dynamic registers

Table 295. Sequential read in dynamic register

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send address MSB (1 byte)
-	ACK	Ninth bit
Dynamic register ADDRESS_LSB	-	Send address LSB (1 byte) Dynamic register is located form address 2000h to 2007
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA	Receive data 0
ACK	-	Ninth bit
-	DATA	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	Data	Receive data n
NO_ACK	-	Ninth bit
Stop	-	End of reading

Table 296. Sequential read in dynamic register and mailbox continuously if fast transfer mode is activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h	-	Send address MSB (1 byte)
-	ACK	Ninth bit
Dynamic Register ADDRESS_LSB	-	Send address LSB (1 byte) Dynamic register is located from the address 2000h to 2007h
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA 0	Receive data 0
ACK	-	Ninth bit
-	DATA 1	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	DATA n	Receive data n (n ≤ 8) Last dynamic register address 2007h
ACK	-	Ninth bit
-	DATA n + 1	Mailbox byte 0
ACK	-	Ninth bit
-	DATA n + 2	Mailbox byte 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	Data n + i	Mailbox byte i (i < 256)
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.6.4 I²C sequential read in mailbox

Table 297. Sequential in mailbox if fast transfer mode is activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h or 21h	-	Send address MSB (1 byte) 2007h < @ 2108h
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte) 2007h < @ 2108h
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	DATA 0	Receive data 0
ACK	-	Ninth bit
-	DATA 1	Receive data 1
ACK	-	Ninth bit
-	...	...
...	-	...
-	Data n	Receive data n
NO_ACK	-	Ninth bit
Stop	-	End of reading

Table 298. Sequential read in mailbox if fast transfer mode is not activated

Request/Response Frame		Comment
Controller drives SDA	Target drives SDA	
Start A6h	-	Device select for writing
-	ACK	Ninth bit
20h or 21h	-	Send address MSB (1 byte) 2007h < @ 2108h
-	ACK	Ninth bit
ADDRESS_LSB	-	Send address LSB (1 byte) 2007h < @ 2108h
-	ACK	Ninth bit
Start A7h	-	Device select for reading
-	ACK	Ninth bit
-	FFh	release SDA
ACK	-	Ninth bit
-	FFh	release SDA
ACK	-	Ninth bit
-	...	...
...	-	...
-	FFh	release SDA
NO_ACK	-	Ninth bit
Stop	-	End of reading

B.7 I²C password relative sequences

B.7.1 I²C write password

Table 299. Write password when the I²C security session is already open and fast transfer mode is not activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
09h	-	Send I2C_PWD MSB address
-	ACK	Ninth bit
00h	-	Send I2C_PWD LSB address
-	ACK	Ninth bit
I2C_PWD_BYTE_7	-	Send I2C_PWD MSB
-	ACK	Ninth bit
I2C_PWD_BYTE_6	DATA 0	Send data
-	ACK	Ninth bit
...	-	...
-	...	...
I2C_PWD_BYTE_0	-	Send I2C_PWD LSB
-	ACK	Ninth bit
07h	-	Write password command
-	ACK	Ninth bit
I2C_PWD_BYTE_7	-	Send I2C_PWD MSB
-	ACK	Ninth bit
I2C_PWD_BYTE_6	DATA 0	Send data
-	ACK	Ninth bit
...	-	...
-	...	...
I2C_PWD_BYTE_0	-	Send I2C_PWD LSB
-	ACK	Ninth bit
Stop	-	Start of I²C password programming

Table 300. Write password when the I²C security session is not open or fast transfer mode activated

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
09h	-	Send I2C_PWD MSB address
-	ACK	Ninth bit
00h	-	Send I2C_PWD LSB address
-	NoACK	Ninth bit
Stop	-	No PWD programming Device return in Standby

B.7.2 I²C present password

Table 301. Present password (whatever status of I²C security session or fast transfer mode)

Request/Response frame		Comment
Controller drives SDA	Target drives SDA	
Start AEh	-	Device select for writing
-	ACK	Ninth bit
09h	-	Send I2C_PWD MSB address
-	ACK	Ninth bit
00h	-	Send I2C_PWD LSB address
-	ACK	Ninth bit
I2C_PWD_BYTE_7	-	Send I2C_PWD MSB
-	ACK	Ninth bit
I2C_PWD_BYTE_6	DATA 0	Send data
-	ACK	Ninth bit
...	-	...
-	...	...
I2C_PWD_BYTE_0	-	Send I2C_PWD LSB
-	ACK	Ninth bit
09h	-	Present password command
-	ACK	Ninth bit
I2C_PWD_BYTE_7	-	Send I2C_PWD MSB
-	ACK	Ninth bit
I2C_PWD_BYTE_6	-	Send data
-	ACK	Ninth bit
...	-	...
-	...	...
I2C_PWD_BYTE_0	-	Send I2C_PWD LSB
-	ACK	Ninth bit
Stop	-	ST25DV with active I2C_PWD. Result is immediate.

