



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

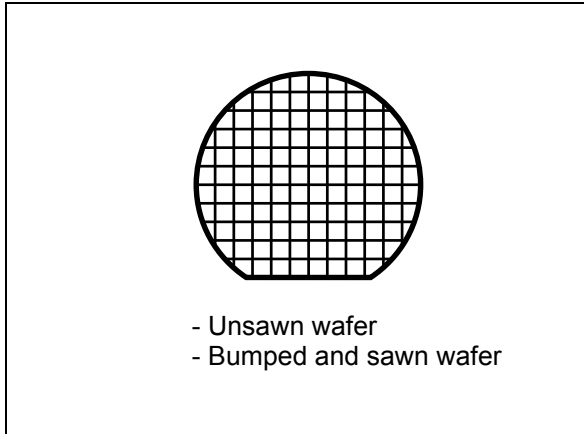
Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



**13.56 MHz short-range contactless memory chip
with 2048-bit EEPROM and anticollision functions**

Datasheet - production data

**Features**

- ISO 14443-2 Type B air interface compliant
- ISO 14443-3 Type B frame format compliant
- 13.56 MHz carrier frequency
- 847 kHz subcarrier frequency
- 106 Kbit/second data transfer
- 8 bit Chip_ID based anticollision system
- 2 count-down binary counters with automated anti-tearing protection
- 64-bit Unique Identifier
- 2048-bit EEPROM with write protect feature
- Read_block and Write_block (32 bits)
- Internal tuning capacitor: 68 pF
- 1 million erase/write cycles
- 40-year data retention
- Self-timed programming cycle
- 5 ms typical programming time

Contents

- 1 Description 7**
- 2 Signal description 9**
 - 2.1 AC1, AC0 9
- 3 Data transfer 10**
 - 3.1 Input data transfer from reader to ST25TB02K (request frame) 10
 - 3.1.1 Character transmission format for request frame 10
 - 3.1.2 Request start of frame 11
 - 3.1.3 Request end of frame 11
 - 3.2 Output data transfer from ST25TB02K to reader (answer frame) 12
 - 3.2.1 Character transmission format for answer frame 12
 - 3.2.2 Answer start of frame 12
 - 3.2.3 Answer end of frame 13
 - 3.3 Transmission frame 13
 - 3.4 CRC 14
- 4 Memory mapping 15**
 - 4.1 Resettable OTP area 16
 - 4.2 32-bit binary counters 17
 - 4.3 EEPROM area 18
 - 4.4 System area 18
 - 4.4.1 OTP_Lock_Reg 19
- 5 ST25TB02K operation 20**
- 6 ST25TB02K states 21**
 - 6.1 Power-off state 21
 - 6.2 Ready state 21
 - 6.3 Inventory state 21
 - 6.4 Selected state 21
 - 6.5 Deselected state 21
 - 6.6 Deactivated state 22

7	Anticollision	23
7.1	Description of an anticollision sequence	24
8	ST25TB02K commands	27
8.1	Initiate() command	28
8.2	Pcall16() command	29
8.3	Slot_marker(SN) command	30
8.4	Select(Chip_ID) command	31
8.5	Completion() command	32
8.6	Reset_to_inventory() command	33
8.7	Read_block(Addr) command	34
8.8	Write_block (Addr, Data) command	35
8.9	Get_UID() command	36
8.10	Power-on state	37
9	Maximum ratings	38
10	RF electrical parameters	39
11	Part numbering	42
Appendix A ISO-14443 Type B CRC calculation		43
Appendix B ST25TB02K command brief		44
Revision history		46

List of tables

Table 1.	Signal names	7
Table 2.	Bit description	11
Table 3.	ST25TB02K memory mapping	15
Table 4.	Resettable OTP area (addresses 0 to 4)	16
Table 5.	Binary counter (addresses 5 to 6)	17
Table 6.	EEPROM (addresses 7 to 63)	18
Table 7.	System area	18
Table 8.	Standard anticollision sequence	25
Table 9.	Example of an anticollision sequence	26
Table 10.	Command code	27
Table 11.	Absolute maximum ratings	38
Table 12.	Operating conditions	39
Table 13.	Electrical characteristics	39
Table 14.	RF characteristics	40
Table 15.	Ordering information scheme (bumped and sawn wafer)	42
Table 16.	Document revision history	46

List of figures

Figure 1.	Logic diagram	7
Figure 2.	Die floor plan and assembly options	8
Figure 3.	10% ASK modulation of the received wave	10
Figure 4.	ST25TB02K request frame character format	10
Figure 5.	Request start of frame	11
Figure 6.	Request end of frame	11
Figure 7.	Wave transmitted using BPSK subcarrier modulation	12
Figure 8.	Answer start of frame	12
Figure 9.	Answer end of frame.	13
Figure 10.	Example of a complete transmission frame	13
Figure 11.	CRC transmission rules	14
Figure 12.	Write_block update in Standard mode (binary format)	16
Figure 13.	Write_block update in Reload mode (binary format).	16
Figure 14.	Countdown example (binary format).	17
Figure 15.	State transition diagram	22
Figure 16.	ST25TB02K Chip_ID description	23
Figure 17.	Description of a possible anticollision sequence	24
Figure 18.	Initiate request format	28
Figure 19.	Initiate response format	28
Figure 20.	Initiate frame exchange between reader and ST25TB02K.	28
Figure 21.	Pcall16 request format	29
Figure 22.	Pcall16 response format.	29
Figure 23.	Pcall16 frame exchange between reader and ST25TB02K	29
Figure 24.	Slot_marker request format	30
Figure 25.	Slot_marker response format	30
Figure 26.	Slot_marker frame exchange between reader and ST25TB02K	30
Figure 27.	Select request format	31
Figure 28.	Select response format.	31
Figure 29.	Select frame exchange between reader and ST25TB02K	31
Figure 30.	Completion request format	32
Figure 31.	Completion response format.	32
Figure 32.	Completion frame exchange between reader and ST25TB02K.	32
Figure 33.	Reset_to_inventory request format.	33
Figure 34.	Reset_to_inventory response format	33
Figure 35.	Reset_to_inventory frame exchange between reader and ST25TB02K	33
Figure 36.	Read_block request format.	34
Figure 37.	Read_block response format	34
Figure 38.	Read_block frame exchange between reader and ST25TB02K	34
Figure 39.	Write_block request format.	35
Figure 40.	Write_block response format	35
Figure 41.	Write_block frame exchange between reader and ST25TB02K.	36
Figure 42.	Get_UID request format	36
Figure 43.	Get_UID response format.	36
Figure 44.	64-bit unique identifier of the ST25TB02K	37
Figure 45.	Get_UID frame exchange between reader and ST25TB02K	37
Figure 46.	ST25TB02K synchronous timing, transmit and receive	41
Figure 47.	Initiate frame exchange between reader and ST25TB02K.	44
Figure 48.	Pcall16 frame exchange between reader and ST25TB02K	44

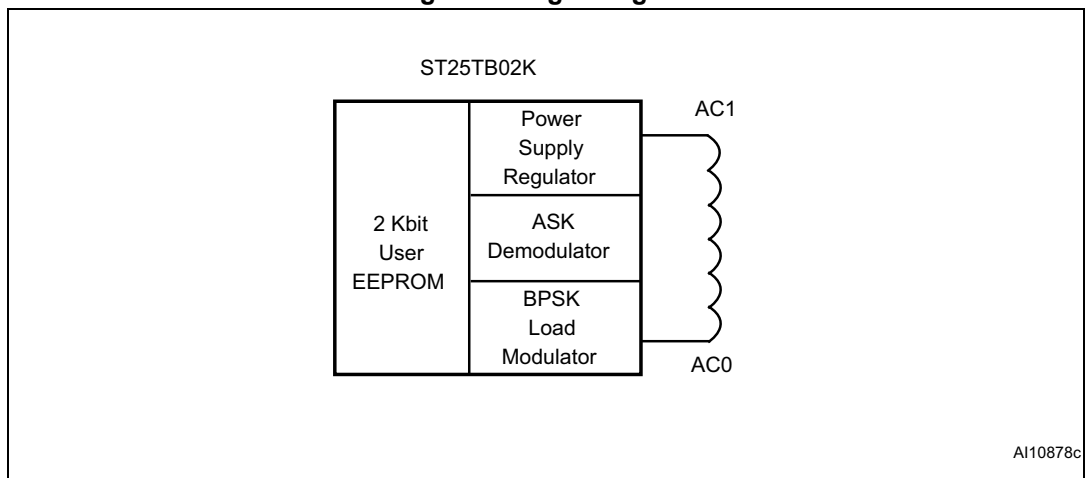
Figure 49.	Slot_marker frame exchange between reader and ST25TB02K	44
Figure 50.	Select frame exchange between reader and ST25TB02K	44
Figure 51.	Completion frame exchange between reader and ST25TB02K	44
Figure 52.	Reset_to_inventory frame exchange between reader and ST25TB02K	45
Figure 53.	Read_block frame exchange between reader and ST25TB02K	45
Figure 54.	Write_block frame exchange between reader and ST25TB02K	45
Figure 55.	Get_UID frame exchange between reader and ST25TB02K	45

1 Description

The ST25TB02K is a contactless memory, powered by an externally transmitted radio wave. It contains a 2048-bit user EEPROM. The memory is organized as 64 blocks of 32 bits. The ST25TB02K is accessed via the 13.56 MHz carrier. Incoming data are demodulated and decoded from the received amplitude shift keying (ASK) modulation signal and outgoing data are generated by load variation using bit phase shift keying (BPSK) coding of a 847 kHz sub-carrier. The received ASK wave is 10% modulated. The data transfer rate between the ST25TB02K and the reader is 106 kbit/s in both reception and emission modes.

The ST25TB02K follows the ISO 14443 - 2 Type B recommendation for the radio-frequency power and signal interface.

Figure 1. Logic diagram



The ST25TB02K is specifically designed for short range applications that need re-usable products. The ST25TB02K includes an anticollision mechanism that allows it to detect and select tags present at the same time within range of the reader. The anticollision is based on a probabilistic scanning method using slot markers.

Table 1. Signal names

Signal names	Description
AC1	Antenna coil
AC0	Antenna coil

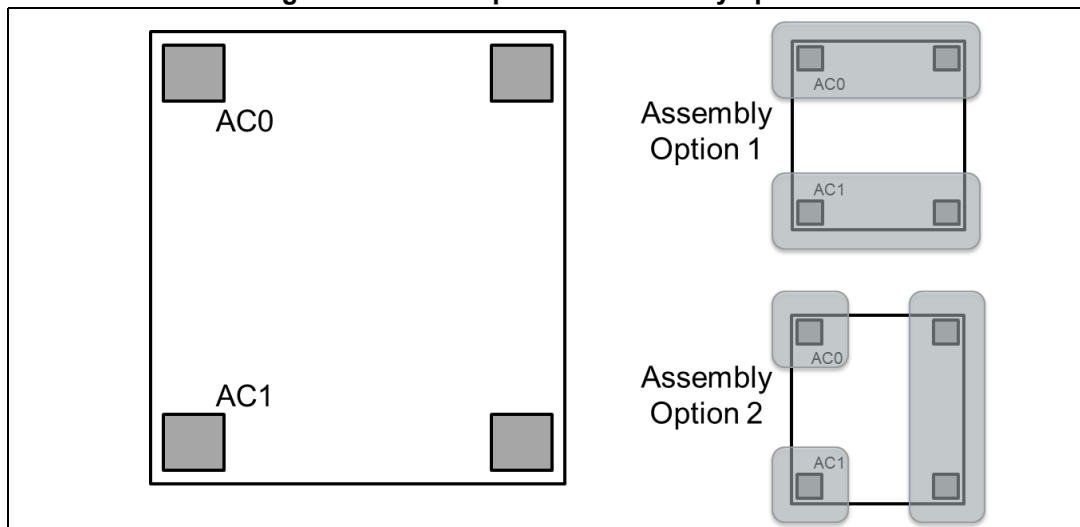
The ST25TB02K contactless EEPROM can be randomly read and written in block mode (each block containing 32 bits). The instruction set includes the following nine commands:

- Read_block
- Write_block
- Initiate
- Pcall16
- Slot_marker
- Select
- Completion
- Reset_to_inventory
- Get_UID

The ST25TB02K memory is organized in three areas, as described in [Table 3](#). The first area is a resettable OTP (one time programmable) area in which bits can only be switched from 1 to 0. Using a special command, it is possible to erase all bits of this area to 1. The second area provides two 32-bit binary counters which can only be decremented. The last area is the EEPROM memory. It is accessible by block of 32 bits and includes an auto-erase cycle during each Write_block command.