Revision history

Table 302. **Document revision history**

Date	Revision	Changes
23-Jun-2021	1	Initial release.
22-Jul-2021	2	Modified the title of the document.
09-Feb-2022	3	Added WLCSP10 package Updated: • *Features* • *Section 1.1: Block diagram* • *Section 1.2: Packaging* • *Section 2.2.2: Low power down (LPD)* • *Section 2.4.1: Driver supply voltage (VDCG)* • *Section 2.4.2: General purpose output (GPO)*
22-Jul-2022	4	Updated: • *Features* • *Figure 15. RFSwitchOff command* • *Figure 16. RFSwitchOn command* • *Section 6.3: Device addressing* • *Figure 81. I²C AC waveforms* • *Section 10.1: SO8N package information* • *Section 10.2: TSSOP8 package information* • *Table 265. Ordering information scheme*
12-Jan-2023	5	Updated: • *Features* • *Section 3.1: Wired interface* • *Section 3.2: Contactless interface* • *Section 7.6.23: Extended Get System Info* • *Section 7.6.30: Manage GPO* • *Section 9.1: Maximum ratings* • *Section 9.2: I²C parameters* • *Section 9.3: GPO characteristics* • *Section 9.4: RF electrical parameters* • *Table 256. RF characteristics*
06-Feb-2023	6	Updated *Table 164. Extended Get System Info request format.*.
05-Oct-2023	7	Updated: • *Features* • *Table 256. RF characteristics* • *Section 10.2: TSSOP8 package information* • *Appendix B: I²C sequences*
30-Jul-2024	8	Updated document title, *Section 2.2.2: Low power down (LPD), Section 5.3: Interface arbitration, Section 6.4.2: I²C sequential write, Section 6.5.3: Sequential read access, Section 7.6.1: RF command code list, Section 7.6.9: Extended Write Single Block, Section 7.6.10: Lock Block, Section 7.6.11: Extended Lock Block, and Section 7.6.30: Manage GPO.I* Updated *Table 3. 12-pin package signal names, Table 17. FTM, Table 18. MB_CTRL_Dyn access, Table 19. MB_CTRL_Dyn, Table 26. RF modes summary, Table 27. RF modes configuration bits and effect on RF requests, Table 28. FIELD_CHANGE when RF is disabled, or in sleep, or off mode, Table 139. Extended Write Multiple Block request format, Table 197. GPOVAL, Table 251. I²C AC characteristics up to 85 °C, Table 252. I²C AC characteristics up to 125 °C, and footnotes of Table 103. Timing values.* Updated *Figure 7. Power-up sequence (no RF field, LPD pin tied to VSS, or package without LPD pin), Figure 17. RF_USER sequence, Figure 23. RF_WRITE sequence, Figure 26. EH delivery state diagram, Figure 35. Read mode sequences, Figure 44. Extended Write Single Block frame exchange, Figure 59. Get Multiple Blocks Security Status frame exchange , Figure 60. Extended Get Multiple Blocks Security Status frame exchange, and Figure 64. Write Dynamic Configuration frame exchange*. Minor text edits across the whole document.
17-Nov-2026	9	Updated:

Date	Revision	Changes
		• Section 2.2.2: Low power down (LPD) • Table 4 and Table 5, Table 6 and Table 7.

Contents

1 **Description** . . . 3
1.1 Block diagram . . . 3
1.2 Packaging . . . 4
2 **Signal descriptions** . . . 7
2.1 Serial link (SCL, SDA) . . . 7
2.1.1 Serial clock (SCL) . . . 7
2.1.2 Serial data (SDA) . . . 7
2.2 Power control (V_{CC}, LPD, V_{SS}) . . . 7
2.2.1 Supply voltage (V_{CC}) . . . 7
2.2.2 Low power down (LPD) . . . 7
2.2.3 Ground (V_{SS}) . . . 7
2.3 RF link (AC0 AC1) . . . 8
2.3.1 Antenna coil (AC0, AC1) . . . 8
2.4 Process control (GPO, V_{DCG}) . . . 8
2.4.1 Driver supply voltage (V_{DCG}) . . . 8
2.4.2 General purpose output (GPO) . . . 8
2.5 Energy harvesting analog output (V_EH) . . . 8
3 **Power management** . . . 9
3.1 Wired interface . . . 9
3.2 Contactless interface . . . 10
4 **Memory management** . . . 11
4.1 Memory organization overview . . . 11
4.2 User memory . . . 12
4.2.1 User memory areas . . . 14
4.3 System configuration area . . . 18
4.4 Dynamic configuration . . . 20
4.5 Fast transfer mode mailbox . . . 21
5 **ST25DVxxKC specific features** . . . 22
5.1 Fast transfer mode (FTM) . . . 23
5.1.1 Fast transfer mode registers . . . 23
5.1.2 Fast transfer mode usage . . . 24
5.2 RF management feature . . . 28
5.2.1 RF management registers . . . 28
5.2.2 RF management . . . 29

5.3 Interface arbitration . . . 31
5.3.1 I²C priority . . . 32
5.4 GPO . . . 33
5.4.1 Interrupt capabilities on RF events . . . 33
5.4.2 Interrupt capabilities on I²C events . . . 42
5.4.3 GPO and power supply . . . 43
5.4.4 GPO registers . . . 44
5.4.5 Configuring GPO . . . 47
5.5 Energy harvesting (EH) . . . 48
5.5.1 Energy harvesting registers . . . 48
5.5.2 Energy harvesting description . . . 49
5.5.3 EH delivery state diagram . . . 49
5.5.4 EH delivery sequence . . . 51
5.6 Data protection . . . 51
5.6.1 Data protection registers . . . 51
5.6.2 Passwords and security sessions . . . 58
5.6.3 User memory protection . . . 61
5.6.4 System memory protection . . . 62
5.7 Device parameter registers . . . 63
6 I²C operation . . . 67
6.1 I²C protocol . . . 67
6.1.1 Start condition . . . 67
6.1.2 Stop condition . . . 68
6.1.3 Acknowledge bit (ACK) . . . 68
6.1.4 Data input . . . 68
6.2 I²C timeout . . . 68
6.2.1 I²C timeout on Start condition . . . 68
6.2.2 I²C timeout on clock period . . . 68
6.3 Device addressing . . . 69
6.4 I²C write operations . . . 70
6.4.1 I²C byte write . . . 70
6.4.2 I²C sequential write . . . 71
6.4.3 Minimizing system delays by polling on ACK . . . 72
6.5 I²C read operations . . . 73
6.5.1 Random address read . . . 74
6.5.2 Current address read . . . 74
6.5.3 Sequential read access . . . 75

6.5.4 Acknowledge in read mode . . . 75
6.6 I²C password management . . . 75
6.6.1 I²C present password command description . . . 75
6.6.2 I²C write password command description . . . 76
7 RF operation . . . 77
7.1 RF communication . . . 77
7.1.1 Access to a ISO/IEC 15693 device . . . 77
7.2 RF communication and energy harvesting . . . 77
7.3 Fast transfer mode mailbox access in RF . . . 77
7.4 RF protocol description . . . 78
7.4.1 Protocol description . . . 78
7.4.2 States referring to RF protocol . . . 78
7.4.3 Modes . . . 79
7.4.4 Request format . . . 80
7.4.5 Request flags . . . 80
7.4.6 Response format . . . 82
7.4.7 Response flags . . . 82
7.4.8 Response and error code . . . 83
7.5 Timing definition . . . 83
7.6 RF commands . . . 85
7.6.1 RF command code list . . . 85
7.6.2 Command codes list . . . 86
7.6.3 General command rules . . . 86
7.6.4 Inventory . . . 87
7.6.5 Stay Quiet . . . 87
7.6.6 Read Single Block . . . 88
7.6.7 Extended Read Single Block . . . 89
7.6.8 Write Single Block . . . 90
7.6.9 Extended Write Single Block . . . 92
7.6.10 Lock Block . . . 93
7.6.11 Extended Lock Block . . . 94
7.6.12 Read Multiple Blocks . . . 95
7.6.13 Extended Read Multiple Blocks . . . 96
7.6.14 Write Multiple Blocks . . . 98
7.6.15 Extended Write Multiple Blocks . . . 99
7.6.16 Select . . . 100
7.6.17 Reset to Ready . . . 101
7.6.18 Write AFI . . . 102

7.6.19 Lock AFI . . . 103
7.6.20 Write DSFID . . . 104
7.6.21 Lock DSFID . . . 105
7.6.22 Get System Info . . . 107
7.6.23 Extended Get System Info . . . 108
7.6.24 Get Multiple Blocks Security Status . . . 112
7.6.25 Extended Get Multiple Blocks Security Status . . . 113
7.6.26 Read Configuration . . . 115
7.6.27 Write Configuration . . . 115
7.6.28 Read Dynamic Configuration . . . 117
7.6.29 Write Dynamic Configuration . . . 118
7.6.30 Manage GPO . . . 119
7.6.31 Write Message . . . 120
7.6.32 Read Message Length . . . 121
7.6.33 Read Message . . . 122
7.6.34 Fast Read Message . . . 123
7.6.35 Write Password . . . 124
7.6.36 Present Password . . . 126
7.6.37 Fast Read Single Block . . . 127
7.6.38 Fast Extended Read Single Block . . . 128
7.6.39 Fast Read Multiple Blocks . . . 129
7.6.40 Fast Extended Read Multiple Blocks . . . 131
7.6.41 Fast Write Message . . . 132
7.6.42 Fast Read Message Length . . . 134
7.6.43 Fast Read Dynamic Configuration . . . 135
7.6.44 Fast Write Dynamic Configuration . . . 136

8 Unique identifier (UID) . . . 138

9 Device parameters . . . 139

9.1 Maximum ratings . . . 139
9.2 I²C parameters . . . 140
9.3 GPO characteristics . . . 147
9.4 RF electrical parameters . . . 148
9.5 Thermal characteristics . . . 149

10 Package information . . . 150

10.1 SO8N package information . . . 150
10.2 TSSOP8 package information . . . 152
10.3 UFDFN8 package information . . . 154