Die floor plan and physical options related to the die assembly are described in [Figure 2](#).

Figure 2. Die floor plan and assembly options



For the option 1 of the die assembly, the CTUN (referenced in [Table 13](#)) can increase from 0.5pF to 1pF. The option 2 of the die assembly is showing a tripod which can be used for physical stability, having no impact on CTUN parameter.

2 Signal description

2.1 AC1, AC0

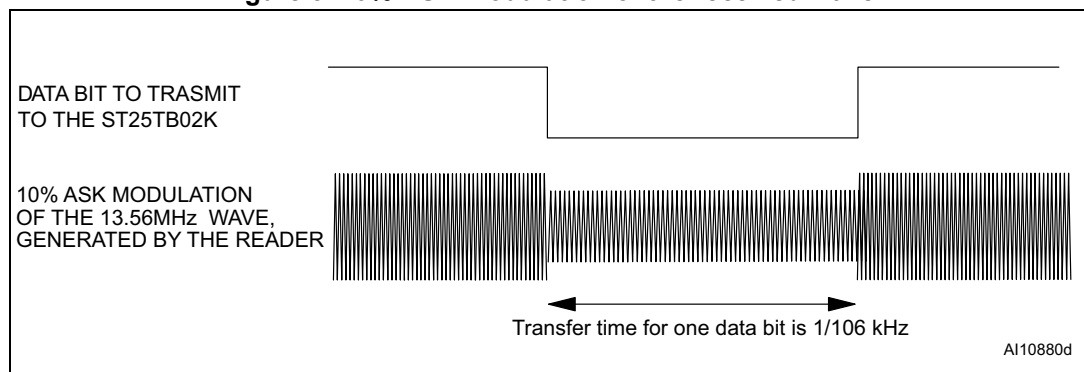
The pads for the Antenna Coil. AC1 and AC0 must be directly bonded to the antenna.

3 Data transfer

3.1 Input data transfer from reader to ST25TB02K (request frame)

The reader must generate a 13.56 MHz sinusoidal carrier frequency at its antenna, with enough energy to “remote-power” the memory. The energy received at the ST25TB02K’s antenna is transformed into a supply voltage by a regulator, and into data bits by the ASK demodulator. For the ST25TB02K to decode correctly the information it receives, the reader must 10% amplitude-modulate the 13.56 MHz wave before sending it to the ST25TB02K. This is represented in *Figure 3*. The data transfer rate is 106 Kbits/s.

Figure 3. 10% ASK modulation of the received wave



3.1.1 Character transmission format for request frame

The ST25TB02K transmits and receives data bytes as 10-bit characters, with the least significant bit (b_0) transmitted first, as shown in *Figure 4*. Each bit duration, an ETU (elementary time unit), is equal to 9.44 μ s (1/106 kHz).

These characters, framed by a start of frame (SOF) and an end of frame (EOF), are put together to form a command frame as shown in *Figure 10*. A frame includes an SOF, commands, addresses, data, a CRC and an EOF as defined in the ISO 14443-3 Type B Standard. If an error is detected during data transfer, the ST25TB02K does not execute the command, but it does not generate an error frame.

Figure 4. ST25TB02K request frame character format

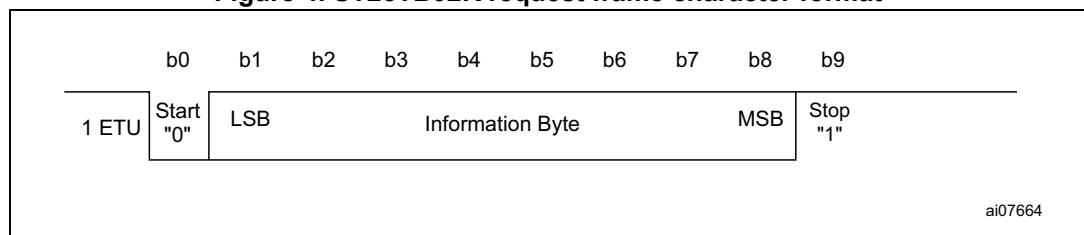


Table 2. Bit description

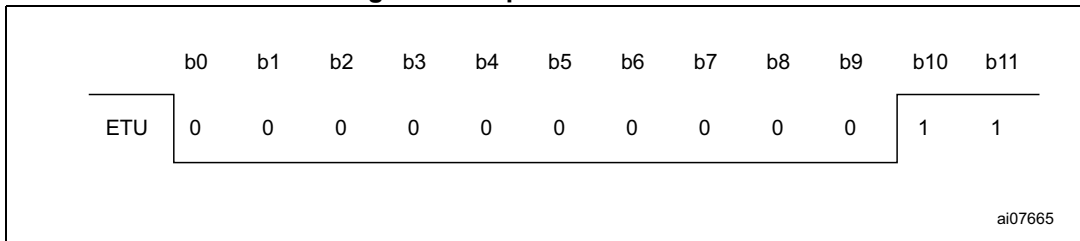
Bit	Description	Value
b ₀	Start bit used to synchronize the transmission	b ₀ = 0
b ₁ to b ₈	Information byte (command, address or data)	The information byte is sent with the least significant bit first
b ₉	Stop bit used to indicate the end of a character	b ₉ = 1

3.1.2 Request start of frame

The SOF described in *Figure 5* is composed of:

- one falling edge,
- followed by 10 ETUs at logic-0,
- followed by a single rising edge,
- followed by at least 2 ETUs (and at most 3) at logic-1.

Figure 5. Request start of frame

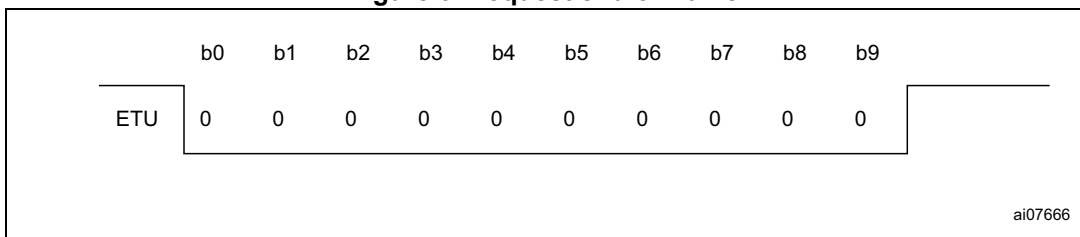


3.1.3 Request end of frame

The EOF shown in *Figure 6* is composed of:

- one falling edge,
- followed by 10 ETUs at logic-0,
- followed by a single rising edge.