10.4 WLCSP10 package information . . . 156
10.5 UFDFPN12 package information . . . 158
11 Ordering information . . . 159
Appendix A Bit representation and coding for fast commands . . . 160
A.1 Bit coding using one subcarrier . . . 160
A.1.1 High data rate . . . 160
A.1.2 Low data rate . . . 160
A.2 ST25DVxxKC to VCD frames . . . 161
A.3 SOF when using one subcarrier . . . 161
A.3.1 High data rate . . . 161
A.3.2 Low data rate . . . 161
A.4 EOF when using one subcarrier . . . 162
A.4.1 High data rate . . . 162
A.4.2 Low data rate . . . 162
Appendix B I²C sequences . . . 163
B.1 Device select codes . . . 163
B.2 I²C Byte writing and polling . . . 163
B.2.1 I²C byte write in user memory . . . 163
B.2.2 I²C byte writing in dynamic registers and polling . . . 165
B.2.3 I²C byte write in mailbox and polling . . . 166
B.2.4 I²C byte write and polling in system memory . . . 167
B.3 I²C sequential writing and polling . . . 168
B.3.1 I²C sequential write in user memory and polling . . . 168
B.3.2 I²C sequential write in mailbox and polling . . . 170
B.4 I²C read current address . . . 171
B.4.1 I²C current address read in user memory . . . 171
B.5 I²C random address read . . . 172
B.5.1 I²C random address read in user memory . . . 172
B.5.2 I²C random address read in system memory . . . 173
B.5.3 I²C random address read in dynamic registers . . . 173
B.6 I²C sequential read . . . 174
B.6.1 I²C sequential read in user memory . . . 174
B.6.2 I²C sequential read in system memory . . . 177
B.6.3 I²C sequential read in dynamic registers . . . 179
B.6.4 I²C sequential read in mailbox . . . 181
B.7 I²C password relative sequences . . . 183
B.7.1 I²C write password . . . 183

B.7.2 I²C present password 185
Revision history 186

List of tables

Table 1. 8-pin packages signal names . . . 4
Table 2. 10-ball package signal names . . . 5
Table 3. 12-pin package signal names . . . 6
Table 4. User memory as seen by RF and by I^2C . . . 13
Table 5. Maximum user memory block and byte addresses and $ENDA_i$ value . . . 14
Table 6. Areas and limit calculation from $ENDA_i$ registers . . . 15
Table 7. ENDA1 access . . . 16
Table 8. ENDA1 . . . 17
Table 9. ENDA2 access . . . 17
Table 10. ENDA2 . . . 17
Table 11. ENDA3 access . . . 17
Table 12. ENDA3 . . . 17
Table 13. System configuration memory map . . . 18
Table 14. Dynamic registers memory map . . . 20
Table 15. Fast transfer mode mailbox memory map . . . 21
Table 16. FTM access . . . 23
Table 17. FTM . . . 23
Table 18. MB_CTRL_Dyn access . . . 23
Table 19. MB_CTRL_Dyn . . . 24
Table 20. MB_LEN_Dyn access . . . 24
Table 21. MB_LEN_Dyn . . . 24
Table 22. RF_MNGT access . . . 28
Table 23. RF_MNGT . . . 28
Table 24. RF_MNGT_Dyn access . . . 28
Table 25. RF_MNGT_Dyn . . . 28
Table 26. RF modes summary . . . 29
Table 27. RF modes configuration bits and effect on RF requests . . . 30
Table 28. FIELD_CHANGE when RF is disabled, or in sleep, or off mode . . . 37
Table 29. GPO interrupt capabilities according to RF field and V_{CC} . . . 44
Table 30. GPO1 access . . . 44
Table 31. GPO1 . . . 44
Table 32. GPO2 access . . . 45
Table 33. GPO2 . . . 45
Table 34. GPO_CTRL_Dyn access . . . 45
Table 35. GPO_CTRL_Dyn . . . 45
Table 36. IT_STS_Dyn access . . . 46
Table 37. IT_STS_Dyn . . . 46
Table 38. Enabling or disabling GPO interruptions . . . 47
Table 39. EH_MODE access . . . 48
Table 40. EH_MODE . . . 48
Table 41. EH_CTRL_Dyn access . . . 48
Table 42. EH_CTRL_Dyn . . . 48
Table 43. Energy harvesting at power-up . . . 49
Table 44. RFA1SS access . . . 51
Table 45. RFA1SS . . . 52
Table 46. RFA2SS access . . . 52
Table 47. RFA2SS . . . 52
Table 48. RFA3SS access . . . 52
Table 49. RFA3SS . . . 53
Table 50. RFA4SS access . . . 53
Table 51. RFA4SS . . . 53
Table 52. I2CSS access . . . 53

Table 53. I2CSS . . . 54
Table 54. LOCK_CCFILE access . . . 55
Table 55. LOCK_CCFILE . . . 55
Table 56. LOCK_CFG access . . . 55
Table 57. LOCK_CFG . . . 55
Table 58. I2C_PWD access . . . 56
Table 59. I2C_PWD . . . 56
Table 60. RF_PWD_0 access . . . 56
Table 61. RF_PWD_0 . . . 56
Table 62. RF_PWD_1 access . . . 57
Table 63. RF_PWD_1 . . . 57
Table 64. RF_PWD_2 access . . . 57
Table 65. RF_PWD_2 . . . 57
Table 66. RF_PWD_3 access . . . 58
Table 67. RF_PWD_3 . . . 58
Table 68. I2C_SSO_Dyn access . . . 58
Table 69. I2C_SSO_Dyn . . . 58
Table 70. Security session type . . . 59
Table 71. LOCK_DSFID access . . . 63
Table 72. LOCK_DSFID . . . 63
Table 73. LOCK_AFI access . . . 63
Table 74. LOCK_AFI . . . 63
Table 75. DSFID access . . . 64
Table 76. DSFID . . . 64
Table 77. AFI access . . . 64
Table 78. AFI . . . 64
Table 79. MEM_SIZE access . . . 64
Table 80. MEM_SIZE . . . 65
Table 81. BLK_SIZE access . . . 65
Table 82. BLK_SIZE . . . 65
Table 83. IC_REF access . . . 65
Table 84. IC_REF . . . 65
Table 85. UID access . . . 66
Table 86. UID . . . 66
Table 87. IC_REV access . . . 66
Table 88. IC_REV . . . 66
Table 89. Device select code . . . 69
Table 90. I2C_CFG access . . . 69
Table 91. I2C_CFG . . . 69
Table 92. Operating modes . . . 70
Table 93. Address most significant byte . . . 70
Table 94. Address least significant byte . . . 70
Table 95. Response dependance upon Request_flags . . . 79
Table 96. General request format . . . 80
Table 97. Definition of request flags 1 to 4 . . . 80
Table 98. Request flags 5 to 8 when Inventory_flag (bit 3) = 0 . . . 81
Table 99. Request flags 5 to 8 when Inventory_flag (bit 3) = 1 . . . 81
Table 100. General response format . . . 82
Table 101. Definitions of response flags 1 to 8 . . . 82
Table 102. Response error code definition . . . 83
Table 103. Timing values . . . 84
Table 104. Command codes . . . 86
Table 105. Inventory request format . . . 87
Table 106. Inventory response format . . . 87
Table 107. Stay Quiet request format . . . 88

Table 108. Read Single Block request format . . . 88
Table 109. Read Single Block response format when Error_flag is not set . . . 88
Table 110. Block security status . . . 88
Table 111. Read Single Block response format when Error_flag is set . . . 89
Table 112. Extended Read Single Block request format . . . 89
Table 113. Extended Read Single Block response format when Error_flag is not set . . . 89
Table 114. Block security status . . . 90
Table 115. Extended Read Single Block response format when Error_flag is set . . . 90
Table 116. Write Single Block request format . . . 90
Table 117. Write Single Block response format when Error_flag is not set . . . 91
Table 118. Write Single Block response format when Error_flag is set . . . 91
Table 119. Extended Write Single Block request format . . . 92
Table 120. Extended Write Single Block response format when Error_flag is not set . . . 92
Table 121. Extended Write Single Block response format when Error_flag is set . . . 92
Table 122. Lock block request format . . . 93
Table 123. Lock block response format when Error_flag is not set . . . 93
Table 124. Lock block response format when Error_flag is set . . . 93
Table 125. Extended Lock Block request format . . . 94
Table 126. Extended Lock Block response format when Error_flag is not set . . . 94
Table 127. Extended Lock Block response format when Error_flag is set . . . 95
Table 128. Read Multiple Blocks request format . . . 95
Table 129. Read Multiple Blocks response format when Error_flag is not set . . . 96
Table 130. Block security status . . . 96
Table 131. Read Multiple Blocks response format when Error_flag is set . . . 96
Table 132. Extended Read Multiple Blocks request format . . . 97
Table 133. Extended Read Multiple Blocks response format when Error_flag is not set . . . 97
Table 134. Block security status . . . 97
Table 135. Extended Read Multiple Blocks response format when Error_flag is set . . . 97
Table 136. Write Multiple Blocks request format . . . 98
Table 137. Write Multiple Blocks response format when Error_flag is not set . . . 98
Table 138. Write Multiple Blocks response format when Error_flag is set . . . 99
Table 139. Extended Write Multiple Block request format . . . 99
Table 140. Extended Write Multiple Blocks response format when Error_flag is not set . . . 100
Table 141. Extended Write Multiple Blocks response format when Error_flag is set . . . 100
Table 142. Select request format . . . 101
Table 143. Select response format when Error_flag is not set . . . 101
Table 144. Select response format when Error_flag is set . . . 101
Table 145. Reset to Ready request format . . . 101
Table 146. Reset to Ready response format when Error_flag is not set . . . 102
Table 147. Reset to Ready response format when Error_flag is set . . . 102
Table 148. Write AFI request format . . . 102
Table 149. Write AFI response format when Error_flag is not set . . . 103
Table 150. Write AFI response format when Error_flag is set . . . 103
Table 151. Lock AFI request format . . . 103
Table 152. Lock AFI response format when Error_flag is not set . . . 104
Table 153. Lock AFI response format when Error_flag is set . . . 104
Table 154. Write DSFID request format . . . 104
Table 155. Write DSFID response format when Error_flag is not set . . . 105
Table 156. Write DSFID response format when Error_flag is set . . . 105
Table 157. Lock DSFID request format . . . 106
Table 158. Lock DSFID response format when Error_flag is not set . . . 106
Table 159. Lock DSFID response format when Error_flag is set . . . 106
Table 160. Get System Info request format . . . 107
Table 161. Get System Info response format when Error_flag is not set . . . 107
Table 162. Memory size . . . 107