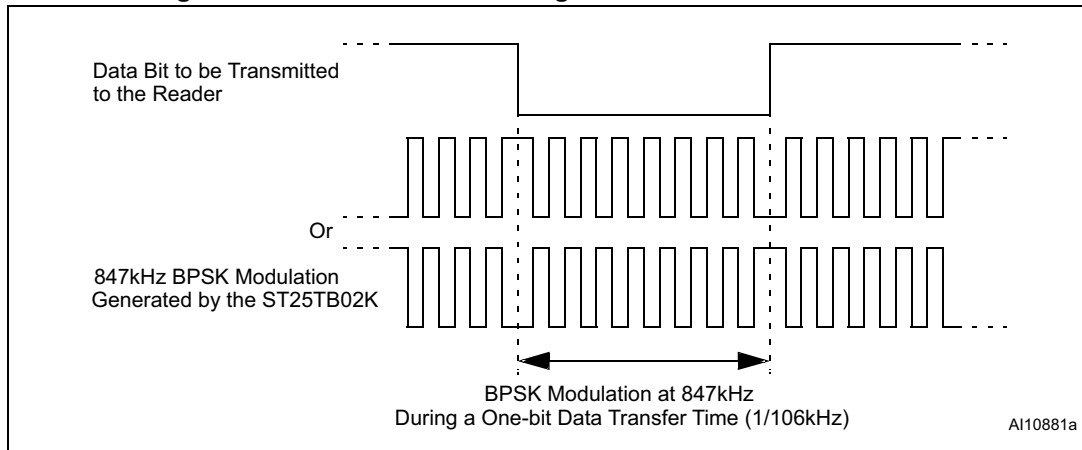
Figure 6. Request end of frame



3.2 Output data transfer from ST25TB02K to reader (answer frame)

The data bits issued by the ST25TB02K use back-scattering. Back-scattering is obtained by modifying the ST25TB02K current consumption at the antenna (load modulation). The load modulation causes a variation at the reader antenna by inductive coupling. With appropriate detector circuitry, the reader is able to pick up information from the ST25TB02K. To improve load-modulation detection, data is transmitted using a BPSK encoded, 847 kHz subcarrier frequency f_s as shown in *Figure 7*, and as specified in the ISO 14443-2 Type B standard.

Figure 7. Wave transmitted using BPSK subcarrier modulation



3.2.1 Character transmission format for answer frame

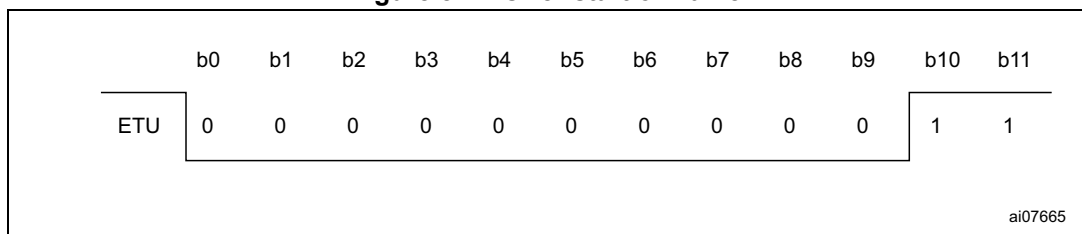
The character format is the same as for input data transfer (*Figure 4*). The transmitted frames are made up of an SOF, data, a CRC and an EOF (*Figure 10*). As with an input data transfer, if an error occurs, the reader does not issue an error code to the ST25TB02K, but it should be able to detect it and manage the situation. The data transfer rate is 106 Kbits/second.

3.2.2 Answer start of frame

The SOF described in *Figure 8* is composed of:

- followed by 10 ETUs at logic-0
- followed by 2 ETUs at logic-1

Figure 8. Answer start of frame

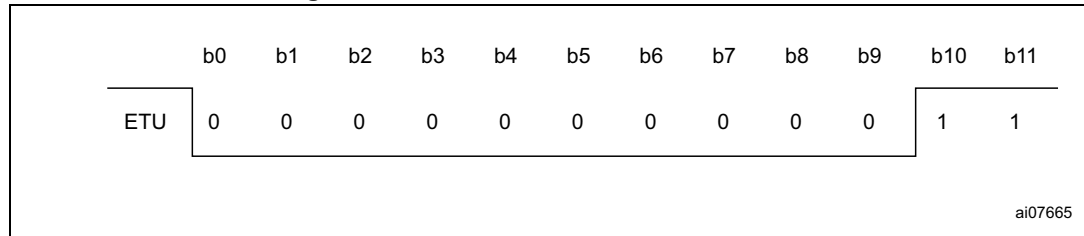


3.2.3 Answer end of frame

The EOF shown in *Figure 9* is composed of:

- followed by 10 ETUs at logic-0,
- followed by 2 ETUs at logic-1.