Table 163. Get System Info response format when Error_flag is set . . . 107
Table 164. Extended Get System Info request format . . . 108
Table 165. Parameter request list . . . 108
Table 166. Extended Get System Info response format when Error_flag is not set . . . 109
Table 167. Response information flag . . . 109
Table 168. Response other fields: VICC memory size . . . 109
Table 169. Response other fields: ICRef . . . 110
Table 170. Response other fields: VICC command list . . . 110
Table 171. Response other fields: VICC command list - Byte 1 . . . 110
Table 172. Response other fields: VICC command list - Byte 2 . . . 110
Table 173. Response other fields: VICC command list - Byte 3 . . . 111
Table 174. Response other fields: VICC command list - Byte 4 . . . 111
Table 175. Extended Get System Info response format when Error_flag is set . . . 111
Table 176. Get Multiple Blocks Security Status request format . . . 112
Table 177. Get Multiple Blocks Security Status response format when Error_flag is not set . . . 112
Table 178. Block security status . . . 112
Table 179. Get Multiple Blocks Security Status response format when Error_flag is set . . . 112
Table 180. Extended Get Multiple Blocks Security Status request format . . . 113
Table 181. Extended Get Multiple Blocks Security Status response format when Error_flags is not set . . . 113
Table 182. Block security status . . . 113
Table 183. Extended Get Multiple Blocks Security Status response format when Error_flag is set . . . 114
Table 184. Read Configuration request format . . . 115
Table 185. Read Configuration response format when Error_flag is not set . . . 115
Table 186. Read Configuration response format when Error_flag is set . . . 115
Table 187. Write Configuration request format . . . 116
Table 188. Write Configuration response format when Error_flag is not set . . . 116
Table 189. Write configuration response format when Error_flag is set . . . 116
Table 190. Read Dynamic Configuration request format . . . 117
Table 191. Read Dynamic Configuration response format when Error_flag is not set . . . 117
Table 192. Read Dynamic Configuration response format when Error_flag is set . . . 117
Table 193. Write Dynamic Configuration request format . . . 118
Table 194. Write Dynamic Configuration response format when Error_flag is not set . . . 118
Table 195. Write Dynamic Configuration response format when Error_flag is set . . . 118
Table 196. Manage GPO request format . . . 119
Table 197. GPOVAL . . . 119
Table 198. Manage GPO response format when Error_flag is not set . . . 120
Table 199. ManageGPO response format when Error_flag is set . . . 120
Table 200. Write Message request format . . . 120
Table 201. Write Message response format when Error_flag is not set . . . 121
Table 202. Write Message response format when Error_flag is set . . . 121
Table 203. Read Message Length request format . . . 121
Table 204. Read Message Length response format when Error_flag is not set . . . 122
Table 205. Read Message Length response format when Error_flag is set . . . 122
Table 206. Read Message request format . . . 122
Table 207. Read Message response format when Error_flag is not set . . . 123
Table 208. Fast Read Message request format . . . 123
Table 209. Fast Read Message response format when Error_flag is NOT set . . . 124
Table 210. Write Password request format . . . 125
Table 211. Write Password response format when Error_flag is not set . . . 125
Table 212. Write Password response format when Error_flag is set . . . 125
Table 213. Present Password request format . . . 126
Table 214. Present Password response format when Error_flag is not set . . . 126
Table 215. Present Password response format when Error_flag is set . . . 126
Table 216. Fast Read Single Block request format . . . 127
Table 217. Fast Read Single Block response format when Error_flag is not set . . . 127

Table 218. Block security status. . . . 127
Table 219. Fast Read Single Block response format when Error_flag is set . . . 128
Table 220. Fast Extended Read Single Block request format . . . 128
Table 221. Fast Extended Read Single Block response format when Error_flag is not set . . . 129
Table 222. Block security status. . . . 129
Table 223. Fast Extended Read Single Block response format when Error_flag is set . . . 129
Table 224. Fast Read Multiple Blocks request format . . . 130
Table 225. Fast Read Multiple Blocks response format when Error_flag is not set . . . 130
Table 226. Block security status if Option_flag is set . . . 130
Table 227. Fast Read Multiple Blocks response format when Error_flag is set . . . 130
Table 228. Fast Extended Read Multiple Blocks request format . . . 131
Table 229. Fast Extended Read Multiple Blocks response format when Error_flag is not set . . . 131
Table 230. Block security status if Option_flag is set . . . 131
Table 231. Fast Extended Read Multiple Blocks response format when Error_flag is set . . . 132
Table 232. Fast Write Message request format . . . 132
Table 233. Fast Write Message response format when Error_flag is not set. . . . 133
Table 234. Fast Write Message response format when Error_flag is set . . . 133
Table 235. Fast Read Message Length request format . . . 134
Table 236. Fast Read Message Length response format when Error_flag is not set . . . 134
Table 237. Fast Read Message Length response format when Error_flag is set . . . 134
Table 238. Fast Read Dynamic Configuration request format. . . . 135
Table 239. Fast Read Dynamic Configuration response format when Error_flag is not set . . . 135
Table 240. Fast Read Dynamic Configuration response format when Error_flag is set . . . 135
Table 241. Fast Write Dynamic Configuration request format. . . . 136
Table 242. Fast Write Dynamic Configuration response format when Error_flag is not set . . . 136
Table 243. Fast Write Dynamic Configuration response format when Error_flag is set. . . . 136
Table 244. UID format . . . 138
Table 245. Absolute maximum ratings . . . 139
Table 246. I²C operating conditions . . . 140
Table 247. AC test measurement conditions . . . 140
Table 248. Input parameters . . . 140
Table 249. I²C DC characteristics up to 85 °C . . . 141
Table 250. I²C DC characteristics up to 125 °C . . . 142
Table 251. I²C AC characteristics up to 85 °C . . . 144
Table 252. I²C AC characteristics up to 125 °C . . . 145
Table 253. GPO DC characteristics up to 85 °C . . . 147
Table 254. GPO DC characteristics up to 125 °C . . . 147
Table 255. GPO AC characteristics . . . 147
Table 256. RF characteristics . . . 148
Table 257. Operating conditions . . . 149
Table 258. Thermal characteristics. . . . 149
Table 259. SO8N - Mechanical data . . . 151
Table 260. TSSOP8 - Mechanical data . . . 152
Table 261. UFDFN8 - Mechanical data . . . 155
Table 262. WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale mechanical data . . . 156
Table 263. WLCSP10 recommended PCB design rules . . . 157
Table 264. UFDFPN12 - Mechanical data . . . 158
Table 265. Ordering information scheme. . . . 159
Table 266. Device select usage . . . 163
Table 267. Byte write in user memory when write operation is allowed . . . 163
Table 268. Polling during programming after byte writing in user memory . . . 164
Table 269. Byte Write in user memory when write operation is not allowed . . . 164
Table 270. Byte Write in dynamic register (if not read only) . . . 165
Table 271. Polling during programming after byte write in dynamic register . . . 165
Table 272. Byte Write in dynamic register if read only . . . 165