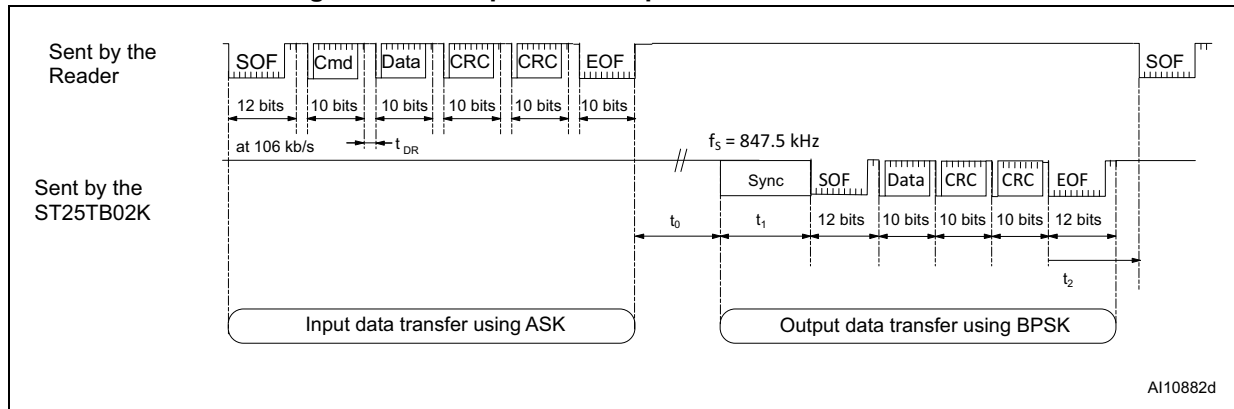
Figure 9. Answer end of frame



3.3 Transmission frame

Between the request data transfer and the answer data transfer, all ASK and BPSK modulations are suspended for a minimum time of $t_0 = 128/f_S$. This delay allows the reader to switch from Transmission to Reception mode. It is repeated after each frame. After t_0 , the 13.56 MHz carrier frequency is modulated by the ST25TB02K at 847 kHz for a period of $t_1 = 128/f_S$ to allow the reader to synchronize. After t_1 , the first phase transition generated by the ST25TB02K forms the start bit ('0') of the answer SOF. After the falling edge of the answer EOF, the reader waits a minimum time, t_2 , before sending a new request frame to the ST25TB02K.

Figure 10. Example of a complete transmission frame



3.4 CRC

The 16-bit CRC used by the ST25TB02K is generated in compliance with the ISO14443 Type B recommendation. For further information, please see [Appendix A](#). The initial register contents are all 1s: FFFFh.

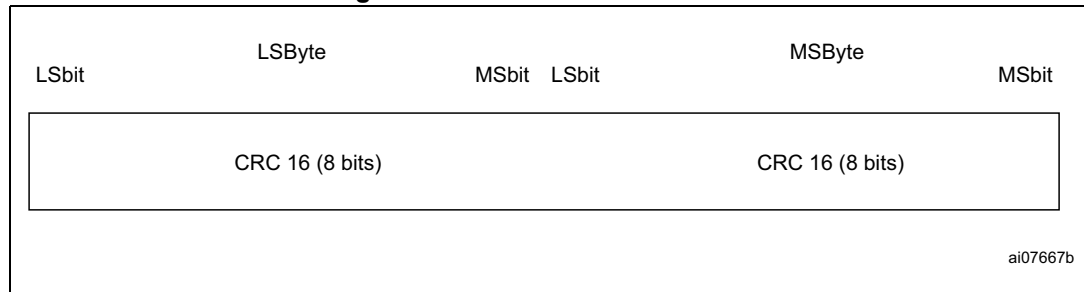
The two-byte CRC is present in every request and in every answer frame, before the EOF. The CRC is calculated on all the bytes between SOF (not included) and the CRC field.

Upon reception of a request from a reader, the ST25TB02K verifies that the CRC value is valid. If it is invalid, the ST25TB02K discards the frame and does not answer the reader.

Upon reception of an answer from the ST25TB02K, the reader should verify the validity of the CRC. In case of error, the actions to be taken are the reader designer's responsibility.

The CRC is transmitted with the least significant byte first and each byte is transmitted with the least significant bit first.

Figure 11. CRC transmission rules



4 Memory mapping

The ST25TB02K is organized as 64 blocks of 32 bits as shown in [Table 3](#). All blocks are accessible by the Read_block command. Depending on the write access, they can be updated by the Write_block command. A Write_block updates all the 32 bits of the block.

Table 3. ST25TB02K memory mapping

Block Address	32-bit block						Description
	MSB b31	b24 b23	b16	b15	b8 b7	LSB b0	
0	32-bit Boolean area						Resettable OTP bit
1	32-bit Boolean area						
2	32-bit Boolean area						
3	32-bit Boolean area						
4	32-bit Boolean area						
5	32 bits binary counter						Count down counter
6	32 bits binary counter						
7	User area						Lockable EEPROM
8	User area						
9	User area						
10	User area						
11	User area						
12	User area						
13	User area						
14	User area						
15	User area						
16	User area						EEPROM
...	User area						
63	User area						
255	OTP_Lock_Reg	ST Reserved					System OTP bits
UID0	64 bits UID area						ROM
UID1							

4.2 32-bit binary counters

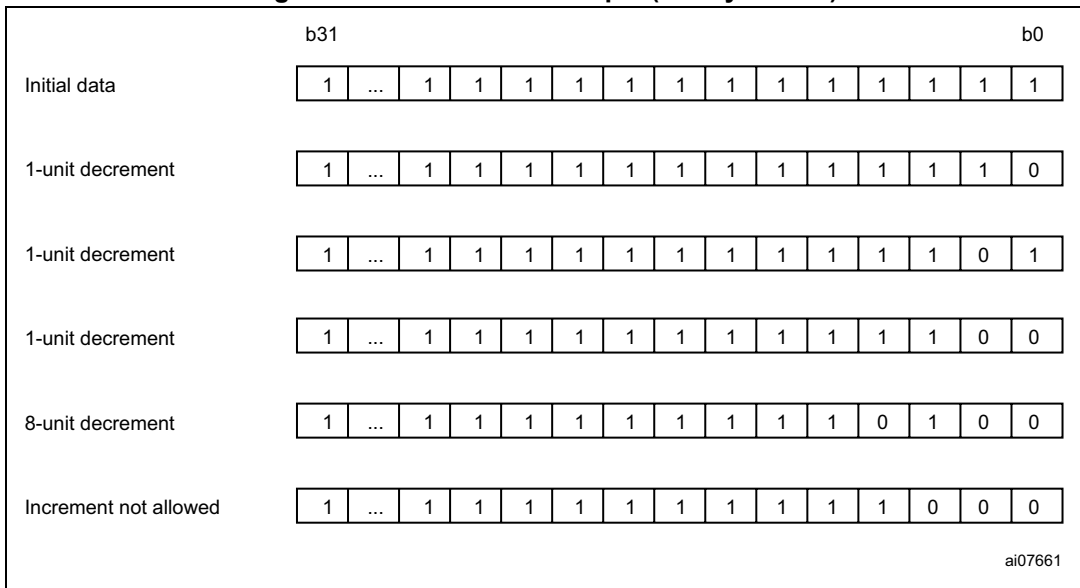
The two 32-bit binary counters are located at block addresses 5 and 6. The ST25TB02K uses dedicated logic that only allows the update of a counter if the new value is lower than the previous one. This feature allows the application to count down by steps of 1 or more. The initial value in Counter 5 is FFFF FFFEh and is FFFF FFFFh in Counter 6. When the reached value is 0000 0000h, the counter is empty and cannot be reloaded. For each counter 5 and 6, the update is done by issuing the Write_block command. The Write_block command writes the new 32-bit value to the counter block address. [Table 5](#) shows examples of how the counters operate.

The counter programming cycles are protected by automated antitearing logic. This function allows the counter value to be protected in case of power down within the programming cycle. In case of power down, the counter value is not updated and the previous value continues to be stored.

Table 5. Binary counter (addresses 5 to 6)

Block Address	32-bit block					Description
	MSB			LSB		
	b31	b24 b23	b16 b15	b8 b7	b0	
5	32-bit Boolean area					Count down counter
6	32-bit Boolean area					

Figure 14. Countdown example (binary format)



The counter with block address 6 controls the reload mode used to reset the resettable OTP area (addresses 0 to 4). Bits b₃₁ to b₂₁ act as an 11-bit Reload counter; whenever one of these 11 bits is updated, the ST25TB02K detects the change and adds an Erase cycle to the Write_block command for locations 0 to 4 (see the “[Resettable OTP area](#)” paragraph).

The Erase cycle remains active until a Power-off or a Select command is issued.

The ST25TB02K’s resettable OTP area can be reloaded up to 2 047 times (2¹¹-1).

4.3 EEPROM area

The 57 blocks between addresses 7 and 63 are EEPROM blocks of 32 bits each (228 bytes in total). (See [Table 6](#) for a map of the area.) These blocks can be accessed using the Read_block and Write_block commands. The Write_block command for the EEPROM area always includes an auto-erase cycle prior to the write cycle.

Blocks 7 to 15 can be write-protected. Write access is controlled by the 8 bits of the OTP_Lock_Reg located at block address 255 (see "[Section 4.4.1: OTP_Lock_Reg](#)" for details). Once protected, these blocks (7 to 15) cannot be unprotected.

Table 6. EEPROM (addresses 7 to 63)

Block Address	MSB		32-bit block				LSB		Description
	b31	b24 b23	b16 b15	b8 b7	b0				
7								Lockable EEPROM	
8									
9									
10									
11									
12									
13									
14									
15									
16								EEPROM	
...									
63									

4.4 System area

This area is used to modify the settings of the ST25TB02K. It contains 2 registers: OTP_Lock_Reg and ST Reserved. See [Table 7](#) for a map of this area.

A Write_block command in this area will not erase the previous contents. Selected bits can thus be set from 1 to 0. All bits previously at 0 remain unchanged. Once all the 32 bits of a block are at 0, the block is empty and cannot be updated any more.

Table 7. System area

Block Address	MSB		32-bit block				LSB		Description
	b31	b24 b23	b16 b15	b8 b7	b0				
255	OTP_Lock_Reg			ST reserved				OTP	

4.4.1 OTP_Lock_Reg

The 8 bits, b_{31} to b_{24} , of the System area (block address 255) are used as OTP_Lock_Reg bits in the ST25TB02K. They control the write access to the 9 EEPROM blocks with addresses 7 to 15 as follows:

- When b_{24} is at 0, blocks 7 and 8 are write-protected
- When b_{25} is at 0, block 9 is write-protected
- When b_{26} is at 0, block 10 is write-protected
- When b_{27} is at 0, block 11 is write-protected
- When b_{28} is at 0, block 12 is write-protected
- When b_{29} is at 0, block 13 is write-protected
- When b_{30} is at 0, block 14 is write-protected
- When b_{31} is at 0, block 15 is write-protected.

The OTP_Lock_Reg bits cannot be erased. Once write-protected, EEPROM blocks behave like ROM blocks and cannot be unprotected.

After any modification of the OTP_Lock_Reg bits, it is necessary to send a Select command with a valid Chip_ID to the ST25TB02K in order to load the block write protection into the logic.

5 ST25TB02K operation

All commands, data and CRC are transmitted to the ST25TB02K as 10-bit characters using ASK modulation. The start bit of the 10 bits, b_0 , is sent first. The command frame received by the ST25TB02K at the antenna is demodulated by the 10% ASK demodulator, and decoded by the internal logic. Prior to any operation, the ST25TB02K must have been selected by a Select command. Each frame transmitted to the ST25TB02K must start with a start of frame, followed by one or more data characters, two CRC bytes and the final end of frame. When an invalid frame is decoded by the ST25TB02K (wrong command or CRC error), the memory does not return any error code.

When a valid frame is received, the ST25TB02K may have to return data to the reader. In this case, data is returned using BPSK encoding, in the form of 10-bit characters framed by an SOF and an EOF. The transfer is ended by the ST25TB02K sending the 2 CRC bytes and the EOF.

6 ST25TB02K states

The ST25TB02K can be switched into different states. Depending on the current state of the ST25TB02K, its logic will only answer to specific commands. These states are mainly used during the anticollision sequence, to identify and to access the ST25TB02K in a very short time. The ST25TB02K provides 6 different states, as described in the following paragraphs and in [Figure 15](#).

6.1 Power-off state

The ST25TB02K is in Power-off state when the electromagnetic field around the tag is not strong enough. In this state, the ST25TB02K does not respond to any command.

6.2 Ready state

When the electromagnetic field is strong enough, the ST25TB02K enters the Ready state. After Power-up, the Chip_ID is initialized with a random value. The whole logic is reset and remains in this state until an Initiate() command is issued. Any other command will be ignored by the ST25TB02K.

6.3 Inventory state

The ST25TB02K switches from the Ready to the Inventory state after an Initiate() command has been issued. In Inventory state, the ST25TB02K will respond to any anticollision commands: Initiate(), Pcall16() and Slot_marker(), and then remain in the Inventory state. It will switch to the Selected state after a Select(Chip_ID) command is issued, if the Chip_ID in the command matches its own. If not, it will remain in Inventory state.

6.4 Selected state

In Selected state, the ST25TB02K is active and responds to all Read_block(), Write_block() and Get_UID() commands. When an ST25TB02K has entered the Selected state, it no longer responds to anticollision commands. So that the reader can access another tag, the ST25TB02K can be switched to the Deselected state by sending a Select(Chip_ID) with a Chip_ID that does not match its own, or it can be placed in Deactivated state by issuing a Completion() command. Only one ST25TB02K can be in Selected state at a time.