Table 273. Byte Write in mailbox when mailbox is free from RF message and fast transfer mode is activated . . . 166
Table 274. Byte Write in mailbox when mailbox is not free from RF message fast transfer mode is not activated . . . 166
Table 275. Byte write if I²C security session is open and register is not RO . . . 167
Table 276. Polling during programming after byte write if I²C security session is open and register is not RO . . . 167
Table 277. Byte write if I²C security session is closed or register is RO . . . 168
Table 278. Sequential write user memory when write operation is allowed and all bytes belong to same area . . . 168
Table 279. Polling during programming after sequential write in user memory when write operation allowed and all bytes belong to the same area . . . 169
Table 280. Sequential write in user memory when write operation is allowed and crossing over area border . . . 169
Table 281. Polling during programming after sequential write in user memory when write operation is allowed and crossing over area border . . . 170
Table 282. Sequential write in mailbox when mailbox is free from RF message and fast transfer mode is activated . . . 170
Table 283. Polling during programming after sequential write in the mailbox . . . 170
Table 284. Current byte read in user memory if read operation is allowed (depending on area protection and RF user security session) . . . 171
Table 285. Current Read in user memory if read operation is not allowed (depending on area protection and RF user security session) . . . 171
Table 286. Random byte read in user memory if read operation is allowed (depending on area protection and RF user security session) . . . 172
Table 287. Random byte read in user memory if operation is not allowed (depending on area protection and RF user security) . . . 172
Table 288. Byte read system memory (Static register or I²C password after a valid Present I²C password) . . . 173
Table 289. Random byte read in dynamic registers . . . 173
Table 290. Sequential read user memory if read operation is allowed (depending on area protection and RF user security session) and all bytes belong to the same area . . . 174
Table 291. Sequential read user memory if read operation allowed (depending on area protection and RF user security session) but crossing area border . . . 175
Table 292. Sequential Read user memory if read operation is allowed (depending on area protection and RF user security session) . . . 176
Table 293. Sequential in read system memory (I²C security session open if reading I2C_PWD) . . . 177
Table 294. Sequential Read system memory when access is not granted (I²C password I2C_PWD) . . . 178
Table 295. Sequential read in dynamic register . . . 179
Table 296. Sequential read in dynamic register and mailbox continuously if fast transfer mode is activated . . . 180
Table 297. Sequential in mailbox if fast transfer mode is activated . . . 181
Table 298. Sequential read in mailbox if fast transfer mode is not activated . . . 182
Table 299. Write password when the I²C security session is already open and fast transfer mode is not activated . . . 183
Table 300. Write password when the I²C security session is not open or fast transfer mode activated . . . 184
Table 301. Present password (whatever status of I²C security session or fast transfer mode) . . . 185
Table 302. Document revision history . . . 186