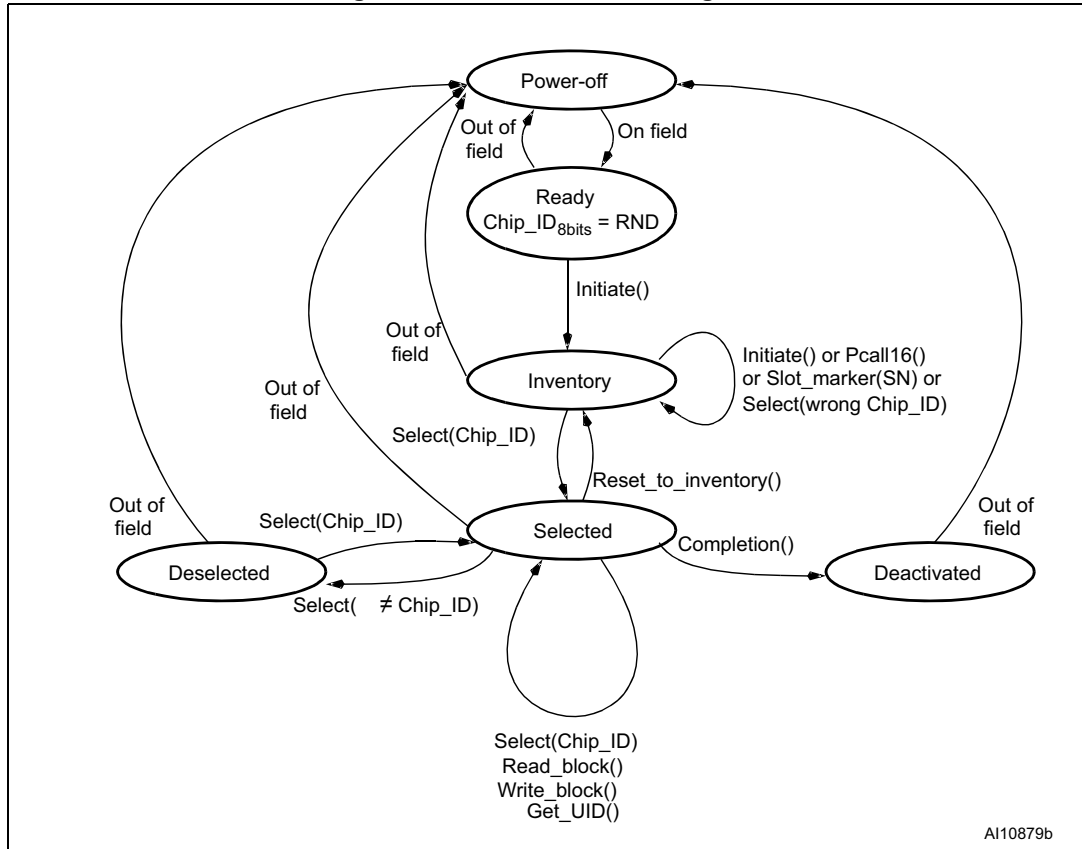
6.5 Deselected state

Once the ST25TB02K is in Deselected state, only a Select(Chip_ID) command with a Chip_ID matching its own can switch it back to Selected state. All other commands are ignored.

6.6 Deactivated state

When in this state, the ST25TB02K can only be turned off. All commands are ignored.

Figure 15. State transition diagram



7 Anticollision

The ST25TB02K provides an anticollision mechanism that searches for the Chip_ID of each device that is present in the reader field range. When known, the Chip_ID is used to select an ST25TB02K individually, and access its memory. The anticollision sequence is managed by the reader through a set of commands described in [Section 8: ST25TB02K commands](#):

- Initiate()
- Pcall16()
- Slot_marker().

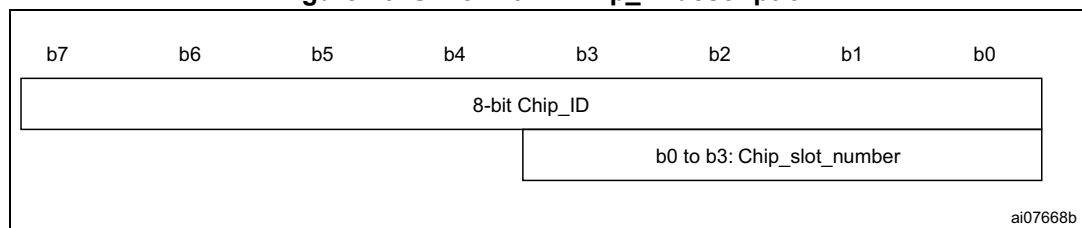
The reader is the master of the communication with one or more ST25TB02K device(s). It initiates the tag communication activity by issuing an Initiate(), Pcall16() or Slot_marker() command to prompt the ST25TB02K to answer. During the anticollision sequence, it might happen that two or more ST25TB02K devices respond simultaneously, so causing a collision. The command set allows the reader to handle the sequence, to separate ST25TB02K transmissions into different time slots. Once the anticollision sequence has completed, ST25TB02K communication is fully under the control of the reader, allowing only one ST25TB02K to transmit at a time.

The Anticollision scheme is based on the definition of time slots during which the ST25TB02K devices are invited to answer with minimum identification data: the Chip_ID. The number of slots is fixed at 16 for the Pcall16() command. For the Initiate() command, there is no slot and the ST25TB02K answers after the command is issued. ST25TB02K devices are allowed to answer only once during the anticollision sequence. Consequently, even if there are several ST25TB02K devices present in the reader field, there will probably be a slot in which only one ST25TB02K answers, allowing the reader to capture its Chip_ID. Using the Chip_ID, the reader can then establish a communication channel with the identified ST25TB02K. The purpose of the anticollision sequence is to allow the reader to select one ST25TB02K at a time.

The ST25TB02K is given an 8-bit Chip_ID value used by the reader to select only one among up to 256 tags present within its field range. The Chip_ID is initialized with a random value during the Ready state, or after an Initiate() command in the Inventory state.

The four least significant bits (b₀ to b₃) of the Chip_ID are also known as the Chip_slot_number. This 4-bit value is used by the Pcall16() and Slot_marker() commands during the anticollision sequence in the Inventory state.

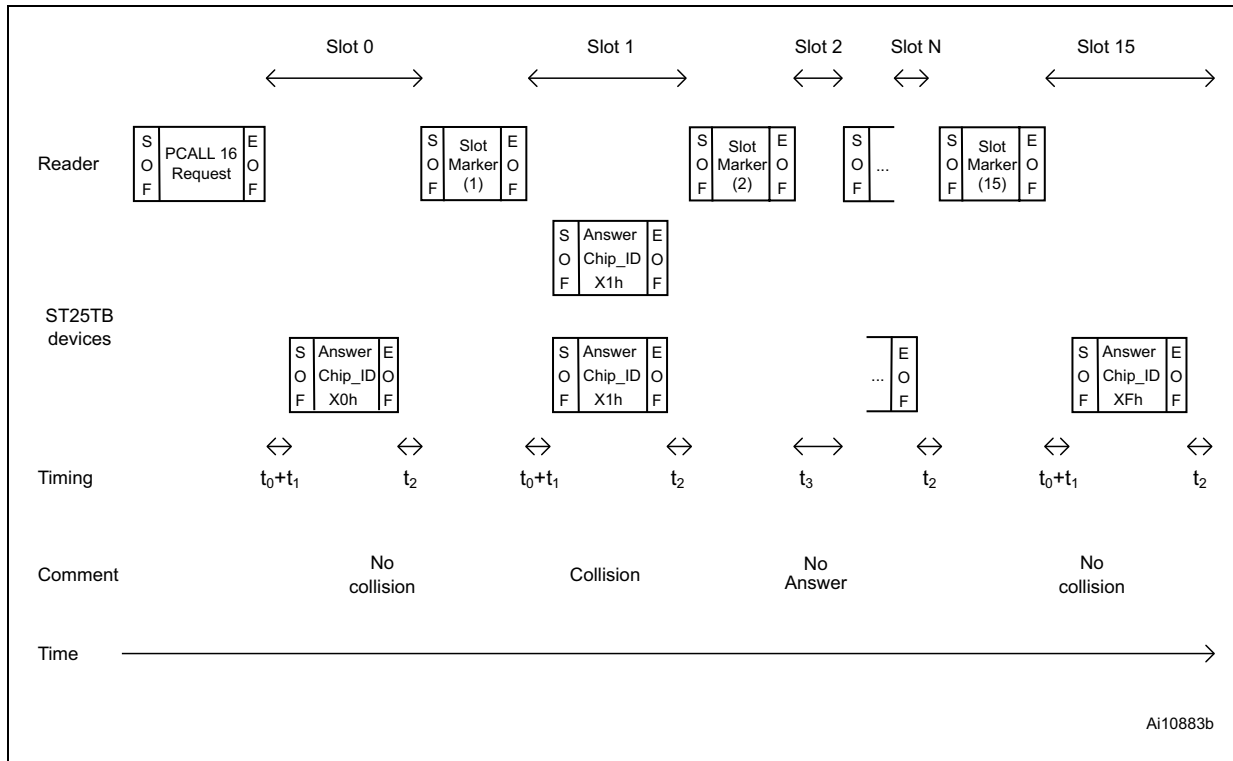
Figure 16. ST25TB02K Chip_ID description



Each time the ST25TB02K receives a Pcall16() command, the Chip_slot_number is given a new 4-bit random value. If the new value is 0000_b, the ST25TB02K returns its whole 8-bit Chip_ID in its answer to the Pcall16() command. The Pcall16() command is also used to define the slot number 0 of the anticollision sequence. When the ST25TB02K receives the Slot_marker(SN) command, it compares its Chip_slot_number with the Slot_number parameter (SN). If they match, the ST25TB02K returns its Chip_ID as a response to the

command. If they do not, the ST25TB02K does not answer. The Slot_marker(SN) command is used to define all the anticollision slot numbers from 1 to 15.

Figure 17. Description of a possible anticollision sequence



1. The value X in the answer Chip_ID means a random hexadecimal character from 0 to F.

7.1 Description of an anticollision sequence

The anticollision sequence is initiated by the Initiate() command which triggers all the ST25TB02K devices that are present in the reader field range, and that are in Inventory state. Only ST25TB02K devices in Inventory state will respond to the Pcall16() and Slot_marker(SN) anticollision commands.

A new ST25TB02K introduced in the field range during the anticollision sequence will not be taken into account as it will not respond to the Pcall16() or Slot_marker(SN) command (Ready state). To be considered during the anticollision sequence, it must have received the Initiate() command and entered the Inventory state.

Table 8 shows the elements of a standard anticollision sequence. (See Table 9 for an example.)

Table 8. Standard anticollision sequence

Step 1	Init:	Send Initiate(). – If no answer is detected, go to step1. – If only 1 answer is detected, select and access the ST25TB02K. After accessing the ST25TB02K, deselect the tag and go to step1. – If a collision (many answers) is detected, go to step2.
Step 2	Slot 0	Send Pcall16(). – If no answer or collision is detected, go to step3. – If 1 answer is detected, store the Chip_ID, Send Select() and go to step3.
Step 3	Slot 1	Send Slot_marker(1). – If no answer or collision is detected, go to step4. – If 1 answer is detected, store the Chip_ID, Send Select() and go to step4.
Step 4	Slot 2	Send Slot_marker(2). – If no answer or collision is detected, go to step5. – If 1 answer is detected, store the Chip_ID, Send Select() and go to step5.
Step N	Slop N	Send Slot_marker(3 up to 14) ... – If no answer or collision is detected, go to stepN+1. – If 1 answer is detected, store the Chip_ID, Send Select() and go to stepN+1.
Step 17	Slot 15	Send Slot_marker(15). – If no answer or collision is detected, go to step18. – If 1 answer is detected, store the Chip_ID, Send Select() and go to step18.
Step 18	-	All the slots have been generated and the Chip_ID values should be stored into the reader memory. Issue the Select(Chip_ID) command and access each identified ST25TB02K one by one. After accessing each ST25TB02K, switch them into Deselected or Deactivated state, depending on the application needs. – If collisions were detected between Step2 and Step17, go to Step2. – If no collision was detected between Step2 and Step17, go to Step1.

After each Slot_marker() command, there may be no answer, one or several answers from the ST25TB02K devices. The reader must handle all the cases and store all the Chip_IDs, correctly decoded. At the end of the anticollision sequence, after Slot_marker(15), the reader can start working with one ST25TB02K by issuing a Select() command containing the desired Chip_ID. If a collision is detected, the reader has to generate a new sequence in order to identify all unidentified ST25TB02K devices in the field. The anticollision sequence can stop when all ST25TB02K devices have been identified.

Table 9 gives an example of anticollision sequence, the gray cells highlight the fact that the related tags are not yet identified. When the tag is identified, the gray color changes to white.