List of figures

Figure 1. Block diagram . . . 3
Figure 2. SO8N package connections . . . 4
Figure 3. TSSOP8 package connections . . . 4
Figure 4. UFDFN8 package connections . . . 5
Figure 5. WLCSP10 package connections . . . 5
Figure 6. UFDFPN12 package connections . . . 6
Figure 7. Power-up sequence (no RF field, LPD pin tied to V_{SS}, or package without LPD pin) . . . 9
Figure 8. RF power-up sequence (no DC supply) . . . 10
Figure 9. Memory organization . . . 12
Figure 10. User memory areas . . . 14
Figure 11. RF to I^2C fast transfer mode operation . . . 25
Figure 12. I^2C to RF fast transfer mode operation . . . 25
Figure 13. Fast transfer mode mailbox access management . . . 27
Figure 14. Arbitration between RF and I^2C . . . 31
Figure 15. RFSwitchOff command . . . 32
Figure 16. RFSwitchOn command . . . 32
Figure 17. RF_USER sequence . . . 34
Figure 18. RF_ACTIVITY sequence . . . 35
Figure 19. RF_INTERRUPT sequence . . . 36
Figure 20. FIELD_CHANGE sequence . . . 37
Figure 21. RF_PUT_MSG sequence . . . 38
Figure 22. RF_GET_MSG sequence . . . 39
Figure 23. RF_WRITE sequence . . . 41
Figure 24. GPO/I2C_WRITE sequence . . . 42
Figure 25. GPO/I2C_RF_OFF sequence . . . 43
Figure 26. EH delivery state diagram . . . 50
Figure 27. ST25DVxxKC Energy Harvesting Delivery Sequence . . . 51
Figure 28. RF security sessions management . . . 60
Figure 29. I^2C security sessions management . . . 61
Figure 30. I²C bus protocol . . . 67
Figure 31. I²C timeout on Start condition . . . 68
Figure 32. Write mode sequences when write is not inhibited . . . 72
Figure 33. Write mode sequences when write is inhibited . . . 72
Figure 34. Write cycle polling flowchart using ACK . . . 73
Figure 35. Read mode sequences . . . 74
Figure 36. I²C present password sequence . . . 76
Figure 37. I²C write password sequence . . . 76
Figure 38. Protocol timing . . . 78
Figure 39. State transition diagram . . . 79
Figure 40. Stay Quiet frame exchange . . . 88
Figure 41. Read Single Block frame exchange . . . 89
Figure 42. Extended Read Single Block frame exchange . . . 90
Figure 43. Write Single Block frame exchange . . . 91
Figure 44. Extended Write Single Block frame exchange . . . 93
Figure 45. Lock Block frame exchange . . . 94
Figure 46. Extended Lock Block frame exchange . . . 95
Figure 47. Read Multiple Blocks frame exchange . . . 96
Figure 48. Extended Read Multiple Blocks frame exchange . . . 98
Figure 49. Write Multiple Blocks frame exchange . . . 99
Figure 50. Extended Write Multiple Blocks frame exchange . . . 100
Figure 51. Select frame exchange . . . 101
Figure 52. Reset to Ready frame exchange . . . 102

Figure 53. Write AFI frame exchange . . . 103
Figure 54. Lock AFI frame exchange . . . 104
Figure 55. Write DSFID frame exchange . . . 105
Figure 56. Lock DSFID frame exchange . . . 106
Figure 57. Get System Info frame exchange . . . 108
Figure 58. Extended Get System Info frame exchange . . . 111
Figure 59. Get Multiple Blocks Security Status frame exchange . . . 113
Figure 60. Extended Get Multiple Blocks Security Status frame exchange . . . 114
Figure 61. Read Configuration frame exchange . . . 115
Figure 62. Write Configuration exchange . . . 117
Figure 63. Read Dynamic Configuration frame exchange . . . 118
Figure 64. Write Dynamic Configuration frame exchange . . . 119
Figure 65. Manage GPO frame exchange . . . 120
Figure 66. Write Message frame exchange . . . 121
Figure 67. Read Message Length frame exchange . . . 122
Figure 68. Read Message frame exchange . . . 123
Figure 69. Fast Read Message frame exchange . . . 124
Figure 70. Write Password frame exchange . . . 126
Figure 71. Present Password frame exchange . . . 127
Figure 72. Fast Read Single Block frame exchange . . . 128
Figure 73. Fast Extended Read Single Block frame exchange . . . 129
Figure 74. Fast Read Multiple Blocks frame exchange . . . 131
Figure 75. Fast Extended Read Multiple Blocks frame exchange . . . 132
Figure 76. Fast Write Message frame exchange . . . 133
Figure 77. Fast Read Message Length frame exchange . . . 134
Figure 78. Fast Read Dynamic Configuration frame exchange . . . 135
Figure 79. Fast Write Dynamic Configuration frame exchange . . . 137
Figure 80. AC test measurement I/O waveform . . . 140
Figure 81. I²C AC waveforms . . . 146
Figure 82. I²C Fast mode (f_C = 1 MHz): maximum R_{bus} value versus bus parasitic capacitance (C_{bus}) . . . 146
Figure 83. ASK modulated signal . . . 149
Figure 84. SO8N - Outline . . . 150
Figure 85. TSSOP8 – Outline . . . 152
Figure 86. TSSOP8 – Footprint example . . . 153
Figure 87. UFDFN8 - Outline . . . 154
Figure 88. WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale package outline . . . 156
Figure 89. WLCSP - 10 balls, 1.649x1.483 mm, 0.4 mm pitch, wafer level chip scale recommended footprint . . . 157
Figure 90. UFDFPN12 - Outline . . . 158
Figure 91. Logic 0, high data rate, fast commands . . . 160
Figure 92. Logic 1, high data rate, fast commands . . . 160
Figure 93. Logic 0, low data rate, fast commands . . . 160
Figure 94. Logic 1, low data rate, fast commands . . . 161
Figure 95. Start of frame, high data rate, one subcarrier, fast commands . . . 161
Figure 96. Start of frame, low data rate, one subcarrier, fast commands . . . 161
Figure 97. End of frame, high data rate, one subcarrier, fast commands . . . 162
Figure 98. End of frame, low data rate, one subcarrier, fast commands . . . 162

IMPORTANT NOTICE – READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice.

In the event of any conflict between the provisions of this document and the provisions of any contractual arrangement in force between the purchasers and ST, the provisions of such contractual arrangement shall prevail.

The purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

The purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of the purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

If the purchasers identify an ST product that meets their functional and performance requirements but that is not designated for the purchasers' market segment, the purchasers shall contact ST for more information.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2026 STMicroelectronics – All rights reserved